

Беспроводные сети и мобильные системы

Лекция 8

Мобильные сети: IP маршрутизация и алгоритмы MANET

Задачи лекции

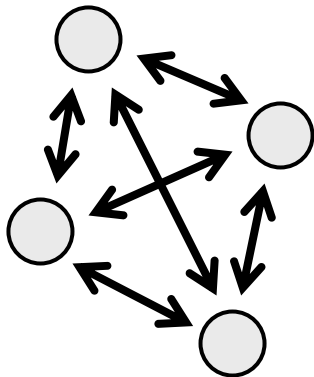
- Представить основные принципы маршрутизации в сетях с коммутацией пакетов
- Описать основные характеристики специализированных мобильных сетей (MANET) и их протоколы маршрутизации
- Описать AODV и OLSR в качестве примера протоколов маршрутизации MANET
- Обсудить публикации, посвященные специализированным мобильным сетям и протоколам маршрутизации MANET

План

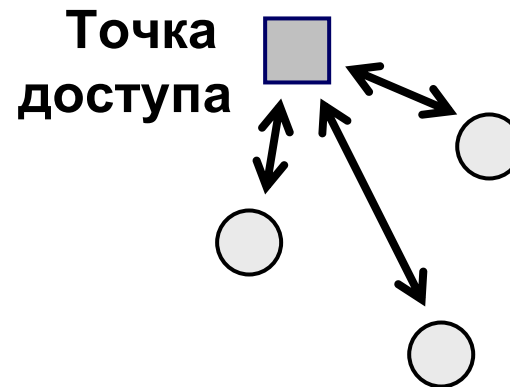
- **Маршрутизация 2 уровня**
- **Основы маршрутизации**
 - Алгоритмы вектора расстояния
 - Алгоритмы состояния связи
- **Специализированные мобильные сети**
- **Примеры протоколов маршрутизации MANET**
 - OLSR
 - AODV

До настоящего времени...

- Узлы базового набора 802.11 или сети Bluetooth соединены непосредственно друг с другом
- В маршрутизации нет необходимости, и IP (уровень 3) практически не работает



Режим Ad Hoc

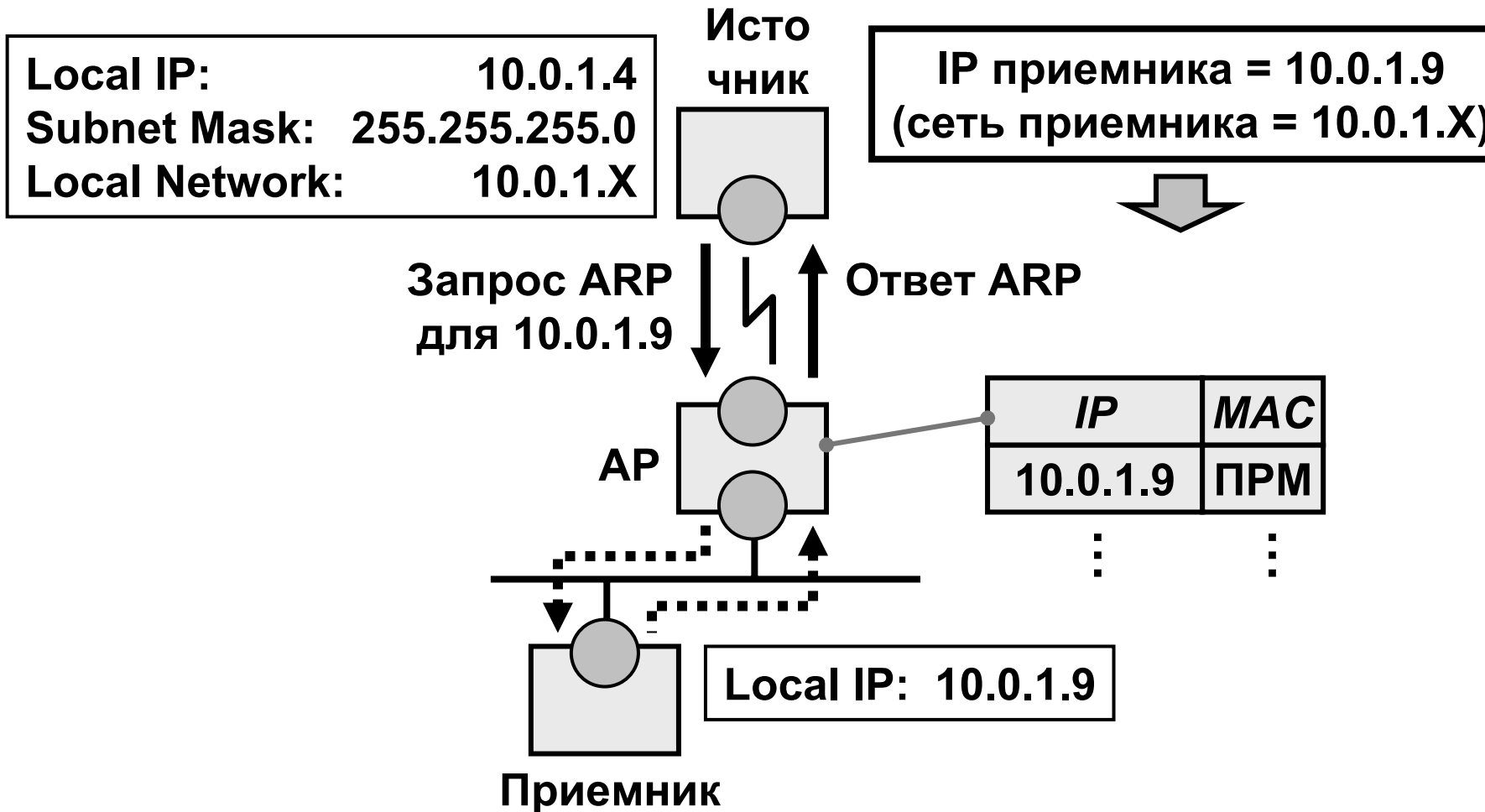


Инфраструктурный режим

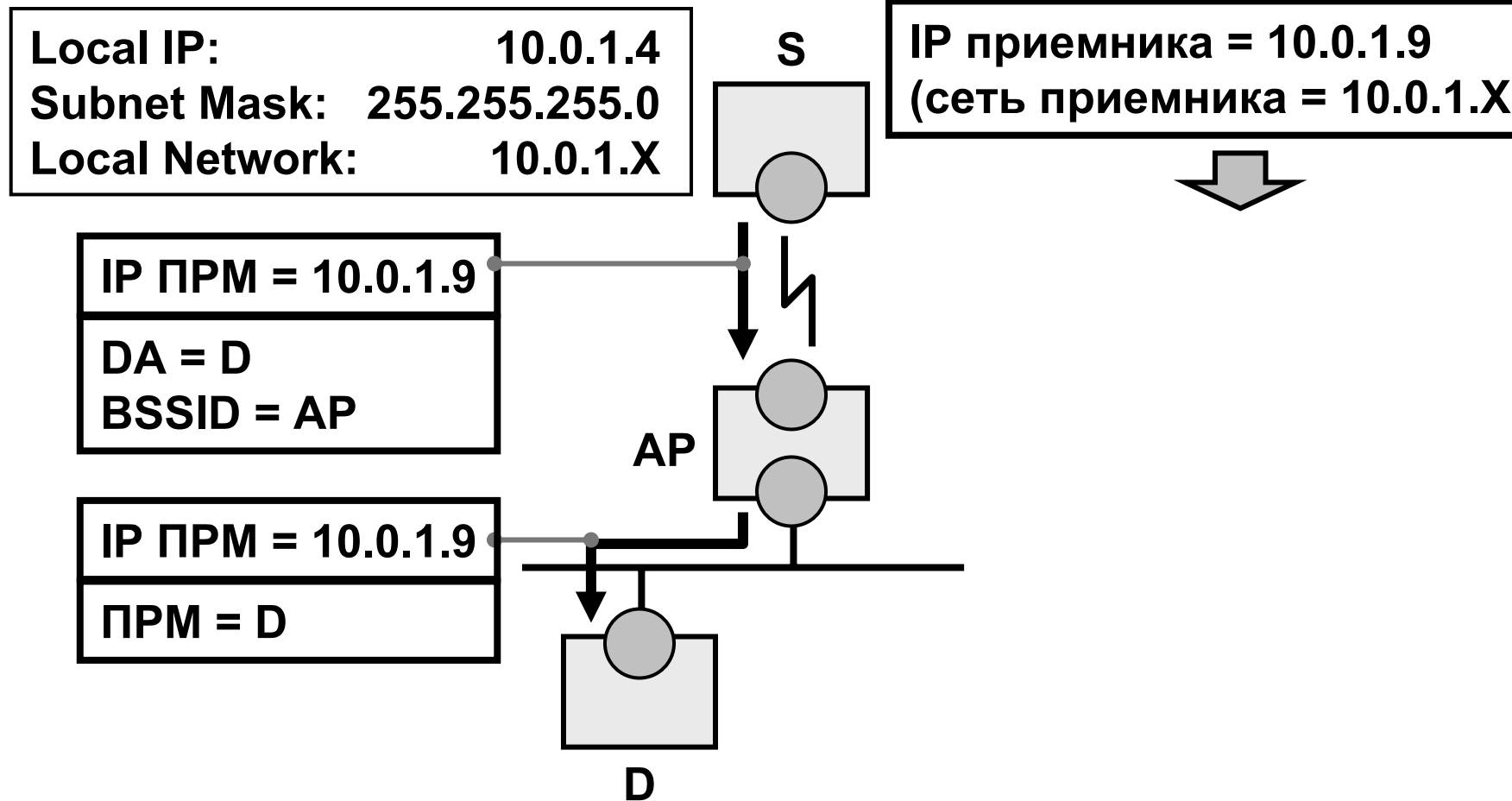
Маршрутизация на уровне 2 (1)

- **Источник определяет, что приемник находится в той же самой подсети IP**
 - При этом обязательно наличие прямой связи между источником и приемником на 2 сетевом уровне
- **ARP позволяет источнику определить адрес (MAC) 2 уровня приемника**
- **Источник капсулирует датаграмму IP во фрейме 2 уровня, соответствующим образом адресует фрейм и передает его**
- **«Межсетевое устройство» уровня 2 (например, мост Ethernet , 802.11 AP) может выполнить некоторые функции перенаправления и маршрутизации**

Маршрутизация на уровне 2 (2)

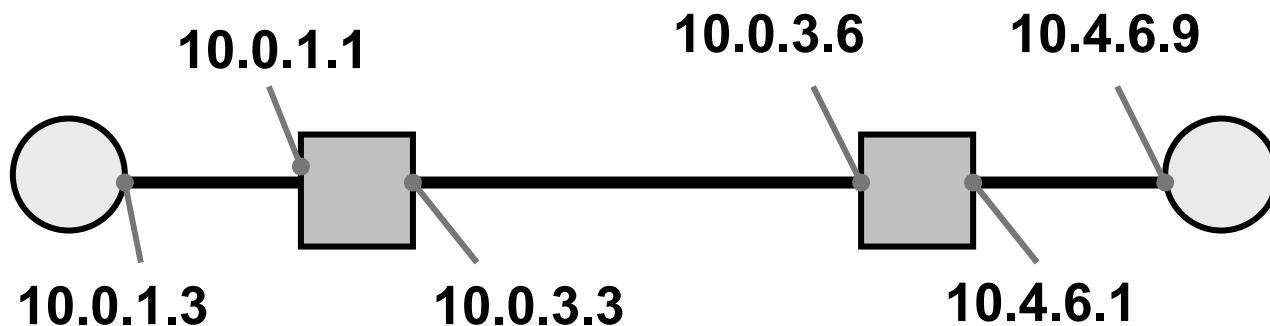


Маршрутизация на уровне 2 (3)



