

**Институт проблем управления
им. В.А. Трапезникова РАН**

УПРАВЛЕНИЕ БОЛЬШИМИ СИСТЕМАМИ

Выпуск 46
Ноябрь 2013

**СБОРНИК
ТРУДОВ**

ISSN 1819-2467

Регистрационный номер Эл. №ФС77-44158 от 09 марта 2011 г.

Москва – 2013

РОССИЙСКАЯ АКАДЕМИЯ НАУК
*Институт проблем управления
им. В.А. Трапезникова*

**УПРАВЛЕНИЕ
БОЛЬШИМИ
СИСТЕМАМИ**

СБОРНИК ТРУДОВ

Выпуск 46

Москва – 2013

КООРДИНАЦИОННЫЙ СОВЕТ

Академики РАН: Васильев С.Н., Емельянов С.В., Коровин С.К., Куржанский А.Б., Федосов Е.А., Черноусько Ф.Л.; члены-корреспонденты РАН: Желтов С.Ю., Каляев И.А., Пархоменко П.П., Попков Ю.С.; д-ра техн. наук: Бутковский А.Г., Дорофеев А.А., Кузнецов О.П., Кульба В.В., Кротов В.Ф., Лотоцкий В.А., Павлов Б.В., Поляк Б.Т., Рутковский В.Ю.

РЕДАКЦИОННАЯ КОЛЛЕГИЯ

Главный редактор: член-корр. РАН Новиков Д.А. **Отв. секретарь:** к.т.н. Губко М.В.
Д-ра техн. наук: проф. Алескерев Ф.Т. (ГУ ВШЭ), проф. Артамонов Е.И. (ИПУ РАН), д-р экон. наук, проф. Архипова М.Ю. (ИПИ РАН), д-ра техн. наук: проф. Афанасьев В.Н. (МИЭМ), проф. Бахтадзе Н.Н. (ИПУ РАН), проф. Бурков В.Н. (ИПУ РАН), проф. Вишневский В.М. (ИППИ РАН), д-р экон. наук, проф. Голиченко О.Г. (ЦЭМИ РАН), д-р физ.-мат. наук, проф. Добровидов А.В. (ИПУ РАН), д-ра техн. наук: проф. Заложнев А.Ю. (ИПУ РАН), проф. Ириков В.А. (МФТИ), проф. Калянов Г.Н. (ИПУ РАН), проф. Касаткин С.И. (ИПУ РАН), проф. Каравай М.Ф. (ИПУ РАН), канд. техн. наук Квинто Я.И. (ИПУ РАН), д-р экон. наук, проф. Клочков В.В. (ИПУ РАН), д-ра техн. наук: проф. Кононенко А.Ф. (ВЦ РАН), канд. техн. наук Куливец С.Г. (ИПУ РАН), проф. Курдюков А.П. (ИПУ РАН), проф. Лебедев В.Г. (ИПУ РАН), к-т техн. наук, доцент Лебедев В.Н. (ИПУ РАН), д-р экон. наук, проф. Ловчиновский Э.В. (ИПУ РАН), д-р техн. наук, проф. Мандель А.С. (ИПУ РАН), д-р экон. наук, проф. Нижегородцев Р.М. (ИПУ РАН), д-ра техн. наук: проф. Новосельцев В.Н. (ИПУ РАН), проф. Орлов А.И. (МВТУ), д-р физ.-мат. наук, проф. Рапопорт Л.Б. (ИПУ РАН), д-р техн. наук, проф. Рыков А.С. (МИСИС), д-р экон. наук, проф. Секерин В.Д. (ИПУ РАН), д-ра техн. наук: проф. Сидельников Ю.В. (МАИ), проф. Совлуков А.С. (ИПУ РАН), д-р экон. наук, проф. Сухарев О.С. (Ин-т экономики РАН), д-ра техн. наук: проф. Уткин В.А. (ИПУ РАН), проф. Хоботов Е.Н. (МВТУ), д-ра физ.-мат. наук: доцент Чеботарев П.Ю. (ИПУ РАН), проф. Чхартишвили А.Г. (ИПУ РАН), проф. Щербаков П.С. (ИПУ РАН).

РЕГИОНАЛЬНЫЕ РЕДАКЦИОННЫЕ СОВЕТЫ

Волгоград – д-ра физ.-мат. наук: проф. Воронин А.А., проф. Лосев А.Г. (ВолГУ); **Воронеж** – д-р техн. наук, проф. Баркалов С.А., д-р физ.-мат. наук, проф. Головинский П.А. (ВГАСУ), д-р техн. наук, проф. Подвальный С.Л. (ВГТУ); **Ижевск** – д-р физ.-мат. наук, проф. Непейвода Н.Н., к-т физ.-мат. наук, проф. Родионов В.И. (УдмГУ); **Иркутск** – д-ра физ.-мат. наук: проф. Бычков И.В., проф. Лакеев А.В. (ИДСТУ СО РАН); **Казань** – д-р физ.-мат. наук, проф. Маликов А.И., д-р техн. наук, проф. Сиразетдинов Р.Т. (КГТУ-КАИ); **Липецк** – д-ра техн. наук: проф. Кузнецов Л.А., проф. Погодаев А.К. (ЛГТУ); **Самара** – д-ра экон. наук: проф. Богатырев В.Д., проф. Гераскин М.И., д-р техн. наук, проф. Засканов В.Г. (СГАУ); **Санкт-Петербург** – д-ра физ.-мат. наук: проф. Петросян Л.А. (СПбГУ), проф. Фрадков А.Л. (ИПМ РАН); **Старый Оскол** – д-р техн. наук, проф. Еременко Ю.И. (СТИ); **Тверь** – д-ра техн. наук: проф. Кузнецов В.Н., проф. Палюх Б.В. (ТГТУ).

Адрес редакции: 117997, г. Москва, ул. Профсоюзная, д. 65.

Адрес в Интернет: ubs.mtas.ru.

Номер гос. регистрации электронного научного издания (ЭНИ): 0421200023.

© ИПУ РАН, 2013

СОДЕРЖАНИЕ

Системный анализ

Бабичев А. В. <i>О согласовании поведения взаимодействующих объектов</i>	6
Зоркальцев В. И., Медвежонков Д. С. <i>Численные эксперименты с вариантами алгоритмов внутренних точек на нелинейных задачах потокораспределения</i>	68
Орлов А. И. <i>О средних величинах</i>	88

Информационные технологии в управлении

Чесноков А. М. <i>Интеллектуальные системы на основе колонок</i>	118
--	-----

Сетевые модели в управлении

Жучков Р. Н. <i>Применение идей прогнозирующего управления в синтезе стабилизирующего управления сетевыми объектами</i>	147
---	-----

Управление в социально-экономических системах

Ахохов А. А., Блудян Н. О., Дорофеюк Ю. А., Чернявский А. Л. <i>Независимая многовариантная экспертиза в задаче управления межрегиональными автобусными перевозками</i>	163
Дьяченко В. К., Тарасенко Л. В., Угольницкий Г. А. <i>Имитационное моделирование социального партнерства в системе дополнительного профессионального образования</i>	179
Колоденкова А. Е. <i>Мягкие вычисления как путь к преодолению не-факторов в задачах оценки жизнеспособности проектов</i>	197

Коргин Н. А., Корепанов В. О. <i>Решение задачи эффективного распределения ресурсов на основе механизма Гровса-Лейдярда при трансферабельной полезности.....</i>	216
Семёнов А. А., Кочемазов С. Е. <i>О дискретно-автоматных моделях конформного поведения</i>	266
Управление техническими системами и технологическими процессами	
<hr/>	
Май Н. Т., Ха В. М., Камаев В. А., Щербаков М. В., Тхай Куанг Винь <i>Моделирование и оптимизация управления интеллектуальной гибридной энергосистемной с источниками возобновляемой энергии.....</i>	293

УДК 519.179.2 + 519.713.8 + 512.579
ББК 22.12. + 22.19

О СОГЛАСОВАНИИ ПОВЕДЕНИЯ ВЗАИМОДЕЙСТВУЮЩИХ ОБЪЕКТОВ

Бабичев А. В.¹

(ФГБУН Институт проблем управления
им. В.А. Трапезникова РАН, Москва)

Статья посвящается задаче согласования поведения множества взаимодействующих объектов. Рассматривается процедура согласования, которая на каждом шаге настраивает один объект и переводит его в состояние, согласованное с состоянием его соседей. При этом может оказаться, что некоторые ранее настроенные объекты перестанут быть согласованными. Исследуются условия, при которых процесс стабилизируется и всё множество взаимодействующих объектов может быть согласовано. Описываются свойства, которыми должны обладать объекты для того, чтобы любое их соединение в сеть было настраиваемым, а также приводится алгоритм настройки для любой такой сети.

Ключевые слова: функциональная сеть, решение системы уравнений, универсальная алгебра, метод итераций.

1. Введение

Постановка. На интуитивном уровне задачу, решаемую в настоящей работе, можно поставить следующим образом.

Имеется система взаимодействующих объектов. Эта система функционирует правильно только в том случае, когда состояние каждого объекта (параметры поведения объекта) «согласовано» с состояниями соседних с ним объектов.

¹ Андрей Владимирович Бабичев, кандидат технических наук, (babichev@ipu.ru).

То, что состояние объекта «согласовано» с состояниями соседей, пока никак не определяется. Предполагается только, что имеется процедура, которая настраивает произвольный объект. Эта процедура переводит заданный объект в состояние, согласованное с состояниями соседей.

Требуется настроить всю систему, при этом шагом настройки является настройка одного объекта (перевод его в состояние согласованное с соседями). Понятно, что при настройке одного элемента мы можем потерять настройки других, поэтому в общем случае нет гарантии, что процесс настройки всей системы может быть успешно завершен. Таким образом, при рассмотрении процесса настройки сразу возникает две задачи: первая – описать класс настраиваемых сетей (сетей, которые могут быть настроены), вторая – описать процедуру настройки (последовательность, в которой следует настраивать элементы сети).

Источники постановки. Исходная постановка возникла в связи с проектом Максима Хомякова «Хаос» (начало 80-х годов [19]). Проект «Хаос» был посвящен созданию базы данных, и в этом проекте был заложен аппарат, хотя и более продвинутый концептуально, но сходный с аппаратом триггеров в современных СУБД. Основной принцип использования триггеров можно описать следующим образом. Каждый триггер можно рассматривать как программу корректировки данных для того, чтобы обеспечить некоторые ограничения целостности. При внесении в базу каких-либо изменений могут нарушиться ограничения целостности – вызываются соответствующие триггеры, корректирующие данные. Применение триггеров может изменить другие ограничения целостности – вызовутся другие триггеры и так далее пока не будет достигнуто согласованное состояние – состояние, в котором все ограничения целостности выполнены. В подобной интерпретации (в терминах триггеров) рассматриваемая задача согласования выглядит следующим образом: как надо организовывать последовательность вызовов триггеров и какие ограничения на вид триггеров надо наложить, чтобы не заикнуться на стадии исполнения транзакции?

Работа была завершена при исследовании задач согласования управленческих решений (смотри работы [14, 15]), проблематику которых демонстрируют приводимый разделе 2 пример.

Используемый аппарат и используемые результаты.

В качестве формальной модели для систем взаимодействующих объектов используются диагональные системы уравнений (системы уравнений вида (*)): каждому объекту взаимно однозначно соответствует переменная, а значение каждой переменной равно текущему состоянию соответствующего объекта. Соответственно, настроенные состояния всей системы взаимодействующих объектов – это решения системы уравнений.

$$(*) \begin{cases} x_1 = t_1(x_1, \dots, x_n) \\ \dots \\ x_n = t_n(x_1, \dots, x_n) \end{cases}$$

Процесс настройки это процесс решения системы уравнений модифицированным методом итераций:

если в классическом методе итераций значения всех переменных одновременно заменяются значениями правых частей соответствующих им уравнений, то в модифицированном методе на каждом шаге вычисляется ровно одна переменная.

Используемый в работе модифицированный метод итераций берёт начало от методики упорядочивания переменных, используемой Бухбергером [2, 4, 8, 9] (1965 г.) при делении многочленов от многих переменных, эту же методику позже использовал Блэнд [16].

В работах [5, 6, 7, 17, 18] исследуется формализация процесса согласования, с одной стороны, более подробная, а с другой, более общая, нежели рассматриваемая в настоящей статье. Более того, тематика этих работ пересекается не только с тематикой статьи, но и с тематикой проекта «Хаос».

В связи с задачей обобщения публикуемых результатов, следует отметить, что классические теоремы о неподвижной точке к рассматриваемым задачам неприменимы (это проявляется в том, что применение классического метода итераций приводит к зацикливанию). Исключением является теорема Брауэра о непре-

рывном отображении шара в себя. Более того, в работах А.В. Малишевского («натуральные системы») [13], которые по тематике пересекаются как с темой настоящей работы, так и с вопросами обобщения теоремы Брауэра, продемонстрировано, что эта теорема при обосновании методов согласования играет ключевую роль.

Принцип изложения. По возможности все формальные определения и утверждения вынесены в приложение, в основном тексте оставлена лишь та часть формального аппарата, без которой нельзя сформулировать главный результат (теорема 1). Существует взаимно однозначное соответствие между диагональными системами уравнений и функциональными сетями (размеченными ориентированными графами с некоторыми дополнительными ограничениями). Поэтому в основном тексте упор делается на более наглядное представление с помощью функциональных сетей.

План изложения. В разделе 2 приводится игрушечный пример взаимодействующих объектов, на котором будут иллюстрироваться все вводимые в основном тексте определения.

В разделе 3 в общих терминах описывается стратегия настройки.

Соответствующие определения и формальное описание алгоритм настройки в терминах функциональных сетей приведены в разделе 4.

В разделе 5 формулируются условия, характеризующие область применимости предлагаемого метода.

В приложение вынесены: описание аппарата, используемого при обосновании метода настройки (приложение 1), и само обоснование (приложения 2, 3).

2. Задача о вечеринке

Чтобы описать проблематику на интуитивном уровне, рассмотрим следующий пример.

Пример 1. (Задача о вечеринке). Представим ситуацию, что организатор вечеринки собирается пригласить в гости троих зна-

комых x , y и z (участники вечеринки) в один из ближайших выходных дней (суббота либо воскресенье): и тот и другой день организатора полностью устраивает. Каждый из знакомых организатора (каждый участник вечеринки) наверняка может придти к нему либо в субботу, либо в воскресенье. Если все не могут собраться (например, x занят в субботу, а y в воскресенье), то встреча может состояться в сокращенном составе (в тот день, когда может z), но лучше все же собраться вчетвером.

У каждого из участников есть наиболее удобный день, когда он может придти в гости, но он, может быть, согласится придти в другой день. Например, если y во что бы то ни стало хочет встретиться с x (только ради этого он и придет на встречу), то y согласится на любой день, который удобен для x . С другой стороны, если x не хочет встречаться с y (но желает встретиться с организатором), то x будет действовать по следующему правилу: x предложит собраться в тот день, который не выбрал y .

При этом организатору не известны мотивы, руководствуясь которыми участники выбирают тот или иной день встречи.

Организатор по очереди договаривается с каждым участником встречи. При каждом разговоре он, во-первых, сообщает собеседнику все известные ему пожелания, а во-вторых, выслушивает пожелание собеседника.

Поскольку мнение каждого участника вечеринки может меняться в зависимости от мнений остальных участников, то, возможно, придется по нескольку раз разговаривать с каждым. Эти переговоры надо провести таким образом, чтобы каждый, кто придет на встречу, остался бы доволен: если y хочет встретиться с x , а x не придет на встречу, то y будет считать вечер испорченным;

Каждый участник, не попавший на вечеринку, должен быть извещен о том, какой день встречи назначен (вдруг он передумает).

Необходимо сделать конечное число звонков и либо назначить день встречи, либо отменить встречу. Каким образом следует проводить переговоры (в каком порядке обзванивать

участников)?•

Чтобы продемонстрировать нетривиальность рассмотренной постановки, опишем, какие проблемы могут возникнуть при проведении переговоров.

Пример 2. На рис. 1 представлены иллюстрации (пока что это неформализованные картинки) возможных взаимоотношений участников вечеринки (следует помнить, что организатор ничего об этих взаимоотношениях не знает):

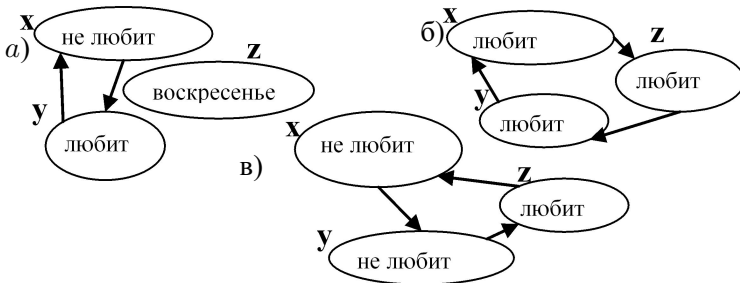


Рис. 1. Варианты взаимоотношений участников

а) мнение z не зависит от мнений x и y и это мнение не меняется в процессе переговоров (он/она по субботам никуда не ездит). Участник x ни в коем случае не хочет быть в одной компании с y (ей удобнее всего день, не удобный для y), y придет в любой день, в который придет x , а без x не придет;

б) участник x придет на встречу только вместе с z , z только с y , а y только вместе с x ;

в) участник y не хочет встречаться с z , x не придет, если будет y , а z придет, только если будет x .

Предположим, что

- при проведении переговоров организатор многократно обзванивает участников по циклу z, y, x , начиная с z (как на рис. 2.и)), надеясь, что предпочтения участников рано или поздно перестанут меняться;

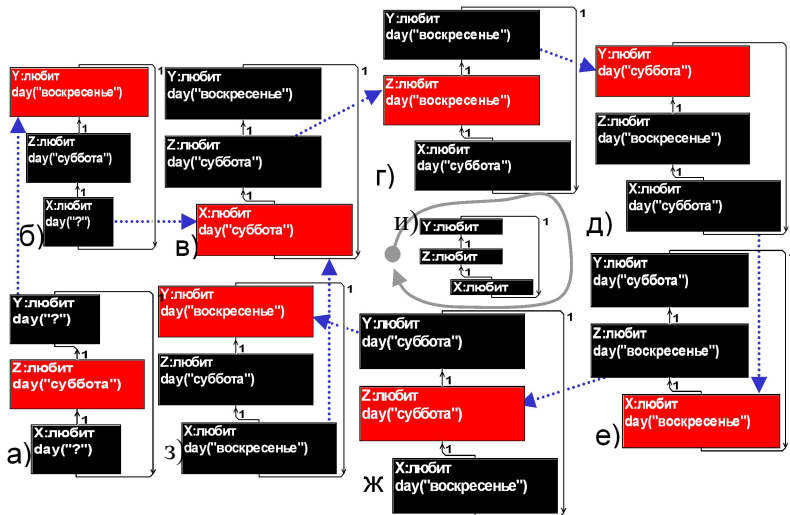


Рис. 2. Пример неправильной стратегии настройки

- до начала переговоров участнику z удобна суббота, а остальным – воскресенье (рис. 2.3);
- взаимоотношения участников (которые не известны организатору) соответствуют диаграмме, представленной на рис. 1.б): все друг друга любят и поэтому, если правильно провести переговоры, все соберутся и будут довольны

Но, как видно из рис. 2, подобная стратегия переговоров обречена на неудачу: при подобной стратегии переговоров процесс настройки не закончится никогда. •

3. Стратегия Киссинджера

Далее описывается техника настройки множества взаимодействующих объектов.

Эту технику, по-видимому, первым строго описал Б. Бухбергер (1965) при описании алгоритма деления многочленов со многими переменными [2, 4, 8, 9]. После Б. Бухбергера этот же алгоритм использовал Р. Блэнд (1977) для поиска базисных решений в задачах линейного программирования [16]. В дальнейшем эта

техника будет проходить под названием стратегия Киссинджера (Г. Киссинджер использовал ее в 1973 г. при проведении трехсторонних переговоров после войны Судного дня).

На содержательном уровне стратегию Бухбергера – Киссинджера – Блэнда можно рассматривать как следующую стратегию согласования. Пусть все объекты перенумерованы. Тогда на каждом шаге должен настраиваться тот ненастроенный объект (ровно один), который имеет наименьший номер.

Определение 1. (*Стратегия Киссинджера*). Пусть \ll – некоторый линейный порядок на конечном множестве взаимодействующих объектов V . Главным объектом из множества V будем называть минимальный по порядку \ll объект, который не является настроенным (главный объект существует тогда и только тогда, когда существуют ненастроенные объекты). Настройкой в соответствии со \ll – стратегией Киссинджера называется последовательность настроек главных объектов.

Таким образом, при настройке по стратегии Киссинджера на каждом шаге настраивается тот ненастроенный объект, который имеет наименьший номер при нумерации по порядку \ll .

Пример 3. Рассмотрим, как стратегия Киссинджера выглядит в применении к задаче о вечеринке.

Прежде всего следует отметить, что в задаче о вечеринке есть процедура, настраивающая произвольный объект – это звонок организатора участнику вечеринки. В то же время нет другого способа определить, в настроенном состоянии находится объект или нет, кроме как настроив его: лишь после настройки (после разговора) организатор вечеринки точно знает, в каком состоянии (в настроенном или нет) находился объект до настройки.

Таким образом, идентифицировать главный объект (и заодно настроить его) можно только обзвонив всех участников по порядку \ll .

Итак, в применении к задаче о вечеринке стратегия Киссинджера выглядит следующим образом:

Назовем звонок информационным, если он каким-нибудь образом меняет представление организатора о предпочтениях участников. Например, самый первый звонок к кому-нибудь всегда является информационным (до звонка организатор ничего не знает о желаниях собеседника), повторный звонок является информационным только в том случае, если пожелание собеседника меняется по сравнению с пожеланиями на предыдущем звонке. Другими словами повторный звонок к объекту o является информационным, если до звонка объект o находился в ненастроенном состоянии.

Алгоритм настройки выглядит следующим образом.

1) Зафиксировать некоторый порядок \ll на элементах множества $\{x, y, z\}$ например $z \ll y \ll x$ (как в примере 2 и на рис. 2.и)).

2) Обзванивать участников в соответствии с порядком \ll , при этом если очередной звонок является информационным, то прервать цикл и начать процесс переговоров заново (начиная с z).

3) Процесс настройки заканчивается, когда

а) либо все последние разговоры со всеми участниками оказались неинформационными, – в этом случае можно назначить день встречи (день который выбрало большинство участников),

б) либо после очередного звонка получены ровно те пожелания, которые были на предыдущих шагах (конфликт интересов, который проявляется как заикливание) – следует отменить встречу, в противном случае среди участников будут обиженные на организатора.

На рис. 3 приведены примеры использования описанного алгоритма при упорядочивании $z \ll y \ll x$ и при различных взаимоотношениях между участниками вечеринки, перечисленных на рис. 1.

Следует подчеркнуть, что рассмотренная постановка – это лишь примитивный пример: каждый объект в этой задаче связан только с одним соседом, в общем же случае связи между объек-

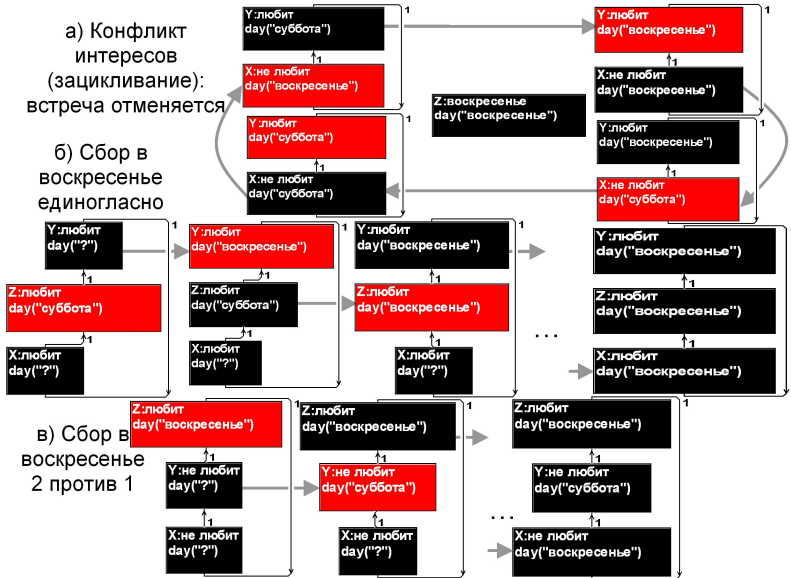


Рис. 3. Примеры использования стратегии Киссинджера

тами могут быть гораздо сложнее. Например, в состав участников вечеринки можно ввести «большевика» (и/или «меньшевика»), этот участник будет подсчитывать сторонников субботы и воскресенья и примыкать к большинству (к меньшинству). Замечательно то, что подобные изменения никак не влияют на работоспособность стратегии Киссинджера. •

На рис. 4 представлена иллюстрация работы программы осуществляющей настройку множества взаимодействующих объектов в соответствии со стратегией Киссинджера: в блоках осуществляется настройка i -го по порядку << объекта и сравнение состояния, полученного после настройки объекта, со старым состоянием объекта (с состоянием, в котором объект находился до шага настройки).

Настоящая работа посвящена исследованию множества ограничений, при которых, действуя по стратегии Киссинджера, гарантированно можно найти согласованное решение.

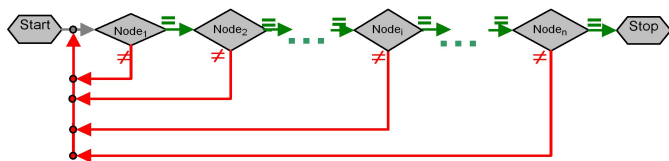


Рис. 4. Настройка по стратегии Киссинджера

4. Функциональные сети

Если для описания алгоритма настройки не требуется никакого аппарата, то для описания области применимости этого алгоритма необходимо уточнить, что понимается под терминами «сеть взаимодействующих объектов», «настроенный объект» и «настройка объекта». Уточнению перечисленных терминов и посвящен настоящий раздел.

В настоящей работе термин «сеть взаимодействующих объектов» отождествляется с термином «функциональная сеть» (формальное определение приводится в приложении: определение /33/²). Единственное, что требуется из этого определения для понимания дальнейшего, это следующее:

- функциональная сеть состоит из функциональных блоков, соединенных между собой (если N — функциональная сеть, то $N.nodes$ — множество блоков присутствующих в ней);
- каждый блок имеет один выход (сигнал, порождаемый функциональным блоком) и какое-то множество (может быть, пустое) пронумерованных входов.

Кроме того, каждому блоку $x \in N.nodes$ соответствует название операции, которую он должен выполнять (это название операции обозначается $N.functor(x)$).

² В приложении приведен список используемых в работе понятий. Поскольку нумерация определений сквозная, то в основном тексте иногда будут встречаться ссылки «вперед»: ссылки на определения из этого списка (первое определение в приложении имеет номер 8).

- при соединении блоков между собой каждый вход каждого используемого блока должен быть подсоединен к выходу какого-нибудь блока (при этом не ограничено число входов, подсоединенных к заданному выходу). При этом блок, к выходу которого подсоединен вход с номером i блока x , обозначается $x[i]$ (определение /34/);
- в соответствии с определением /33/ функциональной сети вместо термина «функциональный блок сети» в дальнейшем будет использоваться термин «вершина сети»
- с понятием «функциональная сеть» тесно связано понятие «диагональная система уравнений»:

Каждой вершине $x \in N.nodes$ функциональной сети N соответствует уравнение вида $x = f(x[1], \dots, x[n])$, где $f = N.functor(x)$ и n — число входов у вершины x . Соответственно, самой сети соответствует система уравнений (характеристическая система уравнений для функциональной сети N обозначается $system(N)$, определение /35/), которая однозначно описывает исходную функциональную сеть.

Пример 4. На рис. 1 приведены примеры функциональных сетей. Стрелки направлены от входа каждой вершины к точке подсоединения этого входа (таким образом, стрелки ориентированы против направления распространения сигналов). Имена операций принимают значения из множества {“любит”, “не любит”, “воскресенье”}. Например, для сети N , представленной на рис. 1а), $N.nodes = \{x, y, z\}$, при этом $N.functor(x) = \text{“не любит”}$, $N.functor(y) = \text{“любит”}$, $N.functor(z) = \text{“воскресенье”}$, $x[1] = y$ и $y[1] = x$.

Для функциональных сетей, представленных на рис. 1, характеристические системы уравнений имеют вид:

$$a) \begin{cases} x = \text{“не любит”}(y), \\ y = \text{“любит”}(x), \\ z = \text{“воскресенье”}; \end{cases} \quad б) \begin{cases} x = \text{“любит”}(z), \\ z = \text{“любит”}(y), \\ y = \text{“любит”}(x); \end{cases} \quad в) \begin{cases} x = \text{“не любит”}(y), \\ y = \text{“не любит”}(z). \end{cases} \bullet$$

Теперь, когда определено понятие «сеть взаимодействующих объектов», можно перейти к определению понятий «настройка объекта», «настроенный объект» и «настроенная сеть».

Для того чтобы определить эти понятия, воспользуемся классическим понятием универсальной алгебры (определение /25/). На содержательном уровне универсальная алгебра (в дальнейшем, поскольку других алгебр в работе не будет, будем пользоваться коротким термином алгебра) – это всего лишь множество не соединенных друг с другом функциональных блоков разного типа: тип блока (или марка изделия, записанная на корпусе) однозначно определяет количество входов у блока и правила его работы. Для того чтобы описать алгебру (или, говоря проще, чтобы описать имеющееся сырьё для производства функциональных сетей), необходимо задать:

- возможный диапазон входных и выходных сигналов имеющихся блоков (это множество всех возможных входных и выходных сигналов называется множеством элементов алгебры);
- множество имеющихся типов блоков (это множество называется множеством функциональных символов алгебры и именно из этого множества выбираются имена операций приписанных вершинам сети $N.functor$);
- правила функционирования блоков каждого типа (эти правила, которые каждому типу блока сопоставляют реализуемую блоком функцию, называются интерпретацией множества функциональных символов. При этом блок, не имеющий входов, является источником постоянного сигнала, а значение выхода у блока с входами функционально зависит от значений сигналов поступающих на его входы).

Таким образом, любая алгебра A задается тройкой (Q, F, μ) , где Q – множество элементов алгебры (диапазон сигналов); F – множество функциональных символов (типы блоков) и μ – интерпретация (правила функционирования блоков). При этом если Q – конечное множество, то A называется конечной алгеброй.

Для выражения того факта, что сеть построена из блоков задаваемых алгеброй A , используется термин сеть [представленная] над алгеброй A (определение /39/).

Пример 5. Рассмотрим функциональные сети, представленные

ные на рис. 1, и опишем алгебру (назовем ее A_{party}), над которой представлены эти схемы.

Эти схемы состоят из следующих типов блоков: “любит” (с одним входом), “не любит” (с одним входом) и “воскресенье” (без входов). В соответствии с условием задачи о вечеринке (примеры 1, 2) множество входных и выходных сигналов блоков принадлежит множеству {суббота, воскресенье}.

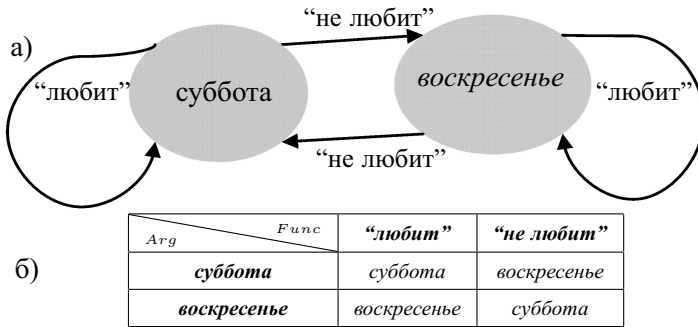


Рис. 5. Алгебра A_{party}

Итак, $A_{party} = (\{суббота, воскресенье\}, \{“любит”, “не любит”, “воскресенье”\}, \mu)$, где $\mu(“воскресенье”) = воскресенье$ (константа), а интерпретация функциональных символов “любит” и “не любит” задана на рис. 5, с помощью диаграммы (рис. 5а) и таблицы (рис. 5б) переходов конечного автомата.

Содержательно: если y “любит” x , то y назначает тот же день встречи, что и x , а, если x “не любит” y , то x назначает день, отличный от дня, назначенного y .•

Определение 2. Пусть N – функциональная сеть (/33/) над алгеброй (/39/) $\overline{A} = (Q, F, \mu)$ (/25/) и V – множество вершин сети N ($V = N.nodes$ /33/). Тогда множеством состояний сети N над A (обозначается $States(A, N)$) называется множество всех отображений из V в Q .

Содержательно функциональная сеть с множеством вершин

V находится в состоянии $p \in States(A, N)$ тогда и только тогда, когда для любой вершины $x \in V$ значение сигнала на выходе x равно $p(x)$.

Следующее определение уточняет, что такое «шаг настройки объекта x » (это переход к новому состоянию, в котором выходной сигнал блока x равен значению функции реализуемой блоком при текущих значениях входных сигналов)

Определение 3. (в /40/ приведено более строгое определение.) Пусть N функциональная сеть над алгеброй A (/39/). Для каждой вершины $x \in N.nodes$ функциональной сети N определим отображение $\langle_{step}^{node} \rangle_x : States(A, N) \rightarrow States(A, N)$ (/2/) следующим образом. Пусть вершина x имеет n входов и пусть f – функция, реализуемая этой вершиной (f – это функция соответствующая функциональному символу $N.functor(x)$ в алгебре A). Тогда для любого состояния $p \in States(A, N)$ $\langle_{step}^{node} \rangle_x(p)$ – это такое состояние $q \in States(A, N)$, что для любой вершины $y \in V$

- $q(y) = p(y)$, если $x \neq y$ и
- $q(x) = f(p(x[1]), \dots, p(x[n]))$ (в соответствии с определением /34/ $x[i]$ – это вершина, порождающая сигнал для i -го входа x).

Замечание . Фактически в определении /3/ описано то, как меняются значения переменных на одном шаге метода итераций (метод итераций применяется для поиска решения характеристической системы уравнений сети), в том случае, когда шаг итерации применяется не ко всем вершинам, а ровно к одной.

Определение 4. Пусть N – функциональная сеть (/33/) над алгеброй A (/39/), V – множество вершин схемы N ($V = N.nodes$ (/33/)) и $p \in States(A, N)$ (/2/). Говорят, что в состоянии p вершина $x \in V$ настроена, если $\langle_{step}^{node} \rangle_x(p) = p$. Состояние p называется настроенным (или согласованным), если в нем все вершины множества V настроены.

Замечание . Равенство $\langle \text{node} \rangle_x(p) = p$ означает, что в состоянии p уравнение вершины $x = f(x[1], \dots, x[n])$ является равенством в алгебре A : что $p(x) = f(p(x[1]), \dots, p(x[n]))$ (здесь f – функция, реализуемая вершиной x).

Таким образом, в процессе настройки функциональной сети N ищется решение характеристической для N системы уравнений (утв. 16.следствие 1),

Пример 6. Любому согласованному в терминах задачи о вечеринке (примеры 1, 2, 3) состоянию какой-нибудь из сетей, представленных на рис. 1, соответствуют такие значения выходных сигналов у вершин x, y, z , которые являются решением соответствующей характеристической системы уравнений в алгебре A_{party} (пример 5). Наоборот: любому решению характеристической системы уравнений (определение /31/) соответствует настроенное состояние сети.●

Определение 5. (В /42/ приведено более общее определение). Пусть N – функциональная сеть над алгеброй A (/39/) и V множество ее вершин. Последовательность $\pi : \{1, 2, \dots\} \rightarrow States(A, N)$ (/2/) (конечная либо бесконечная) состояний схемы называется настроенной последовательностью, если для каждого не последнего члена последовательности с номером i существует такая вершина $x \in V$, что $\pi(i + 1) = \langle \text{node} \rangle_x(\pi(i))$ (/3/).

Настроенная последовательность называется избыточной, если в ней присутствует более одного согласованного состояния.

Замечания. 1. Каждый элемент настроенной последовательности описывает значения переменных на соответствующем шаге итерационного процесса решения характеристической системы уравнений модифицированным методом итераций.

2. Понятие избыточной последовательности вводится как аналог сходящейся последовательности (любую конечную избыточную настроенную последовательность можно продолжить до бесконечной сходящейся последовательности).

Следующее определение является переформулировкой определения стратегии Киссинджера в терминах состояний функциональных схем.

Определение 6. (Более корректно это определение сформулировано в /46/). Пусть N – функциональная сеть над алгеброй A (/39/) и \ll линейный порядок на множестве V вершин схемы N . Для любого ненастроенного состояния $q \in States(A, N)$ (/2/) главной ненастроенной вершиной (обозначается $\langle_{node}^{chief} \rangle(\ll, q)$) будем называть наименьшую по порядку \ll ненастроенную вершину (/4/) из множества V .

Пусть $\pi : \{1, 2, \dots\} \rightarrow States(A, N)$ – настроечная последовательность (/5/) состояний схемы N (/2/). Последовательность π называется \ll последовательностью Киссинджера для схемы N , если для любого не первого члена этой последовательности с номером i выполняется, либо

$$\pi(i) = \langle_{step}^{node} \rangle \langle_{node}^{chief} \rangle(\ll, \pi(i-1))(\pi(i-1)) (/3/),$$

либо $\pi(i) = \pi(i-1)$ и при этом $\pi(i-1)$ – настроенное состояние (/4/).

Замечание . Таким образом, стратегия Киссинджера соответствует модифицированному методу итераций: на каждом шаге итерации вычисляется новое значение ровно одной переменной (той, которая является наименьшей из всех не удовлетворяющих соответствующему равенству переменных).

5. Комбинаторные алгебры

В этом разделе обсуждается следующий вопрос: «В каких случаях гарантирована сходимости последовательности Киссинджера?»

О том, что метод итераций, основанный на стратегии Киссинджера, не слабее обычного метода итераций, позволяет судить хотя бы тот факт, что всякий раз, когда гарантирована сходимости метода итераций (например, все отображения сжимающие либо

все отображения ограничены и монотонны) гарантирована также сходимость и модифицированного метода.

В связи с дальнейшим изложением следует подчеркнуть, что все доказанные результаты касаются лишь очень частного случая: случая, когда сеть рассматривается над конечной алгеброй (в этом случае сходимость эквивалентна тому, что последовательность стабилизируется через конечное число шагов). Для рассмотрения общего случая требуется существенно более сложный аппарат (в частности, требуется понятие точности настройки).

Приводимое далее определение «регулярно устойчивых алгебр» вводится только для сокращения терминологии (для краткого обозначения тех алгебр, на которых стратегия Киссинджера работает). Этот термин нужен только для формулировки теоремы 1 и нигде больше он использоваться не будет. Уже в определении б, чтобы очертить ту область, где есть строгие результаты, термин «сходящаяся последовательность» был заменен на термин «избыточная последовательность».

Определение 7. *(пояснения по поводу использования ссылок вперёд приведены в сноске ² на странице 16) Конечная алгебра A (/25/) называется регулярно устойчивой, если для любой конечной (с конечным множеством вершин) схемы N над A (/39/) и любого линейного порядка \ll на ее вершинах любая бесконечная \ll последовательность Киссинджера (/46/) является избыточной (/42/).*

Конечная алгебра A (/25/) называется устойчивой, если для любой конечной (с конечным множеством вершин) схемы N над A (/39/) существует такой линейный порядок \ll на ее вершинах, что любая бесконечная \ll последовательность Киссинджера (/46/) является избыточной (/42/).

Целью настоящего раздела являются формулировка теоремы 1, в которой утверждается, что класс регулярно устойчивых алгебр совпадает с классом комбинаторных алгебр.

Использование термина «комбинаторная алгебра³» вызвано

³ Рассматриваемые комбинаторные алгебры не имеют отношения к

тем, что в теории конечных автоматов (как уже говорилось, понятие автомата совпадает с понятием унарной алгебры) существует понятие комбинаторный автомат, которое в точности подходит под введенное определение (на самом деле определение комбинаторных алгебр базируется на классическом понятии комбинаторного автомата).

На содержательном уровне комбинаторная алгебра – это алгебра, в которой классический метод итерации работает для любого уравнения от одной переменной. Например, уравнение от одной переменной может иметь вид $x = f(x, g(x, 1))$.

В конечном случае (а именно такой случай и рассматривается в работе) это означает, что при решении уравнения $x = t(x)$ (t – сколь угодно сложное выражение с одной переменной, например $f(x, g(x, 1))$) над конечной комбинаторной алгеброй точное решение отыскивается за конечное число итераций. В бесконечном же случае для работоспособности стратегии Киссинджера потребуются дополнительные ограничения. Одним из таких ограничений является условие равномерной сходимости (хватит и равномерной фундаментальности) последовательности отображений $t(x), t(t(x)), t(t(t(x))), \dots$

Из теории автоматов ([1, 12]) известно следующее необходимое и достаточное условие комбинаторности.

Утверждение 1. *Конечный автомат (Q, F, μ) (конечная унарная алгебра (Q, F, μ) (/25/)) является комбинаторным тогда и только тогда, когда выполняется следующее условие. Для любой входной последовательности α (для алгебраической функции f_α , задаваемой строкой α), для любого состояния $q \in Q$ (для любого элемента алгебры $q \in Q$) и любого натурального числа n из $(f_\alpha)^n(q) = q$ следует $f_\alpha(q) = q$.*

Пример 7. Автомат, представленный на рис. 5, не является комбинаторным автоматом (алгебра A_{party} из примера 5 не является комбинаторной алгеброй), так как

тем комбинаторным алгебрам, которые изучаются в λ -исчислении.

“не любит”²(воскресенье) = “не любит”(“не любит”(воскресенье)) = воскресенье и, в то же время, “не любит”(воскресенье) = суббота. •

Пример 8. Далее приводятся примеры достаточных условий комбинаторности (рассматриваются лишь конечные алгебры)

a) если на элементах алгебры можно задать линейный порядок так, чтобы все алгебраические операции задавали монотонные функции, то алгебра комбинаторна;

b) если на элементах алгебры можно задать метрику, при которой все алгебраические операции задают сжимающее отображение, то алгебра комбинаторна;

c) объединение не пересекающихся по элементам комбинаторных алгебр — комбинаторная алгебра;

d) прямое произведение комбинаторных алгебр — комбинаторная алгебра;

e) гомоморфный образ комбинаторной алгебры — комбинаторная алгебра. •

Теорема 1. Множество регулярно устойчивых алгебр (/7/) совпадает с множеством комбинаторных алгебр (/45/).

Доказательство в приложении (утверждения 25.следствие 2 и 26).

Заключение

В работе были определены ограничения на поведение объектов, обеспечивающие возможность настройки любой функциональной сети, построенной из них (поведение объектов должны задаваться операциями комбинаторной алгебры), и был приведен алгоритм настройки, который гарантированно приводит к успеху при соблюдении введенных ограничений.

Главная тема дальнейших исследований в намеченном направлении: доказать гипотезу о том, что время настройки сети, представленной над фиксированной конечной комбинаторной алгеброй, пропорционально n^2 , где n — число настраиваемых объектов. Еще одно направление — это обобщение приведенного ре-

зультата на случай бесконечных алгебр.

Следует отметить, что алгоритм настройки един для всех комбинаторных алгебр и, следовательно, является хорошим претендентом на эвристический метод настройки схем над произвольными алгебрами (не обязательно комбинаторными и не обязательно конечными): случай, когда не известны ни правила поведения объектов, ни структура их взаимодействий.

Литература

1. АРБИБ М. А., ГИНЗБУРГ С., ЗЕЙГЕР П., и другие *Алгебраическая теория автоматов, языков и полугрупп*. – М.: Статистика, 1975. – 335 с. – [Электронный ресурс]. – URL: <http://bookre.org/reader?file=437779> (дата обращения: 10.07.2013).
2. АРЖАНЦЕВ И.В. *Базисы Грёбнера и системы алгебраических уравнений*. – М.:МЦНМО, 2003. – 68 с. – [Электронный ресурс]. – URL: <http://www.mccme.ru/free-books/dubna/arjantsev.pdf> (дата обращения: 10.07.2013).
3. АРТАМОНОВ В.А., САЛИЙ В.Н. , СКОРНЯКОВ Л.А., ШЕВРИН Л.Н., ШУЛЬГЕЙФЕР Е.Г. *Общая алгебра*. – Т.2. СМБ. – М.:«Наука» Физматлит, 1991. – 479. – [Электронный ресурс]. – URL: <http://bookre.org/reader?file=441272> (дата обращения: 10.07.2013).
4. БУХБЕРГЕР Б., КОЛЛИНЗ ДЖ., ЛООС Р. *Компьютерная алгебра: символьные и алгебраические вычисления*. – М.: Мир, 1986. – 392 с. – [Электронный ресурс]. – URL: <http://bookre.org/reader?file=437353> (дата обращения: 10.07.2013).
5. ВАЛИЕВ М.К., ДЕХТЯРЬ М.И., ДИКОВСКИЙ А.Я. *Системы агентов, управляемых логическими программами: сложность верификации // Программирование*. – 2009. – №5. – С. 37–56. – [Электронный ресурс]. – URL: <http://pagesperso.lina.univ->

- nantes.fr/~dikovsky-a-1/publies/MASJProgr09.pdf (дата обращения: 30.10.2013).
6. ДЕХТЯРЬ М.И., ДИКОВСКИЙ А.Я. *Анализ поведения дискретных динамических систем средствами логического программирования* // Программирование. – 1996. – №3. – С. 3–16. – [Электронный ресурс]. – URL: <http://homepages.tversu.ru/~p000103/ZHPROG.pdf> (дата обращения: 30.10.2013).
 7. ДЕХТЯРЬ М.И. *Обновления баз данных при динамических ограничениях целостности* // В сб. «Системная информатика» – 2002. – №8. – С. 72–142. – [Электронный ресурс]. – URL: <http://homepages.tversu.ru/~p000103/Nov-rus.pdf> (дата обращения: 30.10.2013).
 8. ДЭВЕНПОРТ ДЖ., СИРЭ И., ТУРНЬЕ Э. *Компьютерная алгебра.* – М.: Мир, 1991. – С. 126–139 – [Электронный ресурс]. – URL: <http://bookfi.org/book/758936> (дата обращения: 10.07.2013).
 9. КОКС Д., ЛИТТЛ ДЖ., О’ШИ Д. *Идеалы, многообразия и алгоритмы.* – М.:Мир, 2000. – С. 11–150. – [Электронный ресурс]. – URL: <http://bookre.org/reader?file=440091> (дата обращения: 16.07.2013).
 10. КОН П. *Универсальная алгебра.* – М.: Мир, 1968. – – [Электронный ресурс]. – 352 с. – URL: <http://bookre.org/reader?file=567818> (дата обращения: 10.07.2013).
 11. КУРОШ А.Г. *Лекции по общей алгебре.* – 2-е изд. М.: Физматлит, 1973. – [Электронный ресурс]. – 399 с. – URL: <http://bookre.org/reader?file=441263> (дата обращения: 10.07.2013).
 12. ЛАЛЛЕМАН Ж. *Полугруппы и комбинаторные приложения.* – М.: Мир, 1985. – С. 174–228. – [Электронный ресурс]. – URL: <http://bookre.org/reader?file=440008> (дата обращения:

- 10.07.2013).
13. МАЛИШЕВСКИЙ А.В. *Качественные модели в теории сложных систем.* – М.: «Наука» Физматлит, 1998. – С. 48–281. – [Электронный ресурс]. – URL: <http://bookre.org/reader?file=561940> (дата обращения: 30.10.2013).
 14. НОВИКОВ Д.А. *Управление проектами: организационные механизмы.* – М.: ПМСОФТ, 2007. – [Электронный ресурс]. – 140 с. – URL: <http://www.aup.ru/books/m185/> (дата обращения: 30.10.2013).
 15. ТРАХТЕНГЕРЦ Э.А. *Компьютерная поддержка переговоров при согласовании управленческих решений.* – М.: Синтег, 2003. – 273 с.
 16. BLAND R.G. *New finite pivot rules for simplex method* // *Math. Oper. Res.* – 1977. – Vol. 2. – P. 103–107.
 17. DEKHTYAR M., DIKOVSKY A., DUDAKOV S. *On Complexity of Updates Through Integrity Constraints* // *Proc. of the First Int. Conf. on Computational Logic (CL 2000)*, LNAI, Vol. 1861 2000. – P. 867–881. – [Электронный ресурс]. – URL: <http://pagesperso.lina.univ-nantes.fr/~dikovsky-a-1/publies/complog00.pdf> (дата обращения: 30.10.2013).
 18. DEKHTYAR M., DIKOVSKY A., VALIEV M. *Temporal Verification of Probabilistic Multi-Agent Systems* // In: *Trakhtenbrot/Festschrift*, [A. Aviron et al. (Eds.)] LNCS 4800. – Springer, 2008. – P. 256–265. – [Электронный ресурс]. – URL: http://library.keldysh.ru/prep_vw.asp?pid=2882 (дата обращения: 30.10.2013).
 19. КНОМЯКОВ М., БИДЕР И. *Achieving Workflow Flexibility through Taming the Chaos* // *OOIS 2000 - 6th international conference on object oriented information systems.* – Springer, 2000. – P. 85–92. – [Электронный ресурс]. – URL:

<http://www.ibissoft.se/publications/tamingchaos.pdf>
(дата обращения: 26.05.2013).

Приложение⁴

1. Используемые термины и обозначения

1.1. ЧИСЛА (НАТУРАЛЬНЫЕ)

Операции и отношения: +, -, <, >, ≤, ≥, *max*, *min*.

$\{1, 2, \dots\}$ - $(\{0, 1, \dots\} \stackrel{def}{=} \{1, 2, \dots\} \cup \{0\})$ множество натуральных чисел.

При $n \in \{0, 1, \dots\}$ $\{1, \dots, n\}$ – начальный отрезок натурального ряда длины n (при $n = 0$ $\{1, \dots, n\} = \emptyset$).

$max(s)$ ($min(s)$) – максимальный (минимальный) элемент не пустого множества чисел s (значения $min(s)$ и $max(s)$ не для всех множеств чисел определены, но заведомо определены для непустых конечных множеств, кроме того $min(s)$ определено для любого не пустого множества $s \subseteq \{0, 1, \dots\}$).

1.2. МНОЖЕСТВА

Операции и отношения: \cap , \cup , \setminus , \times , \in , \notin , \subseteq (пересечение, объединение, вычитание, прямое произведение, принадлежность/не принадлежность элемента множеству, включение множества).

\emptyset – пустое множество.

Если s – множество и $n \in \{1, 2, \dots\}$, то $s^n \stackrel{def}{=} s \times \dots \times s$ (прямая

степень множества: $\{a, b\}^1 = \{\{a\}, \{b\}\} \neq \{a, b\}$ и $\{a\}^2 \times \{a\} = \{((a, a), a)\} \neq \{a\} \times \{a\}^2 = \{(a, (a, a))\} \neq \{a\}^3 = \{(a, a, a)\}$).

$|s|$ – мощность множества s .

2^s – множество подмножеств множества s .

⁴ Приложение приводится в авторской редакции

1.3. ОТНОШЕНИЯ И ОТОБРАЖЕНИЯ

Множеством отношений (бинарных отношений) (иногда бинарные отношения называют также ориентированными графами) над Q называется множество всех подмножеств $Q \times Q$ (множество $2^{Q \times Q}$). Тот факт, что $(x, y) \in s$, принято записывать следующим образом: $x \ s \ y$.

Для любых отношений $r, s \subseteq Q \times Q$ (в приведенных далее обозначениях используются квадратные скобки, для того чтобы отличать прямое произведение множеств/отношений и степень/рефлексивно- транзитивное замыкание отношения от прямой степени/итерации множества)

- $r \bullet s \stackrel{def}{=} \{(a, b) : (a, c) \in r \text{ и } (c, b) \in s\}$ (произведение отношений);
- $s^{[0]} \stackrel{def}{=} \{(x, x) : x \in s\}$ (отношение равенства) обозначается $=_s$ (либо $=$ при известном s). При этом $\neq_s \stackrel{def}{=} s^2 \setminus =_s$;
- для любого $i \in \{1, 2, \dots\}$ $s^{[i]} \stackrel{def}{=} s^{[i-1]} \bullet s$ (степень отношения);
- $s^{[-1]} \stackrel{def}{=} \{(a, b) : (b, a) \in s\}$ (обозначение s^{-1} тоже допустимо);
- $s^{[*]} \stackrel{def}{=} \cup \{s^{[i]} : i \in \{0, 1, \dots\}\}$ – рефлексивно-транзитивное замыкание:
- $dom(r) \stackrel{def}{=} \{x : (x, y) \in r\}$;
- $vals(r) \stackrel{def}{=} \{y : (x, y) \in r\}$.

1.3.1. Множество эквивалентностей на множестве Q

(всех тех отношений $r \subseteq Q^2$, которые замкнуты относительно рефлексивно-транзитивного замыкания ($r = r^{[*]}$) и обращения ($r = r^{-1}$)) обозначается $Equivalences(Q)$.

1.3.2. Множество линейных порядков на множестве Q

(всех тех отношений $r \subseteq Q^2$, которые удовлетворяют равенствам: $r = r^{[*]}$, $r \cap r^{-1} = r^{[0]}$, $r \cup r^{-1} = Q^2$), обозначается $\langle \text{Line}_{Orders} \rangle(Q)$

Если Q – конечное непустое множество и $\ll \in \langle \text{Line}_{Orders} \rangle(Q)$ – линейный порядок на Q , то наибольший по порядку \ll элемент

множества Q обозначается $\max_{<<}(s)$.

1.3.3. Множество отображений из Q в Q'

(таких отношений $f \subseteq (Q \cup Q')^2$, что $\text{dom}(f) = Q$, $\text{vals}(f) \subseteq Q'$ и для любого $x \in Q$ $|\{y : (x, y) \in f\}| = 1$) обозначается $Q \rightarrow Q'$.

Тот факт, что $f \in Q \rightarrow Q'$, принято записывать следующим образом $f: Q \rightarrow Q'$ и для любого $x \in Q$ тот единственный элемент $y \in Q'$ для которого $(x, y) \in f$ обозначается $f(x)$.

Отображение $f: Q \rightarrow Q'$ называется обратимым, если $f^{-1} \in \text{vals}(f) \rightarrow Q$.

Отображение $f: Q \rightarrow Q'$ называется отображением на Q' , если $\text{vals}(f) = Q'$.

Обратимое отображение $f: Q \rightarrow Q'$ на Q' называется взаимно однозначным отображением.

Пусть $Q \subseteq \{0, 1, \dots\}$. Тогда отображение $f: Q \rightarrow \{0, 1, \dots\}$ называется монотонным в том и только в том случае, когда для любых $n, m \in \{0, 1, \dots\}$ из $n \leq m$ следует $f(n) \leq f(m)$.

Если $f \in Q \rightarrow Q'$ и s – множество (не требуется $s \subseteq Q$), то отображение $f \cap (s \times Q') \in (s \cap Q) \rightarrow Q'$ принято обозначать через $f|_s$.

Сужение операции произведения отношений на отображения (отношение $\bullet|_{(s \rightarrow Q)^2}$) называется операцией композиции отображений (обозначается \bullet).

Обозначения. Несмотря на то, что отображение является множеством, мы будем использовать для обозначения отображения $f^{[n]}$ то же обозначение, которое используется для прямой степени множества f^n . Конфликт при выборе способа интерпретации обозначения s^n должен разрешаться по следующему правилу: если множество s объявлено как отображение (если указано, что $s \in Q \rightarrow Q'$ для некоторых множеств Q и Q'), то запись s^n следует интерпретировать как $s^{[n]}$, в противном случае как прямая степень множества s .

Утверждение 2. Пусть Q – множество и $r, r', r'' \subseteq Q \times Q$. Тогда $(r \bullet r') \bullet r'' = r \bullet (r' \bullet r'')$ (1.3). При этом если $r, r' \in Q \rightarrow Q$ (1.3.3), то и $r \bullet r' \in Q \rightarrow Q$. Без доказательства

1.4. СТРОКИ И ПОСЛЕДОВАТЕЛЬНОСТИ

Отображения (/1.3.3/) из начального отрезка натурального ряда $\{1, \dots, n\}$, где $n \in \{0, 1, \dots\}$, во множество V (конечные последовательности) называются строками над алфавитом V . При этом отображения из $\{1, 2, \dots\} \rightarrow V$ называются (бесконечными) последовательностями над V .

Множество $\{1, \dots, n\} \rightarrow V$ (множество всех строк длины $n \in \{0, 1, \dots\}$ над алфавитом V) обозначается V^n (так же как прямая степень множества V).

Множество всех строк над алфавитом V обозначается V^* :
 $V^* \stackrel{def}{=} \cup \{V^n : n \in \{0, 1, \dots\}\}$.

Если $\alpha, \beta \in V^*$ – строки над алфавитом V , то

- $length(\alpha)$ – длина строки α ($length(\alpha) \stackrel{def}{=} |dom(\alpha)|$);
- строка длины 0 называется пустой строкой и обозначается

ε (хотя это не существенно для дальнейшего, но на самом деле $\emptyset = \varepsilon$);

- если $\alpha \in V^* \setminus \{\varepsilon\}$ не пустая строка, то для любого $i \in \{1, \dots, length(\alpha)\}$ $\alpha(i)$ – i -й символ строки α (поскольку $\alpha \in \{1, \dots, length(\alpha)\} \rightarrow V$) и $last(\alpha) \stackrel{def}{=} \alpha(length(\alpha))$ – последний символ строки α .

- результат конкатенации α и β (результат приписывания β после α) будет в дальнейшем обозначаться $\alpha \wedge \beta$, либо $\alpha\beta$.

- в дальнейшем, когда это не будет вызывать неоднозначности, элементы алфавита V и строки длины 1 будут обозначаться одинаковым образом (иногда символ алфавита $f \in V$ будет обозначать такую строку α , что $length(\alpha) = 1$ и $\alpha(1) = f$). Но в тех случаях, когда элемент $a \in V$ требуется отличить от строки a длины 1, для обозначения строки используется обозначение $\{a\}$ ($\{a\} \in V^1$ – это строка длины 1, первым символом которой является a). Таким образом $\{a\}(1) = a$ и $length(\{a\}) = 1$.

- любая строка $\alpha \in V^*$ является (по определению /1.4/) отображением, а значит (по определению /1.3.3/), и множеством. Поэтому использование стандартной нотации приводит к тому, что разные объекты: $\alpha \wedge \alpha \dots \wedge \alpha$ (n раз) (строка из V^*) и $\alpha \times \alpha \times \dots \times \alpha$ (n раз), обозначаются одинаковым образом α^n . Поэтому для записи $\alpha \wedge \alpha \dots \wedge \alpha$ (n раз) будет использоваться обозначение $\alpha^{\wedge n}$.

Определение 8. Пусть V – множество. Определим отображения $prefixes: V^* \rightarrow 2^{(V^*)}$ и $suffixes: V^* \rightarrow 2^{(V^*)}$ ((1.4)) следующим образом: для любой строки $\alpha \in V^*$ $prefixes(\alpha) \stackrel{def}{=} \{\alpha' : \alpha' \wedge \beta' = \alpha\}$ и $suffixes(\alpha) \stackrel{def}{=} \{\beta' : \alpha' \wedge \beta' = \alpha\}$.

Определение 9. Пусть V – множество. Определим отображения $Prefix: V^* \times \{0, 1, \dots\} \rightarrow V^*$ и $Concat: (V^*)^* \rightarrow V^*$ следующим образом.

Для любых $\pi \in V^*$ и $i \in \{1, 2, \dots\}$

$$Prefix(\pi, i) \stackrel{def}{=} \begin{cases} \pi, & \text{если } i \notin \text{dom}(\pi) \cup \{0\} \\ \pi', & \text{если } i \in \text{dom}(\pi), \text{length}(\pi') = i \text{ и } \pi' \in \text{prefixes}(\pi) \end{cases} \quad ((1.3.3, 1.4.8)).$$

Для любых $\pi \in (V^*)^*$

$$Concat(\pi) \stackrel{def}{=} \begin{cases} \varepsilon, & \text{если } \pi = \varepsilon \\ Concat(\pi') \wedge \alpha, & \text{если } \pi = \pi' \wedge \{\alpha\} \text{ и } \alpha \in V^* \end{cases} \quad ((1.4)).$$

Определение 10. Пусть V – множество и $r \subseteq V \times V$ – отношение ((1.3)). Продолжением отношения r на строки называется отношение $r' \subseteq V^* \times V^*$ такое, что для любых $\alpha, \beta \in V^*$ ((1.4)) $\alpha r' \beta$ тогда и только тогда, когда выполняются следующие условия:

- $\text{length}(\alpha) = \text{length}(\beta)$ ((1.4))
- для любого $i \in \{1, \dots, \text{length}(\alpha)\}$ выполняется отношение $\alpha(i) r \beta(i)$

В дальнейшем продолжение отношения r на строки будет обозначаться $[*]^r$ (в частности, для любого отношения $r \subseteq V \times V \in [*]^r \varepsilon$).

Утверждение 3. Пусть $r \subseteq V \times V$ – отношение ((1.3)). Тогда

- если $r \in V \rightarrow V$ ((1.3.3)), то $[*]^r \in V^* \rightarrow V^*$ ((10, 1.3.3));
- если r – отношение эквивалентности ((1.3.1)), то $[*]^r$ – отношение эквивалентности ((10, 1.3.1)).

Без
доказательства

Определение 11. Пусть S – множество и $\ll \in \langle \begin{smallmatrix} \text{Line} \\ \text{Orders} \end{smallmatrix} \rangle(S)$ – линейный порядок на множестве S (1.3.2). Для любого конечного подмножества $s \subseteq S$ и любого $x \in s$ номером x в s в соответствии с порядком \ll (обозначается $\text{num}_{\ll, s}(x)$) называется число $\text{num}_{\ll, s}(x) \stackrel{\text{def}}{=} |\{y : y \in s \ \& \ y \ll x\}|$.

Утверждение 4. Пусть S – множество, $\ll \in \langle \begin{smallmatrix} \text{Line} \\ \text{Orders} \end{smallmatrix} \rangle(S)$ – линейный порядок на множестве S (1.3.2) и $s \subseteq S$ конечное подмножество. Тогда отображение $\text{num}_{\ll, s} : s \rightarrow \{1, \dots, |s|\}$ (11) является обратимым отображением на $\{1, \dots, |s|\}$ (отношение $(\text{num}_{s, \ll})^{-1} \subseteq s \times \{1, 2, \dots\}$ является отображением из $1, \dots, |s|$ на s). Без
доказательства

Замечание . Таким образом, $(\text{num}_{s, \ll})^{-1}$ это такая строка длины $|s|$ над алфавитом s , в которой все элементы расположены по возрастанию (для любого $i \in \{1, \dots, |s|\}$ выполняется равенство $(\text{num}_{\ll, s})^{-1}(i) = \max_{\ll}(\{x : x \in s \ \& \ \text{num}_{\ll, s}(x) \leq i\})$).

Определение 12. Пусть Q – конечное множество, $\ll \in \langle \begin{smallmatrix} \text{Line} \\ \text{Orders} \end{smallmatrix} \rangle(Q)$ – линейный порядок на Q (1.3.2), $s \subseteq Q$ и $\varphi : s \rightarrow Q'$ некоторое отображение (1.3.3). Введем следующие обозначения:
Отображение $(\text{num}_{s, \ll})^{-1} : \{1, \dots, |s|\} \rightarrow Q$ (11.1.3) будем обозначать $\text{string}(s, \ll)$ ($\text{string}(s, \ll)$ это такая строка $\alpha \in Q^{|s|}$, что для любых $i, j \in \{1, \dots, |s|\}$ $i \leq j$ тогда и только тогда, когда $\alpha(i) \ll \alpha(j)$).

Отображение $[*]^\varphi(\text{string}(s, \ll)) \in Q'^{|s|}$ (11.10) будем обозначать $\text{string}(s, \ll, \varphi)$ ($\text{string}(s, \ll, \varphi)$ это такая строка $\beta : \{1, \dots, |s|\} \rightarrow Q'$, что для любых $i \in \{1, \dots, |s|\}$ $\beta(i) = \varphi(\text{string}(s, \ll)(i))$: строка $\text{string}(s, \ll, \varphi)$ получается из строки $\lambda = \text{string}(s, \ll)$ заменой каждого символа $\lambda(i)$ на $\varphi(\lambda(i))$).

Определение 13. Пусть V множество, $\alpha \in V^*$ – последовательность и $s \subseteq \{1, 2, \dots\}$. Тогда s – подпоследовательностью последовательности α (обозначается $\text{subseq}(s, \alpha)$ – это последовательность, задаваемая множеством позиций $s \cap \text{dom}(\alpha)$ в строке α) называется

такая последовательность $\beta \in V^*$, что $|\text{dom}(\beta)| = |\text{dom}(\alpha) \cap s|$ и для любого $n \in \text{dom}(\beta)$ выполняется равенство

$$\beta(n) = \alpha(|\{i : i \in \text{dom}(\alpha) \cap s \& i \leq n\}|).$$

Утверждение 5. Пусть V множество, $\alpha \in V^*$ (1.4) - последовательность и $s \subseteq \{1, 2, \dots\}$. Тогда $\text{subseq}(s, \alpha) = \text{string}(s \cap \text{dom}(\alpha), \leq, \alpha|_s)$ (12,13). Без доказательства

Определение 14. Пусть V множество, $V' \subseteq V$ и $\varphi: V' \rightarrow V^*$ некоторое отображение. Строковым гомоморфизмом, порожденным отображением φ , называется отображение $\varphi^{\text{Strings}(V)}: V^* \rightarrow V^*$ такое, что для любой строки $\alpha \in V^*$ и для строки $\pi \in (V^*)^*$ такой, что $\text{dom}(\pi) = \text{dom}(\alpha)$ и для всех $i \in \text{dom}(\pi)$ $\pi(i) = \begin{cases} \varphi(\alpha(i)), & \text{если } \alpha(i) \in V' \\ \{\alpha(i)\}, & \text{если } \alpha(i) \notin V' \end{cases}$, $\varphi^{\text{Strings}(V)}(\alpha) = \text{Concat}(\pi)$ (19).

1.5. ТЕРМЫ

Определение 15. Множеством функциональных символов называется множество F с определенным на нем отображением $\text{arn}: F \rightarrow \{0, 1, \dots\}$ (1.3.3), которое сопоставляет каждому элементу $f \in F$ неотрицательное целое число – арность.

Замечание . В новосибирской школе вместо термина «множество функциональных символов» используется термин «сигнатура»: сигнатура это пара (F, arn) , где F – множество (функциональных символов) и $\text{arn} \in F \rightarrow \{0, 1, \dots\}$.

Определение 16. Пусть F – множество функциональных символов (15) и V – множество функциональных символов арности 0, не пересекающееся с F (множество переменных).

Множеством $\text{Terms}(F, V)$ термов над F и переменными V называется минимальное множество строк над алфавитом $F \cup V$, удовлетворяющее следующим условиям:

- $V^1 \subseteq Terms(F, V)$ (1.4);
- Если $f \in F$ и $\pi \in Terms(F, V)^{arm(f)}$, то $\{f\}^{\wedge} Concat(\pi) \in Terms(F, V)$ (9.1.4).

Термы из множества $Terms(F, \emptyset)$ (выражения без переменных) называются основными над множеством функциональных символов F (или основными).

Замечание . Следует обратить внимание на то, что $Terms(F, V) = Terms(F \cup V, \emptyset)$ и для любых множеств V', V'' $Terms(F, V' \cup V'') = Terms(F \cup V', V'' \setminus V')$.

В дальнейшем терм $ft_1 \dots t_n = \{f\}^{\wedge} t_1 \dots t_n$ будет для наглядности записываться в виде $f(t_1, \dots, t_n)$. То, что такое соглашение корректно провозглашается в следующем утверждении.

Утверждение 6. Пусть F – множество функциональных символов (15) и V – множество функциональных символов арности 0, не пересекающееся с F (множество переменных). Тогда для любого $t \in Terms(F, V)$ $t \neq \varepsilon$ и при этом существует и единственна такая последовательность $\pi \in (Terms(F, V) \setminus \{t\})^*$, что $t = \{t(1)\}^{\wedge} Concat(\pi)$ (9). Без доказательства

Определение 17. Определим отображения

- $topFunctor: Terms(F, V) \rightarrow F \cup V$,
- $Sons: Terms(F, V) \rightarrow Terms(F, V)^*$
- $\langle \substack{sub \\ terms} \rangle: Terms(F, V) \rightarrow 2^{Terms(F, V)}$ и
- для любого множества $W \subseteq V$ $vars_W: Terms(F, V) \rightarrow 2^V$ следующим образом.

Для любых $t \in Terms(F, V) \cup \{\varepsilon\}$ и $W \subseteq V$

- $topFunctor(t) \stackrel{def}{=} t(1)$ (поскольку t – не пустая строка, то значение $t(1)$ определено).
- $Sons(t)$ такая (согласно утв.6 единственная) последовательность $\pi \in Terms(F, V)^*$, что $t = \{t(1)\}^{\wedge} Concat(\pi)$
- $\langle \substack{sub \\ terms} \rangle(t) \stackrel{def}{=} \{ \tau : \tau \in Terms(F, V) \& \alpha^{\wedge} \tau^{\wedge} \beta = t \& \alpha, \beta \in (F \cup V)^* \}$

- $vars_W(t) \stackrel{def}{=} \langle \substack{sub \\ terms} \rangle(t) \cap W$.

В дальнейшем, когда множество переменных V известно из контекста, вместо записи $vars_V(t)$ будет использоваться сокращение $vars(t)$.

Утверждение 7. Для любых $t \in Terms(F, V)$ и $\tau \in \langle \substack{sub \\ terms} \rangle(t)$ справедливо включение $Sons(\tau) \in \langle \substack{sub \\ terms} \rangle(t) \setminus \{t\}^*$.

При этом $length(Sons(\tau)) = arn(\tau(1))$ (это утверждение понадобится в определении /26/). Без
доказательства

1.6. ПОДСТАНОВКИ

Определение 18. Подстановкой (переменных) называется такое отображение (/1.3.3/) $\varphi: Terms(F, V) \rightarrow Terms(F, V)$ (/16/), что для некоторого отображения $f: V \rightarrow Terms(F, V)$ (/16/), справедливо равенство $\varphi = f^{Strings(F \cup V)}|_{Terms(F, V)}$ (/14/).

Множество всех подстановок переменных из V принадлежащих множеству $Terms(F, V) \rightarrow Terms(F, V)$ в дальнейшем обозначается $substitutions(F, V)$.

Если $\varphi \in substitutions(F, V)$ - подстановка, то отображение $f: V \rightarrow Terms(F, V)$ такое, что для любого $x \in V$ $f(x) = \varphi(\{x\})$ будет обозначаться $\varphi|_V$.

Подстановка $\varphi \in substitutions(F, V)$ называется основной, если для любой переменной $x \in V$ $\varphi(\{x\}) \in Terms(F, \emptyset)$ - основной терм. Множество основных подстановок из $substitutions(F, V)$ будет обозначаться $valuations(F, V)$.

Ядром подстановки $\varphi \in substitutions(F, V)$ называется множество $kernel(\varphi) \stackrel{def}{=} \{x : x \in V \& \varphi(x) \neq x\}$ (в частности, ядро тождественной подстановки - пустое множество).

Обозначения. В дальнейшем для любых подстановок

$\varphi, \varphi' \in substitutions(F, V)$ (/18/) их композиция $\varphi \bullet \varphi'$ (/1.3, 1.3.3/) (в соответствии с последующим утверждением 10 $\varphi \bullet \varphi' \in substitutions(F, V)$) будет обозначаться $\varphi'(\varphi)$, либо $\varphi' \varphi$ (в соответствии с утверждением 2 расстановка скобок не существенна: $\varphi \varphi' \varphi'' \stackrel{def}{=} (\varphi(\varphi'))(\varphi'') = \varphi(\varphi'(\varphi''))$).

Кроме того, любое множество S пар вида $x \rightarrow t$, где $x \in V$, $t \in \text{Terms}(F, V)$, такое что $\{x \rightarrow t, x \rightarrow t'\} \subseteq S$ влечет $t = t'$, может быть использовано для обозначения такой подстановки

$\varphi_S \in \text{substitutions}(F, V)$, что

$$\text{kernel}(\varphi) \subseteq \{x : x \rightarrow t \in S\} \text{ и}$$

для любой пары $x \rightarrow t \in S$ $\varphi(x) = t$.

Например, $\{x \rightarrow t\}$ – обозначение для такой подстановки $\varphi \in \text{substitutions}(F, V)$, что $\varphi(x) = t$ и для любого $y \in V \setminus \{x\}$ выполняется равенство $\varphi(y) = y$.

Утверждение 8. Пусть $\varphi : V \rightarrow \text{Terms}(F, V)$ – отображение из множества переменных во множество термов. Тогда существует и единственна подстановка $\varphi' \in \text{substitutions}(F, V)$ такая, что $\varphi' \upharpoonright_V = \varphi$. Без доказательства

Следствие 1. Пусть $\varphi, \varphi' \in \text{substitutions}(F, V)$ и $t \in \text{Terms}(F, V)$. Тогда $\varphi(t) = \varphi'(t)$ тогда и только тогда, когда для любой переменной $x \in \text{vars}_V(t)$ $\varphi(x) = \varphi'(x)$. Без доказательства

Определение 19. Пусть $\varphi : V \rightarrow \text{Terms}(F, V)$ – отображение из множества переменных во множество термов. Подстановка $\varphi' \in \text{substitutions}(F, V)$ такая, что $\varphi' \upharpoonright_V = \varphi$ будет называться φ – подстановкой. В дальнейшем φ – подстановка $\varphi' \in \text{substitutions}(F, V)$ будет обозначаться $\varphi^{\text{Terms}(F)}$, либо, если F известно из контекста, φ^{Terms} .

Утверждение 9. Пусть $\varphi \in \text{substitutions}(F, V)$ и $\psi \in \text{valuations}(F, V)$. Тогда $\psi \bullet \varphi \in \text{valuations}(F, V)$ и $\varphi \bullet \psi = \psi$.

Без доказательства

Утверждение 10. Подстановки (18) замкнуты относительно операции композиции (1.3.3) (Если $\varphi, \varphi' \in \text{substitutions}(F, V)$ (18), то $\varphi \bullet \varphi' \in \text{substitutions}(F, V)$ (1.3. 1.3.3)). Без доказательства

1.7. СИСТЕМЫ УРАВНЕНИЙ

В последующих определениях вводится понятие системы уравнений.

Все определения почти стандартны, за тем исключением, что все уравнения имеют специальный вид: «переменная»= «выражение». Кроме того, накладывается следующее ограничение: у различных уравнений системы должны быть различные левые части.

Определение 20. Уравнениями (или равенствами) над множеством переменных V и множеством функциональных символов F (15) будем называть элементы множества $equations(F, V) \stackrel{def}{=} V \times (Terms(F, V) \setminus V^1)$ (16). В дальнейшем элементы $(x, t) \in equations(F, V)$ будут обозначаться $x=t$, где $x \in V$ – переменная, а $t \in Terms(F, V) \setminus V^1$ (16) ($equations(F, V) = \{x=t : x \in V \& t \in Terms(F, V) \setminus V^1\}$).

Определение 21. Множество $s \subseteq equations(F, V)$ (20) будем называть системой уравнений (диагональной системой уравнений) над переменными из V и функциональными символами из F (15), если из $x=t \in s$ и $x=t' \in s$ следует $t=t'$. Множество всех диагональных систем уравнений над переменными из V и функциональными символами из F будем обозначать через $Systems(F, V)$ ($\emptyset \in Systems(F, V)$).

Определение 22. На множестве систем уравнений определим отображение $\langle_{Vars}^{bound} \rangle : Systems(F, V) \rightarrow 2^V$ (21) следующим образом: для любой системы $s \in Systems(F, V)$ $\langle_{Vars}^{bound} \rangle(s) \stackrel{def}{=} \{x : x = \tau \in s\}$.

Определение 23. Пусть F – множество функциональных символов (15) и V – множество переменных (множество функциональных символов арности 0) не пересекающееся с F . Определим отображение $\langle_{ToSubs}^{System} \rangle : Systems(F, V) \rightarrow substitutions(F, V)$

следующим образом.

Для всех $s \in \text{Systems}(F, V)$ (/21/) $\langle \text{System}_{\text{ToSubs}} \rangle(s)$ это такая подстановка $\varphi \in \text{substitutions}(F, V)$ (/18/), что $\text{kernel}(\varphi) \stackrel{\text{def}}{=}} \{y : y=t \in s \& y \neq t\}$ (/18/) и для любого $x \in \text{kernel}(\varphi)$ $x = \varphi(x) \in s$.

Определение 24. Пусть $s \in \text{Systems}(F, V)$ (/21/).

Множеством внутренних подстановок системы уравнений s , называется наименьшее множество подстановок $\langle \text{system}_{\text{Subs}} \rangle(s) \subseteq \text{substitutions}(F, V)$, удовлетворяющее следующим условиям:

- Для любой системы уравнений $s' \subseteq s$

$$\langle \text{System}_{\text{ToSubs}} \rangle(s') \in \langle \text{system}_{\text{Subs}} \rangle(s);$$
- Если $\varphi, \varphi' \in \langle \text{system}_{\text{Subs}} \rangle(s)$, то $\varphi(\varphi') \in \langle \text{system}_{\text{Subs}} \rangle(s)$.

