Reinforcement Learning in Finance

Week 1: Reinforcement Learning

1-2-1-Option-pricing-as-MDP

Igor Halperin

NYU Tandon School of Engineering, 2017

Corporate defaults: The Merton model

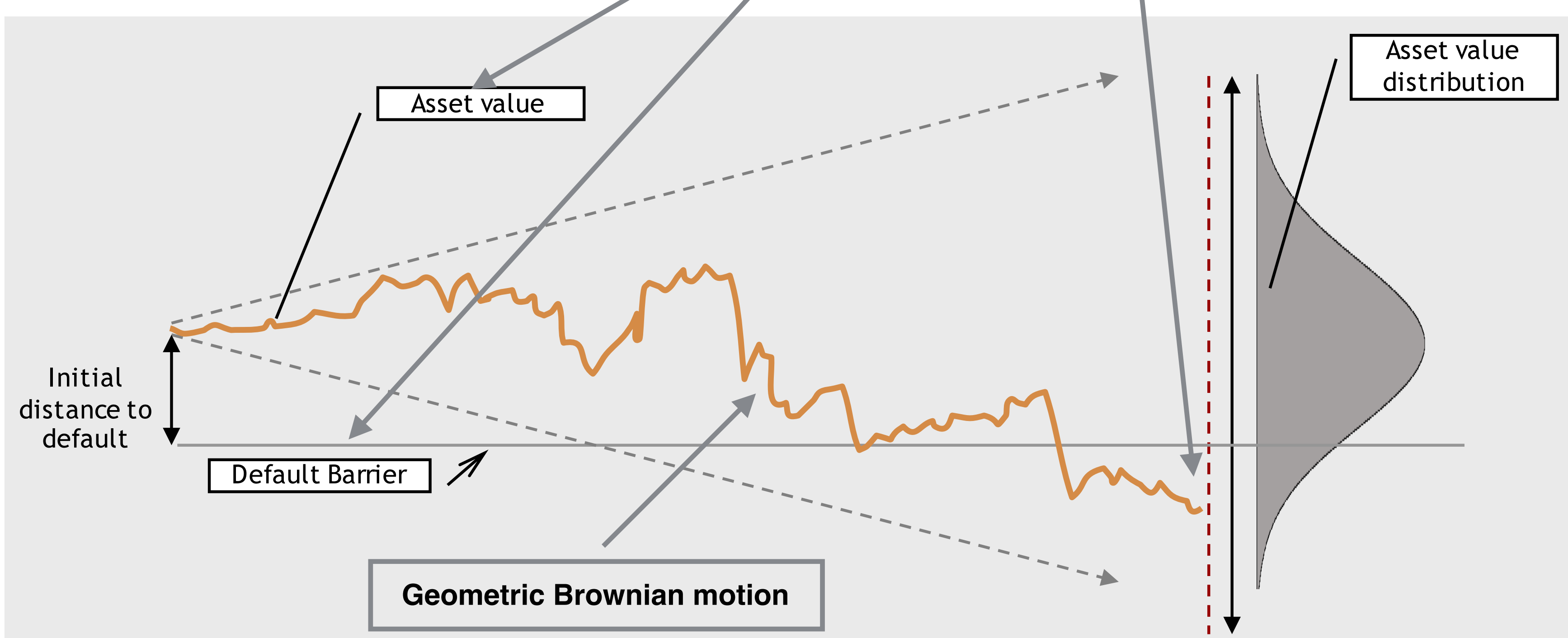
The **Merton model** of corporate defaults (1974-present) is the most popular modeling framework, used as a benchmark for many studies.

