

Практическое занятие № 12. Протоколы динамической маршрутизации.

Цель работы: изучение принципов работы протоколов динамической маршрутизации в сетях связи.

Известно, что маршрутизаторы используют таблицу маршрутизации для определения, куда направить данные. В этой таблице содержится информация о сетевых путях, которая указывает, какие сети подключены к каким интерфейсам маршрутизатора.

При получении пакета данных маршрутизатор ищет в таблице маршрутизации адрес назначения. Если он находит запись о сетевом пути для этого адреса, то перенаправляет пакет через соответствующий интерфейс. В случае отсутствия записи для адреса назначения, пакет будет отброшен.

Различают протоколы, выполняющие статическую и динамическую маршрутизацию. При статической маршрутизации все записи в таблице имеют неизменяемый, статический статус, что подразумевает бесконечный срок их жизни. Записи о маршрутах составляются и вводятся в память каждого маршрутизатора вручную администратором сети. При изменении состояния сети администратору необходимо срочно отразить эти изменения в соответствующих таблицах маршрутизации, иначе может произойти их рассогласование и сеть будет работать некорректно.

При динамической, ее еще называют адаптивной, маршрутизации все изменения конфигурации сети автоматически отражаются в таблицах маршрутизации благодаря протоколам маршрутизации. Эти протоколы собирают информацию о топологии связей в сети, что позволяет им оперативно отражать все текущие изменения. В таблицах маршрутизации при динамической маршрутизации обычно имеется информация об интервале времени, в течение которого данный маршрут будет оставаться действительным. Это время называют временем жизни (TTL) маршрута. Если по истечении времени жизни существование маршрута не

подтверждается протоколом маршрутизации, то он считается нерабочим, и пакеты по нему больше не посылаются. Протоколы динамической маршрутизации бывают внешними (распределенном) и внутренними (централизованном).

При распределенном подходе все маршрутизаторы сети находятся в равных условиях, они находят маршруты и строят собственные таблицы маршрутизации, работая в тесной кооперации друг с другом, постоянно обмениваясь информацией о конфигурации сети. При централизованном подходе в сети существует один выделенный маршрутизатор, собирающий всю информацию о топологии и состоянии сети от других маршрутизаторов. На основании этих данных выделенный маршрутизатор (иногда называемый сервером маршрутов) строит таблицы маршрутизации для остальных маршрутизаторов сети, распространяя их затем по сети, чтобы каждый маршрутизатор получил собственную таблицу и в дальнейшем самостоятельно принимал решение о продвижении каждого пакета

Рассмотрим пример, приведенный на рис. 12.1, для более легкого понимания различий статической и динамической маршрутизации.

