

САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
КАФЕДРА МАТЕМАТИЧЕСКОЙ ТЕОРИИ ИГР И СТАТИСТИЧЕСКИХ РЕШЕНИЙ

Бебяков Александр Михайлович

Выпускная квалификационная работа бакалавра

**Теоретико-игровое моделирование передачи
данных в беспроводных сетях**

Направление 010300

Фундаментальная информатика и информационные технологии

Научный руководитель,
кандидат физ.-мат. наук,
доцент
Седаков А. А.

Санкт-Петербург

2016

Содержание

Введение	3
Постановка задачи	5
Обзор литературы	8
Глава 1. Математическое описание задачи	9
1.1. Модель сети	9
1.2. Биматричная игра передачи данных	11
1.3. Алгоритм Лемке-Хоусона	12
1.4. Расчет характеристик	15
Глава 2. Задача поиска ожидаемого времени завершения передач данных при равновесных стратегиях	17
2.1. Схема поиска	17
2.2. Пример	18
Глава 3. Реализация алгоритма	24
3.1. Анализ и выбор инструментов реализации	24
3.2. Реализация в среде MATLAB	26
Выводы	29
Заключение	30
Список литературы	31

Введение

Телекоммуникационные технологии имеют особое значение в жизни современного общества, поскольку в настоящее время без применения телекоммуникационных систем ни одна служба и ни одно предприятие не сможет проводить оптимальное управление элементами своей структуры, отсутствие доступа к своевременной информации может приводить к огромным издержкам и упущенным выгодам. Во многих отраслях деятельности человека наличие устойчивой оперативной связи является одним из важнейших факторов развития. Таким образом, благосостояние государства напрямую зависят от уровня качества и эффективности телекоммуникационных и информационных технологий.

Применение беспроводных сетей занимает значимое место в комплексе мер, направленных на автоматизацию производственных процессов, организационной и кадровой работы предприятий, а также в системах обеспечения безопасности, поскольку применение кабельных средств связи не всегда экономически целесообразно [1], а внесение кардинальных изменений в инфраструктуру уже существующих систем иногда совершенно неприемлемо. Беспроводные сети сотовой топологии, реализующие технологии выбора эффективных и надежных маршрутов находят применение в производственных помещениях с наличием радиочастотных помех. С анализом данных сетей можно ознакомиться в работе [2].

Идея отделения уровня управления от уровня передачи данных в программно-конфигурируемых сетях предоставляет возможности создания собственных протоколов маршрутизации, а также систем адаптирующихся к изменяющимся условиям функционирования. Подобные технологии очень востребованы так, как позволяют создавать надежные и легкие в эксплуатации решения. В статье [3] отмечаются такие преимущества подхода, как упрощенная настройка сети, возможность более гибкого решения задач безопасности, маршрутизации, управления полосой

пропускания с ориентацией на тонкую настройку систем под задачи каждого конкретного потребителя.

Актуальность теоретико-игрового моделирования передачи данных в беспроводных сетях подтверждается необходимостью разработки и создания систем маршрутизации способных производить оптимальное распределение нагрузки с целью эффективного использования ресурсов сети. Также одной из важных задач является выбор класса систем связи для реализации в различных сферах деятельности общества с учетом требований к качеству обслуживания (QoS) и обеспечению энергоэффективности работы автономных мобильных устройств, простоты, экономичности и скорости создания системы таких устройств.

Теория игр позволяет проводить анализ оптимальных стратегий в процессах с двумя или более сторонами, преследующими свои интересы. Каждый из участников таких процессов, обладая собственной целью, вынужден выбирать стратегию, результат применения которой может зависеть от принятых стратегий остальных участников. Теория игр предоставляет возможность участникам, обладающим информацией о характеристиках сторон, выбирать стратегии, оптимизирующие собственный выигрыш.

Целью данной работы является моделирование передачи данных в беспроводной сети с применением аппарата теории игр и определение требуемых характеристик данной сети. Для достижения поставленной цели решаются задачи построения математической модели передачи данных как неантагонистической игры, реализации алгоритма решения игры с помощью программных средств и задача определения значения заявленных характеристик на основе поведения участников системы.

Постановка задачи

Рассматривается модель программно-определяемой беспроводной сети на базе стандарта 802.11, состоящей из двух подсетей, представляющих организации в сети, автономных общих точек доступа и точки доступа (далее сток), являющейся пограничным маршрутизатором данной сети подключенным к более общей сети.

Каждая из подсетей представлена базовой станцией (далее источник данных) и промежуточными точками доступа данной подсети.

Точка доступа не являющаяся ни источником данных, ни стоком будет называться промежуточным узлом.

Между любыми двумя узлами i и j определена оцененная вероятность t_{ij} успешной передачи данных по прямой связи, которая предполагается неизменной. Подтверждение передачи данных происходит без ошибок. Известна базовая пропускная способность каждого узла системы.

Все точки доступа в сети считаются фиксированными, но возможно временное изменение топологии системы вследствие того, что промежуточные узлы способны переходить в неактивное состояние для экономии электроэнергии или для проведения профилактических действий. Режим работы для каждого из промежуточных узлов задает вероятности нахождения в активном состоянии. Данные изменения топологии контролируются так, чтобы не была нарушена связь между каждым из источников и стоком. В периоды передачи данных изменения состояний узлов не производятся.

Перед источниками поставлена задача выбора маршрута с оптимальным временем передачи информации, исходя из знаний о таких характеристиках сети, как активность промежуточных узлов во время текущего периода пересылки, вероятность успешной передачи информации по непосредственному соединению для любых двух узлов, базовые пропускные способности всех узлов. В качестве оценки времени передачи по маршруту источники используют численную характеристику LAETT [4].

Расчет LAETT для каждого источника производится на основе принятого им маршрута и нагрузки на сеть определяемой путем передачи избранным другим источником. Источники принимают решение о выборе маршрута передачи данных одновременно и независимо. Реализация данного типа маршрутизации требует централизованной фиксации каналов передачи, что достигается в программно-определяемых сетях.

В качестве оценки времени передачи при фиксировании маршрутов x и y источников 1 и 2 соответственно в данной модели применяется характеристика LAETT оцененного времени передачи по маршруту с учетом нагрузки на сеть. Для передачи напрямую между узлами i и j вводится:

$$LAETT_{ij} = ETX_{ij} \times \frac{S}{(RC_i + RC_j) / 2\gamma_{ij}},$$

где S – объем данных, который необходимо переслать из i в j , ETX_{ij} – рассчитываемое количество попыток пересылки данных из i в j [5], γ_{ij} – коэффициент качества связи, величина RC_i задаваемая

$$RC_i = B_i - \sum_{k \in N_i} f_{ik} \cdot \gamma_{ik}$$

характеризует остаточную пропускную способность узла i , B_i – базовая пропускная способность узла i , N_i – множество потоков передачи проходящих через узел i , f_{ik} – скорость передачи через узел i используемая потоком k , γ_{ik} – коэффициент качества связи – полагается равным 1.