A firm is run by equity holders. At time T, they pay the face value of the debt D if the firm (asset) value is larger than D, and keep the remaining amount. If the firm value at time T is less than D, bond holders take over, and recover a "recovery" value V_T, while equity holders get nothing:

$$B_T = \min(V_T, D) = D - \max(D - V_T, 0)$$

$$S_T = \max(V_T - D,\ 0)$$

Limited liability of equity holders



Merton model as a structural default model

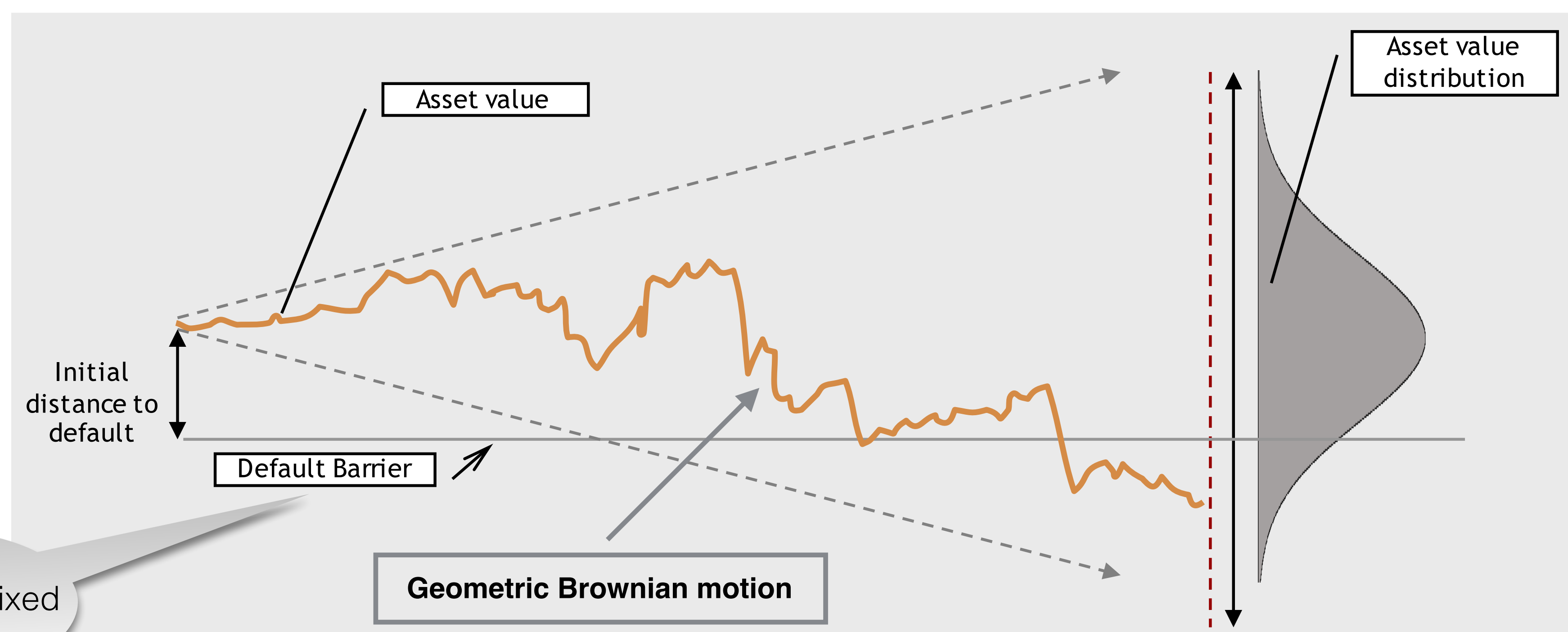
Default probability in the Merton model:

Probabilistic structural model

$$\Pr(default) = \mathbb{E}\left[\mathbb{I}_{V_T < D}\right] = \Pr(V_T < D) = N(-d_2)$$

$$d_2 = \frac{\log \frac{V_t}{D} + \left(r - \frac{\sigma_V^2}{2}\right)(T-t)}{\sigma_V \sqrt{(T-t)}}$$

