DOI 10.37882/2223–2966.2020.08.35

ПРИМЕНЕНИЕ ГЕТЕРОГЕННЫХ ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ К ЗАДАЧАМ ФИНАНСОВОЙ МАТЕМАТИКИ

APPLICATION OF HETEROGENEOUS COMPUTING SYSTEMS TO PROBLEMS OF FINANCIAL MATHEMATICS

N. Shilina

Summary. Currently, the use of computing systems with heterogeneous architecture is becoming more popular. CUDA technology is applied to a huge number of problems, including in financial mathematics. The paper considers various methods of financial mathematics. The Monte Carlo method is used to build options pricing models. These algorithms are well suited for implementation on GPUs, since they are based on a large number of independent operations. The analysis of the efficiency of using GPUs in financial mathematics problems is carried out.

Keywords: Monte Carlo; options; CPU; GPU; CUDA.

Шилина Наталья Владимировна
ФГБОУ ВО «Санкт-Петербургский государственный университет»
n.v.shilina@gmail.com

Аннотация. В настоящее время применение вычислительных систем с гетерогенной архитектурой становится более востребованным. Технология CUDA применяется к огромному количеству задач, в том числе и в финансовой математике. В работе рассматриваются различные методы финансовой математики. Для построения моделей ценообразования опционов используется метод Монте-Карло. Эти алгоритмы хорошо подходят для реализации на графических процессорах, поскольку они основаны на большом количестве независимых операций. Проведен анализ эффективности использования графических процессоров в задачах финансовой математики.

Ключевые слова: Монте-Карло; опционы; CPU; GPU; CUDA.

Введение

В настоящее время, в связи с увеличением вычислений, важной проблемой стала эффективная реализация моделей, используя современные устройства. Как известно, в финансовом мире время играет очень важную роль, любая задержка в обработке информации может привести к огромным экономическим потерям. Поэтому для более эффективной работы используются технологии параллельных вычислений. Эффективным способом решения таких задач является ускорение на графических процессорах. Таким образом можно получить достаточно точные результаты с минимизацией времени.

Одним из важнейших и эффективных инструментов финансовой математики, предназначенных для страхования рисков, и, в свою очередь, ставшие объектом торговли, являются опционы. Так как сделки с опционами также рискованны, проблема определения справедливой цены опциона является весьма актуальной.

Существуют разные виды опционных контрактов, их главное отличие состоит в определении срока и цены исполнения. Чтобы сравнить эффективность применения графических процессоров в задачах финансовой математики, расчеты проводились для европейского и азиатского опционов. Европейский контракт исполняется строго в день экспирации, а цена исполнения азиатского опциона определяется средней стоимостью базового актива за определённый период времени.

Методы финансовой математики и алгоритмы вычислений

Существующие методы финансовой математики для оценки любых опционов можно разделить на две основные группы: аналитические и численные методы.

Аналитические методы — это оценка с использованием математически полученных формул, которые позволяют мгновенно получить значение опциона, однако, доступны эти методы для очень ограниченного набора опционных контрактов (европейские и некоторые экзотические варианты). Примером такой формулы является модель Блэка-Шоулза-Мертона.

Численные методы включают биномиальный метод, метод конечных разностей и метод Монте-Карло. Их преимущество в том, что они позволяют оценить стоимость любых опционов и деривативов. Однако точность этих методов, как правило, не идеальна. Также они требуют большего расчета и времени для получения результата.

В данной работе для программной реализации используется метод Монте-Карло. Суть этого метода состоит в оценке математического ожидания выплаты, которую опцион сгенерирует для своего владельца,