1.8. ИНТЕРПРЕТАЦИЯ И ПОДОБИЕ ТЕРМОВ И ПОДСТАНОВОК

Определение 25. Универсальной алгеброй [10, 11] называется тройка $A = (Q, F, \mu)$, где

Q – не пустое множество элементов алгебры,

F – множество функциональных символов (/15/), такое, что для любого $q \in Q \cap F$ $\text{argn}(q) = 0$

$\mu: F \rightarrow \cup\{Q^i \rightarrow Q : i \in \{0, 1, \dots\}\}$ – интерпретация функциональных символов – отображение, которое каждому $f \in F$ ставит в соответствие отображение $\varphi(f): Q^{\text{argn}(f)} \rightarrow Q$ (/1.3.3/), определенное над последовательностями длины $\text{argn}(f)$ (/1.4/) такое, что для любого $q \in Q \cap F$ $\mu(q)(\varepsilon) = q$.

Содержательно, ограничение $\mu(q)(\varepsilon) = q$ для $q \in Q \cap F$ означает, что элементы алгебры в качестве функциональных символов, используются только для обозначения самих себя.

В частности, если для $c \in F$ $\text{argn}(c) = 0$, то областью определения $\varphi(c)$ является множество, состоящее из единственного элемента ε – последовательность длины 0.

В дальнейшем мы будем пользоваться термином «алгебра» вместо термина «универсальная алгебра».

Если Q и F – конечные множества, то алгебра называется конечной.

Определение 26. Пусть $A=(Q, F, \mu)$ – алгебра (125) и $t \in Terms(F, \emptyset)$. Будем говорить, что отображение $\varphi: \langle \substack{sub \\ terms} \rangle(t) \rightarrow Q$ является A – интерпретацией терма t , если для любого $\tau \in \langle \substack{sub \\ terms} \rangle(t)$ выполняется равенство $\varphi(\tau) = \mu(\tau(1))([\ast]^\varphi(Sons(\tau)))$ (если $\varphi(\tau) = \mu(f)(\varphi(\tau_1), \dots, \varphi(\tau_n))$, где $\tau = f(\tau_1, \dots, \tau_n)$). В соответствии с утверждением 7 определение корректно.

Утверждение 11. Пусть $A=(Q, F, \mu)$ – алгебра (125). Тогда для любого $t \in Terms(F, \emptyset)$ существует и единственна A -интерпретация. Без доказательства

Определение 27. Пусть $A=(Q, F, \mu)$ – алгебра (125). Определим отображение $Interpretation_A: Terms(F, \emptyset) \rightarrow Q$ (116) следующим образом:

для любого $t \in Terms(F, \emptyset)$ (116), $Interpretation_A(t)$ равно $\varphi(t)$, где φ A -интерпретация терма t (согласно утверждению 11 определение корректно)

Определение 28. Алгебра $A=(Q, F, \mu)$ (125) называется выразительной, если $\{Interpretation_A(t) : t \in Terms(F, \emptyset)\} = Q$. Выразительным замыканием A (обозначается $extension(A)$) называется алгебра $(Q, F \cup Q, \mu')$, такая, что $\mu'(f) = \mu(f)$ для всех $f \in F$.

Содержательно, выразительное замыкание алгебры A получается из A добавлением всех ее элементов в качестве констант (функциональных символов нулевой аргности).

Определение 29. Пусть $A=(Q, F, \mu)$ – алгебра (125), и $B=extension(A)$ – её выразительное замыкание (128). В дальнейшем, для любого терма $t \in Terms(F \cup Q, \emptyset)$ элемент алгебры $Interpretation_B(t) \in Q$ будет обозначаться $value_A(t)$.

Кроме того, если V – множество (переменных) не пересекающееся с $F \cup Q$, и $\varphi \in \text{valuations}(F \cup Q, V)$, то $\langle \text{value}_{A}^{\text{subs}}(\varphi) \rangle$ это такое отображение $f: V \rightarrow Q$, что $f(x) = \text{value}_A(\varphi(x))$ для всех $x \in V$.

Определение 30. Пусть $A = (Q, F, \mu)$ – алгебра (25). Будем говорить, что термы $t1, t2 \in \text{Terms}(F \cup Q, V \setminus Q)$ (16) подобны в алгебре A (обозначается $t1 \equiv_{A, \text{terms}} t2$ либо $t1 \equiv_A t2$), если для любой основной подстановки $\varphi \in \text{valuations}(Q, V \setminus Q)$ справедливо равенство $\text{value}_A(\varphi(t1)) = \text{value}_A(\varphi(t2))$ (27). Будем говорить, что подстановки $\varphi, \varphi' \in \text{substitutions}(F \cup Q, V \setminus Q)$ (18) подобны в алгебре A (обозначается $\varphi \equiv_{A, \text{subs}} \varphi'$, либо $\varphi \equiv_A \varphi'$), если для любой переменной $x \in V \setminus Q$ термы $\varphi(x)$ и $\varphi'(x)$ подобны ($\varphi(x) \equiv_A \varphi'(x)$).

Утверждение 12. Пусть $A = (Q, F, \mu)$ – алгебра (25) и V множество (переменных) не пересекающееся с $F \cup Q$. Тогда

- $\equiv_{A, \text{subs}} \in \text{Equivalences}(\text{substitutions}(F \cup Q, V))$ (1.3.1),
- $\equiv_{A, \text{terms}} \in \text{Equivalences}(\text{Terms}(F \cup Q, V))$ и
- для любых $\varphi, \psi, \varphi', \psi' \in \text{substitutions}(F \cup Q, V)$ (18) и любых $t, t' \in \text{Terms}(F \cup Q, V)$ (16) таких, что $\varphi \equiv_{A, \text{subs}} \varphi'$, $\psi \equiv_{A, \text{subs}} \psi'$ и $t \equiv_{A, \text{terms}} t'$ выполняются отношения $\varphi(t) \equiv_{A, \text{terms}} \varphi'(t')$ и $\varphi\psi \equiv_{A, \text{subs}} \varphi'\psi'$ (30). Без доказательства

1.9. РЕШЕНИЯ СИСТЕМ УРАВНЕНИЙ

Далее с помощью введенного определения интерпретации термов определяется понятие решения и параметрического решения системы уравнений.

Определение 31. Пусть $A = (Q, F, \mu)$ – алгебра (25), V – множество (переменных), не пересекающееся с $F \cup Q$, и $s \in \text{Systems}(F, V)$ – система уравнений (21). Отображение $\varphi: V \rightarrow Q$ называется решением s в A , если для любого уравнения $x = t \in s$ (20) справедливо равенство $\varphi(x) = \text{value}_A(\varphi^{\text{Terms}}(t))$ (18, 27) (равенство $\varphi(x) = \text{value}_A(\varphi^{\text{Terms}}(t))$ эквивалентно тождеству $\varphi(x) \equiv_A \varphi^{\text{Terms}}(t)$).

Множество всех решений системы уравнений s в A обозначается $solutions(s, A)$

Определение 32. Пусть $A=(Q, F, \mu)$ – алгебра (25), V – множество (переменных), не пересекающееся с $F \cup Q$, и $s \in Systems(F, V)$ – система уравнений (21). Отображение $\varphi: V \rightarrow Terms(F \cup Q, V)$ (16) называется параметрическим решением s в A , если для любого отображения $\psi: V \rightarrow Q$ отображение $\langle_{Value}^{subs} \rangle_A(\psi^{Terms}(\varphi^{Terms}))$ (29,19) является решением s в A (31/) (любое решение является параметрическим решением, но не наоборот). Множество всех параметрических решений системы $s \in Systems(F, V)$ над A будет обозначаться $\langle_{Solutions}^{Parametric} \rangle_{A, V}(s)$.

Параметрическое решение $\varphi \in \langle_{Solutions}^{Parametric} \rangle_{A, V}(s)$ системы s называется полным, если для любого решения $\psi \in solutions(s, A)$ системы s существует отображение $\lambda: V \rightarrow Q$ такое, что выполняется $\langle_{Value}^{subs} \rangle_A(\lambda^{Terms}(\varphi^{Terms})) = \psi(x)$ (29,19/).

Будем говорить, что параметрическое решение $\varphi \in \langle_{Solutions}^{Parametric} \rangle_{A, V}(s)$ системы $s \in Systems(F, V)$, является свободным решением s , если $\varphi^{Terms} \in \langle_{Subs}^{system} \rangle(s)$ (19,24/). Множество всех свободных решений системы $s \in Systems(F, V)$ будем обозначать $\langle_{Solutions}^{Free} \rangle_A(s)$.

Утверждение 13. Пусть $A=(Q, F, \mu)$ – алгебра (25), V – множество (переменных), не пересекающееся с $F \cup Q$, и

$\varphi \in \langle_{Solutions}^{Parametric} \rangle_{A, V}(s)$ – параметрическое решение (32/) системы уравнений $s \in Systems(F, V)$ (21). Тогда для любой подстановки $\psi \in substitutions(F \cup Q, V)$ справедливо включение $\psi(\varphi^{Terms}(F \cup Q, V)) \in \langle_{Solutions}^{Parametric} \rangle_{A, V}(s)$. Без доказательства

Следствие 1. Пусть $A=(Q, F, \mu)$ – алгебра (25), V – множество (переменных), не пересекающееся с $F \cup Q$, и φ – параметрическое решение (32/) системы уравнений (21) $s \in Systems(F, V)$ ($\varphi \in \langle_{Solutions}^{Parametric} \rangle_{A, V}(s)$). Тогда для любой основной подстановки $\psi \in evaluations(F \cup Q, V)$ справедливо включение $\langle_{Value}^{subs} \rangle_A(\psi \varphi^{Terms}(F \cup Q, V)) \in solutions(s, A)$

(значение $\langle \text{subs} \rangle_A(\psi\varphi^{Terms(F\cup Q, V)})$ определено в силу утв. 9:
 $\psi\varphi^{Terms(F\cup Q, V)} \in \text{evaluations}(F\cup Q, V)$). Без
доказательства

Следующее утверждение фиксирует, тот факт, что при замене любой переменной в правой части уравнения системы на правую часть уравнения, описывающего заменяемую переменную, решения не теряются.

Утверждение 14. (Все свободные решения системы уравнений являются полными решениями). Пусть $A=(Q, F, \mu)$ – алгебра (25) и $s \in \text{Systems}(F, V)$ – система уравнений (21). Тогда любое свободное решение $\varphi: V \rightarrow \text{Terms}(F, V)$ ($\in \langle \text{Free} \rangle_{\text{Solutions}} A(s)$) (32) системы s над A является полным решением s над A (32). Без
доказательства

2. Функциональные сети

2.1. СИНТАКСИС

Определение 33. Функциональной сетью называется четверка $G = (V, R, F, \text{functor})$, где

- V – множество вершин;
- F – множество функциональных символов (15), такое, что $V \cap F = \emptyset$;
- $\text{functor}: V \rightarrow F \cup V$ – такое отображение, что для любой вершины $x \in V$ из $\text{functor}(x) \in V$ следует $\text{functor}(x) = x$;
- $R \subseteq V \times \{1, 2, \dots\} \times V$ – множество дуг, помеченных натуральными числами, такое, что для любой вершины $x \in V$ выполняются следующие условия:

- Если $\text{functor}(x) \in F$, то $\{(i, y) : (x, i, y) \in R\} \in (\{1, \dots, \text{arn}(\text{functor}(x))\} \rightarrow V)$ (15.1.3.3.1.4);
- Если $\text{functor}(x) = x$, то $\{(i, y) : (x, i, y) \in R\} = \emptyset$.

Таким образом, если из вершины $x \in V$ выходит n дуг, то вершине соответствует функциональный блок f с n входами ($\text{functor}(x) = f$, $f \in F$ и $\text{arn}(f) = n$), у которого все входы пронумерованы. Если же вершина x является висячей (нет выходящих из x дуг), то вершине соответствует функциональный блок, не

имеющий входов (генератор постоянного сигнала) либо входной сигнал (в случае, когда $functor(x)=x$).

Функциональная сеть (в дальнейшем сеть) $(V, R, F, functor)$ называется конечной, если V – конечное множество. Везде в дальнейшем под термином сеть будет подразумеваться конечная сеть.

Пусть W множество, в дальнейшем множество всех таких конечных сетей $G = (V, R, F', functor)$, что $V \subseteq W$ и $F' \subseteq F$, будем обозначать $netwares(W, F)$.

Если $G = (V, R, F', functor) \in netwares(W, F)$, то $G.nodes$ является обозначением для V , $G.arcs$ является обозначением для R и $G.functor$ обозначением для $functor$.

Определение 34. Пусть $G = (V, R, F, functor)$ – функциональная сеть (/33/), $x \in V$, $functor(x) \neq x$ и $arn(functor(x)) > 0$. Тогда для $i \in \{1, \dots, arn(functor(x))\}$ $x[i]$ – это такая однозначно определенная вершина, что $(x, i, x[i]) \in R$. В случае, когда $functor(x) = x$, либо когда $functor(x) \neq x$ и $i > arn(functor(x))$, значение $x[i]$ полагается равным \emptyset .

Таким образом, если x – функциональный блок в сети, то $x[i]$ – это тот блок, к которому в сети подсоединен i -ый вход блока x , либо \emptyset , в случае если у x нет входа с номером i .

В следующем определении уточняется, что такое характеристическая система уравнений для функциональной схемы.

Определение 35. Характеристической системой уравнений для сети $N = (V, R, F, functor) \in netwares(W, F)$ (/33/) называется система уравнений $system(N) \in Systems(F, V)$ (/21/) такая, что $system(N) \stackrel{def}{=} \{x = \{f\} \wedge \pi_x :$

$$x \in V \& f = functor(x) \& f \neq x \& \pi_x = \{(i, y) : (x, i, y) \in R\}$$

Пояснение. В силу определения /33/

$\pi_x = \{(i, y) : (x, i, y) \in R\} \in V^{arn(f)}$ (/1.4.15/), а значит определение корректно. Здесь существенным образом используется определение /1.3.3/ понятия отображения. На содержательном уровне $system(N) = \{x = f(x[1], \dots, x[arn(f)]) : x \in V \& f = functor(x) \& f \neq x\}$

2.2. ПЛАН ВЫЧИСЛЕНИЯ И ПОДСТАНОВКА ВЫЧИСЛЕНИЯ

Определение 36. Пусть $N = (V, R, F, functor)$ - функциональная сеть (/33/). Произвольную последовательность $\pi \in (\{1, 2, \dots\} \rightarrow 2^V \setminus \{\emptyset\}) \cup (2^V \setminus \{\emptyset\})^*$ (конечную при $\pi \in (2^V \setminus \{\emptyset\})^*$, либо бесконечную при $\pi \in \{1, 2, \dots\} \rightarrow 2^V \setminus \{\emptyset\}$) будем называть планом настройки сети N . Множество всех конечных планов настройки N (множество всех строк над $2^V \setminus \{\emptyset\}$) будем обозначать $\langle^{Computer}_{Plans}\rangle(N)$, а множество всех планов настройки N будем обозначать $\langle^{AllComputer}_{Plans}\rangle(N)$.

Определение 37. Пусть $N = (V, R, F, functor)$ - функциональная сеть (/33/) и $s \subseteq V$. Итерационной s -подстановкой (обозначается $iteration_{N,s}$) называется такая подстановка $\varphi \in substitutions(F, V)$ (/18/), что

$$kernel(\varphi) = \{x : x \in s \& functor(x) \neq x\} \text{ (/18/) и}$$

$$\text{для любого } x \in kernel(\varphi) \ x = \varphi(x) \in system(N) \text{ (/35/).}$$

Определение 38. Пусть $N = (V, R, F, functor)$ - функциональная сеть (/33/). Определим отображение

$\langle^{Computer}_{Sub}\rangle_N : \langle^{Computer}_{Plans}\rangle(N) \rightarrow substitutions(F, V)$ (/36,18/) во множестве подстановок следующим образом. Для любой строки

$$\alpha \in \langle^{Computer}_{Plans}\rangle(N) \text{ (/36/)}$$

• если $\alpha = \varepsilon$ (/1.4/), то $\langle^{Computer}_{Sub}\rangle_N(\alpha)$ - тождественная подстановка ($kernel(\langle^{Computer}_{Sub}\rangle_N(\varepsilon)) = \emptyset$ (/18/)).

• если $\alpha \neq \varepsilon$ и $\alpha = \alpha(1)^\wedge \alpha'$ (/1.4/), то

$$\langle^{Computer}_{Sub}\rangle_N(\alpha) = iteration_{N,\alpha(1)} \langle^{Computer}_{Sub}\rangle_N(\alpha') \text{ (/18|Обозначения,37:/)}$$

$$\text{Таким образом } \langle^{Computer}_{Sub}\rangle_N(\alpha) = iteration_{N,\alpha(1)} (\langle^{Computer}_{Sub}\rangle_N(\alpha')) =$$

$$\langle^{Computer}_{Sub}\rangle_N(\alpha') \bullet iteration_{N,\alpha(1)}.$$

Множеством подстановок, порождаемых сетью N , называется

$$\text{множество } \langle^{NetWork}_{Subs}\rangle(N) \stackrel{def}{=} \{ \langle^{Computer}_{Sub}\rangle_N(\alpha) : \alpha \in \langle^{Computer}_{Plans}\rangle(N) \}.$$

Утверждение 15. Пусть $N = (V, R, F, functor)$ - функциональная сеть (/33/). Тогда $\langle^{NetWork}_{Subs}\rangle(N) = \langle^{system}_{Subs}\rangle(system(N))$.

Без
доказательства

2.3. СЕМАНТИКА

Определение 39. Говорят, что функциональная сеть $N = (V, R, F, functor)$ (/33/) задана над алгеброй $A = (Q, F', \mu)$ (/25/), если $F \subseteq F'$ и $V \cap (F \cup Q) = \emptyset$.

Определение 40. Пусть $N = (V, R, F, functor)$ – функциональная сеть (/33/) над алгеброй $A = (Q, F, \mu)$ (/25,39/). Для любого множества $s \subseteq V$ на множестве $States(A, N)$ (в соответствии с /2/ $States(A, N) = N.nodes \rightarrow Q$ /33/) определим отображение $step_{A,N,s} : States(A, N) \rightarrow States(A, N)$ следующим образом: для любого состояния $g : V \rightarrow Q$ $step_{A,N,s}(g)$ это такое отображение $g' : V \rightarrow Q$, что

для любого $x \in (V \setminus s) \cup \{y : y \in V \& functor(y) = y\}$
 $g'(x) = g(x)$,

а для любого $x \in s \setminus \{y : y \in V \& functor(y) = y\}$ и для последовательности $\pi \in Q^{arn(functor(x))}$ такой, что $\pi(i) = g(x[i])$, для всех $i \in dom(\pi)$, выполняется $g'(x) = \mu(functor(x))(\pi)$.

Замечание. В частности, для любых $g \in States(A, N)$ и $x \in N.nodes$ $\langle \overset{node}{step} \rangle_x(g) = step_{A,N,\{x\}}(g)$ (/3,40/).

Определение 41. Пусть $N = (V, R, F, functor)$ – функциональная сеть (/33/) над алгеброй $A = (Q, F, \mu)$ (/25,39/). Определим отображение $\langle \overset{effective}{Plans} \rangle_{A,N} : States(A, N) \rightarrow 2^{\langle \overset{Computer}{Plans} \rangle(N)}$ следующим образом.

Для любого $f \in States(A, N)$ и $\pi \in \langle \overset{Computer}{Plans} \rangle(N)$ $\pi \in \langle \overset{effective}{Plans} \rangle_{A,N}(f)$ тогда и только тогда, когда состояние $\langle \overset{Computer}{Result} \rangle_{A,N}(f, \pi)$ является настроенным (/4/: Планы настройки из множества $\langle \overset{effective}{Plans} \rangle_{A,N}(f)$ будут планами полной настройки для состояния f).

Настроечная последовательность π называется полной, если в ней есть согласованное состояние (если существует $i \in dom(\pi)$, такое, что $\pi(i)$ - согласованное состояние). Настроечная последовательность называется избыточной, если в ней присутствует более одного согласованного состояния.

Утверждение 16. Пусть $N = (V, R, F, functor)$ – функциональная сеть ((33)) над алгеброй $A = (Q, F, \mu)$ ((25,39)) и $f \in States(A, N)$. Тогда вершина $x \in V$ настроена в состоянии f в том и только в том случае, когда выполняется ровно одно из следующих условий:

- Либо $functor(x) = x$
- Либо $functor(x) = g \in F$ и отображение $f: V \rightarrow Q$ является решением системы (из одного уравнения) $\{x = \{g\}^{\wedge} \pi\}$, где $\pi \in V^{arn(g)}$ и для любого $i \in dom(\pi)$ $\pi(i) = x[i]$. Без доказательства

Следствие 1. Пусть N – функциональная сеть ((33)) над алгеброй A ((40)). Тогда состояние $f \in States(A, N)$ ((2)) является настроенным ((4)) в том и только в том случае, когда $f \in solutions(system(N), A)$. Без доказательства

Определение 42. Пусть N функциональная сеть ((33)) над алгеброй $A = (Q, F, \mu)$ ((40)). Определим отображения

$$\langle \begin{matrix} Computer \\ Protocol \end{matrix} \rangle_{A,N} : States(A, N) \times \langle \begin{matrix} AllComputer \\ Plans \end{matrix} \rangle(N) \rightarrow States(A, N)^* \text{ и}$$

$$\langle \begin{matrix} Computer \\ Result \end{matrix} \rangle_{A,N} : States(A, N) \times \langle \begin{matrix} Computer \\ Plans \end{matrix} \rangle(N) \rightarrow States(A, N)$$

следующим образом:

- Для любых $f \in States(A, N)$ и $\tau \in \langle \begin{matrix} AllComputer \\ Plans \end{matrix} \rangle(N)$ $\langle \begin{matrix} Computer \\ Protocol \end{matrix} \rangle_{A,N}(f, \tau)$ это такая последовательность π элементов из $States(A, N)$, которая является последовательностью минимальной длины, из всех последовательностей, удовлетворяющих следующим условиям:
 - $1 \in dom(\pi)$ и $\pi(1) = f$;
 - если $i \in dom(\tau) \cap dom(\pi)$, то $i + 1 \in dom(\pi)$ и при этом $\pi(i + 1) = step_{A,N,\tau(i)}(\pi(i))$ ((40)).

- Для любого $\tau \in \langle \begin{matrix} Computer \\ Plans \end{matrix} \rangle(N)$

$$\langle \begin{matrix} Computer \\ Result \end{matrix} \rangle_{A,N}(f, \tau) \stackrel{def}{=} last(\langle \begin{matrix} Computer \\ Protocol \end{matrix} \rangle(f, \tau)) \text{ ((1.4)).}$$

Последовательность состояний

$$\pi \in (\{1, 2, \dots\} \rightarrow States(A, N)) \cup (States(A, N)^*) \text{ ((40))}$$

(конечная, при $\pi \in States(A, N)^*$, либо бесконечная) называется настроенной последовательностью, если существует план настройки

$\tau \in \langle \text{AllComputer} \rangle_{\text{Plans}}(N)$ такой, что $\pi = \langle \text{Computer} \rangle_{\text{Protocol}}(\pi(1), \tau)$.

Определение 43. Пусть N сеть (43/) над алгеброй A (40/). Множеством надежных планов настройки сети N в алгебре A называется такое множество $\langle \text{SaFe} \rangle_A(N) \subseteq \langle \text{Computer} \rangle_{\text{Plans}}(N)$, что для любых $\pi \in \langle \text{SaFe} \rangle_A(N)$ и $f \in \text{States}(A, N)$ $\pi \in \langle \text{effective} \rangle_{\text{Plans}}(f)$.

Утверждение 17. Пусть $G = (V, R, F, \text{functor})$ – функциональная сеть (43/), $A = (Q, F, \mu)$ – алгебра (25/) и $s \subseteq V$. Тогда для любого состояния $f \in \text{States}(A, G)$ выполняется тождество $(\text{step}_{A,G,s}(f))^{Terms(F \cup Q)} \equiv_A f^{Terms(F \cup Q)}(\text{iteration}_{N,s})$ (19/)
(подстановка $\text{iteration}_{N,s}$ определена в 37/).

Доказательство. Пусть $x \in V$ и $\varphi = f^{Terms(F \cup Q)}$ (19/):
 $\varphi \in \text{substitutions}(F \cup Q, V)$). Достаточно доказать, что $\text{step}_{A,G,s}(f)(x) = \text{value}_A(\varphi(\text{iteration}_{N,s})(x))$ (29,40,19,37/). Возможны следующие варианты: $x \notin s$ и $x \in s$.

Пусть $x \notin s$, либо $\text{functor}(x) = x$. Тогда $\text{step}_{A,G,s}(f)(x) = f(x)$ (40/) (по определению 40/ $\text{step}_{A,G,s}$). В то же время, поскольку $x \notin \text{kernel}(\varphi)$, то $\varphi(\text{iteration}_{N,s})(x) = \varphi(\text{iteration}_{N,s}(x)) = \varphi(x)$ (37/),

а значит, $\varphi(\text{iteration}_{N,s})(x) = f^{Terms(F \cup Q)}(x) = f(x)$.

Пусть $x \in s$, $\text{functor}(x) = g$ и $g \neq x$.

Возможны следующие варианты

- 1) $\text{arn}(g) = 0$,
- 2) $\text{arn}(g) = n > 0$

В первом случае $\varphi(\text{iteration}_{N,s}(x)) = \varphi(g) = g$ (37/), а значит $\text{value}_A(\varphi(\text{iteration}_{N,s})(x)) = \mu(g)(\varepsilon)$ (29,37/). В то же время $\text{step}_{A,G,s}(f)(x) = \mu(g)(\varepsilon)$ (40/), по определению $\text{step}_{A,G,s}$ (40/).

Во втором случае $\text{iteration}_{N,s}(x) = g(x[1], \dots, x[n])$ (37,34/), а значит

$\text{value}_A(\varphi(\text{iteration}_{N,s})(x)) =$ (29,37/) $\text{value}_A(\varphi(\text{iteration}_{G,s}(x))) =$
 $\text{Interpretation}_A(g(\varphi(x[1]), \dots, \varphi(x[n]))) =$ (27,37,34/)
 $\mu(g)(\text{Interpretation}_A(\varphi(x[1])), \dots, \text{Interpretation}_A(\varphi(x[n]))) =$
 $\text{step}_{A,G,s}(f)(x)$ (27, 34, 40/). Конец
доказательства

Следствие 1. Пусть $N = (V, R, F, functor)$ – функциональная сеть (/33/) над алгеброй $A = (Q, F, \mu)$ (/25,40/) и $s \subseteq V$. Тогда для любого состояния $f \in States(A, N)$ $step_{A,N,s}(f) = \langle \substack{subs \\ Value} \rangle_A (f^{Terms(F \cup Q)}(iteration_{N,s}))$ (/29,19/). Без
доказательства

Теорема 2. Пусть $N = (V, R, F, functor)$ – функциональная сеть (/33/) над алгеброй $A = (Q, F, \mu)$ (/25,40/) и $\pi \in \langle \substack{Computer \\ Plans} \rangle(N)$. Тогда для любого состояния $f \in States(A, N)$ (/40/) $(\langle \substack{Computer \\ Result} \rangle_{A,N}(f, \pi))^{Terms(F \cup Q)} \equiv_A f^{Terms(F \cup Q)}(\langle \substack{Computer \\ Sub} \rangle_N(\pi))$ (/19, 42, 30, 38/).

Доказательство. Проведем индукцию по длине π .

Базис: $\pi = \varepsilon$ (/1.4/). В этом случае утверждение верно, поскольку $\langle \substack{Computer \\ Result} \rangle_{A,N}(f, \pi) = f$ (/42/), а $\langle \substack{Computer \\ Sub} \rangle_N(\pi)$ (/38/) – тождественная подстановка.

Шаг индукции. Пусть $\pi = \pi(1) \wedge \pi'$ (/1.4/) и пусть

$\alpha = \langle \substack{Computer \\ Protocol} \rangle_{A,N}(f, \pi)$ (/42/). Поскольку $\pi \neq \varepsilon$, то $length(\alpha) > 1$ (/1.4/).

По предположению индукции

$$\langle \substack{Computer \\ Result} \rangle_{A,N}(\alpha(2), \pi')^{Terms(F \cup Q)} \equiv_A \alpha(2)^{Terms(F \cup Q)}(\langle \substack{Computer \\ Sub} \rangle_N(\pi')) \quad (/19,42, 30, 38/).$$

При этом, поскольку $\langle \substack{Computer \\ Result} \rangle_{A,N}(\alpha(2), \pi') = \langle \substack{Computer \\ Result} \rangle_{A,N}(\alpha(1), \pi) = \langle \substack{Computer \\ Result} \rangle_{A,N}(f, \pi)$ (/42/), то $\langle \substack{Computer \\ Result} \rangle_{A,N}(f, \pi)^{Terms(F \cup Q)} \equiv_A \alpha(2)^{Terms(F \cup Q)}(\langle \substack{Computer \\ Sub} \rangle_N(\pi'))$ (/19,42, 30, 38/).

Таким образом, достаточно доказать соотношение

$$\alpha(2)^{Terms(F \cup Q)}(\langle \substack{Computer \\ Sub} \rangle_N(\pi')) \equiv_A (f^{Terms(F \cup Q)})(\langle \substack{Computer \\ Sub} \rangle_N(\pi)) \quad (/19, 38, 30/).$$

Действительно, в силу утверждения 17

$$\alpha(2)^{Terms(F \cup Q)} \equiv_A f^{Terms(F \cup Q)}(iteration_{N,\pi(1)}) \quad (/19, 30, 37/).$$

Таким образом, $\alpha(2)^{Terms(F \cup Q)}(\langle \substack{Computer \\ Sub} \rangle_N(\pi')) \equiv_A f^{Terms(F \cup Q)}(iteration_{N,\pi(1)})\langle \substack{Computer \\ Sub} \rangle_N(\pi')$.

Но по определению /38/

$$\langle \substack{Computer \\ Sub} \rangle_N(iteration_{N,\pi(1)})\langle \substack{Computer \\ Sub} \rangle_N(\pi') = \langle \substack{Computer \\ Sub} \rangle_N(\pi) \quad (/37,38/).$$

Конец
доказательства

Следствие 1. Пусть $G = (V, R, F, functor)$ – функциональная сеть (133), $A = (Q, F, \mu)$ – алгебра (125), $f \in States(A, G)$ – состояние сети (140) и $\pi \in \langle \overset{Computer}{Plans} \rangle(N)$ план вычислений. Тогда $\langle \overset{Computer}{Result} \rangle_{A,N}(f, \pi) = \langle \overset{subs}{Value} \rangle_A(f \overset{Terms}{FUQ} \langle \overset{Computer}{Sub} \rangle_N(\pi))$.
Без доказательства (переформулировка теоремы 2).

Следствие 2. Пусть $N = (V, R, F, functor)$ – сеть (133), $A = (Q, F, \mu)$ – алгебра (125) и $\pi \in \langle \overset{Computer}{Plans} \rangle(N)$ (136). Тогда следующие утверждения равносильны друг другу.

- 1) $\pi \in \langle \overset{Safe}{Plans} \rangle_A(N)$ (143)
- 2) *Отображение $\langle \overset{Computer}{Sub} \rangle_N(\pi)|_V$ (138,18) является свободным (а значит полным) решением (132) $system(N)$ (135) над A*
- 3) *Для любой подстановки $\varphi \in \langle \overset{NetWork}{Subs} \rangle(N)$ (138)*
 $\langle \overset{Computer}{Sub} \rangle_N(\pi)\varphi \equiv A \langle \overset{Computer}{Sub} \rangle_N(\pi)$. *Без доказательства*

3. Обоснование стратегии Киссинджера

В следующем определении вводится обозначение $t^{x,n}$ для терма (116) $\{x \rightarrow t\}^n(x)$ (в соответствии с 1.3.3|Обозначения/ $\varphi^n = \varphi^{[n]}$ является обозначением для подстановки $\underbrace{\varphi \varphi \dots \varphi}_{n \text{ раз}}$).

Определение 44. В дальнейшем запись $t^{x,n}$, где $x \in V$ – переменная, $t \in Terms(F, V)$ – терм (116), а $n \in \{0, 1, \dots\}$, будет использоваться для обозначения терма, представляемого записью $\{x \rightarrow t\}^n(x)$ (18, 1.3.3|Обозначения, 18|Обозначения/: $\{x \rightarrow t\}$ – подстановка, переводящая x в t и оставляющая остальные переменные на месте).

Примеры.

- 1) Если $x \notin vars(t)$ (117) если терм t не содержит переменной x), то при всех $n > 0$ $t^{x,n} = t$ (144).
- 2) Если $t=x$ (если терм является переменной), то при всех $n \geq 0$ $t^{x,n} = x^{x,n} = x$ (144).

Определение 45. Пусть $A = (Q, F, \mu)$ конечная алгебра (/25/). Будем говорить, A – комбинаторная алгебра, если для любого термина $t \in Terms(F, \{x\})$, где $x \notin F \cup Q$ и любого $q \in Q$ существует такое $n \in \{1, 2, \dots\}$, что $\{x \rightarrow q\} t^{x,n} \equiv_A \{x \rightarrow q\} t^{x,n+1}$.

В следующем определении (фактически оно повторяет определение 6 стратегии Киссинджера) вводятся термины и обозначения, используемые при доказательстве.

Определение 46. Пусть $N = (V, R, F, functor)$ конечная функциональная сеть (/33/) над алгеброй A (/40/) и $\ll \in \langle \text{Orders} \rangle(V)$ линейный порядок (/1.3.2/) на множестве вершин V .

Для любого не настроенного состояния $q \in States(A, N)$ (/4,40/) главной не настроенной вершиной (обозначается $\langle \text{chief}_{node} \rangle_{N, A}(\ll, q)$) будем называть наименьшую по порядку \ll не настроенную вершину из множества V (/4/). В тех случаях, когда N и A ясны из контекста будем пользоваться обозначением $\langle \text{chief}_{node} \rangle(\ll, f)$ вместо $\langle \text{chief}_{node} \rangle_{N, A}(\ll, f)$.

Пусть $f \in States(A, N)$. План вычислений $\pi \in \langle \text{AllComputerPlans} \rangle(N)$ называется \ll, f -планом Киссинджера, если π является такой последовательностью минимальной длины из множества $States(A, N)^* \cup (\{1, 2, \dots\} \rightarrow States(A, N))$, что выполняются следующие условия

- если f не настроенное состояние, то $1 \in dom(\pi)$ и $\pi(1) = \{\langle \text{chief}_{node} \rangle(\ll, f)\}$ (/1.3.3,1.4/);
- если $i \in dom(\pi)$ и $g = \langle \text{ComputerResult} \rangle_{A, N}(f, \pi|_{\{1, \dots, i\}})$ не настроенное состояние (/4/), то $i+1 \in dom(\pi)$ (/1.3.3/) и $\pi(i+1) = \{\langle \text{chief}_{node} \rangle(\ll, g)\}$ (/40,46/)

Если последовательность $\pi \in \langle \text{AllComputerPlans} \rangle(N)$ является \ll, f -планом Киссинджера, то последовательность состояний $\langle \text{ComputerProtocol} \rangle_{A, N}(f, \pi)$ (/42/) называется (f, \ll) -настройкой сети N .

3.1. РЕГУЛЯРНАЯ УСТОЙЧИВОСТЬ КОМБИНАТОРНЫХ АЛГЕБР

Ближайшая цель – доказать утверждение 25.следствие 2 о том, что любая конечная комбинаторная алгебра (/45/) регулярно

Единственным трудным промежуточным утверждением в приводимом далее доказательстве утверждения 21, является интуитивно очевидное утверждение 19.следствие 2. Все предшествующие 19.следствие 2 утверждения служат как промежуточный материал для доказательства 19.следствие 2.

На втором этапе (заканчивается утверждением 25.следствие 2)

Доказывается, что при настройках сетей над конечными комбинаторными алгебрами, настройка по стратегии Киссинджера приводит ровно к тому же результату, что и настройка в соответствии с тем планом настройки, который соответствует подстановке $\langle_{Subs}^{Key} \rangle_A(N, \ll)$ (/49/).

Этот план, обозначаемый $\langle_{Plan}^{Key} \rangle_A(N, \ll)$ явным образом описан в определении /51/ (и устроен так, что $\langle_{Subs}^{Key} \rangle_A(N, \ll) = \langle_{Sub}^{Computer} \rangle_N(\langle_{Plan}^{Key} \rangle_A(N, \ll))$).

Поэтому, поскольку $\langle_{Subs}^{Key} \rangle_A(N, \ll)$ задает параметрическое решение план $\langle_{Plan}^{Key} \rangle_A(N, \ll)$ всегда приводит к согласованному состоянию (об этом лишней раз напоминается в тривиальном утверждении 22.следствие 1).

Идея доказательства сводится к следующему:

Если из плана настройки $\langle_{Plan}^{Key} \rangle_A(N, \ll)$ (/51/) сети находящейся в начальном состоянии f выбросить все «бесполезные шаги» (шаги, не изменяющие текущих состояний), то получится \ll, f -план Киссинджера (/46/).

«Бесполезные шаги» определены в /52/: это те номера шагов, которые не входят во множество $\langle_{Set}^{nonRedundant} \rangle_{A, N, \ll}(f)$ (где f – начальное состояние сети N над алгеброй A)

Таким образом, \ll, f -план Киссинджера (/46/) является подпоследовательностью плана $\langle_{Plan}^{Key} \rangle_A(N, \ll)$ (/51/), а протокол настройки Киссинджера (f, \ll)-настройка (/46/) является подпоследовательностью последовательности $\langle_{Protocol}^{Computer} \rangle(f, \langle_{Plan}^{Key} \rangle_A(N, \ll))$.

Доказательство основано на утверждении 23.следствие 1 (все остальные утверждения безыдейны и очевидны, а их доказательства не следует рассматривать иначе как проверку корректности используемых определений)

Первый этап При обосновании будут проводиться некоторые формальные выкладки с композициями подстановок вида $\{x \rightarrow t\}$, $\{x \rightarrow t^{x,n}\}$, $\{x \rightarrow t\}^n$ и т.д. Основной принцип доказательств равенств подобных композиций состоит в следующем. Если φ и φ' две подстановки такие, что $kernel(\varphi) \subseteq \{x\}$ и $kernel(\varphi') \subseteq \{x\}$, то $\varphi = \varphi'$ ($\varphi \equiv_A \varphi'$) тогда и только тогда, когда $\varphi(x) = \varphi'(x)$ ($\varphi(x) \equiv_A \varphi'(x)$). Более общая форма этого приема лежит в определении подстановок и сформулирована в утверждении 8 (определении /30/): две подстановки φ, φ' равны (эквивалентны) тогда и только тогда, когда для любой переменной x равны (эквивалентны) значения $\varphi(x)$ и $\varphi'(x)$.

Утверждение 18. Пусть $t \in Terms(F, V)$ и $x \in V$. Тогда для любых $n, m \in \{0, 1, \dots\}$ $\{x \rightarrow t^{x,n}\} = \{x \rightarrow t\}^n$ (/18|Обозначения, 44/) и $\{x \rightarrow t^{x,m}\} \{x \rightarrow t^{x,n}\} = \{x \rightarrow t^{x,n+m}\}$.

Доказательство. Докажем равенство $\{x \rightarrow t^{x,n}\} = \{x \rightarrow t\}^n$ (/18|Обозначения, 44/): поскольку x единственная переменная которая может измениться в этих подстановках, то (утв. /8/) достаточно доказать, что $\{x \rightarrow t^{x,n}\}(x) = \{x \rightarrow t\}^n(x)$.

$\{x \rightarrow t^{x,n}\}(x) = t^{x,n}$ (поскольку $\{x \rightarrow \tau\}$ (/18|Обозначения/) обозначение такой подстановки φ , что $\varphi(x) = \tau$)

$\{x \rightarrow t\}^n(x) = t^{x,n}$ (по определению /44/ $t^{x,n}$). Конец доказательства.

Определение 47. Пусть $A = (Q, F, \mu)$ – алгебра (/25/) и $\varphi \in substitutions(F \cup Q, V)$ (/18/) – подстановка. Будем говорить, что подстановка $\varphi \in substitutions(F, V)$ (/18/) устойчива над A , если справедливо тождество $\varphi \equiv_A \varphi^2$ (/30/, если $\varphi \equiv_A \varphi(\varphi)$).

Устойчивые подстановки (/47/) будут играть ключевую роль при доказательстве утверждения 25. следствие 2.

Утверждение 19. Пусть $A = (Q, F, \mu)$ – алгебра (/25/) и $\varphi \in substitutions(\overline{F} \cup Q, V)$ (/18/) – устойчивая в A подстановка (/47/). Тогда для любых $x \in V \setminus kernel(\varphi)$ (/18/) и $t \in Terms(F, V)$ (/16/) $\varphi\{x \rightarrow t\}\varphi \equiv_A \{x \rightarrow \varphi(t)\}\varphi$ (/18,18|Обозначения, 30/).

Доказательство. Поскольку φ - устойчивая подстановка, то доказываемое тождество эквивалентно тождеству

$$\varphi\{x \rightarrow t\} \varphi \equiv_A \{x \rightarrow \varphi(t)\} \varphi$$

В силу определения /30/ отношения \equiv_A , достаточно доказать, что для любой переменной $y \in V$

$$(\varphi\{x \rightarrow t\})\varphi(y) \equiv_A (\{x \rightarrow \varphi(t)\}\varphi)\varphi(y) \text{ (/30/). Пусть } \tau = \varphi(y)$$

Докажем равенство $(\varphi\{x \rightarrow t\})(\tau) = \{x \rightarrow \varphi(t)\}\varphi(\tau)$.

Согласно утверждению 8.следствие 1, для доказательства равенства достаточно для каждой переменной $z \in vars(\tau)$ доказать равенство $(\varphi\{x \rightarrow t\})(z) = \{x \rightarrow \varphi(t)\}\varphi(z)$.

Пусть $z \in vars(\tau)$. Возможны варианты 1) $x \neq z$ и 2) $x = z$.

1. Если $z \neq x$, то $(\varphi\{x \rightarrow t\})(z) = \varphi(z)$ и $\{x \rightarrow \varphi(t)\}\varphi(z) = \varphi(z)$.

2. При $x = z$ $(\varphi\{x \rightarrow t\})(z) = (\varphi\{x \rightarrow t\})(x) = \varphi(t)$ и

$$\{x \rightarrow \varphi(t)\}\varphi(z) = \{x \rightarrow \varphi(t)\}\varphi(x) = \{x \rightarrow \varphi(t)\}(x) = \varphi(t). \quad \text{Конец доказательства}$$

Следствие 1. Пусть $A = (Q, F, \mu)$ - алгебра (/25/)

и $\varphi \in substitutions(F \cup Q, V)$ (/18/) - устойчивая в A подстановка (/47/). Тогда для любых $x \in V \setminus kernel(\varphi)$ (/18/), $t \in Terms(F, V)$ (/16/) и $n \in \{0, 1, \dots\}$ выполняются тождества $\{x \rightarrow \varphi(t)^{x,n}\}(\varphi) \equiv_A (\varphi\{x \rightarrow t\})^n \varphi \equiv_A \{x \rightarrow \varphi(t)^{x,n}\}\varphi$ (/18,44,18|Обозначения,1.3.3|Обозначения/).

Доказательство.

Проведем индукцию по n :

Базис тривиален: $(\varphi(\{x \rightarrow t\}))^0 \varphi = \varphi$ и

$$\{x \rightarrow \varphi(t)^{x,0}\} \varphi = \{x \rightarrow x\} \varphi = \varphi.$$

Шаг индукции. Пусть $n > 0$, тогда $(\varphi(\{x \rightarrow t\}))^n \varphi =$

$$(\varphi(\{x \rightarrow t\}))^{n-1} \varphi \{x \rightarrow t\} \varphi \equiv_A \text{/*индуктивное предположение*/}$$

$$\{x \rightarrow \varphi(t)^{x,n-1}\} \varphi \{x \rightarrow t\} \varphi \equiv_A \text{/*утв. 19*/}$$

$$\{x \rightarrow \varphi(t)^{x,n-1}\} \{x \rightarrow \varphi(t)\} \varphi = \text{/* утв. 18*/} \{x \rightarrow \varphi(t)^{x,n}\} \varphi. \quad \text{Конец доказательства}$$

Следствие 2. Пусть $A = (Q, F, \mu)$ - алгебра (/25/)

и $\varphi \in substitutions(F \cup Q, V)$ (/18/) - устойчивая в A подстановка (/47/). Тогда для любых $x \in V \setminus kernel(\varphi)$ (/18/), $t \in Terms(F, V)$ (/16/) и $n \in \{0, 1, \dots\}$ выполняется тождество $(\varphi(\{x \rightarrow t\}))^n(x) \equiv_A \varphi(t)^{x,n}$ (/44,18|Обозначения,1.3.3|Обозначения/).

Доказательство.

$$\begin{aligned}
 (\varphi(\{x \rightarrow t\})^n(x) = /*\varphi(x) = x*/(\varphi(\{x \rightarrow t\}))^n \varphi(x) \equiv_A /*\text{утв.19.следствие 1}*/ \\
 \{x \rightarrow \varphi(t)^{x,n}\} \varphi(x) = /*\varphi(x) = x*/ \\
 \{x \rightarrow \varphi(t)^{x,n}\}(x) = \varphi(t)^{x,n}. \quad \text{Конец доказательства}
 \end{aligned}$$

Утверждение 20. Пусть $A = (Q, F, \mu)$ – конечная комбинаторная алгебра (45). Тогда для любых $t \in \text{Terms}(F, V)$ (16), $x \in V$ и $n \in \{0, 1, \dots\}$ $t^{x,|Q|+n} \equiv_A t^{x,|Q|}$ (44,30). Без доказательства

Определение 48. Пусть F – множество функциональных символов (15) и W – множество не пересекающееся с F . Определим отображение $\text{isolate}: \text{netwares}(W, F) \times 2^W \rightarrow \text{netwares}(W, F)$ (33) следующим образом. Для любой сети $N = (V, R, F, \text{functor}) \in \text{netwares}(W, F)$ (33) и любого множества $s \subseteq W$ $\text{isolate}(N, s) = (V, R', F, \text{functor}')$ это такая сеть, что выполняются следующие условия:

- $R' = R \cap ((W \setminus s) \times \{1, 2, \dots\} \times (W \setminus s))$
 - для любого $x \in V$
- $$\text{functor}'(x) = \begin{cases} x, & \text{если } x \in s \\ \text{functor}(x), & \text{если } x \notin s \end{cases}$$

В дальнейшем вместо записи $\text{isolate}(N, s)$ будем использовать запись $N.\text{isolate}(s)$. При этом будем говорить, что сеть $N.\text{isolate}(s)$ получается из сети N изолированием вершин множества s .

Определение 49. Пусть $A = (Q, F, \mu)$ – конечная алгебра (25), $N = (V, R, F, \text{functor})$ – конечная сеть (33) и \ll – линейный порядок на V . Ключом к решению для тройки (A, N, \ll) назовем - подстановку $\langle_{\text{Subs}}^{\text{Key}} \rangle_A(N, \ll) \in \text{substitutions}(F, V)$, которая определяется индукцией по мощности $\langle_{\text{Vars}}^{\text{bound}} \rangle(\text{system}(N))$ следующим образом:

- Если $|\langle_{\text{Vars}}^{\text{bound}} \rangle(\text{system}(N))| = 0$ (если $\text{system}(N) = \emptyset$), то $\langle_{\text{Subs}}^{\text{Key}} \rangle_A(N, \ll) \stackrel{\text{def}}{=} (= V)^{\text{Terms}}$ - тождественная подстанов-

ка $(kernel(\langle \langle \text{Subs} \rangle \rangle_A(N, \ll)) = \emptyset)$

- Если $|\langle \langle \text{Vars} \rangle \rangle^{bound}(system(N))| > 0$, то

$\langle \langle \text{Subs} \rangle \rangle_A(N, \ll) \stackrel{def}{=} (\varphi(iteration_{N, \{x\}}))^{|Q|} \varphi$, где
 $x = \max_{\ll} (\langle \langle \text{Vars} \rangle \rangle^{bound}(system(N)))$ и $\varphi = \langle \langle \text{Subs} \rangle \rangle_A(N.isolate(\{x\}), \ll)$.

Утверждение 21. Пусть $A = (Q, F, \mu)$ – комбинаторная алгебра (45), $N = (V, R, F, functor)$ - конечная сеть (33) и \ll – линейный порядок на V .

Тогда $\langle \langle \text{Subs} \rangle \rangle_A(N, \ll)|_V$ свободное решение (32) $system(N)$ над A .

Доказательство. Проведем индукцией по

$|\langle \langle \text{Vars} \rangle \rangle^{bound}(system(N))|$.

Базис: (случай $\langle \langle \text{Vars} \rangle \rangle^{bound}(system(N)) = \emptyset$) тривиален (все элементы $States(A, N)$ являются решениями и тождественная подстановка принадлежит $\langle \langle \text{Subs} \rangle \rangle^{NetWork}(N) = \langle \langle \text{Subs} \rangle \rangle^{system}(system(N))$ (24/ и утв. 15)).

Индуктивный шаг. Пусть $\langle \langle \text{Vars} \rangle \rangle^{bound}(system(N)) \neq \emptyset$. Положим

$x \stackrel{def}{=} \max_{\ll} (\langle \langle \text{Vars} \rangle \rangle^{bound}(system(N)))$, $N' \stackrel{def}{=} N.isolate(\{x\})$,

$\theta_x \stackrel{def}{=} iteration_{N, \{x\}}$, $\tau \stackrel{def}{=} \theta_x(x)$ и $\varphi \stackrel{def}{=} \langle \langle \text{Subs} \rangle \rangle_A(N', \ll)$.

Тогда $system(N') = system(N) \setminus \{x = \tau\}$ и $\{x = \tau\} \in system(N)$.

Поскольку $\varphi \in \langle \langle \text{Subs} \rangle \rangle^{NetWork}(N)$ (по индуктивному предположению) и $iteration_{N, \{x\}} \in \langle \langle \text{Subs} \rangle \rangle^{NetWork}(N)$ (38,24), то $\langle \langle \text{Subs} \rangle \rangle_A(N, \ll) \in \langle \langle \text{Subs} \rangle \rangle^{system}(system(N))$ (24, 38/ и утв.15). Следовательно, для доказательства утверждения достаточно убедиться в том, что $\langle \langle \text{Subs} \rangle \rangle_A(N, \ll)|_V = ((\varphi \theta_x)^{|Q|} \varphi)|_V$ является параметрическим решением (32) $system(N)$ в A . Для этого достаточно доказать, что для любого уравнения $y = t \in system(N)$ $(\varphi \theta_x)^{|Q|} \varphi(y) \equiv_A (\varphi \theta_x)^{|Q|} \varphi(t)$.

Возможны следующие варианты $y = x$ и $y \neq x$.

Пусть $y = x$. Тогда $t = \tau$. Необходимо доказать тождество

$t_{leftpart} = (\varphi \theta_x)^{|Q|} \varphi(x) \equiv_A (\varphi \theta_x)^{|Q|} \varphi(\tau) = t_{rightpart}$.

Поскольку (по индуктивному предположению) $\varphi|_V$ – свободное решение $system(N')$, то (теорема 2.следствие 2 пункты 2 и 3) $\varphi \varphi \equiv_A \varphi$, следовательно φ – устойчивая подстановка.

Учитывая устойчивость φ , преобразуем терм $t_{rightpart} =$

$(\varphi \theta_x)^{|Q|} \varphi(\tau)$ следующим образом:

$$\begin{aligned} t_{rightpart} &= (\varphi \theta_x)^{|Q|} \varphi(\tau) = /*\theta_x = \{x \rightarrow \tau\}*/ \\ (\varphi \{x \rightarrow \tau\})^{|Q|} \varphi(\{x \rightarrow \tau\}(x)) &= ((\varphi \{x \rightarrow \tau\})^{|Q|} \varphi\{x \rightarrow \tau\})(x) = \\ (\varphi \{x \rightarrow \tau\})^{|Q|+1}(x) &\equiv_A /*\text{утв.19.следствие 2}*/ \varphi(t)^{x,|Q|+1}. \end{aligned}$$

Аналогично

$$\begin{aligned} t_{leftpart} &= (\varphi \theta_x)^{|Q|} \varphi(x) = //\varphi(x) = x \\ (\varphi \theta_x)^{|Q|}(x) &\equiv_A /*\text{утв.19.следствие 2}*/ \varphi(t)^{x,|Q|}. \end{aligned}$$

Тождество $\varphi(t)^{x,|Q|} \equiv_A \varphi(t)^{x,|Q|+1}$ справедливо в силу утверждения 20.

Пусть $y \neq x$. Поскольку $\varphi|_V$ параметрическое решение (по предположению индукции), то $\psi\varphi(y) \equiv_A \psi\varphi(t)$ для любой подстановки $\psi \in substitutions(F, V)$ в том числе и для $\psi = (\varphi \theta_x)^{|Q|}$. Конец доказательства

Второй этап

Определение 50. Определим отображение

$\langle Iteration_{String} \rangle: \{1, 2, \dots\} \times \{0, 1, \dots\} \rightarrow \{1, 2, \dots\}^*$ следующим образом для любых $n \in \{1, 2, \dots\}$ и $m \in \{0, 1, \dots\}$ $\langle Iteration_{String} \rangle(n, m) =$

$$\begin{cases} \varepsilon, & \text{если } m = 0 \\ ((\langle Iteration_{String} \rangle(n, m-1) \wedge \{m\})^{\wedge n}) \wedge \langle Iteration_{String} \rangle(n, m-1), & \text{если } m > 0 \end{cases}$$

Если V множество и $f: \{1, \dots, n\} \rightarrow V$ отображение, то строку $[*]^f(\langle Iteration_{String} \rangle(n, m))$ ($//10$) будем обозначать $\langle Iteration_{Protocol} \rangle_f(n, m)$.

Пример 9. $\langle Iteration_{String} \rangle(2, 0) = \varepsilon$

$$\begin{aligned} \langle Iteration_{String} \rangle(2, 1) &= ((\varepsilon \wedge \{1\})^{\wedge 2}) \wedge \varepsilon = \{1\} \wedge \{1\} (=“11”) \\ \langle Iteration_{String} \rangle(2, 2) &= (((\langle Iteration_{String} \rangle(2, 1) \wedge \{2\})^{\wedge 2}) \wedge \langle Iteration_{String} \rangle(2, 1) = \\ &= ((\{1\} \wedge \{1\} \wedge \{2\})^{\wedge 2}) \wedge (\{1\} \wedge \{1\}) = \\ &= \{1\} \wedge \{1\} \wedge \{2\} \wedge \{1\} \wedge \{1\} \wedge \{2\} \wedge \{1\} \wedge \{1\} (=“11211211”). \bullet \end{aligned}$$

Определение 51. Пусть $N = (V, R, F, functor)$ - конечная сеть ($//33$), $A = (Q, F, \mu)$ - конечная алгебра ($//25$), $<<$ - линейный порядок на V , $s = \langle bound_{Vars} \rangle(system(N))$ и $f: \{1, \dots, |s|\} \rightarrow V^1$ отображение такое, что для любого $i \in dom(f)$ выполняется равенство $f(i) = \{num_{s, <<}^{-1}(i)\}$.

Ключевым планом для тройки (A, N, \ll) назовем последовательность $\langle \text{Plan} \rangle_A(N, \ll) \in \langle \text{Computer} \rangle_{Plans}(N)$, равную последовательности $\langle \text{Iteration} \rangle_{Protocol}^f(|Q|, |s|)$.

Утверждение 22. Пусть $N = (V, R, F, \text{functor})$ - конечная сеть (43), $A = (Q, F, \mu)$ – конечная алгебра (25) и \ll - линейный порядок на V . Тогда

$$\langle \text{Subs} \rangle_A(N, \ll) = \langle \text{Sub} \rangle_{Computer}^N(\langle \text{Plan} \rangle_A(N, \ll)). \quad \text{Без доказательства}$$

Следствие 1. Пусть $A = (Q, F, \mu)$ – комбинаторная алгебра (45), $N = (V, R, F, \text{functor})$ - конечная сеть (33) и \ll - линейный порядок на V . Тогда для любого состояния $f \in \text{States}(A, N)$ состояние $\langle \text{Result} \rangle_{A,N}(f, \langle \text{Plan} \rangle_A(N, \ll))$ является настроенным.

Доказательство. Пусть $\pi = \langle \text{Plan} \rangle_A(N, \ll)$.

$$\text{Поскольку (утв.22)} \quad \langle \text{Subs} \rangle_A(N, \ll) = \langle \text{Sub} \rangle_{Computer}^N(\pi),$$

то (Теорема 2.следствие 1)

$$\langle \text{Computer} \rangle_{A,N}(f, \pi) = \langle \text{Value} \rangle_A(f^{Terms}(F \cup Q) \langle \text{Subs} \rangle_A(N, \ll)).$$

Поскольку (в силу утв.21) $\langle \text{Subs} \rangle_A(N, \ll)|_V$ - параметрическое решение $system(N)$, то (в силу утв.13.следствие 1)

$$\langle \text{Value} \rangle_A(f^{Terms}(F \cup Q) \langle \text{Subs} \rangle_A(N, \ll)) - \text{решение } system(N),$$

а значит $\langle \text{Result} \rangle_{A,N}(f, \pi)$ – согласованное состояние. Конец доказательства

Утверждение 23. Пусть $n \in \{1, 2, \dots\}$, $m \in \{0, 1, \dots\}$ и $\pi = \langle \text{String} \rangle(n, m)$. Тогда для любого $i \in \text{dom}(\pi)$
 $\langle \text{String} \rangle(n, \pi(i) - 1) \in \text{suffixes}(\text{Prefix}(i - 1, \pi)).$

Доказательство. Для любых $k \in \{1, \dots, m\}$ и $\alpha \in \{1, \dots, m\}$ * обозначим множество

$$\{i : i \in \text{dom}(\alpha) \& \alpha(i) = k\} \text{ через } \langle \text{Occurrences} \rangle^{Var}(k, \alpha),$$

множество $\{\alpha(i) : i \in \text{dom}(\alpha) \& k \leq \alpha(i)\}$ через $\langle \text{Vals} \rangle^{notLess}(k, \alpha)$,

а строку $\text{Prefix}(\min(\langle \text{Occurrences} \rangle^{Var}(k, \alpha)), \alpha)$ через $\langle \text{Prefix} \rangle^{first}(k, \alpha)$.

По определению /50/ $\langle \text{String} \rangle(n, m)$

$$\langle \text{Prefix} \rangle^{first}(k, \pi) = \langle \text{String} \rangle(n, k - 1) \wedge \{k\}.$$

Следовательно для доказательства утверждения достаточно доказать, что для любого $i \in \langle_{Occurrences}^{Var} \rangle(k, \pi)$ выполняется

$$\langle_{Prefix}^{first} \rangle(k, \pi) \in suffixes(Prefix(i, \pi)).$$

Докажем, что для любого $i \in \langle_{Occurrences}^{Var} \rangle(k, \pi)$

$$\langle_{Prefix}^{first} \rangle(k, \pi) \in suffixes(Prefix(i, \pi)).$$

Доказательство проведем индукцией мощности множества

$$\langle_{Vals}^{notLess} \rangle(k, Prefix(i, \pi)).$$

Базис. $|\langle_{Vals}^{notLess} \rangle(k, Prefix(i, \pi))| = 1$. В этом случае

$$\langle_{Vals}^{notLess} \rangle(k, Prefix(i, \pi)) = \{k\}. \text{ Следовательно, (по определению /50/} \\ \langle_{String}^{Iteration} \rangle(n, m)), \text{ для некоторого } n' \in \{1, \dots, n\}$$

$$Prefix(i, \pi) = \langle_{Prefix}^{first} \rangle(k, \pi)^{\wedge n'}. \text{ Таким образом,}$$

$$\langle_{Prefix}^{first} \rangle(k, \pi) \in suffixes(Prefix(i, \pi)).$$

Шаг индукции. Пусть $|\langle_{Vals}^{notLess} \rangle(k, Prefix(i, \pi))| > 1$,

$$k' = \max(\langle_{Vals}^{notLess} \rangle(k, Prefix(i, \pi))) \text{ и } \alpha = \langle_{Prefix}^{first} \rangle(k', \pi). \text{ Тогда}$$

$\langle_{Occurrences}^{Var} \rangle(k, \alpha) \neq \emptyset$ (по определению /50/ $\langle_{String}^{Iteration} \rangle(n, m)$) и для любого $i' \in \langle_{Occurrences}^{Var} \rangle(k, \alpha)$ $\langle_{Prefix}^{first} \rangle(k) \in suffixes(Prefix(i', \pi))$ (по индуктивному предположению).

Следовательно, для любых $n' \in \{1, 2, \dots\}$ и

$$i'' \in \langle_{Occurrences}^{Var} \rangle(k, \alpha^{\wedge n'}) \quad \langle_{Prefix}^{first} \rangle(k) \in suffixes(Prefix(i'', \alpha^n)).$$

В то же время (по определению /50/ $\langle_{String}^{Iteration} \rangle(n, m)$) существует $n'' \in \{1, \dots, n+1\}$ такое, что $Prefix(i, \pi) = Prefix(i, \alpha^{\wedge n''})$. Конец доказательства

Следствие 1. Пусть $A = (Q, F, \mu)$ – конечная комбинаторная алгебра (/45/), $N = (V, R, F, functor)$ – конечная сеть (/33/), \ll – линейный порядок на V , $\pi = \langle_{Plan}^{Key} \rangle_A(N, \ll)$, $i \in dom(\pi)$ и $\pi(i) = \{x\}$. Тогда для любого состояния $f \in States(A, N)$ все вершины множества $V \setminus \{y : x \ll y\}$ являются настроенными в состоянии $\langle_{Protocol}^{Computer} \rangle(f, \pi)(i)$.

Доказательство. Пусть $N' = N.isolate\{(y : x \ll y)\}$.

В силу утверждения 23 $Prefix(i-1, \pi) = \pi' \wedge \pi''$, где $\pi'' = \langle_{Plan}^{Key} \rangle_A(N', \ll)$, а $\pi' \in \langle_{Plans}^{Computer} \rangle(N)$.

Таким образом, для подстановок

$$\varphi = \langle_{Sub}^{Computer} \rangle_N(Prefix(i-1, \pi)) \in \langle_{Subs}^{NetWork} \rangle(N),$$

$$\varphi' \in \langle_{Sub}^{Computer} \rangle_N(\pi') \in \langle_{Subs}^{NetWork} \rangle(N) \text{ и}$$

$\varphi'' = \langle \text{Computer} \rangle_{Sub} \langle \text{NetWork} \rangle_{Subs} (N)$ выполняется соотношение (опр /38/ $\langle \text{Computer} \rangle_{Sub} \varphi = \varphi' \varphi''$).

В силу утверждения 22 $\varphi'' = \langle \text{Key} \rangle_{Subs} A(N', <<)$, а, следовательно (утв.21), $\varphi''|_V$ - параметрическое решение (/32/) $system(N')$ над A . В таком случае (утв. 13.следствие 1) $\langle \text{Value} \rangle_{Subs} A(f^{Terms}(F \cup Q, V) \varphi' \varphi'')$ - решение $system(N')$ над A .

Поскольку (Теорема 2.следствие 1) $\langle \text{Value} \rangle_{Subs} A(f^{Terms}(F \cup Q, V) \varphi' \varphi'') = \langle \text{Computer} \rangle_{Result} A, N(f, Prefix(i - 1, \pi)) = \pi(i)$, то (утв 16.следствие 1) состояние $\pi(i)$ сети N' настроено, а значит (опр. /4/) настроены все вершины из множества $N.isolate(\{y : x << y\})$. Конец доказательства

Определение 52. Пусть $A = (Q, F, \mu)$ - конечная алгебра (/25/), $N = (V, R, F, functor)$ - конечная сеть (/33/), $<<$ - линейный порядок на V и $\pi = \langle \text{Plan} \rangle_{Key} A(N, <<)$.

Определим отображения

$\langle \text{nonRedundant} \rangle_{Set} A, N, << : States(A, N) \rightarrow 2^{\{1, 2, \dots\}}$ и $\langle \text{nonRedundant} \rangle_{Plan} A, N, << : States(A, N) \rightarrow \{1, 2, \dots\}^*$ следующим образом:

Для любого $f \in States(A, N)$

- $\langle \text{nonRedundant} \rangle_{Set} A, N, << (f) \stackrel{def}{=} \{i : i \in dom(\pi) \& \langle \text{Computer} \rangle_{Protocol} (f, \pi)(i) \neq \langle \text{Computer} \rangle_{Protocol} (f, \pi)(i + 1)\}$
- $\langle \text{nonRedundant} \rangle_{Plan} A, N, << (f)$ это строка которая является монотонным и взаимно-однозначным отображением из множества $\{1, \dots, |\langle \text{nonRedundant} \rangle_{Set} A, N, << (f)|\}$ на $\langle \text{nonRedundant} \rangle_{Set} A, N, << (f)$.

Утверждение 24. Пусть $A = (Q, F, \mu)$ - конечная алгебра (/25/), $N = (V, R, F, functor)$ - конечная сеть (/33/), $<<$ - линейный порядок на V , $\tau = \langle \text{Plan} \rangle_{Key} A(N, <<)$, $f \in States(A, N)$,

$s = \langle \text{nonRedundant} \rangle_{Set} A, N, << (f)$ и $\pi = subseq(s, \tau)$

(π - подпоследовательность, протокола τ , порожденная множеством $s \subseteq \tau$). Тогда для любого $i \in \langle \text{nonRedundant} \rangle_{Set} A, N, << (f)$ выполняются следующие равенства:

- 1) $\pi(num_{s, \leq}(i)) = \tau(i)$,
- 2) $\langle \text{Computer} \rangle_{Protocol} (f, \pi)(num_{s, \leq}(i)) = \langle \text{Computer} \rangle_{Protocol} (f, \tau)(i)$

Доказательство.

1. Поскольку $num_{s, \leq} : s \rightarrow \{1, \dots, |s|\}$ взаимно-однозначное отображение (утв. 4), то равенство $\pi(num_{s, \leq}(i)) = \tau(i)$ для всех $i \in s$ равносильно равенству $\pi(j) = \tau(num_{s, \leq}^{-1}(j))$ для всех $j \in dom(\pi)$.

Равенство $\pi(j) = \tau(num_{s, \leq}^{-1}(j))$ следует из утверждения 5: $\pi = subseq(s, \tau) = /*_{утв. 5}*/ string(s \cap dom(\tau), \leq, \tau) = /*_s \subseteq dom(\tau)*/ = string(s, \leq, \tau) = /*оп./12*/ = num_{s, \leq}^{-1} \bullet \tau$, где \bullet - операция композиции отображений. Таким образом $\pi(j) = (num_{s, \leq}^{-1} \bullet \tau)(j) = \tau(num_{s, \leq}^{-1}(j))$. (В частности $\pi(1) = \tau(min(s))$).

2. Равенство

$\langle_{Protocol}^{Computer} \rangle(f, \pi)(num_{s, \leq}(i) + 1) = \langle_{Protocol}^{Computer} \rangle(f, \tau)(i + 1)$ является следствием (по определению /42/ $\langle_{Protocol}^{Computer} \rangle$) равенства $\pi(num_{s, \leq}(i)) = \tau(i)$ (это равенство доказано в пункте 1. утв 24). Действительно, пусть $g = \langle_{Protocol}^{Computer} \rangle(f, \pi)(num_{s, \leq}(i)) = \langle_{Protocol}^{Computer} \rangle(f, \tau)(i)$ и $m = \pi(num_{s, \leq}(i)) = \tau(i)$. Тогда (по определению /42/) $step_{A, N, m}(g) = \langle_{Protocol}^{Computer} \rangle(f, \pi)(num_{s, \leq}(i) + 1) = \langle_{Protocol}^{Computer} \rangle(f, \tau)(i + 1)$.

Вместе с тем (по определению /52/ $\langle_{Set}^{nonRedundant} \rangle$ и по определению /42/ $\langle_{Protocol}^{Computer} \rangle$) $\langle_{Protocol}^{Computer} \rangle(f, \pi)(num_{s, \leq}(1)) = /*оп/52*/ f = /*оп/42*/ \langle_{Protocol}^{Computer} \rangle(f, \tau)(1)$. Конеч доказательства

Утверждение 25. Пусть $A = (Q, F, \mu)$ – комбинаторная алгебра (/45/), $N = (V, R, F, functor)$ – конечная сеть (/33/), \ll – линейный порядок на V и $\pi = \langle_{Plan}^{Key} \rangle_A(N, \ll)$. Тогда для любого состояния $f \in States(A, N)$ и любого $i \in \langle_{Set}^{nonRedundant} \rangle_{A, N, \ll}(f)$ $\pi(i) = \{ \langle_{node}^{chief} \rangle_{N, A}(\ll, \langle_{Protocol}^{Computer} \rangle(f, \pi)(i)) \}$.

Доказательство. Пусть $\pi(i) = \{x\}$. В соответствии с утверждением 23.следствие 1 все вершины множества $V \setminus \{y : x \ll y\}$ являются настроенными в состоянии $\langle_{Protocol}^{Computer} \rangle(f, \pi)(i)$. Поскольку $\langle_{Protocol}^{Computer} \rangle(f, \pi)(i) \neq \langle_{Protocol}^{Computer} \rangle(f, \pi)(i + 1)$ (по определению /52/ $\langle_{Set}^{nonRedundant} \rangle_{A, N, \ll}(f)$), то вершина x не настроена в состоянии $\langle_{Protocol}^{Computer} \rangle(f, \pi)(i)$. Таким образом, x – минимальная по порядку \ll вершина не настроенная в состоянии $\langle_{Protocol}^{Computer} \rangle(f, \pi)(i)$. В таком случае (определение /46/) $x = \langle_{node}^{chief} \rangle_{N, A}(\ll, \langle_{Protocol}^{Computer} \rangle(f, \pi)(i))$.

Конец
доказательства

Следствие 1. Пусть $A = (Q, F, \mu)$ – конечная комбинаторная алгебра (/45/), $N = (V, R, F, functor)$ - конечная сеть и \ll - линейный порядок на V . Тогда для любого состояния $f \in States(A, N)$ подпоследовательность $subseq(\langle \langle^{nonRedundant}_{Set} \rangle_{A, N, \ll}(f), \langle^{Key}_{Plan} \rangle_{A(N, \ll)} \rangle) \in \langle^{Computer}_{Plans} \rangle(N)$ последовательности $\langle^{Key}_{Plan} \rangle_{A(N, \ll)} \in \langle^{Computer}_{Plans} \rangle(N)$ является \ll, f -планом Киссинджера (/46/).

Доказательство. Пусть $s = \langle^{nonRedundant}_{Set} \rangle_{A, N, \ll}(f)$,
 $\tau = \langle^{Key}_{Plan} \rangle_{A(N, \ll)}$ и $\pi = subseq(s, \tau)$.

Возможны следующие варианты $s = \emptyset$ и $s \neq \emptyset$.

Если $s = \emptyset$, то (опр /13/ $subseq(\emptyset, \tau)$) $\pi = \varepsilon$ и (опр /52/ $\langle^{nonRedundant}_{Set} \rangle$) $f = \langle^{Computer}_{Result} \rangle_{A, N}(f, \pi)$ следовательно (утв 22.следствие 1) f настроенное состояние. Следовательно (в соответствии с определением /46/) $\pi = \varepsilon$ является \ll, f -планом Киссинджера (/46/).

Пусть $s \neq \emptyset$. В силу определения /46/ достаточно убедиться в том, что для любого $i \in dom(\pi)$

$$\pi(i) = \{ \langle^{chief}_{node} \rangle_{N, A}(\ll, \langle^{Computer}_{Protocol} \rangle(f, \pi)(i)) \}.$$

По определению /13/ $subseq(s, \tau)$ последнее равносильно, тому, что для любого $j \in s$ $\pi(num_{s, \leq}(j)) = \{ \langle^{chief}_{node} \rangle_{N, A}(\ll, \langle^{Computer}_{Protocol} \rangle(f, \pi)(num_{s, \leq}(j))) \}.$

Согласно утв. 24 $\langle^{Computer}_{Protocol} \rangle(f, \pi)(num_{s, \leq}(j)) = \langle^{Computer}_{Protocol} \rangle(f, \tau)(j)$ и $\pi(num_{s, \leq}(j)) = \tau(j)$, а согласно утв.25

$$\tau(i) = \{ \langle^{chief}_{node} \rangle_{N, A}(\ll, \langle^{Computer}_{Protocol} \rangle(f, \tau)(i)) \}. \quad \text{Конец доказательства}$$

Следствие 2. (Достаточность условия комбинаторности в теореме 1) Любая конечная комбинаторная алгебра (/45/) является регулярно устойчивой (/7/). Без доказательства

3.2. УСТОЙЧИВЫЕ И РЕГУЛЯРНО УСТОЙЧИВЫЕ АЛГЕБРЫ

Следующий пример демонстрирует тот факт, что существуют конечные устойчивые алгебры (/7/), которые не являются комбинаторными (/45/).

Пример 10. Пусть $F = \{\emptyset, v, a, f\}$ – множество функциональных символов (15/), для которого $arn(\emptyset) = arn(v) = 0$, $arn(a) = 1$ и $arn(f) = 2$. Пусть $\tau = \{x \rightarrow v\}((f(x, a(x)))^{x,2}) = f(f(v, a(v)), a(f(v, a(v)))) \in Terms(F, \emptyset)$ (18,44,16/).

Рассмотрим множество $Q = (\langle_{terms}^{sub}(\tau) \setminus \{\tau\} \cup \{\emptyset\}$ (17/: $Q = \{v, a(v), f(v, a(v)), a(f(v, a(v))), \emptyset\}$) и алгебру $A_\tau = (Q, F, \mu)$ (25/), в которой μ определено следующим образом:

- если $c \in \{\emptyset, v\}$, то $\mu(c)(\varepsilon) = c$,
- для любого $t \in Q$ $\mu(a)(t) = \begin{cases} a(t), & \text{если } a(t) \in Q \\ \emptyset, & \text{если } a(t) \notin Q \end{cases}$;
- для любых $t, t' \in Q$

$$\mu(f)(t, t') = \begin{cases} f(t, t'), & \text{если } f(t, t') \in Q \\ v, & \text{если } f(t, t') = \tau \\ \emptyset, & \text{если } f(t, t') \notin Q \cup \{\tau\} \end{cases}$$

На рис 7 представлено строение этой алгебры в виде таблицы операций (7a) и в виде функциональной сети (7б) (вершина соответствующая элементу алгебры \emptyset на рис. 7б не изображена).

Эта алгебра не является комбинаторной (45/), поскольку для терма $t = f(x, a(x)) \in Terms(F, \{x\})$ (16/) выполняется равенство $value_{A_\tau}(\{x \rightarrow v\}t^{x,2}) = v$ (27, 18,44/) и в то же время $value_{A_\tau}(\{x \rightarrow v\}t) = f(v, a(v)) \neq v$ (27, 18/).

Вместе с тем, для любой функциональной сети (33/) $G = (W, R, \{\emptyset, v, a, f\}, functor)$, для любого состояния $g \in States(A_\tau, G)$ (40/) и любого линейного порядка $\ll \in \langle_{Orders}^{Line}(W)$, такого, что из $functor(x) = f$ и $functor(y) = a$ следует $x \ll y$ (g, \ll)- настройка (46/) является полной (41/).

Без доказательства •

Оставшаяся часть раздела посвящена доказательству того, что любая некомбинаторная алгебра (45/) не является регулярно устойчивой (7/).

Определение 53. *Определим отображение*

$EqNet : equations(F, \{x\}) \rightarrow netwares(Terms(F, \emptyset), F)$ (20,33/) *следующим образом. Для любого уравнения $x=t \in equations(F, \{x\})$*

а) $\mu(f)$:

$x_1 \backslash x_2$	v	$a(v)$	$f(v, a(v))$	$a(f(v, a(v)))$	\emptyset
v	\emptyset	$f(v, a(v))$	\emptyset	\emptyset	\emptyset
$a(v)$	\emptyset	\emptyset	\emptyset	\emptyset	\emptyset
$f(v, a(v))$	\emptyset	\emptyset	\emptyset	v	\emptyset
$a(f(v, a(v)))$	\emptyset	\emptyset	\emptyset	\emptyset	\emptyset
\emptyset	\emptyset	\emptyset	\emptyset	\emptyset	\emptyset

$\mu(a)$:

x	$\mu(a)(x)$
v	$a(v)$
$a(v)$	\emptyset
$f(v, a(v))$	$a(f(v, a(v)))$
$a(f(v, a(v)))$	\emptyset
\emptyset	\emptyset

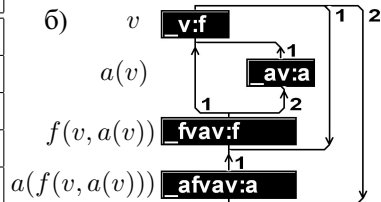


Рис. 7. Устойчивая некомбинаторная алгебра.

(/20/) $EqNet(x=t) = (\langle \substack{sub \\ terms} \rangle(t) \setminus \{x\}, R, F, topFunctor)$ (/17/) такая сеть (/33/), что $(\tau, i, \tau') \in R$ тогда и только тогда, когда выполняются одно из следующих условий:

- либо $Sons(\tau)(i) = x$ и $\tau' = t$ (/17/);
- либо $Sons(\tau)(i) = \tau'$ (/17/).