Необходимость маршрутизации на уровне 3

- Конечно узлы могут не иметь соединения на уровне 2
 - Узлы, которые находятся в разных подсетях IP, т.е., IP сети приемника отличается от IP локальной сети
 - Узлы, которые находятся за пределами радиодиапазона беспроводной сети
- В данном случае необходима маршрутизация на уровне 3



Маршрутизация

- **Маршрутизация состоит из двух основных шагов**
 - Направление пакетов на следующий луч (от входного к выходному интерфейсу в традиционной проводной сети)
 - Определение того, как направлять пакеты (построение таблицы или определение маршрута)
- **Легко направить пакеты, однако тяжело узнать куда (особенно сделать это эффективно)**
 - Найти приемник
 - Минимизировать количество лучей (длину пути)
 - Минимизировать задержку
 - Минимизировать потери пакетов
 - Минимизировать стоимость

Точка принятия решения

- **Маршрутизация источника**
 - Отправитель определяет маршрут и указывает его в заголовке пакета
 - Поддерживается в IP, хотя не является типичной схемой
- **Маршрутизация от луча к лучу (в датаграмме)**
 - Решение о дальнейшем маршруте принимается в каждой точке прохождения
 - Стандартная схема маршрутизации для IP
- **Создание виртуального маршрута**
 - Определение и конфигурирование пути до отправки первого пакета
 - Используется в АТМ (и системах, аналогичных голосовым телефонным)

Таблица маршрутизации

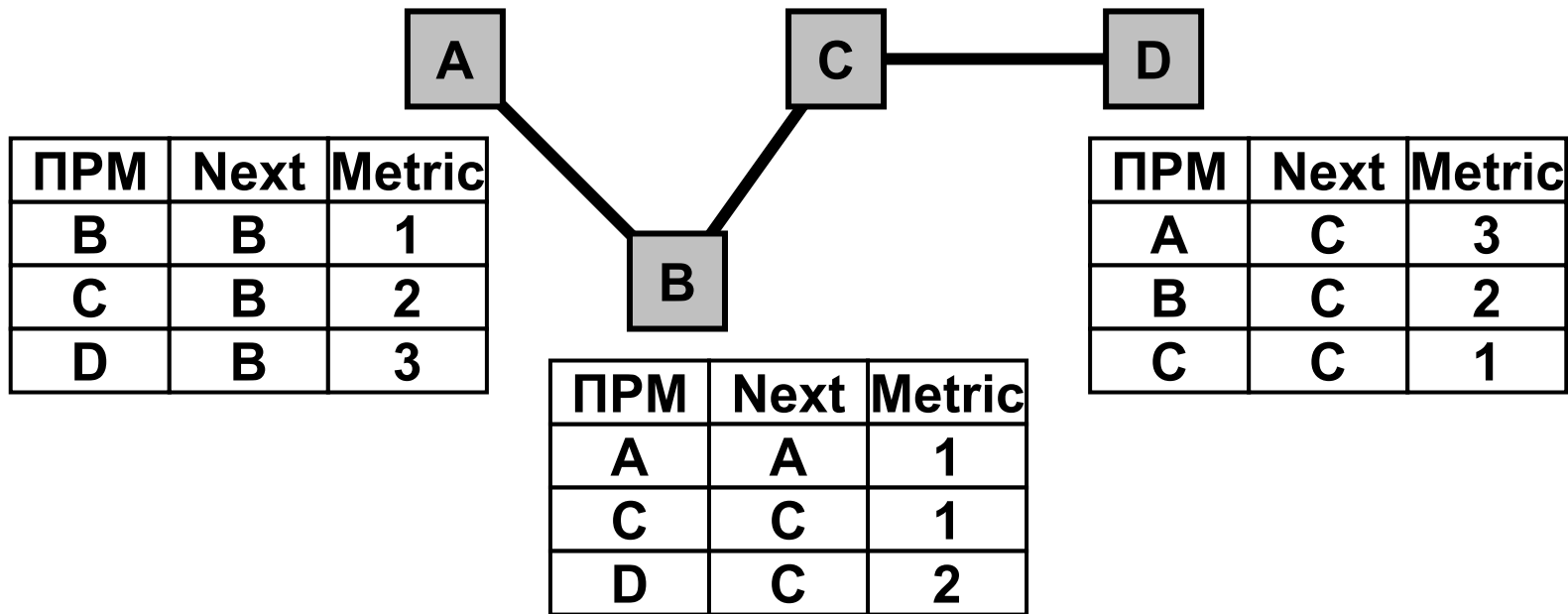
- **Таблица маршрутизации содержит информацию, позволяющую определить, как направлять пакеты**
 - **Маршрутизация источника:** Таблица маршрутизации используется для указания места назначения в пакете
 - **Маршрутизация от луча к лучу:** Таблица маршрутизации используется для поиска следующего луча в нужном направлении
 - **Создание виртуального маршрута:** Таблица маршрутизации используется для поиска пути через сеть
- **Распределенный алгоритм требует построения таблицы маршрутизации**
 - **Алгоритмы вектора расстояния**
 - **Алгоритмы состояния связи**

Алгоритмы вектора расстояния (1)

- **“Расстояние” для каждого соединения в сети является минимизируемой величиной**
 - Каждое соединение может иметь “расстояние” 1 для минимизации количества лучей
 - Алгоритм пытается минимизировать расстояние
- **Таблица маршрутизации в каждом узле...**
 - Указывает следующий луч для каждого приемника
 - Указывает расстояние до приемника
- **Соседи могут обмениваться информацией из своих таблиц для поиска маршрута (либо наилучшего маршрута) к приемнику**

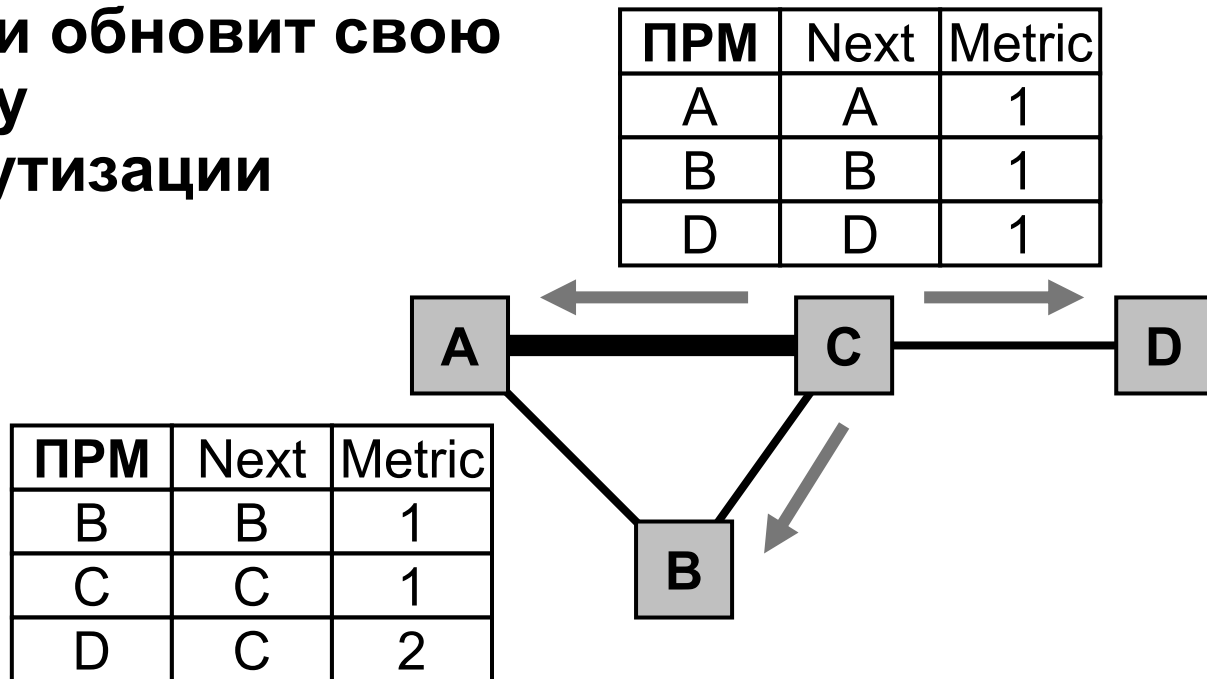
Алгоритмы вектора расстояния (2)

ПРМ	Next	Metric
A	B	2
B	B	1
D	D	1



Алгоритмы вектора расстояния (3)

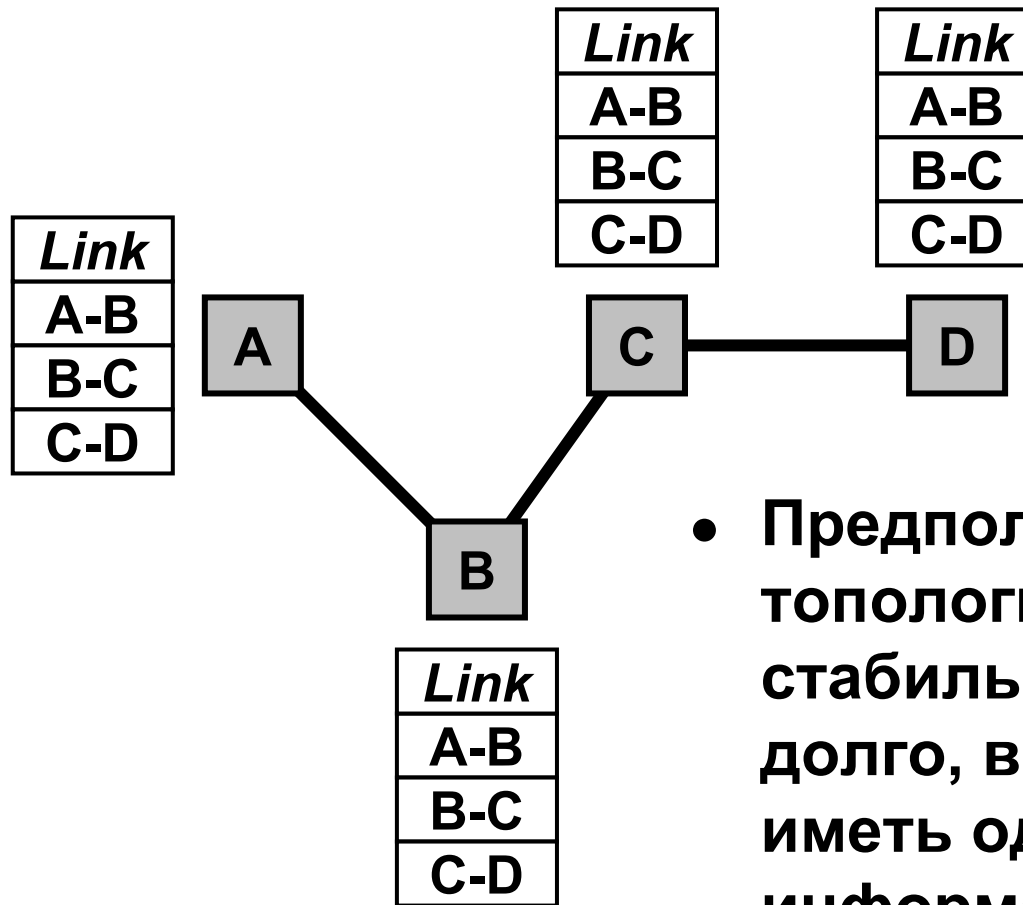
- Узел A узнает от узла C о кратчайшем пути к узлу D и обновит свою таблицу маршрутизации



