Санкт-Петербургский политехнический университет Петра Великого

Физико-механический институт

**Высшая школа прикладной математики и вычислительной физики**

Лабораторная работа   
по дисциплине «Компьютерные сети» на тему

**«Протокол маршрутизации Open Shortest Path First»**

|  |  |  |
| --- | --- | --- |
| Выполнил |  |  |
| студент гр. 5040102/10201 | Кудрявцева В.В. | /\_\_\_\_\_\_\_\_\_\_\_/ |
| Руководитель |  |  |
| доцент, к.ф.-м.н. | Баженов А.Н. | /\_\_\_\_\_\_\_\_\_\_\_/ |

Санкт-Петербург

2022

Оглавление

[**Постановка задачи** 3](#_Toc119008226)

[**Теория** 3](#_Toc119008227)

[**Реализация** 3](#_Toc119008228)

[**Результаты** 4](#_Toc119008229)

[**Выводы** 5](#_Toc119008230)

[**Приложение** 5](#_Toc119008231)

# **Постановка задачи**

Реализовать протокол маршрутизации OSPF. Проверить работоспособность протокола для следующих видов топологии: линейная, кольцевая, звёздная. Проверить возможность перестройки таблиц достижимости в случае стохастического разрыва связи

# **Теория**

Протокол маршрутизации OSPF (Open Shortest Path First – алгоритм предпочтительного выбора кратчайшего маршрута) предназначен для работы в сетях множественного доступа, т.е. сетях, у которых может быть несколько маршрутизаторов, способных общаться друг с другом. Основой работы данного протокола является представление множества сетей, маршрутизаторов и каналов в виде направленного графа. Такое представление позволяет учитывать различные условия и ограничения при выборе кратчайшего пути между любыми двумя маршрутизаторами, а также делить большие системы на области, каждая из которых может обладать своей собственной топологией, условиями выбора маршрутов и другими особенностями.

Принцип работы заключается в следующем:

• После включения маршрутизаторов протокол ищет непосредственно подключённых соседей и устанавливает с ними «дружеские» отношения.

• Затем они обмениваются друг с другом информацией о подключённых и доступных им сетях. То есть они строят карту сети (топологию сети). Данная карта одинакова на всех маршрутизаторах.

• На основе полученной информации запускается алгоритм SPF (Shortest Path First, «выбор наилучшего пути»), который рассчитывает оптимальный маршрут к каждой сети. Данный процесс похож на построение дерева, корнем которого является сам маршрутизатор, а ветвями — пути к доступным сетям. Данный процесс, то есть конвергенция, происходит очень быстро.

# **Реализация**

Система реализована на языке программирования Python. Выделенный маршрутизатор (designated router, DR) — управляет процессом рассылки LSA (link-state advertisement, объявление о состоянии канала) в сети. Каждый маршрутизатор сети устанавливает отношения смежности с DR. Информация об изменениях в сети отправляется маршрутизатором, обнаружившим это изменение, на выделенный маршрутизатор, а тот, в свою очередь, отвечает за то, чтобы эта информация была отправлена остальным маршрутизаторам сегмента множественного доступа.

# **Результаты**

Рассмотрим пример работы программы для линейной топологии с 3 узлами. Здеь и далее: узлы – указаны их номера, связи – список номеров соседних узлов на позиции текущего узла.

Узлы [0, 1, 2]

Связи [[1], [0, 2], [1]]

К сети подключены все 3 узла. Кратчайшие пути:

0: [[0], [0, 1], [0, 1, 2]]

1: [[1, 0], [1], [1, 2]]

2: [[2, 1, 0], [2, 1], [2]]

От сети отключен 2-ой узел. Новые кратчайшие пути:

0: [[0], [0, 1], []]

1: [[1, 0], [1], []]

2: [[], [], [2]]

Теперь рассмотрим пример работы программы для кольцевой топологии с 3 узлами.

Узлы [0, 1, 2]

Связи [[2, 1], [0, 2], [1, 0]]

К сети подключены все 3 узла. Кратчайшие пути:

0: [[0], [0, 1], [0, 2]]

1: [[1, 0], [1], [1, 2]]

2: [[2, 0], [2, 1], [2]]

От сети отключен 1-ый узел. Новые кратчайшие пути:

0: [[0], [], [0, 2]]

1: [[], [1], []]

2: [[2, 0], [], [2]]

Наконец, рассмотрим пример работы программы для звездной топологии с 4 узлами. Центр в узле с индексом 1.

Узлы [0, 1, 2, 3]

Связи [[1], [0, 2, 3], [1], [1]]

От сети отключен 3-ий узел. Новые кратчайшие пути:

0: [[0], [0, 1], [0, 1, 2], []]

1: [[1, 0], [1], [1, 2], []]

2: [[2, 1, 0], [2, 1], [2], []]

3: [[], [], [], [3]]

# **Выводы**

Был реализован и протестирован на различных топологиях протокол OSPF. Данная программа была проверена на трех топологиях, из чего был сделан вывод о ее корректной работе на топологиях: линейная, кольцо, звезда. На основе тестов можно утверждать о работоспособности системы на различных топологиях.

# **Приложение**

Ссылка на GitHub с реализацией:

<https://github.com/VasilisaK/Networks/tree/main/2>