Ultimate Implementation and Analysis of the AMO Algorithm for Approximate Pricing of European-Asian Options

Akiyoshi Shioura
(Tohoku University)

joint work with T. Tokuyama

Summary of This Talk

- option：typical financial derivative
- pricing European-Asian option on binomial model

  --- difficult to compute accurately

  ⇒ approximation
- Aingworth, Motwani & Oldham (SODA00)

  time: $O(kn^2)$, absolute error $O(nX/k)$
- Our Algorithm:

  time: $O(kn^2)$, absolute error $O(X/k)$

n, X: problem parameters, k: time-error tradeoff param.

Option

- option: right to sell (or buy)
  some financial asset (e.g., stock)
  at some point in the future (expiration date)
  for a specified price (strike price)

- gain more benefit by investment
- hedge risk from the fluctuation of stock price

Payoff of Option

Example: option to buy a stock of Google Inc. at the year-end at \$200

- **stock price goes up to \$220 at the year-end**
  ⇒exercise option to buy the stock at \$200
  ⇒sell it for \$220 ⇒ gain \$20 (payoff)
- **stock price goes down to \$170**
  ⇒do not exercise option ⇒ payoff = \$0

Payoff of European Option:

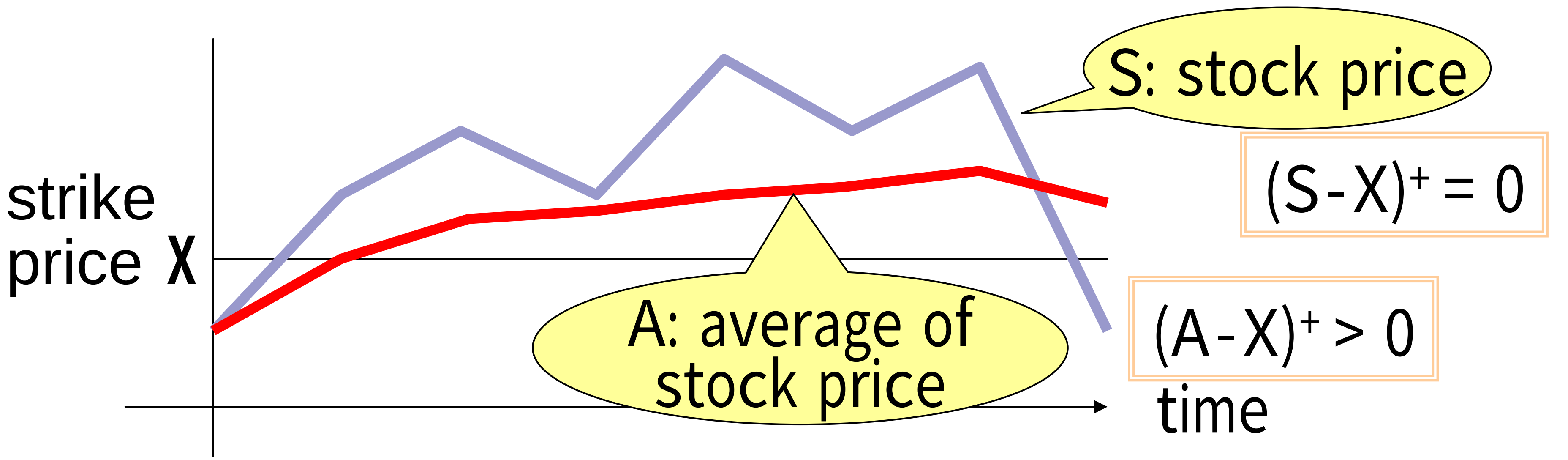
$$(S - X)^{+} = \max\{S - X, 0\}$$

(S：stock price at expiration date,　X：strike price)

