

# Параллельные архитектуры и алгоритмы

## Презентация

Денис Морозов  
morozov.den@samsung.com

<sup>1</sup>Mobile Lab 2  
Samsung R&D

<sup>2</sup>Department of Computer Science, doctoral studies  
University of "Kiev-Mogyla Academy"

Январь, 2012

# План

**1** Параллельные архитектуры

**2** Параллельные алгоритмы

**3** Клеточные автоматы

# Классификация Флинна

Классификация архитектур, основанная на рассмотрении числа потоков инструкций и потоков данных

- SISD - Single Instruction stream over a Single Data stream
- SIMD - Single Instruction Multiple Data
- MISD - Multiple Instruction Single Data
- MIMD - Multiple Instruction Multiple Data

# Классификация Флинна

Классификация архитектур, основанная на рассмотрении числа потоков инструкций и потоков данных

- SISD - Single Instruction stream over a Single Data stream
- SIMD - Single Instruction Multiple Data
- MISD - Multiple Instruction Single Data
- MIMD - Multiple Instruction Multiple Data

# Классификация Флинна

Классификация архитектур, основанная на рассмотрении числа потоков инструкций и потоков данных

- SISD - Single Instruction stream over a Single Data stream
- SIMD - Single Instruction Multiple Data
- MISD - Multiple Instruction Single Data
- MIMD - Multiple Instruction Multiple Data

# Классификация Флинна

Классификация архитектур, основанная на рассмотрении числа потоков инструкций и потоков данных

- SISD - Single Instruction stream over a Single Data stream
- SIMD - Single Instruction Multiple Data
- MISD - Multiple Instruction Single Data
- MIMD - Multiple Instruction Multiple Data

# Классификация Флинна

Классификация архитектур, основанная на рассмотрении числа потоков инструкций и потоков данных

- SISD - Single Instruction stream over a Single Data stream
- SIMD - Single Instruction Multiple Data
- MISD - Multiple Instruction Single Data
- MIMD - Multiple Instruction Multiple Data

# SMP (symmetric multiprocessing)

SMP (symmetric multiprocessing) - симметричная многопроцессорная архитектура. Главной особенностью систем с архитектурой SMP является наличие общей физической памяти, разделяемой всеми процессорами. При работе с SMP-системами используют так называемую парадигму программирования с разделяемой памятью - shared memory paradigm (OpenMP).



Возможны конфликты при обращении к общей памяти

- CREW
- CRCW

# MPP (massive parallel processing)

MPP (massive parallel processing) - массивно-параллельная архитектура. Главная особенность такой архитектуры состоит в том, что память физически разделена. При работе с MPP-системами используют Massive Passing Programming Paradigm - парадигму программирования с передачей данных (MPI, PVM, BSPlib)

# Гибридная архитектура NUMA

Главная особенность гибридной архитектуры NUMA (nonuniform memory access) - неоднородный доступ к памяти. Архитектура cc-NUMA (Cache Coherent Non-Uniform Memory Access) - неоднородный доступ к памяти с обеспечением когерентности кэшей.

# Конвейерные процессоры

выборка команды;  
расшифровка команды;  
выборка необходимых операндов;  
выполнение команды;  
сохранение результатов.

# Матричные процессоры

Наиболее распространенными из систем класса один поток команд - множество потоков данных (SIMD) являются матричные системы, которые лучше всего приспособлены для решения задач, характеризующихся параллелизмом независимых объектов или данных.

Моделью параллельных вычислений на матричных(векторных) ВС являются клеточные автоматы.

# Процессоры баз данных

Создание такого рода систем связывается с реализацией параллелизма при выполнении последовательности операций и транзакций, а также конвейерной потоковой обработки данных.

# Кластеры

Кластер представляет собой два или более компьютеров (часто называемых узлами), объединяемых при помощи сетевых технологий на базе шинной архитектуры или коммутатора и предстающих перед пользователями в качестве единого информационно-вычислительного ресурса. Важна топология связи процессоров в кластерной системе.

