

# NE 155, Class 28, S15

## Taylor Series Methods and Runge-Kutta

guest lecturer: Kathryn Huff

April 3, 2015

---

### 1 Introduction

Recall that the initial value problem takes the form:

$$u'(t) = f(u(t), t), \text{ for } t > t_0 \tag{1}$$

where

$$u(t_0) = u_0 \tag{2}$$

In such a problem, we desire to compute  $u(t_1)$ ,  $u(t_2)$ ,  $u(t_n)$  and so on.

There are many methods for solving initial value problems numerically. This lesson will introduce a simple one, Forward Euler, which is derived from a Taylor series expansion. We will then use Forward Euler to introduce a two-stage explicit Runge-Kutta method as well, for higher accuracy.

## 2 Taylor Series Derivation of Forward Euler

The simplest method for finding  $u(t_n)$  is Forward Euler, which approximates  $u(t_n)$ . Let's call this approximation,  $U^n$ . In this notation, Forward Euler is based on replacing  $u'(t_n)$  with  $(U^{n+1} - U^n)/k$ , where  $k$  is the width of the timestep. The Forward Euler method arises from a Taylor series expansion of  $u(t_{n+1})$  about  $u(t_n)$ :

$$u(t_{n+1}) = u(t_n) + ku'(t_n) + \frac{1}{2}k^2u''(t_n) + \dots \quad (3)$$

With this, the  $O(k^2)$  terms can be dropped to give:

$$u(t_{n+1}) \approx u(t_n) + ku'(t_n) \quad (4)$$

And, based on equation (1) we can replace  $u'(t_n)$  with  $f(u(t_n), t_n)$ :

$$\frac{1}{2}k^2u''(t_n) + \dots \quad (5)$$

$$u(t_{n+1}) = u(t_n) + kf(u(t_n), t_n) \quad (6)$$

This expression gives a truncation error of order  $O(k^2)$ . More accurate schemes can be derived with a Taylor series expansion by retaining higher order terms in equation (3). Since we are only given  $u'(t_n) = f(u(t_n), t_n)$ , however, the computation of such schemes requires repeated recursive differentiation of this function, and can get quite messy.

## 3 Runge-Kutta Methods

Runge-Kutta is a method used in practice to get a higher order approximation *without* explicitly calculating higher order derivatives.

Runge-Kutta uses two stages. The first stage is an update using Euler's method, approximating  $u(t_{n+1/2})$ .

$$U^{n+1/2} = U^n + \frac{1}{2}kf(U^n) \quad (7)$$

$$(8)$$

The second stage evaluates the function,  $f$ , at the midpoint to estimate the slope.

$$U^{n+1} = U^n + kf(U^{n+1/2}) \quad (9)$$

These equations can be combined into a single expression:

$$U^{n+1} = U^n + kf(U^n + \frac{1}{2}kf(U^n)) \quad (10)$$

This approximation, because it uses two points, like a centered approximation, is order  $O(k)$  accurate.

A generic r-stage Runge-Kutta method can be expressed as:

$$Y_1 = U^n + k \sum_{j=1}^r a_{1j}f(Y_j, t_n + c_jk) Y_2 = U^n + k \sum_{j=2}^r a_{2j}f(Y_j, t_n + c_jk) \dots Y_r = U^n + k \sum_{j=r}^r a_{rj}f(Y_j, t_n + c_jk) U^n \quad (11)$$

## 4 Application to PRKE

Each of these can be applied to the PRKE. In particular, let's consider the application of a Forward Euler.

To avoid confusion with the multiplication factor, the width of our timestep will be called  $\Delta t$ .

$$n(t_{n+1}) = n(t) + \Delta t \left[ \frac{\rho(t_n) - \beta}{\Lambda} n(t_n) + \sum_{j=1}^{j=J} \lambda_j \zeta_j \right] \quad (12)$$

$$(13)$$