

Numerical Methods in Astrophysics

## Project 2

# Ordinary Differential Equations, Fourier Transforms and Spectral Analysis

*Saksham Kaushal*

# Contents

<b>1</b>	<b>Introduction</b>	<b>2</b>
<b>2</b>	<b>Part 1 - Ordinary Differential Equations</b>	<b>3</b>
2.1	Methods . . . . .	4
2.2	Results . . . . .	4
2.3	Discussions . . . . .	5
<b>3</b>	<b>Part 2 - Fourier Transform and Spectral Analysis</b>	<b>9</b>
3.1	Methods . . . . .	9
3.2	Results . . . . .	9
3.3	Discussions . . . . .	9
<b>4</b>	<b>Conclusions</b>	<b>9</b>

# **1 Introduction**

## 2 Part 1 - Ordinary Differential Equations

For stars in disk of the galaxy, equation of motion is given by equation 19 of the lecture notes as,

$$\frac{d^2r(t)}{dt^2} = -\nabla\Phi(r) - 2\Omega_B \times v + \Omega_B^2 r, \quad (1)$$

where  $\Omega_B$  is angular velocity and the gravitational potential,  $\Phi(r)$  in the two dimensions for an elongated bar structure is given by equation 20 of lecture notes as,

$$\Phi(x, y) = \frac{v_0^2}{2} \ln(a^2 + x^2 + \frac{y^2}{q^2}). \quad (2)$$

The gradients of gravitational potential along the two axes are given by partial derivatives of equation (2) with respect to  $x$  and  $y$  as follows,

$$\frac{\partial\Phi}{\partial x} = \frac{v_0^2 x}{a^2 + x^2 + \frac{y^2}{q^2}}, \quad (3)$$

$$\frac{\partial\Phi}{\partial y} = \frac{v_0^2 y}{q^2(a^2 + x^2 + \frac{y^2}{q^2})}. \quad (4)$$

We can substitute equations (3) and (4) in equation (1) to obtain the equation of motion that can be solved in our simulation. Equations along the two axes are given by,

$$a_x = \frac{d^2x}{dt^2} = -\frac{v_0^2 x}{a^2 + x^2 + \frac{y^2}{q^2}} + 2\Omega_B v_y + x\Omega_B^2, \quad (5)$$

$$a_y = \frac{d^2y}{dt^2} = -\frac{v_0^2 y}{q^2(a^2 + x^2 + \frac{y^2}{q^2})} - 2\Omega_B v_x + y\Omega_B^2, \quad (6)$$

where it must be noted that for second term on the right hand side of equation (5), the cross product  $\Omega_B \times v_x = -v_y$  and for similar term in equation (6),  $\Omega_B \times v_y = v_x$ , where  $\Omega_B$  has the direction perpendicularly outwards from the x-y plane.

The value of *Jacobi integral* is conserved as a star moves in a rotating barred potential and therefore, minimum dispersion in its values throughout the simulation indicates better numerical integration. Its value is given by

$$E_j \equiv \frac{\dot{r}^2}{2} + \Phi - \frac{(\Omega_B \times r)^2}{2}. \quad (7)$$

From equation 13 of the lecture notes, the numerical error in each integration time-step can be controlled by adapting the step-size accordingly. If we define,

$$\Delta_0 \equiv \epsilon y_{max}, \quad (8)$$

then  $\Delta < \Delta_0$  should be satisfied for each integration step, where  $\Delta$  is the numerical error of a given step,  $\epsilon$  is the parameter defining the accuracy of integration and  $y_{max}$  is the amplitude of function in the range over which function is being integrated. If this inequality does not hold true, the values need to be adjusted accordingly, and the integration step is repeated with reduced step-size until this inequality is satisfied. However, if this inequality does indeed hold true such that  $\Delta \ll \Delta_0$ , integration step-size can be safely increased to reduce the difference between  $\Delta$  and  $\Delta_0$ . The step-size can be update according to equation 14 of lecture notes, given as,

$$h = h_n \left| \frac{\Delta_0}{\Delta} \right|^m, \quad (9)$$

where  $h$  is the new updated step-size,  $h_n$  is the previous step-size and  $m$  is an exponent usually set to different values when increasing or decreasing the step-size, often in the range [0.2, 0.25].

