



ВВЕДЕНИЕ В ПАРАЛЛЕЛЬНЫЕ ВЫЧИСЛЕНИЯ

*Любая достаточно ушедшая вперед
технология неотличима от чуда.*

А. Кларк

Содержание

2

- Мотивация параллельных вычислений
- Суперкомпьютеры
- Параллельная обработка данных
- Ускорение
- Закон Амдала
- Технологии параллельного программирования

Зачем параллельные вычисления?

3

- Прогресс науки и техники
 - ▣ Новые вычислительные задачи
 - ▣ Новые аппаратные архитектуры
- Информационное общество
 - ▣ Больше вычислений – хороших и разных
 - ▣ Больше вычислений за меньшее время
- Конкурентоспособность
 - ▣ Специалист, компания, государство

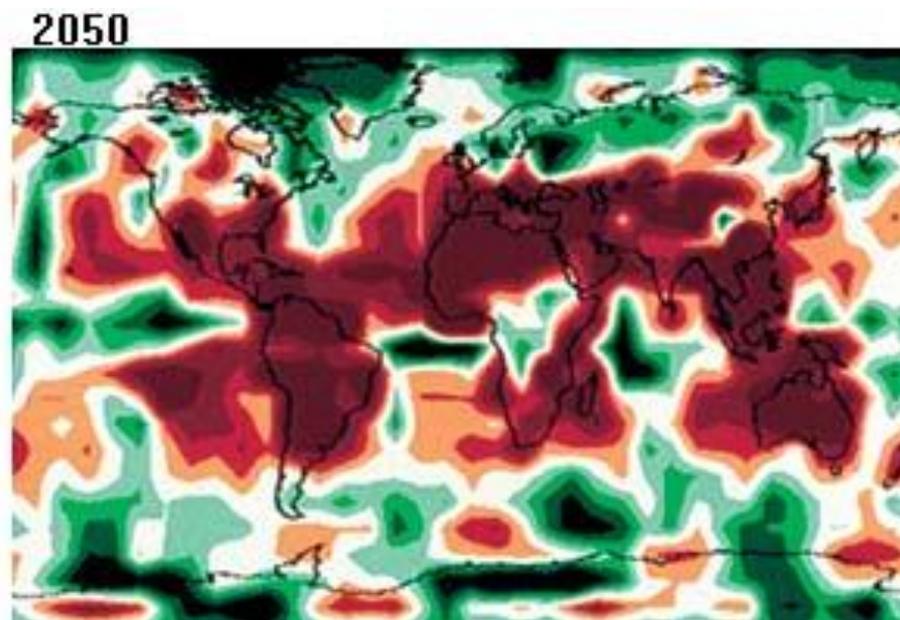
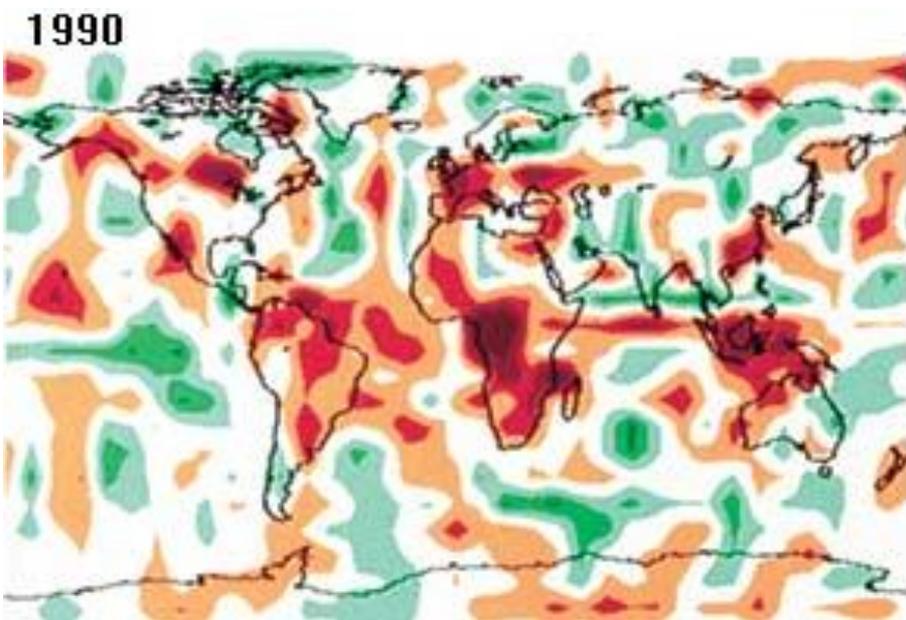
Большая задача: пример

4

- Предметная область
 - ▣ Нефтяной резервуар, пробуренные скважины для откачки нефти и закачки воды
- Задача
 - ▣ Путем моделирования определить необходимость бурения доп. скважин
- Модель
 - ▣ Куб $1000 * 10\ 000 * 10\ 000$ точек,
 - ▣ В каждой точке – от 5 до 20 функций,
 - ▣ Значение функции – решение систем нелинейных уравнений (200-1000 арифм. операций)
 - ▣ Нестационарный процесс – 100-1000 шагов по времени
- Решение
 - ▣ Количество операций: 10^{11} (точек сети) * 10(функций) * 500(операций) * 500(шагов) = $2.5 * 10^{17}$
 - ▣ Производительность персонального компьютера: $2.5 * 10^9$ операций в секунду
 - ▣ **Время решения (одной задачи): $2.5 * 10^{17} / 2.5 * 10^9 = 10^8$ сек. > 3 года**

