Сравнительный анализ производительности алгоритма пересечения автоматов

Решетников Илья

Программная инженерия

23 Сентября 2020г.

Цель:

Определить, какой из двух методов нахождения транзитивного замыкания быстрее: возведение в квадрат или умножение на матрицу смежности.

Оборудование:

Процессор: AMD Ryzen 3600 3.6GHz

RAM: 16Gb DDR4

Метод:

В качестве графов для алгоритма были взяты графы: LUBM300, LUBM500, LUBM1M, LUBM1.5M, LUBM1.9M

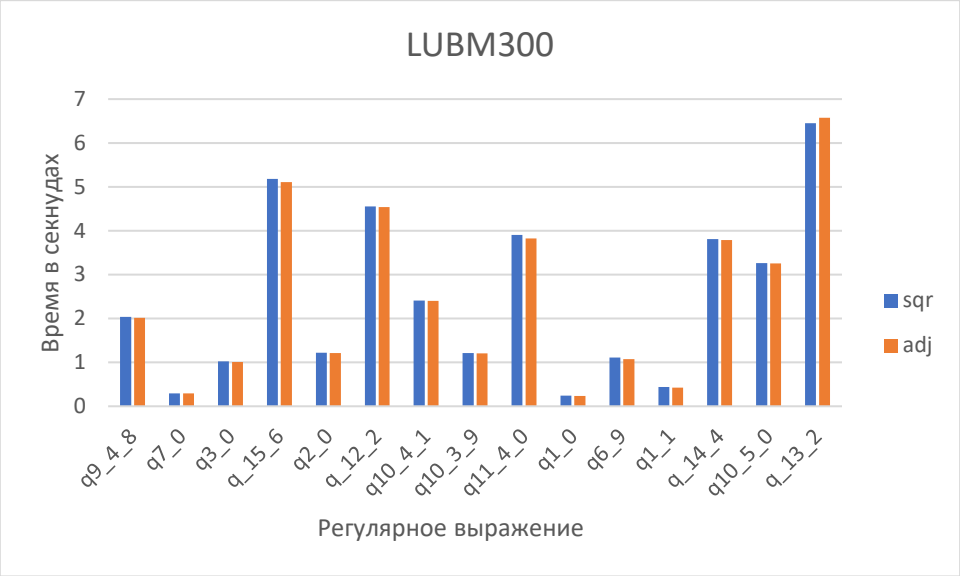
В качестве запросов были выбраны 15 регулярных выражений из папки refinedDataForRPQ: (https://drive.google.com/file/d/158g01o2rpdq5eL3Ari8e5SPbbeZTJspr/view?usp=sharing).

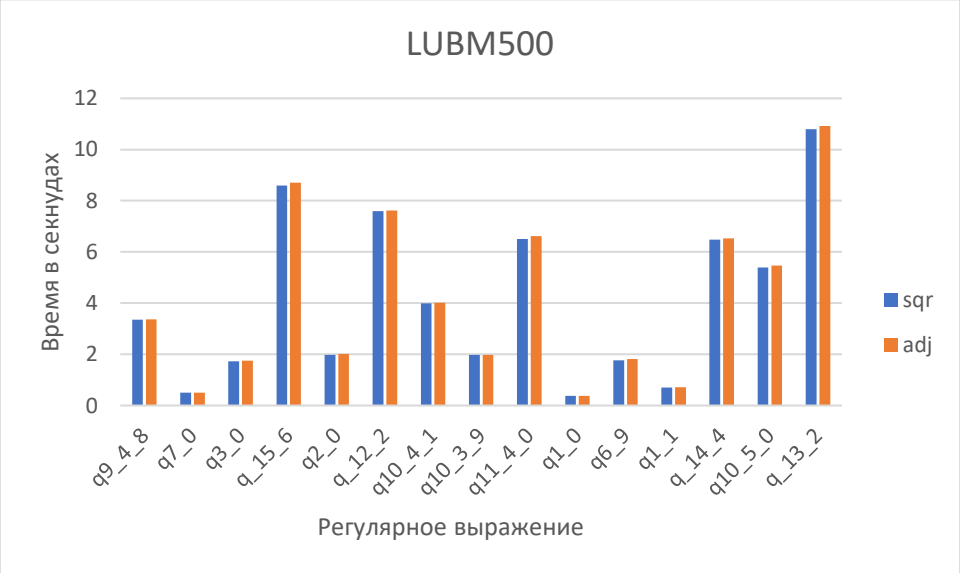
Алгоритмы запускались в Docker контейнере, для каждого графа и набора регулярных выражений была запущена версия алгоритма с поиском транзитивного замыкания посредством возведения в квадрат и умножения на матрицу смежности.

