

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение
высшего образования «Московский физико-технический институт (госу-
дарственный университет)»

ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ МАТЕМАТИЧЕСКИХ МОДЕЛЕЙ

Сборник научных трудов

Москва
МФТИ
2017

УДК 004
ББК 32.81
И74

Рецензенты:

Кафедра математики, эконометрики и информационных технологий Московского государственного института международных отношений (университет)
Министерства иностранных дел Российской Федерации
Заведующий кафедрой к.ф.-м.н., доцент *Н.В. Артамонов*
д.ф.-м.н., профессор *Л.А. Бекларян*

Редакционная коллегия:

д.ф.-м.н., проф. А.И. Лобанов (отв. редактор),
к.ф.-м.н. О.С. Федько (зам. отв. редактора),
д.ф.-м.н., проф., член-корр. РАН И.Б. Петров, к.ф.-м.н., доц. А.Г. Бирюков,
к.ф.-м.н. Черноусова Е.О., Н.И. Дунаева (отв. секретарь)

И74 Информационное обеспечение математических моделей:

сб. науч. тр. – М. : МФТИ, 2017. – 128 с.

ISBN 978-5-7417-0621-3

Сборник предназначен для широкого круга специалистов, занимающихся проблемами управления и обработки информации, математического и компьютерного моделирования.

Отражены результаты научных исследований, ведущихся в Московском физико-техническом институте (государственном университете) в области вычислительной математики, компьютерных и информационных технологий.

**УДК 004
ББК 32.81**

ISBN 978-5-7417-0621-3

© Федеральное государственное автономное образовательное учреждение высшего образования «Московский физико-технический институт (государственный университет)», 2017

ПРЕДИСЛОВИЕ

В настоящее время роль компьютерных технологий в жизни людей чрезвычайно велика. Как следствие, развитие современной науки неизбежно связано с совершенствованием и эффективным использованием в различных сферах деятельности информационных комплексов.

Данный сборник посвящен проблемам построения, анализа математических моделей сложных объектов и явлений, оптимальному управлению системами, задачам автоматизации, совершенствованию информационных технологий.

Статьи сборника отражают тематику и результаты исследований, проводимых в этих областях в МФТИ и связанных с институтом организациях.

Выделяются следующие основные направления работ: математическое и компьютерное моделирование физических процессов, построение, исследование и анализ экономических моделей, задачи машинного обучения.

Среди предметных областей рассматриваются математическое моделирование управляемых процессов, объектов или явлений, механика деформируемого твердого тела, волновая физика, сейсмология, экспериментальная экономика, анализ данных, распределенные вычислительные системы.

Разнообразие подходов, использованных в представленных статьях, включает аппарат методов оптимизации, дискретной математики, теорию обыкновенных дифференциальных уравнений, вероятностно-статистический подход, численные методы, технологии проектирования вычислительных комплексов.

Статьи сборника могут быть интересны специалистам указанных направлений научных исследований.

УДК 517.9

В. И. Голубев, А. В. Фаворская

Московский физико-технический институт

(государственный университет)

E-mail: w.golubev@mail.ru

АЛГОРИТМЫ МИГРАЦИИ БОРНА И КИРХГОФА ДЛЯ УПРУГИХ СРЕД

Целью данной работы являлось сравнение миграционных изображений, полученных с использованием методов Борна и Кирхгофа. Проведено обобщение метода Борна с акустического на упругий случай в приближении однородной фоновой среды. Разработан математический алгоритм, позволяющий на основе сейсмограмм построить миграционное изображение среды. Его работоспособность была протестирована на простейших моделях с синтетическими данными.

Ключевые слова: сейсмическая разведка, математическое моделирование, миграция, упругая среда.

Метод математического моделирования

В ходе проведения сейсмической разведки месторождений углеводородов производится регистрация полевых сейсмограмм, которые в дальнейшем могут быть использованы для восстановления положений отражающих горизонтов в подповерхностном пространстве (задача миграции). В работе [1] был предложен метод, получивший название метода Борна, основанный на акустическом приближении геологической среды. В более общем случае среда описывается моделью линейно-упругого тела. Её состояние определяется гиперболической системой уравнений в частных производных [2]. В настоящей работе на основе данной модели был разработан математический алгоритм миграции в упругой постановке, основанный на разделении сейсмического поля на первичную и вторичную составляющие, пренебрежении вторичным полем в правой части системы уравнений (метод Борна) и вычислении присоединённого оператора [3]. В виду высокой вычислительной сложности задачи разработаны параллельные алгоритмы с использованием технологии OpenMP, позволяющие проводить расчёт одновременно на нескольких вычислительных ядрах процессора.

Результаты моделирования

В качестве тестовой модели была выбрана однородная среда размерами $10 \times 2,5$ км с характеристиками $C_p = 2500$ м/с, $C_s = 1250$ м/с, содержащая в центре шаровое включение с параметрами (медленностями), отличающимися на 10 %. Плотность считалась равной 2500 кг/м³. Источники располагались на глубине 15 м через каждые 50 м и испускали вертикально вниз импульс Рикера с пиковой частотой 25 Гц. Положение приёмников совпадало с положением источником (zero-offset).

Восстановленные методом Борна и Кирхгофа [4] на сетке 50×50 м миграционные изображения представлены на рис. 1. Их анализ показывает, что истинные границы неоднородности восстанавливаются, за исключением участков, близких к вертикальным. Кроме того, изображения содержат множество артефактов и ложных границ, интенсивность которых в случае Борна ниже, чем в случае Кирхгофа. Исследование выполнено при финансовой поддержке гранта Президента РФ № МК-1831.2017.9.

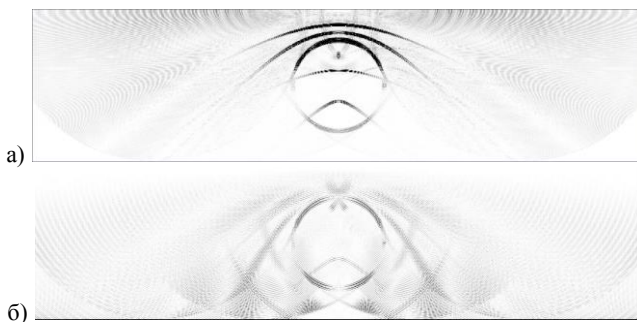


Рис. 1. Миграционные изображения, полученные методом Кирхгофа (а) и Борна (б)

Литература

1. Morse P. M., Feshbach H. Methods of theoretical physics. – New York: McGraw-hill Book, 1953.
2. Седов Л. И. Механика сплошной среды. – М.: Наука, 1970.
3. Zhdanov M. S. Geophysical Inverse Theory and Regularization Problems. – Amsterdam: Elsevier, 2002.
4. Фаворская А. В. Разработка численных методов для моделирования распространения упругих волн в неоднородных средах: автореф. дис. ... канд. физ.-мат. наук. – Долгопрудный: МФТИ, 2015.

Получено 14.04.2017

УДК 517.9

В. И. Голубев, Н. И. Хохлов, Д. П. Григорьевых
Московский физико-технический институт
(государственный университет)
E-mail: k_h@inbox.ru

ЧИСЛЕННЫЙ РАСЧЁТ РАЗРУШЕНИЯ ОБЪЕКТОВ С УЧЁТОМ ИЗМЕНЕНИЯ РЕОЛОГИИ МАТЕРИАЛА

Целью данной работы являлось проведение компьютерного моделирования процесса разрушения преграды под внешним воздействием. На основе анализа напряжённо-деформированного состояния материала определяются места разрушения (появления трещин). В дальнейшем в месте их расположения проводится корректировка тензора напряжения, обеспечивающая учёт изменившейся реологии среды.

Ключевые слова: математическое моделирование, упругая среда, динамическое разрушение.

Метод расчёта

Задача динамического разрушения материала имеет множество приложений: разрушение зданий при землетрясении, гидравлический разрыв пласта в ходе разработки месторождений углеводородов и т.д. С использованием современных высокопроизводительных вычислительных систем возможно проведение высокоточного моделирования процесса разрушения, в том числе с учётом изменения реологии разрушенной части материала.

В общем случае при малых деформациях динамическое поведение среды описывается моделью линейно-упругого тела. Оно определяется гиперболической системой уравнений в частных производных [1]. В настоящей работе на основе данной модели был проведён расчёт интенсивного ударного воздействия, приводящего к разрушению материала. Для описания процесса разрушения использован подход, предложенный в работе [2]. Трещина в материале появляется при превышении одним из главных напряжений максимального напряжения ($\sigma_{\max}$, характеристика материала), и её плоскость перпендикулярна ему. В дальнейшем для учёта изменившейся реологии материала в каждом узле, содержащем трещину, обнуляются нормальные и тангенциальные (по отношению к плоскости

трещины) компоненты тензора напряжений. Определяющая система уравнений решается численного сеточно-характеристическим методом [3] на гексаэдральной сетке.

Результаты моделирования

В качестве тестовой задачи рассчитывалось динамическое взаимодействие сферического тела (ударника) с параллелепипедной преградой в двумерной постановке. Геометрические размеры, упругие и прочностные характеристики задавались в *безразмерных условных единицах*. Диаметр ударника составлял 5, толщина преграды была равна 50, $C_p = 1$, $C_s = 0.7$, $\rho = 1$, $\sigma_{\max} = 0.003$. В начальный момент времени ударник двигался вертикально вниз со скоростью 5.

Геометрия расчётной области с нанесёнными местами растрескивания после отражения ударника представлена на рис. 1. Анализируя полученное пространственное распределение мест разрушений можно выделить: поверхностные трещины, которые соответствуют экспериментально подтверждённым кольцевым трещинам в трёхмерной постановке эксперимента - зоны объёмного разрушения материала непосредственно под контактной площадкой.

Исследование выполнено при финансовой поддержке гранта Президента РФ № МК-2888.2017.9.

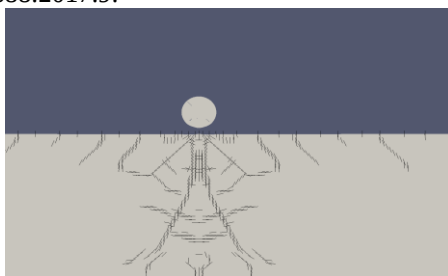


Рис. 1. Трещины в преграде, образовавшиеся из-за воздействия ударника

Литература

1. Седов Л. И. Механика сплошной среды. – М.: Наука, 1970.
2. Майчен Дж., Сак С. Метод расчета «Тензор» // Вычислительные методы в гидродинамике. – М.: Мир, 1967. С. 185–211.
3. Голубев В.И., Петров И.Б., Хохлов Н.И. Численное моделирование сейсмической активности сеточно-характеристическим методом // Журнал вычислительной математики и математической физики. – 2013.– Т. 53. – № 10. – С. 709–1720.

Получено 03.04.2017

УДК 519.633

Д. П. Григорьевых, Н. И. Хохлов

Московский физико-технический институт

(государственный университет)

E-mail: danilgrig@gmail.com

РАЗРУШЕНИЕ ЛЕДЯНОГО ПОЛЯ С ПОМОЩЬЮ ВЫСОКОЧАСТОТНОЙ ВИБРАЦИИ

Исследована возможность разрушения льда с помощью периодических воздействий на верхней поверхности, образующих резонанс. Построена картина разрушений и зависимость максимального напряжения во льду от частоты воздействия (АЧХ). Проведено сравнение продольных и поперечных волн.

Исследование выполнено при финансовой поддержке РФФИ в рамках научного проекта №15-37-20673.

Ключевые слова: сеточно-характеристический метод, разрушение льда, резонанс.

Описание алгоритма

У классического способа разрушения льда взрывами есть большой недостаток — вред окружающей среде, в частности гибель рыбы. При резонансном методе прикладываемые усилия на несколько порядков меньше и не угрожают окружающей среде. На поверхность льда устанавливается массивная платформа, которая может колебаться либо по вертикали, либо по горизонтали (для горизонтальных колебаний требуется закрепление платформы за лед). При таких колебаниях во льду возникают продольные или поперечные волны, которые отражаются от подводной границы. Если период колебаний подобрать равным времени прохождения волны до подводной границы и обратно, то наступит резонанс и напряжения будут увеличиваться с каждым новым циклом, пока лед не разрушится.

Коэффициент отражения равен

$$\alpha = \frac{c_{\text{л}}\rho_{\text{л}} - c_{\text{в}}\rho_{\text{в}}}{c_{\text{л}}\rho_{\text{л}} + c_{\text{в}}\rho_{\text{в}}}.$$

Для продольной волны коэффициент отражения получается около 0.5, тогда как для поперечной он составляет ровно 1 из-за отсутствия поперечной скорости звука в воде. Таким образом, для данной задачи лучше использовать только горизонтальные колебания, и все дальнейшие расчеты приведены для них.

Примеры расчётов

Расчёты проводились для льда толщиной $h = 1 \text{ м}$, $c_1 = 3000 \text{ м/с}$, $c_0 = 2000 \text{ м/с}$, $c_2 = 900 \text{ кг/м}^3$, $\sigma = 1 \text{ МПа}$, амплитуда прикладываемой поверхностной силы $P = 200 \text{ кПа}$. Резонансная частота при таких параметрах получается 1 кГц. На рис. 1 приведён пример разрушения. Видим, что лёд как бы расслоился пополам трещинами, но при этом трещины направлены строго под углом 45 градусов, поэтому в реальности лёд бы разломился на отдельные льдины. Отметим, что амплитуда прикладываемой силы была в 5 раз меньше прочности льда, но за счет резонанса удалось добиться разрушения.

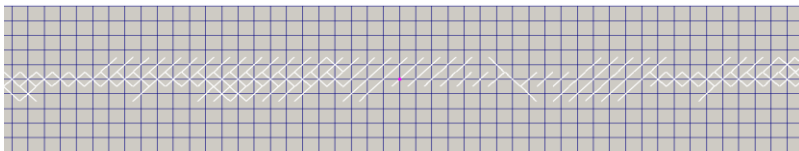


Рис. 1. Разрушения льда резонансными поперечными волнами

Интерес представляет ширина резонансной АЧХ, поскольку в реальности лёд имеет разную толщину и/или параметры и идеальный резонанс не получается. На рис. 2 приведена зависимость резонансного усиления от частоты. Таким образом, при отклонении толщины льда от рассчитанной на 5 см. усиление уменьшается до 7-ми, что все равно довольно много и достаточно для разрушения. Стоит отметить, что максимальное усиление 11 достигается при частоте 1010 Гц, что немного больше теоретически рассчитанной, это объясняется неточностью численной схемы.

Поскольку при правильно подобранной частоте разрушение данным способом происходит за миллисекунды, то, даже заранее не зная точной частоты, можно плавно ее увеличивать и поймать резонанс.

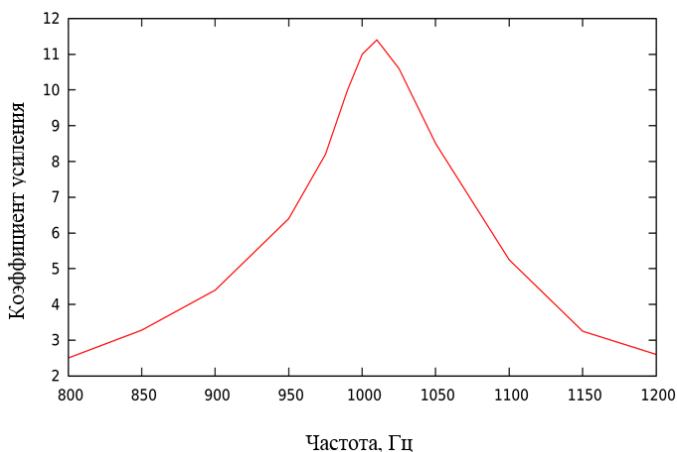


Рис. 2. Зависимость коэффициента усиления приложенного напряжения от частоты

Выводы

Проведен расчет резонансного воздействия на массивное ледяное поле. Получено подтверждение, что данный метод можно использовать для экологически чистого разрушения льда сравнительно малыми нагрузками. При небольших отклонениях параметров льда от идеальных резонанс получается по-прежнему достаточным.

Дальнейшая работа включает следующие этапы:

1. Проведение расчетов в 3D.
2. Поиск схемы с меньшей численной вязкостью.

Литература

1. Гольдштейн Р.В., Осипенко Н.М. Вопросы механики разрушения льда и ледяного покрова при анализе ледовых нагрузок // Вести газовой науки. – 2013. №3 (14). – С. 104–112.
2. Петров И.Б., Холодов А.С. Численное исследование некоторых динамических задач механики деформируемого тела сеточно-характеристическим методом // ЖВМ и МФ. – 1984. – Т. 24, № 5. – С. 722–739.

Получено 18.04.2017

А. М. Иванов, Н. И. Хохлов

Московский физико-технический институт

(государственный университет)

E-mail: ip-e@mail.ru

ЗАДАНИЕ ГРАНИЧНЫХ УСЛОВИЙ PML ДЛЯ МОДЕЛИРОВАНИЯ ВОЛН В УПРУГОЙ СРЕДЕ В СЛУЧАЕ ОСЕВОЙ СИММЕТРИИ

В работе рассматривается применение поглощающих граничных условий PML при моделировании распространения волн в линейно-упругих средах методом FDTD. Описывается операторный подход, позволяющий упростить реализацию граничных условий в случае осевой симметрии среды.

Ключевые слова: FDTD, осесимметричный, PML.

Математическая модель и численный метод

Используется модель линейно-упругой среды [1]. Уравнение, описывающее распространение волн в такой среде, записано в формулировке смещений и упругих напряжений:

$$\begin{cases} \rho \ddot{\mathbf{u}} = \nabla \cdot \mathbf{T}, \\ \mathbf{T} = \lambda(\nabla \cdot \mathbf{u})\mathbf{I} + \mu(\nabla \otimes \mathbf{u} + \mathbf{u} \otimes \nabla), \end{cases}$$

где $\mathbf{u}$ – вектор смещения в каждой точке среды, ρ – плотность среды, λ и μ – параметры Ламе материала, $\mathbf{T}$ – тензор упругих напряжений, $\mathbf{I}$ – единичный тензор. В осесимметричном случае данное уравнение имеет следующий вид в цилиндрических координатах:

$$\rho u_{tt}^r = \frac{1}{r}(r\sigma^{rr})_r + \sigma_z^{rz} - \frac{1}{r}(r\sigma^{\theta\theta})_r + \sigma_r^{\theta\theta},$$

$$\rho u_{tt}^z = \frac{1}{r}(r\sigma^{rz})_r + \sigma_z^{zz},$$

$$\sigma^{rr} = \lambda \left(\frac{1}{r}(ru^r)_r + u_z^z \right) + 2\mu u_r^r,$$

$$\sigma^{rz} = \mu(u_z^r + u_r^z),$$

$$\sigma^{\theta\theta} = \lambda \left(\frac{1}{r} (ru^r)_r + u_z^z \right) + 2\mu \frac{1}{r} u^r,$$

$$\sigma^{zz} = \lambda \left(\frac{1}{r} (ru^r)_r + u_z^z \right) + 2\mu u_z^z.$$

Здесь u^r и u^z – радиальная и вертикальная компоненты вектора смещения, σ^{rr} , σ^{rz} , $\sigma^{\theta\theta}$, σ^{zz} – компоненты тензора упругих напряжений. Нижние индексы обозначают дифференцирование.

Введением симметричных конечно-разностных аппроксимаций производных был реализован конечно-разностный метод на сдвинутой сетке со вторым порядком точности по времени и 8 по пространству.

Задание поглощающих граничных условий PML

В данной работе рассматривается сверточный PML (CPML) [2]. Данный тип PML появился позже первоначально введенного split-field PML (SPML) и является эффективнее по памяти и проще в реализации. Более подробно классификация видов PML представлена в [3].

Граница задается в виде боковых стенок цилиндра на некотором расстоянии от оси. Для корректного задания границы необходимо сделать замену координаты r таким образом, чтобы обеспечить затухание волны при прохождении через PML. Каждое умножение, деление или нахождение производной по r представляется в виде оператора, действующего на некоторое значение u . Тогда замены представимы в виде

$$\frac{\partial u}{\partial r} \rightarrow \frac{\partial u}{\partial \xi} = \frac{\partial u}{\partial r} + \phi,$$

$$\frac{u}{r} \rightarrow \frac{u}{\xi} = \frac{u}{r} + \hat{\phi},$$

$$ru \rightarrow \xi u = ru + \hat{\psi},$$

где значения ϕ , $\hat{\phi}$, $\hat{\psi}$ вычисляются по следующим формулам:

$$\phi^n = e^{-d\Delta t} \phi^{n-1} + (e^{-d\Delta t} - 1) [u_r]^{n-1},$$

$$\hat{\phi}^n = e^{-\hat{d}\Delta t} \hat{\phi}^{n-1} + (e^{-\hat{d}\Delta t} - 1) [u/r]^{n-1},$$

$$\hat{\psi}^n = \hat{\psi}^{n-1} + \hat{d}\Delta t [ru]^{n-1}, \quad \hat{d} = \frac{1}{r} \left(\int_0^r d(\tilde{r}) d\tilde{r} \right).$$

В начале расчета $\phi^0 = 0$, $\hat{\phi}^0 = 0$, $\hat{\psi}^0 = 0$. Здесь Δt – величина шага по времени, верхние индексы обозначают номер шага. Параметр d задает величину поглощения PML.

Следуя указанным выше заменам, для задания PML для членов $(ru^r)_r / r$ потребуется корректировка трех значений: в результате умножения, после дифференцирования и, наконец, после деления на r .

Результаты

Был продемонстрирован способ эффективного задания поглощающих граничных условий типа PML. Хранение дополнительных переменных в памяти ϕ , $\hat{\phi}$, $\hat{\psi}$ требуется только на границах расчетной области.

Проведены тестовые расчеты, в ходе которых получены волновые картины (рис. 1) в разные моменты времени. Варьируя параметр d , удалось уменьшить амплитуду волны на два порядка при прохождении через PML толщиной в 10 узлов.

Исследование выполнено при финансовой поддержке РФФИ в рамках научного проекта № 15-07-01931 А.

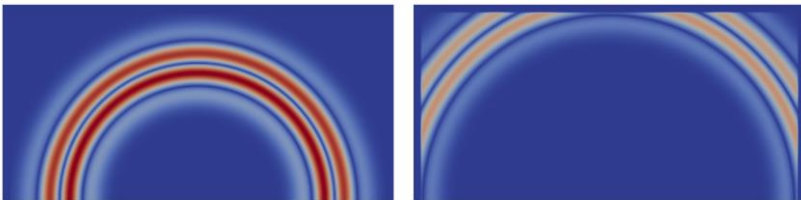


Рис. 1. Волновая картина до и после достижения границы, ось симметрии внизу

Литература

1. *LeVeque R. J.* Finite volume methods for hyperbolic problems. – Cambridge university press, 2002. – V. 31.
2. *Roden J. A., Gedney S. D.* Convolution PML (CPML): An efficient FDTD implementation of the CFS-PML for arbitrary media // Microwave and Optical Technology Letters. – 2000. – V. 27, N 5. – P. 334–339.
3. *Kristek J., Moczo P., Galis M.* A brief summary of some PML formulations and discretizations for the velocity-stress equation of seismic motion // Studia Geophysica et Geodaetica. – 2009. – V. 53, N 4. – P. 459–474.

Получено 30.04.2017

А. А. Скубачевский, Н. И. Хохлов

Московский физико-технический институт

(государственный университет)

E-mail: antonious007@yandex.ru

ЧИСЛЕННЫЙ ЭКСПЕРИМЕНТ ПО РАЗРУШЕНИЮ БЕТОНА ПОСРЕДСТВОМ ВОЗДЕЙСТВИЯ ЭЛЕКТРОМАГНИТНОГО ПОЛЯ**1. Математическая модель**

Для получения результатов, приведенных ниже, используется метод FDTD (Finite-Difference Time-Domain), который применяется для численного решения уравнений Максвелла. Для изотропных, недиспергирующих материалов воспользуемся следующей системой уравнений:

$$\frac{\partial \vec{H}}{\partial t} = -\frac{1}{\mu} \vec{\nabla} \times \vec{E} - \frac{1}{\mu} \sigma^* \vec{H}, \quad (1)$$

$$\frac{\partial \vec{E}}{\partial t} = -\frac{1}{\varepsilon} \vec{\nabla} \times \vec{H} - \frac{1}{\varepsilon} \sigma \vec{E}, \quad (2)$$

где $\vec{E}$ – напряженность электрического поля, $\vec{H}$ – напряженность магнитного поля, ε – диэлектрическая проницаемость, μ – магнитная проницаемость, σ – электрическая проводимость, σ^* – аналогичная магнитная проводимость.

В данном методе пространственные и временные производные физических величин заменяются их конечно-разностными аналогами:

$$\frac{\partial u}{\partial x}(i\Delta x, j\Delta y, k\Delta z, n\Delta t) = \frac{u_{i+1/2,j,k}^n - u_{i-1/2,j,k}^n}{\delta x} + O[(\Delta x)^2], \quad (3)$$

$$\frac{\partial u}{\partial t}(i\Delta x, j\Delta y, k\Delta z, n\Delta t) = \frac{u_{i,j,k}^{n+1/2} - u_{i,j,k}^{n-1/2}}{\delta t} + O[(\Delta t)^2], \quad (4)$$

где $\Delta x, \Delta y, \Delta z, \Delta t$ – величины шага по каждой координате и времени соответственно; u – одна из $E_x, E_y, E_z, H_x, H_y, H_z$. Для производных по y и z конечно-разностные выражения записываются так же.

Поля в правой части уравнений (1, 2) будут записаны для времени $n\Delta t$, а в левой – $(n - 1/2)\Delta t$ и $(n + 1/2)\Delta t$. Для их согласования представим конечно-разностные выражения полей E и H в точке n в следующем виде:

$$E_x|_{i,j,k}^n = \frac{E_x|_{i,j,k}^{n+1/2} + E_x|_{i,j,k}^{n-1/2}}{2}. \quad (5)$$

Конечно-разностные выражения для компонент E_y, E_z, H_x, H_y, H_z записываются аналогично.

Конечно-разностный вариант уравнений (1) и (2) в координатном представлении, используя (3), (4), (5), для E_x – компоненты уравнения (1), можно записать в следующем виде.

Представив (1) и (2) в конечно-разностном виде, используя (3), (4) и (5), имеем для компонент полей, к примеру:

$$E_x|_{i,j+\frac{1}{2},k+\frac{1}{2}}^{n+\frac{1}{2}} = C_a|_{i,j+\frac{1}{2},k+\frac{1}{2}} \times E_x|_{i,j+\frac{1}{2},k+\frac{1}{2}}^{n-\frac{1}{2}} + C_b|_{i,j+\frac{1}{2},k+\frac{1}{2}} \times \\ \times \left(H_z|_{i,j+1,k+\frac{1}{2}}^n - H_z|_{i,j,k+\frac{1}{2}}^n - H_y|_{i,j+\frac{1}{2},k+1}^n + H_y|_{i,j+\frac{1}{2},k}^n \right), \quad (6)$$

где (размер ячейки сетки Δ):

$$C_a|_{i,j,k} = (1 - \frac{\sigma_{i,j,k}\Delta t}{2\varepsilon_{i,j,k}})/(1 + \frac{\sigma_{i,j,k}\Delta t}{2\varepsilon_{i,j,k}}), \quad (7)$$

$$C_b|_{i,j,k} = (\frac{\Delta t}{\varepsilon_{i,j,k}\Delta})/(1 + \frac{\sigma_{i,j,k}\Delta t}{2\varepsilon_{i,j,k}}). \quad (8)$$

Уравнения для остальных компонент выводятся аналогично [1].

В качестве граничных условий в этой модели используются неотражающие условия PML (Perfectly Matched Layer), основой которых является разделение каждой компоненты поля на две взаимно ортогональных, а также задание в граничном слое заданной проводимости [1].

2. Результаты

Рассмотрим следующий численный эксперимент: на сетке 200×400 ячеек, общим размером 5×10 см, электромагнитная волна падает перпендикулярно на плоский слой с заданными диэлектрической и магнитной проницаемостями и проводимостью. В данном эксперименте в качестве материала используется бетон. Электрическое поле в волне задается в

виде импульса Гаусса: $E = e^{-\frac{(t-\Delta t)^2}{\sigma^2}}$. Поле в остальных ячейках сетки рассчитывается с помощью метода FDTD, с использованием TFSF-условий. Используются следующие характеристики бетона: $\varepsilon = 4.5\varepsilon_0$, $\mu = \mu_0$, где ε_0 и μ_0 – соответственно диэлектрическая и магнитная проницаемости вакуума. Диэлектрическая проницаемость и проводимость в численных экспериментах меняется соответственно: от $2\varepsilon_0$ до $8\varepsilon_0$, от 0.1 до 1 (1/(Ом * м)).

На рис. 1 показана зависимость глубины проникновения волны в бетон (м) от проводимости (1/(Ом * м)).

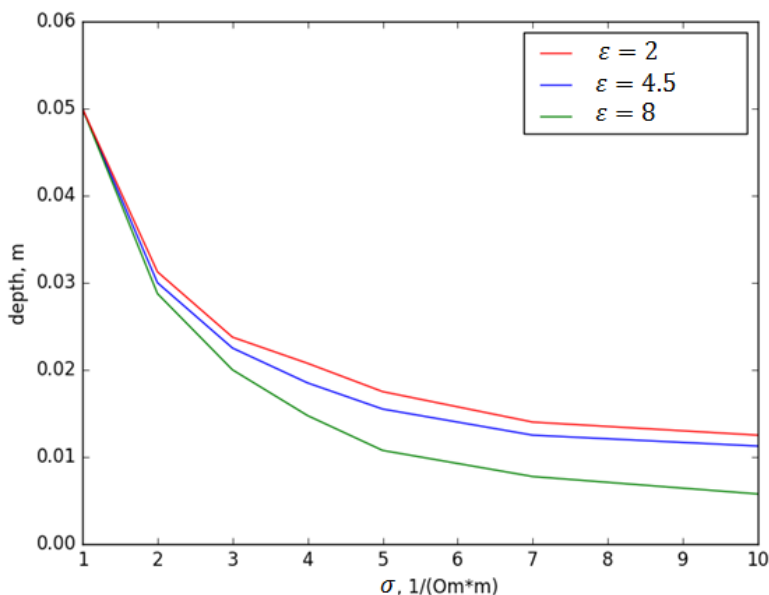


Рис. 1. Зависимость глубины проникновения волны в бетон от проводимости

На рис. 2 показана зависимость плотности тока в бетоне (А/м²) от расстояния до поверхности (м) при проводимости 0.1, 0.5 и 1 (1/(Ом * м)).

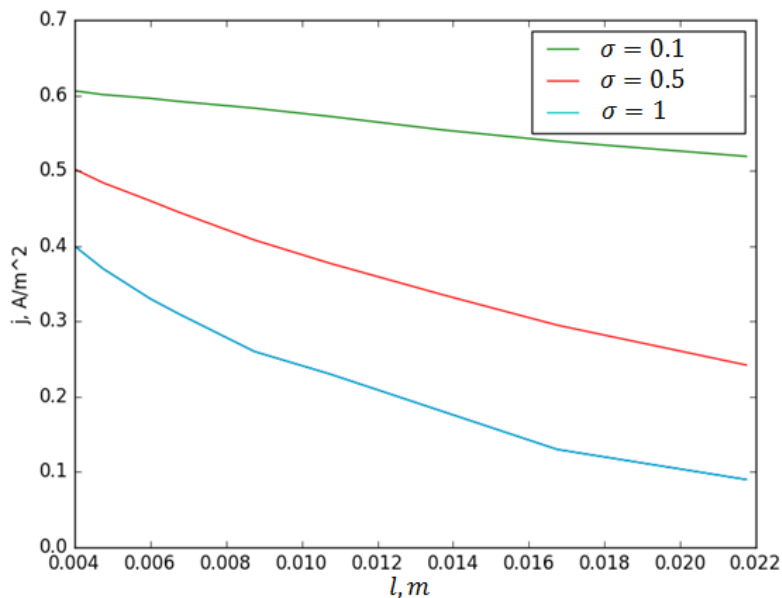


Рис. 2. Зависимость плотности тока в бетоне от расстояния до поверхности

Воздействуем теперь на обводненный бетон, имеющий форму куба размерами $5 \times 5 \times 5$ см, электромагнитным полем. Мощность источника 3 Вт. Амплитуда электрического поля, индуцируемого источником, равна 1000 В/м. Проводимость бетона примем равной 5 Ом/м. Найдем тепло, выделившееся в бетоне за час:

$$Q = J^2 R t,$$

$$R = \frac{l}{\sigma S},$$

$$j = \sigma E,$$

где E – величина электрического поля, σ – проводимость, R – сопротивление, j – плотность тока, S – площадь сечения проводника, l – длина проводника.

Шаг сетки h равен $2.5 \cdot 10^{-4}$ м. Шаг по времени с учетом условия устойчивости: $\tau = \frac{h}{c\sqrt{2}}$.

Пропустив так через бетон один импульс, посчитав возникший ток суммированием по сетке, а потом-выделившееся тепло по вышеуказанной формуле, получим, что за $5.9 \cdot 10^{-11}$ с выделилось $9 \cdot 10^{-11}$ Дж. Таким образом, если пропустить последовательность импульсов, то за 1 с выделится 1.54 Дж, а за 1 ч $5.55 \cdot 10^3$ Дж.

