

CS 530: High-Performance Computing
Seminar 1: A Survey of Computational Physics

Nathan Chapman

Department of Computer Science
Central Washington University

April 29, 2024

Contents

1	Introduction	2
2	Classical Mechanics	3
2.1	N-Body Orbits	4
3	Electromagnetism	7
3.1	Fringing Electric Fields of Non-Ideal Capacitors	8
4	Quantum Mechanics	13
4.1	Quantum Chemistry - Caffeine in Stomach Acid	13
5	Relativity	18
5.1	Effects of Eccentricity on the Precession of Mercury's Perihelion	19
5.2	Leading Order Gravitational Wave Frequencies from Short-Term Post-Newtonian Orbits via Fourth-Order Symplectic Integration	28
6	Conclusion	33

Chapter 1

Introduction

Each of the five pillars of physics (Classical Mechanics, Electromagnetism, Relativity, Quantum Mechanics, and Thermodynamics) has well-posed problems for which the mathematical models are either unreasonable or impossible to solve by hand. As such, computational methods are needed to get an approximate solution.

There are many numerical methods to support the evolution of physical models for each domain. Whether it be using a fourth-order Runge-Kutta method to solve the first-order ordinary differential equation that models a ballistic object under the influence of gravity and air-resistance, or using an eighth-order symplectic Yoshida integrator to model a system of bodies orbiting each other while accounting for the produced gravitational waves, there are methods for every physical context. Though, because these methods are using approximations to their analytic counterparts in calculus, there is error that needs to be considered.

Numerical analysis is the practice of tracking and rigorously accounting for the discrepancies that arise when approximating analytic mathematical methods by their finite and counterparts. These errors can arise not only from approximating descriptions from calculus, but also from operations in linear algebra as they're applied to finding the eigenvalues of quantum operators.

Chapter 2

Classical Mechanics

The “everyday” world as we know it is described by a model of physics that has been studied for millenia. As such, we call this model of the behavior of nature “Classical Mechanics”. Now I hear you ask “Why?”. Well “mechanics” is the study of motion, and we include the “classical” preface to distinguish it from “modern” physics which encompasses all the physics discovered after around 1905 or 1925 (depending on who you ask); those physics will be covered later in sections 4 and 5. For now we take a look at the mathematical model that describes the motion of not-too-small, not-too-large objects moving at slow velocities.

The motion of “classical” objects, ranging in scale from biological cells, sand, ants, birds, cars, airplanes, planets, up to stars, can be described as the solution to what’s called the *equation of motion*. The equation of motion for an object with mass m under the influence of forces F_i is determined by Newton’s 2nd law of motion

$$\sum_i \mathbf{F}_i = m\ddot{\mathbf{x}}, \quad (2.1)$$

where $\ddot{\mathbf{x}}$ represents the second time derivative of the position \mathbf{x} known as *acceleration*. These forces completely determine the how objects move through space in time.

Mathematically, the resulting equation is a vector second-order ordinary differential equation, compactly representing the underlying system of differential equations. In many cases, this differential equation is homogenous to represent no time-dependent external forces supplying the physical system with energy. Likewise, sometimes these forces can be represented by nonlinear terms; making the equation of motion much more complex.

The following is a survey of topics in classical mechanics whose forces yield an equation of motion either too complex or too unreasonable to solve by hand.

2.1 N-Body Orbits

While so-called N-Body problems are prolific in physics and other sciences, a quintessential example is the gravitational interaction of astrophysical bodies i.e. massive bodies in space.

2.1.1 Mathematical Model: Systems of Ordinary Differential Equations

In the case of multi-body gravitational interactions, the governing equation of motion is determined via Newton's law of gravitation between two objects with masses m_i and m_j

$$F_{ij} = \frac{Gm_i m_j}{r_{ij}^2} \hat{r} \quad (2.2)$$

where $G = 6.67 \times 10^{-11} \frac{Nm^2}{kg^2}$ is the universal gravitational constant, r_{ij} is the distance between the two objects, and \hat{r} is the unit vector pointing from object i toward j . Not only are there an equation of motion for each object, but each equation is coupled to every other equation. This results in the three-dimensional vector differential equation [25]

$$\sum_{j;i \neq j} \frac{Gm_i m_j}{((x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2)^{3/2}} \frac{(x_j - x_i)\hat{x} + (y_j - y_i)\hat{y} + (z_j - z_i)\hat{z}}{\sqrt{((x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2)}} = m_i(\ddot{x}_i\hat{x} + \ddot{y}_i\hat{y} + \ddot{z}_i\hat{z}) \quad (2.3)$$

for object i , and a similar equation for every other object. For example, a system of 3 objects interacting only via gravitational forces has the equations of motion

$$\frac{Gm_2}{((x_1 - x_2)^2 + (y_1 - y_2)^2 + (z_1 - z_2)^2)^{3/2}} \begin{bmatrix} x_2 - x_1 \\ y_2 - y_1 \\ z_2 - z_1 \end{bmatrix} + \frac{Gm_3}{((x_1 - x_3)^2 + (y_1 - y_3)^2 + (z_1 - z_3)^2)^{3/2}} \begin{bmatrix} x_3 - x_1 \\ y_3 - y_1 \\ z_3 - z_1 \end{bmatrix} = \begin{bmatrix} \ddot{x}_1 \\ \ddot{y}_1 \\ \ddot{z}_1 \end{bmatrix} \quad (2.4)$$

$$\frac{Gm_1}{((x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2)^{3/2}} \begin{bmatrix} x_1 - x_2 \\ y_1 - y_2 \\ z_1 - z_2 \end{bmatrix} + \frac{Gm_3}{((x_2 - x_3)^2 + (y_2 - y_3)^2 + (z_2 - z_3)^2)^{3/2}} \begin{bmatrix} x_3 - x_2 \\ y_3 - y_2 \\ z_3 - z_2 \end{bmatrix} = \begin{bmatrix} \ddot{x}_2 \\ \ddot{y}_2 \\ \ddot{z}_2 \end{bmatrix} \quad (2.5)$$

$$\frac{Gm_1}{((x_3 - x_1)^2 + (y_3 - y_1)^2 + (z_3 - z_1)^2)^{3/2}} \begin{bmatrix} x_1 - x_3 \\ y_1 - y_3 \\ z_1 - z_3 \end{bmatrix} + \frac{Gm_2}{((x_3 - x_2)^2 + (y_3 - y_2)^2 + (z_3 - z_2)^2)^{3/2}} \begin{bmatrix} x_2 - x_3 \\ y_2 - y_3 \\ z_2 - z_3 \end{bmatrix} = \begin{bmatrix} \ddot{x}_3 \\ \ddot{y}_3 \\ \ddot{z}_3 \end{bmatrix} \quad (2.6)$$

Because this problem is in three dimensions, there are actually 9 differential equations to solve in order to fully model the evolution of the position and velocity over time of each object.

2.1.2 Computational Model

Finite Differences

Because computers can only store numbers of finite precision, approximations must be used in order to implement these differential equations in a form that can be understood by the computer. One such translation is by approximating the analytic derivatives and differential operators with *finite differences*. In its most basic form, the derivative can be approximated as a ratio of differences in the form

$$\frac{df}{dt} \approx \frac{f(t + \Delta t) - f(t)}{\Delta t}. \quad (2.7)$$

This equation is known as the first-order-accurate, forward-finite-difference representation of the first-order derivative. There are similar representations for higher-order derivatives, more accurate approximations, and more accurate approximations of higher-order derivatives. Similarly, there are also *central* and *backward* finite-differences.

Each of these finite-difference schemes can be represented via coefficients. For example, the derivative of order m with accuracy n has $2p + 1 = 2\lfloor \frac{m+1}{2} \rfloor - 1 + n$ central coefficients $a_{-p}, a_{-p+1}, \dots, a_{p-1}, a_p$ defined such that

$$\begin{bmatrix} 1 & 1 & \dots & 1 & 1 \\ -p & -p+1 & \dots & p-1 & p \\ (-p)^2 & (-p+1)^2 & \dots & (p-1)^2 & p^2 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ (-p)^{2p} & (-p+1)^{2p} & \dots & (p-1)^{2p} & p^{2p} \end{bmatrix} \begin{bmatrix} a_{-p} \\ a_{-p+1} \\ \vdots \\ \vdots \\ a_p \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ m! \\ \vdots \\ 0 \end{bmatrix} \quad (2.8)$$

where the only non-zero value on the right side of the equation is at the $m + 1$ -th index.

Once the derivative operators are discretized using finite-differences, there are different algorithms in which the differences are used to solve the differential equations.

Symplectic Integrators

While there are many methods of numerically solving differential equations (known as *integrators*), one of the most important classes when considering a context such as orbital mechanics and oscillatory situations (i.e. Hamiltonian systems) is that of *symplectic* integrators. This type of integrator is so important to computational physics because it conserves energy throughout time. Without energy conservation, the system becomes unphysical, which can lead to unnatural behavior such as a mass on a spring extending to infinity. Two of the simplest symplectic integrators are the *Semi-implicit Euler Method*

$$v_{n+1} = v_n + a_n \Delta t \quad (2.9a)$$

$$x_{n+1} = x_n + v_{n+1} \Delta t \quad (2.9b)$$

and the *Velocity Verlet Algorithm*

$$v_{n+1} = v_n + \frac{a_n + a_{n+1}}{2} \Delta t \quad (2.10a)$$

$$x_{n+1} = x_n + v_n \Delta t + \frac{1}{2} a_n \Delta t^2 \quad (2.10b)$$

Parallel Computing

Because there are so many terms and equations, and each term is only needs the current step, this problem naturally lends itself to being parallelized both over the objects[23] and in time[20].

2.1.3 Numerical Analysis

Discretization

Error associated with the size of the time-step is dictated by which specific finite-difference method is chosen.

Energy Drift

In the case of orbital mechanics, there is no loss of mechanical energy (ignoring the effects of general relativity; more on this later in section 5). Therefore, a constraint is imposed on the system that the total mechanical energy (kinetic energy + potential energy) of the system must remain constant at each point in time. When using non-symplectic methods, this constraint is not adhered to which leads to *energy drift*.

Chapter 3

Electromagnetism

In a general setting, modelling how the electric and magnetic fields change in time and space requires the solutions to *Maxwell's Equations*[13], succinctly represented as

$$\begin{aligned}\nabla \cdot \mathbf{E} &= \frac{\rho}{\epsilon_0} & \nabla \cdot \mathbf{B} &= 0 \\ \nabla \times \mathbf{E} &= -\frac{\partial \mathbf{B}}{\partial t} & \nabla \times \mathbf{B} &= \mu_0 \mathbf{J} + \mu_0 \epsilon_0 \frac{\partial \mathbf{E}}{\partial t}.\end{aligned}\tag{3.1}$$

Expanded, these equations take the form

$$\begin{aligned}\partial_x E_x(t, \mathbf{r}) + \partial_y E_y(t, \mathbf{r}) + \partial_z E_z(t, \mathbf{r}) &= \frac{\rho}{\epsilon_0} & \partial_x B_x(t, \mathbf{r}) + \partial_y B_y(t, \mathbf{r}) + \partial_z B_z(t, \mathbf{r}) &= 0 \\ \begin{bmatrix} \partial_y E_z - \partial_z E_y \\ \partial_z E_x - \partial_x E_z \\ \partial_x E_y - \partial_y E_x \end{bmatrix} &= -\partial_t \begin{bmatrix} B_x \\ B_y \\ B_z \end{bmatrix} & \begin{bmatrix} \partial_y B_z - \partial_z B_y \\ \partial_z B_x - \partial_x B_z \\ \partial_x B_y - \partial_y B_x \end{bmatrix} &= \mu_0 \rho \begin{bmatrix} v_x \\ v_y \\ v_z \end{bmatrix} + \mu_0 \epsilon_0 \partial_t \begin{bmatrix} E_x \\ E_y \\ E_z \end{bmatrix}.\end{aligned}\tag{3.2}$$

In their expanded form, Maxwell's equations form a coupled system of eight *partial* differential equations (i.e. differential equations whose solutions depend on more than one variable). Because each of these equations must be solved simultaneously, modelling the dynamics of the electromagnetic field throughout space is quite challenging. In general solving these equations as presented would require high-performance computing, but there are simpler cases where such care is not needed.

3.1 Fringing Electric Fields of Non-Ideal Capacitors

The following is taken from previous work[5] completed in my undergraduate course in computational physics.

