Лабораторная работа №4

Вычисление среднего арифметического

1. Сложность последовательной реализации

Рассмотрим алгоритм для n чисел:

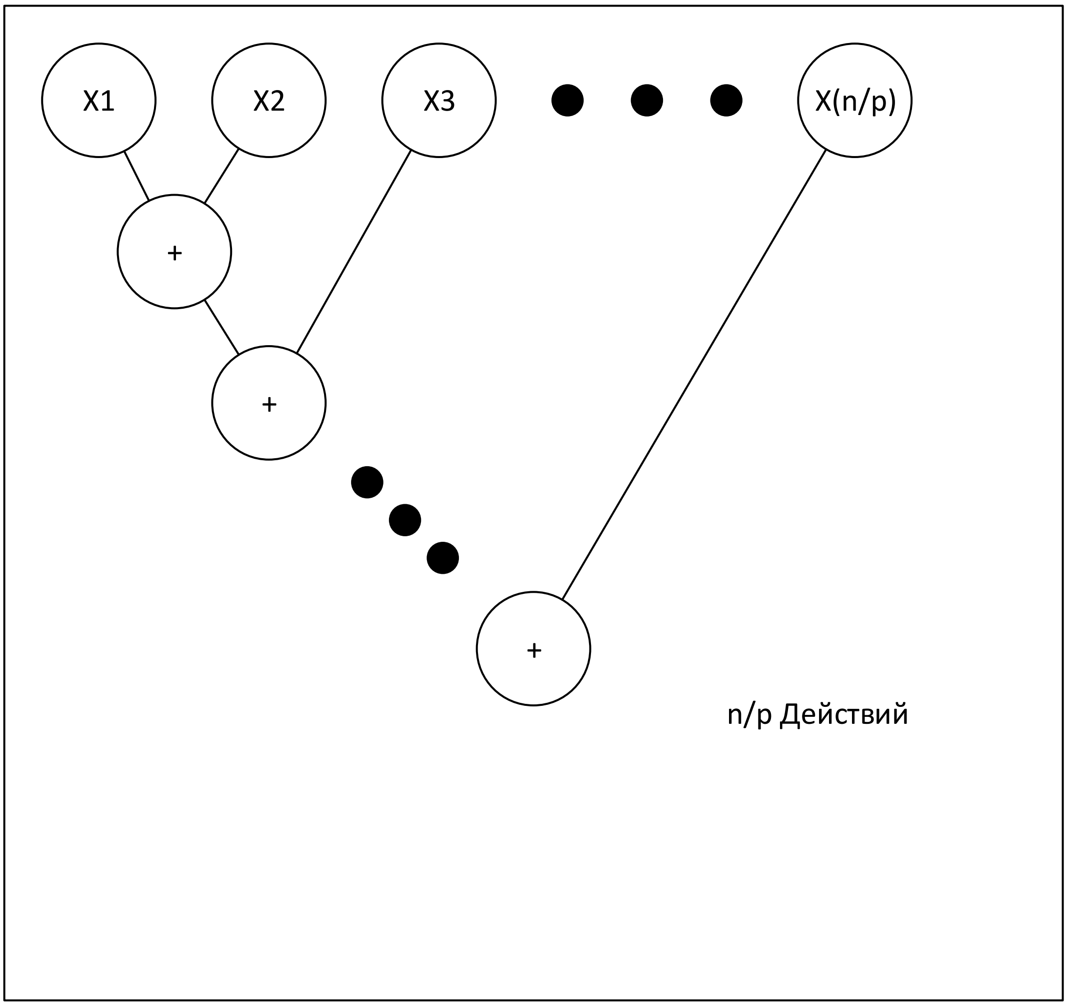
1) Сложить n чисел (n-1 итерация)
2) Разделить сумму на n (1 итерация)

