PART II

Finite Difference Methods for Differential Equations
BOUNDARY VALUE PROBLEMS (I)

- Solving a **TWO–POINT BOUNDARY VALUE PROBLEM** with **DIRICHLET BOUNDARY CONDITIONS**:

\[
\frac{d^2y}{dx^2} = g \quad \text{for } x \in (0, 2\pi) \\
y(0) = y(2\pi) = 0
\]

- Finite–difference approximation:
  - Second–order center difference formula for the interior nodes:
    \[
    \frac{y_{j+1} - 2y_j + y_{j-1}}{h^2} = g_j \quad \text{for } j = 1, \ldots, N
    \]
    where \( h = \frac{2\pi}{N+1} \) and \( x_j = jh \)
  - Endpoint nodes:
    \[
    y_0 = 0 \implies y_2 - 2y_1 = h^2 g_1 \\
    y_{N+1} = 0 \implies -2y_N + y_{N-1} = h^2 g_N
    \]
  - Tridiagonal algebraic system — solved very efficiently with the **THOMAS ALGORITHM** (a version of the Gaussian elimination)
BOUNDARY VALUE PROBLEMS (II)

- Solving a **TWO–POINT BOUNDARY VALUE PROBLEM** with **NEUMANN BOUNDARY CONDITIONS**:
  \[
  \frac{d^2y}{dx^2} = g \quad \text{for } x \in (0, 2\pi)\\
  \frac{dy}{dx}(0) = \frac{dy}{dx}(2\pi) = 0
  \]

- Finite–difference approximation:
  - Second–order center difference formula for the interior nodes:
    \[
    \frac{y_{j+1} - 2y_j + y_{j-1}}{h^2} = g_j \quad \text{for } j = 1, \ldots, N
    \]
  - First–order Forward/Backward Difference formulae to re–express endpoint values:
    \[
    \frac{y_1 - y_0}{h} = 0 \implies y_0 = y_1\\
    \frac{y_{N+1} - y_N}{h} = 0 \implies y_{N+1} = y_N
    \]

  First–order only — **DEGRADED ACCURACY**!

- Tridiagonal algebraic system — Is there any problem? Where?
**Boundary Value Problems (III)**

- In order to retain the **second-order accuracy** in the approximation of the Neumann problem need to use higher-order formulae at endpoints, e.g.

  $$y'_{0} = \frac{-y_{2} + 4y_{1} - 3y_{0}}{2h} = 0 \implies y_{0} = \frac{1}{3}(-y_{2} + 4y_{1})$$

- The first row thus becomes

  $$\frac{2}{3}y_{2} - \frac{2}{3}y_{1} = h^{2}g_{1}$$

  **Second-order accuracy recovered!**
**Boundary Value Problems (IV)**

- **Compact Stencils** — stencils based on three grid points (in every direction) only: \( \{x_{j+1}, x_j, x_{j-1}\} \) at the \( j-th \) node

- Is it possible to obtain higher (then second) order of accuracy on compact stencils? — **YES**!

- Consider the central difference approximation to the equation \( \frac{d^2y}{dy^2} = g \)

\[
\frac{y_{j+1} - 2y_j + y_{j-1}}{h^2} - \frac{h^2}{12} y^{(iv)}_j + O(h^4) = g_j
\]

- Re-express the error term \( \frac{h^2}{12} y^{(iv)}_j \) using the equation in question:

\[
\frac{h^2}{12} y^{(iv)}_j = \frac{h^2}{12} g''_j = \frac{h^2}{12} \left[ \frac{g_{j+1} - 2g_j + g_{j-1}}{h^2} - \frac{h^2}{12} g^{(iv)}_j + O(h^4) \right]
\]

- Inserting into the original finite–difference equation:

\[
\frac{y_{j+1} - 2y_j + y_{j-1}}{h^2} = g_j + \frac{g_{j+1} - 2g_j + g_{j-1}}{12} + O(h^4)
\]

