Политики выделения ресурсов на кластерах с сетью Ангара

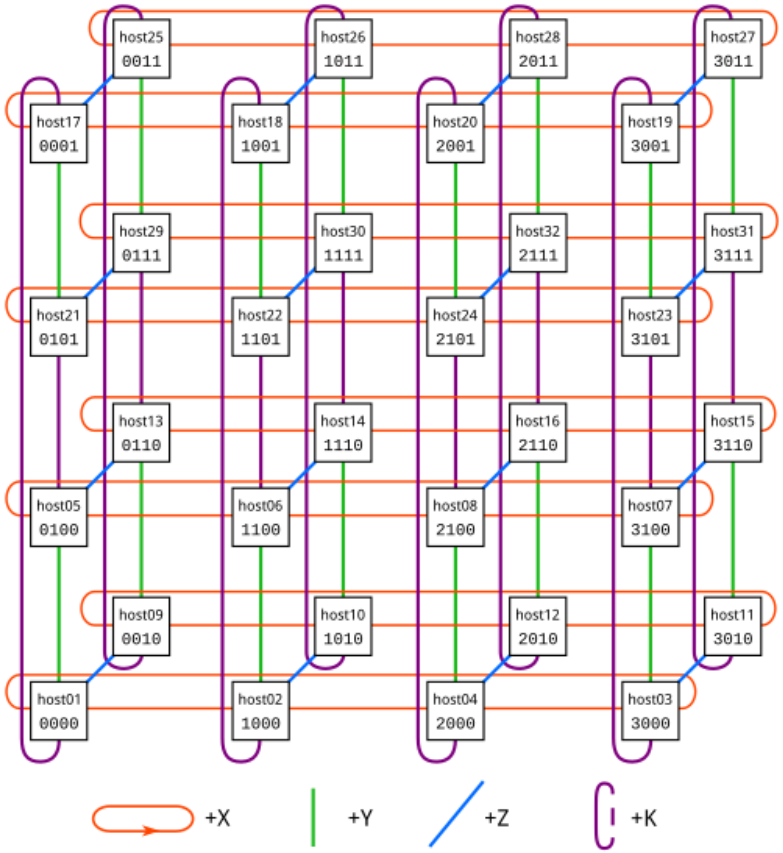
Мукосей А. В.
АО «НИЦЭВТ»

План доклада

- Сеть Ангара
  - Топология многомерный тор
  - Правила маршрутизации
- Текущий алгоритм выбора узлов
- Новый алгоритм выбора узлов
  - Динамическая таблица маршрутизации
  - Устойчивость алгоритма к возникающим отказам в кластере
  - Метод упаковки задания, сокращающий фрагментацию ресурсов
- Результаты

Топология сети Ангара

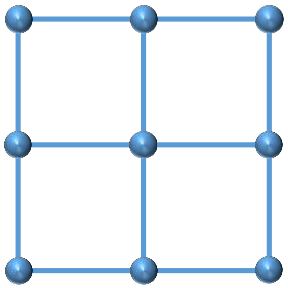
Топология сети Ангара
на кластере DESMOS
4D-тор 4x2x2x2



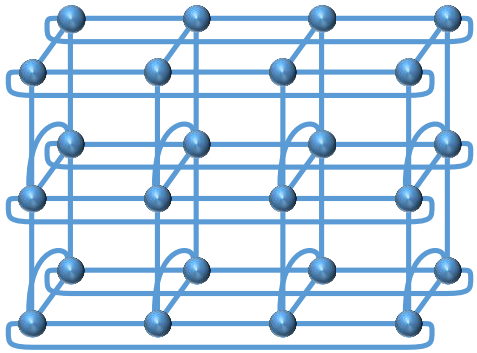
Топология сети – 4D-тор

- Каждый узел имеет до 8 соседей
- Маршрутизация пакетов подчиняется правилам маршрутизации

2D-тор 3x3



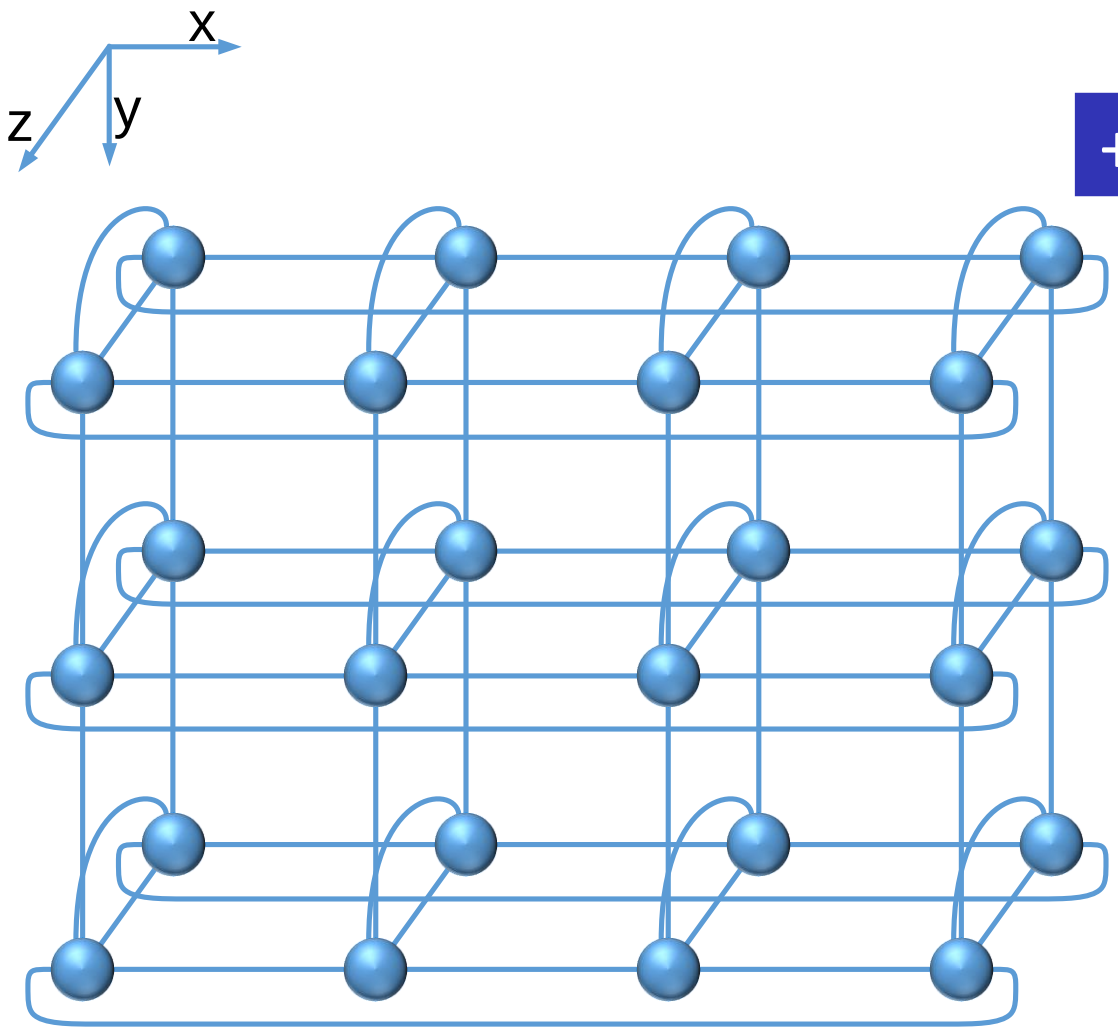
3D-тор 4x3x2



Маршрутизация в сети Ангара

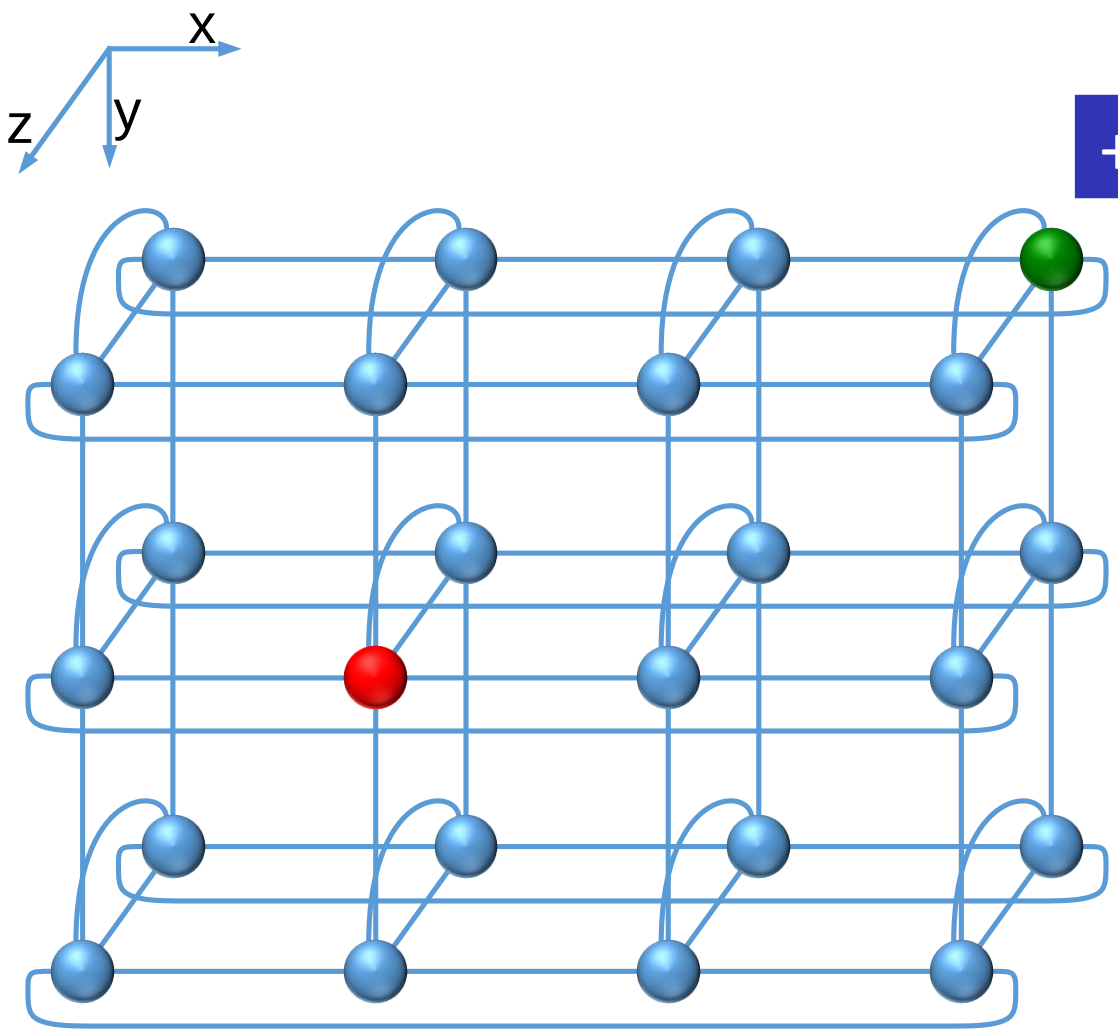
- Правило порядка направлений с использованием битов направлений
  - Маршрут пакета удовлетворяет заданному порядку направлений
  - Маршрут содержит движение только в одну сторону по измерению
- Метод First Step/Last Step
  - Добавляет первый и последний нестандартный шаг, удовлетворяющий правилу порядка направлений

