

# О самоорганизующейся системе светофоров, обеспечивающих бесперебойное движение транспорта

Муравьев А.К.<sup>1</sup>

В работе рассматривается кооперативная игра о доставке транспорта на прямоугольной решетке. Вводятся понятия затора и столкновения автомобилей на ней, а так же автоматов как агентов в светофорах решетки, регулирующих движение.

Показано существование структуры автоматов с локальной областью видимости, приводящей к самоорганизации системы светофоров, обеспечивающих сколь угодно длительную корректную доставку автомобилей без заторов и столкновений для любого количества выходов решетки.

*Ключевые слова:* кооперативная игра, доставка, организация движения транспорта, агент, однородная структура, автомат.

## 1. Введение

Рассматривается следующая кооперативная игра. Дана прямоугольная сетка с  $m$  столбцами и  $n$  строками (Рис. 1). Назовем ее *решеткой* или *полем*.

Ребра решетки — *дороги*, по которым будут двигаться автомобили. Каждое ребро вмещает лишь один автомобиль.

На левой границе поля —  $n + 1$  *вход*, куда будут заезжать автомобили. Присвоим им номера от 0 до  $n$  с порядком нумерации снизу вверх.

Игра происходит дискретно. В каждый такт времени случайно выбирается  $u$  входов без повторений (из равномерного распределения на всех

---

<sup>1</sup>Муравьев Артем Константинович — эксперт Департамента Анализа Данных и Моделирования, "Банк ВТБ"(ПАО), e-mail: muravev.a.k@yandex.ru

Muravev Artem Konstantinovich — expert, Department of Data Analysis and Modelling, VTB Bank

входах), и в них заезжает по 1 автомобилю. За один такт времени автомобиль перемещается с ребра, на котором находится, на какое-то другое, имеющее с первым общую вершину.

Перед началом игры выбирается  $u < (n + 1)$  узлов на правой границе поля, и объявляются *выходами*. Автомобиль, приезжающий в выходной узел, исчезает, и считается доставленным.

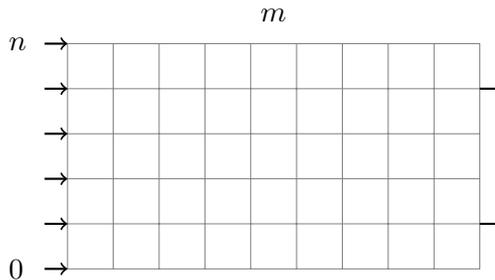


Рис. 1

В каждом узле поля установим *светофор*, регулирующий движение автомобилей через этот узел. По каждой из горизонтальных дорог движение организовано исключительно от входов к выходам (слева направо), а по вертикальным — либо вверх, либо вниз. Соответственно, светофор в каждый такт времени выбирает одно из входящих в него направлений (сверху, снизу или слева), переключает свой сигнал в какое-то выходное направление (вверх, вниз или вправо), и пропускает автомобиль туда.

При этом допустим, что светофорам на  $i$ -й снизу горизонтальной полосе на переключение необходимо  $\frac{i}{n+2}$  времени. Такое допущение позволит разрешать конфликты, связанные с порядком переключения светофоров, и тем самым получить лавинообразную смену логики пропуска, в которой нижние светофоры будут «ведущими», а те, что сверху от них — «ведомыми». При этом каждый светофор все же будет переключаться за один такт игры. Подобный подход использовался в [1, с. 152] для разрешения спорных ситуаций при параллельном обходе информационного графа несколькими вычислителями в модели параллельных алгоритмов поиска.

Если на выбранном выходном направлении на ближайшем к светофору ребре уже стоит автомобиль, и его не пропускает следующий светофор, то и текущий светофор не может пропустить автомобиль в выбранном направлении, и автомобиль стоит на нем в ожидании пропуска.

*Столкновением* назовем ситуацию, когда два светофора, находящиеся на одном и том же ребре решетки, выпустили автомобили на это ребро с противоположных сторон (Рис. 2).

*Задержкой* автомобиля на светофоре назовем ситуацию, когда на светофор с нескольких сторон одновременно приехали автомобили, и он их направляет в одну и ту же сторону, в следствие чего один из автомобилей проезжает на текущем такте времени, заставляя другой в течение этого такта стоять на месте и ждать (Рис. 3). При этом будем считать, что задержек не возникает, если автомобили одновременно приезжают на светофор, и светофор их направляет по разным выходным направлениям (Рис. 4, 5). Аналогом этому в реальной жизни будет организация движения на нерегулируемых перекрестках, где при проезде одновременно двух автомобилей из разных въездов в разные выезды они одновременно выезжают на перекресток, одна из них пропускает другую, и обе завершают маневр практически одновременно.

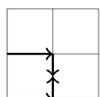


Рис. 2



Рис. 3



Рис. 4



Рис. 5

*Затором* назовем ситуацию, когда входной светофор, на который на данном такте заезжает автомобиль, не может пропустить его в выбранное им направлении из-за задержки автомобиля на нем.

*Цель игры* — организовать доставку автомобилей на выходы таким образом, чтобы при любой длительности игры не создавалось заторов и столкновений. При этом игроками считаем светофоры, регулирующие проезд автомобилей через соответствующие вершины, а  $m$  — длину поля — считаем параметром игры, который выбираем изначально в зависимости от  $n$ ,  $u$ .

Автор выражает благодарность профессору Э. Э. Гасанову за постановку задачи и помощь в работе.

## 2. Постановка задачи

Будем считать, что в светофорах находятся *автоматы*, имеющие конечное число состояний, каждое из которых определяет какое-либо поведение по пропуску автомобилей через него. Каждый автомат принимает на вход ситуацию по въезжающим на него и выезжающим с него авто-

мобилиям, а так же состояния и аналогичные показатели со светофоров в некоторой его окрестности.

Определим *типы светофоров* на поле:

- (1-4) Не угловые светофоры, находящиеся на соответствующей границе поля: верхней, нижней, левой или правой;
- (5-8) Угловые светофоры (по одному типу на каждый угол);
- (9) Внутренние светофоры - не попавшие в предыдущие типы.

И потребуем, чтобы система автоматов в светофорах была *однородной* внутри каждого из типов.

Требуется определить структуру автоматов (а именно - структуру автомата для каждого из типов) таким образом, чтобы в процессе игры описанная система светофоров самоорганизовалась на обеспечение доставки автомобилей со случайных входов на определенные выходы, не известные до начала игры, без заторов и столкновений.

### 3. Описание решения

#### 3.1. Игра с одним выходом

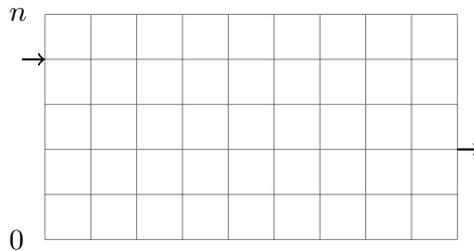


Рис. 6

Рассмотрим вариант игры, когда  $u = 1$  (Рис. 6). То есть, на каждом такте времени в какой-то один случайный вход (выбранный из равномерного распределения на всех входах) заезжает автомобиль, а задача агентов на светофорах - доставить их все на выход, случайно выбранный до начала игры, и не известный изначально, не создавая при этом заторов и не сталкивая автомобили друг с другом.

**Определение 1.** Статичная стратегия игры - это стратегия (структура автоматов в светофорах), при которой в процессе обучения каждый из светофоров переходит в одно единственное финальное состояние, в котором и остается в дальнейшем.

