



Ученые записки

Ульяновского
государственного
университета

Серия

ФУНДАМЕНТАЛЬНЫЕ
ПРОБЛЕМЫ
МАТЕМАТИКИ И
МЕХАНИКИ

Министерство общего и профессионального образования
Российской Федерации

Ученые записки Ульяновского государственного университета

**ФУНДАМЕНТАЛЬНЫЕ ПРОБЛЕМЫ
МАТЕМАТИКИ И МЕХАНИКИ**

Выпуск 3

Под редакцией *Б.Ф.Мельникова*

Ульяновск, 1997

Печатается по решению Редакционно-издательского совета
Ульяновского государственного университета

Рецензент: д. ф.-м. н. *А.С.Андреев*

Фундаментальные проблемы математики и механики: Сборник статей. Выпуск 3 / Под ред. Б.Ф.Мельникова. Ульяновск: УлГУ, 1997. 133 с.

ISBN -5-88866-011-6

Третий сборник из серии, посвященной фундаментальным проблемам математики, прикладной математике, механике и информатике, включает в себя работы преподавателей, аспирантов и соискателей механико-математического факультета УлГУ. Рассматривается широкий круг вопросов, связанных с современными проблемами математики.

В данный сборник вошли работы, выполненные на следующих кафедрах: информационных систем, математической кибернетики и информатики, прикладной математики. (Работы, выполненные на других кафедрах, будут включены в четвертый выпуск).

ISBN -5-88866-011-6

Оглавление

К. Г. Арбеев. Об одном алгоритме нахождения оптимального базиса в моделях многосвязных систем диффузионного типа	5
A. Butov. Arbitrage possibilities for the price processes with fixed terminal values	9
В. К. Горбунов, И. В. Лутошкин. Вторые производные параметризованной задачи оптимального управления	17
Sh. Ishmukhametov. On degrees without strong minimal covers	25
Л. В. Калинин, А. Г. Сковиков. Идентификация нарушений в линейных стохастических системах	32
Е. Н. Маслина. Примеры скобочных структур языка Паскаль	37
Б. Ф. Мельников. Некоторые алгебраические задачи в теории формальных языков	41
Б. Ф. Мельников, А. А. Вахитова. Звездная высота конечного автомата	51
Б. Ф. Мельников, И. В. Муллина. Об одном «быстром» алгоритме построения регулярного выражения по заданному НРС-автомату	58
Е. А. Михеева. О наборах линейных функций из P_2	64
Н. Н. Нечаева. Методика анализа линейных моделей в языке ФС	70
А. Ф. Николаев. Об одной постановке задачи о множественной разладке	73
А. Н. Радионов. Игровые ассистирующие программы	77
А. Г. Сковиков, Л. В. Калинин. Определение моментов нарушений в линейной динамической системе	79
В. В. Угаров. Об одной методике оптимизации ввода данных в автоматизированных системах измерений	84

Ю. В. Цыганова. Разработка программных средств моделирования и исследования стохастических систем оптимальной дискретной фильтрации	88
Г. Ю. Куликов, Н. А. Андрианова. Об оптимальном методе Эйлера-простых итераций для решения дифференциально-алгебраических уравнений	98
Г. Ю. Куликов, А. А. Корнева. Об эффективной реализации итеративных методов Рунге-Кутты для дифференциально-алгебраических уравнений индекса 1	103
Г. Ю. Куликов, С. К. Шиндин. Об одном способе вычисления глобальной погрешности для методов дифференцирования назад	113
Ю. Ю. Терентьева. Помехоустойчивое кодирование при неизвестной статистике источника	120
В. К. Горбунов, В. В. Петрищев. Метод нормальных сплайнов в вырожденных системах дифференциальных уравнений	125

К. Г. Арбеев.

Об одном алгоритме нахождения оптимального базиса в моделях многосвязных систем диффузионного типа

Пусть имеется объект, описываемый системой n стохастических уравнений диффузионного вида:

$$dX_t = B_t(\omega)dt + \Sigma_t(\omega)dW_t. \quad (1)$$

Обозначим H_n^D — линейное пространство, порождаемое процессами вида (1).

Поставим в соответствие каждому из наблюдаемых процессов X_t^i “веса” $v_i \geq 0, i = 1, \dots, n$. Обозначим V — вектор “весов”. Назовем

$$J(V) = \sum_{i=1}^n v_i \quad (2)$$

функционалом “предпочтения”, соответствующим вектору X_t (т.е. его вектору “весов” V).

На основании (2) введем отношение “предпочтения” для базисных векторов пространства H_n^D .

Определение 1: Базис $X_t^{(1)}$ “лучше” (“предпочтительнее”) базиса $X_t^{(2)}$, если $J(V^{(1)}) > J(V^{(2)})$, где $V^{(i)}$ — вектор “весов”, соответствующий базису $X_t^{(i)}, i = 1, 2$.

Определение 2: Базис $X_t^{(opt)}$ назовем оптимальным или наилучшим, если

$$J_{opt} = J(V^{(opt)}) = \max_{X_t^{(k)} \text{ — базис } H_n^D} J(V^{(k)}).$$

Поставим задачу найти оптимальный базис пространства H_n^D , исходя из наблюдения траекторий процессов $X_t^i, i = 1, \dots, n$.

Для нахождения произвольного базиса H_n^D можно воспользоваться соотношением:

$$\bar{X}_t = CX_t \quad (3)$$

где C — матрица перехода к базису $\bar{X}_t$,

$$c_{ij} = -\frac{\phi(\bar{X}_t^j, X_t^i)}{\phi(\bar{X}_t^j, \bar{X}_t^j)}, i > j, c_{ii} = 1, c_{ij} = 0, i < j, i, j = 1, \dots, n. \quad (4)$$

$$\bar{X}_t^j = \sum_{k=1}^{j-1} c_{jk} \bar{X}_t^k + X_t^j, j = 1, \dots, n. \quad (5)$$

Здесь $\phi(.,.)$ — некоторое скалярное произведение, определенное для траекторий процессов X_t^i .

Тогда $\bar{V} = CV$, где $\bar{V}$ — вектор “весов”, соответствующий базису $\bar{X}_t$. Следовательно,

$$J(\bar{V}) = J(CV) = J(V) + \sum_{j=1}^{n-1} \sum_{i=j+1}^n c_{ij} v_j. \quad (6)$$

Из (6) следует, что

$$\max_{\bar{X}_t - \text{базис } H_n^D} J(\bar{V}) = \max_{\bar{X}_t - \text{базис } H_n^D} \left(\sum_{j=1}^{n-1} \sum_{i=j+1}^n c_{ij} v_j \right). \quad (7)$$

Исходя из соотношения (7), предлагается следующий **Алгоритм нахождения оптимального базиса.**

Шаг 1. Оптимизация $\sum_{i=2}^n c_{i1} v_1$:

$$X_t^{i_1} = \arg \max_{X_t^{k_1} \in \{X_t^1, \dots, X_t^n\}} \left(\sum_{i=1, i \neq k_1}^n c_{ik_1} (X^{k_1}) v_{k_1} \right). \quad (8)$$

Обозначим $^{(1)}X_t = A_{1I_1} X_t$, $^{(1)}V = A_{1I_1} V$, где

$$A_{ij} = \begin{pmatrix} 1 & & & & 0 \\ & \ddots & & & \\ & & 0 & \dots & 1 \\ & & \vdots & \ddots & \vdots \\ & & 1 & \dots & 0 \\ & & & & \ddots & \\ 0 & & & & & & 1 \end{pmatrix}, \bar{X}_t^1 = ^{(1)}X_t.$$

Замечание: Из (8) видно, что от выбора дальнейших компонент X_t^i значение максимума не изменится.

...

Шаг N ($N = 2, \dots, n-1$). Оптимизация $\sum_{i=N+1}^n c_{iN} v_N$:

$$X_t^{l_N} = \arg \max_{X_t^{k_N} \in \{(N-1)X_t^N, \dots, (N-1)X_t^1\}} \left(\sum_{i=N, i \neq k_N}^n c_{ik_N} (X^{k_N})^{(N-1)} v_{k_N} \right). \quad (9)$$

Обозначим $^{(N)}X_t = A_{Nl_N} {}^{(N-1)}X_t$, $^{(N)}V = A_{Nl_N} {}^{(N-1)}V$.

Замечание: Из (9) видно, что от выбора дальнейших компонент X_t^i значение максимума не изменится.

Итак, получаем итоговое преобразование исходного вектора наблюдений и соответствующего ему вектора "весов":

$$^{(n)}X_t = A_{(n-1)l_{n-1}} \times \dots \times A_{1l_1} X_t, \quad ^{(n)}V = A_{(n-1)l_{n-1}} \times \dots \times A_{1l_1} V. \quad (10)$$

Получаем оптимальный базис:

$$\bar{X}_t = \bar{C}^{(n)} X_t, \quad (11)$$

где $\bar{c}_{ij} = -\frac{\phi(\bar{X}_t^j, {}^{(n)}X_t^i)}{\phi(\bar{X}_t^j, \bar{X}_t^j)}$, $i > j$, $\bar{c}_{ii} = 1$, $\bar{c}_{ij} = 0$, $i < j$, $i, j = 1, \dots, n$.

В качестве реальных процессов, динамику которых можно описать с помощью системы (1), были взяты курсы валют, предоставляемые *PACIFIC Exchange Rate Service* в сети *INTERNET*. Рассмотрены различные наборы валют и различные периоды наблюдения в диапазоне 02.01.1990 — 30.01.1997 г.г.

В качестве скалярных произведений $\phi(.,.)$ использовались:

- 1) $\phi(X, Y) = [X, Y]_t$.
- 2) $\phi(X, Y) = [\tilde{X}, \tilde{Y}]_t$, где $\tilde{X}_s = \alpha_t^s X_s$, $\alpha_t^s = e^{-\mu(t-s)}$, $\mu > 0$.
- 3) $\phi(X, Y) = \frac{1}{t} \int_0^t (X_s - m_X^t)(Y_s - m_Y^t) ds$, где $m_X^t = \frac{1}{t} \int_0^t X_s ds$.
- 4) $\phi(X, Y) = \frac{1}{t} \int_0^t (\tilde{X}_s - \tilde{m}_X^t)(\tilde{Y}_s - \tilde{m}_Y^t) ds$, где $\tilde{X}_s = \alpha_t^s X_s$, $\tilde{m}_X^t = \frac{1}{t} \int_0^t \alpha_t^s X_s ds$.

Выбор рассматриваемых скалярных произведений был обусловлен следующими соображениями: в 1) и 2) квадратичная вариация означает независимость при предположении гауссовости и линейности диффузионных процессов, в 2) при этом рассматривается естественное и традиционное экспоненциальное затухание. В 3) и 4) рассматриваются аналоги ковариации и соответственно скалярного произведения в L^2 , с традиционным затуханием.

На основе $\phi(.,.)$ для 1)–4) вычислены “веса” v_i для валют, как

$$v_i = \left(\sum_{j=1, j \neq i}^n \left| \frac{\phi(X^i, X^j)}{\sqrt{\phi(X^i, X^i)\phi(X^j, X^j)}} \right| \right)^{-1}. \quad (12)$$

Для расчета “весов” также использовались “расстояния” между валютами:

$$v_i = \sum_{j=1, j \neq i}^n \rho(X^i, X^j), \quad (13)$$

где:

$$1) \quad \rho(X, Y) = \left(\int_0^t \alpha_t^s (\bar{X}_s - \bar{Y}_s)^2 ds \right)^{\frac{1}{2}}, \text{ где } \bar{X}_s = \frac{X_s}{\int_0^s \alpha_t^u du}.$$

$$2) \quad \rho(X, Y) = \int_0^t \alpha_t^s |\bar{X}_s - \bar{Y}_s| ds.$$

$$3) \quad \rho(X, Y) = \sup_{0 \leq s \leq t} |\bar{X}_s - \bar{Y}_s|,$$

что также принято в экономических моделях, но рассматривается не в таком виде, а на интуитивном уровне.

Литература

- [1] Липцер Р.Ш., Ширяев А.Н. Теория мартингалов – М: Наука, 1986.
- [2] Биллингсли П. Сходимость вероятностных мер – М: Наука, 1977.
- [3] Липцер Р.Ш., Ширяев А.Н. Статистика случайных процессов – М: Наука, 1974.

A. Butov.

Arbitrage possibilities for the price processes with fixed terminal values

§ 1. Introduction

The problem of arbitrage possibilities (or no-arbitrage properties) is widely discussed. Nevertheless the major part of general results is formulated in terms of existence of some equivalent measures solving the martingale (or supermartingale) problem (see for example [1]). However the problem of constructive presentation of arbitrage possibilities for general semimartingale models remains.

Here we present some results conserving the analysis of (B, S) -market with the price process $S = (S_t)_{0 \leq t \leq 1}$ having "fixed" deterministic value at the finite time $t = 1$. Such a process is of especial interest due to the future fixation of national foreign exchange rates in Europe. Another general object for such an analysis - price of a bond $S = (S_t)_{0 \leq t \leq 1}$ with terminate price S_1 , different from the numeraire $(B_t)_{0 \leq t \leq 1}$.

In this paper we discuss the simple (but quite general) model of (B, S) -market with $B = (B_t)_{0 \leq t \leq 1}$ - continuous positive nondecreasing process with locally bounded variation, and $S = (S_t)_{0 \leq t \leq 1}$ - continuous positive semimartingale with fixed value $S_1 = \exp\{c\}$, $c = \text{const}$. For such a scheme under simple and clear conditions the arbitrage procedure is presented.

§ 2. The Basic model

Let $\mathcal{B} = (\Omega, \mathcal{F}, \mathbf{F} = (F_t)_{0 \leq t \leq 1}, \mathbf{P})$ be a stochastic basis satisfying the usual conditions (see [2] or [3]). Let $S = (S_t)_{0 \leq t \leq 1}$ be the positive continuous semimartingale on $\mathcal{B}$ (the process of foreign exchange rate), and $B = (B_t)_{0 \leq t \leq 1}$ - the positive nondecreasing continuous interest process with locally bounded variation (numeraire).

We shall consider the processes (B, S) as an analogy to the Black-Scholes model under the condition of fixation of the terminal (positive) value of S : there exists the number c , such that $\mathbf{P}$ -a.s.

$$S_1 = S_0 \exp\{c\} \quad (1)$$

It is clear, that positiv processes B and S can be represented as

$$B_t = B_0 \exp\{b_t\} \quad (2)$$

and

$$S_t = S_0 \exp\left\{p_t - \frac{1}{2}[p, p]_t\right\} \quad (3)$$

with nondecreasing continuous process of locally bounded variation $b = (b_t)_{0 \leq t \leq 1}$, $p = (p_t)_{0 \leq t \leq 1}$ - continuous semimartingale on $\mathcal{B}$, and $b_0 = p_0 = [p, p]_0 = 0$ (here $[X, X]_t$ denotes the quadratic variation of X on $[0, t]$).

For the sake of simplicity we shall suppose that $S_0 = B_0 = 1$, $\mathbf{F} = (F_t^{S, B})_{0 \leq t \leq 1}$ hence $F_0 = \{\emptyset, \Omega\}$. Also we shall restrict the investigation of the processes $b = (b_t)_{0 \leq t \leq 1}$ and $p = (p_t)_{0 \leq t \leq 1}$, by the conditions (4) and (5):

$$\mathbf{P}\{b_1 < \infty\} = 1 \quad (4)$$

(that is b is finite)

$$E[p, p]_1 < \infty \quad (5)$$

that is the semimartingale $p = (p_t)_{0 \leq t \leq 1}$ in canonical decomposition

$$p_t = a_t + m_t, a_0 = m_0 = p_0 = 0 \quad (6)$$

with $a = (a_t)_{0 \leq t \leq 1}$ - the continuous process of locally bounded variation and $m = (m_t)_{0 \leq t \leq 1}$ - continuous local martingale on $\mathcal{B}$ ($m \in \mathcal{M}_{loc}^c(\mathcal{B})$, [3]) has in reality square integrable martingale part ($m \in M^{c,2}(\mathcal{B})$):

$$Em_1^2 = E\langle m \rangle_1 = E[m, m]_1 = E[p, p]_1 < \infty \quad (7)$$

Hence $\mathbf{P}\{|m_1| < \infty\} = \mathbf{P}\{\langle m \rangle_1 < \infty\} = \mathbf{P}\{[p, p]_1 < \infty\} = 1$ and the condition of fixation (1) can be written in the following form:

$$p_1 - \frac{1}{2}[p, p]_1 = c \quad \mathbf{P}\text{-a.s.} \quad (8)$$

or

$$a_1 + m_1 - \frac{1}{2}\langle m \rangle_1 = c \quad \mathbf{P}\text{-a. s.} \quad (9)$$

Note that $E|a_1| \leq |c| + E|m_1| + \frac{1}{2}E\langle m \rangle_1 < \infty$ and along with (4) we have the inequality

$$\mathbf{P}\{|a_1| < \infty\} = 1$$

We should remark that condition (5) is not necessary for the following considerations and can be weakened.

So the processes B and S can also be represented as the solutions of the equations (with $B_0 = S_0 = 1$)

$$B_t = 1 + \int_0^t B_u db_u \quad (10)$$

$$S_t = 1 + \int_0^t S_u dp_u = 1 + \int_0^t S_u da_u + \int_0^t S B_u dm_u \quad (11)$$

For such processes we shall consider an analogy to the Black-Scholes model of (B, S) - market with the capital $X = (X_t)_{0 \leq t \leq 1}$ and strategy (β, γ) :

$$X_t = \gamma_t S_t + \beta_t B_t \quad (12)$$

where γ_t and β_t are the numbers (or amounts) of foreign currency and bonds with price of numeraire accordingly in portfolio at time $t \in [0, 1]$. We shall consider the behaviour of X with $X_0 = 0$ and under the condition of selffinancing:

$$X_t = \int_0^t \gamma_u dS_u + \int_0^t \beta_u dB_u \quad (13)$$

Along with X (and to some extent - instead of X) we shall investigate the process $Y = (Y_t)_{0 \leq t \leq 1}$ of normalized capital

$$Y_t = \frac{X_t}{B_t} \quad (\text{with } Y_0 = 0)$$

or

$$Y_t = \frac{\int_0^t \gamma_u dS_u + \int_0^t \beta_u dB_u}{B_t} \quad (14)$$

Statement 1. *The process $Y = (Y_t)_{0 \leq t \leq 1}$ can be represented as*

$$Y_t = \int_0^t \Gamma_u (dp_u - db_u) \quad (15)$$

with

$$\Gamma_t = \gamma_t \frac{S_t}{B_t} = \gamma_t \exp\{p_t - \frac{1}{2}[p, p]_t - b_t\} \quad (16)$$

Proof. Due to (13) and (10) - (11)

$$X_t = \int_0^t \gamma_u S_u dp_u + \int_0^t \beta_u B_u db_u$$

and as it follows from (12) $\beta_u B_u = X_u - \gamma_u S_u$

$$X_t = \int_0^t \gamma_u S_u dp_u + \int_0^t (X_u - \gamma_u S_u) db_u = \int_0^t X_u db_u + \int_0^t (\gamma_u S_u)(dp_u - db_u) \quad (17)$$

The solution of the equation (17) is

$$X_t = e^{b_t} \int_0^t e^{-b_u} (\gamma_u S_u)(dp_u - db_u) = B_t \int_0^t \Gamma_u (dp_u - db_u).$$

Hence for $Y_t = X_t/B_t$ the equality (15) is true. $\square$

Note that the process $\Gamma = (\Gamma_t)_{0 \leq t \leq 1}$ is setting $\gamma = (\gamma_t)_{0 \leq t \leq 1}$:

$$\gamma_t = \Gamma_t \frac{B_t}{S_t} = \Gamma_t \exp\{-p_t + \frac{1}{2}[p, p]_t + b_t\} \quad (18)$$

and due to the condition of selffinancing - $\beta = (\beta_t)_{0 \leq t \leq 1}$ also is defined by Γ . So we shall use the term "strategy" for Γ as well.

The main aim of this article is to find out the strategy Γ (and hence set (β, γ)) such that proper conditions it results in "arbitrage":

Definition. The strategy Γ results in the arbitrage if for $Y = (Y_t)_{0 \leq t \leq 1}$ with $Y_0 = 0$

$$\mathbf{P}\{Y_1 \geq 0\} = 1 \quad (19)$$

and

$$\mathbf{P}\{Y_1 > 0\} > 0 \quad (20)$$

(or if it is possible $\mathbf{P}\{Y_1 > 0\} = 1$ for strict arbitrage).

The process Y is a continuous semimartingale on B , hence it admits the canonical unique decomposition

$$Y_t = Y_t^c + \tilde{Y}_t \quad (21)$$

with $Y^c = (Y_t^c)_{0 \leq t \leq 1}$ - the local continuous martingale on B (or $Y^c \in \mathcal{M}_{loc}^c(B)$), and $\tilde{Y} = (\tilde{Y}_t)_{0 \leq t \leq 1}$ - the compensator (or dual predictable projection of Y , [2]).

Definition. We shall say that the strategy Γ is admissible if the local martingale Y^c in the decomposition (21) of Y is martingale on

$B: Y^c \in \mathcal{M}^c(B)$ (for the case of continuous martingales with $t \in [0, 1]$ also uniformly integrable, [3]).

This definition excludes the case of infinit investments versulting in the possibility of an achivement by a local martingale of an arbitrage value (as in the

Example. Local martingale $\mu = (\mu_t)_{0 \leq t \leq 1}$ with

$$\mu_t = \int_0^t \frac{I(u \leq \tau)}{1-u} dW_u,$$

W - a standart Wiener process and

$$\tau = \inf \left\{ t : t > 0, \int_0^t \frac{dW_u}{1-u} = \alpha \right\}$$

has the limit $\mu_1 = \lim_{t \uparrow 1} \mu_t = \alpha$ P -a.s. for arbitrage α . Hence $\mu \notin \mathcal{M}^c$ - is not uniformly integrable).

§ 3. The arbitrage possibilities

The arbitrage possibilities of arbitrage in the case of fixation (1): $S_1 = \exp\{c\}$ depended upon the value of c . So we have to investigate three separate cases $c > 0$, $c = 0$, and $c < 0$ resulting in the following

Theorem. 1) Under the condition of fixation (1) $S_1 = \exp\{c\}$ the arbitrage strategy $\Gamma = (\Gamma_t)_{0 \leq t \leq 1}$ in the case

(a) $c < 0$ exists

(b) $c = 0$ exists except the situation of $B_t \equiv S_t \equiv 1, t \in [0, 1]$

(c) $c > 0$ exists when $[p, p]_1$ is strictly uniformly

(*) positive: $[p, p]_1 > \epsilon > 0$ P -a.s.

2) The strategy can be constructed for $c \neq 0$ as

$$\Gamma_t = - \left(\frac{S_t}{B_t} \right)^v \exp \left\{ - \frac{v(v-1)}{2} [p, p]_t \right\} \quad (22)$$

with $v = 1$ for $c < 0$ and $v = 1 + 2c/\epsilon$ for $c > 0$. In the case of $c = 0$

$$\Gamma_t = \begin{cases} -\frac{S_t}{B_t} & \text{if } P\{B_1 > 1\} > 0 \\ 1 & \text{if } P\{B_1 > 1\} = 0 \text{ and } P\{[p, p]_1 > 0\} > 0 \\ \frac{da_t}{d\text{Var}_t a} & \text{if } P\{B_1 > 1\} = 0 \text{ and } P\{[p, p]_1 > 0\} = 0 \end{cases} \quad (23)$$

3) In the case $c \neq 0$ the strategy Γ is admissible if martingale part Γ^c of Γ is a continuous martingale: $\Gamma^c \in \mathcal{M}^c(\mathcal{B})$. For $c = 0$ in the case $P\{B_1 > 1\} = 0$ presented in (23) strategies are admissible, and in the case $P\{B_1 > 1\} > 0$ Γ is admissible under the condition of $\Gamma^c \in \mathcal{M}^c(\mathcal{B})$.

The proof of this theorem consists of the three separate statements devoted to the cases $c < 0$, $c = 0$ and $c > 0$.

Example. Let $B_t \equiv 1, S_t = \exp\{x_t\}$ where $x = (x_t)_{0 \leq t \leq 1}$ is a process of Brownian Gridge with terminal c :

$$dx_t = \frac{c - x_t}{1 - t} dt + dW_t, \quad x_0 = 0$$

with $W = (W_t)_{t \geq 0}$ - a standart Wiener process. Then $[p, p]_t = t$, $\epsilon < 1$, and for $c > 0$

$$\Gamma_t = -\exp\{vx_t\} \exp\left\{-\frac{v(v-1)}{2}t\right\}$$

with $v = 1 + 2c/\epsilon$, and

$$\Gamma_t^2 \geq \exp\{2vx_t\}$$

with Gaussian $x_t \sim \mathcal{N}(tc, t(1-t))$. Hence the martingale part Γ^c of the semimartingale Γ is uniformly integrable.

Proof of the theorem.

(a) The possibilities of arbitrage are evident in the case of negative constant $c < 0$, because then $S_1 < S_0 = 1$, and we can set the portfolio $(\beta_t, \gamma_t) \equiv (-1)$ for all $t \in [0, 1]$, that results in the finite capital $X_1 = B_1 - S_1$ and for $Y_1 = \frac{X_1}{B_1}$

$$Y_1 = 1 - \frac{X_1}{B_1} > 0 \quad \mathbf{P} - \text{a.s.} \quad (24)$$

In this case in reality we suppose that the strategy $\Gamma = (\Gamma_t)_{0 \leq t \leq 1}$ is equal to

$$\Gamma_t = -z_t \quad (25)$$

where $z = (z_t)_{0 \leq t \leq 1}$ is a solution of an equation

$$dz_t = z_t(dp_t - db_t), \quad z_0 = 1 \quad (26)$$

and brence

$$z_t = \exp\left\{p_t - \frac{1}{2}[p, p]_t - b_t\right\} = \frac{S_t}{B_t} \quad (27)$$

$$\Gamma_t = -\exp \left\{ p_t - \frac{1}{2}[p, p]_t - b_t \right\}. \quad (28)$$

This results in

$$Y_t = -(z_t - z_0) = \Gamma_t - \Gamma_0 \quad (29)$$

and a proper restriction for the positive numeraire $\mathbf{P}\{B_1 > 1\} > 0$ coincides with the previous one: for Γ as in (27) $Y_1 = 1 - \frac{1}{B_1}$ and $\mathbf{P}\{Y_1 > 1\} > 0$. The condition of admission of Γ is analogous.

If $B_1 = 1$ $\mathbf{P}$ -a.s., then

$$Y_t = \int_0^t \Gamma_u dp_u$$

and it is obvious that $\Gamma_u \equiv 1$ due to the condition (1) results in

$$Y_1 = p_1 = \frac{1}{2}[p, p]_1.$$

So this strategy satisfies the demand of the p such that $\mathbf{P}\{[p, p]_1 > 0\} > 0$. In the case $B_1 = 0$ and $[p, p]_1 = 0$ $\mathbf{P}$ -a.s. we have

$$Y_t = \int_0^t \Gamma_u da_u.$$

Because $a = (a_t)_{0 \leq t \leq 1}$ is the process of locally bounded variation the strategy Γ with

$$\Gamma_u = \frac{da_u}{d\text{Var}_u a}$$

and values $\Gamma_u \in [-1, +1]$ is well defined, and then $Y_1 = \text{Var}_1 a$ shows that the arbitrage is impossible in the case (b) only for $S_t \equiv B_t \equiv 1$. The statements of case (b) are proved.

(c) The investigation of the case $c > 0$ is analogous to that of (a) with the strategy $\Gamma = (\Gamma_t)_{0 \leq t \leq 1}$ as in (25). But instead of (26) we shall define z as a solution of

$$dz_t = vz_t(dp_t - db_t), \quad z_0 = 1 \quad (30)$$

with some constant $v > 1$. Then (compare with (27))

$$z_t = \exp \left\{ vp_t - \frac{v^2}{2}[p, p]_t - vb_t \right\} = \left(\frac{S_t}{B_t} \right)^v \exp \left\{ -\frac{v(v-1)}{2}[p, p]_t \right\} \quad (31)$$

and due to (15)

$$Y_t = - \int_0^t z_u (dp_u - db_u) = \frac{1}{v} (z_0 - z_t) = \frac{1}{v} (\Gamma_t - \Gamma_0) \quad (32)$$

The formula (32) with the condition of fixation gives

$$\begin{aligned} Y_1 &= \frac{1}{v} (1 - z_1) = \frac{1}{v} \left(1 - \exp \left\{ vp_1 - \frac{v^2}{2} [p, p]_1 - vb_1 \right\} \right) \\ &= \frac{1}{v} \left(1 - \exp \left\{ v \left(c - \frac{v-1}{2} [p, p]_1 - b_1 \right) \right\} \right) \end{aligned} \quad (33)$$

It is clear that for such a strategy the arbitrage is possible under the of $[p, p]_1$: $\exists \epsilon > 0$ such that $\mathbf{P}\{[p, p]_1 > \epsilon\} = 1$. Under this condition we can choose $v = 1 + \frac{2\epsilon}{c}$ and receive

$$v \left[c - \frac{v-1}{2} [p, p]_1 - b_1 \right] = v \left[c - c \frac{[p, p]_1}{\epsilon} - b_1 \right] < 0 \quad \mathbf{P} - \text{a.s.}$$

and hence (see (33)) $Y_1 > 0$ $\mathbf{P}$ -a.s.. Appropriate choice of $\Gamma^c \in \mathcal{M}^c(\mathcal{B})$ is clear. The theorem is proved. $\square$

References

1. *F. Delbaen and W. Schachermayer*. The no-arbitrage property under a change of numeraire.
2. *C. Delacherie*. Capacities et processus stochastique. Springer, Berlin Heidelberg and New York (1972)
3. *J. Jacod, A.N. Sircyev*. Limit theorems for stochastic process. Springer, Berlin Heidelberg and New York (1987)
4. *H. Fölmer and P. Imkeller*. Anticipation cancelled by a Girsanov transformation: a paradox on Wiener space. Ann. Inst. Henri Poincaré. Probabilités et Statistiques, Vol.29, N4, 1993, p.569-586.

В. К. Горбунов, И. В. Лутошкин.**Вторые производные параметризованной задачи оптимального управления**

1. В [1] были разработаны основы метода параметризации (конечномерной редукции) задач оптимального управления достаточно общего вида. При этом управляющая функция представляется в параметрической форме с подвижными моментами переключений и исходная функциональная задача сводится к конечномерной задаче нелинейного программирования. Вычисление целевой и ограничивающих функций и их производных заключается в решении задач Коши для исходного и сопряженных дифференциальных уравнений. Этот подход оказался также перспективным и для развития методов оптимизации второго порядка (со вторыми производными). Последнее может оказаться эффективным направлением для решения задач с особыми управлениями [2], в частности, возникающими в связи с задачами с промежуточными фазовыми ограничениями [3]. Современный интерес к ним стимулируется экономическими проблемами [4,5].

Данная статья посвящена развитию метода [1].

2. Рассмотрим задачу оптимального управления, ограничившись здесь случаем автономной системы, закрепленного начального состояния и терминального функционала:

$$\dot{x} = f(x, u), \quad x(t_0) = x^0, \quad t_0 \leq t \leq T; \quad (1)$$

$$u(t) \in U; \quad (2)$$

$$g_l(x(T)) \leq 0, \quad l = 1, \dots, m; \quad (3)$$

$$J = g_0(x(T)) \rightarrow \min. \quad (4)$$

Фазовая переменная $x = (x_1, \dots, x_n)$, вектор параметров управления $u = (u_1, \dots, u_r)$. Функции $f_i(x, u)$, $1 \leq i \leq n$, и $g_l(z)$, $1 \leq l \leq m$, $z = (z_1, \dots, z_n)$, будем считать дважды непрерывно дифференцируемыми по всем переменным. Множество допустимых значений управления U должно иметь структуру, позволяющую применять методы дифференцируемой оптимизации. Время окончания процесса T может быть свободным и задача (1)-(4) считается разрешимой в классе кусочно-непрерывных функций $u(t)$.

Метод [1] заключается во введении произвольного разбиения промежутка $[t_0, T]$

$$t_0 \leq t_1 \leq \dots \leq t_N \equiv T, \quad (5)$$

и закреплении структуры управления на промежутках $[t_{k-1}, t_k]$, $1 \leq k \leq N$. Ограничимся здесь кусочно-постоянным управлением:

$$u(t) = v^k \in U, \quad t_{k-1} \leq t < t_k, \quad k = 1, \dots, N. \quad (6)$$

При подстановке параметризованного управления (6) в (1) мы получим траекторию $x(t)$, зависящую от параметров управления

$$w^k = (t_k, v^k) \equiv (w_0^k, w_1^k, \dots, w_r^k),$$

$$x(t) = z(t; v^1, t_1, \dots, v^{k-1}, t_{k-1}, v^k), \quad t_{k-1} \leq t < t_k.$$

Функция $z(t; v^1, \dots, t_k, v^{k+1})$ определяется на промежутках $[t_k, t_{k+1})$ интегральными соотношениями, эквивалентными в совокупности задаче Коши (1):

$$\begin{aligned} z(t, v^1) &= x^0 + \int_{t_0}^t f(z(s; v^1), v^1) ds, \quad t_0 \leq t < t_1; \\ z(t; w^1, \dots, w^k, v^{k+1}) &= z(t_k; w^1, \dots, v^k) + \\ &+ \int_{t_k}^t f(z(s; w^1, \dots, v^{k+1}), v^{k+1}) ds, \quad t_k \leq t < t_{k+1}. \end{aligned} \quad (7)$$

Введем функции от управляющих параметров $\{w^k\}$:

$$\varphi_l(w^1, \dots, w^N) = g_l(z(T; w^1, \dots, w^{N-1}, v^N)), \quad l = 0, 1, \dots, m. \quad (8)$$

В терминах этих функций задача (1)-(4) принимает форму задачи нелинейного программирования:

$$\min\{\varphi_0(w^1, \dots, w^N) : \varphi_l(w^1, \dots, w^N) \leq 0, \quad 1 \leq l \leq m, (5), (6)\} \quad (9)$$

3. При сделанных предположениях и при разрешимости задачи Коши (1), (6) функции (8) являются дважды непрерывно дифференцируемыми и к задаче (9) можно применять такие эффективные методы, как квадратично-линейная аппроксимация и метод модифицированных функций Лагранжа [6].