Depends only on the Assets/Debt ratio and asset volatility



The barrier is fixed and static

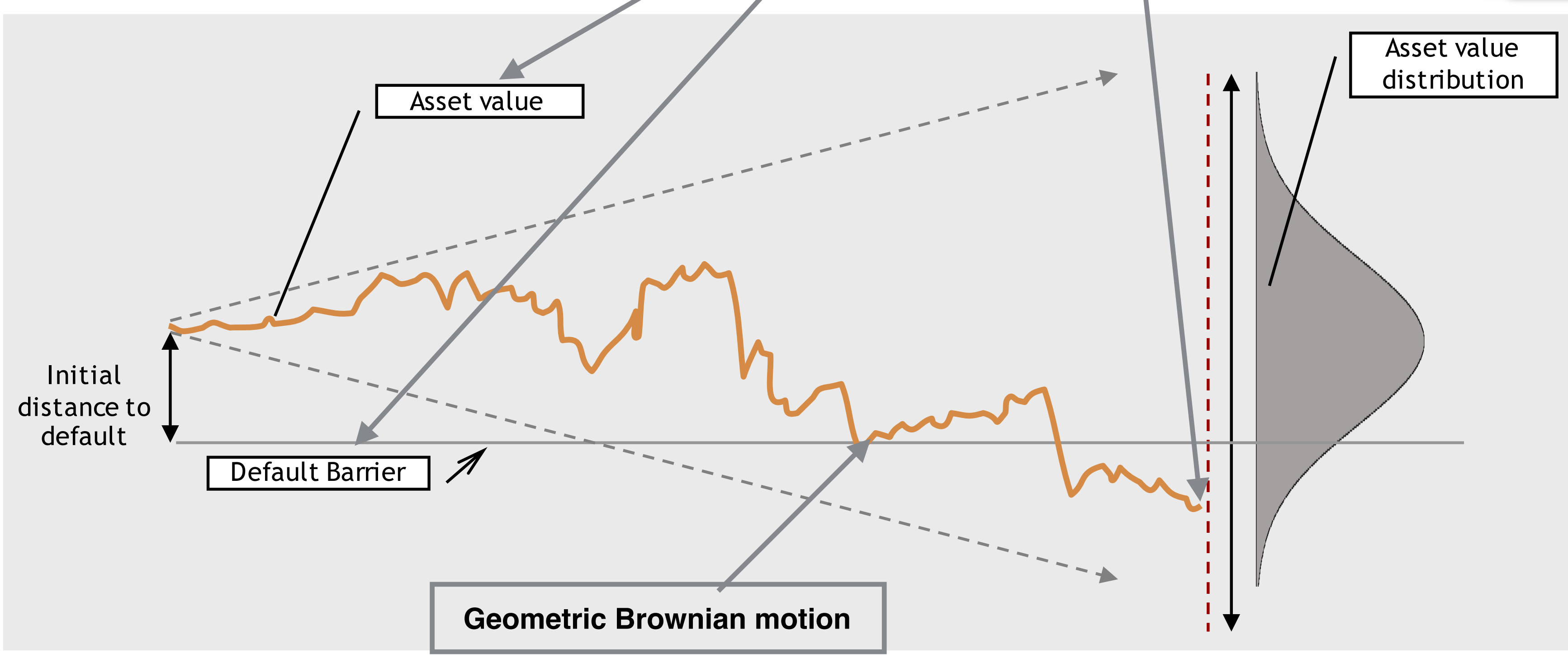
Option pricing: Black-Scholes-Merton

The equity holder offers you to buy her **future** (to be received at time T) stock **now**.
Currently the firm costs $V_0 = \$1\text{m}$, and has debt $D = \$600\text{K}$.
How much should you pay **now**?

$$B_T = \min(V_T, D) = D - \underbrace{\max(D - V_T, 0)}$$

$$S_T = \max(V_T - D,\ 0)$$

Limited liability of equity holders



Stock option: replace the firm value by the stock price!
Stock option is a contract to get a stock for a pre-determined price at some future time