Анализ изменений климата, прогноз погоды

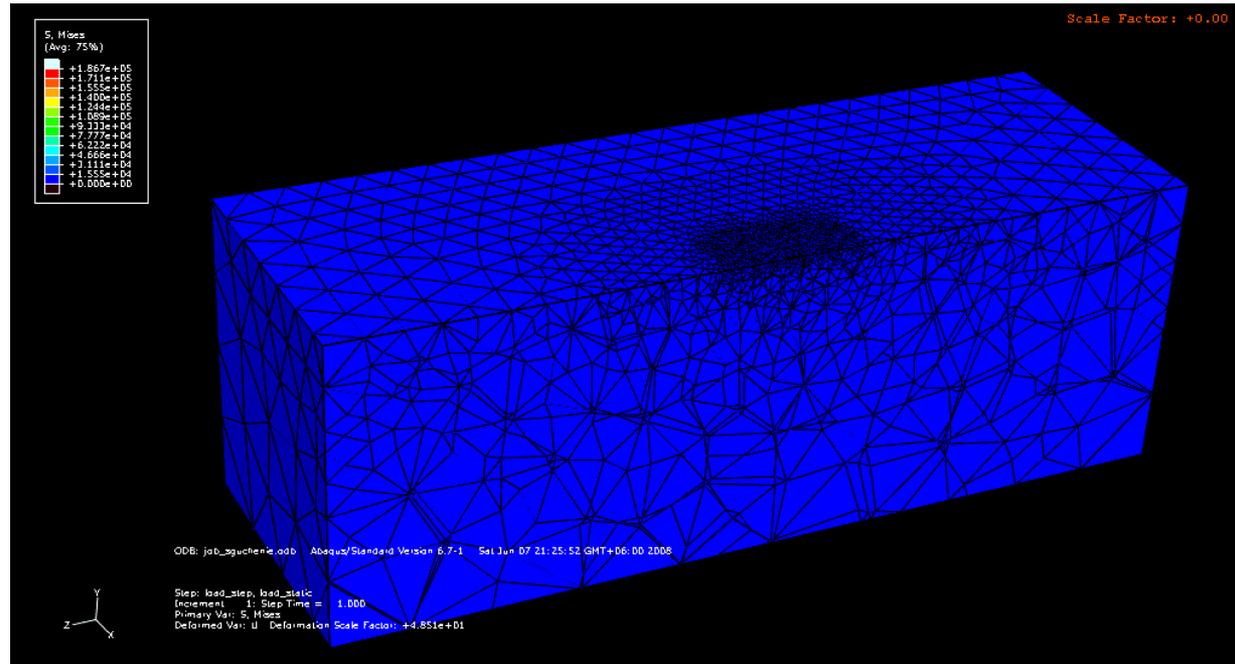
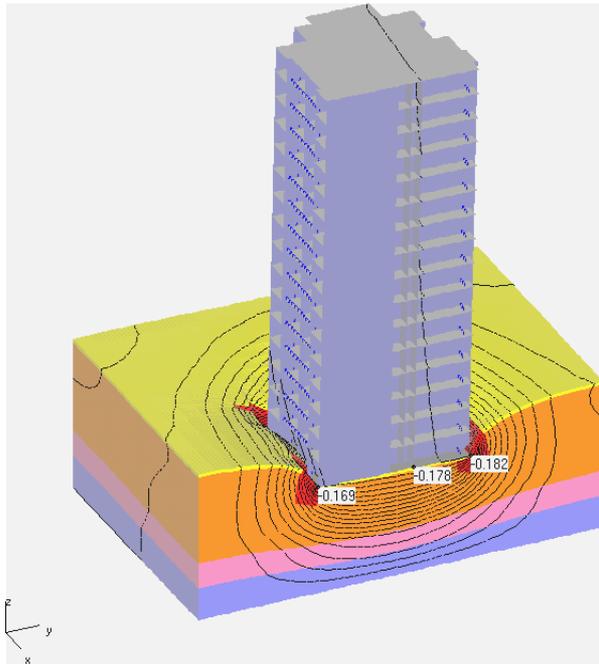
5