Для вывода формул и уравнений дифференцирования функций (8) будем использовать векторно-матричные обозначения, сделав

следующие соглашения. Производные скалярных функций по векторным аргументам

$$\frac{\partial g_l(z)}{\partial z} = \left[\frac{\partial g_l}{\partial z_1}, \dots, \frac{\partial g_l}{\partial z_n} \right]$$

будем понимать как векторы-строки, а производные векторной функции по скалярному параметру

$$\frac{\partial z(t; w^1, \dots, w^k)}{\partial w_\mu^k} = \left[\frac{\partial z_1}{\partial w_\mu^k}, \dots, \frac{\partial z_n}{\partial w_\mu^k} \right]^T \quad (10)$$

($1 \leq \mu \leq r$, $1 \leq j \leq k$), как векторы-столбцы. Соответственно этому будет определяться матричная структура производных векторных функций по векторным аргументам.

Продифференцируем равенство (8) по одному из параметров w_μ^k :

$$\frac{\partial \varphi_l(w^1, \dots, w^N)}{\partial w_\mu^k} = \frac{\partial g_l(z(T; \dots, v^N))}{\partial z} \frac{\partial z}{\partial w_\mu^k}. \quad (11)$$

Производные (10) - это вариации траектории системы (1), (6) по параметрам, определяющим управление. Обозначим их

$$y^{j\mu}(t) = \frac{z(t; v^1, \dots, v^k)}{\partial w_\mu^k}, \quad t_{k-1} \leq t < t_k, \quad 1 \leq j \leq k \leq N. \quad (12)$$

Дифференцируя соотношения (7) по соответствующим переменным, получим задачи Коши, определяющие функции (12). Для вариаций, отвечающих параметрам $w_0^k = t_k$, $1 \leq k \leq N-1$, это [1]

$$\begin{aligned} \dot{y}^{k0}(t) &= \frac{\partial f(x(t), u(t))}{\partial x} y^{k0}, \quad t_k \leq t \leq T, \\ y^{k0}(t_k) &= f(x(t_k), v^k) - f(x(t_k), v^{k+1}); \end{aligned} \quad (13)$$

и для вариаций относительно v_μ^k ($1 \leq \mu \leq r$, $1 \leq k \leq N$) - это

$$\begin{aligned} \dot{y}^{k\mu}(t) &= \frac{\partial f(x(t), u(t))}{\partial x} y^{k\mu} + \theta(t_k - t) \frac{\partial f}{\partial u_\mu}, \quad t_{k-1} \leq t \leq T, \\ y^{k\mu}(t_{k-1}) &= 0. \end{aligned} \quad (14)$$

Здесь и далее функция Хевисайда

$$\theta(t) = \begin{cases} 0, & t \leq 0, \\ 1, & 0 < t. \end{cases}$$

Вариация по конечному времени T конечна:

$$y^{N0}(T) = f(x(T), u(T)). \quad (15)$$

В принципе соотношения (11)–(14) решают проблему первых производных для задачи (9). Для этого следует решить сначала задачу Коши (1), (6), а затем $N(r+1)-1$ задач (13), (14). Однако трудоемкость этой процедуры существенно сокращается с помощью сопряженных переменных.

Определим функцию Гамильтона-Понтрягина

$$H(p, x, u) = \sum_{i=1}^n p_i f_i(x, u),$$

и введем для каждой функции $g_l(z)$ исходной задачи свою сопряженную вектор-функцию $p^l(t) = (p_1^l(t), \dots, p_n^l(t))$:

$$\begin{aligned} \dot{p}^l(t) &= -\frac{\partial H(p^l, x(t), u(t))}{\partial x}, \quad t_0 \leq t \leq T, \\ p^l(T) &= \frac{\partial g_l(x(T))}{\partial z}, \quad l = 0, 1, \dots, m. \end{aligned} \quad (16)$$

Кроме того, введем обозначение

$$M_l(t) = H(p^l(t), x(t), u(t)). \quad (17)$$

Используя свойство постоянства скалярных произведений $\langle p^l(t), y^{k\mu}(t) \rangle$ на общих промежутках определения $[t_k, T]$, конечные и начальные условия, нетрудно получить формулы [1]:

$$\begin{aligned} \frac{\partial \varphi_l(w^1, \dots, w^N)}{\partial t_k} &= M_l(t_k - 0) - M_l(t_k + 0), \quad 1 \leq k \leq N-1; \\ \frac{\partial \varphi_l(w^1, \dots, w^N)}{\partial T} &= M_l(T); \\ \frac{\partial \varphi_l}{\partial v_\mu^k} &= \int_{t_{k-1}}^{t_k} \frac{\partial H(p^l(t), x(t), v^k)}{\partial u_\mu} dt, \quad 1 \leq k \leq N, \quad 1 \leq \mu \leq r. \end{aligned} \quad (18)$$

Теперь для вычисления производных (18) требуется решение помимо основной задачи Коши (1), (6) дополнительно $m+1$ задачу (16). Это число определяется количеством терминальных условий (3) и не зависит от размерности параметризованного управления (5), (6), которое в сложных случаях может быть большим.

4. Изложенная методика дифференцирования функций конечномерной задачи (9) естественно развивается для вторых производных:

$$\begin{aligned} \frac{\partial^2 \varphi_l(w^1, \dots, w^N)}{\partial w_\nu^k \partial w_\mu^j} &= \left[\frac{\partial^2 g_l(x(T))}{\partial z^2} y^{k\mu}(T) \right]^T y^{j\nu}(T) + \\ &+ \frac{\partial g_l(x(T))}{\partial z} \frac{\partial^2 z}{\partial w_\nu^k \partial w_\mu^j}(T; w^1, \dots, v^N), \\ 0 \leq l \leq m \quad (0 \leq \mu, \nu \leq r; \quad 1 \leq j, k \leq N). \end{aligned} \quad (19)$$

Здесь появляются вторые производные траектории по параметрам (вторые вариации траектории), для которых введем обозначения

$$\begin{aligned} Z^{j\mu k\nu}(t) &= \frac{\partial^2 z(t; w^1, \dots, v^s)}{\partial w_\mu^j \partial w_\nu^k}, \quad t_s \leq t < t_{s+1}, \\ \max(j, k) \leq s \leq N-1; \quad 0 \leq \mu, \nu \leq r; \quad 1 \leq j, k \leq N. \end{aligned} \quad (20)$$

Вторые вариации, как и первые, удовлетворяют соответствующим задачам Коши, получаемым двукратным дифференцированием уравнений (7). Для вариаций по паре параметров $(v_\mu^j, v_\nu^k; j \leq k)$ это будет (векторное) дифференциальное уравнение

$$\begin{aligned} \dot{Z}^{j\mu k\nu} &= \frac{\partial f(x(t), u(t))}{\partial x} Z^{j\mu k\nu} + [y^{j\mu}(t)]^T \frac{\partial^2 f}{\partial x^2} y^{k\nu}(t) + \theta(t_k - t) \\ &\quad \left\{ \frac{\partial^2 f}{\partial u_\nu \partial x} y^{j\mu}(t) + \right. \\ &\quad \left. + \delta_{jk} \left[\frac{\partial^2 f}{\partial u_\mu \partial x} y^{k\nu}(t) + \frac{\partial^2 f}{\partial u_\mu \partial u_\nu} \right] \right\}, \quad t_{k-1} \leq t \leq T, \end{aligned} \quad (21)$$

с начальным условием $Z^{j\mu k\nu}(t_{k-1}) = 0$.

Смешанные по временному и v -параметрам вариации описываются в зависимости от чередования возмущений. Вариация типа $(v_\mu^j, t_k; j \leq k)$ описывается задачей

$$\begin{aligned} \dot{Z}^{j\mu k0}(t) &= \frac{\partial f(x(t), u(t))}{\partial x} Z^{j\mu k0} + [y^{j\mu}(t)]^T \frac{\partial^2 f}{\partial x^2} y^{k0}(t), \quad t_k \leq t \leq T, \\ Z^{j\mu k\nu}(t_k) &= \frac{\partial}{\partial x} [f(x(t_k), v^k) - f(x(t_k), v^{k+1})] y^{j\mu}(t_k) + \\ &+ \delta_{jk} \frac{\partial f(x(t_k), v^k)}{\partial u_\mu}. \end{aligned} \quad (22)$$

Вариация параметров $(t_j, v^{kv}; j < k)$ приведет к задаче

$$\dot{Z}^{j0k\nu} = \frac{\partial f(x(t), u(t))}{\partial x} Z^{j0k\nu} + [y^{j0}(t)]^T \frac{\partial^2 f}{\partial x^2} y^{kv}(t) + \theta(t_k - t) \frac{\partial^2 f}{\partial u_\nu \partial x} y^{j\nu}(t),$$

$$t_{k-1} \leq t \leq T, \quad Z^{j0k\nu}(t_{k-1}) = -\delta_{j(k-1)} \frac{\partial f(x(t_{k-1}), v^k)}{\partial u_\nu}. \quad (23)$$

Вторые вариации Z^{j0k0} описаны в [1].

Сокращение трудоемкости вычисления вторых производных (19) на основе исключения вторых вариаций траектории достигается введением сопряженных матричных функций (матричных импульсов [2]) для каждой из терминальных функций $g_l(z)$. Обозначим эти матричные импульсы

$$\Psi^l(t) = \{\Psi_{\alpha\beta}^l(t) : \alpha, \beta = 1, \dots, n\}$$

и определим их с помощью сопряженных функций p^l :

$$\dot{\Psi}^l = -\frac{\partial f(x(t), u(t))}{\partial x} \Psi^l - \Psi^l \frac{\partial f}{\partial x} - \frac{\partial^2 H(p^l(t), x(t), u(t))}{\partial x^2},$$

$$0 \leq t \leq T; \quad (24)$$

$$\Psi^l(T) = \frac{\partial^2 g_l(x(T))}{\partial x^2}. \quad (25)$$

Функции $\{p^l(t), \Psi^l(t)\}$ в совокупности являются сопряженными каждому набору вариаций $\{Z^{j\mu k\nu}(t), y^{j\mu}(t), y^{kv}(t)\}$ в смысле постоянства значений скалярных функций

$$h_{j\mu k\nu}^l(t) = \sum_{\alpha=1}^n \left[p_\alpha^l(t) Z_\alpha^{j\mu k\nu} + \sum_{\beta=1}^n \Psi_{\alpha\beta}^l(t) y_\beta^{j\mu}(t) y_\alpha^{kv}(t) \right] \quad (26)$$

на промежутке $[t_k, T]$. Это постоянство проверяется дифференцированием равенства (26).

Из определения (26), конечных условий (25), (16) и формулы (19) следует равенство

$$\frac{\partial^2 \varphi_l(w^1, \dots, w^N)}{\partial w_\mu^j \partial w_\nu^k} = h_{j\mu k\nu}^l(T).$$

Используя это, дифференциальные уравнения и начальные условия для вторых вариаций (21)-(23), можно убедиться в справедливости следующих формул (везде $j \leq k$):

$$\frac{\partial^2 \varphi_l}{\partial v_\mu^j \partial v_\nu^k} = \int_{t_{k-1}}^{t_k} \left\{ \frac{\partial^2 H(p^l(t), x(t), u(t))}{\partial u_\nu \partial x} y^{j\mu}(t) + [\Psi^l(t) y^{j\mu}]^T \frac{\partial f}{\partial u_\mu} + \right.$$

$$\left. + \delta_{jk} \left[\frac{\partial^2 H}{\partial u_\nu \partial u_\mu} + \frac{\partial^2 H}{\partial u_\mu \partial x} y^{kv}(t) + [\Psi^l(t) y^{kv}(t)]^T \frac{\partial f}{\partial u_\nu} \right] \right\} dt. \quad (27)$$

Пусть $j \leq k$, тогда

$$\frac{\partial^2 \varphi_l}{\partial v_\mu^j \partial t_k} = \left[\frac{\partial H(p^l(t_k), x(t_k), v^k)}{\partial x} - \frac{\partial H(p^l(t_k), x(t_k), v^{k+1})}{\partial x} \right]^T y^{j\mu}(t_k) +$$

$$+ [\Psi(t_k) y^{j\mu}(t_k)]^T y^{k0}(t_k) +$$

$$+ \delta_{jk} \frac{\partial H(p^l(t_k), x(t_k), v^k)}{\partial u_\mu}. \quad (28)$$

Если $j = k - 1$, то

$$\frac{\partial^2 \varphi_l}{\partial t_j \partial v_\nu^k} = -\frac{\partial H(p^l(t_{k-1}), x(t_{k-1}), v^k)}{\partial u_\nu}, \quad (29)$$

$j < k - 1$, то

$$\frac{\partial^2 \varphi_l}{\partial t_j \partial v_\nu^k} = \int_{t_{k-1}}^{t_k} \frac{\partial^2 H}{\partial u_\nu \partial x} y^{j0}(t) + [\Psi y^{j0}(t)]^T \frac{\partial f}{\partial u_\nu}. \quad (30)$$

Формулы для вторых производных φ_l по $\{t_j, t_k\}$ были получены в [1].

Итак, вычисление вторых производных функций конечномерной задачи требует ряд решений задач Коши. Кроме основной задачи Коши (1), (6) требуется также решить $N(r+1) - 1$ задач (13), (14), $m+1$ задач (16) и $m+1$ задач (24), (25). Таким образом, вычисление вторых производных для редуцированной задачи в общем случае существенно сложнее, чем вычисление первых производных. Если функции исходной задачи линейны по фазовым переменным, система (24), (25) становится тривиальной. Это существенно упрощает вычисление вторых производных.

Работа поддержана РФФИ (грант № 96-01-00509).

Литература

1. Горбунов В.К. Метод параметризации задач оптимального управления // Ж. вычисл. матем. и матем. физ. 1979. Т.19. N2. С. 292-303.
2. Габасов Р., Кириллова Ф.М. Особые оптимальные управления. М.: Наука, 1973.
3. Горбунов В.К. Задачи оптимального управления с фазовыми ограничениями и особые управления // Дифф. уравнения и их приложения: тезисы докл. 1-й междун. научно-практ. конф. С-Пб. 1996, С. 58.
4. Тер-Крикоров А.М. Оптимальное управление и математическая экономика. М.: Наука, 1977.
5. Blok M. Dynamic Models of the Firm. Berlin: Springer, 1996.
6. Гилл Ф., Мюррей У., Райт М. Практическая оптимизация. М.: Мир, 1985.

Sh. Ishmukhametov.

On degrees without strong minimal covers

Degree m is a strong minimal cover (s.m.c.) for a degree a , if $a < m$, and for any degree b , $b < m \rightarrow b \leq a$. We develop a new method for constructing turing degrees without s.m.c. and we show that there is a low r.e. degree a , which has no a s.m.c. below $0'$. Then we extend this result to proper classes of the Ershov hierarchy of n -r.e. degrees.

Strong minimal covers play an important role in the theory of degrees D_T for any reducibility r . For example in m -degrees when each degree has a s.m.c. it was possible to prove the isomorphism of any upper cones, extending up to s.m.c. a mapping beginning at bases of these cones (Lachaln [1972], Ershov, Palutin [1975]). The number of new methods were developed in turing degrees to construct minimal degrees and minimal covers in order to characterize partial orderings embeddable into D_T as initial segments. The problem of existence of s.m.c. for turing degrees is connected also with the problem of the non-rigidability of D_T , that is, the existence of non-trivial automorphisms of D_T .

The problem of description of degrees possessing s.m.c. is very complex. The degree 0 is such since there exist minimal degrees. Titgemeyer [1962] showed that three-element chain is embeddable as initial segment of D and such showed that minimal degrees can have s.m.c. Cooper [1971], [1995] constructed a r.e. degree possessing a s.m.c. and Kumabe showed that 1-generic degrees can be such. The latter is especially interesting since by Jockusch [1980] and Haught [1986] each 1-generic degree is r.e. in a lesser 1-generic degree, so by the relativization of Sacks Splitting Theorem no 1-generic degree below $0'$ can be a strong minimal cover.

On the opposite side by Friedberg Inversion Theorem no degree greater than or equal to $0'$ can be a s.m.c. Posner [1977] showed that each high degree (i.e. a degree a with $a' = 0''$) is r.e. in a low degree (a degree b with $b' = 0'$), and so cannot be a s.m.c. (and have a s.m.c. below $0'$).

Jockusch and Posner [1978] showed that all degrees below $0'$ either are low_2 (i.e. satisfy $a'' = 0''$), or have a 1-generic predecessor. Therefore all strong minimal covers for minimal degrees below $0'$ are low_2 .

In [1970] Yates asked, whether each minimal degree has a s.m.c. From those time there was no progress in the decision of this problem. In [1989], p. 527, P. Odifreddi remarked that the construction of minimal degrees without s.m.c. requires new methods of proof. More generally there is no an example of non-trivial definable class of turing degrees which each representative has a s.m.c. We show below that such class cannot contain the class of low degrees or its intersection with n -r.e. degrees (or with α -r.e. degrees if we consider the extension of this hierarchy on constructive ordinal, see Epstein, Haas and Kramer [1981]). Li Angsheng proved that the class of degrees possessing s.m.c. has non-empty intersection with the class of proper n -r.e. degrees for any n (see also Ishmukhametov [ta]).

Our notations are mostly standard and follows Soare [1987]. We denote partial recursive functionals by capital Greek letters, and their uses are denoted by corresponding lower case Greek letters. We assume that all uses are non-decreasing at any of their arguments, and at a stage s all parameters of computations do not exceed s .

Degrees without strong minimal covers

Fix a creative set K and its enumeration. In our construction we deal with partial recursive approximation of Δ_0^2 -sets defined by their reducing procedures to K . We need in few additional definitions.

Definition. Let A^s be a partial recursive approximation for a set A , i.e. such that for each x there is s_0 , $A^s(x) \downarrow = A(x)$ for all $s > s_0$. We say, $A^s(x)$ changes at stage s , if $A^s(x) \neq A^t(x)$, where t is the largest stage, less s , at which $A^t(x) \downarrow$.

Let $pr_A(s, x)$ denote the largest stage $t < s$, at which $A^t(x)$ changed, if such t exists, and 0, otherwise.

Theorem 1. *There is a low r.e. degree a which has no a strong minimal cover below a' .*

Proof. We construct a low r.e. set A such a way that for any Δ_0^2 -set K_e , if $K_e \not\leq_T A$, then there exists a set $M_e <_T A \oplus K_e$, incomparable with A . We use the following list of requirements:

$$\begin{aligned} P_e &: A \neq \Theta_e, \quad e \in \omega, \\ S_{i,j} &: \Phi_i^A(j) \uparrow \rightarrow \exists s_0 \forall s > s_0 \Phi_{i,s}^{A^s}(j) \uparrow \\ \Lambda_e &: K_e = \Psi^K \wedge K_e \not\leq_T M \rightarrow \exists M_e <_T A \oplus K_e \quad A \upharpoonright_T M_e, \end{aligned}$$

where $\{\Phi_e, \Psi_e, \Theta_e\}_{e \in \omega}$ is a recursive enumeration of all possible triples of primitive recursive functionals, K is a creative set.

Each requirement Λ_e we break up into an infinite list of subrequirements:

$$\begin{aligned} R_{e,i} &: M_e \neq \Psi_i^A, \\ Q_{e,j} &: A \neq \Psi_j^{M_e} \end{aligned}$$

Below we describe basic modules for each kind of requirements:

The module for a requirement P alone follows Friedberg and Muchnik:

- (1) Choose a fresh witness x .
- (2) Wait for $\Theta(x) \downarrow = 0$.
- (3) Enumerate x into A .

The module for a requirement $S_{i,j}$ is standard:

- (1) Wait for $\Phi_i^A(j) \downarrow$.
- (2) Restrain A up to $\phi(j)$.
- (3) Wait for $\Phi_i^A(j) \uparrow$, then return to step 1.

The module for a requirement $R_{e,i}$ consists of procedures. Procedure $\langle 0 \rangle$ starts first. Each procedure $\langle n+1 \rangle$ is started by the procedure $\langle n \rangle$. Consider the instructions for a procedure $\langle n \rangle$:

- (1) Choose fresh witnesses y_n for A and $x_n > \max\{n, y_n, pr_{K_e}(s, n)\}$ for M_e .
- (2) Wait for $M_e \upharpoonright x_n + 1 = \Psi_i^A \upharpoonright x_n + 1$. If $K_e(n)$ changes earlier, initialize witnesses and return to step 1. Otherwise, define $\Sigma^A(n) = K_e(n)$ with use $\sigma(n) = \max\{y_n, \psi(x_n)\}$. Start procedure $n+1$.
- (3) Wait for $K_e(n)$ or $A \upharpoonright \sigma(n)$ to change. Then stop procedures $> n$. If the latter occurs first, return to step 2, otherwise go to the next step.
- (4) Change $M_e(x_n)$.

Beginning from this moment carry out the motion procedure for x_n , changing $M_e(x_n)$ each time when $K_e(n)$ changes. Redefine $\Sigma^A(n) = K_e^s(n)$ each time when we see $\Sigma^A(n) \uparrow$ and $M_e \upharpoonright x_n + 1 = \Psi_i^A \upharpoonright x_n + 1$, increasing use $\sigma(n)$, when $\psi(x_n)$ has increased.

Delay procedures $> n$, except appropriate motion procedures, when $K_e(n) \neq \Sigma^A(n)$, and continue at stages, when $K_e(n) = \Sigma^A(n)$.

If procedure $\langle n \rangle$ fails to satisfy $R_{e,i}$ and $K_e(n) \downarrow$, then $K_e(n) = \Sigma^A(n)$. If K_e is total, and each procedure fails to satisfy $R_{e,i}$, then $K_e \leq_T A$, and the overall requirement Λ_e is satisfied.

Notice that after enumerating x_n into M_e , we must change $M_e(x_n)$ each time when $K_e(n)$ changes independently procedure is active or

stopped by a procedure with lesser number, otherwise, M_e is not recursive in $A \oplus K_e$. This can cause injury to higher priority requirements. In order to prevent it, we keep the element y_n outside A as a "killing point" and enumerate it into A when we need to cut a non-desirable connection between $K_e(n)$ and $M_e(x_n)$ for a sake of a higher priority requirement.

(5) If later y_n is enumerated into A , cut the association between $K_e(n)$ and $M_e(x_n)$ and stop to change $M_e(x_n)$ simultaneously with $K_e(n)$.

The module for a requirement $Q_{e,j}$:

(1) Choose a fresh witness z .

(2) Wait for $A \uparrow z + 1 = \Psi_j^{M_e} \uparrow z + 1$. Consider two possible cases:

Case A. There is no $M_e(x_n), x_n < \psi(z)$, associated with $K_e(n), n \in \omega$. Enumerate z into A , and restrain M_e up to $\psi(z)$.

Case B. There are $M_e(x_n), x_n < \psi(z)$, associated with $K_e(n), n \in \omega$.

Enumerate into A each "killing point" y_n with $x_n < \psi(z)$, which are not restrained by higher priority requirements. Cut connection between $M_e(x_n)$ and $K_e(n)$. Then enumerate z into A and restrain M_e up to $\psi(z)$.

If later $M_e \uparrow \psi(z)$ changes, then return to step 1, and begin the procedure over again.

Below we shall prove that our requirements are satisfied:

Lemma 1. *If K_e is total, then M_e is uniformly recursive in $K_e \oplus A$.*

Proof. In order to know, if $z \in M_e$, find the least stage s_0 such that for all $n < z$ $K_e^s(n) = K_e(n)$. If z was connected with some $K_e(n)$ before stage s_0 , then the value $M_e(z)$ changes simultaneously with $K_e(n)$, and this connection can be cut only if the killing point y_n is put into A . So the final value $M_e(z)$ can be found knowing final values $K_e(n)$ and $A(y_n)$. If $M_e(z)$ is connected with some $K_e(n)$ at a later stage $s > s_0$, then $n < z$ and $K_e(n)$ does not change between stages s_0 and s , otherwise $pr_{K_e}(s, n) > z$ and at stage s we cannot connect $M_e(z)$ with $K_e(n)$. Then $K_e^s(n) = K_e(n)$, and $M_e(z) \neq M_e^s(z)$ only if we cut an association between $K_e(n)$ and $M_e(z)$ at stage, when a killing point y_n is enumerated at A . Since a killing point y for z cannot exceed z , so find a stage $s_1 > s_0$, at which all numbers of A less z have been enumerated into A . If at stage s_1 , there is a connection of $M_e(z)$ with $K_e(n)$, then then the final value of $M_e(z)$ depends on the final value of $K_e(n)$, otherwise, $M_e(z)$ does not change after stage s_1 .

Lemma 2. *(The Outcome Lemma.) Each requirement is satisfied and causes a finite injury to lower priority requirements.*

Proof. The proof is evident for requirements P_e and $S_{i,j}$. Assuming the lemma for all previous requirements, consider a requirement $R_{e,i}$.

Case A. K_e is total, and $K_e \not\leq_T A$.

Then, there is a procedure with the least number n such that $K_e(n) \neq \Sigma^A(n)$. Beginning from some stage s_0 $K_e^s(n) \neq \Sigma_s^A(n)$, and new procedures are not started. So only a finite number of procedures are started, and there is a finite number of values $M_e(x_n)$ connected by the module for $R_{e,i}$ with $K_e(n), n \in \omega$. There is a stage $s_1 \geq s_0$, after which no $K_e(n)$, connected with some $M_e(x_n)$, changes, and nothing destroys lower priority requirements.

Case B. K_e is partial. Evidently, all requirements $R_{e,i}, i \in \omega$, and $Q_{e,j}, j \in \omega$, are satisfied.

Notice that the module for $R_{e,i}$ does not change any value $A(y)$ and pose any restrictions on A . So it cannot injury requirements $P, S, R_{j,k}$ and $Q_{j,k}$ with $j \neq e$.

Case C. K_e is total, and $K_e \leq_T A$ via Σ .

Again the overall requirement Λ_e is satisfied, and $R_{e,i}$ does not injury requirements $P, S, R_{j,k}$ and $Q_{j,k}$ with $j \neq e$.

Consider now a requirement $Q_{e,j}$. Assume that all previous requirements restrain a finite segment of A , and put a finite set of numbers into M_e .

If $Q_{e,j}$ is not satisfied and $A = \Psi_j^{M_e}$, then for each witness z there is stage s_z at which $l_Q(e, j, s) > z$. Assume a witness z was chosen after the stage s_0 , at which the last number was enumerated into M_e by the requirements Q of higher than $Q_{e,j}$ priority. At stage s_z , we cut all connections between $K_e(n)$ and $M_e(x_n)$, enumerating into A killing points y_n , not restrained by requirements S of higher priority. By assumption, a fixed number of connections remains after each such enumeration. At stage s_z we satisfy $Q_{e,j}$. Later, the agreement $M(z) = \Psi_j^{M_e}(z)$ can be restored. Since $M_e \uparrow \psi(z)$ is restrained at stage s_z , only elements of M_e connected with some $K_e(n)$ can destroy $\Psi_j^{M_e}(z)$. Let n_0 be the largest n such that the connection of $K_e(n_0)$ with some $M_e(x)$ is restrained by requirements S of higher priority. Repeating the procedure with new z we deal with another distribution of values $K_e(n), n \leq n_0$. When we finish a cycle, we satisfy $Q_{e,j}$ for a new distribution of values $K_e(n), n \leq n_0$. Since there is a finite number of various combinations of $K_e(n), n \leq n_0$, then there

is no more than a finite number of cycles, and $Q_{e,j}$ is satisfied. Notice, that the nodule has a finite outcome independently on K_e is total, partial or reducible to A .

This establishes the proof of the theorem.

Theorem 2. *For any n there is a low properly n -r.e. degree a which has no a strong minimal cover below $0'$.*

Notice that the method of constructing of properly n -r.e. degrees (see Epstein, Haas and Kramer [1981]) is a direct generalization of Cooper method for constructing d.r.e. degrees not equal to any r.e. degree where each requirement is satisfied in a finite number of actions and causes a finite injury to lower priority requirements. Such construction is well compatible with our method. So we drop this proof.

Questions

Below we state open questions connected with this subject. We begin with the reformulation of the Yates problem.

Question 1. *Is it possible to construct a minimal degree without s.m.c. below $0'$?*

Question 2. *Is it possible to define non-trivial class of degrees each member of which has a s.m.c.? Remember that by the mentioned above theorem of Posner there is a definable class (of high degrees) which arbitrary representative has no a strong minimal cover below $0'$.*

Question 3. *Does any jump class contain a representative without s.m.c.?*

Question 4. *Is it possible to locate our construction below arbitrary r.e. degree?*

References

- S.B. Cooper [1996] Strong minimal covers for recursively enumerable degrees, *Mathematical Logic Quarterly*, **42** (1996), pp.191-196
 Y. Ershov [1975] The upper semilattice of numerations of a finite set, *Algebra i Logika* **14**, 258-284 (Russian); *Algebra and Logic* **14**, 159-175 (English translation)

R. Epstein, R. Haas, R. Kramer [1981] Hierarchies of sets and degrees below $0'$, *Logic Year 1979-1980* (M. Lerman, J. Schmerl and R. Soare, editors), *Lecture Notes in Mathematics*, vol. 859, Springer-Verlag, Berlin, pp.32-48

Sh. T. Ishmukhametov [1997] Minimal covers for recursively enumerable degrees, *J. Symb. Log.* to appear

A. H. Lachlan [1972] Recursively enumerable many-one degrees, *Algebra i Logika* **11**, 326-358

P. Odifreddi [1989] *Classical Recursion Theory*, *Studies in Logic and the Foundations of Mathematics*, Vol. 125, North-Holland, Amsterdam.

E. A. Paljutin [1975] Complementations to the paper of Ershov "The upper semilattice of numerations of a finite set," *Algebra i Logika* **14**, 284-287

R. I. Soare [1987] *Recursively Enumerable Sets and Degrees*, *Perspectives in Mathematical Logic*, Springer-Verlag, Berlin, xviii+437 pp. 1987.

C. E. M. Yates [1970] Initial segments of the degrees of unsolvability, Part II: Minimal degrees, *J. Symb. Log.* **35** (1970) pp. 243-266

Л. В. Калинин, А. Г. Сковиков.

Идентификация нарушений в линейных стохастических системах

Рассмотрим задачу оптимального в среднеквадратичном смысле оценивания вектора состояния линейной динамической системы, возмущаемой дискретным белым шумом, свойства которой в произвольные моменты времени могут изменяться. В дальнейшем подобные изменения будем называть нарушениями. В результате каждого нарушения система переходит на новый локально стационарный режим функционирования, описываемый следующими уравнениями:

$$\begin{cases} x_k = \Phi_k^i x_{k-1} + w_k \\ z_k = H_k^i x_k + v_k, \end{cases} \quad (1)$$

где Φ_k^i и H_k^i - известные матрицы перехода и наблюдения; x_k - вектор состояния системы; z_k - доступный для наблюдения вектор измерений; w_k и v_k - независимые нормально распределенные векторы шумов с нулевым средним и известными ковариационными матрицами Q_k^i и R_k^i соответственно; i - номер локально стационарного режима ($i = 0, \dots, N$).

Как известно, эффективность оптимальных алгоритмов оценивания и фильтрации, существенно зависит от степени адекватности принятой модели объекта реальным условиям его функционирования. Следовательно применение известного фильтра Калмана для решения поставленной задачи не обеспечивает заданного качества оценок, а в отдельных случаях может привести к расходимости процесса фильтрации.

Построим математически обоснованное правило обнаружения и идентификации структурных и параметрических изменений в модели объекта, позволяющее принять решение на фиксированном множестве значений функций отношения правдоподобия. Основные выкладки проведем для системы с двумя возможными режимами функционирования ($i = 0, 1$).

Предположим, что начальное состояние системы соответствует режиму функционирования с номером 0. Необходимо по результатам измерений $z_1^k = \{z_1, \dots, z_k\}$ подтвердить или опровергнуть

факт перехода системы на режим функционирования с номером 1.

В работах [1,2] показана возможность решения этой задачи с помощью последовательного критерия отношения вероятностей Вальда, записываемым в следующем виде

$$\begin{cases} 1. \text{ Если } \lambda_k \leq B, \text{ тест прекращают} \\ \quad \text{с выбором гипотезы } G_0 \\ 2. \text{ Если } \lambda_k \geq A, \text{ тест прекращают} \\ \quad \text{с выбором гипотезы } G_1 \\ 3. \text{ Если } B < \lambda_k < A, \text{ тест продолжают} \\ \quad \text{для следующего } (k+1)\text{-го шага} \end{cases} \quad (2)$$

Здесь

$$\lambda_k = \frac{f_{\nu_1^k | G_1}(x_1, \dots, x_k)}{f_{\nu_1^k | G_0}(x_1, \dots, x_k)}$$

отношение функций правдоподобия, где:

G_i - гипотезы, предполагающие функционирование системы в i -ом режиме ($i = 0, 1$);

$\nu_1^k = \{\nu_1, \dots, \nu_k\}$ - последовательность отсчетов процесса обновления, формируемого фильтрами Калмана, построенными в соответствии с различаемыми гипотезами;

$A = (1 - \beta)/\alpha$, - верхний порог решающего правила;

$B = \beta/(1 - \alpha)$; - нижний порог решающего правила;

α и β - вероятности ошибок первого и второго рода соответственно.

Однако, ввиду того, что момент возможного возникновения нарушения априорно неизвестен, приходится вместо одной альтернативной гипотезы G_1 вводить множество гипотез $\{G_{11}, G_{12}, \dots, G_{1k}\}$, предполагающих нарушение в каждый конкретный момент времени с начала наблюдения. С ростом числа измерений k , число альтернативных гипотез также увеличивается, что в свою очередь приводит к росту числа функций отношения правдоподобия и необходимости разработки более сложного и, по существу, эвристического правила распознавания гипотез [1,2]. Отличие предлагаемого в настоящей работе подхода состоит в представлении момента нарушения θ как случайного параметра с равномерным законом распределения.