3.1.1 Mathematical Model

The behavior of the electric field strength in a parallel plate capacitor can be well understood simply using the classical electromagnetism equation $E = \Delta V/d$ where V is the potential difference between the two parallel plates of the capacitor and d is the distance between them. However, this assumes the plates are infinitely large and “close” together. A consequence of this assumption is a constant electric field between the plates and far from the edge in the finite case. We will be exploring the *fringing field*, or the field that exists outside the area between the capacitors and close to the edges.

While the fringing field is extremely difficult to model analytically, we can make use of the fact that the electric field is proportional to the gradient of the electric potential. Since the electric potential is smooth, we can consider the electric potential at any point being the average of the electric potential of its surrounding points. This is a simplified explanation, but as it turns out we can use this idea to model the potential field numerically.

3.1.2 Computational Model

We model the electric potential of a parallel plate capacitor in a grounded box with the *relaxation method* and via the *Gauss-Seidel* (GS) implementation

$$V_{new}[i, j] = \frac{1}{4}(V_{old}[i + 1, j] + V_{new}[i - 1, j] + V_{old}[i, j + 1] + V_{new}). \quad (3.3)$$

To strive for maximum computational efficiency, we augment the GS method with a *Simultaneous Over-Relaxation* (SOR) as well. Our SOR method is implemented by augmenting the new potential as

$$V_{new}[i, j] = \alpha\Delta V + V_{old}[i, j], \quad (3.4)$$

where α is the relaxation coefficient defined by $\alpha = 2/(1 + \pi/L)$, and ΔV is the difference between the new value of the potential and the old one. With the inclusion of SOR, our computation time decreases by a factor of L when compared to not including SOR. Though, since this implementation is done numerically, several parameters of the system must be stipulated beforehand.

First we will model our parallel plate capacitors to have a potential difference of $2V$ (our left capacitor will be $1V$ and our right will be $-1V$), and we will place it in a grounded box of length L . The size of the box, or correspondingly the size of the capacitor, are natural length scales for the problem and one can be defined in terms of the other.

As a preliminary step in simulating the system numerically, we must discretize our region into cells of side length Δx . To make sure our simulation is at least somewhat accurate, Δx should be significantly smaller than the total side length of the grounded/bounding box. This will allow us to assign values of the electric potential to actual areas in the box. This reassignment, from initial “best guess”, takes the form of the GS method, shown in Equation 3.3.

Now that we have a method to determine the potential of a point based on surrounding points, we will need to apply it to our model of the physical system. We will need each ‘cell’ to have an initial value to use our method, and so we will make an ‘initial guess’ for all cells that are not part of the boundary conditions. In our case we used an initial guess of 0 for every point except the capacitors.

To find a reasonable value for the potential at each point we will need to define convergence criteria to compare against in each iteration of the averaging loop. This is necessary because after the first round of averaging only the cells bordering the capacitors will have any change at all. The more averages we do, the more cells begin being affected by this relaxation method. Once the outermost cells are being affected, the cells closest to the capacitors

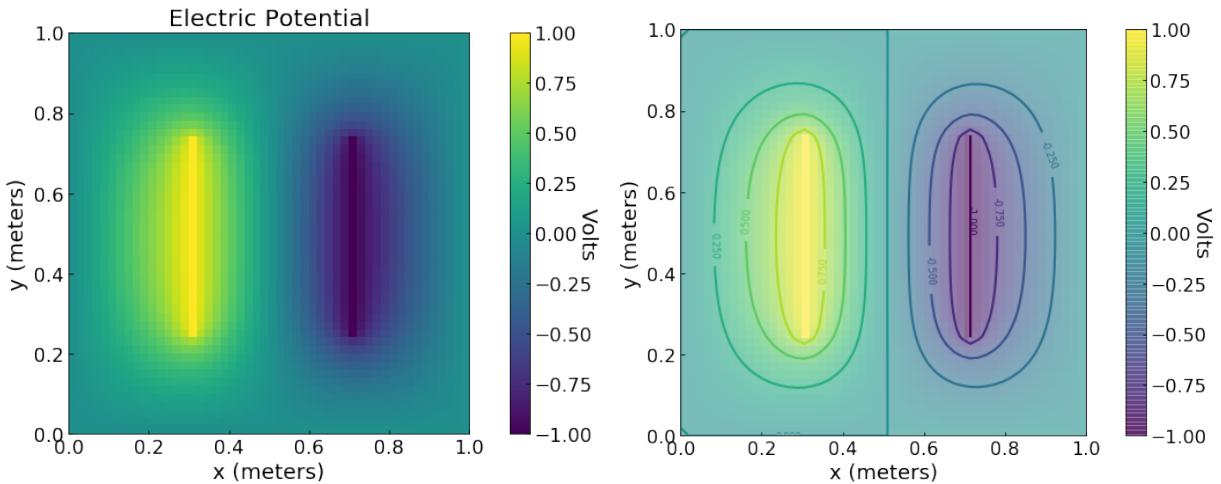


Figure 3.1: The electric potential caused by capacitor, whose plates are held at a potential difference of $V = 2$, with a boundary condition of $V = 0$ at the edges of the box. The left plot shows only the color mapping of the potential while the bottom superimposes equipotential lines separated by $0.25V$ each on top of the color mapping. The initial guess used for the unknown areas was $0V$ and the convergence criteria was 10^{-5} . The box/system is $1\text{m} \times 1\text{m}$, while the cell size is $\frac{1}{50^2} \text{ m}^2$. The separation of the capacitors is 0.4 m and it's length is 0.5 m .

have gone through many iterations of the average, and so they are more close to the true value than the outer cells. Therefore we defined convergence criteria, forcing our code to continue running through the averages and recalculating the potential until the ΔV , or the difference in the potential value before and after going through the most recent round of averaging, is less than our convergence criteria, which we set to 10^{-5} . As the cells closer to the capacitors will reach a point of small ΔV before outer cells will, we have defined our program such that it will continue running until our *maximum* ΔV is less than our convergence criteria.

Now that we have a complete distribution of the electric potential in our system, we can take the negative gradient of the potential to find the electric field at each point. Though, since we're working in a discrete space, we need to rethink about how we take derivatives.

To calculate the electric field we will consider a change in potential over a distance, which can be done with the equation

$$\begin{aligned} E_x &= -\frac{1}{2} \left(\frac{V[i+1,j] - V[i-1,j]}{\Delta x} \right) \\ E_y &= -\frac{1}{2} \left(\frac{V[i,j+1] - V[i,j-1]}{\Delta x} \right). \end{aligned} \quad (3.5)$$

These components can be used to not only show the vector field, but also to give the *magnitude* of the electric field at each point. The magnitude of the electric field at each point is then the norm of the electric field vectors at the respective points.

Using the computational methods described in Sec. ??, we created a plot of the potential field within our range. In Figure 3.1 we are able to observe the potential across our space. As expected our potential is highest at our left capacitor, with a voltage of $+1 \text{ V}$, and lowest at our right capacitor, with a voltage of -1 V . Our boundary conditions at the edge also appear to be near/at zero which is to be expected.

By looking at the second plot in Figure 3.1 we can see that the electric field in between the two capacitors should be fairly constant, which is to be expected as that is the point of capacitors.

Using Equations 3.5, we were able to calculate an electric field for each cell and plot them in a vector plot, shown in

Figure 3.2. The electric field looks much like we would expect it to. As we can see in the bottom plot, the electric field vectors are perpendicular to the equipotential lines, which attests to the physical accuracy of our model. The electric field behaves like we would expect it would, with the fringing field being weaker than the field between the capacitors. The field between the capacitors is relatively constant, which is what we would hope for. We see that there is the highest magnitude electric field right around the edges of the capacitors, which makes sense as that is where the potential will be changing the fastest from the potential of the capacitor to the ground value of the edges.

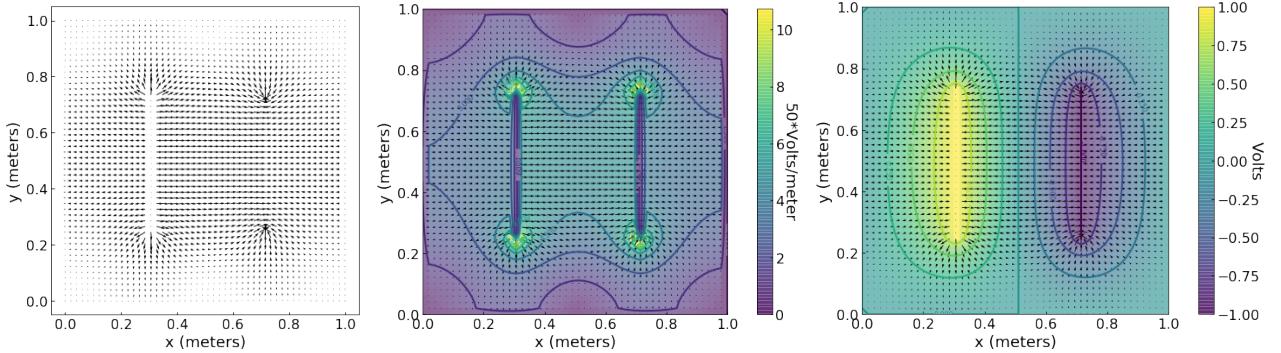


Figure 3.2: (Left) Electric field in the area of interest. (Center) The top plot shows the electric field vectors on their own, which the second superimposes a color map on top that represents the magnitude of the electric field in $\frac{V}{50\text{m}}$, or 50V/m. The third plot superimposes our potential field on top of the electric field lines. Variables used are the same as in Figure 3.1.

3.1.3 Numerical Analysis

As a final test of the physical accuracy of our model, we chose a spot directly centered between the two capacitors and observed how the electric field at this point changed as the separation of capacitors changed. We did this because for an ideal capacitor we can calculate the electric field simply by using Equation ??.

By using both the numerical and analytical solutions shown in Figure 3.3 we can conclude that our model seems to be more accurate for smaller separations and increase in error as the plates get further apart.

As we can see in Figure 3.3, the error increases as the capacitor separation increases, which makes sense if we look back to how we need more iterations of our Gauss-Streisel SOR method to reach cells that are further away from our boundary conditions. This means that the cell in the center that we are observing has probably only recently met the convergence criteria and so is more prone to error than those that continued to be averaged long past meeting the convergence criteria.

This also makes sense physically as Equation ?? assumes a perfect capacitor, which true capacitors will not have perfectly constant electric fields.

As we can explain the error physically and it is small, we can use it as a model for our system.

Finally we observed how cell size affected our results for the electric field.

The trend shows that as cell size is decreased it trends towards lower percent error in the electric field magnitude and thus towards a numerical convergence. We are not quite sure what the bump near 0.03 cell size means and could not identify a physical or numerical cause. As the rest of the points seems to follow a trend, we chose to make the assumption that the point was inconsistent and do not address it further.

Having confirmed the physical validity of our model, we proceeded to investigate how the fringing field varied with respect to plate separation and distance from capacitor.

As we can see from the plots of the electric field in Figure 3.2, the electric field is strongest at the corners of the capacitor and in between the capacitors and gets weaker as we move further away, into the fringing fields.

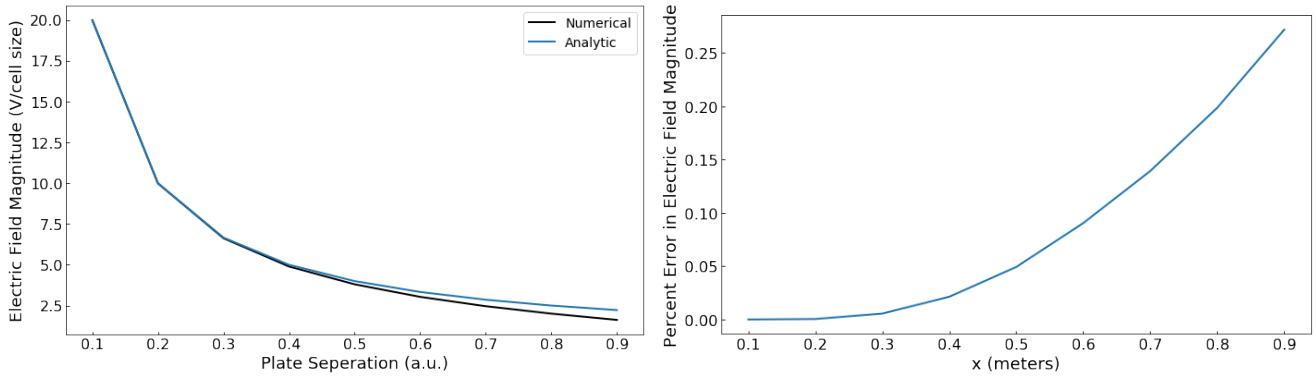


Figure 3.3: The top plot shows the analytic and numerical solution for the electric field magnitude directly between the parallel plate capacitors while the bottom one shows the percent error as a function of plate separation. Other than the varied plate separation, all variables used are the same as used in Figure 3.1.