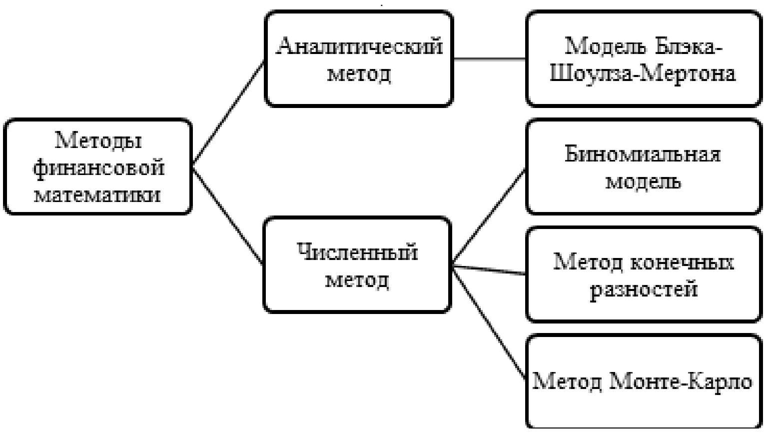

Рис. 1.Методы финансовой математики

многократно генерируя возможные ценовые пути движения акции. Алгоритмы определения цен с использованием метода Монте-Карло реализованы с помощью стохастического дифференциального уравнения.

Рассматривая опцион европейского типа, модель Блэка-Шоулза-Мертона соответствует следующему стохастическому дифференциальному уравнению:

$$dS(t) = rS(t)dt + \sigma dB(t), \qquad (1)$$

где $B(t)$ — стандартный винеровский процесс, $B(t)\sim N(0, 1)$;
S — цена базового актива в момент времени t;
r — годовая безрисковая процентная ставка.

В этом случае формула цены базового актива в день экспирации выглядит следующим образом:

$$S(T) = S_0 e^{\left(r-\frac{\sigma^2}{2}\right)T+\sigma\sqrt{T}\varepsilon}, \qquad (2)$$

где ε — случайная величина, распределенная по нормальному закону (распределение Гаусса), σ — среднеквадратическое отклонение доходности базовой акции.

Применение метода Монте-Карло для европейских опционов описывается следующим алгоритмом:

1. сгенерировать n случайных чисел для ε;
2. используя формулу (2), вычислить цену базового актива $S(T)$ для каждого значения ε;
3. вычислить среднее значение цены базового актива S^{cp};
4. по формуле (3) вычислить цену опциона C:

$$C = e^{-rT} S_{cp} \qquad (3)$$

Используя метод Монте-Карло для азиатского опциона, необходимо сгенерировать случайный временной ряд для цены базового актива за весь период времени. Необходимо разделить этот период времени на m частей. Тогда шаг

$$\Delta t = \frac{T}{m}.$$

В каждый момент времени цена будет зависеть от цены на предыдущем шаге. Формула цены базового актива выглядит следующим образом:

$$S(t_{i+1}) = S(t_i) e^{\left(r-\frac{\sigma^2}{2}\right)\Delta t+\sigma\sqrt{\Delta t}\varepsilon} \qquad (4)$$

где Δt — промежуток времени между двумя измерениями цены.

Алгоритм для вычисления цены азиатского опциона с применением метода Монте-Карло выглядит следующим образом:

1. сгенерировать m случайных чисел ε и найти для каждого из них значение цены базового актива по формуле (4), $(i=0,\ 1,\ \ldots,\ m)$;
2. по следующей формуле находим среднее арифметическое значение полученного ряда:

$$S = \frac{1}{m}\sum_{i=1}^{m} S(t_i) \qquad (5)$$

3. повторить п. 1 и п. 2 n раз;
4. вычислить цену опциона C по формуле:

$$C(s_0, T) = \frac{1}{n} e^{-rT} \sum_{j=1}^{n} S_j \qquad (6)$$

Вычислительные системы и программная реализация

Вычислительные системы — это совокупность аппаратных и программных средств, используемые для решения различных задач. Отличительная особенность вычислительных систем по отношению к электронно-вычислительным машинам заключается в наличие в них нескольких вычислителей, реализующих параллельную обработку. Основной проблемой в области суперкомпьютерных технологий является повышение производительности системы.

На основе принципов конвейерной обработки и суперскалярности были созданы микропроцессоры (CPU) и, позднее, графические ускорители (GPU). Использование графического процессора подходит не только для ускорения трёхмерной графики, но и для решения задач с высокой степенью параллелизма.

Современные графические процессора имеют высокую скорость доступа к модулям памяти, обработка больших объёмов данных может происходить параллельно, а производительность достигает высоких значений.