В рассматриваемой модели S полагается равным 1 для обоих источников, γ_{ij} считается равным 1, ETX_{ij} принимается равным значению обратному вероятности t_{ij} успешной передачи данных от i к j непосредственно, для всех $t_{ij} > 0$:

$$ETX_{ij} = \frac{1}{t_{ij}}$$

Оценкой времени передачи по всему маршруту z является сумма:

$$LAETT(z) = \sum_{(i,j) \in \text{множество дуг маршрута } z} LAETT_{ij} \quad (1)$$

Расчет $LAETT$ для каждого пути передачи может быть произведен после установления обоих маршрутов, поскольку источники выбирают маршруты передачи одновременно и независимо. $LAETT(x, y)$ оценивает время передачи данных по заданному маршруту x с учетом имеющейся нагрузки сети, определяемой наличием передачи данных по маршруту y :

$$LAETT(x, y) = \sum_{(i, j) \in \text{множество дуг маршрута } x} LAETT_{ij}(y), \quad (2)$$

$$LAETT_{ij}(y) = ETX_{ij} \times \frac{1}{(RC_i(y) + RC_j(y)) / 2}$$

Если маршруты проходят через один узел, то при фиксации потоков передачи данных происходит равное распределение пропускной способности узла для обслуживания пересылки по каждому из маршрутов и, следовательно:

$$RC_i(y) = \begin{cases} C_i, & \forall e \in \{\text{множество дуг маршрута } y\} : e = (i, k) \vee e = (k, i) \\ \frac{C_i}{2}, & \text{иначе} \end{cases}, \quad (3)$$

где $RC_i(y)$ – остаточная пропускная способность узла i при наличии нагрузки на сеть определяемой маршрутом передачи y , с учетом равномерного деления базовой пропускной способности $C_i > 0$ узла i для пересылки данных на каждом из маршрутов.

Пусть для оценки эффективности передачи данных в сети, определяемой топологией, пропускными способностями и заданными режимами работы промежуточных узлов, в данной модели применяется функция, принимающая наибольшее среди значений $LAETT$ для источников 1 и 2 в зависимости от выбранных ими маршрутов x и y соответственно:

$$\varphi(x, y) = \max(LAETT(x, y), LAETT(y, x)), \quad (4)$$

Значение функции предполагает собой время завершения всех передач в сети с момента начала одновременной пересылки данных от источников 1 и 2.

Требованием к решению задачи является соблюдение условия равновесия по Нэшу при выборе вероятностного распределения определения маршрутов передачи данных от первого и второго источников.

Обзор литературы

В статье [6] была отмечена значимость задачи моделирования и построения программно-конфигурируемых сетей (ПКС), а также проведено описание разработки автоматизированной программной системы моделирования программно-определяемых сетей основанных на протоколе OpenFlow. Изучение графовых моделей и потоков распределения трафика пользователей при формировании протоколов маршрутизации в ПКС рассматривается в работе [7]. Статья [8] посвящена обзору «метрик» - применяемых протоколами маршрутизации числовых характеристик путей передачи данных. При разработке метрик возможен учет различных факторов системы: количество узлов, через которое потребуется передать пакет информации, энергетическая стоимость пересылки, полоса пропускания канала, надежность соединения. В результате вычисления подобных «метрик» маршрутизаторы получают возможность непосредственного сравнения маршрутов с точки зрения того или иного критерия эффективности. «Метрика» LAETT, применяемая в данной работе, охарактеризована как способствующая распределению нагрузки по всей сети. Классические принципы построения сетей представлены в книге [9]. В работе [10] была рассмотрена возможность применения решения Нэша задачи о сделках в качестве подхода к управлению в крупномасштабных сетях, в том числе маршрутизации. Книга [11] производит обзор элементов теории графов и теории игр и их приложений к различным стратегическим задачам в сетевых моделях.

Глава 1. Математическое описание задачи

1.1. Модель сети

В рамках рассматриваемой задачи предлагается модель системы, состоящей из двух источников данных (обозначены номерами 1 и 2), стока (номер 3), в который необходимо доставлять данные, k_1 промежуточных узлов первой подсети $\{4, \dots, k_1+3\}$, k_2 промежуточных узлов второй подсети $\{k_1+4, \dots, k_1+k_2+3\}$, k_3 общих промежуточных узлов $\{k_1+k_2+4, \dots, k_1+k_2+k_3+3\}$, где $k_1, k_2, k_3 \geq 0$, а также связей с определенным качеством $t_{ij} \in [0; 1]$ между этими элементами, задаваемым в форме матрицы $T [(k) \times (k)]$ (где $k = k_1+k_2+k_3+3$) вероятностей t_{ij} того, что пакет данных, отправленный из i , будет без ошибок принят в j при непосредственной передаче. Для каждого узла i сети указывается его **базовая пропускная способность** $C_i > 0$. Модель наглядно представима в форме графа $G (V, E)$ с вершинами $V = \{v_1, v_2, v_3, \dots, v_k\}$, дугами $E = \{e_{ij}: \text{соединяет } v_i \text{ и } v_j \mid t_{ij} > 0\}$, весами t_{ij} дуг e_{ij} .

В рамках задачи передачи данных от источников в сток выдвинуты следующие предположения относительно вида матрицы T :

Передача пересылаемых данных в источники не предполагается – первые два столбца T нулевые. Передача пересылаемых данных из стока в сеть не производится – третья строка T нулевая. Главная диагональ T заполнена нулями, так как узлы не посылают данные себе. Передача рассматриваемых данных в какую-либо подсеть извне не предполагается – элементы, дополняющие столбцы блока (строки $4, \dots, k_1+3$ и столбцы $4, \dots, k_1+3$) нулевые, за исключением первой строки, и элементы, дополняющие столбцы блока (строки k_1+4, \dots, k_1+k_2+3 и столбцы k_1+4, \dots, k_1+k_2+3) нулевые, за исключением второй строки.

Промежуточные узлы в период передачи данных могут быть активными и неактивными. Состояние сети определяется бинарным вектором $l = (1, 1, 1, l_4, \dots, l_k)$ активности узлов (если узел i активен, то $l_i = 1$, иначе $l_i = 0$). Ему соответствует граф $G_l (V, E_l)$. При неактивности узла, через него не

могут проходить маршруты, все дуги инцидентные его вершине не могут быть использованы для передачи и удаляются, следовательно, $E_l = E \setminus \{e \mid \text{дуга } e \text{ инцидентна } v_i, i: l_i = 0\}$. Вероятности активности каждого узла задаются вектором $p = (1, 1, 1, p_4, \dots, p_k)$ установленного режима работы, где $p_i > 0, i = 4, \dots, k$ и полагается, что два источника и сток активны всегда.

Состояние l системы является допустимым, если в $G_l(V, E_l)$ существует маршрут от первого источника до стока, а также маршрут от второго источника до стока. Обозначим множество допустимых состояний как S . В каждой задаче число $h = |S|$ допустимых состояний однозначно определяется топологией системы, задающей граф $G(V, E)$.

Для каждого состояния $l \in S$ сети вероятность нахождения системы в нем определим как

$$P_l = \frac{\prod_{i=1, \dots, k: l_i=1} p_i \cdot \prod_{i=1, \dots, k: l_i=0} (1-p_i)}{\sum_{s \in S} \left(\prod_{i=1, \dots, k: s_i=1} p_i \cdot \prod_{i=1, \dots, k: s_i=0} (1-p_i) \right)} \quad (5)$$

Таким образом, изначально заданные режимом работы вероятности активности узлов p_i определяют вектор распределения вероятности $P = (P_1, \dots, P_h)$ нахождения системы в допустимых состояниях при условии, что рассматриваются только допустимые состояния.

Источники 1 и 2 одновременно выбирают свой маршрут передачи для текущего состояния l системы среди всех простых цепей (чередующаяся последовательность вершин и рёбер, в которой любые два соседних элемента инцидентны и все вершины различны [12]) от своего передатчика до стока в графе $G_l(V, E_l)$.