- Slight modification of the RHS \( \implies \) **Fourth—Order Accuracy!!!**
BOUNDARY VALUE PROBLEMS (V)

- **COMPACT FINITE DIFFERENCE SCHEMES** —
  - **ADVANTAGES:**
    * Increased accuracy on compact grids
  - **DRAWBACKS:**
    * need to be tailored to the specific equation solved
    * can get fairly complicated for more complex equations
**Initial Value Problems — General Remarks**

- Consider the following Cauchy problem:

\[
\frac{dy}{dt} = f(y,t) \text{ with } y(t_0) = y_0
\]

The independent variable \( t \) is usually referred to as \textit{time}.

- Equations with higher-order derivatives can be reduced to systems of first-order equations.

- Generalizations to systems of ODEs straightforward.

- When the RHS function does not depend on \( y \), i.e., \( f(y,t) = f(t) \), solution obtained via a \textit{quadrature}.

- Assume uniform time-steps ( \( h \) is constant)
INITIAL VALUE PROBLEMS — CHARACTERIZATION OF INTEGRATION METHODS

- **ACCURACY** — unlike in the Boundary Value Problems, there is no *terminal condition* and approximation errors may accumulate in time; consequently, a relevant characterization of accuracy is provided by the **GLOBAL ERROR**

\[
\text{global error} = \text{local error} \times (\text{# of time steps}),
\]

rather than the **LOCAL ERROR**.

- **STABILITY** — unlike in the Boundary Value Problems, where boundedness of the solution at final time is enforced via a suitable *terminal condition*, in Initial Value Problems there is a priori no guarantee that the solution will remain bounded.
INITIAL VALUE PROBLEMS — MODEL PROBLEM

- **STABILITY** of various numerical schemes is usually analyzed by applying these schemes to the following **LINEAR MODEL**:

\[
\frac{dy}{dt} = \lambda y = (\lambda_r + i\lambda_i)y \text{ with } y(t_0) = y_0,
\]

which is stable when \( \lambda_r \leq 0 \).

- **EXACT SOLUTION**:

\[
y(t) = y_0 e^{\lambda t} = \left(1 + \lambda h + \frac{\lambda^2 h^2}{2} + \frac{\lambda^3 h^3}{6} + \ldots\right)y_0
\]

- **MOTIVATION** — consider the following **ADVECTION–DIFFUSION PDE**:

\[
\frac{\partial u}{\partial t} + c \frac{\partial u}{\partial x} - a \frac{\partial^2 u}{\partial x^2} = 0
\]

Taking Fourier transform yields (\( k \) is the wavenumber):

\[
\frac{d\hat{u}_k}{dt} + cik\hat{u}_k + ak^2\hat{u}_k = 0
\]

where
- the real term \( ak^2\hat{u}_k \) represents **DIFFUSION**
- the imaginary term \( cik\hat{u}_k \) represents **ADVECTION**
INITIAL VALUE PROBLEMS —
EXPLICIT EULER SCHEME (I)

• Consider a Taylor series expansion

\[ y(t_{n+1}) = y(t_n) + hy'(t_n) + \frac{h^2}{2}y''(t_n) + \ldots \]

Using the ODE we obtain

\[ y' = \frac{dy}{dt} = f \]
\[ y'' = \frac{dy'}{dt} = \frac{df}{dt} = f_t + ff_y \]

• Neglecting terms proportional to second and higher powers of \( h \) yields the EXPLICIT EULER METHOD

\[ y_{n+1} = y_n + hf(y_n, t_n) \]

• Retaining higher-order terms is inconvenient, as it requires differentiation of \( f \) and does not lead to schemes with desirable stability properties.
**INITIAL VALUE PROBLEMS — EXPLICIT EULER SCHEME (II)**

- **Local error** analysis:
  \[ y_{n+1} = (1 + \lambda h) y_n + [O(h^2)] \]

- **Global error** analysis:
  \[(\text{global error}) = C h^2 \cdot N = C h^2 \cdot \frac{T}{h} = C' h \]