Утверждение 26. (Необходимость условия комбинаторности в теореме 1.) Пусть $A = (Q, F, \mu)$ – конечная некомбинаторная алгебра (/45/). Тогда существует такое уравнение $u \in equations(F, \{x\})$ (/20/), что для некоторого линейного порядка \ll (/1.3.1/) на множестве вершин схемы $EqNet(u)$ (/53/), существует такое состояние $f \in States(A, EqNet(u))$ (/40, 53/), что (f, \ll) -настройка (/46/) схемы $EqNet(u)$ не является полной (/41/).

Доказательство. Поскольку A – некомбинаторная алгебра (/45/), существуют терм $t \in Terms(F, \{x\})$ (/16/), $q \in Q$ и целое

$k > 1$ такие, что $value_A(\{x \rightarrow q\}t^{x,k})=q$ и $value_A(\{x \rightarrow q\}t) \neq q$ (/27, 18, 44/).

Рассмотрим схему $N = EqNet(x = t)$ (/53/) и состояние $f \in States(A, N)$ (/40/) такое, что для любой вершины $\tau \in \langle_{terms}^{sub}(t) \setminus \{t\}$ (/17/ в соответствии с определением /53/ вершины $EqNet$ являются поддермами t) $f(\tau) = value_A(\{x \rightarrow q\}(\tau))$ (/27,18/). Это состояние не согласовано (/4/), и единственной не настроенной вершиной (/4/) является вершина x (поскольку $value_A(\{x \rightarrow q\}(t)) \neq q$ (/27, 18/)).

Рассмотрим такой линейный порядок (/1.3.2/) \ll на множестве $\langle_{terms}^{sub}(t) \setminus \{t\}$ (/17/) ($\ll \in \langle_{Orders}^{Line}(\langle_{terms}^{sub}(t) \setminus \{t\})$), в котором вершина x является наибольшей (/1.3.2/). Тогда (f, \ll) -настройка схемы N (/46/) – бесконечная последовательность (не является полной (/41/)).

Конец
доказательства

ABOUT THE TUNING OF THE INTERACTING OBJECTS BEHAVIOR

Andrey Babichev, Institute of Control Sciences of RAS, Moscow, Cand.Sc. (*babichev@ipu.ru*).

Abstract: We investigate the problem of interacting subjects' coordination and consider a procedure, which iteratively tunes the state of one subject to reconcile it with states of its neighbors. Every step results in coordination of one subject, but other subjects may become uncoordinated even if they were tuned before. We derive the conditions when this process stabilizes. We also discuss subjects' characteristics which guarantee tunability of any network of such subjects, and construct a corresponding tuning algorithm.

Keywords: functional flow block diagram, system of equations solution, universal algebra, iterative method..

Статья представлена к публикации членом редакционной коллегии Б. Т. Поляком

*Поступила в редакцию 22.02.2013.
Опубликована 30.11.2013.*

УДК 519.85+519.863
ББК 22.19

ЧИСЛЕННЫЕ ЭКСПЕРИМЕНТЫ С ВАРИАНТАМИ АЛГОРИТМОВ ВНУТРЕННИХ ТОЧЕК НА НЕЛИНЕЙНЫХ ЗАДАЧАХ ПОТОКОРАСПРЕДЕЛЕНИЯ

Зоркальцев В. И.¹, Медвежонков Д. С.²
(ФГБУН Институт систем энергетики
им. Л.А. Мелентьева СО РАН, Иркутск)

Представлены результаты сравнительных экспериментальных исследований вариантов алгоритмов внутренних точек на нелинейных моделях потокораспределения. В экспериментах выявлено преимущество линейных весовых коэффициентов, деленных на множители Лагранжа, перед квадратичными. Установлено, что при использовании двойственных алгоритмов требуемая точность решения достигается быстрее, чем при использовании прямых алгоритмов внутренних точек.

Ключевые слова: прямые и двойственные алгоритмы внутренних точек, весовые коэффициенты, нелинейные задачи потокораспределения.

1. Введение

В работах [7, 13, 14] рассматривается класс задач выпуклого программирования с линейными ограничениями и сепарабельной целевой функцией, обладающих свойством симметричной двойственности. Этим термином названа ситуация, когда двойственная к двойственной задаче оптимизации совпадает с

¹ Валерий Иванович Зоркальцев, доктор технических наук, профессор, заведующий лабораторией (zork@isem.sei.irk.ru).

² Дмитрий Сергеевич Медвежонков, младший научный сотрудник (dmitry@isem.sei.irk.ru).

исходной. Наличие симметричной двойственности даёт ряд преимуществ, в том числе возможность поиска решения исходной оптимизации на основе алгоритмов поиска оптимального решения двойственной задачи.

Основная цель данной статьи – изложить результаты сравнительных экспериментальных исследований вариантов алгоритмов внутренних точек для решения задач нелинейной оптимизации, обладающих свойством симметричной двойственности. Свойство симметричной двойственности позволяет использовать не только так называемые прямые алгоритмы внутренних точек (осуществляющих монотонное улучшение решения исходной задачи оптимизации), но и двойственные алгоритмы внутренних точек (осуществляющие монотонное улучшение решения двойственной задачи оптимизации).

В данной статье рассматривается класс алгоритмов внутренних точек, пионерами в разработке которых в конце 60-х и в 70-х годах прошлого века были отечественные ученые С.М. Анциз, И.И. Дикин, Ю.Г. Евтушенко, В.Г. Жадан, В.И. Зоркальцев [1, 4, 5, 6, 9]. В дальнейшем с 80-х годов прошлого века эти алгоритмы привлекли повышенное внимание во всём мире. Частными случаями рассматриваемых в этой статье алгоритмов являются широко известные алгоритмы под названием *affine scaling method* и *dual affine scaling method*.

К настоящему времени можно считать общепринятым, что в результате проведенных теоретических и многочисленных экспериментальных исследований установлена высокая вычислительная эффективность рассматриваемых алгоритмов внутренних точек. Например, они, как правило, позволяют быстрее получать решения задач линейного программирования разной размерности, чем хорошо отработанные в результате многолетней вычислительной практики алгоритмы симплекс-метода.

В силу конструктивных особенностей обсуждаемых алгоритмов (обуславливающих их вычислительную эффективность) для их теоретического обоснования нельзя воспользоваться стандартной техникой, изложенной, например, Зангвиллом [8]. К настоящему времени полученные результаты теоретического

обоснования алгоритмов внутренних точек рассматриваемого здесь типа относятся в основном к их использованию для решения задач линейного и отчасти квадратичного программирования. Хотя уже с 70-х годов прошлого века эти алгоритмы успешно применяются при реализации многих моделей энергетики, относящихся к другим классам задач математического программирования. В этой связи особый интерес представляют экспериментальные исследования алгоритмов. Как показывает практика предыдущих экспериментальных исследований, их результаты бывают полезны для выработки рекомендаций по использованию вариантов алгоритмов в конкретных типах задач и для последующего теоретического обоснования алгоритмов.

Здесь будет рассматриваться две альтернативы в выборе вариантов алгоритмов внутренних точек. Одна из них – правила задания так называемых «весовых коэффициентов», с помощью которых во вспомогательной задаче определения направления улучшения решения учитываются ограничения-неравенства исходной задачи оптимизации. В [10–12] на основе аксиоматического подхода вводятся и обсуждаются различные возможные правила задания этих коэффициентов, приводящих к алгоритмам, которые обладают разными свойствами. Наиболее известными являются так называемые квадратичные весовые коэффициенты, введенные в первых работах по данному типу алгоритмов внутренних точек И.И. Дикиным [4]. Результаты развития такого алгоритма и практические приложения отражены в монографии [5]. В дальнейшем в работах зарубежных авторов такой алгоритм внутренних точек получил название «affine scaling method».

У данного алгоритма есть одна нехорошая в вычислительном отношении особенность, проявляющаяся уже при решении задачи линейного программирования. При оптимизации в области допустимых решений шаг, с которым осуществляется улучшение решения по выбранному направлению, неограниченно возрастает. Это означает, что на финальных итерациях неизбежные малые погрешности в решении вспомогательной задачи выбора направления корректировки решения сильно проявляют-

ся в вычислительном процессе, в том числе через выход из области допустимых решений.

В целях борьбы с этим негативным явлением было предложено вместо квадратичных весовых коэффициентов на финальных итерациях использовать линейные весовые коэффициенты, деленные на оценки предыдущих итераций множителей Лагранжа соответствующих ограничений-неравенств [11]. Последующие экспериментальные исследования [2] показали, что таким вариантом алгоритмов внутренних точек вполне можно пользоваться не только на финальных, но и на всех итерациях. Теоретические исследования [12] показали, что эти алгоритмы обладают, так же как и алгоритмы с квадратичными весовыми коэффициентами, линейной скоростью сходимости, асимптотически не зависящей от исходных данных задачи, и могут обладать сверхлинейной скоростью сходимости. Практический интерес и интерес для дальнейших теоретических исследований представляет выяснение вопроса на основе экспериментальных расчетов: как соотносятся по времени счета эти два варианта алгоритмов на задачах с нелинейной целевой функцией?

Вторая альтернатива – выбор между «прямыми» и «двойственными» алгоритмами внутренних точек. Прямые алгоритмы осуществляют монотонное улучшение исходной задачи оптимизации в области решений, удовлетворяющих в строгой форме её ограничениям-неравенствам. На этапе ввода в область допустимых решений на каждой итерации происходит сокращение всех компонент вектора абсолютных значений невязок ограничений-равенств. На втором этапе осуществляется процесс оптимизации (монотонного улучшения значения целевой функции) в области допустимых решений исходной задачи.

При этом на каждой итерации при решении вспомогательной задачи выбора направления корректировки решения вырабатывается также приближение к решению двойственной задачи. Как было подмечено еще в 70-х годах И.И. Дикиным, приближения к решению двойственной задачи хотя и не монотонно в каком либо смысле, но значительно лучше (быстрее) сходятся к оптимальному решению двойственной задачи, чем

сходятся монотонно улучшаемые решения исходной задачи к её оптимальному решению. В дальнейшем этот факт получил теоретическое обоснование [10] на задачах ЛП.

Двойственный алгоритм внутренних точек осуществляет монотонное улучшение решения двойственной задачи. При этом на каждой итерации вырабатывается и приближение к решению исходной задачи. В этом случае, по аналогии с предыдущим, следует ожидать более быструю сходимость приближений исходной задачи к её оптимальному решению, чем сходимость приближений двойственной задачи к её оптимальному решению. Важно также отметить, что для рассматриваемых здесь задач легко получить стартовую точку двойственной задачи, удовлетворяющую всем её ограничениям-неравенствам в строгой форме. Оба эти момента позволяют надеяться, что двойственные алгоритмы могут быть более эффективными для определения с заданной точностью оптимального решения исходной задачи оптимизации.

Заметим, что у всех четырех рассматриваемых здесь алгоритмов объем вычислений на каждой итерации (при решении одной и той же задачи) одинаков. Это позволяет использовать число итераций для достижения решения с одной и той же заданной точностью в качестве сравнительного показателя времени счета.

Конкретным объектом экспериментальных исследований данной статьи будут нелинейные модели потокораспределения [3, 7, 13–16]. Этим термином объединяются нелинейные (с затратами на перевозки нелинейно зависящими от объемов перевозок) транспортные задачи, а также электрические цепи и гидравлические цепи (моделирующие транспортировку по трубопроводам воды, нефти или нефтепродуктов).

2. Взаимно-двойственные задачи оптимизации

Заданы матрица A размера $m \times n$ и множество номеров $J = \{1, \dots, n\}$. Множества L, H являются некоторыми подмноже-

ствами J . Также заданы: вектор $b \in R^m$, вектор $c \in R^n$, величины $\underline{x}_j, j \in L$, и $\bar{x}_j, j \in H$, при этом $\underline{x}_j < \bar{x}_j$ для $j \in L \cap H$.

Обозначим Z множество функций из R в R , которые равны нулю в нуле и имеют непрерывные возрастающие производные. Причем производные этих функций равны нулю в нуле и изменяются от $-\infty$ до $+\infty$.

Заметим, что любой функции из Z можно поставить в соответствие сопряженную функцию из Z , производная которой является обратной функцией к производной исходной функции. Такой переход к новой функции равносителен применению к исходной функции преобразования Лежандра–Фенхеля, известного из выпуклого анализа.

Пусть задан набор функций $F_j(x_j), j \in J$, принадлежащих множеству Z . Введем функции от векторов R^n : $F(x) = \sum_{j \in J} F_j(x_j)$.

Рассмотрим задачу выпуклой оптимизации с выпуклой сепарабельной целевой функцией и линейными ограничениями с вектором переменных $x \in R^n$:

- (1) $F(x) + c^T x \rightarrow \min$
- (2) $Ax = b$,
- (3) $\underline{x}_j \leq x_j, j \in L$,
- (4) $x_j \leq \bar{x}_j, j \in H$.

Назовем (1)–(4) исходной задачей оптимизации. Расширенным решением исходной задачи назовем набор, состоящий из вектора исходных переменных x , вектора $u \in R^m$ множителей Лагранжа ограничений (2), векторов $l \in R^n$ и $h \in R^n$, содержащих множители Лагранжа ограничений (3) и (4), причем $l_j = 0, j \in J \setminus L$ и $h_j = 0, j \in J \setminus H$, а также вектора $y \in R^n$ с компонентами $y_j = f_j(x_j)$, где f_j – производная $F_j, j \in J$.

Обозначим Φ_j функцию из Z , полученную в результате преобразования Лежандра–Фенхеля из функции $F_j \in Z, j \in J$. Введем функцию от вектора $y \in R^n$: $\Phi(y) = \sum_{j \in J} \Phi_j(y_j)$.

В [7] установлено, что двойственную задачу оптимизации к задаче (1)–(4) можно записать в виде

$$(5) \quad \Phi(y) - b^T u - \sum_{j \in L} \underline{x}_j l_j + \sum_{j \in H} \bar{x}_j h_j \rightarrow \min$$

$$(6) \quad y + c - l + h = A^T u,$$

$$(7) \quad l_j \geq 0, \quad j \in L; \quad h_j \geq 0, \quad j \in H,$$

$$(8) \quad l_j = 0, \quad j \in J, \quad j \notin L; \quad h_j = 0, \quad j \in J, \quad j \notin H.$$

Здесь переменными являются векторы $y \in R^n$, $u \in R^m$ и компоненты векторов $l \in R^n$ и $h \in R^n$ соответственно при $j \in L$ и $j \in H$.

Расширенным решением двойственной задачи назовем набор, состоящий из векторов её переменных y , u , l , h , а также вектора $x \in R^n$ множителей Лагранжа ограничений (6).

Рассмотрим систему уравнений и неравенств, включающую условия (2)–(4), (8) и дополнительные ограничения:

$$(9) \quad y_j = f_j(x_j), \quad j \in J,$$

$$(10) \quad y + c - l + h = A^T u,$$

$$(11) \quad l_j = (f_j(x_j) + c_j - [A^T u]_j)_+, \quad j \in L,$$

$$(12) \quad h_j = ([A^T u]_j - f_j(\bar{x}_j) - c_j)_+, \quad j \in H.$$

Здесь f_j – производная функции F_j , $j \in J$, а $(\alpha)_+ = \max\{0, \alpha\}$.

В [7] обосновывается справедливость утверждений следующей теоремы, которая говорит о возможности восстановить решение исходной задачи по решению двойственной. Это дает право пользоваться двойственными алгоритмами внутренних точек для решения задач потокораспределения.

Теорема. Если ограничения исходной задачи (1)–(4) совместны, то решение этой задачи существует и единственно, решение двойственной задачи (5)–(8) существует и единственно по вектору y , решение системы уравнений и неравенств (2)–(4), (8), (9)–(12) существует, единственно по векторам x , y и совпадает с расширенными решениями исходной и двойственной задач. В противном случае решения исходной и двойственной задач не

существуют, система уравнений и неравенств (2)–(4), (8), (9)–(12) несовместна.

Замечание. Ограничения (11), (12) заменяют известные в оптимизации условия дополняющей нежесткости. Эти ограничения лучше в вычислительном отношении, чем билинейные условия дополняющей нежесткости.

3. Прямые алгоритмы внутренних точек

Итерационный процесс прямых алгоритмов внутренних точек решения исходной задачи (1)–(4), заключается в последовательном построении нового приближения:

$$(13) \quad x^{k+1} = x^k + \lambda_k s^k,$$

где s^k – направление корректировки текущего приближения, λ_k – величина шага вдоль этого направления. Процесс начинается из точки x^0 , которая удовлетворяет ограничениям-неравенствам (3)–(4) в строгой форме. Считаем, что матрица A имеет полный ранг, целевая функция в (1) дважды дифференцируема. Обозначим $f'_j(x_j)$ – вторую производную функции $F_j(x_j)$, $j \in J$.

В этом алгоритме выделяются два этапа вычислений. Сначала осуществляется ввод в область допустимых решений, в процессе которого уменьшаются невязки ограничений-равенств (2). На втором этапе осуществляется оптимизация в области допустимых решений.

Алгоритм итеративно будет повторять перечисленный в пунктах 1–6 набор действий.

Пункт 1. Вычисление вектора невязки ограничений (2) для текущего k -го приближения $x^k \in R^n$:

$$(14) \quad r^k = b - Ax^k.$$

Вычислим величину

$$\pi_A^k = \max \{ |r_i^k| : i \in I \}.$$

Зададим $\delta > 0$ – параметр алгоритма, характеризующий допустимую невязку ограничений-равенств (2). Если выполняется неравенство

$$(15) \pi_A^k < \delta,$$

то x^k принадлежит с требуемой точностью допустимой относительно ограничений-равенств (2) области. В этом случае алгоритм переходит на этап оптимизации в области допустимых решений. Если условие (15) не выполняется, то алгоритм остается на этапе ввода в область допустимых решений.

Пункт 2. Решение вспомогательной задачи поиска направления корректировки s^k текущего приближения x^k . В зависимости от этапа, на котором находится алгоритм, эта задача записывается по-разному. Возможны два случая.

1. Если алгоритм находится на этапе ввода в допустимую область, то решаем вспомогательную задачу относительно s :

$$(16) \frac{1}{2} \sum_{j \in J} \frac{(s_j)^2}{d_j^k} \rightarrow \min,$$

$$(17) As = r^k.$$

Величины $d_j^k, j \in J$ – положительные весовые коэффициенты, изменяющиеся по итерациям. Используется два варианта их определения.

1) Квадратичные весовые коэффициенты:

$$(18) d_j^k = (x_j^k - \underline{x}_j)^2, \text{ если } j \in L \text{ и } j \notin H,$$

$$(19) d_j^k = (\bar{x}_j - x_j^k)^2, \text{ если } j \in H \text{ и } j \notin L,$$

$$(20) d_j^k = (\min(x_j^k - \underline{x}_j, \bar{x}_j - x_j^k))^2, \text{ если } j \in L \text{ и } j \in H.$$

2) Линейные весовые коэффициенты, деленные на приближения к множителям Лагранжа l_j^{k-1}, h_j^{k-1} ограничений (3), (4):

$$(21) d_j^k = (x_j^k - \underline{x}_j) / \max(\delta_2, l_j^{k-1}), \text{ если } j \in L \text{ и } j \notin H,$$

$$(22) d_j^k = (\bar{x}_j - x_j^k) / \max(\delta_2, h_j^{k-1}), \text{ если } j \in H \text{ и } j \notin L,$$

$$(23) d_j^k = \frac{\min(x_j^k - \underline{x}_j, \bar{x}_j - x_j^k)}{\max(\delta_2, l_j^{k-1}, h_j^{k-1})}, \text{ если } j \in L \text{ и } j \in H.$$

Здесь δ_2 – параметр, представленный некоторым малым числом и позволяющий избежать деления на ноль; параметры l_j^{k-1}, h_j^{k-1} – это двойственные оценки, вычисляемые на предыду-

щей итерации алгоритма (если алгоритм на первой итерации, то положим $l_j^0 = 0, h_j^0 = 0$).

При $j \notin L \cup H$ значение коэффициентов d_j^k приравняется числу $\omega^k = \max\{\omega_1, \omega_2, \omega_3\}$, где $\omega_1 = \max\{x_j^k - \underline{x}_j : j \in L \setminus H\}$, $\omega_2 = \max\{\bar{x}_j - x_j^k : j \in H \setminus L\}$, $\omega_3 = \max\{\min\{x_j^k - \underline{x}_j, \bar{x}_j - x_j^k\} : j \in L \cap H\}$.

Отметим, что на этом этапе согласно (13), (14), (17) выполняется соотношение

$$(24) \quad r^{k+1} = (1 - \lambda_k)r^k,$$

поэтому на каждой итерации происходит сокращение абсолютных значений всех компонент вектора невязок балансовых ограничений r^k , если λ_k ограничено сверху единицей.

Найдем решение задачи (16), (17), используя правило множителей Лагранжа. Выразим компоненты s из условий оптимальности этой задачи:

$$(25) \quad s_j = d_j^k [A^T u]_j, \quad j \in J,$$

здесь $u \in R^m$ – вектор множителей Лагранжа ограничений (17).

Обозначим D^k – диагональную матрицу с элементами d_j^k на главной диагонали. Подставим полученные в (25) выражения для s в систему уравнений (17). Получаем систему линейных уравнений относительно вектора u :

$$(26) \quad AD^k A^T u = r^k.$$

Матрица $AD^k A^T$ – симметричная, положительно определенная. Решив систему (26), получим вектор u^k . Затем вычислим компоненты вектора s^k , используя правило (25).

II. Если алгоритм находится на этапе оптимизации в области допустимых решений, решаем вспомогательную задачу поиска направления корректировки текущего приближения в виде

$$(27) \quad \sum_{j \in J} (f_j(x_j^k) + c_j) s_j + \frac{1}{2} \sum_{j \in J} f_j'(x_j^k) (s_j)^2 + \frac{1}{2} \sum_{j \in J} \frac{(s_j)^2}{d_j^k} \rightarrow \min,$$

$$(28) \quad As = 0.$$

Здесь величины $d_j^k, j \in J$ вычисляются так же, как и на этапе ввода в область допустимых решений.

Найдем решение задачи (27), (28), используя правило множителей Лагранжа. Выразим компоненты вектора s :

$$(29) \quad s_j = \frac{(A^T u)_j - f_j(x_j^k) - c_j}{f_j'(x_j^k) + (d_j^k)^{-1}}, \quad j \in J,$$

Используем обозначения: G^k – диагональная матрица с n элементами вида $(f_j'(x_j^k) + (d_j^k)^{-1})^{-1}$ на главной диагонали, $f(x^k)$ – вектор с элементами $f_j(x_j^k)$, $j \in J$. Подставим полученные в (29) выражения для компонент вектора s в систему уравнений (28). Получим систему линейных уравнений относительно вектора u :

$$(30) \quad AG^k(A^T u - f(x^k) - c) = 0.$$

Решив систему (30), получим вектор u^k . Затем вычислим s^k , используя (29).

Пункт 3. Проверка выполнения критерия останова. Проверим для текущего приближения к решению задачи (1)–(4) выполнение условий (2)–(4), (8), (9)–(12). Рассчитаем величины l_j^k , h_j^k по формулам (11), (12). Далее рассчитаем компоненты вектора y^k по формулам:

$$(31) \quad y_j^k = [A^T u^k]_j - c_j, \quad j \in J \setminus (L \cup H),$$

$$(32) \quad y_j^k = [A^T u^k]_j - c_j + l_j^k - h_j^k, \quad j \in L \cup H.$$

Рассчитаем величину

$$(33) \quad \pi_Y^k = \max \{ |y_j^k - f_j(x_j^k)| : j \in J \}.$$

Если справедливо условие $\max \{ \pi_Y^k, \pi_A^k \} < \varepsilon$, где ε – параметр допустимой погрешности решения, то найдено решение задачи (1)–(4) с требуемой точностью. В этом случае завершаем работу алгоритма, иначе переходим к следующему пункту.

Пункт 4. Определение шага корректировки решения. Вычислим шаг до ближайшей границы допустимой области, задаваемой ограничениями-неравенствами, по формуле:

$$(34) \quad \lambda_{LH} = \min(\lambda_L, \lambda_H),$$

$$\text{где } \lambda_L = \min \{ (x_j - x_j^k)(s_j^k)^{-1} : j \in L, s_j < 0 \},$$

$$\lambda_H = \min \{ (\bar{x}_j - x_j^k)(s_j^k)^{-1} : j \in H, s_j > 0 \}.$$

Выбор итогового шага λ_k корректировки решения зависит от этапа, на котором находится алгоритм.

1) Если алгоритм находится на этапе ввода в область допустимых решений, то итоговый шаг λ_k вычислим по правилу:

$$\lambda_k = \min(\gamma\lambda_{LH}, 1), \text{ где } 0 < \gamma < 1 \text{ (например, } \gamma = 0,7).$$

Здесь γ – параметр метода, который используется, чтобы после корректировки приближение к решению оставалось внутри допустимой по ограничениям-неравенствами области.

2) Если алгоритм находится на этапе оптимизации в области допустимых решений, то сначала найдем величину λ_p , решив задачу одномерной минимизации целевой функции (1) из точки x^k по направлению s^k :

$$(35) \lambda_p = \arg \min_{\alpha \in R} \{F(x^k + \alpha s^k) + c^T(x^k + \alpha s^k) : 0 < \alpha \leq \lambda_{LH}\}.$$

А затем вычислим итоговый шаг λ_k корректировки решения:

$$(36) \lambda_k = \min(\gamma\lambda_{LH}, \lambda_p).$$

Пункт 5. *Вычисление следующего приближения.* Осуществим итеративный переход, используя найденные s^k и λ_k :

$$x^{k+1} = x^k + \lambda_k s^k.$$

4. Двойственные алгоритмы внутренних точек

Двойственный алгоритм решает двойственную задачу (5)–(8). Он отличается от прямого алгоритма тем, что для него несложно априори сформировать допустимое по ограничениям-равенствам (6) решение, для которого все ограничения-неравенства (7) выполняются в строгой форме. Поэтому одним из преимуществ рассматриваемых двойственных алгоритмов внутренних точек является то, что в них сразу со стартовой точки начинается процесс оптимизации в области допустимых решений.

Для реализации алгоритма необходимо, чтобы целевая функция в (5) была дважды дифференцируемой. Обозначим $\varphi_j(y_j)$ – вторую производную функции $\Phi_j(y_j)$, $j \in J$.

Зададим начальное приближение искоемых векторов $y^0 \in R^n$, $u^0 \in R^m$, $l^0 \in R^n$ и $h^0 \in R^n$ так, чтобы векторы начального приближения удовлетворяли равенствам (6), (8) и неравенствам (7) в строгой форме. Таким образом, алгоритм будет работать на одном этапе – этапе оптимизации в области допустимых решений.

Пункт 1. Поиск направления корректировки текущего приближения. Найдем векторы Δy^k , Δu^k , Δl^k , Δh^k , являющиеся решением задачи

$$(37) \quad \sum_{j \in J} \varphi_j(y_j^k) \Delta y_j + \frac{1}{2} \sum_{j \in J} \varphi_j'(y_j^k) (\Delta y_j)^2 - \sum_{i \in I} b_i \Delta u_i - \sum_{j \in L} \underline{x}_j \Delta l_j + \\ + \sum_{j \in H} \bar{x}_j \Delta h_j + \frac{1}{2} \sum_{j \in L} \frac{(\Delta l_j)^2}{q_j^k} + \frac{1}{2} \sum_{j \in H} \frac{(\Delta h_j)^2}{p_j^k} \rightarrow \min,$$

$$(38) \quad \Delta y + c - A^T \Delta u - \Delta l + \Delta h = 0,$$

$$(39) \quad \Delta l_j = 0, j \in J \setminus L, \quad \Delta h_j = 0, j \in J \setminus H.$$

Для вычисления знаменателей q_j^k и p_j^k функций штрафа в целевой функции (37) использовались два способа.

1) Квадратичные весовые коэффициенты:

$$(40) \quad q_j^k = (l_j^k)^2, j \in L; \quad p_j^k = (h_j^k)^2, j \in H.$$

2) Линейные весовые коэффициенты, деленные на множители Лагранжа:

$$(41) \quad q_j^k = l_j^k / \xi_j^{k-1}, j \in L; \quad p_j^k = h_j^k / \psi_j^{k-1}, j \in H.$$

Здесь величины ξ_j^{k-1} , ψ_j^{k-1} являются приближениями к множителям Лагранжа ограничений-неравенств (7) задачи (5)–(8). Причем, $\xi_j^k = x_j^{k-1} - \underline{x}_j$, $j \in L$, а $\psi_j^k = \bar{x}_j - x_j^{k-1}$, $j \in H$. Величина x_j^{k-1} – это компонента вектора множителей Лагранжа x^{k-1} ограничений-равенств (6). Вектор x^{k-1} вычисляется на предыдущей итерации алгоритма (для первой итерации алгоритма зададим $\xi_j^0=1$, $\psi_j^0=1$).

Найдем решение задачи (37)–(39) используя правило множителей Лагранжа. Выразим компоненты векторов Δy , Δl , Δh из

системы, полученной приравниванием нулю производных функции Лагранжа:

$$(42) \Delta y_j = (x_j - \varphi_j(y_j^k)) / \varphi_j'(y_j^k), \quad j \in J,$$

$$(43) \Delta l_j = (\underline{x}_j - x_j) q_j^k, \quad j \in L,$$

$$(44) \Delta h_j = (x_j - \bar{x}_j) p_j^k, \quad j \in H.$$

Подставим полученные выражения в (38), выразим из получившихся уравнений вектор двойственных оценок x :

$$(45) x = H^k (A^T \Delta u - \Pi^k).$$

Здесь H^k – диагональная матрица с n элементами вида $((\varphi_j'(y_j^k))^{-1} + q_j^k + p_j^k)^{-1}$ на главной диагонали; Π^k – диагональная матрица с n элементами вида $(c_j - \varphi_j'(x_j^k)(\varphi_j'(x_j^k))^{-1} - \underline{x}_j q_j^k + \bar{x}_j p_j^k)$ на главной диагонали. Для элементов матриц H^k и Π^k справедливы соотношения:

$$(46) q_j^k = 0, \quad j \in J \setminus L; \quad p_j^k = 0, \quad j \in J \setminus H.$$

Подставим выражение (45) для x в уравнение (2). Получим систему линейных уравнений относительно вектора Δu :

$$(47) AH^k(A^T \Delta u - \Pi^k) = b.$$

Считаем, что матрица A имеет полный ранг. Тогда матрица AH^kA^T – симметричная, положительно определенная. Решив систему (47), получим вектор Δu^k . Затем найдем векторы Δy^k , Δl^k , Δh^k , используя (39), (42)–(44), и вектор двойственных переменных x^k , используя (45).

Пункт 2. Проверка выполнения критерия останова. Проверим для текущего приближения к решению выполнение условий (2)–(4), (8), (9)–(12). Рассчитаем следующие величины:

$$\pi_{xY}^k = \max \{ |\varphi_j(y_j^k) - x_j^k| : j \in J \},$$

$$\pi_L^k = \max \{ (f_j(\underline{x}_j) + c_j - [A^T u^k]_j)_+ - l_j^k : j \in L \},$$

$$\pi_H^k = \max \{ ([A^T u^k]_j - f_j(\bar{x}_j) - c_j)_+ - h_j^k : j \in H \},$$

$$\pi_{xL}^k = \max \{ (\underline{x}_j - x_j^k)_+ : j \in L \}, \quad \pi_{xH}^k = \max \{ (x_j^k - \bar{x}_j)_+ : j \in H \}.$$

Если справедливо условие $\max\{\pi_{XY}^k, \pi_L^k, \pi_H^k, \pi_{xL}^k, \pi_{xH}^k\} < \varepsilon$, где ε – параметр допустимой погрешности вычислений, то условия оптимальности решаемой задачи оптимизации выполнены с требуемой точностью. В этом случае завершаем работу алгоритма, иначе переходим к пункту 3.

Пункт 3. *Выбор шага корректировки текущего приближения.* Сначала определим шаг корректировки до ближайшей границы допустимой области, задаваемой ограничениями-неравенствами. Шаг до ближайшей границы определяется по формуле $\lambda_{LH} = \min\{\lambda_L, \lambda_H\}$, где

$$\lambda_L = \min\{(-l_j^k)/\Delta l_j^k : j \in L, \Delta l_j^k < 0\},$$

$$\lambda_H = \min\{(-h_j^k)/\Delta h_j^k : j \in H, \Delta h_j^k < 0\}.$$

Найдем величину λ_p , решив задачу одномерной минимизации целевой функции (5) по направлению, задаваемому векторами $\Delta y^k, \Delta u^k, \Delta l^k, \Delta h^k$:

$$(48) \quad \lambda_p = \arg \min_{\alpha \in R} \{\Phi(y^k + \alpha \Delta y^k) - b^T(u^k + \alpha \Delta u^k) - \sum_{j \in L} \bar{x}_j(l_j^k + \alpha \Delta l_j^k) + \sum_{j \in H} \bar{x}_j(h_j^k + \alpha \Delta h_j^k) : 0 < \alpha \leq \lambda_{LH}\}.$$

Вычислим итоговый шаг λ_k корректировки решения:

$$(49) \quad \lambda_k = \min(\gamma \lambda_{LH}, \lambda_p), \text{ где } 0 < \gamma < 1, \text{ (например, } \gamma = 0,7).$$

Пункт 4. Вычисление следующего приближения. Осуществим итеративный переход, используя шаг корректировки λ_k :

$$y^{k+1} = y^k + \lambda_k \Delta y^k; \quad u^{k+1} = u^k + \lambda_k \Delta u^k;$$

$$l_j^{k+1} = l_j^k + \lambda_k \Delta l_j^k, \quad j \in L; \quad l_j^k = 0, \quad j \in J \setminus L;$$

$$h_j^{k+1} = h_j^k + \lambda_k \Delta h_j^k, \quad j \in H; \quad h_j^k = 0, \quad j \in J \setminus H.$$

5. Результаты экспериментальных исследований

Была выполнена программная реализация прямых и двойственных алгоритмов внутренних точек для решения задач потокораспределения на языке C++.

В экспериментальных расчетах использовались два варианта прямого и два варианта двойственного алгоритмов внутренних точек с двумя видами весовых коэффициентов: квадратичными и линейными коэффициентами, деленными на приближения к множителям Лагранжа. В таблице 1 приводятся характеристики решенных в ходе эксперимента задач и результаты расчетов для них. Критерием остановки алгоритмов было выполнение условий (2)–(4), (8), (9)–(12) с максимальной невязкой не более 0,1.

Таблица 1. Результаты расчетов задач потокораспределения с использованием четырех вариантов алгоритмов

Характеристики задач				Количество итераций для вариантов алгоритмов внутренних точек			
Узлов	Ветвей	Двуст. ограничен.	Активн. огранич.	Прямой	Двойств	Прямой	Двойств
				Квадратичн. весовые коэффиц.		Линейные весовые коэффиц.	
25	39	25	3	30	47	24	16
25	39	35	9	63	42	38	20
25	48	40	4	25	24	15	15
25	48	40	11	112	46	60	20
50	60	25	7	47	35	37	23
50	60	50	8	83	41	74	24
50	136	88	39	118	64	25	31
50	136	100	19	70	34	25	21
100	116	20	10	64	15	42	24
100	116	81	7	70	39	24	22
100	195	79	37	65	250	32	27
100	195	90	15	44	28	32	26
200	300	150	10	56	31	26	18
200	300	150	26	121	39	39	23
338	712	500	11	108	86	27	23
338	712	500	16	96	79	39	31
Среднее геометрическое:				66,7	44,4	32,5	22,3

Расчеты показывают, что прямой и двойственный алгоритмы с линейными весовыми коэффициентами в среднем в два раза быстрее (по числу итераций) своих аналогов с квадратич-

ными коэффициентами. Двойственные алгоритмы в среднем в полтора раза быстрее своих прямых аналогов.

Кроме того, был проведен эксперимент для сравнения скорости достижения требуемой точности решения прямым и двойственным алгоритмом по исходным и по двойственным переменным. Точность проверялась относительно предварительно найденного с большой точностью решения. В таблице 2 приводятся результаты расчетов по выборке из 6 задач, для которых условия (2)–(4), (8), (9)–(12) выполнялись с точностью 0,01 на последней итерации.

Таблица 2. Сравнение скорости достижения требуемой точности решения по исходным и по двойственным переменным

	Расчеты прямым алгоритмом			Расчеты двойственным алгоритмом		
	Кол-во итерационных рац.	Точность решения по исходным перемен.	Точность решения по двойствен. перемен.	Кол-во итерационных рац.	Точность решения по исходным перемен.	Точность решения по двойствен. перемен.
25, 39	29	8,04E-06	2,85E-08	20	2,85E-08	1,20E-03
25, 48	29	1,04E-05	7,91E-07	21	2,90E-08	2,60E-03
50, 60	34	7,95E-05	3,11E-07	24	1,27E-08	1,59E-03
50, 136	32	2,26E-05	1,83E-07	25	4,21E-06	1,07E-03
100, 116	31	1,51E-05	4,70E-07	32	1,26E-09	1,53E-03
100, 195	30	4,05E-06	4,19E-08	24	2,00E-06	1,81E-03

Из таблицы 2 следует, что при использовании прямого алгоритма требуемая точность решения достигается быстрее по двойственным переменным, чем по исходным. Для двойственного алгоритма требуемая точность достигается быстрее по исходным переменным, чем по двойственным.

5. Заключение

Проведены экспериментальные исследования прямых и двойственных алгоритмов внутренних точек с различными способами выбора весовых коэффициентов на нелинейных задачах потокораспределения.

Показано, что варианты прямых и двойственных алгоритмов с линейными весовыми коэффициентами, деленными на множители Лагранжа, в среднем в два раза быстрее по числу итераций, чем варианты алгоритмов с квадратичными весовыми коэффициентами. Установлено, что при использовании двойственных алгоритмов требуемая точность решения достигается в среднем в полтора раза быстрее, чем при использовании прямых алгоритмов.

Сделан вывод, что при использовании прямого алгоритма требуемая точность решения достигается быстрее по двойственным переменным, чем по исходным. Для двойственного алгоритма требуемая точность достигается быстрее по исходным переменным, чем по двойственным.

В связи с результатами экспериментов справедлива следующая рекомендация: для более быстрого нахождения решения исходной задачи с заданной точностью целесообразно использовать двойственные алгоритмы внутренних точек с линейными весовыми коэффициентами, деленными на множители Лагранжа.

Литература

1. АНЦЫЗ С.М., ДИКИН И.И. *Об одном численном методе решения задачи линейного программирования и некоторых её обобщений* // В кн.: Управляемые системы. Вып. 3. – Новосибирск, ИМ СО АН СССР, 1969. – С. 54–56.
2. ВОЙТОВ О.Н., ЗОРКАЛЬЦЕВ В.И., ФИЛАТОВ А.Ю. *Определение допустимых режимов электроэнергетических систем алгоритмами внутренних точек* // Сибирский журнал индустриальной математики – 2000. – Том 3, №1(5). – С. 57–65.
3. ДЕННИС Дж.Б. *Математическое программирование и электрические цепи*. – М.: Изд-во иностранной литературы, 1961, – 216 с.
4. ДИКИН И.И. *Итеративное решение задач линейного программирования* // Доклады АН СССР. – 1967. – Т. 174. – С 747–748.

5. ДИКИН И.И., ЗОРКАЛЬЦЕВ В.И. *Итеративное решение задач математич. программирования (алгоритмы метода внутренних точек)*. – Новосибирск: Наука, 1980. – 144 с.
6. ЕВТУШЕНКО Ю.Г., ЖАДАН В.Г. *Релаксационный метод решения задач нелинейного программирования* // Журнал вычислительной математики и математической физики. – 1977. – Том 17, №4. – С. 890–904.
7. ЕПИФАНОВ С.П., ЗОРКАЛЬЦЕВ В.И., МЕДВЕЖОНКОВ Д.С. *Модель гидравлической сети с регуляторами расхода* // Управление большими системами. – 2010. – Специальный выпуск 30.1 «Сетевые модели в управлении». – С. 286–299.
8. ЗАНГВИЛЛ У. *Нелинейное программирование. Единый подход*. – Пер. с англ. под ред. Е.Г. Гольштейна. – М.: «Сов. радио», 1973. – 312 с.
9. ЗОРКАЛЬЦЕВ В.И. *Итеративный алгоритм решения задачи линейного программирования* // В кн.: Алгоритмы и программы решения задач линейной алгебры и мат. программирования. – Иркутск, СЭИ СО АН СССР, 1978. – С. 77–89.
10. ЗОРКАЛЬЦЕВ В.И. *Класс алгоритмов внутренних точек* // Журнал вычислительной математики и математической физики. – 2009. – №12. – С. 3–28.
11. ЗОРКАЛЬЦЕВ В.И. *Методы прогнозирования и анализа эффективности функционирования системы топливоснабжения*. – М.: Наука, 1988. – 144 с.
12. ЗОРКАЛЬЦЕВ В.И. *Проективные алгоритмы оптимизации, использующие множители предыдущей итерации* // Журнал вычисл. Математики и матем. физики. – 1994. – Т. 34, №7. – С. 943–950.
13. ЗОРКАЛЬЦЕВ В.И. *Симметричная двойственность в оптимизации и ее приложения* // Известия высших учебных заведений. Математика, 2006. – №2. – С. 53–59.
14. ЗОРКАЛЬЦЕВ В.И., ХАМИСОВ О.В. *Равновесные модели в экономике и энергетике*. – Новосибирск: Наука, 2006. – 221 с.

15. BERTSEKAS D.P. *Network Optimization: Continuous and Discrete Models*. – Athena Scientific, Belmont, Massachusetts, 1998. – 608 p.
16. ROCKAFELLAR R. *Tyrrell: Network flows and monotropic optimization*. – Pure and Applied Mathematics. – New York: Wiley-Interscience, 1984. – 616 p.

COMPUTATIONAL EXPERIMENTS WITH VARIANTS OF INTERIOR-POINT ALGORITHMS FOR NONLINEAR FLOW DISTRIBUTION PROBLEMS

Dmitry Medvezhonkov, Institute of Energy Systems of SB RAS, Irkutsk, junior research assistant (dmitry@isem.sei.irk.ru).

Valery Zorkaltsev, Institute of Energy Systems of SB RAS, Irkutsk, Doctor of Science, prof., head of laboratory (zork@isem.sei.irk.ru).

Abstract: We present results of computational experiments which compare variants of primal and dual interior-point algorithms for nonlinear problems of flow distribution. The experiments show that linear weight coefficients divided by Lagrange multipliers dominate quadratic weight coefficients. We also show that the required solution accuracy is achieved faster when using dual algorithms, rather than primal ones.

Keywords: primal and dual interior-point algorithms, weight coefficients, nonlinear flow distribution problems.

Статья представлена к публикации членом редакционной коллегии М.В. Губко

*Поступила в редакцию 18.04.2013.
Опубликована 30.11.2013.*

УДК 021.8 + 025.1
ББК 78.34

О СРЕДНИХ ВЕЛИЧИНАХ

Орлов А. И.¹

*(Московский государственный технический университет
им. Н.Э. Баумана, Москва)*

Приведены новые результаты в теории средних. Введены взвешенные средние I типа, соответствуют элементам выборки, и II типа, соответствующие членам вариационного ряда. Прослежена эволюция представлений о расстоянии Кемени и медиане Кемени. Предложена модифицированная медиана Кемени, удобная для вычислений и позволяющая избежать эффекта «центра дырки от бублика». Как обобщение медианы Кемени введены и изучены эмпирические и теоретические средние в пространствах произвольной природы. Для них доказаны законы больших чисел.

Ключевые слова: взвешенные средние, расстояние Кемени, медиана Кемени, эмпирические средние, теоретические средние, законы больших чисел.

1. Введение

В системном анализе и теории принятия решений широко используются средние величины [16, 19]. Классические результаты теории средних собраны в монографии главы итальянской статистической школы Коррадо Джини [3]. За прошедшее время получены новые результаты, которым и посвящена статья.

Так, при усреднении чисел обнаружено наличие двух типов взвешенных средних – I типа, соответствующих элементам

¹ Александр Иванович Орлов, доктор технических наук, доктор экономических наук, кандидат физико-математических наук, профессор (prof-orlov@mail.ru).

выборки, и II типа, соответствующих членам вариационного ряда.

Развитие математического инструментария решения прикладных задач, прежде всего в экспертных технологиях и социологии, привело к необходимости использования средних значений в пространствах нечисловой природы. Сначала в качестве средних значений бинарных отношений применяли медианы Кемени. Затем оптимизационный подход к построению средних величин стал стержнем нечисловой статистики [17] – новой области прикладной математической статистики. Эти новые разделы теории средних величин также рассмотрены в статье.

2. Взвешенные средние I и II типов

Исходные статистические данные, обработка которых необходима для принятия управленческих решений, во многих случаях рассматриваются как выборка x_1, x_2, \dots, x_n , т.е. значения n независимых случайных величин для некоторого элемента пространства элементарных событий, на котором они определены. Если упорядочить элементы выборки, то получим вариационный ряд $x(1) \leq x(2) \leq x(3) \leq \dots \leq x(n)$.

Для описания совокупности в целом используют средние величины. При реальных расчетах применяют средние двух типов. Первый тип – степенные средние, частными случаями которых являются среднее арифметическое, среднее квадратическое, среднее гармоническое. Среднее геометрическое является пределом степенного среднего, когда показатель степени стремится к 0). Более общим видом средних являются средние по Колмогорову:

$$(1) \quad F^{-1} \left(\frac{1}{n} \sum_{i=1}^n F(x_i) \right),$$

где F – строго монотонная функция, F^{-1} – обратная к ней.

Второй тип – структурные средние, прежде всего медиана и мода, а также члены вариационного ряда, минимум, максимум, квартили, децили.

Применение теории измерений позволило установить, как выбирать средние в соответствии со шкалами, в которых измерены исходные данные [16, 17, 19]. Так, для данных, измеренных в порядковой шкале, допустимыми средними являются только члены вариационного ряда. В частности, при нечетном объеме выборки – выборочная медиана, при четном – левая и правая медианы (т.е. два центральных члена вариационного ряда). В шкале интервалов из всех средних по Колмогорову можно использовать только среднее арифметическое. В шкале отношений из всех средних по Колмогорову допустимы только степенные средние и среднее геометрическое.

Взвешенные средние (синоним: средние взвешенные) – это средние величины, в которых усредняемые величины учитываются по-разному, в соответствии с весовыми коэффициентами. Выделим два типа взвешенных средних. Для средних I типа весовые коэффициенты соответствуют элементам выборки. Для средних II типа весовые коэффициенты соответствуют членам вариационного ряда.

Пусть a_1, a_2, \dots, a_n – весовые коэффициенты (веса), т.е. неотрицательные числа, в сумме составляющие 1. Удобно ввести случайные величины $X(a_1, a_2, \dots, a_n)$ и $Y(a_1, a_2, \dots, a_n)$, такие, что $P(X=x_1) = a_1, P(X=x_2) = a_2, \dots, P(X=x_n) = a_n$, в то время как $P(Y=x(1)) = a_1, P(Y=x(2)) = a_2, \dots, P(Y=x(n)) = a_n$. Таким образом, случайные величины X и Y принимают одни и те же значения (перечисленные в выборке x_1, x_2, \dots, x_n), но, вообще говоря, с разными вероятностями. Если все веса равны между собой (и равны $1/n$), то распределения случайных величин X и Y совпадают (и называются эмпирическим распределением).

Взвешенные средние легко определить с помощью введенных случайных величин X и Y . Среднее взвешенное арифметическое I типа – это математическое ожидание X , т.е.

$$(2) \quad M(X) = \sum_{i=1}^n a_i x_i .$$

Среднее взвешенное арифметическое II типа – это математическое ожидание Y , т.е.

$$(3) \quad M(Y) = \sum_{i=1}^n a_i x(i).$$

Ясно, что результаты расчетов по формулам (2) и (3), вообще говоря, различны.

Среднее взвешенное по Колмогорову I типа – это

$$(4) \quad F^{-1}(M(F(X))) = F^{-1}\left(\frac{1}{n} \sum_{i=1}^n a_i F(x_i)\right).$$

Среднее взвешенное по Колмогорову II типа – это

$$(5) \quad F^{-1}(M(F(Y))) = F^{-1}\left(\frac{1}{n} \sum_{i=1}^n a_i F(x(i))\right).$$

Аналогично вводятся выборочная взвешенная медиана I типа – медиана случайной величины X , вероятности совпадения которой с элементами выборки равны заданным весам, и выборочная взвешенная медиана II типа – медиана случайной величины Y , вероятности совпадения которой с членами вариационного ряда равны заданным весам.

При использовании взвешенных средних величин в задачах системного анализа и принятия решений необходимо указывать тип средних, поскольку от типа средних зависят численные значения. В Московском государственном техническом университете им. Н.Э. Баумана при преподавании дисциплины «Прикладная статистика» и в Московском физико-техническом институте при преподавании дисциплины «Методы анализа данных» понятия средних взвешенных величин I типа и II типа входят в учебные программы, однако в научной и учебной литературе разделение средних взвешенных величин I типа и II типа нам не встречалось. Разделение необходимо, поскольку для одних и тех же исходных числовых данных значения одноименных средних взвешенных I и II типов различаются, что при практической работе может привести к недоразумениям.

3. Расстояние Кемени и медиана Кемени

При анализе истории и нынешнего состояния экспертных оценок в нашей стране [12, 18, 20] выявлена исключительная роль расстояния и медианы Кемени в развитии теории экспертных оценок. Несколько ранее была окончательно сформулирована в развернутом виде современная парадигма прикладной статистики [15, 17], и оказалось, что центральным ядром статистических методов является статистика в пространствах произвольной природы, а центральным результатом – закон больших чисел в таких пространствах, состоящий в стремлении эмпирических средних к теоретическому, причем определение эмпирического среднего является непосредственным обобщением медианы Кемени, построенной с помощью расстояния Кемени. Таким образом, в этих двух взаимосвязанных научно-практических областях центральным понятием является медиана Кемени, что оправдывает ее подробное обсуждение в настоящей статье.

Обсудим, какой смысл вкладывают сегодня в понятие «расстояние Кемени».

Первоисточник – книга Дж. Кемени и Дж. Снелла «Mathematical Models in the Social Sciences» (1963), изданная на русском языке в 1972 г. в переводе Б.Г. Миркина под названием «Кибернетическое моделирование. Некоторые приложения» [6]. В ней рассмотрен класс бинарных отношений, названных «упорядочениями». Различные авторы использовали также названия «ранжировки со связями» [23], «квазисерии» [9], «совершенные квазипорядки» [24]. В настоящее время мы используем термин «кластеризованные ранжировки» [2, 15, 17, 20].

Первое достижение Дж. Кемени – аксиоматическое введение расстояния между кластеризованными ранжировками (из предыдущих работ Дж. Кемени (см., в частности, [26]) ясно, что авторство этого достижения принадлежит именно ему).

Как известно, любое бинарное отношение, определенное на конечном множестве из k элементов, может быть поставлено в соответствие квадратной матрице из 0 и 1 порядка k . Пусть A и

B – два бинарных отношения, которым соответствуют матрицы $\|a(i, j)\|$ и $\|b(i, j)\|$ соответственно. В [6] приведена система аксиом, из которой выведен вид расстояния $d(A, B)$ между кластеризованными ранжировками A и B :

$$(6) \quad d(A, B) = \sum_{1 \leq i, j \leq k} |a(i, j) - b(i, j)|$$

(вид расстояния из [6] приведен с точностью до множителя). Этот результат Дж. Кемени породил большое число аналогичных исследований, посвященных выводу вида тех или иных расстояний в различных пространствах из подходящих систем аксиом. Так, Б.Г. Миркин и Л.Б. Черный вывели расстояние (6) между упорядоченными [10] и неупорядоченными [11] разбиениями, автор этой статьи – между толерантностями (см., например, [15, 17]). В сводке (не вполне полной) работ по этому направлению [22] приведена 161 литературная ссылка.

Автор настоящей статьи должен взять на себя ответственность за перенос названия «расстояние Кемени» на расстояния вида (6) в произвольных пространствах бинарных отношений. Именно такое определение мы включили в энциклопедию «Вероятность и математическая статистика» [1, с. 230].

Рассмотрим различные варианты медиан Кемени.

Пусть ответы экспертов представлены как упорядочения (кластеризованные ранжировки) $A(1), A(2), \dots, A(n)$. В книге Дж. Кемени и Дж. Снелла [6] в качестве итогового мнения комиссии экспертов предложено применять «медиану Кемени», т.е. результат минимизации суммы расстояний Кемени от мнений экспертов до произвольного бинарного отношения X :

$$(7) \quad med(A(1), A(2), \dots, A(n)) = \arg \min_{X \in Z} \sum_{1 \leq j \leq n} d(A(j), X).$$

В [6] мнения экспертов – кластеризованные ранжировки, минимизация проводится по пространству Z всех кластеризованных ранжировок. Автор настоящей статьи должен взять на себя ответственность за перенос названия «медиана Кемени» на случай произвольных пространств бинарных отношений [1, с. 229–230]. При этом согласно определению в Энциклопедии

дии минимизация проводится по *тому же* пространству бинарных отношений, в котором лежат мнения экспертов.

Напрашивается следующий шаг в обобщении медианы Кемени: пусть ответы экспертов лежат в пространстве W , в то время как минимизация проводится по пространству Z . В общем виде такая постановка впервые обсуждалась в [21].

Если пространства Z и W различны, возникают новые эффекты. Например, если Z – пространство всех бинарных отношений, то согласно формуле (7) медиана Кемени находится элементарно – по правилу большинства (если в определенной клетке описывающих мнения экспертов матриц единиц больше половины, то в итоговой матрице ставим единицу, аналогично для нулей; если ровно половина, то можно поставить либо 0, либо 1, медиана Кемени является множеством из нескольких бинарных отношений). Этот факт с завидной регулярностью обнаруживают и публикуют наивные авторы. В то же время если W и Z – одно и то же пространство кластеризованных ранжировок, то известные алгоритмы Б.Г. Литвака [8] и В.Н. Жихарева [4, 17] нахождения медианы Кемени достаточно сложны.

Множество B , по которому проводится минимизация, может не быть пространством бинарных отношений, а составлено по иным принципам. В педагогических целях мы использовали в учебниках [15, 17, 20] множество B из небольшого числа элементов (порядка 10), что позволяло ограничиться ручным счетом.

Вместо рассмотренной выше классической медианы Кемени предлагаем применять *модифицированную медиану Кемени*, в которой $B = \{A(1), A(2), \dots, A(n)\}$. Для нее итоговое мнение комиссии экспертов всегда совпадает с мнением одного из экспертов, что позволяет избежать эффекта «центра дырки от бублика» (если предположить, что мнения экспертов равномерно распределены по поверхности тора, то классическая медиана Кемени – центр «дырки от бублика», что делает ее расчет бессмысленным).

Изучение свойств медианы Кемени продолжается [28, 29]. Накапливается опыт ее практического применения [7, 25].

С научной точки зрения актуально сравнение медианы Кемени с другими методами нахождения коллективного мнения экспертов, мнения которых выражены бинарными отношениями. В частности, если ответы экспертов – кластеризованные ранжировки, то методы на основе медианы Кемени и модифицированной медианы Кемени целесообразно сопоставить с методами средних арифметических рангов, медиан рангов, согласования кластеризованных ранжировок [2].

4. Эмпирические и теоретические средние

Одна из основных статистических процедур – вычисление средних величин для тех или иных совокупностей данных. Законы больших чисел гласят: эмпирические средние сходятся к теоретическим. В классическом варианте: выборочное среднее арифметическое при определенных условиях сходится по вероятности при росте числа слагаемых к математическому ожиданию. На основе законов больших чисел обычно доказывают состоятельность различных статистических оценок. В целом эта тематика занимает заметное место в теории вероятностей и математической статистике.

Однако математический аппарат при этом основан на свойствах сумм случайных величин (векторов, элементов линейных пространств). Следовательно, он непригоден для изучения вероятностных и статистических проблем, связанных со случайными объектами нечисловой природы. Это такие объекты, как бинарные отношения, нечеткие множества, вообще элементы пространств без векторной структуры. Объекты нечисловой природы все чаще встречаются в прикладных исследованиях. Много конкретных примеров приведено в [15, 17, 20]. Поэтому необходимо научиться усреднять различные нечисловые данные, т.е. определять эмпирические и теоретические средние в пространствах произвольной природы. Кроме того, представля-

ется полезным получение законов больших чисел в пространствах нечисловой природы.

Для осуществления описанной научной программы необходимо решить следующие задачи.

А. Определить понятие эмпирического среднего.

Б. Определить понятие теоретического среднего.

В. Ввести понятие сходимости эмпирических средних к теоретическому.

Г. Доказать при тех или иных комплексах условий сходимость эмпирических средних к теоретическому.

Д. Обобщив это доказательство, получить метод обоснования состоятельности различных статистических оценок.

Е. Дать применения полученных результатов при решении конкретных задач.

Образцом для определения средних величин является медиана Кемени. Пусть X – пространство произвольной природы, $x_1, x_2, x_3, \dots, x_n$ – его элементы. Чтобы ввести эмпирическое среднее для $x_1, x_2, x_3, \dots, x_n$, будем использовать действительнозначную (т.е. с числовыми значениями) функцию $f(x, y)$ двух переменных со значениями в X . В стандартных математических обозначениях: $f: X^2 \rightarrow R^1$. Величина $f(x, y)$ интерпретируется как показатель различия между x и y : чем $f(x, y)$ больше, тем x и y сильнее различаются. В качестве f можно использовать расстояние в X (в частности, расстояние Кемени, если X – одно из пространств бинарных отношений), квадрат расстояния и т.п.

Определение 1. Средней величиной для совокупности x_1, x_2, \dots, x_n (относительно меры различия f), обозначаемой любым из трех способов: $x_{cp} = E_n(f) = E_n(x_1, x_2, x_3, \dots, x_n; f)$, называем решение оптимизационной задачи

$$(8) \quad \sum_{i=1}^n f(x_i, y) \rightarrow \min, \quad y \in X.$$

Это определение согласуется с классическими определениями средних величин. Если $X = R^1$, $f(x, y) = (x - y)^2$, то x_{cp} – выборочное среднее арифметическое. Если же $X = R^1$, $f(x, y) = |x - y|$, то при $n = 2k + 1$ имеем $x_{cp} = x(k + 1)$, при $n = 2k$ эмпирическое среднее является отрезком $[x(k), x(k + 1)]$. Здесь

через $x(i)$ обозначен i -й член вариационного ряда, построенного по x_1, x_2, \dots, x_n , т.е. i -я порядковая статистика. Таким образом, при $X = R^1$, $f(x, y) = |x - y|$ решение задачи (8) дает естественное определение выборочной медианы. Правда, несколько отличающееся от определения, обычно предлагаемого в курсах «Общей теории статистики», в котором при $n = 2k$ медианой называют полусумму двух центральных членов вариационного ряда $(x(k) + x(k+1))/2$. Иногда $x(k)$ называют левой медианой, а $x(k+1)$ – правой медианой [17].

Решением задачи (8) является множество $E_n(f)$, которое может быть пустым, состоять из одного или многих элементов. Выше приведен пример, когда решением является отрезок. Если $X = R^1 \setminus \{x_0\}$, $f(x, y) = (x - y)^2$, а среднее арифметическое выборки равно x_0 , то $E_n(f)$ пусто.

При моделировании реальных ситуаций часто можно принять, что X состоит из конечного числа элементов. Тогда множество $E_n(f)$ всегда непусто – минимум на конечном множестве обязательно достигается.

Понятия случайного элемента $x = x(\omega)$ со значениями в X , его распределения, независимости случайных элементов используем согласно [1]. Будем считать, что функция f измерима относительно σ -алгебры, участвующей в определении случайного элемента $x = x(\omega)$. Тогда $f(x(\omega), y)$ при фиксированном y является действительной случайной величиной. Предположим, что она имеет математическое ожидание.

Определение 2. Теоретическим средним $E(x, f)$ (другими словами, математическим ожиданием) случайного элемента $x = x(\omega)$ относительно меры различия f называется решение оптимизационной задачи

$$(9) \quad Mf(x(\omega), y) \rightarrow \min, \quad y \in X.$$

Это определение, как и для эмпирических средних, согласуется с классическим. Если $X = R^1$, $f(x, y) = (x - y)^2$, то $E(x, f) = M(x(\omega))$ – обычное математическое ожидание. При этом $Mf(x(\omega), y)$ – дисперсия случайной величины $x = x(\omega)$. Если же $X = R^1$, $f(x, y) = |x - y|$, то $E(x, f) = [a, b]$, здесь $a = \sup\{t:$

$F(t) \leq 0,5$, $b = \inf\{t: F(t) \geq 0,5\}$, где $F(t)$ – функция распределения случайной величины $x = x(\omega)$. Если график $F(t)$ имеет плоский участок на уровне $F(t) = 0,5$, то медиана – теоретическое среднее в смысле определения 2 – является отрезком. В классическом случае обычно говорят, что каждый элемент отрезка $[a; b]$ является одним из возможных значений медианы. Поскольку наличие указанного плоского участка – исключительный случай, то обычно решением задачи (9) является множество из одного элемента $a = b$ – классическая медиана распределения случайной величины $x = x(\omega)$.

Теоретическое среднее $E(x, f)$ можно определить лишь тогда, когда $Mf(x(\omega), y)$ существует при всех $y \in X$. Оно может быть пустым множеством, например, если $X = R^1 \setminus \{x_0\}$, $f(x, y) = (x - y)^2$, $x_0 = M(x(\omega))$. И то, и другое исключается, если X конечно. Однако и для конечных X теоретическое среднее может состоять не из одного, а из многих элементов. Отметим, однако, что в множестве всех распределений вероятностей на X подмножество тех распределений, для которых $E(x, f)$ состоит более чем из одного элемента, имеет коразмерность 1, поэтому основной является ситуация, когда множество $E(x, f)$ содержит единственный элемент.

5. Существование средних величин

Под существованием средних величин будем понимать непустоту множеств решений соответствующих оптимизационных задач.

Если X состоит из конечного числа элементов, то минимум в задачах (8) и (9) берется по конечному множеству. А потому, как уже отмечалось, эмпирические и теоретические средние существуют.

Ввиду важности проблемы существования средних величин в пространствах общей природы приведем теоремы и их доказательства. Топологические термины и результаты будем использовать в соответствии с классической монографией [5]. Так, топологическое пространство называется бикompактным в том и

только в том случае, когда из каждого его открытого покрытия можно выбрать конечное подпокрытие [5, с. 183].

Теорема 1. Пусть X – бикompактное пространство, функция f непрерывна на X^2 (в топологии произведения). Тогда эмпирическое и теоретическое средние существуют.

Доказательство. Функция $f(x_i, y)$ от y непрерывна, сумма непрерывных функций непрерывна, непрерывная функция на бикompакте достигает своего минимума, откуда и следует заключение теоремы относительно эмпирического среднего.

Перейдем к теоретическому среднему. По теореме Тихонова [5, с. 194] из бикompактности X вытекает бикompактность X^2 . Для каждой точки (x, y) из X^2 рассмотрим $\varepsilon/2$ -окрестность в X^2 в смысле показателя различия f , т.е. множество

$$(10) U(x, y) = \{(x', y') : |f(x, y) - f(x', y')| < \varepsilon / 2\}.$$

Поскольку f непрерывна, то множества $U(x, y)$ открыты в рассматриваемой топологии в X^2 . По теореме Уоллеса [5, с. 193] существуют открытые (в X) множества $V(x)$ и $W(y)$, содержащие x и y соответственно и такие, что их декартово произведение $V(x) \times W(y)$ целиком содержится внутри $U(x, y)$.

Рассмотрим покрытие X^2 открытыми множествами $V(x) \times W(y)$. Из бикompактности X^2 вытекает существование конечного подпокрытия $\{V(x_i) \times W(y_i), i = 1, \dots, m\}$. Для каждого x из X рассмотрим все декартовы произведения $V(x_i) \times W(y_i)$, куда входит точка (x, y) при каком-либо y . Таких декартовых произведений и их первых множителей $V(x_i)$ конечное число. Возьмем пересечение таких первых множителей $V(x_i)$ и обозначим его $Z(x)$. Это пересечение открыто, как пересечение конечного числа открытых множеств, и содержит точку x . Из покрытия бикompактного пространства X открытыми множествами $Z(x)$ выберем открытое подпокрытие Z_1, Z_2, \dots, Z_k .

Покажем, что если $x(1)$ и $x(2)$ принадлежат одному и тому же Z_j при некотором j , то

$$(11) \sup\{|f(x(1), y) - f(x(2), y)|, y \in X\} < \varepsilon.$$

Пусть $Z_j = Z(x_0)$ при некотором x_0 . Пусть $V(x_i) \times W(y_i)$, $i \in I$, – совокупность всех тех исходных декартовых произведе-

ний из системы $\{V(x_i) \times W(y_i), i = 1, \dots, m\}$, куда входят точки (x_0, y) при различных y . Покажем, что их объединение содержит также точки $(x(1), y)$ и $(x(2), y)$ при всех y . Действительно, если (x_0, y) входит в $V(x_i) \times W(y_i)$, то y входит в $W(y_i)$, а $x(1)$ и $x(2)$ вместе с x_0 входят в $V(x_i)$, поскольку $x(1)$, $x(2)$ и x_0 входят в $Z(x_0)$. Таким образом, $x(1)$ и $x(2)$ принадлежат $V(x_i) \times W(y_i)$, а потому согласно определению $V(x_i) \times W(y_i)$ справедливы неравенства

$$(12) |f(x(\alpha), y) - f(x_i, y_i)| < \varepsilon / 2, \quad \alpha = 1, 2,$$

откуда и следует неравенство (11).

Поскольку X^2 – бикомпактное пространство, то функция f ограничена на X^2 , а потому существует математическое ожидание $Mf(x(\omega), y)$ для любого случайного элемента $x(\omega)$, удовлетворяющего приведенным выше условиям согласования топологии, связанной с f , и измеримости, связанной с $x(\omega)$. Если x_1 и x_2 принадлежат одному открытому множеству Z_j , то $|Mf(x_1, y) - Mf(x_2, y)| < \varepsilon$, а потому функция $g(y) = Mf(x(\omega), y)$ непрерывна на X . Поскольку непрерывная функция на бикомпактном множестве достигает своего минимума, т.е. существуют такие точки z , на которых $g(z) = \inf\{g(y), y \in X\}$, то теорема 1 доказана.

В ряде интересных для приложений ситуаций X не является бикомпактным пространством. Например, если $X = R^1$. В этих случаях приходится наложить на показатель различия f некоторые ограничения, например, так, как это сделано в теореме 2.

Теорема 2. Пусть X – топологическое пространство, непрерывная (в топологии произведения) функция $f: X^2 \rightarrow R^1$ неотрицательна, симметрична (т.е. $f(x, y) = f(y, x)$ для любых x и y из X), существует число $D > 0$ такое, что при всех x, y, z из X

$$(13) f(x, y) \leq D\{f(x, z) + f(z, y)\}.$$

Пусть в X существует точка x_0 такая, что при любом положительном R множество $\{x: f(x, x_0) \leq R\}$ является бикомпактным. Пусть для случайного элемента $x(\omega)$, согласованного с топологией в рассмотренном выше смысле, существует $g(x_0) = Mf(x(\omega), x_0)$.

Тогда существуют (т.е. непусты) математическое ожидание $E(x, f)$ и эмпирические средние $E_n(f)$.

Замечание. Условие (13) – некоторое обобщение неравенства треугольника. Например, если g – метрика в X , а $f = g^p$ при некотором натуральном p , то для f выполнено соотношение (13) с $D = 2^p$.

Доказательство. Изучим функцию $g(y) = Mf(x(\omega), y)$. В соответствии с условием (13) имеем

$$(14) \quad f(x(\omega), y) \leq D\{f(x(\omega), x_0) + f(x_0, y)\}.$$

Поскольку по условию теоремы $g(x_0)$ существует, а потому конечно, то из оценки (14) следует существование и конечность $g(y)$ при всех y из X . Докажем непрерывность этой функции.

Рассмотрим шар (в смысле меры различия f) радиуса R с центром в x_0 , т.е. множество $K(R) = \{x: f(x, x_0) \leq R\}$, $R > 0$. В соответствии с условием теоремы $K(R)$ как подпространство топологического пространства X является бикompактным. Рассмотрим произвольную точку x из X . Справедливо разложение

$$(15) \quad \begin{aligned} f(x(\omega), y) &= f(x(\omega), y)\chi(x(\omega) \in K(R)) + \\ &+ f(x(\omega), y)\chi(x(\omega) \notin K(R)), \end{aligned}$$

где $\chi(C)$ – индикатор множества C . Из разложения (15) следует, что

$$(16) \quad \begin{aligned} g(y) &= Mf(x(\omega), y)\chi(x(\omega) \in K(R)) + \\ &+ Mf(x(\omega), y)\chi(x(\omega) \notin K(R)). \end{aligned}$$

Рассмотрим второе слагаемое в (16). В силу (14)

$$(17) \quad \begin{aligned} f(x(\omega), y)\chi(x(\omega) \notin K(R)) &\leq D\{f(x(\omega), x_0) \times \\ &\times \chi(x(\omega) \notin K(R)) + f(x_0, y)\chi(x(\omega) \notin K(R))\}. \end{aligned}$$

Возьмем математическое ожидание от обеих частей (17):

$$(18) \quad Mf(x(\omega), y)\chi(x(\omega) \notin K(R)) \leq D \int_R^{+\infty} tdP\{f(x(\omega), x_0) \leq t\} + Df(x_0, y)P(x(\omega) \notin K(R)).$$

В правой части (18) оба слагаемых стремятся к 0 при безграничном возрастании R : первое – в силу того, что

$$(19) \quad g(x_0) = Mf(x(\omega), x_0) = \int_0^{+\infty} tdP(f(x(\omega), x_0) \leq t) < \infty,$$

второе – в силу того, что распределение рассматриваемого случайного элемента сосредоточено на X и

$$(20) \quad X \setminus \bigcup_{R>0} K(R) = \emptyset.$$

Пусть $U(x)$ – такая окрестность x (т.е. открытое множество, содержащее x), для которой $\sup\{f(y, x), y \in U(x)\} < +\infty$. Поскольку $f(y, x_0) \leq D(f(x_0, x) + f(x, y))$, то на основе неравенства (18) заключаем, что при безграничном возрастании R

$$(21) \quad Mf(x(\omega), y)\chi(x(\omega) \notin K(R)) \rightarrow 0$$

равномерно по $y \in U(x)$. Пусть $R(0)$ таково, что левая часть (21) меньше $\varepsilon > 0$ при $R > R(0)$ и, кроме того, $y \in U(x) \subseteq K(R(0))$. Тогда при $R > R(0)$

$$(22) \quad |g(y) - g(x)| \leq Mf(x(\omega), y)\chi(x(\omega) \in K(R)) - Mf(x(\omega), x)\chi(x(\omega) \in K(R)) + 2\varepsilon.$$

Нас интересует поведение выражения в правой части формулы (22) при $y \in U(x)$. Рассмотрим f_1 – сужение функции f на замыкание декартова произведения множеств $U(x) \times K(R)$, и случайный элемент $x_1(\omega) = x(\omega)\chi(x(\omega) \in K(R))$. Тогда

$$(23) \quad Mf(x(\omega), y)\chi(x(\omega) \in K(R)) = Mf_1(x_1(\omega), y) = g_1(y)$$

при $y \in U(x)$, а непрерывность функции $g_1(y)$ из (23) была доказана в теореме 1. Последнее означает: существует окрестность $U_1(x)$ точки x такая, что

$$(24) \quad |Mf_1(x_1(\omega), y) - Mf_1(x_1(\omega), x)| < \varepsilon$$

при $y \in U_1(x)$. Из (22) и (24) вытекает, что

$$(25) \quad |g(y) - g(x)| < 3\varepsilon$$

при $y \in U(x) \cap U_1(x)$, что и доказывает непрерывность функции $g(x)$.

Докажем существование математического ожидания $E(x, f)$. Пусть число $R(0)$ таково, что

$$(26) \quad P(x(\omega) \in K(R(0))) > \frac{1}{2}.$$

Пусть H – некоторая константа, значение которой будет выбрано позже. Рассмотрим точку x из множества $K(HR(0))^C$ – дополнения $K(HR(0))$, т.е. из внешности шара радиуса $HR(0)$ с центром в x_0 . Пусть $x(\omega) \in K(R(0))$. Тогда в силу неравенства $f(x_0, x) \leq D\{f(x_0, x(\omega)) + f(x(\omega), x)\}$ имеем

$$(27) f(x(\omega), x) \geq \frac{1}{D} f(x_0, x) - f(x_0, x(\omega)) \geq \frac{HR(0)}{D} - R(0).$$

Выбирая H достаточно большим, получим с учетом условия (26), что при $x \in K(HR(0))^C$ справедливо неравенство

$$(28) Mf(x(\omega), x) \geq \frac{1}{2} \left(\frac{HR(0)}{D} - R(0) \right).$$

Можно выбрать H так, чтобы правая часть (28) превосходила $g(x_0)$.

Сказанное означает, что $\text{Argmin } g(x)$ достаточно искать внутри бикompактного множества $K(HR(0))$. Из непрерывности функции g вытекает, что ее минимум достигается на указанном бикompактном множестве, а потому – и на всем X . Существование (непустота) теоретического среднего $E(x, f)$ доказана.

Докажем существование эмпирического среднего $E_n(f)$. Есть искушение проводить его дословно так же, как и доказательство существования математического ожидания $E(x, f)$, лишь с заменой $1/2$ в формуле (28) на частоту попадания элементов выборки x_i в шар $K(R(0))$. Эта частота, очевидно, стремится к вероятности попадания случайного элемента x в $K(R(0))$, большей $1/2$ в соответствии с (26). Однако это рассуждение показывает лишь, что вероятность непустоты $E_n(f)$ стремится к 1 при безграничном росте объема выборки. Точнее, оно показывает, что

$$(29) \lim_{n \rightarrow \infty} P\{E_n(f) \neq \emptyset \wedge E_n(f) \subseteq K(HR(0))\} = 1.$$

Поэтому пойдем другим путем, не опирающимся к тому же на вероятностную модель выборки. Положим

$$(30) R(1) = \max \{f(x_i, x_0), i = 1, 2, \dots, n\}.$$

Если x входит в дополнение шара $K(HR(1))$, то аналогично (27) имеем

$$(31) f(x_i, x_0) \geq \frac{HR(1)}{D} - R(1).$$

При достаточно большом H из (30) и (31) следует, что

$$(32) \sum_{i=1}^n f(x_i, x_0) \leq nR(1) < \sum_{i=1}^n f(x_i, x), \quad x \in \{K(HR(1))\}^C.$$

Из (32) следует, что Argmin достаточно искать на $K(HR(1))$. Заключение теоремы 2 вытекает из того, что на бикompактном пространстве $K(HR(1))$ минимизируется непрерывная функция.

Теорема 2 полностью доказана. Перейдем к законам больших чисел.

6. О формулировках законов больших чисел

Пусть x, x_1, x_2, \dots, x_n – независимые одинаково распределенные случайные элементы со значениями в X . Закон больших чисел – это утверждение о сходимости эмпирических средних к теоретическому среднему (математическому ожиданию) при росте объема выборки n , т.е. утверждение о том, что

$$(33) E_n(f) = E_n(x_1, x_2, x_3, \dots, x_n; f) \rightarrow E(x, f).$$

при $n \rightarrow \infty$. Однако и слева, и справа в формуле (33) стоят, вообще говоря, множества. Поэтому понятие сходимости в (33) требует обсуждения и определения.

В силу классического закона больших чисел при $n \rightarrow \infty$

$$(34) \frac{1}{n} \sum_{i=1}^n f(x_i, y) \rightarrow Mf(x, y)$$

в смысле сходимости по вероятности, если правая часть существует (теорема А.Я. Хинчина, 1923 г.).

Если пространство X состоит из конечного числа элементов, то из соотношения (34) легко вытекает (см., например, [13, с. 192–193]), что

$$(35) \lim_{n \rightarrow \infty} P\{E_n(f) \subseteq E(x, f)\} = 1.$$

Другими словами, $E_n(f)$ является состоятельной оценкой $E(x, f)$.

Если $E(x, f)$ состоит из одного элемента, $E(x, f) = \{x_0\}$, то соотношение (35) переходит в следующее:

$$(36) \lim_{n \rightarrow \infty} P\{E_n(f) = \{x_0\}\} = 1.$$

Однако с прикладной точки зрения доказательство соотношений (35)–(36) не дает достаточно уверенности в возможности использования $E_n(f)$ в качестве оценки $E(x, f)$. Причина в том, что в процессе доказательства объем выборки предполагается настолько большим, что при всех $y \in X$ одновременно левые части соотношений (34) сосредотачиваются в непересекающихся окрестностях правых частей.

Замечание. Если в соотношении (34) рассмотреть сходимость с вероятностью 1, то аналогично (35) получим так называемый усиленный закон больших чисел [13, с. 193–194]. Согласно этой теореме с вероятностью 1 эмпирическое среднее $E_n(f)$ входит в теоретическое среднее $E(x, f)$, начиная с некоторого объема выборки n , вообще говоря, случайного, $n = n(\omega)$. Мы не будем останавливаться на сходимости с вероятностью 1, поскольку в соответствующих постановках, подробно разобранных в монографии [13], нет принципиальных отличий от случая сходимости по вероятности.