Согласно теореме умножения вероятностей и условию согласованности функций плотности распределения справедливо соотно-

шение:

$$\begin{aligned} f_{\nu_1^k|G_1}(x_1, \dots, x_k) &= \int_{-\infty}^{+\infty} f_{\nu_1^k, \theta|G_1}(x_1, \dots, x_k, y) dy = \\ &= \int_{-\infty}^{+\infty} f_{\nu_1^k|G_1, \theta}(x_1, \dots, x_k) f_{\theta|G_1}(y) dy \end{aligned} \quad (3)$$

где

$$f_{\theta|G_1}(y) = f_{\theta}(y) = \begin{cases} \frac{1}{k}; & 0 < y < k \\ 0; & \text{иначе} \end{cases}$$

$f_{\nu_1^k, \theta|G_1}(x_1, \dots, x_k, y)$ - условная плотность распределения обобщенного вектора отсчетов процесса обновления ν_1^k , снимаемых с фильтров Калмана, отвечающих гипотезе $G_1 \theta$ - нарушение произошло в момент времени θ . В отношении каждого из альтернативных фильтров Калмана это означает, что он должен активизироваться в соответствующий параметру θ момент времени.

Поскольку в нашем случае θ - дискретная случайная величина, интеграл в выражении (3) следует заменить суммой, а вместо непрерывной функции плотности распределения θ ввести соответствующие вероятности:

$$\begin{aligned} f_{\nu_1^k|G_1}(x_1, \dots, x_k) &= \sum_{i=1}^k f_{\nu_i^k|G_1}(x_i, \dots, x_k) p(\theta = i) = \\ &= \sum_{i=1}^k f_{\nu_i^k|G_1}(x_i, \dots, x_k) \frac{1}{k} \end{aligned}$$

Следовательно отношение функций правдоподобия преобразуется к виду:

$$\lambda_k = \frac{\frac{1}{k} \sum_{i=1}^k f_{\nu_i^k|G_1}(x_i, \dots, x_k) f_{\nu_{i-1}^k|G_0}(x_1, \dots, x_{i-1})}{f_{\nu_1^k|G_0}(x_1, \dots, x_k)} \quad (4)$$

Учитывая независимость отсчетов процесса обновления ν_1^k пере-

пишем (4) в виде:

$$\lambda_k = \frac{1}{k} \sum_{i=1}^k \left(\frac{\prod_{j=i}^k f_{\nu_{ij}|G_1}(x_j)}{\prod_{j=i}^k f_{\nu_{ij}|G_0}(x_j)} \right)$$

Каждый обновляющий процесс $\{\nu_{ij}\}$ формируется отдельным фильтром Калмана, настроенным в соответствии с гипотезой G_1 и подключенным в i -ый момент времени.

Как известно, каждый случайный вектор ν_{ij} нормально распределен с нулевым математическим ожиданием и ковариационной матрицей Σ_i , вычисляемой фильтром Калмана. Следовательно

$$f_{\nu_{ij}|G_1}(x_j) = \frac{1}{(2\pi)^{m/2} |\Sigma_{ij}|^{1/2}} \exp \left[-\frac{1}{2} \nu_{ij}^T \Sigma_{ij}^{-1} \nu_{ij} \right],$$

где m - размерность вектора измерений x_j ; $|\Sigma_{ij}|$ - детерминант ковариационной матрицы в j -ый момент времени.

Обозначим

$$K_i = \frac{1}{(2\pi)^{m/2} |\Sigma_{ij}|^{1/2}}$$

Следовательно

$$\begin{aligned} \lambda_k &= \frac{1}{k} \sum_{i=1}^k \left(\frac{\prod_{j=i}^k K_1 \exp \left[-\frac{1}{2} \nu_{ij}^T \Sigma_{ij}^{-1} \nu_{ij} \right]}{\prod_{j=i}^k K_0 \exp \left[-\frac{1}{2} \nu_{j0}^T \Sigma_{j0}^{-1} \nu_{j0} \right]} \right) = \\ &= \frac{1}{k} \sum_{i=1}^k \prod_{j=i}^k \frac{K_1}{K_0} \exp \left[\frac{1}{2} \left(\nu_{j0}^T \Sigma_{j0}^{-1} \nu_{j0} - \nu_{j1}^T \Sigma_{j1}^{-1} \nu_{j1} \right) \right] = \\ &= \frac{1}{k} \sum_{i=1}^k \left(\frac{K_1}{K_0} \right)^{k-i} \exp \left[\sum_{j=i}^k \frac{1}{2} \left(\nu_{j0}^T \Sigma_{j0}^{-1} \nu_{j0} - \nu_{j1}^T \Sigma_{j1}^{-1} \nu_{j1} \right) \right] \end{aligned} \quad (5)$$

$\langle \text{раздел описаний} \rangle ::= \langle \text{раздел описаний } 1 \rangle$
 $\{ \langle \text{описание внутренних процедур} \rangle \}$

$\langle \text{раздел описаний } 1 \rangle ::= \{ \text{label } \langle \text{список меток} \rangle \}$
 $\{ \text{const } \langle \text{список констант} \rangle \}$
 $\{ \text{type } \langle \text{описание типов} \rangle \}$
 $\{ \text{var } \langle \text{список переменных} \rangle \}$

$\langle \text{раздел операторов} \rangle ::= \text{begin } \langle \text{операторы} \rangle \text{ end};$

Процедуры языка Паскаль являются рекурсивными конструкциями, поскольку в разделе описаний могут быть, в свою очередь, определены внутренние процедуры. Конструкции $\langle \text{заголовок процедуры} \rangle$, $\langle \text{раздел описаний} \rangle$, $\langle \text{раздел операторов} \rangle$ и другие более подробно рассматривать не будем, чтобы не нагромождать описание (для определения грамматической конструкции $\langle \text{описание процедуры} \rangle$ эти конкретизации не понадобятся). Приведенное определение для нетерминала $\langle \text{описание процедуры} \rangle$ задает язык

$\text{procedu } \Psi^*(\mu, L) \text{ d};$

где L — язык, определенный следующим образом:

$L = \{ \text{re abcd en} \mid \begin{array}{l} \text{a есть } \langle \text{идентификатор процедуры} \rangle, \\ \text{b есть } \langle \text{список формальных параметров} \rangle, \\ \text{c есть } \langle \text{раздел описаний} \rangle, \\ \text{d есть } \text{begin } \langle \text{операторы} \rangle \text{ } \end{array} \}$

$\mu = A \times B$, где

$A = \{ \text{re abc procedu} \mid \begin{array}{l} \text{a есть } \langle \text{идентификатор процедуры} \rangle, \\ \text{b есть } \langle \text{список формальных параметров} \rangle, \\ \text{c есть } \langle \text{раздел описаний } 1 \rangle \end{array} \}$

$B = \{ \text{d; begin } \langle \text{операторы} \rangle \text{ en } \}$

В этом примере, так же как и в примере 8.2 статьи [1], условия (*) могут, вообще говоря, не выполняться. Например, множество A не удовлетворяет условию $\text{Pr}(A)$, если среди его элементов встречаются такие:

$\text{a1=re p(i:real); var a:real; procedu}$

$\text{a2=re p(i:real); var a:real; procedu1:integer; procedu}$
 То есть множество A может не удовлетворять условию $\text{Pr}(A)$, если слово procedu встречается внутри конструкции $\langle \text{раздел описаний } 1 \rangle$. Обобщим приведенное замечание.

Утверждение 1. (достаточное условие префиксности множества)
 Если

$\forall u \in A : u = vx, \text{ где } x \in \Sigma^+, v \in \Sigma^* \text{ и}$

$\nexists w \in A : w = axbx, \text{ где } a, b \in \Sigma^*,$

то множество A префиксно.

Утверждение 2. (достаточное условие суффиксности множества)
 Если

$\forall u \in A : u = xv, \text{ где } x \in \Sigma^+, v \in \Sigma^* \text{ и}$

$\nexists w \in A : w = xbx, \text{ где } a, b \in \Sigma^*,$

то множество A суффиксно.

Доказательство этих утверждений опирается непосредственно на определение префиксного (суффиксного) множества.

Пример 2. Рассмотрим условный оператор языка Паскаль и покажем, что приведенная конструкция удовлетворяет условиям (*) (для множеств A, B, L выполнены соответствующие утверждения 1, 2).

$\langle \text{условный оператор} \rangle ::= \text{if } \langle \text{условие} \rangle \text{ then } \langle \text{оператор1} \rangle$
 $\text{else } \langle \text{оператор2} \rangle;$

Здесь $\langle \text{оператор1} \rangle$ — любой оператор языка Паскаль, в том числе $\langle \text{условный оператор} \rangle$ (таким образом $\langle \text{условный оператор} \rangle$ является рекурсивной конструкцией языка Паскаль). Потребуем, чтобы $\langle \text{оператор2} \rangle$ не являлся условным оператором и не содержал внутри себя условного оператора (иначе условия (*) не выполняются). Мы можем определить язык конструкции $\langle \text{условный оператор} \rangle$ таким образом:

$i\Psi^*(L, \mu);$

При этом

$L = \{ \text{f } \langle \text{условие} \rangle \text{ then } \langle \text{оператор1} \rangle$
 $\text{else } \langle \text{оператор2} \rangle \}$

В определении L считаем, что $\langle \text{оператор1} \rangle$ не является условным оператором. Множество L является префиксным, поскольку является префиксной конструкцией $f \langle \text{условие} \rangle \text{ then } \langle \text{оператор1} \rangle \text{ else}$ (по утверждению 1, else не может встречаться в левой части этой конструкции) и все операторы языка Паскаль удовлетворяют условию префиксности. Множество L суффиксно, поскольку суффиксна конструкция $\text{then } \langle \text{оператор1} \rangle \text{ else } \langle \text{оператор2} \rangle$ (по утверждению 2, else не может встречаться в правой части этой конструкции) и ни одно $\langle \text{условие} \rangle$ Паскаля не может начинаться с 'f'.

$$\mu = A \times B, \text{ где}$$

$A = \{f \langle \text{условие} \rangle \text{ then } i\}$

$B = \{\text{else } \langle \text{оператор2} \rangle\}$

При этом множество A удовлетворяет утверждению 1 (здесь $x = \text{then } i$), множество B удовлетворяет утверждению 2 (здесь $x = \text{else}$) и выполнены условия (*).

Литература

[1] О. А. Дубасова, Б. Ф. Мельников: *Об одном расширении класса контекстно-свободных языков*. Программирование (РАН), 6 (1995), стр. 46-58.

[2] А. И. Марченко, Л. А. Марченко: *Программирование в среде Turbo Pascal*. Москва, Бином Универсал, 1997.

Б. Ф. Мельников.

Некоторые алгебраические задачи в теории формальных языков

§ 1. Введение

Настоящая статья является дополнением значительно большей по объему статьи [1]. Здесь приведены доказательства нескольких утверждений, которые были опущены в той работе.

В данной статье рассматриваются некоторые проблемы равенства в глобальных надмоноидах свободных моноидов.¹ Эти объекты также являются моноидами, см. [2]. Чаще всего свободные моноиды и их глобальные надмоноиды встречаются в теории формальных языков, которую иногда даже считают одним из приложений алгебры полугрупп. По терминологии теории формальных языков, глобальные надмоноиды свободных моноидов являются объектами, состоящими из множеств слов (языков) с операцией конкатенации, а единицей глобального надмоноида является язык $\{e\}$, состоящий из единственного слова – пустого.

Важно отметить, что всюду в настоящей статье будет допускаться случай, когда моноиды имеют бесконечное число образующих (атомов). То есть – по терминологии теории формальных языков – будут рассматриваться языки, заданные над произвольным алфавитом – возможно, бесконечным.

В глобальных надмоноидах рассматриваются некоторые проблемы равенства (задачи поиска необходимых и достаточных условий специальных равенств). Среди этих проблем – исследование равенства $A^m = B^n$. Доказывается, например, что в случае префиксных множеств это равенство влечет наличие у множеств A и B общего корня, определяемого обычным образом.

¹ Определения всех алгебраических понятий, употребляемых здесь и далее, можно найти, например, в [2]. Далее обычно будем говорить просто «глобальные надмоноиды». По этому поводу отметим заранее, что рассматриваемые далее проблемы равенства выполняются в глобальных надмоноидах не только свободных моноидов, но и более общих объектов – моноидов с левым законом сокращения. Однако для того, чтобы строго это доказать, или чтобы и для последних объектов были верными приведенные далее доказательства, необходимо было бы и для них строго определить понятия, соответствующие понятиям длины, префиксности и т.п., а также доказать некоторые свойства этих понятий.

Критерии рассматриваемых равенств могут быть использованы в различных задачах теории формальных языков. Например, в [3,4] и обобщающей эти работы статье [1] эти факты и им подобные были использованы при описании т.н. «простых скобочных» подклассов класса контекстно-свободных языков с разрешимой проблемой эквивалентности, а в [5] – при доказательстве необходимых условий равенства двух КС-языков специального вида – «скобочных» языков. По-видимому, доказанные ниже и в [1] критерии могут быть применены для исследования проблемы эквивалентности и в других подклассах класса КС-языков.

В связи с двумя упомянутыми выше фактами:

- о бесконечности рассматриваемых нами алфавитов;
- о применении доказанных далее утверждений в некоторых задачах теории контекстно-свободных языков,

– отметим следующее обстоятельство. Контекстно-свободные языки определяются над конечным алфавитом, причем для каждого нетерминала имеется конечное множество правил (см. [6,7] и мн. др.). Однако для исследования многих конкретных КС-языков часто оказывается удобным рассматривать и специальные расширения класса контекстно-свободных языков, – например, допускать языки над бесконечным алфавитом, или рассматривать грамматики с бесконечным числом правил (со специальными ограничениями на множества правых частей правил каждого конкретного нетерминала). По этому поводу см. также [1].

Теперь – о применяемых в работе обозначениях. «Главным» алфавитом в дальнейшем является Σ , и, как уже было отмечено, число букв в нем не обязательно конечно. Все рассматриваемые слова и языки задаются, если не сказано иного, над Σ . При этом для упрощения формулировок утверждений все языки (синонимы: элементы глобального надмоноида, множества слов), обозначаемые заглавными латинскими буквами, предполагаются непустыми и, если не сказано иного, отличными от нуля глобального надмоноида $\emptyset$. Иногда будут рассматриваться и «вспомогательные» алфавиты, также, возможно, бесконечные, обозначаемые Δ , обычно с нижними индексами.

Для слов и языков над всеми алфавитами будут использоваться стандартные обозначения алгебры и теории формальных языков – см. [2,6,7] и др. Кроме них, будут применяться и нестандартные обозначения, введенные в [1,8], например, $\|A\|$ с различными нижними

индексами, а также $\text{pr}(\Delta)$. Отметим, что в обеих этих статьях поясняется, среди прочего, смысл нижних индексов у алфавитов Δ . Из указанных обозначений наиболее важным для настоящей статьи является $\text{Pr}(A)$ (условие префиксности языка A), оно равносильно следующему: $(\forall u, v \in A)(u \notin \text{pref}(v))$.

§ 2. Поиск левого делителя в глобальном надмоноиде

Доказательства следующих трех утверждений (1–3) не сложны. Будем обычно пользоваться ими без специальных ссылок.

Утверждение 1. Если $Pr(A)$ и $AB = AC$, то $B = C$. $\square$

Это – левый закон сокращения, но не для слов, а для языков, обладающих префиксным свойством.

Утверждение 2. Если $Pr(A)$ и $Pr(B)$, то $Pr(AB)$. $\square$

Утверждение 3. Если $Pr(AB)$, то $Pr(B)$. $\square$

Как было сказано во введении, рассматриваемые далее факты были сформулированы в [1]. Однако в связи с ограничением на объем той работы, а также в связи с прикладным характером задач, рассматриваемых журналом «Программирование», их доказательства были опущены. Отметим еще, что там же, в [1], приведены поясняющие примеры конкретных языков.

Утверждение 4. Пусть $Pr(A)$, $Pr(B)$, и $AX = BY$ для каких-либо X и B . Тогда $(\exists C) (A = BC \text{ или } B = AC)$.²

Доказательство. Если $A = B$, то сформулированное утверждение выполняется для $C = \{e\}$. Поэтому будем полагать далее, что $A \neq B$.

Предположим, что $u \in A$ и

$$uv \in B \quad (1)$$

для некоторых слов u и $v \neq e$ (случай $u \in B$, $uv \in A$ рассматривается аналогично). Тогда согласно $Pr(A)$ и $AX = BY$ выполнено условие $vY \subseteq X$. Поэтому для произвольного $v' \in A$ имеем следующее: $(\forall y \in Y) (u'vy \in AX = BY)$, и вследствие условия $Pr(B)$ получаем $u'v \in B$.

Таким образом, для рассматриваемого v выполняется утверждение $(\forall u' \in A) (u'v \in B)$. Следовательно, для множества C , состоящего из всех слов v , которые могут быть выбраны согласно (1), выполнено равенство $AC = B$. $\square$

² Вместо префиксности B можно потребовать суффиксность Y . Доказательство немного изменится, см. [1].

Заметим, что оба случая одновременно – и $(\exists u, v \neq e) (u \in A, uv \in B)$, и $(\exists u, v \neq e) (u \in B, uv \in A)$ ³, – не могут иметь места из-за $Pr(A)$.

Утверждение 5. Пусть $Pr(A)$, $Pr(L)$, и $A^m L = LB^n$ для некоторых $m, n \in \mathbb{N}$.⁴ Тогда $(\exists \tilde{A}) (AL = L\tilde{A})$, причем $Pr(\tilde{A})$ и $\tilde{A}^m = B^n$.

Доказательство. Если $A = \{e\}$, то вследствие условия $L = LB^n$ выполнено $B = \{e\}$, и доказываемое утверждение очевидно. Поэтому будем далее полагать, что $A \neq \{e\}$.

Для доказательства построим конечную последовательность множеств L_i , где $i \in \overline{1, r}$ для некоторого натурального r , следующим образом. Пусть $i = 0$ и $L_0 = L$. По условию, выполнено $Pr(L_i)$, кроме того $A^m L_i = L_i B^n$ для выбранного нами $i = 0$ и заданных m, n, A и B .

Теперь предположим, что для некоторого i выполнены условия $Pr(L_i)$ и $A^m L_i = L_i B^n$. Обозначив $X = A^{m-1} L_i$ и $Y = B^n$, получаем, что согласно утверждению 4, для некоторого языка C выполнено или

$$A = L_i C, \quad (2)$$

или

$$L_i = AC. \quad (3)$$

Если выполняется первое условие – (2), то $(L_i C)^m L_i = L_i B^n$, т.е. $L_i (CL_i)^m = L_i B^n$, и обозначив $\tilde{A} = CL_i$, получаем $\tilde{A}^m = B^n$. Вследствие $Pr(L_i)$ выполнено условие $Pr(\tilde{A})$ согласно (2).

Если же выполняется (3), то обозначив $L_{i+1} = C$, мы можем записать заданное условие в виде $A^m A L_{i+1} = A L_{i+1} B^n$, поэтому $A^m L_{i+1} = L_{i+1} B^n$. Таким образом, мы смогли построить множество (L_{i+1}) для нового индекса $i + 1$. Заметим, что $\|L_{i+1}\|_{\min} < \|L_i\|_{\min}$ (т.к. $L_i = A L_{i+1}$ и $A \neq \{e\}$), поэтому построенная последовательность множеств – конечная, и для некоторого натурального i выполнено условие (2).

Покажем, что $AL = L\tilde{A}$. Пусть i – такое число, что выполнено условие (2). На основе метода построения L_i заключаем,

³ По доказанному выше, эти условия эквивалентны равенствам $A = BC$ и $B = AC'$ соответственно, причем $C \neq \{e\}$ и $C' \neq \{e\}$.

⁴ В [1] было наложено дополнительное ограничение – суффиксность B . Аналогично для дальнейших утверждений: в соответствующих им утверждениях работы [1] были наложены более жесткие ограничения, достаточные, однако, для решаемых в [1] задач.

что выполнено равенство $L = A^i L_i$. Из (2) и $\tilde{A} = CL_i$ следует $AL_i = L_i \tilde{A}$. Поэтому $A^{i+1} L_i = A^i L_i \tilde{A}$, а из последнего получаем $AL = L \tilde{A}$. $\square$

Утверждение 6. Пусть $Pr(A)$ и для некоторых B, X и Y выполнены равенства $AX = XB$ и $AY = YB$. Пусть также $\|X\|_{\min} \leq \|Y\|_{\min}$. Тогда $(\exists Z)(Y = XZ)$.

Доказательство. Если $A = \{e\}$ (что равносильно $B = \{e\}$), то выполнение доказываемого условия очевидно. Поэтому будем далее предполагать, что $A \neq \{e\}$ и $B \neq \{e\}$.

Будем в этом доказательстве использовать такое сокращение: $\|Q\| = \|Q\|_{\min}$ для произвольного Q .

Построим конечную последовательность пар множеств (X_i, Y_i) , такую что при каждом i для X_i вместо X и Y_i вместо Y выполнены условия доказываемого утверждения. Базис этого рекуррентного процесса построения пар множеств следующий: $X_0 = X$, $Y_0 = Y$. Процесс финиширует, если

$$\|X_i\| < \|A\|. \quad (4)$$

Опишем шаг процесса.

По предположению индукции, $\|X_i\| \geq \|A\|$, поэтому согласно утверждению 4, $X_i = AC$ для некоторого C . $\|Y_i\| \geq \|X_i\|$, тогда $\|Y_i\| \geq \|A\|$, поэтому $Y_i = AC'$ для некоторого языка C' . Обозначив $X_{i+1} = C$ и $Y_{i+1} = C'$, получаем, что согласно $AX_i = X_i B$ и $AY_i = Y_i B$, выполнены условия $AA X_{i+1} = AX_{i+1} B$ и $AA Y_{i+1} = AY_{i+1} B$, т.е. $AX_{i+1} = X_{i+1} B$ и $AY_{i+1} = Y_{i+1} B$. Таким образом, мы построили новую пару множеств. Заметим еще, что $X = A^i X_i$ и $Y = A^i Y_i$ для произвольного i .

Описанный процесс не является бесконечным, т.к. согласно $A \neq \{e\}$, выполнены условия $\|X_{i+1}\| < \|X_i\|$ и $\|Y_{i+1}\| < \|Y_i\|$. Таким образом, мы можем рассмотреть число i , для которого верно неравенство (4). Согласно утверждению 4, выполнено условие $A = X_i D$ для некоторого множества D . Возможны два варианта:

- $\|Y_i\| < \|A\|$, поэтому $A = Y_i D'$ для некоторого D' , тогда $Y_i D' = X_i D$, и аналогично утверждению 4 доказываем, что $Y_i = X_i Z$ для некоторого Z .
- $\|Y_i\| \geq \|A\|$, поэтому согласно утверждению 4, $Y_i = AD''$ для некоторого D'' , а отсюда $Y_i = X_i DD''$.

Итак, в обоих случаях выполнено равенство $Y_i = X_i Z$ для некоторого языка Z . Поэтому $A^i Y_i = A^i X_i Z$, и, согласно ранее отмеченному, $Y = XZ$. $\square$

Утверждение 7. Пусть $AL = L \tilde{A}$, $B = h_A(C)$ для некоторого $C \in \text{tr}(\Delta)$ и $BL = L \tilde{B}$ для некоторого $\tilde{B}$. Тогда $\tilde{B} = h_{\tilde{A}}(C)$.

Перед доказательством приведем две группы примеров, описывающих возможные множества C :

- L состоит ровно из одного слова, тогда C – произвольный язык из множества $\text{tr}(\Delta)$;
- $C = \Delta^n$ для некоторого $n \in \mathbb{N}$.

В обоих случаях, конечно, должны выполняться и остальные сформулированные условия.

Других примеров автору не известно. И, по-видимому, верно и более сильное утверждение, требующее, кроме $\tilde{B} = h_{\tilde{A}}(C)$, еще и выполнения одного из приведенных в примерах условий. Отметим еще, что в [1] этого более сильного утверждения (если оно и верно) не понадобилось: равенство $C = \Delta^n$ там было получено на основе других соображений.

Доказательство утверждения 7. Будем в этом доказательстве использовать такое сокращение: $\|Q\| = \|Q\|_{\max}$ для произвольного Q .

Докажем требуемый факт индукцией по $\|C\|$. Базис очевиден, т.к. существует единственный язык C из множества $\text{tr}(\Delta)$, для которого $\|C\| = 1$: это – $C = \Delta$, а из последнего получаем $B = A$ и $\tilde{B} = \tilde{A}$. Докажем шаг индукции.

Пусть $C = C_1 \cup C_2 \Delta$, где $\|C_2\|_{\vee} = \|C\| - 1$ и $\|C_1\| < \|C\|$. Заметим, что $C' = C_1 \cup C_2 \in \text{tr}(\Delta)$, причем такое представление C в виде объединения множеств C_1 и $C_2 \Delta$ всегда единственно. Обозначим $B_1 = h_A(C_1)$, $B_2 = h_A(C_2)$, $B' = h_A(C')$. Тогда согласно заданному условию, $BL = L \tilde{B}$, из чего следуют равенства $B_1 L = L \tilde{B}_1$ для $\tilde{B}_1 = h_{\tilde{A}}(C_1)$, а также $h_A(C_2 \Delta) L = L X$ для некоторого X . Согласно последнему равенству, $h_A(C_2) L = L \tilde{B}_2$ для $\tilde{B}_2 = h_{\tilde{A}}(C_2)$, т.е. $B_2 L = L \tilde{B}_2$. Поэтому

$$B' = h_A(C') = h_A(C_1 \cup C_2) =$$

$$h_A(C_1) \cup h_A(C_2) = L \tilde{B}_1 \cup L \tilde{B}_2 = L \tilde{B}'$$

для некоторого $\tilde{B}'$. Согласно предположению индукции, $\tilde{B}' = h_{\tilde{A}}(C')$. Тогда

$$Lh_{\tilde{A}}(C) = Lh_{\tilde{A}}(C_1) \cup Lh_{\tilde{A}}(C_2\Delta) = L\tilde{B}_1 \cup Lh_{\tilde{A}}(C_2)h_{\tilde{A}}(\Delta) = \\ L\tilde{B}_1 \cup L\tilde{B}_2\tilde{A} = B_1L \cup B_2AL = (B_1 \cup B_2A)L = BL.$$

Таким образом, $Lh_{\tilde{A}}(C) = BL$, и вследствие $BL = L\tilde{B}$ выполнено равенство $\tilde{B} = h_{\tilde{A}}(C)$, доказывающее шаг индукции. $\square$

§ 3. Об общем корне в глобальном надмоноиде

Утверждение 8. Пусть $Pr(A)$ и для некоторых $m, n \in \mathbb{N}_0$

$$A^m = B^n. \quad (5)$$

Тогда $(\exists C; m', n')(A = C^{n'}, B = C^{m'})$.

Доказательство. Без ограничения общности можно предполагать, что $m \leq n$. Тогда согласно утверждению 4, $A = BQ$ для некоторого множества Q (допускаем возможность $Q = \{e\}$). Выберем максимальное число p_1 , для которого $(\exists Q)(A = B^{p_1}Q)$. Обозначим $A_1 = Q$ (это множество однозначно определяется согласно представлению A в виде $A = B^{p_1}A_1$); тогда мы можем записать (5) таким образом:

$$(B^{p_1}A_1)^m = B^n. \quad (6)$$

Из последнего,

$$A_1(B^{p_1}A_1)^{m-1} = B^{n-p_1} \quad (7)$$

(при этом очевидно, что $n \geq p_1$).

Согласно (7) и утверждению 4, $B = A_1Q$ для некоторого Q (т.к. равенство $A_1 = B^{p_1}Q$ для произвольного $Q \neq \{e\}$ невозможно из-за способа выбора p_1). Выберем аналогично число q_1 и введем множество B_1 согласно равенству $B = A_1^{q_1}B_1$ (число q_1 также максимально возможное). На основе введенных обозначений мы можем записать (5) таким образом:

$$((A_1^{q_1}B_1)^{p_1}A_1)^m = (A_1^{q_1}B_1)^n. \quad (8)$$

Продолжая этот процесс аналогично проделанному в доказательствах утверждений 5 и 6, определяем конечные последовательности пар множеств A_i и B_i . При этом числа p_i, q_i и языки A_i, B_i определяются из условий $A_{i-1} = B_{i-1}^{p_i}A_i$ и $B_{i-1} = A_{i-1}^{q_i}B_i$. Таким образом, после этих переобозначений мы можем переписать исходное равенство (5) следующими способами:

$$((\dots(((B_{i-1})^{p_{i-1}}A_i)^{q_{i-1}}B_{i-1})^{p_{i-2}}\dots B_{i-1})^{p_1}A_i)^m = \\ = (\dots(((B_{i-1})^{p_{i-1}}A_i)^{q_{i-1}}B_{i-1})^{p_{i-2}}\dots B_{i-1})^n, \quad (9)$$

$$((\dots(((A_i)^{q_{i-1}}B_i)^{p_{i-1}}A_i)^{q_{i-2}}\dots B_i)^{p_1}A_i)^m = \\ = (\dots(((A_i)^{q_{i-1}}B_i)^{p_{i-1}}A_i)^{q_{i-2}}\dots B_i)^n \quad (10)$$

соответственно. Заметим, что равенства (9) и (10) выполняются и на первом шаге (т.е. для множеств A_1 и B_1), если считать $A_0 = A$ и $B_0 = B$; эти равенства на первом шаге суть условия (6) и (8). Возможность выбора p_{i-1} и A_i для каждого следующего i — следствие (10), а возможность выбора q_i и B_i — следствие (9).

Описанный процесс конечен, т.к. для произвольного рассматриваемого i выполнены условия $\|A_i\|_{\min} < \|A_{i-1}\|_{\min}$ и $\|B_i\|_{\min} < \|B_{i-1}\|_{\min}$. Поэтому для некоторого $r \in \mathbb{N}$ либо $p_r = 0$, либо $q_r = 0$. Пусть $p_r = 0$ (случай $q_r = 0$ рассматривается аналогично). Тогда $A_r = B_r$ (поскольку неравенство $A_r \neq B_r$ противоречит тому, что число q_{r-1} в (10) — максимально возможное). Обозначив $C = A_r = B_r$, мы легко доказываем равенства $A = C^{n'}$ и $B = C^{m'}$ для некоторых $n', m' \in \mathbb{N}$ индукцией по i для $i = r, r-1, \dots, 1, 0$, используя при этом (9) и (10). $\square$

Замечание. Если в доказанном утверждении обозначить через k наибольший общий делитель чисел m и n , то очевидно, что $n' = n/k$ и $m' = m/k$. Вообще, можно провести аналогию между утверждением 8 — с одной стороны, и с другой стороны, легко доказываемым фактом из области арифметики, тоже формулирующимся с помощью (5), где, в отличие от доказанного утверждения, A, B и C — некоторые натуральные числа, причем C является общим корнем из чисел A и B (корнем разных степеней).

1. О. А. Дубасова, Б. Ф. Мельников: Об одном расширении класса контекстно-свободных языков. – *Программирование* (принято к публикации, 1995).
2. *Общая алгебра. Том 2*, М., Наука, 1991.
3. B. Melnikov: Semigroups and equivalence problem in subclasses of the CF-languages class. *Moscow 1993, NTAMCS, Intern. Conf. Abstracts*, 66–67.
4. Б. Ф. Мельников: Некоторые проблемы равенства в глобальном надмоноиде свободного моноида. – В кн.: *Фундаментальные проблемы математики и механики (Программа «Университеты России»)*, М., Изд-во Моск. ун-та, 1994.
5. Б. Ф. Мельников: Некоторые следствия условия эквивалентности однозначных скобочных грамматик, *Вестник Моск. ун-та, сер. Вычисл. матем. и киб-ка*, 3 (1991) 51–53.
6. А. Ахо, Дж. Ульман: *Теория синтаксического анализа, перевода и компиляции. Том 1*, М., Мир, 1978.
7. А. Саломеа: *Жемчужины теории формальных языков*, М., Мир, 1986.
8. Б. Ф. Мельников: Алгоритм проверки равенства бесконечных итераций конечных языков. – *Вестник Моск. ун-та, сер. Вычисл. матем. и киб-ка*, 4 (1996) 49–54.

Б. Ф. Мельников, А. А. Вахитова.

Звездная высота конечного автомата

§ 1. Основные определения

В настоящей статье предлагается способ сведения проблемы звездной высоты (см., например, [1]; впервые эта проблема решена в 1987 г., см. [2]) к задаче построения специального конечного автомата.

Используются стандартные обозначения ([1] и др.), а также терминология и обозначения из [3, 4]. Часть материала данной статьи была опубликована в [4]. В настоящей работе улучшены некоторые доказательства и даны новые примеры; а приведенный в § 3 материал не имеет аналогий с материалом [4].

Будем рассматривать (недетерминированный) конечный автомат $K = (Q, \Sigma, \delta, S, F)$, допускаем возможности $|S| \geq 1$ и $|F| \geq 1$. Пусть $|Q| = n$ (состояний); тогда существуют $n!$ различных инъективных функций ω вида $\omega : Q \rightarrow \mathbf{R}^+$. Для каждой из возможных функций ω , вообще говоря, различны определенные согласно [3] регулярные выражения, задающие язык автомата K .

Рассмотрим следующее определение.

Определение 1. Звездная высота конечного автомата $K = (Q, \Sigma, \delta, S, F)$ есть

$$sh(K) = \min_{\omega} sh(\mathcal{R}_{K\omega}).$$

Заметим, что если автомат K допускает язык L , то для любой инъективной функции $\omega : Q \rightarrow \mathbf{R}^+$ выполнено неравенство

$$sh(\mathcal{R}_{K\omega}) \geq sh(L). \quad (1)$$

Рассмотрим примеры. Граф переходов первого автомата, допускающего язык $(ab^*c)^*$, изображен на рис. 1.

