

# Physcis 20.3 Numerical Solutions of Ordinary Differential Equations

Sophie Li

February 2019

## 1 The Version Control Log

```
commit d07572c0147efd0bb0822f40193b25178725838c
Author: Sophie Li <shli@caltech.edu>
Date: Thu Feb 21 01:21:40 2019 -0800
```

changes before trying to make a log file

```
commit 49a2b64a1503ba1df8236152eb326d8efed01362
Author: Sophie Li <shli@caltech.edu>
Date: Wed Feb 20 17:21:24 2019 -0800
```

started to make makefile

```
commit d29843c5f04a3e0219dda54493aa35690b25843d
Author: Sophie Li <shli@caltech.edu>
Date: Wed Feb 20 16:32:29 2019 -0800
```

changes so that images could be produced through sys.argv

```
commit 0010a31e51aa81fc76ec828b4370c386bd32c58f
Author: Sophie Li <shli@caltech.edu>
Date: Wed Feb 20 15:25:45 2019 -0800
```

putting everything into the repos

```
commit 1e3011f15451b8afbd8847bebc52562cacbffcd2
Author: Sophie Li <shli@caltech.edu>
Date: Wed Feb 13 14:55:03 2019 -0800
```

Add concerns about effects of Mars' moons on Wolfman

```
commit 98922d113ca1d740d35612ef64f31f85080629ef
```

Author: Sophie Li <shli@caltech.edu>  
Date: Wed Feb 13 14:48:22 2019 -0800

Start notes on mars as a bas

## 2 The Makefile

```
# Make images
all: expliciteuler.png globalerrorexp.png totalenergy.png phaseexp.png impliciteuler.png gl

# Explicit Images
expliciteuler.png : numsol.py
python3 numsol.py 0.004 10000 expliciteuler

globalerrorexp.png : numsol.py
python3 numsol.py 0.005 50000 globalerrorexp

totalenergy.png : numsol.py
python3 numsol.py 0.004 100000 totalenergy

phaseexp.png : numsol.py
python3 numsol.py 0.004 50000 phaseexp

# Implicit Images
impliciteuler.png : numsolimp.py
python3 numsolimp.py 0.004 10000 impliciteuler

globalerrorimp.png : numsolimp.py
python3 numsolimp.py 0.004 50000 globalerrorimp

totalenergyimp.png : numsolimp.py
python3 numsolimp.py 0.004 100000 totalenergyimp

phaseimp.png : numsolimp.py
python3 numsolimp.py 0.004 50000 phaseimp

# Truncation error images
truncerror.png : numsolimp.py
python3 truncerror.py

# Symplectic Images
phasesymp.png : numsolsym.py
python3 numsolsym.py 0.004 50000 phasesymp

phasesymplarge.png : numsolsym.py
```

```
python3 numsolsym.py 0.04 50000 phasesymplarge
```

```
totalenergysym.png : numsolsym.py  
python3 numsolsym.py 0.004 5000 totalenergysym
```

```
numsol.pdf : main.tex  
pdflatex main.tex
```

