FPGA-based Reconfigurable Computing for Pricing Multi-Asset Barrier Options

RAHUL SRIDHARAN, GEORGE COOKE, KENNETH HILL, HERMAN LAM, ALAN GEORGE, SAAHPC '12, PROCEEDINGS OF THE 2012 SYMPOSIUM ON APPLICATION ACCELERATORS IN HIGH PERFORMANCE COMPUTING,PAGES 34-43

Ishan Dalal

Barrier Option

A **barrier option** is a contract whose payoff automatically drops to zero when the value of the simulated asset return passes through some pre-defined barrier level, or remains at zero until the value passes through a barrier.

- History – Barrier Options were created to provide the insurance value of an option without charging as much premium.

  Example – If you believe that IBM share will go up this year, but are willing to bet that it won't go above $200, then you can buy the barrier and pay less premium than the vanilla option.

- **Multi-Asset Barrier Options** – They are path-dependent exotic options consisting of two or more underlying assets (like stocks, bonds etc).

Types of Barrier Options

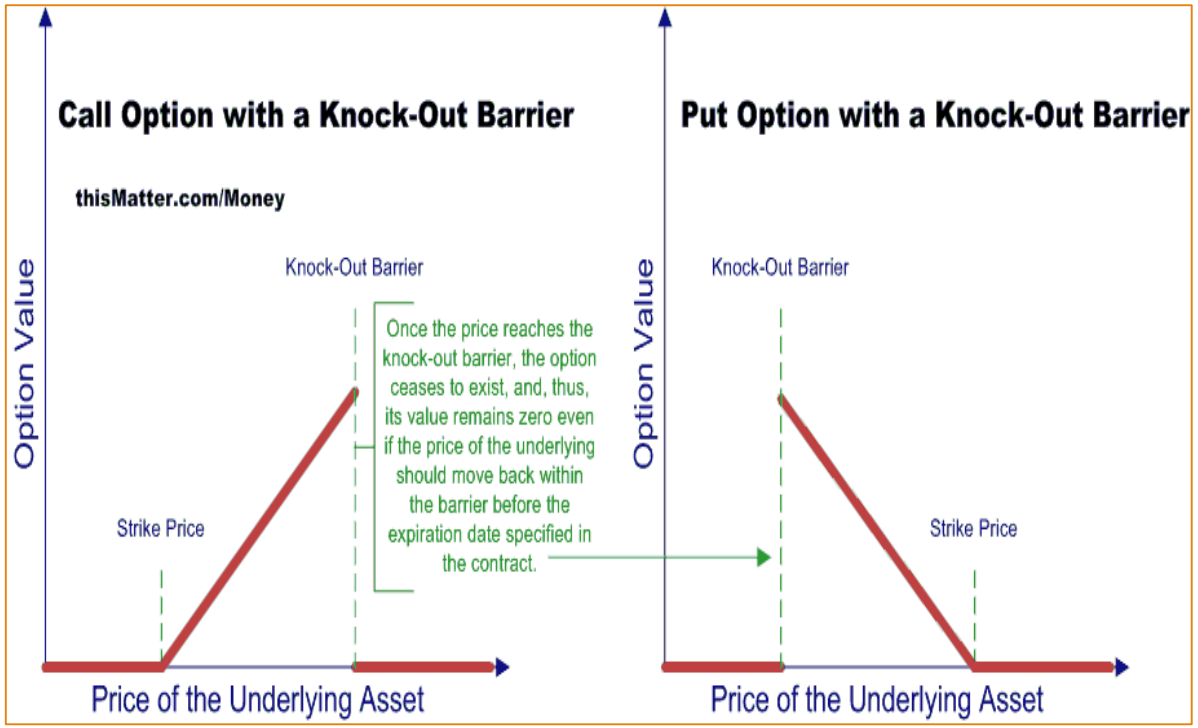
Up & Out : Spot Prices starts below the barrier level and has to move up for the option to be knocked out.

Down & Out : Spot Prices starts above the barrier level and has to move down for the option to become null and void.

Up & In : Spot prices starts below the barrier level and has to move up for the option to become activated.

Down & In : Spot prices starts above the barrier level and has to move down for the option to become activated.

Our Focus would be on Down & Out Barrier options in this paper.



Models for Options Pricing

- Options Pricing are used to predict future value of every asset that the bank trades.
- Main Goal is to determine the range over which an asset's value is expected to change i.e. measure of its volatility.

Two Main Models -:

1. Black - Scholes Model – Used to calculate the price of a simple 'vanilla' option treats volatility as a constant parameter.
   - Shortcoming – It exposes the trader to arbitrage by other traders as long he/she owns the option contract.
2. Heston Model – Calculate the price by considering volatility in the above model with an stochastic variable. Heston Model is used for Barrier pricing since barrier options are very sensitive to volatility. Slightest variation in underlying assets can cause the price of the option to tip over the barrier rendering the option to be worthless.

- Accelerating these complex pricing models enables banks to explore a vast array of underlying assets.

