2. Моделирование и анализ параллельных вычислений.

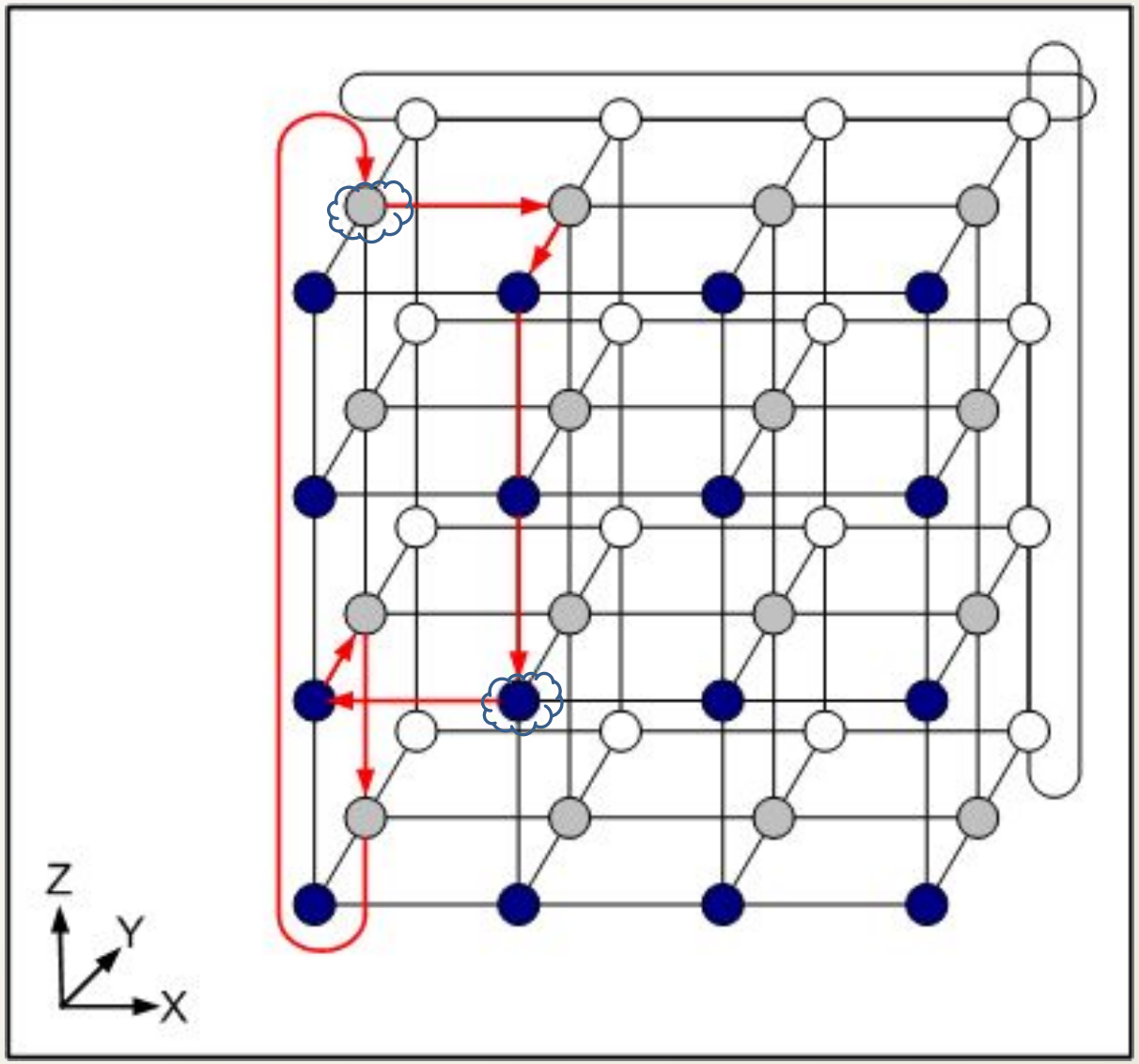
Коммуникационная трудоемкость параллельных алгоритмов.