Прогноз роста засушливых областей, вызванных глобальным потеплением

Расчет конструкций

6



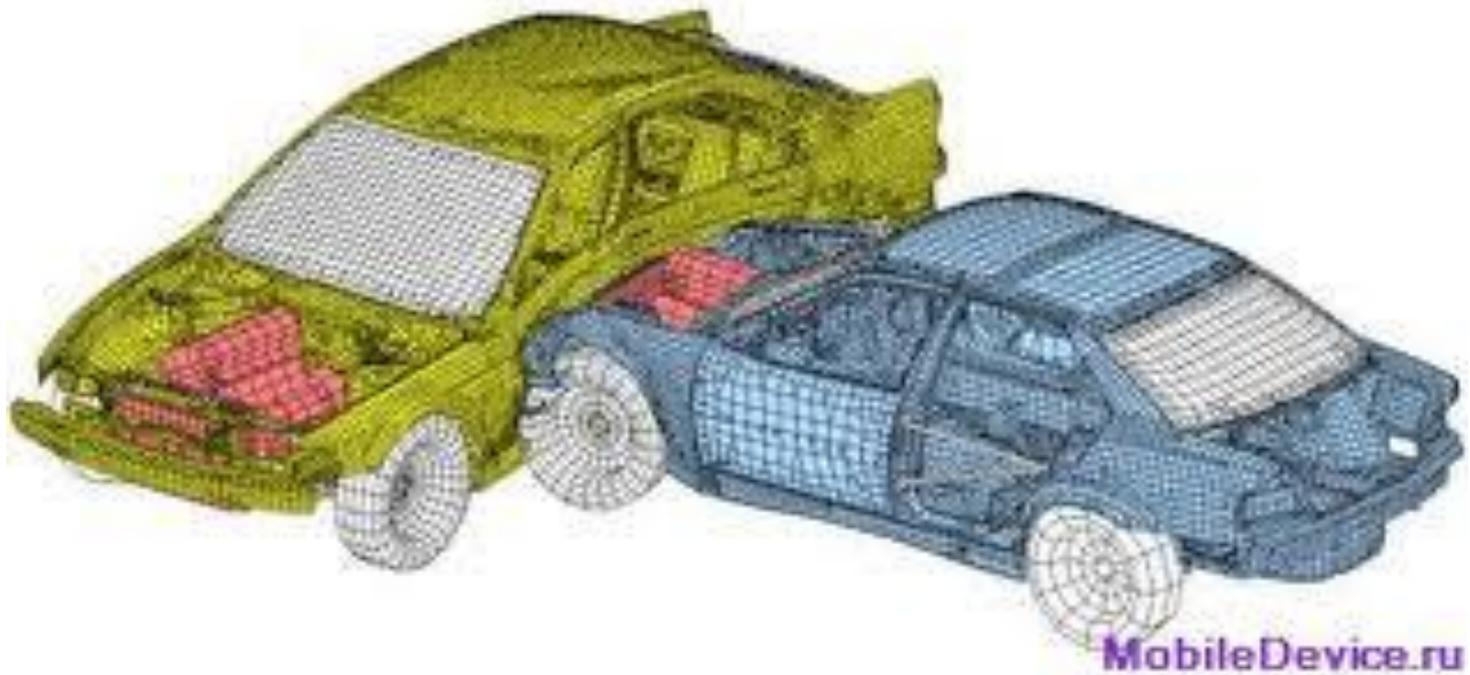
Расчет реалистичных изображений

7



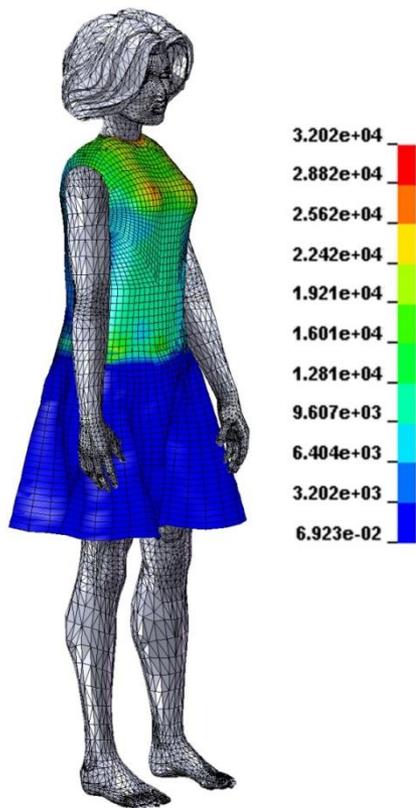
Виртуальные испытания

8

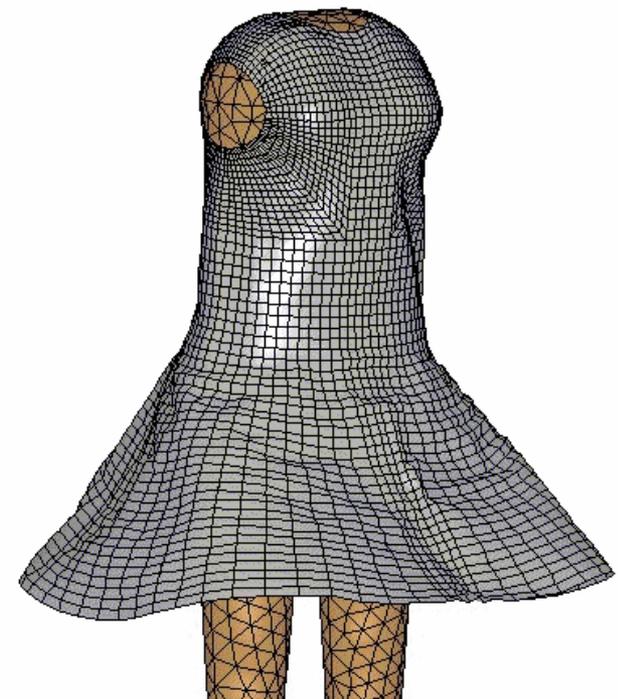


Виртуальные испытания

9



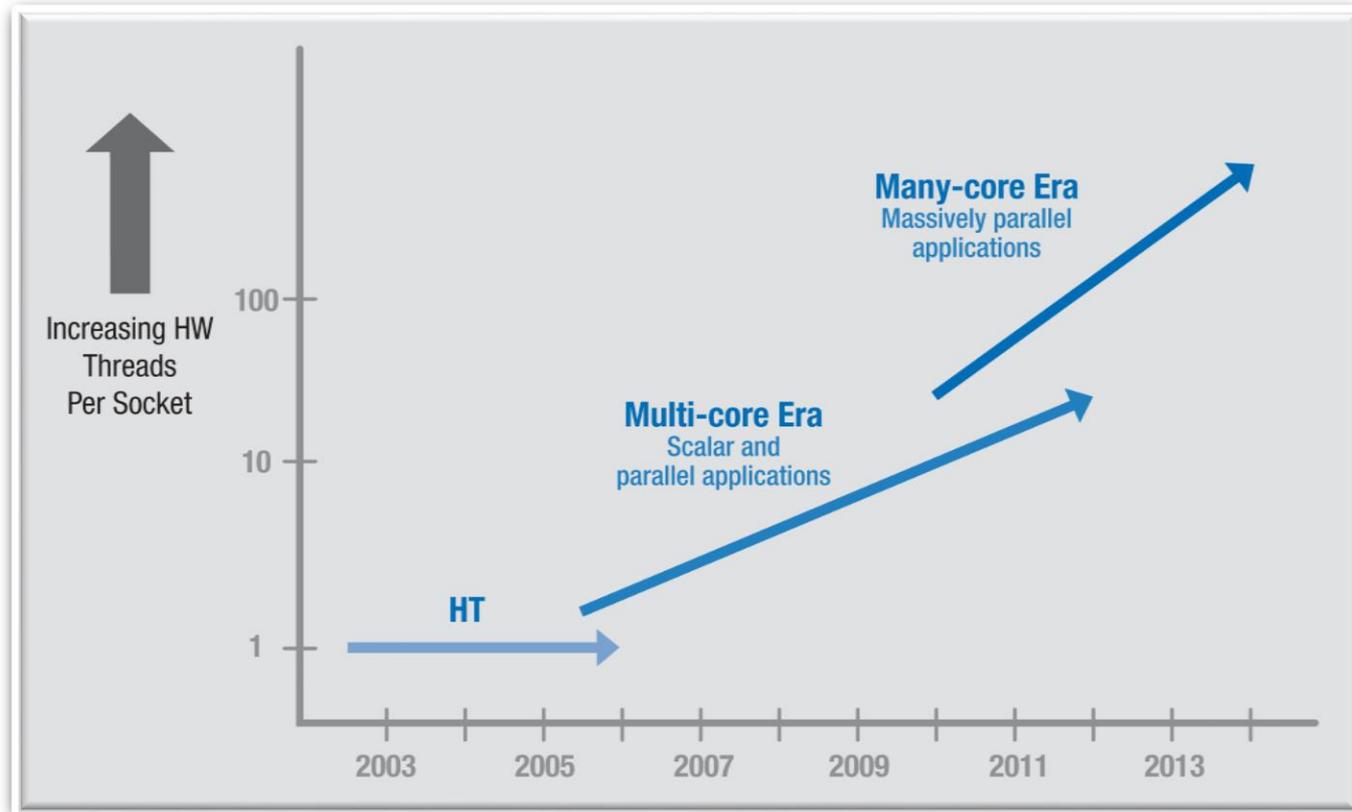
Механизм «одевания»
виртуального платья на
манекен



Деформирование
виртуального платья под
действием силы тяжести

Многоядерные процессоры

10



Platform 2015: Intel Processor and Platform Evolution for the Next Decade. White Paper. Intel Corporation, 2005.

Суперкомпьютер

11

- *Суперкомпьютер* – вычислительная система, обладающая предельными характеристиками по производительности среди компьютерных систем, имеющих в данное время.
- Другие определения ☺
 - ▣ Суперкомпьютер – любой компьютер, весящий более одной тонны.
 - ▣ Суперкомпьютер – любой компьютер, который создал Сеймур Крей.
- Рейтинг-листы суперкомпьютеров
 - ▣ Мировой: TOP500 (top500.org)
 - ▣ Российский: TOP50 (supercomputers.ru)