**Определение 2.** Динамичная стратегия игры - это стратегия, не являющаяся статичной.

**Теорема 1.** Для  $m \geq 1$  и любых  $n$  существует статичная стратегия игры всех игроков, приводящая к сколь угодно длительной доставке автомобилей со случайных входов на определенный выход сети без заторов и столкновений.

*Доказательство.* Рассмотрим конфигурацию светофоров, изображенную на Рис. 7. Потребуем на каждом из светофоров на правой границе поля установить приоритет при проезде для автомобилей, подъезжающих по горизонтальным дорогам, нежели тем, что уже едут по правой границе. Полученную конфигурацию будем обозначать как  $K_{n,m}$ .

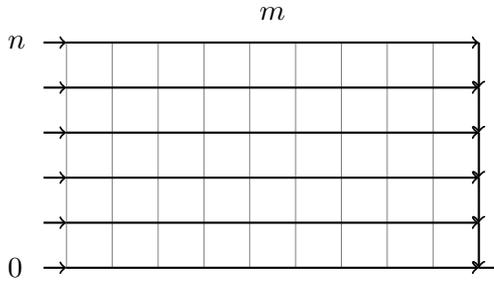


Рис. 7:  $K_{n,m}$

Очевидно, что такая конфигурация обеспечивает доставку на выход в углу решетки без столкновений. Покажем, что она также обеспечивает отсутствие заторов. Для этого последовательно докажем несколько предложений.

**Предложение 1.** Конфигурация  $K_{1,1}$  обеспечивает доставку автомобилей в правый нижний угол на поле  $1 \times 1$  без заторов.

*Доказательство.* Для удобства введем обозначения светофоров ( $S_i$ ) и дорог ( $a_i^j$ ) на поле, как показано на Рис. 8.

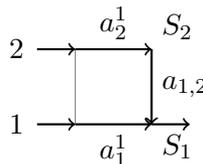


Рис. 8

Учитывая, что на  $S_1$  приоритет при проезде имеют автомобили, подъезжающие со стороны дороги  $a_1^1$ , то на этом светофоре задержка может возникнуть только со стороны дороги  $a_{1,2}$ . Тогда на входе 1 тут заторов быть не может, поэтому для затора при выбранной конфигурации необходима задержка на светофоре  $S_2$  со стороны  $a_2^1$ .

Очевидно, что из-за вышеупомянутого приоритета на светофоре  $S_1$  автомобиль, заехавший на такте  $t$  на вход 1 (на ребро  $a_1^1$ ), обязательно покинет это ребро на такте  $t + 1$ .

Теперь рассмотрим случай, когда на такте  $t$  автомобиль заехал во вход 2 (на ребро  $a_2^1$ ). Получается, что в момент  $t$  в первый вход автомобиль не заезжал, и на ребре  $a_1^1$  на такте  $t$  пусто, потому как даже если туда и заезжал автомобиль ранее, он уже проехал в силу его приоритета на светофоре  $S_1$ . Из-за этого на такте  $t + 1$  автомобиль, возможно бывший на ребре  $a_{1,2}$ , проехал на выход, и в любом случае ребро  $a_{1,2}$  освобождается, и туда проезжает автомобиль с ребра  $a_2^1$ .

Тем самым, получаем, что автомобиль, заехавший на такте  $t$  на вход 2 (на ребро  $a_2^1$ ), точно покинет это ребро на такте  $t + 1$ .

Итого получаем, что для каждого из светофоров  $S_i, i = 1, 2$ , справедливо утверждение, что на  $S_i$  не возникнет задержек со стороны  $a_i^1$ . То есть, не возникнет заторов на входах 1, 2.

□

**Предложение 2.** Конфигурация  $K_{n,1}$  обеспечивает доставку автомобилей в правый нижний угол на поле  $n \times 1$  без заторов.

*Доказательство.* Введем обозначения, аналогичные предыдущим (Рис. 9).

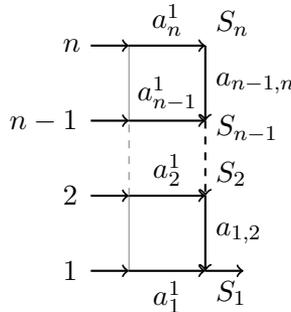


Рис. 9

Будем доказывать предположение индукцией по  $n$ .

- *База:*  $n = 1$  — очевидно;  $n = 2$  — по Предл. 1.
- *Индуктивный переход:* Пусть утверждение выполнено для  $n - 1$ . Докажем его для  $n$ .

Добавление дополнительного входа не влияет на ситуацию на ребрах  $a_i^1, i = 1 \dots n - 1$ , так как у автомобилей на этих дорогах на светофорах  $S_i, i = 1 \dots n - 1$ , приоритет. При этом для  $n - 1$  по предположению индукции на  $S_i$  не может быть задержек со стороны дорог  $a_i^1$ . Поэтому и для  $n$  на светофорах  $S_i, i = 1 \dots n - 1$ , со стороны дорог  $a_i^1$  задержек быть не может, соответственно, не может быть заторов на входах  $1 \dots n - 1$ .

Покажем, что на  $S_n$  также не может быть задержки со стороны дороги  $a_n^1$ .

Пусть на такте  $t$  автомобиль заехал во вход  $n$  (на ребро  $a_n^1$ ). Тогда что в момент  $t$  ни в какой из входов  $1 \dots n - 1$  автомобили не заезжали, и на ребрах  $a_i^1, i = 1 \dots n - 1$ , на такте  $t$  пусто, потому как даже если туда и заезжали автомобили ранее, они уже проехали в силу отсутствия задержек на светофорах  $S_i, i = 1 \dots n - 1$ , со стороны дорог  $a_i^1, i = 1 \dots n - 1$ . Из-за этого на такте  $t + 1$  цепочка автомобилей, возможно бывших на ребрах  $a_{i,i+1}, i = 1 \dots n - 1$ , проезжает вперед к выходу, и в любом случае ребро  $a_{n-1,n}$  освобождается, и туда проезжает автомобиль с ребра  $a_n^1$ .

Тем самым, получаем, что автомобиль, заехавший на такте  $t$  на вход  $n$  (на ребро  $a_n^1$ ), точно покинет это ребро на такте  $t + 1$ . То есть, задержек на светофоре  $S_n$  не возникнет, соответственно, не возникнет и заторов на входе  $n$ .

□

**Предложение 3.** *Конфигурация  $K_{n,m}$  обеспечивает доставку автомобилей в правый нижний угол на поле  $n \times m$  без заторов.*

*Доказательство.* Снова вводим соответствующие обозначения (Рис. 10).

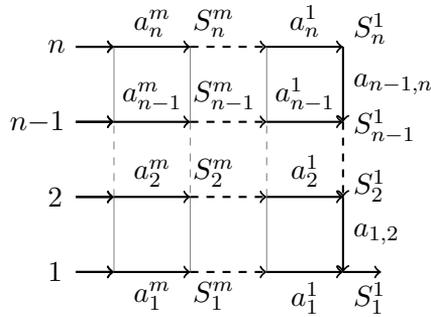


Рис. 10

Докажем предложение индукцией по  $t$ .

- *База:*  $t = 1$  — по Предл. 2.
- *Индуктивный переход:* Пусть выполнено для  $t - 1$ . Докажем для  $t$ .

Для  $t - 1$  по предположению индукции не возникнет заторов на светофорах  $S_i^m, i = 1 \dots n$ , поэтому для  $t$  на тех же самых светофорах не возникнет задержек со стороны дорог  $a_i^m, i = 1 \dots n$ . Поэтому заторов на входах  $1 \dots n$  так же не возникнет.

