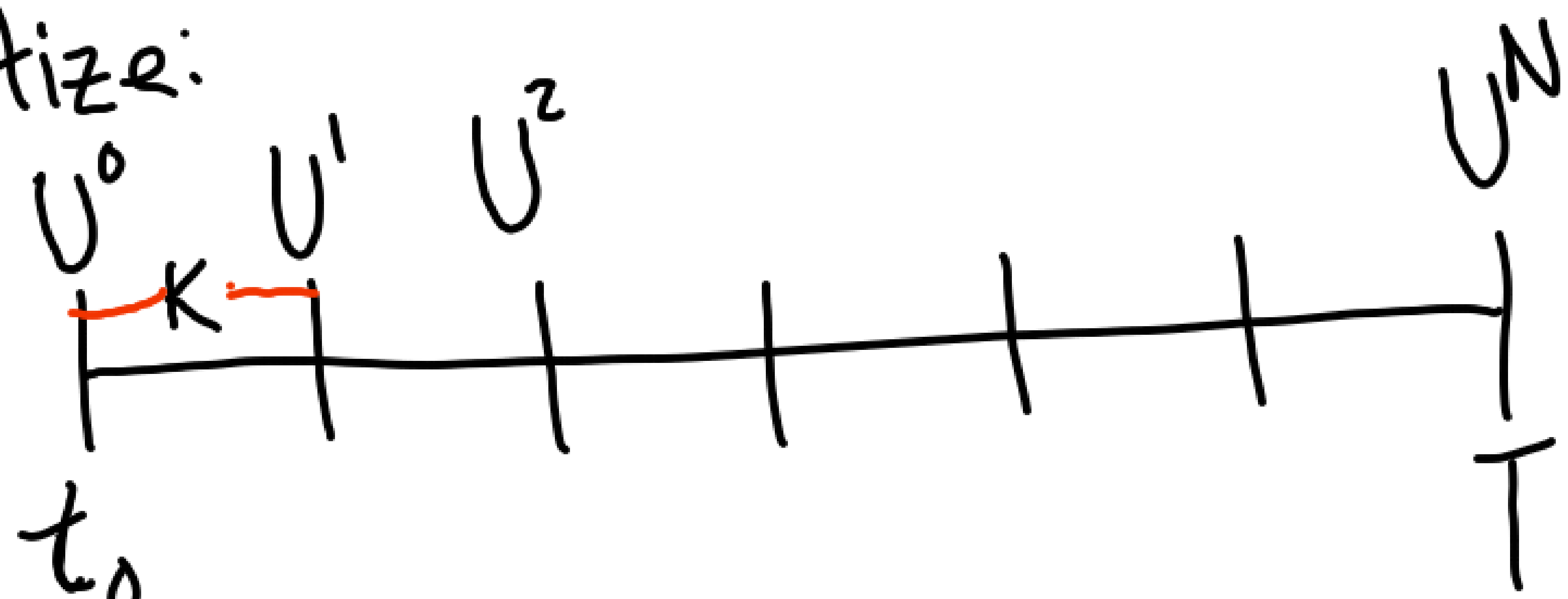
# Numerical Methods for the initial value problem

$$u'(t) = f(u) \qquad u(t): \mathbb{R} \to \mathbb{R}^m$$
$$u(t_0) = \eta \qquad f: \mathbb{R}^m \to \mathbb{R}^m$$
$$t \in [t_0, T]$$

Discretize:




$$N = \frac{T - t_0}{k}$$

We will march forward in time, computing $U^1$, then $U^2$, etc.

Computational cost:

$$N \times (\text{cost per step})$$
$$= (\text{cost per step}) \times \frac{T - t_0}{k}$$

# Some basic methods

## Explicit Euler

$$U^{n+1} = U^n + kf(U^n)$$

or $$\frac{U^{n+1} - U^n}{k} = f(U^n)$$

Conditionally stable.

## Implicit Euler

$$U^{n+1} = U^n + kf(U^{n+1})$$

Need to solve algebraic eqns. at each step.

Unconditionally stable.

## Trapezoidal method

$$U^{n+1} = U^n + \frac{K}{2}\left(f(U^n) + f(U^{n+1})\right)$$

## Leapfrog Method

$$U^{n+1} = U^{n-1} + 2Kf(U^n)$$

$$\frac{U^{n+1} - U^{n-1}}{2K} = f(U^n)$$

## Local Truncation Error

Trapezoidal:

$$\frac{U^{n+1} - U^n}{K} = \frac{1}{2}\left(f(U^n) + f(U^{n+1})\right)$$

Substitute $U^n \to u(t_n)$:

$$\frac{u(t_{n+1}) - u(t_n)}{K} = \frac{1}{2}\left(f(u(t_n)) + f(u(t_{n+1}))\right)$$

$$u(t_{n+1}) = u(t_n) + Ku'(t_n) + \frac{K^2}{2}u''(t_n) + \frac{K^3}{6}u'''(t_n) + O(K^4)$$