Barrier Options & Reconfigurable Computing (1/2)

- Multi-Asset Barrier Options are particularly attractive because they give the investor a single relatively cheap contract that captures the relationship among related sets of assets and offers some protection against the complicated volatility relationships that make the risk of the contract uncertain. Heston Model is the perfect choice for these complex covariance structures.
- High-Performance Reconfigurable Computing (RC) can be applied to such business relevant financial application using Monte Carlo (MC) methods.
- FPGA-based system architecture maps to the system of heston stochastic differential equations (SDEs) to price multi-asset barrier.
- It Consists of parallel set of MC cores, each capable of simulating multiple Monte Carlo paths. Each MC Core is designed to be customizable so that core for the model can be easily replaced.
- In the current design, Heston core based on full truncation Euler discretization method is used as model core. Different payoff calculator kernels can also be used to compute various payoffs such as vanilla portfolios, barriers, look-backs etc.

Barrier Options & Reconfigurable Computing (2/2)

- Advantages of using FPGA :

- FPGA implementation would greatly accelerate the valuation of multi-asset barrier options enabling banks to manage sophisticated contracts more flexibly and precisely.
- Computing with FPGA is much more scalable than the same with CPUs.
- Novel Financial Products can be a reality if limits to valuating the contract are relaxed.
- Investors would be able to purchase barrier options contracts of indices of their own devising or options contracts that simulate variations of popular indices or options contracts that simulate variations of popular indices or ETFs (Exchange – traded funds).
- Also applicable to calculate other exotic multi-asset option classes such as lookbacks, rainbows and Asian-style options.

Heston Model

- The evaluation of the underlying asset price St and the variance process Vt is described by the Stochastic Differential Equations :

$$dS(t) = rS(t)dt + \sqrt{V(t)}\, S(t)dW_s(t) \qquad (1)$$

$$dV(t) = -k(V(t) - \theta)dt + \omega\sqrt{V(t)}\, dW_v(t) \qquad (2)$$

Where r is the risk-free interest rate, k is the rate of mean-reversion of the variance, θ is the long-term variance and ω is the volatility of volatility. The terms dW_s and dW_v are independent Brownian motions which drive the two processes with a correlation co-efficient ρ between them.

- Different values for ρ affect the skewness of an asset's log-return distribution. ω affects the peak of distribution and higher the value means that volatility is more volatile. K represents the degree of volatility clustering.

Thus these parameters of Heston model affect the implied volatility and can produce different distributions which overcomes the shortcomings of Black-Scholes model leading to a more accurate pricing of options.

Discretization Scheme

- Barrier Options constructed using multiple underlying assets are best priced using Monte Carlo methods.
- MC simulations require efficient discretization of the continuous-time Heston SDEs. Euler-Maruyama scheme discretizes the continuous-time functions at finite time intervals simulating the stock and variance process at these discrete intervals.
- The scheme produces the least discretization bias of all Euler schemes and is given as :

$$S(t+\Delta) = S(t) + r(t)\Delta + \sqrt{V(t)^+}\sqrt{\Delta}\, Z_x \qquad (3)$$

$$V(t+\Delta) = V(t) + k(\theta - V(t)^+)\Delta + \omega\sqrt{V(t)^+}\, Z_v\sqrt{\Delta} \qquad (4)$$

Where x+= max(x, 0) and equations (3) & (4) represent the asset and volatility motions respectively.

Disadvantage :

- Bias introduced increases as the number of full truncation projections applied to the variance process increases.

Multi-Asset Heston Model

- Extension of the single-asset model to multiple dimensions requires the simulation of a system of correlated single-assets SDEs. For a d-dimensional system, we have

$$\frac{dS_i(t)}{S_i(t)} = \mu_i dt + \sigma_i dX_i(t), \quad i = 1,2,\ldots,d \qquad (5)$$

Assumption : Cross-correlation between the volatility processes is zero.

- The extension of Equations (3) & (4) to simulate a system consisting of multiple underlings leads us to:

$$S(t+\Delta) = S(t) + r(t) + \sqrt{V(t)^+}\sqrt{\Delta}\varepsilon^s \qquad (6)$$

$$V(t+\Delta) = V(t) + k(\theta - V(t)^+)\Delta + \omega\sqrt{V(t)^+}Z_x\sqrt{\Delta}\varepsilon^v \qquad (7)$$

Where $\varepsilon^s = \sum_{j=1}^{d} A_{ij} Z_{\Delta j}$, for i = 1,2,….,d. The matrix A is chosen to be cholesky square root of $\sum$ with $AA^T = \sum$.

Algorithm for Multi-Asset d-dimensional Heston Model

1) Generate a set of Gaussian random numbers $z_1, \ldots\ldots, z_d$ at each time step in the time grid.

2) Compute $\epsilon_i^s = Ax\ Z^T$ for each individual asset in the system

3) For the variance process compute

$$\varepsilon_i^v = \rho_i \varepsilon_i^s Z_v \sqrt{1 - \rho_i^2}\ , \text{where} Z_v \text{ is an independent gaussian random variable.}$$

4) Simulate the next time step for the volatility and asset process according to (6) & (7) respectively.

Multi – Asset Barrier Options Pricing

- A down and out Barrier is denoted by the payoff equation :

$1\{ T(b) > T \}\ (S_{(T)} - K)^+$ (8)

Where K is the strike price of the option and 1 {T(b) > T} is an indicator function equal to 1 if the barrier is not hit $\{S(t_i) < B\}$ before the option expires and is 0 otherwise.

