Analyzing wide binary orbit stability with MOND theory and external field effects

by

Jart Koster

to obtain the degree of Bachelor of Science at the Delft University of Technology,
to be defended publicly on Friday July 12, 2024 at 11:00 AM.

Student number: 5636728
Thesis committee: Dr. P. M. Visser — Supervisor
Dr. S. W. H. Eijt — Supervisor
Dr. ir. W. G. M. Groenevelt
Dr. M. Blaauboer

Abstract

The rotational velocities of stars at the edges of galaxies are higher than predicted by Newtonian dynamics. One theory that is capable of explaining this anomaly is called MOdified Newtonian Dynamics (MOND). To further test this MOND theory, the wide binary test has been suggested. In the wide binary test, the stability of wide binary systems is tested. The MOND theory deviates significantly from the Newtonian theory when accelerations are smaller than $a_0 = 1.2 \cdot 10^{-10}$ m/s^2, which is present in wide binary systems. The MOND theory is non-linear, which introduces a non-trivial effect of the external field (EF) of the Milky Way on the wide binary orbit. To perform the wide binary test, an iterative particle mesh N-body code was used with initial conditions taken from the Gaia mission database. Additionally, the EF was modeled in three ways, where the Newton potential adjustment model was found to be the most realistic. The results include nine simulations, split up into three models and three different regimes. The models include the Newton model, the MOND without EF model, and the MOND with EF model. The regimes consist of the Newtonian regime ($a_{\text{internal}} > a_0$), the transition regime ($a_{\text{internal}} \approx a_0$), and the deep MOND regime ($a_{\text{internal}} < a_0$). In the Newton model, only the Newtonian regime gave stable orbits. The MOND without EF model gave stable orbits for all regimes, and the MOND with EF model gave stable orbits only for the Newtonian regime. The MOND without EF gave stable wide binary orbits, whereas the other models did not. If additional measurements detect these wide binary systems, then the MOND without EF model gains credibility. The orbits of the Newton model and the MOND with EF model were similar, with only a small difference. This is because the strength of the external field is $1.6 \cdot a_0$, which makes the external field dominant over the internal accelerations, making the simulation Newtonian-like. To determine which model is correct, additional long-term measurements of Gaia are required. Further research recommendations include improving the performance and accuracy of the code. This is done by better implementation of the fast Fourier transform, translating the code into a faster programming language than Python, using adaptive time steps, and reducing self-interaction. The most important extension to the model is to introduce the galactic tidal effect.

Contents

1 Introduction

1.1. Development of gravitational theories

The phenomenon of gravitation has been studied by humankind for over two thousand years. The earliest thoughts and observations date back to ancient Greece, where the Greek philosopher Aristotle developed some of the earliest ideas about gravitation. Aristotle thought that the speed at which an object falls is proportional to its mass. The ideas of Aristotle about gravity were dominant for nearly two millennia, but in Delft, around the year 1586, Simon Stevin conducted experiments that would test the ideas of Aristotle. Stevin dropped two lead spheres, one light, and one heavy, from the tower of the New Church. He observed that the two spheres reached the ground at the same time, which disproved the ideas of Aristotle. In 1590, Galileo Galilei conducted similar experiments and laid a strong experimental foundation for future scientists to build on. In 1687, building upon the groundwork of Galileo, Sir Isaac Newton published the revolutionizing work ”Philosophiæ Naturalis Principia Mathematica” (Mathematical Principles of Natural Philosophy), where he proposed the law of universal gravitation. The Newtonian theory of gravity proved highly successful for the next two centuries. During this period, observational measurements of the planets in the Solar System improved. In 1859, an anomaly in the orbital precession of Mercury was first noticed, which was inconsistent with Newtonian theory. Later in 1882, Simon Newcomb estimated the precession to be 43 arcseconds per century. In 1915, the theory of general relativity was proposed by Albert Einstein, which was able to explain the anomaly of Mercury's trajectory. The theory of general relativity predicted an extra precession of 42.9799 arcseconds per century[1]. Then in 1919, another verification of general relativity came through observational evidence of the gravitational deflection of starlight passing near the Sun. Since 1919, observational measurements of other galaxies have improved and another anomaly, inconsistent with general relativity, was discovered by Vera Rubin in the 1970s[2]. These observations which challenge general relativity are discussed next.

1.2. Galaxy rotation curve

During the second half of the twentieth century, measurements of stars in nearby disc galaxies improved. These measurements introduced a new anomaly. The stars in the outer regions of these galaxies showed a significantly higher rotational speed than was expected based on Newtonian gravity and general relativity. The rotation curve of a galaxy represents the orbital velocity of stars as a function of their distance from the galactic center. Based on Newtonian gravity and general relativity, these rotation curves should decline, but the observed rotation curves were constant or even increasing. This anomaly was not just a coincidence for a few disc galaxies but was found to be present in a majority of disc galaxies. In figure (1.1), an example rotation curve of the NGC 6503 spiral galaxy is shown[3].

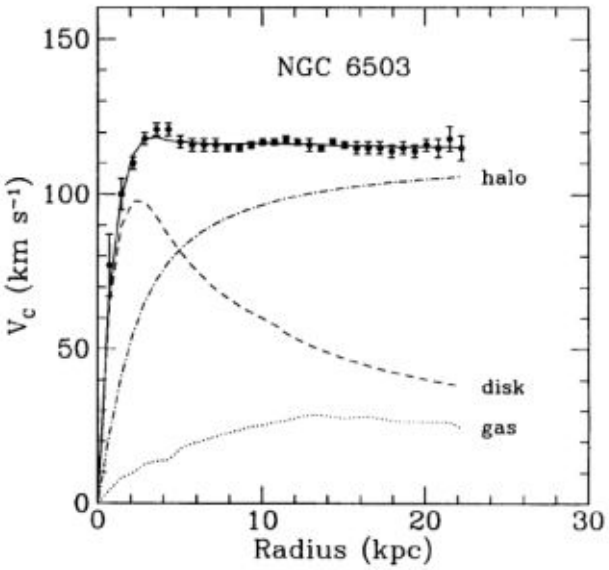


Figure 1.1: The radial velocity of the stars V_c against the radius in the dwarf spiral galaxy NGC 6503. The halo, disk, and gas lines show the contribution of the halo, disk, and gas to the total radial velocity. The halo refers to the dark matter in the galaxy, the disk to the other stars, and the gas to the interstellar gas. The total radial velocity line stays constant as the radius increases, which is shown by measurements of the radial velocity of stars[3].

The observations show a constant rotation curve, while the curve indicated with 'disk' shows a $1/\sqrt{r}$-decline as expected based on Newtonian gravity and general relativity. The line with the label 'gas' is the contribution that the gas in the galaxy has on the total rotational velocity of the stars. Adding the 'disk' and 'gas' lines, according to $v = \sqrt{v_{\text{disk}}^2 + v_{\text{gas}}^2}$, does not give the observed rotational velocity. This discrepancy led astronomers to develop new hypotheses to explain the unexpected behavior of the rotation curves. The most notable hypotheses are discussed next.

1.3. Dark matter or MOND

To explain the anomaly of the rotation curves, astronomers developed several new hypotheses to try to explain the constant rotation curves. The most accepted hypothesis by astronomers is the dark matter hypothesis. This hypothesis aims to explain the rotation curve anomaly by introducing a new form of matter, called dark matter, which is only able to interact gravitationally. Multiple forms of dark matter are proposed, which are known as Cold Dark Matter (CDM), Warm Dark Matter (WDM), and Hot Dark Matter (HDM). The temperature adjective refers to the velocity that the dark matter has, with cold having the least velocity. Under these types of dark matter, CDM is the most accepted. When taking these forms of matter into account in the equations of general relativity, it can explain numerous anomalies, including the rotation curve anomaly, but it still has some difficulties in other areas.

Another hypothesis was proposed by Mordehai Milgrom in 1983 as an alternative to the dark matter hypothesis. This new hypothesis is known as MOdified Newtonian Dynamics (MOND), which is primarily focused on explaining the rotation curve anomaly without needing to invoke dark matter. Instead of using the existing equations for gravity, MOND changes these equations in the low gravitation limit, thereby keeping Newtonian gravitation intact for larger accelerations than $a_0 = 1.2 \cdot 10^{-10}$ m/s^2. In the region around this turning point acceleration a_0, the transition from the Newtonian regime to the MOND regime proceeds smoothly. The theory of MOND, however, does break with general relativity. In MOND, the internal dynamics of a small subsystem depends on the external field, while in general relativity it is independent of the external field due to the Strong Equivalence Principle (SEP). Therefore, a majority of astronomers are skeptical about the MOND hypothesis, and they are also skeptical because MOND, just like dark matter, has difficulties in other areas.

Extensive research has been done to validate both hypotheses. The search for dark matter has been going on for multiple decades, but direct evidence of the existence of dark matter has not been found. For MOND, several methods have been proposed which should be able to test the MOND hypothesis. The method that will be examined in this research is the wide binary test, discussed in the

next section.

1.4. Wide binary test

In this research, the method for testing the MOND hypothesis is the wide binary test. A wide binary system is a system of two stars that orbit each other, but these stars are separated such that the mutual attraction is near to the MOND transition acceleration a_0. By modeling and simulating this system, the effects of MOND on the trajectory of the stars in the wide binary system can be analyzed. The trajectories can be compared to the Newtonian trajectories, and the effect of adding an external field can be analyzed. By using real data about wide binary systems, the stability of these systems can be tested in both the Newtonian and MOND cases.

This research consists of modeling, simulating, and analyzing wide binary systems based on a particle mesh method. In recent years, there have been many studies that either claimed to have found proof for MOND or stated that they had disproven MOND. An overview of this debate is given in the following subsection.

1.4.1 Scientific discussion

To start, in 1983 Mordehai Milgrom proposed an alternative theory to dark matter, called MOdified Newtonian Dynamics (MOND)[4]. The next year Felten pointed out that the MOND theory does not conserve momentum[5]. This led Milgrom and Bekenstein to develop a better MOND theory, derivable from a Lagrangian, which solved the problems from the original MOND theory brought up by Felten. In the following years, more research was performed on the effect of MOND on galaxies. In 1991 Begeman et al. studied the effect that dark matter and MOND had on the rotation curves of several galaxies. They concluded that MOND is the best phenomenological description of the rotation curve anomaly in galaxies [6]. Almost a decade later in 2002, Sanders and McGaugh came to a similar conclusion, because MOND was able to explain several aspects of the observed properties of bound gravitating systems, in addition to the remarkable rotation curves following from MOND[7].

In the 2010s, the wide binary test started to become a topic of discussion. The idea that systems with low acceleration environments could test the MOND hypothesis was discussed theoretically before the 2010s, but the explicit discussion of the wide binary test started in the 2010s. One of the first studies dedicated to wide binaries was from Hernandez et al. [8]. In this study, the opportunity to test the MOND hypothesis by analyzing wide binary systems was first proposed. After the study of Hernandez et al. the method of testing MOND by analyzing wide binary systems gained traction and led to more extensive studies. In the years after 2012, the Gaia mission from the European Space Agency started, which gathered high-precision data from more than a billion stars in the Milky Way, including wide binary systems. Later in 2018, Banik and Zhao used the data from the Gaia mission to test MOND[9]. Banik and Zhao did not draw hard conclusions about MOND but rather emphasized the importance of testing MOND through the wide binary test and the importance of more detailed wide binary data. In 2018 Pittordis and Sutherland also studied the wide binary test and concluded that it is a promising test for testing MOND, but that more wide binary data was needed[10]. In 2018, after the publications of Pittordis and Sutherland, the Data Release 2 (DR2) from the Gaia mission was released. In 2019 Pittordis and Sutherland used this new DR2 to conduct a similar study as the 2018 study, and this time, by the use of the more detailed data, they concluded that Newtonian dynamics was preferred over MOND if the MOND theory did not include an external field. These conclusions were indecisive when an external field was added. In the early 2020s, the studies published about the wide binary test had more hard conclusions about the value of the MOND theory. In 2023 Pittordis and Sutherland used the Early Data Release 3 (EDR3) from the Gaia mission to select a large sample of 73,159 wide binaries. Similar to their 2019 paper, they claimed that Newtonian gravity was preferred over the MOND theory, but that improved data and follow-up research could change this conclusion[11]. Then, in July 2023 Chae used the new Data Release 3 (DR3) from the Gaia mission to analyze wide binaries. Chae found that the MOND theory was strongly preferred over the Newtonian dynamics in the small acceleration regime[12]. Chae found an exceptionally strong significance of 10σ that MOND was preferred over Newtonian gravity. Later in November 2023, Banik et al. using the DR3 database, tested MOND with the wide binary test. Banik et al. found a completely contradictory result to the conclusions of Chae. Banik et al. found that Newtonian dynamics was preferred over MOND with a 19σ confidence, and excludes MOND at a 16σ confidence[13]. These contradictory results show that the MOND discussion

is still highly debated. Just over a month later in December 2023, Hernandez and Chae criticized the Banik et al. paper[14]. They claimed that the methodological shortcomings of the Banik et al. study were the reason that Banik et al. came to the wrong conclusion about Newtonian gravity being preferred over MOND.

The future of the wide binary test is full of new research because of the active discussion. The prospect of more and higher precision data from the Gaia Data Release 4, based on 66 months of data, looks promising for the wide binary test. For this DR4, scientists have to wait until at least the end of 2025. If the DR4 does not provide decisive data, then all eyes are on Data Release 5, which is planned to be released after 2030.

1.5. Research outline

This research will aim at testing the MOND theory through the wide binary test. It will do so by modeling, simulating, and analyzing wide binaries near the Sun. The research outline, given by chapter, is given below:

1. Introduction: The background information about MOND and the wide binary test is described.
2. Theory: The needed theory for developing the MOND theory is given, along with other theories related to MOND.
3. Method: The approach to modeling and simulating the wide binary test is outlined.
4. Initial conditions: Background information is given about how the initial conditions of the simulation were chosen. The data from the Gaia Data Release 3 and the physical interpretation of the data are discussed.
5. External field effect: The methods used for modeling the external field effect in the simulation are discussed.
6. Simulation results: The result of the simulation of wide binaries is analyzed.
7. Research recommendations: Improvements to the simulation are given for further research.
8. Conclusion: The findings of the simulation results are summarized and interpreted. The most important research recommendations are given.

2

Theory

2.1. Newtonian dynamics

Consider a system with N particles with masses $m_1, m_2, ..., m_N$ and positions $\boldsymbol{r}_1, \boldsymbol{r}_2, ..., \boldsymbol{r}_N$. According to Newtonian dynamics, the gravitational field $\boldsymbol{g}(\boldsymbol{r})$ is given by:

$$\boldsymbol{g}(\boldsymbol{r}) = \sum_{i=1}^{N} -\frac{Gm_i}{\|\boldsymbol{r} - \boldsymbol{r}_i\|^3} (\boldsymbol{r} - \boldsymbol{r}_i) \tag{2.1}$$

where N is the number of particles, G is the gravitational constant, m_i is the mass of particle i, and $\boldsymbol{r}_i$ is the position of particle i.

A shortcoming of equation (2.1) is that it considers point masses $m_1, m_2, ..., m_N$, which is an inaccurate model of a star because the mass of a star is not concentrated in a single point, but is distributed across the star's volume. This problem is solved by noticing that a non-point mass M is a sum of infinitesimally small point masses, which yields a gravitational field for one non-point mass M:

$$\boldsymbol{g}(\boldsymbol{r}) = \sum_{i=1}^{\infty} -\frac{G\, dm_i}{\|\boldsymbol{r} - \boldsymbol{r}_i\|^3} \cdot (\boldsymbol{r} - \boldsymbol{r}_i) \overset{(1)}{=} \iiint -G\rho(\boldsymbol{s}) \frac{(\boldsymbol{r} - \boldsymbol{s})}{\|\boldsymbol{r} - \boldsymbol{s}\|^3}\, d^3\boldsymbol{s} \tag{2.2}$$

where ρ is the mass density, and $dm_i = \rho d^3\boldsymbol{s}$ for all i is used in (1).

Applying the divergence operator to both sides gives:

$$\boldsymbol{\nabla} \cdot \boldsymbol{g}(\boldsymbol{r}) = \boldsymbol{\nabla} \cdot \iiint -G\rho(\boldsymbol{s}) \frac{(\boldsymbol{r} - \boldsymbol{s})}{\|\boldsymbol{r} - \boldsymbol{s}\|^3}\, d^3\boldsymbol{s} = -G \iiint \rho(\boldsymbol{s}) \boldsymbol{\nabla} \cdot \left(\frac{(\boldsymbol{r} - \boldsymbol{s})}{\|\boldsymbol{r} - \boldsymbol{s}\|^3} \right) d^3\boldsymbol{s} \tag{2.3}$$

and when using the identity

$$\boldsymbol{\nabla} \cdot \left(\frac{\boldsymbol{r}}{\|\boldsymbol{r}\|^3} \right) = 4\pi\delta(\boldsymbol{r}) \tag{2.4}$$

in equation (2.3), where $\delta(\boldsymbol{r})$ is the Dirac delta distribution, the following is obtained:

$$\boldsymbol{\nabla} \cdot \boldsymbol{g}(\boldsymbol{r}) = -4\pi G \iiint \rho(\boldsymbol{s}) \delta(\boldsymbol{r} - \boldsymbol{s})\, d^3\boldsymbol{s} \overset{(2)}{=} -4\pi G\rho(\boldsymbol{r}) \tag{2.5}$$

where in (2), the Dirac delta identity:

$$\iiint f(\boldsymbol{r}) \delta(\boldsymbol{r} - \boldsymbol{r}_0)\, d^3\boldsymbol{r} = f(\boldsymbol{r}_0) \tag{2.6}$$

has been used.

Equation (2.5) is also known as Gauss's law:

$$\boldsymbol{\nabla} \cdot \boldsymbol{g}(\boldsymbol{r}) = -4\pi G\rho(\boldsymbol{r}) \tag{2.7}$$

Equation (2.7) can be rewritten into Poisson's equation for Newtonian gravity by expressing the gravitational field $\boldsymbol{g}(\boldsymbol{r})$ in terms of a potential $\Phi_N(\boldsymbol{r})$, where $\boldsymbol{g}(\boldsymbol{r}) = -\boldsymbol{\nabla}\Phi_N(\boldsymbol{r})$.

Using this result in equation (2.7) yields Poisson's equation for Newtonian gravity:

$$\boldsymbol{\nabla}^2\Phi_N(\boldsymbol{r}) = 4\pi G\rho(\boldsymbol{r}) \tag{2.8}$$

Here $\boldsymbol{\nabla}^2 = \boldsymbol{\nabla}\cdot\boldsymbol{\nabla}$ is the Laplace operator. It is important to note that the Poisson equation has the boundary condition: $\boldsymbol{\nabla}\Phi_N = \boldsymbol{0}$ as $r \longrightarrow \infty$. This is because the gravitational force vanishes at infinity, so $\boldsymbol{\nabla}\Phi_N$ vanishes. Also note that $\boldsymbol{g} \propto \frac{1}{r^2}$, thus $\Phi_N \propto \frac{1}{r}$, which means that $\Phi =$ constant as $r \longrightarrow \infty$.

To describe the dynamics of the system, another equation is needed. This equation is known as Newton's second law and is given by:

$$\boldsymbol{F} = m\boldsymbol{a} \tag{2.9}$$

Poisson's equation for Newtonian gravity, equation (2.8), together with Newton's second law, equation (2.9), describes the dynamics of the system.

2.1.1 Lagrangian

Newtonian dynamics can be reformulated in terms of a Lagrangian density. The Lagrangian density formulation of Newtonian dynamics is given by:

$$\mathcal{L}(\Phi_N, \boldsymbol{\nabla}\Phi_N) = -\frac{1}{8\pi G}\left(\boldsymbol{\nabla}\Phi_N(\boldsymbol{r},t)\right)^2 - \rho(\boldsymbol{r},t)\Phi_N(\boldsymbol{r},t) \tag{2.10}$$

where Φ_N is the Newtonian potential, and ρ is the mass density.

In the case of Newtonian gravity, one of the Euler-Lagrange equations is given by:

$$\frac{\partial\mathcal{L}}{\partial\Phi_N} = \boldsymbol{\nabla}\cdot\left(\frac{\partial\mathcal{L}}{\partial\left(\boldsymbol{\nabla}\Phi_N\right)}\right) \tag{2.11}$$

Using the Lagrangian from equation (2.10) in the Euler-Lagrange equation (2.11), the following is obtained:

$$-\rho(\boldsymbol{r},t) = \boldsymbol{\nabla}\cdot\left(-\frac{1}{8\pi G}2\left(\boldsymbol{\nabla}\Phi_N\left(\boldsymbol{r},t\right)\right)\right) = \left(-\frac{1}{4\pi G}\boldsymbol{\nabla}^2\Phi_N\left(\boldsymbol{r},t\right)\right) \tag{2.12}$$

When multiplying both sides by $-4\pi G$, equation (2.12) reduces to the Poisson's equation for Newtonian gravity, given in equation (2.8).

2.2. Modified Newtonian dynamics

As already mentioned in the introduction, the rotational velocity of stars on the edge of the galaxy is greater than expected from Newtonian dynamics and general relativity. To account for these unexpected measurements, Mordehai Milgrom developed the theory of modified Newtonian dynamics, abbreviated as MOND. In MOND, the theoretical radial velocities of stars on the edge of the galaxy should increase to be in line with measurements. To do this, Milgrom initially postulated a change in Newton's second law, given by:

$$\boldsymbol{F} = m\mu\left(\frac{|\boldsymbol{a}|}{a_0}\right)\boldsymbol{a} \tag{2.13}$$

where μ is called the interpolation function, and where a_0 is a fundamental constant in the MOND theory, with a value $a_0 = 1.2\cdot 10^{-10}$m/s^2. The interpolation function is unknown, but it needs to have several properties in the high and low acceleration limits. These properties are given as follows:

$$\mu\left(\frac{|\boldsymbol{a}|}{a_0}\right) \longrightarrow 1, \text{ as } |\boldsymbol{a}| \gg a_0 \tag{2.14}$$

$$\mu\left(\frac{|\boldsymbol{a}|}{a_0}\right) \longrightarrow \frac{|\boldsymbol{a}|}{a_0}, \text{ as } |\boldsymbol{a}| \ll a_0 \tag{2.15}$$

The condition in equation (2.14) ensures that the high acceleration limit converges to the Newtonian dynamics regime, which is needed because Newtonian dynamics is an accurate model for $|\boldsymbol{a}| \gg a_0$. This

first approach, however, is problematic because it is not consistent with Newton's third law. This can be illustrated by imagining a 2-body system with masses m and M. Then using Newtonian dynamics for the force $\boldsymbol{F}$, equation (2.13) becomes:

$$m\mu\left(\frac{|\boldsymbol{a}_m|}{a_0}\right)|\boldsymbol{a}_m| = \frac{GmM}{|\boldsymbol{r}|^2} = M\mu\left(\frac{|\boldsymbol{a}_M|}{a_0}\right)|\boldsymbol{a}_M| \tag{2.16}$$

Newton's third law states that $m|\boldsymbol{a}_m| = M|\boldsymbol{a}_M|$, and this results in:

$$\mu\left(\frac{|\boldsymbol{a}_m|}{a_0}\right) = \mu\left(\frac{|\boldsymbol{a}_M|}{a_0}\right) \tag{2.17}$$

Thus the interpolation function μ needs to be constant, but this is inconsistent with the property from equation (2.15). To tackle this inconsistency, Milgrom and Bekenstein derived the MOND theory from a Lagrangian.