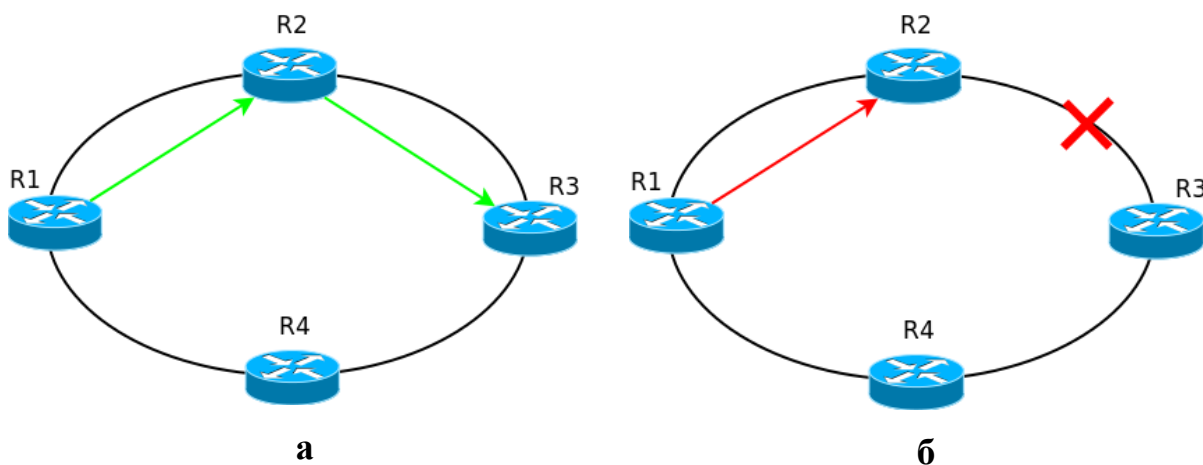


Рисунок 12.1 –а) пример топологии сети, б) разрыв соединения

На рисунке 12.1(а) приведен пример топологии сети. Если в такой сети разорвать соединение между маршрутизаторами R2 и R3, то пакеты с R1 будут уходить по-прежнему на R2, где будут уничтожены, потому что их некуда отправить.

Протоколы динамической маршрутизации в течение нескольких секунд (а то и миллисекунд) узнают о проблемах на сети и перестраивают свои таблицы маршрутизации, и в вышеописанном случае пакеты будут отправляться уже по актуальному маршруту (см. рис 12.2)

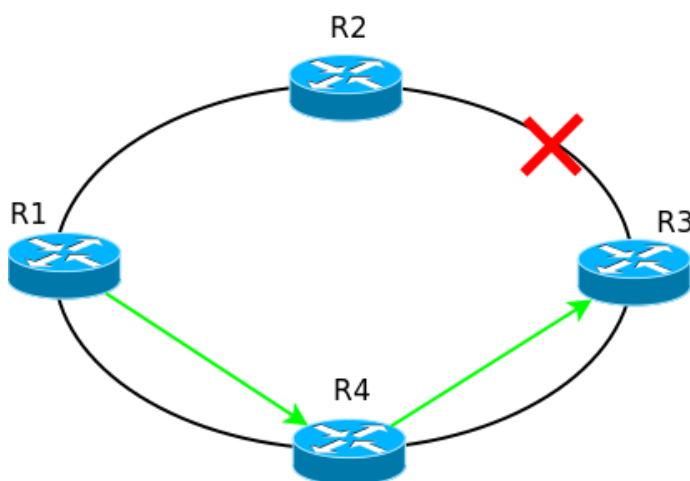


Рисунок 12.2 – Изменение маршрута в сети связи

Итак, протоколы динамической маршрутизации используются для автоматической настройки маршрутов в сетях и обеспечения их надежной работы. Они делятся на две основные категории: внутренние (IGP, Interior Gateway Protocol) и внешние (EGP, Exterior Gateway Protocol) протоколы. IGP используются для обмена информацией о маршрутах внутри одной автономной системы, в то время как EGP используются для обмена информацией между различными автономными системами.

Автономная система (Autonomous System, AS) – группа сетей и маршрутизаторов, находящихся под единым административным управлением и использующих единую политику маршрутизации [3].

