TABLE of CONTENTS

1) INTRODUCTION

Multiple mass-spring systems are used in most of engineering modeling and solving many engineering problems. A system consisting of masses connected to each other with springs is a classical system with different degrees of freedom. For example, a system consisting of three masses and two springs connecting the masses has 3 degrees of freedom. This shows that the configuration of the system can be defined with 3 different generalized coordinates which are the position vectors of the masses(x_1, x_2, x_3). The motions of the masses can be described with 3 different second degree differential equations. In the simplest form of this configuration, external factors such as friction force and air resistance can be ignored, and only eleastic forces based on Hooke's Law can be considered. In general, the character of the movement can be determined by 3 different eigenfrequencies based on system parameters, masses and spring constants. Also, the movements of the masses are strongly dependent on their initial positions and initial velocities. So far, one solution to this type of system was considered but there is another useful solution using Verlet Algorithm. Both solution will give a harmonic response since springs are such tools trying to reach their equilibrium positions, doing so, they also overshoot and undershoot equilibrium positions since there were initial inputs preventing the system from re-reaching its equilibrium. In this paper, the nature of the multiple mass-spring system is considered by using 2 different methods and investigating position-time and velocity-time relations.

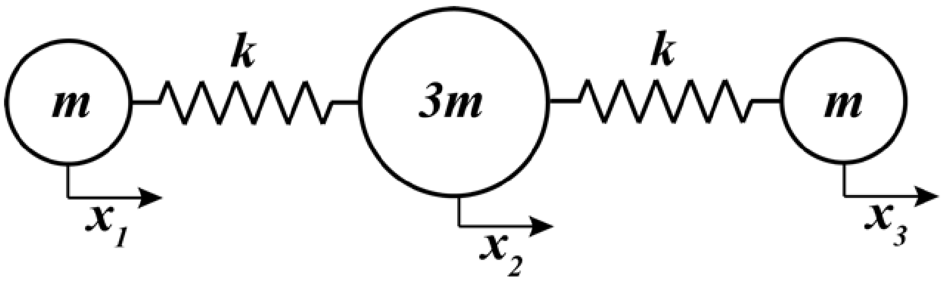

Figure.1: Multiple mass-spring system consisting of three masses and two springs

In the configuration above(Figure.1) There are three free-to-move masses that have different initial positions and velocities(velocities do not have to be different). To give an example, if one moves the m_3 to the right, all configuration is affected by this external effect since the masses are free to move and springs undertake connection mission which allows other masses get affected from the external effect. An external effect does not have to be applied to see the configuration change with time. The masses already have their initial velocities and positions affecting each other by the help of springs.

2) METHODS-DISCUSSION

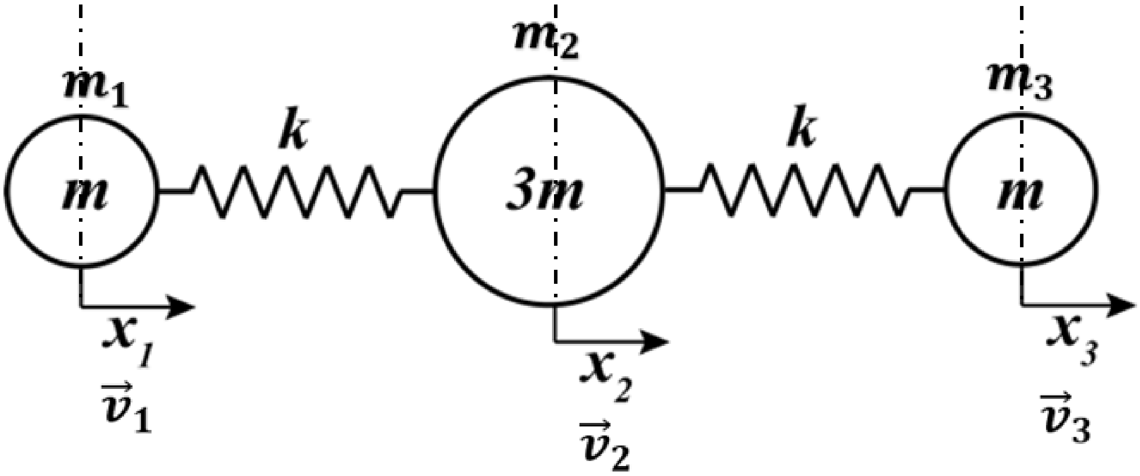

Figure.2: Multiple mass-spring system, velocity and position components shown

x_1, x_2 and x_3 are the deviation positions from the dashed lines.

m, 3m and m are the masses, and v_1, v_2 and v_3 are the velocities of m_1, m_2 and m_3 respectively.

$k_1$and k_2 are the spring constants and they are both equal to k.

