ОЦЕНИВАНИЕ РИСКА ПРОГНОЗИРОВАНИЯ ОДНОРОДНЫХ КОНЕЧНЫХ ЦЕПЕЙ МАРКОВА

К. С. Матецкий

ПОСТАНОВКА ЗАДАЧИ

Цепи Маркова $x_t \in A$, $t \in \mathbf{N}$, где пространство состояний $A = \{0,1,...,N-1\}$ – конечное множество мощности $2 \le N < +\infty$, широко применяются при математическом моделировании сложных систем и процессов в экономике, технике, медицине, социологии, генетике и других приложениях [1–4]. На практике часто возникают задачи прогнозирования будущих значений цепи Маркова по наблюдаемой реализации и оценивания риска прогнозирования [3–4].

Рассмотрим случай, когда подлежащий прогнозированию временной ряд x_t, является однородной стационарной цепью Маркова с конечным пространством состояний A, стационарным распределением вероятностей $\pi = (\pi_0, \pi_1, ..., \pi_{N-1})' \in [0,1]^N$ и матрицей вероятностей одношаговых переходов $P = (p_{ij}) \in [0,1]^{N \times N}$:

$$\mathrm{P}\{x_1 = i\} = \pi_i,\ \mathrm{P}\{x_{t+1} = j \mid x_t = i\} = p_{ij},\ i, j \in A;\ \sum_{i \in A} \pi_i = 1;\ \sum_{j \in A} p_{ij} = 1,\ i \in A.$$

Временной ряд наблюдается в течение $T \ge 1$ последовательных единиц времени, и зарегистрирована реализация $X_1^T = (x_1, x_2, ..., x_T) \in A^T$.

Задача состоит в построении прогнозирующей статистики будущего значения $x_{T+\tau}$ по наблюдаемой реализации X_1^T и исследовании риска прогнозирования для (0-1)-функции потерь: $r(T,\tau) = P\{\hat{x}_{T+\tau} \ne x_{T+\tau}\}$.

алгоритмы прогнозирования

Теорема 1. Если x_t – однородная стационарная цепь Маркова с известными параметрами π, P, то оптимальная по критерию минимума вероятности ошибки (риска для (0-1)-функции потерь) прогнозирующая статистика имеет вид:

$$\hat{x}_{T+\tau} = \arg\max_{j \in A} p^{(\tau)}_{x_T, j}, \qquad (1)$$

где $p_{ij}^{(\tau)}$ – (i,j)-й элемент матрицы P^τ. При этом достигается минимум риска:

$$r_0(\tau) = \mathrm{P}\{\hat{x}_{T+\tau} \neq x_{T+\tau}\} = 1 - \sum_{i \in A} \pi_i \max_{j \in A} p_{ij}^{(\tau)} .$$

Если матрица вероятностей переходов P априорно не известна, то строится ее оценка максимального правдоподобия $\hat{P} = (\hat{p}_{ij})$ по наблюдаемой реализации X_1^T:

$$\hat{p}_{ij} = \begin{cases} \sum_{t=1}^{T-1} \delta_{x_t,i} \delta_{x_{t+1},j} \Big/ \sum_{t=1}^{T-1} \delta_{x_t,i}, \text{ если } \sum_{t=1}^{T-1} \delta_{x_t,i} > 0, \\ 1/N, \text{ если } \sum_{t=1}^{T-1} \delta_{x_t,i} = 0, \end{cases}$$

где $\delta_{ij} = \{1, i = j; 0, i \neq j\}$, $i, j \in A$, – символ Кронекера. Тогда «подстановочная» прогнозирующая статистика получается из (1):

$$\hat{x}_{T+\tau} = \arg\max_{j \in A} \left(\hat{P}^{\tau}\right)_{x_T, j} . \qquad (2)$$

ЗНАЧЕНИЕ РИСКА ПРОГНОЗИРОВАНИЯ

Введем обозначение: F_{ij}, $i, j \in A$ – множества $(N \times N)$-матриц с элементам из $\mathbf{N} \cup \{0\}$, удовлетворяющих условиям:

$$\sum_{u,v \in A} f_{uv} = T - 1, \; f_{u\cdot} - f_{\cdot u} = \delta_{ui} - \delta_{vj}, \; u \in A .$$

Теорема 2. В случае $\tau = 1$ риск прогнозирования равен

$$r(T, \tau) = 1 - \sum_{i,j,k \in A} \pi_i p_{jk} \sum_{\substack{F \in F_{ij}: \\ f_{jk} > f_{jl}, l \neq k}} \frac{\prod_{u \in A} f_{u\cdot}!}{\prod_{u,v \in A} f_{uv}!} F_{ji}^{*} \prod_{u,v \in A} p_{uv}^{f_{uv}} ,$$

где F_{ji}^{*} – алгебраическое дополнение к элементу (j,i) матрицы $F^{*} = (f_{uv}^{*})$ с элементами

$$f_{uv}^{*} = \begin{cases} \delta_{uv} - f_{uv} / f_{u\cdot}, & \text{если } f_{u\cdot} > 0, \\ \delta_{uv}, & \text{если } f_{u\cdot} = 0. \end{cases}$$

ВЕРХНЯЯ ОЦЕНКА РИСКА ПРОГНОЗИРОВАНИЯ

По наблюдаемой цепи Маркова x_t построим новый дискретный временной ряд $y_t = (x_t, x_{t+1})$, $t \geq 1$; y_t является стационарной цепью Марко-

ва с матрицей вероятностей одношаговых переходов $\overline{P} = (\overline{p}_{(ij)(kl)})$ и вектором стационарных распределений $\overline{\pi} = (\overline{\pi}_{(ij)})$: $\overline{p}_{(ij)(kl)} = \delta_{jk} p_{kl}$, $\overline{\pi}_{(ij)} = \pi_i p_{ij}$, $i, j, k, l \in A$.

Теорема 3. Если в каждой i-ой строке матрицы P существует единственный максимальный элемент с номером $i^* \in A$, то справедлива верхняя асимптотическая оценка риска прогнозирования:

$$r(T,1) \le r_0 + \left(2^{N-1} - 1\right) \sum_{i \in A} \pi_i p_{ii^*} \max_{k \in A \setminus \{i^*\}} \{\nu_{(ik)(ii^*)}\} \frac{1}{T} + O\left(\frac{1}{T^2}\right), \qquad (3)$$

где $\nu_{(ik)(ii^*)} = (2\overline{\pi}_{(ik)}(z_{(ik)(ik)} - z_{(ik)(ii^*)}) + 2\overline{\pi}_{(ii^*)}(z_{(ii^*)(ii^*)} - z_{(ii^*)(ik)}) - (\overline{\pi}_{(ik)} +$ $+ \overline{\pi}_{(ii^*)}) - (\overline{\pi}_{(ik)} - \overline{\pi}_{(ii^*)})^2) / (\overline{\pi}_{(ik)} - \overline{\pi}_{(ii^*)})^2$, $z_{(ij)(kl)}$ – $((i,j),(k,l))$-й элемент фундаментальной матрицы Z цепи Маркова y_t.

АППРОКСИМАЦИЯ РИСКА ПРОГНОЗИРОВАНИЯ НА ОСНОВЕ ЦПТ

Теорема 4. В случае $N \ge 2$, $\tau \ge 1$, если $(N-1) \times (N-1)$-матрицы A_{ij}, $i, j \in A$, с элементами

$$(A_{ij})_{kl} = \sum_{t=0}^{\tau-1} p_{ii}^{(t)} \left(1 + p_{lk}^{(\tau-t-1)} - p_{lj}^{(\tau-t-1)}\right), \; k, l \in A \setminus \{j\},$$

невырождены, то для риска прогнозирования статистики (2) верна сходимость:

$$r(T,\tau) - r_+(T,\tau) \xrightarrow{T \to +\infty} 0,$$

где $r_+(T,\tau) = 1 - \sum_{i,j \in !} \pi_i p_{ij}^{(\tau)} \int_0^{+\infty} \dots \int_0^{+\infty} n_{N-1}\left(x \middle| \mu_{ij}^{(\tau)}, \frac{1}{T-1} \Xi_{ij}^{(\tau)}\right) dx_0 \dots dx_{N-2}$, n_{N-1} – плотность $(N-1)$-мерного нормального закона распределения с параметрами $\mu_{ij}^{(\tau)} = \left(p_{ij}^{(\tau)} - p_{ik}^{(\tau)}\right) \in \mathbf{R}^{N\text{-}1}$, $k \in A \setminus \{j\}$, и $\Xi_{ij}^{(\tau)} = A_{ij} \Sigma_{ij} A_{ij}'$, $\Sigma_{ij} = (\sigma_{kl}) \in \mathbf{R}^{(N-1)\times(N-1)}$, $\sigma_{kl} = \frac{1}{\pi_i} p_{ik}(\delta_{kl} - p_{il})$, $k, l \in A \setminus \{j\}$.

Из теоремы 4 следует, что функцию $r_+(T,\tau)$ можно использовать в качестве аппроксимации риска прогнозирования цепи Маркова.

ЧИСЛЕННЫЕ ЭКСПЕРИМЕНТЫ

В ходе численных экспериментов иллюстрировались полученные результаты. Значения параметров: N=2,

$$P = \begin{pmatrix} 0.1 & 0.9 \\ 0.9 & 0.1 \end{pmatrix}, \ \pi = \begin{pmatrix} 0.5 \\ 0.5 \end{pmatrix}.$$

Моделировались цепи Маркова, затем методом Монте-Карло оценивался риск прогнозирования их будущих значений при горизонте прогнозирования $\tau = 1$, вычислялись оценки $r^+(T) = r_0 + A/T$ (где константа A определена в правой части (3)). Число прогонов в методе Монте-Карло $K=10^5$. Результаты экспериментов приведены на рисунке 1.

