Notes on numerical solution of differential equations

Some definitions, for those who don't know:

- A "differential equation" is any equation that relates a thing to its derivatives. For instance, Newton's second law of motion for a mass bouncing on a string can be written $-kx = m\frac{\partial^2 x}{\partial t^2}$, where I've put in Hooke's law for the force.
- A "system of differential equations" is a group of equations that relate a bunch of things to their derivatives.
- A "first-order differential equation" is one involving only first derivatives.
- A "second-order differential equation" involves second derivatives.

A great many physics phenomena are described by (systems of) differential equations, so we need a way to solve them numerically. In our class, we're going to focus on time-variation problems of the following form:

Given the initial state of some system y, and the differential equation(s) that tell us how y changes with time, figure out what the system y looks like for all future times – that is, find $y(t)$.

For instance, we are very interested in computing the motions of objects, the evolution of electromagnetic fields, the propagation of heat, and so on, and these techniques give us a way to study that.

In general, these situations often take the form of a set of coupled second-order differential equations. However, such a situation can be reduced to a set of first-order DE's. So, in these notes I will build up the general case in three steps:

- A single first-order DE: the cooling cup of tea
- A single second-order DE: the swinging pendulum
- Many second-order DE's: celestial motion

1 Euler's method for a single first-order DE

Throughout I will use time as the independent variable, since this is usually the case we care about. However, everything works the same if some other coordinate is the independent variable.

A first-order differential equation has the form

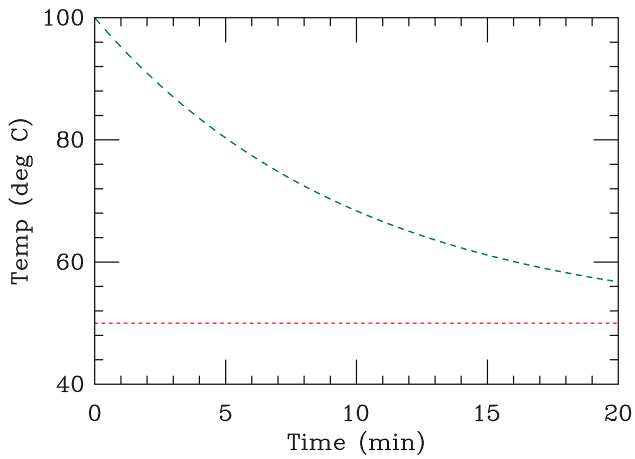
$$\frac{\partial y}{\partial t} = f(y, t) \tag{1}$$

where y is a generic dependent variable and $f(y, t)$ gives its rate of change over time as a function of its value (and, sometimes, the time). Note the notation: f here is not an arbitrary function, but the *specific function that tells you how fast y changes.* A specific example is Newton's law of cooling

$$\frac{\partial T}{\partial t} = f(t) = -k(T - T_a) \tag{2}$$

which is the first problem you'll be solving. In words, it just says that **an object cools at a rate proportional to the difference in temperature between it and the room around it.**

This is the same behavior as you get from a charging/discharging capacitor (in electronics), incidentally. It behaves like this:



The behavior of a cooling thermos of tea (in a very hot room!) The green plot gives the temperature of the tea; the red line represents the ambient temperature of 50° C (perhaps you're in Arizona in June, or the Physics Clinic in April!)

We need to figure out how to make a computer calculate this. The solution, as with most things in computational physics, involves *thinking about how you would get a bunch of middle-schoolers with calculators to do it*, then simply telling the computer to do that.

Let's make a table. For illustration, I'll use $k = 0.5\text{min}^{-1}$, with $T_a = 50$ and $T_0 = 100$ – that is, we're starting with tea at 100 degrees that is cooling down in a room that is 50 degrees.