Parameter	Variable	Value
$\Omega_B$ (Angular velocity)	om	1.0
$v_0$	v0	1.0
$a$	a	0.03
$q$	q	0.8
$t_1$ (Initial time)	x1	0.0
$t_2$ (Final time)	x2	5.0
$x$	ys(1)	0.0
$v_x$	ys(2)	-1.68
$y$	ys(3)	0.087
$v_y$	ys(4)	0.0

**Table 1:** Initial parameters of the simulation, variable names assigned to them in the FORTRAN code and their initial values set for the first simulation.

## 2.1 Methods

In the FORTRAN code *orbit19.f*, the four elements of array **ys** represent the values of  $x$ ,  $v_x$ ,  $y$  and  $v_y$  respectively. Value of  $a$ ,  $q$ ,  $v_0$  and  $\Omega_B$  are represented by variables **a**, **q**, **v0** and **om**. The difference between the variables **x2** and **x1** indicates the time for which the simulation takes place. This is essentially different from time-step of the simulation, that depends on the value of  $\epsilon$ , which parameterizes the accuracy of integration. The four elements of array **dxdy** represent the values of  $v_x$ ,  $a_x$ ,  $v_y$  and  $a_y$ . The values of  $a_x$  and  $a_y$  are calculated using the equations (5) and (6), and encoded in the subroutine **derivs** in the code. The Jacobi integral, given by equation (7) is assigned the variable **ej**.

The following methods were employed for different tasks performed for this part of the project:

1. The missing  $x$  and  $y$  components of the equation of motion for a star in barred potential well are encoded in the subroutine **derivs** of FORTRAN program *orbit19.f* as,

$$\begin{aligned} \text{dxdy}(2) &= -y(1)*v0**2/(a**2+y(1)**2+(y(3)**2/q**2))+2*om*y(4)+y(1)*om**2, \text{ and} \\ \text{dxdy}(4) &= -y(3)*v0**2/(q**2*(a**2+y(1)**2)+y(3)**2)-2*om*y(2)+y(3)*om**2 . \end{aligned}$$

These are the  $a_x$  and  $a_y$  components, given by equations (5) and (6). The  $v_x$  and  $v_y$  components were already assigned to array elements **dxdy**(1) and **dxdy**(3) respectively.

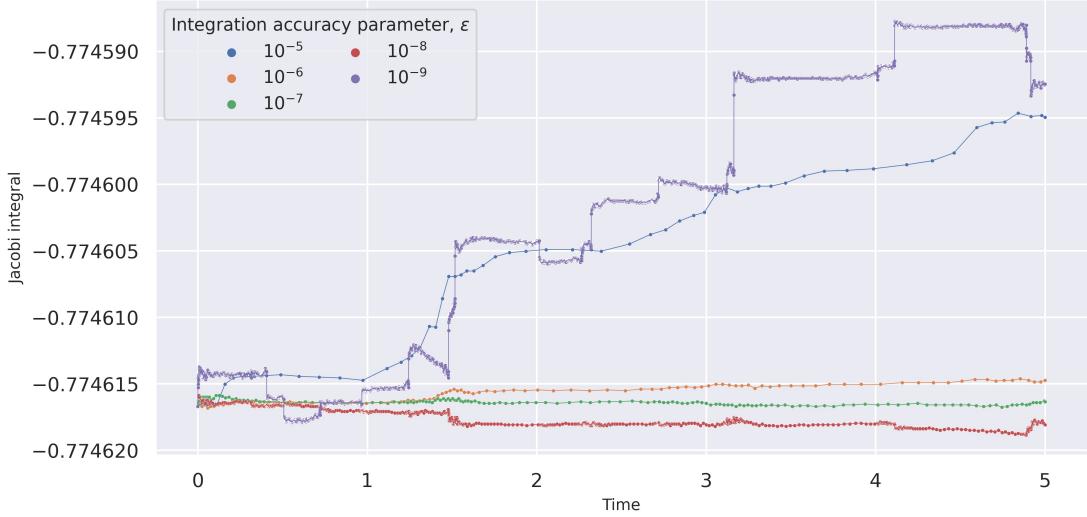
2. Initial parameters of the simulation for this task, along with their values and variable names in the program are mentioned in Table 1.
3. The value of  $\epsilon$  varies the number of time-steps involved, and therefore controls the accuracy of integration. For the initial conditions given in Table 1, simulation is run for values of  $\epsilon$  equal to  $10^{-5}$ ,  $10^{-6}$ ,  $10^{-7}$ ,  $10^{-8}$  and  $10^{-9}$ . The value of Jacobi integral, defined above, is inspected for each of these values of  $\epsilon$  and plotted on a graph using Python and its associated libraries like numpy, pandas, matplotlib and seaborn. All subsequent plotting is performed in a similar manner.
4. The value of variable **x2**, defining the final time of simulation, is changed from initial value of 5.0 to a new value of 20.0, keeping all other parameters unaltered. The trajectories calculated in both the cases are then plotted.
5. The value of  $v_x$  assigned to array element **ys**(2) is changed to values -1.64, -1.635, -1.63, -1.58, -1.15, -0.6, -0.524 and -0.4, and the simulation is run for each of these values to obtain the corresponding trajectories which are plotted for a final time of 20.0.