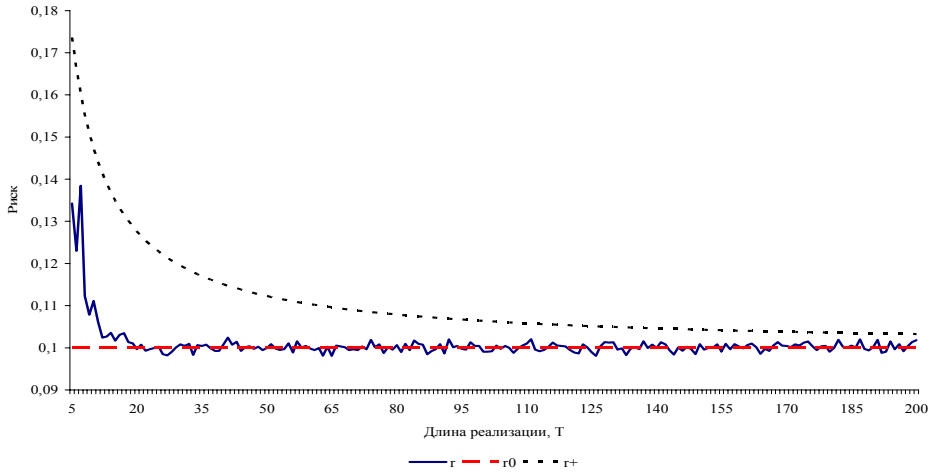

Рис.1. Графики оценок риска

В ходе численных экспериментов получено, что точность главного члена оценки (3) существенно зависит от матрицы P. Эта точность ниже, если матрица вероятностей переходов P имеет доминирующие диагональные элементы, по сравнению со случаем, когда диагональные элементы не являются доминирующими.

Литература

1. *Basawa I. V., Rao B. L. C.* Statistical interference for stochastic processes. N.Y. // Academic Press. 1980.
2. *Дуб Дж*. Вероятностные процессы. // М. ИЛ. 1956.
3. *Харин Ю. С.* Оптимальность и робастность в статистическом прогнозировании. // Мн. БГУ. 2008.
4. *Харин Ю. С.* Вероятностно-статистический анализ цепей Маркова высокого порядка // Вестн. Белорус. ун-та. Сер. 1. 2006. N 3. С. 80–86.

ТЕСТИРОВАНИЕ WEB-ПРИЛОЖЕНИЙ

А. Г. Морейнис

В настоящее время все большее распространение получают приложения, взаимодействующие с пользователем посредством Web-интерфейса.

Поэтому актуальным вопросом для Web-приложений является обеспечение качества работы, т.е. программы должны выдавать те страницы, которые ожидает увидеть пользователь. Таким образом, необходимо решать задачу функционального тестирования, например, с помощью нагрузочного тестирования. Большинство Web-приложений постоянно развиваются и модифицируются, поэтому важным является наличие регрессионных тестов для того, чтобы удостовериться, что результаты внесения изменений не нарушают функциональности приложения. Непосредственное тестирование Web-приложений человеком отнимает много времени и в случае больших приложений малоэффективно, т.е. переход по различным ссылкам внутри приложения и анализа отображаемых страниц, работа с большим количеством входных параметров, работающих с базами данных сложной структуры и с большим количеством записей. При разработке таких программных продуктов вопрос об автоматизации процесса их тестирования стоит особенно остро.

Одними из самых распространенных видов тестирования Web-приложений являются [4, 5]:

- Тестирование Web-приложений «на проникновение» основано на проверке ввода на корректность.
- Функциональное тестирование или тестирование черного ящика– это тестирование программного обеспечения в целях проверки реализуемости функциональных требований, т.е. способности программного обеспечения в определенных условиях решать задачи, которые нужны пользователям.
- Использование Selenium для тестирования Web-приложений на различных браузерах представляет собой инструментальное средство приемочного тестирования, тесты которого выполняются непосредственно в браузере.
- Тестирование Web-приложений с помощью Ruby – это высокоуровневая методика тестирования Web-приложений, которая используется как для системного, так и для приемочного тестирования. Ключевую роль при этом играет Ruby-библиотека Application Testing In Ruby, которая позволяет запрограммировать действия браузера Internet Explorer на языке Ruby так, чтобы можно было автоматизировать значительную часть ручной работы тестеров по заполнению форм, переходу по ссылкам и т.д.
- Нагрузочное тестирование Web-приложений – это запись трафика, который генерируется при общении локального компьютера с сервером, а затем использует записанный трафик для эмуляции действий пользователей.

- Приемочное тестирование основано на тестировании группы связанных классов, которые создают желаемый результат;
- Применение инструментов браузеров Internet Explorer, Mozila Firefox, Opera, Google Chrome ориентированы на Web-разработчиков для создания, редактирования и анализа Web-страниц.
- Проанализировав различные виды тестирования Web-приложений, мною было разработано тестирующее Web-приложение, которое предназначено для Web-разработчиков с целью оптимизации рабочего процесса. Разработанное Web-приложение основано на следующем:
- Структуру любого Web-приложения можно представить в виде дерева или любой другой древовидной структуры;
- Для анализа функциональности Web-приложения потребуется следующее [3]:
    - кнопки;
    - гипертексты;
    - картинки;
    - JavaScript;
    - URL;
    - links;
    - формы.
- Информация страницы содержит следующее:
    - все возможные пути переходов с текущей страницы;
    - информация о странице;
    - структурное содержание;
    - время обработки;
    - пройденные и не пройденные тесты.
- Настройки:
    - ограниченное или неограниченное количество тестов;
    - точка входа;
    - предмет тестирования;
    - корректные и некорректные параметры;
    - установка маршрута для тестирования конкретной области web-приложения;
    - тестирование определенных страниц;
    - задание количества пользователей с личными настройками.

Выше изложенный подход тесно пересекается с тестированием Web-приложений «на проникновение», с помощью Ruby, с использованием Selenium, с помощью инструментов браузеров Internet Explorer, Mozila Firefox, Opera, Google Chrome посредством ссылок, полей для ввода, гипертекстов, форм.

Для реализации проекта я использовал Ext GWT [1, 2], что позволило ускорить разработку проекта, уменьшило затраты на разработку интерфейса. Проект разделен на две части: серверную и клиентскую, взаимодействие между которыми происходит в асинхронном режиме, что позволяет улучшить передачу данных и показать работоспособность программы разработчику. Для сохранения и получения важной информации используется MySQL, в котором хранится конфиденциальная информация пользователей сайтов. Разработанное Web-приложение позволяет программисту ускорить разработку web-приложения, т.к. помогает найти ошибки на ранних этапах разработки, что увеличивает устойчивость и уменьшает стоимость проекта. Также Web-приложение позволяет получить абстрактное представление о структуре разрабатываемого или тестируемого web-приложения, которое даёт лучшее представление и отделяет разработку от тестирования. Так разработчику не придётся тратить время на ожидание работы сотрудника по тестированию Web-приложения и избавляет самого от тестирования, что позволяет сконцентрироваться на разработке, устранении неполадок. В результате автоматизации процесса тестирования Web-приложений устраняется человеческий фактор в поиске существующих ошибок, что позволяет сократить время разработки и поддержки самого проекта и отпадает необходимость привлечения дополнительных ресурсов.

При тестировании Web-приложений необходимо грамотно выбирать инструменты для их тестирования, поскольку этот выбор сделает более эффективной работу, как разработчика, так и сотрудника по тестированию Web-приложений.

С моей точки зрения, тестирование web-приложений “на проникновение”, с помощью Ruby, с использованием Selenium, с помощью инструментов браузеров IE, Mozilla Firefox, Opera, Google Chrome лучше всего подходят для тестирования Web-приложения на клиентской стороне на начальном этапе, что позволяет без перегрузки сервера сократить время на исправление, добавление информации в Web-приложении. А функциональное, нагрузочное, приёмочное виды тестирования больше всего подходят для серверной части Web-приложения и должны применяться на первых этапах разработки, т.к. правильная разработка на данных этапах не приводит к увеличению стоимости проекта и появлению более серьёзных ошибок.

Разработанное Web-приложение тесно связано с тестированием Web-приложений «на проникновение», с помощью Ruby, с использованием Selenium, с помощью инструментов браузеров IE, Mozilla Firefox, Opera, Google Chrome.

Литература

1. *Grant K. S*. Developing with Ext GWT Enterprise RIA Development. Apress. 2009.
2. *Dewsbury R*. Google Web Toolkit. Prentice Hall. 2007.
3. *Фридл Дж*. Регулярные выражения. Символ-Плюс. 2008.
4. *Брайан А*. Тестирование и оптимизация веб-сайтов: руководство по Google Website Optimizer. Диалектика. 2009.
5. *Стотлемайер Д.* Тестирование Web-приложений. КУДИЦ-Образ. 2003.

ЗАДАЧА ОПТИМАЛЬНОГО ПОКРЫТИЯ ИЗМЕНЕННЫХ ДАННЫХ

Столяров В. О., Шавлак М. Ю.

ВВЕДЕНИЕ

Задача оптимального покрытия измененных данных рассматривается при передаче модифицированного изображения с экрана одного устройства на экран другого устройства и является частным случаем задачи покрытия множества. Для ее решения были рассмотрены следующие алгоритмы:

- Сеточный алгоритм,
- Жадный алгоритм.

Введем определение эффективности решения. Под эффективностью здесь понимается совокупность атрибутов решения: интерактивность, рациональное использование канала передачи данных (оптимальность сжатия), экономия трафика (минимизация данных на отправку).

