



Адрес редакции
670000, г. Улан-Удэ,
ул. Смолина, 24а
E-mail: univer@bsu.ru

Адрес издателя
670000, г. Улан-Удэ,
ул. Смолина, 24а
E-mail: riobsu@gmail.com

Свидетельство о регистрации
ПИ №ФС77-36152 от 06 мая
2009 г. Федеральная служба по
надзору в сфере связи, инфор-
мационных технологий
и массовых коммуникаций (Рос-
комнадзор)

Стиль авторов статей сохранен
Компьютерная верстка
Д.О. Трунина

Подписано в печать 25.12.14.
Формат 60 x 84 1/8.
Уч.-изд. л. 4,56. Усл. печ. л. 9,3.
Тираж 1000. Заказ 307.
Цена свободная.
Дата выхода в свет 26.12.14.

Отпечатано в типографии
Издательства БГУ
670000, г. Улан-Удэ,
ул. Сухэ-Батора, 3а

Учредитель
ФГБОУ ВПО «Бурятский государственный университет»

ВЕСТНИК БУРЯТСКОГО ГОСУДАРСТВЕННОГО УНИВЕРСИТЕТА

Издается с 1997 г.

Выходит 15 раз в год

Выпуск МАТЕМАТИКА И ИНФОРМАТИКА 9 (3) / 2014

Журнал включен Высшей аттестационной комиссией в Перечень ведущих рецензируемых научных журналов и изданий, в которых должны быть опубликованы основные научные результаты диссертаций на соискание ученых степеней доктора и кандидата наук

Редакционный совет «Вестника БГУ»

С.В. Калмыков, чл.-кор. РАО, д-р пед. наук, проф. (председатель); В.Е. Архинчеев, д-р физ.-мат. наук, доц.; Н.И. Атанов, д-р экон. наук, проф.; Т.С. Базарова, д-р пед. наук, доц.; А.С. Булдаев, д-р физ.-мат. наук, проф.; Д.И. Бураев, д-р ист. наук, проф.; А.В. Гаськов, д-р пед. наук, проф.; Н.Ж. Дагбаева, д-р пед. наук, проф.; Ц.З. Доржиев, д-р биол. наук, проф.; С.С. Имхелова, д-р филол. наук, проф.; Л.П. Ковалева, канд. филол. наук, проф.; К.Б-М. Митупов, д-р ист. наук, проф.; И.И. Осинский, д-р филос. наук, проф.; М.Н. Очиров, д-р пед. наук, проф.; В.В. Хахинов, д-р хим. наук, проф.; В.Е. Хитрихеев, д-р мед. наук, проф.

Редакционная коллегия выпуска

А.С. Булдаев, д-р физ.-мат. наук, проф. (главный редактор); С.Н. Васильев, акад. РАН; В.И. Гурман, д-р техн. наук, проф.; Ж.Г. Дамбаев, д-р техн. наук, проф.; А.В. Данеев, д-р техн. наук, проф.; А.И. Кожанов, д-р физ.-мат. наук, проф.; А.В. Лакеев, д-р физ.-мат. наук, проф.; А.Д. Мижидон, д-р техн. наук, проф.; Ни Минь Кань, д-р наук, проф.; Б. Очирбат, д-р наук, проф.; В.А. Срочко, д-р физ.-мат. наук, проф.; Д.Ш. Ширанов, д-р физ.-мат. наук, проф.; Р. Энхбат, д-р наук, проф.



1. Управляемые системы и методы оптимизации

УДК 517.977

© А.С. Булдаев

ОПТИМИЗАЦИЯ УПРАВЛЯЮЩИХ ФУНКЦИЙ И ПАРАМЕТРОВ В НЕЛИНЕЙНЫХ СИСТЕМАХ НА ОСНОВЕ ЗАДАЧ О НЕПОДВИЖНОЙ ТОЧКЕ¹

Предлагается новый подход в классе нелинейных задач оптимального управления, содержащих одновременно управляющие функции и параметры, основывающийся на решении специальной задачи о неподвижной точке конструируемых операторов в пространстве управлений. Рассматриваемая задача о неподвижной точке дает возможность получать новые условия оптимальности управления в рассматриваемом классе оптимизационных задач и строить улучшающие управления.

Ключевые слова: управляемая система, задача о неподвижной точке, условия оптимальности

© A.S. Buldaev

OPTIMIZATION OF THE MANAGING FUNCTIONS AND PARAMETERS IN NONLINEAR SYSTEMS ON BASIS OF THE FIXED POINT PROBLEMS

A new approach to the class of nonlinear optimal control problems containing both managing functions and parameters, is proposed on basis of the solution of special fixed point problems for operators constructed in the space of controls. The fixed point problem makes it possible to obtain new conditions for optimal control in the class of optimization problems and build improving controls.

Keywords: controlled system, fixed point problem, conditions of optimality

Введение

Многие задачи теории управления в актуальных приложениях могут быть представлены как задачи оптимального управления с параметрами, в которых кроме неизвестных параметров требуется дополнительно определить и управляющую функцию. Например, к таким задачам сводятся заменой переменной времени задачи с нефиксированной длительностью окончания процесса управления. Известные подходы к решению рассматриваемых задач основываются на необходимых [1] или достаточных [2] условиях оптимальности.

В данной работе предлагается новый подход, который является развитием и обобщением предыдущих результатов, полученных в классах нелинейных задач оптимального управления [3-5]. Метод иллюстрируется в рамках следующего класса задач оптимального управления:

$$\Phi(\sigma) = \varphi(x(t_1), \omega) + \int_T F(x(t), u(t), \omega, t) dt \rightarrow \inf_{\sigma \in \Omega}, \quad (1)$$

$$\dot{x}(t) = f(x(t), u(t), \omega, t), \quad x(t_0) = a, \quad u(t) \in U, \quad \omega \in W, \quad a \in A, \quad t \in T = [t_0, t_1], \quad (2)$$

в котором $x(t) = (x_1(t), \dots, x_n(t))$ – вектор состояния, $u(t) = (u_1(t), \dots, u_m(t))$ – вектор управляющей функции, $\omega = (\omega_1, \dots, \omega_l)$ и $a = (a_1, \dots, a_n)$ – векторные управляющие параметры. Множества $U \subseteq R^m$, $W \subseteq R^l$, $A \subseteq R^n$ замкнуты и выпуклы. Интервал T фиксирован. В качестве допустимых управляющих функций рассматривается множество V кусочно-непрерывных на T функций со значениями в множестве U . $\sigma = (u, \omega, a)$ – допустимое управление со значениями в множестве $\Omega = V \times W \times A$.

Предполагаются выполненными следующие предположения:

1) функция $\varphi(x, \omega)$ непрерывно-дифференцируема на $R^n \times W$, функции $F(x, u, \omega, t)$, $f(x, u, \omega, t)$ и их частные производные по x , u , ω непрерывны по совокупности аргументов (x, u, ω, t) на множестве $R^n \times U \times W \times T$;

¹ Работа выполнена при финансовой поддержке РФФИ, гранты 12-01-00914-а, 12-01-98011-р_сибирь_а, 13-01-92200-Монг_а

2) функция $f(x, u, \omega, t)$ удовлетворяет условию Липшица по x в $R^n \times U \times W \times T$ с константой $L > 0$: $\|f(x, u, \omega, t) - f(y, u, \omega, t)\| \leq L\|x - y\|$.

Условия гарантируют существование и единственность решения $x(t, \sigma)$, $t \in T$ системы (2) для любого допустимого управления $\sigma \in \Omega$.

Рассмотрим стандартные конструкции: функцию Понтрягина с сопряженной переменной $\psi \in R^n$

$$H(\psi, x, u, \omega, t) = \langle \psi, f(x, u, \omega, t) \rangle - F(x, u, \omega, t)$$

и стандартную сопряженную систему

$$\dot{\psi}(t) = -H_x(\psi(t), x(t), u(t), \omega, t), \quad t \in T, \quad \psi(t_1) = -\varphi_x(x(t_1), \omega), \quad (3)$$

на основе которых формулируются известные необходимые и достаточные условия оптимальности [1,2].

Для допустимого управления $\sigma \in \Omega$ обозначим $\psi(t, \sigma)$, $t \in T$ - решение стандартной сопряженной системы (3) при $x(t) = x(t, \sigma)$ и аргументах u , ω , соответствующих компонентам управления σ .

Одним из основных подходов к оптимизации управления является последовательное решение задач улучшения управления, в которых для заданного управления $\sigma^I \in \Omega$ требуется найти управление $\sigma^{II} \in \Omega$ с условием $\Phi(\sigma^{II}) - \Phi(\sigma^I) \leq 0$.

В данной работе для решения задачи улучшения предлагается решить специальную задачу о неподвижной точке конструируемого оператора в пространстве управлений с помощью рассматриваемых операций на максимум. Решение предлагаемой задачи о неподвижной точке позволяет улучшать неоптимальные управления, удовлетворяющие принципу максимума, и получать новые необходимые условия оптимальности в рассматриваемом классе задач.

1. Формула приращения функционала

Далее для удобства частное приращение произвольной вектор-функции $g(y_1, \dots, y_l)$ по переменным y_{s_1} , y_{s_2} будем обозначать

$$\Delta_{y_{s_1} + \Delta y_{s_1}, y_{s_2} + \Delta y_{s_2}} g(y_1, \dots, y_l) = g(y_1, \dots, y_{s_1} + \Delta y_{s_1}, \dots, y_{s_2} + \Delta y_{s_2}, \dots, y_l) - g(y_1, \dots, y_l).$$

Приращение функционала (1) на допустимых управлениях σ , σ^I в соответствии с введенным обозначением выписывается в виде:

$$\Delta_\sigma \Phi(\sigma^I) = \Delta_{x(t_1, \sigma), \omega} \varphi(x(t_1, \sigma^I), \omega^I) + \int_T \Delta_{x(t, \sigma), u(t), \omega} F(x(t, \sigma^I), u^I(t), \omega^I, t) dt. \quad (4)$$

Дополнительно обозначим $\Delta x(t) = x(t, \sigma) - x(t, \sigma^I)$, $\Delta \omega = \omega - \omega^I$, $\Delta a = a - a^I$.

Приращение терминальной части функционала в выражении (4) можно записать в виде:

$$\Delta_{x(t_1, \sigma), \omega} \varphi(x(t_1, \sigma^I), \omega^I) = \Delta_\omega \varphi(x(t_1, \sigma), \omega^I) + \Delta_{x(t_1, \sigma)} \varphi(x(t_1, \sigma^I), \omega^I).$$

Введем непрерывную кусочно-дифференцируемую вектор-функцию $p(t) = (p_1(t), \dots, p_n(t))$, $t \in T$ с условиями:

$$p(t_1) = -\varphi_x(x(t_1, \sigma^I), \omega^I) - q, \quad (5)$$

где величина q удовлетворяет алгебраическому уравнению

$$\langle \varphi_x(x(t_1, \sigma^I), \omega^I) + q, \Delta x(t_1) \rangle = \Delta_{x(t_1, \sigma)} \varphi(x(t_1, \sigma^I), \omega^I). \quad (6)$$

При этом по определению полагаем $q = 0$ в случае линейности функции φ по x , а также в случае $x(t_1, \sigma) = x(t_1, \sigma^I)$.

Тогда приращение терминальной части функционала в выражении (4) можно записать в виде:

$$\begin{aligned} \Delta_{x(t_1, \sigma), \omega} \varphi(x(t_1, \sigma^l), \omega^l) &= \Delta_{\omega} \varphi(x(t_1, \sigma), \omega^l) - \langle p(t_1), \Delta x(t_1) \rangle = \\ &= \Delta_{\omega} \varphi(x(t_1, \sigma), \omega^l) - \langle p(t_0), \Delta a \rangle - \int_T \frac{d}{dt} \langle p(t), \Delta x(t) \rangle dt = \\ &= \Delta_{\omega} \varphi(x(t_1, \sigma), \omega^l) - \langle p(t_0), \Delta a \rangle - \\ &- \int_T \left\{ \langle \dot{p}(t), \Delta x(t) \rangle + \langle p(t), \Delta_{x(t, \sigma), u(t), \omega} f(x(t, \sigma^l), u^l(t), \omega^l, t) \rangle \right\} dt. \end{aligned}$$

На основе полученного соотношения с помощью функции Понтрягина приращение функционала (4) можно представить в виде:

$$\begin{aligned} \Delta_{\sigma} \Phi(\sigma^l) &= \Delta_{\omega} \varphi(x(t_1, \sigma), \omega^l) - \langle p(t_0), \Delta a \rangle - \\ &- \int_T \left\{ \langle \dot{p}(t), \Delta x(t) \rangle - \Delta_{x(t, \sigma), u(t), \omega} H(p(t), x(t, \sigma^l), u^l(t), \omega^l, t) \right\} dt = \\ &= \Delta_{\omega} \varphi(x(t_1, \sigma), \omega^l) - \langle p(t_0), \Delta a \rangle - \\ &- \int_T \left\{ \langle \dot{p}(t), \Delta x(t) \rangle + \Delta_{\omega} H(p(t), x(t, \sigma), u(t), \omega^l, t) + \right. \\ &\left. + \Delta_{u(t)} H(p(t), x(t, \sigma), u^l(t), \omega^l, t) + \Delta_{x(t, \sigma)} H(p(t), x(t, \sigma^l), u^l(t), \omega^l, t) \right\} dt. \end{aligned} \quad (7)$$

Введем дифференциально-алгебраическую систему для функции $p(t)$ с условиями (5), (6) в форме:

$$\dot{p}(t) = -H_x(p(t), x(t, \sigma^l), u^l(t), \omega^l, t) - r(t), \quad (8)$$

где кусочно-непрерывная величина $r(t) = (r_1(t), \dots, r_n(t))$, $t \in T$ определяется в каждый момент времени $t \in T$ из алгебраического уравнения

$$\begin{aligned} \langle H_x(p(t), x(t, \sigma^l), u^l(t), \omega^l, t) + r(t), \Delta x(t) \rangle &= \\ &= \Delta_{x(t, \sigma)} H(p(t), x(t, \sigma^l), u^l(t), \omega^l, t). \end{aligned} \quad (9)$$

При этом по определению полагаем $r(t) = 0$ в случае линейности функций F , f по x , а также в случае равенства $x(t, \sigma) = x(t, \sigma^l)$.

Тогда в силу дифференциально-алгебраической системы (8), (9) для $p(t)$ с начальными условиями (5), (6) формула приращения (7) принимает вид:

$$\begin{aligned} \Delta_{\sigma} \Phi(\sigma^l) &= -\Delta_{\omega} \{ -\varphi(x(t_1, \sigma), \omega^l) + \int_T H(p(t), x(t, \sigma), u(t), \omega^l, t) dt \} - \\ &- \langle p(t_0), \Delta a \rangle - \int_T \Delta_{u(t)} H(p(t), x(t, \sigma), u^l(t), \omega^l, t) dt. \end{aligned} \quad (10)$$

Для удобства записи явной зависимости $p(t)$ от управления введем новую конструкцию: модифицированную дифференциально-алгебраическую сопряженную систему в форме

$$\dot{p}(t) = -H_x(p(t), x(t), u(t), \omega, t) - r(t), \quad (11)$$

$$\langle H_x(p(t), x(t), u(t), \omega, t) + r(t), y(t) - x(t) \rangle = \Delta_{y(t)} H(p(t), x(t), u(t), \omega, t) \quad (12)$$

с краевыми условиями

$$p(t_1) = -\varphi_x(x(t_1), \omega) - q, \quad (13)$$

$$\langle \varphi_x(x(t_1), \omega) + q, y(t_1) - x(t_1) \rangle = \Delta_{y(t_1)} \varphi(x(t_1), \omega), \quad (14)$$

в которой по определению полагаем $q = 0$, $r(t) = 0$ в случае линейности функций φ , F , f по x (линейной по состоянию задачи (1), (2)), а также в случае $x(t, \sigma) = x(t, \sigma^l)$ при соответствующих $t \in T$.

В линейной по состоянию задаче (1), (2) модифицированная сопряженная система (11)-(14) по определению совпадает со стандартной сопряженной системой (3).

Для допустимых управлений $\sigma \in \Omega$, $\sigma^l \in \Omega$ обозначим $p(t, \sigma^l, \sigma)$, $t \in T$ - решение модифицированной сопряженной системы (11)-(14) при $x(t) = x(t, \sigma^l)$, $y(t) = x(t, \sigma)$, $u(t) = u^l(t)$, $\omega = \omega^l$. Из определения следует очевидное равенство $p(t, \sigma, \sigma) = \psi(t, \sigma)$, $t \in T$.

Формула приращения (10) в новых обозначениях, в которых указывается явная зависимость фазовых и сопряженных переменных от управления, принимает вид

$$\Delta_{\sigma} \Phi(\sigma^I) = -\Delta_{\omega} \left\{ -\varphi(x(t_1, \sigma), \omega^I) + \int_T H(p(t, \sigma^I, \sigma), x(t, \sigma), u(t), \omega^I, t) dt \right\} - \left\langle p(t_0, \sigma^I, \sigma), \Delta a \right\rangle - \int_T \Delta_{u(t)} H(p(t, \sigma^I, \sigma), x(t, \sigma), u^I(t), \omega^I, t) dt. \tag{15}$$

Отметим нестандартный вид формулы (10), которая не содержит каких-либо остаточных членов разложений. Формулы приращения функционалов от управления подобного вида получили название «нелокальных» формул. Соответствующие методы улучшения управления, основывающиеся на таких формулах, обладают свойством нелокальности улучшения и позволяют получать новые условия оптимальности, а также возможность улучшать управления, удовлетворяющие принципу максимума [3 - 5].

Используемые в работе обозначения решений фазовых и сопряженных систем, явно указывающие их зависимость от управления, являются удобными для формулировки и анализа условий улучшения и оптимальности управления в форме задач о неподвижной точке.

2. Задача о неподвижной точке на основе операций на максимум

При заданном $\sigma^I \in \Omega$ определим оператор $A^* : \sigma \rightarrow \sigma^*$ на множестве допустимых управлений Ω следующими выражениями:

$$u \rightarrow u^*, \quad u^*(t) = \arg \max_{\tilde{u} \in U} H(p(t, \sigma^I, \sigma), x(t, \sigma), \tilde{u}, \omega^I, t), \quad t \in T,$$

$$\omega \rightarrow \omega^*, \quad \omega^* = \arg \max_{\tilde{\omega} \in W} \left\{ -\varphi(x(t_1, \sigma), \tilde{\omega}) + \int_T H(p(t, \sigma^I, \sigma), x(t, \sigma), u(t), \tilde{\omega}, t) dt \right\},$$

$$a \rightarrow a^*, \quad a^* = \arg \max_{\tilde{a} \in A} \left\langle p(t_0, \sigma^I, \sigma), \tilde{a} \right\rangle.$$

Задача о неподвижной точке $\sigma = A^*(\sigma)$ для рассматриваемого оператора A^* определяется следующей системой уравнений:

$$u(t) = \arg \max_{\tilde{u} \in U} H(p(t, \sigma^I, \sigma), x(t, \sigma), \tilde{u}, \omega^I, t), \quad t \in T, \tag{16}$$

$$\omega = \arg \max_{\tilde{\omega} \in W} \left\{ -\varphi(x(t_1, \sigma), \tilde{\omega}) + \int_T H(p(t, \sigma^I, \sigma), x(t, \sigma), u(t), \tilde{\omega}, t) dt \right\}, \tag{17}$$

$$a = \arg \max_{\tilde{a} \in A} \left\langle p(t_0, \sigma^I, \sigma), \tilde{a} \right\rangle. \tag{18}$$

Предположим, что задача (16)-(18) имеет решение $\sigma^{II} = (u^{II}, \omega^{II}, a^{II})$ (возможно, не единственное) и управление u^{II} является кусочно-непрерывным. Тогда, в силу определения отображения A^* , получаем

$$\Delta_{u^{II}(t)} H(p(t, \sigma^I, \sigma^{II}), x(t, \sigma^{II}), u^I(t), \omega^I, t) \geq 0, \quad t \in T,$$

$$\Delta_{\omega^{II}} \left\{ -\varphi(x(t_1, \sigma^{II}), \omega^I) + \int_T H(p(t, \sigma^I, \sigma^{II}), x(t, \sigma^{II}), u^{II}(t), \omega^I, t) dt \right\} \geq 0,$$

$$\left\langle p(t_0, \sigma^I, \sigma^{II}), a^{II} - a^I \right\rangle \geq 0.$$

Отсюда и из формулы (15) следует $\Delta_{\sigma^{II}} \Phi(\sigma^I) \leq 0$.

Введем отображение

$$u^*(p, x, t) = \arg \max_{\tilde{u} \in U} H(p, x, \tilde{u}, \omega^I, t), \quad p \in R^n, \quad x \in R^n, \quad t \in T.$$

Краевая задача улучшения в пространстве фазовых и сопряженных переменных, эквивалентная задаче о неподвижной точке (16)-(18), имеет вид

$$\dot{x}(t) = f(x(t), u^*(p(t), x(t), t), \omega^I, t), \quad x(t_0) = a^*, \tag{19}$$

$$\dot{p}(t) = -H_x(p(t), x^I(t), u^I(t), \omega^I, t) - r(t), \tag{20}$$

$$\left\langle H_x(p(t), x^I(t), u^I(t), \omega^I, t) + r(t), x(t) - x^I(t) \right\rangle = \Delta_{x(t)} H(p(t), x^I(t), u^I(t), \omega^I, t) \tag{21}$$

$$p(t_1) = -\varphi_x(x^I(t_1), \omega^I) - q, \tag{22}$$

$$\langle \varphi_x(x^l(t_1), \omega^l) + q, x(t_1) - x^l(t_1) \rangle = \Delta_{x(t_1)} \varphi(x^l(t_1), \omega^l), \quad (23)$$

в которой

$$\omega^* = \arg \max_{\tilde{\omega} \in W} \{-\varphi(x(t_1), \tilde{\omega}) + \int_T H(p(t), x(t), u^*(p(t), x(t), t), \tilde{\omega}, t) dt)\}, \quad (24)$$

$$a^* = \arg \max_{\tilde{a} \in A} \langle p(t_0), \tilde{a} \rangle. \quad (25)$$

Предположим, что решение $(x(t), p(t))$, $t \in T$ интегро-дифференциально-алгебраической краевой задачи (19) – (25) существует (возможно, не единственное), а управляющая функция, формируемая по правилу

$$u^H(t) = u^*(p(t), x(t), t), \quad t \in T,$$

является кусочно-непрерывной. Сформируем

$$\omega^H = \arg \max_{\tilde{\omega} \in W} \{-\varphi(x(t_1), \tilde{\omega}) + \int_T H(p(t), x(t), u^H(t), \tilde{\omega}, t) dt)\},$$

$$a^H = \arg \max_{\tilde{a} \in A} \langle p(t_0), \tilde{a} \rangle,$$

Тогда $x(t) = x(t, \sigma^H)$, $p(t) = p(t, \sigma^l, \sigma^H)$ и, в силу определения σ^H , получаем

$$\Delta_{u^H(t)} H(p(t, \sigma^l, \sigma^H), x(t, \sigma^H), u^H(t), \omega^l, t) \geq 0,$$

$$\Delta_{\omega^H} \{-\varphi(x(t_1, \sigma^H), \omega^l) + \int_T H(p(t, \sigma^l, \sigma^H), x(t, \sigma^H), u^H(t), \omega^l, t) dt\} \geq 0,$$

$$\langle p(t_0, \sigma^l, \sigma^H), a^H - a^l \rangle \geq 0.$$

Отсюда и из формулы (15) следует, что $\Delta_{\sigma^H} \Phi(\sigma^l) \leq 0$.

Эквивалентность краевой задачи (19)-(25) и задачи о неподвижной точке (16)-(18) понимается в следующем смысле. Пусть пара $(x(t), p(t))$, $t \in T$, является решением краевой задачи (19)-(25). Тогда управление σ^H , формируемое по указанному выше правилу, является решением задачи о неподвижной точке (16)-(18). Наоборот, пусть допустимое управление σ^H является решением задачи (16)-(18). Тогда пара $(x(t, \sigma^H), p(t, \sigma^l, \sigma^H))$, $t \in T$ является решением задачи (19)-(25).

Таким образом, метод улучшения управления $\sigma^l \in \Omega$ состоит в решении задачи о неподвижной точке (16)-(18) или в решении эквивалентной ей краевой задачи (19)-(25).

Установим связь необходимых условий оптимальности управления с задачей о неподвижной точке.

Необходимые условия оптимальности управления $\sigma \in \Omega$ в задаче (1), (2) в форме операции на максимум, аналогичной работе [1], можно получить на основе формулы приращения (15) в следующем виде:

$$u(t) = \arg \max_{\tilde{u} \in U} H(\psi(t, \sigma), x(t, \sigma), \tilde{u}, \omega, t), \quad t \in T, \quad (26)$$

$$\omega = \arg \max_{\tilde{\omega} \in W} \left\langle -\varphi_\omega(x(t_1, \sigma), \omega) + \int_T H_\omega(\psi(t, \sigma), x(t, \sigma), u(t), \omega, t) dt, \tilde{\omega} \right\rangle, \quad (27)$$

$$a = \arg \max_{\tilde{a} \in A} \langle \psi(t_0, \sigma), \tilde{a} \rangle. \quad (28)$$

Условия (26)-(28) обобщают принцип максимума, полученный в [1] для частного случая задачи (1), (2) (отсутствие зависимости функции φ от параметра ω , фиксированное начальное условие a для фазовой переменной x в момент времени t_0).

Обозначим $\Omega(\sigma^l) \subseteq \Omega$ - множество допустимых неподвижных точек задачи (16)-(18). Пусть $\sigma^l \in \Omega(\sigma^l)$, т.е. выполняются равенства:

$$u^l(t) = \arg \max_{\tilde{u} \in U} H(\psi(t, \sigma^l), x(t, \sigma^l), \tilde{u}, \omega^l, t), \quad t \in T, \quad (29)$$

$$\omega^l = \arg \max_{\tilde{\omega} \in W} \{-\varphi(x(t_1, \sigma^l), \tilde{\omega}) + \int_T H(\psi(t, \sigma^l), x(t, \sigma^l), u^l(t), \tilde{\omega}, t) dt)\}, \quad (30)$$

$$a^l = \arg \max_{\tilde{a} \in A} \langle \psi(t_0, \sigma^l), \tilde{a} \rangle. \quad (31)$$

В линейной по ω задаче (1), (2) условия (29)-(31) совпадают с необходимыми условиями опти-

мальности (26)-(28) для управления $\sigma^l \in \Omega$. Следовательно, справедливо следующее утверждение.

Лемма. В линейной по переменной ω задаче (1), (2) управление $\sigma \in \Omega$ удовлетворяет условиям принципа максимума (26)-(28) тогда и только тогда, когда $\sigma \in \Omega(\sigma)$.

Следствие (принцип максимума в терминах задачи о неподвижной точке). Пусть управление $\sigma \in \Omega$ является оптимальным в линейной по ω задаче (1), (2). Тогда $\sigma \in \Omega(\sigma)$.

Другие следствия (в линейной по ω задаче (1), (2)).

1. Задача о неподвижной точке (16)-(18), и, следовательно, краевая задача (19)-(25), всегда разрешимы для управления, удовлетворяющего принципу максимума.
2. В случае неединственности решения задачи о неподвижной точке (16)-(18) появляется принципиальная возможность строгого улучшения управления, удовлетворяющего принципу максимума. Такая возможность иллюстрируется в работах [3 - 5] для частных случаев задачи (1), (2).
3. Отсутствие неподвижных точек в процедуре улучшения управления свидетельствует о неоптимальности управления.

Отметим, что условия принципа максимума (26)-(28) можно рассматривать как отдельную независимую задачу о неподвижной точке $\sigma = \bar{A}(\sigma)$ в пространстве управлений для определяемого правыми частями этих условий соответствующего оператора $\bar{A}: \sigma \rightarrow \bar{\sigma}$:

$$u \rightarrow \bar{u}, \bar{u}(t) = \arg \max_{\tilde{u} \in U} H(\psi(t, \sigma), x(t, \sigma), \tilde{u}, \omega, t), t \in T,$$

$$\omega \rightarrow \bar{\omega}, \bar{\omega} = \arg \max_{\tilde{\omega} \in W} \left\langle -\varphi_{\omega}(x(t_1, \sigma), \omega) + \int_T H_{\omega}(\psi(t, \sigma), x(t, \sigma), u(t), \omega, t) dt, \tilde{\omega} \right\rangle,$$

$$a \rightarrow \bar{a}, \bar{a} = \arg \max_{\tilde{a} \in A} \langle \psi(t_0, \sigma), \tilde{a} \rangle.$$

Таким образом, для поиска «подозрительных на оптимальность» (экстремальных) управлений можно рассматривать систему уравнений, определяющую задачу о неподвижной точке $\sigma = \bar{A}(\sigma)$:

$$u(t) = \arg \max_{\tilde{u} \in U} H(\psi(t, \sigma), x(t, \sigma), \tilde{u}, \omega, t), t \in T,$$

$$\omega = \arg \max_{\tilde{\omega} \in W} \left\langle -\varphi_{\omega}(x(t_1, \sigma), \omega) + \int_T H_{\omega}(\psi(t, \sigma), x(t, \sigma), u(t), \omega, t) dt, \tilde{\omega} \right\rangle,$$

$$a = \arg \max_{\tilde{a} \in A} \langle \psi(t_0, \sigma), \tilde{a} \rangle,$$

или эквивалентную интегро-дифференциально-алгебраическую краевую задачу принципа максимума:

$$\dot{x}(t) = f(x(t), \bar{u}(\psi(t), x(t), \bar{\omega}, t), \bar{\omega}, t), x(t_0) = \bar{a},$$

$$\dot{\psi}(t) = -H_x(\psi(t), x(t), \bar{u}(\psi(t), x(t), \bar{\omega}, t), \bar{\omega}, t),$$

$$\psi(t_1) = -\varphi_x(x(t_1), \bar{\omega}),$$

в которой

$$\bar{u}(\psi, x, \omega, t) = \arg \max_{\tilde{u} \in U} H(\psi, x, \tilde{u}, \omega, t), \psi \in R^n, x \in R^n, \omega \in W, t \in T,$$

$$\bar{\omega} = \arg \max_{\tilde{\omega} \in W} \left\langle -\varphi_{\omega}(x(t_1), \bar{\omega}) + \int_T H_{\omega}(\psi(t), x(t), \bar{u}(\psi(t), x(t), \bar{\omega}, t), \bar{\omega}, t) dt, \tilde{\omega} \right\rangle,$$

$$\bar{a} = \arg \max_{\tilde{a} \in A} \langle \psi(t_0), \tilde{a} \rangle.$$

Отметим, что краевая задача улучшения (19)-(25) существенно проще по свойствам гладкости правой части системы для сопряженной переменной и структуре зависимости параметра ω от фазовых и сопряженных переменных, чем краевая задача принципа максимума.

Предлагаемый подход к улучшению управления проиллюстрируем на примере линейной по состоянию задачи (1), (2), в которой модифицированная сопряженная система (11)-(14) совпадает со стандартной сопряженной системой (3). В этом случае задача о неподвижной точке (16)-(18) существенно упрощается и принимает следующий вид

$$u(t) = \arg \max_{\tilde{u} \in U} H(\psi(t, \sigma^l), x(t, \sigma), \tilde{u}, \omega^l, t), t \in T,$$

$$\omega = \arg \max_{\tilde{\omega} \in W} \left\{ -\varphi(x(t_1, \sigma), \tilde{\omega}) + \int_T H(\psi(t, \sigma^l), x(t, \sigma), u(t), \tilde{\omega}, t) dt \right\},$$

$$a = \arg \max_{\tilde{a} \in A} \left\langle \psi(t_0, \sigma^l), \tilde{a} \right\rangle.$$

При этом уравнение для компоненты a является независимым от остальных уравнений системы.

