Forward shooting grid method for arithmetic average options

Mircea Simionica

Applied Numerical Finance assignment

December 3, 2015

Overview

Average options

- Asian options provide a cost-efficient way of hedging
- More attractive to some investors because less expensive and less volatile
- Payoff:

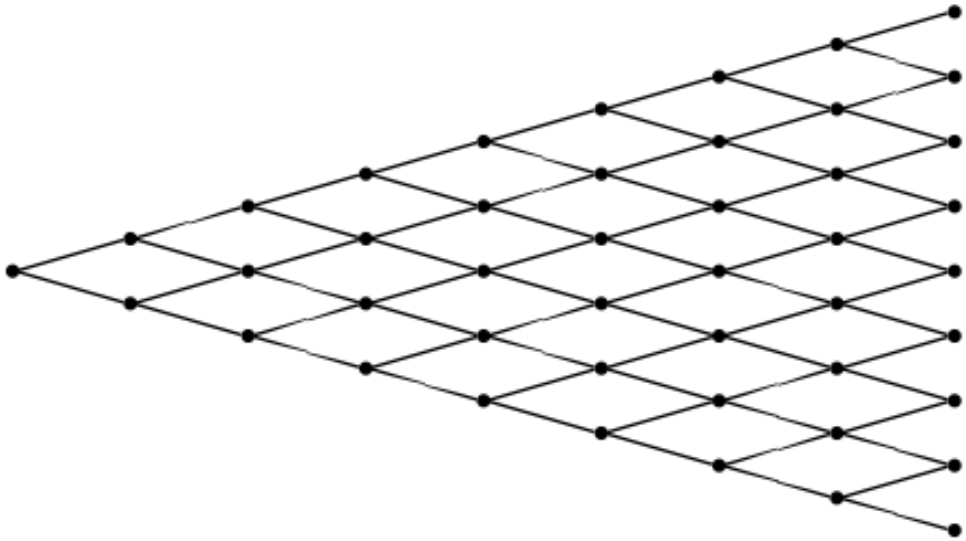
$$\text{Average call (fixed strike)} \rightarrow X(T) = (S_{average} - K)^+$$
$$\text{Asian call (floating strike)} \rightarrow X(T) = (S_T - S_{average})^+$$

- Standard form of averaging is arithmetic $\Rightarrow$ valuation is not trivial
  - Arithmetic average has no simple analytic shape
  - In classical binomial model the number of averages grows exponentially with the size of the tree

Forward shooting grid for arithmetic average options

Steps

1. Build tree for S

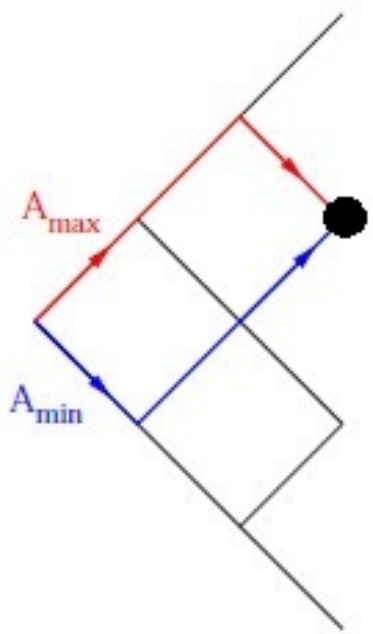


Forward shooting grid for arithmetic average options

Forward

Steps

1. Build tree for S
2. Shoot averages



Forward shooting grid for arithmetic average options

Backward ⏮

Steps

1. Build tree for S
2. Shoot averages
3. Backward recursion

Numerical and financial remarks

Numerical POV

- How to space values in the average vector
- Interpolation type
- Number of time steps and dimension of the average vector

Financial POV

- What about discrete sampling?
- Greeks

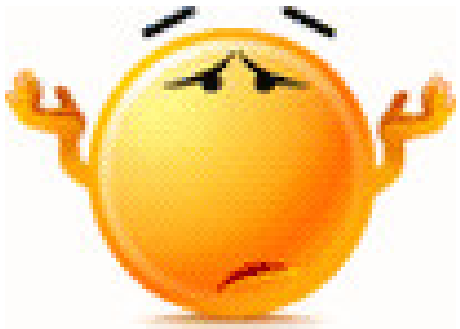
Programming remarks

Main issue

Data structure that will store the lattices.

Bad version

1. Lattice for S held in a matrix of doubles
2. Averages and option prices held in a field (matrix of vectors)



$\Rightarrow$ perfect recipe for memory bottlenecks

Programming remarks

Main issue

Data structure that will store the lattices.