Передача данных. Коммуникационная трудоемкость алгоритмов

- В рассмотренных оценках **не учтены затраты времени на передачу данных.**
- Основа для характеристики передачи данных **– алгоритмы маршрутизизации (АМ)**.
- АМ определяет **путь передачи данных** от CPU1 (источника сообщения) до CPU2 (адресата доставки).
- **Классификация АМ**:
  - **Оптимальные** (определяют **кратчайшие** пути передачи данных) и **неоптимальные** АМ.
  - **Адаптивные** (учитывают **загрузку** каналов связи) и **неадаптивные** АМ.

Пример оптимальных АМ

- Алгоритмы, основанные на **покоординатной** маршрутизации (*dimension ordered routing*) – **поочередный** поиск путей передачи данных для **каждой размерности** топологии сети.
- Пример: **алгоритм XY-маршрутизации** для решетки:
  - Передача данных по **горизонтали** до пересечения с вертикалью CPU2
  - Передача данных по **вертикали и т.д.**


Z
Y
X

Характеристики коммуникационной составляющей длительности выполнения параллельного алгоритма в МВС

- **Время передачи данных** определяют:
  - **Время начальной подготовки** сообщения для передачи, поиска маршрута в сети – $t_н$
  - **Время передачи служебных данных** (*заголовок сообщения, диагностический блок*) между соседними CPU (имеющими между собой физический канал передачи данных) – t_c
  - **Время передачи одного байта по одному каналу** (*определяется полосой пропускания каналов сети*) – $t_к = 1/R$, где R - ширина полосы, количество битов, передаваемых за 1сек.

Методы передачи данных

1. **Метод передачи данных (сообщений) как неделимых блоков информации** (*store-and-forward routing, SFR*):
   - CPU1
     - Готовит данные (сообщение) для передачи
     - Определяет CPU2 для пересылки (промежуточный)
     - Запускает операцию пересылки данных
   - CPU2
     - Принимает **полностью** все пересылаемые данные
     - Выполняет пересылку далее по маршруту

Время пересылки m байт по маршруту длины l (через l узлов) :

$$t_{пд} = t_н + (mt_к + t_с)l$$

Для «длинных» сообщений, где можно пренебречь временем пересылки служебных данных:

$$t_{пд} = t_н + mt_к l$$

Методы передачи данных

2. **Метод передачи пакетов** – сообщение состоит из блоков информации (пакетов) (*cut-through-routing, CTR*)

- CPU1
  - Готовит данные (сообщение) в виде пакетов для передачи
  - Определяет CPU2 для пересылки (промежуточный)
  - Запускает операцию пересылки пакетов
- CPU2
  - Принимает **пакет**
  - Выполняет пересылку далее по маршруту как только получил и обработал заголовок (учитывает t_c)

Время пересылки m байт по маршруту длины l:

$$t_{пд} = = t_н + mt_к + t_c l$$

Преимущества и недостатки CTR

- Ускоряет пересылку данных.
- Снижает потребность в памяти для хранения пересылаемых данных и организации приема-передачи сообщений.
- Для передачи могут использоваться одновременно разные коммуникационные каналы (в зависимости от топологии сети).
- Требует разработки более сложного аппаратного и программного обеспечения сети.
- Может увеличить накладные расходы (время подготовки и время передачи служебных данных),
- При передаче пакетов возможно возникновение конфликтных ситуаций.

Классификация операций передачи данных в МВС

- **передача** данных (сообщений):
  - между двумя CPU сети,
  - от одного CPU всем остальным CPU сети,
  - от всех CPU всем CPU сети,
  - то же для различных сообщений;
- **прием** данных (сообщений):
  - на один CPU от всех CPU сети,
  - на каждом CPU от всех CPU сети,
  - то же для различных сообщений.

