

Домашнее задание №2

Описать вычислительный узел

Наименование и краткая характеристика CPU :

- Модель процессора (Model name): Intel(R) Xeon(R) Gold 6248 CPU @ 2.50GHz
- Архитектура (Architecture): x86_64
- Число логических процессоров (CPU(s)): 80.

$$\text{CPU(s)} = \text{Core(s) per socket} \times \text{Socket(s)} \times \text{Thread(s) per core}$$

- Число потоков (Thread(s) per core): 2
- Количество физических ядер в одном сокете (Core(s) per socket): 20
- Число сокетов (Socket(s)): 2
- Тактовая частота (CPU MHz):
 - Максимальная частота (CPU max MHz): 3900.0000 MHz
 - Минимальная частота (CPU min MHz): 1000.0000 MHz
- NUMA: Система поддерживает NUMA

Наименование сервера : ProLiant XL270d Gen10

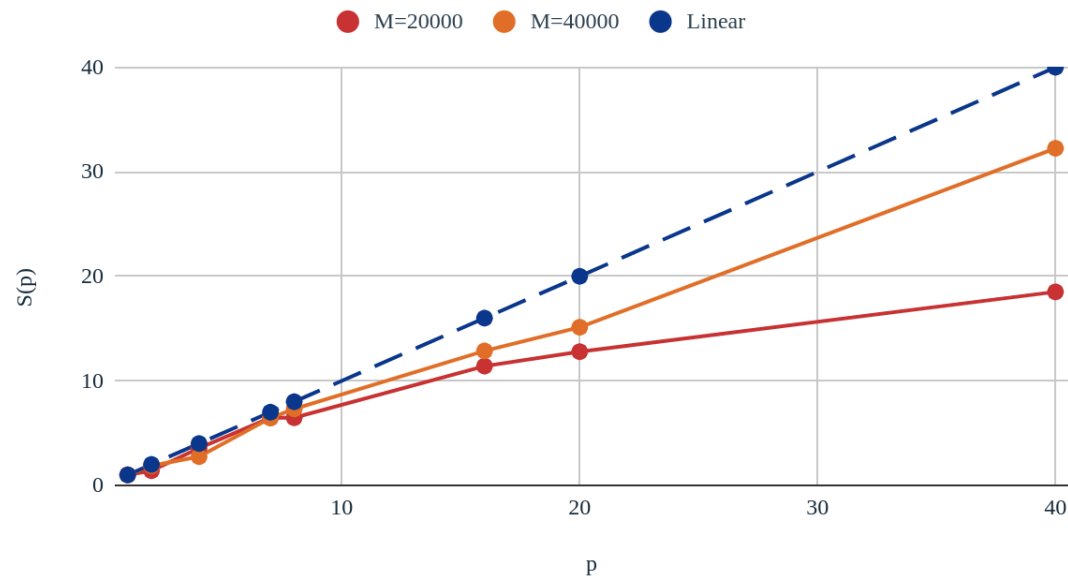
- NUMA node(s) : 2
 - node 0 size : 385636 MB
 - node 1 size : 387008 MB
- Операционная система : Ubuntu 22.04.5 LTS

Задание 1

Умножение матрицы на вектор с параллельной инициализацией массивов

M=N	Количество потоков															
	1		2		4		7		8		16		20		40	
	T(1)	S(1)	T(2)	S(2)	T(4)	S(4)	T(7)	S(7)	T(8)	S(8)	T(16)	S(16)	T(20)	S(20)	T(40)	S(40)
20000 (~3GiB)	1,2697	1	0,9073	1,3994	0,3573	3,5536	0,1966	6,4583	0,1964	6,4649	0,1113	11,408	0,0993	12,787	0,0686	18,509
40000 (~12GiB)	5,0632	1	2,7016	1,8741	1,8435	2,7465	0,786	6,4417	0,6931	7,3052	0,3937	12,861	0,3347	15,128	0,157	32,250