Правило порядка направлений с использованием битов направлений



+x	+y	+z	-x	-y	-z

Правило порядка направлений с использованием битов направлений

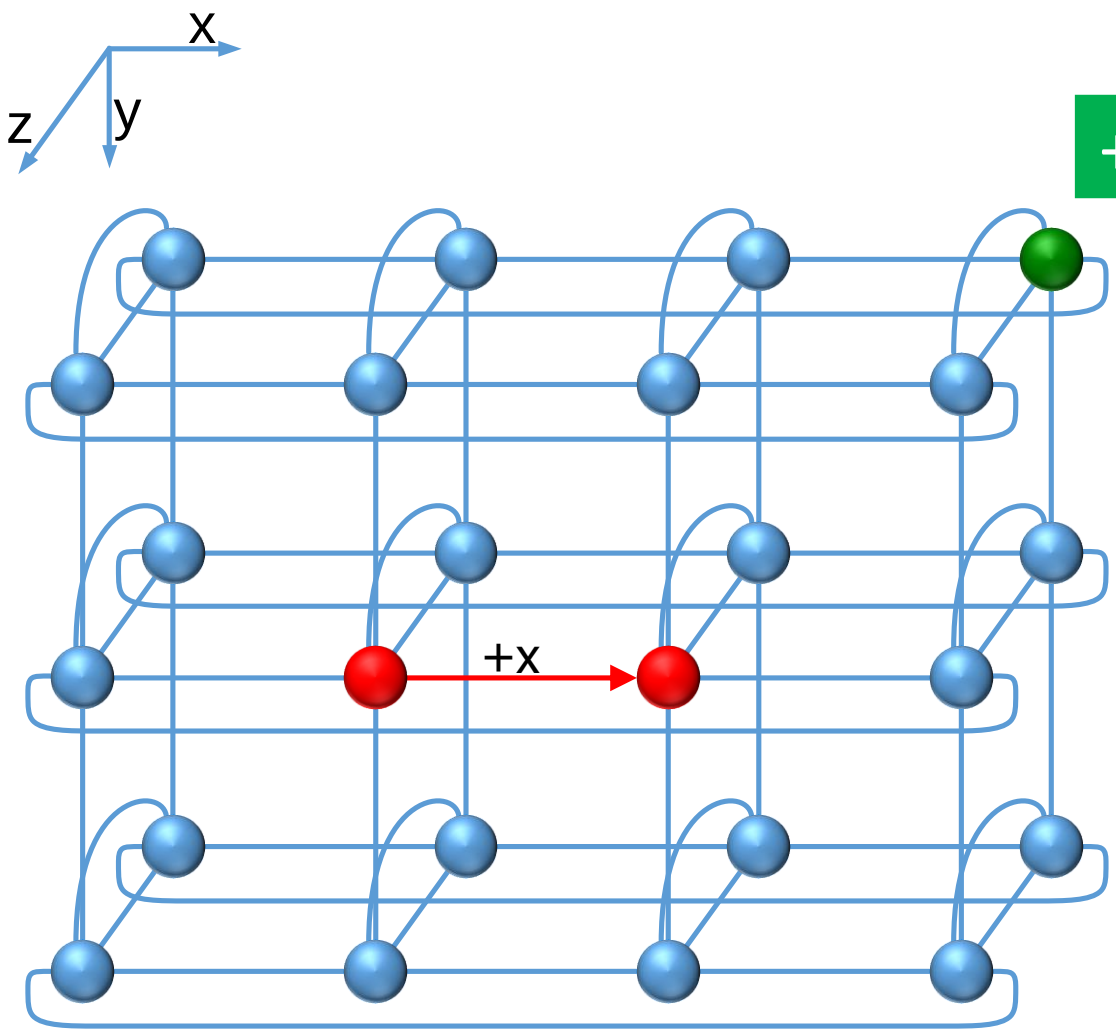


+x	+y	+z	-x	-y	-z

- Стартовый узел

- Конечный узел

Правило порядка направлений с использованием битов направлений

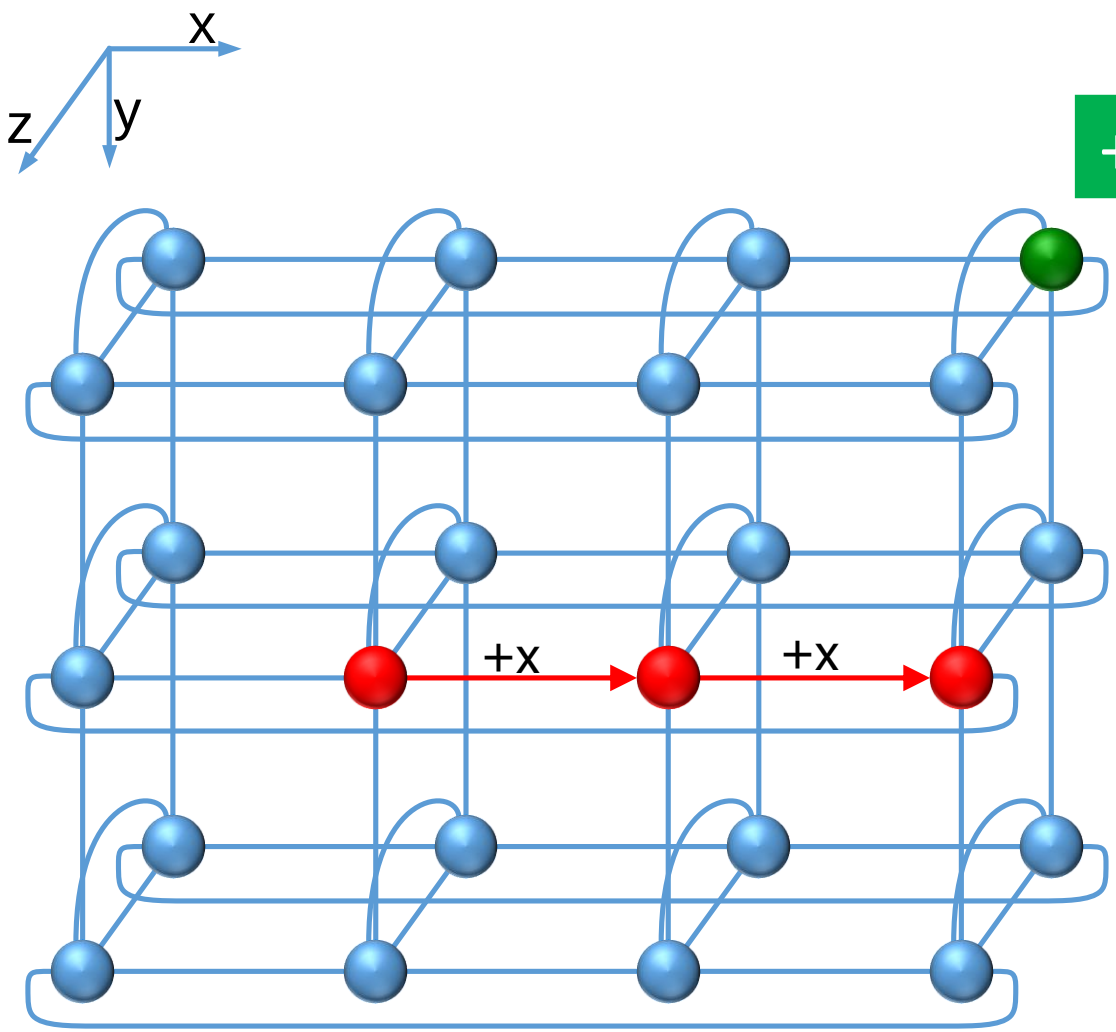


+x	+y	+z	-x	-y	-z

- Стартовый узел

- Конечный узел