Первоначально CPU и GPU создавались для определенного класса задач, а системы были однородными (состоящими из одного или нескольких компьютеров с одинаковой архитектурой). В настоящее время часто используют гетерогенные системы. Гетерогенные системы могут использовать универсальный процессор CPU и графический GPU совместно.

Стандартной гетерогенной системой является совокупность одного CPU и одного или более GPU. Однако GPU используется как сопроцессор к центральному процессору, который является «хостом», и называется «устройством».

GPGPU (General Purpose computing for GPU) — это техника использования GPU для расчетов, которые обычно выполняются на CPU.

Существуют различные платформы для GPGPU, такие как OpenCL, AMD FireStream, Nvidia CUDA и другие.

В данной работе используется платформа Nvidia CUDA (Compute Unified Device Architecture). CUDA использует CPU и GPU: на CPU выполняется последовательная часть кода, на GPU — параллельные участки кода, выполняемые одновременно несколькими потоками.

Для запуска параллельной реализации алгоритма использованы GPGPU NVidia Tesla M2050.

Для генерации случайных чисел с равномерным и нормальным распределением на GPU используется библиотека cuRAND.

Для симуляции Монте-Карло используется квазислучайный генератор. Генератор квазислучайных чисел уменьшает вероятность повторного появления случайного числа, поэтому этот генератор покрывает пространство равномернее.

Для того чтобы использование GPU было эффективным, необходимо использовать некоторые оптимизированные под него параллельные алгоритмы. В работе используются два алгоритма обработки массивов — параллельной редукции и префиксных сумм.

Численные эксперименты

Все вычисления произведены для опциона «колл». Опцион «колл» дает право купить актив в будущем по заранее установленной цене.

Начальные данные представлены ниже:

- цена базового актива $S_0 = 100$;
- цена исполнения опциона K = 110;
- время до истечения срока действия опциона T = 1 год;
- среднеквадр. отклонение доходности базовой акции σ = 0,3;
- безрисковая процентная ставка r = 0,1.

Подставив эти данные в формулу Блэка-Шоулза-Мертона,

$$C(S,t) = SN(d_1) - Ke^{-r(T-t)}N(d_2)$$

$$d_1 = \frac{\ln(\left(\frac{S}{K}\right) + \left(r + \frac{\sigma^2}{2}\right)(T-t)}{\sigma\sqrt{T-t}}$$

$$d_2 = d_1 - \sigma\sqrt{T-t}$$

получим следующие значения:

$$d_1 = \frac{\ln\left(\frac{100}{110}\right) + (0.1 + \frac{0.3^2}{2})}{0.3} = 0.1656 \approx 0.17$$

$$d_2 = \frac{\ln\left(\frac{100}{110}\right) + (0.1 - \frac{0.3^2}{2})}{0.3} = -0.1344 \approx -0.13$$

$$N(d_1) = N(0.17) = 0.5675$$

$$N(d_2) = N(-0.13) = 1 - N(0.13) = 1 - 0.5517 = 0.4483$$

$$C(S_0, 0) = 100 \times 0.5675 - 110 \times e^{-0.1} \times 0.4483 = 56.75 - 44.62 = 12.13$$

Таким образом, цена европейского опциона с такими исходными данными равна 12.13

Изначально было сгенерировано малое количество случайных чисел. Из-за этого сходимость к значению, по-

Рис. 2. Оценка европейского опциона методом Монте-Карло (вар. 1)