### 3 The Source Code

```
\documentclass{article}  
\usepackage[utf8]{inputenc}  
\usepackage{graphicx,amsmath,subcaption,verbatim}  
  
\title{Phycis 20.3 Numerical Solutions of Ordinary Differential Equations}  
\author{Sophie Li}  
\date{February 2019}  
  
\begin{document}  
  
\maketitle  
  
\section{The Version Control Log}  
\verbatiminput{log.txt}  
\section{The Makefile}  
\verbatiminput{Makefile}  
\section{The Source Code}  
\verbatiminput{main.tex}  
\section{The Motion of a Mass on a Spring}  
Figure \ref{fig:expliciteuler} shows a few cycles of oscillation when the explicit Euler method  
\begin{figure}[h]  
    \centering  
    \includegraphics[width = \textwidth]{Images/expliciteuler.png}  
    \caption{Motion of a mass on a spring implementing the explicit Euler method}  
    \label{fig:expliciteuler}  
\end{figure}  
  
\section{Analytic Solution}  
\subsection{Derivation of the Analytic Solution}  
Rearranging  $F=ma=-kx$  we have that  $a=-\frac{k}{m}x$ . Let  $a=\frac{d^2x}{dt^2}$  and  $\frac{d^2x}{dt^2}=-\frac{k}{m}x$ .  
This can be rewritten in terms of two new constants  $B$  and  $C$ .  

$$x(t)=B\cos{\omega t}+C\sin{\omega t}$$
  
By finding the derivative of the above equation we get that  

$$v(t)=-B\omega\sin{\omega t}+C\omega\cos{\omega t}$$
  
To find the constants  $C$  and  $D$ , substitute  $t=0$  into both of the equations.
```

```


$$v_0 = v(0) = -\omega \sin\{0\} + C\omega \cos\{0\} = \omega C$$

So,  $C = \frac{v_0}{\omega}$ 

$$x_0 = x(0) = B\cos\{0\} + C\sin\{0\} = B$$

So,  $B = x_0$ 
Therefore we have the general solution

$$x(t) = x_0 \cos\{\omega t\} + \frac{v_0}{\omega} \sin\{\omega t\}$$


$$v(t) = -x_0 \omega \sin\{\omega t\} + v_0 \cos\{\omega t\}$$

But because  $\omega = 1$  in this case we have that the final analytical solution to the mass
\begin{equation}
x(t) = x_0 \cos\{t\} + v_0 \sin\{t\}
\end{equation}
\begin{equation}
v(t) = -x_0 \sin\{t\} + v_0 \cos\{t\}
\end{equation}
\subsection{Global Errors}
Figure \ref{fig:globalerrorexp} shows that as the global error increases exponentially with
\begin{figure}[h]
\centering
\includegraphics[width = \textwidth]{Images/globalerrorexp.png}
\caption{Global errors between the analytic solution and the explicit Euler method}
\label{fig:globalerrorexp}
\end{figure}

\section{Truncation Error}
Figure \ref{fig:truncerror} shows that the truncation error is proportional to  $h^4$  for small
\begin{figure}[h]
\centering
\includegraphics[width = \textwidth]{Images/truncerror.png}
\caption{Maximum value of  $x_{\text{analytic}} - x$  versus  $h$  for several different  $h$ 's}
\label{fig:truncerror}
\end{figure}

\section{Numerical Evolution of Total Energy}
One can calculate the normalized total energy of a physical system by  $E = x^2 + v^2$ . Figure \ref{fig:totalenergy}
\begin{figure}[h]
\centering
\includegraphics[width = \textwidth]{Images/totalenergy.png}
\caption{Long term evolution of the total energy with time, explicit Euler method}
\label{fig:totalenergy}
\end{figure}

\section{The Implicit Euler Equation}
\subsection{Expressions for the Implicit Euler Method}
The implicit Euler method requires the unknown values of  $x_{i+1}$  to update the known values
\begin{equation}
\begin{pmatrix}
\end{pmatrix}

```

```

1&-h\\
h&1
\end{pmatrix}
\cdot
\begin{pmatrix}
x_{i+1}\\
v_{i+1}
\end{pmatrix}
=
\begin{pmatrix}
x_i\\
v_i
\end{pmatrix}
\end{equation}
However, we want to find an expression for the unknown  $x_{i+1}$  in terms of the known values
\begin{equation}
\begin{pmatrix}
1&-h\\
h&1
\end{pmatrix}^{-1}
= \frac{1}{1+h^2} \cdot
\begin{pmatrix}
1&h\\
-h&1
\end{pmatrix}
=
\begin{pmatrix}
\frac{1}{1+h^2} & \frac{h}{1+h^2} \\
\frac{-h}{1+h^2} & \frac{1}{1+h^2}
\end{pmatrix}
\end{equation}
We can then use this inverse matrix to find the values of  $x_{i+1}$  and  $v_{i+1}$  in terms of
\begin{equation}
\begin{pmatrix}
\frac{1}{1+h^2} & \frac{h}{1+h^2} \\
\frac{-h}{1+h^2} & \frac{1}{1+h^2}
\end{pmatrix}
\cdot
\begin{pmatrix}
x_i\\
v_i
\end{pmatrix}
=
\begin{pmatrix}
x_{i+1}\\
v_{i+1}
\end{pmatrix}