- With multiple underlyings we modify the barrier equation (8) to calculate to price Worst-of-N call options where N is the number of underlyings.

$S_{(T)} = \min\{ S_i(T), S_2(T), \ldots\ldots S_N(T)\}$ (9)

Algorithm for Pricing Multi-asset barriers

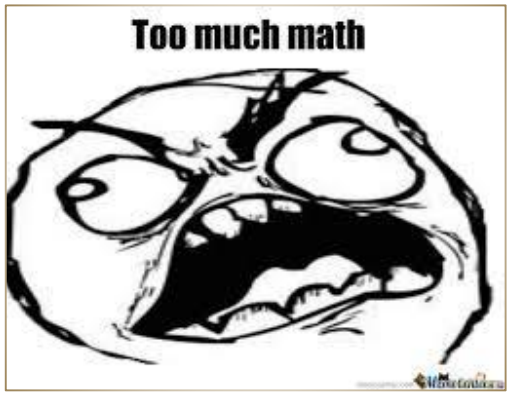
1) Calculate the next value of the options for each individual asset in a system based on equations (6) & (7).

2) Test it against the conditions of the barrier such that all assets in system satisfy the condition.

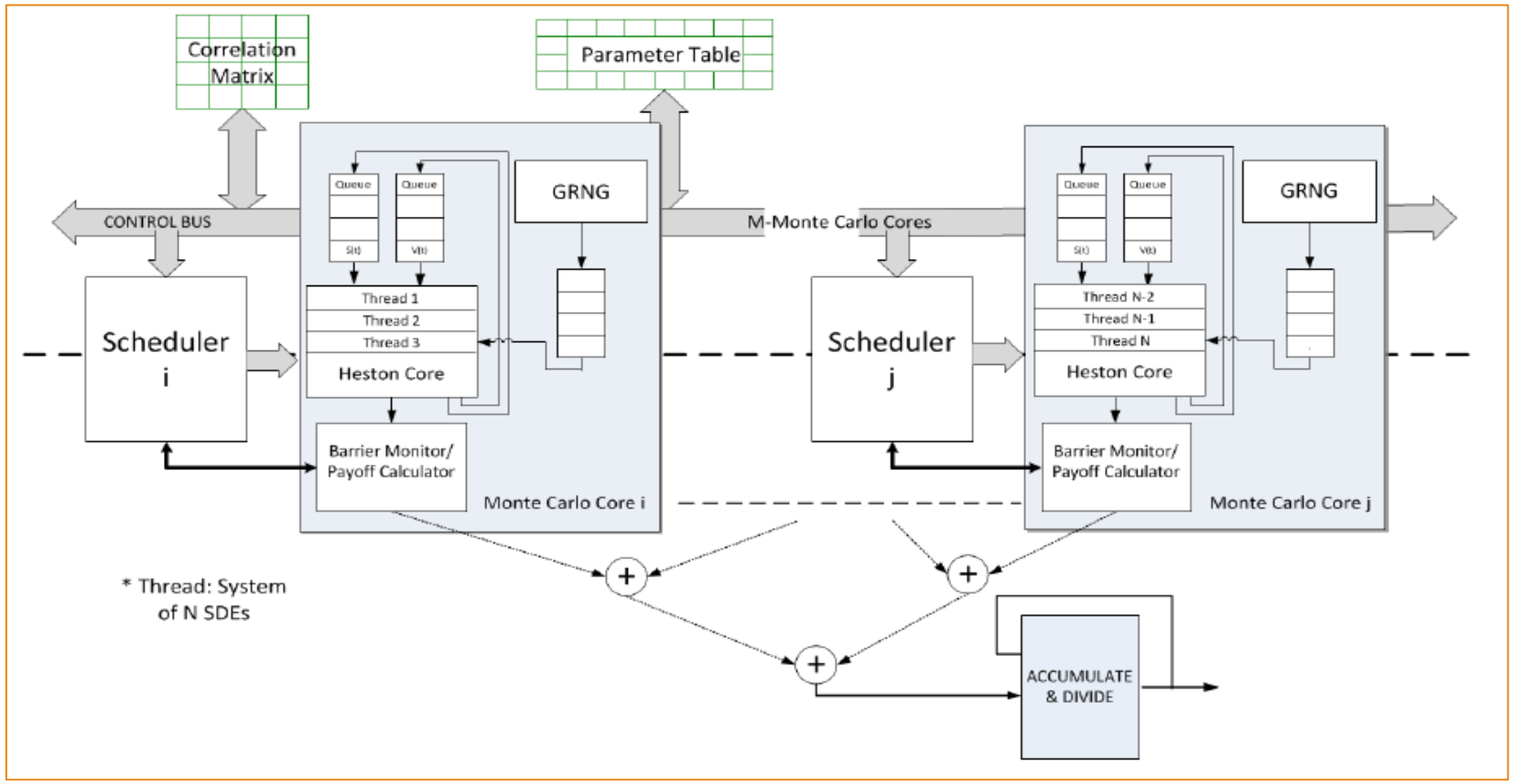
3) If the barrier is breached, value is zero for rest of path.

4) Repeat for multiple Monte Carlo Paths.

5) Average the paths to determine the value of the barrier using the payoff equations (8) & (9).



