Computational Finance

Part II: Derivatives Pricing

Mahesan Niranjan

School of Electronics and Computer Science
University of Southampton

Slides are prompts (for me). You are expected to make your own notes when material is explained using the whiteboard, and by referring to textbooks and papers.

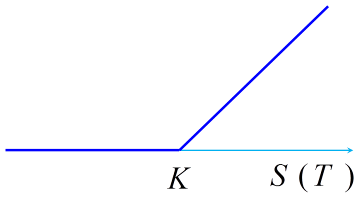
Spring Semester 2016/17

Derivatives Pricing

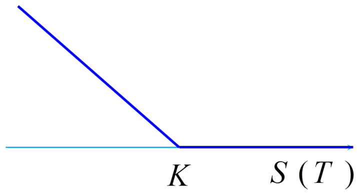
- Efficiency, no-arbitrage and fair price
- Example:
  - Price today $S(0)$
  - A and B enter into a *future* contract to sell/buy at price F at time T
  - A borrows $S(0)$ from the bank, buys the asset and waits till T
  - At time T, A owes the bank $S(0)\exp(rT)$ and has the asset to sell to B.
  - $F = S(0)\exp(rT)$, else arbitrage opportunity

Options

- Call: right to buy at price K at time T



- Put: right to sell at price K at time T



- Exercise of contract
  - European style: only at time T
  - American style: any time in $0 \rightarrow T$

- Example: Put-Call Parity

K: strike price

  - Portfolio P_1: European Call + cash $K \exp(-rT)$
  - Portfolio P_2: European Put + one share of underlying stock
  - Values at time $t = 0$

P_1	$C + K \exp -rT$
P_2	$P + S(0)$

  - Value of portfolios at time $t = T$

$S(T) > K$	P_1	$[S(T) - K] + K = S(T)$
	P_2	$0 + S(T) = S(T)$
$S(T) < K$	P_1	$0 + K = K$
	P_2	$[K - S(T)] + S(T) = K$

  - Both portfolios having the same value at time $t = T$ should also have the same value at $t = 0$.

$$C + K \exp(-rT) = P + S(0)$$

- Geometric Brownian motion for stock price

$$\begin{aligned} dS(t) &= \mu S(t)dt + \sigma S(t)dW(t) \\ \frac{dS(t)}{S(t)} &= \mu dt + \sigma dW(t) \end{aligned}$$

- Stochastic differential equation for the log of the process

$$F(S,t) = \log S(t)$$

- Ito's lemma tells us about increments dF
- Terms needed to apply Ito's lemma

$$\boxed{\begin{aligned} \frac{\partial F}{\partial t} &= 0 \\ \frac{\partial F}{\partial S} &= \frac{1}{S} \\ \frac{\partial^2 F}{\partial S^2} &= -\frac{1}{S^2} \end{aligned}}$$

$$\begin{aligned} dF &= \left(\frac{\partial F}{\partial t} + \mu S\frac{\partial F}{\partial S} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 F}{\partial S^2}\right) dt + \sigma S \frac{\partial F}{\partial S} dW \\ &= \left(\mu - \frac{\sigma^2}{2}\right) dt + \sigma dW \end{aligned}$$

$$\log S(t) = \log S(0) + \left(\mu - \frac{\sigma^2}{2}\right) t + \sigma dW(t)$$

- $dW(t) = \epsilon\sqrt{t}$ where $\epsilon \sim \mathcal{N}(0,1)$

$$\log S(t) \sim \mathcal{N}\left[\log S(0) + \left(\mu - \frac{\sigma^2}{2}\right) t, \;\; \sigma^2 t\right]$$

- Log of asset price has a normal distribution
- Also

$$S(t) = S(0) \exp\left((\mu - \sigma^2/2)t + \sigma\sqrt{t}\epsilon\right)$$

Black-Scholes Model

- Model

$$dS(t) = \mu S(t)dt + \sigma S(t)dW(t)$$

- Change in option price

$$df = \frac{\partial f}{\partial t}dt + \frac{\partial f}{\partial S}dS + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 f}{\partial S^2}dt$$

- At maturity

$$F(S(T), T) = \max\{S(T) - K, 0\}$$

- Consider a portfolio
  - Own Δ stocks (long)
  - One call option sold

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

$$\begin{aligned} d\Pi &= \Delta dS - df \\ &= \left(\Delta - \frac{\partial f}{\partial S}\right) dS - \left(\frac{\partial f}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 f}{\partial S^2}\right) \end{aligned}$$

- Term in dS (stochastic) can be eliminated by choosing Δ

$$\Delta = \frac{\partial f}{\partial S}$$

- With this choice of Δ (balance between short and long), the portfolio is riskless.
- $d\Pi = r\Pi dt$
- Eliminating $d\Pi$

$$\left(\frac{\partial f}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 f}{\partial S^2}\right) dt = r\left(f - S\frac{\partial f}{\partial S}\right) dt$$

$$\frac{\partial f}{\partial t} + rS\frac{\partial f}{\partial S} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 f}{\partial S^2} - rf = 0$$

- Partial differential equation

$$\frac{\partial f}{\partial t} + rS\frac{\partial f}{\partial S} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 f}{\partial S^2} - rf = 0$$

- Boundary condition
  - European Call: $f(S,T) = \max\{S-K, 0\}$
  - European Put: $f(S,T) = \max\{S-K, 0\}$
- Black-Scholes