European-Asian Option

- payoff of European-Asian option
  depends on average of stock price A
  during whole period

payoff: $(A - X)^+ = \max\{A - X, 0\}$



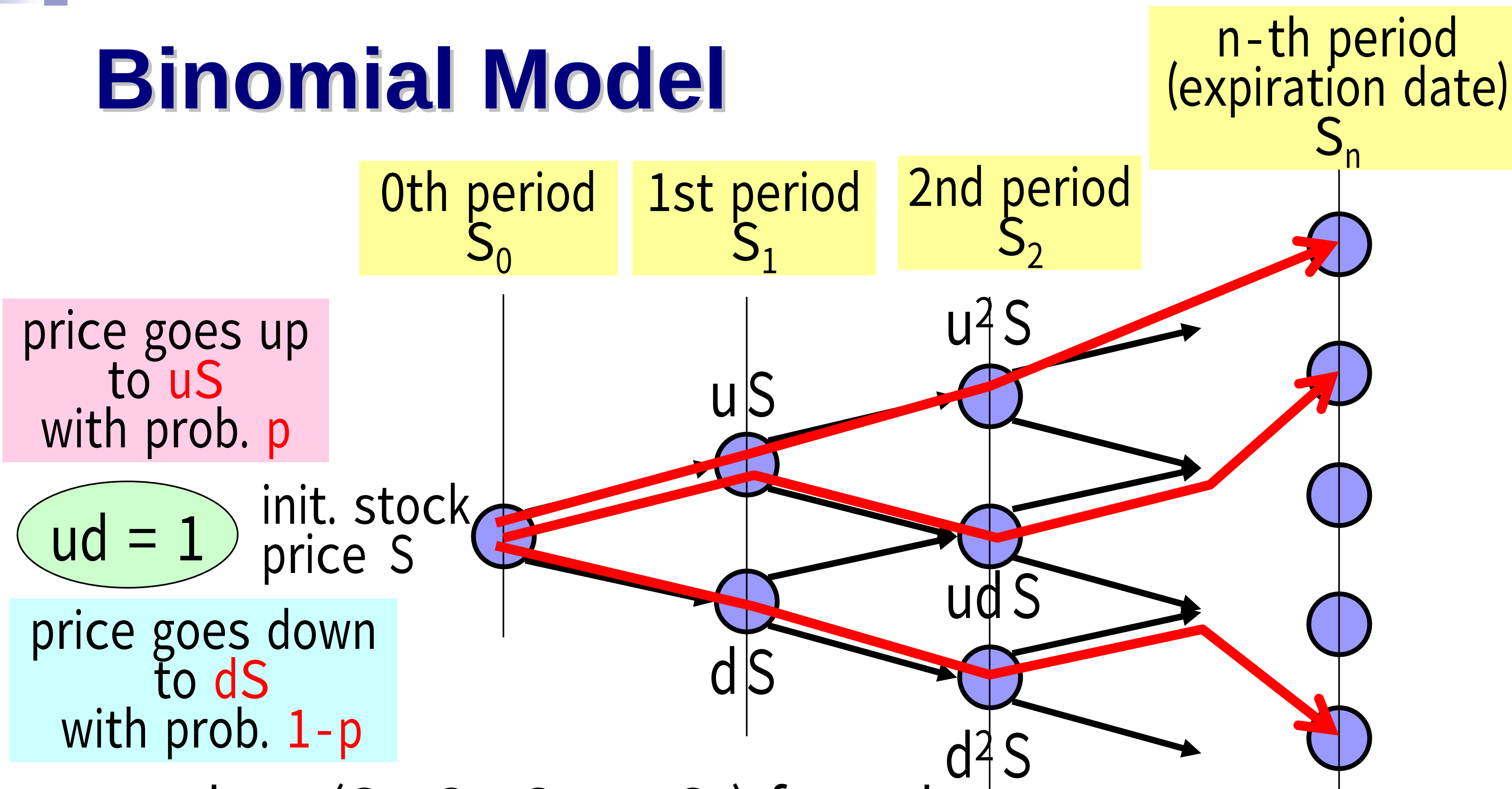
safe against fluctuation of stock price

Computation of Option Price

- price of option = discounted expected value of payoff
  --- need to model the movement of stock price

- Our model: binomial model (discrete model)
  - **proposed by** Cox, Ross & Rubinstein (1979)
  - **represent stock price movement by a binomial tree**
  - **can compute exact option price by D P**

Binomial Model



- a path $\mathbf{P}=(S_0, S_1, S_2, \ldots, S_n)$ from the root to a leaf represents the movement of stock price
- payoff of European-Asian option = $\left(\frac{\sum_{i=0}^{n} S_i}{n+1} - X\right)^+$

