



Уральский
федеральный
университет

Параллельные вычисления

Введение в параллельные вычисления

Созыкин Андрей Владимирович

К.Т.Н.

Заведующий кафедрой высокопроизводительных компьютерных технологий
Институт математики и компьютерных наук

Многопоточные и параллельные вычисления

Многопоточные вычисления:

- Несколько задач, которые нужно выполнять одновременно
- Производители и потребители
- Работа программы и отслеживание действий пользователя
- Запросы к Web-серверу

Параллельные вычисления:

- Одна задача, которую нужно разбить на несколько частей для ускорения
- Умножение матриц
- Быстрое преобразование Фурье

Архитектура параллельных систем

Системы с общей разделяемой памятью:

- Мультипроцессоры

Системы с распределенной памятью:

- Мультикомпьютеры

Гибридные системы

Архитектура параллельных систем

Системы с общей разделяемой памятью:

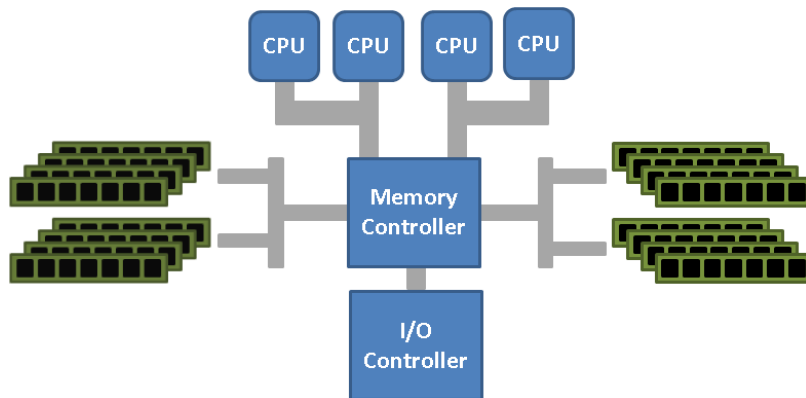
- Мультипроцессоры

Системы с распределенной памятью:

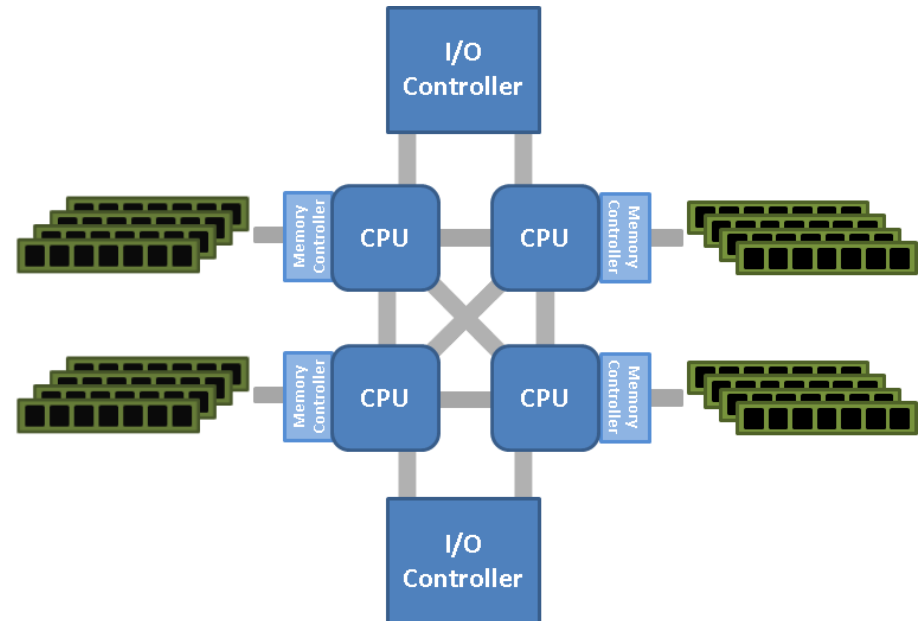
- Мультикомпьютеры

Гибридные системы

Системы с общей памятью



Симметричные
мультипроцессоры (SMP)



Неоднородный доступ к памяти
(NUMA)