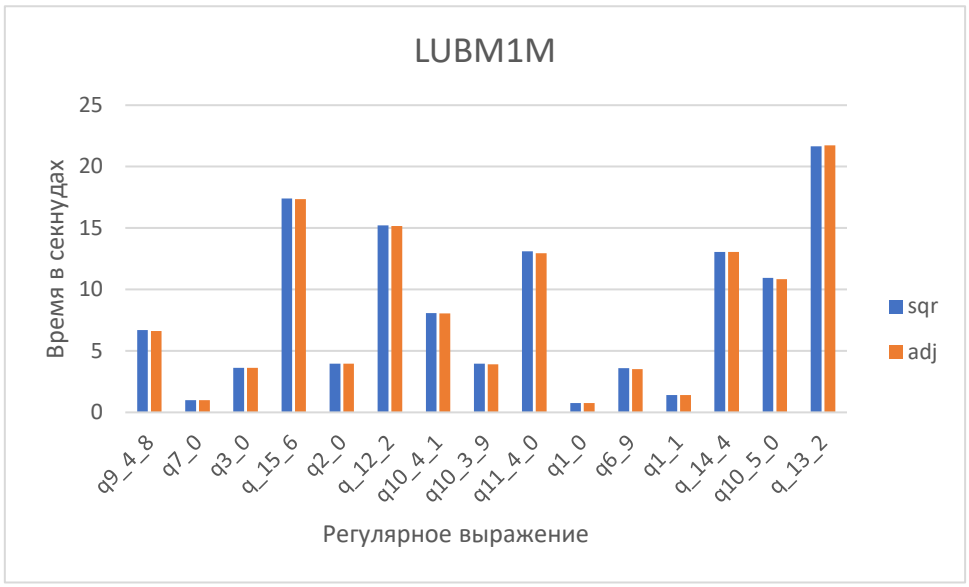
Так же были выполнены контрольные запуски, которые показали, что результаты для одинаковых запросов находятся в пределах погрешности (~0.1 секунда)

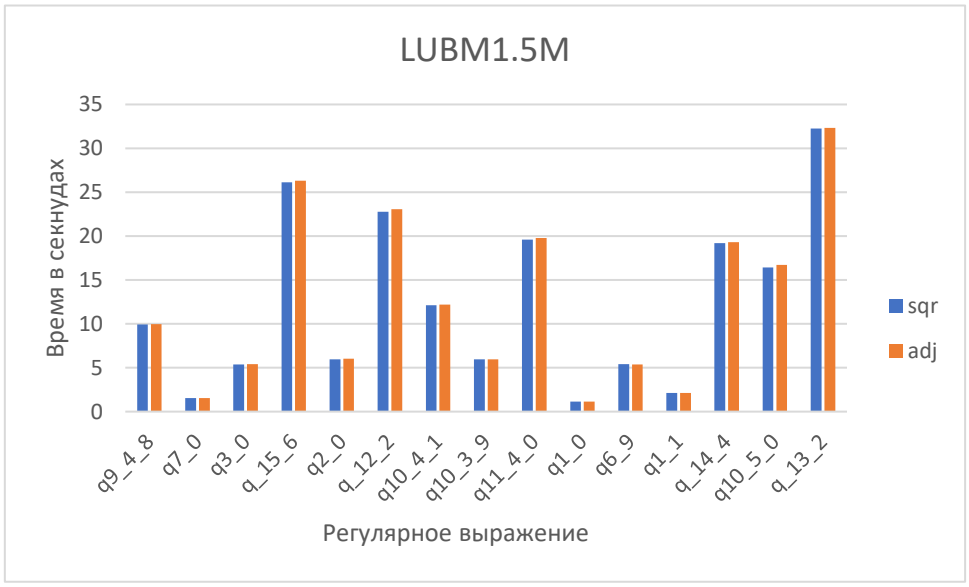
Результаты:

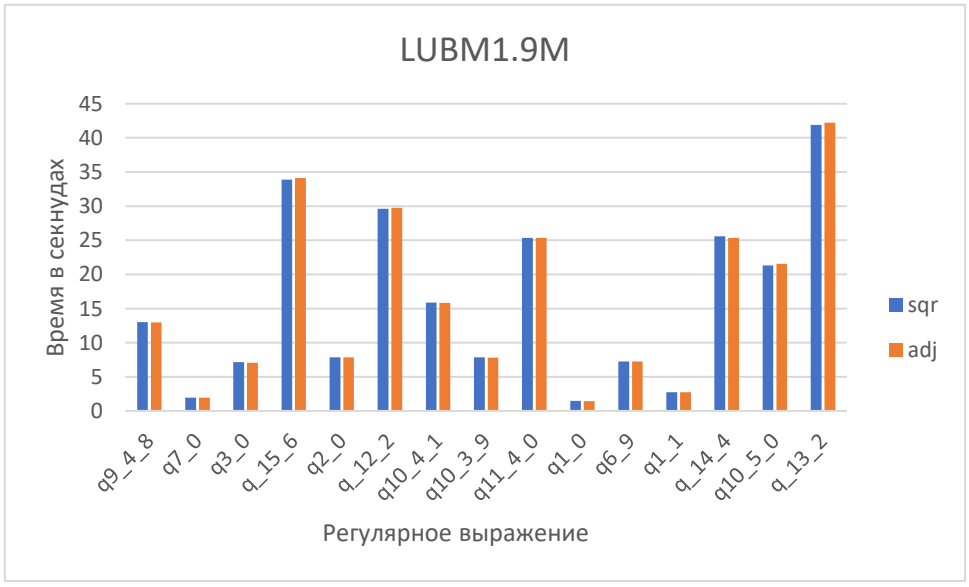
На диаграммах синие столбцы обозначают время работы алгоритма с использованием поиска транзитивного замыкания с помощью возведения в квадрат, а оранжевые столбцы – с помощью умножения на матрицу смежности






LUBM1M
Время в секнудах
25
20
15
10
5
0
q9_4_8
q7_0
q3_0
q_15_6
q2_0
q_12_2
q10_4_1
q10_3_9
q11_4_0
q1_0
q6_9
q1_1
q_14_4
q10_5_0
q_13_2
Регулярное выражение
sqr
adj


LUBM1.5M
Время в секнудах
35
30
25
20
15
10
5
0
q9_4_8
q7_0
q3_0
q_15_6
q2_0
q_12_2
q10_4_1
q10_3_9
q11_4_0
q1_0
q6_9
q1_1
q_14_4
q10_5_0
q_13_2
Регулярное выражение
sqr
adj


LUBM1.9M
Время в секнудах
45
40
35
30
25
20
15
10
5
0
q9_4_8
q7_0
q3_0
q_15_6
q2_0
q_12_2
q10_4_1
q10_3_9
q11_4_0
q1_0
q6_9
q1_1
q_14_4
q10_5_0
q_13_2
Регулярное выражение
sqr
adj