Если X не является конечным, например, $X = R^1$, то соотношения (35) и (36) неверны. Поэтому необходимо искать иные формулировки закона больших чисел. В классическом случае сходимости выборочного среднего арифметического к математическому ожиданию можно записать закон больших чисел так: для любого $\varepsilon > 0$ справедливо предельное соотношение

$$(37) \lim_{n \rightarrow \infty} P\{\bar{x} \in (M(x_1) - \varepsilon, M(x_1) + \varepsilon)\} = 1.$$

В этом соотношении, в отличие от (35), речь идет о попадании эмпирического среднего $E_n(f)$, совпадающего в рассматриваемом частном случае со средним арифметическим результатов наблюдений, не непосредственно внутрь теоретического среднего $E(x, f)$, а в некоторую *окрестность* теоретического среднего.

Обобщим эту формулировку. Как задать окрестность теоретического среднего в пространстве произвольной природы? Естественно взять его окрестность, определенную с помощью какой-либо метрики. Однако полезно обеспечить на ее дополне-

нии до X *отделенность* множества значений $Mf(x(\omega), y)$ как функции y от минимума этой функции на всем X .

Поэтому мы сочли целесообразным определить такую окрестность с помощью самой функции $Mf(x(\omega), y)$.

Определение 3. Для любого $\varepsilon > 0$ назовем ε -пяткой функции $g(x)$ множество

$$(38) K_\varepsilon(g) = \{x : g(x) < \inf\{g(y), y \in X\} + \varepsilon, x \in X\}.$$

Таким образом, в ε -пятку входят все те x , для которых значение $g(x)$ либо минимально, либо отличается от минимального (точнее, от инфимума – точной нижней грани) не более чем на ε . Так, для $X = R^1$ и функции $g(x) = x^2$ минимум равен 0, а ε -пятка имеет вид интервала $(-\varepsilon^{0,5}; \varepsilon^{0,5})$. В формулировке (37) классического закона больших чисел утверждается, что при любом $\varepsilon > 0$ вероятность попадания среднего арифметического в ε^2 -пятку математического ожидания стремится к 1. Поскольку $\varepsilon > 0$ произвольно, то вместо $\varepsilon^{0,5}$ -пятки можно говорить о ε -пятке, т.е. перейти от (37) к эквивалентной записи

$$(39) \lim_{n \rightarrow \infty} P\{\bar{x} \in K_\varepsilon(M[x_1(\omega) - x]^2)\} = 1.$$

Соотношение (39) допускает непосредственное обобщение на общий случай пространств произвольной природы.

СХЕМА ЗАКОНА БОЛЬШИХ ЧИСЕЛ.

Пусть x, x_1, x_2, \dots, x_n – независимые одинаково распределенные случайные элементы со значениями в пространстве произвольной природы X с показателем различия $f: X^2 \rightarrow R^1$. Пусть выполнены некоторые математические условия регулярности (они формулируются в конкретных теоремах). Тогда для любого $\varepsilon > 0$ справедливо предельное соотношение

$$(40) \lim_{n \rightarrow \infty} P\{E_n(f) \subseteq K_\varepsilon(E(x, f))\} = 1.$$

Аналогичным образом может быть сформулирована и общая идея усиленного закона больших чисел. Ниже приведены две конкретные формулировки «условий регулярности».

7. Теоремы о законах больших чисел

Законы больших чисел. Начнем с рассмотрения естественного обобщения конечного множества – бикompактного пространства X .

Теорема 3. В условиях теоремы 1 справедлив закон больших чисел – соотношение (40).

Доказательство. Воспользуемся построенным при доказательстве теоремы 1 конечным открытым покрытием $\{Z_1, Z_2, \dots, Z_k\}$ пространства X таким, что для него выполнено соотношение (11). Построим на его основе разбиение X на непересекающиеся множества W_1, W_2, \dots, W_m (объединение элементов разбиения W_1, W_2, \dots, W_m составляет X). Это можно сделать итеративно. На первом шаге из Z_1 следует вычесть Z_2, \dots, Z_k – это и будет W_1 . Затем в качестве нового пространства надо рассмотреть разность X и W_1 , а покрытием его будет $\{Z_2, \dots, Z_k\}$. И так до k -го шага, когда последнее из рассмотренных покрытий будет состоять из единственного открытого множества Z_k . Остается из построенной последовательности W_1, W_2, \dots, W_k вычеркнуть пустые множества, которые могли быть получены при осуществлении описанной процедуры (поэтому, вообще говоря, m может быть меньше k).

В каждом из элементов разбиения W_1, W_2, \dots, W_m выберем по одной точке, которые назовем центрами разбиения и соответственно обозначим w_1, w_2, \dots, w_m . Это и есть то конечное множество, которым можно аппроксимировать бикompактное пространство X . Пусть y входит в W_j . Тогда из соотношения (11) вытекает, что

$$(41) \left| \frac{1}{n} \sum_{i=1}^n f(x_i, y) - \frac{1}{n} \sum_{i=1}^n f(x_i, w_j) \right| < \varepsilon.$$

Перейдем к доказательству соотношения (40). Возьмем произвольное $\delta > 0$. Рассмотрим некоторую точку b из $E(x, f)$. Доказательство будет основано на том, что с вероятностью, стремящейся к 1, для любого y вне $K_\delta(E(x, f))$ выполнено неравенство

$$(42) \quad \frac{1}{n} \sum_{i=1}^n f(x_i, y) > \frac{1}{n} \sum_{i=1}^n f(x_i, b).$$

Для обоснования этого неравенства рассмотрим все элементы разбиения W_1, W_2, \dots, W_m , имеющие непустое пересечение с внешностью δ -пятки $K_\delta(E(x, f))$. Из неравенства (41) следует, что для любого y вне $K_\delta(E(x, f))$ левая часть неравенства (42) не меньше

$$(43) \quad \min_j \left(\frac{1}{n} \sum_{i=1}^n f(x_i, w_j) \right) - \varepsilon,$$

где минимум берется по центрам всех элементов разбиения, имеющим непустое пересечение с внешностью δ -пятки. Возьмем теперь в каждом таком разбиении точку v_i , лежащую вне δ -пятки $K_\delta(E(x, f))$. Тогда из неравенств (11) и (43) следует, что левая часть неравенства (42) не меньше

$$(44) \quad \min_j \left(\frac{1}{n} \sum_{i=1}^n f(x_i, v_j) \right) - 2\varepsilon.$$

В силу закона больших чисел для действительных случайных величин (т.е. обычного закона больших чисел теории вероятностей) каждая из участвующих в соотношениях (42) и (44) средних арифметических имеет своими пределами соответствующие математические ожидания, причем в соотношении (42) эти пределы не менее $Mf(x_1, b) + \delta - 2\varepsilon$, поскольку точки v_i лежат вне δ -пятки $K_\delta(E(x, f))$. Следовательно, при $\delta - 2\varepsilon > 0$ и достаточно большом n , обеспечивающем необходимую близость рассматриваемого конечного числа средних арифметических к их математическим ожиданиям, справедливо неравенство (42).

Из неравенства (42) следует, что пересечение $E_n(f)$ с внешностью множества $K_\delta(E(x, f))$ пусто. При этом точка b может входить в $E_n(f)$, а может и не входить. Во втором случае $E_n(f)$ состоит из иных точек, входящих в $K_\delta(E(x, f))$. Теорема 3 доказана.

Если X не является бикompактным пространством, необходимо суметь оценить рассматриваемые суммы «на периферии», вне бикompактного ядра, которое обычно выделяется естествен-

ным путем. Один из возможных комплексов условий сформулирован выше в теореме 2.

Теорема 4. В условиях теоремы 2 справедлив закон больших чисел, т.е. соотношение (40).

Доказательство. Будем использовать обозначения, введенные в теореме 2 и при ее доказательстве. Пусть r и R , $r < R$, – положительные числа. Рассмотрим точку x в шаре $K(r)$ и точку y вне шара $K(R)$. Поскольку $f(x_0, y) \leq D\{f(x_0, x) + f(x, y)\}$, то

$$(45) \quad f(x, y) \geq \frac{1}{D} f(x_0, y) - f(x_0, x) \geq \frac{R}{D} - r.$$

Положим

$$(46) \quad g_n(x) = g_n(x, \omega) = \frac{1}{n} \sum_{i=1}^n f(x_i, x).$$

Сравним $g_n(x_0)$ и $g_n(y)$. Выборку x_1, x_2, \dots, x_n разобьем на две части. В первую часть включим те элементы выборки, которые входят в $K(r)$, во вторую – все остальные (т.е. лежащие вне $K(r)$). Множество индексов элементов первой части обозначим $I = I(n, r)$. Тогда в силу неотрицательности f имеем

$$(47) \quad g_n(y) \geq \frac{1}{n} \sum_{i \in I} f(x_i, y),$$

а в силу неравенства (45)

$$(48) \quad \sum_{i \in I} f(x_i, y) \geq \left(\frac{R}{D} - r \right) |I(n, r)|,$$

где $|I(n, r)|$ – число элементов в множестве индексов $I(n, r)$. Следовательно, на основе (47) и (48) заключаем, что для определенной формулой (46) функции $g_n(y)$ справедливо неравенство

$$(49) \quad g_n(y) \geq \frac{1}{n} \left(\frac{R}{D} - r \right) J,$$

где $J = |I(n, r)|$ – биномиальная случайная величина $B(n, p)$ с вероятностью успеха $p = P\{x_i(\omega) \in K(r)\}$. По теореме Хинчина для $g_n(x_0)$ справедлив (классический) закон больших чисел. Пусть $\varepsilon > 0$. Выберем $n_1 = n_1(\varepsilon)$ так, чтобы при $n > n_1$ было выполнено соотношение

$$(50) P\{g_n(x_0) - g(x_0) > \varepsilon\} < \varepsilon,$$

где $g(x_0) = Mf(x_1, x_0)$. Выберем r так, чтобы вероятность успеха $p > 0,6$. По теореме Бернулли можно выбрать натуральное число n_2 (как функцию ε) так, чтобы при $n > n_2$ было выполнено неравенство

$$(51) P\{J > 0,5n\} > 1 - \varepsilon.$$

Выберем R так, чтобы

$$(52) \frac{1}{2} \left(\frac{R}{D} - r \right) > g(x_0) + \varepsilon.$$

Тогда

$$(53) K_\varepsilon(g) \subseteq K(R)$$

и согласно (49)–(52) при $n > n_3 = \max(n_1, n_2)$ с вероятностью не менее $1 - \varepsilon$ выполнено неравенство $g_n(y) > g_n(x_0)$ для любого y вне $K(R)$. Из (53) следует, что минимизировать g_n достаточно внутри бикompактного шара $K(R)$, при этом $E_n(f)$ не пусто и

$$(54) P\{E_n(f) \subseteq K(R)\} \geq 1 - 2\varepsilon.$$

Пусть G_n и G – сужения g_n и $g(x) = Mf(x(\omega), x)$, соответственно, на $K(R)$ как функций от x . В силу (53) справедливо равенство

$$(55) K_\varepsilon(G) \subseteq K(R)$$

Согласно доказанной выше теореме 3 найдется $n_4 = n_4(\omega)$ такое, что при $n > n_4$

$$(56) P\{K_0(G_n) \subseteq K_\varepsilon(g)\} > 1 - \varepsilon.$$

Согласно (54) $K_0(G_n) = E_n(f)$ при $n > n_3$ с вероятностью не менее $1 - 2\varepsilon$. Следовательно, при $n > n_5(\varepsilon) = \max(n_3, n_4)$ имеем

$$(57) P\{E_n(f) \subseteq K_\varepsilon(g)\} > 1 - 3\varepsilon,$$

что и завершает доказательство теоремы 4.

Справедливы и иные варианты законов больших чисел, полученные, в частности, в статье [14]. Первое доказательство закона больших чисел (в весьма частном случае) было получено еще Дж. Кемени [27].

8. Изучение свойств медианы Кемени при конечных объемах выборок

Рассмотрим на основе развитой выше теории частный случай пространств нечисловой природы – пространства бинарных отношений на конечном множестве и медианы Кемени как способ нахождения эмпирического среднего. Поскольку число бинарных отношений на конечном множестве конечно, то эмпирические и теоретические средние для произвольных показателей различия существуют и справедливы доказанные выше законы больших чисел.

Бинарные отношения, в частности, упорядочения, часто используются для описания мнений экспертов. Тогда расстояние Кемени измеряет близость мнений экспертов, а медиана Кемени позволяет находить итоговое усредненное мнение комиссии экспертов. Расчет медианы Кемени обычно включают в информационное обеспечение систем принятия решений с использованием оценок экспертов. Речь идет, например, о математическом обеспечении *автоматизированного рабочего места* «Математика в экспертизе» (АРМ «МАТЭК») [20], предназначенного, в частности, для использования при проведении экспертиз в задачах экологического страхования. Поэтому представляет большой практический интерес численное изучение свойств медианы Кемени при конечном объеме выборки. Такое изучение дополняет описанную выше асимптотическую теорию, в которой объем выборки предполагается безгранично возрастающим ($n \rightarrow \infty$).

С помощью специально разработанной программной системы В.Н. Жихарев вместе с нами провел ряд серий численных экспериментов по изучению свойств выборочных медиан Кемени (в пространстве ранжировок без связей). Представление о полученных результатах дается таблице 1, взятой из статьи [4]. В каждой серии методом статистических испытаний определенное число раз моделировался случайный и независимый выбор экспертных ранжировок, а затем находились все медианы Кемени для смоделированного набора мнений экспертов. При этом в

сериях 1–5 распределение ответа эксперта предполагалось равномерным на множестве всех ранжировок. В серии 6 это распределение являлось монотонным относительно расстояния Кемени с некоторым центром (о понятии монотонности см. [17, гл.1]), т.е. вероятность выбора определенной ранжировки убывала с увеличением расстояния Кемени этой ранжировки от центра. Таким образом, серии 1–5 соответствуют ситуации, когда у экспертов нет почвы для согласия, нет группировки их мнений относительно некоторого единого среднего группового мнения, в то время как в серии 6 есть единое мнение – описанный выше центр, к которому тяготеют ответы экспертов.

Таблица 1. Результаты вычислительного эксперимента по изучению медианы Кемени

Номер серии	1	2	3	4	5	6
Число испытаний	100	1000	50	50	1000	1000
Количество объектов	5	5	7	7	5	5
Количество экспертов	10	30	10	30	10	10
Частота непустого пересечения	0,85	0,58	0,52	0,2	0,786	0,911
Среднее отношение диаметров	0,283	0,124	0,191	0,089	0,202	0,044
Средняя мощность медианы	5,04	2,41	6,4	2,88	3,51	1,35
Максимальная мощность медианы	30	14	19	11	40	12

Результаты, приведенные в таблице 1, можно комментировать разными способами. Неожиданным явилось большое число элементов в выборочной медиане Кемени – как среднее, так и особенно максимальное. Одновременно обращает на себя внимание убывание этих чисел при росте числа экспертов и особенно при переходе к ситуации реального существования группового мнения (серия 6). Достаточно часто один из ответов экспертов входит в медиану Кемени (т.е. пересечение множества ответов экспертов и медианы Кемени непусто), а диаметр медианы как множества в пространстве ранжировок заметно меньше диамет-

ра множества ответов экспертов. По этим показателям – наилучшее положение в серии 6. Грубо говоря, всяческие «патологии» в поведении медианы Кемени наиболее резко проявляются в ситуации, когда ее применение не имеет содержательного обоснования, т.е. когда у экспертов нет основы для согласия, их ответы равномерно распределены на множестве ранжировок.

Увеличение числа испытаний в 10 раз при переходе от серии 1 к серии 5 не очень сильно повлияло на приведенные в таблице характеристики, поэтому представляется, что суть дела выявляется при числе испытаний (в методе Монте-Карло), равном 100 или даже 50. Увеличение числа объектов или экспертов увеличивает число элементов в рассматриваемых пространствах ранжировок, а потому уменьшается частота попадания какого-либо из мнений экспертов внутрь медианы Кемени. А также отношение диаметра медианы к диаметру множества экспертов и число элементов медианы Кемени (среднее и максимальное). Можно сказать так: увеличение числа объектов или экспертов уменьшает степень дискретности задачи, приближает ее к непрерывному случаю, а потому уменьшает выраженность различных «патологий».

Есть много интересных направлений исследований, которые здесь не рассматриваем. Они связаны со сравнением медианы Кемени с другими методами усреднения мнений экспертов, например, с нахождением итогового упорядочения по методу средних рангов [20]. А также с использованием малых окрестностей ответов экспертов для поиска входящих в медиану ранжировок (с целью сокращения расчетов). Или с построением теоретических и численных оценок скорости сходимости в законах больших чисел.

Центральная роль в нечисловой статистике эмпирических и теоретических средних, законов больших чисел и их обобщений, предназначенных для изучения решений экстремальных статистических задач, продемонстрирована в [17, 20]. Современная теория экспертных оценок является прикладным «зеркалом» нечисловой статистики. Значение постановок настоящей статьи

для разработки и применения экспертных технологий обсуждается в [18, 20].

Литература

1. *Вероятность и математическая статистика: Энциклопедия* / Гл. ред. акад. РАН Ю.В. Прохоров. – М.: Большая Российская Энциклопедия, 1999. – 910 с.
2. ГОРСКИЙ В.Г., ГРИЦЕНКО А.А., ОРЛОВ А.И. *Метод согласования кластеризованных ранжировок* // Автоматика и телемеханика. – 2000. – №3. – С. 179–187.
3. ДЖИНИ К. *Средние величины* / Пер. с итал. – Москва: Статистика, 1970. – 448 с.
4. ЖИХАРЕВ В.Н., ОРЛОВ А.И. *Законы больших чисел и состоятельность статистических оценок в пространствах произвольной природы* // Статистические методы оценивания и проверки гипотез: межвуз. сб. науч. тр. – Пермь: Изд-во Пермского госуд. ун-та, 1998. – С. 65–84.
5. КЕЛЛИ Дж. *Общая топология*. – М.: Наука, 1968. – 384 с.
6. КЕМЕНИ Дж., ШЕЛЛ Дж. *Кибернетическое моделирование: Некоторые приложения*. – М.: Советское радио, 1972. – 192 с.
7. *Космонавтика XXI века. Попытка прогноза до 2101 года* // Под ред. акад. РАН Б.Е. Чертока. – М.: РТСофт, 2010. – 864 с.
8. ЛИТВАК Б.Г. *Экспертная информация: методы получения и анализа*. – М.: Радио и связь, 1982. – 184 с.
9. МИРКИН Б.Г. *Проблема группового выбора*. – М.: Наука, 1974. – 256 с.
10. МИРКИН Б.Г., ЧЕРНЫЙ Л.Б. *Об измерении близости между разбиениями конечного множества объектов* // Автоматика и телемеханика. – 1970. – №5. – С. 120–127.
11. МИРКИН Б.Г., ЧЕРНЫЙ Л.Б. *Некоторые свойства пространства разбиений* // Математический анализ экономических моделей: Ч. III. – Новосибирск: ИЭиОПП СО АН СССР, 1972. – С. 126–147.

12. НОВИКОВ Д.А., ОРЛОВ А.И. *Экспертные оценки – инструменты аналитика* // Заводская лаборатория. – 2013. – Т. 79, №4. – С. 3–4.
13. ОРЛОВ А.И. *Устойчивость в социально-экономических моделях*. – М.: Наука, 1979. – 296 с.
14. ОРЛОВ А.И. *Асимптотика решений экстремальных статистических задач* // Анализ нечисловых данных в системных исследованиях. Сборник трудов. Вып.10. – М.: Всесоюзный научно-исследовательский институт системных исследований, 1982. – С. 4–12.
15. ОРЛОВ А.И. *Прикладная статистика*. – М.: Экзамен, 2006. – 671 с.
16. ОРЛОВ А.И. *Теория принятия решений*. – М.: Экзамен, 2006. – 574 с.
17. ОРЛОВ А.И. *Организационно-экономическое моделирование: учебник: в 3 ч. Ч.1. Нечисловая статистика*. – М.: Изд-во МГТУ им. Н.Э. Баумана, 2009. – 541 с.
18. ОРЛОВ А.И. *О развитии экспертных технологий в нашей стране* // Заводская лаборатория. – 2010. – Т. 76, №11. – С. 64–70.
19. ОРЛОВ А.И. *Организационно-экономическое моделирование: теория принятия решений* : учеб. для вузов. – М. : КноРус, 2011. – 568 с.
20. ОРЛОВ А.И. *Организационно-экономическое моделирование: учебник: в 3 ч. Ч.2. Экспертные оценки*. – М.: Изд-во МГТУ им. Н.Э. Баумана, 2011. – 486 с.
21. ОРЛОВ А.И. *Роль медиан Кемени в экспертных оценках и статистическом анализе данных* // Теория активных систем: Труды международной научно-практической конференции (14–16 ноября 2011 г., Москва, Россия). Том I. / Под ред. В.Н. Буркова, Д.А. Новикова. – М.: ИПУ РАН, 2011. – С. 172–176.
22. РАУШЕНБАХ Г.В. *Меры близости и сходства в социологии* // Анализ нечисловой информации в социологических исследованиях. – М.: Наука, 1985. – С. 169–203.

23. ХОЛЛЕНДЕР М., ВУЛФ Д. *Непараметрические методы статистики*. – М.: Финансы и статистика, 1983. – 518 с.
24. ШРЕЙДЕР Ю.А. *Равенство, сходство, порядок*. – М.: Наука, 1971. – 256 с.
25. JACKSON B.N., SCHNABLE P.S., ALURU S. *Consensus Genetic Maps as Median Orders from Inconsistent Sources* // IEEE/ACM Transactions on computational biology and bioinformatics. – 2008. – Vol. 5, №2. – P. 161–171.
26. KEMENY J. *Mathematics without Numbers* // Daedalus. – 1959. – Vol. 88. – P. 577–591.
27. KEMENY J. *Generalized random variables* // Pacific Journal of Mathematics. – 1959. – Vol. 9. – P. 1179–1189.
28. KLAMLER Ch. *Kemeny's rule and Slater's rule: A binary comparison* // Economics Bulletin. – 2003. – Vol. 4, №35. – P. 1–7.
29. MERLIN V., SAARI D.G. *A geometric examination of Kemeny's rule* // Social Choice and Welfare. – 2000. – Vol. 17, № 3. – P. 403–438.

ON MEAN VALUES

Alexander Orlov, Bauman Moscow State Technical University, Moscow Institute of Physics and Technology (State University), Moscow, DSc (economics), DSc (technics), PhD (mathematics), professor (prof-orlov@mail.ru).

Abstract: New results in the theory of means (mean values) are given. We introduce the weighted means of type I corresponding to the sample, and those of type II corresponding to the set of order statistics. Evolution of ideas on Kemeny distance and Kemeny median is traced. We also propose a computation-friendly variation of Kemeny median, which avoids the “center of the bagel hole” effect. Then we generalize the Kemeny median by virtue of empirical and theoretical means in abstract spaces and prove the laws of large numbers for them.

Keywords: weighted means, Kemeny distance, Kemeny median, empirical means, theoretical means, laws of large numbers.

*Статья представлена к публикации
членом редакционной коллегии Д.А. Новиковым*

*Поступила в редакцию 08.05.2013.
Опубликована 30.11.2013.*

УДК 004.8
ББК 32.813

ИНТЕЛЛЕКТУАЛЬНЫЕ СИСТЕМЫ НА ОСНОВЕ КОЛОНОК

Чесноков А. М.¹

*(ФГБУН Институт проблем управления
им. В.А.Трапезникова РАН, Москва)*

В статье рассматриваются интеллектуальные системы на основе колонок. Приводятся основные понятия и определения. Формулируются прямая и обратная задачи. Приводятся решения этих задач и оценки вычислительной эффективности. На этой основе предлагается решение задачи классификации и доказывается возможность реализации произвольных булевых функций. В конце дается оценка классов задач, которые могут решаться с помощью интеллектуальных систем на основе колонок.

Ключевые слова: искусственный интеллект, интеллектуальные системы на основе колонок, колонка.

1. Введение

Начало работ над рассматриваемой моделью интеллектуальных систем связано с публикациями [4] и [5], которые, с одной стороны, помогли составить примерные представления об общем облике систем на основе колонок, с другой, позволили выделить элементарные задачи, без решения которых невозможно функционирование таких систем, а также помогли остановить свой выбор на простом математическом объекте, используемом в качестве колонки. Вообще, простота, высокий уровень общности и невведение новых понятий и ограничений до тех

¹ Александр Михайлович Чесноков, старший научный сотрудник, кандидат технических наук (alex-ches@yandex.ru).

пор, пока в этом не возникает настоятельная необходимость, – это те положения, которые в значительной мере определяли процесс построения модели. Полученная в результате модель систем на основе колонок чрезвычайно проста.

Имеется пусть и очень большое, но конечное множество имен, которые используются для наименования объектов произвольной природы. В этом множестве выделяются непересекающиеся подмножества, получившие название областей имен. В реальных задачах выделение областей может быть связано, например, с типизацией или с выделением «миров» (domains), как в некоторых реализациях Пролога. Одной из важнейших причин выделения областей является требование обеспечить отсутствие случайных совпадений имен в различных частях большой системы и тем самым исключить возможность возникновения соответствующих ошибок. Для рассматриваемой модели не важно, какие причины могут приводить к выделению областей. Главное, эти причины существуют и это нашло отражение в модели.

Под образом понимается любое конечное множество имен, принадлежащих тем или иным областям имен. Образы, входящие в любое множество образов, могут быть наименованы при помощи имен некоторой области, т.е. каждому образу из множества может быть сопоставлено его имя. Упорядоченная пара (имя образа, образ) получила название колонки.

Система на основе колонок представляет собой одно или несколько множеств колонок и работающий с ними механизм (машина колонок), который, получая информацию о внешнем мире в виде образов, формирует новые колонки, изменяет или удаляет уже существующие и выполняет другие необходимые операции. В основе процесса накопления знаний в подобных системах лежит запоминание новых образов под определенными именами. При этом элементарными базовыми задачами, без которых невозможно функционирование системы, являются прямая задача – по образу получить его имя, и обратная – по имени получить соответствующий образ.

Для данной модели систем на основе колонок сразу возникают три основных вопроса:

1. Что из себя представляют множества колонок, с которыми придется иметь дело системе?
2. Существуют ли методы решения базовых задач?
3. Какие вообще задачи могут решать такие простые системы?

Ответу на эти вопросы и посвящена данная работа.

В разделе 2 приводятся основные понятия и определения, относящиеся к системам на основе колонок: имя, образ, колонка, индекс как множество колонок. Рассматриваются первичные колонки и колонки с классами. Показана связь между индексом как множеством колонок и обычным индексом, заданным в виде таблицы ссылок. В разделе 3 сначала показывается существование решения базовых задач для любого типа образов. Затем для образов в виде конечных неупорядоченных множеств и конечных последовательностей или векторов предлагается более эффективный метод решения. Наконец в разделе 4 показано как базовые задачи используются для решения более сложных задач. Сначала рассматривается задача классификации, затем доказывается возможность реализации в системах на основе колонок произвольных булевых функций. На основе этих результатов дается оценка тех классов задач, для которых могут использоваться интеллектуальные системы на основе колонок.

2. Основные понятия

2.1. ИМЕНА, ОБРАЗЫ, КОЛОНКИ

Рассматривается конечное множество U – множество имен некоторых объектов произвольной природы. Не ограничивая общности, будем считать, что оно является подмножеством множества целых чисел Z . Множество U конечное, однако никаких ограничений на его размеры не накладывается. При необходимости его всегда можно увеличить, добавив необходимое число элементов.

В множестве имен U выделяются непересекающиеся подмножества имен, получившие название *областей имен*. При необходимости любая область имен может быть в любой момент расширена, а также может быть введена новая область имен.

Любое конечное множество имен, принадлежащих тем или иным областям имен, называется *образом*.

Рассмотрим некоторое конечное множество образов P . Его можно перенумеровать, используя для этого элементы некоторой области имен U' , $|U'| = |P|$:

$$P = \{p_i \mid i \in U'\},$$

где $|\cdot|$ – мощность множества. То, что образы множества P перенумерованы, означает, что установлено взаимно однозначное соответствие $\varphi: i \leftrightarrow p_i$ между областью имен U' и множеством образов P .

Колонкой называется упорядоченная пара (i, p_i) , где i – имя колонки, $p_i = \varphi(i)$ – образ, содержащийся в колонке. Колонка обозначается как $(i \mid p_i)$. Также будет использоваться обозначение $i \rightarrow p_i$. В этом случае будем говорить, что имя колонки i является *ссылкой* или *указателем* на содержащийся в колонке образ p_i . В свою очередь, часто будет говориться, что образ p_i имеет имя i или известен под именем i . Под этим будет подразумеваться то, что образ p_i содержится в колонке $(i \mid p_i)$. Отображение $\varphi: i \rightarrow p_i$ будет называться *отображением наименования*.

2.2. ПЕРВИЧНЫЕ КОЛОНКИ

В образы колонок могут входить имена других колонок, т.е. имя в образе одной колонки может быть именем другой колонки и служить указателем на соответствующий образ. В образ также могут входить имена, которые еще не использовались для наименования ни одного образа. Такие не связанные ни с одним образом имена будут называться *чистыми* или *пустыми именами*. Положив, что чистое имя ссылается на пустое множество, его можно рассматривать как колонку вида $(i \mid \emptyset)$ или $i \rightarrow \emptyset$. Таким образом, можно считать, что в образе одной колонки содержатся имена других колонок, каждое из которых служит указателем на соответствующий

образ, возможно, пустой. В результате образуется показанная на рис. 1 сложная структура колонок (для наглядности на рисунке дублируются имена колонок).

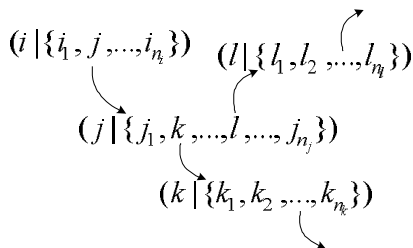


Рис. 1. Структура из колонок

Рассматриваемые простейшие колонки называются *первичными колонками*. Множество первичных колонок будет обозначаться как C^1 .

2.3. ФАКТОРИЗАЦИЯ КОЛОНОК, КОЛОНКИ С КЛАССАМИ

Отличительной особенностью первичных колонок C^1 является взаимно однозначное соответствие между единственным именем и единственным образом. Однако реально может возникнуть необходимость в том, чтобы образ имел несколько имен. Следовательно, должны иметься несколько колонок с разными именами и одним и тем же образом. Если рассмотреть отображение наименования $\varphi: i \rightarrow p$, которое имени ставит в соответствие образ, то все эти имена образуют класс эквивалентности, состоящий из имен с одним и тем же образом, т.е. имеет место *факторизация колонок по образу*. Взаимно однозначное соответствие между именем колонки и образом сохраняется, но как взаимно однозначное соответствие класса имен, указывающих на один и тот же образ: $[i] \leftrightarrow p$, где $[i]$ – класс эквивалентных имен. Отображение наименования $\varphi: [i] \rightarrow p$ определяется как $\varphi([i]) = \varphi(i_k)$ для $\forall i_k \in [i]$. Таким образом, кроме колонок C^1 необходимо также рассматривать и колонки вида $([i] | p_i)$:

$$\begin{array}{ccc} p & \dots & p \\ \uparrow & \dots & \uparrow \\ i_1 & \dots & i_m \end{array} \Rightarrow \begin{array}{c} p \\ \uparrow \\ [i] = \{i_1, \dots, i_m\} \end{array}$$

Пусть теперь имеет место противоположный случай, когда несколько образов имеют одно и то же имя. Теперь класс эквивалентности возникает по отображению $\varphi^{-1}: p \rightarrow i$, которое образу ставит в соответствие его имя. В класс входят все образы с одним и тем же именем, т.е. имеет место *факторизация колонок по имени*. Так же, как и в предыдущем случае, взаимно однозначное соответствие сохраняется. Однако на этот раз в виде взаимно однозначного соответствия класса образов, имеющих одно и то же имя: $i \leftrightarrow [p]$, где $[p]$ – класс эквивалентных образов. Отображение $\varphi^{-1}: [p] \rightarrow i$ определяется как $\varphi^{-1}([p]) = \varphi^{-1}(p_k)$ для $\forall p_k \in [p]$. Таким образом, кроме колонок C^1 также должны рассматриваться и колонки вида $(i | [p])$:

$$\begin{array}{ccc} p_1 & \dots & p_m \\ \uparrow & \dots & \uparrow \\ i & \dots & i \end{array} \Rightarrow \begin{array}{c} [p] = \{p_1, \dots, p_m\} \\ \uparrow \\ i \end{array}$$

Объединив два рассмотренных выше случая факторизации, получим колонки вида $([i] | [p_i])$. Множество таких колонок будет называться *множеством колонок с классами* и обозначаться через C^2 . Колонки множества C^2 – это колонки, имена которых могут представлять классы эквивалентных имен, а состоящие из них образы – классы эквивалентных образов. Очевидно, $C^1 \subset C^2$.

Колонки множества C^2 можно привести к виду, когда образы колонок не являются классами эквивалентности, т.е. когда классы эквивалентности встречаются только среди имен.

Пусть имеется некоторая колонка $(i | [p]) \in C^2$, где образ колонки – это класс $[p] = \{p_1, \dots, p_m\}$, $m > 1$. Используем m чистых имен i_k для того, чтобы наименовать все образы p_k , входящие в класс эквивалентности $[p]$. Получим m колонок $(i_k | p_k)$. Сформируем образ $\{i_1, \dots, i_m\}$, состоящий из имен этих колонок. Тогда исходную колонку $(i | [p])$ можно заменить $m + 1$ колонкой $(i | \{i_1, \dots, i_m\})$, $(i_k | p_k)$, $k = 1, \dots, m$:

$$i \rightarrow [p] = i \rightarrow \{p_1, \dots, p_m\} \Rightarrow i \rightarrow \{i_1, \dots, i_m\}$$

$$\begin{array}{ccc} & p_1 & p_m \\ & \uparrow & \uparrow \\ & \dots & \dots \end{array}$$

На месте колонки $(i | [p])$ появилась первичная колонка $(i | \{i_1, \dots, i_m\})$ и m колонок $(i_k | p_k)$. Действуя аналогичным образом, можно привести конечное множество колонок S^2 к виду, когда классы эквивалентности будут встречаться только среди имен.

2.4. ИНДЕКС

Индексом называется любое конечное множество колонок. Если это специально не оговаривается, то считается, что индекс – неупорядоченное множество колонок

Состав любого индекса может меняться за счет добавления или удаления колонок. Одна колонка – это также индекс, следовательно, множество колонок S^2 входит в множество индексов. Поэтому добавление и удаление колонок индекса – это операции на множестве индексов. Эти операции будут называться сложением и вычитанием индексов и обозначаться через $+$ и $-$.

Как уже говорилось, любая колонка – это индекс. Интересно, что справедливо и обратное – индекс можно представить как колонку. Действительно, как и любое конечное множество колонок, индекс можно наименовать, используя некоторое чистое имя i для образа, который состоит из имен входящих в индекс колонок. Имя этой колонки-индекса можно интерпретировать как имя индекса.

$$\begin{array}{ccc} p_k & p_l & p_m \\ \uparrow & \dots & \uparrow & \dots & \uparrow \\ k, \dots, l, \dots, m \end{array} \Rightarrow i \rightarrow \{k, \dots, l, \dots, m\}$$

$$\begin{array}{ccc} & p_k & p_l & p_m \\ & \uparrow & \uparrow & \uparrow \\ & \dots & \dots & \dots \end{array}$$

Индекс как безымянное множество колонок Индекс как колонка – наименованное множество колонок

Из этих колонок-индексов в свою очередь могут образовываться индексы и т.д. Можно выделить *уровни индекса по построению*: L^1 – индексы из колонок, L^2 – индексы из колонок-

индексов уровня L^1 , L^3 – индексы из колонок-индексов уровня L^2 и т.д. Принадлежность колонки-индекса к одному из уровней может интерпретироваться как положение индекса в иерархии индексов (рис. 2).

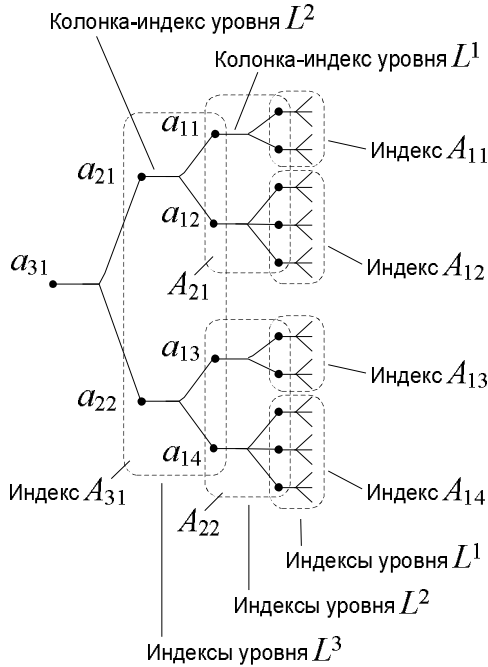


Рис. 2. Уровни индекса по построению

В качестве примера рассмотрим конечное множество $A = \{A_1, \dots, A_n\}$, где A_k – индексы уровня L^1 . Легко видеть, что A – это индекс уровня L^2 . Для этого достаточно воспринимать наши обозначения A_k как имена соответствующих колонок-индексов. Можно явным образом наименовать индексы A_k , используя для этого чистые имена a_k , $k = 1, \dots, n$. В результате будем иметь индекс A уровня L^2 , состоящий из колонок-индексов $a_k \rightarrow A_k$, где ссылка указывает на образ $\{i_1^k, \dots, i_{m_k}^k\}$, состоящий из имен колонок индекса A_k :

$$\begin{array}{ccc}
 A_1 & A_n & p_1^k, \dots, p_{m_k}^k \\
 \uparrow & \dots & \uparrow \quad \dots \quad \uparrow \\
 A = \{a_1, \dots, a_n\}, & a_k \rightarrow A_k = a_k \rightarrow \underbrace{\{i_1^k, \dots, i_{m_k}^k\}}_{A_k}
 \end{array}$$

Любой из уровней индекса является множеством колонок и, следовательно, может быть наименован, т.е. представлен в виде колонки. Образ колонки-уровня L^k состоит из имен всех колонок-индексов, образующих уровень L^k . Имя колонки может интерпретироваться как имя уровня L^k .

Вообще, какие бы причины ни приводили к формированию тех или иных множеств колонок, не будет получено ничего кроме колонок. Это одна из основных особенностей модели, основанной на колонках.

2.5. ПРЕДСТАВЛЕНИЕ ИНДЕКСА В ВИДЕ ТАБЛИЦЫ

Индекс A может быть представлен в виде таблицы, состоящей из вертикальных колонок (столбцов) переменной высоты. В нижней строке таблицы, под чертой, имена колонок. Над именем каждой колонки перечислены все имена, входящие в образ колонки. По умолчанию считается, что в таблице имена колонок и имена в образах принадлежат различным областям имен.

Для индекса в виде неупорядоченного множества колонок порядок записи колонок в таблице может быть произвольным. Если образы представляют собой неупорядоченные множества имен, то и порядок записи имен в образах колонок также может быть произвольным.

Если образы в колонках индекса представляют собой упорядоченные множества, то запись имен в образах колонок выполняется в определенном порядке, например, снизу вверх, т.е. первое имя образа в первой строке над чертой, второе – во второй и т.д.

Индекс, состоящий из первичных колонок, может быть сразу записан в виде таблицы. Для более сложных колонок могут потребоваться дополнительные преобразования.

Пусть A – представленный в виде таблицы индекс с неупорядоченными образами, не содержащий пустых колонок. Тогда для него можно построить *обратный индекс*, который будет обозначаться как A^{-1} . Для этого берется множество U_p всех имен, содержащихся в образах всех колонок индекса A . Каждое имя, принадлежащее U_p , используются как имя одной из колонок формируемого обратного индекса A^{-1} . При этом в образ колонки индекса A^{-1} с именем $k \in U_p$ включаются те и только те имена колонок индекса A , которые содержат k в своем образе. Другими словами, если имя $k \in U_p$ содержится только в образах колонок $(i_1 | p_1), \dots, (i_m | p_m) \in A$, то колонка обратного индекса A^{-1} с именем k будет иметь вид $(k | \{i_1, \dots, i_m\})$.

Например, в колонку с именем 4 приведенного ниже обратного индекса A^{-1} включены имена колонок 2 и 3 – имена всех колонок индекса A , которые содержат в образе имя 4.

$$\begin{array}{ccc}
 A & \Rightarrow & A^{-1} & \Rightarrow & (A^{-1})^{-1} \\
 \begin{array}{c} 5 \\ 5 \quad \boxed{4} \\ 3 \quad 3 \quad 5 \\ 1 \quad 2 \quad \boxed{4} \\ \hline 1 \quad 2 \quad 3 \end{array} & & \begin{array}{c} 3 \\ 2 \quad \boxed{3} \quad 2 \\ 1 \quad 2 \quad 1 \quad 2 \quad 1 \\ \hline 1 \quad 2 \quad 3 \quad \boxed{4} \quad 5 \end{array} & & \begin{array}{c} 5 \\ 5 \quad 4 \\ 3 \quad 3 \quad 5 \\ 1 \quad 2 \quad 4 \\ \hline 1 \quad 2 \quad 3 \end{array}
 \end{array}$$

Если для приведенного в примере индекса A^{-1} построить обратный индекс $(A^{-1})^{-1}$, то снова будет получен индекс A . Можно показать [2], что для представленного в виде таблицы индекса A с неупорядоченными образами, не содержащего пустых колонок, существует единственный обратный индекс A^{-1} , причем $(A^{-1})^{-1} = A$.

2.6. ВРЕМЯ, СОБЫТИЯ

С помощью образов могут быть представлены временные процессы. Для этого используются образы в виде конечных последовательностей вида $p = (i_1, \dots, i_k)$, $i_l \in U_l$, $1 \leq k \leq n$, где U_k – область имен k -й координаты, n – максимальная размерность рассматриваемых конечных последовательностей. Для

того чтобы последовательность $p = (i_1, \dots, i_n)$ можно было рассматривать как временной процесс, достаточно сопоставить порядковым номерам координат дискретные моменты времени $k \rightarrow t_k, k = 1, \dots, n$.

Абстрактный образ, который может быть привязан к любому конечному отрезку дискретного времени соответствующей длины, получил название *шкалы времени*. Более точно под шкалой времени $\tau = (t_1, t_2, \dots, t_n)$ понимается образ в виде конечной последовательности с именами из области имен моментов времени T . При этом имена, являющиеся координатами шкалы τ , образуют упорядоченную последовательность, т.е. из $k > i$ следует, что $t_k > t_i$, где $t_k, t_i \in T$. Также возможны шкалы с обратным порядком, где из $k > i$ следует, что $t_k < t_i$. Тот факт, что момент времени t_k является координатой шкалы времени τ , будет обозначаться как $t_k \in \tau$ и про него будет говориться, что это элемент или момент шкалы τ .

Может существовать несколько шкал времени. По умолчанию считается, что элементы различных шкал принадлежат различным областям имен.

Для того чтобы привязать колонку ($i | p$) к моменту времени t шкалы τ , достаточно сформировать колонку ($t | i$). В результате образуется цепочка ссылок $t \rightarrow i \rightarrow p$. При этом факторизация по моменту времени t , т.е. факторизация по имени t , приведет к формированию образа из имен всех колонок, относящихся к этому моменту времени:

$$\begin{array}{ccccccc}
 p_1 & p_2 & & p_m & & p_1 & p_2 & p_m \\
 \uparrow & \uparrow & \dots & \uparrow & & \uparrow & \uparrow & \dots & \uparrow \\
 i_1 & i_2 & & i_m & \Rightarrow & t & \rightarrow & \{i_1, i_2, \dots, i_m\} \\
 \uparrow & \uparrow & \dots & \uparrow & & & & & \\
 t & t & & t & & & & &
 \end{array}$$

Любая колонка вида ($t | p_i$) называется *событием*, а образ p_i — *состоянием* в момент времени $t \in \tau$. На приведенной выше схеме событиями являются колонки ($t | i_k$) и ($t | \{i_1, i_2, \dots, i_m\}$).

2.7. УПОРЯДОЧЕННЫЙ ИНДЕКС

Наличие образов в виде конечных упорядоченных множеств, приводит к появлению упорядоченных множеств колонок, т.е. *упорядоченных индексов* или *индексов с упорядоченным множеством колонок*. Ниже показан безымянный упорядоченный индекс, состоящий из колонок $(i_k | p_k)$:

$$\begin{array}{cccc} p_1 & p_2 & & p_m \\ \uparrow & \uparrow & \dots & \uparrow \\ (i_1, & i_2, & \dots, & i_m) \end{array} = \left(\begin{array}{ccc} p_1 & p_2 & p_m \\ \uparrow & \uparrow & \dots & \uparrow \\ i_1, & i_2, & \dots, & i_m \end{array} \right)$$

Практический интерес представляет упорядоченный индекс, колонки которого представляют собой m последних событий.

Пусть имеется некоторая шкала времени $\tau = (1, 2, \dots, n)$. Рассмотрим упорядоченный индекс A_m , число колонок которого не превосходит m , где $1 < m < n$. Первоначально индекс $A_m = \emptyset$. В момент времени 1 в индекс A_m добавляется колонка $(1 | s_1)$, в момент времени 2 – колонка $(2 | s_2)$ и т.д., где s_k – состояние в момент времени $k \in \tau$. В момент времени m индекс A_m будет состоять из колонок $(1 | s_1), (2 | s_2), \dots, (m | s_m)$, т.е. будет содержать m последних событий. Так как индекс A_m не может содержать более чем m колонок, то в момент времени $m + 1$ из него сначала необходимо удалить самую старую колонку $(1 | s_1)$, и только потом можно будет добавить новую колонку $(m + 1 | s_{m+1})$. Индекс A_m по-прежнему будет содержать m последних событий. Аналогично, в момент времени $m + 2$ будет удалена самая старая колонка $(2 | s_2)$ и добавлена новая $(m + 2 | s_{m+2})$ и т.д. В любой момент времени k , $m \leq k < n$, индекс A_m будет содержать m колонок $(k - m + 1 | s_{k-m+1}), \dots, (k | s_k)$, представляющих собой m последних событий.

3. Системы на основе колонок, прямая и обратная задачи

В общем виде интеллектуальную систему на основе колонок можно представить в виде одного или нескольких конечных

множеств колонок, т.е. индексов, и работающего с ними механизма, который, получая информацию о внешнем мире в виде образов, формирует новые колонки, изменяет уже существующие, удаляет ненужные и выполняет другие необходимые операции (рис. 3).

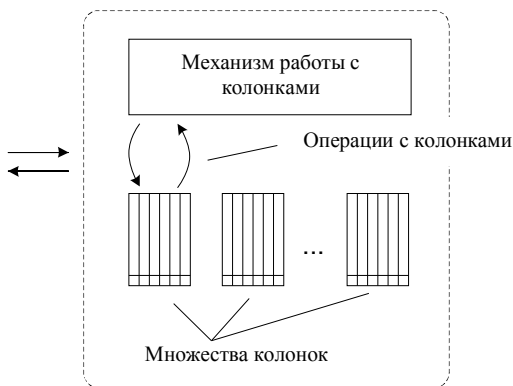


Рис. 3. Система на основе колонок

Знания в рассматриваемых системах представлены с помощью колонок, а в основе процесса накопления знаний лежит запоминание новых образов под определенными именами. Очевидно, базовыми задачами для подобных систем являются *прямая задача* – по образу получить его имя, и *обратная задача* – по имени получить соответствующий образ.

Существует общий метод решения этих задач, применимый к любым типам образов. Этим методом является метод, использующий поэлементное сравнение.

Известные системе образы хранятся в индексе A , который представляет собой множество колонок вида $(i | p_i)$, где p_i – образ, известный под именем i . В исходном состоянии $A = \emptyset$. Любой образ p , для которого надо решить прямую задачу поэлементно сравнивается с образами всех колонок индекса A . Если найден совпадающий образ p_i , то его имя, т.е. имя колонки $(i | p_i) \in A$, является именем образа p и решением прямой задачи. В противном случае выбирается некоторое чистое имя i_p из

соответствующей области имен и выполняется сложение $A + (i_p | p)$. Если образ p появится снова, то поэлементное сравнение даст для него имя i_p .

Еще более просто решается обратная задача. Если имеется имя i , для которого необходимо решить обратную задачу, то соответствующий образ равен образу p_i колонки $(i | p_i) \in A$. Если колонки с таким именем не существует, то i – чистое имя.

Очевидно, для образов тех или иных типов могут существовать более эффективные методы решения прямой и обратной задачи. Например, применение хэш-функций для образов в виде конечных последовательностей или векторов может значительно сократить объемы поэлементного сравнения. Для этих же образов более эффективным оказывается и метод пересечений.

Идея метода пересечений восходит к книжному алфавитному указателю. В нем для каждой рубрики приведено множество указателей на те страницы книги, где эту рубрику можно обнаружить. Запросу из нескольких рубрик, очевидно, соответствует пересечение множеств указателей для этих рубрик. Подобные методы с таблично заданными индексами используются в поисковых машинах интернета, в частности, в Google [3]. В работах [1, 5] метод пересечений применяется для решения задач распознавания.

Далее в разделе метод пересечений рассматривается применительно к базовым задачам для наиболее очевидных и простых типов образов – образов в виде конечных неупорядоченных множеств и образов в виде конечных последовательностей или векторов. Предлагаемый в данной работе метод характеризуется полным отсутствием необходимости в поэлементном сравнении. Кроме того, для обратной задачи предложено решение, в котором для того, чтобы обеспечить компромисс между быстродействием и объемом требуемой памяти, используется упорядоченный индекс.

3.1. МЕТОД ПЕРЕСЕЧЕНИЙ ДЛЯ ОБРАЗОВ В ВИДЕ КОНЕЧНЫХ НЕУПОРЯДОЧЕННЫХ МНОЖЕСТВ

3.1.1. ПРЯМАЯ ЗАДАЧА

Будем полагать, что образы представляют собой конечные неупорядоченные множества имен вида $p = \{i_1, \dots, i_m\}$, $m \geq 1$, где $i_k \in U'$, U' – некоторая область имен. Очевидно, любой такой образ $p \in P$, где P – множество всех подмножеств множества U' , исключая пустое.

Для решения прямой задачи будет использоваться индекс A . В исходном состоянии индекс $A = \emptyset$.

Пусть на вход системы пришел некоторый образ $p_1 \in P$. Так как система еще ничего не знает, то p_1 – новый неизвестный образ. У системы есть область имен U'' , которые она может использовать для наименования неизвестных образов. Если через $U_p \subset U''$ обозначить множество имен всех известных образов, то для наименования образа p_1 система может взять любое чистое имя из $U'' \setminus U_p$. Пусть выбрано имя i_1 . Для запоминания образа p_1 под именем i_1 выполняется сложение $A + (p_1 | \{i_1\})$, т.е. к индексу A добавляется $n_1 = |p_1|$ колонок $(k | \{i_1\})$ для всех элементов $k \in p_1$.

Если системе опять будет предъявлен образ p_1 , то пересечение n_1 образов колонок индекса A для всех имен $k \in p_1$, очевидно, даст множество, состоящее из одного имени i_1 , под которым системе известен образ p_1 , т.е. $\bigcap_{k \in p_1} a_k = \{i_1\}$, где a_k – образ колонки $(k | a_k) \in A$.

Обозначим через $\eta(p)$ пересечение образов колонок $(k | a_k) \in A$ для всех имен $k \in p$, т.е. $\eta(p) = \bigcap_{k \in p} a_k$. При этом

будем полагать, что если в индексе A отсутствует колонка с именем k , то $a_k = \emptyset$.

Предположим, что теперь на вход поступил некоторый образ $p_2 \in P$. Рассмотрим для него пересечение $\eta(p_2)$. Очевидно, равенство $\eta(p_2) = \{i_1\}$ возможно только в том случае, если $p_2 \subset p_1$, $|p_2| \leq |p_1|$. Причем при $|p_2| = |p_1|$ имеет место равенство

$p_2 = p_1$. Если же $p_2 \not\subset p_1$, то всегда $\eta(p_2) = \emptyset$, так как в индексе A просто отсутствуют колонки с именами, принадлежащими множеству $p_2 \setminus p_1$.

Следовательно, для образа p_2 возможны три случая:

1. $\eta(p_2) = \{i_1\}$, $|p_2| = |p_1|$ при $p_2 = p_1$;
2. $\eta(p_2) = \{i_1\}$, $|p_2| < |p_1|$ при $p_2 \subset p_1$;
3. $\eta(p_2) = \emptyset$ при $p_2 \not\subset p_1$.

Если рассматривается вариант задачи, когда система работает с образами одинаковой мощности, возможны только случаи 1 и 3. При этом равенство $\eta(p_2) = \emptyset$ является признаком того, что образ p_2 неизвестен системе и его надо запомнить. При $\eta(p_2) = \{i_1\}$ образ p_2 известен системе под именем i_1 и $p_2 = p_1$.

Если же на вход системы поступают образы различной мощности, то равенство $\eta(p_2) = \emptyset$ также является признаком того, что образ p_2 неизвестен системе и его надо запомнить. Однако при $\eta(p_2) = \{i_1\}$ возможны случаи 1 и 2 – либо входной образ известен под именем i_1 , либо входной образ неизвестен, но представляет собой подмножество образа, известного под именем i_1 . Чтобы выделить случай 2, необходимо выполнить сравнение мощностей образов. Для этого в системе используется функция $n(i)$, которая определена на множестве U_p и задана с помощью множества пар (аргумент, значение). Каждый раз, когда система запоминает новый образ p , в определение функции $n(i)$ добавляется новая пара $(i, |p|)$, где i – имя, под которым запоминается образ p , $|p|$ – его мощность.

Вернемся к моменту, когда на вход системы поступил некоторый образ p_2 . Если $\eta(p_2) = \{i_1\}$ и $n(i_1) = |p_2|$, то образ p_2 известен системе под именем i_1 и $p_2 = p_1$.

Если $\eta(p_2) = \emptyset$ или если $\eta(p_2) = \{i_1\}$, но $n(i_1) \neq |p_2|$, то образ p_2 является новым и его надо запомнить. Системой выбирается любое чистое имя $i_2 \in U'' \setminus U_p$, под которым теперь будет известен образ p_2 , и выполняется сложение $A + (p_2 | \{i_2\})$. В определение функции $n(i)$ добавляется новая пара $(i_2, |p_2|)$, а решением прямой задачи для образа p_2 будет имя i_2 .

Очевидно, при выполнении сложения $A + (p_2 | \{i_2\})$ факторизация по имени приведет к тому, что любая колонка индекса A с именем $k \in p_1 \cap p_2$ будет иметь вид $(k | \{i_1, i_2\})$. Если рассмотреть все возможные образы индекса A , то факторизация по образу показывает, что полученный в результате сложения индекс A состоит из трех колонок $(p_1 \setminus p_2 | \{i_1\})$, $(p_1 \cap p_2 | \{i_1, i_2\})$ и $(p_2 \setminus p_1 | \{i_2\})$, где $p_1 \setminus p_2$, $p_1 \cap p_2$ и $p_2 \setminus p_1$ – классы эквивалентности.

Аналогично тому, как это делалось для образа p_2 , будут запоминаться и другие новые образы p_3, p_4, p_5 и т.д. Запоминание l новых образов можно представить в виде цепочки сложений:

$$A = \sum_{k=1}^l (p_k | \{i_k\}), \quad n(i) = \bigcup_{k=1}^l (i_k | p_k),$$

где i_k – имя образа p_k .

Запоминание любого образа p будет выполняться только при условии, что $\eta(p) = \emptyset$ или $\eta(p) \neq \emptyset$ и $n(i) \neq |p|$ для $\forall i \in \eta(p)$. Покажем, что в этом случае при появлении любого образа $p \in P$ всегда можно однозначно сказать, известен ли этот образ, и определить его имя.

Рассмотрим любой образ $p \in P$. Пусть $\eta(p) = \emptyset$. Это означает, что образ p не предъявлялся системе, так как в противном случае по построению существовали бы колонки для $\forall k \in p$ и пересечение $\eta(p)$ содержало бы имя, под которым известен образ p .

Пусть теперь $\eta(p) \neq \emptyset$ и $n(i) \neq |p|$ для $\forall i \in \eta(p)$. Это также означает, что образ p не предъявлялся системе, так как в противном случае по построению пересечение $\eta(p)$ содержало бы имя i , под которым известен образ p , и выполнялось бы равенство $n(i) = |p|$.

Наконец, пусть $\eta(p) \neq \emptyset$ и существует по крайней мере одно имя $i \in \eta(p)$, для которого $n(i) = |p|$. Это означает, что образ p известен системе под именем i . Покажем, что такое имя является единственным.

Предположим, что $\exists i, i' \in \eta(p)$, $i \neq i'$ и $n(i) = n(i') = |p|$. Так как образ p не может одновременно получить сразу два имени,

то это означает, что он сначала получил одно имя, а потом в одном из повторных предъявлений – второе. Это невозможно по построению, т.е. $i = i'$, и если существует такое имя $i \in \eta(p)$, что $n(i) = |p|$, то это имя является единственным.

Таким образом, рассмотренный метод дает решение прямой задачи для любого образа $p \in P$. Схему его работы можно описать следующим образом.

В исходном состоянии $A = \emptyset$ и $n(i) = \emptyset$.

1. Для предъявленного образа $p \in P$ вычисляется пересечение $\eta(p)$.

2. Если $\eta(p) \neq \emptyset$ и $\exists i \in \eta(p)$ такое, что $n(i) = |p|$, то имя i единственное, представляет собой имя, под которым известен образ p , и является решением прямой задачи.

3. В противном случае, если $\eta(p) = \emptyset$ или $\eta(p) \neq \emptyset$, но $n(i) \neq |p|$ для $\forall i$, образ p является новым. Для него выбирается любое чистое имя $i \in U'' \setminus U_p$, выполняется сложение $A + (p \mid \{i\})$, а в определение функции $n(i)$ добавляется пара $(i, |p|)$. Имя i , под которым теперь будет известен образ p , является решением прямой задачи.

Здесь необходимо сделать одно важное замечание. Выбор для нового образа чистого имени из множества $U'' \setminus U_p$ обеспечивает выполнение условия, которое для правильной работы приведенного метода является необходимым. Это условие состоит в том, что каждый образ получает единственное уникальное имя, т.е. разные образы не могут иметь одно и то же имя.

Легко показать, что при нарушении этого условия рассмотренный метод не сможет различать образы.

Действительно, пусть два разных образа p_1 и p_2 имеют одно и то же имя i и $|p_1| = |p_2| = n$. Рассмотрим любой образ p_3 , $|p_3| = n$, который имеет часть элементов из образа p_1 , а оставшуюся часть из образа p_2 . Очевидно, образ p_3 не равен образам p_1 и p_2 , но при этом $\eta(p_3) = \eta(p_1) = \eta(p_2) = i$ и $|p_3| = |p_1| = |p_2| = n(i)$, т.е. данный метод не сможет отличить образ p_3 от образов p_1 и p_2 .

3.1.2. ОБРАТНАЯ ЗАДАЧА

Обратная задача состоит в том, чтобы по имени образа i получить сам образ p_i . Она будет решаться обычным образом. Для этого кроме индекса A будет использоваться дополнительный индекс B , в который при запоминании образа p под именем i будет добавляться колонка $(i | p)$.

В начале работы $A = \emptyset$, $B = \emptyset$ и $n(i) = \emptyset$.

Если образ p является новым, то для него выбирается некоторое чистое имя $i \in U'' \setminus U_p$ и выполняются сложения $A + (p | \{i\})$, $B + (i | p)$. В определение функции $n(i)$ добавляется пара $(i, |p|)$. Таким образом, процесс запоминания l новых образов можно представить цепочкой сложений:

$$A = \sum_{k=1}^l (p_k | \{i_k\}), \quad B = \sum_{k=1}^l (i_k | p_k), \quad n(i) = \bigcup_{k=1}^l (i_k, |p_k|).$$

Пусть решается обратная задача и предъявлено некоторое имя $i \in U''$. Если имя i представляет собой имя известного образа, т.е. $i \in U_p$, то решением обратной задачи является образ p_i колонки $(i | p_i) \in B$. В противном случае имя i является чистым.

Нетрудно заметить, что $B = A^{-1}$. Для доказательства достаточно использовать определение обратного индекса по отношению к колонкам $(p_k | \{i_k\})$ и $(i_k | p_k)$.

ПРИМЕР. Пусть индексы A , B и функция $n(i)$ имеют вид:

A		B														
3 3 3		3														
1 2 1 2	$n(i)$	3 4 2														
1 2 3 4	<table border="1" style="border-collapse: collapse; text-align: center;"> <tr> <td style="padding: 2px 5px;">i</td> <td style="padding: 2px 5px;">1</td> <td style="padding: 2px 5px;">2</td> <td style="padding: 2px 5px;">3</td> </tr> <tr> <td style="padding: 2px 5px;">n</td> <td style="padding: 2px 5px;">2</td> <td style="padding: 2px 5px;">2</td> <td style="padding: 2px 5px;">3</td> </tr> </table>	i	1	2	3	n	2	2	3	<table border="1" style="border-collapse: collapse; text-align: center;"> <tr> <td style="padding: 2px 5px;">1</td> <td style="padding: 2px 5px;">2</td> <td style="padding: 2px 5px;">1</td> </tr> <tr> <td style="padding: 2px 5px;">1</td> <td style="padding: 2px 5px;">2</td> <td style="padding: 2px 5px;">3</td> </tr> </table>	1	2	1	1	2	3
i	1	2	3													
n	2	2	3													
1	2	1														
1	2	3														

Легко видеть, что системе известны три образа: образ $\{1, 3\}$ под именем 1, образ $\{2, 4\}$ под именем 2 и образ $\{1, 2, 3\}$ под именем 3.

Пусть на входе появился образ $p = \{1, 2, 3\}$. Пересечение $\eta(p) = \{3\}$ и $n(3) = |p| = 3$. Следовательно, образ p известен системе под именем 3.

Пусть теперь пришел образ $p = \{2, 3\}$. Пересечение $\eta(p) = \{3\}$, но $n(3) = 3 \neq |p|$. Следовательно, образ p является новым. Для него выбирается имя $4 \in U'' \setminus U_p$. Выполняется сложение $A + (p | \{4\})$ и $B + (4 | p)$, а в определении функции $n(i)$ добавляется пара $(4, 2)$. В результате получим:

A		B										
4 4		3										
3 3 3	$n(i)$	3 4 2 3										
1 2 1 2	<table border="1" style="border-collapse: collapse; text-align: center;"> <tr> <td style="border: none;">i</td> <td>1</td> <td>2</td> <td>3</td> <td>4</td> </tr> <tr> <td style="border: none;">n</td> <td>2</td> <td>2</td> <td>3</td> <td>2</td> </tr> </table>	i	1	2	3	4	n	2	2	3	2	1 2 1 2
i	1	2	3	4								
n	2	2	3	2								
1 2 3 4		1 2 3 4										

Если на входе опять появится образ $p = \{2, 3\}$, то $\eta(p) = \{3, 4\}$, причем $n(4) = |p| = 2$, что означает, что образ p известен системе под именем 4.

Пусть теперь решается обратная задача и предъявлено имя $i = 3$. Так как $i \in U_p$, то образ p_i равен образу колонки 3 индекса B , т.е. $p_i = \{1, 2, 3\}$. ■

Приведенное выше решение обратной задачи имеет один недостаток – в индексе B должны сохраняться все известные образы. В результате значительно возрастает объем требуемой памяти. Очевидным выходом является сохранение в индексе B лишь части известных образов, например, тех, которые встречаются на отрезке из m последних моментов времени. При этом индекс B будет представлять собой упорядоченный индекс, содержащий m последних событий, каждое из которых связано с появлением некоторого образа на входе системы (см. раздел 2.7). Для всех остальных образов может использоваться более медленная схема восстановления образа путем просмотра колонок индекса A [2].

3.2. МЕТОД ПЕРЕСЕЧЕНИЙ ДЛЯ ОБРАЗОВ В ВИДЕ КОНЕЧНЫХ ПОСЛЕДОВАТЕЛЬНОСТЕЙ ИЛИ ВЕКТОРОВ

Рассмотрим теперь прямую и обратную задачи для образов в виде конечных последовательностей или векторов. Будем полагать, что в этом случае, образы $p = (i_1, i_2, \dots, i_m)$ принадлежат множеству $P = \bigcup_{m=1}^n P^m$, где $P^m = U_1 \times \dots \times U_m$, U_k – область

имен k -й координаты. Длина или размерность последовательности p будет обозначаться через $|p|$.

При решении прямой задачи будет использоваться индекс $A = \{A_1, A_2, \dots, A_n\}$, который представляет собой индекс уровня L^2 (см. раздел 2.4), где A_k – индекс для k -й координаты.

Кроме индекса A также будет использоваться функция $m(i)$, которая будет содержать длину известных последовательностей. В исходном состоянии $A_k = \emptyset$ ($k = 1, \dots, n$), $m(i) = \emptyset$. Запоминание нового образа $p = (i_1, i_2, \dots, i_m)$ под некоторым именем i выполняется как покоординатное сложение $A_k + (i_k | \{i\})$, $k = 1, \dots, m$, где i_k – имя, являющимся k -й координатой входного образа p :

$$A + (p | \{i\}) = \{A_1 + (i_1 | \{i\}), \dots, A_m + (i_m | \{i\})\}.$$

Кроме этого, в определении функции $m(i)$ добавляется пара $(i, |p|)$. Таким образом, запоминание системой l новых образов можно представить в виде цепочки сложений:

$$A = \sum_{k=1}^l (p_k | \{i_{p_k}\}), \quad m(i) = \bigcup_{k=1}^l (i_{p_k}, |p_k|),$$

где i_{p_k} – имя образа p_k .

Обозначим через $\eta(p)$ покоординатное пересечение $\eta(p) = \bigcap_{k=1}^m a_{i_k}$, где a_{i_k} – образ колонки $(i_k | a_{i_k}) \in A_k$, i_k – имя, являющееся k -й координатой входного образа p .

Пусть на вход системы пришел некоторый образ $p = (i_1, i_2, \dots, i_m)$.

Очевидно, если образ p неизвестен и не является подпоследовательностью известных образов, то $\eta(p) = \emptyset$. Если $\eta(p) \neq \emptyset$

и $|p| \neq m(i)$ для $\forall i \in \eta(p)$, то образ p также неизвестен, но его координаты совпадают с частью координат известных образов, имена которых принадлежат множеству $\eta(p)$. Наконец, если $\eta(p) \neq \emptyset$ и существует имя $i \in \eta(p)$, такое, что $|p| = m(i)$, то образ p известен под именем i . Причем в последнем случае такое имя единственное, так как повторное запоминание известного образа под другим именем невозможно.

Схема работы метода имеет вид.

В исходном состоянии $A_k = \emptyset$ ($k = 1, \dots, n$), $m(i) = \emptyset$.

1. Для появившегося образа $p \in P$ вычисляется покоординатное пересечение $\eta(p)$.

2. Если $\eta(p) \neq \emptyset$ и $\exists i \in \eta(p)$ такое, что $n(i) = |p|$, то имя i единственное, представляет собой имя образа p и является решением прямой задачи.

3. В противном случае, если $\eta(p) = \emptyset$ или $\eta(p) \neq \emptyset$, но $n(i) \neq |p|$ для $\forall i \in \eta(p)$, образ p является новым. Для него выбирается любое чистое имя $i \in U'' \setminus U_p$, выполняется сложение $A + (p \mid \{i\})$, а в определение функции $m(i)$ добавляется пара $(i, |p|)$. Имя i , под которым теперь будет известен образ p , является решением прямой задачи.

Как и ранее необходимым условием для правильной работы метода является то, что каждый образ получает единственное уникальное имя, что обеспечивается выбором чистого имени из множества $U'' \setminus U_p$. При нарушении этого условия метод не сможет различать образы.

Действительно, пусть образы p_1 и p_2 , имеющие одно и то же имя i , отличаются всеми координатами и $|p_1| = |p_2|$. Тогда, взяв образ той же размерности, у которого одна часть координат равна координатам образа p_1 , а вторая часть – координатам образа p_2 , получим образ p_3 , отличный от p_1 и p_2 . При этом, очевидно, $\eta(p_3) = \eta(p_1) = \eta(p_2) = i$ и $|p_3| = |p_1| = |p_2| = m(i)$, т.е. метод не сможет отличить образ p_3 от образов p_1 и p_2 .

Обратная задача решается аналогично тому, как это делалось для образов в виде неупорядоченных множеств. При этом используется дополнительный индекс B , с которым при запоми-

нании образа p складывается колонка $(i | p)$. Запоминание l новых образов можно представить цепочкой сложений:

$$A = \sum_{k=1}^l (p_k | \{i_{p_k}\}), \quad B = \sum_{k=1}^l (i_{p_k} | p_k), \quad m(i) = \bigcup_{k=1}^l (i_{p_k} | p_k |),$$

где i_{p_k} – имя образа p_k .

ПРИМЕР. Пусть для $n = 3$ имеются следующие индексы A , B и функция $m(i)$:

A_1			A_2			A_3			B		
									3	2	2
4	2	2				4			3	1	1
1	3		3	4	1	3	2		1	3	3
1	2	3	1	2	3	1	2	3	1	2	3

$m(i)$

i	1	2	3	4
m	2	3	3	3

Легко видеть, что в индексе $A = \{A_1, A_2, A_3\}$ под именем 1 запомнен образ (1, 3), под именем 2 – образ (3, 1, 3), под именем 3 – образ (3, 1, 2), а под именем 4 – образ (1, 2, 2).

Пусть на вход пришел образ $p = (3, 1, 2)$. Пересечение $\eta(p)$, равное пересечению образа колонки индекса A_1 с именем 3, образа колонки индекса A_2 с именем 1 и образа колонки индекса A_3 с именем 2, будет состоять из одного имени 3, при этом $m(3) = 3$. Это означает, что на входе образ, известный под именем 3.

Пусть теперь на входе появился образ $p = (3, 1)$. Пересечение $\eta(p) = \{2, 3\}$, но $m(2) = m(3) \neq 2$. Следовательно, p – это неизвестный новый образ. Из множества $U'' \setminus U_p$ для него выбирается имя 5. Затем выполняются сложения $A + ((3, 1) | 5)$ и $B + (5 | (3, 1))$, а в определение функции $m(i)$ добавляется пара (5, 2). В результате получим:

	A_1		A_2		A_3		B				
	5		5				3	2	2		
4	2		2		4		3	1	1	2	1
1	3		3	4	1		1	3	3	1	3
1	2	3	1	2	3	1	2	3	4	5	

$m(i)$

i	1	2	3	4	5
m	2	3	3	3	2

Если на входе снова появится образ $p = (3, 1)$, то пересечение $\eta(p) = \{2, 3, 5\}$, причем $m(5) = 2$. Следовательно, на входе образ, известный под именем 5.

Пусть теперь решается обратная задача и на вход пришло имя $i = 3$. Так как $i \in U_p$, то образ p_i равен образу колонки 3 индекса B , т.е. $p_i = (3, 1, 2)$.

3.3. ВЫЧИСЛИТЕЛЬНАЯ ЭФФЕКТИВНОСТЬ МЕТОДА ПЕРЕСЕЧЕНИЙ

Пусть решается прямая задача для образов в виде конечных последовательностей или векторов одинаковой размерности. Пусть также l – это число запомненных образов, n – размерность вектора, m – число возможных значений каждой координаты.

Рассмотрим число операций, которые необходимо выполнить при решении прямой задачи методом поэлементного сравнения и с помощью рассмотренного выше метода пересечений.

При поэлементном сравнении известные системе образы хранятся в индексе A , который представляет собой множество колонок вида $(i | p_i)$, где p_i – образ, известный под именем i . Если на входе появился некоторый вектор $p = (i_1, i_2, \dots, i_n)$, то для решения прямой задачи выполняется его покоординатное сравнение с образами всех колонок индекса A . Если найден совпадающий образ p_i , то его имя, т.е. имя колонки $(i | p_i) \in A$, является именем образа p и решением прямой задачи. Максимальное

число операций, которые при этом необходимо выполнить, равно $O_M = nl$.

Оценим теперь число операций, необходимое для решения прямой задачи при помощи метода пересечений с использованием индекса $A = \{A_1, A_2, \dots, A_n\}$. Каждая из m колонок индекса A_k ($k = 1, \dots, n$) содержит в среднем l/m имен. При вычислении пересечения $\eta(p)$ необходимо выполнить $O_A = nl/m$ операций, т.е. в m раз меньше чем при покоординатном сравнении. Другими словами, чем больше имен использует система для представления значений координат, т.е. чем точнее она отражает реальные объекты, тем она быстрее работает¹. При большой величине m выигрыш по сравнению с покоординатным сравнением может быть довольно значительным. Например, пусть запомнено $l = 10^5$ образов размерности $n = 10^2$, каждая координата которых может иметь $m = 10^3$ значений. Тогда при покоординатном сравнении необходимо выполнить 10 млн. операций, а при вычислении пересечения $\eta(p)$ только 10 тыс. операций. Кроме этого, следует отметить, что метод пересечений хорошо распараллеливается.

Оценки памяти, необходимой для этих двух методов, совпадают. И в том, и в другом случае в памяти должно храниться nl целых чисел.

4. Классы задач, для решения которых могут использоваться системы на основе колонок

Базовые задачи служат основой для решения других задач. В данном разделе это сначала будет показано на примере задачи классификации, а затем будет доказана возможность реализации в рассматриваемых системах произвольных булевых функций. Эти результаты позволят оценить те клас-

¹ Аналогичное утверждение справедливо и для образов в виде неупорядоченных множеств. В этом случае m – число различных имен, участвующих в формировании образов.

сы задач, для которых могут применяться интеллектуальные системы на основе колонок.

4.1. ЗАДАЧА КЛАССИФИКАЦИИ

Рассмотрим задачу классификации, в которой любой объект из некоторой проблемной области описывается с помощью образа $p \in P$ и принадлежит одному или нескольким классам объектов. Задача состоит в том, чтобы по входному образу $p \in P$ определить, к какому классу или классам он принадлежит.

Одно из основных отличий данной задачи от рассмотренных выше связано с наименованием неизвестных образов. Если ранее система присваивала новому образу любое имя из имеющихся у нее чистых имен, то теперь она должна связать этот образ с совершенно определенным множеством имен классов.

Будем полагать, что для каждого нового образа система имеет возможность получить множество имен классов, к которым этот образ принадлежит. В этом случае для решения задачи классификации можно применить любой из методов решения прямой задачи, используя дополнительный индекс B для запоминания имен классов. Действительно, если входной образ p является новым, то достаточно его запомнить под именем i_p – прямая задача, а в индекс B добавить колонку $(i_p | q_p)$, где $q_p = \{c_1, \dots, c_{n_p}\}$ – множество имен классов, к которым принадлежит образ p , $c_k \in U_q$, U_q – область имен классов. Тогда при предъявлении образа p сначала определяется его имя i_p , а затем по нему колонка $(i_p | q_p)$ индекса B . Образ этой колонки q_p и есть множество имен классов, к которым принадлежит входной образ p .

4.2. БУЛЕВЫ ФУНКЦИИ

Рассмотрим произвольную булеву функцию $f: B^n \rightarrow B$, $B = \{0, 1\}$. Она определяет разбиение B^n на классы эквивалентности:

$$B_0^n = \{x \in B^n \mid f(x) = 0\},$$

$$B_1^n = \{x \in B^n \mid f(x) = 1\},$$

где $B_0^n \cup B_1^n = B^n$, $B_0^n \cap B_1^n = \emptyset$.

Очевидно, для того чтобы реализовать функцию f с помощью системы на основе колонок, достаточно решить задачу классификации для двух классов B_0^n и B_1^n . Положим для наглядности, что классы B_0^n и B_1^n получили имена 0 и 1 из некоторой области имен U_q . Тогда для любого вектора $p \in B^n$ будем иметь: если p принадлежит классу с именем 0, то функция $f(p) = 0$; если же p принадлежит классу с именем 1, то $f(p) = 1$.

Так как мы имеем дело с разбиением множества B^n всего на два класса, то задачу можно значительно упростить, оставив только один класс, например, B_1^n . Тогда для $\forall p \in B^n$, если p принадлежит классу 1, то $f(p) = 1$, в противном случае $f(p) = 0$.

Данная задача классификации может быть решена так, как это описывалось в предыдущем разделе. При этом может использоваться любой подходящий метод решения прямой задачи. Таким образом, любая булева функция $f: B^n \rightarrow B$, $B = \{0, 1\}$ может быть реализована в системе на основе колонок.

Покажем более подробно, как это будет выглядеть при использовании метода пересечений.

Итак, пусть образы представляют собой вектора $p \in P$, $P = U_B^n$, где $U_B = \{0, 1\}$ – область имен для любой координаты.

Так как все вектора имеют одну и ту же размерность, а класс всего один, то функцию $m(i)$ и индекс B можно опустить.

