

Отчет по лабораторной работе №2.
Реализация протокола динамической маршрутизации Open
Shortest Path First

Мануилов Георгий, студент гр. 3640102/80201

1 Постановка задачи

Требуется разработать систему из неограниченного количества взаимодействующих друг с другом маршрутизаторов, которые организуются в сеть и обеспечивают передачу сообщений от каждого маршрутизатора к каждому по кратчайшему пути.

2 Технические подробности реализации

Программа-маршрутизатор реализована на языке программирования C++. Программа может функционировать в двух режимах: в режиме обычного маршрутизатора и в режиме выделенного маршрутизатора (Designated Router). Сеть задается конфигурацией множества обычных маршрутизаторов с явным указанием смежных маршрутизаторов и конфигурацией одного выделенного маршрутизатора. Для упрощения реализации сеть не содержит резервного выделенного маршрутизатора. Взаимодействие между процессами осуществляется с помощью очередей сообщений посредством библиотеки Boost.Interprocess.

Программа разделена на следующие сущности:

- **OneWayTransducer** - примитивный приемник/передатчик. Не контролирует доставку, работает только в одну сторону, утилизирует одну очередь сообщений.
- **Topology** - класс, описывающий топологию сети. По сути представляет из себя обертку над графом с набором высокоуровневых методов.
- **DesignatedRouter** - выделенный маршрутизатор. Хранит внутри себя экземпляр топологии сети. Обрабатывает сообщения маршрутизаторов об изменении топологии сети и транслирует их остальным маршрутизаторам.
- **CommonRouter** - обычный маршрутизатор. Хранит внутри себя экземпляр топологии сети. После запуска начинает непрерывный обмен сообщениями с соседями. При появлении нового соседа или потере старого посылает сообщение об изменении топологии выделенному маршрутизатору. При получении сообщения от выделенного маршрутизатора вносит изменения в свой экземпляр топологии сети и перестраивает кратчайшие маршруты от себя до каждого узла в сети с помощью алгоритма Дейкстры.

Программа-маршрутизатор получает на вход путь к конфигурационному файлу, который содержит в себе следующие поля:

- **id** - уникальный идентификатор маршрутизатора
- **neighbours** - список смежных маршрутизаторов и стоимость каналов связи с ними

Исходный код решения, инструкции по сборке и запуску, а также примеры представлены в git репозитории по ссылке https://github.com/dev0x13/networks_labs/tree/master/lab2.

3 Примеры работы программы

Примеры конфигурации сети можно найти в папке репозитория https://github.com/dev0x13/networks_labs/tree/master/lab2/presets. Процессы запускаются асинхронно. Процесс выделенного маршрутизатора после завершения (завершение происходит, если в течение 5 секунд от маршрутизаторов не поступило ни одного сообщения об изменении топологии) сохраняет текущую топологию сети в файл DOT, который затем можно визуализировать с помощью утилиты graphviz (визуализации для примеров приведены на рис. 1, 2, 3).

Вывод примера "cross":

```
$ ./router cfg/cross/dr.cfg
[DesignatedRouter] Received topology update from DR_r1_receive
[DesignatedRouter] Received topology update from DR_r2_receive
[DesignatedRouter] Received topology update from DR_r3_receive
[DesignatedRouter] Received topology update from DR_r3_receive
[DesignatedRouter] No topology updates during 5000 ms, terminating

$ ./router cfg/cross/r1.cfg
[CommonRouter r1] New neighbour discovered: 'r3'
[CommonRouter r1] Received topology update from DR
[CommonRouter r1] Shortest paths found
[CommonRouter r1] Received topology update from DR
[CommonRouter r1] Shortest paths found
[CommonRouter r1] Received topology update from DR
[CommonRouter r1] Shortest paths found
[CommonRouter r1] Neighbour 'r3' is dead
[CommonRouter r1] No topology updates during 2000 ms, terminating

$ ./router cfg/cross/r2.cfg
[CommonRouter r2] New neighbour discovered: 'r3'
[CommonRouter r2] Received topology update from DR
[CommonRouter r2] Shortest paths found
[CommonRouter r2] Received topology update from DR
[CommonRouter r2] Shortest paths found
[CommonRouter r2] Received topology update from DR
[CommonRouter r2] Shortest paths found
[CommonRouter r2] Neighbour 'r3' is dead
[CommonRouter r2] No topology updates during 2000 ms, terminating

$ ./router cfg/cross/r3.cfg
[CommonRouter r3] Received topology update from DR
[CommonRouter r3] Shortest paths found
[CommonRouter r3] Received topology update from DR
[CommonRouter r3] Shortest paths found
[CommonRouter r3] Neighbour 'r1' is dead
[CommonRouter r3] Neighbour 'r2' is dead
[CommonRouter r3] New neighbour discovered: 'r1'
[CommonRouter r3] New neighbour discovered: 'r2'
[CommonRouter r3] New neighbour discovered: 'r4'
[CommonRouter r3] New neighbour discovered: 'r5'
[CommonRouter r3] No topology updates during 2000 ms, terminating

$ ./router cfg/cross/r4.cfg
[CommonRouter r4] Received topology update from DR
[CommonRouter r4] Shortest paths found
[CommonRouter r4] Received topology update from DR
[CommonRouter r4] Shortest paths found
[CommonRouter r4] Received topology update from DR
[CommonRouter r4] Shortest paths found
```

```

[CommonRouter r4] Neighbour 'r3' is dead
[CommonRouter r4] New neighbour discovered: 'r3'
[CommonRouter r4] Received topology update from DR
[CommonRouter r4] Shortest paths found
[CommonRouter r4] Neighbour 'r3' is dead
[CommonRouter r4] No topology updates during 2000 ms, terminating

```

```

$ ./router cfg/cross/r5.cfg
[CommonRouter r5] Received topology update from DR
[CommonRouter r5] Shortest paths found
[CommonRouter r5] Received topology update from DR
[CommonRouter r5] Shortest paths found
[CommonRouter r5] Received topology update from DR
[CommonRouter r5] Shortest paths found
[CommonRouter r5] New neighbour discovered: 'r3'
[CommonRouter r5] Neighbour 'r3' is dead
[CommonRouter r5] No topology updates during 2000 ms, terminating

```

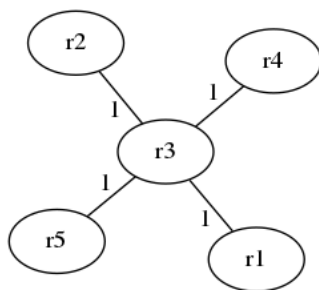


Рис. 1: Пример "cross"

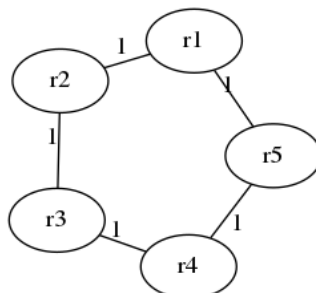


Рис. 2: Пример "circle"

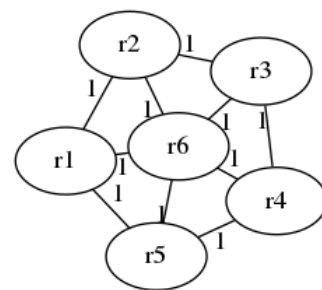


Рис. 3: Пример "full"