# Закон Амдала

Самое важное в вопросе эффективности параллельного кода  
- это хороший дизайн самого алгоритма.

$$S(p) = \frac{t_s}{ft_s + (1 - f)t_s/p} = \frac{p}{1 + (p - 1)f}$$



# Примеры алгоритмов

- алгоритм сложения сдвигиванием
- скалярное произведение векторов
- умножение матрицы на вектор
- умножение матриц
- алгоритм параллельной сортировки Бэтчера
- нахождение решения уравнений в конечных разностях
- свертка с большим ядром над полем комплексных чисел

# Примеры алгоритмов

- алгоритм сложения сдвигиванием
- скалярное произведение векторов
- умножение матрицы на вектор
- умножение матриц
- алгоритм параллельной сортировки Бэтчера
- нахождение решения уравнений в конечных разностях
- свертка с большим ядром над полем комплексных чисел

# Примеры алгоритмов

- алгоритм сложения сдвигиванием
- скалярное произведение векторов
- умножение матрицы на вектор
- умножение матриц
- алгоритм параллельной сортировки Бэтчера
- нахождение решения уравнений в конечных разностях
- свертка с большим ядром над полем комплексных чисел

# Примеры алгоритмов

- алгоритм сложения сдвигиванием
- скалярное произведение векторов
- умножение матрицы на вектор
- умножение матриц
- алгоритм параллельной сортировки Бэтчера
- нахождение решения уравнений в конечных разностях
- свертка с большим ядром над полем комплексных чисел

# Примеры алгоритмов

- алгоритм сложения сдвигиванием
- скалярное произведение векторов
- умножение матрицы на вектор
- умножение матриц
- алгоритм параллельной сортировки Бэтчера
- нахождение решения уравнений в конечных разностях
- свертка с большим ядром над полем комплексных чисел

# Примеры алгоритмов

- алгоритм сложения сдвигиванием
- скалярное произведение векторов
- умножение матрицы на вектор
- умножение матриц
- алгоритм параллельной сортировки Бэтчера
- нахождение решения уравнений в конечных разностях
- свертка с большим ядром над полем комплексных чисел

# Примеры алгоритмов

- алгоритм сложения сдвигиванием
- скалярное произведение векторов
- умножение матрицы на вектор
- умножение матриц
- алгоритм параллельной сортировки Бэтчера
- нахождение решения уравнений в конечных разностях
- свертка с большим ядром над полем комплексных чисел

# Примеры алгоритмов

- алгоритм сложения сдвигиванием
- скалярное произведение векторов
- умножение матрицы на вектор
- умножение матриц
- алгоритм параллельной сортировки Бэтчера
- нахождение решения уравнений в конечных разностях
- свертка с большим ядром над полем комплексных чисел



# Алгоритм Бэтчера

## Определение

- 1 начальная установка  $p$  . Установить  $p \leftarrow 2^{t-1}$  , где  $t = \log_2 N$  - наименьшее целое число, такое, что  $2^t = N$  . (Шаги 2-5 будут выполняться с  $p = 2^{t-1}, 2^{t-2}, \dots, 1$  .)
- 2 начальная установка  $q, r, d$  . Установить  $q \leftarrow 2^{t-1}, r \leftarrow 0, d \leftarrow p$  .
- 3 цикл по  $i$  . Для всех  $t$  , таких, что  $0 \leq i < N - d$  и  $i \wedge p = r$  , выполнять шаг 4.  
Затем перейти к шагу 5.
- 4 Сравнение/обмен  $0 \leq i < N - d, K_{i+1} \leftrightarrow K_{i+d+1}$
- 5 Цикл по  $q$  . Если  $q \neq p$  , установить  $d \leftarrow q - p, q \leftarrow q/2, r \leftarrow p$  и возвратиться к шагу 3.
- 6 Цикл по  $p$  . Установить  $p \leftarrow \lfloor p/2 \rfloor$  . Если  $p > 0$  , возвратиться к шагу 2.

# Алгоритм Бэтчера

## Определение

- 1 начальная установка  $p$  . Установить  $p \leftarrow 2^{t-1}$  , где  $t = \log_2 N$  - наименьшее целое число, такое, что  $2^t = N$  . (Шаги 2-5 будут выполняться с  $p = 2^{t-1}, 2^{t-2}, \dots, 1$  .)
- 2 начальная установка  $q, r, d$  . Установить  $q \leftarrow 2^{t-1}, r \leftarrow 0, d \leftarrow p$  .
- 3 цикл по  $i$  . Для всех  $t$  , таких, что  $0 \leq i < N - d$  и  $i \wedge p = r$  , выполнять шаг 4.  
Затем перейти к шагу 5.
- 4 Сравнение/обмен  $0 \leq i < N - d, K_{i+1} \leftrightarrow K_{i+d+1}$
- 5 Цикл по  $q$  . Если  $q \neq p$  , установить  $d \leftarrow q - p, q \leftarrow q/2, r \leftarrow p$  и возвратиться к шагу 3.
- 6 Цикл по  $p$  . Установить  $p \leftarrow \lfloor p/2 \rfloor$  . Если  $p > 0$  , возвратиться к шагу 2.

