



## ОЦЕНКА ЭФФЕКТИВНОСТИ ПАРАЛЛЕЛЬНЫХ ВЫЧИСЛЕНИЙ

В наш век передовой техники неэффективность и  
непроизводительность есть грех перед Святым Духом.  
*О. Хаксли*

Суперкомпьютеры и их применение

---

---

---

---

---

---

---

---

## Содержание

2

- Показатели эффективности параллельного алгоритма
  - Ускорение
  - Эффективность
  - Стоимость
- Оценка максимально достижимого параллелизма
  - Закон Амдала
  - Закон Густафсона
- Анализ масштабируемости параллельного алгоритма

Суперкомпьютеры и их применение © М.Л. Цымблер

---

---

---

---

---

---

---

---

## Показатели эффективности

3

- Ускорение* относительно последовательного выполнения вычислений
- Эффективность* использования процессоров
- Стоимость* вычислений

Суперкомпьютеры и их применение © М.Л. Цымблер

---

---

---

---

---

---

---

---

## Ускорение

4

- *Ускорение (speedup)*, получаемое при использовании параллельного алгоритма для  $p$  процессоров, по сравнению с последовательным вариантом выполнения вычислений:

$$S_p(n) = \frac{T_1(n)}{T_p(n)}$$

- $n$  – параметр вычислительной сложности решаемой задачи (например, количество входных данных задачи)

Суперкомпьютеры и их применение © М.Л. Цымблер

---

---

---

---

---

---

---

---

## Абсолютное и относительное ускорение

5

- Величину ускорения называют *абсолютной*, если в качестве  $T_1$  берется время выполнения наилучшего последовательного алгоритма.
- Величину ускорения называют *относительной*, если в качестве  $T_1$  берется время выполнения параллельного алгоритма на одном процессоре.

Суперкомпьютеры и их применение © М.Л. Цымблер

---

---

---

---

---

---

---

---

## Линейное и сверхлинейное ускорение

6

- *Линейное (linear)* или *идеальное (ideal)* ускорение имеет место при  $S_p = p$ .
- *Сверхлинейное (superlinear)* ускорение имеет место при  $S_p > p$ .
- Неравноправность выполнения последовательной и параллельной программ (например, недостаток оперативной памяти).
  - Нелинейный характер зависимости сложности решения задачи от объема обрабатываемых данных.
  - Различия вычислительных схем последовательного и параллельного методов.

Суперкомпьютеры и их применение © М.Л. Цымблер

---

---

---

---

---

---

---

---

## Эффективность

7

- *Эффективность (efficiency)* – средняя доля времени выполнения параллельного алгоритма, в течение которого процессоры реально используются для решения задачи.

$$\square E_p(n) = \frac{T_1(n)}{p \cdot T_p(n)} = \frac{S_p(n)}{p}$$

Суперкомпьютеры и их применение © М.Л. Цымблер

---

---

---

---

---

---

---

---

## Ускорение vs эффективность

8

- Ускорение и эффективность – 2 стороны одной медали: попытки повышения качества параллельных вычислений по одному из показателей может привести к ухудшению качества по другому показателю.

Суперкомпьютеры и их применение © М.Л. Цымблер

---

---

---

---

---

---

---

---

## Стоимость вычислений

9

- *Стоимость (cost)* параллельных вычислений

$$C_p = p \cdot T_p$$

- *Стоимостно-оптимальный (cost-optimal)* параллельный алгоритм – алгоритм, стоимость которого является пропорциональной времени выполнения наилучшего последовательного алгоритма.

Суперкомпьютеры и их применение © М.Л. Цымблер

---

---

---

---

---

---

---

---

## Можно ли достичь тах параллелизма?

10

- Получение идеальных величин  $S_p=p$  для ускорения и  $E_p=1$  для эффективности может быть обеспечено не для всех вычислительно трудоемких задач.
- Достижению максимального ускорения может препятствовать существование в выполняемых вычислениях последовательных расчетов, которые не могут быть распараллелены.