□

Таким образом, получили, что конфигурация  $K_{n,m}$  обеспечивает требуемую доставку на поле  $n \times t$ . Однако, лишь в правый нижний угол.

Пусть теперь выход на  $l$ -й горизонтальной дороге,  $l > 0$ . В таком случае рассмотрим модернизацию конфигурации  $K_{n,m}$ , изображенную на Рис. 11:

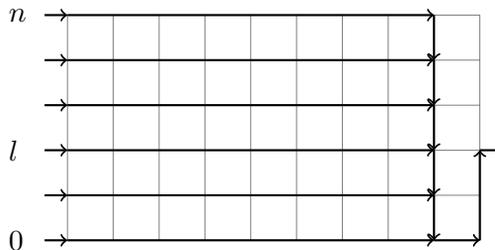


Рис. 11

В таком случае так же выполнены все необходимые условия. Заметим наконец, что здесь  $t \geq 1$ .

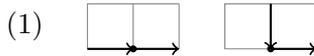
□

Покажем теперь, что можно организовать обучение агентов (светофоров) показанной стратегии в процессе игры с заранее неизвестными параметрами.

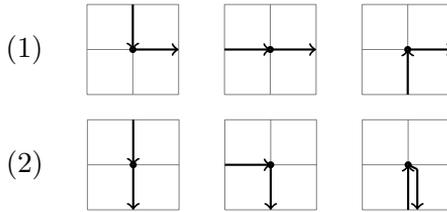
**Теорема 2.** *Существует алгоритм обучения светофоров (то есть конфигурация автоматов) с локальным полем видимости радиуса 1, приводящий к организации требуемой доставки.*

*Доказательство.* Определим состояния автоматов для разных типов светофоров:

1. Светофоры на горизонтальной границе поля (тип *HE*):

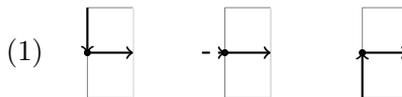


2. Внутренние светофоры (тип *CORE*):



Здесь так же определим, что на каждом из светофоров приоритет при проезде имеет автомобиль, приближающийся слева.

3. Светофоры на вертикальной границе со входами (тип *VEIN*):



4. Светофоры на вертикальной границе с выходами (тип *VEOUT*):



Здесь так же определим, что приоритет при проезде будет иметь автомобиль, приближающийся слева.

Светофоры на углах снизу и сверху будут повторять второй паттерн поведения (1), и его же в зеркальном (относительно горизонтали) виде соответственно.

Опишем теперь структуру автоматов в светофорах.

Для светофоров типов  $HE$ ,  $VEIN$ ,  $VEOUT$  состояние всего одно, изначально его и фиксируем.

Опишем автомат для типа  $CORE$  (Рис. 12). Петли на состояниях будем опускать, подразумевая, что все входы, не описанные в триггерах, переводят состояние в себя же.

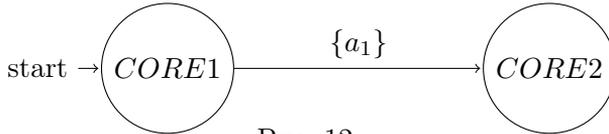
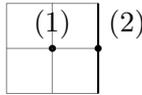


Рис. 12

Опишем указанный на схеме триггер перехода  $a_1$ .



$(1) - CORE1, (2) - VEOUT \Rightarrow$

$a_1 : (1) : CORE1 \rightarrow CORE2$

□

### 3.2. Игра с двумя выходами

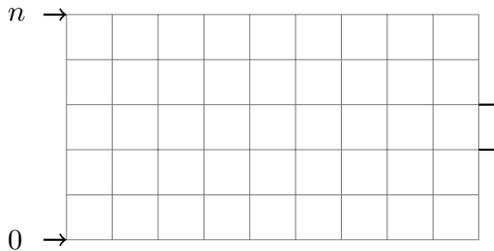


Рис. 13

Теперь рассмотрим вариант игры, когда  $u = 2$  (Рис. 13). То есть, на каждом такте времени в два случайных входа (выбранных без повторений из равномерного распределения на всех входах) заезжает по автомобилю, а задача агентов на светофорах - доставить их все в два выхода, случайно выбранные до начала игры, и не известные изначально, не создавая при этом заторов и не сталкивая автомобили друг с другом.



Так происходит до тех пор, пока верхний автомобиль, двигаясь в это время по все той же «лесенке», не приблизится к нижнему на расстояние в 2, либо 1 ячейку (если  $(j - i)$ -четно, то они сблизятся на расстояние в 2 ячейки, а дальше сблизать нельзя, так как они столкнутся на светофоре между ними; если  $(j - i)$ -нечетно (как на Рис. 13), то они сблизятся на расстояние в 1 ячейку). Как только это произошло - верхний автомобиль начинает двигаться параллельно нижнему (повторяя его траекторию), пока они вместе не достигнут  $n$ -го слоя.

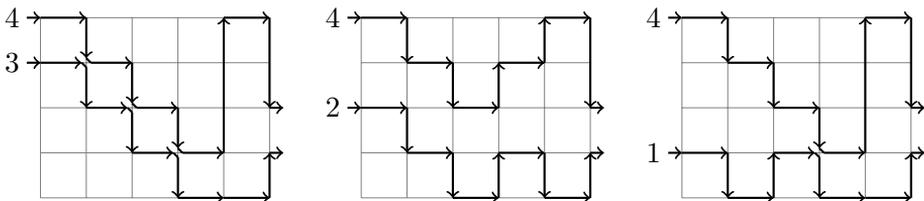
На  $n$ -й слой автомобили выезжают параллельно друг другу по горизонтальным дорогам, расположенным около нижней границы поля, так что:

- Нижний автомобиль выезжает либо по нижней граничной дороге, либо по соседней к ней;
- верхний автомобиль выезжает по дороге, параллельной той, с которой выезжает нижний, и отстоящей от нее вверх на 1 либо 2 ячейки.

Этап 2: Перемещаем автомобили по горизонтальным дорогам прямо до предпоследнего вертикального ряда светофоров, на нем разводим их в противоположные вертикальные направления, доводя до противоположных границ поля, и доставляя через углы решетки по дальней границе до нужных выходов.

Получаем, что при таком раскладе всегда автомобиль, заехавший в верхний вход, едет на верхний выход, а тот, что заехал в нижний вход - отправляется в нижний выход. Тем самым, поток автомобилей на верхние входы равен потоку автомобилей на верхний выход, и аналогично с нижними входами и выходами.

