MAS212 Assignment #2: Investigating a dynamical system: Example report

Dr. Sam Dolan (s.dolan@sheffield.ac.uk)

This report would score $\sim$ **85%** *on the mark scheme.*

1 Part 1: Introduction

1(a). In this assignment I have investigated the dynamics of a two-dimensional system governed by a pair of second-order differential equations,

$$\begin{aligned} \ddot{x} &= -x - 2xy, \\ \ddot{y} &= -y - x^2 + y^2. \end{aligned} \quad (1)$$

Here the dots denote derivatives with respect to time, so $\ddot{x} = \frac{d^2x}{dt^2}$, $\ddot{y} = \frac{d^2y}{dt^2}$. The integral I used Python and the `scipy.integrate.odeint()` function to calculate some example trajectories $(x(t), y(t))$, and to investigate the key properties of the system.

1(b) Ball rolling on a hill. Imagine a ball rolling on a surface of height $h(x, y)$, accelerated downhill by the force of gravity, in a uniform gravitational field with $g = 1$. Newton's second law tells us that the acceleration vector $\ddot{\mathbf{x}}$ is proportional to the gradient of the height function[1]. In vector form, $\frac{d^2\mathbf{x}}{dt^2} = -\boldsymbol{\nabla} h$, where $\mathbf{x}(t) = x(t)\boldsymbol{i} + y(t)\boldsymbol{j}$ and $\boldsymbol{\nabla} = \boldsymbol{i}\frac{\partial}{\partial x} + \boldsymbol{j}\frac{\partial}{\partial y}$. In coordinate form, this equation is

$$\ddot{x} = -\frac{\partial h}{\partial x}, \qquad \ddot{y} = -\frac{\partial h}{\partial y}. \quad (2)$$

The height function $h(x, y)$ is defined by

$$h(x, y) = \frac{1}{2}\left(x^2 + y^2\right) + x^2 y - \frac{1}{3}y^3. \quad (3)$$

Its partial derivatives are

$$\frac{\partial h}{\partial x} = \frac{1}{2}.2x + 2xy = x + 2xy, \quad (4a)$$

$$\frac{\partial h}{\partial y} = \frac{1}{2}.2y + x^2 - \frac{1}{3}.3y^2 = y + x^2 - y^2. \quad (4b)$$

Inserting (4) into the equations (2) yields

$$\ddot{x} = -x - 2xy, \qquad \ddot{y} = -y - x^2 + y^2, \quad (5)$$

which matches Eq. (1).

[1] Here we neglect friction and the moment of inertia of the ball

Since the right-hand side of Eq. (1) is the gradient of a scalar function $h(x, y)$, it follows that each trajectory has a conserved energy E, defined by

$$E = \frac{1}{2}\left(\dot{x}^2 + \dot{y}^2\right) + h(x, y). \tag{6}$$

I now consider how E changes along a trajectory. To calculate its time derivative, I apply the chain rule as follows:

$$\begin{aligned}
\dot{E} = \frac{dE}{dt} &= \frac{1}{2}\frac{d}{dt}\left(\dot{x}^2 + \dot{y}^2\right) + \frac{d}{dt}h(x, y), && \text{(7a)}\\
&= \frac{1}{2}\left(2\dot{x}\ddot{x} + 2\dot{y}\ddot{y}\right) + \dot{x}\frac{\partial h}{\partial x} + \dot{y}\frac{\partial h}{\partial y}, && \text{(7b)}\\
&= \dot{x}\left(\ddot{x} + \frac{\partial h}{\partial x}\right) + \left(\ddot{y} + \frac{\partial h}{\partial y}\right), && \text{(7c)}\\
&= 0 \qquad \Rightarrow \quad E = \text{const.} && \text{(7d)}
\end{aligned}$$

The terms in parantheses on the final line are zero, by application of Eq. (2). Hence E is a **constant of motion** along a trajectory.