Источники стремятся минимизировать ожидаемое время передачи данных, в качестве которого принимается характеристика $LAETT(x, y)$ (2), которая оценивает время передачи данных по заданному маршруту x с учетом имеющейся нагрузки сети, определяемой наличием пересылки данных по маршруту y .

1.2. Биматричная игра передачи данных

Рассмотрим семейство биматричных игр $\Gamma_l (A_l, B_l) = (X_l, Y_l, A_l, B_l)$ определяемых допустимыми состояниями $l \in S$.

В каждой из игр $\Gamma_l (A_l, B_l)$ игрок 1 – первый источник способный производить запрос выделения маршрута передачи данных среди работающих узлов в сток. Игрок 2 - второй источник со способностью запроса фиксации маршрута пересылки своих данных через активные узлы в сток.

Чистая стратегия i игрока 1 в состоянии l есть маршрут передачи данных x_i среди конечного множества всех маршрутов $X_l, |X_l| = m$ (простых цепей от первого источника данных до стока через промежуточные узлы) игрока 1 в допустимом состоянии l .

Чистой стратегией i игрока 2 в состоянии l является маршрут передачи данных y_i среди конечного множества всех маршрутов $Y_l, |Y_l| = n$ (простых цепей от второго источника данных до стока через промежуточные узлы) игрока 2 в допустимом состоянии l .

Так как игроки стремятся минимизировать ожидаемое время передачи по выбранному ими маршруту, в качестве значения выигрыша игрока 1 (2) при реализации ситуации в чистых стратегиях (x_i, y_j) , где $x_i \in X_l, y_j \in Y_l$, можно принять число обратное значению характеристики $LAETT$, вычисленному для маршрута $x = x_i (y_j)$, принимая во внимание нагрузку на сеть, создаваемую передачей данных игроком 2 (1) по маршруту $y = y_j (x_i)$. Значение $LAETT(x, y)$ вычисляется по формуле (2), где

$$ETX_{ij} = \frac{1}{t_{ij}} \quad (6)$$

Тогда значения a_{ij} (b_{ij}) матрицы A_l (B_l) выигрышей игрока 1 (2) задаются как:

$$a_{ij} = \frac{1}{LAETT(x_i, y_j)} \quad (7)$$

$$b_{ij} = \frac{1}{LAETT(y_j, x_i)} \quad (8)$$

Множеством смешанных стратегий игрока 1 в состоянии l является $U_l = \{u \mid ue^m = 1, u \geq 0^m, u \in R^m\}$, где $e^m = (1, \dots, 1)^T \in R^m$, $0^m = (0, \dots, 0)^T \in R^m$.

Множеством смешанных стратегий игрока 2 в состоянии l является $V_l = \{v \mid ve^n = 1, v \geq 0^n, v \in R^n\}$, где $e^n = (1, \dots, 1)^T \in R^n$, $0^n = (0, \dots, 0)^T \in R^n$.

В состоянии l выигрыши игроков 1 и 2 в смешанных стратегиях в ситуации (u, v) определим как $K_{1l}(u, v) = uA_l v^T$, $K_{2l}(u, v) = uB_l v^T$, $u \in U_l, v \in V_l$.

Таким образом, вводится смешанное расширение $\Gamma_l^*(A_l, B_l)$ игры $\Gamma_l(A_l, B_l)$ – бескоалиционная игра двух лиц $\Gamma_l^*(A_l, B_l) = (U_l, V_l, K_{1l}, K_{2l})$.

В рамках поставленной задачи решением игры является нахождение пары (u^*, v^*) , которая является равновесия по Нэшу:

$$\sum_{i=1}^m \sum_{j=1}^n u_i^* v_j^* \cdot a_{ij} \geq \sum_{i=1}^m \sum_{j=1}^n u_i v_j^* \cdot a_{ij}, \text{ для } \forall u_i \in U_l \quad (9)$$

$$\sum_{i=1}^m \sum_{j=1}^n u_i^* v_j^* \cdot b_{ij} \geq \sum_{i=1}^m \sum_{j=1}^n u_i^* v_j \cdot b_{ij}, \text{ для } \forall v_j \in V_l \quad (10)$$

Известно [13], что равновесие по Нэшу в смешанных стратегиях существует в конечных биматричных играх.

Решение может быть найдено алгоритмом Лемке-Хоусона с входными параметрами (A_l, B_l) .

1.3. Алгоритм Лемке-Хоусона

Алгоритм, вычисляющий равновесие по Нэшу в смешанных стратегиях в биматричных играх, заданных входными параметрами – (A, B) – матрицами $[m \times n]$ выигрышей игроков 1 (m чистых стратегий) и 2 (n чистых стратегий) соответственно. Доказано [14], что равновесие будет найдено алгоритмом при условии, что A, B – матрицы, состоящие из положительных элементов.

Пусть $M = \{1, 2, \dots, m\}$ и $N = \{m+1, m+2, \dots, m+n\}$ – множества индексов чистых стратегий игроков 1 и 2 соответственно. Спектром смешанной стратегии $u = (u_1, \dots, u_m)$ игрока 1 называется множество $S(u) = \{i \mid u_i > 0\}$. Спектр смешанной стратегии $v = (v_{m+1}, \dots, v_{m+n})$ игрока 2 определяется аналогично. (u, v) – равновесие по Нэшу в игре, заданной (A, B) , со скалярами $\alpha = uAv^T$ и $\beta = uBv^T$ выигрышей игроков 1 и 2 соответственно

тогда и только тогда, когда для любого $i \in S(u)$ и $j \in S(v)$ выполняется $(Av^T)_i = \alpha \geq (Av^T)_k, k = 1, \dots, m$ и $(uB)_j = \beta \geq (uB)_k, k = m+1, \dots, m+n$, что аналогично утверждению: либо $u_i = 0$, либо чистая стратегия j - лучшая ответная стратегия второго игрока, также либо $v_j = 0$, либо чистая стратегия i - лучшая ответная стратегия второго игрока.

Тогда можем записать для любого равновесия по Нэшу (u, v) :

$$Av^T \leq \alpha \cdot e^m,$$

$$B^T u^T \leq \beta \cdot e^n,$$

или

$$Av^T + z^T = \alpha \cdot e^m,$$

$$B^T u^T + r^T = \beta \cdot e^n,$$

где $e^m = (1, \dots, 1)^T \in R^m, e^n = (1, \dots, 1)^T \in R^n, z = (z_1, \dots, z_m) \geq 0, r = (r_{m+1}, \dots, r_{m+n}) \geq 0, z_i \neq 0$ тогда и только тогда, когда $u_i = 0, i = 1, \dots, m, r_j \neq 0$ тогда и только тогда, когда $v_j = 0, j = m+1, \dots, m+n$. Можем получить:

$$Av'^T + z'^T = e^m, \quad (11)$$

$$B^T u'^T + r'^T = e^n, \quad (12)$$

где $v' = v / \alpha, z' = z / \alpha, u' = u / \beta, r' = r / \beta$. Можем заметить, что $v = v' \alpha, u = u' \beta$ и задача поиска u и v сводится к нахождению u' и v' . Запишем формулы (11), (12) в матричном виде:

$$(E_m A) (z' v')^T = e^{m+n}, \quad (13)$$

$$(B^T E_n) (u' r')^T = e^{n+n}, \quad (14)$$

где E_m и E_n - единичные матрицы порядка m и n соответственно. Из ограничений на вид z и u , а также на вид r и v , можно определить, что $m+n$ переменных из $2m+2n$ (z_i' и u_i' , при $i = 1, \dots, m, r_j'$ и v_j' , при $j = m+1, \dots, m+n$) должны быть равны нулю, а оставшиеся вычислимы из формул (13), (14).

