Московский Государственный Университет им. М.В. Ломоносова
Факультет Вычислительной Математики и Кибернетики
Кафедра Суперкомпьютеров и Квантовой Информатики

Практикум на ЭВМ: 6 семестр.

Отчёт № 4.

Анализ параллельной библиотеки на MPI, реализующей некоторые квантовые гейты

Работу выполнил

Федоров В. В.

Москва 2021

Постановка задачи и формат данных.

Задача: Реализовать параллельную библиотеку, реализующую гейты H, Hn, NOT, ROT, CNOT, CROT, проанализировать зависимость времени выполнения гейтов Hn b CNOT от числа кубитов, пары номеров преобразуемых кубитов (для CNOT) и числа процессов на системе BlueGene.

Описание программы

В предположении что число процессов $p=2^s$, где $s\in\mathbb{N}, s\leq\frac{n}{2}$ разделим вектор состояния на $2^{(s+2)}$ блоков одного размера, тогда каждому процессу достается 4 блока. В таком случае либо блоки можно будет разделить между процессами так, чтобы элементы двух блоков конкретного процесса были попарно соседними по k_1-му и k_2-му кубитам, либо соседние элементы уже находятся в одном и том же блоке, и тогда блоки можно распределить между процессорами как угодно, но для простоты будем выдавать процессам пары соседних блоков. Рассмотрим пример с n = 4 и s = 2:

Номер блока	Номер элемента
0	00000
	00001
1	00010
	00011
2	00100
	00101
3	00110
	00111
4	01000
	01001
5	01010
	01011
6	01100
	01101
7	01110
	01111
8	10000
	10001
9	10010
	10011
10	10100
	10101
11	10110
	10111

12	11000
	11001
13	11010
	11011
14	11100
	11101
15	11110
	11111

Номер блока	1,2	1,3	1,4	2,3	2,4	3,4
0	0	0	0	0	0	0
1	1	1	0	1	0	0
2	2	0	1	0	1	0
3	3	1	1	1	1	0
4	0	2	2	0	0	1
5	1	3	2	1	0	1
6	2	2	3	0	1	1
7	3	3	3	1	1	1
8	0	0	0	2	2	2
9	1	1	0	3	2	2
10	2	0	1	2	3	2
11	3	1	1	3	3	2
12	0	2	2	2	2	3
13	1	3	2	3	2	3
14	2	2	3	2	3	3
15	3	3	3	3	3	3

Таким образом, массив распределений M однозначно определяется для k_1 и k_2 при $k_1 < k_2 < s+2$. Его можно доопределить:

$$M(k,k) = M(k, s+2)$$
$$M(k_2, k_1) = M(k_1, k_2)$$
$$при\ k_1 > s+2 > k_2 : M(k_1, k_2) = M(k_1, s+2)$$
$$при\ k_1 \geq k_2 > s+2 : M(k_1, k_2) = M(s+1, s+2)$$

Тогда если необходимо перераспределить блоки массива между процессами из состояния соседства по кубитам k_1 и k_2 в состояние соседства по k_1' и k_2' каждый процесс может построить массив процессов, от которых нужно получить новые блоки и которым нужно послать старые.

Тестирование

Каждый из 6 гейтов тестировалась при помощи режима «f» на 5 заранее сгенерированных файлах с n = 16 для всех количеств процессов от 1 до 16, равных степени двойки и для:

- k = 1, 7, 16 для однокубитных гейтов
- k_1, k_2 = (1, 7), (7, 16), (16, 1) для двухкубитных гетйов

Результаты выполнения

Результаты для Hn

Кол-во кубитов	К-во процессов	Время работы, с	Ускорение	Эффективность
25	32	1,198420	1,000000	1,000000
	64	0,637195	1,880774	0,940387
	128	0,353148	3,393535	0,848384
	256	0,203416	5,891474	0,736434
	512	0,097306	12,316031	0,769752
26	32	1,850910	1,000000	1,000000
	64	1,310330	1,412553	0,706276
	128	0,721916	2,563886	0,640971
	256	0,412755	4,484282	0,560535
	512	0,194352	9,523493	0,595218
27	32	5,066530	1,000000	1,000000
	64	2,683880	1,887763	0,943882
	128	1,479120	3,425368	0,856342
	256	0,840716	6,026447	0,753306
	512	0,294537	17,201676	1,075105

Результаты для CNOT для 1, 13

Кол-во кубитов	Кол-во нитей	Время работы, с	Ускорение	Эффективность
25	32	0,181472	1,000000	1,000000
	64	0,098975	1,833512	0,916756
	128	0,057949	3,131565	0,782891
	256	0,041495	4,373336	0,546667
	512	0,015131	11,993391	0,749587
26	32	0,363576	1,000000	1,000000
	64	0,151499	2,399857	1,199929
	128	0,116451	3,122137	0,780534
	256	0,081963	4,435844	0,554481
	512	0,029838	12,185162	0,761573

27	32	0,727053	1,000000	1,000000
	64	0,397273	1,830109	0,915055
	128	0,172485	4,215167	1,053792
	256	0,163892	4,436171	0,554521
	512	0,059175	12,286593	0,76791

Результаты для CNOT для 13, n

Кол-во кубитов	Кол-во нитей	Время работы, с	Ускорение	Эффективность
25	32	0,062223	1,000000	1,000000
	64	0,041483	1,499975	0,749987
	128	0,020744	2,999537	0,749884
	256	0,010375	5,997311	0,749664
	512	0,005191	11,987481	0,749218
26	32	0,082955	1,000000	1,000000
	64	0,082958	0,999960	0,499980
	128	0,041481	1,999812	0,499953
	256	0,020744	3,999084	0,499886
	512	0,010375	7,995740	0,499734
27	32	0,331807	1,000000	1,000000
	64	0,165904	1,999994	0,999997
	128	0,082956	3,999790	0,999948
	256	0,041481	7,999012	0,999876
	512	0,020744	15,995632	0,999727

Результаты для CNOT для n, 1

Кол-во кубитов	Кол-во нитей	Время работы, с	Ускорение	Эффективность
25	32	0,173975	1,000000	1,000000
	64	0,097076	1,792153	0,896076
	128	0,056248	3,093004	0,773251
	256	0,030344	5,733405	0,716676
	512	0,014811	11,746734	0,734171
26	32	0,352402	1,000000	1,000000
	64	0,144708	2,435263	1,217631
	128	0,112337	3,137007	0,784252
	256	0,080484	4,378529	0,547316
	512	0,029151	12,089014	0,755563
27	32	0,700861	1,000000	1,000000

	64	0,387534	1,808515	0,904257
	128	0,225328	3,110404	0,777601
	256	0,160515	4,366327	0,545791
	512	0,043418	16,142102	1,008881

Основные выводы.

Странное распределение времени работы для CNOT(13, n) можно объяснить тем, что номера этих кубитов больше s + 2, а следовательно, фактической пересылки данных не осуществляется. В остальном каких-либо значимых зависимостей эффективности от числа процессор и кубитов не обнаружено.