```

```

v_{i+1}
\end{pmatrix}
\label{linsys2}
\end{equation}
Using equation \ref{linsys2} we can get the implicit Euler method in terms of  $x_i$  and  $v_i$ 
\begin{equation}
x_{i+1}=\frac{x_i}{1+h^2}+\frac{v_i}{1+h^2}, v_{i+1}=\frac{v_i}{1+h^2}-\frac{x_i}{1+h^2}
\end{equation}
If we plot these expressions much we get a graph like in figure \ref{fig:impliciteuler}. The
\begin{figure}[h]
\centering
\includegraphics[width = \textwidth]{Images/impliciteuler.png}
\caption{Implementing the implicit Euler equation}
\label{fig:impliciteuler}
\end{figure}
\subsection{Global Errors}
If we plot the global errors generated using the implicit functions we get a graph like in
\begin{figure}[h]
\centering
\includegraphics[width = \textwidth]{Images/globalerrorimp.png}
\caption{Global errors between the analytic solution and the implicit Euler method}
\label{fig:globalerrorimp}
\end{figure}
\subsection{Evolution of Total Energy}
Unlike in figure \ref{fig:totalenergy} for the explicit Euler equation, where the energy grows
\begin{figure}[h]
\centering
\includegraphics[width = \textwidth]{Images/totalenergyimp.png}
\caption{Long term evolution of the total energy with time, implicit Euler method}
\label{fig:totalenergyimp}
\end{figure}

\section{Phase-space Trajectories}
Figure \ref{fig:phase} shows very clearly that the errors in the explicit method deviate pos
\begin{figure}[h]
\centering
\begin{subfigure}{0.7\textwidth}
\includegraphics[width =\textwidth]{Images/phaseexp.png}
\caption{Phase-space trajectory produced by the explicit Euler method}
\end{subfigure}
\begin{subfigure}{0.7\textwidth}
\includegraphics[width =\textwidth]{Images/phaseimp.png}
\caption{Phase-space trajectory produced by the implicit Euler method}
\end{subfigure}
\caption{Phase-space trajectories produced by the implicit and explicit Euler methods. The}
\label{fig:phase}

```

```

\end{figure}

\section{Symplectic Euler Method}
\subsection{Phase-space trajectory}
When the phase space for the symplectic method is plotted along side the analytic for the va
\begin{figure}[h]
    \begin{subfigure}{0.5\textwidth}
        \includegraphics[width =\textwidth]{Images/phasesymp.png}
        \caption{Phase-space trajectory produced by the symplectic method at $h=0.004$}
    \end{subfigure}
    \begin{subfigure}{0.5\textwidth}
        \includegraphics[width =\textwidth]{Images/phasesymplarge.png}
        \caption{Phase-space trajectory produced by the symplectic method at $h=0.04$}
    \end{subfigure}
    \caption{Phase-space trajectories produced by the symplectic Euler method at different v
    \label{fig:phasesymp}
\end{figure}
\subsection{Evolution of total energy}
Figure \ref{fig:totalenergysym} shows the long term evolution of the total energy of the sym

\end{document}

```

## 4 The Motion of a Mass on a Spring

Figure 1 shows a few cycles of oscillation when the explicit Euler method is implemented. This graph was made with initial condition set to  $h = 0.004$ ,  $N = 10000$ ,  $x_0 = 2$ ,  $y_0 = 2$ ,  $t_0 = 0$ .

## 5 Analytic Solution

### 5.1 Derivation of the Analytic Solution

Rearranging  $F = ma = -kx$  we have that  $a = -\frac{k}{m}x$ . Let  $a = \frac{d^2x}{dt^2}$  and  $\frac{k}{m} = \omega^2$ . We then have  $a = \frac{d^2x}{dt^2} = \omega^2x$ . Using the Ansatz principle we guess that the solution to this equation is in its most general form

$$x(t) = A \cos \phi \cos \omega t - A \sin \phi \sin \omega t$$

. This can be rewritten in terms of two new constants  $B$  and  $C$ .