Алгоритмы состояния связи (1)

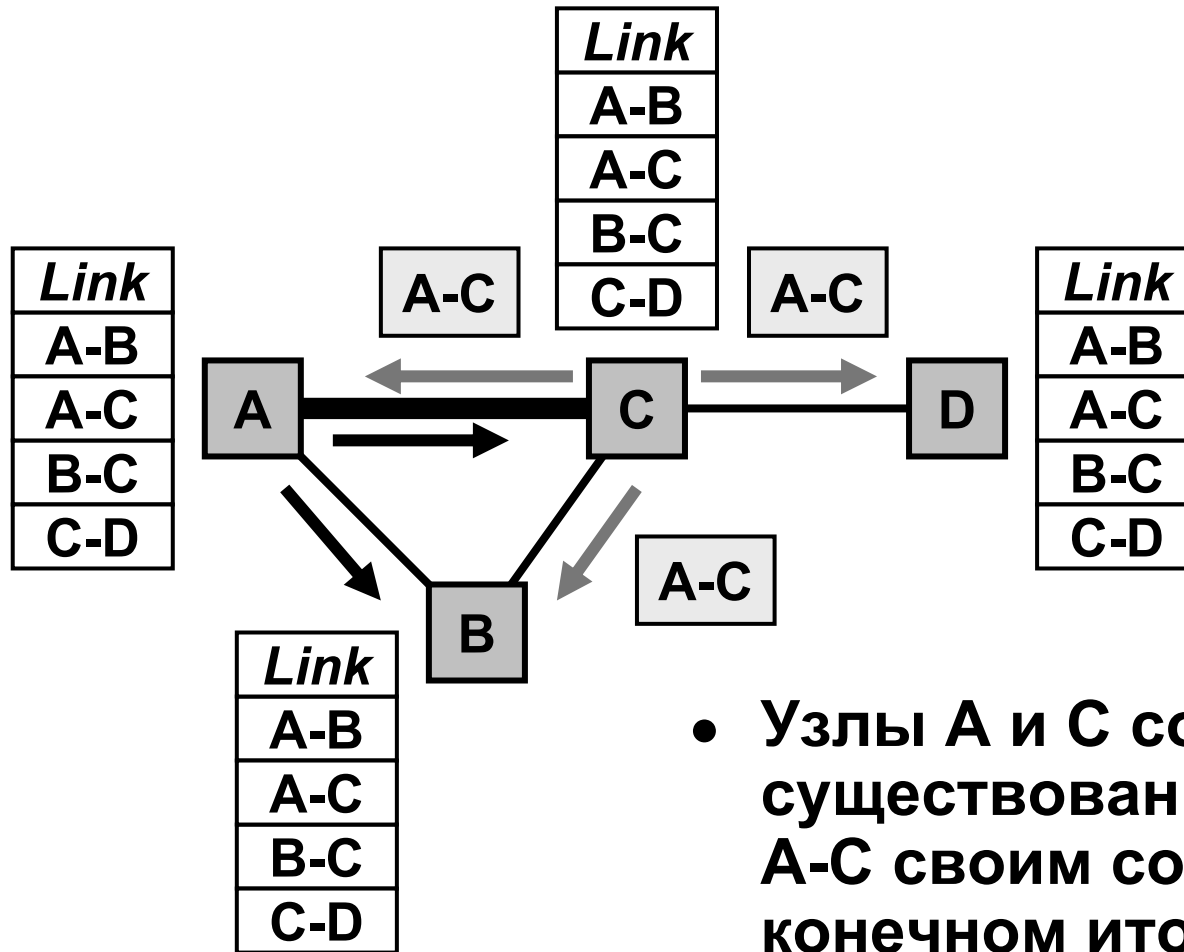
- Каждый узел делится информацией о своих соединениях, так что все узлы могут построить карту полной сетевой топологии
- Информация о соединении обновляется тогда, когда оно изменяет свое состояние
 - Состояние соединения проверяется посылкой коротких пакетов “hello” соседям
- Обладая полной информацией о топологии, узел может определить наилучший маршрут от источника

Алгоритмы состояния связи (2)



- Предположим, что топология остается стабильной достаточно долго, все узлы будут иметь одинаковую информацию о топологии

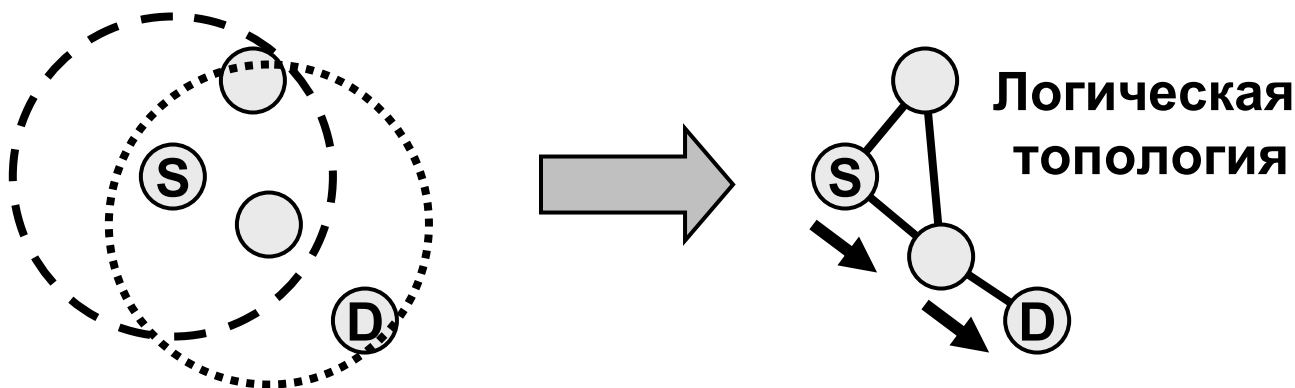
Алгоритмы состояния связи (3)



- Узлы A и C сообщают о существовании соединения A-C своим соседям, и, в конечном итоге, всей сети

MANET

- Специализированная мобильная сеть (MANET) характеризуется...
 - Многолучевым маршрутом, так что узлы, не имеющие соединения на Уровне 2, могут соединяться с помощью маршрутизации Уровня 3
 - Беспроводными соединениями
 - Мобильными узлами



MANET по сравнению с традиционной маршрутизацией (1)

- В MANET каждый узел является потенциальным маршрутизатором, в то время как большинство узлов традиционной проводной сети не управляют маршрутом пакетов
 - Узлы принимают и передают свои собственные пакеты, а также перенаправляют пакеты для других узлов
- Топология в сетях MANET является динамической из-за подвижности узлов, в то время как она остается статичной в традиционных сетях
- Маршрутизация в сетях MANET должна учитывать информацию обоих уровней 2 и 3, в то время как традиционные протоколы работают только с информацией Уровня 3
 - Информация связного уровня может показать наличие соединения и помех

MANET по сравнению с традиционной маршрутизацией (2)

- Топология MANET стремится к тому, чтобы иметь намного меньше избыточных соединений по сравнению с обычными сетями
- «Маршрутизатор» MANET обычно имеет одиночный интерфейс, в то время как традиционный маршрутизатор образует отдельный интерфейс для каждой сети, с которой он соединен
 - При передаче маршрутизируемый пакет посылается вперед, однако также посылается и предыдущему передатчику
- Свойства канала, такие как емкость и коэффициент ошибок, являются примерно постоянными в традиционных сетях, однако могут изменяться в сетях MANET

MANET по сравнению с традиционной маршрутизацией (3)

- **Взаимное влияние является отличительной особенностью MANET, но не традиционных сетей**
 - Например, пакеты, направляемые от В к С, конкурируют с новыми пакетами, отправляемыми от А к В
- **Каналы могут являться асимметричными посредством некоторых технологий Уровня 2**
 - Заметим, что IEEE 802.11 MAC предполагает наличие симметричных каналов
- **Эффективное управление энергопотреблением является отличительной особенностью MANET, в то время как на него обычно не обращают внимания в традиционных сетях**
- **MANET может иметь каналы выхода в фиксированную сеть, однако обычно является «шлейфной сетью», в то время как традиционные сети могут шлейфными либо транзитными**

MANET по сравнению с традиционной маршрутизацией (4)

- В MANET существуют физические ограничения по безопасности по сравнению с традиционной сетью
 - Увеличивается возможность прослушивания, имитации соединения и попыток взлома
- Традиционные протоколы маршрутизации для проводных сетей плохо работают в большинстве сетей MANET
 - Сети MANET слишком динамичны
 - Беспроводные соединения приводят к таким проблемам, как взаимное влияние, ограниченная емкость и т.д.