Пусть u' характеризуется ярлыком $L(u')$, v' - ярлыком $L(v')$:

$$L(u') = \{i \mid u_i' = 0\} \cup \{j \mid r_j' = 0\}$$

$$L(v') = \{i \mid v_i' = 0\} \cup \{j \mid z_j' = 0\}$$

Ярлык пары (u', v') определяется как

$$L(u', v') = L(u') \cup L(v')$$

Говорится, что (u', v') имеет полный ярлык, если $L(u', v') = M \cup N = \{1, 2, \dots, m+n\}$.

Ситуация, выражаемая парой смешанных стратегий $(u = u'\beta, v = v'\alpha)$, является равновесием по Нэшу в биматричной игре (A, B) тогда и только тогда, когда (u', v') имеет полный ярлык.

Множество всех возможных пар (u', v') можно представить многогранником $W = \{(u', v') \in R^m \times R^n \mid u' \geq 0, ue^m = 1, Av'^T \leq e^m, v' \geq 0, ve^m = 1, B^T u'^T \leq e^m\}$, где

$$u = \frac{u'}{\sum_i u'_i},$$

$$v = \frac{v'}{\sum_j v'_j}$$

Идея алгоритма состоит в направленном переборе пар (u', v') , находящихся в вершинах W совершая переходы по ребрам таким образом, чтобы попеременно заменять один из элементов ярлыков $L(u')$ и $L(v')$, до тех пор, пока не будет найдена пара с полным ярлыком. Поиск начинается с выхода из вершины $((0, \dots, 0), (0, \dots, 0))$ имеющей полный ярлык.

Рассмотрим следующую интерпретацию алгоритма [15].

Начальному шагу алгоритма соответствует назначение таблиц $(B^* R^* \lambda) = (B^T E_n e^n)$ и $(Z^* A^* \gamma) = (E_m A e^m)$ соответственно, где $e^n = (1, \dots, 1)^T \in R^n$, $e^m = (1, \dots, 1)^T \in R^m$. Задается столбец ярлыков $L_1 = (m+1, m+2, \dots, m+n)$ для первой таблицы и $L_2 = (1, 2, \dots, m)$ для второй. В качестве вспомогательных значений для таблиц 1 и 2 определяются $h_1 = 1$, $h_2 = 1$ соответственно. Полагается, что для следующего шага будет использована таблица $(B^* R^* \lambda)$ и ярлык $i = 1$.

Далее преобразования таблиц $(B^* R^* \lambda)$ и $(Z^* A^* \gamma)$ совершаются попеременно (с текущей таблицей $D = (B^* R^* \lambda)$ или $(Z^* A^* \gamma)$, определенным для нее столбцом ярлыков $L = L_1$ или L_2 соответственно, вспомогательным

значением $h = h_1$ или h_2 соответственно и последним из извлеченных ярлыков i) следующим образом:

- 1) Из неотрицательных элементов столбца d^i (i столбец таблицы D) найти такой элемент $c = d_{ji}$, что отношение $d_{j(m+n)} / d_{ji}$ является минимальным;
- 2) для каждой строки d_k , $k \neq j$ произвести преобразование: $d_k := (c \cdot d_k - d_{ki} \cdot d_j) / h$, где h – вспомогательное значение для таблицы D ;
- 3) в столбце ярлыков L для таблицы D извлечь ярлык j и заменить на i ;
- 4) вспомогательное значение h для таблицы D заменить на c .

На каждом шаге $u' = (u_1', \dots, u_m')$, $v' = (v_{m+1}', \dots, v_{m+n}')$:

$$u_i' = \begin{cases} 0 & i \notin L_1 \\ \lambda_k, \text{ где } k: L_{1k} = i \text{ иначе} & \end{cases},$$

$$v_i' = \begin{cases} 0 & i \notin L_2 \\ \gamma_k, \text{ где } k: L_{2k} = i \text{ иначе} & \end{cases},$$

$$L(u') = (M \cup N) \setminus \{i \mid i \in L_1\},$$

$$L(v') = (M \cup N) \setminus \{i \mid i \in L_2\}$$

Алгоритм завершает работу, когда $L(u', v') = L(u') \cup L(v') = M \cup N$.

Результатом является равновесная по Нэшу ситуация (u, v) , где

$$u = \frac{u'}{\sum_i u_i'},$$

$$v = \frac{v'}{\sum_j v_j'}$$

1.4. Расчет характеристик.

Результаты моделирования могут быть использованы для оценки различных характеристик модели сети, находящихся в зависимости от определенного поведения источников.

В качестве подобных оценок в каждом из состояний $l \in S$ относительно принятых распределений вероятностей u_l и v_l случайного выбора маршрута будем рассматривать функции вида:

$$\varphi_l(u, v) = \sum_{i=1}^m \sum_{j=1}^n u_i v_j \cdot \varphi(x_i, y_j), \quad (15)$$

где m и n – число маршрутов в состоянии $l \in S$ для первого и второго источника соответственно, $u_l : u_l e^m = 1, u_l \geq 0^m, u \in R^m, v_l : v_l e^n = 1, v_l \geq 0^n, v_l \in R^n, u_l$ и v_l – распределения вероятности выбора маршрутов для источников 1 и 2 соответственно.

Так как допустимые состояния сети независимы и составляют полную группу событий в данной задаче, общую характеристику модели сети будем определять как математическое ожидание соответствующих величин в состояниях $l \in S$:

$$\Phi = \sum_{l \in S} P_l \cdot \varphi_l(u_l, v_l) \quad (16)$$

В данной работе будем рассматривать характеристику, расчет которой основан на формуле (4).

Глава 2. Задача поиска ожидаемого времени завершения передач данных при равновесных стратегиях

2.1. Схема поиска.

1. Определить количества промежуточных узлов k_1, k_2, k_3 в первой подсети, второй подсети и общих промежуточных узлов соответственно.
2. Задать элементы (p_4, \dots, p_k) вектора $p = (1, 1, 1, p_4, \dots, p_k)$ вероятностей активности узлов, где $k = k_1 + k_2 + k_3 + 3$.
3. Указать матрицу T , элементы t_{ij} которой заданы вероятностью успешной передачи данных от узла i узлу j по прямому беспроводному подключению в рамках рассматриваемой задачи (см. требования к T в пункте 1.1) при условии, что все узлы находятся в активном состоянии.
4. Задать базовые пропускные способности узлов C_i .
5. Определить граф $G(V, E)$ через множество вершин $V = \{v_1, v_2, v_3, \dots, v_k\}$ соответствующих узлам сети и множество связей $E = \{e_{ij}: \text{соединяет } v_i \text{ и } v_j \mid t_{ij} > 0\}$.
6. Произвести в графе $G(V, E)$ поиск всех простых цепей с началом в вершине v_1 и концом в вершине v_3 , обозначив их множество как M .
7. Произвести в графе $G(V, E)$ поиск всех простых цепей с началом в вершине v_2 и концом в вершине v_3 , обозначив их множество как N .
8. Перебирая все возможные бинарные векторы состояний $l = (1, 1, 1, l_4, \dots, l_k)$, определить каждое допустимое состояние следующим образом: для каждого $i: l_i = 0$ удалить из множеств M_l и N_l (изначально принимающих значения M и N соответственно) простые цепи содержащие v_i . Если в результате и M_l , и N_l содержат как минимум один элемент, то состояние сети l является допустимым, а M_l и N_l образуют множества маршрутов в состоянии l для игроков 1 и 2 соответственно. Множество всех допустимых состояний – S .