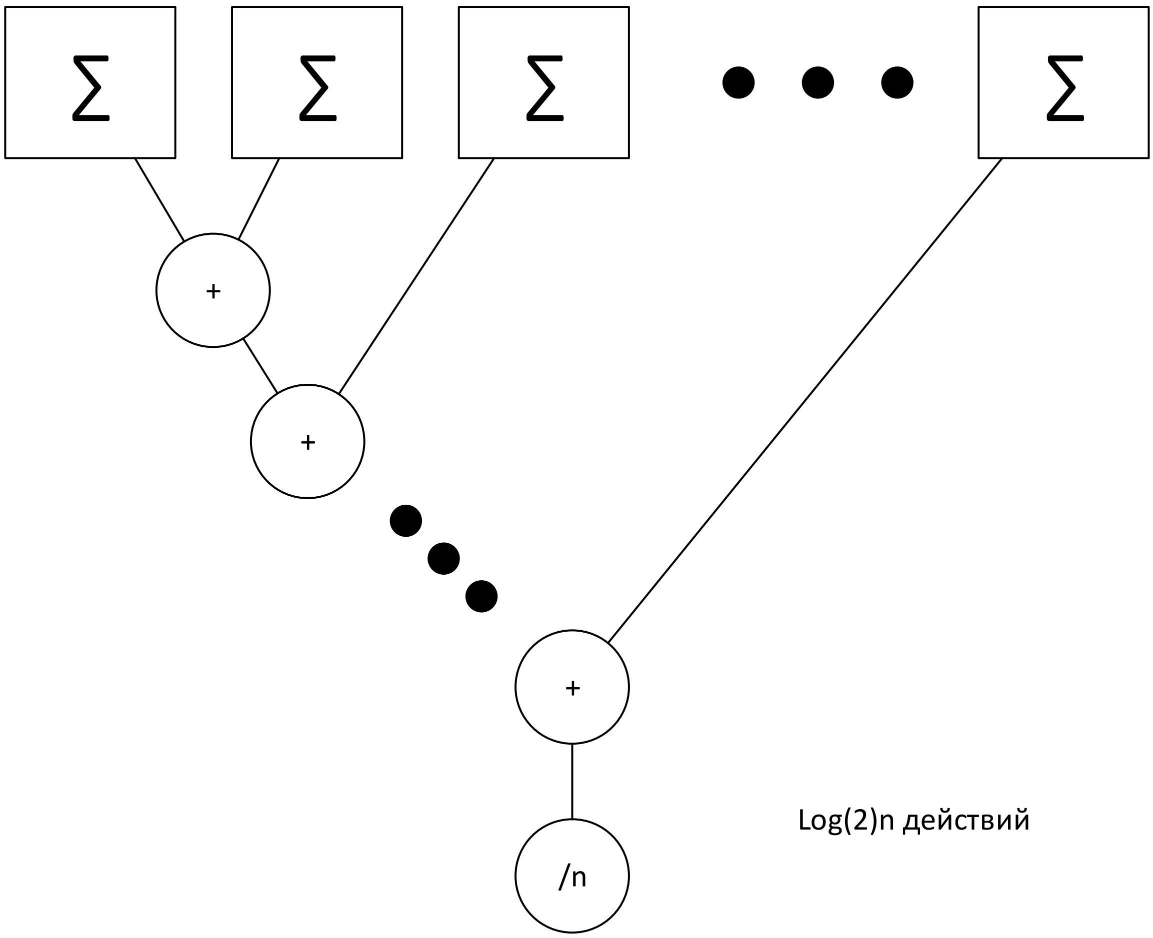
Итого для такой реализации оценка сложности O(n)

2. Схема распараллеливания алгоритма

Пусть у нас есть P процессоров, за счет ассоциативности сложения можно разделить входные данные на p блоков, внутри них провести сложение, затем сложить результаты в блоках и разделить на n

3. Описание в виде «Операции и операнды»





4. Сложность последовательной реализации

1) При последовательной реализации мы в начале выполняем n-1 действие для сложения а затем еще одно действие – деление на n, тогда:

$$T_1(n) = n$$

2) Пусть у нас не ограничено число процессоров. Рассмотрим каскадную схему ($P \geq n$)

$$T_\infty(n) = \log_2 n$$

3) Ускорение

$$S_p = T_1(n) / T_\infty(n) = n / \log_2 n$$

4) Эффективность

$$E_p = T_1(n)/p*T_p(n) = n/n(\log_2 n) \rightarrow 0 \ (n \rightarrow \infty)$$

По теореме о нижней оценке времени, время выполнения параллельного алгоритма будет не меньше чем $\log_2 n$. Алгоритм, предложенный выше, действует при $p \geq n/\log_2 n$, тогда

$$Tp \leq 2\log_2 n$$

Найдем ускорение

$$S_p = n/2\log_2 n$$

И эффективность

$$E_p = n/(p2\log_2 n) \geq n/((n/\log_2 n)\log_2 n) = 1$$

Таким образом максимальная эффективность достигается при $P = n/\log_2 n$

Найдем максимальное ускорение по закону Амдала: доля вычислений, которую нельзя производить параллельно стремится к 0 при $n \to \infty$. Таким образом получаем в общем виде:

$$S_{max} \to 1/(1/p)=p$$

В нашем случае:

$$S_{max} \to n/\log_2 n$$