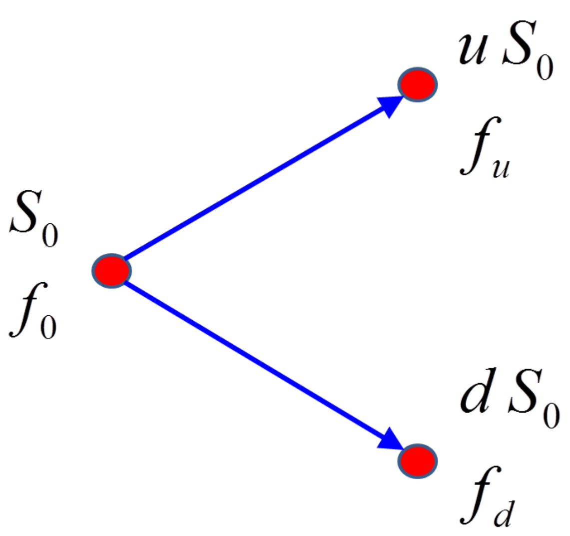
$$C = S_0\,\mathcal{N}(d_1) = K\,\exp(-rT)\mathcal{N}(d_2)$$

$$\begin{aligned} d_1 &= \frac{\log(S_0/K) + (r+\sigma^2/2)T}{\sigma\sqrt{T}} \\ d_2 &= \frac{\log(S_0/K) + (r-\sigma^2/2)T}{\sigma\sqrt{T}} \\ &= d_1 - \sigma\sqrt{T} \\ \mathcal{N}(x) &= \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{x} \exp(-y^2/2)dy \end{aligned}$$

- Put-Call parity

$$P = K\exp(-rT)\mathcal{N}(-d_2) - S_0\mathcal{N}(-d_1)$$

Binomial Lattice



Options Pricing on a Binomial Model

- Construct a portfolio:
  - A riskless bond, initial price $B_0 = 1$ and future value $B_1 = \exp(r\delta t)$
  - Underlying asset, initial value S_0
  - Number of stocks Δ, number of bonds Ψ
- Initial value of this portfolio

$$\Pi_0 = \Delta S_0 + \Psi$$

- Future value depends on price movement up or down

$$\begin{cases} \Pi_u = \Delta S_0 u + \Psi \exp(r\delta t) \\ \Pi_d = \Delta S_0 d + \Psi \exp(r\delta t) \end{cases}$$

- We can solve for a portfolio that will replicate option payoff

$$\begin{aligned} \Delta S_0 u + \Psi \exp(r\delta t) &= f_u \\ \Delta S_0 d + \Psi \exp(r\delta t) &= f_d \end{aligned}$$

... and solve for Δ and Ψ

- ... solving

$$\begin{aligned} \Delta &= \frac{f_u - f_d}{S_0(u-d)} \\ \Psi &= \exp(-r\delta t)\frac{uf_d - df_u}{u-d} \end{aligned}$$

- No arbitrage $\implies$ initial value of this portfolio should be f_0

$$\begin{aligned} f_0 &= \Delta S_0 + \Psi \\ &= \frac{f_u - f_d}{(u-d)} + \exp(-r\delta t)\frac{uf_d - df_u}{u-d} \\ &= \exp(-r\delta t)\left\{\frac{--}{u-d} f_u + \frac{--}{u-d} f_d\right\} \end{aligned}$$

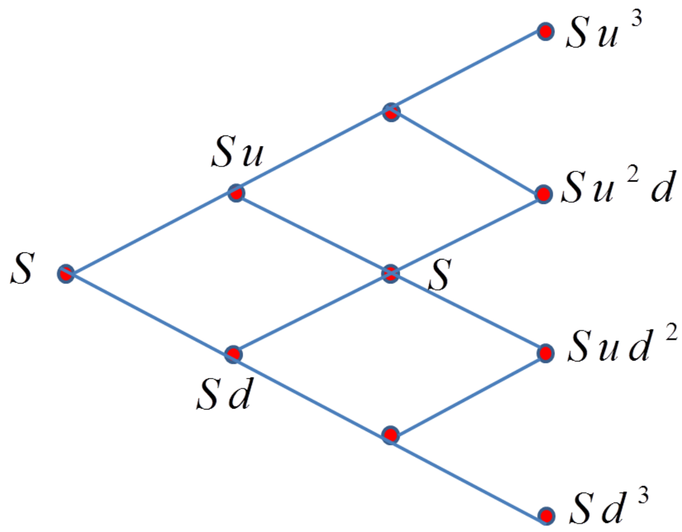
- Defining probabilities

$$\pi_u = \frac{\exp(r\delta t) - d}{u-d} \text{ and } \pi_d = \frac{u - \exp(r\delta t)}{u-d}$$

option price interpreted as discounted expected value

$$f_0 = \exp(-r\delta t)\left(\pi_u f_u + \pi_d f_d\right)$$

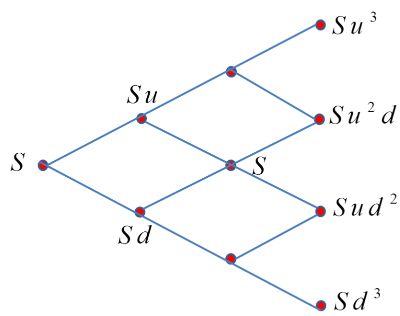
Binomial Lattice



Calibrating a Binomial Lattice

- When are these equivalent?

$$dS = r\,S\,dt + \sigma\,S\,dW$$



- Log normal distribution

$$\log(S_{t+\delta t}) \sim \mathcal{N}\left(\left(r - \sigma^2/2\right),\ \sigma^2\,\delta t\right)$$

- Mean and variance of log normal distribution
  (log of the variable is normal, what is mean and variance of the variable?)

$$\begin{aligned} E\,[\,S_{t+\delta t}\,] &= \exp(r\,\delta t) \\ \mathrm{Var}\,[\,S_{t+\delta t}\,] &= \exp(2r\delta t)\left(\exp(\sigma^2\delta t) - 1\right) \end{aligned}$$

Calibrating binomial lattice (cont'd)

- Mean for the lattice

$$E\left[S_{t+\delta t}\right] = p\,u\,S_t + (1-p)\,d\,S_t$$

- Equating the means...

$$puS_t + (1-p)dS_t = \exp(r\,\delta t)\,S_t$$

$$p = \frac{\exp(r\,\delta t) - d}{u - d}$$

- Variance on the lattice

$$\begin{aligned}\mathrm{Var}\left[S_{t+\delta t}\right] &= E\left[S_{t+\delta t}^2\right] - E^2\left[S_{t+\delta t}\right] \\ &= S_t^2\left(pu^2 + (1-p)d^2\right) - S_t^2\exp(2r\delta t)\end{aligned}$$