As for the dependence on distance between the capacitors, we were able to plot the electric field for a chosen point outside the capacitors and plot it as a function of changing separation, shown in Figure 3.4.

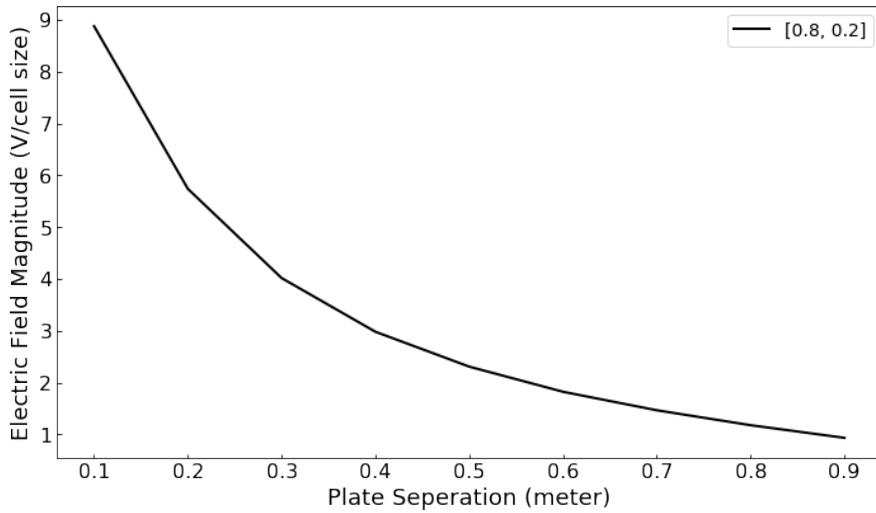


Figure 3.4: The numerically calculated electric field magnitude is plotted above as a function of the separation between the capacitors.

In this figure, we can observe that the electric field magnitude at a spot that is 0.5 meters across the cell horizontally and 0.2 meters down from the top ground boundary condition decreases as the plate separation increases.

Our model for the fringing electric field around a capacitor has created a system that seems physically reliable, as shown through its similarity to the numerical solution in Figure 3.3 and when evaluating cell size. Our model has allowed us to observe the fringing field for varying separations and distances from the capacitor.

Our model is limited in the resolution to which we can model the system, i.e. how small the cell size can be, due to computational time. We used a cell size of 1/50 m for all of our plots except when we were varying cell size and this achieved a resolution that was relatively smooth and resulted in enough data points to evaluate its accuracy, as shown in Figure 3.3. If we made the cell size smaller, we would get smoother plots of the electric field magnitude and could model the relation to changing variables a bit more accurately, but as we had to calculate so many points for each plot the computation time took significantly longer than the added resolution was worth.

Finally, we saw a weird 'bump' in our plot of the percent error in Figure 3.5 which interrupted what seemed to be a fairly smooth trend. We were unable to identify the cause of the bump, but as it was small and seemed to still

trend towards lower values overall, we are comfortable saying that the percent error in the electric field magnitude decreases with decreasing cell size.

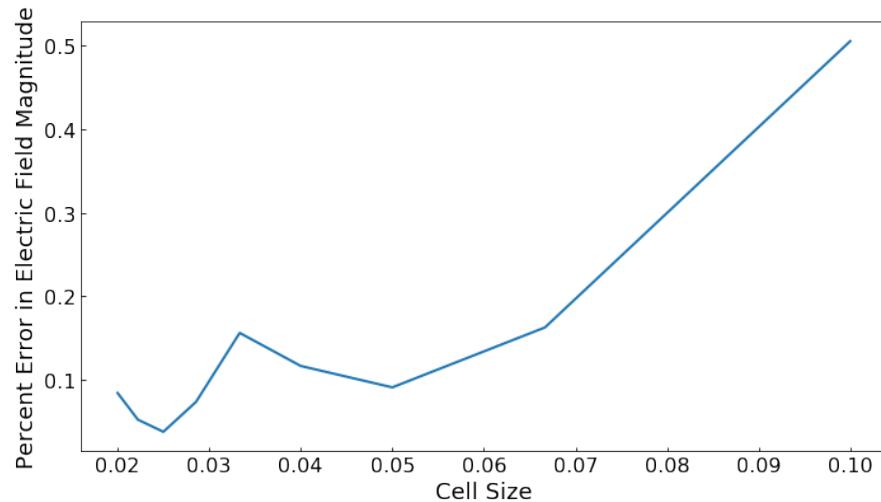


Figure 3.5: The percent error in the Electric Field when using the cell size of 0.01 m as a baseline. Other than varying cell size, all variables used are the same as in Figure 3.1.

Chapter 4

Quantum Mechanics

4.1 Quantum Chemistry - Caffeine in Stomach Acid

The following is work I completed for the final project in my graduate course “Chemistry 581: Computational Quantum Chemistry” [6].

4.1.1 Mathematical Model

At the heart of quantum mechanics lies the (time-independent) *Schrödinger Equation*

$$H\psi = E\psi \tag{4.1}$$

where H is a Hermitian operator (i.e. matrix in a basis which is its own conjugate-transpose) known as the *Hamiltonian* and E is the (scalar) energy. Much like how Newton’s 2nd Law determines the motion of classical matter, Schrödinger’s equation describes the motion of quantum matter. Thus, the Schrödinger equation is the equation of motion in quantum mechanics.

The hamiltonian for a collection of molecules and/or atoms can be represented as [11]

$$H = -\underbrace{\sum_i \frac{\hbar}{2m_e} \nabla_i^2}_{\text{e motion}} - \underbrace{\sum_k \frac{\hbar}{2m_e} \nabla_k^2}_{\text{Z motion}} - \underbrace{\sum_{i,k} \frac{e^2 Z_k}{r_{ik}}}_{\text{e-Z interaction}} + \underbrace{\sum_{i < k} \frac{e^2}{r_{ik}}}_{\text{e-e interaction}} + \underbrace{\sum_{k < l} \frac{e^2 Z_k Z_l}{r_{kl}}}_{\text{Z-Z interaction}} \tag{4.2}$$

where \hbar, m_e, e are physical constants, Z is an atomic number, r is the distance between two objects, and i, j run over electrons, k, l run over nuclei. Additionally, recall that ∇^2 is the Laplacian differential operator defined as $\nabla_i^2 = \partial_{x_i}^2 + \partial_{y_i}^2 + \partial_{z_i}^2$, and $r_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2}$.

While there are “mathematically motivated” methods (e.g. Runge-Kutta) of numerically solving a second-order (linear!) partial differential equation in $3N$ dimensions ($3N$ because each of the N particles has kinetic and interaction energies in 3 dimensions), because of the physical context of the Schrödinger equation, another method can be used: Hartree-Fock Self-Consistent Field Theory (SCF).

The *variational principle* of quantum mechanics *effectively* states that for a “trial” wave function ψ' , the associated energy as given from the Schrödinger equation E satisfies $E \geq E_0$, where E_0 is the energy of the “true” solution ψ_0 . Additionally, the trial wave function ψ can be represented as a linear combination $\psi = \sum_i c_i \phi_i$ of orthonormal (i.e. vectors that are orthogonal and have unit magnitude) functions ϕ_i from a *basis set*¹. Physically, these basis

¹3-21G is a simple choice due to its balance between accuracy and speed [11].

functions represent single-electron molecular orbitals. The combination of the variational principle and the ability to represent the trial wave function as a linear combination with unknown weights of known functions, transforms the problem of finding the solution to a partial differential equation in $3N$ dimensions into a *minimization problem*.

An example to highlight computational quantum chemistry, because there are many interacting electrons, is in considering the optimal configuration of a caffeine molecule in hydrochloric acid.

4.1.2 Computational Model

The Hartree-Fock self-consistent field method (outlined in figure 4.2) (HF SCF) is a way of iteratively solving the many-electron Schrödinger equation $H|\psi\rangle = E|\psi\rangle$ for the molecular wave function $|\psi\rangle$. This is done by replacing the scalar operator $\hat{U}_{ee} = \sum_{i < k} \frac{e^2}{r_{ik}}$ representing the electron-electron repulsion energy by its expectation value $\langle \hat{U}_{ee} \rangle = \langle \psi_0 | \hat{U}_{ee} | \psi_0 \rangle$, where $|\psi_0\rangle$ is an initial guess e.g. a Gaussian. The solution $|\psi_1\rangle$ of that Schrödinger equation is then used to repeat the process, yielding a (assumingly) more accurate solution $|\psi_2\rangle$. This nesting process continues until some convergence threshold is met, such as the energy of system doesn't change by more than some amount.

Considering a single molecule of caffeine (as shown in figure 4.1) in the presence of a single HCl molecule while using water as a solvent, the Schrödinger equation can be solved to optimize the configuration of each of these molecules using the computational quantum chemistry software *Gaussian*. Using Density Functional Theory (a highly accurate method in which to implement the SCF), the system can be optimized to the configuration shown in figure 4.3.

Furthermore, the highest and lowest occupied molecular orbitals can be seen in figure 4.4.

Finally, the vibrational modes corresponding to the two tallest peaks in the IR spectrum (figure 4.5) of the optimal configuration are shown in the animated (depending on the pdf viewer used) figure 4.6.

4.1.3 Numerical Analysis

Many molecules and other chemical systems are well documented on PubChem, including their DFT-optimized configuration. Comparisons between the unoptimized and optimized systems are shown in figures 4.7 and 4.8.

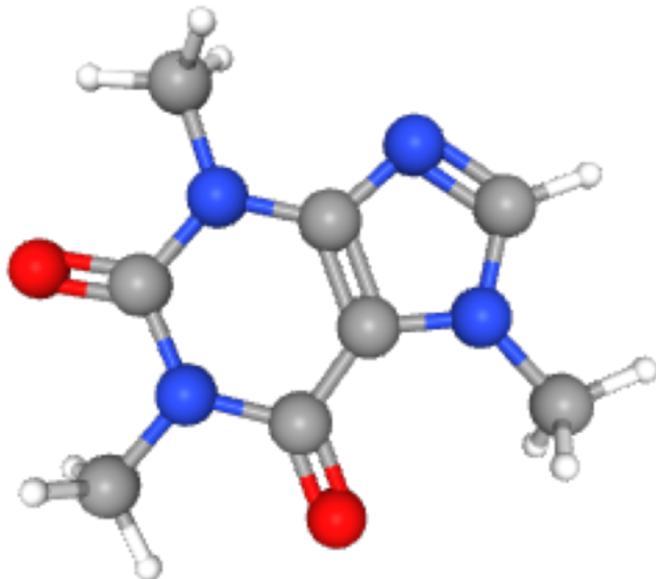


Figure 4.1: A caffeine molecule in its optimal configuration

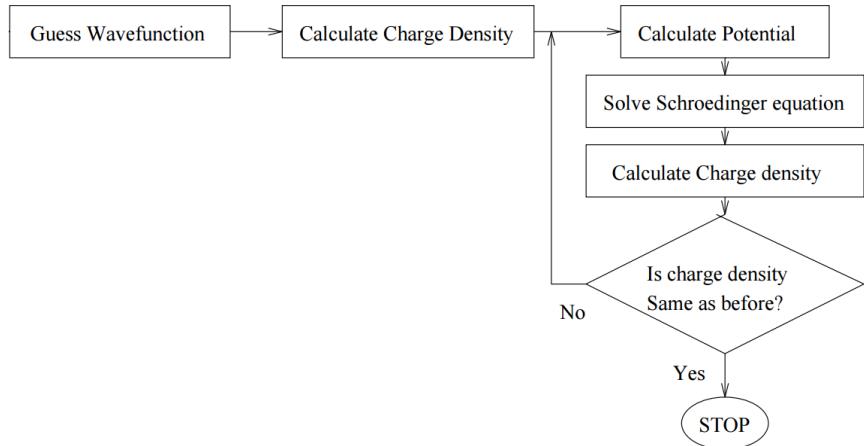


Figure 4.2: Diagram[14] showing the iterative nature of the Hartree-Fock Self Consistent Field Method.

