Нижегородский государственный университет
им. Н.И.Лобачевского

Факультет Вычислительной математики и кибернетики

Параллельные численные методы

Метод прогонки

При поддержке компании Intel

Баркалов К.А.,
Кафедра математического обеспечения ЭВМ

Содержание

Постановка задачи

- Рассмотрим систему из n линейных алгебраических уравнений вида

$$
\begin{aligned}
a_{11}x_1 &+ a_{12}x_2 &+ \ldots + a_{1n}x_n &= b_1 \\
a_{21}x_1 &+ a_{21}x_2 &+ \ldots + a_{2n}x_n &= b_2 \\
\ldots \\
a_{n1}x_1 &+ a_{n2}x_2 &+ \ldots + a_{nn}x_n &= b_n
\end{aligned}
$$

- В матричном виде система может быть представлена как

$$Ax=b$$

- $A=(a_{ij})$ есть вещественная матрица размера n×n; b и x – вектора из n элементов.
- Будем искать значения вектора неизвестных x, при которых выполняются все уравнения системы.

Ленточные матрицы

- Матрица A называется *ленточной*, когда все ее ненулевые элементы находятся вблизи главной диагонали, т.е.
$$a_{ij}=0, \text{ если } j > i + k_1,\ i > j + k_2, \text{ где } k_1, k_2 < n.$$
Числа k_1 и k_2 называют шириной верхней и нижней полуленты.
Тогда $k_1 + k_2 + 1$ – ширина ленты матрицы.
- Важным классом являются трехдиагональные матрицы (при $k_1 = k_2 = 1$).
- Подобные системы уравнений возникают в задаче сплайн-интерполяции, при решении дифференциальных уравнений.
- Для решения задач с трехдиагональными матрицами существуют специальные методы.

Трехдиагональные матрицы

a1	c1											f1
b2	a2	c2										f2
	b3	a3	c3									f3
		b4	a4	c4								f4
			b5	a5	c5							f5
				b6	a6	c6						f6
					b7	a7	c7					f7
						b8	a8	c8				f8
							b9	a9	c9			f9
								b10	a10	c10		f10
									b11	a11	c11	f11
										b12	a12	f12

Метод прогонки (прямой ход)

- Предположим, что выполнено $x_{i-1}=\alpha_i x_i+\beta_i$ (1)
- Подставим (1) в *i*-е уравнение системы, получим

$$(\alpha_i a_i+c_i)x_i+b_i x_{i+1}=f_i-a_i\beta_i \qquad (2)$$

- Сравнивая (2) и выражение $x_i=\alpha_{i+1}x_{i+1}+\beta_{i+1}$, получим

$$\alpha_{i+1}=\frac{-b_i}{a_i\alpha_i+c_i} \qquad \beta_{i+1}=\frac{f_i-a_i\beta_i}{a_i\alpha_i+c_i} \qquad (3)$$

- Из первого уравнения $c_1x_1+b_1x_2=f_1$ находим

$$\alpha_2=-b_1/c_1,\ \beta_2=f_1/c_1.$$

- Затем вычисляем α_{i+1},β_{i+1} ,используя (3) при $i=2,\ldots,n-1$.

Метод прогонки (обратный ход)

- Определим x_n из последнего уравнения

$$x_{n-1}=\alpha_n x_n+\beta_n \qquad x_n=\frac{f_n-a_n\beta_n}{a_n\alpha_n+c_n}$$
$$a_n x_{n-1}+c_n x_n=f_n$$

- Используя $x_i=\alpha_{i+1}x_{i+1}+\beta_{i+1}$, находим все x_i, $i=n-1,\ldots,1$.

- Теорема
- Метод прогонки будет вычислительно устойчивым, если коэффициенты системы уравнений удовлетворяют условиям диагонального преобладания

$$|c_1|\geq|b_1|,\ |c_n|\geq|a_n|,$$
$$|c_i|>|a_i|+|b_i|\ ,\ i=2,\ldots,n-1.$$

Левая прогонка

- Рассмотренный в предыдущем пункте метод прогонки, при котором определение x_i происходит последовательно справа налево, называют *правой прогонкой*.
- Аналогично выписываются формулы *левой прогонки*
- Прямой ход

$$\xi_n = -a_n / c_n,\ \xi_i = \frac{-a_i}{c_i + b_i \xi_{i+1}},\ \eta_n = f_n / c_n,\ \eta_i = \frac{f_i - b_i \eta_{i+1}}{c_i + b_i \xi_{i+1}},\ i = n-1, \ldots, 2.$$