... which from the dynamical model is...

$$\mathrm{Var}\left[S_{t+\delta t}\right] = S_t^2\exp(2r\delta t)\left(\exp(\sigma^2\delta t) - 1\right)$$

(cont'd)

- Equating the two variances

$$S_t^2\exp(2r\delta t)\left(\exp(\sigma^2\delta t) - 1\right) = S_t^2\left(pu^2 + (1-p)d^2\right) - S_t^2\exp(2r\delta t)$$

Which reduces to

$$\exp(2r\delta t + \sigma^2\delta t) = pu^2 + (1-p)d^2$$

Substitute for p and simplify

$$\exp(2r\delta t + \sigma^2\delta t) = (u+d)\exp(r\delta t) - 1$$

...and because $u = 1/d$,

$$u^2\exp(r\delta t) - u\left(1 + \exp(2r\delta t + \sigma^2\delta t)\right) + \exp(r\delta t) = 0$$

... a quadratic equation in u.

$$u = \frac{(1 + \exp(2r\delta t + \sigma^2 \delta t)) + \sqrt{(1 + \exp(2r\delta t + \sigma^2\delta t)^2 - 4\exp(2r\delta t)}}{2\exp(r\delta t)}$$

Taylor series expansion of $\exp(x)$

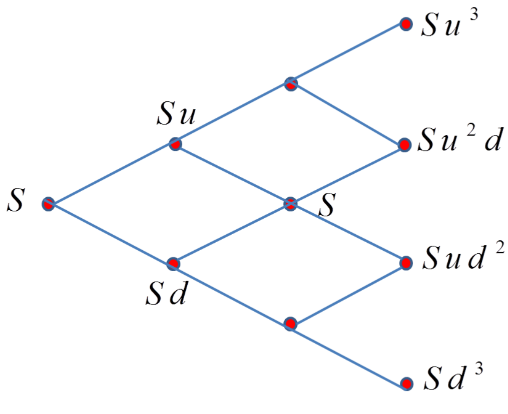
$$\left(1 + \exp(2r\delta t + \sigma^2\delta t)\right)^2 - 4\exp(2r\delta t) \approx (2 + (2r + \sigma^2)\delta t)^2 - 4(1 + 2r\delta t) \approx 4\sigma^2\delta t$$

$$\begin{aligned}
u &\approx \frac{2 + (2r + \sigma^2)\delta t + 2\sigma\sqrt{\delta t}}{2\exp(r\delta t)} \\
&\approx \left(1 + r\delta t + \frac{\sigma^2}{2}\delta t + \sigma\sqrt{\delta t}\right)(1 - r\delta t) \\
&\approx 1 + r\delta t + \frac{\sigma^2}{2}\delta t + \sigma\sqrt{\delta t} - r\delta t \\
&= 1 + \sigma\sqrt{\delta t} + \frac{\sigma^2}{2}\delta t
\end{aligned}$$

Calibrating the Binomial Lattice (cont'd)

$$\begin{aligned}
u &= \exp(\sigma\sqrt{\delta t}) \\
d &= \exp(-\sigma\sqrt{\delta t}) \\
p &= \frac{\exp(r\delta t) - d}{u - d}
\end{aligned}$$

$$dS = r\,S\,dt + \sigma\,S\,dW$$



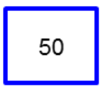
Example

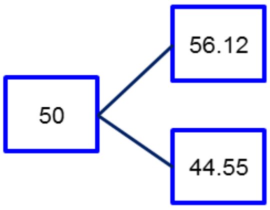
- European call option; $S_0 = K = 50$; $r = 0.1$; $\sigma = 0.4$; maturity in five months.

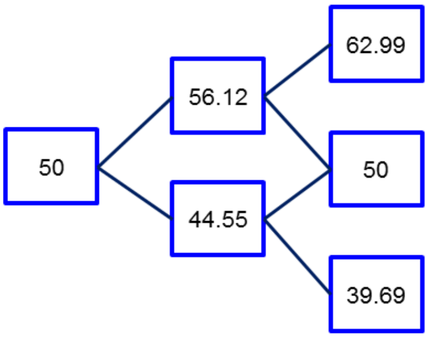
```
>> call = blsprice(50, 50, 0.1, 5/12, 0.4)
call =
       6.1165
```

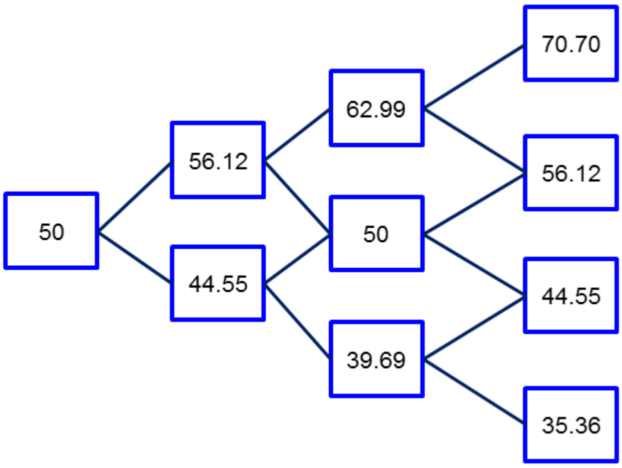
We can now build the lattice

δt	$1/12$	0.0833
u	$\exp(\sigma\sqrt{t})$	1.1224
d	$1\,/\,u$	0.8909
p	$(exp(r\delta t) - d)\,/\,(u - d)$	0.5073