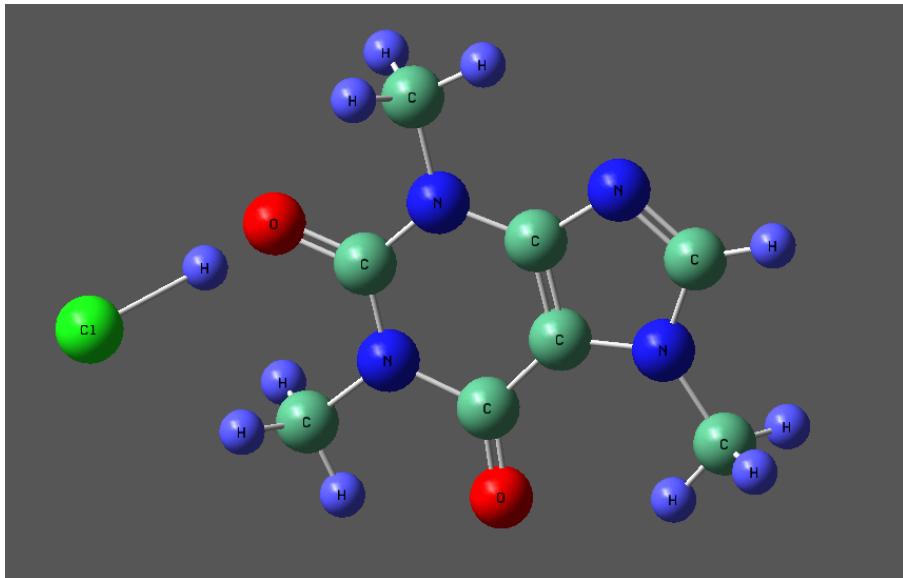
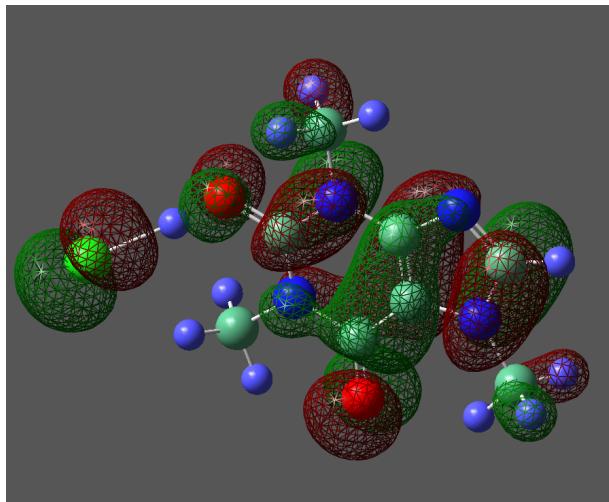
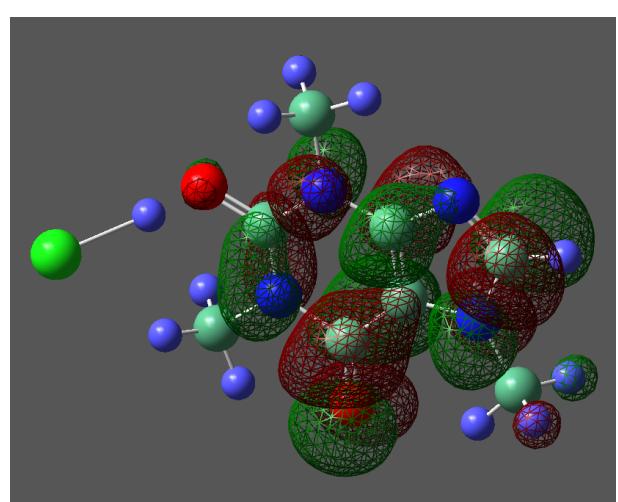


Figure 4.3: DFT optimized caffeine and HCl in water



(a) Highest Occupied Molecular Orbital



(b) Lowest Occupied Molecular Orbital

Figure 4.4: Frontier molecular orbitals

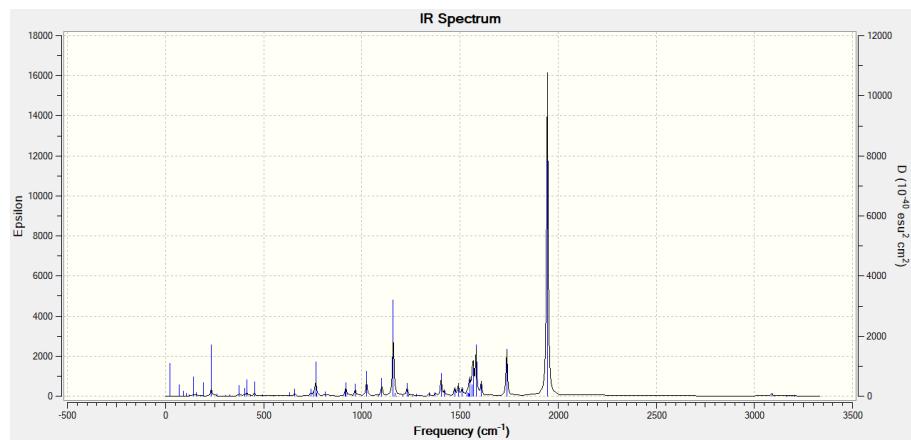


Figure 4.5: IR spectrum

(a) Vibrational mode corresponding to the second tallest peak in the IR spectrum above

(b) Vibrational mode corresponding to the tallest peak in the IR spectrum above

Figure 4.6: Vibrational modes of the two largest peaks in the IR spectrum

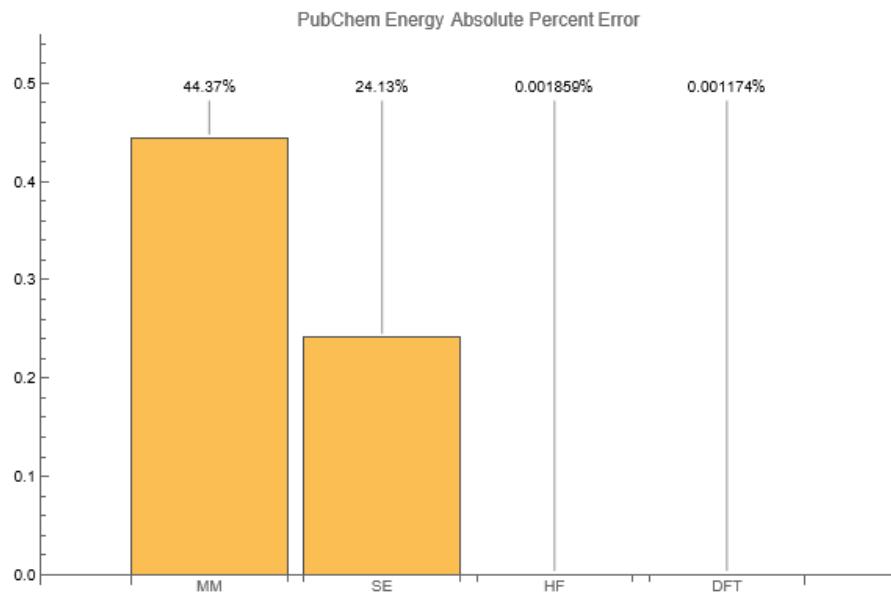


Figure 4.7: Absolute percent error in energy for different methods for the caffeine molecule as given by PubChem

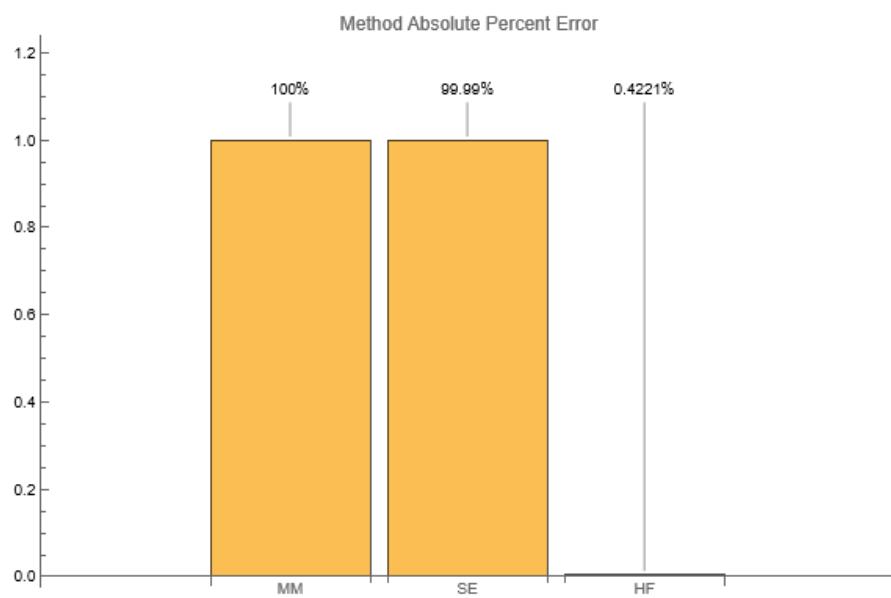


Figure 4.8: Absolute percent error in energy for different methods compared to DFT

Chapter 5

Relativity

One of the most fundamental concepts in analytic geometry is the coordinate system. Coordinate systems are composed of scale, direction, and origin¹. Given two coordinate systems, a position in space can be represented in either system, but there also exists a transformation between the two coordinate systems so that if you know the position in one coordinate system and the transformation between the systems, you can calculate the position in the other coordinate system. This remains true even if the origin of the coordinate system is moving. The study of how physical properties are represented and change between these coordinate systems is called *relativity*.

While there is Galilean relativity, which deals with objects whose speed is unbounded, more accurate models of physics have been introduced[12] with a “universal speed limit”, bounding the speed at which any object can travel. This model is called *Special Relativity*, and this relatively small change (no pun intended) has drastic consequences when considering objects moving at near this speed limit. These relativistic effects are highlighted when there are many objects to compare²[18]. While it is possible to describe the physics of objects moving near the universal speed limit using simple mathematical tools, in order to have a unified theory that describes both these effects and those that arise when considering how mass plays a role, more complex mathematical tools need to be considered. As such, more complex Computational Model are needed to get a definitive picture of physics in space and time.

A different approach to describing Special Relativity is to consider physics not just happening *in* space and *over* time, but rather on a single backdrop that unifies these concepts called *spacetime*. When considering relativistic effects on this spacetime backdrop, there is not only a coupling between motion relative to different observers, but also between mass and the spacetime itself. Special Relativity ignores this mass-spacetime coupling, resulting in a *special case* of *General Relativity*. In the presence of an object, spacetime bends and warps in a way proportional to the object’s mass and energy. Mathematically, this warping is represented as the curvature of a manifold as described by Differential Geometry. In the end, this complex description leads to a system of of coupled, nonlinear, partial differential equations that are most often impossible to solve analytically, so numerical methods must be employed.

The system of ten coupled, nonlinear, second-order, partial differential equations in four dimensions[1] as mentioned above is called the *Einstein Field Equations* (EFEs). The EFEs are notoriously complicated and yield very few closed-form solutions[21], each only being defined for situations with a high-degree of symmetry. Numerical methods, such as the ADM formalism[3], have been employed to find approximate solutions to the EFEs defining the field of *Numerical Relativity*. The physical scenarios considered, such as binary black hole mergers[24] and ray tracing near a black hole[15], yield forms of the EFEs that must be numerically solved with high-performance computing[19, 2, 10].

¹More technically, these terms scale → algebraic field, direction → a set of basis vectors

²[OpenRelativity](#) visualization

The effects of General Relativity (GR) are *mostly* unobservable within our solar system. One of the exceptions being the precession of Mercury's perihelion. Due to Mercury's close proximity to the Sun, and its higher-than-average eccentricity, Mercury's orbit around the sun changes in ways unpredicted by Newtonian gravity. The reason for this discrepancy is that while Mercury orbits the sun, gravitational waves are released and carry energy away from the pair, causing Mercury's motion to change. This phenomenon has been explored in two aspects: modelling how the eccentricity of Mercury's orbit affects the precession of its perihelion^[8], and analysizing the “chirp” that arises from the leading-order frequencies of the emitted gravitaitonal waves when the Sun and Mercury merge^[9].

5.1 Effects of Eccentricity on the Precession of Mercury's Perihelion

The following is work I completed for the final project in my undergraduate course “Physics 486: Computational Physics”^[8].