2.2.1 AQUAL

As discussed, Milgrom and Bekenstein derived the MOND theory from a Lagrangian, which is given by:

$$\mathcal{L}(\Phi, \boldsymbol{\nabla}\Phi) = -\frac{1}{8\pi G}a_0^2\mathscr{F}\left(\frac{|\boldsymbol{\nabla}\Phi|^2}{a_0^2}\right) - \rho(\boldsymbol{r},t)\Phi(\boldsymbol{r},t) \tag{2.18}$$

Following a similar approach as in section (2.1.1), first the Euler-Lagrange equation is used, giving the following:

$$-\rho(\boldsymbol{r},t) = -\frac{1}{8\pi G}a_0^2\boldsymbol{\nabla}\cdot\left(2\frac{\boldsymbol{\nabla}\Phi}{a_0^2}\mathscr{F}'\left(\frac{|\boldsymbol{\nabla}\Phi|^2}{a_0^2}\right)\right) = -\frac{1}{4\pi G}\boldsymbol{\nabla}\cdot\left(\mu\left(\frac{|\boldsymbol{\nabla}\Phi|}{a_0}\right)\boldsymbol{\nabla}\Phi\right) \tag{2.19}$$

where $\mu\left(\frac{|\boldsymbol{\nabla}\Phi|}{a_0}\right) = \mathscr{F}'\left(\frac{|\boldsymbol{\nabla}\Phi|^2}{a_0^2}\right)$.

By multiplying both sides of equation (2.19) by $-4\pi G$, the non-linear Poisson equation results:

$$\boldsymbol{\nabla}\cdot\left(\mu\left(\frac{|\boldsymbol{\nabla}\Phi|}{a_0}\right)\boldsymbol{\nabla}\Phi\right) = 4\pi G\rho(\boldsymbol{r},t) \tag{2.20}$$

Equation (2.20) also needs boundary conditions, which are given by: $\boldsymbol{\nabla}\Phi \longrightarrow \boldsymbol{0}$ as $r \longrightarrow \infty$.

When rewriting the right hand side of equation (2.20) using the standard Poisson equation (2.8), the following results:

$$\boldsymbol{\nabla}\cdot\left(\mu\left(\frac{|\boldsymbol{\nabla}\Phi|}{a_0}\right)\boldsymbol{\nabla}\Phi\right) = \boldsymbol{\nabla}^2\Phi_N \tag{2.21}$$

which can be written into:

$$\boldsymbol{\nabla}\cdot\left(\mu\left(\frac{|\boldsymbol{\nabla}\Phi|}{a_0}\right)\boldsymbol{\nabla}\Phi - \boldsymbol{\nabla}\Phi_N\right) = 0 \tag{2.22}$$

By noting that the divergence of the curl of a vector field is always zero, the factor between brackets can be written as the curl of some vector field. Thus the following vector field $\boldsymbol{H}$ can be introduced:

$$\mu\left(\frac{|\boldsymbol{\nabla}\Phi|}{a_0}\right)\boldsymbol{\nabla}\Phi - \boldsymbol{\nabla}\Phi_N = \boldsymbol{\nabla}\times\boldsymbol{H} \tag{2.23}$$

which can be written in terms of the gravitational fields:

$$\mu\left(\frac{g}{a_0}\right)\boldsymbol{g} = \boldsymbol{g}_N + \boldsymbol{\nabla}\times\boldsymbol{H} \tag{2.24}$$

For a large radius such that the local mass distribution can be approximated as a point mass with total mass M, the following results:

$$\mu\left(\frac{g}{a_0}\right)\boldsymbol{g} = \frac{-GM\boldsymbol{r}}{r^3} + \boldsymbol{\nabla}\times\boldsymbol{H}. \tag{2.25}$$

Now the r dependence of $\boldsymbol{\nabla} \times \boldsymbol{H}$ will be analyzed. $\boldsymbol{H}$ can be seen as a vector potential, with $\boldsymbol{u} = \boldsymbol{\nabla} \times \boldsymbol{H}$. Then $\boldsymbol{H}$ can be written as:

$$\boldsymbol{H} = \frac{1}{4\pi} \int \frac{\boldsymbol{\nabla} \times \boldsymbol{u}\left(\boldsymbol{r}'\right)}{|\boldsymbol{r} - \boldsymbol{r}'|} \mathrm{d}^3 r' \tag{2.26}$$

Then $\boldsymbol{u}$ can be multipole expanded. Milgrom and Bekenstein proved that the monopole term vanishes. This leaves the dipole term as the next lowest contributing multipole term, which gives $\boldsymbol{u} \propto r^{-3}$, so $\boldsymbol{\nabla} \times \boldsymbol{H} \propto r^{-3}$. Thus when using this result in equation (2.25), the following results:

$$\mu\left(\frac{g}{a_0}\right) \boldsymbol{g} = \frac{-GM\boldsymbol{r}}{r^3} + \boldsymbol{O}\left(r^{-3}\right) \tag{2.27}$$

Now as the radius is large, the system is in the deep MOND regime, where $\mu(x) = x$, giving:

$$\boldsymbol{g} = -(MGa_0)^{1/2} \boldsymbol{r}/r^2 + \boldsymbol{O}\left(r^{-2}\right) \tag{2.28}$$

and thus in the limit of $r \longrightarrow \infty$:

$$\Phi = (MGa_0)^{1/2} \ln\left(r/r_0\right) + O\left(r^{-1}\right) \tag{2.29}$$

for some arbitrary radius r_0. Note that, as $r \longrightarrow \infty$, that $\Phi \longrightarrow \infty$, so all systems are bounded in deep MOND. Now, several important physical properties will be explored, starting with the conservation of momentum.

Conservation of momentum

Consider a volume α which encloses all mass in the considered system, and such that ρ vanishes at the surface of α. Then the total momentum in α is given by:

$$\boldsymbol{P}(t) = \int_\alpha \rho(\boldsymbol{r}, t) \boldsymbol{v}(\boldsymbol{r}, t) \mathrm{d}^3 r \tag{2.30}$$

The change in total momentum over time is thus given by:

$$\begin{aligned}
\dot{\boldsymbol{P}}(t) &= \int_\alpha \frac{\mathrm{d}}{\mathrm{d}t} \left(\rho(\boldsymbol{r}, t) \boldsymbol{v}(\boldsymbol{r}, t)\right) \mathrm{d}^3 r = \int_\alpha \frac{\mathrm{d}}{\mathrm{d}t} \left(\rho(\boldsymbol{r}, t)\right) \boldsymbol{v}(\boldsymbol{r}, t) + \frac{\mathrm{d}}{\mathrm{d}t} \left(\boldsymbol{v}(\boldsymbol{r}, t)\right) \rho(\boldsymbol{r}, t) \mathrm{d}^3 r \\
&= \int_\alpha \left(\frac{\partial \rho(\boldsymbol{r}, t)}{\partial t} + \frac{\partial \rho(\boldsymbol{r}, t)}{\partial \boldsymbol{r}} \cdot \frac{\partial \boldsymbol{r}}{\partial t}\right) \boldsymbol{v}(\boldsymbol{r}, t) + \left(\frac{\partial \boldsymbol{v}(\boldsymbol{r}, t)}{\partial t} + \frac{\partial \boldsymbol{v}(\boldsymbol{r}, t)}{\partial \boldsymbol{r}} \cdot \frac{\partial \boldsymbol{r}}{\partial t}\right) \rho(\boldsymbol{r}, t) \mathrm{d}^3 r \\
&= \int_\alpha \frac{\partial \rho(\boldsymbol{r}, t)}{\partial t} \boldsymbol{v}(\boldsymbol{r}, t) + \boldsymbol{\nabla} \left(\rho(\boldsymbol{r}, t)\right) \cdot \boldsymbol{v}(\boldsymbol{r}, t) \boldsymbol{v}(\boldsymbol{r}, t) + \frac{\partial \boldsymbol{v}(\boldsymbol{r}, t)}{\partial t} \rho(\boldsymbol{r}, t) + \boldsymbol{\nabla} \left(\boldsymbol{v}(\boldsymbol{r}, t)\right) \cdot \boldsymbol{v}(\boldsymbol{r}, t) \rho(\boldsymbol{r}, t) \mathrm{d}^3 r
\end{aligned} \tag{2.31}$$

For the first term, the continuity equation is needed, which is given by:

$$\frac{\partial \rho(\boldsymbol{r}, t)}{\partial t} = -\boldsymbol{\nabla} \cdot \left(\rho(\boldsymbol{r}, t) \boldsymbol{v}(\boldsymbol{r}, t)\right) \tag{2.32}$$

By substituting equation (2.32) in equation (2.31), the first term can be replaced:

$$\begin{aligned}
\int_\alpha \frac{\partial \rho(\boldsymbol{r}, t)}{\partial t} \boldsymbol{v}(\boldsymbol{r}, t) \mathrm{d}^3 r &= -\int_\alpha \boldsymbol{\nabla} \cdot \left(\rho(\boldsymbol{r}, t) \boldsymbol{v}(\boldsymbol{r}, t)\right) \boldsymbol{v}(\boldsymbol{r}, t) \mathrm{d}^3 r \\
&= \int_\alpha -\boldsymbol{\nabla} \left(\rho(\boldsymbol{r}, t)\right) \cdot \boldsymbol{v}(\boldsymbol{r}, t) \boldsymbol{v}(\boldsymbol{r}, t) - \boldsymbol{\nabla} \cdot \left(\boldsymbol{v}(\boldsymbol{r}, t)\right) \rho(\boldsymbol{r}, t) \boldsymbol{v}(\boldsymbol{r}, t) \mathrm{d}^3 r
\end{aligned} \tag{2.33}$$

Now using equation (2.33) in equation (2.31), the following results:

$$\begin{aligned}
\dot{\boldsymbol{P}}(t) = \int_\alpha & -\boldsymbol{\nabla} \left(\rho(\boldsymbol{r}, t)\right) \cdot \boldsymbol{v}(\boldsymbol{r}, t) \boldsymbol{v}(\boldsymbol{r}, t) - \boldsymbol{\nabla} \cdot \left(\boldsymbol{v}(\boldsymbol{r}, t)\right) \rho(\boldsymbol{r}, t) \boldsymbol{v}(\boldsymbol{r}, t) \\
& + \boldsymbol{\nabla} \left(\rho(\boldsymbol{r}, t)\right) \cdot \boldsymbol{v}(\boldsymbol{r}, t) \boldsymbol{v}(\boldsymbol{r}, t) + \frac{\partial \boldsymbol{v}(\boldsymbol{r}, t)}{\partial t} \rho(\boldsymbol{r}, t) + \boldsymbol{\nabla} \left(\boldsymbol{v}(\boldsymbol{r}, t)\right) \cdot \boldsymbol{v}(\boldsymbol{r}, t) \rho(\boldsymbol{r}, t) \mathrm{d}^3 r \\
= \int_\alpha & \frac{\partial \boldsymbol{v}(\boldsymbol{r}, t)}{\partial t} \rho(\boldsymbol{r}, t) \mathrm{d}^3 r
\end{aligned} \tag{2.34}$$

and now substituting $\frac{\partial \boldsymbol{v}(\boldsymbol{r},t)}{\partial t} = \boldsymbol{g} = -\boldsymbol{\nabla}\Phi$, and by substituting $\rho(\boldsymbol{r},t)$ from equation (2.20) into equation (2.34), the following results:

$$\dot{\boldsymbol{P}}(t) = -\frac{1}{4\pi G}\int_{\alpha} \boldsymbol{\nabla}\Phi\boldsymbol{\nabla}\cdot\left(\mu\left(\frac{|\boldsymbol{\nabla}\Phi|}{a_0}\right)\boldsymbol{\nabla}\Phi\right)\mathrm{d}^3r \tag{2.35}$$

Using integration by parts, the following results:

$$4\pi G\dot{\boldsymbol{P}}(t) = -\int_{\alpha}\boldsymbol{\nabla}\left(\boldsymbol{\nabla}\Phi\cdot\left(\mu\left(\frac{|\boldsymbol{\nabla}\Phi|}{a_0}\right)\boldsymbol{\nabla}\Phi\right)\right)\mathrm{d}^3r + \int_{\alpha}\mu\left(\frac{|\boldsymbol{\nabla}\Phi|}{a_0}\right)\boldsymbol{\nabla}\Phi\boldsymbol{\nabla}^2\Phi\mathrm{d}^3r \tag{2.36}$$

By now noting that:

$$\boldsymbol{\nabla}\mathscr{F}\left(\frac{|\boldsymbol{\nabla}\Phi|^2}{a_0^2}\right) = \mathscr{F}'\left(\frac{|\boldsymbol{\nabla}\Phi|^2}{a_0^2}\right)\boldsymbol{\nabla}\left(\frac{\boldsymbol{\nabla}\Phi}{a_0}\right)^2 = \mathscr{F}'\left(\frac{|\boldsymbol{\nabla}\Phi|^2}{a_0^2}\right)\frac{1}{a_0^2}2\boldsymbol{\nabla}\Phi\boldsymbol{\nabla}^2\Phi \tag{2.37}$$

which is equal to, when substituting $\mu(\boldsymbol{\nabla}\Phi/a_0) = \mathscr{F}'((|\boldsymbol{\nabla}\Phi|/a_0)^2)$:

$$\mu\left(\frac{|\boldsymbol{\nabla}\Phi|}{a_0}\right)\boldsymbol{\nabla}\Phi\boldsymbol{\nabla}^2\Phi = \frac{a_0^2}{2}\boldsymbol{\nabla}\mathscr{F}\left(\frac{|\boldsymbol{\nabla}\Phi|^2}{a_0^2}\right) \tag{2.38}$$

By now plugging the result from equation (2.38) into equation (2.36), the following results:

$$4\pi G\dot{\boldsymbol{P}}(t) = -\int_{\alpha}\boldsymbol{\nabla}\left(\boldsymbol{\nabla}\Phi\cdot\left(\mu\left(\frac{|\boldsymbol{\nabla}\Phi|}{a_0}\right)\boldsymbol{\nabla}\Phi\right)\right)\mathrm{d}^3r + \int_{\alpha}\frac{a_0^2}{2}\boldsymbol{\nabla}\mathscr{F}\left(\frac{|\boldsymbol{\nabla}\Phi|^2}{a_0^2}\right)\mathrm{d}^3r \tag{2.39}$$

and by using the gradient theorem (proof in appendix), the result is:

$$4\pi G\dot{\boldsymbol{P}}(t) = -\int_{\partial\alpha}\mu\left(\frac{|\boldsymbol{\nabla}\Phi|}{a_0}\right)\boldsymbol{\nabla}\Phi\boldsymbol{\nabla}\Phi\cdot\mathrm{d}\boldsymbol{A} + \frac{a_0^2}{2}\int_{\partial\alpha}\mathscr{F}\left(\frac{|\boldsymbol{\nabla}\Phi|^2}{a_0^2}\right)\mathrm{d}\boldsymbol{A} \tag{2.40}$$

The integrand for the first integral decreases like $\propto r^{-3}$, as $\mu\left(\frac{|\boldsymbol{\nabla}\Phi|}{a_0}\right)\boldsymbol{\nabla}\Phi = \boldsymbol{g}_N + \boldsymbol{O}(r^{-3}) \propto r^{-2}$ equation (2.27) and $\boldsymbol{\nabla}\Phi = \boldsymbol{g} \propto r^{-1}$ equation (2.28). Therefore the first integral vanishes as $r \longrightarrow \infty$.

Now the second integral in equation (2.40) needs to be analyzed. If the standard interpolation function $\mu(x) = 1/(\sqrt{1+1/x^2})$ is used, then the following results:

$$\begin{aligned}\mathscr{F}'\left(x^2\right) &= \mu(x) = \frac{1}{\sqrt{1+\frac{1}{x^2}}}\\ \mathscr{F}\left(x^2\right) &= \frac{\ln\left|\sqrt{1+\frac{1}{x^2}}-1\right| - \ln\left|\sqrt{1+\frac{1}{x^2}}+1\right|}{2} + x^2\sqrt{1+\frac{1}{x^2}}\end{aligned} \tag{2.41}$$

If then a Taylor series is made of $\mathscr{F}\left(x^2\right)$ around $x = 0$, since $\boldsymbol{\nabla}\Phi \longrightarrow 0$ as $r \longrightarrow \infty$, then the following Taylor series results:

$$\frac{2}{3}x^3 - \frac{1}{5}x^5 + \frac{3}{28}x^7 - \frac{5}{72}x^9 + \boldsymbol{O}\left(x^{11}\right) \tag{2.42}$$

from which can be seen that the integrand in the second integral $\mathscr{F}\left(\frac{|\boldsymbol{\nabla}\Phi|^2}{a_0^2}\right) \propto |\boldsymbol{\nabla}\Phi|^3$, and from equation (2.28) $|\boldsymbol{\nabla}\Phi| \propto r^{-1}$. Therefore $\mathscr{F}\left(\frac{|\boldsymbol{\nabla}\Phi|^2}{a_0^2}\right) \propto r^{-3}$, and thus the second integral also vanishes as $r \longrightarrow \infty$. Thus the result is:

$$\dot{\boldsymbol{P}}(t) = 0 \tag{2.43}$$

which proves the conservation of momentum.

Conservation of energy

The next important physical property is the conservation of energy. The total energy in an isolated system with a volume α enclosing all mass is given by:

$$E = -\int_\alpha \mathcal{L}(\Phi, \boldsymbol{\nabla}\Phi)\mathrm{d}^3 r + E_k = -L + E_k \tag{2.44}$$

where

$$E_k = \frac{1}{2}\int_\alpha \rho v^2 \mathrm{d}^3 r \tag{2.45}$$

with $v = |\boldsymbol{v}|$. This results in

$$E = \int_\alpha \frac{1}{2}\rho v^2 + \frac{1}{8\pi G}a_0^2 \mathscr{F}\left(\frac{|\boldsymbol{\nabla}\Phi|^2}{a_0^2}\right) + \rho(\boldsymbol{r}, t)\Phi(\boldsymbol{r}, t)\mathrm{d}^3 r \tag{2.46}$$

Now taking the time derivative of E results in:

$$\begin{aligned} \dot{E} &= -\frac{\mathrm{d}L}{\mathrm{d}t} + \frac{\mathrm{d}E_k}{\mathrm{d}t} \\ &= -\frac{\delta L}{\delta \Phi}\frac{\delta \Phi}{\delta t} - \frac{\delta L}{\delta \rho}\frac{\delta \rho}{\delta t} + \frac{\mathrm{d}E_k}{\mathrm{d}t} \end{aligned} \tag{2.47}$$

Starting with the first term gives:

$$\begin{aligned} -\frac{\delta L}{\delta \Phi}\frac{\delta \Phi}{\delta t} &= -\int_\alpha \left(\rho + \frac{a_0^2}{8\pi G}\left(\frac{\partial \mathscr{F}}{\partial \Phi} - \boldsymbol{\nabla}\cdot\frac{\partial \mathscr{F}}{\partial \boldsymbol{\nabla}\Phi}\right)\right)\frac{\partial \Phi}{\partial t}\mathrm{d}^3 r \\ &= -\int_\alpha \left(\rho + \frac{1}{4\pi G}\left(-\boldsymbol{\nabla}\cdot\left(\mu\left(\frac{|\boldsymbol{\nabla}\Phi|}{a_0}\right)\boldsymbol{\nabla}\Phi\right)\right)\right)\frac{\partial \Phi}{\partial t}\mathrm{d}^3 r \\ &= -\int_\alpha (\rho - \rho)\frac{\partial \Phi}{\partial t}\mathrm{d}^3 r = 0 \end{aligned} \tag{2.48}$$

Here the functional derivative has been used, which is:

$$\frac{\delta L}{\delta \Phi} = \rho + \frac{a_0^2}{8\pi G}\left(\frac{\partial \mathscr{F}}{\partial \Phi} - \boldsymbol{\nabla}\cdot\frac{\partial \mathscr{F}}{\partial \boldsymbol{\nabla}\Phi}\right) \tag{2.49}$$

Continuing with the second term gives:

$$-\frac{\delta L}{\delta \rho}\frac{\delta \rho}{\delta t} = \int_\alpha \Phi\frac{\partial \rho}{\partial t}\mathrm{d}^3 r \tag{2.50}$$

and the third term gives:

$$\begin{aligned} \frac{\mathrm{d}E_k}{\mathrm{d}t} &= \frac{1}{2}\int_\alpha \frac{\mathrm{d}\rho}{\mathrm{d}t}v^2 + \frac{\mathrm{d}v^2}{\mathrm{d}t}\rho\mathrm{d}^3 r \\ &= \int_\alpha \frac{\mathrm{d}\boldsymbol{v}}{\mathrm{d}t}\cdot\boldsymbol{v}\rho\mathrm{d}^3 r \\ &= \int_\alpha -\boldsymbol{\nabla}\Phi\cdot\boldsymbol{v}\rho\mathrm{d}^3 r \\ &= -\int_\alpha \boldsymbol{\nabla}\cdot(\Phi\boldsymbol{v}\rho) - \boldsymbol{\nabla}\cdot(\boldsymbol{v}\rho)\,\Phi\mathrm{d}^3 r \\ &= -\int_\alpha \frac{\partial \rho}{\partial t}\Phi\mathrm{d}^3 r \end{aligned} \tag{2.51}$$

where $\frac{\mathrm{d}\rho}{\mathrm{d}t} = 0$ because there is no change in mass according to Milgrom and Bekenstein, $\frac{\mathrm{d}\boldsymbol{v}}{\mathrm{d}t} = -\boldsymbol{\nabla}\Phi$, and the term $\boldsymbol{\nabla}\cdot(\Phi\boldsymbol{v}\rho)$ vanishes because $\Phi\boldsymbol{v}\rho = 0$ on the boundary of α, and where the continuity equation (2.32) has been used in the last line.

Combining the three terms gives:

$$\dot{E} = 0 + \int_\alpha \Phi\frac{\partial \rho}{\partial t}\mathrm{d}^3 r - \int_\alpha \frac{\partial \rho}{\partial t}\Phi\mathrm{d}^3 r = 0. \tag{2.52}$$

Thus there is conservation of energy.

2.3. Tully-Fisher relation

The Tully-Fisher relation is an empirical observation about the relation between the baryonic mass in a galaxy and the rotational velocities of the stars. The origins of this empirical relation date back to 1977, when Tully and Fisher [15] found a correlation between the global neutral hydrogen line profile width, the 21 cm hydrogen emission line width, and the absolute optical magnitude of galaxies. In this correlation, the magnitude of galaxies can be related to the mass of the galaxy through empirical mass-luminosity relations. The width of the 21 cm hydrogen emission line can be related to the rotational velocity at the edges of the galaxy by the Doppler shift. Due to the Doppler shift, the 21 cm line gets widened in the red and blue direction because the stars and gas at either end of the galaxy move in opposite directions, creating a blue and red shift. The 21 cm line also gets shifted by the general movement of the whole galaxy. Torres-Flores et al. found, using the above-mentioned relations, that the baryonic mass is proportional to the rotational velocity to the power of 3.64 ± 0.28 [16]. Thus the proportionality $M \propto v^{3.64\pm0.28}$ was found.