Считая плотность бетона $\rho = 2500 \text{ кг/м}^3$, теплоемкость бетона $c_6 = 1000 \text{ Дж/(кг} \cdot \text{К)}$, а объем керна $V = 1.25 \cdot 10^{-4} \text{ м}^3$, имеем

$$\Delta T = \frac{Q}{\sigma \rho V c_6} \approx 17 \text{ К.}$$

Покажем, что в данном случае излучением можно пренебречь.

$Q_{\text{изл}} = I \cdot S = \sigma^* T^4 S = 5.67 \cdot 10^{-8} \cdot 17^4 \cdot 3 \cdot 10^{-3} \text{ Дж}$, что много меньше суммарного тепла.

I – плотность излучения, $S = 3 \cdot 10^{-3} \text{ м}^2$ – площадь поверхности, $\sigma^* = 5.67 \cdot 10^{-8} \frac{\text{Дж}}{\text{м}^2 \text{К}^2}$ – постоянная Стефана – Больцмана.

Рассчитываем теперь, на сколько нагрелась вода в керне. Находим, что вода составляет одну десятую от массы бетона. Считаем, что все тепло сосредоточено в воде. Объем и плотность бетона указаны выше. Итак, температура воды изменилась на:

$$\Delta T = \frac{Q}{0.1 M c_w} = 40 \text{ К,}$$

где удельная теплоемкость воды $c_w = 4200 \text{ Дж/(кг} \cdot \text{К)}$.

Посчитаем изменение давления:

$$\Delta p = \frac{\alpha_T}{\alpha_p} \Delta T = 4 \cdot 10^6 \text{ Па,}$$

где $\alpha_T = 10^{-4} \text{ К}^{-1}$; $\alpha_p = 10^{-9} \text{ Па}^{-1}$.

Учитывая, что прочность бетона на разрыв составляет 400 кПа, заключаем, что бетон разорвет при таком нагреве.

Литература

1. Taflove A., Hagness S. Computational electrodynamics: the finite-difference time-domain method. 3d edition. Artech House. London, 2005. P. 51–80.

Получено 20.04.2017

П. В. Стогний

Московский физико-технический институт

(государственный университет)

E-mail: stognii@phystech.edu

МОДЕЛИРОВАНИЕ ВОЛНОВЫХ ПРОЦЕССОВ В ПОДЛЕДНИКОВОМ ОЗЕРЕ ВОСТОК

Целью данной работы является исследование волновых процессов в подледниковом озере Восток с использованием сеточно-характеристического метода. Исследованы отклики от различных слоев, которые окружают область нахождения данного озера.

Ключевые слова: волновые процессы, сеточно-характеристический метод.

Введение

Подледниковое озеро Восток находится в Восточной Антарктиде и является полностью изолированным от подледниковых водоемов, находящихся вблизи озера [1]. Данный водный объект отделен от атмосферы ледниковым покровом, мощность которого колеблется от 1950 до 4350 м. Данное озеро представляет крайний интерес для исследований в области сейсморазведки.

Математическая модель

Озеро Восток имеет большие размеры. Объем воды в нем составляет примерно 6100 км³, а абсолютные высоты озера изменяются от –600 м в северной части до –150 м в южной части [2]. В данной работе представлены результаты моделирования небольшого участка озера в двумерном случае с помощью сеточно-характеристического метода [3].

Длина модели по оси x бралась равной 4700 м. Высота ледяного покрова составляла 3900 м, водной части озера – 800 м, слоя грунта – 1000 м. Шаг по оси x и по оси y равнялся 1 м. Скорость продольных волн в ледовом покрове равнялась 3940 м/с, скорость поперечных волн – 2493 м/с. Скорость продольных волн в водном слое составляла 1500 м/с. Скорость продольных волн в слое грунта равнялась 3000 м/с, скорость поперечных волн – 1875 м/с. Шаг по времени брался равным 0.0001 с.

Всего было сделано 50 000 шагов по времени. Источник и приемники сигнала располагались на поверхности ледового покрова.

Схематичное изображение модели можно увидеть на рис. 1.



Рис. 1. Схематичное изображение модели

Волновые картины

Волновые картины, полученные в результате расчета данной модели, представлены на рис. 2. Хорошо различимы отклики от каждого составляющего слоя рассчитываемой модели. Заметны дополнительные вклады в результате отражений от краев расчетной области. В дальнейших исследованиях следует увеличить ширину модели для устранения данных эффектов.

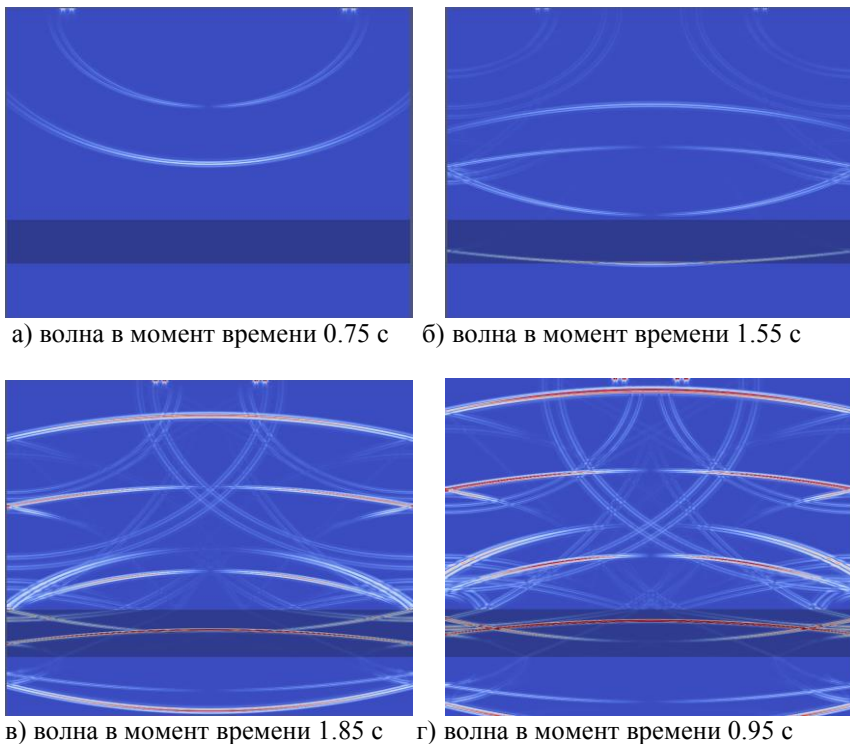


Рис. 2. Волновые картины в различные моменты времени

Сейсмограмма

Сейсмограмма модели озера Восток представлена на рис. 3. Из рисунка хорошо видны отклики от каждого из слоев: лед, вода, грунт. Есть дополнительные отклики, связанные с большой толщиной льда – 3900 м, что требует дополнительных исследований.

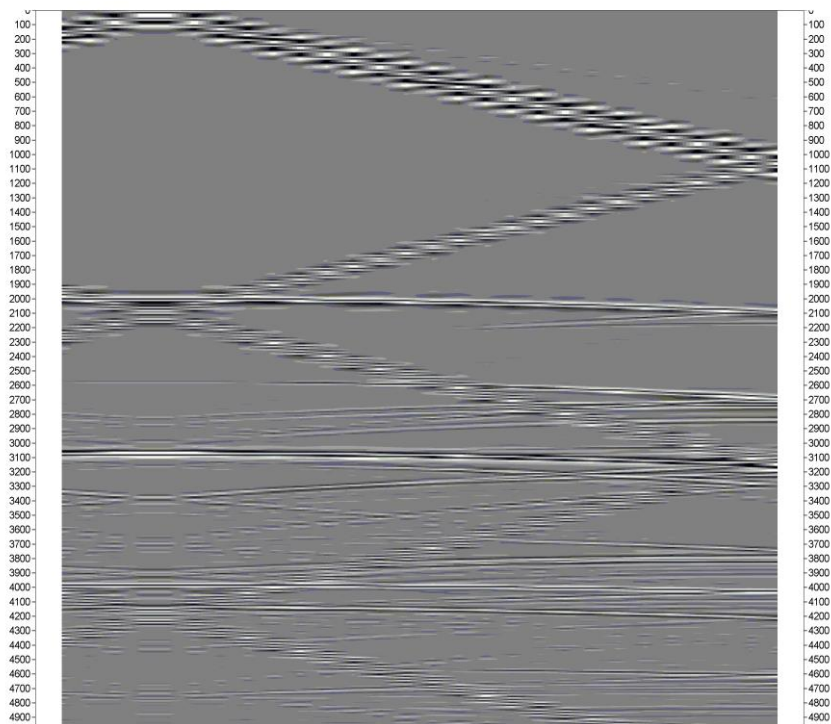


Рис. 3. Сейсмограмма в результате расчета 50 000 шагов модели озера Восток

Выводы

Было проведено моделирование озера Восток в двумерном случае в условиях, приближенных к реальным. Были получены четкие картины отражений от различных геологических слоев, непосредственно прилегающих к озеру. Из сейсмограммы модели (рис. 3) следует, что данная модель нуждается в увеличении для исследования большого количества сейсмических откликов от ледового покрова и водного слоя.

Исследование выполнено при финансовой поддержке РФФИ в рамках научного проекта № 16-07-00233 А.

Литература

4. *Масолов В.Н., Попов С.В., Лукин В.В., Попков А.М.* Рельеф дна и водное тело подледникового озера Восток, Восточная Антарктида // ДАН. – 2010. – Т. 433, № 5. – С. 693–698.
5. *Попов С.В., Масолов В.Н., Лукин В.В.* Озеро Восток, Восточная Антарктида: мощность ледника, глубина озера, подледный и коренной рельеф // Лед и снег. – 2011. – Т. 1, № 113. – С. 25–35.
6. *Фаворская А.В., Петров И.Б., Петров Д.И., Хохлов Н.И.* Численное моделирование волновых процессов в слоистых средах в условиях Арктики // Математическое моделирование. – 2015. – Т. 27, № 11. – С. 63–75.

Получено 30.04.2017

УДК 519.83

Д. М. Старков¹, И. С. Меньшиков^{1,2}

¹Московский физико-технический институт
(государственный университет)

²Вычислительный центр им. А.А. Дородницына ФИЦ ИУ РАН
E-mail: denstar08@gmail.com

РАВНОВЕСИЯ В ИНФОРМАЦИОННЫХ КАСКАДАХ

Целью данной работы является исследование равновесий в игре типа «информационный каскад» и сравнение их с результатами лабораторного эксперимента. Информационные каскады моделируют «стадное» поведение в социальных сетях и на финансовых рынках (см. [1, 2]).

Ключевые слова: информационные каскады, равновесие Нэша, равновесие дискретного отклика, QRE, лабораторные эксперименты.

Математическая модель

Определим множество игроков $T = \{1, 2, \dots, T\}$, которые поочередно делают выбор из двух альтернатив A и B . Для любого $t \in T$ обозначим $c_t \in \{A, B\}$ выбор игрока t . Одна из альтернатив выбирается природой как правильная, обозначим этот выбор $\omega \in \{A, B\}$. Игрок получает выигрыш 1, если выбирает верную альтернативу ($c_t = \omega$) и не получает ничего в противном случае. Разумеется, игрокам неизвестна правильная альтернатива ω , и изначально они считают оба состояния равновероятными. Кроме того, каждый игрок получает личный сигнал s_t о правильном состоянии. Сигнал формируется следующим образом:

$$\omega = A \Rightarrow \begin{cases} s_t = a, c \text{ вероятностью } q > \frac{1}{2}, \\ s_t = b, c \text{ вероятностью } 1 - q, \end{cases}$$

$$\omega = B \Rightarrow \begin{cases} s_t = b, c \text{ вероятностью } q > \frac{1}{2}, \\ s_t = a, c \text{ вероятностью } 1 - q. \end{cases}$$

Игроку t неизвестны сигналы, полученные другими игроками, однако известны выборы игроков, сделавших ход до него $H_t = \{c_1, c_2, \dots, c_{t-1}\}$ ($H_1 = \emptyset$). Обозначим через $p_t \equiv P(\omega = A | H_t)$ мнение игрока t о правильном состоянии на основании истории H_t . Ясно, что $p_1 = \frac{1}{2}$. По правилу Байеса, получив сигнал $s_t = a$, игрок t считает состояние A правильным с вероятностью:

$$\pi_t^a(p_t) \equiv P(\omega = A | H_t, s_t = a) = \frac{qp_t}{qp_t + (1-q)(1-p_t)}.$$

Аналогично, вероятность того, что, получив сигнал $s_t = b$, игрок t считает состояние A правильным, есть

$$\pi_t^b(p_t) \equiv P(\omega = A | H_t, s_t = b) = \frac{(1-q)p_t}{(1-q)p_t + q(1-p_t)}.$$

Заметим, что $\forall p_t : 0 < p_t < 1 \mapsto \pi_t^a(p_t) > p_t > \pi_t^b(p_t)$, т.е. игрок t , получив сигнал a , верит в то, что $\omega = A$ с большей вероятностью, чем при получении противоположного сигнала.

Назовем сигнал q **корректным**, если он указывает на состояние ω , выбранное природой, в противном случае назовем сигнал **некорректным**.

Каскадом будем называть любую последовательность из не менее трех одинаковых ходов подряд. **Длиной каскада** будем считать число повторов минус два. Первые два повтора отнесем к стадии *формирования* каскада.

Так, в последовательности $\{A, B, A, A, A, A, A\}$ каскад длиной 3 *сформировался* на шаге $t = 5$ при $T = 7$ игроках.

В последовательности $\{A, A, A, B\}$ каскад длиной 1 формируется на шаге $t = 3$ и *разрушается* на шаге $t = 4$.

Назовем каскад **правильным**, если он выявляет выбор природы ω , в противном случае назовем каскад **неправильным**.

Заметим, что после разрушения каскада может возникнуть новый каскад. Если новый каскад повторяет состояние первого каскада, назовем его

повторным, в противном случае – *обратным*. Отдельно выделим каскады, переходящие из неправильного состояния в правильное, и назовем их *самокорректирующимися*.

Равновесие Нэша

Рассмотрим оптимальное поведение с учетом рациональности всех игроков. Без ограничения общности будем считать, что A – это правильное состояние.

Игрок 1 действует в соответствии с сигналом $c_1 = \begin{cases} A, & \text{если } s_1 = a, \\ B, & \text{если } s_1 = b. \end{cases}$

Игрок 2 повторяет выбор первого, если получает сигнал $s_2 = c_1$, и следует своему сигналу ($c_2 = s_2$) с вероятностью $\beta \geq \frac{1}{2}$ в противном случае.

Для третьего агента есть 2 возможные ситуации:

(i) первые два выбора совпали, в этом случае игрок 3 делает такой же выбор, как предыдущие игроки, независимо от своего сигнала (начинается каскад);

(ii) выборы предыдущих игроков не совпали, тогда третий игрок оказывается в положении первого, следуя своему сигналу ($c_3 = s_3$).

В случае (i) четвертый игрок оказывается в ситуации третьего и делает такой же выбор, как все предыдущие игроки, не обращая внимания на свой сигнал, в случае (ii) – в позиции второго игрока, и т.д.

Таким образом, в равновесии Нэша после некоторого чередования ходов может сформироваться только один каскад.

Равновесие дискретного отклика

Теперь рассмотрим равновесие квантильного отклика (QRE [2-4]) для модели, описанной выше. В QRE к выигрышу каждого игрока t добавляются возмущения ε_t^A и ε_t^B для выборов A и B соответственно. Обозначим $\theta_t = (s_t, \varepsilon_t)$, где $\varepsilon_t \equiv \varepsilon_t^A - \varepsilon_t^B$ – тип игрока. Предполагается, что ε_t – независимые одинаково распределенные величины, имеющие логистическое распределение с параметром λ . Возмущение ε_t может быть интерпретировано как стохастическая часть принятия решения из-за ограниченной рациональности либо как индивидуальные предпочтения игрока, возникающие по каким-либо другим причинам.

Охарактеризуем оптимальное решение агента t с учетом его типа θ_t и истории H_t (которая определяет общее мнение p_t). Ожидаемый выигрыш при выборе A : $u(A) = \pi_t^{s_t}(p_t) + \varepsilon_t$, при выборе B : $u(B) = 1 - \pi_t^{s_t}(p_t)$.

С учетом логистического распределения ε_t получаем вероятность выбора A , равную

$$\begin{aligned} P(c_t = A | H_t, s_t) &= P(u(A) > u(B)) = \\ &= P(\varepsilon_t > 1 - 2\pi_t^{s_t}(p_t)) = 1 - P(\varepsilon_t \leq 1 - 2\pi_t^{s_t}(p_t)) = \\ &= 1 - \frac{1}{1 + \exp(-\lambda(1 - 2\pi_t^{s_t}(p_t)))} = \frac{1}{1 + \exp(\lambda(1 - 2\pi_t^{s_t}(p_t)))}. \end{aligned}$$

Выбор B будет сделан с дополнительной вероятностью.

При $\lambda \rightarrow \infty$ выборы полностью рациональны, и ε_t перестает оказывать влияние на них. В итоге получаем равновесие Нэша с не более чем одним каскадом, описанное в выше. При $\lambda \rightarrow 0$ выборы A и B равновероятны, независимо от сигнала.

Описание эксперимента

С целью проверки качества вышеописанных моделей в Лаборатории экспериментальной экономики МФТИ среди студентов был проведен эксперимент, воспроизводящий модель информационных каскадов в контролируемых условиях. В эксперименте приняли участие 14 человек, было проведено 20 повторений (попыток) игры.

Реализованная в среде z-Tree [5] игра описывается следующим образом. Есть две урны: A , в которой 3 красных и 2 белых шара, и B , в которой 2 красных и 3 белых шара. До начала каждой попытки случайным образом определяется, из какой урны будут доставаться шары. Участники знают только, что урны равновероятны.

В каждой попытке участники упорядочиваются случайным образом. Каждому участнику приватно показывается, какого цвета шар вынут из урны, после чего шар в урну возвращается. Участник должен угадать урну: A или B . Кроме приватного сигнала (цвет шара) участник видит выборы урн всех участников до него. Все, кто правильно угадал урну, в конце периода получают по 50 очков, а кто не угадал – 0 очков. Очки по попыткам суммируются.

Результаты

В результатах эксперимента оценивались следующие параметры: правильность, длина и начало каскадов, а также правильность выбора игроков при получении корректного и некорректного сигналов.

Всего в эксперименте наблюдалось 28 каскадов, из которых 12 оказались неправильными. Распределение каскадов по длине и по точке начала представлено на рис. 1. В табл. 1 представлены средние значения для серии из 20 попыток.

Чаще всего встречались каскады длины 2 или 4: 25% всех каскадов имеют длину 2 и столько же длину 4. В среднем, неправильные каскады короче правильных.

Относительно точек начала каскадов на рис. 1 можно выделить 3 пика: 3, 6 и 11. На этих шагах каскады начинались чаще всего. Распределены правильные и неправильные каскады по точкам начала более или менее одинаково.

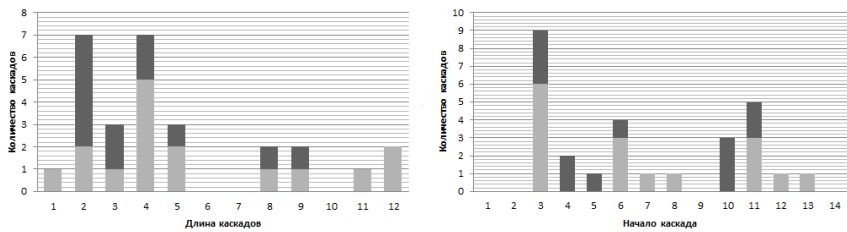


Рис.1. Слева: количество каскадов различной длины, справа: количество каскадов, начавшихся на заданном шаге. Темным цветом обозначены неправильные, светлым – правильные каскады на обеих диаграммах

Таблица 1

Средние значения длины и точки начала каскадов

Каскады	Длина	Точка начала
Все	4,86	6,75
Правильные	5,63	6,81
Неправильные	3,83	6,67

Рассмотрим агрегированное поведение участников эксперимента. Разделим все выборы участника на 4 типа: следовал сигналу и выиграл, сле-

довал сигналу и проиграл, не следовал сигналу и выиграл, не следовал сигналу и проиграл. В табл. 2 представлены доли выборов каждого типа от всех выборов. Можно заметить, что в целом игроки предпочитают следовать сигналу, и это дает выигрыш в 65% случаев, что приблизительно соответствует вероятности правильного сигнала.

Таблица 2

Доля выигрышей при следовании и игнорировании сигнала.

	Выиграл	Проиграл	Доля выигрышей
Следовал сигналу	45,36%	24,64%	64,80%
Не следовал сигналу	12,86%	17,14%	42,86%

В табл. 3 представлена информация, аналогичная табл. 2, но только для выборов, произведенных в каскадах. В каскаде игроки чаще игнорируют свой сигнал, и доля выигрышей при этом повышается.

Таблица 3

Доля выигрыша при следовании сигнала в каскадах.

	Выиграл	Проиграл	Доля выигрышей
Следовал сигналу	45,59%	16,91%	72,94%
Не следовал сигналу	20,59%	18,38%	52,83%

Рассмотрим динамику изменения правильности выбора игроков в зависимости от их очередности и корректности полученных сигналов. Рисунок 2 иллюстрирует данную зависимость. Из рис. 2 можно сделать вывод, что вероятность правильного выбора выше при получении корректного сигнала. Однако в точках 3, 7 и 10 доля выигрышей при получении некорректного сигнала практически равна или превышает аналогичный параметр при получении корректного сигнала. В данных точках или в соседних с ними, как видно на рис. 1, чаще всего образуются каскады, т.е. игроки на этих шагах больше, чем обычно, полагаются на выборы предшественников, игнорируя приватный сигнал.

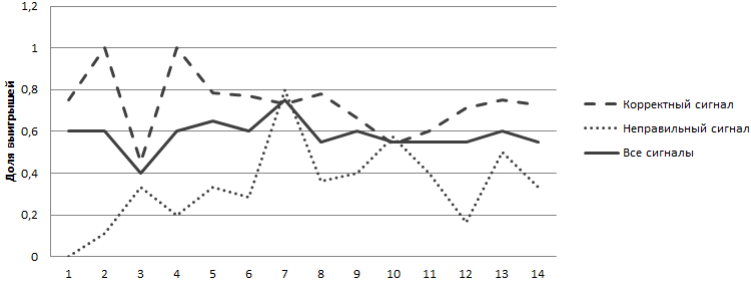


Рис. 2. Зависимость правильности выбора от очередности и корректности сигнала. Пунктирная линия – корректные сигналы, точечная – некорректные сигналы, сплошная – все сигналы

Параметры равновесия дискретного отклика

Проследим динамику изменения мнений в зависимости от λ . Для заданного p_t существуют ровно два значения для p_{t+1} в зависимости от выбора c_t . Обозначим

$$p_{t+1} = P(\omega = A | H_t, c_t) = \begin{cases} p_t^+ \equiv P(\omega = A | H_t, c_t = A), \\ p_t^- \equiv P(\omega = A | H_t, c_t = B). \end{cases}$$

Посчитаем вероятность перехода от p_t к p_t^+ от p_t^- (т.е. вероятность выбора A и B для правильных состояний $\omega = A$ и $\omega = B$).

Вероятность выбора A при $\omega = A$:

$$\begin{aligned} T_t^{\omega=A} &= P(c_t = A | H_t, \omega = A) = P(c_t = A | H_t, s_t = a)P(s_t = a | \omega = A) + \\ &+ P(c_t = A | H_t, s_t = b)P(s_t = b | \omega = A) = \\ &= \frac{q}{1 + \exp(\lambda(1 - 2\pi_t^a(p_t)))} + \frac{1-q}{1 + \exp(\lambda(1 - 2\pi_t^b(p_t)))}, \\ 1 - T_t^{\omega=A} &\text{ – вероятность выбора } B. \end{aligned}$$

Аналогично для состояния $\omega = B$ вероятность, что агент t выберет A , равна

$$\begin{aligned} T_t^{\omega=B} &= \frac{1-q}{1 + \exp(\lambda(1 - 2\pi_t^a(p_t)))} + \frac{q}{1 + \exp(\lambda(1 - 2\pi_t^b(p_t)))}, \\ 1 - T_t^{\omega=B} &\text{ – вероятность выбора } B. \end{aligned}$$

Используя правило Байеса, можем посчитать значения, которые может принимать p_{t+1} :

$$p_t^+ \equiv P(\omega = A | H_t, c_t = A) = \frac{P(\omega = A, c_t = A | H_t)}{P(c_t = A | H_t)} = \frac{p_t T_t^{\omega=A}}{p_t T_t^{\omega=A} + (1-p_t) T_t^{\omega=B}},$$

$$p_t^- \equiv P(\omega = A | H_t, c_t = B) = \frac{p_t (1 - T_t^{\omega=A})}{p_t (1 - T_t^{\omega=A}) + (1-p_t) (1 - T_t^{\omega=B})}.$$

Таким образом, можно неявно записать: $p_t = p_t(c_1, \dots, c_{t-1} | \lambda)$. Вероятность того, что при наборе $\{\lambda, s_t, (c_1, \dots, c_{t-1})\}$ игрок t выберет A , есть

$$P(c_t = A | \lambda, s_t, c_1, \dots, c_{t-1}) = \frac{1}{1 + \exp\left(\lambda \left(1 - 2\pi_t^{s_t}(c_1, \dots, c_{t-1} | \lambda)\right)\right)},$$

$$P(c_t = B | \lambda, s_t, c_1, \dots, c_{t-1}) = 1 - P(c_t = A | \lambda, s_t, c_1, \dots, c_{t-1}).$$

Вероятность получения последовательности выборов $c = (c_1, \dots, c_T)$ при заданной последовательности сигналов есть

$$l(c | \lambda) = \prod_{t=1}^T P(c_t | \lambda, s_t, c_1, \dots, c_{t-1}).$$

Наконец, предполагая независимость между последовательностями, получим вероятность наблюдения набора из M последовательностей $\{c^1, \dots, c^M\}$: $L(c^1, \dots, c^M | \lambda) = \prod_{m=1}^M l(c^m | \lambda)$, которую максимизируем по λ : $L(c^1, \dots, c^M | \lambda) \rightarrow \max_{\lambda}$ (метод максимального правдоподобия).

На языке MATLAB была написана процедура, которая на основании экспериментальных данных возвращает $\log L$, получая на вход параметр λ . После максимизации логарифма функции правдоподобия получаем следующие параметры равновесия дискретного отклика:

$$\lambda = 4.22, \quad \log L = -149.9.$$

Сравнение равновесий с экспериментом

Для найденного параметра λ и последовательности сигналов, реализованной в эксперименте, было сгенерировано 100 разыгрываний данной игры, в которой каждый фиктивный игрок использовал смешанную стра-

тегию, соответствующую найденному равновесию дискретного отклика. Аналогичная процедура была проведена для фиктивного разыгрывания равновесия Нэша. Средние показатели этих разыгрываний в сравнении с экспериментом приведены в табл. 4.

Из табл. 4 следует, что QRE более приближено к реальным показателям, чем равновесие Нэша. Хорошо в модели QRE оцениваются число каскадов, соотношение правильных и неправильных, а также точка начала каскада.

Таблица 4

Сравнение параметров каскадов

	Эксперимент	Равновесие Нэша	Квантильное равновесие
Число каскадов	28	20	30
Число правильных	16	13	20
Процент правильных	57,14%	65,00%	66,67%
Средняя длина	4,86	10,75	2,76
Среднее начало	6,75	4,1	6,77

Сравним динамику правильности выборов игроков, полученную на основе экспериментальных данных, а также симуляций на основе равновесия Нэша и квантильного равновесия. На рис. 3 показаны графики для каждого набора данных. В случае с QRE было проведено 100 симуляций и представлены средние для них значения.

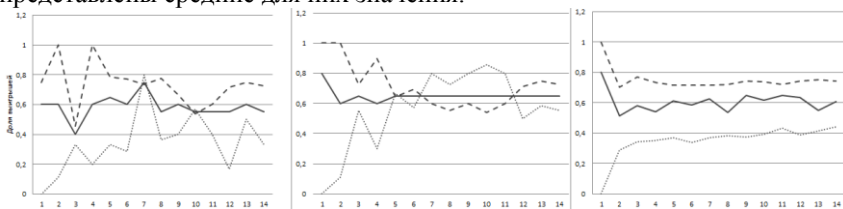


Рис. 3. Слева направо: экспериментальные данные, равновесие Нэша, QRE (среднее по 100 симуляциям). Обозначения такие же, как на рис. 2

Как и в экспериментальных данных, общая правильность выбора в QRE (сплошная линия) флуктуирует около значения 0.6, а не выравнивается на 5-м шаге, как в равновесии Нэша.

Заключение

Проведенные эксперименты продемонстрировали формирование информационных каскадов в лабораторных условиях. Многообразие вариантов формирования и разрушения каскадов лишь в малой степени соответствует равновесию Нэша. При отходе от гипотезы о полной рациональности участников в пользу равновесия дискретного отклика (QRE) получается более адекватная модель агрегированного поведения участников. Это подтверждается путем сравнения двух вариантов фиктивного разыгрывания (на основе равновесия Нэша и на основе QRE) и наблюдаемым поведением участников эксперимента.

Работа выполнена при финансовой поддержке грантом РФФИ 16-01-00633А.

Авторы выражают искреннюю признательность А. А. Коновалову [6], любезно предоставившему возможность использовать свою программу на z-Tree в наших экспериментах.

Литература

1. *Bikhchandani, S., Hirshleifer, D., and Welch, I.* A Theory of Fads, Fashion, Custom, and Cultural Change as Information Cascades // *Journal of Political Economy*. October, 1992. P. 100.
2. *McKelvey, R.D. and Palfrey, T.R.* Quantal Response Equilibria in Normal Form Games // *Games and Economic Behavior*. 1995. 7. P. 6–38.
3. *McKelvey R.D., Palfrey T.R.* Quantal response equilibria for extensive form games // *Experimental economics*. 1998. 1. P. 9–41.
4. *Goeree J.K., Palfrey T. R., Rogers B.W., McKelvey R.D.* Self-Correcting information cascades // *California Institute of Technology, Division of the humanities and social sciences*. – 2004, Social working paper 1197.
5. *Fischbacher U.* z-Tree: Zurich toolbox for ready-made economic experiments // *June 2007*. V. 10. Issue 2. P. 171–178.
6. *Коновалов А., Паклина Т.* Информационные каскады с участием экспертов: теория и эксперимент // *Научные доклады Института экономики Уральского отделения РАН*. Екатеринбург, 2010, <http://www.studfiles.ru/preview/4198249/>.

Получено 20.04. 2017

В. А. Дьяченко^{1,2}, И. С. Меньшиков^{1,3}

¹Московский физико-технический институт
(государственный университет)

²Институт системного программирования РАН

³Вычислительный центр им. А. А. Дородницына ФИЦ ИУ РАН

E-mail: vdyachenko@ispras.ru

МОДЕЛИРОВАНИЕ ИНДИВИДУАЛЬНОГО ПОВЕДЕНИЯ УЧАСТНИКОВ ДВОЙНОГО АУКЦИОНА С ЗАКРЫТЫМИ СТАВКАМИ

В двойном аукционе с закрытыми ставками теоретическое равновесие Байеса–Нэша (РБН) известно [1, 2]. Эксперименты, проведенные в Лаборатории экспериментальной экономики МФТИ, показывают, что на агрегированном уровне некоторые особенности РБН выявляются. В то же время индивидуальное поведение участников весьма разнообразно и сильно отличается от равновесного. Короткая серия повторений эксперимента в рамках лаборатории не позволяет адекватно оценить стратегию каждого участника. Предлагается по наблюдаемому поведению участников смоделировать их стратегии и проводить длинную серию вычислительных экспериментов для всесторонней оценки эффективности индивидуального поведения и сравнения его с равновесным. Данный подход возможно обобщить на достаточно широкий класс лабораторных игр.

Ключевые слова: равновесие Байеса–Нэша, двойной аукцион, лабораторные эксперименты, игроки-роботы.

Постановка задачи

В двойном аукционе участвуют два игрока: продавец (seller) и покупатель (buyer). Они одновременно и независимо друг от друга подают свои заявки в виде некоторого числа. Для продавца число P_s означает желание продать некоторый неделимый объект по данной цене или выше. Для покупателя число P_b означает желание купить этот объект по данной цене или ниже.

Сделка. Если $\Delta = P_b - P_s \geq 0$, то сделка заключается по цене $P = \frac{P_s + P_b}{2}$, т.е. по средней из возможных цен при данных заявках.

Если $\Delta < 0$, то сделка не заключается, и оба участника получают выигрыш, равный 0.

Если сделка заключена по цене P , то выигрыши участников подсчитываются с учетом их индивидуальной оценки стоимости объекта.

Для продавца выигрыш от сделки равен $P - v_s$.

Для покупателя выигрыш от сделки равен $v_b - P$.

Каждый участник знает только свою оценку, а про оценку стоимости другого известно только, что она является случайной величиной, равномерно распределенной от a до b для продавца и от c до d для покупателя.

Двойной аукцион можно представить в виде следующей байесовской игры.

Действия: ставки игроков p_b и p_s .

Тип игрока: индивидуальные стоимости товара: $v_s \in [a, b]$ и $v_b \in [c, d]$.

Представления: равномерное распределение стоимостей на $[a, b]$ и $[c, d]$.

Выигрыши: $U_b = v_b - \frac{p_b + p_s}{2}$; $U_s = \frac{p_b + p_s}{2} - v_s$, при условии, что $p_b \geq p_s$, иначе $U_b = U_s = 0$.