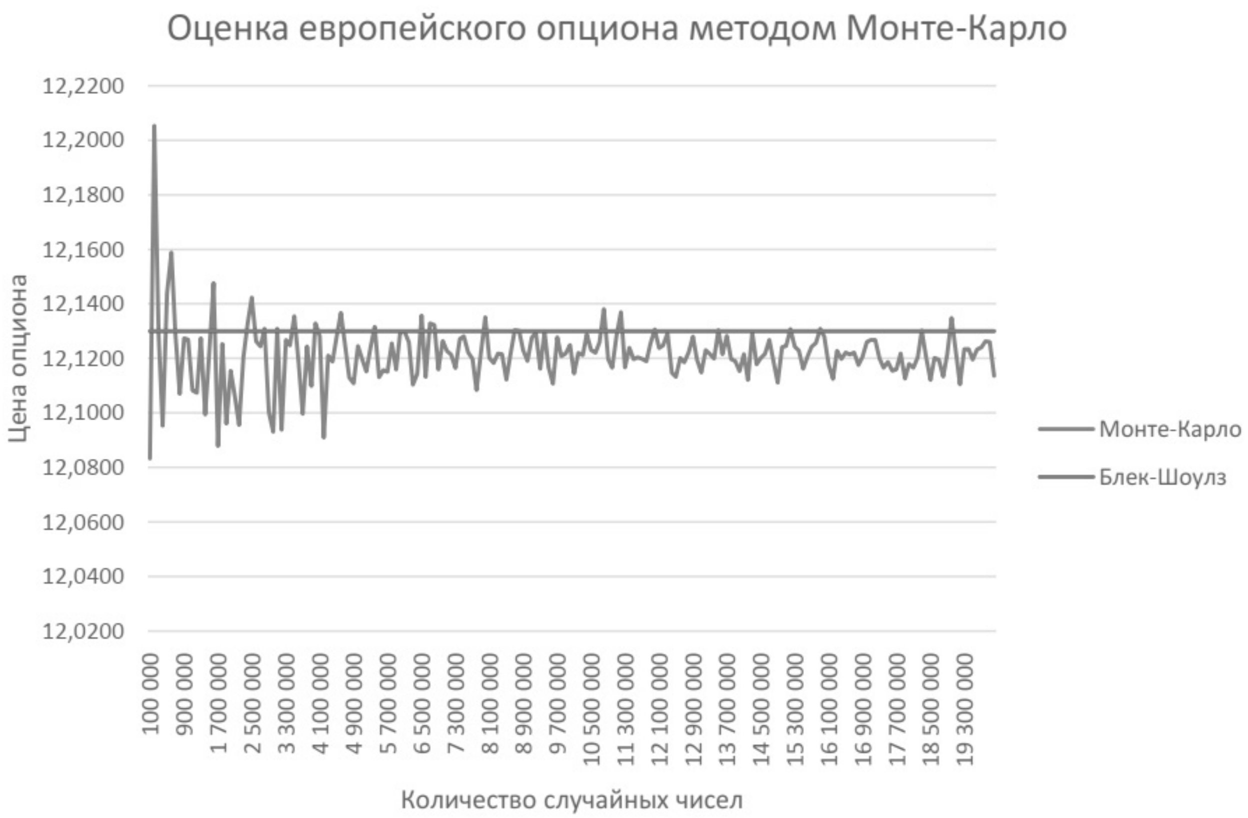

Рис. 3. Оценка европейского опциона методом Монте-Карло (вар. 2)

Таблица 1. Сравнительный анализ скорости вычислений на CPU и GPU, европейский опцион

Кол-во случайных чисел	Время на CPU, мс	Время на GPU, мс
361 000	59	13
2 762 600	397	14
6 892 600	991	19
10 893 600	1602	22
15 253 600	2276	27
20 296 000	3122	30

Рис. 4. Ускорение вычислений на GPU

лученному по формуле Блэка-Шоулза-Мертона, недостаточная. На рис. 2 изображен график, на котором красной линией обозначено это значение.

Если количество случайных величин увеличить, то сходимость будет намного больше. На рис. 3 изображен такой график. Можно заметить, что при количестве случайных величин, равных 20 млн., сходимость достаточно высокая.

Также важной задачей является сравнить скорость вычислений на CPU и GPU. Сравнительный анализ представлен в табл. 1:

Таким образом, при 361 тыс. случайных чисел, ускорение равно 4.5, а при 20.2 млн.— 104.

Следовательно, использование GPU для вычисления цены европейского опциона является эффективным.

Стоит заметить, что с увеличением количества случайных чисел ускорение на GPU возрастает. Это можно заметить на графике рис. 4. CPU исполняет поток инструкций последовательно, несмотря на максимальную производительность. В то время как GPU исполняет большое количество потоков параллельно.