FPGA – Based RC System Architecture



Architecture Details

- The Design leverages an early termination condition of “out” barrier options to efficiently schedule MC paths across multiple cores in a single FPGA and across multiple FPGAs. Each MC core is associated with a scheduler, which monitors the simulation to perform this early termination and run-time scheduling of the MC threads.

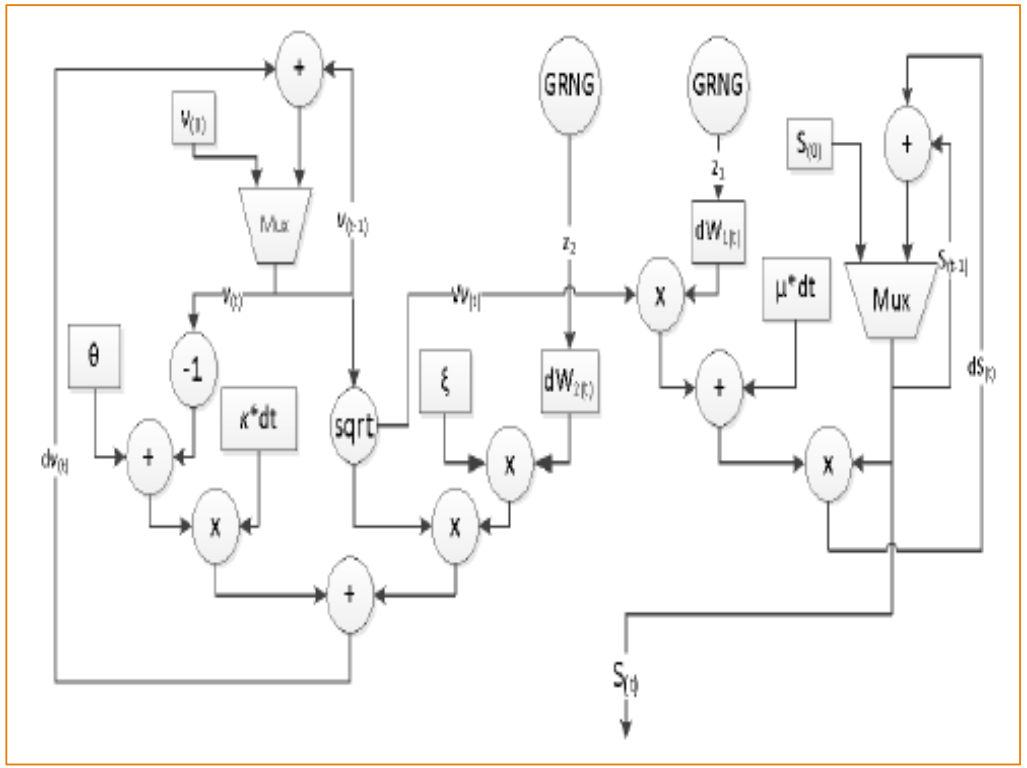
- A d-dimensional multi-asset barrier model involves the forward simulation of a system of SDEs. Parallelization of these forward motions simulations is achieved at two levels:

1. Multiple underlying assets belonging to a single system of SDEs are time-multiplexed within a Heston core. In the current architecture, such a system of SDEs is defined as thread.

2. Multiple systems of SDEs or threads are simulated independently across all the cores of a single FPGA and across multiple FPGAs.

Model (Heston) Core

- Heston Core has a pipelined architecture simulating the asset and volatility processes as scheduled by the scheduler at each stage of the pipeline.
- The scheduling process is abstracted away from the MC core making it modular enough to support future extensions or be replaced by other model cores.
- Figure shows the design and data flow of such a Heston core to perform the forward simulations of equations (6) & (7).



Inputs to MC core

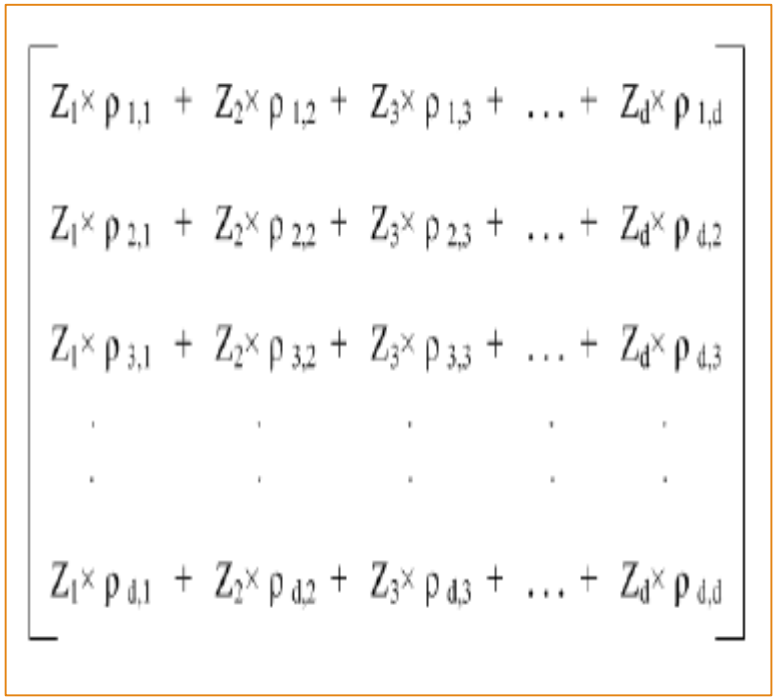
A MC core has two inputs :

1) Correlation Matrix :

Cross-correlations are inputted to the MC cores through the correlation matrix and the resulting vector ε^S is generated by computing the matrix product between a normally distributed random number vector and the correlation matrix.