Стратегии игроков есть функции $p_s(v_s): [a, b] \rightarrow [a, d]$ и $p_b(v_b): [c, d] \rightarrow [a, d]$. При этом на p_s и p_b накладываются ограничения: $p_s \geq v_s$, $p_b \leq v_b$, т.е. для игроков отсутствует возможность «уйти в минус».

В рамках эксперимента в Лаборатории экспериментальной экономики МФТИ с участием студентов 5 курса в 2017 году значения индивидуальных стоимостей товара лежали на отрезках $v_s \in [0, 60]$ и $v_b \in [40, 100]$. Эксперимент состоял из 30 периодов, в каждом из которых участники автоматически случайным образом разбивались на пары. В каждой паре один участник получал роль продавца, а другой – роль покупателя. После этого они играли друг с другом в двойной аукцион с вышеуказанными распределениями v_s и v_b . Как было доказано в работе [2], в этом случае

существует невырожденное теоретическое РБН в классе кусочно-линейных стратегий: $\left[\max\left(\frac{2}{3}v_s + 25; 35\right), \min\left(\frac{2}{3}v_B + \frac{25}{3}; 65\right) \right]$.

В то же время возникает ряд новых задач.

- Теоретическое РБН может не подходить как эталонная стратегия, при сравнении с которой оценивается эффективность стратегий участников, так как в целом поведение игроков отличается от предсказанного математической теорией, и потому какая-то другая стратегия в реальном эксперименте может оказаться более эффективной, нежели РБН. В связи с этим нужно иметь способ оценки эффективности самой РБН при имеющемся наборе игроков.
- Лабораторные эксперименты ограничены по числу периодов, и потому на маленьких выборках возможна ситуация, когда участник использует стратегию, близкую к оптимальной, но, тем не менее, получает достаточно невысокий выигрыш из-за неблагоприятных индивидуальных стоимостей товара. Необходимо иметь возможность оценивать эффективность стратегии участника, абстрагируясь от полученного им выигрыша в рамках эксперимента. Также хочется иметь представление о зависимости стабильности исхода эксперимента от числа периодов.

Предлагаемый подход к исследованию стратегий

Основная проблема оценки эффективности стратегий участников эксперимента состоит в небольшом числе периодов. Это ограничение приводит к тому, что порой трудно понять, в чем кроется причина полученного участником результата – в его стратегии либо же в значениях стоимостей, сгенерированных генератором псевдослучайных чисел. Чтобы проанализировать поведение игроков, необходимо получить возможность наблюдения итогов эксперимента с гораздо более значительным числом периодов, достаточном для того, чтобы сгладить влияние генератора псевдослучайных чисел на конечный результат участников. При этом такое число периодов, при котором можно было бы оставить сомнения в значительном влиянии генератора на итог, значительно превышает возможное значение для проведения реального эксперимента. В связи с этим остается возможность только смоделировать новый вычислительный эксперимент, используя данные о поведении участников в исходной игре.

В рамках рассматриваемого эксперимента было сыграно 30 периодов. Можно заметить, что роли продавца и покупателя в данной игре симметричны, т.е. если из 100 вычесть стоимости покупателя, то такое распреде-

ление стоимостей совпадет с распределением стоимостей продавца. Из этого можно ожидать, что, применив такую же операцию к ставкам участника эксперимента в роли покупателя, мы получим значения, близкие к ставкам игрока в роли продавца при близких стоимостях продавца к соответствующим стоимостям покупателя. В идеальном случае если для фиксированного игрока в каких-то двух разных периодах, в который он имел разные роли, выполняется соотношение $v_s = 100 - v_B$, то ожидается $p_s = 100 - p_B$. Исследование зависимостей ставок от стоимостей и ролей в исходном эксперименте показало, что поведение большинства участников в целом соответствует приведенному рассуждению.

Исходя из вышесказанного, предлагаемый подход к моделированию нового вычислительного эксперимента выглядит следующим образом.

1. Определим операцию **инвертирования** значений стоимости и ставки для игрока в роли покупателя:

$$v'_B = 100 - v_B, \quad p'_B = 100 - p_B.$$

2. Для каждого участника фиксируем отдельную **таблицу** зависимости сделанных в исходной игре ставок от стоимостей, причем в случае, когда игрок действовал в качестве покупателя, рассматриваются его инвертированные значения. Таким образом, для каждого участника имеется таблица из 30 отображений из значений на множестве $[0, 60]$ в значения на множестве $[0, 100]$. Обозначим множество стоимостей, входящих в эту таблицу, через V' , а множество ставок – через P' . Ставку, соответствующую стоимости $v \in V'$, обозначим через $p(v)$.

3. Определим **функцию генерирования** ставки $f : [0, 60] \rightarrow [0, 100]$ следующим образом:

$$f(x) = \begin{cases} p(v_0); & p(v_0) \geq x, \\ x; & p(v_0) \leq x, \end{cases} \quad \text{где } v_0 = \underset{v \in V'}{\operatorname{argmin}} |x - v|.$$

Таким образом, функция f от числа x возвращает значение ставки $p_0 = p(v_0)$, соответствующей наиболее близкой к x стоимости v_0 из таблицы стоимостей участника, если это значение ставки больше x . В случае если $p_0 < x$, то из этого следует, что $v_0 \leq p_0 < x$, и при этом v_0 достаточно близко к x . В связи с этим можно утверждать, что при близких к v_0 стоимостях игрок в ис-

ходной игре называл бы близкие к стоимостям ставки, поэтому принимаем в этом случае $f(x) = x$.

4. Совокупность таблицы участника исходного эксперимента и функции генерирования ставки будем сопоставлять соответствующему игроку. Будем называть эту сущность **роботом**.
5. Далее запускается эксперимент с теми же правилами, что и в исходном случае: роботы случайным образом объединяются в пары, генератор псевдослучайных чисел для каждого робота формирует число v_{new} из отрезка $[0,60]$, после чего определяется ставка $p_{new} = f(v_{new})$. Для покупателя происходит дополнительная операция, переводящая полученные стоимость и ставку на соответствующие значения. Эта операция обратна по смыслу определенному ранее инвертированию, но совпадает по производимому при этом действию: стоимости и ставки вычитаются из 100.
6. После этого для каждой пары определяется, состоялась ли сделка, после чего для каждого участника в паре подсчитывается выигрыш в периоде и общий выигрыш.

В соответствии с приведенным описанием было проведено несколько сеансов моделирования игры. Число периодов было увеличено до 30 000. В исходном эксперименте все результаты делились на 5 с целью нормирования, потому в моделированных играх общие результаты делились на 5000 с целью сравнения с исходным экспериментом. Данные представлены в табл. 1.

Таблица 1

№	I	II	III	IV	V	VI	VI.	VIII	IX
1	90.2	77.8	77.5	76.9	77.5	78.4	78.2	-	-
2	82.9	79.1	80.5	79.4	79.8	80.7	80.5	-	-
3	101.4	116.3	117.3	115.9	114.9	116.4	116.3	111.8	111.3
4	116.5	117.7	117.6	116.7	116.4	118	118.4	112.4	112.3
5	103.8	117.4	117.3	117.4	116.1	117.4	116.8	112.5	112.8
6	68.4	103.5	103.8	102.3	101.9	104	103.9	96.5	96
7	93.9	109.4	108.5	110.8	108.1	110.9	110.5	105.4	104.2
8	120.5	102.5	103.3	101.8	101.6	102.8	103.6	97.6	97.6

9	77.6	110.4	110.9	109.5	109.2	111.3	110.9	107.5	106.7
10	117.7	114.1	113.2	112.2	112.4	113.3	114.4	108.8	108.8
11	78	115.5	116.1	114.6	114.8	115.8	115.8	111.2	111.1
12	113.3	102.9	104.2	104	101.8	104.1	104.3	97.1	97.1
13	88.4	108.2	108.2	106.4	107.5	109.1	108.8	103	102.1
14	111.5	102.7	103	101.4	102.5	102	103.4	98.3	98.8
bot1	-	-	-	113.1	112.4	113	113.6	108.7	108.9
bot2	-	-	-	112.9	113.2	-	-	109.3	109
bot3	-	-	-	-	-	114.6	113.3	-	-
avg	97.4	105.5	105.8	106	105.6	107	107	105.7	105.5

В столбце «№» перечислены номера участников исходного эксперимента, а также роботы в моделируемых играх. Bot1 и bot2 – роботы, реализующие РБН, bot3 – робот, реализующий стратегию вида

$$p = \begin{cases} 40, & v \leq 40, \\ v, & v > 40. \end{cases}$$

Данный робот был добавлен, т.к. придерживавшиеся близких стратегий участники 8 и 10 в исходном эксперименте добились наилучших результатов (участник 8 значительно отклонялся от указанной стратегии приблизительно в 23% случаев, участник 10 – ни разу). Последняя строка отображает данные о среднем выигрыше за сессию. Столбец I содержит информацию о результатах эксперимента, столбцы II–IX отражают итоги моделирований с 30 000 периодами. Знаком «-» отмечены роботы, отсутствовавшие в соответствующем сеансе игры. Последняя строка содержит информацию о среднем результате в сеансе. Как можно заметить, аналогичные варианты моделирования идут парами, при этом максимальное различие составило 2.5% (для робота 7 в моделированиях IV–V).

Выводы, которые можно сделать на основании полученных данных, следующие.

1.

Результаты, полученные в рамках исходного эксперимента в среднем ниже результатов моделирований II и III. Это может свидетельствовать о том, что при 30-периодной игре сгенерированные значения стоимостей оказались не очень удачными. Данный вывод подтверждается исследованием

исходной игры: только у 4 участников из 14 в роли продавцов средняя стоимость была ниже медианного значения 30, и всего у 3 игроков в роли покупателей средняя стоимость превышала 70.

2. Среднее различие значений в парах аналогичных моделирований составило порядка 0.5%, что позволяет говорить о достаточно высокой степени детерминированности получаемых при моделировании результатов.

3. Во всех моделируемых экспериментах лидерами по числу набранных очков стали роботы, соответствующие участникам 3, 4, 5, 11. Их стратегии сложно строго формализовать, но детальное рассмотрение показало, что в целом они обладают схожими чертами и более агрессивны, нежели предсказывается РБН.

4. В то же время игроки 8 и 10, показавшие максимальные результаты в исходном эксперименте, не являлись лидерами ни в одной из сессий моделирования. Это обстоятельство в совокупности с большими изменениями результатов ряда игроков в моделируемых играх по сравнению с исходной игрой (в особенности это заметно на участниках 6, 9 и 11) указывает на высокую роль значений генерируемых стоимостей в играх с небольшим числом периодов.

5. При подключении к модели роботов, играющих РБН или упомянутую выше пороговую стратегию, их результат не превосходит ни в одной сессии результата лидеров, указанных в пункте 3. Это свидетельствует о том, что в рассматриваемой игре теоретическое РБН не является максимизирующей выигрыш стратегией. Данная тенденция сохраняется даже в случае исключения из моделирования игроков 1 и 2, игравших самые мягкие стратегии (в большинстве случаев эти игроки в качестве ставок называли свои сгенерированные стоимости).

Интерес представляет также вопрос о необходимом числе периодов для получения стабильных результатов при моделировании эксперимента. Для исследования этого вопроса запускались моделирования в парах с числом периодов 30, 100, 300, 1000, 3000, 10 000, 30 000 (для этого случая использовались ранее полученные данные), 100 000, 300 000 и определялись среднее и максимальное отклонения соответствующих результатов в паре в каждом случае. Получились следующие значения:

	30	100	300	1000	3000
avg, %	26.4	12.3	8.3	2.2	1.4
max, %	64.8	33.6	23.9	4.6	7.2
	10 000	30 000	100 000	300 000	
avg, %	1	0.8	0.3	0.2	
max, %	2.3	2.5	0.9	0.5	

Таким образом, при увеличении числа периодов до 10 000 средний результат в двух разных моделированиях отличался примерно на 1%, а при увеличении числа периодов до 100 000 ограничение в 1% было достигнуто уже для максимального колебания.

Заключение

По результатам проведенной работы можно сделать следующие выводы.

1. В рамках лабораторного и вычислительных экспериментов теоретическое РБН в кусочно-линейных стратегиях не являлось оптимальной стратегией, максимизирующей ожидаемый выигрыш игрока.
2. В лабораторных условиях ввиду невозможности проведения значительного числа периодов результат участника существенно зависит от сгенерированных для него значений стоимостей. Для уменьшения этой зависимости до величины порядка 1% необходимо провести около 10 000 периодов.

Рассмотренный подход можно обобщить на широкий класс лабораторных игр. Например, существуют вариации рассматриваемой игры, в которых сделка заключается по цене одной из сторон (по цене продавца или по цене покупателя), или же в которых введены не две стороны, а три (продавец, покупатель, транспортер).

Работа выполнена при финансовой поддержке грантом РФФИ 16-01-00633А.

Литература

1. *Меньшиков И.С.* Лекции по теории игр и экономическому моделированию. – 2 изд. – М.: ООО «Контакт Плюс», 2010.
2. *Меньшиков И.С., Милякин С.Р.* Равновесные стратегии в двойном аукционе с закрытыми ставками: теория и эксперимент // Моделирование процессов обработки информации. – М.: МФТИ, 2014. – С. 52– 61.

Получено 15.04.2017 г.

УДК 519.83

А. И. Шилова¹, И. С. Меньшиков^{1,2}

¹Московский физико-технический институт
(государственный университет)

²Вычислительный центр им. А. А. Дородницына ФИЦ ИУ РАН
E-mail: shilova@phystech.edu

ИССЛЕДОВАНИЕ ИГРЫ СОБСТВЕННИК–ДИРЕКТОР–АГЕНТ

Целью данной работы является исследование игры Собственник–Директор–Агент. Была построена теоретико-игровая модель, соответствующая проблеме конфликта интересов внутри фирмы, а также был проведен анализ этой модели и базирующийся на ней лабораторный эксперимент. Приводится сравнение двух моделей игры с добавлением Директора и без него как с позиций теории игр, так и на основе анализа результатов проведенных лабораторных экспериментов.

Ключевые слова: асимметричность информации, совершенное по подыграм равновесие Нэша, игра собственник–агент, лабораторный эксперимент.

Введение

Игра Собственник–Агент известна довольно давно. В основе интереса к этому взаимодействию лежит тот факт, что во главе каждой организации стоит собственник, и когда возникают какие-то задачи, то собственник, как правило, не выполняет их самостоятельно, а перепоручает их некоторому нанятому агенту, которому предложено некоторое возмещение за выполненную работу. Пример из современного общества: акционеры компании и генеральный директор (топ менеджер). В дальнейшем для простоты будем везде рассматривать данную игру с точки зрения фирмы.

Центральная идея, которая лежит в основе данной модели, состоит в том, что у собственника всегда много дел, чтобы разрешать самому проблемы фирмы. Для этого он нанимает агента. Но помимо этого, занятость собственника означает также и отсутствие времени на контроль над действиями агента. Отсюда появляется проблема асимметричности информации у игроков. Чтобы это компенсировать, собственники могут ввести дополнительные поощрения для агентов, такие как бонусы, размер и вы-

плата которых зависят от того, как хорошо агент выполнит задание. Другой вариант: собственник может нанять дополнительно человека, задача которого будет следить за работой агента. В фирме эту роль исполняет совет директоров.

Обычно, говоря о чьей-то работе, не задумываешься, как много действий может стоять за ней. Возьмем хотя бы работу учителя: он должен научить детей писать, читать, считать, но также творчески мыслить, уметь работать в команде и т.д. Вводя всякого рода поощрения для учителей, нужно понять, как измерить качество их работы. Первые три упомянутые умения у учеников можно с легкостью проверить при помощи тестов, тогда в контракте для учителей можно прописать строго установленную ставку, которая им выплачивается за каждую отличную оценку. Это называется объективное оценивание выполненной работы. Однако проверить творческое мышление и социальные навыки у детей уже не так просто, нельзя установить точный числовой критерий, который легко отслеживается. Отсюда может возникнуть следующая проблема: ты получаешь то, за что заплатил, описанное в [1]. При данном объективном оценивании учителя не заинтересованы в том, чтобы развивать остальные навыки у своих учеников. В противоположность этому можно использовать субъективное оценивание работы: собственник сам решает хорошо или плохо выполнил работу агент и соответственно со своей оценкой выплачивает бонус.

Игра Собственник–Агент с двумя видами контрактов

В этой игре будет две роли, причем агент должен выполнить два задания, то есть выбрать

$$e_1, e_2 : e_i \in \{1, 2, \dots, 10\}, i = 1, 2,$$

а собственник должен выбрать тип контракта: сдельная ставка или бонус. В контракте первого типа используется объективное оценивание, а в контракте второго типа субъективное оценивание, то есть в случае сдельной ставки прибыль агента напрямую зависит от количества прикладываемых усилий по первому заданию, а в случае бонусного контракта собственник уже после получения своей прибыли решает: вознаградить агента бонусом или нет. Эксперимент взят из [2]. Ниже представлены выигрыши игроков собственника и агента, где каждое первое уравнение из двух уравнений соответствуют выигрышу, полученному в случае контракта типа сдельная ставка, а второе уравнение – бонусному контракту.

$$\begin{cases} 10e_1e_2 - w - se_1, \\ 10e_1e_2 - w - b, \\ w + se_1 - c(e_1 + e_2), \\ w + b - c(e_1 + e_2). \end{cases}$$

В данных обозначениях w представляет зарплату агента, s – размер сдельной ставки, b – размер бонуса, а $c(e_1, e_2)$ – функцию затрат, которая выглядит следующим образом:

$$c(e_1 + e_2) = \max\{5(e_1 + e_2), 10(e_1 + e_2) - 50\}.$$

Несложно найти для данной модели совершенное по подыграм равновесие Нэша. Если собственник выбрал сдельную ставку, то у агента нет мотивации работать по второму заданию, то есть он выбирает минимум усилий по этому заданию и $e_2 = 1$. Если внимательно посмотреть на функцию затрат $c(e_1 + e_2)$, то очевидно, что агент выберет $e_2 = 1$, если $s < 5$; $1 \leq e_2 \leq 9$, если $s = 5$; $e_2 = 9$, если $5 < s \leq 10$; $e_2 = 10$, если $s \geq 10$. Рассмотрев все возникшие ситуации, несложно убедиться, что собственник выберет $s = 5$, а $w = 5$. В таком случае агент может выбрать $e_1 = 9, e_2 = 1$. Выигрыши собственника и агента соответственно равны $M^P = 10 \times 1 \times 9 - 5 - 5 \times 9 = 40$ и $M^A = 5 + 5 \times 9 - 50 = 0$. Это не единственное равновесие Нэша, но это равновесие доставляет наилучшие выигрыши для обоих игроков.

Если же собственник выбрал бонусный контракт, то, очевидно, собственник в равновесии не выплатит бонус агенту, потому что это не в его интересах. Агент должен это понимать, поэтому, согласившись на контракт, он предпримет минимальные усилия по обоим заданиям. Чтобы агент принял контракт, необходимо $w \geq 10$. Значит, равновесие Нэша, совершенное по подыграм: $w = 10, b = 0, e_1 = 1, e_2 = 1$. При таких значениях параметров и собственник, и агент получают нулевой выигрыш.

Из этих рассуждений следует, что бонусный контракт уступает по выигрышам сдельной ставке. Значит, если оба игрока действуют чисто в своих собственных интересах, то собственник должен предпочитать контракт типа сдельной ставки.

Игра Собственник–Директор–Агент в случае бонусного контракта

Привнесем изменения в игру, описанную ранее. Теперь в контракте собственник также прописывает (e_1^*, e_2^*) уровни желаемых усилий по каждому заданию. Главное, добавим в игру еще одну роль: Директора. Директор будет выполнять мониторинговую функцию, таким образом, он будет выравнивать асимметричность информации. Данная идея была взята из [3].

В игре директор начинает действовать после того, как агент примет решение согласиться на предлагаемый контракт, и до того, как агент выберет уровень усилий (e_1^A, e_2^A) . Директор будет решать, как строго он будет осуществлять контроль над действиями агента, при этом агент, когда совершает действия по каждому из заданий, знает уровень контроля. Директор, как и агент, выбирает уровень усилий (e_1^D, e_2^D) , и тогда, если он выбрал $e_1^D = 10$, то агент не может отклониться от e_1^* , не неся убытков, если $e_1^D = 9$, то агент может отклониться от e_1^* только на 1, не неся убытков и т. д. Если агент несет убытки, то это значит, что часть обещанных ему денег, определяемая функцией $v(e_1^A, e_2^A, e_1^D, e_2^D, e_1^*, e_2^*)$, переходит директору (теперь переменные и функции, относящиеся к агенту, носят индекс A, а к директору — D). Доход директора будет складываться из постоянной зарплаты w^D , а также доли от прибыли собственника p (аналог того, что в жизни некоторые директора владеют акциями компаний, в которых они работают). В итоге мы получаем следующие выигрыши для игроков:

$$M^P = 10(1 - p)e_1^A e_2^A - w^A - w^D - b;$$

$$M^D = 10pe_1^A e_2^A + v(e_1^A, e_2^A, e_1^D, e_2^D, e_1^*, e_2^*)w^A + w^D - c^D(e_1^D + e_2^D);$$

$$M^A = (1 - v(e_1^A, e_2^A, e_1^D, e_2^D, e_1^*, e_2^*))w^A + b - c^A(e_1^A + e_2^A);$$

где положим

$$c^D(e_1^D + e_2^D) = \frac{1}{2}c^A(e_1^D + e_2^D),$$

а функция $c^A(e)$ совпадает с функцией затрат для предыдущей игры. Далее будем использовать кусочно-линейную функцию $0 \leq v \leq 1$ такую, что при $(e_1^*, e_2^*) = (10, 10)$ она принимала бы значение, равное 1, если агент совершает минимум усилий по каждому заданию, в то время как директор выбирает не минимум усилий по мониторингу выполнения каждого зада-

ния, и тогда директор может его наказать так, что он забирает всю зарплату агента; и функция v принимала бы значение 0, если усилия агента по обоим заданиям не меньше, чем усилия директора по мониторингу, т.е. директор не может обнаружить какие-то «недоработки».

Используя метод обратной индукции для данной игры, можно найти совершенное по подыграм равновесие Нэша. Оно существует и не является единственным. Самое выгодное равновесие для собственника таково. Сначала собственник предлагает агенту контракт: $w^A = 180, b = 0, (e_1^*, e_2^*) = (10, 10)$, а директору: $w^D = 425/99, p = 7/99$. Затем агент и директор соглашаются с предложенным им контрактом. Потом директор решает, что будет следить за агентом с усилиями $(e_1^D, e_2^D) = (10, 10)$. Агент в свою очередь решает, что он будет работать с $(e_1^A, e_2^A) = (10, 10)$. Тогда выигрыши у участников будут следующие: $M^P = 745, M^D = 0, M^A = 30$. Это будет одним из равновесий Нэша, так как при изменении одного из параметров, за который отвечает игрок, он не улучшает его выигрыш.

Есть также и другие равновесия Нэша. Например, равновесием Нэша будет:

$$w^A = 90, b = 0, (e_1^*, e_2^*) = (7, 7), w^D = 0, p = 1/4, \\ (e_1^A, e_2^A, e_1^D, e_2^D) = (5, 5, 9, 9).$$

Выигрыши: $M^P = 97.5, M^D = 17.5, M^A = 20$. Это равновесие также можно вычислить при помощи обратной индукции.

Экспериментальные результаты

Для каждой из описанных выше моделей были проведены эксперименты в лаборатории Экспериментальной экономики МФТИ на основе программы, разработанной в среде z-Tree [4]. Ниже будут приведены краткий обзор главных результатов экспериментов и сравнение результатов для игр с директором и без директора.

Эксперимент Собственник–Агент проводился в три сессии. В первых двух сессиях принимали участие студенты четвертого курса ФУПМ, изучавшие основы теории игр, которые сформировали 4 пары, а в третьей сессии в основном присутствовали студенты МФТИ с первых двух курсов, которые поделились на 6 пар. В каждой сессии было 10 оплачиваемых периодов и 1 пробный, то есть всего было 11 периодов, а контрактов в итоге было 154. В начале каждого периода участники случайным образом распределялись в пары из собственника и агента, и каждый раз их

роль была тоже случайна. Перед экспериментами участники были проинструктированы, и перед началом эксперимента они ответили на контрольные вопросы. В ходе эксперимента участники могли пользоваться вспомогательной программой поддержки принятия решений для подсчета предварительных выигрышей по контрактам.

В результате были получены данные, где 6 контрактов оказались сбойными и всего 26 неразумных контрактов. Сбойными контрактами будем считать те, которые являются заранее невыгодными для собственника при выполнении агентом работы на рекомендуемом уровне и выплате всех поощрений, то есть $10e_1^*e_2^* - w - se_1^* < 0$ или $10e_1^*e_2^* - w - b < 0$. Неразумными контрактами назовем те, которые заранее невыгодны для некоторого игрока: для собственника или агента, то есть либо контракт сбойный, либо выполняется $w + se_1^* - c(e_1^* + e_2^*) < 0$ или $w + b - c(e_1^* + e_2^*) < 0$. При анализе результатов мы пренебрегаем такими контрактами, потому что они являются результатами ошибок или попыток обмануть агента. Получается, что теперь мы изучаем только 128 контрактов. Также для игр без директора не будем рассматривать отдельно случаи с игроками опытными и неопытными, так как ввиду простоты модели будем считать, что различия не столь существенны и ими можно пренебречь.

Один из интересных результатов этого эксперимента был факт того, что большая часть собственников предлагает агентам бонусный контракт, несмотря на то, что с точки зрения равновесия Нэша контракт типа сдельной ставки приносит большую прибыль собственнику. В итоге 76% контрактов были бонусными.

Также можно отметить, что при контракте типа сдельной ставки агенты стараются концентрировать свои усилия на первое задание, тогда как игнорируют второе задание. При бонусном контракте тратят примерно одинаковые усилия на оба задания. При этом $e_1 + e_2$ в среднем не сильно меняется в зависимости от различных контрактов.

Более того, в случае бонусного контракта чем больше усилий агент тратит на задания, тем более щедро вознаграждает его собственник. Причем бонус может в несколько раз превышать количество затрат на усилия, требуемые на выполнение этой работы.

Самый важный результат этого эксперимента – доминирование по Парето бонусного контракта над контрактом типа сдельная ставка. Следовательно, бонусный контракт является лучшим выбором с точки зрения совокупного выигрыша (суммарный выигрыш собственника и агента).

Наконец, можно найти наиболее распространенный сценарий игры. В этой игре большинство игроков предпочитали играть по следующей схеме: собственники назначали маленькую зарплату и довольно существенный бонус, размер которого может превышать зарплату в несколько раз. Здесь средняя зарплата $w \approx 35$, а средний выплаченный бонус равен $b \approx 285$. Таким образом, собственники страховали свой выигрыш от больших потерь в случае слабых усилий агента по каждому из заданий, но в то же время если агент выполняет по максимуму свою работу, то собственники не жадничали и нередко вознаграждали агентов бонусами (31 из 97 бонусных контрактов). Скорее всего, это результат таких социальных проявлений, как благодарность или справедливость.

Все перечисленные результаты согласуются с результатами, описанными в [2].

Перейдем теперь к игре с директором. Эксперимент Собственник–Директор–Агент проводился в две сессии. В первой сессии принимали участие 6 студентов, которые знают основы теории игр. Во второй сессии принимали участие 12 студентов МФТИ с разных курсов, которые в большинстве своем не знают основы теории игр. В первой сессии было проведено один пробный период и 10 оплачиваемых периодов, то есть всего 11 периодов, во второй сессии один пробный период и 8 оплачиваемых периодов, всего 9 периодов. Таким образом, было составлено 58 контрактов (22 в первой сессии и 36 во второй). В начале каждого периода игроки случайным образом распределялись на тройки Собственник–Директор–Агент, а также каждый период им давалась новая случайная роль.

Перед самым экспериментом они ознакомились с инструкцией, а также был разобран пример, где $p = 0.3$, $w^A = 350$, $w^D = 0$, $b = 0$, $(e_1^*, e_2^*) = (10, 10)$, которые вели к выигрышам $M^P = 350$, $M^D = 225$ и $M^A = 200$ при максимальных усилиях агента и директора. В этом примере доля директора и зарплата агента очень высоки, таким образом, директору выгодно выполнять свои обязанности на лучшем уровне, потому что в таком случае он получит большой доход, иначе может даже нести убытки, когда директор работает по максимуму, у агента не возникает соблазна отклоняться от заданного собственником уровня усилий, так как иначе он рискует потерять доход, который почти в два раза больше, чем затраты агента на совершение суммарного усилия. Выигрыши различных ролей отличаются уже не столь значительно, как выигрыши, получающиеся при совершенном по подыграм равновесии Нэша, о котором говорилось ранее, и поэтому такой фак-

тор, как справедливость, не должен сильно влиять на поведение игроков в рассмотренном примере.

Результаты сессий значительно отличаются, что объясняется «опытностью» участников первой сессии, поэтому в дальнейшем будем рассматривать их по отдельности.

Одним из позитивных эффектов появления директора в игре является уменьшение рисков, связанных с выигрышами как и собственника, так и агента, то есть выигрыши становятся более стабильными, менее диверсифицированными. Этот факт ясно отображается на рис. 1.

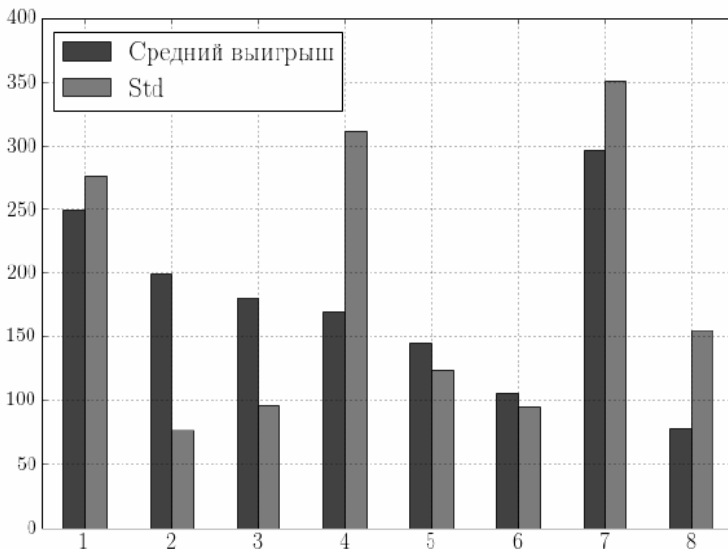


Рис. 1. Диаграмма, отображающая средние выигрыши и стандартные ошибки, где 1, 2, 3 – опытные собственник, агент, директор соответственно, 4, 5, 6 – неопытные собственник, агент и директор, 7 – собственник в игре без директора, 8 – агент в игре без директора

Средние выигрыши агентов выше в игре с директором, чем без него, но средние выигрыши собственников ниже. Но, с другой стороны, стандартные отклонения от среднего значения ниже как и у опытных игроков, так и у неопытных в игре с директором. Собственники при отсутствии директора нередко несли убытки, которые в некоторых случаях достигали значения 500. С появлением директора выигрыши игроков сглаживаются: не так часто случались отрицательные выигрыши, но при этом собственники не выигрывали такие большие суммы, как 1000, которые достига-

лись при бонусном контракте, когда зарплата была нулевая, а бонус высоким, но его не выплачивали, теперь собственники выплачивают свою долю директорам и высокую зарплату агентам, чтобы система работала, как следует. Средняя зарплата агента при участии опытных игроков составляла примерно $w^A = 306$, а в игре с неопытными игроками – около 207. Доля директора от дохода собственника была $p \approx 0.27$ у опытных и $p \approx 0.18$ у неопытных, средняя зарплата директора $w^D = 3$ у опытных и $w^D = 26$ у неопытных, средний объявленный бонус у опытных игроков был $b \approx 70$ и всего случаев выплаты бонуса – 8 раз из 22, а у неопытных $b \approx 155$, всего был выплачен в четырех случаях из 36. Стоит отметить, что величины, которые встречаются у игроков опытных, почти совпадают с теми, которые были предложены в примере.

Интересный факт, что если посмотреть на 10 контрактов из второй сессии, которые принесли наибольшую прибыль собственников, то легко заметить, что они были реализованы без участия директора. Более того, собственники сами форсировали отказ директора от участия, создав совсем невыгодные условия для директора: $p = 0$, $w^D = 0$. В итоге получались ситуации игры Собственник–Агент, где агенты выполняли задания на высшем уровне, а собственники не выплачивали бонус. Объяснением этому факту может более высокая сложность игры с директором, чем без него.

Что касается усилий агента, то у опытных игроков почти все агенты сработали с усилиями (10,10). У неопытных картина уже другая, но разница между усилиями по первому заданию и по второму заданию почти отсутствует. Также преимущественно выбирались высокие усилия по обоим заданиям, причем пара (10,10) была реализована 12 раз из 36, что составляет большинство по сравнению с другими парами. Так как стал чаще выбираться максимум усилий по каждому из заданий, то совокупный выигрыш собственника, директора и агента приближался к оптимальному значению. Отсюда следующий результат, что появление роли директора в игре увеличивает общественную прибыль, что подтверждает рис. 2.

Заключение

Проведенные лабораторные эксперименты по игре Собственник–Агент подтвердили широкое применение участниками бонусных контрактов, хотя это и противоречит теоретическому равновесию в данной игре. Добавление в лабораторную игру роли директора положительно влияет на

согласование интересов всех участников, что не вытекает однозначно из свойств найденного в работе совершенного по подыграм равновесия Нэша, однако наблюдается в экспериментах, особенно с опытными участниками, которые адекватно понимают специфику взаимодействия собственника, директора и агента.