Приведем примеры описанной выше стратегии для  $n = 4, m = 5$ , выходов с индексами 1 и 2, и всевозможными парами входов (Рис. 16):



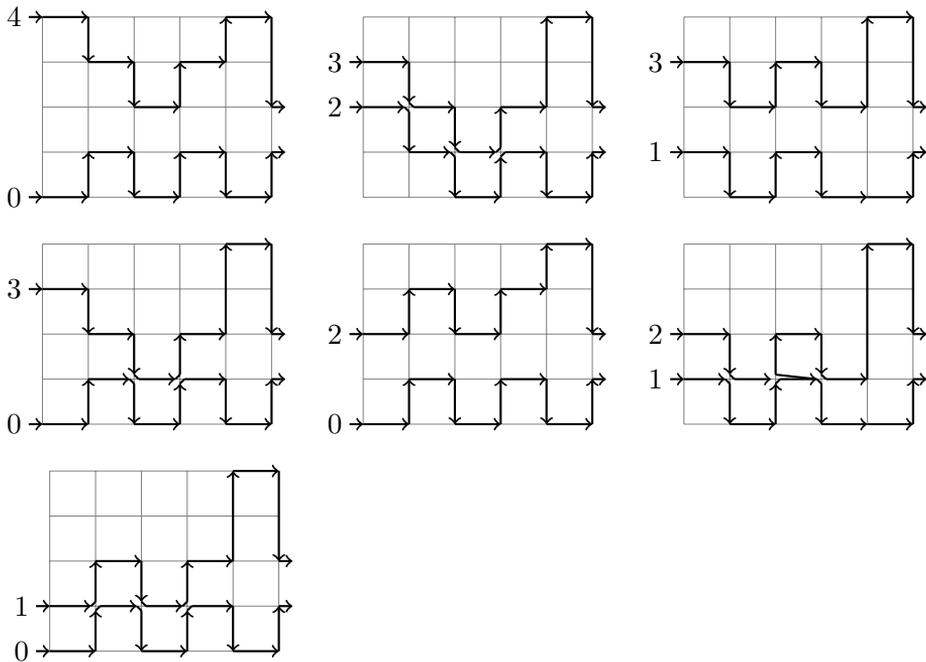


Рис. 16

Покажем теперь, что данная стратегия верна.

Очевидно, что автомобили с разных входов при такой стратегии не встретятся друг с другом. Поэтому достаточно доказать, что не будет образовываться заторов и столкновений из-за автомобилей, захвативших в разные такты времени.

На Этапе 1 - автомобили движутся с постоянной скоростью по слоям, поэтому:

1. автомобили, въехавшие друг за другом с разницей в 1 такт, будут ехать по разным направлениям (одни - вертикально, другие - горизонтально), поэтому заторов и столкновений не возникнет;
2. автомобили, въехавшие с разницей в  $\geq 2$  такта, будут ехать на разных слоях, поэтому встретиться они не могут, следовательно, заторов и столкновений так же не возникнет.

На Этапе 2 - доставка к каждому из выходов превращается в аналог доставки с одним выходом, описанной выше. Там для нее и показана корректность.

Осталось показать ограничения на длину поля  $m$ . Для первого этапа, очевидно, необходима длина  $m_1 = n$  (так как в худшем случае, когда верхний автомобиль едет с последнего входа, для сдвига его на нижнюю около-границную дорогу необходимо ровно  $n$  слоев). Для второго этапа - справедливо ограничение, показанное выше для игры с одним выходом, то есть  $m_2 \geq 1$ . В итоге получаем

$$m = m_1 + m_2 \geq n + 1.$$

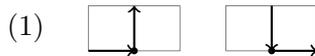
□

Покажем теперь, что можно организовать обучение агентов (светофоров) показанной стратегии в процессе игры с заранее неизвестными параметрами.

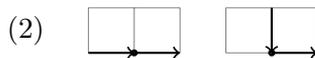
**Теорема 4.** *Существует алгоритм обучения светофоров (то есть конфигурация автоматов) с локальным полем видимости радиуса 2, приводящий к организации требуемой доставки.*

*Доказательство.* Определим состояния автоматов для разных типов светофоров:

1. Светофоры на горизонтальной границе поля (тип *HE*):

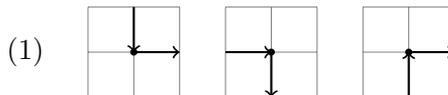


Светофоры границы на 1 этапе.

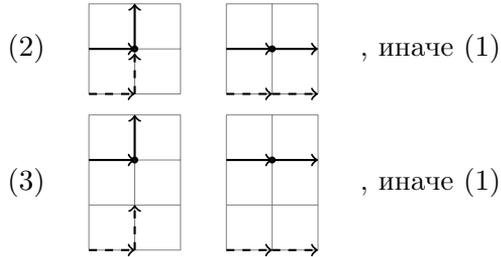


Светофоры границы на этапе 2.

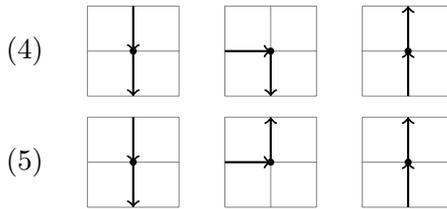
2. Внутренние светофоры (тип *CORE*):



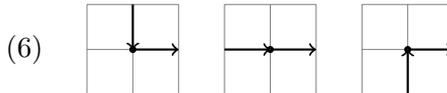
Начальное состояние внутреннего светофора с логикой «сближения».



Здесь (2) / (3) обозначают поведение светофора, который повторяет за светофором ниже него на 1 / 2 ячейки, если на этот светофор автомобиль приехал одновременно и параллельно тому, что приехал на текущий.

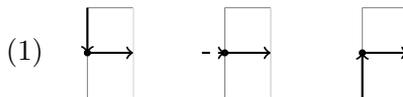


(4) / (5) - это состояния светофоров на предпоследней вертикальной границе.

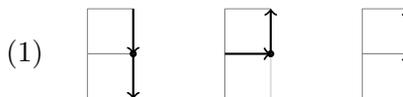


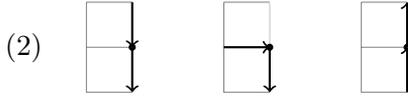
(6) - состояние внутреннего светофора на этапе 2, где автомобили едут параллельно друг другу.

3. Светофоры на вертикальной границе со входами (тип *VEIN*):



4. Светофоры на вертикальной границе с выходами (тип *VEOUT*):





Оба состояния эквивалентны с точки зрения нашей стратегии доставки.

Светофоры на углах снизу и сверху будут повторять второй паттерн поведения (1) или (2) соответственно.

Опишем теперь структуру автоматов в светофорах.

Для светофоров типа *VEIN* состояние всего одно, посему тут нечего обучать, изначально фиксируем его.

Для светофоров типа *VEOUT* расставим изначально случайно либо (1), либо (2), кроме угловых, про которые описано выше.

Опишем автоматы для типов *HE* (Рис. 17) и *CORE* (Рис. 18). Петли на состояниях будем опускать, подразумевая, что все входы, не описанные в триггерах, переводят состояние в себя же.

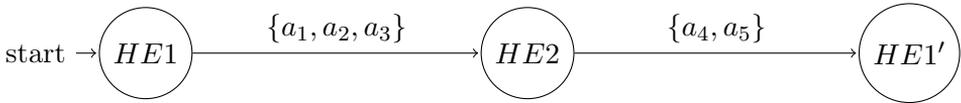


Рис. 17

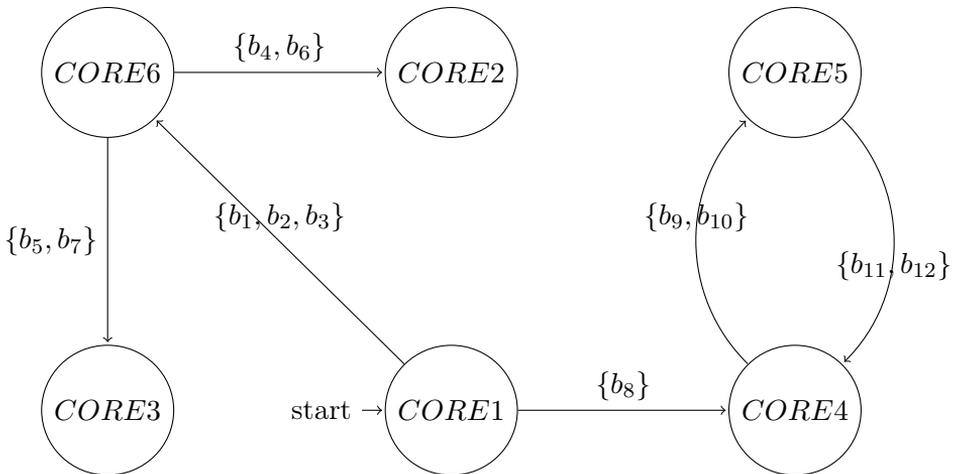
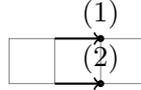


Рис. 18

$HE1'$  здесь - состояние автомата, в котором поведение агента совпадает с поведением  $HE1$ , но состояние отличается от изначального.