Общая разделяемая память

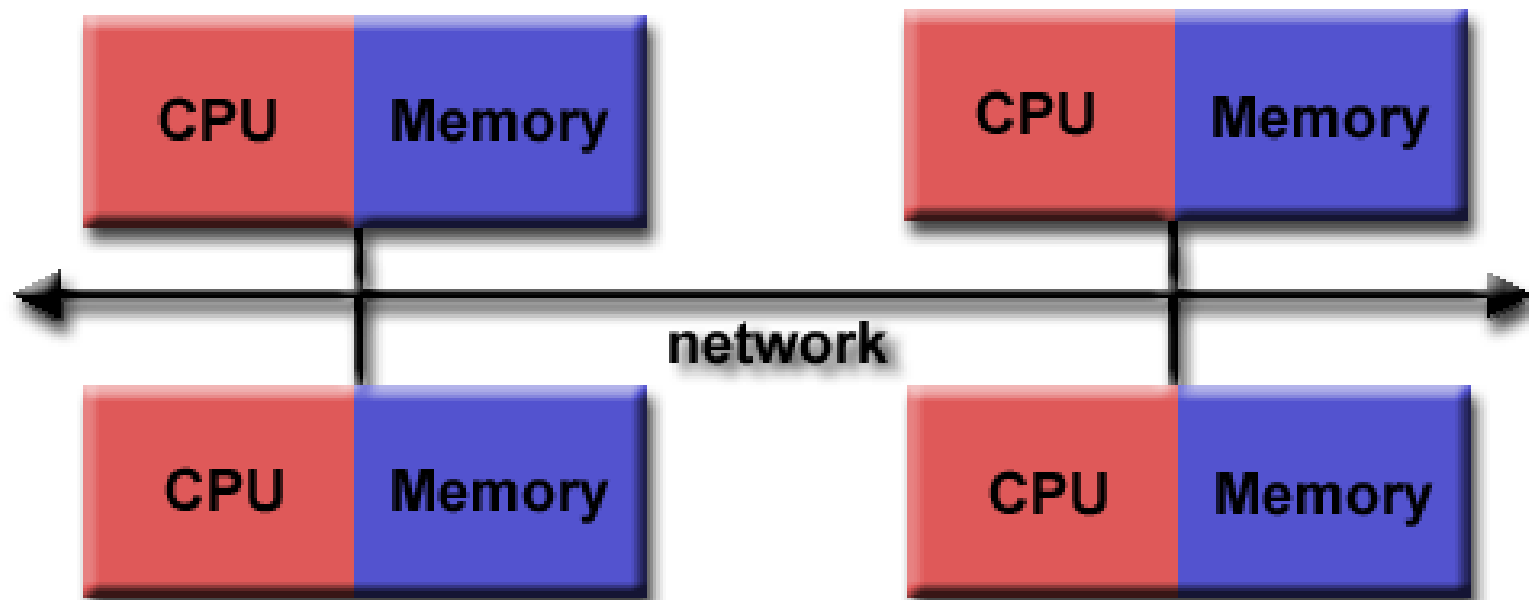
Преимущества:

- Высокая скорость доступа к данным
- Привычная модель программирования

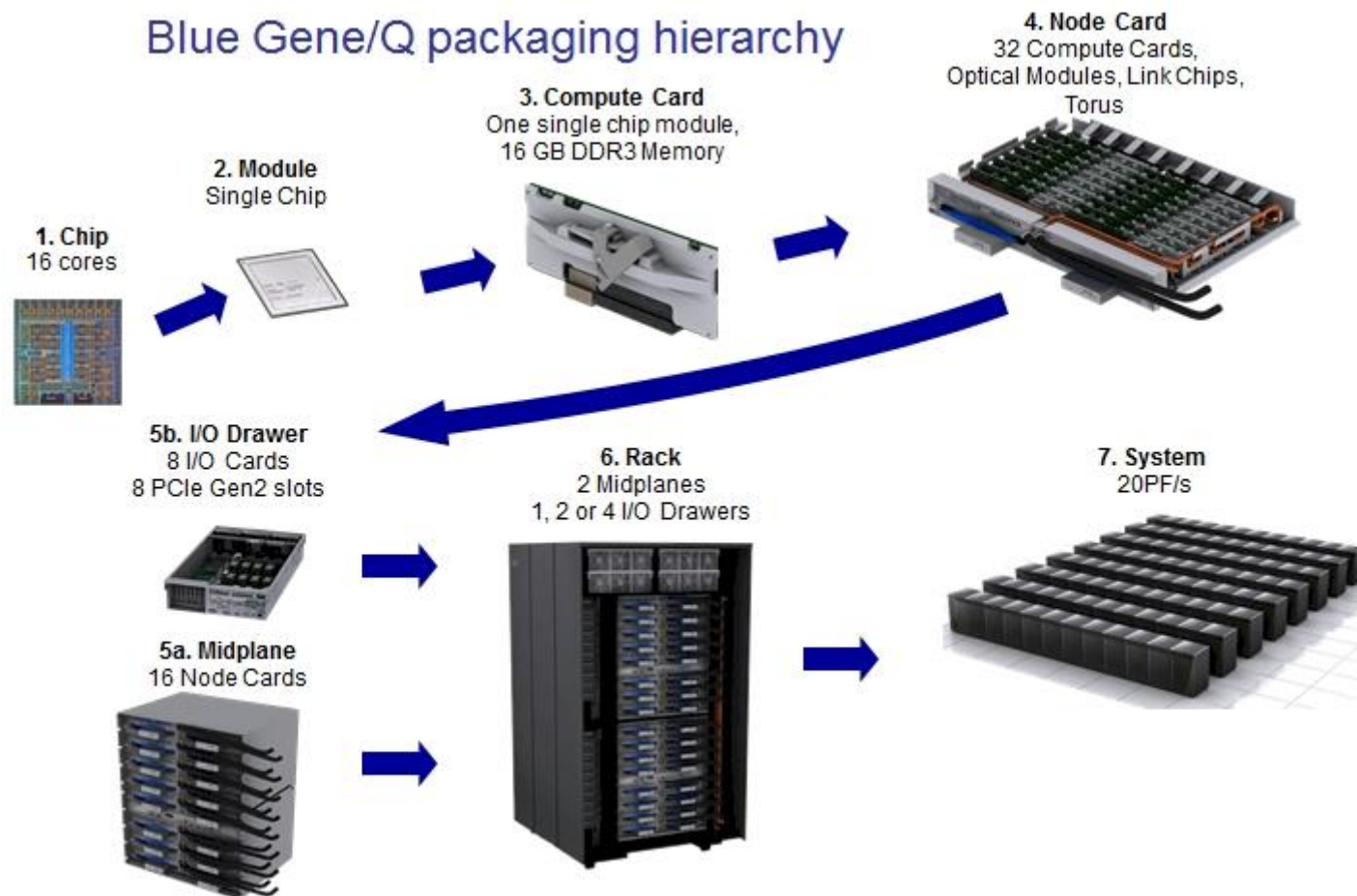
Проблемы:

- Условия гонок при работе с общими данными
- Необходимо обеспечить когерентность кэшей
- Плохая масштабируемость

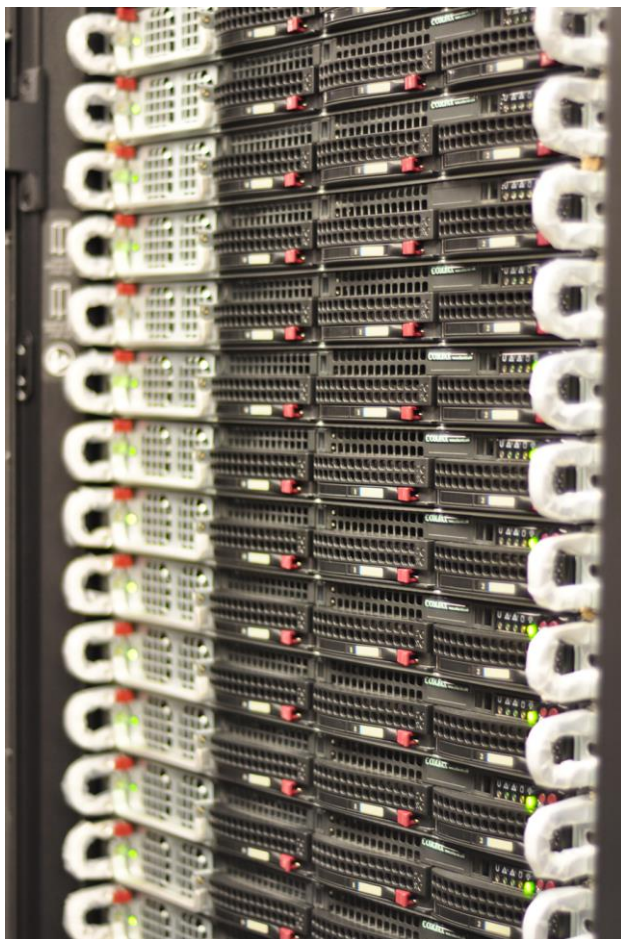
Системы с распределенной памятью



Массивно-параллельные системы (MPP)