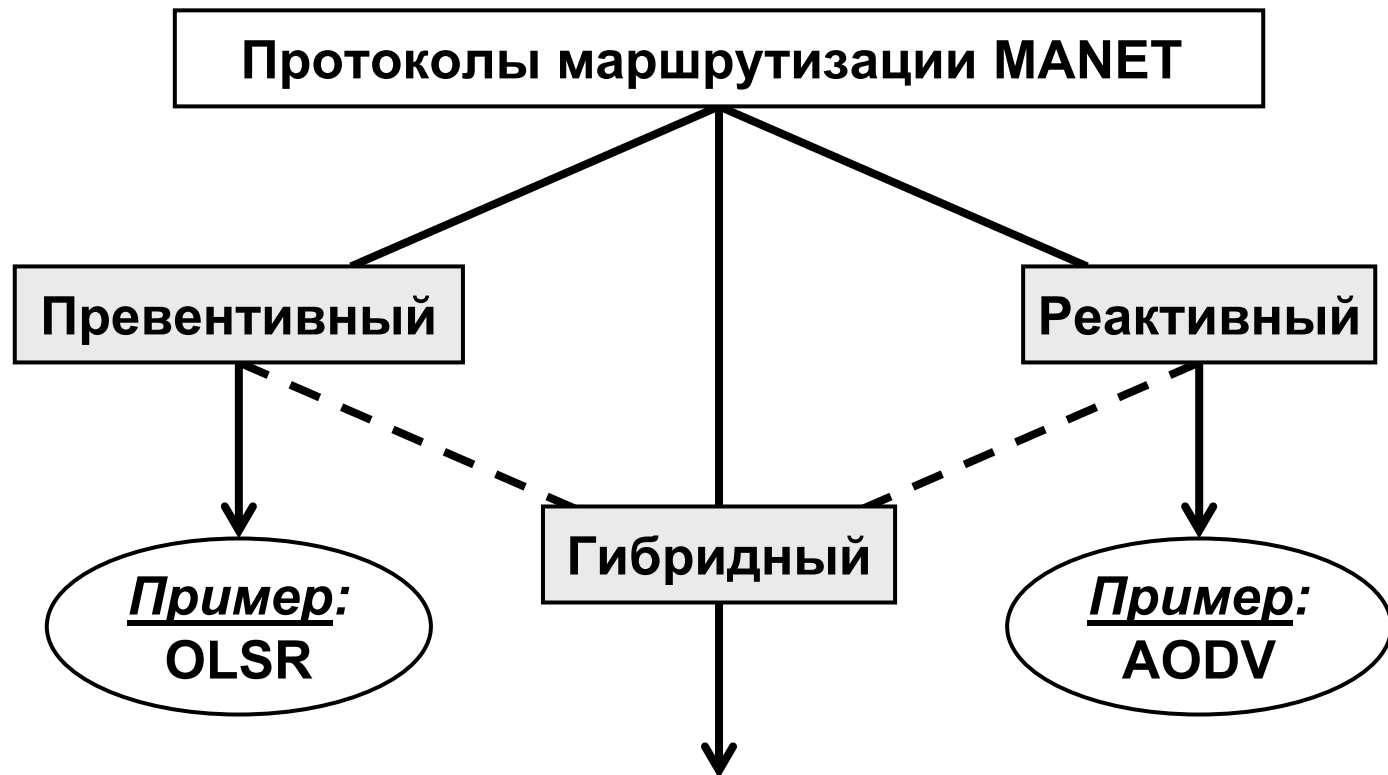
Маршрутизация MANET

- Узлы должны определять, каким образом направлять пакеты
 - Маршрутизация источника: решение о маршруте принимается отправителем
 - Маршрутизация от луча к лучу: решение о маршруте принимается в каждой промежуточной точке
- Трудности в достижении наилучшего качества
 - Маршрут изменяется с течением времени из-за подвижности узлов
 - Желательно избежать длинных задержек при посылке пакетов в первый раз
 - Желательно уменьшить затраты на поиск и обслуживание маршрута
 - Желательно исследовать максимально возможное количество узлов для поиска наилучшего пути и уменьшения вероятности разветвления

Принципы маршрутизации MANET

- **Принятие решения**
 - Превентивное или управляемое таблицей – обеспечивается таблицами маршрутизации
 - Реактивное или по запросу – маршрут рассчитывается по мере необходимости
- **Структура сети**
 - Иерархическая – из некоторого набора узлов формируется иерархия, которая отражается в алгоритме маршрутизации
 - Может использовать превентивный протокол внутри кластера или зоны
 - Может использовать реактивный протокол для прокладки маршрутов между различными «кластерными вершинами»
 - Неиерархическая – решение принимается с учетом всех узлов сети

Типы маршрутизации MANET



Общие особенности

- Протоколы маршрутизации MANET должны...
 - Находить путь от источника к приемнику
 - Поддерживать этот путь (например, если уехал промежуточный узел, и путь разрушился)
 - Определять механизмы обмена информацией о маршрутизации
- Реактивные протоколы
 - Находят маршрут в том случае, когда нужно передать пакет и для него нет известного пути
 - Пытаются изменить путь, если произошла ошибка маршрутизации
- Превентивные протоколы
 - Находят маршрут заранее для всех возможных пар источник-приемник
 - Периодически обновляют информацию о маршрутизации для поддержки путей

Рабочая группа IETF MANET (1)

<http://www.ietf.org/html.charters/manet-charter.html>

“Целью данной рабочей группы является стандартизация протоколов маршрутизации IP, функционально пригодных для беспроводных приложений как внутри статичной, так и динамической топологии сети. Основным исходным положением является то, что беспроводные интерфейсы имеют некоторые уникальные характеристики маршрутизации, а также то, что топология узлов внутри маршрутизируемого беспроводного региона может быть в высшей степени динамичной из-за движения и других факторов.”

Рабочая группа IETF MANET (2)

- В настоящее время проводится работа по продвижению четырех протоколов маршрутизации MANET до экспериментального статуса RFC
 - Специализированный протокол вектора расстояния по запросу (AODV)
 - Протокол динамической маршрутизации источника (DSR)
 - Протокол оптимизированной маршрутизации состояния соединения (OLSR)
 - Протокол топологического оповещения, базирующийся на обратном продвижении данных (TBRPF)
- URL
 - <http://www.ietf.org/html.charters/manet-charter.html>
 - http://protean.itd.nrl.navy.mil/manet/manet_home.html

OLSR

- **Протокол оптимизированной маршрутизации состояния соединения(OLSR)**
 - В процессе утверждения IETF Experimental RFC
- **Ссылки**
 - C. Adjih, *et al.*, “Optimized Link State Routing Protocol,” IETF Internet Draft, draft-ietf-manet-olsr-08.txt, March 3, 2003.
 - P. Jacquet, P. Muhlethaler, T. Clausen, A. Laouiti, A. Qayyum, and L. Viennot, “Optimized Link State Routing Protocol for Ad Hoc Networks,” *Proceedings IEEE INMIC*, 2001, pp. 62-68.

Концепция OLSR (1)

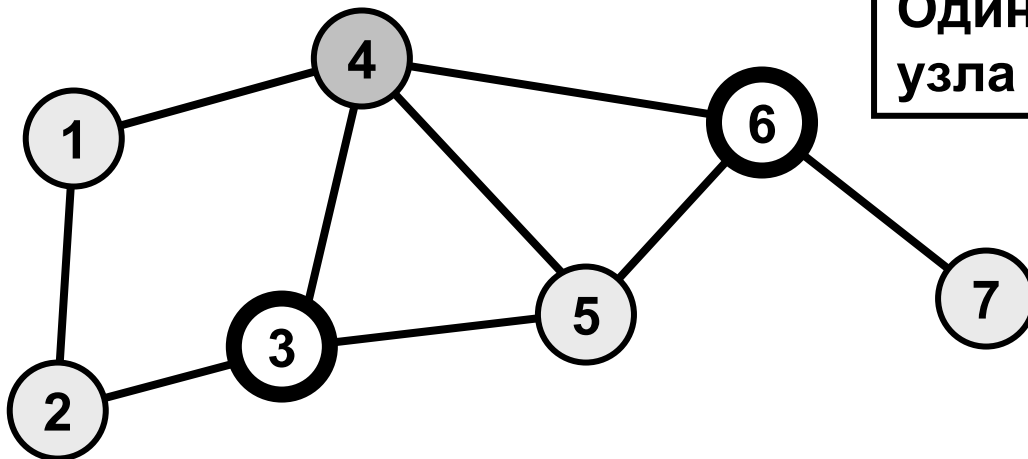
- Превентивный (управляемый таблицей) протокол маршрутизации
 - Маршрут доступен сразу, как только он нужен
- Базируется на алгоритме состояния соединения
 - Обычно в протоколах состояния соединения все узлы хранят информацию о соседях, но не в OLSR
- Узлы предоставляют информацию только о тех соединениях с соседями, которые находятся в *многоточечном наборе релейного селектора*
 - Уменьшается размер управляющих пакетов
- Уменьшается флуд за счет того, что в сеть посылается информация только об узлах *многоточечного реле*
 - Уменьшается количество управляющих пакетов за счет уменьшения количества дублирующих передач

Концепция OLSR (2)

- Не требует безотказной передачи, поскольку обновления посылаются периодически
- Не требует упорядоченной доставки, поскольку для предотвращения неправильной интерпретации истекших сообщений используется последовательная нумерация
- Использует маршрутизацию от луча к лучу
 - Прокладка маршрута основывается на динамических табличных записях, поддерживаемых в промежуточных узлах

Многоточечные реле

- Каждый узел N выбирает набор соседних узлов в качестве многоточечного реле, $MPR(N)$, которое передает управляющие пакеты от N
 - Соседи, отсутствующие в $MPR(N)$, обрабатывают управляющие пакеты от N , но не ретранслируют их
- $MPR(N)$ выбирается таким образом, чтобы все двухлучевые соседи N перекрывались однолучевыми соседями $MPR(N)$

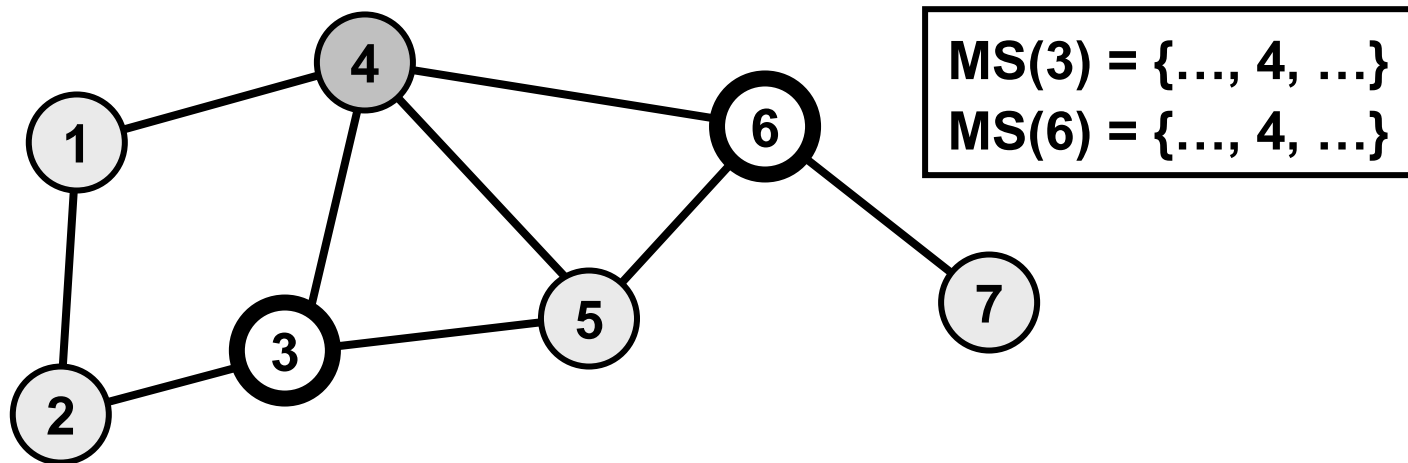


Один оптимальный набор для узла 4: $MPR(4) = \{ 3, 6 \}$

Существует ли другой оптимальный $MPR(4)$?