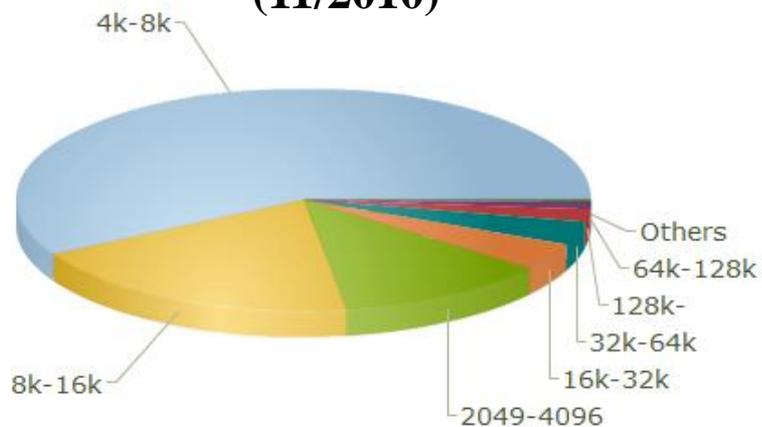
Суперкомпьютеры TOP500

12

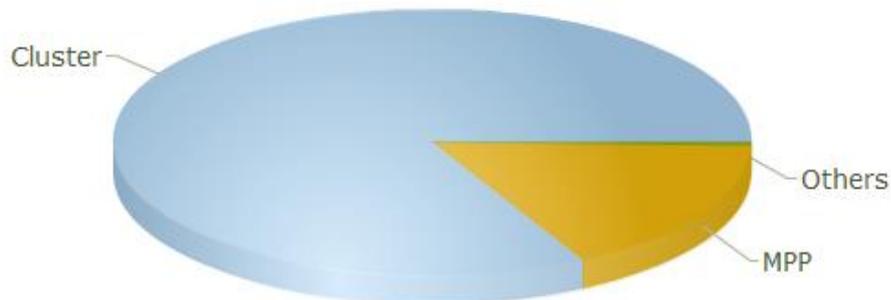
TOP 10 Systems - 11/2010

1	Tianhe-1A - NUDT TH MPP, X5670 2.93Ghz 6C, NVIDIA GPU, FT-1000 8C
2	Jaguar - Cray XT5-HE Opteron 6-core 2.6 GHz
3	Nebulae - Dawning TC3600 Blade, Intel X5650, NVidia Tesla C2050 GPU
4	TSUBAME 2.0 - HP ProLiant SL390s G7 Xeon 6C X5670, Nvidia GPU, Linux/Windows
5	Hopper - Cray XE6 12-core 2.1 GHz
6	Tera-100 - Bull bullx super-node S6010/S6030
7	Roadrunner - BladeCenter QS22/LS21 Cluster, PowerXCell 8i 3.2 Ghz / Opteron DC 1.8 GHz, Voltaire Infiniband
8	Kraken XT5 - Cray XT5-HE Opteron 6-core 2.6 GHz
9	JUGENE - Blue Gene/P Solution
10	Cielo - Cray XE6 8-core 2.4 GHz

Количество процессоров / системы (11/2010)



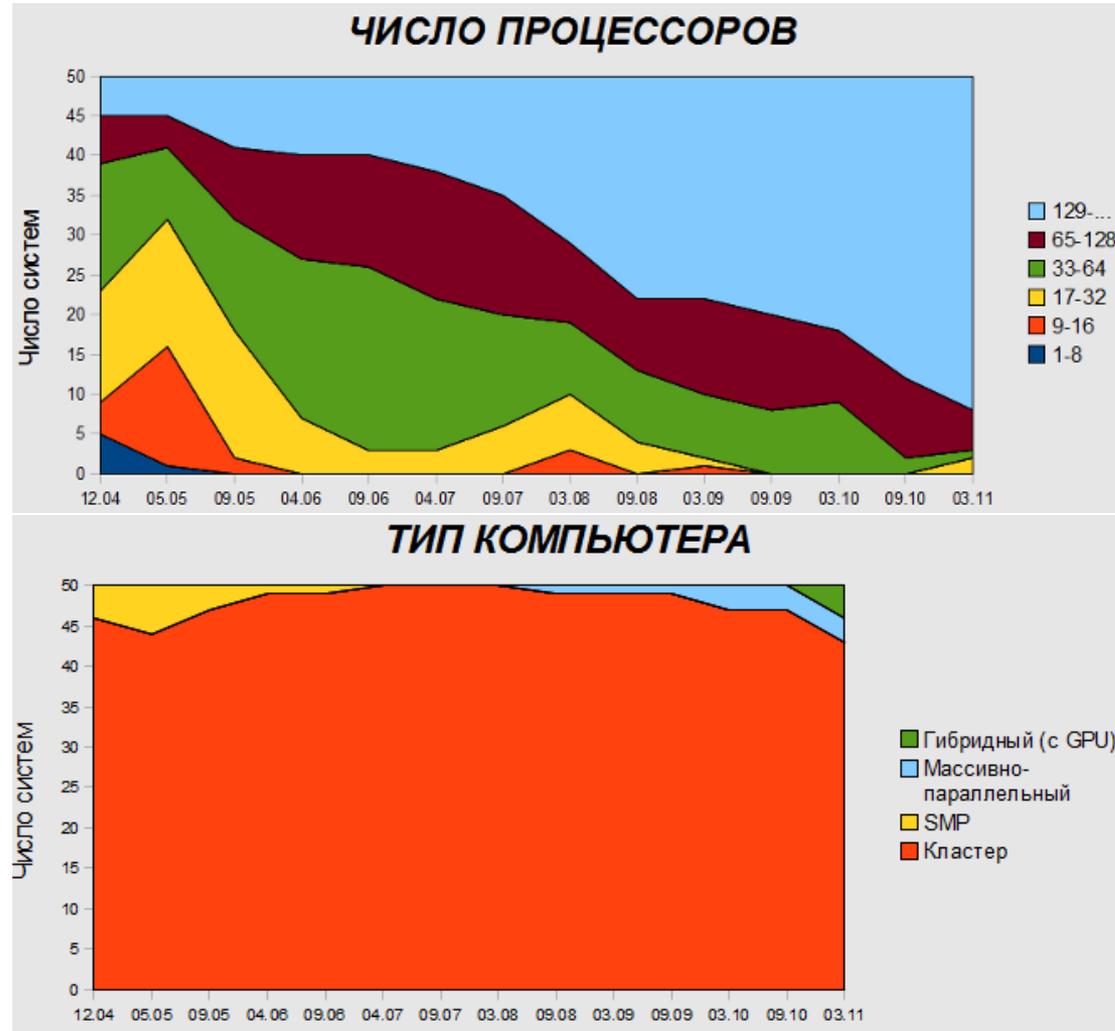
Архитектура / системы (11/2010)



Суперкомпьютеры TOP50

13

N	Место	Кол-во CPU/ядер
1	Москва Московский государственный университет имени М.В.Ломоносова 2010 г.	10260/44000
2	Москва РНИЦ Курчатовский институт 2010 г.	2576/10304
3	Челябинск Южно-Уральский государственный университет 2010 г.	1472/8832
4	Москва Межведомственный суперкомпьютерный центр Российской академия наук 2009 г.	2332/10344
5	Москва Московский государственный университет имени М.В.Ломоносова 2008 г.	1250/5000
6	Москва Government 2010 г.	1216/4864
7	Москва ИПМ им. М.В.Келдыша РАН 2010 г.	128/3456
8	Москва Government 2010 г.	1096/4384
9	Москва Banking 2009 г.	1650/6600
10	Москва Classified 2010 г.	1624/6496