If m_1 moves with x_1, m_2 moves with x_2 and m_3 moves with x_3 then the masses will have acceleration due to deviation from springs' equilibrium poisitons, that is, springs will get stretched or compressed.

In general: $F = -kx \quad \text{where F is force, k is spring constant and x is deviation from equilibrium}$

Using the force equations for each of the masses:

$m_1 a_1 = k_1 x_2 - k_1 x_1 = k_1(x_2 - x_1)\ where\ a_1\ is\ the\ acceleration\ of\ m_1$

$m_2 a_2 = k_1 x_1 - k_1 x_2 + k_2 x_3 - k_2 x_2 = k_1(x_1 - x_2) + k_2(x_3 - x_2)$
$where\ a_2\ is\ the\ acceleration\ of\ m_2$

$m_3 a_3 = k_2 x_2 - k_2 x_3 = k_2(x_2 - x_3)\ where\ a_3\ is\ the\ acceleration\ of\ m_3$

Accelerations can be rewritten in the form of second derivative of positions wrt. time:

$$m_1 \frac{d^2 x_1}{dt^2} = (x_2 - x_1)$$

$$m_2 \frac{d^2 x_2}{dt^2} = k_1(x_1 - x_2) + k_2(x_3 - x_2)$$

$$m_3 \frac{d^2 x_3}{dt^2} = (x_2 - x_3)$$

These three equations can be represented in one matrix equation:

$$\underbrace{\frac{d^2}{dt^2}\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}}_{\overline{a(t)}} = \underbrace{\begin{bmatrix} -\frac{k_1}{m_1} & \frac{k_1}{m_1} & 0 \\ \frac{k_1}{m_2} & \frac{-k_1-k_2}{m_2} & \frac{k_2}{m_2} \\ 0 & \frac{k_2}{m_2} & -\frac{k_2}{m_3} \end{bmatrix}}_{\bar{\bar{M}}} * \underbrace{\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}}_{\overline{x(t)}}$$

The left hand side is the acceleration vectors, the secınd term on the right hand side is the position vectors and the remaining matrix is the rest of the equation.

2.a) Analytical Solution

For the analytical solution, eigenvectors and eigenvalues are needed.

Eigenvalues and eigenvectors make it easier to find solutions by transforming a linear operation into simpler and separate problems. For example, they are used in a multiple spring-mass system as discussed here to further simplify a complex systems equation.

To give an example how to derive eigenvectors and eigenvalues:

$$\mathrm{A} = \begin{bmatrix} \mathrm{a}_1 & \mathrm{a}_2 & \mathrm{a}_3 \\ \mathrm{b}_1 & \mathrm{b}_2 & \mathrm{b}_3 \\ \mathrm{c}_1 & \mathrm{c}_2 & \mathrm{c}_3 \end{bmatrix} \text{ where the entries } \mathrm{a}_1,\ \mathrm{a}_2, \text{up to } \mathrm{c}_3 \text{ are real numbers}$$

Trace is defined as the sum of the entries in the diagonal:

$$\mathrm{tr(A)} = \mathrm{a}_1 + \mathrm{b}_2 + \mathrm{c}_3$$

Determinant of the matrix:

$$|\mathrm{A}| = \mathrm{a}_1 * \mathrm{b}_2 * \mathrm{c}_3 + \mathrm{a}_2 * \mathrm{b}_3 * \mathrm{c}_1 + \mathrm{a}_3 * \mathrm{b}_1 * \mathrm{c}_2 - \mathrm{a}_3 * \mathrm{b}_2 * \mathrm{c}_1 - \mathrm{a}_2 * \mathrm{b}_1 * \mathrm{c}_3 - \mathrm{a}_1 * \mathrm{b}_3 * \mathrm{c}_2$$

when it comes to how to find eigenvectors and eigenvalues, they are the numbers λ and vectors v that satisfy the matrix equation:

$\mathrm{A} * \mathrm{v} = \lambda * \mathrm{v}$ or $(\mathrm{A} - \lambda\mathrm{I})\mathrm{v} = 0$ where I is the 3x3 identity matrix.

which results in:

$$det\begin{vmatrix} \mathrm{a}_1 - \lambda & \mathrm{a}_2 & \mathrm{a}_3 \\ \mathrm{b}_1 & \mathrm{b}_2 - \lambda & \mathrm{b}_3 \\ \mathrm{c}_1 & \mathrm{c}_2 & \mathrm{c}_3 - \lambda \end{vmatrix} = 0$$

$$= (\mathrm{a}_1 - \lambda)(\mathrm{b}_2 - \lambda)(\mathrm{c}_3 - \lambda) + \mathrm{a}_2\mathrm{b}_3\mathrm{c}_1 + \mathrm{a}_3\mathrm{b}_1\mathrm{c}_2 - \mathrm{a}_3(\mathrm{b}_2 - \lambda)\mathrm{c}_1 - \mathrm{a}_2\mathrm{b}_1(\mathrm{c}_3 - \lambda) - (\mathrm{a}_1 - \lambda)\mathrm{b}_3\mathrm{c}_2$$

The equation above will result in 3 different values of λ, each corresponding to an eigenvalue. Then, using $(\mathrm{A} - \lambda\mathrm{I})\mathrm{v} = 0$, eigenvectors set(v) can be derived too.