Our Problem

compute the expected payoff of European-Asian option on the binomial model

$$E\left(\left(\frac{\sum_{i=0}^{n} S_i}{n+1} - X\right)^{+}\right)$$

- payoff is dependent on the path **P**=(S_0, S_1, S_2, ..., S_n) (path-dependent option)
- payoff is nonlinear w.r.t. the running total $\Sigma_i S_i$
  ⇒ need enumeration of all the paths
  ⇒ exponential time
- computation of the price of path-dependent option is #P-hard

Approximation Algorithms for Pricing European-Asian Option

- Monte Carlo Method
  - based on path sampling
  - error bound depends on the volatility of stock price

- Other methods
  - based on heuristics
  - no theoretical error bound

AMO Algorithm and its Variants

Aingworth, Motwani & Oldham (2000)
time: $O(kn^2)$
abs. err.: $O(nX/k)$
DP + bucketing

independent of volatility

Dai, Huang & Lyuu (2002)
abs. err.: $O\left(\frac{\sqrt{n}X}{k}\right)$
adjust # of buckets

Ohta, Sadakane, Shioura & Tokuyama (2002)
abs. err.: $O\left(\frac{n^{\frac{1}{4}}X}{k}\right)$
randomization

Our Result

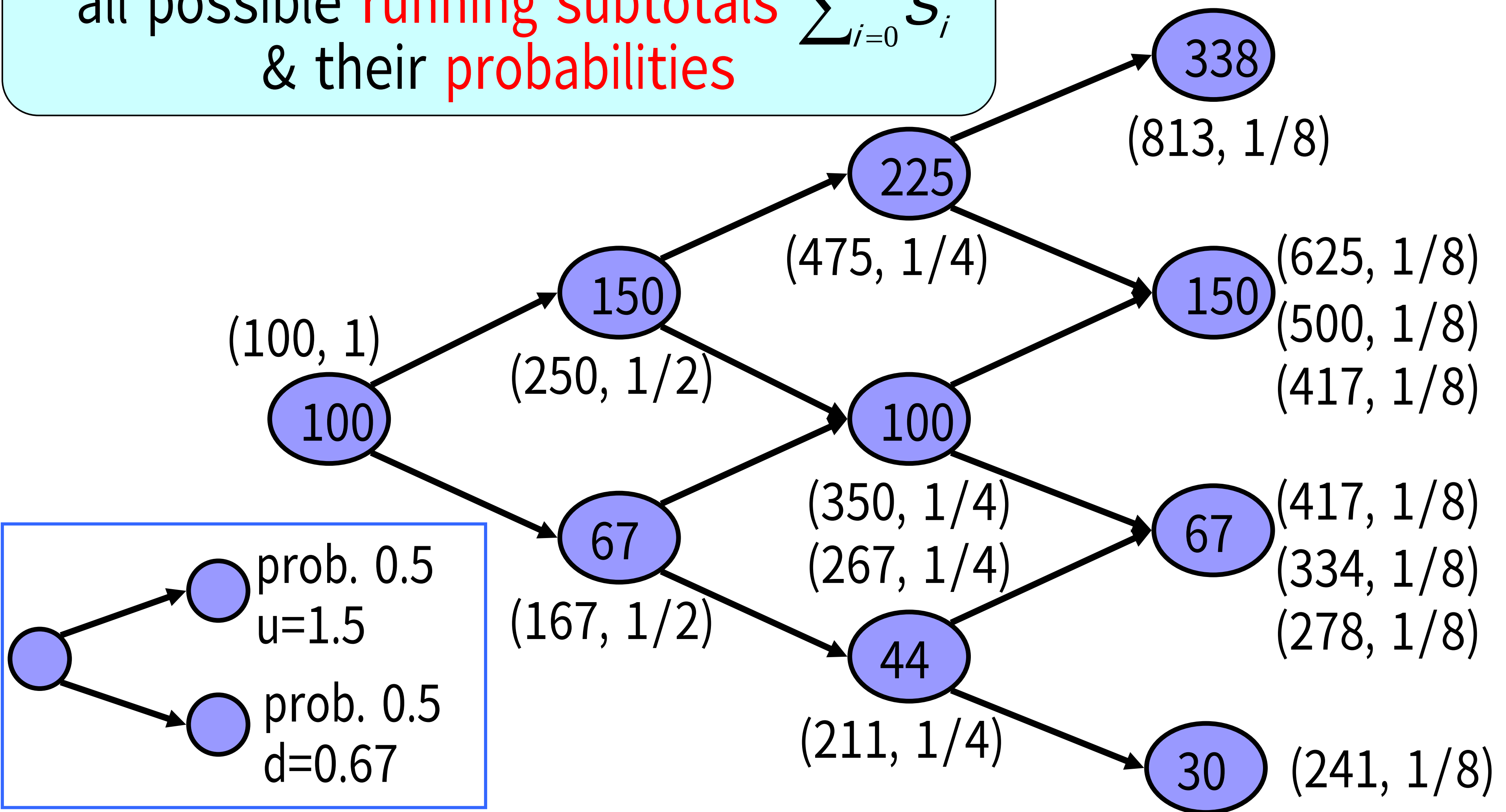
Shioura & Tokuyama (2004)
abs. err.: $O\left(\frac{X}{k}\right)$
use both ideas