Таблица 2. Сравнительный анализ скорости вычислений на CPU и GPU, азиатский опцион

Кол-во случайных чисел	Время на CPU, мс	Время на GPU, мс
361 000	57	139
2 762 600	368	194
6 892 600	919	493
10 893 600	1472	785
15 253 600	2165	1143
20 296 000	2984	1530

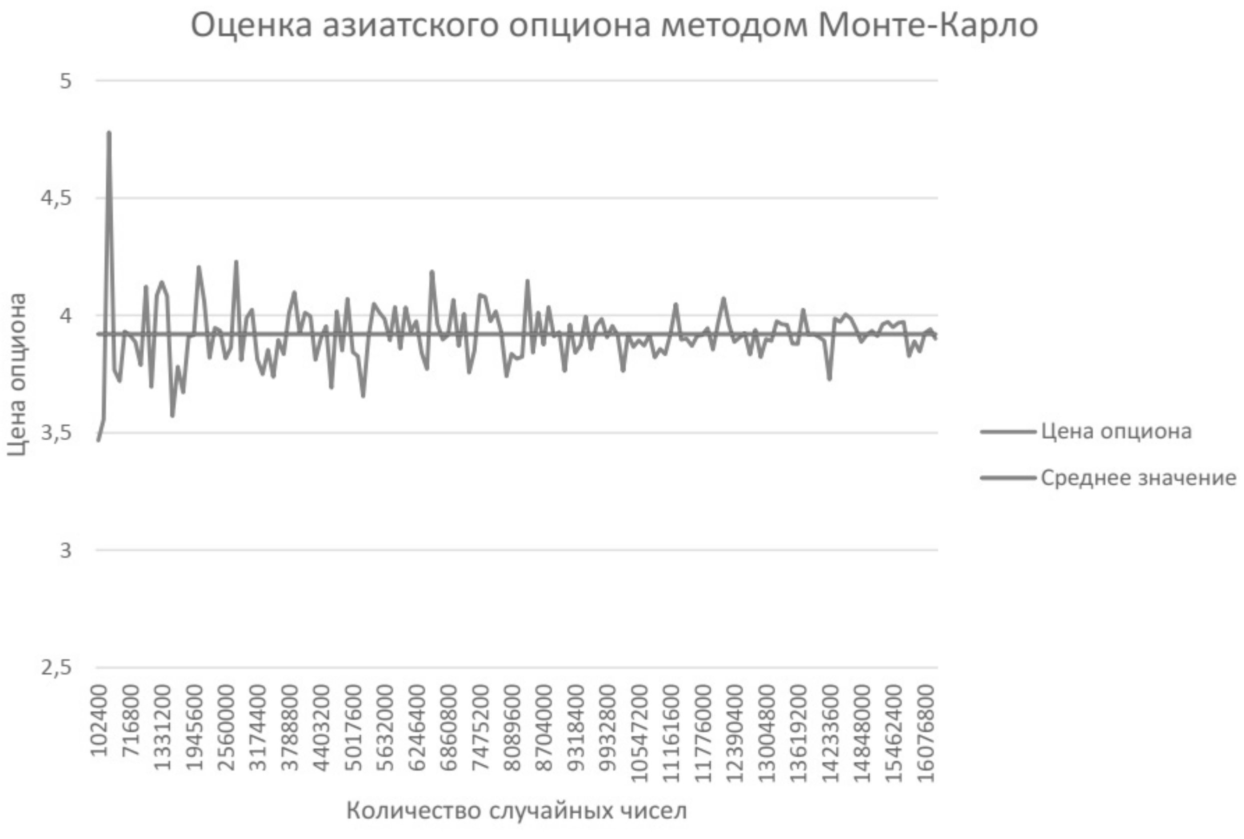

Рис. 5. Оценка азиатского опциона методом Монте-Карло

Для азиатского опциона вычисления сложнее, необходимо считать цену базового актива для каждого отрезка времени.

В расчетах используются те же начальные данные. Среднее значение такого опциона равно 3.95. На графике рис. 5 представлена сходимость к этому значению.

Сравнительный анализ скорости вычислений на CPU и GPU представлен в табл. 2:

Среднее ускорение равно 2. Следует отметить, что по сравнению с европейским вариантом ускорение значительно ниже. Причины этого — характеристики азиатских опционов. Как упоминалось ранее, на каждой итерации необходимо рассчитать промежуточное зна-

Рис. 6. Скорость вычисления азиатского опциона на CPU и GPU

чение цены базового актива и использовать алгоритм префиксных сумм.