Fortunetely, computers do those calculations for users.

All these complicated calculations can be done much easily if the matrix is symmetric. Therefore next step is the converting $\bar{\bar{M}}$ into a symmetric matrix.

To make $\bar{\bar{M}}$ symmetric: $\overline{x(t)} \longrightarrow \overline{\sqrt{m}x(t)}$

Applying this transformation in matrix equation derived above:

$$\underbrace{\frac{d^2}{dt^2}\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}}_{\overline{a(t)}} = \underbrace{\begin{bmatrix} -\frac{k_1}{m_1} & \frac{k_1}{m_1} & 0 \\ \frac{k_1}{m_2} & \frac{-k_1-k_2}{m_2} & \frac{k_2}{m_2} \\ 0 & \frac{k_2}{m_2} & -\frac{k_2}{m_3} \end{bmatrix}}_{\bar{\bar{M}}} * \underbrace{\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}}_{\overline{x(t)}} \longrightarrow$$

$$\underbrace{\frac{d^2}{dt^2}\begin{bmatrix} \sqrt{m_1}x_1 \\ \sqrt{m_2}x_2 \\ \sqrt{m_3}x_3 \end{bmatrix}}_{\bar{u}} = \underbrace{\begin{bmatrix} -\frac{k_1}{\sqrt{m_1m_1}} & \frac{k_1}{\sqrt{m_1m_2}} & 0 \\ \frac{k_1}{\sqrt{m_2m_1}} & \frac{-k_1-k_2}{\sqrt{m_2m_2}} & \frac{k_2}{\sqrt{m_2m_3}} \\ 0 & \frac{k_2}{\sqrt{m_3m_2}} & -\frac{k_2}{\sqrt{m_3m_3}} \end{bmatrix}}_{-\overline{\overline{DM}}\ \text{(-dynamical matrix)}} \underbrace{\begin{bmatrix} \sqrt{m_1}x_1 \\ \sqrt{m_2}x_2 \\ \sqrt{m_3}x_3 \end{bmatrix}}_{\bar{u}}$$

$$\frac{d^2}{dt^2}\bar{u} = -\overline{\overline{DM}} * \bar{u}$$

The solution of this new matrix equation will exactly the same three equations derived first:

$$m_1\frac{d^2x_1}{dt^2} = (x_2 - x_1)$$

$$m_2\frac{d^2x_2}{dt^2} = k_1(x_1 - x_2) + k_2(x_3 - x_2)$$

$$m_3\frac{d^2x_3}{dt^2} = (x_2 - x_3)$$

$$\frac{d^2}{dt^2}\bar{u} = -\overline{\overline{DM}} * \bar{u}$$

$$\bar{u} = \sum_{i=1}^{n=3} (c_1 \cos(\omega t) + c_2 \sin(\omega t))\, \bar{e_i}$$

$$\frac{d^2}{dt^2}\bar{u} = -\omega^2 * \bar{u} \qquad \textit{and also using } \frac{d^2}{dt^2}\bar{u} = -\overline{\overline{DM}} * \bar{u}$$

$$-\omega^2 \bar{e} = -\overline{\overline{DM}} * \bar{e} \qquad\qquad \overline{\overline{DM}} * \bar{e} = \omega^2 \bar{e}$$

Overall, when the eigenvalues and eigenvectors of the symmetric matrix are found, eigenvalues will correspond to ω^2 values and eigenvectors will correspond to $\bar{e_i}$ values.