5.1.1 Mathematical Model

Newtonian gravity is simple enough that it is a center piece of all introductory physics courses. This placement and difficulty is primarily due to the fact that the only mathematical theory encapsulated in this formalism is basic vectors in a Euclidean space, or rather arrows in our 3D world. Of course the form of gravity a la Newton is

$$\mathbf{g} = -\frac{Gm}{r^3} \mathbf{r} = -\frac{Gm}{r^2} \hat{\mathbf{r}}, \quad (5.1)$$

where of course, G_N is the Newtonian gravitational constant, m is the mass of the *central mass*, or rather the object that is causing the gravity, and r is the magnitude of the separation \mathbf{r} between the central mass and a point in space. This form works extremely well for everyday situations like objects interacting with the Earth, or *most* planets interacting with the Sun. It is in the latter case, namely Mercury interacting with the Sun, where we find Newtonian gravity is not suitable to describe objects that are interacting via a stronger gravity. For such situations we must extend our model to the standard treatment of gravity, General Relativity (GR).

General Relativity is, to put it *absurdly* lightly, a step up in difficulty from Newtonian gravity. Albert Einstein's famous and elegant theory needs much more difficult mathematical machinery to describe the gravitational interactions between objects. Without going into too much detail, GR requires the use of a field of math called *Differential Geometry* (DG) which describes geometric properties, like distance, in/on curved spaces/surfaces. The full general relativistic description of gravity gives the Einstein Field Equations (EFE)^[22]

$$G_{\mu\nu} = \frac{8\pi G_N}{c^4} T_{\mu\nu} \quad (5.2)$$

where G is the Einstein tensor that describes how space and time curve, and T is the stress-energy tensor that describes how much energy there is in a region. There are few analytic solutions to this equation, and therefore few instances of exact descriptions of “true” gravity. Therefore, we must usually resort to approximate numerical solutions in order to model the systems that require the full form of GR. It turns out, even approximating solutions to the EFE are really hard. Though, there are situations that lend themselves to being modeled via a mix of Newtonian gravity and GR.

GR is definitely needed and to be fully used in the case of masses strongly interacting via gravitation, e.g. near black holes and neutron stars, but what about when the objects aren't massive enough, or are far enough away that they experience GR effects only slightly? This is called the *weak field limit* (WFL). The set of systems that obey Newtonian gravity fall into this regime, but also the ones that behave slightly non-Newtonian-ly. In the WFL, such systems can be modeled with a combination of Newtonian gravity and GR via corrections to the Newtonian form of gravity (Eq. 5.1). This combination of theories takes the form of adding corrections to the Newtonian form of gravity as

$$\mathbf{g} = -\frac{c^2}{2} \frac{r_s}{r^2} \left(1 + 3 \frac{r_L^2}{r^2} \right) \hat{r}, \quad (5.3)$$

where r_s is the Schwarzschild radius of a mass defined by

$$r_s = \frac{2G_N}{c^2} m, \quad (5.4)$$

where m is the mass of the object, and

$$r_L^2 = \frac{(\mathbf{r} \times \dot{\mathbf{r}})^2}{c^2}. \quad (5.5)$$

Even though the WFL can be modeled via modified Newtonian dynamics (MOND), Newtonian gravity is still easier to implement and understand. So the natural question arises, at what point are the GR effects significant enough to invoke, at least, MOND? In figure (5.1), we can see that there is a small but significant difference between Newtonian gravity and the MOND gravity as the distance between masses approaches zero. At a certain point though, the difference is indeed negligible, and Newtonian gravity can be used without worry.

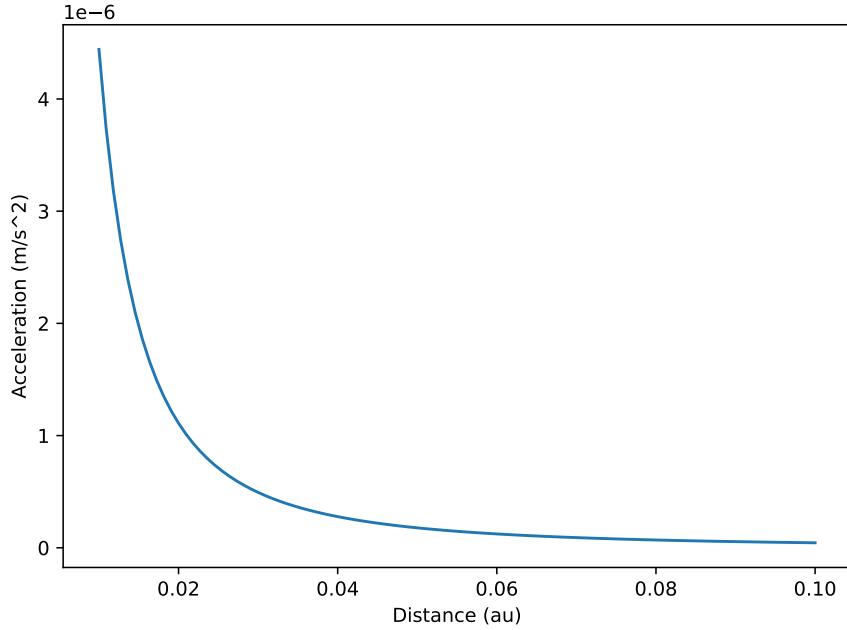


Figure 5.1: The absolute difference in gravity between the Newtonian model and the GR model. The mass used was one solar mass, and the velocity was Mercury's average orbital velocity.

With this form of gravity, any investigation of a system in the WFL can be carried out as it would with Newtonian gravity. Though in this work, the precession of periapsides is done computationally, there is an analytic form of the angular precession due to GR.

GR directly tells us that the radial equation for an orbit is defined as[22]

$$r(\phi) = \frac{(1 - e^2)a}{1 + e \cos[(1 - \delta\phi_0/2\pi)\phi]} \quad (5.6)$$

From equation (5.6), the angular precession of the orbit is exactly

$$\delta\phi_0 = \frac{6\pi M}{a(1-e^2)} \quad (5.7)$$

where M is mass “causing” the gravity, and a, e are the semi-major-axis and eccentricity respectively.

5.1.2 Computational Model

Initialization & first step: To start the simulation of these two celestial bodies, one of which is fixed in space at the origin, the program takes as its initialization parameters the masses of the two orbiting bodies, the initial position of the orbiting body at its periapsis, the initial velocity of the orbiting body at its periapsis, the number of orbits the body will undergo, and whether or not the gravity used will be the modified gravity (Eq. 5.3) or the standard Newtonian gravity (Eq. 5.1). From these parameters, the gravity due to the fixed mass is found at the initial position of the orbiting mass, and the first step in the orbiting mass’s orbit is found through an Euler-Cromer method as

$$\mathbf{v}_1 = \mathbf{v}_0 + \mathbf{g}_0 \Delta t \quad (5.8a)$$

$$\mathbf{r}_1 = \mathbf{r}_0 + \mathbf{v}_1 \Delta t \quad (5.8b)$$

where \mathbf{g}_0 is the gravity at the initial position \mathbf{r}_0 acting on the orbiting mass with velocity \mathbf{v}_0 , and Δt is the time step. We make note of the velocity dependence of gravity here because in Newtonian gravity, the strength of the gravitational acceleration only depends on the distance from the mass, whereas in GR, the strength of gravity is dependant upon the distance and velocity of the mass on which gravity is acting.

Time step: Here we define our time step as above as

$$\Delta t = \frac{2v_0}{\alpha a_0}, \quad (5.9)$$

where α is the factor by which we decrease the time step, and in this work $\alpha = 2000$ (While this seems drastic in the sense that this produces on average a time step of about 15 minutes/sub-hour precision, if any smaller value was chosen, the end results *seemed* completely unphysical). We choose this time step so the Euler-Cromer evolution of the position is indeed a valid one. Since the time step is defined as such, it is actually determined by the initial properties of the system, and not in fact a parameter that is given by the user. After the first step is taken, the model transitions to evolving under a Verlet method.

Evolution: Whereas some models use forward biased methods, like the Euler method [17], here we use the time-reversible Verlet method

$$y_{i+1} = 2y_i - y_{i-1} + \left. \frac{d^2y}{dt^2} \right|_{y_i, t_i} \Delta t^2 \quad (5.10)$$

to ensure energy conservation. Conservation of energy is of the utmost importance in long-term gravitational simulations because over such long time periods, the error/energy-leak accumulates and makes the end result completely inaccurate. Now we evolve our system using this time step and integration method.

The main evolution of the system is defined by

$$\mathbf{r}_{i+1} = 2\mathbf{r}_i - \mathbf{r}_{i-1} + \mathbf{g}(\mathbf{r}_i, \mathbf{v}_i, t_i)\Delta t^2 \quad (5.11a)$$

$$\mathbf{v}_{i+1} = \frac{\mathbf{r}_{i+1} - \mathbf{r}_{i-1}}{2\Delta t} \quad (5.11b)$$

where the position \mathbf{r} evolves under our Verlet integration, and the velocity \mathbf{v} is determined via a simple difference quotient. Through this updating algorithm, the system evolves throughout its orbit for one orbital period.

Orbit: The system steps through time under (Eq. 5.11) for one orbital period. This orbital period τ is found through the classical Keplerian formula[25]

$$\tau^2 = \frac{4\pi^2\mu}{G_N m_1 m_2} a^3 \quad (5.12)$$

where G_N is the Newtonian gravitational constant, m_1, m_2 are the masses of the objects, $\mu = \frac{m_1 m_2}{m_1 + m_2}$ is the reduced mass of the system, and a is the semi-major-axis (SMA). Here the SMA is defined in terms of other parameters of the system as

$$a = \frac{c}{1 - \epsilon^2} \quad (5.13)$$

where c is the *principal orbital parameter* defined by

$$c = \frac{l^2}{G_N m_1 m_2 \mu}, \quad (5.14)$$

where l is the magnitude of the angular momentum defined by

$$l = m_2 \|\mathbf{r}_0 \times \mathbf{v}_0\| \quad (5.15)$$

and ϵ is the eccentricity of the orbit defined by

$$\epsilon = \frac{c}{r_0} - 1 \quad (5.16)$$

After one orbit has been simulated, the periapsis of that orbit is found by simply selecting the position that is closest to the fixed body. Once the location of the periapsis is known, and since we're interested in how the periapsis precesses, the angle of the periapsis is found with respect to the initial position (i.e. the x-axis because we started there) in arcseconds. We choose to work in units of arcseconds because we know that the perihelion precession of Mercury due to GR is about 43 arcseconds per century and so the average precession angle should be on that scale.

Now that we have an algorithm to evolve the system for one orbital period, we let the system evolve for as many periods as we want, and measure the positions, velocities, periapsides, the angles of the periapsides, and the eccentricities as we go.

After several orbits have been simulated, we perform a linear regression (with numpy's polyfit function) on the angles of the periapsides throughout the orbits to find the average rate of precession of the orbiting body. (All that work for one number!)

5.1.3 Numerical Analysis

Of course no computational model is complete without an analysis of the numerical accuracy. While there is no *easy* analytic solution to this problem, we can test the sub-processes that govern the overall results.

As above, several parameters are not calculated through recursive formulations but rather through their closed form mathematical definitions. Even though these properties have a closed form definition, computational calculations still carry error.

	$r_{S,\odot}$	g_{\oplus}	$\epsilon_{\text{Mercury}}$	τ_{Mercury}
Predicted	2.953 km	9.807 m/s ²	0.205	88 Earth days
Calculated	2.949 km	9.813 m/s ²	0.206	88.1 Earth days
Difference	0.004 km	0.006 m/s ²	0.001	0.1 Earth days
% Difference	0.16	0.06	0.49	0.11

Table 5.1: The calculated and measured values of the Schwarzschild radius of the Sun, the gravity at the surface of the Earth, the eccentricity of Mercury, and the orbital period of Mercury, along with their absolute and percent differences (i.e. percent difference = $|\text{predicted} - \text{calculated}|/\text{predicted}$).

In table (5.1) we compare the values calculated here against the either measured or more accurately computed values. We can see all properties are within 1% of their “accurate” counter-parts. While we can quite easily quantify the error in these parameters, we can also at least qualitatively describe the error in the evolution of the system as well.

To test the orbital procedure, we test the trajectory of Earth around the Sun because we know it should be mostly circular. In figure (5.2), we see how the trajectory of the Earth behaves for different time steps. The different trajectories in the figure are not actually the time steps exactly, but rather the scaling parameter α as in equation (5.9). In each of these simulations, Earth started at its average orbital distance (1 AU) on the x-axis, with its average orbital velocity orthogonal to the position ($v = 2\pi(1\text{AU})/(1\text{year})$). We can see that even for relatively small scaling factors ($\alpha = 20$ compared to the factor used for the results was $\alpha = 2000$), the trajectory of Earth converges to its known circular orbit. Another property we can test to get at least a qualitative idea of the error in our procedure is how the rate of precession varies with the length of the simulation.

