Application 9.4
The Rayleigh and van der Pol Equations

The British mathematical physicist Lord Rayleigh (John William Strutt, 1842–1919) introduced an equation of the form

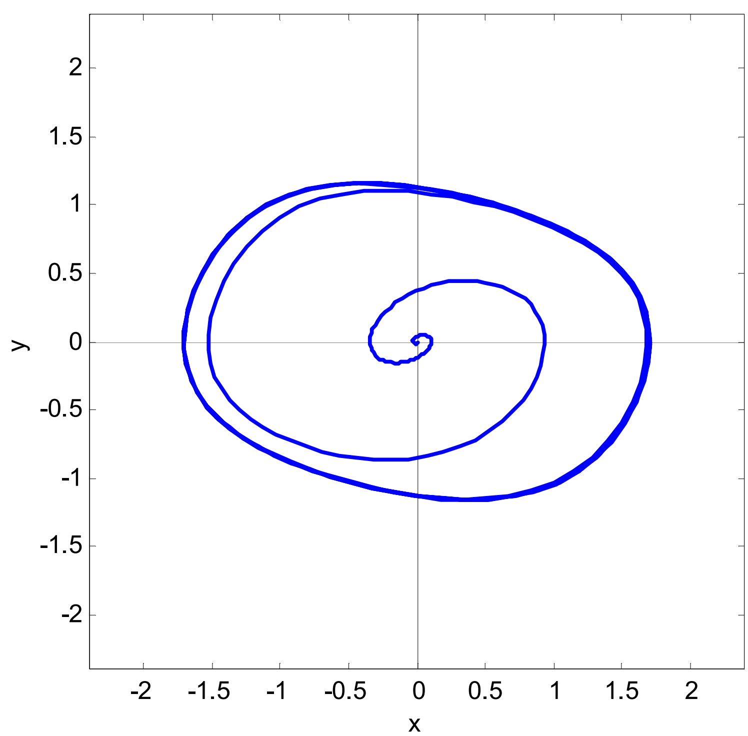
$$mx'' + kx = ax' - b(x')^3 \tag{1}$$

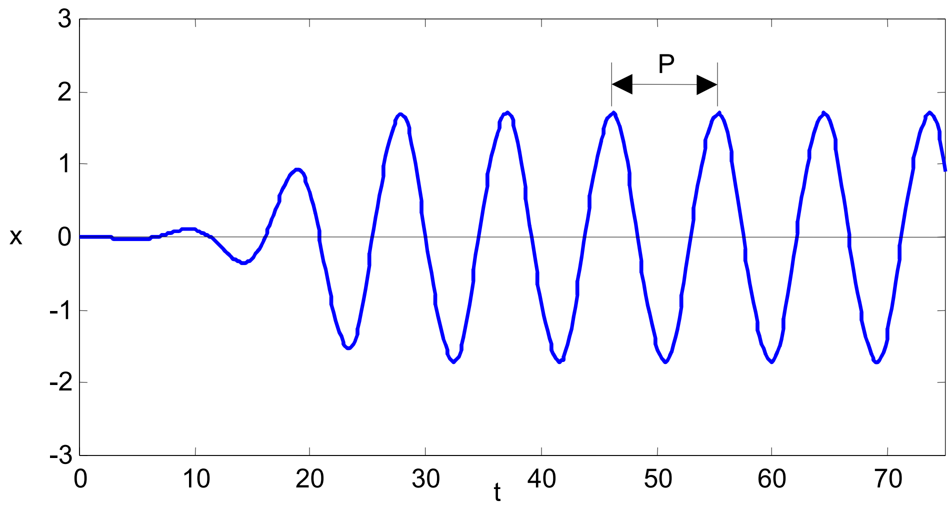
(with non-linear velocity damping) to model the oscillations of a clarinet reed. With $y = x'$ we get the autonomous system

$$x' = y, \qquad y' = \frac{1}{m}\left(-kx + ay - by^3\right). \tag{2}$$

A typical phase portrait for this system is shown in Fig. 9.4.15 in the text, where we see outward and inward spiral trajectories converging to a "limit cycle" solution that corresponds to periodic oscillations of the reed.