Сначала при помощи индекса $A = \{A_1, \dots, A_n\}$ запоминаются все вектора класса B_1^n :

$$A = \sum_{k=1}^l (p_k | \{i_k\}),$$

где $p_k \in B_1^n$, i_k – имя вектора p_k , $l = |B_1^n|$.

В результате системе известны только вектора класса B_1^n . Других векторов она не знает.

Далее система работает только в режиме классификации. Пусть $p \in P$ – произвольный входной вектор. Если $\eta(p) \neq \emptyset$, то

вектор p известен системе, $p \in B_1^n$ и $f(p) = 1$. В противном случае $f(p) = 0$. ■

Этот результат показывает возможность решения с помощью рассматриваемых систем тех задач, для которых может применяться логический подход. Это относится и к логическому выводу, используемому, в частности, в синтетических задачах. Вместе с результатами раздела 4.1 это позволяет говорить о принципиальной возможности применения интеллектуальных систем на основе колонок для решения различных задач классификационного и синтетического типа.

Литература

1. МИХАЙЛОВ А.М. *Распознавание образов с помощью их индексирования* // Автоматика и телемеханика. – 2012. – №4. – С. 151–161.
2. ЧЕШОКОВ А.М. *Введение в общую теорию колонок*. – М.: ИПУ РАН. – 2012. – 141 с.
3. BRIN S., PAGE L. *The Anatomy of a Large-Scale Hypertextual Web Search Engine* // Proc. Seventh International Conference on World Wide Web (WWW7). – Amsterdam: Elsevier, 1998. – P. 107–117.
4. HAWKINS J. *On Intelligence*. – New York: Henry Holt and Company, 2005. – 262 p.
5. MIKHAILOV A. *Biologically Inspired Artificial Neural Cortex and its Formalism* // World Academy of Science, Engineering and Technology. – August 2009. – Vol. 56. – P. 121.

COLUMNS-BASED INTELLIGENT SYSTEMS

Alexander Chesnokov, Institute of Control Sciences of RAS, Moscow, Cand. Sc. (alex-ches@yandex.ru).

Abstract: The paper considers columns-based intelligent systems and introduces basic definitions and notions. The direct and inverse problems of patterns are introduced. Solutions to these problems and their computational complexity are discussed. Also, a solution to pattern classification problem and a method of Boolean functions implementation are introduced. Finally, a class of problems that can be solved with columns-based intelligent systems is identified.

Keywords: artificial intelligence, columns-based intelligent systems, column.

Статья представлена к публикации членом редакционной коллегии О.П. Кузнецовым

*Поступила в редакцию 13.03.2013.
Опубликована 30.11.2013.*

УДК 62.50
ББК Ж 30

ПРИМЕНЕНИЕ ИДЕЙ ПРОГНОЗИРУЮЩЕГО УПРАВЛЕНИЯ В СИНТЕЗЕ СТАБИЛИЗИРУЮЩЕГО УПРАВЛЕНИЯ СЕТЕВЫМИ ОБЪЕКТАМИ¹

Жучков Р. Н.²

*(Арзамасский политехнический институт (филиал)
Нижегородского государственного технического университета
им. Р.Е. Алексеева, Арзамас)*

Рассматривается линейная система, в которой регулятор обменивается информацией с объектом управления через сеть, в которой возможны потери пакетов данных. Строится динамический регулятор с обратной связью по вектору выхода. Для получения оценки вектора состояния в любой момент дискретного времени k используются методы теории прогнозирующего управления, которые позволяют избежать необходимости явного учета смены структурного состояния системы в моменты потери пакетов данных.

Ключевые слова: прогнозирующее управление, фильтр Калмана, сетевые системы управления, линейные матричные неравенства, линейные дискретные системы.

Основные идеи прогнозирующего управления

Модель прогнозирующего управления относится к классу алгоритмов управления, которые используют модели предсказания будущих реакций системы. Изначально подход разрабатывался для удовлетворения специфических потребностей управления электростанций и нефтеперерабатывающих заводов. В настоящее

¹ Работа выполнена при частичной финансовой поддержке РФФИ (грант 13-08-01092_а) и Министерства образования и науки РФ, ФЦП «КАДРЫ» (соглашение 8846).

² Роман Николаевич Жучков, аспирант, (roman_jkv@mail.ru).

время прогнозирующее управление можно найти в самых различных областях применения: пищевой, автомобильной и аэрокосмической промышленности.

Суть подхода в следующем: на каждом интервале управления алгоритм пытается оптимизировать будущее поведение системы путем вычисления последовательности будущих управлений. Последовательность управлений рассчитывается таким образом, чтобы оптимизировать будущее поведение системы в течение интервала времени, получившего название горизонта предсказаний. Первое управление из полученной последовательности отправляется объекту, и в следующий момент времени задача управления решается заново, используя обновленные измерения.

Идеи прогнозирующего управления можно проследить до 1960х годов [12], но интерес к этим методам начал быстро расти только в 1980х годах после публикаций первых работ по прогнозирующему управлению: Identification and Command (IDCOM) [19], динамическому матричному управлению (Dynamic Matrix Control [9, 10]) и первому всестороннему изложению идеи обобщенного прогнозирующего управления Generalized Predictive Control (GPC) [6, 7]. Хотя в изначальном виде идеи, лежащие в основе DMC и GPC, схожи, DMC был задуман для многомерного управления с ограничениями, в то время как GPC в первую очередь подходит для одной переменной, а также адаптивного управления.

Название «прогнозирующее управление» связано с идеей использования явной модели объекта, который используется для предсказания будущего поведения выхода системы. Это предсказание позволяет решать задачи оптимального управления в реальном времени, где ошибки слежения, а именно отличия между предсказанным выходом и желаемым выходом сведены к минимуму за время горизонта предсказания, при условии ограничений на входы и выходы. Когда модель является линейной, то задача оптимизации является квадратичной, если показатель качества представляется через L_2 норму, или линейной, если выражает-

ся через L_1/L_∞ -норму. Полученное управление применяется в соответствии с основным принципом прогнозирующего управления: в момент времени t только первый член из предсказанной последовательности оптимальных управлений применяется к объекту. Затем решается задача управления для времени $t + 1$.

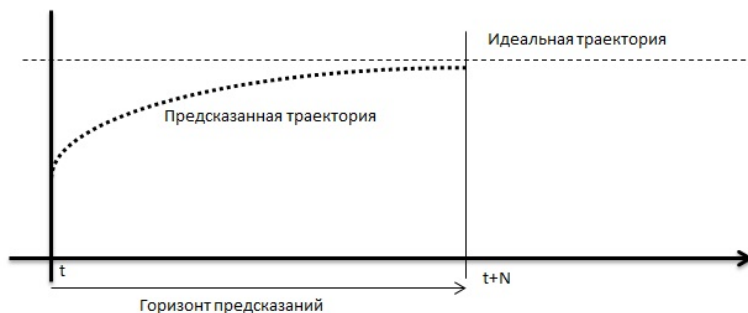


Рис. 1. Прогнозательное управление

На рис. 1 показана основная идея прогнозирующего управления [17]. Мы ограничимся обсуждением системы с одним входом и одним выходом. Будем рассматривать систему с дискретным временем. Текущее состояние обозначено как шаг t . На рисунке показаны две траектории: пунктирная – идеальная траектория и предсказанная траектория (обозначена точками). Предсказанная траектория начинается с текущего момента t и определяет траекторию, двигаясь по которой объект должен вернуться на идеальную траекторию. Прогнозирующий регулятор имеет собственную модель, которая используется для прогнозирования поведения системы внутри горизонта предсказаний. В простейшем случае мы можем попытаться выбрать предсказанную траекторию таким образом, чтобы совместить ее с идеальной в конце горизонта предсказаний.

Вопросы возможности оптимизации в реальном времени, устойчивости и качества широко изучены для систем, описываемых линейными моделями (книги [3–5, 8, 16, 20]). Значительный прогресс в применении прогнозирующего управления для

гибридных систем, дискретных систем, систем с логическими условиями, эвристического анализа был получен в [2].

Приведем несколько результатов, полученных в области прогнозирующего управления для управления сетевыми системами. В [15] предложено стабилизирующее управление для сетевой системы с потерями пакетов данных между сенсором и регулятором. В [11] предложен метод для случая двусторонней потери данных. В этой работе зависящая от потери пакетов данных функция Ляпунова используется для стабилизации замкнутой системы. В [13] авторы предложили сетевую систему управления, которая использует предсказания, чтобы компенсировать запаздывания сигналов и потери пакетов данных при передаче между объектом и системой управления. Для того чтобы анализировать свойства системы, они ввели понятие последовательности предсказаний, что позволяет определять свойства сети, необходимые для достижения устойчивости замкнутой системы.

В предыдущих работах автора [1] потеря пакетов данных рассматривалась как одно из структурных состояний сетевой системы управления. Однако, если говорить об объекте управления, то в такие моменты он не переходит в какое-то новое состояние, продолжая функционировать в обычном режиме. Переключение сконструированной системы управления в другое структурное состояние характеризует не сам объект, для которого строится контур управления, но информацию о нем.

С этой точки зрения применение принципов прогнозирующего управления очень привлекательно именно с точки зрения нахождения системы в одном структурном состоянии, когда информация о ней доступна. Хотя нужно помнить, что иногда эта информация будет предсказанной, а не истинной.

Сформулируем здесь основные идеи, использованные в предыдущей работе при построении алгоритмов управления сетевыми системами управления. Во-первых, принималась гипотеза о возможности разделения системы на объект и наблюдатель. После этого матрицы усиления для наблюдателя и регулятора находились по отдельности. Во-вторых, для наблюдателя и регу-

лятора рассматривались возможности потери пакетов данных и находились либо разные матрицы усиления для разных структурных состояний, либо единая матрица для всех структурных состояний.

Используя идеи прогнозирующего управления, алгоритм управления сетевой системой предлагается модифицировать следующим образом: введем горизонт предсказаний, равный максимально возможному количеству потерянных пакетов данных. Эта величина является характеристикой сетевого канала обмена данными и может быть выбрана с запасом. Далее вводятся два буфера: с измерениями и управлениями, в которых будут храниться предсказанные измерения и управления. Ниже будет показано, как для построения оценки вектора состояния с учётом прогнозирующего управления используется алгоритм фильтра Калмана. Основываясь на предсказаниях текущего и будущих состояний объекта управления строится последовательность управлений, которая высылается объекту. Обратим здесь внимание на то обстоятельство, что в отличие от работы [18], например, будет использоваться одна и та же матрица усиления обратной связи для текущего и будущих моментов времени. Сделано это из того предположения, что изначально для системы может быть найдена единая матрица усиления обратной связи и потери пакетов данных не меняют ее структуру.

1. Схема функционирования системы

Пусть имеется сетевая система управления, изображенная на рис. 2. Работа ее построена следующим образом:

- 1) объект управления формирует измерения в виде текущего и предсказанных измерений на несколько шагов вперед (горизонт событий);
- 2) объект управления пересылает сформированный пакет системе управления;

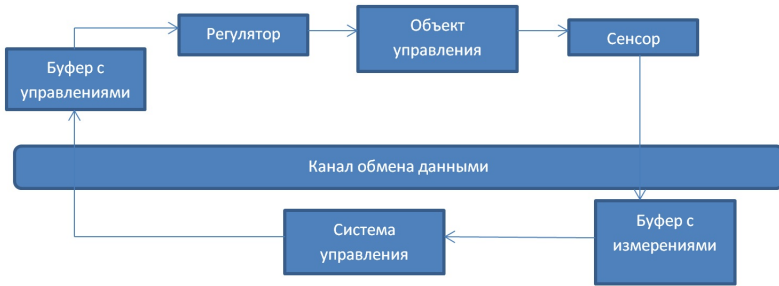


Рис. 2. Сетевая система управления

- 3) если потери пакета не произошло, принятый пакет записывается в буфер; в случае, если произошла потеря, берутся предсказанные для этого шага измерения из буфера;
- 4) строится наблюдатель;
- 5) формируется пакет из оценки состояния на текущий момент и предсказанные состояния на несколько шагов вперед;
- 6) пакет пересылается объекту управления;
- 7) на стороне объекта управления, если не произошло потери пакета, полученный пакет записывается в буфер;
- 8) берется управление либо с текущего момента, либо предсказания с предыдущих шагов.

2. Уравнения системы

Рассмотрим линейную систему с разностным уравнением следующего вида:

$$(1) \quad \begin{aligned} x_{k+1} &= Ax_k + Bu_k, \\ y_k &= Cx_k, \end{aligned}$$

где x_{k+1} – n -мерный вектор состояния перехода; x_k – n -мерный вектор исходного состояния; u_k – m -мерный вектор управления; y_k – l -мерный вектор измерений; k – дискретное время, выраженное в числе интервалов дискретности длительности Δt ; матрицы $A \in \mathbb{R}^{n \times n}$ и $B \in \mathbb{R}^{n \times m}$ – матрицы перехода вектора состояния и усиления вектора управления соответственно.

Как и ранее [1], введем гипотезу о возможности разделения и будем искать управление и оценку вектора состояния по отдельности. Полезным свойством использования прогнозирующего является то, что нет необходимости учитывать переключения системы при построении стабилизирующего управления $u_k = -G\hat{x}_k$, так как даже при потери пакета данных можно использовать сохраненную в буфере информацию с предыдущих шагов. В данном случае будем использовать метод функций Ляпунова и линейные матричные неравенства для нахождения матрицы усиления обратной связи G . В случае отсутствия переключений формулы являются тривиальными и поэтому не здесь приводятся.

Замечание 1. Обратим внимание на то, что в отличие от предыдущей работы [1], где система меняла свои структурные состояния, гипотеза о возможности разделения системы может быть легко доказана. Для этого рассмотрим систему (1) и наблюдатель, записанный для этой системы:

$$(2) \quad \begin{aligned} x_{k+1} &= Ax_k - BG\hat{x}_k, \\ \hat{x}_{k+1} &= Ax_k - BG\hat{x}_k + K(y_k - C\hat{x}_k). \end{aligned}$$

Можно показать, что, перейдя во втором уравнении от вектора оценки к вектору ошибки $\tilde{x} = x - \hat{x}$, система (2) запишется в следующем виде:

$$(3) \quad \begin{aligned} x_{k+1} &= (A - BG)x_k + BG\tilde{x}_k, \\ \tilde{x}_{k+1} &= (A - KC)\tilde{x}_k, \end{aligned}$$

или в матричном виде:

$$(4) \quad \begin{bmatrix} x_{k+1} \\ \tilde{x}_{k+1} \end{bmatrix} = \begin{bmatrix} A - BG & BG \\ 0 & A - KC \end{bmatrix} \begin{bmatrix} x_k \\ \tilde{x}_k \end{bmatrix}.$$

Видно, что система (4) имеет блочную верхнетреугольную матрицу, собственные значения которой равны собственным значениям блочно-диагональной матрицы (или собственным значениям разделенной системы).

Итак, основная задача теперь состоит в том, чтобы обладать оценкой вектора состояния для любого дискретного момента времени k , в независимости от того, замкнут ли сетевой канал обмена данными. Для этого, опираясь на идеи прогнозирующего управления, сформируем расширенный вектор состояния:

$$(5) \quad \begin{aligned} x_{k+1} &= Ax_k + Bu_k, \\ x_{k+2} - Ax_{k+1} &= Bu_{k+1}, \\ x_{k+N+1} - Ax_{k+N} &= Bu_{k+N}; \end{aligned}$$

$$(6) \quad \begin{aligned} y_k &= Cx_k, \\ y_{k+1} &= Cx_{k+1}, \\ y_{k+2} &= Cx_{k+2}, \\ &\vdots \\ y_{k+N} &= Cx_{k+N}. \end{aligned}$$

Здесь N – размер буфера (горизонта событий).

Перепишем систему (5) в следующем виде:

$$(7) \quad \begin{bmatrix} I & 0 & \dots & 0 & 0 \\ -A & I & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & -A & I \end{bmatrix} \begin{bmatrix} x_{k+1} \\ x_{k+2} \\ x_{k+3} \\ \vdots \\ x_{k+N+1} \end{bmatrix} =$$

$$\begin{bmatrix} A & 0 & 0 & \dots & 0 \\ 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 0 \end{bmatrix} \begin{bmatrix} x_k \\ x_{k+1} \\ x_{k+2} \\ \vdots \\ x_{k+N} \end{bmatrix} + \begin{bmatrix} B & 0 & 0 & \dots & 0 \\ 0 & B & 0 & \dots & 0 \\ 0 & 0 & B & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & B \end{bmatrix} \begin{bmatrix} u_k \\ u_{k+1} \\ u_{k+2} \\ \vdots \\ u_{k+N} \end{bmatrix},$$

$$(8) \quad \begin{bmatrix} y_k \\ y_{k+1} \\ y_{k+2} \\ \vdots \\ y_{k+N} \end{bmatrix} = \begin{bmatrix} C & 0 & 0 & \dots & 0 \\ 0 & C & 0 & \dots & 0 \\ 0 & 0 & C & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & C \end{bmatrix} \begin{bmatrix} x_k \\ x_{k+1} \\ x_{k+2} \\ \vdots \\ x_{k+N} \end{bmatrix}.$$

Введем расширенные вектора состояния и измерений $X = [x_k \ x_{k+1} \ \dots \ x_{k+N}]$, $Y = [y_k \ y_{k+1} \ \dots \ y_{k+N}]$, тогда система (7) запишется в следующем виде:

$$(9) \quad \begin{aligned} A_+ X_{k+1} &= A_0 X_k + B_0 u_k, \\ Y_k &= C_0 X_k. \end{aligned}$$

Обратим внимание, что в векторе измерений Y_k только измерения y_k являются действительным выходом системы 1, остальной набор y_{k+1}, \dots, y_{k+N} – это предсказанные измерения, которые могут быть сформированы из оценки расширенного вектора состояния, например.

В уравнениях (5)–(9) отсутствуют внешние возмущения, хотя в системе 1 допустимо их появление. Объяснение здесь следующее: при прогнозировании будущих состояний и измерений мы исходим из того предположения, что в исходной системе шумы, если они существуют, имеют нулевое математическое ожидание. В нашем случае эти шумы будут скомпенсированы при построении оценки вектора состояния с помощью использования фильтра Калмана.

Для определения состояния системы (9) будем использовать уравнения линейного фильтра Калмана:

$$(10) \quad \hat{X}_{k+1} = A_+^{-1} A_0 \hat{X}_k + A_+^{-1} B U_k + K(Y - C_0 \hat{X}_k).$$

В случае потери пакета данных матрица C_0 не меняется.

Замечание 2. В случае потери пакета данных логично бы

выглядело использование матрицы C_0 следующего вида:

$$(11) \quad Y = \begin{bmatrix} C & 0 & 0 & \dots & 0 \\ 0 & C & 0 & \dots & 0 \\ 0 & 0 & C & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 0 \end{bmatrix},$$

где отсутствующий пакет заменен нулями. Однако численные эксперименты показали, что такая замена ухудшает характеристики процесса сходимости фильтра Калмана, оригинальная версия которого чувствительна к сменам структурных состояний системы.

Неизменная же матрица C_0 говорит о том, что вместо потерянных измерений используются не нулевые данные, а последние полученные измерения.

Отметим здесь, что внутри горизонта предсказаний прогнозируемые состояния и измерения выбираются для каждой системы индивидуально. Они могут зависеть от таких факторов, как динамика системы и уровень внешнего воздействия. Еще раз напомним здесь основную идею прогнозирующего управления: внутри горизонта предсказаний модель поведения системы выбирается исходя из желаемого поведения системы, и поэтому может варьироваться в довольно широких пределах.

3. Пример

В качестве примера рассмотрим задачу построения стабилизирующего управления полетом квадрокоптера. Линейная система, характеризующая малые отклонения от положения равновесия по каналу угла крена и нулевого положения в пространстве, была взята из [14].

Для непрерывного времени система может быть записана следующим образом:

$$(12) \quad \ddot{\phi} = \frac{1}{I}\Gamma,$$

$$(13) \quad \ddot{x} = -g\phi + \frac{1}{Lm}\Gamma + \frac{1}{m}d_t.$$

Здесь m – масса квадрокоптера; I – момент инерции относительно оси X ; L – расстояние от плоскости винтов до двигателя; d_t – случайное воздействие, которое может характеризовать силу ветра, например.

Введем новый вектор неизвестных $X = [\phi \ \dot{\phi} \ x \ \dot{x}]$ и положим в качестве измеряемых величин $\dot{\phi}$ и x .

Для внесения возмущений в систему будем использовать функцию d_t следующего вида (см. рис. 3).

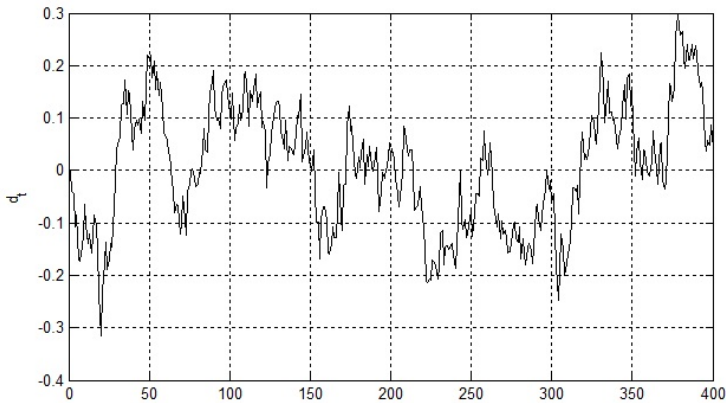


Рис. 3. Случайное воздействие

В качестве параметров системы (12) выберем следующие значения: $m = 6$ кг, $L = 200$ мм, $I = 0,24$ кг · м².

Тогда матрицы дискретной системы примут следующий вид:

$$A = \begin{bmatrix} 1,0000 & 0,0200 & 0 & 0 \\ 0 & 1,0000 & 0 & 0 \\ -0,0020 & -0,0000 & 1,0000 & 0,0200 \\ -0,1962 & -0,0020 & 0 & 1,0000 \end{bmatrix},$$

$$B = \begin{bmatrix} 0,0008 \\ 0,0833 \\ 0,0002 \\ 0,0166 \end{bmatrix}.$$

В работе [14] показано, что при выбранных параметрах система (12) является неустойчивой.

При построении системы выберем размер буфера $N = 5$. В общем случае размер буфера должен выбираться исходя из прогнозируемого количества последовательно потерянных пакетов данных и возможностей вычислительной системы. В случае превышения прогнозируемого числа потерянных пакетов вероятен уход системы со стабилизированной траектории.

Приведем траектории стабилизированной системы (см. рис. 4).

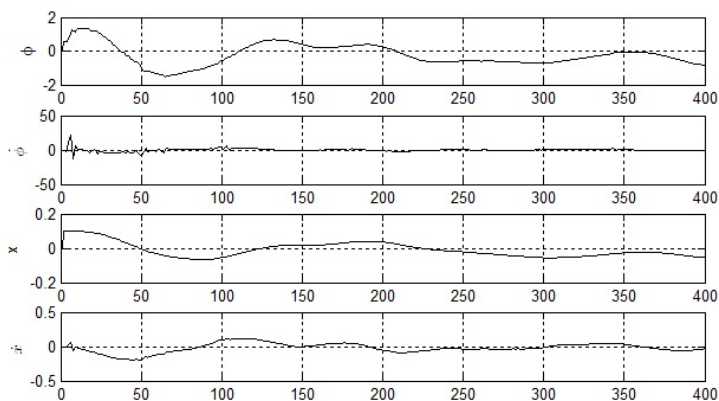


Рис. 4. Траектории стабилизированной системы

Кроме того, приведем распределение количества потерянных пакетов данных (см. рис. 5).

Видно, что система стабилизирована, несмотря на большое количество потерянных пакетов данных.

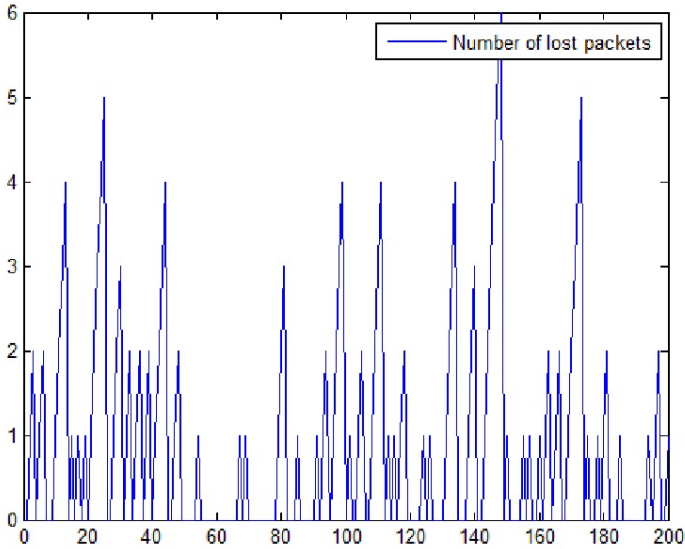


Рис. 5. Количество потерянных пакетов

Литература

1. ЖУЧКОВ Р.Н., ПАКШИН П.В. *Стабилизирующее сетевое управление линейными дискретными системами в условиях потери пакетов данных* // Управление большими системами. – 2011. – №33. – С. 113–126.
2. BEMPORAD A., MORARI M. *Control of systems integrating logic, dynamics, and constraints* // Automatica. – 1999. – Vol. 35. – P. 407–427.
3. BERBER R. *Methods of Model Based Process Control* // NATO ASI Series E: Applied Sciences. –Vol. 293.– Kluwer Academic Publications, Dordrecht, 1995. – 826 p.
4. BITMEAD R.R. *Adaptive Optimal Control. The Thinking Man's GPC* // International Series in Systems and Control

- Engineering, 1990. – 820 p.
5. CAMACHO E.F., BORDONS C. *Model Predictive Control in the Process Industry* // In: *Advances in Industrial Control*. – Springer Verlag, 1995. – 239 p.
 6. CLARKE D.W., MOHTADI C., TUFFS P.S. *Generalized predictive control - I. The basic algorithm* // *Automatica*. – 1987. – Vol. 23. – P. 137–148.
 7. CLARKE D.W., MOHTADI C., TUFFS P.S. *Generalized predictive control. - II. Extensions and interpretations* // *Automatica*. – 1987. – Vol. 23. – P. 149–160.
 8. CLARKE D.W. *Advances in Model-Based Predictive Control*. – Oxford University Press, 1994. – 552 p.
 9. CUTLER C.R., RAMAKER B.L. *Dynamic matrix control – A computer control algorithm* // AIChE 86th National Meeting, Houston, 1979.
 10. CUTLER C.R., RAMAKER B.L. *Dynamic matrix control – A computer control algorithm* // Joint Automatic Control Conf., San Francisco, 1980.
 11. DING B. *Stabilization of linear systems over networks with bounded packet loss and its use in synthesizing model predictive control* // In: *Proceedings of 8th, International Conference on Control and Automation*. 2010. – P. 2258–2263.
 12. GARCIA C.E., PRETT D.M., MORARI M. *Model predictive control: Theory and practice – a survey* // *Automatica*. – 1989. – Vol. 25. – P. 335–348.
 13. GRUNE L., PANNEK J. AND WORTHMANN K. *A networked unconstrained nonlinear mpc scheme* // In: *Proc. European Control Conference*. – 2009. – P. 91–96.
 14. HUA MINH-DUC, HAMEL T., MORIN P. *Introduction to Feedback Control of Underactuated VTOL Vehicles: A Review of Basic Control Design Ideas and Principles* // *Control Systems, IEEE*. – 2013. – Vol. 33, No. 1. – P. 61–75.
 15. LI Z.J., SUN D.H., SHI Y.T. ET AL. *A stabilizing model predictive control for network control system with data packet*

- dropout* // Journal of Control Theory and Application. – 2009. – Vol. 7. – P. 281–284.
16. MARTIN SANCHEZ J.M., RODELLAR J. *Adaptive Predictive Control* // International Series in Systems and Control Engineering, 1996. – P. 263–283.
 17. BEMPORAD A., MORARI M. *Robust Model Predictive Control: A Survey* // Robustness in Identification and Control. – 1999. – Vol. 245. – P. 207–226.
 18. NGUYEN Q.T., VESELY V. *Design of Robust Networked Predictive Control Systems with Packet Loss* // Preprints of 4th IFAC Nonlinear Model Predictive Control Conference International Federation of Automatic Control, 2012. – P. 362–357.
 19. RICHALET J., RAULT A., TESTUD J.L. AND OTHERS *Model predictive heuristic control: applications to industrial processes* // Automatica. – 1978. – Vol. 14. – P. 413–428.
 20. SOETERBOEK R. *Predictive Control - A United Approach* // International Series in Systems and Control Engineering, 1992. – 300 p.

PREDICTIVE CONTROL APPROACH IN NETWORKED CONTROL SYSTEMS

Roman Zhuchkov, graduate student (Arzamas Polytechnic Institute of R.E. Alekseev Nizhny Novgorod State Technical University, 19, Kalinina Street, Arzamas, 607227, Russia, (roman_jkv@mail.ru)).

Abstract: We consider a stabilization problem in a networked control system with packets dropouts and build dynamic feedback control using system outputs. Predictive control approach is used to obtain estimates of plant state at each time moment k , which makes unnecessary taking into account plant state switching when a packet dropout occurs.

Keywords: predictive control, Kalman filter, networked control systems, linear matrix inequalities, linear discrete systems.

Статья представлена к публикации членом редакционной коллегии А. Л. Фрадковым

Поступила в редакцию 22.06.2013.

Опубликована 30.09.2013.

УДК: 656.1.347.453

ББК 3.32.965.32.965.9

НЕЗАВИСИМАЯ МНОГОВАРИАНТНАЯ ЭКСПЕРТИЗА В ЗАДАЧЕ УПРАВЛЕНИЯ МЕЖРЕГИОНАЛЬНЫМИ АВТОБУСНЫМИ ПЕРЕВОЗКАМИ¹

Ахохов А. А.², Блудян Н. О.³

*(Московский автомобильно-дорожный государственный
технический университет, Москва)*

Дорофеюк Ю. А.⁴, Чернявский А. Л.⁵

*(ФГБУН Институт проблем управления
им. В.А.Трапезникова РАН, Москва)*

Предложен новый подход к разработке схемы управления межрегиональными автобусными перевозками, базирующийся на созданном авторами методе независимой многовариантной экспертизы (НМВЭ). Отличительная особенность НМВЭ состоит в том, что выявляются и разрабатываются варианты решения независимых проблем, покрывающих исходную задачу. Эффективность метода подтверждена результатами его применения к конкретной прикладной задаче.

¹ Работа выполнена при частичной финансовой поддержке РФФИ, проекты 11-07-00735, 11-07-00178, 12-07-00540, 13-07-00992.

² Али Асламбекович Ахохов, аспирант (ar@madi.ru).

³ Норайр Оганесович Блудян, доктор технических наук, профессор (ar@madi.ru).

⁴ Юлия Александровна Дорофеюк, кандидат технических наук, старший научный сотрудник (dorofeyuk_julia@mail.ru).

⁵ Александр Леонидович Чернявский, кандидат технических наук, доцент (aschern@ipu.ru).

Ключевые слова: межрегиональные автобусные перевозки, организационное управление, регулирование рынка, независимая многовариантная экспертиза.

1. Введение

В работе [2] была поставлена задача организации и управления рынком межрегиональных автобусных перевозок, там же отмечалось, что одним из наиболее проблемных направлений межрегиональных пассажирских автоперевозок являются перевозки между Северо-Кавказским федеральным округом (СКФО) и Москвой. В работе [1] было показано, что проблема регулирования рынка межрегиональных автобусных перевозок плохо поддается формализации, требует учёта множества факторов, не отражаемых в статистической отчетности, а также согласования противоречивых интересов сторон, участвующих в процессе пассажирских перевозок. При решении такого рода проблем невозможно обойтись без экспертов, т.е. специалистов, по роду своей работы хорошо знакомых с теми или иными аспектами исследуемой проблемы. Однако классические методы экспертизы в данном случае неприменимы. В [1] для решения поставленной задачи была разработана модификация метода коллективной многовариантной экспертизы [3, 4, 7] – **метод независимой многовариантной экспертизы (НМВЭ)**. Такая экспертиза предназначена для решения межведомственных проблем, требующих учёта интересов значительных групп людей, причём интересы разных групп, как правило, не совпадают, а иногда противоречат друг другу.

Ради справедливости необходимо отметить, что метод НМВЭ помимо существенных преимуществ по отношению к классическим методам коллективной экспертизы [3, 6], имеет один недостаток – значимо более высокие требования к квалификации и опыту консалтинговой группы, организующей и проводящей экспертизу. Как следствие, существенно возрастают нагрузки на консультантов и ответственность за качество проводимой экспертизы.

Поскольку настоящая работа в основном посвящена результатам проведенной экспертизы по решению конкретной прикладной задачи – формирование системы управления межрегиональными автобусными перевозками между Северо-Кавказским федеральным округом и Москвой, то основные теоретические и процедурно-алгоритмические аспекты метода НМВЭ далее изложены конспективно, чтобы читатель понимал основные положения и особенности реализации метода НМВЭ. Подробное описание метода содержится в [1], все заинтересованные в детальном ознакомлении с таким описанием могут получить его в этой публикации.

2. Основные этапы НМВЭ

В соответствии с предложенным в [1] методом, процедура НМВЭ разбивается на следующие шесть этапов:

1. На первом этапе составляется предварительный список специалистов и участников рынка, которые потенциально могут быть экспертами. Желательно, чтобы в этом списке по возможности были представлены все основные заинтересованные стороны. На этом этапе активно используются процедуры типа «снежный ком» [6].

2. На втором этапе в процессе интервью с потенциальными экспертами из предварительного списка формируется список из n относительно независимых проблем $C_{pr} = \{PR_1, \dots, PR_n\}$, где PR_i – формальное описание i -й проблемы. Такие формальные описания должны удовлетворять следующим условиям. Во-первых, эти проблемы в совокупности покрывают общую проблему экспертизы PR_{com} (в нашем случае это разработка системы управления межрегиональными автобусными перевозками между СКФО и Москвой). Во-вторых, сами проблемы должны быть **достаточно независимы**. В [1] показано, что это означает в теоретико-множественных терминах. На этом же этапе из общего списка C_{pr} выделяется подсписок проблем $C_{pr}^{nk} = \{PR_1^{nk}, \dots, PR_{n_k}^{nk}\}$, по способам решения которых, имеются

различные мнения (нет консенсуса), где PR_i^{nk} – формальное описание i -й проблемы, для которой нет консенсуса, а n_{nk} – общее число таких проблем. При этом при формировании этого подписка консультанты оценивают качество и убедительность аргументации потенциальных экспертов.

3. На третьем этапе производится оценка компетентности потенциальных экспертов. Алгоритм оценки компетентности, разработанный в рамках НМВЭ, существенно отличается от процедуры оценки условной компетентности, используемой в классических схемах коллективной многовариантной экспертизы [3, 4, 7]. В нашем случае необходимо оценивать компетентность эксперта не в целом по исследуемой задаче, а по каждой из n относительно независимых проблем из списка C_{pr} , и в первую очередь – по проблемам из подписка C_{pr}^{nk} (по способам решения которых нет консенсуса). В [1] приведено описание предложенного итерационного алгоритма оценки компетентности. Информация, используемая алгоритмом, формируется из результатов заполнения трёх типов анкет для j -го (оцениваемого) эксперта. На $(t + 1)$ -й итерации компетентность j -го эксперта по проблеме PR_i оценивается как средневзвешенная сумма оценок этого эксперта другими экспертами и консультантами:

$$(1) \quad u_j^{t+1}(PR_i) = \frac{1}{\sum_{s=1}^{l_1+l_2+1} v_s^t(PR_i)} \sum_{s=1}^{l_1+l_2+1} v_s^t(PR_i) k_j^{(s)}(PR_i), \quad j = 1, \dots, l,$$

где l_1 – число экспертов, оценивавших j -го эксперта по проблеме PR_i ; l_2 – число консультантов, оценивавших j -го эксперта по проблеме PR_i ; 1 в $(l_1 + l_2 + 1)$ соответствует самооценке j -го эксперта по проблеме PR_i ; $k_j^{(s)}(PR_i)$ – оценка j -го эксперта s -м экспертом по проблеме PR_i , а $v_s^t(PR_i)$ – веса, определяемые на t -й итерации уравнением

$$(2) \quad v_s^t(PR_i) = \begin{cases} u_s^t(PR_i), & \text{если эксперт компетентен для } PR_i, \\ 0, & \text{если эксперт некомпетентен для } PR_i. \end{cases}$$

Для каждой из оценок $u_j^{t+1}(PR_i)$ подсчитывается нижняя граница доверительного интервала $\Delta_j^{t+1,(L)}$. Затем проверяется гипотеза, что интервал (a, b) с заданным уровнем доверительной вероятности (в работе выбиралось значение 0,9) включает математическое ожидание случайной величины $u_j^{t+1}(PR_i)$, где a и b – заранее выбранные константы.

Выбор констант a и b в значительной степени осуществляется экспертным путём и требует определённого опыта при решении конкретных задач описанного типа. В нашем случае величина a выбирается из требования минимально допустимого уровня компетентности потенциального эксперта (в шкале $[0, 1]$). Очевидно, что выбор порога a в значительной степени определяется «потенциалом» набора экспертов, из которого выбираются те, кто будет участвовать в экспертизе. В работе значение a для различных проблем PR_i выбиралось в диапазоне 0,5–0,7, в зависимости от «сложности» самой проблемы. Значение константы b напрямую зависит от размера выборки и дисперсии оценок компетентности для тестируемого набора экспертов. В работе $b = (0,8 - 0,9)a$ для «обычных» выборок и $b = 0,7a$ – для малых выборок (с экспертной поправкой, зависящей от дисперсии оценок, полученных в результате тестирования) [1].

Если гипотеза выполняется, то j -й эксперт считается компетентным по проблеме PR_i , и некомпетентным – в противном случае. В результате получаем распределение экспертов по группам компетенции по проблеме PR_i (компетентен – некомпетентен), а также с помощью (1) и (2) новые значения весовых коэффициентов $v_s^{t+1}(PR_i)$. Алгоритм прекращает работу на $(N + 1)$ -й итерации, при условии, что для всех s справедливо равенство $v_s^N(PR_i) = v_s^{N+1}(PR_i)$. На этой итерации определяются окончательные оценки компетентности j -го эксперта по проблеме PR_i для всех i и j .

На первом (начальном) шаге алгоритма производится простое усреднение (без взвешивания) оценок $k_j^{(s)}(PR_i)$, поскольку

только на этом шаге появятся первые оценки уровня компетентности экспертов, которые будут использованы при расчёте весов $v_s^t(PR_t)$ для второго шага ($t = 2$).

4. На четвёртом этапе происходит формирование экспертных комиссий. Для каждой проблемы, по схеме решения которой среди компетентных (для этой проблемы) экспертов достигнут консенсус, создаётся единственная экспертная комиссия, состоящая из таких экспертов. В задачу каждой такой комиссии входит подготовка для ЛПР развёрнутых предложений по схеме, бизнес-плану, правовой и нормативной базе и пр., необходимых для решения этой проблемы.

Для разработки решения каждой из проблем PR_i^{nc} , по которым имеются разногласия (не достигнут консенсус), формируются несколько экспертных комиссий, в каждую из которых входят компетентные эксперты, имеющие приблизительно одинаковые мнения по решению этой конкретной проблемы (а не по совокупности проблем, как в [3, 4, 7]). В [1] был разработан специальный алгоритм формирования таких экспертных комиссий, использующий экспертно-классификационный подход к анализу и структуризации многомерных данных [5]. В результате применения этого алгоритма к имеющимся данным, для каждой проблемы PR_i^{nc} будет получено распределение компетентных (по этой проблеме) экспертов по r_{opt} экспертным комиссиям, где r_{opt} – число таких комиссий, полученное с помощью экспертно-компьютерной процедуры выбора оптимального числа классов в задачах кластерного анализа [5].

Следует подчеркнуть, что процедура формирования экспертных комиссий, в каждую из которых входят «единомышленники» по экспертируемой условно-независимой проблеме PR_i^{nc} , весьма устойчива к возможному влиянию мнения большинства (даже подавляющего большинства – «мейнстрима»), что свойственно многим процедурам коллективной экспертизы [3, 4]. Дело в том, что распределение экспертов по комиссиям происходит исключительно на основании информации о близости точек зрения экспертов на способы решения проблемы PR_i^{nc} ,

при условии, что все они являются компетентными по этой проблеме.

5. На этом этапе и происходит собственно работа сформированных экспертных комиссий. Для проблем PR_i^{nc} , по которым сформировано несколько экспертных комиссий, как и в [4, 7], организуется перекрёстная экспертиза вариантов решения: каждая комиссия разрабатывает свой вариант предложений по решению проблемы (вариант №1) и высказывает критические замечания о вариантах, разработанных другими комиссиями. На следующем шаге каждая комиссия анализирует полученные от других комиссий замечания на вариант №1 и формирует вариант №2, учитывающий эти замечания и т.д. Перекрёстная экспертиза заканчивается на шаге N , на котором для каждой комиссии комплекты: варианты № N и № $(N - 1)$ плюс замечания на них других экспертных комиссий полностью совпадают. Результатом обсуждения является либо некий согласованный вариант, либо перечень вариантов с указанием преимуществ и недостатков каждого из них. Эксперты участвуют в обсуждении заочно, получая от консультантов полную информацию о мнениях других экспертов и результаты их статистической обработки.

Гипотетически возможны случаи, когда выявленный набор проблем не является полностью независимым, т.е. выбор варианта решения некоторой проблемы будет зависеть от того, какие решения будут приняты по другим проблемам. В таких случаях процедура перекрестной экспертизы становится многоэтапной. А именно, на каждом этапе перекрестной экспертизы экспертные комиссии по i -й условно-независимой проблеме получают результаты этого же этапа перекрестной экспертизы по j -й проблеме (согласованный вариант либо перечень вариантов с указанием преимуществ и недостатков каждого из них). На следующем этапе проводится следующий цикл перекрестной экспертизы для экспертных комиссий по i -й условно-независимой проблеме с учётом полученных результатов. Этап является заключительным, если его резуль-

таты для всех комиссий по всем взаимосвязанным проблемам являются идентичными.

6. По итоговым результатам обсуждения консалтинговая группа разрабатывает проект решения проблемы в целом с указанием преимуществ и недостатков предлагаемого решения. В [4] окончательное решение предлагается оставить за руководством организации (ЛПР). В нашем случае функции ЛПР может выполнять представительство (филиал) Агентства автомобильного транспорта Министерства транспорта РФ в СКФО.

3. Основные проблемы, подлежащие обсуждению

В [1] на первых двух этапах экспертизы в качестве предварительных результатов были сформулированы следующие проблемы, подлежащие обсуждению:

1. Разработать порядок формирования, утверждения, функционирования и ликвидации межрегиональных маршрутов регулярных перевозок, а также соответствующей нормативной базы. В частности, необходимо ответить на вопросы:

1.1. Кто может выступать в роли перевозчика (юридическое лицо, индивидуальный предприниматель и т.п.).

1.2. Каким требованиям должен удовлетворять перевозчик (требования к маршрутам, транспортным средствам, расписанию перевозок, безопасности перевозок и т.п.).

1.3. Порядок формирования тарифов на перевозки.

1.4. Порядок пользования услугами автовокзалов, автостанций и парковочных стоянок.

2. Решить проблему конечных пунктов следования междугородных автобусов, прибывающих из СКФО. Здесь возможны следующие варианты:

2.1. Сохранить существующую ситуацию, когда конечными пунктами являются рынки и торговые центры на территории Москвы (вариант, больше всего устраивающий пассажиров).

2.2. Установить в качестве конечных пунктов автовокзалы и парковочные зоны вблизи конечных станций метро (вариант, больше всего устраивающий городские службы г. Москва).

2.3. Компромиссные варианты, ограничивающие набор конечных пунктов, время стоянки и т.п.

3. Решить проблему повышения антитеррористической безопасности. Здесь возможны следующие варианты:

3.1. Установить ограничения на заказные междугородные рейсы (например, на множество конечных пунктов) с целью уменьшить долю нелегальных рейсов.

3.2. Установить для регулярных рейсов порядок продажи билетов с обязательным предъявлением паспорта (как это делается на авиа- и железнодорожном транспорте).

3.3. Установить ту или иную форму контроля перевозимых грузов.

Очевидно, что любые дополнительные ограничения ухудшают рыночную ситуацию, поэтому здесь необходимо найти компромисс между степенью безопасности и интересами участников рынка.

4. Следует ли создать на уровне СКФО организацию для централизованного решения проблем регулирования рынка автобусных перевозок? Какие функции было бы целесообразно поручить этой организации:

4.1. Ведение переговоров с администрацией г. Москвы по отстаиванию интересов участников рынка межрегиональных автобусных перевозок со стороны СКФО.

4.2. Централизованную продажу билетов на межрегиональные рейсы.

4.3. Развитие инфраструктуры межрегиональных автобусных перевозок.

5. В случае положительного решения по вопросу 4, какую правовую форму должна иметь такая организация.

6. Какие правовые акты, и на каком уровне должны быть приняты для реализации решений по пунктам 1–5.

В настоящей статье рассматриваются результаты, полученные на последующих этапах экспертизы.

4. Результаты экспертизы

На первом этапе экспертизы был сформирован предварительный список из более сотни кандидатов в эксперты из числа участников рынка и представителей заинтересованных организаций (см. таблицу 1).

Заметим, что относительное количество экспертов, представляющих участников рынка и разные организации, не имеет особого значения, поскольку решения принимаются не голосованием, а исходя из предлагаемой экспертами аргументации, причем эксперты, стоящие на разных позициях, не контактируют друг с другом, обмениваясь мнениями только через консультантов.

Таблица 1. Представительство участников рынка и организаций в экспертизе

Участники рынка и организации	Кол-во экспертов
Перевозчики (индивидуальные предприниматели)	16
Перевозчики (юридические лица)	14
Специалисты по автоперевозкам из НИИ и ВУЗов	16
Представители администрации и транспортной инфраструктуры субъектов СКФО	25
Представители Правительства и организаций транспортной инфраструктуры г. Москвы	12
Представители Министерства транспорта и автотранспортных организаций Московской области	6
Агентство автотранспорта Минтранса РФ	3
Российский автотранспортный союз	2
Московский транспортный союз	4
Некоммерческое партнерство «Единая транспортная система «Автобусные линии страны»	3
Некоммерческое партнерство «Развитие автовокзалов страны»	3

В соответствии с методикой экспертизы эксперты последовательно обсуждали приведенные выше проблемы 1–6. При наличии разных мнений проводилась перекрестная экспертиза: консультанты сообщали мнения одних экспертов другим (их оппонентам), возражения вторых – первым и т.д., до тех пор, пока эксперты либо приходили к общему мнению, либо уже не могли привести новые аргументы.

Обсуждение дало следующие результаты.

1. По проблеме 1 возникли следующие разногласия:

1.1. Эксперты из ГКУ «Организатор перевозок» и один эксперт из Агентства автомобильного транспорта предлагали не регистрировать в качестве регулярных перевозчиков простые товарищества, мотивируя это тем, что в таком случае имеется риск массового образования мнимых товариществ, которые будут использовать «серые схемы», вплоть до перепродажи маршрутов. Однако другие эксперты, лучше знающие реальную ситуацию, убедительно показали, что при небольших объемах рынка и фактическом доминировании на нём индивидуальных предпринимателей такой опасности нет.

1.2. Эксперты из Агентства автомобильного транспорта предлагали отказывать в регистрации маршрута, если он дублирует ранее установленные межрегиональный маршруты и, кроме того, не позволяет существенным образом улучшить транспортное обслуживание населения, либо дополнительные объемы перевозок, предусмотренные таким маршрутом, могут быть выполнены перевозчиком, который осуществляет перевозки по ранее установленному маршруту. Однако они не смогли опровергнуть доводы оппонентов, указавших, что критерии «дублирования» невозможно точно определить; кроме того, подобные нормы ограничивают допуск на рынок новых перевозчиков, что влечет снижение уровня конкуренции и, как следствие, качества обслуживания пассажиров.

По остальным вопросам, касающимся технических требований к транспортным средствам, минимальной информации, предоставляемой перевозчиком в заявлении о регистрации маршрута, порядке и сроках рассмотрения таких заявлений, а

также порядка прекращения регистрации при нарушении перевозчиком установленных требований, разногласий между экспертами не было. В целях стимулирования рынка, право устанавливать тарифы, решено предоставить перевозчикам, с условием не увеличивать тарифы чаще, чем раз в год. Владелец автовокзала (автостанции) не вправе отказаться от заключения договора с перевозчиком, которому выдано свидетельство об осуществлении перевозок по межрегиональному маршруту, в состав которого включен автовокзал (автостанция). Тарифы за услуги автовокзала (автостанции) должны быть едиными для всех перевозчиков.

2. По проблеме 2 разногласия возникли по вопросу о конечных пунктах межрегиональных маршрутов в Москве. Эксперты из ГКУ «Организатор перевозок» требовали установить в качестве конечных пунктов автовокзалы и парковочные зоны вблизи конечных станций метро (ссылаясь на очень сложную транспортную обстановку в Москве). Этот вариант оказался неприемлемым для перевозчиков, основными пассажирами которых являются «челноки», везущие грузы на рынки и торговые центры. Помимо этого чисто «рыночного» довода они выдвинули следующий аргумент: поскольку грузы все равно должны быть доставлены в торговые точки, их децентрализованная доставка самими пассажирами еще более ухудшит транспортную обстановку, так как значительно растянет процесс во времени и в пространстве. Кроме того, анализ, проведенный специалистами из ГКУ «Организатор перевозок» по просьбе и с участием консультантов, показал, что у всех торговых точек на самом деле есть возможность кратковременной парковки автобуса для выгрузки товаров. В результате был принят вариант 2.2, устраивающий перевозчиков и пассажиров, с условием, что по окончании разгрузки автобус отгоняется в согласованную зону парковки, формирование таких зон, их обустройство и контроль функционирования находится в компетенции Департамента транспорта Правительства Москвы. Список таких парковок и условия их эксплуатации согласовываются с Агентством автомобильного транспорта Минтранса РФ.

3. По проблеме 3 разногласий не было: все эксперты придерживались мнения, что единственно приемлемый способ повысить антитеррористическую безопасность – установить для регулярных рейсов порядок продажи билетов с обязательным предъявлением паспорта (как это делается на авиа- и железнодорожном транспорте). Введение ограничений на множество конечных пунктов или дополнительных (не предусмотренных действующим законодательством) мер по контролю перевозимых грузов существенно ухудшит рыночную ситуацию, польза же от такого рода мер проблематична. Вместе с тем, учитывая, что решения, принятые по проблемам 1 и 2, максимально облегчают доступ на рынок, эксперты предложили для повышения антитеррористической безопасности ввести административную ответственность (в основном в виде штрафа):

- за посадку в автобус пассажира, следующего без именного билета;
- за размещение автобусов, используемых на межрегиональных маршрутах, в ночное время в неустановленных местах;
- за посадку пассажиров в неустановленных местах;
- за использование на межрегиональных маршрутах автобусов, на которые не выдана карта соответствующего маршрута или с иными характеристиками, чем это предусмотрено данной картой.

4. По проблемам 4 и 5 было принято решение отложить обсуждение до завершения реорганизации Министерства транспорта РФ. Когда реально начнет работу Агентство автомобильного транспорта РФ и его региональные представительства, будет большая ясность по вопросу о том, какие функции можно поручить таким представительствам и нужно ли создавать еще одну организацию для централизованного выполнения каких-либо вспомогательных функций по поддержке рынка межрегиональных автобусных перевозок. Тогда же можно будет вернуться к проблеме 6 (о необходимых правовых актах).

5. Заключение

Разработан новый метод коллективной многовариантной экспертизы (независимая многовариантная экспертиза (НМВЭ)), учитывающий особенности задачи управления межрегиональными автобусными перевозками. Метод НМВЭ имеет следующие значимые отличия от обычных методов коллективной многовариантной экспертизы:

- заочная форма работа экспертов, когда каждый эксперт получает информацию об оценках других экспертов из той же экспертной комиссии от консультантов; по этой же схеме работает процедура перекрёстной экспертизы;

- разработана новая процедура выявления относительно независимых проблем, связанных с исходной задачей; совокупное решение этих независимых проблем обеспечивает решение исходной задачи в целом;

- в рамках НМВЭ разработан новый алгоритм оценки компетентности экспертов, который использует процедуру «взвешивания» оценок каждого из группы экспертов, проводящих оценивание компетентности некоторого эксперта, с помощью коэффициента, зависящего от уровня его компетентности;

- разработана специальная процедура формирования экспертных комиссий, использующая высокоэффективные алгоритмы кластерного анализа и выбора оптимального числа классов.

Приведены результаты проверки эффективности разработанного метода НМВЭ на примере решения конкретной прикладной задачи – формирование системы управления межрегиональными автобусными перевозками между Северо-Кавказским федеральным округом и Москвой.

Литература

1. АХОХОВ А.А., БЛУДЯН Н.О., ДОРОФЕЮК Ю.А., ЧЕРНЯВСКИЙ А.Л. *Разработка системы управления межрегиональными автобусными перевозками на базе методов*

- коллективной многовариантной экспертизы // Проблемы управления. – 2013. – №5. – С. 29–38.
2. БЛУДЯН Н.О., ПАСЫНСКИЙ А.А., АХОХОВ А.А. *Комплексный проект организации межрегиональных автобусных перевозок (на примере направления «Москва – Северо-Кавказский федеральный округ»)* // Организация автомобильных перевозок пассажиров и грузов на современном этапе: Сборник научных трудов под ред. Н.О. Блудяна. – М.: МАДИ, 2011. – С. 26–34.
 3. ДОРОФЕЮК А.А., ЧЕРНЯВСКИЙ А.Л. *Консультативная работа по совершенствованию управления в организационных системах (методологические основы)* // Сб.: Методы и алгоритмы анализа эмпирических данных. – М.: ИПУ, 1988. – С. 5–11.
 4. ДОРОФЕЮК А.А., ЧЕРНЯВСКИЙ А.Л., ПОКРОВСКАЯ И.В. *Экспертные методы анализа и совершенствования систем управления* // Автоматика и телемеханика. – 2004. – №10. – С. 172–188.
 5. ДОРОФЕЮК Ю.А. *Комплекс алгоритмов экспертно-классификационного анализа для решения прикладных задач* // 4-я международная конференция по проблемам управления: Сборник трудов. – М.: ИПУ РАН, 2009. – С. 373–379.
 6. ПАНКОВА Л.А., ПЕТРОВСКИЙ А.М., ШНЕЙДЕРМАН М.В. *Организация экспертизы и анализ экспертной информации.* – М.: Наука, 1984. – 260 с.
 7. ПОКРОВСКАЯ И.В., ГОЛЬДОВСКАЯ М.Д., КИСЕЛЁВА Н.Е. *Методы многовариантной экспертизы в задачах поддержки принятия решений в социально-экономических системах управления* // Управление развитием крупномасштабных систем (MLSD'2012): Материалы Шестой международной конференции. Том I. – М.: ИПУ РАН, 2012. – С. 322–324.

INDEPENDENT MULTIVARIATE EXPERTISE IN TRANS-REGIONAL BUS SERVICE MANAGEMENT

Ali Ahohov, Moscow State Automobile and Road Technical University, Moscow, PhD student (ap@madi.ru).

Norair Bludyan, Moscow State Automobile and Road Technical University, Moscow, Doctor of Science, professor (ap@madi.ru).

Yulia Dorofeyuk, Institute of Control Sciences of RAS, Moscow, Cand.Sc., senior research assistant (dorofeyuk_julia@mail.ru).

Alexander Chernyavskiy, Institute of Control Sciences of RAS, Moscow, Cand.Sc., assistant professor, senior research assistant (achern@ipu.ru).

Abstract: We propose a new approach to a trans-regional bus service management scheme design. This approach is based on the original method of independent multivariate expertise (IMVE). A distinctive feature of IMVE is that experts identify and develop independent problems solutions covering the whole original task. Efficiency of the method is verified by results of its application to a concrete management problem.

Keywords: trans-regional bus transportation, organizational management, market regulation, independent multivariate expertise.

*Статья представлена к публикации
членом редакционной коллегии В.В. Клочковым*

Поступила в редакцию 24.07.2013.

Опубликована 30.11.2013.

УДК 519.876
ББК 74.6

ИМИТАЦИОННОЕ МОДЕЛИРОВАНИЕ СОЦИАЛЬНОГО ПАРТНЕРСТВА В СИСТЕМЕ ДОПОЛНИТЕЛЬНОГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ

Дьяченко В. К.¹, Тарасенко Л. В.², Угольницкий Г. А.³
(Южный федеральный университет, Ростов-на-Дону)

Предлагается теоретико-игровая модель социального партнерства в системе дополнительного профессионального образования. Описываются подходы к идентификации модели и ее исследованию на основе имитационного моделирования.

Ключевые слова: социальное партнерство, дополнительное профессиональное образование, имитационное моделирование, идентификация, динамические игры.

1. Введение

Проблема дополнительного профессионального образования (ДПО), несомненно, актуальна [1, 4, 9–11]. Система профессионального образования должна быть достаточно гибкой для того, чтобы готовить кадры, соответствующие требованиям рынка и способные при необходимости изменять свою специализацию. Одним из наиболее эффективных путей решения этой проблемы является развитие социального партнерства, позволяющего объединить и обеспечить совместное участие в подго-

¹ Владимир Константинович Дьяченко, магистр прикладной математики и информатики (v.k.diachenko@gmail.com).

² Лариса Викторовна Тарасенко, доктор социологических наук, профессор (socmodel@mail.ru).

³ Геннадий Анатольевич Угольницкий, доктор физико-математических наук, профессор (ougoln@mail.ru).

товке кадров всех связанных с этим субъектов: работодателей, образовательных учреждений и студентов.

Под социальным партнерством в сфере ДПО понимается особая система совместной деятельности между субъектами образовательного процесса, характеризующаяся доверием, общими целями и ценностями, обеспечивающая подготовку высококвалифицированных специалистов, конкурентоспособных и мобильных на рынке труда.

Будем рассматривать в качестве участников социального партнерства учебные заведения (и их преподавателей), студентов, работодателей (предприятия и организации). Это партнерство на практике проявляется в распространении образовательных договоров, расширении мест учебно-производственной практики, обеспечении участия предприятий в учебно-воспитательном процессе, практической подготовке специалистов, принятии выпускников на работу и т. д.

Результатом социально-партнерского взаимодействия является рост качества обучения студентов, в том числе количества студентов, получивших повышенный уровень квалификации и мотивацию к изучению новых технологий.

Социальное партнерство в образовании рассматривалось многими авторами, но моделирование этого процесса развито слабо [14, 16, 17]. Первые авторские разработки представлены в [12].

Представленная в данной работе модель позволяет провести исследование партнерства в системе ДПО относительно уровня профессиональной подготовки. Создана теоретико-игровая модель, её исследование проведено в режиме имитационного моделирования [2, 3, 6, 7, 8, 13]. В работе использовались статистические данные и результаты опросов относительно ДПО и качества подготовки специалистов [5, 11]. Описаны принципы определения параметров модели. С помощью компьютерной имитации проанализировано поведение модели для определенного набора сценариев, сделаны некоторые выводы и даны рекомендации.

2. Модель социального партнерства в системе ДПО

2.1. ОБЩЕЕ ОПИСАНИЕ МОДЕЛИ

В предлагаемой модели рассматривается социальное партнерство между следующими субъектами: работодателями, студентами и вузом.

Целевые функции субъектов:

$$J_P = \sum_{t=0}^T g_P(u_P(t), u_B(t), u_C(t), x(t)) \rightarrow \max, \quad u_P(t) \in U_P;$$

$$(1) \quad J_B = \sum_{t=0}^T g_B(u_P(t), u_B(t), u_C(t), x(t)) \rightarrow \max, \quad u_B(t) \in U_B;$$

$$J_C = \sum_{t=0}^T g_C(u_P(t), u_B(t), u_C(t), x(t)) \rightarrow \max, \quad u_C(t) \in U_C;$$

$N = \{P, B, C\}$ – множество субъектов управления, а именно: P – работодатель; B – ВУЗ; C – студент;

Уравнение динамики состояния модели:

$$(2) \quad x(t+1) = x(t) + f(x(t), u_P(t), u_B(t), u_C(t)), \quad x(0) = x_0.$$

Здесь $u_P(t)$, $u_B(t)$, $u_C(t)$ – стратегии поведения указанных субъектов, описывающие усилия, направленные на развитие системы социального партнерства. Для количественного измерения усилий под ними понимается финансирование данной системы субъектами. Таким образом, они получают возможность стимулировать её развитие с помощью ассигнования денежных средств из своего бюджета;

U_P, U_B, U_C – области допустимых стратегий поведения;

J_P, J_B, J_C – функционалы выигрыша субъектов;

g_P, g_B, g_C – текущие функции выигрыша субъектов.

$T = 4$ (период 5 лет).

Стратегии определяют долю годового бюджета, которую субъект ассигнует на ДПО:

$u_{Pi}(t)$ – доля годового бюджета, которую каждый опрошенный работодатель готов тратить на ДПО, $U_{Pi}=[0,1]$;

$u_{Ci}(t)$ – доля годового бюджета, которую каждый опрошенный слушатель готов тратить на ДПО, $U_{Ci}=[0,1]$;

$u_B(t)$ – доля годового бюджета, которую ВУЗ готов тратить на ДПО, $U_B = [0, 1]$.

Данная модель универсальна, причем в качестве субъектов рассматриваются множества, состоящие из различных элементов, статистические данные и данные опросов для которых известны. Таким образом, они являются одновременно субъектами управления и респондентами: $P = \{P_1, \dots, P_r\}$ определяет конечное множество учреждений; $C = \{C_1, \dots, C_s\}$ – конечное множество студентов (слушателей курсов ДПО); поскольку исследование основано на статистических данных Южного федерального университета, то рассматривается один ВУЗ: B .

Элементы определяются исходя из количества участвующих в социальном партнерстве и проходящих соответствующие анкетирования субъектов. Конкретные числовые данные о количестве субъектов определяются согласно данным опросов и собранной статистики. В модели рассматривается следующая ситуация: множество Работодатель представлено пятью организациями, а множество Студент соответствует группе из десяти человек.

Стратегии поведения субъектов в определенный момент времени вычисляются как среднее арифметическое стратегий элементов соответствующих множеств.

Стратегия поведения P :

$$(3) \quad u_P(t) = \frac{1}{r} \sum_{i=1}^r u_{P_i}(t); u_P(t) \in U_P, \text{ где } r = 5.$$

Стратегия поведения C :

$$(4) \quad u_C(t) = \frac{1}{s} \sum_{j=1}^s u_{C_j}(t); u_C(t) \in U_C, \text{ где } s = 10.$$

Стратегия поведения B :

$$(5) \quad u_B(t); u_B(t) \in U_B.$$

Увеличение мощности рассматриваемых множеств не повлечет принципиальных изменений результатов моделирования, однако будет иметь следующие технические последствия:

1) относительно стратегий управления – изменится общая стратегия множества субъектов, которая определяется как среднее арифметическое стратегий элементов;

2) относительно данных опросов – изменится начальное значение переменной состояния модели, поскольку она определяется на основе статистических данных, полученных для определенного набора субъектов.

Переменная состояния модели $x(t)$, в зависимости от идентификации конкретной модели, определяет количественный показатель, характеризующий определенное свойство системы. Далее будет рассмотрена идентификация модели относительно уровня профессиональной подготовки студентов;

f – функция изменения переменной состояния модели в зависимости от действий субъектов.

Для задания системной динамики используем логистическое уравнение (уравнение динамики численности) Ферхюльста. В качестве f берем:

$$(6) f(x(t), u_P(t), u_B(t), u_C(t)) = h(u_P(t), u_B(t), u_C(t))x(t)\left(1 - \frac{x(t)}{K}\right);$$

Таким образом, вместо постоянного коэффициента прироста в базовой модели Ферхюльста здесь используется функция h , зависящая от действий субъектов. Если $h > 0$, то уровень подготовки растет, иначе он падает (как и в базовой модели). В данной работе рассмотрен случай $h > 0$ (см. (7)); в дальнейшем предполагается учесть и вторую возможность;

K – максимально возможное в данных условиях значение переменной состояния модели;

h – функция прироста:

$$(7) h(u_P(t), u_B(t), u_C(t)) = \sum_{i=1}^3 a_i u_i(t); a_i \geq 0; \sum_{i=1}^3 a_i = 1; i = P, B, C;$$

a_i – относительные веса факторов влияния;

При оценке роста рассматриваем влияние стратегий субъектов, оценивая при этом соответствующие веса.

Так как максимальное значение суммарного влияния равно единице и социальные партнеры преследуют общие цели,

логично рассмотреть равные значения факторов весов. Но в данной системе более значимая роль возлагается на студентов, ведь именно они являются будущими специалистами и в первую очередь развитие системы ДПО сказывается на них. Исходя из этих соображений, значения весов выбраны следующим образом (см. таблицу 1):

Таблица 1. Относительные веса факторов влияния a_i

Вес фактора	Работодатель	ВУЗ	Студент
a_i	0,3	0,3	0,4

Согласно основной гипотезе исследования, более развитые социально-партнерские отношения соответствуют более высокому уровню подготовки. Так как социально-партнерские отношения зависят от усилий всех партнеров, то целесообразно исследовать два варианта параметризации функций выигрыша субъектов.

1) Эгоистический подход. Если говорить о текущем периоде времени, то естественно считать, что g_i убывает по u_i и возрастает по остальным аргументам («принцип безбилетника»). Этот вариант рассматривает эгоистический подход – экономия субъектом личных усилий.

Таким образом, возникает задача согласования частных (экономия усилий) и общих интересов в системе социального партнерства. В этом случае в качестве функций выигрыша мы берем:

$$(8) g_i(u_P(t), u_B(t), u_C(t), x(t)) = \frac{b_j u_j(t) + b_k u_k(t) + b_x x(t)}{1 + b_i u_i(t)}; i, j, k = P, B, C;$$

b_i – относительные веса;

2) Кооперативный подход. Такая параметризация описывает желаемое (идеальное) состояние более развитых отношений в системе социального партнерства, когда ее субъекты добровольно и осознанно вкладывают ресурсы в развитие партнерства. Этот вариант определяет кооперативный подход, где функции

выигрыша субъектов становятся возрастающими по всем аргументам:

(9) $g_i(u_P(t), u_B(t), u_C(t), x(t)) = b_P^i u_P(t) + b_B^i u_B(t) + b_C^i u_C(t) + b_x^i x(t)$, $i = P, B, C$; b_j^i – относительная значимость фактора для субъекта i ($i = P, B, C$; $j = P, B, C, x$).

Определение значений b_i и b_j^i предусматривает анализ взаимосвязей, отражающих социальные, экономические и другие связи между субъектами.

Отметим, что значениями функций g_i (а соответственно и общих функционалов выигрыша субъектов J_i , $i = P, B, C$) выступают некоторые абстрактные полезности, являющиеся результатом развития системы социального партнерства по тому или иному сценарию. Здесь следует сделать два замечания. Во-первых, в реальной жизни выигрыши субъектов по своей природе являются векторами, т.е. включают несколько показателей. Однако теория игр с векторными функциями выигрыша пока находится в зачаточном состоянии. Во-вторых, использование даже скалярных критериев выигрыша с ясным «физическим» смыслом (например, дохода) было бы, несомненно, более убедительным. Но и идентификация таких показателей представляет отдельную сложную задачу, поэтому пока мы ограничиваемся сравнительным анализом различных значений абстрактной полезности при разных вариантах развития социального партнерства.

2.2. ИДЕНТИФИКАЦИЯ МОДЕЛИ УПРАВЛЕНИЯ ДИНАМИКОЙ УРОВНЯ ПРОФЕССИОНАЛЬНОЙ ПОДГОТОВКИ

Будем считать, что переменная состояния модели $x(t)$ характеризует уровень профессиональной подготовки студентов.

При оценке начального уровня подготовки использовалась методика Дональда Киркпатрика [15]. Он описал модель оценки эффективности обучения, согласно которой существует четыре уровня оценки результатов обучения. Мы оцениваем эти показатели с помощью статистических данных анализа результатов анкетирования работодателей и студентов [5, 11]:

1) Реакция: насколько обучение и преподавание понравилось участникам. При оценке используем результаты опроса по оцениванию студентами качества преподавания различных дисциплин в Южном федеральном университете.

2) Усвоение: какие навыки, техники работы и умения были получены. Рассматриваем данные об успеваемости студентов. Этот показатель также зависит от наличия соответствующей информационной и материально-технической базы, позволяющей получить и усвоить новые знания и умения.

3) Поведение: как в результате обучения изменилось поведение, действия участников в рабочей обстановке, умеют ли обучаемые выполнять все предлагаемые задания и представлять результаты своего труда. Использовались результаты опросов, проведенных среди работодателей.

4) Результат: каковы результаты обучения, удовлетворенность профессиональностью выпускников. При оценке рассматривались данные опросов среди работодателей по теме «Уровень профессиональных знаний и навыков выпускников».

На данном этапе начальное значение уровня профессиональной подготовки оцениваем с помощью очевидной формулы

$$(10) x_0 = \frac{1}{4} \sum_{i=1}^4 x_i .$$

При задании x_i рассматриваются результаты опросов и статистические данные относительно оценки профессиональных характеристик молодых специалистов, оценки студентами качества преподавания в ЮФУ, удовлетворенности информационной и материально-технической базой образовательного процесса и другие. Данные соответствующих таблиц представлены в отчетах [5, 11]. Они даны в виде процентов и рассматриваются в рамках модели в виде долей единицы. Ответы типа «Удовлетворен», «Скорее удовлетворен», «Очень высокое» и «Высокое» означают удовлетворенность и их значения в рамках определенного вопроса складываются. Вариант «Затрудняюсь ответить» означает, что участник опроса может склоняться как к положительной оценке, так и к отрицательной. Мы разделяем их в

соотношении 1:1, соответственно считаем $\frac{1}{2}$ от данного показателя. С учетом этого получаются следующие значения:

$$x = [0,911; 0,823; 0,559; 0,617]$$

$K = 1$ – максимально возможное в данных условиях значение уровня профессиональной подготовки;

Определение факторов влияния и весов проведено с помощью рассмотрения взаимосвязей, отношений между субъектами и их целей по нижеописанным принципам:

1) Эгоистический подход.

Данный подход предусматривает экономию личных усилий. Основная роль принадлежит Студенту – только он своими усилиями способен добиться результата и именно его уровень подготовки играет роль в данной модели. Далее по убыванию идут ВУЗ (предоставляющий базу для обучения) и Работодатель (определяющий требования на рынке труда). Соответственно, если субъекты выбирают данный вариант, то они в большей степени заинтересованы экономией средств, нежели ростом уровня подготовки. Значения весов факторов влияния b_i представлены в таблице 2.

Таблица 2. Относительные веса факторов влияния b_i

Вес фактора j	Работодатель	ВУЗ	Студент	Уровень подготовки
b_i	0,25	0,3	0,35	0,1

2) Кооперативный подход.

Студент. Целью партнерства является получение знаний и повышения уровня подготовки. Соответственно со стороны Студента наибольшим значением оценивается фактор уровня подготовки – он и является определяющим. Чуть меньше оцениваются свои усилия. Значимость ВУЗа и Работодателя равны и составляют минимальные значения.

ВУЗ. Максимально оценивает значимость своих усилий – ведь на его базе происходит подготовка специалистов. Далее стоит значимость Студента и уровня подготовки, они оцениваются чуть меньшими равными значениями. Работодатель влияет сравнительно в небольшой степени на процесс обучения, соответственно оценивает минимальным значением.

Работодатель. Его цель в первую очередь повышение уровня подготовки кадров, далее равными значениями оцениваются значимость усилий Студента и своих собственных. Роль ВУЗа определяется минимальным значением, ведь он обучает в основном по общей программе, а на получения специалистом необходимых навыков (в большинстве случаев получаемых в рамках не основной программы обучения) влияют остальные факторы.

Значимости факторов приведены в таблице 3.

Таблица 3. Относительная значимость фактора j для субъекта i , (b_j^i , $i = P, B, C$; $j = P, B, C, x$)

Значимость фактора j для субъекта i	Работодатель	ВУЗ	Студент	Уровень подготовки
Работодатель	2/8	1/8	2/8	3/8
ВУЗ	1/8	3/8	2/8	2/8
Студент	1/7	1/7	2/7	3/7

2.3. ПЛАНИРОВАНИЕ И ПРОВЕДЕНИЕ ВЫЧИСЛИТЕЛЬНЫХ ЭКСПЕРИМЕНТОВ

Стратегии исследования

Исследование модели проводилось в имитационном режиме с помощью метода сценариев [3,6]. Сценарии формируются согласно возможному поведению субъектов. Исходя из смысла рассматриваемого сценария финансирования, определяются и задаются определенные стратегии субъектов.

Для простоты будем задавать сразу стратегии субъектов без расчета среднего арифметического стратегий его элементов. При этом для каждого сценария, в разрезе одного периода времени, значения стратегий субъектов возьмем равными, т.е. $u_P(t) = u_C(t) = u_B(t) = u_O(t)$, при $t = 0, \dots, 4$. Это означает, что у всех субъектов доли ассигнования денежных средств в определенный рассматриваемый период времени равны. В работе рассмотрено шесть сценариев финансирования:

1) Максимальный – соответствует максимально возможному финансированию, когда весь бюджет выделяется на ДПО. Стратегии задаются следующим образом: $u_O(t) = 1$;

2) Средний – задает промежуточные значения, когда половина бюджета ассигнуется на ДПО. В этом случае стратегии определяются так: $u_O(t) = 0,5$;

3) Минимальный – определяет значение, при котором ассигнуется малая часть бюджета. Рассматриваются следующие стратегии: $u_O(t) = 0,2$;

4) Отсутствия финансирования – предполагает, что средства на ДПО выделяться не будут. Стратегии: $u_O(t) = 0$;

5) Уменьшения финансирования – задает ситуацию, когда изначально на ДПО выделяется некоторая значительная часть бюджета и со временем происходит уменьшение финансирования до небольшого значения. Значения стратегий субъектов управления: $u_O(t) = 0,8 - 0,15t$;

6) Увеличения финансирования – описывает ситуацию противоположную сценарию уменьшения финансирования. Стратегии субъектов: $u_O(t) = 0,2 + 0,15t$.

В первых четырёх сценариях стратегии остаются неизменными в течение рассматриваемых периодов, а в пятом и шестом происходит изменение долей ассигнования денежных средств в разных периодах.

Обработка и анализ результатов моделирования

Обработка проводилась с помощью анализа графиков и сравнения значений целевых функций и переменной состояния модели для различных сценариев. Графики целевых функций Работодателя при сценарии увеличения и уменьшения финансирования приведены на рис. 1, а для сценария максимального финансирования и отсутствия финансирования на рис. 2. Функция $J1p(t)$ соответствует целевой функции Работодателя J_P из (1) при использовании функций выигрыша эгоистического подхода, аналогично $J2p(t)$, но при использовании функций выигрыша кооперативного подхода.

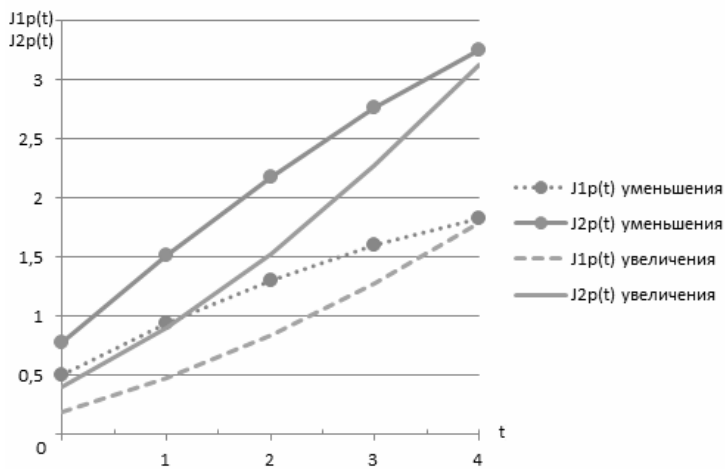


Рис. 1. Сравнение графиков роста целевой функции Работодателя при сценариях уменьшения и увеличения финансирования

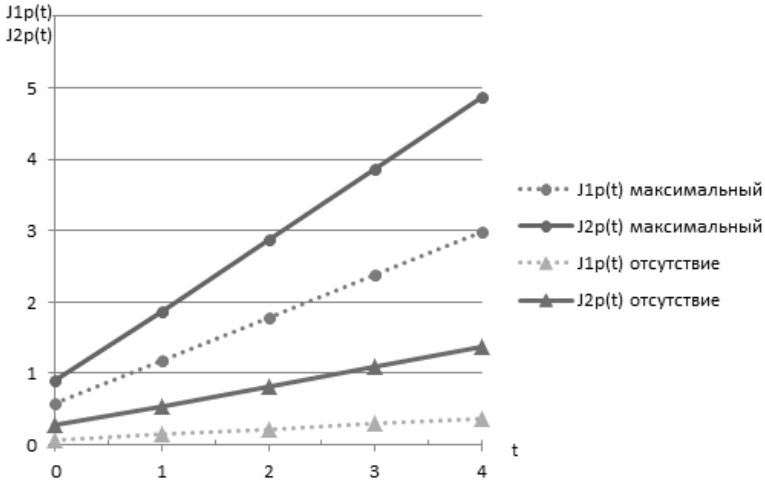


Рис. 2. Сравнение графиков роста целевой функции Работодателя при сценариях максимального и отсутствия финансирования

Как и следовало ожидать, наилучшие результаты достигаются при использовании сценария максимального финансирования, а наихудшие при отсутствии финансирования. Значения целевых функций и переменной состояния относительно сценариев располагаются по убыванию следующим образом: максимальный, уменьшения финансирования, средний, увеличения финансирования, минимальный и отсутствия финансирования.

