Chapter 5 Finite Difference Methods

YORK UNIVERSITÉ UNIVERSITY

redefine THE POSSIBLE.

YORK
UNIVERSITÉ
UNIVERSITY

References

1. Chapters 5 and 9, Brandimarte
2. Section 17.8, Hull
3. Chapter 7, “Numerical analysis”, Burden and Faires

YORK
UNIVERSITÉ
UNIVERSITY

Outline

- Finite difference (FD) approximation to the derivatives
- Explicit FD method
- Numerical issues
- Implicit FD method
- Crank-Nicolson method
- Dealing with American options
- Further comments

Chapter 5 Finite Difference Methods

5.1 Finite difference approximations

YORK UNIVERSITÉ UNIVERSITY

redefine THE POSSIBLE.

YORK UNIVERSITÉ UNIVERSITY

Finite-difference mesh

- Aim to approximate the values of the continuous function *f*(t, S) on a set of discrete points in (t, S) plane
- Divide the S-axis into equally spaced nodes at distance ΔS apart, and, the t-axis into equally spaced nodes a distance Δt apart
- (t, S) plane becomes a mesh with mesh points on

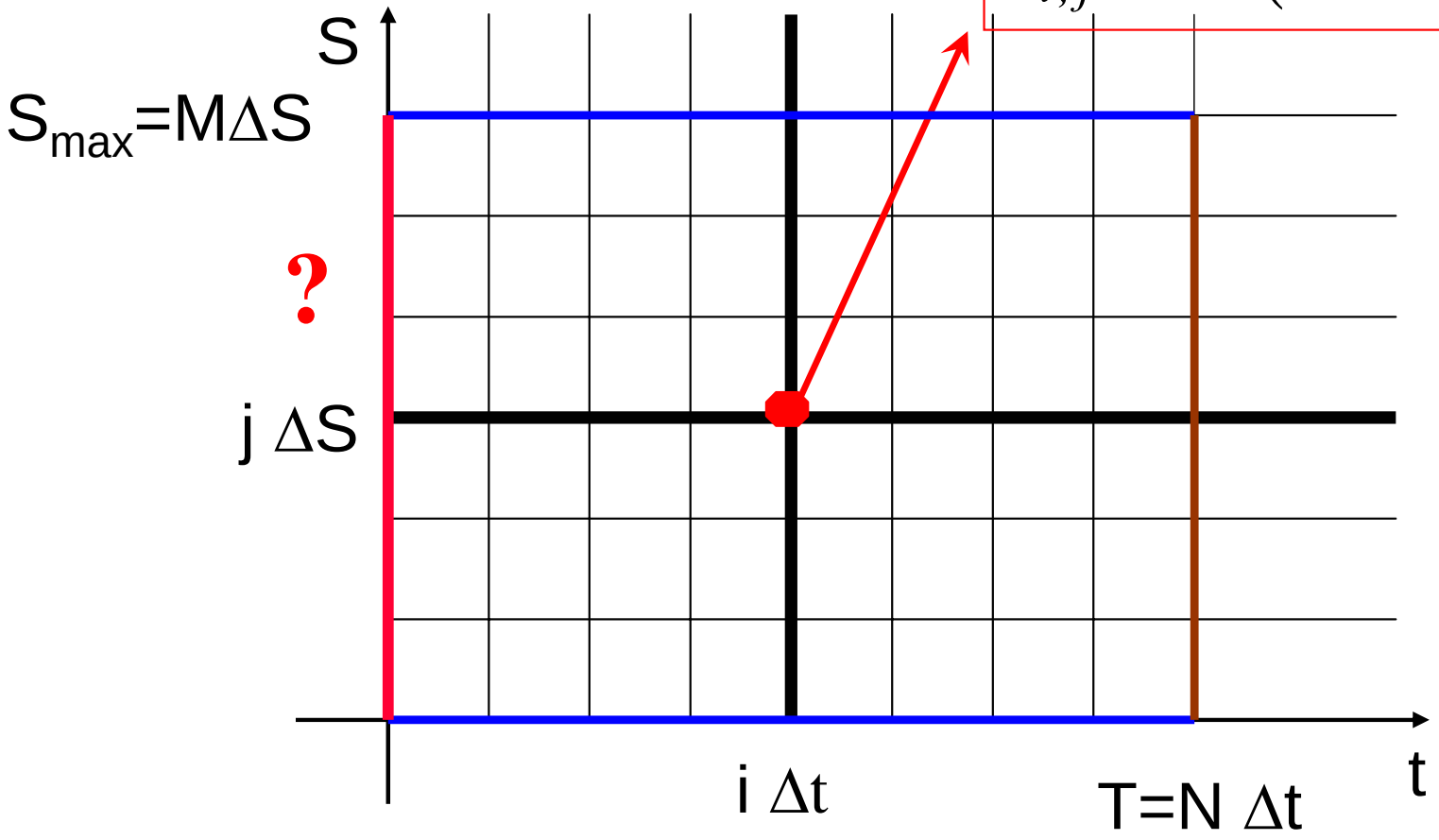
  (i Δt, jΔS)
- We are interested in the values of *f*(t, S) at mesh points

  (i Δt, jΔS), denoted as

$$f_{i,j} = f\left(i\Delta t, j\Delta S\right)$$

The mesh for finite-difference approximation

$$f_{i,j} = f\left(i\Delta t, j\Delta S\right)$$



YORK UNIVERSITÉ UNIVERSITY

Black-Scholes Equation for a European option with value V(S,t)

$$\frac{\partial V}{\partial t}+\frac{1}{2}\sigma^2 S^2\frac{\partial^2 V}{\partial^2 S}+rS\frac{\partial V}{\partial S}-rV=0 \qquad (5.1)$$

where $0 < S < +\infty$ and $0 \le t < T$

with proper final and boundary conditions

Notes:

This is a second-order hyperbolic, elliptic, or parabolic, forward or backward partial differential equation

Its solution is sufficiently well behaved ,i.e. well-posed

YORK
UNIVERSITÉ
UNIVERSITY

Finite difference approximations

The basic idea of FDM is to replace the partial derivatives by approximations obtained by Taylor expansions near the point of interests

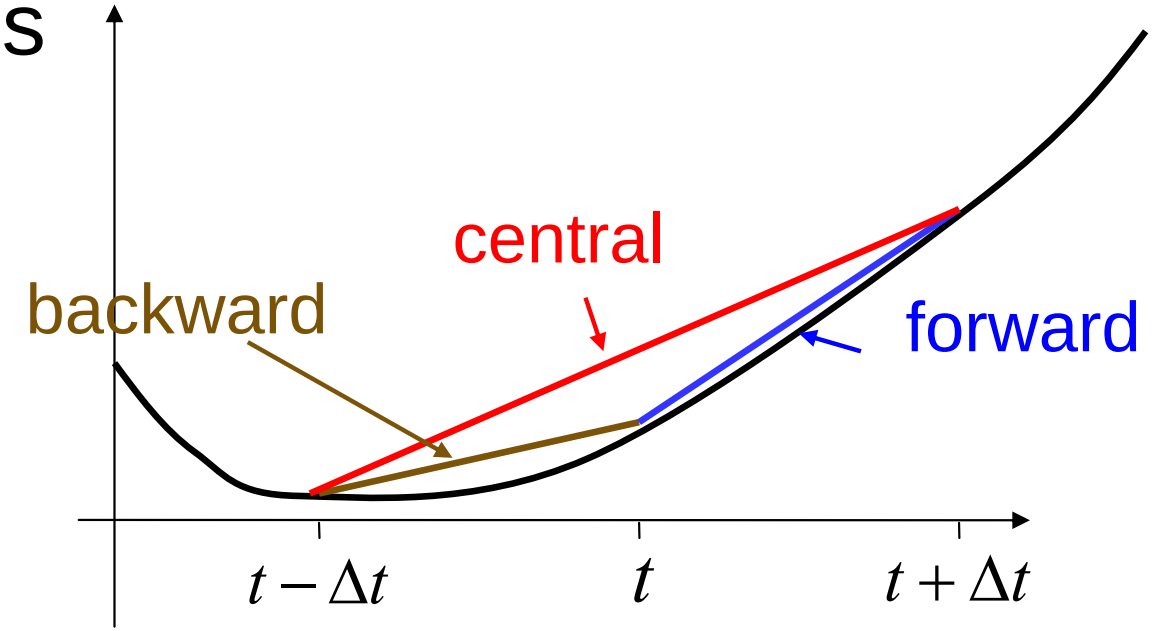
For example,

$$\frac{\partial f(S,t)}{\partial t} = \lim_{\Delta t \to 0} \frac{f(S,t+\Delta t) - f(S,t)}{\Delta t} \approx \frac{f(S,t+\Delta t) - f(S,t)}{\Delta t}$$

for small Δt, using Taylor expansion at point (S,t)

$$f(S,t+\Delta t) = f(S,t) + \frac{\partial f(S,t)}{\partial t}\Delta t + O\left((\Delta t)^2\right)$$

Forward-, Backward-, and Central-difference approximation to 1st order derivatives



Forward: $$\frac{\partial f(t,S)}{\partial t} \approx \frac{f(t+\Delta t,S)-f(t,S)}{\Delta t} + O(\Delta t)$$

Backward: $$\frac{\partial f(t,S)}{\partial t} \approx \frac{f(t,S)-f(t-\Delta t,S)}{\Delta t} + O(\Delta t)$$

Central: $$\frac{\partial f(t,S)}{\partial t} \approx \frac{f(t+\Delta t,S)-f(t-\Delta t,S)}{2\Delta t} + O\left((\Delta t)^2\right)$$