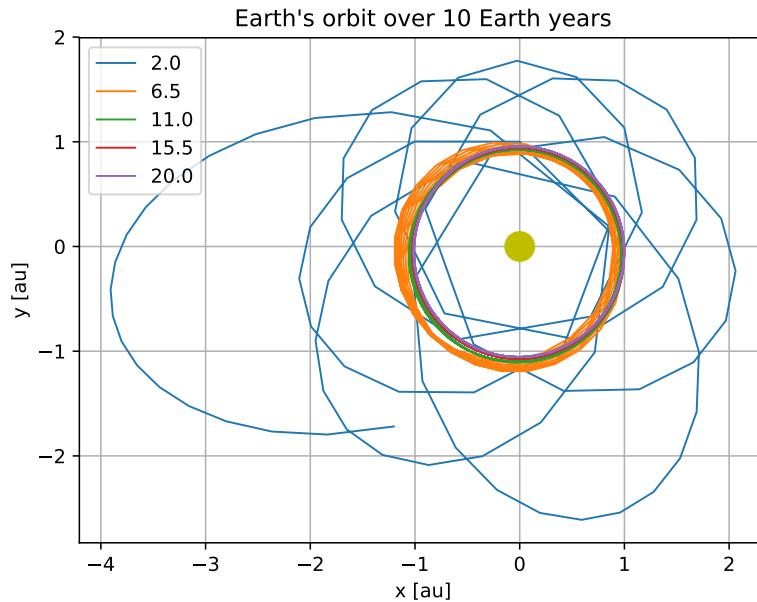


Figure 5.2: The trajectory of Earth over 10 Earth years for several different scaling parameters α , and therefore different time steps.

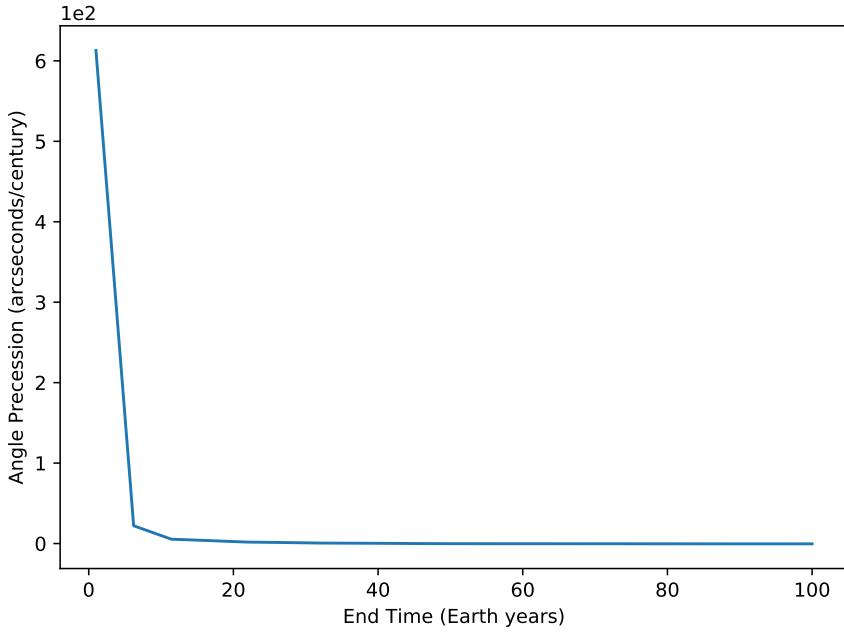


Figure 5.3: The precession rate as a function of the length of the simulation. The parameters used in this simulation were that of the Sun-Mercury system.

In figure (5.3) we can see that the rate of precession is on the scale of hundreds of arcseconds per century for a relatively short simulation time of around one Earth year. The precession rate quickly decreases and seems to converge as the length of the simulation is increased past 20 Earth years. We stop this test at a length of 100 Earth years because, in this simulation, the parameters used were that of the Sun-Mercury system, and the known value of Mercury's perihelion shift is about 43 arcseconds per century.

Now that we're (hopefully) at least somewhat confident the algorithm produces physically accurate results, let's analyze the Sun-Mercury system.

5.1.4 Results

Starting off, all the following results were done for the Sun-Mercury system, and with Mercury starting at its perihelion (about 0.3 AU) on the x-axis with its velocity at that point, a time-step scaling parameter of $\alpha = 2000$, and a simulation time of 100 Earth years. We start at perihelion because then we know that the velocity is orthogonal to the position, and it's a good starting point nonetheless. Before we actually look at some results, let's see the actual strength of the GR effects at Mercury. In figure (5.4), we can see the effects of GR are *quite* minuscule, on the order of nanometers, compared to the closer regimes as shown in figure (5.1).

Since the effects of GR are so small at Mercury, the orbit should be almost negligibly close to the Newtonian orbit. That is, we shouldn't be able to see any difference over any reasonable time scale. In figure (5.5), that's exactly what we see.

Though if we take a closer look, specifically at the angle of the perihelion at each orbit, we can see GR's almost hidden effects. In figure (5.6), we can see that the perihelion actually oscillates in its angle, while still having a trend to decrease linearly. It's here that the importance of the (ridiculously) small time step (of scale factor $\alpha = 2000$) is needed. If the time step were any bigger the angular precession would not be on the right scale as the known 43 arcseconds per century.

That being said, I also don't know why the angle jumps up and oscillates with such high amplitude. The main takeaway here though is that the overall behavior of the perihelion is that it does linearly shift with time. In this

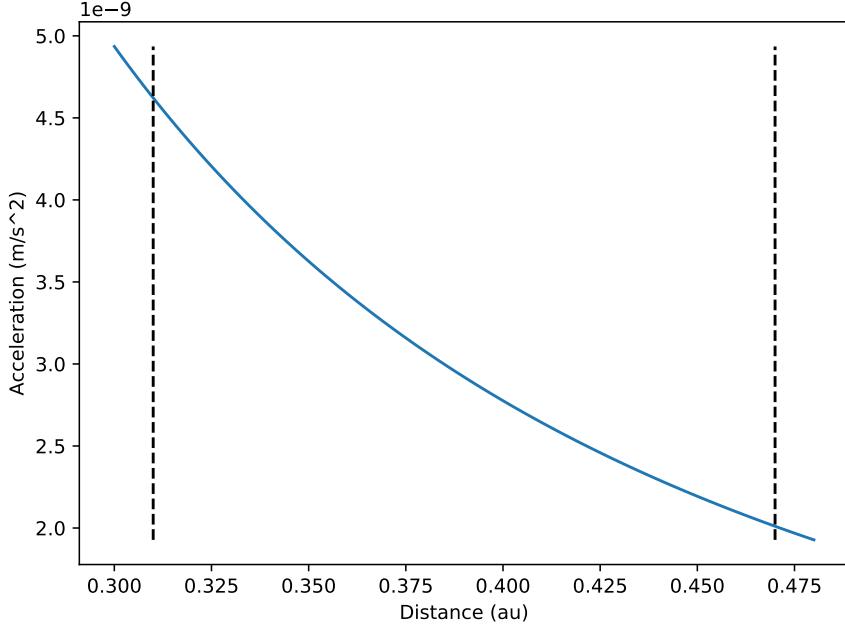


Figure 5.4: The strength of the gravitational acceleration from the Sun due to GR in the regime of Mercury's orbital distance. The vertical dashed, black lines correspond to the distance between Mercury and the Sun at Mercury's perihelion and its aphelion.

case the overall rate of precession is around the known value (in magnitude) of 43 arcseconds per century, but, getting back to the main idea, what how would this change with different eccentricity?

In figures (5.7) and (5.8), we see how the rate of precession changes as the eccentricity of the orbit changes. Well, the overall behavior is there. As we increase eccentricity, the rate of precession decreases.

In this analysis, only eccentricities $\epsilon < 0.8$ were considered because the implementation evolved the simulation in time, and since an increase in eccentricity corresponds to a non-linear increase in orbital period, the computation times associated with eccentricities $\epsilon > 0.8$ were unreasonably large.

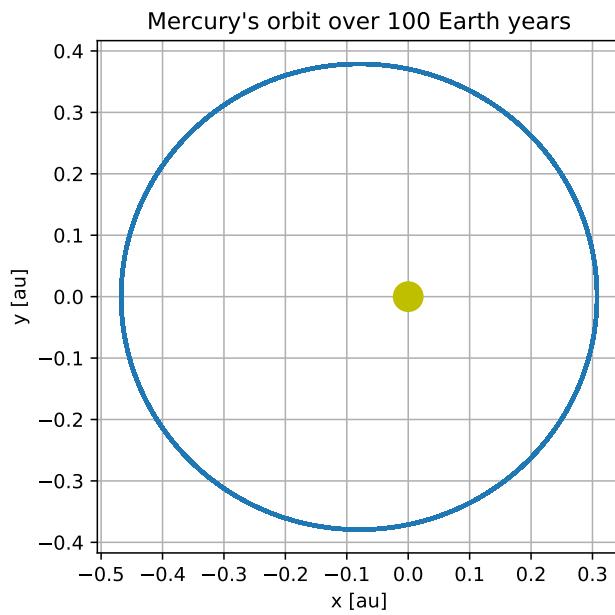


Figure 5.5: The trajectory of Mercury under the influence of MOND over the course of 100 Earth years.

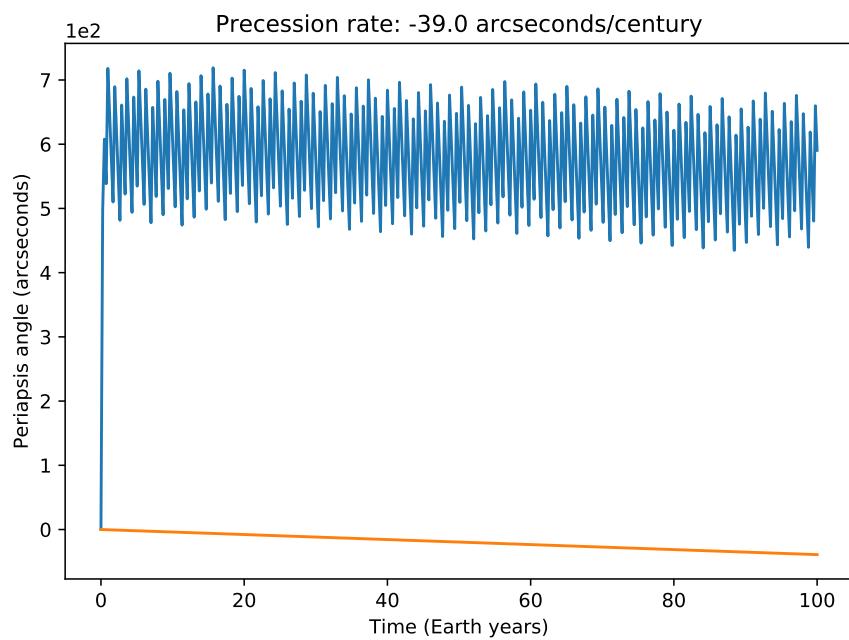


Figure 5.6: The angular precession of the perihelion of Mercury over the course of 100 Earth years.

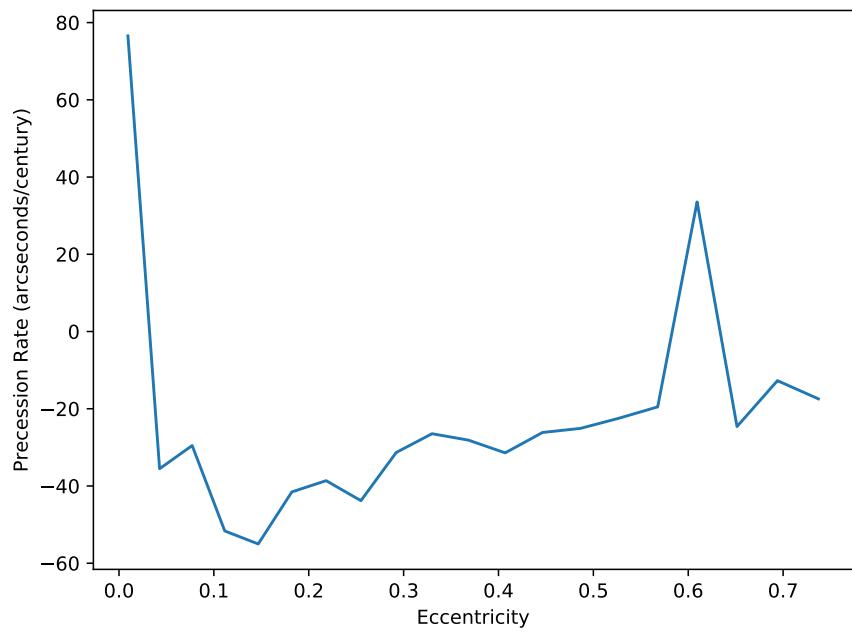


Figure 5.7: The rate precession of Mercury if it had different eccentricities.