9. В каждом состоянии $l \in S$ определить множества чистых стратегий $X_l = (x_1, \dots, x_m)$ игрока 1, $Y_l = (y_1, \dots, y_n)$, перечислив маршруты из M_l и N_l соответственно.
10. Вычислить матрицы A_l и B_l , используя (7) и (8).
11. Используя алгоритм Лемке-Хоусона на начальных данных (A_l, B_l) найти равновесную по Нэшу ситуацию в чистых стратегиях (u_l^*, v_l^*) .
12. Вычислить $\varphi_l(u_l^*, v_l^*)$ по формуле (15).
13. Найти P используя (5), Произвести расчет Φ по формуле (16).

2.2. Пример.

1. Зададим количества промежуточных узлов $k_1 = 1, k_2 = 1, k_3 = 8$ в первой подсети, второй подсети и общих промежуточных узлов соответственно.
2. Укажем $p = (1, 1, 1, 0.6, 0.7, 0.95, 0.9, 0.7, 0.6, 0.8, 0.6, 0.8, 0.8)$.
3. Определим матрицу T :

$$T = \begin{pmatrix} 0 & 0 & 0 & 0.85 & 0 & 0.93 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.8 & 0.7 & 0.7 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.9 & 0.9 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.95 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.85 & 0.87 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.85 & 0 & 0 & 0 & 0.87 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.87 & 0 & 0 & 0.8 & 0 & 0.83 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.8 & 0 & 0 & 0 & 0.75 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.87 & 0 & 0 & 0 & 0.9 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.83 & 0 & 0.9 & 0 & 0.8 & 0.6 \\ 0 & 0 & 0.75 & 0 & 0 & 0 & 0 & 0 & 0.75 & 0 & 0.8 & 0 & 0 \\ 0 & 0 & 0.8 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.6 & 0 & 0 \end{pmatrix}$$

4. Определим $C = (2, 4, 4, 2, 1, 3, 2, 2, 2, 2, 2, 2, 2)$.

5. На рис. 1 представлен граф $G(V, E)$.

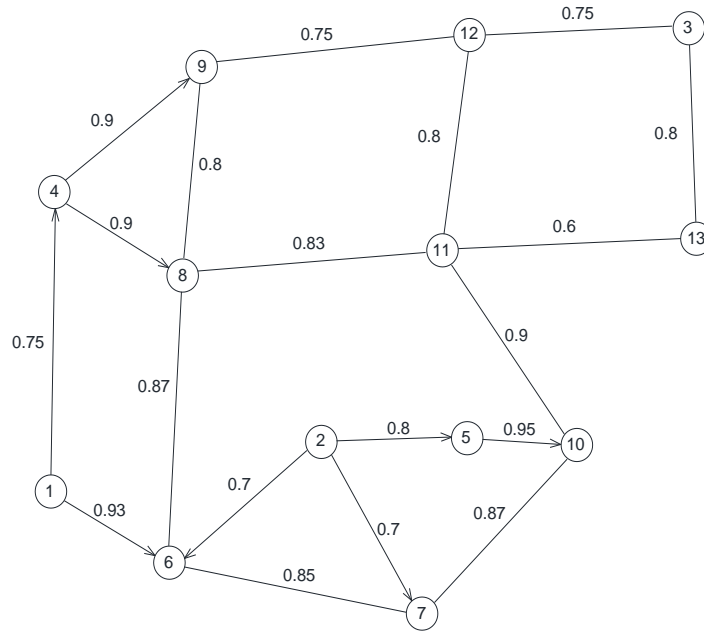


Рис. 1.

6. Выпишем все простые цепи в графе $G(V, E)$ с началом в вершине v_1 и концом в вершине v_3 в $M = \{(1, 4, 8, 6, 7, 10, 11, 12, 3), (1, 4, 8, 6, 7, 10, 11, 13, 3), (1, 4, 8, 9, 12, 3), (1, 4, 8, 9, 12, 11, 13, 3), (1, 4, 8, 11, 12, 3), (1, 4, 8, 11, 13, 3), (1, 4, 9, 8, 6, 7, 10, 11, 12, 3), (1, 4, 9, 8, 6, 7, 10, 11, 13, 3), (1, 4, 9, 8, 11, 12, 3), (1, 4, 9, 8, 11, 13, 3), (1, 4, 9, 12, 3), (1, 4, 9, 12, 11, 13, 3), (1, 6, 7, 10, 11, 8, 9, 12, 3), (1, 6, 7, 10, 11, 12, 3), (1, 6, 7, 10, 11, 13, 3), (1, 6, 8, 9, 12, 3), (1, 6, 8, 9, 12, 11, 13, 3), (1, 6, 8, 11, 12, 3), (1, 6, 8, 11, 13, 3)\}$ где каждому x - элементу M соответствует маршрут проходящий последовательно через вершины с индексами равными элементам вектора x . $|M| = 19$.
7. Выпишем все простые цепи в графе $G(V, E)$ с началом в вершине v_2 и концом в вершине v_3 в $N = \{(2, 5, 10, 7, 6, 8, 9, 12, 3), (2, 5, 10, 7, 6, 8, 9, 12, 11, 13, 3), (2, 5, 10, 7, 6, 8, 11, 12, 3), (2, 5, 10, 7, 6, 8, 11, 13, 3), (2, 5, 10, 11, 8, 9, 12, 3), (2, 5, 10, 11, 12, 3), (2, 5, 10, 11, 13, 3), (2, 6, 7, 10, 11, 8, 9, 12, 3), (2, 6, 7, 10, 11, 12, 3), (2, 6, 7, 10, 11, 13, 3), (2, 6, 8, 9, 12, 3), (2, 6, 8, 9, 12, 11, 13, 3), (2, 6, 8, 11, 12, 3), (2, 6, 8, 11, 13, 3), (2, 7, 6, 8, 9, 12, 3), (2, 7, 6, 8, 9, 12, 11, 13, 3), (2, 7, 6, 8, 11, 12, 3), (2, 7, 6, 8, 11, 13, 3), (2, 7, 10, 11, 8, 9, 12, 3), (2, 7, 10, 11, 12, 3), (2, 7, 10, 11, 13, 3)\}$, где

каждому y - элементу N соответствует маршрут проходящий последовательно через вершины с индексами равными элементам вектора y . $|N| = 21$.

8. Множество S допустимых состояний содержит 178 векторов по 13 элементов.

Рассмотрим шаги 9-12 только для состояния $l = (1, 1, 1, 1, 1, 0, 0, 1, 1, 1, 1, 1, 1)$, которое определяет граф $G_l(V, E_l)$:

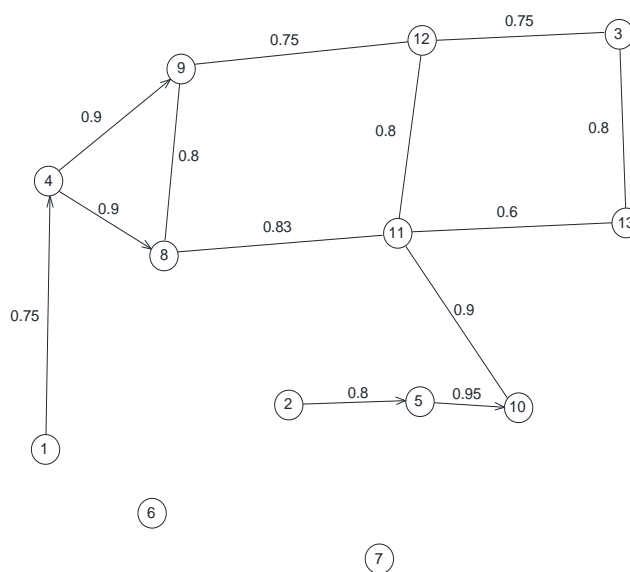


Рис. 2.