To find c_1 and c_2, the initial values of positions and velocities will help.

$$\overline{\sqrt{m}x} = \sum_{i=1}^{n=3} (c_{1i} \cos(\omega t) + c_{2i} \sin(\omega t))\, \bar{e_i}$$

For t=0 $\sin(\omega t)$ will be 0 and $\cos(\omega t)$ will be 1:

$\overline{\sqrt{m}x} = \sum_{i=1}^{n=3} c_{1i}\, \bar{e_i}$ if one multiplies both sides with permission conjugate of e_1, then the result will be $c_1 = e_1' \sqrt{m}x$, where e_1' is the permission conjugate of e_1. When the process is applied for e_2, e_3; c_2, c_3 are found. In general:

$$\bar{c} = \overline{\overline{V}}' \overline{\sqrt{m}x} \quad \textit{where } \overline{\overline{V}}' = \begin{bmatrix} e_1' \\ e_2' \\ e_3' \end{bmatrix}$$

Taking derivative of this equation will give a new equaiton about velocities.

$$\overline{\sqrt{m}v} = \sum_{i=1}^{n=3} (-c_{1i}\, \omega \sin(\omega t) + c_{2i} \omega \cos(\omega t))\, \bar{e_i}$$

For t=0 $\sin(\omega t)$ will be 0 and $\cos(\omega t)$ will be 1:

$\overline{\sqrt{m}v} = \sum_{i=1}^{n=3} c_{2i}\, \omega \bar{e_i}$ since the only unknown here is c_{2i}, it can be calculated using the process above.

Solving the problem analytically gives:

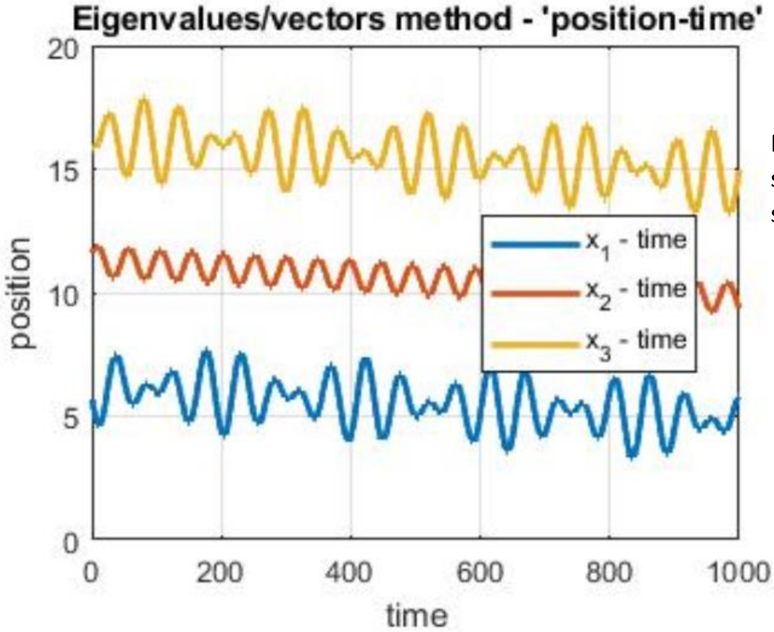

Figure.3: position-time plot of the solution obtained from the analytical solution

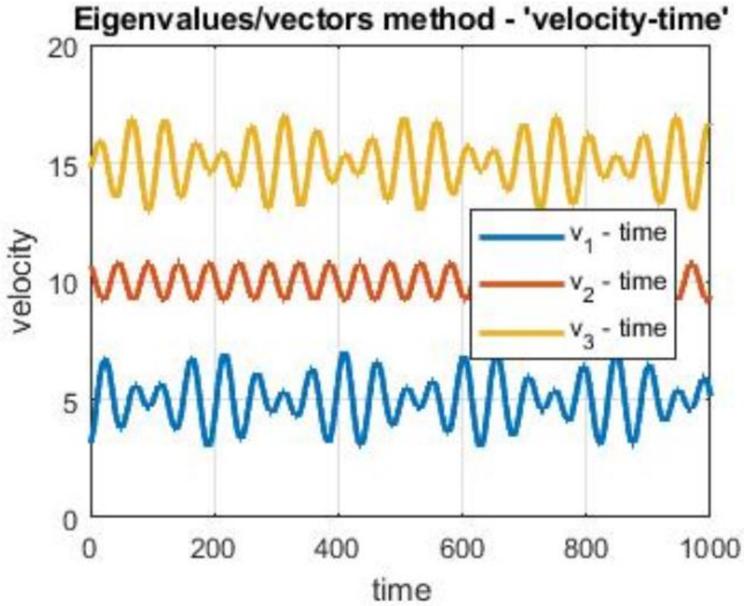

Figure.4: velocity-time plot of the solution obtained from the analytical solution

Here, in figures 3 and 4, the initial positions and initial velocities are randomly taken between 1 and 2.
m(mass constant) and k(spring constant) are used such that they are equal to 1.

In the plots above $x_1(t = 0) = 0.8552, x_2(t = 0) = 1.5376, x_3(t = 0) = 0.8834$,
$v_1(t = 0) = -1.9255$, $v_2(t = 0) = 0.6991$ and $v_3(t = 0) = -0.2460$.

System's behaviour and masses's motion are investigated 1000/0.1 times with 0.1 differences. Therefore 10000 results are shown in the plots above.