Оценки трудоемкости для различных топологий

Топология	**Диаметр**
Граф	1
Звезда	2
Линейка	p - 1
Кольцо	p/2
Решетка (2D)	2(√p - 1)

- **Диаметр** – определяет время передачи данных, **max расстояние** между 2 CPU сети (*расстояние равно величине кратчайшего пути*).

Передача между двумя CPU сети (топология «кольцо»)

- Для оценки нужно:
  - Определить алгоритм пересылки.
  - В формулы вместо l подставить значение диаметра
- Передача сообщений

$$t_{пд} = t_н + mt_к p/2$$

(«длинные» сообщения)

- Передача пакетов

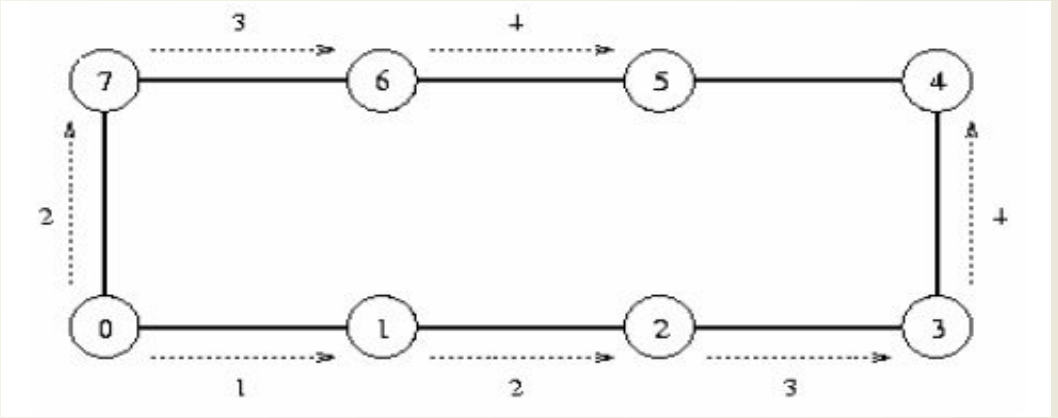