2) Parameter Table :

Each row of the parameter table represents input parameters corresponding to each of the asset in a thread. The parameter table is implemented as distributed Block RAMs in the FPGA and is also controlled by the scheduler.



Correlation Matrix

Thread Scheduler

- Each MC core is associated with thread scheduler which performs early termination and run-time scheduling of the heston paths. A system of multiple underlying assets and their corresponding volatility processes are defined as threads. The evaluation of multiple-assets corresponds to the evolution of threads across multiple MC cores with each thread representing a single MC path.

- For a Heston Core with pipeline depth of M and a thread with N assets, we can simulate independently M/N threads per core.

- For an FPGA implementing K cores, one can thus simulate **K x (M/N)** threads or MC paths at any given time.

  - *Example – For a configuration with 4 underlying assets, we have K =36, M = 16 and N =4. So, 144 threads or MC paths can be simulated at a given time instant.*

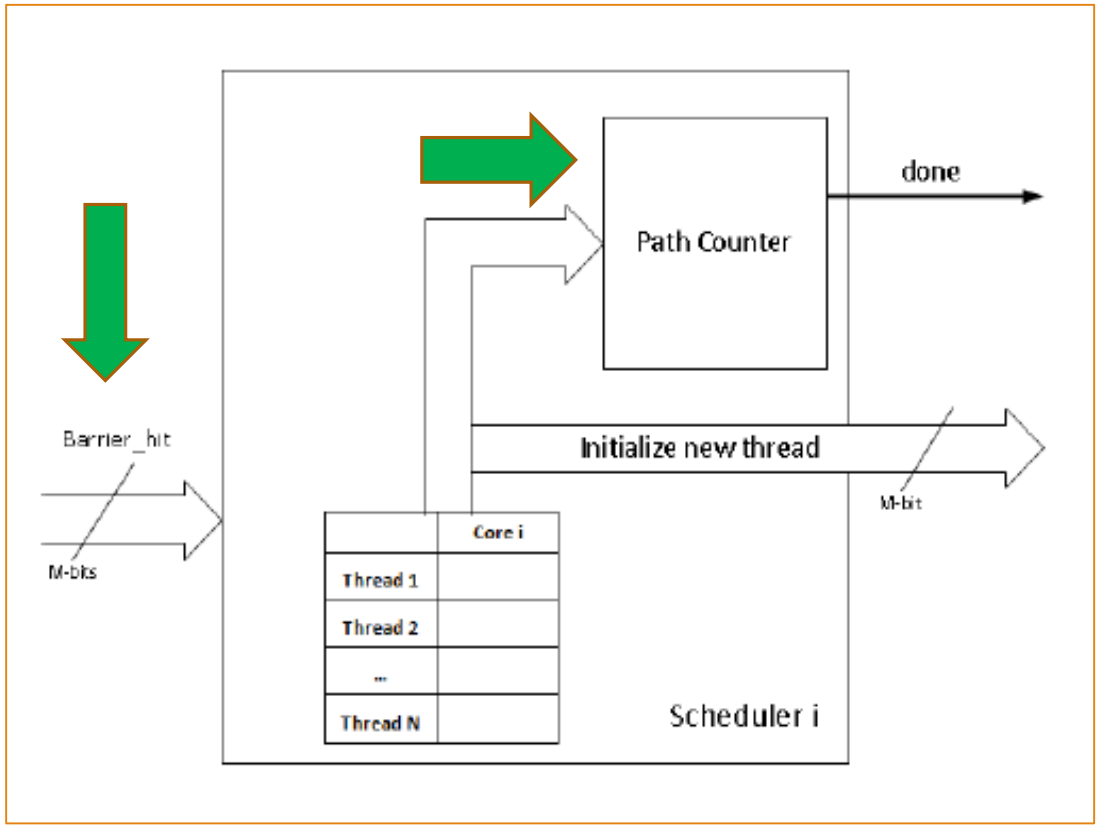
Early Termination

- Early termination ensures that a new thread begins its simulation immediately upon the termination of the current executing threads in a core.
- The scheduler initiates new thread in the pipeline under either of the following two conditions:

1) A thread reaches its maturity period T without breaching the barrier.
2) One of more assets constituting a thread breach the barrier at time instant t.

Data Structure of Thread Scheduler

- A Barrier Hit Signal indicates to the scheduler if the barrier has been breached by an asset in any of the scheduled threads.
- A path counter register is updated when a new thread is scheduled in a core. This register keeps the track of the total number of MC paths simulated and initiates transfer to the host once a desired number of paths have been simulated.



Discrete-time barrier monitoring and pricing

- The output from the heston core is the input to the barrier monitor/ payoff calculator which checks for the condition when the barrier is breached and calculates the resulting payoff.
- The barrier monitor component is gated, which enables it to be switched on or off as defined by an application.
- *Example – For a knock-out barrier, when switched on, the monitoring unit assets a corresponding barrier_hit signal in the event of a breach. A breach by any asset in the thread triggers the scheduler to initiate a new thread in the pipeline (early termination).*

Note : In-stream signaling is used to map an asset to its corresponding thread within the barrier monitor and payoff calculator.