Многоточечный набор релейного селектора

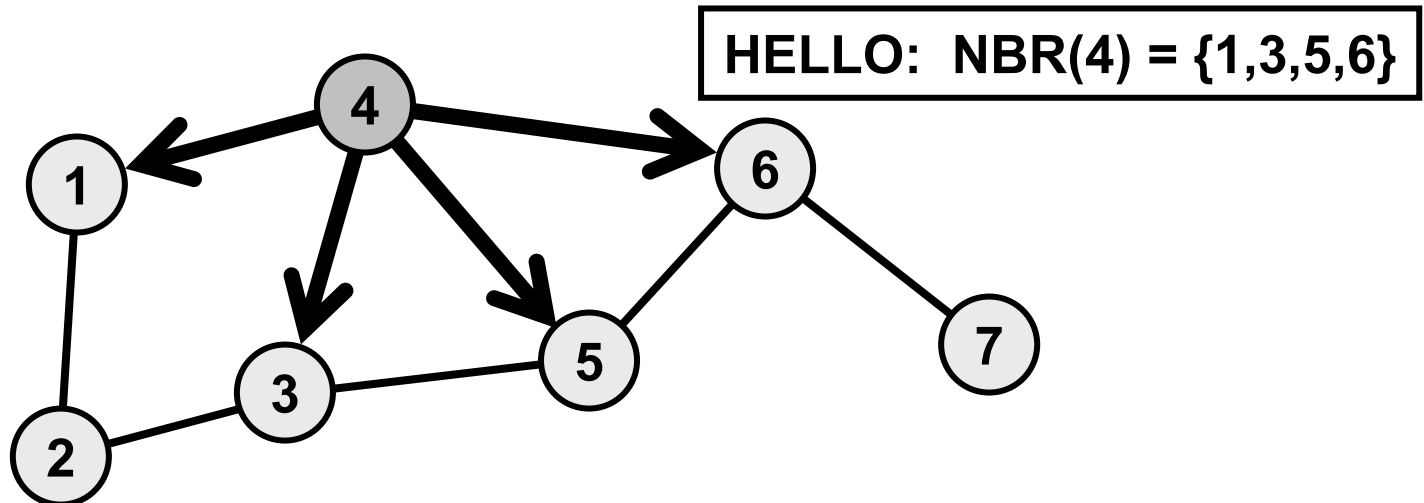
- Многоточечный набор релейного селектора для узла N , $MS(N)$, это набор узлов, которые включили узел N в свои многоточечные релейные наборы
 - В управляющих сообщениях будут представлены только соединения $N-M$, для всех M , так чтобы $N \in MS(M)$



(предполагается наличие двунаправленных соединений)

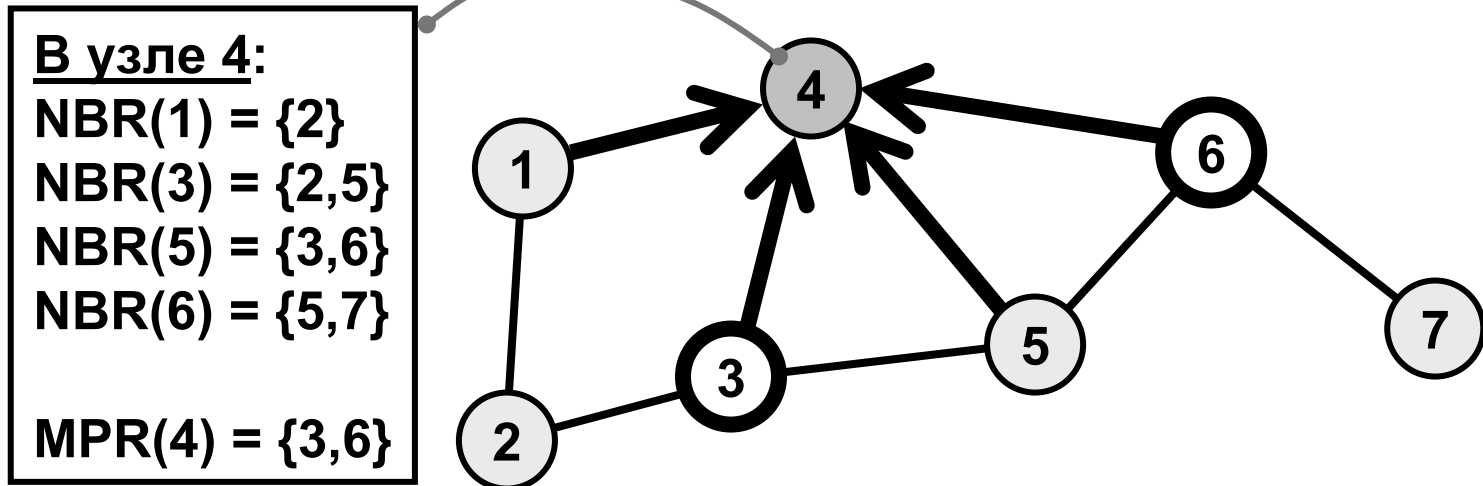
Сообщения HELLO (1)

- Каждый узел использует сообщения HELLO для формирования своего набора MPR
- Все узлы периодически посылают сообщения HELLO своим однолучевым соседям (соединения двунаправленные)
- Сообщения HELLO не пересылаются



Сообщения HELLO (2)

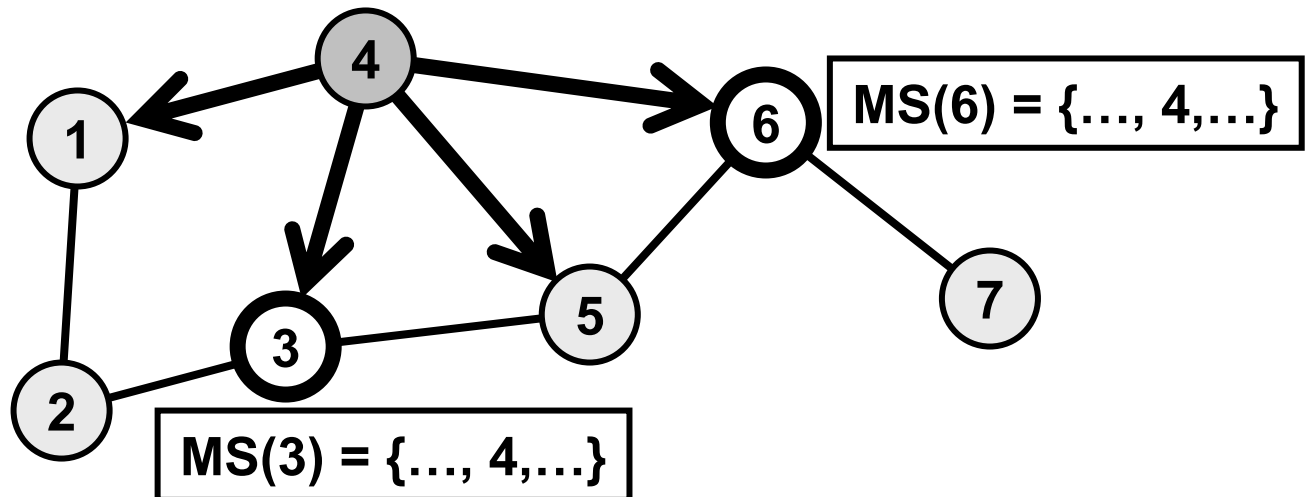
- Используя список соседей из принятых сообщений HELLO, узлы могут установить своих двухлучевых соседей и определить оптимальный (или близкий к оптимальному) набор MPR
- Данному набору MPR присваивается номер
 - Последовательный номер инкрементируется всякий раз, когда вычисляется новый набор



Сообщения HELLO (3)

- Последовательные сообщения HELLO указывают соседей, которые находятся в наборе MPR данного узла
- Набор MPR вычисляется повторно, если обнаруживается изменение одно- или двухлучевых соседей

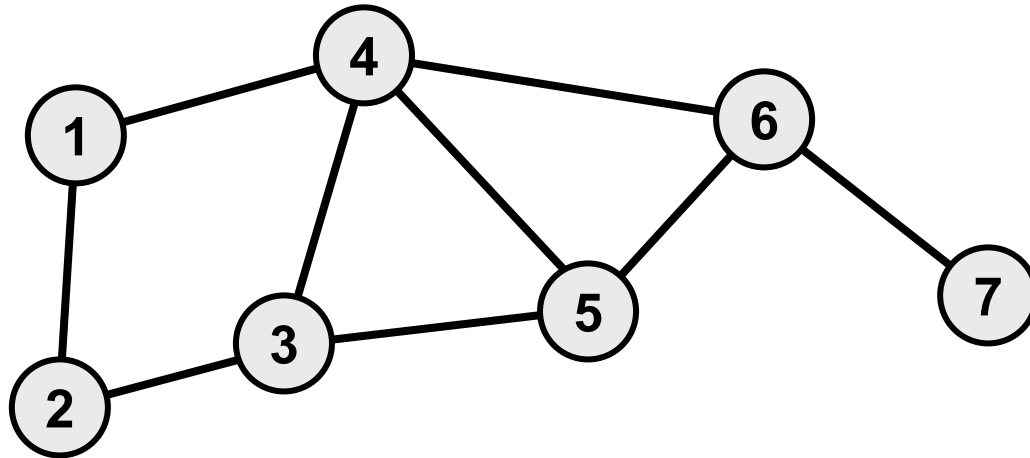
HELLO: $\text{NBR}(4) = \{1, 3, 5, 6\}$, $\text{MPR}(4) = \{3, 6\}$



Сообщения ТС

- Узлы посылают информацию о топологии в управляющих сообщениях ТС
 - Список оповещенных соседей (информация о соединении)
 - Последовательный номер (для предотвращения использования устаревшей информации)
- Узлы генерируют сообщения ТС только для тех соседей, которые находятся в их наборе MS
 - Только узлы MPR генерируют сообщения ТС
 - Не все узлы оповещаются
- Узлы обрабатывают все принимаемые сообщения ТС, но перенаправляют их только в том случае, если отправитель находится в их наборе MS
 - Только узлы MPR распространяют сообщения ТС

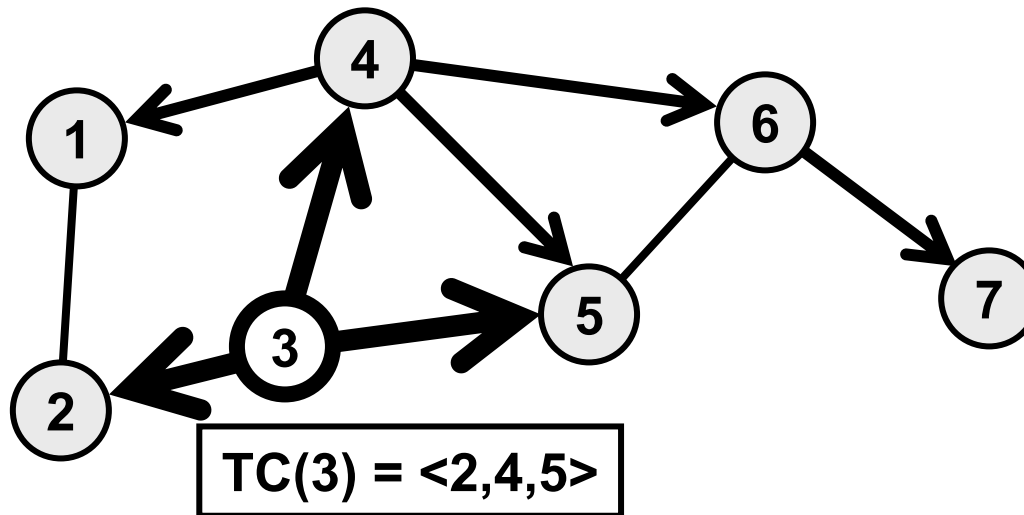
Пример OLSR (1)