YORK
UNIVERSITÉ
UNIVERSITY

Symmetric Central-difference approximations to 2nd order derivatives

$$\frac{\partial^2 f(t,S)}{\partial S^2} \approx \frac{f(t,S+\Delta S)-2f(t,S)+f(t,S-\Delta S)}{(\Delta S)^2}+O\left((\Delta S)^2\right)$$

Use Taylor's expansions for $f(t,S+\Delta S)$ and $f(t,S-\Delta S)$ around point (t,S):

$f(t,S+\Delta S)=?$

+

$f(t,S-\Delta S)=?$

YORK UNIVERSITÉ UNIVERSITY

Finite difference approximations

Forward Difference: $\frac{\partial f}{\partial t} \approx \frac{f_{i+1,j} - f_{i,j}}{\Delta t}, \quad \frac{\partial f}{\partial S} \approx \frac{f_{i,j+1} - f_{i,j}}{\Delta S}$

Backward Difference: $\frac{\partial f}{\partial t} \approx \frac{f_{i,j} - f_{i-1,j}}{\Delta t}, \quad \frac{\partial f}{\partial S} \approx \frac{f_{i,j} - f_{i,j-1}}{\Delta S}$

Central Difference: $\frac{\partial f}{\partial t} \approx \frac{f_{i+1,j} - f_{i-1,j}}{2\Delta t}, \quad \frac{\partial f}{\partial S} \approx \frac{f_{i,j+1} - f_{i,j-1}}{2\Delta S}$

As to the second derivative, we have:

$$\frac{\partial^2 f}{\partial S^2} \approx \left(\frac{f_{i,j+1} - f_{i,j}}{\Delta S} - \frac{f_{i,j} - f_{i,j-1}}{\Delta S} \right) \Big/ \Delta S$$

$$= \frac{f_{i,j+1} - 2f_{i,j} + f_{i,j-1}}{(\Delta S)^2}$$

Finite difference approximations

- Depending on which combination of schemes we use in discretizing the equation, we will have explicit, implicit, or Crank-Nicolson methods
- We also need to discretize the boundary and final conditions accordingly. For example, for European Call,

Final Condition:

$$f_{N,j} = max\left(j\Delta S - K, 0\right), \quad \text{for } j = 0,1,\ldots,M$$

Boundary Conditions:

$$\begin{cases} f_{i,0} = 0 \\ f_{i,M} = S_{max} - Ke^{-r(N-i)\Delta t} \end{cases}, \quad \text{for } i = 0,1,\ldots,N$$

where $S_{max} = M\Delta S$.

Chapter 5 Finite Difference Methods

5.2.1 Explicit Finite-Difference Method

YORK UNIVERSITÉ UNIVERSITY

redefine THE POSSIBLE.

YORK UNIVERSITÉ UNIVERSITY

Explicit Finite Difference Methods

In $\frac{\partial f}{\partial t} + rS\frac{\partial f}{\partial S} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 f}{\partial S^2} = rf$, at point $(i\Delta t,\ j\Delta S)$, set

backward difference: $\frac{\partial f}{\partial t} \approx \frac{f_{i,j} - f_{i-1,j}}{\Delta t}$

central difference: $\frac{\partial f}{\partial S} \approx \frac{f_{i,j+1} - f_{i,j-1}}{2\Delta S}$,

and

$$\frac{\partial^2 f}{\partial S^2} \approx \frac{f_{i,j+1} + f_{i,j-1} - 2f_{i,j}}{\Delta S^2}, \quad rf = rf_{i,j}\ ,\ S = j\Delta S$$

YORK UNIVERSITÉ UNIVERSITY

Explicit Finite Difference Methods

Rewriting the equation, we get an explicit scheme:

$$f_{i-1,j} = a_j^* f_{i,j-1} + b_j^* f_{i,j} + c_j^* f_{i,j+1} \qquad (5.2)$$

where

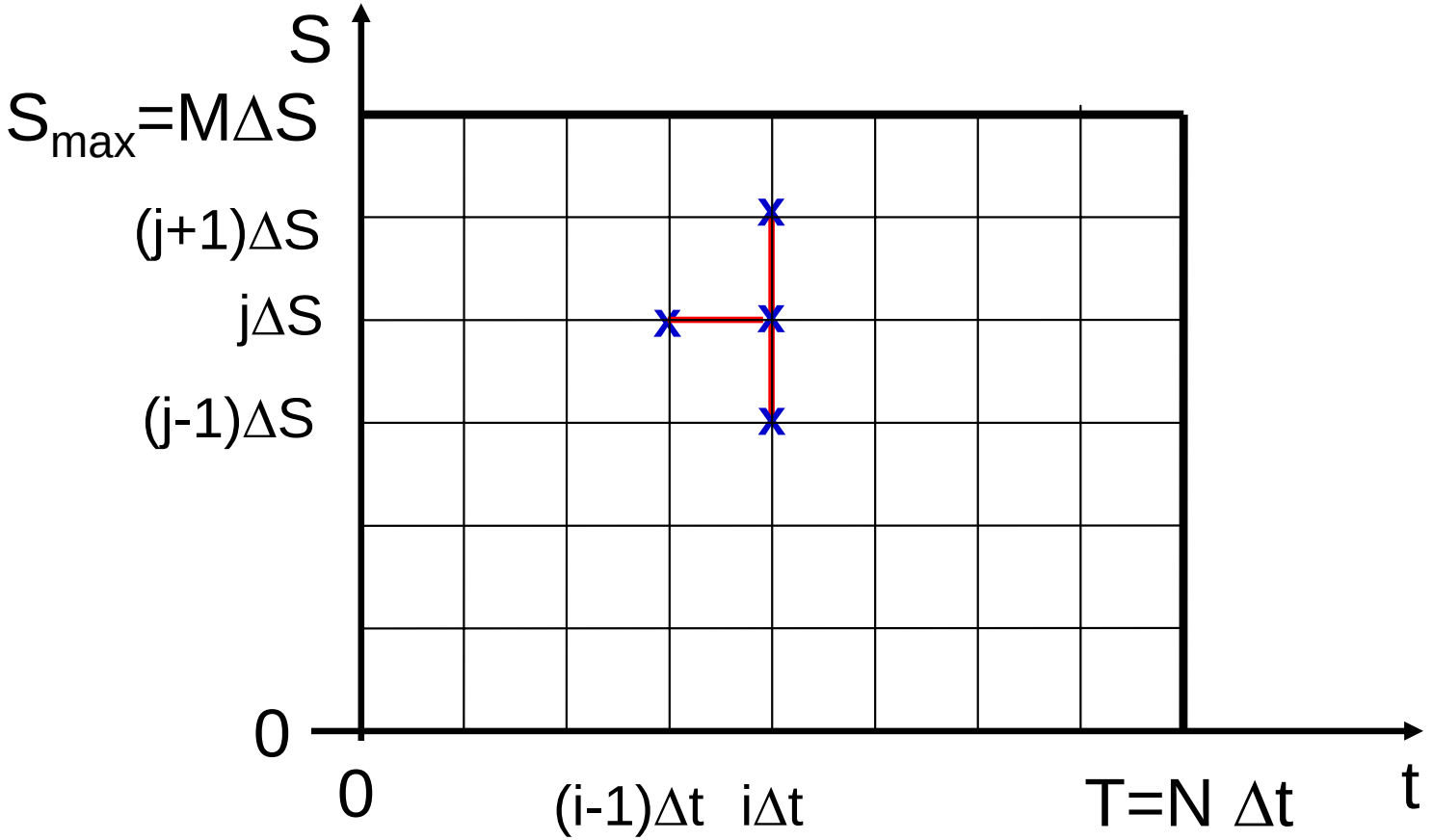
$$a_j^* = \frac{1}{2}\Delta t\left(\sigma^2 j^2 - rj\right)$$

$$b_j^* = 1 - \Delta t\left(\sigma^2 j^2 + r\right)$$

$$c_j^* = \frac{1}{2}\Delta t\left(\sigma^2 j^2 + rj\right)$$

for $i = N-1, N-2, \ldots, 1, 0$ and $j = 1, 2, \ldots, M-1$.

Numerical Computation Dependency



YORK
UNIVERSITÉ
UNIVERSITY

Implementation

1. Starting with the final values $f_{N,j}$, we apply (5.2) to solve $f_{N-1,j}$ for $1 \le j \le M-1$. We use the boundary condition to determine $f_{N-1,0}$ and $f_{N\text{-}1,M}$.

2. Repeat the process to determine $f_{N-2,j}$ and so on

YORK UNIVERSITÉ UNIVERSITY

Example

We compare explicit finite difference solution for a European put with the exact Black-Scholes formula, where $T = 5/12$ yr, S_0=\$50, K = \$50, σ=30%, r = 10%.

Black-Scholes Price: \$2.8446

EFD Method with S_{max}=\$100, $\Delta S=2$, $\Delta t=5/1200$: \$2.8288

EFD Method with S_{max}=\$100, $\Delta S=1$, $\Delta t=5/4800$: \$2.8406

YORK UNIVERSITÉ UNIVERSITY

Example (Stability)

We compare explicit finite difference solution for a European put with the exact Black-Scholes formula, where T = 5/12 yr, S_0=\$50, K = \$50, σ=30%, r = 10%.

Black-Scholes Price: \$2.8446

EFD Method with S_{max}=\$100, ΔS=2, Δt=5/1200: \$2.8288

EFD Method with S_{max}=\$100, ΔS=1.5, Δt=5/1200: \$3.1414

EFD Method with S_{max}=\$100, ΔS=1, Δt=5/1200: -\$2.8271E22

Chapter 5 Finite Difference Methods

5.2.2 Numerical Stability

YORK UNIVERSITÉ UNIVERSITY

redefine THE POSSIBLE.

