

CS341 SYSTEMS BIOLOGY I

LECTURE 15

NUMERICAL METHODS FOR SOLVING DIFFERENTIAL EQUATIONS (CONT)

So far, we have implemented Runge-Kutta methods with error $\mathcal{O}(\Delta t)$ (forward Euler) and $\mathcal{O}(\Delta t^2)$ (explicit trapezoidal; also called Heun's Method). Today, we implement a Runge-Kutta method with error $\mathcal{O}(\Delta t^4)$ – RK4 (for 4-th order Runge-Kutta method).

As we learned with the explicit trapezoidal method, more accurate Runge-Kutta methods take into account more information within the time step. Today, we see even more information being taken into account and the smallest error yet.

RK4

Just like the forward Euler and explicit trapezoidal methods relate to numerical integration of a known function $f(t)$, RK4 is related to Simpson's rule for numerical integration of a known function. Whereas the forward Euler and explicit trapezoidal methods calculate the area under straight lines, Simpson's rule calculates the area under a quadratic polynomial that intersects the known function f at the beginning, middle, and end of the interval from a to b . On the Wikipedia page (https://en.wikipedia.org/wiki/Simpson's_rule) for Simpson's Rule, they show that the following formula computes the area under that quadratic polynomial. They also show the derivation, so it is worth checking out.

$$\frac{b-a}{6} \left(f(a) + 4f\left(\frac{a+b}{2}\right) + f(b) \right)$$

The RK4 method solves $\frac{dy(t)}{dt} = f(t, y(t))$ for $\hat{y}(t)$ at each time step by estimating the y and f values of the midpoint of the interval and using a weighted average of those f estimates to take one step. It begins like the explicit trapezoidal method to take a step to the midpoint. Then, we use the slope estimated at the midpoint to take a step to the end of the interval, where we compute another slope. Finally, we use a weighted average of the slopes at the beginning, middle, and end of the time step to take our ultimate step. (See Figure 1 for an illustration.)

The method uses the same format as Simpson's rule, except that the contribution of f at the midpoint involves two separate estimates of the midpoint. The method is

$$\begin{aligned}
f_1 &= f(t, \hat{y}(t)) \\
f_2 &= f\left(t + \frac{\Delta t}{2}, \hat{y}(t) + \frac{\Delta t}{2} f_1\right) \\
f_3 &= f\left(t + \frac{\Delta t}{2}, \hat{y}(t) + \frac{\Delta t}{2} f_2\right) \\
f_4 &= f(t + \Delta t, \hat{y}(t) + \Delta t f_3) \\
\hat{y}(t + \Delta t) &= \hat{y}(t) + \frac{\Delta t}{6} \cdot (f_1 + 2f_2 + 2f_3 + f_4)
\end{aligned}$$