$$x(t) = B \cos \omega t + C \sin \omega t$$

. By finding the derivative of the above equation we get that

$$v(t) = -B\omega \sin \omega t + C\omega \cos \omega t$$

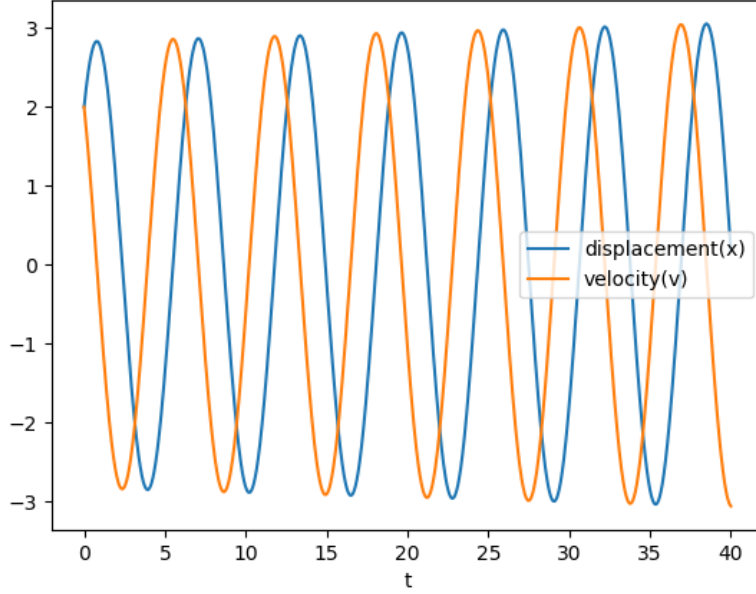


Figure 1: Motion of a mass on a spring implementing the explicit Euler method

To find the constants  $C$  and  $D$ , substitute  $t = 0$  into both of the equations.

$$v_0 = v(0) = -\omega \sin 0 + C\omega \cos 0 = \omega C$$

So,  $C = \frac{v_0}{\omega}$

$$x_0 = x(0) = B \cos 0 + C \sin 0 = B$$

So,  $B = x_0$  Therefore we have the general solution

$$x(t) = x_0 \cos \omega t + \frac{v_0}{\omega} \sin \omega t$$

$$v(t) = -x_0 \omega \sin \omega t + v_0 \cos \omega t$$

But because  $\omega = 1$  in this case we have that the final analytical solution to the mass of the spring is

$$x(t) = x_0 \cos t + v_0 \sin t \quad (1)$$

$$v(t) = -x_0 \sin t + v_0 \cos t \quad (2)$$

## 5.2 Global Errors

Figure 2 shows that as the global error increases exponentially with time. This graph was made with initial conditions set to  $h = 0.005$ ,  $N = 50000$ ,  $x_0 = 2$ ,  $y_0 = 2$ ,  $t_0 = 0$ . The increasing error could have also been seen at a lower  $N$ , however, a larger  $N$  was chosen to demonstrate the exponential increase.



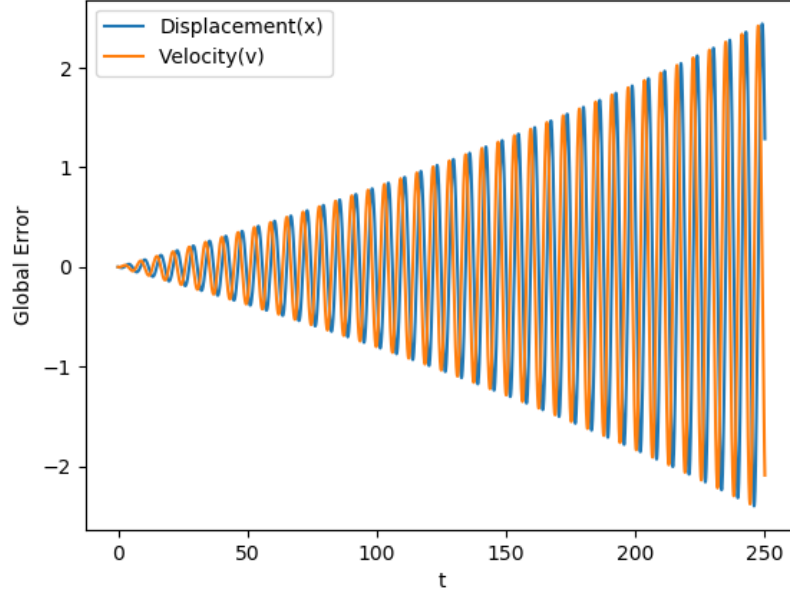


Figure 2: Global errors between the analytic solution and the explicit Euler method

## 6 Truncation Error

Figure 3 shows that the truncation error is proportional to  $h$  for small values of  $h$ . We can see that as  $h$  increases, so does the truncation error. I chose  $h_0 = 0.001$  and then calculated the error for  $h_0, h_0/2, h_0/4, h_0/8, h_0/16$  to produce this plot.