$$t_{пд} = = t_н + mt_к + t_с p/2$$

Передача от одного CPU всем остальным CPU сети

single-node broadcast

Прием на одном CPU от всех остальных CPU сети

single-node accumulation

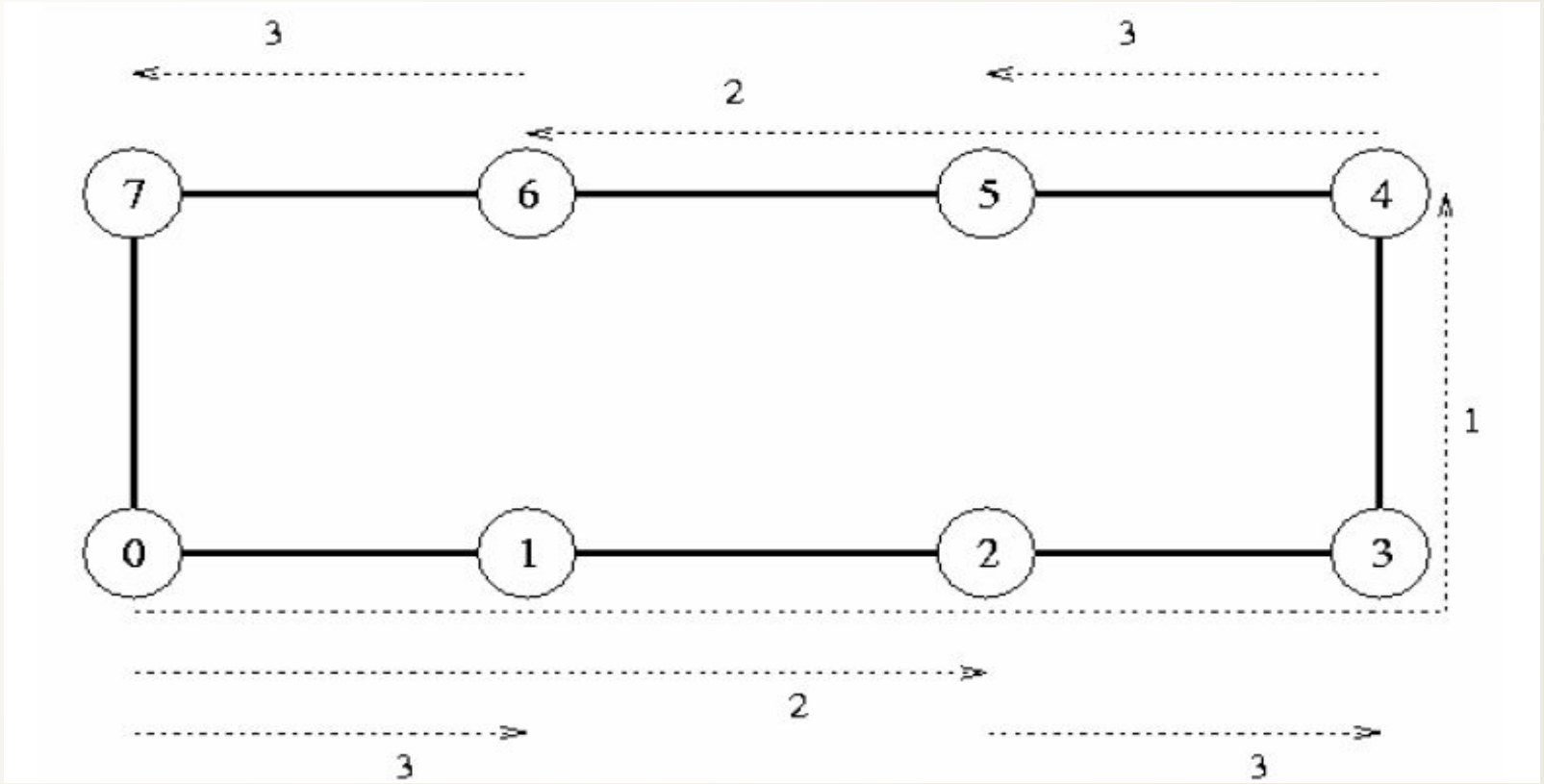


- Передача сообщений:
  - От источника к 2 соседним CPU
  - 2 соседних – далее по сети (кольцо)

$$t_{пд} = (t_н + mt_к)p/2$$

- Передача пакетов – «каскадно»
  - От источника к CPU на расстоянии $p/2$
  - CPU1 и CPU2 – CPU на расстоянии $p/4$
  - …

$$t_{пд} = \sum_{i=1}^{\log_2 p} (t_н + mt_к + t_c p/2^i) = (t_н + mt_к)\log_2 p + t_c(p-1)$$