- Обратный ход

$$x_1 = \frac{f_1 - b_1 \eta_2}{b_1 \xi_2 + c_1},\ x_{i+1} = \xi_{i+1} x_i + \eta_{i+1},\ i = 1, \ldots, n-1.$$

Левая прогонка

- Предположим, что $x_{i+1}=\xi_{i+1}x_i+\eta_{i+1}$
- исключим из i-го уравнение системы переменную x_{i+1}

$$a_i x_{i-1}+c_i x_i+b_i(\xi_{i+1}x_i+\eta_{i+1})=f_i$$

$$x_i=\frac{-a_i}{c_i+b_i\xi_{i+1}}x_{i-1}+\frac{f_i-b_i\eta_{i+1}}{c_i+b_i\xi_{i+1}}$$

- Сравнивая с $x_i=\xi_i x_{i-1}+\eta_i$, получаем искомые формулы прямого и обратного хода левой прогонки

Встречная прогонка

- Комбинация левой и правой прогонок дает метод *встречной прогонки*, которые допускает распараллеливание на два потока.
- Разделим систему между двумя потоками – первый будет оперировать уравнениями с номерами $1 \le i \le p$, второй – уравнениями $p \le i \le n$, $p = \lceil n/2 \rceil$
- В первом потоке по формулам (1.4) вычисляются коэффициенты α_i, β_i, при $1 \le i \le p$; во втором потоке по формулам (1.8) вычисляются коэффициенты ξ_i, η_i, $p \le i \le n$.
- При $i=p$ проводится сопряжение решений в форме (1.6) и (1.9): находим значение x_p из системы

$$x_p = \alpha_{p+1} x_{p+1} + \beta_{p+1}$$
$$x_{p+1} = \xi_{p+1} x_p + \eta_{p+1}$$

- В первом потоке находим x_i при $1 \le i < p$, а во втором все x_i при $p < i \le n$.

Встречная прогонка в двух потоках

a1	c1											f1
b2	a2	c2										f2
	b3	a3	c3									f3
		b4	a4	c4								f4
			b5	a5	c5							f5
				b6	a6	c6						f6
					b7	a7	c7					f7
						b8	a8	c8				f8
							b9	a9	c9			f9
								b10	a10	c10		f10
									b11	a11	c11	f11
										b12	a12	f12

Встречная прогонка в двух потоках

a1	c1											f1
	a2	c2										f2
	b3	a3	c3									f3
		b4	a4	c4								f4
			b5	a5	c5							f5
				b6	a6	c6						f6
					b7	a7	c7					f7
						b8	a8	c8				f8
							b9	a9	c9			f9
								b10	a10	c10		f10
									b11	a11		f11
										b12	a12	f12

Встречная прогонка в двух потоках

a1	c1											f1
	a2	c2										f2
		a3	c3									f3
		b4	a4	c4								f4
			b5	a5	c5							f5
				b6	a6	c6						f6
					b7	a7	c7					f7
						b8	a8	c8				f8
							b9	a9	c9			f9
								b10	a10			f10
									b11	a11		f11
										b12	a12	f12

Встречная прогонка в двух потоках

a1	c1											f1
	a2	c2										f2
		a3	c3									f3
			a4	c4								f4
			b5	a5	c5							f5
				b6	a6	c6						f6
					b7	a7	c7					f7
						b8	a8	c8				f8
							b9	a9				f9
								b10	a10			f10
									b11	a11		f11
										b12	a12	f12

Встречная прогонка в двух потоках

a1	c1											f1
	a2	c2										f2
		a3	c3									f3
			a4	c4								f4
				a5	c5							f5
				b6	a6	c6						f6
					b7	a7	c7					f7
						b8	a8					f8
							b9	a9				f9
								b10	a10			f10
									b11	a11		f11
										b12	a12	f12

Встречная прогонка в двух потоках

a1	c1											f1
	a2	c2										f2
		a3	c3									f3
			a4	c4								f4
				a5	c5							f5
					a6	c6						f6
					b7	a7						f7
						b8	a8					f8
							b9	a9				f9
								b10	a10			f10
									b11	a11		f11
										b12	a12	f12