Эквивалентная краевая задача улучшения (19)-(25) сводится к специальной интегродифференциальной задаче Коши

$$\dot{x}(t) = f(x(t), u^*(p(t, \sigma^l), x(t), t), \omega^*, t), \quad x(t_0) = a^*,$$

в которой

$$u^*(p, x, t) = \arg \max_{\tilde{u} \in U} H(p, x, \tilde{u}, \omega^l, t), \quad p \in R^n, \quad x \in R^n, \quad t \in T,$$

$$\omega^* = \arg \max_{\tilde{\omega} \in W} \left\{ -\varphi(x(t_1), \tilde{\omega}) + \int_T H(\psi(t, \sigma^l), x(t), u^*(\psi(t, \sigma^l), x(t), t), \tilde{\omega}, t) dt \right\},$$

$$a^* = \arg \max_{\tilde{a} \in A} \left\langle \psi(t_0, \sigma^l), \tilde{a} \right\rangle.$$

Таким образом, в данном примере задача о неподвижной точке и краевая задача для поиска улучшающего управления значительно проще задачи о неподвижной точке и краевой задачи для поиска экстремального управления.

3. Вычислительные особенности

Для решения задачи о неподвижной точке

$$\sigma = A^*(\sigma), \quad \sigma \in \Omega, \quad (32)$$

можно использовать известный в вычислительной математике метод последовательных приближений и его модификации [6]. В частности, можно применить явный метод простой итерации при $k \geq 0$, имеющий форму:

$$\sigma^{k+1} = A^*(\sigma^k), \quad \sigma^0 \in \Omega.$$

Для улучшения сходимости итерационного процесса задачу о неподвижной точке (32) можно преобразовать к эквивалентной задаче с параметром $\delta > 0$:

$$\sigma = \sigma + \delta(\sigma - A^*(\sigma)), \quad \sigma \in \Omega, \quad (33)$$

на основе которой получаем модификацию итерационного процесса:

$$\sigma^{k+1} = \sigma^k + \delta(\sigma^k - A^*(\sigma^k)), \quad \sigma^0 \in \Omega.$$

Основным условием сходимости метода простой итерации является выполнение свойства «сжимания» [6] для оператора правой части задачи о неподвижной точке. Поэтому, выбирая достаточно малый параметр $\delta > 0$, можно регулировать сходимость рассматриваемой модификации (33) метода простой итерации.

Расчет задачи о неподвижной точке производится до первого улучшения исходного управления σ^l . Далее строится новая задача улучшения для полученного управления σ'' и расчет повторяется. Итерации улучшения продолжаются до тех пор, пока не выполнится условие

$$\left| \Phi(\sigma'') - \Phi(\sigma^l) \right| \leq \varepsilon \left| \Phi(\sigma^l) \right|,$$

где $\varepsilon > 0$ - заданная точность расчета.

Явный метод простой итерации и его модификации можно использовать также для решения задачи о неподвижной точке $\sigma = \bar{A}(\sigma)$ с целью поиска экстремальных управлений.

Заключение

Предлагаемый подход последовательного решения задач улучшения на основе поиска неподвижных точек в рассматриваемом классе задач оптимального управления характеризуется отсутствием процедуры игольчатого или выпуклого варьирования управлений и обладает принципиальной возможностью строгого улучшения неоптимальных экстремальных управлений. В целом, оптимизация управлений на основе расчета конструируемых задач о неподвижной точке явными методами последовательных приближений сводится к последовательному решению задач Коши для фазовых и сопряженных переменных. Указанные свойства являются важными факторами повышения вычислительной и качественной эффективности решения задач оптимального управления.

Литература

1. Понтрягин Л.С., Болтянский В.Г., Гамкрелидзе Р.В., Мищенко Е.Ф. Математическая теория оптимальных процессов. – М.: Наука, 1976. – 392 с.
2. Батурин В.А., Урбанович Д.Е. Приближенные методы оптимального управления, основанные на принципе расширения. – Новосибирск: Наука, 1997. – 175 с.
3. Булдаев А.С. Методы возмущений в задачах улучшения и оптимизации управляемых систем. – Улан-Удэ: Изд-во Бурятск. гос. ун-та, 2008. – 260 с.
4. Булдаев А.С. Улучшение управлений в нелинейных системах на основе краевых задач / А.С. Булдаев, О.В. Моржин // Известия Иркутского госуниверситета. Серия «Математика». – 2009. – Т.2, №1. – С. 94-106.
5. Булдаев А.С. Методы неподвижных точек в задачах параметрической оптимизации систем / А.С. Булдаев, И.-Х.Д. Хишектуева // Автоматика и телемеханика. – 2013. – №12. – С. 5-14.
6. Самарский А.А., Гулин А.В. Численные методы. – М.: Наука, 1989. – 432 с.

Булдаев Александр Сергеевич, доктор физико-математических наук, профессор кафедры прикладной математики Бурятского государственного университета, тел. (301-2) 217733, E-mail: buldaev@mail.ru.

Buldaev Alexander Sergeevich, doctor of physical and mathematical science, professor of applied mathematics department of Buryat State University.

УДК 517.977

© Е.В. Дрыганова

МЕТОД ОПТИМИЗАЦИИ ЛИНЕЙНО УПРАВЛЯЕМЫХ ДИСКРЕТНЫХ СИСТЕМ¹

Предлагается проекционный метод нелокального улучшения управлений в классе линейных задач оптимального управления дискретными системами.

Ключевые слова: дискретная система, нелокальное улучшение, проекционные методы.

© E. V. Dryganova

THE METHOD OF LINEAR CONTROLLED DISCRETE SYSTEM OPTIMIZATION

The article deals with the projective method of nonlocal improvement of controlling parameters in the class of linear problem of optimal controlled discrete systems.

Keywords: discrete system, nonlocal improvement, projective methods.

Введение

В работах [1-3] в классах линейных и нелинейных непрерывных задач оптимального управления построены методы нелокального улучшения управляющих функций, основанные на точных формулах приращения целевых функционалов. Характерной положительной чертой этих методов является нелокальность улучшения, т.е. отсутствие трудоемкой операции слабого или игольчатого варьирования управления на каждой итерации улучшения управления.

В данной работе рассматривается итерационный метод нелокального улучшения управления для линейных дискретных задач оптимального управления.

1. Проекционный метод нелокального улучшения

Рассматривается билинейная дискретная задача оптимального управления

$$\Phi(u) = \varphi(x(t_N)) + \sum_{i=0}^{N-1} F(x(t_i), u(t_i), t_i) \rightarrow \min_{u \in V}, \quad (1)$$

$$x(t_{i+1}) = f(x(t_i), u(t_i), t_i), \quad x(t_0) = x_0, \quad u(t_i) \in U \subseteq E^m, \quad t_i \in T, \quad i = \overline{0, N-1}, \quad (2)$$

где $x(t_i) = (x_1(t_i), \dots, x_n(t_i))$ - состояние системы, $u(t_i) = (u_1(t_i), \dots, u_m(t_i))$ - управление, $t_i \in T = \{t_0, t_1, \dots, t_{N-1}, t_N\}$; моменты t_0, t_N и состояние x_0 заданы.

В задаче (1)-(2):

1) функция $\varphi(x)$ линейна по x , интегрант $F(x, u, t)$ и вектор-функция $f(x, u, t)$ линейны по x и u на множестве $E^n \times U \times T$.

2) множество $U \subseteq E^m$ выпуклое и замкнутое, $V = \{v : v(t) \in U, t \in T\}$ - множество допустимых дискретных управлений.

Определим необходимые конструкции для задачи (1)-(2). Введем сопряженную переменную $\psi \in E^n$ и рассмотрим функцию Понтрягина

$$H(\psi, x, u, t) = \langle H_1(\psi, x, t), u \rangle + H_0(\psi, x, t).$$

Определим сопряженную систему

$$\psi(t_k) = -H_x(\psi(t_{k+1}), x(t_k), u(t_k), t_k), \quad \psi(t_N) = -\varphi_x(x(t_N)), \quad (3)$$

где $k = \overline{0, N-1}$.

Обозначим через $x(t_i, u), \psi(t_i, u), t_i \in T$ - решения фазовой и сопряженной (2),(3) систем при заданном управлении u .

¹ Работа выполнена при финансовой поддержке РФФИ, гранты 12-01-00914-а, 12-01-98011-р_сибирь_а, 13-01-92200-Монг_а

В [4] показано, что приращение функционала в нелинейной задаче для допустимых процессов $(x(t_i, u), u)$ и $(x(t_i, v), v)$ записывается в виде

$$\Phi(v) - \Phi(u) = - \sum_{i=0}^{N-1} (H(\psi(t_{i+1}, u), x(t_i, v), v(t_i), t_i) - H(\psi(t_{i+1}, u), x(t_i, v), u(t_i), t_i)). \quad (4)$$

Следует отметить, что формула (4) точная в смысле отсутствия остаточных членов разложений в ряд Тейлора по x, u .

В линейной задаче (1),(2) имеем

$$\begin{aligned} & H(\psi(t_{i+1}, u), x(t_i, v), v(t_i), t_i) - H(\psi(t_{i+1}, u), x(t_i, v), u(t_i), t_i) = \\ & = \langle H_1(\psi(t_{i+1}, u), x(t_i, v), t_i), v(t_i) - u(t_i) \rangle. \end{aligned}$$

Тогда формула (4) принимает вид

$$\Phi(v) - \Phi(u) = - \sum_{i=0}^{N-1} \langle H_1(\psi(t_{i+1}, u), x(t_i, v), t_i), v(t_i) - u(t_i) \rangle. \quad (5)$$

Поставим задачу улучшения заданного управления $u \in U$: найти управление $v \in U$ с условием $\Phi(v) \leq \Phi(u)$.

Проекционный метод нелокального улучшения:

1) Для заданной допустимой пары $(x(t_i, u), \psi(t_i, u))$, $t_i \in T$, определим вспомогательное отображение с параметром $\alpha > 0$

$$\bar{u}(x, t) = P_U(u(t) + \alpha H_1(\psi(t, u), x, t)), t \in T, x \in E^n,$$

где P_U - оператор проектирования на множество U в евклидовой норме.

2) Найдем решение $x(t_i)$ фазовой системы

$$x(t_{i+1}) = f(x(t_i), \bar{u}(x(t_i), t_i), t_i), x(t_0) = x_0, t_i \in T, i = \overline{0, N-1},$$

и построим выходное управление $v(t_i) = \bar{u}(x(t_i), t_i), t_i \in T$.

Покажем выполнения свойства улучшения $\Phi(v) \leq \Phi(u)$ на выходном управлении $v(t_i), t_i \in T$.

Согласно свойству проекций [5] справедливо неравенство

$$\langle H_1(\psi(t_{i+1}, u), x(t_i, v), t_i), v(t_i) - u(t_i) \rangle \geq \frac{1}{\alpha} \|v(t_i) - u(t_i)\|^2, \alpha > 0,$$

с учетом которого получаем оценку уменьшения целевого функционала на процессах $(x(t_i, u), u(t_i)), (x(t_i, v), v(t_i)), t_i \in T$:

$$\begin{aligned} \Phi(v) - \Phi(u) &= - \sum_{i=0}^{N-1} \langle H_1(\psi(t_{i+1}, u), x(t_i, v), t_i), v(t_i) - u(t_i) \rangle \leq \\ &\leq - \frac{1}{\alpha} \sum_{i=0}^{N-1} \|v(t_i) - u(t_i)\|^2. \end{aligned} \quad (6)$$

Отсюда заключаем, что $\Phi(v) - \Phi(u) \leq 0$, т.е. проекционный метод нелокального улучшения для заданного $u \in V$ позволяет получить улучшающее управление $v \in V$ с оценкой улучшения (6).

2. Пример

Проиллюстрируем работу метода на простом примере [6]:

$$\begin{aligned} \Phi(u) &= \int_1^2 (x(t) + u(t)) dt \rightarrow \min, \\ \dot{x} &= u(t), x(1) = 5, u \in U = [-3, 1], t \in T = [1, 2]. \end{aligned}$$

В данной задаче легко находится аналитическое решение $u^* = -3, \Phi(u^*) = 0.5$.

Дискретный вариант задачи имеет вид

$$\Phi(u) = \sum_{i=0}^{N-1} (x(t_i) + u(t_i)) \rightarrow \min,$$

$$x(t_{i+1}) = x(t_i) + u(t_i), x(1) = 5, u \in U = [-3, 1], i = \overline{0, N-1}$$

$$t_i \in T = [t_0, \dots, t_N], t_0 = 1, t_2 = 1 + \Delta t, \dots, t_{N-1} = 2 - \Delta t, t_N = 2,$$

где $\Delta t > 0$ – шаг дискретизации.

В данном случае

$$H(\psi, x, u, t) = \psi(x + u) - (x + u), \quad H_1 = \psi - 1, \quad H_0 = (\psi - 1)x.$$

Сопряженная система, решение которой $\psi(t_i), t_i \in T$ не зависит от управления и состояния, имеет вид

$$\psi(t_i) = 1 - \psi(t_{i+1}), \quad \psi(2) = 0, \quad i = \overline{0, N-1}.$$

Для заданного управления u отображение \bar{u} с параметром $\alpha > 0$ имеет вид

$$\bar{u}(x, t_i) = P_U(u(t_i) + \alpha(\psi(t_{i+1}) - 1)).$$

Задача улучшения для управления u состоит в решении фазовой системы

$$x(t_{i+1}) = x(t_i) + \bar{u}(x(t_i), t_i), \quad x(1) = 5, \quad i = \overline{0, N-1},$$

с выходным управлением $v(t_i) = \bar{u}(x(t_i), t_i)$.

Для найденного улучшающего управления v процесс повторяется. Критерий окончания итераций улучшения $|\Phi(v) - \Phi(u)| \leq \varepsilon |\Phi(u)|$ с параметром $\varepsilon = 10^{-5}$. Расчеты проводились с различным шагом дискретизации Δt , который менялся в диапазоне от 10^{-1} до 10^{-4} .

Расчетное значение $\tilde{u} \approx -3$ с заданной точностью ε было получено на второй итерации метода для различных входных управлений u при $\alpha = 10^2$. Результаты расчетов отражены в таблицах 1-4 (u - начальное значение управления, $\Phi(\tilde{u})$ - расчетное значение функционала).

u	$\Phi(\tilde{u})$
3.0	0.65
-2.0	0.65
-1.0	0.65
0	0.65
1.0	0.65

Таблица 1. $\Delta t = 10^{-1}$

u	$\Phi(\tilde{u})$
3.0	0.515
-2.0	0.515
-1.0	0.515
0	0.515
1.0	0.515

Таблица 2. $\Delta t = 10^{-2}$

u	$\Phi(\tilde{u})$
3.0	0.5015
-2.0	0.5015
-1.0	0.5015
0	0.5015
1.0	0.5015

Таблица 3. $\Delta t = 10^{-3}$

u	$\Phi(\tilde{u})$
3.0	0.50015
-2.0	0.5001
-1.0	0.50078
0	0.5005
1.0	0.5005

Таблица 4. $\Delta t = 10^{-4}$

Результирующие значения целевого функционала, как и следовало ожидать, при уменьшении шага дискретизации приближаются к оптимальному значению непрерывной задачи $\Phi(u^*) = 0.5$.

Заключение

Разработанный алгоритм для линейных дискретных задач не использует операцию выпуклого варьирования, характерную для градиентных методов. Это свойство является существенным фактором для повышения эффективности решения дискретных оптимизационных задач.

Литература

1. Срочко В.А. Итерационные методы решения задач оптимального управления. – М.: Физматлит, 2000. – 160 с.
2. Булдаев А.С. Методы возмущений в задачах улучшения и оптимизации управляемых систем. – Улан-Удэ: Изд-во Бурятского госуниверситета, 2008. – 260 с.
3. Булдаев А.С., Моржин О.В. Модификация метода проекций для улучшения нелинейных управлений // Вестник Бурятского государственного университета. – 2010. – Вып. 9: Математика и информатика. – С. 10-17.
4. Моржин О.В. Нелокальное улучшение управлений нелинейными дискретными системами // Программные системы: теория и приложения: электронный научный журнал. – 2010. – Т.1. – №1. – С. 21-44.
5. Васильев Ф.П. Методы оптимизации. – М.: Факториал-Пресс, 2002. – 824 с.
6. Васильев О.В., Аргучинцев А.В. Методы оптимизации в задачах и упражнениях. – М.: Физматлит, 1999. – 208 с.

Дрыганова Екатерина Вячеславовна, ассистент кафедры прикладной математики Бурятского государственного университета, тел. 89148312588, e-mail: dev8@mail.ru

Dryganova Ekaterina Vyacheslavovna, assistant at the chair of applied mathematics of the Buryat State University.

УДК 517.977

© Д.О. Трунин, Б. Очирбат, Д. Ганхуяг

ПРОЕКЦИОННАЯ ПРОЦЕДУРА НЕЛОКАЛЬНОГО УЛУЧШЕНИЯ УПРАВЛЯЕМЫХ СИСТЕМ С ФУНКЦИОНАЛЬНЫМИ ОГРАНИЧЕНИЯМИ¹

В статье предлагается процедура нелокального улучшения допустимых управлений в нелинейных по состоянию задачах оптимального управления с функциональными ограничениями на основе операции проектирования.

Ключевые слова: задача оптимального управления, нелокальное улучшение, функциональные ограничения, проекционная процедура.

© D.O. Trunin, B.Ochirbat, D. Gankhuyag

PROJECTING PROCEDURE OF NONLOCAL IMPROVING CONTROLLED SYSTEMS WITH FUNCTIONAL CONSTRAINTS

In the article a nonlocal improvement procedure of admissible control for nonlinear on a state optimal control problems with functional constraints based on projecting operation is proposed.

Keywords: optimal control problem, nonlocal improving, functional constraints, projecting procedure.

Введение

В [1] для полиномиальных по состоянию задач оптимального управления со свободным правым концом построены методы нелокального улучшения управлений, основанные на нестандартных формулах приращения функционала без остаточных членов разложений. Отсутствие операции варьирования управлений и возможность улучшения управлений, удовлетворяющих принципу максимума, обуславливают повышенную эффективность построенных методов. В работе [2] эти методы обобщены на класс нелинейных задач оптимального управления со свободным правым концом, а в [5] – на класс нелинейных по состоянию задач оптимального управления с функциональными ограничениями. В данной статье предлагается процедура нелокального улучшения допустимых управлений для нелинейных задач оптимального управления с дополнительными функциональными ограничениями на основе операции проектирования.

1. Постановка задачи

Рассматривается задача оптимального управления [3, 4] с функциональными ограничениями типа равенства

$$\dot{x} = f(x, u, t), \quad t \in T = [t_0, t_1], \quad (1)$$

$$x(t_0) = x^0, \quad u(t) \in U, \quad (2)$$

$$\Phi_0(u) \rightarrow \min, \quad (3)$$

$$\Phi_i(u) = 0, \quad i = \overline{1, s}. \quad (4)$$

Функционалы (3), (4) имеют вид

$$\Phi_i(u) = \varphi_i(x(t_1)) + \int_T F_i(x, u, t) dt, \quad i = \overline{0, s}.$$

Функция $f(x, u, t)$ и функции $F_i(x, u, t)$, $i = \overline{0, s}$ линейны по управлению.

Задачу (1)-(4) сведем к задаче оптимального управления с частично закрепленным правым концом [5]. Для этого преобразуем функциональные ограничения (4)

$$\Phi_i(u) = \varphi_i(x^0) + \int_T [\langle \varphi_{ix}(x), f(x, u, t) \rangle + F_i(x, u, t)] dt, \quad i = \overline{1, s}.$$

Вводя в рассмотрение дополнительные фазовые переменные

$$y_i(t) = \int_{t_0}^t [\langle \varphi_{ix}(x(\tau)), f(x(\tau), u(\tau), \tau) \rangle + F_i(x(\tau), u(\tau), \tau)] d\tau, \quad i = \overline{1, s},$$

¹ Работа выполнена при финансовой поддержке РФФИ (проекты 12-01-00914-а, 12-01-98011-р_сибирь_а, 13-01-92200-Монг_а)

дополним систему (1) уравнениями вида

$$\dot{y}_i = \langle \varphi_{ix}(x), f(x, u, t) \rangle + F_i(x, u, t), \quad i = \overline{1, s}, \quad (5)$$

начальные условия (2) – условиями

$$y_i(t_0) = 0, \quad i = \overline{1, s}. \quad (6)$$

Тогда ограничения (4) переписутся в виде

$$y_i(t_1) = -\varphi_i(x^0), \quad i = \overline{1, s}. \quad (7)$$

Нетрудно видеть, что задача (1)-(3), (5)-(7) является задачей оптимального управления с частично закрепленным правым концом.

Таким образом, далее будем рассматривать задачу оптимального управления с частично закрепленным правым концом

$$\dot{x} = f(x, u, t), \quad t \in T = [t_0, t_1], \quad (8)$$

$$x(t_0) = x^0, \quad u(t) \in U, \quad (9)$$

$$\Phi(u) = \varphi(x(t_1)) + \int_T F(x, u, t) dt \rightarrow \min, \quad (10)$$

$$x_i(t_1) = x_i^1, \quad i = \overline{1, m}, \quad m \leq n, \quad (11)$$

в которой $x = (x_1(t), x_2(t), \dots, x_n(t))$ – вектор состояния, $u = (u_1(t), u_2(t), \dots, u_r(t))$ – вектор управления, интервал T фиксирован, $x^0 \in R^n$ – заданный вектор, $x_i^1, i = \overline{1, m}$ – заданные числа, функция $\varphi(x)$ не зависит от первых m компонент вектора x ; U – выпуклое компактное множество в R^r . Функции $f(x, u, t)$, $\varphi(x)$, $F(x, u, t)$ непрерывно дифференцируемы по своим аргументам в областях определения, функции $f(x, u, t)$ и $F(x, u, t)$ линейны по u .

В качестве доступных управлений рассматривается множество кусочно-непрерывных функций со значениями в множестве $U \subset R^r$

$$V = \{u \in PC^r(T) : u(t) \in U, t \in T\}.$$

Для каждого доступного управления $u \in V$ обозначим $x(t, u), t \in T$ – решение задачи Коши (8), (9) при $u = u(t)$.

Определим множество допустимых управлений

$$W = \{u \in V : x_i(t_1, u) = x_i^1, i = \overline{1, m}\}.$$

В задаче (8)-(11) составим нормальный функционал Лагранжа

$$L(u, \lambda) = \Phi(u) + \sum_{i=1}^m \lambda_i (x_i(t_1) - x_i^1).$$

Функция Понтрягина с сопряженной переменной $p \in R^n$ имеет вид

$$H(p, x, u, t) = \langle p, f(x, u, t) \rangle - F(x, u, t).$$

В силу линейности по u функций $f(x, u, t)$ и $F(x, u, t)$ функция Понтрягина также линейна по управлению

$$H(p, x, u, t) = H_0(p, x, t) + \langle H_1(p, x, t), u \rangle.$$

Приращение функционала Лагранжа на паре доступных управлений (u^0, v) в соответствии с [2] имеет вид

$$L(v, \lambda) - L(u^0, \lambda) = - \int_T \langle H_1(p(t, u^0, v, \lambda), x(t, v), t), v(t) - u^0(t) \rangle dt, \quad (12)$$

где $p(t, u^0, v, \lambda)$ – решение модифицированной дифференциально-алгебраической сопряженной системы

$$\dot{p} = -H_x(p, x(t, u^0), u^0(t), t) - r(t), \quad (13)$$

$$\langle H_x(p, x(t, u^0), u^0(t), t), x(t, v) - x(t, u^0) \rangle + \langle r(t), x(t, v) - x(t, u^0) \rangle = \quad (14)$$

$$= H(p, x(t, v), u^0(t), t) - H(p, x(t, u^0), u^0(t), t), \quad (15)$$

$$p_i(t_1) = -\lambda_i, \quad i = \overline{1, m},$$

$$p_j(t_1) = -\varphi_{x_j}(x(t_1, u^0)) - q_j, \quad j = \overline{m+1, n}, \quad (16)$$

$$\begin{aligned} & \sum_{j=m+1}^n \left[\varphi_{x_j}(x(t_1, u^0)) (x_j(t_1, v) - x_j(t_1, u^0)) \right] + \\ & + \sum_{j=m+1}^n \left[q_j (x_j(t_1, v) - x_j(t_1, u^0)) \right] = \varphi(x(t_1, v)) - \varphi(x(t_1, u^0)). \end{aligned} \quad (17)$$

Для управления $u^0 \in V$ образуем аналогично [1, 4] вектор-функцию

$$u^\alpha(p, x, t) = P_U \left(u^0(t) + \alpha H_1(p, x, t) \right), \quad p \in R^n, \quad x \in R^n, \quad \alpha > 0,$$

где P_U – оператор проектирования на множество U в евклидовой норме.

Функция $u^\alpha(p, x, t)$ непрерывна по совокупности (p, x) на $R^n \times R^n$ и кусочно-непрерывна по $t \in T$, причем имеет место оценка [1, 4]

$$\langle H_1(p, x, t), u^\alpha(p, x, t) - u^0(t) \rangle \geq \frac{1}{\alpha} \|u^\alpha(p, x, t) - u^0(t)\|^2. \quad (18)$$

2. Процедура нелокального улучшения

Поставим задачу улучшения управления $u^0 \in W$: найти управление $v \in W$ со свойством $\Phi(v) \leq \Phi(u^0)$.

1. Для заданного $\alpha > 0$ найдем решение $(x(t), p(t))$, $t \in T$ дифференциально-алгебраической краевой задачи

$$\begin{aligned} \dot{x} &= f(x, u^\alpha(p, x, t), t), \quad t \in T, \\ \dot{p} &= -H_x(p, x(t, u^0), u^0(t), t) - r(t), \\ & \langle H_x(p, x(t, u^0), u^0(t), t), x - x(t, u^0) \rangle + \langle r(t), x - x(t, u^0) \rangle = \\ & = H(p, x, u^0(t), t) - H(p, x(t, u^0), u^0(t), t), \\ x(t_0) &= x^0, \quad x_i(t_1) = x_i^1, \quad i = \overline{1, m}, \\ p_j(t_1) &= -\varphi_{x_j}(x(t_1, u^0)) - q_j, \quad j = \overline{m+1, n}, \\ & \sum_{j=m+1}^n \left[\varphi_{x_j}(x(t_1, u^0)) (x_j(t_1) - x_j(t_1, u^0)) \right] + \\ & + \sum_{j=m+1}^n \left[q_j (x_j(t_1) - x_j(t_1, u^0)) \right] = \varphi(x(t_1)) - \varphi(x(t_1, u^0)). \end{aligned} \quad (19)$$

2. Сформируем управление $v(t) = u^\alpha(p(t), x(t), t)$, $t \in T$.

Предположим, что решение $(x(t), p(t))$, $t \in T$ краевой задачи (19) (возможно, не единственное) существует на T . Тогда $x(t) = x(t, v)$ и $v \in W$.

Покажем свойство улучшения для выходных управлений.

Действительно, решение $p(t)$, $t \in T$ является решением системы (13), (14) и удовлетворяет условиям (16), (17).

Обозначим

$$\bar{\lambda}_i = -p_i(t_1), \quad i = \overline{1, m}.$$

Тогда условия (15) выполняются и $p(t) = p(t, u^0, v, \bar{\lambda})$, $t \in T$.

Следовательно, в соответствии с формулой приращения (12) и оценкой (18) выходное управление v обеспечивает невозрастание функционала Лагранжа

$$L(v, \bar{\lambda}) \leq L(u^0, \bar{\lambda}).$$

Отсюда, в силу допустимости управлений u^0 , v получаем

$$\Phi(v) \leq \Phi(u^0).$$

Отметим, что в силу оценки (18) выходное управление обеспечивает строгое улучшение целевого функционала, если управления u^0 и v не совпадают.

Алгебраические соотношения краевой задачи (19) всегда можно разрешить по аналогии с [2] относительно величин $r(t)$, q_j и свести дифференциально-алгебраическую краевую задачу (19) к обычной дифференциальной задаче. Определяя различные однозначные способы разрешения алгебраических уравнений, можно получать модификации метода улучшения с различными дифференциальными краевыми задачами.

3. Пример

Рассмотрим задачу оптимального управления

$$\begin{aligned} \dot{x}_1 &= u, \quad \dot{x}_2 = -x_1^2, \quad t \in T = [0, \pi], \\ x_1(0) &= 0, \quad x_2(0) = 0, \quad |u(t)| \leq 1, \\ \Phi(u) &= x_2(\pi) \rightarrow \min, \quad x_1(\pi) = 0. \end{aligned}$$

В данном случае

$$H = p_1 u - p_2 x_1^2.$$

Поставим задачу улучшения допустимого управления $u^0(t) \equiv 0$, которому соответствуют фазовые траектории $x_1(t, u^0) \equiv 0$, $x_2(t, u^0) \equiv 0$, $t \in T$ и значение целевого функционала $\Phi(u^0) = 0$.

В данной задаче

$$u^\alpha(p, x, t) = \begin{cases} 1, & \alpha p_1 > 1, \\ -1, & \alpha p_1 < -1, \\ \alpha p_1, & -1 \leq \alpha p_1 \leq 1. \end{cases}$$

Таким образом, краевая задача улучшения принимает вид

$$\begin{aligned} \dot{x}_1 &= u^\alpha(p, x, t), \quad \dot{x}_2 = -x_1^2, \quad x_1(0) = 0, \quad x_2(0) = 0, \quad x_1(\pi) = 0, \\ \dot{p}_1 &= -r_1(t), \quad \dot{p}_2 = -r_2(t), \quad p_2(\pi) = -1, \\ r_1(t)x_1 + r_2(t)x_2 &= -p_2 x_1^2. \end{aligned}$$

Полагая $r_2(t) \equiv 0$ (тогда $p_2(t) \equiv -1$), получим краевую задачу

$$\begin{aligned} \dot{x}_1 &= u^\alpha(p, x, t), \quad x_1(0) = 0, \quad x_1(\pi) = 0, \\ \dot{p}_1 &= -x_1. \end{aligned}$$

Положим значение параметра $\alpha = 1$ и подберем решение, соответствующее условию $|p_1(t)| \leq 1$, $t \in T$.

Краевая задача улучшения примет вид

$$\begin{aligned} \dot{x}_1 &= p_1, \quad x_1(0) = 0, \quad x_1(\pi) = 0, \\ \dot{p}_1 &= -x_1. \end{aligned}$$

Очевидно, краевая задача имеет решения вида

$$x_1(t) = C \sin t, \quad p_1(t) = C \cos t, \quad t \in T,$$

где C – произвольная постоянная, $|C| \leq 1$.

Таким образом, например, допустимое управление

$$v(t) = \cos t, \quad t \in T$$

с соответствующими фазовыми траекториями

$$x_1(t, v) = \sin t, \quad x_2(t, v) = \frac{1}{4}(\sin 2t - 2t), \quad t \in T$$

строго улучшает исходное управление u^0 :

$$\Phi(v) = -\frac{\pi}{2} < \Phi(u^0) = 0.$$

Заключение

Предлагаемая процедура обеспечивает нелокальное улучшение допустимых управлений без

процедуры варьирования в окрестности улучшаемого управления с выполнением всех функциональных ограничений. Это свойство является существенным фактором повышения эффективности решения нелинейных задач оптимального управления с функциональными ограничениями.

Литература

1. Булдаев А.С. Методы возмущений в задачах улучшения и оптимизации управляемых систем. – Улан-Удэ: Изд-во Бурят. гос. ун-та, 2008. – 260 с.
2. Булдаев А.С., Моржин О.В. Улучшение управлений в нелинейных системах на основе краевых задач // Известия Иркутского государственного университета. – Сер. Математика. – 2009. – Т. 2. – № 1. – С. 94-106.
3. Васильев О.В. Лекции по методам оптимизации. – Иркутск: Изд-во Иркут. гос. ун-та, 1994. – 340 с.
4. Срочко В.А. Итерационные методы решения задач оптимального управления. – М.: Физматлит, 2000. – 160 с.
5. Трунин Д.О., Ганхуяг Д. Метод нелокального улучшения управляемых систем с функциональными ограничениями // Вестник Бурятского государственного университета. – Вып. 9(2). – Математика, информатика. – 2014. – С. 33-37.

Трунин Дмитрий Олегович, кандидат физико-математических наук, старший преподаватель кафедры прикладной математики Бурятского государственного университета, тел.: (301-2) 217733, e-mail: hint@rambler.ru

Очирбат Батор, Ph.D., профессор, Монгольский университет науки и технологий, Монголия, Улан-Батор.