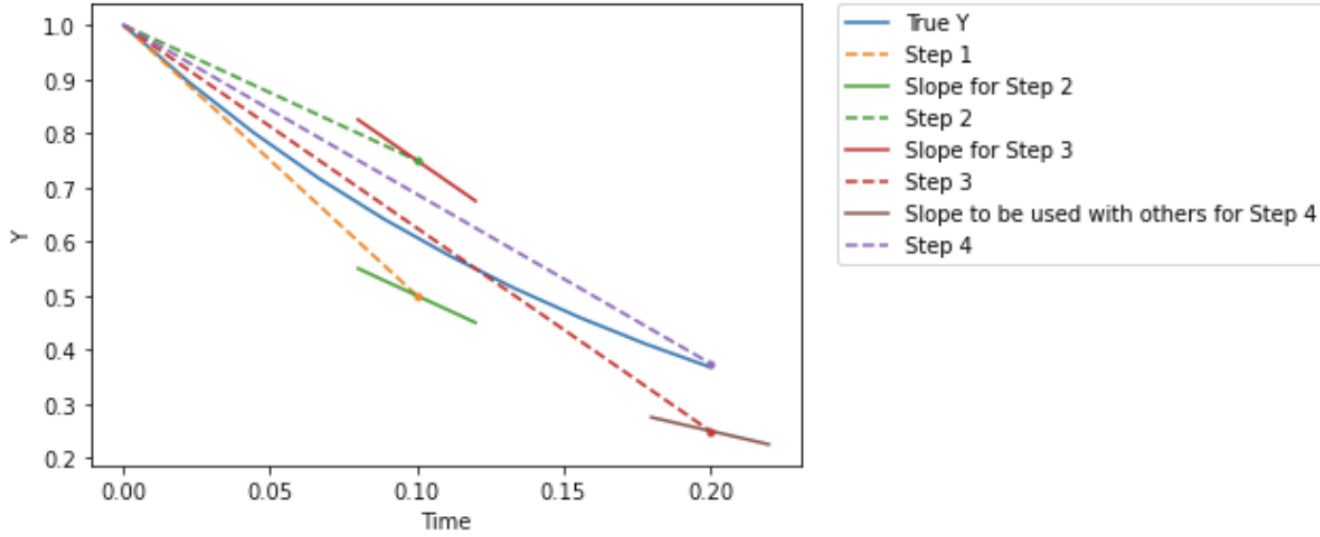


FIGURE 1. RK4 method used to estimate the one time step of test equation (degradation model) with degradation rate constant 5, initial condition $y(0) = 1$, and a time step of 0.2. We show the true solution (y ; solid blue line), the ultimate RK4 estimate for the end of the time step (purple dot), and the intermediate computations. The first step is a forward Euler step half-way through the interval (orange dashed line to orange dot). The slope at the orange dot (green line) is then used to take a second step (green dashes line), estimating the value of y half-way through the interval (green dot; this is akin to an explicit trapezoidal step for the half-interval). The slope at the green dot (red line) is then used to estimate the full step (red dashed line, ending in red dot). The slope at the red dot is our fourth slope estimate. A weighted sum of the slopes is then used to take the final step (purple dashed line to the purple dot).

The error from RK4 ($\mathcal{O}(\Delta t^4)$) will generally be smaller than the error from both explicit trapezoidal ($\mathcal{O}(\Delta t^2)$) and forward Euler ($\mathcal{O}(\Delta t)$). (See Figure 2.)

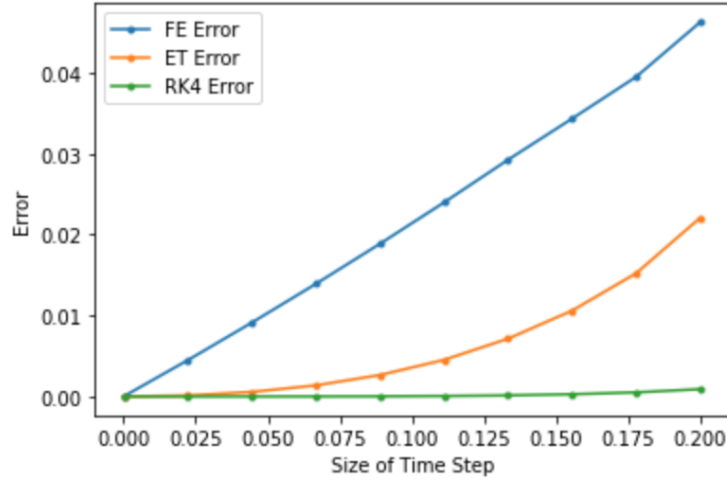


FIGURE 2. For the test equation with 2 ODE's with degradation rates of 2 and 5, integrated from $y_0=(1,1)$ from $t=0$ to $t=4$, we solved the equations with the forward Euler (FE), explicit trapezoidal (ET), and RK4 methods for time steps ranging from $1e-4$ to 0.2 . The error from each simulation was computed as the mean over time of the Euclidean distance between the estimated solution and the true solution at each time step.