На рис. 3 представлено сравнение динамики роста уровня профессиональной подготовки в зависимости от используемого сценария по периодам.

При этом для сценария увеличения и уменьшения финансирования в течение рассматриваемого временного отрезка прослеживается следующее поведение функций выигрыша: при сценарии уменьшения – значения функций выигрыша вначале показывают высокие значения, но уменьшаются к окончанию рассматриваемого временного отрезка, а для сценария увеличения – наоборот. Это означает, что субъекты, начиная «много»

вкладывая в ДПО в начале, устанавливают высокое значение переменной состояния. Это позволяет со временем понижать долю ассигнований. При этом установившийся уровень будет способствовать хорошим результатам социального партнерства в ближайшей перспективе.

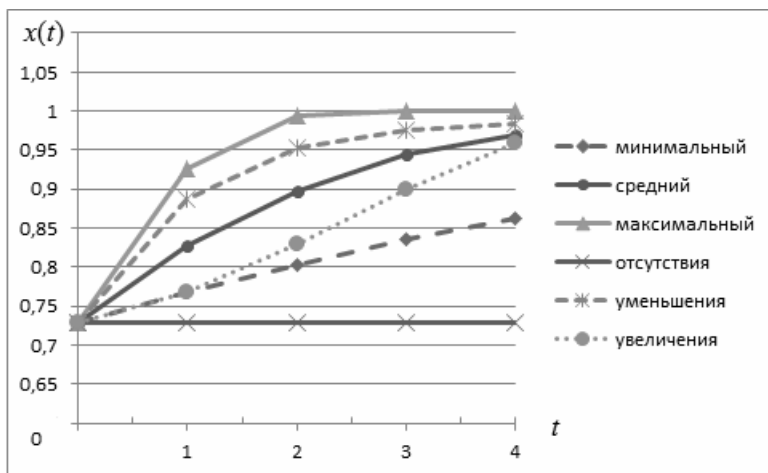


Рис. 3. Сравнение уровня профессиональной подготовки при различных сценариях

Сравнение значений целевых функции субъектов показало, что нельзя сказать, что кто-то больше выигрывает от партнерства, а кто-то меньше. При рассмотрении каждой конкретной стратегии значения целевой функции для определенного субъекта на фоне остальных может принимать как наибольшее (это свидетельствует о том, что данная стратегия этому субъекту подходит больше, чем другим), так и наименьшее значение (относительно значений целевых функций других субъектов) – при другой стратегии.

Для двух вариантов функций выигрыша (эгоистический и кооперативный подход) проведено сравнение значений целевых функций. Сущность этих подходов качественно раскрывается с течением времени. С ростом финансирования наблюдается

увеличение разрыва между подходами. Сравнение наглядно демонстрирует преимущества более высокого уровня социальной интеграции: значение целевой функции в случае кооперации показывает большее значение по сравнению с эгоистическим подходом: для субъекта Работодатель в 3,75 раза (отсутствие финансирования) и 1,64 раза (максимальный сценарий), соответственно для Студента при тех же сценариях в 4,29 раза и в 2,04 раза; для ВУЗа в 2,5 раза при отсутствии финансирования и в 1,76 раза при медианном сценарии.

3. Заключение

В статье описана проблема развития ДПО, причины её появления, обоснована актуальность развития партнерских отношений в данной области и рассмотрены основные субъекты. Построена математическая модель социального партнерства в системе дополнительного профессионального образования. Проведена идентификация параметров и оценка весов факторов влияния в соответствии с данными социологических опросов. Для исследования модели был использован метод сценариев имитационного моделирования.

Применение сценария максимально возможного финансирования денежных средств ведет к развитию сферы ДПО и показывает самый быстрый рост уровня подготовки и удовлетворенности совместной деятельностью партнерами.

Одним из основных результатов является вывод о необходимости объединения усилий субъектов социального партнерства. Действительно, на рассматриваемом периоде использование функций выигрыша кооперативного подхода, характеризующих более развитые социально-партнерские отношения, приводит к ускоренному росту значений целевых функций субъектов по сравнению с эгоистическим подходом. Это демонстрирует преимущества более высокого уровня социальной интеграции.

В любом случае, реализация программы социального партнерства требует обязательного объединения усилий всех субъектов. Именно оно позволит объединить все заинтересованные в

подготовке специалистов стороны, способствуя увеличению уровня профессиональной подготовки и достижению субъектами своих целей.

Литература

1. ГЛУШАНОК Т.М. *Социальное партнерство как средство повышения качества профессионального образования // Современные проблемы науки и образования.* – 2008. – №6. – С. 80–83. – [Электронный ресурс]. – URL: <http://www.science-education.ru/30-1144> (дата обращения: 02.11.13).
2. ЕМЕЛЬЯНОВ А.А. *Имитационное моделирование в экономических информационных системах.* – М.: Финансы и статистика, 2002. – 368 с.
3. КЕЛЬТОН Д.Б., ЛОУ А.М. *Имитационное моделирование.* – СПб.: Питер, 2004. – 847 с.
4. МАРЬИН А. *Социальное партнерство в образовании.* – [Электронный ресурс]. – URL: <http://www.proza.ru/2011/01/21/398>. (дата обращения: 02.11.13).
5. *Отчет по социологическому исследованию «Удовлетворенность студентов обучением в университете».* – [Электронный ресурс]. – URL: <http://umu.sfedu.ru/images/stories/ОККО/2012.docx> (дата обращения: 02.11.13).
6. ПАВЛОВСКИЙ Ю.Н. *Имитационные модели и системы.* – М. 2000.
7. ПАРИНОВ С.И. *Новые возможности имитационного моделирования социально-экономических систем // Искусственные сообщества.* – 2007. – №3–4. – С. 26–61. – [Электронный ресурс]. – URL: <http://simulation.su/uploads/files/default/parinov.pdf>. (дата обращения: 02.11.13).
8. СОВЕТОВ Б.Я., ЯКОВЛЕВ С.Я. *Моделирование систем.* – М.: Высшая школа, 2003. – 295 с.
9. ТАРАСЕНКО Л.В. *Дополнительное профессиональное образование в современной России: становление институ-*

- та вторичной профессиональной социализации. – Ростов-на-Дону: Изд-во ЮФУ, 2009. – 288 с.
10. ТАРАСЕНКО Л.В. *Дополнительное профессиональное образование: становление нового социального института*. – Ростов-на-Дону: Изд-во РГУ, 2001.
 11. ТАРАСЕНКО Л.В., НОР-АРЕВЯН О.А. *Специфика профессиональной специализации современного российского студенчества (на примере вузов Ростовской области)*. – Ростов-на-Дону, 2013.
 12. ТАРАСЕНКО Л.В., УГОЛЬНИЦКИЙ Г.А., ДЬЯЧЕНКО В.К. *Теоретико-игровая формализация динамики уровня доверия между субъектами социального партнерства в системе дополнительного профессионального образования* // Инженерный вестник Дона. – 2013. – №1. – [Электронный ресурс]. – URL: <http://ivdon.ru/magazine/archive/n1y2013/1554> (дата обращения: 02.11.13).
 13. УГОЛЬНИЦКИЙ Г.А. *Устойчивое развитие организаций*. – М.: Физматлит, 2011. – 320 с.
 14. FANDEL G., GIESE A., MOHN B. *Measuring synergy effects of a Public Social Private Partnership (PSPP) project* // Int. J. Production Economics. – 2012. – №140. – P. 815–824.
 15. KIRKPATRICK D.L., KIRKPATRICK J.D. *Implementing the Four Levels: A Practical Guide for Effective Evolution of Training Programs*. – Berrett-Koeler Publishers, 2007. – 153 p.
 16. TALMAN D., YANG Z. *A model of partnership formation* // Journal of Mathematical Economics. – 2011. – №47. – P. 206–212.
 17. ZAHARATUL A.A.Z., ABD SHUKOR H., IR. GHAZARI A.A. *Smart tri-partite partnership: polytechnic – industry – student* // Procedia-Social and Behavioral Sciences. – 2012. – №31. – P. 517–521.

**COMPUTER SIMULATION OF SOCIAL PARTNERSHIP IN
THE SYSTEM OF ADVANCED PROFESSIONAL
EDUCATION**

Larisa Tarassenko, Southern Federal University, Rostov-on-Don,
Doctor of Social Sciences, Professor (socmodel@mail.ru).

Gennady Ougolnitsky, Southern Federal University, Rostov-on-Don,
Doctor of Sciences, Professor (ougoln@mail.ru).

Vladimir Dyachenko, Southern Federal University, Rostov-on-Don,
Master of Sciences in Applied Mathematics and Computer Science
(v.k.diachenko@gmail.com).

Abstract: A game-theoretic model of the system of advanced professional education is proposed. Some approaches to the model identification and investigation based on simulation modeling are considered.

Keywords: social partnership, advanced professional education, simulation, identification, dynamic games.

*Статья представлена к публикации
членом редакционной коллегии Р.М. Нижегородцевым*

Поступила в редакцию 31.07.2013.

Опубликована 30.11.2013.

УДК 004.021 + 004.827 + 519.81
ББК 32.81

МЯГКИЕ ВЫЧИСЛЕНИЯ КАК ПУТЬ К ПРЕОДОЛЕНИЮ НЕ-ФАКТОРОВ В ЗАДАЧАХ ОЦЕНКИ ЖИЗНЕСПОСОБНОСТИ ПРОЕКТОВ

Колоденкова А. Е.¹

*(Уфимский государственный авиационный
технический университет, Уфа)*

Рассматриваются теоретические аспекты НЕ-факторов и методы их учета в задачах оценки жизнеспособности проектов. Предлагаются нечетко-множественный, нечетко-интервальный и генетический подходы к решению данной задачи на основе формирования многокритериальной оценки и выборе приемлемой проектной альтернативы. Данные подходы позволяют повысить обоснованность принятия решения о возможности реализации проекта и за счет этого снизить риск его неудачного завершения.

Ключевые слова: проект, НЕ-факторы, оценка жизнеспособности проекта, мягкие вычисления.

1. Введение

В современных условиях разработка проектов в области энергетики, транспорта, связи и других отраслей промышленности является крайне важной для российской экономики. С ростом масштабов и сложности таких проектов, вовлечением в них большого числа участников и организаций, возрастанием требований к срокам его осуществления, использованию финансовых, материальных и трудовых ресурсов появляется потребность в

¹ Анна Евгеньевна Колоденкова, кандидат технических наук, доцент (anna82_42@mail.ru).

повышении эффективности управления проектами за счет специальных (профессиональных) методов управления, позволяющих осуществлять контроль проектов и обеспечивать их выполнение в срок и в рамках заданного бюджета [3, 8, 11, 12].

В связи с этим все большую актуальность и значимость с финансовой, экономической и других точек зрения приобретают проблемы концептуального проектирования и, в частности, оценки жизнеспособности проекта, направленной на выявление и снижение возможных проектных рисков, а также на сокращение управленческих ошибок, принимаемых руководителем проекта в условиях неопределенности.

Несмотря на большое число работ многих отечественных и зарубежных авторов, посвященных различным научно-исследовательским и опытно-конструкторским проектам, проблема оценки жизнеспособности проекта до сих пор остается открытой, а существующие модели проектов не позволяют рассматривать все ресурсы во взаимосвязи, контролировать и управлять проектом на любой стадии, работать с несколькими проектами одновременно, а также не обеспечивают необходимый уровень объективности оценок.

В настоящей работе рассмотрены методы вычислительного интеллекта, позволяющие преодолеть *НЕ*-факторы в задачах оценки жизнеспособности проектов.

2. Особенности НЕ-факторов в концептуальном проектировании проектов

В настоящее время методология управления проектами, бесспорно, является оправдавшей себя на практике новой формой управленческой деятельности. Достигнутое развитие управления проектами в современной теории и практике, накопленные знания и опыт по управлению проектами в различных сферах сделали возможным и необходимым создание системной модели [5]. По мнению В.И. Воропаева и Г.И. Секлетовой, «системная модель необходима для того, чтобы: выработать общий язык и терминологию; создать основу для разработки

моделей, методов и средств решения задач управления проектами ...».

В связи с этим на основе систематизации обзора литературных источников на рис. 1 предложена *системная модель управления проектом*, основными элементами которой являются: основные ограничения, учитываемые при согласовании разнообразных требований проекта; функции управления проектом, осуществляющиеся разработчиками на всех стадиях жизненного цикла (ЖЦ) проекта; методы и технологии учета НЕ-факторов, используемые при разрешении сложных слабоструктурированных и неструктурированных проблем.

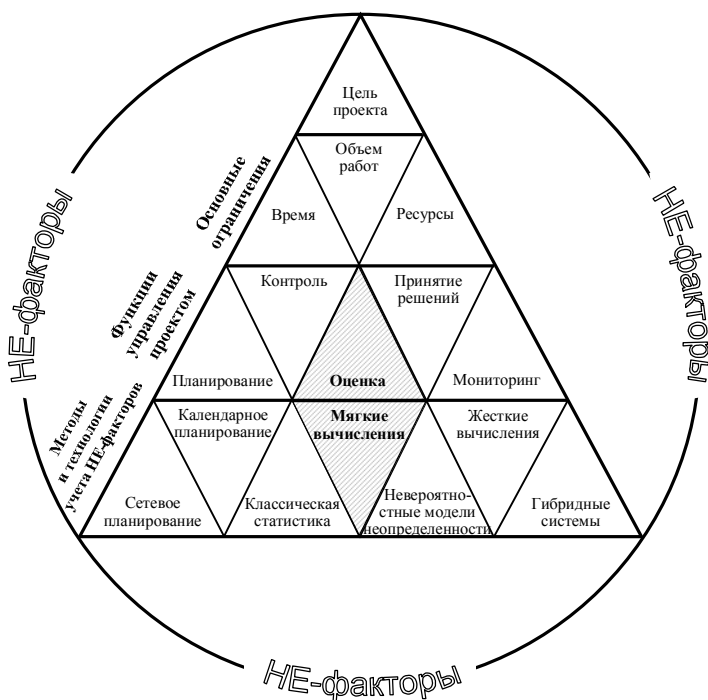


Рис. 1. Системная модель управления проектом

На рис. 1 серым цветом выделены те функции и методы, которым будут посвящены следующие разделы статьи. Отметим, что приведенное разделение методов учета *НЕ*-факторов носит условный характер, поскольку на практике они пересекаются и взаимодействуют между собой на различных этапах принятия решения.

В настоящей работе под *оценкой реалистичности проекта*, именуемой часто как реализуемость (В.В. Липаев), осуществимость (С. Орлик, И. Соммервилл (I. Sommerville)) или жизнеспособность (И.И. Мазур, А.Е. Колоденкова) понимается наличие необходимых ресурсов для осуществления и условий для реализации проекта. Однако такая оценка сопровождается *НЕ*-факторами [13, 14], негативно влияющими на ход выполнения проекта и, прежде всего, на планирование работ и принятие управленческих решений по проекту. Сюда можно отнести неполноту исходных данных для проектирования, изменение требований, сроков и объема выделяемых ресурсов на проектирование, высокую занятость разработчиков.

В работе [11] дан обзор *НЕ*-факторов, а также введены их классификация и методы их учета применительно к задаче оценки жизнеспособности проекта. Среди *НЕ*-факторов, затрудняющих оценку жизнеспособности проекта, выделены следующие: *неточность* (величина, которая может быть получена с точностью, не превышающей некоторый порог, определенный природой соответствующего параметра), *нечеткость* (величина, связанная с отсутствием точных границ соответствующего параметра), *неоднозначность* (величина, отражающая множество альтернатив оцениваемого параметра неравномерно с точки зрения некоторой конкретной семантики).

Из теории и практики системной инженерии известно, что учитывать *НЕ*-факторы при управлении проектами только лишь формальными методами невозможно. Здесь весьма перспективно использование бурно развивающегося в рамках теории искусственного интеллекта научного направления, провозглашенного Л. Заде и получившего название «*мягкие вычисления*» (*Soft Computing*) или «*вычислительный интеллект*»

(*Computational Intelligence*) [17], позволяющие «устранить» неопределенности и придать проблеме оценки жизнеспособности проекта количественную определенность.

3. Состояние развития методов и технологий мягких вычислений в задачах оценки жизнеспособности проектов

Мягкие вычисления положены в основу многочисленных программ («Real World Computing», «New Information Processing Technology» и др.) разработки и создания адаптивных, эволюционирующих, сверхвысокопроизводительных вычислительных машин шестого поколения и, в отличие от традиционных жестких вычислений (*Hard Computing*), нацелены на максимальное приспособление к реальной действительности.



Рис. 2. Классификация вычислительных методов и технологий учета НЕ-факторов в задачах оценки жизнеспособности проектов

На рис. 2 предложена классификация вычислительных методов и технологий учета *НЕ*-факторов в задачах оценки жизнеспособности проектов.

Применительно к оценке жизнеспособности проекта наиболее интересны следующие составляющие: нечеткие методы, байесовские сети доверия, генетические алгоритмы и агентный подход. Далее рассмотрим жизнеспособность проекта на базе концепции нечетких, интервальных подходов и генетических алгоритмов.

4. Нечетко-множественный и нечетко-интервальный подходы к оценке жизнеспособности проекта

В последнее десятилетие среди различных направлений вычислительного интеллекта на одно из ведущих мест все больше претендует *теория нечетких множеств* [2, 7].

При решении задач, в которых доминирующими факторами являются неточность и неполнота информации, особую роль играют *лингвистические переменные*, значениями которых являются слова и предложения естественного языка [7].

Таким образом, теория нечетких множеств позволяет дать строгое математическое описание в действительности расплывчатых утверждений, реализуя тем самым попытку преодолеть лингвистический барьер между человеком, суждения и оценки которого являются приближенными и нечеткими.

В ситуации, когда информация о ресурсах проекта является неполной и нечеткой, целесообразно использовать нечетко-множественный подход, описанный в работах [9–11], основанный на использовании аппарата нечетких множеств и лингвистических переменных.

Идея *нечетко-множественного подхода* к оценке жизнеспособности проекта заключается в анализе построенных альтернатив разработки проекта в виде сетевых графиков выполнения проектных работ формировании специально сформированного *нечеткого показателя жизнеспособности* $\Delta^{(1)}$

в виде критического пути, длительность которого представляется нечетким числом

$$\Delta^{(i)} = [T^{(i)-}, T^{(i)+}], \quad i = 1, \dots, m,$$

с возможностью последующего выбора приемлемой альтернативы

$$D = \arg \max \{ \mu_D(F_1), \mu_D(F_2), \dots, \mu_D(F_m) \},$$

где $T^{(i)-}$, $T^{(i)+}$ – минимальное и максимальное время выполнения i -й альтернативы; $\mu_D(F_m)$ – функция принадлежности альтернатив; m – число альтернатив разработки проекта.

Решение задачи расчета показателя $\Delta^{(i)}$ сводится к нахождению по имеющейся функции принадлежности объема работ и известной производительности разработчиков функции принадлежности времени работ. В основу предлагаемого подхода положен образ нечеткого множества при четком и нечетком отображении, позволяющий рассчитывать в виде трапециевидного нечеткого числа ожидаемые времена выполнения работ, за которые может быть выполнен заданный объем работ.

В случае, когда имеется лишь интервальная исходная информация, т.е. известны лишь границы изменения значений анализируемого параметра разрабатываемого проекта, целесообразно к оценке жизнеспособности проекта использовать нечетко-интервальный подход [11].

Идея нечетко-интервального подхода заключается в анализе построенных альтернатив разработки проекта в виде сетевых графиков выполнения проектных работ и формировании скалярных интервальных многокритериальных оценок вида

$$P(x_i) = \sum_{j=1}^n w_j^h p_j^h(x_i), \quad i = 1, \dots, m, \quad j = 1, \dots, n,$$

с возможностью последующего выбора приемлемой альтернативы:

$$x^0 = \arg \max_{x_i \in X} P(x_i), \quad i = 1, \dots, m,$$

где w_j^h – нормализованный интервальный коэффициент относительной важности j -го частного критерия альтернатив $x_i \in X$; $p_j^h(x_i)$ – нормализованные интервальные частные критерии

альтернатив $x_i \in X$; m – число альтернатив разработки проекта; n – количество частных критериев.

Таким образом, предлагаемые выше подходы позволяют на стадии анализа жизнеспособности проекта принять решение либо об отказе от анализируемого варианта проекта, либо о запуске его в производство, либо о формировании его новых альтернативных вариантов в условиях неточной и нечеткой исходной информации.

5. Генетические алгоритмы в задачах управления проектами: исследования, практическое применение, программные средства

Другим перспективным направлением исследований в области вычислительного интеллекта являются *генетические алгоритмы*, сформулированные в 1975 г. Д. Холландом (J. Holland), являются алгоритмами поисковой оптимизации, основанными на математическом моделировании биологических механизмов с помощью принципов популяционной генетики, позволяющими находить оптимальные решения.

Основным преимуществом генетических алгоритмов является универсальность, простота их реализации, а также относительно высокая скорость работы в задачах с неопределенными исходными данными, для которых нет четких алгоритмов решения. Однако при всех видимых достоинствах основными их недостатками является относительно высокая вычислительная стоимость и отсутствие гарантии того, что будет найдено оптимальное решение за приемлемое время [6].

В период бурного расцвета информационных технологий идеи Д. Холланда и его последователей находят все более широкое применение для решения задач календарного планирования с ограниченными ресурсами, которые относятся к классу *NP*-трудных задач; комбинаторных задач, а также задач поиска оптимальных решений, формирования моделей и прогнозирования значений различных показателей [4, 15]. По утверждению Д. Холланда, «в больших и сложных про-

блемных областях, при определенных условиях, генетические алгоритмы будут, как правило, показывать оптимальные или близкие к ним результаты».

В последние годы на рынке программного обеспечения появилось множество специализированных и универсальных программных средств, реализующие генетические алгоритмы [1, 16]. Наиболее известными и популярными в России являются следующие программные средства, применяемые в задачах оценки жизнеспособности проектов (рис. 3).



Рис. 3. Классификация программных средств, реализующие генетические алгоритмы в задачах оценки жизнеспособности проектов

Однако приходится констатировать, что большинство программных средств разработано под определенные практические задачи; имеют ограниченный набор генетических операторов и тестовых функций; продаются по высокой цене; плохо приспособлены для модификаций, а также не позволяют сохранять результаты поиска. В связи с этим необходимо создавать инструментальные средства (программные системы), которые позво-

ляли бы исследовать и доказывать эффективность предложенных методик на тестовых функциях в решении практических задач.

6. Применение генетического алгоритма к оценке жизнеспособности проекта

В ситуации, когда исходные данные о ресурсах (финансовых, временных, трудовых и др.) на проектирование являются нечеткими и часто изменяющимися, автором предлагается к оценке жизнеспособности проекта использовать подход с применением генетического алгоритма.

Идея предложенного подхода к оценке жизнеспособности проекта заключается в анализе возможных альтернатив разработки проекта. Оценка каждой i -й альтернативы разработки проекта осуществляется на основе найденной ожидаемой стоимости альтернативы $s(x_i)$, $i = 1, \dots, m$, и формировании интервальной многокритериальной оценки $P(x_i)$.

Прежде чем приступить к описанию формальной схемы оценки жизнеспособности проекта, опишем вводимые допущения:

- ♦ во-первых, разработано и анализируется четыре альтернативы разработки проекта (x_1, x_2, x_3, x_4) ;

- ♦ во-вторых, рассмотрено четыре частных критерия жизнеспособности альтернатив, представленные в виде нечетких трапециевидных чисел, причем их границы не пересекаются;

- ♦ в-третьих, частный критерий стоимости альтернатив обозначен через $s(x_i)$, $s(x_i) = (s_{i,1}, s_{i,2}, s_{i,3}, s_{i,4})$ ($s_{i,1}$ – пессимистическая оценка стоимости; $[s_{i,2}; s_{i,3}]$ – интервал возможной стоимости; $s_{i,4}$ – оптимистическая оценка стоимости), $i = 1, \dots, 4$.

Пусть имеется ограниченное множество из m допустимых альтернатив разработки проекта $X = \{x_1, x_2, \dots, x_m\}$, где каждая альтернатива $x_i \in X$, $i = 1, \dots, m$, оценивается кортежем из n (ненормализованных) интервальных частных критериев $K = \langle k_j(x_i) \rangle$, $j = 1, \dots, n$. Ставится задача нахождения вектора

ожидаемой стоимости для каждой i -й альтернативы разработки проекта:

$$s(x_i) = (s_{i,1}, s_{i,2}, s_{i,3}, s_{i,4}), i = 1, \dots, m,$$

применительно к оценке обобщенной полезности, характеризующей жизнеспособность каждой i -й альтернативы разработки проекта:

$$P(x_i) = \sum_{j=1}^n w_j^H p_j^H(x_i), i = 1, \dots, m, j = 1, \dots, n,$$

с возможностью последующего выбора приемлемой альтернативы.

Здесь w_j^H – нормализованный интервальный коэффициент относительной важности j -го частного критерия альтернатив $x_i \in X$; $p_j^H(x_i)$ – нормализованные интервальные частные критерии альтернатив $x_i \in X$ ($0 < p_j^H(x_i) < 1$).

Таким образом, задача оценки жизнеспособности альтернатив разработки проекта описывается следующей целевой функцией

$$(1) P(x_i) = \sum_{j=1}^n w_j^H p_j^H(x_i) \rightarrow \max, x_i \in X,$$

и ограничениями

$$(2) w_j^H = [\alpha_{jmin}; \alpha_{jmax}],$$

$$(3) s_{min} \leq s(x_i) \leq s_{max},$$

где $\sum_{j=1}^n \alpha_{jmin} < 1$, $\sum_{j=1}^n \alpha_{jmax} > 1$; s_{min} и s_{max} – значения минимальной

и максимальной стоимости альтернативы.

Поскольку решаемая задача имеет комбинаторный характер и может быть сформулирована как поиск наилучшего решения, то целесообразно применить генетический алгоритм, состоящий из следующих **десяти шагов**:

На первом шаге вводятся исходные данные: $k_f(x_i)$ – ненормализованные интервальные частные критерии альтернатив; s_{min} и s_{max} – значения минимальной и максимальной стоимости альтернативы; kol – количество итераций; d – позиция кроссинговера; q – позиция мутации; w_j – интервальный

коэффициент относительной важности j -го частного критерия альтернатив.

На втором шаге определяются необходимые условия оптимизации: целевая функция (1) и ограничения (2)–(3).

На третьем шаге формируются особи-родители (четыре варианта векторов стоимости для каждой i -й альтернативы разработки проекта) с использованием генератора случайных чисел и учетом ограничения (3):

$$s^P(x_1) = (s_{1,1}^P, s_{1,2}^P, s_{1,3}^P, s_{1,4}^P), \quad s^P(x_2) = (s_{2,1}^P, s_{2,2}^P, s_{2,3}^P, s_{2,4}^P), \\ s^P(x_3) = (s_{3,1}^P, s_{3,2}^P, s_{3,3}^P, s_{3,4}^P), \quad s^P(x_4) = (s_{4,1}^P, s_{4,2}^P, s_{4,3}^P, s_{4,4}^P).$$

На четвертом шаге производится операция одноточечного кроссинговера над полученными вариантами особей-родителей (точка кроссинговера $d < 4$). В результате операции кроссинговера получается 12 особей-потомков:

$$s^{\Pi}(x_1) = (s_{1,1}^P, s_{1,d}^P, s_{2,d+1}^P, s_{2,4}^P), \quad s^{\Pi}(x_2) = (s_{2,1}^P, s_{2,d}^P, s_{1,d+1}^P, s_{1,4}^P), \\ s^{\Pi}(x_1) = (s_{1,1}^P, s_{1,d}^P, s_{3,d+1}^P, s_{3,4}^P), \quad s^{\Pi}(x_3) = (s_{3,1}^P, s_{3,d}^P, s_{1,d+1}^P, s_{1,4}^P), \\ s^{\Pi}(x_1) = (s_{1,1}^P, s_{1,d}^P, s_{4,d+1}^P, s_{4,4}^P), \quad s^{\Pi}(x_4) = (s_{4,1}^P, s_{4,d}^P, s_{1,d+1}^P, s_{1,4}^P), \\ s^{\Pi}(x_2) = (s_{2,1}^P, s_{2,d}^P, s_{3,d+1}^P, s_{3,4}^P), \quad s^{\Pi}(x_3) = (s_{3,1}^P, s_{3,d}^P, s_{2,d+1}^P, s_{2,4}^P), \\ s^{\Pi}(x_2) = (s_{2,1}^P, s_{2,d}^P, s_{4,d+1}^P, s_{4,4}^P), \quad s^{\Pi}(x_4) = (s_{4,1}^P, s_{4,d}^P, s_{2,d+1}^P, s_{2,4}^P), \\ s^{\Pi}(x_3) = (s_{3,1}^P, s_{3,d}^P, s_{4,d+1}^P, s_{4,4}^P), \quad s^{\Pi}(x_4) = (s_{4,1}^P, s_{4,d}^P, s_{3,d+1}^P, s_{3,4}^P).$$

Получаемые особи-потомки проверяются на соответствие заданному ограничению (3).

На пятом шаге производится операция одноточечной мутации полученных 12 особей-потомков (точка мутации $q < 4$). Из пришедшей особи-потомка получается новая особь-потомок с мутированными генами $\bar{s}^M(x_1) = (s_{1,1}^M, s_{1,q}^M, \bar{s}_{1,q+1}^M, s_{1,4}^M)$ с учетом ограничения (3). Аналогичным образом данная операция производится и над остальными особями-потомками.

На шестом шаге осуществляется нормализация частных критериев $p_j^n(x_i)$ для каждой i -й альтернативы разработки проекта

$$p_j^H(x_i) = \frac{k_j(x_i)}{k_j^+}, \quad 0 < p_j^H(x_i) < 1, \quad i = 1, \dots, m, \quad j = 1, \dots, n,$$

а также интервальных коэффициентов w_j^H относительной важности частных критериев:

$$w_j^H = \frac{w_j}{\sum_{j=1}^n w_j}, \quad j = 1, \dots, n,$$

где $k_j(x_i)$ – интервал j -го частного критерия i -й альтернативы разработки проекта; k_j^+ – максимальный интервал j -го частного критерия на данном множестве альтернатив $x_i \in X$; w_j – интервальный коэффициент относительной важности j -го частного критерия; $w_j^H = [\alpha_{jmin}; \alpha_{jmax}]$ – нормированный интервальный коэффициент относительной важности j -го частного критерия. Аналогичные расчеты производятся и для критерия стоимости.

Следует заметить, что интервальные частные критерии $k_j(x_i)$, $j = 1, \dots, n$, не должны пересекаться, поскольку только в этом случае устанавливаются отношения «больше» или «меньше» [11].

На седьмом шаге рассчитываются значения целевой функции (3) для всех особей популяции, а именно для всех вариантов векторов ожидаемой стоимости альтернатив проекта.

На восьмом шаге отбираются четыре особи (четыре вектора) из 12 возможных с наибольшими значениями целевой функции, которые будут родителями для следующей итерации (поколения), или же в случае выполнения всех итерации – результатом вычислений.

В предложенной задаче рассматривается четыре альтернативы разработки проекта, т.е. отбирается всего четыре вектора ожидаемой стоимости альтернатив проекта. Отбор особей производится ранговым методом, т.е. особи популяции ранжируются по значениям их функции приспособленности (ранжирование проводится по возрастанию величин). Так как оценки обобщенной полезности $P(x_i)$ представлены в виде трапециевидных

нечетких чисел, то для выбора наилучшей альтернативы проекта применяется метод Чью–Парка, позволяющий каждому трапециевидному числу поставить в соответствии четкое число Sr ($P(x_i)$).

На девятом шаге сохраняются полученные результаты и производится проверка останова. Если заданное количество итерации выполнено, то переходим к 10 шагу, иначе возвращаемся к 4 шагу.

На десятом шаге осуществляется вывод результатов (векторы ожидаемой стоимости альтернатив $s(x_i)$ и оценки Sr ($P(x_i)$) для каждой i -й альтернативы разработки проекта).

Заметим, что генетический алгоритм не совершенен, поскольку работает с задачей, являющейся упрощенной моделью реального процесса оценки жизнеспособности проекта и учитываются не все частные критерии.

Пример. Целью вычислительных экспериментов является нахождение оптимального решения задачи оценки жизнеспособности альтернативы разработки проекта для возможности дальнейшей его реализации. Далее рассмотрим работу генетического алгоритма, описанного выше на конкретном примере.

Пусть имеется четыре альтернативы (x_1, x_2, x_3, x_4) разработки проекта, описываемые совокупностью четырех частных критериев жизнеспособности: k_1 – время выполнения проекта, мес.; k_2 – трудоемкость реализации проекта, чел./мес.; k_3 – вероятность успеха проекта в условиях возникновения ситуаций, при которых цели, поставленные в проекте, могут быть не достигнуты полностью или частично, а также примерная стоимость проекта $s(x_i) = [600; 800]$ тыс. руб. ($i = 1, \dots, 4$). Ставится задача оценки альтернатив с возможностью последующего выбора приемлемой альтернативы разработки проекта.

Расчетные значения частных критериев для каждой альтернативы представлены в таблице 1.

Предположим, что основе опроса мнения группы разработчиков известны следующие весовые коэффициенты относительной важности частных критериев в виде трапециевидных чисел:

$$w_1 = [3; 3,5; 3,8; 4,5], \quad w_2 = [1; 1,3; 1,5; 2],$$

$$w_3 = [2; 2,4; 2,6; 3], \quad w_{4(s)} = [3; 3,2; 3,5; 4].$$

где $w_4(s)$ – весовой коэффициент относительной важности критерия стоимости альтернативы.

Таблица 1. Расчетные значения частных критериев жизнеспособности альтернатив

Альтернативы проекта	Частные критерии жизнеспособности альтернатив		
	k_1	k_2	k_3
x_1	[9,2; 9,4; 9,6; 9,9]	[17; 19; 20; 21]	[0,8; 0,82; 0,83; 0,84]
x_2	[12; 12,4; 12,7; 13]	[6; 7; 9; 10]	[0,84; 0,85; 0,86; 0,87]
x_3	[8; 8,2; 8,4; 8,5]	[22; 23; 25; 26]	[0,7; 0,72; 0,73; 0,74]
x_4	[10; 10,3; 10,4; 10,5]	[10; 12; 15; 17]	[0,87; 0,88; 0,89; 0,9]

Начальные значения параметров генетического алгоритма следующие: точка кроссовера (скрещивания) – 1; точка мутации – 1; количество итераций – 10.

В результате тестовой работы разработанного программного обеспечения, особенностью которого является работа с интервальными значениями, были получены следующие решения, представленные в таблице 2.

Таблица 2. Результаты оптимизации

Альтернативы проекта	Вектор стоимости проекта	Оценка обобщенной полезности
x_1	[643; 692; 732; 784]	0,677
x_2	[643; 692; 787; 791]	0,668
x_3	[660; 674; 787; 797]	0,261
x_4	[643; 692; 772; 795]	0,543

Из таблицы 2 видно, что альтернатива x_1 со значениями стоимости [643; 692; 732; 784] и соответствующей ей обобщенной полезностью $F(x_1) = 0,677$ является наилучшей альтернативой разработки проекта.

Вектор распределения [643; 692; 732; 784] означает, что реализация альтернативы x_1 разработки проекта составит от 643 до 784 тыс. руб.

7. Перспективные исследования жизнеспособности проектов в условиях неопределенности

На сегодняшний день методы вычислительного интеллекта далеко не исчерпали свой потенциал, так как наблюдается их совершенствование во всех высокотехнологичных отраслях экономики (Минобороны России, МВД России, МЧС России и другие федеральные органы исполнительной власти) и многих отраслях промышленности (оборонная, атомная, машиностроение и др.). Это особенно важно в современном мире, в котором общество не может успешно развиваться без рационального управления сложными проектами.

Весьма перспективными направлениями вычислительного интеллекта в области концептуального проектирования, которые могут привести к качественным изменениям в технике и технологиях, являются *многоагентные технологии* для решения задач распределения ресурсов при управлении проектом, а также *метод рассуждений на основе прецедентов (CBR-метод)* для решения задачи выбора жизнеспособного проекта в условиях, когда невозможно или нецелесообразно построение математических моделей.

8. Заключение

В настоящей работе предложен подход к оценке жизнеспособности проектов с применением генетического алгоритма для информационно-управляющих и вычислительных систем, в основе которого лежит нахождение ожидаемой стоимости альтернативы проекта, а также формирование интервальных многокритериальных оценок проекта.

Предлагаемый подход отличается от известных тем, что, во-первых, частные критерия жизнеспособности альтернатив про-

екта представлены в виде нечетких трапециевидных чисел; во-вторых, позволяет руководителям проекта на ранней стадии его разработки выполнить сравнительный анализ различных вариантов реализации проектов, что приводит к повышению обоснованности принятия решения о возможности его реализации в условиях неточной и нечеткой исходной информации, тем самым снижает риск его неудачного завершения; в-третьих, позволяет формализовать процедуру оценки проекта, что делает ее возможным реализовать в виде программной компоненты в информационно-управляющих и вычислительных системах.

Литература

1. АНДРЕЙЧИКОВ А.В., АНДРЕЙЧИКОВА О.Н. *Интеллектуальные информационные системы*: Учебник. – М.: Финансы и статистика, 2004. – 424 с.
2. БОРИСОВ А.Н., КРУМБЕРГ П.О., ФЕДОРОВ И.П. *Принятие решений на основе нечетких моделей. Примеры использования*. – Рига «Зинатне», 1990 – 184 с.
3. БУРКОВ В.Н., НОВИКОВ Д.А. *Как управлять проектами*: Научно-практическое издание. – М.: СИНТЕГ – ГЕО, 1997. – 188 с.
4. ВАСИЛЬЕВ А.С., МАТВЕЙКИН В.Г. *Модификация генетического алгоритма для решения задачи календарного планирования с ограниченными ресурсами* // Вестник ТГТУ. [Электронный ресурс]. – URL: http://vestnik.tstu.ru/rus/t_14/pdf/14_2_012.pdf (дата обращения: 07.03.13).
5. ВОРОПАЕВ В.И., СЕКЛЕТОВА Г.И. *Системное представление управления проектами*. Учебное пособие. – М.: ГОУ ДПО ГАСИС, 2008. – 13 с.
6. ГЛАДКОВ Л.А., КУРЕЙЧИК В.В., КУРЕЙЧИК В.М. *Генетические алгоритмы*: Учебное пособие. – М.: Физматлит, 2006. – 320 с.
7. ЗАДЕ Л.А. *Понятие лингвистической переменной и его применение к принятию приближенных решений*. – М.: Мир, 1976. – 166 с.

8. КОЛОДЕНКОВА А.Е. *Анализ жизнеспособности – важная стадия жизненного цикла инновационных программных проектов* // Программная инженерия. – 2010. – №1. – С. 21–30.
9. КОЛОДЕНКОВА А.Е. *Нечетко-множественный подход к оценке реалистичности альтернатив программного обеспечения мехатронных систем* // Мехатроника, автоматизация, управление. – 2011. – №4. – С. 45–53.
10. КОЛОДЕНКОВА А.Е. *НЕ-факторы и методы вычислительного интеллекта в концептуальном проектировании* // Интегрированные модели и мягкие вычисления в искусственном интеллекте: Сборник трудов VI-й Междунар. науч.-практ. конф. – М.: Физматлит. – 2011. – Т. 2. – С. 710–721.
11. КОЛОДЕНКОВА А.Е. *Оценка жизнеспособности программных проектов в условиях нечеткости исходных данных* // Программная инженерия. – 2011. – №5. – С. 10–16.
12. МАТВЕЕВ А.А., НОВИКОВ Д.А., ЦВЕТКОВ А.В. *Модели и методы управления портфелями проектов*. – М.: ПМСОФТ, 2005. – 206 с.
13. НАРИНЬЯНИ А.С. *Введение в недоопределенность* // Информационные технологии. Приложение. – 2007. – №4 – 32 с.
14. НАРИНЬЯНИ А.С. *НЕ-ФАКТОРЫ: STATE OF ART* // Научная сессия МИФИ-2004. – Т. 3. – С. 26–30.
15. ПОПОВ В.А., БЕРДОЧНИК А.В. *Задача комбинаторной оптимизации в управлении предприятием на основе генетического алгоритма*. [Электронный ресурс]. – URL: <http://www.khai.edu/csp/nauchportal/Arhiv/REKS/2009/REKS409/PopBerd.pdf> (дата обращения: 07.06.13).
16. FILHO J.R., ALIPPI C., TRELEAVEN P. *Genetic Algorithm Programming Environments Department of Computer Science – University College London*. [Электронный ресурс]. – URL: http://www.masters.donntu.edu.ua/2006/kita/chvala/library/filho_genetic.pdf (дата обращения: 07.06.13).

17. ZADEH L. *Fuzzy Logic, Neural Networks, and Soft Computing* // Communications of the ACM. – March, 1994. – Vol. 37, №3. – P. 77–84.

**SOFT COMPUTING AS A WAY TO OVERCOMING
NOT-FACTORS IN THE PROBLEMS OF THE PROJECTS
VIABILITY ESTIMATION**

Anna Kolodenkova, Ufa State Aviation Technical University, Ufa,
Cand.Sc., assistant professor (anna82_42@mail.ru).

Abstract: We study theoretical aspects of Not-factors and methods of their accounting in the problem of projects viability estimation, and suggest multi-fuzzy, fuzzy-interval and genetic-algorithm approaches based on formation of multi-criteria evaluations and choice of an acceptable design alternative. The proposed routines help to make more reliable decisions on possibility of project implementation and, thus, to lower risk of project failure.

Keywords: project, not-factors, project viability estimation, soft computing.

*Статья представлена к публикации
членом редакционной коллегии В.Д. Богатыревым*

*Поступила в редакцию 11.06.2013.
Опубликована 30.11.2013.*

УДК 519
ББК 32.81

РЕШЕНИЕ ЗАДАЧИ ЭФФЕКТИВНОГО РАСПРЕДЕЛЕНИЯ РЕСУРСОВ НА ОСНОВЕ МЕХАНИЗМА ГРОВСА-ЛЕЙДЯРДА ПРИ ТРАНСФЕРАБЕЛЬНОЙ ПОЛЕЗНОСТИ¹

Коргин Н. А.²,

*(ФГБУН Институт проблем управления
им В.А. Трапезникова РАН, Москва,
Московский физико-технический институт, Москва)*

Корепанов В. О.³

*(ФГБУН Институт проблем управления
им В.А. Трапезникова РАН, Москва)*

Решается задача разработки механизма распределения ограниченных ресурсов, эффективного в смысле максимизации суммарной полезности получателей ресурсов в условиях, когда возможна передача полезности между ними. В качестве решения предлагается адаптация механизма Гровса–Лейдярда, первоначально предложенного для решения задачи определения объема выпуска коллективного блага.

Ключевые слова: дизайн механизмов, реализация по Нэшу, распределение ресурсов.

¹ Работа при поддержке РФФИ, грант № 12-07-3124412.

² Николай Андреевич Коргин, кандидат технических наук, доцент (nkorgin@ipi.ru, Москва, ул. Профсоюзная, д. 65, тел. (495) 335-60-37).

³ Всеволод Олегович Корепанов, кандидат технических наук (Москва, ул. Профсоюзная, д. 65, тел. (495) 334-79-00).

1. Введение

Одной из ключевых проблем в теории управления в социальных и экономических системах является проблема минимизации потерь из-за отсутствия полной информации, необходимой для принятия управленческих решений. При этом специфика социально-экономических систем позволяет выделить отдельный класс задач – в которых необходимая информация для принятия решений недоступна лицу, принимающему решения, но доступна другим участникам системы, интересы которых затрагивают принимаемые решения – задачи принятия решений в условиях неполной асимметричной информированности [11]. Для данного класса задач исследуется возможность разработки механизмов, в которых лицо, принимающее решение, получает необходимую для этого информацию от остальных участников системы и которые позволяют получать эффективные решения, позволяющие достигать максимальной суммарной полезности всех участников системы.

В рамках современной теории разработки механизмов принятия решений в социально-экономических системах (*mechanism design*) можно выделить два класса задач: разработку механизмов в условиях нетрансферабельной полезности, когда передача полезности между участниками системы невозможна, и в условиях трансферабельной полезности – когда возможна передача полезности между участниками системы.

При этом классическим результатом, например [2, 27], является тот факт, что для первого класса задач за счет передачи информации от участников системы к ЛПР невозможно обеспечить ту же эффективность принятия решений, что и в условиях полной информированности ЛПР о необходимых параметрах системы. В то время как для второго класса задач это возможно – эффективность механизмов в условиях неполной асимметричной информированности может быть не ниже, чем в условиях полной информированности.

Исследования в области разработки эффективных механизмов для второго класса задач были начаты еще в семидесятых годах прошлого столетия [19, 21, 32]. Но все эти механизмы

имели определенные недостатки, затрудняющие их практическую реализацию.

В частности, механизм Гровса–Лейдярда [19] не гарантировал индивидуальной рациональности получаемого решения, так как оно не являлось равновесием Линдаля. Механизм Гурвича [21] использовал структуру сообщений от участников системы ЛПР, затрудняющее его практическую реализацию. Механизм Волкера [32] давал неустойчивое решение. Кроме того, эффективные решения в последних двух механизмах не могли быть достигнуты в рамках обучающей динамики, т.е. их реализация на практике была крайне затруднительна.

В данный момент наблюдается очередная волна интереса к данной проблеме: появился целый ряд новых публикаций, содержащих результаты теоретических исследований и имитационных экспериментов – см., например [12, 20, 28, 31]. В основном исследования сосредоточены на разработке механизмов, реализующих равновесие Линдаля при определении уровня производства коллективного блага. При этом общим недостатком большинства предлагаемых в данных работах механизмов являются:

- 1) сложные сообщения от участников системы к ЛПР;
- 2) несбалансированность побочных платежей при неравновесных заявках.

Задача распределения ограниченных ресурсов с помощью нерыночных механизмов при нетрансферабельной полезности рассматривалась как отечественными авторами, например [2, 10], так и зарубежными [13, 30]. При этом практическая актуальность разработки таких механизмов является актуальной и на данный момент [17, 26].

Публикации, посвященные построению эффективных механизмов распределения ресурсов при наличии трансферабельной полезности, в основном рассматривают эту задачу для моделей мультиагентных систем со сложной сетевой структурой. В первую очередь, исследуются так называемые «аукционные подходы» определения приоритетности потребителей (см., например [14]), в которых распределение ресурсов определяется на основе того, кто из претендентов назовет большую цену за единицу ресурса, по которой он готов компенсировать осталь-

ным претендентам то количество ресурса, которое они недополучают. Однако основной акцент в этих моделях делается на сложные процедуры распределения ресурсов на сетевых структурах [22, 24], а сами механизмы строятся на основе подходов к построению неманипулируемых механизмов (Викри–Гровса–Кларка) [23].

Исследования ведутся также в направлении построения «квази»-оптимальных механизмов, которые могут реализовывать почти оптимальное распределение ресурсов, например, [27].

Отдельным направлением следует выделить работы в области построения процедур распределенной оптимизации, где в последнее время были получены результаты, очень тесно пересекающиеся с теорией разработки механизмов [15]. Однако в настоящее время эти итеративные процедуры не представлены в форме механизмов, что не позволяет исследовать их теоретико-игровые свойства.

Наиболее близкой к предлагаемым авторами в данной статье подходу можно считать работу [20], в которой был предложен механизм, реализующий равновесие Вальраса при распределении индивидуальных благ. Но:

- 1) предложенное ими решение не позволяет учитывать ограниченность распределяемого ресурса;
- 2) механизм обладает все той же сложной структурой заявок, о которой уже упоминалось выше.

Нами предлагается развитие механизма Гровса–Лейдярда, предложенное в [8] для решения задачи активной экспертизы при трансферабельной полезности. В частности, было показано, что в рамках решения задачи активной экспертизы механизм Гровса–Лейдярда является индивидуально рациональным (что являлось, пожалуй, его ключевым недостатком при решении задач определения уровня коллективного блага). Данный механизм предполагается адаптировать для задачи распределения ресурсов, так как можно предположить, что механизм распределения ресурсов также будет реализовывать индивидуально-рациональные решения и можно будет обеспечить сбалансированность трансферов как в эффективном решении, так и вне его.

В основе адаптации лежит идея представления задачи распределения ресурсов как многокритериальной задачи активной экспертизы, целесообразность и продуктивность которой была продемонстрирована в [7] для задачи распределения ресурсов при нетрансферабельной полезности. Суть идеи очень проста: задачу распределения ресурсов предлагается трактовать не как задачу распределения индивидуальных благ, а как задачу многокритериального выбора, в которой каждый из агентов может сообщить, каким он хочет видеть значение многокритериального коллективного блага – распределения ресурсов между всеми агентами.

Структура дальнейшего изложения такова. В разделе 1 описываются формальная постановка задачи эффективного распределения ресурсов (максимизирующего сумму полезностей всех агентов) и модель, для которой ищется решение этой задачи. В разделе 2 описывается предлагаемый механизм распределения ресурсов на основе многокритериального голосования и доказывается, что он реализует эффективное распределение ресурсов между агентами. Раздел 3 посвящен описанию процесса итеративных переговоров на основе предлагаемого механизма, который предполагается применять в условиях, когда каждый агент может не обладать полной информацией о функциях полезности всех претендентов на ресурс. В разделе 4 исследуется построение «редуцированной» версии предложенного механизма, в которой каждый агент будет сообщать только то, сколько ресурса он хочет получить сам. Основные результаты статьи иллюстрируются примерами, базирующимися на одной постановке задачи распределения ресурсов, которая также лежит в основе описанной в разделе 5 деловой игры, предназначенной для экспериментальной апробации разработанного механизма. Кроме того, в разделе 5 приведены результаты нескольких пробных игр.

2. Постановка задачи и основные определения

Формально, задача распределения ресурсов записывается следующим образом. *Организационная система* состоит из одного *центра* и множества $N = \{1, \dots, n\}$ *агентов*. У центра

имеются ресурсы в ограниченном количестве – $R \in \mathbb{R}_+^1$, которые могут быть распределены между агентами в любой пропорции.

Полезность каждого агента $i \in N$ относительно количества выделяемых ему ресурсов $x_i \in [0, R]$ определяется функцией $u_i(\bullet) : \mathbb{R}_+^1 \rightarrow \mathbb{R}^1$, принадлежащей некоторому множеству допустимых функций полезности U_i .

Обозначим множество допустимых распределений ресурса как

$$A = \{x = (x_1, \dots, x_n) : \sum_{i \in N} x_i \leq R, x \in \mathbb{R}_+^n\},$$

множество возможных профилей полезности агентов как

$$U = \{u = (u_1(\bullet), \dots, u_n(\bullet)) : u_i(\bullet) \in U_i, i \in N\}.$$

«Базовая» задача заключается в нахождении такого отображения $g(\bullet) : U \rightarrow A$, которое является утилитарно эффективным, т.е. максимизирует суммарную полезность всех агентов от распределенного ресурса для любого из возможных профилей полезности $u \in U$:

$$(1) \quad g(u) \in \underset{x \in A}{\text{Arg max}} \sum_{i \in N} u_i(x_i).$$

Однако даже если решение (1) существует, может оказаться, что оно *манипулируемо* [11] (или *несовместимо со стимулами*, например [27]). То есть $\exists u \in U$ и $\exists k \in N$ что найдется профиль полезности $\tilde{u} = (\tilde{u}_k, u_{-k}) \in U$ такой, что

$$u_k(g_k(\tilde{u})) > u_k(g_k(u)),$$

где u_{-k} – профиль полезности всех агентов за исключением k ; $u = (u_k, u_{-k})$, а $g_k(u)$ – количество ресурса, выделяемое агенту k при профиле полезности u . В рамках данной статьи будем рассматривать следующее множество профилей полезности \hat{U} :

1) функция полезности любого агента строго вогнута, не убывает и дважды непрерывно дифференцируема;

2) $\forall u \in \hat{U}$ решение задачи (1) является внутренним.

Очевидно, что в рамках данных предположений решение задачи (1) будет существовать и будет единственным, что позволит сосредоточиться на проблеме совместимости со стимулами. В качестве иллюстрации рассмотрим следующий пример.

Пример 1. Пусть полезность каждого из агентов описывается функцией вида¹

$$(2) \quad u_i(x) = \sqrt{r_i + x_i}, \quad i \in N,$$

где r_i – собственные «резервы» агента i , известные лишь ему. Максимум суммарной полезности всех агентов будет достигаться при выделении каждому агенту ресурса в количестве

$$x_i = (R + \sum_{i \in N} r_i) / n - r_i, \quad i \in N.$$

То есть для решения задачи (1) от агента необходимо получить информацию о значении r_i . Очевидно, что если агента спросить о значении r_i , то ему будет выгодно не сообщать правду, а занижить сообщаемое значение. Что и означает, что полученное правило эффективного распределения ресурсов несовместимо со стимулами агентам достоверно раскрывать информацию о своей функции полезности. ■²

Будем называть *механизмом* набор $\rho = \langle S, \pi, t \rangle$, где $S = \times_{i \in N} S_i$ – некоторое множество допустимых действий агентов; $\pi(\bullet): S \rightarrow A$ – некоторая процедура, отображающая действия агентов в множество допустимых распределений ресурсов; $t(\bullet): S \rightarrow \mathbb{R}^n$ – некоторая процедура *трансфера* полезностей агентов. Обозначим $\Gamma(\rho) = \langle N, S, \varphi_{u, \rho} \rangle$ – игру, индуцированную механизмом, где $\varphi_{u, \rho} = \{\varphi_1, \dots, \varphi_n\}$ – профиль предпочтений агентов, определяемый на основе их профиля полезности $u \in U$ и процедур $\pi(\bullet)$ и $t(\bullet)$:

$$\varphi_i(s) = u_i(\pi(s)) - t_i(s), \quad i \in N.$$

В рамках данной статьи рассматривается следующая постановка задачи распределения ресурсов: возможно ли найти механизм, который позволит *реализовать по Нэшу* эффективное распределение ресурсов в случае, если решение задачи (1)

¹ Выбор постановки задачи распределения ресурсов, которая используется для иллюстрации основных результатов статьи – числа агентов, вида и параметров функций полезности агентов и параметров механизма, будет объяснен в разделе 6, где описывается постановка деловой игры, разработанной для экспериментальной апробации разработанного механизма.

² Здесь и далее будем подобным образом обозначать конец примеров.

несовместимо со стимулами – т.е. $\forall u \in U$ в игре будет единственное равновесие Нэша $s^*(u) \in S$:

$$\forall i \in N, \forall \tilde{s}_i \in S_i \varphi_i(s^*(u)) > \varphi_i(\tilde{s}_i, s_{-i}^*(u)),$$

такое, что $\pi(s^*(u)) = g(u)$.

Кроме того, сам механизм можно считать эффективным, только если $\sum_{i \in N} \varphi_i(s^*(u)) = \sum_{i \in N} u_i(g(u))$. Что подразумевает сбалансированность платежей:

$$\sum_{i \in N} t_i(s^*(u)) = 0.$$

3. Применение механизма Гровса–Лейдярда для решения задачи распределения ресурсов

Рассмотрим механизм $\rho = \langle S, \pi, t \rangle$. Каждый агент $i \in N$ сообщает, каким он видит распределение ресурсов в системе, причем от любого из агентов он может потребовать «пополнить» ресурс. Требуется лишь, чтобы предлагаемое распределение удовлетворяло первоначальному ресурсному ограничению:

$$S_i = \{s_i \in \mathbb{R}^n : \sum_{j \in N} s_{ji} \leq R\}, \quad S = \times_{i \in N} S_i,$$

где s_{ji} – заявка агента i о том, какое количество ресурса он считает нужным выделить агенту j .

Процедура $\pi(s) = \{x_1(s), \dots, x_n(s)\}$, определяющее распределение ресурсов, усредняет заявки всех агентов:

$$(3) \quad x_i(s) = \frac{1}{n} \sum_{j=1}^n s_{ij}, \quad i \in N.$$

Трансферы агентов определяются следующим образом. С каждого из агентов взимается штраф за «разногласие», определяемый следующим образом:

$$(4) \quad p_i(s) = \beta \sum_{j=1}^n (x_j(s) - s_{ji})^2, \quad i \in N.$$

Параметр $\beta \geq 0$ можно трактовать как силу штрафов. Часть $\alpha \in [0, 1]$ всех собранных штрафов возвращается агентам – делится поровну между ними, поэтому полный трансфер записывается следующим образом:

$$(5) \quad t_i(s) = p_i(s) - \frac{\alpha}{n} \sum_{j=1}^n p_j(s), \quad i \in N,$$

Нераспределенная часть штрафов «сжигается» (или остается у лица, распределяющего ресурс).

Параметр α удобно трактовать как *балансирующий коэффициент*, так как при $\alpha = 1$ трансферы всегда сбалансированы:

$$\forall s \in S \quad \sum_{i=1}^n t_i(s) = 0.$$

Кроме того, при $\alpha = 1$ механизм можно трактовать как «квадратичное правило» Гровса-Лейдярда [12, 19], применяемое к n задач определения объема производства коллективного блага при отсутствии затрат на производство этих коллективных благ и наличии ограничения на суммарный объем производства ($\sum_{i \in N} x_i \leq R$), связывающий эти задачи¹.

Наличие совместных ограничений не позволяет сразу утверждать о применимости результатов, полученных при исследовании «квадратичного правила» Гровса-Лейдярда для рассматриваемой нами задачи, и требует дальнейших теоретических исследований.

Опишем некоторые очевидные, но полезные для дальнейшего исследования свойства данного механизма и индуцированной им игры с функциями предпочтений агентов

$$\varphi_i(s) = u_i(\pi(s)) - t_i(s), \quad i \in N.$$

Обозначим $s_{-i} \in S_{-i}$ – *обстановку* для агента $i \in N$, $S_{-i} = \times_{j \in N \setminus \{i\}} S_j$. Обозначим $br_i(s_{-i}) \in S_i$ – *функцию наилучшего ответа* агента:

$$br_i(s_{-i}) \in \underset{s_i \in S_i}{\text{Arg max}} \varphi_i(s_i, s_{-i}).$$

Тогда можно записать следующее утверждение:

¹ В Приложении к статье приведено пояснение данного утверждения.

Лемма 1. Пусть $n > \alpha + 1$. Тогда $\forall u \in \widehat{U}$, $\forall i \in N$, $\forall s_{-i} \in S_{-i}$ функция предпочтения $\varphi_i(s_i, s_{-i})$ вогнута и $\exists! br_i(s_{-i}) = \arg \max_{s_i \in S_i} \varphi_i(s_i, s_{-i})$.

Доказательство леммы 1 и других утверждений приведены в Приложении.

Так как в исследуемой игре множества S_i – выпуклые подмножества \mathbb{R}^n , а функции $\varphi_i(s_i, s_{-i})$ в соответствии с леммой 1 вогнуты по s_i и, по построению, непрерывны по s_{-i} , то в этой игре существуют равновесия по Нэшу, см., например, [4].

Важным следствием из леммы 1 является тот факт, что при $n = 2$ принудительная балансировка трансферов невозможна, так как допустимы только значения $\alpha < 1$. Кроме того, если исследовать зависимость $br_i(s_{-i})$ от параметров α и β , которую будем обозначать как $br_i(s_{-i}, \alpha, \beta)$, то $\forall i \in N$, $\forall s_{-i} \in S_{-i}$ $br_i(s_{-i}, \alpha, \beta) = br_i(s_{-i}, 0, \beta)$, где

$$\tilde{\beta} = \beta \frac{n - \alpha - 1}{n - 1}.$$

Иными словами, функции наилучших ответов агентов для механизма с параметрами α и β эквивалентны функциям наилучших ответов для механизма с параметрами $\tilde{\alpha} = 0$ и $\tilde{\beta}$, при которых трансферы (5) агентов определяются более простой формулой:

$$t_i(s) = \tilde{\beta} \sum_{j=1}^n (x_j(s) - s_{ji})^2.$$

Обозначим $x_{j-i} = \frac{1}{n-1} \sum_{k \in N \setminus \{i\}} s_{jk}$ – количество ресурса, кото-

рое предлагают выделить агенту $j \in N$ все агенты за исключением агента $i \in N$. Очевидно, что если агент i согласится с «мнением» остального «общества» относительно выделяемого агенту j ресурса, т.е. $s_{ji} = x_{j-i}$, то в соответствии с процедурой $\pi(s)$ получится, что $x_j = x_{j-i}$.

Обозначим $\forall i, j \in N$ $\Delta_{ji} = br_{ji}(s_{-i}) - x_{j-i}$,

$A_i = \max\{0; \Delta_{ii} + \sum_{j \in N} x_{j-i} - R\}$. С учетом введенных обозначений,

функции наилучшего ответа $br_i(s_{-i}) = \{br_{ji}(s_{-i})\}_{j \in N}$, $i \in N$, определяются следующим утверждением.

Лемма 2. Пусть $n > \alpha + 1$. Тогда $\forall u \in \hat{U}$, $\forall i \in N$, $\forall s_{-i} \in S_{-i}$ Δ_{ii} определяется из решения уравнения¹

$$(6) \quad u'_i \left(\frac{1}{n} \Delta_{ii} + x_{i-i} \right) = 2\tilde{\beta} \frac{n-1}{n} ((n-1)\Delta_{ii} + A_i),$$

$$a \quad \forall j \in N \setminus \{i\} \quad \Delta_{ji} = A_j / (1-n).$$

Определив функции наилучших ответов агентов с помощью леммы 2, можно осуществить поиск равновесных по Нэшу сообщений агентов как неподвижных точек отображения $BR(\bullet) = \{br_i(s_{-i})\}_{i \in N} : S \rightarrow S$.

Кроме того, лемма 2 формально обосновывает следующее рациональное поведение агентов в индуцированной игре. Если агенту целесообразно просить для себя большее количество ресурса, чем ему предлагает общество, то возникающий от этого «дефицит» в рамках своей заявки агенту оптимально устранять, уменьшая предлагаемые обществом остальным агентам заявки на одинаковую величину – $\forall j \in N \setminus \{i\} \quad \Delta_{ji} = \Delta_{ii} / (1-n)$. При этом, если в соответствии с заявками остальных агентов, необходимо распределять весь ресурс ($\sum_{j \in N} x_{j-i} = R$), уравнение (6)

приобретает следующий вид:

$$(7) \quad u'_i \left(\frac{1}{n} \Delta_{ii} + x_{i-i} \right) = 2\tilde{\beta}(n-1)\Delta_{ii},$$

$$\text{и } \Delta_{ji} = \Delta_{ii} / (1-n), \quad j \in N \setminus \{i\}.$$

Введем обозначение $\Delta = \sum_{i \in N} s_{ii}^* - R$, которое можно интерпретировать как «дефицит» ресурса в системе – разницу между суммой того, что каждый агент просит выделить себе, и доступ-

¹ $u'_i(\bullet)$ обозначает частную производную $u_i(\bullet)$ по x_i , $u''_i(\bullet)$ – вторую частную производную по x_i .

ным количеством ресурса. Через $u_i'^{-1}(\bullet)$ обозначим функцию, обратную к $u'(\bullet)$.

Утверждение 1. $\forall u \in \widehat{U}$, $\forall \alpha \in [0, \min(1, n-1)]$, $\forall \beta > 0$ в игре $\Gamma(\rho)$ существует единственное равновесие Нэша $s^* \in S$, такое что

$$\pi(s^*) = \arg \max_{x \in A} \sum_{i \in N} u_i(x_i).$$

При этом s^* и $\pi(s^*)$ связаны следующими соотношениями:

$$\forall i \in N \quad x_i(s^*) = s_{ii}^* - \frac{\Delta}{n},$$

$$\forall j \in N \setminus \{i\} \quad x_j(s^*) = s_{ji}^* + \frac{\Delta}{n(n-1)},$$

где Δ является единственным решением уравнения

$$(8) \quad \sum_{i \in N} u_i'^{-1}(2\tilde{\beta}\Delta) = R.$$

Таким образом, утверждение 1 показывает, что исследуемый нами механизм ρ обеспечивает решение задачи (1) как равновесие Нэша в индуцированной им игре агентов $\Gamma(\rho)$, которое единственно, и дает понимание того, как связаны равновесные заявки агентов с параметрами механизма и решением задачи (1).

Из утверждения 1 можно получить следующие свойства предложенного механизма. В первую очередь, определим трансферы агентов (5) в равновесии:

$$(9) \quad t_i(s^*) = \tilde{\beta}(1-\alpha) \frac{\Delta^2}{n(n-1)}, \quad i \in N.$$

Из (9) очевидным образом следует, что трансферы всех агентов при любых значениях параметров механизма одинаковы. А в сбалансированном механизме ($\alpha = 1$) в равновесии трансферы отсутствуют. Это можно трактовать следующим образом: механизм ρ позволяет использовать трансферы как «угрозу», без необходимости их осуществления при достижении эффективного распределения ресурсов. Более того, один из основных недостатков механизма Гровса–Лейдярда при применении его в задаче определения уровня производства коллективного блага состоял в том, что решение могло оказаться инди-

видуально нерациональным для отдельных агентов (см., например [19]). Отсутствие трансферов в случае применения сбалансированного механизма к рассматриваемой нами задаче означает, что если оптимальное распределение ресурсов является индивидуально рациональным, то и решение игры будет индивидуально рациональным для всех агентов.

Однако обеспечить максимум суммарной полезности агентов при применении механизма можно только при $\alpha = 1$ ¹.

Следствие 1. $\forall u \in \hat{U} \sum_{i=1}^n t_i(s^*) = 0$ тогда и только тогда, когда $\alpha = 1$.

Любопытным является также тот факт, что в несбалансированном механизме (при $\alpha < 1$) итоговый трансфер для каждого агента уменьшается с ростом силы штрафов, так как (9) можно записать как

$$t_i(s^*) = \frac{(1 - \alpha)}{\beta n(n - \alpha - 1)} u_i^2(x_i(s^*)),$$

откуда видно, что для заданных N , $u \in \hat{U}$ и R размеры трансферов зависят только от α и β . Причем они обратно пропорциональны последнему параметру – силе штрафов. То есть, увеличивая силу штрафов, трансферы агентов также можно сделать сколь угодно малыми.

Таким образом, предложенный механизм позволяет реализовать эффективное распределение ресурсов, как единственное равновесие Нэша, и при сбалансированных платежах является эффективным.

Проиллюстрируем предложенный механизм на решении задачи распределения ресурсов из примера 1.

Пример 2. Пусть 3 агента претендуют на ограниченный ресурс, доступный в количестве $R = 115$. Полезность каждого из агентов описывается функцией (2). Собственные резервы агентов заданы набором $r = \{1; 9; 25\}$, где r_i – резервы агента i , известные лишь ему.

¹ Если отказаться от требования возрастания функций полезности агентов, то платежи могут оказаться сбалансированными при $\alpha < 1$, если решение задачи (1) будет внутренним.

В этом случае эффективным будет распределение $x=\{49;41;25\}$, а полезность любого из агентов будет примерно 7,07.

Если применить предложенный механизм с параметрами $\alpha = 1$, $\beta = 0,0005$, то в индуцированной им игре агентов равновесными будут следующие заявки (с точностью до второго знака после запятой):

$$s_1 = \{96,14; 17,43; 1,43\},$$

$$s_2 = \{25,43; 88,14; 1,43\},$$

$$s_3 = \{25,43; 17,43; 72,14\}.$$

Усреднение этих заявок для каждого из агентов дает эффективное распределение ресурса $x=\{49;41;25\}$. То есть каждый из агентов просит себе примерно на 47,14 большее количество ресурса, чем получает, занижая свои заявки для остальных на 23,57.

В соответствии с (4), штраф за разногласие для любого из агентов составят $p_i \approx 1,67$, $i \in \{1;2;3\}$. Поэтому трансферы очевидным образом равны 0 и сбалансированы.

Увеличение силы штрафов в два раза ($\beta = 0,001$) приведет к уменьшению «разногласий» в два раза – каждый агент будет просить себе примерно на 23,57 большее количество ресурса, чем получает, занижая свои заявки для остальных на 11,785:

$$s_1 = \{72,57; 29,215; 13,215\},$$

$$s_2 = \{37,215; 64,57; 13,215\},$$

$$s_3 = \{37,215; 29,215; 48,57\}.$$

Очевидно, что распределение ресурсов будет также оптимальным – $x = \{49;41;25\}$. А штрафы за разногласие также уменьшатся вдвое – $p_i \approx 0,83 \quad \forall i \in \{1;2;3\}$. Трансферы агентов, опять же, будут равны 0 и останутся сбалансированными.

Если же оставить силу штрафов $\beta = 0,0005$, но «отменить» полностью балансировку – $\alpha = 0$, то заявки агентов будут в точности совпадать с описанными в предыдущем абзаце. Не изменятся и штрафы за разногласие, но трансферы агентов уже не будут сбалансированными – $t_i = p_i \approx 0,83 \quad \forall i \in \{1;2;3\}$. Суммарная полезность агентов уменьшится на 2,5. ■

4. Исследование сходимости процесса итеративных переговоров на основе предложенного механизма

Реализация равновесия Нэша требует от агентов полной информированности о параметрах индуцированной механизмом игры, что крайне редко встречается в практических задачах распределения ресурсов. Исследуем возможность применения механизма в условиях, когда каждый агент может знать только свою функцию полезности, доступное количество ресурса, общее число агентов и механизм. Для определения распределения ресурсов между ними используется следующий *итеративный процесс переговоров* $I\rho$ на основе предложенного механизма $\rho = \langle S, \pi, t \rangle$:

$$x(\tau) = \pi(s(\tau)), \quad \varphi_i(\tau) = u_i(x(\tau)) - t(s(\tau)),$$

где $s(\tau) = (s_1(\tau), \dots, s_n(\tau)) \in S$ – сообщения агентов на итерации $\tau \geq 1$. Процесс переговоров продолжается до такой итерации T , на которой агенты перестанут менять свои заявки: $s(T-1) = s(T)$.

Для ответа на вопрос, может ли сойтись данный процесс к эффективному распределению за конечное число итераций, необходимо сделать предположения о том, как принимает решения о своей заявке каждый из агентов на каждой из итераций процесса и исследовать его свойства, как свойства дискретного динамического процесса. Примером простейшей гипотезы о принятии решений агентами является *динамика Курно*, в рамках которой каждый агент выбирает свое действие как наилучший ответ на действия всех остальных агентов на предыдущей итерации, см., например [12, 28]:

$$s_i(\tau) = br_i(s_{-i}(\tau - 1)).$$

$$\text{Обозначим } |s_i(\tau)| = \sum_{j \in N} s_{ji}(\tau).$$

Лемма 3. *Если для некоторого $\tau \geq 1$ $s(\tau) \in S$ таково, что $|\pi(s(\tau))| < R$, то при поведении агентов в соответствии с динамикой Курно $|\pi(s(\tau + 1))| > |\pi(s(\tau))|$. Если же $|\pi(s(\tau))| = R$, то и $|\pi(s(\tau + 1))| = R$.*

То есть если в рамках динамики Курно $s(\tau)$ обеспечивает не полное распределение ресурса, то $s(\tau+1) = br(s(\tau))$ будет увеличивать количество распределяемого ресурса. Если же $s(\tau)$ распределяет ресурсы полностью, то $s(\tau+1) = br(s(\tau))$ также будет распределять ресурсы полностью.

Проведем анализ итеративного процесса в области $\bar{S} = \{s \in S : \forall i \in N |s_i| = R\}$. По сути, необходимо проверить, является ли равновесие Нэша притягивающей неподвижной точкой отображения $BR(\bullet)$ (см., например [1]):

$$\forall \{s, s'\} \in \bar{S}^2 \quad d(s, s') > d(BR(s), BR(s')),$$

где $d(s, s')$ – расстояние в \bar{S} (может быть выбрана произвольная метрика).

Соответственно, если механизм порождает игру, в которой отображение $BR(\bullet)$ сжимающее для некоторого $u \in \hat{U}$, то он также называется *сжимающим* для данного $u \in \hat{U}$ [20, 31]. Если механизм является сжимающим, то для целого ряда гипотез поведения агентов, включая динамику Курно, итеративный процесс при заданном профиле полезностей $u \in \hat{U}$ будет сходиться к $s^*(u)$. В случае нашего механизма ситуация немного сложнее. Обозначим $BR^2(\bullet) = BR(BR(\bullet)) : S \rightarrow S$ – «двойное отображение», построенное на основе $BR(\bullet)$.

Лемма 4. $\forall u \in \hat{U}$, такого, что $\forall s \in \bar{S}$ существует конечная $C \in \mathbb{R}_+^1 : \max_{i \in N} (-u_i''(x_i(s))) \leq C$, то найдутся такие

$\tilde{\beta} \geq \frac{1}{2n} \max_{i \in N} (-u_i''(x_i(s)))$, при которых $BR^2(\bullet)$ является сжимающим отображением.

Так как отображение $BR^2(\bullet)$ сжимающее, то оно имеет единственную неподвижную точку. Очевидно, что неподвижная точка отображения $BR(\bullet)$, которую мы нашли ранее, будет также и неподвижной точкой отображения $BR^2(\bullet)$. Поэтому можно сформулировать следующее утверждение.

Утверждение 2. Эффективное распределение ресурса реализуется в итеративном процессе I_r для любого профиля предпочтений из \hat{U} , такого, что существует конечная $C \in \mathbb{R}_+^1 : \max_{i \in N} (-u_i''(x_i(s))) \leq C$, а агенты действуют по динамике Курно.

К сожалению, результаты по стратегиям поведения агентов, отличным от динамики Курно [9, 12, 28], требуют дополнительной проверки для рассматриваемого механизма, так как механизм не является сжимающим в классическом определении. Однако результаты, полученные для динамики Курно, имеют самостоятельную ценность, потому что именно подобная динамика реализуется в алгоритмах распределенной оптимизации в мультиагентных системах по типу [15]. Исследуемый нами механизм может быть применен для решения аналогичных задач.

Кроме того, открытым остается вопрос о скорости сходимости итеративного процесса к равновесию Нэша. Из доказательства леммы 4 следует, что делать β очень большой не целесообразно: в этом случае в итеративном процессе переговоров агенты будут стремиться минимизировать значения штрафов – процесс будет достаточно быстро сходиться к среднему арифметическому стартовых заявок агентов, затем медленно двигаться в сторону эффективного распределения.

Проиллюстрируем полученные в данном разделе результаты на нескольких примерах.

Пример 3. Действие механизма при поведении агентов в соответствии с динамикой Курно.

Рассмотрим постановку задачи распределения из примера 1, взяв параметры механизма, рассмотренные в примере 2: $\alpha = 1$, $\beta = 0,0005$. Как видно из примеров 1 и 2, при данных параметрах штрафы за разногласие в равновесии ($\approx 1,67$) в некотором смысле «сопоставимы» с той полезностью, которую получает любой из агентов при эффективном распределении ресурсов – $\approx 7,07$. Рис. 1 демонстрирует динамику выигрышей (который могли бы получить агенты, если бы переговоры закончились бы на этой итерации) и заявок агентов (про количество ресурса для агента 1) в случае, если на первой итерации каждый агент попросил отдать весь ресурс ему.

Параметры $\alpha = 1$, $\beta = 0,0005$ удовлетворяют условию леммы 4 той части области S , где любой из агентов получает не менее $31 - r_i$ единицы ресурса.