2.b) Solution with Verlet Algorithm

$\overline{a(t)} = \overline{\overline{M}} * \overline{x(t)}$

j(t) is the time derivative of acceleration(jerk)

$$x(t + \Delta t) = x(t) + v(t)\Delta t + \frac{a(t)(\Delta t^2)}{2}$$

$$v(t + \Delta t) = v(t) + a(t)\Delta t + \frac{j(t)(\Delta t^2)}{2}$$

$$a(t + \Delta t) = a(t) + j(t)\Delta t \rightarrow j(t) = \frac{a(t+\Delta t) - a(t)}{\Delta t}$$

$$v(t + \Delta t) = v(t) + a(t)\Delta t + \frac{\Delta t}{2}\big(a(t + \Delta t) - a(t)\big)$$

Rewriting velocity equation:

$$v(t + \Delta t) = v(t) + \left(\frac{a(t) + a(t + \Delta t)}{2}\right)\Delta t$$

$$\underbrace{\frac{d^2}{dt^2}\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}}_{\overline{a(t)}} = \underbrace{\begin{bmatrix} -\frac{k_1}{m_1} & \frac{k_1}{m_1} & 0 \\ \frac{k_1}{m_2} & \frac{-k_1 - k_2}{m_2} & \frac{k_2}{m_2} \\ 0 & \frac{k_2}{m_2} & -\frac{k_2}{m_3} \end{bmatrix}}_{\overline{\overline{M}}} * \underbrace{\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}}_{\overline{x(t)}}$$

Even if the initial accelerations are not known at the beginning, they can be derived using the matrix equation above.

$a(t) = M * x(t)$

Overall the required equations to solve the problem are as follows:

$a(t) = M * x(t)$

$$x(t + \Delta t) = x(t) + v(t)\Delta t + \frac{a(t)(\Delta t^2)}{2}$$

$a(t + \Delta t) = M * x(t + \Delta t)$

$$v(t + \Delta t) = v(t) + \left(\frac{a(t) + a(t + \Delta t)}{2}\right)\Delta t$$

Solving the problem using Verlet algorithm gives:

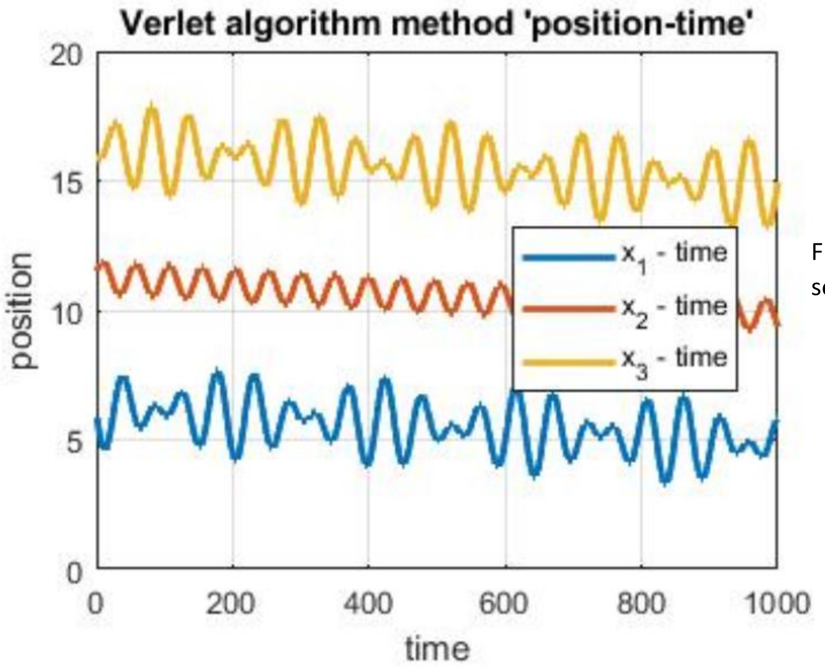

Figure.5: position-time plot of the solution obtained using Verlet algorithm

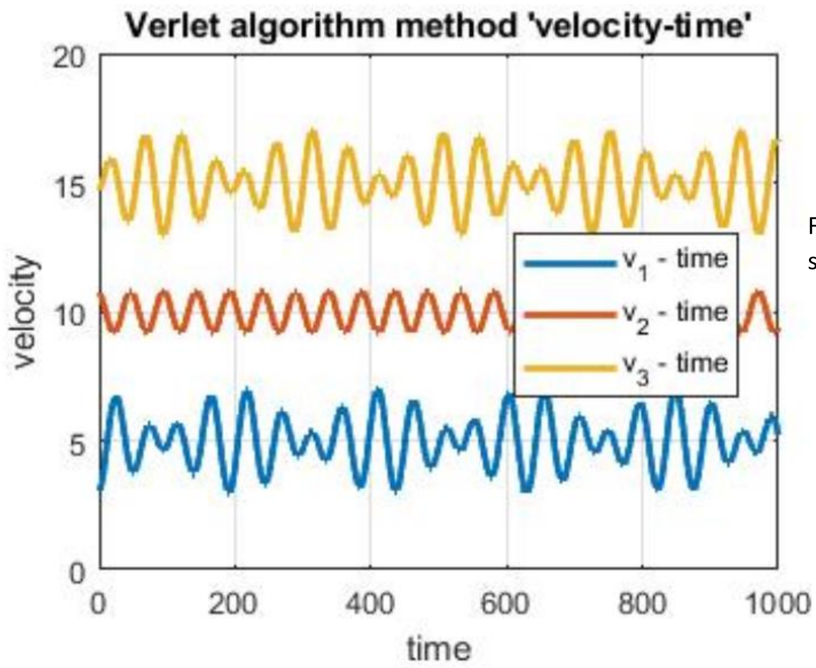

Figure.6: velocity-time plot of the solution obtained using Verlet algorithm

Here, in figures 5 and 6, the initial positions and initial velocities are randomly taken between 1 and 2.
m(mass constant) and k(spring constant) are used such that they are equal to 1.

In the plots above $x_1(t = 0) = 0.8552, x_2(t = 0) = 1.5376, x_3(t = 0) = 0.8834$,
$v_1(t = 0) = -1.9255$, $v_2(t = 0) = 0.6991$ and $v_3(t = 0) = -0.2460$ as in the analytical solution.

System's behaviour and masses's motion are investigated 1000/0.1 times with 0.1 differences. Therefore 10000 results are shown in the plots above. For each time step, time is increased with 0.1 from the previous one, meaning $\Delta t = 0.1$.

3) COMPARISON of TWO METHODS

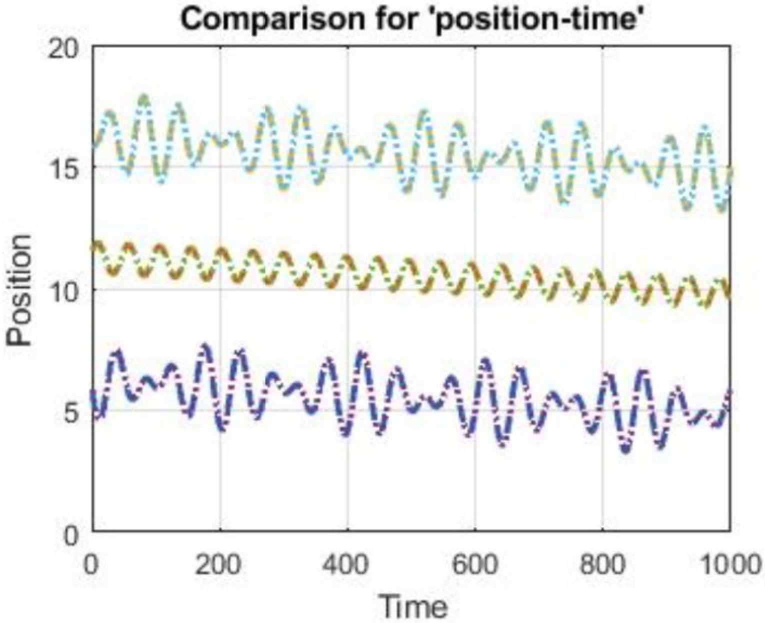

Figure.7: position-time plot comparison of solutions obtained from two different methods

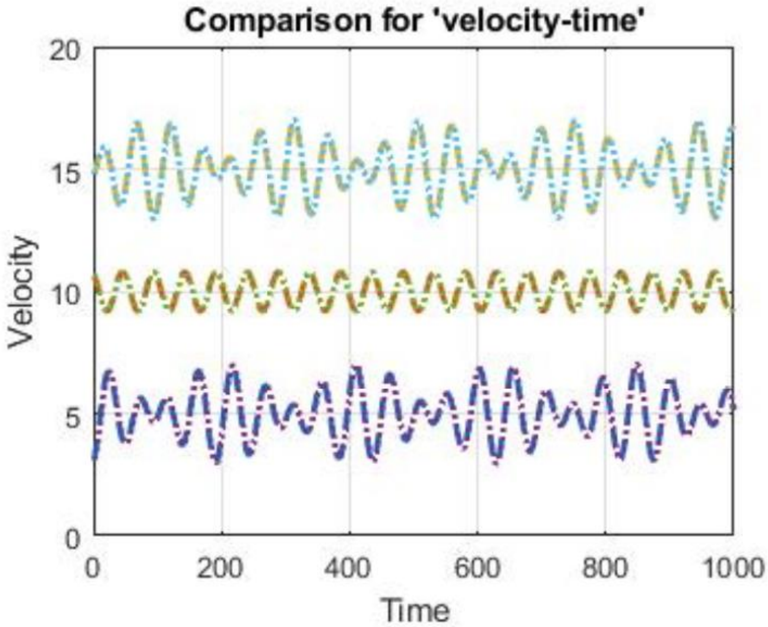

Figure.8: velocity-time plot comparison of solutions obtained from two different methods