n disappears!

(n : depth of binomial tree, X: strike price, k: positive integer)

Exact Algorithm by DP

at each node of binomial tree, compute all possible running subtotals $\sum_{i=0}^{t} S_i$ & their probabilities



AMO Algorithm (1)

- # of running subtotals can be exponential

⇒ approximate running subtotals by bucketing

interval	running subtotal & probability
400 300	(310, 0.05)
300 200	(205, 0.15) (240, 0.12) (285, 0.20)
200 100	(170, 0.10) (150, 0.10) (110, 0.10)
100 0	(80, 0.05) (30, 0.01)

round up
running subtotals
&
sum up
probabilities
in each bucket

400 300	(400, 0.05)
300 200	(300, 0.47)
200 100	(200, 0.30)
100 0	(100, 0.06)

AMO Algorithm (2)

- k: # of buckets at each node
  ⇒ error bound ≦ max. value of running subtotal/k

Proposition:
running subtotal $\sum_{i=0}^{t} S_i$ is ≧ (n+1)X
at the t-th period

⇓

- option will be exercised at the expiration date
- conditional expectation of the payoff
  can be computed easily

⇒ error bound of AMO algorithm = (n+1) X/k

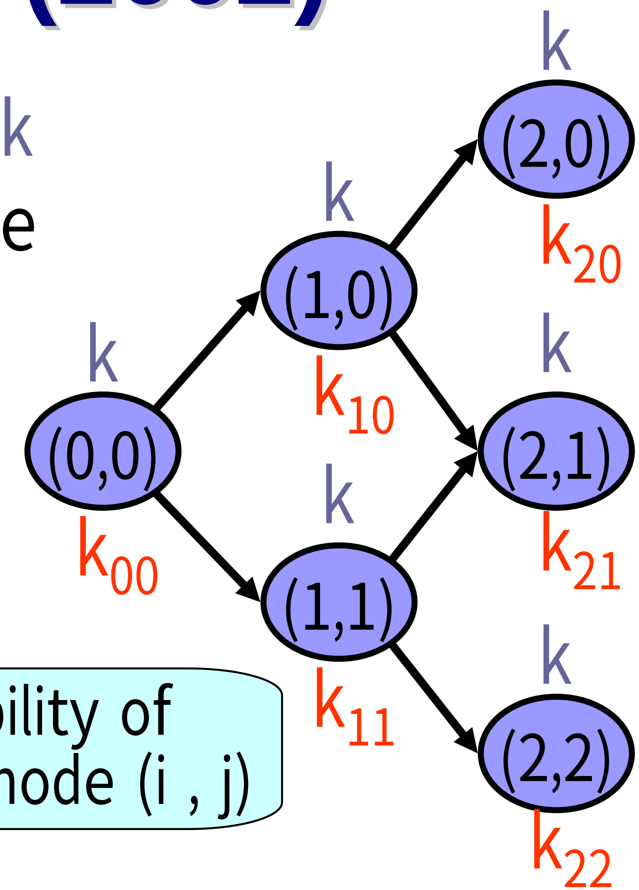
Algorithm by Dai et al. (2002)