Ускорение в зависимости от количества потоков



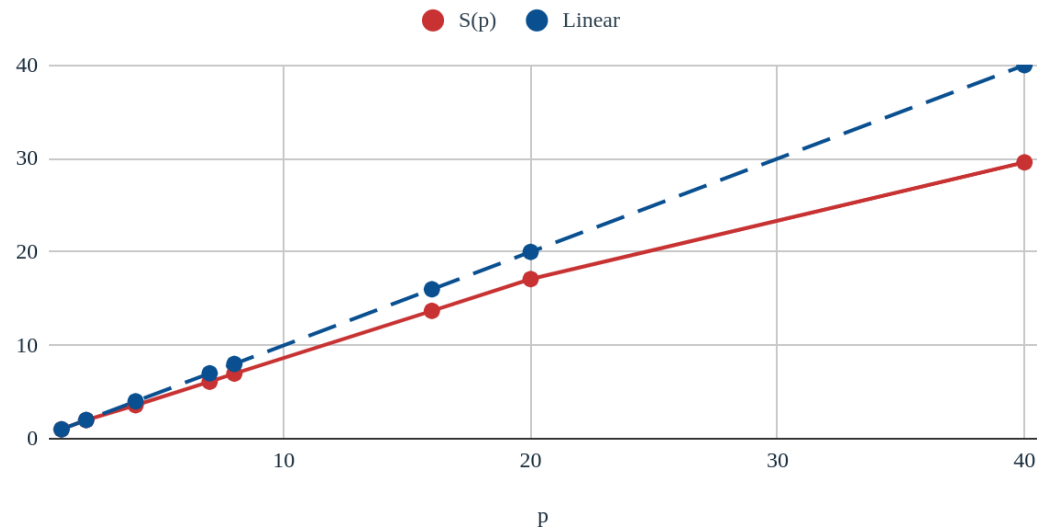
Ускорение увеличивается с количеством потоков, параллельная обработка эффективно сокращает время вычислений. Для обоих наборов данных (малого и большого размера) ускорение увеличивается с увеличением числа потоков. Ускорение не является линейным, что указывает на наличие накладных расходов, связанных с параллельным выполнением. Для большего размера массива ускорение выше, чем для меньшего, особенно при большом количестве потоков.

Задание 2

Параллельная версия программы численного интегрирования

nsteps	Количество потоков															
	1		2		4		7		8		16		20		40	
	T(1)	S(1)	T(2)	S(2)	T(4)	S(4)	T(7)	S(7)	T(8)	S(8)	T(16)	S(16)	T(20)	S(20)	T(40)	S(40)
40 000 000	0,6392	1	0,3212	1,9900	0,1784	3,5829	0,1048	6,0992	0,0917	6,9706	0,0467	13,687	0,0374	17,091	0,0216	29,592

Ускорение в зависимости от количества потоков (nsteps = 40 000 000)



Ускорение значительно увеличивается с ростом числа потоков. Например, при использовании 40 потоков ускорение достигает почти 30, что свидетельствует о значительном улучшении производительности по сравнению с однопоточным выполнением.

Система эффективно масштабируется с увеличением числа потоков, что позволяет лучше распределять вычислительные задачи и уменьшать время выполнения.

Задание 3

Параллельная реализация решения системы линейных алгебраических уравнений с помощью OpenMP

M=N=40000	Количество потоков																	
	1		2		4		7		8		16		20		40		80	
	T(1)	S(1)	T(2)	S(2)	T(4)	S(4)	T(7)	S(7)	T(8)	S(8)	T(16)	S(16)	T(20)	S(20)	T(40)	S(40)	T(80)	S(80)
static	167,95	1,000	100,16	1,677	62,634	2,681	36,561	4,594	33,512	5,012	11,507	14,59	11,118	15,11	8,2438	20,37	8,4682	19,83
auto	167,65	1,000	100,35	1,671	56,967	2,943	34,40	4,873	28,135	5,959	11,660	14,38	9,5374	17,578	10,029	16,716	9,2924	18,04
guided	163,78	1,000	87,949	1,862	42,955	3,813	23,925	6,845	23,739	6,899	15,025	10,90	11,709	13,99	7,194	22,766	7,173	22,831
dynamic	172,31	1,000	128,92	1,337	120,31	1,432	112,14	1,537	97,456	1,768	65,533	2,629	57,697	2,986	38,382	4,489	38,883	4,432
#pragma omp parallel	175,05	1,000	154,28	1,135	58,361	2,999	39,648	4,415	34,65	5,051	11,766	14,88	9,925	17,638	9,597	18,24	8,714	20,09

При увеличении числа потоков до 8–16 наблюдается почти линейное ускорение.

После 16 потоков эффективность масштабирования падает.

Наиболее эффективные стратегии: static, guided, auto.

Auto стратегия показывает наилучшую масштабируемость, особенно при большом количестве потоков.

Оптимальное число потоков: **8–16**, после чего эффект от увеличения количества потоков снижается.

Использование dynamic нецелесообразно для данной задачи.

Масштабируемость ограничена — после 16–20 потоков начинается падение эффективности.

static, auto, guided, dynamic и parallel