## 7 Numerical Evolution of Total Energy

One can calculate the normalized total energy of a physical system by  $E = x^2 + v^2$ . Figure 4 shows how this energy evolves over a long period of time. The trend we can see is that the energy increases exponentially with time. We can see that this is similar to the global error which we also saw to be increasing over time, see figure 2. For this physical system we would actually expect the energy to remain constant as energy should be conserved. However, it is evident that due to the exponentially increasing error,  $E$  also increases exponentially. The parameters used for the plot of figure 4 are  $h = 0.004, N = 100000, x_0 = 2, y_0 = 2, t_0 = 0$ .

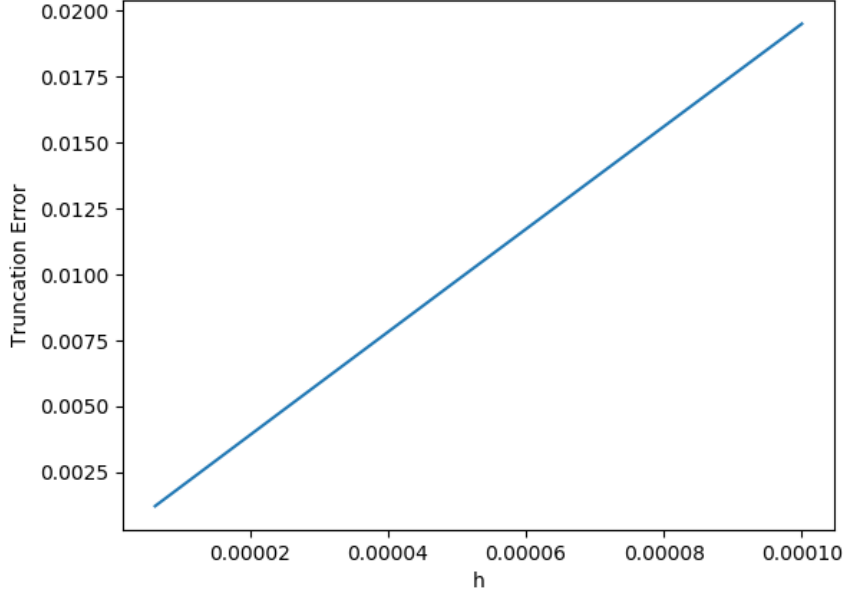


Figure 3: Maximum value of  $x_{analytic} - x$  versus  $h$  for several different  $h$

## 8 The Implicit Euler Equation

### 8.1 Expressions for the Implicit Euler Method

The implicit Euler method requires the unknown values of  $x_{i+1}$  to update the known values of  $x_i$ . This can be encoded in the linear system

$$\begin{pmatrix} 1 & -h \\ h & 1 \end{pmatrix} \cdot \begin{pmatrix} x_{i+1} \\ v_{i+1} \end{pmatrix} = \begin{pmatrix} x_i \\ v_i \end{pmatrix} \quad (3)$$

However, we want to find an expression for the unknown  $x_{i+1}$  in terms of the known value  $x_i$ . To do this we must first find the inverse of the first matrix. This gives

$$\begin{pmatrix} 1 & -h \\ h & 1 \end{pmatrix}^{-1} = \frac{1}{1+h^2} \cdot \begin{pmatrix} 1 & h \\ -h & 1 \end{pmatrix} = \begin{pmatrix} \frac{1}{1+h^2} & \frac{h}{1+h^2} \\ \frac{-h}{1+h^2} & \frac{1}{1+h^2} \end{pmatrix} \quad (4)$$

We can then use this inverse matrix to find the values of  $x_{i+1}$  and  $v_{i+1}$  in terms of  $x_i$  and  $v_i$ . We get the linear system

$$\begin{pmatrix} \frac{1}{1+h^2} & \frac{h}{1+h^2} \\ \frac{-h}{1+h^2} & \frac{1}{1+h^2} \end{pmatrix} \cdot \begin{pmatrix} x_i \\ v_i \end{pmatrix} = \begin{pmatrix} x_{i+1} \\ v_{i+1} \end{pmatrix} \quad (5)$$