Next will be shown that the Tully-Fisher relation found by Torres-Flores et al. supports MOND. To show this, the non-linear Poisson equation is needed:

$$\boldsymbol{\nabla} \cdot \left(\mu \left(\frac{|\boldsymbol{\nabla}\Phi|}{a_0} \right) \boldsymbol{\nabla}\Phi \right) = 4\pi G\rho \tag{2.53}$$

In the deep MOND regime, $\mu\left(\frac{|\boldsymbol{\nabla}\Phi|}{a_0}\right) = \frac{|\boldsymbol{\nabla}\Phi|}{a_0}$, and thus the MOND Poisson equation becomes:

$$\boldsymbol{\nabla} \cdot (|\boldsymbol{\nabla}\Phi|\boldsymbol{\nabla}\Phi) = 4\pi a_0 G\rho \tag{2.54}$$

Now when integrating over some spherical volume V with some radius R and assuming that the mass distribution ρ is spherically symmetric, the following results:

$$\iiint_V \boldsymbol{\nabla} \cdot (|\boldsymbol{\nabla}\Phi|\boldsymbol{\nabla}\Phi)\,\mathrm{d}V = \iiint_V 4\pi a_0 G\rho \mathrm{d}V = 4\pi a_0 G \iiint_V \rho \mathrm{d}V \tag{2.55}$$

with the right-hand side integral reducing to the total mass M inside V. Using the divergence theorem on the left-hand side results in the following:

$$\oiint_S (|\boldsymbol{\nabla}\Phi|\boldsymbol{\nabla}\Phi) \cdot \hat{\boldsymbol{r}} \mathrm{d}S = \iiint_V \boldsymbol{\nabla} \cdot (|\boldsymbol{\nabla}\Phi|\boldsymbol{\nabla}\Phi)\,\mathrm{d}V = 4\pi a_0 GM \tag{2.56}$$

where $S = \partial V$ is the surface of the volume V. The mass distribution is assumed to be spherically symmetric and the surface S is spherical. Therefore $\boldsymbol{\nabla}\Phi$ points in the $\hat{\boldsymbol{r}}$ direction, and the absolute value of $\boldsymbol{\nabla}\Phi$ is constant at the surface S, with a value g_{MOND}. This results in the following:

$$\oiint_S (|\boldsymbol{\nabla}\Phi|\boldsymbol{\nabla}\Phi) \cdot \hat{\boldsymbol{r}} \mathrm{d}S = 4\pi R^2 g_{\mathsf{MOND}}^2 = 4\pi a_0 GM \tag{2.57}$$

which simplifies to:

$$g_{\mathsf{MOND}}^2 = \frac{a_0 GM}{R^2} \tag{2.58}$$

which gives:

$$g_{\mathsf{MOND}} = \sqrt{\frac{a_0 GM}{R^2}} \tag{2.59}$$

By setting the MOND acceleration equal to the centrifugal force, the following follows:

$$\frac{v_\infty^2}{R} = \sqrt{\frac{a_0 GM}{R^2}} \tag{2.60}$$

where v_∞ is the orbital velocity at the edges of the galaxy. Continuing gives:

$$v_\infty^2 = \sqrt{a_0 GM} \tag{2.61}$$

which gives the desired equation that can explain the empirical Tully-Fisher relation:

$$v_\infty^4 = a_0 GM \tag{2.62}$$

where we see that $M \propto v^4$ in the deep MOND regime. The MOND theory thus not only can explain the anomaly in the rotation curves but can also explain empirical observations like the Tully-Fisher relation.

3

Method

In 2023, Joost de Nijs developed a three-dimensional iterative particle mesh code to solve for the non-linear Poisson equation, given by equation (2.20). This chapter will explain the methods used for the simulations by starting with the simpler Newtonian particle mesh method.

3.1. Newtonian particle mesh

Consider a system with N particles. To solve for the motion of these N particles, the individual force on each of the particles can be calculated by adding the gravitational field of all other $N-1$ particles. The computation time, however, scales like O(N^2). Another way to solve for the motion of the particles is to solve for the potential. To solve for the potential Φ, the Poisson equation is needed, which is given by:

$$\boldsymbol{\nabla}^2\Phi = 4\pi G\rho \tag{3.1}$$

with boundary condition:

$$|\boldsymbol{\nabla}\Phi| \longrightarrow 0 \qquad \text{for } |\boldsymbol{r}| \longrightarrow \infty \tag{3.2}$$

To solve this equation, both the time and space are discretized. The time is discretized such that there is a constant time step Δt. The cubic space (L^3) is discretized on a grid, made up of equally sized cubic cells (ΔL^3), with each having a corresponding grid point on the grid.

3.1.1 Mass distribution

The first step for calculating the potential is to distribute the mass of the N particles among the cells, which is equivalent to assigning masses to the grid points. If the mass of a particle is just assigned to its cells grid point, then the resolution is low, since a particle close to one of the corners of the cell gives the same result as a particle in the opposite corner of the cell. To increase the resolution, another mass distribution technique is used which allows masses to be placed in between grid points. There are several interpolation methods to do this like: nearest grid point, cloud-in-cell, and using a Gaussian weight function. The simulations in this research used the Gaussian weight function. The interpolation function is given by:

$$M\left((i,j,k),(x,y,z)\right) = mW\left(\frac{|i-x|}{\Delta L}\right)W\left(\frac{|j-y|}{\Delta L}\right)W\left(\frac{|k-z|}{\Delta L}\right) \tag{3.3}$$

where M is the mass given to the grid point with position (i,j,k), (x,y,z) is the position of the particle, m is the mass of the particle and $W\left(\frac{|i-x|}{\Delta L}\right)$ is the Gaussian interpolation function given by:

$$W(x) = \frac{1}{\sigma\sqrt{2\pi}}\exp\left(\frac{-x^2}{2\sigma^2}\right) \tag{3.4}$$

where σ is the standard deviation with a value of $\sigma = 1$ pixel in this research.

3.1.2 Fourier transform

An often-used technique to solve differential equations, like the Poisson equation, is to use the Fourier transform. The Fourier transform takes a function as input and has a function as output. The Fourier transform of a function $f\colon \mathbb{R}^3 \longrightarrow \mathbb{R}$ is given by:

$$\mathcal{F}\left[f\right](\boldsymbol{k}) = \hat{f}(\boldsymbol{k}) = \frac{1}{(2\pi)^{3/2}} \iiint_{\mathbb{R}^3} f(\boldsymbol{r}) \mathrm{e}^{-\mathrm{i}\boldsymbol{k}\cdot\boldsymbol{r}} \mathrm{d}^3 r \tag{3.5}$$

The inverse Fourier transform of $\hat{f}\colon \mathbb{R}^3 \longrightarrow \mathbb{R}$ is given by:

$$\mathcal{F}^{-1}\left[\hat{f}\right](\boldsymbol{r}) = f(\boldsymbol{r}) = \frac{1}{(2\pi)^{3/2}} \iiint_{\mathbb{R}^3} \hat{f}(\boldsymbol{k}) \mathrm{e}^{\mathrm{i}\boldsymbol{k}\cdot\boldsymbol{r}} \mathrm{d}^3 k \tag{3.6}$$

And the Fourier transform of a function $\boldsymbol{f}\colon \mathbb{R}^3 \longrightarrow \mathbb{R}^3$ is given by:

$$\mathcal{F}\left[\boldsymbol{f}\right](\boldsymbol{k}) = \hat{\boldsymbol{f}}(\boldsymbol{k}) = \frac{1}{(2\pi)^{3/2}} \iiint_{\mathbb{R}^3} \boldsymbol{f}(\boldsymbol{r}) \mathrm{e}^{-\mathrm{i}\boldsymbol{k}\cdot\boldsymbol{r}} \mathrm{d}^3 r \tag{3.7}$$

The inverse Fourier transform of $\hat{\boldsymbol{f}}\colon \mathbb{R}^3 \longrightarrow \mathbb{R}^3$ is given by:

$$\mathcal{F}^{-1}\left[\hat{\boldsymbol{f}}\right](\boldsymbol{r}) = \boldsymbol{f}(\boldsymbol{r}) = \frac{1}{(2\pi)^{3/2}} \iiint_{\mathbb{R}^3} \hat{\boldsymbol{f}}(\boldsymbol{k}) \mathrm{e}^{\mathrm{i}\boldsymbol{k}\cdot\boldsymbol{r}} \mathrm{d}^3 k \tag{3.8}$$

When using the Fourier transform operation on the left-hand side of the Poisson equation, the following results:

$$\begin{aligned} \mathcal{F}\left[\boldsymbol{\nabla}^2\Phi(\boldsymbol{r})\right](\boldsymbol{k}) &= \frac{1}{(2\pi)^{3/2}} \iiint_{\mathbb{R}^3} \boldsymbol{\nabla}^2 \left(\Phi(\boldsymbol{r})\right) \mathrm{e}^{-\mathrm{i}\boldsymbol{k}\cdot\boldsymbol{r}} \mathrm{d}^3 r \\ &= \frac{1}{(2\pi)^{3/2}} \left[\iiint_{\mathbb{R}^3} \boldsymbol{\nabla} \cdot \left(\boldsymbol{\nabla}\left(\Phi(\boldsymbol{r})\right) \mathrm{e}^{-\mathrm{i}\boldsymbol{k}\cdot\boldsymbol{r}}\right) \mathrm{d}^3 r - \iiint_{\mathbb{R}^3} \boldsymbol{\nabla}\left(\Phi(\boldsymbol{r})\right) \cdot \boldsymbol{\nabla} \mathrm{e}^{-\mathrm{i}\boldsymbol{k}\cdot\boldsymbol{r}} \mathrm{d}^3 r \right] \end{aligned} \tag{3.9}$$

where the first integral is zero, because $\boldsymbol{\nabla}\Phi \longrightarrow \boldsymbol{0}$, as $r \longrightarrow \infty$, and because of the divergence theorem. Continuing:

$$\begin{aligned} \mathcal{F}\left[\boldsymbol{\nabla}^2\Phi(\boldsymbol{r})\right](\boldsymbol{k}) &= \frac{1}{(2\pi)^{3/2}} \left[\iiint_{\mathbb{R}^3} \boldsymbol{\nabla}\left(\Phi(\boldsymbol{r})\right) \cdot \mathrm{i}\boldsymbol{k} \mathrm{e}^{-\mathrm{i}\boldsymbol{k}\cdot\boldsymbol{r}} \mathrm{d}^3 r \right] \\ &= \frac{\mathrm{i}\boldsymbol{k}}{(2\pi)^{3/2}} \cdot \left[\iiint_{\mathbb{R}^3} \boldsymbol{\nabla}\left(\Phi(\boldsymbol{r}) \mathrm{e}^{-\mathrm{i}\boldsymbol{k}\cdot\boldsymbol{r}}\right) \mathrm{d}^3 r - \iiint_{\mathbb{R}^3} \Phi(\boldsymbol{r}) \boldsymbol{\nabla} \mathrm{e}^{-\mathrm{i}\boldsymbol{k}\cdot\boldsymbol{r}} \mathrm{d}^3 r \right] \end{aligned} \tag{3.10}$$

where again the first integral is zero, because $\Phi \longrightarrow 0$, as $r \longrightarrow \infty$, and because of the gradient version of the divergence theorem (shown in appendix). Continuing:

$$\mathcal{F}\left[\boldsymbol{\nabla}^2\Phi(\boldsymbol{r})\right](\boldsymbol{k}) = \frac{-|\boldsymbol{k}|^2}{(2\pi)^{3/2}} \left[\iiint_{\mathbb{R}^3} \Phi(\boldsymbol{r}) \mathrm{e}^{-\mathrm{i}\boldsymbol{k}\cdot\boldsymbol{r}} \mathrm{d}^3 r \right] = -|\boldsymbol{k}|^2 \mathcal{F}\left[\Phi(\boldsymbol{r})\right](\boldsymbol{k}) \tag{3.11}$$

Thus the Poisson equation can be solved by taking the Fourier transform, which gives:

$$\hat{\Phi} = -4\pi G \frac{\hat{\rho}}{|\boldsymbol{k}|^2} \tag{3.12}$$

for $\boldsymbol{k} \neq \boldsymbol{0}$. By taking the inverse Fourier transform of equation (3.12), the solution is found:

$$\Phi = -4\pi G \mathcal{F}^{-1}\left[\frac{\hat{\rho}}{|\boldsymbol{k}|^2}\right](\boldsymbol{r}) \tag{3.13}$$

By calculating the potential from the Poisson equation, the acceleration field can then be determined by:

$$\boldsymbol{a} = -\boldsymbol{\nabla}\Phi \tag{3.14}$$

3.1.3 Discrete Fourier transform

The previous subsection discussed the Fourier transform for continuous functions, but the code is discrete. Therefore, the discrete Fourier transform (DFT) is used instead, which is an approximation of the Fourier transform. The DFT converts a finite sequence of equally spaced samples of a function into an equally long and equally spaced sequence of numbers. If the input sequence is given by $x_0, x_1, ..., x_{N-1}$, then the DFT is given by:

$$X_k = \sum_{n=0}^{N-1} x_n \cdot \mathrm{e}^{-\mathrm{i}2\pi \frac{k}{N} n} \tag{3.15}$$

and its inverse (IDFT) is given by:

$$x_n = \frac{1}{N} \sum_{k=0}^{N-1} X_k \cdot \mathrm{e}^{\mathrm{i}2\pi \frac{k}{N} n} \tag{3.16}$$

As the particle mesh method uses three-dimensional matrices, the three-dimensional discrete Fourier transform is needed. If the input is given by numbers x_{n_1,n_2,n_3} with $0 \leq n_i \leq N_i - 1$ for $i = 1, 2, 3$, then the three-dimensional discrete Fourier transform is given by:

$$X_{k_1,k_2,k_3} = \sum_{n_1=0}^{N_1-1} \sum_{n_2=0}^{N_2-1} \sum_{n_3=0}^{N_3-1} x_{n_1,n_2,n_3} \cdot \mathrm{e}^{-\mathrm{i}2\pi \left(\frac{n_1 k_1}{N_1} + \frac{n_2 k_2}{N_2} + \frac{n_3 k_3}{N_3} \right)} \tag{3.17}$$

where $0 \leq k_i \leq N_i - 1$ for $i = 1, 2, 3$.

The inverse three-dimensional discrete Fourier transform is given by:

$$x_{n_1,n_2,n_3} = \frac{1}{N_1 N_2 N_3} \sum_{k_1=0}^{N_1-1} \sum_{k_2=0}^{N_2-1} \sum_{k_3=0}^{N_3-1} X_{k_1,k_2,k_3} \cdot \mathrm{e}^{\mathrm{i}2\pi \left(\frac{n_1 k_1}{N_1} + \frac{n_2 k_2}{N_2} + \frac{n_3 k_3}{N_3} \right)} \tag{3.18}$$

and in this research $N_1 = N_2 = N_3 = N$.

Fast Fourier transform

The fast Fourier Transform (FFT) is an algorithm that computes the DFT and IDFT. The complexity of computing the DFT or the IDFT is $\boldsymbol{O}(n^2)$ with n the input size. The FFT reduces this complexity to $\boldsymbol{O}(n \log(n))$. This complexity depends on the factorization of n, with smaller prime factors being the faster method.

3.1.4 Assigning accelerations

Now that we have described a method to numerically solve the potential, the acceleration field needs to be calculated. The acceleration field is given by $\boldsymbol{a} = -\boldsymbol{\nabla}\Phi$, but Φ is discretized. To calculate $\boldsymbol{\nabla}\Phi$, the following finite difference method was used, where (i, j, k) is some grid coordinate:

$$\boldsymbol{\nabla}\Phi(i,j,k) = \left(\frac{\Phi(i+1,j,k) - \Phi(i-1,j,k)}{2\Delta L}, \frac{\Phi(i,j+1,k) - \Phi(i,j-1,k)}{2\Delta L}, \frac{\Phi(i,j,k+1) - \Phi(i,j,k-1)}{2\Delta L} \right)^{\mathsf{T}} \tag{3.19}$$

Now that $\boldsymbol{a}(i,j,k) = -\boldsymbol{\nabla}\Phi(i,j,k)$ is known at each grid point, the acceleration needs to be assigned back to the particles. Initially, the mass of the particles was distributed among neighboring grid points with the Gaussian interpolation function from equation (3.4). A similar approach is used to determine the acceleration of the particles. Only the grid points to which mass has been distributed are used, which is the set of grid points C. Now, the acceleration is given by summing over the grid points in C, taking into account the corresponding weight:

$$\boldsymbol{a}(x,y,z) = \sum_{(i,j,k) \in C} \boldsymbol{a}(i,j,k) W\left(\frac{|i-x|}{\Delta L} \right) W\left(\frac{|j-y|}{\Delta L} \right) W\left(\frac{|k-z|}{\Delta L} \right) \tag{3.20}$$

3.1.5 Leapfrog integration

In the particle mesh method, the positions and velocities of each particle need to be calculated after each time step. For this calculation, leapfrog integration was used. Let i be the current time step, then the leapfrog method to calculate the position and velocity for time step $i+1$ looks as follows:

$$\begin{aligned} \boldsymbol{x}_{i+1} &= \boldsymbol{x}_i + \boldsymbol{v}_i \Delta t + \frac{1}{2}\boldsymbol{a}_i \Delta t^2 \\ \boldsymbol{v}_{i+1} &= \boldsymbol{v}_i + \frac{1}{2}(\boldsymbol{a}_i + \boldsymbol{a}_{i+1})\Delta t \end{aligned} \tag{3.21}$$

3.1.6 Complete Newtonian method

In the previous subsections, all necessary steps for the Newtonian particle mesh method have been discussed. Thus the complete method is given below:

1. Distribute the mass of the particles among the neighboring grid points as described in subsection (3.1.1). Summing the contribution of each particle gives the mass density ρ on the grid.
2. Use the FFT to Fourier transform ρ.
3. Calculate the Fourier transformed potential, as described in equation (3.13).
4. Calculate the potential using the inverse FFT.
5. Calculate the acceleration field by using the finite difference method as described in subsection (3.1.4).
6. Use the acceleration field to calculate the acceleration of the particles.
7. Calculate the new positions and velocities of the particles using the Leapfrog method as described in subsection (3.1.5).

3.2. MOND particle mesh

The particle mesh method for MOND is similar to that for Newtonian dynamics. The main difference is the calculation of the acceleration field because the Poisson equation for MOND is non-linear and thus it cannot be solved directly using Fourier transforms. To understand the MOND particle mesh method, first, the Helmholtz decomposition needs to be discussed.

3.2.1 Helmholtz decomposition

The Helmholtz decomposition decomposes a vector field $\boldsymbol{F}(\boldsymbol{r})$ into a curl-free vector field $\boldsymbol{G}(\boldsymbol{r})$ and a divergence-free vector field $\boldsymbol{H}(\boldsymbol{r})$ such that:

$$\begin{aligned} &\boldsymbol{F}(\boldsymbol{r}) = \boldsymbol{G}(\boldsymbol{r}) + \boldsymbol{H}(\boldsymbol{r}) \\ &\boldsymbol{G}(\boldsymbol{r}) = -\boldsymbol{\nabla}\Phi \\ &\boldsymbol{\nabla} \times \boldsymbol{G}(\boldsymbol{r}) = \boldsymbol{0} \\ &\boldsymbol{H}(\boldsymbol{r}) = \boldsymbol{\nabla} \times \boldsymbol{A} \\ &\boldsymbol{\nabla} \cdot \boldsymbol{H}(\boldsymbol{r}) = \boldsymbol{0} \end{aligned} \tag{3.22}$$

where Φ is a scalar potential and $\boldsymbol{A}$ is a vector potential. In the code, these curl-free and divergence-free vector fields need to be computed. This can be done using the Fourier transform in the following way:

$$\boldsymbol{\nabla} \cdot \boldsymbol{F}(\boldsymbol{r}) = \boldsymbol{\nabla} \cdot (\boldsymbol{G}(\boldsymbol{r}) + \boldsymbol{H}(\boldsymbol{r})) = \boldsymbol{\nabla} \cdot \boldsymbol{G}(\boldsymbol{r}) \tag{3.23}$$

$$\begin{aligned} \mathcal{F}\left[\boldsymbol{\nabla} \cdot \boldsymbol{G}(\boldsymbol{r})\right](\boldsymbol{k}) &= \frac{1}{(2\pi)^{3/2}} \iiint_V \boldsymbol{\nabla} \cdot \boldsymbol{G}(\boldsymbol{r}) \mathrm{e}^{-\mathrm{i}\boldsymbol{k}\cdot\boldsymbol{r}} \mathrm{d}^3 r \\ &= \frac{1}{(2\pi)^{3/2}} \left[\iint_S \mathrm{e}^{-\mathrm{i}\boldsymbol{k}\cdot\boldsymbol{r}} \boldsymbol{G}(\boldsymbol{r}) \cdot \mathrm{d}\boldsymbol{A} - \iiint_V \boldsymbol{\nabla} \mathrm{e}^{-\mathrm{i}\boldsymbol{k}\cdot\boldsymbol{r}} \cdot \boldsymbol{G}(\boldsymbol{r}) \mathrm{d}^3 r \right] \end{aligned} \tag{3.24}$$

and when assuming that $\boldsymbol{G}(\boldsymbol{r}) \longrightarrow 0$ as $r \longrightarrow \infty$, the first surface integral vanishes:

$$\mathcal{F}\left[\boldsymbol{\nabla} \cdot \boldsymbol{G}(\boldsymbol{r})\right](\boldsymbol{k}) = \frac{\mathrm{i}\boldsymbol{k}}{(2\pi)^{3/2}} \cdot \iiint_V \mathrm{e}^{-\mathrm{i}\boldsymbol{k}\cdot\boldsymbol{r}} \boldsymbol{G}(\boldsymbol{r}) \mathrm{d}^3 r = \mathrm{i}\boldsymbol{k} \cdot \boldsymbol{\mathcal{F}}\left[\boldsymbol{G}(\boldsymbol{r})\right](\boldsymbol{k}) \tag{3.25}$$

Similarly:

$$\mathcal{F}\left[\boldsymbol{\nabla} \cdot \boldsymbol{F}(\boldsymbol{r})\right](\boldsymbol{k}) = \mathrm{i}\boldsymbol{k} \cdot \boldsymbol{\mathcal{F}}\left[\boldsymbol{F}(\boldsymbol{r})\right](\boldsymbol{k}) \tag{3.26}$$

Plugging equations (3.25) and (3.26) into equation (3.23) gives:

$$\boldsymbol{k} \cdot \boldsymbol{\mathcal{F}}\left[\boldsymbol{F}(\boldsymbol{r})\right](\boldsymbol{k}) = \boldsymbol{k} \cdot \boldsymbol{\mathcal{F}}\left[\boldsymbol{G}(\boldsymbol{r})\right](\boldsymbol{k}) \tag{3.27}$$

Now note that:

$$\boldsymbol{\mathcal{F}}\left[\boldsymbol{G}(\boldsymbol{r})\right](\boldsymbol{k}) = \boldsymbol{\mathcal{F}}\left[-\boldsymbol{\nabla}\Phi(\boldsymbol{r})\right](\boldsymbol{k}) = -\mathrm{i}\boldsymbol{k}\mathcal{F}[\Phi](\boldsymbol{k}) \tag{3.28}$$

So this shows that $\boldsymbol{\mathcal{F}}\left[\boldsymbol{G}(\boldsymbol{r})\right](\boldsymbol{k})$ is parallel to $\boldsymbol{k}$ giving:

$$\boldsymbol{\mathcal{F}}\left[\boldsymbol{G}(\boldsymbol{r})\right](\boldsymbol{k}) = \mathcal{F}\left[\boldsymbol{G}(\boldsymbol{r})\right](\boldsymbol{k}) \frac{\boldsymbol{k}}{k} \tag{3.29}$$

Plugging equation (3.29) into equation (3.27), and when multiplying by $\frac{\boldsymbol{k}}{k^2}$, gives:

$$\boldsymbol{\mathcal{F}}\left[\boldsymbol{G}(\boldsymbol{r})\right](\boldsymbol{k}) = \frac{\boldsymbol{k} \cdot \boldsymbol{\mathcal{F}}\left[\boldsymbol{F}(\boldsymbol{r})\right](\boldsymbol{k})}{k^2} \boldsymbol{k} \tag{3.30}$$

Now taking the inverse Fourier transform gives $\boldsymbol{G}(\boldsymbol{r})$, and thus also $\boldsymbol{H}(\boldsymbol{r}) = \boldsymbol{F}(\boldsymbol{r}) - \boldsymbol{G}(\boldsymbol{r})$.