Начиная с итерации 8, получаемый каждым агентом ресурс отличается от оптимального не более чем на 1. Оптимальное

распределение достигается только на 39-й итерации. На любой итерации после 39-й агенты не меняют своих заявок. ■

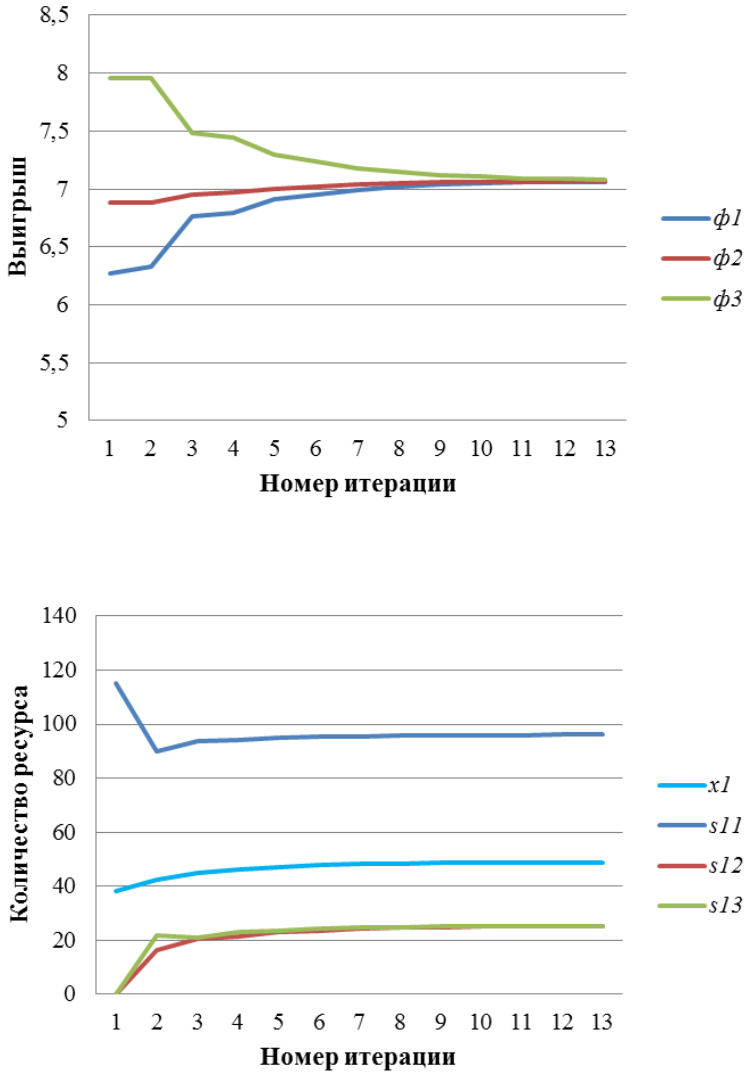


Рис. 1. Выигрыши агентов и их заявки про ресурс для агента 1

Динамика Курно, как уже было сказано выше, одна из базовых моделей принятия решений. И если для искусственных мультиагентных систем данная динамика может быть «запрограммирована» априори [15], то принятие решений реальными людьми крайне редко может быть описано в рамках данной динамики [17]. Более того, ожидать, что рациональный субъект может следовать данной модели принятия решений можно лишь в том случае, если она будет увеличивать его выигрыш.

Приведем пример модели поведения рационального поведения агентов, в которой эффективное распределение ресурсов не может быть достигнуто в процессе итеративных переговоров.

Пример 4. *Отказ агентов от динамики Курно.*

Если в модели, использованной в предыдущих примерах, ресурс между агентами делится поровну, то агент 3 ($r_3 = 25$) получает полезность $\approx 7,95$, т.е. больше, чем при эффективном распределении ресурса – $\approx 7,07$. Агенты 1 ($r_1 = 1$) и 2 ($r_2 = 9$) получают при этом меньшую полезность, чем при эффективном распределении ресурсов. То есть агент 3 предпочел бы, чтобы ресурс делился поровну по эффективному распределению, в то время как агенты 1 и 2 – наоборот.

Рассмотрим динамику заявок агентов, полученную в примере 3, в которой на первой итерации каждый агент попросил отдать весь ресурс себе. Из (3) следует, что в этом случае ресурс между ними будет поделен поровну, а из (5) – что трансфер каждого агента будет равен 0. Поэтому выигрыш каждого из агентов будет в точности равен полезности от полученного ресурса.

Если агент 3 действует по динамике Курно, выбирая на каждой итерации свою заявку, как наилучший ответ на обстановку на предыдущей итерации, то его выигрыш в игре (полезность минус трансферы) будет уменьшаться с каждой итерацией, за исключением первой (см рис. 1). Более того, выигрыш, который он может ожидать, выбирая наилучший ответ, будет всегда меньше, чем тот, который реализуется на соответствующей итерации. В отличие от двух других агентов, чьи выигрыши будут возрастать.

Вот почему у агента 3 может возникнуть мотивация отказаться следовать динамике Курно как стратегии своего поведе-

ния. В частности, он может не уменьшать то количество ресурса, которое он просит себе, все время требуя отдать весь ресурс ему.

На рис. 2 приведены графики изменения выигрышей агентов и ресурса, который будет получать агент 1 для случая, если агент 3 решил не менять заявку $s_{33} = R$ на всех итерациях, но минимизирует свой трансфер за счет выбора заявок s_{13} и s_{23} в соответствии с динамикой Курно.

Агенты 1 и 2 действуют в соответствии с динамикой Курно до 10-й итерации. На второй итерации выигрыш агента 3 значительно уменьшается, а выигрыши агентов 1 и 2 растут. Но, начиная с 3-й итерации и вплоть до 15-й, выигрыши агентов 1 и 2 оказываются меньше, чем на итерации 1. Более того, начиная с 13-й итерации заявки агентов 1 и 2 оказываются «равновесными» в том смысле, что каждому из них невыгодно от них отклоняться, при условии, что агент 3 не будет менять свою заявку¹. Для агента 3 сообщение $s_{33} = R$ не является компонентой его наилучшего ответа на всех итерациях вплоть до 14-й: $R \neq br_{33}(s_{-3}(\tau - 1))$.

На 15-й итерации агент 2 отказывается от модели поведения по динамике Курно. Мотивацией данного отказа может служить тот факт, что выигрыш его все время убывал, в отличие от агента 1, чей выигрыш пускай и незначительно, но возрастал. А, как было упомянуто выше, после 13-й итерации заявки агентов перестали меняться. Поэтому на 15-й итерации агент 2 меняет свою заявку про тот ресурс, который он просит себе на стартовую — $s_{22}(15) = R$. И далее не меняет s_{22} , выбирая в соответствии с динамикой Курно лишь компоненты заявки s_{12} и s_{32} , минимизируя свой трансфер. На итерации 15 он несколько теряет в выигрыше, но начиная с 16-й итерации и далее агент 2 получает больший выигрыш, чем при выборе им стратегии наилучших ответов на ранних итерациях.

Начиная с итерации 15, только агент 1 выбирает свою заявку как наилучший ответ на заявки оппонентов. При этом агенты 2 и 3 просят для него отрицательное количество ресурса, минимизируя свои трансферы.

¹ При моделировании с точностью 10^{-3} .

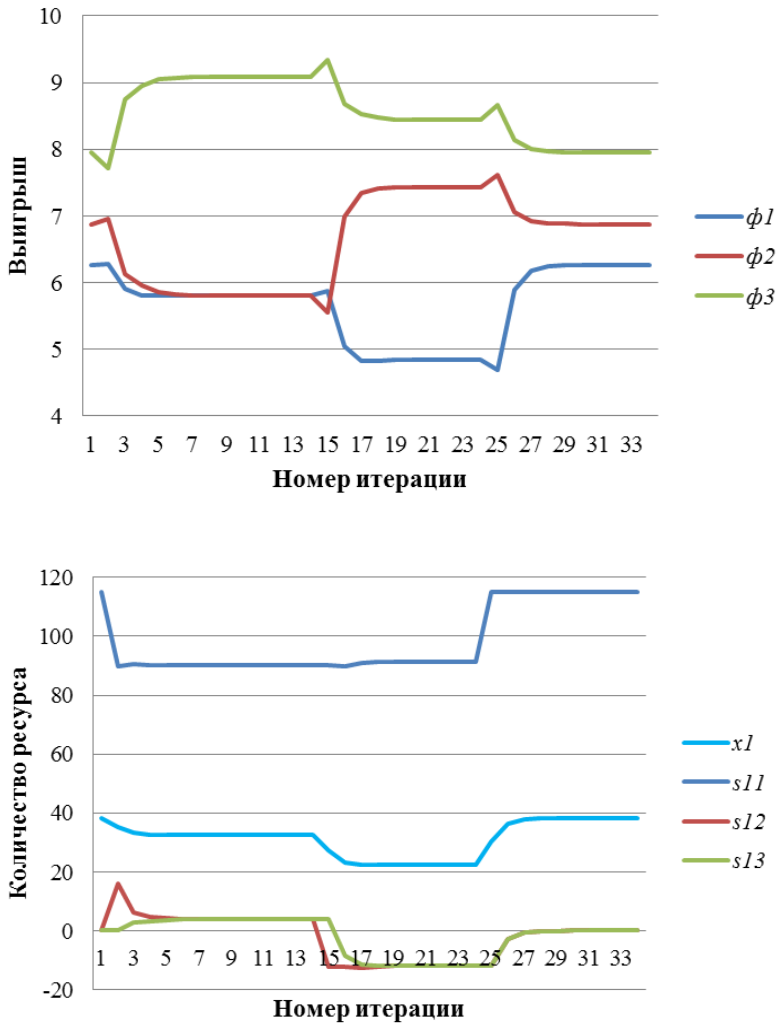


Рис. 2. Выигрыши агентов и их заявки про ресурс для агента 1 при последовательном отказе агентов следовать динамике Курно

Но, начиная с 20-й итерации тот ресурс, что агент 1 может получить, как и его наилучший ответ, перестают сильно менять-

ся, а его выигрыш оказывается $\approx 4,85$. Начиная с 23-й итерации в рамках рассматриваемой нами точностью агент 1 не может улучшить свой выигрыш, меняя свои заявки в рамках динамики Курно. Если агент 1 откажется следовать динамике Курно, то он может увеличить свой выигрыш. В нашем примере на итерации 25 он перестает следовать наилучшим ответам и просит отдать весь ресурс себе. И далее не меняет s_{11} , выбирая в соответствии с динамикой Курно лишь компоненты заявки s_{21} и s_{31} , минимизируя свой трансфер. В этом случае, если все агенты не меняют заявки про ресурс для себя, но выбирают заявки про остальных, минимизируя свой трансфер, то начиная с итерации 34 заявки, выигрыши всех агентов и выделяемый им ресурс оказываются такими же, как и на первой итерации.

На всех итерациях, кроме 2-й, выигрыш агента 3 оказывается больше, чем при эффективном распределении ресурсов, поэтому выбранная им стратегия может считаться рациональной (точнее ограниченно рациональной). При этом эту стратегию поведения можно охарактеризовать, как «несговорчивую».

Суммарный выигрыш агентов меньше максимального, что естественно. ■

Действуя в рамках «робастного» подхода, можно утверждать, что пример 4 иллюстрирует тот факт, что итеративный переговорный процесс не гарантирует эффективного распределения ресурсов при достаточно рациональных гипотезах поведения агентов.

5. Уменьшение размерности пространства сообщений агентов

Предлагаемый механизм требует от агентов сообщения полного вектора распределения ресурсов. Что может быть затруднительным на практике, особенно если число агентов велико. Кроме того, агенты могут использовать возможность сообщать полный вектор для кооперации друг с другом, сообщая скоординированную заявку. В частности, возможны ситуации, в которых сообщество разделяется на две группы. Тогда их взаимодействие можно рассматривать как игру двух агентов, что при сбалансированных платежах может не позволить реализовать эффективное распределение, так как в этом случае окажет-

ся, что $\tilde{\beta} = 0$ и у двух групп будут отсутствовать стимулы договариваться. Поэтому целесообразным представляется исследование возможности исключения кооперации между агентами за счет координации своих заявок.

Полученные в разделе 3 результаты по виду функций наилучших ответов агентов мотивируют исследовать возможность построения модификации исследуемого механизма, в которой каждый агент будет сообщать лишь то, сколько ресурса хочет получить лично он – $\hat{\rho} = \langle \hat{S}, \hat{\pi}, \hat{t} \rangle$, где $\hat{S}_i \subseteq \mathbb{R}$.

Сохраним обозначение заявки агента s_{ii} , $i \in N$.

Из утверждения 1 следует, что в равновесии каждый агент по процедуре $\hat{\pi}: \mathbb{R}^n \rightarrow X$ должен получать ресурс в количестве

$$(10) \hat{x}_i = s_{ii} - \frac{\sum_{i \in N} s_{ii} - R}{n}.$$

Предположим, что таким же образом ресурс будет распределяться для всех возможных заявок агентов, таких что $\sum_{i \in N} s_{ii} - R \geq 0$. В случае отсутствия дефицита каждый агент будет получать, сколько просит: $\forall i \in N \ x_i = s_{ii}$.

С каждого агента будем брать платеж $\hat{t}_i = \hat{\beta}(s_{ii} - x_i)^2$. Очевидно, что платежи всех агентов будут одинаковы, поэтому балансировка платежей недопустима – в противном случае реальный платеж каждого агента всегда будет нулевым. Справедливы следующее утверждения.

Лемма 5. *Механизм $\hat{\rho} = \langle \hat{S}, \hat{\pi}, \hat{t} \rangle$ реализует эффективное распределение ресурсов как единственное равновесие Нэша в индуцированной им игре агентов $\Gamma(\hat{\rho})$.*

Лемма 5 позволяет сформулировать утверждение об эквивалентности между механизмами $\rho = \langle S, \pi, t \rangle$ и $\hat{\rho} = \langle \hat{S}, \hat{\pi}, \hat{t} \rangle$. Эквивалентными считаются механизмы (распределения ресурсов), которые для любого профиля предпочтений агентов реализуют одинаковое распределение ресурсов.

Утверждение 3. *Пусть механизм $\rho = \langle S, \pi, t \rangle$ задан параметрами $\beta > 0$, $\alpha < 1$. Механизм $\hat{\rho} = \langle \hat{S}, \hat{\pi}, \hat{t} \rangle$, в котором*

$$\hat{\beta} = \beta \frac{n - \alpha - 1}{n(n - 1)^2}:$$

1) эквивалентен механизму $\rho = \langle S, \pi, t \rangle$;

2) равновесные заявки агентов в механизме $\hat{\rho} = \langle \hat{S}, \hat{\pi}, \hat{t} \rangle$ совпадают с равновесными заявками про собственный ресурс в механизме $\rho = \langle S, \pi, t \rangle$.

Таким образом, для отдельных параметров механизма $\rho = \langle S, \pi, t \rangle$ оказывается возможным построение эквивалентного ему механизма $\hat{\rho} = \langle \hat{S}, \hat{\pi}, \hat{t} \rangle$, в котором все агенты сообщают только заявку на ресурс для себя. Однако балансировка этого механизма невозможна, поэтому, в отличие от механизма $\rho = \langle S, \pi, t \rangle$, его нельзя считать эффективным. Но для него справедлива аналогичная зависимость абсолютных платежей агентов от силы штрафов:

$$\hat{p}_i(s^*) = \frac{n(n-1)}{\hat{\beta}} \frac{\partial u_i}{\partial x_i} (\hat{x}_i(s^*)).$$

То есть имеется возможность сделать платежи агентов сколь угодно малыми, что позволяет трактовать его, по аналогии с [29], как почти эффективный.

Однако из-за ненулевых платежей в механизме $\hat{\rho} = \langle \hat{S}, \hat{\pi}, \hat{t} \rangle$ нельзя гарантировать, что будет обеспечена индивидуальная рациональность решения для всех агентов.

В тоже время на основе итеративного процесса переговоров $I\rho$ можно предложить «редуцированный» итеративный процесс переговоров $\hat{I}\rho$, в котором на каждой итерации у агентов спрашиваются только их заявки про ресурс для себя – $s_{ii}(\tau)$, $i \in N$. Причем $\forall i \in N \ s_{ii}(1) \leq R$. При этом заявки агента о том, какое количество ресурса следует выделять каждому из остальных агентов, определяются следующим образом.

На первой итерации, если поданные заявки агентов не могут быть удовлетворены, то считается, что агент предлагает весь оставшийся после удовлетворения его заявки ресурс поделить поровну между остальными агентами. То есть если

$$\sum_{i \in N} s_{ii}(1) > R,$$

$$\text{то } \forall i \in N, \ \forall j \in N \setminus \{i\} \ s_{ji}(1) = \frac{R - s_{ii}}{n-1}.$$

Если в системе нет дефицита ресурсов, то считается, что все агенты согласны с поданными заявками. То есть при

$$\sum_{i \in N} s_{ii}(1) \leq R$$

$$\forall i \in N, \forall j \in N \setminus \{i\} \quad s_{ji}(1) = s_{ij}(1).$$

Для любой итерации $\tau > 1$ для каждого из агентов его заявки про количество ресурса для каждого из остальных агентов рассчитываются как его наилучший ответ в игре $\Gamma(\rho)$ на обстановку на предыдущей итерации:

$\forall j \in N \setminus \{i\} \quad s_{ji}(\tau) = br_{ji}(s_{-i}(\tau - 1))$, предполагая, что

$$br_{ii}(s_{-i}(\tau - 1)) = s_{ii}(\tau).$$

То есть при

$$s_{ii}(\tau) + \sum_{j \in N \setminus \{i\}} x_{j-i}(\tau - 1) < R$$

$$s_{ji}(\tau) = x_{j-i}(\tau - 1).$$

Иначе

$$s_{ji}(\tau) = x_{j-i}(\tau - 1) - \frac{1}{n-1} A_i,$$

где $A_i = s_{ii} + \sum_{j \in N \setminus \{i\}} x_{j-i} - R$.

В рамках данного итеративного процесса переговоров динамика Курно подразумевает, что каждый агент будет выбирать свою заявку на каждой итерации как решение уравнения (6), беря в качестве обстановки заявки всех агентов про количество ресурса для него $s_{-i}(\tau - 1)$.

Очевидно, что если итеративный процесс $I\rho$ обеспечивает сходимость к равновесию Нэша в игре $\Gamma(\rho)$ в предположении, что агенты действуют в соответствии с динамикой Курно, то и $\hat{I}\rho$ также обеспечит сходимость к этому равновесию.

Предлагаемый итеративный процесс $\hat{I}\rho$ исключает возможность кооперации агентов путем координации заявок, так как каждый агент может выбирать заявку «про себя», а механизм будет перераспределять создаваемый этой заявкой дефицит поровну между всеми другими агентами. Но он остается уязвимым к поведению, иллюстрируемому рис. 2 – когда некоторые агенты (или все) не действуют по динамике Курно. Более того, для ситуации, рассмотренной в примере 4, для агента 3 не существует угрозы кооперации агентов 1 и 2.

6. Экспериментальная апробация

Для экспериментальной апробации полученных теоретических результатов и результатов компьютерного моделирования была разработана деловая игра и информационная система для ее проведения на основе программы zTree [16].

В данном разделе описывается серия игр, проведенная с целью проверки предлагаемого механизма ρ в максимально свободных условиях и с различными участниками:

– в системе были реализованы одновременно оба итеративных процесса переговоров $I\rho$ и $\dot{I}\rho$, игроки могли свободно выбрать между ними в процессе игры.

– игрокам не запрещалось общаться в реальности во время проведения экспериментов (игр).

Игрокам предлагалась следующая ситуация, приближенная к реальности: игроки-студенты делят между собой время на консультацию у преподавателя; выигрыш каждого игрока является оценкой за экзамен и монотонно зависит от полученного времени консультации, определяясь функцией (2), где тип i -го игрока r_i определял его начальные «знания», известные лишь ему.

Параметры игры подбирались таким образом, чтобы выигрыш каждого игрока в Парето-оптимуме составлял примерно 7 баллов по десятибалльной системе (4 по пятибалльной). Для получения большего балла (оценки отлично) игроку нужно было «выторговать» большее количество ресурсов, чем в оптимальном распределении.

Порядок проведения игр был следующий:

1. Теоретическое объяснение игры и применяемого механизма распределения ресурса.
2. Обучающая игра с возможностью обсудить с ведущим непонятную ситуацию.
3. Серия реальных игр;
4. Объявление результатов.

Организаторы игры разбивали имеющийся коллектив на группы (по 3 или по 5 человек), каждая группа играла свою игру. Если коллектив мог быть разбит на несколько групп, то

перед каждой следующей игрой разбиение коллектива на группы изменялось случайно. Номера (и соответственно типы) игрокам назначались случайно перед *каждой* игрой и в течение одной игры не менялись. Таким образом, если человек играл в нескольких играх, его тип мог меняться от игры к игре.

Среди участников игр были студенты МФТИ (факультеты РТК и ИБС) и сотрудники ИПУ РАН, при этом игры проводились по отдельности: со студентами ФРТК МФТИ (бакалавриат), со студентами ФИБС МФТИ (магистратура) и с сотрудниками ИПУ РАН.

Игрокам также раздавались распечатанные инструкции к игре и к информационной системе проведения игры. Полный текст инструкции игры доступен по адресу <http://www.mtas.ru/games/gl>.

Исходной, «Базовой», являлась игра, описанная в примере 2: число игроков $n = 3$, распределяемое время консультаций $R = 115$, собственные знания (типы агентов) принадлежат набору $r = \{1; 9; 25\}$, причём каждый тип может принадлежать только одному игроку. Параметры механизма $\alpha = 1$ и $\beta = 0,0005$ были выбраны по следующим соображениям. Во-первых, штрафы игроков в равновесии ($\approx 1,67$, см. пример 2) были сопоставимы с той полезностью, которую получают игроки получают при оптимальном распределении ресурсов ($\approx 7,07$, см. пример 2). Во-вторых, данные параметры механизма удовлетворяют требованиям утверждения 2 в значительной области возможных значений вектора распределения ресурсов – см. пример 3.

Каждый игрок знал все перечисленные выше параметры, но не знал, какими именно типами из набора обладают другие игроки. Кроме того, априори игрокам не было известно, кто именно из других участников эксперимента играет с ним в одной игре. Однако с учетом того, что общение между участниками игр не ограничивалось, данную информацию игроки могли получить в ходе игры.

Игра (в группе) заканчивалась, если все игроки перестали менять свои заявки или было достигнуто 100 итераций ($\tau \leq 100$, игра заканчивается *после* совершения 100-й итерации). При этом выигрыш игрока определяется как его выигрыш в игре на последней итерации.

Хотя динамика Курно не приводит игроков к ситуации выдачи кому-либо отрицательного ресурса, в реальной ситуации нельзя исключить такой возможности, поэтому была реализована система штрафов, при которой сильно (10 000 баллов) штрафуются игрок, получивший отрицательное количество ресурса и тот игрок, заявка которого оказала решающее влияние на это.

На каждой итерации игрок мог выбирать между действием по $I\rho$, определяя самостоятельно все компоненты своей заявки (полный вектор распределения ресурсов) или по $\hat{I}\rho$, определяя лишь компоненту заявки про собственный ресурс, делегируя системе выбор других компонент заявки.

Кроме «Базовой» игры в эксперименте использовались следующие ее модификации:

1. «5 игроков», в которой $r = \{1; 9; 25; 1; 9\}$ и $R = 205$.
2. «Без балансировки»: $\alpha = 0$.

В таблице 1 приведено общее число игр каждого типа и общее число итераций.

Таблица 1. Типы игр

Тип игры	Комбинация параметров (α, β, n)	Количество игр	Общее количество итераций
Базовая	(1; 0,0005; 3)	6	553
5 игроков	(1; 0,0005; 5)	3	123
Без балансировки	(0; 0,0005; 3)	10	93

Общий итог данной серии экспериментов следующий.

1. Ни в одной из проведенных игр не наблюдалась динамика Курно.

2. Эффективное распределение ресурсов (а также почти) было достигнуто в 8 играх (в «Базовых» – 1, в «5 игроков» – 1, в «Без балансировки» – 6), причем во всех этих случаях эффективное распределение было достигнуто в результате полной кооперации агентов.

3. Кооперативное поведение наблюдалось в 13 играх (в «Базовых» – 3, в «5 игроков» – 3, в «Без балансировки» – 7), причем образование полной коалиции наблюдалось в 12 случаях. Игры,

в которых отсутствует кооперативное поведение, будем называть некооперативными, другие – кооперативными: (не)кооперативные «Базовые» игры и т.п. В одной игре образовалось две коалиции игроков (двое против одного в игре «Без балансировки»).

4. Среднее число итераций – 14,95, среднее число итераций по типам игр: «Базовые» – 33, «5 игроков» – 15, «Без балансировки» – 4,1».

Некооперативное поведение наблюдалось лишь в 6 играх – по три в «Базовых» и в «Без балансировки», но эти игры сильно отличаются даже по среднему количеству итераций, поэтому сравнивать их между собой не стоит.

Сравним средние по играм заявки игроков о себе на последних итерациях (в конце игры) с заявками игроков о себе, равновесными по Нэшу, т.е. насколько далеко средние заявки о себе отличались от равновесных по Нэшу в конце игр. В трёх некооперативных «Базовых» играх средние заявки о себе игроков типа 1, 2 и 3 представлены в таблице 2: жирным шрифтом выделены заявки игроков о себе, через знак «/» – равновесные по Нэшу заявки игроков о себе. Видно, что отличие от равновесной по Нэшу заявки существенное и разное по направлению для игроков разного типа. Так, заявки 1 и 2-го игроков для себя меньше равновесных, тогда как заявка 3-го игрока больше.

Таблица 2. Средние по играм заявки игроков на последней итерации «Базовых» некооперативных игр.

	Заявка игрока 1	Заявка игрока 2	Заявка игрока 3
Об игроке 1	63 / 72,6	42,8 / 37,2	29,6 / 37,2
Об игроке 2	32 / 29,2	50,3 / 64,6	12,05 / 29,2
Об игроке 3	20 / 13,2	21,9 / 13,2	61,7 / 48,6

Посмотрим, какие при этом на последних итерациях получаются распределения ресурсов в среднем в отличие от эффективного распределения и распределения поровну. По таблице 3 видно, что среднее распределение ресурса не строго приближается к эффективному из-за 2-го игрока, а в сумме даже меньше общего количества 115 единиц.

Таблица 3. Средние по играм получаемые ресурсы игроков на последней итерации «Базовых» некооперативных игр

	Ресурс игрока 1	Ресурс игрока 2	Ресурс игрока 3
Поровну	38 1/3	38 1/3	38 1/3
Среднее	45,1	31,5	34,5
Эффект	49	41	25

По таблице 3 видно, что второй игрок в среднем получает значительно меньше как ресурса поровну, так и ресурса при эффективном распределении, следующая таблица показывает, что это повлияло на его выигрыш. В таблице 4 представлены сравнения средних по играм выигрышей на последней итерации. При том, что заявки и выдаваемый ресурс достаточно далеки от равновесных, выигрыши игроков не слишком далеки от эффективных.

Таблица 4. Средние по играм выигрыши игроков на последней итерации «Базовых» некооперативных игр

	Выигрыш 1-го	Выигрыш 2-го	Выигрыш 3-го	Сумма
Поровну	6,3	6,9	7,9	21,1
Средний	6,8	6,5	7,4	20,7
Эффективный	7	7	7	21,2

Таким образом, интересно, что игроки 2-го типа выигрывают меньше, чем игроки 1-го и 3-го типов. Это же оказалось верно и для некооперативных игр «Без балансировки».

Для статистического подтверждения различия поведения игроков разных типов был проведён тест Kruskal–Wallis [25] для 3 групп: заявок о себе трёх типов игроков «Базовых» игр по всем итерациям. Значение теста составило величину порядка 10^{-5} при значении порога 0,05, что означает, что заявки о себе для разных типов игроков «Базовых» игр статистически различны.

Далее приведём графики средних заявок игроков о себе s_{ii} по всем итерациям для кооперативных и некооперативных «Базовых» игр. На рис. 3 сплошная линия – линия средних заявок от всех игр, остальные прерывистые линии – линии

средних заявок для конкретных игр, над графиком надписано общее среднее по всем итерациям (на рис. За также видно, что две игры закончились раньше 30 итераций).

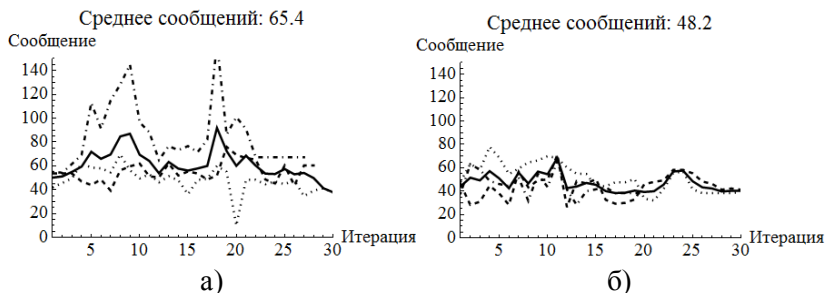


Рис. 3. Графики средних сообщений о себе для некооперативных (а) и кооперативных (б) «Базовых игр»

Заметим, что в некооперативных играх среднее заявок о себе больше, тогда как характер поведения отдельных игр различен. В играх «Без балансировки» среднее заявок о себе также выше в некооперативном случае. Возможно, это результат «борьбы» игроков за свой ресурс в некооперативных случаях.

Ненаблюдаемость динамики Курно в поведении реальных игроков в течение всей игры была вполне ожидаема. Но в ходе эксперимента мы пытались также получить ответ на вопрос, наблюдается ли данное поведение локально (в рамках отдельных итераций).

Для этого проанализируем заявки всех игроков на всех итерациях на предмет соответствия следующим моделям поведения:

- «Неподвижные заявки»;
- «Индикаторное поведение» [3];
- «Наилучший ответ».

Для пояснения данных моделей поведения введем следующие дополнительные понятия. Будем называть направлением вектора – вектор знаков его компонент, а направлением изменения вектора заявок $s_i(\tau)$ i -го игрока на итерации τ – направлением $\Delta_i(\tau) = s_i(\tau) - s_i(\tau - 1)$. Определим

$$s_i(\tau, \gamma) = s_i(\tau - 1) + \gamma [br_i(s_{-i}(\tau - 1)) - s_i(\tau - 1)]$$

К классу «Неподвижные заявки» относятся следующие формальные модели поведения:

- «Не двигается (C)»: $\{s_i(\tau) = s_i(\tau, 0)\}$,
- «Почти не двигается (C(0,1))»: $\{s_i(\tau) \mid s_i(\tau) = s_i(\tau, \gamma), -0,1 < \gamma < 0,1\}$, то есть относительное изменение вектора заявок игрока не превышает 1/10 расстояния до наилучшего ответа по каждой компоненте.

К классу «Индикаторное поведение»:

- «Индикаторное поведение (IB)»: $\{s_i(\tau) \mid s_i(\tau) = s_i(\tau, \gamma), 0 < \gamma \leq 1\}$, это классическое определение индикаторного поведения [3],
- «В сторону наилучшего ответа (IB+)» $\{s_i(\tau) \mid s_i(\tau) = s_i(\tau, \gamma), 0 < \gamma\}$, таким образом, заявка игрока двигается в сторону наилучшего ответа, но может выйти сколь угодно далеко за него.

К классу «Наилучший ответ»:

- «Наилучший ответ с точностью ε (BR(ε))»: $\{s_i(\tau) \mid s_i(\tau) = s_i(\tau, \gamma), 1 - \varepsilon < \gamma \leq 1 + \varepsilon\}$. BR(0) – динамика Курно: $s_i(\tau) = br_i(s_{-i}(\tau - 1))$,
- «Наилучший ответ по собственной заявке с точностью ε (BR_i(ε))»: $\{s_i(\tau) \mid s_{ii}(\tau) = s_{ii}(\tau, \gamma), 1 - \varepsilon < \gamma \leq 1 + \varepsilon\}$.
- «Наилучший ответ по остальным (BR_{-i}(ε))»: $\{s_i(\tau) \mid s_{-ii}(\tau) = s_{-ii}(\tau, \gamma), 1 - \varepsilon < \gamma \leq 1 + \varepsilon\}$.

Рассматривать будем $\varepsilon \in \{0; 0,1; 0,2\}$: «0» соответствуют точные модели, а остальным значениям модели «с точностью», выбор именно таких значений ε обуславливается эмпирически желанием выбрать малые окрестности, при этом значение 0,2 – это уже не совсем малая величина, так как она соответствует не абсолютному, а относительному расстоянию до наилучшего ответа.

Таблица 5 показывает соответствие заявок исследуемых типов игр классам и моделям поведения.

Из таблицы ясно, что хорошей модели, объясняющей поведение игроков, пока не удалось найти – из содержательных моделей ни одна не выходит за 10% встречаемости, за исключением моделей «(Почти) не двигается», которые

объясняют около трети всех заявок. Кстати, именно такой модели поведения соответствует динамика, описанная в примере 4.

Таблица 5. Анализ поведения

Имя набора	Весь набор	Тип 1	Тип 9	Тип 25
Всего заявок	553	183	185	185
C	141 (25,5%)	44 (24,0%)	58 (31,4%)	39 (21,1%)
$C(0,1)$	191 (34,5%)	66 (36,1%)	75 (40,5%)	50 (27,0%)
IB	36 (6,5%)	13 (7,1%)	8 (4,3%)	15 (8,1%)
$IB+$	58 (10,5%)	18 (9,8%)	17 (9,2%)	23 (12,4%)
$BR(0)$	0 (0,0%)	0 (0,0%)	0 (0,0%)	0 (0,0%)
$BR(0,1)$	0 (0,0%)	0 (0,0%)	0 (0,0%)	0 (0,0%)
$BR(0,2)$	2 (0,4%)	1 (0,5%)	0 (0,0%)	1 (0,5%)
$BR_i(0,1)$	10 (1,8%)	1 (0,5%)	4 (2,2%)	5 (2,7%)
$BR_i(0,2)$	18 (3,3%)	4 (2,2%)	5 (2,7%)	9 (4,9%)
$BR_{.i}(0,1)$	2 (0,4%)	0 (0,0%)	0 (0,0%)	2 (1,1%)
$BR_{.i}(0,2)$	3 (0,5%)	1 (0,5%)	0 (0,0%)	2 (1,1%)

Если расположить модели в порядке убывания их встречаемости в объёме данных, то приблизительно первые четыре: «в сторону BR », $BR_i(0,2)$, IB , «за BR ». Первая и последняя модели слишком абстрактные, а вторая и третья говорят о том, что улучшение заявки по ресурсу для себя и индикаторное поведение занимают некоторое заметное место среди поведения игроков. Интересно также, что наибольшее количество заявок, совпадающих с IB , было у игроков с типом $r = 25$.

Приведём ещё графики, показывающие наглядно динамику средних заявок о себе в соотнесении с наилучшими ответами. На рис. 4 светлые столбцы соответствуют наилучшим ответам, а тёмные – средним заявкам игроков, высота столбцов показывает величину отклонения заявки на предыдущей итерации от наилучшего ответа и от заявки на текущей итерации соответственно. Отклонения от наилучших ответов взяты по модулю, а тёмные столбцы вниз соответствуют случаям, когда заявка на текущей итерации была в другую сторону от наилучшего ответа на ситуацию с предыдущей итерации. Видно, что игроки каждого типа порой ведут себя нерационально.

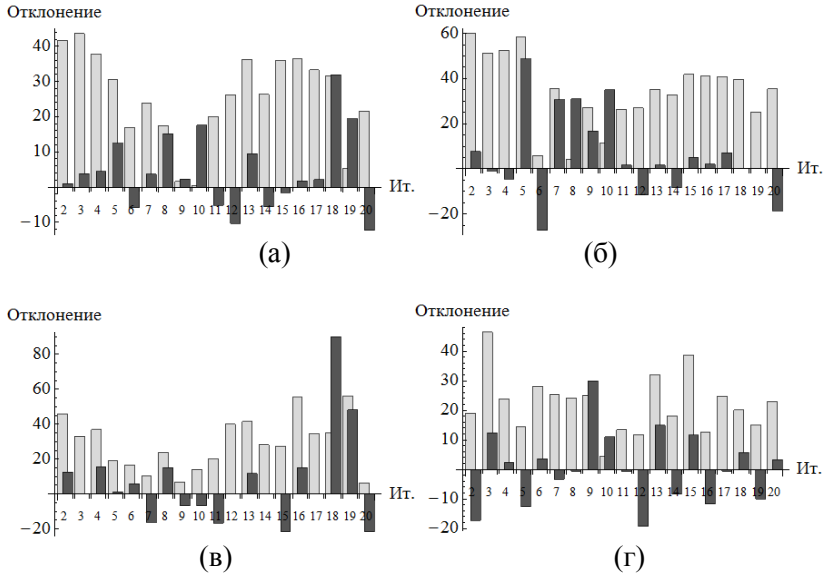


Рис. 4. Динамика средних заявок о себе в сравнении с наилучшими ответами в «Базовых» некооперативных играх.
 Средние (а) – по всем типам; (б) – по первому типу;
 (в) – по второму; (г) – по третьему типу

Заявки, которые не удалось идентифицировать с помощью предложенных моделей, содержат компоненты, которые изменялись не по направлению к наилучшему ответу по данной компоненте относительно заявки на предыдущей итерации (т.е. не могла считаться рациональной вообще). Модель, адекватно объясняющую такое поведение, авторам пока не удалось предложить. Возможно, что игроки действуют «нерационально», пытаясь бороться за ресурс и не учитывая штрафы. Или наоборот пытаются предсказать поведение оппонентов на несколько итераций вперёд, т.е. применять рефлексивную модель. В дальнейшем можно исследовать ходы игроков на совпадение с рефлексивными моделями, как, например, в [9]. Отдельный вопрос для анализа – игры, в которых проявилось кооперативное поведение с более чем двумя коалициями.

Следующий планируемый этап проверки механизма – проведение серии игр с «редуцированным» механизмом, реализующим только итеративный процесс $\hat{I}\rho$, который отсекает некоторые нежелательные модели поведения игроков – в частности кооперацию путем согласования сообщаемых заявок.

7. Заключение

Основным результатом данной статьи можно считать тот факт, что авторам удалось на основе подхода, предложенного в [7] – представления задачи распределения ресурсов как задачи многокритериального голосования, – предложить механизм, эффективно решающий задачу распределения ресурсов на основе механизма, однокритериального голосования. Предложен механизм, обеспечивающий эффективное распределение ресурса как единственное равновесие Нэша в индуцированной им игре агентов и обеспечивающий максимум суммарной полезности агентов.

Разработанный авторами инструментарий для проведения экспериментальных игр позволяет проводить апробацию применимости механизма для решения различных прикладных задач распределения ресурсов.

Однако остаётся открытым целый ряд вопросов, которые требуют дальнейшего теоретического и экспериментального исследования, среди которых особенно хочется выделить следующие.

1. Ослабление ограничений на класс функций полезности агентов. Для большинства прикладных задач в лучшем случае можно будет обеспечить выполнимость условия вогнутости. В статье, посвященной применению данного механизма для решения задачи активной экспертизы [8], было показано, что положительные результаты о существовании эффективного решения как равновесия Нэша в игре агентов могут быть распространены на класс кусочно-линейных вогнутых функций полезности. Однако динамические свойства механизма не исследовались. Поэтому представляется интересным распространение получен-

ных в данной статье результатов на класс кусочно-линейных функций полезности.

2. Исследование свойств механизма и процессов итеративных переговоров в рамках различных гипотез о поведении агентов, так как в статье была исследована лишь «простейшая» модель поведения – динамика Курно. Причем, как это было показано в статье, в игре, кроме единственного и эффективного равновесия Нэша, могут существовать и равновесия других типов, например равновесие в безопасных стратегиях [6], не позволяющие обеспечить эффективность распределения ресурсов. В защиту предложенного механизма следует отметить, что данный вопрос вообще является открытым и актуальным для теории построения эффективных экономических механизмов.

3. Исследование методов обеспечения устойчивости механизма к кооперативному поведению участников, в рамках которого участники делятся на несколько коалиций. Полученные в статье результаты позволяют предположить, что наихудшей с точки зрения механизма ситуацией является разделение всего сообщества на две коалиции. Остальные коалиционные конфигурации не должны представлять проблем для работоспособности механизма. Однако это вопрос требует более детального изучения, включая сопоставление с кооперативными моделями распределения ресурсов [5, 10].

8. Приложение

Сопоставление механизма $\rho = \langle S, \pi, t \rangle$ с «квадратичным правилом» Гровса–Лейдярда.

Будем обозначать $s^k = \{s_{ki}\}_{i \in N}$ – вектор заявок всех агентов относительно ресурса, выделяемого агенту $k \in N$.

Очевидно, что (3) тогда можно записать следующим образом:

$$x_k(s) = x_k(s^k) = \frac{1}{n} \sum_{i=1}^n s_{ki}.$$

Тогда $\forall i \in N$ выражение (4) можно переписать как

$$p_i(s) = \sum_{k=1}^n p_{ki}(s^k),$$

где

$$p_{ki}(s^k) = \beta(x_k(s^k) - s_{ki})^2.$$

А выражение (5) можно представить как

$$t_i(s) = \sum_{k=1}^n t_{ki}(s^k),$$

где

$$t_{ki}(s^k) = p_{ki}(s^k) - \frac{\alpha}{n} \sum_{j=1}^n p_{kj}(s^k).$$

Введем обозначения $t^k = \{t_{ki}\}_{i \in N}$ и $p^k = \{p_{ki}\}_{i \in N}$. Тогда механизм $\rho = \langle S, \pi, t \rangle$ можно рассматривать как набор из n механизмов $\{\rho_k\}_{k \in N}$, где $\rho_k = \langle \mathbb{R}^n, x_k(s^k), t^k \rangle$, связанные следующим совместными ограничениями:

$$\forall k \in N \sum_{i \in N} s_{ki} \leq R.$$

Рассмотрим отдельный механизм $\rho_k = \langle \mathbb{R}^n, x_k(s^k), t^k \rangle$ при условии, что $\alpha = 1$. Сделаем замену переменных, приводящих описание механизма в соответствии с описанием в [19]:

$$m^i = \frac{s_{ki}}{n}, \quad m = \{m^i\}_{i \in N} \quad \bar{m} = \frac{1}{n} \sum_{i \in N} m^i.$$

Тогда получаем, что

$$x_k(m) = \sum_{i \in N} m^i,$$

$$\bar{m} = \frac{1}{n} x_k(m),$$

$$t_{ki}(m) = \frac{\beta}{n^2} ((\bar{m} - m^i)^2 - \frac{1}{n} \sum_{j \in N} (\bar{m} - m^j)^2).$$

Эта запись эквивалентна записи квадратичного механизма Гровса–Ледйярда, приведенной в [19] на стр. 1491 с учетом ремарки 8 и при условии, что стоимость производства общественного блага (в обозначениях, используемых в [19]) $q = 0$.

Доказательство леммы 1:

По сути, необходимо доказать, что $\forall i \in N, \forall s_{-i} \in S_{-i}$ существует единственный максимум $\varphi_i(s_i, s_{-i})$ по s_i .

Очевидно, что $\forall i \in N$ функции $p_i(s)$ являются строго выпуклыми.

Покажем, что при $n > \alpha + 1 \quad \forall i \in N$ функции $t_i(s)$ являются тоже строго выпуклыми. $\forall j \in N \quad \forall s \in S$

$$\frac{\partial t_i(s)}{\partial s_{ji}} = 2\beta \left[(x_j - s_{ji}) \left(\frac{1}{n} - 1 + \frac{\alpha}{n} \right) - \frac{\alpha}{n^2} \sum_{k \in N} (x_j - s_{jk}) \right].$$

Так как $\forall j, k \in N \quad \sum_{k \in N} s_{jk} = nx_j$, то

$$\frac{\partial t_i(s)}{\partial s_{ji}} = 2\beta \left[(x_j - s_{ji}) \left(\frac{1}{n} - 1 + \frac{\alpha}{n} \right) \right] = \frac{n - \alpha - 1}{n - 1} \frac{\partial p_i(s)}{\partial s_{ji}}.$$

Так как $n > 1$, то при $n > \alpha + 1$ знаки $\frac{\partial t_i(s)}{\partial s_{ji}}$ и $\frac{\partial p_i(s)}{\partial s_{ji}}$ совпадают, а любые производные высших порядков двух функций будут пропорциональны друг другу с коэффициентом $\frac{n - \alpha - 1}{n - 1}$.

Следовательно, если $p_i(s)$ строго выпукла, то $t_i(s)$ также строго выпукла при $n > \alpha + 1$. Поэтому, функция $\varphi_i(s) = u_i(\pi(s)) - t_i(s)$ строго вогнута, откуда следует единственность ее максимума по $s_i \quad \forall s_{-i} \in S_{-i}$.

Доказательство леммы 2:

Для любого агента $i \in N$ выбор наилучшего ответа в соответствии с леммой 1 является задачей выпуклой оптимизации с лагранжианом

$$L_i(s_i) = \varphi_i(s_i, s_{-i}) + \lambda_i \left(\sum_{j \in N} s_{ji} - R \right).$$

Следовательно:

$$\frac{\partial L_i(s_i)}{\partial s_{ii}} = u'_i(x_i) \frac{1}{n} - 2\tilde{\beta}(x_i - s_{ii}) \left(\frac{1}{n} - 1 \right) + \lambda_i,$$

$$\frac{\partial L_i(s_i)}{\partial s_{ji}} = -2\tilde{\beta}(x_j - s_{ji}) \left(\frac{1}{n} - 1 \right) + \lambda_i, \quad j \in N \setminus \{i\},$$

$$\frac{\partial L_i(s_i)}{\partial \lambda_i} = \sum_{j \in N} s_{ji} - R.$$

Откуда следует, что в оптимальном решении $\forall j \in N \setminus \{i\} \Delta_{ji}$ одинаково, так как

$$\Delta_{ji} = br_{ji}(s_{-i}) - x_{j-i} = \frac{n^2}{2\tilde{\beta}(n-1)^2} \lambda_i.$$

1. Пусть решение задачи является внутренним:

$$\sum_{j \in N} br_{ji}(s_{-i}) \leq R, \quad \lambda_i = 0.$$

Тогда $\forall j \in N \setminus \{i\}$ $x_j = br_{ji}(s_{-i})$, откуда очевидным образом получаем, что $br_{ji}(s_{-i}) = x_{j-i}$ и $\Delta_{ji} = 0$. С учетом $\Delta_{ii} = br_{ii}(s_{-i}) - x_{i-i}$ получаем, что

$$\sum_{j \in N} br_{ji}(s_{-i}) = \Delta_{ii} + \sum_{j \in N} x_{j-i} \leq R.$$

То есть $A_i = 0$. Поэтому $br_{ii}(s_{-i})$ определяется из решения уравнения $u'_i(x_i) = 2\tilde{\beta}(x_i - s_{ii})(1-n)$, которое эквивалентно

$$u'_i\left(\frac{1}{n}\Delta_{ii} + x_{i-i}\right) = 2\tilde{\beta}\frac{(n-1)^2}{n}\Delta_{ii}.$$

Что и требовалось показать.

2. Пусть решение задачи граничное: $\sum_{j \in N} br_{ji}(s_{-i}) = R$, $\lambda_i < 0$.

Тогда с учетом того, что $\sum_{j \in N \setminus \{i\}} \Delta_{ji} = R - \Delta_{ii} - \sum_{j \in N} x_{j-i}$, получаем,

что $\sum_{j \in N \setminus \{i\}} \Delta_{ji} < 0$, так как $\Delta_{ii} > 0$, что следует из свойств класса

\hat{U} . Откуда получаем

$$\Delta_{ii} + \sum_{j \in N} x_{j-i} - R = \frac{n^2}{2\tilde{\beta}(1-n)} \lambda_i,$$

причем

$$\Delta_{ii} + \sum_{j \in N} x_{j-i} \geq R.$$

Следовательно,

$$u'_i\left(\frac{1}{n}\Delta_{ii} + x_{i-i}\right) = 2\tilde{\beta}\frac{n-1}{n}((n-1)\Delta_{ii} + A_i),$$

и $\forall j \in N \setminus \{i\}$ $\Delta_{ji} = A_i / (1-n)$, где

$$A_i = \Delta_{ii} + \sum_{j \in N} x_{j-i} - R.$$

Что и требовалось показать.

Доказательство утверждения 1:

1. Покажем, что $\forall u \in \hat{U}$ для неподвижной точки выполняется условие $\forall i \in N \sum_{j \in N} s_{ji}^* = R$.

Очевидно, что только в этом случае $\sum_{i \in N} x_i(s^*) = R$.

Для неподвижной точки $BR(s^*) = s^*$ и $\pi(BR(s^*)) = \pi(s^*)$.
Что эквивалентно выполнению системы равенств

$$\sum_{i \in N} br_{ji}(s_{-i}^*) = nx_j(s^*), \quad j \in N,$$

Откуда следует, что

$$(11) \sum_{j \in N} \sum_{i \in N} br_{ji}(s_{-i}^*) = n \sum_{j \in N} x_j(s^*).$$

Покажем, что последнее равенство выполняется только при $\sum_{j \in N} x_j(s^*) = R$.

Из определения Δ_{ji} следует, что $\forall \{i, j\} \in N^2$

$$br_{ji}(s_{-i}^*) = x_j(s^*) + \frac{n-1}{n} \Delta_{ji}.$$

Из леммы 2 следует, что $\forall j \in N$

$$\sum_{j \in N} br_{ji}(s_{-i}^*) = \sum_{j \in N} x_j(s^*) + \frac{n-1}{n} (\Delta_{ii} - A_i).$$

Соответственно,

$$\sum_{j \in N} \sum_{i \in N} br_{ji}(s_{-i}^*) = n \sum_{j \in N} x_j(s^*) + \frac{n-1}{n} \sum_{i \in N} (\Delta_{ii} - A_i).$$

То есть из (11) следует, что для неподвижной точки должно выполняться условие

$$\sum_{i \in N} (\Delta_{ii} - A_i) = 0.$$

Если $\sum_{i \in N} x_i(s^*) < R$, то

$$\exists K \subseteq N = \{k \in N : \sum_{j \in N} s_{jk}^* < R\}.$$

То есть для агентов из K решение задачи поиска наилучшего ответа является внутренним. Из леммы 2 получаем, что $\forall l \in K A_k = 0, \forall i \in N \setminus K$

$$A_i = \Delta_{ii} + \sum_{j \in N} x_{j-i} - R.$$

Следовательно,

$$\sum_{i \in N} (\Delta_{ii} - A_i) = \sum_{l \in K} \Delta_{ll} + \sum_{i \in N \setminus K} (R - \sum_{j \in N} x_{j-i}).$$

Из свойств класса \widehat{U} следует, что $\forall i \in N \Delta_{ii} > 0$, а из того, что решение внутреннее, получаем, что $\forall i \in N \setminus K$

$$R - \sum_{j \in N} x_{j-i} > 0.$$

Таким образом, если для $s^* \in S \sum_{i \in N} x_i(s^*) < R$ то

$$\sum_{i \in N} (\Delta_{ii} - A_i) > 0,$$

что недопустимо для неподвижной точки.

Если $\sum_{i \in N} x_i(s^*) = R$, то $\forall i \in N A_i = \Delta_{ii}$ и

$$\sum_{i \in N} (\Delta_{ii} - A_i) = 0.$$

То есть (11) выполняется только при $\sum_{i \in N} x_i(s^*) = R$.

2. Покажем, что равновесные по Нэшу заявки агентов определяют эффективное распределение ресурса. Так как $\forall i \in N A_i = \Delta_{ii}$, то из (7) получаем, что для любой неподвижной точки s^* верно

$$\sum_{i \in N} u'_i(x_i(s^*)) = 2\tilde{\beta}(n-1) \sum_{i \in N} \Delta_{ii} = 2\tilde{\beta}n(\sum_{i \in N} s_{ii}^* - R).$$

Из леммы 2, получаем, что $\forall i \in N$

$$x_i = \frac{1}{n}(nx_i + \frac{n-1}{n} \Delta_{ii} - \frac{1}{n} \sum_{j \in N \setminus \{i\}} \Delta_{jj}).$$

Следовательно, $\forall i \in N$

$$(n-1)\Delta_{ii} = \sum_{j \in N \setminus \{i\}} \Delta_{jj}.$$

Решение данной системы единственное, так как это система n линейно независимых уравнений с n переменными. Оно имеет следующий вид: $\forall \{i, j\} \in N^2 \Delta_{ii} = \Delta_{jj}$.

С учетом того, что $\Delta = \sum_{i \in N} s_{ii}^* - R$, получаем, что $\forall i \in N$

$$(12) \Delta_{ii} = \frac{1}{n-1} \Delta.$$

Следовательно, $\forall i \in N$

$$u_i'(x_i(s^*)) = 2\tilde{\beta}(\sum_{i \in N} s_{ii}^* - R).$$

Таким образом, получаем, что для любого равновесного по Нэшу набора заявок агентов s^*

$$\sum_{i \in N} x_i(s^*) = R$$

и $\forall i \in N u_i'(x_i(s^*)) = \lambda$, где $\lambda = 2\tilde{\beta}\Delta$.

То есть $\pi(s)$ является решением задачи (1).

3. Покажем, что равновесие Нэша единственно. Задача (1) выпуклая и обладает единственным решением, т.е. λ и $x_i(s^*)$, $i \in N$ определены однозначно. Поэтому s^* также единственно, так как из (12) следует, что

$$\forall i \in N s_{ii}^* = x_i(s^*) - \frac{\lambda}{2\tilde{\beta}n}.$$

4. Наконец, покажем, как определить Δ в равновесии.

Из того, что $\forall i \in N u_i'(x_i(s^*)) = 2\tilde{\beta}\Delta$, получаем, что $x_i(s^*) = u_i'^{-1}(2\tilde{\beta}\Delta)$. Из свойств класса \hat{U} следует, что $\forall u \in \hat{U}$ и $\forall i \in N u_i'^{-1}(\bullet)$ – строго убывающая функция.

Следовательно, $\sum_{i \in N} u_i'^{-1}(2\tilde{\beta}\Delta)$ также является строго убывающей по Δ . Поэтому уравнение

$$\sum_{i \in N} u_i'^{-1}(2\tilde{\beta}\Delta) = R$$

всегда имеет (в предположении, что в решении задачи (1) ресурс должен распределяться между всеми агентами) единственное решение, которое и определяет Δ в равновесии.

Доказательство следствия 1:

Если $\alpha < 1$, то $\sum_{i=1}^n t_i(s^*) = 0$ только при $\Delta = 0$. Из (2) получаем, что при этом $\forall i \in N \ u'_i(x^*) = 0$, что невозможно из определения \hat{U} .

Доказательство леммы 3:

Если $|\pi(s(\tau))| < R$, то из утверждения 1 следует, что

$$\sum_{j \in N} \sum_{i \in N} br_{ji}(s_{-i}) > n \sum_{j \in N} x_j(s).$$

То есть $n|\pi(s(\tau+1))| > n|\pi(s(\tau))|$. При этом $n|\pi(s(\tau+1))| - n|\pi(s(\tau))| \rightarrow 0$ только при $\sum_{i \in N} u'_i(x_i(s(\tau+1))) \rightarrow 0$,

что означает, что решение задачи (1) должно быть «почти» внутренним, а это не так.

Если $|\pi(s(\tau))| = R$, то

$$\sum_{j \in N} \sum_{i \in N} br_{ji}(s_{-i}) = n \sum_{j \in N} x_j(s).$$

То есть $n|\pi(s(\tau+1))| = n|\pi(s(\tau))| = nR$.

Доказательство леммы 4:

Исследуем отображение $BR(\bullet) : S \rightarrow S$. Исследуем его сжимаемость с помощью матрицы Якоби

$$J(s) = \left[\frac{\partial br_{ij}}{\partial s_{lk}}(s) \right]_{i,j,l,k \in N}.$$

Достаточным условием сжимаемости для некоторого набора стратегий агентов $s \in S$ будет выполнение условия $\|J(s)\| < 1$ для произвольной матричной нормы [1]. По аналогии с [31] будем исследовать максимальную строчную норму –

$$\|J(s)\| = \max_{\{i,j\} \in N^2} \sum_{\{l,k\} \in N^2} \left| \frac{\partial br_{ij}}{\partial s_{lk}} \right|.$$

Легко получить, что $\forall i \in N$ при $\tilde{\beta} > 0$ и при вогнутых функциях полезности выполняется

$$\sum_{\{l,k\} \in N^2} \left| \frac{\partial br_{ii}}{\partial s_{lk}} \right| = \sum_{j \in N \setminus \{i\}} \left| \frac{\partial br_{ii}}{\partial s_{ij}} \right| = (n-1) \left| \frac{2\tilde{\beta}n + u_i''(x_i)}{-u_i''(x_i) + 2\tilde{\beta}n(n-1)} \right| < 1.$$

Введем обозначения

$$D_i = \frac{2\tilde{\beta}n + u_i''(x_i)}{-u_i''(x_i) + 2\tilde{\beta}n(n-1)}, \quad \bar{D} = \max_{i \in N} |D_i|, \quad \underline{D} = \min_{i \in N} D_i.$$

Тогда $(n-1)\bar{D} < 1$, $(n-1)\underline{D} < 1$.

Мы предполагаем, что каждый агент действует по динамике Курно, что означает:

$$br_{ji} = x_{j-i} + \frac{1}{n-1}(x_{i-i} - br_{ii}), \quad j \in N \setminus \{i\},$$

где $x_{i-j} = \frac{1}{n-1} \sum_{k \in N \setminus \{j\}} s_{ik}$,

откуда получаем, что $\forall j \in N \setminus \{i\}$

$$\sum_{\{l,k\} \in N^2} \left| \frac{\partial br_{ji}}{\partial s_{lk}} \right| = \sum_{k \in N \setminus \{i\}} \frac{1}{n-1} + \frac{1}{n-1} \sum_{k \in N \setminus \{i\}} \left(\frac{1}{n-1} - D_i \right).$$

Следовательно, с учетом того, что $(n-1)D_i < 1$

$$\sum_{\{l,k\} \in N^2} \left| \frac{\partial br_{ji}}{\partial s_{lk}} \right| = 1 + \frac{1}{n-1} - D_i > 1.$$

То есть отображение $BR(\bullet): S \rightarrow S$ не удовлетворяет достаточным условиям сжимаемости.

Исследуем отображение $BR^2(\bullet) = BR(BR(\bullet)): S \rightarrow S$. С учетом написанного выше верно, что $\forall i \in N$

$$\sum_{\{l,k\} \in N^2} \left| \frac{\partial br_{ii}^2}{\partial s_{lk}} \right| = |D_i| \sum_{j \in N \setminus \{i\}} \left(1 + \frac{1}{n-1} - D_j \right).$$

Кроме того, с учетом того, что каждый агент действует по динамике Курно, получаем, что $\forall j \in N \setminus \{i\}$

$$\sum_{\{l,k\} \in N^2} \left| \frac{\partial br_{ji}^2}{\partial s_{lk}} \right| = \left(1 + \frac{1}{n-1} - D_i \right) \sum_{j \in N \setminus \{i\}} |D_j|.$$

Откуда получаем, что $\forall i \in N$

$$\sum_{\{l,k\} \in N^2} \left| \frac{\partial br_{ii}^2}{\partial s_{lk}} \right| \leq \bar{D}(n - (n-1)\underline{D}),$$

и $\forall j \in N \setminus \{i\}$

$$\sum_{\{l,k\} \in N^2} \left| \frac{\partial br^2_{ji}}{\partial s_{lk}} \right| \leq \bar{D}(n - (n-1)\underline{D}).$$

То есть если выполняется условие $\bar{D}(n - (n-1)\underline{D}) < 1$, то отображение $BR^2(\bullet)$ является сжимающим.

Проанализируем, при каких $\tilde{\beta}$ это может быть достигнуто.

Если $\tilde{\beta} \geq \frac{1}{2n} \max_{i \in N} (-u_i''(x_i(s)))$, то $\underline{D} \geq 0$. Тогда всегда можно

обеспечить выполнение условия $\bar{D}(n - (n-1)\underline{D}) \leq 1$ выбором

$\underline{D} \leq \bar{D} < \frac{1}{n-1}$, так как $\underline{D}(n - (n-1)\underline{D}) < 1$ при $\underline{D} < \frac{1}{n-1}$. Увели-

чение значения $\tilde{\beta}$ обеспечивает $\underline{D} \rightarrow \bar{D} \rightarrow \frac{1}{n-1}$, однако при

этом $\bar{D}(n - (n-1)\underline{D}) \rightarrow 1$. Поэтому делать $\tilde{\beta}$ очень большим нецелесообразно.

Таким образом, показано, что выбором параметра $\tilde{\beta}$ механизма можно обеспечить сжимаемость отображения $BR^2(\bullet)$ при $\max_{i \in N} (-u_i''(x_i(s))) \leq C$.

Доказательство утверждения 2:

Из леммы 3 получаем, что итеративный процесс, стартовав в S , перейдет в \bar{S} . Из леммы 4 следует, что при выполнении условия $\max_{i \in N} (-u_i''(x_i(s))) \leq C$ выбором β можно обеспечить сходимость пары итеративных процессов $s(\tau+2) = BR^2(s(\tau))$ и $s(\tau+3) = BR^2(s(\tau+1))$, где $s(\tau+1) = BR(s(\tau))$, $\tau \geq 1$, к одному и тому же равновесию Нэша, так как оно является единственной неподвижной точкой для каждого из этих процессов.

Доказательство леммы 5:

В индуцированной механизмом $\hat{\rho} = \langle \hat{S}, \hat{\pi}, \hat{t} \rangle$ игре $\Gamma(\hat{\rho})$ агенты обладают функциями предпочтения $\hat{\phi}_i(s) = u_i(\hat{x}_i(s)) - \hat{p}_i(s)$. Очевидно, что при $\hat{\beta} > 0$ эти функции вогнутые.

Наилучший ответ агента $i \in N$ будет определяться как решение (единственное) уравнения

$$u_i'(\hat{x}_i)(1 - \frac{1}{n}) - 2\hat{\beta}(br_{ii} - \hat{x}_i)\frac{1}{n} = 0.$$

С учетом (5) получаем, что равновесные сообщения агентов удовлетворяют следующей системе уравнений:

$$u_i'(\hat{x}_i^*)(n-1) = 2\hat{\beta} \frac{\sum_{i \in N} br_{ii} - R}{n}, \quad i \in N.$$

По аналогии с леммами 1 и 2, очевидно, что решение этой системы единственно. Более того, это решение обеспечивает $\forall i \in N \quad u_i'(\hat{x}_i^*) = const$, что соответствует решению задачи (1).

Доказательство утверждения 3:

Эквивалентность механизмов очевидным образом следует из того факта, что решение задачи (1) единственно.

Из леммы 5 получаем, что в игре $\Gamma(\hat{\rho})$ равновесные заявки агентов удовлетворяют системе уравнений

$$u_i'(\hat{x}_i^*)(n-1) = 2\hat{\beta} \frac{\sum_{i \in N} br_{ii} - R}{n}, \quad i \in N.$$

Из утверждения 1 следует, что в игре $\Gamma(\rho)$

$$u_i'(x_i^*) = 2\tilde{\beta}(\sum_{i \in N} br_{ii} - R).$$

Следовательно, $\forall i \in N \quad br_{ii} = br_{ii}$ при $\hat{\beta} = \tilde{\beta} \frac{1}{n(n-1)}$. То есть

$$\hat{\beta} = \beta \frac{n - \alpha - 1}{n(n-1)^2}.$$

Литература

1. БОСС В. *Лекции по математике: дифференциальные уравнения. Т. 2.* – М.: Едиториал УРСС, 2004. – 208 с.
2. БУРКОВ В.Н., ДАНЕВ Б., ЕНАЛЕЕВ А.К. *Большие системы: моделирование организационных механизмов.* – М.: Наука, 1989. – 248 с.
3. БУРКОВ В.Н., ДЖАВАХАДЗЕ Г.С., ДИНОВА Н.И., ЩЕПКИН Д.А. *Применение игрового имитационного моделиро-*

- вания для оценки эффективности экономических механизмов. – М.: ИПУ РАН, 2003. – 51 с.
4. ГУБКО М.В., НОВИКОВ Д.А. *Теория игр в управлении организационными системами*. – М.: Синтег, 2002. – 148 с.
 5. ГУБКО М.В., СПРЫСКОВ Д.С. *Учет кооперативных взаимодействий в механизмах планирования* // Управление большими системами. Выпуск 2. – М.: Фонд «Проблемы управления», 2000. – С. 28–38.
 6. ИСКАКОВ М.Б. *Равновесие в безопасных стратегиях и равновесия в угрозах и контругрозах в некооперативных играх* // Автоматика и телемеханика. – 2008. – №2. – С. 114–134.
 7. КОРГИН Н.А. *Представление механизма последовательного распределения ресурсов как неманипулируемого механизма многокритериальной активной экспертизы* // Управление большими системами. – 2012. – №36. – С. 186–208.
 8. КОРГИН Н.А., ХРИСТЮК А.А. *Эффективный механизм активной экспертизы с платой за участие как инструмент принятия согласованных решений* // Вестник Воронежского государственного технического университета. – Т. 7, №6. – С. 117–121.
 9. КОРЕПАНОВ В.О., НОВИКОВ Д.А. *Метод рефлексивных разбиений в моделях группового поведения и управления* // Проблемы управления. – 2011. – №1. – С. 21–32.
 10. МАЗАЛОВ В.В., МЕНЧЕР А.Э., ТОКАРЕВА Ю.С. *Переговоры. Математическая теория*. – Санкт-Петербург-Москва-Краснодар: Лань, 2012. – 304 с.
 11. НОВИКОВ Д.А. *Теория управления организационными системами*. – 3-е изд. – М.: Издательство физико-математической литературы, 2012. – 604 с.
 12. ARIFOVIC J., LEDYARD J.O. *A behavioral model for mechanism design: Individual evolutionary learning* // Journal of Economic Behavior and Organization. – 2011. – №78. – P. 375–395.
 13. BARBERÁ S., JACKSON M., NEME A. *Strategy-Proof Allotment Rules* // Games and Economic Behavior. – 1997. – Vol. 18, Issue 1. – P. 1–21.

14. BASAR T., MAHESWAREN R. *Social welfare of selfish agents: Motivating efficiency for divisible resources* // Proc. Control Decision Conf. (CDC). – 2004. – P. 361–395.
15. BOYD S., PARIKH N., CHU E. *Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers* // Foundations and Trends in Machine Learning. – 2011. – Vol. 3, №1. – P. 1–122.
16. FISCHBACHER U. *z-Tree – Zurich Toolbox for Ready-made Economic Experiments* // Experimental Economics. – 2007. – Vol. 10, №2. – P. 171–178.
17. FUDENBERG D., LEVINE D., *Theory of Learning in Games* – MIT, 1999. – 276 p.
18. GOETZ R., MARTINEZ Y., AND JOFRE R. *Water allocation by social choice rules: The case of sequential rules* // Ecological Economics. – 2008. – №65(2). – P. 304–314.
19. GROVES T., LEDYARD J.O. *The Existence of Efficient and Incentive Compatible Equilibria with Public Goods* // Econometrica. – 1980. – №6. – P. 1487–1506.
20. HEALY P., MATHEVET L. *Designing stable mechanisms for economic environments, Theoretical Economics* // Econometric Society. – (в печати).
21. HURWICZ L. *Outcome functions yielding Walrasian and Lindahl allocations at Nash equilibrium points* // Review of Economic Studies. – 1979. – №46. – P. 217–225.
22. JAIN R., WALRAND J. *An efficient nash-implementation mechanism for divisible resource allocation* // Automatica. – 2010. – Vol. 46, №8. – P. 1276–1283.
23. JOHARI R., TSITSIKLIS J.N. *Efficiency of Scalar-Parameterized Mechanisms* // Operations Research. – 2009. – №57. – P. 823–839.
24. KAKHBOD A., TENEKETZIS D. *An efficient game form for unicast service provisioning* // IEEE Trans. Autom. Control. – 2012. – Vol. 57, №2. – P. 392–404.
25. KRUSKAL W.H., WALLIS W.A. *Use of ranks in one-criterion variance analysis* // J. Amer. Statist. Assoc. – 1952. – Vol. 47. – P. 583–621.
26. LEFEBVRE M. *Sharing Rules for Common-Pool Resources when Self-insurance is Available: an Experiment* // Working Pa-

- pers 11-22, LAMETA, Universit'y of Montpellier. – 2012. – P. 42.
27. MASKIN E. *The Theory of Nash Equilibrium: A Survey* / In: Hurwicz L., Schmeidler D., Sonnenschein H. *Social Goals and Social Organization*. – Cambridge: Cambridge University Press, 1985. – P. 173–204.
28. MATHEVET L. *Supermodular mechanism design* // *Theoretical Economics*, Econometric Society. – 2010. – Vol. 5(3). – P. 403–443.
29. MOULIN H. *An efficient and almost budget balanced cost sharing method* // *Games and Economic Behavior*. – 2010. – Vol. 70, Issue 1. – P. 107–131.
30. SPRUMONT Y. *The division problem with single-peaked preferences: A characterization of the uniform rule* // *Econometrica*. – 1991. – Vol. 59. – P. 509–519.
31. VAN ESSEN M. *A note on the stability of Chen's Lindahl mechanism* // *Social Choice and Welfare*. – 2012. – Vol. 38(2). – P. 365–370.
32. WALKER M. *A simple incentive compatible scheme for attaining Lindahl allocations* // *Econometrica*. – 1981. – №49. – P. 65–71.

EFFICIENT SOLUTION OF ALLOTMENT PROBLEM WITH GROVES-LEDYARD MECHANISM UNDER TRANSFERABLE UTILITY

Korgin Nikolay, Institute of Control Sciences of RAS, Moscow, Cand.Sc., leading scientist, Moscow Institute of Physics and Technology, associate professor (nkorgin@ipu.ru).

Korepanov Vsevolod, Institute of Control Sciences of RAS, Moscow, Cand.Sc., senior scientist.

Abstract: We consider a problem of allocation of limited amount of divisible good among a finite number of agents under transferable utility. We design a mechanism which implements a Pareto efficient allocation as a Nash equilibrium of the induced game. This mechanism is an adaptation of the Groves-Ledyard “quadratic government” that was initially offered for the problem of public good.

Keywords: mechanism design, Nash implementation, allotment problem.

Статья представлена к публикации членом редакционной коллегии В.Н. Бурковым

Поступила в редакцию 22.02.2013.

Опубликована 30.11.2013.

УДК 021.8 + 025.1
ББК 78.34

О ДИСКРЕТНО-АВТОМАТНЫХ МОДЕЛЯХ КОНФОРМНОГО ПОВЕДЕНИЯ¹

Семенов А. А.², Кочемазов С. Е.³

*(Институт динамики систем и теории управления СО РАН,
Иркутск)*

В работе для исследования феномена конформного поведения вводится дискретно-автоматная модель. Концептуально данная модель близка к дискретным моделям, используемым в компьютерной биологии для описания динамических процессов в генных сетях. В рамках введенной модели исследуются мультиагентные системы, в которых помимо агентов-конформистов, принимающих решения в зависимости от мнения их окружения, имеются агенты, которые никогда не меняют своего мнения (так называемые «агитаторы» и «лоялисты»). Исследуются задачи поиска таких начальных размещений относительно небольшого числа агитаторов (лоялистов), которые способствуют переходу системы в состояние с большинством действующих (бездействующих) агентов. Использованный в работе вычислительный аппарат, основанный на SAT-подходе, позволил с помощью обычного ПК решать соответствующие задачи для случайным образом сгенерированных систем на 100 вершинах.

Ключевые слова: модели конформного поведения, дискретно-автоматные модели, символьные алгоритмы, SAT.

¹ Работа выполнена при поддержке гранта РФФИ №11-07-00377а.

² Александр Анатольевич Семенов, кандидат технических наук, доцент (biclor.rambler@yandex.ru).

³ Степан Евгеньевич Кочемазов, программист (veinamond@gmail.com).

Введение

В современных реалиях чрезвычайно актуальными являются вопросы, связанные с описанием динамики и прогнозированием коллективного поведения. Это обусловлено, в первую очередь, существенно возросшей в последние десятилетия скоростью обмена информацией как между отдельными людьми, так и между их группами, построенными в соответствии с различными принципами (группы по интересам, корпоративные группы и т.п.). Наблюдающиеся кризисные ситуации в экономике и труднопредсказуемые социальные процессы демонстрируют слабую эффективность имеющихся подходов к управлению сложными мультиагентными системами и несовершенство методик, применяемых для прогнозирования их поведения.

Проблемам математического моделирования коллективного поведения посвящено множество работ. Кратко охарактеризуем наиболее значимые с нашей точки зрения.

Предлагаемые и изучаемые нами модели можно отнести к пороговым. По-видимому, первые пороговые модели, описывающие динамику коллективных процессов, были предложены в статье [33]. В дальнейшем пороговые модели коллективного поведения в различных аспектах изучались в большом числе работ: [1–4, 23, 24] и др.

В ряде статей динамика коллективного поведения анализируется с теоретико-игровых позиций [1–4]. Так, в [4] описана общая теоретико-игровая модель конформного поведения, включающая пороговую составляющую и учитывающая различные виды взаимного влияния агентов.

В работе [11] для анализа коллективного поведения применяется вероятностный подход. Конкретно для прогнозирования перехода каждого агента в некоторое состояние используются соотношения между априорной и апостериорной вероятностями такого перехода. При этом априорная вероятность выражает только личное мнение агента, а апостериорная – это вероятность принятия агентом решения после учета мнения его окружения.

В статьях [23,24] в рамках пороговых моделей анализируются различные виды влияния одних агентов на других. Поведение агента при этом определяется информацией о связях и поведении его соседей. Фактически для описания взаимодействия агентов в рамках некоторой социальной группы в этих работах используются графы: агенты интерпретируются вершинами графа, а дуги или ребра графа определяют влияние одних агентов на других. Более точным термином для таких моделей является «сеть», поскольку вершины соответствующих графов могут быть неравноправными, а ребра (дуги) нести различную смысловую нагрузку (как правило, в форме приписанных им чисел).

В последние годы появилось довольно много работ, в которых изучаются как статические ([27, 40]), так и динамические [5, 27, 40] свойства социальных и информационных сетей. Книга [5] содержит целый ряд объединенных игровой составляющей подходов к моделированию динамики социальных сетей.

В рассматриваемых нами автоматных моделях конформного поведения, как и в «играх на сетях» [5], структура сети во времени никак не меняется. Однако автоматные модели не относятся к теоретико-игровым. В этих моделях функция, определяющая динамику сети, является дискретной функцией, для которой состояния равновесия и циклические режимы можно естественным образом интерпретировать на так называемых «графах состояний». Такого сорта «дискретно-автоматные» модели социальных групп имеют общие черты с моделями генных сетей, исследуемыми в информационной биологии [6–8, 16, 29, 30, 35].

Далее мы описываем простейшую дискретно-автоматную модель конформного поведения и изучаем в ее рамках ряд задач, имеющих, с нашей точки зрения, реальный практический смысл. Главная привлекательная черта дискретно-автоматных моделей состоит в возможности использовать для их численного исследования развитый аппарат символьных вычислений, показывающий хорошие результаты при решении трудных комбинаторных задач верификации и криптоанализа.

Приведем краткий план статьи. В первом разделе мы по ана-

логии с известными моделями, исследуемыми в информационной биологии, вводим простейшую дискретно-автоматную модель конформного поведения. Во втором разделе рассматриваются некоторые дополнительные детализации этой модели. В частности, в модель вводятся агенты, которые всегда принимают только одно решение – «действовать» или «бездействовать». Такие агенты называются соответственно агитаторами или лоялистами. Наличие агитаторов и лоялистов придает модели новый комбинаторный смысл. В частности, предполагая, что любой агент может являться агитатором, можно искать такие размещения (называемые далее небезопасными) относительно небольшого числа агитаторов, которые способны перевести систему в «критическое» состояние, если под таковым понимать, например, состояние, когда подавляющее большинство агентов принимают решение «действовать». Аналогичные по смыслу задачи можно ставить для лоялистов. Для систем с агитаторами, лоялистами и простыми агентами-конформистами интерес представляют так называемые «начально-упорядоченные» состояния, когда все простые агенты (не являющиеся ни агитаторами, ни лоялистами) принимают одно и то же решение, например, «бездействовать». Мы доказываем, что дискретный автомат, моделирующий конформное поведение, из любого состояния начальной упорядоченности при любом размещении агитаторов и лоялистов обязательно достигнет стационарного состояния за не более чем $n - A - L$ шагов, где n – общее число агентов, A – число агитаторов, L – число лоялистов. Данный факт позволяет построить относительно простые алгоритмы сведения задач поиска небезопасных размещений агитаторов/лоялистов к задачам поиска решений систем булевых уравнений и, в конечном счете, к SAT-задачам. Информация о таких алгоритмах сведения, а также об основных алгоритмах, используемых при решении SAT-задач, приведена в третьем разделе. В четвертом разделе приводятся результаты вычислительных экспериментов, в которых на случайным образом сгенерированных сетях мы демонстрируем, что зачастую подавляющее большинство простых агентов-конформистов, переведен-

ных в состояние действия агитаторами, можно вернуть в состояние бездействия, заменив часть простых агентов на лоялистов. В заключении кратко обсуждаются основные результаты статьи в сравнении с близкими по смыслу известными результатами, касающимися динамики коллективного поведения.

1. Дискретно-автоматная модель конформного поведения

Пусть A – произвольное конечное множество символов. Через A^n обозначается множество всевозможных последовательностей (слов) длины n , составленных из символов множества A . Будем рассматривать дискретные функции следующего вида:

$$(1) \quad f_G : \{0, \dots, r\}^n \rightarrow \{0, \dots, r\}^n, r \in \mathbb{N}, n \in \mathbb{N},$$

задаваемые при помощи ориентированных графов. Граф G , задающий функцию (1) и иногда называемый сетью, имеет n вершин, называемых агентами. Предполагается, что в каждый момент времени $t \in \{0, 1, \dots\}$ произвольной вершине v_i с номером $i, i \in \{1, \dots, n\}$, графа G приписано число $x_i(t) \in \{0, 1, \dots, r\}$, которое называется весом вершины v_i в момент времени t . Переходу от момента t к моменту $t + 1$ соответствует синхронный пересчет весов всех вершин. Правила пересчета не зависят от конкретного значения t и целиком определяются структурой графа.

Перечисленные объекты определяют дискретную динамическую мультиагентную систему. Набор весов всех вершин графа G в произвольный момент времени называется вектором состояния или состоянием данной системы. Можно заметить, что переходы между состояниями, совершаемые системой, похожи на переходы, которые совершает детерминированный конечный автомат (ДКА) – различные векторы состояний можно рассматривать как аналоги состояний ДКА. Поэтому отображения вида (1) называются автоматными или дискретно-автоматными [6, 7].