Option pricing: Black-Scholes-Merton

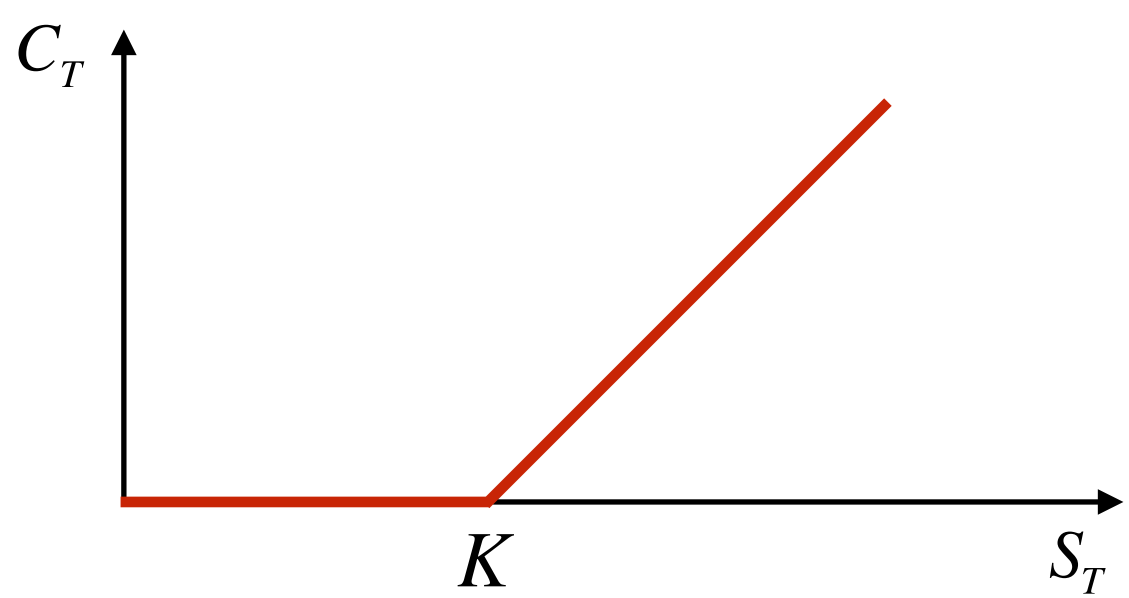
Stock option is a contract to get a stock S_T for a pre-determined price K at some future time T
E.g. the current stock price is $S_0 = \$100$, and the strike $K = \$110$.
How much should you pay **now**?

$$C_T = \max(S_T - K,\ 0)$$

Option payoff at maturity T

Stock price at maturity T

Option strike K



The BSM model: lognormal stock dynamics

Lognormal process for the stock price in continuous time in the BSM model (a Geometric Brownian motion with a drift):

$$\frac{dS_t}{S_t} = \mu\, dt + \sigma\, dW_t$$

Equivalently (by Ito's lemma!):

$$d \log S_t = \left(\mu - \frac{\sigma^2}{2} \right) dt + \sigma\, dW_t$$

μ - the drift, σ - volatility rate, W_t - standard Brownian motion

Discretize the stock price

Introduce a new variable :

$$X_t = -\left(\mu - \frac{\sigma^2}{2}\right)t + \log S_t$$

Then :

$$d\,X_t = -\left(\mu - \frac{\sigma^2}{2}\right)dt + d\log S_t = \sigma\,dW_t$$

Therefore X_t is a standard Brownian motion, scaled by volatility σ
If we know the value of X_t in a given scenario, then S_t is also known:

$$S_t = e^{X_t + \left(\mu - \sigma^2/2\right)t}$$

Discrete-time approximation:

$$\Delta X_t = \sigma\sqrt{\Delta t}\,\varepsilon_t, \quad \varepsilon_t \sim N(0,1)$$