A better way to do it

1. Lattice for S held in a sparse matrix
2. Averages held in a C++ vector< data type > (STL dynamic container)
3. Option prices just a vector of vectors (for every timestep we only need data from the step ahead)

Some numerical results

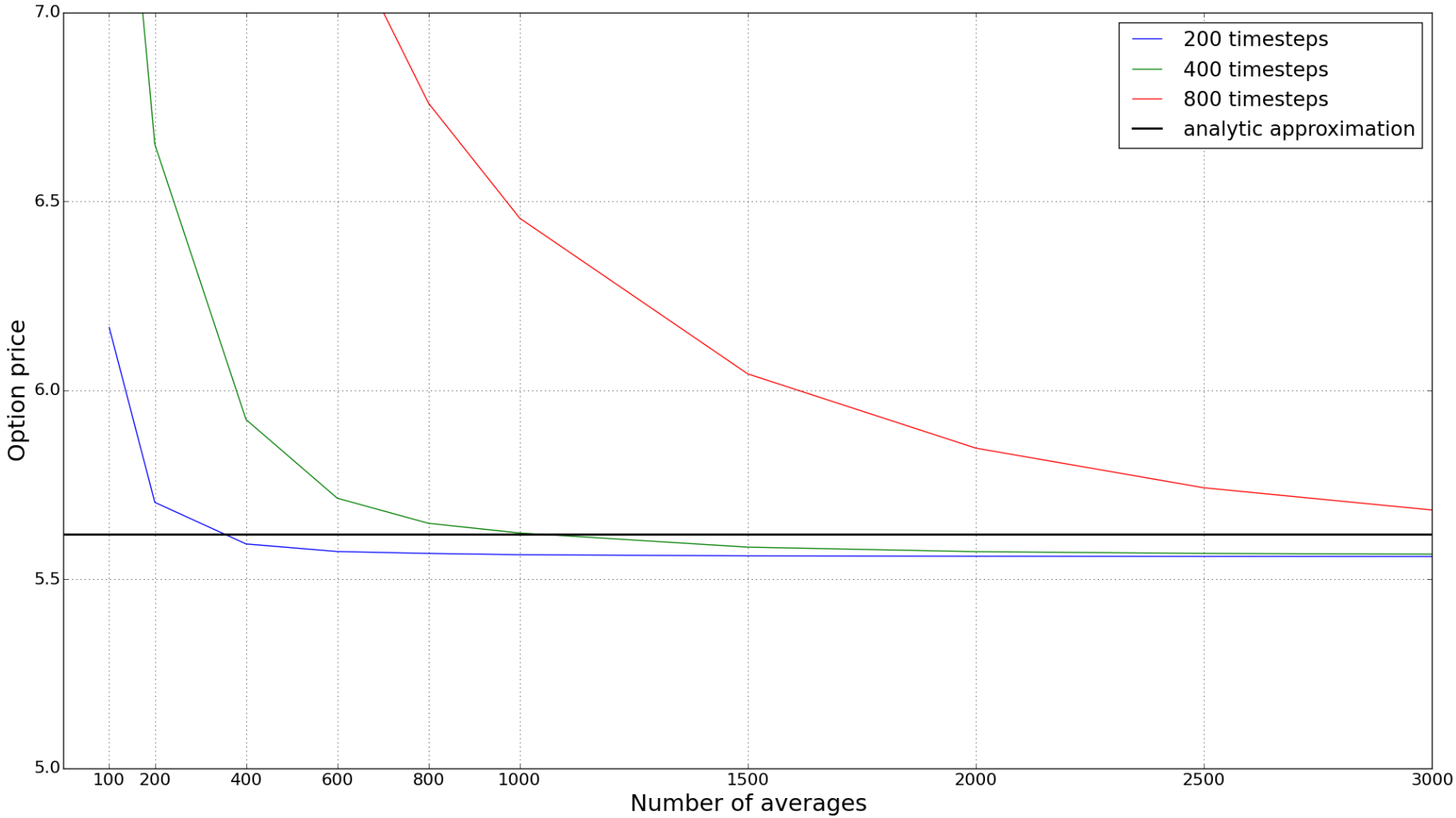
Arithmetic average option from Hull (page 613)

$S_0 = 50, K = 50, r = 0.10, \sigma = 0.40, T = 1$

Analyitic approximation with continuous averaging: 5.62

# time steps	# averages	Option price	Time (seconds)
200	100	6.16638	0.796763
200	200	5.70347	1.98051
200	400	5.59344	5.96032
200	600	5.57341	12.2608
200	800	5.56831	20.5464
200	1000	5.56491	31.1804
200	1500	5.56191	67.1026
200	2000	5.56086	116.902
200	2500	5.56038	179.459
200	3000	5.56011	259.534

Some convergence results



The end