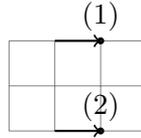
Опишем указанные на схемах триггеры переходов. Толстые линии будут обозначать участки дорог, на которых в рассматриваемый момент времени стоят автомобили.

1.  $\{a_1\}$



$(1) - CORE1, (2) - HE1 \Rightarrow$   
 $a_1 : (2) : HE1 \rightarrow HE2$

2.  $\{a_2\}$



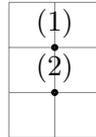
$(1) - CORE1, (2) - HE1 \Rightarrow$   
 $a_2 : (2) : HE1 \rightarrow HE2$

3.  $\{b_1\}$



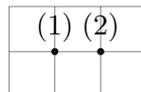
$(1) - CORE1, (2) - HE2 \Rightarrow$   
 $b_1 : (1) : CORE1 \rightarrow CORE6$

4.  $\{b_2\}$



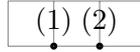
$(1) - CORE1, (2) - CORE6 \Rightarrow$   
 $b_2 : (1) : CORE1 \rightarrow CORE6$

5.  $\{b_3\}$



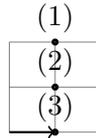
(1) – *CORE6*, (2) – *CORE1*  $\Rightarrow$   
 $b_3 : (2) : \textit{CORE1} \rightarrow \textit{CORE6}$

6.  $\{a_3\}$



(1) – *HE2*, (2) – *HE1*  $\Rightarrow$   
 $a_3 : (2) : \textit{HE1} \rightarrow \textit{HE2}$

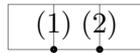
7.  $\{a_4, b_4, b_5\}$



(1) – *CORE6*, (2) – *CORE6*, (3) – *HE2*  $\Rightarrow$   
 $b_4 : (1) : \textit{CORE6} \rightarrow \textit{CORE3}$   
 $b_5 : (2) : \textit{CORE6} \rightarrow \textit{CORE2}$   
 $a_4 : (3) : \textit{HE2} \rightarrow \textit{HE1}'$

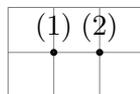
Здесь рассматривается случай, когда на нижний светофор автомобиль приехал, а ни на один из верхних в радиусе 2 не приехало ничего.

8.  $\{a_5\}$



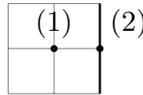
(1) – *HE2*, (2) – *HE1'*  $\Rightarrow$   
 $a_5 : (1) : \textit{HE2} \rightarrow \textit{HE1}'$

9.  $\{b_6, b_7\}$



(1) – *CORE6*, (2) – *CORE2*  $\Rightarrow$   
 $b_6 : (1) : \textit{CORE6} \rightarrow \textit{CORE2}$ ;  
(1) – *CORE6*, (2) – *CORE3*  $\Rightarrow$   
 $b_7 : (1) : \textit{CORE6} \rightarrow \textit{CORE3}$

10.  $\{b_8\}$



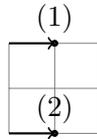
(1) – *CORE1*, (2) – *VEOUT*  $\Rightarrow$   
 $b_8 : (1) : CORE1 \rightarrow CORE4$

11.  $\{b_9, b_{11}\}$



(1) – *CORE4*, на (2) так же заезжает автомобиль  $\Rightarrow$   
 $b_9 : (1) : CORE4 \rightarrow CORE5$ ;  
 (2) – *CORE5*, на (1) так же заезжает автомобиль  $\Rightarrow$   
 $b_{11} : (2) : CORE5 \rightarrow CORE4$

12.  $\{b_{10}, b_{12}\}$



(1) – *CORE4*, на (2) так же заезжает автомобиль  $\Rightarrow$   
 $b_{10} : (1) : CORE4 \rightarrow CORE5$   
 (2) – *CORE5*, на (1) так же заезжает автомобиль  $\Rightarrow$   
 $b_{12} : (2) : CORE5 \rightarrow CORE4$

В итоге, после первого проезда пары, где один из автомобилей идет из самого дальнего (левого верхнего) входа, получим, что у все светофоры типа *HE* на нижней границе до  $n$ -го проезда находятся в состояниях *HE1/HE1'*, а все после - в состоянии *HE2*. Внутренние светофоры до  $n$ -го слоя, удаленные от нижней границы на 1, будут все в состоянии *CORE2*, а те, что удалены на 2 ячейки - в состоянии *CORE3*. Остальные внутренние светофоры останутся в состоянии *CORE1*. В силу определения правил проезда для вышеупомянутых состояний, получим обученный первый этап.

Все внутренние светофоры после  $n$ -го проезда и до предпоследнего будут в состоянии *CORE6*, а те, что на предпоследнем слое, будут адаптироваться под пары автомобилей, приезжающие с первого этапа. Тем самым, и в силу определения правил проезда для состояний, получим обученный второй этап.  $\square$

### 3.3. Игра с произвольным числом выходов

Наконец, рассмотрим вариант игры, когда  $u > 2$ .

**Теорема 5.** Для  $t \geq n - u + 1$  и любых  $n$  и  $u$  существует динамичная стратегия игры всех игроков, приводящая к сколь угодно длительной доставке автомобилей с и случайных входов на  $u$  определенных выходов сети без заторов и столкновений.

*Доказательство.* Обобщим стратегию, разработанную для двух входов, на общий случай. Примеры: Рис. 19 ( $n = 6, t = 10, u = 3$ ), Рис. 20 ( $n = 8, t = 11, u = 4$ ).

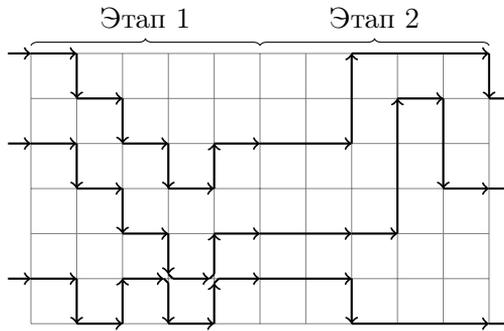


Рис. 19

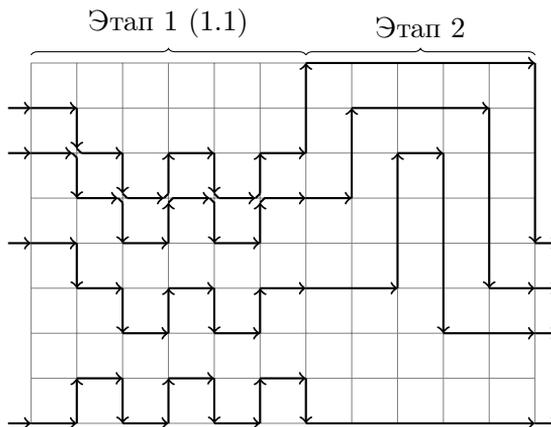


Рис. 20

На первом этапе будем вести автомобиль с нижнего входа так же, как и в прошлый раз, прижав к нижней границе, но не до  $n$ -го слоя, а до  $n - u + 2$ -го.

