

Отчет по лабораторной работе №3.  
Реализация системы адаптивного управления

Мануилов Георгий, студент гр. 3640102/80201

# 1 Постановка задачи

Требуется разработать систему адаптивного управления в контексте компьютерных сетей. В работе был реализован вариант легенды о зеркалах Архимеда: дано Солнце, которое может менять свое положение; некоторое количество зеркал, которые могут поворачиваться, и точка, в которой требуется сфокусировать солнечные лучи, отраженные зеркалами.

## 2 Технические подробности реализации

Программа реализована на языке программирования C++. Каждый из агентов является отдельным процессом, взаимодействие между которыми осуществляется с помощью очередей сообщений посредством библиотеки Boost.Interprocess.

Программа реализует систему из взаимодействующих агентов четырех типов:

- **ControlNode** - агент, организующий работу агентов-исполнителей.
- **WorkerNode** - агент-исполнитель, непосредственно занимающийся фокусировкой своего зеркала.
- **SunNode** - агент, представляющий Солнце.
- **FocusNode** - агент, представляющий точку фокуса.

Для упрощения программы были приняты следующие допущения:

- Каналы между агентами являются надежными, то есть сообщение между ними не требует использования протокола автоматического запроса повторной передачи.
- Сеть стабильна, то есть ее топология не может поменяться с течением времени.

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/lab3](https://github.com/dev0x13/networks_labs/tree/master/lab3).

## 3 Высокоуровневый алгоритм функционирования системы

1. **ControlNode** устанавливает топологию сети так же, как это происходило в лабораторной работе №2 за исключением поиска кратчайшего пути.
2. **ControlNode** валидирует топологию сети: требуется, чтобы это была линия.
3. **ControlNode** находит крайний узел **WorkerNode**, имеющий только одного соседа, и посылает ему сообщение *INVOKE*, запуская тем самым непрерывный процесс адаптивной фокусировки.
4. **WorkerNode**, получивший сообщение, получает координаты Солнца и передает информацию об отраженном луче **FocusNode**.
5. **FocusNode** сверяет информацию со своими координатами и дает обратную связь в виде значения интенсивности света в фокусе: если отраженный луч попал в фокус, значение будет увеличено.
6. **WorkerNode** получает значение интенсивности в фокусе и, в случае если значение не увеличилось с предыдущей итерации, поворачивает свое зеркало на небольшой угол и отправляет **FocusNode** новую информацию об отраженном луче. Процедура повторяется до достижения фокусировки.
7. После фокусировки **WorkerNode** посылает сообщение *INVOKE* своему соседу (либо отличного от того, от которого пришло предыдущее сообщение *INVOKE*, либо тому, от которого пришло сообщение в случае если он является последним в цепи). Таким образом, процесс фокусировки непрерывен и зациклен. Это необходимо, так как координаты **SunNode** постоянно меняются вне зависимости от действий других агентов.

Типы сообщений, используемых агентами для взаимодействия по приведенным выше протоколам и их описания представлены в таблице 1.

Отправитель	Получатель	Тип сообщения	Описание
WorkerNode	SunNode	PING	Запрос на получение координат Солнца
WorkerNode	ControlNode	TOPOLOGY_OPERATION	Сообщение об изменении топологии сети
WorkerNode	FocusNode	VECTOR_AND_POINT	Координаты WorkerNode + отраженный вектор
WorkerNode	FocusNode	PING	Запрос на получение интенсивности в фокусе
WorkerNode	WorkerNode	PING	Сообщение ping
WorkerNode	WorkerNode	PONG	Ответ на сообщение ping
WorkerNode	WorkerNode	INVOKE	Сообщение о передаче потока управления
ControlNode	WorkerNode	TOPOLOGY_OPERATION	Сообщение об изменении топологии сети
ControlNode	WorkerNode	INVOKE	Сообщение о передаче потока управления
SunNode	WorkerNode	VECTOR	Текущие координаты Солнца
FocusNode	WorkerNode	INTENSITY	Текущая интенсивность в фокусе

Таблица 1: Типы сообщений, используемые агентами для взаимодействия

## 4 Примеры работы программы

Пример конфигурация системы можно найти в папке репозитория [https://github.com/dev0x13/networks\\_labs/tree/master/lab3/presets](https://github.com/dev0x13/networks_labs/tree/master/lab3/presets). Процессы запускаются асинхронно. Солнце изменяет свои координаты по оси абсцисс от -4 до 4. Приращение координаты - 1 единица в секунду. Конфигурация примера представлена на рис. 1. Каждый из агентов имеет безусловное время жизни, поэтому после нескольких циклов фокусировки все агенты завершают свою работу.