Работа выполнена при финансовой поддержке грантом РФФИ 16-01-00633А.

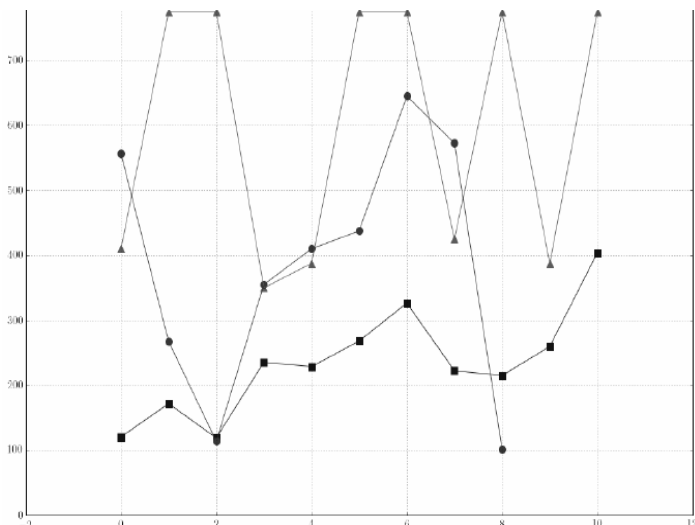


Рис. 2. Средние совокупные выигрыши в зависимости от номера периода. Линия с квадратными маркерами соответствует игре без директора, линия с круглыми маркерами – игре с директором с неопытными игроками, линия с треугольными маркерами – игре с директором с опытными игроками

Литература

1. *Kerr, Steven.* On the Folly of Rewarding A, While Hoping for B // New York, Academy of Management Journal, 1975.
2. *Fehr, Ernst; Schmidt, Klaus M.* Fairness and Incentives in a Multi-Task Principal-Agent Model // Munich Discussion Paper No. 2004-8 – Department of Economics University of Munich, 2004.
3. *Thomas Ruediger Smith.* Agency theory and its consequences. Master Thesis // Copenhagen Business School, 2011.
4. *Fischbacher U.* z-Tree: Zurich toolbox for ready-made economic experiments // Experimental Economics. – 2007. – V. 10. – Is. 2. – P. 171–178.

УДК 519.865, 330.16

Д. К. Бойко¹, И. С. Меньшиков^{1,2}

¹Московский физико-технический институт
(государственный университет)

²Вычислительный центр им. А. А. Дородницына ФИЦ ИУ РАН
E-mail: dmitriy-russian@yandex.ru

ЛАБОРАТОРНЫЙ АНАЛИЗ ИГРОВОЙ МОДЕЛИ МНОГОСТОРОННЕГО КОНФЛИКТА

Целью данной работы является исследование конфликта трёх и более лиц на базе простой интуитивно-понятной игровой модели. Проводятся аналитическое и компьютерное исследования данной игры, описано проведение экспериментов в лабораторных условиях, сравниваются результаты лабораторных и виртуальных численных экспериментов. Модель конфликта может найти применение в социальных и политических ситуациях, а также при разработке компьютерных и салонных игр [1–2].

Ключевые слова: дуэль трёх лиц, конфликты, психологические типы, совершенное по подыграм равновесие Нэша, кингмейкинг.

Математическая модель

В результате изучения и анализа современных настольных игр было замечено, что многие из них можно свести к простой математической игровой модели. Например, под её описание подойдут правила «сражений» из игры “Inis”¹ и большинство игр американской школы [1].

Играют N лиц ($N > 2$). Каждому участнику выдаётся некоторое количество «жизней» – L ($L \in \mathbb{N}, L > 0$). Между игроками определяется неизменный порядок хода. В свой ход активный игрок выбирает одного из своих оппонентов (не себя!), который ещё не выбыл из игры, и уменьшает ему количество жизней на единицу. Любая информация о совершенных ходах доступна каждому игроку. Игра заканчивается тогда, когда число игроков с отличным от нуля значением жизней станет равно S (число победителей S ($0 < S < N$) устанавливается до начала игры). Игрок, у

¹ <https://boardgamegeek.com/boardgame/155821/inis>

которого число жизней стало равно нулю, считается выбывшим и пропускает свои ходы до конца игры. Пусть u – выигрыш игрока. Все невыбывшие игроки объявляются победителями ($u = 1$), остальные игроки – проигравшими ($u = 0$).

В данной игре информация совершенная и полная. Из этого следует, что в ней существует совершенное по подыграм равновесие Нэша. Для каждой игры с конкретными заданными параметрами оно находится с помощью обратной индукции [3]. Сложность с нахождением данного равновесия возникает из-за неоднозначности выигрышей. Для данной работы используется *аксиома симметричного равновесия*, которая гласит, что если для игрока несколько вариантов одинаково выигрышны, то он выберет один из них случайным образом с одинаковой вероятностью, будучи абсолютно непредвзятым к своим оппонентам.

Кингмейкинг

Частым явлением в компьютерных и салонных играх, моделирующих войны и конфликты, является кингмейкинг (kingmaking). Суть его в том, что на последних ходах партии отстающий игрок своими действиями определяет, кто из претендентов на победу выиграет, а кто – нет. Некоторые люди испытывают фрустрацию, когда такой момент наступает. Вместе с тем кингмейкинг является неотъемлемой частью игры, поскольку важно предсказать то, как к тебе отнесётся проигрывающий оппонент, будет ли он мстить [1].

Так как мы можем вычислить выигрыши игроков в равновесии, то определим, что *кингмейкер* (в данной игре) – игрок, у которого в начальной позиции игры выигрыш равен нулю при условии, что оппоненты играют рационально.

Очевидно, что в игре, в которой всего три игрока, и у каждого только по одной жизни на старте, кингмейкером является первый игрок, определяющий победителя среди своих оппонентов. На данный момент удалось доказать, что в этой модели игры в случае трёх игроков, придерживающихся аксиомы симметричного равновесия, всегда существует хотя бы один кингмейкер. Доказательство проводится с помощью индукции и разбора всевозможных вариантов ходов. Как следствие, номер очередности хода игрока-кингмейкера определяется по формуле $i = (l_1 + l_2 + l_3) \bmod 3 + 1$, где l_1 , l_2 , l_3 – жизни игроков. Данное следствие применимо только для случаев, когда нет такого игрока, который был бы

абсолютным победителем за счёт того, что у его оппонентов мало жизней, и они ничего не могут сделать, чтобы не дать ему победить.

Компьютерные исследования

В рамках исследования было написано приложение, которое позволяет проанализировать игры с заданными параметрами. Во-первых, приложение даёт возможность найти выигрыши игроков. Таким образом, удалось увидеть выше описанный эффект кингмейкинга. Также стоит отметить, что приложение оптимизировано с помощью методов динамического программирования, а потому позволяет получить результаты за полиномиальное, а не экспоненциальное время. Во-вторых, можно получить все симметрично-равновесные траектории, т.е. такие реализации игры, в которых все игроки действуют рационально и подчиняются аксиоме симметричного равновесия. Как оказалось, что количество таких траекторий очень быстро растёт в зависимости от числа стартовых жизней и числа участников (см. табл. 1).

Т а б л и ц а 1

Количество всех симметрично-равновесных траекторий. Строка сверху – число стартовых жизней (одинаковое для всех игроков), столбец слева – число игроков

	1	2	3	4	5	6
3	2	4	7	51	121	980
4	6	8	56	260	1146	1053
5	24	84	390	1525	26158	1470976
6	120	992	1170	16150	49804	1724168

Игры с одной стартовой жизнью

При изучении графиков выигрышей игроков в зависимости от их номера хода в играх с одной стартовой жизнью было замечено, что картина приближается к синусоиде по мере увеличения числа игроков в игре. Для изучения этого явления была выведена рекуррентная формула выигрыша $u(i, n)$, где n – число игроков, i – номер интересующего игрока:

$$u(1, n) = u(n-1, n-1),$$

$$i \neq 1: u(i, n) = \frac{1}{n-1}((i-2)u(i-2, n-1) + (n-i)u(i-1, n-1)),$$

$$\text{база: } u(1, 2) = 1, u(2, 2) = 0.$$

Очевидно, что вычисление значений работает за полиномиальное время ($O(n^2)$).

В результате наблюдений был обнаружен странный эффект. На рис. 1 изображена волна длиной немного больше одного периода (числа игроков). Данный эффект повторяется, например, на рис. 2. В то же время на других графиках волна имеет целый период (см. рис. 3).

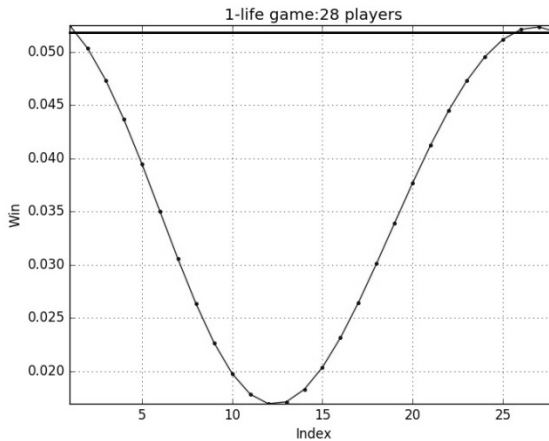


Рис. 1. Зависимость выигрыша от номера игрока. Игра 28 лиц

Горизонтальная черта на рис. 1 указывает на упомянутый факт.

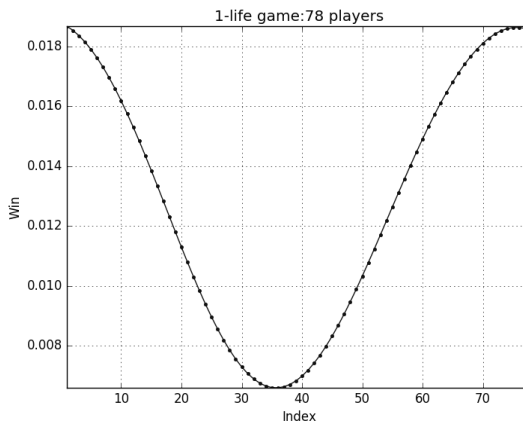


Рис. 2. Зависимость выигрыша от номера игрока. Игра 78 лиц

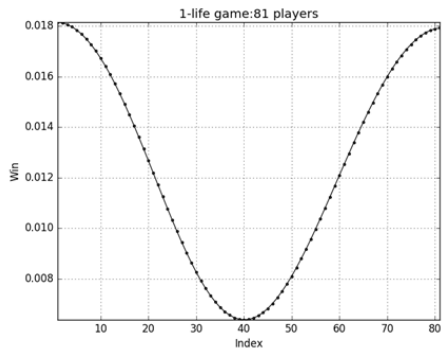


Рис. 3. Зависимость выигрыша от номера игрока. Игра 81 лица.

Замечено, что фаза повторяется всякий раз, когда число игроков становится примерно в 2,71 раз больше (например, табл. 2; игра с 81 игроками есть на рис. 3).

Т а б л и ц а 2

Первые 6 значений, при которых рисунок графиков подобен

Число игроков	Отношение соседних значений
11	2,73
30	2,70
81	2,72
220	2,71
597	2,71
1618	

Возникает проблема с описанием функции выигрыша $u(i,n)$ через $f(t) = a + b \cdot \sin(\alpha + \beta t)$, $t \in [0,n]$, где a, b, α, β зависят только от n . Легко показать, что для $n = 8k$, $k \in \mathbb{N}$ найдутся значения $f(t)$ как рациональные, так и иррациональные, а все значения $u(i,n)$ рациональные.

Коэффициент раскулачивания

Раскулачивание («bashing the leader», дословно «битьё лидера») – явление, когда игроков с большим числом жизней активней выбирают в качестве цели агрессивных действий, чем игроков с меньшим числом жизней [1].

Коэффициент раскулачивания – число, отображающее выбор игрока на конкретном одном ходу траектории игры с точки зрения дифференциации числа жизней у его противников:

$$\phi(s) = \frac{l_s - \min_{i \neq k} l_i}{\max_{i \neq k} l_i - \min_{i \neq k} l_i},$$

где $\phi(s)$ – коэффициент раскулачивания, k – активный игрок, s – выбор игрока k , l_i – количество жизней у не выбывшего еще игрока i . Если знаменатель в формуле равен нулю, то считается, что коэффициент раскулачивания не определён.

Определим *коэффициент раскулачивания игрока* $\langle \phi \rangle$ как среднее арифметическое ϕ для определённого игрока за все ходы.

На графике (см. рис. 4–5) представлены ϕ и $\langle \phi \rangle$ игры 5 участников с одинаковым количеством стартовых жизней, равным 7, продолжающейся до одного выжившего. Игра моделировалась на компьютере, а виртуальные игроки вели себя рационально, соблюдая аксиому симметричного равновесия. На рис. 4 по горизонтальной оси отложены интервалы от 0 до 1 коэффициента раскулачивания, а по вертикальной – процент ходов с таким коэффициентом. Как видно, 24% – это выбор самого отстающего игрока, 46% – самого «живого», а 30% – оставшиеся стратегии.

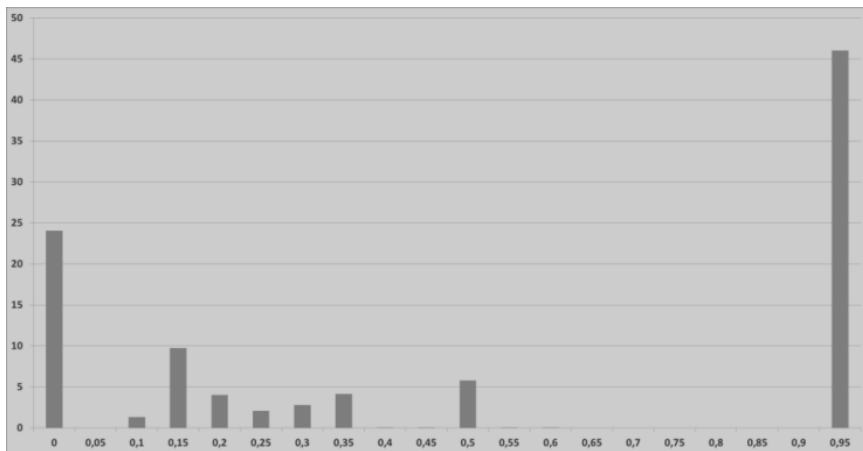


Рис. 4. Распределение коэффициента раскулачивания ϕ

На графике (см. рис. 5) по горизонтальной оси отложены интервалы от 0 до 1 коэффициента раскулачивания игроков, а по вертикальной – процент таких коэффициентов. Стоит заметить, что данный график отражает то, что виртуальные игроки часто смешивают примерно поровну две основные стратегии: выбор самого сильного и выбор самого слабого противника. Среднее значение коэффициента раскулачивания игроков равно 43,6%, а стандартное отклонение – 4,82%.

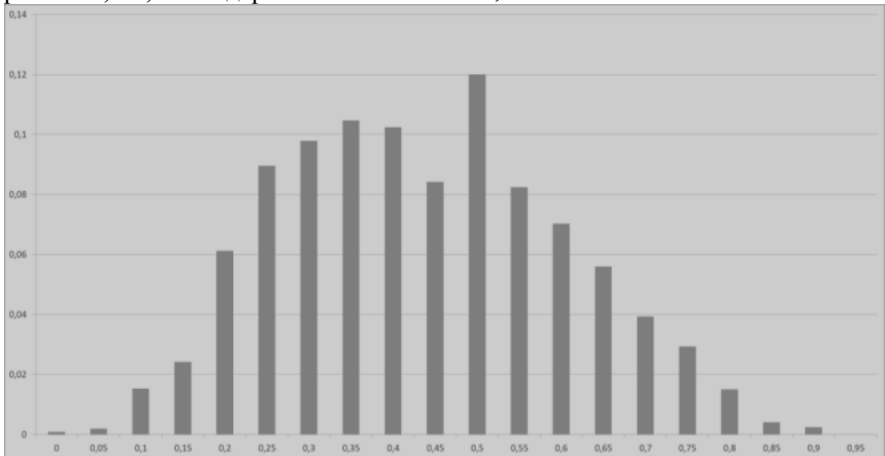


Рис. 5. Распределение коэффициента раскулачивания $\langle \phi \rangle$

Лабораторные эксперименты

В Лаборатории экспериментальной экономики МФТИ был проведён ряд экспериментов, в которых участвовали 11 студентов. Следует отметить, что все участники эксперимента высказали свои мнения по эксперименту и данной модели игры: они разделились примерно пополам: кому-то игра понравилась, кому-то – нет. Нашёлся человек, который посчитал, что «таких игр быть не должно» (данный игрок принимал участие во всех играх и всегда быстро становился выбывшим). Другие заявляли, что «это напоминает игру «Мафия»², а «победа сильно зависит от везения». Было замечено, что некоторые игроки пребывали во фрустрации, а другие были настроены достаточно азартно. Участники, которые приходили на эксперимент два раза (перерыв был неделя), и сначала не понимали, в чём

² [https://ru.wikipedia.org/wiki/Мафия_\(игра\)](https://ru.wikipedia.org/wiki/Мафия_(игра))

смысл такого эксперимента, впоследствии отметили, что для них игра вполне приобрела смысл. Всего было проведено две игры с 11-ю игроками, 8 игр с тремя игроками и 4 игры с 6-ю игроками.

Для проведения экспериментов было создано приложение на платформе Java 8, реализующее игру по сети и использующее библиотеки графического интерфейса Swing. Приложение доступно для скачивания на сайте w17502.vdi.mipt.ru.

Понимание правил игры и того, как пользоваться приложением, у участников не вызывало сомнений.

Все участники до игры проходили психологические тесты: MBTI (Myers-Briggs Type Indicator) и Enneagram (Эннеаграмма) [4–5].

Коэффициент раскулачивания и психологический тип

Данный параметр слабо коррелирует со всеми психологическими параметрами, кроме как с «Перфекционист». Трактовать это можно так, что люди данной категории боятся дисбаланса и отсутствия равновесия. Из проведённых опытов: средний коэффициент раскулачивания игроков равен 51,8%, минимум равен 11,7%, максимум равен 90,6%, стандартное отклонение равно 26,2% (см. табл. 3).

Т а б л и ц а 3

Корреляция коэффициента раскулачивания игроков с их психологическими параметрами

Участник	1	2	3	4	5	6	7	8	9	Корреляция
1. Перфекционист	14	11	19	17	16	14	22	23	23	0,72
2. Помощник	15	11	22	19	22	19	17	9	7	-0,20
3. Мотиватор	19	14	16	13	8	14	13	19	22	-0,05
4. Романтик	14	25	24	20	20	24	23	19	10	-0,29
5. Мыслитель	20	20	11	7	13	11	15	19	20	0,09
6. Лоялист	15	20	17	19	22	26	15	22	9	-0,26
7. Энтузиаст	17	18	11	19	16	8	10	5	22	0,04
8. Лидер	20	9	10	17	9	8	14	16	19	0,18
9. Миротворец	10	16	14	13	18	20	15	12	12	-0,16
Extraversion	19	2	16	17	12	6	2	8	12	-0,05
Introversion	8	26	11	10	15	20	24	23	14	0,12
Sensing	17	7	4	23	23	21	16	28	10	0,24
Intuition	9	20	18	6	2	5	12	4	19	-0,12
Thinking	28	11	7	20	14	10	8	28	27	0,22
Feeling	0	8	15	6	9	11	15	2	2	-0,07
Judging	18	0	18	19	14	25	21	26	12	0,02
Perception	9	29	11	11	17	2	10	2	16	0,09
Коэффициент раскулачивания игрока	0,12	0,3	0,45	0,52	0,91	0,04	0,67	0,85	0,82	

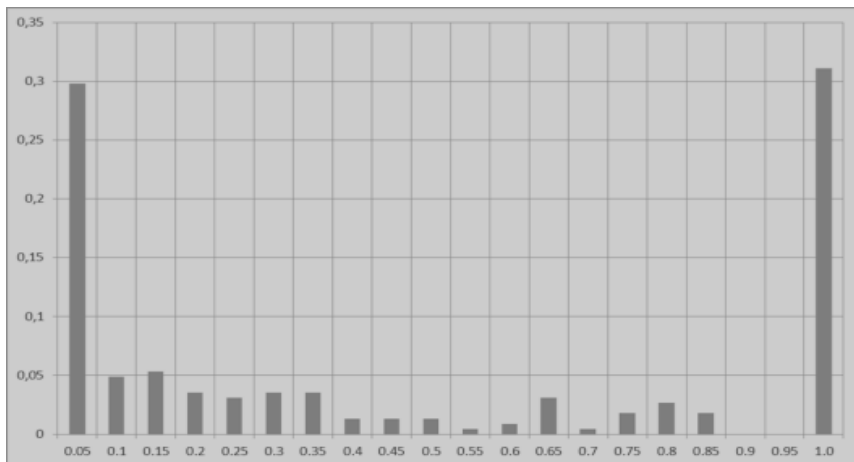


Рис. 6. Распределение коэффициента раскулачивания ϕ на основе экспериментов

На рис. 6 представлено распределение коэффициента раскулачивания на основе экспериментов. По горизонтальной оси отложены интервалы от 0 до 1 коэффициента раскулачивания, а по вертикальной – процент ходов с таким коэффициентом. Заметим, что по сравнению с теоретическими результатами в экспериментах процент «бития» самых сильных и процент «бития» самых слабых почти одинаков. Это может возникать из-за такого социального момента, как поведение агрессивной толпы.

Коэффициент инертности выбора

Инертность выбора показывает, насколько часто игрок меняет свою стратегию, то есть выбирает других игроков, по ходу всей игры.

$$\sigma_i = \sqrt{\frac{\sum_{j=1}^N p_j^i (k_j^i - k^i)^2}{\sum_{j=1}^N p_j^i}},$$

где σ_i – средняя инертность выбора i -го игрока, p_j^i – количество возможностей (число раз) в течение всей игры выбрать i -м игроком j -го игрока, k_j^i – количество выбора i -м игроком j -го игрока в течение всей игры, $\langle k^i \rangle$ – среднее число ходов на одного противника i -го игрока.

Т а б л и ц а 4

Корреляция инертности выбора с психологическими параметрами

Участник	1	2	3	4	5	6	7	8	9	Корреляция
1. Перфекционист	14	11	19	17	16	14	22	23	23	-0,61
2. Помощник	15	11	22	19	22	19	17	9	7	-0,03
3. Мотиватор	19	14	16	13	8	14	13	19	22	-0,30
4. Романтик	14	25	24	20	20	24	23	19	10	0,14
5. Мыслитель	20	20	11	7	13	11	15	19	20	0,22
6. Лоялист	15	20	17	19	22	26	15	22	9	-0,07
7. Энтузиаст	17	18	11	19	16	8	10	5	22	0,63
8. Лидер	20	9	10	17	9	3	14	16	19	-0,41
9. Миротворец	10	16	14	13	18	20	15	12	12	0,32
Extraversion	19	2	16	17	12	6	2	8	12	-0,20
Introversion	8	26	11	10	15	20	24	23	14	0,13
Sensing	17	7	4	23	23	21	16	28	10	-0,56
Intuition	9	20	18	6	2	5	12	4	19	0,50
Thinking	28	11	7	20	14	10	8	28	27	-0,31
Feeling	0	8	15	6	9	11	15	2	2	0,09
Judging	18	0	18	19	14	25	21	26	12	-0,95
Perception	9	29	11	11	17	2	10	2	16	0,94
Инертность выбора	1,17	2,44	1,25	1,09	1,76	1,09	0,98	0,49	1,51	

Инертность тем выше, чем игрок чаще выбирает одного и того же игрока, и тем ниже, когда он чаще меняет цель. Согласно табл. 4, видна сильная обратная корреляция этого параметра с Judging-параметром и сильная корреляция Perception-параметром. Перфекционисты стремятся выбирать разных оппонентов для удара, участники-энтузиасты пытаются остановиться на одной цели.

Другие характеристики

В эксперименте также записывалось время хода игрока. Сильная обратная корреляция видна с таким типом как «Помощник». Также наблюдается средняя корреляция в связи с интроверсией-экстраверсией и thinking-feeling (см. табл. 5).

Т а б л и ц а 5

Корреляция времени хода (время хода в секундах)

Участник	1	2	3	4	5	6	7	8	9	Корреляция
1. Перфекционист	14	11	19	17	16	14	22	23	23	0,52
2. Помощник	15	11	22	19	22	19	17	9	7	-0,77
3. Мотиватор	19	14	16	13	8	14	13	19	22	0,46
4. Романтик	14	25	24	20	20	24	23	19	10	-0,41
5. Мыслитель	20	20	11	7	13	11	15	19	20	0,58
6. Лоялист	15	20	17	19	22	26	15	22	9	-0,07
7. Энтузиаст	17	18	11	19	16	8	10	5	22	-0,22
8. Лидер	20	9	10	17	9	8	14	16	19	0,28
9. Миротворец	10	16	14	13	18	20	15	12	12	-0,11
Extraversion	19	2	16	17	12	6	2	8	12	-0,44
Introversion	8	26	11	10	15	20	24	23	14	0,51
Sensing	17	7	4	23	23	21	16	28	10	0,37
Intuition	9	20	18	6	2	5	12	4	19	-0,12
Thinking	28	11	7	20	14	10	8	28	27	0,51
Feeling	0	8	15	6	9	11	15	2	2	-0,46
Judging	18	0	18	19	14	25	21	26	12	0,17
Perception	9	29	11	11	17	2	10	2	16	-0,21
Среднее время хода	4,82	5,50	3,31	4,19	5,39	6,07	6,04	8,71	7,96	

В ходе игры участник может сделать ход, который будет считаться симметрично-равновесным. Но также он может и выбрать такую траекторию, на которой имеет меньшую вероятность победы. Вполне можно подсчитать, используя теоретические результаты для конкретной игры, сколько таких «ошибок» сделал каждый игрок.

Из табл. 6 видно, что у «мыслителей» (Эннеграмма) и «Thinking» (МВТИ) корреляция отрицательная и равна 72,4% и 51,8% соответственно. «Перфекционисты» тоже совершают мало ошибок, т.к. мы уже знаем, что бить самого сильного – достаточно неплохая стратегия. Из двух таблиц с номером 3 и 6 также можно получить, что количество ошибок отрицательно коррелирует с коэффициентом раскулачивания игрока на уровне 86,6%. А вот «помощникам» и «миротворцам» ошибки вполне присущи – 51,8% и 59,4% соответственно.

Т а б л и ц а 6

Корреляция числа ошибок и психологических типов

Участник	1	2	3	4	5	6	Корреляция
1. Перфекционист	16	14	22	23	23	9	-0,60
2. Помощник	22	19	17	9	7	22	0,52
3. Мотиватор	8	14	13	19	22	20	0,04
4. Романтик	20	24	23	19	10	12	0,29
5. Мыслитель	13	11	15	19	20	9	-0,72
6. Лоялист	22	26	15	22	9	19	0,37
7. Энтузиаст	16	8	10	5	22	19	-0,26
8. Лидер	9	8	14	16	19	16	-0,24
9. Миротворец	18	20	15	12	12	18	0,59
Extraversion	12	6	2	8	12	22	0,03
Introversion	15	20	24	23	14	6	-0,08
Sensing	23	21	16	28	10	8	-0,27
Intuition	2	5	12	4	19	13	0,03
Thinking	14	10	8	28	27	20	-0,52
Feeling	9	11	15	2	2	4	0,41
Judging	14	25	21	26	12	15	0,41
Perception	17	2	10	2	16	15	-0,37
Число ошибок	2	6	5	3	2	6	

Отдельно стоит отметить влияние пола участника на результат. Представители женского пола почти всегда выигрывали игру или же оставались предпоследними. Если говорить об отношении числа ходов существования участника в игре (пока его жизнь была больше нуля) к общей продолжительности игры, то процент нахождения в игре для девушек равен 96,2%, в то время как у представителей мужского пола он равен 63,9% (см. табл. 7).

Т а б л и ц а 7

Взаимосвязь пола участника с его временем нахождения в игре

	Отношение числа ходов до выбывания к полному числу ходов	Среднее для группы
Участники мужского пола	0,62	0,64
	0,32	
	0,78	
	0,82	
	0,43	
Участники женского пола	0,85	0,96
	1,00	
	0,92	
	0,97	

Заключение

В результате аналитических и компьютерных исследований получено первое понимание природы рассматриваемых игр. В частности, установлено, что игроки изначально поставлены в неравные условия, что связано с номером хода. Эксперименты показали, что люди разных психологических типов ведут себя по-разному, и лишь отчасти их поведение согласовано с теоретическим равновесием.

Работа выполнена при финансовой поддержке грантом РФФИ 16-01-00633A.

Литература

1. Woods, S. (2012). Eurogames: the design, culture and play of modern European board games. Jefferson, North Carolina 28640, United States of America: McFarland & Company, Inc., Publishers.
2. Brams, S. J. (1994). Theory of Moves. Cambridge University Press.
3. Меньшиков И. (2010). Лекции по теории игр и экономическому моделированию. М.: ООО «Контакт Плюс».
4. Myers–Briggs Type Indicator.
https://en.wikipedia.org/wiki/Myers%E2%80%93Briggs_Type_Indicator
5. Enneagram of Personality. https://en.wikipedia.org/wiki/Enneagram_of_Personality

Получено 20.04 2017

Д. А. Смаль

Московский физико-технический институт

(государственный университет)

E-mail: darya.maksimova@phystech.edu

ИССЛЕДОВАНИЕ ПАРАДОКСА БРАЕССА МЕТОДАМИ ЭКСПЕРИМЕНТАЛЬНОЙ ЭКОНОМИКИ ДЛЯ ТРАНСПОРТНОЙ СЕТИ С АСИММЕТРИЧНЫМИ ЗАТРАТАМИ*

Целью данной работы является исследование парадокса Браесса для лабораторной транспортной сети с асимметричными затратами на передвижения участников сети. Одной из основных задач исследования было экспериментально показать, что парадокс Браесса не только теоретическое явление, выполняющееся для классической сети Браесса, но и явление, которое можно воспроизвести в лабораторных условиях для сложной транспортной сети.

Ключевые слова: парадокс Браесса, транспортная сеть, экспериментальная экономика, эксперимент.

Введение

Парадокс Браесса был сформулирован немецким математиком в 1968 г. На примере простой модели перегруженной транспортной сети Браесс показал, что при добавлении новой дуги в сеть участники передвижения начинают независимо друг от друга искать лучшие для себя пути перемещения из общей точки отправления в общую точку прибытия для всех пользователей сети. Но тогда при определенных обобщенных ценах дуг пути и числе участников передвижения равновесные затраты на передвижения всех участников возрастут. В этом и заключается парадокс.

Классический пример парадокса Браесса построен на основе простой транспортной сети, представленной на рис. 1.

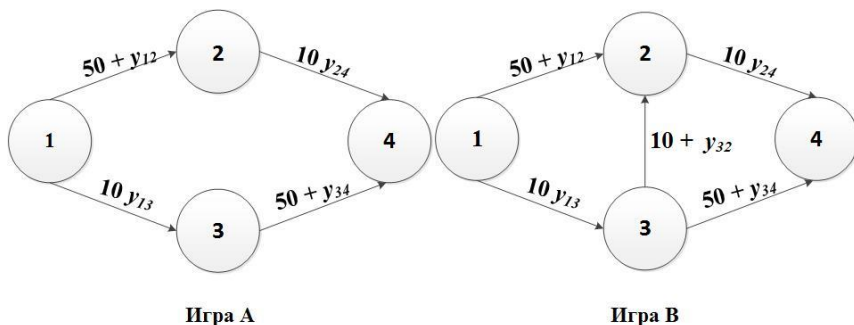


Рис. 1. Классическая транспортная сеть Браесса

Каждый участник имеет возможность выбора одного из двух путей для игры А и одного из трех путей для игры В. Если количество участников передвижения равно 6, то равновесием Нэша для игры А является деление потоков поровну по двум имеющимся путям, при этом время в пути составляет $T = 83$ мин, в игре В равновесное распределение – это деление потоков поровну между тремя путями, при этом время в пути равно $T = 92$ мин. Таким образом, наблюдается парадокс Браесса.

Парадокс Браесса для классической сети рассмотрен во многих работах, в том числе были проведены эксперименты и в Лаборатории экспериментальной экономики МФТИ [1]. Однако многие исследователи считают, что парадокс Браесса – чисто теоретическое явление и не может проявиться в реальной транспортной системе. Данное утверждение основано на том, что классическая транспортная сеть Браесса слишком простая и с симметричными затратами на передвижения, что не соответствует реальной транспортной ситуации.

В связи с описанными выше причинами был проведен эксперимент для топологически более сложной транспортной сети с асимметричными затратами.

Парадокс Браесса для сложной транспортной сети

Для приближения исследования к более реальной транспортной ситуации в данной работе рассмотрена топологически более сложная транспортная система по сравнению с классической транспортной сетью парадокса Браесса. За основу при построении данной транспортной сети была взята сеть из работы [2]. Как уже было сказано ранее, основными слабыми сторонами классической сети являются малое количество маршрутов

движения, а также симметричные затраты на передвижение по дугам, что совершенно не соответствует реальной ситуации на дорогах городов. В связи с этим именно такие моменты и были учтены при задании более сложной транспортной сети. Транспортная сеть, на которой было проведено исследование парадокса Браесса в данной работе, представлена на рис. 2. В дальнейшем будем называть ее сложной транспортной сетью или просто сложной сетью.