3
3
2
7
6
5
4
1
0
1
2
3
2
3
3

Передача от всех CPU всем остальным CPU сети

multinode broadcast

Прием на всех CPU от всех остальных CPU сети

multinode accumulation

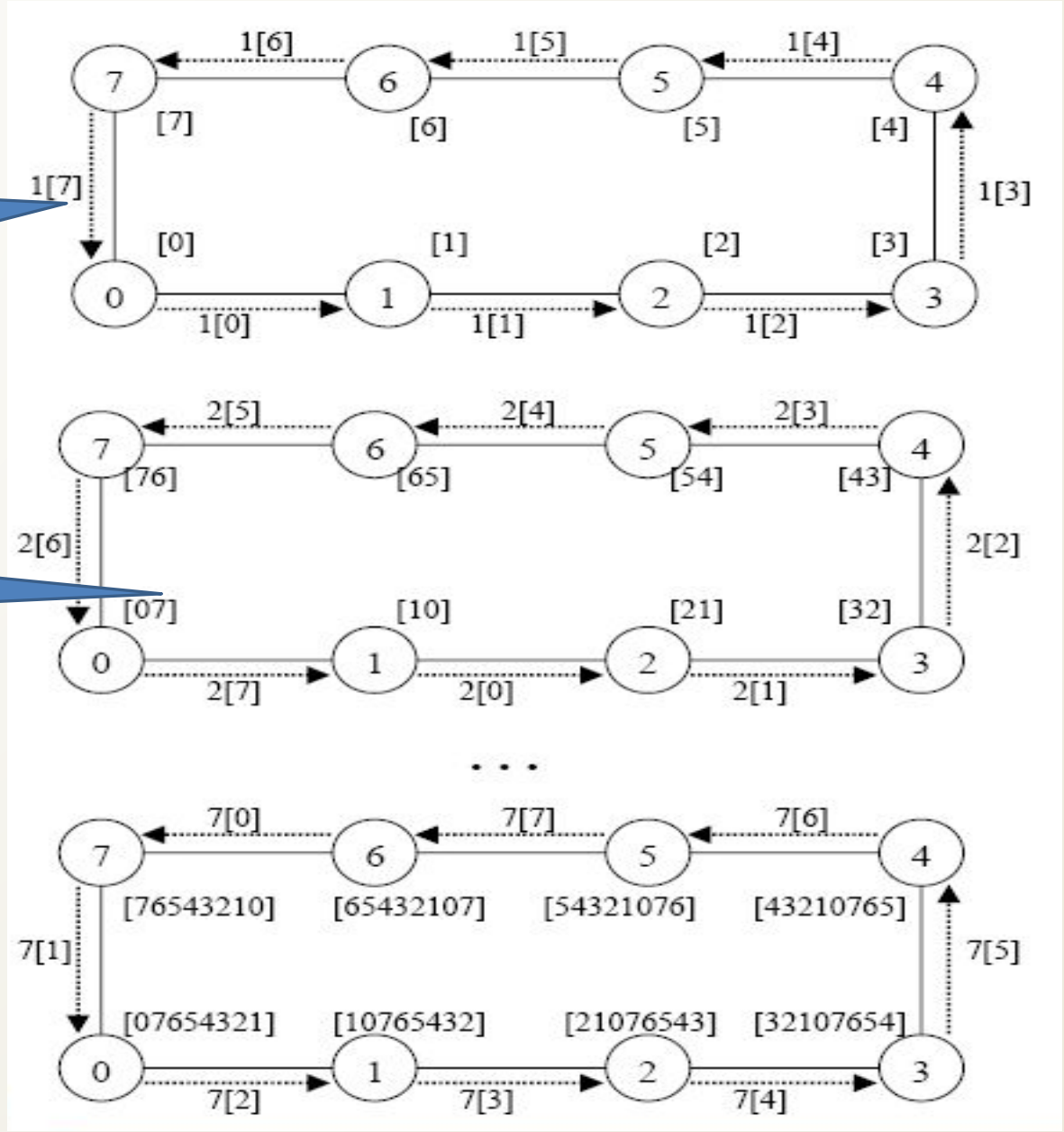
- Передача сообщений:
  - Все CPU могут одновременно рассылать сообщение в определенном направлении по кольцу
  - Рассылка закончится через $p-1$ шаг

$$t_{пд} = (t_н + mt_к)(p-1)$$

- Передача пакетов
  - Обобщение алгоритмов одиночной рассылки на случай множественной приводит к **перегрузке** каналов передачи данных
  - По одной линии собирается очередь - несколько пакетов, ожидающих передачи.
  - Теряется преимущество пакетной передачи.