# Алгоритм Бэтчера

## Определение

- 1 начальная установка  $p$  . Установить  $p \leftarrow 2^{t-1}$  , где  $t = \log_2 N$  - наименьшее целое число, такое, что  $2^t = N$  .  
(Шаги 2-5 будут выполняться с  $p = 2^{t-1}, 2^{t-2}, \dots, 1$  .)
- 2 начальная установка  $q, r, d$  . Установить  $q \leftarrow 2^{t-1}, r \leftarrow 0, d \leftarrow p$  .
- 3 цикл по  $i$  . Для всех  $t$  , таких, что  $0 \leq i < N - d$  и  $i \wedge p = r$  , выполнять шаг 4.  
Затем перейти к шагу 5.
- 4 Сравнение/обмен  $0 \leq i < N - d, K_{i+1} \leftrightarrow K_{i+d+1}$
- 5 Цикл по  $q$  . Если  $q \neq p$  , установить  $d \leftarrow q - p, q \leftarrow q/2, r \leftarrow p$  и возвратиться к шагу 3.
- 6 Цикл по  $p$  . Установить  $p \leftarrow \lfloor p/2 \rfloor$  . Если  $p > 0$  , возвратиться к шагу 2.

# Алгоритм Бэтчера

## Определение

- 1 начальная установка  $p$  . Установить  $p \leftarrow 2^{t-1}$  , где  $t = \log_2 N$  - наименьшее целое число, такое, что  $2^t = N$  .  
(Шаги 2-5 будут выполняться с  $p = 2^{t-1}, 2^{t-2}, \dots, 1$  .)
- 2 начальная установка  $q, r, d$  . Установить  $q \leftarrow 2^{t-1}, r \leftarrow 0, d \leftarrow p$  .
- 3 цикл по  $i$  . Для всех  $t$  , таких, что  $0 \leq i < N - d$  и  $i \wedge p = r$  , выполнять шаг 4.  
Затем перейти к шагу 5.
- 4 Сравнение/обмен  $0 \leq i < N - d, K_{i+1} \leftrightarrow K_{i+d+1}$
- 5 Цикл по  $q$  . Если  $q \neq p$  , установить  $d \leftarrow q - p, q \leftarrow q/2, r \leftarrow p$  и возвратиться к шагу 3.
- 6 Цикл по  $p$  . Установить  $p \leftarrow \lfloor p/2 \rfloor$  . Если  $p > 0$  , возвратиться к шагу 2.

# Алгоритм Бэтчера

## Определение

- 1 начальная установка  $p$  . Установить  $p \leftarrow 2^{t-1}$  , где  $t = \log_2 N$  - наименьшее целое число, такое, что  $2^t = N$  .  
(Шаги 2-5 будут выполняться с  $p = 2^{t-1}, 2^{t-2}, \dots, 1$  .)
- 2 начальная установка  $q, r, d$  . Установить  $q \leftarrow 2^{t-1}, r \leftarrow 0, d \leftarrow p$  .
- 3 цикл по  $i$  . Для всех  $t$  , таких, что  $0 \leq i < N - d$  и  $i \wedge p = r$  , выполнять шаг 4.  
Затем перейти к шагу 5.
- 4 Сравнение/обмен  $0 \leq i < N - d, K_{i+1} \leftrightarrow K_{i+d+1}$
- 5 Цикл по  $q$  . Если  $q \neq p$  , установить  $d \leftarrow q - p, q \leftarrow q/2, r \leftarrow p$  и возвратиться к шагу 3.
- 6 Цикл по  $p$  . Установить  $p \leftarrow \lfloor p/2 \rfloor$  . Если  $p > 0$  , возвратиться к шагу 2.

# Алгоритм Бэтчера

## Определение

- 1 начальная установка  $p$  . Установить  $p \leftarrow 2^{t-1}$  , где  $t = \log_2 N$  - наименьшее целое число, такое, что  $2^t = N$  .  
(Шаги 2-5 будут выполняться с  $p = 2^{t-1}, 2^{t-2}, \dots, 1$  .)
- 2 начальная установка  $q, r, d$  . Установить  $q \leftarrow 2^{t-1}, r \leftarrow 0, d \leftarrow p$  .
- 3 цикл по  $i$  . Для всех  $t$  , таких, что  $0 \leq i < N - d$  и  $i \wedge p = r$  , выполнять шаг 4.  
Затем перейти к шагу 5.
- 4 Сравнение/обмен  $0 \leq i < N - d, K_{i+1} \leftrightarrow K_{i+d+1}$
- 5 Цикл по  $q$  . Если  $q \neq p$  , установить  $d \leftarrow q - p, q \leftarrow q/2, r \leftarrow p$  и возвратиться к шагу 3.
- 6 Цикл по  $p$  . Установить  $p \leftarrow \lfloor p/2 \rfloor$  . Если  $p > 0$  ,  
возвратиться к шагу 2.