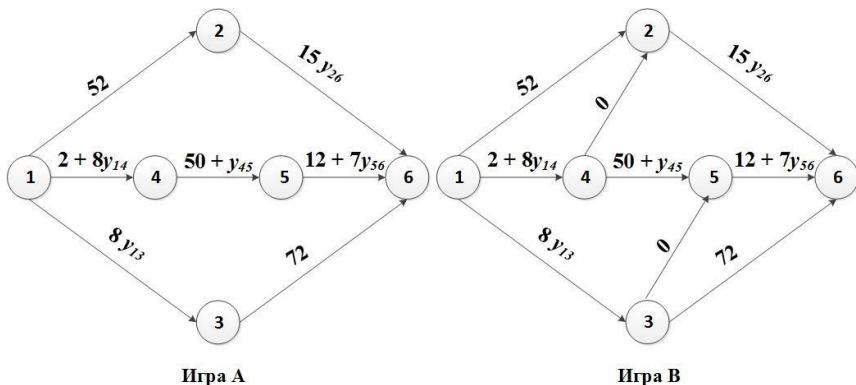


Рис. 2. Сложная транспортная сеть

Как и в классическом примере парадокса Браесса, рассматриваются два случая: первый случай (игра А) – до строительства дополнительной дороги или моста, второй случай (игра В) – после. В игре А для сложной транспортной сети участникам передвижения необходимо выбрать один из трех предложенных маршрутов передвижения для того, чтобы добраться из общей точки отправления в общую точку прибытия. В игре В было построено два новых моста, которые добавили две дополнительные дороги для передвижения. Таким образом, участникам необходимо выбрать из пяти возможных маршрутов. Причем, что очень интересно и важно, стоимость передвижения по вновь построенным мостам является нулевой, т.е. при передвижении по мосту участники не потратят ни одной условной единицы затрат. Для сравнения в классической сети участники выбирают один из двух путей в игре А и один из трех путей в игре В.

Рассмотрим равновесие Нэша для сложной сети и покажем, что в рассматриваемой сложной сети наблюдается парадокс Браесса. Пусть число участников передвижения равно 12. В таком случае равновесием Нэша

для игры А является следующее деление участников по маршрутам передвижения:

$$x_1 = x_{126} = 4,$$

$$x_2 = x_{1456} = 3,$$

$$x_3 = x_{136} = 5,$$

где x_k – число участников, выбравших k -й путь.

При этом полное равновесное время передвижения из точки отправления в точку прибытия $T = 112$ мин.

Для игры В равновесием Нэша является следующее деление участников по маршрутам передвижения:

$$x_1 = x_{126} = 0,$$

$$x_2 = x_{1456} = 0,$$

$$x_3 = x_{136} = 0,$$

$$x_4 = x_{1426} = 5,$$

$$x_5 = x_{1356} = 7,$$

где x_k – число участников, выбравших k -й путь.

При этом полное равновесное время передвижения из точки отправления в точку прибытия $T = 117$ мин.

В игре В участники полностью отказываются от маршрутов без мостов в пользу вновь построенных дополнительных путей, но при этом увеличивают время передвижения при равновесии. Таким образом, наблюдается эффект увеличения затрат на передвижение при добавлении дополнительных ребер в транспортный граф. Участники начинают искать лучшие для себя пути передвижения, тем самым ухудшая положение в целом для всех участников, т.е. в рассматриваемой сложной транспортной сети возникает парадокс Браесса.

Эксперимент и его результаты

Для того чтобы проверить на практике выполнение парадокса Браесса для сложной транспортной сети, в Лаборатории экспериментальной экономики МФТИ были проведены эксперименты, результаты которых и представлены в данной работе. Для проведения эксперимента в лаборатории была создана программа, использующая оболочку z-Tree (университет Цюриха, Швейцария, [3]), которая воспроизводила два описанных вида сложной транспортной сети. В эксперименте участвовали 12 человек, которые ранее не были знакомы с парадоксом Браесса. Участники выби-

рали один из предложенных маршрутов сначала в игре А, затем в игре В. В первой части эксперимента участникам была показана транспортная сеть игры А, транспортная сеть игры В была показана только после завершения первого этапа эксперимента, т.е. после завершения игры А. Во время эксперимента проведено по три сессии для каждой игры, и каждая сессия состояла из 40 периодов. Участники выбирали путь индивидуально без возможности обсудить свой выбор. Цель игры для участников – достигнуть пункта назначения, затратив при этом как можно меньше времени.

Результат 1

На основе данных, полученных по итогам эксперимента, были рассчитаны несколько показателей, которые представлены в данной работе. Первым показателем является частота выбора пути, т.е. количество раз, которое каждый участник выбирал тот и иной путь за всю игру, усредненное по участникам. В табл. 1 приведены значения частоты выбора каждого пути для игры А и игры В.

Таблица 1

Частота выбора пути					
	1-2-6	1-4-5-6	1-3-6	1-4-2-6	1-3-5-6
Игра А	13,42	11,42	15,17	-	-
Игра В	1,28	0,58	0,89	15,56	21,69

Данные результаты показывают, что участники передвижения практически равнозначно выбирали пути в игре А, не выделяя какой-то определенный маршрут. Но в игре В наблюдается явное предпочтение в выборе дополнительных маршрутов, которые и являются равновесными в данном случае. Неравновесные маршруты, можно сказать, игнорировали участники эксперимента.

Результат 2

Еще одним показателем, который был проанализирован в данном исследовании, является среднее число участников. Среднее число участников – это усредненное по периодам число участников, выбравших тот или иной путь. В табл. 2 представлено среднее число участников, выбравших каждый из возможных маршрутов для игры А и игры В, также показано равновесие Нэша для каждой из игр, которое было рассмотрено ранее.

Таблица 2

Среднее число участников					
	1-2-6	1-4-5-6	1-3-6	1-4-2-6	1-3-5-6
Игра А	4,0	3,4	4,6	-	-
Равновесие	4	3	5	-	-
Игра В	0,4	0,2	0,3	4,7	6,5
Равновесие	0	0	0	5	7

Если мы посмотрим на среднее число участников, то увидим сходимость к равновесию как в игре А, так и в игре В, что экспериментально показывает выполнение парадокса Браесса. В игре В участники эксперимента предпочитали для своего передвижения пути с бесплатными для проезда мостами, считая, что данные маршруты являются более выгодными для них, однако же, наоборот, ухудшали общее положение всех участников передвижения.

Несмотря на действительно хорошие результаты эксперимента, представленные в табл. 1 и 2, которые явно показывают тот факт, что эксперимент Браесса не только теоретическое явление, но и выполняется в лабораторных условиях, некоторые исследователи могут обратить внимание на то, что такие показатели, как частота выбора пути и среднее число участников, являются агрегированными, поэтому не дают абсолютно ясной картины эксперимента.

В связи с этим в работе были проанализированы не только усредненные по периодам данные, но и рассмотрен каждый период игры в отдельности. На рис. 3 представлен график, показывающий число участников передвижения, выбравших каждый из возможных путей во всех периодах игры В.

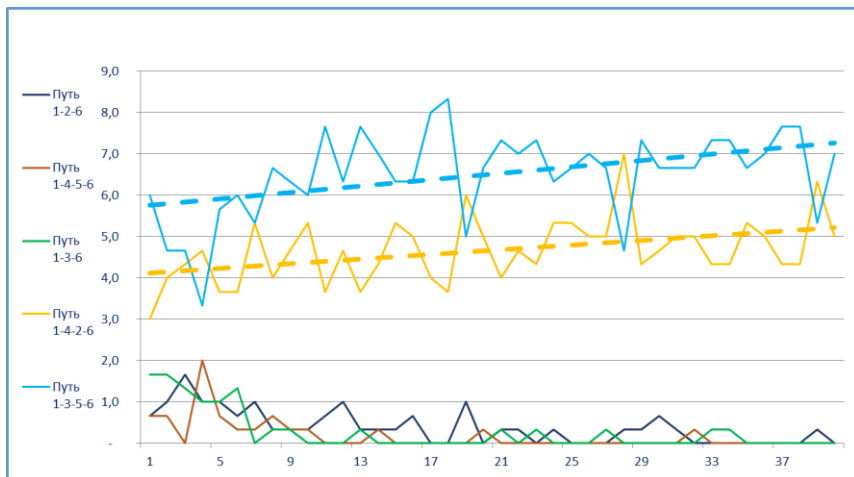


Рис. 3. Среднее число участников по периодам

Как видно из рис. 3, график только подтвердил результаты, полученные при рассмотрении усредненных показателей. Напомним, что во время эксперимента игроки сначала выбирали пути для передвижения в транспортной сети, состоящей из трех маршрутов, и не знали о транспортной сети с двумя дополнительными маршрутами, и только после окончания игры А участникам эксперимента рассказали об игре В с бесплатными мостами для передвижения. По данной причине в первых пяти-семи периодах игры участники продолжали выбирать неравновесные маршруты, но уже после седьмого периода игры участники практически полностью игнорировали пути без мостов и отдавали предпочтение в пользу равновесных маршрутов. При этом из графика видно, что к концу игры распределение участников по маршрутам передвижения практически сходится к равновесию Нэша, а именно 5 участников выбирают путь 1–4–2–6 и 7 участников выбирают путь 1–3–5–6. Если провести эксперимент с большим количеством сессий и большим количеством периодов в каждой сессии, то можно будет наблюдать более явную сходимость к равновесию Нэша к концу игры.

Результат 3

Для того чтобы более наглядно показать сходимость к равновесию Нэша в игре В и тем самым выполнение парадокса Браесса на практике, а

не только в теории, в данной работе был проведен один дополнительный анализ данных, полученных по итогам проведенных экспериментов. Ранее было показано, что в основном игроки при выборе маршрута отдают свое предпочтение в пользу равновесных маршрутов. Однако тогда возникают вопросы: какое количество участников все-таки выбирает неравновесные маршруты, при этом выбываясь из равновесия, и насколько быстро происходит сходимость к равновесию в игре? Ответ на эти вопросы можно увидеть на диаграмме, представленной на рис. 4.

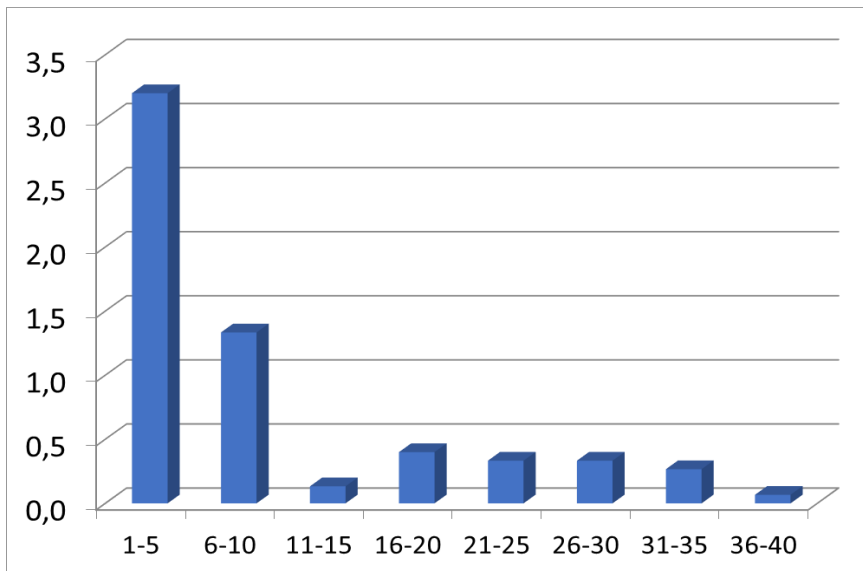


Рис. 4. Среднее число участников, выбравших неравновесные пути, в разрезе периодов

На диаграмме показано среднее число участников эксперимента, выбравших неравновесные пути для передвижения в игре В в разбивке по периодам игры. Из диаграммы видно, что в первых пяти периодах неравновесные пути выбрали 3–4 участника, в следующие пять периодов игры – 1 участник, а после неравновесные пути, можно сказать, не выбирались. Таким образом, было получено еще одно подтверждение выполнения парадокса Браесса в лабораторных условиях для сложной сети.

По итогам эксперимента можно сделать следующие выводы. Парадокс Браесса – это не только теоретическое явление, парадокс можно вос-

произвести в лабораторных условиях для сложной транспортной системы. Для того чтобы наиболее четко наблюдать проявления парадокса Браесса на практике, необходимо проводить эксперименты с большим количеством периодов и участников игры, объединенных в различные группы для разных сессий эксперимента.

Литература

1. *Гасникова Е.В.* Моделирование динамики макросистем на основе концепции равновесия // дисс. к.ф.-м.н. – М.: МФТИ, 2012.
2. *Rapoport A., Kugler T., Dugar S., Gisches E. J.* Braess paradox in the laboratory: experimental study of route choice in traffic networks with asymmetric costs // Decision modeling and behavior in complex and uncertain environments. – 2005. – V. 21. – P. 309–337.
3. *Fischbacher U.* z-Tree: Zurich toolbox for ready-made economic experiments // Experimental Economics. – 2007. – V. 10. – Is. 2. – P. 171–178.

Получено 15.04.2017 г.

УДК 004.056

Б. В. Багно, Д. А. Подлесных

Московский физико-технический институт

(государственный университет)

E-mail: bogdan.bagno@phystech.edu

МОДЕЛЬ АВТОМАТИЧЕСКОЙ СИСТЕМЫ ПРОВЕРКИ ЛАБОРАТОРНЫХ РАБОТ ПО КУРСУ БАЗ ДАННЫХ

Целью данной работы является создание модуля для проверяющей системы Ejudge, позволяющего автоматически проверять и учитывать знания студентов по курсам баз данных и информационной безопасности.

Ключевые слова: базы данных, Ejudge.

Цель работы

В рамках курсов, связанных с базами данных, на хакатонах и на соревнованиях по информационной безопасности «Capture the flag» (далее CTF) требуется проверить умение студентов работать с системами управления базами данных (далее СУБД).

Целью данной работы является создание модуля для проверяющей системы Ejudge, позволяющего объективно и быстро оценивать знания и умения студентов, при этом предоставляющего гибкость в выборе используемых инструментов для работы с базами данных, включая возможность выбрать предпочитаемую СУБД из предоставляемых вариантов.

Возможности проверяющей системы

Проверяющей системе необходимо обладать гибким набором методов проверки заданий, включая как тесты, так и автоматическую проверку SQL-запросов и даже крупных лабораторных работ, например:

- сравнение результата выполненного студентом запроса с эталоном («флагом» в терминах CTF);
- сравнение результата автоматического выполнения контрольного запроса с эталонным после приведения обучающимся базы данных в требуемое в задании состояние;
- сравнение структуры и содержимого эталонной и тестовой (после выполнения лабораторной работы) баз данных.

Совокупность перечисленных стратегий позволяет использовать различные задания, обеспечивая при этом объективность и формальность требований к проходящим курс или сдающим экзамен.

Этапы реализации

Первый прототип такой системы был использован в МФТИ в апреле 2017 года на хакатоне компании «Сбербанк-Технологии», в котором требовалось разработать базу данных с требуемыми характеристиками. Для этого участникам были созданы контейнеры в облаке кафедры информатики и вычислительной математики на базе Parallels Cloud Server с СУБД по желанию заказчика. Сдача архива с решениями проводилась через проверяющую систему Ejudge.

Следующим этапом будут задачи для вступительных экзаменов в магистратуру по специальности «прикладные математика и информатика». У каждого абитуриента будет доступ к своему контейнеру с базой данных, в которой несложным SQL-запросом будет находиться ответ на его задачу – «флаг», подобно варианту соревнований CTF, основанных на задачах. Проверка также будет проводиться с помощью Ejudge.

Третий этап – задачи для Летней олимпиадной школы. При сдаче задания сервер по RSA – ключу будет заходить по ssh на контейнер школьника и SQL-запросом будет проверять соответствие состояния базы данных нужному.

В дальнейшем система будет использоваться на курсе по выбору «Информационная безопасность», а после наработки достаточного количества задач возможно её применение в курсах баз данных ФПМИ и ФРКТ.

Получено 20.04.2017

УДК 004.4'275

А. А. Шумилин

Московский физико-технический институт

(государственный университет)

E-mail: aleksandr.shumilin@phystech.edu

ПРОГРАММНОЕ СРЕДСТВО ДЛЯ ДВУХСТАДИЙНОЙ ВИЗУАЛИЗАЦИИ РЕЗУЛЬТАТОВ ЧИСЛЕННЫХ ЭКСПЕРИМЕНТОВ

Целью данной работы является разработка вспомогательного программного средства, позволяющего упростить процесс подготовки визуальных материалов на основе результатов численных экспериментов. Исследован предоставляемый ПО ParaView программный интерфейс и выполнена реализация простого инструмента, активирующего запись действий пользователя и воспроизводящего их на удалённом сервере с более детальными входными данными.

Ключевые слова: визуализация, ParaView, SSH.

Постановка проблемы

Получаемые в рамках численных экспериментов данные обычно привязаны к трёхмерной расчётной сетке и не могут быть тривиально визуализированы на плоскости экрана или бумаги. Для отображения их в целях анализа и, быть может, последующей демонстрации на семинарах и конференциях возникает необходимость строить различные разрезы, показывать с разных ракурсов, а то и записывать видео «облёта» расчётного объёма со всех сторон. Дополнительное усложнение появляется, когда приходится иметь дело с четвёртым измерением — временем, т. е. показывать изменения расчётного объёма во времени, а также с тем, что каждому узлу расчётной сетки соответствует множество значений параметров, например, три компоненты скорости и три компоненты смещения от начального положения, кроме того, может быть выбрано отображение функции от этих переменных, например, абсолютной величины скорости или давления. И хотя уже многие годы существует программное обеспечение, позволяющее выполнять все вышеперечисленные и многие другие операции, строя при этом изображение на лету, время отклика интерфейса

получается неприемлемым для интерактивной работы в силу большого количества узлов сетки — десятки, а то и сотни миллионов точек.

Описание решения

В связи с вышеописанной проблемой, возникает следующее соображение: сначала из полноценных дампов расчётных сеток построить прорежённые версии с значительно меньшим количеством узлов, открыть в программе визуализации эти прорежённые версии и записать все действия пользователя по их отрисовке (это будет первая стадия), а затем, после того как продуман и одобрен пользователем сценарий отображения, в пакетном режиме (возможно, вообще на удалённом сервере, а не на сравнительно слабом ПК пользователя) воспроизвести все эти действия уже на полноценных дампах сетки. В связи с тем, что правильное (с максимальным сохранением информации о неоднородностях) уменьшение плотности сетки дампа весьма специфично для модели, а также из соображений повторного использования уже готового кода для работы с расчётной сеткой и записью её в файл, прореживание выполняется в рамках ПО для численного моделирования, разрабатываемого на кафедре. Отдельное программное средство, созданное в рамках данной работы, занимается тем, что запускает ПО визуализации в режиме записи действий пользователя, затем извлекает из его памяти через предоставляемый программный интерфейс (API) список действий, находит в нём вызовы функций загрузки входных файлов, запрашивает у пользователя путь к каталогу с полными версиями сеток, проверяет, все искомые файлы что там удалось? Запрашивает пути к тем, что не были найдены, а затем запускает фоновую отрисовку всех записанных на первой стадии действий на локальном ПК или на выбранном сервере, подключаясь к нему по стандартному протоколу SSH с помощью библиотеки Paramiko [1]. В настоящее время реализована поддержка наиболее часто используемого на кафедре ПО визуализации — ParaView [2] через соответствующий API [3] для языка Python, однако при необходимости средство может быть доработано и для другого ПО визуализации, например, MayaVi2 и VisIt.

Литература

1. <https://github.com/paramiko/paramiko>
2. <https://www.paraview.org/>
3. <https://www.paraview.org/ParaView3/Doc/Nightly/www/py-doc/>

Получено 5.06.2017

УДК 004.622

Д. Ю. Дербышев

Московский физико-технический институт

(государственный университет)

E-mail: kirikusan@gmail.com

УСКОРЕНИЕ СБОРА СТАТИСТИКИ С ВИРТУАЛЬНЫХ МАШИН ВНУТРИ ПАКЕТА LIBVIRT

Целью данной работы является ускорение сбора статистики с виртуальных машин, осуществляемого через вызов `virshdomstats` пакета `libvirt`.

Ключевые слова: виртуальные машины, сбор статистики.

Описание работы

Сбор статистики с виртуальных машин необходим для эффективного управления принадлежащей им памяти. Ускорение сбора данной статистики и уменьшение нагрузки, создаваемой в процессе, уменьшат нагруженность и повысят производительность физической системы. Было произведено два улучшения.

Для получения статистики от всех виртуальных машин по различным системам (полный перечень которых содержится в мануале команды `virsh domstats`) было необходимо сделать один вызов `virConnectGetAllDomainStats`, возвращающий статистику по всем запрашиваемым статистикам всех виртуальных машин, и еще по одному вызову `virDomainMemoryStats` за каждую виртуальную машину для получения статистики потребления памяти. Серия изменений позволила изменить код `libvirt` в [1] так, что теперь достаточно только вызова `virConnectGetAllDomainStats`.

Основной цикл работы `libvirt` в функции `virEventRunDefaultImpl` содержит циклический запуск имплементации `virEventPollRunOnce`, ожидающей и обрабатывающей события. Реализация данной функции была написана через системный вызов `poll`, что требовало каждый раз аллокации памяти для массива, линейного поиска всех необходимых дескрипторов для ожидания. Внутри ядра операционной системы это также приводило к дополнительному копированию массива дескрипторов. Исходя из предположения, что изменения списка этих дескрипторов (происходящие,

чаще всего, при старте и остановке виртуальных машин), являются более редкими событиями, чем обращения к этим дескрипторам, в [2] предложена имплементация `virEventPollRunOnce` с использованием системного вызова `epoll`. За счет одного системного вызова при любом изменении списка ожидаемых файловых дескрипторов становится возможным избежать любых аллокаций памяти в основном цикле работы `libvirt`, а также делает скорость его работы не зависящей от количества запущенных виртуальных машин.

Результаты

Достигнуто ускорение сбора статистики с виртуальных машин, осуществляемого через вызов `virshdomstats` пакета `libvirt`. Для этого было предоставлено две серии патчей, улучшающих производительность. На машине с 35-ю виртуальными машинами при сборе статистики раз в 5 секунд, достигнуто ускорение работы `libvirt` на 20 процентов.

Литература

1. *Derbyshev D.* `libvirt: qemu: return balloon statistics when all domain statistics reported`
<https://libvirt.org/git/?p=libvirt.git;a=commit;h=438c204763f7d8eed79554075f5633545a4a5df1>
2. *Derbyshev D.* `libvirt: vireventpoll implimentation using epoll`
<http://www.spinics.net/linux/fedora/libvir/msg143288.html>

Получено 31.05.2017

УДК 681.3.06

К. А. Щербатов

Московский физико-технический институт

(государственный университет)

E-mail: kshcherbatov@gmail.com

AWS API-СОВМЕСТИМОЕ ШИФРОВАНИЕ ДЛЯ OBJECT STORAGE SDS-РЕШЕНИЯ

Целью данной работы является исследование проблем, связанных с обеспечением безопасности пользовательских данных в объектном хранилище, разработка и внедрение таких механизмов.

Ключевые слова: объектное хранилище, криптография, SDS, S3, SSE.

Первичная обработка заголовков RESTful-API запроса

Server-side encryption with customer-provided keys (SSE-C) в объектном хранилище предполагает отправку дополнительных заголовков S3-запросов.

```
GET /example-object HTTP/1.1
Host: example-bucket.s3.amazonaws.com
Accept: */*
Authorization: authorization string
Date: Wed, 28 May 2014 19:24:44 +0000
x-amz-server-side-encryption-customer-key:
g0lCfA3Dv40jZz5SQJ1ZukLRFqtI5WorC/8SEKEXAMPLE
x-amz-server-side-encryption-customer-key-MD5: ZjQrne1X/iTcskBY2m3example
```

Рис. 1. Запрос GET, использующий SSE-C заголовки

Принятый в OStorage gateway S3-запрос проверяется перед обработкой. Пресекаются попытки общения не по TLS: ключи, передаваемые таким образом могут быть перехвачены (Eva). При промышленной установке хранилище использует только HTTPS; nginx будет отклонять HTTP-обращения, и запросы не попадут в OStorage gateway; в противном случае небезопасное соединение определим по URI и отклоним.

Далее проверяется MD5 ключа (представленного base64) и осуществляется работа с метаданными.

Сначала вычисляется HMAC для ключа и для запросов на чтение сверяется с хранимым. При вычислении используется SHA256. В случае несовпадения значений запрос отклоняется до попыток расшифровать информацию.

Заметим, ничто не мешает пользователю использовать один ключ для разных объектов или для разных версий одного объекта. Система, использующая его непосредственно, была бы уязвимой. Поэтому на смену предоставленному значению следует создать новое: пару ключ и вектор инициализации, – уникальное для каждой сущности; а для вычисления HMAC – уникальную соль. Соответственно все данные будут всегда представлены разным набором байт, а HMAC одинаковых ключей в метаданных не совпадут.

Таким образом, хранимые с объектом метаданные для пользовательского eKey-значения выглядят так:

$ivSalt, eKeySalt, eKeyHMAC = HMAC(sha256, eKey, eKeySalt)$

Шифрование осуществляется с помощью

$(iv, key) = EVP_BytesToKey(sha256, eKey, ivSalt)$

Шифрование объекта

Блочное шифрование AES256 может осуществляться в разных режимах. В силу специфики объектного хранилища, объекты в котором создаются, но не изменяются, выбран режим сцепления блоков шифр текста (CBC), не обладающий недостатками сохранения статистических особенностей открытого текста режима простой замены.

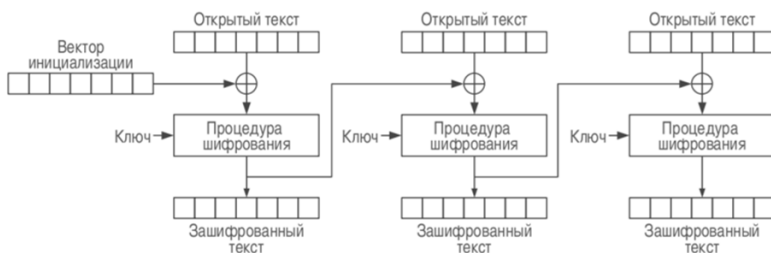


Рис. 2. CBC режим шифрования

S3-протокол поддерживает ranged requests, позволяющие считать часть объекта. Отметим, блоки зависимы: шифр-блок используется как вектор инициализации для следующего. Если бы объект был редактируе-

мым, все байты, следующие за местом изменений, менялись. В случае ranged read достаточно выравнять запросы по границам шифр-блоков.

Криптографические операции осуществляются в составе модуля формирования сообщений для SDS-хранилища. Сначала поступает анонс операции. В его состав включены начальный вектор инициализации и ключ, сгенерированные ранее. При необходимости запрос выравняется.



Рис. 3. Выравнивание запросов на чтение

Затем принимается серия сообщений по fastcgi. Интерфейсы OpenSSL выдают зашифрованные блоки, из которых формируется msg', уходящее на запись.

Результаты

Целью данной работы была разработка функционирующего прототипа, осуществляющего SSE-C логику в составе Object Storage и безопасно хранящего sensitive metadata. Полученный прототип был протестирован адаптированными fuzzing-тестами Ceph «S3 compatibility tests».

Литература

1. Amazon Web Services, «AWS Key Management Service Cryptographic Details», Amazon Web Services, Inc., August 2016.
2. Singh K. Ceph Cookbook, Birmingham, UK: Packt Publishing Ltd, 2016.
3. Denis St. T., Johnson S. Cryptography for Developers, Rockland, Massachusetts, US: Syngress Publishing, 2006.
4. Таненбаум Э., Бос Х. Современные операционные системы. 4-е издание, СПб: Питер, 2015.

Получено 30.04.2017

УДК 517.9

Н. Н. Ефанов, П. В. Емельянов

Московский физико-технический институт

(государственный университет)

E-mail: nefanov90@gmail.com

ПОСТРОЕНИЕ ФОРМАЛЬНОЙ ГРАММАТИКИ СИСТЕМНЫХ ВЫЗОВОВ

Целью данной работы является формализация задачи восстановления последовательностей системных вызовов, приводящих к некоторой целевой конфигурации дерева процессов Linux. Исследована комбинаторная сложность задачи. Ввиду оценок количества генерируемых деревьев предложено решение задачи на основании анализа записи дерева процессов как строки, порождаемой в некоторой формальной грамматике, которая также построена и исследована. По результатам исследований рассмотрен вариант реализации анализатора.

Ключевые слова: математическая модель, операционные системы, системные вызовы, формальные грамматики, дерево разбора, промежуточное представление

Задача о восстановлении дерева процессов

Задача о восстановлении дерева процессов вместе с порождающими цепочками системных вызовов возникает в различных компьютерных проектах, ориентированных на поддержку миграции виртуальных сред и процессов [1–2]. На текущий момент задача не имеет полноценного решения, что обуславливает актуальность проведения данной работы.

Сформулируем задачу: получая на вход снимок состояния дерева процессов в некоторый момент времени (к примеру, посредством утилиты ps), получить на выходе последовательности системных вызовов, выполнив которые можно получить входящее дерево из некоторого (предварительно заданного) начального состояния. На момент подготовки публикации были использованы следующие ограничения на задачу.

1. Для воссоздания дерева процессов используются исключительно цепочки запускаемых из пространства пользователя системные вызовы.
2. На вход подаётся корректное дерево, то есть существует цепочка системных вызовов, приводящая к нему из начального состояния. Если такая цепочка не найдена, дерево считается некорректным.
3. Используются системные вызовы:

- a) `fork()`, создающий процесс-потомок,
 - b) `setsid()`, создающий новую сессию с лидером-вызвавшим процессом
 - c) `setpgid()`, назначающий процессу группу.
- В решении считается, что данный системный вызов замкнут на себя, то есть вызывается только от имени процесса, изменяющего группу.
- d) `exit()`, завершающий вызвавший процесс и переназначающий INIT-процессу потомков [1–2].

Данные системные вызовы реализуют минимальное количество операций, необходимых для всевозможных преобразований над деревом процессов.

Оценки из перечислительной комбинаторики

Оценим количество различных конфигураций дерева процессов в зависимости от иерархий, сессий и групп процессов. Для этого рассчитаем количество различных деревьев процессов, полученных из INIT – процесса [1] посредством вызовов `fork()`, `setsid()`, `setpgid()`. Представим процесс-родитель узлом дерева с тройкой значений (идентификатор процесса, группы, сессии). Некоторый процесс-потомок данного процесса запишем после “ $\Rightarrow$ ”. При этом не будем предъявлять жесткие требования к идентификатору процесса, кроме указанных выше. Это означает, что деревья $(1\ 1\ 1) \Rightarrow (2\ 1\ 1)$ и $(1\ 1\ 1) \Rightarrow (5\ 1\ 1)$ являются неразличимыми.

1. С одним лишь `fork`. Зафиксируем количество процессов и установим, сколько различных деревьев можно получить на этих процессах, полагая системный вызов `fork` единственным способом порождения процесса-потомка.

Утверждение 1. Количество различных деревьев из n процессов, полученных посредством системного вызова `fork`, равняется количеству остовных деревьев полного графа с n вершинами.

Доказательство. Пронумеруем каждую вершину соответствующим ей идентификатором процесса (PID). Для получения одного дерева достаточно соединить все вершины так, чтобы в полученном графе не было циклов. Любое дерево, полученное таким соединением, будет остовным деревом исходного графа по определению. Что и требовалось доказать.

Для вычисления количества различных остовных деревьев можно применить формулу Келли:

$$N = n^{n-2}, \text{ где } n - \text{число вершин.}$$

Утверждение 2. Количество различных деревьев-процессов со свободным диапазоном идентификаторов $2 \dots 2^n - 1$ и выделенными идентификаторами³ 0 и 1 равняется

$$F(n) = \sum_{i=2}^{n-1} i^{i-2}.$$

Доказательство методом математической индукции.

1) $n = 2$, тогда, $F(2) = 2^0$, что соответствует единственной конфигурации с процессами 0 и 1.

$$2) \quad n = k - 1, \text{ тогда } F(k - 1) = \sum_{i=2}^{k-2} i^{i-2}.$$

3) $n = k$. Для меньшего числа процессов все различные конфигурации уже перечислены на предыдущем шаге. Осталось перечислить все возможные конфигурации из k -процессов, что можно сделать, согласно утверждению 1, по формуле

$$4) \quad F(k) = \sum_{i=2}^{k-2} i^{i-2} + (k-1)^{(k-1)-2} = \sum_{i=2}^{k-1} i^{i-2},$$

что и требовалось доказать.

2. С различимостью сессий – setsid. Системный вызов setsid, вызываемый для создания новой сессии, устанавливает вызвавшему процессу идентификатор сессии и группы, равные идентификатору процесса, – задаёт процесс лидером созданной сессии и группы. При этом вызвавший процесс не должен быть лидером группы. Следовательно, setsid можно корректно применить только один раз для каждого из процессов.

³ Идентификаторы 0 и 1 используются для процессов SCHED и INIT соответственно.

SCHED – процесс-планировщик, не создающий потомков, INIT – процесс-инициализатор, являющийся общим прародителем процессов в системе: любой процесс получается цепочкой системных вызовов из INIT.

Утверждение 3. Учёт в формуле (2) различимости по сессиям даёт количество различных деревьев процессов, равное

$$F(n) = \sum_{i=2}^{n-1} i^{i-2} \cdot 2^{i-1} \cdot \sum_{j=0}^{2^l-1} 2^{(\mathbf{f}(j) \cdot \mathbf{k})},$$

где $\mathbf{f}(j)$ – вектор-функция размерности l -количества уровней дерева, возвращающая j -й бинарный вектор в порядке инкрементального перечисления, $\mathbf{k}$ – вектор, составленный из суммарного количества потомков на уровнях $1 \dots 1$ соответственно.