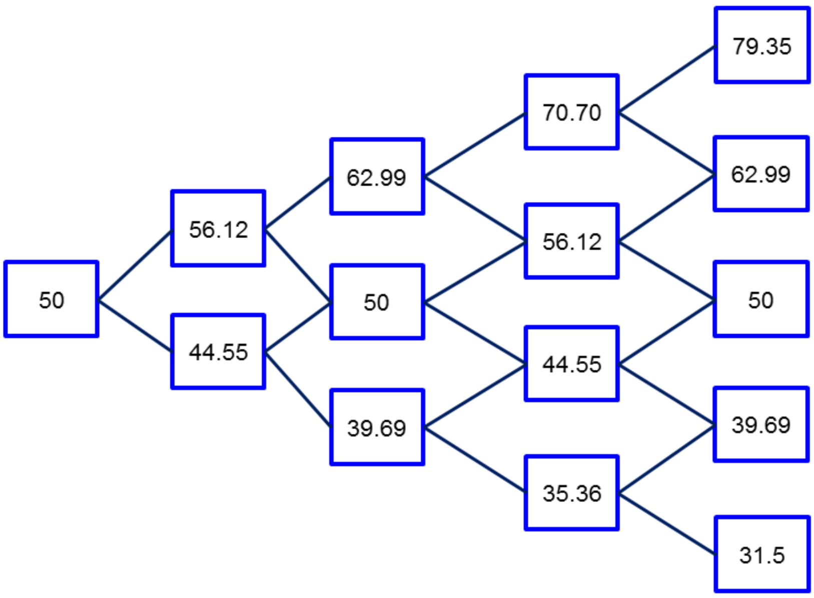



50
56.12
44.55

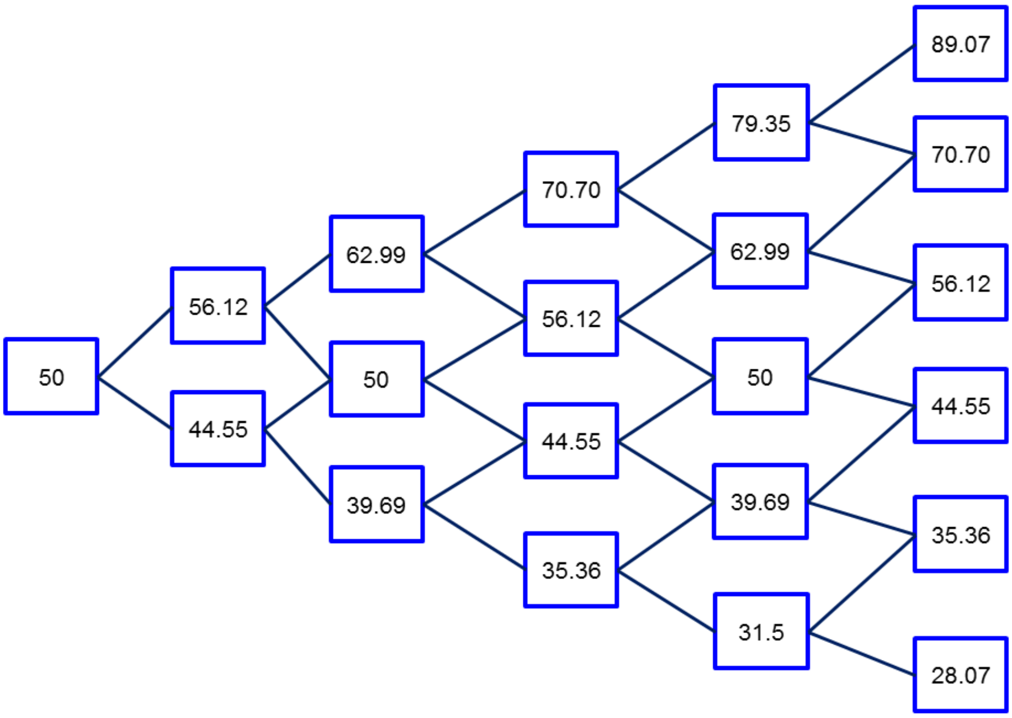

50
56.12
44.55
62.99
50
39.69

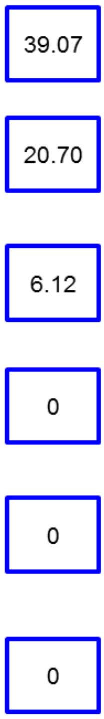


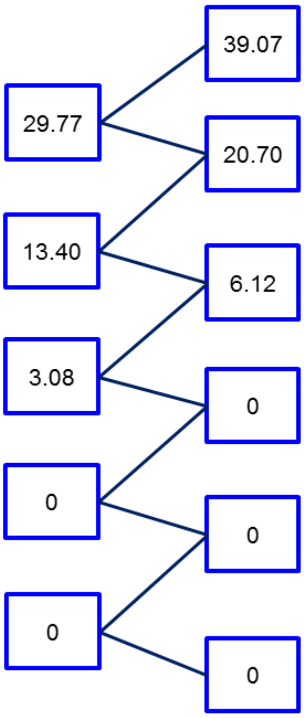
50
56.12
44.55
62.99
50
39.69
70.70
56.12
44.55
35.36

