# Computational physics

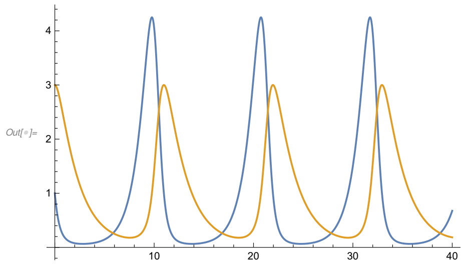
Jona Ackerschott, Julian Mayr

## Problem set 4

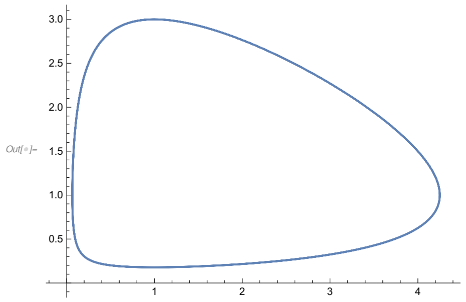
## Problem 1

Copying the given Mathematica code into our own notebook returns a plot of the numerical solution

```
In[•]:= NDSolve[{u'[t] == u[t] (1 - v[t]), v'[t] == 0.5 v[t] (u[t] - 1), u[0] == 1, v[0] == 3},
         {u, v}, {t, 0, 40}]

Out[•]= {{u → InterpolatingFunction[ Domain: {{0., 40.}} Output: scalar ],
          v → InterpolatingFunction[ Domain: {{0., 40.}} Output: scalar ]}}

In[•]:= Plot[{Evaluate[u[t] /. %2], Evaluate[v[t] /. %2]}, {t, 0, 40}]
```



```
In[•]:= ParametricPlot[{Evaluate[{u[t], v[t]} /. %2]}, {t, 0, 40}]
```

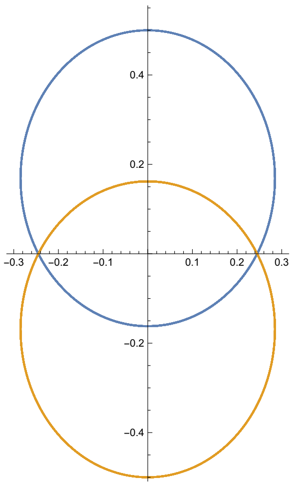


The function *NDSolve* creates an object which contains Mathematica's numerical interpretation of the ODE. This object can be evaluated using *Evaluate*, and then plotted using *Plot*, or *ParametricPlot* for a parametric plot.
This can also be used to solve vector valued differential equations, like the 2-body-problem.

This was done in the following notebook:

```
In[•]:=

In[•]:= S = NDSolve[{u''[t] == 2 * (v[t] - u[t]) / Norm[v[t] - u[t]]^3,
     v''[t] == 2 * (u[t] - v[t]) / Norm[v[t] - u[t]]^3, u[0] == {0, 0.5},
     v[0] == {0, -0.5}, u'[0] == {0.7, 0}, v'[0] == {-0.7, 0}}, {u, v}, {t, 0, 40}]

Out[•]= {{u → InterpolatingFunction[ Domain: {{0., 40.}}
                                     Output dimensions: {2} ],

     v → InterpolatingFunction[ Domain: {{0., 40.}}
                                Output dimensions: {2} ]}}

In[•]:= ParametricPlot[{Evaluate[u[t] /. S], Evaluate[v[t] /. S]}, {t, 0, 40}]
```

Out[•]=




The typical 2-Body-Problem Movement can be seen clearly. There is also no euler-like error to be seen, because *NDSolve* uses a pretty complex combination of ODE solving algorithms. For additional accuracy, *PrecisionGoal* and *AccuracGoal* variables can be given, though they are not necessary for this problem.

# Problem 2

Through examination of the population dynamics equation

$$\frac{\mathrm{d}N}{\mathrm{d}t} = rN(1 - N/K) - \frac{BN^2}{A^2 + N^2} \tag{1}$$

one sees immediately, using the dimensions of N (dimension: count, here named C) and t (dimension: time, here named T) that $[r] = \frac{1}{CT}$, $[K] = C$, $[A] = C$ and $[B] = \frac{C}{T}$. So with the definitions $\kappa = \frac{K}{A}$, $\varrho = \frac{A}{B}r$, $\tau = \frac{B}{A}t$ and $n = \frac{N}{A}$ one obtains a dimensionless form of (1).

$$\frac{\mathrm{d}n}{\mathrm{d}\tau} = \varrho n\left(1 - \frac{n}{\kappa}\right) - \frac{n^2}{1 + n^2} \tag{2}$$

From that stationary points can be found by imposing $\frac{\mathrm{d}n}{\mathrm{d}\tau} = 0$

$$\begin{aligned} &\varrho n\left(1 - \frac{n}{\kappa}\right) - \frac{n^2}{1 + n^2} = 0 \\ \Leftrightarrow\ &\varrho n\left(1 - \frac{1}{\kappa}n + n^2 - \frac{1}{\kappa}n^3\right) - n^2 = 0 \\ \Leftrightarrow\ &n^3 - \kappa n^2 + \left(1 + \frac{\kappa}{\varrho}\right)n - \kappa = 0 \end{aligned} \tag{3}$$

fig2_1.pdf

Figure 1: Stationary points of (2) in dependence of the parameter $\varrho$. If the plot is multivalued there exists more than one solution to (3)

Now while setting $\kappa = 7.5$, one can compute the zeros or rather the stationary points from this equation, for a specific value for $\varrho$, by applying the Mathematica method `NRoots` on (3). For values of $\varrho$ from $-1$ to 1 this leads to the plot shown in figure 1.
This graph also shows the points at which the number of solution changes from 1 to 3 (or rather from 1 to 2 to 3) or reversed.
The stability of the stationary points can be determined by the sign of the derivative of (3) with respect to $n$ at the specific root. The derivative at the respective points can be determined in Mathematica as well and a plot of these derivative for each value of $\varrho$ from -1 to 1 is shown below in figure 2.
Now one can simplify identify stable points by a negative derivative and unstable points by a positive derivative. This statement holds, because if the derivative at the stationary point is negative, than a small deviation of this point would lead to a sign of $\frac{\mathrm{d}n}{\mathrm{d}\tau}$ „in the opposite direction“ of the deviation, which means that the value will get closer to the stationary point for a later time. Analogous the value will get farther away from the stationary point in such a case, if the derivative is positive. So, one can seamply read off the stability from figure 2.

fig2_2.pdf

Figure 2: Derivative of (3) with respect to $n$ at all roots in dependence of $\varrho$. The colors correspond to the colors of the respective roots in figure 1.

```
In[1]:=  kappa=7.5

In[2]:=  p[n_,rho_]:=n^3 - kappa*n^2+(1+kappa/rho)*n-kappa

In[3]:=  R1[rho_]:= NRoots[p[n,rho],n][[1]][[2]]
         R2[rho_]:= NRoots[p[n,rho],n][[2]][[2]]
         R3[rho_]:= NRoots[p[n,rho],n][[3]][[2]]

In[4]:=  DN[n_,rho_]:= D[p[m,rho],m] /. m -> n

In[5]:=  Plot[{R1[rho],R2[rho],R3[rho]},{rho,-1,1},PlotRange->{-10,15},ImageSize->Large]

In[6]:=  Plot[{DN[R1[rho],rho],DN[R2[rho],rho],DN[R3[rho],rho]},{rho,-1,1},PlotRange->{-
```

Source code 1: Mathematica input for problem 2