Правило порядка направлений с использованием битов направлений

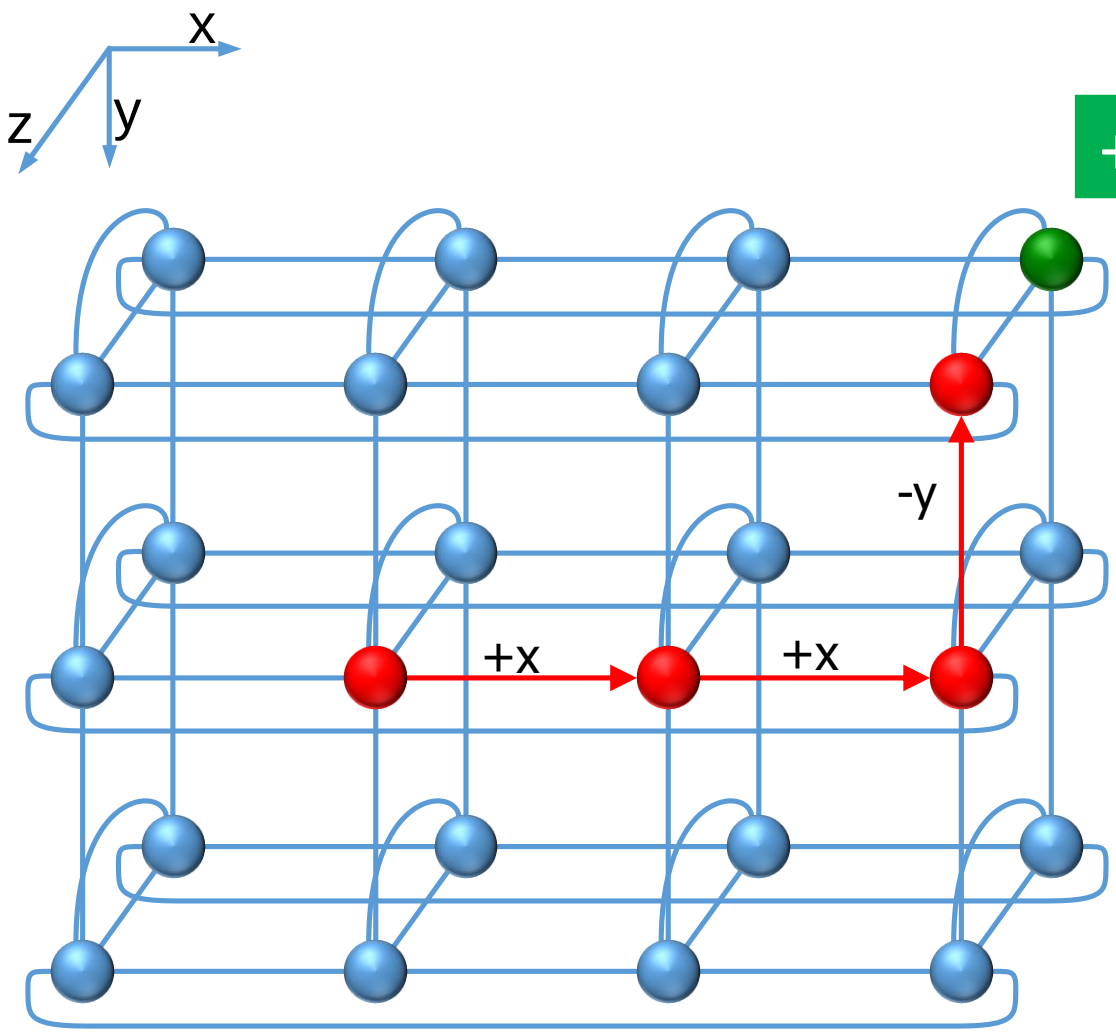


+x	+y	+z	-x	-y	-z

- Стартовый узел

- Конечный узел

Правило порядка направлений с использованием битов направлений

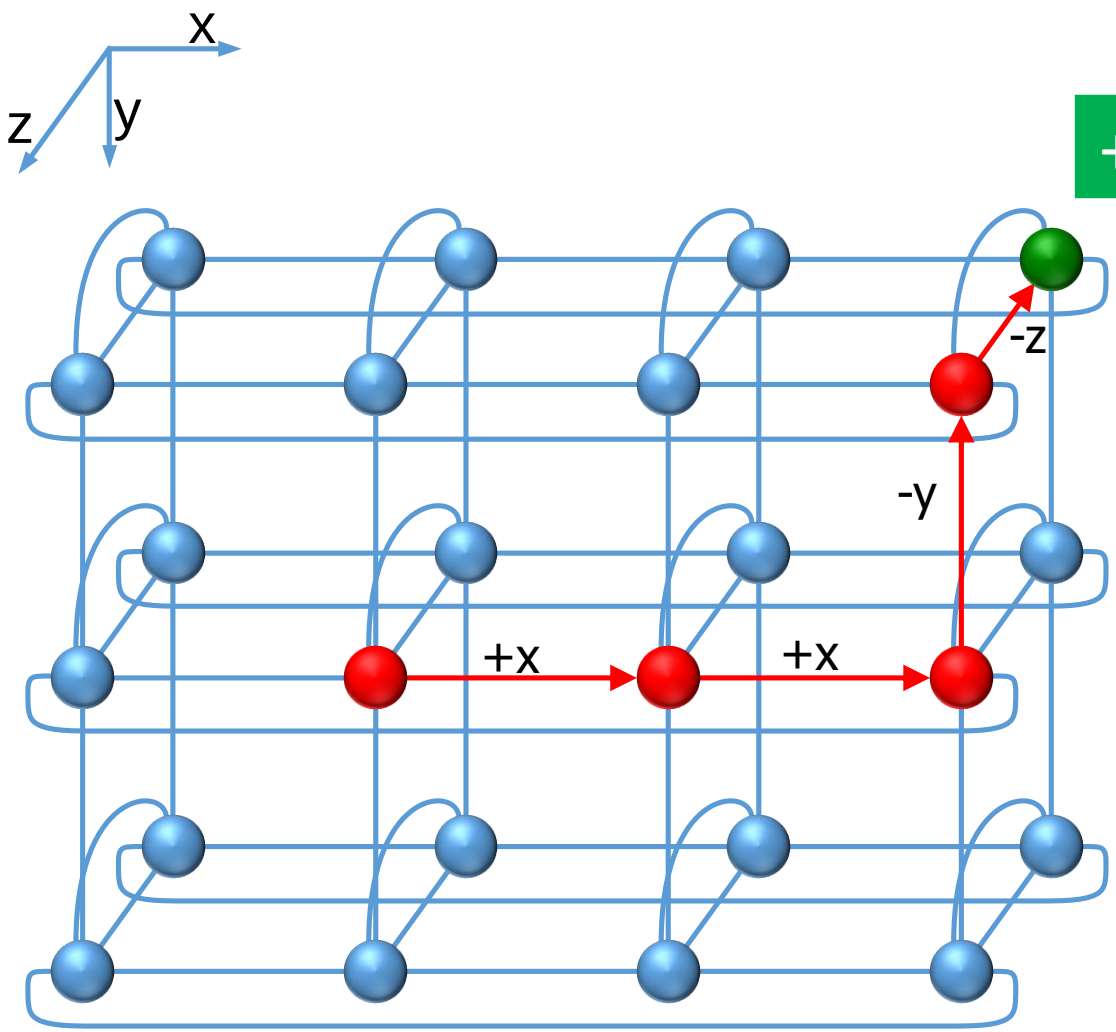


+x	+y	+z	-x	-y	-z

- Стартовый узел

- Конечный узел

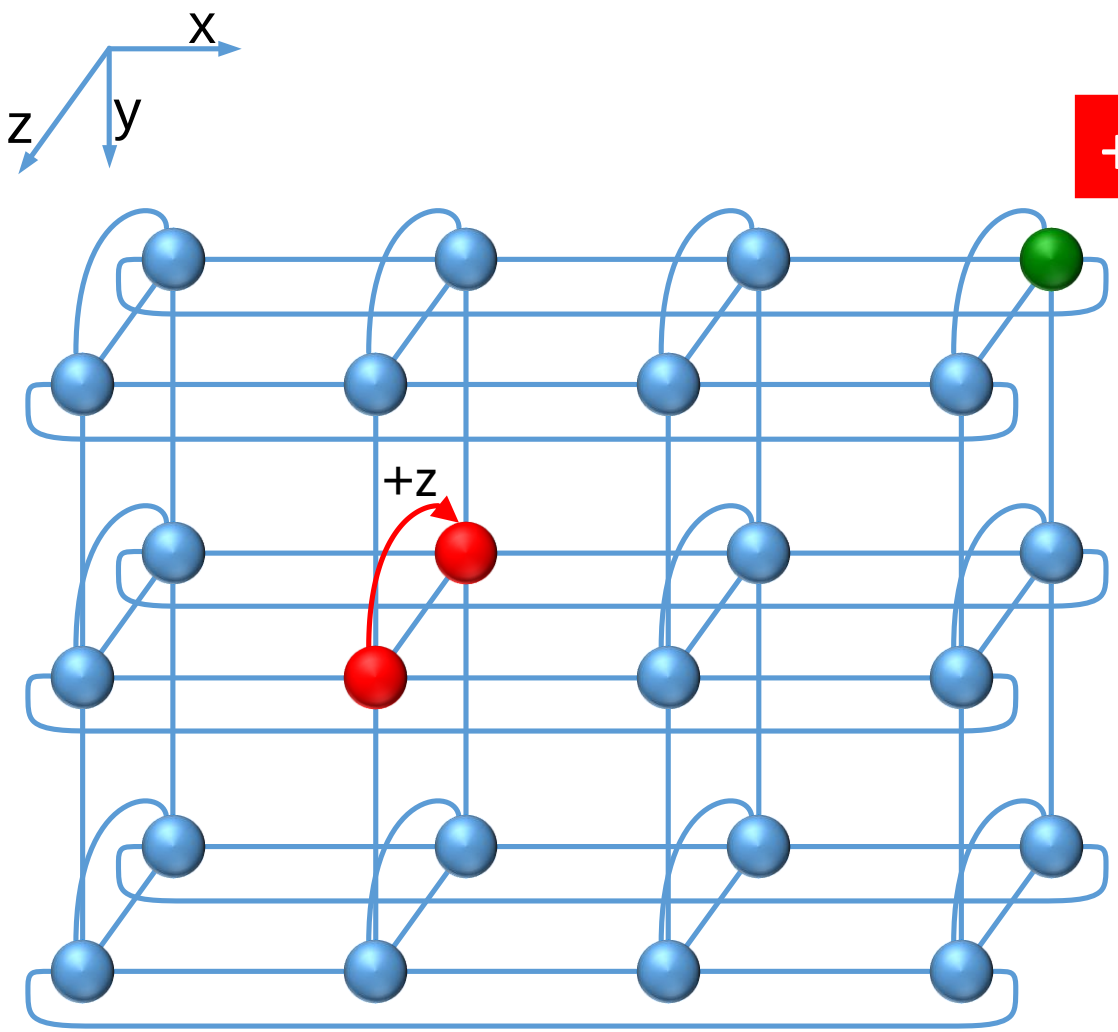
Правило порядка направлений с использованием битов направлений



