

ЭФФЕКТИВНОСТЬ ПАРАЛЛЕЛЬНЫХ ВЫЧИСЛЕНИЙ В СРЕДЕ R

В. И. Аникин, О. В. Аникина, О. Н. Чичаева

Кафедра Информационный и электронный сервис, Поволжский государственный университет сервиса
Кафедра Прикладная математика и информатика, Тольяттинский государственный университет
Факультет техники и технологий сервиса, Поволжский государственный университет сервиса
Тольятти, РФ

E-mail: anikin_vi@mail.ru, blue-waterfall@yandex.ru, OlgachiNik@yandex.ru

Проведено сравнение временной эффективности выполнения векторных и матричных операций в R. Показано, что при больших размерах векторов и матриц отношение времен выполнения последовательного и параллельного алгоритмов асимптотически стремится к константе, составляющей для векторов величину в несколько сотен, для матриц – в несколько тысяч раз.

ВВЕДЕНИЕ

R - интерпретируемый язык программирования для статистической обработки и визуализации данных, а также свободная программная среда с открытым исходным кодом [1]. Базовая установка R предоставляет команды, решающие многие общие математические и научно-технические задачи. R обладает обширными графическими возможностями и поддерживает векторизованные вычисления, что позволяет создавать качественные графики при работе с пространственно-распределенными данными и выполнять параллельные вычисления, избегая использования циклов и обеспечивая значительное ускорение вычислений при обработке объемных данных [2, 3].

Практически значимыми и актуальными в современных наукоемких технологиях стали задачи большой размерности и вычислительной сложности, программные системы обработки потоков задач и обслуживания потоков запросов со значительными объемами обрабатываемых данных, эффективные программные средства для бортовых вычислительных систем, работающих в режиме реального времени в условиях ограниченных вычислительных ресурсов.

Целью работы являлось сравнение временной эффективности последовательных и параллельных алгоритмов в R. Для достижения поставленной цели решались следующие задачи:

- реализовать алгоритмы последовательного и параллельного вычисления суммы элементов и скалярного произведения, вычисления среднего, минимального значения, стандартного отклонения, ковариации, корреляции, сортировки векторов;
- реализовать алгоритмы последовательного и параллельного вычисления для матриц: произведения, определителя, обратной матрицы, решения системы линейных уравнений, сингулярного разложения, ковариации, корреляции;
- определить временную эффективность перечисленных последовательных и парал-

лельных алгоритмов и сравнить их друг с другом.

I. ПАРАЛЛЕЛЬНЫЕ И ПОСЛЕДОВАТЕЛЬНЫЕ ВЕКТОРНЫЕ ВЫЧИСЛЕНИЯ

Эксперимент по сравнению временной эффективности последовательных и параллельных алгоритмов векторных вычислений проводился для следующих функций R: суммирование элементов вектора, скалярное произведение двух векторов. Значения элементов векторов задавались случайным образом с помощью функции `runif()`.

Для определения затраченного времени вычислений использовалась функция `proc.time()`, которая возвращает пользовательское, системное и реальное время, прошедшее от начала процесса. Результаты численного эксперимента для параллельных и последовательных векторных вычислений представлены в табл. 1–3, где t_1 , t_2 – время выполнения последовательного и параллельного алгоритма, n – размерность векторов.

Таблица 1 – Время выполнения алгоритмов вычисления суммы элементов вектора

n	2000	4000	8000	16000	32000
t1,мс	6.412	12.67	25.14	50.38	101.3
t2,мс	0.019	0.029	0.038	0.08	0.13
t1/t2	338	437	661	630	779

n	64000	128000	256000	512000
t1,мс	201	403	811	1624
t2,мс	0.29	0.54	1.10	2.29
t1/t2	693	745	736	710

Таблица 2 - Время выполнения алгоритмов вычисления скалярного произведения векторов

n	2000	4000	8000	16000	32000
t1,мс	9.71	19.52	39.01	77.62	159.3
t2,мс	0.101	0.188	0.354	0.701	1.416
t1/t2	96	104	110	111	113

n	64000	128000	256000	512000
t1,мс	314.4	624.1	1303.6	2562.8
t2,мс	2.891	5.625	11.248	22.156
t1/t2	109	111	116	116