$$f(u(t_{n+1})) = u' + Ku'' + \frac{K^2}{2}u''' + \frac{K^3}{6}u'''' + O(K^4)$$

$$\frac{\cancel{u}+\cancel{ku'}+\frac{k^2}{2}u''+\frac{k^3}{6}u'''+O(k^4)-\cancel{u}}{k}=\frac{1}{2}\left(\cancel{u'}+\underline{ku''}+\underline{\frac{k^2}{2}u'''}+\frac{k^3}{6}u^{(4)}+O(k^4)+\cancel{u'}\right)+\tau^n$$

$$\cancel{\frac{k}{2}u''}+\frac{k^2}{6}u'''+O(k^3)=\cancel{\frac{k}{2}u''}+\frac{k^2}{4}u'''+O(k^3)+\tau^n$$

$$\tau^n=\frac{-k^2}{12}u'''(t_n)+O(k^3)$$

2nd-order accurate

---

One-step error

If we write $U^{n+1}=U^n+\frac{k}{2}\left(f(U^n)+f(U^{n+1})\right)$

And substitute: $u(t_{n+1})=u(t_n)+\frac{k}{2}\left(f(u(t_n))+f(u(t_{n+1}))\right)+k\tau^n$

We call $\mathcal{L}^n=k\tau^n$ the "one-step error"

# How to achieve higher-order accuracy

1. Use more derivatives — multiderivative
2. Use more evaluations of $f$ — multi-stage
3. Use more previous steps — multistep

# Multiderivative Methods

$$u(t_{n+1}) = u(t_n) + k u'(t_n) + \frac{k^2}{2} u''(t_n) + \frac{k^3}{6} u'''(t_n) + O(k^4)$$

$$U^{n+1} = U^n + k f(U^n) + \frac{k^2}{2} f' f(U^n) + \frac{k^3}{6}\left(f''(f,f) + f'f'f\right)$$

3rd-order accurate

$$\frac{d^2}{dt^2} u_i(t) = \frac{d}{dt} f_i(u(t)) = \sum_{j=1}^{m} \frac{\partial f_i}{\partial u_j(t)} u_j'(t)$$

$$= \sum_{j=1}^{m} \frac{\partial f_i}{\partial u_j} f_j = f' f$$

$$\frac{d^3 u}{dt^3} = \frac{d}{dt}\left(\sum_{j=1}^{m} \frac{\partial f_i}{\partial u_j} f_j\right)$$

$$= \sum_{j=1}^{m} \sum_{\ell=1}^{m} \frac{\partial^2 f_i}{\partial u_j \partial u_\ell} f_j$$

$$+ \sum_{j=1}^{m} \frac{\partial f_i}{\partial u_j} \sum_{\ell=1}^{m} \frac{\partial f_j}{\partial u_\ell} f_\ell$$

$$= f''(f,f) + f'f'f$$

# ② Multi-stage (Runge-Kutta)

Example: (midpoint method)

$$U^* = U^n + \frac{k}{2} f(U^n)$$

$$U^{n+1} = U^n + k f(U^*)$$

$$\begin{array}{|cc} 0 & 0 \\ \frac{1}{2} & 0 \\ \hline 0 & 1 \end{array}$$

$$\boxed{\begin{aligned} Y_1 &= U^n \\ Y_2 &= U^n + \tfrac{1}{2} k f(Y_1) \\ U^{n+1} &= U^n + k f(Y_2) \end{aligned}}$$

$$U^{n+1} = U^n + k f\left(U^n + \frac{k}{2} f(U^n)\right)$$

$$u(t_{n+1}) = u(t_n) + k f\left(u(t_n) + \frac{k}{2} f(u(t_n))\right) + k\tau^n$$

$$f\left(u + \frac{k}{2} f(u)\right) = f(u) + \frac{k}{2} f' f(u) + \frac{k^2}{8} f''(f,f) + \mathcal{O}(k^3)$$

$$\cancel{u} + \cancel{ku'} + \cancel{\frac{k^2}{2} u''} + \frac{k^3}{6} u''' = \cancel{u} + \cancel{kf} + \cancel{\frac{k^2}{2} f'f} + \frac{k^3}{8} f''(f,f) + \mathcal{O}(k^4) + k\tau^n$$

$$\tau^n = \mathcal{O}(k^2)$$

RK methods

$$Y_i = U^n + k\sum_{j=1}^{s} a_{ij} f(Y_j)$$

$$U^{n+1} = U^n + k\sum_{j=1}^{s} b_j f(Y_j)$$

$$\begin{array}{c|c} & A \\ \hline & b^T \end{array}$$

4th-order Runge-Kutta

$$\begin{array}{c|cccc} & 0 & 0 & 0 & 0 \\ & \frac{1}{2} & 0 & 0 & 0 \\ & 0 & 1 & 0 & 0 \\ & 0 & 0 & 1 & 0 \\ \hline & \frac{1}{6} & \frac{1}{3} & \frac{1}{3} & \frac{1}{6} \end{array}$$

Pros/Cons

+ Self-starting
+ Easy to adapt $k$
− Multiple evaluations of $f$ per step

## ③ Multistep

Example: Leapfrog

$$U^{n+1} = U^{n-1} + 2Kf(U^n)$$

Pros/Cons

\+ Only evaluate f once per step

\- Not self-starting

\- Hard to adapt K