Доказательство. Каждый процесс, не являющийся лидером группы, может вызвать `setsid()`, причем только один раз. Порядок вызова для процесса с потомками также имеет значение: потомок может быть порождён как до образования новой сессии, так и после, то есть существует $(1 + 2^{k_{children}})$ – вариантов для потомков процесса, причем каждый из них также может также сделать `setsid()`, добавляя $2^{k_{children}}$ вариантов для каждой из конфигураций. Проводя обход в ширину из корня дерева, считающийся нулевым уровнем глубины, получим выражение для поправки:

$(1 + 2^{k_1}) \cdot 2^{k_1} \cdot \dots \cdot (1 + 2^{k_l}) \cdot 2^{k_l}$, раскрывая скобки в котором, а также учитывая, что $k_1 + \dots + k_l = i - 1$ для i процессов, получаем выражение $2^{i-1} \cdot (1 + 2^{k_1} + \dots + 2^{k_1 k_2} + \dots + 2^{k_1 k_l} + \dots + 2^{k_1 \dots k_l})$, что эквивалентно поправке из утверждения.

3. С различимостью групп – `setpgid()`. Данный системный вызов назначает процессу, не являющемуся лидером, новую группу, если группа существует в одной сессии с процессом либо создает новую группу с указанным процессом-лидером. Предполагается, что процесс назначает группу сам себе для упрощения анализа. Учет групп вносит поправку, аналогичную виду поправки для сессий, за исключением того, что в первом сомножителе степень имеет вид $i - 1 - LC_{sg}$, где LC_{sg} – количество процессов, являющихся лидерами сессии либо группы, а во втором сомножителе нужно рассматривать отдельные суммы по потомкам, лежащим в одной сессии (можно считать от лидера сессии). Это очевидно из вышеописанной логики работы `setpgid()`.

Ввиду того, что n в рассматриваемой задаче может достигать $2^{16} - 1$, количество возможных конфигураций дерева процессов достаточно вели-

ко для применения жадного поиска на графах, а создание базовых шаблонов для конструирования дерева приводит к необходимости тщательного профилирования системы, с учётом активности отдельных компонентов и приложений, что затрудняет восстановление для произвольной системы. Исходя из вышесказанного, авторами статьи было сконструировано решение в рамках задания правил системных вызовов в некоторой формальной грамматике над множеством строк в специальной нотации, описанной в следующем разделе.

Формальная грамматика системных вызовов

Ввиду оценок количества генерируемых деревьев предложено решение задачи на основании анализа записи дерева процессов как строки, порождаемой в некоторой формальной грамматике. Отдельный процесс в данной нотации записывается как “ $p\ g\ s\ [^*]$ ”, где p , g , s – идентификаторы процесса, группы и сессии, а символы “ $[^*]$ ” – терминальные символы начала и конца списка потомков данного процесса. Таким образом поддерживается структура дерева уже на входной строке, что позволяет осуществлять разбор по правилу для `fork()` за один проход по ней. Само правило для `fork()` имеет вид: $fork(*\ *\ *\ [^*]) \Rightarrow *\ *\ *\ [^*\ \backslash 2\ \backslash 3\]\ \backslash 4]$.

Правило для `setsid()` имеет вид: $setsid(*\ *\ *\ [^*]) \Rightarrow \backslash 1\ \backslash 1\ \backslash 1\ \backslash 4]$.

Правило для `setpgid()` имеет вид:

$\{p\ p\ *\ [^*],\ setpgid(p,\ *\ *\ \backslash 3\ [^*])\} \Rightarrow \{p\ p\ *\ [^*],\ \backslash 1\ \backslash 1\ \backslash 3\ \backslash 4] \mid p\ p\ \backslash 3\ \backslash 4]\}$, где нетерминальные символы “ $\{$ ”, “ $\}$ ” следует интерпретировать как «в системе существует или существовала такая конфигурация».

На данном этапе грамматика уже имеет тип 1, ввиду правила для `setpgid`. Выделяя вторую часть правила $| p\ p\ *\ [^*]$, соответствующую контекстно-свободному случаю создания процессом своей группы, а также кешируя встреченные сессии при первом проходе по строке, можно осуществить поддержку `setsid()`, `setpgid()` на этом проходе. Результатом последнего будет первое промежуточное представление – дерево с частично восстановленными системными вызовами, покрывающими вышеуказанные случаи для `fork()`, `setsid()`, `setpgid()`.

Разработанный авторами анализатор строки по данной грамматике осуществляет на первом этапе восходящий разбор без учета контекста следующим образом: вводится стековая память с механизмом стекового кадра для каждого из процессов. Каждый процесс сохраняется на стеке в виде пары строк:

1) «состояние»-> «системный вызов» $\Rightarrow \dots \Rightarrow$ «состояние»,

2) метаданные: ссылки на родителя и процессы-потомки.

Ссылки на конкретное состояние родителя получаются в дальнейшем и хранятся в специальном упорядоченном словаре.

При встрече символа “[“ создается новый стековый кадр, а при встрече “]“ осуществляется его размотка, с восстановлением системных вызовов по бесконтекстным частям правил. По окончании ввода в стеке содержится первое промежуточное представление, по которому осуществляется нисходящий разбор с проверкой контекста. По окончании процедуры в стеке хранится дерево разбора.

Проблема укорачивания грамматики на системном вызове exit()

Авторы рассматривают проблему нетерминальных exit(), при которых информации о вышедших процессах во входной строке не содержится. Правило для exit(): $\{1 \ * \ * \ [*], \text{exit}(* \ * \ * \ [*])\} \Rightarrow 1 \ \backslash 1 \ \backslash 2 \ [\ \backslash 3 \ \backslash 7 \]$, которое делает конструируемую грамматику укорачивающей, рассматривается в случае, если никакими другими правилами невозможно присоединить потомка к INIT-процессу. В таком случае применяются 3 эвристики:

1) сессии потомка не существует ниже по стеку. Тогда создается потомок INIT с идентификатором, равным идентификатору данной сессии, и производится setsid(). Данный процесс помечается как вышедший и кладется на стек с соответствующей пометкой. Исследуемый потомок присоединяется к созданному процессу;

2) сессия существует, но в ней нет вышедших процессов. Вышедший процесс присоединяется к лидеру сессии, после чего к нему присоединяется исследуемый через последовательность setpgid(), setpgid(). Восстановленный процесс помечается как вышедший и кладется на стек аналогично эвристике 1;

3) сессия существует, в ней есть вышедшие процессы. Находится подходящий вышедший процесс, после чего к нему присоединяется исследуемый через последовательность setpgid(), setpgid(). В случае, если подходящих процессов не найдено, реализуется эвристика 2.

Пример работы анализатора

Рассмотрим строку "1 1 1 [5 4 3 [] 3 3 3 [4 4 3 [] 9 3 3 []]]". Состояние стека после первого прохода по строке:

```
0:[1, 1, 1]
...
4:[3, 1, 1], 'setsid', [3, 3, 3] # процесс получен в setsid(fork(INIT))
5:[1, [7, 9]] # ссылки на родителя и потомков
6:[4, 3, 3], 'setpgid', [4, 4, 3] # бесконтекстный setpgid()
7:[5,[]]
...
```

Также на данном этапе собираются некоторые данные для второй фазы разбора: упорядоченный словарь упорядоченных списков групп по сессиям и пр. Далее производится нисходящий разбор по КЗ-правилам для setsid(), setpgid(), а также обрабатывается проверка и восстановление для exit(). Стек имеет вид:

```
...
2:[5, 4, 3]
3:[11,[]] # процесс назначен потомком восстановленного
...
6:[4, 3, 3], 'setpgid', [4, 4, 3]
7:[5,[11]] # добавлена ссылка на вышедший процесс
...
10:[10, 4, 3], 'exit()' # вышедший процесс
11:[11, [3]] #
```

Эквивалентное дерево разбора:

```
[1,1,1]
└─ [3 1 1] →setsid()→[3 3 3]
    │           └─[9 3 3]
    └─[4 3 3]→setpgid(4)→[4 4 3]
                        └─[10 4 3]→exit()
                            └─[5 4 3]
```

Здесь запись “└─” эквивалентна системному вызову ‘fork()’

Заключение

Построенное решение представляет собой двухпроходный анализатор [3], временная сложность которого, с учетом использования эффективных структур данных, составляет $O(N \log(N) \log(G) \log(S))$, где N, G, S – число процессов, групп и сессий соответственно. Добавление новых системных вызовов и поддержка дополнительных атрибутов, таких как, например, открытые файлы, усложнит алгоритм анализа, как минимум, в $\log(K)$ раз, где K – количество дополнительных элементов, подлежащих проверке в нисходящем разборе. Введение новых сложных правил, подобных нетерминальному `exit()`, существенно усложняет работу анализатора. Грамматика, представленная такими правилами, является грамматикой типа 0. Такие усложнения могут затруднить интеграцию анализатора в практические приложения. Ввиду данных предположений, авторы планируют разрабатывать альтернативные нотации и способы решения.

Литература

1. *Лав Роберт*. Linux. Системное программирование. – 2-е изд. – М.: Вильямс, 2014.
2. *Ефанов Н.Н.* Восстановление состояния Linux-процесса оптимальными последовательностями системных вызовов на основе марковских преобразователей. //Тезисы 59-й научно-практической конференции МФТИ. М.: МФТИ, 2016.
3. *Ахо Альфред, Лам Моника, Сети Рави, Ульман Джеффи*. Компиляторы: принципы, технологии и инструментарий. – 2-е изд. – М.: Вильямс, 2008.

В. В. Рязанов

Московский физико-технический институт

(государственный университет)

E-mail: vasyarv@mail.ru

ПОИСК ИНФОРМАТИВНЫХ ПОДМНОЖЕСТВ ПРИЗНАКОВ И ОБЪЕКТОВ В ЗАДАЧАХ РАСПОЗНАВАНИЯ ПО ПРЕЦЕДЕТАМ

В различных прикладных областях возникает задача классификации с учителем (задача распознавания), когда начальная информация о множестве объектов представлена выборкой их признаковых описаний с указанием метки одного из подмножеств (класса) общей генеральной совокупности. Задача классификации по прецедентам состоит в том, чтобы по признаковому описанию отнести его в один из классов по данной начальной информации. Часто число признаков задается в соответствующей предметной области, а длина признаковых описаний выбирается по принципу «чем больше длина выборки, тем лучше». Возникают естественные подзадачи обработки начальной информации: поиск информативных признаков, объектов, выбросов, признаковых подпространств, преобразований признаков и т.д. В настоящей статье рассматривается один подход к одновременному поиску информативных признаков и информативных объектов, обеспечивающих наименьшее число ошибок на заданной начальной информации. В качестве базовой модели использовались алгоритмы вычисления оценок. Формируется билинейная форма, зависящая от весов признаков и объектов, принимающих значения 0 и 1, либо непрерывный аналог. С данной формой связан функционал качества распознавания. Создан алгоритм минимизации билинейной формы, имеющий линейную сложность в локальной окрестности. В итоге находятся признаки и объекты, имеющие единичную информативность. Приведены результаты экспериментов.

Ключевые слова: классификация, сокращение данных, билинейная форма.

Введение

Задачи предобработки множеств прецедентов, в частности выделения информативных признаков и объектов таблицы обучения, имеют большое значение при классификации. Действительно, выделяя информативные подтаблицы обучения, мы получаем более точные и быстрые алгоритмы классификации.

При слишком большом росте числа обучающих объектов некоторые классификаторы теряют способность обработать данные и строить реше-

ние. Небольшие выборки позволяют так же специалистам непосредственно изучать и визуализировать данные.

Обычно оптимизируют только признаковое пространство. Здесь часто используют алгоритмы, основанные на теории информации и корреляции [1]. В них вводятся критерии на основе энтропии, статистические эвристики и изучается взаимная корреляция признаков. Довольно популярны методы, основанные на поочередном добавлении/удалении признаков, и валидации на другой выборке [2]. Их недостатками являются долгое время работы, случайности при поиске оптимума, зависимость от валидационной выборки.

Среди современных подходов широко распространены методы отбора признаков, основанные на бинарных деревьях. В [3] рассматривается классический подход SSV (Separability Split Value) к ранжированию признаков: важность признака тем выше, чем чаще он встречается в узлах решающего дерева и чем раньше происходит по нему разбиение.

Задаче отбора объектов посвящено меньшее число публикаций, хотя она не менее важна при классификации. Часто ограничиваются простым эмпирическим поиском аномалий или фильтрацией объектов по какому-либо критерию, чтобы исключить явно нерелевантные объекты. Здесь есть подходы, основанные на методе k ближайших соседей [4], или других классификаторах.

В настоящей работе рассматривается одновременное выделение информативных подмножеств признаков и объектов, обеспечивающих наилучшую точность классификации. В качестве базового алгоритма используется модификация метода «алгоритм вычисления оценок» [5]. Вводятся параметры $\mathbf{x} = (x_1, x_2, \dots, x_n)$ и $\mathbf{y} = (y_1, y_2, \dots, y_m)$, характеризующие информативность признаков и объектов,

$x_i \in [0, 1]$, $i = 1, 2, \dots, n$, $y_i \in [0, 1]$, $i = 1, 2, \dots, m$. Значения 0 и 1 соответствуют отсутствию или учету признака/объекта. При классификации контрольной выборки берется линейная функция от оценок объектов за свои и чужие классы, которая путем преобразований сводится к билинейной форме $F(\mathbf{x}, \mathbf{y})$. В работе так же предложен локальный алгоритм оптимизации билинейного функционала, имеющий линейную сложность в окрестности O^2 от начальной точки. При формировании билинейного критерия участвует параметр t , отражающий число классов. Его значение выбирается из оптимизации стандартного функционала качества классификации.

Постановка задачи

Рассмотрим стандартную постановку задачи распознавания прецедентов. Пусть дано множество M объектов $\mathbf{z}$, представленных в виде их признаков описаний. Для простоты положим $\mathbf{z} \in R^n$. Множество $M = \bigcup_{i=1}^l K_i, K_i \cap K_j = \emptyset, i \neq j$, состоит из классов $K_i, i = 1, 2, \dots, l$. Информация о данном разбиении представлена в обучающей выборке $Z = \{\mathbf{z}_i, i = 1, 2, \dots, m\}$, состоящей из репрезентативных представителей своих классов: $Z = \bigcup_{i=1}^l K_i^*, K_i^* \subset K_i, |K_i^*| = n_i$. Задача состоит в сопоставлении $\forall \mathbf{z} \in R^n$ к одному из классов.

Далее рассмотрим «Алгоритмы вычисления оценок» [5], модификация которого используется в качестве базового алгоритма в настоящей работе. Считается, что все обучающие объекты $Z = \{\mathbf{z}_i, i = 1, 2, \dots, m\}$ разделены на l непересекающихся классов. Фиксируется натуральное число $k, 1 \leq k \leq n$. Объект $\mathbf{z}$ сравнивается со всеми обучающими объектами по всем подмножествам признаков $\Omega \subseteq \{1, 2, \dots, n\}, |\Omega| = k$, имеющих длину k для вычисления «степени близости» объекта к каждому из классов:

$$\Gamma_i(\mathbf{z}) = \frac{1}{n_i} \sum_{\mathbf{z}_t \in K_i} \sum_{\Omega: |\Omega|=k} B_{\Omega}(\mathbf{z}_t, \mathbf{z}).$$

Используется следующая функция близости двух объектов:

$$B_{\Omega}(\mathbf{z}_t, \mathbf{z}) = \begin{cases} 1, & |z_{ij} - z_j| \leq \varepsilon_j, \forall j \in \Omega, \\ 0, & \text{иначе.} \end{cases}$$

В [5] показано, что $\Gamma_i(\mathbf{z}) = \frac{1}{n_i} \sum_{\mathbf{z}_t \in K_i} C_{d(\mathbf{z}_t, \mathbf{z})}^k$, где

$d(\mathbf{z}_t, \mathbf{z}) = |\{j : |z_{ij} - z_j| \leq \varepsilon_j, j = 1, 2, \dots, n\}|$. Здесь $\varepsilon_j, j = 1, 2, \dots, n$ – настраиваемые параметры. Обычно они задаются как

$$\varepsilon_j = \frac{2}{m(m-1)} \sum_{\substack{u, v=1, 2, \dots, m \\ u \neq v}} |z_{uj} - z_{vj}|, j = 1, 2, \dots, n.$$

$\Gamma_i(\mathbf{z}), i = 1, 2, \dots, l$ объекту $\mathbf{z}$ присваивается класс K_j , имеющий максимальную оценку: $\Gamma_j(\mathbf{z}) > \Gamma_i(\mathbf{z}), i, j = 1, 2, \dots, l, i \neq j$. В противном случае происходит отказ от классификации объекта $\mathbf{z}$.

Введем параметры $\mathbf{x} = (x_1, x_2, \dots, x_n)$, которые буду обозначать «веса признаков», и параметры $\mathbf{y} = (y_1, y_2, \dots, y_m)$, обозначающие «веса объектов».

Далее модифицируем алгоритм АВО и его оценки $\Gamma_j(\mathbf{z})$ для тестируемых объектов $\mathbf{z} = (z_1, z_2, \dots, z_n)$ за класс K_j , добавив туда параметры $\mathbf{x}, \mathbf{y}$:

$$\Gamma_j(\mathbf{z}) = \frac{1}{|K_j|} \sum_{z_\nu \in K_j} y_\nu \sum_{\Omega \in \Omega_A} (\sum_{i \in \Omega} x_i) B_\Omega(\mathbf{z}_\nu, \mathbf{z}), \text{ где } |K_j| - \text{количество объектов в}$$

классе K_j , Ω – некоторое опорное множество (подмножество признаков) из совокупности опорных множеств Ω_A алгоритма АВО, $B_\Omega(\mathbf{z}_\nu, \mathbf{z})$ – функция близости объекта $\mathbf{z}$ к объекту из обучающей выборки $\mathbf{z}_\nu$. Можно

показать, что
$$\Gamma_j(\mathbf{z}) = \frac{1}{|K_j|} \sum_{z_i \in K_j} y_i \left(\sum_{t \in J(\mathbf{z}, \mathbf{z}_i)} x_t \right) C_{d(\mathbf{z}, \mathbf{z}_i)-1}^{k-1}, \quad \text{где}$$

$J(\mathbf{z}, \mathbf{z}_i) = \{ \nu : |z_{i\nu} - z_\nu| \leq \varepsilon_\nu, \nu = 1, \dots, n \}$, $d(\mathbf{z}, \mathbf{z}_i) = |J(\mathbf{z}, \mathbf{z}_i)|$, $k, \varepsilon_1, \varepsilon_2, \dots, \varepsilon_n$ – некоторые параметры классифицирующего алгоритма. При такой постановке в качестве опорных множеств используются всевозможные подмножества из k признаков.

Введем агрегированный функционал, который будет характеризовать обобщенное качество классификации. Для этого просуммируем с положительным коэффициентом τ_1 все оценки за «свои» классы и с отрицательным τ_2 – за «чужие». Без ограничения общности можно положить $\tau_1 = 1, \tau_2 = -t$.

Пусть дана контрольная выборка объектов $\mathbf{z}'_r$, содержащая представителей всех классов. Итоговый функционал будет выглядеть следующим образом:
$$f(\mathbf{x}, \mathbf{y}, t) = \sum_{\alpha=1}^l \sum_{z'_r \in K_\alpha} \Gamma_\alpha(\mathbf{z}'_r) - t \sum_{\alpha=1}^l \sum_{z'_r \notin K_\alpha} \Gamma_\alpha(\mathbf{z}'_r).$$
 Нетрудно показать, что он является билинейной формой от параметров $x_1, x_2, \dots, x_n, y_1, y_2, \dots, y_m$.

Итак, рассматривается следующая оптимизационная задача:

$$f(\mathbf{x}, \mathbf{y}, t) = \sum_{i=1}^n \sum_{j=1}^m c_{ij}(t) x_i y_j \rightarrow \max_{\mathbf{x} \in X, \mathbf{y} \in Y}.$$

При оптимизации данной формы можно рассматривать два варианта областей X и Y : $1 \geq x_i \geq 0$, $i = 1, 2, \dots, n$, $1 \geq y_j \geq 0$, $j = 1, 2, \dots, m$, либо $x_i \in \{0, 1\}$, $i = 1, 2, \dots, n$, $y_j \in \{0, 1\}$, $j = 1, 2, \dots, m$, для непрерывного и дискретного случая. Во втором случае выбор параметров $x_1, x_2, \dots, x_n$, $y_1, y_2, \dots, y_m$ означает выделение подтаблицы среди исходных данных. Величина t выступает параметром и не участвует в оптимизационном процессе.

Локальный шаг оптимизации

Рассмотрим предложенный выше метод локальной оптимизации. Зафиксируем некоторое $t = t_0 = \text{const}$. Тогда функционал принимает следующую форму:

$$f(\mathbf{x}, \mathbf{y}, t_0) = \sum_{i=1}^n \sum_{j=1}^m c_{ij}(t_0) x_i y_j = F(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^n \sum_{j=1}^m C_{ij} x_i y_j \rightarrow \max_{\mathbf{x} \in X, \mathbf{y} \in Y}.$$

Пусть $\tilde{\mathbf{x}} = (\tilde{x}_1, \dots, \tilde{x}_n) \in \{0, 1\}$, $\tilde{\mathbf{y}} = (\tilde{y}_1, \dots, \tilde{y}_m) \in \{0, 1\}$ – некоторая текущая точка в процессе оптимизации, $d(\mathbf{x}, \tilde{\mathbf{x}})$ – метрика Хемминга. Алгоритм является приближенным и состоит в локальной минимизации линейной формы, которая является приближением вариации функционала в рассматриваемой окрестности. Добавим следующие обозначения: $O_{k_x k_y}(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}) = \{\mathbf{x}, \mathbf{y} : d(\mathbf{x}, \tilde{\mathbf{x}}) = k_x, d(\mathbf{y}, \tilde{\mathbf{y}}) = k_y\}$, то есть $O_{k_x k_y}(\tilde{\mathbf{x}}, \tilde{\mathbf{y}})$ – множество таких векторов $\mathbf{x}, \mathbf{y}$, что они отличаются ровно в k_x признаках и в k_y векторах. Обозначим

$$F^{20}(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}) = \max_{O_{20}(\tilde{\mathbf{x}}, \tilde{\mathbf{y}})} F(\mathbf{x}, \tilde{\mathbf{y}}),$$

$$F^{02}(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}) = \max_{O_{02}(\tilde{\mathbf{x}}, \tilde{\mathbf{y}})} F(\tilde{\mathbf{x}}, \mathbf{y}),$$

$$F^{11}(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}) = \max_{O_{11}(\tilde{\mathbf{x}}, \tilde{\mathbf{y}})} F(\mathbf{x}, \mathbf{y}),$$

$$u_j(\mathbf{x}) = \sum_{i=1}^n C_{ij} x_i, \quad v_i(\mathbf{y}) = \sum_{j=1}^m C_{ij} y_j.$$

Далее покажем, что каждая из величин $F^{20}(\tilde{\mathbf{x}}, \tilde{\mathbf{y}})$, $F^{02}(\tilde{\mathbf{x}}, \tilde{\mathbf{y}})$, $F^{11}(\tilde{\mathbf{x}}, \tilde{\mathbf{y}})$ может быть вычислена за линейное $O(n+m)$ время, то есть локальный поиск минимума в окрестности $O^2(\tilde{\mathbf{x}}, \tilde{\mathbf{y}})$ имеет линейную сложность по отношению к числу признаков и объектов.

Сперва отметим, что поиск оптимума в окрестности $\mathbf{x}, \mathbf{y} \in O^1(\tilde{\mathbf{x}}, \tilde{\mathbf{y}})$, очевидно, линеен. Далее возможны 3 случая:

1) $\mathbf{x}, \mathbf{y} \in O_{11}(\tilde{\mathbf{x}}, \tilde{\mathbf{y}})$,

$$x_i = \begin{cases} \tilde{x}_i, & i \neq i_1, \\ 1 - \tilde{x}_i, & \text{иначе,} \end{cases} \quad y_i = \begin{cases} \tilde{y}_i, & i \neq i_2, \\ 1 - \tilde{y}_i, & \text{иначе,} \end{cases}$$

где i_1, i_2 — произвольная фиксированная пара координат. В этом случае

$$\begin{aligned} \Delta F(\mathbf{x}, \mathbf{y}) &= F(\mathbf{x}, \mathbf{y}) - F(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}) = \\ &= v_{i_1}(\tilde{\mathbf{y}})(x_{i_1} - \tilde{x}_{i_1}) + u_{i_2}(\tilde{\mathbf{x}})(y_{i_2} - \tilde{y}_{i_2}) + C_{i_1 i_2}(x_{i_1} - \tilde{x}_{i_1})(y_{i_2} - \tilde{y}_{i_2}), \\ \Delta F(\mathbf{x}, \mathbf{y}) &\leq v_{i_1}(\tilde{\mathbf{y}})(x_{i_1} - \tilde{x}_{i_1}) + u_{i_2}(\tilde{\mathbf{x}})(y_{i_2} - \tilde{y}_{i_2}) + \max_{i_2} |C_{i_1 i_2}| = \Delta_1 F(\mathbf{x}, \mathbf{y}), \\ \Delta F(\mathbf{x}, \mathbf{y}) &\leq v_{i_1}(\tilde{\mathbf{y}})(x_{i_1} - \tilde{x}_{i_1}) + u_{i_2}(\tilde{\mathbf{x}})(y_{i_2} - \tilde{y}_{i_2}) + \max_{i_1} |C_{i_1 i_2}| = \Delta_2 F(\mathbf{x}, \mathbf{y}). \end{aligned}$$

Очевидно, что перебор всех значений $\Delta_1 F(\mathbf{x}, \mathbf{y}), \Delta_2 F(\mathbf{x}, \mathbf{y})$ в поисках оптимума имеет сложность $O(n+m)$.

2) $\mathbf{x}, \tilde{\mathbf{y}} \in O_{20}(\tilde{\mathbf{x}}, \tilde{\mathbf{y}})$, где $x_i = \tilde{x}_i, i \notin \{i_1, i_2\}$. Очевидно, что $x_i \in \{0, 1\}, i \in \{i_1, i_2\}$. Отсюда следует

$$\begin{aligned} \Delta F(\mathbf{x}, \tilde{\mathbf{y}}) &= F(\mathbf{x}, \tilde{\mathbf{y}}) - F(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}) = v_{i_1}(\tilde{\mathbf{y}})(x_{i_1} - \tilde{x}_{i_1}) + v_{i_2}(\tilde{\mathbf{y}})(x_{i_2} - \tilde{x}_{i_2}), \\ \varphi_1 &= \min_{\mathbf{x}, \tilde{\mathbf{y}} \in O^2(\tilde{\mathbf{x}}, \tilde{\mathbf{y}})} \Delta F(\mathbf{x}, \tilde{\mathbf{y}}). \end{aligned}$$

2) Аналогично, третий случай $\tilde{\mathbf{x}}, \mathbf{y} \in O_{02}(\tilde{\mathbf{x}}, \tilde{\mathbf{y}})$, $y_i = \tilde{y}_i, i \notin \{i_1, i_2\}$, $y_i \in \{0, 1\}, i \in \{i_1, i_2\}$,

$$\Delta F(\tilde{\mathbf{x}}, \mathbf{y}) = F(\tilde{\mathbf{x}}, \mathbf{y}) - F(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}) = u_{i_1}(\tilde{\mathbf{x}})(y_{i_1} - \tilde{y}_{i_1}) + u_{i_2}(\tilde{\mathbf{x}})(y_{i_2} - \tilde{y}_{i_2}),$$

$$\varphi_2 = \min_{\tilde{x}, y \in O^2(\tilde{x}, \tilde{y})} \Delta F(\tilde{x}, y).$$

Очевидно, что поиск величин φ_1 и φ_2 имеет сложность $O(n + m)$.

Так как все из предложенных вариантов имеют линейную сложность, то и общая сложность их перебора так же составит $O(n + m)$.

Результаты численных экспериментов

В настоящей работе теоретические результаты были протестированы на различных данных. В ходе эксперимента исследовались следующие зависимости: точность классификации на оптимизированной выборке (ассигасу или доля верно распознанных объектов контрольной выборки), значение функционала F , а также площадь (произведение числа признаков на число объектов) новой выборки по отношению к исходной (share). В ходе экспериментов случайная доля начальных компонент векторов x и y зануляется, а затем проводится оптимизация описанного выше функци-

онала. Если $\frac{\sum x_i}{|x_i|} = 1$, то мы не удаляем признаки, если $\frac{\sum y_i}{|y_i|} = 0.3$, то в исходной выборке мы оставили 30% случайных объектов. Так же на каждой итерации отслеживается доля сохраненных признаков и доля сохраненных объектов.

Первая, исследуемая под названием “Credit”, представляет собой выборку для анализа кредитного скоринга (одобрить ли кредит заемщику). В выборке представлено 690 объектов, описанных 15 признаками.

Эксперимент 1, $\frac{\sum x_i}{|x_i|} = 1$, $\frac{\sum y_i}{|y_i|} = 1, t = 2$:

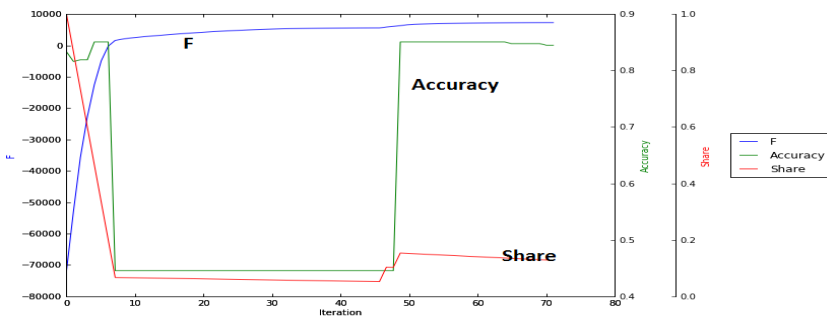


Рис. 1. Credit: Accuracy, F, share

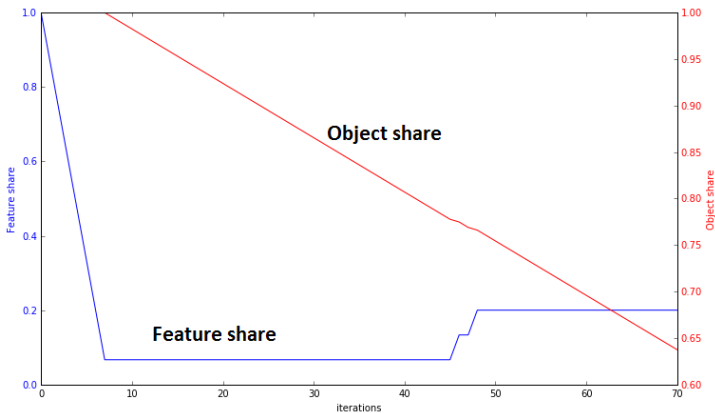


Рис. 2. Credit: Доля объектов, доля признаков

Из графиков можно сделать следующие выводы. Во-первых, можно заметить, что оптимизация функционала помогла сократить выборку на 50% при сохранении качества классификации на прежнем уровне. Во-вторых, алгоритм на разных итерациях проводил поочередный отбор признаков и объектов, и в оптимальной точке выборка лишилась части признаков и части объектов, то есть алгоритм на практике работает корректно по отношению к обоим векторам $\mathbf{X}$ и $\mathbf{Y}$. Можно отметить еще интересный момент – на определенной итерации качество классификации существенно уменьшилось, что связано, по всей видимости, с «ошибочным» удалением важного признака. Но в дальнейшем алгоритм включил признак обратно в обучающие данные. Похожие результаты в данной задаче получаются и при других параметрах t , и при другой начальной точке.

Вторая задача [6], которая исследовалась в ходе работы, посвящена классификации сигналов в ионосфере. Объекты характеризуются 17 пульсациями, записанными высокочастотными антеннами, и имеют в итоге 34 признака. Объекты разделены на обучающую и валидационную выборки, каждая из которых содержит в себе около 200 объектов.