3.2.2 Complete MOND method

The MOND particle mesh method follows the same steps as the Newtonian particle mesh method, except for the calculation for the acceleration field. To solve the non-linear Poisson equation for MOND, P.M. Visser transformed the non-linear partial differential equation into a system of coupled linear partial differential equations and one algebraic equation. If $\boldsymbol{g}_M = -\boldsymbol{\nabla}\Phi$ is the MOND acceleration field, and $\boldsymbol{g}_N = \boldsymbol{\nabla}\Phi_N$ the Newtonian acceleration field, then the following holds:

$$\begin{cases} \boldsymbol{\nabla} \cdot \boldsymbol{g}_N = -4\pi G\rho & (3.31) \\ \boldsymbol{\nabla} \times \boldsymbol{g}_N = \boldsymbol{0} & (3.32) \\ \boldsymbol{F} = \boldsymbol{g}_N + \boldsymbol{H} & (3.33) \\ \boldsymbol{\nabla} \cdot \boldsymbol{H} = 0 & (3.34) \\ \boldsymbol{g}_M = \nu\left(\dfrac{F}{a_0}\right)\boldsymbol{F} & (3.35) \\ \boldsymbol{\nabla} \times \boldsymbol{g}_M = \boldsymbol{0} & (3.36) \\ \boldsymbol{F} = \mu\left(\dfrac{g_M}{a_0}\right)\boldsymbol{g}_M & (3.37) \end{cases}$$

where equations (3.35) and (3.37) are the inverse of each other, where:

$$\mu(x)x = y = \frac{F}{a_0} \text{ , and } \quad \nu(y)y = x = \frac{g_M}{a_0}. \tag{3.38}$$

If for μ, the standard interpolation function

$$\mu(x) = \frac{x}{\sqrt{1+x^2}} \tag{3.39}$$

is chosen, then $\nu(y)$ is given as:

$$\nu(y) = \sqrt{\frac{1}{2} + \frac{1}{2}\sqrt{1+\frac{4}{y^2}}}. \tag{3.40}$$

The non-linear Poisson equation is equivalent to equations (3.31) to (3.37):

$$-4\pi G\rho \overset{(3.31)}{=} \boldsymbol{\nabla}\cdot\boldsymbol{g}_N \overset{(3.33)}{=} \boldsymbol{\nabla}\cdot\boldsymbol{F} - \boldsymbol{\nabla}\cdot\boldsymbol{H} \overset{(3.34)}{=} \boldsymbol{\nabla}\cdot\boldsymbol{F} \overset{(3.37)}{=} \boldsymbol{\nabla}\cdot\left(\mu\left(\frac{g_M}{a_0}\right)\boldsymbol{g}_M\right), \tag{3.41}$$

hence the MOND acceleration field can be found in equation (3.35).

The complete MOND method is listed below, where the first five steps are the same as in the Newton case:

1. Distribute the mass of the particles among the neighboring grid points as described in subsection (3.1.1). Summing the contribution of each particle gives the mass density ρ on the grid.
2. Use the FFT to Fourier transform ρ.
3. Calculate the Fourier transformed potential, as described in equation (3.13).
4. Calculate the potential using the inverse FFT.
5. Calculate the acceleration field by using the finite difference method as described in subsection (3.1.4).
6. Create a vector field $\boldsymbol{H}_d$ with only zero's.
7. Iterate the following steps:
   (a) Let $\boldsymbol{F} \overset{(3.33)}{=} \boldsymbol{g}_N + \boldsymbol{H}_d$
   (b) Set $\boldsymbol{g}_M \overset{(3.35)}{=} \nu\left(\frac{F}{a_0}\right)\boldsymbol{F}$
   (c) Calculate the curl-free projection $\boldsymbol{g}_c$ of $\boldsymbol{g}_M$
   (d) Let $\boldsymbol{F} \overset{(3.37)}{=} \mu\left(\frac{g_c}{a_0}\right)\boldsymbol{g}_c$
   (e) Set $\boldsymbol{H} \overset{(3.33)}{=} \boldsymbol{F} - \boldsymbol{g}_N$
   (f) Project $\boldsymbol{H}$ onto its divergence-free part $\boldsymbol{H}_d$
   (g) Go back to step (a), now using the updated $\boldsymbol{H}_d$ field.
8. Take the Fourier transform of the acceleration field $\boldsymbol{g}_c$.
9. Calculate the transformed MOND potential $\hat{\Phi} = -\mathrm{i}\boldsymbol{k}\cdot\boldsymbol{g}_c$.
10. Calculate the MOND potential Φ by taking the inverse Fourier transform of $\hat{\Phi}$.
11. Calculate the MOND acceleration field by using the finite difference method as discussed in subsection (3.1.4).
12. Use the acceleration field to calculate the acceleration of the particles.
13. Calculate the new positions and velocities of the particles using the Leapfrog method as described in subsection (3.1.5).

Paul Visser et al. showed that for most simulations, four iterations in step (7g) were sufficient for accurate results[17].

3.3. Parallel programming

The code developed by Joost de Nijs used the CPU (Central Processing Unit) to perform the calculations. The CPU is fast in performing a series of tasks, known as serial processing. The GPU (Graphical Processing Unit) is slower than the CPU when performing a series of tasks, but it is much faster when performing tasks in parallel, known as parallel processing. To perform tasks in parallel is to perform several tasks of the same type at once. Examples of tasks that the GPU can perform much faster than the CPU are: FFT, elementwise operations on an array, the sum of an array, and matrix multiplication.

Figure (3.1) shows the amount of speedup that can be achieved when switching from the CPU to the GPU [18].

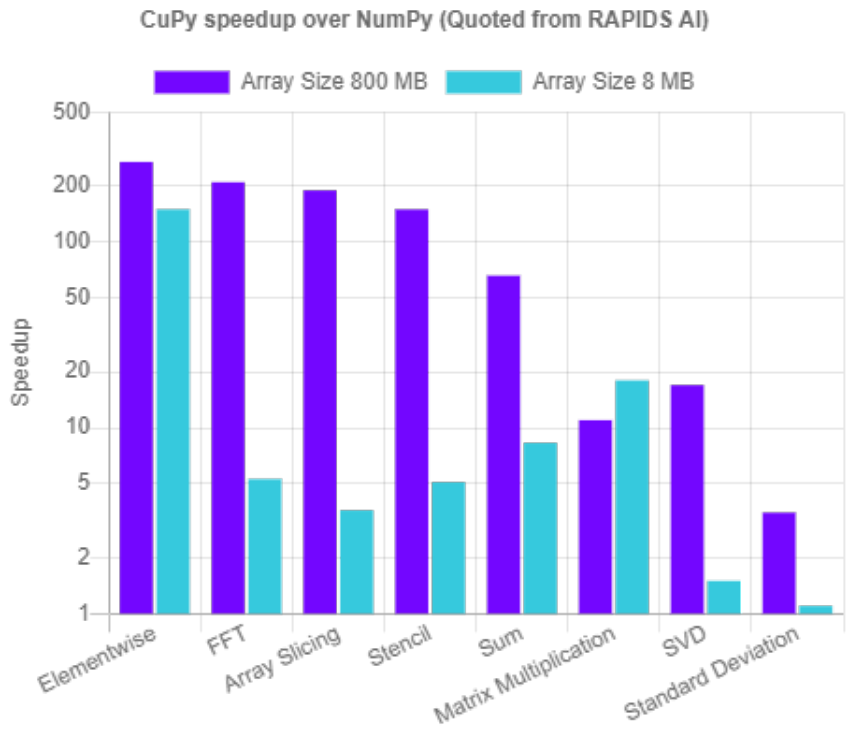


Figure 3.1: On the x-axis, different types of tasks are shown. On the y-axis, the speedup of CuPy compared to NumPy is shown. All tasks shown have a significant speedup for large array sizes. The speedup of CuPy compared to NumPy is larger for larger arrays, except for matrix multiplication[18].

When using CuPy instead of NumPy in the simulations, the GPU code is faster for large array sizes. The simulation computation time, however, is expected to be lower for smaller array sizes as well, based on figure (3.1). Thus there must be another factor that could explain this computation time. This factor was found to be the task of distributing the mass across the grid points. This mass distribution task was performed on the GPU and was coded as a serial processing task. After manually parallelizing this task (not using the CuPy library), the computation time for the GPU code was much faster than the previous GPU time and was also much faster than the CPU time.

3.3.1 Simulation performance

In this subsection, the relation between computation time and the number of pixels will be shown. After this, the relation between computation time and the number of simulated particles is shown. For each of these relations, three different codes will be compared. These codes will be a CPU-based code, a GPU-based code, and a GPU-based code with mass and acceleration distribution parallelization. In each figure, all simulations had 100 time steps.

In figure (3.2), the relation between the computation time and the number of pixels is shown. In all simulations, two particles were simulated.

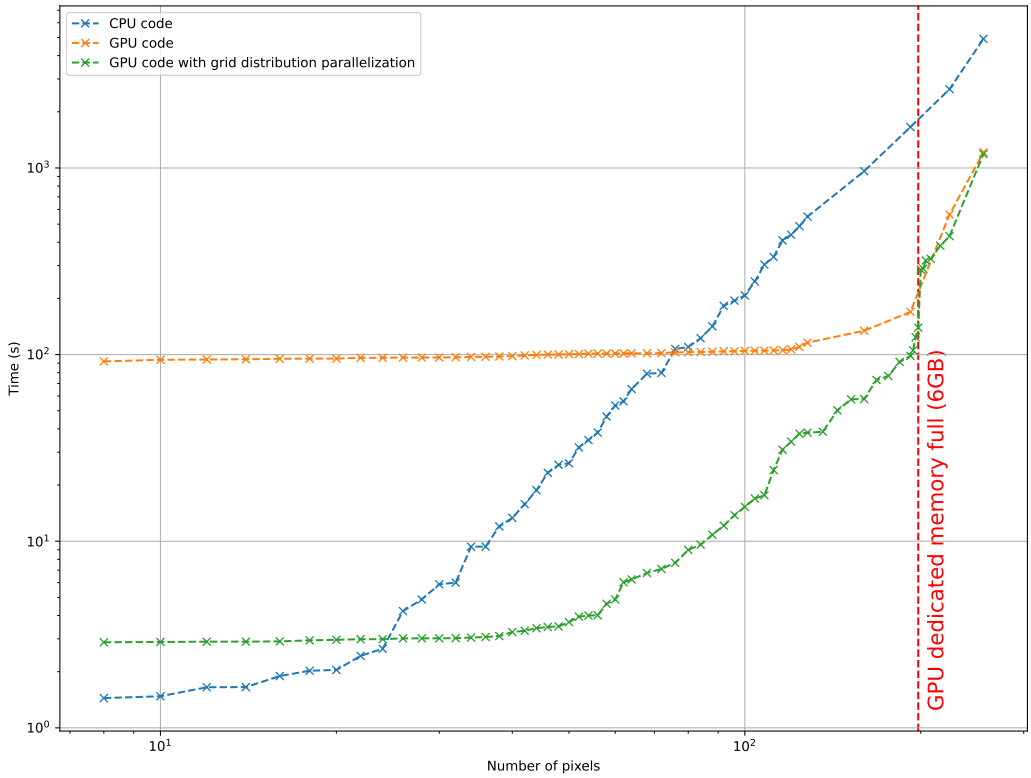


Figure 3.2: The computation time of a simulation with two stars plotted against the number of pixels p, where the grid is made up of $p \times p \times p$ pixels. The blue, orange, and green lines show the CPU code, GPU code, and GPU code with mass and acceleration distribution parallelization. The CPU code is initially faster than both GPU codes, because of the GPU overhead time. The orange line has a large overhead time, which originates from the mass and acceleration distribution. As the number of pixels increases, the CPU computation time increases earlier than the GPU codes, because the parallelization advantage of the GPU becomes significant at larger array sizes. At around 200 pixels indicated with the red dashed line, the GPU dedicated memory is full, resulting in the use of the CPU memory which introduces a significant increase in computation time (which is not present in the CPU code). None of the lines are smooth, which results from the FFT, where the FFT is faster if the number of pixels p has small prime factors.

In figure 3.2, the CPU code initially shows the fastest performance due to its lack of overhead compared to the GPU codes. The GPU code without parallelization has a significant overhead time from the mass and acceleration distribution tasks. This overhead is reduced when these tasks are parallelized, as indicated by the green line. As the number of pixels increases, the GPU codes eventually outperform the CPU code in performance. This is explained by figure 3.1, where larger array sizes show a greater performance increase. Additionally, for the orange line, the computation time for FFT eventually becomes dominant over the distribution time.

Around 200 pixels, the dedicated GPU memory becomes full, resulting in the use of slower CPU memory and thereby significantly increasing computation time, which does not happen for the CPU code. The non-smooth lines originate from the fact that FFT performs optimally when the number of pixels has small prime factors, ideally powers of two.

Now, the relation between the computation time and the number of simulated particles is shown in figure (3.3). In these simulations, the number of pixels is 128.

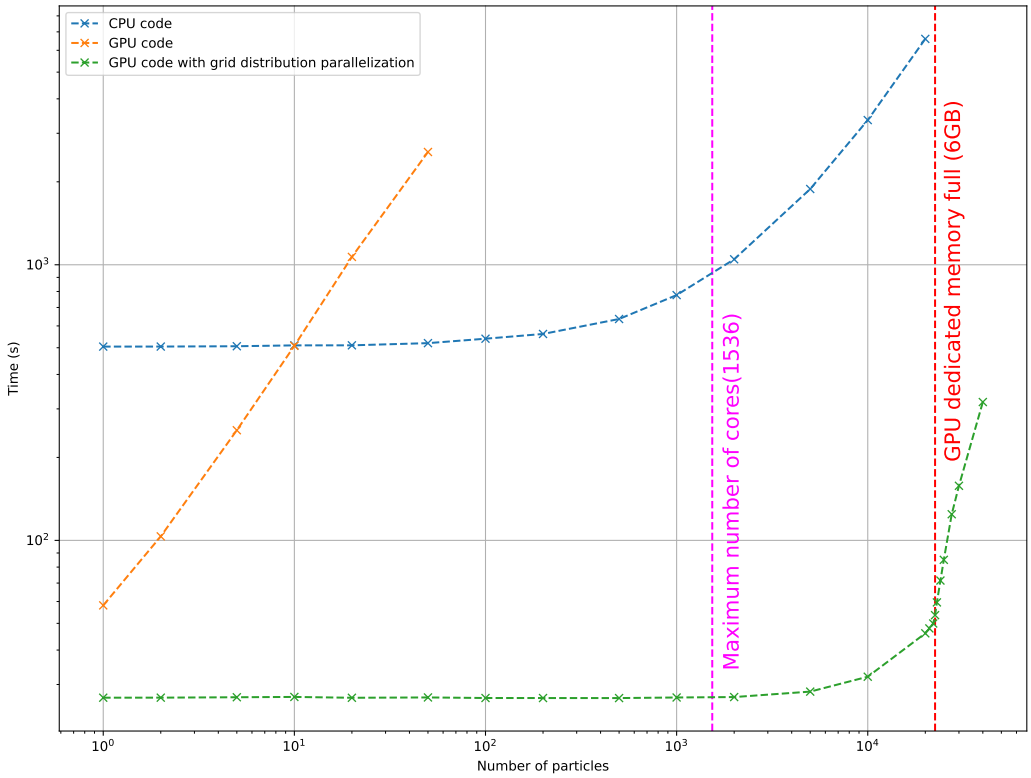


Figure 3.3: The computation time of a simulation with a $128 \times 128 \times 128$ grid plotted against the number of particles simulated. The blue, orange, and green lines show the CPU code, GPU code, and GPU code with mass and acceleration distribution parallelization. The blue line starts with a large overhead, because of the computation time of the FFT with a $128 \times 128 \times 128$ grid. At larger particle numbers, the mass and acceleration distribution computation time becomes significant. The computation time of the orange line exists mainly from distributing the mass and acceleration across the grid. Therefore, the orange line is linear, as the mass and acceleration distribution is linear with the number of particles. The green line has parallelized both the FFT and the mass and acceleration distributions. Therefore the computation time is constant, until around 1536, indicated with the pink dashed line. This is because the number of cores is 1536, after which the computation time slightly starts to increase. Around the 22500 particles indicated with the red dashed line, the GPU dedicated memory is full. Now the CPU memory has to be used which introduces a significant increase in computation time.

In figure (3.3), the orange line shows a linear relationship. This is because the distribution task is dominant in the computation time, and this task is linear with the number of particles. The overhead time for the blue line results from the FFT time, but the distribution task eventually becomes dominant. The green line has both the advantage of a fast FFT and of a fast distribution task. The green line starts to increase after about 1536, which is the number of cores used. Around 22500 particles, the dedicated GPU memory becomes full, resulting in the use of slower CPU memory and thereby significantly increasing computation time.

GPU memory optimization

The vertical line in figures (3.2) and (3.3) indicates the point at which the GPU dedicated memory is full. After this point, the performance of the code decreases significantly. This raises the question of whether the GPU memory can be optimized to use as little memory as possible. This way the performance of higher resolution simulations gets increased significantly. There are a few steps that could be taken to keep the used GPU memory as low as possible, beginning with using a different data type.

Data type:

The standard data type used in most NumPy and CuPy functions is float64. It costs 8 bytes (64 bits) to store data with data type float64. If the data type is switched from float64 to float32, the used GPU memory could potentially be halved because float32 uses only 4 bytes instead of the 8 bytes from float64. One might ask if instead of float32, float16 could be used. Although this seems a logical next step, it comes with difficulties. One of the difficulties is that the biggest number that can be represented with float16 is 65504. To understand where the number 65504 comes from, one needs to first understand how a number is stored in memory. To start, for a float16 number there are 16 bits available for storage. In figure 3.4 the way a float16 number is stored is shown[19].

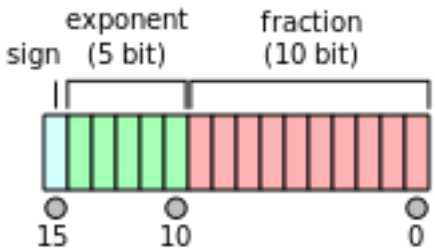


Figure 3.4: Diagram of how a number with data type float16 is stored in memory. There are 16 bits available for storage. The first bit is stored for the sign of the number, where 0 represents a positive number and 1 represents a negative number. The next 5 bits are used for the exponent. The exponent is stored with a -15 bias, meaning that from the exponent 15 is subtracted. The exponent gives a value of $2^{\text{exponent}-15}$. The last 10 bits are called the fraction/mantissa and are used for the significant digits of the number. The mantissa indicates what fraction from the $2^{\text{exponent}-15}$ should be taken. This fraction has an implicit 1 added to it. If the mantissa is 500, then the resulting number will be $\pm 2^{\text{exponent}-15} \cdot (1 + 500/2^{10})$ [19].

A float16 number consists of three parts: the sign bit, the exponent, and the mantissa. The sign bit is the first bit from 16, which indicates the sign of the number where 0 is positive and 1 is negative. The next 5 bits are for the exponent. The exponent value is biased, the decimal value of the 5 bits will be decreased by 15 to give the actual exponent value. So far, the float16 number looks as follows: $\pm 2^{\text{exponent}}$, where exponent $=$ decimal value of the 5 bits $-$ 15. Next up is the mantissa, which represents the significant digits of the number. The mantissa is a fraction that is taken of the $\pm 2^{\text{exponent}}$, with an implicit 1 added to the fraction. Thus if the 10 mantissa bits have a decimal value of 500, then the resulting number will be: $\pm 2^{\text{exponent}} \cdot (1 + 500/2^{10})$.

Now to come back to the highest possible value that a float16 can represent, a calculation is made. Take the first bit to be positive, the 5 exponent bits to be maximum, which is in decimal value 31, and the 10 mantissa bits maximum, which is in decimal value 1023. Then the maximum number representable by float16 is $+2^{31-15} \cdot (1 + 1023/1024) = 131008$, which is bigger than 65504. This is because when the 5 exponent bits are all 1, thus having the highest exponent, and when the mantissa are all 0, then the number is represented as infinity. If the mantissa is not all 0's, then NaN (Not a Number) returns. Thus not all exponent bits can be 1, so the next highest is 11110, which is 30 in decimal. This results in the number $+2^{30-15} \cdot (1 + 1023/1024) = 65504$.

Similarly, the smallest, non-zero, positive number which can be represented by float16 is: $+2^{0-15} \cdot (1 + 0/1024) = 3.051757 \cdot 10^{-5}$, but again an exception is made when the exponent bits are all 0. In this case, if all mantissa bits are 0, the resulting number is 0. If some mantissa bits are 1, then the number will be given by: $\pm 2^{-14} \cdot (0 +$ mantissa decimal$/1024)$. Note that the implicit one before the fraction part is now zero. Now, if taking the mantissa bits to be minimum without all being 0, then we get that they are all 0 except for the last bit. Then the smallest possible positive non-zero is: $+2^{-14} \cdot (0 + 1/1024) = 5.96046448 \cdot 10^{-8}$.

In the simulations when the mass is distributed with the Gaussian weight function, the mass numbers can get below the smallest positive non-zero number $5.96046448 \cdot 10^{-8}$. Also, the potential energy will get below the smallest possible number -65504. Moreover, the parallelized GPU code does not have a float16 implementation. Therefore, using the float16 datatype in the simulation is not possible. It may be possible to scale numbers outside the float16 number range such that they fit in the float16 number range, but the lack of float16 implementation makes the use of float16 not possible.

Data management:

On top of using a different data type, the amount of data that needs to be stored in memory can be optimized. This means that large matrices that were only needed for a few operations should be deleted once they serve no purpose anymore. If this is not done then there are large portions of the memory that are taken up by useless data, which in turn leads to less space for useful data.

Memory types and their performance:

There are two distinct memory types that the GPU can use. The first is VRAM (Video Random-Access Memory), and the second is RAM (Random-Access Memory). VRAM is also known as dedicated GPU memory and, in relation to the GPU, RAM is known as shared GPU memory. Both VRAM and RAM are types of memory that store data that is actively used by the CPU and/or GPU. The VRAM is exclusively used by the GPU, while the RAM is primarily used by the CPU, but can also be used by the GPU. For a GPU, using VRAM is faster than using RAM. There are two main reasons why VRAM is called faster than RAM. These are the higher bandwidth of VRAM and the lower latency of VRAM. The higher bandwidth means that the data transfer from VRAM to the GPU and from the GPU to VRAM is much higher when compared to RAM to GPU and GPU to RAM data transfer. The second reason why VRAM is faster than RAM is because VRAM has a lower latency than RAM. The latency is the time it takes the GPU to get or write data to the VRAM. Latency is lower for VRAM because it is optimized for the GPU and because it is physically next to the GPU, whereas the RAM is not physically close and not optimized for the GPU. As mentioned earlier, the GPU can also use RAM. When the VRAM is full, the GPU will use the RAM as overflow memory. For the GPU to now use RAM, the RAM has to be transported via the PCIe bus, which although fast, is significantly slower than the VRAM to GPU data transfer. This performance decrease can be seen in figures (3.2) and (3.3), where the point at which the VRAM is full is indicated with a vertical line.

In figures (3.5) and (3.6) the effect of memory optimization on the computation time can be seen. In these figures, the memory-optimized code (in black) is compared to the fastest non-optimized code from figures (3.2) and (3.3) (which is still in green).