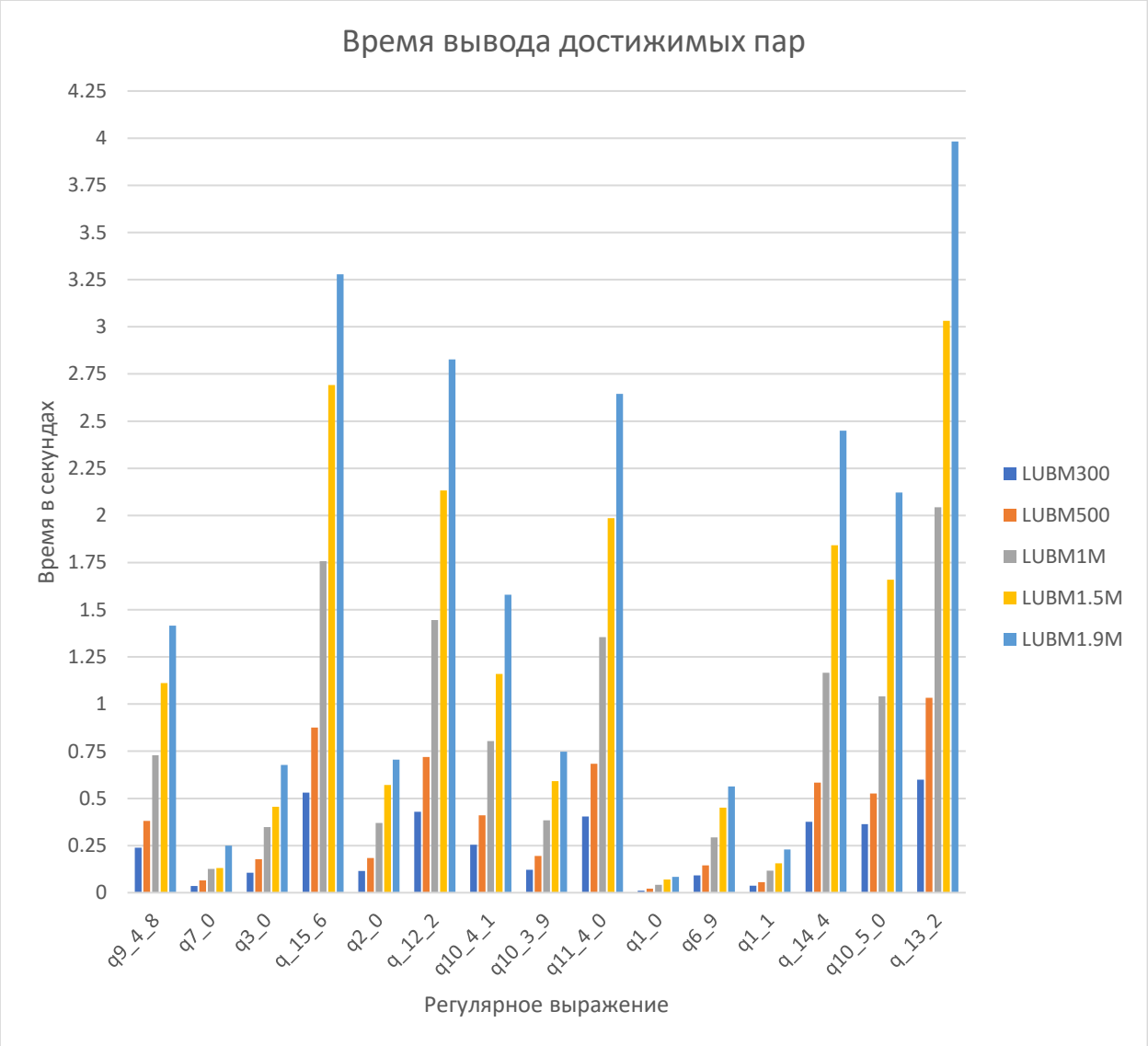
Вывод:

Судя по полученным графикам, нельзя однозначно сказать, какой алгоритм работает быстрее.

На достаточно больших запросах быстрее работает возведение в квадрат, в то время как на небольших - умножение на матрицу смежности

Дополнительно:

Так же было произведен замер времени, потраченного на вывод достижимых пар:



И подсчет контрольных сумм (маркированные вершины графа)

	LUBM300	LUBM500	LUBM1M	LUBM1.5M	LUBM1.9M
q9_4_8	327484	546592	1086330	1629948	2114445
q7_0	15810	26347	52921	79226	102615
q3_0	127707	213023	424165	636197	824897
q_15_6	554772	924073	1845255	2767182	3586030
q2_0	165983	276611	550900	826151	1070782
q_12_2	734831	1224013	2449749	3672395	4757607
q10_4_1	392027	653840	1305222	1957131	2536733
q10_3_9	173996	289965	577723	866306	1122791
q11_4_0	448994	746894	1495432	2240872	2902396
q1_0	7797	12993	26098	39071	50606
q6_9	233411	389473	774484	1161810	1506660
q1_1	44735	74299	148437	222456	287937
q_14_4	708542	1182394	2351709	3527852	4575422
q10_5_0	558405	929841	1858748	2785877	3609707
q_13_2	1033644	1724885	3436537	5154384	6683838