## 2.2 Results

- For different values of  $\epsilon$ , the variations in values of Jacobi integral are plotted in figure 1 upto a final time of 5.0. The variations are most prominent for highest and lowest values of  $\epsilon$ , and least for intermediate values. Analytically, the Jacobi integral is a conserved quantity for a three-body

problem in a non-inertial frame of reference. Therefore, minimum variation in its value corresponds to most accurate numerical integration, which eventually gives us the most suitable value of  $\epsilon$ .

From the graph in figure 1 and using the values of standard deviations for different values of  $\epsilon$  from table 2, we can conclude that  $\epsilon = 10^{-7}$  is the most suitable value for integration.



**Figure 1:** Variation of Jacobi integral with time for different values of  $\epsilon$ , a parameter that signifies the accuracy of integration.

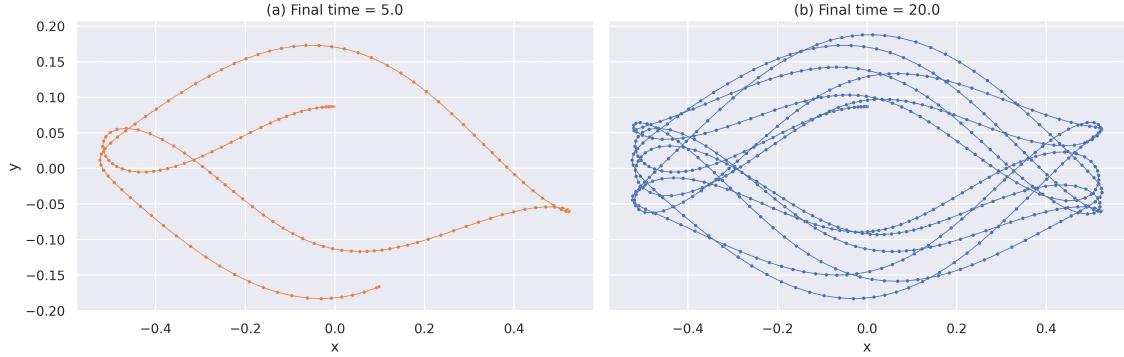
$\epsilon$	$\sigma_{E_j}$
$10^{-5}$	$7.326 \times 10^{-6}$
$10^{-6}$	$6.490 \times 10^{-7}$
$10^{-7}$	$1.887 \times 10^{-7}$
$10^{-8}$	$6.918 \times 10^{-7}$
$10^{-9}$	$1.013 \times 10^{-5}$

**Table 2:** Standard deviation ( $\sigma_{E_j}$ ) in values of Jacobi Integral for different values of integration accuracy parameter ( $\epsilon$ ).

- The trajectories for a star with initial parameters defined in section 2.1, for final times of 5.0 and 20.0 in simulation are plotted in figure 2.
- The calculated trajectories for a final time of 20.0 corresponding to initial values of  $v_x = -1.68, -1.64$  and  $-1.635$  are plotted for comparison in figure 3-(Top). The individual trajectories are plotted separately in figures 2(b) and 3-(Bottom).
- Similar plots for initial values of  $v_x = -1.63, -1.58, -1.15, -0.6, -0.524$  and  $-0.4$  are plotted for comparison in figure 4-(Top) and separately in figure 4-(Bottom).