MPR(1) = { 4 }
MPR(2) = { 3 }
MPR(3) = { 4 }
MPR(4) = { 3, 6 }
MPR(5) = { 3, 4, 6 }
MPR(6) = { 4 }
MPR(7) = { 6 }

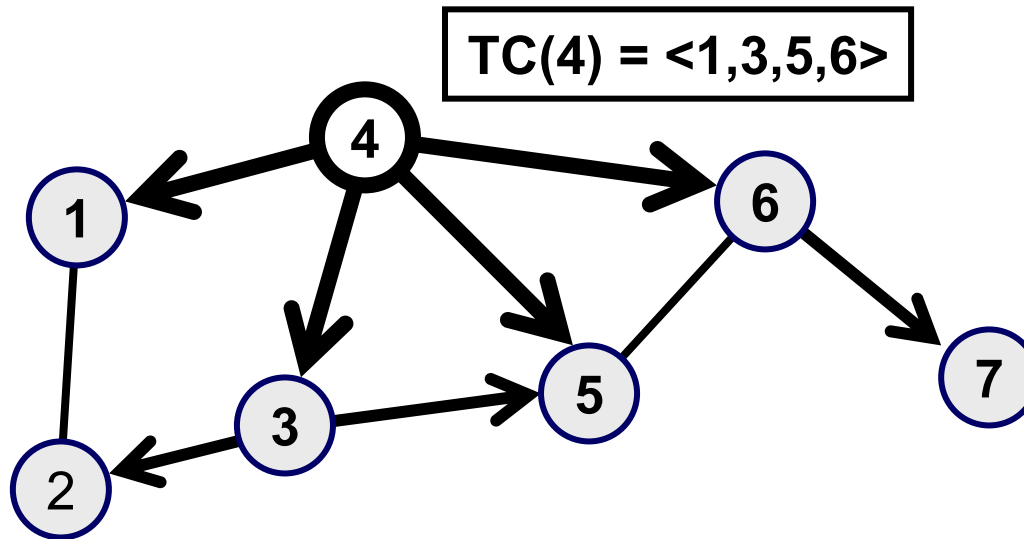
MS(1) = { }
MS(2) = { }
MS(3) = { 2, 4, 5 }
MS(4) = { 1, 3, 5, 6 }
MS(5) = { }
MS(6) = { 4, 5, 7 }
MS(7) = { }

Пример OLSR (2)



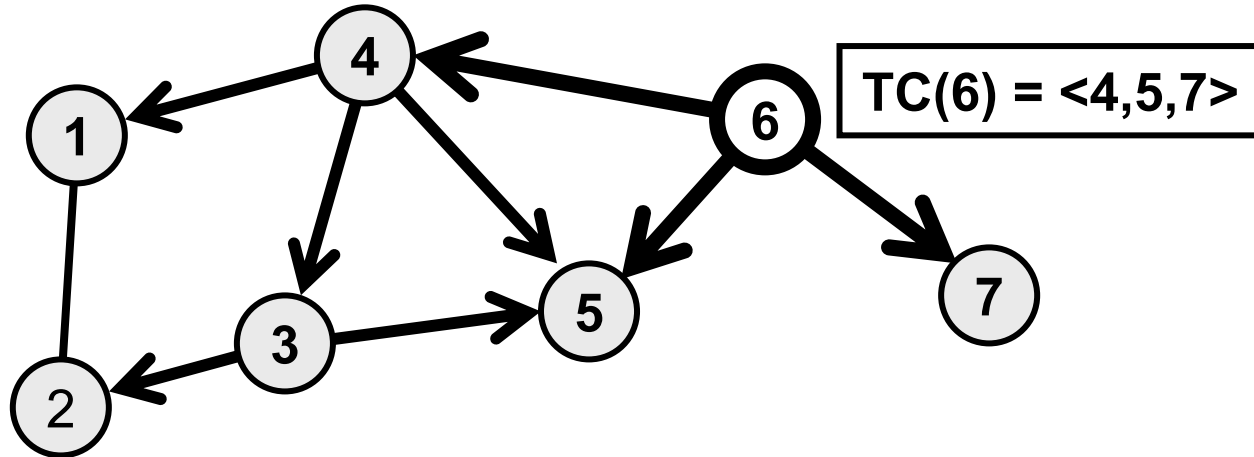
- Узел 3 генерирует сообщение TC, оповещая узлы из $MS(3) = \{2, 4, 5\}$
- Узел 4 пересылает сообщение TC узла 3, поскольку Узел 3 $\in MS(4) = \{1, 3, 5, 6\}$
- Узел 6 пересылает $TC(3)$, поскольку Узел 4 $\in MS(6)$

Пример OLSR (3)



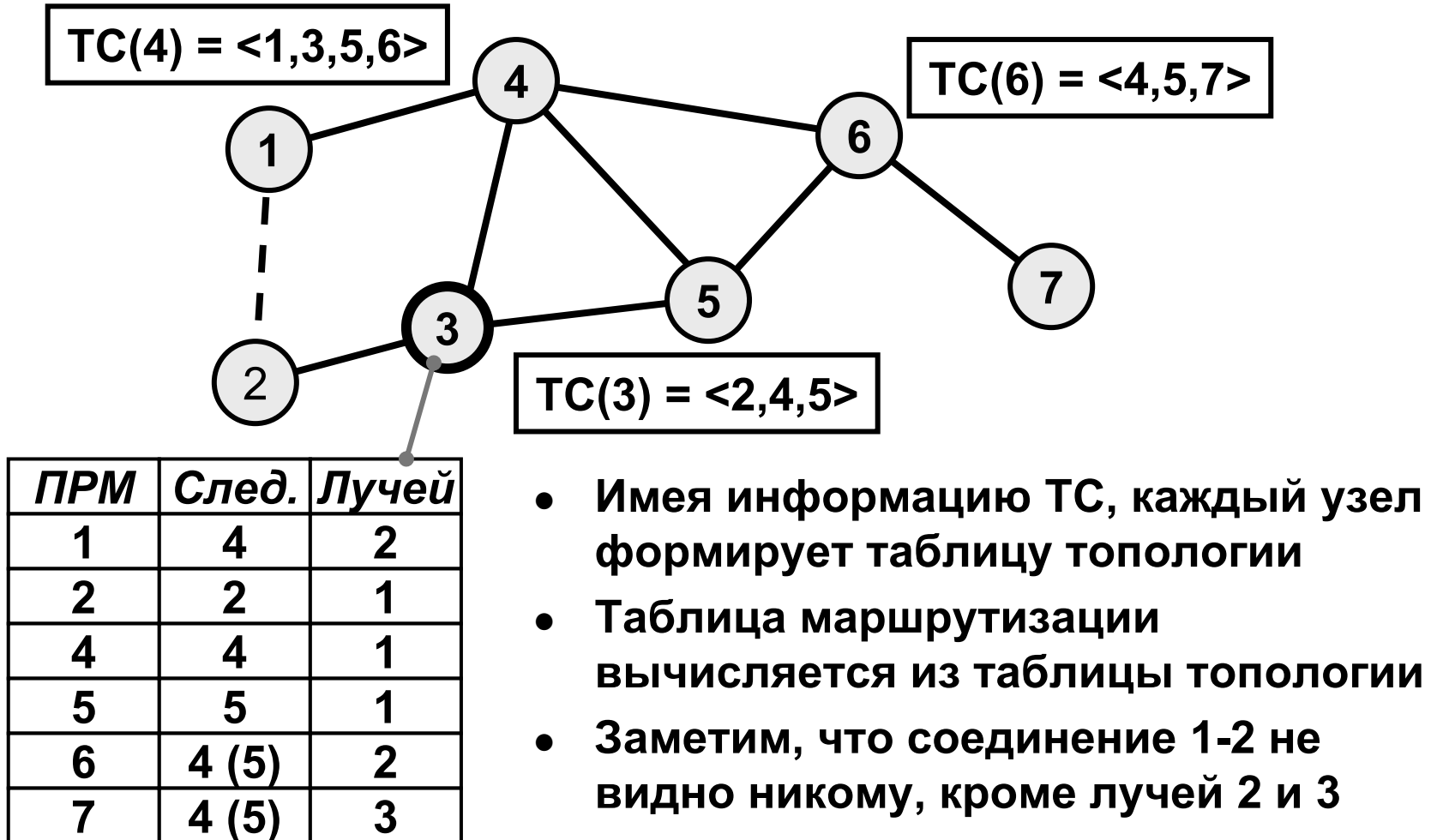
- Узел 4 генерирует сообщение TC, оповещая узлы из $MS(4) = \{1, 3, 5, 6\}$
- Узлы 3 и 6 пересылают TC(4), поскольку Узел 4 $\in MS(3)$ и Узел 4 $\in MS(6)$

Пример OLSR (4)



- Узел 6 генерирует сообщение TC, оповещая узлы из $MS(6) = \{4, 5, 7\}$
- Узел 4 пересылает TC(6) от Узла 6, а Узел 3 пересылает TC(6) от Узла 4
- После того, как Узлы 3, 4 и 6 сгенерировали сообщения TC, у всех узлов будет информация о состоянии соединения для любого маршрута

Пример OLSR (5)



AODV

- **AODV: Специализированный протокол вектора расстояния по запросу**
 - В процессе утверждения IETF Experimental RFC
- **Ссылки**
 - C. E. Perkins, E. M. Belding-Royer, and S. R. Das, “Ad hoc On-Demand Distance Vector (AODV) Routing,” IETF Internet Draft, draft-ietf-manet-aodv-13.txt, Feb. 17, 2003 (work in progress).
 - C. E. Perkins and E. M. Royer, “Ad hoc On-Demand Distance Vector Routing,” *Proceedings 2nd IEEE Workshop on Mobile Computing Systems and Applications*, February 1999, pp. 90-100.

Концепция AODV (1)

- **Протокол маршрутизации исключительно по запросу**
 - Узел не выполняет поиск маршрута или его поддержку до тех пор, пока ему не понадобится маршрут к другому узлу или пока он не предложит свои услуги в качестве промежуточного узла
 - Узлы, находящиеся в стороне активного маршрута, не поддерживают информацию о маршрутизации и не участвуют в формировании таблицы маршрутизации
- **Использует оповещательный механизм поиска маршрута**
- **Использует маршрутизацию от луча к лучу**
 - Маршрутизация основана на динамической таблице, поддерживаемой в промежуточных узлах
 - Аналогичен DSR, но DSR использует маршрутизацию источника

Концепция AODV (2)

- Для определения наличия местных соединений используются локальные сообщения HELLO
 - Можно уменьшить время отклика на запросы маршрута
 - Можно инициировать обновление по мере необходимости
- Маршрутам и записям в таблице маршрутизации присваиваются последовательные номера
 - Используются для замещения устаревшей информации
- Каждый узел поддерживает два счетчика
 - Последовательный счетчик узла
 - Широковещательный ID

Запрос маршрута AODV (1)

- Иницируется в том случае, когда узел хочет соединиться с другим узлом, но не знает маршрута
- Узел-источник посылает широковещательный пакет с запросом маршрута (RREQ) своим соседям

type	flags	resvd	hopcnt
broadcast_id			
dest_addr			
dest_sequence_#			
source_addr			
source_sequence_#			

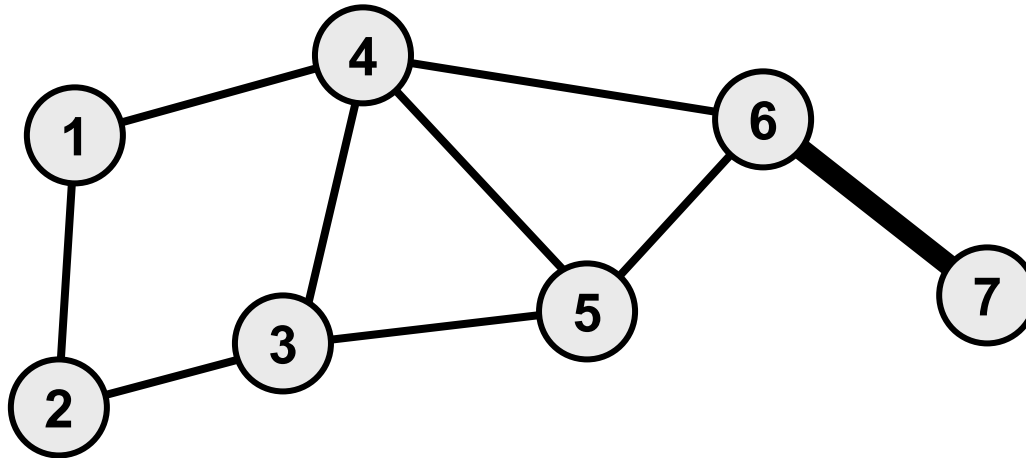
Запрос маршрута AODV (2)