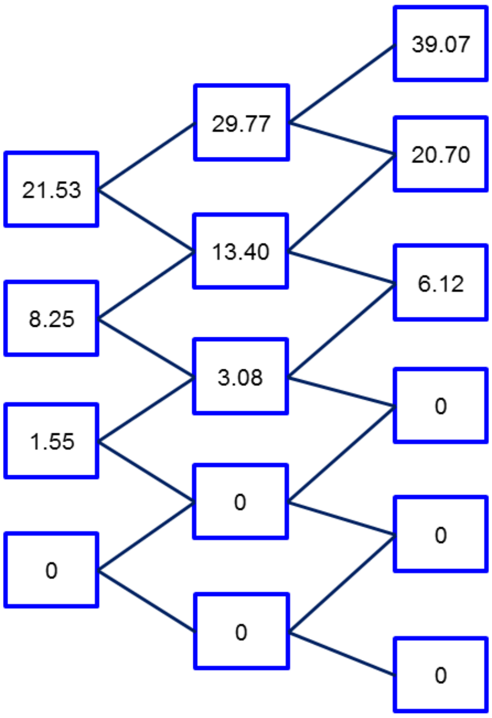


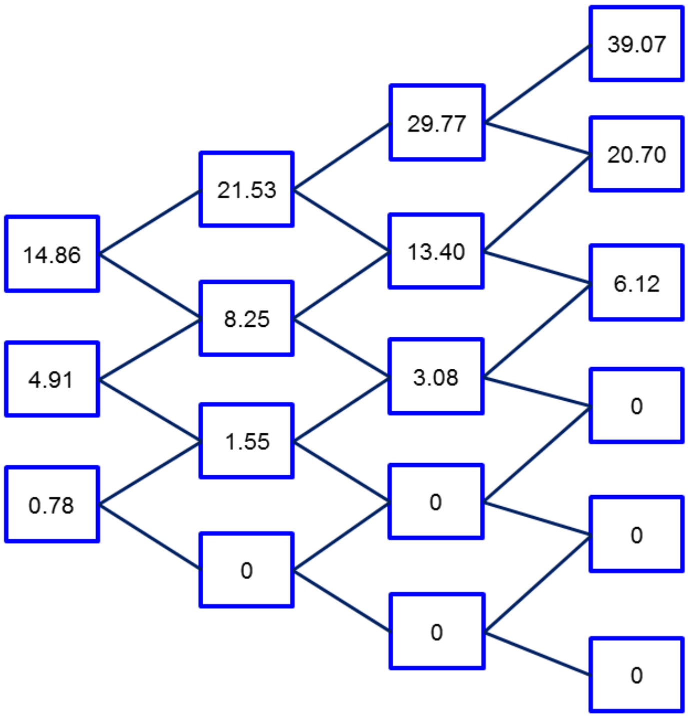
50
56.12
44.55
62.99
50
39.69
70.70
56.12
44.55
35.36
79.35
62.99
50
39.69
31.5

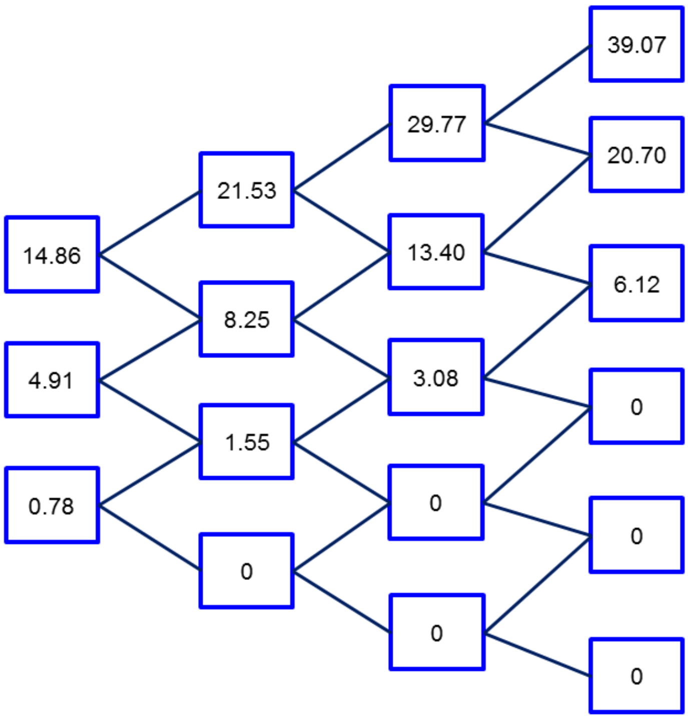

50
56.12
44.55
62.99
50
39.69
70.70
56.12
44.55
35.36
79.35
62.99
50
39.69
31.5
89.07
70.70
56.12
44.55
35.36
28.07