Обозначим вектор состояния рассматриваемой системы в момент времени t через $w(t)$. Поскольку множество всех различных состояний описанной системы по мощности не превос-

ходит $(r + 1)^n$, то для произвольного t_0 , $t_0 \geq 0$, обязательно найдутся такие k, m , $0 \leq k < m$, что $w(t_0 + k) = w(t_0 + m)$. В данной ситуации говорим, что последовательность состояний $w(t_0 + k), \dots, w(t_0 + m)$ образует цикл длины $m - k$. Цикл длины 1 называется стационарным состоянием или неподвижной точкой отображения (1).

Граф состояний дискретно-автоматной мультиагентной системы – это граф на $(r + 1)^n$ вершинах, обозначаемый через Γ_G . Каждой вершине соответствует уникальный вектор состояния. Вершины w, w' графа Γ_G соединены дугой (w, w') (направленной от w к w') тогда и только тогда (по определению), когда результатом применения отображения (1) к вектору состояния w является вектор состояния w' . Ситуация (w, w) , т.е. петля, соответствует неподвижной точке отображения (1).

Как уже отмечалось выше, отображения вида (1) активно используются в информационной биологии для моделирования динамики генных сетей. По-видимому, первой дискретной моделью генной сети была модель С. Кауффмана [35], пример которой мы приводим на рис. 1. На рисунке изображен граф G сети с булевыми функциями пересчета весов, заданными таблицами истинности. Справа от графа сети изображен граф состояний Γ_G данной мультиагентной системы. Можно видеть, что соответствующее отображение не имеет неподвижных точек, но имеет цикл длины 3. Динамика поведения вершин в данной модели определяется булевыми функциями, которые задаются таблицами истинности. Конечно же, такой способ крайне неэффективен – для сети на 100 вершинах построить соответствующие таблицы истинности нельзя ни за какое разумное время. В [29] рассматривается модель Кауффмана, функции пересчета весов вершин в которой задаются булевыми формулами.

В работе [6] (см. также [16]) были введены дискретно-автоматные отображения вида (1), функции пересчета весов вер-

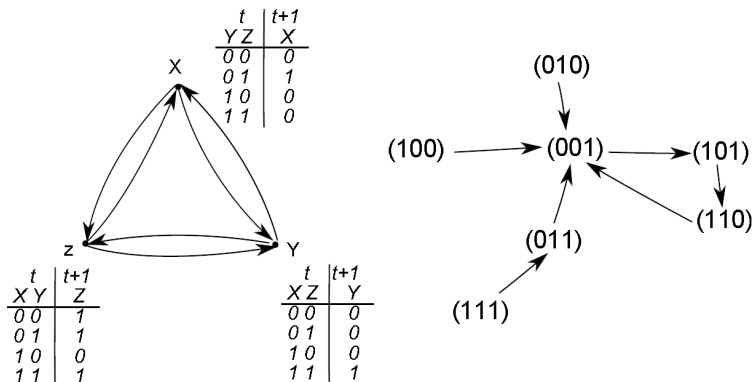


Рис. 1. Пример генной сети С. Кауффмана [35]

шин в которых задаются при помощи следующих правил:

$$(2) \quad x_i(t+1) = \begin{cases} x_i(t) + 1, & \text{если } \left(\sum_{v_j \in V_i} x_j(t) = 0 \right) \text{ и } (x_i < r), \\ x_i(t) - 1, & \text{если } \left(\sum_{v_j \in V_i} x_j(t) > 0 \right) \text{ и } (x_i > 0), \\ x_i(t), & \text{иначе.} \end{cases}$$

Здесь и далее через V_i обозначено множество всех вершин графа G , дуги из которых входят в вершину v_i (таким образом, можно сказать, что вершины из V_i образуют «окружение», непосредственно влияющее на агента v_i). Для автоматных отображений с функциями весов вида (2) (так называемых «аддитивных автоматов»), в которых граф G является циркулянтном, в [6] были найдены необходимые и достаточные условия существования неподвижных точек.

В статье [7] для численного поиска неподвижных точек аддитивных автоматов (2), задаваемых случайными графами, был использован SAT-подход [22]. Удавалось находить неподвижные точки таких отображений, заданных случайными графами на 200 и более вершинах, с использованием обычного ПК. Более подробная информация о SAT-подходе будет приведена в третьем разделе.

Далее мы вводим и исследуем простейшую дискретно-автоматную модель конформного поведения. В плане используемых понятий отправной точкой для нас стала работа [4]. При этом мы отмечаем, что рассматриваемая далее модель не учитывает теоретико-игровые аспекты, изученные в [4].

Определение 1. *Рассматриваем ориентированный граф G на n вершинах. Каждая вершина G интерпретирует агента, который может находиться в двух состояниях: состояние 0 – бездействие, и состояние 1 – действие. Произвольный агент v_i , $i \in \{1, \dots, n\}$, называется θ_i -конформистом, если в момент времени $t + 1$ он принимает решение «действовать», когда более $\lfloor \theta_i \cdot |V_i| \rfloor$ агентов из множества V_i в момент времени t приняли решение «действовать», в противном случае v_i принимает решение «бездействовать». Число $\theta_i \in [0, 1]$ называем далее порогом конформности агента (вершины) v_i . Вершина v_i имеет в момент времени t вес $x_i(t) = 1$, если v_i находится в состоянии «действовать», и $x_i(t) = 0$, если v_i находится в состоянии «бездействовать».*

Таким образом, в соответствии с определением 1, для любого агента v_i , $i \in \{1, \dots, n\}$, динамику изменения его веса можно определить следующими соотношениями:

$$(3) \quad x_i(t + 1) = \begin{cases} 1, & \sum_{v_j \in V_i} x_j(t) > \lfloor \theta_i \cdot |V_i| \rfloor, \\ 0, & \sum_{v_j \in V_i} x_j(t) \leq \lfloor \theta_i \cdot |V_i| \rfloor. \end{cases}$$

Данные соотношения задают дискретную функцию $f_G : \{0, 1\}^n \rightarrow \{0, 1\}^n$.

Описанная модель представляется вполне согласующейся с реальностью – например, при $\theta_i = \frac{1}{2}$ имеем агента, который принимает решение «действовать», только если большинство агентов, напрямую влияющих на него, принимают решение «действовать» (подобного рода ситуации весьма распространены на практике).

Далее мы исследуем возможность численного решения некоторых задач, естественным образом связанных с описанной моделью конформного поведения.

Определение 2. Для рассматриваемой модели конформного поведения (3) одну итерацию синхронного пересчета весов всех вершин графа G далее называем контактом.

Обозначим через $wt_\chi(w(t))$ вес Хэмминга вектора состояния $w(t)$ рассматриваемой системы в момент времени t .

Задача 1. Для заданных порогов конформности всех агентов, заданных чисел $P, Q \in \{1, \dots, n\}$, $Q > P$, и числа k выяснить, существует ли такое начальное состояние системы $w(0)$, что $wt_\chi(w(0)) \leq P$, а $wt_\chi(w(k)) \geq Q$.

Грубо говоря, ищется начальное состояние, в котором решение «действовать» принимают не более P агентов, но после k контактов решение «действовать» принимают уже не менее Q агентов.

2. Дополнительные детализации рассматриваемой модели

Далее мы вводим в модель (3) дополнительные детализации, согласующиеся с реальными ситуациями и имеющие вполне конкретный комбинаторный смысл. А именно, будем полагать, что некоторые агенты всегда находятся в одном состоянии и ни при каких обстоятельствах это состояние не изменяют.

Определение 3. Предположим, что в рамках рассматриваемой дискретно-автоматной модели (3) могут существовать агенты, которые в любом векторе состояния независимо от мнения их окружения принимают решение «действовать». Назовем таких агентов агитаторами. Также полагаем, что могут существовать агенты, всегда принимающие решение «бездействовать». Назовем их лоялистами. Полагаем, что остальные агенты являются конформистами с индивидуальными уровнями конформности и принимают решения в зависимости от окружающей обстановки. Этих агентов называем простыми агентами.

Определение 4. Рассматриваем произвольную мультиагентную систему в рамках модели конформного поведения (3), в которой имеются A агитаторов и L лоялистов, $A + L < n$.

Если все простые агенты системы в начальном состоянии принимают решение «действовать» либо все они принимают решение «бездействовать», то назовем данное состояние системы начально-упорядоченным относительно действия (бездействия).

Покажем, что справедливо следующее утверждение.

Утверждение 1. *Рассматриваем произвольную мультиагентную систему в рамках модели конформного поведения (3). Предположим, что в данной системе имеется A , $A \geq 0$, агитаторов, L , $L \geq 0$, лоялистов, и $n - A - L > 0$ простых агентов. Тогда для любого расположения агитаторов и лоялистов, любых уровней конформности простых агентов из любого начально-упорядоченного состояния описанная система достигнет стационарного состояния за не более чем $n - A - L$ контактов.*

Доказательство. Докажем справедливость утверждения для случая, когда система находится в начально-упорядоченном состоянии, в котором все простые агенты принимают решение «бездействовать». Предположим, что осуществляется один контакт между всеми агентами системы. Если ни один из простых агентов не изменил своего состояния, то имеем неподвижную точку (агитаторы и лоялисты не меняются по определению). Пусть на первом шаге некоторые простые агенты изменили свое состояние с 0 на 1. Пусть v – произвольный такой агент. Сказанное означает, что v изменил свое состояние с 0 на 1 только потому, что в его непосредственном окружении было достаточное (с позиции его уровня конформности) число агитаторов. Тогда этот агент уже не изменит свое состояние 1 ни на одной из последующих итераций (агитаторы не изменяют свое состояние). Если в результате второго контакта ни один простой агент не изменяет своего состояния, имеем неподвижную точку. Пусть v – произвольный агент, изменивший свое состояние на втором шаге. В силу сказанного выше, v изменил свое состояние с 0 на 1. Это произошло потому, что в непосредственном окружении v было достаточное (для его уровня конформности) число агитаторов и агентов, перешедших в состояние 1 после первого шага. Однако все эти агенты не изменяют своего состояния во всех последующих итерациях. Поэтому

состояние 1 в дальнейшем не изменит и рассматриваемый агент. Далее по аналогии. Очевидно, что в любом случае не позднее, чем через $n - A - L$ итераций система перейдет в стационарное состояние. Случай с начальной упорядоченностью относительно действия разбирается аналогичным образом. Утверждение 1 доказано.

Данное утверждение представляется весьма полезным, поскольку его можно использовать для относительно эффективной оценки возможности достижения системой «критических» состояний (о соответствующей технике более подробно будет сказано в следующем разделе).

Определение 5. *Про мультиагентную систему в рамках модели (3) скажем, что она является (A, L, α) -критической относительно действия (бездействия), если существует такое расположение A агитаторов и L лоялистов, что, стартуя из начально-упорядоченного состояния относительно бездействия (действия), система через некоторое число контактов переходит в состояние, в котором $> \alpha \cdot n$, $\alpha \in (0, 1)$, агентов находятся в состоянии действия (бездействия).*

Понятие (A, L, α) -критической системы вполне согласуется с традиционными представлениями о «небезопасной социальной группе». Действительно, предположим, что система является (A, L, α) -критической относительно действия, например с $\alpha = \frac{9}{10}$. Это означает, что можно так расположить A агитаторов (относительно L лоялистов), что, стартуя из состояния, в котором все простые агенты бездействуют, система через некоторое число контактов придет в состояние, в котором $> 90\%$ агентов действуют. В соответствии с утверждением 1 для этого потребуется не более $n - A - L$ контактов. Заметим, что если система является (A, L, α) -критической, например относительно действия, то при соответствующем размещении агитаторов и лоялистов, стартуя из любого состояния (а не только из начально-упорядоченного), за не более чем $n - A - L$ контактов она достигнет состояния, в котором $> \alpha \cdot n$ агентов будут действовать. Это состояние, правда, может не быть стационарным. Если факт действия такого числа

агентов воспринимается как негативное свойство рассматриваемой системы, то ее, конечно же, следует считать небезопасной.

В контексте введенных понятий естественным образом возникают следующие задачи исследования конформного поведения.

Задача 2. *Рассматривается мультиагентная система в рамках модели конформного поведения (3) с A агитаторами. Для состояния «бездействия» всех простых агентов требуется проверить, существует ли такое расположение агитаторов, при котором через некоторое число шагов ($\leq n - A$) более $\alpha \cdot n$ агентов будут находиться в состоянии действия (для различных уровней конформности агентов и различных α). Аналогичные задачи можно рассмотреть для лоялистов и для смешанных слушателей.*

Задача 3. *Предположим, что найдено некоторое размещение агитаторов, являющееся решением задачи 2. Рассматриваем систему с данным размещением агитаторов и полагаем, что L простых агентов становятся лоялистами. Требуется найти такое их расположение, чтобы начальное состояние данной системы, в котором все простые агенты бездействуют, было неподвижной точкой. Данная ситуация означает полное подавление лоялистами агитаторов.*

Задача 4. *Предположим, что найдено некоторое размещение A агитаторов, являющееся решением задачи 2. Рассматриваем систему с этим размещением агитаторов и полагаем, что L простых агентов становятся лоялистами. Требуется найти такое их расположение, что, стартуя из начального состояния, в котором все оставшиеся простые агенты действуют, через некоторое число контактов (не превосходящее $n - A - L$) система перейдет в состояние, в котором действуют $\leq \beta \cdot n$ агентов ($\beta < \alpha$). Данную ситуацию можно рассматривать как перевод лоялистами системы из «опасного» состояния в «более безопасное».*

3. Переход к SAT-задачам. Используемые алгоритмы решения SAT-задач

3.1. МЕТОДЫ СВЕДЕНИЯ КОМБИНАТОРНЫХ ЗАДАЧ К SAT

Известно довольно много примеров сведения различных комбинаторных задач к булевым уравнениям и, в конечном счете, к SAT-задачам. Напомним, что SAT-задачами (SAT – сокращение от Satisfiability) [22] называются задачи поиска решений булевых уравнений вида $KНФ = 1$, где КНФ – конъюнктивная нормальная форма [18]. Теоретическая возможность эффективных процедур сведения к SAT широкого класса комбинаторных проблем вытекает из теоремы С.А. Кука 1971 г. [25]. Общие принципы построения сведений к SAT, главным образом, для систем различных ограничений (в рамках общей проблемы CSP – Constraint Satisfaction Problem), а также большое число ссылок можно найти в обзорной статье [42].

Мотивация перехода к SAT-задачам состоит в том, что алгоритмы их решения – это на сегодняшний день, пожалуй, наиболее эффективные эвристические комбинаторные алгоритмы, подтверждающие свою практическую применимость на задачах символьной верификации и даже на таком классе аргументированно трудных тестов, как задачи криптоанализа.

Для сведения к SAT задач поиска неподвижных точек и циклических режимов довольно сложных дискретно-автоматных отображений, моделирующих динамические процессы в генных сетях, в [8] был применен специальный программный комплекс TransAlg [12]. Данный программный комплекс работает с описаниями функций вида (1) в форме C-подобных программ. Результатом трансляции такой программы является не машинный код, а система булевых уравнений, которая в дальнейшем при помощи преобразований Цейтина [17] сводится к SAT-задаче.

Для сведения к SAT перечисленных выше задач в рамках модели конформного поведения (3) можно применить более адресный подход, использующий известные методы булевого кодирования целочисленных неравенств [31]. Кратко опишем здесь

соответствующую технику.

Итак, рассматриваем модель вида (3). Предположим, что осуществляется k итераций синхронного пересчета весов вершин графа G (контактов). Состояние вершины v_i , $i \in \{1, \dots, n\}$, после контакта с номером $t \in \{1, \dots, k\}$ будем кодировать булевой переменной x_i^t ; x_i^0 кодирует начальное состояние v_i .

Чтобы пометать, какие вершины являются агитаторами или лоялистами, введем два дополнительных набора булевых переменных: $\{a_i\}_{i=1}^n$, $\{l_i\}_{i=1}^n$. Полагаем, что при $a_i = 1$, $l_i = 0$ вершина v_i является агитатором, при $a_i = 0$, $l_i = 1$ – лоялистом, а при $a_i = 0$, $l_i = 0$ – простым агентом; ситуация $a_i = 1$, $l_i = 1$ невозможна, и данный факт кодируется дизъюнктом $(\neg a_i \vee \neg l_i)$.

Пусть v_i – простой агент-конформист. Введем обозначение $\Theta_i = \lfloor \theta_i \cdot |V_i| \rfloor$. Переменная x_i^{t+1} принимает значение 1 тогда и только тогда, когда

$$(4) \quad \sum_{v_j \in V_i} x_j^t > \Theta_i.$$

Для булевого кодирования ограничений вида (4) можно использовать различные способы. Фактически мы должны эффективно записать условия истинности предиката $P_{\Theta_i}(x_{j_1}^t, \dots, x_{j_{|V_i|}}^t)$, который истинен на том и только том наборе значений переменных $x_{j_1}^t, \dots, x_{j_{|V_i|}}^t$, на котором выполнено (4). Эти условия истинности могут быть записаны в виде системы булевых уравнений, от которой при помощи преобразований Цейтина [17] делается эффективный переход к одному уравнению вида КНФ = 1, т.е. к некоторой SAT-задаче.

Простейший способ построения такого рода системы уравнений – закодировать алгоритм подсчета числа единиц в векторе $x_{j_1}^t, \dots, x_{j_{|V_i|}}^t$. Однако существуют более эффективные способы. В частности, для кодирования условий вида (4) можно использовать технику работы с так называемыми «cardinality-ограничениями» (cardinality constraints [19, 20, 31, 37, 44]). Наиболее эффективные реализации этой техники, описанные в [19, 31], используют сортирующие сети. Основная идея данного подхода весьма проста и состоит в следующем. Мы можем отсортиро-

вать биты в произвольном булевом векторе b_1, \dots, b_m по возрастанию (полагаем, что старшие биты находятся слева), рассматривая их как натуральные числа из множества $\{0, 1\}$. Пусть (s_1, \dots, s_m) – результат сортировки. Очевидно, что $\sum_{j=1}^m b_j > q$, $q \in \{0, 1, \dots, m-1\}$, тогда и только тогда, когда $s_{q+1} = 1$. Остается выбрать алгоритм сортировки с наиболее компактной булевой кодировкой. Наилучшим в этом смысле является алгоритм, использующий сортирующую сеть Батчера [10, 21].

При кодировании работы сортирующей сети со входом (b_1, \dots, b_m) и выходом (s_1, \dots, s_m) возникает $O(m \cdot \log^2 m)$ дополнительных переменных и $O(m \cdot \log^2 m)$ дизъюнктов. Легко понять, что в общем случае при кодировании с использованием сортирующих сетей k контактов в рассматриваемой модели конформного поведения и число переменных, и число дизъюнктов будут ограничены сверху величиной $O(k \cdot n^2 \cdot \log^2 n)$. Учитывая, что основной интерес, в силу утверждения 1, представляют ситуации, когда $k \leq n - A - L$, можно считать SAT-подход вполне применимым для исследования моделей вида (3) с графами на сотнях вершин.

Для преобразования в SAT условий вида

$$\begin{aligned} wt_\chi(w(0)) &\leq P, \\ wt_\chi(w(0)) &\geq Q, \\ wt_\chi(a_1, \dots, a_n) &\leq A, \\ wt_\chi(l_1, \dots, l_n) &\leq L, \end{aligned}$$

также применима техника булевого кодирования cardinality-ограничений, использующая сортирующие сети.

3.2. АЛГОРИТМЫ РЕШЕНИЯ SAT-ЗАДАЧ

Здесь мы кратко останавливаемся на наиболее эффективных алгоритмах, используемых для решения SAT-задач. Практически в основе всех современных «промышленных» SAT-решателей, гарантирующих точное решение произвольной SAT-задачи, лежит алгоритм DPLL [26], а точнее его нехронологические версии, базирующиеся на идеях, впервые высказанных в работе [36].

Сам по себе алгоритм DPLL – это направленный обход дерева поиска с бэктрекингом и правилом распространения ограничений, названным впоследствии «Unit Propagation» [28]. Описанный в работе [36] алгоритм GRASP в дополнение к DPLL использует память для хранения информации о ходе поиска в форме булевых ограничений, называемых конфликтными дизъюнктами. Конфликтные дизъюнкты позволяют точно выявлять ответственные за конфликт присвоения, что дает в ряде случаев возможность откатываться на более ранние уровни решения, чем уровень, который предшествует конфликтному уровню. Возможность такого рода «глубоких откатов» получила название «нехронологический бэктрекинг» или «бэкджампинг». В работе [47] был описан целый ряд конструкций, дополняющих общие идеи GRASP-а, что привело в итоге к возникновению нового поколения SAT-решателей, успешно применяемых к широкому классу комбинаторных задач.

Эффективность любого комбинаторного алгоритма должна подтверждаться его применимостью к решению аргументированно трудных тестов. Под аргументированно трудным мы понимаем тест, эквивалентный задаче, относительно которой есть четкая уверенность в ее высокой вычислительной сложности. Хорошим классом таких тестов являются задачи криптоанализа. В последние несколько лет растет интерес к применению SAT-подхода для решения этих задач: [13, 14, 38, 39, 43, 45].

Еще одну немаловажную положительную черту SAT-подхода составляют весьма естественные стратегии распараллеливания SAT-задач. Это позволяет использовать для их решения интенсивно развивающиеся в последние годы параллельные и распределенные вычислительные технологии и системы. В работах [13, 43] для криптоанализа алгоритма поточного шифрования A5/1 был использован распределенный SAT-решатель. Данный подход получил развитие в виде проекта добровольных распределенных вычислений SAT@home [9, 41]. Решению SAT-задачи в SAT@home предшествует стадия препроцессинга, на которой ищется декомпозиционное множество, используемое для распа-

ралливания исходной задачи. При этом применяется специальная техника прогнозирования трудоемкости распределенного решения SAT-задач, базирующаяся на методе Монте-Карло [15].

Ряд примеров успешного применения SAT-подхода для поиска неподвижных точек и циклических режимов дискретно-автоматных отображений, используемых для моделирования динамики генных сетей, содержится в работах [7, 8].

4. Вычислительные эксперименты

Предложенная выше методика анализа дискретно-автоматных моделей конформного поведения была протестирована на случайным образом сгенерированных тестах. Конечно, такие тесты не связаны с реальными ситуациями коллективного поведения. Однако основной нашей целью на данном этапе была демонстрация принципиальной возможности решения соответствующих комбинаторных задач для размерностей, простой перебор для которых невозможен в принципе. На текущий момент мы никак не задействовали параллельные вычисления. Кроме этого, для рассматриваемых классов задач хорошие результаты могут давать различные неполные алгоритмы, которые также пока не применялись.

Во всех примерах рассматривались графы на $n = 100$ вершинах, которые генерировались по схеме, схожей с известной моделью Эрдеша–Реньи (G_{np} -графы, [27, 40]). А именно, в матрице смежности любая клетка, не находящаяся на главной диагонали, заполнялась единицей с вероятностью p и нулем с вероятностью $1 - p$; на главной диагонали везде ставились нули, что соответствовало отсутствию петель в графе. Отличие от модели Эрдеша–Реньи лишь в том, что генерируемые графы являются ориентированными, и их матрицы смежности не обязаны быть симметрическими. Конформность агентов также расставлялась случайным образом – для каждой вершины параметр θ_i генерировался как случайное число из отрезка $[0, 1]$.

Для каждой из задач 1–4 были сгенерированы по 10 тестов для значений параметра $p = 0, 2, p = 0, 3, p = 0, 5$. Во всех вы-

282

числениях использовался SAT-решатель minisat2.2 [46], который запускался на одном ядре Core i7-3770k (16 Gb RAM, Ubuntu 12.04). При этом было выставлено ограничение по времени в 1200 секунд (20 минут), по достижении которого вычисление прерывалось с результатом «решение не найдено». В приведенных ниже таблицах содержатся средние по 10 тестам (для каждого значения p) данные.

В таблице 1 представлены результаты вычислительных экспериментов для задачи 1 с параметрами $n = 100$, $P = 40$, $Q = 80$ и $k = 10$. Отметим, что среди КНФ, SAT-задачи для которых были дорешены за отведенный лимит времени, были как выполнимые, так и невыполнимые. Таким образом, в этих случаях SAT-решатель либо находил соответствующее размещение агентов, являющееся решением задачи, либо доказывал отсутствие такого размещения.

Таблица 1. Результаты вычислительных экспериментов для задачи 1

Вероятность дуги	Средний размер КНФ, Кб	Решено тестов	Среднее время решения, с
0,2	20049	10/10	144,57
0,3	35906	5/10	618,15
0,5	69047	7/10	487,40

В тестах по задаче 2 использовались следующие параметры: $n = 100$, $A = 20$, $\alpha = 0,8$, $k = 10$. Результаты тестов показаны в таблице 2.

Таблица 2. Результаты вычислительных экспериментов для задачи 2

Вероятность дуги	Средний размер КНФ, Кб	Решено тестов	Среднее время решения, с
0,2	20471	10/10	4,08
0,3	34844	10/10	18,63
0,5	68980	10/10	97,19

Для каждого теста по задаче 2, используя найденное размещение агитаторов, строился соответствующий тест для задачи 3 с $L = 15$. Отметим, что неподвижные точки находились очень быстро. Соответствующие результаты представлены в таблице 3.

Таблица 3. Результаты вычислительных экспериментов для задачи 3

Вероятность дуги	Средний размер КНФ, Кб	Решено тестов	Среднее время решения, с
0,2	2093	10/10	0,07
0,3	3373	10/10	0,20
0,5	6468	10/10	0,20

Используя размещения агитаторов, найденные в тестах по задаче 2, строились тесты для задачи 4 с $L = 30$, $\beta = 0,2$ и $k = 10$. Соответствующие результаты приведены в таблице 4.

Таблица 4. Результаты вычислительных экспериментов для задачи 4

Вероятность дуги	Средний размер КНФ, Кб	Решено тестов	Среднее время решения, с
0,2	20166	10/10	1,22
0,3	34540	10/10	2,74
0,5	68642	10/10	35,05

Еще раз акцентируем внимание на представленных в таблице 4 результатах. Итак, на первом этапе (т.е. при решении задачи 2) мы нашли такое расположение 20 агитаторов, что за 10 контактов система переходила в состояние, в котором более 80 агентов действовали (на этом этапе в системе не было лоялистов). Затем мы предполагали, что из 80 простых агентов-конформистов некоторые 30 агентов становятся лоялистами. Требовалось найти такое расположение этих 30 агентов, чтобы система из состояния действия 50 оставшихся простых агентов и 20 агитаторов за 10 контактов перешла в состояние, в котором решение «действовать» принимают только 20 агитаторов. Во всех тестовых задачах

соответствующие конфигурации были найдены.

5. Заключение

Статья посвящена исследованию коллективного поведения с позиций дискретно-автоматных моделей, схожих по своим свойствам с дискретными моделями генных сетей, изучаемыми в информационной биологии. Мотивация работы с данным классом моделей заключается в том, что для их численного исследования может быть применен аппарат символьных вычислений, демонстрирующий хорошие результаты на аргументированно трудных тестах из символьной верификации и криптоанализа.

В статье подробно рассмотрена простейшая дискретно-автоматная модель, в рамках которой естественным образом могут быть поставлены задачи описания поведения «небезопасных социальных групп». В частности, рассмотрена задача поиска начальных состояний с относительно малым числом действующих агентов, из которых за небольшое число контактов возможны переходы в состояния с большинством действующих агентов. Далее в модель введены агенты, никогда не меняющие своего состояния (действия – агитаторы либо бездействия – лоялисты). Для этого варианта модели установлено, что для достижения ею стационарного состояния из состояния начальной упорядоченности требуется не более $n - A - L$ контактов (n – общее число агентов, A – число агитаторов, L – число лоялистов). Если достигнутое стационарное состояние таково, что более $\alpha \cdot n$, $\alpha \in (0, 1)$, агентов в нем, например, действуют, притом что в начальном состоянии все простые агенты бездействовали, то соответствующая система называется (A, L, α) -критической. Системы данного типа при α , близких к 1, можно считать небезопасными в том плане, что для них можно указать размещение агитаторов, которое переведет почти всех бездействующих простых агентов в состояние действия. Однако в некоторых случаях ситуацию можно исправить, сделав относительно малую часть простых агентов лоялистами – система из состояния действия всех простых агентов через некоторое число контактов переходит в состояние, в котором число действу-

ющих агентов не превосходит $\beta \cdot n$, где $\beta < \alpha$.

На наш взгляд немаловажный аспект моделирования коллективного поведения составляет проблема построения таких моделей (безусловно, адекватных реальным ситуациям), для которых возможно эффективное их численное исследование. Причем важно не только наблюдать за их эволюцией во времени (что можно считать простой «прогонкой» модели), но и уметь находить конфигурации, соответствующие различным «экстремальным» формам поведения модели (например, искать как устойчивые, так и неустойчивые состояния, циклические режимы и т.п.). В ряде работ для решения этих проблем используется теоретико-игровой подход [1–5, 24] (поиск равновесий по Нэшу, состояний, оптимальных по Парето, и др.). Для моделей, рассматриваемых в настоящей статье, данные задачи решаются без привлечения теории игр, а лишь отталкиваясь от весьма простых свойств дискретных функций.

В будущем интерес представляет развитие предложенного подхода в ряде естественных направлений. Например, не составит большого труда при трансляции в SAT учитывать различные виды нагрузки дуг графа G , которая может отображать, например «социальное давление» (как в [4]) либо какие-то другие факторы взаимного влияния агентов. Также интерес представляет более детальный анализ возможности замены простых агентов на лоялистов в задаче 4. В частности, в будущем можно будет учитывать в модели фактор «цены» перехода от простого агента к лоялисту – в реальных ситуациях различные переходы такого типа могут быть существенно неравноценными.

Сформулированное выше условие критичности для систем с агитаторами и лоялистами можно усилить, используя формализм дважды квантифицированных булевых формул (2-QBF) [22]. Скажем, если обозначить через S – размещение A агитаторов, а через T – размещение L лоялистов, то усиленное условие «критичности» системы может выглядеть следующим образом:

$$\exists S \forall T \mathfrak{R}(G, f_g, S, T, \alpha) = 1,$$

где \mathfrak{R} – предикат, принимающий значение «истина», если си-

стема, будучи начально-упорядоченной с бездействием простых агентов, переходит в критическое (с параметром α) состояние относительно действия. Для проверки наличия этого свойства, в соответствии с утверждением 1, достаточно рассмотреть поведение системы при $\leq n - A - L$ контактах. Для численного исследования достижимости такого рода ситуаций могут быть использованы известные 2-QBF-решатели (например, [32, 34] и др.).

На данном этапе для решения сформулированных в работе задач использовались стандартные «промышленные» SAT-решатели (в частности, minisat2.2, [46]), основной областью применения которых является символьная верификация. Данные решатели всегда выдают точное решение SAT-задачи, т.е. либо находят выполняющий КНФ набор, либо доказывают ее невыполнимость.

Отметим, что в рассмотренных в статье задачах в случае существования одного решения зачастую существует много других решений. Для таких задач вполне могут дать хорошие результаты различные неполные алгоритмы – например, те или иные стратегии локального поиска. В ближайшем будущем предполагается использовать такие алгоритмы для работы с дискретно-автоматными моделями конформного поведения, существенно превосходящими по размерности модели, рассмотренные в статье.

Авторы выражают глубокую благодарность Владимиру Валентиновичу Брееру за его ценные замечания, позволившие устранить целый ряд неточностей, имевшихся в первоначальной версии работы.

Литература

1. БРЕЕР В.В. *Теоретико-игровая модель неанонимного порогового конформного поведения* // Управление большими системами. – 2010. – № 31. – С. 162–176.
2. БРЕЕР В.В., НОВИКОВ Д.А. *Пороговые модели взаимного страхования* // Математическая теория игр и ее приложения. – 2011. – Том 3, №4. – С. 3–22.

3. БРЕЕР В.В., НОВИКОВ Д.А. *Пороговая модель коррупционного поведения* // Системы управления и информационные технологии. – 2011. – №3. – С. 73–75.
4. БРЕЕР В.В. *Теоретико-игровые модели конформного поведения* // Автоматика и телемеханика. – 2012. – №10. – С. 111–126.
5. ГУБАНОВ Д.А., НОВИКОВ Д.А., ЧХАРТИШВИЛИ А.Г. *Социальные сети: модели информационного влияния, управления и противоборства*. – М.: Физматлит. 2010. – 228 с.
6. ГРИГОРЕНКО Е.Д., ЕВДОКИМОВ А.А., ЛИХОШВАЙ В.А., ЛОБАРЕВА И.А. *Неподвижные точки и циклы автоматных отображений, моделирующих функционирование генных сетей* // Вестник Томского гос. ун-та. Приложение. – 2005. – №14. – С. 206–212.
7. ЕВДОКИМОВ А.А., КОЧЕМАЗОВ С.Е., СЕМЕНОВ А.А. *Применение символьных вычислений к исследованию дискретных моделей некоторых классов генных сетей* // Вычислительные технологии. – 2011. – Т. 16, №1. – С. 30–47.
8. ЕВДОКИМОВ А.А., КОЧЕМАЗОВ С.Е., ОТПУЩЕННИКОВ И.В., СЕМЕНОВ А.А. *Символьные алгоритмы решения булевых уравнений в применении к исследованию дискретных моделей генных сетей* // Материалы XVI Международной конференции «Проблемы теоретической кибернетики». Нижний Новгород, 2011. – С. 151–154.
9. ЗАЙКИН О.С., СЕМЕНОВ А.А., ПОСЫПКИН М.А. *Процедуры построения декомпозиционных множеств для распределенного решения SAT-задач в проекте добровольных вычислений SAT@HOME* // Управление большими системами. – 2013. – Т. 43. – С. 138–156.
10. КОРМЕН Т., ЛЕЙЗЕРСОН Ч., РИВЕСТ Р. *Алгоритмы. Построение и анализ*. – М. МЦНМО, 2002. – 960 с.
11. КРАСНОЩЕКОВ П.С. *Простейшая математическая модель поведения. Психология конформизма* // Математическое моделирование. – 1998. – Т. 10, №7. – С. 76–92.

12. ОТПУЩЕННИКОВ И.В., СЕМЕНОВ А.А. *Технология трансляции комбинаторных проблем в булевы уравнения* // Прикладная дискретная математика. – 2011. – №1. – С. 96–115.
13. ПОСЫПКИН М.А., ЗАЙКИН О.С., БЕСПАЛОВ Д.В., СЕМЕНОВ А.А. *Решение задач криптоанализа поточных шифров в распределенных вычислительных средах* // Труды ИСА РАН. – 2009. – № 46. – С. 119–137.
14. СЕМЕНОВ А.А., ЗАЙКИН О.С., БЕСПАЛОВ Д.В., УШАКОВ А.А. *SAT-подход в криптоанализе некоторых систем поточного шифрования* // Вычислительные технологии. – 2008. – Т. 13, №6. – С. 134–150.
15. СЕМЕНОВ А.А., ЗАЙКИН О.С. *Алгоритмы построения декомпозиционных множеств для крупноблочного распараллеливания SAT-задач* // Известия ИГУ. Серия: Математика. – 2012. – Т. 5, №4. – С. 79–94.
16. *Системная компьютерная биология* / Под ред. Н.А. Колчанова, С.С. Гончарова, В.А. Лихошвая, В.А. Иванисенко. – Новосибирск: Изд-во СО РАН, 2008. – 767 с.
17. ЦЕЙТИН Г.С. *О сложности вывода в исчислении высказываний* // Записки научных семинаров ЛОМИ АН СССР. – 1968. – Т. 8. – С. 234–259.
18. ЯБЛОНСКИЙ С.В. *Введение в дискретную математику*. – М.: Наука, 1986. – 384 с.
19. ASIN R., NIEUWENHUIS R., OLIVERAS A., RODRIGUEZ-CARBONELL E. *Cardinality Networks: a theoretical and empirical study* // Constraints – 2011. – Vol. 16, №2. – P. 195–221.
20. BAILLEUX O., BOUFGHAD Y. *Efficient CNF encoding of boolean cardinality constraints* // LNCS. – 2003. – Vol. 2833. – P. 108–122.
21. BATCHER K.E. *Sorting Networks and their Applications* // In Proc. of AFIPS. – 1968. – Vol. 32. – P. 307–314.
22. *Handbook of Satisfiability*. // Eds. Biere A., Heule V., van Maaren H., Walsh T. – IOS Press, 2009. – 980 p.

23. BRAUN N. *Individual Thresholds and Social Diffusion // Rationality and Society* – 1995. – №7. – P. 167–182. 6
24. CHWE M. *Structure and Strategy in Collective Action // AJS* – 1999. – Vol. 105, №1. – P. 128–156.
25. COOK S.A *The complexity of theorem-proving procedures // Third annual ACM symposium on Theory of computing.* – Ohio, USA. ACM, 1971. – P. 151–159.
26. DAVIS M., LOGEMANN G., LOVELAND D. *A machine program for theorem proving // Communication of the ACM.* – 1962. – Vol. 5, Issue 7. – P. 394–397.
27. DOROGOVTSEV S.N., GOLTSEV A.V., MENDES J.F.F. *Critical phenomena in complex networks // Rev. Mod. Phys.* – 2008. – Vol. 80. – P. 1275–1335.
28. DOWLING W., GALLIER J. *Linear-time algorithms for testing the satisfiability of propositional Horn formulae // Journal of Logic Programming.* – 1984. – №1(3). – P. 267–284.
29. DUBROVA E., TESLENKO M., MARTINELLI A. *Kauffman networks: analysis and applications // Proc. Of ICCAD – 2005.* – P. 479–484.
30. DUBROVA E., TESLENKO M. *A SAT-Based Algorithm for Finding Attractors in Synchronous Boolean Networks // IEEE/ACM Transactions on Computational Biology and Bioinformatics.* – 2011. – Vol. 8, №5. – P. 1393–1399.
31. EEN N., SORENSSON N. *Translating Pseudo-Boolean Constraints into SAT // Journal on Satisfiability, Boolean Modeling and Computation.* – 2006. – Vol. 2. – P. 1–26.
32. GIUNCHIGLIA E., MARIN P., NARIZZANO M *Reasoning with quantified boolean formulas. In Handbook of Satisfiability (editors: A. Biere, M.Heule, H. van Maaren, T. Walsh).* – IOS Press. – 2009. – P. 761–780.
33. GRANOVETTER M. *Threshold Models of Collective Behavior // AJS* – 1978. – Vol. 83, №6. – P. 1420–1443.
34. JANOTA M., MARQUES-SILVA J.P. *Abstraction-Based Algorithm for 2QBF // LNCS.* – 2011. – Vol. 6695. – P. 230–

- 244.
35. KAUFFMAN S. A. *Metabolic stability and epigenesis in randomly constructed genetic nets* // Theor. Biol. – 1969. – Vol. 22, №3. – P. 437–467.
 36. MARQUES-SILVA J.P., SAKALLAH K.A. *GRASP: A search algorithm for propositional satisfiability* // IEEE Transactions on Computers. – 1999. – Vol. 48, №5. – P. 506–521.
 37. MARQUES-SILVA J., LYNCE I. *Towards Robust CNF Encodings of Cardinality Constraints* // LNCS. – 2007. – Vol. 4741. – P. 483–497.
 38. MCDONALD C., CHARNES C., PIEPRZYK J. *Attacking Bivium with MiniSat* // Tech. Rep. – 2007/040 – ECRYPT Stream Cipher Project. 2007.
 39. MIRONOV I., ZHANG L. *Applications of SAT Solvers to Cryptanalysis of Hash Functions* // LNCS. – 2006. – Vol. 4121 – P. 102–115.
 40. NEWMAN M.E.J. *The structure and function of Complex Networks* // SIAM Review. – 2003. – Vol. 45, №2. – P. 167–256.
 41. POSYPKIN M., SEMENOV A., ZAIKIN O. *Using BOINC desktop grid to solve large scale SAT problems.* // Computer Science Journal. – 2012. – Vol. 13, № 1. – P. 25–34.
 42. PRESTWICH S. *CNF encodings* // In Handbook of Satisfiability / Eds. A.Biere, M.Heule, H. van Maaren, T. Walsh). – IOS Press, 2009. – P. 75–97.
 43. SEMENOV A., ZAIKIN O., BESPALOV D., POSYPKIN M. *Parallel logical cryptanalysis of the generator A5\1 in BNB-Grid system* // LNCS. – 2011. – Vol. 6873. – P. 473–483.
 44. SINZ C. *Towards an Optimal CNF Encoding of Boolean Cardinality Constraints* // LNCS. – 2005. – Vol. 3709. – P. 827–831.
 45. SOOS M., NOHL K., CASTELLUCCIA C. *Extending SAT Solvers to Cryptographic Problems* // LNCS. – 2009. – Vol. 5584. – P. 244–257.
 46. *The MiniSat Page* [Электронный ресурс] URL:

<http://www.minisat.se/> (дата обращения: 15.10.2013).

47. ZHANG L., MADIGAN C.F., MOSKEWICZ M.H., MALIK S. *Efficient conflict driven learning in a boolean satisfiability solver* // In Proc. of ICCAD. – 2001. – P. 279–285.

ANALYSIS OF SOME DISCRETE-AUTOMATON MODELS OF COLLECTIVE BEHAVIOR

Alexander Semenov, Institute of System Dynamics and Control Theory SB RAS, Irkutsk, Cand.Sc., assistant professor (biclop Rambler@yandex.ru).

Stepan Kochemazov, Institute of System Dynamics and Control Theory SB RAS, Irkutsk, researcher (veinamond@gmail.com).

Abstract: We analyze the phenomenon of collective behavior using a discrete-automaton model. This model is similar to those used in modern computational biology to explain dynamical processes in gene networks. Within the proposed framework we study a multi-agent system, which contains not only conformists who make their decisions depending on other agents' opinions but also agents with fixed decisions (the, so called, «agitators» and «loyalists»). We solve the problems of of small number of agitators (loyalists) placement, which forces the system to transit to a state where more than a half of agents is active (inactive). Our computational method based on SAT-approach made it possible to solve corresponding problems for systems with a hundred vertices with PC.

Keywords: models of collective behavior, discrete-automaton models, symbolic algorithms, SAT.

Статья представлена к публикации членом редакционной коллегии Д. А. Новиковым

Поступила в редакцию 26.08.2013.

Опубликована 30.11.2013.

УДК 004.032.26 + 338.27 + 004.896
ББК 31.15

**МОДЕЛИРОВАНИЕ И ОПТИМИЗАЦИЯ
УПРАВЛЕНИЯ ИНТЕЛЛЕКТУАЛЬНОЙ ГИБРИДНОЙ
ЭНЕРГОСИСТЕМОЙ С ИСТОЧНИКАМИ
ВОЗОБНОВЛЯЕМОЙ ЭНЕРГИИ**

Май Н. Т.¹, Ха В. М.², Камаев В. А.³, Щербаков М. В.⁴,
*(Волгоградский государственный технический
университет, Волгоград, Россия)*

Тхай Куанг Винь⁵
*(Институт информационной технологии, Ханой,
Вьетнам)*

Цель исследования заключается в минимизации потерь и затрат на потребление электроэнергии. Представлена интеллектуальная система управления процессом переключения между источниками энергии в реальном времени с учетом прогнозированных данных о потреблении и производстве электроэнергии. В качестве метода управления используется генетический алгоритм для формирования оптимальной стратегии покупки электроэнергии. Для обоснования предлагаемого подхода выполнены испытания на разработанной модели гибридной энергетической установки.

¹ Нгок Тханг Май, аспирант (Волгоград, пр. Ленина, д.65, тел. (8442) 24-81-00, mnthang464@mail.ru).

² Ван Муон Ха, студент (Волгоград, пр. Ленина, д.65, тел. (8442) 24-81-00, muon.ha@mail.ru).

³ Валерий Анатольевич Камаев, доктор технических наук, профессор (Волгоград, пр. Ленина, д.65, тел. (8442) 24-81-00, kataev@unix.cad.vstu.ru).

⁴ Максим Владимирович Щербаков, кандидат технических наук (Волгоград, пр. Ленина, д.65, тел. (8442) 24-81-00, vstu@gebeus.ru).

⁵ Куанг Винь Тхай, директор института информационной технологии (Ханой, Вьетнам, tqvinh@ioit.ac.vn).

Ключевые слова: гибриды, интеллектуальная энергосистема, источник возобновляемой энергии, нейронная сеть, прогноз, оптимизация, HRES.

1. Введение

Эффективное управление энергосистемами является важной задачей в энергетической политике многих стран в связи с глобальным потеплением и значительным увеличением цен на традиционные источники энергии. Однако управление энергопотоками в подобных системах является сложной задачей, так как зависит от различных факторов. Более того, процесс управления энергосистемой также как и выбор источника электроэнергии, должен осуществляться в режиме реального времени для обеспечения максимального баланса между поставщиком и потребителем.

В работе предлагается интеллектуальная гибридная энергосистема с источниками возобновляемой энергии, функционирование которой основано на прогнозе потребления и производства электроэнергии в системе. На основе модели энергосистемы можно формировать прогноз потребления, производства электроэнергии, а также вырабатывать оптимальную стратегию переключения между источниками электроэнергии в системе.

2. Интеллектуальная гибридная энергетическая система

2.1. СТРУКТУРА ИНТЕЛЛЕКТУАЛЬНОЙ ГИБРИДНОЙ ЭНЕРГОСИСТЕМЫ

Под интеллектуальной гибридной энергосистемой с источниками возобновляемой энергии будем понимать энергосистему, которая объединяет множество традиционных источников и источников возобновляемой энергии в единую энергосистему с применением устройств, реализующих интеллектуальные алгоритмы для управления компонентами системы.

В данной работе гибридная энергосистема, схема которой представлена на рис. 1, включает в себя следующие компоненты: солнечные панели (1); аккумуляторные батареи (АКБ) (2); контроллер заряда-разряда (3); метеостанция (4); цифровой счётчик электроэнергии (5, 6); двунаправленный инвертор (7) и интеллектуальный контроллер (8) для управления энергопотоками; сервер (9).

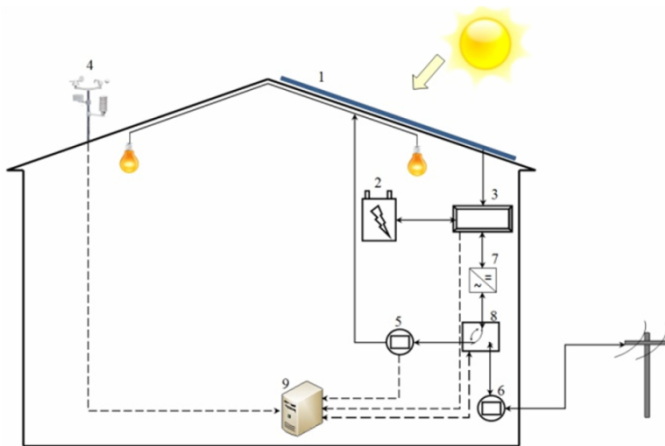


Рис. 1. Схема гибридной энергосистемы

2.2. ХАРАКТЕРИСТИКИ ОСНОВНЫХ КОМПОНЕНТОВ СИСТЕМЫ

Внешняя электросеть. Внешняя электросеть подключается к системе при условии недостаточной мощности других источников (например, солнечных панелей), а батарейный блок уже разряжен до минимального значения (10%).

Солнечная панель (СП). Солнечная панель характеризуется номинальной мощностью. Номинальная мощность характеризует количество электроэнергии, которое может произвести фотоэлектрический модуль при оптимальных условиях. Номинальная мощность солнечной панели зависит от солнечного облучения и температуры на поверхности панели. Диаграмма зависимости номинальной мощности солнечной панели 100 Вт от солнечного облучения при условии, что

температура на поверхности панели равна 25 °С, представлена на рис. 2 [7].

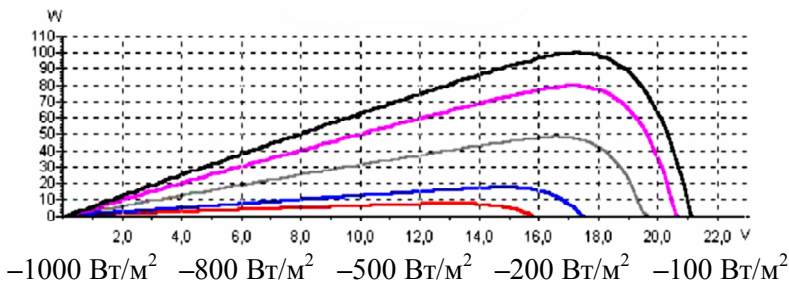


Рис. 2. Зависимость номинальной мощности солнечной панели от солнечного облучения

Мощность СП определяется по формуле [5]:

$$(1) \quad P_{\text{солн}} = \begin{cases} P_{\text{СТ}} \frac{G_{\text{СИ}}}{G_{\text{СТ}}} (1 + \alpha(T_{\text{СП}} - T_{\text{СИ}})), & \text{если } G_{\text{СИ}} \geq C, \\ 0, & \text{если } G_{\text{СИ}} < C; \end{cases}$$

где $P_{\text{солн}}$ – мощность, вырабатываемая солнечной панелью (Вт);

$P_{\text{СТ}}$ – мощность солнечной панели в стандартных условиях (Standard Test Condition – STC), т.е. мощность солнечной радиации 1000 Вт/м², температура элементов – 25 °С и солнечный спектр на широте 45°) (Вт) [8];

$G_{\text{СИ}}$ – коэффициент солнечного излучения (лк);

$G_{\text{СТ}}$ – коэффициент солнечного излучения в стандартных условиях (лк);

α – температурный коэффициент (°С);

$T_{\text{СП}}$ – температура поверхности солнечной панели (°С);

$T_{\text{СИ}}$ – устанавливается равной 25 °С (температура – в стандартных условиях);

C – коэффициент производства энергии, который зависит от характеристики солнечной панели (безразмерный).

Блок аккумуляторных батарей (АКБ). АКБ обеспечивают запас электрической энергии, необходимой для электроснабжения потребителей в ночное время, в облачную погоду. Также обеспечивается компенсация пиковых нагрузок, которые не могут быть покрыты фотоэлектрическими

модулями. Кроме этого, АКБ выполняет роль стабилизатора напряжения на нагрузке, так как выходное напряжение СП в соответствии с его вольт-амперной характеристикой может изменяться в широких пределах. АКБ характеризуется степенью зарядки (SOC – Stage Of Charge). Во избежание снижения способности работы АКБ необходимо выполнять следующие условия:

- изменение степени зарядки в час не должно превышать 10% максимального значения;
- степени зарядки не должно быть меньше 10% максимального значения.

Контроллер заряда-разряда. Контроллер заряда-разряда – электронное устройство, которое защищает АКБ от чрезмерной зарядки и глубокой разрядки, что позволяет значительно продлить срок службы аккумулятора. Контроллер заряда-разряда обеспечивает защиту от коротких замыканий в нагрузке и от подключения АКБ обратной полярностью.

Метеостанция. Метеостанция (может быть цифровой) – комплекс для ежедневного мониторинга погоды, температуры, давления и влажности воздуха. В памяти метеостанции сохраняются данные о погоде за все время ее использования. Данные о погоде могут использоваться для прогнозирования производства электроэнергии солнечными панелями.

Цифровой счётчик электроэнергии. Цифровой счётчик электроэнергии применяется для мониторинга, а также статистических исследований, таких как вычисление средней мощности потребления нагрузки и ее дисперсии, хранения информации о накопленной энергии за произвольные промежутки времени. В экспериментах, рассмотренных в статье, используются многотарифные счетчики электроэнергии.

Двунравленный инвертор. Для преобразования постоянного тока, вырабатываемого солнечной батареей, в переменный ток синусоидальной формы необходим инвертор. Основными техническими характеристиками инвертора являются величина напряжения на входе и выходе, выходная частота и мощность. Качественный инвертор должен обеспечивать высокий КПД и стабилизацию выходного

напряжения, низкий коэффициент гармоник и способность выдерживать возможные перегрузки.

Переключатель (реле). Переключатель – механизм, управляемый контроллером и осуществляющий переключение энергетических потоков в соответствии с командами, поступающими от сервера.

Сервер. Сервер применяется для обработки и хранения данных из метеостанции и счётчика электроэнергии, прогнозирования потребления и производства электроэнергии в ближайшее время, а также форматирования стратегии покупки электроэнергии в зависимости от тарифного плана.

3. Построение модели интеллектуальной гибридной энергетической системы

В статье интеллектуальная гибридная энергетическая система моделируется в среде симуляции PVSOL. Среда PVSOL обеспечивает возможность построения гибридной энергетической системы, построения 3D-модели и моделирование с учетом реальных данных о погоде (температуры, коэффициента солнечного облучения, скорости ветра).

Объект моделирования – энергетическая система корпуса «В» Волгоградского государственного технического университета (ВолгГТУ) в 2011 году. Входные данные моделирования – данные о погоде и о потреблении электроэнергии здания по часам. Данные о погоде измеряются метеостанцией «Волгоград первый». Выходные данные – стратегия покупки электроэнергии, представлена в виде массива из 24 элементов, соответствующих 24 часам в день. Значение элемента массива равно –1 при продаже электроэнергии из внешней сети; 0 при использовании собственной электроэнергии; 1 при покупке электроэнергии.

Модель корпуса «В» ВолгГТУ с установленной гибридной энергетической системой представлена на рис. 3.

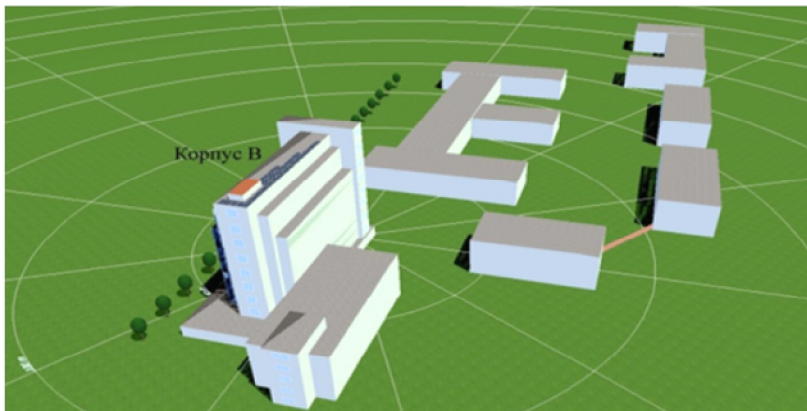


Рис. 3. Модель корпуса «В» ВолГТУ

Производство электроэнергии солнечными панелями зависит от погодных условий. По данным о погоде можно определить место нахождения и угол наклона солнечных панелей, а также максимальную мощность производства электроэнергии. Диаграмма зависимости погодных условий (среднемесячной температуры воздуха, коэффициента солнечного облучения и скорости ветра) от времени в месте проведения испытания за каждый месяц 2011 года показана на рис. 4.

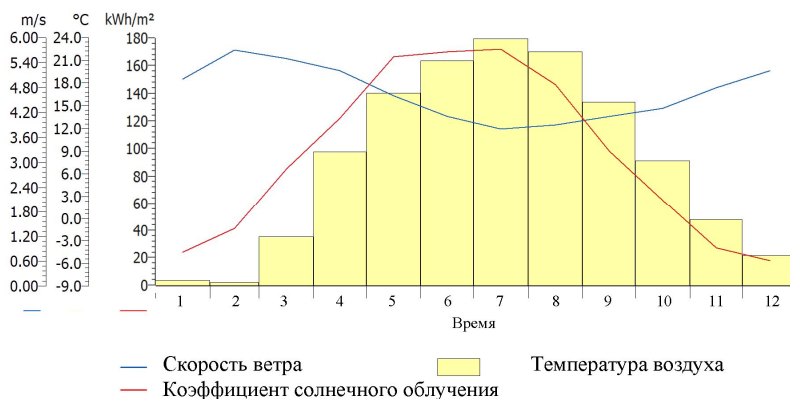


Рис. 4. Зависимость погодных условий от времени в 2011 году

По результатам анализа данных об электропотреблении в [2] предлагается, что:

– электропотребление в выходные и в праздничные дни меньше чем в рабочие дни (электропотребление в субботу равно 60%, а в воскресенье равно 10% среднего потребления рабочих дней);

– энергопотребление в промежутках времени с 9 до 11 часов и с 17 часов до 21 часа локально максимальное, и, соответственно, это пиковая цена электроэнергии.

Профиль моделируемого потребления электроэнергии корпуса «В» ВолгГТУ в 2011 году показан на рис. 5.

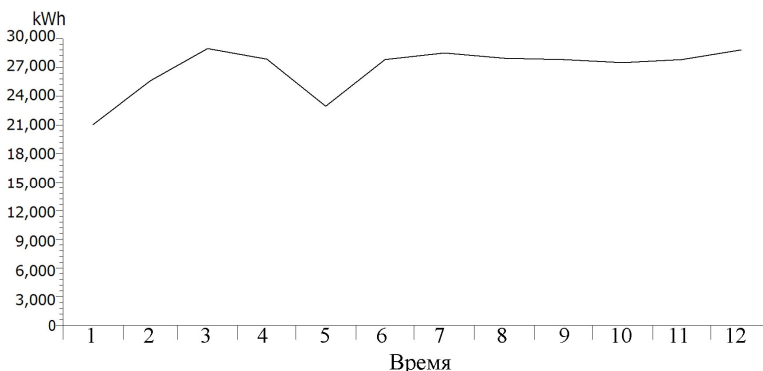


Рис. 5. Профиль потребления электроэнергии корпуса «В» ВолгГТУ за каждый месяц 2011 года

Выходные данные моделирования используются для прогнозирования электропотребления и производства электроэнергии в следующие дни. В работе для формирования прогноза применяются нейронные сети. Оценка качества прогнозирования выполняется с использованием оценок MAE (Mean Absolute Error – средняя абсолютная ошибка) и MSE (Mean Square Error – среднеквадратичное отклонение) [3, 6].

Функция потерь (целевая функция) J определяется по формуле

$$(2) \quad J = \sum_{i=1}^{24} k_i \cdot (p_i^{\text{потр}} - p_i^{\text{произв}}) + \sum_{i=1}^{24} (100\% - S_i) \cdot k_i \cdot E_{\text{макс}}$$

где k_i – тарифный план электроэнергии в промежутке времени i (руб/кВт·ч);

$p_i^{\text{потр}}$ – энергопотребление в промежутке времени $[i, i+1]$ (кВт·ч);

$p_i^{\text{произв}}$ – энергопроизводство в промежутке времени $[i, i+1]$ (кВт·ч);

S_i – состояние АКБ в момент времени i (%);

$E_{\text{макс}}$ – максимальная электроемкость АКБ (кВт·ч или А·ч).

Так как цель исследования заключается в формировании оптимальной стратегии переключения между источниками энергии, формулу (1) можно переписать с учетом «коэффициента стратегии покупки электроэнергии» str_i ($i = 1, \dots, 24$), зависящего от текущего тарифного плана электроэнергии

$$(3) \quad J = \sum_{i=1}^n [(p_i^{\text{потр}} - p_i^{\text{произв}}) + (100\% - S_i) \cdot E_{\text{макс}}] \cdot k_i \cdot str_i,$$

$$(4) \quad str_i = \begin{cases} -1, & \text{при продаже электроэнергии,} \\ 0, & \text{при использовании собственной электроэнергии,} \\ +1, & \text{при покупке электроэнергии.} \end{cases}$$

В каждый момент времени система находится в одном из трех состояний: использование собственной электроэнергии, продажа электроэнергии и покупка электроэнергии. Входные данные системы сохраняется каждый час, таким образом, имеются 24 наблюдения в день. Цель оптимизации стратегии покупки электроэнергии заключается в нахождение таких значений str_i ($i = 1, \dots, 24$) при которых достигается минимальное значение целевой функции J . Количество возможных стратегий покупки электроэнергии равно 3^{24} . Для уменьшения времени вычислений в статье используется генетический алгоритм. Параметры для алгоритма представляются таким образом: количество особей – 100, количество поколений – 1000, вероятность скрещивания – 80%, вероятность мутации – 5%, число генов особи – 24. Для формирования стратегии переключения между источниками электроэнергии выделяются следующие этапы генетического алгоритма:

Шаг 1. Задание целевой функции J для особей популяции.

Шаг 2. Создание начальной популяции.

Шаг 3. Если количество поколений $i < 100$

- {*
- размножение (скрещивание);*
- мутирование;*
- вычисление значений целевой функции для всех особей;*
- формирование нового поколения.*
- }*

Шаг 4. Определение оптимального решения.

Показатель эффективности создания интеллектуальной гибридной энергосистемы E определяется по формуле:

$$(5) \quad E = \frac{C_{\text{без HRES}} - C_{\text{HRES}}}{C_{\text{обор.}}},$$

где $C_{\text{без HRES}}$ – затраты на электроэнергию без применения интеллектуальной гибридной энергетической системы;

C_{HRES} – затраты на электроэнергию с применением интеллектуальной гибридной энергетической системы;

$C_{\text{обор.}}$ – затраты на приобретение оборудования системы.

Затраты на приобретение оборудования системы равны сумме затрат на приобретения компонент гибридной системы:

$$(6) \quad C_{\text{обор.}} = n_{\text{СП}} \cdot C_{\text{СП}} + n_{\text{АКБ}} \cdot C_{\text{АКБ}} + C_0$$

где $n_{\text{СП}}$ – количество солнечных панелей;

$C_{\text{СП}}$ – стоимость одной солнечной панели;

$n_{\text{АКБ}}$ – количество АКБ;

$C_{\text{АКБ}}$ – стоимость одного АКБ;

C_0 – стоимость остальных компонентов системы.

Чем выше показатель эффективности E , тем меньше время окупаемости системы. Следовательно, в статье проводится испытание для определения количества установленных солнечных панелей и АКБ в корпусе «В» ВолгГТУ, а также оптимальной стратегии покупки электроэнергии.

4. Результаты испытаний

Для апробации предлагаемых подходов была построена модель гибридной энергетической системы в среде PVSOL.

Данные о потреблении и производстве электроэнергии измерялись каждый час и сохранялись на сервере [2].

Испытание интеллектуальной гибридной энергетической системы проводилось в течение произвольно выбранного дня 14.9.2011 г. Для прогнозирования потребления и производства электроэнергии 14.9.2011 г. использовались нейронные сети. Согласно [1], на потребление электроэнергии в момент времени t наибольшее влияние оказывает потребление электроэнергии в этот же момент времени 1 неделю, 2 недели, 3 недели и 4 недели назад. Следовательно, входными сигналами нейронных сетей в момент t являются потребление и производство электроэнергии в моменты $t = 24.1.7$, $t = 24.2.7$, $t = 24.3.7$, $t = 24.4.7$. Структура нейронной сети для проведения испытаний показана на рис. 6.

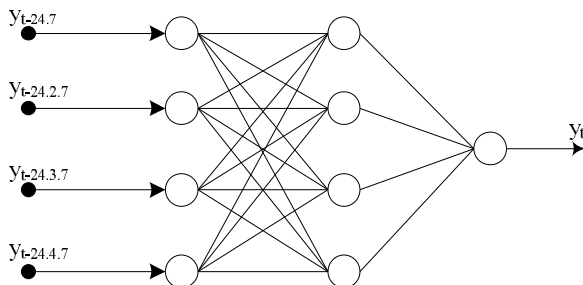


Рис. 6. Структура нейронной сети для прогнозирования потребления и производства электроэнергии

Значения модельных и прогнозируемых данных о потреблении электроэнергии для проведения испытания показаны на рис. 7.

На основе полученных данных в процессе моделирования можно по формуле (1) определить количество электроэнергии, вырабатываемой одной солнечной панелью. Следовательно, можно определить количество электроэнергии, вырабатываемой $n_{СП}$ солнечными панелями.

В статье проводилось 60 испытаний с разными конфигурациями системы (разными количества установленными солнечными панелями и АКБ) для определения оптимальной

конфигурации гибридной энергетической системы корпуса «В» ВолгГТУ. Оценка качества построенной интеллектуальной гибридной энергетической системы определяет коэффициентом эффективности по формуле (5).

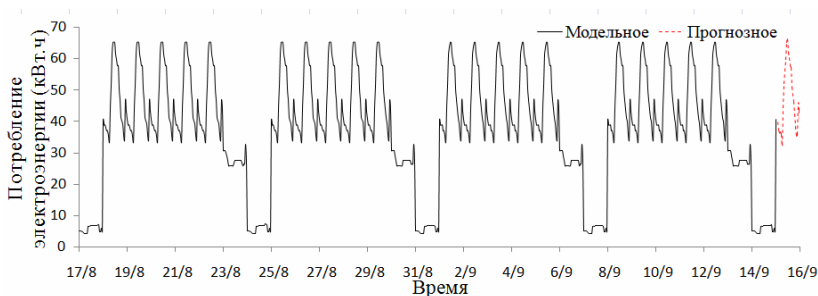


Рис. 7. Модельное потребление электроэнергии в корпуса «В» ВолгГТУ с 17.8.2011 г. по 14.9.2011 г.

В статье стратегия покупки электроэнергии формируется на основе прогнозируемых данных о потреблении и производстве электроэнергии с учетом многотарифного плана электроэнергии. График тарифного плана электроэнергии города Волгограда по трем зонам суток показывается на рис. 8.

На рис. 9. изображена поверхность коэффициентов эффективности (ось Z) в зависимости от числа установленных солнечных панелей (ось X) и максимальной емкости АКБ (ось Y). На поверхности определена точка, характеризующая оптимальную конфигурацию системы: количество солнечных панелей – 110, максимальная емкость АКБ – 458 А.ч, номинальное напряжение АКБ – 24 В.

График производства электроэнергии, вырабатываемой солнечными панелями в оптимальной конфигурации, представлен на рис. 10.

В данной статье использован генетический алгоритм для выбора оптимальной стратегии покупки электроэнергии возобновляемой энергетической системы корпуса «В» ВолгГТУ. В результате функционирования алгоритма оптимальная стратегия покупки электроэнергии в точке максимального коэффициента эффективности представлена на рис. 11.

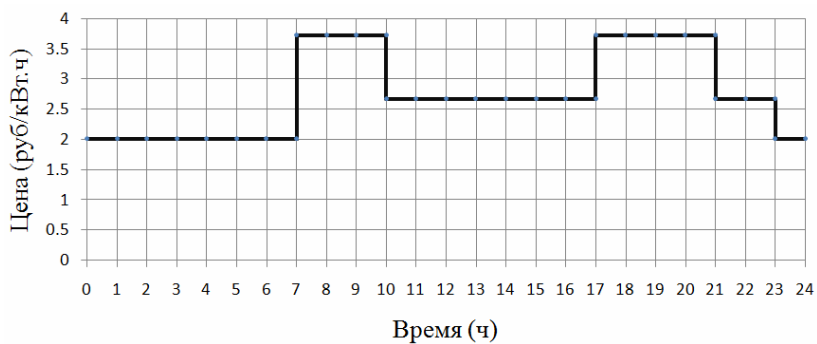


Рис. 8. Тарифный план электроэнергии по трем зонам суток

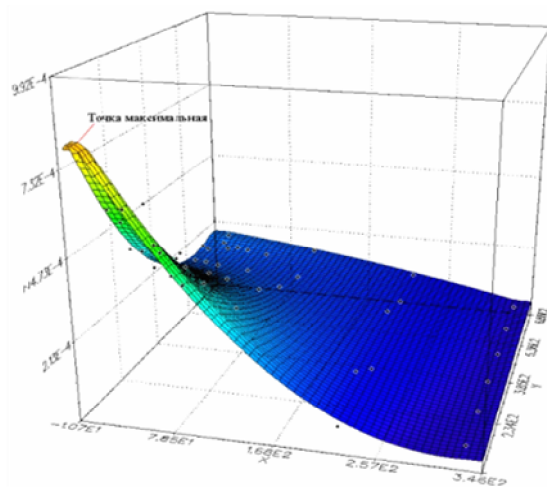


Рис. 9. Поверхность коэффициентов эффективности с разными конфигурациями системы

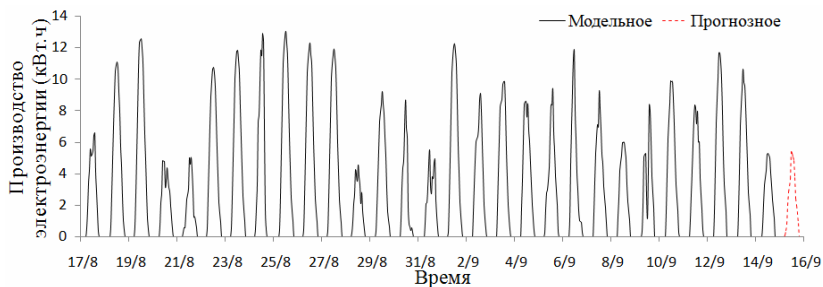


Рис. 10. Производство электроэнергии корпуса «В» ВолгГТУ с 17.8.2011 г. по 14.9.2011 г.

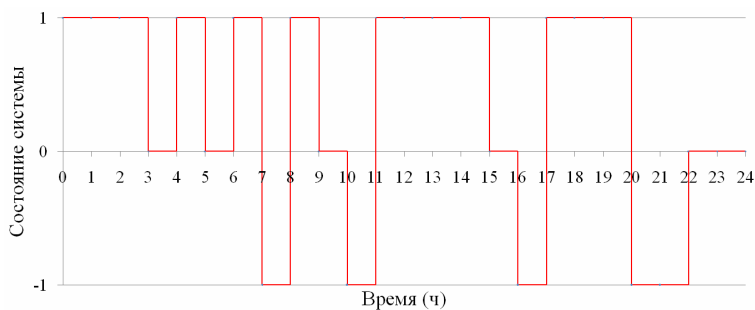


Рис. 11. Диаграмма переключения между источниками энергии 14.9.2011 г.

Качество управления системой оценивается через процент сокращения затрат на электроэнергию до и после применения метода формирования стратегии покупки электроэнергии. Таблица 1 содержит информацию о гибридной энергетической системе.

Таблица 1. Параметры построенной гибридной энергетической системы

Название	Описание
Прогнозируемые затраты (руб.) (14.9.2011 г.)*	822
Реальные затраты (руб.) (14.9.2011 г.)*	830

Название	Описание
Затраты без применения гибридной энергетической системы (руб.) (14.9.2011 г.)**	2756
Время вычисления (сек.) (14.9.2011 г.)	165
Процент сокращения затрат (%)	70
Стоимость системы (млн. руб.)***	2,5

* расчет по формуле (3);

** расчет по формуле (2);

*** расчет по формуле (6).

5. Заключение

В статье рассмотрена модель гибридной энергосистемы с источниками возобновляемой энергии в корпусе «В» ВолГТУ. В результате моделирования гибридной энергосистемы выявлены оптимальная конфигурация гибридной энергосистемы и оптимальная стратегия покупки электроэнергии 14.9.2011 г. Преимущество применения подхода заключается в том, что он позволяет определить оптимальную стратегию переключения между внешней электросетью и источниками возобновляемой энергии в зависимости от текущего тарифного плана текущего состояния системы, прогнозируемого состояния. В результате этого могут быть сокращены затраты на электроэнергию.

Литература

1. КАМАЕВ В.А., ЩЕРБАКОВ М.В., ПАНЧЕНКО Д.П. *Применение коннективистских систем для прогнозирования потребления электроэнергии в торговых центрах // Управление большими системами. – 2010. – №31. – С. 92–109.*
2. МАЙ Н.Т. *Мультиагентный метод управления энергопотоками в гибридной энергосистеме с источниками возобновляемой энергии / Н. Т. Май, В. А. Камаев, М. В. Щербаков, Т. Х. Чинь // Прикаспийский журнал: управление и высокие технологии. – 2013. – Т. 22, №2. – С. 30–41.*

3. ЩЕРБАКОВ М.В. *ICDMS – Программное обеспечение как сервис для решения задач идентификации на основе коннективистских систем* // Известия Волгоградского государственного технического университета. – 2009. – Т. 12, №7. – С. 88–91.
4. ЯНОВСКИЙ Т.А., ЩЕРБАКОВ М.В. *Методика предварительного анализа данных в автоматизированных системных прогнозирования потребления электроэнергии* // Вестник компьютерных и информационных технологий. – 2012. – №3. – С. 21–26.
5. CAISHENG W. *Modeling and control of hybrid wind/photovoltaic/fuel cell distributed generation systems*. – A dissertation submitted in partial fulfillment of the requirement for the degree of Doctor of Philosophy in Engineering, 2006. – 402 p.
6. TRAN T.H.O., TRAN H.L., DONG S.T.C. *Parallel neural network application in forecasting power demand* // Development of science and technology magazine. – 2010. – Vol. 10, №11. – P. 57–73.
7. *PV*SOL® Expert Version 5.5 Design and Simulation of Photovoltaic Systems Manual*. [Электронный ресурс]. – URL: http://www.valentin.de/sites/default/files/downloads/handbueche_r/en/manual-pvsol.pdf (дата обращения: 17.10.2013).
8. *The encyclopedia of alternative energy and sustainable living* [Электронный ресурс]. – URL: http://www.daviddarling.info/encyclopedia/S/AE_standard_test_conditions.html (дата обращения: 17.10.2013).

SIMULATION AND OPTIMAL CONTROL OF INTELLIGENT HYBRID RENEWABLE ENERGY SYSTEM

Mai Ngoc Thang, Volgograd State Technical University,
Volgograd, graduate student (mnthang464@mail.ru).

Ha Van Muon, Volgograd State Technical University, Volgograd,
student (muon.ha@mail.ru).

Kamaev Valerii Anatolevich, Volgograd State Technical
University, Volgograd, CAD Dept. , professor
(kamaev@unix.cad.vstu.ru).

Shcherbakov Maxim Vladimirovich, Volgograd State Technical
University, Volgograd, Cand.Sc., CAD Dept., Senior lecturer
(vstu@gebeus.ru).

Thai Quang Vinh, Institute of Information Technology,
Hanoi, Vietnam (tqvinh@ioit.ac.vn).

Abstract: Nowadays energy management is an important subject for intelligent control. The problem is to find a real-time energy sources' switching strategy, which minimizes wastes and energy cost. We consider an intelligent hybrid energy system (HRES), which optimizes a number of criteria, such as efficiency and energy costs, by switching between energy sources in real-time based on energy consumption and production forecasts. The switching strategy is found with a genetic algorithm. Efficiency of the developed control system as compared to existing energy systems is verified with computer simulation.

Keywords: hybrid, smart grid, renewable energy source, neural network, forecast, optimization, HRES.

*Статья представлена к публикации
членом редакционной коллегии М.В. Губко*

*Поступила в редакцию 19.06.2013.
Опубликована 30.11.2013.*

www.mtas.ru

**ИНТЕРНЕТ-сайт теории управления
организационными системами**

ТЕОРИЯ УПРАВЛЕНИЯ ОРГАНИЗАЦИОННЫМИ СИСТЕМАМИ

ИНСТИТУТ ПРОБЛЕМ УПРАВЛЕНИЯ ИМ. В.А. ТРАПЕЗНИКОВА РАН



Целью сайта является предоставление специалистам по теории и практике управления организационными системами (ученым, преподавателям, аспирантам, студентам, а также реальным управленцам) доступа к ресурсам, отражающим современное состояние теории и возможности обмена идеями и результатами.

На сайте имеются разделы:

Теория – с обзором теории управления организационными системами, глоссарием, информацией для аспирантов;

Практика – с обзором результатов внедрения механизмов управления в реальных организациях;

Библиография – около 2500 публикаций по теории управления, снабжена классификатором и аннотациями;

Электронная библиотека – около 700 полнотекстовых монографий, статей и учебных пособий;

а также многое другое.

На сайте работает **форум**, на котором можно обсудить вопросы, относящиеся к математике, экономике, управлению организациями, узнать новости теории управления и ознакомиться с планируемыми конференциями и семинарами.

ubs.mtas.ru

Интернет-сайт электронного периодического
научного издания «Управление большими
системами: сборник трудов»

УПРАВЛЕНИЕ БОЛЬШИМИ СИСТЕМАМИ

ЭЛЕКТРОННЫЙ СБОРНИК НАУЧНЫХ ТРУДОВ
ИНСТИТУТА ПРОБЛЕМ УПРАВЛЕНИЯ ИМ. В.А. ТРАПЕЗНИКОВА РАН

С 1998 года Институт проблем управления РАН выпускает периодический сборник трудов ученых, занимающихся разработкой и исследованием математических моделей управления большими (социально-экономическими, организационными, организационно-техническими и др.) системами. Все статьи, публикуемые в сборнике, проходят рецензирование ведущими специалистами по теории управления.

С 2006 года сборник "Управление большими системами" вместе с ведущим журналом ИПУ РАН "Проблемы управления" – включены в Российский индекс научного цитирования (РИНЦ).

С июля 2007 года Сборник входит в список ВАК (перечень ведущих рецензируемых научных журналов и изданий, в которых должны быть опубликованы основные научные результаты диссертации на соискание ученой степени доктора и кандидата наук):

* по управлению, вычислительной технике и информатике (для докторов наук);

* по электронике, измерительной технике, радиотехнике и связи; по энергетике (для кандидатов наук).

**Уважаемые коллеги! Приглашаем Вас опубликовать
Вашу статью в очередном выпуске сборника
"Управление большими системами"!**

Периодичность сборника - 4 раза в год. Время выхода прошедшей рецензирование статьи - 3-4 месяца. Плата с авторов за публикацию рукописей не взимается.