

Задание №2

**Параллельная программа на MPI, которая реализует
однокубитное квантовое преобразование**

Факультет: Вычислительной математики и кибернетики

Кафедра: Суперкомпьютеров и квантовой информатики

Группа: 323

Студент: Ни Юлия Авроровна

1. Формулировка задания

1. Разработать схему распределенного хранения данных и параллельный алгоритм для реализации однокубитного квантового преобразования на кластерной системе.

2. Реализовать параллельную программу на C++ с использованием OpenMP, которая выполняет однокубитное квантовое преобразование над вектором состояний длины 2^n , где n – количество кубитов, по указанному номеру кубита k . Описание однокубитного преобразования дано ниже в разделе методические рекомендации[1]. Для работы с комплексными числами возможно использование стандартной библиотеки шаблонов[2].

3. Протестировать программу на системе Polus. В качестве теста использовать преобразование Адамара по номеру кубита:

а) который соответствует Вашему номеру в списке группы плюс 1 (5)

б) 1

с) n

Начальное состояние вектора должно генерироваться случайным образом.

Заполнить таблицу и построить график зависимости ускорения параллельной программы от числа процессоров для каждого из случаев а)-с).

4. Написать отчет, который будет содержать листинг программы, а так же результаты выполнения пунктов 2-3.

2. Таблицы

Полученные значения усреднены по результатам 3 испытаний для каждого числа кубитов и процессоров.

1. Для 1-го кубита:

Количество кубитов	Количество процессоров	Время работы программы (сек)	Ускорение
20	1	0,02921610977	1
	2	0,02153412573	1,356735357
	4	0,01204061815	2,426462612
	8	0,004435705167	6,586576128
	16	0,00234373885	12,73483192
	32	0,001842395055	15,85767921
	64	0,0012689013784	23,67775387
24	1	0,4660883848	1
	2	0,4001661062	1,164737287
	4	0,219017248	2,128089861

	8	0,1977999173	5,35636289
	16	0,0928329234	7,028182834
	32	0,04384198872	12,300655793
	64	0,02588548708	22,98821398
28	1	19,54713112	1
	2	12,96302598	1,830802317
	4	6,975853714	3,802113106
	8	4,674021613	4,182079746
	16	1,4852021023	12,934888123
	32	0,97938181	16,560800987
	64	0,763517116	21,37970525

2. Для 5-го кубита:

Количество кубитов	Количество процессоров	Время работы программы (сек)	Ускорение
20	1	0,02708994961	1
	2	0,02340899	1,241606323
	4	0,01697242768	1,596115188
	8	0,00351626224	7,704189209
	16	0,02448391032	13,23848293
	32	0,001636282074	16,55579441
	64	0,0009121928219	27,21753761
24	1	0,4426021321	1
	2	0,4068756401	1,087806908
	4	0,2178967115	2,031247416
	8	0,1993340707	2,220403821
	16	0,0923828381	9,328392394
	32	0,04753039737	10,83455155
	64	0,02290700914	22,48091565
28	1	19,8791299	1
	2	11,05286101	1,943367182
	4	6,936425553	2,965904024
	8	4,140994097	5,47627929
	16	1,548934293	11,23882942
	32	0,984789935	16,89101472
	64	0,929188736	19,31533956

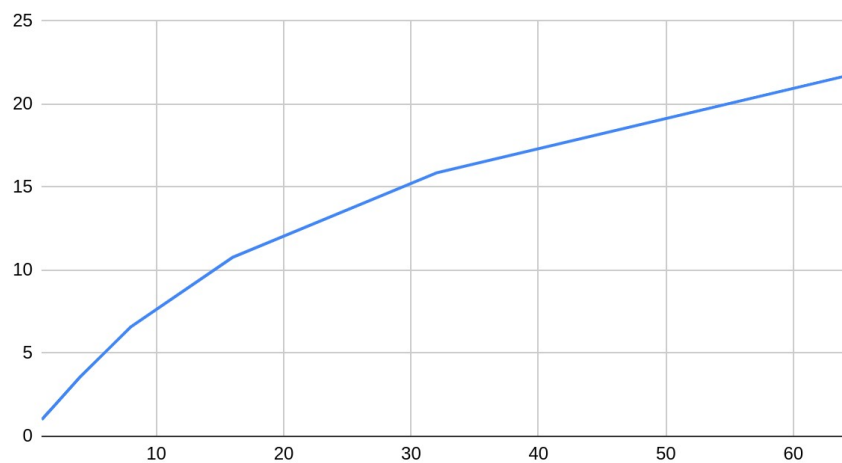
3. Для n-го кубита:

Количество кубитов	Количество процессоров	Время работы программы (сек)	Ускорение
20	1	0,02845502455	1
	2	0,02775765317	1,025123571
	4	0,01912971168	1,487477962
	8	0,004432088512	6,420229305
	16	0,003021283842	12,39192203
	32	0,001844702453	15,4252652
	64	0,0007847848514	36,25837642
24	1	0,5177485249	1
	2	0,4056208944	1,276434553
	4	0,2133630617	2,426608059
	8	0,1984751539	4,608631432
	16	0,1023948543	9,388192234
	32	0,04394179998	11,7825971
	64	0,02370973135	21,83696295
28	1	20,10482587	1
	2	12,9651554	1,860092862
	4	6,977521267	2,881370776
	8	4,046318113	5,848838251
	16	1,932838450	11,29393243
	32	0,950364572	17,934030565
	64	0,9270015508	21,68801751

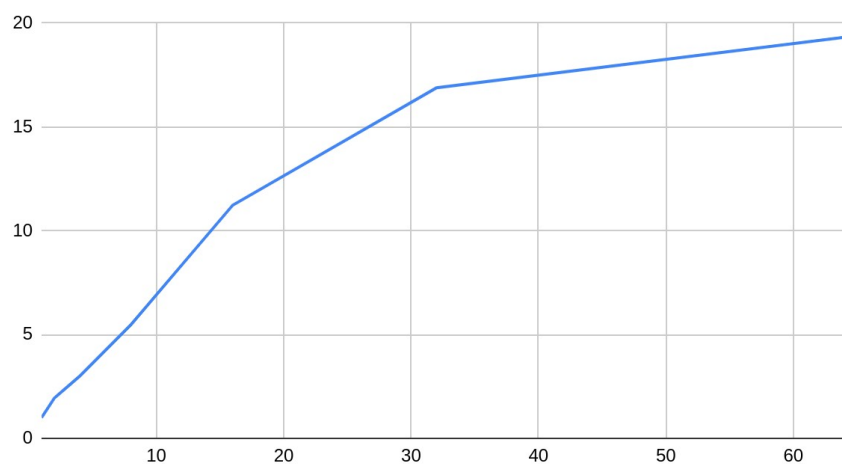
3. Графики

Ниже приведены графики ускорения для 28 кубитов.

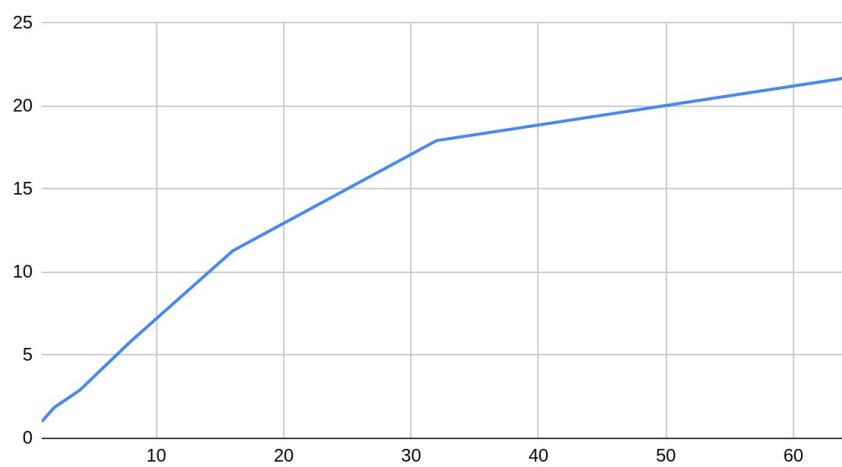
1-ый кубит



5-ый кубит



n-ый кубит



На всех графиках заметна экспоненциальная зависимость ускорения от числа потоков. Это обусловлено накладными расходами и ограниченностью эффективности распараллеливания.

Были получены схожие значения для каждой серии испытаний в трех выбранных случаях. Незначительные различия могут быть связаны с вычислительными затратами, текущей нагрузкой на систему.