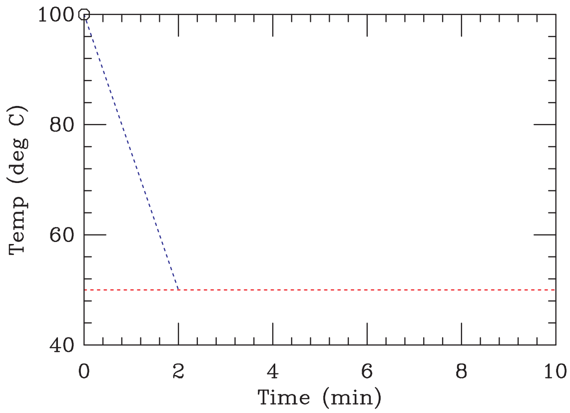
Time (min)	Temp (C)	Slope: $\frac{\partial T}{\partial t}$
0	100	
1		
2		
3		
4		
5		

Our job is to fill in all the missing temperatures, using the only information we have, the differential equation

$$\frac{\partial T}{\partial t} = -k(T - T_a).$$

This equation tells us the slope of the curve, given the current temperature T; it comes out to $(100-50)\times 0.5$ = 25 degrees per minute.

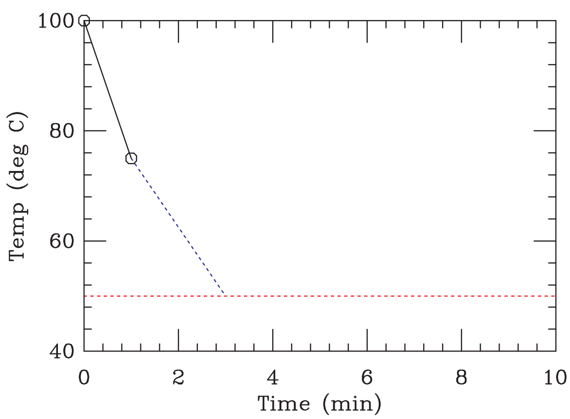
I can show what I know now both in tabular form and as a graph:



Time (min)	Temp (C)	Slope: $\frac{\partial T}{\partial t}$
0	100	-25
1		
2		
3		
4		
5		

Here the black circle at $(t = 0, T = 100)$ indicates that I know the temperature is 100 degrees at $t = 0$; the blue dotted line shows my knowledge of the slope at this point.

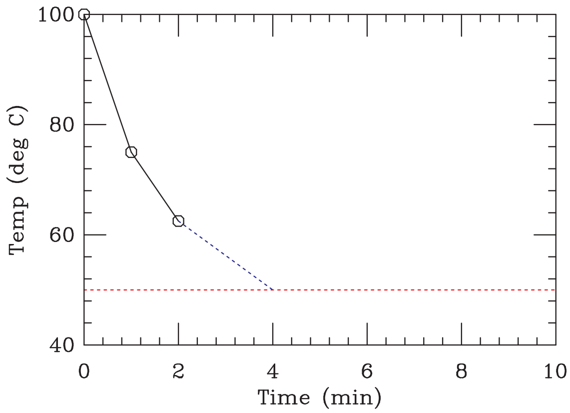
Well, if I need to estimate the temperature after 1 minute, and I know the tea is cooling at 25 degrees per minute, then I should just guess that the $T(1) = 100 - 25 = 75$ degrees. Graphically, what I'm doing is evaluating the slope at $(t = 0, T = 100)$ and following that down to $t = 1$.



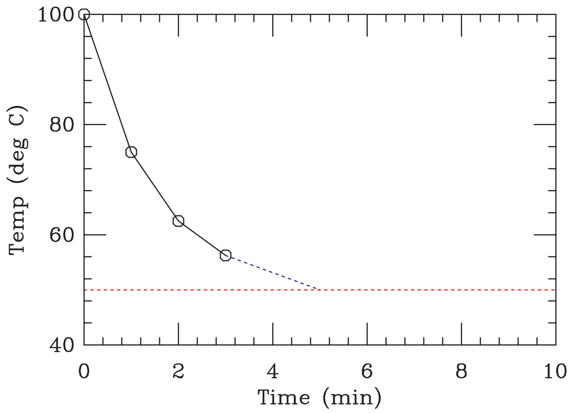
Time (min)	Temp (C)	Slope: $\frac{\partial T}{\partial t}$
0	100	-25
1	75	-12.5
2		
3		
4		
5		

Now I have some new information on both my table and graph: the point $(t = 1, T = 75)$, and the ability to calculate that the slope here is -12.5 degrees per minute. That's reflected in both the chart and the graph above.