Thus, the scheme is
- locally **second-order** accurate
- globally (over the interval \([t_0, t_0 + Nh]\)) **first-order** accurate
**Initial Value Problems — Explicit Euler Scheme (III)**

- Stability (for the model problem)

\[ y_{n+1} = y_n + \lambda h y_n = (1 + \lambda h) y_n \]

Thus, the solution after \( n \) time steps

\[ y_n = (1 + \lambda h)^n y_0 \triangleq \sigma^n y_0 \implies \sigma = 1 + \lambda h \]

For large \( n \), the numerical solution remains stable iff

\[ |\sigma| \leq 1 \implies (1 + \lambda_r h)^2 + (\lambda_i h)^2 \leq 1 \]

- **CONDITIONALLY STABLE** for real \( \lambda \)
- **UNSTABLE** for imaginary \( \lambda \)
**INITIAL VALUE PROBLEMS — IMPLICIT EULER SCHEME (I)**

- **Implicit Schemes** — based on approximation of the RHS that involve $f(y_{n+1}, t)$, where $y_{n+1}$ is the unknown to be determined.

- **Implicit Euler Scheme** — obtained by neglecting second and higher-order terms in the expansion:

\[
y(t_n) = y(t_{n+1}) - hy'(t_{n+1}) + \frac{h^2}{2} y''(t_{n+1}) - \ldots
\]

Upon substitution \( \frac{dy}{dt} \bigg|_{t_{n+1}} = f(y_{n+1}, t_{n+1}) \) we obtain

\[
y_{n+1} = y_n + hf(y_{n+1}, t_{n+1})
\]

The scheme is

- locally **SECOND–ORDER** accurate
- globally (over the interval \([t_0, t_0 + Nh]\)) **FIRST–ORDER** accurate
INITIAL VALUE PROBLEMS — IMPLICIT EULER SCHEME (II)

- Stability (for the model problem):

\[ y_{n+1} = y_n + \lambda h y_{n+1} \implies y_{n+1} = (1 - \lambda h)^{-1} y_n \]

\[ y_{n+1} = \left( \frac{1}{1 - \lambda h} \right)^n y_0 \triangleq \sigma^n y_0 \implies \sigma = \frac{1}{1 - \lambda h} \]

\[ |\sigma| \leq 1 \implies (1 - \lambda r h)^2 + (\lambda i h)^2 \geq 1 \]

Implicit Euler scheme is thus stable for
- all stable model problems
- most unstable model problems

- **Remark:** When solving systems of ODEs of the form \( \mathbf{y} = \mathbf{A}(t)\mathbf{y} \), each implicit step requires solution of an algebraic system: \( \mathbf{y}_{n+1} = (I - h\mathbf{A})^{-1}\mathbf{y}_n \)

- Implicit schemes are generally hard to implement for nonlinear problems
**Initial Value Problems — Crank–Nicholson Scheme (I)**

- Obtained by approximating the formal solution of the ODE
  \[ y_{n+1} = y_n + \int_{t_n}^{t_{n+1}} f(y, t) \, dt \] using the *Trapezoidal Quadrature*:

  \[ y_{n+1} = y_n + \frac{h}{2} [f(y_n, t_n) + f(y_{n+1}, t_{n+1})] \]

  The scheme is
  
  - locally *Third–Order* accurate
  
  - globally (over the interval \([t_0, t_0 + Nh]\)) *Second–Order* accurate

- Stability (for the model problem):
  
  \[ y_{n+1} = y_n + \frac{\lambda h}{2} (y_{n+1} + y_n) \implies y_{n+1} = \left(1 + \frac{\lambda h}{2} \right) \left(1 - \frac{\lambda h}{2} \right)^n y_0 \]

  \[ y_{n+1} = \left(1 + \frac{\lambda h}{2} \right)^n \Delta \sigma^n y_0 \implies \sigma = \frac{1 + \frac{\lambda h}{2}}{1 - \frac{\lambda h}{2}} \]

  \[ |\sigma| \leq 1 \implies \Re(\lambda h) \leq 0 \]