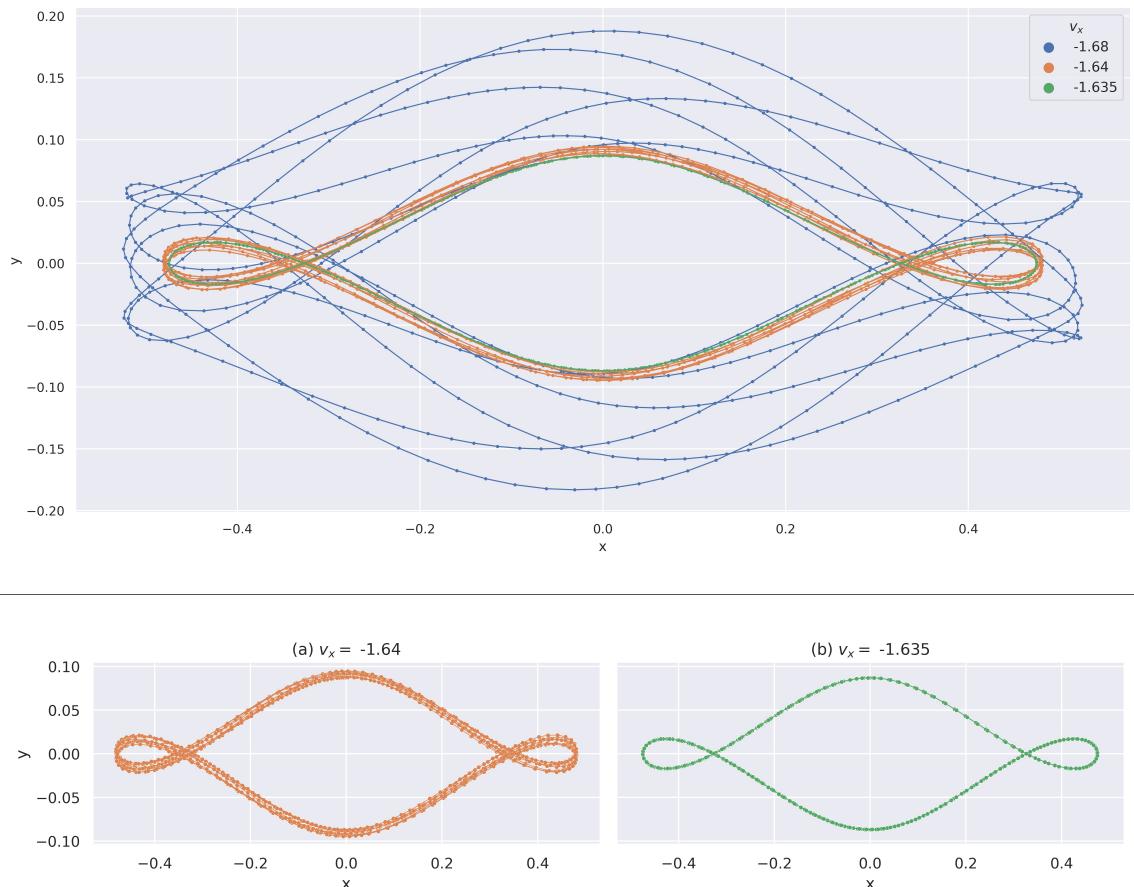
### 2.3 Discussions

- The accuracy of integration increases with reduction in time-step, although it increases execution time. However, below a certain value of integration accuracy parameter,  $\epsilon$ , the accuracy tends to decrease, as seen in figure 1.
- The trajectory followed by the star with  $v_x = -1.635$  in figure 3(b) is a case of *stable periodic orbit*. Such an orbit has a closed geometry and the motion repeats after a fixed interval of time.