Встречная прогонка в двух потоках

a1	c1											f1
	a2	c2										f2
		a3	c3									f3
			a4	c4								f4
				a5	c5							f5
					a6	c6						f6
						a7						f7
						b8	a8					f8
							b9	a9				f9
								b10	a10			f10
									b11	a11		f11
										b12	a12	f12

Встречная прогонка в двух потоках

a1	c1											f1
	a2	c2										f2
		a3	c3									f3
			a4	c4								f4
				a5	c5							f5
					a6							f6
						a7						f7
							a8					f8
							b9	a9				f9
								b10	a10			f10
									b11	a11		f11
										b12	a12	f12

Встречная прогонка в двух потоках

a1	c1											f1
	a2	c2										f2
		a3	c3									f3
			a4	c4								f4
				a5								f5
					a6							f6
						a7						f7
							a8					f8
								a9				f9
								b10	a10			f10
									b11	a11		f11
										b12	a12	f12

Параллельная блочная прогонка

- Применим блочный подход к разделению данных: пусть каждый поток обрабатывает $m = \lfloor n/p \rfloor$ строк матрицы A, т.е. k-й поток обрабатывает строки с номерами $1+(k-1)m \le i \le km$. Будем предполагать, что число уравнений в системе кратно числу потоков.
- В пределах полосы матрицы можно организовать исключение поддиагональных элементов матрицы (*прямой ход* метода).
  вычитание строки i, умноженной на константу b_{i+1}/a_i, из строки $i+1$ с тем, чтобы результирующий коэффициент при неизвестной x_i в $(i+1)$-й строке оказался нулевым

Параллельная блочная прогонка

a1	c1											d1
b2	a2	c2										d2
	b3	a3	c3									d3
		b4	a4	c4								d4
			b5	a5	c5							d5
				b6	a6	c6						d6
					b7	a7	c7					d7
						b8	a8	c8				d8
							b9	a9	c9			d9
								b10	a10	c10		d10
									b11	a11	c11	d11
										b12	a12	d12

Параллельная блочная прогонка

a1	c1											d1
	a2	c2										d2
	b3	a3	c3									d3
		b4	a4	c4								d4
			b5	a5	c5							d5
			f6		a6	c6						d6
					b7	a7	c7					d7
						b8	a8	c8				d8
							b9	a9	c9			d9
							f10		a10	c10		d10
									b11	a11	c11	d11
										b12	a12	d12

Параллельная блочная прогонка

a1	c1											d1
	a2	c2										d2
		a3	c3									d3
		b4	a4	c4								d4
			b5	a5	c5							d5
			f6		a6	c6						d6
			f7			a7	c7					d7
						b8	a8	c8				d8
							b9	a9	c9			d9
							f10		a10	c10		d10
							f11			a11	c11	d11
							f12			b12	a12	d12

Параллельная блочная прогонка

a1	c1											d1
	a2	c2										d2
		a3	c3									d3
			a4	c4								d4
			b5	a5	c5							d5
			f6		a6	c6						d6
			f7			a7	c7					d7
			f8				a8	c8				d8
							b9	a9	c9			d9
							f10		a10	c10		d10
							f11			a11	c11	d11
							f12				a12	d12

Параллельная блочная прогонка

- Если исключение первым потоком поддиагональных переменных не добавит в матрицу новых коэффициентов, то исключение поддиагональных элементов в остальных потоках приведет к возникновению столбца отличных от нуля коэффициентов: во всех блоках (кроме первого) число ненулевых элементов в строке не изменится, но изменится структура уравнений. Модификации также подвергнутся элементы вектора правой части.
- Затем выполняется обратный ход алгоритма – каждый поток исключает наддиагональные элементы, начиная с последнего

Параллельная блочная прогонка

a1	c1											d1
	a2	c2										d2
		a3		g3								d3
			a4	c4								d4
			b5	a5	c5							d5
			f6		a6	c6						d6
			f7			a7		g7				d7
			f8				a8	c8				d8
							b9	a9	c9			d9
							f10		a10	c10		d10
							f11			a11		d11
							f12				a12	d12

Параллельная блочная прогонка

a1	c1											d1
	a2			g2								d2
		a3		g3								d3
			a4	c4								d4
			b5	a5	c5							d5
			f6		a6			g6				d6
			f7			a7		g7				d7
			f8				a8	c8				d8
							b9	a9	c9			d9
							f10		a10			d10
							f11			a11		d11
							f12				a12	d12