Второй снизу ведем так же, как вели верхний в случае двух входов, - «прижимаем» к нижнему на 2, либо 1 ячейку, и ведем параллельно.

И так далее, каждый следующий по порядку с низу автомобиль ведем, «прижимая» к предыдущему на 2, либо 1 ячейку.

Для корректного сближения в таком случае нам понадобится  $(n - u + 2)$  слоя. Данную границу устанавливаем как необходимое количество слоев для «прижатия» всех автомобилей друг к другу снизу поля (в ситуации, когда самый дальний автомобиль заезжает в  $n$ -й вход, а расстояния между каждой парой соседних автомобилей после «прижатия» равно 1, дальний автомобиль нужно будет доставить на  $(u - 1)$ -ю горизонтальную дорогу, для чего нужно  $(n - u + 1)$  вертикальный сдвиг, и 1 слой оставляем на выезд на горизонтальную дорогу в конце).

Имеем  $m_1 \geq n - u + 2$ .

Далее, на первой части Этапа 2, как и в прошлый раз, отправляем автомобили прямо в сторону выходов, параллельно друг другу, до  $(m - 2u + 3)$  слоя (см. ниже).

Пусть номера выходов в порядке возрастания:  $j_1 \dots j_u$ .

На Этапе 2:

1. Отправляем первый снизу автомобиль в нижний выход строго по нижней границ;
2. Все остальные въехавшие автомобили переводим на вертикальные параллельные дороги с направлением вверх, сохраняя «очередность», то есть  $(u - i)$ -й снизу автомобиль поедет по  $i$ -й вертикальной дороге Этапа 2 вверх;
3. Затем разворачиваем дороги через горизонтальные проезды вниз, обеспечивая при этом сближение траекторий каждой пары соседних автомобилей до расстояния 1 между ними;
4. Затем ведем  $u - i$ -й автомобиль до  $i$ -го сверху входа, и по горизонтальным проездам выводим на выход. То есть, с ближайшей к концу поля вертикальной дороги автомобиль направляем на последний выход, со второй с конца - на предпоследний, и т.д.

Для такой схемы на втором этапе необходимое количество слоев:

$$m_2 \geq 2(u - 1) - 1 = 2u - 3.$$

Учитывая, что на Этап 1 по-прежнему необходимо количество слоев

$$m_1 \geq n - u + 2,$$

получаем итоговую оценку на длину поля, необходимую для реализации описанной схемы:

$$m = m_1 + m_2 \geq n - u + 2 + 2u - 3 = n + u - 1.$$

При такой стратегии так же получаем выполнение всех необходимых условий и разделение потоков с разных входов на разные выходы.  $\square$

Покажем теперь возможность обучения в данном случае.

**Теорема 6.** *Для  $m \geq n + u - 1$  существует алгоритм обучения светофоров (то есть конфигурация автоматов) с локальным полем видимости радиуса 2, приводящий к организации требуемой доставки.*

*Доказательство.* Использовать подход, в точности аналогичный случаю с 2 выходами в контексте обучения тут не получится, так как он основан на переходах  $\{a_1, a_2\}$ , где обнаруживается, с какого момента можно пускать автомобили напрямую через слои (то есть где заканчивается «сближение» автомобилей у нижней границы). Это обнаружение происходит при параллельном «достаточно близком» проезде двух автомобилей по нижней границе и рядом с ней, что в случае двух входов позволяет определить порог слоев «сближения». В случае же  $u > 2$  входов такой порог должен быть установлен в момент, когда на одном слое проезжают параллельно  $u$  автомобилей, причем каждая пара соседних находится «достаточно близко» друг к другу. В общем случае такую ситуацию не получается отловить при радиусе видимости светофора  $< n$ .

Однако, если исключить из стратегии часть Этапа 2, где автомобили едут параллельно друг другу, то можно не отлавливать слой, с которого начинается отправка автомобилей по прямой после «сближения» (то есть конец Этапа 1), а вместо этого зафиксировать  $2u - 3$  слоя для этапа 2, а все, что до них - отдать под поведение «сближения» (то есть по всем слоям 1 этапа автомобили будут двигаться «змейкой», как двигались при «сближении», просто после сближения их траектории будут параллельны траектории движения нижнего автомобиля, описанной выше в алгоритме доставки для  $u = 2$ ). Пример: Рис. 21 ( $n = 6, m = 10, u = 3$ ).

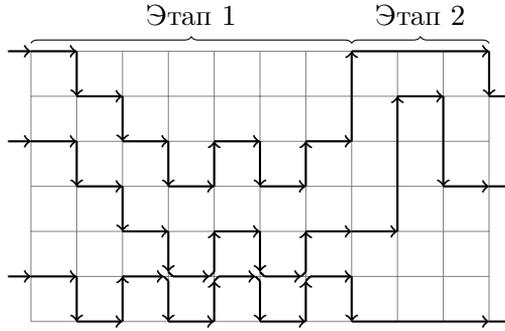


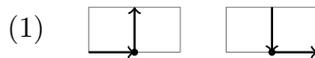
Рис. 21

Недостаток данной стратегии в том, что средняя скорость движения по слоям меньше, чем в предыдущей стратегии, в силу отсутствия того, что слои, по которым раньше мы пропускали автомобили прямо, теперь подчиняются логике «сближения», поэтому при использовании нынешней стратегии автомобили будут доставляться медленнее.

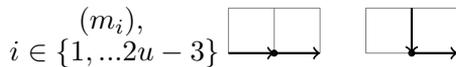
Однако, жертвуя скоростью доставки, мы получаем стратегию, для которой возможно обучение. Как раз его и осталось описать.

Начнем с описания состояний и структуры автоматов для разных типов светофоров:

1. Светофоры на горизонтальной границе поля (тип *HE*):

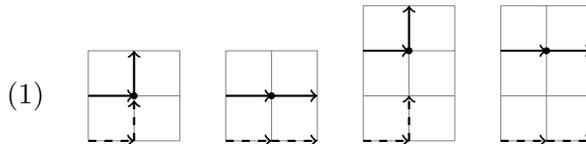


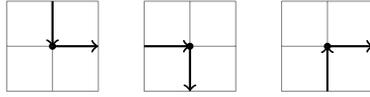
Это состояние для *HE*-светофоров на Этапе 1.



Это группа состояний светофоров второго этапа (послойно).

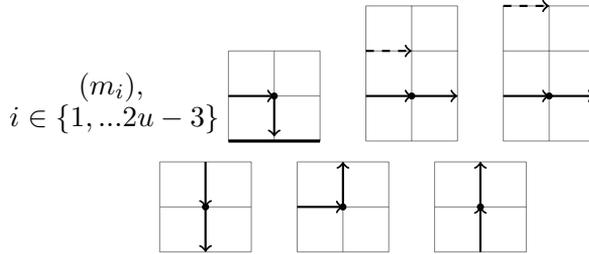
2. Внутренние светофоры (тип *CORE*):





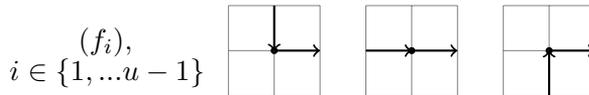
Здесь имеется в виду, что если на светофор приезжает автомобиль, и на соседний к нему снизу светофор, либо на соседний через 1 ячейку, в то же время параллельно приезжает автомобиль, и его направляет либо вправо, либо вниз, то и текущий светофор поступает так же, направляя вправо или вниз соответственно. Во всех остальных случаях его срабатывают правила, описанные на нижних рисунках.