YORK
UNIVERSITÉ
UNIVERSITY

Numerical Accuracy

- The problem itself
- The discretization scheme used
- The numerical algorithm used

YORK UNIVERSITÉ UNIVERSITY

Conditioning Issue

Suppose we have mathematically posed problem:

$$y = f(x)$$

where y is to evaluated given an input x.

Let $x^* = x + \delta x$ for small change δx.

If hen $f(x^*)$ is near $f(x)$, then we call the problem is $well\text{-}conditioned$. Otherwise, it is ill-posed/ill-conditioned.

YORK UNIVERSITÉ UNIVERSITY

Conditioning Issue

- Conditioning issue is related to the problem itself, not to the specific numerical algorithm; Stability issue is related to the numerical algorithm
- One can not expect a good numerical algorithm to solve an ill-conditioned problem any more accurately than the data warrant
- But a bad numerical algorithm can produce poor solutions even to well-conditioned problems

YORK UNIVERSITÉ UNIVERSITY

Conditional Issue

The concept "near" can be measured by further information about the particular problem:

$$\frac{\left\| f(x) - f(x^*) \right\|}{\left\| f(x) \right\|} \le C \frac{\left\| \delta x \right\|}{\left\| x \right\|} \qquad \left(f(x) \neq 0 \right)$$

where C is called *condition number* of this problem. If C is large, the problem is ill-conditioned.

YORK UNIVERSITÉ UNIVERSITY

Floating Point Number & Error

Let x be any real number.

Infinite decimal expansion : $x = \pm .x_1 x_2 \cdots x_d \cdots 10^e$

Truncated floating point number : $x \approx fl(x) = \pm .x_1 x_2 \cdots x_d 10^e$

where $x_1 \neq 0, 0 \leq x_i \leq 9,$

d : an integer, precision of the floating point system

e : an bounded integer

Floating point or **roundoff error** : $fl(x) - x$

YORK UNIVERSITÉ UNIVERSITY

Error Propagation

When additional calculations are done, there is an accumulation of these floating point errors.

Example : Let $x = -0.6667$ and $fl(x) = -0.667\ 10^0$ where $d = 3$.

Floating point error : $fl(x) - x = -0.0003$

Error propagation : $fl(x)^2 - x^2 = 0.00040011$

YORK UNIVERSITÉ UNIVERSITY

Numerical Stability or Instability

Stability ensures if the error between the numerical soltuion and the exact solution remains bounded as numerical computation progresses.

That is, $f\left(x^*\right)$(the solution of a slightly perturbed problem) is near $f^*\left(x\right)$(the computed solution).

Stability concerns about the behavior of $\left|f_{i,j} - f\left(i\Delta t, j\Delta S\right)\right|$ as numerical computation progresses for fixed discretization steps Δt and ΔS.

YORK UNIVERSITÉ UNIVERSITY

Convergence issue

Convergence of the numerical algorithm concerns about the behavior of $\left| f_{i,j} - f\left(i\Delta t, j\Delta S\right)\right|$ as $\Delta t, \Delta S \to 0$ for fixed values $\left(i\Delta t, j\Delta S\right)$.

For well-posed linear initial value problem,

Stability $\Leftrightarrow$ Convergence

(Lax's equivalence theorem, Richtmyer and Morton, "Difference Methods for Initial Value Problems" (2nd) 1967)

Numerical Accuracy

- These factors contribute the accuracy of a numerical solution. We can find a “good” estimate if our problem is well-conditioned and the algorithm is stable

$$\left.\begin{array}{l}\text{Stable: } f^*(x) \approx f(x^*) \\ \text{Well-conditioned: } f(x^*) \approx f(x)\end{array}\right\} f^*(x) \approx f(x)$$

Chapter 5 Finite Difference Methods

5.2.3 Financial Interpretation of Numerical Instability

YORK UNIVERSITÉ UNIVERSITY

redefine THE POSSIBLE.

YORK UNIVERSITÉ UNIVERSITY

Financial Interpretation of instability (Hall, page 423-4)

If $\partial f / \partial S$ and $\partial^2 f / \partial S^2$ are assumed to be the same at $(i+1, j)$ as they are at (i, j), we obtain equations of the form:

$$f_{i,j} = \hat{a}_j f_{i+1,j-1} + \hat{b}_j f_{i+1,j} + \hat{c}_j f_{i+1,j+1} \qquad (5.3)$$

where

$$\hat{a}_j = \frac{1}{1+r\Delta t}\left(\frac{1}{2}\Delta t\sigma^2 j^2 - \frac{1}{2}\Delta t\, r\, j\right) = \frac{1}{1+r\Delta t}\pi_d$$

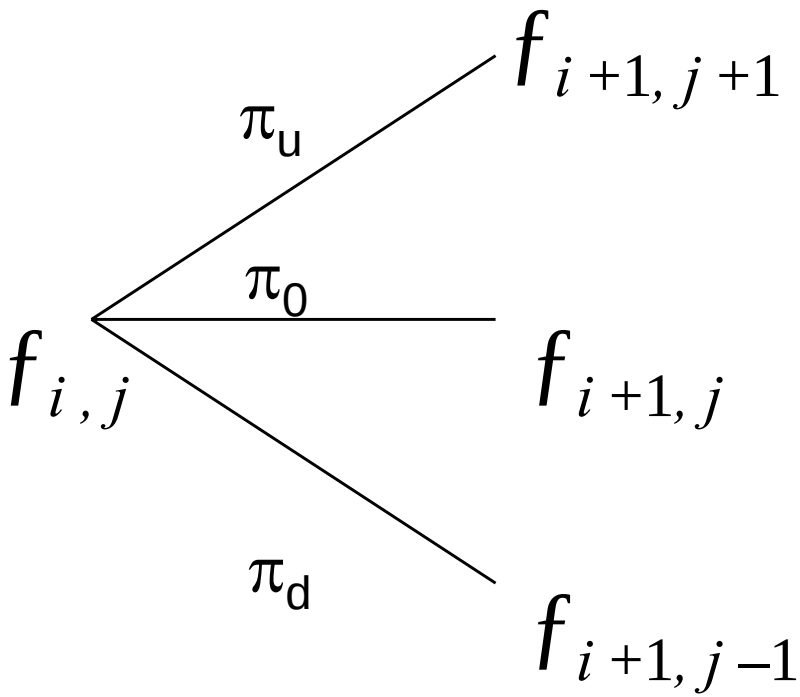
$$\hat{b}_j = \frac{1}{1+r\Delta t}\left(1-\sigma^2 j^2 \Delta t\right) = \frac{1}{1+r\Delta t}\pi_0$$

$$\hat{c}_j = \frac{1}{1+r\Delta t}\left(\frac{1}{2}\Delta t\sigma^2 j^2 + \frac{1}{2}\Delta t\, r\, j\right) = \frac{1}{1+r\Delta t}\pi_u$$

for $i = N-1,\ N-2,\ \ldots,\ 1,\ 0$ and $j = 1,\ 2,\ \ldots,\ M-1$.

YORK UNIVERSITÉ UNIVERSITY

Explicit Finite Difference Methods



These coefficients can be interpreted as probabilities times a discount factor. If one of these probability < 0, instability occurs.

YORK UNIVERSITÉ UNIVERSITY

Explicit Finite Difference Method as Trinomial Tree

Check if the mean and variance of the

Expected value of the increase in asset price during Δt:

$$\mathrm{E}[\Delta] = -\Delta S\,\pi_d + 0\pi_0 + \Delta S\,\pi_u = rj\Delta S\,\Delta t = rS\Delta t$$

Variance of the increment:

$$\mathrm{E}\left[\Delta^2\right] = -(\Delta S)^2\,\pi_d + 0\pi_0 + (\Delta S)^2\,\pi_u = \sigma^2 j^2 (\Delta S)^2 \Delta t = \sigma^2 S^2 \Delta t$$

$$\mathrm{Var}[\Delta] = \mathrm{E}\left[\Delta^2\right] - \mathrm{E}^2[\Delta] = \sigma^2 S^2 \Delta t - r^2 S^2 (\Delta t)^2 \approx \sigma^2 S^2 \Delta t$$

which is coherent with geometric Brownian motion in a risk-neutral world

YORK UNIVERSITÉ UNIVERSITY

Change of Variable

Define $Z = ln\, S$. The B-S equation becomes

$$\frac{\partial f}{\partial t} + \left(r - \frac{\sigma^2}{2} \right) \frac{\partial f}{\partial Z} + \frac{\sigma^2}{2} \frac{\partial^2 f}{\partial Z^2} = rf$$

The corresponding difference equation is

$$\frac{f_{i+1,j} - f_{i,j}}{\Delta t} + \left(r - \frac{\sigma^2}{2} \right) \frac{f_{i+1,j+1} - f_{i+1,j-1}}{2\Delta Z} + \frac{\sigma^2}{2} \frac{f_{i+1,j+1} - 2f_{i+1,j-1} + f_{i+1,j+1}}{\Delta Z^2} = rf_{i,j}$$

or

$$f_{i,j} = \alpha_j^* f_{i+1,j-1} + \beta_j^* f_{i+1,j} + \gamma_j^* f_{i+1,j+1} \qquad (5.4)$$

YORK UNIVERSITÉ UNIVERSITY

Change of Variable

where

$$\alpha_j^* = \frac{1}{1+r\Delta t}\left[-\left(r-\frac{\sigma^2}{2}\right)\frac{\Delta t}{2\Delta Z}+\frac{\sigma^2}{2}\frac{\Delta t}{\Delta Z^2}\right]$$

$$\beta_j^* = \frac{1}{1+r\Delta t}\left(1-\frac{\Delta t}{\Delta Z^2}\sigma^2\right)$$

$$\gamma_j^* = \frac{1}{1+r\Delta t}\left[\left(r-\frac{\sigma^2}{2}\right)\frac{\Delta t}{2\Delta Z}+\frac{\sigma^2}{2}\frac{\Delta t}{\Delta Z^2}\right]$$

It can be shown that it is numerically most efficient if $\Delta Z = \sigma\sqrt{3\Delta t}$.