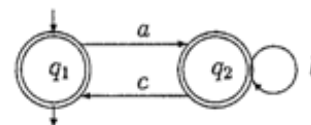


Рис. 1

Для того, чтобы посчитать звездную высоту этого автомата, нужно рассмотреть 2 функции ω , назовем их ω_1 и ω_2 . Во-первых, рассмотрим функцию $\{\omega_1(q_1) = 1, \omega_1(q_2) = 2\}$. Для нее согласно доказанному в [3] получаем $R_{q_2} = b^*$, $R_{q_1} = (ab^*c)^*$ и

$$\mathcal{R}_{K\omega_1} = (ab^*c)^* + ab^*c \quad (2)$$

(некоторые символы ϕ опущены, других эквивалентных преобразований не делалось). Аналогично, рассматривая функцию $\{\omega_2(q_1) = 2, \omega_2(q_2) = 1\}$, получаем для нее (также согласно [3]) следующее: $R_{q_1} = \phi^*$, $R_{q_2} = (ca + b)^*$ и

$$\mathcal{R}_{K\omega_2} = \phi^* + a(ca + b)^*c. \quad (3)$$

Посчитав звездную высоту (2) и (3) (звездную высоту регулярных выражений – см., например, [1]), получаем, что звездная высота рассматриваемого автомата равна 1.

Заметим, что автомат на рис. 1 можно считать каноническим. (Если опускать бесполезное состояние. Так поступать можно, т.к. согласно приведенным выше определениям, от бесполезного состояния звездная высота автомата не зависит.) Поэтому очевидно, что его звездная высота равна звездной высоте допускаемого им языка. Однако существуют канонические автоматы, чьи звездные высоты превосходят звездные высоты их языков. Например см. рис. 2 (снова исследуем язык, задаваемый регулярным выражением $(a + ab + ba)^*$), рассмотрим этот пример кратко.

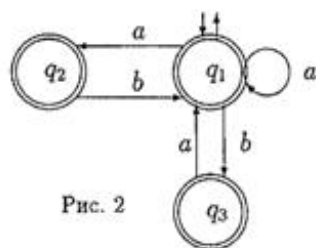


Рис. 2

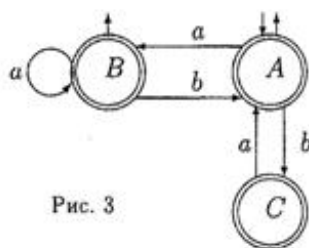


Рис. 3

Звездная высота автомата равна 1, однако легко посчитать, что звездная высота эквивалентного канонического автомата равна 2 (см. рис. 3). Более того, звездная высота канонического автомата, допускающего «похожий» язык $(a + ab + ca)^*$, равна 3.

Согласно приведенному в [3] и данным выше определениям, выполнено следующее утверждение.

Утверждение 1. Звездная высота любого регулярного выражения R_q , определяемого согласно [3]¹, может быть посчитана следующим образом:

- если $\Omega(q) \neq \emptyset$, то полагаем $sh_q^1 = \max_{q' \in \Omega(q)} sh(R_{q'}) + 1$, иначе считаем $sh_q^1 = 0$;
- если существует $a \in \Sigma$, для которого $\delta(q, a) \neq \emptyset$, то полагаем $sh_q^2 = 1$, иначе считаем $sh_q^2 = 0$;

тогда $sh(R_q) = \max(sh_q^1, sh_q^2)$. $\square$

Не ограничивая общности, мы можем рассматривать только такие автоматы, которые не содержат бесполезных состояний (это было отмечено ранее), а также недостижимых (поскольку согласно данным выше определениям и материалу статьи [3], от этих состояний звездная высота автомата не зависит). Согласно определению $\Omega(q)$, для таких автоматов выполняется следующее утверждение.

Утверждение 2. $sh(\mathcal{R}_{K\omega}) = \max_{q \in Q} sh(R_q)$. $\square$

Заметим, что способ подсчета звездной высоты автомата, использующий утверждение 2, значительно проще, чем приведенный ранее – использующий $n!$ биактивных функций. В настоящее время авторы располагают подробными описаниями алгоритмов для обоих способов подсчета; эти алгоритмы, а также оценки их эффективности, готовятся к опубликованию.

§ 2. Постановка задачи для автоматов

По заданному регулярному выражению можно построить такие конечный автомат $K = (Q, \Sigma, \delta, \{s\}, \{f\})$ и инъективную функцию $\omega : Q \rightarrow \mathbf{R}^+$, что построенное согласно [3] регулярное выражение $\mathcal{R}_{K\omega}$ (эквивалентное заданному, но, вообще говоря, не совпадающее с ним) будет иметь звездную высоту, в точности равную звездной высоте заданного выражения. При этом нестрогое неравенство (1) превращается в равенство.

Все автоматы, которые мы будем рассматривать далее, будут иметь ровно 1 стартовое состояние и ровно 1 финальное. Эти

¹ Ниже, при рассмотрении разных автоматов с одноименными состояниями, будем обозначать звездную высоту этого выражения записью $sh_{K\omega}(q)$.

состояния будут обозначены s и f с индексами, соответствующими их регулярным выражениям; так же будут проиндексированы и функции ω . Отметим, что $\omega(s)$ всегда будет минимальным значением функции ω .

Теорема 3. Для произвольного регулярного выражения r существуют эквивалентный конечный автомат $K_r = (Q_r, \Sigma, \delta_r, \{s_r\}, \{f_r\})$ и функция $\omega_r: Q_r \rightarrow \mathbb{R}^+$, такие что $sh(r) = sh(\mathcal{R}_{K_r \omega_r})$.

Доказательство. Согласно определению регулярного выражения и его звездной высоты, достаточно доказать утверждение, соответствующее формулировке теоремы, для автоматов $K_\emptyset$, K_e и K_a (база индукции), а также – если утверждение доказано для двух регулярных выражений p и t – доказать его для регулярных выражений $(p+t)$, (pt) и (p^*) .

Автоматы $K_\emptyset$, K_a и K_e см. на рис. 4–6.

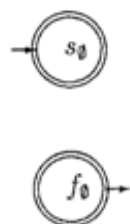


Рис. 4

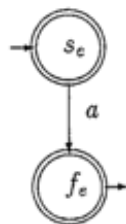


Рис. 5

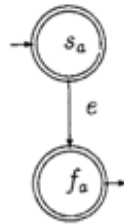


Рис. 6

При этом для каждого случая можно считать

$$\omega(s) = 0, \quad \omega(f) = 1.$$

Отметим, что пару (K_e, ω_e) нужно указать непосредственно, использовать описанный далее индуктивный шаг «звезда» – применяя его к паре $(K_\emptyset, \omega_\emptyset)$ – нельзя: согласно [1], $sh(e) = 0$.

Построение автомата K_{p+t} по K_p и K_t см. на рис. 7.

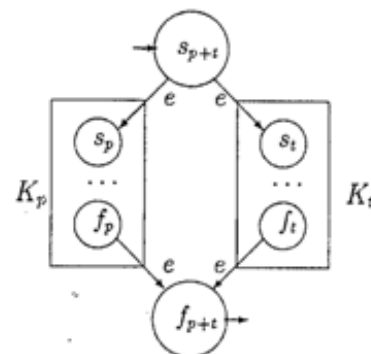


Рис. 7

При этом функция ω_{p+t} определяется следующим образом:

- $\omega_{p+t}(s_{p+t})$ – некоторое положительное число, меньшее m_p ;
- для каждого q из K_p , $\omega_{p+t}(q) = \omega_p(q)$;
- для каждого q из K_t , $\omega_{p+t}(q) = M_p + \omega_t(q)$.

Корректность описанного построения пары – следствие утверждений 1 и 2 и определения множеств $\Omega(q)$ (поскольку в графе переходов K_{p+t} по построению не существует путей из какой-либо вершины множества K_t в какую-либо вершину множества K_p).

Аналогично для пары (K_{pt}, ω_{pt}) ; граф переходов автомата изображен на рис. 8.

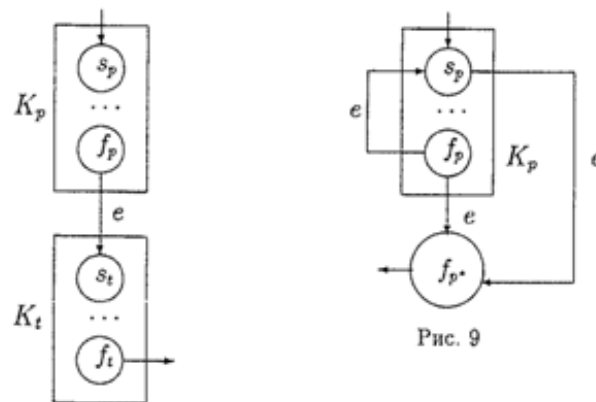


Рис. 8

Построение автомата K_{p^*} по K_p см. на рис. 9. Для q из K_p полагаем $\omega_{p^*}(q) = \omega_p(q)$, и, например, $\omega_{p^*}(f_{p^*}) = M_p + 1$. По

построению K_p , все пути, имеющиеся в графе переходов K_p и отсутствовавшие в графе переходов K_p , должны содержать s_p . Таким образом, в автомате K_p для произвольного $q \neq s_p$, множества $\Omega_2(q, q')$ (см. об этой функции в [3]) совпадают с одноименными множествами в K_p при любых q' и q'' , и только для вершины s_p мы получаем новые пути в множествах $\Omega_{s_p}(s_p, q)$ и $\Omega_{s_p}(q, s_p)$. Поэтому согласно утверждению 1, $sh(R_{s_p})$ (в автомате K_p) увеличивается на 1 по сравнению с $sh(R_{s_p})$ (посчитанной для автомата K_p), и согласно утверждению 2, $sh(\mathcal{R}_{K_p, \omega_p})$ также увеличивается на 1 по сравнению с $sh(\mathcal{R}_{K_p, \omega_p})$. По определению звездной высоты регулярного выражения получаем, что пара (K_p, ω_p) удовлетворяет равенству $sh(p^*) = sh(\mathcal{R}_{K_p, \omega_p})$. $\square$

Итак, для произвольного регулярного выражения – в том числе для имеющего среди всех ему эквивалентных минимальную звездную высоту – мы построили конечный автомат, имеющий такую же звездную высоту. Таким образом, проблему звездной высоты можно переформулировать следующим образом: нужно построить конечный автомат, имеющий среди всех ему эквивалентных минимальную звездную высоту, – и когда мы построим один из таких оптимальных автоматов, то перебрав все $n!$ биекций $\omega: Q \rightarrow \bar{1}, n$ (где n – количество его состояний), мы получим среди образующихся регулярных выражений $\mathcal{R}_{K, \omega}$ искомое, т.е. имеющее минимально возможную звездную высоту.

§ 3. Изменение звездной высоты при объединении состояний

В данном разделе объединение состояний автомата понимается согласно [3].

Утверждение 4. Пусть задан автомат

$$K = (Q, \Sigma, \delta, S, F),$$

а состояние $q \in Q$ такое, что

$$\omega(q) = \min_{r \in Q} \omega(r).$$

Пусть также для некоторого $r \in Q$ выполнено равенство $\varphi_K^{in}(r) = \varphi_K^{in}(q)$. Тогда для автомата

$$K_1 = \mathcal{I}^q(K)$$

выполнено следующее: $sh(K_1) = sh(K)$.²

Доказательство непосредственно следует из описания алгоритма объединения состояний автомата (см. [3, разд. 5]) и утверждения 2 настоящей статьи. $\square$

Заметим, что согласно приведенному в § 2 способу построения автомата, эквивалентного заданному регулярному выражению с такой же, как у последнего, звездной высотой, мы всегда можем считать, что функция ω имеет минимум в единственном стартовом состоянии автомата. Поэтому мы при необходимости можем считать, что в автомате (определяющем регулярное выражение, для которого мы исследуем звездную высоту) нет состояний (кроме стартового s), имеющих такое же, как у s , значение функции φ^{in} .

Если же рассматривать зеркальные автоматы (также см. [3]), имеющие, как несложно убедиться, ту же самую звездную высоту, то следствие, сформулированное в предыдущем абзаце, можно аналогично переформулировать для единственного финального состояния f и функции φ^{out} .

Литература

1. А. Саломая: *Жемчужины теории формальных языков*, М., Мир, 1986.
2. K. Hashiguchi: Algorithms for determining relative star height and star height, *Inform. Comput.*, 78 (1987) 124–169.
3. Б. Ф. Мельников: Еще раз о конечных автоматах. – В кн.: *Фундаментальные проблемы математики и механики*, Изд-во Ульяновского ун-та, 1996, 148–158.
4. Б. Ф. Мельников: *Подклассы класса контекстно-свободных языков*, М., Изд-во Моск. ун-та, 1995.

² Равенство $\mathcal{L}(K_1) = \mathcal{L}(K)$ – следствие утверждения 5.1 статьи [3].

Б. Ф. Мельников, И. В. Муллина.

Об одном «быстром» алгоритме построения регулярного выражения по заданному НРС-автомату

Потребность в формальном грамматическом описании различных языков возникает во многих областях науки, и поэтому теория таких языков становится междисциплинарным предметом исследования. Рассмотрение некоторых весьма простых и естественных устройств для задания языков показывает, что кажущиеся совершенно различные классы устройств (формализмы) эквивалентны в том смысле, что задают один и тот же класс языков. Это дает возможность при решении различных задач выбирать более эффективный метод задания языка. При этом в определенных случаях для изучения свойств одного и того же языка необходимо одновременно иметь его задание в виде различных классов устройств. Так, для регулярных языков давно доказано, что их можно задавать при помощи одного из следующих «канонических» формализмов:

- конечного автомата;
- детерминированного конечного автомата;
- праволинейной грамматики;
- регулярного выражения.

(см. [1] и мн. др.). Однако возможны и другие формализмы, «неканонические». Например, на основании доказанного в [2,3] к этим формализмам можно добавить т. н.

- $\pi(1)$ -подкласс класса последовательностных языков.

В данной заметке предлагается оригинальный алгоритм построения регулярного выражения (РВ), эквивалентного заданному недетерминированному конечному автомату Рабина-Скотта (КА). При построении данного алгоритма использовался *улучшенный вариант* идеи построения РВ, эквивалентного КА. Эта идея взята из различных вариантов доказательства теоремы Клини, являющейся теоретическим обоснованием существования РВ, эквивалентного КА (см. [1-3]).

Описываемый ниже алгоритм был реализован и опробован на многих известных примерах. В настоящее время готовится к публикации статья с подробным доказательством его корректности (схема доказательства приведена в настоящей заметке) и оценкой его эффективности.

Некоторые определения

1. Состояние (вершину) q будем называть *достижимым из состояния p* , если в графе переходов рассматриваемого КА имеется путь из p в q . См. аналогичные определения в книгах по теории графов [4,5].

2. Говоря, что язык задается последовательностью значений функции переходов, будем подразумевать задание языка при помощи КА, значения функции переходов которого описаны этой последовательностью.

3. Будем говорить, что состояние (вершина) $q \in Q$ имеет n *входов*, если существует ровно n значений функции переходов вида $\delta(p, a) = q$, где $a \in \Sigma$, $p \in Q$ (т.е., говоря на языке теории графов, в эту вершину входят ровно n ребер).

4. Будем говорить, что состояние (вершина) $q \in Q$ имеет n *выходов*, если существует ровно n значений функции переходов вида $\delta(q, a) = p$, где $a \in \Sigma$, $p \in Q$ (т.е. из этой вершины выходят ровно n ребер).

Замечания

1. Ниже немного отступим от классической теории, и вторым параметром в функции переходов δ у нас будет не символ алфавита Σ , а некоторое РВ. Возможность такого обобщения очевидна, и подавляющее большинство утверждений теории регулярных языков при этом остаются истинными (доказательства соспадают с «каноническими»). Первоначально

$$\forall a \in \Sigma, q, p \in Q \text{ полагаем } \delta(q, a) = p$$

(т.е. просто «приписываем» дуге РВ a вместо «приписанной ранее» буквы a).

2. Если автомат имеет несколько выходных состояний, то перед работой алгоритма добавляем новое состояние, которое будет единственным выходным; все исходные выходные соединяются с ним пустыми переходами.

3. Аналогичное упрощение возможно и для входных состояний, однако при этом, по-видимому, получаемый алгоритм не упрощается.

4. Приводимый ниже алгоритм будет формулироваться, в отличие от большинства алгоритмов дискретной математики и теории формальных языков, «в духе» [6, 7] (и мн. др.), т.е. «сверху-вниз». Поэтому в описании алгоритма постоянно опускаются слова «см. ниже», относящиеся к описываемым далее подалгоритмам.

Алгоритм построения РВ по заданному КА

Итак, пусть задан конечный автомат

$$K = (Q, \Sigma, \delta, S, f).$$

Для каждого стартового состояния $s \in S$ выбираем значения функции переходов δ , которые включают только *достижимые состояния* из стартового состояния s , и располагаем их в определенной последовательности. Обозначим эту последовательность $D(s)$. Построив $D(s)$ для каждого из стартовых состояний, строим регулярные выражения (пусть $re(D(s))$).

На выходе искомого РВ является объединением всех найденных $re(D(s))$.

1. Рекурсивная процедура построения $D(s)$

Эту процедуру будем обозначать $A(q)$. Суть $A(q)$ заключается в следующем. На вход подается состояние q и данная процедура выбирает все значения функции переходов для этого состояния:

$$\delta(q, A) = q_j, \text{ где } A - \text{некоторое РВ, } q_j \in Q.$$

Для каждого выбранного значения:

1. Осуществляется проверка, нет ли среди значений функции переходов значения, которое отличалось бы от найденного лишь РВ, т.е. значения типа:

$$\delta(q, B) = q_j, \text{ где } B - \text{некоторое РВ, } A \neq B.$$

Если такое значение найдено, то оно удаляется из списка значений функции переходов, а в найденном значении $\delta(q, A) = q_j$ РВ A заменяется на РВ $(A + B)$.

2. Каждое выбранное значение функции переходов $\delta(q, A) = q_j$ помещается в последовательность $D(s)$, и, если не выполняется условие останова рекурсии, для него происходит обращение к $A(q_j)$.

3. Условие останова рекурсии заключается в следующем: если для выбранного значения $\delta(q, A) = q_j$, то q_j «уже находится в обработке», т.е. ранее уже произошло обращение к $A(q_j)$.

Процедуру $A(q)$ запускаем, подав на вход стартовое состояние s . Результатом работы данной процедуры является построенная последовательность значений функции переходов $D(s)$.

Замечание. Описанная процедура $A(q)$ не является бесконечной, т.к., во-первых, определено условие останова рекурсии, и, во-вторых, любое значение функции переходов имеет право быть выбранным только один раз.

2. Процедура $R(D(s))$

Суть $R(D(s))$ заключается в следующем. На вход подается выбранная последовательность значений функции переходов $D(s)$, а на выходе получается РВ, язык которого задан $D(s)$.

Процедура $R(D(s))$ преобразует последовательность $D(s)$, выполняя в цикле три алгоритма. Первым выполняется *алгоритм 1*, причем столько раз, сколько его можно применить, затем выполняются *алгоритм 2* и *алгоритм 3*. Далее снова вызывается *алгоритм 1*, и так далее (цикл). Как только ни один из алгоритмов не применим - выходим из цикла.

Алгоритм 1 (избавление от циклов). Ищем в последовательности $D(s)$ значения типа:

$$\delta(q, A) = q, \text{ где } A - \text{некоторое РВ.}$$

Если такое значение найдено, то данный алгоритм считаем применимым к последовательности $D(s)$.

Пусть нашлось значение $\delta(q, A) = q$. Приступаем к преобразованию $D(s)$. Для этого:

- Фиксируем q и A .
- Удаляем из $D(s)$ найденное значение $\delta(q, A) = q$.
- Для всех $p \in Q$ находим в $D(s)$ для зафиксированного q все значения типа $\delta(q, B) = p$, где B - некоторое РВ. После этого заменяем в них B на $(A) * B$.

(Конец описания алгоритма 1.)

Замечание. Обозначим через $Q_s \subseteq Q$ состояния, которые используются в значениях функции переходов, записанных в $D(s)$.

Алгоритм 2 (избавление от промежуточных состояний). Выбираем состояние $q \in Q_s$ имеющее минимальное количество входов, т.е. состояние, имеющее минимальное количество значений функции переходов типа $\delta(p, A) = q$, где A — некоторое РВ. Указанное действие проделываем для последовательности $D(s)$ для всех $p \in Q_s$.

При этом состояние q не должно являться ни стартовым, ни финальным. Если такое состояние нашлось, то данный алгоритм считаем применимым к последовательности $D(s)$.

Пусть нашлось состояние q с соответствующими ему значениями $\delta(p_i, A) = q$. Тогда приступаем к преобразованию $D(s)$ для каждого i :

- Фиксируем p_i, A, q .
- Удаляем из $D(s)$ значение $\delta(p_i, A) = q$.
- Для зафиксированного q и всех $r \in Q$ находим в $D(s)$ все значения типа $\delta(q, B) = r$, где B — некоторое РВ. Для каждого из них преобразуем последовательность $D(s)$ следующим образом:
 - добавляем в $D(s)$ значение $\delta(p_i, AB) = r$.
 - сделав последнее возможное преобразование, удаляем из $D(s)$ все $\delta(q, B) = r$ и, после этого, удаляем из Q_s состояние q .

(Конец описания алгоритма 2.)

Алгоритм 3 (объединение одинаковых значений). Для всех $q, p \in Q$ ищем в последовательности $D(s)$ значение типа

$$\delta(q, A) = p, \quad \delta(q, B) = p,$$

где A и B — некоторые РВ.

Если такое значение найдено, то данный алгоритм считаем применимым к последовательности $D(s)$. В этом случае приступаем к преобразованию $D(s)$:

- фиксируем B ;

- удаляем из $D(s)$ найденное значение $\delta(q, B) = p$;
- в значении $\delta(q, A) = p$ изменяем A на $B + A$.

(Конец описания алгоритма 3.)

Литература

- [1] А. Саломая: *Жемчужины теории формальных языков*, М., Мир, 1986.
- [2] Б. Ф. Мельников: Об одной классификации последовательностных контекстно-свободных языков и грамматик. — *Вестник Моск. ун-та, сер. Вычисл. матем. и киб-ка*, 3 (1993) 64–69.
- [3] Б. Ф. Мельников: *Подклассы класса контекстно-свободных языков*, М., Изд-во Моск. ун-та, 1995.
- [4] О. Оре: *Теория графов*, М., Наука, 1980.
- [5] Ф. Харари: *Теория графов*, М., Мир, 1973.
- [6] У. Дал, Э. Дейкстра, К. Хоор: *Структурное программирование*, М., Мир, 1978.
- [7] Э. Йордан: *Структурное программирование и проектирование программ*, М., Мир, 1979.

Е. А. Михеева.

О наборах линейных функций из P_2

Под двужанной логикой P_2 понимается класс всех функций (от любого числа переменных), переменные которых определены на множестве E_2 и значения которых лежат в E_2 , с операцией суперпозиции над этими функциями, где $E_2 = \{0, 1\}$. Через

$$E_2^n = \underbrace{E_2 \times \dots \times E_2}_{n \text{ раз}}, n \geq 1$$

обозначаем декартово произведение множеств E_2 . Набор

$$(\alpha_1, \dots, \alpha_n) \in E_2^n$$

для краткости обозначим через $\tilde{\alpha}$. Функции вида

$$f(x_1, \dots, x_n) = a_1 x_1 + \dots + a_n x_n + a_{n+1} \pmod{2},$$

где $a_1, \dots, a_{n+1} \in E_2$, называются линейными функциями из P_2 .

1. Все функции, встречающиеся в данной работе, считаем линейными функциями из P_2 . Будем называть наборы из E_2^n точками. Набор функций

$$\langle f_1(x_1, \dots, x_n), \dots, f_s(x_1, \dots, x_n) \rangle \quad (1)$$

будем называть кортежем, а набор $(f_1(\tilde{\alpha}), \dots, f_s(\tilde{\alpha}))$ значением кортежа (1) в точке $\tilde{\alpha}$. Через $\bar{x}$ обозначаем функцию, определенную по формуле

$$\bar{x} = \begin{cases} 1, & \text{если } x = 0, \\ 0, & \text{если } x = 1. \end{cases}$$

Через $|A|$ обозначаем мощность множества A . Непустое подмножество J множества $\{1, 2, \dots, n\}$ называем направлением.

2. Введем теперь понятие самодвойственного множества. Для направления J и точки $\tilde{\alpha} \in E_2^n$ определим точку $\tilde{\alpha}^J$, положив для каждого $i, 1 \leq i \leq n$,

$$\alpha_i^J = \begin{cases} \bar{\alpha}_i, & \text{если } i \in J, \\ \alpha_i, & \text{если } i \notin J. \end{cases} \quad (2)$$

Для множества $A \subseteq E_2^n$ положим $A^J = \{\tilde{\alpha}^J | \tilde{\alpha} \in A\}$.

Замечание 1. Нетрудно понять, что $(\tilde{\alpha}^J)^J = \tilde{\alpha}$, $(A^J)^J = A$ и $|A^J| = |A|$.

Определение 1. Множество A самодвойственно по направлению J , если $A^J \subseteq A$.

Замечание 2. Условие $A^J \subseteq A$ равносильно условию $A^J = A$, так как по замечанию 1 $|A^J| = |A|$.

Назовем направление J возможным направлением самодвойственности для A , если $\tilde{\alpha}$ и $\tilde{\alpha}^J \in A$ хотя бы для одной точки $\tilde{\alpha}$.

Определение 2. Множество A самодвойственно, если A самодвойственно по любому возможному направлению самодвойственности.

Из определений 1 и 2 непосредственно вытекают 2 следствия.

Следствие 1. Если $|A| = 1$, то множество A самодвойственно.

Следствие 2. Множество $B = A \cup A^J$ самодвойственно по направлению J .

Докажем 2 леммы о самодвойственных множествах.

Лемма 1. Если A самодвойственно по направлению J_1 и J_2 , $J_1 \neq J_2$, то A самодвойственно по направлению

$$J = (J_1 \setminus J_2) \cup (J_2 \setminus J_1) = (J_1 \cup J_2) \setminus (J_1 \cap J_2).$$

Доказательство. Допустим $\tilde{\alpha} \in A$. Покажем сначала, что $(\tilde{\alpha}^{J_1})^{J_2} = \tilde{\alpha}^J$. Действительно из формулы (2) вытекает, что для всякого $i, 1 \leq i \leq n$,

$$\beta_i = \alpha_i^{J_1} = \begin{cases} \bar{\alpha}_i, & \text{если } i \in J_1, \\ \alpha_i, & \text{если } i \notin J_1, \end{cases} = \quad (3)$$

$$= \begin{cases} \bar{\alpha}_i, & \text{если } i \in J_1 \setminus J_2 \text{ или } i \in J_1 \cap J_2, \\ \alpha_i, & \text{если } i \in J_2 \setminus J_1 \text{ или } i \notin J_1 \cup J_2; \end{cases}$$

$$\beta_i^{J_2} = \begin{cases} \bar{\beta}_i, & \text{если } i \in J_2 \setminus J_1 \text{ или } i \in J_2 \cap J_1, \\ \beta_i, & \text{если } i \in J_1 \setminus J_2 \text{ или } i \notin J_2 \cup J_1. \end{cases} \quad (4)$$

Из равенств (3) и (4) получаем, что

$$(\alpha_i^{J_1})^{J_2} = \begin{cases} \bar{\alpha}_i, & \text{если } i \in J_2 \setminus J_1 \text{ или } i \in J_1 \setminus J_2, \\ \alpha_i, & \text{если } i \in J_2 \cap J_1 \text{ или } i \notin J_1 \cup J_2, \end{cases} = \\ = \begin{cases} \bar{\alpha}_i, & \text{если } i \in J, \\ \alpha_i, & \text{если } i \notin J, \end{cases} = \alpha_i^J.$$

Следовательно, $(\bar{\alpha}^{J_1})^{J_2} = \bar{\alpha}^J$.

Пусть $\bar{\alpha} \in A$. Тогда $\bar{\alpha}^{J_1} \in A$, поскольку A самодвойственно по направлению J_1 , и $(\bar{\alpha}^{J_1})^{J_2} \in A$, поскольку A самодвойственно по направлению J_2 , т.е. $\bar{\alpha}^J \in A$. Следовательно, A самодвойственно по направлению J .

Лемма 2. Для любых направления J и самодвойственного множества A либо $A \cap A^J = \emptyset$, либо $A = A^J$, и множество $B = A \cup A^J$ самодвойственно.

Доказательство. Допустим $A \cap A^J \neq \emptyset$. Тогда некоторая точка $\bar{\alpha}$ принадлежит $A \cap A^J$. Найдется точка $\bar{\beta} \in A$ такая, что $\bar{\beta}^J = \bar{\alpha}$, поскольку $\bar{\alpha} \in A^J$. Точка $\bar{\beta}^J$ тоже лежит в A , поскольку $\bar{\beta}^J = \bar{\alpha}$ и $\bar{\alpha} \in A$. Следовательно, J — возможное направление самодвойственности для A . Так как A самодвойственно, то $A^J = A$ по замечанию 2.

Итак, либо $A \cap A^J = \emptyset$, либо $A^J = A$. Если $A^J = A$, то $B = A$ и поэтому множество B самодвойственно.

Пусть $A \cap A^J = \emptyset$. По следствию 2 B самодвойственно по направлению J . Пусть I — возможное направление самодвойственности для B , т.е. $\bar{\alpha}$ и $\bar{\alpha}^I$ лежат в B для некоторой точки $\bar{\alpha}$, $I \neq J$. Возможны четыре случая.

Случай 1: $\bar{\alpha}$ и $\bar{\alpha}^I$ принадлежат A . В этом случае I — направление самодвойственности для A , поэтому $A^I = A$ и, следовательно,

$$(A^J)^I = (A^I)^J = A^J.$$

Но тогда

$$B^I = A^I \cup (A^J)^I = A \cup A^J = B$$

и, следовательно, B самодвойственно по направлению I . (Из доказательства леммы 1 нетрудно понять, что $(\bar{\alpha}^J)^I = (\bar{\alpha}^I)^J$, поэтому $(A^J)^I = (A^I)^J$.)

Случай 2: $\bar{\alpha}$ и $\bar{\alpha}^I$ принадлежат A^J . Найдется точка $\bar{\beta} \in A$ такая, что $\bar{\beta}^J = \bar{\alpha}$. Далее, $(\bar{\alpha}^I)^J \in A$ по замечанию 1. Но

$$(\bar{\alpha}^I)^J = ((\bar{\beta}^J)^I)^J = ((\bar{\beta}^I)^J)^J = \bar{\beta}^I.$$

Получаем, что $\bar{\beta}$ и $\bar{\beta}^I$ принадлежат A . По случаю 1 B самодвойственно по направлению I .

Случай 3: $\bar{\alpha} \in A$ и $\bar{\alpha}^I \in A^J$. Найдется точка $\bar{\beta} \in A$ такая, что $\bar{\beta}^J = \bar{\alpha}^I$. Пусть $K = (J \setminus I) \cup (I \setminus J)$. Получаем, что $\bar{\alpha}$ и $\bar{\alpha}^K$ лежат в A , поскольку

$$\bar{\alpha}^K = (\bar{\alpha}^J)^I = (\bar{\alpha}^I)^J = (\bar{\beta}^J)^J = \bar{\beta}.$$

По случаю 1 B самодвойственно по направлению K . Кроме того, B самодвойственно по направлению J , согласно следствию 2. По лемме 1 B самодвойственно по направлению $(J \setminus K) \cup (K \setminus J) = I$.

Случай 4: $\bar{\alpha} \in A^J$ и $\bar{\alpha}^I \in A$. Этот случай сводится к случаю 3. Итак, в любом случае множество B самодвойственно по направлению I . Следовательно, B самодвойственно.

3. Теперь мы можем доказать основной результат.

Теорема. Пусть D — множество значений кортежа (1) линейных функций из P_2 , где $s \geq 1, n \geq 1$. Тогда найдется число m , $0 \leq m \leq \min(n, s)$, такое, что $|D| = 2^m$. При этом множество D самодвойственно и каждое значение из D кортеж (1) принимает в 2^{n-m} точках.

Доказательство. Так как каждая функция $f_i (1 \leq i \leq s)$ принимает не более двух значений, то $|D| \leq 2^s$. С другой стороны, $|D| \leq |E_2^n| = 2^n$. Значит, $|D| \leq 2^{\min(n, s)}$. Если все функции кортежа (1) константы, то кортеж принимает только одно значение, откуда $2^0 \leq |D|$. Если мы покажем, что $|D| = 2^m$, то получим, что $0 \leq m \leq \min(n, s)$.

Докажем теперь утверждение теоремы индукцией по числу переменных в функциях кортежа (1).

Базис: $n = 1$. Каждая функция $f_i(x_1)$, $1 \leq i \leq s$, либо константа β_i из E_2 , либо функция $x_1 + d_i$, где $d_i \in E_2$ (здесь и далее сложение ведется по модулю 2). Обозначим через I множество индексов i , для которых $f_i(x_1) = x_1 + d_i$.

Если множество I пусто, то $D = \{(\beta_1, \dots, \beta_s)\}$. По следствию 1 множество D самодвойственно. Возьмем $m = 0$. Тогда $|D| = 2^0$

и значение $(\beta_1, \dots, \beta_s)$ кортеж (1) принимает на множестве E_2 , т.е. в 2^{1-0} точках.

Пусть множество I непусто. Тогда $D = \{\bar{\sigma}, \bar{\sigma}^I\}$, где для каждого $i, 1 \leq i \leq s$,

$$\sigma_i = \begin{cases} \alpha_i, & \text{если } i \in I, \\ \beta_i, & \text{если } i \notin I. \end{cases}$$

Очевидно, для D возможно только одно направление самодвойственности - направление I , и D самодвойственно по этому направлению. Получаем, что D самодвойственно, $|D| = 2^m$, где $m = 1$, и каждое значение из D кортеж (1) принимает в 2^{n-m} точках, где $n = 1, m = 1$.

Индукционный шаг. Пусть для $n = t$ утверждение леммы верно, покажем, что оно верно для $n = t + 1$. Для этого считаем, что в кортеже (1) n равно числу $t + 1$.