(c) Stationary points of $h(x, y)$. At a stationary point, the partial derivatives of the height function are zero, that is,

$$x(1 + 2y) = 0, \qquad y(1 - y) + x^2 = 0. \tag{8}$$

The first equation implies that either (i) $x = 0$, or (ii) $y = -1/2$. The second equation implies for case (i) that $y = 0$ or $y = 1$, and for case (ii) that $x^2 = 3/4 \Rightarrow x = \pm\frac{\sqrt{3}}{4}$. Hence there are four stationary points, at the (x, y) coordinates

$$(0, 0), \quad (0, 1), \quad \left(-\tfrac{\sqrt{3}}{2}, -\tfrac{1}{2}\right), \quad \left(+\tfrac{\sqrt{3}}{2}, -\tfrac{1}{2}\right). \tag{9}$$

To classify these points, I first calculated the Jacobian matrix (also known as the Hessian matrix) of second partial derivatives,

$$J = \begin{pmatrix} \frac{\partial^2 h}{\partial x^2} & \frac{\partial^2 h}{\partial x \partial y} \\ \frac{\partial^2 h}{\partial y \partial x} & \frac{\partial^2 h}{\partial y^2} \end{pmatrix} = \begin{pmatrix} 1 + 2y & 2x \\ 2x & 1 - 2y \end{pmatrix}. \tag{10}$$

For a function of two variables, the stationary points can be classified using the trace and determinant of the Jacobian matrix. The trace and determinant of the Jacobian are $\mathrm{Tr(J)} = 2$ and $\det(J) = 1 - 4(x^2 + y^2)$. At $(0, 0)$ the trace is positive and the determinant is positive, so $(0, 0)$ is a **local minimum**. At $(0, 1)$ the trace is positive and the determinant is negative ($\det J = -3$), so this stationary point is a **saddle point**. Similarly, at $\left(\pm\frac{\sqrt{3}}{2}, -\frac{1}{2}\right)$, the trace is positive and the determinant is negative, so these are also saddle points. To the north, south-west and south-east are valleys, where $h < 0$, and to the south, north-west and north-east are valleys, where $h > 1/6$.

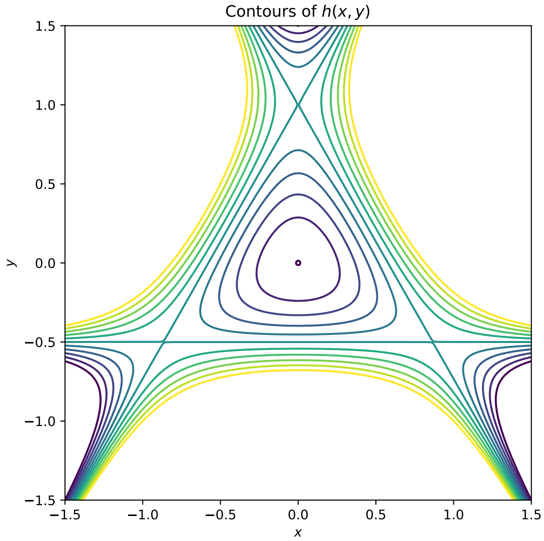

Figure 1: A contour plot of the height function $h(x, y)$ defined in Eq. (3).

Part 2: Bound trajectories

(a) A contour plot. Figure 1 shows a contour plot of the height function $h(x, y)$ defined in Eq. (3). The minimum at $(0, 0)$, which appears as a small circle on the plot, is at height $h = 0$. The 'special' contour that joins the three saddle points in a triangle is at height $h(0, 1) = \frac{1}{2} - \frac{1}{3} = \frac{1}{6}$. A central triangular basin is enclosed by this contour. Inside the central basin, $0 \leq h \leq 1/6$. The height function has a three-fold symmetry; it is invariant under a rotation around the origin of 120° and 240°.

(b) First-order form. Before I can use Python to calculate trajectories, I first need to convert the differential equations to first-order form. This can be done by introducing a pair of new variables $p_x = \dot{x}$ and $p_y = \dot{y}$, so that (1) may be written as a set of four first-order equations,

$$\begin{aligned} \dot{x} &= p_x \ , & \dot{p}_x &= -x - 2xy \ , \\ \dot{y} &= p_y \ , & \dot{p}_y &= -y - x^2 + y^2 \ , \end{aligned} \tag{11}$$

Here I have use $\dot{p}_x = \ddot{x}$ and $\dot{p}_y = \ddot{y}$, which follows from the definitions above.

(c) Example trajectories. Figure 2 shows three example trajectories in the central basin. In each case I took an initial condition with $y = 0$ and $p_x = 0$, and I chose initial values of x and p_y such that the energy E defined in (6) is less than $E_c = 1/6$. By choosing a starting point inside the basin, and by choosing an energy $E < E_c$, the trajectory will remain 'bound' within the special contour with $h = E_c = 1/6$.