Производительность суперкомпьютеров

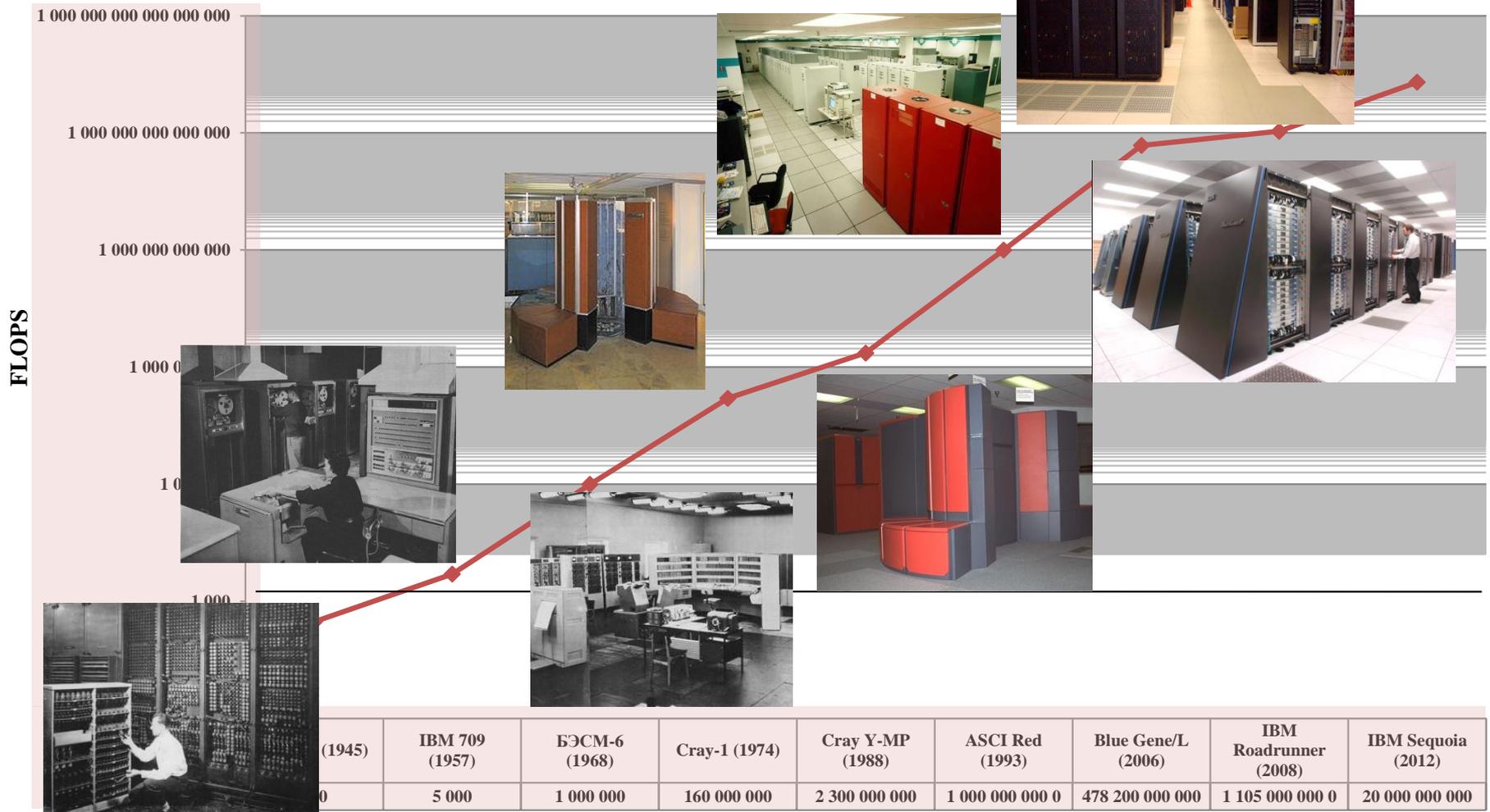
14

- *FLOPS (Floating point Operations Per Second, флонс)* – мера производительности, показывающая, сколько операций с плавающей запятой в секунду выполняет данная вычислительная система.

Название	год	FLOPS
флонс	1941	10
килофлонс	1949	10 ³
мегафлонс	1964	10 ⁶
гигафлонс	1987	10 ⁹
терафлонс	1997	10 ¹²
петафлонс	2008	10 ¹⁵
эксафлонс	≈2021	10 ¹⁸
зеттафлонс	≈2037	10 ²¹
йоттафлонс	≈2066	10 ²⁴
ксерафлонс	≈2082	10 ²⁷

Производительность суперкомпьютеров

15



Суперкомпьютеры ЮУрГУ

16

Вычислительный
кластер

Infinity

Пиковая
производительность

333 Gflops

2004 г.



Intel Xeon64
DP 3,2 ГГц

Вычислительный
кластер

СКИФ Урал

Пиковая производительность

16 Teraflops

2008 г.



Intel Xeon E5472
3 ГГц

Суперкомпьютер

СКИФ-Аврора
ЮУрГУ

Пиковая производительность

24 Teraflops

2010 г.



Intel Xeon x5570
2,93 ГГц

Суперкомпьютер

СКИФ-Аврора
ЮУрГУ

Пиковая производительность

117.6 Teraflops

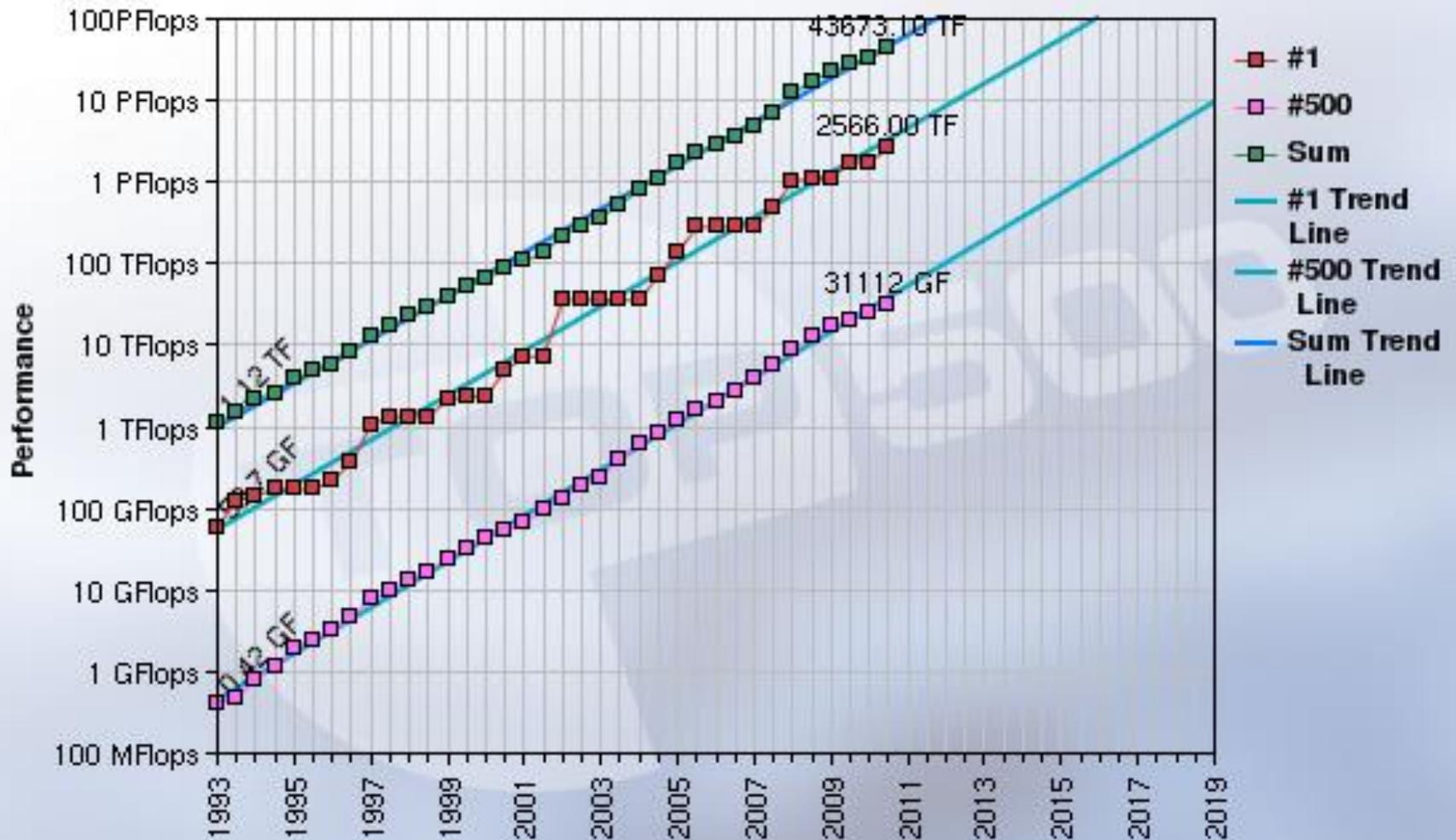
2010 г.