Суперкомпьютеры и их применение © М.Л. Цымблер

---

---

---

---

---

---

---

---

---

---

## Закон Амдала

11

- Задаёт связь между ожидаемым ускорением параллельных реализаций алгоритма и последовательным алгоритмом в предположении, что размер задачи остается постоянным.
- Пусть  $f$  – доля последовательных вычислений в алгоритме. Тогда



Джин Амдал (р. 1922)

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

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

Суперкомпьютеры и их применение © М.Л. Цымблер

---

---

---

---

---

---

---

---

---

---

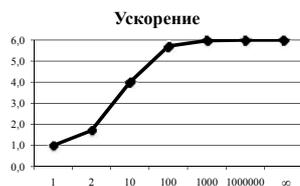
## Закон Амдала

12

- 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 ≈ 60
1000000	30 + 300/1000000	+ 30 ≈ 60



Суперкомпьютеры и их применение © М.Л. Цымблер

---

---

---

---

---

---

---

---

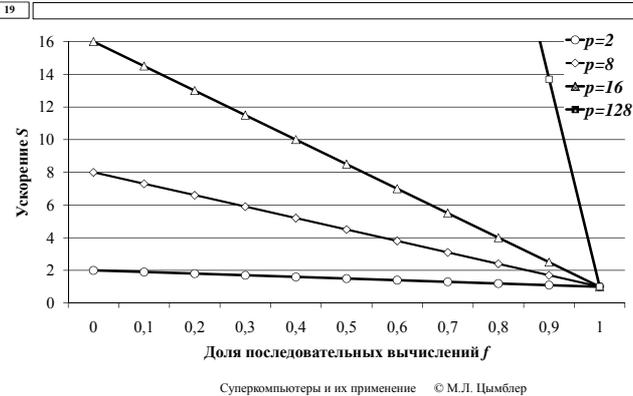
---

---





## Закон Густафсона




---

---

---

---

---

---

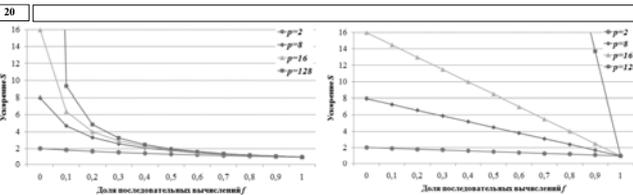
---

---

---

---

## Законы Амдала и Густафсона



- Уменьшение времени выполнения vs увеличение объема решаемой задачи
- Увеличение объема решаемой задачи приводит к увеличению доли параллельной части, т.к. последовательная часть не изменяется.

---

---

---

---

---

---

---

---

---

---

## Законы Амдала и Густафсона

21

Челябинск 60 км оз. Еловое

30 км за 1 час

- Закон Амдала
  - Независимо от того, как быстро будет ехать машина вторую половину пути, невозможно достигнуть средней скорости 90 км/ч до приезда в пункт назначения
    - Например, до приезда даже очень быстрая езда позволит достигнуть лишь средней скорости 60 км/ч.
- Закон Густавсона
  - Независимо от того, как долго или как медленно двигалась машина первую половину пути, при наличии достаточного количества времени и протяженности дороги средняя скорость машины в конечном итоге всегда достигнет значения 90 км/ч
    - Например, при скорости 150 км/ч на второй половине пути по приезде в пункт назначения средняя скорость равна 90 км/ч.

Суперкомпьютеры и их применение © М.Л. Цымблер

---

---

---

---

---

---

---

---

---

---

## Масштабируемость алгоритмов

22

- Параллельный алгоритм называют *масштабируемым (scalable)*, если при росте числа процессоров он обеспечивает увеличение ускорения при сохранении постоянного уровня эффективности использования процессоров.
- При анализе масштабируемости необходимо учитывать *накладные расходы (total overhead)*, на организацию взаимодействия процессоров, синхронизацию параллельных вычислений и др.