Это состояние *CORE*-светофоров на Этапе 1.

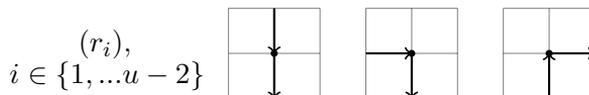


Это группа состояний светофоров второго этапа (послойно).

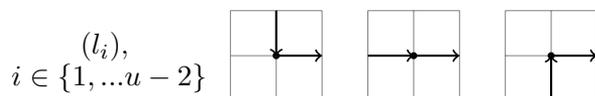
Первое правило означает, что если светофор находится рядом с нижней границе, то автомобили слева он направляет на границу. Если на светофор приехал автомобиль, и параллельно ему приехал автомобиль на соседний сверху светофор, либо отстоящий на 2 ячейки вверх, то текущий светофор направляет машинку прямо. Во всех остальных случаях он пользуется правилами, изображенными на нижних рисунках.



Имеем группу состояний, описывающих одно и то же поведение, и использующихся для разной длины горизонтальных проездов к выходам после разворота на вертикали.



Данная группа состояний, описывающих одно и то же поведение, нужна для организации поворотов параллельных траекторий движения автомобилей с горизонтального на вертикальное направление вниз, то есть перед проездом к выходам.



Данная группа состояний, описывающих одно и то же поведение, нужна для организации поворотов параллельных траекторий движения автомобилей с вертикального направления на горизонтальное, влево, во время разворота с попарным «сближением» на расстояние 1.

3. Угловые светофоры при границе с выходами (тип *ANG*):



Начальное состояние для нижнего угла.



Начальное состояние для верхнего угла.

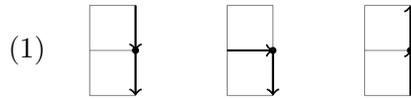


Состояние, когда нижний угол является выходом.

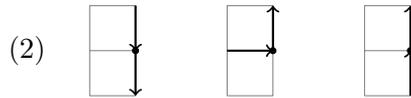


Состояние, когда верхний угол является выходом.

4. Светофоры на вертикальной границе с выходами (тип *VOUT*):



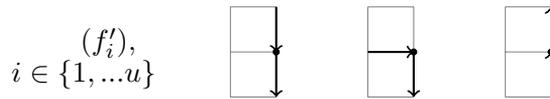
Изначальное состояние рассматриваемых светофоров.



Состояния светофоров под нижним входом.

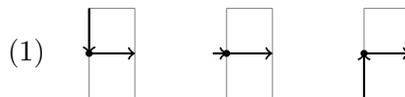


Группа состояний для обозначения различных выходов.

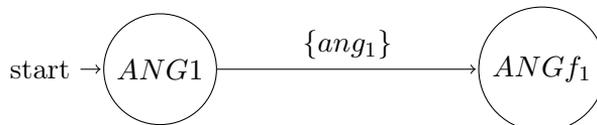


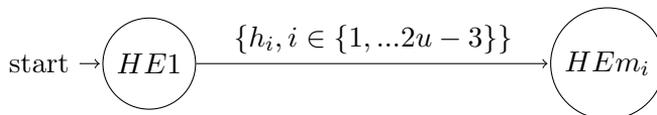
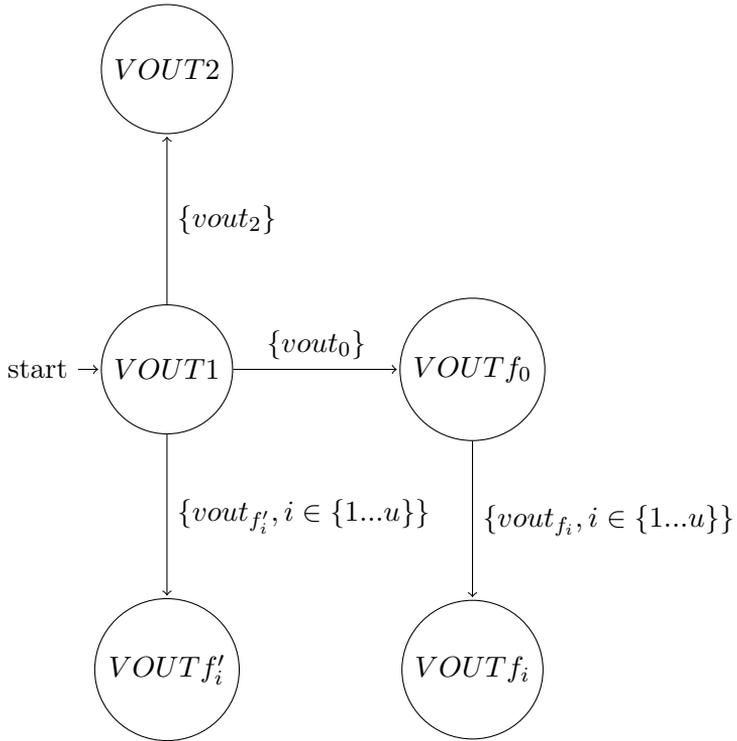
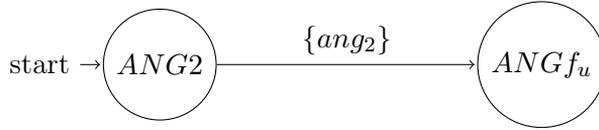
Группа состояний с поведением, аналогичным первому, обозначающие невыходные светофоры, с индексом, соответствующим ближайшему снизу выходному светофору.