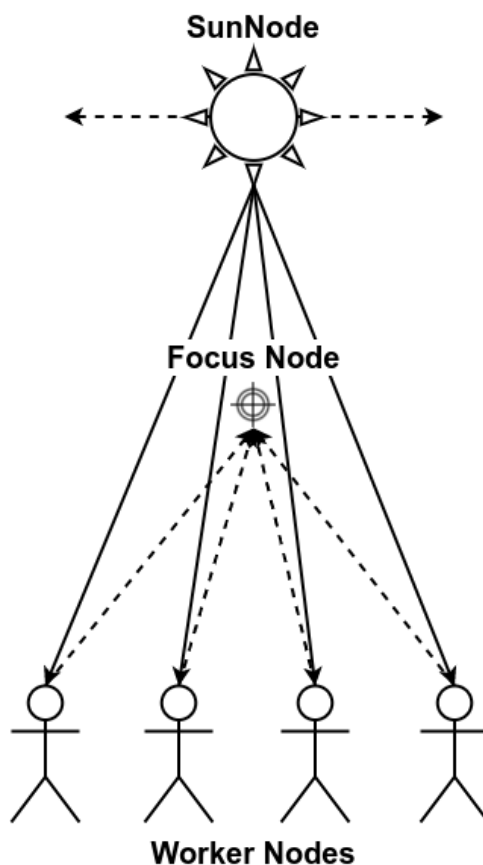


Рис. 1: Конфигурация примера из папки *presets*/

Вывод примера:

```
george@george:~/networks_labs/lab3/presets$ ./run.sh
[WorkerNode r1] New neighbour discovered: 'r2'
[ControlNode] Received topology update from DR_r1_backward
[WorkerNode r2] New neighbour discovered: 'r1'
[WorkerNode r3] Received topology update from control node
[WorkerNode r3] New neighbour discovered: 'r2'
[WorkerNode r2] New neighbour discovered: 'r3'
[ControlNode] Received topology update from DR_r3_backward
[WorkerNode r1] Received topology update from control node
[WorkerNode r4] Received topology update from control node
[WorkerNode r4] Received topology update from control node
[WorkerNode r3] New neighbour discovered: 'r4'
[ControlNode] Received topology update from DR_r3_backward
[WorkerNode r4] New neighbour discovered: 'r3'
[WorkerNode r1] Received topology update from control node
[WorkerNode r2] Received topology update from control node
[ControlNode] Topology researched, determining invocation order
[WorkerNode r4] Invoked by control node
[ControlNode] Terminated
[WorkerNode r4] Focused for Sun coord (-0.7, 5) after 40 rotations
[WorkerNode r3] Invoked by neighbour 'r4'
[WorkerNode r3] Focused for Sun coord (-0.8, 5) after 7 rotations
[WorkerNode r2] Invoked by neighbour 'r3'
[WorkerNode r2] Focused for Sun coord (-0.8, 5) after 1 rotations
[WorkerNode r1] Invoked by neighbour 'r2'
[WorkerNode r1] Focused for Sun coord (-1.2, 5) after 14 rotations
[WorkerNode r2] Invoked by neighbour 'r1'
[WorkerNode r2] Focused for Sun coord (-1.3, 5) after 2 rotations
[WorkerNode r3] Invoked by neighbour 'r2'
[WorkerNode r3] Focused for Sun coord (-1.3, 5) after 21 rotations
[WorkerNode r4] Invoked by neighbour 'r3'
[WorkerNode r4] Focused for Sun coord (-1.4, 5) after 35 rotations
[WorkerNode r3] Invoked by neighbour 'r4'
[WorkerNode r3] Focused for Sun coord (-1.4, 5) after 17 rotations
[WorkerNode r2] Invoked by neighbour 'r3'
[WorkerNode r2] Focused for Sun coord (-1.5, 5) after 16 rotations
[WorkerNode r1] Invoked by neighbour 'r2'
[WorkerNode r1] Focused for Sun coord (-1.6, 5) after 7 rotations
[WorkerNode r2] Invoked by neighbour 'r1'
[WorkerNode r2] Focused for Sun coord (-1.6, 5) after 16 rotations
[WorkerNode r3] Invoked by neighbour 'r2'
[WorkerNode r3] Focused for Sun coord (-1.7, 5) after 13 rotations
[WorkerNode r4] Invoked by neighbour 'r3'
[WorkerNode r4] Focused for Sun coord (-1.9, 5) after 15 rotations
[WorkerNode r3] Invoked by neighbour 'r4'
[WorkerNode r3] Focused for Sun coord (-1.9, 5) after 17 rotations
[WorkerNode r2] Invoked by neighbour 'r3'
[WorkerNode r2] Focused for Sun coord (-2.1, 5) after 17 rotations
[WorkerNode r1] Invoked by neighbour 'r2'
[WorkerNode r1] Focused for Sun coord (-2.2, 5) after 16 rotations
[SunNode] Terminated
[FocusNode] Terminated
[WorkerNode r1] Terminated
[WorkerNode r2] Terminated
[WorkerNode r3] Terminated
[WorkerNode r4] Terminated
```