The first plot in Fig. 2 shows a low-energy trajectory that orbits around the centre of the basin at $(0, 0)$. It appears as a deformed ellipse that is precessing in a clockwise sense in a regular manner. This trajectory shows a three-fold symmetry. The second plot shows a trajectory with a two-fold symmetry about $x = 0$ that is nearly – but not quite – periodic.

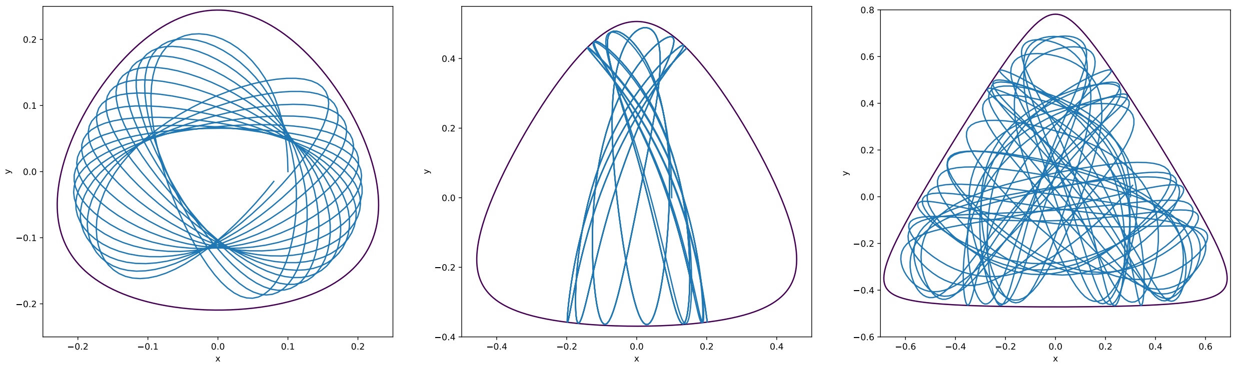

Figure 2: Three example trajectories. In each case, the initial condition was $x(0) = x_0$, $y(0) = 0$, $p_x(0) = 0$ and $p_y(0) = v_0$, where x_0 and v_0 are positive constants. The trajectories were found by solving the differential equations (11) numerically in Python using the `scipy.integrate.odeint()` function, and the plots were created using `matplotlib.pyplot.plot()`.

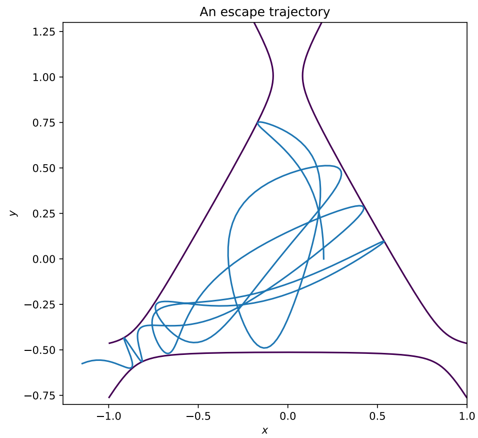

Figure 3: An example of a trajectory with energy $E > 1/6$ that escapes from the central basin via the upper exit and, having done so, rolls down the hill rapidly.

In other words, it is nearly-but-not-quite closed. The third plot shows a 'chaotic' trajectory that explores are closed region of (x, y) space without any obvious regularity.

Part 3: Escape trajectories

If the energy E exceeds $E_c = 1/6$, then the trajectory may escape from the central basin – but this is not necessarily guaranteed. For energies above-but-close-to E_c, we may examine which of the three exits the trajectory escapes through. Predicting which exit the trajectory will escape from is not straightforward, and can depend sensitively on the initial conditions, and on the accuracy of the numerical method. To get accurate results, I found it necessary to reduce the tolerance of the numerical integrator, that is, to set the `atol` and `rtol` optional arguments of `solve_ivp()` to lower values than their defaults.