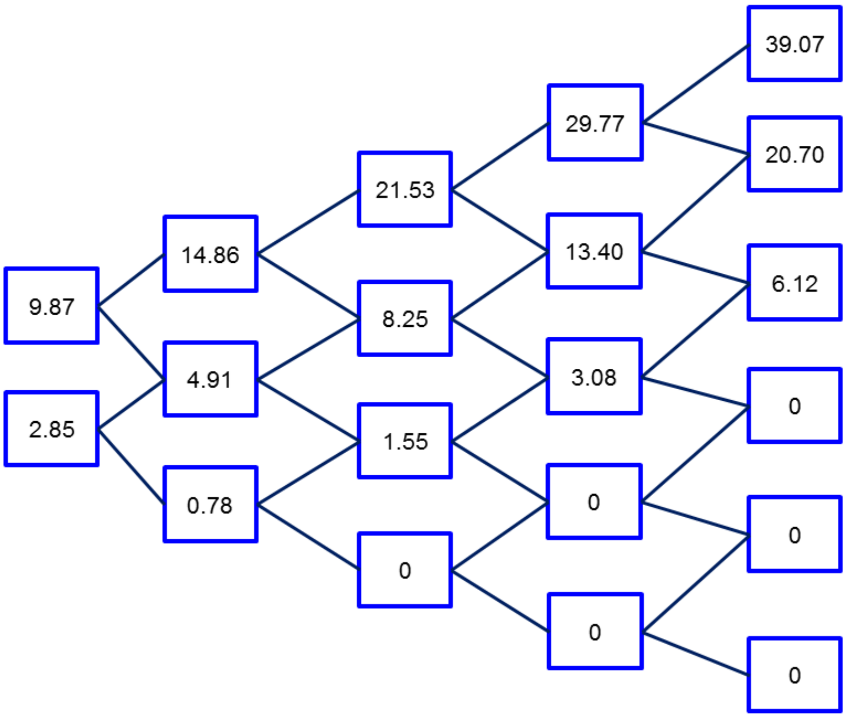

39.07
20.70
6.12
0
0
0

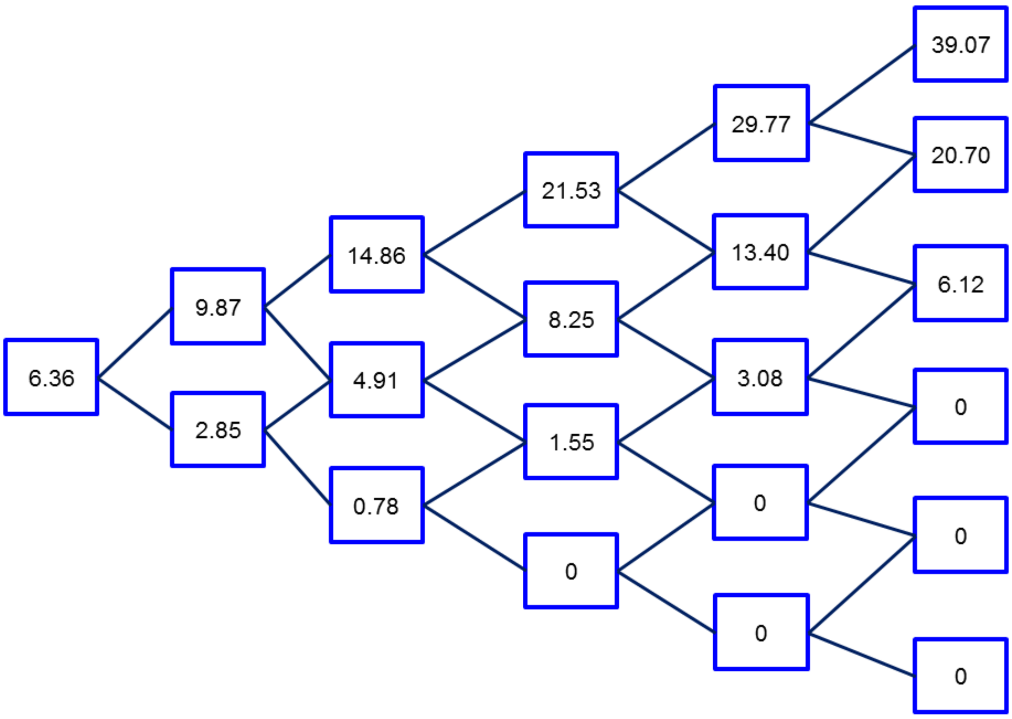

39.07
29.77
20.70
13.40
6.12
3.08
0
0
0
0
0


39.07
29.77
21.53
20.70
13.40
8.25
6.12
3.08
1.55
0
0
0
0
0
0


39.07
29.77
20.70
21.53
13.40
14.86
6.12
8.25
4.91
3.08
0
1.55
0.78
0
0
0
0
0


39.07
29.77
20.70
21.53
13.40
14.86
6.12
8.25
4.91
3.08
0
1.55
0.78
0
0
0
0
0


39.07
29.77
20.70
21.53
13.40
14.86
6.12
9.87
8.25
4.91
3.08
0
2.85
1.55
0.78
0
0
0
0
0


39.07
29.77
20.70
21.53
13.40
14.86
6.12
9.87
8.25
6.36
4.91
3.08
0
2.85
1.55
0.78
0
0
0
0
0

Pricing European Call Option by Binomial Lattice

```
function [price, lattice] = LatticeEurCall(S0,K,r,T,sigma,N)

deltaT = T/N;
u=exp(sigma * sqrt(deltaT));
d=1/u;
p=(exp(r*deltaT) - d)/(u-d);
lattice = zeros(N+1,N+1);

for i=0:N
   lattice(i+1,N+1)=max(0 , S0*(u^i)*(d^(N-i)) - K);
end

for j=N-1:-1:0
   for i=0:j
      lattice(i+1,j+1) = exp(-r*deltaT) * ...
         (p * lattice(i+2,j+2) + (1-p) * lattice(i+1,j+2));
   end
end

price = lattice(1,1);
```


Pricing American Style Put Option

```
function price = AmPutLattice(S0,K,r,T,sigma,N)
deltaT = T/N;
u=exp(sigma * sqrt(deltaT));
d=1/u;
p=(exp(r*deltaT) - d)/(u-d);
discount = exp(-r*deltaT);
p_u = discount*p;
p_d = discount*(1-p);
SVals = zeros(2*N+1,1);
SVals(N+1) = S0;

[...]
```

Pricing American Style Put Option (cont'd)

```
function price = AmPutLattice(S0,K,r,T,sigma,N)



[...]

for i=1:N
    SVals(N+1+i) = u*SVals(N+i);
    SVals(N+1-i) = d*SVals(N+2-i);
end
PVals = zeros(2*N+1,1);
for i=1:2:2*N+1
    PVals(i) = max(K-SVals(i),0);
end

[...]
```


Pricing American Style Put Option (cont'd

```
function price = AmPutLattice(S0,K,r,T,sigma,N)



[...]
for tau=1:N
    for i= (tau+1):2:(2*N+1-tau)
        hold = p_u*PVals(i+1) + p_d*PVals(i-1);
        PVals(i) = max(hold, K-SVals(i));
    end
end
price = PVals(N+1);
```

- Decisions at every point during backtracking

$$f_{i,j} \;=\; \max\left\{K - S_{i,j},\;\; \exp(-r\delta t)\left(p\, f_{i+1,j+1} \,+\, (1-p)\, f_{i,j+1}\right)\right\}$$

- We will look at inference as expectations...

$$E\left[g(X)\right] = \int_{-\infty}^{+\infty} g(x) f_x(x)\, dx$$

- Consider the integral

$$I = \int_0^1 g(x)\, dx$$

- Think of this as computing the expected value
  (of a function of a uniform random variable):

$$E\left[g(U)\right], \quad \text{where} \quad U \sim (0, 1)$$

- We approximate the integral by

$$\widehat{I}_m = \frac{1}{m} \sum_{i=1}^{m} g(U_i)$$

- Where will we use this?
- European call option

$$f = \exp(-rT)\, E\left[f_T\right]$$

- f_T is payoff at maturity T; fair price is discounted expected payoff
- $f_T = \max \left\{ 0, \quad S(0) \exp((r - \sigma^2/2)T + \sigma\sqrt{T}\epsilon) - K \right\}$

```
% BlsMC1.m
function Price = BlsMC1(S0,K,r,T,sigma,NRepl)
nuT = (r - 0.5*sigma^2)*T;
siT = sigma * sqrt(T);
DiscPayoff = exp(-r*T)*max(0, S0*exp(nuT+siT*randn(NRepl,1))-K);
Price = mean(DiscPayoff);
```

```
> S0=50; K=60; r=0.05; T=1; sigma=0.2;
> randn('state', 0);
> BlsMC1(S0, K, r, T, sigma, 1000)
ans =
   1.2562
```

Is this a good approach?