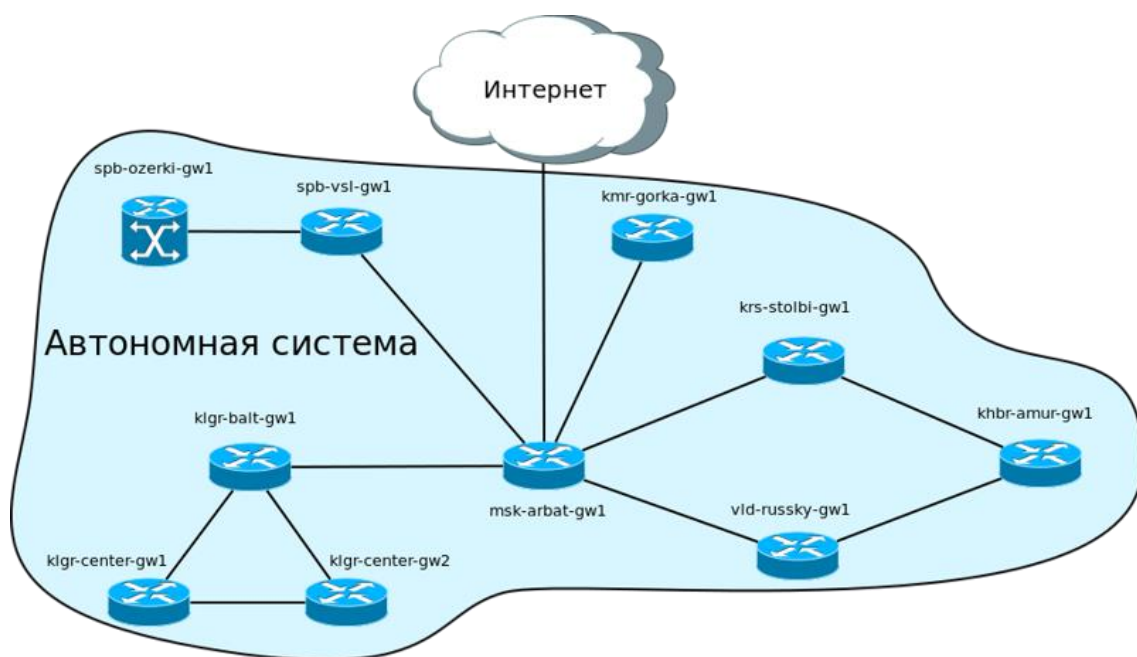


Рисунок 12.3 – Пример автономной системы



Рисунок 12.4 – Классификация протоколов динамической маршрутизации

Коренные различия между этими двумя видами протоколами внутренней маршрутизации состоят в следующем:

- типе информации, которой обмениваются роутеры: таблицы маршрутизации у дистанционно-векторных и таблицы топологии у маршрутизации по состоянию канала;
- процессе выбора лучшего маршрута;
- количестве информации о сети, которое содержит каждый роутер: Distance-Vector знает только своих соседей, Link State имеет представление обо всей сети.

Рассмотрим каждый тип подробнее. Distance-Vector Routing Protocol (дистанционно векторные протоколы), работа которых основана на вычислении метрики – расстояния до сети назначения. Под расстоянием понимают количество узлов (участков сети), которые необходимо пройти пакету до сети назначения [3, 20].

Эта маршрутизация базируется на алгоритме Белмана-Форда. Через определённые моменты времени маршрутизатор передаёт соседним маршрутизаторам всю свою таблицу маршрутизации. Такие простые протоколы как RIP и IGRP просто распространяют информацию о таблицах маршрутов через все интерфейсы маршрутизатора в широковещательном режиме без уточнения точного адреса конкретного соседнего маршрутизатора.

Принцип работы дистанционно-векторного алгоритма следующий:

- каждый маршрутизатор поддерживает таблицу маршрутизации, в которой содержится информация о доступных маршрутах и их метриках (например, количество прыжков до целевого узла);
- регулярно каждый маршрутизатор отправляет сообщения (distance vectors) своим соседям, содержащие информацию о его известных маршрутах и их метриках;
- при получении сообщения от соседнего маршрутизатора, маршрутизатор обновляет свою таблицу

маршрутизации, сравнивая информацию о маршрутах и выбирая наилучший путь к каждому целевому узлу;

- если обновление таблицы маршрутизации происходит (например, изменение метрики или недоступность маршрута), маршрутизатор отправляет обновленную информацию своим соседям;

- процесс обмена сообщениями и обновления таблицы маршрутизации повторяется до тех пор, пока все маршрутизаторы в сети не сойдутся на оптимальных путях к целевым узлам.

Таким образом, дистанционно-векторный алгоритм позволяет каждому маршрутизатору в сети определять наилучшие пути к целевым узлам на основе обмена информацией с соседними узлами. Однако, из-за своей простоты и некоторых ограничений (например, проблемы сходимости в больших сетях), этот алгоритм может быть менее эффективным в сложных сетевых средах.