Параллельная блочная прогонка

a1				g1								d1
	a2			g2								d2
		a3		g3								d3
			a4	c4								d4
			b5	a5				g5				d5
			f6		a6			g6				d6
			f7			a7		g7				d7
			f8				a8	c8				d8
							b9	a9				d9
							f10		a10			d10
							f11			a11		d11
							f12				a12	d12

Параллельная блочная прогонка

- После выполнения обратного хода матрица стала блочной. Исключим из нее внутренние строки каждой полосы, в результате получим систему уравнения относительно части исходный неизвестных
- Данная система будет содержать 2*p* уравнений, и будет трехдиагональной. Ее можно решить последовательным методом прогонки. После того, как эта система будет решена, станут известны значения неизвестных на границах полос разделения данных. Далее можно за один проход найти значения внутренних переменных.

Параллельная блочная прогонка

a1		g1				d1
	a4	c4				d4
	b5	a5		g5		d5
	f8		a8	c8		d8
			b9	a9		d9
			f12		a12	d12

Параллельная блочная прогонка

- После выполнения обратного хода матрица стала блочной. Исключим из нее внутренние строки каждой полосы, в результате получим систему уравнения относительно части исходный неизвестных
- Данная система будет содержать $2p$ уравнений, и будет трехдиагональной. Ее можно решить последовательным методом прогонки. После того, как эта система будет решена, станут известны значения неизвестных на границах полос разделения данных. Далее можно за один проход найти значения внутренних переменных.

Параллельная блочная прогонка (вариант 2)

- Рассмотренный способ распараллеливания уже дает хорошие результаты, но можно использовать лучшую стратегию исключения неизвестных.
- Прямой ход нового алгоритма будет таким же.
- Обратный ход: каждый поток исключает наддиагональные элементы, начиная со своего предпоследнего, и заканчивая последним для предыдущего блока матрицы.

Параллельная блочная прогонка (вариант 2)

a1	c1											d1
	a2	c2										d2
		a3	c3									d3
			a4	c4								d4
			b5	a5	c5							d5
			f6		a6	c6						d6
			f7			a7	c7					d7
			f8				a8	c8				d8
							b9	a9	c9			d9
							f10		a10	c10		d10
							f11			a11	c11	d11
							f12				a12	d12

Параллельная блочная прогонка (вариант 2)

a1	c1											d1
	a2		g2									d2
		a3	c3									d3
			a4	c4								d4
			b5	a5	c5							d5
			f6		a6		g6					d6
			f7			a7	c7					d7
			f8				a8	c8				d8
							b9	a9	c9			d9
							f10		a10		g10	d10
							f11			a11	c11	d11
							f12				a12	d12

Параллельная блочная прогонка (вариант 2)

a1			g1									d1
	a2		g2									d2
		a3	c3									d3
			a4	c4								d4
			b5	a5			g5					d5
			f6		a6		g6					d6
			f7			a7	c7					d7
			f8				a8	c8				d8
							b9	a9			g9	d9
							f10		a10		g10	d10
							f11			a11	c11	d11
							f12				a12	d12

Параллельная блочная прогонка (вариант 2)

a1			g1									d1
	a2		g2									d2
		a3	c3									d3
			a4				g5					d4
			b5	a5			g5					d5
			f6		a6		g6					d6
			f7			a7	c7					d7
			f8				a8				g8	d8
							b9	a9			g9	d9
							f10		a10		g10	d10
							f11			a11	c11	d11
							f12				a12	d12

Параллельная блочная прогонка (вариант 2)

- Можно сформировать вспомогательную задачу меньшего размера. Исключим из матрицы все строки каждой полосы, кроме последней, в результате получим систему уравнения относительно части исходный неизвестных.
- Даная система будет содержать всего *p* уравнений, и также будет трехдиагональной. Ее можно решить последовательным методом прогонки.
- После того, как эта система будет решена, станут известны значения неизвестных на нижних границах полос разделения данных. Далее можно за один проход найти значения внутренних переменных в каждом потоке.