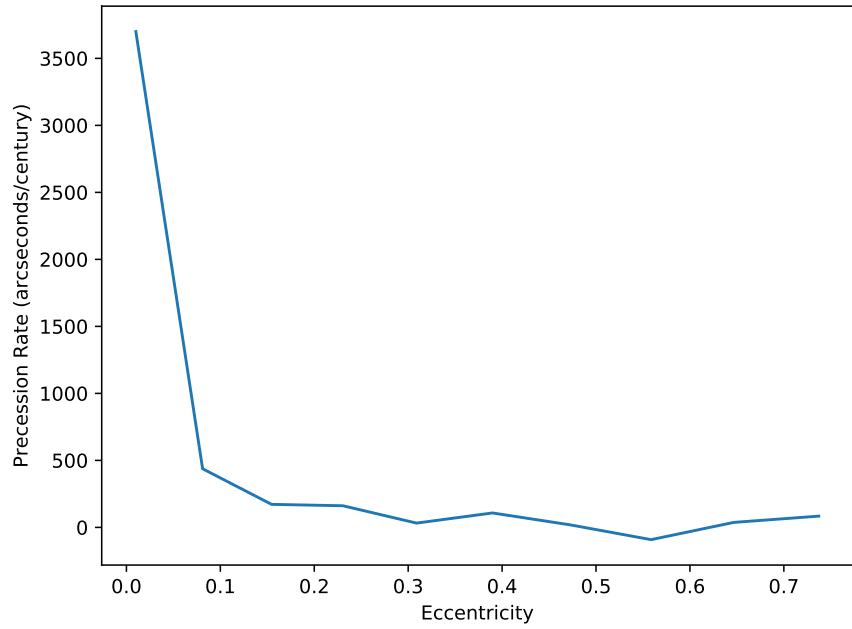


Figure 5.8: The rate precession of Mercury if it had different eccentricities. Each simulation here was done with a simulation time of 20 Earth years.

5.2 Leading Order Gravitational Wave Frequencies from Short-Term Post-Newtonian Orbits via Fourth-Order Symplectic Integration

The following is work I completed for the final project in my graduate course “Physics 561: Advanced Computational Physics”[9].

5.2.1 Mathematical Model

At its core, General Relativity (GR) is composed of 16 highly-coupled, nonlinear partial differential equations known as the Einstein Field Equations. These equations are only able to be analytically solved without approximation for a few particularly symmetric physical systems³. The remaining systems require either to be approximated or solved numerically. One type of system that actually falls in both categories is those where the gravitational effects are sufficiently weak. These are called “weak-field” systems (WFS). GR also predicts that all systems with a changing mass distribution (e.g. orbits) emit gravitational waves (GW), WFS by naturally emit low-amplitude gravitational waves.

While WFS can be approximated in several ways, we will further focus our attention on slightly changing Newtonian physics to account for only the strongest relativistic effects. This is called a Post-Newtonian approximation (PNA). PNA is also able to pick out the strongest part of the emitted low-amplitude gravitational waves. The strongest part being the leading order frequency of the gravitational wave.

Even though weak-field systems are able to be solved analytically after being first-order approximated, the analytic methods are still quite complex and near unapproachable. In this case, it would be easier (and for more complex systems, necessary) to solve and simulate the system with numerical methods. When simulated, finding the properties of the system is just a matter of directly measuring them.

With all of these in hand, the rest of this paper focuses on numerically simulating the dynamics of particles (such as Mercury) in WFS under a PNA in order to measure the leading order frequency of the emitted gravitational waves.

Inspiral & Gravitational Waves

The most common way GW are produced is when a planet or star orbits around its parent star. As these waves are emitted, they carry energy away from the system. Then, due to conservation of momentum, the radius of the orbit shrinks and the speed of the object increases. This increase in speed and proximity yields the emission of higher amplitude GW. These higher amplitude GW carry even more energy away from the system, cause the orbit to shrink and speed up. It is obvious these effects yield a positive feedback loop (Fig 5.9) that will only cease when the object collapses into its parent. In the context of binary neutron star and black hole mergers, this moment is known as coalescence as they physically merge into a single body.

Post-Newtonian Approximations

In this investigation, the PNA is implemented by incorporating a “relativistic” factor into Newtonian gravity (Eq. 5.17) (in terms of the Schwarzschild radius of the parent body)[16].

Newtonian	Relativistic Correction
$\ddot{\vec{r}} = -\frac{c^2}{2} \frac{r_S}{r^2} \frac{\vec{r}}{r}$	$\ddot{\vec{r}} = -\frac{c^2}{2} \frac{r_S}{r^2} \underbrace{\left(1 + 3 \frac{r_L^2}{r^2}\right)}_{\text{Relativity}} \frac{\vec{r}}{r}$

³See the famous Schwarzschild solution.

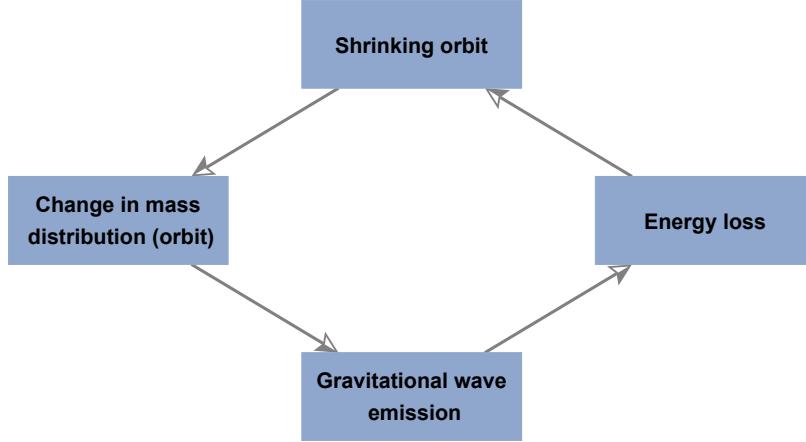


Figure 5.9: The emission of gravitational waves induces a feedback loop that causes more and more, stronger and stronger gravitational waves.

Since PNA is only valid for WFS, the system cannot be modeled up to coalescence as the GW emitted at and relatively prior are well within the strong-field regime. PNA does yield valid modelling until the time scale on which the orbital radius shrinks is comparable to the orbital period. This point is called the adiabatic limit[4]. Therefore, the investigations presented here will focus on the relatively early stages of the collapse. It is also the case that in WFS under a PNA, the leading order gravitational wave frequency is double the orbital frequency[4].

5.2.2 Computational Model

For many-iteration gravitational simulations such as these, it is best to use an algorithm that has little error in energy conservation. There is the class of numerical integrator called “symplectic” integrators that do just this. There are a few different, easily-implementable, symplectic methods that could and have been used in the past for simulations such as those in this investigation. The first-order symplectic Euler-Cromer[16] method and the second-order symplectic Verlet[7] methods have been used in the past to relative success but for much smaller time scales.

Since the time scale on which the dynamics of a solar system significantly change is on the order of millions of years, it is also prudent to use an algorithm that is still substantially accurate for larger time steps. The natural choice, given the previously used methods and time scales, is to increase the order of accuracy of the integrator. Such methods seem to be novel, but both analytic and numeric work has been done to find them[27].

$4\mathcal{O}$ symplectic Yoshida integrator[27]

$$x_i^0 = x_i \quad x_i^n = x_i^{n-1} + c_n v_i^{n-1} \Delta t \quad x_{i+1} = x_i^4 \quad (5.18)$$

$$v_i^0 = v_i \quad v_i^n = v_i^{n-1} + d_n \underbrace{a(x_i^n, v_i^{n-1})}_{\text{“leapfrog”}} \Delta t \quad v_{i+1} = v_i^4 \quad (5.19)$$

The algorithm in Eq. (5.18) has been modified from its original presentation to have the acceleration include a dependence on the velocity. The original algorithm only had acceleration dependent on position.

5.2.3 Numerical Analysis

Previous work[26] has been done with similar integrators such as Wisdom-Holman in very similar systems. The results of those investigations (Fig 5.10) showed that integrators such as the fourth-order symplectic Yoshida, the error of the algorithm was and stayed sufficiently small for simulations within the time scale considered here.

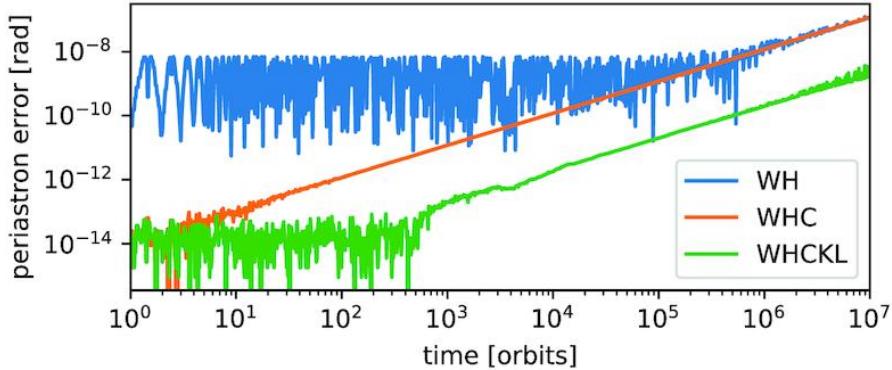


Figure 5. Periastron error in a long-term simulation with one planet and general relativistic corrections using the WH, WHC, and WHCKL integrators.

Mon Not R Astron Soc, Volume 490, Issue 4, December 2019, Pages 5122–5133, <https://doi.org/10.1093/mnras/stz2942>
The content of this slide may be subject to copyright: please see the slide notes for details.



Figure 5.10: Significantly low error for symplectic integrators in a weakly relativistic system.

5.2.4 Results

One of the most famous results for weak-field effects is the precession of Mercury's perihelion. Over the course of 10^5 Earth-years, this model shows (Fig: 5.11) significant precession. The angle of the perihelion with respect to the positive x-axis show is known to have a precession rate of about $43''/\text{century}$. So in order to validate the results we have from this investigation, we calculate the precession of measured periapsides. Our results are inline with the observed value. For each orbit, if the orbital period is measured, then the leading order of the emitted GW is twice that of the orbital period[16]. Even for 100,000 years, there is insignificant change in the frequency of the emitted GW.

5.2.5 Discussion

These results show that a WFS under PNA evolved with a fourth-order symplectic integrator yields simulated properties that are inline with experimental observation. These results also show the time scale on which the dynamics of the solar system significantly change is even much longer than 100,000 years.

5.2.6 Conclusion

Overall, analytic descriptions of general relativistic systems is near impossible for any moderately real system. Because of this, numerical is often preferred if not needed. Such simulations are even more implementable if they are approached by simply augmenting Newtonian physics for weak-field systems. These approaches are yield relatively accurate results, both in error and in agreement with observation, if implemented with a high-order symplectic integrator.