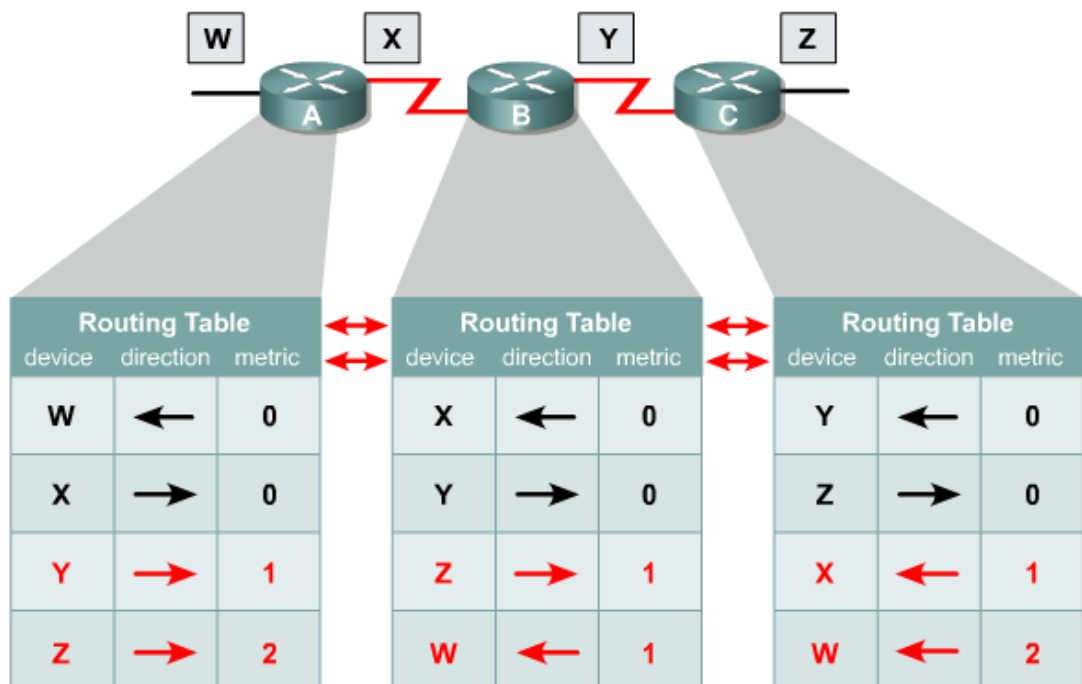


Рисунок 12.5 – Дистанционно-векторная маршрутизация

Примеры протоколов:

- Routing Information Protocol (RIP);
- Enhanced Interior Gateway Routing Protocol (EIGRP).

Link-State Routing Protocol (протоколы маршрутизации по состоянию канала). В протоколах маршрутизации по состоянию канала каждый маршрутизатор обладает информацией о полной топологии сети. Затем каждый маршрутизатор независимо вычисляет наилучший следующий переход от него для каждого возможного пункта назначения в сети, используя локальную информацию о топологии. Совокупность наилучших путей формирует таблицу маршрутизации [20].

Это контрастирует с протоколами маршрутизации по вектору расстояния, которые работают за счет того, что каждый узел делится своей таблицей маршрутизации со своими соседями. В протоколе состояния канала единственная информация, передаваемая между узлами, – это информация, используемая для построения подключений.

Эти протоколы предлагают лучшую масштабируемость и сходимость по сравнению с дистанционно-векторными протоколами. Протокол базируется на алгоритме Дейкстры, который часто называют алгоритмом «кратчайший путь – первым» (shortest path first (SPF)).

Примеры протоколов маршрутизации по состоянию канала:

- Open Shortest Path First (OSPF);
- Intermediate System to Intermediate System (IS-IS).

Алгоритмы поиска путей отличаются для разных протоколов. Рассмотрим наиболее применимые из них.

Алгоритм Беллмана-Форда – это алгоритм для нахождения кратчайших путей во взвешенном графе с отрицательными весами рёбер. Основные этапы алгоритма представлены ниже.

На первом этапе происходит инициализация: устанавливаются расстояния до всех вершин равным бесконечности, кроме начальной вершины, расстояние до которой значение расстояния равно нулю.