Gaussian Random Number Generator (GRNG)

- Two inversion – based random number generators, each capable of generating one normally distributed random number per clock cycle, are incorporated into each MC core.
- The inverse CDF of a uniformly generated random is evaluated to produce a pseudo-random number with the required Gaussian distribution.
- Piecewise polynomial approximation and hierarchical segmentation using pre-computed lookup tables are used to compute the ICDF of the uniform random number in hardware.

Results

- Design is evaluated in two ways :

1) Validating the design and implementation of the Heston Model by comparing its output to approximated single-asset as well as multi-asset option prices calculated analytically.

2) Comparing the performance of FGPA implementation against an SSE2 optimized single-threaded C program.

Target Platform : NOVO - G (UF) – Stratix IV E530 FPGAs.

- Being embarrassingly parallel, design is evenly partitioned based on the number of paths executed in each FPGA. Outputs from different FPGAs are gathered in a host machine to calculate the value of the discounted payoff.

1) Design Validation & Verification

- To validate the Heston Model, its model parameters using a non-linear square fit of the parameters to observed market data are calibrated.
- The parameter table containing exactly similar parameters for all the underlying assets is generated. The correlation matrix used is an identity matrix which ensures that the simulation of each underlying asset acts as an independent MC path.
- Payoff function is configured to compute a vanilla call option.
- The Table contains 5 test cases which compares hardware output against the true value of the option (calculated analytically).

TABLE I. Single-asset Heston model verification

Spot Price	Strike Price	Initial Vol.	Time to Mature	True Value	Hardware Output
100	100	0.04	1 year	10.3009	10.2923
123.4	123.4	0.01994	1 year	13.85	13.8211
100	100	0.04	10 years	13.0847	13.0915
100	100	0.09	5 years	34.9998	34.9814
100	100	1	1 year	39.725	40.3231

Results Analysis

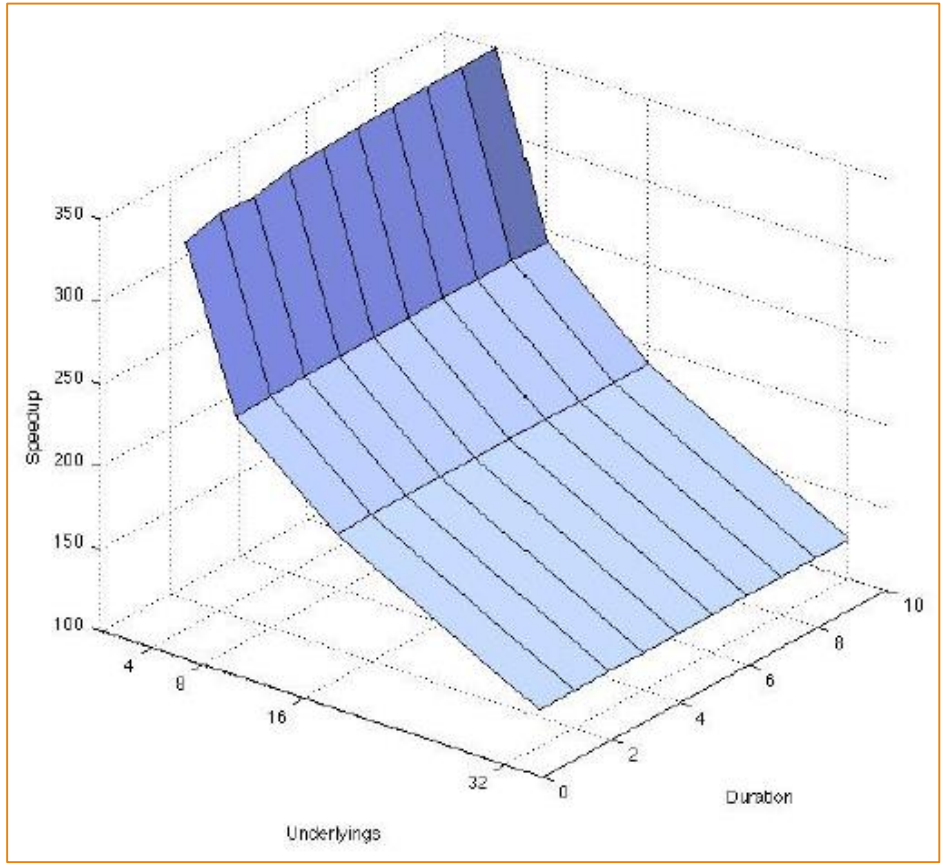
- A degree of bias can be expected from the hardware output resulting from the discretization error of the full truncation scheme.
- Despite the bias, the model is considered to be validated as long as this error is within limit of the implied volatility.
- Replacing the underlying discretization scheme with more sophisticated techniques such as Quadratic Exponential (QE) scheme, and improving the parameter calibration algorithm, will result in the Heston model pricing options with a higher degree of accuracy.