Для построения эффективного решения требуется решить две алгоритмические проблемы. Первая – как эффективно покрыть изменения на экране, произошедшие за определенное время прямоугольниками пикселей. Критерий эффективности в данном случае – количество прямоугольников. Задача алгоритма – минимизировать это количество. Вторая проблема – как эффективно сжать эти прямоугольники перед отправкой на устройство клиента. В этом случае критерий эффективности – баланс между степенью сжатия и временем сжатия/распаковки.

Постановка алгоритмической задачи

В результате очередного обновления экрана перед серверным приложением ставится задача оптимального покрытия изменённых точек прямоугольниками, в общем случае известная как “задача о минимальном покрытии множества”. Формулируется она следующим образом. Пусть даны множество $M = \{1,...,m\}$ и набор его подмножеств $M_1,...,M_n$ таких,

что $\bigcup_{j=1}^{n} M_j = M$. Совокупность подмножеств $M_j, j \in J \subseteq \{1,...,n\}$, называется покрытием множества M, если $\bigcup_{j \in J} M_j = M$. Каждому M_j приписан вес $c_j \geq 0$. Требуется найти покрытие минимального суммарного веса. Задача называется невзвешенной, если все подмножества M_j имеют единичные веса.

В общем виде задача о покрытии NP-трудна, поэтому логично предположить, что следует искать приближенные алгоритмы полиномиальной сложности, доставляющие решения со значениями целевой функции, близкими к оптимуму. Рассмотрим некоторые примеры приближённых алгоритмов для решения поставленной задачи.

СЕТОЧНЫЙ АЛГОРИТМ

Самый простой на первый взгляд и, тем не менее, достаточно эффективный алгоритм, разработанный в настоящей работе, заключается в разбиении экрана на множество непересекающихся прямоугольников по сетке. Очевидно, что при таком выборе семейства множеств M_j, существует единственное покрытие M. Алгоритм необычайно прост в реализации: достаточно посетить каждую точку единожды, чтобы построить покрывающее множество. Таким образом, он имеет минимальную возможную сложность O(NxM).

Быстрота работы делает его хорошим выбором на высокоскоростных соединениях, где время, затраченное на вычисления может быть больше задержки передачи данных по сети. Алгоритм в худшем случае даёт четырёхкратный проигрыш по сравнению с оптимальным решением. Однако эта граница редко достигается на практике. На рисунке 1 показано сеточное покрытие текстовых изменений (ввод A,B,C).

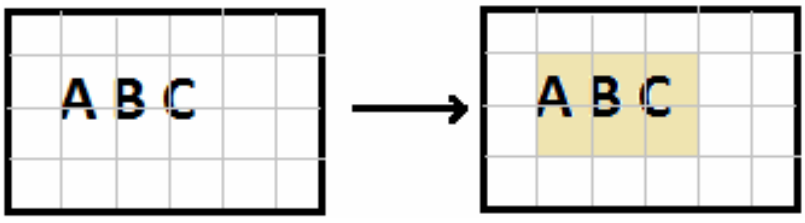

Рис. 1. Схема работы сеточного алгоритма

ЖАДНЫЙ АЛГОРИТМ

Рассмотрим жадный алгоритм для покрытия изменившихся точек экрана размером MxN квадратами шириной W. Идея алгоритма проста: на каждой итерации выбирать квадраты с максимальным количеством покрываемых точек. На рисунке 2 представлена демонстрация его работы на измененной области в форме треугольника.

Рис. 2. Работа жадного алгоритма

В [1] показано, что относительная погрешность $\delta \le 1 + \ln \max_{j=1,..,n} |M_j|$. Также было доказано, что в невзвешенном случае для этого алгоритма справедливо неравенство $\delta \le \ln m - \ln \ln m + 0{,}78$, что близко к известной нижней оценке $\delta \ge (1-\varepsilon)\ln m$.

Решение можно разбить на два этапа: построение множества квадратов с ассоциированным весом и сортировка их по убыванию. Весом c(A) квадрата A назовём количество точек, которые он покрывает. Как и в случае с кэшированием, функция c(A) обладает свойством, позволяющим за один проход по экрану вычислить её для всех квадратов.

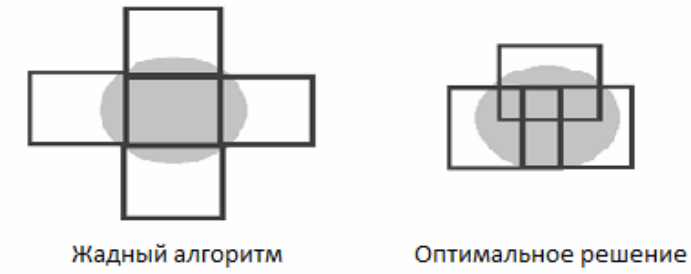

Рис. 3. Сравнение результатов работы жадного алгоритма и оптимального решения

Таким образом, сложность построения множества квадратов порядка O(N*M). В результате также получим количество квадратов порядка O(N*M). Сортировка их по убыванию в среднем O(N*M*log(N*M)). Как альтернатива, можно использовать бинарное дерево поиска для сортировки, однако этот подход не даст нам уменьшения алгоритмической сложности задачи. Выборку из отсортированной последовательности можно произвести за время порядка O(N*M). Таким образом, алгоритм

даст нам результат за время порядка O(N*M*log(N*M)), что достаточно эффективно для целей, поставленных в настоящей работе, однако алгоритм не лишён недостатков. На рисунке 3 представлен случай, когда такой алгоритм даёт неоптимальный результат.

ДОПОЛНИТЕЛЬНЫЕ ОПТИМИЗАЦИИ

В качестве дополнительной оптимизации был использован кэш. В кэше находятся наиболее часто используемые измененные данные. В него попадают все изменения, но остаются только наиболее часто используемые. Были расмотрены две реализации кэша контекстно-зависимая и контекстно-инвариантная. В контекстно-зависимой реализации измененные данные покрываются прямоугольными множествами, и далее выполняется поиск вхождения этих покрытий в кэше. Эта реализация отвечает требованиям высокой эффективности по скорости работы кэша. В контекстно-независимой реализации строится максимальное покрытие измененных данных существующими элементами в кэше. Несмотря на то, что построение этого покрытия требует дополнительных вычислительных ресурсов, эта реализация дает максимально возможное использование данных из кэша, что отвечает требованиям минимизации отправляемых данных.

ЗАКЛЮЧЕНИЕ

В работе реализованы оба алгоритма и протестированы в контексте задачи удаленного управления компьютером посредством портативных устройств, произведена оценка их эффективности. К преимуществам сеточного алгоритма можно отнести простоту реализации; он имеет наименьшую сложность из всех алгоритмов покрытия. Но недостатком является то, что в худшем случае он демонстрирует результат в четыре раза хуже оптимального решения.

Жадный алгоритм более сложен в реализации, но в условиях ограниченной пропускной способности сети он более эффективен, хотя в некоторых случаях явно уступает сеточному в построении наиболее компактного покрытия.

Литература

1. *Еремеев А. В., Заозерская Л. А. ,Колоколов А. А.* Задача о покрытии множества: сложность, алгоритмы, экспериментальные исследования //Дискретный анализ и исследование операций. Сер. 2. 2000. № 2. С. 22–46.

ВЛИЯНИЕ ФУНКЦИОНАЛЬНЫХ ИСКАЖЕНИЙ РАСПРЕДЕЛЕНИЯ НАБЛЮДЕНИЙ НА ХАРАКТЕРИСТИКИ ПОСЛЕДОВАТЕЛЬНОГО КРИТЕРИЯ

С. Ю. Чернов

Последовательный подход [1] используется для статистического решения многих практических задач, в которых имеется необходимость проверки гипотетических предположений о параметрах исследуемого процесса. При выполнении модельных предположений такой подход требует в среднем меньшее число наблюдений для принятия решений среди всех возможных статистических критериев с такими же значениями вероятностей ошибок. Однако на практике искажения в предполагаемой модели могут оказать существенное влияние на значения вероятностных характеристик последовательных критериев. В [3] построен минимаксный робастный последовательный критерий для дискретного распределения вероятностей наблюдений. В [4] проведен анализ робастности, когда имеют место “засорения”, предложенные Тьюки и Хьюбером [2]. В данной работе закон распределения наблюдений отличается от распределения, используемого при построении теоретической модели, а в качестве меры различия используется L_1 метрика. При указанных предположениях проведен анализ робастности вероятностных характеристик ПКОВ проверки двух простых гипотез и предложен способ повышения робастности ПКОВ.

Пусть на измеримом пространстве $(\Omega, \Im)$ наблюдается последовательность независимых одинаково распределенных случайных величин $x_1, x_2, ... \in \mathbf{R}$, имеющих плотность распределения вероятностей $f(x;\theta)$ с параметром $\theta \in \Theta = \{\theta_0, \theta_1\}$, истинное значение которого неизвестно. Обозначим функцию распределения вероятностей наблюдений x_i, $i = 1,2,\ldots$, через $F(x,\theta)$; $\Lambda_n = \Lambda_n(x_1,...x_n) = \sum_{t=1}^{n} \lambda_t$, где

$$\lambda_t = \lambda(x_t) = \ln\big(f(x_t,\theta_1)/f(x_t,\theta_0)\big). \quad (1)$$

Относительно параметра θ имеются две простые гипотезы $H_0: \theta = \theta_0$, $H_1: \theta = \theta_1$. Для проверки данных гипотез используется последовательный критерий отношения вероятностей (ПКОВ) [1]:

$$N = \min\{n \in \mathrm{N} : \Lambda_n \notin (C_-, C_+)\}, \quad (2)$$

$$d = 1_{[C_+,+\infty)}(\Lambda_N), \quad (3)$$

где N – случайный момент остановки, после которого принимается решение d в соответствии с (3). В (2), (3) $C_-, C_+ \in \mathbf{R}$, $C_- < C_+$ – заданные параметры критерия, называемые порогами. На практике для их задания пользуются соотношениями [1] $C_- = \ln\left(\beta_0/(1-\alpha_0)\right)$, $C_+ = \ln\left((1-\beta_0)/\alpha_0\right)$, где $\alpha_0, \beta_0 \in (0,1)$ – величины, близкие к приемлемым значениям вероятностей ошибок I и II рода.

