1 Ordinary Differential Equations

Physical systems evolves in space and time, and very often they are described by a ordinary differential equations (ODE) and/or partial differential equations (PDE). We will first consider ODE, the difference between an ODE and a PDE is that an ODE only describes the changes in one spatial dimension *or* time. Whereas a PDE describes a system that evolves in the $x-$, $y-$, $z-$ dimension and/or in time. For ODEs a popular method of choice is usually the Runge Kutta of fourth order. In the following we will spend a significant amount of time to explore one of the simplest algorithm, Eulers method. Sometimes this is exactly the algorithm you would like to use, but with very little extra effort much more sophisticated algorithms can easily be implemented. However, all these algorithms, will at some point run into the same kind of troubles if used reckless. Thus we will use the Eulers method as a play ground, investigate when the algorithm run into trouble and suggests ways to fix it, these approaches can easily be extended to the higher order methods.

2 A Simple Model for Fluid Flow

Let us consider a simple example from chemical engineering, a continuous stirred tank reactor (CSTR), see figure 1. The flow is incompressible ($q_{\text{out}} = q_{\text{in}}$), a fluid is entering on the top and exiting at the bottom, the tank has a fixed volume V. Assume that the tank is filled with saltwater, and that freshwater is pumped into it, how much time does it take before 90% of the saltwater is replaced with freshwater? The tank is *well mixed*, illustrated with the propeller, this means that at every time the concentration is uniform in the tank, i.e. that $C(t) = C_{\text{out}}(t)$.

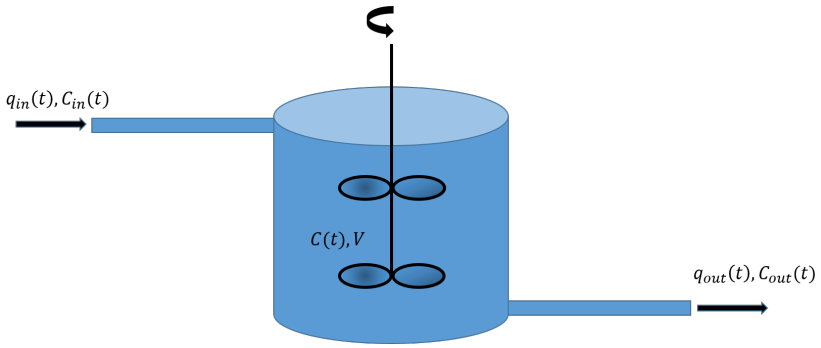

Figure 1: A continuous stirred tank model, $C(t) = C_{\text{out}}(t)$, and $q_{\text{out}} = q_{\text{in}}$.

The concentration C is measured in gram of salt per liter water, and the flow rate q is liter of water per day. The model for the salt balance in this system

can be described in words by:

$$\begin{aligned}[\text{accumulation of salt}] = [\text{salt into the system}] - [\text{salt out of the system}] \\ + [\text{generation of salt}]. \end{aligned} \tag{1}$$

In our case there are no generation of salt within the system so this term is zero. The flow of salt into the system during a time Δt is: $q_{\text{in}}(t) \cdot C_{\text{in}}(t) \cdot \Delta t = q(t) \cdot C_{\text{in}}(t) \cdot \Delta t$, the flow of salt out of the system is: $q_{\text{out}}(t) \cdot C_{\text{out}}(t) \cdot \Delta t = q(t) \cdot C(t) \cdot \Delta t$, and the accumulation during a time step is: $C(t + \Delta t) \cdot V - C(t) \cdot V$, hence:

$$C(t + \Delta t) \cdot V - C(t) \cdot V = q(t) \cdot C_{\text{in}}(t) \cdot \Delta t - q(t) \cdot C(t) \cdot \Delta t. \tag{2}$$

Note that it is not a priori apparent, which time the concentrations and flow rates on the right hand side should be evaluated at, we could have chosen to evaluate them at $t + \Delta t$, or at any time $t \in [t, t + \Delta t]$. We will return to this point later in this chapter. Dividing by Δt, and taking the limit $\Delta t \to 0$, we can write equation (2) as:

$$V \frac{dC(t)}{dt} = q(t) \left[C_{\text{in}}(t) - C(t) \right]. \tag{3}$$

Seawater contains about 35 gram salt / liter fluid, if we assume that the fresh water contains no salt, we have the boundary conditions $C_t extin(t) = 0$, $C(0)$ =35gram/l. The equation (3) reduces to (assuming a constant flow rate q):

$$V \frac{dC(t)}{dt} = -qC(t), \tag{4}$$

this equation can easily be solved, by dividing by C, multiplying by dt and integrating:

$$\begin{aligned} \int_{C_0}^{C} \frac{dC}{C} &= -q \int_0^t dt, \\ C(t) &= C_0 e^{-t/\tau}, \text{ where } \tau \equiv \frac{V}{q}. \end{aligned} \tag{5}$$

This equation can be inverted to give $t = -\tau \ln[C(t)/C]$. If we assume that volume of the tank is 1m^3=1000liters, and that the flow rate is 1 liter/min, we find that τ=1000min=0.69days and that it takes about 1.6 days to reduce the concentration by 90% to 3.5 gram/liter.