Рассмотрим два следующих кортежа:

$$\langle f_{01}(x_1, \dots, x_t), \dots, f_{0s}(x_1, \dots, x_t) \rangle, \quad (5)$$

$$\langle f_{11}(x_1, \dots, x_t), \dots, f_{1s}(x_1, \dots, x_t) \rangle; \quad (6)$$

где $f_{ji}(x_1, \dots, x_t) = f_i(x_1, \dots, x_t, j) (j = 1, 2; i = 1, 2, \dots, s)$.

Обозначим через D_0 и D_1 множества значений кортежей (5) и (6), соответственно.

По индукционному предположению найдется число $m_0, 0 \leq m_0 \leq \min(t, s)$, и множества точек $A_1, \dots, A_{2^{m_0}}$ из E_2^t такие, что:

а) множество D_0 самодвойственно, $|D_0| = 2^{m_0}, D_0 = \{\bar{\sigma}_1, \dots, \bar{\sigma}_{2^{m_0}}\}$;

б) на множестве A_j кортеж (5) принимает значение $\bar{\sigma}_j$,

$$|A_j| = 2^{t-m_0} \quad (1 \leq j \leq 2^{m_0}).$$

Далее возможны два случая:

Случай 1: x_{t+1} - фиктивная переменная во всех функциях кортежа (1).

Тогда $D_1 = D_0$ и на множестве A_j кортеж (6) принимает значение $\bar{\sigma}_j, (1 \leq j \leq 2^{m_0})$. Положим $m = m_0, B_j = A_j \times E_2$ для $j = 1, 2, \dots, 2^{m_0}$. Имеем:

а) $D = D_0$, поэтому самодвойственно, $|D| = |D_0| = 2^m$, $0 \leq m \leq \min(t + 1, s)$;

б) $|B_j| = 2^{(t+1)-m}$ и на B_j кортеж (1) принимает значение $\bar{\sigma}_j, (1 \leq j \leq 2^m)$.

Следовательно, утверждение теоремы верно для кортежа (1).

Случай 2: x_{t+1} существенна хотя бы в одной функции кортежа (1).

Обозначим через I множество индексов i , таких, что $f_i(x_1, \dots, x_{t+1})$ существенно зависит от x_{t+1} . Тогда на множестве A_j кортеж (1) принимает значение $\bar{\sigma}_j^I$ для каждого $j, 1 \leq j \leq 2^m$. Следовательно, $D_1 = D_0^I$.

По лемме 2 множество $D = D_0 \cup D_1$ самодвойственно и либо $D_1 = D_0$, либо $D_1 \cap D_0 = \emptyset$. Если $D_1 = D_0$, то как в случае 1 получаем, что утверждение теоремы верно для кортежа (1).

Пусть $D_0 \cap D_1 = \emptyset$. Тогда $D_1 = \{\bar{\sigma}_1^I, \dots, \bar{\sigma}_{2^{m_0}}^I\}$ и на множестве A_j кортеж (6) принимает значение $\bar{\sigma}_j^I (1 \leq j \leq 2^{m_0})$. Положим $m = m_0 + 1$,

$$B_j = \begin{cases} A_j \times \{0\}, & \text{если } j = 2r - 1, \\ A_j \times \{1\}, & \text{если } j = 2r, \end{cases}$$

для $j = 1, 2, \dots, 2^m$. Имеем:

а) D -самодвойственно, $|D| = |D_0| + |D_1| = 2^m, 0 \leq m \leq \min(t + 1, s)$;

б) $|B_j| = |A_j| = 2^{(t+1)-m}$ и на B_j кортеж (1) принимает значение $\bar{\sigma}_r$, если $j = 2r - 1$, и значение $\bar{\sigma}_r^I$, если $j = 2r (1 \leq j \leq 2^m)$.

Следовательно, утверждение теоремы верно для кортежа (1).

Итак, утверждение теоремы верно при $n = t + 1$, если верно при $n = t$. На этом индукционный шаг заканчивается.

По индукции получаем, что утверждение теоремы верно для любого $n \geq 1$. Теорема доказана.

Литература

1. Яблонский С.В., Гаврилов Г.П., Кудрявцев В.Б. *Функции алгебры логики и классы Поста.* // М.: Наука, 1966.

Н. Н. Нечаева.

Методика анализа линейных моделей в языке ФС

В языке ФС математическая модель (ММ), к которой приводится описание модели объекта на входном языке, имеет вид системы дифференциально - алгебраических уравнений (ДАУ), разрешенной относительно производных:

$$Ax'(t) = Bx(t) + c(x, t), \quad (1)$$

где $t \in J = [\alpha, \beta] \subset R$, $x'(t)$, $x(t) \in R^n$, A , B - матрицы размера $n \times n$, причем A - вырожденная матрица, $c(x, t)$ - вектор-функция внешних воздействий и связей (отображение $c: D \subset R^{n+1} \rightarrow R^n$), причем $x^0 = x(t_0)$.

При реализации статического анализа данная система ДАУ (1) преобразуется в систему алгебраических уравнений для последующего ее исследования. При реализации динамического анализа ММ имеет вид системы ДАУ (линейной или нелинейной).

Если отсутствуют нелинейные связи между подсистемами объекта, математическая модель принимает вид системы линейных дифференциально - алгебраических уравнений:

$$Ax'(t) = Bx(t) + c(t), \quad (2)$$

где A , B - матрицы размера $n \times n$, причем A - вырожденная матрица, $c(t)$, $x(t)$ - известная и искомая вектор-функции, $c(t)$, $x(t) \in R^n$.

Для интегрирования данной системы в математическом блоке языка ФС используются:

- 1) неявный метод первого порядка [1], построенный на основе формулы Обрешкова

$$(A - \tau B)x_{k+1} = Ax_k + \tau c(t_{k+1}), \quad \tau = T/N, \quad (3)$$

- 2) двухшаговая (трехточечная) разностная схема [1,3]

$$A(3x_{k+1} - 4x_k + x_{k-1})/(2\tau) = Bx_{k+1} + c(t_{k+1}), \quad (4)$$

которая обладает достоинствами неявного метода Эйлера и, к тому же, вторым порядком сходимости [2].

Доказательство применимости этих методов можно найти в [1,2,3]. Основным условием их сходимости является достаточная гладкость функции $c(t)$. Поэтому то обстоятельство, что при построении ММ используются только гладкие функции и их линейные комбинации дает основание для выполнения условия сходимости применяемых методов.

В результате применения формулы (3) к системе линейных ДАУ (2) получим следующую систему уравнений:

$$\begin{cases} \sum_j (a_{1j} - \tau b_{1j}) x_j^{k+1} = \sum_j a_{1j} x_j^k + \tau c_1(t_{k+1}), \\ \dots \dots \dots \\ \sum_j (a_{nj} - \tau b_{nj}) x_j^{k+1} = \sum_j a_{nj} x_j^k + \tau c_n(t_{k+1}), \end{cases} \quad j = 1, \dots, n,$$

которую решаем на каждом шаге методом Гаусса с выбором главного элемента по строке.

Рекуррентную формулу (4) можно переписать следующим образом:

$$(3A - 2\tau B)x_{k+1} = A(4x_k - x_{k-1}) + 2\tau c(t_{k+1}). \quad (5)$$

В разностной схеме (4) необходима информация о двух предыдущих точках. Поэтому для нахождения стартовых значений для метода (4) используем метод 2-го порядка, полученный на основе формулы Обрешкова [1]

$$(A - \frac{\tau}{2}B)x_{k+1} = (A + \frac{\tau}{2}B)x_k + \frac{\tau}{2}[c(t_{k+1}) + c(t_k)]. \quad (6)$$

В результате применения к системе (2), получим систему алгебраических уравнений:

$$\begin{cases} \sum_j (a_{1j} - \frac{\tau}{2}b_{1j}) x_j^{k+1} = \sum_j (a_{1j} + \frac{\tau}{2}b_{1j}) x_j^k + \frac{\tau}{2}[c_1(t_{k+1}) + c_1(t_k)], \\ \dots \dots \dots \\ \sum_j (a_{nj} - \frac{\tau}{2}b_{nj}) x_j^{k+1} = \sum_j (a_{nj} + \frac{\tau}{2}b_{nj}) x_j^k + \frac{\tau}{2}[c_n(t_{k+1}) + c_n(t_k)], \end{cases} \quad j = 1, \dots, n,$$

для решения которой удобно использовать метод исключений Гаусса.

Применяя на следующих шагах (5) к (2), будем иметь следующую систему уравнений:

$$\begin{cases} \sum_j (3a_{1j} - 2\tau b_{1j}) x_j^{k+1} = \sum_j a_{1j} (4x_j^k - x_j^{k-1}) + 2\tau c_j(t_{k+1}), \\ \dots \dots \dots \\ \sum_j (3a_{nj} - 2\tau b_{nj}) x_j^{k+1} = \sum_j a_{nj} (4x_j^k - x_j^{k-1}) + 2\tau c_j(t_{k+1}) \end{cases},$$

$$j = 1, \dots, n.$$

Как и в предыдущих случаях данную систему алгебраических уравнений решаем относительно x_j ($j = 1, \dots, n$) на каждом шаге, используя метод Гаусса.

Рассмотренные методы интегрирования линейных систем ДАУ реализованы в математическом блоке анализа языка ФС и показали свою практическую эффективность.

Литература

1. Бояринцев Ю.Е.; Данилов В.А., Логинов А.А., Чистяков В.Ф. Численные методы решения сингулярных систем. - Новосибирск: Наука, 1989.
2. Данилов В.А. Двухшаговая схема для решения системы обыкновенных дифференциальных уравнений с вырожденной матрицей перед производными. // Приближенные методы решения операторных уравнений и их приложения. - Иркутск: СЭИ СО АН СС-СР, 1982.
3. Ракитский Ю.В., Устинов С.М., Черноруцкий И.Г. Численные методы решения жестких систем. - М.: Наука, 1979.

А. Ф. Николаев.

Об одной постановке задачи о множественной разладке

§1. Введение

Задача о скорейшем обнаружении разладки в непрерывном времени сформулирована и решена Ширяевым А.Н. в [1]. При этом доступный наблюдению процесс $x = (x_t)_{t \geq 0}$, $x_0 = 0$ имеет дифференциал вида

$$dx_t = rI\{t > \theta\}dt + \sigma dW_t,$$

где θ момент наступления разладки.

В настоящей работе предполагается, что процесс $x = (x_t)_{t \geq 0}$, $x_0 = 0$, доступный наблюдению, имеет дифференциал вида

$$dx_t = \sum_{i=1}^n a_i I\{t \geq \tau_i\}dt + \sigma dW_t,$$

где τ_i , $i = 1, \dots, n$ представляют собой моменты появления множественной разладки.

В основе предлагаемого способа оценивания этих моментов лежит метод, используемый Ширяевым А.Н. в задаче о последовательном различении двух простых гипотез о среднем значении винеровского процесса (см. [1]).

§2. Постановка задачи и формулировка основных результатов.

Пусть на вероятностном пространстве $(\Omega, \mathcal{F}, P)$ заданы:

- 1) последовательность, состоящая из n случайных величин τ_i :

$$\tau_i = \tau_{i-1} + \eta_i, \quad \tau_0 = 0,$$

$$P\{\eta_i \geq t\} = e^{-\lambda t}, \quad i = 1, \dots, n;$$

- 2) непрерывный справа процесс $\theta = (\theta_t)_{t \geq 0}$ с

$$\theta_t = \sum_{i=1}^n a_i I\{t \geq \tau_i\}, \quad \theta_0 = 0, \quad (1)$$

где a_i некоторые отличные от 0 константы, такие что этот процесс марковский.

Наблюдению доступен стохастический процесс $x = (x_t)_{t \geq 0}$ с дифференциалом

$$dx_t = \theta_t dt + \sigma dW_t, \quad x_0 = 0, \quad (2)$$

где σ некоторая константа, такая что $\sigma \geq \epsilon > 0$, $W = (W_t)_{t \geq 0}$ — винеровский процесс, независимый от $\{\eta_i, i = 1, \dots, n\}$.

По реализации процесса x требуется построить оценки случайных величин $\tau_i, i = 1, \dots, n$.

Пусть $F = (F_t)_{t \geq 0}$, где $F_t = \sigma\{x_s, s \leq t\}$. Введем множество значений уровней марковского процесса θ , определенного в (1):

$$A = \{A_i : A_0 = 0, A_i = \sum_{k=1}^i a_k, i = 1, \dots, n\}. \quad (3)$$

Пусть

$$D_t = \{d_t = d_t(\omega) : d_t \in A, d_t \sim F_t \text{ измерима}\}.$$

$d_t \in D_t$ является решающей функцией, которая позволяет определить уровень, на котором находится процесс θ в момент времени t . С каждой решающей функцией можно соотнести потери, связанные с ошибочностью заключительного решения. Их можно описать следующим образом:

$$Q(t, d_t) = \{x_i, \text{ если } (\theta_t(\omega) \neq A_i, d_t(\omega) = A_i), i = 0, \dots, n,$$

где весовые коэффициенты $x_i > 0$. Тогда средние потери от принятия решения $d_t = d_t(\omega)$ имеют вид:

$$\rho(t, d_t) = \sum_{i=0}^n x_i P\{(d_t = A_i) \cap (\theta_t \neq A_i)\}. \quad (4)$$

Задача состоит в минимизации функционала потерь, определенного в (4):

$$\rho(t, d_t) \rightarrow \inf_{d_t \in D_t}. \quad (5)$$

Обозначим апостериорную вероятность того, что процесс θ_t находится на уровне A_i

$$\pi_t^i = P\{\theta_t = A_i | F_t\}. \quad (6)$$

Пусть

$$m_t = \min_{0 \leq i \leq n} x_i(1 - \pi_t^i). \quad (7)$$

Введем также функцию N , определяющую по индексированному элементу множества его индекс:

$$N(x_t) = l. \quad (8)$$

Теорема. Пусть a_i , участвующие в определении марковского процесса θ в (1) таковы, что множество A в (3) состоит из различных элементов. Тогда решением задачи (2), (5) является функция

$$d_t^* = A_{N(m_t)},$$

где m_t определено в (7), N определено в (8). Дифференциал процесса π_t^i имеет вид:

$$d\pi_t^i = L\pi_t^i dt + \pi_t^i \frac{A_i - \bar{A}}{\sigma} dW_t, \quad \pi_0^i = \sigma_{i0}, \quad (9)$$

где σ_{ki} — символ Кронеккера,

$$L\pi_t^i = \begin{cases} -\lambda\pi_t^i, & \text{если } i = 0 \\ \lambda(\pi_t^{i-1} - \pi_t^i), & \text{если } i = 1, \dots, n \\ 0, & \text{если } i = n, \end{cases} \quad (10)$$

$$\bar{A}_t = \sum_{i=1}^n A_i \pi_t^i, \quad (11)$$

а винеровский процесс $W = (W_t)_{t \geq 0}$ имеет вид

$$\bar{W}_t = \int_0^t \frac{dx_s - \bar{A}_s ds}{\sigma}. \quad (12)$$

Для нас практический интерес представляет

Следствие. Оценки неизвестных моментов разладки имеют вид

$$\tau_i^* = \inf\{t \geq \tau_{i-1}^* : d_t^* = A_i\}, \quad i = 1, \dots, n, \quad \tau_0^* = 0. \quad (13)$$

Идея доказательства состоит в преобразовании функционала потерь, введенного в (4), к виду, из которого сразу следует решение задачи (2), (5). Для этого вводится новая мера

$$\tilde{P}(\cdot) = P(\cdot | \theta_t \neq A_t),$$

которая оказывается абсолютно непрерывной относительно исходной меры P . В силу формулы Байеса

$$1 - \pi_t^i = P\{\theta_t \neq A_t\} E\left[\frac{d\tilde{P}}{dP}(\omega) | F_t\right],$$

где π_t^i определено в (6). Следовательно

$$P\{(d_t = A_t) \cap (\theta_t \neq A_t)\} = E[I\{d_t = A_t\} * (1 - \pi_t^i)].$$

Следовательно

$$\rho(t, d_t) = E \sum_{i=0}^n [x_i (1 - \pi_t^i) I\{d_t = A_i\}]. \quad (14)$$

Из данного представления легко получить, что $\forall d_t \in D_t$

$$\rho(t, d_t) \geq E[m_t \sum_{i=0}^n I\{d_t = A_i\}],$$

причем равенство возможно, если

$$d_t^* = A_{N(m_t)}. \quad (15)$$

Таким образом, d_t^* - искомое решение задачи (2), (5). Для завершения доказательства необходимо полностью определить d_t^* , а для этого надо уметь вычислять π_t^i . Используя теорему о фильтрации для марковских процессов со счетным числом состояний (см. [2], глава 9). Следуя обозначениям этой теоремы, получаем формулы (9) - (12).

Доказательство следствия основано на том факте, что функция d_t^* , построенная в теореме, позволяет определить в каждый момент времени уровень, на котором находится марковский процесс θ . Выбирая наименьший момент времени t , при котором d_t^* принимает заданное значение A_k , получаем оценку для момента разладки τ_k (см. (13)).

Литература

- [1]. Липцер Р. Ш., Ширяев А. Н. Статистика случайных процессов, Наука, М. 1974, 696 с.;
- [2]. Липцер Р. Ш., Ширяев А. Н. Теория мартингалов Наука, М. 1986, 512 с.

А. Н. Радионов.

Игровые ассистирующие программы

Игровые программы-ассистенты, предназначены для оценки выбранного игроком хода и дачи рекомендаций по дальнейшему развитию игры. Оценивание может осуществляться перебором возможных вариантов развития партии и вычислением выигрыша противника в самом благоприятном для него варианте. Эту величину программа получает после просмотра партии на некоторую фиксированную глубину и предлагает её игроку-партнеру в качестве оценки конкретной позиции. Предполагается также умение программы показывать критические ветви игры, то есть ту последовательность ходов, при которой и та и другая сторона играет максимально выгодно. Использование подобных программ в игре даст возможность раньше заметить "ловушки" противника, уменьшит количество тактических ошибок, возникающих из-за невнимательности, а также поможет научиться реально оценивать игровую ситуацию.

Вариант ассистирующей программы был написан в системе Delphi 2.0 для MS Windows 95. Он ассистирует (во всяком случае есть на это надежда) при игре в шахматы. Ассистирование в этой программе происходит следующим образом. Играющий делает пробный ход, на экране появляются возможные варианты ответов противника с пометками-оценками этих ответов. Количество вариантов ответа противника ограничивается косвенно некоторым заранее заданным относительным значением оценки позиции. Это нужно для того, чтобы не рассеиваться на невыгодные противнику ходы (хотя при желании этот ограничитель можно ослабить) и сделать количество вариантов более обозримым. "Относительность" ограничителя оценок обосновывается тем, что на протяжении партии одна из сторон может выигрывать и при этом уровень оценок другой стороны будет понижаться, что в итоге может привести к тому, что любые ответы проигрывающей стороны окажутся ниже ограничителя заданного абсолютно. Ограничитель задается через процентное отношение. Например, если ограничитель равен 10%, то для пользователя невидны 10% вариантов ответа противника. Согласно эвристике, предложенной в методе минимакса ([3]), вероятность нахождения самого опасного ответа противника увеличивается с

ростом глубины дерева перебора. Помимо "предсказания" ответов противника, программа способна показывать на заданную глубину критические ветви развития игры, начиная с некоторой позиции. Поэтому пользователь имеет возможность моделирования дальнейшего развития партии, с помощью чего могут раскрыться "коварные" замыслы противника.

На основании методов, описанных в [1-3], возможно построение программ-ассистентов для других областей практики, в том числе содержащих трудно формализуемые факторы.

Литература

1. А. Н. Радионов. "Критерии риска в программных моделях игр с нулевой суммой" в сборнике "Фундаментальные проблемы математики и механики", издательство Ульяновского Государственного Университета, 1996.
2. А. Н. Радионов. "Некоторые критерии риска в играх с неполной информацией" в сборнике "Тезисы докладов на XI Международной конференции по проблемам теоретической кибернетики", Ульяновск 1996, стр. 168, издательство СВНЦ, 1996.
3. Г. М. Адельсон-Вельский, В. Л. Арлазаров, М. В. Довской "Программирование игр". Москва, "Наука", 1978.

А. Г. Сковиков, Л. В. Калинин.

Определение моментов нарушений в линейной динамической системе

Рассмотрим линейную динамическую систему, возмущаемую гауссовским дискретным белым шумом w_t и заданную следующими уравнениями

$$x_t = \Phi_t x_{t-1} + w_t, \quad (1)$$

$$z_t = H_t x_t + v_t, \quad (2)$$

где Φ_t и H_t - матрицы размеров $(n \times n)$ и $(m \times n)$ соответственно. Совместно независимые векторы w_t и v_t размерности n и m нормально распределены с нулевыми математическими ожиданиями и ковариационными матрицами Q_t и R_t размеров $(n \times n)$ и $(m \times m)$ соответственно.

При заданных матрицах Φ_t, H_t, Q_t, R_t оптимальное решение задачи оценивания получается с помощью известного фильтра Калмана. Этот фильтр оптимален в том смысле, что дает последовательную несмещенную оценку $\hat{x}(t|t)$ вектора состояния системы с минимальной дисперсией компонент в виде линейной комбинации измерений $z_i, i = 1, 2, \dots, t$.

Неадекватность модели системы, принятой при построении фильтра Калмана, реальным условиям ее функционирования (назовем такую ситуацию - нарушением) приведет к значительному ухудшению качества работы данного фильтра. Это обстоятельство можно использовать при обнаружении момента нарушения в линейной динамической системе. Т.е. мы будем определять сбой системы, фиксируя потерю свойств оптимальности фильтром Калмана.

Процедура обнаружения расходимости процесса калмановской фильтрации должна основываться на обработке доступной для наблюдения информации о фактическом состоянии системы. С этой целью может быть использована обновляющая последовательность экстраполированных оценок и измерений, теоретически имеющая нулевое математическое ожидание и ковариацию, вычисляемую согласно уравнению

$$\Sigma_t = H_t P(t|t-1) H_t^T + R_t,$$

где $P(t|t-1)$ - матрица ковариации ошибок оптимального одношагового предсказания.

Если в обновляющей последовательности появляется смещение, то можно предположить появление расхождений в принятой на данный момент модели системы и ее действительного состояния [1].

Возрастание ошибки фильтрации при расходимости процесса рекуррентной обработки данных часто имеет не знакопеременный, а монотонный характер. В этом случае для обнаружения потери фильтром Калмана свойства оптимальности может быть использована концепция последовательного анализа Вальда.

Предположим, что характер расходимости процесса фильтрации такой, что обновляющая последовательность сохраняет нормальный закон распределения вероятностей. Причем разница между фактическим значением ковариации обновляющей последовательности и значением, вычисляемым фильтром пренебрежимо мала. Введем в рассмотрение три гипотезы:

$$\begin{cases} \mathcal{H}_0: & \nu(i) \sim \mathcal{N}(0, \Sigma_t) \\ \mathcal{H}_1: & \nu(i) \sim \mathcal{N}(c, \Sigma_t) \\ \mathcal{H}_2: & \nu(i) \sim \mathcal{N}(-c, \Sigma_t), \end{cases}$$

где c - некоторый постоянный вектор.

Первая гипотеза $\mathcal{H}_0$ предполагает оптимальность фильтра и, следовательно, адекватность принятой модели системы реальным условиям функционирования. Согласно двум другим ($\mathcal{H}_1$ и $\mathcal{H}_2$) фильтр потерял свойство оптимальности, т.е. произошло нарушение. Поскольку знак появляющегося смещения среднего значения невязки в общем случае априорно неизвестен, последнее событие должно описываться двумя гипотезами. Предполагая независимость отсчетов последовательности невязок измерений, запишем ее функции правдоподобия для введенных гипотез

$$\begin{cases} P(\nu(1), \dots, \nu(t) | \mathcal{H}_0) = \prod_{i=1}^t P(\nu(i) | \mathcal{H}_0), \\ P(\nu(1), \dots, \nu(t) | \mathcal{H}_1) = \prod_{i=1}^t P(\nu(i) | \mathcal{H}_1), \\ P(\nu(1), \dots, \nu(t) | \mathcal{H}_2) = \prod_{i=1}^t P(\nu(i) | \mathcal{H}_2). \end{cases} \quad (3)$$

Решающее правило обнаружения расходимости процесса калмановской фильтрации при заданных вероятностях α и β ошибок

первого и второго рода записывается следующим образом

$$\begin{cases} 1. \text{ Если } \lambda_{i0}(t) \leq B, \text{ тест прекращают} \\ \quad \text{с выбором гипотезы } \mathcal{H}_0. \\ 2. \text{ Если } \lambda_{i0}(t) \geq A \text{ или } \lambda_{20}(t) \geq A, \\ \quad \text{тест прекращают с выбором гипотезы } \mathcal{H}_1 \text{ или } \mathcal{H}_2. \\ 3. \text{ Если } B < \lambda_{i0}(t) < A, t=1,2, \text{ тест продолжают для} \\ \quad \text{следующего } (t+1) \text{ го шага,} \end{cases} \quad (4)$$

где

$$\lambda_{i0}(t) = \ln \frac{P(\nu(1), \dots, \nu(t) | \mathcal{H}_i)}{P(\nu(1), \dots, \nu(t) | \mathcal{H}_0)}$$

есть логарифм отношения правдоподобия.

В работе [2] показано, что выражение для вычисления логарифма отношения правдоподобия можно переписать в удобной рекуррентной форме

$$\lambda_{i0}(t) = \lambda_{i0}(t-1) + \mu_{i0}(t) \quad (5)$$

где $\mu_{i0}(t) = c^T \Sigma_t^{-1} [\nu(t) - \frac{1}{2}c], i=1,2$.

Так как момент возможного возникновения расходимости неизвестен, необходимо формировать функционалы вида (5), начиная с каждого очередного момента времени, т.е. число таких функционалов будет возрастать. Избежать этого можно, приняв во внимание следующие соображения. Предполагая, что перед началом тестирования условие адекватности модели соблюдалось, можно не обращать внимание на выход величины логарифма отношения правдоподобия за нижний критический порог B решающего правила, так как это событие не несет в себе новой информации. Важно только сохранять неизменным начальное расстояние между верхним критическим порогом A и постоянно уменьшающейся при оптимальности фильтра величиной $\lambda_{i0}(t)$. Для этого сделаем верхний порог скользящим вниз, т.е. для каждого отрицательного значения $\mu_{i0}(t)$ на такую же величину должно уменьшиться текущее значение порога A . В результате сохраняется первоначальная ширина коридора неопределенности для выбора одной из гипотез $\mathcal{H}_1$ или $\mathcal{H}_2$. Если $\mu_{i0}(t) > 0, i=1,2$, значение порога A не изменяется. Следовательно, решающее правило (4) преобразуется к виду:

- $$\left\{ \begin{array}{l} 1. \text{ Если } \lambda_{10}(t) \geq A, \text{ тест прекращают с выбором} \\ \text{ гипотезы } \mathcal{H}_1 \text{ и вырабатывают соответствующий} \\ \text{ сигнал подозрения.} \\ 2. \text{ Если } \lambda_{20}(t) \geq A, \text{ тест прекращают с выбором} \\ \text{ гипотезы } \mathcal{H}_2 \text{ и вырабатывают соответствующий} \\ \text{ сигнал подозрения.} \\ 3. \text{ Иначе тест продолжают для шага } t+1. \end{array} \right. \quad (6)$$

Компоненты вектора λ с определяются экспериментально, так чтобы обеспечить высокую чувствительность процедуры подозрения.

В общем случае правило (6) формирует сигнал подозрения с некоторым запаздыванием. Поэтому целесообразно дополнить его инструментом обнаружения наиболее правдоподобного момента возникновения нарушения.

Пусть при обработке t -го измерения величина $\lambda_{i0}(t)$ превысила верхний критический порог A . Введем функцию правдоподобия появления расходимости фильтра на l -ом шаге

$$\Lambda_l = P(\nu(1), \dots, \nu(l-1) | \mathcal{H}_0) P(\nu(l), \dots, \nu(t) | \mathcal{H}_1). \quad (7)$$

Логарифм выражения (7) равен

$$\ln \Lambda_l = \sum_{i=1}^{l-1} \ln \frac{P(\nu(i) | \mathcal{H}_0)}{P(\nu(i) | \mathcal{H}_1)} + \sum_{i=1}^t \ln (P(\nu(i) | \mathcal{H}_1)).$$

Второе слагаемое уравнения (7) есть постоянная. Следовательно, максимум этого уравнения по l совпадает с максимумом первого слагаемого. Следовательно, максимально правдоподобный момент возникновения нарушения - следующий за моментом времени, которому соответствует минимум логарифма отношения правдоподобия (5).

Таким образом, используя процедуру обнаружения смещения среднего значения обновляющей последовательности, можно зафиксировать момент нарушения в линейной динамической системе (1), (2).

Литература

1. Малютин Ю.М., Экало А.В. Применение ЭВМ для решения задач идентификации объектов. Л.: Изд-во Ленинградского ун-та, 1988.

2. Семущин И.В., Калинин Л.В. Обнаружение нарушений в моделях стохастических систем // Измерительная техника - 1996. - N 3 - С. 9-11.

В. В. Угаров.

Об одной методике оптимизации ввода данных в автоматизированных системах измерений

Функционирование автоматизированных систем измерений (АСИ) базируется на обработке поступающей информации от внешних источников и выводе результирующей информации на устройства отображения, хранения и на исполнительные устройства.

Наряду с получением информации в реальном масштабе времени от датчиков, в практике работы АСИ часто встречается задача обработки данных, полученных опосредствованно. В частности, получение информации с мест, не оборудованных датчиками, не имеющих электронных каналов связи с АСИ (локальные сети, модемная связь по телефонным каналам и т. п.), производится на материальных носителях (чаще всего на бумаге), доставка которых происходит с существенной временной задержкой.

Рассмотрим вопросы ввода информации, размещенной на так называемых бланках, т. е. бумажных носителях, форма которых, разметка и алгоритм занесения информации жестко фиксированы (маршрутные карты, анкеты, отчетные формы).

Такие бланки широко применяются при проведении серий измерений в научных исследованиях, в финансово-экономической деятельности, в социологических опросах и т. д.

Рассмотрим более подробно структуру бланка. Бланк — это совокупность полей, используемых для занесения либо метки, либо числового или текстового значения. Каждое поле имеет наименование и/или уникальный (неповторяющийся) номер.

Например, бланки анкет для опроса населения состоят из полей вопросов и совокупности альтернативных ответов, один или несколько которых респондент должен пометить.

В современных условиях все еще в большинстве случаев данные с бланков вводятся вручную. Автоматический ввод с помощью сканера и программ распознавания экономически приемлим в условиях больших объемов сильно формализованной информации, ввод которой периодически повторяется. Кроме того, бланки должны быть изготовлены с высоким качеством, заполнение бланков про-

изводится в основном пометками, в некоторых случаях могут вводиться цифровые данные.

В случае же нерегулярных исследований с небольшим объемом информации ввод производится вручную оператором компьютера.

Проблема ввода заключается прежде всего в уменьшении затрат на ввод данных с бланка анкеты в компьютер, т. е. на преобразование информации в электронную форму и снижении ошибок ввода.

В качестве примера приведем фрагмент бланка, заполняемого при испытании контактной электрической аппаратуры:

10. Серия испытаний.

Дата проведения	19.10.97	—	51
Кол. испытаний	151	—	52
Шифр испытания	C14-5	—	53

11. Параметры испытательного стенда.

Напряжение (вольт)			
	220 вольт	✓	62
	160 вольт	—	63
	110 вольт	—	64
Тип контактора			
	KHT-4345y	—	71
	KHT-4367и	✓	72

На фрагменте показаны формы для заполнения ответов на два запроса с номерами 10 и 11. На бланке поля 51..53 заполняются текстовыми и цифровыми данными, а поля 62..64, 71..72 метками, соответственно выбранным ответам. Выбор одного или нескольких ответов из предлагаемых часто более эффективен, чем формирование собственного ответа, и сопровождается меньшими ошибками.

Бланки такого типа, как показывает анализ более пятнадцати их видов, заполняется респондентами далеко не всегда полностью. В частности, на выборке из 180 бланков трех видов один бланк в среднем имеет 6,9 запросов, 57,4 альтернативных ответов, из них заполнено 17,4. Таким образом, бланк в среднем заполняется на 30,3%.

Большинство пакетов обработки данных требуют при вводе бланков информацию о каждом поле ответа, независимо от того, заполнено оно или нет. К ним, в том числе, относятся и MS Access, MS Excel, Paradox, DBase и т. д.

Основной особенностью ввода в этих пакетах и им подобных является необходимость прохода через все поля бланка, даже если в бланке эти поля не помечены и в них нет данных. Если поля заполнены только на 30%, то оператору необходимо выполнить в три раза больше нажатий на клавиши, чтобы ввести данные с бланка. Чтобы переходить непосредственно от одного заполненного поля к другому заполненному полю, можно применить манипулятор "мышь", но в этом случае частые переключения с клавиатуры на "мышь" резко замедляют действия оператора.

В данной работе предлагается выполнять ввод данных с клавиатуры только заполненных полей бланка. Если поле бланка имеет пометку о выборе альтернативного ответа, вводится только < номер поля >, если же поле заполнено числовой или текстовой информацией, то вводится связанная пара < значение поля > - < номер поля >. Поля связываются некоторым символом, в данном случае тире. В строке ввода данные отделены друг от друга символом-разделителем, например, запятой. Пример строки ввода некоторого бланка:

.12,5 - 1,4,11,45 - 17,67 - 18,28,31,33,113 - 35

Пусть точка и следующее за ней число в начале строки будет номером бланка, тогда последующие числа есть параметры бланка. В приведенном примере поле "1" имеет значение "5", поля "4" и "11" отмечены меткой, поля "17" и "18" имеют значения соответственно "45" и "67" и т. д.