Эксперимент 2, $\frac{\sum x_i}{|x_i|} = 0.95$, $\frac{\sum y_i}{|y_i|} = 7$, $t = 5$:

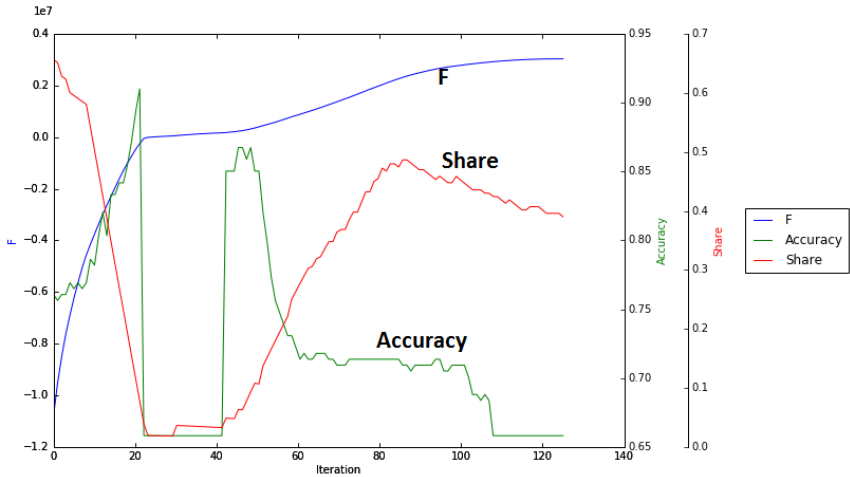


Рис. 3. Ионосфера: Accuracy, F, share

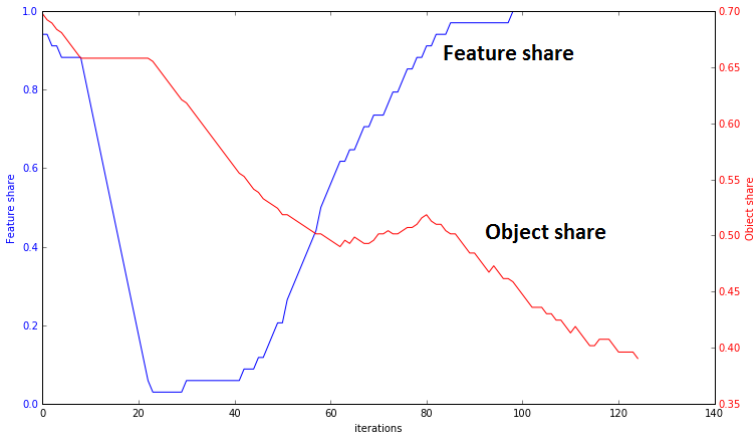


Рис. 4. Ионосфера: доля объектов, доля признаков

Наблюдаемые закономерности во второй задаче во многом повторяют аналогичные из задачи Credit. Спустя определенное число итераций ассигнату увеличивается при уменьшении размера выборки. Алгоритм так же производит поочередную обработку признаков и объектов. Главное отли-

чие от задачи Credit состоит в том, что спустя некоторое число итераций точность классификации начинает падать, но если отслеживать эту метрику отдельно, то можно выбрать итерацию, где ассигасу значительно выросло, а размер выборки уменьшился в несколько раз.

Заключение

Результаты, полученные в ходе предварительных экспериментов, демонстрируют способность алгоритма справляться с возложенной на него задачей – отбором признаков и объектов, сохраняя при этом качество классификации на прежнем уровне. Алгоритм так же не концентрируется исключительно на отборе признаков или на отборе объектов, а производит операции с ними поочередно.

Так же следует отметить, что линейная локальная сложность оптимизации позволяет проводить эксперименты быстро и эффективно.

Литература

1. Yu L., Liu H. Feature Selection for High-Dimensional Data: A Fast Correlation-Based Filter Solution // Proceedings of the 20th International Conference on Machine Learning (ICML-03). 2003. P. 856–863.
2. Pudil P., Novovičová J., Kittler J. Floating search methods in feature selection // Pattern Recognition Letters. 1994. V.15, issue 11. P. 1119–1125.
3. N. Jankowski, K. Grabczewski Feature Selection with Decision Tree Criterion. Hybrid Intelligent Systems // International Conference on. 2005. V. 00, N. , P. 212–217.
4. Hart P. The condensed nearest neighbor rule // IEEE Transactions on Information Theory. 1968. V. 14. issue 3. P. 515–516.
5. Журавлев Ю.И. Избранные научные труды. М.: Издательство «Магистр», 1998. – 420 с.
6. Lichman. UCI Machine Learning Repository [<http://archive.ics.uci.edu/ml>] // Irvine, CA: University of California, School of Information and Computer Science

Получено 20.04.2017

УДК 629.7

С. Л. Семаков^{1,2}, И. С. Семаков³

¹Московский физико-технический институт

(государственный университет), ²Финансовый университет,

³Московский авиационный институт (государственный университет)

E-mail: slsemafov@yandex.ru

АНАЛИЗ ПОСАДКИ С ТОЧКИ ЗРЕНИЯ ЕЁ БЕЗОПАСНОСТИ В ГРАЖДАНСКОЙ АВИАЦИИ

Предложена схема вероятностной оценки безопасной посадки гражданских самолетов.

Ключевые слова: самолет, безопасная посадка, случайный процесс, вероятность.

В работе [1] в качестве одного из приложений результатов работы [2] был предложен и практически реализован эффективный метод расчета вероятности безопасной посадки самолета, когда до полета с помощью математического моделирования движения самолета оценивается вероятность успешного приземления. В настоящей работе предлагается использовать результаты работы [1] для решения задачи об оценке точности и безопасности посадки в другой постановке, когда после реального полета нужно проанализировать имевшую место реализацию случайного процесса посадки и по ней апостериори оценить мастерство пилота с точки зрения вероятности успешного приземления. Такая постановка задачи особенно актуальна в гражданской авиации (ГА) [3].

Идея решения задачи в этой постановке состоит в следующем. Известно, что при заходе на посадку движение самолета достаточно точно может быть описано решением $y(t)$ линейной системы дифференциальных уравнений со случайными возмущениями, которое при определенных условиях на возмущения можно, в свою очередь, аппроксимировать n -мерным гауссовским процессом. Процессы $y(t)$, описывающие посадку конкретного воздушного судна при одних и тех же случайных возмущениях, но при различных манерах пилотирования, имеют, вообще говоря, различные характеристики. Если процесс $y(t)$ является гауссовским, то его плотность распределения полностью определяется моментами первого и второго порядков, так что это различие в характеристиках будет проявляться лишь

в значениях указанных моментов. Моменты же могут быть оценены с помощью известных методов (см., например, [4]) по реализации процесса $y(t)$, получаемой в результате расшифровки записей многоканальных средств регистрации параметров (МСПП) полета. Такими средствами оснащены все суда ГА.

Таким образом, чтобы оценить искомую вероятность, нужно выполнить следующую последовательность действий. Сначала по данным МСПП расшифровать реализацию процесса $y(t)$, соответствующую движению воздушного судна при конкретной посадке. Затем по этой реализации определить моменты первого и второго порядков процесса $y(t)$, причем, как правило, моменты первого порядка характеризуют номинальную траекторию посадки и известны заранее. После этого однозначно определяется плотность совместного распределения компонентов процесса $y(t)$, что позволяет по формулам, приведенным в [1], оценить искомую вероятность и погрешность этой оценки.

Литература

1. *Семаков С.Л.* Применение известного решения одной задачи о достижении границ немарковским процессом к оценке вероятности успешного приземления самолета // Известия РАН. Теория и системы управления. – 1996. – № 2. – С. 139–145.
2. *Семаков С.Л.* Вероятность первого достижения уровня компонентом многомерного процесса на заданном промежутке с соблюдением ограничений на его другие компоненты // Теория вероятностей и ее применения. – 1989. – Т. 34, № 2. – С. 402–406.
3. Инструкция по организации на предприятиях гражданской авиации систематического сбора, обработки, обобщения и анализа полетной информации. – М.: Воздушный транспорт, 1984.
4. *Бендат Дж., Пирсол А.* Прикладной анализ случайных данных. – М.: Мир, 1989.

Получено 20.04.2017

УДК 519.8

А. А. Лагуновская

Московский физико-технический институт

(государственный университет)

E-mail: a.lagunovskaya@phystech.edu

О МОДИФИКАЦИЯХ МЕТОДА ПРОЕКЦИИ ГРАДИЕНТА

Предлагаются модификации метода проекции градиента (МПГ) для задач выпуклой оптимизации, обобщающие метод проекции градиента для негладких функций (метод проекции субградиента (МПСГ)). Рассматривается наиболее предпочтительный выбор направления наибольшего убывания функции, что важно для численных экспериментов. Доказываются теоремы о сходимости рассмотренных модификаций метода проекции субградиента.

Ключевые слова: метод проекции градиента, метод проекции субградиента, выпуклая оптимизация, негладкая оптимизация, правило одномерной минимизации выбора шага, выбор направления спуска.

Введение

Рассматривается задача

$$f(x) \rightarrow \min, \quad x \in Q \subseteq R^n, \quad (1)$$

где Q — выпуклое множество, для которого легко найти проекцию некоторой точки $z \notin Q$ на множество Q , то есть

$$\bar{z} = \pi_Q(z) = \arg \min_{x \in Q} \|x - z\|^2. \quad (2)$$

Целевая функция $f: R^n \rightarrow R$ является выпуклой. Целевая функция может быть гладкой, такой что: $\|\nabla f(x) - \nabla f(y)\| \leq L\|x - y\|$, $x, y \in Q$, где $\nabla f(x)$ — градиент функции $f(x)$, L — его константа Липшица; или негладкой функцией, для которой возможно найти ее субдифференциал $\partial f(x)$ в точке $x \in Q$.

Метод проекции градиента (МПГ) для гладких функций был предложен многими авторами [1, 2, 3, 4]. Его общая схема имеет следующий вид: пусть $x_0 \in Q$, тогда итерации МПГ определяются по правилу:

$$x_{k+1} = x_k + \lambda_k p_k, k = 1, 2, \dots, p_k = \bar{x}_k - x_k, \bar{x}_k = \pi_Q(x_k - \alpha_k \nabla f(x_k)), \quad (3)$$

где число α_k выбирается по некоторому правилу [5]. Шаг λ_k выбирается из условий дробления шага [5] или $\lambda_k = \arg \min_{\lambda} f(x_k + \lambda p_k)$, $\lambda \in (0, \lambda_{\max})$, где $\lambda_{\max} = \max \{ \lambda : x = (x_k + \lambda p_k) \in Q \}$ [5]. Однако в учебно-научной литературе часто рассматривается МПГ [5, 6], когда

$$\lambda_k = 1, \quad x_{k+1} = \pi_Q(x_k - \alpha_k \nabla f(x_k)). \quad (4)$$

МПГ для задачи (1), когда $f(x)$ негладкая выпуклая функция, имеет вид [5]:

$$x_{k+1} = \pi_Q(x_k - \alpha_k g_k), \quad (5)$$

где $g_k \in \partial f(x_k)$ — некоторый субградиент, $\partial f(x_k)$ — субдифференциал

$f(x)$ в точке x_k , $\alpha_k > 0$ — число такое, что $\alpha_k \xrightarrow{k \rightarrow \infty} 0$, $\sum_{k=1}^{\infty} \alpha_k = \infty$. В [6]

этот метод также называется методом проекции субградиента (МПСГ).

Как показывают теоретические оценки и вычислительная практика, субградиентные методы сходятся сравнительно медленно и не являются достаточно эффективным способом минимизации негладких функций [5, 6]. Нам видятся две причины этого недостатка субградиентных методов, делающие их малоэффективными: возможность появления в методе таких $g_k \in \partial f(x_k)$, которые не являются направлениями убывания $f(x)$, а также малая длина шага $\alpha_k : \alpha_k \xrightarrow{k \rightarrow \infty} 0$, замедляющая уменьшение значения $f(x)$ на k -й итерации. В настоящей статье делается попытка устранить отмеченные недостатки субградиентных методов.

Модификации метода проекции субградиента⁴

Пусть в задаче (1) $f(x)$ — выпуклая негладкая функция на выпуклом замкнутом множестве $Q \subseteq R^n$. Для некоторой точки $x_0 \in R^n$ введем множество уровня значений функции $M(x_0) = \{x \in R^n : f(x) \leq f(x_0)\}$ и конус $K(x_0)$ возможных направлений к множеству $M(x_0)$ в точке $x_0 \in R^n : K(x_0) = \{h \in R^n : h = \lambda(x - x_0), \lambda > 0, x \in M(x_0)\}$, конус конической оболочки $\partial f(x_0)$:

⁴ Более точное название будет: метод проекции антисубградиента.

$$K_{\partial f} = \text{Cone } \partial f(x_0) = \{h \in R^n : h = \mu g, \mu \geq 0, \forall g \in \partial f(x_0)\}. \quad (6)$$

В монографии [6] показано, что $K(x_0) = \{h \in R^n : h^T g \leq 0, \forall g \in \partial f(x_0)\}$.

Определение [6]. Вектор $h \in R^n$ называется направлением скорейшего спуска в точке x , если функция $f(x)$ по этому направлению убывает наиболее быстро:

$$h = \arg \min_{\|y\|=1} f'(x, y)$$

Лемма 1. Справедливы следующие утверждения:

1. $K(x_0) = -K_{\partial f}^*$, где $K_{\partial f}^*$ — конус, сопряженный к конической оболочке $\partial f(x_0)$.
2. Пусть $h \in \text{ri } K(x_0)$, где $\text{ri } K(x_0)$ — относительная внутренность $K(x_0)$, тогда производная по направлению h : $f'(x_0, h) < 0$, то есть h — направление убывания функции $f(x)$.
3. Вектор $h = -\frac{\pi_{\partial f}(0)}{\|\pi_{\partial f}(0)\|}$, $f'(x_0, h) = -\|\pi_Q(0)\|$.

Доказательство

1. В монографии [7] показано, что $K(x_0)$, определенный в (5), можно представить в виде $K(x_0) = \{h \in R^n : h^T g \leq 0, \forall g \in \partial f(x_0)\}$. Отсюда также следует, что $K(x_0) = \{h \in R^n : h^T g \leq 0, \forall g \in K_{\partial f}\}$, где $K_{\partial f}$ определено в (6). Теперь из такого представления $K(x_0)$ следует, что $h \in -K_{\partial f}^*$ и $K(x_0) = -K_{\partial f}^* = -(\text{Cone } \partial f(x_0))^*$.
2. Пусть $h \in \text{ri } K(x_0)$ либо $h \in -\text{ri } K_{\partial f}^*$. Производная по направлению вычисляется по формуле [5]: $f'(x_0, h) = \max_{g \in \partial f} g^T h$. Очевидно, что $g \in \partial f \subset \text{Cone } \partial f(x_0)$. Возьмем $\|g\|=1, \|h\|=1$, тогда $g^T h = \cos \varphi$, где φ — угол между g и h . Пусть $g \in \text{Cone } \partial f(x)$, а $h \in \text{ri } (\text{Cone } \partial f(x))^*$, тогда $g^T h = \cos \varphi \leq 0$, но так как $h \in -\text{ri } K(x_0)$, то $g^T h < 0$. Теперь $f'(x_0, h) = \max_{g \in \partial f} g^T h < 0$, так как

$\partial f(x)$ - выпуклый компакт. Таким образом, $h \in \text{ri } K(x_0)$ - направление убывания функции $f: R^n \rightarrow R$.

3. По определению $f'(x, h) = \min_{\|h\|=1} \max_{g \in \partial f} g^T h$. Переставляя $\min_{\|h\|=1}$ и

$$\max_{g \in \partial f}, \text{ запишем: } f'(x, h) = \max_{g \in \partial f} \min_{\|h\|=1} g^T h = \max_{g \in \partial f} \frac{-g^T g}{\|g\|} = -\min_{g \in \partial f} \|g\| = -\|\pi_{\partial f}(0)\|.$$

$$\text{Значит } h = -\frac{\pi_{\partial f}(0)}{\|\pi_{\partial f}(0)\|}.$$

Лемма доказана.

Лемма 25. Пусть $f: R^n \rightarrow R$ - выпуклая негладкая функция. Тогда для любых $x, y \in R^n$ имеет место равенство $f(x) - f(y) = g_c^T(x - y)$, $g_c \in \partial f(x_c)$, x_c из интервала (x, y) .

Доказательство

Рассмотрим функцию одной переменной $\psi(\lambda)$, $\lambda \in R$:

$$\psi(\lambda) = f(x_k + \lambda(x_{k+1} - x_k)) = f(x_k + \lambda s_k), \lambda \in [0, 1], \psi(0) = f(x_k), \psi(1) = f(x_{k+1}).$$

Очевидно, что эта функция выпукла. Проведем прямую

$$y(\lambda) = \psi(0) + \lambda(\psi(1) - \psi(0)).$$

$$\frac{dy(\lambda)}{d\lambda} = \psi(1) - \psi(0),$$

тогда $dy(\lambda)/d\lambda$ — это средняя скорость убывания функции $\psi(\lambda)$ на отрезке $\lambda \in [0, 1]$. Проведем также прямую $z(\lambda)$ параллельно прямой $y(\lambda)$, касающуюся графика функции $\psi(\lambda)$ в точке λ_c при $x_k = x_k + \lambda_c(x_{k+1} - x_k)$. Очевидно, что λ_c и x_k существуют, а также существует и $\partial f(x_k)$. Прямая $z(\lambda)$ является опорной гиперплоскостью в R^2 в точке λ_c (рис. 1): $z(\lambda) = f(x_k) + \mu(f(x_{k+1}) - f(x_k))$, $\mu \in R^1$.

В R^n существует прямая, проходящая через x_k : $\hat{l} = \{x: x = x_k + \mu p_k, \mu \in R^1\}$ параллельно исходной прямой $l = \{x \in R^n: x = x_k + \mu p_k, \mu \in R^1\}$.

⁵ Данная лемма излагается в другой формулировке в монографии [11], но, на наш взгляд, приводимое доказательство более наглядно.

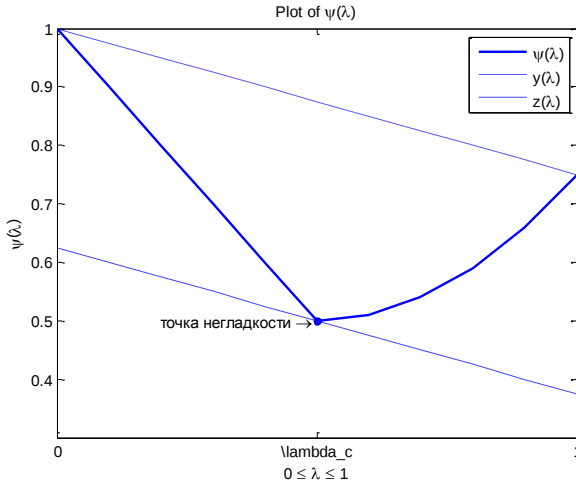


Рис. 1

Субдифференциал $\partial f(x_k)$ в точке x_k :

$$\partial f(x_k) = \left\{ g \in R^n : f(x) - f(x_k) \geq g^T(x - x_k), \forall x \in R^n \right\}.$$

Для каждого фиксированного $g \in \partial f(x_k)$ в точке x_k существует опорная гиперплоскость:

$$\Gamma = \left\{ \begin{pmatrix} x \\ u \end{pmatrix} \in R^{n+1} : u = f(x_k) + g^T(x - x_k) \right\} \quad \forall x \in R^n.$$

Если f — дифференцируемая функция, то $\partial f(x_k) = \nabla f(x_k)$ — единственная опорная плоскость, и справедлива теорема о среднем (формула конечных приращений Лагранжа). Такой случай в МПГ хорошо изучен [7, 8]. Рассмотрим случай, когда функция f — недифференцируемая. Подставив $x - x_k = \mu p_k$, $\mu \in R$ в уравнение для гиперплоскости Γ , получим в этом выражении совокупность всех прямых, каждая из которых определится конкретным субградиентом $g \in \partial f(x_k)$ и является опорной прямой в точке λ_c и соответственно в точке x_k . Прямая $z(\lambda)$ является одной из них, определяемой некоторым субградиентом g_k . Таким обра-

зом, существует такой g_k , что выполнено равенство: $f(x_{k+1}) - f(x_k) = g_k(x_{k+1} - x^*)$.

Лемма доказана.

Обозначим

$$A_g = (-K(x_k)) \cap \partial f(x_k). \quad (7)$$

Теперь рассмотрим следующую схему МПСГ.

Пусть

$$x_0 \in Q, \quad x_{k+1} = x_k + \lambda_k p_k, \quad k = 1, 2, \dots, \quad p_k = \bar{x}_k - x_k, \quad \bar{x}_k = \pi_Q(x_k - \alpha_k g_k), \quad (8)$$

где $g_k \in \partial f(x_k)$. Если $\lambda_k = 1$, то $x_{k+1} = \pi_Q(x_k - \alpha_k g_k)$. Если $\lambda_k \neq 1$, то

$$\lambda_k = \arg \min_{\lambda} f(x_k + \lambda p_k).$$

Определение. Пусть даны последовательности $\{x_k\}, \{g_k\}$, $g_k \in \partial f(x_k)$.

Субградиенты $g_k = g(x_k)$, $g_k = g(x_k)$ удовлетворяют условию Липшица с константой $L \geq 0$ на интервале (x_k, x_{k+1}) , если

$$\|g_k - g_k\| \leq L \|x_k - x_k\|, \quad x_k \in (x_k, x_{k+1}).$$

Теорема 1. Пусть $f: R^n \rightarrow R$ — выпуклая негладкая функция. Последовательность $\{g_k\}_{k=0,1,2,\dots}$ удовлетворяет условию Липшица с константой $L \geq 0$ на интервале (x_k, x_{k+1}) . Множество $M(x_0)$, $x_0 \in Q$ — выпуклый компакт, числа α_k удовлетворяют условию $0 \leq \alpha_k \leq 1/(L+c)$, $c > 0$.

а) Если $\lambda_k = 1$, тогда решение задачи (1) точка x^* существует и метод МПСГ по схеме (8) сходится к некоторой точке $x^* = \arg \min_{x \in Q} f(x)$.

б) Если $\lambda_k = \arg \min_{\lambda} f(x_k + \lambda p_k)$, тогда решение задачи (1) точка x^* существует и метод МПСГ по схеме (8) сходится к некоторой точке $x^* = \arg \min_{x \in Q} f(x)$.

Доказательство

Решение задачи существует по теореме Вейерштрасса, так как $M(x_0)$ — компакт, а $f(x)$ — непрерывная функция.

а) Для проекции $\bar{a} \in Q$ точки $a \in R^n$ на множество Q справедливо неравенство $(\bar{a} - a)^T (x - \bar{a}) \geq 0 \quad \forall x \in Q$ [8]. Пусть $\bar{a} = x_{k+1}$, $a = x_k - \alpha_k g_k$, $x = x_k$, $p_k = x_{k+1} - x_k$ тогда, подставляя в неравенство, получим [8]:

$$\|p_k\|^2 + \alpha_k g_k^T p_k \leq 0. \quad (9)$$

Рассмотрим равенство из Леммы 2, где

$$f(x_{k+1}) = f(x_k) + g_k^T (x_{k+1} - x_k), \quad x_k \in (x_k, x_{k+1}).$$

Сделаем следующие преобразования:

$$\begin{aligned} f(x_{k+1}) &= f(x_k) + g_k^T p_k + g_k^T p_k - g_k^T p_k = f(x_k) + g_k^T p_k + p_k (g_k^T - g_k^T) \leq \\ &\leq f(x_k) + g_k^T p_k + L \|x_k - x_k\| \|p_k\| \leq f(x_k) + g_k^T p_k + L \|p_k\|^2 = \\ &= f(x_k) + \frac{1}{\alpha_k} (\|p_k\|^2 + \alpha_k g_k^T p_k) + \left(L - \frac{1}{\alpha_k} \right) \|p_k\|^2. \end{aligned}$$

Учитывая (9) и неравенство $\alpha_k \leq 1/L + c$, получим

$$f(x_{k+1}) - f(x_k) \leq -c \|p_k\|^2. \quad (10)$$

Из (10) следует, что $f(x_{k+1}) < f(x_k)$ и МПСГ является методом спуска. Так как $M(x_0)$ — компакт, то $\lim_{k \rightarrow \infty} (f(x_{k+1}) - f(x_k)) = 0$ и $\lim_{k \rightarrow \infty} \|p_k\| = \lim_{k \rightarrow \infty} \|x_{k+1} - x_k\| = 0$ и $\lim_{k \rightarrow \infty} x_k = x$. Далее из [8, с.101] следует, что $g_k^T (x) (x - x) \geq 0$, то есть выполняется условие оптимальности в предельной точке $x = x^*$, являющейся решением задачи.

б) Обозначим последовательность $\{x_k\} = \{x_k^1\}$, если $\lambda_k = 1$ и $\{x_k\} = \{x_k^2\}$, если $\lambda_k = \arg \min_{\lambda} f(x_k + \lambda p_k)$. Используя (10), для $\lambda_k = \lambda_k^1$ и $\lambda_k = \lambda_k^2$ запишем: $f(x_{k+1}^2) - f(x_k^2) \leq f(x_{k+1}^1) - f(x_k^1) \leq -c \|p_k\|^2$. Следовательно, и при $\lambda_k = \lambda_k^2$ имеет место сходимость к решению: $\lim_{k \rightarrow \infty} x_k = x^*$.

Теорема доказана.

В рассмотренной выше схеме МСПГ для решения задачи (1) используется проекция точки $a_k = x_k - \alpha_k g_k$. Идейно близок к схеме (8) так называемый Метод зеркального спуска (МЗС) [9, 10]. В этом методе вместо точки $x_{k+1} = \pi_Q(x_k - \alpha_k g_k)$ (5) определяется точка

$$x_{k+1} = \arg \min_{x \in Q} (\alpha_k g_k^T (x - x_k) + V(x, x_k)), \quad (11)$$

где $V(x, x_k)$ — функция Брегмана.

$$V(x, x_k) = d(x) - d(x_k) - \nabla d^T(x_k)(x - x_k) \quad \forall x, x_k \in Q,$$

где $d(x)$ — прокс-функция, которая выбирается исходя из специфики задачи, а в общем случае требуется сильная выпуклость прокс-функции относительно рассматриваемой нормы в вещественном пространстве. Градиент функции Брегмана равен: $\nabla_x V(x, x_k) = \nabla d(x) - \nabla d(x_k)$.

Лемма 3. В итерационном процессе МПСГ (схема (13)) для $k=0,1,2,\dots$ справедливо неравенство $p_k^T p_k + \bar{\alpha}_k g_k^T p_k \leq 0$, где $\bar{\alpha}_k = \alpha_k / \beta_k$, $\beta_k \in (\lambda_{\min}, \lambda_{\max})$, $\lambda_{\min}, \lambda_{\max}$ — минимальное и максимальное собственные числа положительно определенной матрицы $d''(x_c)$, $x_c \in (\bar{x}_k, x_k)$, $p_k = \bar{x}_k - x_k$.

Доказательство

Критерий оптимальности выпуклой функции $f(x)$, $x \in Q$ в точке x^* [5]:

$$\nabla f(x^*)^T (x - x^*) \geq 0 \quad \forall x \in Q. \quad \text{Критерий оптимальности функции}$$

$$F(x, x_k) = \alpha_k g_k^T (x - x_k) + V(x, x_k) \quad \text{в точке } \bar{x}_k \text{ при } x = x_k:$$

$$\nabla_x F(x, x_k)^T (x_k - \bar{x}_k) = [\alpha_k g_k + \nabla d(\bar{x}_k) - \nabla d(x_k)]^T (x_k - \bar{x}_k) \geq 0,$$

$$[\alpha_k g_k^T p_k + (\nabla d(\bar{x}_k) - \nabla d(x_k))^T] p_k \leq 0,$$

$$\alpha_k g_k^T p_k + \beta_k p_k^T p_k \leq 0, \quad (12)$$

$$\text{где } \beta_k = (\nabla d(\bar{x}_k) - \nabla d(x_k))^T p_k / p_k^T p_k = p_k^T d''(x_c) p_k / p_k^T p_k,$$

$$x_c \in (\bar{x}_k, x_k), \beta_k \in (\lambda_{\min}, \lambda_{\max}). \text{ Также функция } d(x) \text{ — сильно выпукла. Тогда}$$

$$\forall x \in Q: 0 < \lambda_{\min} \leq \lambda_{\max} < \infty, \lambda_{\min}, \lambda_{\max} \text{ — собственные числа матрицы } d''(x).$$

Представим теперь неравенство (12) в виде $p_k^T p_k + \bar{\alpha}_k g_k^T p_k \leq 0$, где $\bar{\alpha}_k = \alpha_k / \beta_k$, $k=0,1,\dots$ Лемма доказана.

Замечание 1. В методе МПСГ (схема (8)) справедливо неравенство (9), при этом число α_k определяется из условия $\alpha_k \leq 1/L + c$. В схеме (13) потребуем для числа α_k выполнения условия $\alpha_k \leq \beta_k / L + c$ или $\bar{\alpha}_k = \alpha_k / \beta_k \leq 1/L + c$.

В известных вариантах МЗС [9, 10] шаг $\alpha_k \rightarrow 0$, например, $\alpha_k = c/\sqrt{k}$, где c — некоторая константа $c > 0$. В предлагаемом ниже варианте МПСГ α_k выбирается по другому правилу.

$$\begin{aligned} \text{Пусть } x_0 \in Q \subseteq R^n, \quad x_{k+1} = x_k + \lambda_k p_k, \quad k = 1, 2, \dots, \quad p_k = \bar{x}_k - x_k, \\ \bar{x}_k = \arg \min_{x \in Q} (\alpha_k g_k^T (x - x_k) + V(x, x_k)), \quad g_k \in \partial f(x_k). \end{aligned} \quad (13)$$

Если $\lambda_k = 1$, то $x_{k+1} = \bar{x}_k$. При $\lambda_k \neq 1$ $\lambda_k = \arg \min_{\lambda} f(x_k + \lambda p_k)$

Теорема 2. Пусть $f: R^n \rightarrow R$ — выпуклая негладкая функция. Последовательность $\{g_k\}_{k=0,1,2,\dots}$ удовлетворяет условию Липшица с константой $L \geq 0$ на интервале (x_k, x_{k+1}) . Множество $M(x_0)$, $x_0 \in Q$ — выпуклый компакт, числа α_k удовлетворяют условию $0 \leq \underline{\alpha} \leq \alpha_k \leq \frac{\beta_k}{L+c}$, $c > 0$.

а) Если $\lambda_k = 1$, тогда решение задачи (1) точка x^* существует и метод МПСГ по схеме (12) сходится к некоторой точке $x^* = \arg \min_{x \in Q} f(x)$.

б) Если $\lambda_k = \arg \min_{\lambda} f(x_k + \lambda p_k)$, тогда решение задачи (1) точка x^* существует и метод МПСГ по схеме (13) сходится к некоторой точке $x^* = \arg \min_{x \in Q} f(x)$.

Доказательство проводится аналогично доказательству Теоремы 1.

Замечание 2. В Теоремах 1, 2 $g_k \in \text{ri}A_g$, значит по Лемме 1 направление $-g_k$ является направлением убывания функции $f(x)$ в точке x_k .

Замечание 3. В Теоремах 1, 2 множество $M(x_0)$ — выпуклый компакт. Это требование не слишком ограничительно. Достаточно потребовать: множество $M_1(x_0) = M(x_0) \cap Q$ — выпуклый компакт.

Для реализации МПСГ необходимо в точке x_k знать $\partial f(x_k)$ и некоторый $g_k \in \partial f(x_k) \cap (-K(x_k))$, тогда $-g_k \in -\partial f(x_k) \cap K(x_0)$. Нахождение такого субградиента является новой вспомогательной задачей. Но можно указать один субградиент $g_k = \pi_{\partial f(x_k)}(0)$, который определяет направление наискорейшего спуска (Лемма 1), и для многих практических

задач легко находится. Несложно убедиться, что $g_k \in \partial f(x_k) \cap (-K(x_k))$.

Для определения g_k надо решить задачу

$$g_k = \pi_{\partial f(x_k)}(0) = \arg \min_{g \in \partial f(x_k)} (g^T g). \quad (14)$$

Так как при решении практических задач $\partial f(x_k)$ известен (если $\partial f(x_k)$ неизвестен, то задача МПСТ не решается), то полагаем, что решить задачу (14) возможно; если же задача (14) трудно решается, то находить g_k нерационально. В последнем случае надо находить g_k , используя специфику задачи.

В качестве условия окончания МПСТ можно взять следующие неравенства: $\max \left(\|p_k\|, \left| g_k^T p_k \right| \right) < \varepsilon_1$ или $\|\pi_Q(x_k - \alpha g_k) - x_k\| < \varepsilon_2$ (при достаточно большом α) и достаточно малых числах $\varepsilon_1 > 0$, $\varepsilon_2 > 0$.

Литература

1. Rosen J.B. The gradient projection method for nonlinear programming: Part I // SIAM, 1960. — V. 8(1). — P. 181—217.
2. Goldstein A.A. Convex programming in Hilbert space // Bull. Amer. Math. Soc. 1964. — V. 70. — № 5. — P. 709—710.
3. Демьянов В.Ф., Рубинов А.М. Приближенные методы решения экстремальных задач. — Л.: ЛГУ, 1968.
4. Левитин Е.С., Поляк Б.Т. Методы минимизации при наличии ограничений // ЖВМ. 1966. — Т. 6. — № 5. — С. 787—823.
5. Сухарев А.Г., Тимохов А.В., Федоров В.В. Методы оптимизации. Учебное пособие. — 2 изд. — М.: Физматлит, 2005. — 368 с.
6. Поляк Б.Т. Введение в оптимизацию. — М.: Наука, 1983. — 392 с.
7. Демьянов В.Ф., Васильев Л.В. Недифференцируемая оптимизация. — М.: Наука, 1981.
8. Жадан В.Г. Методы оптимизации. Часть 2. Численные алгоритмы: учебное пособие. — М.: МФТИ, 2015.
9. Немыровский А.С., Юдин Д.Б. Сложность задач и эффективность методов оптимизации. М.: Наука. Главная редакция физико-математической литературы, 1979. — 384 с.
10. Beck A., Teboulle M. Mirror descent and nonlinear projected subgradient methods for convex optimization // Oper. Res. Lett. 2003. — V. 31. — P. 167—175.
11. Кларк Ф. Оптимизация и негладкий анализ. — М.: Наука, 1988. — 280 с.