2) Performance Evaluation on Novo-G

- Baseline : Single-threaded version running on a single core as well as a multi-threaded version running on a server node consisting of two 8-core E5-2687 processors.
- A multi-asset worst-of-N barrier contract is run for 1,000,000 paths as the baseline.
- Box-Muller transform is used to generate normally distributed random numbers in software.
- Speedup evaluations are performed on a single Stratix IV FPGA and then scaled up to multiple FPGAs on Novo-G. FGPA – based design is compiled for a clock frequency of 125 Mhz.
- Four different design configurations (4, 8, 16 and 32 underlying assets) are considered while evaluating performance.
- An O(m x n) relationship exists between the number of underlying assets and the number of multiplications required to calculate these cross-correlations, where m is the number of MC cores and n is the number of underlying assets. This results in a tradeoff between the number of assets in a system and the total number of cores in an FPGA.

Speedup Comparison using a single FPGA

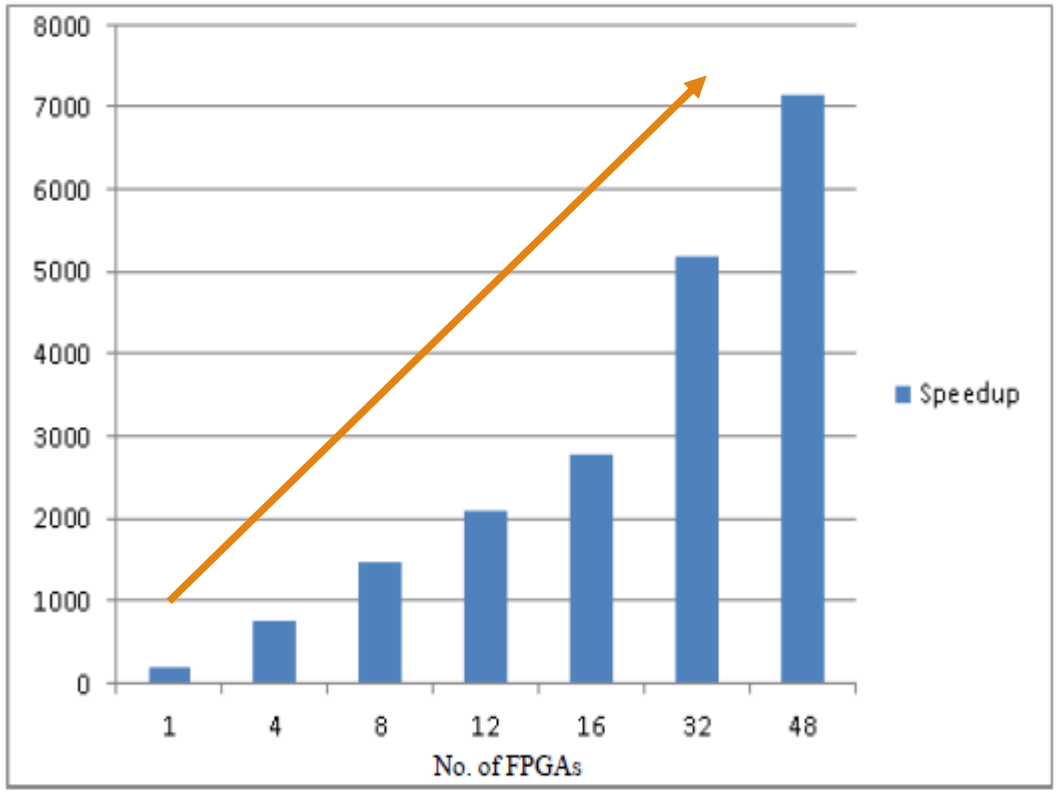
- For a duration = 10 years, the speedup reduces from 350 to 123 as we increase the number of underlying assets in a thread from 4 to 32.
- The design is limited by number of DSP multipliers in the FPGA which is determined by the number of underlying assets.
- The number of cores that can fit in an FPGA is reduced with an increase in the number of assets.
- For an average case of 16 assets per thread with a maturity period of 10 years, the observed speedup on single stratix IV FPGA is 189.