A note on the CSTR.

You might think that the CSTR is a very simple model, but this type of model is the basic building blocks in chemical engineering. By putting CSTR tanks in series and/or connecting them with pipes, the efficiency of manufacturing various type of chemicals can be investigated. Altough the CSTR is an idealized model for the part of a chemical factory, it is actually a *very good* model for fluid flow in a porous media. By connecting

CSTR tanks in series, one can model how chemical tracers propagate in the subsurface. The physical reason for this is that dispersion in porous media will play the role of the propellers and mix the concentration uniformly.

3 Eulers Method

In a more complicated case, several tanks in series, with varying degree of flow rate or if salt was generated in the tank, it might be very hard to solve this equation analytically. Actually we already have developed a numerical algorithm to solve equation (3), before we arrived at equation (3) in equation (2). This is a special case of Eulers method, which is basically to replace the derivative in equation (3), with $(C(t+\Delta t) - C(t))/\Delta t$. By rewriting equation (2), so that we keep everything related to the new time step, $t + \Delta t$, on one side, we get:

$$C(t+\Delta t) = C(t) + \frac{\Delta t}{\tau}\left[C_{\text{in}}(t) - C(t)\right], \tag{6}$$

we introduce the short hand notation: $C(t) = C_n$, and $C(t+\Delta t) = C_{n+1}$, hence the algorithm can be written more compact as:

$$C_{n+1} = \left(1 - \frac{\Delta t}{\tau}\right) C_n + \frac{\Delta t}{\tau} C_{\text{in},n}, \tag{7}$$

We have implemented equation (7), and in figure 2 the result is shown.

Clearly we see that the results are dependent on the step size, as the step increases the numerical solution deviates from the analytical solution. At some point the numerical algorithm fails completely, and produces results that have no meaning.

3.1 Error Analysis - Eulers Method

There are two obvious questions:

1. when does the algorithm produce unphysical results?
2. what is an appropriate step size?

Let us consider question 1 first, clearly when the concentrations gets negative the solution is unphysical. From equation (7), we see that when $\Delta t/\tau > 1$, there is a chance that concentrations become negative. For this specific case (the CSTR), there is a clear physical interpretation of this condition, inserting $\tau = V/q$, we can rewrite the condition $\Delta t/\tau < 1$ as $q\Delta t < V$. $q\Delta t$ is equal to the volume into the tank during one time step, which means that whenever we *flush more than one tank volume through the tank during one time step, the algorithm fails.* Whenever that happens we have no idea of the new concentration in the tank. This makes sense, because we calculate the new concentration as a modification to the old concentration, and if most (or all) the water has been replaced the new concentration is far a way from the old.

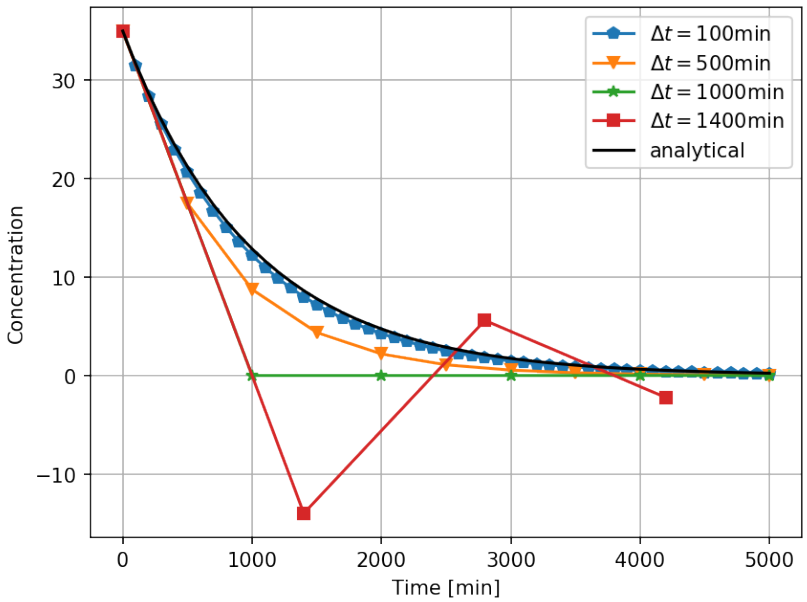

Figure 2: The concentration in the tank for different step size Δt.

The second question is a bit more difficult to answer. One strategy could be to simply use the results from chapter [Taylor], in that chapter we showed that a step size of 10^{-8} is the best we can get by using a first order Taylor approximation. How does the value 10^{-8} relate to a step size of 1min? In order to see the connection with equation (3), we can rewrite the equation in a dimensionless time by making the following substitution: $t \to t/\tau$. As we found earlier $\tau = 1000$min, thus a step size of 1min would correspond to a dimensionless timestep of $\Delta t \to$1min/1000min$= 10^{-3}$. This number can be directly compared to the value 10^{-8}, which is the lowest value we can choose without getting into trouble with round off errors on the machine.