The figure below shows an *xy*-trajectory with parameter values $m = 2$ and $k = a = b = 1$. The period $P \approx 9.2$ (and hence the frequency) of these oscillations can be measured (approximately) as indicated on the *tx*-solution curve plotted at the top of the next page. This period of oscillation depends only on the parameters *m, k, a,* and *b* in Eq. (1), and is independent of the initial conditions (why?).





Choose your own parameters m, k, a, and b (perhaps the least four nonzero digits in your student ID number), and use an available ODE plotting utility to plot both xy-trajectories and tx-solution curves for the resulting Rayleigh equation. Change *one* of your parameters (perhaps m) to see how the amplitude and frequency of the resulting periodic oscillations are altered.

Van der Pol's Equation

Figure 9.4.17 in the text shows a simple RLC circuit in which the usual (passive) resistance R has been replaced by an active element (such as a vacuum tube or semiconductor) across which the voltage drop V is given by a known function $f(I)$ of the current I. Of course, $V = f(I) = RI$ for a standard resistor. If we substitute $f(I)$ for RI in the familiar RLC-circuit equation $LI'' + RI' + I/C = 0$ of Section 3.7, we get the second-order equation

$$LI'' + f'(I)\,I' + I/C = 0. \tag{3}$$

In a 1924 study of oscillator circuits in early commercial radios, Balthasar van der Pol (1889–1959) assumed the voltage drop to be given by a nonlinear function of the form $f(I) = bI^3 - aI$, which with Eq. (3) becomes

$$LI'' + (3bI^2 - a)I' + I/C \;=\; 0. \tag{4}$$

This equation is closely related to Rayleigh's equation, and has phase portraits resembling the one shown for Rayleigh's equation. Indeed, differentiation of the second equation in (2) and the substitution $x' = y$ yield the equation

$$my'' + (3by^2 - ay)\,y' + ky \;=\; 0 \tag{5}$$

which has the same form as Eq. (4).

If we denote by τ the time variable in Eq. (4) and make the substitutions $I = px,\ \ t = \tau/\sqrt{LC}$, the result is

$$\frac{d^2x}{dt^2}+\left(3bp^2x^2-a\right)\sqrt{\frac{C}{L}}\,\frac{dx}{dt}+x = 0.$$

With $p=\sqrt{a/(3b)}$ and $\mu = a\sqrt{C/L}$ this gives the standard form

$$x'' + \mu(x^2-1)x' + x = 0 \tag{6}$$

of *van der Pol's equation.*

For every nonnegative value of the parameter μ, the solution of van der Pol's equation with $x(0) = 2,\ x'(0) = 0$ is periodic, and the corresponding phase plane trajectory is a limit cycle to which the other trajectories converge (as illustrated in Fig. 9.4.15 in the text). It will be instructive for you to solve van der Pol's equation numerically and plot this periodic trajectory for a selection of values from $\mu = 0$ to $\mu = 1000$ or more. With $\mu = 0$ it is a circle of radius 2 (why?). Figure 9.4.18 in the text shows the periodic trajectory with $\mu = 1$, and Fig. 9.4.19 shows the corresponding $x(t)$ and $y(t)$ solution curves.

You might also plot other trajectories that are "attracted" from within and from without by the limit cycle (like those shown in Fig. 6.4.18). The origin looks like a spiral point in Fig. 9.4.18. Indeed, show that (0,0) is a spiral source for van der Pol's equation if $0 < \mu < 2$, but is a nodal source if $\mu \ge 2$.

In the sections that follow we illustrate the *Maple*, *Mathematica*, and MATLAB techniques needed for these investigations.