Ганхуяг Данзан, Ph.D., профессор, Монгольский университет науки и технологий, Монголия, Улан-Батор, e-mail: gankhuyagd@gmail.com

Trunin Dmitry Olegovich, candidate of physical and mathematical sciences, senior lecturer, applied mathematics department, Buryat State University, e-mail: hint@rambler.ru

Ochirbat Baatar, Ph.D., professor, Mongolian University of Science and Technology, Mongolia, Ulanbaatar.

Gankhuyag Danzan, Ph.D., professor, Mongolian University of Science and Technology, Mongolia, Ulanbaatar, e-mail: gankhuyagd@gmail.com

2. Информационные системы и технологии

УДК 519.682

© А.С. Бурлаков, А.Е. Хмельнов

ЯЗЫК СПЕЦИФИКАЦИИ АРХИТЕКТУРЫ ЭВМ И ПЕРИФЕРИЙНЫХ УСТРОЙСТВ

В данной статье рассмотрен формальный язык, предназначенный для спецификации архитектуры ЭВМ и её оборудования, а также программное средство для отладки спецификаций, написанных на этом языке. Приведены некоторые синтаксические конструкции с пояснением их семантики, дан краткий обзор способов разбора текста. Разрабатываемый язык может найти применения для документирования особенностей архитектуры, а также в программах использующих тем или иным образом подобную информацию.

Ключевые слова: языки программирования, синтаксический анализ, интерпретаторы, эмуляторы ЭВМ.

© A.S. Burlakov, A.E. Hmelnov

THE LANGUAGE OF SPECIFICATION OF COMPUTER ARCHITECTURE AND PERIPHERAL DEVICES

The paper observes the formal language designed for computers architecture and it's peripherals specification, it also observes the software developed to debug specifications written in this language. Syntactic construction are brought as an example and their semantic is explained and parsing methods are briefly reviewed. The language developed may be applied for documenting of architecture particularities as well as in the programs someway using information of computer architecture.

Keywords: programming languages, parsing, interpreters, computer emulators.

Введение

В технической литературе, немало книг посвящено описанию различного оборудования ЭВМ, в главной степени описанию архитектуры процессора. Полное описание архитектуры x86-64 приводится на 3439ти страницах трех томов документации Intel [1]. Отдельный сайт посвящен описанию архитектуры ARM процессоров [2]. Учебники и справочники языка assembler для конкретной архитектуры часто содержат её подробное описание.

Многие программы, такие как ассемблеры, дизассемблеры, компиляторы, эмуляторы ЭВМ и даже драйверы устройств используют информацию об архитектуре компьютера. В отличие от технической документации, в программах описание архитектуры входит в состав исходного кода и составлено на некотором языке программирование.

В данной статье рассматривается язык спецификации архитектуры ЭВМ и периферийного оборудования, который может быть применён как для документирования, так и для формального представления и использования в программах. Это обеспечивается за счет того, что язык является высокоуровневым и простым для понимания человеком, но в тоже время однозначно интерпретируемым машиной.

При создании языка был учтен опыт авторов составляющих учебники и справочники посвященные описанию архитектуры ЭВМ [3-6]. В особенности был использован источник [3], где автор посвятил отдельную главу проблеме описания архитектуры ЭВМ.

Для спецификации машинных команд были использованы наработки авторов из источника [7]. В отличие от подхода, приведенного авторами в [7], наш язык не ограничивается описанием операндов процессорных команд, а позволяет целиком описывать архитектуру компьютера.

Кроме самого языка, в статье большое внимание уделено отладке написанных на этом языке спецификаций, для которой было разработано программное средство, способное понимать конструкции языка и на их основе эмулировать работу соответствующей архитектуры.

1. Синтаксис и семантика языка

Язык спецификации оборудования ЭВМ удовлетворяет парадигме объектно-ориентированного программирования – он поддерживает инкапсуляцию, полиморфизм и единичное наследование.

Основной структурной (не синтаксической) единицей языка является класс, реализующий логику работы соответствующего устройства. Взаимодействие программы и текста описания основано на событийном программировании. Пример приведен ниже, в разделе посвященном скелету эмулятора и виртуальной машине. За основу синтаксиса языка спецификации был взят язык программирования C++ ISO/IEC с некоторыми заимствованиями из языка C++/CLI и других языков программирования.

Кроме стандартных лексических конструкций свойственных большинству императивных языков программирования, в языке спецификации реализован декларативный оператор для описания команд процессора, операторы для работы с частью переменной (оператор извлечения последовательности байт и бит), оператор для работы с несколькими переменными (оператор конкатенации) и другие [3,8].

В силу того, что язык спецификации является интерпретируемым, появляется возможность использования ссылок, указывающих не на конкретную область памяти (как в случае использования компилируемых языков вроде C++), а на динамические конструкции вроде l-value выражений [9].

Одним из возможных терминалов l-value выражения является переменная, которая может быть либо скалярной числовой переменной, либо свойством объекта (property), которое обладает методами записи и чтения. Другим из возможных терминалов l-value выражения может быть функция, которая возвращает ссылочный тип данных. Проверка корректности l-value выражений происходит во время и после разбора текста описания. Нетерминалами l-value выражения являются операторы конкатенации, извлечения последовательности байт и бит.

```
lvalue_expr: nested_id          // Переменная либо св-во
| nested_id '(' expr_list ')' // Метод
| lvalue_expr '[' expr_num ']' // Извл-е одного байта
// Извлечение последовательности байт
| lvalue_expr '[' expr_num PERIOD expr_num ']'
| lvalue_expr '<' expr_num '>' // Извл-е одного бита
// Извлечение последовательности бит
| lvalue_expr '<' expr_num PERIOD expr_num '>'
| lvalue_expr ':' lvalue_expr // Конкатенация
;
```

Рисунок 1. Синтаксис l-value выражения

Рассмотрим пример использования l-value выражения.

```
bits<32> A = 0xffffffff;
bits<1> CF = 0;
CF:A<16..0> = 0x1cdcd; // CF = 1, A = 0xffffcdcd
```

Рисунок 2. Пример использования l-value выражения

На рисунке 2, в первых двух строках объявляются переменные A и CF длиной 32 и 1 бит соответственно. В третьей строке происходит операция присваивания, где первые 16 разрядов числа 0x1cdcd будут записаны в переменную A, а 17ый разряд будет записан во флаг CF. Таким образом, после выполнения третьей строки, переменные A и CF изменят свои значения на 0xffffcdcd и 1 соответственно.

Рассмотрим другой пример объявления подпрограмм (функций) и использования ссылочного типа данных.

```
bits<?>& R(bits<3> code, bits<1> W = 1)
{
  if(W)
    case(code){
      maskb      0:  0:  0 then return EAX;
      maskb      0:  0:  1 then return ECX;
      maskb      0:  1:  0 then return EDX;
      maskb      0:  1:  1 then return EBX;
      maskb      1:  0:  0 then return ESP;
      maskb      1:  0:  1 then return EBP;
      maskb      1:  1:  0 then return ESI;
      maskb      1:  1:  1 then return EDI;
    }
}
```

```

else
  case(code){
    maskb      0:  0:  0 then return AX;
    maskb      0:  0:  1 then return CX;
    maskb      0:  1:  0 then return DX;
    maskb      0:  1:  1 then return BX;
    maskb      1:  0:  0 then return SP;
    maskb      1:  0:  1 then return BP;
    maskb      1:  1:  0 then return SI;
    maskb      1:  1:  1 then return DI;
  }
};

void MOV(bits<?>& RegD, bits<?>& RegS )
{
  RegD = RegS;
  ...
}
...
MOV(R(0), R(1));

```

Рисунок 3. Пример объявления функции и использования ссылок

На рисунке 3 объявлены две функции: функция MOV, которая принимает на вход ссылки на регистр-источник и регистр-приемник, и функция R, возвращающая ссылку на соответствующий регистр выбираемый оператором case в зависимости от кода регистра. Например, вызов MOV(R(0), R(1)) в итоге выполнит присваивание EAX = ECX.

Оператор case по синтаксису и семантике напоминает одноименный оператор языка Transact SQL. В отличие от оператора множественного ветвления switch языка C++, case способен работать не только с константными значениями, но и с переменными, выражениями и битовыми масками. Последняя возможность является особенно удобной при описании формата команд процессора (см. Рисунок 4).

```

bits<3> Ns, Nd; // Номер регистра источника и приемника
bits<1> W;      // ? dword: word
unsigned short opcode = m_memory.ReadByte(EIP);
case(opcode)
{
  maskb 1:0:0:0:1:0:1:W:1:1:Nd:Ns then
    MOV(R(Ns,W), R(Nd,W));
    break;

  maskb 0:1:0:0:Nd:0           then
    INC(R(Nd));
    break;
}

```

Рисунок 4. Использование оператора case на примере описания формата команд MOV и INC (см. Рисунок 2) процессора архитектуры IA-32 [1]

В декларативном смысле, оператор case – это n-местный предикат, где n – это число вхождений переменных в конкатенацию заключенную между ключевыми словами maskb и then. (Для восьмеричной или шестнадцатеричной записи, используются masko или maskh соответственно, а сами числа записываются без префиксов 0 или 0x.) Предикат возвращает истину, если существует такое значение переменных входящих в конкатенацию, при которых она совпадает с аргументом оператора case. В этом случае, в аргументы входящие в конкатенацию записываются соответствующие значения, и далее выполняется последовательность инструкций, следующая за ключевым словом then. Последовательность инструкций при этом может носить императивный характер, и если она заканчивается оператором break или return, то проверка следующего предиката не происходит, и все последующие конструкции maskb ... then внутри текущего оператора case игнорируются.

2. Разбор текста спецификации

Для разбора текста спецификации используется синтаксический анализатор, сгенерированный программой Bison в совокупности с лексическим анализатором, полученным программой Flex. Поддерживается модульность на уровне файлов и проверка синтаксиса на этапе разбора текста. В силу того, что в синтаксисе языка присутствуют выражения, которые невозможно однозначно свернуть, используя только один предиктивный символ, было решено отказаться от алгоритма LALR(1) в пользу GLR. Основная разница этих алгоритмов заключается в том, что алгоритм GLR в отличие от алгоритма LALR(1) может выполнять откат более чем на один уровень при построении дерева разбора, в случае несовпадения продукции [10-12].

3. Структура программного средства для отладки языка

Для отладки языка спецификации разработано программное средство, способное понимать текст описания и эмулировать работу соответствующего оборудования. Суть отладки заключается в наблюдении за состоянием параметров системы в различные моменты времени. Общая архитектура системы представлена на схеме ниже.

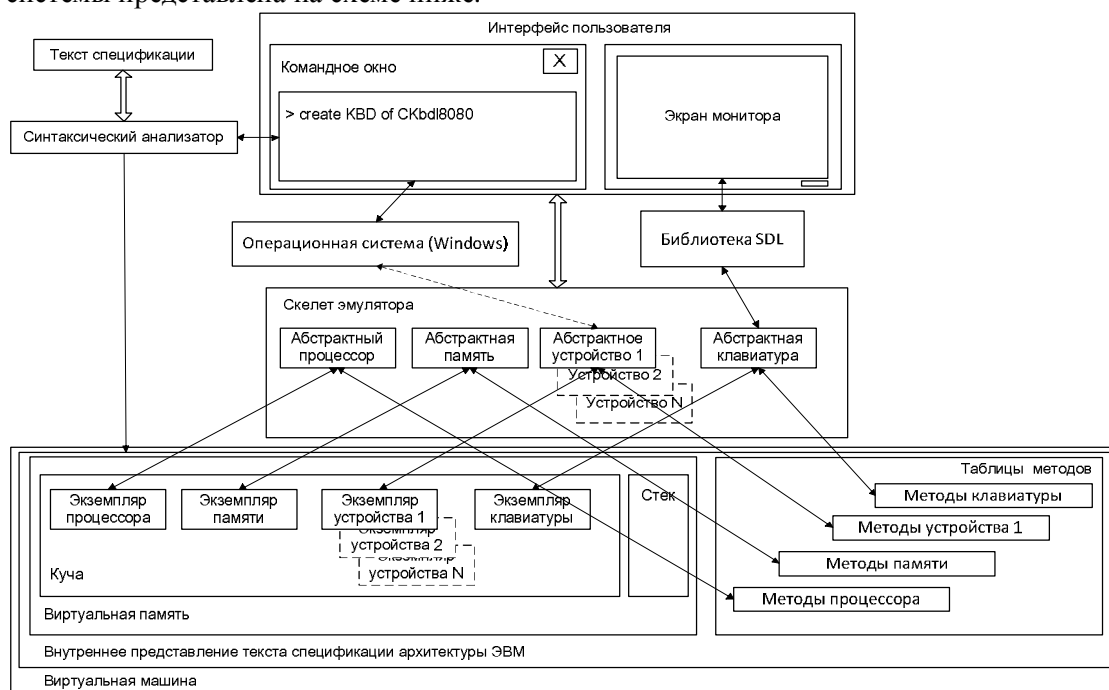


Рисунок 5. Схема программы

В состав программного средства (далее эмулятор) входят: пользовательский интерфейс, скелет эмулятора, виртуальная машина и синтаксический анализатор. Текст спецификации располагается в отдельном файле. Эмулятор работает следующим образом: после запуска приложения пользователь выбирает виртуальное оборудование, которое необходимо создать, после чего происходит разбор текста спецификации запрашиваемого оборудования во внутреннее представление, загружается образ памяти и начинается выполнение эмуляции.

Для взаимодействия пользователя с эмулятором используется двухоконный пользовательский интерфейс. Первое окно является консольным, создается средствами операционной системы Windows и предназначено для конфигурирования и управления эмулятором. Второе окно является графическим, создается средствами библиотеки SDL и представляет собой экран виртуального графического устройства, например VGA монитора. Обработка сообщений окна, например нажатие клавиши или щелчок мыши реализуется за счет использования соответствующей библиотеки (Win API либо SDL) [13].

Скелет эмулятора – это абстрактная реализация архитектуры фон-Неймана. Другими словами – это ЭВМ, которая состоит из стандартных устройств: процессора, памяти и устройств ввода-вывода, алгоритм работы которых определены лишь до некоторой степени. Назовем эти устройства абстрактными. Для того чтобы заставить функционировать абстрактное устройство надлежа-

шим образом, необходимо доопределить логику его работы в отдельном текстовом файле на языке спецификации.

Взаимодействие текста описания со скелетом эмулятора основано на событийном программировании. Для того, чтобы доопределить логику работы устройства, необходимо создать класс устройства, который будет полностью удовлетворять требованиям интерфейса, а именно он должен реализовывать все виртуальные методы устройства. Если метод устройства не реализован, то вместо него может быть использован стандартный метод абстрактного устройства, либо, если такой отсутствует, приложение вернет ошибку.

Рассмотрим пример. Абстрактное устройство клавиатуры способно получать сообщения библиотеки SDL о нажатии клавиши и её код. То, каким образом интерпретировать данное событие, определено в файле спецификации (см. Рисунок 6).

```
class CKbdI8080 : public CKbd
{
    ...
    void KeyDown(bits<32> code)
    {
        ...
    }
    void Compile(CComputer* pc){};
}
```

Рисунок 6. Фрагмент описания устройства типа клавиатура

Получив сообщение о нажатии клавиши, абстрактное устройство клавиатуры вызовет метод `KeyDown` передав в параметре её код. После завершения работы метода `KeyDown`, управление будет возвращено абстрактной клавиатуре. Сама абстрактная клавиатура не способна напрямую работать с текстом описания, для этого к ней должен быть привязан экземпляр класса `CKbdI8080`, созданный внутри виртуальной машины. Для корректной работы, класс `CKbdI8080`, как уже было сказано должен реализовать все методы абстрактной клавиатуры, регламентируемые интерфейсом.

Для исполнения кода методов используется виртуальная машина. Состояние исполнения запоминается в памяти, которая состоит из кучи и стека. Аналогично тому, как это делают компиляторы языков программирования вроде Delphi и C++, в стеке интерпретатора располагаются локальные переменные, объявленные в тексте спецификации оборудования, а в куче – глобальные переменные, которые являются экземплярами классов описывающих это оборудование. Так, экземпляр класса `CKbdI8080` будет располагаться именно в куче.

Грубо говоря, привязка экземпляра устройства – это передача абстрактному устройству двух указателей, один из которых указывает на начало области памяти в куче, где располагается экземпляр класса устройства, а другой – указатель таблицы методов этого класса. Метод содержит в себе дерево инструкций. Выполнение метода - это обхода его дерева и изменение состояния виртуальной памяти.

Виртуальная машина реализована по принципу абстрактной стековой машины. Перед вызовом подпрограммы, вызывающая программа помещает аргументы на стек в порядке справа налево, таким образом, что указатель `this` всегда находится по определённому смещению. Вызываемая подпрограмма сохраняет фрейм стека, а после выполнения аргументы удаляются со стека, и на вершину помещается результат соответствующей длины.

Аналогично клавиатуре, происходит взаимодействие других устройств эмулятора. Список создаваемых устройств может быть перечислен в файле сценария эмуляции, либо введен в главное окно приложения (см. Рисунок 7).

```
create PC of CComputerI8080
create CPU of CCpuI8080
create RAM of CMemI8080
create KBD of CKbdI8080
...
compile
load mon32.bin into RAM
start
```

Рисунок 7. Пример конфигурирования эмулятора

В данном примере в первых четырех строках сценария эмуляции, виртуальная машина создает экземпляры классов `CComputerI8080`, `CSpuI8080`, `CMemI8080` и `CKbdI8080` и привязывает эти экземпляры к соответствующим абстрактным устройствам, за каждым из которых зарезервировано имя: `PC`, `CPU`, `RAM`, `KBD` и т.п. для обозначения компьютера, центрального процессора, оперативной памяти и клавиатуры соответственно. Выполнение команды `compile` сопровождается вызовом виртуального метода `Compile` для каждого созданного объекта (см. Рисунок 6). Команда `load` вызывает одноименный метод для объекта, закрепленного за устройством `RAM`, передавая в него файл образа памяти. Последняя команда запускает выполнение эмуляции.

Имена команд не чувствительны к регистру, однако некоторые операнды команд, например имя класса – чувствительны. Все команды можно разбить на три группы: команды создания и конфигурирования оборудования, команды управления эмулятором и вспомогательные команды. Важно отметить, что все команды не являются частью языка спецификации и используются только в сценарии эмуляции.

Цель разделения одного устройства на две сущности заключается в перенесении части логики работы устройства из текста описания в программу, что обеспечивает баланс между удобством и гибкостью настройки программы. Так, неизменяемые и очевидные алгоритмы уже запрограммированы в самом эмуляторе, в соответствующих абстрактных устройствах, а все тонкости реализации, которые варьируются в зависимости от модели устройств, описаны в файле спецификации.

Заключение

В статье были рассмотрены вопросы, связанные с разработкой языка спецификации архитектуры ЭВМ и периферийных устройств, такие как: выбор синтаксиса, семантика языковых конструкций, способы отладки языка. Текст описания архитектуры является простым для понимания специалистом и в тоже время формальным, для использования в компьютерной программе. Таким образом, язык описания архитектуры ЭВМ может найти применение в программах, использующих информацию об архитектуре компьютера, например в ассемблерах, дизассемблерах, компиляторах, эмуляторах и тд.

Помимо самого языка, уделено внимание программному средству для отладки спецификаций составленных на этом языке. На данный момент были проведены успешные испытания работоспособности описания архитектур `I8080` и `PDP11`, происходит разработка описания реального режима процессора архитектуры `IA-32`.

Литература

1. Intel® 64 and IA-32 Architectures Software Developer's Manual Combined Volumes: 1, 2A, 2B, 2C, 3A, 3B and 3C.
2. ARM Information Center, <http://infocenter.arm.com/>
3. Бруснецов Н.П. Микрокомпьютеры. – М.: Наука, Гл. ред. Физ.-мат. лит., 1985. – С. 34-64.
4. Майерс Г. Архитектура современных ЭВМ: В 2-кн. Кн. 1. Пер. с англ. – М.: Мир, 1985. С. 10-15.
5. Брамм П., Брамм Д. Микропроцессор 80386 и его программирование: Пер. с английского. – М.: Мир, 1990. – 448 с.
6. Таненбаум Э. Архитектура компьютера. 5-е изд. – СПб.: Питер, 2013. – 844 с.
7. Ramsey N., Fernandez M. "Specifying Representation of Machine Instructions", ACM Trans. Program. Lang. Syst. Vol 19, No 3. 1997, pp. 492-524.
8. Бурлаков А.С. Описание семантики машинных команд // Материалы Всерос. молодежной науч.-практ. конф. с международ. участ. (Иркутск, 21-23 марта, 2013 г.). – Иркутск: Изд-во ИрГТУ, 2013. – С. 137-141.
9. Ахо А., Сети Р., Ульман Д. Компиляторы: принципы, технологии и инструменты: Пер. с англ. – М.: Издательский дом "Вильямс", 2003. – С. 376-424.
10. Donald E. Knuth. "On the Translation of Languages from Left to Right", Information and control 1965, Vol 8, pp. 608-639.
11. Фридл Дж. Регулярные выражения. Библиотека программиста. – СПб.: Питер, 2001. – С. 200-218.
12. Bison – GNU parser generator, <http://www.gnu.org/software/bison/>
13. Simple DirectMedia Layer – Homepage, <http://www.libsdl.org/index.php>

Бурлаков Алексей Сергеевич, старший преподаватель кафедры информационных технологий Иркутского государственного университета (664003, г. Иркутск, ул. Карла Маркса, 1), тел. (3952) 453112, E-mail: alex_burlakov@mail.ru

Хмельнов Алексей Евгеньевич, кандидат технических наук, старший научный сотрудник института динамики систем и теории управления СО РАН (664033, г. Иркутск, ул. Лермонтова, 134, а/я 292), тел. (395-2) 453071, E-mail hmelnov@icc.ru

Burlakov Aleksey Sergeevich, senior teacher of department of Informational Technologies of Irkutsk State University (664003, Irkutsk, Karla Marksa str, 1).

Hmelnov Aleksey Evgenjevich, candidate of technical science, senior scientist of Institute for System Dynamics and Control Theory of Siberian Branch of Russian Academy of Sciences (664033, Irkutsk, Lermontova str, 134, POB 292).

УДК 681.3.06+37.01:014.544+629.7.066

© Г.С. Курганская

САМООРГАНИЗАЦИЯ ИНТЕРНЕТ-ОБРАЗОВАНИЯ В ОБЛАЧНЫХ ТЕХНОЛОГИЯХ

В статье представлена точка зрения автора на архитектуру платформы ГЕКАДЕМ облачного Интернет-образования, которая базируется на принципах самоорганизации как процесса формирования кластеров знаний, так и их использования при Интернет-образовании на основе облачных технологий. Рассматриваются возможности мультиагентного подхода, как базовой модели архитектуры платформы и приводятся перечень ключевых агентов функциональных подсистем.

Ключевые слова: представление знаний, информационные технологии, Интернет-обучение, самоорганизация сложных систем, облачные технологии, мультиагентные системы

© G.S. Kurganskaya

SELF-ORGANIZATION OF INTERNET LEARNING IN CLOUD TECHNOLOGIES

Multi-agent approach up to cloud-based Internet Learning self-organization has been considered. Principles of self-organization of complicated dynamic systems and possibilities of multi-agent approach have been discussed. Some main ideas of architecture of multi-agent cloud-based Internet learning intellectual platform HECADEM 5.0 are described.

Keywords: knowledge representation, information technology, Internet learning, self-organization of complex system, multi-agent system, cloud-based technology.

Введение

Стремительные изменения происходят во всех сферах деятельности человека, в том числе в сфере экономики, науки и образования. На наш взгляд, основным двигателем этих изменений является постоянное появление новых средств инфо-телекоммуникационных технологий, которые реализуют сущность информации – нарушение закона сохранения. Несомненно, облачные технологии являются серьезным шагом в этом направлении, поскольку предлагают открытый доступ к распределенной обработке и хранению данных. Использование облачных технологий в образовании формирует принципиально новые возможности. Облако Интернет-обучения состоит из огромного числа относительно автономных интеллектуальных ресурсов – кластеров знаний и независимых пользователей, число которых может иметь порядок тысяч, сотен тысяч или даже более. Более того, и ресурсы, и пользователи распределены в пространстве, образуя сетевую структуру, каждый пользователь имеет собственную цель, которая не должна согласовываться с глобальной целью всей системы, но любой пользователь должен получить доступ к необходимым ресурсам и может быть вынужден или заинтересован во взаимодействии с другими пользователями. При этом отдельные пользователи и ресурсы обладают весьма ограниченной информацией о других компонентах облака, в том числе их целях и возможностях, и они ничего не знают о топологии системы в целом. Более того, само облако функционирует в Интернет, т.е. в открытой динамической внешней среде, и при этом каждый компонент системы может воспринимать только собственное, весьма ограниченное локальное окружение. Говорить об оптимальном централизованном управлении нереально не только из-за огромной размерности задачи управления, непредсказуемости динамики ее структуры, динамики внешней среды, но также и по организационным соображениям, поскольку в облаке и не должно быть единого центра управления. На наш взгляд, ответом на такой вызов является механизм самоорганизации. Этот термин в науке встречается еще в работах Р. Декарта, но активно этот подход стал активно развиваться в прошлом веке в рамках искусственного интеллекта, одной из таких работ был проект СОЧМСАНИ (Самоорганизующаяся ЧеловекоМашинная Система Автоматизации Научных Исследований), разрабатываемый под руководством академика В.М. Матросова в 80-х годах ИрВЦ СО АН, ныне ИДСТУ РАН.

1. Основы самоорганизации сложных систем

Рассмотрим основные принципы самоорганизации [1]:

- Автономность – взаимодействие с внешним миром допустимо, но управление извне недопустимо.
- Глобальный порядок – он возникает в системе только благодаря локальным взаимодействиям ее компонент.
- Эмерджентность – свойства, которые не могут быть выведены из свойств автономных сущностей; они проявляются только на верхнем уровне.
- Адаптивность – способность системы изменять свое поведение и переходить в новое устойчивое состояние со сменой организационной структуры.
- Нелинейная динамика, неустойчивость и чувствительность к вариациям начального состояния и малым вариациям параметров системы;
- Множественность устойчивых состояний системы (аттракторов).
- Сложность – она вызывается тем, что такая система, как правило, состоит из огромного числа взаимодействующих компонент.
- Простота правил, которые используются компонентами системы в процессах взаимодействия; но в итоге это приводит к сложному поведению системы в целом
- Иерархическая структура – самоорганизующаяся система описывается, по крайней мере, на двух уровнях, а именно, на уровне локальных компонент системы и на метауровне, где проявляются эмерджентные свойства системы.

2. Самоорганизация Интернет-облака платформы ГЕКАДЕМ 5.0

В настоящее время предлагаются различные подходы к эффективной реализации принципов самоорганизации, в том числе и разрабатываемая в Иркутском государственном университете на основе KFS модели представления знаний интеллектуальная платформа ГЕКАДЕМ 5.0 [2]. В рамках этого подхода образование рассматривается как информационное взаимодействие, при котором параллельно происходят как процессы формирования кластеров знаний, их динамической структуризации, так и открытого доступа к ним.

В разрабатываемой версии рассматриваются следующие уровни кластеров знаний, которые могут быть ресурсами в Интернет-облаке образования:

- *Базовые элементы*: отдельный учебный материал в любой форме; процедура проверки знаний: индивидуальное задание, тест любого формата и т.п.; площадки совместной работы: групповой проект, семинар, вебинар, wiki-проект и др.
- *Учебный блок* (тема) формируются из базовых элементов, он является основой KFS модели представления знаний и не может быть пустым. Блок формирует некое новое знание из входных знаний, которые могут формироваться в других блоках.

Конечная совокупность учебных блоков, на которой задано такое отношение частичного порядка, образует *учебный курс*, а учебные курсы, для которых устанавливается такой же частичный порядок, в свою очередь, могут сформировать *учебную программу*. Понятно, что можно как детализировать, так и проводить дальнейшие обобщения.

Таким образом, ресурсы Интернет-облака обучения представляют граф сложной динамической структуры, необязательно связный. В состав ресурсов также входит свободное программное обеспечение, необходимое для работы отдельных интеллектуальных ресурсов.

Пользователями облака могут быть как отдельные люди, так и группы людей, в том числе организации и компании.

Таким образом, именно принципы самоорганизации адекватны динамической структуре взаимодействующих процессов формирования и освоения знаний в Интернет-облаке ГЕКАДЕМ 5.0.

3. Мультиагентная Интернет-платформа ГЕКАДЕМ 5.0

В настоящее время в качестве технологически реализуемого решения, которое может позволить преодолеть трудности практического создания приложений для реализации механизмов самоорганизации сложных динамических систем является архитектура и технология мультиагентных систем.

Рассмотрим возможности мультиагентного подхода:

- число агентов может исчисляться миллионами и динамически изменяется;
- число соседей – сотни;
- локальность взаимодействий – каждый узел “знает” только своих соседей и взаимодействует только с ними (peer-to-peer взаимодействие);
 - адаптация и смена структуры – локальный выбор новых соседей (кто о ком должен знать, чтобы сеть решала свою задачу оптимально), самоконфигурация сети.
 - эмерджентность,
 - формирование паттернов– возникновение альянсов и коалиций.

Таким образом, основные требования самоорганизации Интернет–обучения полностью обеспечиваются при мультиагентных подходе:

- автономность – способность управлять своим целенаправленным поведением без внешнего вмешательства;
- способность воспринимать внешнюю среду и воздействовать на нее;
- наличие инфраструктуры, поддерживающей распределенное взаимодействие автономных сущностей (агентов) или физически (через коммуникации), или виртуально, т.е. путем обмена сообщениями, в том числе – на уровне парных взаимодействий.

Рассмотрим, как мультиагентный подход будет использован в архитектуре интеллектуальной Интернет-платформы обучения ГЕКАДЕМ 5.0. В состав самоорганизующейся мультиагентной системы Интернет-обучения будут входить программные интеллектуальные агенты, представляющие как интересы всех групп пользователей, так и сами интеллектуальные ресурсы.

Агенты, представляющие интересы пользователей, решают задачи, специфические для каждой группы пользователей, ниже приводятся некоторые из них

Разработчики интеллектуальных ресурсов

- поиск необходимой информации в Интернет в целом;
- поиск подходящих информационных ресурсов в облаке ГЕКАДЕМ;
- проверка на оригинальность (плагиат);
- анализ востребованности разработанных интеллектуальных ресурсов с различных точек зрения;
- и др.

Студенты

- планирование индивидуального пошагового пути изучения на разных уровнях;
- поиск подходящих информационных ресурсов в облаке ГЕКАДЕМ;
- поиск необходимой информации в Интернет в целом;
- организация совместной работы на соответствующих площадках;
- анализ личного успеха;
- и др.

Модераторы (Тьюторы)

- отслеживание исполнения процедур проверки знаний;
- анализ результатов процедур проверки знаний;
- мониторинг исполнения графиков изучения и индивидуальных планов;
- организация совместной работы на соответствующих площадках;
- и др.

Аналитики

- анализ востребованности разработанных интеллектуальных ресурсов с различных точек зрения;
- проверка на оригинальность (плагиат) разработанных интеллектуальных ресурсов;
- выявление потребностей в новых интеллектуальных ресурсах разных уровней;
- и др.

Очевидно, что для решения этих задач необходимо использовать интеллектуальные агенты разных типов: по мониторингу и наблюдению, по добыче и анализу данных, а также персональные агенты.

Условия работы интеллектуальных агентов Интернет-платформы обучения ГЕКАДЕМ:

- Запросы поступают на выполнение тех или иных работ поступают случайно.
- Запросы имеет разную трудоемкость.

▪ Невозможно предсказать, какие интеллектуальные ресурсы потребуются для их исполнения.

▪ Интеллектуальные ресурсы динамичны по своей природе: они могут появляться, модифицироваться и прекращать свое существование в любое время.

Рассмотрим агенты, представляющие интересы интеллектуальных ресурсов облака Интернет-обучения ГЕКАДЕМ.

Каждому ресурсу ставится в соответствие агент, основной задачей которого является изменение состояния соответствующего объекта. Состояние объекта может включать следующие характеристики:

Базовые

- тип объекта;
- время создания;
- формат;
- IP адрес;
- и др.