+x	+y	+z	-x	-y	-z

- Стартовый узел
- Конечный узел

Правило порядка направлений с использованием битов направлений

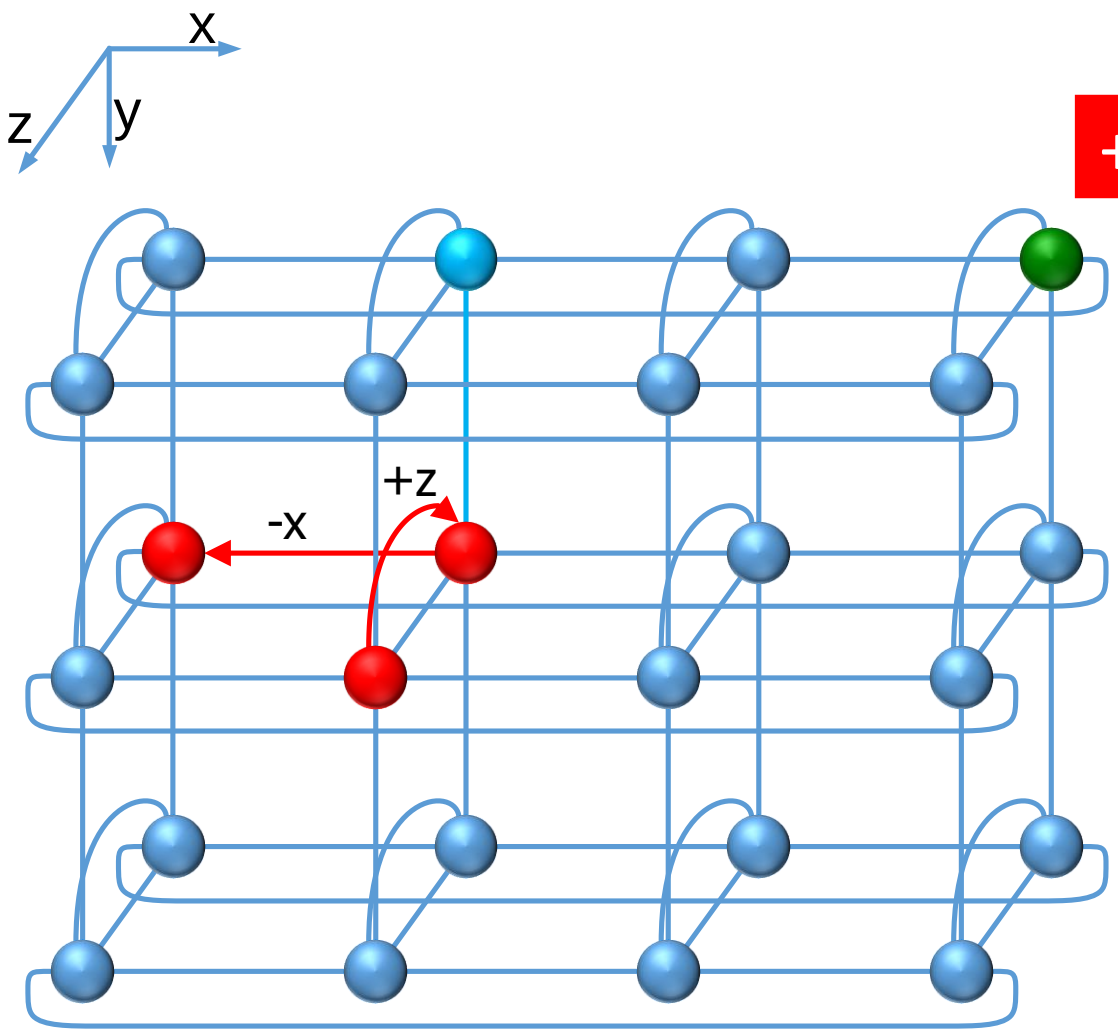


+x	+y	+z	-x	-y	-z

- Стартовый узел

- Конечный узел

Правило порядка направлений с использованием битов направлений

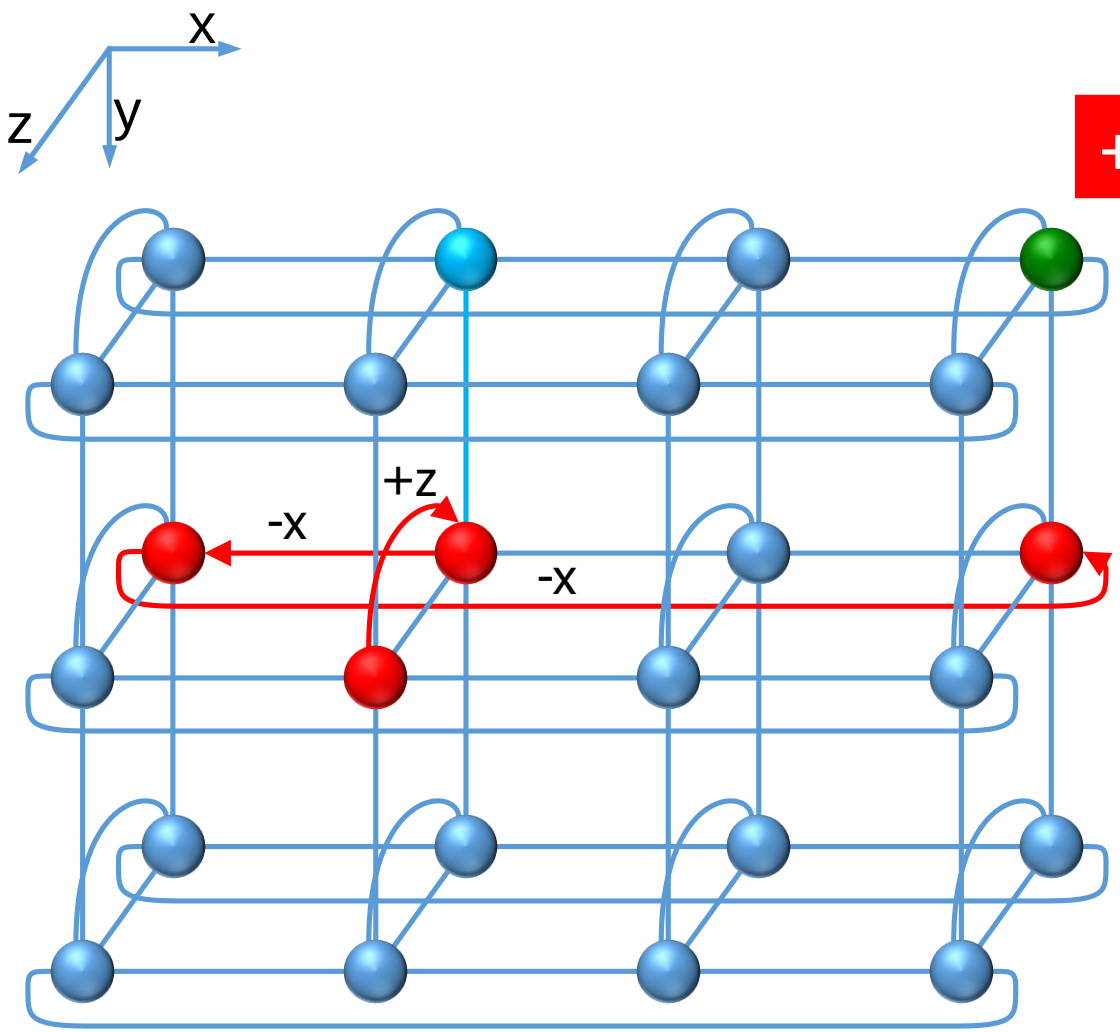