(a) An escape trajectory. Figure 3 shows an example of an escape trajectory, and the

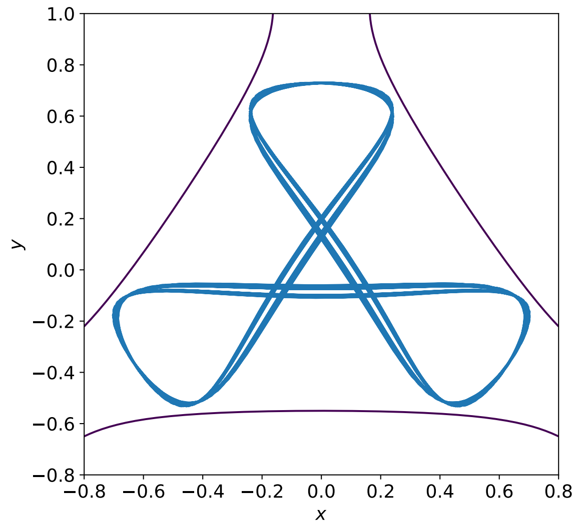

Figure 4: Showing a trajectory with $E > E_c$ that does not leave the central basin.

contour $h(x, y) = E$ which the trajectory may not cross. After a period of moving within the central basin, it eventually departs from the basin via the south-west exit.

I found this trajectory using `scipy.integrate.solve_ivp()` in Python. This function, unlike `odeint()`, allows one to specify an event condition to terminate the solver once a criterion is met. I used the criterion $x^2 + y^2 > r^2$ (with $r = 1.3$ typically) to determine when the trajectory had left the system.

(b) *(Challenging)* Students should choose **only one** of the two tasks below:

1. An eternal orbit. I set out to look for orbits with $E > E_c$ which nevertheless remain inside the basin without escaping to infinity. An example of such an orbit in shown in Fig. 4; this trajectory start on the symmetry axis $x = 0$ with $y = -0.06$, $p_x = 0.64$ and $p_y = 0$. The plot shows a trajectory with a 3-fold symmetry which is nearly periodic. I have verified (numerically) that it does not escape in the domain $0 \leq t \leq 500$. The orbit appears to be stable under small perturbations of the initial conditions, and it seems likely that there is a periodic orbit "close by".

2. Exit by initial conditions. *(Hardest)*. Figure 5 shows the fate of 50×50 trajectories that start with $y(0) = 0, p_y(0) = 0$ and $x(0) = x_0$ and $p_y(0) = v_0$ in the ranges 0 to 0.8. The colour indicates whether the trajectory was still in the central basin after $t = 100$ (black), or escaped via the top, bottom-left or bottom-right corners. The quarter-circle corresponds to orbits with $E < E_c$. For $E > E_c$ there is some evidence of fractal-like structure in the boundaries of the **exit basins** (the coloured regions) on space of initial conditions. This deserves some further investigation.

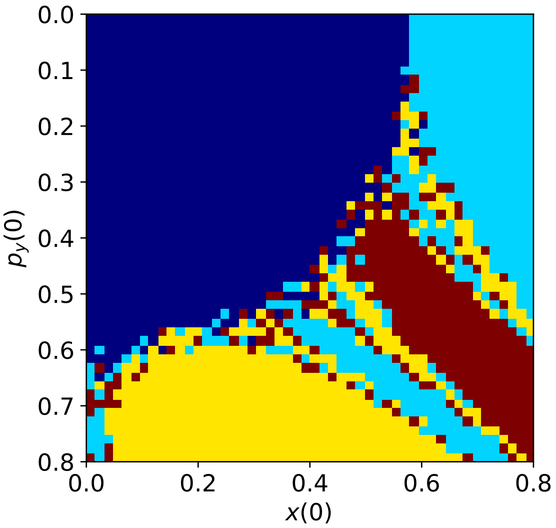

Figure 5: Showing the fate of a trajectory with initial conditions $x(0) = x_0$, $p_y(0) = v_0$ and $y(0) = 0$, $p_x(0) = 0$ after $t = 100$. *Key:* Dark blue = remains in the central basin. Light blue / yellow / red: escaped via exit 1, 2, 3.

Conclusion

I have investigated the key features of the trajectories of a 2D dynamical system. This system is governed by a potential function $h(x, y)$ with a three-fold symmetry, and three saddle points on the same contour which define a basin region. I found that the orbits can be bound or unbounded; regular or chaotic in nature; and can exhibit a two-fold or three-fold symmetry. I found a nearly-periodic orbit above the threshold for escape, and I presented some numerical evidence that escape basins on the space of initial conditions have fractal, rather than regular, boundaries.