YORK UNIVERSITÉ UNIVERSITY

Reduced to Heat Equation

Get rid of the varying coefficients S and S² by using change of variables:

$$S = Ee^{x}, t = T - 2\tau/\sigma^{2}, V(S,t) = Ee^{-\frac{1}{2}(k-1)x-\frac{1}{4}(k+1)^{2}\tau}u(x,\tau)$$

$$k = r\Big/\left(\frac{1}{2}\sigma^{2}\right)$$

Equation (5.1) becomes heat equation (5.5):

$$\frac{\partial u}{\partial \tau} = \frac{\partial^{2} u}{\partial x^{2}} \qquad (5.5)$$

for $-\infty < x < +\infty$ and $\tau > 0$

YORK UNIVERSITÉ UNIVERSITY

Explicit Finite Difference Method

With $u_n^m = u(n\delta x, m\delta\tau)$, this involves solving a system of finite difference equations of the form :

$$\frac{u_n^{m+1} - u_n^m}{\delta\tau} + O(\delta\tau) = \frac{u_{n+1}^m - 2u_n^m + u_{n-1}^m}{(\delta x)^2} + O\left((\delta x)^2\right)$$

Ignoring terms of $O(\delta\tau)$ and $O\left((\delta x)^2\right)$, we can approximat e this by

$$u_n^{m+1} = \alpha u_{n+1}^m + (1-2\alpha)u_n^m + \alpha u_{n-1}^m$$

where $\alpha = \dfrac{\delta\tau}{(\delta x)^2}$, for $-N^- \le n \le N^+$ and $m = 0,1,\ldots M$ $\left(= \dfrac{\frac{1}{2}\sigma^2 T}{\delta\tau}\right)$

YORK UNIVERSITÉ UNIVERSITY

Stability and Convergence

(P. Wilmott, et al, Option Pricing)

Stability:

The solution of Eqn (5.5) is

i) Stable if $0 < \alpha = \frac{\delta\tau}{(\delta x)^2} \le \frac{1}{2}$; ii) Unstable if $\alpha > \frac{1}{2}$

Convergence:

If $0 < \alpha \le \frac{1}{2}$, then the explicit finite-difference approximation converges to the exact solution as $\delta\tau, \delta x \to 0$

(in the sense that $u_n^m \to u(n\delta x, m\delta\tau)$ as $\delta\tau, \delta x \to 0$)

Rate of Convergence is $O(\delta\tau)$

Chapter 5 Finite Difference Methods

5.3.1 Implicit Finite-Difference Method

YORK UNIVERSITÉ UNIVERSITY

redefine THE POSSIBLE.

YORK UNIVERSITÉ UNIVERSITY

Implicit Finite Difference Methods

In $\frac{\partial f}{\partial t} + rS\frac{\partial f}{\partial S} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 f}{\partial S^2} = r f$, we use

forward difference: $\frac{\partial f}{\partial t} \approx \frac{f_{i+1,\mathrm{j}} - f_{i,j}}{\Delta t}$

central difference: $\frac{\partial f}{\partial S} \approx \frac{f_{i,j+1} - f_{i,j-1}}{2\Delta S}$,

and

$$\frac{\partial^2 f}{\partial S^2} \approx \frac{f_{i,j+1} + f_{i,j-1} - 2f_{i,j}}{\Delta S^2}, \quad rf = rf_{i,j}$$

YORK UNIVERSITÉ UNIVERSITY

Implicit Finite Difference Methods

Rewriting the equation, we get an implicit scheme:

$$a_j f_{i,j-1} + b_j f_{i,j} + c_j f_{i,j+1} = f_{i+1,j} \qquad (5.6)$$

where

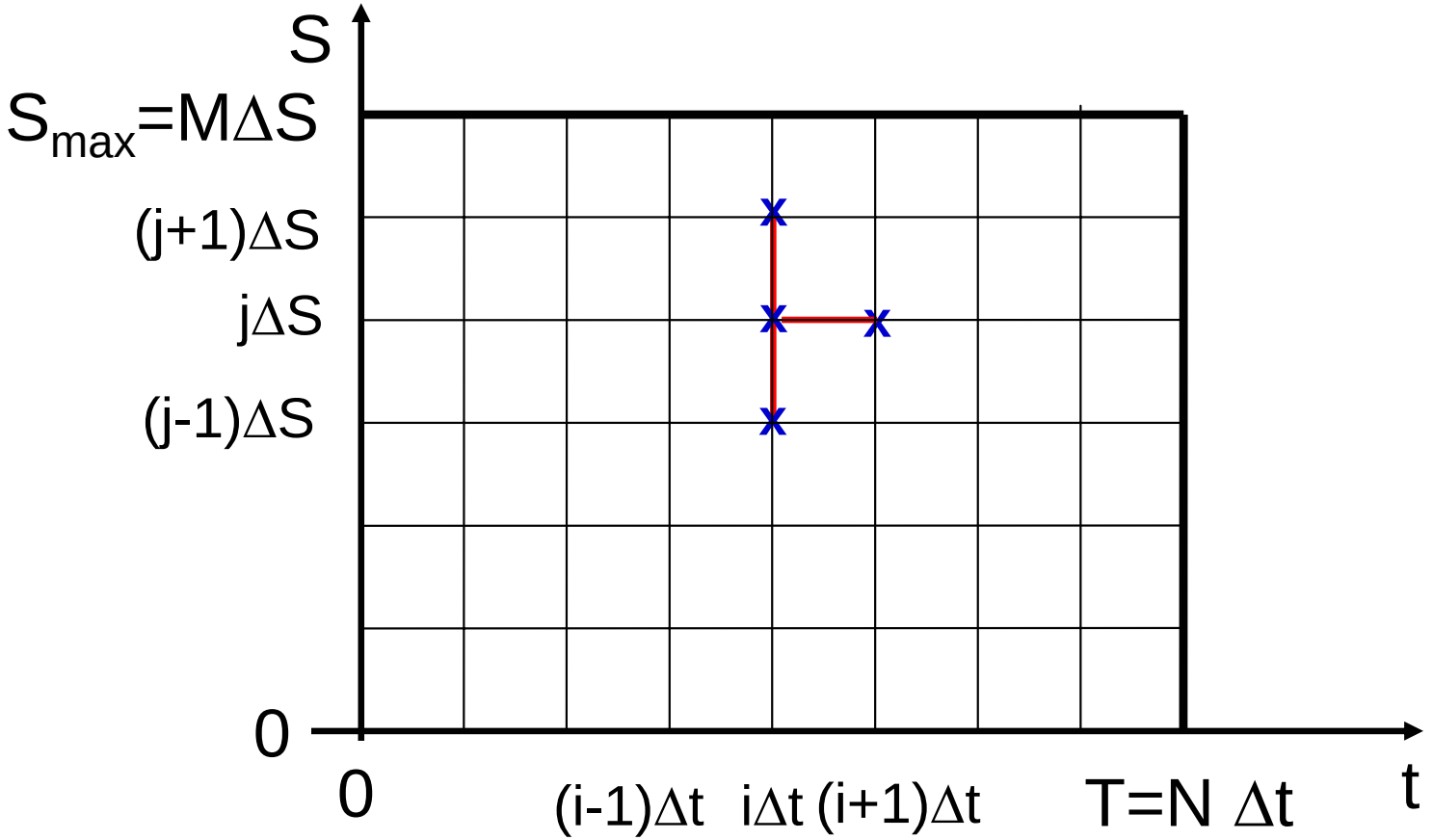
$$a_j = \frac{1}{2}\Delta t\left(-\sigma^2 j^2 + rj\right)$$

$$b_j = 1 + \Delta t\left(\sigma^2 j^2 + r\right)$$

$$c_j = -\frac{1}{2}\Delta t\left(\sigma^2 j^2 + rj\right)$$

for $i = N-1,\ N-2,\ \ldots,\ 1,\ 0$ and $j = 1,\ 2,\ \ldots,\ M-1$.

Numerical Computation Dependency



Implementation

Equation (5.6) can be rewritten in matrix form:

$$\mathbf{C}\mathbf{f}_i = \mathbf{f}_{i+1} + \mathbf{b}_i \qquad (5.7)$$

where $\mathbf{f}_i$ and $\mathbf{b}_i$ are $(M-1)$ dimensional vectors

$$\mathbf{f}_i = \left[f_{i,1}, f_{i,2}, f_{i,3} \cdots, f_{i,M-1} \right]^T, \mathbf{b}_i = \left[-a_1 f_{i,0}, 0, 0, \cdots, 0, -c_{M-1} f_{i,M} \right]^T$$

and $\mathbf{C}$ is $(M-1)\times(M-1)$ symmetric matrices

$$\mathbf{C} = \begin{bmatrix} b_1 & c_1 & 0 & \cdots & 0 \\ a_2 & b_2 & c_2 & \ddots & 0 \\ 0 & a_3 & b_3 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & c_{M-2} \\ 0 & \cdots & 0 & a_{M-1} & b_{M-1} \end{bmatrix}$$

YORK
UNIVERSITÉ
UNIVERSITY

Implementation

1. Starting with the final values $f_{N,j}$, we need to solve a linear system (5.7) to obtain $f_{N-1,j}$ for $1 \le j \le M-1$ using LU factorization or iterative methods. We use the boundary condition to determine $f_{N-1,0}$ and $f_{N\text{-}1,M}$.