График на рис. 6 показывает, как меняется скорость расчета цены азиатского опциона в зависимости от использования разных процессоров.

Следовательно, использование GPU для вычисления цены азиатского опциона с помощью стохастического дифференциального уравнения является неэффективным.

Выводы

В данной работе были проведены численные эксперименты с использованием технологии CUDA. Показаны методы Монте-Карло для расчета цен европейских и азиатских опционов с использованием стохастических дифференциальных уравнений. Проведен анализ эффективности использования графических процессоров в задачах финансовой математики.

Были достигнуты следующие результаты программной реализации методов Монте-Карло на модуле обработки NVidia Tesla M2050 GPGPU:

- ускорение в 104 раза для расчетов цен европейских опционов с использованием стохастического дифференциального уравнения;
- ускорение в 2 раза для расчетов цен азиатских опционов с использованием стохастического дифференциального уравнения.

В результате вычислительных экспериментов по расчету цены азиатского опциона с использованием стохастического дифференциального уравнения использование графического процессора оказалось неэффективным.

ЛИТЕРАТУРА

1. Экономико-математическое моделирование финансового рынка / В. К. Бурлачков, А. В. Гусаков // Финансовый менеджмент.— 2008.— N.5.— С. 135–143.
2. Вычисления на GPU: мифы и реальность. [Электронный ресурс]. Режим доступа: https://compress.ru/article.aspx?id=23724
3. Джон К. Халл. Опционы, фьючерсы и другие производные финансовые инструменты. Изд. дом. Вильямс, 2014. 1072 с.
4. Bluemke A. How to Invest in Structured Products: A Guide for nvestors and Asset Managers. Wiley Finance, 2009.

5. Глухов М. Оценка опционов методом Монте-Карло // Futures&Options. 2009. № 4
6. F. Black, M. Scholes. The pricing of options and corporate liabilities // Journal Political Economy, 1973. Vol. 81. No. 3. P. 637–659
7. Особенности и перспективы применения модели Блэка-Шоулза для российского рынка / А. А. Масалова, А. А. Гладилин // Современная экономика.— 2019.— с. 150–153.
8. Cox J.C., Ross S. A., Rubinstein M. Option pricing: a simplified approach//Journal of Financial Economics, September, 1979. 7. P. 229.
9. Ворошилова Наталья Александровна Сравнительный анализ методов моделирования стоимости опционов // Научный журнал КубГАУ — Scientific Journal of KubSAU. 2007. № 26.
10. Boyle Ph. Options: a Monte Carlo approach //Journal of FinancialEconomics. 1977. 4. P. 323–338.
11. Соболь И.М., Численные методы Монте-Карло. М.: Наука, 1973. 307 с.
12. Ермаков С. М. Метод Монте-Карло в вычислительной математике: Вводный курс. СПб.: Невский Диалект; М.: БИНОМ. Лаборатория знаний, 2009. 192с.
13. Hongbin Zhang. Pricing Asian Options using Monte Carlo Methods. Department of Mathematics Uppsala University, 2009. 36 с.
14. Михаил Глухов. Оценка экзотических опционов методом МонтеКарло. Futures&Options, май 2009. 40–49 с
15. Hyesoon Kim, Richard Vuduc, Sara Baghsorkhi. Performance Analysis and Tuning for General Purpose Graphics Processing Units (GPGPU) // Morgan & Claypool Publishers, 2012
16. А.В. Боресков, А. А. Харламов. Основы работы с технологией CUDA. Изд. дом. ДМК Пресс, 2010, 232 стр.
17. А.В. Боресков и др. Предисл.: В. А. Садовничий. Параллельные вычисления на GPU. Архитектура и программная модель CUDA: Учебное пособие. Изд-во Московского университета, 2012, 336 стр.
18. Официальный сайт Nvidia http://www.nvidia.ru/object/gpucomputingapplications-ru.html

Журнал «Современная наука: актуальные проблемы теории и практики»

Санкт-Петербургский Государственный Университет