Структурные

- родители;
- дети;
- соседи;
- и др.

Рыночные

- рейтинг;
- цена;
- и др.

Предполагается, что при взаимодействии агентов в основном будут передаваться маркеры, хотя возможен и обмен семантическими сообщениями. Информационный поток от агента обычно не имеет конкретного адресата, а размещается в общем информационном поле, в котором другие агенты будут искать информацию, и в случае успеха выполнять некоторые действия.

Для самоорганизации функционирования ресурсных агентов будут в основном использоваться механизм прямого воздействия, а для самоорганизации процесса разработки интеллектуального ресурса будут также применяться и механизмы косвенного взаимодействия, в том числе распределенного обучения с подкреплением.

Наиболее сложным является процесс обучения в облаках, и поэтому самоорганизация именно этого процесса принципиально важна. Автор полагает, что здесь наиболее эффективным будет рыночный механизм самоорганизации. При этом следует обратить внимание, что при реализации этого механизма агенту помимо информации от ближайших соседей может потребоваться также параметры, характеризующие всю систему - например, средняя, цена ресурса такого типа, что противоречит принципу локальности взаимодействий при мультиагентном подходе. При этом эти глобальные характеристики (эмерджентности) тоже должны быть динамичными и адекватно отражать текущее состояние на рынке интеллектуальных ресурсов в облаке. Содержательно рыночная модель самоорганизации может описываться как аукцион, в котором группы агентов ведут переговоры с целью продажи/покупки конкретных ресурсов. Разные алгоритмы проведения аукционов в основном базируются на методах теории игр.

Интересной формой рыночного механизма является сеть потребностей и возможностей, разработанная в Самарской научной школе [3]. В соответствии с концепцией ПВ-сети, прикладная МАС строится из большого числа крайне простых автономных агентов, которые способны воспринимать ситуацию на рынке, принимать решения и взаимодействовать с другими близкими агентами. В рассматриваемой модели потребности в ресурсах того или иного типа возникают динамически (по мере поступления заказов), а ресурсы используются и освобождаются также динамически. В модели ПВ-сети под каждый заказ на ресурс генерируется агент потребности, который нацелен на поиск ресурса с нужными свойствами за наименьшую цену. Аналогично, для каждого ресурса генерируется агент возможности, который ответственен за наилучшее использование ресурса, например, получение большей платы за ресурс. Предполагается, что на рынке интеллектуальных ресурсов Интернет-облака присутствует одновременно множество агентов потребностей, занимающихся поиском ресурсов, и агентов возможностей, которые предлагают ресурсы. Рынок в модели ПВ-сетей обеспечивает "встречи" агентов потребностей и

агентов возможностей и реализацию для любой такой пары агентов операции "мэтчинга". Суть этой операции состоит в том, что агенты сравнивают предлагаемый и потребный сервисы, и если предлагаемый ресурс имеет соответствующий тип и удовлетворяет требованиям по качеству, то агент потребностей может "зарезервировать" этот сервис, либо, если найденный ресурс по каким-то характеристикам лучше, чем подобный, уже зарезервированный, то начинаются «переговоры», в которых агент потребности пытается отказаться от ранее зарезервированного ресурса. Поведение агента возможностей аналогично, если он встречает агента потребностей, который готов заплатить ему больше, то точно также может начаться цепочка переговоров для заключения новой сделки без ущерба для агента возможностей.

Заключение

Автор считает, что новым является подход к организации Интернет-обучения на основе облачных технологий, который базируется на принципах самоорганизации мультиагентных интеллектуальных систем. Предлагаемая архитектура мультиагентной платформы Интернет-обучения ГЕКАДЕМ 5.0 позволит реализовать миссию Интернет-обучения «От каждого по способностям, каждому по потребностям».

Литература

1. Городецкий В.И. Самоорганизация и многоагентные системы. Модели многоагентной самоорганизации // Известия РАН "Теория и системы управления". – 2012. – № 2. – С. 92-120.
2. Курганская Г.С. Интернет–образование на основе технологии облачных вычислений// Информационные и математические технологии в науке и управлении / Труды XVII Байкальской Всероссийской конференции «Информационные и математические технологии в науке и управлении». Часть III. – Иркутск: ИСЭМ СО РАН, 2012. – С. 250-255.
3. Виттих В.А., Скобелев П.О. Мультиагентные модели взаимодействия для построения сетей потребностей и возможностей в открытых системах. // Автоматика и Телемеханика. – 2003. – №1. – С. 177-185.

Курганская Галина Сергеевна, доктор физико-математических наук, зав. кафедрой ИТ в управлении Иркутского государственного университета, тел. +79148833079, E-mail: galina@buk.irk.ru

Kurganskaya Galina Sergeevna, PhD in Mathematics, Head of IT department, Irkutsk State University.

УДК 004.431.2

© А.А. Михайлов

ПРОМЕЖУТОЧНОЕ ПРЕДСТАВЛЕНИЕ ПОДПРОГРАММ В ЗАДАЧЕ ДЕКОМПИЛЯЦИИ ОБЪЕКТНЫХ ФАЙЛОВ DCUIL

В работе рассматривается одна из задач декомпиляции – генерация промежуточного представления программы. Описаны некоторые оптимизации над представлением, нацеленные на улучшение качества генерируемого кода. Также приведена архитектура разработанного декомпилятора *DCUIL2PAS*.

Ключевые слова: декомпиляция, CIL, dcuil.

© А.А. Mikhailov

THE INTERMEDIATE REPRESENTATION OF SUBROUTINES IN THE TASK OF DECOMPILATION OF THE OBJECT FILES DCUIL

In paper considered one of decompiling tasks – generation intermediate representation of the program. Some are described optimization over this representation, the qualities of a generated code aimed at improving. The architecture of the developed *DCUIL2PAS* decompiler is also given.

Keywords: reverse engineering, CIL, dcuil.

Введение

Для разработки большинства сложных программных систем в среде *Delphi* часто используются сторонние компоненты, предоставляемые в виде скомпилированных модулей. Такой подход существенно сокращает время и стоимость разработки программного обеспечения. С другой стороны, наличие сторонних модулей уменьшает надежность программного обеспечения с точки зрения информационной безопасности из-за возможного наличия уязвимостей. Кроме того сторонние компоненты могут содержать ошибки, исправление которых может оказаться затруднительным из-за невозможности связаться с разработчиком, отсутствием исходных кодов у разработчика (в связи с их утратой), и т. д. Также, в некоторых случаях может потребоваться доработка сторонних модулей, без возможности использования исходных кодов.

Сторонние модули чаще всего распространяются в закрытом формате *dcu* (*Delphi compiled unit*). В ИДСТУ СО РАН Хмельновым А.Е. разработан инструмент *dcu32int* [1] для разбора модулей *Delphi*. Данный инструмент позволяет получить исходный код на языке ассемблера и интерфейсную часть модуля. Результат разбора в таком виде в отличие от программы на языке высокого уровня не предоставляет необходимого уровня абстракции для того, чтобы за приемлемое время и трудозатраты идентифицировать алгоритмические конструкции, а также отследить способы взаимодействия элементов программы.

1. Виды промежуточных представлений

Наиболее часто используемыми формами промежуточными представлениями являются ориентированный граф, трехадресный код, префиксная и постфиксная запись [2].

Одной из наиболее распространённых форм промежуточного представления является форма статического одиночного присваивания (Static Single Assignment, SSA). Применению данной формы в задаче декомпиляции посвящена докторская диссертация [3] Майка Ван Еммерика, результатом которой является декомпилятор *Boomerang* [4] использующий форму SSA. Также данная форма представления используется в таких декомпиляторах, как *dcc* [5], *REC* [6], *SmartDec* [7], *Hex-Rays* [8] и других.

Основной особенностью программы представленной в форме SSA является то, что любая переменная может быть определена только один раз. Для комбинации двух определений в местах схождения потока данных применяется соглашение о записи, называемое φ-функцией. Ниже приведен пример перевода программы в SSA форму (слева исходная программа, справа, в форме SSA).


```
if a > b then
  Result := a
else
  Result := b;
```

```
if a > b then
  Result1 := a
else
  Result2 := b;
Result3 := φ(Result1, Result2);
```

Другой формой промежуточного представления является синтаксическое дерево программы. Ту же самую информацию, но в более компактном виде дает представление в виде ориентированного ациклического графа. Здесь общие подвыражения объединены в одну вершину.

На рис 1. приведены представления для выражения $a := b + c * b$ в виде синтаксического дерева (рис 1. а) и ориентированного ациклического графа (рис 1. б).

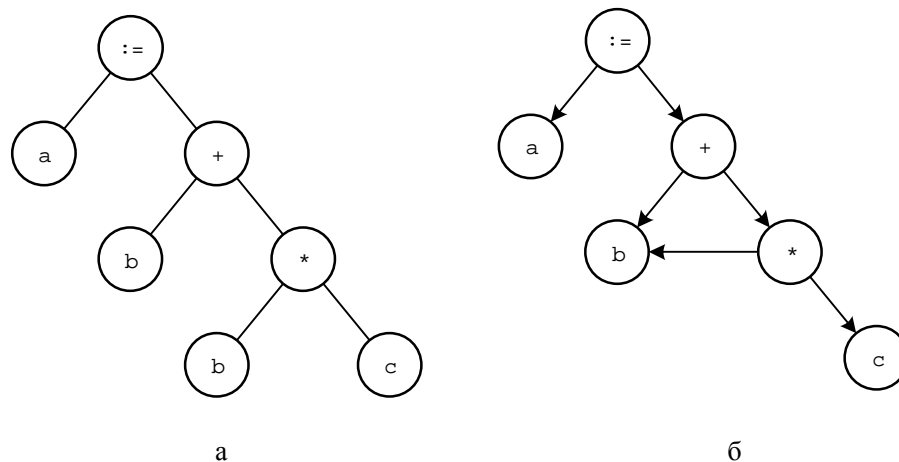


Рис. 1. Виды промежуточного представления.

2. Формат объектных файлов *dcuil*

В объектных файлах Delphi в отличие от исполняемого файла в формате *PE* программа оказывается более структурированной, например, выделены блоки памяти соответствующие коду каждой процедуры; имеется информация о типах данных; может присутствовать отладочная информация. Такая информация отсутствует в обычных исполняемых файлах. В общем виде формат файла скомпилированных юнитов Delphi выглядит следующим образом: сначала идет небольшой заголовок, в котором содержится общая информация о файле, такая как размер, время компиляции и т. д. После заголовка следует поток теговой информации. Для обобщения теги можно разделить на следующие группы:

- Описания включаемых модулей и объектных файлов.
- Импортируемых из этих модулей определений (типов данных, процедур, и т. д.).
- Описания определений (типов данных, процедур и функций, и т. д.) из данного модуля.
- Блок памяти, составленный из блоков для процедур и функций, образов констант, и т. д.
- Информация для редактора связей (в какие места блока памяти необходимо занести адреса, получаемые из других модулей).
- Отладочная информация.

Блоки кода состоят из наборов *CIL* инструкций, предназначенных для выполнения виртуальной машиной *.NET*. Байт код *CIL* имеет следующие преимущества по сравнению с машинным кодом:

- код отделен от данных;
- машина является стековой, причем стек жестко типизирован;

- стек используется, как правило, только для хранения промежуточных результатов;
- большинство команд CIL получают свои аргументы на стеке, удаляют их со стека и помещают вместо них результат(ы) вычисления;
- машина является объектно-ориентированной: структура CIL отражает разбиение кода на классы, методы и т. д.

3. Промежуточное представление

Промежуточное представление *CILIR (CIL Intermediate Representation)* разработанное в декомпиляторе *DCUIL2PAS* реализовано в виде иерархии классов и строится для каждого базового блока графа потоков управления. Сначала определяется начальное состояние каждого блока, которое задаёт значения параметров и аргументов функции, а также состояние стека. Затем каждой инструкции *CIL* путём последовательного обхода линейного участка кода ставится в соответствие выражение (экземпляр класса), реализующий семантику «опкода».

Поскольку переменная на линейном участке кода может иметь несколько вхождений в выражения, то при генерации новых выражений для экономии памяти каждая переменная снабжается счетчиком ссылок. Вначале счетчики всех переменных устанавливаются равными 1. При каждом вхождении переменной в выражение счетчик увеличивается на единицу. При переопределении переменной ей присваивается ссылка на новый объект, счетчик ссылок при этом снова устанавливается равным 1. В дальнейшем для вычисления результатов выражений используются ссылки на существующие объекты.

Иерархия классов, реализующая промежуточное представление изображена на рис 2.

Базовый класс *TCILExpr* реализует логику работы со счетчиками ссылок и задает набор обязательных методов:

- *Eval* – вычисляет значение выражения;
- *Eq (E: TCILExpr)* – возвращает true если переданное выражение эквивалентно текущему ;
- *AsString (BrRq: boolean)* – возвращает текстовое представление выражения;
- *Show* – выводит на печать текстовое представление выражения ;

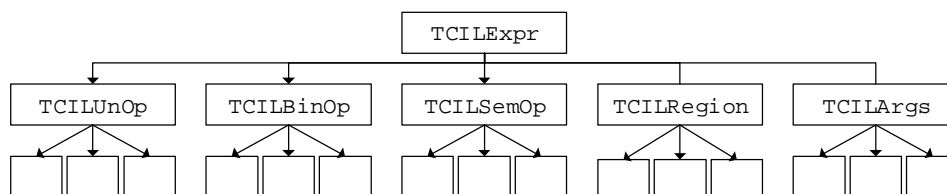


Рис. 2. Иерархия классов промежуточного представления

Наследники класса *TCILUnOp* описывают все «опкоды» аргументом которых является один единственный операнд. Аналогично все наследники *TCILBinOp* описывают семантику всех бинарных операций. Аргументы и локальные переменные наследуются от класса *TCILArgs*. Для описания оставшихся неописанных инструкций в качестве базового используется класс *TCILSemOp*.

Все выделяемые регионы в процессе анализа потоков управления, также являются наследниками базового класса *TCILExpr*. Каждому из выделяемых регионов соответствует свой класс, реализующий его семантику. Список классов реализующих регионы в программе *DCUIL2PAS*: *TCILIfThenElse*, *TCILWhile*, *TCILRepeat*, *TCILCaseSt*, *TCILBasicBlock*.

Все условные и безусловные переходы в процессе разбора преобразуются в выражение вида – *CILCond (Next, Trg; Cond)*, где *Next* и *Trg* – это блоки назначения перехода, *Cond* – условие перехода.

Для логических выражений, включающих логическое сложение, умножение компилятор может генерировать код для операторов условного перехода двумя разными способами:

1. Условие будет преобразовано в последовательность инструкций, результат будет помещен на стек, для последующего извлечения в качестве аргумента для «опкода» условного перехода. То есть вычислять их в процессе выполнения программы.

2. С помощью условных выражений, основанных на следующих правилах:
A and B => **if A then B else False**,
A or B => **if A then True else B**.

В первом случае результатом вычисления выражения извлеченного со стека в качестве операнда условного перехода будет исходное логическое условие. Во втором случае объединение сложных выражений происходит в процессе их вычисления по заранее определённым набору правил. Например, выражение `IfThenElse(Cond, IfThenElse(Cond1, !True1, !False), !False)` будет приведено к виду `IfThenElse(Cond and Cond1, True1, False)`. В случае если условное `IfThenElse(Cond, IfThenElse(True, (Cond1, True, False)), False)` выражение содержится в ветке `False`, тогда выражение примет следующий вид: `IfThenElse(Cond or Cond1, True, False)`. Аналогичным образом объединение происходит для циклов. Выражение вида `WhileSt(Cond, WhileSt(Cond1, Trg))` будет приведено к виду `WhileSt(Cond and Cond1, Trg)`.

Например, промежуточное представление для кода:

```
function IfSt1 (a: Integer; b: Integer; c: Integer): Integer;  
var  
  Result: Integer;  
begin  
  if (a > b) then  
    if (a > c) then  
      if (b > c) then  
        Result := 1  
      else  
        Result := 0  
    else  
      Result := 0  
  else  
    Result := 0;  
end;
```

будет выглядеть следующим образом:

```
IfThenElse(Cond, IfThenElse(Cond1, IfThenElse(Cond2, True, False),  
False), False)
```

После применения оптимизации объединения условий код примет следующий вид:

```
function IfSt1 (a: Integer; b: Integer; c: Integer): Integer;  
var  
  Result: Integer;  
begin  
  if (a > b) and (a > c) and (b > c) then  
    Result := 1;  
  else  
    Result := 0;  
end;
```

4. Архитектура декомпилятора

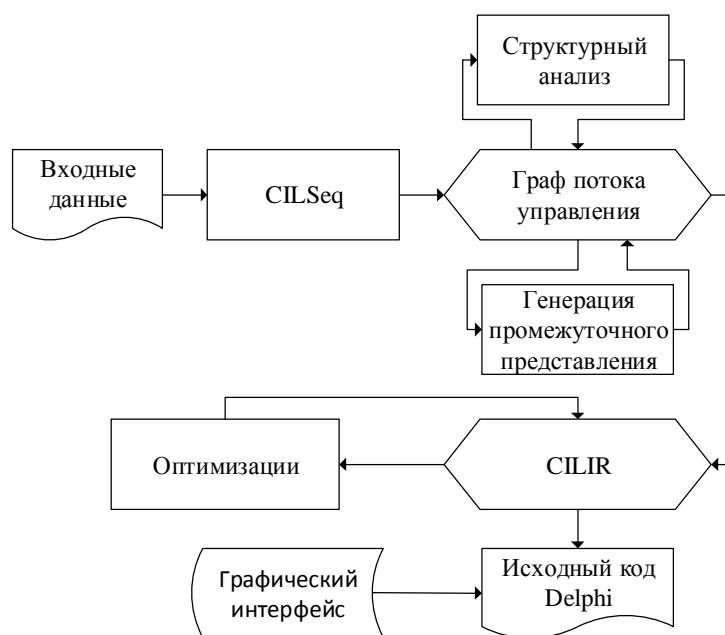


Рис. 3. Архитектура декомпилятора DCUIL2PAS

На рис. 3 многоугольниками представлены компоненты декомпилятора, а стрелками отображается поток данных между ними. В архитектуре *DCUIL2PAS* можно выделить следующие составные блоки (рис 3):

- *CILSeq* – это внутренне представление входной программы в виде последовательности CIL инструкций;
- Модуль *CILCtrlflowGraph* выполняет построение графа потоков управления;
- Модуль *структурного анализа* восстанавливает высокоуровневые конструкции языка посредством модификации графа потоков управления;
- Модуль *промежуточного представления* отвечает за генерацию промежуточного кода;
- Модуль *оптимизаций* производит некоторые изменения промежуточного представления для улучшения качества генерируемого кода. Одной из важных оптимизаций является объединение условий операторов ветвления;
- Модуль *GUI* реализует графический интерфейс пользователя с поддержкой подсветки синтаксиса.

На вход декомпилятору подаются файлы в формате *dcuil*. С помощью функционала реализованного в программе *dscu32int[1]* выделяется блоки кода для процедур и функций. С помощью реализованного на основе библиотеки *Mono[9]* дизассемблера генерируется байт-код CIL соответствующий подпрограммам. Далее строится граф потока управления путем выделения базовых блоков и связей между ними. Для каждого базового блока строится промежуточное представление. Каждой инструкции ставится в соответствие выражение, полностью описывающее семантику кода. При этом производятся все необходимые операции со стеком, локальными переменными и параметрами подпрограммы. Далее с помощью методов структурного анализа и дерева доминирующих вершин производится структурный анализ. Полученное промежуточное представление подвергается некоторым оптимизациям, нацеленным на улучшения качества производимого кода. На выходе декомпилятор производит код на языке *Delphi* семантически эквивалентный исходной программе.

5. Пример декомпиляции файла в формате *dcuil*

Для демонстрации результатов работы декомпилятора *DCUIL2PAS* рассмотрим следующий небольшой пример. В листинге 1 представлен результат разбора функции, полученного с помощью программы *dcu32int*:

```
function CmpIf (n: Integer; b: Integer): Integer;
var
  Result: Integer;
  i: Integer;
begin [Flags:3013,MaxStack:3,CodeSz:37,LocalVarSigTok:0]
00: . |16 | ldc_i4_0
01: . |0A | stloc_0
02: . |03 | ldarg_1
03: . |1B | ldc_i4_5
04: ю. |FE 02| cgt
06: . |02 | ldarg_0
07: . |1D | ldc_i4_7
08: ю. |FE 02| cgt
0A: a |61 | xor
0B: ,. |2C 06 | brfalse_s $13
0D: . |06 | ldloc_0
0E: . |1B | ldc_i4_5
0F: X |58 | add
10: . |0A | stloc_0
11: +. |2B 04 | br_s $17
13: . |06 | ldloc_0
14: . |18 | ldc_i4_2
15: X |58 | add
16: . |0A | stloc_0
17: . |03 | ldarg_1
18: .. |1F 0F | ldc_i4_s 15
1A: /. |2F 06 | bge_s $22
1C: . |06 | ldloc_0
1D: . |1B | ldc_i4_5
1E: X |58 | add
1F: . |0A | stloc_0
20: +. |2B 04 | br_s $26
22: . |06 | ldloc_0
23: . |18 | ldc_i4_2
24: X |58 | add
25: . |0A | stloc_0
26: . |03 | ldarg_1
27: .< |1F 3C | ldc_i4_s 60
29: /. |2F 06 | bge_s $31
2B: . |06 | ldloc_0
2C: . |1B | ldc_i4_5
2D: X |58 | add
2E: . |0A | stloc_0
2F: +. |2B 04 | br_s $35
31: . |06 | ldloc_0
32: . |18 | ldc_i4_2
33: X |58 | add
34: . |0A | stloc_0
35: . |06 | ldloc_0
36: * |2A | ret
end;
```

Листинг 1. Функция на СIL.

Результат разбора в таком виде не предоставляет необходимого уровня абстракции для его анализа специалистом за приемлемое время и трудозатраты. Для того чтобы провести анализ такого кода, необходимо знать состояние стека, значения переменных и аргументов функции в каждый момент выполнения программы. Также необходимо знать семантику всех представленных «опкодов».

В листинге 2 приведен результат декомпиляции функции из листинга 1. Результат разбора в таком виде семантически эквивалентен исходному коду, не содержит операторов неструктурных переходов и представляет собой код на языке высокого уровня Delphi.

```
function CmpIf (n: Integer; b: Integer): Integer;  
var  
    Result: Integer;  
    i: Integer;  
begin  
    Result := 0;  
    if (((b > 5) mod (n > 7)) <> 0) then  
        Result := Result + 5  
    else  
        Result := Result + 2;  
    if (b < 15) then  
        Result := Result + 5  
    else  
        Result := Result + 2;  
    if (b < 60) then  
        Result := Result + 5  
    else  
        Result := Result + 2;  
end;
```

Листинг 2. Результат декомпиляции функции из листинга 1.

Заключение

В работе рассмотрена задача декомпиляции на примере объектных файлов *dcuil*. Предложено и реализовано промежуточное представление, позволяющее генерировать семантически эквивалентный код на языке высокого уровня Delphi.

Высокий уровень абстракции байт-кода СІЛ с информацией о типах данных и именах переменных содержащейся в объектных файлах *dcuil* позволяет добиться высокого качества генерируемого кода.

Литература

1. <http://hmelnov.icc.ru/DCU/index.ru.html> // DCU32INT - Программа для разбора юнитов Delphi.
2. Ахо А., Лам М., Сети Р., Ульман Д.Д. Компиляторы: принципы, технологии и инструменты. 2-е изд. – Вильямс, 2008. – 1184 с.
3. Michael James Van Emmerik. Static Single Assignment for Decompileation.
4. <http://boomerang.sourceforge.net> // A general, open source, retargetable decompiler of machine code programs.
5. <http://www.itee.uq.edu.au/cristina/d-cc.html#thesis> // Cifuentes C. Reverse compilation techniques. — 1994.
6. <http://www.backerstreet.com/rec/rec.htm> // REC Studio 4 - Reverse Engineering Compiler
7. <http://smartdec.ru> // SmartDec PUSHING NATIVE CODE DECOMPILATION TO THE NEXT LEVEL
8. <https://www.hex-rays.com/index.shtml> // Hex-Rays
9. http://www.mono-project.com/Main_Page // cross platform, open source .NET development framework.

Михайлов Андрей Анатольевич, младший научный сотрудник Института динамики систем и теории управления Сибирского отделения Российской академии наук, e-mail: mikhailov@icc.ru

Mikhailov Andrey Anatolievich, junior researcher, Institute of system dynamics and control theory SB RAS.

УДК 004.42

© Г.В. Попков

О ПРОБЛЕМЕ ЖИВУЧЕСТИ ТЕЛЕКОММУНИКАЦИОННЫХ СЕТЕЙ

В статье рассматриваются общие подходы к построению модели информационной сети испытывающей разрушающие информационные воздействия. Даны общие определения живучести, надёжности элементов сети, так и сети связи в целом. В статье даются комплексные показатели живучести информационной сети, а так же рассмотрены факторы внешнего и внутреннего воздействия на сети связи. Дана классификация типовых удалённых атак на информационные сети.

Ключевые слова: разрушающие информационные воздействия, информационная сеть, живучесть, надёжность сети связи, сети связи.

© G.V. Popkov

ABOUT SURVIVABILITY PROBLEM OF TELECOMMUNICATION NETWORKS

This article discusses general approaches to model building information network experiencing destructive informational influence. Given the general definition of survivability, reliability of the network elements and networks in general. The article gives a comprehensive vitality indexes information network, as well as factors external and internal impacts on the communication network. Given the classification of the type of remote attacks on information networks.

Keywords: destructive informational influence, information network, survivability, reliability of the communication network, the communication network.

Введение

Объектом исследования являются информационные сети, построенные на основе современных информационно-коммуникационных технологий, и происходящие в них процессы преобразования информации связанные с её сохранением и передачей от источника до получателя. В последнее время участились так называемые кибер – атаки, на различные ведомства, отдельные сайты, сервера режимных организаций, в связи с этим все актуальней становятся вопросы, связанные с защитой информации на всех уровнях информационных сетей. Важным становится разработка отечественного программного обеспечения и аппаратных средств для предотвращения такого рода атак и разрушающих информационных воздействий на сегменты информационных сетей.

1.1. Концептуальная модель информационной сети

Информационная сеть есть совокупность информационных средств, предназначенных для реализации следующих операций над информацией: получения, сбора, передачи, обработки, хранения, поиска, предоставления, использования и др.[1]

Макроструктура информационной сети представлена на рис. 1. и включает три основные компоненты: терминальный комплекс, сеть электросвязи, информационную и вычислительную базы.



Рис. 1. Макроструктура информационной сети.

Все компоненты сети представляют собой совокупность технических и программных средств.

Терминальный комплекс содержит все виды пользовательских устройств, начиная от обычного телефонного аппарата и включая все виды современных абонентских устройств: черно-белый и цветной

видеотелефоны, факсимильный аппарат, персональную ЭВМ, локальную вычислительную сеть, сигнально-контрольное и специальное оборудование.

Сеть электросвязи по известному определению есть совокупность каналов и трактов, образующих первичную сеть и построенные на её основе вторичные сети, предназначенные для удовлетворения потребностей пользователей по доставке любой информации, преобразованной в сигналы электросвязи [2].

Первичная сеть предназначена для образования каналов и трактов между узлами (пунктами) связи, используемыми вторичной сетью. В общем случае она состоит из линий связи различной физической природы, систем передачи и системы управления [3].

В качестве линейных средств в первичной сети применяются кабельные (в т.ч. ВОЛС), радиорелейные, тропосферные и спутниковые системы передачи, а также средства прямой радиосвязи различных диапазонов волн.

Вторичная сеть предназначена для непосредственного обеспечения передачи или обмена сообщениями заданного вида между абонентами, и предоставления различных услуг связи пользователям сети.

Вторичная сеть состоит из каналов первичной сети, используемых данной сетью, аппаратуры преобразования, обработки и ввода в канал связи сообщений данного вида информации, устройств коммутации сообщений или каналов, а также системы управления [3].

Важным элементом структуры вторичных сетей являются оконечные узлы связи (рис.2).

Оконечный узел предоставляет абоненту все виды услуг обеспечиваемых данной вторичной сетью.



Рис. 2. Оконечный узел.

Основными структурными элементами оконечного узла могут быть пункт или станция каналообразования (системы передачи, кросс каналов), пункт управления, оборудование коммутации и маршрутизации, радиоцентры, оборудование абонентского доступа и абонентская сеть, системы электропитания и др.

Информационная и вычислительная базы

представляют собой распределённые по территории системы банков данных и систему компьютеров большой производительности, обеспечивающих вычислительную поддержку терминальных комплексов.

По определению сеть электросвязи предназначена для доставки любой информации между терминальными комплексами и банками данных в заданном сочетании [3]. Под термином

"доставка" понимается передача информации по каналам связи, коммутация, хранение информации на узлах коммутации в процессе её передачи и коммутации, а также хранение как самостоятельная функция с различной длительностью и делением на оперативное, длительное и архивное время.

Для облегчения разработки методов оценки живучести множество сетей связи условно разбиваются на два класса – двухполюсные и многополюсные. При наличии в системе связи только двух оконечных узлов – система двухполюсная, если количество узлов более двух – многополюсная [3]. Понятие двухполюсной сети целесообразно использовать в случае необходимости оценки живучести связи на том или ином информационном направлении, т.е. между определённой парой полюсов многополюсной системы связи.

Таким образом, система доставки информации включает в себя множество связанных между собой элементов, отказ или повреждение каждого из которых оказывает влияние различной степени на работоспособность системы в целом.

1.2. Понятия надёжности и живучести, их различия

Оба понятия связаны с работоспособностью системы связи во времени, т.е. с выполнением заданных функций в установленном объёме на требуемом уровне качества в течение определённого периода эксплуатации или в произвольный момент. Различия этих понятий обусловлены прежде всего различиями причин или факторов, нарушающих нормальное функционирование системы, и характером нарушений.

Надёжность системы связи [3] есть её свойство обеспечивать связь, сохраняя во времени значения установленных показателей качества в заданных условиях эксплуатации. Она отражает влияние на работоспособность системы главным образом внутрисистемного фактора – случайных отказов техники, вызываемых физико-химическими процессами старения аппаратуры, дефектами технологии её изготовления или ошибками обслуживающего персонала.

В то время, как *живучесть* [4] характеризует устойчивость системы связи против действия причин, лежащих вне системы и приводящих к разрушениям, значительным повреждениям или временной потере работоспособности всей сети или некоторой части её элементов – узлов, пунктов, станций и линий связи. Все причины можно разделить на два класса: случайные или стихийные и преднамеренные. К стихийным (случайным) факторам относятся такие, как гроза, землетрясение, оползни, разливы рек, ошибки персонала, пользователей и т.п., а к преднамеренным – огневые, в том числе и ракетно-ядерные удары противника в условиях войны, использование высокоточных видов вооружений, диверсионных и информационных разрушающих воздействий.

Различия в причинах нарушения связи обуславливают существенные отличия в их проявлении, характере и масштабности нарушений связи, их продолжительности, путях и способах их устранения и повышению устойчивости системы. Если поток случайных отказов техники приводит к нарушению лишь отдельных связей и обладает свойством ординарности (малая вероятность одновременного отказа нескольких связей), то нарушения работы системы указанными выше факторами живучести обладают существенно иными свойствами. Особенно это характерно для преднамеренного поражения системы, когда одновременно может выводиться из строя значительная часть или даже вся система связи. Интервалы между воздействиями факторов живучести на систему связи не подчиняются экспоненциальному закону, что характерно для случайных отказов аппаратуры, а преднамеренные воздействия противника, кроме того, не являются случайными. Нарушения работы отдельных связей из-за случайных отказов или ошибок обслуживающего персонала обычно кратковременны, так как они в большинстве случаев быстро устраняются, что же касается нарушений связи из-за разрушения отдельных её элементов – узлов, линий то они несравненно более продолжительны. Если первые исчисляются обычно минутами или десятками минут, то вторые часами и сутками, поскольку это связано с большим объёмом восстановительных работ.

В силу небольших перерывов связи из-за случайных отказов многие связи могут обходиться без резервирования. С учётом же аспекта живучести без резерва не обойтись, так как обслуживаемые процессы управления, как правило, не допускают столь длительного отсутствия связи.