Intel Xeon x5680
3,33 ГГц

Производительность суперкомпьютеров

17



Персональный суперкомпьютер

18

- Персональный мини-кластер T-Edge Mini компании "Т-Платформы"
 - 4 двухпроцессорных узла на базе 4-ядерных процессоров Intel Xeon
 - Оперативная память – до 128 Гб
 - Сеть передачи данных – Gigabit Ethernet или InfiniBand
 - Пиковая производительность – 384 GFlops
 - Размеры – 57×33×76 см

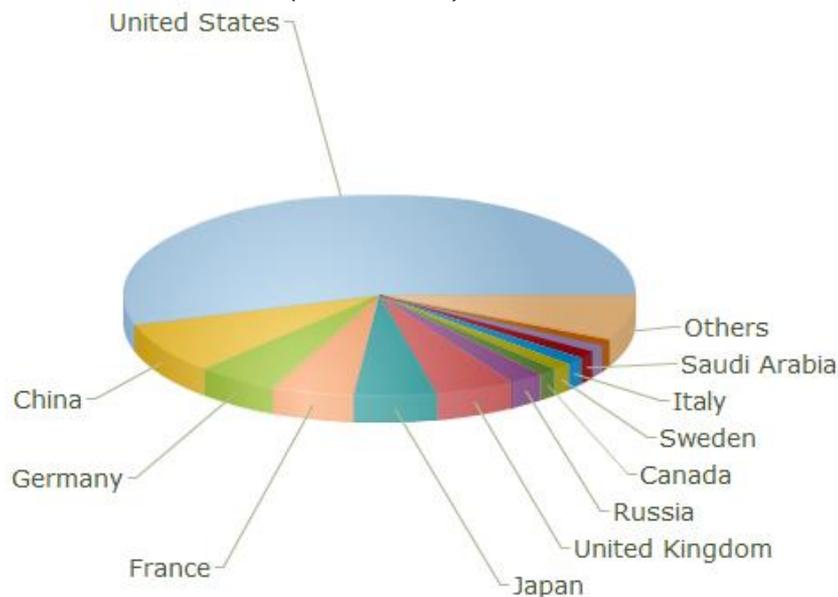


Конкурентоспособность

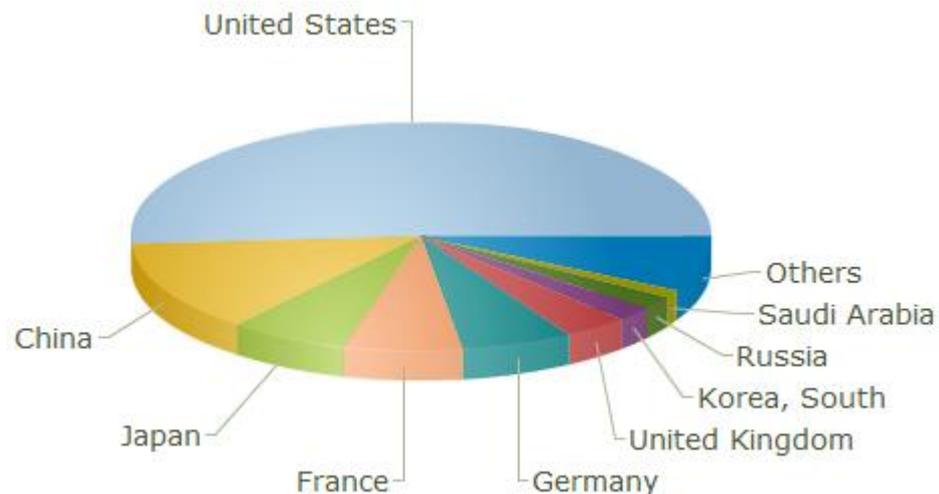
19

- Конкурентоспособность государства в современных условиях определяется уровнем развития параллельных и суперкомпьютерных вычислительных технологий.

**Страны / системы
(11/2010)**



**Страны / производительность
(11/2010)**



Режимы выполнения задач

20

□ *Последовательный*

- задачи решаются последовательно в некотором порядке.

□ *Многозадачный (псевдопараллельный)*

- для выполнения нескольких задач используется единственный процессор (*разделение времени*: В каждый момент времени может исполняться единственная задача).

□ *Параллельный*

- для выполнения нескольких задач используется несколько процессорных устройств.

Последовательная обработка

21



Псевдопараллельная обработка

22



Параллельная обработка

23



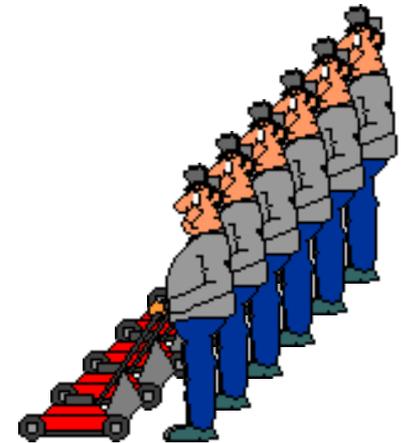
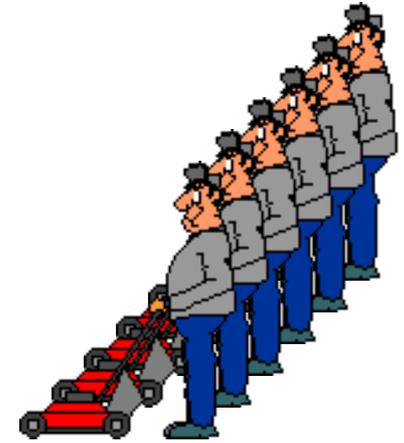
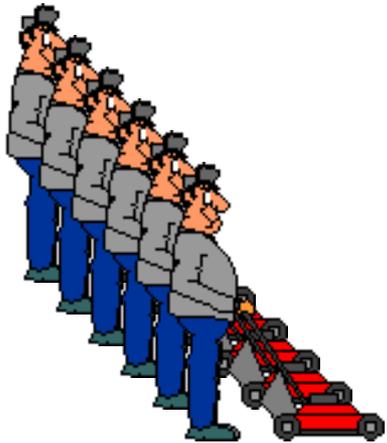
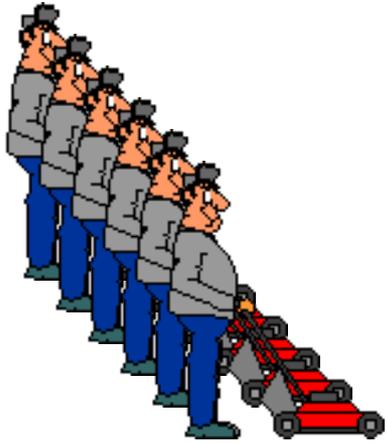
Виды параллелизма