Using *Maple*

To plot a solution curve of the Rayleigh system in (2) we need only load the **DEtools** package and use the **DEplot** function. For instance, if we first specify the parameter values

```
m := 2:    k := 1:
a := 1:    b := 1:
```

and define the differential equations

```
deq1 :=    diff(x(t),t) = y:
deq2 :=  m*diff(y(t),t) = -k*x + a*y - b*y^3:
```

then the commands

```
with(DEtools):
DEplot([deq1,deq2], [x,y],
        t=0..75, x=-3..3, y=-3..3,
        {[x(0)=0.01, y(0)=0]}, stepsize=0.1,
        linecolor=blue, arrows=none);
```

plot the outward spiraling xy-trajectory with initial conditions $x(0) = 0.01$, $y(0) = 0$, on the interval $0 \le t \le 75$ with step size $h = 0.1$. Next, the command

```
DEplot([deq1,deq2], [x,y], t=0..75,
        {[x(0)=0.01, y(0)=0]}, stepsize=0.1,
        scene = [t,x], linecolor=blue, arrows=none);
```

plots the corresponding tx-solution curve, on which the approximate period of oscillation can be measured.

Using *Mathematica*

To plot a solution curve for the Rayleigh system in (2) we need only specify the parameter values

```
m = 2;     k = 1;
a = 1;     b = 1;
```

define the differential equations

```
deq1 =      x'[t] ==  y[t];
deq2 =   m*y'[t] == -x[t] + a*y[t] - b*y[t]^3;
```

and then use **`NDSolve`** to integrate numerically. For instance, the command

```
soln = NDSolve[ {deq1,deq2, x[0]==0.01, y[0]==0},
                {x[t],y[t]}, {t,0,75} ]
```

yields an approximate solution on the interval $0 \le t \le 75$ satisfying the initial conditions $x(0) = 0.01$, $y(0) = 0$. Then the command

```
ParametricPlot[
     Evaluate[{x[t],y[t]} /. soln], {t,0,75}]
```

plots the corresponding xy-trajectory, and the command

```
Plot[Evaluate[ x[t] /. soln ], {t,0,75}]
```

plots the corresponding tx-solution curve, on which the approximate period of oscillation of the prey population can be measured.

Using MATLAB

To plot solution curves for the van der Pol system

$$x' = y, \qquad y' = -\mu(x^2 - 1)\,y - x \tag{7}$$

corresponding to Eq. (6), we first define the system by means of the m-file

```
function  yp = vanderpol(t,y)
% vanderpol.m
yp = y;
x = y(1);
y = y(2);
mu = 2;
yp(1) = y;
yp(2) = -mu*(x^2 - 1)*y - x;
```

Note that the value $\mu = 2$ is specified in this function file. Then we use **`ode45`** to integrate numerically. For instance, the command

```
[t,y] = ode45('vanderpol', [0:0.04:20], [2; 0]);
```

yields an approximate solution on the interval $0 \le t \le 20$ satisfying the initial conditions $x(0) = 2,\ y(0) = 0$. Then the command