+x	+y	+z	-x	-y	-z

- Стартовый узел

- Конечный узел

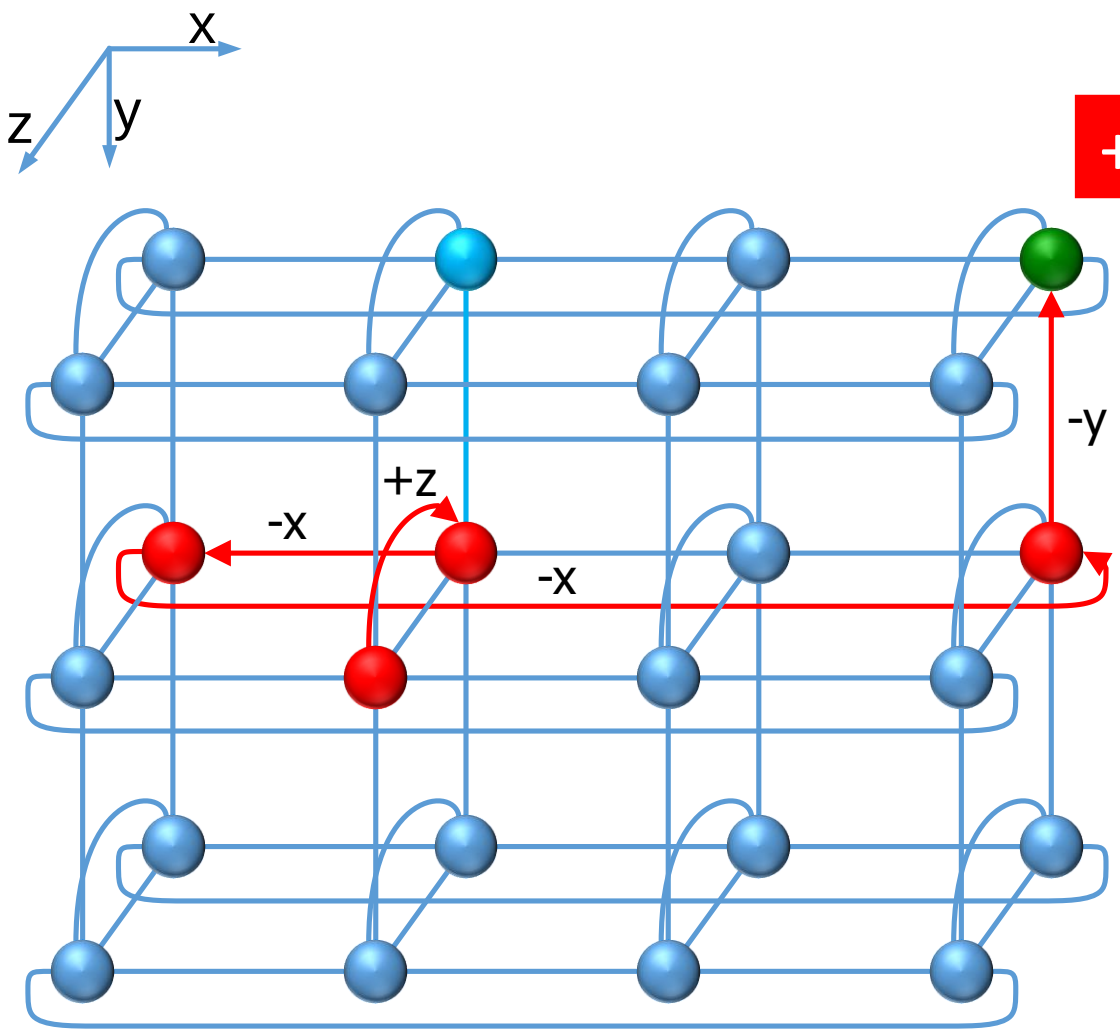
Правило порядка направлений с использованием битов направлений



+x	+y	+z	-x	-y	-z

- Стартовый узел
- Конечный узел

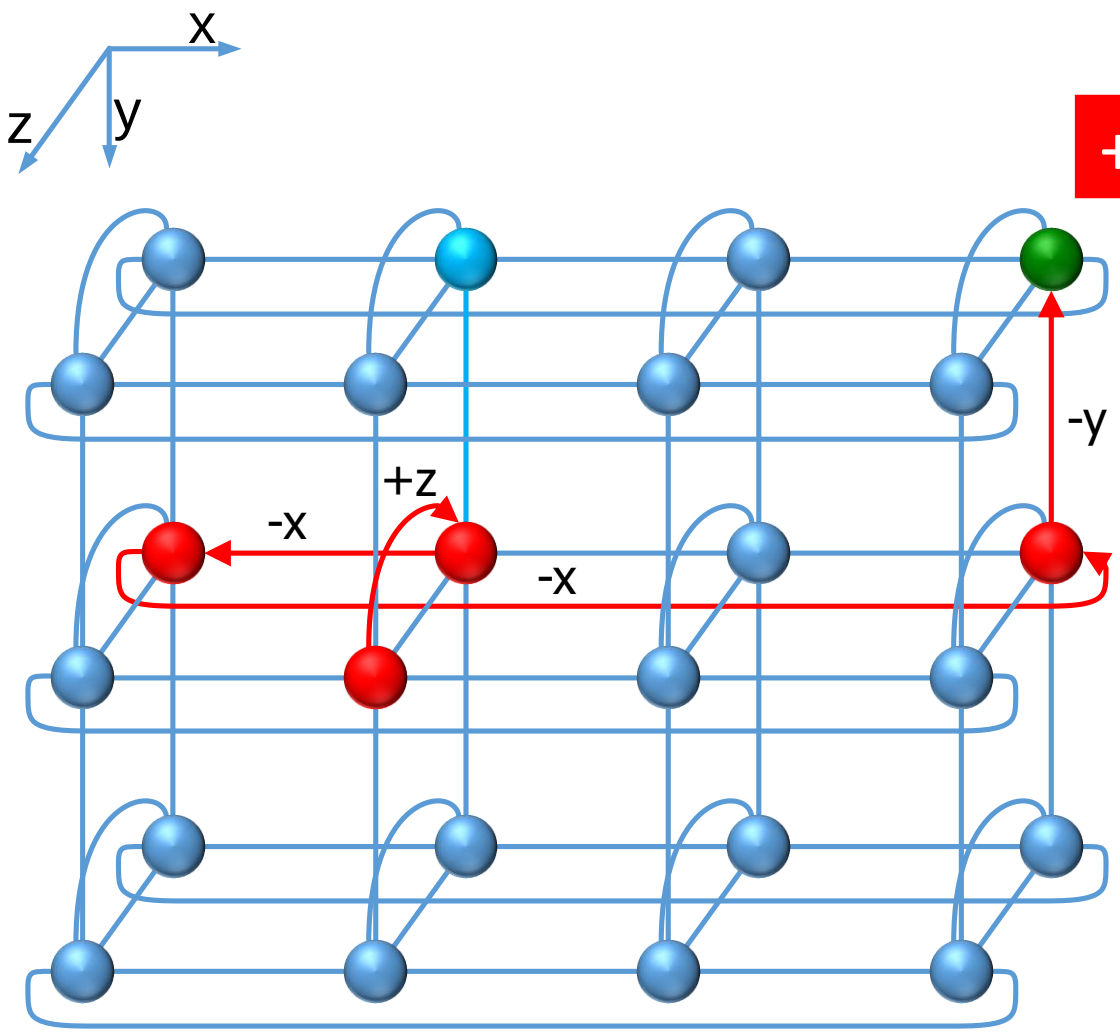
Правило порядка направлений с использованием битов направлений