AMO algorithm：use the same number k of buckets at each node

⇒ set the number of buckets k_{ij} at the node (i, j) flexibly

error bound $\sum_{i=0}^{n}\sum_{j=0}^{i}\frac{X}{k_{ij}}\omega(i,j)$

(probability of reaching node (i , j))



- **adjust # of buckets k_{ij} to minimize error bound under the condition $\Sigma k_{ij} = kn^2$**

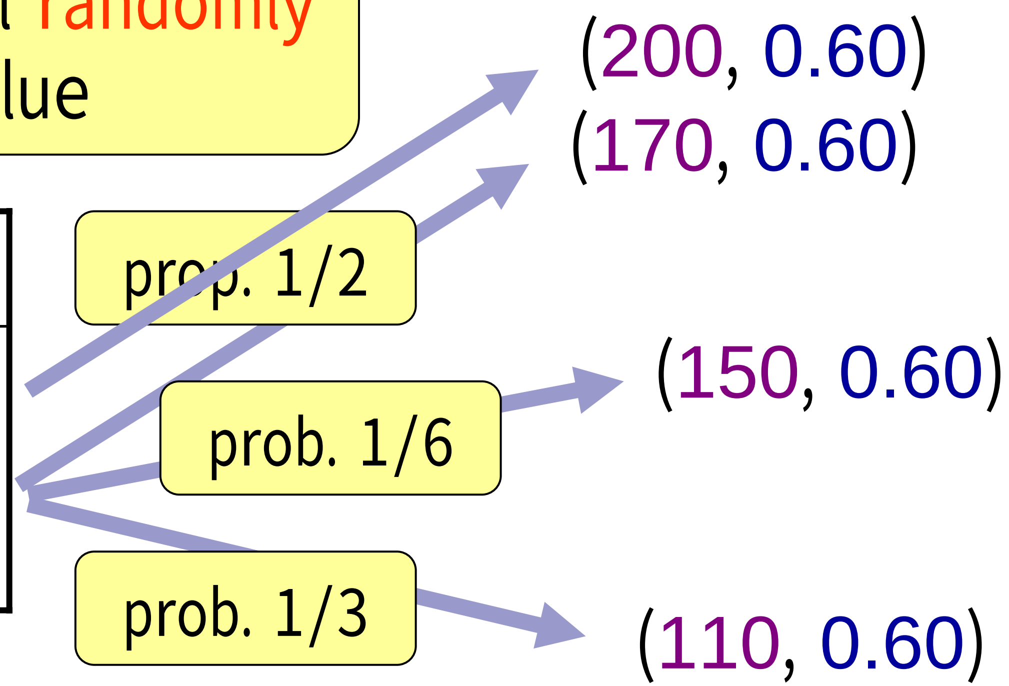
⇒ error bound $O\left(\frac{\sqrt{n}X}{k}\right)$

Algorithm by Ohta et al. (2002)

AMO algorithm：approximate running subtotals in a bucket by rounding-up

choose a running subtotal randomly as approximate value

interval	running subtotal & probability
200	(170, 0.30)
	(150, 0.10)
100	(110, 0.20)



Analysis of Ohta et al. (2002)

- regard the behavior of randomized algorithm
  as stochastic process ⇒ Martingale
  expectation of the error by random choice
  of running totals at a node = 0

⇒ apply Azuma's inequality (1967)

error bound $O\left(\frac{n^{\frac{1}{4}}X}{k}\right)$ (with high probability)

analysis is difficult

Our Algorithm

- set the number of buckets k_{ij} at node (i, j) flexibly
- random choice of running subtotal

⇒ error bound $O\left(X \sqrt{\sum_{i=1}^{n} \sum_{j=0}^{i} \left[\frac{\omega(i,j)}{k_{ij}} \right]^2} \right)$

- **adjust # of buckets k_{ij} to minimize error bound under the condition $\Sigma k_{ij} = kn^2$**

⇒ error bound $O\left(\frac{X}{k} \right)$

analysis is quite easy!

Open Problems

- derandomization of our algorithm with the same error bound
- approximation of American-Asian option
- analysis of error bound compared to exact price