In figure.7, the position-time plots of the solution from two different methods discussed previously are compared. Indeed, they are plotted in the same grap to compare. Since they gave the same result, the plots of them coincide with each other.

The velocity-time plots of the solution from two separate methods discussed earlier are compared as shown in Figure.8. Indeed, to compare, they are plotted in the same graph. Since they obtained the same result, their plots coincide with each other as in position-time comparison.

4) CONCLUSION

The multiple mass-spring system is consisting of 3 masses and 2 springs. Since the system is physically symetric, which is consisting of two same masses at the ends and two same type of springs, the middle mass that has 3m mass will have simple harmonic behaviour. When one looks at the position-time and velocity-time plots of the second mass(3m), s/he will see that the behaviour of its motion looks like simple harmonic behavior. Simple harmonic motion, in physics, repetitive movement back and forth through an equilibrium, or central, position, so that the maximum displacement on one side of this position is equal to the maximum displacement on the other side. The time interval of each complete vibration is the same, which is 0.1 in this system. This is caused by the symmetry of the physical system, displacements of first and third masses affect the middle mass symmetrically so that it has a simple harmonic motion. In the position-time plot of the second mass it can have a downward or upward shape but even in this situation, the equilibrium of its harmonic motion takes shape of downward or upward line. Also, since the velocity is the first derivative of the position, the velocity-time plot will have only horizontal simple harmonic motion plot type. The derivative of position at a specific point will have the same magnitude after one grid(0.1), that is why velocity-time plot of the second mass has horizontal simple harmonic motion plot type.

In the analytical solution, the motion of the system is investigated continuously while Verlet algorith solution gives a very close approximation that is so small. Multiplying with Δt is the proof of this idea. Also, at each time step, the previous result is used not to lose information.

With different initial position and velocity values, the plots change. Sometimes, it goes upward sometimes downward, this is the result of initial values.

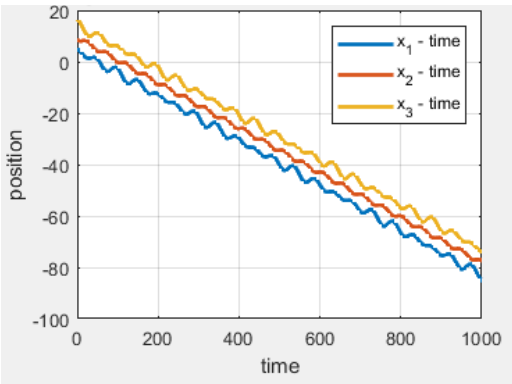

Figure.9: Downward behavior of the motion

The initial positions:
-0.2487 -1.5319 1.2587

The initial velocities:
-0.7006 -1.0151 -0.6291

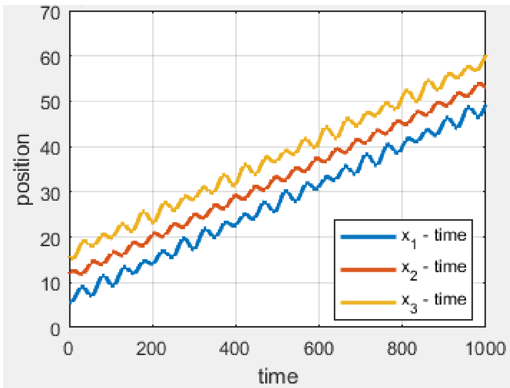

Figure.10: Upward behavior of the motion

The initial positions:
0.6301 1.8037 0.8894

The initial velocities:
-0.3997 1.3275 -1.4626

5) CODE REVIEW

```
clc;
clear;
clear all;
dt = 0.1;
A = 1000;
x = zeros(3,A);
v = zeros(3,A);
a = zeros(3,A);
```

The time grid is defined as 0.1.
The total time is defined as 1000.
And 3x1000 matrices are opened to fill later

```
xmin=-2; xmax=2;
vmin=-2; vmax=2;
random_displacements = xmin + rand(1,3)*(xmax-xmin);
random_velocities = vmin + rand(1,3)*(vmax-vmin);

x(1,1) = random_displacements(1);
x(2,1) = random_displacements(2);
x(3,1) = random_displacements(3);

v(1,1) = random_velocities(1);
v(2,1) = random_velocities(2);
v(3,1) = random_velocities(3);
```