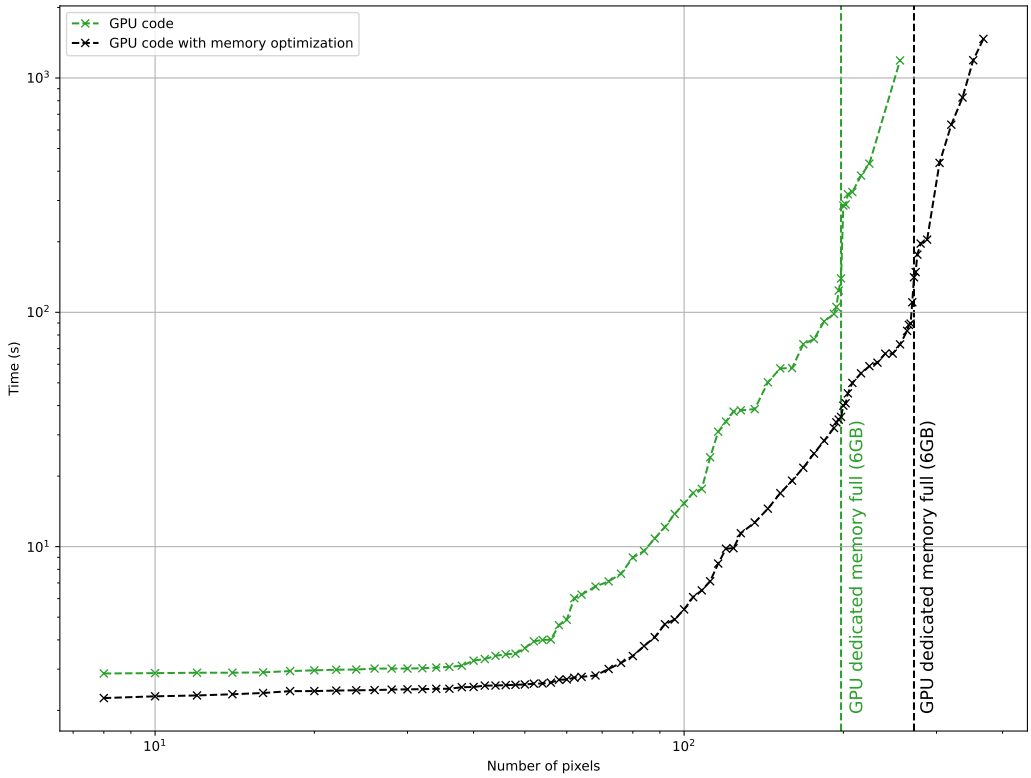


Figure 3.5: The computation time of a simulation with two stars plotted against the number of pixels p, where the grid is made up of $p \times p \times p$ pixels. The green line is the non-optimized code, while the black line is optimized. It can be seen that the optimized code has a lower overhead time because less data has to be transported. When the GPU memory becomes full, indicated by the vertical line in the respective colors, the CPU memory has to be used which introduces a significant increase in computation time. The increase in performance time and the dedicated GPU memory becoming full happens at a larger number of pixels for the optimized code. Both can be explained because less data has to be processed.

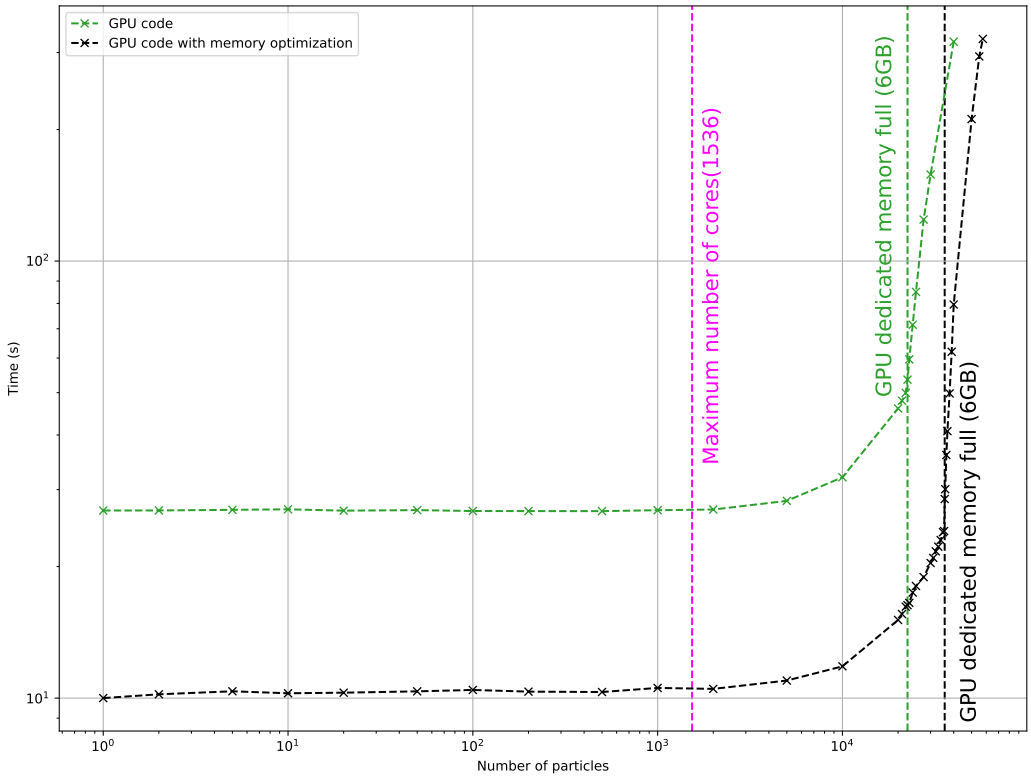


Figure 3.6: The computation time of a simulation with a $128 \times 128 \times 128$ grid plotted against the number of particles simulated. The green line is the non-optimized code, while the black line is optimized. It can be seen that the optimized code has a lower overhead time because less data has to be transported. When the GPU memory becomes full, indicated by the vertical line in the respective color, the CPU memory has to be used which introduces a significant increase in computation time. The dedicated GPU memory becoming full happens at a larger number of particles for the optimized code because less data has to be processed. The computation time is constant, until around 1536, indicated with the pink dashed line. This is because the number of cores is 1536, after which the computation time slightly starts to increase.

As can be seen from figures (3.5) and (3.6), the computation time is significantly lower for all cases. This may be explained by the fact that less data has to be transferred from the VRAM (and RAM) to the GPU, and thus the data transfer time (and thus total time) is decreased. The maximum size of the simulation is higher for the total pixels and for the number of particles. This may be explained by the fact that useless data is replaced by useful data, and thus there is more room for bigger simulation sizes.

4

Initial conditions

This chapter describes how the initial conditions for the simulations have been chosen. In this research, real wide binary systems are modeled and simulated. Therefore, the use of real wide binary data is useful for determining the initial conditions. Luckily, the European Space Agency (ESA) launched the Gaia mission in 2013 with the main goal of measuring the positions and velocities of approximately one billion stars from the Milky Way galaxy. Therefore, the Gaia database is well suited for retrieving the starting conditions of wide binary systems. The first section describes which data the Gaia database provides, the second section describes how this data can be used in our simulations and the third section examines the quality of the data from the Gaia database.

4.1. Gaia data

The Gaia satellite is a space telescope of the ESA and is designed for the precise measurement of the positions and movements of stars. The Gaia satellite is equipped with three main instruments: the astrometry instrument for measuring angular positions, the photometric instrument for measuring the luminosity of stars in the 320-1000 nm spectral band, and the radial velocity spectrometer used for determining the velocity along the line of sight.

The Gaia data is publicly available to use and can be accessed through the Gaia archive. From the Gaia archives, all observations from the Gaia mission can be retrieved. This research is only interested in wide binary data with the seven position, velocity, and mass parameters. Therefore the database which is used comes from the paper Pittordis and Sutherland [11] where a careful wide binary sample selection has been performed on the data from the Gaia databases. The Gaia data has been released in the following stages: DR1, DR2, EDR3, and DR3 (where DR stands for Data Release, and EDR for Early Data Release). The database from Pittordis and Sutherland only uses the data from DR2 and the EDR3.

Now that a suitable data set has been found, the data which is in it has to be understood which will be done in the following subsections. The only data that is of interest is the position, the velocity, and the mass of the stars in the wide binary. For a better understanding, the position and velocity data are given in coordinates that are closely related to spherical coordinates with the Earth in the center.

4.1.1 Parallax

To position a star in spherical coordinates, the distance from Earth to the star has to be known. This distance can be determined by the parallax, which is usually a very small angle. The parallax is determined by two observations. The first observed angle is made at an arbitrary time. The second observed angle is made exactly half a year later such that the Earth has moved 2 AU. By now also measuring the angle to the star, the distance to that star can be estimated. In figure (4.1), the observations and the desired angle can be seen geometrically for more clarity.

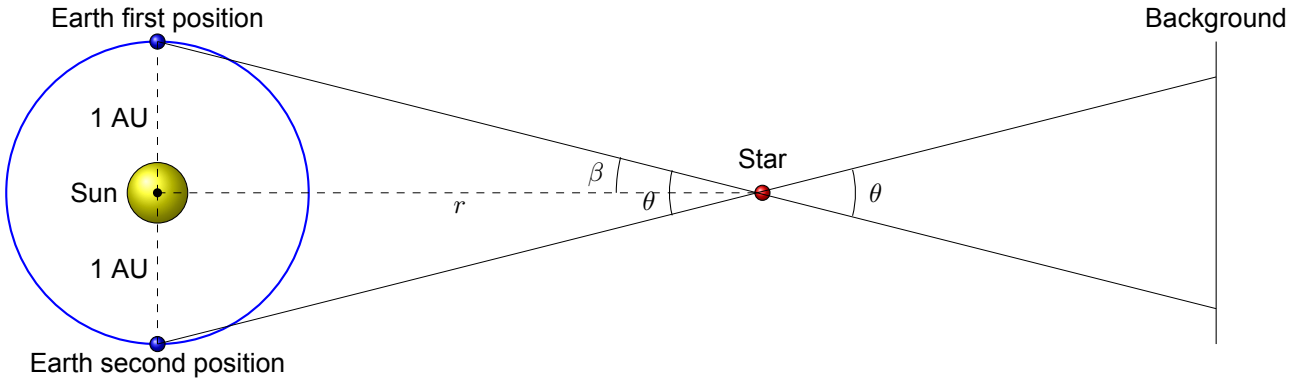


Figure 4.1: A schematic overview of the parallax. The position in the sky of a star is measured at the first and second Earth positions. Between these measurements, the star seems to be shifted when compared to the background stars. This angular shift is denoted as the apex angle θ. The parallax β is then defined to be half of the apex angle: $\beta = \theta/2$.

The angle θ in figure (4.1) can be measured by looking at the apparent shift of the star as the Earth has made half of its orbit. In this measurement, the assumption is that the background is stationary. With this assumption, the shift of the star against the background as the Earth moves can be measured. The angle θ is called the apex angle, but we are interested in the parallax which is half of the apex angle θ, denoted as the angle β. With elementary trigonometry, the distance r can be calculated:

$$r = \frac{1\text{ AU}}{\tan\beta} \approx \frac{1\text{ AU}}{\beta} \quad \text{, as } \tan(\beta) \approx \beta \text{ for small angles } \beta \tag{4.1}$$

Now that the distance to the star has been calculated, there are two more positional parameters left which will be handled next.

4.1.2 Right ascension and declination

The next positional parameters are the, in spherical coordinates, azimuthal angle φ and the polar angle θ. In astronomy, two different names for these angles are used which are called the right ascension (instead of φ) and the declination (instead of θ). Another difference is that the right ascension is not measured in units of radians or degrees, but rather in units of hour angles. In hour angles 360°/24 = 1h (h for hour) and 360°/(24·60) = 1m (m for minutes) and 360°/(24·60·60) = 1s (s for seconds). The declination is also slightly different than the polar angle because it ranges from -90° to +90° instead of ranging from 0° to 180°. Apart from these two differences, they have the same function as their spherical counterparts. Another definition that needs to be addressed is where the right ascension and declination are defined to be 0. To define this point, the equatorial plane (plane of Earth's equator) is intersected with the ecliptic plane (plane of Earth's orbit around the Sun). Then a line results, giving a possibility of two directions as seen from Earth. The direction in which spring starts in the northern hemisphere of the Earth is defined to be the zero point for both the right ascension and the declination and is called the March equinox. In most cases, a year is mentioned for the definition of the zero point direction because, due to precession, the Earth's equatorial plane may change slightly over the years. In figure (4.2) a geometric overview has been given of the different planes and the right ascension and declination angles.

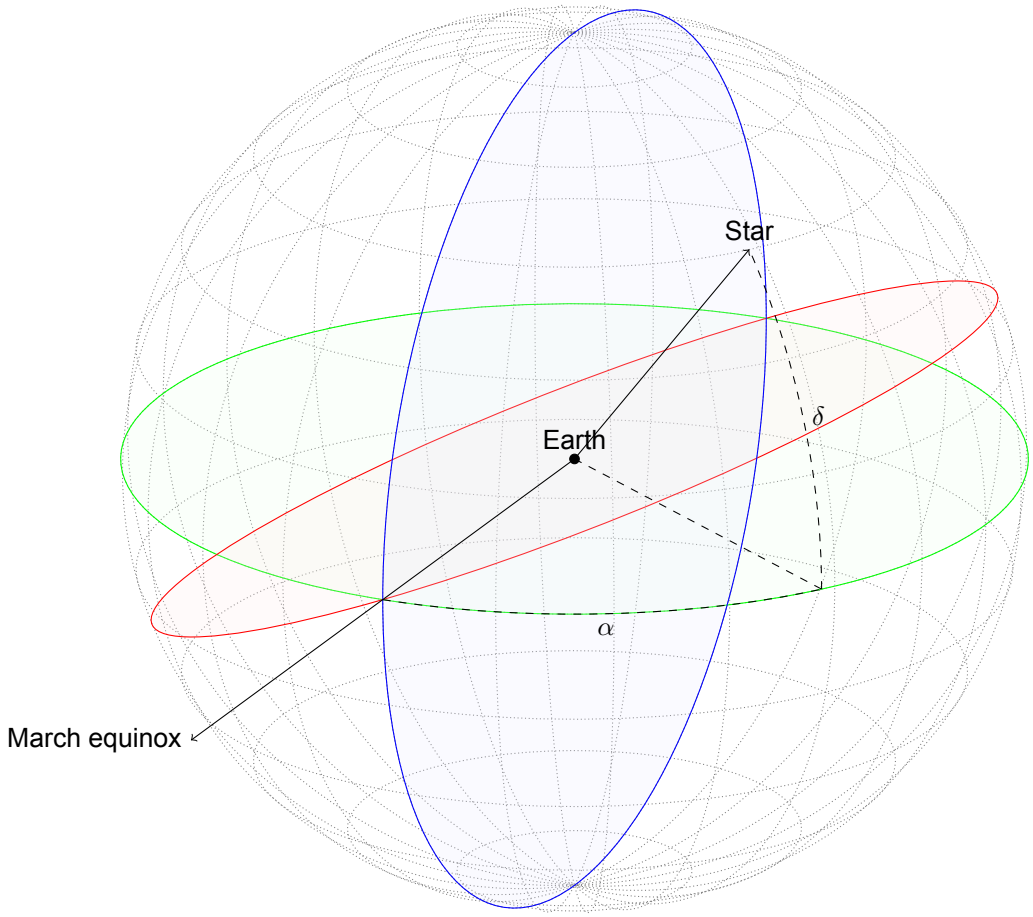


Figure 4.2: A geometric overview of the right ascension α and declination δ coordinates. The green plane represents the Earth's equatorial plane, the red plane represents the ecliptic plane and the blue plane represents the plane where right ascension is zero (or 12 h). The March equinox is also shown, which is the direction in which the right ascension and the declination are zero. To locate a star in the sky, only the right ascension and the declination are necessary.

Once the parallax, the right ascension, and the declination have been measured, the three-dimensional position of a star is known. Therefore, the next step is to discuss the velocity of the star.

4.1.3 Proper motion

The proper motion of a star is the apparent motion of the star in the sky. There are thus two directions of proper motion, which are the proper motion in the right ascension direction and the proper motion in the declination direction. For both the right ascension and the declination, the proper motion in the respective direction is the change of the angle over some time. The Gaia missions have started with collecting data since 2013, so there have been multiple years for the measurement of the proper motions. Both proper motions are given in units of mas/year. The mas stands for milliarcsecond, where arcsecond stand for $\frac{1}{3600}$ of 1°, and thus milliarcsecond stands for $\frac{1}{3600000}$ of 1°, and where 1° is equal to $\pi/180$ radians. The accuracy of the Gaia telescope (for bright enough stars) is 24 microarcseconds, thus it is possible to measure in the milliarcsecond scale[20].

With the proper motion measured, there is one remaining velocity parameter, the radial velocity, which will be discussed next.

4.1.4 Doppler shift

To measure the radial velocity of stars, the wavelength of the light emitted by the stars becomes important. When stars move away from Earth, the wavelength is stretched out resulting in a larger wavelength, and when stars move towards Earth, the wavelength is compressed and thus the wavelength gets shorter. The spectrum of the, already Doppler-shifted, emitted light from the stars can then be matched to known spectra of elements at rest. Then by determining the magnitude of the shifted spectrum, the radial velocity can be calculated.

4.1.5 Mass of the stars

Now that the position and velocity are known, the mass of the stars needs to be determined. There are several ways in which the mass of a star can be determined. The first is the mass-luminosity relation, given by:

$$\frac{L}{L_\odot} = \left(\frac{M}{M_\odot}\right)^{3.5} \tag{4.2}$$

where L is the luminosity of the star, $L_\odot$ is the luminosity of the Sun, M is the mass of the star and $M_\odot$ is the mass of the Sun. The power term of 3.5 is valid for most stars, but for red giants and white dwarfs, a different power term should be chosen. Another way to derive the masses of the stars in a binary system is by using Kepler's third law:

$$\frac{P^2}{a^3} = \frac{4\pi^2}{G(M_1 + M_2)} \tag{4.3}$$

where P is the orbital period, a is the semi-major axis, M_1 and M_2 are the masses of the first and the second star and G is the gravitational constant. Therefore, when P and a are known, the total mass of the binary system is known. Then by measuring how far the stars are removed from the center of mass/rotational axis, the individual star mass can be calculated:

$$a_1 M_1 = a_2 M_2 \tag{4.4}$$

where a_1 and a_2 are the distances from the corresponding star to the center of mass. Together with the total mass $M = M_1 + M_2$, the individual masses can be calculated.

4.2. From Gaia data to useful data

Now that it is clear what the Gaia data represents, it needs to be put into Cartesian coordinates. It also needs to be put into a certain orientation, because the external field needs to be aligned along the z-axis. First, the position vector of a star with Earth as its starting point needs to be put into Cartesian coordinates:

$$\boldsymbol{r}_{\text{Earth-star}} = \begin{pmatrix} \cos(\alpha)\cos(\delta) \\ \sin(\alpha)\cos(\delta) \\ \sin(\delta) \end{pmatrix} \cdot r \tag{4.5}$$

with $\boldsymbol{r}_{\text{Earth-star}}$ the vector from Earth to the star, α the right ascension, δ the declination and r the distance from Earth to the star given by:

$$r = \frac{1 \text{ AU}}{\tan\beta} \tag{4.6}$$

where β is the parallax.

A similar procedure can be done to get the vector from Earth to the center of the Milky Way, giving the vector $\boldsymbol{r}_{\text{Earth-Milky Way}}$. Then the vector from the center of the Milky Way to the star is given by: $\boldsymbol{r}_{\text{Milky Way-star}} = \boldsymbol{r}_{\text{Earth-star}} - \boldsymbol{r}_{\text{Earth-Milky Way}}$. When taking a weighted average of the position vectors, the direction from the center of mass to the center of the Milky Way, $\boldsymbol{r}_{\text{c.o.m.-Milky Way}}$, is known.

The velocity vector also needs to be put into Cartesian coordinates, which is given by:

$$\boldsymbol{v} = \begin{pmatrix} -r\sin(\alpha)\tan(v_\alpha) - r\cos(\alpha)\sin(\delta)\tan(v_\delta) + \cos(\alpha)\cos(\delta)v_r \\ r\cos(\alpha)\tan(v_\alpha) - r\sin(\alpha)\sin(\delta)\tan(v_\delta) + \sin(\alpha)\cos(\delta)v_r \\ r\cos(\delta)\tan(v_\delta) + \sin(\delta)v_r \end{pmatrix} \tag{4.7}$$

with v_α the proper motion in the right ascension direction, v_δ the proper motion in the declination direction, v_r the velocity in the radial direction and r the distance from the Earth to the star.

Now that the position and the velocity have been put into Cartesian coordinates, it is also useful to use a Galilean transformation to put the center of mass at the origin and to give the center of mass no velocity. This is done in the following way:

$$\boldsymbol{r}_1 = \boldsymbol{r}_{\text{Milky Way-star-1}} - \frac{M_1 \boldsymbol{r}_{\text{Milky Way-star-1}} + M_2 \boldsymbol{r}_{\text{Milky Way-star-2}}}{M_1 + M_2} \tag{4.8}$$

and for star 2:

$$\boldsymbol{r}_2 = \boldsymbol{r}_{\text{Milky Way-star-2}} - \frac{M_1 \boldsymbol{r}_{\text{Milky Way-star-1}} + M_2 \boldsymbol{r}_{\text{Milky Way-star-2}}}{M_1 + M_2} \tag{4.9}$$

This can also be done for the velocity:

$$\boldsymbol{v}_1 = \boldsymbol{v}_{1,\text{old}} - \frac{M_1 \boldsymbol{v}_{1,\text{old}} + M_2 \boldsymbol{v}_{2,\text{old}}}{M_1 + M_2} \tag{4.10}$$

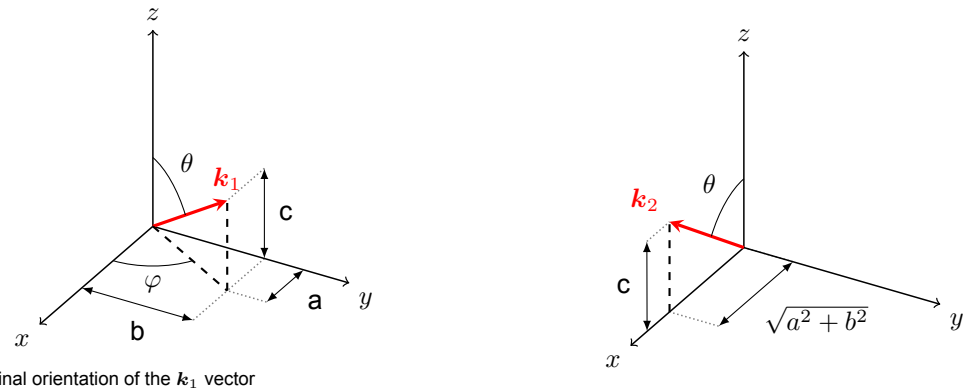
and for star 2:

$$\boldsymbol{v}_2 = \boldsymbol{v}_{2,\text{old}} - \frac{M_1 \boldsymbol{v}_{1,\text{old}} + M_2 \boldsymbol{v}_{2,\text{old}}}{M_1 + M_2} \tag{4.11}$$

The direction to the center of the Milky Way is still the same as before the Galilean transformation. As will be mentioned in the next chapter, the simulation needs to align the external field (direction to the center of the Milky Way) to the z-axis. To do this, an arbitrary vector representing the original direction to the center of the Milky Way $\boldsymbol{k}_1 = \begin{pmatrix} a \\ b \\ c \end{pmatrix}$ is rotated to align with the z-axis by first rotating, in the x-y plane, this arbitrary vector over an angle $\varphi = -\arctan(b/a)$:

$$\boldsymbol{k}_2 = \begin{pmatrix} \cos(\varphi) & -\sin(\varphi) & 0 \\ \sin(\varphi) & \cos(\varphi) & 0 \\ 0 & 0 & 1 \end{pmatrix} \boldsymbol{k}_1 \tag{4.12}$$

Now $\boldsymbol{k}_2$ is rotated into the x-z plane as shown in figure (4.3).