Предлагаемый способ ввода позволяет примерно в два раза сократить количество нажатий клавиш по сравнению с традиционным, что способствует уменьшению количества ошибок. Некоторое увеличение трудоемкости связано с необходимостью ввода номеров полей бланка. Косвенным преимуществом предлагаемого метода является возможность ввода данных с бланков с помощью любого текстового редактора, в т. ч. и на компьютерах с низкими аппаратными и программными возможностями. Однако перед обработкой введенных данных их необходимо преобразовать с помощью дополнительной программы в форму, используемую пакетом обработки. Эта же программа может и обнаруживать ошибки ввода еще на предварительной стадии, что повышает надежность получаемых результатов.

Было проведено несколько серий обработки результатов измерений, которые показали высокую эффективность данной методики ввода.

Литература

1. Верник А.Н., Угаров В.В., Чунаева М.С., Эткин А.Е. *Исследование параметров помехоустойчивости канала связи в системе измерения временной характеристики в процессе отключения коммутационной аппаратуры*. N гос. регистрации: 01.84.0022513. Инвентарный номер: 02.8.90.015921. УДК 681.327.8: 621.391.823. 1984 г. х/д 29-84-1.
2. Сальман Л.А. *Создание программного обеспечения для решения задач оптимизации* // Зарубежная радиоэлектроника, М.: "Радио и связь", 1988, вып. 9. - с. 54-61.

Ю. В. Цыганова.

Разработка программных средств моделирования и исследования стохастических систем оптимальной дискретной фильтрации

Введение

Во многих приложениях, в том числе при оценке элементов движения объектов, перед разработчиками возникает задача построения устойчивой вычислительной схемы фильтрации Калмана, поскольку последовательные вычисления в соответствии со стандартным фильтром могут привести к потере ковариационной матрицы ошибок свойства положительной определенности. К настоящему моменту для преодоления этой проблемы накоплен достаточно большой набор вычислительных методов оптимальной дискретной фильтрации. Все эти методы используют различные способы декомпозиции, факторизации и триангуляризации матриц, участвующих в уравнениях фильтра, что значительно улучшает их вычислительные качества.

В работе рассматриваются две формы дискретного фильтра: ковариационная и информационная. Для сравнительного изучения взяты шесть ковариационных и три информационных алгоритма: стандартные ковариационный и информационный алгоритмы Калмана, стабилизированные ковариационный и информационный алгоритмы, квадратно-корневые ковариационный и информационный алгоритмы Поттера, квадратно-корневой алгоритм Карлсона, алгоритм Бирмана и алгоритм Бар-Ицхака-Медана, работающий с редуцированным фильтром.

Для проведения сравнительного исследования алгоритмов оптимальной дискретной фильтрации разработана интегрированная среда, позволяющая моделировать стохастические системы дискретного времени. Данная программа дает возможность проводить расчеты, используя различные модификации фильтра Калмана, а также сравнивать их эффективность применительно к конкретной задаче.

Постановка задачи

Рассмотрим модель динамической системы в дискретном времени, которая описывается следующими уравнениями состояния объекта и измерителя:

$$x(t+1) = \Phi(t+1, t)x(t) + G_d(t)w_d(t) \quad (1)$$

$$z(t) = H(t)x(t) + v(t) \quad (2)$$

где $x(t)$, $z(t)$, $w_d(t)$, соответственно, векторы в R^n , R^m и R^s . Шумы $w_d(t)$ и $v(t)$ образуют гауссовы белые последовательности с нулевыми средними и матрицами ковариаций, соответственно, $Q_d(t)$ и $R(t)$. Матрица $\Phi(t+1, t)$ — переходная матрица состояния от момента времени t к моменту $t+1$. Вектор $x(t_0)$ — начальное значение вектора состояния с математическим ожиданием $\bar{x}_0$ и матрицей ковариации P_0 .

Для построения вычислительной схемы фильтрации и нахождения оптимальной оценки вектора состояния x для заданной дискретной модели объекта применяются алгоритмы, основанные на стандартном фильтре Калмана.

Вычислительные методы оптимальной дискретной фильтрации, основанные на стандартной форме фильтра Калмана.

В случае ковариационной формы фильтра векторное m -мерное измерение можно представить как последовательность m скалярных измерений. Когда матрица R диагональна, компоненты вектора z могут рассматриваться как независимые измерения, а вычисления производиться последовательно от скаляра к скаляру. Если R не диагональна, тогда можно использовать следующую процедуру. Вычисляют разложение по Холецкому $R = LL^T$ и решают, используя обратную подстановку, относительно z^* и H^* (L — нижнетреугольная матрица) уравнения $Lz^* = z$, $LH^* = H$.

Это преобразование приводит к новому шумовому процессу с единичной ковариационной матрицей. Поэтому расчет может вестись последовательно по компонентам z^* . Заметим, что треугольная форма матрицы L такова, что z_i^* (i -й компонент вектора z^*) является просто линейной функцией i компонент вектора z . По сравнению с методами, использующими векторные измерения, приведенная процедура преобразования представляется более

привлекательной как более эффективная в вычислительном отношении и менее чувствительная к плохо обусловленным измерениям.

Учитывая соотношения двойственности этапов обработки измерения и экстраполяции в ковариационной и информационной формах фильтра, на этапе экстраполяции вычисления могут производиться последовательно s раз, если матрица Q_d диагональная. Если это не так, то можно также использовать разложение Холецкого, т.е. представить Q_d в виде LL^T , а затем сделать следующую замену: $Q_d := I$; $G_d := G_d L$, так как матрицы G_d и Q_d определяются соотношением $G_d Q_d G_d^T = G_d L L^T G_d^T$.

Методы, использующие ковариационную форму фильтра.

1. Стандартный алгоритм Калмана.

Стандартный алгоритм Калмана записывается следующим образом:

1) Этап экстраполяции

а) Оценка: $\tilde{x} := \Phi \hat{x}$; $\hat{x}(t_0) := \bar{x}_0$

б) Ковариация ошибки этой оценки: $\tilde{P} := \Phi \tilde{P} \Phi^T + G_d Q_d G_d^T$

2) Этап обработки измерения

(m -кратное повторение процедуры "скалярного" обновления):

$\alpha := h^T \tilde{P} h + r$; $K := \tilde{P} h / \alpha$; $\tilde{P} := \tilde{P} - K h \tilde{P}$;

$\tilde{x} := \tilde{x} + K(z - h^T \tilde{x})$,

в котором h — i -й столбец матрицы H^T ; $r = r_i$; z — i -й элемент вектора z ($i = 1, 2, \dots, m$).

2. Стабилизированный алгоритм.

Этап экстраполяции стабилизированного алгоритма в точности совпадает с этапом экстраполяции стандартного алгоритма Калмана. Для этапа обработки измерения Джозеф [2] предложил использовать общую формулу, справедливую для матрицы P при любом, не обязательно оптимальном K :

$$\tilde{P} = (I - K H) \tilde{P} (I - K H)^T + K R K^T.$$

При оптимальном $K = \tilde{P} H^T (H \tilde{P} H^T + R)^{-1}$ она превращается в $\tilde{P} = \tilde{P} - K H \tilde{P}$.

При таких вычислениях результирующая матрица сохраняет симметричность; кроме того, исчезает опасная для потери положительной определенности операция вычитания матриц, однако сложность вычислений возрастает почти вдвое.

3. Квадратно-корневой алгоритм Поттера.

Вместо матриц $\tilde{P}$ и $\tilde{P}$, по своей природе положительно определенных, оперируют с их квадратными корнями $\tilde{S}$ и $\tilde{S}$, отвечающими равенствам $\tilde{S} \tilde{S}^T = \tilde{P}$, $\tilde{S} \tilde{S}^T = \tilde{P}$. Конечно, эти соотношения не дают однозначного определения квадратных корней. Однако отсутствие в общем случае однозначности не вызывает беспокойства, поскольку эти корни однозначно определяются по разложению Холецкого (например, LL^T -разложение).

Основная идея метода фильтрации с использованием квадратного корня состоит в замене уравнений алгоритма Калмана на аналогичные, предназначенные для последовательного расчета $\tilde{S}$ и $\tilde{S}$. Такой подход оправдывается тем, что произведение $\tilde{S} \tilde{S}^T$ никогда не может быть неопределенным даже с учетом ошибок округления, в то время как ошибки округления иногда делают неопределенным вычисленное значение матрицы P .

На этапе экстраполяции, с учетом разложения матрицы $\tilde{P}$, уравнение для нее принимает следующий вид: $\tilde{S} \tilde{S}^T = \Phi \tilde{S} \tilde{S}^T \Phi^T + G_d Q_d G_d^T$. Непосредственно вычислить матрицу $\tilde{S}$ можно путем построения ортогональной матрицы T размерности $(n+s) \times (n+s)$, такой что

$$T \begin{bmatrix} \tilde{S}^T \Phi^T \\ U^T G_d^T \end{bmatrix} = \begin{bmatrix} \tilde{S}^T \\ 0 \end{bmatrix},$$

$Q_d = U U^T$, что обеспечивает, например, модифицированный алгоритм Грама-Шмидта. В результате после этапа экстраполяции матрица $\tilde{S}$ всегда будет получаться треугольной.

На этапе обработки измерения уравнение $\tilde{P} := \tilde{P} - K h \tilde{P}$ можно переписать в виде $\tilde{S} \tilde{S}^T := \tilde{S} (I - f f^T / \alpha) \tilde{S}^T$. Потребуем так выбрать число β , чтобы обеспечить тождественность: $I - f f^T / \alpha = (I - \beta f f^T) (I - \beta f f^T)$, где $\dim I = n$.

Из получающегося отсюда квадратного уравнения, с учетом $\alpha = f^T f + r$, выбираем одно решение $\beta = (1/\alpha) / (1 + \sqrt{r/\alpha})$, защищенное от операции вычитания положительной величины $\sqrt{r/\alpha}$ в знаменателе.

Вычисление S равносильно счету с двойной точностью P в стандартном алгоритме Калмана. Кроме того, устранена опасность утраты матрицей P свойства положительной определенности, что вело бы к расходимости оценок вектора состояния. Недостатком данного алгоритма является наличие операций извлечения квадратного корня.

4. Алгоритм Бирмана.

Основная идея алгоритма состоит в разложении ковариационной матрицы P в произведение двух треугольных матриц и диагональной матрицы. Можно рассматривать два варианта алгоритма Бирмана: LD -алгоритм и UD -алгоритм Бирмана. В данной работе рассматривается только LD -алгоритм. В LD -алгоритме используется разложение матрицы $P = LDL^T$, где L — нижнетреугольная матрица с единичной диагональю, D — диагональная матрица.

На этапе экстраполяции, с учетом разложения матрицы $\tilde{P}$, уравнение для нее принимает следующий вид:

$$\tilde{P} = \tilde{L}\tilde{D}\tilde{L}^T - \Phi\tilde{L}\tilde{D}\tilde{L}^T\Phi^T + G_dQ_dG_d^T.$$

Факторы $\tilde{L}$ и $\tilde{D}$ вычисляются путем применения на этапе экстраполяции Модифицированного взвешенного алгоритма Грама-Шмидта. Однако для применения этого алгоритма необходимо, чтобы матрица Q_d была диагональной. Тогда матрица $\tilde{P}$ может быть представлена в следующем виде: $\tilde{P} = [\Phi\tilde{L} \ G_d] \text{Diag}(\tilde{D}, Q_d) [\Phi\tilde{L} \ G_d]^T$, т.е. $\tilde{P} = WDW^T$, где $W = [\Phi\tilde{L} \ G_d] = \begin{bmatrix} w_1^T \\ \dots \\ w_n^T \end{bmatrix}$, $\dim W = n \times (n+s)$ и $D = \text{Diag}(\tilde{D}, Q_d) = \text{Diag}(D_1, \dots, D_N)$, $N = n+s$, что и требуется в алгоритме Грама-Шмидта. Поэтому будем считать, что матрица Q_d — диагональная. (Если это не так, тогда применив к ней разложение Холецкого, т.е. представив Q_d в виде $Q_d = LDL^T$, матрицы Q_d и G_d можно заменить на D и G_dL , т.к. $G_dQ_dG_d^T = (G_dL)D(G_dL)^T$).

На этапе обработки измерения используются разложения: $\tilde{P} = \tilde{L}\tilde{D}\tilde{L}^T$, $\tilde{P} = \tilde{L}\tilde{D}\tilde{L}^T$ где L — нижнетреугольные с единичной диагональю матрицы, D — диагональные матрицы.

Данный алгоритм не содержит операции извлечения квадратного корня, а работа с треугольными матрицами требует меньшего числа арифметических операций по сравнению с обычными.

5. Квадратно-корневой алгоритм Карлсона.

Этап экстраполяции для данного алгоритма в точности совпадает с соответствующим этапом в алгоритме Поттера. Этап обработки измерения отличается от соответствующего этапа в алгоритме Поттера тем, что последовательность вычислений получается на основании этапа обработки измерения алгоритма Бирмана следующим образом. В качестве $\tilde{S}$ берется произведение $\tilde{L}\tilde{D}^{1/2}$ и учитывая это, модифицируются имеющиеся уравнения для алгоритма Бирмана.

6. Алгоритм Бар-Ицхака-Медана.

Во многих приложениях выбор измерительных средств таков, что в измерение z попадает лишь часть (заранее известная) элементов оцениваемого вектора x . Пусть эта часть — первые q элементов в каждой строке матрицы H : $H = [H^{mq}|0]$ (1), где H^{mq} — ненулевая подматрица размеров $m \times q$; 0 — нулевая подматрица размеров $m \times s$; $s = n - q$, $n > 1$, $1 \leq q < n$. Это означает, что оцениваемый вектор x разделяется на две части: $x = (x^q|x^s)^T$, где первая часть x^q размерности q попадает в вектор измерений z , а вторая часть x^s — нет. Соответственно матрицы $\tilde{P}$ и $\hat{P}$ рассматривают поблочно: $P = \begin{bmatrix} P^{qq} & (P^{sq})^T \\ P^{sq} & P^{ss} \end{bmatrix}$ (2) (верхние двойные индексы, как и в (1), обозначают размеры подматриц).

При свойстве (1) стандартный алгоритм Калмана распадается на независимый редуцированный фильтр порядка q для измеряемых компонент оцениваемого вектора x :

$$(3) \quad K^{qm} = \tilde{P}^{qq}(H^{mq})^T[H^{mq}\tilde{P}^{qq}(H^{mq})^T + R]^{-1},$$

$$(4) \quad \tilde{P}^{qq} = \tilde{P}^{qq} - K^{qm}H^{mq}\tilde{P}^{qq},$$

$$(5) \quad \hat{x}^q = \tilde{x}^q + K^{qm}(z - H^{mq}\tilde{x}^q)$$

и фильтр порядка $s = n - q$, зависящий от предыдущего фильтра, для неизмеряемых компонент вектора x (Bar-Itzhack, 1980):

$$(6) \quad K^{sq} = \tilde{P}^{sq}(\tilde{P}^{qq})^{-1},$$

$$(7) \quad \tilde{P}^{sq} = K^{sq}\tilde{P}^{qq},$$

$$(8) \quad \tilde{P}^{ss} = \tilde{P}^{ss} - K^{sq}(\tilde{P}^{sq} - \tilde{P}^{sq})(K^{sq})^T,$$

$$(9) \quad \hat{x}^s = \tilde{x}^s + K^{sq}(\hat{x}^q - \tilde{x}^q)$$

Полученный алгоритм лишь выделяет редуцированный фильтр, но для него задача фильтрации остается существенной. Для ее решения делают переход к LD -факторизованному алгоритму.

Пусть при свойстве (1) в стандартном алгоритме Калмана используются разложения: $\tilde{P} = \tilde{L}\tilde{D}\tilde{L}^T$, $\tilde{P} = \tilde{L}\tilde{D}\tilde{L}^T$, где L —

$\begin{bmatrix} L^{qq} & 0 \\ L^{sq} & L^{ss} \end{bmatrix}, \quad D = \begin{bmatrix} D^q & 0 \\ 0 & D^s \end{bmatrix}, \quad (10)$ (верхние индексы — размеры матричных блоков); L — нижнетреугольная с единичной диагональю матрицы, D — диагональные матрицы.

Тогда этап обработки измерения в стандартном алгоритме Калмана эквивалентен следующему алгоритму (Bar-Itzhack, Medan, 1983):

- 1) Выполнить LD -алгоритм Бирмана вместо редуцированного фильтра порядка q (q вместо n , т.е. по данным $\tilde{L}^{qq}, \tilde{D}^q, \tilde{x}^q$ найти $\tilde{L}^{sq}, \tilde{D}^s, \tilde{x}^s$).
- 2) Вычислять:
 - (11) $K^{sq} = \tilde{L}^{sq}(\tilde{L}^{qq})^{-1}$,
 - (12) $\tilde{L}^{sq} = K^{sq}\tilde{L}^{qq}$,
 - (13) $\tilde{L}^{ss} = \tilde{L}^{ss}, \quad \tilde{D}^s = \tilde{D}^s$,
 - (14) $\tilde{x}^s = \tilde{x}^s + K^{sq}(\tilde{x}^q - \tilde{x}^q)$.

Данный алгоритм очень привлекателен своей простотой, но применять его следует, учитывая тот факт, что выбор LD -факторизации здесь не произволен, а определен стремлением обеспечить независимость редуцированного фильтра правым, а не левым размещением нулей в (1). Существенный выигрыш в объеме вычислений у данного алгоритма по сравнению с другими ковариационными алгоритмами происходит при q значительно меньшем, чем n .

Методы, использующие информационную форму фильтра.

1. Стандартный информационный алгоритм.

Стандартный информационный алгоритм при последовательных вычислениях (на этапе экстраполяции) представим следующим образом:

- 1) Этап экстраполяции (s -кратное повторение процедуры)

$$M := \Phi^{-T} \hat{\Lambda} \Phi^{-1}; \quad \alpha := g_d^T M g_d + 1/q_d; \quad N := M g_d / \alpha;$$

$$\tilde{\Lambda} := M - N g_d^T M; \quad \tilde{d} := \Phi^{-T} \hat{d} - N g_d^T \Phi^{-T} \hat{d},$$
 где g_d — i -й столбец матрицы G_d , $q_d = q_{di}$, $i = 1, \dots, s$, а $\tilde{x}$ представляет собой решение уравнения $\tilde{\Lambda} \tilde{x} = \tilde{d}$. (После первых s вычислений Φ заменяется на I).

- 2) Этап обработки измерения

$$\hat{d} := \tilde{d} + H^T R^{-1} z; \quad \hat{\Lambda} := \tilde{\Lambda} + H^T R^{-1} H,$$
 где $\hat{x}$ представляет собой решение уравнения $\hat{\Lambda} \hat{x} = \hat{d}$.

2. Стабилизированный информационный алгоритм.

Этап обработки измерения в данном алгоритме совпадает с соответствующим этапом предыдущего алгоритма. На этапе экстраполяции, как и в ковариационном стабилизированном алгоритме, уравнение $\tilde{\Lambda} = M - N G_d^T M$ заменяется уравнением

$$\tilde{\Lambda} = (I - N G_d^T) M (I - N G_d^T)^T + N Q_d^{-1} N^T.$$

3. Квадратно-корневой алгоритм Поттера.

Точно также, как и в алгоритме Поттера для ковариационной формы фильтра, вместо матриц $\tilde{\Lambda}$ и $\hat{\Lambda}$ оперируют с их квадратными корнями $\tilde{R}$ и $\hat{R}$, отвечающими равенствам $\tilde{R}^T \tilde{R} = \tilde{\Lambda}$, $\hat{R}^T \hat{R} = \hat{\Lambda}$.

На этапе обработки измерения применяется алгоритм Хаусхолдера для построения ортогональной матрицы T такой, что $T \begin{bmatrix} \tilde{R}^T \\ V^T H \end{bmatrix} = \begin{bmatrix} \hat{R} \\ 0 \end{bmatrix}$, где $R^{-1} = V V^T$.

Возможности интегрированной среды для сравнительного исследования различных вычислительных методов оптимальной дискретной фильтрации.

Для изучения и сравнения эффективности алгоритмов оптимальной дискретной фильтрации разработана интегрированная среда. Данный программный продукт обладает следующими возможностями:

Программа позволяет вводить параметры задачи в дискретной форме (объект плюс измеритель) любой размерности, сохранять и редактировать данные, проводить расчеты по заданным алгоритмам, сохранять и просматривать результаты расчетов в форме отчетов и графиков.

Реализована возможность компьютерного моделирования поведения объекта, измерителя, а также шумов на объекте и измерителе. В программе обеспечена возможность сравнения результатов работы различных алгоритмов.

Данная интегрированная среда написана на языке высокого уровня с использованием технологии ООП, обеспечивающей ее

дальнейшую модификацию и расширение. Программа обладает удобным пользовательским интерфейсом.

При реализации алгоритмов, в особенности информационных, возникает необходимость в обращении матрицы. Возможная большая размерность решаемой задачи предъявляет высокие требования к эффективности данной операции. Метод, реализованный в данной программе и предназначенный для обращения матрицы, основан на UL -разложении исходной матрицы (U — верхняя треугольная матрица с единичной диагональю, а L — нижняя треугольная матрица) с выбором ведущего элемента по строке. Для повышения эффективности алгоритма при выборе ведущего элемента использовался вектор перестановок.

Так как рассматриваемые вычислительные методы оптимальной дискретной фильтрации (за исключением стандартных и стабилизированных алгоритмов) используют методы декомпозиции, факторизации и триангуляризации матриц, то эти требуемые методы также реализованы в данном программном продукте.

Программа содержит функцию для генерации случайных величин, распределение которых подчиняется нормальному закону. Параметрами для данной функции являются математическое ожидание и среднее квадратичное отклонение. С помощью данной функции в программе моделируются шумы в объекте и измерителе с пуассоновскими математическими ожиданиями и заданными матрицами ковариаций Q_d и R . Результаты моделирования нормально распределенных случайных величин были протестированы.

В программе также имеются средства для построения графиков выбранных сигналов по графическим данным, полученным в результате работы одного из реализованных алгоритмов.

Заключение

Дан краткий обзор современных вычислительных методов оптимальной дискретной фильтрации. Для сравнительного исследования были выбраны шесть алгоритмов в ковариационной форме и три алгоритма в информационной форме.

Разработана интегрированная среда для изучения рассмотренных вычислительных методов оптимальной дискретной фильтрации. Она обладает средствами для ввода и редактирования задач, проведения расчетов с использованием любого из предложенных алгоритмов, создания файлов с отчетом о проведенных расчетах, построение графиков по накопленным данным.

Программа обладает удобным пользовательским интерфейсом. Ее преимуществом является также и то, что она позволяет решать задачи любой размерности, так как все матрицы, участвующие в расчетах, являются динамическими. Применение при разработке программы методов ООП обеспечивает возможность ее дальнейшего расширения.

Литература

- 1) Каминский П.Г., Брайсон А.Е., Шмидт С.Ф. Обзор современных методов дискретной фильтрации, использующих квадратные корни матриц. — Зарубежная радиоэлектроника, 1973, #6, с. 37-53.
- 2) Bierman G.J. Factorization methods for discrete sequential estimation. — N.-Y.: Acad. Press, 1977.
- 3) Семин И.В. Эффективные алгоритмы обновления оценок по измерениям. — Судостроительная промышленность, 1991, вып. 27, с. 55-62.

Г. Ю. Куликов, Н. А. Андрианова.

Об оптимальном методе Эйлера-простых итераций для решения дифференциально-алгебраических уравнений

Рассмотрим систему дифференциально-алгебраических уравнений вида

$$x'(t) = g(x(t), y(t)), \quad (1a)$$

$$y(t) = f(x(t), y(t)), \quad (1б)$$

$$x(0) = x^0, \quad y(0) = y^0, \quad (1в)$$

где $t \in [0, T]$, $x(t) \in \mathbb{R}^m$, $y(t) \in \mathbb{R}^n$, $g: D \subset \mathbb{R}^{m+n} \rightarrow \mathbb{R}^m$, $f: D \subset \mathbb{R}^{m+n} \rightarrow \mathbb{R}^n$, и начальные условия (1в) заданы корректно, т.е., $y^0 = f(x^0, y^0)$. Предположим, что на компактном множестве D_1 задача (1) удовлетворяет условиям.

I. Условие гладкости. Отображение $G: D_1 \subset \mathbb{R}^{m+n} \rightarrow \mathbb{R}^{m+n}$ дифференцируемо по Фреше на множестве D_1 и

$$\|\partial G(z') - \partial G(z'')\| \leq \gamma \|z' - z''\| \quad \forall z', z'' \in D_1,$$

где γ — некоторая константа.

II. Условие ограниченности.¹

$$\|\partial f(z)\| \leq d < 1 \quad \forall z \in D_1.$$

III. Условие включения². Существует выпуклое множество D_0 такое, что $z^0 \in D_0 \subset D_1$.

Известно [1], что в этом случае задача (1) имеет единственное решение $z(t) \in D_0$.

В [2] для решения задачи (1) предложен комбинированный метод Эйлера-простых итераций (Эпи)

$$x_{k+1}^i = x_k + \tau g(x_{k+1}^{i-1}, y_{k+1}^{i-1}), \quad (2a)$$

¹ Здесь и далее норма понимается в смысле $\|z\| = \max_{1 \leq i \leq m+n} |z_i|$

² Включение $D_0 \subset D_1$ означает, что множество D_0 содержится в D_1 вместе с некоторой окрестностью

$$y_{k+1}^i = f(x_{k+1}^{i-1}, y_{k+1}^{i-1}), \quad (2б)$$

$$x_0 = x^0, \quad y_0 = y^0, \quad (2в)$$

$i = 1, 2, \dots, N$, $k = 0, 1, \dots, K-1$. Там же доказано, что при выполнении условий I–III приближенное решение, полученное с помощью метода (2), сходится к точному решению задачи (1) при $\tau \rightarrow 0$ и $N \rightarrow \infty$ со скоростью $O(d^N + \tau)$.

Условие II накладывает слишком жесткое ограничение на функцию $f(x, y)$. Попытаемся его ослабить, расширяя тем самым класс задач, решаемых Эпи-методом.

Нетрудно видеть, что задача (1) эквивалентна

$$x'(t) = g(x(t), y(t)), \quad (3a)$$

$$y(t) = y(t) - \Lambda(t)h(x(t), y(t)), \quad (3б)$$

$$x(0) = x^0, \quad y(0) = y^0, \quad (3в)$$

где

$h(x(t), y(t)) = y(t) - f(x(t), y(t))$, $\Lambda(t) = \text{diag}(\lambda_1(t), \lambda_2(t), \dots, \lambda_n(t))$ — диагональная матрица размера n и $\lambda_i(t) \neq 0$, $i = 1, 2, \dots, n$ при $t \in [0, T]$. Таким образом, мы получили однопараметрическое семейство задач (3) со свободным параметром $\Lambda(t)$ эквивалентных задаче (1). Поэтому, применяя для решения (3) метод (2), мы получаем однопараметрическое семейство Эпи-методов для решения задачи (1)

$$x_{k+1}^i = x_k + \tau g(x_{k+1}^{i-1}, y_{k+1}^{i-1}), \quad (4a)$$

$$y_{k+1}^i = y_{k+1}^{i-1} - \Lambda_{k+1} h(x_{k+1}^{i-1}, y_{k+1}^{i-1}), \quad (4б)$$

$$x_0 = x^0, \quad y_0 = y^0, \quad (4в)$$

$i = 1, 2, \dots, N$, $k = 0, 1, \dots, K-1$. В дальнейшем матрицу Λ будем называть параметром Эпи-метода. Отметим, что при $\Lambda \equiv E$, где E — единичная матрица размера n , метод (4) эквивалентен методу (2).

Итак, задача о выборе оптимального параметра в Эпи-методе заключается в том, чтобы найти матрицу $\Lambda(t)$, минимизирующую константу d в условии ограниченности и, следовательно, расширяющую область применения Эпи-метода. Метод (4) с оптимальным параметром будем называть оптимальным Эпи-методом.

Следующие две теоремы доказывают существование оптимального Эпи-метода.

Теорема 1. Пусть задача (1) удовлетворяет на множестве D_1 условию диагонального преобладания

$$|\partial h(z)_{i,m+i}| > \sum_{j=1, j \neq m+i}^{m+n} |\partial h(z)_{i,j}|, \quad \forall z \in D_1, \quad (5)$$

$i = 1, 2, \dots, n$. Тогда существует такая матрица $\Lambda(t)$, что для задачи (4) будет выполнено условие ограниченности на множестве D_1 .

Доказательство. Из условия диагонального преобладания и определения функции $h(x, y)$ следует

$$\left| 1 - \frac{\partial f_i(z)}{\partial y_i} \right| > \sum_{j=1, j \neq m+i}^{m+n} \left| \frac{\partial f_i(z)}{\partial z_j} \right|, \quad \forall z \in D_1, \quad (6)$$

$i = 1, 2, \dots, n$. Введем функцию $\epsilon = \epsilon(z) : D_1 \subset \mathbb{R}^{m+n} \rightarrow \mathbb{R}^n$

$$\epsilon_i(z) = \left| 1 - \frac{\partial f_i(z)}{\partial y_i} \right| - \sum_{j=1, j \neq m+i}^{m+n} \left| \frac{\partial f_i(z)}{\partial z_j} \right|, \quad (7)$$

$i = 1, 2, \dots, n$. Тогда из (6) и (7) следует, что $\epsilon_i(z) > 0$, $i = 1, 2, \dots, n$, на множестве D_1 .

Якобиан правой части задачи (3) имеет вид

$$\begin{pmatrix} -\lambda_1 \frac{\partial f_1(z)}{\partial x_1} & \dots & -\lambda_1 \frac{\partial f_1(z)}{\partial x_m} & 1 - \lambda_1 \left(1 - \frac{\partial f_1(z)}{\partial y_1} \right) & \dots & -\lambda_1 \frac{\partial f_1(z)}{\partial y_n} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ -\lambda_n \frac{\partial f_n(z)}{\partial x_1} & \dots & -\lambda_n \frac{\partial f_n(z)}{\partial x_m} & -\lambda_n \frac{\partial f_n(z)}{\partial y_1} & \dots & 1 - \lambda_n \left(1 - \frac{\partial f_n(z)}{\partial y_n} \right) \end{pmatrix}$$

Тогда для нормы этой матрицы справедливо

$$\|\partial(y - \Lambda f(z))\| = \max_{1 \leq i \leq m+n} \left(\left| 1 - \lambda_i \left(1 - \frac{\partial f_i(z)}{\partial y_i} \right) \right| + |\lambda_i| \sum_{j=1, j \neq m+i}^{m+n} \left| \frac{\partial f_i(z)}{\partial z_j} \right| \right). \quad (8)$$

Рассмотрим функцию $F = F(\Lambda, z) : D_1 \subset \mathbb{R}^{m+2n} \rightarrow \mathbb{R}^n$

$$F_i(\lambda_i, z) = \left| 1 - \lambda_i \left(1 - \frac{\partial f_i(z)}{\partial y_i} \right) \right| + |\lambda_i| \sum_{j=1, j \neq m+i}^{m+n} \left| \frac{\partial f_i(z)}{\partial z_j} \right|. \quad (9)$$

Из (7) и (9) следует

$$F_i(\lambda_i, z) = \left| 1 - \lambda_i \left(1 - \frac{\partial f_i(z)}{\partial y_i} \right) \right| + |\lambda_i| \left| 1 - \frac{\partial f_i(z)}{\partial y_i} \right| - |\lambda_i| \epsilon_i(z). \quad (10)$$

Зафиксируем произвольное $z \in D_1$ и будем рассматривать $F_i(\lambda_i, z)$ как функцию одной переменной λ_i , $i = 1, 2, \dots, n$. Нетрудно видеть, что справедлива оценка

$$\tilde{F}_i(\lambda_i, z) = \left| 1 - \lambda_i \left(1 - \frac{\partial f_i(z)}{\partial y_i} \right) \right| + |\lambda_i| \left| 1 - \frac{\partial f_i(z)}{\partial y_i} \right| \geq 1 \quad \forall \lambda_i \in \mathbb{R}. \quad (11)$$

В силу (6) величина $1 - \frac{\partial f_i(z)}{\partial y_i}$ имеет постоянный знак. Тогда $\tilde{F}_i(\lambda_i, z) \equiv 1$ на отрезке $\left[0, \left(1 - \frac{\partial f_i(z)}{\partial y_i} \right)^{-1} \right]$ если $1 - \frac{\partial f_i(z)}{\partial y_i} > 0$ или на отрезке $\left[-\left(1 - \frac{\partial f_i(z)}{\partial y_i} \right)^{-1}, 0 \right]$ если $1 - \frac{\partial f_i(z)}{\partial y_i} < 0$.

В силу строгой положительности величины $|\lambda_i| \epsilon_i(z)$ функция $F_i(\lambda_i, z) < 1$ на одном из указанных выше отрезков для λ_i . Отсюда следует, что для задачи (3) выполнено условие ограниченности. Более того, из компактности отрезка и непрерывности функции $F_i(\lambda_i, z)$ мы получаем существование оптимального параметра $\Lambda(t)$. Теорема доказана.

Справедливо и обратное утверждение

Теорема 2. Пусть существует диагональная матрица $\Lambda(t)$ такая, что для задачи (3) на D_1 выполняется условие ограниченности. Тогда на D_1 для задачи (1) выполняется условие диагонального преобладания (5).

Доказательство. Из (10) и (11) следует, что $|\lambda_i| \epsilon_i(z) > 0$. Тогда (6) и (7) доказывают теорему 2.

Итак, мы доказали, что из условия диагонального преобладания следует существование оптимального параметра. При этом легко видеть, что условие диагонального преобладания действительно является более слабым, чем условие ограниченности (второе следует из первого). Теперь мы докажем единственность оптимального параметра.

Теорема 3. Пусть задача (1) удовлетворяет на множестве D_1 условию диагонального преобладания (5). Тогда существует единственный оптимальный Эпи-метод с матрицей Λ_{k+1} , где $\lambda_i = (1 - \partial f(z_k)_{i,m+i})^{-1}$.