24

- Многопроцессорная обработка
 - Несколько процессоров используются для решения одной и той же задачи
 - Задача разбивается на подзадачи
 - Каждый процессор выполняет свою подзадачу
- Конвейерная обработка
 - Несколько устройств используются для решения различных задач
 - Задача разбивается на этапы
 - Каждому устройству сопоставляется этап, который оно выполняет, и два других устройства: поставщик и потребитель
- Векторная обработка
 - Использование специального процессора, который выполняет операцию над векторами как одну команду
- Векторно-конвейерная обработка

Многопроцессорная обработка

25



Конвейерная обработка

26



Генри Форд (1863-1947)

Первый сборочный конвейер
на автозаводе Форда

Векторная обработка

28



Копировально-фрезерный станок для обработки отверстий замка SZS-100. Обеспечивает сверление 3-х параллельных отверстий для замочного паза.

Векторно-конвейерная обработка

29



Конвейер по
сборке
автомобилей на
заводе "Вольво"

Ускорение

30

□ *Ускорение (speedup)* параллельного алгоритма для p процессоров

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

□ Разновидности

▣ *Абсолютное*: T_1 – время выполнения лучшего последовательного алгоритма

▣ *Относительное*: T_1 – время выполнения параллельного алгоритма на одном процессоре

▣ *Линейное*: $S_p = p$

▣ *Сверхлинейное*: $S_p > p$

Закон Амдала

31

- Ускорение алгоритма за счет его распараллеливания ограничено долей последовательных вычислений этого алгоритма.
- Пусть f – доля последовательных вычислений в алгоритме. Тогда

$$\square S_p = \frac{T_1}{T_p} = \frac{f + (1-f)}{f + \frac{1-f}{p}} = \frac{1}{f + \frac{1-f}{p}} \quad \text{т.е.}$$

$$S_p = \frac{1}{f + \frac{1-f}{p}}$$

$$\square \lim_{p \rightarrow \infty} S_p = \frac{1}{f}$$



Джин Амдал
(р. 1922)

Закон Амдала: пример

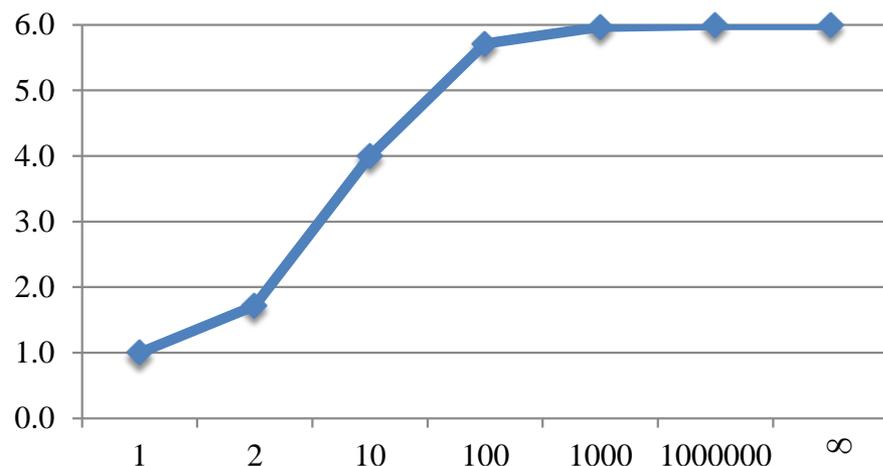
32

- Покраска забора (300 досок)
 - ▣ Подготовка – 30 мин.
НЕ распараллеливается
 - ▣ Покраска (одной доски) – 1 мин.
РАСПАРАЛЛЕЛИВАЕТСЯ
 - ▣ Уборка – 30 мин.
НЕ распараллеливается



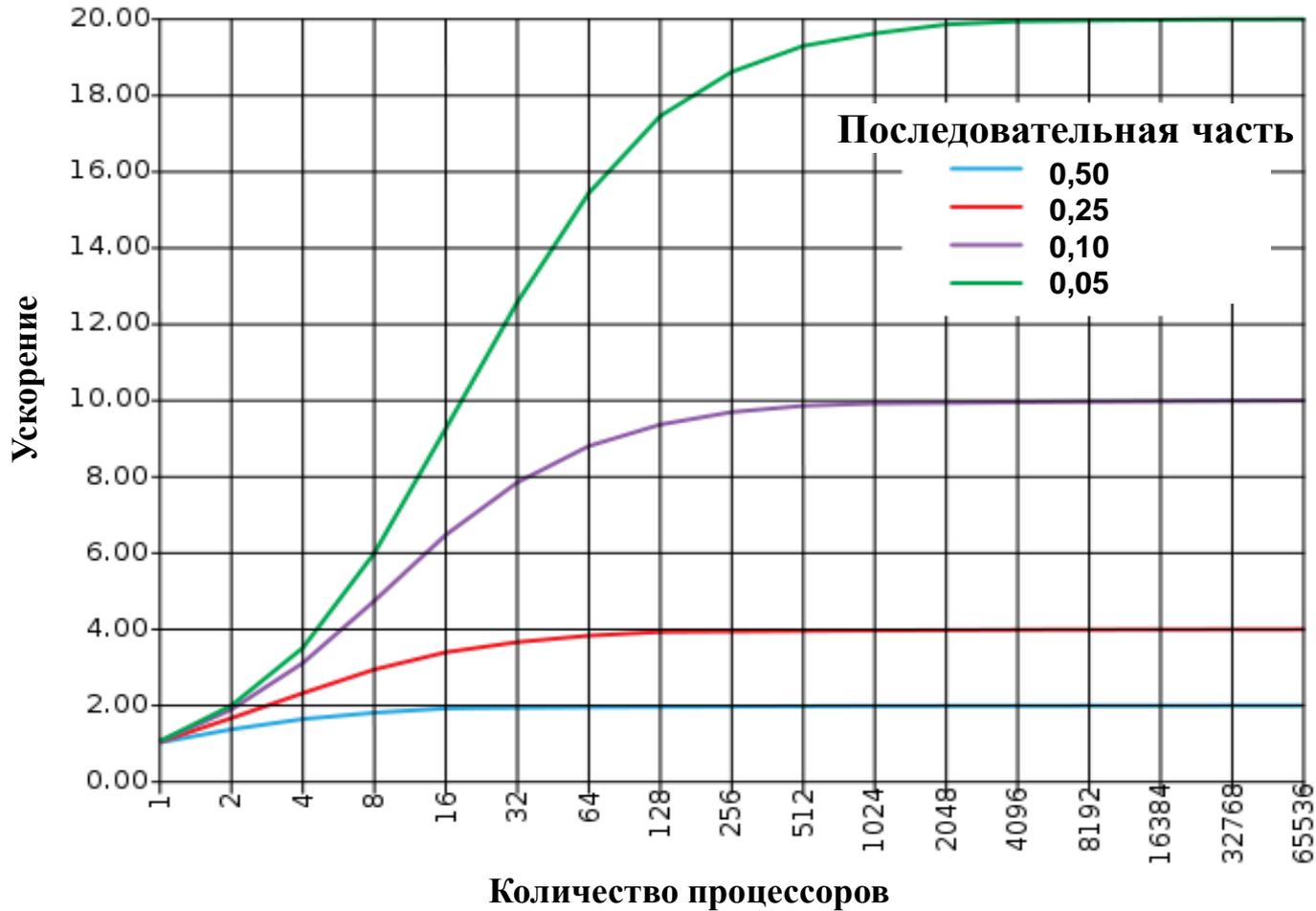
Количество маляров	Время покраски	
1	$30 + 300/1$	$+ 30 = 360$
2	$30 + 300/2$	$+ 30 = 210$
10	$30 + 300/10$	$+ 30 = 90$
100	$30 + 300/100$	$+ 30 = 63$
1000	$30 + 300/1000$	$+ 30 \cong 60$
1000000	$30 + 300/1000000$	$+ 30 \cong 60$