Вычислительные кластеры



Кластер из рабочих станций (CoW)



Распределенная память

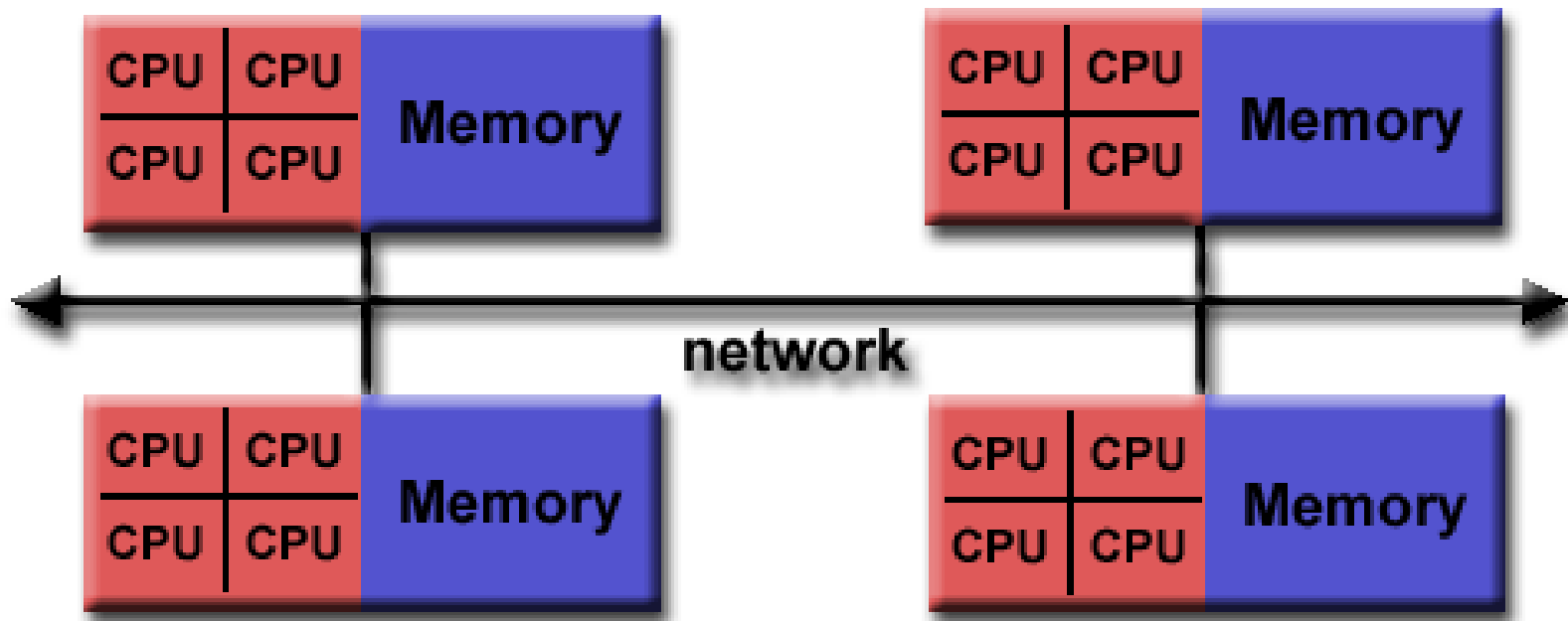
Преимущества:

- Высокая масштабируемость
- Нет условий гонок
- Не нужно обеспечивать когерентность кэшей

Проблемы:

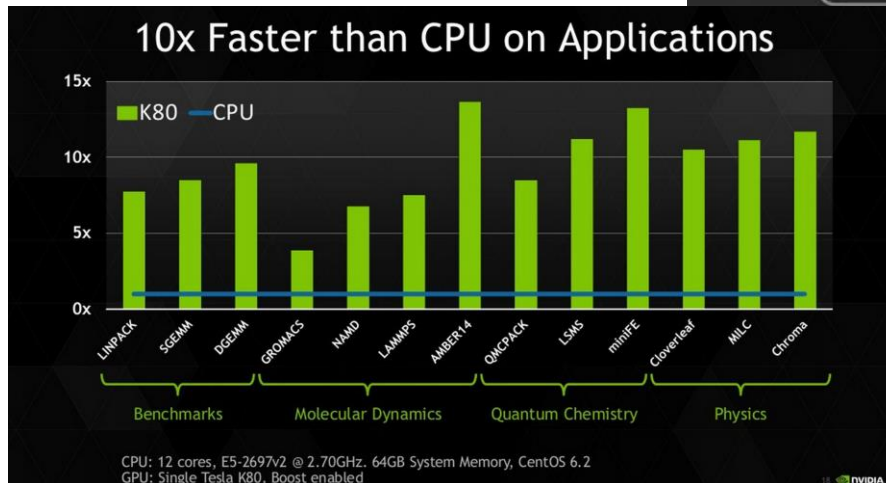
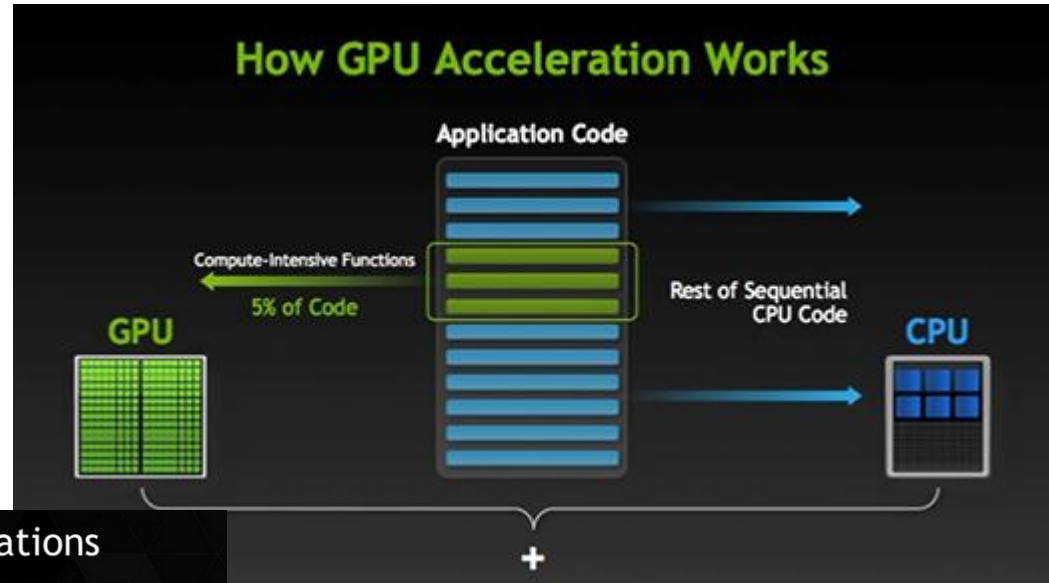
- Нет возможности напрямую обращаться к памяти другого узла
- Модель программирования на основе передачи сообщений
- Высокие задержки при передаче данных через сеть

Гибридные системы

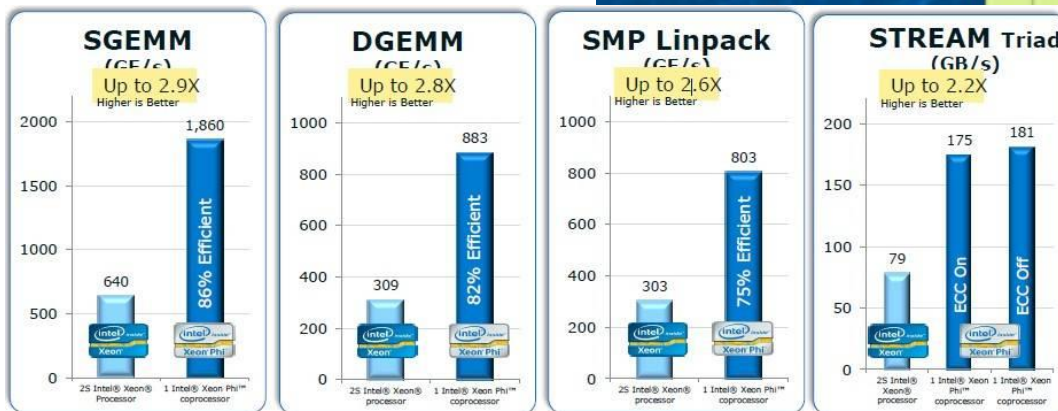
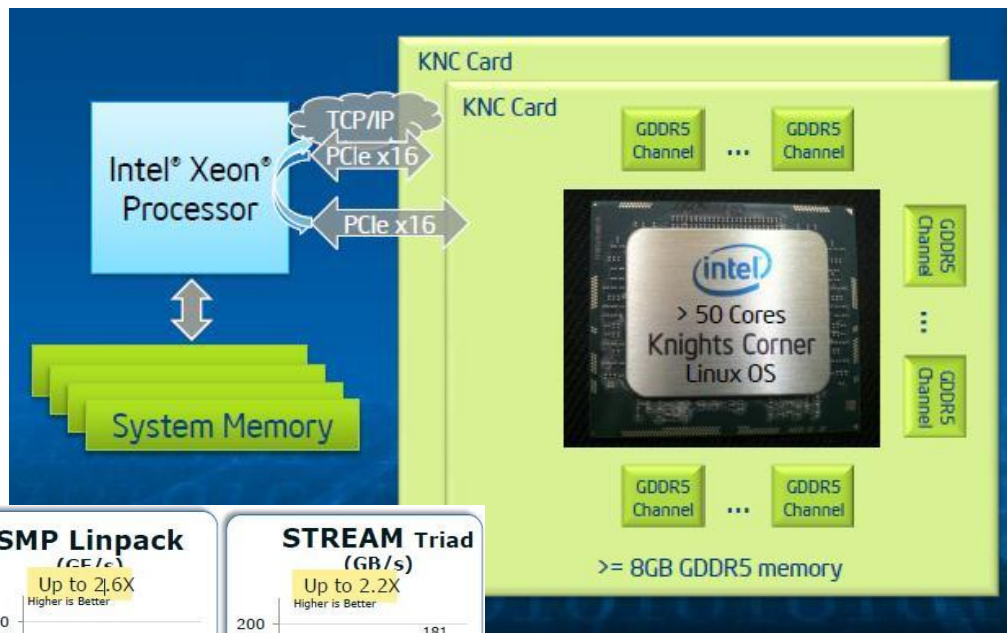