# Алгоритм Бэтчера

## Определение

- 1 начальная установка  $p$  . Установить  $p \leftarrow 2^{t-1}$  , где  $t = \log_2 N$  - наименьшее целое число, такое, что  $2^t = N$  .  
(Шаги 2-5 будут выполняться с  $p = 2^{t-1}, 2^{t-2}, \dots, 1$  .)
- 2 начальная установка  $q, r, d$  . Установить  $q \leftarrow 2^{t-1}, r \leftarrow 0, d \leftarrow p$  .
- 3 цикл по  $i$  . Для всех  $t$  , таких, что  $0 \leq i < N - d$  и  $i \wedge p = r$  , выполнять шаг 4.  
Затем перейти к шагу 5.
- 4 Сравнение/обмен  $0 \leq i < N - d, K_{i+1} \leftrightarrow K_{i+d+1}$
- 5 Цикл по  $q$  . Если  $q \neq p$  , установить  $d \leftarrow q - p, q \leftarrow q/2, r \leftarrow p$  и возвратиться к шагу 3.
- 6 Цикл по  $p$  . Установить  $p \leftarrow \lfloor p/2 \rfloor$  . Если  $p > 0$  , возвратиться к шагу 2.

# Алгоритм Бэтчера

## Пример

p	4	2	2	1	1	1
q	4	4	2	4	2	1
r	0	0	2	0	1	1
d	4	2	2	1	3	1

1  $1 \leftrightarrow 5, 2 \leftrightarrow 6, 3 \leftrightarrow 7, 4 \leftrightarrow 8$

2  $1 \leftrightarrow 3, 2 \leftrightarrow 4, 5 \leftrightarrow 7, 6 \leftrightarrow 8$

3  $3 \leftrightarrow 5, 4 \leftrightarrow 6$

4  $1 \leftrightarrow 2, 3 \leftrightarrow 4, 5 \leftrightarrow 6, 7 \leftrightarrow 8$

5  $2 \leftrightarrow 5, 4 \leftrightarrow 7$

6  $2 \leftrightarrow 3, 4 \leftrightarrow 5, 6 \leftrightarrow 7$



# Алгоритм Бэтчера

## Пример

p	4	2	2	1	1	1
q	4	4	2	4	2	1
r	0	0	2	0	1	1
d	4	2	2	1	3	1

**1**  $1 \leftrightarrow 5, 2 \leftrightarrow 6, 3 \leftrightarrow 7, 4 \leftrightarrow 8$

2  $1 \leftrightarrow 3, 2 \leftrightarrow 4, 5 \leftrightarrow 7, 6 \leftrightarrow 8$

3  $3 \leftrightarrow 5, 4 \leftrightarrow 6$

4  $1 \leftrightarrow 2, 3 \leftrightarrow 4, 5 \leftrightarrow 6, 7 \leftrightarrow 8$

5  $2 \leftrightarrow 5, 4 \leftrightarrow 7$

6  $2 \leftrightarrow 3, 4 \leftrightarrow 5, 6 \leftrightarrow 7$

# Алгоритм Бэтчера

## Пример

p	4	2	2	1	1	1
q	4	4	2	4	2	1
r	0	0	2	0	1	1
d	4	2	2	1	3	1

**1**  $1 \leftrightarrow 5, 2 \leftrightarrow 6, 3 \leftrightarrow 7, 4 \leftrightarrow 8$

**2**  $1 \leftrightarrow 3, 2 \leftrightarrow 4, 5 \leftrightarrow 7, 6 \leftrightarrow 8$

**3**  $3 \leftrightarrow 5, 4 \leftrightarrow 6$

**4**  $1 \leftrightarrow 2, 3 \leftrightarrow 4, 5 \leftrightarrow 6, 7 \leftrightarrow 8$

**5**  $2 \leftrightarrow 5, 4 \leftrightarrow 7$

**6**  $2 \leftrightarrow 3, 4 \leftrightarrow 5, 6 \leftrightarrow 7$

# Алгоритм Бэтчера

## Пример

p	4	2	2	1	1	1
q	4	4	2	4	2	1
r	0	0	2	0	1	1
d	4	2	2	1	3	1

**1**  $1 \leftrightarrow 5, 2 \leftrightarrow 6, 3 \leftrightarrow 7, 4 \leftrightarrow 8$

**2**  $1 \leftrightarrow 3, 2 \leftrightarrow 4, 5 \leftrightarrow 7, 6 \leftrightarrow 8$

**3**  $3 \leftrightarrow 5, 4 \leftrightarrow 6$

4  $1 \leftrightarrow 2, 3 \leftrightarrow 4, 5 \leftrightarrow 6, 7 \leftrightarrow 8$