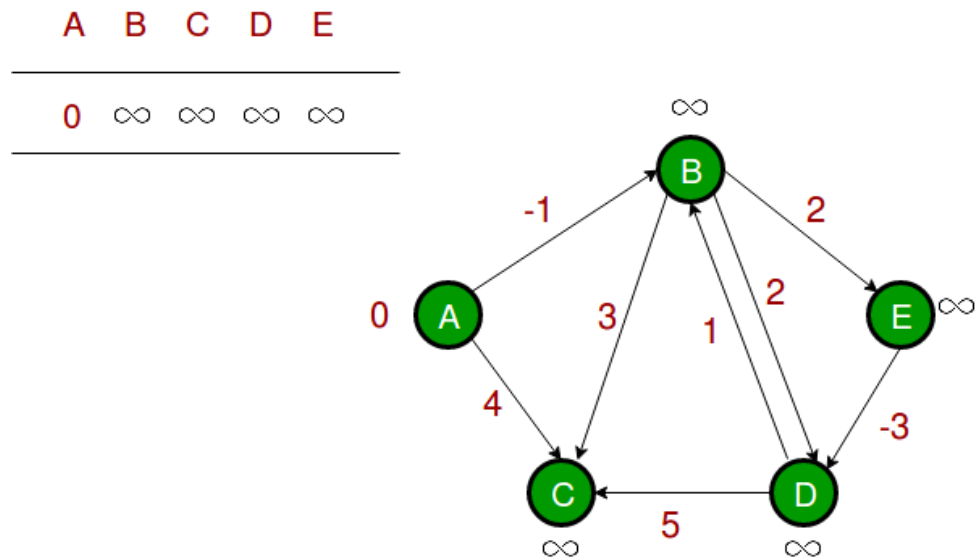


Рисунок 12.6 – Первый этап алгоритма Беллмана-Форда

На втором этапе выполняется релаксация рёбер: повторяется следующий процесс $|V|-1$ раз, где $|V|$ - количество вершин в графе, то есть происходит проход по всем рёбрам графа и обновляется расстояние до каждой вершины, если найден более короткий путь через текущее ребро.

Вернемся к примеру. Пусть ребра обрабатываются в следующем порядке: (B, E), (D, B), (B, D), (A, B), (A, C), (D, C), (B, C), (E, D). Получаем расстояния, когда проход по ребрам был совершен первый раз. Первая строка показывает начальные расстояния, вторая строка показывает расстояния, когда ребра (B, E), (D, B), (B, D) и (A, B) обрабатываются. Третья строка показывает расстояние при обработке (A, C). Четвертая строка показывает, что происходит, когда обрабатываются (D, C), (B, C) и (E, D) (рис. 12.7).