You should be able to see how this goes at this point: we figure out the current slope from the current temperature, and then use the current temperature and that slope to estimate the next temperature. Taking more steps we get:

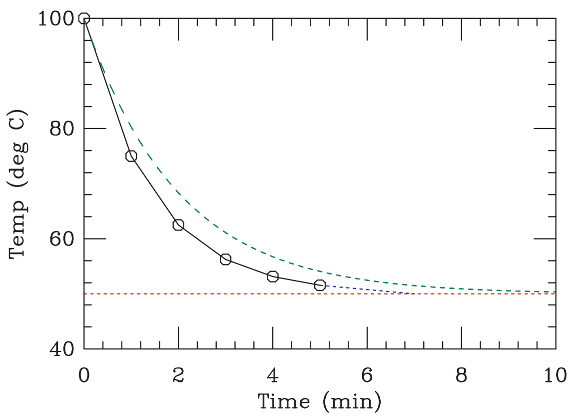


Time (min)	Temp (C)	Slope: $\frac{\partial T}{\partial t}$
0	100	-25
1	75	-12.5
2	62.5	-6.25
3		
4		
5		



Time (min)	Temp (C)	Slope: $\frac{\partial T}{\partial t}$
0	100	-25
1	75	-12.5
2	62.5	-6.25
3	56.25	-3.125
4		
5		

And taking two more steps:

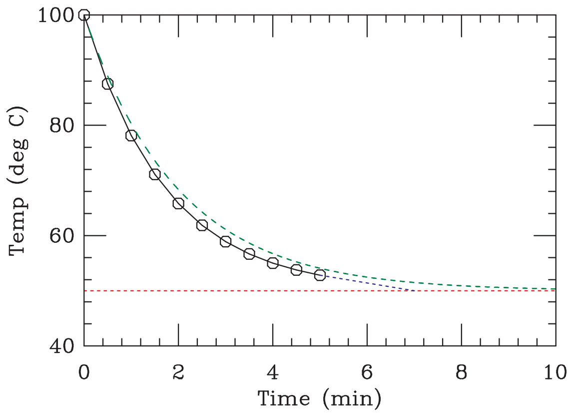


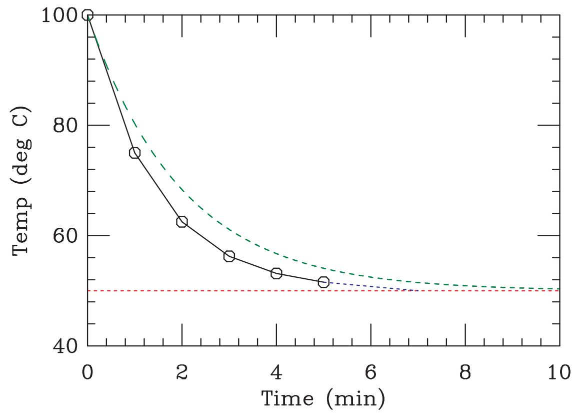
Time (min)	Temp (C)	Slope: $\frac{\partial T}{\partial t}$
0	100	-25
1	75	-12.5
2	62.5	-6.25
3	56.25	-3.125
4	53.125	-1.5625
5	51.5625	0.78125

This last figure also shows the "exact" solution. What has gone wrong? You can probably guess: during that first minute, we used a cooling rate of 25 degrees per minute. However, as the tea cools, its cooling rate goes down: it wasn't 25 deg/min for that *entire* time, only at the very start. You probably have two ideas for what I should do:

- Use smaller steps
- Do something akin to the midpoint rule

We'll talk about the equivalent to the midpoint rule later. Smaller steps help, as you might expect. Here I'm using a step of 0.5 minute on the left, and 0.5 minute on the right:





You can see by eye that the error here is smaller.