Сформулируем следующие предположения: П1) функция $f(x,\theta)$ имеет конечные производные 1-го и 2-го порядка по переменной x, а также $f(x,\theta) \neq 0$, $\theta \in \Theta$; П2) функция $\lambda(x)$, определенная (1), строго монотонна по переменной x, а также имеет отличную от нуля производную 1-го порядка.

Без ограничения общности будем считать, что истинной гипотезой является H_0 (случай H_1 рассматривается аналогично).

Для оценивания вероятностей ошибочных решений последовательного критерия (2), (3) воспользуемся подходом, изложенным в [5]. Разобьем интервал (C_-, C_+) на m промежутков длиной $h = (C_+ - C_-)/m$, $m \in \mathbf{N}$ – параметр разбиения (аппроксимации). Введем случайные последовательности ($i \in \mathbf{N}$):

$$\Lambda_n^{\pm} = \sum_{t=1}^{n} \lambda_t^{\pm};\ \lambda_1^- = C_- + \left[\frac{\lambda_1 - C_-}{h}\right]h,\ \lambda_t^- = \left[\frac{\lambda_t}{h}\right]h,\ t \geq 2;\ \lambda_i^+ = \lambda_i^- + h\,.$$

Построим поглощающие цепи Маркова L_n^-, L_n^+ со множеством значений $\{0, 1, ..., m, m+1\}$ и поглощающими состояниями 0 и $m+1$:

$$L_n^- = \begin{cases} 0, & \Lambda_n^- \in (-\infty, C_- - h], \\ i, & \Lambda_n^- = C_- + (i-1)h, \quad i = \overline{1,m}, \\ m+1, & \Lambda_n^- \in [C_+, \infty), \end{cases} \quad L_n^+ = \begin{cases} 0, & \Lambda_n^+ \in (-\infty, C_-], \\ i, & \Lambda_n^+ = C_- + ih, \quad i = \overline{1,m}, \\ m+1, & \Lambda_n^+ \in [C_+ + h, \infty). \end{cases}$$

В соответствии с [5], векторы вероятностей начальных состояний и матрицы вероятностей переходов цепей Маркова L_n^- и L_n^+ могут быть записаны в явном виде.

Пусть α^- и α^+ – вероятности поглощения цепей Маркова L_n^- и L_n^+ в состоянии $(m+1)$. В [5] показано, что вероятности α^-, α и α^+ удовлетворяют неравенству $\alpha^- \leq \alpha \leq \alpha^+$ и соотношению при $h \to 0$ $\alpha^+ - \alpha^- = O(h)$. Поэтому в качестве точечного приближения неизвестного значения α выбирается $\hat{\alpha}_m = \left(\alpha^+ + \alpha^-\right)/2$, причем

$|\alpha - \hat{\alpha}_m| \le \left(\alpha^+ - \alpha^-\right)/2$. В дальнейшем вместо вероятности α будем анализировать величины α^- и α^+.

Пусть наблюдения x_n, $n \in \mathrm{N}$, имеют плотность распределения вероятностей $h(x,\theta)$, которая может отличаться от теоретической плотности распределения вероятностей $f(x,\theta)$. Однако известно, что расстояние в L_1 метрике между $h(x,\theta)$ и $f(x,\theta)$ не превышает ε:

$$\int_{\mathrm{R}} |h(x,\theta) - f(x,\theta)|\, dx \le \varepsilon, \tag{4}$$

где $0 \le \varepsilon \le \varepsilon_0$, причем величина ε_0 задается заранее. Множество плотностей распределения вероятностей $h(x,\theta)$, удовлетворяющих (4) при фиксированном ε, обозначим $L_1(f,\varepsilon)$.

Цепи Маркова L_n^- и L_n^+ в случае, когда наблюдения имеют плотность распределения вероятностей $h(\cdot,\theta)$, обозначим соответственно $L_n^-(h)$ и $L_n^+(h)$. Пусть $\alpha^-(h,\varepsilon)$ и $\alpha^+(h,\varepsilon)$ – вероятности поглощения цепей Маркова $L_n^-(h)$ и $L_n^+(h)$ в состоянии $m+1$.

Пусть $g_- = F^{-1}\left(\dfrac{\varepsilon}{2}\right)$, $g_+ > \lambda^{-1}((m-1)h)$. Обозначим

$$\bar{f}(x,\theta) = 1_{(g_-,+\infty)}(x) f(x,\theta) + \frac{\varepsilon}{2}\delta(x - g_+), \tag{5}$$

где $\delta(\cdot)$ – дельта-функция Дирака [6].

Теорема 1. Если для искаженной модели наблюдений (4) выполнены предположения П1 и П2, то величины $\alpha^+(h,\varepsilon)$, удовлетворяют неравенству $\alpha^+(h,\varepsilon) \le \alpha^+(\bar{f},\varepsilon)$.

Следствие 1. Вероятность ошибки первого рода $\alpha^+(\bar{f},\varepsilon)$ монотонно возрастает по переменной ε, в частности, для любого ε, $0 \le \varepsilon \le \varepsilon_0$, выполняется неравенство $\alpha^+(\bar{f},\varepsilon) \le \alpha^+(\bar{f},\varepsilon_0)$.

Рассмотрим плотность распределения вероятностей

$$h^g(x,\theta) = 1_{[g_-,g_+]}(x) h(x,\theta) + \varepsilon_- \delta(x - g_-) + \varepsilon_+ \delta(x - g_+), \tag{6}$$

где g_- и g_+ – заданные параметры усечения наблюдения, имеющего плотность распределения вероятностей $h(x,\theta)$ и функцию распределения вероятностей $H(x,\theta)$, $\varepsilon_- = H(g_-,\theta)$, $\varepsilon_+ = 1 - H(g_+,\theta)$.

Пусть

$$\bar{f}^g(x,\theta) = 1_{[g_-,g_+]} f(x,\theta) + (\varepsilon_- - \varepsilon/2)\,\delta(x-g_-) + (\varepsilon_+ - \varepsilon/2)\,\delta(x-g_+).$$

Если $h \in L_1(f,\varepsilon)$, то $h^g \in L_1(f^g,\varepsilon)$, и следовательно, $\bar{f}^g \in L_1(f^g,\varepsilon)$.

Теорема 2. Если для искаженной модели наблюдений (6) выполнены предположения П1 и П2, то величины $\alpha^+(h^g,\varepsilon)$, удовлетворяют неравенству $\alpha^+(h^g,\varepsilon) \le \alpha^+(\bar{f}^g,\varepsilon)$.

Следствие 2. Вероятность ошибки первого рода $\alpha^+(\bar{f}^g,\varepsilon)$ монотонно возрастает по переменной ε, в частности, для любого ε, $0 \le \varepsilon \le \varepsilon_0$, выполняется неравенство $\alpha^+(\bar{f}^g,\varepsilon) \le \alpha^+(\bar{f}^g,\varepsilon_0)$.

Литература

1. *Вальд А*. Последовательный анализ. М. 1960.
2. *Хьюбер П*. Робастная статистика. М. 1984.
3. *Kharin A, Kishylau D*. Robust sequential testing of hypotheses on discrete probability distributions // Austrian Journal of Statistics. V. 34. 2005. № 2. P. 153-162.
4. *Charnou S*. Sequential test robustifications for simple hypothesis under outliers. // Abstracts of the International Conference on Robust Statistics. Parma. 2009. P. 22.
5. *Kharin A., Chernov S*. Error Probabilities Evaluation for Sequential Testing of Simple Hypotheses on Data from Continuous Distribution // Proc. of the Pattern Recognition and Information Processing (PRIP). Minsk. 2009. P. 63–66.
6. *Кудрявцев Л. Д*. Курс математического анализа. М. 1981.

ОБ ОЦЕНКЕ ПАРАМЕТРА ПОЛОЖЕНИЯ α – УСТОЙЧИВЫХ РАСПРЕДЕЛЕНИЙ

Чэнь Хайлун

На основании свойства устойчивых распределений доказано равенство, для получения преобразования с α – устойчивым распределением. Методом характеристических функций (CF) получаем оценки параметров α и σ, затем с помощью преобразования предлагается метод оценки параметра положения μ α – устойчивого распределения при $\alpha \in (0;2]$.

1. МЕТОД CF ДЛЯ ОЦЕНКИ ПАРАМЕТРОВ.

Случайная величина X называется устойчивой, если ее логарифм характеристической функции имеет вид:

$$\psi(\theta) = \ln(\varphi(\theta)) = \begin{cases} -\sigma^\alpha |\theta|^\alpha + i\left(\mu\theta + \sigma^\alpha |\theta|^\alpha \beta\, sign(\theta)\tan\dfrac{\pi\alpha}{2}\right), \alpha \ne 1, \\ -\sigma|\theta| + i\left(\mu\theta - \sigma|\theta|\beta\dfrac{2}{\pi} sign(\theta)\ln|\theta|\right), \quad \alpha = 1. \end{cases} \tag{1}$$

где $\varphi(\theta) = E\exp[i\theta X]$, $\alpha \in (0,2]$, $\beta \in [-1,1]$, $\sigma > 0$, $\mu \in R$, $\theta \in R$. В этом случае будем писать $X \sim S_\alpha(\sigma,\beta,\mu)$. Если в соотношении (1) $\beta = 0$, то устойчивые распределения называются симметричными.