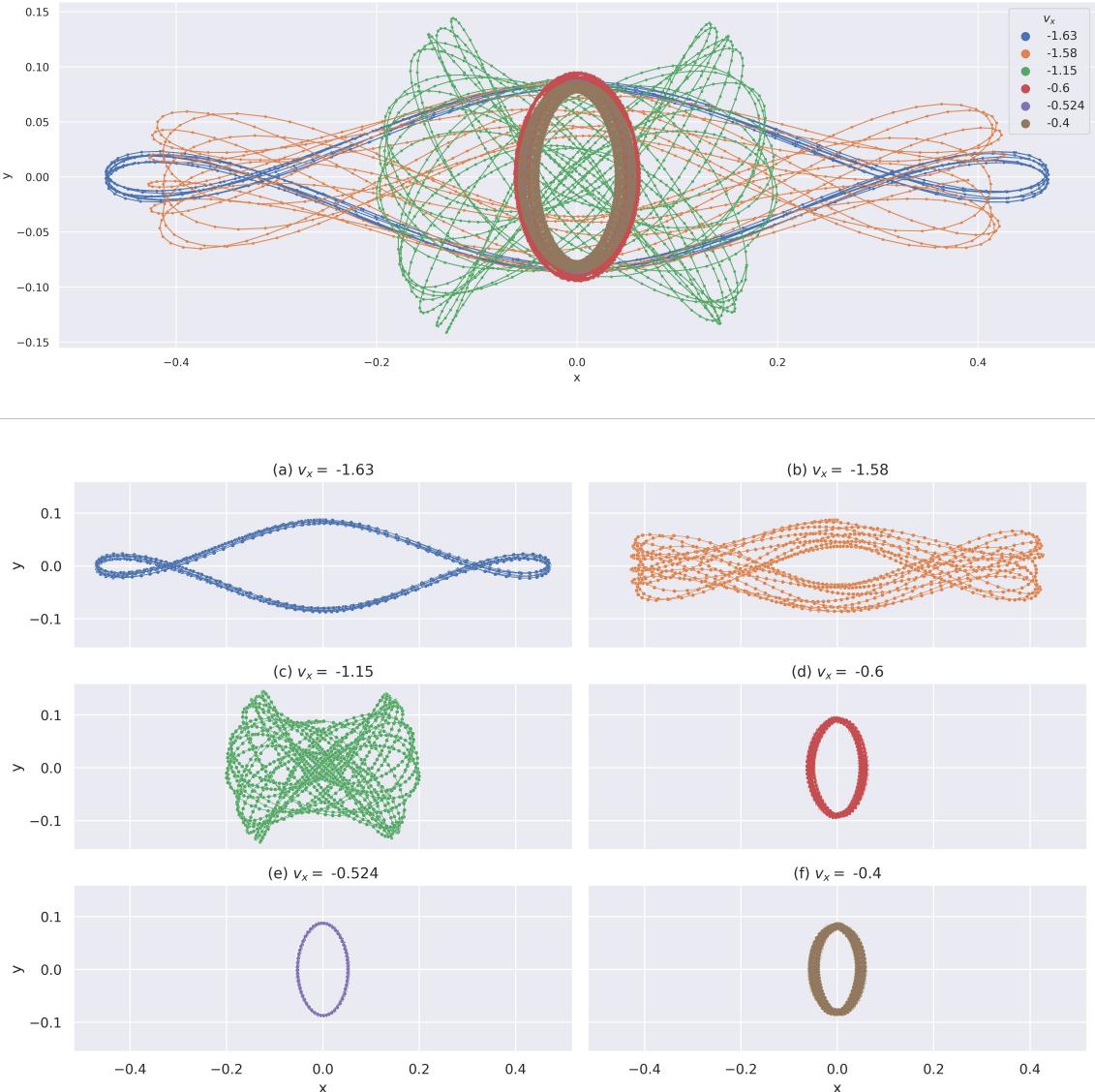


**Figure 2:** Trajectories followed over (a) 5.0 units and (b) 20 units of time by a star with initial parameters,  $(x, y) = (0, 0.087)$  and  $(v_x, v_y) = (-1.68, 0)$ .

