Numerical integration of ODEs

ME 416 - Prof. Tron

Thursday 21st March, 2019

For some classes of ODEs (linear ODEs are an important case), it is possible to find explicit, analytical solutions (see the Wikipedia page for a summary). When such solutions cannot be found, or are too computationally expensieve, one can find an *approximate solution* numerically. There are many different *ODE solvers* (methods) for this, offering different tradeoffs between computation requirements, accuracy of the solution, and types of problems they can be applied to.

At a high level, given the *continuous time* ODE $\dot{x}(t) = f(x(t))$ (this denomination is given because here we can evaluate the equation for any t), an ODE solver defines a discrete increasing sequence of times, a *grid*, $t_0 = 0, t_1, t_2, \ldots$, and returns a *discrete time* solution $x[k]$ that approximates the true continuous time solution $x(t)$. Here, k is an integer for the discretized time (notice the use of square brackets), and the approximation of the solution is intended in the sense that $x[k] \approx x(t_k)$.

There are a few things to keep in mind:

- A solution x[k] computed by a solver does not say anything for instants t that are not in the grid $t_0, t_1, t_2, \ldots$ (although it is always possible to interpolate).
- Since solvers compute solutions progressively (starting from t_0 and moving forward), as a rule of thumb, the accuracy of the solution reduces as t increases (i.e., the errors accumulate).
- As another rule of thumb, using a denser grid (i.e., selecting a larger number of instants t_k for a same period of time) increases the accuracy of the solution, at the expense of a higher computational load.

1 Euler method

Euler explicit method (a.k.a. forward Euler method) is the simplest ODE solver. The time grid is defined by picking a *step size* h and then letting $t_k = kh$ (i.e., $t_{k+1} = t_k + h$ and the time instants are regularly spaced).

Then, the solution is built incrementally following the iteration $x[k+1] = x[k] + hf(x[k])$.

Derivation from approximation of the derivative Assume that we have computed the solution up to t_k, i.e., we have $x[k] \approx x(kh)$. From the definition of the derivative, we have

$$\dot{x}(t) = \lim_{h \to \infty} \frac{x(t+h) - x(t)}{h} = f(x(t)), \tag{1}$$

where the last equality comes from the ODE.

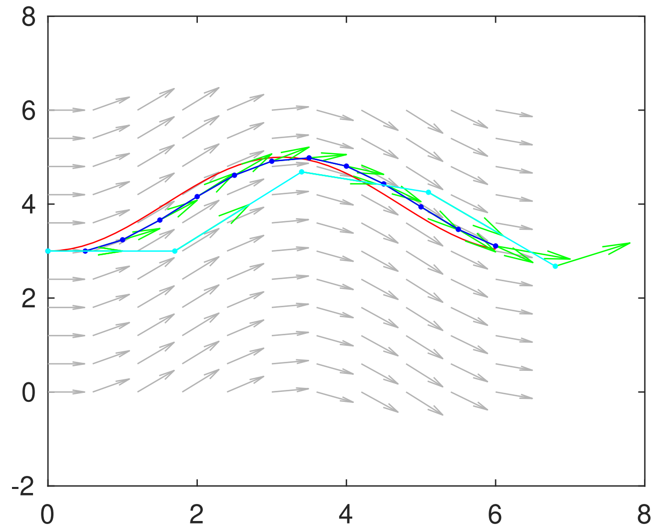

Figure 1: Example of application of Euler's method on the field $f(x) = \begin{bmatrix} 1 \\ \sin(x) \end{bmatrix}$. Gray arrow: field $f(x)$. Red: true solution. Blue and Cyan: Euler's approximate solution with $h = 0.5$ and $h = 1.7$. Green arrows: field $f(x)$ evaluated at the points in Euler's solutions.

We can approximate the derivative by approximating the limit with a small, but finite $h > 0$. Performing this approximation for $t = kh$ we have

$$\frac{x(kh + h) - x(kh)}{h} \approx f(x(kh)), \tag{2}$$

which, using the approximation $x[k] \approx x(kh)$, becomes

$$\frac{x((k + 1)h) - x(kh)}{h} \approx \frac{x[k + 1] - x[k]}{h} f(x[k])). \tag{3}$$

Rearranging, we obtain the Euler's method equation:

$$x[k + 1] = x[k] + hf(x[k])). \tag{4}$$

Graphical interpretation Graphically, Euler's method can be intepreted as: starting from the solution $x[k]$ at the current step, compute the field $f()$ at that point, follow the arrow after scaling it by h, let this new point be the value for $x[k + 1]$ at the successive time step, and then repeat. An example of the application of Euler's method for two different values of h is shown in Figure 1. Notice that both Euler's solution capture the main undulatory behavior of the solution, but the solution with $h = 0.5$ is much closer to the true solution.

2 Numerical stability

Euler's method can be numerically unstable, meaning that the numerical solution grows very large for equations where the exact solution does not. A typical way to counter this problem is to pick a smaller step size h.

TODO: show result on rotational field

Figure 2: Trivia: Euler's method is the method that saves the day in the movie "Hidden Figures". Credit: TM and (C) 2017 Twentieth Century Fox Film Corporation.