Wireless, mobile networking

Wireless TCP

- Потери пакетов в беспроводных сетях обычно возникают из-за
  - Битовых ошибок
  - Переадресации (абонент переходит из одной ячейки сети в другую)
  - Перегрузки (редко)
  - Изменение порядка передачи данных (редко, однако в мобильных ad hoc сетях может возникать достаточно часто)
- КлассическийTCP предполагает потери пакетов из-за
  - Перегрузки
  - Изменение порядка передачи пакетов (редко)
- Потери пакетов в беспроводной сети интерпретируются TCP как потери из-за перегрузок, что не плохо соответствует действительности

Возможные подходы для решения задачи

- Улучшения канального уровня (коррекция ошибок, повторная передача)
  - Влияние на время передачи и подтверждения передачи, большие вариации времени могут приводит к таймаутам
  - Не проблема при предсказуемых таймаутах
  - Проблема в беспроводных сетях с низкими скоростями так как установленное время таймаута может быть большим
- Иной транспортный уровень, например I-TCP [BakreBadri95]
- TCP имеет информацию о канального уровня (Snoop)[Hari et al 96]
- Явно заданные алгоритмы предупреждений о потерях

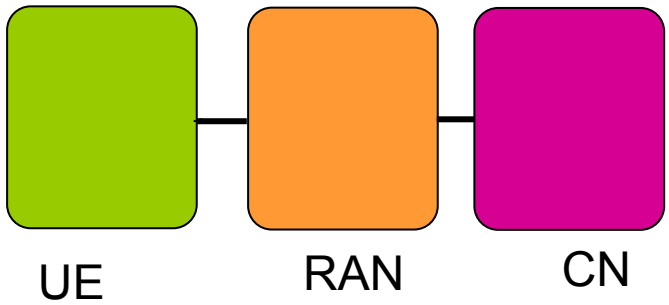
Особенности повторных передач пакетов на канальном уровне

- Сколько повторных передач предпринимать на канальном уровне перед тем, как считать передачу невозможной?
  - Ограниченное количество – ненадежный канальный уровень
  - Неограниченное – надежный канальный уровень
- Что запускает механизм повторных передач на канальном уровне?
  - Процедуры тайм-аутов на канальном уровне
  - Служебные пакеты канального уровня (negative acks, dupacks, sacks)
- Сколько времени требуется на повторную передачу на канальном уровне?
  - Small fraction of end-to-end TCP RTT
  - Large fraction/multiple of end-to-end TCP RTT
- Как должен передавать пакеты канальный уровень – в порядке приема или в порядке передачи (in-order)?
  - Канальному уровню потребуется буферизовать пакеты и изменять их порядок следования для приема по порядку (in-order)

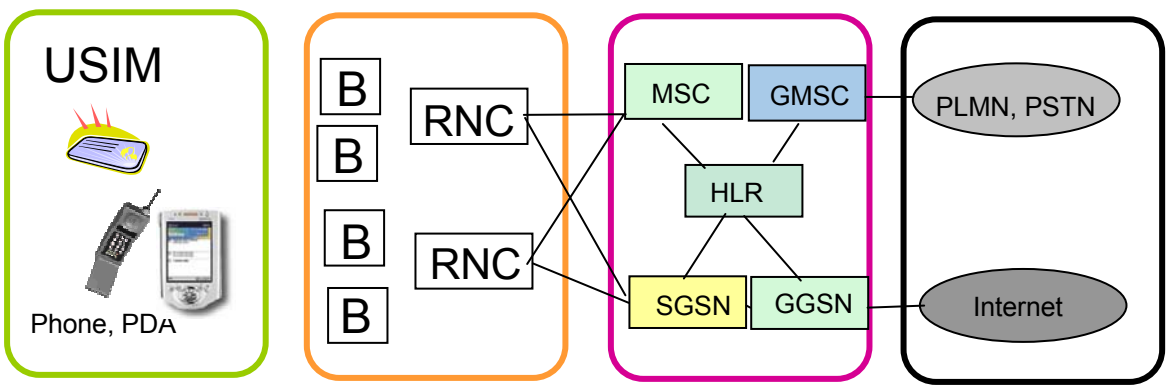
Применимость схем работы канального уровня

- Когда гарантированная доставка пакетов улучшает производительность TCP?
- Если он обеспечивает доставку пакетов в правильном порядке
- Таймауты для повторной передачи протокола TCP выбираются с учетом дополнительных задержек, связанных с повторными передачами на канальном уровне
- Другая проблема состоит в том, что размер пакетов канального уровня может быть меньше MSS TCP

3G сети

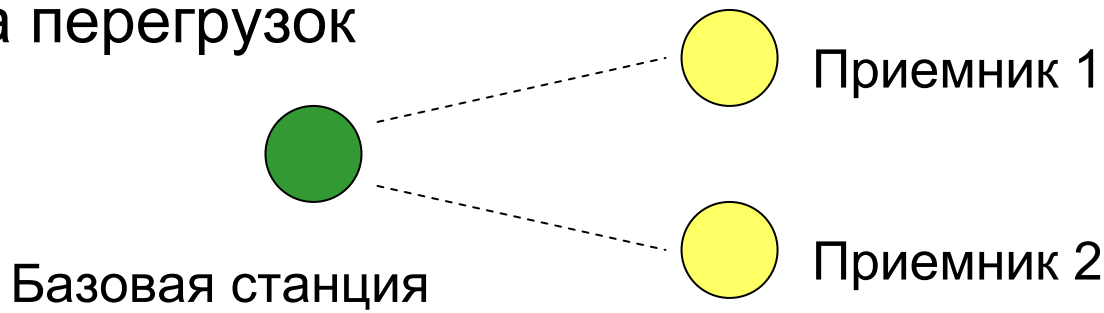


- Оборудование пользователя(UE), Сеть радио доступа (UTRAN), Основная сеть (CN)
  - UE состоит из оборудования пользователя USIM + ME
  - RAN состоит из базовых станций (B) и контроллеров радиосети
  - Основная сеть (CN) состоит из элементов передачи голоса (GSM) и элементов передачи данных (GPRS)



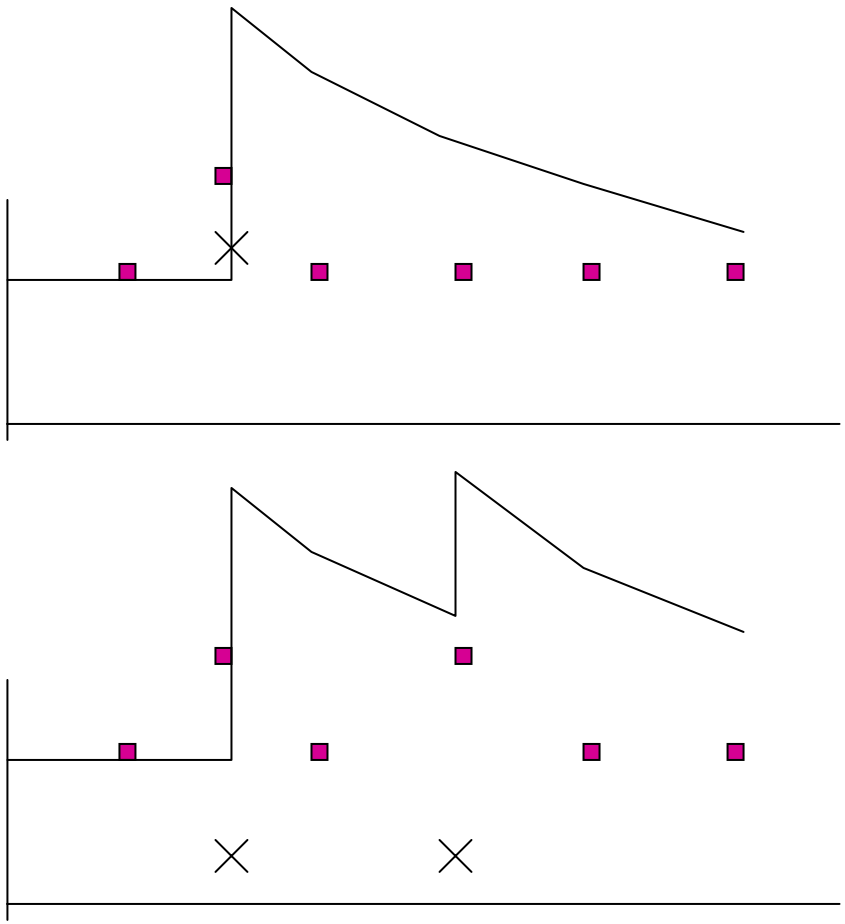
Особенности повторных передач на канальном уровне

- Повторные передачи могут приводить к потерям из-за перегрузок



- Попытки повторно передать пакет в начале очереди уменьшает доступную полосу канала и может привести увеличению размеров очереди на базовой станции
- Если очередь заполняется, пакеты могут теряться, сигнализируя отправителю перегрузку
- Планирование загрузки канала приводит к необходимости использования изменяемой битовой скорости

Изменения RTO (Retransmission TimeOut)



Потери пакетов в беспроводной сети

RTT (Round Trip Time) оценка

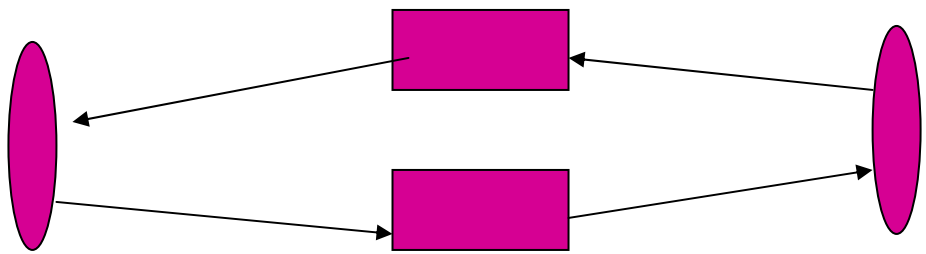
RTO

Влияние битовой скорости и вариаций задержек

- Сжатие Ack пакетов
- Прием ack пакетов в пачке приводит к быстрому освобождению очереди пакетов на прием
- Прием пачек при плохой связи приводит к множественным потерям пакетов
- Множественные потери пакетов приводят к сильному уменьшению пропускной способности TCP

Механизм управления ACK

- Необходимо запоминать количество полученных ack (ожидаемых пакетов)
- Выделять буфер размера, соответствующий ожидаемым пакетам Data queue



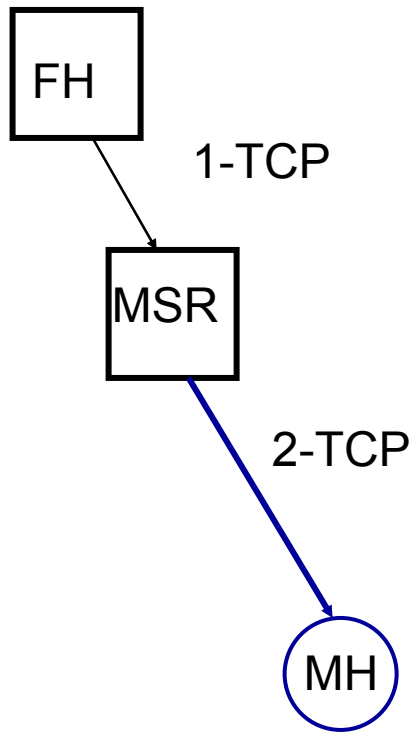
Беспроводная сеть

Очередь Ack

Проводная сеть

I-TCP

- Использует разделенные соединения
  - Сквозное соединение разбивается на соединение с проводной сетью и беспроводной
  - Беспроводная часть TCP может быть оптимизирована для беспроводной сети
  - TCPоптимизируется под потребности конкретной сети



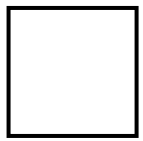
Подход с разбиением сети

- Разбиение соединений приводит к существованию двух различных потоков. Отсюда возможно принятие различных решений при потерях пакетов
- В беспроводной сети ➔ попытаемся еще
- В проводной сети ➔ не пытаемся
- Возможно настроить TCP для реализации этих механизмов

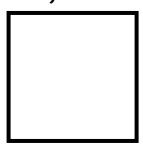
Установление TCP соединений

- FH должен ожидать соединение от MH, а не от MSR
- MH должен открывать соединение с FH и не должен ничего знать о MSR
- MH имеет библиотеку I-TCP которая перехватывает запрос и устанавливает соединение с MSR
- MSR соединяется с FH но <адресом MH и port #> указанными FH

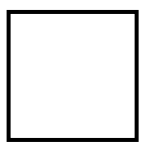
<mh, port_mh, FH, port_FH>



<mh, port_mh, msr, port_msr>



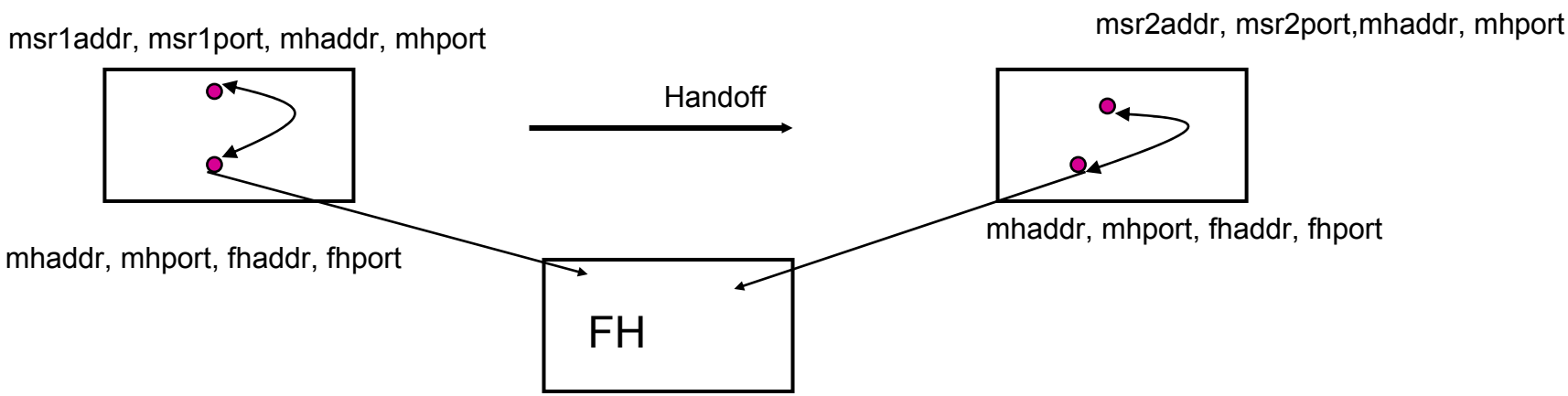
msr, port_msr, mh, port_mh



FH, port_FH,mh,port-mh

I-TCP хэндофф

- Когда MH перемещается в новое место, MH устанавливает соединение с новым MSR
- Новый MSR получает информацию о TCP соединениях от предыдущего MSR



Особенности I-TCP

- Скрывает потери пакетов, отправленных в беспроводной сети
- TCP в беспроводной сети может быть оптимизирован независимо
- Высокая производительность в случае использования в WAN сетях
- Повторные передачи возникают только при плохой связи
- Быстрое восстановление из-за небольшого RTT в беспроводной сети
- Хэндофф требует передачу состояния от MSR к MSR
- Требуется дополнительное место в буферах, накладные расходы в MSR
- Нарушение прозрачности соединений должно быть скрыто на уровне приложений
- Отказ MSR может привести к потере состояния TCP