To find a random variable for positions and velocities, the limits for them are decided 1 and 2.

```

disp("These are the initial positions:")
disp(random_displacements)
disp("These are the initial velocities:")
disp(random_velocities)


m = 1;
k = 1;

k1 = k;
k2 = k;
m1 = m;
m2 = 3*m;
m3 = m;
```

Masses and spring constants are randomly chosen.

```
%Analytical Solution using eigenvalues and eigenvectors
sm = sqrt([m1 m2 m3]');

Dynamic_Matrix = [
k1/sqrt(m1*m1) -k1/sqrt(m1*m2) 0
-k1/sqrt(m2*m1) (k1+k2)/sqrt(m2*m2) -k2/sqrt(m2*m3)
0 -k2/sqrt(m3*m2) k2/sqrt(m3*m3)
];

[V,D] = eig(Dynamic_Matrix);

w = sqrt(diag(D));
c_1 = V'*(sm.*x(:,1));
c_2 = (V'*(sm.*v(:,1)))./w;
```

sm is the square root of the masses and it is in some matrix form consisting of 3 different masses. Dynamic matrix same as above of the system is written. Then, eigenvalues and eigenvectors of this system are found. V is eigenvectors, D is eigenvalues. c_1 and c_2 constants are found as discussed before.

```
xc = (V*(c_1.*(cos(w*(1:A)*dt)) + c_2.*(sin(w*(1:A)*dt)))))./sm;
sc = (V*(c_1.*w.*(-sin(w*(1:A)*dt)) + c_2.*w.*(cos(w*(1:A)*dt))))./sm;
```

Position equation(xc) and velocity equation(sc) are written here. sc is the time derivative of xc

```
subplot(2,3,1);
plot(xc'+repelem(5*(1:3),A,1),'LineWidth',2);
legend("x_1 - time","x_2 - time","x_3 - time");
xlabel('time');
ylabel('position');
grid on;
title("Eigenvalues/vectors method - 'position-time'");
```

Position-time plot derived from analytical solution

```
subplot(2,3,4);
plot(sc'+repelem(5*(1:3),A,1),'LineWidth',2);
legend("v_1 - time","v_2 - time","v_3 - time");
xlabel('time');
ylabel('velocity');
grid on;
title("Eigenvalues/vectors method - 'velocity-time'");
```

Velocity-time plot derived from analytical solution

```
%Solution with Verlet algorithm
M = [
-k1/m1 k1/m1 0
k1/m2 -(k1+k2)/m2 k2/m2
0 k2/m3 -k2/m3
];
```

Negative of dynamic matrix is written here.

```
for t = 1 : A - 1
    a(:,t) = M*x(:,t);
    x(:,t+1) = x(:,t) + v(:,t)*dt + a(:,t)*dt^2/2;
    a(:,t+1) = M*x(:,t+1);
    v(:,t+1) = v(:,t) + (a(:,t) + a(:,t+1))/2*dt;
end
```

First column of the acceleration gives initial accelerations of the masses. For example x(:,t+1) means corresponding column gives the x values in column t+1. Doing so up to t = 999 collects all velocity and position data.

```
subplot(2,3,2)
plot(x'+repelem(5*(1:3),A,1),'LineWidth',2);
legend("x_1 - time","x_2 - time","x_3 - time");
xlabel('time');
ylabel('position');
grid on;
title("Verlet algorithm method 'position-time'");
```

Position-time plot derived from Verleth algorithm solution.

```
subplot(2,3,5)
plot(v'+repelem(5*(1:3),A,1),'LineWidth',2);
legend("v_1 - time","v_2 - time","v_3 - time");
xlabel('time');
ylabel('velocity');
grid on;
title("Verlet algorithm method 'velocity-time'");
```

Velocity-time plot derived from Verleth algorithm solution.

```
%Comparison
subplot(2,3,3)
plot(xc'+repelem(5*(1:3),A,1),"--",'LineWidth',2);
hold on;
plot(x'+repelem(5*(1:3),A,1),":",'LineWidth',2);
title("Comparison for 'position-time'")
xlabel('Time');
ylabel('Position');
grid on;
```

Comparison graph of position-time plots of two different methods

```
subplot(2,3,6)
plot(sc'+repelem(5*(1:3),A,1),"--",'LineWidth',2);
hold on;
plot(v'+repelem(5*(1:3),A,1),":",'LineWidth',2);
title("Comparison for 'velocity-time'")
xlabel('Time');
ylabel('Velocity');
grid on;
```

Comparison graph of velocity-time plots of two different methods