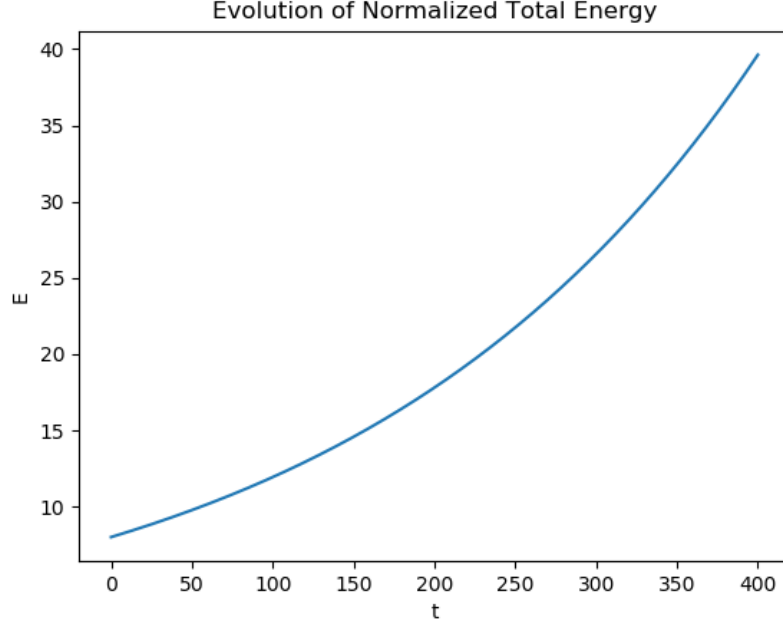


Figure 4: Long term evolution of the total energy with time, explicit Euler method

Using equation 5 we can get the implicit Euler method in terms of  $x_i$  and  $v_i$

$$x_{i+1} = \frac{x_i}{1+h^2} + \frac{v_i}{1+h^2}, v_{i+1} = \frac{v_i}{1+h^2} - \frac{x_i}{1+h^2} \quad (6)$$

If we plot these expressions much we get a graph like in figure 5. This plot looks extremely similar to the plot generated when implementing the explicit Euler equations. We can compare it to figure 1 because the parameters used are exactly the same.

## 8.2 Global Errors

If we plot the global errors generated using the implicit functions we get a graph like in fig 6. This graph shows an evolution of total energy similar to figure 2 for the explicit Euler equation. However, the error does not exponentially increase like in figure 2, instead it first increases and then tails off in a logarithmic fashion. Again, the graphs are comparable because I used the same parameters as for the explicit method.

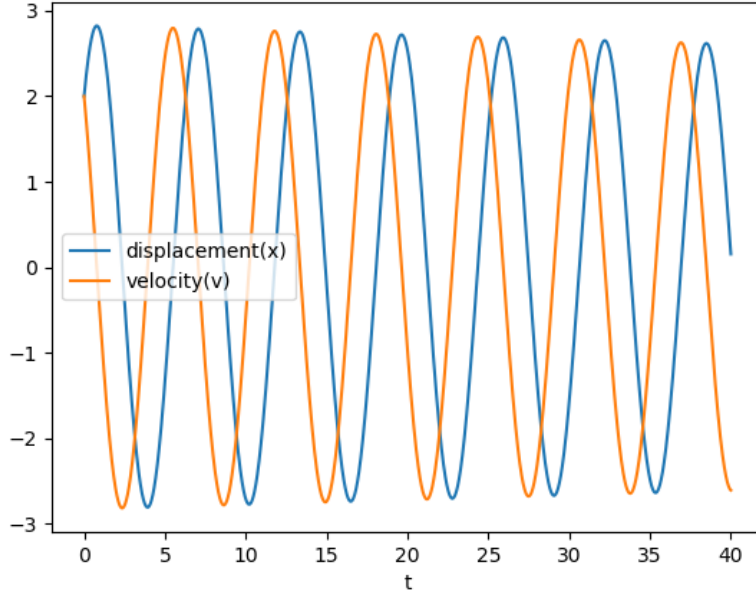


Figure 5: Implementing the implicit Euler equation

### 8.3 Evolution of Total Energy

Unlike in figure 4 for the explicit Euler equation, where the energy grew exponentially, the energy in figure 7 decreases exponentially. Again, the energy is not constant. Thus, the implicit Euler method also does not model the physical system such that energy is conserved. We saw that the error instead of increasing exponentially only had a logarithmic increase. It is because of this that we see a decrease in energy and the errors become greater for the implicit Euler method.