Основная идея метода CF состоит в том, чтобы по выборочным данным оценить характеристическую функцию. Затем используя действительную и мнимую часть логарифма характеристической функции оценить α, σ и β, μ. Однако, в связи с тем, что характеристическая функция α – устойчивых распределений меняет вид при $\alpha = 1$, то трудно получить точную оценку параметра μ (параметра положения).

2. МЕТОД ОЦЕНКИ ПАРАМЕТРА ПОЛОЖЕНИЯ.

В настоящее время в литературе мало методов для оценки параметра μ. Когда $\alpha \to 1$ и $\beta \neq 0$, то сложно оценить μ, и даже в случае $\mu = 0$, нет хорошего способа для точного оценивания μ. В [1] даётся подходящая оценка параметра μ в случае $\alpha \in (1,2]$. Чтобы достичь подходящей оценки параметра μ для случая $\alpha \in (0,2]$, на основании свойства устойчивых распределений [1], используется следующая теорема:

Теорема [2]. Пусть Xk – независимые одинаково распределенные устойчивые случайные величины с параметрами α, σ, β, μ, т.е. Xk ~ $S\alpha(\sigma,\beta,\mu)$ тогда:

$$Z = \sum_{k=1}^{n} a_k X_k \sim S_\alpha\left(\left(\sum_{k=1}^{n}|a_k|^\alpha\right)^{1/\alpha}\sigma, \frac{\sum_{k=1}^{n} a_k^{\langle\alpha\rangle}}{\sum_{k=1}^{n}|a_k|^\alpha}\beta, \sum_{k=1}^{n} a_k\mu\right),$$

где $a_k^{\langle\alpha\rangle} = sign(a_k)|a_k|^\alpha$.

Из теоремы могут быть выведены 3 типа преобразований: XC – центрирующее преобразование, XD – выравнивающее преобразование, XS – симметричное преобразование.

Лемма 1. Пусть Xk ~ $S\alpha(\sigma,\beta,\mu)$, $k = \overline{1,n}$ и $X_k^C \sim X_{3k} + X_{3k-1} - 2X_{3k-2}$, тогда:

$$X_k^C \sim S_\alpha\left(\left(2+2^\alpha\right)^{1/\alpha}\sigma, \left(\frac{2-2^\alpha}{2+2^\alpha}\right)\beta, 0\right);$$

Доказательство. В типе преобразования XC $a_1 = 1$, $a_2 = 1$, $a_3 = -2$, то

$$\left(\sum_{k=1}^{3}|a_k|^{\alpha}\right)^{1/\alpha}\sigma=\left(1^{\alpha}+1^{\alpha}+|-2|^{\alpha}\right)^{1/\alpha}\sigma=\left(2+2^{\alpha}\right)^{1/\alpha}\sigma,$$

$$\frac{\sum_{k=1}^{3}a_k^{\langle\alpha\rangle}}{\sum_{k=1}^{3}|a_k|^{\alpha}}\beta=\frac{1^{\alpha}+1^{\alpha}-|-2|^{\alpha}}{1^{\alpha}+1^{\alpha}+|-2|^{\alpha}}\beta=\frac{2-2^{\alpha}}{2+2^{\alpha}}\beta,$$

$$\sum_{k=1}^{3}a_k\mu=(a_1+a_2+a_3)\mu=(1+1-2)\mu=0.$$

Лемма доказана.

Лемма 2. Пусть Xk ~ Sα(σ,β,μ), $k=\overline{1,n}$ и $X_k^D\sim X_{3k}+X_{3k-1}-2^{1/\alpha}X_{3k-2}$, тогда:

$$X_k^D\sim S_{\alpha}\left(4^{1/\alpha}\sigma,0,\left(2-2^{1/\alpha}\right)\mu\right).$$

Доказательство. В типе преобразования XD $a_1 = 1$, $a_2 = 1$, $a_3 = -2^{1/\alpha}$, то

$$\left(\sum_{k=1}^{3}|a_k|^{\alpha}\right)^{1/\alpha}\sigma=\left(1^{\alpha}+1^{\alpha}+\left|-2^{1/\alpha}\right|^{\alpha}\right)^{1/\alpha}\sigma=4^{1/\alpha}\sigma,$$

$$\frac{\sum_{k=1}^{3}a_k^{\langle\alpha\rangle}}{\sum_{k=1}^{3}|a_k|^{\alpha}}\beta=\frac{1^{\alpha}+1^{\alpha}-\left|-2^{1/\alpha}\right|^{\alpha}}{1^{\alpha}+1^{\alpha}+\left|-2^{1/\alpha}\right|^{\alpha}}\beta=\frac{2-2}{2+2}\beta=0,$$

$$\sum_{k=1}^{3}a_k\mu=(a_1+a_2+a_3)\mu=\left(1+1-2^{1/\alpha}\right)\mu=\left(2-2^{1/\alpha}\right)\mu.$$

Лемма доказана.

Лемма 3. Пусть Xk ~ Sα(σ,β,μ), $k=\overline{1,n}$ и $X_k^S\sim X_{2k}-X_{2k-1}$, тогда:

$$X_k^S\sim S_{\alpha}\left(2^{1/\alpha}\sigma,0,0\right).$$

Доказательство. В типе преобразования XS $a_1 = 1$, $a_2 = -1$, то

$$\left(\sum_{k=1}^{2}|a_k|^{\alpha}\right)^{1/\alpha}\sigma=\left(|a_1|^{\alpha}+|a_2|^{\alpha}\right)^{1/\alpha}\sigma=\left(1^{\alpha}+|-1|^{\alpha}\right)^{1/\alpha}\sigma=(1+1)^{1/\alpha}\sigma=2^{1/\alpha}\sigma,$$

$$\frac{\sum_{k=1}^{2} a_k^{\langle\alpha\rangle}}{\sum_{k=1}^{2}|a_k|^{\alpha}}\beta = \frac{sign(a_1)|a_1|^{\alpha} + sign(a_2)|a_2|^{\alpha}}{|a_1|^{\alpha} + |a_2|^{\alpha}}\beta = \frac{1^{\alpha} - |-1|^{\alpha}}{1^{\alpha} + |-1|^{\alpha}}\beta = \frac{1-1}{2}\beta = 0,$$

$$\sum_{k=1}^{2} a_k \mu = (a_1 + a_2)\mu = (1-1)\mu = 0.$$

Лемма доказана.

Рассмотрим n независимых наблюдений $X_1,\ldots,Xn$ за случайной величиной $Xk \sim S_\alpha(\sigma,\beta,\mu)$.

Нетрудно заметить, что длина последовательности XC и XD это не более 1/3 длины исходной последовательности, а длина последовательности XS – не более 1/2 длины исходной последовательности. В связи с тем, что преобразование XD может преобразовать случайную последовательность $Xk \sim S_\alpha(\sigma,\beta,\mu)$ в симметричное распределение $X_k^D \sim S_{\alpha^D}\left(\sigma^D, 0, \mu^D\right)$ и μD – медиана XkD [3], тогда:

$$\mu = \mu^D\left(2 - 2^{1/\alpha}\right)^{-1}. \qquad (2)$$

3. РЕЗУЛЬТАТЫ МОДЕЛИРОВАНИЯ.

Учитывая, что длина последовательности должна быть кратна 3, смоделируем 9999 значений α – устойчивой случайной величины $S_\alpha(2,–0.9,1)$ и оценим параметр положения μ формула (2), при $\alpha \in [0.2, 1.8]$ (см. рис. 1, 2).

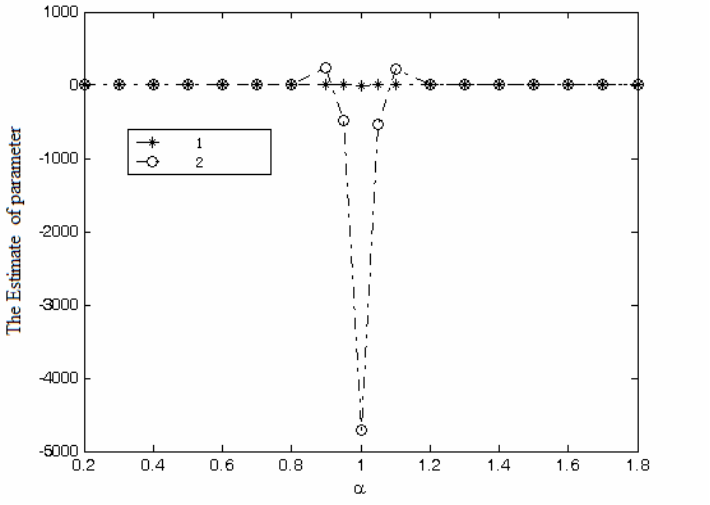

Рис.1. Оценка параметра μ для $S_\alpha(2,–0.9,1)$:
1–метод статьи;2–метод CF

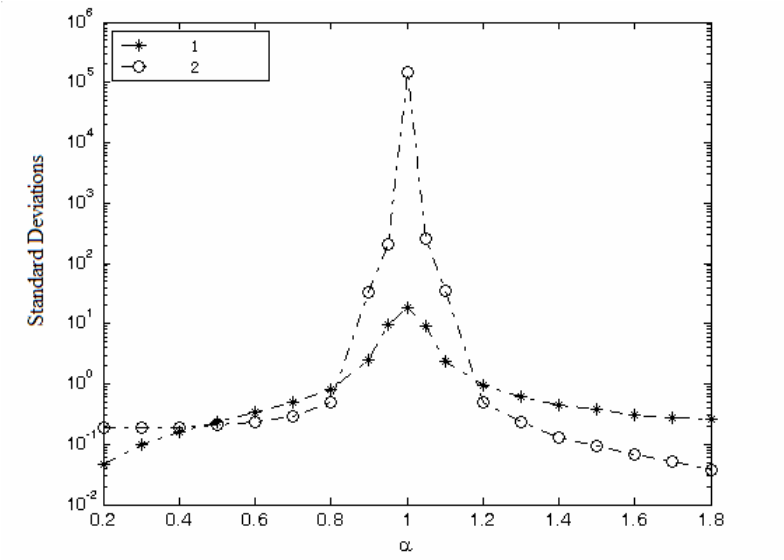

Рис.2. Стандартное отклонение параметра μ для $S_\alpha(2,–0.9,1)$:
1–метод статьи;2–метод CF

Для несимметричного распределения $S_\alpha(2,–0.9,1)$ метод оценки параметра положения μ при α, близким к 1, лучше метода CF.

