

1 Overview

1.1 Location `$<APPSDKSamplesInstallPath>\samples\bolt\`

1.2 How to Run See the *Getting Started* guide for how to build samples. You must first compile the sample. Use the command line to change to the directory where the executable is located. The pre-compiled sample executable is at `$<APPSDKSamplesInstallPath>\samples\bolt\bin\x86\` for 32-bit builds, and `$<APPSDKSamplesInstallPath>\samples\bolt\bin\x86_64\` for 64-bit builds.

Type the following command(s).

1. `BlackScholesBolt`
This runs the program with the default options; $s = (256 * 1024)$.
2. `BlackScholesBolt -h`
This prints the help file.
3. `BlackScholesBolt_TBB -h`
This command generates a build with the multiCoreCpu path (the Thread Building Block library), enabled.

1.3 Command Line Options Table 1 lists and briefly describes the command line options.

Table 1 Command Line Options

Short Form	Long Form	Description
-h	--help	Shows all command options and their respective meaning.
	--device	Explicit device selection for Bolt [auto/openCL/multiCoreCpu/SerialCpu]
-q	--quiet	Quiet mode. Suppresses most text output.
-e	--verify	Verify results against reference implementation.
-t	--timing	Print timing related statistics.
-v	--version	Bolt library and runtime version string.
-x	--samples	Number of sample input values to be calculated.
-i	--iterations	Number of iterations.

Note: The `--device multiCoreCpu` option becomes available when the sample is compiled with `ENABLE_TBB` defined. Microsoft Visual Studio build configurations `Debug_TBB` and `Release_TBB` are created for this purpose. These configurations have `ENABLE_TBB` defined to enable the TBB path (multiCoreCpu) for all the AMD BOLT functions used in the sample.

2 Introduction

The Option pricing is a very important problem encountered in financial engineering. This sample shows an implementation of the Black-Scholes model for European options.

The most common definition of an *option* is an agreement between two parties, the *option seller* and the *option buyer*, whereby the option buyer is granted a right (but not an obligation), secured by the option seller, to carry out some operation (or *exercise* the option) at some moment in the future (see reference 1). The pre-determined price is referred to as the *strike price*, and the future date is called the *expiration date*.

The two primary option types are:

- A *call option* grants its holder the right to *buy* the *underlying asset* at a *strike price* at some moment in the future.
- A *put option* gives its holder the right to *sell* the *underlying asset* at a *strike price* at some moment in the future.

There are several factors to consider in regard to options, mostly depending on when the option can be exercised.

European options can be exercised only on the expiration date. American-style options are more flexible: they can be exercised at any time up to, and including, the expiration date; as such, they are generally priced at least as high as corresponding European options. Other types of options are path-dependent or have multiple exercise dates (Asian, Bermudian). For a call option, the profit made at the exercise date is the difference between the price of the asset on that date and the strike price, minus the option price paid. For a put option, the profit made at the exercise date is the difference between the strike price and the price of the asset on that date, minus the option price paid. Thus, the price of the asset at expiration date and the strike price strongly influence how much is paid for an option.

Other important factors in the price of an option are:

- The time to the expiration date, T : Longer periods imply a wider range of possible values for the underlying asset on the expiration date. This means more uncertainty about the value of the option.
- The riskless rate of return, r , which is the annual interest rate of bonds or other “risk-free” investments: Any amount of dollars, P , is guaranteed to be worth $P \cdot e^{rT}$ dollars T years from now if placed today in one of these investments; in other words, if an asset is worth P dollars T years from now, it is worth $P \cdot e^{-rT}$ today.

3 Black-Scholes Model

The Black-Scholes model (see reference 2) provides a partial differential equation (PDE) for the evolution of an option price under certain assumptions. For European options, a closed-form solution exists for this PDE.