(a) The original orientation of the $\boldsymbol{k}_1$ vector with polar angle θ and azimuthal angle φ.

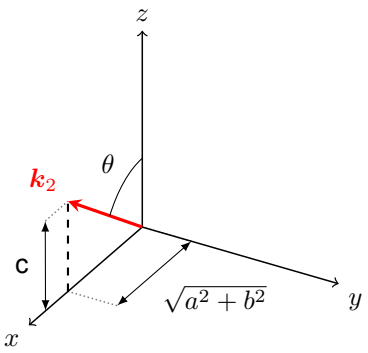
(b) The orientation of the $\boldsymbol{k}_2$ vector with polar angle θ.

Figure 4.3: The rotation along the z-axis over an angle $\varphi = -\arctan(b/a)$ of the $\boldsymbol{k}_1$ vector in (a) to the $\boldsymbol{k}_2$ vector in (b).

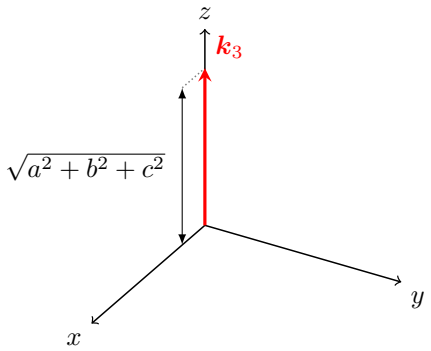
Then it needs to be rotated, in the x-z plane, over an angle $\theta = -\arctan\left(\sqrt{a^2 + b^2}/c\right)$:

$$\boldsymbol{k}_3 = \begin{pmatrix} \cos(\theta) & 0 & -\sin(\theta) \\ 0 & 1 & 0 \\ \sin(\theta) & 0 & \cos(\theta) \end{pmatrix} \boldsymbol{k}_2 \tag{4.13}$$

with $\boldsymbol{k}_3$ pointing in the z-direction, as shown in figure (4.4).



(a) The orientation of the $\boldsymbol{k}_2$ vector with polar angle θ.



(b) The orientation of the $\boldsymbol{k}_3$ vector which is parallel to the z-axis.

Figure 4.4: The rotation along the z-axis over an angle $\theta = -\arctan\left(\sqrt{a^2+b^2}/c\right)$ of the $\boldsymbol{k}_2$ vector in (a) to the $\boldsymbol{k}_3$ vector in (b).

Thus $\boldsymbol{r}_1$, $\boldsymbol{r}_2$, $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$ need to be rotated along the angles φ and θ in a similar manner as $\boldsymbol{k}_1$ where $\boldsymbol{k_1}$ represents the position vector to the center of the Milky Way $\boldsymbol{r}_{\text{c.o.m.-Milky Way}}$. Nothing has to be done extra to the mass of the stars. The seven variables to describe a star in space have now been determined so that they can be used in the simulations.

4.3. Quality of data

Once the data has been processed, the question should be asked if the data is of high enough quality. This means that the data is a reliable and accurate representation of the actual stars. Several factors can influence the quality of the data. Firstly, the data selection procedure, as also described in Pittordis and Sutherland, is explained. Secondly, flyby events will be discussed. At last, the uncertainty of the measurements will be discussed.

4.3.1 Data selection

The Gaia Early Data Release 3 (EDR3) has made available data of about 1.8 billion stars in the Milky Way. Not all of these stars are part of a wide binary system. Therefore the right data must be selected so that wide binary systems are left. To do this, Pittordis and Sutherland started with selecting stars which have a parallax $\omega > \frac{10}{3}$ mas, which is all stars which have a distance < 300 pc. Hereafter, stars were selected which have a Gaia broadband magnitude of $G < 17$. A decrease in the magnitude G value corresponds to an increase in the apparent brightness of the star. Thus stars that were bright enough were selected. Then another quality cut was made where all stars with an absolute declination $|\delta| < 15°$ were removed from the data sample. This $|\delta| < 15°$ cut was made because, in this region in the sky, there is a high star density due to the galactic plane. Several other parts of the sky were removed because high-density open clusters were located there. Thus by removing this part of the sky, contamination issues around high densities are removed[21]. The parts of the sky that were removed due to these clusters are shown in table (4.1).

Table 4.1: The list of sky coordinates that are removed from the wide binary selection. This was done to remove contamination issues around high-density sky regions.

Cluster	Right ascension limits (degree)	Declination limits (degree)
Blanco 1	(0,2)	(-31,-29)
Pleiades	(54,59)	(+22,+27)
Praesepe	(129,131)	(+18,+22)
Upper Scorpius	(238,248)	(-29,-19)

These quality cuts resulted in a sample with 2,101,920 stars.

Then Pittordis and Sutherland searched for wide binary systems by selecting all stars that had another star in a ≤ 50 kAU projected distance, taking into account consistent parallax distances. The

parallax consistency was checked by restricting that the difference in parallax distances was smaller than four times the combined uncertainty of the distances.

$$|d_1 - d_2| \leq 4\sqrt{\sigma_{d_1}^2 + \sigma_{d_2}^2} = 4\sigma_d \tag{4.14}$$

with d_1, d_2 the star distances, σ_{d_1}, σ_{d_2} the distance uncertainties and σ_d the combined uncertainty. For these selected stars, the stars which had a projected velocity difference $|\boldsymbol{v}_1 - \boldsymbol{v}_2| \leq 3$ km/s were selected. This search for wide binary systems resulted in a sample with 92,677 candidate wide binaries. If a star was present in multiple wide binary candidates, then one or both candidates were removed. This was done to account for binary systems which had more than two stars. To account for poor quality of measurements and calibration errors, another quality cut has been made which is based on the methods developed in Arenou et al. (2018) [22]. After this final quality cut, the sample that has been analyzed contains 73,159 wide binary candidates.

4.3.2 Flyby events

Not all wide binary candidates which have been selected are real wide binaries. There is a possibility that a so-called flyby event has been sampled as a wide binary candidate. A flyby event happens when two stars, which do not form a binary system, approach each other in such a way that, despite all quality cuts, is detected as a binary candidate. In Newtonian dynamics, a flyby event happens when the total mechanical energy is positive:

$$0 < E = K + U = \frac{|\boldsymbol{v}_1 - \boldsymbol{v}_2|^2}{2} - \frac{G(m_1 + m_2)}{|\boldsymbol{r}_1 - \boldsymbol{r}_2|^2} \tag{4.15}$$

The stars from the flyby event follow unbounded hyperbolic trajectories. Whether a candidate wide binary is a real wide binary needs to be determined by simulating the trajectory of the stars in the wide binary system.

4.3.3 Measurement uncertainty

The quality of the data is dependent on the uncertainty of the measurements. The order of the Gaia measurement uncertainty is given for each of the seven parameters discussed in the 'Gaia data' section. In table (4.2), the uncertainties of all Gaia measurements as discussed in subsection (4.1) are shown. The dependence of these uncertainties on the optical magnitude is also shown.

Table 4.2: The type of measurement on the left with the corresponding uncertainties on the right. The uncertainties of all measurement types are dependent on the optical magnitude G of the star. The fainter the star (higher G value), the more uncertainty in the measurement[23][24].

Parallax	0.02–0.04 mas at $G < 15$ 0.1 mas at G = 17
Right ascension and declination	0.02–0.03 mas at $G < 15$ 0.08 mas at G = 17
Proper motion	0.02−0.04 mas yr^{-1} at $G < 15$ 0.1 mas yr^{-1} at G = 17
Mean G-band photometry	0.3 mmag at $G < 13$ 1 mmag at G = 17
Mean G_{BP}-band photometry	0.9 mmag at $G < 13$ 12 mmag at G = 17
Mean G_{RP}-band photometry	0.6 mmag at $G < 13$ 6 mmag at G = 17
Radial velocity	200-350 m s^{-1} at 4$< G <$8 1.4-3.7 km s^{-1} at G=11.75

The uncertainty as a fraction of the total measurement is also interesting. The minimum parallax used in this research is $\omega = \frac{10}{3}$ mas. Thus the maximum uncertainty as a fraction $\frac{0.1}{10/3} = 0.03$ is 3%. For

the optical magnitudes, this fraction is at most $\frac{0.012}{17} = 0.00071$ is 0.071%. The other measurements for right ascension, proper motion, and radial velocity are dependent on individual cases, where in some cases the fractional uncertainty can become significant, especially when the measurement is close to zero.

5

External field effect

In this chapter three different methods for modelling the external field effect will be discussed. The effect that the external field effect has on the trajectory of the stars in the wide binary will later be examined.

5.1. Modelling the EFE

The first model that will be discussed is the capacitor model, the second model is the Newtonian potential model, and the third is the MOND model.

5.1.1 Capacitor approach

The external field is, by approximation, homogeneous along the trajectory of the center of mass of the wide binary. The electric field which is created by a capacitor is homogeneous between the plates of the capacitor. The Newtonian gravitational field is similar to the electric field from Coulomb's law. Therefore, the external field could be modeled by considering two plates with a constant mass density, as shown in figure (5.1).

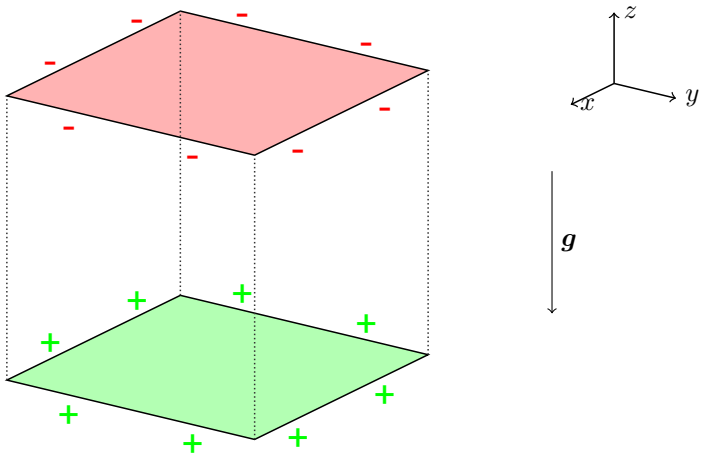


Figure 5.1: Schematic overview of the capacitor, with a negative mass density on the upper plate, and positive on the lower plate. This is done to create an acceleration field $\boldsymbol{g}$ in the negative z direction.

The only difference between the laws of electromagnetism and the Newtonian gravity laws is that there are negative charges but there are no negative masses. It is physically impossible to have negative masses, but in a computer simulation, it is possible to model it. As shown in figure (5.1), when placing a plate with negative mass density at $z = L$, and a plate with the same but positive mass density at $z = 0$, a capacitor is formed. Here the negatively charged plate repels the mass from the stars in the wide binary, and the green plate attracts.

In the simulation, however, an infinitely thin plate is not possible to simulate since the code uses a particle mesh. Therefore, the upper and lower plates are in actuality modeled as plates with a width of one pixel. The adjusted schematic overview of the capacitor is now given in figure (5.2).

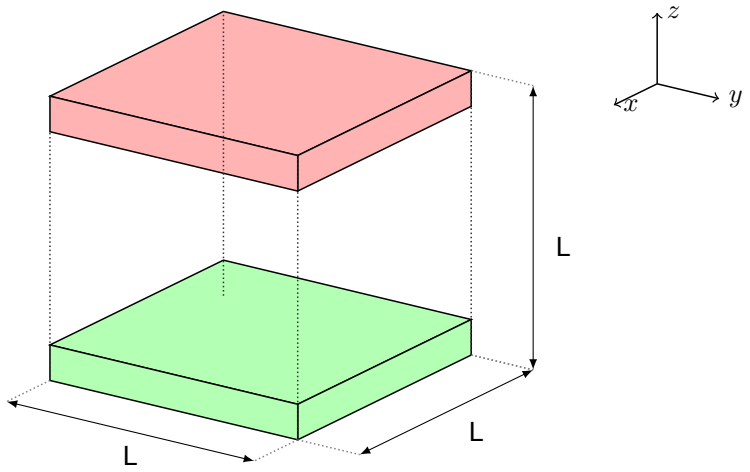


Figure 5.2: Schematic overview of the capacitor with the plates having a small width of one pixel, since the code can not simulate an infinitely thin plate. The upper plate has a negative mass density and the lower plate has a positive mass density. This is done to create an acceleration field in the negative z direction.

For the one capacitor model, as shown in figures (5.1) and (5.2), the strength of the capacitor field is given by:

$$a = \frac{GM_p}{L^2} = \frac{GM_c \cdot p^2}{(p \cdot \Delta L)^2} = \frac{GM_c}{\Delta L^2} \tag{5.1}$$

where a is the strength of the capacitor field, L is the length of the side of the cube, G is the gravitational constant, M_p is the mass of the positive plate, M_c is the mass per pixel, ΔL is the length of the pixel and p is the number of pixels along the x-axis (where the y-axis and z-axis have the same number of pixels since the grid is a cube).

As discussed in chapter (3), the code uses the fast Fourier transform, which introduces periodicity. This implies that the the capacitor plates extend to infinity in the x-axis and the y-axis, and that copies of the capacitor plates appear above and below the original capacitor plates at an interval of L. Therefore the model that is actually in the code due to the periodicity is shown in figure (5.3), where the periodicity in the x-axis and y-axis is not shown.

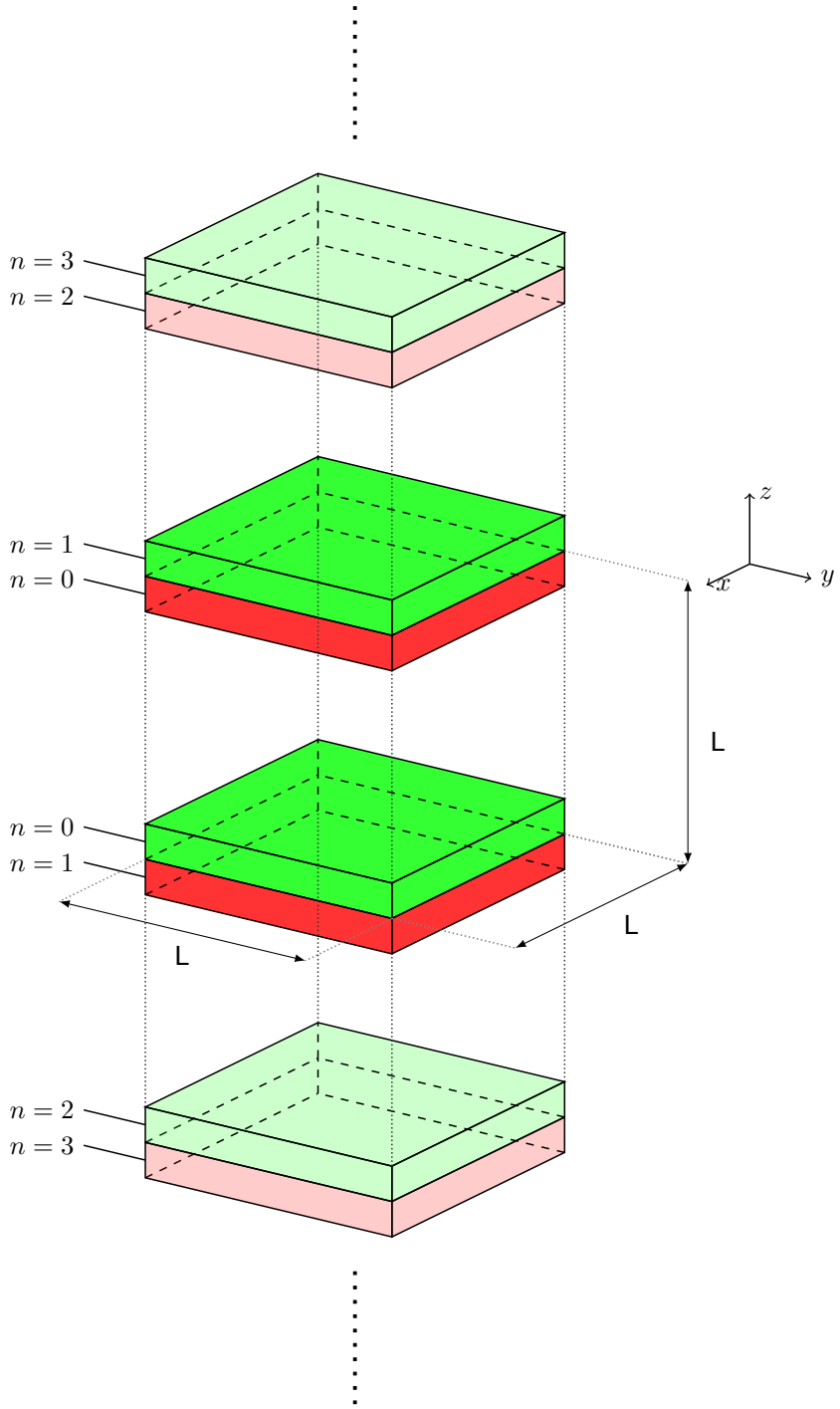


Figure 5.3: Schematic overview of the capacitor with the plates having a small width of one pixel, since the code can not simulate an infinitely thin plate. The capacitors are repeated along the z-axis with periodicity L since the code uses the FFT, which introduces a periodicity. The plates with the same n value form a capacitor. For an even n, the upper plate has a negative mass density and the lower plate has a positive mass density. For an odd n, the upper plate has a positive mass density and the lower plate has a negative mass density. For the $n = 2k$ and $n = 2k + 1$ ($k \in \mathbb{Z}$) pair, the $n = 2k$ capacitor acceleration field is slightly stronger than the $n = 2k + 1$ capacitor acceleration field, since the plates of the $n = 2k$ capacitor have smaller separation. When adding the contribution of all capacitors, the resulting acceleration field will be in the negative z direction.

In figure (5.3), the contribution of each capacitor plate is indicated by the brightness of the plate. The original plates and the first repeated plates contribute the most, as they are closer to the original

space. Now the actual strength of all these capacitors combined is given by:

$$a = \sum_{n=0}^{\infty} (-1)^n \frac{GM_p}{L_n^2} \tag{5.2}$$

where L_n is given by:

$$L_n = \begin{cases} 2(\frac{p}{2} - 1 + p\frac{n}{2})\Delta L, & \text{for } n \text{ even} \\ 2(\frac{p}{2} + p\frac{n-1}{2})\Delta L, & \text{for } n \text{ odd} \end{cases} \tag{5.3}$$

and where the capacitor with the value $n = 0$ is the original capacitor as seen in figure (5.2), and where $n = 1$ is the first removed capacitor , $n = 2$ is the second removed capacitor, and so on. The sign of the mass in the plate is reversed for all odd n capacitors.

After the $n = 50$ capacitor, the effect of adding the contribution of more capacitors becomes insignificant.

The resulting potential along the z-direction together with the analytic Newtonian potential and their difference, is given in figure (5.4). The analytic potential has been scaled by dividing it by $\mu(a/a_0)$. In this figure, $x = y = p/2$ and no stars are simulated.

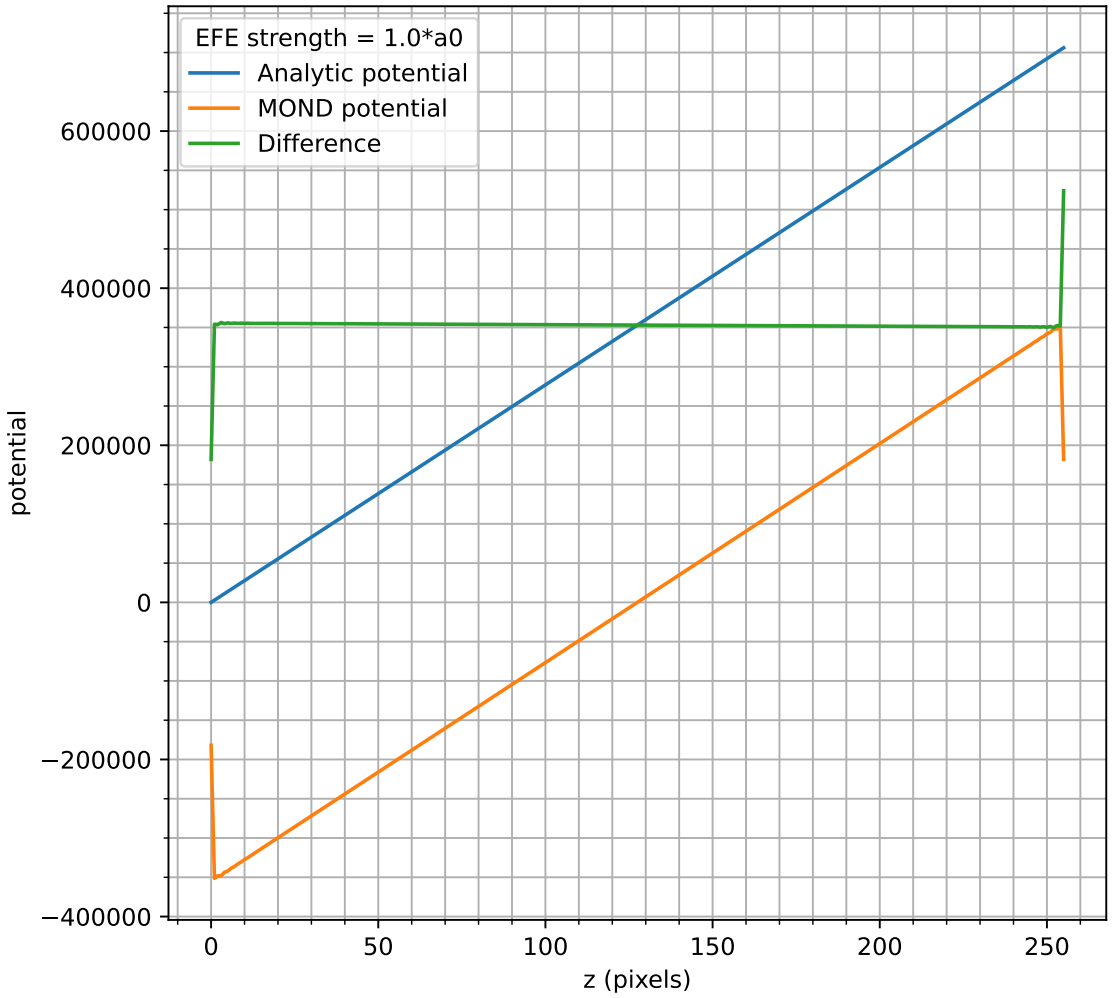


Figure 5.4: The potential resulting from the capacitor model plotted against the z coordinate in the grid. For this plot, $x = y = p/2$ and no stars are simulated. The blue line is the analytic potential, the orange line is the MOND potential, and the green line is the difference between the blue and orange lines. The blue and orange lines seem to be shifted, but this is not problematic since shifting a potential by a constant does not change the physics. A small downward slope is visible in the green line, which is problematic since this changes the physics.

In figure (5.4), there is an almost constant difference. This is because the algorithm shifts the MOND

potential such that the average value of the MOND potential is zero. This difference, however, is not problematic since the acceleration field $\boldsymbol{a} = -\boldsymbol{\nabla}\Phi = -\boldsymbol{\nabla}(\Phi + \text{const.})$ does not change when you add a constant to the potential.

What is most important is that the difference line slightly slopes downwards, which implies that there is a significant difference between the analytic and actual solution.