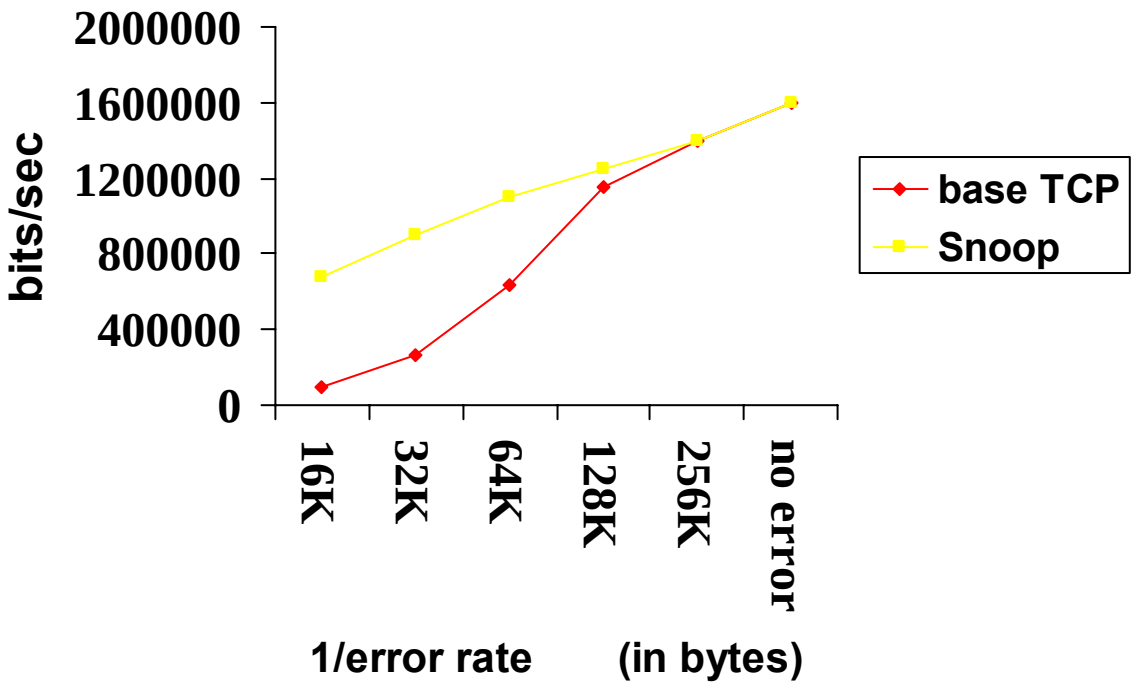
Особенности Snoop

- В отличие от I-TCP, соединение остается прозрачным
- Высокая пропускная способность при средних уровнях ошибок
- Бесполезен, если заголовки TCP шифруются
- Не может быть использован на несимметричных каналах

Snoop : Основная идея

- Данные от FH к MH
  - Кэшируем неподтвержденные TCP пакеты
  - Делаем локальные повторные передачи
- Данные от MH к FH
  - Обнаруживаем потерянные пакеты
  - Посылаем негативные ACK

Эффективность Snoop



Беспроводной канал 2 Mbps

Достоинства Snoop

- Snoop предупреждает быструю повторную передачу несмотря на ошибка при передаче и доставку не по порядку в беспроводной сети
- Работает при небольших размерах окон

Недостатки Snoop

- Канальный уровень на базовой станции должен иметь информацию о транспортном (TCP)
- Бесполезен в случае использования шифрованных заголовков (IPsec)
- Не может быть использован если TCP пакеты и ACK пакеты идут разными маршрутами (не через базовую станцию)

FH -> MH : Snoop_data() – case 1 and 2

- New Packet in normal TCP sequence

  Normal case

  Add to snoop cache

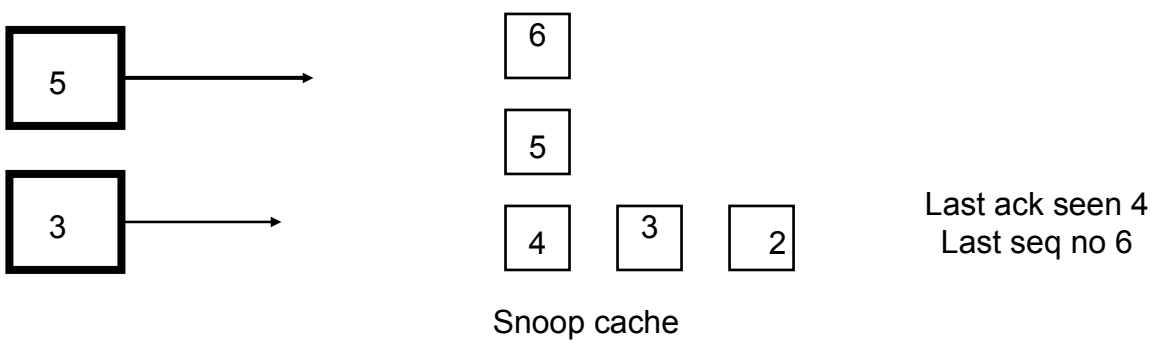
  Forward to MH

- Out of sequence packet cached earlier

  Fast Retransmission/timeout at sender due to

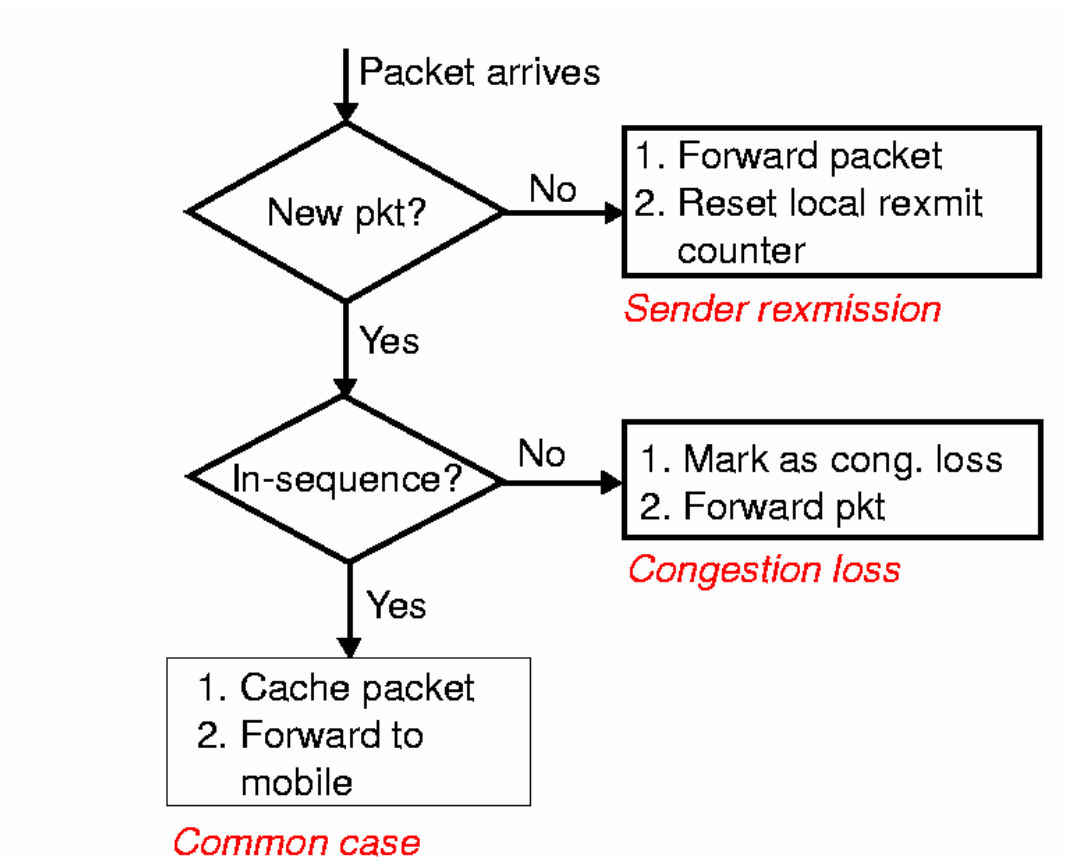
  A)Loss in wireless link (if last ACK is < current seq.no.): ***Forward to MH***

  B) Loss of previous ACK (if last ACK > current seq.no.): ***Send ACK to FH (similar to last one seen) with MH address and port***

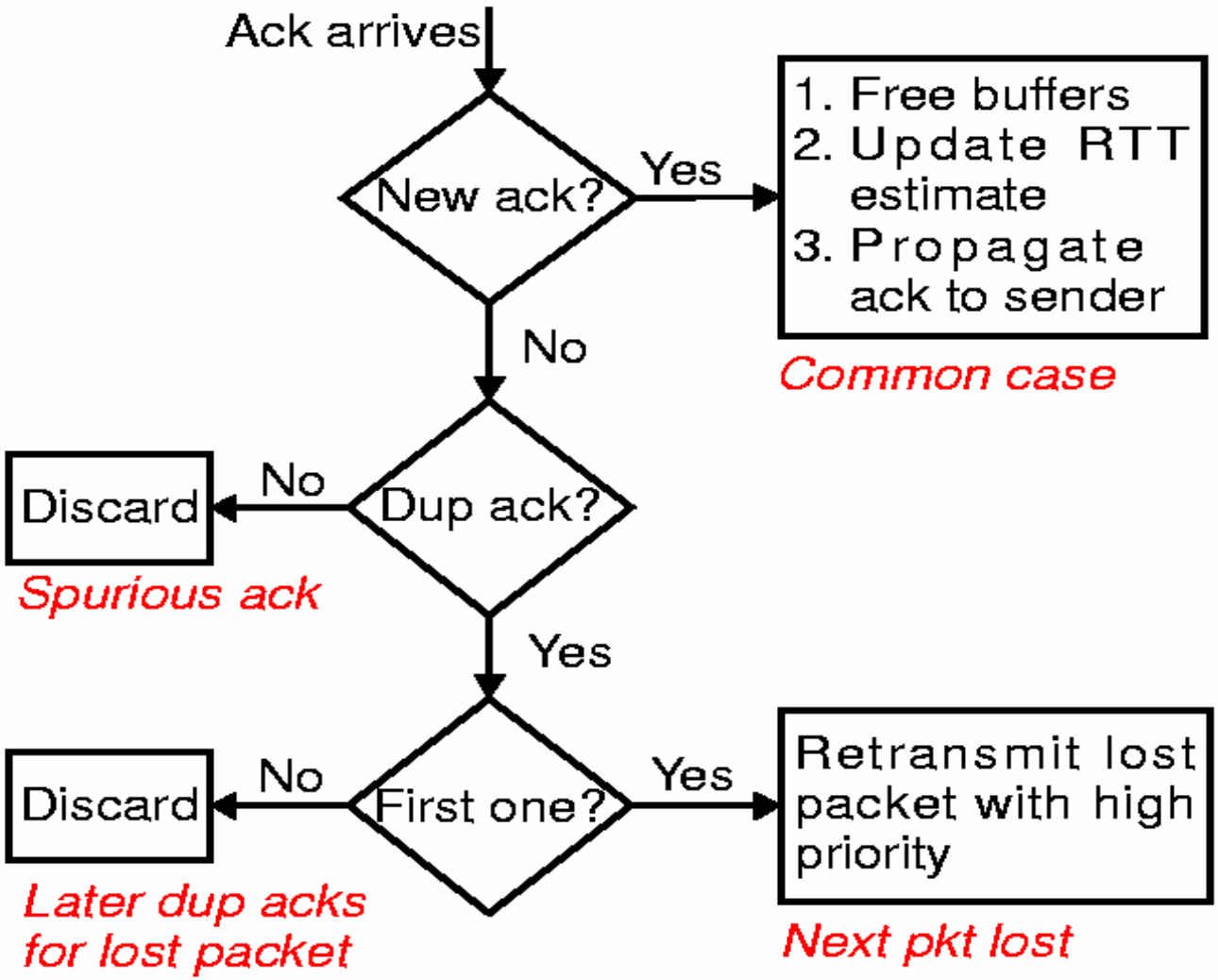


Snoop: FH -> MH

Data Processing



Snoop: ACK Processing



Проблемы

- Беспроводная сеть
  - Проблемы из-за потерь
  - Проблемы из-за изменений в задержках, скорости
- Мобильность конечной точки (абонента)
  - Изменения адресации при перемещении абонента
- Изменения топологии
  - Конфигурация изменяется при перемещении узлов сети

Мобильные пользователи

- Бурный рост использования карманных ПК
- Беспроводной доступ в любом месте и любое время
  - Хотя бы плохое подключение доступно в любом месте
  - Множественное во времени и месте подключение
- Пользователи могут быть подключены во время движения
- Пользователи могут подключены и отключены в различных сетях

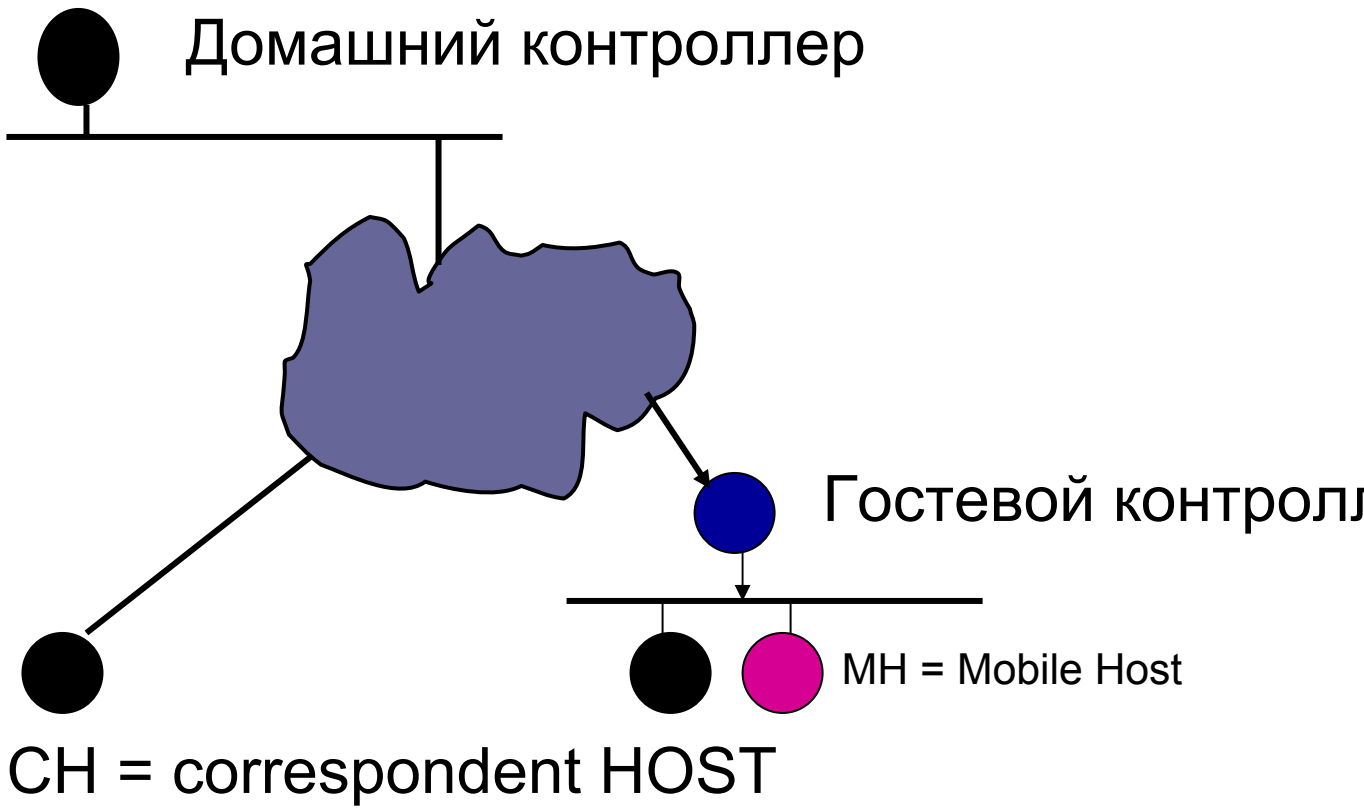
Мобильность и доступность соединения

- Новые проблемы для исследования
  - Постоянная связь для мобильного узла сети
  - Прозрачное перемещение между сетями
- Мобильные системы
  - Перемещение с места на место при постоянном подключении к беспроводной сети
  - Перемещение с места на место при роуминге между различными точками доступа
- Зачем поддерживать постоянное соединение?
  - Избежание перезапуска сетевых приложений

Проблемы IP адресации

- Узлы и интерфейсы Internet идентифицируются IP адресом
  - Сервисы доменных имен преобразуют имя узла IP адрес
  - IP адрес идентифицирует узел или интерфейс и определяет сеть, в которой он находится
  - Смешанное именование и местонахождение
- Перемещение в другую сеть требует изменение сетевого адреса
  - Это изменит идентификатор узла
  - Как можно передать данные этому узлу?