Equation 1 $V_{\text{call}} = S \cdot \text{PHI}(d_1) - X \cdot e^{-rT} \cdot \text{PHI}(d_2)$

Equation 2 $V_{\text{put}} = X \cdot e^{-rT} \cdot \text{PHI}(-d_2) - S \cdot \text{PHI}(-d_1)$

Equation 3
$$d_1 = \frac{\log\left(\frac{S}{X}\right) + \left(r + \frac{v^2}{2}\right)T}{v\sqrt{T}}$$

Equation 4
$$d_2 = \frac{\log\left(\frac{S}{X}\right) + \left(r - \frac{v^2}{2}\right)T}{v\sqrt{T}}$$

where:

V_{call} is the price for an option call.

V_{put} is the price for an option put.

$PHI(d)$ is the cumulative normal distribution function.

S is the current option price.

X is the strike price.

T is the time to expiration.

r is the continuously compounded risk free interest rate.

v is the implied volatility for the underlying stock.

The cumulative normal distribution function (see reference 3) is computed using the Abromowitz-Stegun approximation.

4 Implementation Details

This sample shows the usability of Bolt library's `transform()` as an alternative to `std::transform()` to calculate Black-Scholes' call and put prices. Each iteration of the *Functor* calculates the sample of the call and put price from a given sample of the stock price, strike price, time to expiration, volatility, and sigma. This sample shows the usage of `device_vectors` to allocate buffers on an available device. The input options that are populated on the host are then manually replicated onto the device (using `device_vector`) to gain a performance boost while running the sample using the Bolt library.

The advantage of defining a Functor as a `BOLT_FUNCTOR` is that this code is usable by both `std::transform()` and `bolt::cl::transform()` without modification to user code elsewhere.

5 Recommended Input Option Settings

For the best performance, enter the following on the command line: `-x 262144 -i 100 -q -t`

6 References

1. [http://en.wikipedia.org/wiki/Option_\(finance\)](http://en.wikipedia.org/wiki/Option_(finance)).
2. http://en.wikipedia.org/wiki/Black-Scholes#The_model.
3. Fischer Black and Myron Scholes (1973). "The Pricing of Options and Corporate Liabilities." *Journal of Political Economy* 81 (3): 637-654.

Contact

Advanced Micro Devices, Inc.
One AMD Place
P.O. Box 3453
Sunnyvale, CA, 94088-3453
Phone: +1.408.749.4000

For AMD Accelerated Parallel Processing:

URL: developer.amd.com/appsdk
Developing: developer.amd.com/
Forum: developer.amd.com/opencvforum



The contents of this document are provided in connection with Advanced Micro Devices, Inc. ("AMD") products. AMD makes no representations or warranties with respect to the accuracy or completeness of the contents of this publication and reserves the right to make changes to specifications and product descriptions at any time without notice. The information contained herein may be of a preliminary or advance nature and is subject to change without notice. No license, whether express, implied, arising by estoppel or otherwise, to any intellectual property rights is granted by this publication. Except as set forth in AMD's Standard Terms and Conditions of Sale, AMD assumes no liability whatsoever, and disclaims any express or implied warranty, relating to its products including, but not limited to, the implied warranty of merchantability, fitness for a particular purpose, or infringement of any intellectual property right.

AMD's products are not designed, intended, authorized or warranted for use as components in systems intended for surgical implant into the body, or in other applications intended to support or sustain life, or in any other application in which the failure of AMD's product could create a situation where personal injury, death, or severe property or environmental damage may occur. AMD reserves the right to discontinue or make changes to its products at any time without notice.

Copyright and Trademarks

© 2013 Advanced Micro Devices, Inc. All rights reserved. AMD, the AMD Arrow logo, ATI, the ATI logo, Radeon, FireStream, and combinations thereof are trademarks of Advanced Micro Devices, Inc. OpenCL and the OpenCL logo are trademarks of Apple Inc. used by permission by Khronos. Other names are for informational purposes only and may be trademarks of their respective owners.