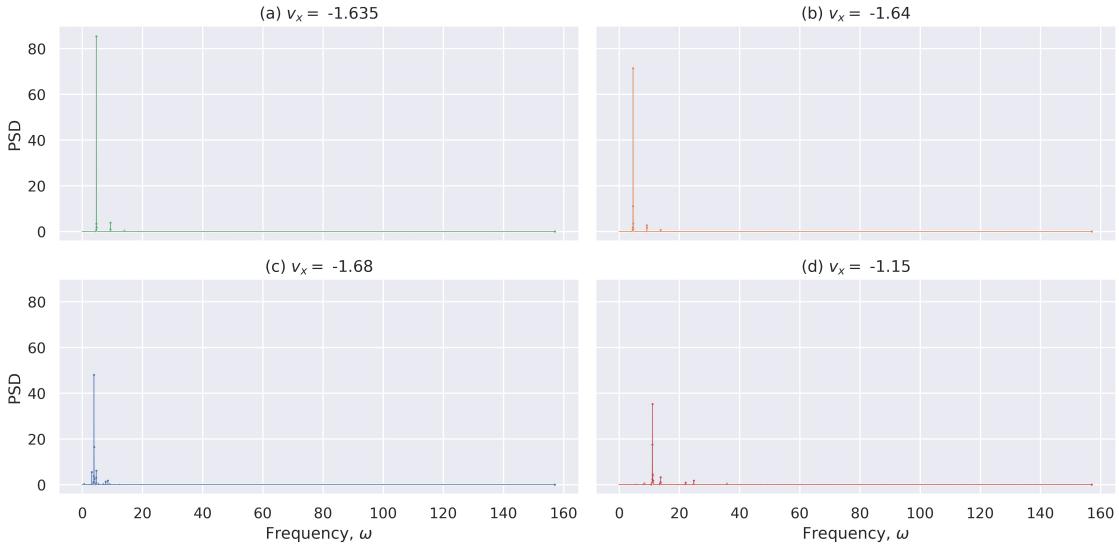
- Slight variation in the parameters from the case of  $v_x = -1.635$ , we get trapped orbits. Examples of such orbits are for  $v_x = -1.64$  and  $-1.68$ . A larger deviation in parameters from the stable periodic orbit leads to a less tightly bound trapped orbit, as in the case of  $v_x = -1.68$ . Such orbits are neither periodic nor closed and tend to stay in the vicinity of the parent stable periodic orbit.
- Similar trend can be noted in figure 4, where the stable periodic orbit occurs at  $v_x = -0.524$  (figure 4(e)). A morphological transition in orbits can be seen as we move from one stable periodic orbit ( $v_x = -1.635$ ) to another ( $v_x = -0.524$ ). Accordingly, the orbit for  $v_x = -1.15$ , for which the initial parameters deviate the most, from the two periodic stable orbits, ends up being the trapped orbit with most loosely bound morphology.
- With decrease in  $v_x$ , we get orbits that are more tightly bound to the center of the potential well. This trend can be seen through the decreasing span of orbits along x-axis as we move to lower values of  $v_x$  in figures 3 and 4.



**Figure 3:** (Top) A comparison plot showing trajectories followed by a star for different initial values of  $v_x$ . (Bottom  $1 \times 2$  grid) Trajectories followed by a star for initial values of  $v_x$  equal to (a)  $-1.64$ , and (b)  $-1.635$  shown separately. Note that the trajectory for  $v_x = -1.68$  is shown individually in figure 2(b).



**Figure 4:** (Top) A comparison plot showing trajectories followed by a star for different initial values of  $v_x$ . (Bottom  $3 \times 2$  grid) Trajectories followed by a star for initial values of  $v_x$  equal to (a)  $-1.63$ , (b)  $-1.58$ , (c)  $-1.15$ , (d)  $-0.6$ , (e)  $-0.524$  and (f)  $-0.4$  shown separately.



**Figure 5:** One-sided power spectral density (in %) vs frequency plots for various initial values of  $v_x$ .

### 3 Part 2 - Fourier Transform and Spectral Analysis

#### 3.1 Methods

In the FORTRAN code *fourier19.f*, the setup for the simulation is performed in a manner similar to that in section 2.1, with  $v_x = -1.635$ . The simulation produces output files containing values of the frequency running number ( $k$ ), the power spectral density as percentage ( $prw$ ) and the logarithm of this value ( $\log prw$ ). The value of  $k$  can be converted to frequency,  $\omega$  using the relation,

$$\omega = \frac{2\pi(k-1)}{KMAXX * step}, \quad (10)$$

where  $KMAXX = 2^{13} = 8192$  and  $step = 0.02$  are predefined in the program.