+x	+y	+z	-x	-y	-z

- Стартовый узел
- Конечный узел

Правило порядка направлений с использованием битов направлений



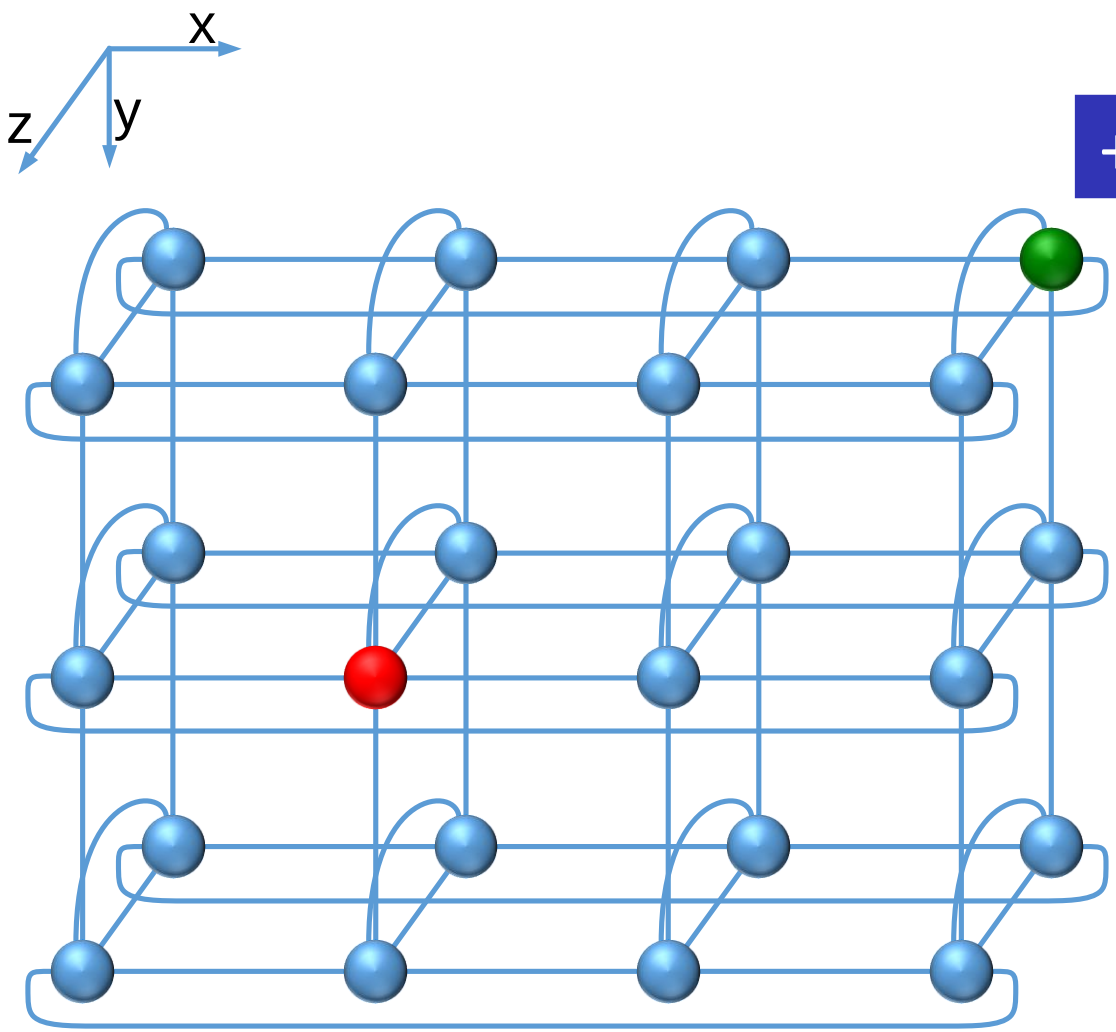
+x	+y	+z	-x	-y	-z

- Стартовый узел

- Конечный узел

Всего путей не более 2^n, где n – размерность системы

Правило порядка направлений с использованием битов направлений

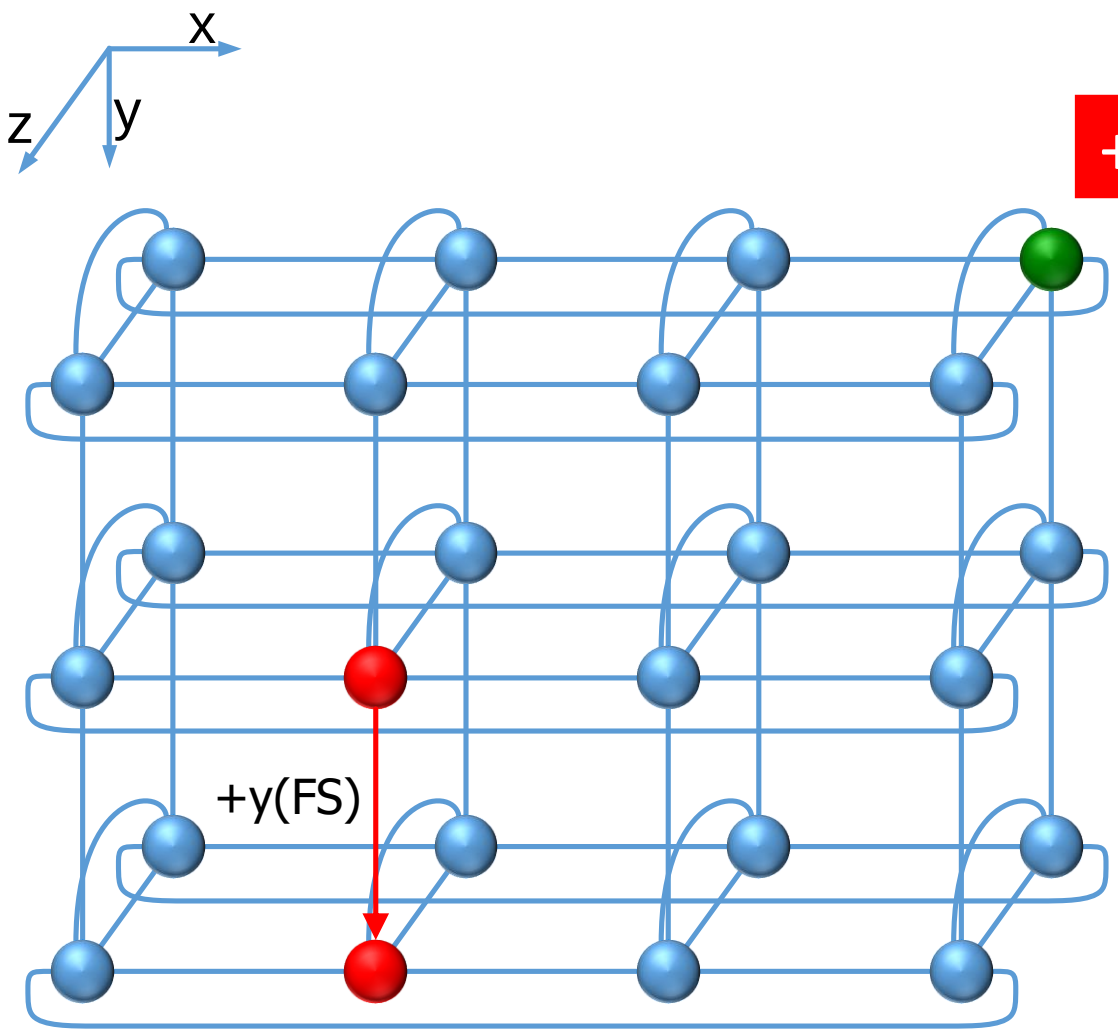


+x	+y	+z	-x	-y	-z

- Стартовый узел

- Конечный узел

Правило порядка направлений с использованием битов направлений

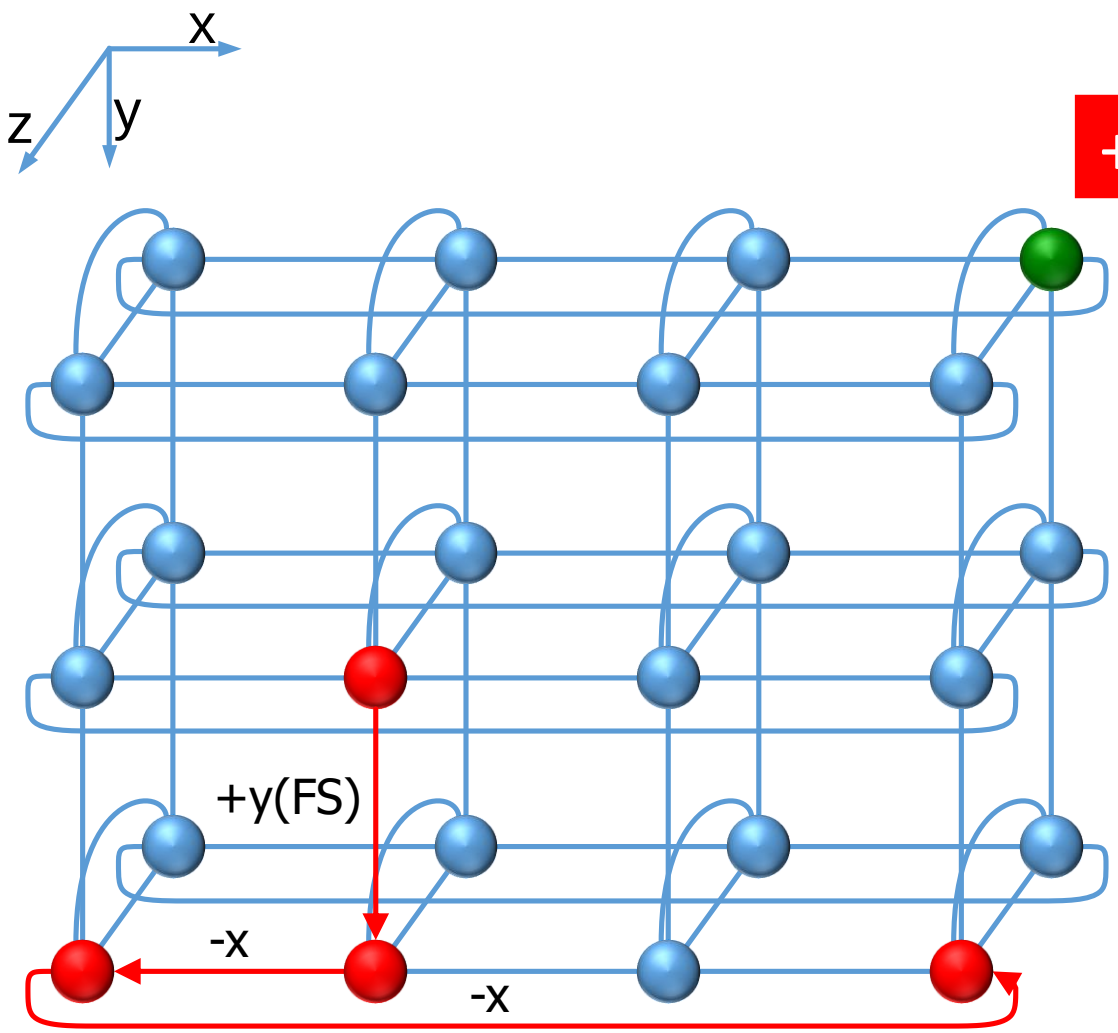


+x	+y	+z	-x	-y	-z

- Стартовый узел

- Конечный узел

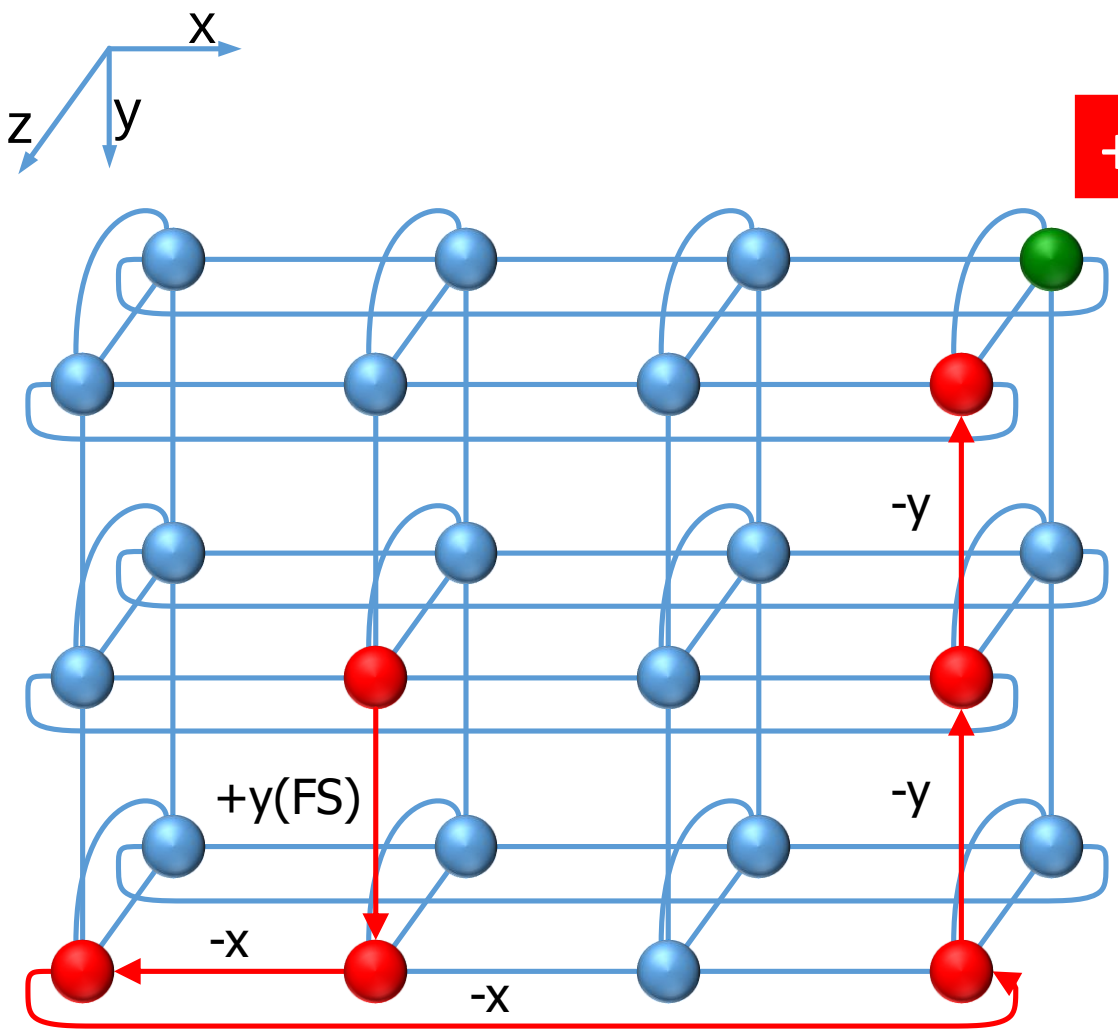
Правило порядка направлений с использованием битов направлений



+x	+y	+z	-x	-y	-z

- Стартовый узел
- Конечный узел

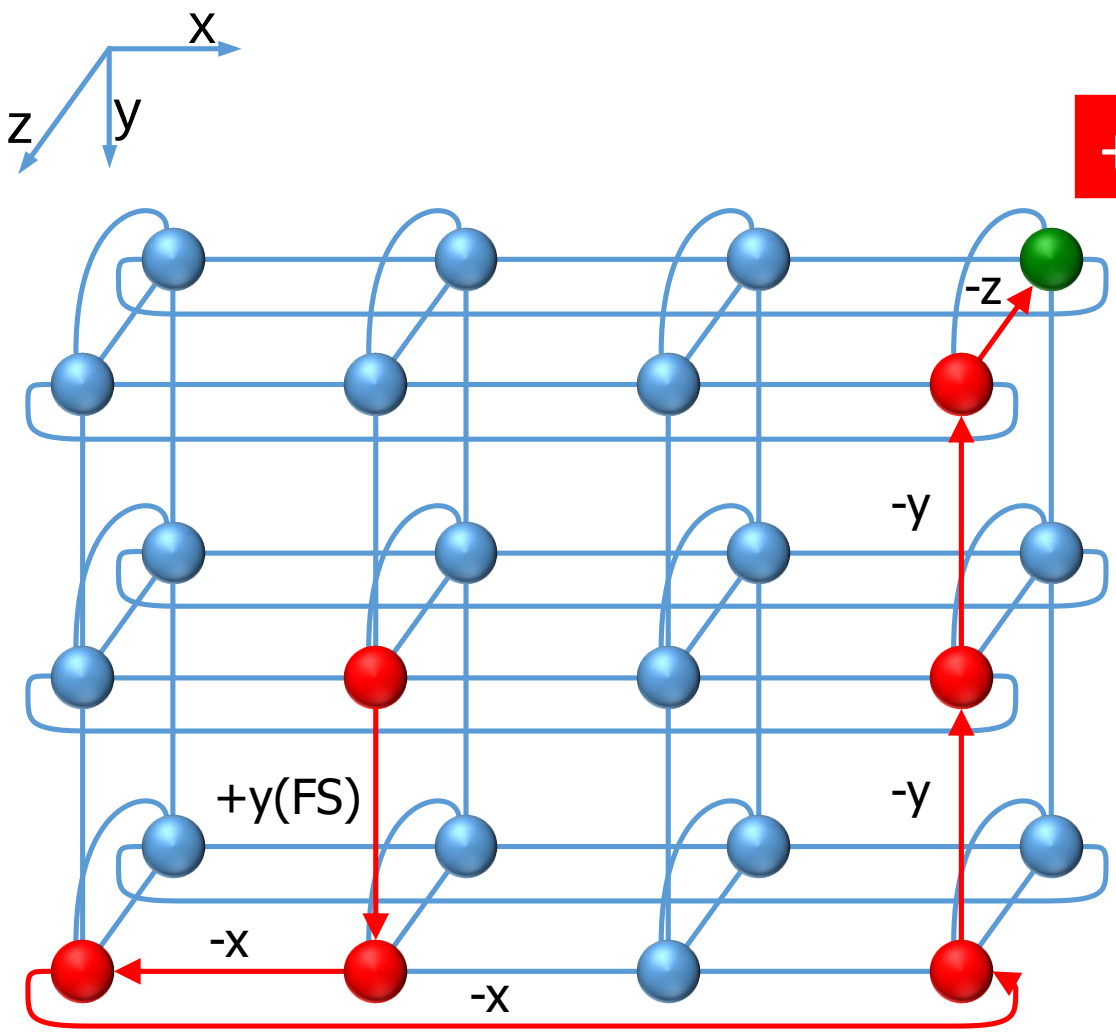
Правило порядка направлений с использованием битов направлений



+x	+y	+z	-x	-y	-z

- Стартовый узел
- Конечный узел

Правило порядка направлений с использованием битов направлений



+x	+y	+z	-x	-y	-z

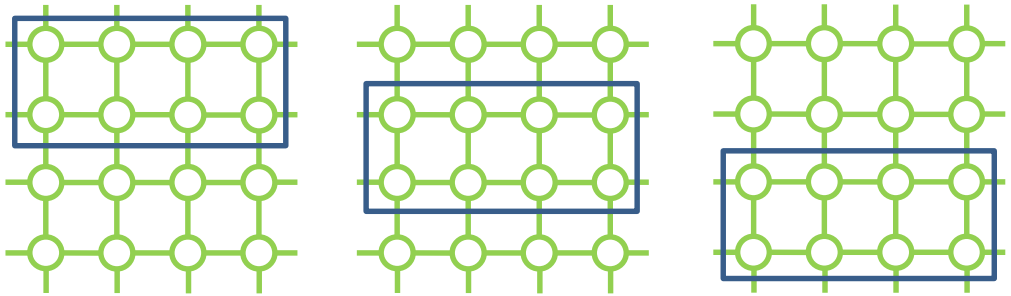
- Стартовый узел
- Конечный узел

Постановка задачи выбора узлов

- Необходимо выбрать требуемое число узлов из доступных, если такое возможно
  - Выбранные узлы, должны быть *маршрутизируемы*
  - Пути между этими узлами, не должны пересекаться с путями из других множеств узлов

Текущий алгоритм выбора узлов

- Для требуемого числа узлов *m* и допустимого числа транзитных узлов *t* строятся всевозможные разложения чисел *m,m+1,…,m+t* на *n* множителей, таких что $1 \le m_i \le d_i$. Где d_i длина измерения.
  - Для системы 4x4 для требуемого числа узлов 8 и допустимого числа транзитных узлов 1 будут получены следующие разложения: 2x4, 4x2, 3x3.
- Найденные разложения сортируются по среднему диаметру.
  - Средний диаметр – это средняя длина пути.
- По найденным разложениям ищутся все возможные наборы узлов, которые можно покрыть этими разложениями



Недостатки текущего алгоритма

- *Таблица маршрутизации* фиксирована
  - При выходе из строя узла или линка пропадает связность всех узлов использующих этот узел или линк для маршрута
- *Таблица маршрутизации* – набор путей между каждой парой узлов в кластере, по одному пути на пару узлов.