Случайные отказы характерны для отдельных устройств, линий или каналов связи. При этом отказ одного аппарата на узле связи обычно не вызывает отказов других комплектов аппаратуры, а тем более целого элемента или всего узла связи. Исключения составляют общие коммутаторы и

агрегаты электропитания. Поэтому при расчёте надёжности системы или сети связи отказы её структурных элементов, не имеющих общих устройств, считаются взаимонезависимыми.

Поражающие факторы могут выводить из строя одновременно несколько элементов узла, и даже системы. Так если воздействию подвергается узел связи, то могут оставаться непораженными лишь его вынесенные пункты и станции. Поскольку аппаратура систем передачи обычно размещается в составе ядра узла связи, при поражении узла с большой вероятностью выходят из строя подходящие к нему кабельные линии связи. Соответственно могут быть нарушены все связи, осуществляемые данным узлом. В период массированного ядерного удара противника взаимная зависимость событий-поражений распространяется на все элементы системы. Естественно, что эти особенности нельзя не учитывать при расчёте живучести систем и сетей связи.

Следует иметь в виду и неодинаковую степень погрешности исходных данных для оценки надёжности и живучести систем. По эксплуатационно-техническим отказам техники и линий связи имеется обширный статистический материал. И хотя достоверность исходных данных по надёжности техники связи представляет пока что известную проблему, достигнутая их точность несравненно выше точности исходных данных для анализа живучести системы связи. К примеру, не существует пока еще достаточных научных основ прогнозирования землетрясений. Естественно не может быть сколько-нибудь необходимая для расчётов живучести вероятность того, что в течение определённого периода в интересующем районе землетрясения не будет, нельзя достоверно описать его параметры. Весьма проблематично также и прогнозирование поражающего воздействия противника.

Из этого факта следует, что в настоящее время и в обозримом будущем оценка живучести сети (системы) связи может быть лишь грубо приближённой, ориентировочной. Поэтому и методы её расчёта могут быть приближенными, лишь бы вносимая ими погрешность не превышала неточности исходных данных.

Методика расчёта показателей надёжности, к которым обычно предъявляются высокие требования по точности, в некоторых случаях могут использоваться для расчёта для оценки живучести системы связи, однако для сокращения трудоёмкости расчётов, исходя из изложенного выше, они могут быть существенно упрощены. Если же для анализа и синтеза живучести системы связи разрабатываются специальные методы, то излишнее их усложнение во имя повышения точности было бы неоправданно.

Неодинаковы также и критерии отказов связи между полюсами системы. Если при оценке надёжности связь между парой полюсов может считаться отказавшей при наличии $k < n$ заданных связей ($n=1,2, \dots$), то при оценке живучести критерием отказа связи между двумя полюсами обычно является полное отсутствие связи любого вида.

Таким образом, надёжность и живучесть – различные понятия и самостоятельные проблемы, требующие своих решений при разработке и совершенствовании систем и сетей связи.

1.3. Определение живучести

Под живучестью мы будем понимать устойчивость системы связи к повреждению элементов стихийными факторами и преднамеренными воздействиями противника.

Устойчивость - свойство системы связи, заключающееся в её способности осуществлять своевременную передачу информации в необходимом объёме и с качеством не хуже заданного при определённых условиях функционирования [4].

Живучесть сети связи включает в себя два основных свойства, каждое из которых разбивается на два частичных (рис. 3).

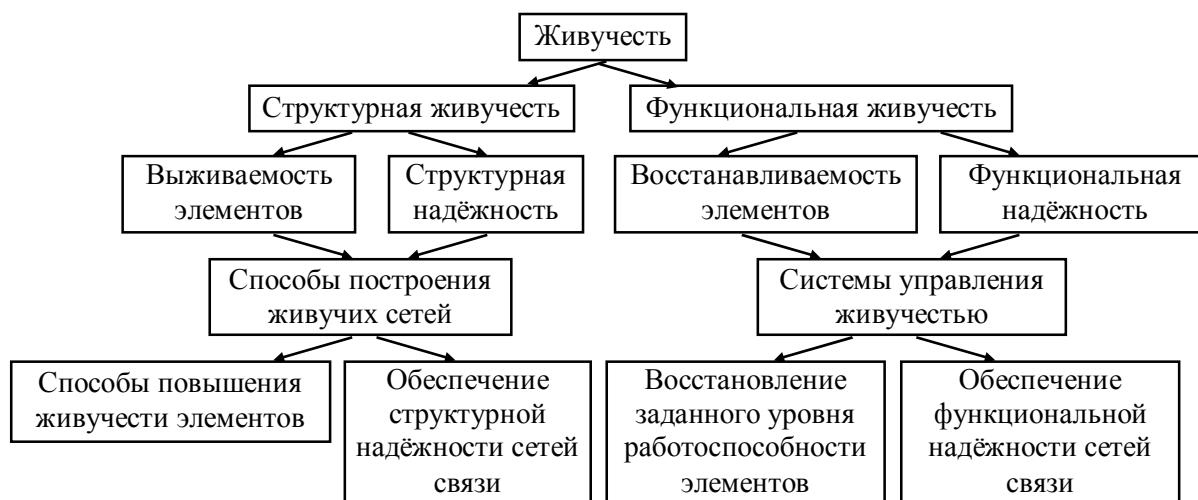


Рис. 3. Основные свойства и системы обеспечения живучести систем связи

Структурная живучесть - живучесть (выживаемость) системы связи в течение некоторого времени при пассивном противодействии повреждениям (случайным или целенаправленным) элементов сети.

Пассивное противодействие осуществляется при нормальном режиме работы системы связи за счёт защищённости элементов, включенного резерва, структурной организации и принятого алгоритма управления маршрутизацией.

Таким образом, данное понятие включает как живучесть элементов, так и сети связи в целом.

Выживаемость элементов - структурная живучесть элементов системы связи.

Структурная надёжность - структурная живучесть системы связи при заданном алгоритме (процессе) разрушения элементов.

Функциональная живучесть - живучесть (выживаемость) системы связи в течение некоторого времени при активном противодействии повреждениям (случайным или целенаправленным) элементов сети.

Активное противодействие в основном осуществляется во время действия разрушающих факторов и определяется системами восстановления элементов и управления на сетях связи.

Таким образом, при исследовании функциональной живучести сеть моделируется динамическими объектами.

Восстанавливаемость элементов - функциональная живучесть элементов системы связи.

Функциональная надёжность - функциональная живучесть системы связи при заданном алгоритме (процессе) разрушения элементов сети.

1.4. Критерии живучести

Работоспособность сети (элемента):

- сеть (элемент) работоспособна, если она способна выполнять все или заданную часть функций в полном или частичном объёме.

Критерий выживаемости элемента - время, при котором элемент остаётся работоспособным с начала его разрушения заданными способами и средствами или количественная мера определённых средств разрушения, не нарушающих работоспособность элемента.

Критерий восстанавливаемости элементов - время, за которое может быть восстановлена работоспособность элемента при заданном уровне резервирования и системы техобслуживания или количественная мера определённых ресурсов, восстанавливающих работоспособность элементов при заданной степени разрушения.

К ресурсам восстановления можно отнести: резервы элементов сети в целом, отдельных блоков аппаратуры связи для ремонта элементов, количество людей, предназначенных для восстановления, наличие предприятий изготовляющих новые элементы и т.д.

В качестве показателей живучести элементов можно принять:

- вероятность отказа или количество ресурса разрушения;

- среднее время восстановления или количество ресурса восстановления.

Критерий структурной надёжности - число отказавших элементов, не нарушающих работоспособность сети связи, т.е. заданная часть абонентов сети из выделенных пунктов связи (или всех) будет соединена связующей системой (коммуникацией) при отказе любых k элементов сети связи.

Под коммуникацией будем понимать некоторую двухполюсную сеть с необходимыми ограничениями (расстояние между полюсами, связность, пропускная способность). Поэтому необходимо определить множество работоспособных коммуникаций и типы отказов элементов.

Например, пункты x и y сети связи соединены коммуникацией тогда и только тогда, когда между x и y существует не менее T независимых цепей, длина которых превосходит d , а суммарная пропускная способность не меньше заданной величины.

Критерий функциональной надёжности - время активного сохранения работоспособности сети связи при заданной интенсивности отказов её элементов.

Под активным сохранением понимается возможность противостоять разрушениям с помощью системы управления живучестью сети связи. В частности, за счёт восстановления структуры и функций первичной и вторичной сетей, систем управления потоками, нагрузкой и маршрутизацией. При этом для определения функциональной надёжности необходимо полностью определить процесс отказов элементов сети.

В этом случае, когда процесс отказов детерминирован, критерий функциональной надёжности можно видоизменить следующим образом - число восстановленных коммуникаций между выделенными абонентами сети в определённый отрезок времени и при фиксированном процессе отказов. Легко понять, что эти два критерия эквивалентны, так как для любой сети из одного критерия следует второй.

Показатели живучести - количественная характеристика одного или нескольких свойств, составляющих живучесть систем связи и её элементов.

Из этого определения непосредственно следуют несколько признаков, по которым можно классифицировать показатели живучести - различают показатели живучести элементов или системы связи в целом:

- единичный (комплексный показатель живучести относится к одному (нескольким) из свойств, составляющих живучесть системы связи;
- однозначный (многозначный) показатель живучести выражается одним числом (вектором или функцией);
- непосредственный показатель живучести соответствует выбранному критерию живучести, а опосредованный показатель функционально связан с непосредственным показателем и оценивает последний сверху или снизу;
- детерминированный (стохастический) показатель живучести характеризует свойства живучести в детерминированных (стохастических) сетях связи;
- детерминированные показатели живучести можно подразбить на прямые и обратные. Так как в любом критерии живучести рассматриваются два фактора, то, зафиксировав один, по другому проводят оптимизацию;
- стохастические показатели живучести также разбиваются на два подтипа: вероятностные - характеризуют вероятность появления некоторого события; числовые - математическое ожидание некоторой случайной величины (время, число элементов, коммуникаций и т.д.).

1.5. Показатели функциональной надёжности системы связи

Для детерминированных сетей связи, прежде всего, интересны показатели, характеризующие возможности систем управления по поддержанию работоспособности сети в заданных пределах на определённый отрезок времени. Здесь возможны два подхода к формированию показателей:

Первый - заключается в оценке возможности системы управления восстановить максимальное число необходимых связей с заданной пропускной способностью за счёт фиксированных средств обеспечения функциональной надёжности сети. Таким образом, при этом подходе фиксируется разрушение и ресурс восстановления (необходимость восстановить определённые связи в течение некоторого времени).

Например, число восстановленных связей за счёт управления структурой вторичной сети при заданных резервах и характере разрушения элементов сети, оценка возможности системы управления оптимально перестраивать структуру вторичной сети с целью повышения работоспособности.

Второй подход заключается в оценке возможности системы управления восстановить работоспособность на необходимом уровне за счёт минимально возможного резерва.

Из-за сложности определения оптимального разрушения сети для заданного обеспечения живучести более широкое распространение получили стохастические показатели:

- вероятность того, что сеть будет в работоспособном состоянии в случае воздействия на нее поражающих факторов, а также прогнозируемая длительность восстановления связи;
- среднее значение вероятности выживания связи между одной парой полюсов;
- математическое ожидание времени восстановления работоспособности;
- математическое ожидание времени перестройки вторичной сети;
- математическое ожидание времени передачи сообщения при воздействии разрушающих факторов.

Необходимо также рассматривать показатели связанные непосредственно с сообщениями передаваемыми по сети.

В качестве таких показателей можно рассмотреть следующие:

1. $\gamma_n = (1 - M/N)$ – коэффициент потерь сообщений, где N – число всех сообщений, а M – число полученных сообщений.
2. Вероятность прихода исходного сообщения в заданный интервал времени.
3. Вероятность того, что сообщение не будет изменено.
4. Математическое ожидание времени нахождения сообщения (пакета) в сети.
5. Вероятность передачи и приема сообщения не по заданному адресу.
6. Вероятность перехвата и прочтения сообщения посторонними лицами.

Перечисленные показатели не могут быть вычислены непосредственно по модели сети, так как здесь существенно завязаны средства воздействия на сеть. Разумеется, структура и система управления сетью также существенно влияет на численные значения этих показателей.

Поэтому данные показатели могут быть найдены только с помощью имитационных моделей.

2.1. Классификация информационных атак в информационных сетях и методы их обнаружения

В разделе описаны виды угроз и методы доступа к информационным ресурсам, приведена классификация информационных разрушающих воздействий (удаленных атак) на распределённые вычислительные системы с учётом эталонной модели ISO/OSI и методы их обнаружения, а также рассмотрены характеристики типовых удалённых атак и их классификация [5].

Виды угроз и методы доступа к информационным ресурсам:

Основные понятия, которыми оперирует теория компьютерной безопасности это **угрозы, уязвимости и атаки.**

Угроза безопасности компьютерной системы - это потенциально возможное событие (воздействие), которое в случае своей реализации может стать причиной нарушения целостности информации, её потери или замены.

Уязвимость компьютерной системы - это некая ее характеристика, которая делает возможным возникновение угрозы.

Атака на компьютерную систему - это действие, которое заключается в поиске и использовании той или иной уязвимости. Таким образом, атака - это реализация угрозы.

Существует три основных вида угроз безопасности - это угрозы раскрытия, целостности и отказа в обслуживании.

Угроза раскрытия имеет место, когда получен доступ к некоторой конфиденциальной информации, хранящейся в вычислительной системе или передаваемой от одной системы к другой.

Угроза целостности включает в себя любое изменение (модификацию или удаление) данных, хранящихся в вычислительной системе или передаваемых из одной системы в другую.

Угроза отказа в обслуживании возникает, когда в результате некоторых действий блокируется доступ к некоторому ресурсу вычислительной системы. Блокирование может быть постоянным, так чтобы запрашиваемый ресурс никогда не был получен, или оно может вызвать только задержку запрашиваемого ресурса, достаточно долгую для того, чтобы он стал бесполезным.

Угрозы, могут быть как случайными, так и умышленными (преднамеренно создаваемыми). К случайным угрозам можно отнести:

1. Ошибки обслуживающего персонала и пользователей - потери информации, связанные с неправильным хранением архивных данных, случайное уничтожение или изменение данных.

2. Сбой оборудования и электропитания - сбой кабельной системы, перебои электропитания, сбой дисковых систем, сбой систем архивации данных, сбои работы серверов, рабочих станций, сетевых карт и т.д.

3. Потери информации из-за некорректной работы программного обеспечения - потеря или изменение данных при ошибках в программном обеспечении, потери при заражении системы компьютерными вирусами.

4. Потери, связанные с несанкционированным доступом - случайное ознакомление с конфиденциальной информацией посторонних лиц.

К умышленным угрозам относятся:

1. Несанкционированный доступ к информации и сетевым ресурсам;

2. Раскрытие и модификация данных и программ, их копирование;

3. Раскрытие, модификация или подмена трафика вычислительной сети;

4. Разработка и распространение компьютерных вирусов, ввод в программное обеспечение "логических бомб";

5. Кража магнитных носителей и расчётных документов, разрушение архивной информации или умышленное её уничтожение;

6. Фальсификация сообщений, отказ от факта получения информации или изменение времени её приёма;

7. Перехват и ознакомление с информацией, передаваемой по каналам связи и т.п. [6].

К сетевым системам применим специфический вид атак, обусловленный распределенностью ресурсов и информации в пространстве. Это так называемые сетевые или удаленные атаки. Они характерны тем, что нападающий может находиться за тысячи километров от атакуемого объекта, и нападению может подвергаться не конкретный компьютер, а информация, передающаяся по сетевым соединениям.

Таким образом, под удаленной атакой понимается информационное разрушающее воздействие на распределенную вычислительную сеть (РВС), программно осуществляемое по каналам связи. Это определение охватывает такие особенности сетевых систем как распределенность компьютеров и распределенность информации. Далее будут рассмотрены удаленные атаки на инфраструктуру и протоколы сети. Эти удаленные атаки используют уязвимости в сетевых протоколах и инфраструктуре сети. При этом под инфраструктурой сети понимается сложившаяся система организации отношений между объектами сети и используемые в сети сервисные службы.

2.1.1 Классификация удаленных атак информационные сети.

Удаленные атаки классифицированы в таблице 1 по следующим признакам:

По характеру воздействия

• **Пассивное** (класс 1.1) - это воздействие, которое не оказывает непосредственного влияния на работу системы, но может нарушать ее политику безопасности. Примером пассивного типового удаленного воздействия в РВС служит прослушивание канала связи в сети.

• **Активное** (класс 1.2) – это воздействие, оказывающее непосредственное влияние на работу системы (изменение конфигурации РВС, нарушение работоспособности и т. д.) и нарушающее принятую в ней политику безопасности. Практически все типы удаленных атак являются активными воздействиями.

По цели воздействия

Этот классификационный признак является прямой проекцией трех основных типов угроз - раскрытия, целостности и отказа в обслуживании.

• **Нарушение конфиденциальности информации либо ресурсов системы** (класс 2.1) - возможность перехвата информации для получения к ней доступа, но невозможность ее

модификации. Следовательно, перехват информации ведет к нарушению ее конфиденциальности. Примером перехвата информации может служить прослушивание канала в сети. В этом случае имеется несанкционированный доступ к информации без возможности ее искажения. Очевидно также, что нарушение конфиденциальности информации является пассивным воздействием.

- **Нарушение целостности информации** (класс 2.2) - возможность искажения информации означает либо полный контроль над информационным потоком между объектами системы, либо возможность передачи сообщений от имени другого объекта. Таким образом, очевидно, что искажение информации ведет к нарушению ее целостности. Примером удаленной атаки, целью которой нарушение целостности информации, может служить типовая удаленная атака (УА) "Ложный объект РВС".

- **Нарушение работоспособности (доступности) системы** (класс 2.3) Принципиально другой целью атаки является нарушение работоспособности системы. В этом случае не предполагается получение атакующим несанкционированного доступа к информации. Его основная цель - добиться, чтобы операционная система на атакуемом объекте вышла из строя и для всех остальных объектов системы доступ к ресурсам атакованного объекта был бы невозможен. Примером удаленной атаки, целью которой является нарушение работоспособности системы, может служить типовая УА "Отказ в обслуживании".

По условию начала осуществления воздействия

Удаленное воздействие, также как и любое другое, может начать осуществляться только при определенных условиях. В распределенных ВС существуют три вида условий начала осуществления удаленной атаки:

- **Атака по запросу от атакуемого объекта** (класс 3.1) предполагает ожидание передачи от потенциальной цели атаки запроса определенного типа, который и будет условием начала осуществления воздействия.

- В случае **атаки по наступлению ожидаемого события на атакуемом объекте** (класс 3.2) осуществляется постоянное наблюдение за состоянием операционной системы удаленной цели атаки и при возникновении определенного события в этой системе начинается воздействие.

- При **безусловной атаке** (класс 3.3) атака осуществляется немедленно и безотносительно к состоянию системы и атакуемого объекта.

По наличию обратной связи с атакуемым объектом

- При удаленной атаке **с обратной связью** (класс 4.1) на некоторые запросы, переданные на атакуемый объект, атакующему требуется получить ответ, т.е. между атакующим и целью атаки существует обратная связь, которая позволяет атакующему адекватно реагировать на все изменения, происходящие на атакуемом объекте.

- Удаленным **атакам без обратной связи (однаправленная атака)** (класс 4.2) не требуется реагировать на какие-либо изменения, происходящие на атакуемом объекте. Атаки данного вида обычно осуществляются передачей на атакуемый объект одиночных запросов, ответы на которые атакующему не нужны. Подобную УА можно называть однаправленной удаленной атакой. Примером однаправленных атак является типовая УА "Отказ в обслуживании".

По расположению субъекта атаки относительно атакуемого объекта

С точки зрения удаленной атаки чрезвычайно важно, как по отношению друг к другу располагаются субъект и объект атаки, то есть в одном или в разных сегментах они находятся.

- В случае **внутрисегментной атаки** (класс 5.1), субъект и объект атаки находятся в одном сегменте.

- При **межсегментной атаке** (класс 5.2) субъект и объект атаки находятся в разных сегментах. Под сегментом сети понимается физическое объединение хостов.

Данный классификационный признак позволяет судить о так называемой "степени удаленности" атаки. На практике межсегментную атаку осуществить значительно труднее, чем внутрисегментную, при этом межсегментная удаленная атака представляет гораздо большую опасность, чем внутрисегментная по той причине, что объект атаки и атакующий могут находиться на расстоянии многих тысяч километров друг от друга, что может существенно воспрепятствовать мерам по отражению атаки.

Таблица 1. Классификация типовых удалённых атак на распределённые ВС

	Типовая удаленная атака	Характер воздействия		Цель воздействия			Условие начала осуществления воздействия			Наличие обратной связи с атакуемым объектом		Расположение субъекта атаки относительно атакуемого объекта		Уровень модели OSI						
		1.1	1.2	2.1	2.2	2.3	3.1	3.2	3.3	4.1	4.2	5.1	5.2	6.1	6.2	6.3	6.4	6.5	6.6	6.7
	Класс воздействия																			
1.	Анализ сетевого трафика	+	-	+	-	-	-	-	+	-	+	+	-	-	+	-	-	-	-	-
2.	Подмена доверенного объекта РВС	-	+	+	+	-	-	+	-	+	+	+	+	-	-	+	+	-	-	
3.	Внедрение в РВС ложного объекта путем навязывания ложного маршрута	-	+	+	+	+	-	-	+	+	+	+	+	-	-	+	-	-	-	
4.	Внедрение в РВС ложного объекта путем использования недостатков алгоритмов удаленного поиска	-	+	+	+	-	+	-	+	+	-	+	+	-	+	+	+	-	-	
5.	Отказ в обслуживании	-	+	-	-	+	-	-	+	-	+	+	+	-	+	+	+	+	+	

Заключение

Распределенные ВС являются открытыми системами. Любой сетевой протокол обмена, как и любую сетевую программу, можно с той или иной степенью точности спроецировать на эталонную семиуровневую модель OSI. Такая многоуровневая проекция позволит описать в терминах модели OSI функции, заложенные в сетевой протокол или программу. Удаленная атака также является сетевой программой. В связи с этим представляется логичным рассматривать удаленные атаки на распределенные ВС, проецируя их на эталонную модель ISO/OSI. Актуальность создания противодействующего программного обеспечения и аппаратных средств на всех уровнях не вызывает сомнений.

Литература

1. Захаров Г.П. Сети с искусственным интеллектом: концепция построения и вопросы терминологии. Системы и средства телекоммуникаций. – 1993. – №1. – С.3-8.
2. Давыдов Г.Б., Рогинский В.Н., Толчан А.Я. Сети электросвязи. – М.: Связь, 1977. – 360 с.
3. Шмалько А.В. Цифровые сети связи: основы планирования и построения. – М.: Эко-Трендз, 2001.
4. Попков В.К. Математические модели живучести сетей связи. – Новосибирск: ВЦ СО РАН, 1990. – 235 с.
5. Блукке В.П., Попков В.К. Классификация информационных атак в распределённых вычислительных системах. – Труды ИВМ и МГ СО РАН. Сер. Информатика. – Новосибирск, 2002. – Вып.4. – С.11-24.

Попков Глеб Владимирович, кандидат технических наук, доцент, старший научный сотрудник Института вычислительной математики и математической геофизики СО РАН, моб. тел. 89134789130, e-mail: glebpopkov@rambler.ru

Popkov Gleb Vladimirovich, candidate of technical sciences, associate professor, senior researcher, Institute of Mathematics and mathematical Geophysics SB RAS.

УДК 681.51

© Л.Н. Федорченко

ЭКВИВАЛЕНТНОСТЬ КАК ОТНОШЕНИЕ ПОДОБИЯ В ТРАНСЛЯЦИИ ЯЗЫКОВ

В статье представлен краткий обзор различных типов преобразований на грамматиках, применяемых при автоматической генерации анализаторов языка.

Ключевые слова: эквивалентные преобразования грамматик, структурная эквивалентность, отношение подобия.

© L.N. Fedorchenko

EQUIVALENCE AS THE SIMILARITY RELATION IN LANGUAGE TRANSLATION

The paper provides with a brief overview of the different types of transformations on grammars used in the automatic generation of parsers.

Keywords: equivalent transformations of grammars, structural equivalence, similarity relation.

Введение

Каждое преобразование, выполняемое на трансляционных КС-грамматиках [1]–[3] может задаваться как отношение подобия, определяемое на классах грамматик. Такие отношения имеют непосредственную связь с сохранением семантического значения грамматических конструкций при эквивалентных преобразованиях синтаксиса, так как конечной целью трансляции является получение последовательности действий, предписываемых вычислительной среде.

Из грамматик наиболее простыми для разработки являются *регулярные грамматики* [1], которые порождают регулярные языки, обрабатываемые конечно-автоматными механизмами. Чем левее класс грамматик в иерархии Хомского, тем более эффективный алгоритм анализа, а значит и трансляции, можно построить по спецификации. Поэтому на практике широко применяется подкласс грамматик типа 0 (по иерархии Хомского) – контекстно-свободные грамматики (КСГ), относительная простота которых позволяет применять для анализа КС-языков эффективные полиномиальные алгоритмы. Например, алгоритм Эрли (Early) [4] для КСГ имеет вычислительную сложность $O(n^3)$, где n – длина входной цепочки. Более сильные ограничения на КС-языки ($LR(k), LL(k)$) позволяют строить анализаторы, имеющие линейную сложность разбора. Ценой за высокую эффективность является усложнение грамматик, снижение наглядности спецификации синтаксиса, необходимость применения средств уточнения анализа. Всё это выводит спецификацию языка за рамки единого формализма.

Перевод грамматик из одного класса в другой является задачей эквивалентных преобразований, которые предполагают такое изменение правил грамматики, при котором язык (множество слов), порождаемый исходной грамматикой, совпадает с языком, порождаемым преобразованной грамматикой.

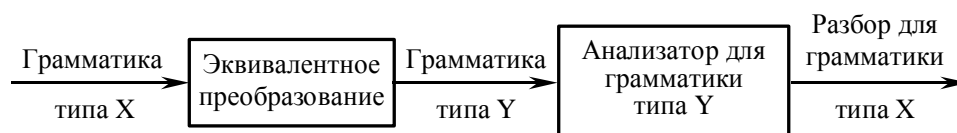


Рис. 1. Схема применения эквивалентного преобразования при разборе

Таким образом, разработчики трансляторов и инструментальных систем, автоматизирующих процесс разработки, при настройке синтаксиса всегда стремятся преобразовать его таким образом, чтобы получить грамматику, принадлежащую к наиболее простому классу иерархии Хомского или к тому подклассу КСГ, для которого есть встроенный в системе автоматического построения трансляторов метод синтеза анализатора языка.

При автоматизированной реализации языка дополнительно к задаче упрощения грамматики приходится решать задачу разрешения конфликтных ситуаций в процессе построения распознавателя (анализатора) языка, связанную с языковой неоднозначностью и недетерминированностью распознающего автомата. Снять эти проблемы можно только с

помощью эквивалентного преобразования грамматики, которое позволяет преобразовать правила грамматики таким образом, что распознающий автомат не содержит конфликтных состояний.

1. Слабая и структурная эквивалентность

По определению две КС-грамматики, порождающие один и тот же язык, слабо эквивалентны. Если две грамматики G_1 и G_2 слабо эквивалентны, тогда для каждого дерева разбора T_1 грамматики G_1 существует дерево разбора T_2 грамматики G_2 , которые имеют одну и ту же крону, т.е. порождают одно и то же слово, и наоборот. В этом смысле слабая эквивалентность является простейшим отношением грамматического подобия КС-грамматик. Но такое отношение ничего не говорит о форме деревьев вывода и о связи между ними. Как правило, отношения грамматического подобия вводятся именно с целью описать такие связи между деревьями выводов двух грамматик, и почти всегда они предполагают слабую эквивалентность. Другим полюсом в иерархии понятий грамматического подобия является структурная эквивалентность, предложенная в работах [5] – Макнотомом и в [6]–[7] – Паулом и Ангером. Для любой КС-грамматики $G = (N, T, P, S)$ определяют скобочную версию $G_{()}$ следующим образом:

$$G_{()} = (N, T \cup \{ (,) \}, P_{()}, S),$$

где $P_{()} = \{ A \rightarrow (\alpha) \mid A \rightarrow \alpha \in P \}$.

Таким образом, $L(G_{()})$ является линейно-скобочным кодированием деревьев вывода языка $L(G)$.

Определение 1. Две КС-грамматики G_1 и G_2 структурно эквивалентны, если $L(G_{1()}) = L(G_{2()})$.

В этом случае для каждого дерева вывода в одной КС-грамматике существует конгруэнтное дерево вывода того же слова в другой грамматике, т.е. дерево, отличающееся от первого разве что пометками не конечных вершин. Требуя соответствия уже не всех деревьев вывода для каждого выводимого слова, а лишь некоторых (но, по крайней мере, одного), можно ввести отношение частично структурной эквивалентности. Однако, в то время как проблема структурной эквивалентности КС-грамматик оказывается алгоритмически разрешимой [5], [6], проблема частичной структурной эквивалентности КС-грамматик алгоритмически неразрешима.

В работе [6] авторами представлен алгоритм для порождения структурно эквивалентных простых детерминированных грамматик, соответствующих данной КС-грамматике. Если такой грамматики не существует, то алгоритм заканчивает работу с указанием на этот факт. В [7] этот результат расширен до $LL(1)$ -грамматик без ϵ -правил (правила с пустым порождением).

Другие результаты исследований о структурной эквивалентности появились в работах [8] – Prather и [9] – Taniguchi & Kasami, в которых представлены преобразования КС-грамматики G_1 к структурно-эквивалентной КС-грамматике G_2 , которая имеет либо минимальное число нетерминальных символов, либо минимальное число правил.

Понятие с ограничениями большими, чем структурная эквивалентность, было введено Гинзбургом и Харрисоном в [10, 11].

Пусть G – КС-грамматика, в которой все правила перенумерованы; пусть $\Delta = \{ [i,]_i \mid 1 \leq i \leq |P| \}$. Определим скобочную версию $G_{[]}$:

$$G_{[]} = \{ N, T \cup \Delta, P_{[]}, S \},$$

где $P_{[]} = \{ A \rightarrow [i, \alpha]_i \mid A \rightarrow \alpha \text{ из } P \text{ с номером } i \}$

Определение 2. Две КС-грамматики G_1 и G_2 *сильно структурно эквивалентны*, если существуют перенумерованные правила в P_1 и P_2 , такие что при этой нумерации $L(G_{1[]}) = L(G_{2[]})$.

Каждое слово в языке $L(G_{[]})$ не только содержит достаточно информации, чтобы определить структуру дерева вывода (так же как в $L(G_{()})$), но также достаточно информации, чтобы пометить внутренние вершины дерева.

Проблема сильной структурной эквивалентности алгоритмически разрешима, так как существует только конечное число нумераций, а для всякой такой нумерации скобочная версия $G_{[\]}$ определяет простой детерминированный язык (S -язык). Преобразование $G_{[\]}$ в S -грамматику прямое. Кореньяк и Хопкрофт в [12] доказали, что проблема, порождают ли две S -грамматики один и тот же язык, алгоритмически разрешима. Сравнительно недавно в работе [13] этот результат был повторён.

Хотя скобочные грамматики $G_{()}$, $G_{[\]}$ и обеспечивают более строгое понятие грамматического подобия, все же они слишком ограничены: деревья выводов идентичны с точностью до меток нетерминалов.

Многие авторы пытались ослабить эти отношения грамматического подобия: примерами такого рода отношений грамматического подобия являются грамматические функторы и грамматические формы. Грамматические функторы (X -функторы) были введены Хотцем (Hotz) как специальные функторы на категориях, связанных с грамматиками общей фразовой структуры, а впоследствии рассматривались с более практической точки зрения – в связи с проблемой синтаксического анализа [15].

Грамматические формы впервые появились в работе Кремерс и Гинзбурга [14], где из основной (master) грамматики с помощью подстановок нетерминальных и терминальных символов получают другие грамматики, которые обладают теми же анализирующими свойствами, что и основная грамматика.