Доказательство. В теореме 1 мы показали, что $\tilde{F}_i(\lambda_i, z) \equiv 1$ на отрезке $\left[0, \left(1 - \frac{\partial f_i(z)}{\partial y_i}\right)^{-1}\right]$ если $1 - \frac{\partial f_i(z)}{\partial y_i} > 0$ или на отрезке $\left[-\left(1 - \frac{\partial f_i(z)}{\partial y_i}\right)^{-1}, 0\right]$ если $1 - \frac{\partial f_i(z)}{\partial y_i} < 0$. Поэтому в силу линейности слагаемого $|\lambda_i|c_i(z)$ относительно λ_i в (10) и условия диагонального преобладания для оптимального параметра мы имеем $\lambda_i - \left(1 - \frac{\partial f_i(z)}{\partial y_i}\right)^{-1}$. Теорема доказана.

Таким образом, теорема 3 позволяет находить оптимальный Эпи-метод для задачи (1). Причем, для этого мы должны вычислять только диагональные элементы матрицы $\partial f(z)$. Более того, если матрица $\partial f(z)$ известна полностью, то, вычисляя константу d для каждой точке сетки, мы можем контролировать сходимость оптимального Эпи-метода.

Литература

1. Куликов Г.Ю. О численном решении автономной задачи Коши с алгебраической связью на фазовые переменные (невырожденный случай) // Вестн. МГУ. Сер. матем., механ. 1993. № 3. С. 10-14.
2. Куликов Г.Ю. О численном решении автономной задачи Коши с алгебраической связью на фазовые переменные // Ж. вычисл. матем. и матем. физ. 1993. Т. 33. № 4. С. 522-540.

Г. Ю. Куликов, А. А. Корнева.

Об эффективной реализации итеративных методов Рунге-Кутты для дифференциально-алгебраических уравнений индекса 1

§ 1. Введение

Рассмотрим систему дифференциально-алгебраических уравнений вида

$$x'(t) = g(x(t), y(t)), \quad (1.1a)$$

$$y(t) = f(x(t), y(t)), \quad (1.1б)$$

$$x(0) = x^0, \quad y(0) = y^0, \quad (1.1в)$$

где $t \in [0, T]$, $x(t) \in \mathbb{R}^m$, $y(t) \in \mathbb{R}^n$, $g : D \subset \mathbb{R}^{m+n} \rightarrow \mathbb{R}^m$, $f : D \subset \mathbb{R}^{m+n} \rightarrow \mathbb{R}^n$, и начальные условия (1.1в) заданы корректно, т.е., $y^0 = f(x^0, y^0)$.

В [1], [2] для решения задачи (1.1) предложены два класса комбинированных численных методов с постоянным и переменным шагом интегрирования. Во-первых, применяя к задаче (1.1) l -стадийный метод Рунге-Кутты (РК), мы получаем следующую дискретную задачу:

$$x_{ki} = x_k + \tau_k \sum_{j=1}^l a_{ij} g(x_{kj}, y_{kj}), \quad (1.2a)$$

$$y_{ki} = f(x_{ki}, y_{ki}), \quad i = 1, 2, \dots, l, \quad (1.2б)$$

$$x_{k+1} = x_k + \tau_k \sum_{i=1}^l b_i g(x_{ki}, y_{ki}), \quad (1.2в)$$

$$y_{k+1} = f(x_{k+1}, y_{k+1}), \quad k = 0, 1, \dots, K-1, \quad (1.2г)$$

$$x_0 = x^0, \quad y_0 = y^0, \quad (1.2д)$$

где a_{ij} , b_i ($i, j = 1, 2, \dots, l$) — коэффициенты РК формулы.

Обозначим через $z(t)$ вектор, составленный из компонент векторов $x(t)$ и $y(t)$ ($z(t) = (x(t), y(t))^T \in \mathbb{R}^{m+n}$), а через G отображение, полученное объединением отображений g и f ($G = (g, f)^T : D \subset \mathbb{R}^{m+n} \rightarrow \mathbb{R}^{m+n}$). Пусть $z(t_k)$ — значение точного решения

задачи (1.1) в точке t_k , $\bar{z}_k$ — значение точного решения задачи (1.2) в точке t_k и $\bar{z}_k = \bar{z}_k(N)$ — значение приближенного решения задачи (1.2) в точке t_k , полученное после N итераций некоторого итерационного метода.

Дополнительно введем вектор

$$Z_{k+1} = (z_{k1}, \dots, z_{kl}, z_{k+1})^T \in \mathbb{R}^{(m+n)(l+1)}$$

и определим отображение

$$\bar{G}_k^T: D \subset \mathbb{R}^{(m+n)(l+1)} \rightarrow \mathbb{R}^{(m+n)(l+1)},$$

$k = 0, 1, \dots, K-1$, следующим образом:

$$\bar{G}_k^T Z_{k+1} = \left(\bar{x}_k + \tau \sum_{j=1}^l a_{1j} g(z_{kj}), f(z_{k1}), \dots, \bar{x}_k + \tau \sum_{j=1}^l a_{lj} g(z_{kj}), f(z_{kl}), \right. \\ \left. \bar{x}_k + \tau \sum_{i=1}^l b_i g(z_{ki}), f(z_{k+i}) \right)^T.$$

Во-вторых, применяя к задаче (1.2) полный и модифицированный методы Ньютона, с использованием введенных выше обозначений получаем два класса комбинированных численных методов для решения задачи (1.1).

Метод Рунге-Кутты-Ньютона (РКН):

$$Z_{k+1}^i = Z_{k+1}^{i-1} - \partial \bar{F}_k^T (Z_{k+1}^{i-1})^{-1} \bar{F}_k^T Z_{k+1}^{i-1}, \quad (1.3a)$$

$$Z_{k+1}^0 = (\bar{z}_k, \dots, \bar{z}_k)^T \in \mathbb{R}^{(m+n)(l+1)}, \quad (1.3b)$$

$$\bar{z}_k = z_k^N, \quad k = 0, 1, \dots, K-1, \quad i = 1, \dots, N,$$

$$\bar{Z}_0 = Z^0 = (z^0, \dots, z^0)^T \in \mathbb{R}^{(m+n)(l+1)}, \quad (1.3c)$$

где $\bar{F}_k^T = E_{(m+n)(l+1)} - \bar{G}_k^T$ и $\partial \bar{F}_k^T (Z_{k+1}^{i-1})$ есть якобиан отображения $\bar{F}_k^T$, вычисленный в точке Z_{k+1}^{i-1} ($E_{(m+n)(l+1)}$ — единичная матрица размера $(m+n)(l+1)$).

Метод Рунге-Кутты-модифицированный Ньютона (РКМН):

$$Z_{k+1}^i = Z_{k+1}^{i-1} - \partial \bar{F}_k^T (\bar{Z}_k)^{-1} \bar{F}_k^T Z_{k+1}^{i-1}, \quad (1.4a)$$

$$Z_{k+1}^0 = (\bar{z}_k, \dots, \bar{z}_k)^T \in \mathbb{R}^{(m+n)(l+1)}, \quad (1.4b)$$

$$\bar{z}_k = z_k^N, \quad k = 0, 1, \dots, K-1, \quad i = 1, 2, \dots, N,$$

$$\bar{Z}_0 = Z^0 = (z^0, \dots, z^0)^T \in \mathbb{R}^{(m+n)(l+1)}. \quad (1.4c)$$

Предположим, что на компактном множестве D_1 задача (1.1) удовлетворяет условиям.

I. *Условие гладкости.* Отображение $G: D_1 \subset \mathbb{R}^{m+n} \rightarrow \mathbb{R}^{m+n}$ имеет на множестве D_1 непрерывные частные производные до порядка $s+2$ включительно, где s — порядок РК формулы, положенной в основу задачи (2.2).

II. *Условие невырожденности.* Матрица $E_n - \partial_y f(z)$ невырождена для любого $z \in D_1$.

III. *Условие включения*¹. Существует выпуклое множество D_0 такое, что $z^0 \in D_0 \subset D_1$.

Известно [3], что в этом случае задача (1.1) имеет единственное решение $z(t) \in D_0$.

В [2], [4] доказано, что при выполнении условий I–III и достаточном числе итераций N методы (1.3) и (1.4) имеют одинаковый порядок, совпадающий с порядком РК формулы, положенной в основу этих методов. Однако достаточное число итераций для каждого из методов свое. Так, РКН метод имеет максимальный порядок, если число итераций

$$N \geq \log_2(s+1). \quad (1.5)$$

Для РКМН метода мы имеем другую оценку:

$$N \geq s. \quad (1.6)$$

Таким образом, из (1.5) и (1.6) следует, что РКН и РКМН методы могут быть достаточно эффективными при использовании на практике. Однако, если размерность задачи (1.1) велика, то встает вопрос об оптимальном использовании алгоритмов (1.3) и (1.4). Тогда, принимая во внимание, что основные затраты машинного времени связаны с решением линейных систем вида:

$$\partial \bar{F}_k^T (Z_{k+1}^{i-1}) (Z_{k+1}^i - Z_{k+1}^{i-1}) = \bar{F}_k^T (Z_{k+1}^{i-1}), \quad (1.7)$$

нашей задачей является оптимизация решения задачи (1.7) по затратам машинного времени и оперативной памяти.

¹ Включение $D_0 \subset D_1$ означает, что множество D_0 содержится в D_1 вместе с некоторой окрестностью

§ 2. Оптимизация по времени

В [2] для решения (1.7) предложен модифицированный метод Гаусса, учитывающий структуру матрицы этой системы. Нетрудно видеть, что якобиан системы (1.2), используемый в качестве такой матрицы, имеет следующую блочную структуру:

$$\partial \bar{F}_k^T(Z) = \begin{pmatrix} \partial \bar{F}_k^T(Z)_1 \\ \partial \bar{F}_k^T(Z)_2 \\ \vdots \\ \partial \bar{F}_k^T(Z)_l \end{pmatrix},$$

где каждый блок $\partial \bar{F}_k^T(Z)_i$, $i = 1, 2, \dots, l$, представляет собой матрицу размера $(m+n) \times (m+n)l$ вида

$$\partial \bar{F}_k^T(Z)_i =$$

$$\left(\begin{array}{cccccc} O(\tau) & \dots & O(\tau) & 1+O(\tau) & \dots & O(\tau) & O(\tau) & \dots & O(\tau) & O(\tau) & \dots & O(\tau) \\ \vdots & & \vdots & \vdots & & \vdots & \vdots & & \vdots & \vdots & & \vdots \\ O(\tau) & \dots & O(\tau) & O(\tau) & \dots & 1+O(\tau) & O(\tau) & \dots & O(\tau) & O(\tau) & \dots & O(\tau) \\ 0 & \dots & 0 & x & \dots & x & x & \dots & x & 0 & \dots & 0 \\ \vdots & & \vdots & \vdots & & \vdots & \vdots & & \vdots & \vdots & & \vdots \\ 0 & \dots & 0 & x & \dots & x & x & \dots & x & 0 & \dots & 0 \end{array} \right) \left. \begin{array}{l} \\ \\ \\ \end{array} \right\} \begin{array}{l} m \\ \\ n \end{array}$$

$\underbrace{\hspace{10em}}_{(m+n) \times (i-1)} \quad \underbrace{\hspace{10em}}_m \quad \underbrace{\hspace{10em}}_n \quad \underbrace{\hspace{10em}}_{(m+n) \times (l-i)}$

Здесь x обозначает произвольный, в общем случае ненулевой, элемент. Кроме того, мы учли, что решение (1.7) распадается на решение двух независимых систем линейных уравнений размерности $(m+n)l \times (m+n)l$ и $n \times n$ [2]. Поэтому далее мы изучаем возможность эффективного решения большей системы.

Учет структуры якобиана системы (1.2) позволил значительно сократить число арифметических операций при решении системы (1.7), что привело к существенной экономии машинного времени. Однако можно заметить, что подход, изложенный в [2], будет более эффективным, если в системе (1.1) переменные x и y поменять местами. Такая перестановка переменных приведет к перестановке первых m строк в матрице $\partial \bar{F}_k^T(Z)_i$ с последними n строками. Тогда при прямом ходе метода Гаусса нулевые элементы подматриц $\partial \bar{F}_k^T(Z)_i$, $i = 1, 2, \dots, l$, заполнятся не будут, что положительно скажется на точности и скорости решения системы (1.7). Кроме того, такое преобразование позволит сохранить структуру якобиана при прямом ходе метода Гаусса, а, следовательно, возможность

сокращения числа операций и при обратном ходе. Единственным относительным недостатком будет потеря диагонального преобладания, которое давало выбор главного элемента на диагонали матрицы $\partial \bar{F}_k^T(Z)$. В данном случае главный элемент следует выбирать по всей активной подматрице. Это несколько замедляет работу алгоритма, но положительно скажется на точности, так как деление будет происходить на максимальный по модулю элемент, в то время как при учете диагонального преобладания деление происходит на диагональный элемент по модулю больший некоторого наперед заданного ϵ , что несколько хуже.

Итак, чтобы оценить эффективность модифицированного метода Гаусса с перестановкой переменных x и y , давайте подсчитаем общее число умножений и делений для этого метода при решении системы (1.7). Тогда сравнивая это число с аналогичными числами для обычного метода Гаусса и модифицированного метода, предложенного в [2], мы сможем оценить преимущество нового подхода.

Прежде всего подсчитаем число действий деления и умножения, необходимых для прямого метода Гаусса с учетом структуры матрицы $\partial \bar{F}_k^T(Z)$ и перестановки переменных x и y .

Число делений для приведения этой матрицы к верхнетреугольному виду равно

$$\sum_{j=1}^l \left(\sum_{i=1}^m (m+n-i) + \sum_{i=1}^m ((m+n)(l-j) + m-i) \right) = \frac{l(m+n)(ml+n-1)}{2} \quad (2.1)$$

Нетрудно подсчитать, что LU-разложение блока $\partial \bar{F}_k^T(Z)_j$, $j = 1, 2, \dots, l$, требует

$$\sum_{i=1}^n (m+n-i)(m(l-j+1)+n-i) + \sum_{i=1}^m ((m+n)(l-j)+m-i)(m(l-j+1)-i) \quad (2.2)$$

операций умножения. Вычислим значение этой суммы. Используя формулы для суммы арифметической прогрессии и суммы квадра-

тов членов натурального ряда, для первого слагаемого (2.2) имеем

$$\sum_{i=1}^n (m+n-i)(m(l-j+1)+n-i) = \sum_{i=1}^n (m+n-i)^2 + \\ + \sum_{i=1}^n m(l-j)(m+n-i) = \frac{(m+n-1)(m+n)(2(m+n)-1)}{6} - \\ - \frac{(m-1)m(2m-1)}{6} + \frac{m^2(2m+n-1)(l-j)}{2}. \quad (2.3)$$

В свою очередь, для второго слагаемого справедливо

$$\sum_{i=1}^m ((m+n)(l-j)+m-i)(m(l-j+1)-i) = m^2(m+n) \times \\ \times (l-j)^2 + \frac{(2m+n)m(m-1)(l-j)}{2} + \frac{m(m-1)(2m-1)}{6}. \quad (2.4)$$

Суммируя (2.3) и (2.4) по всем j от 1 до l , получаем общее число действий умножения для факторизации всей матрицы $\partial \bar{F}_k^T(Z)$

$$\frac{(m+n-1)(m+n)(2(m+n)-1)l}{6} + \\ + \frac{m(m+n)(2m+n-2)l(l-1)}{4} + \frac{m^2(m+n)l(l-1)(2l-1)}{6}. \quad (2.5)$$

Преобразование правой части системы (1.7) потребует, соответственно, $(m+n)l$ делений и

$$\sum_{j=1}^l \left(\sum_{i=1}^n (m(l-j+1)+n-i) + \sum_{i=1}^m (m(l-j+1)-i) \right) = \\ = \frac{(m+n)l(ml+n-1)}{2} \\ \text{умножений, суммируя которые, получаем} \\ \frac{(m+n)l(ml+n+1)}{2}. \quad (2.6)$$

Итого, из (2.1), (2.5) и (2.6) следует, что общее количество операций деления и умножения при прямом ходе модифицированного метода Гаусса с перестановкой переменных есть

$$(m+n)l(ml+n) + \frac{(m+n-1)(m+n)(2(m+n)-1)l}{6} + \\ + \frac{m(m+n)(2m+n-2)l(l-1)}{4} + \frac{m^2(m+n)l(l-1)(2l-1)}{6}. \quad (2.7)$$

На обратный ход метода Гаусса потребуется следующее число операций умножения:

$$\sum_{j=0}^{l-1} \left(\sum_{i=0}^{n-1} (i+m) + \sum_{i=0}^{m-1} (i+(m+n)j) \right) = \frac{n(n-1)l}{2} + \\ + \frac{m(m+1)l}{2} + \frac{(m+n)ml(l-1)}{2} + mnl. \quad (2.8)$$

Окончательно, суммируя (2.7) и (2.8), приходим к полному числу операций умножения и деления, затрачиваемому на решение системы (1.7) модифицированным методом Гаусса с перестановкой переменных x и y . Проиллюстрируем преимущество нового метода на конкретном примере. Положим для определенности $m=n=2$ и вычислим общее число операций умножения и деления при $l=1, 2, 3, 4$. Из таблицы 1 следует, что даже при этих небольших значениях параметров m и n при $l=4$ новая модификация метода Гаусса дает практически трехкратное сокращение числа операций по сравнению обычным методом Гаусса и двухкратное — по сравнению с модификацией из статьи [2]. Естественно, с учетом нелинейной зависимости числа операций от параметров m , n и l мы вправе ожидать еще более впечатляющих результатов при росте этих параметров.

Таблица 1. Число операций деления и умножения, затрачиваемое при решении системы (1.7)

Число стадий РК формулы	Метод Гаусса	Модифицированный метод Гаусса	Модифицированный метод Гаусса с перестановкой переменных
$l=1$	36	36	36
$l=2$	232	180	128
$l=3$	716	496	308
$l=4$	1616	1048	608

§ 3. Оптимизация по памяти

При реализации на ЭВМ обычного метода Гаусса для решения системы линейных алгебраических уравнений $((m+n)l)$ -го порядка хранятся все $((m+n)l)^2$ элементов матрицы. Однако с учетом структуры якобиана $\partial \bar{F}_k^T(Z)$, число хранимых элементов

можно сократить. Таким резервом являются незаполняющиеся нулевые блоки матрицы системы (1.7), исключение которых влечет серьезную экономию компьютерной памяти. В модификации из статьи [2] необходимо хранить только $(m+n)l(l(2m+n)+n)/2$ элементов, а учитывая перестановку переменных x и y — только $(m+n)l(n+ml)$ элементов. Как и раньше, приведем конкретные данные в байтах для $m = n = 2$ и $l = 1, 2, 3, 4$ при условии, что для хранения элементов матрицы используются переменные типа *double*. Результаты таблицы 2 позволяют говорить о более экономном расходе памяти ЭВМ при использовании модифицированного метода Гаусса с перестановкой переменных x и y , что дает возможность для решения систем дифференциально-алгебраических уравнений большой размерности.

Таблица 2. Оперативная память, затрачиваемая при решении системы (1.7)

Число стадий РК формулы	Метод Гаусса	Модифицированный метод Гаусса	Модифицированный метод Гаусса с перестановкой переменных
$l=1$	16	16	16
$l=2$	512	448	304
$l=3$	1162	960	768
$l=4$	2048	1664	1280

§ 4. Численные эксперименты

В качестве теста используем следующую задачу:

$$x_1'(t) = 10t \cdot \exp(5(y_2(t) - 1)) \cdot x_2(t), \quad (4.1a)$$

$$x_2'(t) = -2t \cdot \ln(y_1(t)), \quad (4.1б)$$

$$y_1(t) = x_1(t)^{\frac{1}{5}}, \quad (4.1в)$$

$$y_2(t) = (x_2(t)^2 + y_2(t)^2)/2, \quad t \in [t_0, t_0 + T], \quad (4.1г)$$

с начальными условиями:

$$x_1(t_0) = \exp(5 \sin(t_0^2)), \quad x_2(t_0) = \cos(t_0^2), \quad (4.2a)$$

$$y_1(t_0) = \exp(\sin(t_0^2)), \quad y_2(t_0) = \sin(t_0^2) + 1. \quad (4.2б)$$

Задача (4.1) имеет точное решение:

$$x_1(t) = \exp(5 \sin(t^2)), \quad x_2(t) = \cos(t^2), \quad (4.3a)$$

$$y_1(t) = \exp(\sin(t^2)), \quad y_2(t) = \sin(t^2) + 1. \quad (4.3б)$$

Таблица 3. Глобальные ошибки при решении задачи (3.1) РК методом с постоянным шагом $\tau = 10^{-3}$ и числом итераций $N = 4$

Число стадий РК формулы	Метод Гаусса	Модифицированный метод Гаусса	Модифицированный метод Гаусса с перестановкой переменных
$l=1$	$1.394 \cdot 10^{-4}$	$1.394 \cdot 10^{-4}$	$1.394 \cdot 10^{-4}$
$l=2$	$1.718 \cdot 10^{-10}$	$1.718 \cdot 10^{-10}$	$1.718 \cdot 10^{-10}$
$l=3$	$2.498 \cdot 10^{-16}$	$2.498 \cdot 10^{-16}$	$2.498 \cdot 10^{-16}$
$l=4$	$1.665 \cdot 10^{-18}$	$1.665 \cdot 10^{-18}$	$1.665 \cdot 10^{-18}$

Таблица 4. Затраты машинного времени (в сек.) при решении задачи (3.1) РК методом с постоянным шагом $\tau = 10^{-3}$ и числом итераций $N = 4$

Число стадий РК формулы	Метод Гаусса	Модифицированный метод Гаусса	Модифицированный метод Гаусса с перестановкой переменных
$l=1$	0.22	0.22	0.22
$l=2$	1.10	0.71	0.66
$l=3$	3.02	1.81	1.27
$l=4$	6.48	3.40	2.26

Чтобы сравнить эффективность трех модификаций метода Гаусса мы возьмем 4 разновидности РК методов порядков 2, 4, 6 и 8 с числом стадий, соответственно, 1, 2, 3 и 4 [5] и применим к задаче (4.1) на интервале $[1.0708712, 1.4123836]$. В таблице 3 содержатся глобальные ошибки, а в таблице 4 затраты машинного времени в секундах. В каждой точке сетки мы делали 4 итерации метода Ньютона, чтобы гарантировать максимальный порядок для каждого комбинированного метода. Из результатов численных экспериментов следует, что, сохраняя точность РК методов, новый подход к решению линейных систем позволяет действительно уменьшить затраты машинного времени в несколько раз.

Литература

1. Куликов Г.Ю. Теоремы сходимости для итеративных методов Рунге-Кутты с постоянным шагом интегрирования // Ж. вычисл. матем. и матем. физ. 1996. Т. 36. № 8. С. 73–89.
2. Kulikov G.Yu., Thomsen P.G. Convergence and implementation of implicit Runge-Kutta methods for DAEs. Technical report 7/1996, IMM, Technical University of Denmark, Lyngby, 1996.
3. Куликов Г.Ю. О численном решении автономной задачи Коши с алгебраической связью на фазовые переменные (невыврожденный случай) // Вестн. МГУ. Сер. матем., механ. 1993. № 3. С. 10–14.
4. Куликов Г.Ю. О численном решении задачи Коши для системы дифференциально-алгебраических уравнений с помощью неявных методов Рунге-Кутты с переменным шагом интегрирования // Вестн. МГУ. Сер. матем., механ. 1997. № 5. С. 7–11.
5. Хайпер Э., Нерсетт С., Ваннер Г. Решение обыкновенных дифференциальных уравнений. Нежесткие задачи. М.: Мир, 1990.

Г. Ю. Куликов, С. К. Шиндин.

Об одном способе вычисления глобальной погрешности для методов дифференцирования назад

§1. Введение

В настоящее время все известные численные методы с автоматическим выбором шага интегрирования основаны на вычислении главного члена локальной погрешности и последующем выборе шага максимального размера для заданной точности решения задачи. В свою очередь для определения главного члена локальной ошибки используется либо правило Рунге, либо способ, основанный на методах разного порядка [1–3]. Такой подход к построению оптимального по затратам машинного времени разбиения отрезка интегрирования имеет три существенных недостатка. Во-первых, малость главного члена локальной ошибки гарантирует малость самой ошибки лишь при достаточно малом диаметре разбиения. Во-вторых, учет только локальной погрешности не гарантирует достаточную малость глобальной ошибки. И, наконец, как реализация правила Рунге, так и использование методов разных порядков требуют значительных вычислительных затрат при решении обыкновенных дифференциальных уравнений неявными методами с автоматическим выбором шага.

В настоящей статье предложен новый способ практического вычисления главных членов локальной и глобальной ошибок для методов дифференцирования назад. Показано, что предложенные алгоритмы особенно эффективны для неявных методов. Рассмотрены особенности контроля локальной и глобальной ошибок для неявных методов дифференцирования назад.

§2. Вычисление главного члена локальной погрешности

Рассмотрим систему дифференциальных уравнений

$$x'(t) = g(t, x(t)), \quad x(t_0) = x_0, \quad t \in [t_0, t_0 + T], \quad (2.1)$$

с достаточно гладкой функцией $g: \mathbb{R}^{n+1} \rightarrow \mathbb{R}^n$. Для численного решения системы (2.1) введем на отрезке $[t_0, t_0 + T]$ произвольную неравномерную сетку

$$\omega_\tau = \{t_{k+1} = t_k + \tau_k, k = 0, 1, \dots, K-1, t_K = t_0 + T\}$$

с диаметром $\tau = \max_{0 \leq k \leq K-1} \{\tau_k\}$ и применим неявный l -шаговый метод дифференцирования назад

$$\sum_{i=0}^l a_i(k) x_{k+1-i} = \tau_k g(t_{k+1}, x_{k+1}), \quad k = l-1, l, \dots, K-1, \quad (2.2a)$$

где

$$a_0(k) = \sum_{i=1}^l \frac{\tau_k}{t_{k+1} - t_{k+1-i}}, \quad a_i(k) = \sum_{j=i}^l \tau_k \frac{\prod_{m=1}^{j-1} (t_{k+1} - t_{k+1-m})}{\prod_{m=0, m \neq i}^j (t_{k+1-i} - t_{k+1-m})} \quad (2.2b)$$

Будем считать, что стартовые значения x_i , $i = 0, 1, \dots, l-1$, заданы.

Введем функцию

$$L(t_{k+1}, x(t), \tau_k) = \sum_{i=0}^l a_i(k) x(t_{k+1-i}) - \tau_k g(t_{k+1}, x(t_{k+1})), \quad (2.3)$$

$k = 0, 1, \dots, K-1$, которая называется *погрешностью аппроксимации метода дифференцирования назад*. Известно [1-3], что метод (2.2) имеет порядок s тогда и только тогда, когда для любого многочлена $q(t)$ степени не выше s выполняется

$$L(t, q(t), \tau) \equiv 0. \quad (2.4)$$

Для явных методов дифференцирования назад $s = l-1$, для неявных $s = l$ [2, 3].

Введем обозначения: $x(t_k)$ — значение точного решения задачи (2.1) в точке t_k , x_k — значение приближенного решения этой задачи, полученного с помощью метода (2.2), в точке t_k . Тогда,

вычитая (2.2a) из (2.3), имеем

$$\begin{aligned} a_0(k)(x(t_{k+1}) - x_{k+1}) &= - \sum_{i=1}^l a_i(k)(x(t_{k+1-i}) - x_{k+1-i}) + \\ &+ \tau_k (g(t_{k+1}, x(t_{k+1})) - g(t_{k+1}, x_{k+1})) + L(t_{k+1}, x(t), \tau_k) = \\ &= - \sum_{i=1}^l a_i(k)(x(t_{k+1-i}) - x_{k+1-i}) + \tau_k \partial_x g(t_{k+1}, x_{k+1}) \times \\ &\times (x(t_{k+1}) - x_{k+1}) + L(t_{k+1}, x(t), \tau_k) + O((x(t_{k+1}) - x_{k+1})^2), \end{aligned} \quad (2.5)$$

$k = 0, 1, \dots, K-1$. Обозначим погрешность метода (2.2) в точке t_{k+1-i} через $\Delta x_{k+1-i} = x(t_{k+1-i}) - x_{k+1-i}$. Предполагая погрешность в точках t_{k+1-i} , $i = 1, 2, \dots, l$, известной, из (2.5) мы найдем погрешность метода (2.2) в точке t_{k+1}

$$\begin{aligned} \Delta x_{k+1} &= (a_0(k)E_n - \tau_k \partial_x g(t_{k+1}, x_{k+1}))^{-1} \left(- \sum_{i=1}^l a_i(k) \Delta x_{k+1-i} + \right. \\ &\left. L(t_{k+1}, x(t), \tau_k) \right) + O((\Delta x_{k+1})^2), \end{aligned} \quad (2.6)$$

$k = 0, 1, \dots, K-1$. Здесь E_n — единичная матрица размера n .

Положим $\Delta x_{k+1-i} = 0$, $i = 1, 2, \dots, l$, тогда формула (2.6) дает погрешность на шаге для метода (2.2)

$$\Delta \tilde{x}_{k+1} = (a_0(k)E_n - \tau_k \partial_x g(t_{k+1}, \tilde{x}_{k+1}))^{-1} L(t_{k+1}, x(t), \tau_k) + O((\Delta \tilde{x}_{k+1})^2),$$

где $\tilde{x}_{k+1}$ — приближенное решение в точке t_{k+1} , найденное при условии $x_k = x(t_k)$. Так как $L(t_{k+1}, x(t), \tau_k) = O(\tau^{s+1})$ и $\Delta \tilde{x}_{k+1} = O(\tau^{s+1})$, то с точностью до членов более высокого порядка имеем

$$\Delta \tilde{x}_{k+1} \cong (a_0(k)E_n - \tau_k \partial_x g(t_{k+1}, \tilde{x}_{k+1}))^{-1} L(t_{k+1}, x(t), \tau_k), \quad (2.7)$$

$k = l-1, l, \dots, K-1$.

Соотношение (2.7) дает локальную ошибку метода (2.2) с точностью $O(\tau^{2s+2})$. Остается только определить величину $L(t_{k+1}, x(t), \tau_k)$. Из соотношений (2.3) и (2.4) имеем

$$\begin{aligned} L(t_{k+1}, x(t), \tau_k) &= \\ &= \frac{(-1)^s}{(s+1)!} x^{(s+1)}(t_{k+1}) \left(\sum_{i=1}^l a_i(k) \left(\sum_{j=1}^i \tau_{k-j} \right)^{s+1} + (s+1) \tau_k^{s+1} \right) + \\ &+ O(\tau^{s+2}). \end{aligned}$$

Откуда, пренебрегая членами порядка $O(\tau^{s+2})$, окончательно получим

$$L(t_{k+1}, x(t), \tau_k) \cong \frac{(-1)^s}{(s+1)!} x^{(s+1)}(t_{k+1}) \left(\sum_{i=1}^l a_i(k) \left(\sum_{j=1}^i \tau_{k-j} \right)^{s+1} + (s+1) \tau_k^{s+1} \right), \quad k = l-1, l, \dots, K-1, \quad (2.8)$$

где $\tau_{k-j} = t_{k+1} - t_{k+1-j}$. Тогда из (2.7) и (2.8) имеем

$$\psi(t_k) \tau_k^{s+1} = \frac{(-1)^s}{(s+1)!} (a_0(k) E_n - \tau_k \partial_x g(t_{k+1}, \tilde{x}_{k+1}))^{-1} x^{(s+1)}(t_{k+1}) \times \\ \times \left(\sum_{i=1}^l a_i(k) \left(\sum_{j=1}^i \tau_{k-j} \right)^{s+1} + (s+1) \tau_k^{s+1} \right), \quad (2.9)$$

$k = l-1, l, \dots, K-1$, где $\psi(t_k)$ — коэффициент главного члена разложения локальной погрешности $\Delta \tilde{x}_{k+1}$ в ряд Тейлора по степеням τ_k .

Итак, из (2.9) следует, что для вычисления главного члена локальной погрешности метода (2.2) достаточно знать $s+1$ -ую производную точного решения задачи (2.1). Для приближенного вычисления этой величины воспользуемся интерполяционным многочленом Ньютона, построенном по точкам $g(t_{k+1-i}, x_{k+1-i})$, $i = 0, 1, \dots, l$. Тогда, принимая во внимание, что для методов дифференцирования назад $s \leq l$ и дифференцируя этот многочлен s раз, мы получаем величину $x_{k+1}^{(s+1)} = x(t_{k+1})^{(s+1)} + O(\tau)$. Поэтому для главного члена локальной ошибки из (2.9) следует

$$\psi(t_k) \tau_k^{s+1} = \frac{(-1)^s}{(s+1)!} (a_0(k) E_n - \tau_k \partial_x g(t_{k+1}, \tilde{x}_{k+1}))^{-1} x_{k+1}^{(s+1)} \times \\ \times \left(\sum_{i=1}^l a_i(k) \left(\sum_{j=1}^i \tau_{k-j} \right)^{s+1} + (s+1) \tau_k^{s+1} \right), \quad (2.10)$$

$k = l-1, l, \dots, K-1$. Таким образом, формула (2.10) позволяет очень эффективно находить главный член локальной ошибки для методов дифференцирования назад, особенно для неявных методов.

§ 3. Вычисление главного члена глобальной погрешности

В § 2 был предложен эффективный способ вычисления главного члена локальной погрешности методов дифференцирования назад.

Но, как об этом говорилось ранее, знания только главного члена локальной ошибки недостаточно для решения задачи (2.1) с заданной точностью. В этом разделе будет дан практический способ оценки глобальной погрешности метода (2.2). Предложенный способ не требует больших вычислительных затрат и будет особенно эффективен для неявных методов (2.2).