The Euler method consists of doing the simplest thing you can think of: you approximate the value of the function at the end of each timestep by its value at the beginning, plus the rate of change times the length. In symbols,

$$y(t+h) = y(t) + hf\left[y\left(t\right)\right]; \tag{3}$$

that is, the value after a timestep is equal to the value before the step, plus the length of the step times the rate of change. Since we need a value of y to evaluate the rate of change, we use the value at the beginning of the interval $y(t)$. For instance, for Newton's law of cooling, $f(T) = -k(T - T_a)$, and Euler's algorithm is

$$T(t+h) = T(t) - hk(T - T_a) \tag{4}$$

Since this uses a left-hand-rule-type approach (estimate the slope during the interval using its value at the beginning), This is a "first-order" algorithm.

You've now learned how to solve first-order differential equations (here "first-order" means that the DE includes only first derivatives, as opposed to "first-order" in the previous sentence – damn mathematicians!).

We now have two possible next steps:

- Learn how to do something like the midpoint rule, to get better accuracy for a given timestep
- Learn how to handle second-order differential equations, like Newton's second law

You can learn about those in any order; I'll introduce the midpoint-like algorithm first.

2 The RK2 algorithm

The RK2 algorithm is analogous to the midpoint rule. Here, we would like to use the function's slope *halfway through the interval*, rather than at the beginning. We'd like to do something like this:

$$y(t+h) = y(t) + hf\left[y\left(t+\frac{h}{2}\right)\right]; \tag{5}$$

The only trouble is that we don't know $y\left(t+\frac{h}{2}\right)$ – in the Newton's law of cooling example, the temperature halfway through the interval. However, we can estimate it using the Euler method:

$$y(t+\frac{h}{2}) = y(t) + \frac{h}{2}f\left[y\left(t\right)\right]; \tag{6}$$

This gives the following procedure for RK2:

1. Take an Euler-method half step to estimate the value of the variable y at $t + h/2$
2. Use that value $y(t + h/2)$ to compute the slope over the whole interval, and take a full step using that slope

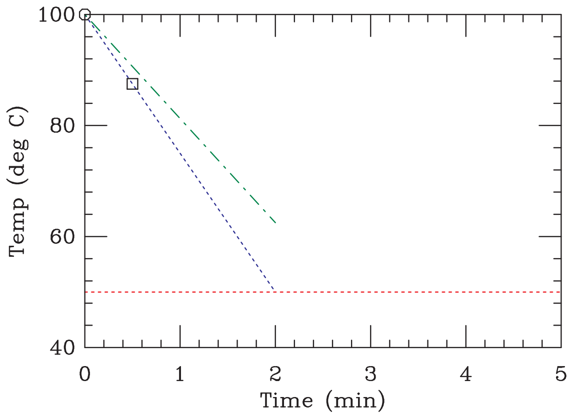
In symbols, we have

$$y\left(t+\frac{h}{2}\right) = y(t) + \frac{h}{2}f\left[y\left(t\right)\right] \tag{7}$$

$$y(t+h) = y(t) + hf\left[y\left(t+\frac{h}{2}\right)\right] \tag{8}$$

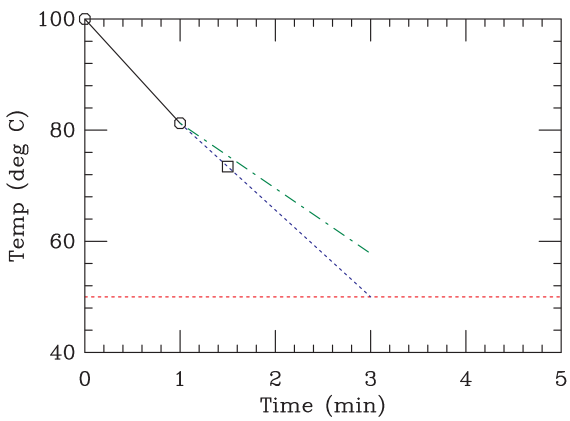
Let me now illustrate this in pictures, using $dt = 1$ minute. Our procedure is:

- Use the temperature we know at (at $(t = 0, T = 100)$ to compute the slope there
- Follow that slope out for a time interval equal to $dt/2$ (half a minute) to get an estimate of the temperature at $t = 0.5$ minute, the midpoint of the interval
- Use this estimate of the temperature at 0.5 minute to compute an estimate of the slope during the whole interval. This uses the same logic as the midpoint rule: "the average slope during the interval" (the thing we want) is well-estimated by "the slope at the midpoint of the interval".
- Based on that estimate of the slope, starting with *the original temperature value at the beginning of the interval*, figure out the temperature at the end.
- Repeat until golden brown around the edges.

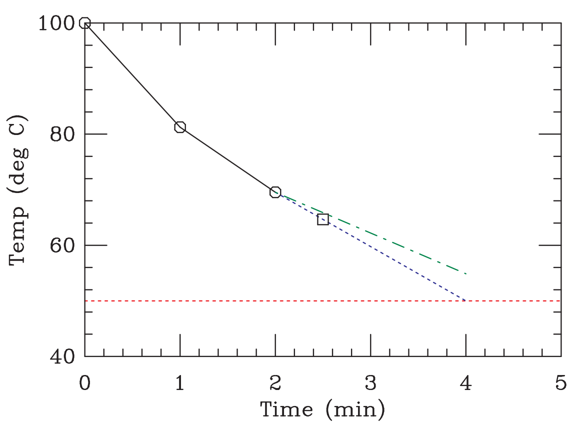


- The circle is the point we started with
- The blue dotted line is the estimate of the slope at that temperature, $T(0)$
- The square is the estimate of the temperature halfway through the step, T_{half}
- The green dashed line is the revised estimate of the slope during the interval, based on the value of T_{half}.

Now, this is where we are:

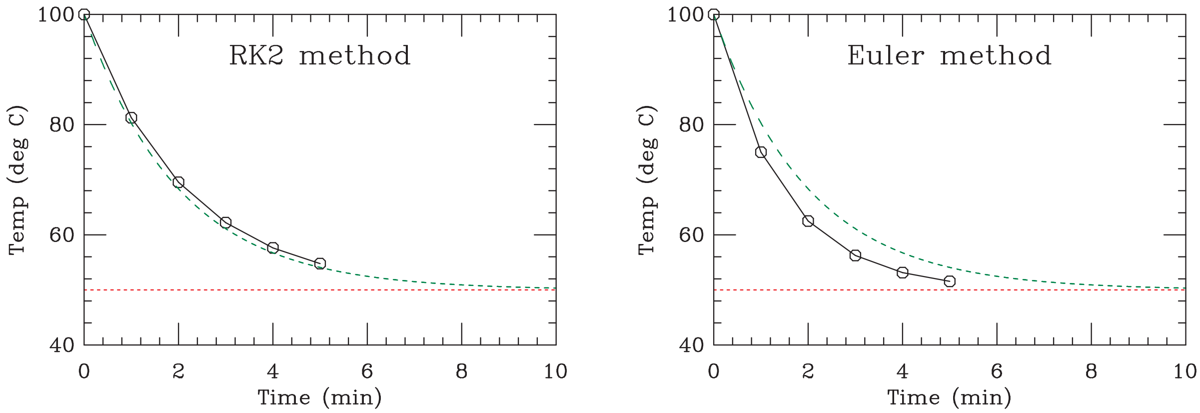


- Important: notice that the first step is taken starting from the *beginning* value – the point $(0, 100)$ – rather than the halfway estimate. The halfway estimate is used *only* to estimate the slope used for the full step, and then forgotten.
- The black line represents the first full step (which becomes part of our solution). The blue dotted line is the slope estimate at $T(1)$, used to calculate an estimate for the next value of T_{half}; the green dashed line is the estimate of the slope based on *that* temperature, which we will use for the third step...



- ... and so on.