  **Stable** for all model ODEs with stable solutions
**INITIAL VALUE PROBLEMS — LEAPFROG SCHEME (I)**

- **LEAPFROG** as an example of a **TWO–STEP METHOD**:
  \[ y_{n+1} = y_{n-1} + 2h\lambda y_n \]

- **CHARACTERISTIC EQUATION** for the **AMPLIFICATION FACTOR** \((y_n = \sigma^n y_0)\):
  \[ \sigma^2 - 2h\lambda\sigma - 1 = 0 \]

  where roots give the amplification factors:
  \[
  \sigma_1 = \lambda h + \sqrt{1 + \lambda^2 h^2} \approx 1 + \lambda h + \frac{\lambda^2 h^2}{2} + \ldots = e^{\lambda h} + O(h^3)
  \]
  \[
  \sigma_2 = \lambda h - \sqrt{1 + \lambda^2 h^2} \approx -(1 - \lambda h + \frac{\lambda^2 h^2}{2} - \ldots) = -e^{-\lambda h} + O(h^3)
  \]

  Thus, the scheme is
  - locally **THIRD–ORDER** accurate
  - globally (over the interval \([t_0, t_0 + Nh]\)) **SECOND–ORDER** accurate
INITIAL VALUE PROBLEMS — LEAPFROG SCHEME (II)

- Stability for diffusion problems \( \lambda = \lambda_r \):

\[
\sigma_1 = \lambda h + \sqrt{1 + \lambda_r^2 h^2} > 1 \quad \text{for all} \quad h > 0
\]

Thus the scheme is **UNCONDITIONALLY UNSTABLE** for diffusion problems!

- Stability for advection problems \( \lambda = i\lambda_i \):

\[
\sigma_{1/2}^2 = 1 \quad (!!!) \quad \text{for} \quad h < \frac{1}{|\lambda_i|}
\]

Thus, the scheme is **CONDITIONALLY UNSTABLE** and **NON–DIFFUSIVE** for advection problems!

- **QUESTION** — analyze dispersive (i.e., related to \( \arg(\sigma) \)) errors of the leapfrog scheme.
INITIAL VALUE PROBLEMS — MULTISTEP PROCEDURES

• General form of a MULTISTEP PROCEDURE:

\[ \sum_{j=1}^{p} \alpha_j y_{n+j} = h \sum_{j=1}^{q} \beta_j f(y_{n+j}, t_{n+j}) \]

with characteristic polynomials

\[ \xi_p(z) = \alpha_p z^p + \alpha_{p-1} z^{p-1} + \cdots + \alpha_0 \]
\[ \zeta_q(z) = \beta_q z^q + \beta_{q-1} z^{q-1} + \cdots + \beta_0 \]

– if \( p > q \) — EXPLICIT SCHEME

– if \( p \leq q \) — IMPLICIT SCHEME

• A \((\xi, \zeta)\) –procedure converges uniformly in \([a, b]\), i.e.,

\[ \lim_{h \to 0} \max_{t_n \in [a, b]} |y_n - y(t_n)| = 0 \]

if:

– the following consistency conditions are verified: \( \xi(1) = 0 \) and \( \xi'(1) = \zeta'(1) \) (CONSISTENCY CONDITION)

– all roots of the polynomial \( \xi(z) \) are such that \( |z_i| \leq 1 \) and the roots with \( |z_k| = 1 \) are simple (STABILITY CONDITION)
INITIAL VALUE PROBLEMS — RUNGE–KUTTA METHODS (I)

- General form of a FRACTIONAL STEP METHOD:
  \[ y_{n+1} = y_n + \gamma_1 h k_1 + \gamma_2 h k_2 + \gamma_3 h k_3 + \ldots \]
  
  where
  \[ k_1 = f(y_n, t_n) \]
  \[ k_2 = f(y_n + \beta_1 h k_1, t_n + \alpha_1 h) \]
  \[ k_3 = f(y_n + \beta_2 h k_1 + \beta_3 h k_2, t_n + \alpha_2 h) \]