9. Определим множества чистых стратегий – маршрутов игроков в графе $G_l(V, E_l)$ (рис. 2.)

$X_l = (x_1 (1, 4, 8, 9, 12, 3), x_2 (1, 4, 8, 9, 12, 11, 13, 3), x_3 (1, 4, 8, 11, 12, 3), x_4 (1, 4, 8, 11, 13, 3), x_5 (1, 4, 9, 8, 11, 12, 3), x_6 (1, 4, 9, 8, 11, 13, 3), x_7 (1, 4, 9, 12, 3), x_8 (1, 4, 9, 12, 11, 13, 3)), |X_l| = 8$.

$Y_l = (y_1 (2, 5, 10, 11, 8, 9, 12, 3), y_2 (2, 5, 10, 11, 12, 3), y_3 (2, 5, 10, 11, 13, 3)), |Y_l| = 3$.

10. Вычислим A_l и B_l :

$$A_l = \begin{pmatrix} 0.2083 & 0.2820 & 0.3224 \\ 0.1450 & 0.1772 & 0.1733 \\ 0.2140 & 0.2447 & 0.2901 \\ 0.2342 & 0.2715 & 0.2249 \\ 0.1688 & 0.2123 & 0.2456 \\ 0.1812 & 0.2321 & 0.1972 \\ 0.2816 & 0.3423 & 0.4037 \\ 0.1770 & 0.1993 & 0.1944 \end{pmatrix}$$

$$B_l = \begin{pmatrix} 0.1658 & 0.2874 & 0.3110 \\ 0.1511 & 0.2450 & 0.2251 \\ 0.1737 & 0.2450 & 0.2718 \\ 0.1882 & 0.2905 & 0.2251 \\ 0.1511 & 0.2450 & 0.2718 \\ 0.1680 & 0.2905 & 0.2251 \\ 0.1847 & 0.2874 & 0.3110 \\ 0.1724 & 0.2450 & 0.2251 \end{pmatrix}$$

11. Составим таблицы $(B^* R^* \lambda)$:

L_1	1	2	3	4	5	6	7	8	9	10	11	λ
9	0.17	0.15	0.17	0.19	0.15	0.17	0.18	0.17	1	0	0	1
10	0.29	0.25	0.25	0.29	0.25	0.29	0.29	0.25	0	1	0	1
11	0.31	0.23	0.27	0.23	0.27	0.23	0.31	0.23	0	0	1	1

$(Z^* A^* \gamma)$:

L_2	1	2	3	4	5	6	7	8	9	10	11	γ
1	1	0	0	0	0	0	0	0	0.21	0.28	0.32	1
2	0	1	0	0	0	0	0	0	0.14	0.18	0.17	1
3	0	0	1	0	0	0	0	0	0.21	0.24	0.29	1
4	0	0	0	1	0	0	0	0	0.23	0.27	0.22	1
5	0	0	0	0	1	0	0	0	0.17	0.21	0.25	1
6	0	0	0	0	0	1	0	0	0.18	0.23	0.20	1
7	0	0	0	0	0	0	1	0	0.28	0.34	0.40	1
8	0	0	0	0	0	0	0	1	0.18	0.20	0.19	1

Начнем с введения ярлыка 1 в первую таблицу:

L_1	1	2	3	4	5	6	7	8	9	10	11	λ
9	0.00	0.01	0.01	0.02	0.00	0.01	0.01	0.02	0.31	0.00	-0.17	0.15
10	0.00	0.01	0.00	0.03	0.00	0.03	0.00	0.01	0.00	0.31	-0.29	0.02
1	0.31	0.23	0.27	0.23	0.27	0.23	0.31	0.23	0.00	0.00	1.00	1.00

$d_1 = 0.3110$. Был извлечен ярлык 11.

$L(u',v') = \{2, 3, 4, 5, 6, 7, 8, 9, 10, 11\}$.

L_2	1	2	3	4	5	6	7	8	9	10	11	γ
1	0.40	0.00	0.00	0.00	0.00	0.00	-0.32	0.00	-0.01	0.00	0.00	0.08
2	0.00	0.40	0.00	0.00	0.00	0.00	-0.17	0.00	0.01	0.01	0.00	0.23
3	0.00	0.00	0.40	0.00	0.00	0.00	-0.29	0.00	0.00	0.00	0.00	0.11
4	0.00	0.00	0.00	0.40	0.00	0.00	-0.22	0.00	0.03	0.03	0.00	0.18
5	0.00	0.00	0.00	0.00	0.40	0.00	-0.25	0.00	0.00	0.00	0.00	0.16
6	0.00	0.00	0.00	0.00	0.00	0.40	-0.20	0.00	0.02	0.03	0.00	0.21
11	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.28	0.34	0.40	1.00
8	0.00	0.00	0.00	0.00	0.00	0.00	-0.19	0.40	0.02	0.01	0.00	0.21

$d_2 = 0.4037$. Был извлечен ярлык 7.

$L(u',v') = \{2, 3, 4, 5, 6, 7, 8, 9, 10, 11\}$.

L_1	1	2	3	4	5	6	7	8	9	10	11	λ
9	-0.01	0.01	0.00	0.02	0.00	0.01	0.00	0.01	0.31	0.00	-0.18	0.13
10	0.00	0.01	0.00	0.03	0.00	0.03	0.00	0.01	0.00	0.31	-0.29	0.02
7	0.31	0.23	0.27	0.23	0.27	0.23	0.31	0.23	0.00	0.00	1.00	1.00

$d_1 = 0.3110$. Был извлечен ярлык 1.

$L(u',v') = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11\}$.

В результате работы алгоритма Лемке-Хоусона найдена равновесная по Нэшу ситуация - пара $(u_l^*, v_l^*) = ((0, 0, 0, 0, 0, 0, 1, 0), (0, 0, 1))$, которая соответствует выбору маршрутов (1, 4, 9, 12, 3) и (2, 5, 10,

11, 13, 3) источниками 1 и 2 соответственно, когда система находится в состоянии $l = (1, 1, 1, 1, 1, 0, 0, 1, 1, 1, 1, 1, 1)$.

12. В состоянии l значение характеристики $\varphi_l(u_l^*, v_l^*) = 1 \cdot 1 \cdot \varphi_l(x_7, y_3) = \max(LAETT_l(x_7, y_3), LAETT_l(y_3, x_7)) = \max(2.4771, 3.2156) = 3.2156$, где x_7 соответствует (1, 4, 9, 12, 3), y_3 соответствует (2, 5, 10, 11, 13, 3).

В данной системе существует еще 177 состояний, в которых алгоритм находит результаты.

13. P – вектор распределения вероятностей реализации каждого из допустимых состояний содержит 178 значений. Ожидаемое время завершения передач данных $\Phi = 3.754$.