Comparing this RK2 method to Euler's method shows how much more accurate it is:



3 Euler's method for a single second-order DE

A second-order differential equation can be reduced to two first-order DE's. In general, given the second-order differential equation

$$\frac{\partial^2 y}{\partial t^2} = f(y) \tag{9}$$

we can separate it into two *first-order DE's*:

$$\frac{\partial y}{\partial t} = \dot{y} \tag{10}$$

$$\frac{\partial \dot{y}}{\partial t} = f(y, \dot{y}) \tag{11}$$

at the cost of introducing an extra *dynamical variable* $\dot{y}$. Dynamical variables are simply those variables that the simulation is evolving through time – for instance, positions and velocities.

Then you do a straightforward Euler update for each:

$$y(t+h) = y(t) + h\dot{y}(t) \tag{12}$$

$$\dot{y}(t+h) = \dot{y}(t) + hf\left[y(t), \dot{y}(t)\right] \tag{13}$$

As an example, for the pendulum we have

$$\frac{\partial \theta}{\partial t} = \omega \tag{14}$$

$$\frac{\partial \omega}{\partial t} = -g/L \sin\theta \tag{15}$$

and the Euler update looks like

$$\theta_{new} = \theta_{old} + h\omega_{old} \tag{16}$$

$$\omega_{new} = \omega_{old} + h\frac{g}{L}\sin\theta_{old} \tag{17}$$

$$\tag{18}$$

where I have used some new notation that should be easy to figure out.

4 RK2 for a single second-order DE

The approach here is just the same. The only trick is that you must take *both* half-steps together, since you need both y and $\dot{y}$ evaluated at the halfway point to take the full step. The algorithm can be written

$$y\left(t + h/2\right) = y(t) + \frac{h}{2}\dot{y}(t) \tag{19}$$

$$\dot{y}\left(t + h/2\right) = \dot{y}(t) + \frac{h}{2}f\left[y(t), \dot{y}(t)\right] \tag{20}$$

$$y(t+h) = y(t) + h\dot{y}(t + h/2) \tag{21}$$

$$\dot{y}(t+h) = \dot{y}(t) + hf\left[y\left(t + h/2\right), \dot{y}\left(t + h/2\right)\right] \tag{22}$$

For the pendulum in particular, our y is the angle θ and $\dot{y}$ is the angular velocity ω. Thus we have:

$$\theta_{1/2} = \theta_{old} + \frac{h}{2}\omega_{old} \tag{23}$$

$$\omega_{1/2} = \omega_{old} - \frac{h}{2}\frac{g}{L}\sin\theta_{old} \tag{24}$$

$$\theta_{new} = \theta_{old} + h\omega_{1/2} \tag{25}$$

$$\omega_{new} = \omega_{old} - h\frac{g}{L}\sin\theta_{1/2} \tag{26}$$

$$\tag{27}$$

5 Euler's method for a system of many DE's

In general, a large system of DE's with N dynamical variables (which may be a combination of velocities and positions, as in the previous example, or may be something else) may be written

$$\frac{\partial y_1}{\partial t} = f_1(y_1, y_2, y_3, y_4...)\frac{\partial y_2}{\partial t} = f_2(y_1, y_2, y_3, y_4...)\frac{\partial y_3}{\partial t} = f_3(y_1, y_2, y_3, y_4...)\frac{\partial y_4}{\partial t} = f_4(y_1, y_2, y_3, y_4...), \tag{28}$$

where each rate of change f_i depends on *all* of the dynamical variables y_i. In compact vector form, we can represent the vector of dynamical variables as $\vec{y}$ and their rates of change as $\vec{f}(\vec{y})$. This is a vector function (it gives us a rate of change for each y_i) which has a vector parameter (it needs the value of all y_i's). Then the collection of DE's can be written simply as

$$\frac{\partial \vec{y}}{\partial t} = \vec{f}(\vec{y}) \tag{29}$$

and the Euler algorithm becomes

$$\vec{y}(t+h) = \vec{y}(t) + h\vec{f}\left[\vec{y}\,(t)\right]; \tag{30}$$

Note that all I did here is put vector signs over everything. It is easiest to understand how these algorithms work if you think of your collection of dynamical variables as a single object, and perform any operations on all of them at once. For HW4, for instance, you have four of them.