Основная идея



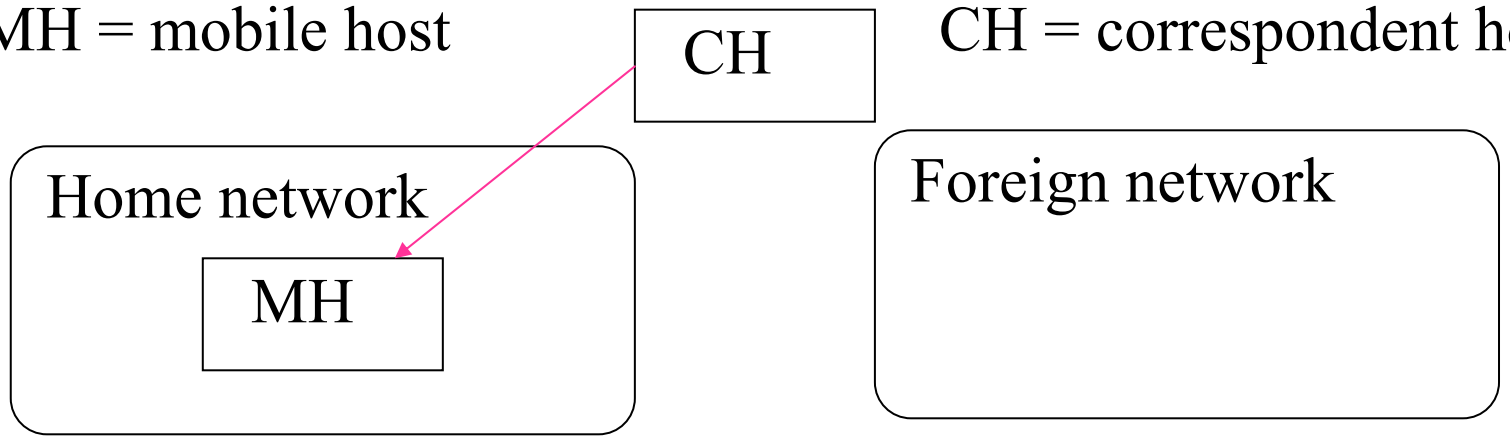
Основная идея

- Мобильный пользователи, подключенные к чужой сети получают адрес в гостевой сети
- Посредством DHCP
- Посредством гостевого контроллера
- Регистрация в локальном контроллере (LA)
- LA обладает списком гостевых узлов зарегистрированных в сети

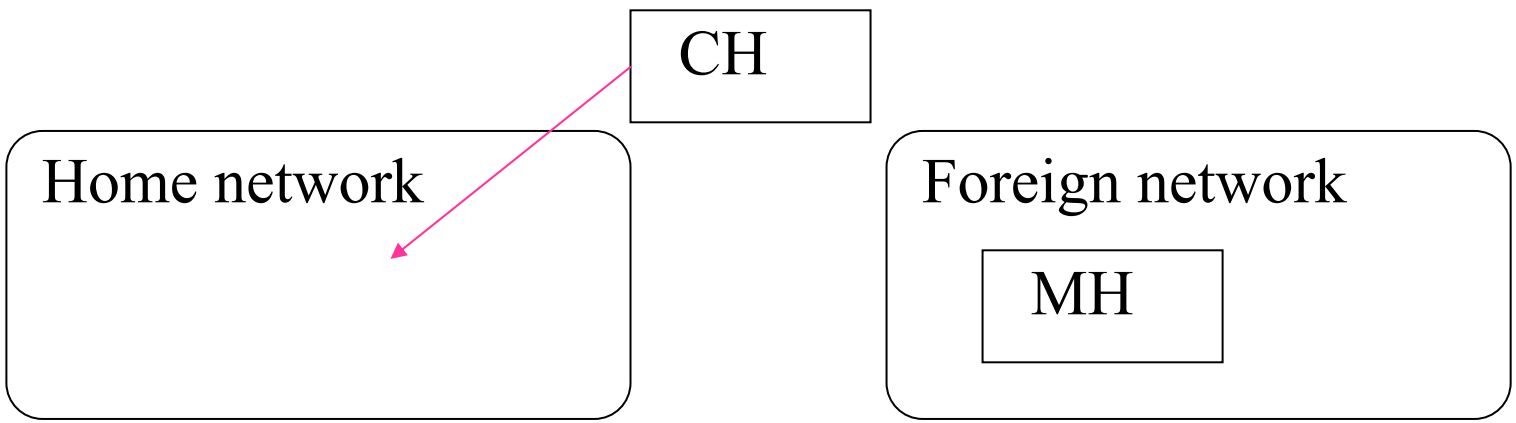
Маршрутизация для мобильных узлов

MH = mobile host

CH = correspondent host



Как прозрачно передать пакеты перемещающимся узлам?



Mobile IP (Dave Johnson, C Perkins)

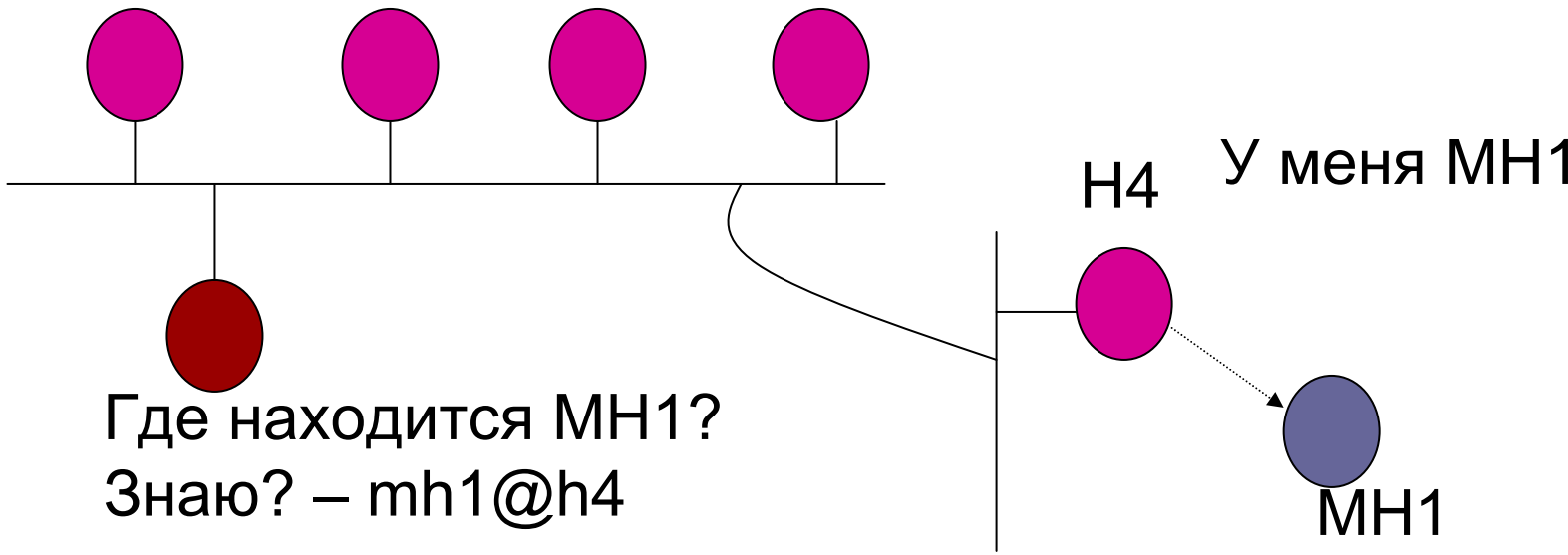
- Paper describes Internet Mobile Host protocol
- Correspondent hosts don’t need to know about mobility
- Allow a mobile host to send and receive packets with its permanent IP addres
- Maintain tcp connections across moves
- Simple
- Provides for route optimization
- Many possible techniques, many variants

Mobile IP (Dave Johnson, C Perkins)

- Статья описывает Internet Mobile Host протокол
- CH (узел-получатель) не нуждается в информации о мобильности
- Позволяет передавать и принимать мобильному узлу посредством постоянного IP адреса
- Поддерживает TCP соединения во время перемещения
- Простой
- Обеспечивает оптимизацию маршрутизации
- Множество возможных подходов и вариантов

Использование ARP

- Специально разработанный маршрутизатор proxy-arps для мобильного узла



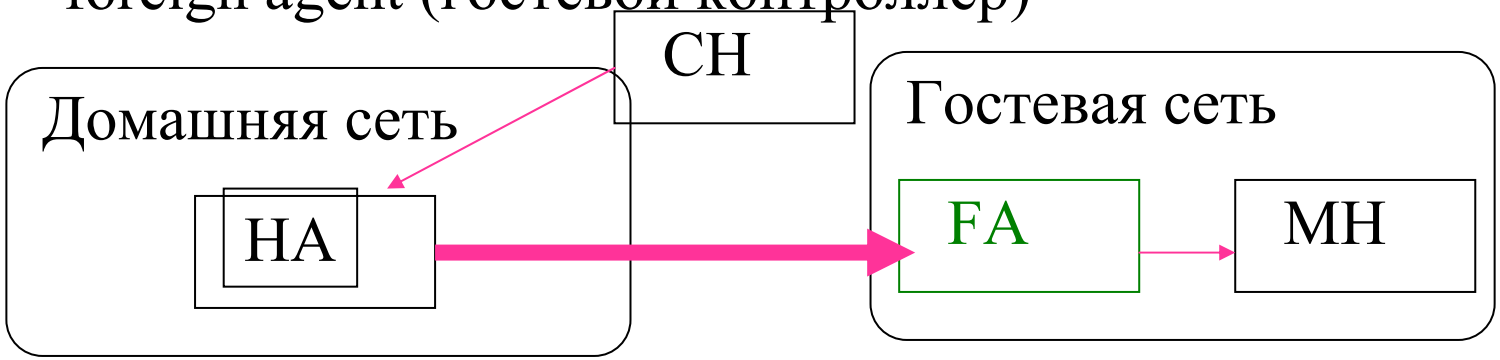
Mobile IP – передача мобильным узлам

MH = mobile host (мобильный узел)

CH = correspondent host (узел, с которым осуществляется связь, узел-корреспондент)

HA = home agent (домашний контроллер)

FA = foreign agent (гостевой контроллер)



- MH регистрирует свой новый адрес гостевой сети в HA, по которому с ним можно связаться, посредством FA
- HA туннелирует пакеты FA
- FA декапсулирует (разворачивает) пакеты и доставляет их MH

IP-in-IP (Инкапсуляция пакетов)

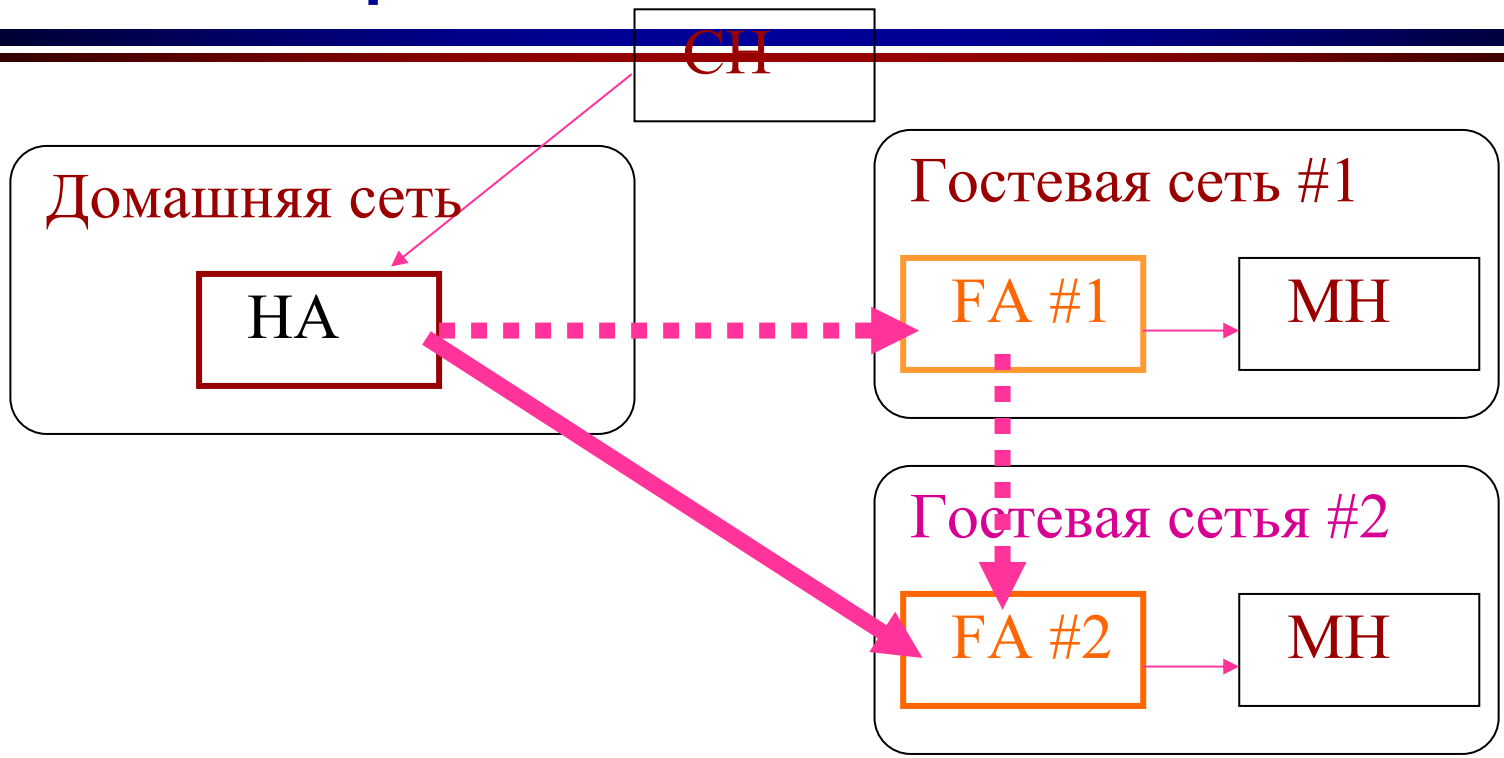
Пакеты от CH к MH

Source address = адрес CH
Destination address = домашний IP адрес MH
Тело пакета

HA перехватывает пакет и туннелирует его

Source address = адрес HA
Destination address = действительный адрес MH
Source address = адрес CH
Destination address = домашний IP адрес MH
Начальное тело пакета

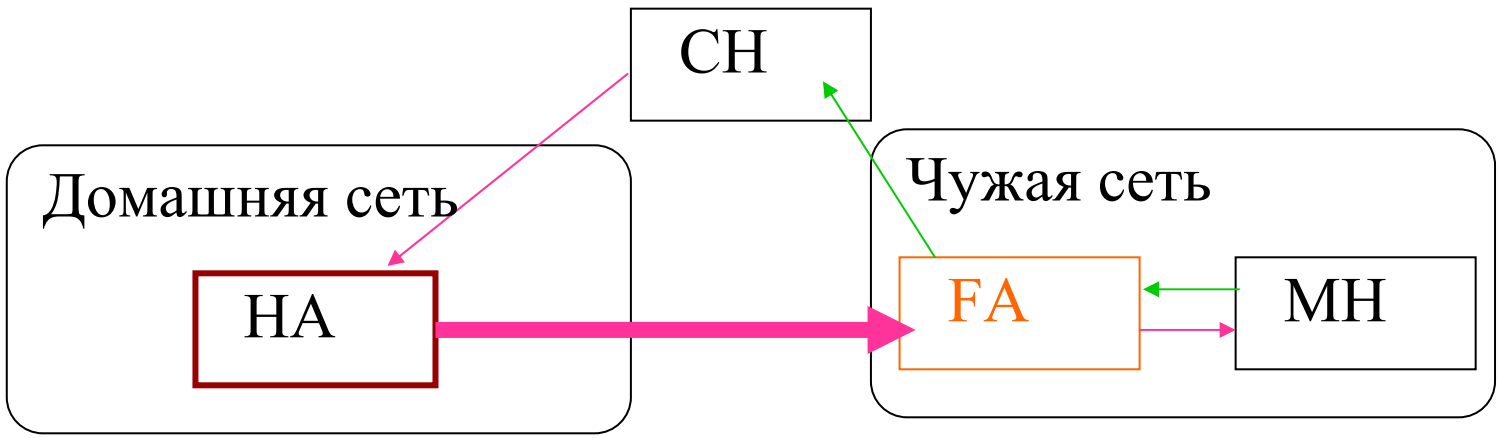
Когда мобильный узел перемещается снова



- MH регистрирует новый адрес (FA #2) посредством HA & FA #1
- HA туннелирует пакеты FA #2, который доставляет их MH
- Пакеты при передаче могут быть переадресованы от FA #1 к FA #2