Литература

1. *G. Samorodnistsky , M. Naqqu*. Stable Non-Gaussian Random Processes: Stochastic Models with Infinite Variance. // Chapman and Hall. New York. London. 1994.
2. *E. E. Kuruoglu*. Density parameter estimation of skewed α-stable distributions. // Signal Processing. IEEE Transaction on. Oct. 2001. 49(10). P. 2192–2201.
3. *В. М. Золотарев*. Одномерные устойчивые распределения. // В.М. Золотарев. М. Наука. 1983. С. 304.

РАЗРАБОТКА РАСПРЕДЕЛЁННЫХ СИСТЕМ НАДЁЖНОГО ХРАНЕНИЯ ДАННЫХ

В.О. Шукело

ВВЕДЕНИЕ

В эпоху глобальной информатизации в обществе накапливается огромное количество информации в электронном виде, которое представляет интерес для будущего. В связи с этим возникает проблема надежного сохранения накопленной информации. Технические устройства хранения данных не являются абсолютно надежными. Кроме того для надёжного хранения данных не могут использоваться централизованные системы. Такие системы должны иметь распределенный характер.

В работе исследуется проблема построения корпоративных сетей с надёжным хранением данных в течение длительного периода времени с непрерывным доступом к ним. Децентрализованные сети предлагается строить на основе технологии распределенных хеш-таблиц.

РАСПРЕДЕЛЁННЫЕ ХЕШ-ТАБЛИЦЫ

Для построения масштабируемых децентрализованных систем будем использовать распределённые хеш-таблицы DHT (Distributed Hash Table) – правила, по которым узлы связываются друг с другом и передают сообщения друг другу.

В DHT отсутствует явная иерархия между узлами. Все узлы считаются равноправными. Распределённая хеш-таблица состоит из M узлов – элементов, которые могут хранить данные и связываться с другими узлами. У каждого узла есть ключ K – строка из N бит, у каждого блока данных также есть ключ – также строка из N бит.

Ключи образуют метрическое пространство, вводящее понятие логического расстояния между узлами, между данными и между данными и узлами. Метрика может быть разной в зависимости от реализации DHT [1–2].

Для любого ключа среди всех узлов есть ближайший узел. Назовём его хозяином данного ключа K. Таким образов всё пространство ключей делится на M частей, у каждой части есть свой узел-хозяин.

Распределённая хеш-таблица – это децентрализованная отказоустойчивая масштабируемая сеть, каждый узел которой может для любого ключа может быстро найти его хозяина. Гарантируется, что в правильно построенной сети от любого узла можно перейти к хозяину ключа K по связям между соседними узлами, при этом каждый следующий узел будет ближе предыдущего к нашему ключу K.

Различные DHT отличаются топологией и алгоритмом маршрутизации [3]. Выбор топологии определяет те ограничения, которые накладываются на соседей (сеть подбирает соседей автоматически, без участия администратора, и не любые два узла могут быть соседями).

Сеть состоит из узлов, которые связываются с соседями, а так же направляют поступающие сообщения своим соседям. Соседями могут быть только те узлы, ключи которых соответствуют определённым требованиям. Жёсткость этих требований позволяет регулировать отказоустойчивость сети – чем меньше подходящих узлов, тем меньше соседей (эффективнее работает сеть), но больше проблем в случае сбоя сети. Промежуточный вариант – содержание “пассивных” соседей, не использующихся по умолчанию, а вызываемых в случае сбоя.

Для достижения максимального быстродействия, каждое поступившее сообщение направляется соседу, определённому алгоритмом. Более жёсткие алгоритмы ускоряют доставку сообщений, но в случае выхода выбранного маршрута из строя снижается надёжность. Как и в случае с соседями, могут быть запасные, менее оптимальные варианты маршрутов, к которым узел переходит в случае неработоспособности основного варианта.

Чтобы сеть была отказоустойчивой, каждый узел должен “знать” больше соседей, чем требуется для корректной работы и “уметь” маршрутизировать сообщения неоптимальным путём, в обход сбоев.

ОЦЕНКА НАДЁЖНОСТИ РАБОТЫ СЕТИ

Предположим, что количество узлов сети постоянно (вышедшие из строя узлы мгновенно заменяются новыми), что количество единиц данных в сети постоянно (утерянные данные дополняются новыми), что на передачу/сохранение единицы данных затрачивается фиксированное время, что все данные имеют (должны иметь) равное заданное число копий, что полезная нагрузка на сеть отсутствует (её наличие можно симулировать увеличением времени на обработку единицы данных). Также разобьём время на дискретные отрезки, в течении которых будем считать, что каждый узел или работает нормально, или теряет все свои данные в начале этого отрезка времени. Данное требование может соответствовать требованию начинать процедуру восстановления после сбоев в определённые моменты времени.

Введём обозначения: n – число узлов, k – число копий для каждого данного, s – число файлов во всей сети (не считая копий), Th – “период полураспада” одного узла, то есть время, за которое узел выйдет из строя с вероятностью ½, t - время обработки одной копии файла, T - “отчётный период” — время, для которого вычисляется количество потерянных данных. В каждом узле хранится sk/n копий данных. Значит, время обслуживания всех копий данных Ts = tsk/n. Вероятность выхода из строя одного узла за период Ts равна: f(Th, Ts)=1-2- Ts/ Th. Вероятность выхода ровно m узлов за период времени Ts равна: fm(Th, Ts,n)= Cnm(f)m(1-f)n-m.

Каждое данное присутствует на k узлах из n. Значит, общее количество вариантов распределения данных по узлам есть Cnk. При выходе из строя m узлов теряется Cnk возможных вариантов распределения. Поскольку данные распределяются по узлам случайным образом, то при

выходе из строя m узлов теряется Lm(n)= Cmk/ Cnk от всех данных, то есть sLm единиц данных. Учитывая случаи выхода 0,1,...,n узлов с соответствующими вероятностями и потерями данных, получаем, что за период времени Ts мы теряем единиц данных:

$$L_{T_s} = \sum_{i=0}^{n} f_i \, (s \, L_i)$$

Значит, за весь отчётный период мы потеряем единиц данных:

$$L_T(n, k, s, T_h, t) = \frac{T}{T_s} L_{T_s}.$$

СИСТЕМА МОДЕЛИРОВАНИЯ СЕТЕЙ ДЛЯ НАДЁЖНОГО ХРАНЕНИЯ ДАННЫХ

Общая схема системы моделирования приведена на рис.2. Основные объекты, с которыми работает эмулятор, это узлы, виртуальная сеть, виртуальные файлы, сообщения между узлами и генератор полезной нагрузки.

Моделирование начинается с построения сети. При создании каждого узла в очередь сообщений помещается специальное событие со случайным временем, вызывающее удаление создаваемого узла. Узлы создаются, устанавливают связи друг с другом, однако потом начинают исчезать. Взамен каждого исчезнувшего узла добавляется новый, с другим ключом. По сообщениям узлы распознают исчезнувших соседей и перестраивают связи.

В результате работы программы моделирования можно получать граф построенной сети, выводимый в текстовый файл, смотреть количество оставшихся файлов в моделируемой сети, разнообразные статистические параметры, выполняется визуализация графа сети.

ЗАКЛЮЧЕНИЕ

Технология распределённых хеш-таблиц — хорошая база для построения надёжных хранилищ данных. Использование программы моделирования позволит заранее оценить надёжность и характеристики сети в различных конфигурациях.

Это может быть полезно как при разработке самих алгоритмов обеспечения надёжности, так и для подбора параметров их работы перед развёртыванием настоящей, физической сети для хранения данных.

Рис.2. Интерфейс системы моделирования

Литература

1. Kademlia [Электронный ресурс] // Википедия. Режим доступа: http://en.wikipedia.org/w/index.php?title=Kademlia&oldid=340790072.
2. Distributed Hash Table [Электронный ресурс] // Википедия. Режим доступа: http://en.wikipedia.org/w/index.php?title=Distributed_hash_table&oldid=341573197.
3. *Gummadi K.* The Impact of DHT Routing Geometry on Resilience and Proximity [Электронный ресурс] // K. Gummadi, R. Gummadi, S. Gribble, S. Ratnasamy, S. Shenker, I. Stoica. Режим доступа: http://www.cs.washington.edu/homes/gribble/papers/p1101-gummadi.pdf .

МОДЕЛИ РАСПОЗНАВАНИЯ НА ОСНОВЕ ПРЕЦЕДЕНТНОГО И ЛОГИЧЕСКОГО ПРЕДСТАВЛЕНИЯ ИНФОРМАЦИИ

О. В. Шут

ВВЕДЕНИЕ

Основные концепции теории распознавания приобретают все большее признание в качестве фактора, существенного для построения современных информационных систем. Соответствующие задачи являются объек-

тами междисциплинарных исследований, проводимых в рамках теории информации, статистики, физики, химии, лингвистики, психологии, биологии, физиологии и медицины [1].