- Choose \( \gamma_i, \beta_i \) and \( \alpha_i \) to match as many expansion coefficients as possible in
  \[ y(t_{n+1}) = y(t_n) + h y'(t_n) + \frac{h^2}{2} y''(t_n) + \frac{h^3}{6} y'''(t_n) \ldots \]
  \[ y' = f \]
  \[ y'' = f_t + f f_y \]
  \[ y''' = f_{tt} + f_t f_y + 2 f f_{yt} + f^2 f_{yt} + f^2 f_{yy} \]

- Runge—Kutta methods are SELF–STARTING with fairly good stability and accuracy properties.
**Initial Value Problems — Runge–Kutta Methods (II)**

- **RK4** — an ODE “workhorse”:
  
  \[ y_{n+1} = y_n + \frac{h}{6} k_1 + \frac{h}{3} (k_2 + k_3) + \frac{h}{6} k_4 \]
  
  \[ k_1 = f(y_n, t_n) \]
  
  \[ k_2 = f(y_n + \frac{h}{2} k_1, t_n + \frac{1}{2}) \]
  
  \[ k_3 = f(y_n + \frac{h}{2} k_2, t_n + \frac{1}{2}) \]
  
  \[ k_4 = f(y_n + h k_3, t_{n+1}) \]

- The amplification factor:
  
  \[ \sigma = 1 + \lambda h + \frac{\lambda^2 h^2}{2} + \frac{\lambda^3 h^3}{6} + \frac{\lambda^4 h^4}{24} \]

  Thus, stability iff \(|\sigma| \leq 1\)

- **Accuracy**:
  
  \[ e^{\lambda h} = \sigma + O(h^5) \]

  Thus, the scheme is
  
  - locally **FIFTH–ORDER** accurate
  - globally (over the interval \([t_0, t_0 + Nh]\)) **FOURTH–ORDER** accurate
INITIAL VALUE PROBLEMS — Runge’s Principle

Let \((k + 1)\) be the order of the local truncation error; denote \(Y(t, h)\) an approximation of the exact solution \(y(t)\) computed with the step size \(h\); then at \(t = t_0 + 2nh\):

\[
y(t) - Y(t, h) \approx C 2n h^{k+1} = C(t - t_0) h^k
\]

\[
y(t) - Y(t, 2h) \approx C n (2h)^{k+1} = C(t - t_0) 2^k h^k
\]

Subtracting:

\[
Y(t, 2h) - Y(t, h) \approx C(t - t_0)(1 - 2^k) h^k
\]

Thus, we can obtain an estimate of the ABSOLUTE ERROR based on solution with two step–sizes only:

\[
y(t) - Y(t, h) \approx \frac{Y(t, h) - Y(t, 2h)}{2^k - 1}
\]

Runge’s principle is very useful for ADAPTIVE STEP SIZE REFINEMENT.
INITIAL VALUE PROBLEMS — LAX EQUIVALENCE THEOREM

- Consider an INITIAL VALUE PROBLEM

\[ \frac{du}{dt} = Lu \quad \text{with} \quad u(t_0) = u_0 \]

and assume that it is well-posed, i.e., it admits solutions which are unique and stable

- Consider a numerical method defined by a finite-difference operator \( C(h) \) such that the approximate solution is given by

\[ u_h(nh) = C(h)^n u_0, \quad n = 1, 2, \ldots \]

- The above method is CONSISTENT iff \( \frac{C(h)-I}{h} \) is a convergent approximation of the operator \( L \)

- LAX THEOREM — For a CONSISTENT difference method STABILITY is equivalent to CONVERGENCE

\[ ^a\text{For a more technical discussion, see § 5.2 in Atkinson & Han (2001)} \]
INITIAL VALUE PROBLEMS — CONSERVATION PROPERTIES (I)

- Is **Accuracy** and **Stability** all that matters?

