

# ПАРАЛЛЕЛЬНОЕ ПРОГРАММИРОВАНИЕ

---

Параллельные вычисления

# ПАРАЛЛЕЛЬНОЕ ПРОГРАММИРОВАНИЕ

**1. Программирование на параллельных вычислительных структурах**

**2. Выявление естественного параллелизма задачи и синтез специализированной вычислительной структуры**

**Параллельная ярусная форма представления**



# ХАРАКТЕРИСТИКИ ПАРАЛЛЕЛЬНЫХ ЯРУСНЫХ ФОРМ ПАРАЛЛЕЛЬНЫХ АЛГОРИТМОВ

**Ярус** - каждая группа операций параллельной формы – зависящих от начальных данных или результатов выполнения операций в предыдущих группах

**Высота параллельной формы** — число групп в форме

**Ширина параллельной формы** — максимальное число операций в ярусе

**Максимальная параллельная форма** — параллельная форма алгоритмов с минимальной высотой – определяет минимальное время исполнения на гипотетической машине

# ПРОБЛЕМА ПОСТРОЕНИЯ ПАРАЛЛЕЛЬНЫХ ФОРМ С ЗАДАННЫМИ СВОЙСТВАМИ

**Вариант 1: 4 процессора. Решение:  $h = 3$ ,  $\Delta = 4$ ,  $\tau_0 = 5$ ,  $\tau_w = 3$**

Данные	$a_1, a_2$	$a_3, a_4$	$a_5, a_6$	$a_7, a_8$	Простой
Ярус 1	$a_1 * a_2$	$a_3 * a_4$	$a_5 * a_6$	$a_7 * a_8$	-
Ярус 2	$(a_1 * a_2 + a_3 * a_4)$	-	$(a_1 * a_2 + a_3 * a_4)$	-	2 ФУ
Ярус 3	$() * ()$	-	-	-	3 ФУ

**Вариант 2: 2 процессора. Решение:  $h = 4$ ,  $\Delta = 2$ ,  $\tau_0 = 1$ ,  $\tau_w = 4$**

Данные	$a_1, a_2 / a_5, a_6$	$a_3, a_4 / a_7, a_8$	Простой
Ярус 1	$a_1 * a_2$	$a_3 * a_4$	-
Ярус 2	$a_5 * a_6$	$a_7 * a_8$	-
Ярус 3	$(a_1 * a_2 + a_3 * a_4)$	$(a_5 * a_6 + a_7 * a_8)$	-
Ярус 4	$() * ()$	-	1 ФУ

# ПРОБЛЕМА ПОСТРОЕНИЯ ПАРАЛЛЕЛЬНЫХ ФОРМ С ЗАДАННЫМИ СВОЙСТВАМИ

**Вариант 3: 2 процессора. Решение:  $h = 5$ ,  $\Delta = 2$ ,  $\tau_0 = 3$ ,  $\tau_w = 5$**

Данные	$a_1, a_2 / a_5, a_6$	$a_3, a_4 / a_7, a_8$	Простой
Ярус 1	$a_1 * a_2$	$a_3 * a_4$	-
Ярус 2	$(a_1 * a_2 + a_3 * a_4)$	$a_5 * a_6$	-
Ярус 3	-	$a_7 * a_8$	1 ФУ
Ярус 4	$(a_5 * a_6 + a_7 * a_8)$	-	1 ФУ
Ярус 5	$( ) * ( )$	-	1 ФУ

# ПРОБЛЕМА ПОСТРОЕНИЯ ПАРАЛЛЕЛЬНЫХ ФОРМ С ЗАДААННЫМИ СВОЙСТВАМИ

**Вариант 1: Процесс сдваивания.  $h = \log_2 n$ ,  $\Delta = n / 2$ ,  $\tau_w = \log_2 n$**

Данные	a1, a2	a3, a4	a5, a6	a7, a8	Простой
Ярус 1	$a1 * a2$	$a3 * a4$	$a5 * a6$	$a7 * a8$	-
Ярус 2	$(a1 * a2) * (a3 * a4)$	-	$(a5 * a6) * (a7 * a8)$	-	2 ФУ
Ярус 3	$(a1 * a2 * a3 * a4) * (a5 * a6 * a7 * a8)$	-	-	-	3 ФУ

**Вариант 2: 4 процессора.  $h = 3$ ,  $\Delta = 4$ ,  $\tau_0 = 5$ ,  $\tau_w = 3$**

Данные	a1, a2	a3, a4	a5, a6	a7, a8	$\tau_0$
Ярус 1	$a1 * a2$	$a3 * a4$	$a5 * a6$	$a7 * a8$	
Ярус 2	$(a1 * a2) * a3$	$(a1 * a2) * (a3 * a4)$	$(a5 * a6) * a7$	$(a5 * a6) * (a7 * a8)$	
Ярус 3	$(a1 * a2 * a3 * a4) * a5$	$(a1 * a2 * a3 * a4) * a5 * a6$	$(a1 * a2 * a3 * a4) * (a5 * a6 * a7)$	$(a1 * a2 * a3 * a4) * (a5 * a6 * a7 * a8)$	