6 RK2 for a system of many DE's

You should already be able to guess how this goes: this is the same as RK2 for a single variable, except now we think about updating an entire *set* (which I've written as $\vec{y}$) of dynamical variables. Some of these may be position variables (in which case their rate of change is a corresponding velocity variable), and some may be velocity variables (in which case their rates of change are given by Newton's second law), but it doesn't matter; the procedure is just "do a half-step for everything to get the values at the midpoint, and then use those values to do a full step".

We can write this in vector form as follows:

$$\vec{y}(t+\frac{h}{2}) = \vec{y}(t) + \frac{h}{2}\vec{f}\left[\vec{y}\,(t)\right] \tag{31}$$

$$\vec{y}(t+h) = \vec{y}(t) + h\vec{f}\left[\vec{y}\left(t+\frac{h}{2}\right)\right] \tag{32}$$

where I have just scribbled vector signs above all the y's and h's that appeared in the one-equation RK2. It really is the same – you just need to update a set of variables, rather than just one.

7 Orbits around a fixed star

Since planets orbit in a plane, we can restrict ourselves to the $x-y$ plane. This means we need four dynamical variables: $x, y, v_x, andv_y$. Let's put the Sun at the origin; then our variables tell us the position and velocity of the planet. We are going to have the set of DE's

$$
\begin{aligned}
\dot{x} &= v_x && (33)\\
\dot{y} &= v_y && (34)\\
\dot{v}_x &= a_x && (35)\\
\dot{v}_y &= a_y && (36)\\
& && (37)
\end{aligned}
$$

Now we just need to figure out what a_x and a_y are.

Newton's law of gravity tells us

$$\vec{F}_g = (-\hat{r})\frac{GMm}{r^2} \tag{38}$$

where $\hat{r}$ is a unit vector pointing from the Sun to the planet, M is the mass of the Sun, and m is the mass of the planet. To compute the acceleration due to this force, we can use Newton's second law, which gives us

$$m\vec{a} = (-\hat{r})\frac{GMm}{r^2} \rightarrow \vec{a} = (-\hat{r})\frac{GM}{r^2} \tag{39}$$

Since we will be working in Cartesian coordinates, we need to find the xi and $y-$ components of $\vec{a}$. This is done readily with the $\hat{r}-$trick. Note that

$$\hat{r} = \frac{\vec{r}}{|r|}. \tag{40}$$

This makes sense: $\hat{r}$ is a vector in the direction of r, but with length 1, so we just take $\vec{r}$ and divide it by its length $|r|$. Going forward I will drop the absolute value signs: $r \equiv |r| = \sqrt{x^2+y^2}$. Now we convert everything into x and y components:

$$
\begin{aligned}
\vec{r}_x &= x && (41)\\
\vec{r}_y &= y && (42)\\
\hat{r}_x &= \frac{x}{r} && (43)\\
\hat{r}_y &= \frac{y}{r} && (44)\\
r &= \sqrt{x^2+y^2} && (45)
\end{aligned}
$$

We can substitute this into Newton's law of gravity to get the x and y coordinates of the acceleration, which are what we want:

$$
\begin{aligned}
\vec{a} =& (-\hat{r})\frac{GM}{r^2} && (46)\\
\vec{a} =& (-\vec{r})\frac{GM}{r^3} && (47)\\
a_x =& \frac{-GMx}{r^3} && (48)\\
a_y =& \frac{-GMy}{r^3} && (49)
\end{aligned}
$$

where $r = \sqrt{x^2 + y^2}$. This gives us what we need: four DE's that give the rates of change of our four dynamical variables:

$$
\begin{aligned}
\dot{x} =& v_x && (50)\\
\dot{y} =& v_y && (51)\\
\dot{v}_x =& \frac{-GMx}{(x^2 + y^2)^{\frac{3}{2}}} && (52)\\
\dot{v}_y =& \frac{-GMy}{(x^2 + y^2)^{\frac{3}{2}}} && (53)\\
& && (54)
\end{aligned}
$$