В задаче распознавания используются два основных способа представления начальной информации:

- логический: представление информации в виде правил;
- прецедентный: представление информации путем непосредственного указания объектов.

Актуальной является задача разработки алгебраической конструкции, которая позволяла бы переходить от правил к множеству объектов обучающей выборки и от множества объектов к правилу.

1. ОСНОВНЫЕ ОПРЕДЕЛЕНИЯ И ПОСТАНОВКА ЗАДАЧИ

Воспользуемся моделью описания объектов, предложенной в [2]. Пусть объект обладает n признаками, имеющими конечное множество значений. Будем называть такие признаки *номинальными* [3]. Обозначим через S_i множество признаков, из которого выбирается i-й признак, а через D_i – множество значений этого признака. *Объектом* будем называть отображение вида

$$p : S_1 \times S_2 \times ... \times S_n \to D_1 \times D_2 \times ... \times D_n$$

Если объект p обладает признаками $s_1, s_2, ..., s_n$, принимающими значения $d_1, d_2, ..., d_n$ соответственно, будем записывать это в виде

$$p(s_1, s_2, ..., s_n) = (d_1, d_2, ..., d_n)$$

Объекты будем считать *равными*, если множества их признаков равны, а значения соответствующих признаков совпадают.

Набором объектов или просто *набором* будем называть множество объектов, в котором все объекты обладают одними и теми же признаками.

Наборы P и Q будем считать *равными*, если для любого объекта, входящего в P, найдется равный ему объект, входящий в Q, и наоборот.

В [2] введены операции отрицания, умножения и сложения объектов и наборов и доказаны многие свойства этих операций.

В настоящей работе исследованы эквивалентность операций над объектами, обладающими номинальными признаками, булевым операциям, и полнота системы операций алгебры объектов. Разработаны алгоритмы перехода от логического представления информации к прецедентному и от прецедентного к логическому и дана оценка их сложности.

2. АЛГЕБРЫ ОБЪЕКТОВ И ЛОГИКИ

Исследуем соответствие операций над наборами объектов булевым операциям в классической алгебре логики. Для этого занумеруем все объекты и произвольному объекту p поставим в соответствие код

$$C(p) = (0...0 \quad 1 \quad 0...0),$$

где единица стоит в позиции, порядковый номер которой совпадает с номером этого объекта.

Каждому набору объектов сопоставим следующий код:

$$C(P) = \bigvee_{p \in P} C(p)$$

Соответствие между наборами (объектами) и их кодами является взаимно однозначным.

Будем считать операции алгебры объектов *эквивалентными* булевым операциям, если для любых наборов P и Q выполняются равенства:

$$C(P) \wedge C(Q) = C(P \wedge Q),\ C(P) \vee C(Q) = C(P \vee Q),\ \overline{C(P)} = C(\overline{P}).$$

Теорема 1. Операции $\neg, \wedge, \vee$ в алгебре объектов, обладающих номинальными признаками, эквивалентны булевым операциям $\neg, \wedge, \vee$.

Исследуем полноту системы операций алгебры объектов. В [2] показано, что любой набор может быть представлен в виде суммы его объектов, а любой объект может быть представлен в виде произведения его признаков.

Теорема 2. Система операций над наборами $\{\neg, \wedge, \vee\}$ является полной.

Следствие 1. Системы операций над наборами $\{\neg, \wedge\}$ и $\{\neg, \vee\}$ являются полными.

3. АЛГОРИТМЫ ПРЕОБРАЗОВАНИЯ СПОСОБОВ ПРЕДСТАВЛЕНИЯ ИНФОРМАЦИИ

Одним из основных способов представления начальной информации в задаче распознавания образов с обучением является указание правила, позволяющего отнести рассматриваемый объект к тому или иному классу объектов. Распространенным является представление такого правила в виде одной или нескольких функций алгебры логики, каждая из которых описывает принадлежность объекта конкретному классу.

Рассмотрим случай, когда все объекты имеют одинаковое число признаков, а все признаки принимают k значений из множества $D = \{0,1,...,k-1\}$. Все прочие случаи легко сводятся к указанному.

Пусть правило φ описывает принадлежность объекта p классу Y. Запишем φ в виде

$$\varphi(d_1,...,d_n)=\begin{cases}k-1, p\in Y\\ 0, p\notin Y\end{cases}$$

Для перехода от логического представления к прецедентному, т.е. для построения набора объектов, описываемых правилом φ, были разработаны следующие алгоритмы:

- алгоритм на основе алгебры объектов (алгоритм A_1)
- алгоритм на основе алгебры логики (алгоритм A_2)

Общая схема алгоритма A_1:

1. Каждой переменнной в φ ставится в соответствие объект.
2. Выполняются операции над объектами, соответствующие операциям k-значной логики в φ.
3. Если в φ используются не все признаки, то набор объектов, полученный на шаге 2, дополняется недостающими признаками.

Общая схема алгоритма A_2:

1. Правило φ приводится к виду СДНФ.
2. Каждой переменнной в СДНФ ставится в соответствие объект.
3. Выполняются операции над объектами, соответствующие операциям в СДНФ.

Разработан также алгоритм для перехода от прецедентного представления множества объектов к логическому (алгоритм B).

Общая схема алгоритма B:

1. Каждому признаку заданного набора объектов ставится в соответствие переменная.
2. Для каждого объекта выполняется конъюнкция переменных, соответствующих его признакам.
3. В качестве искомого правила берется дизъюнкция выражений, полученых на шаге 2.

Введем следующие обозначения: $P=A_i(\varphi)$ – набор P является результатом алгоритма A_i для правила φ; $\varphi=B(P)$ – правило φ является результатом алгоритма B для набора P.

Теорема 3. Для произвольного правила φ и произвольного объекта $p(s_1,s_2,...,s_n)=(d_1,d_2,...,d_n)$ справедливы утверждения:

1. $\varphi(d_1,...,d_n)=k-1\Leftrightarrow(B\circ A_i(\varphi))(d_1,...,d_n)=k-1$
2. $p\in P\Leftrightarrow p\in(A_i\circ B(P))$

Таким образом, преобразования, производимые алгоритмами A_i и B, являются взаимно обратными.

Дадим оценку сложности разработанных алгоритмов. Пусть заданы наборы P и Q, содержащие r_1 и r_2 объектов и обладающие n_1 и n_2 признаками соответственно.

Теорема 4. Алгоритм A_1 строит набор $P \vee Q$ за N_1 операций, где

$N_1 = (k^{n1} + k^{n2})(r_1 + r_2) - r_1 r_2$

Теорема 5. Алгоритм A_2 строит набор $P \vee Q$ за N_2 операций, где

$N_2 = k^{n-n1} r_1 + k^{n-n2} r_2$

Теорема 6. Сложность алгоритма B составляет $O(nr)$, где r – количество объектов набора, n – количество признаков.

Если правило φ задано в виде ДНФ, то алгоритм A_2 является более эффективным, чем алгоритм A_1. Однако в общем случае, если φ представляет собой произвольную функцию k-значной логики, предпочтительнее использовать алгоритм A_1, т.к. сложность первого этапа алгоритма A_2 значительно возрастает.

Оценки сложности всех вышеописанных алгоритмов были подтверждены серией компьютерных экспериментов.

Литература

1. *Гонсалес Р., Ту Дж.* Принципы распознавания образов. М.: Мир. 1978. С. 412.
2. *Рябцев А.В.* Алгебры для представления обучающей информации в задачах распознавания образов. // Цифровая обработка. Мн. 2002. вып.6. С.80–94.
3. Интернет-адрес: http://dic.academic.ru/ dic.nsf/ruwiki/969082.
4. *Яблонский С. В.* Введение в дискретную математику. М.: Наука. 1986. С. 384.

ОБ УСТОЙЧИВОСТИ ЯВНОЙ РАЗНОСТНОЙ СХЕМЫ ДЛЯ КВАЗИЛИНЕЙНОГО ПАРАБОЛИЧЕСКОГО УРАВНЕНИЯ

Р. М. Якубук

При исследовании разностных схем одним из наиболее важных является вопрос устойчивости разностного решения относительно малого возмущения входных данных. Принципиальное отличие исследования устойчивости и монотонности в нелинейном случае заключается в необходимости дополнительного получения априорных оценок для всех производных, входящих в нелинейную часть. В [1–3] проводится краткий обзор работ по данному направлению, указывается важная взаимосвязь корректности, устойчивости и монотонности разностных задач. В [1]

проводится исследование устойчивости явной схемы, аппроксимирующей начально-краевую задачу для многомерного квазилинейного параболического уравнения с квадратичной нелинейностью. В [2] исследуется монотонность и устойчивость неявной разностной схемы для той же задачи. В [3] рассматривается устойчивость по отношению к возмущению начальных данных безитерационной неявной разностной схемы, аппроксимирующей начально-краевую задачу для одномерного квазилинейного уравнения. Особенно важно подчеркнуть, что все исследования проводятся только в предположении на входные данные задачи.