Ускорение



Закон Амдала

33



Закон Амдала: пример

34

$$\square S_p = \frac{1}{f + \frac{1-f}{p}} = \frac{p}{1 + f \cdot (p-1)}$$

□ Пусть имеется последовательная программа, состоящая из двух независимых частей: **A** (75%) и **B** (25%).

▣ Распараллеливание части **B** в 5 раз даст ускорение

$$\frac{5}{1 + 0.75 * (5 - 1)} = 1.25$$

▣ Распараллеливание части **A** в 2 раза даст ускорение

$$\frac{2}{1 + 0.25 * (2 - 1)} = 1.60$$



Закон Густафсона

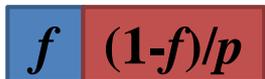
35

- Закон Амдала предполагает, что количество процессоров и доля параллельной части программы независимы, что не совсем верно.
 - Как правило, задача с фиксированным объемом данных не запускается на различном количестве процессоров (за исключением академических исследований), а объем данных изменяется в соответствии с количеством процессоров.
 - Вместо вопроса об ускорении на p процессорах рассмотрим вопрос о замедлении вычислений при переходе на один процессор.

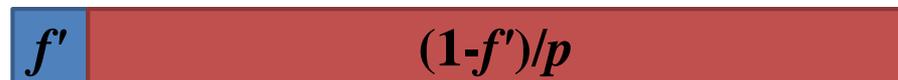


Дж. Густафсон
(р. 1955)

Запуск на последовательном процессоре Гипотетический запуск на последовательном процессоре



Запуск на параллельном процессоре



Запуск на параллельном процессоре

Закон Густафсона

36

- Закон Амдала предполагает, что количество процессоров и доля параллельной части программы независимы, что не совсем верно.
 - Как правило, задача с фиксированным объемом данных не запускается на различном количестве процессоров (за исключением академических исследований), а объем данных изменяется в соответствии с количеством процессоров.
 - Вместо вопроса об ускорении на p процессорах рассмотрим вопрос о замедлении вычислений при переходе на один процессор.

$$S_p = \frac{f + (1-f) \cdot p}{f + (1-f)} = f + p - f \cdot p = p - f \cdot (p-1)$$

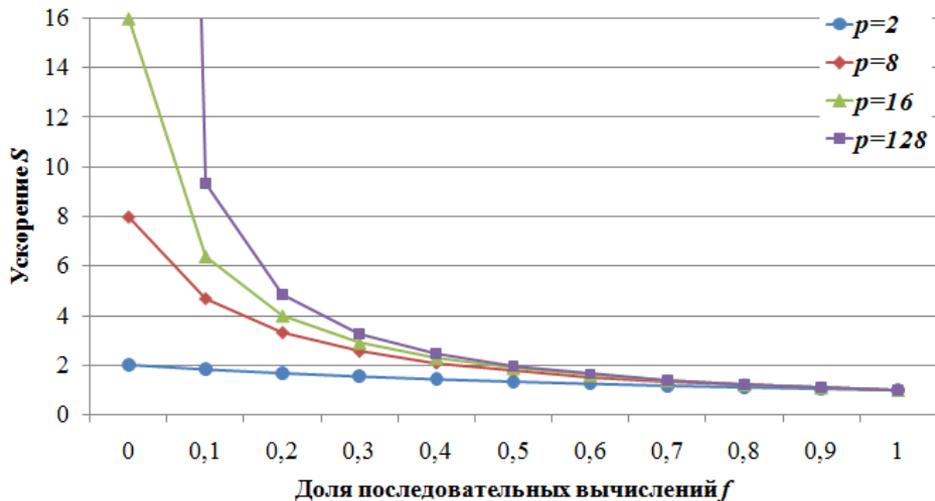
$$S_p = p - f \cdot (p-1)$$



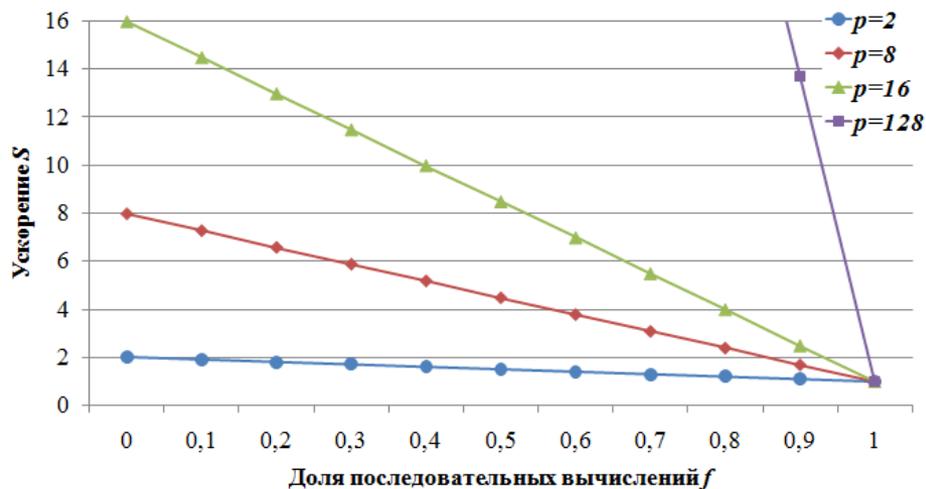
Дж. Густафсон
(р. 1955)

Законы Амдала и Густафсона

37



□ Сложность задачи постоянна



□ Сложность задачи возрастает

Параллелизм по данным

38



- Разбиение данных на непересекающиеся подмножества
- Независимое выполнение однотипных вычислений над подмножествами

Параллелизм по задачам

39



- Разбиение задачи на непересекающиеся подзадачи
- Независимое выполнение вычислений каждой подзадачи