- **Conservation Properties** — conservation by the numerical method (i.e., in the discrete sense) of various invariants the original equation may possess
  
  - **Remark** — conservation properties are particularly relevant for solution of Hamiltonian / hyperbolic systems

- Example — conservation of the solution norm:
  
  - In the continuous setting (assume $u = |u|e^{i\phi}$)
    
    $\frac{du}{dt} = i\lambda_i u \iff \begin{cases} 
    \frac{d|u|}{dt} = 0 \implies |u(t)| = |u_0|, \\
    \frac{d\phi}{dt} = \lambda_i,
  \end{cases}$

  - In the discrete setting: $|u_h(nh)| = |u_h((n-1)h)| = \cdots = |u_h(0)|$

  Necessary and sufficient condition for discrete conservation: $\exists h, \ |\sigma(h)| = 1$


**INITIAL VALUE PROBLEMS — CONSERVATION PROPERTIES (II)**

- **Implicit Euler —**

  \[
  |\sigma| = \left| \frac{1}{1 - i\lambda_i h} \right| = \frac{1}{\sqrt{1 + \lambda_i^2 h^2}} = 1 - \frac{1}{2} \lambda_i^2 h^2 + \cdots < 1 \quad \text{for all } h
  \]

  The scheme is thus **DISSIPATIVE** (i.e., not conservative)

- **Fourth–Order Runge–Kutta —**

  \[
  |\sigma| = \left| 1 + i\lambda_i h - \frac{\lambda_i^2 h^2}{2} - i\frac{\lambda_i^3 h^3}{6} + \frac{\lambda_i^4 h^4}{24} \right| = \frac{1}{24} \sqrt{576 - 8\lambda_i^6 h^6 + \lambda_i^8 h^8}
  \]

  \[
  = 1 - \frac{1}{144} \lambda_i^6 h^6 + \cdots < 1 \quad \text{for small } h
  \]

  The scheme is thus **DISSIPATIVE** (i.e., not conservative)

- **Leapfrog —**

  \[|\sigma_{1/2}| \equiv 1 \quad \text{for all } h < \frac{1}{|\lambda_i|}\]

  The scheme is thus **CONSERVATIVE** for all time–steps for which it is stable!!! Leapfrog is an example of a **SYMPLECTIC INTEGRATOR** which are designed to have good conservation properties.
**Finite Differences for PDEs Review**

- Classification of linear PDEs in 2D: consider $u : \Omega^2 \to \mathbb{R}$ and $A, B, C \in \mathbb{R}$ such that
  
  $$A \frac{\partial^2 u}{\partial x^2} + B \frac{\partial^2 u}{\partial x \partial y} + C \frac{\partial^2 u}{\partial y^2} + f(x, y, u) = 0$$

- **Elliptic Problems**: $B^2 - 4AC < 0$
  - Poisson equation:
    $$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = g(x, y)$$

- **Parabolic Problems**: $B^2 - 4AC = 0$
  - Heat equation:
    $$\frac{\partial u}{\partial t} = a \left( \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) + g(x, y)$$

- **Hyperbolic Problems**: $B^2 - 4AC > 0$
  - Wave equation:
    $$\frac{\partial^2 u}{\partial t^2} = a \left( \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) + g(x, y)$$
Finite Differences for PDEs
Elliptic Problems

- See Homework Assignment # 1 ...
**Finite Differences for PDEs**

**Parabolic Problems**

\[
\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2} \quad (I)
\]