Параллельная блочная прогонка (вариант 2)

a4	g4		d4
f8	a8	g8	d8
	f12	a12	d12

Оценка эффективности

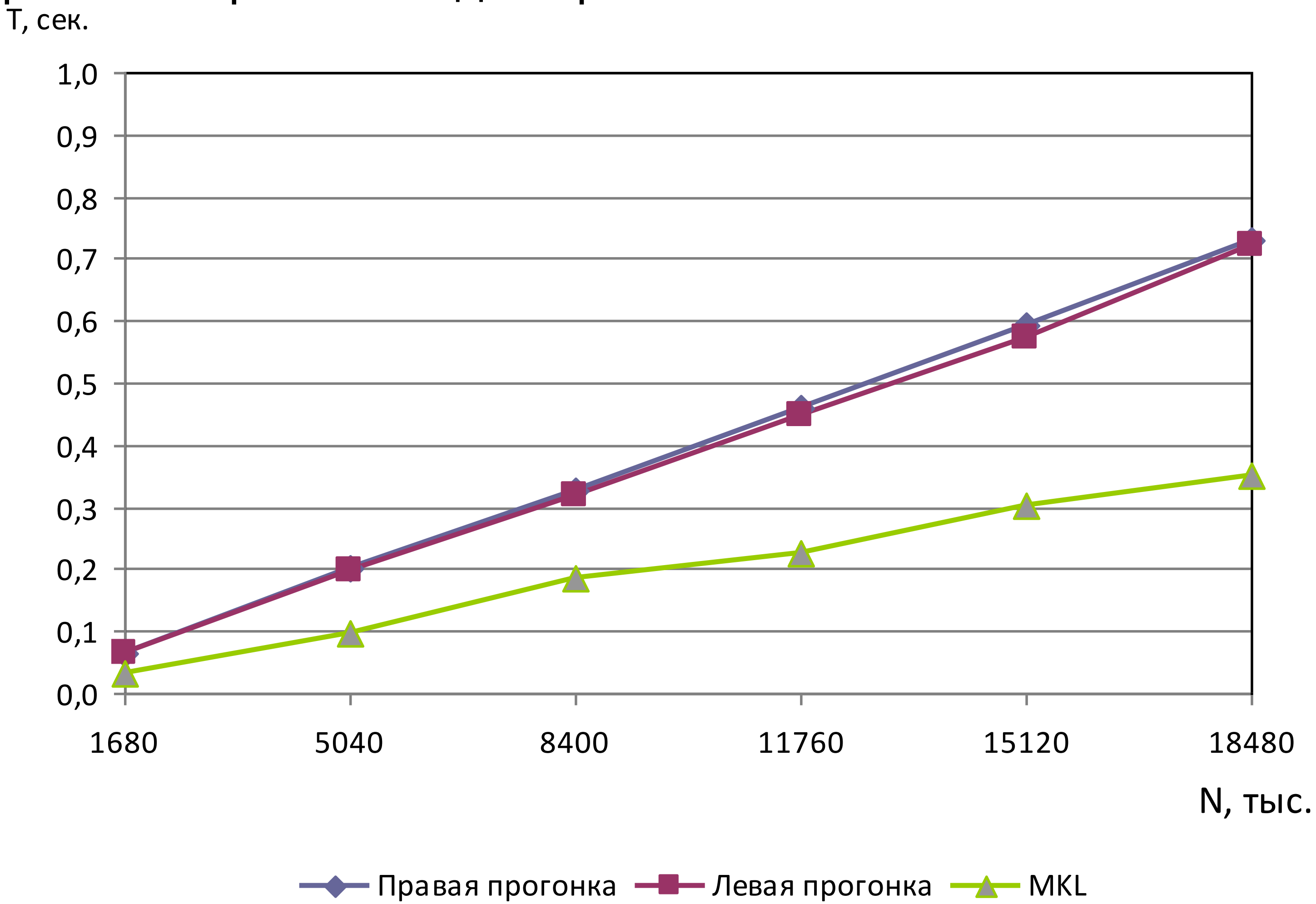
- Ускорение $$S_p = \frac{T_1}{T_p} \approx \frac{n}{2\frac{n}{p} + p} = p\frac{n}{2n + p^2}$$