5  $2 \leftrightarrow 5, 4 \leftrightarrow 7$

6  $2 \leftrightarrow 3, 4 \leftrightarrow 5, 6 \leftrightarrow 7$

# Алгоритм Бэтчера

## Пример

p	4	2	2	1	1	1
q	4	4	2	4	2	1
r	0	0	2	0	1	1
d	4	2	2	1	3	1

**1**  $1 \leftrightarrow 5, 2 \leftrightarrow 6, 3 \leftrightarrow 7, 4 \leftrightarrow 8$

**2**  $1 \leftrightarrow 3, 2 \leftrightarrow 4, 5 \leftrightarrow 7, 6 \leftrightarrow 8$

**3**  $3 \leftrightarrow 5, 4 \leftrightarrow 6$

**4**  $1 \leftrightarrow 2, 3 \leftrightarrow 4, 5 \leftrightarrow 6, 7 \leftrightarrow 8$

**5**  $2 \leftrightarrow 5, 4 \leftrightarrow 7$

**6**  $2 \leftrightarrow 3, 4 \leftrightarrow 5, 6 \leftrightarrow 7$

# Алгоритм Бэтчера

## Пример

p	4	2	2	1	1	1
q	4	4	2	4	2	1
r	0	0	2	0	1	1
d	4	2	2	1	3	1

1  $1 \leftrightarrow 5, 2 \leftrightarrow 6, 3 \leftrightarrow 7, 4 \leftrightarrow 8$

2  $1 \leftrightarrow 3, 2 \leftrightarrow 4, 5 \leftrightarrow 7, 6 \leftrightarrow 8$

3  $3 \leftrightarrow 5, 4 \leftrightarrow 6$

4  $1 \leftrightarrow 2, 3 \leftrightarrow 4, 5 \leftrightarrow 6, 7 \leftrightarrow 8$

5  $2 \leftrightarrow 5, 4 \leftrightarrow 7$

6  $2 \leftrightarrow 3, 4 \leftrightarrow 5, 6 \leftrightarrow 7$

# Алгоритм Бэтчера

## Пример

p	4	2	2	1	1	1
q	4	4	2	4	2	1
r	0	0	2	0	1	1
d	4	2	2	1	3	1

1  $1 \leftrightarrow 5, 2 \leftrightarrow 6, 3 \leftrightarrow 7, 4 \leftrightarrow 8$

2  $1 \leftrightarrow 3, 2 \leftrightarrow 4, 5 \leftrightarrow 7, 6 \leftrightarrow 8$

3  $3 \leftrightarrow 5, 4 \leftrightarrow 6$

4  $1 \leftrightarrow 2, 3 \leftrightarrow 4, 5 \leftrightarrow 6, 7 \leftrightarrow 8$

5  $2 \leftrightarrow 5, 4 \leftrightarrow 7$

6  $2 \leftrightarrow 3, 4 \leftrightarrow 5, 6 \leftrightarrow 7$

# Разностная схема

Уравнение в конечных разностях:

$$\frac{\partial^2 f}{\partial x^2} - \frac{\partial f}{\partial y} = 0$$

# Свертка

$$(x^{n_1} + a_{n_1-1}x^{n_1-1} + \dots + a_0) \cdot (x^{n_2} + b_{n_2-1}x^{n_2-1} + \dots + b_0)$$

$$(x^{n_1} + a_{n_1-1}x^{n_1-1} + \dots + a_0) \cdot (x - x_1) \cdot \dots \cdot (x - x_{n_2})$$



# Литература I



Дональд Э. Кнут

Искусство программирования, том 3. Сортировка и поиск

Вильямс, 2011



Т. Тоффоли, Н. Марголус

Машины клеточных автоматов

Мир, 1991.



Дж.фон Нейман

Теория самовоспроизводящихся автоматов

Мир, 1971.



Миронов А.А., Карпов А.Н.

Параллельные алгоритмы обработки данных

<http://www.viva64.com/ru/a/0032/>