Получено 02.06.2017

УДК 518.12

*А. Г. Бирюков, А. В. Чернов*Московский физико-технический институт
(государственный университет)

E-mail: alexmipt@mail.ru

**ГРАДИЕНТНО-СОГЛАСОВАННЫЕ МЕТОДЫ РЕШЕНИЯ ЗАДАЧ
БЕЗУСЛОВНОЙ МИНИМИЗАЦИИ**

Предложено теоретическое описание одного класса методов решения задачи БМ. Основным его свойством является то, что вектор направления убывания функции строго ограничен от ортогональности с вектором-градиентом целевой функции. Доказаны теоремы о сходимости методов этого класса к стационарной точке. Показано, что известные классические методы решения задач БМ для сильно выпуклой функции принадлежат этому классу и имеют линейную скорость сходимости.

Ключевые слова: градиентно-согласованные методы, методы безусловной минимизации, скорость сходимости численных методов, градиентные методы, метод Ньютона

В статье рассматривается класс численных методов решения задачи БМ (1), в котором целевая функция один или два раза непрерывно дифференцируема:

$$\min f(x), \quad x \in R^n. \quad (1)$$

Хотя допустимым множеством задачи (1) является все пространство R^n , методы поиска локального минимума функции строятся на подмножестве $G \subset R^n$, связанном с некоторой начальной точкой $x^0 \in R^n$, иными словами, решается задача

$$f(x) \rightarrow \min_{x \in G}, \quad G = \left\{ x \in R^n : f(x) \leq f(x^0), x^0 \in R^n \right\}. \quad (2)$$

Очевидно, что множества решения задачи (1) и задачи (2) совпадают. Численные методы решения задачи БМ имеют следующую итерационную схему.

Пусть x^* – локальное решение задачи (1), $x^* = \lim_{k \rightarrow \infty} x^k$, а последовательность x^k строится по правилу

$$x^k = x^{k-1} + \lambda_k p^k, \quad k = 1, 2, \dots, \quad (3)$$

где p^k – вектор убывания функции $f(x)$, λ_k – число, которое назовем шагом по направлению убывания. Итак, численный метод генерирует две

последовательности: $\{x^k\}_{k=0}^{\infty}$ и $\{f(x^k)\}_{k=0}^{\infty}$, причем последовательность $\{f(x^k)\}_{k=0}^{\infty}$ релаксационная, т.е. $f(x^k) \leq f(x^{k-1})$, $k = 1, 2, \dots$. Преимущественно в исследуемых методах последовательность $\{f(x^k)\}_{k=0}^{\infty}$ строго релаксационная, т.е. $f(x^k) < f(x^{k-1})$, $k = 1, 2, \dots$. Именно свойство релаксационности гарантирует, что вся последовательность $\{x^k\}_{k=0}^{\infty}$ содержится во множестве G .

Реальные методы решения задачи БМ должны иметь конечное число шагов, поэтому вводятся условия останова методов [1,4]

Рассмотрим некоторые общие свойства численных методов решения задачи БМ. При этом будем полагать, что функция $f(x)$ дифференцируема на G .

Определение 1. Вектор $p^k \in R^n$ называется направлением убывания функции $f(x)$ в точке x^{k-1} , если существует такое $\bar{\lambda} \in R_+$ $\forall \lambda \in (0, \bar{\lambda})$, что выполнено неравенство $f(x^{k-1} + \lambda p^k) < f(x^{k-1})$.

Определение 2. Последовательность точек $\{x^k\}_{k=0}^{\infty}$ называется строго релаксационной для функции $f(x)$, если $\forall k > 0$ выполнено неравенство $f(x^k) < f(x^{k-1})$.

Определение 3. Открытый конус $K_y = \{p \in R^n : \langle \nabla f(x^{k-1}), p \rangle < 0\}$ называется конусом направлений убывания функции $f(x)$ в точке x^{k-1} .

Такое название для конуса K_y обосновывает следующая лемма.

Лемма 1 [2]. Пусть функция $f(x)$ дифференцируема в точке x^{k-1} и $p^k \in K_y$. Тогда p^k – вектор направления убывания функции в точке x^{k-1} .

Итак, для построения строго релаксационной последовательности $\{x^k\}_{k=0}^{\infty}$ для функции $f(x)$, определяемой (1), надо брать вектор $p^k \in K_y$ и соответствующий ему шаг λ_k , который обеспечивает строгое убывание функции $f(x)$.

Рассмотрим теперь способы выбора шага λ_k .

1. Способ дробления (иногда в литературе, например [2], этот способ называют правилом Армихо).

В этом случае $\lambda_k = \lambda_k^1$, если для некоторого числа λ_k выполнено условие

$$f(x^k) - f(x^{k-1}) \leq \varepsilon \lambda_k \langle p^k, \nabla f(x^{k-1}) \rangle, \quad (4)$$

где $\varepsilon \in (0,1)$ и $\langle p^k, \nabla f(x^{k-1}) \rangle < 0$.

Если условие (4) не выполняется, то шаг λ_k надо “дробить”, т.е. уменьшать до тех пор, пока данное условие не будет выполняться.

2. Исчерпывающий поиск.

В этом случае $\lambda_k = \lambda_k^2$, т.е. тому значению параметра λ , при котором реализуется минимум функции $\psi(\lambda)$ при $\lambda > 0$, т.е.

$$\lambda_k^2 = \arg \min_{\lambda > 0} f(x^{k-1} + \lambda p^k).$$

Безусловно, существуют и другие способы выбора длины шага λ_k , однако перечисленные наиболее часто применяются на практике.

Теперь введем класс методов решения задачи БМ, определив некоторые свойства последовательности векторов $\{p^k\}_{k=1}^{\infty}$.

Определение 4. Последовательность векторов $\{p^k\}_{k=1}^{\infty} \in R^n \setminus \{0_n\}$ называется градиентно-согласованной с последовательностью точек $\{x^k\}_{k=0}^{\infty}$, если существуют такие числа $\alpha > 0$ и $\beta > 0$, что для любого $k > 0$ выполняются условия

$$\|p^k\| \leq \alpha \|\nabla f(x^{k-1})\|; \quad \beta \|\nabla f(x^{k-1})\|^2 \leq -\langle p^k, \nabla f(x^{k-1}) \rangle. \quad (5)$$

Методы решения задачи БМ, для которых выполнено условие (5), будем называть градиентно-согласованными методами (ГСМ).

Другое определение градиентно-согласованных методов введено в монографии [3].

Определение 5 [3]. Пусть функция $f : G \in R^n \rightarrow R^1$ дифференцируема на G и $\{x^k\}_{k=0}^{\infty} \in G$ – заданная последовательность. Последовательность $\{p^k\}_{k=1}^{\infty} \in R^n$ ненулевых векторов называется градиентно-согласованной с

последовательностью $\{x^k\}_{k=0}^\infty \in G$, если существует такая F -функция σ , что

$$\left\langle \nabla f(x^k), \frac{p^k}{\|p^k\|} \right\rangle \geq \sigma(\|\nabla f(x^k)\|),$$

где $k = 0, 1, 2, \dots$. При этом F -функция σ определяется следующим образом:

$$\sigma : [0; \infty) \rightarrow [0; \infty),$$

причем, если $\lim_{k \rightarrow \infty} \sigma(t_k) = 0$, то $\lim_{k \rightarrow \infty} t_k = 0$.

Таким образом, определение 4 градиентно-согласованных методов является некоторым вариантом градиентно-согласованных методов из определения 5.

Имеет место следующая лемма.

Лемма 2. В определении 4 градиентно-согласованных методов для чисел α и β справедливо неравенство $\beta \leq \alpha$.

Доказательство: Пусть φ_k – угол между векторами p^k и $\nabla f(x^{k-1})$. Тогда из второго неравенства (3) видно, что $-\cos \varphi_k > 0$ или $-\langle p^k, \nabla f(x^{k-1}) \rangle > 0$. Учитывая связь нормы и скалярного произведения $\langle a, b \rangle \leq \|a\| \cdot \|b\|$ и выше сказанное, из (5) получаем

$$\beta \|\nabla f(x^{k-1})\|^2 \leq \|p^k\| \cdot \|\nabla f(x^{k-1})\|.$$

Вспомним теперь первое неравенство из (5) и продолжим цепочку неравенств далее:

$$\|p^k\| \cdot \|\nabla f(x^{k-1})\| \leq \alpha \|\nabla f(x^{k-1})\|^2.$$

Объединяя полученные неравенства, находим утверждение леммы: $\beta \leq \alpha$.

Неравенство $-\langle p^k, \nabla f(x^{k-1}) \rangle > 0$ означает, что угол φ_k между векторами p_k и $\nabla f(x^{k-1})$ всегда больше прямого угла.

Обозначим $\bar{p}_k$, $\bar{\varphi}_k$ предельные, т.е. граничные, значения вектора p_k и угла φ_k в конусе K_y причем $\beta/\alpha = -\cos \bar{\varphi}_k$.

Введенные понятия позволяют легко доказать теоремы о свойствах сходимости градиентно-согласованных методов.

Для доказательства сходимости последовательности точек $x^k \rightarrow x^*$, $k \rightarrow \infty$ будем рассматривать два способа выбора чисел λ_k : $\lambda_k = \lambda_k^1$ или $\lambda_k = \lambda_k^2$.

В указанных предположениях верна следующая лемма.

Лемма 3. Пусть функция $f(x)$ дифференцируема на множестве G , а её градиент удовлетворяет условию Липшица с константой $L > 0$:

$$\|\nabla f(x) - \nabla f(y)\| \leq L\|x - y\|.$$

Тогда

1. Если $\forall k > 0$ выполняется $\langle p^k, \nabla f(x^{k-1}) \rangle < 0$ и числа λ_k удовлетворяют условиям

$$0 < \lambda_k \leq \frac{2(\varepsilon - 1)}{L\|p^k\|^2} \langle p^k, \nabla f(x^{k-1}) \rangle \quad \forall \varepsilon \in (0, 1), \quad (6)$$

то имеет место оценка

$$f(x^k) - f(x^{k-1}) \leq \varepsilon \lambda_k \langle p^k, \nabla f(x^{k-1}) \rangle,$$

т.е. выполняется условие (4) в методе дробления шага λ_k^1 .

2. Если последовательность $\{p^k\}_{k=1}^\infty \in R^n \setminus \{0_n\}$ градиентно согласована, то для λ_k выполняются условия

$$0 < \lambda_k \leq \frac{2(1 - \varepsilon)\beta}{L\alpha^2} \quad \forall \varepsilon \in (0, 1).$$

(7)

Пункт 1 леммы 3 доказан в [1], поэтому будет приведено только доказательство пункта 2.

Доказательство.

В условиях пункта 2 леммы 3 уточним неравенство (6) для шага λ_k :

$$\frac{2(\varepsilon - 1) \langle p^k, \nabla f(x^{k-1}) \rangle}{L\|p^k\|^2} \geq \frac{2(1 - \varepsilon)\beta \|\nabla f(x^{k-1})\|^2}{L\|p^k\|^2} \geq \frac{2(1 - \varepsilon)}{L\alpha^2} = \bar{\lambda} \geq \lambda_k > 0.$$

Т.е. получено неравенство (7). Из последнего неравенства можно сделать вывод, что шаг λ_k можно выбирать из условия $0 < \underline{\lambda} \leq \lambda_k \leq \bar{\lambda}$, где $\underline{\lambda} > 0$ выбирается из условия $\underline{\lambda} \leq \bar{\lambda}$.

Приведенные утверждения позволяют доказать следующую теорему о сходимости градиентно-согласованных методов при решении задач БМ.

Теорема 1. Пусть функция $f(x)$, $x \in G$, ограничена снизу, выполнены условия леммы 3, а шаг λ_k может выбираться как по способу дробления, так и по способу исчерпывающего поиска, т.е. $\lambda_k = \lambda_k^1$ и $\lambda_k = \lambda_k^2$. Тогда

$$\lim_{k \rightarrow \infty} \|\nabla f(x^k)\| = 0.$$

Доказательство. 1. Пусть длина шага выбирается способом дробления $\lambda_k = \lambda_k^1$. Тогда в силу того, что выполняются условия леммы 3 и последовательность $\{p^k\}_{k=1}^\infty$ градиентно согласована, получаем следующую оценку:

$$\begin{aligned} f(x^{k-1} + \lambda_k p^k) - f(x^{k-1}) &\leq \varepsilon \lambda_k \langle p^k, \nabla f(x^{k-1}) \rangle \leq \\ &\leq -\varepsilon \lambda_k \beta \|\nabla f(x^{k-1})\|^2 = -\varepsilon \tilde{\lambda} \beta \|\nabla f(x^{k-1})\|^2, \end{aligned} \quad (8)$$

где $\lambda_k = \tilde{\lambda}$, $\tilde{\lambda} \in [\underline{\lambda}, \bar{\lambda}]$ — некоторая константа; $\underline{\lambda}$ и $\bar{\lambda} = 2(1-\varepsilon)\beta/(L\alpha^2)$ определены в лемме 3. Из приведенной оценки видно, что $f(x^k) < f(x^{k-1})$. Тогда в силу ограниченности снизу функции $f(x)$ получим $\lim_{k \rightarrow \infty} \|\nabla f(x^k)\| = 0$.

1. Пусть $\lambda_k = \lambda_k^2$. Определим две последовательности точек $\{x^{k,1}\}_{k=1}^\infty$ и $\{x^{k,2}\}_{k=1}^\infty$, получающиеся когда $\lambda_k = \lambda_k^1$ и $\lambda_k = \lambda_k^2$ соответственно. Очевидно, что $f(x^{k,2}) - f(x^{k-1}) \leq f(x^{k,1}) - f(x^{k-1})$. Тогда, используя неравенство (8), находим

$$f(x^{k,2}) - f(x^{k-1}) \leq -\varepsilon \tilde{\lambda} \beta \|\nabla f(x^{k-1})\|^2.$$

Таким образом, как и в предыдущей части доказательства:

$$\lim_{k \rightarrow \infty} \|\nabla f(x^k)\| = 0.$$

В случае сильной выпуклости функции $f(x)$ результат, полученный в теореме 1, может быть уточнен для градиентно-согласованных методов.

Теорема 2. Пусть функция $f(x)$ сильно выпукла на множестве G с константой сильной выпуклости χ и выполнены условия теоремы 1. Тогда градиентно-согласованный метод сходится к единственной точке x^* с линейной скоростью, т.е. существует такое число $q \in (0,1)$, что

$$f(x^k) - f(x^*) \leq q^2 (f(x^k) - f(x^*)); \quad \|x^k - x^*\| \leq Cq^k. \quad (9)$$

Здесь $k \in N$, $C = \sqrt{(f(x^0) - f(x^*))/\chi}$, $q = \sqrt{1 - 4\chi\epsilon\tilde{\lambda}\beta}$, $q \in (0,1)$, $\epsilon \in (0,1)$, а $\tilde{\lambda} \in [\underline{\lambda}, \bar{\lambda}]$ определена в теореме 1.

Доказательство. Так как функция $f(x)$ сильно выпукла на множестве G , то множество G выпукло, ограничено и замкнуто [1], причем $f(x)$ достигает своего минимума в точке $x^* \in \text{int } G$ при $f(x^*) < f(x^0)$. Так как функция $f(x)$ дифференцируемая, то $\|\nabla f(x^*)\| = 0$.

Рассмотрим теперь условия сходимости для градиентно-согласованной последовательности с конкретным выбором шага λ_k .

1. Пусть $\lambda_k = \lambda_k^1$. Так как выполнены условия теоремы 1, то имеет место оценка:

$$f(x^k) - f(x^{k-1}) \leq -\epsilon\tilde{\lambda}\beta \|\nabla f(x^{k-1})\|^2.$$

Воспользуемся оценкой для сильно выпуклой функции [1]:

$$\|\nabla f(x^k)\|^2 \geq 4\chi(f(x^k) - f(x^*)).$$

При $q = \sqrt{1 - 4\chi\epsilon\tilde{\lambda}\beta}$ получим

$$\begin{aligned} f(x^k) - f(x^*) &\leq f(x^{k-1}) - f(x^*) - 4\chi\epsilon\tilde{\lambda}\beta (f(x^{k-1}) - f(x^*)) = \\ &= (f(x^{k-1}) - f(x^*)) \cdot (1 - 4\chi\epsilon\tilde{\lambda}\beta) = q^2 (f(x^{k-1}) - f(x^*)). \end{aligned}$$

Проанализируем значения величины q . Возьмем $\tilde{\lambda} = \bar{\lambda} = 2(1 - \epsilon)\beta/L\alpha^2$ (согласно лемме 3). Подставляя указанное значение $\tilde{\lambda}$, находим следующую формулу:

$$q^2 = 1 - 4\varepsilon(1 - \varepsilon) \frac{2\chi}{L} \left(\frac{\beta}{\alpha} \right)^2.$$

Т.к. $4\varepsilon(1 - \varepsilon) \leq 1$ при $\varepsilon \in (0, 1)$, $2\chi/L \leq 1$, $\beta \leq \alpha$ (лемма 2), то для величины q получим оценку: $q \in [0, 1)$. Отметим, что оценка $2\chi/L \leq 1$ следует из неравенств

$$2\chi \|x - x^*\| \leq \|\nabla f(x)\| \leq L \|x - x^*\|.$$

Левое неравенство определяется свойствами сильно выпуклой функции, а правое неравенство получено из условия Липшица для градиента целевой функции $f(x)$.

Т.к. $q < 1$, то при $k \rightarrow \infty$ метод имеет линейную скорость сходимости (иными словами, $f(x)$ сходится с линейной скоростью к $f(x^*)$).

Теперь оценим отклонение $\|x^k - x^*\|$. С учетом неравенства

$$f(x^k) - f(x^*) \leq q^2 (f(x^{k-1}) - f(x^*))$$

и оценки

$$\chi \|x^k - x^*\|^2 \leq (f(x^k) - f(x^*)),$$

получим

$$\|x^k - x^*\|^2 \leq \frac{f(x^k) - f(x^*)}{\chi} \leq \frac{q^2}{\chi} (f(x^{k-1}) - f(x^*)) \leq \frac{q^{2k}}{\chi} (f(x^0) - f(x^*)),$$

или

$$\|x^k - x^*\| \leq Cq^k; \quad C = \sqrt{(f(x^0) - f(x^*)) / \chi}.$$

Таким образом, для способа выбора длины шага дроблением теорема доказана.

2. Пусть $\lambda_k = \lambda_k^2$.

Обозначим через $\{x^{k,1}\}_{k=1}^\infty$ и $\{x^{k,2}\}_{k=1}^\infty$ последовательности векторов, полученные в случаях, когда $\lambda_k = \lambda_k^1$ и $\lambda_k = \lambda_k^2$. Тогда имеют место неравенства

$$f(x^{k,2}) - f(x^*) \leq f(x^{k,1}) - f(x^*) \leq q^2 (f(x^{k-1}) - f(x^*)),$$

где первое неравенство очевидно, а второе получено в пункте 1 данного доказательства. Следовательно, оценки для скорости сходимости, полученные в 1 и определяющие линейную сходимость метода, справедливы и в случае $\lambda_k = \lambda_k^2$ для скорости сходимости последовательностей $\{f(x^{k,2})\}_{k=1}^\infty$ и $\{x^{k,2}\}_{k=1}^\infty$.

Теоремы 1 и 2 характеризуют свойства сходимости для класса градиентно-согласованных методов. На основе этих теорем приведем оценки скорости сходимости для трех классических методов, применяемых для решения задач БМ.

1. Метод градиентного спуска или градиентный метод. Решение x^* задачи БМ: $\min f(x)$ при $x \in R^n$. Здесь $x^* = \lim_{k \rightarrow \infty} x^k$, а $\{x^k\}_{k=0}^\infty$ – последовательность точек, которая определяется $x^k = x^{k-1} - \lambda_k \nabla f(x^{k-1})$, $k = 1, 2, \dots$, $\lambda_k = \lambda_k^1$ (способ дробления) или $\lambda_k = \lambda_k^2$ (исчерпывающий поиск). В случае варианта исчерпывающего поиска градиентный метод называется методом наискорейшего спуска [2].

Теорема 3. Пусть функция $f(x)$, $x \in G \subset R^n$, сильно выпукла на множестве G с константой κ и градиент функции $\nabla f(x)$ удовлетворяет условию Липшица с константой L . Тогда градиентный метод сходится к единственной точке минимума с линейной скоростью:

$$f(x^k) - f(x^*) \leq g^2 (f(x^{k-1}) - f(x^*)),$$

$$\|x^k - x^*\| \leq Cg^k, \quad k = 1, 2, \dots,$$

где $C = \sqrt{(f(x^0) - f(x^*)) / \kappa}$, $g^2 = 1 - 8\varepsilon(1 - \varepsilon)\kappa / L$.

При $\varepsilon = 0.5$ получаем $g^2 = 1 - 2\kappa / L$.

Доказательство. Очевидно, что если $p^k = -\nabla f(x^{k-1})$, то $\alpha = \beta = 1$ и из формулы (9) теоремы 2 получим требуемое значение для коэффициента скорости сходимости g . Значение константы C приведено в теореме 2.

Замечание 1. [2]. Если функция $f(x)$ дважды непрерывно дифференцируема и выполнено условие

$$my^T y \leq y^T f''(x)y \leq My^T y, \quad x \in G, y \in R^n,$$

$$0 < m = 2\kappa \leq M, \quad \lambda_k = \lambda_k^2,$$

то

$$L = M, \quad 2\kappa = m, \quad \|x^k - x^*\| \leq \bar{g} \|x^{k-1} - x^*\|,$$

где $\bar{g} = (M - m)/(M + m)$. Т.е. имеем также линейную скорость сходимости. При этом очевидно, что $\bar{g} \leq \sqrt{1 - m/M}$.

Замечание 2. Если $\varepsilon = 0.5$, то $\lambda_k = 1/L$. Это значение шага λ_k применяется в быстром градиентном методе [4].

2. Метод сопряженных градиентов.

Метод сопряженных градиентов (схема Флетчера–Ривса) [5] имеет следующий вид:

$$x^{k+1} = x^k + \lambda_k p^k; \quad p^k = -r^k + \beta_k p^{k-1};$$

$$r^k = \nabla f(x^k); \quad \beta_k = \begin{cases} \|r^k\| / \|r^{k-1}\|, & k \neq 0, n, 2n, \dots; \\ 0, & k = 0, 2n, \dots \end{cases}$$

Индексы k имеют вид: $k = mn + i$, где $i = \overline{0, n-1}$, $m = 0, 1, 2, 3, \dots$. В этой схеме после n шагов происходит обновление вектора p^k , т.е. $p^k = -r^k$. Данный метод глубоко исследован, его подробное описание можно найти, например, в [1, 5, 6] и для него справедлива следующая теорема:

Теорема 4. [5]. Пусть функция $f(x)$ сильно выпукла и в окрестности единственной точки минимума x^* вторая производная функции $f''(x)$ удовлетворяет условию Липшица. Тогда для метода сопряженных градиентов справедлива оценка

$$\|x^{(m+1)n} - x^*\| \leq C \|x^{mn} - x^*\|^2, \quad m \geq m_0.$$

Здесь m_0 – некоторый номер блочной итерации, т.е. скорость сходимости метода квадратичная.

Приведем оценку скорости сходимости МСГ при $i < n$, когда $f(x)$ – квадратичная функция.

Теорема 5. [5]. Пусть выполнены условия теоремы 4 и $k = mn + i$, где $i < n$. Тогда при $g = (\sqrt{M} - \sqrt{m})/(\sqrt{M} + \sqrt{m})$ имеет место оценка

$$\|x^k - x^*\| \leq 2\sqrt{M/m} g^k \|x^0 - x^*\|.$$

Рассмотрим МСГ как градиентно-согласованный метод, причем для МСГ используем схему Полака–Рибьера [6], в которой β_k определяется следующим образом:

$$\beta_k = \begin{cases} \langle r^k, r^k - r^{k-1} \rangle / \|r^{k-1}\|^2, & k \neq 0, n, 2n, \dots; \\ 0, & k = 0, 2n, \dots \end{cases}$$

Теорема 6. Пусть функция $f(x)$, $x \in R^n$, дважды непрерывно дифференцируема и существуют такие константы m и M , $0 < m \leq M < \infty$, что

$$my^T y \leq y^T f''(x) y \leq My^T y, \quad x \in R^n, y \in R^n.$$

Тогда для МСГ поиска минимума функции $f(x)$ при $x \in R^n$ и $\lambda_k = \lambda_k^2$, $k = mn + i$, где $0 \leq i < n$, $m = 0, 1, 2, \dots$, имеет место оценка коэффициента скорости сходимости:

$$g = \sqrt{1 - m^3 / (M(m + M)^2)}.$$

Доказательство. Пусть $i = 0$, тогда на такой итерации выполняется шаг градиентного спуска $p^k = -\nabla f(x^k)$ и по теореме 3

$$f(x^k) - f(x^*) \leq g^2 (f(x^{k-1}) - f(x^*)), \quad g^2 = 1 - m/M.$$

Пусть теперь $i \neq 0$, $k \neq mn$. Тогда в силу того, что МСГ – градиентно-согласованный метод, имеем

$$\langle p^k, \nabla f(x^{k-1}) \rangle = -\langle \nabla f(x^{k-1}), \nabla f(x^{k-1}) \rangle.$$

т.е. $\beta = 1$. В [6] приведена оценка

$$\|p^k\| \leq \|\nabla f(x^{k-1})\| \sqrt{1 + M/m},$$

т.е. $\alpha = \sqrt{1 + M/m}$. Тогда по теореме 2 при $\beta = 1$ получаем, что $\forall \varepsilon \in (0, 1)$:

$$g^2 = 1 - 4\varepsilon(1 - \varepsilon)m/M\alpha^2.$$

При $\varepsilon = 0.5$ получаем

$$g^2 = 1 - m/M\alpha^2 = 1 - m^3 / (M(m + M)^2).$$

Таким образом, скорость сходимости метода линейная.

3. Метод Ньютона.

Данный метод широко известен (его подробное описание можно найти, например, в [1, 2, 5]). Схема метода имеет следующий вид:

$$x^* = \lim_{k \rightarrow \infty} x^k; \quad x^{k+1} = x^k + \lambda_k p^k; \quad p^k = -f''^{-1}(x^k) \nabla f(x^k).$$

Величина шага λ_k определяется двумя способами:

- на начальном этапе используется правило дробления $\lambda_k = \lambda_k^1$ при $0 < \varepsilon < 1/2$ или исчерпывающий поиск $\lambda_k = \lambda_k^2$;
- при $\lambda_k = 1$ и $0 < \varepsilon < 1/2$ проверяется условие:

$$f(x^k) - f(x^{k-1}) \leq -\varepsilon \lambda_k p_k^T \nabla f(x^{k-1}).$$

Если условие выполняется, то $\lambda_k = 1$. В противном случае применяется правило дробления $\lambda_k = \lambda_k^1$ или исчерпывающий поиск $\lambda_k = \lambda_k^2$.

После того как точка x^k будет принадлежать области сходимости метода Ньютона, то $\lambda_k = 1$ и метод сходится со сверхлинейной (квадратичной в случае липшицевости второй производной функции $f''(x)$) скоростью.

Теорема ниже определяет оценку скорости сходимости метода Ньютона для тех случаев, когда величина шага определяется по правилу дробления $\lambda_k = \lambda_k^1$ или посредством исчерпывающего поиска $\lambda_k = \lambda_k^2$ в том случае, когда x^k не принадлежит области сверхлинейной сходимости метода Ньютона.

Теорема 7. Пусть функция $f(x)$ дважды непрерывно дифференцируема и существуют такие константы m и M , $0 < m \leq M < \infty$, что при $x \in G$ выполнено условие

$$m y^T y \leq y^T f''(x) y \leq M y^T y, \quad x \in R^n, y \in R^n,$$

Тогда для метода Ньютона решения задачи безусловной минимизации функции $f(x)$ имеет место следующая оценка коэффициента скорости сходимости:

$$g = \sqrt{1 - 4\varepsilon(1 - \varepsilon)(m/M)^3}, \quad \varepsilon \in (0, 1/2).$$

Доказательство. В условиях теоремы метод Ньютона градиентно согласован и $\alpha = 1/m$:

$$\|p^k\| \leq \|f^{n-1}(x^{k-1})\nabla f(x^{k-1})\| \leq \frac{1}{m}\|\nabla f(x^{k-1})\|.$$

Значение коэффициента $\beta = 1/M$ найдем из оценки:

$$-\langle p^k, \nabla f(x^{k-1}) \rangle = -\langle f^{n-1}(x^{k-1})\nabla f(x^{k-1}), \nabla f(x^{k-1}) \rangle \geq \frac{1}{M}\|\nabla f(x^{k-1})\|^2.$$

Тогда по теореме 2 при $0 < \varepsilon < 1/2$:

$$g^2 = 1 - 4\varepsilon(1-\varepsilon)\frac{m}{M}\left(\frac{\beta}{\alpha}\right)^2 = 1 - 4\varepsilon(1-\varepsilon)\left(\frac{m}{M}\right)^3.$$

Следовательно, метод Ньютона сходится линейно вне области его сверхлинейной сходимости.

Заключение

В статье предложена схема анализа сходимости градиентно-согласованных методов решения задач БМ. Достоинством ГСМ является то, что доказательство сходимости их к решению очень простое и применимо для многих известных методов БМ. Однако общность подхода не всегда позволяет учесть специфику свойств сходимости конкретных методов и оценки скорости их сходимости как ГСМ, хотя и являются оценками одного порядка (например, линейная скорость сходимости), но менее точны по сравнению с известными оценками.

Литература

1. *Сухарев А.Г., Тимохов А.В., Федоров В.В.* Курс методов оптимизации – М.: Физматлит, 2005.
2. *Жадан В.Г.* Методы оптимизации. Ч.2. Численные алгоритмы. – М.: МФТИ, 2015.
3. *Ортега Дж. and Рейнболт В.* Итерационные методы решения нелинейных систем уравнений со многими неизвестными. – М.: Мир, 1975.
4. *Нестеров Ю.Е.* Введение в выпуклую оптимизацию. – М.: МЦНМО, 2010.
5. *Поляк Б.Т.* Введение в оптимизацию. – Изд. 2-е, испр. и доп. – М.: ЛЕНАНД, 2014.
6. *Полак Э.* Численные методы оптимизации. – М.: Мир, 1974.

Получено 20.04.2017

СОДЕРЖАНИЕ

Предисловие.....	3
1. <i>В. И. Голубев, А. В. Фаворская.</i> Алгоритмы миграции Борна и Кирхгофа для упругих сред.....	4
2. <i>В. И. Голубев, Н. И. Хохлов, Д. П. Григорьевых.</i> Численный расчёт разрушения объектов с учётом изменения реологии.....	6
3. <i>Д. П. Григорьевых, Н. И. Хохлов.</i> Разрушение ледяного поля с помощью высокочастотной вибрации	8
4. <i>А. М. Иванов, Н. И. Хохлов.</i> Задание граничных условий RML для моделирования волн в упругой среде в случае осевой симметрии.....	11
5. <i>А. А. Скубачевский, Н. И. Хохлов.</i> Численный эксперимент по разрушению бетона посредством воздействия электромагнитного поля.....	14
6. <i>П. В. Стогний.</i> Моделирование волновых процессов в подледниковом озере Восток	19
7. <i>Д. М. Старков, И. С. Меньшиков.</i> Равновесие в информационных каскадах	24
8. <i>В. А. Дьяченко, И. С. Меньшиков.</i> Моделирование индивидуального поведения участников двойного аукциона с закрытыми ставками	34
9. <i>А. И. Шилова, И. С. Меньшиков.</i> Исследование игры Собственник–Директор–Агент	42
10. <i>Д. К. Бойко, И. С. Меньшиков.</i> Лабораторный анализ игровой модели многостороннего конфликта	52
11. <i>Д. А. Смаль.</i> Исследование парадокса Браесса методами экспериментальной экономики для транспортной сети с асимметричными затратами	65
12. <i>Б. В. Багно, Д. А. Подлесных.</i> Модель автоматической системы проверки лабораторных работ по курсу баз данных.....	74
13. <i>А. А. Шумилин</i> Программное средство для двухстадийной визуализации результатов численных экспериментов	76

14. *Д. Ю. Дербышев.* Ускорение сбора статистики с виртуальных машин внутри пакета libvirt78
15. *К. А. Щербатов.* AWS API-шифрование для Object Storage SDS-решения80
16. *Н. Н. Ефанов, П. В. Емельянов.* Построение формальной грамматики системных вызовов83
17. *В. В. Рязанов.* Поиск информативных подмножеств признаков и объектов в задачах распознавания по прецедентам90
18. *С. Л. Семаков, И. С. Семаков.* Анализ посадки с точки зрения её безопасности в гражданской авиации.....100
19. *А. А. Лагуновская.* О модификациях метода проекции градиента102
20. *А. Г. Бирюков, А. В. Чернов.* Градиентно-согласованные методы решения задач безусловной минимизации112

Редакторы: *В. А. Дружинина, И. А. Волкова.*

Корректоры: *О. П. Котова, Л. В. Себова*

Компьютерная верстка *Е. А. Казёнова*

Подписано в печать 18.07.2017. Формат $60 \times 84 \frac{1}{16}$. Усл. печ. л. 8,0.
Уч.- изд. л. 6,8. Тираж 150 экз. экз. Заказ № 161.

Федеральное государственное автономное образовательное учреждение
высшего образования «Московский физико-технический институт
(государственный университет)»

141700, Московская обл., г. Долгопрудный, Институтский пер., 9

Тел. (495) 408-58-22. E-mail: rio@mipt.ru

Отдел оперативной полиграфии «Физтех-полиграф»

141700, Московская обл., г. Долгопрудный, Институтский пер., 9

Тел. (495) 408-84-30. E-mail: polygraph@mipt.ru