Mobile IP – от мобильного узла

Мобильный узел также отправляет пакеты



•Мобильные узел домашний IP адрес как адрес источник

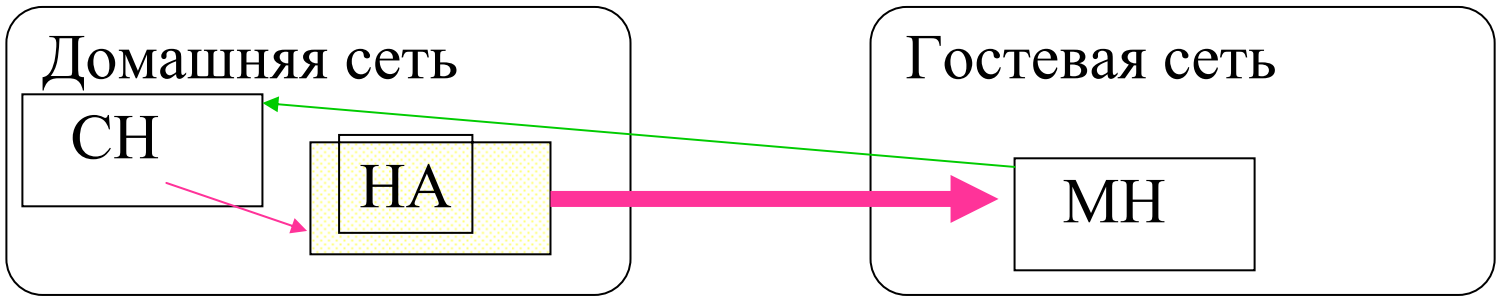
-Меньшая задержка

-Прозрачно для узла-корреспондента

-Нет очевидной потребности инкапсулировать пакеты для CH

• Это называется «треугольный маршрут»

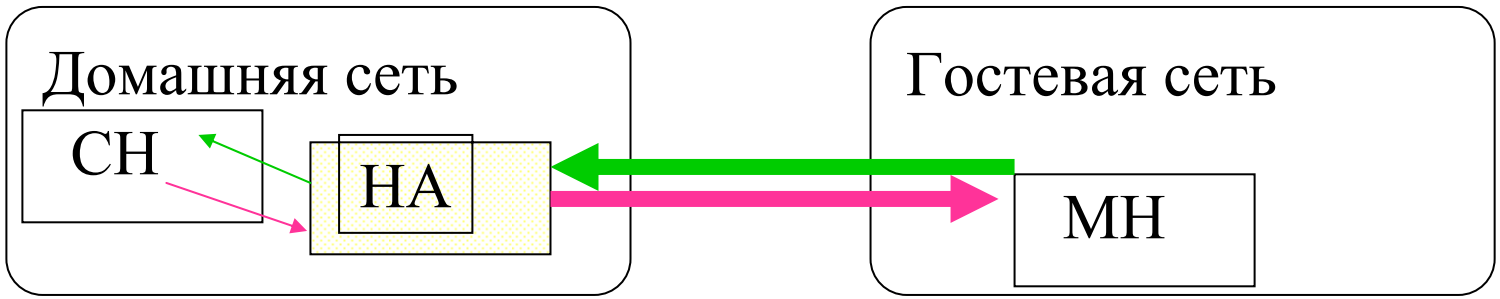
Проблемы фильтрации при вхождении\выхождении в гостевую сеть



•Мобильные узлы используют свой домашний IP адрес как адрес отправителя

•Безопасные пограничные маршрутизаторы отбросят такой пакет

Решение: двунаправленный туннель

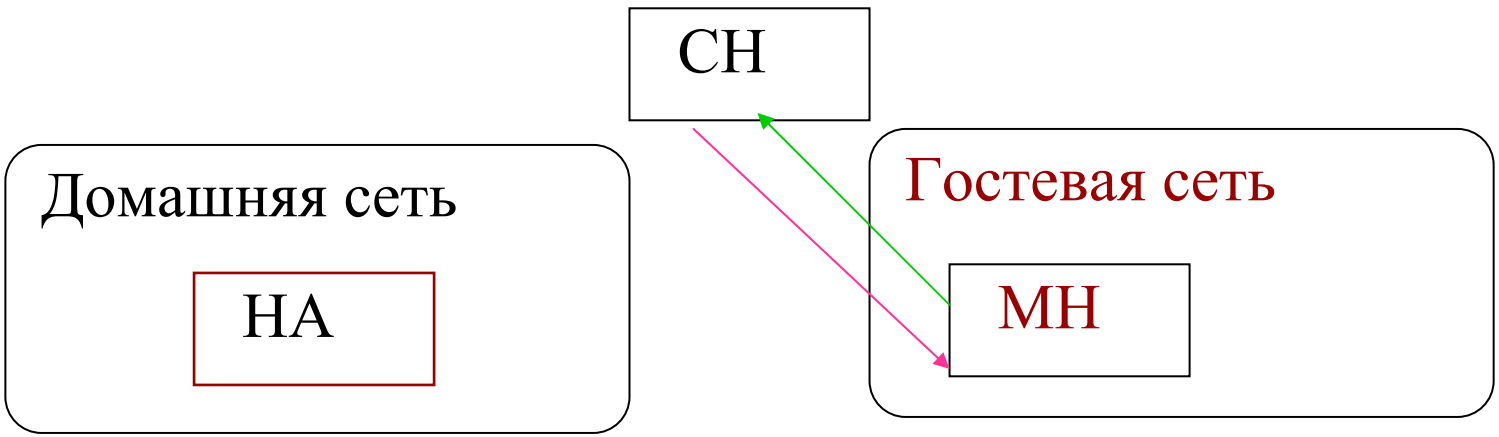


•Обеспечивает выбор безопасного маршрута через HA в оба направления

•Это самый медленный, но наиболее простой традиционный способ

Этот метод известен как «квадро-маршрутизация»

Решение: больше гибкости



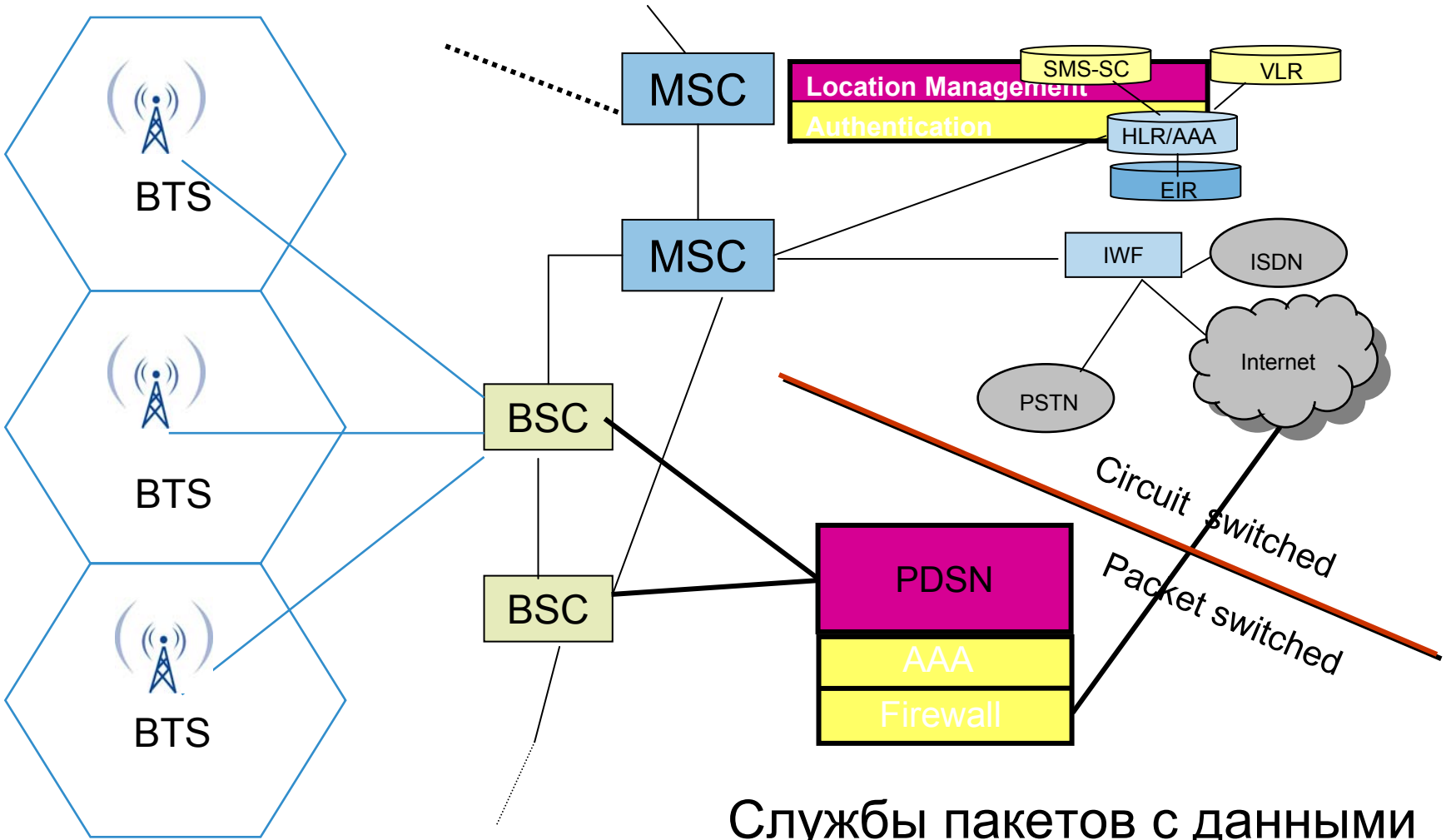
•Использование care-of адреса и непосредственная посылка пакета

-Это непосредственный IP адрес!

•В целом:

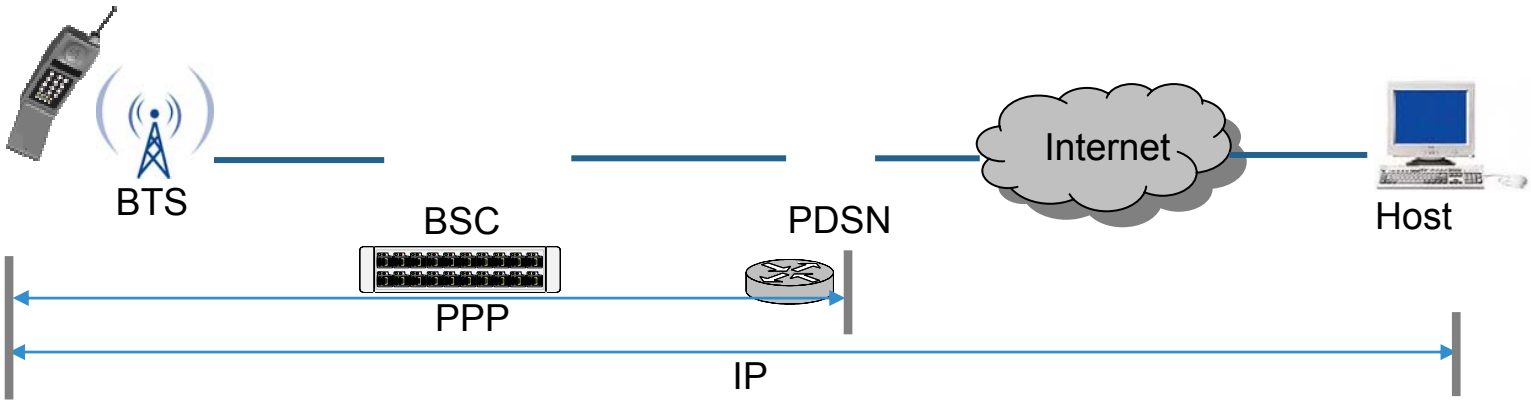
-MH должен обладать гибкостью для адаптации к условиям

Архитектура сети (1xRTT)



Службы пакетов с данными

Поддержка IP



- PDSN ограничивает PPP соединение
- IP назначается DHCP
  - IP принадлежит домену PSDN
- IP адрес изменяется при перемещении к новому PSDN
- PPP соединение должно быть инициировано мобильной станцией, а не сетью

Мобильный Internet 4x4.

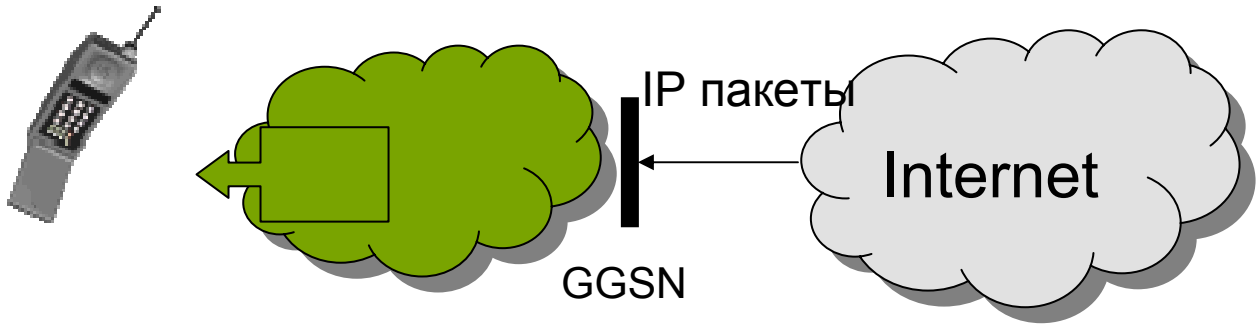
Proceedings of the ACM Mobility 4x4 (S. Cheshire, M. Baker *SIGCOMM'96 Conference)*

	Исходящий не прямой, Инкапсулированный	Исходящий прямой, Инкапсулированный	Исходящий прямой, Домашний адрес	Исходящий прямой, Временный адрес
Входящий непрямой, инкапсулированный	наиболее надежный, наименее эффективный	Требует декапсуляции в СН	Необходимо отсутствие безопасных маршрутизаторов на пути	
Входящий прямой, инкапсулированный		Требует полностью специализированного для мобильных приложений СН	Необходимо отсутствие безопасных маршрутизаторов на пути	
Входящий прямой, домашний адрес			Требует нахождение в одном сегменте сети	
Входящий прямой, временный адрес				Наиболее эффективный, отсутствие поддержки мобильности

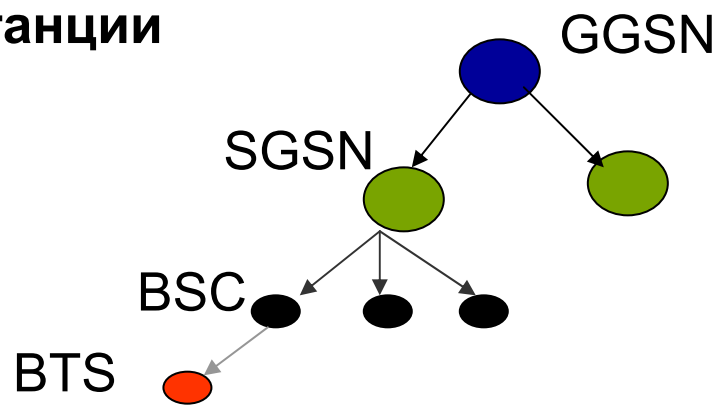
GPRS

- Сеть с коммутацией пакетов
- Доставляет пакеты мобильным устройствам
- Сосуществует с сетью GSM
- Сеть с коммутацией пакетов и коммутацией каналов, использующая один радио ресурс
- Отсутствие хранения и перенаправления для GPRS
- IP пакеты могут быть переданы и получены из сети GPRS