2. Repeat the process to determine $f_{N-2,j}$ and so on

YORK UNIVERSITÉ UNIVERSITY

Example

We compare implicit finite difference solution for a European put with the exact Black-Scholes formula, where T = 5/12 yr, S_0=\$50, K = \$50, σ=30%, r = 10%.

Black-Scholes Price: \$2.8446

IFD Method with S_{max}=\$100, ΔS=2, Δt=5/1200: \$2.8194

IFD Method with S_{max}=\$100, ΔS=1, Δt=5/4800: \$2.8383

YORK UNIVERSITÉ UNIVERSITY

Example (Stability)

We compare implicit finite difference solution for a European put with the exact Black-Scholes formula, where T = 5/12 yr, S_0=\$50, K = \$50, σ=30%, r = 10%.

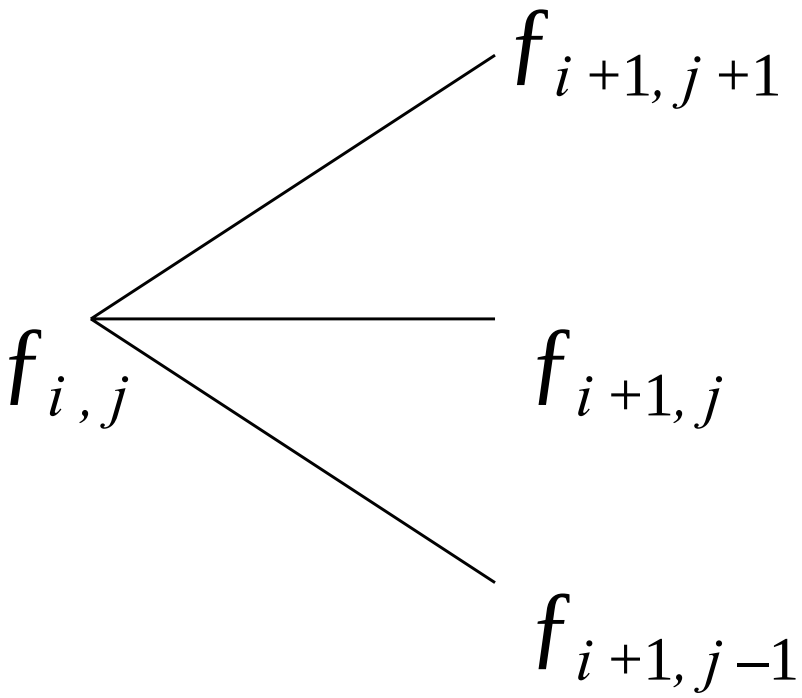
Black-Scholes Price: \$2.8846

IFD Method with S_{max}=\$100, ΔS=2, Δt=5/1200: \$2.8288

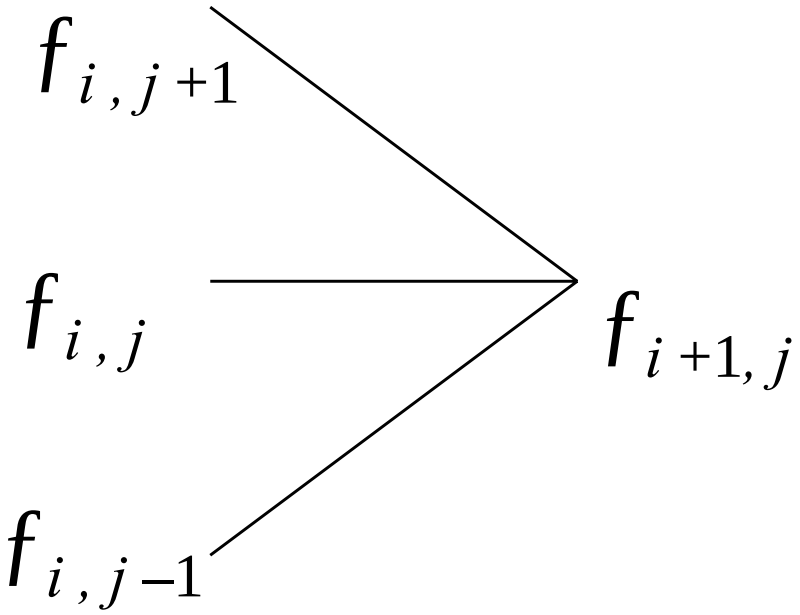
IFD Method with S_{max}=\$100, ΔS=1.5, Δt=5/1200: \$3.1325

IFD Method with S_{max}=\$100, ΔS=1, Δt=5/1200: \$2.8348

Implicit vs Explicit Finite Difference Methods



Explicit Method



Implicit Method

(always stable)

YORK UNIVERSITÉ UNIVERSITY

Implicit vs Explicit Finite Difference Method

- The explicit finite difference method is equivalent to the trinomial tree approach:
  - Truncation error: $O(\Delta t)$
  - Stability: not always
- The implicit finite difference method is equivalent to a multinomial tree approach:
  - Truncation error: $O(\Delta t)$
  - Stability: always

YORK UNIVERSITÉ UNIVERSITY

Other Points on Finite Difference Methods

- It is better to have ln S rather than S as the underlying variable in general
- Improvements over the basic implicit and explicit methods:
  - Crank-Nicolson method, average of explicit and implicit FD methods, trying to achieve
    - Truncation error: $O((\Delta t)^2)$
    - Stability: always

Chapter 5 Finite Difference Methods

5.3.2 Solving a linear system using direct methods

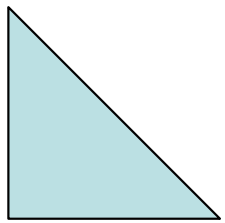
YORK UNIVERSITÉ UNIVERSITY

redefine THE POSSIBLE.

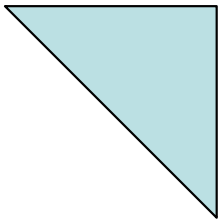
Solve Ax=b

$$A\,x=b$$

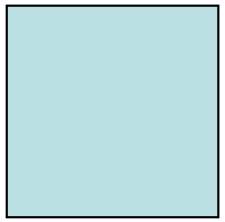
Various shapes of matrix A



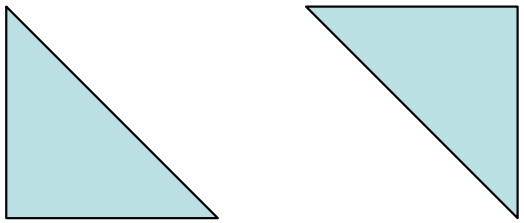
Lower triangular



Upper triangular



General



5.3.2.A Triangular Solvers

YORK UNIVERSITÉ UNIVERSITY

Example: 3 x 3 upper triangular system

$$\begin{bmatrix} 4 & 6 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 4 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 100 \\ 10 \\ 20 \end{bmatrix}$$

$$\Rightarrow x_3 = 20/4 = 5$$

$$\Rightarrow x_2 = 10 - x_3 = 5$$

$$\Rightarrow 4x_1 = 100 - x_3 - 6 * x_2 = 65$$

$$\therefore x_1 = 65/4$$

YORK UNIVERSITÉ UNIVERSITY

Solve an upper triangular system Ax=b

$$\begin{pmatrix} 0 & \cdots & 0 & a_{ii} & \cdots & a_{in} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} = a_{ii}x_i + \sum_{j>i} a_{ij}x_j = b_i, i = 1, \ldots, n$$

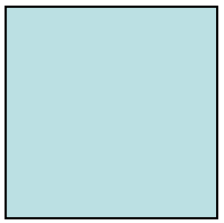
$$\Rightarrow x_n = b_n / a_{nn}$$

$$x_i = \left(b_i - \sum_{j>i} a_{ij}x_j \right) \Big/ a_{ii}, i = 1, \cdots n-1$$

YORK
UNIVERSITÉ
UNIVERSITY

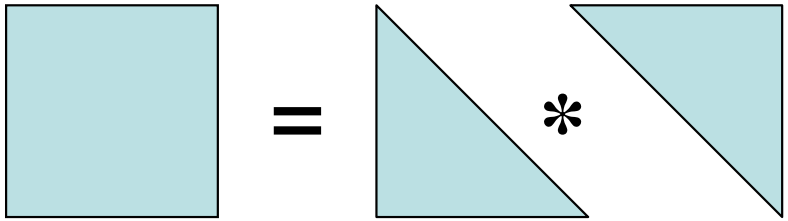
Implementation

```
Function x = UpperTriSolve(A, b)
        n = length(b);
        x(n) = b(n)/A(n,n);
        for i = n-1:-1:1
                sum = b(i);
                for j = i+1:n
                        sum = sum - A(i,j)*x(j);
                end
                x(i) = sum/A(i,i);
        end
```



5.3.2.B Gauss Elimination

To solve Ax=b for general form A



To solve Ax = b :

Suppose A = LU. Then

$$Ax = L\underline{Ux} = b$$

$$\Downarrow$$

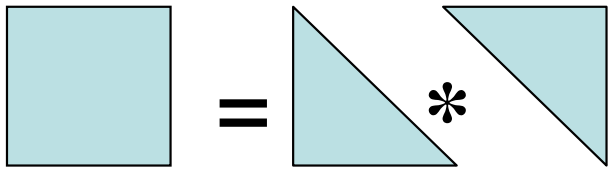
$$z$$

Solve two triangular systems :