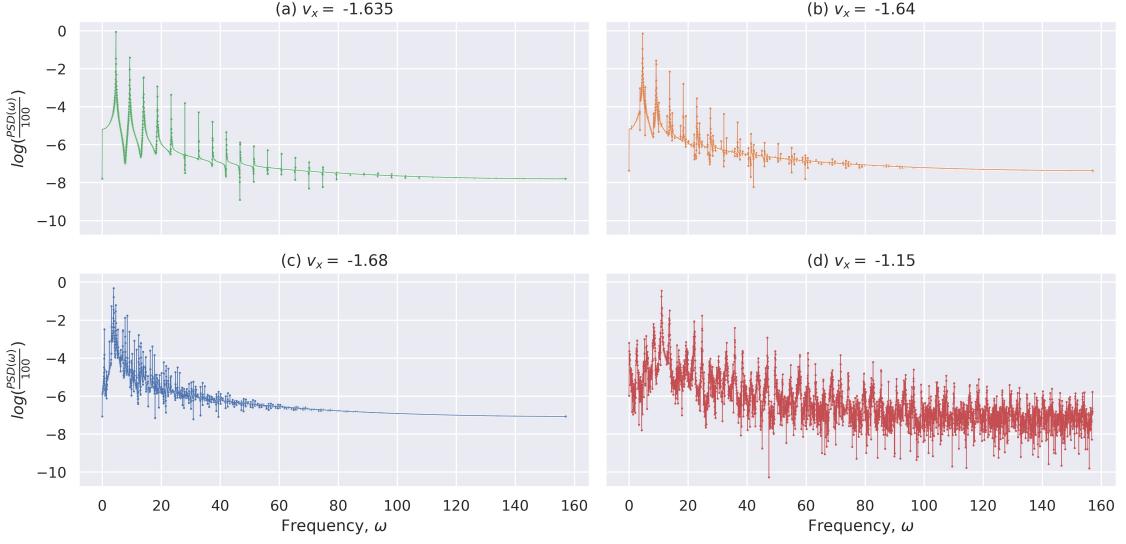
The value of  $v_x$ , assigned to variable `ys(2)`, is set to  $-1.635$ ,  $-1.64$ ,  $-1.68$  and  $-1.15$  in separate runs of the simulation, each producing a different output file to be analyzed.

#### 3.2 Results

- One-sided PSD for  $v_x = -1.635$  is plotted on linear scale in figure 5(a) and on logarithmic scale in figure 6(a). The initial few major peaks are listed in table 3a.
- One-sided PSD for  $v_x = -1.64$  is plotted on linear scale in figure 5(b) and on logarithmic scale in figure 6(b). The initial few major peaks are listed in table 3b.
- One-sided PSD for  $v_x = -1.68$  is plotted on linear scale in figure 5(c) and on logarithmic scale in figure 6(c). The initial few major peaks are listed in table 3c.
- One-sided PSD for  $v_x = -1.15$  is plotted on linear scale in figure 5(d) and on logarithmic scale in figure 6(d).

#### 3.3 Discussions

### 4 Conclusions



**Figure 6:** Logarithm (base 10) of one-sided power spectral density vs frequency plots for various initial values of  $v_x$ .

<b>k</b>	<b><math>\omega</math></b>	<b>k</b>	<b><math>\omega</math></b>
123	4.678	98	3.72
245	9.357	121	4.602
367	14.036	144	5.484
488	18.676	218	8.322
610	23.355	240	9.165
732	28.03	264	10.086
854	32.71	337	12.885
...		360	13.767
<b>(a) <math>v_x = -1.635.</math></b>		383	14.65
		457	17.526
		480	18.37
		503	19.29
		...	
<b>(b) <math>v_x = -1.64.</math></b>		*	20
		*	64
		*	83
		*	102
		*	122
		*	141
		*	166
		*	185
		*	204
		*	223
		*	242
		*	267
		*	281
		*	305
		*	324
		*	344
		...	
<b>(c) <math>v_x = -1.68.</math></b> The starred rows comprise of peaks with a component > 0.1% of total power.			

**Table 3:** Frequency running number ( $k$ ) and corresponding frequency values ( $\omega$ ) for few initial peaks in the data for different initial values of  $v_x$ . Algebraic representation of accompanying dependent frequencies in terms of independent frequencies is also given in the unlabeled rightmost column.