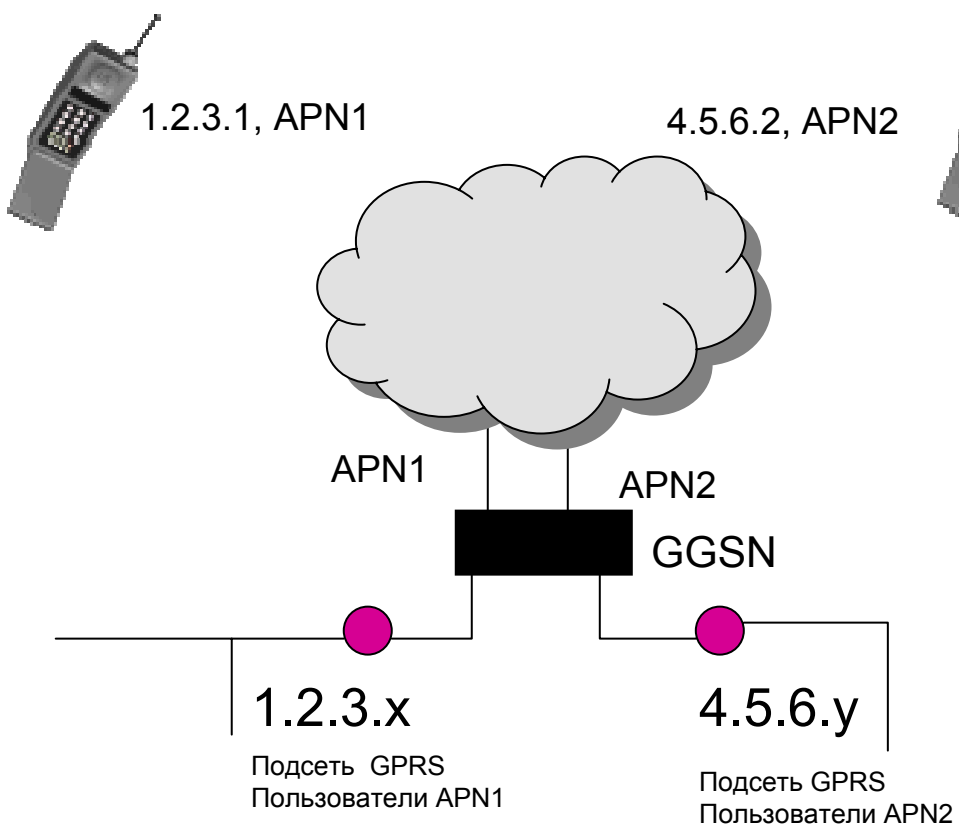
GPRS



Иерархия элементов сети маршрутизирет IP пакеты мобильной станции



Адресация в сети GPRS



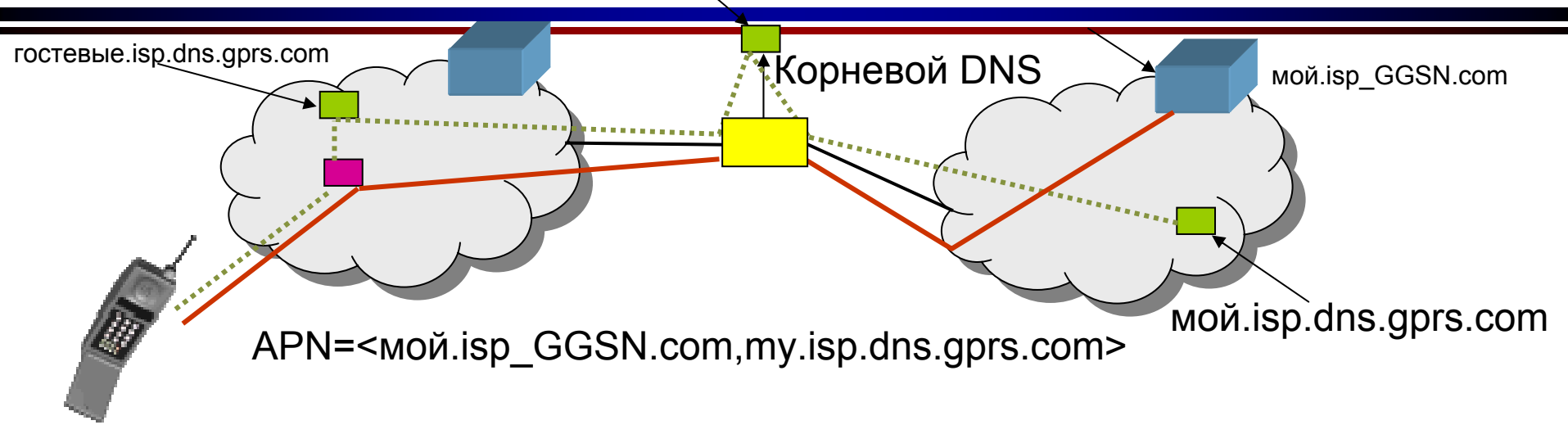
Маршрутизация IP пакетов к мобильному устройству

- Мобильный адрес принадлежит оператору
- APN является оператором сети и адреса получены из адресного пространства оператора сети
- Мобильные адреса принадлежат компании
- APN является точкой входа для GGSN
- Пограничный маршрутизатор должен быть подключен к GGSN и маршрут «по умолчанию» для адресов GPRS установлен на соединение с GGSN
- Частный мобильный адрес
- Публичный мобильный адрес

Роуминг

- Во время подключения к гостевому провайдеру, абонент может использовать APN, предоставленный домашней или гостевой сетью
- Пользователь может выбрать или APN гостевой сети, или APN домашней сети или иной вариант.
- Две возможных схемы маршрутизации, основанных на выборе APN:
  - 1) квадро-маршрутизация 2) треугольная маршрутизация
- В случае 1) SGSN находит домашний GGSN и тунеллирует все исходящие пакеты домашней GGSN
- Входящие пакеты для мобильной станции находят маршрут из домашней GGSN в гостевую GGSN
- В случае 2) исходящий траффик выходит через гостевую APN (GGSN)

Роуминг



- …. Найти IP адрес гостевой GGSN посредством локального DNS, корневого DNS, и домашнего DNS используя информацию в APN
- Установить IP туннель от SGSN к GGSN

Протоколы маршрутизации для ad-hoc сетей

- Два класаа
- Проактивные
  - Непрерывно обновляют информацию о доступности узлов в сети
  - Когда необходимо найти маршрут, он тут же доступен
    - DSDV by Perkins and Bhagwat (SIGCOMM 94)
    - Destination Sequenced Distance vector
- Реактивные
  - Построение маршрута происходит по мере надобности
  - Поддержка маршрутизации необходима для обеспечения информации о неправильных маршрутах
    - DSR by Johnson and Maltz
    - AODV by Perkins and Royer
- Гибридные
  - Zone routing protocol (ZRP)

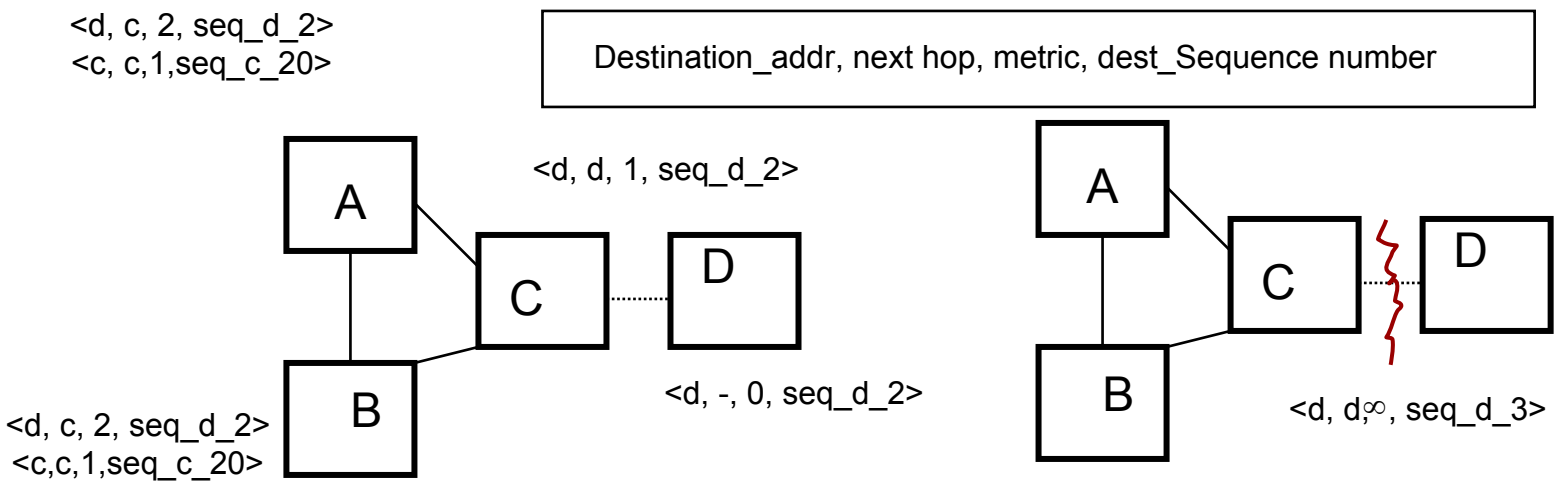
DSDV (Perkins, Bhagwat)

- Каждый узел поддерживает таблицу маршрутизации
- ID узла, количество хопов, номер последовательности (порождаемый узлом-получателем)
  - Похоже на RIP (за исключений номера последовательности)
  - Необходимо преодолеть проблему замедленной передачи «плохих новостей» (недоступность, перегрузка и т.п.) протоколом RIP
- Каждая мобильная станция объявляет о себе каждому из своих текущих соседей и своей таблице маршрутизации
- DSDV обеспечивает единственный путь для маршрутизации между каждой пары отправитель-получатель
  - Параметры: интервал обновления (частота широковещательной передачи), время стабилизации (как долго ждать перед переадресацией новых маршрутов), как долго ждать перед объявлением маршрута неактуальным

Номер последовательности

- DSDV связывает каждый маршрут с номером последовательности и считает маршрут с большим значением номера последовательности более предпочтительным, но имеющим меньшую метрику
- Каждый узел в сети объявляет о себе монотонно увеличивая свой четный номер последовательности
- Когда узел B решает, что маршрут до узла C исчез, он объявляет маршрут до узла D с бесконечной метрикой и нечетным номер последовательности (на 1 больше, чем предыдущее значение)

DSDV



- Новые обновления отправляются как нечетные номера
- Нарушенные соединения отправляются как четные номера (на единицу больше, чем отправленный D)
- Заметим, что <d, d, ∞ ,seq_d_3> генерируется узлом сети C
- Когда узел получает обновление с бесконечной метрикой и предыдущий номер последовательности с конечной метрикой меньшей, чем метрика обновления, информация распространяется об этом немедленно ∞
- Информация передается быстро и используется всеми узлами для обнаружения нарушенного канала

Распространение маршрутной информации

- При получении информации о новом маршруте может быть более целесообразно подождать выявления маршрута с лучшей метрикой
- Использовать маршрут с предыдущим номером последовательности для маршрутизации, но подождать объявлять об этом маршруте
- Две таблица: Маршрутная таблица и таблица объявлений маршрутов
- Поддерживать текущее среднее значение времени для недавних обновлений
- Выждать время равное beta* среднее время установки маршрута для этого получателя

Проблемы

- Когда инициировать обновление маршрутной информации
- При получении бесконечной метрики?
  - немедленно
- При получении нового номера последовательности
  - Непонятно, что делать(немедленно или отложить)
- При получении новой метрики
  - Подождать некоторое время перед распространением
  - Но использовать новый маршрут для перенаправления
- Подходит для малой и средней мобильности

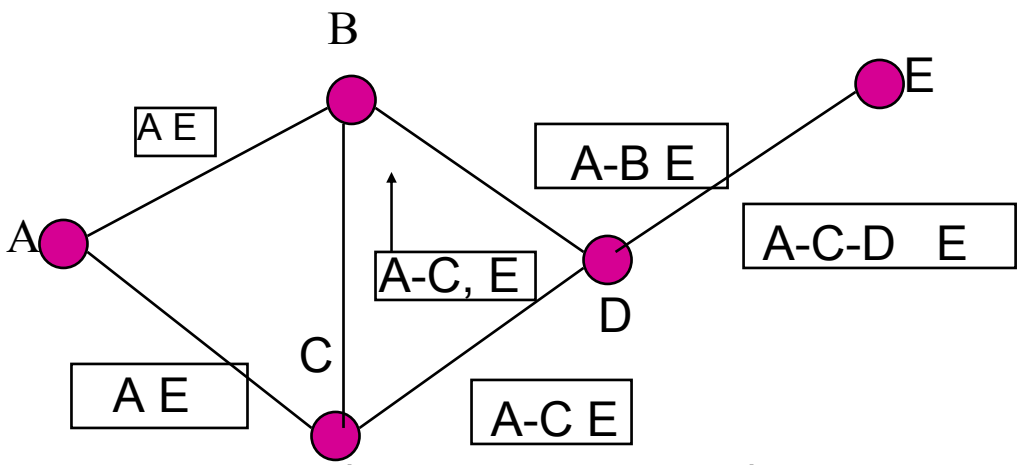
Dynamic source routing (DSR) Johnson, Maltz, Broch

- Реактивный протокол маршрутизации
- Избегает больших периодических обновлений
  - Преодолевает проблемы связанные с протоколами, передающими много данных, для беспроводных сетей (мощность, полоса пропускания, избыточность)
- Маршруты определяются как законченные пути от отправителя к получателю
- Промежуточные узлы не нуждаются в обновлении маршрутной информации
- Отсутствие периодического распространения маршрутной информации, отсутствие протоколов обнаружения соседей

Обнаружение маршрута

- Источник наводняет сеть пакетами обнаружения маршрута
- Пакет RREQ идентифицирует узел получатель
- Если процедура обнаружения маршрута успешна, узел отправитель получает ответный маршрутный пакет, в котором перечислена последовательность прыжков, с помощью которых получатель может быть достигнут
- Некоторые другие узлы, обладающие информации о получателе, так же могут ответить
- Узлы запоминают\подслушивают маршруты
  - Маршрутный кэш используется для ограничения распространения маршрутных запросов

Обнаружение маршрута



- Маршрутный ответ может быть послан как обратный маршрут
- Или послан по любому другому маршруту
- Или передан как новый пакет запроса маршрута к источнику

Маршрутный ответ в DSR

- Маршрутный ответ может быть отправлен по обратному пути в Route Request (RREQ) только в случае, если каналы гарантированно двунаправлены
  - Для подтверждения этого, RREQ должна быть переадресована только если соединение, по которому он был получен, двунаправленное
- Если разрешены однонаправленные (асимметричные) соединения, то RREP может нуждаться в поиске маршрута от получателя к отправителю
  - Если получателю неизвестен маршрут к узлу отправителю
  - Если обнаружение маршрута было инициировано получателем для маршрута к отправителю, тогда маршрутный ответ совмещается с маршрутным запросом от отправителя к получателю

Оптимизация DSR: Кэширование маршрутов

- Каждый узел сети кэширует новый маршрут, информация о котором получена любым способом
- S посылает RREQ и получает RREP во время поиска маршрута до D
  - Когда S находит маршрут [S,A, B, C,D] к узлу D, узел S также обнаруживает маршрут [S,A,B] до узла B и [S, A, B, C] до узла C
- D принимает RREQ от других узлов
  - Когда узел D получает RREQ [S,A,B,C] предназначенный узлу C, узел D получает маршрут [B, A,S] до узла S
- D перенаправляет RREP некоторому узлу
  - Когда узел D перенаправляет RREP [S,A,D,C,F], узел D получает маршрут [D,C,F] до узла F
- Когда узел B перенаправляет данные [S,A, B, C,D] он получает маршрут [C,D] до узла D

Использование кэширования маршрутов

- Когда узел S узнает, что маршрут до узла D более не существует, он использует иной маршрут, имеющийся в кэше. Если такой записи нет, узел S инициирует поиск маршрута путем посылки запроса на маршрут.
- Узел X при получении Route Request для некоторого узла D может послать Route Reply, если узел X знает маршрут к узлу D
- Использование кэширования
  - Может ускорить поиск маршрута
  - Может уменьшить распространение запросов на маршрут