Существуют и другие примеры отношений грамматического подобия. В работе [16] – (Hunt, Rozenkrantz) многие из них обсуждаются с точки зрения сложности разбора.

2. Отношения гомоморфизма

В наиболее ранних работах можно найти примеры преобразований на КС-грамматиках, которые применялись в практических ситуациях при построении компиляторов (к примеру, устранение левой или правой рекурсий в правилах грамматики). Никаких общих определений отношений грамматического подобия в тех случаях дано не было, хотя большинство высказанных тогда идей легли в основу более поздних формальных понятий [17]

С целью выразить отношение грамматического подобия более точно, появилось много новых формальных понятий. В работах [18]–[21] даны определения и описаны результаты для грамматических гомоморфизмов, изоморфизмов, слабых покрытий и покрытий по Рейнольдсу. В этих определениях сделан акцент на правила грамматики, а не на множества выводов или грамматических разборов. Так отношение гомоморфной эквивалентности КС-грамматик, являясь более тонким, чем отношение структурной, сильно структурной и частично структурной эквивалентности может быть определено следующим образом:

КС-грамматика $G = (N, T, P, S)$ называется гомоморфно эквивалентной грамматике $G' = (N', T', P', S')$, если существует гомоморфизм $\psi: G \rightarrow G'$ и $L(G) = L(G')$. В этом случае гомоморфизм ψ представляется тройкой отображений $\psi_1: G \rightarrow G'$, $\psi_2: T \rightarrow T'$, $\psi_3: P \rightarrow P'$ и $\psi_1(S) = S'$.

В работе [18] Грей и Харрисон определили понятие грамматического покрытия, которое описывает отношение между множествами деревьев вывода двух КС-грамматик. Интерес авторов к этому понятию был основан исключительно на его применении в области синтаксического анализа языков.

Если ограничиться моделью анализа, в которой каждое предложение исходного языка даётся описанием его деревьев вывода в виде строки правил грамматики (или меток, идентифицирующих эти правила), тогда соответствие двух КС-грамматик, которое описывается грамматическим покрытием, является отношением между такими описаниями деревьев вывода для данного предложения.

Результат синтаксического анализа можно считать аргументом семантического отображения. В случае, когда КС-грамматика G' покрывает КС-грамматику G , разбор предложения w относительно G' может быть отображён гомоморфизмом на разбор той же w относительно G . Отсюда следует вывод, что язык $L(G)$ можно анализировать относительно преобразованной грамматики G' , а затем использовать первоначальное семантическое отображение, как на рисунке 1.

Неформально говоря, основная идея в том, что та КС-грамматика G , которая «трудна» для анализа или не удовлетворяет определённым ограничениям, диктуемым методом синтаксического анализа, преобразуется в КС-грамматику G' , которую можно анализировать более простым анализатором. Последующий синтаксический анализ проводится относительно преобразованной G' , а после этого результат анализа (разбора) отображается покрывающим гомоморфизмом на соответствующий разбор относительно G .

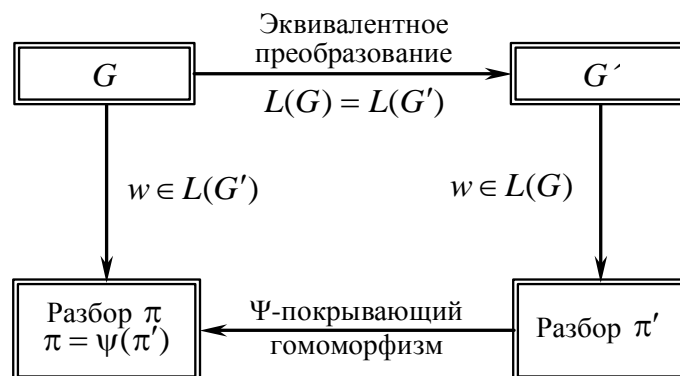


Рис. 2. Схема эквивалентного преобразования с использованием покрытий

Заключение

Подводя итог вышесказанному, можно сказать, что в том случае, если синтаксические правила, которые подаются на вход инструментальной системе, не специфицируют грамматику, для которой выбран тип анализатора может быть построен, возникают следующие ситуации:

- система диагностирует о своей неспособности построить анализатор и выдает информацию, почему она терпит неудачу; эта информация может быть использована разработчиком языка для того, чтобы изменить (не обязательно эквивалентно преобразовать) эти правила (и связанные с ними семантики) с целью сделать их подходящими для системы;
- система может применить эквивалентные преобразования к входной грамматике для того, чтобы сделать её адекватной методу. Но это должно делаться так, чтобы первоначальные семантики сохранялись;
- синтаксические правила на входе системы не специфицируют грамматику, которая годится для метода анализа, заложенного в инструментальной системе, но у разработчика есть возможность обеспечить систему дополнительной информацией (например, пополнить новыми правилами), достаточной для построения правильного анализатора.

Следует отметить, что вторая и третья альтернативы могут быть скомбинированы. Более того, если синтаксические правила (с дополнительной информацией или без неё) не специфицируют грамматику желаемого типа, то целесообразно, чтобы система могла принять решение, которое ведёт к построению анализатора. В этом случае пользователю не нужно заботиться о конечной форме грамматики. Но эквивалентные преобразования могут изменять структуру грамматики. Это означает то, что преобразованная грамматика не обязательно выполняет тот же самый перевод к семантическим действиям, что и первоначальная (исходная грамматика). Однако, если преобразование делается таким образом, что можно провести разбор в терминах исходной грамматики, тогда можно как бы «обмануть» пользователя инструментальной системы. Эта идея была проиллюстрирована в начале главы на рисунке 2.

Литература

1. Ахо А. В., Сети Р., Ульман Д. Д., Компиляторы: принципы, технологии и инструменты. – М.: Вильямс, 2001.
2. Федорченко Л.Н. Регуляризация контекстно-свободных грамматик. / LAP LAMBERT Academic Publishing GmbH & Co. KG Dudweiler Landstr. 99, 66123 Saarbrücken, Germany. 2011. С 180. ISBN: 978-3-8443-5360-0.
3. Федорченко Л.Н. Регуляризация контекстно-свободных грамматик на основе эквивалентных преобразований синтаксических граф-схем. // Труды СПИИРАН. 2010. вып.4(15). С. 213–230. ISSN 2078-9181.

4. Early J. Ambiguity and precedence in syntax description / *Acta Informatica* 4:2, 1975, pp. 183–192.
5. Mc Naughton, R. The development of formal language theory since 1956 / *Internat. J. Found. Comput. Sci.* (1990), p. 355–368.
6. Paull M. and Unger S. Structural equivalence of context-free grammars / *Jornal of Computer and System Sciences* 2 (1968), 427–463.
7. Paull M.C., and Unger S.H. Structural Equivalence of LL-k Grammars // *IEEE Conference Record of Ninth Annual Symposium on Switching and Automata Theory*, 1968 p. 160–175.
8. Prather R.E. Regular Expressions for Program Computations, *American Mathematical Monthly*, 104, No. 2, 1997.
9. Kenichi Taniguchi, Tadao Kasami: A Result on the Equivalence Problem for Deterministic Pushdown Automata / *J. Comput. Syst. Sci.* 13(1): 38–50 (1976)
10. Ginsburg S. A survey of grammar forms / *Acta Cybernetica*, 3(4):269–280, 1977.
11. Ginsburg S., Harrison Michael A.: Bracketed context free languages. (2005)
12. Korenjak A. J. and Hopcroft J. E. Simple deterministic languages // *IEEE Conf. Record of 7th Annual Symposium on Switching and Automata Theory*, 1966, pp.36–46.
13. Salomaa K. and Yu S. Decidability of structural equivalence of EOL grammars / *Theoretical Computer Science* 82 – 1991.
14. Cremers A B , and Ginsburg S., Context-free grammar forms / *J Comput Syst Sci* 11 (1975), 86–117.
15. Hotz G. Normal-form transformations of context-free grammars / *Acta Cybernetica*, 4(1):65–84, 1978.
16. Hunt III H. B. and Rosenkrantz D. J., Complexity of grammatical similarity relations // *In: Procs of a Conf. on Theoretical Computer Science Waterloo 1977*, 139–145.
17. Федорченко Л.Н., Мартыненко Б.К. Эквивалентные преобразования КСР грамматик в регулярной форме в практике построения языковых процессоров. Часть первая. Определение и распознавание КСР-языков посредством синтаксических граф-схем. – Ленинград, АН СССР, Ленинградский Научно-исследовательский Вычислительный Центр, 1983.
18. Gray J. N. and Harrison M. A., On the covering and. reduction problems for context free grammars. *Journal of the. ACM* 19, 675–698. 1972.
19. Ginsburg S., Harrison Michael A.: Bracketed context free languages. (2005)
20. Hunt III H. B. and Rosenkrantz D. J., Complexity of grammatical similarity relations // *In: Procs of a Conf. on Theoretical Computer Science Waterloo 1977*, 139–145.
21. Reynolds J.C., and Haskell R. Grammatical coverings. Unpub. memorandum, Syracuse University, 1970. From *ACM Transactions on Programming Languages and Systems (TOPLAS)*. V. 9 , Issue 4 (Oct 1987), pp.543–566 1987 ISSN:0164–0925.

Федорченко Людмила Николаевна, кандидат технических наук, старший научный сотрудник лаборатории прикладной информатики Федерального государственного бюджетного учреждения науки Санкт-Петербургский институт информатики и автоматизации Российской академии наук (СПИИРАН). Тел. +7 (812) 328-1919. E-mail: LNF@ias.spb.su

Fedorchenko Ludmila Nikolaevna, Senior Researcher, PhD, St.Petersburg Institute for Informatics and Automation of the Russian Academy of Sciences (SPIIRAS).

УДК 621.391

© М.В. Харинов

УЛУЧШЕНИЕ КАЧЕСТВА ПРИБЛИЖЕНИЯ ЦИФРОВОГО ИЗОБРАЖЕНИЯ НА ОСНОВЕ ИЕРАРХИЧЕСКОЙ СЕГМЕНТАЦИИ

В работе предлагается метод оптимизации качества кусочно-постоянного приближения цифрового изображения данным числом связанных сегментов посредством итеративного применения комбинированной операции слияния/разделения.

Ключевые слова: цветное изображение, сегмент, слияние, разделение, бинарная иерархия, среднеквадратичное отклонение, минимизация.

© M. V. Kharinov

IMPROVING THE QUALITY OF DIGITAL IMAGE APPROXIMATION BASED ON HIERARCHICAL SEGMENTATION

The paper proposes a method for optimizing quality of the piecewise constant approximation of a digital image with a given number of connected segments by an iterative applying of a combination of merge and split operations.

Keywords: color image, segment, merging, splitting, binary hierarchy, standard deviation, minimization.

Введение

Работа относится к области автоматического выделения объектов на цветных изображениях или *сегментации* изображений, которая понимается в классическом смысле разделения исходного изображения на сегменты, которые состоят из связанных пикселей [1-3]. Качество приближений изображения с усредненными по сегментам или *кластерам* из нескольких сегментов значениями пикселей, принято оценивать [4-5] по среднеквадратичному отклонению σ приближения от изображения или суммарной квадратичной ошибке $E = 3N\sigma^2$, где N — число пикселей в изображении, а коэффициент 3 учитывает число цветовых R, G, B компонент.

Если при данном числе кластеров достигается минимально возможное значение σ или E , то приближение изображения считается *оптимальным*. Вычисление оптимальных приближений цифрового изображения связными сегментами является, на сегодняшний день, нерешенной проблемой. При этом минимально достижимое значение ошибки аппроксимации E в расчетах оказывается неизвестным, а формальное сравнение результатов сегментации по значениям E или σ с неопределенной «точкой отсчета» недостаточно достоверным и компенсируется субъективной оценкой адекватности выделения зрительно наблюдаемых объектов. Для обеих оценок представляет интерес разработка методов оптимизации качества аппроксимации изображения по стандартному среднеквадратичному отклонению σ или E .

В статье предлагается вариант метода улучшения качества приближения за счет минимизации ошибки E аппроксимации изображения посредством слияния/разделения сегментов изображения, которые составляют бинарную иерархию.

1. Постановка задачи

Рассмотрим заданное разбиение цветного изображения на k сегментов. Заполнение сегментов средними значениями пикселей порождает *кусочно-постоянное приближение* изображения, качество которого характеризуется некоторым значением среднеквадратичного отклонения σ приближения от изображения и соответствующей суммарной квадратичной ошибкой E .

Полагается, что для каждого сегмента заданного разбиения определено множество вложенных сегментов, которые составляют бинарную иерархию. Иными словами, любой сегмент из $n > 1$ пикселей разделяется на пару вложенных, и иерархия состоит из $2n - 1$ сегментов. Разделение сегмента данного разбиения надвое преобразует разбиение в разбиение из $k + 1$ сегментов, а слияние пары смежных сегментов порождает новый сегмент и новое разбиение из $k - 1$ сегментов. Комбинированная операция слияния/разделения сегментов преобразует разбиение изображения в

новое разбиение с исходным числом k сегментов. Требуется снизить при этом ошибку аппроксимации E изображения кусочно-постоянным приближением, индуцируемым преобразованным разбиением.

2. Кластеризация и сегментация

Классическим методом, обеспечивающим эффективную минимизацию суммарной квадратичной ошибки E при иерархической кластеризации многомерных данных, является метод Уорда [6], в котором выполняется итеративное попарное слияние кластеров, отвечающих минимальному приращению $\Delta E_{merge} = \min$ суммарной квадратичной ошибки E . Выражение для приращения $\Delta E_{merge}(1, 2) = E(1 \cup 2) - E(1) - E(2)$ суммарной квадратичной ошибки E при слиянии кластеров 1 и 2 из n_1 и, соответственно, n_2 пикселей дается формулой:

$$\Delta E_{merge}(1, 2) = \frac{\|I_1 - I_2\|^2}{\frac{1}{n_1} + \frac{1}{n_2}} \geq 0, \quad (1)$$

где I_1, I_2 — трехмерные значения средних яркостей по цветовым компонентам изображения, а знак $\|\cdot\|^2$ обозначает евклидову норму их разности.

В результате применения метода Уорда [6] вычисляется последовательность разбиений множества пикселей на $N_0, N_0 - 1, \dots, 3, 2, 1$ кластеров, где N_0 — количество кластеров в некотором начальном разбиении множества пикселей изображения. Метод Уорда обеспечивает безусловную минимизацию суммарной квадратичной ошибки E посредством перебора всех пар кластеров. Вычислительная сложность метода с ростом N_0 возрастает квадратично. Поэтому, в случае изображений, для применения метода Уорда необходимо предварительно существенно уменьшить начальное число кластеров N_0 по сравнению с числом пикселей N , например, как в [7].

Если рассматриваемые пары кластеров в методе Уорда ограничить множеством *смежных* кластеров, имеющих пиксели с минимальным расстоянием друг от друга, то вычисления будут выполняться по модели сегментации Мамфорда–Шаха [8-12], конкретно, версии [11] (без учета границ между сегментами). При этом если в начальном разбиении кластеры пикселей совпадают с сегментами изображения, в частности, отождествляются с отдельными пикселями, то они совпадают с сегментами и в остальных разбиениях изображения, так как операция слияния смежных сегментов не выводит из множества сегментов.

По сравнению с методом Уорда безусловной кластеризации пикселей изображения модель Мамфорда–Шаха сегментации изображения обеспечивает *условную* минимизацию суммарной квадратичной ошибки посредством перебора ограниченного множества пар смежных сегментов, что влечет увеличение ошибки E аппроксимации изображения. Однако, модель Мамфорда–Шаха обеспечивает аппроксимацию изображения любым числом сегментов от 1 до N и снижает вычислительную сложность до линейной зависимости от числа N пикселей в изображении.

Для сохранения отмеченных преимуществ, но компенсации ошибки E аппроксимации изображения кусочно-постоянными приближениями, получаемыми в модели Мамфорда–Шаха или других моделях, и разработан метод улучшения качества сегментации цветового изображения.

3. Улучшение качества сегментации цветового изображения

Обсуждаемый метод улучшения качества приближения цветового изображения из k сегментов относится к итеративным методам. Логика вычислений подобна расчетам по классическому методу кластеризации K -средних [5, 6, 13, 14] или его современному обобщению « K -средних без средних» [15, 16]. При этом наш метод на каждой итерации уменьшает ошибку E аппроксимации изображения и завершается, когда уменьшение ошибки оказывается не выполнимо в силу нарушения установленного критерия продолжения обработки, иначе — при выполнении альтернативного «правила останова».

В отличие от других методов, в нашем методе полагается, что для каждого из k сегментов рассматриваемого разбиения изображения задается бинарная иерархия вложенных сегментов, которая для каждого сегмента как самостоятельного изображения рассчитывается «от пикселей» в алгоритме слияния пар смежных сегментов, например, в модели Мамфорда–Шаха. При этом любому

исходному или вложенному сегменту i , содержащему более одного пикселя, сопоставляется приращение $\Delta E_{divide}(i)$ суммарной квадратичной ошибки E , которое описывает приращение суммарной квадратичной ошибки при разделении сегмента i надвое и равно взятому с обратным знаком приращению $\Delta E_{merge}(i', i'')$ ошибки E при формировании сегмента i посредством слияния пары вложенных сегментов i' и i'' :

$$\Delta E_{divide}(i) \equiv -\Delta E_{merge}(i', i''), \tag{2}$$

где $i = i' \cup i''$. Сегментам, совпадающим с отдельными пикселями, сопоставляются нулевые приращения $\Delta E_{merge}(i)$.

В остальном метод тривиален. На данной итерации анализируются отрицательные приращения суммарной квадратичной ошибки, отвечающие разделению надвое сегментов текущего разбиения изображения, которые сравниваются с ее положительными приращениями при слиянии пар смежных сегментов этого разбиения. Если находится хотя бы один сегмент 1, разделение которого на пару вложенных сопровождается приращением суммарной квадратичной ошибки по абсолютной величине превышающему приращение суммарной квадратичной ошибки при слиянии смежных сегментов 2 и 3, отличных от 1, то операция разделения надвое сегмента 1 коммутрует со слиянием сегментов 2 и 3, и комбинация операций приводит к отрицательному приращению $\Delta E_{combine}(1, 2, 3)$ суммарной квадратичной ошибки E :

$$\Delta E_{combine}(1, 2, 3) = \Delta E_{divide}(1) + \Delta E_{merge}(2, 3) < 0. \tag{3}$$

При нахождении троек сегментов, удовлетворяющих (3), условие обработки считается выполненным, и производится разделение одного из сегментов в комбинации со слиянием пары других. Тройки преобразуемых сегментов выбираются из числа возможных так, чтобы максимально уменьшить суммарную квадратичную ошибку E .

Затем, для укрупненного сегмента и новых пар сегментов вычисляются приращения суммарной квадратичной ошибки, и процесс преобразования разбиения итеративно повторяется, пока значения $\Delta E_{combine}(1, 2, 3)$ не окажутся неотрицательными для всех троек анализируемых сегментов.

4. Результаты эксперимента

Рис.1 иллюстрирует улучшение качества сегментации изображения на примере стандартного цветового изображения «Лена». Слева показано исходное изображение, в центре — его начальное приближение из 100 сегментов, справа — улучшенное приближение с тем же самым количеством сегментов. Под приближениями указаны значения среднеквадратичного отклонения σ приближения от изображения.

Начальное приближение изображения (в центре) получено в скоростной авторской версии [12] модели Мамфорда–Шаха. В начальном приближении встречаются пиксели 100 цветов, т.к., в данном случае, каждый сегмент отличается от других по средней яркости.



Рис. 1. Улучшение качества приближения цветового изображения

Обработанное приближение (справа) по количеству сегментов и цветов совпадает с начальным, но отличается меньшим значением ошибки аппроксимации E и среднеквадратичного отклонения σ пикселей приближения от пикселей исходного изображения. Характерно, что формальная оценка улучшения качества приближения согласуется с визуальной оценкой — на обработанном приближении проявляется большее количество объектов, наблюдаются на исходном изображении

Заключение

Таким образом, в статье представлен метод улучшения качества приближения изображения при фиксированном числе сегментов, который применим для улучшения результатов сегментации, выполненной по различным алгоритмам.

Очевидным развитием метода является его применение для улучшения качества иерархической сегментации, которое по завершению уточнения деталей аналитической и программной реализации планируется обсудить в дальнейших работах.

Литература

1. Дуда Р., Харт П. Распознавание образов и анализ сцен. – М.: Мир, 1976. – 512 с.
2. Прэйт У. Цифровая обработка изображений. – М.: Мир, 1982. – 790 с.
3. Гонсалес Р., Вудс Р. Цифровая обработка изображений. – М.: Техносфера, 2005. – 1073 с.
4. Мандель И.Д. Кластерный анализ. – М.: Финансы и статистика, 1988. – 176 с.
5. Айвазян С.А., Бухштабер В.М., Енюков И.С., Мешалкин Л.Д. Прикладная статистика: Классификация и снижение размерности. – М.: Финансы и статистика, 1989. – 607 с.
6. Ward J.H., Jr. Hierarchical grouping to optimize an objective function. // J. Am. Stat. Assoc. 1963. – Vol. 58, – Issue 301, – P. 236-244.
7. Харинов М.В. Альтернатива иерархическому методу Оцу для цветового изображения // Вестн. Бурят. гос. ун-та. – 2014. – №9. – С. 64-72.
8. Mumford D., Shah J. Optimal Approximations by Piecewise Smooth Functions and Associated Variational Problems // Communications on Pure and Applied Mathematics. 1989. – Vol. XLII, – № 4. – P. 577-685.
9. Koeperle G., Lopez C., Morel J. A Multiscale Algorithm for Image Segmentation by Variational Method // SIAM Journal on Numerical Analysis. 1994. – Vol. 31, – № 1, – P. 282-299.
10. Redding N.J., Crisp D.J., Tang D.H., Newsam G.N. An efficient algorithm for Mumford–Shah segmentation and its application to SAR imagery // Proc. Conf. Digital Image Computing Techniques and Applications (DICTA '99). 1999. – P. 35-41.
11. Бугаев А.С., Хельвас А.В. Поисковые исследования и разработка методов и средств анализа и автоматического распознавания потоковой информации в глобальных информационных системах. Шифр «Ладкан» // Отчет по НИР. – М.: Изд-во МФТИ, 2001. – Т. 1. – 140 с.
12. Kharinov M.V. Adaptive Dichotomous Image Segmentation Toolkit // Pattern Recognition and Image Analysis: Advances in Mathematical Theory and Applications. – 2012. – Vol. 22. – № 1. – P. 228-235.
13. Steinhaus H. Sur la division des corps materiels en parties // Bull. Acad. Polon. Sci., 1956. C1. III – Vol. IV. – P. 801-804.
14. Jain A.K. Data Clustering: 50 Years Beyond K–Means // Pattern Recognition Letters, 2010. – Vol. 31. – № 8. – P. 651-666.
15. Харинов М.В. Обобщение трех подходов к оптимальной сегментации цифрового изображения // Труды СПИИРАН. 2013. – Вып. 2 (25). – С. 294-316.
16. Dvoenko S.D. Meanless k -means as k -meanless clustering with the bi-partial approach // Pattern Recognition and Information Processing (PRIP'2014) / Proc. of the 12th Int. Conf., Minsk, 2014. – P. 50-54.

Харинов Михаил Вячеславович, кандидат технических наук, старший научный сотрудник лаборатории Прикладной информатики Санкт–Петербургского института информатики и автоматизации, тел. (812) 328-1919, E-mail: khar@iias.spb.su

Kharinov Mikhail Vyacheslavovich, candidate of technical sciences, senior researcher of Laboratory of Applied Informatics of St. Petersburg Institute for Informatics and Automation of RAS, ph. (812) 328-1919, E-mail: khar@iias.spb.su

3. Функциональный анализ и дифференциальные уравнения

УДК 517.956

© В.В. Кибирев

ФОРМУЛА ГРИНА В ТЕОРИИ ПОТЕНЦИАЛА

В данной работе рассматривается формула Грина и ее приложения. При помощи этой формулы исследуется поведение потенциалов простого и двойного слоев и их производных при переходе некоторой точки через заданную поверхность.

Ключевые слова: формула Грина, гармоническая функция, потенциалы простого и двойного слоев.

© V.V. Kibirev

GREEN'S FORMULA IN THE THEORY OF POTENTIAL

In the article Green's formula and its applications are considered. By means of this formula the behavior of simple and double layers potentials and their derivatives are studied while passing some point through the given surface.

Keywords: Green's formula, harmonic function, simple and double potentials.

Введение

Важным разделом теории уравнений с частными производными является теория краевых задач для эллиптических уравнений и систем уравнений. Среди таких задач наибольший интерес представляют так называемые нефредгольмовые краевые задачи, исследование которых, как правило, сводится к изучению сингулярных интегральных уравнений, причем для этих задач нарушается альтернатива Фредгольма. Благодаря разработанности теории одномерных сингулярных интегральных уравнений [5,6] краевые задачи для эллиптических уравнений с двумя независимыми переменными в настоящее время полностью изучены [2,6], что нельзя сказать о краевых задачах для эллиптических уравнений с многими независимыми переменными. Ряд важных вопросов в этой области не решен до сих пор, так как нет достаточно общих методов исследований.

Поэтому важно рассмотреть некоторые вопросы теории линейных эллиптических уравнений в трехмерном случае и сделать некоторые обобщения на многомерный случай.

Постановка задачи

В теории потенциала важную роль играет формула Грина, являющаяся следствием формулы Гаусса-Остроградского [3]. Ее можно написать в двух следующих видах:

$$\int_D (u_x v_x + u_y v_y + u_z v_z) d\omega + \int_D v \Delta u d\omega = \int_\Gamma v \frac{\partial u}{\partial n} d\sigma, \text{ или} \quad (1)$$

$$\int_D (u \Delta v - v \Delta u) d\omega = \int_\Gamma \left(u \frac{\partial v}{\partial n} - v \frac{\partial u}{\partial n} \right) d\sigma,$$

где $d\omega = dx dy dz$ – элемент объема, $d\sigma$ – элемент площади поверхности Γ , $\frac{\partial}{\partial n}$ – дифференцирование по направлению внешней относительно D нормали. Вторая формула (1) следует из первой. В первой формуле предполагается, что функции u и v непрерывны в $D \cup \Gamma$, первые производные v и вторые производные u непрерывны в D , а первые производные u непрерывны в $D \cup \Gamma$. Во второй формуле предполагается, что u , v и их первые производные непрерывны в $D \cup \Gamma$, а вторые производные непрерывны в D . Считается, что граница Γ области D кусочно-гладкая и существуют все интегралы, фигурирующие в выражениях.

Доказательства основных теорем

Если в первой формуле (1) положим $\Delta u = 0$, $v = 1$, то получим следующее утверждения.

Теорема 1. Если гармоническая функция регулярна в ограниченной области D и непрерывно дифференцируема в замкнутой области $D \cup \Gamma$, то интеграл по поверхности Γ от ее нормальной производной равен нулю.

Теорема 2. Пусть задан кусок поверхности Γ , ограничиваемой кривой C , и точка P , не принадлежащая Γ . Тогда потенциал двойного слоя с постоянной плотностью $\sigma = 1$ поверхности Γ в точке P по абсолютной величине равен телесному углу, под которым кривая C видна из точки P . В частности, потенциал двойного слоя поверхности, ограничивающей область D , имеет постоянное значение -4π во всех внутренних точках D , а вне D равен нулю.

Доказательство. Построим коническую поверхность H , образованную отрезками прямых, соединяющими точку P с точками кривой C . Рассмотрим сферу $K(\varepsilon)$ радиуса ε с центром в точке P . Поверхности Γ , H и $K(\varepsilon)$ ограничивают некоторую область $\Omega(\varepsilon)$. В этой области функция r^{-1} регулярна, и на поверхности H выполняется равенство $\frac{\partial}{\partial n}(r^{-1}) = 0$. В силу теоремы 1 имеем

$$\int_{\Gamma} \frac{\partial}{\partial n} \left(\frac{1}{r} \right) dS + \int_H \frac{\partial}{\partial n} \left(\frac{1}{r} \right) dS + \int_{K(\varepsilon)} \frac{\partial}{\partial n} \left(\frac{1}{r} \right) dS = 0.$$

Так как $\frac{\partial}{\partial n}(r^{-1}) = -\varepsilon^{-1}$ на $K(\varepsilon)$, последний интеграл в этой формуле вычисляется явно и равен углу при вершине построенной конической поверхности Γ . Для того чтобы доказать второе утверждение теоремы, заметим, что знак этого угла однозначно определяется заданием положительной и отрицательной сторон поверхности Γ , а затем поверхность Γ разобьем на две некоторой кривой C . Теперь для внутренних точек D углы складываются, а для внешних вычитаются.

Если и в первой формуле (1) положим $u = v$, $\Delta u = 0$, то получим тождество

$$D(u) \equiv \int_D (u_x^2 + u_y^2 + u_z^2) d\omega = \int_{\Gamma} u \frac{\partial u}{\partial n} dS, \quad (2)$$

справедливое для любой регулярной в D гармонической и непрерывно дифференцируемой в $D \cup \Gamma$ функции u . Интеграл $D(u)$ называется интегралом Дирихле. Из тождества (2) вытекает справедливость следующего утверждения.

Теорема 3. Пусть u - регулярная в области D непрерывно дифференцируемая в замкнутой области $D \cup \Gamma$ гармоническая функция. Тогда: а) если u обращается в нуль на границе Γ области D , то она обращается в нуль тождественно в D ; б) если нормальная производная $\frac{\partial u}{\partial n}$ обращается в нуль на границе, то u постоянна в D .

Пусть во второй формуле (1) область D является шаром радиуса R с центром в точке P . Рассмотрим концентрический шар B радиуса $R_0 < R$, положим $v = r^{-1}$ и напишем эту формулу для сферического слоя, ограничиваемого границей Γ шара D и границей Σ шара B . Тогда для любой гармонической регулярной в D функции u с учетом теоремы 1. Получим

$$\frac{1}{4\pi R_0^2} \int_{\Sigma} u dS = \frac{1}{4\pi R^2} \int_{\Gamma} u dS.$$

Воспользовавшись теоремой о среднем и устремив R_0 к нулю, получим формулу

$$u(P) = \frac{1}{4\pi R^2} \int_{\Gamma} u(Q) dS, \tag{3}$$

выражающую так называемую теорему о среднем для гармонических функций. Хотя при выводе формулы (3) функция u предполагалась дифференцируемой в замкнутом шаре $D \cup \Gamma$, нетрудно показать, что для справедливости формулы достаточно непрерывности u в $D \cup \Gamma$. При помощи формулы (3) можно доказать следующее утверждение.

Теорема 4. Пусть гармоническая функция u регулярна в области D и непрерывна в замкнутой области $D \cup \Gamma$. Тогда она принимает свои максимальное и минимальное значения на границе Γ области D ; максимум и минимум достигаются внутри области тогда и только тогда, когда u постоянна.

Утверждение теоремы называется принципом максимума для гармонических функций.

В формулах Грина (1) предполагалось, что функции v и u имеют в области D все производные, фигурирующие в этих формулах. Теперь возьмем функцию

$$v(\xi, \eta, \zeta) = r^{-1} + \omega(\xi, \eta, \zeta)$$

с характеристической особенностью в фиксированной точке $P = (x, y, z)$. Функция ω предполагается непрерывной и непрерывно дифференцируемой в замкнутой области $D \cup \Gamma$ и дважды дифференцируемой в D . Окружим точку P шаром $\Sigma(\varepsilon)$ достаточно малого радиуса $\varepsilon > 0$ центром в точке P . Из области D выбросим пересечение $D \cap \Sigma(\varepsilon)$, а для оставшейся части $D(\varepsilon)$ области D напишем формулы (1) и устремим затем ε к нулю. В том случае, когда точка P лежит на Γ , будем предполагать, что Γ имеет непрерывно меняющуюся касательную плоскость в окрестности точки P . Совершив предельный переход при $\varepsilon \rightarrow 0$, получим

$$\int_D (u_{\xi} v_{\xi} + u_{\eta} v_{\eta} + u_{\zeta} v_{\zeta}) d\omega + \int_D u \Delta v d\omega = \lambda u + \int_{\Gamma} u \frac{du}{dn} dS, \tag{4}$$

$$\int_D (u \Delta v - v \Delta u) d\omega = \lambda u + \int_{\Gamma} (u \frac{\partial v}{\partial n} - v \frac{\partial u}{\partial n}) dS,$$

где $d\omega$ - элемент объема, а

$$\lambda = \begin{cases} 4\pi, & P \in D \\ 2\pi, & P \in \Gamma \\ 0, & P \notin D \cup \Gamma \end{cases}.$$

На функции u и w накладываются те же ограничения, что и на функции u и v в формулах (1).