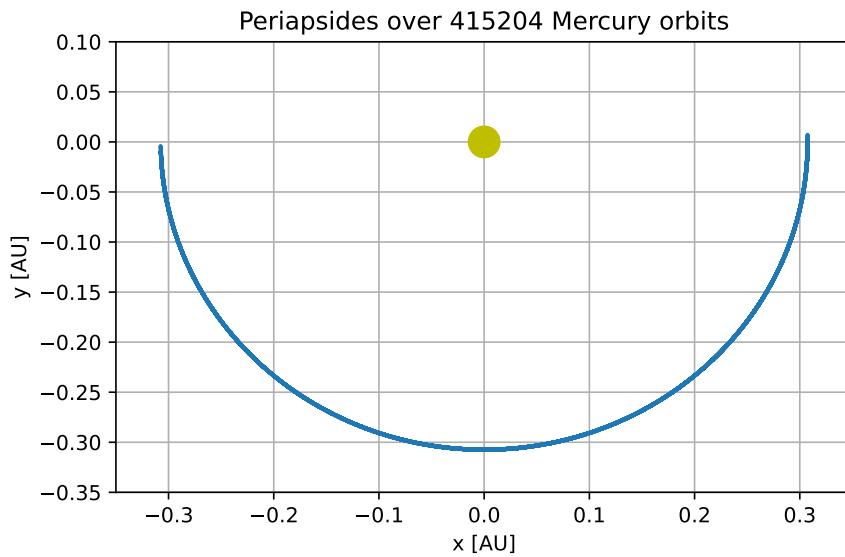


Figure 5.11: Precession of Mercury's perihelion

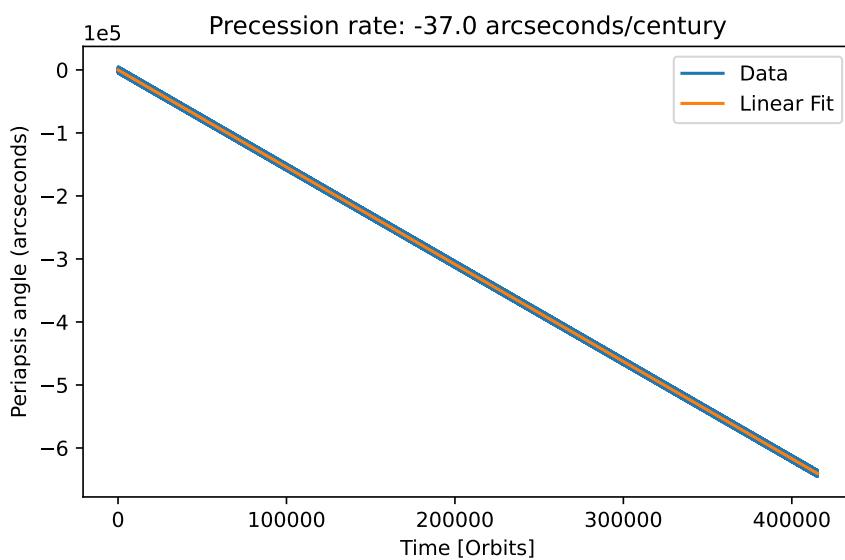


Figure 5.12: Linear change in Mercury's perihelion angle.

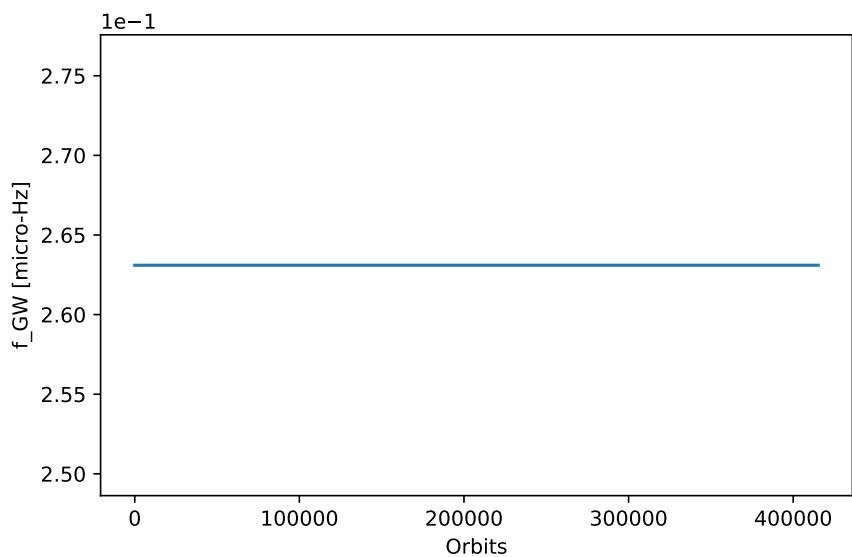


Figure 5.13: Leading order frequency Mercury's GW

Chapter 6

Conclusion

Computational physics covers all domains of physics and science. Because real world problems are so much more complex than theory can manipulate, numerical methods are needed in order to make realistic predictions. Many of these numerical implementations require the use of high-performance methods. Computational physics is a vast field with many opportunities in which to apply both a deep understanding of physics, math, and computer science.

Bibliography

- [1] G. Smith (<https://physics.stackexchange.com/users/199630/g-smith>). *Difficulty of numerically solving Einstein equations*. Physics Stack Exchange. URL:<https://physics.stackexchange.com/q/443754> (version: 2018-11-28). eprint: <https://physics.stackexchange.com/q/443754>. URL: <https://physics.stackexchange.com/q/443754>.
- [2] Tomas Andrade et al. “GRChombo: An adaptable numerical relativity code for fundamental physics”. In: *Journal of Open Source Software* 6.68 (2021), p. 3703. DOI: [10.21105/joss.03703](https://doi.org/10.21105/joss.03703). URL: <https://doi.org/10.21105/joss.03703>.
- [3] Richard Arnowitt, Stanley Deser, and Charles W. Misner. “Republication of: The dynamics of general relativity”. In: *General Relativity and Gravitation* 40.9 (Aug. 2008), pp. 1997–2027. ISSN: 1572-9532. DOI: [10.1007/s10714-008-0661-1](https://dx.doi.org/10.1007/s10714-008-0661-1). URL: <http://dx.doi.org/10.1007/s10714-008-0661-1>.
- [4] Thomas W. Baumgarte and Stuart L. Shapiro. *Numerical Relativity: Starting from Scratch*. Cambridge University Press, 2021. DOI: [10.1017/9781108933445](https://doi.org/10.1017/9781108933445).
- [5] Valerie Beale and Nathan Chapman. “Distance and Plate Separation Dependence of Fringing Fields of Parallel Plate Capacitors”. In: (2019).
- [6] Nathan Chapman. *A Toy Model of the Interaction between Caffeine and Stomach Acid*. 2023.
- [7] Nathan Chapman. “Effects of Eccentricity on the Precession of Orbital Periapsides due to General Relativity in the Weak Field Limit”. In: (June 2019). eprint: \url{https://github.com/NonDairyNeutrino/ug-computation/blob/1e8f14936b1d7f5ddd2cd29961eee56ba7820a30/computational_physics/Weak_Field_GR_Eccentricity_Effects/Weak_Field_GR_Eccentricity_Effects.pdf}.
- [8] Nathan Chapman. “Effects of eccentricity on the precession of orbital periapsides due to General Relativity in the weak field limit”. In: (2019).
- [9] Nathan Chapman. “Leading Order Gravitational Wave Frequencies from Short-Term Post-Newtonian Orbits via Fourth-Order Symplectic Integration”. In: (2022).
- [10] Katy Clough et al. “GRChombo: Numerical relativity with adaptive mesh refinement”. In: *Classical and Quantum Gravity* 32.24 (Dec. 2015), p. 245011. DOI: [10.1088/0264-9381/32/24/245011](https://doi.org/10.1088/0264-9381/32/24/245011). URL: <https://doi.org/10.1088/0264-9381/32/24/245011>.
- [11] C.J. Cramer. *Essentials of Computational Chemistry: Theories and Models*. Wiley, 2013. ISBN: 9781118712276. URL: <https://books.google.com/books?id=k4R6cf7I7q0C>.
- [12] Albert Einstein. “On the electrodynamics of moving bodies”. In: (1905).
- [13] D.J. Griffiths. *Introduction to Electrodynamics*. Cambridge University Press, 2017. ISBN: 9781108357142. URL: <https://books.google.com/books?id=Kh4xDwAAQBAJ>.
- [14] David Hanson et al. “8.3: Hartree-Fock Equations are Solved by the Self-Consistent Field Method”. In: LibreTexts.org. Chap. 8: Multielectron Atoms.
- [15] Oliver James et al. “Gravitational lensing by spinning black holes in astrophysics, and in the movie Interstellar”. In: *Classical and Quantum Gravity* 32.6 (Feb. 2015), p. 065001. DOI: [10.1088/0264-9381/32/6/065001](https://doi.org/10.1088/0264-9381/32/6/065001). URL: <https://doi.org/10.1088/0264-9381/32/6/065001>.
- [16] C Körber et al. “A primer to numerical simulations: the perihelion motion of Mercury”. In: *Physics Education* 53.5 (June 2018), p. 055007. DOI: [10.1088/1361-6552/aac487](https://doi.org/10.1088/1361-6552/aac487). URL: <https://doi.org/10.1088/1361-6552/aac487>.

- [17] Christopher Körber et al. “A primer to numerical simulations: the perihelion motion of Mercury”. In: *Physics Education* 53.5 (2018), p. 055007.
- [18] MIT Game Lab. *OpenRelativity*. Online. 2012. URL: <http://gamelab.mit.edu/research/openrelativity/>.
- [19] Zhi-Han Li, Chen-Qi Li, and Long-Gang Pang. *Solving Einstein equations using deep learning*. 2023. arXiv: 2309.07397 [gr-qc].
- [20] Jacques-Louis Lions, Yvon Maday, and Gabriel Turinici. “A “parareal” in time discretization of PDEs”. In: *Comptes Rendus de l’Académie des Sciences - Series I - Mathematics* 332.7 (2001), pp. 661–668. ISSN: 0764-4442. DOI: [https://doi.org/10.1016/S0764-4442\(00\)01793-6](https://doi.org/10.1016/S0764-4442(00)01793-6). URL: <https://www.sciencedirect.com/science/article/pii/S0764444200017936>.
- [21] C.W. Misner et al. *Gravitation*. Princeton University Press, 2017. ISBN: 9781400889099. URL: <https://books.google.com/books?id=zAAuDwAAQBAJ>.
- [22] M Misner, K. Thorne, and J. Wheeler. *Gravitation*. Freeman, 1972.
- [23] Dhananjay Saikumar. *Parallel N-body simulations of planetary systems: a direct approach*. 2022. arXiv: 2208.13562 [astro-ph.EP].
- [24] Simulating eXtreme Spacetimes. *Two Black Holes Merge into One*. Youtube. 2016. URL: https://youtu.be/I_88S8DWbcU?si=fKwrNrQl8TBqaPIg.
- [25] J.R. Taylor. *Classical Mechanics*. G - Reference,Information and Interdisciplinary Subjects Series. University Science Books, 2005. ISBN: 9781891389221. URL: <https://books.google.com/books?id=P1kCtNr-pJsC>.
- [26] Jack Wisdom and Matthew Holman. “Symplectic maps for the N-body problem.” In: *aj* 102 (Oct. 1991), pp. 1528–1538. DOI: [10.1086/115978](https://doi.org/10.1086/115978).
- [27] Haruo Yoshida. “Construction of higher order symplectic integrators”. In: *Physics Letters A* 150.5 (1990), pp. 262–268. ISSN: 0375-9601. DOI: [https://doi.org/10.1016/0375-9601\(90\)90092-3](https://doi.org/10.1016/0375-9601(90)90092-3). URL: <https://www.sciencedirect.com/science/article/pii/0375960190900923>.

Index

- Acceleration, 3
- Basis Set, 13
- Density Functional Theory, 14
- Derivative, 3, 5
 - First Order, 5
 - Order, 5
- Differential Equation, 4, 5
 - Coupled, 7
 - Homogenous, 3
 - Ordinary, 2
 - Second Order, 3
 - System, 3, 7
- Differential Equations
 - Partial, 7
- Discretize, 5
- Einstein Field Equations, 18
- Electromagnetism, 2
- Energy, 3, 6
 - Conservation, 5
 - Drift, 6
- Equation of Motion, 3, 4
- Error, 6
- Euler, 5
- Finite Differences, 5, 6
 - Accuracy, 5
 - Backward, 5
 - Central, 5
 - Coefficients, 5
 - First Order, 5
 - Forward, 5
- Gravity, 4
- Hamiltonian, 5, 13
- Implicit, 5
- Integrator, 5
- Symplectic, 2, 5, 6
- Mass, 5
- Maxwell's Equations, 7
- Mechanics, 2, 3, 5, 6
 - Classical, 2, 3
 - Quantum, 2
- Minimization, 14
- N-Body, 4
- Newton
 - 2nd Law, 3, 13
 - Gravitation, 4
- Nonlinear, 3
- Numerical Analysis, 2
- Oscillatory, 5
- Parallel, 6
- Position, 3, 4
- Relativity, 2, 18
 - General, 18
 - Numerical, 18
 - Special, 18
- Runge-Kutta, 2
- Schrödinger
 - Equation, 13
- Spacetime, 18
- Spring, 5
- Step
 - Time, 6
- Symplectic, 2
- System
 - Coordinate, 18
- Thermodynamics, 2
- Time, 5, 6
- Variational Principle, 13
- Velocity, 4, 5
- Verlet, 5