Новый алгоритм выбора узлов

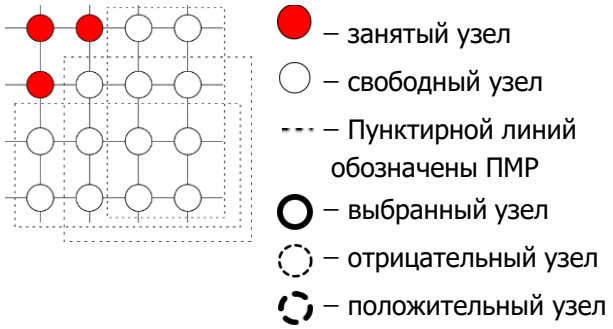
- Аналогично предыдущему, строит различные разложения для требуемого числа узлов и допустимого числа транзитных узлов
- По найденным разложениям ищутся все возможные наборы узлов, в покрытии которых содержится как минимум необходимое число требуемых узлов.
  - В покрытии могут присутствовать как сломанные узлы, так и занятые
- Для всех найденных множеств проверяется связность – наличие как минимум 1 пути между любыми парами узлов этого множества
- Если множество маршрутизируемо, для него строится оптимальная таблица маршрутизации
- Из данного набора можно выбрать наилучшее решение
  - По наименьшему диаметру
  - По наименьшей максимальной загрузке
  - По наименьшему числу транзитных узлов

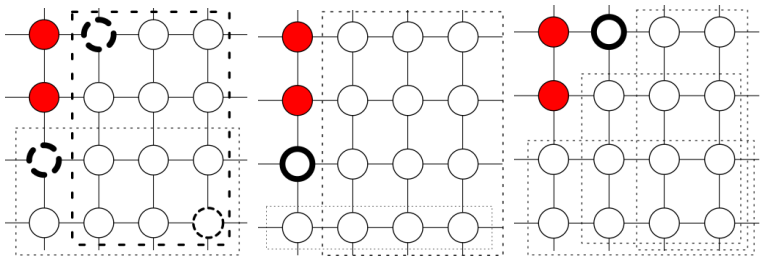
Достоинство нового метода

Благодаря динамической таблице маршрутизации новый алгоритм выделения узлов устойчив к возникающим отказам в кластере и может найти больше решений

Алгоритм упаковки

- Для сокращения фрагментации системы в результате работы кластера был разработан алгоритм упаковки.
  - Было введено понятие прямоугольника максимального размера (ПМР)
- В текущем состояние кластера ищутся все ПМР
- В порядке возрастания размера ПМР в них ищутся все решения из предыдущего слайда
- При *примерке* решения оцениваются:
  - получившиеся в результате ПМР
  - расстояние от решения до ПМР, к которому его примерили
- При нахождении первого подходящего решения алгоритм завершает свою работу





Алгоритм упаковки

- На втором этапе оптимизации фрагментации изменен порядок упаковки заданий
- Допустим есть набор заданий, требующих ресурсов одновременно, и общее число требуемых узлов не превышает число узлов доступных для выделения
- Алгоритм пробует различные порядки упаковки заданий
  - результатом является оптимальное выделение узлов
- Так как не все задания могут быть выделены, алгоритму позволяется менять порядок упаковки, который не приведет к изменению порядка очереди

Исследование. Симулятор

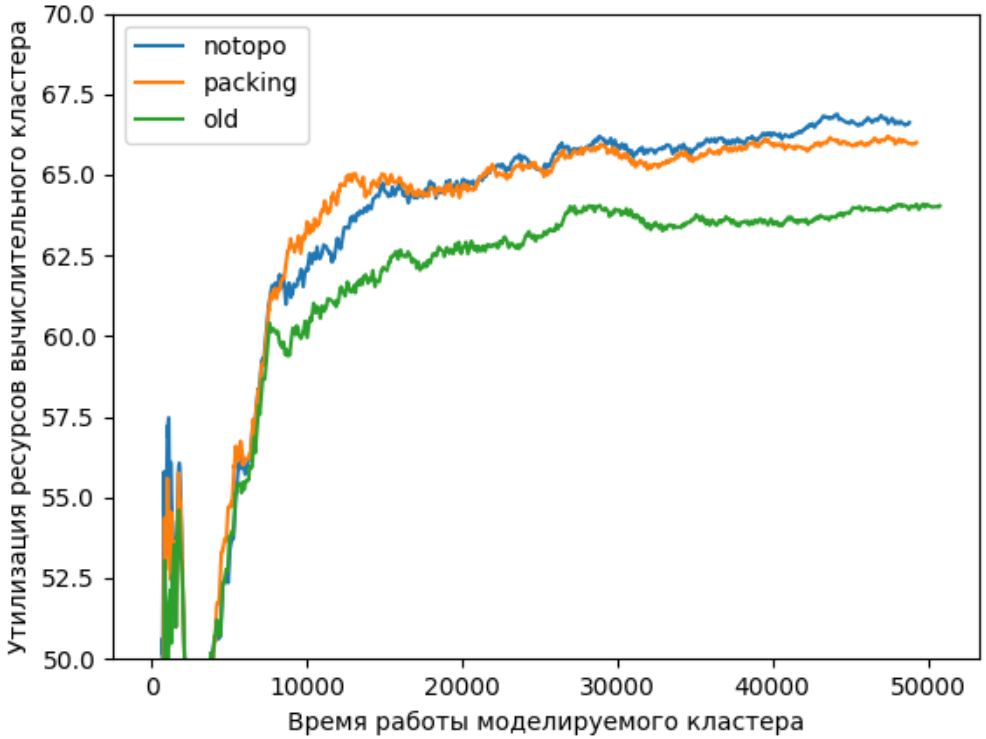
- Исследования проводились на симуляторе вычислительной системы
  - На вход симулятору подается набор заданий с параметрами:
    - время постановки задания в очередь;
    - требуемое число узлов;
    - требуемое время выделения узлов
  - Симулятор по очереди берет задания и отправляет их на обработку менеджеру ресурсов

Исследование. Метрики

- Для оценки качества алгоритмов выделения узлов предложены следующие метрики:
  - Утилизация вычислительного кластера: $U = \frac{\sum_{i=1}^{N} \frac{T_i}{T}}{N}$, где T_i – полное время выделения *i*-го узла, *T* – полное время симуляции, *N* – число узлов в кластере
  - Оценка ожидания задания в очереди: $D = \frac{\sum_{i=1}^{k} \frac{Q_i}{W_i}}{k}$, где Q_i – время ожидания задания в очереди, W_i – требуемое время выделения узлов для *i*-го задания, *k* – число заданий в очереди

Моделирование кластера 4x2x2x2

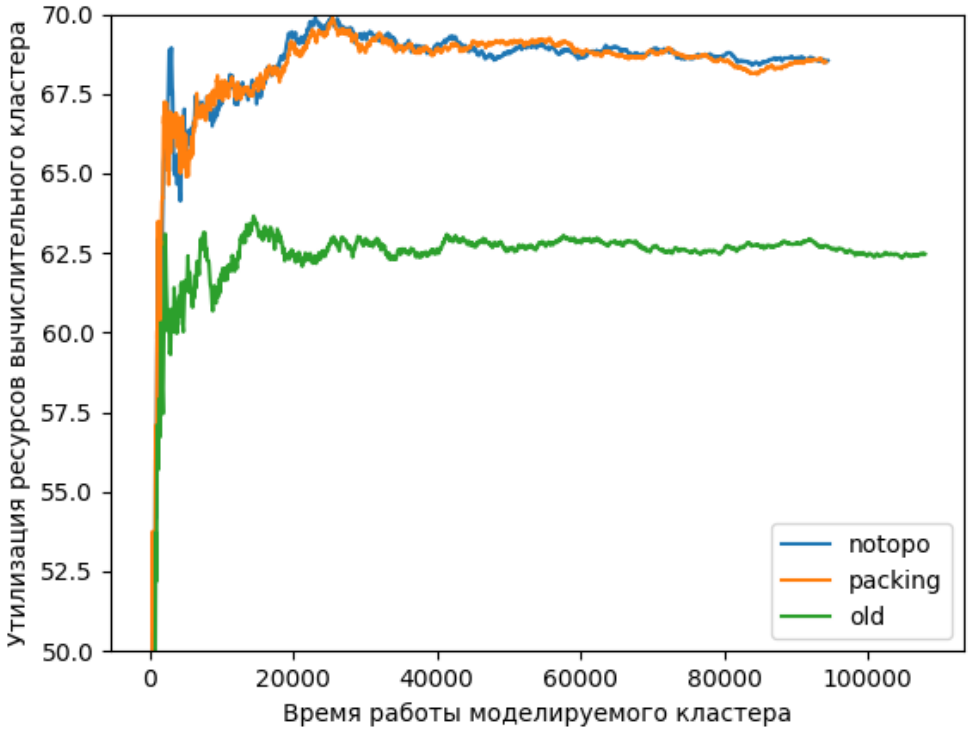
- Процентное соотношение задач в очереди
  - 1 – 25%
  - 2 – 20%
  - 4 – 15%
  - 8 – 15%
  - 16 – 15%
  - 32 – 10%
- Среднее время задания:
  - 150 сек.
- Общее число заданий:
  - 1311
- Оценка ожидания задания в очереди:
  - Packing – 40,91
  - Old – 49,73
- Средняя утилизация
  - Packing – 66%
  - Old – 64%



- Время обработки всех заданий
  - Packing – 49272 сек.
  - Old – 50782 сек.

Моделирование кластера 4x3x3

- Процентное соотношение задач в очереди
  - 1 – 5%
  - 2 – 20%
  - 4 – 20%
  - 8 – 25%
  - 16 – 15%
  - 32 – 10%
  - 36 – 5%
- Среднее время задания:
  - 150 сек.
- Общее число заданий:
  - 2244
- Оценка ожидания задания в очереди:
  - Packing – 79,65
  - Old – 157,25
- Средняя утилизация
  - Packing – 68,5%
  - Old – 62,5%



- Время обработки всех заданий
  - Packing – 94209 сек.
  - Old – 108011 сек.

Результаты

- Разработан новый алгоритм предоставления ресурсов с сетью Ангара
- По предварительным результатам алгоритм показывает улучшение:
  - На 2% увеличилась загрузка кластера для системы 4x2x2x2 и на 6% – для 4x3x3
  - На 20% уменьшилась оценка ожидания задания в очереди для системы 4x2x2x2 и на 49% – для 4x3x3
  - На 3% уменьшилось полное время работы симулируемого кластера для системы 4x2x2x2 и на 13% – для 4x3x3

Спасибо за внимание!