- **Последовательные номера**
 - Последовательный номер источника показывает «степень свежести» обратного маршрута к источнику
 - Последовательный номер приемника показывает «степень свежести» обратного маршрута к приемнику
- **Каждый сосед, принимающий RREQ, либо...**
 - Возвращает пакет с ответом маршрута (RREP), либо
 - Пересылает RREQ своим соседям
- **Поля (source_addr, broadcast_id) уникально идентифицируют RREQ**
 - broadcast_id инкрементируется для каждого посылаемого пакета RREQ
 - Приемники могут распознать и удалить дублирующие пакеты RREQ

Запрос маршрута AODV (3)

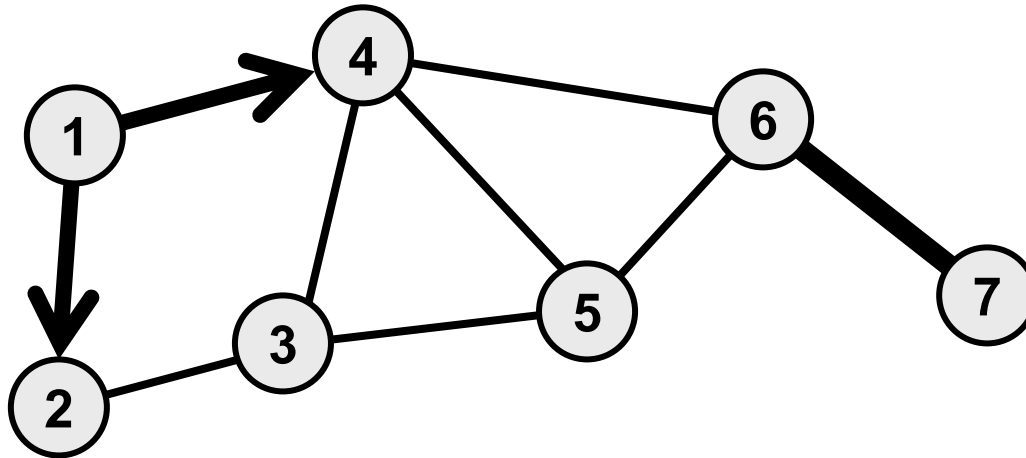
- Если узел *не может* ответить на RREQ, то
 - Узел инкрементирует счетчик лучей
 - Узел сохраняет информацию, необходимую для поддержки обратного маршрута (AODV предполагает наличие симметричных соединений):
 - Идентификатор соседа, который прислал данный пакет RREQ
 - IP адрес места назначения
 - IP адрес источника
 - Широковещательный ID
 - Последовательный номер узла источника
 - Время истечения записи для обратного маршрута

Пример AODV (1)



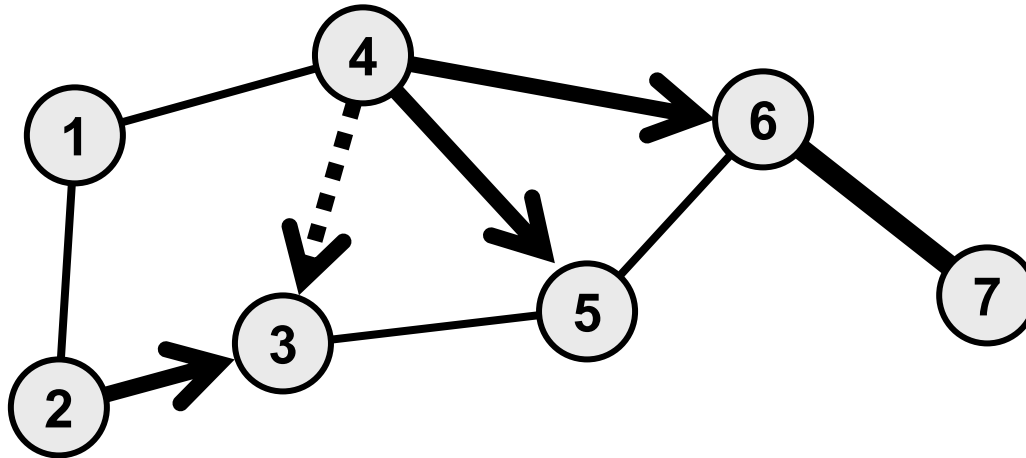
- Узел 1 хочет послать пакет Узлу 7
- Предположим, что Узел 6 знает текущий маршрут к Узлу 7
- Предположим, что в сети больше не существует какой-либо информации о маршрутизации (относящейся к Узлу 7)

Пример AODV (2)



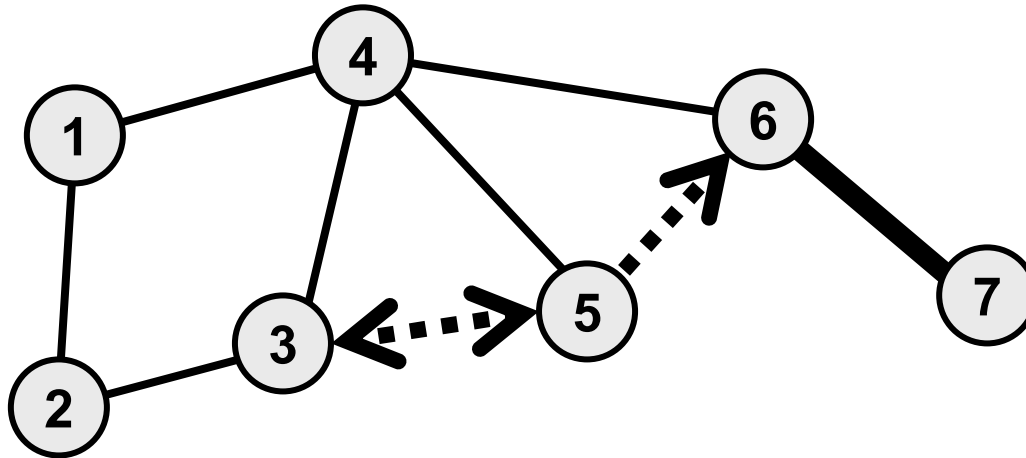
- **Узел 1 посылает пакет RREQ своим соседям**
 - **source_addr = 1**
 - **dest_addr = 7**
 - **broadcast_id = broadcast_id + 1**
 - **source_sequence_# = source_sequence_# + 1**
 - **dest_sequence_# = last dest_sequence_# for Node 7**

Пример AODV (3)



- Узлы 2 и 4 проверяют, что это новый RREQ и что `source_sequence_#` не устарел относительно обратного маршрута к Узлу 1
- Узлы 2 и 4 пересылают RREQ
 - Обновляют `source_sequence_#` для Узла 1
 - Инкрементируют `hop_cnt` в пакете RREQ

Пример AODV (4)



- **RREQ приходит на Узел 6, которому известен маршрут к Узлу 7**
 - Узел 6 должен проверить, чтобы последовательный номер приемника был меньше или равен последовательному номеру приемника, который он сохранил для Узла 7
- **Узлы 3 и 5 будут ретранслировать пакет RREQ, однако приемники посчитают его дубликатом**

Ответ о маршруте AODV (1)

- Если узел принимает пакет RREQ и у него есть текущий маршрут к месту назначения, то он отошлет однонаправленный пакет с ответом маршрута (RREP) тому соседу, от которого он принял пакет RREQ

type	flags	rsvd	prsz	hopcnt
dest_addr				
dest_sequence_#				
source_addr				
lifetime				

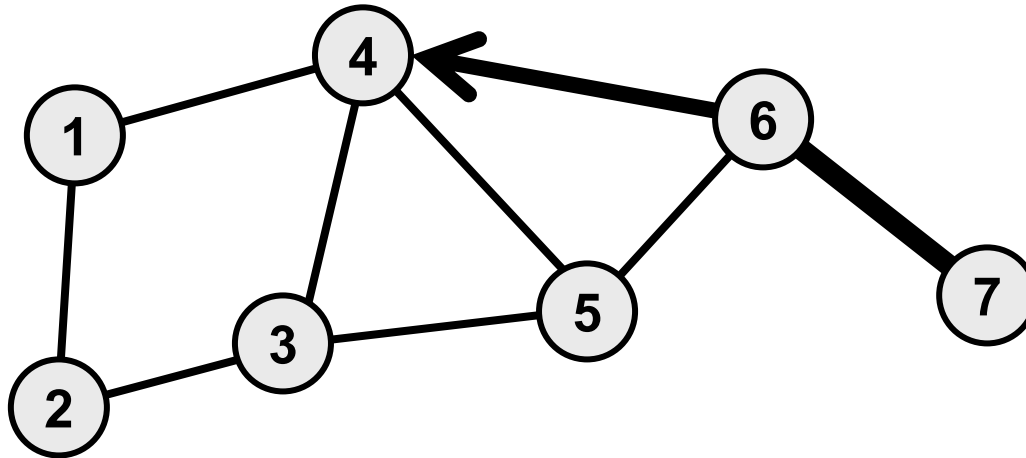
Ответ о маршруте AODV (2)

- Промежуточные узлы будут ретранслировать первый RREP по направлению к источнику, используя кэшированные записи обратного маршрута
- Остальные пакеты RREP отбрасываются до тех пор, пока...
 - Номер `dest_sequence_#` больше предыдущего, либо
 - `destination_sequence_#` такой же, но `hop_cnt` меньше (т.е., это лучший путь)
- RREP eventually makes it to the source, which can use the neighbor sending the RREP as its next hop for sending to the destination
- Cached reverse routes will timeout in nodes not seeing a RREP packet

Ответ о маршруте AODV (2)

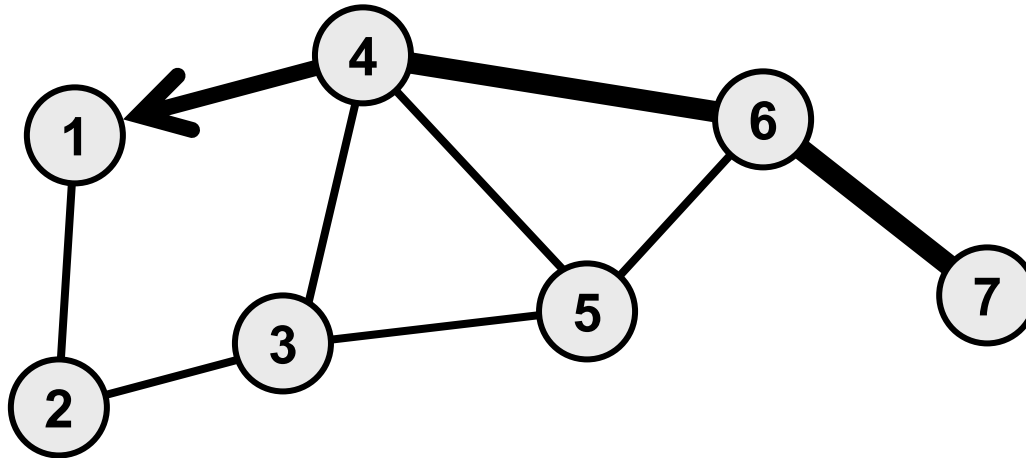
- Промежуточные узлы будут ретранслировать первый RREP по направлению к источнику, используя кэшированные записи обратного маршрута
- Остальные пакеты RREP отбрасываются до тех пор, пока...
 - Номер `dest_sequence_#` больше предыдущего, либо
 - `destination_sequence_#` такой же, но `hop_cnt` меньше (т.е., это лучший путь)
- В конце концов RREP достигает источника, который может использовать соседа, приславшего RREP, в качестве следующего луча для пересылки информации к месту назначения
- Кэшированные обратные маршруты будут истекать по времени в тех узлах, которые не видят пакета RREP