1) solve z from Lz = b

2) solve x from Ux = z

Gauss Elimination



Goal: Make A an upper triangular matrix through using fundamental row operations

For example, to zero the elements in the lower triangular part of A

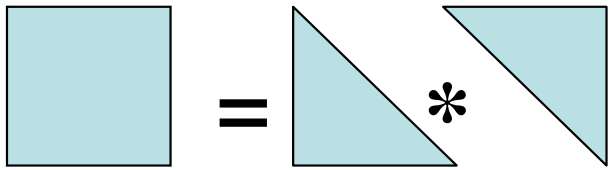
1) Zero a_{21}

$$\begin{bmatrix} 1 & 0 & 0 \\ -\frac{a_{21}}{a_{11}} & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ 0 & a_{22} - \frac{a_{21}a_{12}}{a_{11}} & a_{23} - \frac{a_{21}a_{13}}{a_{11}} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}$$

$$\Downarrow$$

$$\mathbf{E}_{21}\mathbf{A}$$

Gauss Elimination



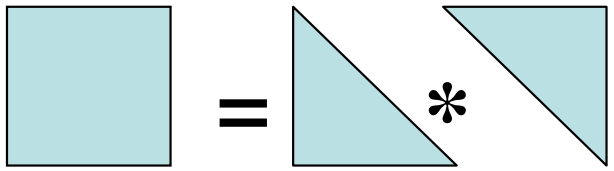
2) Zero a_{31}

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -\frac{a_{31}}{a_{11}} & 0 & 1 \end{bmatrix} \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ 0 & a_{22}-\frac{a_{21}a_{12}}{a_{11}} & a_{23}-\frac{a_{21}a_{13}}{a_{11}} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ 0 & a_{22}-\frac{a_{21}a_{12}}{a_{11}} & a_{23}-\frac{a_{21}a_{13}}{a_{11}} \\ 0 & a_{32}-\frac{a_{31}a_{12}}{a_{11}} & a_{33}-\frac{a_{31}a_{13}}{a_{11}} \end{bmatrix}$$

$$\Downarrow$$

$$\mathbf{E}_{31}\mathbf{E}_{21}\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ 0 & \tilde{a}_{22} & \tilde{a}_{23} \\ 0 & \tilde{a}_{32} & \tilde{a}_{33} \end{bmatrix}$$

Gauss Elimination



3) Zero $\tilde{a}_{32}$

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -\dfrac{\tilde{a}_{32}}{\tilde{a}_{22}} & 1 \end{bmatrix} \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ 0 & \tilde{a}_{22} & \tilde{a}_{23} \\ 0 & \tilde{a}_{32} & \tilde{a}_{33} \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ 0 & \tilde{a}_{22} & \tilde{a}_{23} \\ 0 & 0 & \tilde{a}_{33} - \dfrac{\tilde{a}_{32}\tilde{a}_{23}}{\tilde{a}_{22}} \end{bmatrix} \equiv \mathbf{U}$$

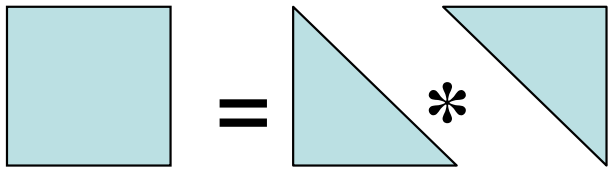
$$\Downarrow$$

$$\mathbf{E}_{32}\mathbf{E}_{31}\mathbf{E}_{21}\mathbf{A} = \mathbf{U}$$

$$\Downarrow$$

lower triangular

Gauss Elimination



Claim 1:

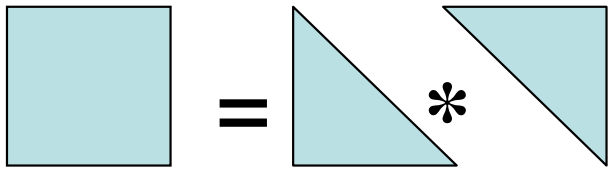
$$\mathbf{E}_{32}\mathbf{E}_{31}\mathbf{E}_{21} = \begin{bmatrix} 1 & 0 & 0 \\ -\frac{a_{21}}{a_{11}} & 1 & 0 \\ -\frac{a_{31}}{a_{11}} & -\frac{\tilde{a}_{32}}{\tilde{a}_{22}} & 1 \end{bmatrix}$$

Claim 2:

$$\left(\mathbf{E}_{32}\mathbf{E}_{31}\mathbf{E}_{21}\right)^{-1} = \begin{bmatrix} 1 & 0 & 0 \\ \frac{a_{21}}{a_{11}} & 1 & 0 \\ \frac{a_{31}}{a_{11}} & \frac{\tilde{a}_{32}}{\tilde{a}_{22}} & 1 \end{bmatrix}$$

YORK UNIVERSITÉ UNIVERSITY

LU Factorization



Therefore, through Gauss Elimination, we have

$$\mathbf{E}_{32}\mathbf{E}_{31}\mathbf{E}_{21}\mathbf{A} = \mathbf{U}$$

$$\mathbf{A} = \left(\mathbf{E}_{32}\mathbf{E}_{31}\mathbf{E}_{21}\right)^{-1}\mathbf{U}$$

$$\mathbf{A} = \mathbf{L}\mathbf{U}$$

It is called LU factorization. When A is symmetric,

$\mathbf{A} = \mathbf{L}\mathbf{L}^T$ which is called *Cholesky Decomposition*

YORK
UNIVERSITÉ
UNIVERSITY

To solve Ax=b for general form A

To solve $Ax = b$ becomes

1) Use Gauss elimination to make A upper triangular, i.e.

$$L^{-1}Ax = L^{-1}b \Rightarrow Ux = z$$

2) Solve x from $Ux = z$

This suggests that when doing Gauss elimination, we can do it to the augmented matrix $[A\ b]$ associated with the linear system.

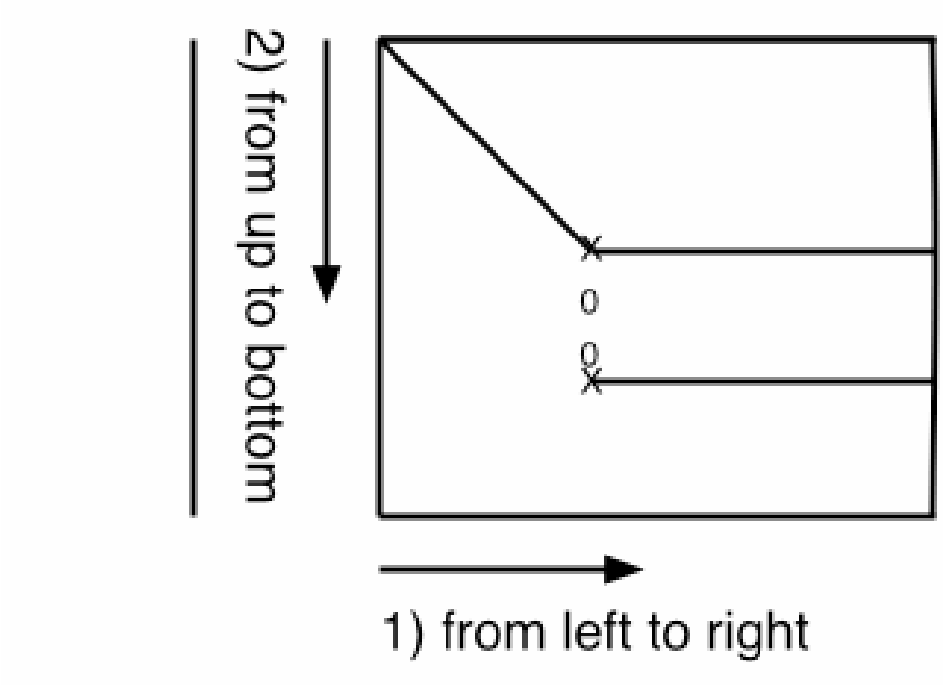
YORK UNIVERSITÉ UNIVERSITY

An exercise

$$\begin{bmatrix} 2 & -1 & \\ -1 & 2 & -1 \\ & -1 & 2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 4 \end{bmatrix}$$

```
% build the matrix A
A = [2, -1, 0; -1, 2, -1; 0, -1, 2]

% build the vector b
x_true = [1:3]';
b = A * x_true;

% lu decomposition of A
[l, u] = lu(A)

% solve z from lz = b where z = ux
z = l\b;

% solve x from ux = z
x = u\z
```

General Gauss Elimination



row i

row j - $\frac{a_{ji}}{a_{ii}}$ * **row i**

What if we have zero or very small diagonal elements?

Somtimes, we need to permute rows of A so that Gauss eliminatio n can be computed or computed stably, i.e., PA = LU. This is called partial pivoting.

For example

$$\mathbf{A} = \begin{bmatrix} 0 & 1 \\ 2 & 3 \end{bmatrix}$$

$$\mathbf{PA} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}\begin{bmatrix} 0 & 1 \\ 2 & 3 \end{bmatrix} = \begin{bmatrix} 2 & 3 \\ 0 & 1 \end{bmatrix}$$

$$\mathbf{A} = \begin{bmatrix} 0.0001 & 1 \\ 1 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 10{,}000 & 1 \end{bmatrix}\begin{bmatrix} 0.0001 & 1 \\ 0 & -9999 \end{bmatrix} = \mathbf{LU}$$

$$\mathbf{PA} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}\begin{bmatrix} 0.0001 & 1 \\ 1 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0.0001 & 1 \end{bmatrix}\begin{bmatrix} 1 & 1 \\ 0 & .9999 \end{bmatrix} = \mathbf{LU}$$

YORK UNIVERSITÉ UNIVERSITY

Can we always have A = LU?