- **Crank–Nicholson Method** \((x_j = j\Delta x, j = 1, \ldots, M, t = n\Delta t, \quad n = 1, \ldots, N)\):
  - **Spatial derivative:**
    \[
    \left( \frac{\partial^2 u}{\partial x^2} \right)_j^n = \frac{u_{j+1}^n - 2u_j^n + u_{j-1}^n}{(\Delta x)^2} + O((\Delta x)^2)
    \]
  - **Time derivative:**
    \[
    \left( \frac{\partial u}{\partial t} \right)_j^{n+1} = \frac{u_{j+1}^{n+1} - u_j^n}{\Delta t} + O(\Delta t) = \frac{1}{2} \left[ \left( \frac{\partial^2 u}{\partial x^2} \right)_j^{n+1} + \left( \frac{\partial^2 u}{\partial x^2} \right)_j^n \right] + O((\Delta t)^2)
    \]
    \[
    u_{j+1}^{n+1} - u_j^n = \frac{\Delta t}{2(\Delta x)^2} \left( u_{j+1}^{n+1} - 2u_j^{n+1} + u_{j-1}^{n+1} + u_j^n - 2u_j^n + u_{j-1}^n \right) + O((\Delta x)^2 + (\Delta t)^2)
    \]
  - **Thus, defining** \(r = \frac{\Delta t}{(\Delta x)^2}\), **we have at every time step** \(n\)
    \[
    -ru_{j+1}^{n+1} + 2(1+r)u_j^{n+1} - ru_{j-1}^{n+1} = ru_{j+1}^n + 2(1-r)u_j^n + ru_{j-1}^n
    \]
    which for \(U^n = [u_1^n, \ldots, u_M^n]^T\) can be written as an algebraic system
    \[
    (2\mathbb{I} - A)U^{n+1} = (2\mathbb{I} + A)U^n, \text{ where } A \text{ is a tridiagonal matrix}
    \]
FINITE DIFFERENCES FOR PDES
PARABOLIC PROBLEMS $\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2}$ (II)

• **θ Method**
  - allow for a more general approximation in time of the RHS ($\theta \in [0,1]$)

\[
\left( \frac{\partial u}{\partial t} \right)_j^{n+1} = \frac{u_j^{n+1} - u_j^n}{\Delta t} + O(\Delta t) = \frac{1}{2} \left[ \theta \left( \frac{\partial^2 u}{\partial x^2} \right)_j^{n+1} + (1 - \theta) \left( \frac{\partial^2 u}{\partial x^2} \right)_j^n \right] + O(\Delta t)
\]

  - special cases

* $\theta = 0 \implies$ **Explicit method**: $U_j^{n+1} = A_0 U^n_j$
* $\theta = \frac{1}{2} \implies$ **Crank–Nicolson method** (see previous slide)
* $\theta = 1 \implies$ **Implicit method**: $A_1 U_j^{n+1} = U^n_j$

• **Stability**:
  - The **Explicit Scheme** is **stable** for $r = \frac{\Delta t}{2(\Delta x)^2} < \frac{1}{2}$
  - The **Crank–Nicolson** and **Implicit Scheme** are **stable** for all $r$
Finite Differences for Differential Equations

**Finite Differences for PDEs**

### Hyperbolic Problems

\[ \frac{\partial^2 u}{\partial t^2} = \frac{\partial^2 u}{\partial x^2} \] (I)

- **Spatial derivative:**
  \[
  \left( \frac{\partial^2 u}{\partial x^2} \right)_j^n = \frac{u_{j+1}^n - 2u_j^n + u_{j-1}^n}{(\Delta x)^2} + O((\Delta t)^2)
  \]

- **Time derivative:**
  \[
  \left( \frac{\partial^2 u}{\partial t^2} \right)_j^n = \frac{u_{j+1}^{n+1} - 2u_j^{n+1} + u_{j-1}^{n-1}}{(\Delta t)^2} + O((\Delta t)^2) = \left( \frac{\partial^2 u}{\partial x^2} \right)_j^n + O((\Delta t)^3)
  \]

  \[
  u_j^{n+1} = \frac{(\Delta t)^2}{(\Delta x)^2} (u_{j+1}^n + u_{j-1}^n) - u_j^{n-1} + 2 \left( 1 - \frac{(\Delta t)^2}{(\Delta x)^2} \right) u_j^n + O((\Delta x)^2 + (\Delta t)^4)
  \]

- **Stability for** \( \frac{(\Delta t)^2}{(\Delta x)^2} \leq 1 \)

- **Remark:** need two initial conditions!