В цилиндрической области $\bar{Q}_T = \bar{\Omega} \times \{0 \le t \le T\}$, $\bar{\Omega} = \left\{ x \in \mathbf{R}^p, \right.$ $\left. x = \left(x_1, \ldots, x_p\right), 0 \le x_m \le l_m, m = \overline{1,p} \right\}$, $\bar{\Omega} = \Omega \cup \Gamma$, Γ – граница области $\bar{\Omega}$, Ω – внутренняя часть области, $\Gamma_T = \Gamma \times \{0 \le t \le T\}$, $Q_T = \Omega \times \{0 < t \le T\}$, рассмотрим краевую задачу

$$\frac{\partial u}{\partial t} = \sum_{m=1}^{p} \frac{\partial}{\partial x_m} \left(k(u) \frac{\partial u}{\partial x_m} \right) + f(x,t), \quad (x,t) \in Q_T, \tag{1}$$

$$u(x,t)|_{\Gamma_T} = \mu(x,t), \qquad u(x,0) = u_0(x), \quad x \in \bar{\Omega}. \tag{2}$$

Уравнение (1) может быть записано в виде

$$\frac{\partial u}{\partial t} = \Delta \Phi(u) + f(x,t), \quad (x,t) \in Q_T,$$

где $\Delta u = \sum_{i=1}^{p} \frac{\partial^2 u}{\partial x_i^2}$, $\Phi(u) = \int_a^u k(v)dv$, $a = \text{const} \in \mathbb{R}$.

Пусть Φ удовлетворяет следующим условиям:

$$\exists \bar{D}_k \quad \forall v \in \bar{D}_k \quad 0 < \nu_1 \le \Phi'(v) = k(v) \le \nu_2 < \infty, \tag{3}$$

где $\bar{D}_k$ – замкнутое связное подмножество $\mathbb{R}$, функция $k(v)$ непрерывно дифференцируема на $\bar{D}_k$.

Кроме того, будем предполагать выполнение ограничений:

1° $\Phi(u_0) \in C^4(\bar{\Omega})$, $u_0 \in C(\bar{\Omega})$; $\kappa_1 \le u_0(x) \le \kappa_2$, $x \in \bar{\Omega}$, $\kappa_3 \le \mu(x,t) \le \kappa_4$, $(x,t) \in \Gamma_T$, где $\kappa_i = \text{const} \in \bar{D}_k$, i=1,2,3,4.

2° Выполнено одно из условий

$$\forall v \in \bar{D}_k \quad k'(v) \ge 0 \quad \text{и} \quad \Delta \Phi(u_0) + f(x,0) \le -\varepsilon_1, \quad f \ge 0, \quad \frac{\partial f}{\partial t}, \frac{\partial \mu}{\partial t} \le 0,$$

или

$$\forall v \in \bar{D}_k \quad k'(v) \le 0 \quad \text{и} \quad \Delta\Phi(u_0) + f(x,0) \ge \varepsilon_1, \quad f \le 0, \quad \frac{\partial f}{\partial t}, \frac{\partial \mu}{\partial t} \ge 0,$$

где $\varepsilon_1 = \text{const} > 0$.

Здесь и далее, если это не оговорено особо, предполагается, что переменные x, t принадлежат тому подмножеству $\bar{Q}_T$, на котором определены все функции, входящие в выражение.

Если u_0, μ, f являются достаточно гладкими функциями своих аргументов, начальные и граничные условия согласованы при $x \in \Gamma$, $t = 0$, выполнены условия (3) равномерной параболичности, области значений u_0, μ являются подмножествами $\bar{D}_k$, то существует единственное обобщённое решение задачи (1), (2) [4, с. 513].

На сетке $\bar{\omega}^{p+1} = \bar{\omega}_h \times \bar{\omega}_\tau$, $\bar{\omega}_h = \left\{ x_k = \left(x_1^{(i_1)}, \ldots, x_p^{(i_p)} \right), x_k^{(i_k)} = i_k h_k, i_k = \overline{0, N_k}, h_k N_k = l_k, k = \overline{1, p} \right\}$, $\bar{\omega}_\tau = \left\{ t_n = n\tau, n = \overline{0, N_0}, \tau N_0 = T \right\} = \omega_\tau \cup \{0\}$, $\omega^{p+1} = \omega_h \times \omega_\tau$ дифференциальную задачу (1), (2) аппроксимируем явной разностной схемой:

$$y_t = \sum_{m=1}^{p} \frac{1}{h_m^2} (\Phi(y_{(-1_m)}) - 2\Phi(y) + \Phi(y_{(+1_m)})) + f, \quad (x,t) \in \omega^{p+1}, \tag{4}$$

$$y|_{\gamma_h} = \mu(x,t), \quad x \in \gamma_h, \quad t \in \omega_\tau, \qquad y|_{t=0} = u_0(x), \quad x \in \bar{\omega}_h. \tag{5}$$

Здесь и ниже мы используем следующие обозначения [5]:

$$y = y(x,t), (x,t) \in \bar{\omega}^{p+1}, \hat{y} = y(x, t+\tau),$$
$$y_{(\pm 1_m)} = y(x_{(\pm 1_m)}, t) = y(x_1, \ldots, x_m \pm h_m, \ldots, x_p, t),$$
$$y_t = (\hat{y} - y)/\tau, \; y_{\bar{x}_m x_m} = (y_{(+1_m)} - 2y + y_{(-1_m)})/h_m^2,$$
$$y^n = y(x,t_n), \; \mu^n = \mu(x,t_n), \; f^n = f(x,t_n), \; t_n = n\tau.$$

Пусть для схемы (4), (5) верно следующее условие:

3° Выполнены следующие ограничения на шаги

$$h \le h_0, \quad h = \max_{1 \le m \le p} h_m, \quad h_0 = \sqrt{12\varepsilon_1 / p\kappa}, \quad \kappa = \max_{1 \le m \le p} \left\| \frac{\partial^4 \Phi(u_0)}{\partial x_m^4} \right\|_{C(\bar{\Omega})}.$$

В следующей теореме приводятся оценки решения разностной схемы (4), (5).

Теорема 1. Пусть выполнены условия $1° - 3°$ и

$$\min_{x\in\bar{\Omega}} u_0(x)=\kappa_1,\quad \max_{x\in\bar{\Omega}} u_0(x)=\kappa_2,\quad \|\mu\|_{C(\Gamma_T)}<\infty,\quad \|f\|_{C(Q_T)}<\infty.$$

Если для $n=0,\ldots,N_0-1$

$$\max_{v\in[\kappa_1^n,\kappa_2^n]} k(v)\sum_{m=1}^{p}\frac{2\tau}{h_m^2}\le 1,$$

где $\kappa_1^n=\min\left\{\min_{x\in\gamma_h}\mu^n,\kappa_1\right\}$, $\kappa_2^n=\max\left\{\max_{x\in\gamma_h}\mu^n,\kappa_2\right\}$, то для $n=0,\ldots,N_0$

$$\kappa_1^n\le y^n\le\kappa_2^n.$$

Следствие. При выполнении условий теоремы 1 для $n=0,\ldots,N_0$

$$K_1\le y^n\le K_2,\quad \text{где}\quad K_1=\kappa_1^{N_0},\quad K_2=\kappa_2^{N_0},\quad K_1,K_2\in\bar{D}_k.$$

Рассмотрим дифференциальную задачу (1), (2) с возмущёнными начальными и граничными условиями и правой частью

$$\frac{\partial\tilde{u}}{\partial t}=\Delta\Phi(\tilde{u})+\tilde{f}(x,t),\quad x\in\Omega,\quad 0<t\le T, \tag{6}$$

$$\tilde{u}(x,t)|_{\Gamma_T}=\tilde{\mu}(x,t),\qquad \tilde{u}(x,0)=\tilde{u}_0(x),\quad x\in\bar{\Omega}. \tag{7}$$

Аппроксимируем задачу (6), (7) разностной схемой

$$\tilde{y}_t=\sum_{m=1}^{p}\frac{1}{h_m^2}(\Phi(\tilde{y}_{(-1_m)})-2\Phi(\tilde{y})+\Phi(\tilde{y}_{(+1_m)}))+\tilde{f},\quad (x,t)\in\omega^{p+1}, \tag{8}$$

$$\tilde{y}|_{\gamma_h}=\tilde{\mu}(x,t),\quad x\in\gamma_h,\quad t\in\omega_\tau,\qquad \tilde{y}|_{t=0}=\tilde{u}_0(x),\quad x\in\bar{\omega}_h. \tag{9}$$

Для задачи (6), (7) и схемы (8), (9) будем предполагать выполнение условий $1°-3°$.

Для решения разностной схемы (8), (9) верны оценки такого же типа, как в теореме 1. Обозначим через $\tilde{K}_1$, $\tilde{K}_2$ нижнюю и верхнюю границы решения схемы (8), (9).

Введём обозначения

$$K_3=\min\left\{K_1,\tilde{K}_1\right\},\ K_4=\max\left\{K_2,\tilde{K}_2\right\}.$$

Теорема 2. Если выполнены условия $1°-3°$ для дифференциальных задач и аппроксимирующих их разностных схем и

$$\max_{v\in[K_3,K_4]} k(v)\sum_{m=1}^{p}\frac{2\tau}{h_m^2}\le 1,$$

то разностное решение устойчиво по отношению к малым возмущениям начальных и граничных условий на временном интервале $[0,T]$.

Литература

1. *P. Matus, S. Lemeshevsky* Stability and monotonicity of difference schemes for nonlinear scalar conservation laws and multidimensional quasi-linear parabolic equations // Comput. Meth. Appl. Math. 2009. Vol. 9. N. 3. P. 253–280.
2. *Матус П. П.* Устойчивость по начальным данным и монотонность неявной разностной схемы для однородного уравнения пористой среды с квадратичной нелинейностью // Дифференц. уравнения. 2010. Т. 46. № 7.
3. *P. Matus* Stability of difference schemes for nonlinear time-dependent problems // Comput. Meth. Appl. Math. 2003. Vol. 3. N. 2. P. 313–329.
4. *Ладыженская О.А., Солонников В.А., Уральцева Н.Н.* Линейные и квазилинейные уравнения параболического типа. М.: Наука. 1967.
5. *Самарский А. А.* Теория разностных схем. М.: Наука. 1989.