## 9 Phase-space Trajectories

Figure 8 shows very clearly that the errors in the explicit method deviate positively and the errors in the implicit method deviate negatively. We can see this because in the explicit Euler phase-space the deviation from the analytic solutions are towards the outside of the circle. This means that there are always positive errors calculated by this solution. On the other hand, for the implicit Euler method, the deviations are inside the orange circle and therefore, the errors were always negative.

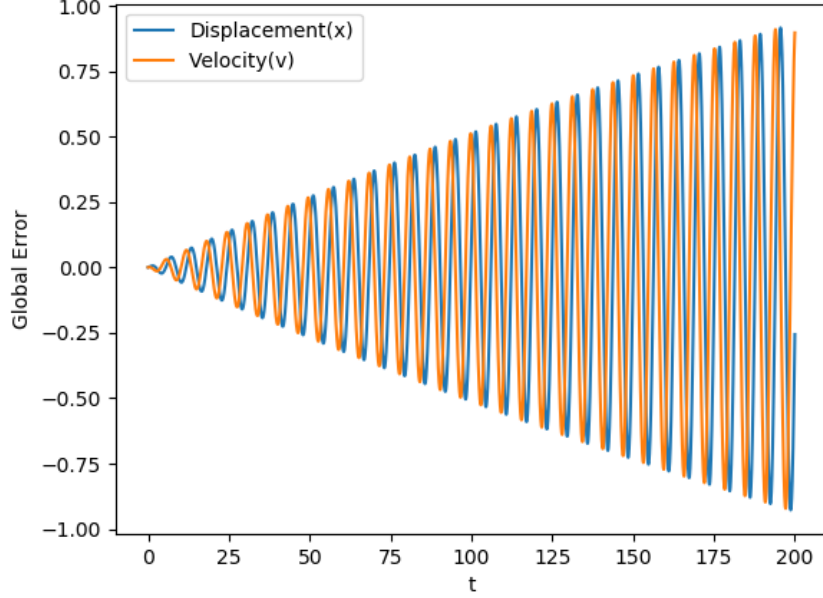


Figure 6: Global errors between the analytic solution and the implicit Euler method

## 10 Symplectic Euler Method

### 10.1 Phase-space trajectory

When the phase space for the symplectic method is plotted along side the analytic for the value of  $h = 0.004$  that was also used in figure 8, we can see no difference between the analytic solutions and the solutions generated by the symplectic method. However, if the value of  $h$  is decreased to  $h = 0.04$  we have figure 9(b). The circle of the symplectic method slightly deviates from the analytic solutions, however, it corrects itself and returns back. Thus, it never diverges in one particular direction from the orange circle like in figure 8.

### 10.2 Evolution of total energy

Figure 10 shows the long term evolution of the total energy of the symplectic Euler method compared to the analytical total energy. The analytic energy holds a constant value as expected because energy is conserved in the physical system. We can see that the energy of the symplectic grows, however when it reaches a certain height it returns to the total energy for the analytic solution. This suggests that the error of the symplectic method is in a sense, self-correcting.