Обратная сторона кэширования маршрутов

- Устаревший кэш может привести к потери производительности
- С течением времени и изменением положения узлов, кэшированные маршрут могут стать неправильными
- Отправитель может попробовать несколько неправильных маршрутов из своего кэша или ответов из кэшей других узлов, пока не найдет правильный

Сравнение эффективности

- DSDV хорошо работает при небольшой мобильности узлов
  - Высокий уровень доставки пакетов (малые потери)
  - Не работает при большой мобильности узлов
- DSR хорошо работает при любой мобильности узлов
  - Большие накладные расходы на поддержание таблиц маршрутизации и управляющих пакетов
  - Расширяем для больших сетей

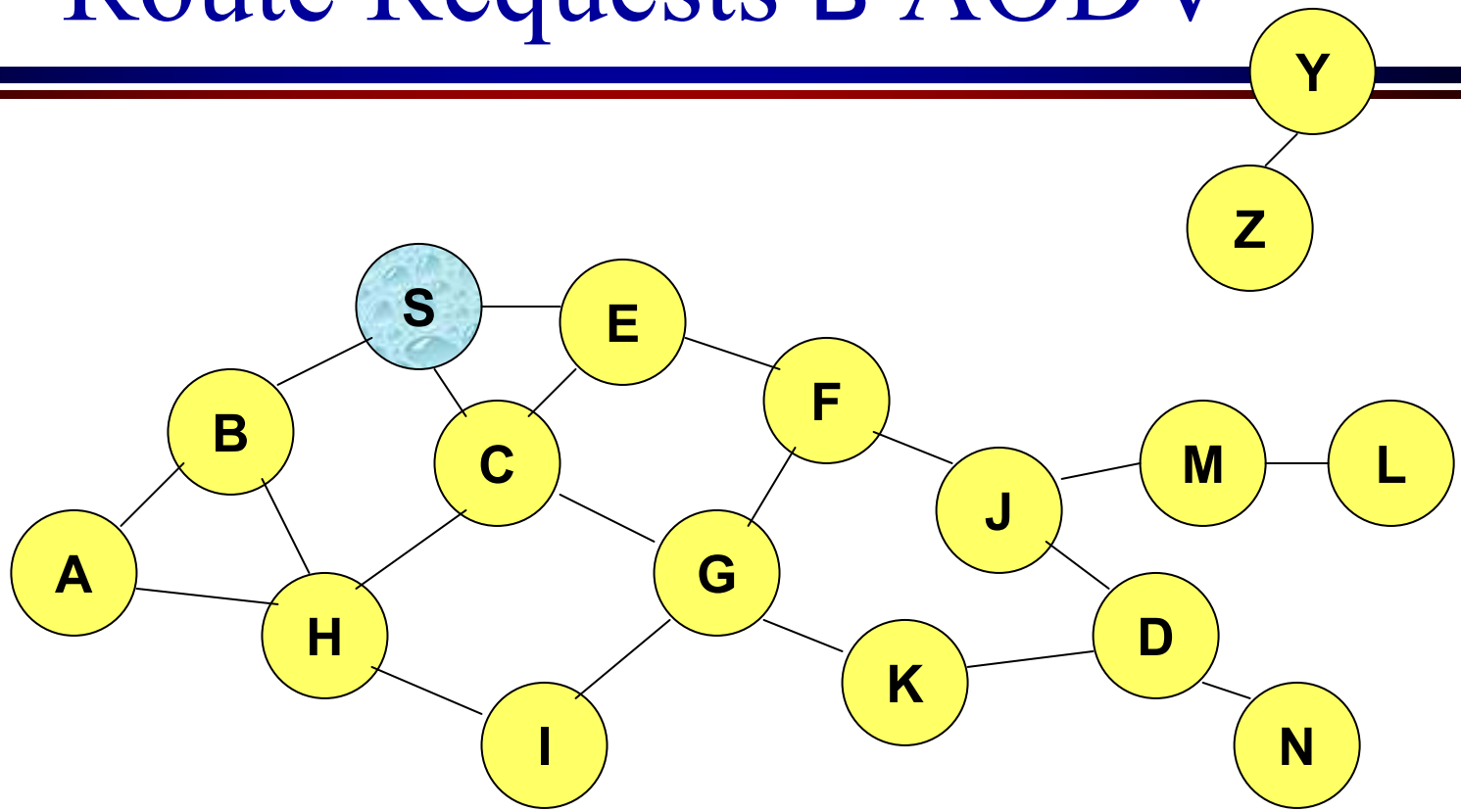
Ad Hoc On-Demand Distance Vector Routing (AODV) [Perkins и Royer]

- DSR включает маршрутную информацию в заголовки пакетов
- Большие заголовки уменьшают эффективность работы сети
  - Особенно когда размер передаваемых данных невелик
- AODV пытается улучшить DSR путем хранения маршрутных таблиц в узлах, таким образом пакеты данных не должны содержать данные о маршрутах
- AODV поддерживает желательные особенности DSR, связанные с тем, что маршруты поддерживаются только между узлами, которым нужна связь

AODV

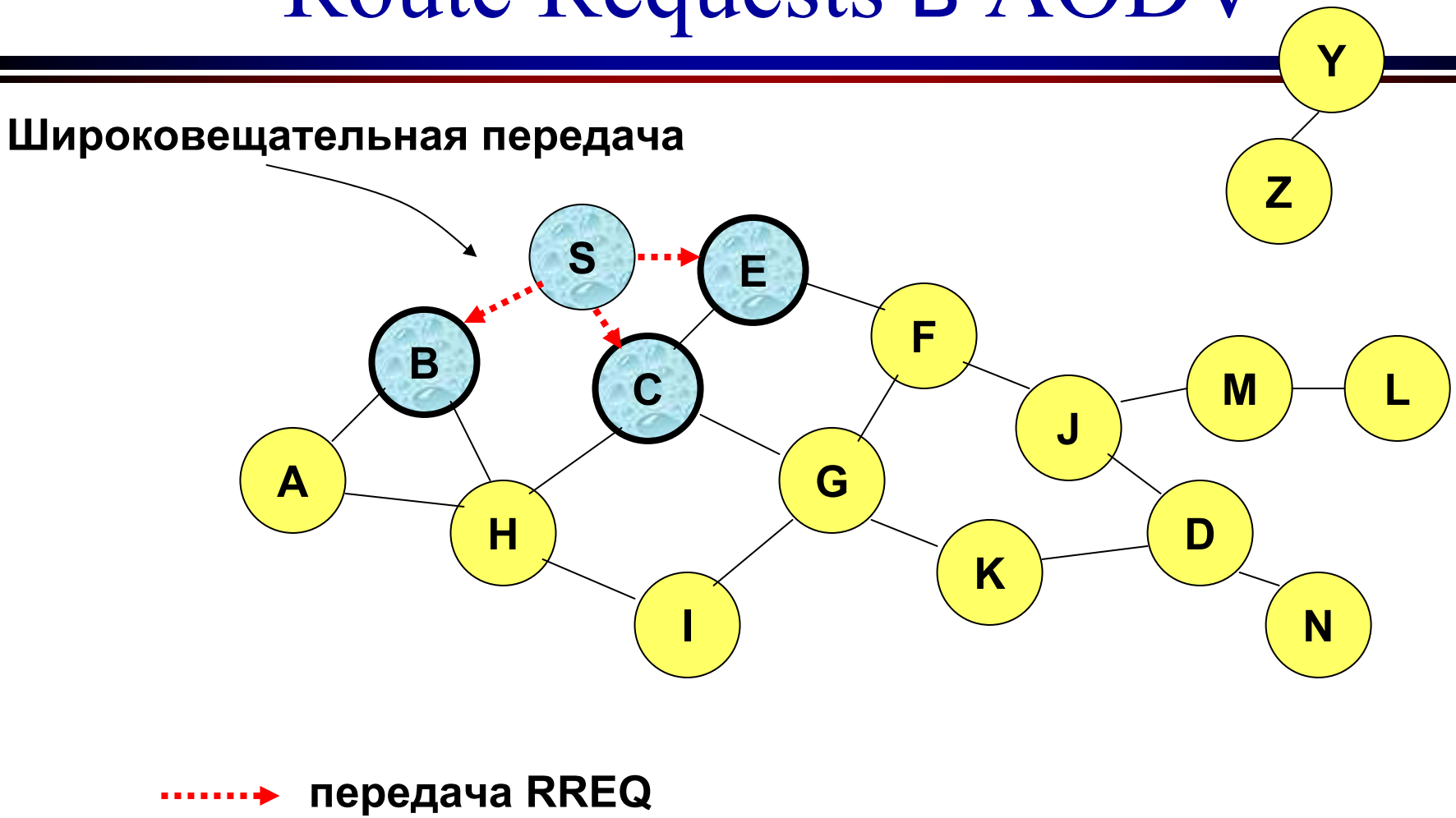
- Route Requests (RREQ) передаются почти так же, как и в DSR
- Когда узел повторно широковещательно передает Route Request, он устанавливает обратный маршрут по направлению к источнику
  - AODV предполагает наличие симметричных двунаправленных связей
- Когда получатель принимает Route Request, он отвечает посылкой Route Reply
- Route Reply перемещается обратно вдоль пути, по которому был передан Route Request

Route Requests в AODV

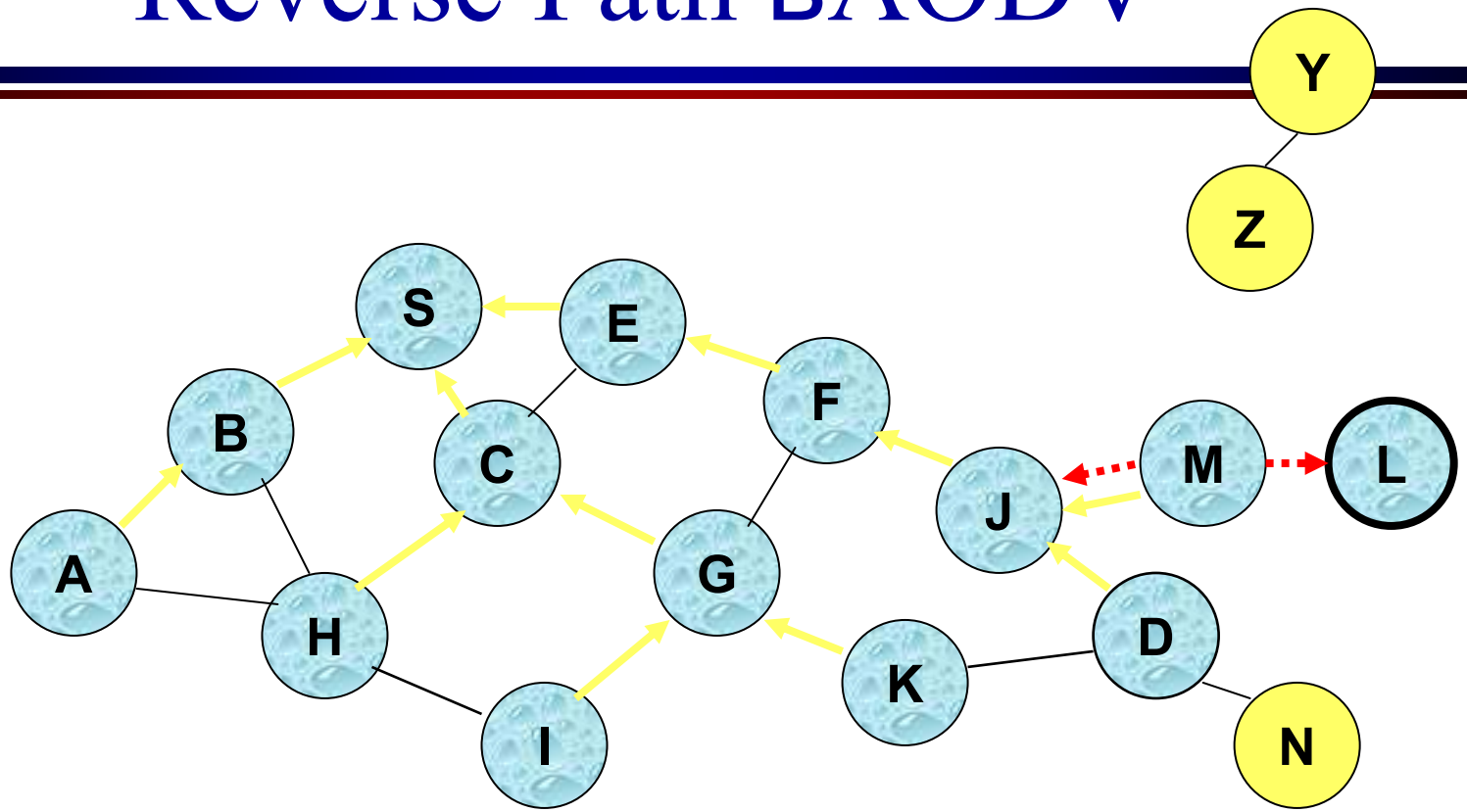


Узел, получивший RREQ для D от S

Route Requests в AODV

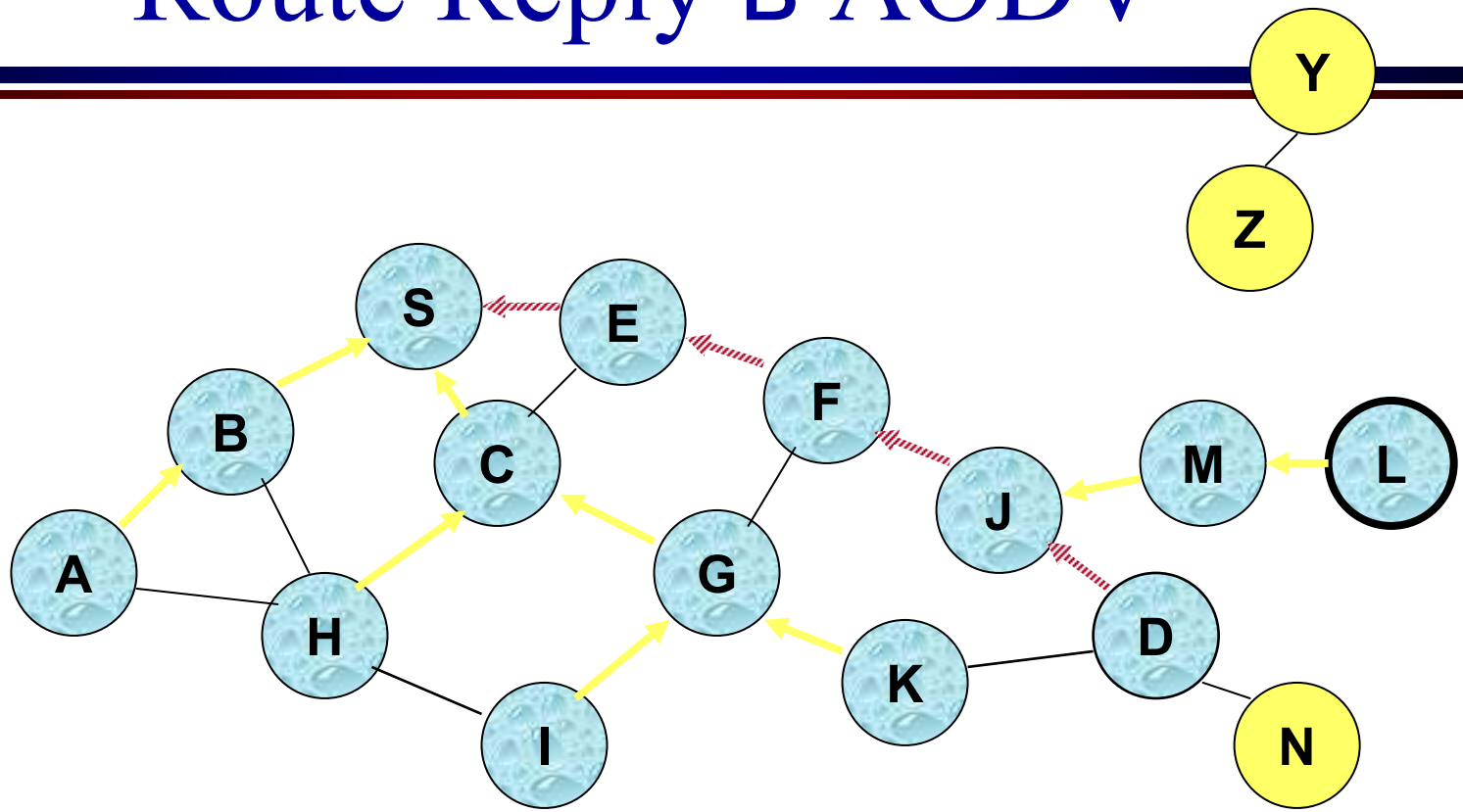


Reverse Path вAODV



- **Узел D не перенаправляет RREQ, так как узел D является целью поиска маршрута**