Глава 3. Реализация алгоритма

3.1. Анализ и выбор инструментов реализации

Алгоритм строит математическую модель сети по исходным данным путем построения графа. В результате преобразований графа находятся все допустимые состояния системы, в каждом из которых определяются возможные маршруты. Из множества маршрутов игроки стараются выбрать те, в которых ожидаемое время передачи данных меньше. При возникновении конфликта интересов в сокращении времени передачи информации используется принцип равновесия по Нэшу для определения смешанной стратегии каждого игрока – вероятностного распределения выбора маршрутов. Подобная информация может быть использована для расчета различных численных характеристик модели, предполагая, что поведение игроков соответствует спрогнозированному.

Для получения результатов разработанной схемы, необходимого при решении поставленной задачи, требуется инструмент для реализации этого алгоритма. В качестве такого инструмента было выбрано вычислительное устройство (компьютер) с установленным специализированным программным обеспечением.

Для реализации алгоритма были доступны следующие пакеты программ:

- MATLAB
- GNU Octave
- Scilab
- Mathcad

MATLAB

Matlab – это кроссплатформенный пакет программ предназначенный для математических вычислений, моделирования различных процессов и программирования. Он ориентирован прежде всего на работу с матрицами, численные расчеты, визуализацию и создание приложений. Большая часть

функций написана на внутреннем языке и доступна для прочтения и модификации.

Пакет включает в себя множество модулей, которые являются совместимыми приложениями позволяющими добавлять к проектам специализированный функционал, такой как: проектирование и моделирование динамических систем (Simulink), использование модели конечных автоматов (Stateflow), работа с нейронными сетями и нечеткой логикой, оптимизация алгоритмов, а также взаимодействие с другими программными продуктами.

Matlab имеет богатые возможности по визуализации различного вида информации. Они включают в себя двухмерные и трехмерные графики, анимацию, а также интегрирование готовых моделей из САПР. [16]

GNU Octave

GNU Octave – система, предназначенная для математических вычислений, имеющая открытый исходный код. Она использует язык совместимый с Matlab. Функционал тоже практически соответствует этому программному продукту.

Octave предоставляет возможности для решения математических задач, моделирования, работы с вещественными и комплексными матрицами, решения дифференциальных уравнений, работы с линейными и нелинейными алгебраическими уравнениями, интегрирования функций. Также дополняется динамически загружаемыми модулями. [17]

Scilab

Scilab – пакет программ для различных расчетов. По функционалу во многом схож с Matlab. Имеет множество инструментов для работы с функциями, полиномами, матрицами, дифференциальными уравнениями, интегральными функциями. [18]

Совместим с Matlab, LabVIEW, C++ и Java. Scilab кроссплатформенный и имеет открытый исходный код.

Mathcad

Mathcad – программа для создания интерактивных документов с вычислениями и визуализацией. Математические функции задаются в графическом виде, а не как программный код, что делает их более удобными для восприятия. В Mathcad есть интеграция с другими программными продуктами базами данных. Возможности Mathcad расширяются подключением специализированных библиотек. [19]

В качестве среды для реализации алгоритма был выбран пакет MATLAB ввиду того, что в нем реализовано множество функций обработки матриц и представлены инструменты обработки больших объемов данных.

3.2. Реализация в среде MATLAB

Был реализован в формате функции $[u, v] = \text{Lemke_Howson}(A, B)$ рассмотренный в пункте 4.3. алгоритм Лемке-Хоусона поиска равновесия по Нэшу в смешанных стратегиях в биматричных играх. Функция принимает на вход две матрицы A и B , представляющие собой матрицы полезности для игроков 1 и 2 соответственно. Выходными значениями функции являются два вектора u и v - равновесные по Нэшу смешанные стратегии игроков 1 и 2, которые удовлетворяют (9), (10).

Составлена функция $[Z, I] = \text{DFS}(T, t, s)$, производящая поиск в графе определенном матрицей смежности T всех простых цепей z_i (выражаемых в форме векторов с последовательным перечислением номеров точек маршрута) с началом в точке с номером s и концом в вершине с номером t и соотнесенных с ними вспомогательных бинарных векторов $\text{id}x_i$. Эти векторы содержат true для $\text{id}x_{ij}$, если z_i содержит вершину j , false иначе. Перечисление цепей реализовано поиском в глубину. Входные данные – матрица T , целые числа t и s . Выход – массив ячеек $Z = \{z_i\}$ и логическая матрица $I = (\text{id}x_{ij})$.

Реализована функция $[\text{laett}] = \text{Laett}(ETX, RC, z)$ расчета численной характеристики $LAETT(z)$ (1) на основании входных данных: матрица $ETX = (ETX_{ij})$, вектор $RC = (RC_i)$, вектор маршрута z . Выход – число laett - значение $LAETT(z)$ (1).

В функции $[A_, B_] = \text{Laetts}(\text{ETX}, C, X, Y, I_x, I_y)$ производится расчет матриц $A = (a_{ij})$ и $B = (b_{ij})$ на основании входных данных: матрица ETX, вектор C, массив ячеек маршрутов первого игрока $X = \{x_i\}$ и второго игрока $Y = \{y_i\}$, соответствующие им вспомогательные логические матрицы $I_x = (\text{idx}_{x_{ik}})$, $I_y = (\text{idx}_{y_{jk}})$. Перечисляя все $i = 1, \dots, |X|$ и $j = 1, \dots, |Y|$ в цикле находятся векторы RC ($\text{RC}_k = C_k/2$, при $k: \text{idx}_{x_{ik}} = \text{true}$ и $\text{idx}_{y_{jk}} = \text{true}$, $\text{RC}_k = C_k$, иначе) (3) и $a_{ij} = \text{Laett}(\text{ETX}, \text{RC}, x_i)$, $b_{ij} = \text{Laett}(\text{ETX}, \text{RC}, y_j)$. Выход – матрицы $A_ = (a_{ij})$ и $B_ = (b_{ij})$.

Функция $[u, v, \text{value}] = \text{Step}(\text{ETX}, C, X_1, Y_1, I_{1x}, I_{1y})$ производит определение распределений u и v (выбора маршрутов из X_1 и Y_1 со вспомогательными логическими матрицами I_{1x} и I_{1y} соответственно определенными в некотором состоянии) являющихся равновесными по Нэшу смешанными стратегиями в игре определяемой матрицами выигрышей $A = 1./A_$ и $B = 1./B_$ (7), (8) (где $[A_, B_] = \text{Laetts}(\text{ETX}, C, X_1, Y_1, I_{1x}, I_{1y})$). Решение игры осуществляется функцией $[u, v] = \text{Lemke_Howson}(A, B)$. На основании данных стратегий можно определить $\text{value} = \max(u \cdot A \cdot v, u \cdot B \cdot v)$ (4) – ожидаемое время с начала пересылки до завершения всех передач данных. Входные параметры: матрица ETX, вектор C, массивы ячеек маршрутов X_1 и Y_1 , соответствующие им вспомогательные логические матрицы I_{1x} и I_{1y} . Выход функции – векторы u и v , скаляр value .

Функция $[Z_1, I_1] = \text{Ch}(Z, I, 1)$ предназначена для выбора множества маршрутов Z_1 в состоянии 1 из множества маршрутов $Z = \{z_i\}$ применяя соотнесенную с ним матрицу $I = \{\text{idx}_{ij}\}$ следующим образом: если множество $\{1_j \mid j: \text{idx}_{ij} = \text{true}\}$ состоит из элементов равных true (проверяется функцией $\text{all}()$), то в Z_1 добавляется маршрут z_i , в I_1 добавляется строка idx_i . Входные данные – массив ячеек $Z = \{z_i\}$, логическая матрица $I = (\text{idx}_{ij})$, логический вектор 1. На выходе функции – массив ячеек $Z_1 = \{z_{1i}\}$ и логическая матрица $I_1 = (\text{idx}_{1ij})$.