- Для систем с общей памятью $n >> p$, поэтому

ускорение $$S_p = p\frac{n}{2n + p^2} \approx \frac{p}{2}$$

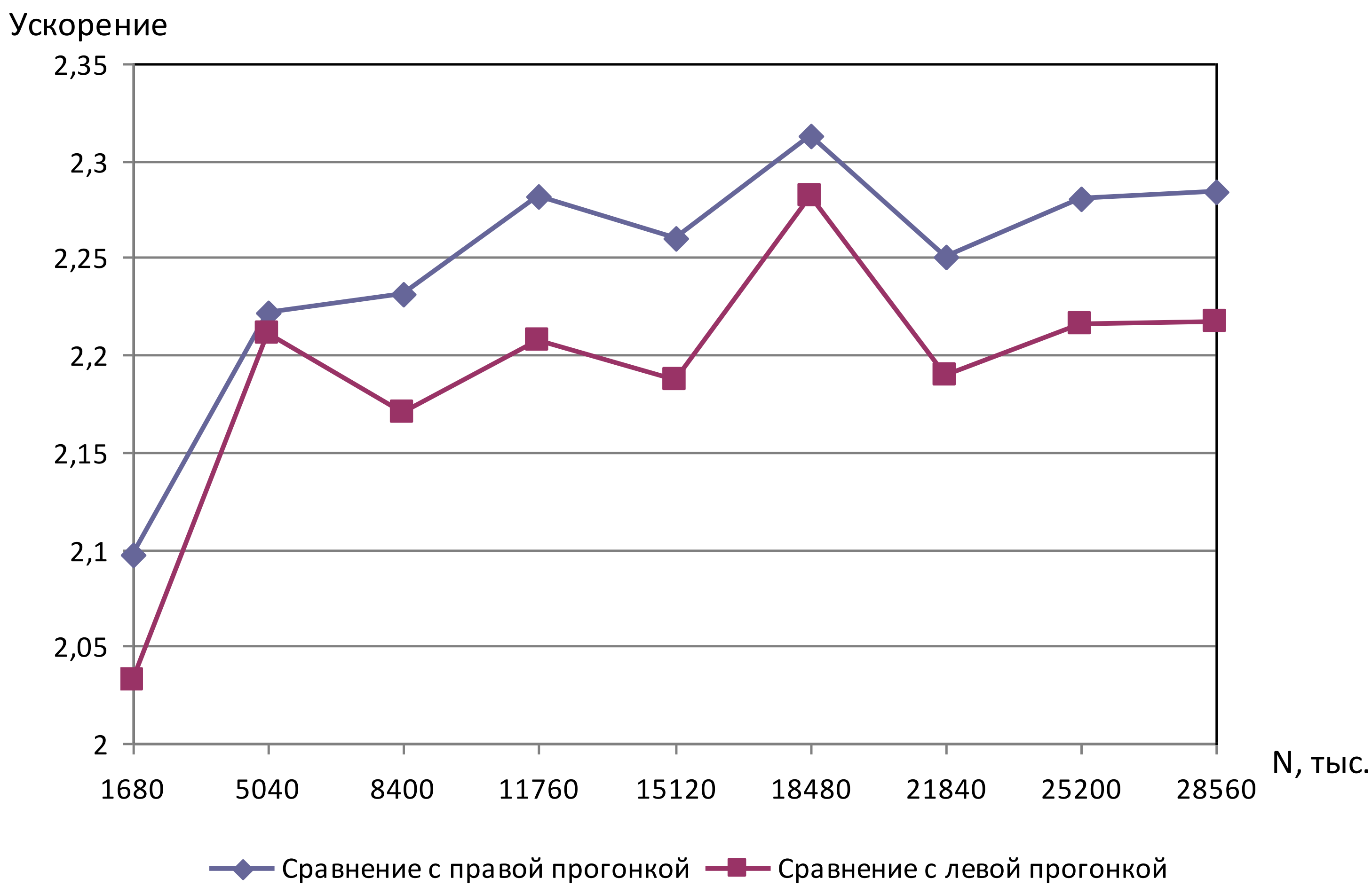
Результаты экспериментов

- Сравнение разных видов прогонки



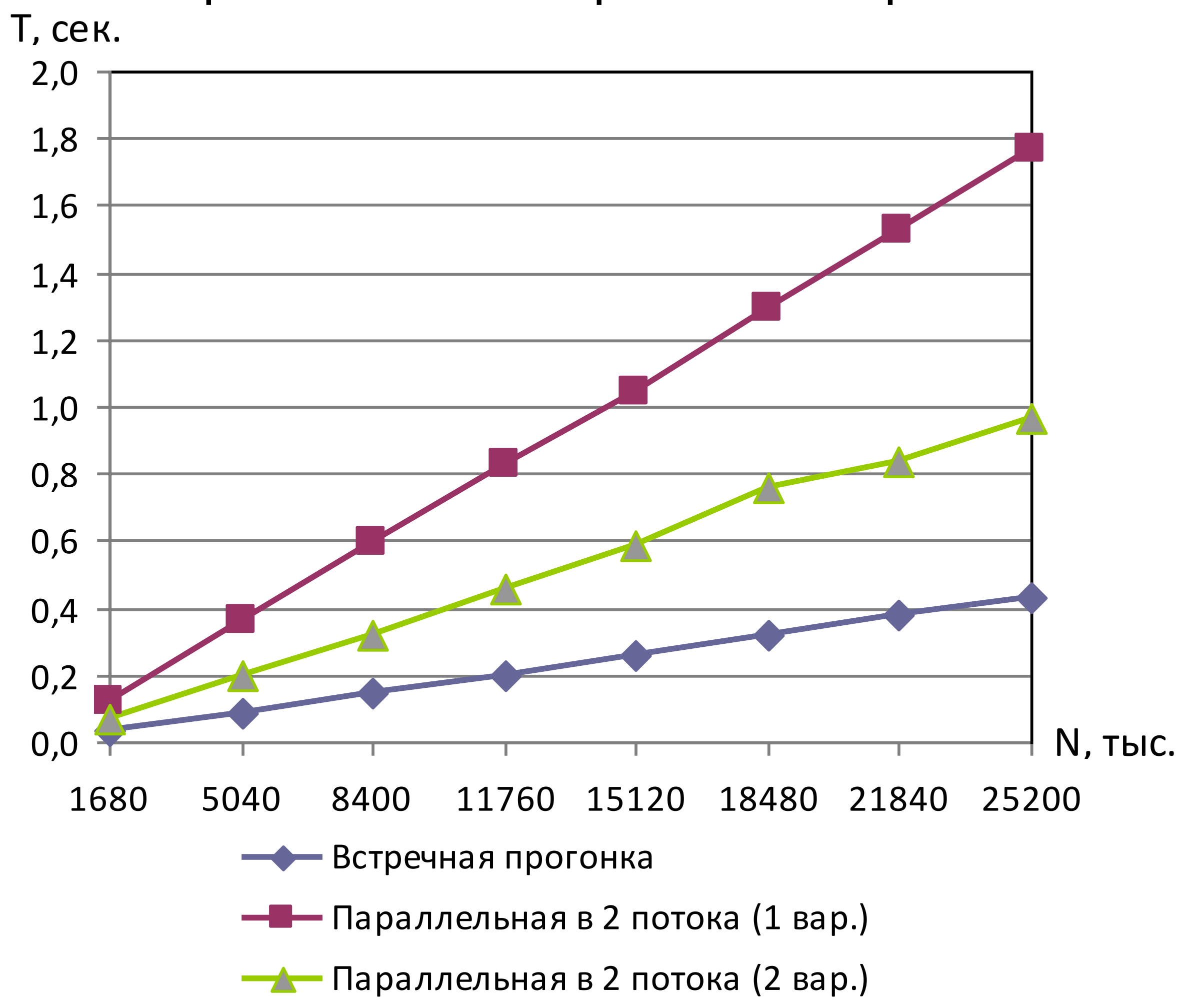
Результаты экспериментов

- Ускорение параллельной встречной прогонки



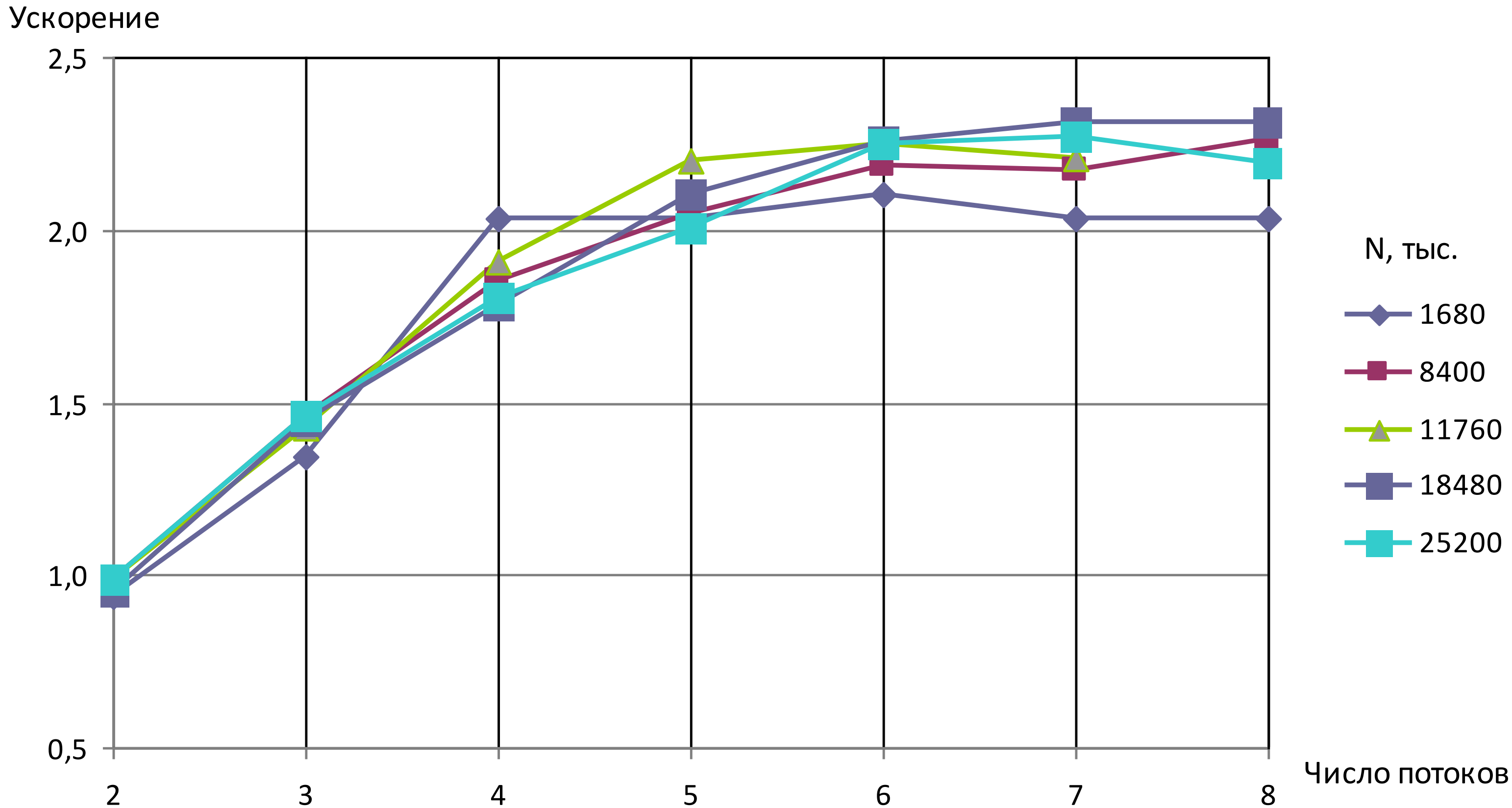
Результаты экспериментов

- Сравнение параллельной встречной и параллельной блочной прогонки



Результаты экспериментов

- Ускорение параллельной блочной прогонки



Литература

1. Вержбицкий В.М. Численные методы (математический анализ и обыкновенные дифференциальные уравнения). – М.: Высшая школа, 2001.
2. Самарский А.А., Гулин А.В. Численные методы. – М.: Наука, 1989.
3. Голуб Дж., Ван Лоун Ч. Матричные вычисления. – М.: Мир, 1999.

Ресурсы сети Интернет

4. Интернет-университет суперкомпьютерных технологий. [http://www.hpcu.ru].
5. Intel Math Kernel Library Reference Manual. [http://software.intel.com/sites/products/documentation/hpc/mkl/mklman.pdf].

Авторский коллектив

- Баркалов Константин Александрович,
  к.ф.-м.н., старший преподаватель кафедры Математического обеспечения ЭВМ факультета ВМК ННГУ.
  barkalov@fup.unn.ru

- Коды учебных программ разработаны Маловой Анной и Сафоновой Яной