https://computing.llnl.gov/tutorials/parallel_comp/

Ускорители вычислений



Ускорители вычислений: Xeon Phi



TOP500 суперкомпьютеров мира

www.top500.org

RANK	SITE	SYSTEM	CORES	RMAX (TFLOP/S)	RPEAK (TFLOP/S)	POWER (KW)
1	National Super Computer Center in Guangzhou China	Tianhe-2 (MilkyWay-2) - TH-IVB-FEP Cluster, Intel Xeon E5-2692 12C 2.200GHz, TH Express-2, Intel Xeon Phi 31S1P NUDT	3,120,000	33,862.7	54,902.4	17,808
2	DOE/SC/Oak Ridge National Laboratory United States	Titan - Cray XK7 , Opteron 6274 16C 2.200GHz, Cray Gemini interconnect, NVIDIA K20x Cray Inc.	560,640	17,590.0	27,112.5	8,209
3	DOE/NNSA/LLNL United States	Sequoia - BlueGene/Q, Power BQC 16C 1.60 GHz, Custom IBM	1,572,864	17,173.2	20,132.7	7,890
4	RIKEN Advanced Institute for Computational Science (AICS) Japan	K computer, SPARC64 VIIIfx 2.0GHz, Tofu interconnect Fujitsu	705,024	10,510.0	11,280.4	12,660
5	DOE/SC/Argonne National Laboratory United States	Mira - BlueGene/Q, Power BQC 16C 1.60GHz, Custom IBM	786,432	8,586.6	10,066.3	3,945

Измерение производительности

FLOPS:

- Floating-point operations per second
- Количество операций с плавающей точкой в секунду

Производительность:

- Пиковая – максимально возможная для процессора, на практике не достижима
- Реальная – производительность на реальных задачах (тест Linpack – перемножение матриц)

Теория параллельных вычислений

Производительность:

- Сокращение времени выполнения одной задачи (ускорение)
- Увеличение количества выполняемых за единицу времени задач (пропускная способность)
- Сокращение энергопотребления

Ускорение

$$S_p = \frac{T_1}{T_p}$$

- T_1 – время выполнение программы на одном исполнительном устройстве
- T_p – время выполнение программы на P исполнительных устройств

Эффективность

$$E_p = \frac{S_p}{P} = \frac{T_1}{P T_p}$$

- S_p – ускорение выполнения программы на P исполнительных устройств
- T_1 – время выполнения программы на одном исполнительном устройстве
- T_p – время выполнения программы на P исполнительных устройств

Ускорение

$$E = 1$$

- Идеальный случай

$$E < 1$$

- Типичный случай
- Последовательные части алгоритма
- Накладные расходы

$$E > 1$$

- Суперлинейное ускорение

Ускорение

Какое максимальное ускорение может быть достигнуто?