No!

If $det\left(\mathbf{A}\left(1{:}k,1{:}k\right)\right) \neq 0$ for k=1,…,n-1,

then A $\in \mathbb{R}^{nxn}$ has an LU factorization.

Proof:

See Golub and Loan, Matrix Computation, 3rd edition

Chapter 5 Finite Difference Methods

5.4.1 Crank-Nicolson Method

YORK UNIVERSITÉ UNIVERSITY

redefine THE POSSIBLE.

YORK UNIVERSITÉ UNIVERSITY

Explicit Finite Difference Methods

With $\frac{\partial f}{\partial t} + rS\frac{\partial f}{\partial S} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 f}{\partial S^2} = rf,$

we can obtain an explicite form:

$$\frac{f_{i+1,j} - f_{i,j}}{\Delta t} + O(\Delta t) =$$

$$-rj\Delta S \frac{f_{i+1,j+1} - f_{i+1,j-1}}{2\Delta S} - \frac{1}{2}\sigma^2 j^2 (\Delta S)^2 \frac{f_{i+1,j+1} + f_{i+1,j-1} - 2f_{i+1,j}}{\Delta S^2}$$

$$+rf_{i+1,j} \quad + O\left((\Delta S)^2\right)$$

YORK UNIVERSITÉ UNIVERSITY

Implicit Finite Difference Methods

With $$\frac{\partial f}{\partial t} + rS\frac{\partial f}{\partial S} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 f}{\partial S^2} = r f,$$

we can obtain an implicit form :

$$\frac{f_{i+1,j} - f_{i,j}}{\Delta t} + O(\Delta t) =$$

$$-rj\Delta S \frac{f_{i,j+1} - f_{i,j-1}}{2\Delta S} - \frac{1}{2}\sigma^2 j^2 (\Delta S)^2 \frac{f_{i,j+1} + f_{i,j-1} - 2f_{i,j}}{\Delta S^2}$$

$$+rf_{i,j} + O\left((\Delta S)^2\right)$$

YORK UNIVERSITÉ UNIVERSITY

Crank-Nicolson Methods: average of explicit and implicit methods (Brandmarte, p485)

$$\frac{f_{i+1,j}-f_{i,j}}{\Delta t}+O\left((\Delta t)^2\right)=$$

$$-\frac{rj\Delta S}{2}\left(\frac{f_{i+1,j+1}-f_{i+1,j-1}}{2\Delta S}+\frac{f_{i,j+1}-f_{i,j-1}}{2\Delta S}\right)$$

$$-\frac{1}{4}\sigma^2 j^2(\Delta S)^2\left(\frac{f_{i+1,j+1}+f_{i+1,j-1}-2f_{i+1,j}}{\Delta S^2}+\frac{f_{i,j+1}+f_{i,j-1}-2f_{i,j}}{\Delta S^2}\right)$$

$$+\frac{r}{2}\left(f_{i+1,j}+f_{i,j}\right)+O\left((\Delta S)^2\right)$$

YORK UNIVERSITÉ UNIVERSITY

Crank-Nicolson Methods

Rewriting the equation, we get an Crank-Nicolson scheme:

$$-\alpha_j f_{i,j-1} + (1-\beta_j) f_{i,j} - \gamma_j f_{i,j+1} =$$

$$\alpha_j f_{i+1,j-1} + (1+\beta_j) f_{i+1,j} + \gamma_j f_{i+1,j+1} \qquad (5.5)$$

where

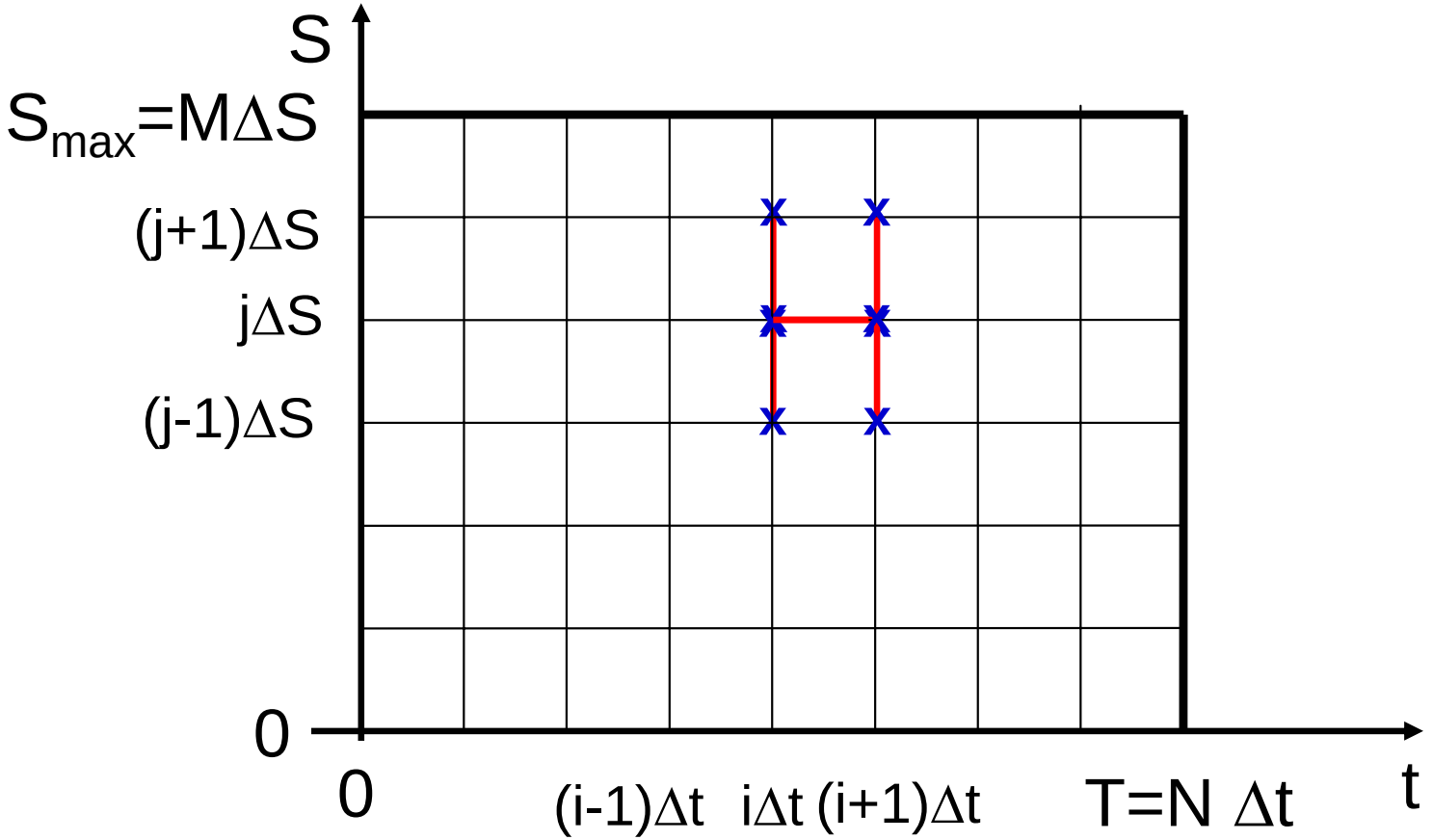
$$\alpha_j = \frac{\Delta t}{4}\left(\sigma^2 j^2 - rj\right)$$

$$\beta_j = -\frac{\Delta t}{2}\left(\sigma^2 j^2 + r\right)$$

$$\gamma_j = \frac{\Delta t}{4}\left(\sigma^2 j^2 + rj\right)$$

for $i = N-1, N-2, \ldots, 1, 0$ and $j = 1, 2, \ldots, M-1$.

Numerical Computation Dependency



Implementation

Equation (5.5) can be rewritten in matrix form:

$$\mathbf{M}_1\mathbf{f}_i = \mathbf{M}_2\mathbf{f}_{i+1} + \mathbf{b}$$

where $\mathbf{f}_i$ and $\mathbf{b}_i$ are $(M-1)$ dimensional vectors

$$\mathbf{f}_i = \left[f_{i,1}, f_{i,2}, f_{i,3} \cdots, f_{i,M-1} \right]^T,$$

$$\mathbf{b} = \left[\alpha_1 \left(f_{i,0} + f_{i+1,0} \right), 0 \cdots, \gamma_{M-1} \left(f_{i,M} + f_{i+1,M} \right) \right]^T$$

and $\mathbf{M}_1$ and $\mathbf{M}_2$ are $(M-1)\times(M-1)$ symmetric matrices

$$\mathbf{M}_1 = \begin{bmatrix} 1-\beta_1 & -\gamma_1 & 0 & \cdots & 0 \\ -\alpha_2 & 1-\beta_2 & \gamma_2 & \ddots & 0 \\ 0 & \alpha_3 & & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & -\gamma_{M-2} \\ 0 & \cdots & 0 & -\alpha_{M-1} & 1-\beta_{M-1} \end{bmatrix}$$

YORK UNIVERSITÉ UNIVERSITY

Implementation

and

$$\mathbf{M}_2 = \begin{bmatrix} 1+\beta_1 & \gamma_1 & 0 & \cdots & 0 \\ \alpha_2 & 1+\beta_2 & \gamma_2 & \ddots & 0 \\ 0 & \alpha_3 & & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \gamma_{M-2} \\ 0 & \cdots & 0 & \alpha_{M-1} & 1+\beta_{M-1} \end{bmatrix}$$

YORK UNIVERSITÉ UNIVERSITY

Example

We compare Crank-Nicolson Methods for a European put with the exact Black-Scholes formula, where T = 5/12 yr, S_0=\$50, K = \$50, σ=30%, r = 10%.

