

Тестирование программ на QR разложение матрицами Хаусхолдера

1 Программа на C

Вычисление времени выполнения и ошибки $\frac{\|QR-A\|_F}{\|A\|_F}$ для случайно-сгенерированной матрицы A :

Размерность N	Время выполнения(сек)	Точность
256	0.151295	$0.206673 * 10^{-14}$
512	1.362996	$0.325643 * 10^{-14}$
1024	13.198033	$0.332653 * 10^{-14}$
2048	108.055010	$0.442351 * 10^{-14}$

Если скомпилировать программу на gcc с флагом оптимизации -O3, то получим такой результат:

Размерность N	Время выполнения(сек)	Точность
256	0.034629	$0.159191 * 10^{-14}$
512	0.288620	$0.357033 * 10^{-14}$
1024	4.677073	$0.798845 * 10^{-14}$
2048	44.991274	$0.577649 * 10^{-14}$

Вычисление ошибки $\frac{\|QR-A\|_F}{\|A\|_F}$ для случайно-сгенерированной матрицы A_k вида

$$A_k = H_1 \text{diag}(\sigma_1, \dots, \sigma_n) H_2$$

где H_1, H_2 - случайные матрицы Хаусхолдера, и $\frac{\sigma_1}{\sigma_n} = 10^k$, причем $\sigma_1 > \sigma_2 > \dots > \sigma_{n-1} > \sigma_n$:

Размерность N	Значение k	Точность
1024	0	$0.397757 * 10^{-14}$
1024	4	$0.786147 * 10^{-14}$
1024	8	$0.801663 * 10^{-14}$
1024	12	$0.758398 * 10^{-14}$
1024	16	$0.765435 * 10^{-14}$
1024	20	$0.681711 * 10^{-14}$

Программа переписана с использованием библиотеки BLAS, сборка на компиляторе gcc:

Размерность N	Время выполнения(сек)	Точность
256	0.029024	$0.15 * 10^{-14}$
512	0.254628	$0.21 * 10^{-14}$
1024	2.382585	$0.35 * 10^{-14}$
2048	20.111016	$0.56 * 10^{-14}$

2 Программа на Fortran

Вычисление времени выполнения и ошибки $\frac{\|QR-A\|_F}{\|A\|_F}$ для случайно-сгенерированной матрицы A :

Размерность N	Время выполнения(сек)	Точность
256	0.144997	$0.290257 * 10^{-14}$
512	1.164560	$0.212471 * 10^{-14}$
1024	13.105832	$0.313008 * 10^{-14}$
2048	115.974884	$0.649089 * 10^{-14}$

Если скомпилировать программу на gfortran с флагом оптимизации -O3, то получим такой результат:

Размерность N	Время выполнения(сек)	Точность
256	0.025870	$0.159191 * 10^{-14}$
512	0.218357	$0.247889 * 10^{-14}$
1024	3.211895	$0.310742 * 10^{-14}$
2048	31.507637	$0.510653 * 10^{-14}$

Вычисление ошибки $\frac{\|QR-A\|_F}{\|A\|_F}$ для случайно-сгенерированной матрицы A_k вида

$$A_k = H_1 \text{diag}(\sigma_1, \dots, \sigma_n) H_2$$

где H_1, H_2 - случайные матрицы Хаусхолдера, и $\frac{\sigma_1}{\sigma_n} = 10^k$, причем $\sigma_1 > \sigma_2 > \dots \sigma_{n-1} > \sigma_n$:

Размерность N	Значение k	Точность
1024	0	$0.400560 * 10^{-14}$
1024	4	$0.772869 * 10^{-14}$
1024	8	$0.747257 * 10^{-14}$
1024	12	$0.838277 * 10^{-14}$
1024	16	$0.887717 * 10^{-14}$
1024	20	$1.013665 * 10^{-14}$

После замены циклов на работу со срезами, без опции оптимизации:

Размерность N	Время выполнения(сек)	Точность
256	0.008615	$0.200607 * 10^{-14}$
512	0.759589	$0.420117 * 10^{-14}$
1024	6.479639	$0.389186 * 10^{-14}$
2048	51.728363	$0.668168 * 10^{-14}$

После замены циклов на работу со срезами, с опцией оптимизации -O3:

Размерность N	Время выполнения(сек)	Точность
256	0.0072361	$0.159900 * 10^{-14}$
512	0.627861	$0.348846 * 10^{-14}$
1024	6.162336	$0.324858 * 10^{-14}$
2048	50.556617	$0.531992 * 10^{-14}$

Программа переписана с использованием библиотеки BLAS, сборка на компиляторе gfortran:

Размерность N	Время выполнения(сек)	Точность
256	0.0027562	$0.159116 * 10^{-14}$
512	0.235498	$0.231419 * 10^{-14}$
1024	2.441742	$0.379806 * 10^{-14}$
2048	20.028060	$0.422820 * 10^{-14}$

3 Лараск и mkl

Компиляция на `icc` и `gfort` с ключом `-mkl=sequential` и `-O3`. Программа написанная на Blas Vs программа написанная на Лараск.

Язык	Лараск/Blas	Размерность N	Время выполнения(сек)
C	Лараск	2048	0.64
C	Blas	2048	10.65
Fortran	Лараск	2048	0.65
Fortran	Blas	2048	10.16