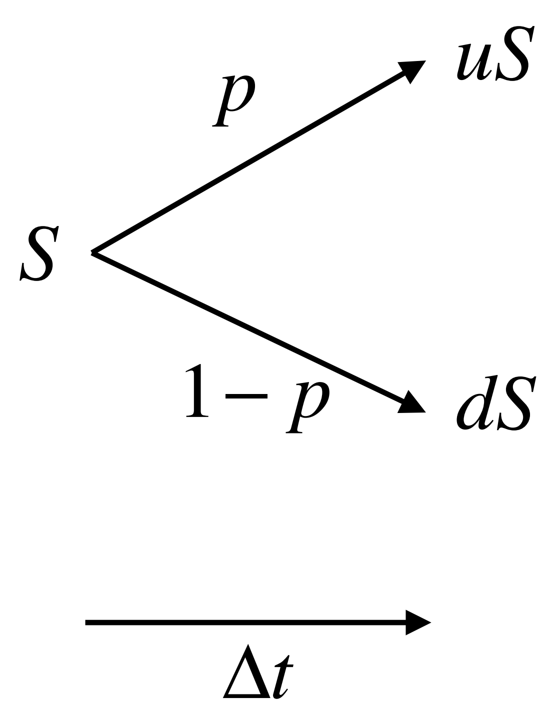
Binomial tree approximation

Discrete-time approximation for continuous-space dynamics:

$$S_t = e^{X_t + \left(\mu - \sigma^2/2\right)t}$$

$$\Delta X_t = \sigma \sqrt{\Delta t}\, \varepsilon_t, \quad \varepsilon_t \sim N(0,1)$$

The binomial model gives a discrete-state approximation to this dynamics. The stock can risk to a value uS with probability p, or fall to a value dS with probability $1-p$, with $0 < d < 1 < u$



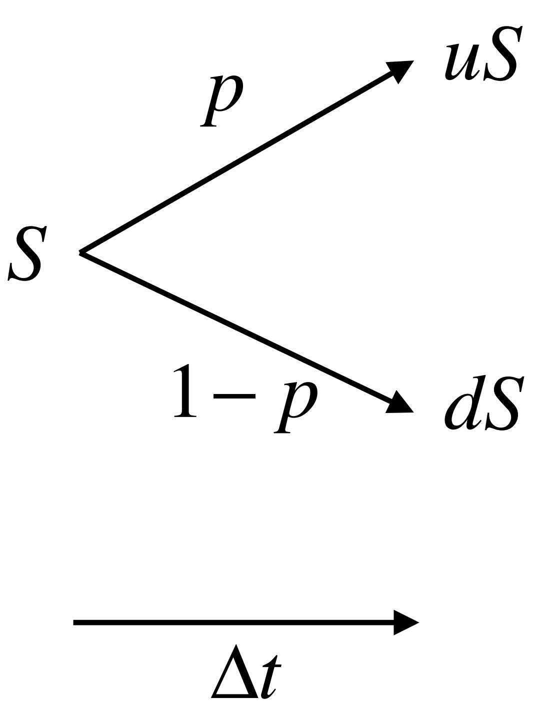
Binomial model: choice of parameters

We can choose three parameters to match the mean and standard deviation of the future stock price.

Mean $E[S] = Se^{r\Delta t} = puS + (1-p)dS$

Variance: $Var[S] = S^2 e^{2r\Delta t}\left(e^{\sigma^2 \Delta t} - 1\right) = S^2\left(pu^2 + (1-p)d^2\right)$

Have two constraints for three parameters:



$$pu + (1-p)d = e^{r\Delta t}$$

$$pu^2 + (1-p)d^2 = e^{(2r+\sigma^2)\Delta t}$$

Popular choices:

Cox, Ross, Rubinstein: $u = \dfrac{1}{d}$

Jarrow, Rudd: $p = \dfrac{1}{2}$

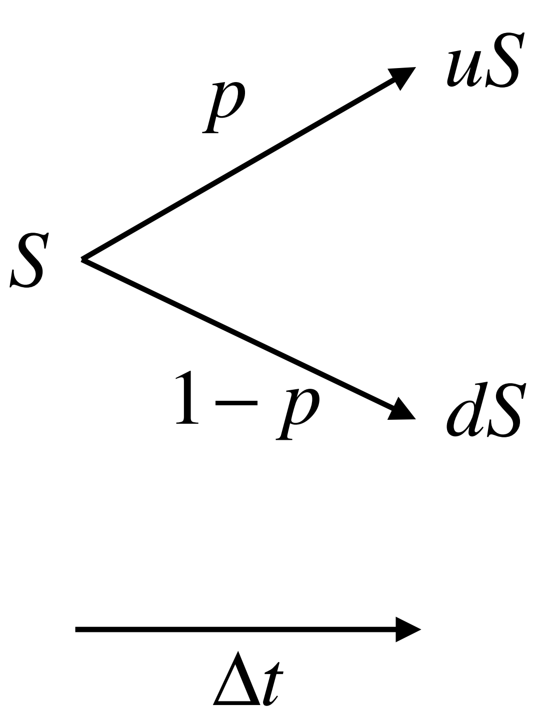
Binomial model: Cox, Ross, Rubinstein

Have two constraints for three parameters:

$$pu + (1-p)d = e^{r\Delta t}$$

$$pu^2 + (1-p)d^2 = e^{(2r+\sigma^2)\Delta t}$$

Cox, Ross, Rubinstein (CRR): $u = \dfrac{1}{d}$



$$u = e^{\sigma\sqrt{\Delta t}}$$

$$d = e^{-\sigma\sqrt{\Delta t}}$$

$$p = \frac{a-d}{u-d}$$

$$a = e^{r\Delta t}$$

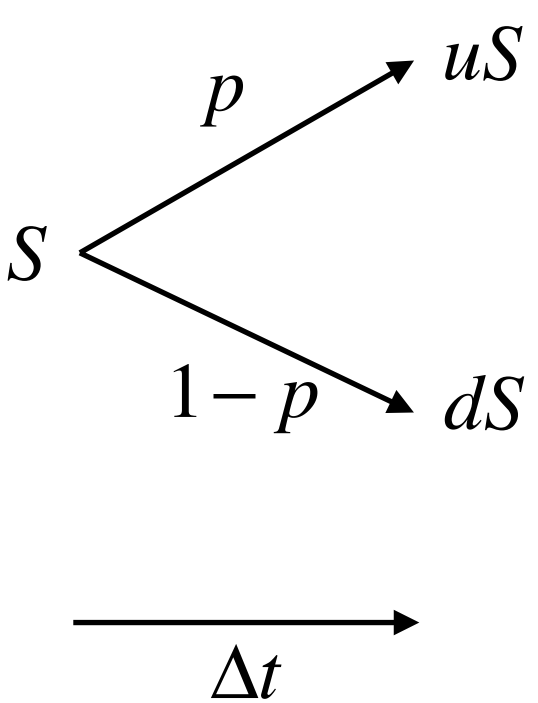
Binomial model: Jarrow-Rudd

Have two constraints for three parameters:

$$pu + (1-p)d = e^{r\Delta t}$$

$$pu^2 + (1-p)d^2 = e^{(2r+\sigma^2)\Delta t}$$

Jarrow-Rudd: $p = \dfrac{1}{2}$



$$u = e^{(r-\sigma^2/2)\Delta t + \sigma\sqrt{\Delta t}}$$

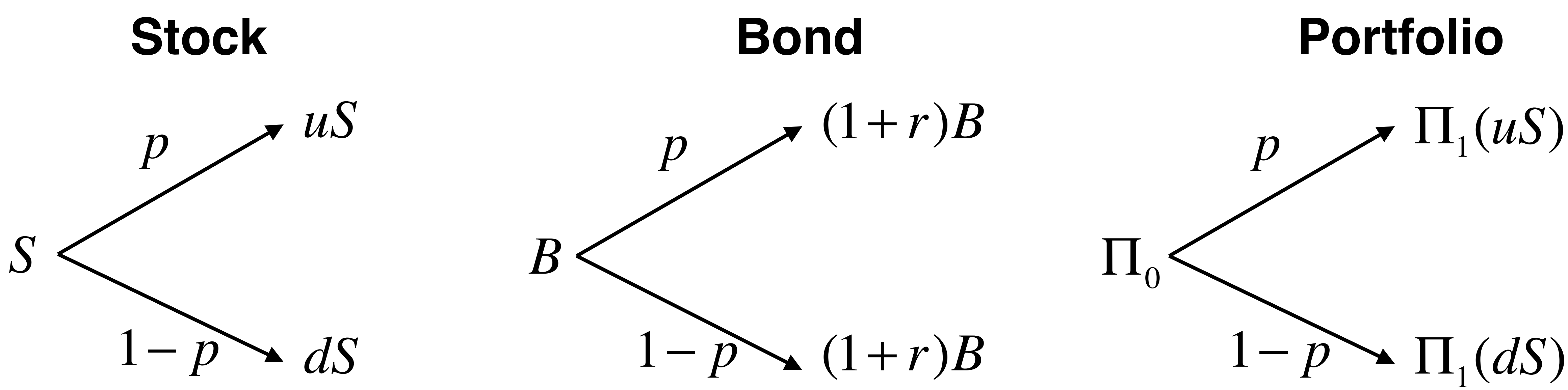
$$d = e^{(r-\sigma^2/2)\Delta t - \sigma\sqrt{\Delta t}}$$

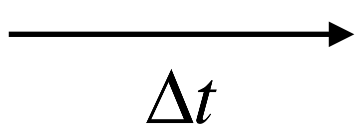
$$p = \frac{a-d}{u-d} = \frac{1}{2}$$

$$a = e^{r\Delta t}$$

Binomial model: pricing by replication

Arbitrage pricing: price the stock by constructing a replicating portfolio of a stock and a bond.





$$\Pi_0 = \Delta S + \theta B$$

$$\Pi_1(uS) = \Delta uS + \theta(1+r)B$$

$$\Pi_1(dS) = \Delta dS + \theta(1+r)B$$

Binomial model: pricing by replication

Arbitrage pricing: price the stock by constructing a replicating portfolio of a stock and a bond.

Stock option

$$\begin{tikzcd}[column sep=large] & C_1(uS) = \max(uS - K, 0) \\ C_0 \arrow[ur, "p"] \arrow[dr, "1-p"'] & \\ & C_1(dS) = \max(dS - K, 0) \end{tikzcd}$$

Portfolio

$$\begin{tikzcd}[column sep=large] & \Pi_1(uS) \\ \Pi_0 \arrow[ur, "p"] \arrow[dr, "1-p"'] & \\ & \Pi_1(dS) \end{tikzcd}$$

price matched in every state of the world

$$C_0 = \Pi_0 = \Delta S + \theta B$$

$$C_1(uS) = \Pi_1(uS) = \Delta uS + \theta(1+r)B$$

$$C_1(dS) = \Pi_1(dS) = \Delta dS + \theta(1+r)B$$

$$\Pi_0 = \Delta S + \theta B$$

$$\Pi_1(uS) = \Delta uS + \theta(1+r)B$$

$$\Pi_1(dS) = \Delta dS + \theta(1+r)B$$

Binomial model: optimal replication

Arbitrage pricing: price the stock by constructing a replicating portfolio of a stock and a bond.