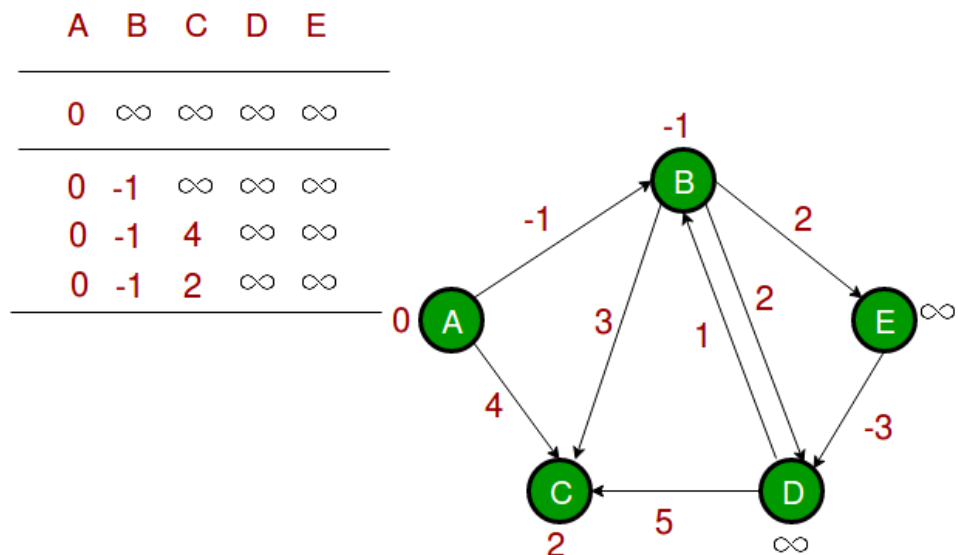


Рисунок 12.7 – Второй этап алгоритма Беллмана-Форда

На третьем этапе выполняется проверка наличия циклов отрицательного веса: после выполнения предыдущего этапа, проверяется наличие циклов отрицательного веса. Если после $|V|-1$ итераций расстояния до вершин продолжают меняться, то это означает наличие цикла отрицательного веса (рис. 12.8).

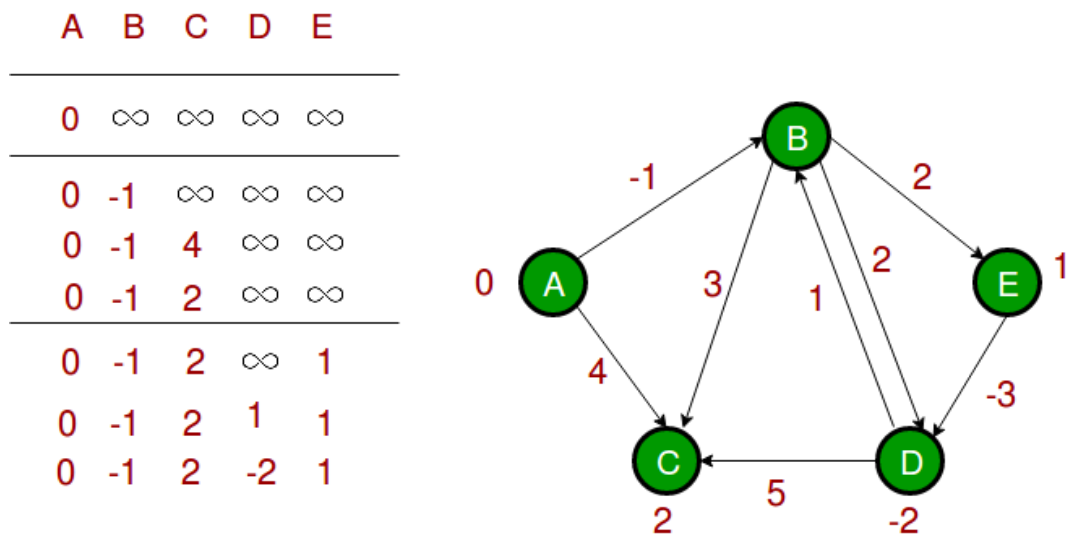


Рисунок 12.8 – Третий этап алгоритма Беллмана-Форда

На четвертом, заключительном этапе выполняется восстановление кратчайшего пути: если цикл отрицательного веса не обнаружен, то можно

восстановить кратчайший путь от начальной вершины до любой другой вершины, используя информацию о предыдущих вершинах.

Алгоритм Беллмана-Форда работает за время $O(|V|*|E|)$, где $|V|$ - количество вершин, $|E|$ - количество рёбер в графе. Он может быть использован для нахождения кратчайших путей в графах с отрицательными весами рёбер, но не допускает циклов отрицательного веса.

Алгоритм Дейкстры – это алгоритм для нахождения кратчайших путей от одной из вершин графа до всех остальных вершин во взвешенном графе без циклов отрицательного веса.

Алгоритм Дейкстры находит кратчайшие пути от начальной вершины до всех остальных в графе с неотрицательными весами рёбер. На первом этапе происходит инициализация: расстояния до всех вершин, кроме начальной, устанавливаются как бесконечные, а расстояние до стартовой вершины принимается равным нулю. Формируется множество непосещённых вершин, включающее все вершины графа.