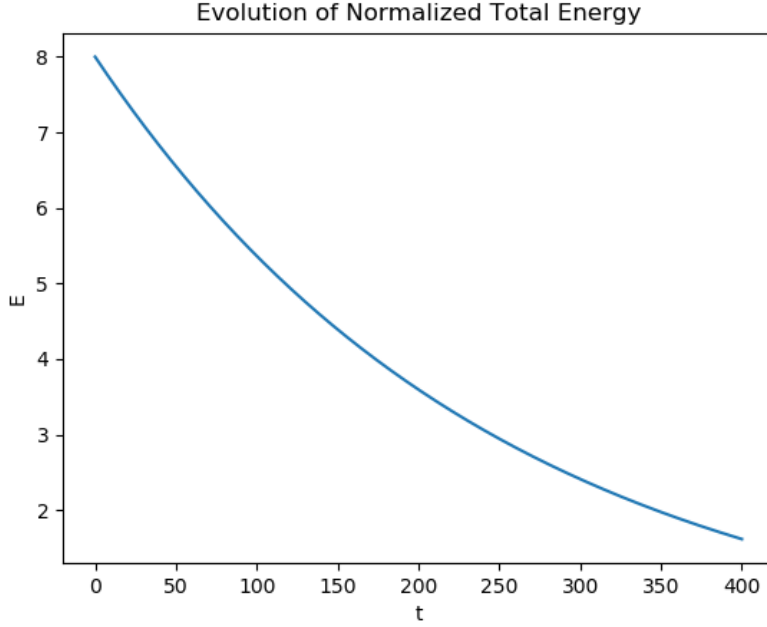
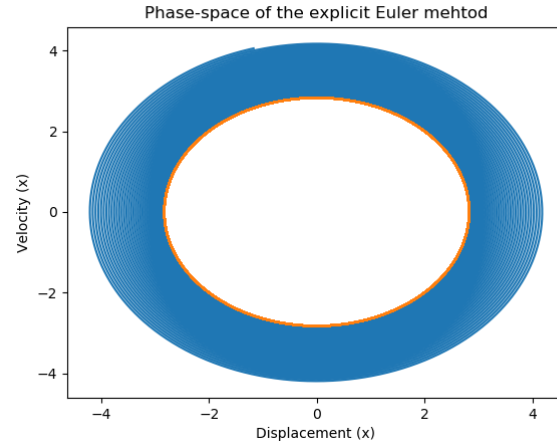
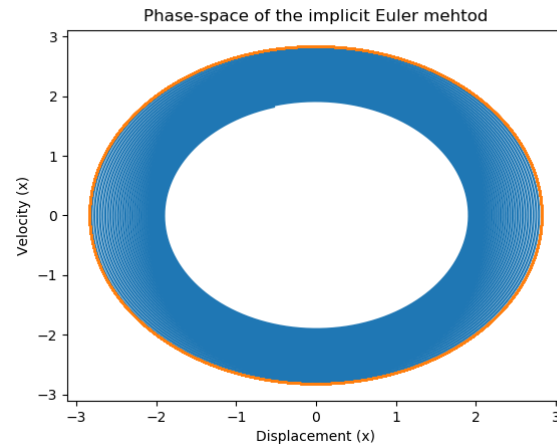


Figure 7: Long term evolution of the total energy with time, implicit Euler method

Although it does not fully conserve energy, it returns back to a given point. And therefore it never diverges significantly like in the explicit or implicit Euler methods. We can also relate figure 10 to 9(b). The way the energy error is corrected back is similar to the way that the symplectic trajectory deviated yet returned to the trajectory of the analytic solution. This suggests that the error created by the symplectic method is both positive and negative and thus self-correcting.

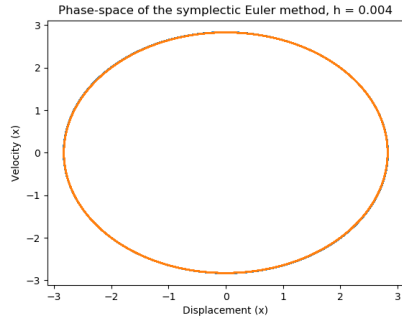


(a) Phase-space trajectory produces by the explicit Euler method

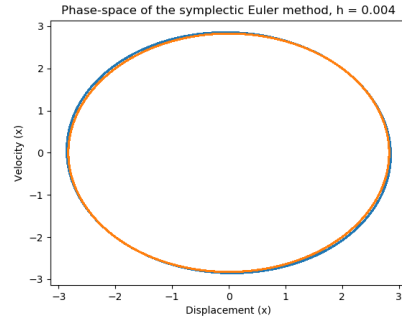


(b) Phase-space trajectory produces by the implicit Euler method

Figure 8: Phase-space trajectories produced by the implicit and explicit Euler methods. The orange circle in both represents the trajectory produced by the analytic solutions



(a) Phase-space trajectory produced by the symplectic method at  $h = 0.004$



(b) Phase-space trajectory produced by the symplectic method at  $h = 0.04$

Figure 9: Phase-space trajectories produced by the symplectic Euler method at different values of  $h$ . The orange circle in both represents the trajectory produced by the analytic solutions