- Different answers on different runs

```
> S0=50; K=60; r=0.05; T=1; sigma=0.2;
> randn('state', 0);
> BlsMC1(S0, K, r, T, sigma, 1000)
ans =
   1.2562
> BlsMC1(S0, K, r, T, sigma, 1000)
ans =
   1.8783
> BlsMC1(S0, K, r, T, sigma, 1000)
ans =
   1.7864
```

- What if we had large number of samples?

```
> BlsMC1(S0, K, r, T, sigma, 1000000)
ans =
   1.6295
> BlsMC1(S0, K, r, T, sigma, 1000000)
ans =
   1.6164
> BlsMC1(S0, K, r, T, sigma, 1000000)
ans =
   1.6141
```


Sampling: Inverse Transform

- Sample X from $f(x)$; Cumulative distribution $F(x)$

> - Draw $U \sim U(0.1)$
> - Return $X = F^{-1}(U)$

$$
\begin{aligned}
P\{X \leq x\} &= P\left\{F^{-1}(U) \leq x\right\} \\
&= P\{U \leq F(x)\} \\
&= F(x)
\end{aligned}
$$

- Example: Exponential distribution $X \sim \exp(\mu)$
- Cumulative

$$F(x) = 1 - \exp(-\mu\, x)$$

- Inverse

$$x = -\frac{1}{\mu} \log(1 - U)$$

- Distributions of U and $(1 - U)$ are the same
  Hence return: $-\log(U)/\mu$

Sampling: Acceptance-Rejection Method

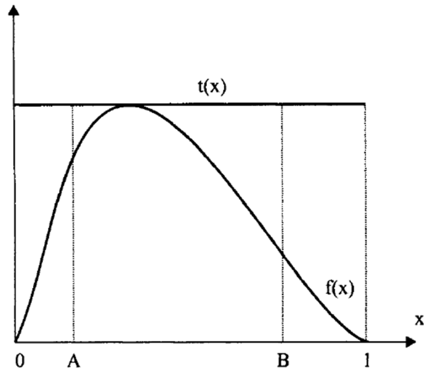
- Probability density function: $f(x)$
- Consider a known function $t(x)$, such that

$$t(x) \geq f(x), \quad \forall x \in \mathcal{I}$$

- $\mathcal{I}$ is the support for f (region in which it is defined)
- $t(x)$ is a probability density of normalized

$$r(x) = t(x)/c \quad c = \int_{\mathcal{I}} t(x)\, dx$$

1. Generate $Y \sim r$
2. Generate $U \sim U(0, 1)$
3. If $U \leq f(Y)/t(Y)$ return $X = Y$
   Else go to 1



- On average c trials to accept a sample.

Homework
Page 235, Brandimarte

- $f(x) = 30(x^2 - 2x^3 + x^4), \quad x \in [0, 1]$
- Algorithm

1. Draw U_1 and U_2
2. If $U2 \leq 16(U_1^2 - 2U_1^3 + U_1^4)$
   accept $X = U_1$
   Else
   go to 1

- Exercise:
  - Draw the graph of $f(x)$
  - Simulate 1000 samples using above algorithm
  - Draw a histogram to the same scale as $f(x)$ – do they match? Is it better with 100000 samples?
  - On average, how many trials were needed through the accept-reject loop for each sample?

Variance Reduction

- Independent samples X_i
- Sample mean (estimates mean $\mu = E[X_i]$ from n samples)

$$\overline{X}(n) = \frac{1}{n}\sum_{i=1}^{n} X_i$$

- Sample variance

$$S^2(n) = \frac{1}{(n-1)}\sum_{i=1}^{n}\left[X_i - \overline{X}(n)\right]^2$$

- Error of the estimator

$$\begin{aligned} E\left[(\overline{X}(n) - \mu)^2\right] &= \mathrm{Var}\left[\overline{X}(n)\right] \\ &= \mathrm{Var}\left[\frac{1}{n}\sum_{i=1}^{n} X_i\right] \\ &= \frac{1}{n^2} \times n \times \sigma^2 = \frac{\sigma^2}{n} \end{aligned}$$

- Two points:
  - More samples n reduces the variance in estimation
  - Variance reduction schemes can control σ^2

Variance reduction: Antithetic Sampling

- Pair of sequences

$$\left\{ \begin{array}{cccc} X_1^{(1)} & X_1^{(2)} & \dots & X_1^{n} \\ X_2^{(1)} & X_2^{(2)} & \dots & X_2^{n} \end{array} \right\}$$

- Columns (horizontally) are independent
- $X_1^{(i)}$ and $X_2^{(i)}$ are dependent.
- Sample is a function of each pair: $X^{(i)} = \left(X_1^{(i)} + X_2^{(i)}\right)/2$
- Variance

$$\begin{aligned} \mathrm{Var}\left[\overline{X}(n)\right] &= \frac{1}{n}\mathrm{Var}\left[X^{(i)}\right] \\ &= \frac{1}{4n}\left\{\mathrm{Var}(X_1^{(i)}) + \mathrm{Var}(X_2^{(i)}) + 2\,\mathrm{Cov}(X_1^{(i)}, X_2^{(i)})\right\} \\ &= \frac{1}{2n}\mathrm{Var}(X)\,(1 + \rho) \end{aligned}$$

- Uniform random number $\{U_k\}$ and $\{1 - U_k\}$ as sequences.