5.1.2 Changing the potential

Instead of creating an external field by modeling a capacitor with positive and negative masses, the potential could directly be changed. If we want to model the external field in the z-direction, then we need to add $z \cdot a$ to the Newtonian potential. The resulting potential is given in figure (5.5), together with the analytic potential and the difference. The analytic potential has again been scaled by dividing it by $\mu(a/a_0)$. Again, $x = y = p/2$ and no stars are simulated.

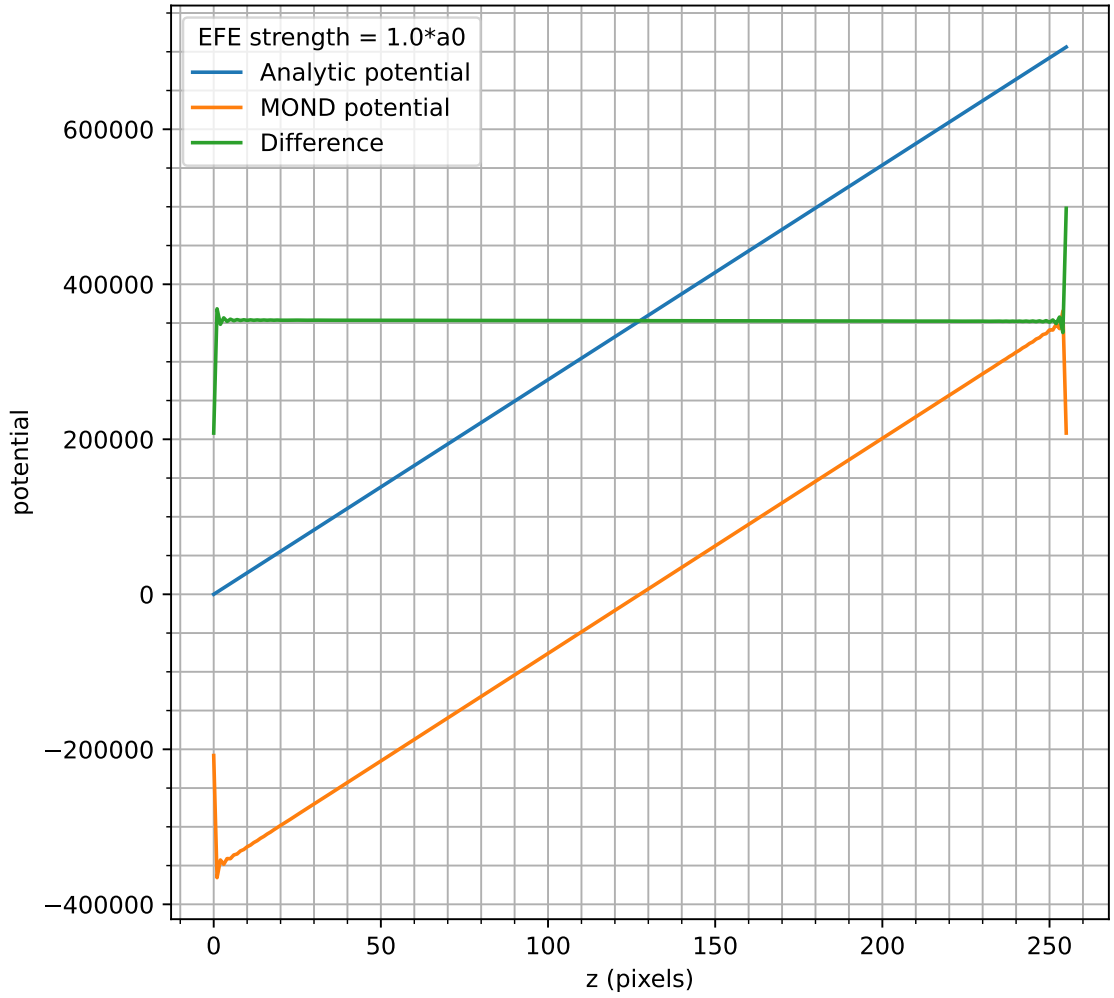


Figure 5.5: The potential resulting from changing the potential model plotted against the z coordinate in the grid. For this plot, $x = y = p/2$ and no stars are simulated. The blue line is the analytic potential, the orange line is the MOND potential, and the green line is the difference between the blue and orange lines. The blue and orange lines seem to be shifted, but this is not problematic since shifting a potential by a constant does not change the physics. There is no slope visible in the green line, which means that this model results in a realistic model for the external field. Large oscillations are visible at the boundary, but this is not problematic since the stars will be far removed from the edge.

In figure (5.5), the difference has some oscillations at the sides of the grid and is constant in the middle of the grid. Therefore this model is preferred over the capacitor model, as the constant behaviour in the middle of the grid is preferred over a slightly changing potential. The stars in the simulations won't come close to the side of the grid, so the oscillations there are insignificant.

5.1.3 Changing the MOND algorithm

Another method to model the external field is to change the MOND algorithm. This is done by adding an extra step between steps 7(c) and 7(d) in the complete MOND method list (7g). This extra step is to add a constant acceleration $\boldsymbol{a}_{\text{EF}} = (0, 0, a)^\intercal$ to the curl-free acceleration field $\boldsymbol{g}_c$. The resulting MOND potential, analytic potential, and their difference is shown in figure (5.6). The analytic potential has been scaled by dividing it by $\mu(a/a_0)$. Again, $x = y = p/2$ and no stars are simulated.

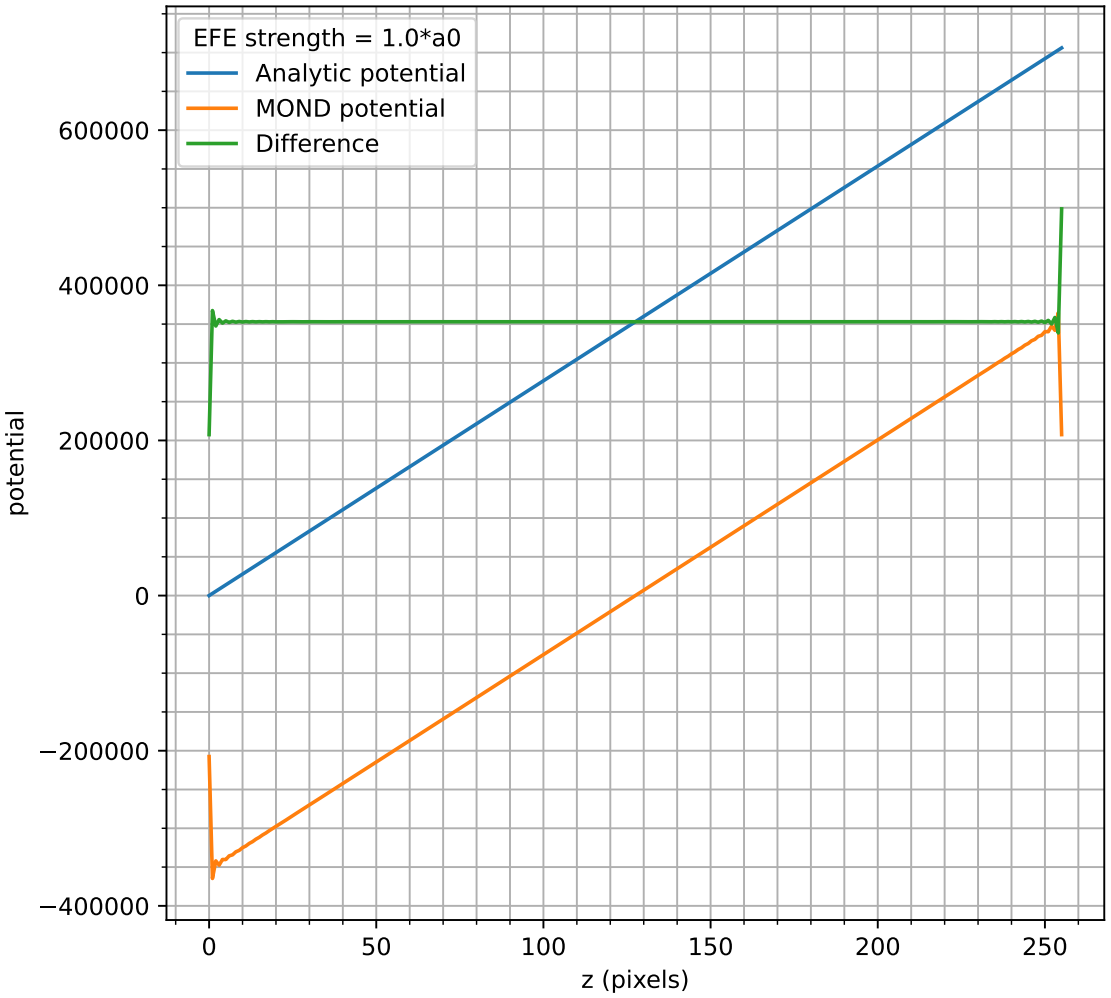


Figure 5.6: The potential resulting from the MOND algorithm change model plotted against the z coordinate in the grid. For this plot, $x = y = p/2$ and no stars are simulated. The blue line is the analytic potential, the orange line is the MOND potential, and the green line is the difference between the blue and orange lines. The blue and orange lines seem to be shifted, but this is not problematic since shifting a potential by a constant does not change the physics. There is no slope visible in the green line, which means that this model results in a realistic model for the external field. Large oscillations are visible at the boundary, but this is not problematic since the stars will be far removed from the edge.

In figure (5.6), the difference has some oscillations at the sides of the grid and is constant in the middle of the grid. This model looks similar to the changing the Newtonian potential model. This model is also preferred over the capacitor model, as the constant behaviour in the middle of the grid is preferred over a slightly changing potential.

6

Simulations

In this chapter, the results for several wide binary simulations will be shown. At first, the Newton model will be shown. The Newton model is obtained by setting $\mu = 1$. For the Newton model, the effect of adding an external field will not be noticeable, since the Poisson equation for Newtonian dynamics is linear. Secondly, the MOND model without external field will be modeled. In this simulation, the standard interpolation function will be used, which is $\mu(x) = x/\sqrt{1+x^2}$. Lastly, the MOND model with the external field will be simulated, again using the standard interpolation function.

In these three different models, several wide binary systems will be shown, with different separations between the stars. An important property is the MOND radius, given by:

$$r_{\text{MOND}} = \sqrt{\frac{GM}{a_0}} \tag{6.1}$$

which is the distance from some star with mass M, such that the Newtonian acceleration will be a_0. This gives a good indication if the system is Newtonian, at the border of Newton to MOND or in deep MOND.

One of the simulations will show a wide binary system which is in the Newtonian regime, which is when the separation of the stars is smaller than both MOND radii. Another simulation is at the transition, which is when the MOND radii are near the separation. The last simulation is for the deep MOND regime, which is when both MOND radii are smaller than the star separation. This is done to show the effect that MOND theory has on the stability of the orbits of the wide binary. Table (6.1) shows the MOND radii for both stars and their initial separation for each selected wide binary system.

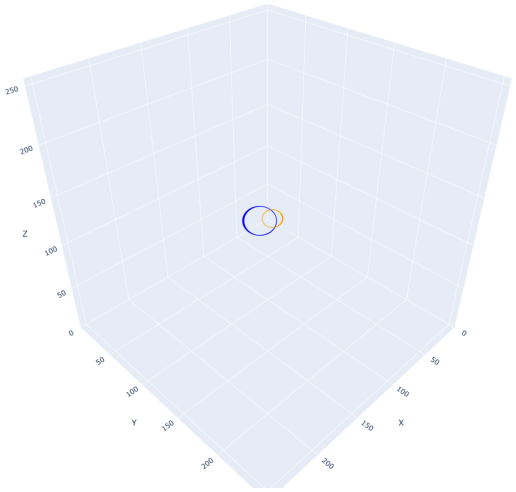
Table 6.1: The MOND radii for both stars in the wide binary system, and the initial separation of these stars are listed for the selected wide binary systems. The MOND effects become visible when the separation between the stars is larger than both MOND radii. When the MOND radii are close to the separation, these MOND effects are in the transition from the Newtonian regime to the deep MOND regime.

	MOND radius 1 (AU)	MOND radius 2 (AU)	Initial separation (AU)
Newtonian regime	5426	6932	1578
Transition regime	5392	5466	5100
Deep MOND regime	6438	6073	28893

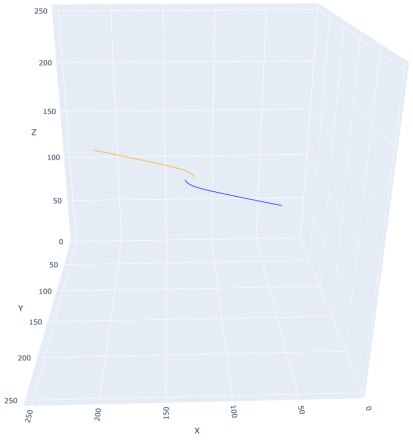
Each simulation has a grid of $256 \times 256 \times 256$ pixels. The number of time steps ranges between 1000 and 4000, and the size of one time step ranges from 1 kyr to 10 kyr, depending on which wide binary is simulated.

6.1. Newton model

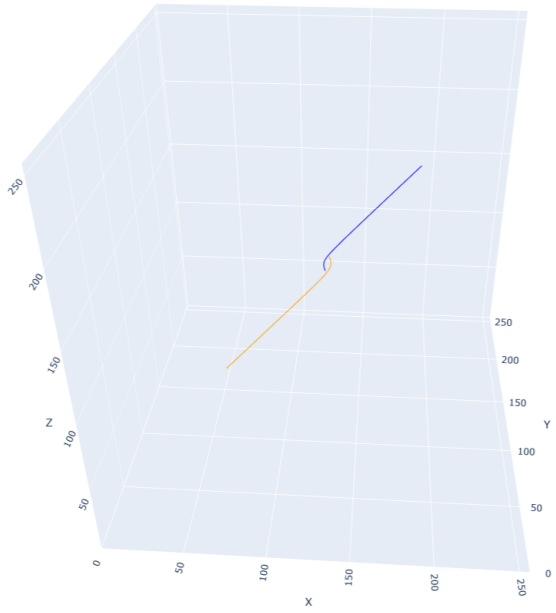
In this section, the results of the Newton model will be shown. In figures (6.1a), (6.1b) and (6.1c), the orbits of the three different regimes (Newtonian, transition, deep MOND) will be shown, starting with the Newtonian regime.



(a) The orbit of a wide binary in the Newtonian regime. The time step size is 1 kyr, and 1000 time steps were simulated.



(b) The trajectory of a candidate wide binary system in the transition regime. The time step size is 1 kyr, and 1000 time steps were simulated.



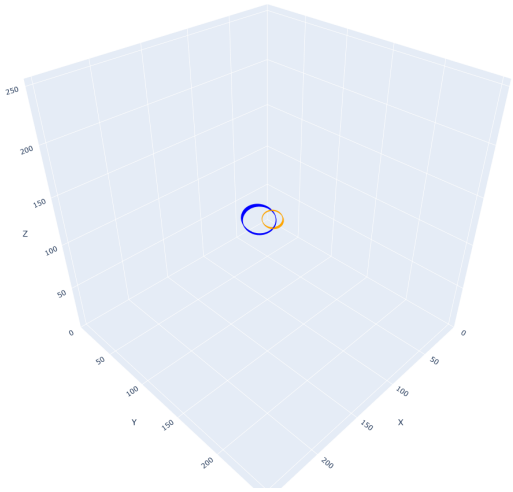
(c) The trajectory of a candidate wide binary system in the deep MOND regime. The time step size is 10 kyr, and 1000 time steps were simulated.

Figure 6.1: The orbits of the Newton model are shown. In (a), (b), and (c), the orbits of the Newtonian, transition, and deep MOND regimes are shown respectively. The grid is $256 \times 256 \times 256$ pixels.

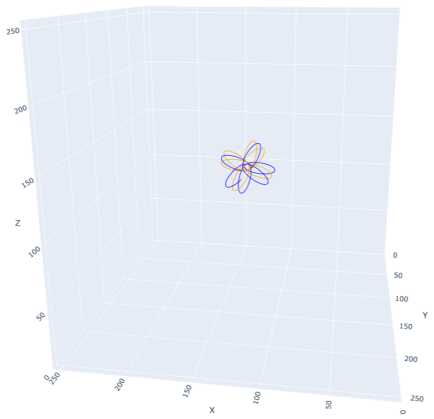
Figure (6.1) shows one stable orbit from the Newtonian regime, and two unstable orbits from the transition and deep MOND regimes. The internal accelerations from the Newtonian regime simulation are larger than a_0, as the initial separation is smaller than both MOND radii. For the other two, this is not the case and the orbits are unstable. It may be that the transition and deep MOND regimes are flyby-like events in reality.

6.2. MOND without external field model

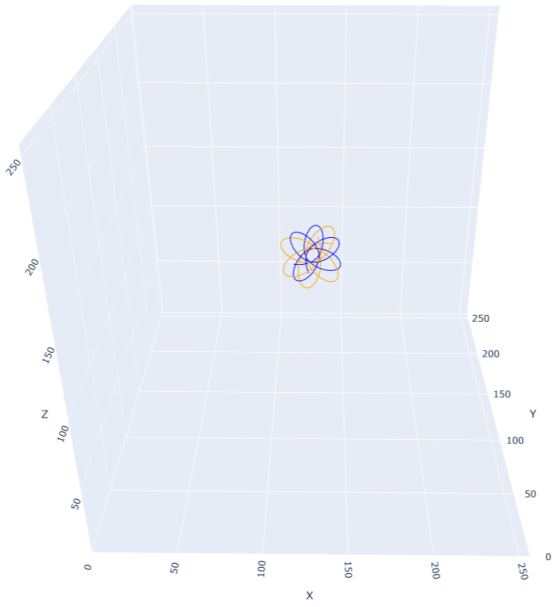
The MOND model without an external field shows the effect that the MOND theory has on orbits of wide binaries. The result of the Newtonian, transition, and deep MOND regimes are shown in figures (6.2a), (6.2b), and (6.2c) respectively.



(a) The orbit of a wide binary in the Newtonian regime. The time step size is 1 kyr, and 1000 time steps were simulated.



(b) The orbit of a wide binary in the transition regime. The time step size is 1 kyr, and 1000 time steps were simulated.



(c) The orbit of a wide binary in the deep MOND regime. The time step size is 10 kyr, and 1000 time steps were simulated.

Figure 6.2: The orbits of the MOND without external field model are shown. In (a), (b), and (c), the orbits of the Newtonian, transition, and deep MOND regimes are shown respectively. The grid is $256 \times 256 \times 256$ pixels.

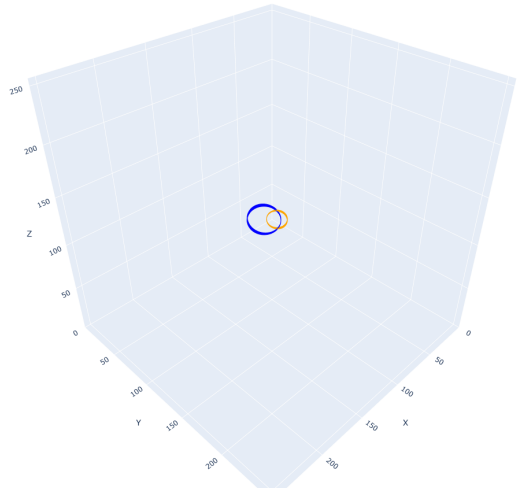
In figure (6.2), all orbits are stable. This is expected since the wide binaries were selected based on this fact. In the database, there were also a lot of flyby-like events with the MOND without EF model. These were not selected because they would be flyby-like for each model. Only the stable orbits

were selected such that they could be compared to the other models. Some of the flyby-like events in the Newton model can result in stable orbits in the MOND without EF model. When comparing the Newtonian regime orbit from the MOND without EF model to the Newtonian regime orbit from the Newton model, almost nothing changes. This is because in the Newton model $\mu = 1$, and in the MOND model with the Newtonian regime μ is close to 1, as the internal accelerations are larger than a_0. Therefore, there is almost no difference between these orbits. The trajectories of the transition and deep MOND regimes are now stable. The MOND theory has the effect of increasing the internal accelerations, resulting in a stable orbit as can be seen in figure (6.2). It is thus an interesting result that the MOND without EF model can give stable orbits, whereas the Newton model gave unstable orbits.

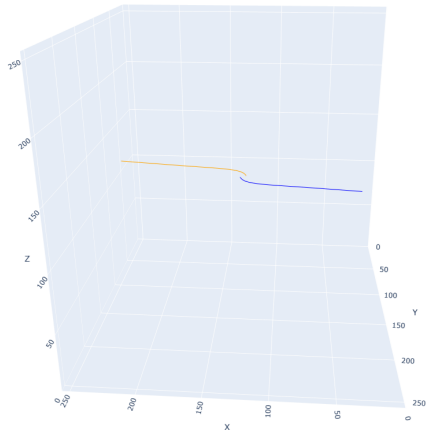
More measurements over a longer time are needed to conclude whether the real orbits look like the orbits from figure (6.2), or that the trajectories look more like the flyby-like event from figure (6.1). The time step sizes are in the order of 10^3 years, making the orbit time in the order of 10^4 and 10^5 years. High-precision measurements are available for about a period of 20 years, making the small deviations between figures (6.1) and (6.2) possible.

6.3. MOND with external field model

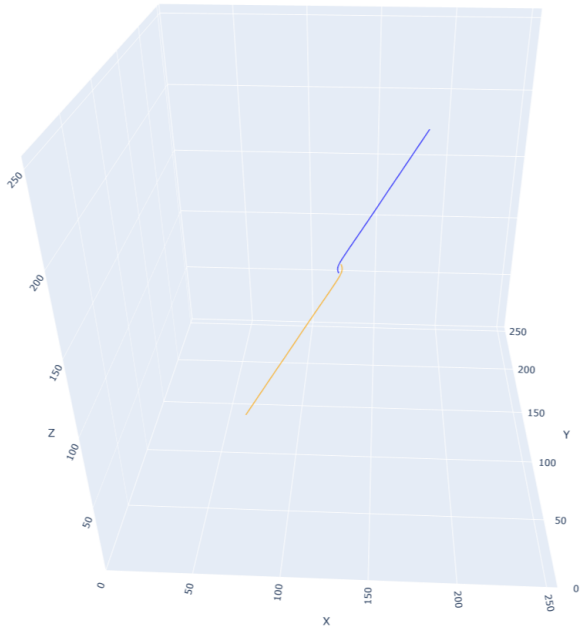
In the MOND model with the effect of an external field, the dynamics of the orbits may change. The strength of the external field was taken to be the same strength as the sun experiences, which is about $1.6a_0$. For this external field strength, the interpolation function is already on the Newtonian side, since the internal + external acceleration is larger than a_0. The result of the Newtonian regime, the transition regime, and the deep MOND regime are shown in figures (6.3a), (6.3b) and (6.3c) respectively.



(a) The orbit of a wide binary in the Newtonian regime. The time step size is 1 kyr, and 1000 time steps were simulated.



(b) The trajectory of a candidate wide binary system in the transition regime. The time step size is 1 kyr, and 1000 time steps were simulated.



(c) The trajectory of a candidate wide binary system in the deep MOND regime. The time step size is 10 kyr, and 1000 time steps were simulated.

Figure 6.3: The orbits of the MOND with external field model are shown. In (a), (b), and (c), the orbits of the Newtonian, transition, and deep MOND regimes are shown respectively. The external field strength is $1.6a_0$, which is the same strength that the Sun experiences. The external field is modeled by changing the Newtonian potential. The grid is $256 \times 256 \times 256$ pixels.