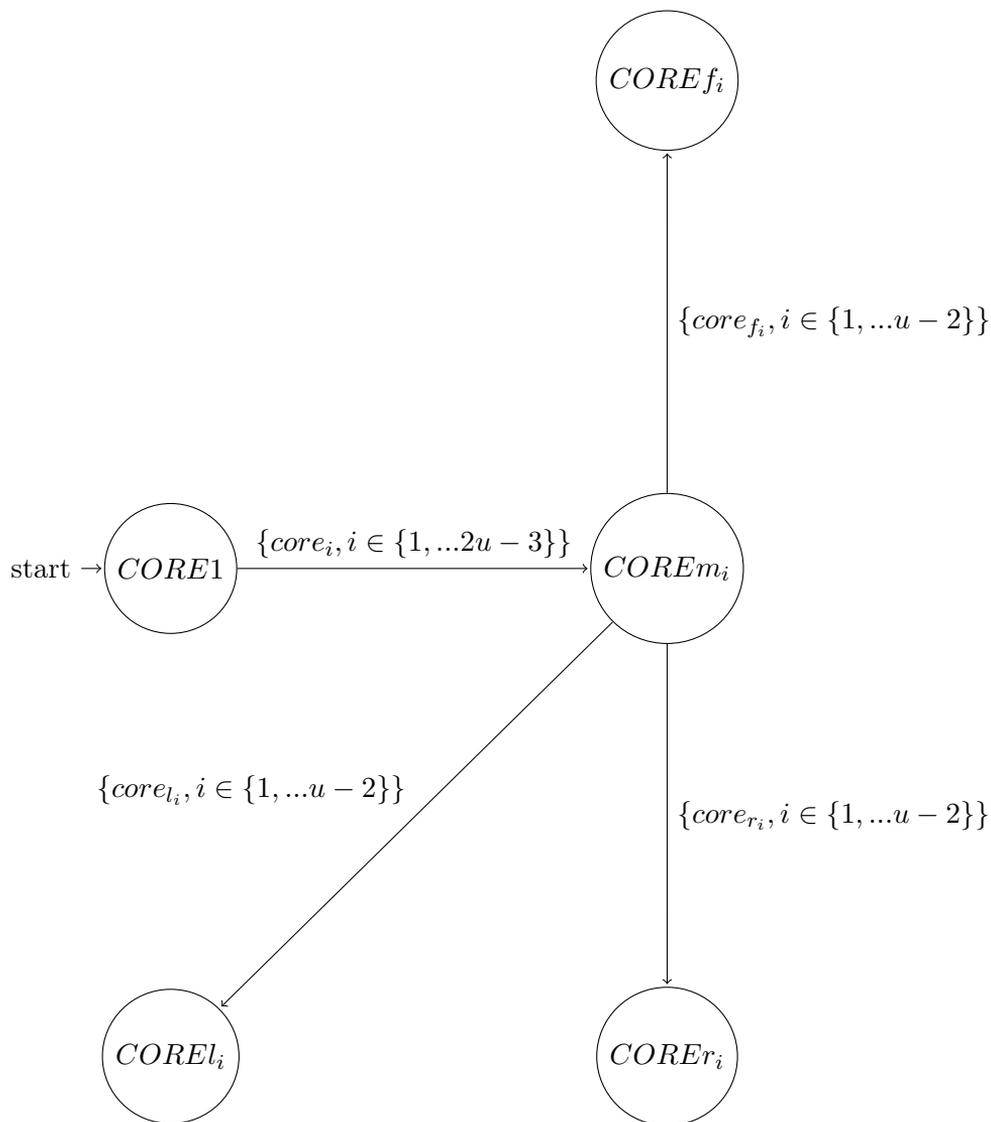
5. Светофоры на вертикальной границе со входами (тип *VEIN*):



Структура автоматом (для всех типов кроме *VEIN*, у которых всего одно состояние):



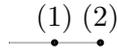




На рисунках группы состояний объединяем в одно, оставляя лишь общее обозначение (напр.  $VOUT_{f_i}$ ), однако подразумеваем, что к каждому из состояний группы идет по ребру.

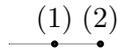
Приступим к описанию переходов:

1.  $\{he_1\}$



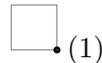
(1) — HE1, (2) — ANG  $\Rightarrow$   
 $he_1 : (1) : HE1 \rightarrow HE m_1$

2.  $\{he_i\}$



(1) — HE1, (2) — HE  $m_i, i \in \{1, \dots, 2u - 2\} \Rightarrow$   
 $he_{i+1} : (1) : HE1 \rightarrow HE m_{i+1}$

3.  $\{ang_1\}$



(1) — ANG1, (1) - объявлен выходом  $\Rightarrow$   
 $ang_1 : (1) : ANG1 \rightarrow ANG f_1$

4.  $\{ang_2\}$



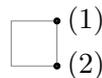
(1) — ANG2, (1) - объявлен выходом  $\Rightarrow$   
 $ang_2 : (1) : ANG2 \rightarrow ANG f_2$

5.  $\{vout_0\}$



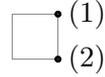
(1) — VOUT1, (1) - объявлен выходом  $\Rightarrow$   
 $vout_0 : (1) : VOUT1 \rightarrow VOUT f_0$

6.  $\{vout_2\}$



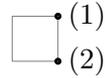
(1) — VOUT1, (2) — ANG1 или VOUT2  $\Rightarrow$   
 $vout_2 : (1) : VOUT1 \rightarrow VOUT2$

6.  $\{vout_{f_1}\}$



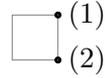
(1) –  $VOUT f_0$ , (2) –  $VOUT2$  или  $ANG1 \Rightarrow$   
 $vout_{f_1} : (1) : VOUT f_0 \rightarrow VOUT f_1$

7.  $\{vout_{f'_i}\}$



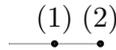
(1) –  $VOUT1$ , (2) –  $VOUT f_i$  или  $VOUT f'_i, i \in \{1, \dots, u\} \Rightarrow$   
 $vout_{f'_i} : (1) : VOUT1 \rightarrow VOUT f'_i$

8.  $\{vout_{f_i}\}$



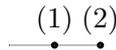
(1) –  $VOUT f_0$ , (2) –  $VOUT f_i$  или  $VOUT f'_i, i \in \{1, \dots, u - 1\} \Rightarrow$   
 $vout_{f_{i+1}} : (1) : VOUT f_0 \rightarrow VOUT f_{i+1}$

9.  $\{core_i\}$



(1) –  $CORE1$ , (2) –  $VOUT$  или  $ANG \Rightarrow$   
 $core_1 : (1) : CORE1 \rightarrow COREm_1$   
(1) –  $CORE1$ , (2) –  $COREm_i, i \in \{1, \dots, 2u - 4\} \Rightarrow$   
 $core_{i+1} : (1) : CORE1 \rightarrow COREm_{i+1}$

10.  $\{core_{f_i}\}$



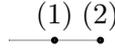
(1) –  $COREm_1$ , (2) –  $VOUT f_i, i \in \{2, \dots, u - 1\} \Rightarrow$   
 $core_{f_{u-i}} : (1) : COREm_1 \rightarrow CORE f_{u-i}$   
(1) –  $COREm_k$ , (2) –  $CORE f_i, i \in \{2, \dots, u - 1\} \Rightarrow$   
 $core_{f_{i-1}} : (1) : COREm_k \rightarrow CORE f_{i-1}$

11.  $\{core_{r_i}\}$



$(1) - COREm_1, (2) - ANG \Rightarrow$   
 $core_{r_1} : (1) : COREm_1 \rightarrow COREr_1$   
 $(1) - COREm_k, (2) - COREr_i, i \in \{1, \dots, u-3\} \Rightarrow$   
 $core_{r_{i+1}} : (1) : COREm_k \rightarrow COREr_{i+1}$

12.  $\{core_{l_{u-2}}\}$



$(1) - COREm_k, (2) - COREr_{u-2} \Rightarrow$   
 $core_{l_{u-2}} : (1) : COREm_k \rightarrow COREl_{u-2}$

13.  $\{core_{l_i}\}$



$(1) - COREm_k, (2) - COREl_i, i \in \{2, \dots, u-2\} \Rightarrow$   
 $core_{l_{i-1}} : (1) : COREm_k \rightarrow COREl_{i-1}$

В итоге через  $max\{2u-2, n\}$  шагов игры второй слой будет обучен, а первый - обучен изначально.

Теорема доказана. □

## Список литературы

- [1] Э.Э. Гасанов, В.Б. Кудрявцев, *Теория хранения и поиска информации*, ФИЗМАТЛИТ, М., 2002.

### On the self-organizing system of the traffic lights that ensure uninterrupted traffic

Muravev A.K.

The present paper considers the cooperative game about the transport delivery on the rectangular grid. The concepts of congestion and cars collision on it, as well as the automata as the agents in

the traffic lights of the grid regulating movement of transport, are introduced.

It's proved that there exists a structure of the automata with local area of visibility leading to the self-organization of the system of traffic lights ensuring an arbitrarily long, correct transport delivery without any congestions and cars collisions for any number of grid's outputs.

**Keywords:** cooperative game, delivery, traffic organization, agent, homogeneous structure, automaton.