Пример AODV (5)



- Узел 6 знает маршрут к Узлу 7 и посылает RREP к Узлу 4
 - `source_addr = 1`
 - `dest_addr = 7`
 - `dest_sequence_# = maximum` (собственный последовательный номер, `dest_sequence_#` из RREQ)
 - `hop_cnt = 1`

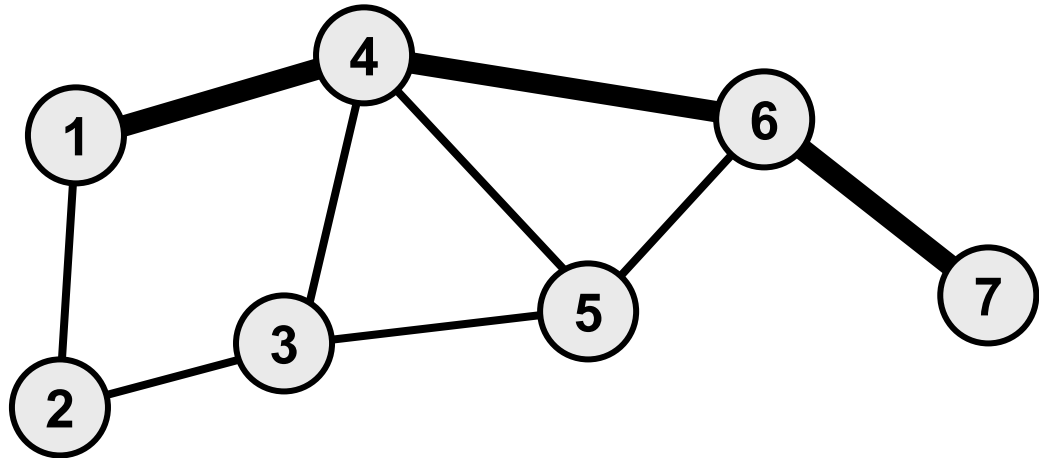
Пример AODV (6)



- Узел 4 проверяет, что это **новый** ответ о маршруте (случай, показанный здесь) либо ответ, содержащий **наименьшее** число лучей, и, если это так, пересылает пакет **RREP** к Узлу 1
 - Инкрементируется `hop_cnt` в пакете RREP

Пример AODV (7)

ПРМ	След.	Лучей
7	4	3

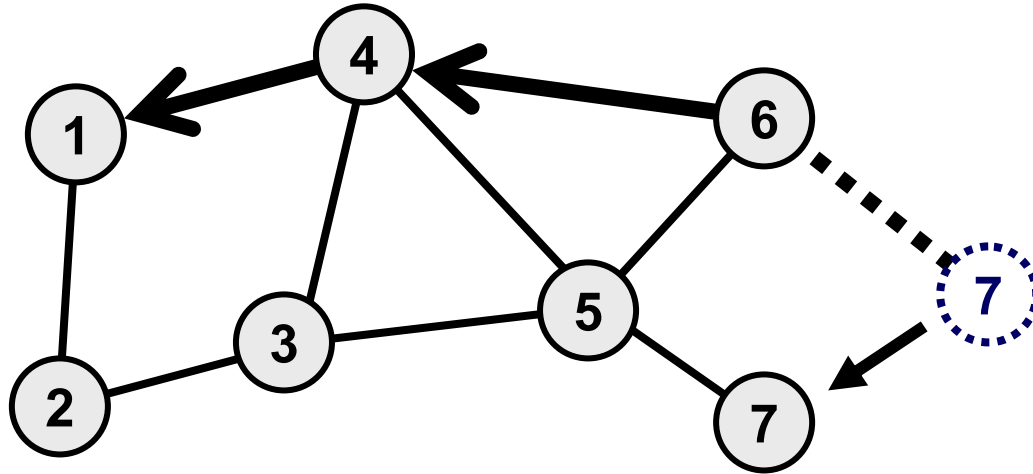


- Узел 1 теперь знает маршрут из трех лучей к Узлу 7 и может сразу использовать его для отправки пакета данных
- Заметим, что первый пакет данных, будет задержан до тех пор, пока не возвратится первый RREP

Обслуживание маршрута в AODV

- **Изменение маршрута может быть обнаружено...**
 - Из-за пропажи периодических сообщений HELLO
 - Из-за аварии на связном уровне
 - При ошибке передачи пакета к следующему лучу (может быть обнаружена с помощью прослушивания ретрансляции, если это не конечная точка назначения)
- **Вышестоящий (по направлению к источнику) узел, обнаруживший ошибку, передает пакет ошибочного маршрута (RERR) с новым последовательным номером места назначения и количеством лучей, равным бесконечности (недостижимый маршрут)**
- **Источник (или другой узел маршрута) может снова построить путь, послав пакет RREQ**

Пример AODV (8)



- Предположим, что Узел 7 уехал, и связь 6-7 разорвалась
- Узел 6 посылает пакет RERR с указанием разрушенного маршрута
- RERR распространяется обратно к Узлу 1
- Узел 1 может начать поиск нового маршрута

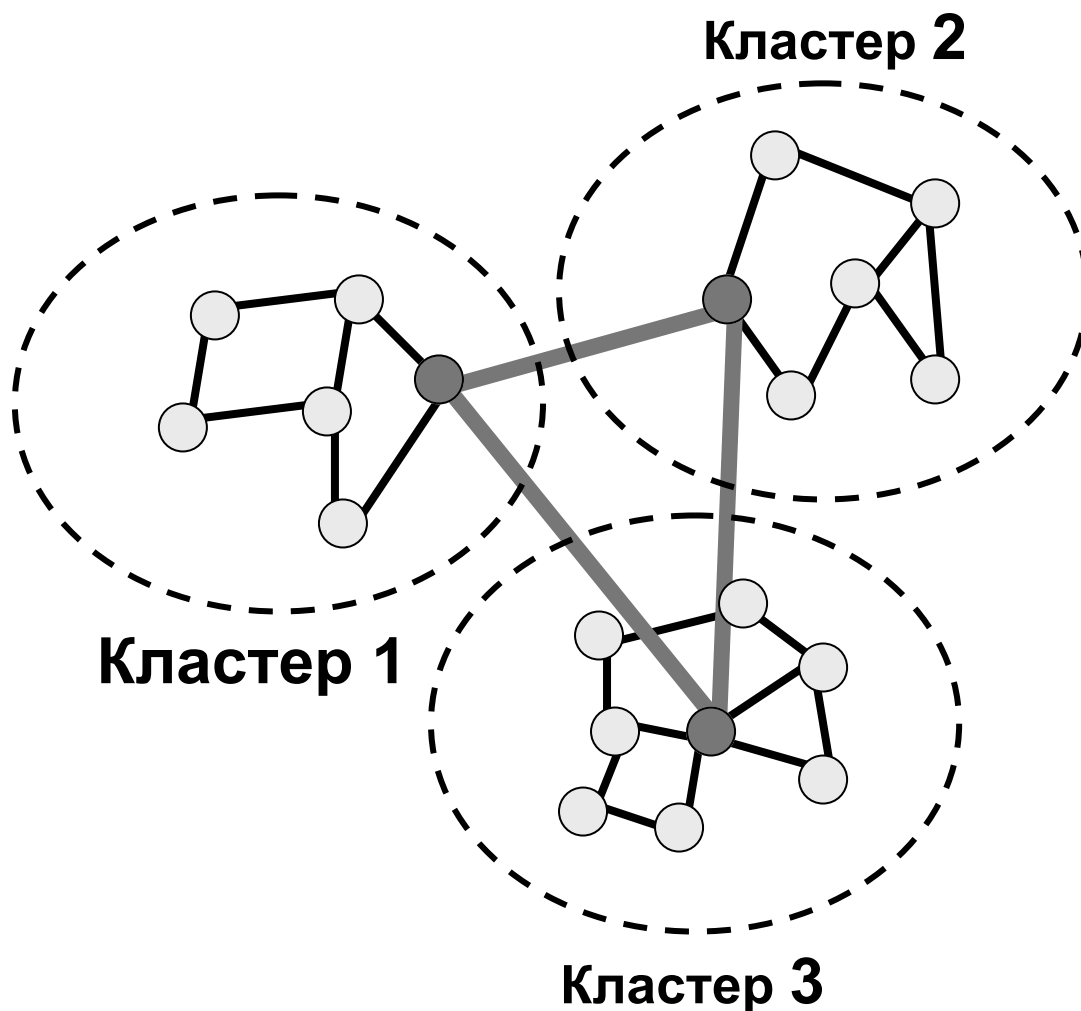
Иерархические алгоритмы (1)

- **Расширяемость – протоколы MANET зачастую не слишком хороши для больших сетей (особенно не очень плотных)**
 - **Глобальная топология базируется на соединении каждого мобильного узла**
- **Для обеспечения расширяемости можно использовать кластеры**
 - **Кластеры формируются (динамическим образом) для создания и поддержки иерархии**
 - **Глобальная маршрутизация осуществляется между кластерами**
 - **Локальная маршрутизация осуществляется между узлами внутри кластера**
 - **Для расширения иерархии могут быть сформированы кластеры из кластеров (суперкластеры)**
 - **Аналогично принципам подсетей IP**

Иерархические алгоритмы (2)

- В каждом кластере один из узлов обозначается как *вершина кластера*
 - Он отвечает за маршрутизацию данных между кластерами
 - Может являться специализированным узлом, либо быть назначенным алгоритмом кластеризации
- Алгоритмы
 - Кластеризация – формирование кластеров
 - Идентификация вершины кластера – может являться составной частью алгоритма кластеризации
 - Маршрутизация – по-прежнему требуется какой-либо алгоритм маршрутизации
 - Назначается каждому уровню иерархии

Пример иерархического алгоритма



Заключение

- Маршрутизация на Уровне 3 необходима для расширения беспроводных мобильных сетей за локальные пределы непосредственно соединенных друг с другом узлов
- Специализированные мобильные сети используют многолучевую маршрутизацию, чтобы поддерживать соединения в сети с динамической топологией
- Маршрутизацию MANET трудно сделать хорошо – здесь нужно одновременно решать проблемы нестабильности беспроводных соединений и их мобильности
- Представлены реактивные и превентивные протоколы маршрутизации MANET
- MANET пока является узкоспециальным нишевым приложением и отработан достаточно слабо