Black-Scholes Price: \$2.8446

CN Method with S_{max}=\$100, ΔS=2, Δt=5/1200: \$2.8241

CN Method with S_{max}=\$100, ΔS=1, Δt=5/4800: \$2.8395

YORK UNIVERSITÉ UNIVERSITY

Example (Stability)

We compare Crank-Nicolson Method for a European put with the exact Black-Scholes formula, where T = 5/12 yr, S_0=\$50, K = \$50, σ=30%, r = 10%.

Black-Scholes Price: \$2.8446

CN Method with S_{max}=\$100, ΔS=1.5, Δt=5/1200: \$3.1370

CN Method with S_{max}=\$100, ΔS=1, Δt=5/1200: \$2.8395

YORK
UNIVERSITÉ
UNIVERSITY

Example: Barrier Options

Barrier options are options where the payoff depends on whether the underlying asset's price reaches a certain level during a certain period of time.

Types:

knock-out: option ceases to exist when the asset price reaches a barrier

Knock-in: option comes into existence when the asset price reaches a barrier

YORK UNIVERSITÉ UNIVERSITY

Example: Barrier Option

We compare Crank-Nicolson Method for a European down-and-out put, where T = 5/12 yr, S_0=\$50, K = \$50, S_b=\$50 σ=40%, r = 10%.

What are the boundary conditions for S?

YORK UNIVERSITÉ UNIVERSITY

Example: Barrier Option

We compare Crank-Nicolson Method for a European down-and-out put, where T = 5/12 yr, S_0=\$50, K = \$50, S_b=\$50 σ=40%, r = 10%.

Boundary conditions are $f\left(t, S_{max}\right) = 0$ and $f\left(t, S_b\right) = 0$

Exact Price (Hall, p533-535): \$0.5424

CN Method with S_{max}=\$100, ΔS=0.5, Δt=1/1200: \$0.5414

Appendix A.

Matrix Norms

YORK UNIVERSITÉ UNIVERSITY

redefine THE POSSIBLE.

Vector Norms

- Norms serve as a way to measure the length of a vector or a matrix
- A vector norm is a function mapping $\mathbf{x} \in \Re^n$ to a real number $\|\mathbf{x}\|$ s.t.
  - $\|\mathbf{x}\| > 0$ for any $\mathbf{x} \neq \mathbf{0}$; $\|\mathbf{x}\| = 0$ iff $\mathbf{x} = 0$
  - $\|c\,\mathbf{x}\| = |c|\|\mathbf{x}\|$ for any $c \in \Re$
  - $\|\mathbf{x} + \mathbf{y}\| \leq \|\mathbf{x}\| + \|\mathbf{y}\|$ for any $\mathbf{x}, \mathbf{y} \in \Re^n$
- There are various ways to define a norm

$$\|\mathbf{x}\|_p \equiv \left(\sum_{i=1}^{n} |x_i|^p \right)^{1/p} \quad (\text{p} = 2 \text{ is the Euclidean norm})$$

$$\|\mathbf{x}\|_\infty \equiv \max_{1 \leq i \leq n} |x_i|$$

– For example, $\mathbf{v} = [2 \;\; 4 \;\; -1 \;\; 3]$. $\|\mathbf{v}\|_1 = ?$, $\|\mathbf{v}\|_\infty = ?$, $\|\mathbf{v}\|_2 = ?$

Matrix Norms

- Similarly, a matrix norm is a function mapping $\mathbf{A} \in \Re^{m \times n}$ to a real number $\|\mathbf{A}\|$ s.t.
  - $\|\mathbf{A}\| > 0$ for any $\mathbf{A} \neq \mathbf{0}$; $\|\mathbf{A}\| = \mathbf{0}$ iff $\mathbf{A} = \mathbf{0}$
  - $\|c\ \mathbf{A}\| = |c| \|\mathbf{A}\|$ for any $c \in \Re$
  - $\|\mathbf{A} + \mathbf{B}\| \leq \|\mathbf{A}\| + \|\mathbf{B}\|$ for any $\mathbf{A}$, $\mathbf{B}\ \in \Re^{m \times n}$
- Various commonly used matrix norms

$$\|\mathbf{A}\|_p \equiv \sup_{\mathbf{x} \neq 0} \frac{\|\mathbf{A}\mathbf{x}\|_p}{\|\mathbf{x}\|_p} \qquad \|\mathbf{A}\|_F \equiv \left(\sum_{i=1}^{m} \sum_{j=1}^{n} |a_{ij}|^2 \right)^{1/2}$$

$$\|\mathbf{A}\|_1 \equiv \max_{1 \leq j \leq n} \sum_{i=1}^{m} |a_{ij}| \qquad \|\mathbf{A}\|_\infty \equiv \max_{1 \leq i \leq m} \sum_{j=1}^{n} |a_{ij}|$$

$\|\mathbf{A}\|_2 \equiv \sqrt{\rho(\mathbf{A}^T \mathbf{A})}$, the spectral norm, where

$\rho(\mathbf{B}) \equiv \max\{|\lambda_k| : \lambda_k$ is an eigenvalue of $\mathbf{B}\}$

YORK UNIVERSITÉ UNIVERSITY

An Example

$$\mathbf{A} = \begin{bmatrix} 2 & 4 & -1 \\ 3 & 1 & 5 \\ -2 & 3 & -1 \end{bmatrix}$$

$$\|\mathbf{A}\|_\infty = ? \qquad \|\mathbf{A}\|_2 = ?$$

$$\|\mathbf{A}\|_1 = ? \qquad \|\mathbf{A}\|_F = ?$$

YORK
UNIVERSITÉ
UNIVERSITY

Basic Properties of Norms

Let $\mathbf{A}, \mathbf{B} \in \Re^{n\times n}$ and $\mathbf{x}, \mathbf{y} \in \Re^{n}$. Then

1. $\|\mathbf{x}\| \geq 0$; and $\|\mathbf{x}\| = 0 \Leftrightarrow \mathbf{x} = \mathbf{0}$
2. $\|\mathbf{x} + \mathbf{y}\| \leq \|\mathbf{x}\| + \|\mathbf{y}\|$
3. $\|\alpha\, \mathbf{x}\| = |\alpha| \|\mathbf{x}\|$ where α is a real number
4. $\|\mathbf{A}\mathbf{x}\| \leq \|\mathbf{A}\| \|\mathbf{x}\|$
5. $\|\mathbf{A}\mathbf{B}\| \leq \|\mathbf{A}\| \|\mathbf{B}\|$

YORK UNIVERSITÉ UNIVERSITY

Condition number of a square matrix

All norms in $\Re^n$ $\left(\Re^{m\times n}\right)$ are equivalent. That is, if $\|\bullet\|_\alpha$ and $\|\bullet\|_\beta$ are norms on $\Re^n$, then $\exists\, c_1, c_2 > 0$ such that for all $\mathbf{x} \in \Re^n$, we have

$$c_1 \|\mathbf{x}\|_\alpha \le \|\mathbf{x}\|_\beta \le c_2 \|\mathbf{x}\|_\alpha$$

Condition Number of A Matrix: $C \equiv \|\mathbf{A}\|\|\mathbf{A}^{-1}\|$, where $A \in \Re^{n\times n}$.

The condition number gives a measure of how close a matrix is close to singular. The bigger the C, the harder it is to solve **Ax = b**.

YORK
UNIVERSITÉ
UNIVERSITY

Convergence

- **vectors $\mathbf{x}^k$ converges to $\mathbf{x}$** $\Leftrightarrow \|\mathbf{x}^k - \mathbf{x}\|$ **converges to 0**

- **matrix $\mathbf{A}^k \to \mathbf{0}$** $\Leftrightarrow \|\mathbf{A}^k - \mathbf{0}\| \to 0$

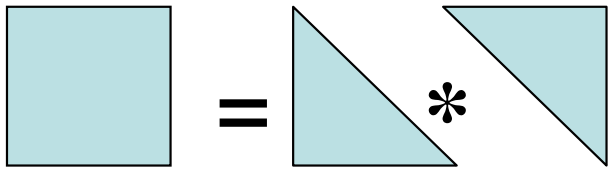
Appendix B.

Basic Row Operations

YORK UNIVERSITÉ UNIVERSITY

redefine THE POSSIBLE.

Basic row operations



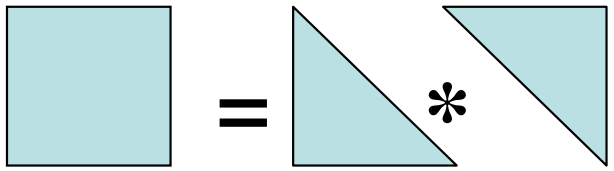
Three kinds of basic row operations:

1) Interchange the order of two rows or (equations)

$$\begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} = \begin{bmatrix} a_{21} & a_{22} & a_{23} \\ a_{11} & a_{12} & a_{13} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}$$

YORK UNIVERSITÉ UNIVERSITY

Basic row operations



2) Multiply a row by a nonzero constant

$$\begin{bmatrix} c & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} = \begin{bmatrix} c\,a_{11} & c\,a_{12} & c\,a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}$$

3) Add or subtract rows

$$\begin{bmatrix} 1 & 0 & 0 \\ -1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21}-a_{11} & a_{22}-a_{12} & a_{23}-a_{13} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}$$