Application

```
function [Price, CI] = BlsMC2(S0,K,r,T,sigma,NRepl)
nuT = (r - 0.5*sigma^2)*T;
siT = sigma * sqrt(T);
DiscPayoff = exp(-r*T)*max(0, S0*exp(nuT+siT*randn(NRepl,1))-K);
[Price, VarPrice, CI] = normfit(DiscPayoff);
```

```
function [Price, CI] = BlsMCAV(S0,K,r,T,sigma,NPairs)
nuT = (r - 0.5*sigma^2)*T;
siT = sigma * sqrt(T);
Veps = randn(NPairs,1);
Payoff1 = max( 0 , S0*exp(nuT+siT*Veps) - K);
Payoff2 = max( 0 , S0*exp(nuT+siT*(-Veps)) - K);
DiscPayoff = exp(-r*T) * 0.5 * (Payoff1+Payoff2);
[Price, VarPrice, CI] = normfit(DiscPayoff);
```


Homework

Test the two functions: `BlsMC` and `BlsMCAV`

(Brandimarte, p248)

```
> randn('state', 0)
> [Price, CI] = BlsMC2(50,50,0.05,1,0.4,200000)
Price=
          9.0843
CI =
          9.0154
          9.1532
\pause
> (CI(2)-CI(1))/Price
ans =
        0.0152
\pause
> randn('state', 0)
> [Price, CI] = BlsMCAV(50,50,0.05,1,0.4,200000)
Price=
          9.0553
CI =
           8.9987
           9.1118
\pause
> (CI(2)-CI(1))/Price
ans =
```

Assignment 3

- We have seen three tools for pricing options
  - Closed form Black-Scholes
  - Binomial lattice
  - Monte Carlo

- How well can the relationship between asset price and option price be approximated?

Hutchinson *et al.* (1994) "A nonparametric approach to pricing and hedging derivative securities via learning networks", *Journal of Finance* **49**(3): 851

$$\boldsymbol{x} = [\, S/X \quad (T-t)\,]^T$$

$$c = \sum_{j=1}^{J} \lambda_j \, \phi_j(\boldsymbol{x}) + \boldsymbol{w}^T \boldsymbol{x} + w_0$$

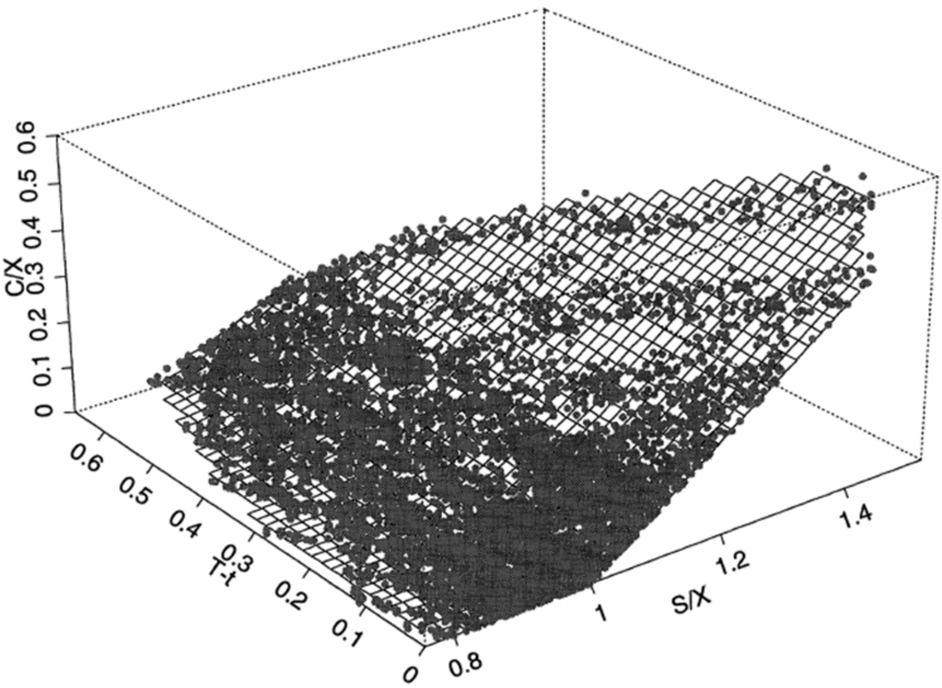


Figure 4. Simulated call option prices normalized by strike price and plotted versus

$$
\begin{aligned}
\widehat{C/X} = &-0.06\sqrt{\begin{bmatrix} S/X - 1.35 \\ T - t - 0.45 \end{bmatrix}' \begin{bmatrix} 59.79 & -0.03 \\ -0.03 & 10.24 \end{bmatrix} \begin{bmatrix} S/X - 1.35 \\ T - t - 0.45 \end{bmatrix} + 2.55} \\
&- 0.03\sqrt{\begin{bmatrix} S/X - 1.18 \\ T - t - 0.24 \end{bmatrix}' \begin{bmatrix} 59.79 & -0.03 \\ -0.03 & 10.24 \end{bmatrix} \begin{bmatrix} S/X - 1.18 \\ T - t - 0.24 \end{bmatrix} + 1.97} \\
&+ 0.03\sqrt{\begin{bmatrix} S/X - 0.98 \\ T - t + 0.20 \end{bmatrix}' \begin{bmatrix} 59.79 & -0.03 \\ -0.03 & 10.24 \end{bmatrix} \begin{bmatrix} S/X - 0.98 \\ T - t + 0.20 \end{bmatrix} + 0.00} \\
&+ 0.10\sqrt{\begin{bmatrix} S/X - 1.05 \\ T - t + 0.10 \end{bmatrix}' \begin{bmatrix} 59.79 & -0.03 \\ -0.03 & 10.24 \end{bmatrix} \begin{bmatrix} S/X - 1.05 \\ T - t + 0.10 \end{bmatrix} + 1.62} \\
&+ 0.14S/X - 0.24(T - t) - 0.01. \qquad (9)
\end{aligned}
$$