Рассмотрим уравнение (2.6), которое мы вывели для ошибки метода (2.2). Предполагая теперь погрешность в точках t_{k+1-i} , $i = 1, 2, \dots, l$, известной и пренебрегая членами порядка $O(\tau^{2s})$, мы находим погрешность метода (2.2) в точке t_{k+1}

$$\Delta x_{k+1} \cong (a_0(k) E_n - \tau_k \partial_x g(t_{k+1}, x_{k+1}))^{-1} \left(- \sum_{i=1}^l a_i(k) \Delta x_{k+1-i} + L(t_{k+1}, x(t), \tau_k) \right), \quad k = 0, 1, \dots, K-1. \quad (3.1)$$

В § 2 мы доказали, что для вычисления приближенного значения функции $L(t_{k+1}, x(t), \tau_k)$ можно использовать следующую формулу:

$$L(t_{k+1}, x(t), \tau_k) \cong \frac{(-1)^s}{(s+1)!} \tilde{x}_{k+1}^{(s+1)} \left(\sum_{i=1}^l a_i(k) \left(\sum_{j=1}^i \tau_{k-j} \right)^{s+1} + (s+1) \tau_k^{s+1} \right), \quad k = l-1, l, \dots, K-1. \quad (3.2)$$

Таким образом, формулы (3.1) и (3.2) позволяют вычислять главный член погрешности метода (2.2) для любой сетки w_τ с достаточно малым диаметром τ с точностью $O(\tau^{s+1})$.

§ 4. Неявные методы дифференцирования назад

До сих пор мы предполагали, что приближенное решение задачи (2.1), полученное с помощью многошагового метода (2.2), известно точно. Однако для неявных методов это, вообще говоря, не так. В случае применения таких методов для решения нелинейных задач мы получаем (2.2) в виде системы нелинейных алгебраических уравнений, точное решение которой, за редким исключением, не может быть найдено. Поэтому очень важно рассмотреть возможность использования приближенных решений задачи (2.2) для вычисления локальной и глобальной погрешностей метода дифференцирования назад.

Обозначим через $\tilde{x}_k = \tilde{x}_k(N)$ приближенное решение задачи (2.2), полученное за N итераций некоторым итерационным методом. Тогда, принимая во внимание тот факт, что задача Коши для

системы дифференциальных уравнений является частным случаем задачи Коши для системы дифференциально-алгебраических уравнений, мы можем использовать для решения (2.1) следующие три метода [5]:

$$x_{k+1}^i = \bar{G}_{k+1-l}^{\tau} x_{k+1}^{i-1}, \quad i = 1, 2, \dots, N, \quad (4.1a)$$

$$t_{k+1} = t_k + \tau_k, \quad x_{k+1}^0 = \bar{H}_{l-1}(t_{k+1}), \quad k = l-1, l, \dots, K-1, \quad (4.1b)$$

$$\bar{x}_k = x_k, \quad k = 0, 1, \dots, l-1; \quad (4.1c)$$

$$x_{k+1}^i = x_{k+1}^{i-1} - \partial \bar{F}_{k+1-l}^{\tau}(x_{k+1}^{i-1})^{-1} \bar{F}_{k+1-l}^{\tau} x_{k+1}^{i-1}, \quad i = 1, 2, \dots, N, \quad (4.2a)$$

$$t_{k+1} = t_k + \tau_k, \quad x_{k+1}^0 = H_{l-1}(t_{k+1}), \quad k = l-1, l, \dots, K-1, \quad (4.2b)$$

$$\bar{x}_k = x_k, \quad k = 0, 1, \dots, l-1; \quad (4.2c)$$

$$x_{k+1}^i = x_{k+1}^{i-1} - \partial \bar{F}_{k+1-l}^{\tau}(x_{k+1}^0)^{-1} \bar{F}_{k+1-l}^{\tau} x_{k+1}^{i-1}, \quad i = 1, 2, \dots, N, \quad (4.3a)$$

$$t_{k+1} = t_k + \tau_k, \quad x_{k+1}^0 = \bar{H}_{l-1}(t_{k+1}), \quad k = l-1, l, \dots, K-1, \quad (4.3b)$$

$$\bar{x}_k = x_k, \quad k = 0, 1, \dots, l-1, \quad (4.3c)$$

где

$$\bar{G}_{k+1-l}^{\tau} x = - \sum_{i=1}^l \frac{a_i(k)}{a_0(k)} \bar{x}_{k+1-i} + \tau_k \frac{b_0(k)}{a_0(k)} g(t_{k+1}, x),$$

$\bar{F}_{k+1-l}^{\tau} = E_{m+n} - \bar{G}_{k+1-l}^{\tau}$ и $\bar{H}_{l-1}(t)$ — интерполяционный многочлен Ньютона $l-1$ -ой степени, построенный по точкам $\bar{x}_j, j = k, k-1, \dots, k-l+1$. Таким образом, все три метода представляют собой комбинацию неявного метода дифференцирования назад (2.2) и одного из следующих итерационных процессов: метода простых итераций, полного или модифицированного метода Ньютона. Причем для ускорения сходимости используется предиктор порядка $l-1$.

В силу теорем 5–7 из [5] и замечаний статьи [4] для ошибок методов (4.1)–(4.3) справедливы соответственно оценки:

$$x(t_k) - \bar{x}_k(N) = O(\tau^{\xi}), \quad k = l, l+1, \dots, K, \quad (4.4)$$

где $\xi = \min\{N + \zeta - 1, r, s\}$, $\zeta = \min\{l-1, r, s\}$ и стартовые значения заданы с точностью $O(\tau^r)$, $r \geq 1$;

$$x(t_k) - \bar{x}_k(N) = O(\tau^{\xi}), \quad k = l, l+1, \dots, K, \quad (4.5)$$

где $\xi = \min\{(\zeta+1)2^N - 2, r, s\}$;

$$x(t_k) - \bar{x}_k(N) = O(\tau^{\xi}), \quad k = l, l+1, \dots, K, \quad (4.6)$$

где $\xi = \min\{(\zeta+1)(N+1) - 2, r, s\}$. Оценки (4.4)–(4.6) являются основополагающими для нахождения локальной и глобальной ошибок для методов (4.1)–(4.3). Из них следует, что если погрешность итерационного метода есть $O(\tau^{s+2})$, то все результаты § 2 и § 3 остаются справедливыми для методов (4.1)–(4.3). Поэтому нетрудно видеть, что при $N \geq 4$ для метода (4.1) и $N \geq 2$ для методов (4.2), (4.3), формулы (2.10) и (3.1) позволяют вычислять главные члены локальной и глобальной ошибок комбинированных методов.

Литература

1. Базвалов Н.С., Жидков Н.П., Кобельков Г.М. Численные методы. М.: Наука, 1987.
2. Хайрер Э., Нерсетт С., Ваннер Г. Решение обыкновенных дифференциальных уравнений. Нежесткие задачи. М.: Мир, 1990.
3. Арушанян О.Б., Залеткин С.Ф. Численное решение обыкновенных дифференциальных уравнений на Фортране. М.: Изд-во МГУ, 1990.
4. Куликов Г.Ю. Теоремы сходимости для итеративных методов Рунге-Кутты с постоянным шагом интегрирования // Ж. вычисл. матем. и матем. физ. 1996. Т. 36. № 8. С. 73–89.
5. Kulikov G.Yu. Convergence results for iterative implicit multistep methods with fixed step size. Technical report 8/1996, IMM, Technical University of Denmark, Lyngby, 1996.

Ю. Ю. Терентьева.

Помехоустойчивое кодирование при неизвестной статистике источника

В настоящей работе представлен способ преобразования дискретной информации, гарантирующий помехоустойчивую передачу данных по каналу связи при определенных условиях. В основу данного способа положен метод универсального кодирования источника независимых сообщений, изложенный в работах [1,2].

§ 1. Постановка задачи

Пусть имеется источник S , порождающий числа из множества A неотрицательных десятичных целых чисел, $N_i \in A, 1 \leq j \leq m$. Предположим, что статистика источника неизвестна. Обозначим через V последовательность чисел $N_1, N_2, \dots, N_p$, порожденных источником S . Необходимо разработать схему помехоустойчивого кодирования последовательности V для последующей передачи ее по бинарному каналу связи.

§ 2. Длина кодового сообщения

Как указано в [2], любое неотрицательное целое число может быть закодировано в двоичный код заданного ранга. Пусть - некоторая целочисленная функция длины кода C числа N , закодированного с рангом R . Согласно следствию 2 теоремы [2] при выполнении условия $T(P^*, R) \leq N < T(P^* + 1, R)$, где T - кодирующая матрица, минимальной длиной кода является номер строки P^* матрицы T . Следовательно, вид функции $L(N, R)$ может быть задан исходя из кодирующей матрицы. Определим вид функции $L(N, R)$. Для этого зафиксируем некоторый ранг, который обозначим через R . Введем функцию $F(N) = L(N, R)$. Докажем следующее утверждение.

Утверждение 1. Функция $F(N)$ является монотонно неубывающей.

Доказательство. Пусть N_1, N_2 - некоторые неотрицательные целые числа, причем $N_2 > N_1$. Обозначим через C_1 и C_2 - коды чисел N_1 и N_2 соответственно (ранг кодов, упоминавшийся выше, равен R). Возможны два случая.

1. $F(N_1) = F(N_2)$. Тогда определим $P = F(N_1)$.
2. $F(N_1) \neq F(N_2)$. Тогда $P = \max\{F(N_1), F(N_2)\}$. Пусть ψ - оператор декодирования, действующий на множестве двоичных кодов. Тогда $N_1 = ((C_1))$ и $N_2 = ((C_2))$. В силу утверждения 4 [2] операция декодирования задает биективное отображение множества кодов длины P ранга R во множество $\{0, 1, 2, \dots, C_P^R\}$. Следовательно, коды C_1 и C_2 отличны друг от друга, и между ними можно установить отношение предпочтения, предварительно с помощью левосторонней конкатенации нулей код меньшей длины доведя до кода длины P . Заметим, что подобная конкатенация образует код длины P , менее предпочтительный по отношению к другому коду, имеющему изначально большую длину. Отношение предпочтения кодов означает следующее: код $C = (c_1, c_2, \dots, c_p)$ предпочтительней кода $D = (d_1, d_2, \dots, d_p)$ (обозначим $C \gg D$), если существует номер k такой, что $c_k > d_k$ и не существует номера m , такого, что $m < k, c_m < d_m$.

Поскольку операция декодирования биективна, и $\psi(C_1) < \psi(C_2)$, то по утверждению 1 [2] $C_1 \ll C_2$. Из определения предпочтительности и учитывая то, что значимый код числа начинается с единицы согласно алгоритму матрично-рангового кодирования, имеем неравенство $F(N_1) \leq F(N_2)$. Следовательно, справедливо предположение: $N_2 > N_1 \Rightarrow F(N_2) \geq F(N_1)$. Это в свою очередь доказывает монотонное неубывание функции $F(N)$.

Введем ряд обозначений. Пусть $h(M, R) = \text{card}\{N : L(N, R) = M\}$. Определим функцию $S(k, R)$ как

$$S(k) = \begin{cases} 1, & k = R \\ \sum_{i=1+R} h(i, R) + 1, & k > R \end{cases}$$

Утверждение 2. Функция $L(N, R)$ может быть вычислена следующим образом: $L(N, R) = k$, где k удовлетворяет условиям $S(k-1, R) < N+1 \leq S(k, R)$.

Доказательство. В силу монотонного неубывания функции $F(N)$, а также утверждения 4 [2] справедливо $h(i, R) = C_i^R - C_{i-1}^R$ для всех $i > R$. Поэтому функция $L(N, R)$ может быть однозначно определена.

§ 3. Построение специальной функции для реализации помехоустойчивого кодирования

Для реализации помехоустойчивого кодирования необходимо построить преобразование функции $L(N, R)$, удовлетворяющее ряду условий. Необходимое преобразование, а также ряд вышеупомянутых условий рассмотрим в следующем утверждении.

Утверждение 3. Существует преобразование функции $L(N, R)$, которое обеспечивает возможность однозначного декодирования при любом количестве искажений, если длина кодового слова передана верно.

Доказательство. Предположим, что по каналу связи должен быть передан поток целых неотрицательных чисел. Заметим, что любое целое неотрицательное число N^* может быть приведено к числу с любым основанием, т.е. существует $w \in \{1, 2, 3, \dots\}$, вектор $(\alpha_n, \alpha_{n-1}, \dots, \alpha_0)$, такие что $N^* = (\alpha_n, \alpha_{n-1}, \dots, \alpha_1, \alpha_0) (w^n, w^{n-1}, \dots, w, 1)^T$, т.е. представлено скалярным произведением двух векторов $(0 \leq \alpha_i < w, \alpha_i \in Z)$.

Таким образом, для передачи числа N^* достаточно передать вектор $(\alpha_n, \alpha_{n-1}, \dots, \alpha_1, \alpha_0)$, каждая компонента которого заключена в диапазоне от 0 до $w-1$. Следовательно, задача свелась к передаче потока чисел в диапазоне от 0 до $w-1$.

Зададим преобразование функции $L(N, R)$. Пусть $\psi_1(N, R)$ - функция нумерации значений $L(N, R)$. В качестве $\psi_1(N, R)$ можно взять следующую функцию:

$$\psi_1(N, R) = wN + R$$

$$\psi_1^{-1}(N, R) = \begin{cases} N = \psi_{1(N)}^{-1}(k) = [k/w] - 1 \\ R = \psi_{1(R)}^{-1}(k) = (k) \bmod(w) \end{cases}$$

Пусть ψ_2 задаст перестановку некоторого множества $\{1, 2, 3, \dots, Z\}$. Введем функцию

$$\tilde{L}(N, R) = L(\psi_{1(N)}^{-1}(\psi_1(N, R), \psi_1(R)^{-1}(\psi_2(\psi_1(N, R)))).$$

Обязательным требованием для преобразования ψ_2 является то, что оно должно задавать такую перестановку, чтобы обеспечить выполнение следующего условия: для любых R_1 и R_2 , таких, что $R_1 \neq R_2$ справедливо $\tilde{L}(N, R_1) \neq \tilde{L}(N, R_2)$. Это требование заведомо выполнимо, т.к. N и R неограничены. При выполнении

этого условия следует стремиться к тому, чтобы сумма была минимальной (это необходимо для того, чтобы средняя длина кодового слова была как можно меньшей).

Таким образом, существует преобразование функции $L(N, R)$, на основе которого может быть построен алгоритм помехоустойчивой передачи потока данных при технической фиксации длины кодового сообщения.

§ 4. Алгоритм помехоустойчивой передачи потока данных

Пусть необходимо обеспечить помехоустойчивую (до 100%) передачу множества $\{N_i\}$, где $N_i \in \{0, 1, \dots, w-1\}$. Предположим, что первое число потока N_1 известно передатчику и приемнику.

КОДИРОВАНИЕ

Шаг 1. Если $N_2 = 0$, положить $R_1 = w$, иначе R_1 положить равным $\tilde{N}_2$ ($\tilde{N}_2$ - последующее число).

Шаг 2. $K_1 = \psi_1(N_1, N_2)$

Шаг 3. $K_2 = \psi_2(N_1, N_2)$

Шаг 4. $\tilde{N}_1 = \psi_{1(N)}^{-1}(K_2)$

$\tilde{R}_1 = \psi_{1(R)}^{-1}(K_2)$.

Шаг 5. Коду (N_1, R_1) сопоставляется код $(\tilde{N}_1, \tilde{R}_1)$. Кодирование окончено.

ДЕКОДИРОВАНИЕ Пусть получен код длиной L . Приемник ожидает число N_1 (заметим, что первое число потока согласовано кодером и декодером).

Шаг 1. $R_1 = \tilde{L}_R^{-1}(L, N_1)$

Шаг 2. Если $R_1 = w$, то N_2 положить равным нулю, иначе N_2 присвоить значение R_1 . Шаг 3. Положить N_1 равным N_2 . Декодирование окончено.

Данный способ преобразования дискретной информации для последующей передачи ее по каналу связи предусматривает возможности:

- 1) 100%-ой помехоустойчивости при технической фиксации длины кодового сообщения;
- 2) параллельной передачи другого (независимого от первого) потока дискретной информации, что в свою очередь способствует повышению скорости передачи данных по каналу связи;
- 3) обеспечения секретности за счет вариативности ранга кода, а также за счет вышеизложенного 2);

4) контроля избыточности кодируемых сообщений за счет вариативности w (см. утверждение 3);

5) параллельного диагностирования канала связи.

Кроме того, следует отметить, что разработанный метод помехоустойчивого кодирования может эффективно применяться к архивам, а также последовательностям с неизвестной статистикой с целью передачи их по бинарному каналу связи.

Литература

1. В.Ф.Бабкин. Метод универсального кодирования независимых сообщений неэкспоненциальной трудоемкости. // Проблемы передачи информации. - 1971. - Т.7, 4. - С.13-21.

2. А.А.Смагин, Ю.Ю.Терентьева. Способ преобразования дискретной информации. Фундаментальные проблемы математики и механики: Ученые записки УлГУ. Вып.1. Часть 2. Ульяновск: УлГУ, 1996, - 144с.

В. К. Горбунов, В. В. Петрищев.

Метод нормальных сплайнов в вырожденных системах дифференциальных уравнений

1. Актуальность развития численных методов решения систем обыкновенных дифференциальных уравнений обосновывается наличием в этом базовом классе вычислительных задач ряда особенностей, не охватываемых известными методами общего назначения [1]. Это "жесткие" и дифференциально-алгебраические системы, задачи на бесконечных промежутках, другие типы особенностей. Для таких задач обычно строятся специальные методы, часто основанные на достаточно сложной алгебраической теории [2,3].

В [4,5] был предложен метод нормальной сплайн-коллокации для интегро-дифференциальных линейных уравнений, неразрешенных относительно производных. Минимальные требования, обеспечивающие его сходимость, допускают теоретически все отмеченные особенности. Метод был реализован для систем второго порядка и показал высокую эффективность для "жесткого" случая с малым параметром при старшей производной. В данной работе излагается вычислительная схема метода для систем дифференциальных уравнений произвольного порядка и приводятся результаты решения ряда "жестких" и вырожденных задач.

2. Рассмотрим систему уравнений

$$A(t)\frac{dx(t)}{dt} + B(t)x(t) = f(t), \quad a \leq t \leq b, \quad (1)$$

с условиями

$$Cx(a) + Dx(b) = g, \quad (2)$$

где $x, f, g \in R^n$, $A(t), B(t), C, D$ - квадратные матрицы порядка n . Система (1), (2) считается разрешимой в гильбертовом-соболевском пространстве $W_{2,n}^2(a, b)$. В случае неединственности под решением задачи будем понимать нормальное решение. В данной работе при реализации метода используется норма

$$\|x\| = \left[\sum_{i=1}^n \left[x_i^2(a) + \left(\frac{dx_i(a)}{dt} \right)^2 + \int_a^b \left(\frac{d^2 x_i(s)}{ds^2} \right)^2 ds \right] \right]^{1/2}. \quad (3)$$

Излагаемый ниже метод допускает вырождение матрицы $A(t)$. Этим охватывается проблема решения дифференциально-алгебраических систем [4], причем в универсальной схеме.

Метод коллокации основан на переходе от функциональных уравнений (в данном случае (1)) к конечной системе равенств на некоторой сетке $a \leq t_1 < t_2 < \dots < t_m \leq b$, т.е. система (1) замещается системами

$$A(t_k) \frac{dx(t_k)}{dt} + B(t_k)x(t_k) = f(t_k), \quad 1 \leq k \leq m. \quad (4)$$

Совокупность систем (4) конечна и не может определять (вместе с условиями (2)) решение $x(t)$ однозначно. Однако система (2), (4) имеет единственное нормальное решение, которое обозначим x^m . Это решение представляется в виде линейной комбинации набора функций, порождаемых структурой системы и выбранной нормой. Для пояснения этого и изложения алгоритма метода системе (2), (4) необходимо записать в координатной форме, введя обозначения

$$l_{ik}(x) = \begin{cases} \sum_{j=1}^n [c_{ij}x_j(0) + d_{ij}x_j(1)], & k=0, \\ \sum_{j=1}^n [a_{ij}(t_k)dx_j(t_k)/dt + b_{ij}x_j(t_k)], & 1 \leq k \leq m. \end{cases} \quad (5)$$

Соответственно, система (2), (4) принимает вид

$$l_{i0}(x) = g_i, \quad l_{ik}(x) = f_i(t_k), \quad 1 \leq i \leq n, \quad 1 \leq k \leq m. \quad (6)$$

Значения координат элементов $W_{2,n}^2(a, b)$ $x_j(t)$ и их производных $dx_j(t)/dt$ в точках t_k являются линейными непрерывными функционалами в пространстве скалярных функций $W_{2,n}^2(a, b)$, норма которого в соответствии с (3)

$$\|x_j\| = \left[x_j^2(a) + \left(\frac{dx_j(a)}{dt} \right)^2 + \int_a^b \left(\frac{d^2x_j(s)}{ds^2} \right)^2 ds \right]^{1/2}. \quad (7)$$

Эти элементарные функционалы порождают составные функционалы (также линейные и непрерывные) (5), уже в $W_{2,n}^2(a, b)$. Соответственно, каждое из $m \times n + n$ равенств (6) определяет гиперплоскость и вся система - пересечение гиперплоскостей. Это пересечение выпукло и замкнуто, а значит оно имеет элемент минимальной нормы x^m . Этот элемент и является аппроксимацией искомого нормального (в общем случае) решения x^0 задачи (1), (2).

Аппроксимирующая задача является частным случаем задач, определяющих обобщенные сплайны [6]. Особенность минимизируемого функционала (3) определяет название ее решения x^m как нормальный обобщенный сплайн.

Сходимость x^m к x^0 (в норме $W_{2,n}^2$) обеспечивается при произвольном измельчении сеток [4,5] (максимальный шаг должен стремиться к нулю). Отметим, что это существенно упрощает реализацию метода нормальной сплайн-коллокации относительно известных методов сплайн-коллокации [7], основанных на априорном представлении решения в виде некоторого сплайна. Кроме того, для нашего метода имеет место следующая оценка точности аппроксимации, важная для процесса сгущения сеток.

Обозначим невязку уравнения (1) на решении x^m

$$\varphi^m(t) = A(t) \frac{dx^m(t)}{dt} + B(t)x^m(t) - f(t). \quad (8)$$

Имеет место оценка [5]

$$\|x^m - x^0\| \leq C \|\varphi^m\|. \quad (9)$$

Здесь C - константа, ограничивающая (равномерно) нормы обратных операторов задач минимизации нормы (3) при условиях (6). Так как сходимость в норме $W_{2,n}^2$ влечет равномерную сходимость, и невязки $\varphi^m(t_k) = 0$, то в процессе реализации метода для повышения точности следует переходить к новым сеткам, добавляя узлы в области наибольших промежуточных (между узлами) значений $\varphi^m(t)$.

3. Особенностью метода нормальной сплайн-коллокации является представление линейных непрерывных функционалов (5) коллокационной системы (6) (в соответствии с теоремой Ф.Рисса) в каноническом виде - скалярном произведении, соответствующем норме (3). Эти функционалы пространства векторных функций $x(t) = (x_1(t), \dots, x_n(t))$ строятся из элементарных функционалов $x_j(t_k)$ и $dx_j(t_k)/dt$ в пространстве скалярных функций $W_{2,1}^2(a, b)$ с нормой (7).

В дальнейшем ограничимся единичным отрезком $[0,1]$, выполняя при реализации метода соответствующее преобразование исходной задачи.

Представление элементарных функционалов в $W_{2,1}^2$ осуществляется с помощью воспроизводящего ядра (функции Грина некоторой краевой задачи [4,5]), которое для нормы (7) при $a=0$ и $b=1$ имеет вид

$$G(s, t) = \begin{cases} 1 + (s + s^2/2)t - s^3/6, & 0 \leq s \leq t \leq 1, \\ G(t, s), & 0 \leq t \leq s \leq 1. \end{cases}$$

Воспроизводящее ядро $G(s, t)$ при фиксированном втором аргументе рассматривается как элемент $G(\cdot, t) \in W_{2,1}^2$ и определяет значения функций $x_j(t)$ в точках t_k через скалярные произведения, соответствующие норме (7) [4,5]:

$$x_j(t_k) = \langle G(\cdot, t_k), x_j \rangle = x_i(0)G(0, t_k) + \frac{dx_i(0)}{dt} \frac{\partial G(0, t_k)}{\partial s}. \quad (10)$$

Значение производной в точках имеет представление

$$\frac{dx_i(t_k)}{dt} = \left(\frac{\partial G(\cdot, t_k)}{\partial t}, x_i \right) \quad (11)$$

Имея эти представления, построим канонические образы линейных функционалов (5) пространства $W_{2,n}^2$, определяющих левые части решаемой системы (6).

Каноническими образами этих функционалов являются вектор-функции $h^\mu(s)$, принадлежащие $W_{2,n}^2$ (соответственно $h_j^\mu \in W_{2,1}^2$) и такие, что

$$l_{ik}(x) = \langle h^{\mu(i,k)}, x \rangle_n = \sum_{j=1}^n \langle h_j^{\mu(i,k)}, x_j \rangle. \quad (12)$$

Здесь $\langle \cdot, \cdot \rangle_n$ является скалярным произведением в $W_{2,n}^2$. Соответствие индексов установим формулой

$$\mu(i, k) = nk + i, \quad 1 \leq i \leq n, \quad 0 \leq k \leq m. \quad (13)$$

В силу определений (5), (11) система (6), содержащая $N = m \times n + n$ уравнений, записывается в каноническом виде

$$\langle h^\mu, x \rangle_n = \bar{f}_\mu, \quad 1 \leq \mu \leq N, \quad (14)$$

где функции h^μ в соответствии с (5), (10), (12) имеют компоненты

$$h_j^{\mu(i,k)}(s) = \begin{cases} c_{ij}G(s, 0) + d_{ij}G(s, 1), & k = 0, \\ a_{ij}(t_k)\partial G(s, t_k)/\partial t + b_{ij}(t_k)G(s, t_k), & 1 \leq k \leq m, \end{cases} \quad (15)$$

и правая часть (14)

$$\bar{f}_{\mu(i,k)} = \begin{cases} g_i, & k = 0, \\ f_i(t_k), & 1 \leq k \leq m. \end{cases} \quad (16)$$

Итак, аппроксимирующая задача метода нормальной сплайн-коллокации заключается в нормальном решении системы линейных уравнений (в гильбертовом пространстве $W_{2,n}^2$) (13). Это решение представляется в виде

$$x^m(s) = \sum_{\mu=1}^N u_\mu h^\mu(s), \quad (17)$$

и коэффициенты u_μ определяются системой линейных уравнений

$$\sum_{\nu=1}^N g_{\mu\nu} u_\nu = \bar{f}_\mu, \quad 1 \leq \mu \leq N, \quad (18)$$

где $g_{\mu\nu}$ - коэффициенты матрицы Грама системы h^μ , т.е.

$$g_{\mu\nu} = \langle h^\mu, h^\nu \rangle_n = \sum_{j=1}^n \langle h_j^\mu, h_j^\nu \rangle \equiv \sum_{j=1}^n g_j^{\mu\nu}. \quad (19)$$

В соответствии с (15) и свойствами (10), (11) воспроизводящего ядра $G(s, t)$ [4,5], компоненты правой части (19) $g_j^{\mu\nu}$ имеют следующий вид:

$$g_j^{\mu(p,k), \nu(q,l)} = \begin{cases} c_{pj}c_{qj}G(0, 0) + (c_{pj}d_{qj} + c_{qj}d_{pj})G(0, 1) + \\ d_{pj}d_{qj}G(1, 1), & k = l = 0; \\ c_{pj} \left[a_{qj}(t_l) \frac{\partial G(0, t_l)}{\partial t} + b_{qj}(t_l)G(0, t_l) \right] + \\ d_{pj} \left[a_{qj}(t_l) \frac{\partial G(1, t_l)}{\partial t} + b_{qj}(t_l)G(1, t_l) \right], & 0 = k < l \leq m; \\ a_{pj}(t_k)a_{qj}(t_l) \frac{\partial^2 G(t_k, t_l)}{\partial s \partial t} + b_{qj}(t_l)b_{pj}(t_k)G(t_k, t_l) + \\ [a_{pj}(t_k)b_{qj}(t_l) + a_{qj}(t_l)b_{pj}(t_k)] \frac{\partial G(t_k, t_l)}{\partial t}, & 1 \leq k \leq l \leq m. \end{cases} \quad (20)$$

Матрица Грама $\{g_{\mu\nu}\}$ симметрична, поэтому достаточно сформировать ее коэффициенты при $1 \leq \mu \leq \nu \leq N$. Из формулы (13) при этом следуют ограничения для вычислений (19): $1 \leq k \leq l \leq m$, и если $k = l$, то $1 \leq p \leq q \leq n$.

Таким образом, реализация метода нормальной сплайн-коллокации при заданном разбиении отрезка сводится к формированию коэффициентов матрицы Грама в соответствии с (19), (20), решению системы линейных уравнений (18) с симметричной положительно определенной (как правило) матрицей $\{g_{\mu\nu}\}$ и построению решения $x^m(s)$ в произвольных точках $s \in [0, 1]$ по (15), (17).

Для решения системы (17) в общем случае следует использовать метод квадратного корня [1] и в случае плохой обусловленности или вырождения матрицы Грама эффективным является метод сопряженных направлений, удобный для итеративной регуляризации вычислительного процесса [6].

4. Описанный метод нормальной сплайн-коллокации реализован и опробован на ряде тестовых примеров, в частности, неразрешенных относительно производных. Приведем результаты решения двух сингулярных задач Коши.

Пример 1. Задача с вырожденной матрицей [2]:

$$\begin{pmatrix} 1 & t \\ 1 & t \end{pmatrix} \times \dot{x}(t) + \begin{pmatrix} -1 & 0 \\ 0 & -1 \end{pmatrix} \times x(t) = \begin{pmatrix} (t+1)^2 \\ (t+1)^2 - 1 \end{pmatrix},$$

$$x(0) = \begin{pmatrix} 1 \\ 2 \end{pmatrix}, \quad t \in [0, 1],$$

точное решение которой

$$x(t) = ((1+t)^2, (1+t)^2 + 1).$$

В [2] задача решалась сведением к минимизации функционала $J(u) = \|A(t) \times u(t) - B(t) \times x(t) - f(t)\|_{2(n)}^2$ градиентным методом. При этом интервал $[0, 1]$ разбивался на 40 участков дискретизации с шагом $h = 0.025$. Критерий окончания счета $J(u^e) \leq \varepsilon = 0.00001$. Получены значения $x_1^e(1) = 3.99524437546$, $x_2^e(1) = 4.99615829771$. Промежуточные результаты удовлетворяют оценке $|x_j^e(t_i) - x_j^*(t_i)| \leq 0.005$, $j = 1, 2$; $t_i = h \times i$, $i = 0, 40$.

Основные результаты эксперимента иллюстрирует Таблица 1:

Таблица 1.

N	$x_1(1)$	$\Delta x_1(1)$	$x_2(1)$	$\Delta x_2(1)$	$\max \Delta x$
40	3.99524	0.004756	4.99616	0.003842	0.005
7	3.99546	0.004539	4.99546	0.004539	0.005
8	3.99653	0.003474	4.99653	0.003474	0.0035
40	3.99986	0.000139	4.99986	0.000139	0.00014

Результаты решения задачи автором (первая строка) и наши сведены в табл.1. В первой строке приведены результаты В.Ф.Чистякова (на сетке 40 узлов) и в остальных - полученные методом нормальных сплайнов с числом узлов коллокации N. В последней колонке указана округленная (с превышением) величина погрешности координат решения в узлах сетки с 40 узлами.

Пример 2. Задача с вырожденной матрицей [3, С.208-209].

$$\begin{pmatrix} t^2 & 2t^2 & t^2 & t \\ t^2 & t^2 & 2t^2 & t \\ t^2 & t^2 & t^2 & t \\ 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{pmatrix} = \begin{pmatrix} 3 & 3t & t & 0 \\ t & 3 & 3t & 0 \\ t & t & 3 & 0 \\ t & t & t & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} +$$

$$\begin{pmatrix} 4t^2 - 3t - 3 \\ 4t^2 - 3t - 3 \\ 3t^2 - t - 3 \\ -3t - 1 \end{pmatrix} \exp(t), t \in [1, 2],$$

$$s = 3, x(1) = (e, e, e, e)^T, \quad x_i(t) = \exp(t), \quad i = 1..4$$

Результаты счета приведены в таблице 2.

Таблица 2.

N	15	30	60
Δ_1	0.01969865	0.00493497	0.00123407
Δ_2	0.002548	0.000929	0.00032

Результаты [3] получены специальным коллокационно-разностным методом. Соответствующие отклонения от точного решения (в норме l_1) приведены в первой строке табл.2. Во второй строке приведены отклонения (в норме l_2) решения, полученного методом нормальных сплайнов.

Работа поддержана РФФИ (грант № 96-01-00509).

Литература

1. Бахвалов Н.С., Жидков Н.П., Кобельников Г.М.: Численные методы, М.Мир, 1988.
2. Чистяков В.Ф.: К методам решения сингулярных линейных систем обыкновенных дифференциальных уравнений // "Вырожденные системы обыкновенных дифференциальных уравнений", Новосибирск: Наука, 1982.
3. Бояринцев Ю.Е., Данилов В.А., Логинов А.А., Чистяков В.Ф.: Численные методы решения сингулярных систем. - Новосибирск: Наука, 1980.
4. Горбунов В.К.: Метод нормальной сплайн-коллокации. // ЖВМиМФ, 1989, N2, с.212-224.
5. Горбунов В.К.: Экстремальные задачи обработки результатов измерений, -Фрунзе: Илим, 1990.
6. Горбунов В.К., Кохановский И.И.: Итеративная регуляризация на основе сингулярного анализа. Вестник Моск. ун-та. Сер.15. (ВМК). 1995. N1, 17-20
7. Завьялов Ю.С., Квасов Б.И., Мирошниченко В.Л.: Методы сплайн-функций. М.: Наука 1980.

Ученые записки

Ульяновского государственного университета

Фундаментальные проблемы математики и механики

Выпуск 3

Под редакцией *Б.Ф.Мельникова*

ЛР №021233 от 23.06.97.

Подписано в печать 22.12.97. Формат 84x108/32.

Бумага книжно-журнальная. Гарнитура TimesET.

Усл. печ. л. 6,9. Уч.-изд. л. 6,1. Тираж 100 экз.

Заказ №148/ *817*

Отпечатано с оригинал-макета
в лаборатории оперативной полиграфии
Ульяновского государственного университета
432700, г.Ульяновск, ул. Л.Толстого, 42