

Анализ производительности алгоритмов умножения матриц

Левин Леонид

21 мая 2024 г.

Введение

Умножение матриц является одной из ключевых операций в линейной алгебре и часто используется в различных областях науки и техники, включая компьютерное зрение, машинное обучение и обработку сигналов. В данной работе рассматриваются три метода умножения матриц: *обычное умножение матриц*, умножение с использованием *SIMD (Single Instruction Multiple Data)*, и *алгоритм Винограда*. Цель исследования — проанализировать производительность этих трех алгоритмов при различных уровнях оптимизации компилятора.

Методы

В данной работе реализованы и протестированы три метода умножения матриц:

- **Обычное умножение матриц** — используется тройной цикл для выполнения операции умножения.
- **Умножение с использованием SIMD** — применение инструкций ARM NEON для параллельной обработки данных.
- **Алгоритм Винограда** — оптимизированный алгоритм умножения матриц, который уменьшает количество необходимых умножений.

Анализ производительности проводился с использованием следующих уровней оптимизации компилятора g++:

- **-O0** — без оптимизации.
- **-O1** — базовая оптимизация.
- **-O2** — умеренная оптимизация.
- **-O3** — агрессивная оптимизация.

Результаты

Ниже приведены графики зависимости времени выполнения от размера квадратной матрицы для каждого уровня оптимизации.

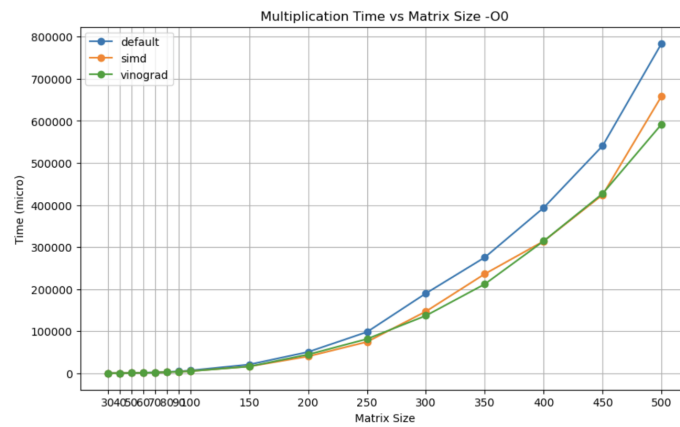


Рис. 1: Производительность при оптимизации -O0

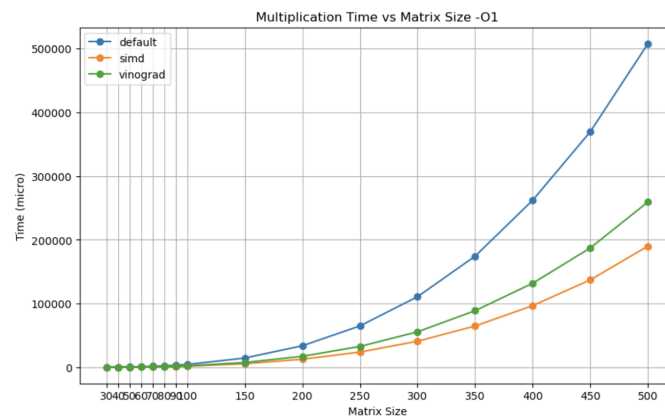


Рис. 2: Производительность при оптимизации -O1

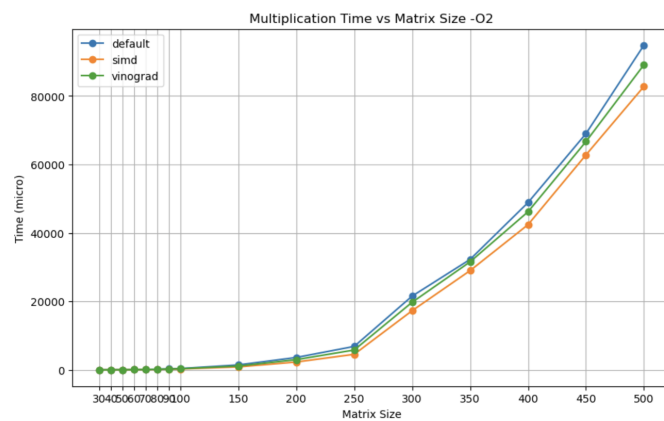


Рис. 3: Производительность при оптимизации -O2

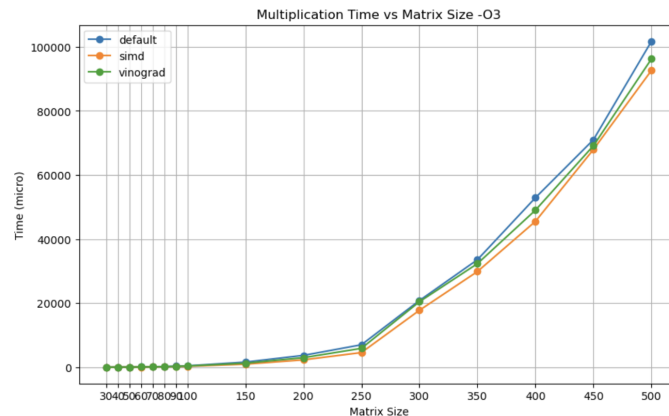


Рис. 4: Производительность при оптимизации -O3

Анализ графиков

Проанализировав графики, можно сделать следующие выводы:

- При уровне оптимизации -O0 (рис. 1) наблюдается значительная разница во времени выполнения между тремя методами. Обычное умножение матриц показывает худшие результаты, тогда как SIMD и алгоритм Винограда более эффективны.
- При уровне оптимизации -O1 (рис. 2) время выполнения уменьшается для всех методов. Однако, методы SIMD и Винограда по-прежнему показывают лучшую производительность по сравнению с обычным умножением.
- При уровнях оптимизации -O2 и -O3 (рис. 3, 4) наблюдается сходное снижение времени выполнения для всех методов, причем -O2 немного лучше. Алгоритмы SIMD и Виноград продолжают демонстрировать значительное преимущество в производительности.

Выводы

Результаты исследования показывают, что использование SIMD и алгоритма Винограда значительно повышает производительность умножения матриц по сравнению с обычным методом. При этом применение различных уровней оптимизации компилятора приводит к сокращению времени выполнения для всех рассматриваемых методов, особенно при использовании высоких уровней оптимизации (-O2 и -O3).