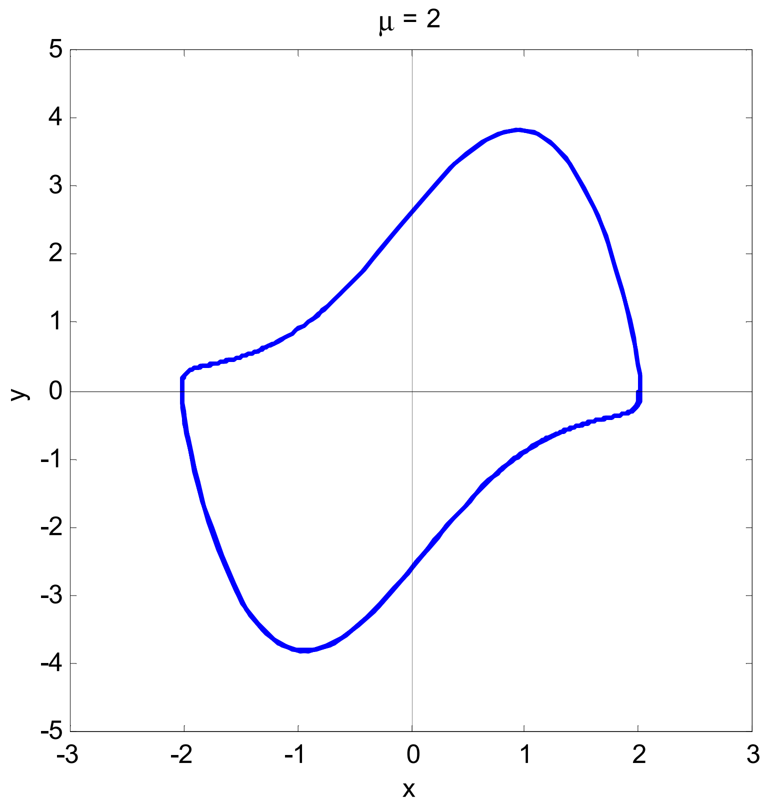
```
plot(y(:,1), y(:,2))
```

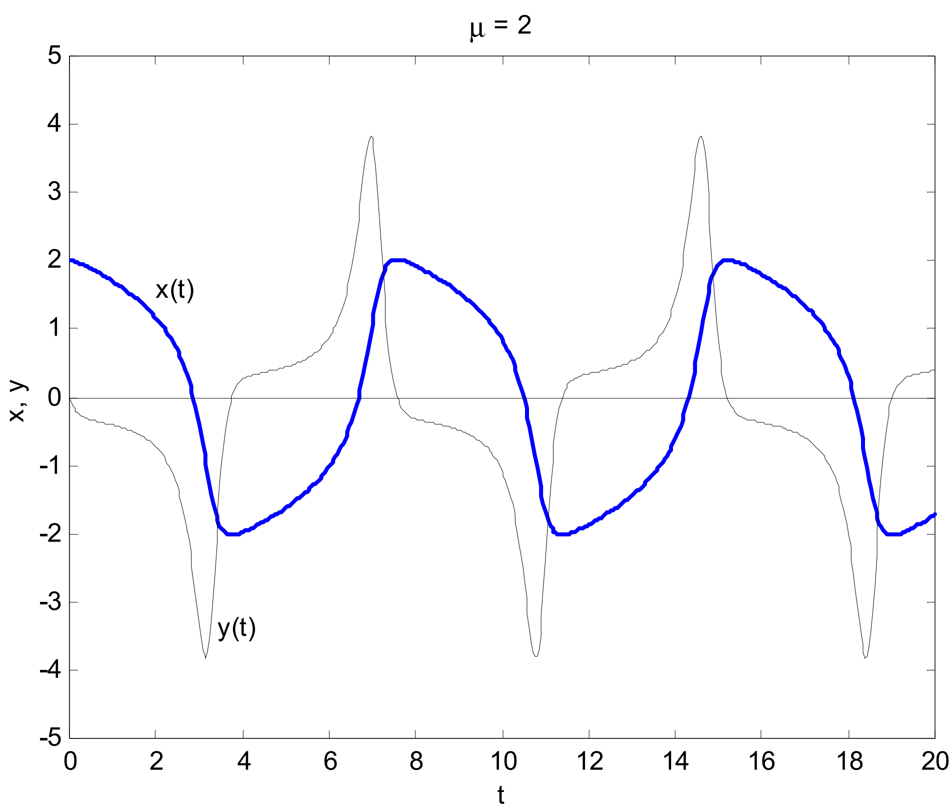
plots the corresponding xy-solution curve shown at the top of the next page. Finally, the command

```
plot(t,y(:,1), t,y(:,2) )
```

plots the corresponding tx- and ty-solution curves, on which the approximate period of oscillation can be measured. This plot is shown at the bottom of the next page.

When μ is large, van der Pol's equation is quite "stiff" and the periodic trajectory is more eccentric as illustrated in Fig. 9.4.20 in the text, which was plotted using MATLAB's stiff ODE solver **`ode15s`**. Indeed, the ability to plot Fig. 9.4.20 accurately — and especially the corresponding $x(t)$ and $y(t)$ solution curves analogous to those in Fig. 9.4.19 — is a good test of the robustness of your computer system's ODE solver.


μ = 2
5
4
3
2
1
0
-1
-2
-3
-4
-5
y
-3
-2
-1
0
1
2
3
x


μ = 2
x(t)
y(t)
x, y
t
0
2
4
6
8
10
12
14
16
18
20