Если во второй формуле (4) положим $\omega = 0$, то для любой дважды дифференцируемой в области D и непрерывно дифференцируемой в $D \cup \Gamma$ функции u получим представление

$$u = -\frac{1}{4\pi} \int_D \frac{\Delta u}{r} d\omega + \frac{1}{4\pi} \int_{\Gamma} \frac{1}{r} \frac{\partial u}{\partial n} dS - \frac{1}{4\pi} \int_{\Gamma} u \frac{\partial}{\partial n} \left(\frac{1}{r}\right) dS \tag{5}$$

в виде суммы трех потенциалов. Для гармонической функции имеем $\Delta u = 0$, поэтому формула принимает вид

$$u = \frac{1}{4\pi} \int_{\Gamma} \frac{1}{r} \frac{\partial u}{\partial n} dS - \frac{1}{4\pi} \int_{\Gamma} u \frac{\partial}{\partial n} \left(\frac{1}{r}\right) dS, \tag{5a}$$

т.е. гармоническую функцию можно выразить в виде суммы потенциалов простого и двойного слоев. Если во второй формуле (4) будем считать $\Delta \omega = 0$, а в качестве v возьмем произвольное фундаментальное решение Ω уравнения Лапласа, то получим формулу

$$u = \frac{1}{4\pi} \int_{\Gamma} \Omega \frac{\partial u}{\partial n} dS - \frac{1}{4\pi} \int_{\Gamma} u \frac{\partial \Omega}{\partial n} \left(\frac{1}{r}\right) dS, \tag{6}$$

где $\Omega = r^{-1} + \omega$ фундаментальное решение с особенностью в точке P .

Обобщения формул (1) на многомерный случай следующие:

$$\int_D \left(\sum_{i=1}^n \frac{\partial u}{\partial x_i} \frac{\partial v}{\partial x_i} + v \Delta u \right) d\omega = \int_{\Gamma} v \frac{\partial u}{\partial n}, \quad (7)$$

$$\int_D (u \Delta v - v \Delta u) d\omega = \int_{\Gamma} \left(u \frac{\partial v}{\partial n} - v \frac{\partial u}{\partial n} \right) dS,$$

а формулы (4) принимают вид

$$\int_D \left(\sum_{i=1}^n \frac{\partial u}{\partial \xi_i} \frac{\partial v}{\partial \xi_i} + u \Delta w \right) d\omega = \lambda u + \int_{\Gamma} u \frac{\partial v}{\partial n} dS,$$

$$\int_D (u \Delta w - v \Delta u) d\omega = \lambda u + \int_{\Gamma} \left(u \frac{\partial v}{\partial n} - v \frac{\partial u}{\partial n} \right) dS,$$

$$\lambda = \begin{cases} \omega_n, & P \in D, \\ \frac{\omega_n}{2}, & P \in \Gamma, \\ 0, & P \notin D \cup \Gamma \end{cases}$$

$$\omega_n = \frac{2(\sqrt{\pi})^n}{\Gamma(\frac{n}{2})}, \quad v = \frac{1}{n-2} r^{2-n} + \omega.$$

При помощи формулы Грина исследуем поведение потенциалов простого и двойного слоев и их производных при переходе точки $P = (x, y, z)$ через поверхность Γ . Пусть точка P_0 лежит на Γ и поверхность Γ в этой точке имеет непрерывные главные кривизны. С центром в точке P_0 построим шар $\Sigma(p)$ настолько малого радиуса, чтобы пересечение его с Γ было связным куском гладкой поверхности. Обозначим через D ту часть шара $\Sigma(p)$, для которой положительное направление нормали к Γ является направлением внешней относительно D нормали. Плотности потенциалов простого и двойного слоя будем считать непрерывными, по Гельдеру.

Теорема 5. При переходе через поверхность Γ значение потенциала двойного слоя в точке P_0 имеет скачок, описываемый формулами

$$u^+(P_0) - u(P_0) = 2\pi\sigma(P_0), \quad u^-(P_0) - u(P_0) = -2\pi\sigma(P_0),$$

где u^+ - предел потенциала двойного слоя и при стремлении точки P к точке P_0 с положительной стороны поверхности Γ , а u^- - аналогичный предел при стремлении P к P_0 с отрицательной стороны поверхности Γ .

Потенциал простого слоя непрерывен при переходе через точку P_0 , а его нормальная производная имеет скачок

$$\left[\frac{\partial u}{\partial n} \right] = \frac{\partial u}{\partial n^+} - \frac{\partial u}{\partial n^-} = 4\pi\rho(P_0),$$

где $\frac{\partial}{\partial n^+}$ - дифференцирование по направлению положительной нормали к поверхности Γ в точке

P_0 , а $\frac{\partial}{\partial n^-}$ - дифференцирование по направлению отрицательной нормали.

Доказательство. Утверждение теоремы докажем для того случая, когда плотность σ потенциала двойного слоя дважды дифференцируема и допускает продолжение до дважды дифференцируемой в области D функции $\sigma(x, y, z)$. Положим в первой формуле (4) $u = \sigma$, $w = 0$. Имеем

$$\lambda\sigma + \int_{\Gamma} \sigma \frac{\partial}{\partial v} \left(\frac{1}{r} \right) dS = \int_D \left[\sigma_{\xi} \frac{\partial}{\partial \xi} \left(\frac{1}{r} \right) + \sigma_{\eta} \frac{\partial}{\partial \eta} \left(\frac{1}{r} \right) + \sigma_{\xi} \frac{\partial}{\partial \xi} \left(\frac{1}{r} \right) \right] d\omega.$$

Правая часть этого равенства непрерывна при переходе точки P через поверхность Γ , поэтому должна быть непрерывной и левая часть, а непрерывность выражения

$$\lambda\sigma + \int_{\Gamma} \sigma \frac{\partial}{\partial \nu} \left(\frac{1}{r} \right) dS$$

и есть первое утверждение теоремы.

Второе утверждение теоремы доказывается аналогично, только функция u выбирается так, что на Γ она тождественно равна нулю, а $\frac{\partial u}{\partial n} = \rho$, где ρ - плотность потенциала простого слоя.

Заключение

В данной работе с помощью формулы Грина исследовано поведение гармонической функции в заданной области и на ее границе.

Литература

1. Бицадзе А.В. Некоторые классы уравнений в частных производных. – М.: «Наука», 1981. – 448 с.
2. Гахов Ф.Д. Краевые задачи. – М.: «Наука», 1977. – 640 с.
3. Гурса Э. Курс математического анализа. – М.-Л.: ОНТИ НКТП СССР, 1936. – Т.1. – 592 с.
4. Кибирев В.В. К задаче о наклонной производной с линейными коэффициентами для гармонических функций. – Диф. уравнения. – 1980. – Т.16. – №1. – С.80-85.
5. Михлин С.Г. Многомерные сингулярные интегралы и интегральные уравнения. – М.: Физматиз., 1962. – 254 с.
6. Мухелишвили Н.И. Сингулярные интегральные уравнения. – М.: Наука, 1968. – 512с.
7. Янушаускас А. К задаче о наклонной производной для гармонических функций трех независимых переменных. // Сиб. матем. журнал. – 1967. – Т.8. – №2. – С.447-462.
8. Янушаускас А. Аналитические и гармонические функции многих переменных. – Новосибирск: Наука, 1981. – 184 с.

Кибирев Владимир Васильевич, кандидат физико-математических наук, профессор кафедры прикладной математики Бурятского государственного университета, тел.(8301-2)217573, dekanat_imi@bsu.ru

Kibirev Vladimir Vasilievich, candidate of physical and mathematical sciences, the professor of the applied mathematics department of the Buryat State University.

4. Математическое моделирование

УДК 519.711.3

© В.Е. Архинчев, Е.Ю. Андреева

ВВЕДЕНИЕ В КВАНТОВЫЕ ВЫЧИСЛЕНИЯ И УПРАВЛЕНИЕ ДВИЖЕНИЕМ ЧАСТИЦЫ В МАГНИТНОМ ПОЛЕ¹

В рамках решения проблемы квантовых вычислений рассмотрена задача оптимального управления движением частицы в магнитном поле. Рассмотрены различные воздействия, для которых получены оценки времени достижения заданной точки. Показано, что при движении в магнитном поле (по окружности) время достижения заданной точки траектории зависит как от амплитуды воздействия, так и от спектра воздействия вынуждающей силы. Выполнены расчеты времени достижения для некоторых видов воздействия (управляющего поля).

Ключевые слова: квантовые вычисления, управление, движение частиц, магнетизм.

© V.E. Arkhincheev, E.Y. Andreeva

INTRODUCTION INTO QUANTUM COMPUTING AND PARTICAL MOTION CONTROL IN A MAGNETIC FIELD

As part of the solution of the quantum computing problem the optimal control problem of particle motion in a magnetic field was considered. Various impacts with estimation reach time obtained were considered. Specified trajectory point reach time while the circumferential motion in a magnetic field was shown to depend on both the impact amplitude and the driving force impact spectrum. Calculations of reach time for certain types of impact (control field) were made.

Keywords: quantum computing, control, particle motion, magnetism.

Введение

В настоящее время, как за рубежом, так и в России ведутся работы по поиску материалов для разработки кубитов как основы элементной базы для квантовых вычислений. Работа ведется по нескольким направлениям, но наиболее перспективным из них представляется возможность построения кубитов (элементов квантовых вычислений) на базе низкоразмерных квантовых полупроводниковых систем с квантовыми точками [1]. Наряду с разработкой перспективных материалов важной задачей является разработка как новых, так и адаптированных под конкретные квантовые наноструктуры алгоритмов и схем квантовых вычислений, позволяющих реализовать логические схемы и элементы. С 1994 г., когда Шор [2] предложил эффективный алгоритм полиномиального (степенного) типа для решения задачи факторизации больших чисел на квантовых компьютерах, началась интенсивная разработка алгоритмов квантовых вычислений. Существенной особенностью квантовых компьютеров является возможность осуществления Фурье-преобразования за полиномиальное (степенное) число операций в зависимости от размера задачи. В этом состоит принципиальное отличие квантовых компьютеров от классических, где для этого необходимо экспоненциальное число операций. К настоящему моменту известны три класса квантовых алгоритмов. Первый класс алгоритмов предназначен для решения задач типа факторизации чисел, второй класс алгоритмов разработан для моделирования квантовых явлений [3], третий класс алгоритмов предназначен для поиска объекта в неструктурированной базе данных [4].

В настоящей работе обсуждается возможность управления квантовыми спиновыми состояниями в квантовых точках как основы построения логических элементов на базе квантовых полупроводниковых систем на примере эквивалентной задачи о движении частицы в магнитных полях. Эта задача является хорошо известной задачей и поэтому может быть использована для моделирования более сложных задач, в частности, для моделирования задач управления движением с помощью внешних воздействий. Кроме того, эта задача эквивалентна задаче управления переносом

¹ Работа выполнена при финансовой поддержке проекта №3484 МОиН РФ

спина [5, 7] и поэтому представляет значительный интерес для дальнейшего изучения проблемы переноса квантовой информации [6] и в целом квантовых вычислений. В связи с этим детальное исследование представляется необходимым для дальнейшего изучения проблемы передачи квантовой информации, в т. ч. и путем переноса спина.

Статья построена следующим образом. В п. 1 рассмотрено движение частицы (спина) в магнитном поле; получены общие выражения. В п. 2 проанализированы критерии оптимального выбора времени достижения и выполнены вычисления для конкретных видов управления: от импульсного до степенного и гауссового спектров воздействий вынуждающей силы. В п. 3 обсуждены полученные результаты.

1. Постановка задачи. Уравнения движения в магнитном поле с управлением.

Как известно, движение частицы (спина) в магнитных полях описывается следующим уравнением:

$$m \frac{dV}{dt} = \frac{q}{c} [V \times B] + U(t) \frac{q}{c} [V \times h],$$

где V – скорость частицы, B – постоянное магнитное поле, m – масса частицы, q – заряд частицы, $H(t) = U(t)h$ – переменное магнитное поле, h – единичный вектор, $U(t)$ – модуль управления.

Рассматривая уравнение по компонентам, и вводя вектор-функцию состояния $X = (x_1, x_2)$, $X \in R^2$, $U \in R$, сформулируем следующую модель управляемой системы:

$$\dot{X} = (A + U(t)C)X,$$

где матрицы A и C определяются следующими выражениями: $A = \begin{pmatrix} 0 & \Omega \\ -\Omega & 0 \end{pmatrix}$, $C = \begin{pmatrix} 0 & \omega \\ -\omega & 0 \end{pmatrix}$, а

параметры Ω и ω – следующими: $\Omega = \frac{qB}{mc}$ и $\omega = \frac{qh}{mc}$.

Как уже отмечалось выше, эта модель описывает различные физические задачи, в частности, движение частицы в магнитном поле и прецессию спина в магнитном поле. С точки зрения теории управления сформулированная задача относится к классу вырожденных задач оптимального управления, поскольку в ней имеются дополнительные инварианты – интегралы движения в соответствии с законом сохранения энергии:

$$X_1^2 + X_2^2 = const, \quad (m = const)$$

Задача управления движением частицы в магнитном поле формулируется следующим образом: определить время достижения заданной точки в зависимости от параметров управления – амплитуды и времени управляющего импульса, а также от спектра внешних воздействий. Для этого строится функционал описывающий достижение заданной точки в пространстве:

$$L(T_f) = 1 - \left((x_1(T_f), x_1^f) + (x_2(T_f), x_2^f) \right) \tag{1}$$

Здесь $X^f = (x_1^f, x_2^f)$ – вектор-функция конечного состояния, $X(T_f) = (x_1, x_2)$ – вектор-функция текущего состояния. Оптимальное время достижения заданной точки определяется из условия экстремума (минимума) функционала

Движение частиц в магнитном поле при различных внешних воздействиях. Магистральные решения и зависимость фазы частиц от амплитуды, продолжительности и спектра воздействия.

В качестве примера рассмотрим движение частицы по окружности. Наиболее просто задача решается в комплексном виде. Для этого введем комплексную переменную $z = x_1 + ix_2 = |z|e^{-i\varphi}$, тогда система уравнений движения преобразуется к следующему простому виду:

$$\begin{aligned} \frac{d|z|}{dt} &= 0, \\ \frac{d\varphi}{dt} &= \Omega + U(t) \end{aligned} \tag{2}$$

Первое уравнение соответствует закону сохранения энергии (сохранению амплитуды движения). Второе уравнение описывает изменение фазы при движении по окружности под действием управления. Таким образом, движение по окружности при заданном радиусе полностью определяется значениями его фазы. В свою очередь изменение фазы определяется внешним воздействием.

Согласно уравнению (2) общее решение имеет вид:

$$\varphi(t) = \Omega t + \omega \int U(t) dt \quad (3)$$

2. Управление движением

Согласно сформулированным выше условиям для оптимального управления определения времени достижения заданной точки необходимо построить функционал (1).

Выберем в качестве текущей точку на окружности

$$z = x_1 + ix_2 = |z| e^{-i\varphi},$$

где фаза определяется уравнением (3). В качестве конечной точки достижения точку на окружности под углом 45° :

$$z^* = x_1^* + ix_2^* = |z| e^{-i\pi/4}$$

Тогда искомым функционал примет вид:

$$L(T_f) = 1 - \frac{y^2}{2} \left(1 + \sin \left\{ 2 \left(\omega t + \int U(t) dt \right) \right\} \right)$$

Соответственно, условие минимума функционала определяется условием равенства фазы:

$$\varphi(T_F) = \omega T_F + \int_0^{T_F} U(t) dt = 2\pi \quad (4)$$

Таким образом, время оптимального достижения заданной точки на траектории T_f определяется функциональным уравнением (4).

Зависимость фазы частиц от амплитуды, продолжительности и спектра воздействия

В отсутствие внешнего воздействия $U(t)$ получим обычное стандартное решение с начальной угловой скоростью Ω :

$$\begin{cases} V_x(t) = V_0 \cos(\Omega t + \varphi) \\ V_y(t) = V_0 \sin(\Omega t + \varphi) \end{cases}$$

Тогда время достижения

$$T_F = \frac{\Delta\varphi}{\omega} = \frac{\pi}{4\omega}.$$

В математике эти решения известны как *магистральные* решения [5, 7].

Ниже мы проанализируем зависимость фазы движения частицы по окружности от вида, продолжительности управления и вычислим время достижения. Рассмотрим движение частицы при различных видах внешнего управления (управления).

1) В случае импульсных внешних воздействий функция управления имеет вид:

$$\begin{aligned} U(t) &= U_0 \theta(t - \varepsilon) - \theta(t) \text{ либо} \\ U(t) &= U_0 \sum_n (t - n\tau) - \theta(t). \end{aligned}$$

Тогда время достижения T_F может быть найдено из следующего выражения:

$$\begin{aligned} 2\pi &= \omega T_F + u_0 \tau. \\ T_F &= \frac{2\pi - U_0 \tau}{\omega} = \frac{2\pi}{\omega} - \tau \left(\frac{\omega_0}{\omega} \right). \end{aligned}$$

Время достижения уменьшается по сравнению со случаем отсутствия воздействия за счет амплитуды воздействия ω_0 . При выполнении условия

$$\omega_0 \tau = 2\pi$$

достижение будет мгновенным:

$$T_F = 0.$$

2) В случае экспоненциально затухающих воздействий функция управления имеет вид:

$$U(t) = U_0 e^{-\alpha t}.$$

В этом случае время достижения T_F может быть найдено следующим образом:

$$2\pi = \omega_0 T_F + \frac{U_0}{2} \left[1 - e^{(-2T)} \right].$$

Рассмотрим случаи:

а) $\alpha T_F \ll 1$, тогда $T_F = \frac{\pi/4}{\omega_0 + \alpha}$.

В случае адиабатического воздействия, время достижения увеличивается.

В случае быстрых воздействий

б) $\alpha T_F \gg 1$, тогда $T_F = \frac{\pi/4 - \frac{U_0}{\alpha}}{\omega}$.

В случае Гауссова воздействия функция управления имеет вид:

$$U(t) = U_0 e^{(-\beta^2 t^2)}.$$

$$\int_0^{T_F} U(t) dt = \sqrt{\frac{\pi}{\beta}} \operatorname{erfc}(\beta T_F),$$

где $\operatorname{erfc}(x) = \int_0^x e^{-x^2} dx$.

Рассмотрим следующие случаи:

а) $\beta T_F \ll 1$, тогда $(\omega + \beta) T_F = \frac{\pi}{4}$, отсюда $T_F = \frac{\pi}{4(\omega + \beta)}$.

б) $\beta T_F \gg 1$, тогда $\omega T_F + \frac{U_0 \sqrt{\pi}}{\beta} = \frac{\pi}{4}$ и $T_F = \frac{\frac{\pi}{4} - \frac{U_0 \sqrt{\pi}}{\beta}}{\omega}$.

3) В случае степенного распределения функция управления имеет вид:

$$U(t) = U_0 \left(\frac{t}{\tau} \right)^\gamma, \quad \gamma > 0.$$

$$\int_0^{T_F} U(t) dt = \frac{U_0}{\gamma + 1} \left(\frac{t}{\tau} \right)^{\gamma+1} = \frac{1}{\gamma + 1} \frac{U_0}{\tau^\gamma} T_F^{\gamma+1}$$

$$\frac{\pi}{4} = \frac{U_0}{(\gamma + 1)\tau^\gamma} T_F^{\gamma+1} + \omega T_F,$$

$$\left(\omega_0 \tau \right) \left(\frac{T_F}{\tau} \right)^{1-|\gamma|} + \omega T_F = \frac{\pi}{4}, \quad -\infty < \gamma \leq -1.$$

Рассмотрим три случая:

а) $-\infty < \gamma < -1$, тогда:

$$\omega_0 \tau \left(\frac{T_F}{\tau} \right)^{1-|\gamma|} + \omega T_F = \frac{\pi}{4},$$

$$-\omega_0\tau\left(\frac{T_F}{\tau}\right)^{1-|\gamma|} \ll \omega T_F,$$

тогда $T_F = \frac{\pi}{4\omega}$ при сильных воздействиях.

В обратном случае

$$-\omega_0\tau\left(\frac{T_F}{\tau}\right)^{1-|\gamma|} \gg \omega T_F,$$

тогда $\frac{T_F}{\tau} = \left(\frac{\pi}{4\omega_0\tau}\right)^{\frac{1}{|\gamma|-1}}$ при слабых воздействиях.

$$T_F \rightarrow \infty.$$

На малых временах достижений зависимость возрастает.

б) $-1 < \gamma < 0$, тогда:

$$(\omega_0\tau)\left(\frac{T_F}{\tau}\right)^{\gamma+1} + \omega T_F = \frac{\pi}{4}.$$

На малых временах время достижения не зависит от спектра распределения по временам:

$$(\omega_0\tau)\left(\frac{T_F}{\tau}\right)^{1-|\gamma|} + \omega T_F = \frac{\pi}{4}.$$

$$\text{При } T_F \rightarrow \infty \quad T_F = \left(\frac{\pi}{4\omega_0\tau}\right)^{\frac{1}{1-|\gamma|}}.$$

На больших временах время достижения определяется спектром распределения.

в) $\gamma > 0$

Тогда $(T_F)^{\gamma+1} + \omega T_F = \frac{\pi}{4}$, откуда следует, что

$$T_F = \tau \left(\frac{\pi}{4\omega_0\tau}\right)^{\frac{1}{\gamma+1}}.$$

На малых временах при слабом воздействии время достижения нечувствительно к спектру и тогда $T_F = \frac{\pi}{4\omega}$.

На больших временах при сильном воздействии время достижения определяется видом спектра.

Заключение

Таким образом, в настоящей статье рассмотрено движение частиц, как без внешнего воздействия, так и под действием внешних полей. Исследована возможность управления движения и достижения заданной точки при различных видах внешних воздействий. Установлена зависимость времени достижения заданной точки траектории от спектра воздействия; особенно сильно эта зависимость выражена для степенного спектра.

Авторы выражают благодарность проф. В.И. Гурману, проф. В.Ф. Кротову за интересные обсуждения.

Литература

1. Валиев К.А. Квантовые компьютеры и квантовые вычисления // Успехи физических наук. – 2005. – Вып. 175. – С. 3-39.
2. Shor P.W. Algorithms for Computation: Discreta Logarithms and Factoring // Proc. of the 35th Annual Symp. on the Foundation of Computer Science, Los Alamitos, CA, USA, 1994. P. 124-134.
3. Feynman R.P. Simulating physics with computers // Int. J. Theor. Phys. 21. 1982. P. 467.

4. Grover L.K. A fast quantum mechanical algorithm for database search // Proc. of the 28th Annual ACM Symp. on Theory of Computation (Philadelphia, Pennsylvania), ACM Press, New York, 1996. Pp. 212-218.
5. Гурман В.И. Принцип расширения в задачах управления. – М.: Наука, Физматлит, 1997. – 288 с.
6. Архинчеев В.Е. Введение в квантовые вычисления и управление спиновыми состояниями // Химическая физика и мезоскопия. – 2013. – Т. 15. – №1. – С. 101-104.
7. Гурман В.И. Преобразования управляемых систем для исследования импульсных режимов. – Автоматика и Телемеханика. – 2009. – № 4. – С. 89-97.

Архинчеев Валерий Ефимович, доктор физико-математических наук, проректор Бурятского государственного университета по научно-исследовательской работе, тел. (301-2) 656789, e-mail: varkhin@mail.ru

Андреева Екатерина Юрьевна, научный сотрудник Центра космических услуг Бурятского государственного университета, тел. (301-2) 248150, e-mail: ekaterinamar@gmail.com

Arkhincheev Valery Efimovich, doctor of physical and mathematical sciences, Vice-rector of the Buryat State University for research work.

Andreeva Ekaterina Yurievna, research associate of the Space services center of the Buryat State University.

УДК 534.22

© Т.Г. Дармаев, Ж.Г. Дамбаев

МАТЕМАТИЧЕСКАЯ МОДЕЛЬ ПОТОКА НАД КОЛЕБЛЮЩЕЙСЯ ПЛАСТИНКОЙ¹

В данной работе на основе метода последовательных аппроксимаций уравнений Навье-Стокса для завихренности получены уравнения и граничные условия для колеблющихся пластин с учетом амплитуды до второго порядка. С помощью системы Mathematica получены аналитические решения и построены графики векторных полей скоростей для пластины бесконечной длины.

Ключевые слова: уравнения Навье-Стокса, метод последовательных аппроксимаций, колеблющаяся пластина, поле скоростей.

© T.G. Darmaev, Zh.G. Dambaev

MATHEMATICAL MODEL OF THE STREAM OVER THE FLUCTUATING PLATE

In this work on the basis of the method of successive approximations of Navier-Stokes equations for vorticity the equations and boundary conditions for the fluctuating plates taking into account amplitude to the second order are received. By means of Mathematica system analytical solutions are received and schedules of vector fields of speeds for a plate of infinite length are constructed.

Keywords: Navier-Stokes equations, method of successive approximations, fluctuating plate, fields of speeds.

Введение

В 1787 г. немецкий физик Хладни показал, что при колебании пластины смычком частицы песка образуют самоорганизующие симметричные структуры. Модели минивентиляторов в виде вибрирующих пластин бесконечной и конечной длин для микроэлектронных устройств получены в работе [1]. В работе [2] идея Хладни применяется в наномеханике для разделения наночастиц по размерам акустическими волнами разных частот.

В данной работе на основе метода последовательных аппроксимаций [3] уравнений Навье-Стокса для завихренности получены уравнения и граничные условия для колеблющихся пластин с учетом амплитуды до второго порядка. С помощью системы Mathematica получены аналитические решения и построены графики векторных полей скоростей для пластины бесконечной длины.

1. Метод последовательных аппроксимаций

Рассмотрим пластину колеблющейся в продольном направлении по закону $w(x,t)$ с амплитудой $A \ll 1$ и гармонической частотой σ .

В двумерном течении вязкой несжимаемой жидкости вектор завихренности имеет лишь одну ненулевую компоненту:

$$\omega_z = \omega = \frac{\partial U}{\partial x} - \frac{\partial V}{\partial y} = -\frac{\partial^2 \psi}{\partial x^2} - \frac{\partial^2 \psi}{\partial y^2} \equiv -\nabla^2 \psi,$$

где функция тока $\psi(x,y)$ определяется следующим образом:

$$U = \frac{\partial \psi}{\partial y}, \quad V = -\frac{\partial \psi}{\partial x}.$$

Далее из уравнений Навье-Стокса получаем уравнение для завихренности:

$$\frac{\partial \omega}{\partial t} + U \frac{\partial \omega}{\partial x} + V \frac{\partial \omega}{\partial y} = \nu \nabla^2 \omega \quad (1)$$

где ν – кинематическая вязкость.

Используя метод последовательных аппроксимаций [3] разлагаем функцию тока:

$$\psi(x, y, t) = \psi_1(x, y, t) + \psi_2(x, y, t) + \psi_3(x, y, t) + \dots, \quad (2)$$

где $\psi_i = O(A^i)$, $i = 1, 2, \dots$

¹ Работа выполнена в рамках государственного задания №2014/312 МОиН РФ

Подставляя (2) в (1) и приравнивая члены одинакового порядка по амплитуде до 2-го порядка, получаем:

$$\begin{cases} \frac{\partial \nabla^2 \psi_1}{\partial t} = \nu \nabla^4 \psi_1, \\ F(x, y, t) + \frac{\partial \nabla^2 \psi_2}{\partial t} = \nu \nabla^4 \psi_2, \end{cases} \quad (3)$$

где $F(x, y, t) \equiv \frac{\partial \psi_1}{\partial y} \frac{\partial \nabla^2 \psi_1}{\partial x} - \frac{\partial \psi_1}{\partial x} \frac{\partial \nabla^2 \psi_1}{\partial y}$.

Из условий прилипания на пластине получаем нелинейные граничные условия при $y = w(x, t)$ (подвижная пластина):

$$U = \frac{\partial \psi}{\partial y} = 0, \quad V = -\frac{\partial \psi}{\partial x} = \frac{\partial w}{\partial t} \quad (4)$$

Учитывая малость амплитуды разлагаем (4) в ряд Тейлора при $y=0$ (неподвижная пластина):

$$\frac{\partial \psi}{\partial y}(x, w, t) = \frac{\partial \psi}{\partial y}(x, 0, t) + w \frac{\partial^2 \psi}{\partial y^2}(x, 0, t) + \frac{w^2}{2!} \frac{\partial^3 \psi}{\partial y^3}(x, 0, t) + O(A^4)$$

$$\frac{\partial \psi}{\partial x}(x, w, t) = \frac{\partial \psi}{\partial x}(x, 0, t) + w \frac{\partial^2 \psi}{\partial x \partial y}(x, 0, t) + \frac{w^2}{2!} \frac{\partial^3 \psi}{\partial x \partial y^2}(x, 0, t) + O(A^4)$$

Подставляя далее в (4) и приравнивая одинаковые члены по амплитуде получаем:

$$\frac{\partial \psi_1}{\partial y}(x, 0, t) = 0, \quad \frac{\partial \psi_1}{\partial x}(x, 0, t) = -\frac{\partial w}{\partial t}, \quad (5)$$

$$\frac{\partial \psi_2}{\partial y}(x, 0, t) = -w(x, t) \frac{\partial^2 \psi_1}{\partial y^2}(x, 0, t), \quad \frac{\partial \psi_2}{\partial x}(x, 0, t) = -w(x, t) \frac{\partial^2 \psi_1}{\partial x \partial y}(x, 0, t).$$

Из условий затухания колебаний жидкости на бесконечности дополнительно получаем, что ψ_1, ψ_2 и их градиенты стремятся к нулю при $y \rightarrow \infty, \quad x \rightarrow \pm\infty$:

$$\begin{aligned} \psi_1(x, \infty, t) = \frac{\partial \psi_1}{\partial x}(x, \infty, t) = \psi_2(x, \infty, t) = \frac{\partial \psi_2}{\partial x}(x, \infty, t) = 0, \\ \psi_1(\pm\infty, y, t) = \frac{\partial \psi_1}{\partial x}(\pm\infty, y, t) = \psi_2(\pm\infty, y, t) = \frac{\partial \psi_2}{\partial x}(\pm\infty, y, t) = 0. \end{aligned} \quad (6)$$

2. Расчеты

Из полученной выше рекуррентной системы краевых задач (3), (5), (6) были вычислены аналитически в системе Mathematica методом разделения переменных функции тока (2) стационарных течений до второго порядка разложения по амплитуде.

На рис.1-2 приведены векторные поля скоростей функций тока 1-го и 2-го порядка разложения по амплитуде соответственно.