На каждом шаге алгоритма из непосещённых вершин выбирается вершина с минимальным текущим расстоянием. Для этой вершины выполняется релаксация рёбер – проверка и возможное обновление расстояний до смежных вершин. Если через текущую вершину обнаруживается более короткий путь к соседней вершине, соответствующее расстояние уменьшается. После обработки всех соседей текущая вершина исключается из множества непосещённых.

Процесс повторяется до тех пор, пока все вершины не будут посещены. В результате работы алгоритма формируются кратчайшие расстояния от начальной вершины до каждой достижимой вершины графа, а также становится возможным восстановление самих путей путём отслеживания вершин-предшественников.

Алгоритм гарантированно находит оптимальные решения для графов без рёбер с отрицательным весом, демонстрируя при этом полиномиальную

сложность, что делает его эффективным инструментом решения задач маршрутизации и поиска оптимальных путей в различных приложениях.

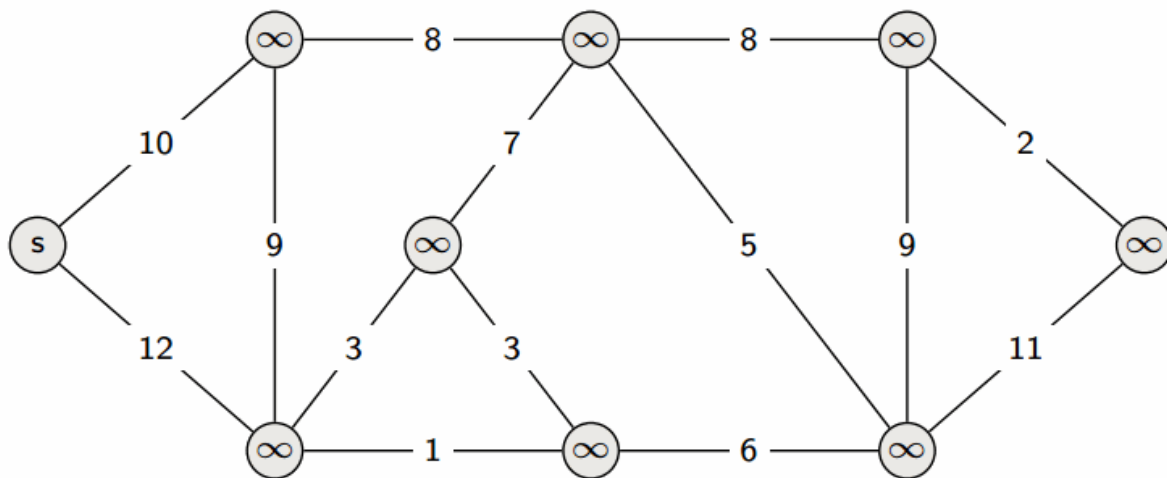


Рисунок 12.9 – Пример поиска маршрута на графе на основе алгоритма Дейкстры

Алгоритм Дейкстры работает за время $O(|V|^2)$ для реализации с использованием матрицы смежности и за $O((|V|+|E|)\log|V|)$ для реализации с использованием кучи (при использовании кучи, где $|V|$ - количество вершин, $|E|$ - количество рёбер в графе).

Протокол маршрутной информации RIP (Routing Information Protocol) является протоколом маршрутизации дистанционно-векторного типа и чаще всего используется в небольших сетях. Для IP-сетей имеются две версии RIP – RIPv1 (не поддерживает бесклассовую адресацию) и RIPv2 (поддерживает бесклассовую адресацию). Так как способ построения таблиц маршрутизации в обеих версиях протокола принципиально не отличается, мы ограничимся описанием работы версии 1.

Для пояснения принципов работы протокола RIP, рассмотрим следующую схему на рис. 12.10.

Контрольные вопросы к материалу П/З № 12:

1. Чем статическая маршрутизация отличается от динамической?
2. В чем заключается классификация протоколов динамической маршрутизации?
3. Как работает протокол маршрутизации RIP (Routing Information Protocol)?
4. Какие критерии выбора оптимального маршрута используются в алгоритмах поиска путей?
5. Как происходит процесс обмена информацией о маршрутах между маршрутизаторами при использовании протокола OSPF?