Закон Амдала

$$T_1 = W_{\text{ser}} + W_{\text{par}}$$

$$T_p \geq W_{\text{ser}} + W_{\text{par}} / P$$

$$S_P \leq \frac{W_{\text{ser}} + W_{\text{par}}}{W_{\text{ser}} + W_{\text{par}}/P}.$$

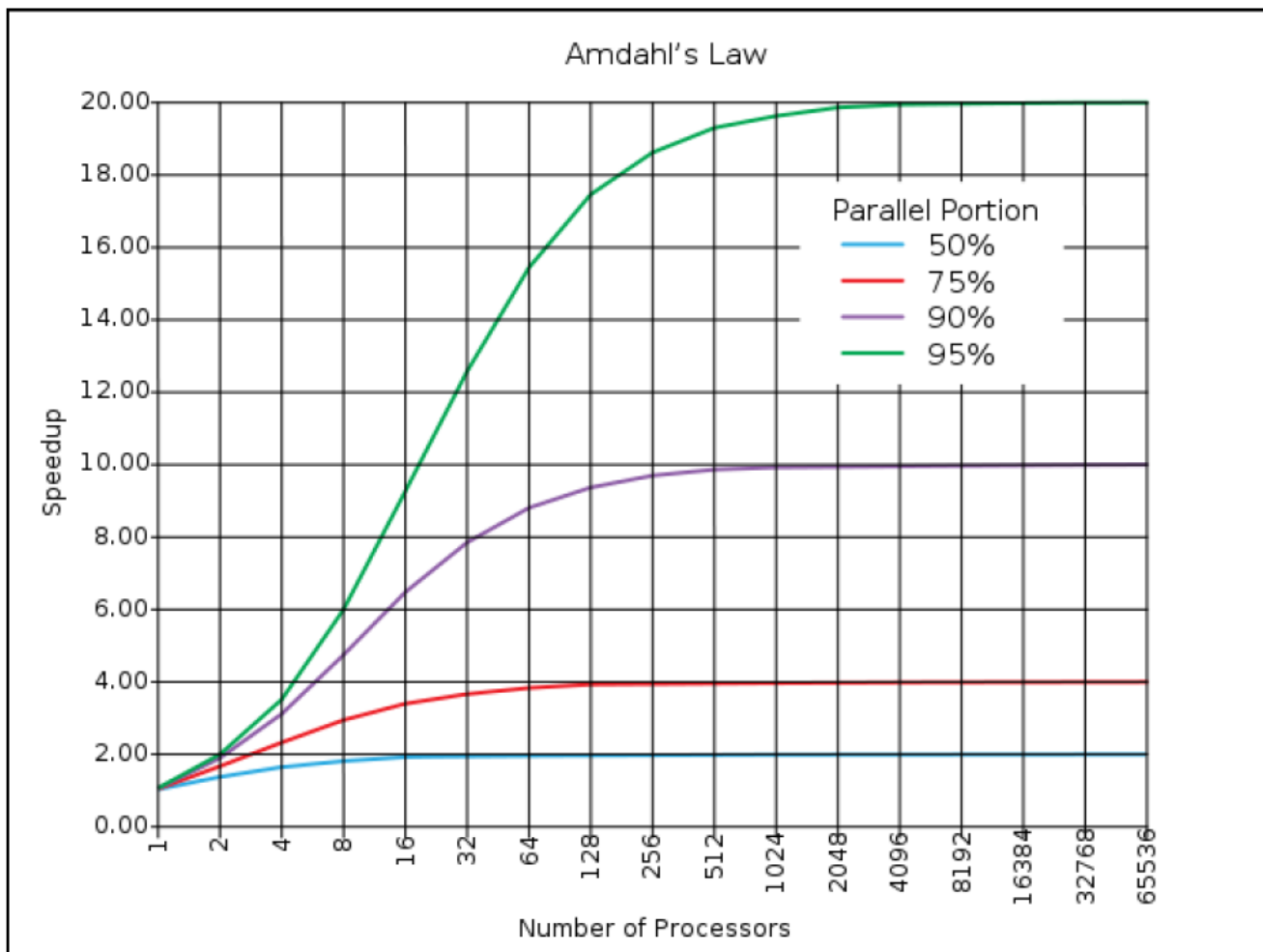
$$W_{\text{ser}} = f T_1,$$

$$W_{\text{par}} = (1 - f) T_1.$$

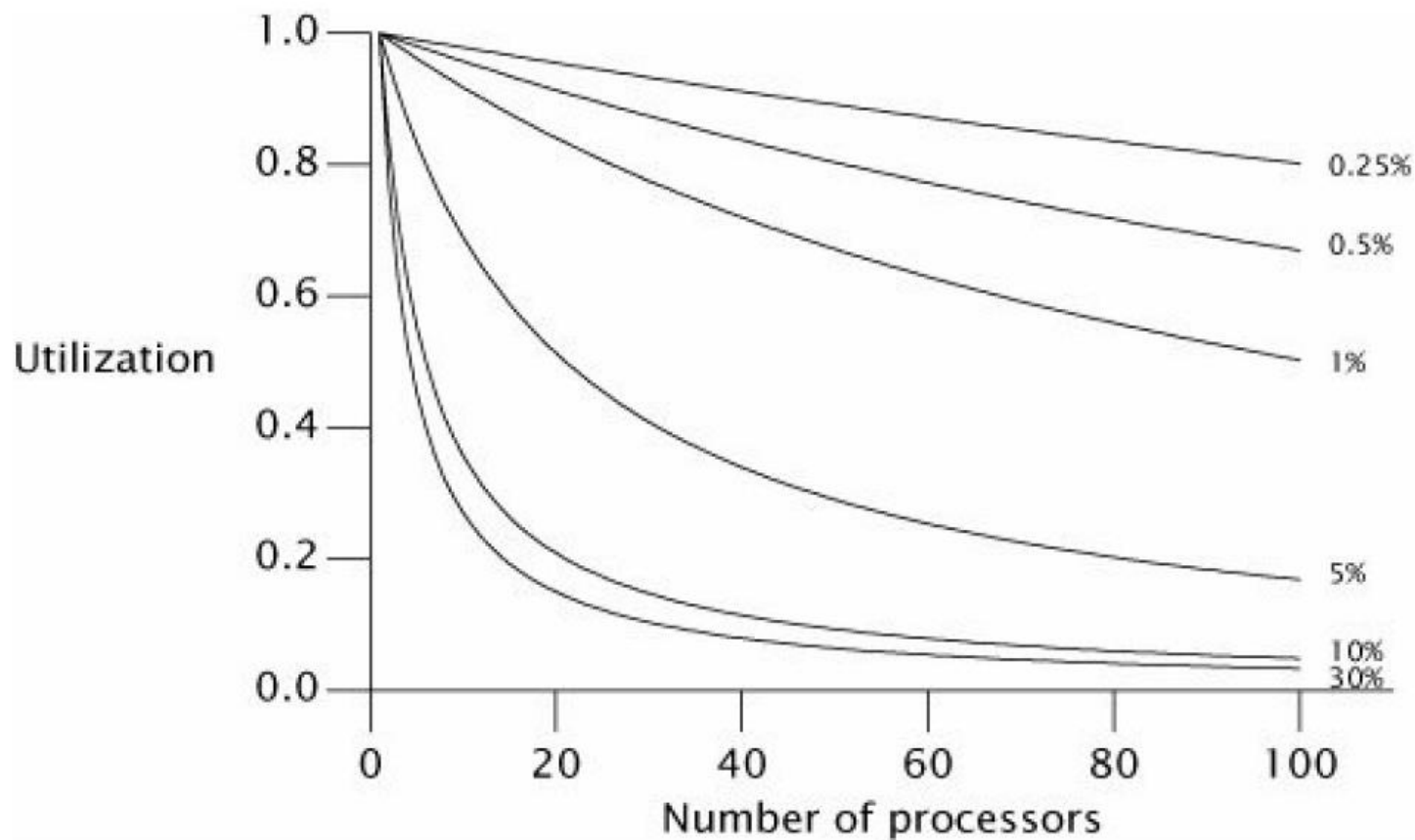
$$S_P \leq \frac{1}{f + (1 - f)/P}.$$

$$S_{\infty} \leq \frac{1}{f}.$$

Закон Амдала. Ускорение



Закон Амдала. Эффективность



Закон Густавсона-Барсиса

Джон Густавсон в Sandia National Labs получил ускорение для некоторых алгоритмов в 1000 раз

Закон Густавсона-Барсиса

- Speedup should be measured by scaling the problem to the number of processors, not by fixing the problem size.
- Ускорение нужно измерять, увеличивая объем данных с количеством процессоров, а не фиксируя размер задачи

Умножение матриц

- I/O – N^2
- Вычисления – N^3

Что будем изучать

OpenMP

- Системы с общей памятью

MPI

- Системы с распределенной памятью

Вопросы?