Performance using Multiple FPGAs

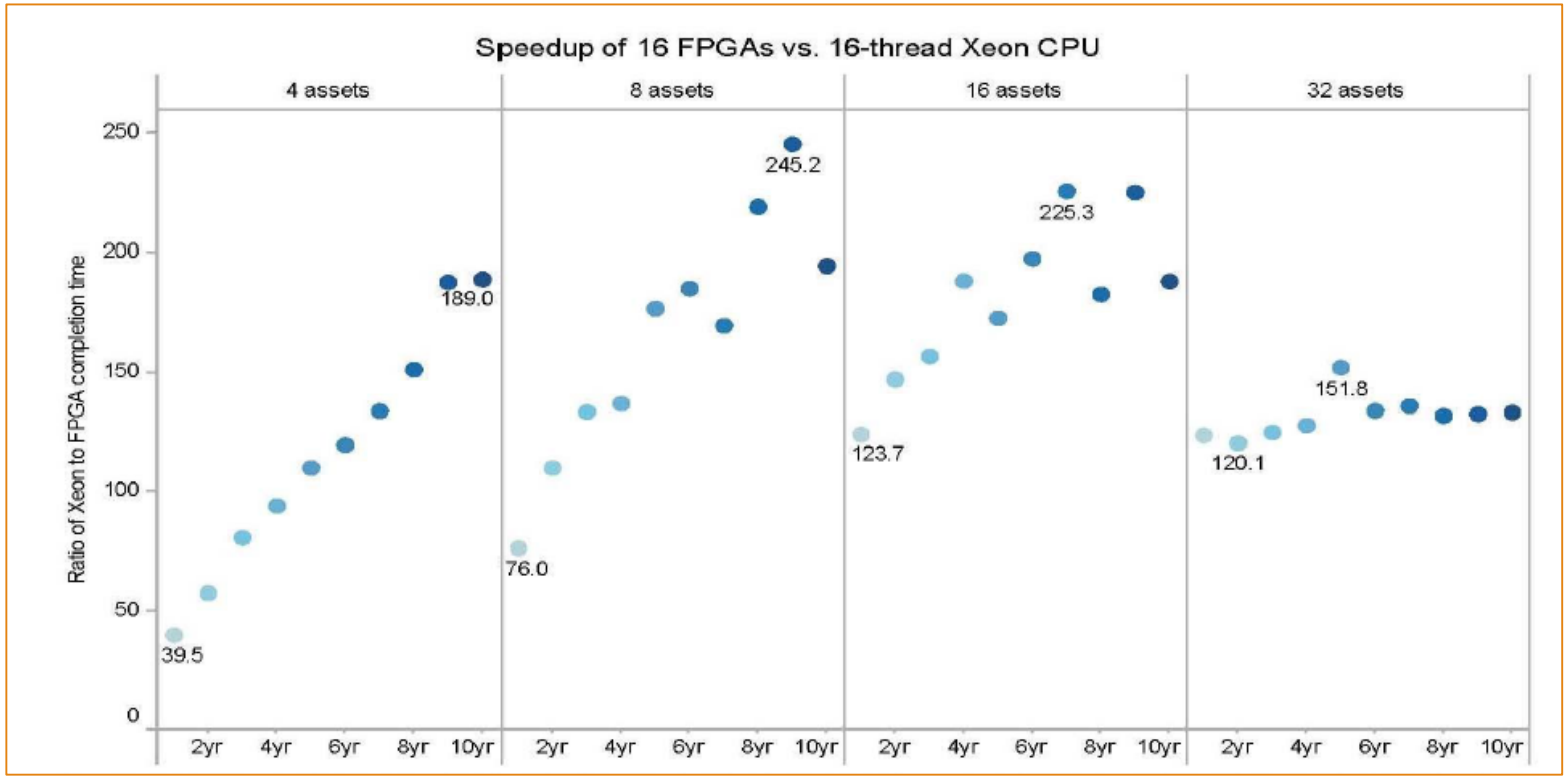
- Design is scaled across multiple FPGAs in a server and across multiple servers of Novo-G.
- The average case of 16 assets per thread with a maturity period of 10 years is used.
- It is seen that scaling the design across 48 FPGAs results in an overall speedup of 7134, as compared to software run time of 4415 seconds.

- Thus this design choice is embarrassingly parallel amenable to both partitioning and scaling across multiple FPGAs since each MC path simulation is an independent process within the model core.



Speedup achieved w.r.t. Multi-threaded Baseline

- Speedup achieved by a single server of Novo-G consisting of 16 FPGAs when compared to a multi-threaded baseline running a single server consisting of 16 E5-2687 cores is compared.



Analysis

- The speedup achieved reduced with increasing the number of underlying assets.
- Within each underlying asset, the observed speedup increases with the duration. This can be attributed to the smaller problem size for smaller values of duration.
- As the duration increases, the amount of time spent by the FPGA performing useful work increases to the communication overhead.
- The non-linearity associated with scaling reduces as the number of underlying assets increases.
- Increasing the FPGA computational density and reducing associated parallelization overheads would further improve the performance, allowing to compute higher dimension options in a flexible manner.

Shortfalls of the Study

- Euler discretization scheme produces large biases as number of full truncation projections increases. Quadratic Exponential (QE) scheme which has smaller biases could have been used and hardware output would have been much closer to True values.
- Another source of error in the design is the assumption that correlation between volatilities of different assets is zero. Advanced methods have been developed using to calculate these correlations.
- Performance comparisons in terms of energy utilizations on FPGAs and Intel E5 could have been made to know if there are any trade-offs between the architectures.
- Study could have been made all-inclusive by implementing the design on GPUs and performance comparisons could have been made in terms of speedups & energy consumption.

Conclusions

- Heston Model using the Euler discretization scheme gives pricing of the options within the accepted limits of implied volatility of the true (analytical) value. Thus being accurate, the design can be applied to produce market-consistent results.
- Modularity is obtained at the levels of options pricing model and discretization schemes. This can be leveraged to use different payoff calculator kernels to compute various payoffs such as vanilla portfolios, barriers, look-backs etc.
- Good Speedups obtained with single FPGAs. But due to an upper bound on logic resources on FPGAs, speedup is found to reduce as the number of underlying assets are increased.
- Design is scalable using a machine like Novo-G and large speedup of 7134 is obtained with 16 underlying assets with 48 FPGAs.
- Great accelerations are obtained in valuation of the options through this design which can enable the banks to manage sophisticated contracts more flexibly and precisely.

Questions ?

Thank You !