# ПРОБЛЕМА ПОСТРОЕНИЯ ПАРАЛЛЕЛЬНЫХ ФОРМ С ЗАДАННЫМИ СВОЙСТВАМИ

Вычислить:  $\prod_{i=1}^{n-1} a_i$

**Вариант 3: 1 процессор.  $h = 7$ ,  $\Delta = 1$ ,  $\tau_0 = 0$ ,  $\tau_w = 7$**

1	$a_1 * a_2$
2	$(a_1 * a_2) * a_3$
3	$(a_1 * a_2 * a_3) * a_4$
4	$(a_1 * a_2 * a_3 * a_4) * a_5$
5	$(a_1 * a_2 * a_3 * a_4 * a_5) * a_6$
6	$(a_1 * a_2 * a_3 * a_4 * a_5 * a_6) * a_7$
7	$(a_1 * a_2 * a_3 * a_4 * a_5 * a_6 * a_7) * a_8$

$$\left( \prod_{i=1}^n a_i \right)_{\text{Алгоритм 2}} = \left( \prod_{i=1}^n a_i \right)_{\text{Алгоритм 3}}$$

— ассоциативность умножения

# ПРОБЛЕМА ПОСТРОЕНИЯ ПАРАЛЛЕЛЬНЫХ ФОРМ С ЗАДАННЫМИ СВОЙСТВАМИ

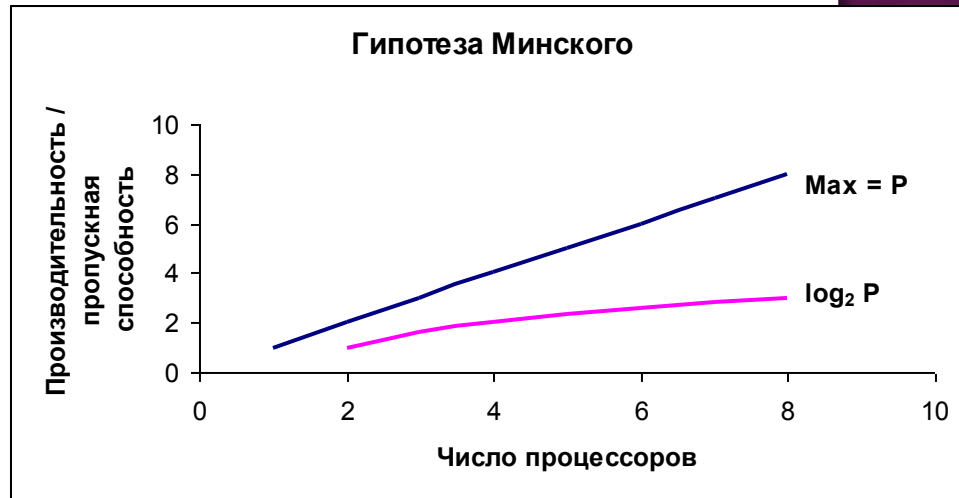
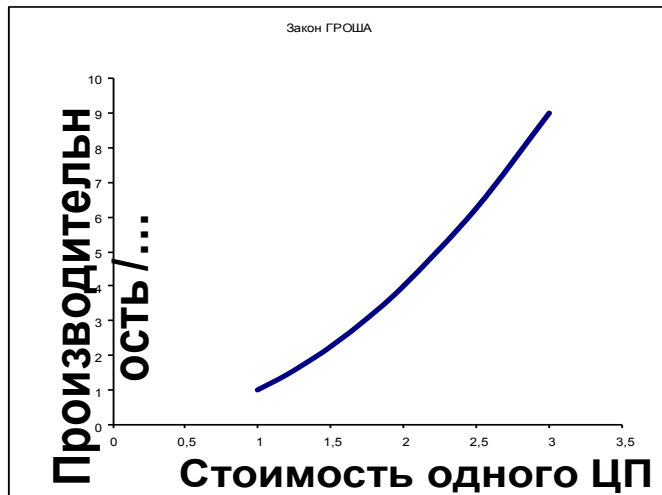
1. Лишние операции
2. Чем больше промежуточных результатов требуется, тем эффективней параллельный алгоритм

**Устойчивость параллельных алгоритмов хуже, чем последовательных.**

**При большом числе процессоров эти области, по крайней мере, различны**



# ЗАКОН ГРОША И ГИПОТЕЗА МИНСКОГО



# ЗАДАЧИ, КРИТЕРИИ, ВОПРОСЫ

## Новые задачи:

- ☐ Исключение взаимных блокировок и бесполезных обменов сообщениями
- ☐ Защита от нежелательных состязаний
- ☐ Избежать образования слишком большого количества параллельных процессов
- ☐ Обнаружение завершения программы простейшим способом.

## Новые критерии:

- ☐ Ускорение программы в зависимости от числа процессоров
- ☐ Затраты времени на синхронизацию
- ☐ Влияние размера задачи на ускорение
- ☐ Максимальное число занятых процессоров при решении задачи
- ☐ Детерминизм выполнения программы

## Вопросы нового типа:

- ☐ Размер порции параллельной программы?
- ☐ Сколько параллельных процессов должно быть образовано?
- ☐ Какой вид синхронизации процессов должен быть принят?
- ☐ Как должен управляться доступ к разделяемым данным?
- ☐ Как гарантировать детерминированное выполнение программы?
- ☐ Как разделить на подзадачи, чтобы параллельное оборудование использовалось эффективно?