In figure (6.3), the only stable orbit is the orbit from the Newtonian regime. The addition of an external field with a strength $1.6a_0$ which is larger than a_0 causes the interpolation function to move closer to the Newtonian dynamics ($\mu = 1$). The Newtonian regime is thus still stable because the internal acceleration already was in the Newtonian regime. This implies that the interpolation function already was behaving like Newtonian dynamics. Since Newtonian dynamics is linear, the effect of adding an external field on the internal dynamics is unnoticeable. The effect of adding an external field to the transition and deep MOND regime cases, however, causes the interpolation function to move from the transition/deep MOND dynamics to Newtonian dynamics. As seen from figure (6.3), the orbits of these two cases are unstable which is in confirmation with the Newtonian model from figure (6.1).

A closer comparison between figures (6.1) and (6.3) shows that the deep MOND regime trajectories are almost unchanged, while the transition regime trajectory changes slightly. The transition regime trajectory is more horizontal (same z value) when an external field is applied.

6.4. Overview of results

In table (6.2), the stability results of the above-mentioned simulations are shown.

Table 6.2: Overview of the stability of the orbit for the different models and the different regimes. The Newtonian regime and the MOND without external field model are always stable. The MOND with external field model introduces an external field with strength $1.6a_0$, which causes the system to behave more Newtonian, resulting in the same column as the Newton model.

Regime \ Model	Newton	MOND without external field	MOND with external field
Newtonian	Stable	Stable	Stable
Transition	Unstable	Stable	Unstable
Deep MOND	Unstable	Stable	Unstable

The columns of the Newtonian model and the MOND with EF model are the same. This can be explained by the fact that the added external field causes the total acceleration on a particle to be above the a_0 acceleration. This is because the external field strength is $1.6a_0$. Therefore, the interpolation function μ will behave Newtonian-like, resulting in the same results as the Newton model.

In figure (6.4), 25 more simulations were performed on a $128 \times 128 \times 128$ pixels grid. The results are split up into the three different models and in five MOND radii to separation fractions f. f is defined to be the average of the two MOND radii, divided by the initial separation. For the Newtonian regime, $f \gg 1$, while for the deep MOND regime $f \ll 1$. On the y-axis, the fraction of stable orbits is shown.

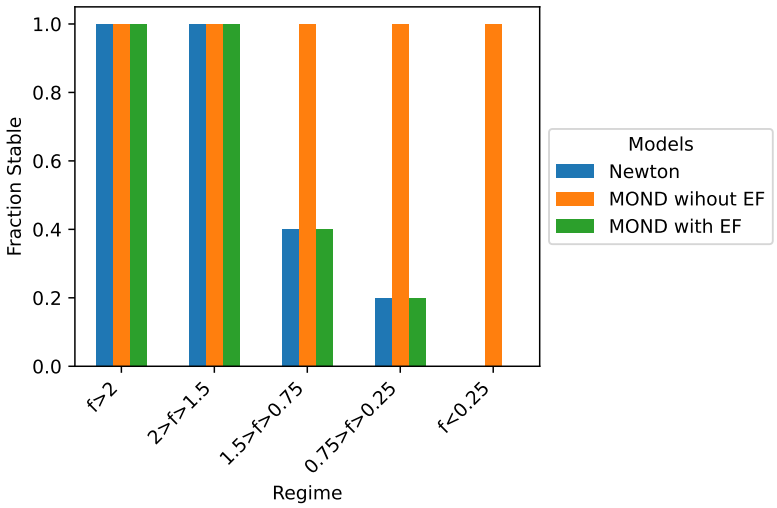


Figure 6.4: On the x-axis the different regimes are shown. The f value indicates the ratio between the average MOND radius and the separation. On the y-axis, the fraction of stable orbits is shown. Thus for the Newtonian regime ($f > 1.5$), all models give stable orbits. For the deep MOND regime ($f < 0.25$), only the MOND without EF model gives stable orbits. For the transition regime ($0.25 < f < 1.5$), a part of the simulations give stable orbits, with the more Newtonian regimes ($1.5 > f > 0.75$) giving a higher fraction.

Figure (6.4) shows that for the Newtonian regimes, all wide binary candidates that have a stable orbit in the MOND without EF model also have a stable orbit in the Newton model and in the MOND with EF model. In the transition regime, a part of the simulations gives stable orbits, with the more Newtonian-like regimes ($1.5 > f > 0.75$) giving a higher fraction of stable orbits. For the deep MOND regime, there are no stable orbits when ignoring the MOND without EF model. Due to the lack of long-time high-precision measurements of the orbits of the wide binaries, no conclusions can be made about whether the MOND models (with and without EF) are accurate. What can be concluded, however, is that the MOND without EF model can give stable orbits where the other models could not.

If stable orbits from the MOND without EF model are observed in the deep MOND regime, and if these orbits match with the simulations, then the MOND theory would gain credibility. If these orbits are not observed, then this is not the end for MOND. It would be unexpected if these orbits were measured because the external field was ignored, which is unphysical. The Newton model and the MOND with EF model give similar results, but the shape of the orbits is slightly different. Therefore, to test MOND, the real orbits in the Newtonian regime need to be compared to the simulated orbits for the Newton model and the MOND with EF model. If the real orbit data matches with the Newton model, then the MOND theory would lose credibility. If, on the other hand, the data matches with the MOND with EF model, then the MOND theory would gain credibility.

6.4.1 Energy conservation

An important physical property is the conservation of energy. As shown in subsection (2.2.1), the total energy E_{tot} can be split up into a kinetic term E_k, and the Lagrangian term L which consist of two potential terms E_{grav} and E_{pot}, where:

$$\begin{aligned} E_k &= \int_\alpha \frac{1}{2}\rho\, v^2 \mathrm{d}^3 r \\ E_{\text{grav}} &= \int_\alpha \frac{a_0^2}{8\pi G} \mathscr{F}\left(\frac{|\boldsymbol{\nabla}\Phi|^2}{a_0^2}\right) \mathrm{d}^3 r \\ E_{\text{pot}} &= \int_\alpha \rho\Phi \mathrm{d}^3 r \\ E_{\text{tot}} &= E_k + E_{\text{grav}} + E_{\text{pot}} \end{aligned} \tag{6.2}$$

In figure (6.5), the energy-time plot for each energy term from equation (6.2) is shown. Figure (6.5) came from the MOND without external field model with the transition regime simulation.

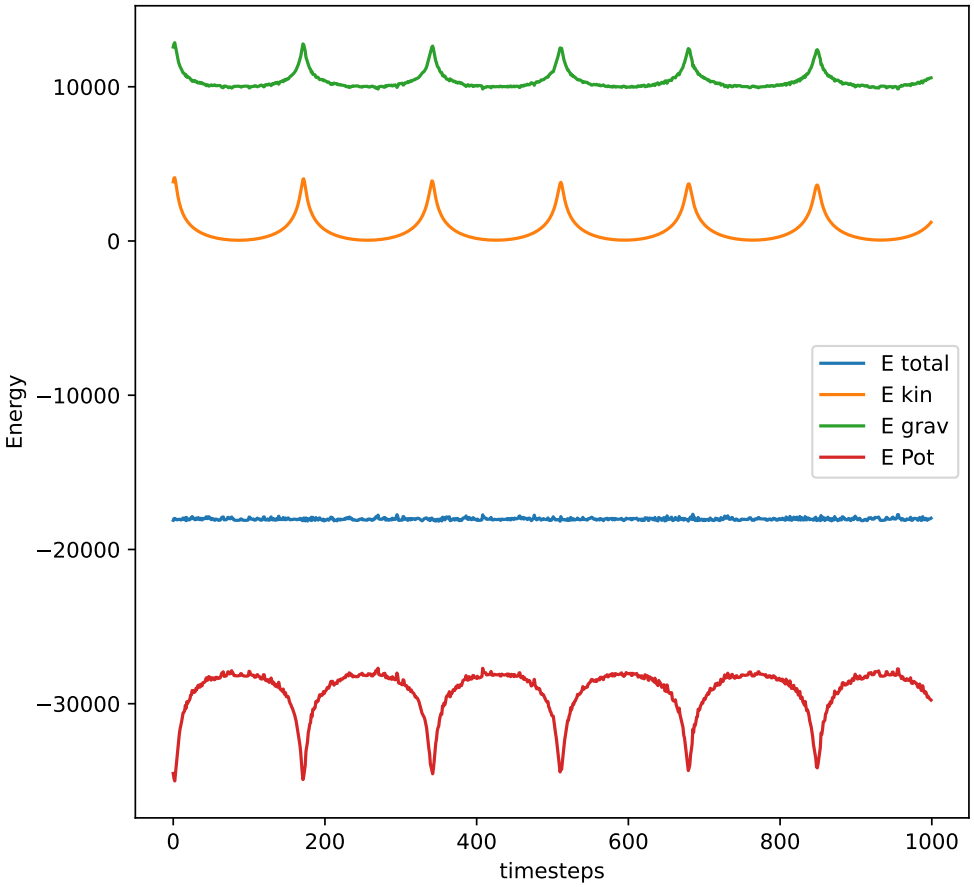


Figure 6.5: The energy-time plot for the simulation with the MOND without EF model, and where the wide binary is in the transition regime. The total energy E_{tot}, kinetic energy E_{kin}, gravitational field energy E_{grav}, and the potential energy E_{pot} are represented by the blue, orange, green and red lines. The total energy is the sum of the other three energies. The total energy is conserved, as can be seen in the figure.

The total energy is conserved as can be seen from figure (6.5). The total energy E_{tot} line is not smooth, which can be explained by the small numerical errors.

7

Research recommendations

In this chapter, research recommendations are given for further research. As the theory of modified Newtonian dynamics is still highly debated, it is of great importance to conduct more thorough research on the topic of wide binaries. The research recommendations have been divided into three subcategories.

The first subcategory is aimed at improving the performance of the code. It is important to improve the performance since more wide binaries can be analyzed in less time, giving a better understanding of real wide binaries. Also, when more pixels can be used, or when simulating more time steps becomes more feasible, the accuracy of the simulation increases.

The second subcategory is the improvement of the accuracy of the simulations. As deviations from Newtonian dynamics may be small, the accuracy of the simulation is of great importance. If for example, the effects of MOND are studied in the solar system, the deviations from Newtonian dynamics are small but measurable. Therefore increasing the accuracy of the simulation is crucial for this follow-up research.

The third subcategory is the extension of the model. This has nothing to do with improving the simulation but is aimed at representing reality in the best way inside the simulation. In this study, several physical aspects were ignored. If these aspects are incorporated into the simulation, the result will become more realistic. The more realistic the simulation, the better the MOND theory can be tested.

7.1. Improving code performance

Improving the performance of the code is strongly recommended. The better the performance of the code, the less time it takes to simulate a wide binary system. The computation time has already been significantly increased when compared to the original computation time from Joost de Nijs. Some research recommendations for speeding up the code from Paul Visser, Stephan Eijt, Joost de Nijs [17] were implemented. Therefore, the remaining research recommendations for improving the performance are listed below:

- Better implementation of the Fast Fourier Transform
- Translating the code into a faster programming language (currently Python)
- Implementing Fast Fourier Transforms which are compatible with float16 on the GPU
- Using sparse arrays

The current implementation of the FFT uses the CuPy library. The CuPy library uses some optimizations from the cuFFT library. The cuFFT library is optimized for computing FFTs on NVIDIA GPUs. Therefore, it may be possible to improve the code by using the cuFFT library to optimize the computation of the FFT. The second improvement may be made by translating the Python code into a faster programming language. Several parts of the code use libraries which are written in C, and thus are already translated. The rest of the code may benefit from a translation to a faster language, but these parts of the code already have low computation time compared to the already translated parts. Another improvement

may be made by making the FFT on the GPU compatible with the float16 datatype. Currently, the CuPy implementation of the FFT is not compatible with float16. If this is implemented, the used memory may be halved, which leads to lower computation times. The last recommended improvement is the use of sparse arrays. Sparse arrays are used when the majority of the array is filled with zeros. The sparse array then only stores the non-zero elements in memory, leading to less memory used when the array contains enough zeros. In this research, the density arrays usually only contain non-zero elements around the stars, while the rest is zero. Therefore, improvement in computation time and memory usage may be achievable by using sparse arrays.

7.2. Increase accuracy

The MOND hypothesis needs to be thoroughly tested. This research is aimed at testing the MOND theory by analyzing wide binary systems. The MOND theory could also be tested by looking for small deviations of a planet's trajectory when compared to the Newtonian dynamics inside the Solar System. For both these cases, the accuracy of the simulation is important, since a small deviation may or may not disprove MOND. Several recommendations for improving the accuracy of the simulation are listed below:

- Improving performance
- Using adaptive time steps
- Reducing self-interaction
- Use higher-order numerical integration methods
- Using the particle mesh Ewald method

As already mentioned, increasing the performance of the code leads to better accuracy because more pixels or smaller time steps can be used without changing computation time. By using more pixels, the simulation has a better resolution. Smaller time steps create more accurate star trajectories, especially when stars are in a close encounter.

Another improvement can be made by introducing adaptive time steps, meaning that the time steps are not constant, but decrease as the stars in the wide binary come closer, and become bigger as the stars separate. This way, the advantage of higher accuracy when there are close encounters is used while not impacting the overall computation time significantly.

In the simulations, the masses of the stars are distributed among the neighboring grid points. Once distributed, these grid points with the distributed mass exert a force on their neighboring grid points. This phenomenon is known as self-gravitation. The effect of self-gravitation is unphysical and should be removed for better accuracy. One could for example test with different mass distribution functions and find the optimal distribution function. Other methods for removing the effect of self-gravitation could be found in Ballenegger et al. [25].

In this research, the leapfrog numerical integration method is used. The leapfrog method is of second order and is known for its good conservation of energy properties. To increase the accuracy of the simulation, higher-order integration methods could be used like the fourth-order Yoshida's method which is based on similar principles as the leapfrog method.

7.3. Model extensions

In this research, a simple model for wide binary systems is used. The physical components which are modeled are the stars themselves, and the external field. There are, however, far more physical aspects present in a real wide binary system. The most notable of these aspects are listed below in order of importance:

- Galactic tidal effect
- Third star
- Mass transfer between stars

The most important model extension is the addition of the galactic tidal effect. The stars in a wide binary system are typically separated by thousands of AU. Due to this large separation, the gravitational field of the Milky Way is stronger for the star which is nearer to the galactic center. Thus the non-homogeneity of the external field is taken into account, whereas the external field was assumed to be homogeneous in this research. By taking the galactic tides into account, the eccentricity and inclination of the wide binary orbit may change. For further research, it may be interesting to study the effect that a smaller third star has on the orbit of the wide binary. The last model extension is the mass transfer between stars. This model extension is significant for binary systems where the separation is small because the stars are then able to exchange mass. For a wide binary system where separations are larger, this effect is negligible.

8

Conclusion

In this thesis, the modified Newtonian dynamics theory has been tested with the wide binary test, where the stability of the orbit of a wide binary system is tested. To do this, an iterative particle mesh N-body code was used to simulate wide binary systems. The masses, initial positions, and initial velocities for the wide binary systems were taken from a wide binary data set which is based on the data from the ESA Gaia mission. The modified Newtonian dynamics theory is non-linear, thus the external field of the Milky Way on a smaller subsystem like the wide binary is non-trivial. Therefore, the external field was also modeled. Furthermore, the code performance was increased by using the parallelization capabilities of the GPU.

Three models were simulated, each with the same three wide binary systems. The simulated models were the Newtonian model, the MOND model without an external field, and the MOND model with an external field. The three wide binary systems are characterized by the magnitude of the internal acceleration. The first system is in the Newtonian regime, which means that the internal accelerations are larger than the MOND transition acceleration $a_0 = 1.2 \cdot 10^{-10}$ m/s^2. The second system is the transition regime, which is when the internal accelerations are close to a_0. The third system is the deep MOND regime, which is when the internal accelerations are smaller than a_0.

The result of the Newton model was that only the Newtonian regime resulted in stable orbits. The MOND without external field model gave stable orbits for each regime. There were also a significant amount of flyby-like systems for the MOND without external field model but these were left out as these systems would be flyby-like for each model. The MOND with external field model resulted in similar results as the Newton model, for both only the Newtonian regime was stable. The orbits of the Newtonian model and the MOND with external field model look the same but are slightly different. This is because the external field is dominant over the internal accelerations, making the simulation Newtonian-like, but not fully Newtonian. At last, the total energy in the system is conserved. More measurements from the Gaia mission over a longer time may be able to confirm one of the three models. Currently, the conclusion that could be made is that the MOND model without EF can produce stable wide binary orbits where the other models could not. Another conclusion is that the Newton model and MOND with EF model produce similar results but give slightly different orbits. More long-term measurements are needed to make conclusions about which model is correct.

Further research recommendations include increasing the performance of the code, increasing the accuracy of the simulation, and extending the model. Increasing the performance of the code consists of translating the code from Python into a faster programming language, improving the implementation of the FFT, and implementing float16 compatibility for the FFT. Improving the accuracy of the code can be done by using adaptive time steps, improving the performance of the code, reducing self-interaction, using higher-order numerical methods, and using the particle mesh Ewald method. For further model extensions, the galactic tidal effect is the most important extension.

Bibliography

[1] Ryan S. Park et al. Precession of mercury's perihelion from ranging to the messenger spacecraft. *The Astronomical Journal*, 153, February 2017.

[2] Vera C. Rubin and Jr. Ford, W. Kent. Rotation of the Andromeda Nebula from a Spectroscopic Survey of Emission Regions. *The Astrophysical Journal*, 159:379, February 1970.

[3] Katherine Freese. Dark matter. `https://ned.ipac.caltech.edu/level5/Sept17/Freese/Freese2.html`, 2017. Accessed: 2024-06-14.

[4] M. Milgrom. A modification of the Newtonian dynamics as a possible alternative to the hidden mass hypothesis. *The Astrophysical Journal*, 270:365–370, July 1983.

[5] J. E. Felten. Milgrom's revision of Newton's laws - Dynamical and cosmological consequences. *The Astrophysical Journal*, 286:3–6, November 1984.

[6] K. G. Begeman, A. H. Broeils, and R. H. Sanders. Extended rotation curves of spiral galaxies : dark haloes and modified dynamics. *Monthly Notices of the Royal Astronomical Society*, 249:523, April 1991.

[7] Robert H. Sanders and Stacy S. McGaugh. Modified newtonian dynamics as an alternative to dark matter. *Annual Review of Astronomy and Astrophysics*, 40(1):263–317, September 2002.

[8] X Hernandez, M A Jiménez, and C Allen. Wide binaries as a critical test for gravity theories. *Journal of Physics: Conference Series*, 405:012018, December 2012.

[9] Indranil Banik and Hongsheng Zhao. Testing gravity with wide binary stars like α Centauri. *Monthly Notices of the Royal Astronomical Society*, 480(2):2660–2688, 07 2018.

[10] Charalambos Pittordis and Will Sutherland. Testing modified-gravity theories via wide binaries and GAIA. *Monthly Notices of the Royal Astronomical Society*, 480(2):1778–1795, October 2018.

[11] Charalambos Pittordis and Will Sutherland. Wide binaries from gaia edr3: preference for gr over mond? *The Open Journal of Astrophysics*, 6, February 2023.

[12] Kyu-Hyun Chae. Breakdown of the newton–einstein standard gravity at low acceleration in internal dynamics of wide binary stars. *The Astrophysical Journal*, 952(2):128, jul 2023.

[13] Indranil Banik, Charalambos Pittordis, Will Sutherland, Benoit Famaey, Rodrigo Ibata, Steffen Mieske, and Hongsheng Zhao. Strong constraints on the gravitational law from Gaia DR3 wide binaries. *Monthly Notices of the Royal Astronomical Society*, 527(3):4573–4615, 11 2023.

[14] X. Hernandez, Kyu-Hyun Chae, and A. Aguayo-Ortiz. A critical review of recent gaia wide binary gravity tests, 2024.

[15] R. B. Tully and J. R. Fisher. A new method of determining distances to galaxies. *Astronomy & Astrophysics*, 54:661–673, February 1977.

[16] S. Torres-Flores, B. Epinat, P. Amram, H. Plana, and C. Mendes de Oliveira. Ghasp: an hα kinematic survey of spiral and irregular galaxies - ix. the near-infrared, stellar and baryonic tully-fisher relations : Ghasp - ix. the tully-fisher relation. *Monthly Notices of the Royal Astronomical Society*, 416(3):1936–1948, August 2011.

[17] P.M. Visser, S.W.H. Eijt, and J.V. de Nijs. Fast particle-mesh code for milgromian dynamics. *Astronomy & Astrophysics*, 681, 2024.

[18] Cupy. `https://cupy.dev/`. Accessed: 04-06-2024.

[19] CMG Lee. Ieee 754r half floating point format. Wikimedia Commons, 2010. Accessed: 2024-06-06.

[20] European Space Agency (ESA). Gaia Overview. https://www.esa.int/Science_Exploration/Space_Science/Gaia/Gaia_overview, Accessed: June 11, 2024.

[21] Charalambos Pittordis and Will Sutherland. Testing modified gravity with wide binaries in Gaia DR2. *Monthly Notices of the Royal Astronomical Society*, 488(4):4740–4752, 07 2019.

[22] Arenou et al. Gaia Data Release 2. Catalogue validation. *Astronomy & Astrophysics*, 616:A17, August 2018.

[23] Gaia Collaboration. Gaia early data release 3 - summary of the contents and survey properties. *A & A*, 649:A1, 2021.

[24] D. Katz, P. Sartoretti, M. Cropper, et al. Gaia data release 2: Properties and validation of the radial velocities. *Astronomy & Astrophysics*, 622:A205, February 2019.

[25] V. Ballenegger, J.J. Cerdà, and C. Holm. Removal of spurious self-interactions in particle–mesh methods. *Computer Physics Communications*, 182(9):1919–1923, 2011. Computer Physics Communications Special Edition for Conference on Computational Physics Trondheim, Norway, June 23-26, 2010.

A

Appendix

The proof for the gradient theorem is given below.

Let ϕ be a scalar field, $\boldsymbol{k}$ a constant vector, V some volume, and S the surface of V. Then the following holds:

$$\boldsymbol{k} \cdot \left(\iiint_V \boldsymbol{\nabla} \phi \mathrm{d}^3 r \right) = \iiint_V \boldsymbol{\nabla} \cdot (\boldsymbol{k}\phi) \mathrm{d}^3 r \tag{A.1}$$

This holds because $\boldsymbol{k} \cdot \boldsymbol{\nabla}\phi = \boldsymbol{\nabla} \cdot (\boldsymbol{k}\phi) - \phi(\boldsymbol{\nabla} \cdot \boldsymbol{k})$, and $\boldsymbol{\nabla} \cdot \boldsymbol{k} = 0$ as $\boldsymbol{k}$ is a constant vector. Thus $\boldsymbol{k} \cdot \boldsymbol{\nabla}\phi = \boldsymbol{\nabla} \cdot (\boldsymbol{k}\phi)$. Continuing with the divergence theorem gives:

$$\iiint_V \boldsymbol{\nabla} \cdot (\boldsymbol{k}\phi) \mathrm{d}^3 r = \iint_S \phi \boldsymbol{k} \cdot \hat{\boldsymbol{n}}_s \mathrm{d}^2 r = \boldsymbol{k} \cdot \left(\iint_S \phi \hat{\boldsymbol{n}}_s \mathrm{d}^2 r \right) \tag{A.2}$$

where $\hat{\boldsymbol{n}}_s$ is the outpointing unit normal on the boundary $\mathrm{d}^2 r$. Since this is true for all constant vectors $\boldsymbol{k}$, the following holds:

$$\iiint_V \boldsymbol{\nabla} \phi \mathrm{d}^3 r = \iint_S \phi \hat{\boldsymbol{n}}_s \mathrm{d}^2 r \tag{A.3}$$