Таблица 3 – Время параллельного выполнения основных векторных операций

n	t[мс], ср. знач.	t[мс], мин. знач.	t[мс], станд. откл.	t[мс], ковар.
5120	0.108	0.148	0.379	0.630
20480	0.296	0.548	0.852	1.754
81920	1.19	2.26	3.07	6.29
327680	4.26	8.68	11.62	23
1310720	17.1	34.5	181	89.5
5242880	66	142	0.472	344
20971520	265	585	730	1405

n	t[мс], коррел.	t[мс], “quick” сорт.	t[мс], “shell” сорт.
5120	0.6725	1.558	2.283
20480	2.31	6.936	10.83
81920	8.36	30.78	57.37
327680	33.06	141.48	271.1
1310720	124.3	591.9	1355
5242880	500	2596	6332
20971520	1980	11175	31150

II. ПАРАЛЛЕЛЬНЫЕ И ПОСЛЕДОВАТЕЛЬНЫЕ МАТРИЧНЫЕ ВЫЧИСЛЕНИЯ

Временная эффективность последовательных и параллельных алгоритмов определялась для произведения двух квадратных матриц размерностью $n \times n$. Элементы матриц задавались случайным образом с помощью функции `gnorm()`. В табл. 4 представлено время выполнения последовательного и параллельного алгоритмов для произведения двух квадратных матриц.

Таблица 4 - Время выполнения алгоритмов вычисления произведения двух матриц

n	16	32	64	128	256
t1,мс	20.0	157	1364	9860	79140
t2,мс	0.0099	0.049	0.29	2.14	16.31
t1/t2	2011	3222	4755	4605	4853

В табл. 5 представлены результаты численного эксперимента для матричных операций. Эксперимент проводился для параллельных алгоритмов вычисления следующих функций: определителя матрицы, обратной матрицы, решения системы линейных уравнений, сингулярного разложения, ковариации, корреляции. Результаты вычислений представлены в табл. 5 по возрастанию времени выполнения. Из таблицы видно, что наименьшее время затрачено на

вычисление определителя матрицы, а наибольшее - на вычисление корреляционной матрицы.

Таблица 5 – Время параллельного выполнения основных матричных операций

n	t[мс], опред- тель	t[мс], обр. матрица	t[мс], решен. уравн.
8	0.015	0.031	0.044
16	0.021	0.042	0.070
32	0.035	0.094	0.168
64	0.11	0.41	0.55
128	0.64	2.59	2.40
256	4.33	19.26	12.46
512	29.4	142.4	70
1024	219	1298	372
2048	1990	9740	2460
4096	16450	76630	18990

n	t[мс], синг. разл.	t[мс], ковар.	t[мс], коррел.
8	0.082	0.16705	0.1646
16	0.168	0.2004	0.19625
32	0.512	0.3908	0.432
64	2.170	1.7116	2.3704
128	11.43	11.35	14.227
256	76.94	87.95	112.0
512	528.8	692.2	889.4
1024	4369	5425	7031
2048	33750	44660	54390
4096	278990	357520	430340

III. ЗАКЛЮЧЕНИЕ

В ходе исследования было выявлено, что параллельные алгоритмы значительно ускоряют работу программы. При больших размерах векторов и матриц отношение времен выполнения последовательного и параллельного алгоритмов асимптотически стремится к константе, составляющей для векторов величину в несколько сотен, для матриц – в несколько тысяч раз.

СПИСОК ЛИТЕРАТУРЫ

1. Кабаков, Р. И. R в действии. Анализ и визуализация данных в программе R / Р. И. Кабаков. – М.: ДМК Пресс, 2014. – 588 с.: ил.
2. Рычков А.Д. Численные методы и параллельные вычисления: учеб. пособие / А.Д. Рычков – Новосибирск: Изд-во СибГУТИ, 2007. – 144 с.
3. З. Мاستицкий, С.Э. Статистический анализ и визуализация данных с помощью R / С. Э. Мاستицкий, В. К. Шитиков. - 2014. – Режим доступа: <http://r-analytics.blogspot.com>