Суперкомпьютеры и их применение © М.Л. Цымблер

---

---

---

---

---

---

---

---

## Анализ масштабируемости

23

- Накладные расходы  $T_0 = pT_p - T_1$
- Время решения задачи  $T_p = \frac{T_1 + T_0}{p}$
- Ускорение  $S_p = \frac{T_1}{T_p} = \frac{pT_1}{T_1 + T_0}$
- Эффективность  $E_p = \frac{S_p}{p} = \frac{T_1}{T_1 + T_0} = \frac{1}{1 + \frac{T_0}{T_1}}$

Суперкомпьютеры и их применение © М.Л. Цымблер

---

---

---

---

---

---

---

---

## Анализ масштабируемости

24

- Если сложность решаемой задачи является фиксированной ( $T_1 = const$ ), то при росте числа процессоров эффективность, как правило, будет убывать за счет роста накладных расходов  $T_0$ .
- При фиксации числа процессоров эффективность использования процессоров можно улучшить путем повышения сложности решаемой задачи  $T_1$ .
- При увеличении числа процессоров в большинстве случаев можно обеспечить определенный уровень эффективности при помощи соответствующего повышения сложности решаемых задач.

Суперкомпьютеры и их применение © М.Л. Цымблер

---

---

---

---

---

---

---

---

## Анализ масштабируемости

25

- Пусть  $E=const$  – это желаемый уровень эффективности выполняемых вычислений. Тогда

$$\frac{T_0}{T_1} = \frac{1-E}{E}, \text{ или } T_1 = kT_0, \text{ где } k = \frac{E}{1-E}$$

- Данную зависимость  $n=F(p)$  между сложностью решаемой задачи и числом процессоров называют *функцией изоэффективности (isoefficiency function)*.

Суперкомпьютеры и их применение © М.Л. Цымблер

---

---

---

---

---

---

---

---

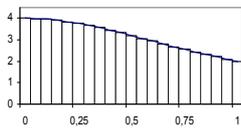
## Пример

26

- Значение числа  $\pi$  может быть получено при помощи интеграла

$$\pi = \int_0^1 \frac{4}{1+x^2} dx$$

- Численное интегрирование выполняется методом прямоугольников



Суперкомпьютеры и их применение © М.Л. Цымблер

---

---

---

---

---

---

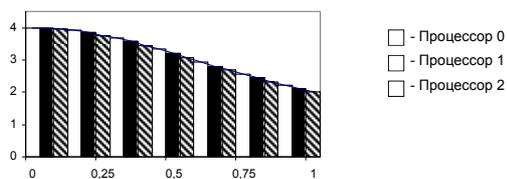
---

---

## Пример

27

- Распределяем вычисления циклически между  $p$  процессорами.
- Суммируем получаемые на отдельных процессорах частные суммы.



Суперкомпьютеры и их применение © М.Л. Цымблер

---

---

---

---

---

---

---

---

## Пример

28

- $n$  – количество разбиений отрезка  $[0;1]$
- Вычислительная сложность задачи  
 $W=T_1=6n$
- Количество узлов сетки на отдельном процессоре  
 $m=\lceil n/p \rceil \leq n/p + 1$
- Объем вычислений на отдельном процессоре  
 $W_p=6m=6n/p+6$

Суперкомпьютеры и их применение © М.Л. Цымблер

---

---

---

---

---

---

---

---

## Пример

29

- Время параллельного решения задачи  
 $T_p=6n/p+6+\log_2 p$
- Ускорение  
 $S_p=T_1/T_p=6n/(6n/p+6+\log_2 p)$
- Эффективность  
 $E_p=6n/(6n+6p+p\log_2 p)$
- Функция изоэффективности  
 $W=k(pT_p-W)=k(6p+p\log_2 p)$

Суперкомпьютеры и их применение © М.Л. Цымблер

---

---

---

---

---

---

---

---

## Заключение

30

- Показатели эффективности параллельного алгоритма
  - Ускорение
  - Эффективность
  - Стоимость
- Оценка максимально достижимого параллелизма
  - Закон Амдала
  - Закон Густафсона
- Анализ масштабируемости параллельного алгоритма

Суперкомпьютеры и их применение © М.Л. Цымблер

---

---

---

---

---

---

---

---