Route Reply в AODV



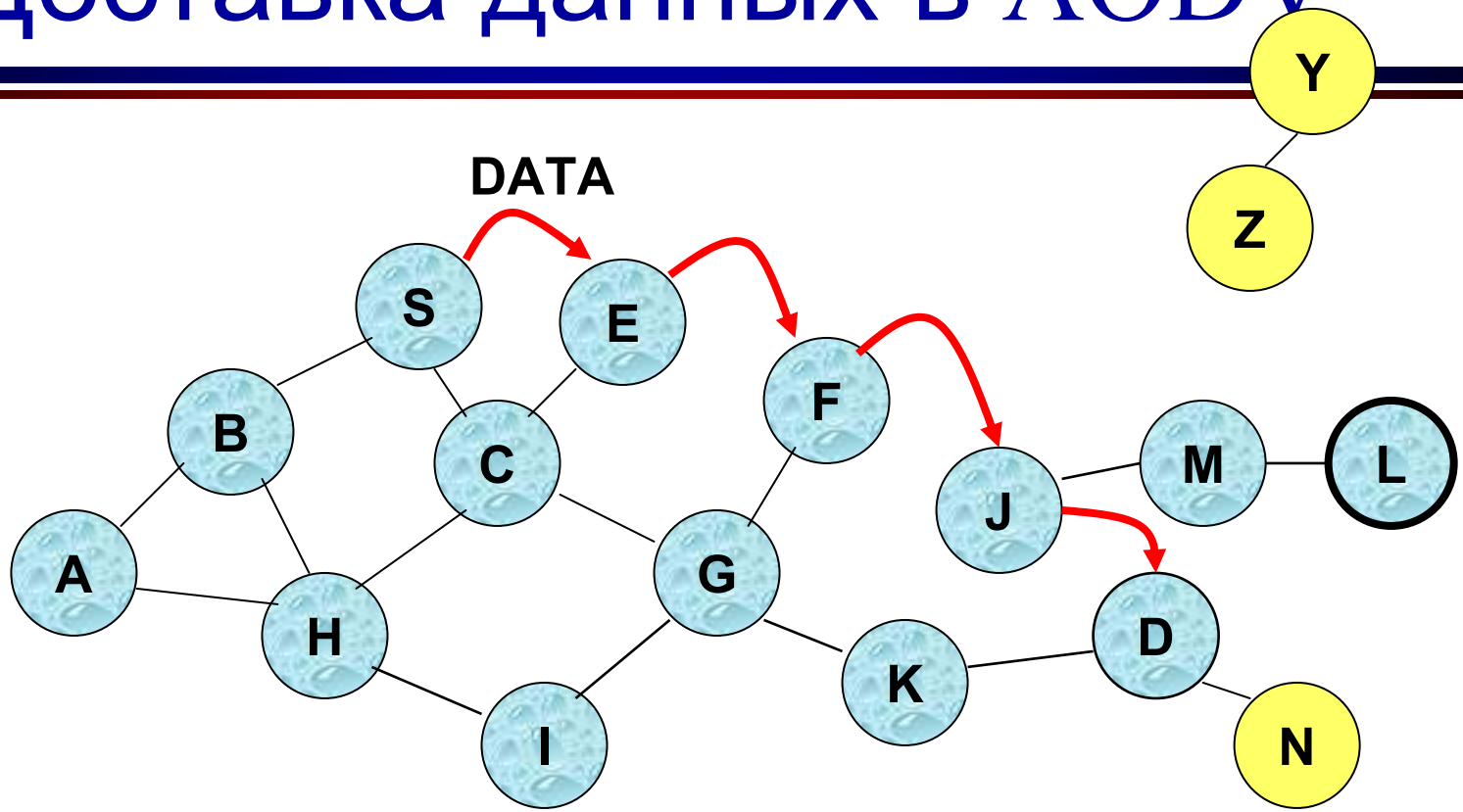
Каналы, по которым пройдет RREP

Route Reply в AODV

- Промежуточные узлы (не получатель) может так же послать Route Reply (RREP), если он знает более новый путь, чем известный отправителю S

- Для того, чтобы определить известен ли маршрут промежуточному узлу, используются *destination sequence numbers* (номер последовательности получателя)
- Вероятность того, что промежуточный узел пошлет Route Reply при использовании AODV не настолько велика, как в случае DSR
  - Новому Route Request, посланному узлом S для получателя, назначается более высокое значение номера последовательности. Промежуточный узел, имеющий информацию о маршруте, но меньший номер последовательности, не может послать Route Reply

Доставка данных в AODV



Таблицы маршрутизации используются для переадресации пакетов.

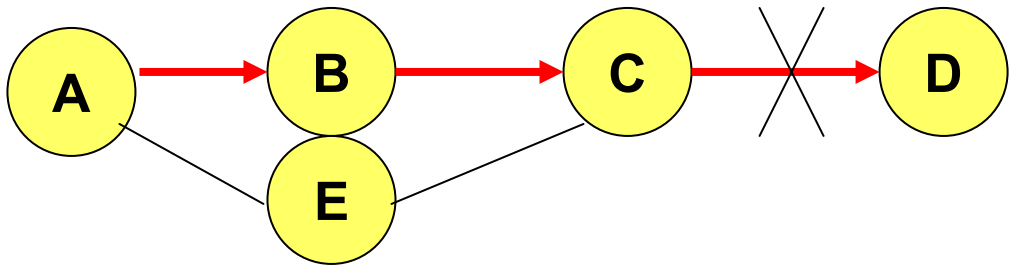
Маршруты не включаются в заголовок пакетов.

Destination Sequence Number (номер последовательности)

- Когда узел D принимает Route Request с номером последовательности N, узел D устанавливает номер последовательности в N, если он уже не больше N

Зачем нужны номера последовательностей в AODV

- Чтобы избежать старых несуществующих маршрутов
  - Чтобы определить, какой маршрут новее
- Чтобы избежать маршрутов-колец



  - Предположим, что A не знает об отказе канала C-D так как RERR, отправленный C потерян
  - C начинает поиск маршрута до D. Узел A принимает RREQ (например, C-E-A)
  - Узел A будет отвечать до тех пор, пока знает маршрут до D через B
  - Результат - цикл (например, C-E-A-B-C)

Выводы: AODV

- Не требуется включать маршруты в заголовки пакетов
- Узлы поддерживают таблицу маршрутизации только для активных маршрутов
- В большинстве случаев адрес следующего узла для каждого получателя храниться в таблице узлов
  - DSR может поддерживать несколько маршрутов для одного получателя
- Неиспользуемые маршруты устаревают даже если топология не меняется

Location-Aided Routing (LAR)

- Использует информацию о положении узлов для улучшения механизмов поиска маршрутов
- Использует информацию о положении для ограничения флуда при посылке запросов на поиск маршрута
  - Информация о положении может быть получена с помощью GPS
- Когда S пытается найти маршрут до D, S вычисляет ожидаемую зону для D
- Ожидаемая зона определяется как регион, в котором ожидается обнаружить текущее положение получателя
  - Ожидаемый регион определяется на базе старой информации о положении узла и информации о скорости объекта
- Route requests ограничены зоной запросов, в которую входят ожидаемая зона и местоположение отправителя

Концепция двух зон

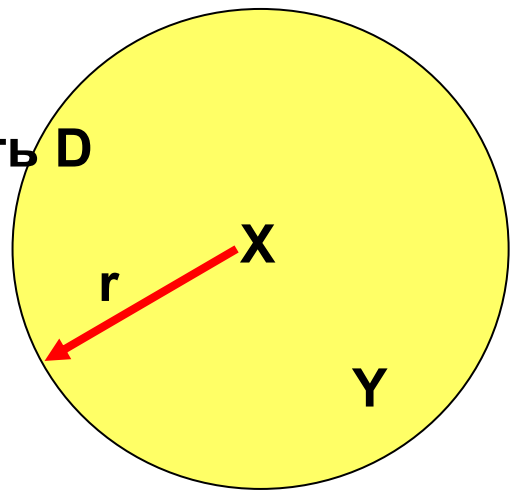
- LAR использует знание ожидаемой зоны, где отправитель имеет информацию о местонахождении получателя D
  - Если нет, используется DSR
- LAR использует информацию о местонахождении для ограничения области поиска нового маршрута – зона запроса
- Процедура похожа на DSR : LAR выполняет поиск маршрута при помощи ограниченного флудинга

Ожидаемая зона в LAR

X = последнее известное местоположение D, в момент времени t_0

Y = текущее положение узла D в момент времени t_1, неизвестно узлу S

r = (t_1 - t_0) * предполагаемая скорость D



Ожидаемая зона

Зона запроса в LAR

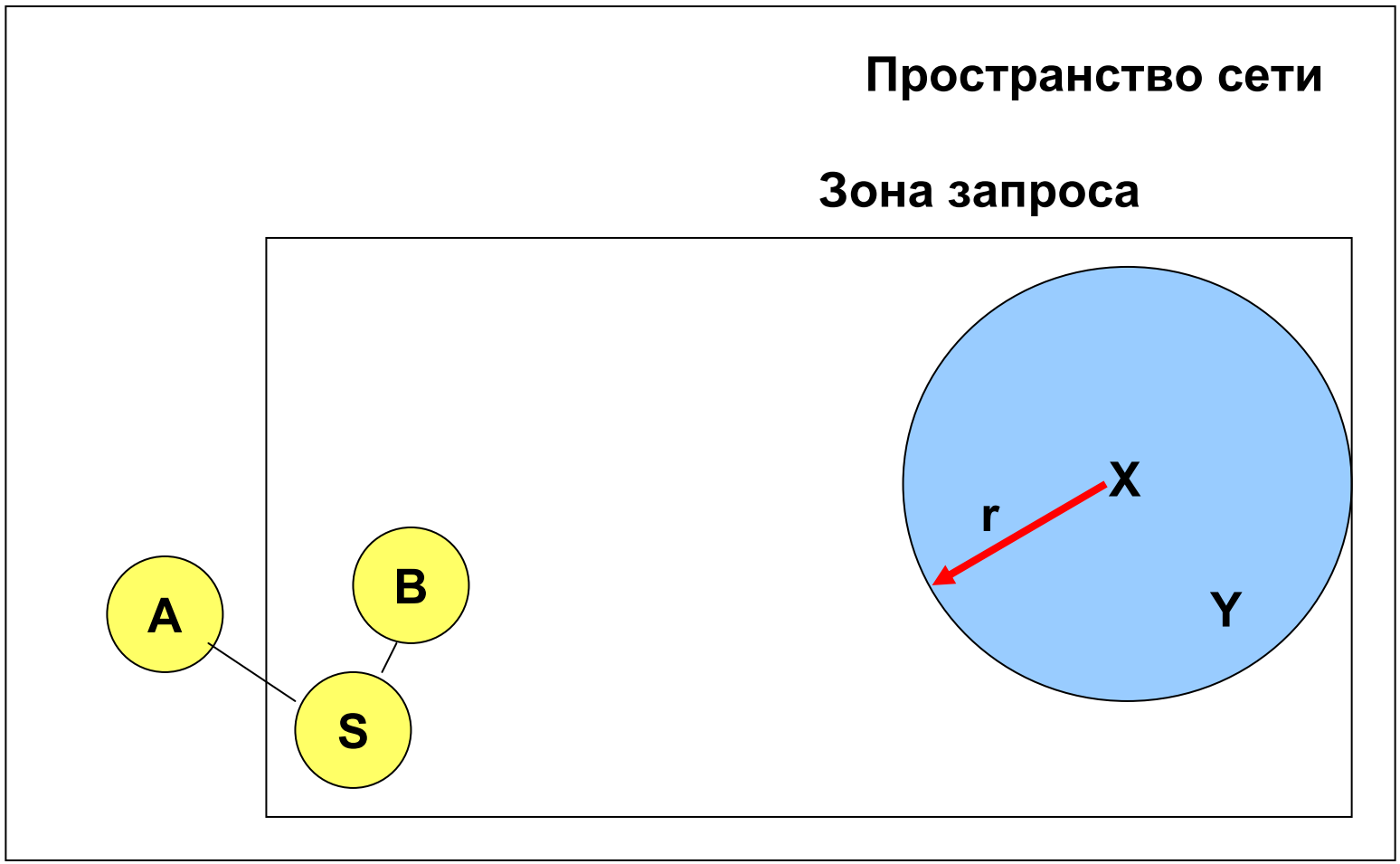


Схема 1 LAR

Определение зоны запроса

- Схема 1:
- Вычисление круглой области в котором должен быть обнаружен получатель
- Во время процедуры поиска маршрута, только узлы в этой зоне перенаправляют запрос
- RREP может использовать такую же процедуру для посылки ответа на запрос

LAR

- Только узлы с зоне запроса перенаправляют запросы маршрутов
- Если процедура обнаружения маршрута с использованием малой зоны запроса неудачна, отправитель инициирует процедуру поиска маршрута (после паузы) в большей зоне
  - Большая зона может быть всей сетью
- Остальные механизмы такие же, как и в DSR

Схема 2 LAR

- Вычисляет предполагаемое расстояние до получателя
- Расстояние включается в сообщение route request
- Узел передает сообщение дальше только если расстояние до получателя (имеющееся у этого узла) меньше или равно расстоянию, записанного в сообщении (или как минимум на δ больше)
- Поле расстояния обновляется перед передачей пакета