Функция $[\text{result}] = \text{job}(k, T, p, C)$ определяет матрицу $\text{ETX} = (\text{etx}_{ij})$ ($\text{etx}_{ij} = 0$, если $t_{ij} = 0$, $\text{etx}_{ij} = 1/t_{ij}$, иначе) (6) $T_ = (t_{ij})$ ($t_{ij} = 0$, если $t_{ij} = 0$, $t_{ij} = 1$,

иначе), находит $[X, I_x] = \text{DFS}(T, 1, 3)$, $[Y, I_y] = \text{DFS}(T, 2, 3)$. Затем, последовательно перечисляя все бинарные векторы вида $l = (1, 1, 1, l_4, \dots, l_k)$, находятся множества маршрутов в каждом состоянии $[X_1, I_{1x}] = \text{Ch}(X, I_x, l)$, $[Y_1, I_{1y}] = \text{Ch}(Y, I_y, l)$ и определяется допустимость состояния l (X_1 и Y_1 не пусты). Для допустимого состояния l функцией $[u_1, v_1, \text{values}_1] = \text{Step}(\text{ETX}, C, X_1, Y_1, I_{1x}, I_{1y})$ находятся равновесные по Нэшу распределения u_1 и v_1 вероятностей выбора маршрутов из X_1 и Y_1 , в P записывается значение соответствующее l ($P_l = \text{prod}(p(l)) \cdot \text{prod}(1-p(\sim l))$). После завершения всех итераций цикла вычисляется $P = P / \text{sum}(P)$ (5) и $\text{result} = P \cdot \text{values}$ (16) – искомая числовая характеристика. Входные параметры: число k узлов в сети, матрица T со свойствами таковой из пункта 1.1., p – вектор, C – вектор. Выход – скаляр result .

Выводы

В ходе работы была проведена разработка схемы поиска числовых характеристик беспроводной сети с применением аппарата теории игр для моделирования передачи данных как неантагонистической игры.

Данная схема поиска определяет равновесные по Нэшу смешанные стратегии выбора маршрута передачи данных из источника в сток и на их основании дает числовую характеристику сети. В работе характеристикой выступает значение ожидаемого времени завершения передач данных, но возможен поиск и других числовых характеристик сети определяемых на равновесных стратегиях. Представленная схема будет работать для таких характеристик похожим образом, и находить требуемые решения.

Работа схемы проиллюстрирована на примере решения задачи поиска указанной характеристики. При выполнении работы были реализованы функции в среде MATLAB для автоматизации расчетов необходимых при решении задачи.

Заключение

В процессе проведенной работы получены следующие результаты:

1. Составлено математическое описание задачи выбора маршрута передачи данных в беспроводных сетях, моделируемой в форме неантагонистической игры.

2. Реализован алгоритм теоретико-игрового моделирования передачи данных в беспроводных сетях для исполнения средствами пакета MATLAB.

3. Рассмотрен метод поиска числовых характеристик сети на основании моделирования равновесного по Нэшу выбора маршрутов передачи данных в системе.

Возможна дальнейшая разработка методов поиска характеристик для подобных систем. В качестве альтернативного направления исследования допустимо рассмотрение координации действий при выборе маршрутов.

Список цитируемой литературы

1. Пятибратов А. П., Гудыно Л. П., Кириченко А. А. Вычислительные системы, сети и телекоммуникации / Под ред. А. П. Пятибратова. М.: КНОРУС, 2013. С. 273-309.
2. Байчаров С. Выбор технологии беспроводного обмена данными для решения задач автоматизации систем жизнеобеспечения офисно-производственных помещений // Беспроводные технологии. Электрон. журн. 2007. № 2. URL: http://www.wireless-e.ru/articles/technologies/2007_2_58.php (дата обращения 20.04.2016).
3. Коломеец А. Е., Сурков Л. В. Моделирование сетей SDN в среде Nicira // Инженерный вестник МГТУ им. Н.Э. Баумана. Электрон. журн., 2014. № 6. URL: <http://engbul.bmstu.ru/doc/708106.html> (дата обращения 17.03.2016).
4. Aiache H., Conan V., Lebrun L., Rousseau S. A load dependent metric for balancing Internet traffic in Wireless Mesh Networks // 5th IEEE International Conference on Mobile Ad Hoc and Sensor Systems, 2008. P. 629-634.
5. Esposito P. M., Campista M. E. M., Moraes I. M., Costa L. H. M. K., Duarte O. C. M. B., Rubinstein M. G. Implementing the Expected Transmission Time Metric for OLSR Wireless Mesh Networks // 1st IFIP Wireless Days, 2008. P. 1-5.
6. Сурков Л. В., Коломеец А. Е. Автоматизированная программная система моделирования SDN-сетей // Инженерный вестник МГТУ им. Н.Э. Баумана. Электрон. журн., 2015. № 3. URL: <http://engbul.bmstu.ru/doc/708106.html> (дата обращения 17.03.2016).
7. Лемешко А. В., Вавенко Т. В. Разработка и исследование потоковой модели адаптивной маршрутизации в программно-конфигурируемых сетях с балансировкой нагрузки // Доклады ТУСУР, 2013. №3 (29) С. 100-108.
8. Датьев И. О. Маршрутные метрики многошаговых беспроводных самоорганизующихся сетей // Труды Кольского научного центра РАН,

2015. №3 (29) С. 115-136.
9. Дибров М. В. Маршрутизаторы. Красноярск: СФУ, 2008. 389 с.
 10. Blocq G., Orda A. How good is bargained routing? // INFOCOM, 2012 Proceedings IEEE, 2012. P. 2453-2461.
 11. Easley D., Kleinberg J. Networks, Crowds, and Markets: Reasoning about a Highly Connected World. Cambridge University Press, 2010. 833 p.
 12. Новожилова Л. М. Конечные графы и их приложения. СПб: Издательство СПбГУ, 2008. 234 с.
 13. Петросян Л. А., Зенкевич Н. А., Шевкопляс Е. В. Теория игр. 2-е изд., перераб. и доп. СПб.: БХВ-Петербург, 2012. С. 129-134.
 14. Бородакий Ю.В., Загребаев А.М., Крицына Н.А., Кулябичев Ю.П., Шумилов Ю.Ю. Нелинейное программирование в современных задачах оптимизации. М.: НИЯУ МИФИ, 2011. С. 105-125.
 15. V. von Stengel Equilibrium computation for two-player games in strategic and extensive form // Algorithmic Game Theory, Cambridge Univ. Press, Cambridge, 2007. P. 53-78.
 16. Иглин С. П. Теория вероятностей и математическая статистика на базе MATLAB. Харьков: НТУ "ХПИ", 2006. 612 с.
 17. Алексеев Е. Р., Чеснокова О. В. Введение в Octave для инженеров и математиков. М.: ALT Linux, 2012. 368 с.
 18. Тропин И. С., Михайлова О. И., Михайлов А. В. Численные и технические расчеты в среде Scilab. М.: 2008. 65 с.
 19. Очков. В.Ф. Mathcad 14 для студентов и инженеров: русская версия. СПб.: BHV, 2009. 512 с.