Stock option

$$
\begin{tikzcd}
 & C_1(uS) = \max(uS - K, 0) \\
C_0 \arrow[ur, "p"] \arrow[dr, "1-p"'] & \\
 & C_1(dS) = \max(dS - K, 0)
\end{tikzcd}
$$

Portfolio

$$
\begin{tikzcd}
 & \Pi_1(uS) \\
\Pi_0 \arrow[ur, "p"] \arrow[dr, "1-p"'] & \\
 & \Pi_1(dS)
\end{tikzcd}
$$

$$
\begin{aligned}
C_0 &= \Pi_0 = \Delta S + \theta B \\
C_1(uS) &= \Pi_1(uS) = \Delta uS + \theta(1+r)B \\
C_1(dS) &= \Pi_1(dS) = \Delta dS + \theta(1+r)B
\end{aligned}
$$

$$
\Delta = \frac{C_1(uS) - C_1(dS)}{uS - dS}
$$

$$
\theta B = \frac{uC_1(uS) - dC_1(dS)}{(1+r)(u-d)}
$$

Choosing the **control** (Delta) this way completely **eliminates risk** of the option! This happens **only** for the binomial model in discrete time, and for the BSM in continuous time!

Constructing the Markov chain

Re-cap: We introduced a new variable :

$$X_t = -\left(\mu - \frac{\sigma^2}{2}\right)t + \log S_t$$

Then :

$$d\,X_t = -\left(\mu - \frac{\sigma^2}{2}\right)dt + d\log S_t = \sigma\, dW_t$$

Therefore X_t is a standard Brownian motion, scaled by volatility σ
If we know the value of X_t in a given scenario, then S_t is also known:

$$S_t = e^{X_t + \left(\mu - \sigma^2/2\right)t}$$

Discrete-time approximation:

$$\Delta X_t = \sigma\sqrt{\Delta t}\,\varepsilon_t, \quad \varepsilon_t \sim N(0,1)$$

As X_t is a stationary process without a drift (a martingale!), it stays around the current value in the sense of expectations, thus is easier to discretize!

Constructing the Markov chain

Instead of a tree, we want to discretize the dynamics to a Markov Chain (first-order Markov process with discrete states), to convert this problem to a Markov Decision Problem (MDP)

Given: initial stock price S_0
Want: create a set of representative logarithmic stock prices

$$\left[\log S_0 - I_p, \log S_0 + I_p\right]$$

$$I_p = \delta(m)\sigma\sqrt{T\Delta t}$$

Here (Duan and Simonato, 2001)

Δt - the time step (in years)
m - the number of discrete states (should be an odd number)
T - the option maturity (in years)
σ - stock volatility (in annualized terms)
$\delta(m)$- a scaling factor, pick $\delta(m) = 2 + \log(\log(m))$

Grid:
$$p_i = \log S_0 + \frac{2i - m - 1}{m - 1} I_p, \quad i = 1, \ldots, m$$

Markov chain: transition probabilities

Create cells separated by values of the grid:

$$C_1 = (c_1, c_2), C_i = [c_i, c_{i+1}), i = 2,\ldots,m$$

Here (Duan and Simonato, 2001)

$$c_1 = -\infty, c_i = \frac{p_i + p_{i-1}}{2}, i = 2,\ldots,m, c_{m+1} = +\infty,$$

$$p_i = \log S_0 + \frac{2i - m - 1}{m - 1} I_p, \quad i = 1,\ldots,m$$

Midpoints of cells are the values

Transition probabilities between the cells:

$$p_{ij} = N\left(\frac{c_{j+1} - p_i - (\mu - 0.5\sigma^2)\Delta t}{\sigma\sqrt{\Delta t}}\right) - N\left(\frac{c_j - p_i - (\mu - 0.5\sigma^2)\Delta t}{\sigma\sqrt{\Delta t}}\right)$$

Here $N(x)$ is the cumulative normal distribution

Alternative approach: match moments of the lognormal process (Kushner, 1990)