Схемы LAR

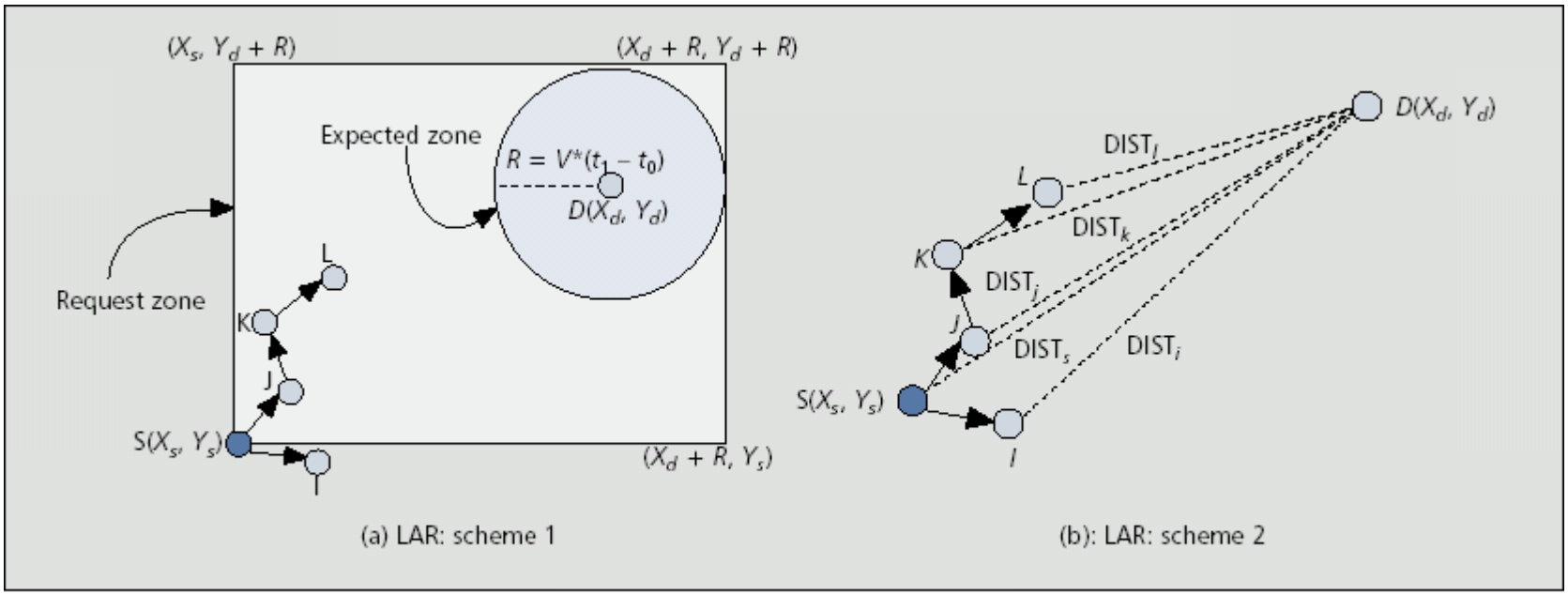
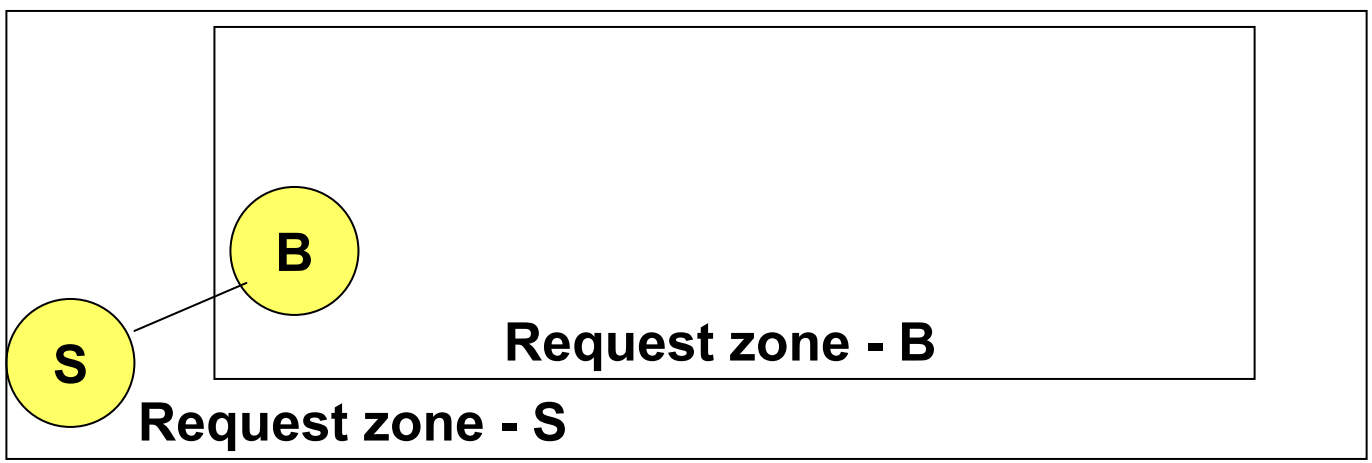


Figure 6. *LAR: limited flooding of route request: a) scheme 1: expected zone; b) scheme 2: closer distances.*

Варианты LAR: Адаптивная зона запроса

- Каждый узел может изменить зону запроса, указанную в перенаправленном запросе
- Измененная зона запроса может быть определена, используя более актуальную информацию и может быть меньше, чем первоначальная



Варианты LAR: Подразумеваемая зона запроса

- В предыдущей схеме, route request непосредственно указана зона запроса
- Альтернативный подход: Узел X перенаправляет route request полученный от узла Y если X считает себя ближе к ожидаемой зоне по сравнению с Y
- Это способ пытается переместить route request физически ближе к получателю после каждой перепосылки

Location Aided Routing (LAR)

- Достоинства
  - Уменьшает размер области, в которой идет лавина
  - Уменьшает накладные расходы
- Недостатки
  - Узлы должны знать свои координаты
  - Не учитывается возможное существование преград для радиопередач

Гибридные протоколы

Zone Routing Protocol (ZRP) [Haas98]

Zone routing protocol содержит

- Проактивный протокол: которые про-активно обновляет состояние сети и поддерживает маршруты независимо от наличия траффика на маршруте
- Реактивный протокол: только определяет маршрут до получателя если есть данные для передачи

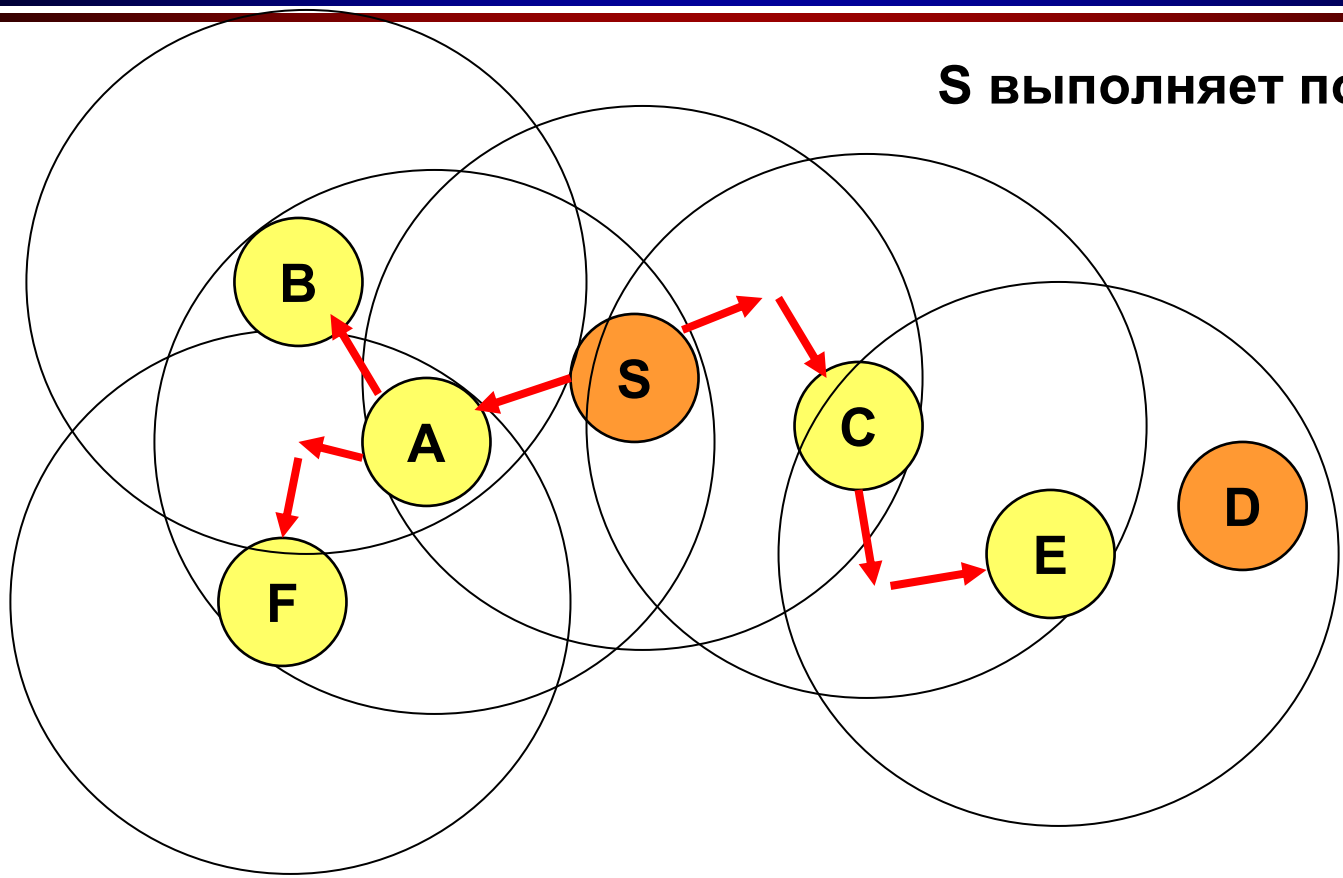
ZRP

- Все узлы на расстоянии *d* от узла X считаются в зоне маршрутизации узла X
- Все узлы на точном расстоянии *d* считаются периферийными узлами зоны маршрутизации X

ZRP

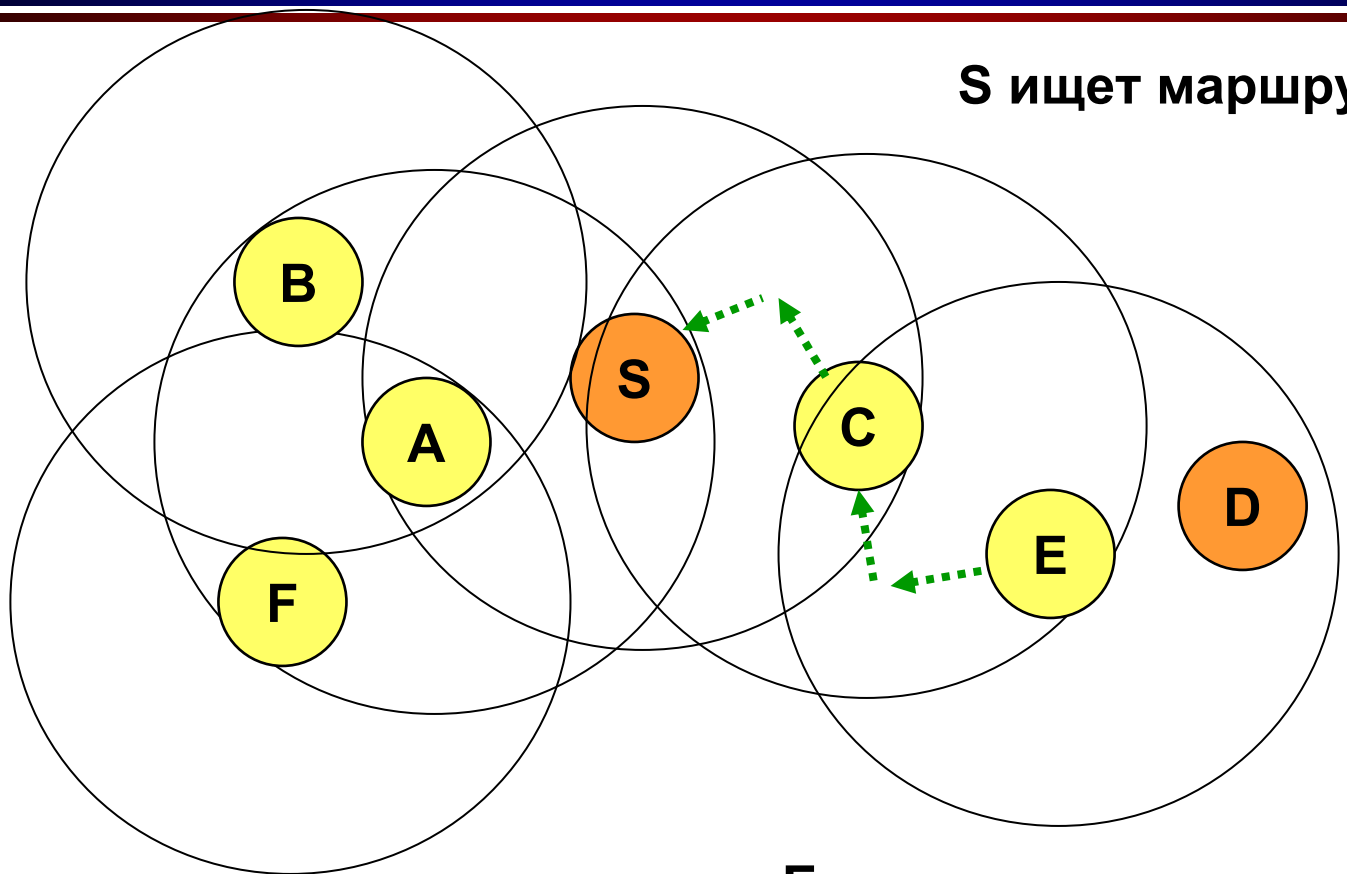
- Маршрутизация вне зоны: Проактивно поддерживать состояние информационных соединений для связей на коротких расстояниях от любого узла
  - Маршруты до узлов на коротких расстояниях таким образом поддерживаются проактивно (используя, например, протокол link state или distance vector)
- Маршрутизация внутри зоны: Использует протокол обнаружения маршрута для обнаружения маршрутов к далеким узлам. Поиск похож на DSR за исключением того, что запрос распространяется через периферийные узлы

ZRP: пример
Радиус зоны = $d = 2$



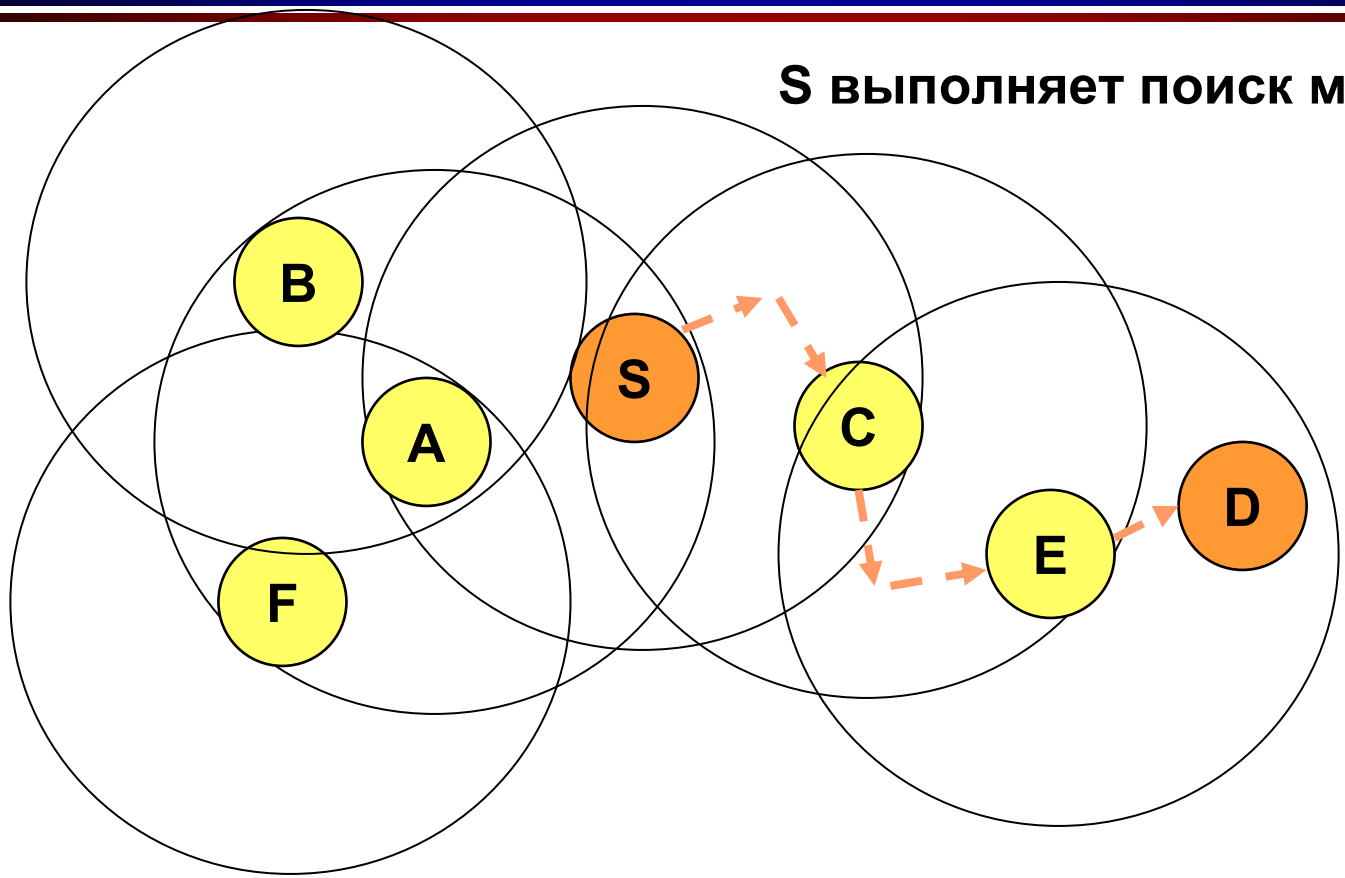
ZRP: пример

$d = 2$



ZRP: Пример

$d = 2$



Маршрут, по которому передаются данные