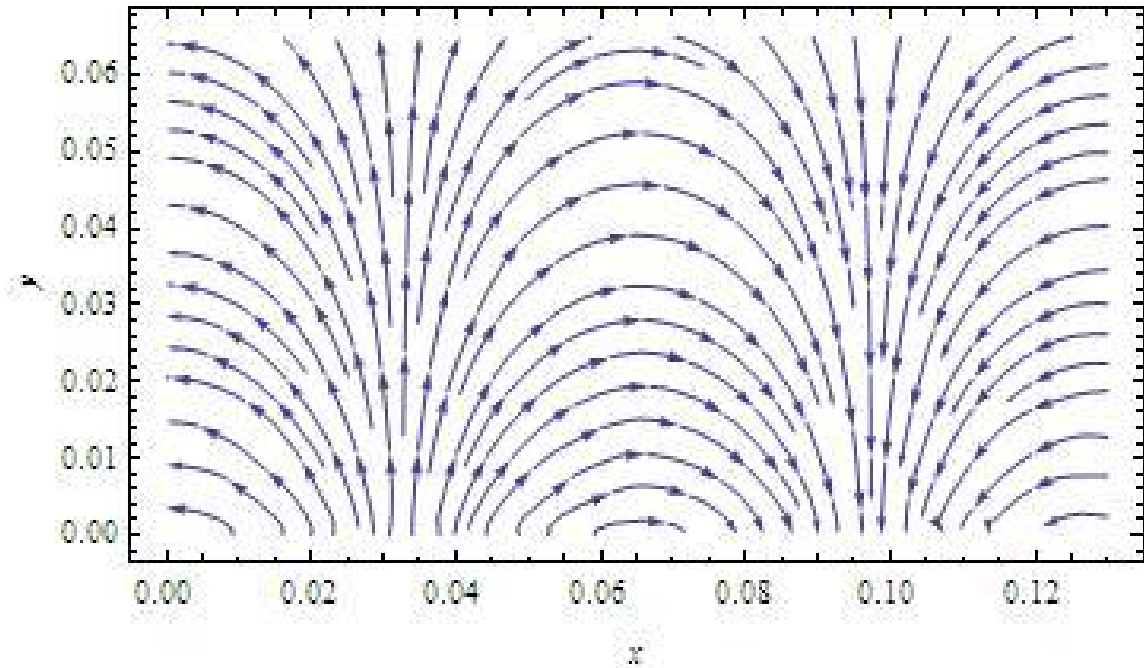


Рис. 1. Векторное поле функции тока $\psi_1(x, y)$

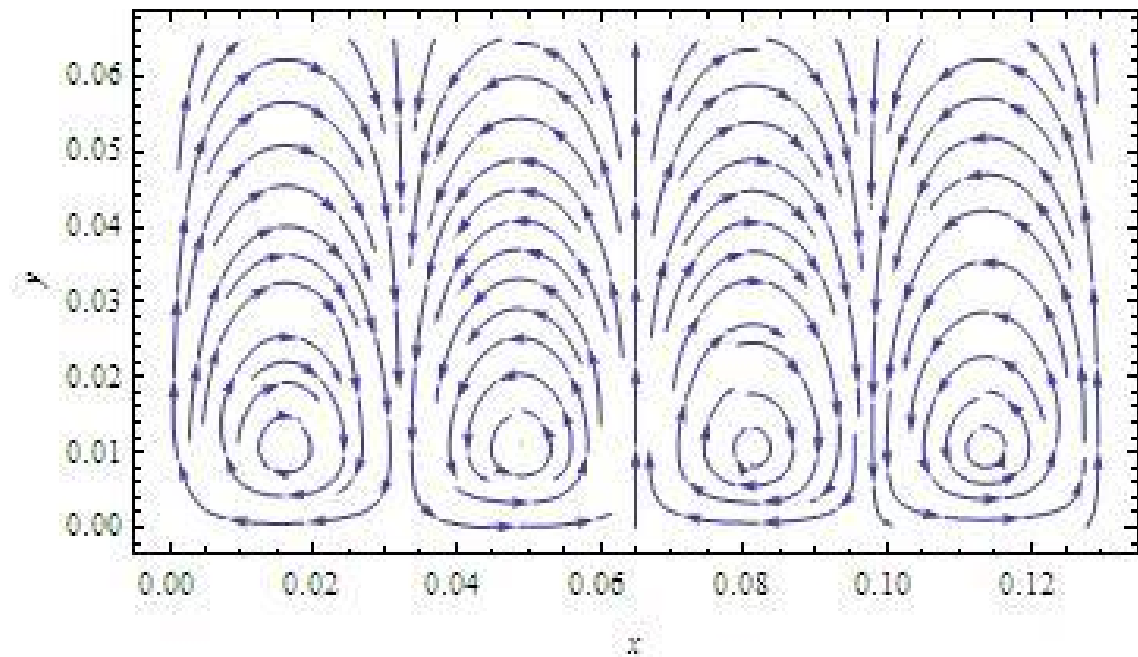


Рис. 2. Векторное поле функции тока $\psi_2(x, y)$

В расчетах использовались следующие данные:

$$w(x, t) = A \cos\left(\frac{\pi}{L}x\right) e^{j\sigma t}, \quad j = \sqrt{-1},$$

$$\nu = 1.45 \times 10^{-5} \text{ m}^2 / \text{s}, \quad L = 6.5 \text{ cm}, \quad A = 0.08 \text{ cm}, \quad \sigma = 120\pi \text{ rad} / \text{s}.$$

Заключение

Из рис.1-2 видно, что учет разложения по амплитуде до второго порядка показывает, что течение имеет вихревую структуру с периодом равным $L/2$. Соседние вихри вращаются во взаимно противоположных направлениях с центрами вблизи пластины. Отток от пластины происходит в окрестностях точек покоя, а подтекание к пластине в окрестностях пучностей стоячей волны.

Литература

1. Acikalin T. Two-dimensional streaming flows induced by resonating thin beams / T. Acikalin, A.Ramah, S.V.Garimella // J.Acoust.Soc. Am. – 2003. – V.114. – N.4 - pt.1 – pp.1785-1795.
2. Dorrestijn M. Chladni Figures Revisited based on nanomechanics / Dorrestijn M. et al // Physical Review Letters. – 2007. – V.98. – 026102.
3. Slichting H. Boundary layer theory. – New-York: McGraw-Hill, 1955.

Дармаев Тумэн Гомбоцыренович, кандидат физико-математических наук, доцент кафедры прикладной математики Бурятского государственного университета, тел. (301-2) 221215, E-mail: dtg@bsu.ru

Дамбаев Жаргал Гомбоевич, доктор технических наук, профессор, заведующий лабораторией оптимального управления Научно-образовательного и инновационного центра системных исследований и автоматизации БГУ, 670000, г.Улан-Удэ, ул.Смолина 24а, тел.(3012)221215, E-mail: g.dambaev@rambler.ru

Darmaev Tumen Gombotsyrenovich, candidate of physical and mathematical sciences, docent of department applied mathematics of Buryat State University.

Dambaev Zhargal Gomboevitch, doctor of technical sciences, professor, the head of laboratory of optimal control of the Scientific and Educational Innovation Centre For System Research and Automation at Buryat State University.

УДК 537.876.4:537.622.6

© Г.Б. Итигилов, Д.Ш.Ширапов

ЧИСЛЕННЫЙ РАСЧЕТ ДИСПЕРСИИ ГИБРИДНЫХ ВОЛН В ГИРОТРОПНЫХ ОГРАНИЧЕННЫХ ОБЛАСТЯХ ЭЛЛИПТИЧЕСКОЙ ФОРМЫ ПРИ ПРОДОЛЬНОМ НАМАГНИЧИВАНИИ

Проведен численный анализ с построением соответствующих графиков впервые полученных дисперсионных уравнений гибридных волн в гиротропных ограниченных областях эллиптической формы при продольном намагничивании.

Ключевые слова: электромагнитная волна, ограниченная гиротропная область, уравнения Максвелла, продольное намагничивание, дисперсионное уравнение.

© G.B.Itigilov, D.Sh.Shirapov

NUMERICAL SOLUTION OF DISPERSION OF HYBRID WAVES IN GYROTROPIC LIMITED AREAS OF THE ELLIPTIC FORM IN CASE OF LONGITUDINAL MAGNETIZATION

The numerical analysis with creation of the appropriate diagrams for the first time of the received dispersing equations of hybrid waves in gyrotropic limited areas of the elliptic form is carried out in case of longitudinal magnetization.

Keywords: electromagnetic wave, limited gyrotropic area, Maxwell's equations, longitudinal magnetization, dispersing equation.

Введение

Известно, что направление поворота поляризации электромагнитной волны (ЭМВ) в продольно-намагниченной гиротропной ограниченной области не зависит от направления распространения волны и изменяется на обратное при изменении направления постоянного намагничивания. Это свойство, являющееся проявлением невзаимности используется в сверхвысокочастотных ферритовых устройствах поляризационного или фарадеевского типа [1, 2].

В настоящее время распространение ЭМВ в продольно-намагниченных ограниченных прямоугольных и круглых областях достаточно хорошо изучены [1-3]. Но распространение ЭМВ в гиротропных эллиптических областях мало исследованы и носят фрагментарный характер [4].

Численный расчет дисперсии

Уравнения Максвелла для гармонических процессов без наведенных токов и зарядов имеют вид [1]:

$$\begin{cases} \operatorname{rot} \bar{H} = jw\varepsilon \bar{E}; & \operatorname{rot} \bar{E} = -jw\bar{B}; \\ \operatorname{div} \bar{E} = 0; & \operatorname{div} \bar{B} = 0, \end{cases} \quad (1)$$

где \bar{E}, \bar{H} - соответственно напряженности электрического и магнитного полей; ε - абсолютная диэлектрическая проницаемость среды, $\varepsilon \bar{E} = \bar{D}$ - электрическая индукция, \bar{B} - магнитная индукция, j - мнимая единица, w - циклическая частота.

При распространении волны в магнитогиротропной среде магнитная индукция \bar{B} в системе (1) примет следующий вид:

$$\bar{B} = \|\mu\| \bar{H}. \quad (2)$$

При продольном намагничивании, когда направление внешнего намагничивающего постоянного магнитного поля совпадает с направлением распространения ЭМВ (волна распространяется вдоль координаты Z), тензор магнитной проницаемости феррита, как следует из [1], имеет вид:

$$\|\mu\| = \begin{bmatrix} \mu & j\kappa & 0 \\ -j\kappa & \mu & 0 \\ 0 & 0 & \mu_{\parallel} \end{bmatrix}, \quad (3)$$

где $\mu = \mu_0 + \mu_0 \frac{w_0 w_M}{w_0^2 - w^2}$, $k = \mu_0 \frac{w w_M}{w_0^2 - w^2}$, $w_M = \mu_0 Y M_0$, $Y = 1.76 \cdot 10^{11} \frac{Kл}{кз}$ - гиромангнитное от-
ношения для спина электрона, $w_0 = \mu_0 Y H_0$ - частота ферромагнитного резонанса, μ_0 - магнитная
постоянная, M_0 - намагниченность феррита, H_0 - намагничивающее внешнее магнитное поле.

Знаки перед недиагональными компонентами в (3) могут быть противоположными, если взять
 $k = -\mu_0 \frac{w w_M}{w_0^2 - w^2}$.

Ранее в работе [5], используя разработанный метод инвариантных преобразований для получе-
ния обобщенных уравнений Гельмгольца при продольном намагничивании, применяя к получен-
ным уравнениям методы укорочения исходного дифференциального уравнения [6,7] и разделения
переменных [8], были поставлены и впервые решены краевые задачи с получением дисперсион-
ного уравнения для ограниченной гиротропной эллиптической области при продольном намагни-
чивании, которое имеет вид:

$$\left[-\left(k_{\perp}^2 - \gamma_z^2 - \frac{4q_1}{e^2} \right) \frac{4q_2}{e^2} \frac{Ce'_m(\xi_0, q_1)}{Ce_m(\xi_0, q_1)} + \left(k_{\perp}^2 - \gamma_z^2 - \frac{4q_2}{e^2} \right) \frac{4q_1}{e^2} \frac{Ce'_m(\xi_0, q_2)}{Ce_m(\xi_0, q_2)} \right] +$$

$$+ j \frac{\gamma_z^2 \varepsilon k a^2 \Lambda_1 \Lambda_2}{\mu^2} \left\{ \begin{array}{l} \frac{ce'_m(\varphi, q_2)}{ce_m(\varphi, q_2)} \\ - \frac{ce'_m(\varphi, q_1)}{ce_m(\varphi, q_1)} \end{array} \right\} + j \frac{\gamma_z^3 w \varepsilon k}{\mu^2} \left\{ \begin{array}{l} \frac{ce'_m(\varphi, q_1)}{ce_m(\varphi, q_1)} \Lambda_1 - \\ - \frac{ce'_m(\varphi, q_2)}{ce_m(\varphi, q_2)} \Lambda_2 \end{array} \right\} = 0, \tag{4}$$

где $k_{\perp}^2 = w^2 \varepsilon \mu_{\perp}$, γ_z - постоянная распространения, $Ce_m(\xi, q_{1,2})$ - четные модифицированные
функции Матье I-го рода (с целым индексом) и их производные $Ce'_m(\xi, q_{1,2})$, ξ_0 - граничный эл-

липс, $q_1 = \frac{e^2 \left(k_{\perp} - \gamma^2 - \Lambda_1 \gamma w \varepsilon \frac{k}{\mu} \right)}{4}$ и $q_2 = \frac{e^2 \left(k_{\perp} - \gamma^2 - \Lambda_2 \gamma w \varepsilon \frac{k}{\mu} \right)}{4}$ - параметры функций Матье,
 $ce_m(\varphi, q_{1,2})$ - четные обыкновенные функции Матье I-го рода целого порядка m и их производные
 $ce'_m(\varphi, q_{1,2})$, $\Lambda_{1,2}$ - корни уравнения $\gamma w \varepsilon \frac{k}{\mu} \Lambda^2 + \left(w^2 \varepsilon \mu_{\parallel} - \frac{\mu_{\parallel}}{\mu} \gamma^2 - w^2 \varepsilon \mu_{\perp} + \gamma^2 \right) \Lambda - \gamma w \mu_{\parallel} \frac{k}{\mu} = 0$.

Известно, что в ограниченных эллиптических областях распространяются четные и нечетные
волны [8]. Выражение (4) описывает распространение четных волн. Для получения дисперсионно-
го уравнения для нечетных волн в (4) надо сделать следующую замену:

$$\left\{ \begin{array}{l} Ce(\xi_0, q_{1,2}) \rightarrow Se(\xi_0, q_{1,2}), Ce'(\xi_0, q_{1,2}) \rightarrow Se'(\xi_0, q_{1,2}); \\ ce(\varphi, q_{1,2}) \rightarrow se(\varphi, q_{1,2}), ce'(\varphi, q_{1,2}) \rightarrow se'(\varphi, q_{1,2}), \end{array} \right. \tag{5}$$

где $Se(\xi_0, q_{1,2}), Se'(\xi_0, q_{1,2})$ - нечетные присоединенные (модифицированные) функции Матье I-го
рода (с целым индексом) и их производные, $se(\varphi, q_{1,2}), se'(\varphi, q_{1,2})$ - нечетные обыкновенные
функции Матье I-го рода целого порядка m и их производные.

Дисперсионное уравнение (4) не решается аналитически. Поэтому для численного решения (4)
был разработан комплекс модифицированных программ для стандартного пакета Maple.

На рисунках 1, 2 представлены графики зависимостей постоянных распространения $\frac{\gamma_z}{k_z}$ от
напряженности намагничивающего магнитного поля $\frac{w_0}{w}$ для гиротропной эллиптической облас-
ти при продольном намагничивании с длиной большой полуоси $s=0,016$ м. и эксцентриситетам
 $E=0.75$ при частоте $w = 6.28 \cdot 10^{10}$ Гц и намагниченности феррита $w_M = 0.15 \cdot w$.

На рисунке 1 и на всех последующих рисунках нижние индексы «С» и «S» означают четную
и нечетную моды, верхние индексы «+» и «-» означают правое и левое направления вращения,
каждая цифра нижних индексов «11» и «12» определяет число полуволн, укладываемых вдоль

поперечных координатных осей эллиптической ограниченной области: первая цифра означает периодичность поля по координате φ , а вторая - по ξ . При $\frac{w_0}{w} = 1$ наступает ферромагнитный резонанс (на всех рисунках вертикальная пунктирная линия).

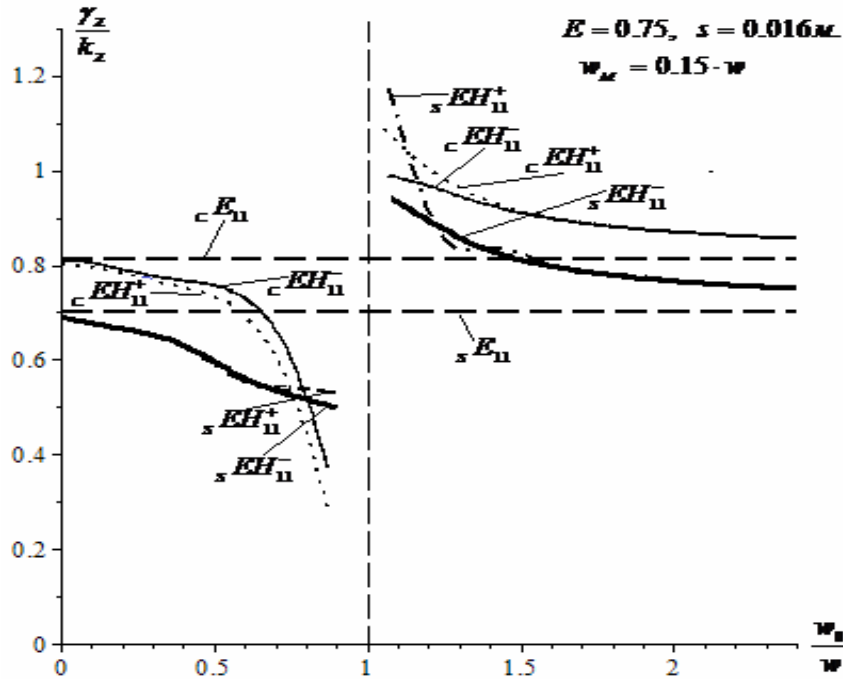


Рис. 1. Зависимости постоянных распространения от напряженности намагничивающего поля для эксцентриситета $E=0.75$, полученные при решении уравнений (4) (показаны только EH-моды).

На рисунке 1 горизонтальными пунктирными линиями показаны моды при отсутствии магнитного поля: $c E_{11}$, $s E_{11}$.

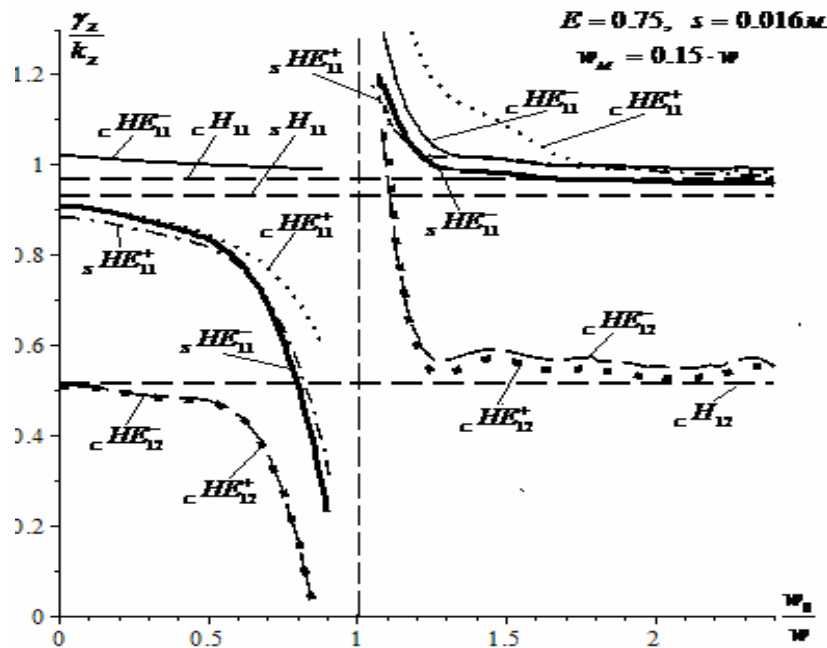


Рис. 2. Зависимости постоянных распространения от напряженности намагничивающего поля для эксцентриситета $E=0.75$, полученные при решении уравнений (4) (показаны только HE-моды).

На рисунке 2 горизонтальными пунктирными линиями показаны моды при отсутствии магнитного поля: $c H_{11}$, $s H_{11}$, $c H_{12}$.

Заключение

Из данных графиков следует, что порядок появления основной и высших мод зависит от эксцентриситета. Также анализ дисперсионных характеристик и их графиков показывает, что при продольном намагничивании в ограниченной гиротропной эллиптической области каждая ЭМВ может быть рассмотрена как две волны, распространяющиеся с разными фазовыми скоростями. Эта особенность может быть применена для разработки ферритовых устройств сверхвысоких частот на основе эллиптических направляющих систем с гиротропным заполнением

Литература

1. Микаэлян А.Л. Теория и применение ферритов на сверхвысоких частотах. – Л.:Госэнергоиздат, 1963. – 664 с.
2. Гуревич А.Г., Мелков Г.А. Магнитные колебания и волны. – М.: Физматлит, 1994. – 464 с.
3. Лакс Б., Баттон К. Сверхвысокочастотные ферриты и ферритмагнетики. Пер. с англ. М.: Мир, 1965. – 676 с.
4. Гончаров А.М., Карпенко В.А. Основы теории оптических волноводов. Изд. 2-е, испр. – М.:Едиториал УРСС, 2004. – 240 с.
5. Итигилов Г.Б., Ширапов Д.Ш. Дисперсионные характеристики гибридных волн в ограниченных эллиптических гиротропных областях при продольном намагничивании // Вестник Бурятского государственного университета. Математика, информатика.– 2013. – №2. – С. 70-74.
6. Назаров А.В. Электромагнитные волны в структурах, содержащих продольно намагниченные ферритовые слои / А.В. Назаров, С.Б. Раевский // Физика волновых процессов и радиотехнические системы. – 2007. – Т.10. – № 1. – С. 76-82.
7. Сул Г., Уокер Л. Вопросы волноводного распространения электромагнитных волн в гиротропных средах. – Пер. с англ. / под ред. Мироманова Г. М.: Изд. иностр. литературы, 1955. – 192 с.
8. Мак-Лахлан Н.В. Теория и приложение функций Матье. Пер. с англ. В.А. Братановского. М.: Изд. иностранной литературы, 1953. – 475 с.

Итигилов Гарма Борисович, старший преподаватель кафедры “Электронные вычислительные системы” Восточно-Сибирского государственного университета технологий и управления, тел. 8-3012-21-53-14, E-mail: Gablz@mail.ru

Ширапов Дашиадондок Шагдарович, д.ф.-м.н., профессор кафедры прикладной математики Бурятского государственного университета, заведующий кафедрой «Электронные вычислительные системы» Восточно-Сибирского государственного университета технологий и управления, тел. 8-3012-21-53-14, E-mail: Shir@esstu.ru

Itigilov Garma Borisovich, senior teacher of chair “Electronic computer systems” of the East Siberian State University of Technology and Management, tel. 8-3012-21-53-14, E-mail: Gablz@mail.ru

Shirapov Dashadondok Shagdarovich, the doctor of physical and mathematical sciences, the professor of applied mathematics department of Buryat State University, head of the chair “Electronic computer systems” of the East Siberian State University of Technology and Management, tel. 8-3012-21-53-14, E-mail: Shir@esstu.ru

Содержание

1. Управляемые системы и методы оптимизации

Булдаев А.С. Оптимизация управляющих функций и параметров в нелинейных системах на основе задач о неподвижной точке	3
Дрыганова Е.В. Метод оптимизации линейно управляемых дискретных систем.....	11
Трунин Д.О., Очирбат Б., Ганхуяг Д. Проекционная процедура нелокального улучшения управляемых систем с функциональными ограничениями.....	15

2. Информационные системы и технологии

Бурлаков А.С., Хмельнов А.Е. Язык спецификации архитектуры ЭВМ и периферийных устройств	20
Курганская Г.С. Самоорганизация Интернет-образования в облачных технологиях.....	27
Михайлов А.А. Промежуточное представление подпрограмм в задаче декомпиляции объектных файлов dsuil	32
Попков Г.В. О проблеме живучести телекоммуникационных сетей.....	39
Федорченко Л.Н. Эквивалентность как отношение подобия в трансляции языков.....	49
Харинов М.В. Улучшение качества приближения изображения на основе иерархической сегментации.....	54

3. Функциональный анализ и дифференциальные уравнения

Кибирев В.В. Формула Грина в теории потенциала	58
--	----

4. Математическое моделирование

Архинчеев В.Е., Андреева Е.Ю. Введение в квантовые вычисления и управление движением частицы в магнитном поле	63
Дармаев Т.Г., Дамбаев Ж.Г. Математическая модель потока над колеблющейся пластинкой	69
Итигилов Г.Б., Ширапов Д.Ш. Численный расчет дисперсии гибридных волн в гиротропных ограниченных областях эллиптической формы при продольном намагничивании.....	73

Contents

1. Controlled systems and methods of optimization

Buldaev A.S. Optimization of the managing functions and parameters in nonlinear systems on basis of the fixed point problems.....	3
Dryganova E.V. The method of linear controlled discrete system optimization	11
Trunin D.O., Ochirbat B., Gankhuyag D. Projecting procedure of nonlocal improving controlled systems with functional constraints	15

2. Information systems and technologies

Burlakov A.S., Hmelnov A.E. The language of specification of computer architecture and peripheral devices	20
Kurganskaya G.S. Self-organization of Internet learning in cloud technologies	27
Mihailov A.A. The intermediate representation of subroutines in the task of decompilation of the object files dcuil	32
Popkov G.V. About survivability problem of telecommunication networks.....	39
Fedorchenko L.N. Equivalence as the similarity relation in language translation	49
Kharinov M.V. Improving the quality of digital image approximation based on hierarchical segmentation	54

3. Functional analysis and differential equations

Kibirev V.V. Green's formula in the theory of potential	58
---	----

4. Mathematical modeling

Arkhincheev V.E., Andreeva E.Y. Introduction into quantum computing and partial motion control in a magnetic field	61
Darmaev T.G., Dambaev Zh.G. Mathematical model of the stream over the fluctuating plate	69
Itigilov G.B., Shirapov D.Sh. Numerical solution of dispersion of hybrid waves in gyrogropic limited areas of the elliptic form in case of longitudinal magnetization	73

ВЕСТНИК БУРЯТСКОГО ГОСУНИВЕРСИТЕТА

Вестник БГУ включен в подписной каталог Роспечати за № 18534 и Перечень изданий Российской Федерации, где должны быть опубликованы основные научные результаты диссертаций на соискание ученых степеней доктора и кандидата наук.

На основании постановления заседания Ученого совета БГУ за № 10 от 28 мая 2009 г. в «Вестнике БГУ» в 2014 г. публикуются статьи по следующим направлениям:

1. Педагогика (январь)

гл. ред. Дагбаева Нина Жамсуевна – тел. 21-04-11; 44-23-95

эл. адрес: vestnik_pedagog@bsu.ru

2. Экономика. Право (февраль)

гл. ред. Атанов Николай Иванович – тел. 21-37-44

эл. адрес: vestnik_econom@bsu.ru

3. Химия, физика (март)

гл. ред. Хахинов Вячеслав Викторович – тел. 43-42-58

эл. адрес: khakhinov@mail.ru

4. Биология, география (март)

гл. ред. Доржиев Цыдып Заятуевич – тел. 21-03-48

эл. адрес: vestnik_biolog@bsu.ru

5. Психология, социальная работа (апрель)

гл. ред. Базарова Татьяна Содномовна – тел. 21-26-49

эл. адрес: decspf@mail.ru

6. Философия, социология, политология, культурология (апрель)

гл. ред. Осинский Иван Иосифович – тел. 21-05-62

эл. адрес: intellige2007@rambler.ru

7. История (май)

гл. ред. Митупов Константин Батомункич – тел. 21-64-47

эл. адрес: vestnik_history@bsu.ru

8. Востоковедение (май)

гл. ред. Бураев Дмитрий Игнатьевич – тел. 44-25-22

эл. адрес: gailia@mail.ru

9. Математика, информатика (июнь)

гл. ред. Булдаев Александр Сергеевич – тел. 21-97-57

эл. адрес: vestnik_bsu_math@rambler.ru

10. Филология (сентябрь)

гл. ред. Имixelова Светлана Степановна – тел. 21-05-91

эл. адрес: 223015@mail.ru; map1955@mail.ru

11. Романо-германская филология (сентябрь)

гл. ред. Ковалева Лариса Петровна – тел. 21-17-98

эл. адрес: klp@bsu.ru, khida@mail.ru

12. Медицина, фармация (октябрь)

гл. ред. Хитрихеев Владимир Евгеньевич – тел. 44-82-55

эл. адрес: vestnik_medicine@bsu.ru

13. Физкультура и спорт (октябрь)

гл. ред. Гаськов Алексей Владимирович – тел. 21-69-89

эл. адрес: gaskov@bsu.ru

14. Философия, социология, политология, культурология (ноябрь)

гл. ред. Осинский Иван Иосифович – тел. 21-05-62

эл. адрес: intellige2007@rambler.ru

15. Теория и методика обучения (декабрь)

гл. ред. Очиров Михаил Надмитович – тел. 21-97-57

эл. адрес: vestnik_method@bsu.ru

Требования к оформлению статей, представляемых в «Вестник БГУ»

Отбор и редактирование публикуемых статей производятся редакционной коллегией из ведущих ученых и приглашенных специалистов.

В «Вестник БГУ» следует направлять статьи, отличающиеся высокой степенью научной новизны и значимостью. Каждая статья имеет УДК, а также письменный развернутый отзыв (рецензию) научного руководителя или научного консультанта, заверенный печатью.

Автор статьи обязан заключить лицензионный договор о предоставлении неисключительных прав на использование созданного им произведения (статьи) ФГБОУ ВПО «Бурятский государственный университет». Образец лицензионного договора представлен на сайте БГУ.

Общие требования	Тексты представляются в электронном и печатном виде. Файл со статьей может быть на дискете или отправлен электронным письмом. На последней странице – подпись автора(ов) статьи. Название статьи и аннотация даются и на английском языке. После аннотации дать ключевые слова на русском и английском языках.
Электронная копия	Текстовый редактор Microsoft Word (версии 6.0, 7.0, 97). В имени файла указывается фамилия автора.
Параметры страницы	Формат А4. Поля: правое – 15 мм, левое – 25 мм, верхнее, нижнее – 20 мм.
Форматирование основного текста	С нумерацией страниц. Абзацный отступ – 5 мм. Интервал – полуторный.
Гарнитура шрифта	Times New Roman. Обычный размер кегля – 14 пт. Список литературы и аннотация – 12 пт.
Объем статьи (ориентировочно)	Кратких сообщений – до 3 с., статей на соискание ученой степени кандидата наук – 7–12 с., на соискание ученой степени доктора наук – 8–16 с.
Сведения об авторах	Указываются фамилия, имя, отчество (полностью), ученая степень, звание, должность и место работы, адрес с почтовым индексом, телефоны/факсы, e-mail (на русском и английском языках)

- Список литературы – все работы необходимо пронумеровать, в тексте ссылки на литературу оформлять в квадратных скобках.

- Материалы, не соответствующие предъявленным требованиям, к рассмотрению не принимаются. Все статьи проходят проверку в системе «Антиплагиат. ВУЗ».

- Решение о публикации статьи принимается редакцией «Вестника БГУ». Корректурa авторам не высылаeтся, присланные материалы не возвращаются.

- Статьи принимаются в течение учебного года.

- Допустима публикация статей на английском языке, сведения об авторах, название и аннотацию которых необходимо перевести на русский язык.

- Формат журнала 60x84 1/8.

- Статья должна содержать минимум таблиц, формул, рисунков и графиков. Их присутствие допускается только в тех случаях, если описать процесс в текстовой форме невозможно или нецелесообразно. Желательно использование только вертикальных таблиц и рисунков. Все объекты должны быть черно-белыми без оттенков. Все формулы должны быть созданы с использованием компонента Microsoft Equation или в виде четких картинок. Символы можно вставлять с помощью операции в Word (Вставка – Символ). Диаграммы располагаются в тексте с использованием программы Microsoft Excel (Вставка – Объект – Создание – Диаграмма Microsoft Excel). Рисунки и графики должны иметь четкое изображение и быть выдержаны в черно-белой гамме, лучше применять штриховку (Формат автофигуры – Цвета и линии – Цвет – Способы заливки – Узор). Схемы создаются с помощью панели инструментов Рисование. Фотографии и рисунки в формате *.tif или *.jpg должны иметь разрешение не менее 300 dpi. Диаграммы, формулы, рисунки, графики должны прилагаться отдельными файлами, чтобы издательство имело возможность ввести в них правки.

Стоимость обработки 1 с. (формата А4) для преподавателей БГУ составляет 200 р., для остальных – 400 р. Для аспирантов – бесплатно.

Адрес: 670000, г. Улан-Удэ, ул. Смолина, 24 а, Издательство БГУ.

Факс (301-2)-21-05-88

Оплата производится при получении счета от бухгалтерии БГУ.