Чье сообщение передается

Чьи сообщения пришли



Цикл 1 рассылки сообщений

Цикл 2 рассылки сообщений

Цикл p-1 рассылки сообщений

Обобщенная передача от одного CPU всем CPU сети

single-node scatter (рассеивание)

Обобщенный прием на одном CPU от всех CPU сети

single-node gather (сбор)

- Передача **разных** сообщений:
  - От источника половину сообщений для рассылки соседнему CPU
  - И т.д.

$$t_{пд} >= \alpha t_н + m t_к (p-1)$$

- Передача пакетов
  - Сопоставима по трудности с multi, т.к. разные данные не могут взаимодействовать при пересылке.

Обобщенная передача от всех CPU всем CPU сети
Обобщенный прием на всех CPU от всех CPU сети

total exchange

- Передача сообщений:
  - Все CPU одновременно рассылают свои сообщения в определенном направлении соседу по кольцу.
  - Все CPU отбирают среди полученных сообщений адресованные им.
  - Остальные сообщения пересылаются дальше.

$$t_{пд} = (t_н + 0.5p\ mt_к)(p-1)$$

- Передача пакетов
  - Преимущества у пакетной передачи нет.

Оценки коммуникационной трудоемкости для кластеров

- **Кластер** – группа выделенных рабочих станций (объединены в ЛВС, работают как **единый** вычислительный ресурс, используется **серийное** оборудование).
- Использование коммуникаторов (*hub, switch*)
- **Топология сети кластера – полный граф (*l=1*)**, **но**:
  - Hub **–**: в каждый момент передача данных только между 2 узлами.
  - Switch **+**: м.б. взаимодействие >1 непересекающихся пар узлов.
- Основной способ выполнения коммуникационных операций – **пакетный метод** (часто на основе протокола TCP/IP)

Оценка трудоемкости операции передачи данных между 2 узлами кластера

- **Подход 1**:
  - $t_н$ не зависит от объема данных,
  - $t_с$ не зависит от числа пакетов

$$t_{пд} = t_н + mt_к + t_с$$

Ограничения не соответствуют действительности

Оценка времени (трудоемкости) **неточна**

Оценка трудоемкости операции передачи данных между 2 узлами кластера

- **Подход 2**:
  - **Учитывается**
    - n - число пакетов, $n = m/(V_{max} - V_c)$
    - V_c - объем служебных данных в каждом пакете,
    - V_{max} - максимально возможный для сети размер пакета,
    - $tнач_0$ – аппаратная (сетевая) задержка (латентность),
    - $tнач_1$ – время подготовки к передаче в сети 1 байта.
  - **Предполагается**
    - Подготовка данных для 2,3, … пакетов совмещена с пересылкой предшествующих пакетов.
    - Нужно учитывать рост объема передаваемой информации за счет добавления служебных данных (заголовков пакетов)

Оценка трудоемкости операции передачи данных между 2 узлами кластера

- Подход 2 – **итоговое соотношение**

$$t_{пд} = \begin{cases} t_{нач_0} + m \cdot t_{нач_1} + (m + V_c) \cdot t_к, & n = 1 \\ t_{нач_0} + (V_{\max} - V_c) \cdot t_{нач_1} + (m + V_c \cdot n) \cdot t_к, & n > 1 \end{cases}$$

Оценка трудоемкости операции передачи данных между 2 узлами кластера

- **Подход 3** – ***модель Хокни*** (R.W. Hocney, 1994) – используется для грубых оценок трудоемкости

$$t_{пд} = t_н + mt_к = t_н + m/R$$

- Оценки через вычислительные эксперименты на кластере:
  - $t_н$ - время передачи сообщения длины 0 для подходов 1 и 3,
  - $tнач_0$, $tнач_1$ для подхода 2 - можно оценить через аппроксимацию $t_{пд}$ - времени передачи сообщений размером от 0 до V_{max}
  - $R = max\ (t_{пд} / m)$ при варьировании m