Notice.

It is a good idea to formulate the equation in terms of dimensionless variables. From a pure modeling point of view it is a way of getting some understanding of what combination of the parameters that controls the bahaviour of the system. For the case of the CSTR, there is a time scale $\tau = V/q$, which is an intrisic measure of time in the system. No matter what the flow rate trhough the tank or the volume of the tank is, it will always take 0.1τ before the concentration in the tank is reduced by 90%. From an algorithmic point of view, we can choose the same step size regardless of

the volume or the flow rate through the tank. Thus we don not need to rewrite the algorithm each time the physical system changes.

As already mentioned a step size of 10^{-8}, is propbably the smallest we can choose with respect to round off errors, but it is larger than nessecary and would lead to large simulation times. If it takes 1 second to run the simulation with a step size of 10^{-3}, it would take 10^5 seconds or 1 day with a step size of 10^{-8}. To continue the error analyses, we write our ODE as:

$$\frac{dy}{dt} = f(y,t), \tag{8}$$

or in discrete form:

$$\frac{y_{n+1} - y_n}{h} + \frac{h}{2}y''(\eta) = f(y,t).$$

$$y_{n+1} = y_n + hf(y,t) - \frac{h^2}{2}y''(\eta). \tag{9}$$

h is now the step size, equal to Δt if the derivative is with respect to t or Δx if the derivative is respect to x etc. Note that we have also included the error term related to the numerical derivative, $\eta \in [t, t+h]$. At each step we get an error term, and the total error for all the setps is:

$$\sum_{n=0}^{N-1} \frac{h^2}{2}y''(\eta) = \frac{h^2}{2}\sum_{n=0}^{N-1} f'(y,\eta) \simeq \frac{h}{2}\int_{t_0}^{t_f} f'(y,\eta)d\eta$$

$$= \frac{h}{2}\left[f(y(t_f),t_f) - f(y(t_0),t_0)\right]. \tag{10}$$

Note that when we replace the sum with an integral in the equation above, this is only correct if the step size is not too large. From equation (10) we see that even if the error term on the numerical derivative is h^2, the error term on the solution is proportional to h (one order lower), because we accumulate errors for each step.

In the follwo

thus we know that we make the equation dimensionless by By visual inspection we see that a step size of 1min give a numerical solution that lies on top of the analytical solution. In chapter In the cases where this is not possible, because there is no known analytical solution one decreases the step size systematically and investigate if the numerical solution changes. For the case with the CSTR, by visual inspection, we see that a about 1min gives a numerical solution that is close to the analytical solution. When discussing small and large step sizes it usually a good idea to use dimensionless numbers, which in this case can be obtained by changing variables from $t \to t/\tau$, hence Δt=1min $\to \Delta t$ = 1min/1000=10^{-3}. From section [Taylor] we know

Note that there are many ways of making an equation dimensionless, and usually some trial and error must be done in order to find the

A formal error analysis can be with numerical

```
import matplotlib.pyplot as plt
import numpy as np

def analytical(x,tau,ci):
    d=x/tau
    return ci*np.exp(-d)

def one_step(c_old, c_in, tau_inv,dt):
    fact=dt*tau_inv
    return (1-fact)*c_old+fact*c_in

def euler(c_into,c_init,t_final,vol,q,dt):
    f=[];t=[]
    tau_inv = q/vol
    c_in    = c_into #freshwater into tank
    c_old   = c_init #seawater present
    ti=0.
    while(ti <= t_final):
        t.append(ti); f.append(c_old)
        c_new = one_step(c_old,c_in,tau_inv,dt)
        c_old = c_new
        ti   += dt
    return t,f
dt = [100,500,1000,1400]
t=[];f=[]
t_final=5000;c_into = 0; c_init = 35
vol=1000;q=1
for dti in dt:
    ti,fi = euler(c_into,c_init,t_final,vol,q,dti)
    t.append(ti)
    f.append(fi)
# note that t has to converted to array in order to
# use simple operations like / * etc.
f_an = analytical(np.array(t[0]),vol/q,c_init)
symb = ['-p','-v','-*','-s']
fig = plt.figure(dpi=150)
for i in range(0,len(dt)):
    plt.plot(t[i], f[i], symb[i], label='$\Delta t =$'+str(dt[i])+'min')
plt.plot(t[0], f_an, '-', color='k',label='analytical')
plt.legend(loc='upper right', ncol=1)
plt.grid()
plt.ylabel('Concentration')
plt.xlabel('Time [min]')
# Save the plot in a file
plt.savefig('../fig-ode/euler.png', bbox_inches='tight',transparent=True)
plt.show()
```