

УДК 519.68

## МНОГОМЕТОДНАЯ ОПТИМИЗАЦИЯ УПРАВЛЕНИЯ В СЛОЖНЫХ ПРИКЛАДНЫХ ЗАДАЧАХ

© 2019 г. А. И. Тятюшкин

(664033 Иркутск, ул. Лермонтова, 134, ФГБУН Ин-т динамики систем и теории управления СО РАН, Россия)

e-mail: [tjat@icc.ru](mailto:tjat@icc.ru)

Поступила в редакцию 05.05.2018 г.  
Переработанный вариант 16.05.2018 г.

Для получения численного решения краевой задачи с достаточно высокой точностью конструируется алгоритм, состоящий из итераций градиентного метода и итераций метода квазилинеаризации. Для нахождения “идеального” решения многокритериальной задачи оптимального управления предлагаются прямой и двойственный алгоритмы, которые обеспечивают эффективный поиск как коэффициентов свертки, так и оптимального управления. Эффективность предложенных многометодных алгоритмов показана на решении прикладных задач. Библ. 10. Фиг. 1.

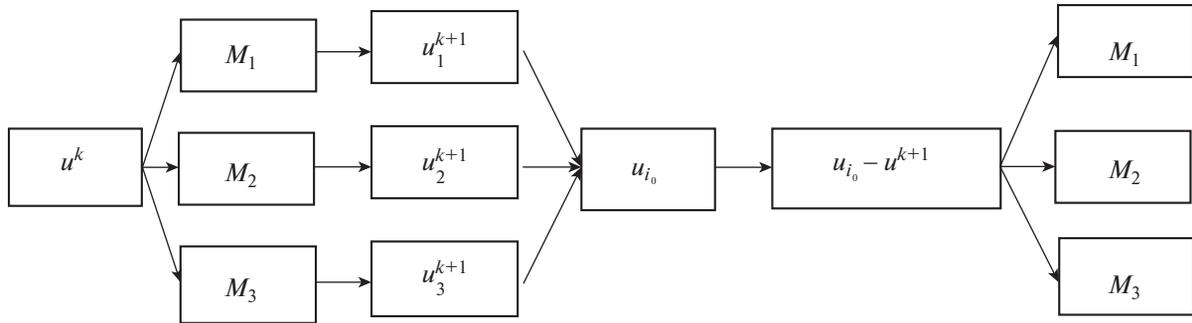
**Ключевые слова:** многометодная оптимизация, оптимальное управление, краевая задача, многокритериальная задача, градиентный метод, принцип максимума, метод квазилинеаризации.

**DOI:** 10.1134/S0044466919020145

### ВВЕДЕНИЕ

В процессе решения сложной задачи оптимального управления обычно последовательно реализуются различные алгоритмы, одни из которых имеют достаточно широкую область сходимости и, следовательно, допускают задание грубого начального приближения, а другие начинают с хорошего приближения и позволяют получать решение задачи с требуемой точностью. Так как большинство численных методов решения нелинейных задач сходится к управлению, удовлетворяющим только необходимым условиям оптимальности, то такая многометодная технология оптимизации (см. [1]) дает возможность избежать попадания в локальный экстремум и повышает надежность расчета оптимального управления для нелинейной системы.

Многометодная технология решения задач оптимального управления заключается в том, что параллельно используется сразу несколько итерационных методов оптимизации для поиска решения одной и той же задачи. Основной проблемой применения многометодной технологии при численном решении задач оптимального управления является выбор метода для эффективного продолжения процесса оптимизации с того момента, когда ухудшилась сходимость текущего метода. Существующие программные средства для установления наиболее подходящего метода в основном ориентированы на последовательное применение методов и выполнение нескольких итераций каждым из них. Уровень интеллекта таких программ обычно оценивается по их возможностям осуществлять выбор метода без участия пользователя. Современные операционные системы позволяют обеспечить решение задачи путем организации параллельных вычислительных потоков для одновременного проведения расчетов несколькими методами. В каждом таком потоке можно реализовывать итерационный процесс одного из методов оптимизации и решение одной задачи вести несколькими методами одновременно. На многопроцессорных компьютерах для реализации каждого метода удобнее использовать отдельный процессор. После нахождения очередного приближения все методы оцениваются, например, по полученному приращению функционала, и из них выбирается наиболее эффективный метод для продолжения оптимизации, а полученное этим методом приближение передается остальным методам в качестве начального для выполнения следующей итерации. Начиная с этого приближения, всеми методами вновь выполняется одна или несколько итераций, и среди полученных приближений опять берется то, на котором достигнуто меньшее значение функционала.



**Фигура.** Схема выполнения  $k + 1$ -й итерации многометодным алгоритмом для группы из трех методов:  $M_1, M_2, M_3$ .

Продолжая таким образом итерационный процесс до получения приближения, на котором с заданной точностью будет выполнен критерий оптимальности, найдем приближенное решение задачи. При этом решение будет найдено многометодным алгоритмом, состоящим из последовательности шагов разных методов, подключаемых к процессу оптимизации с целью ускорения его сходимости. Например, в случае параллельного использования трех методов (см. фигуру) лучшее приближение будет определяться по максимуму приращения функционала, полученного на данной итерации каждым из трех методов

$$u_{i_0} = \arg \max_{i \in \{1,2,3\}} (I(u_i^k) - I(u_i^{k-1})).$$

Затем это приближение передается всем трем методам для выполнения следующей итерации:

$$u_i^{k+1} = u_{i_0}, i = 1, 2, 3.$$

Заметим, что из одной и той же группы методов оптимизации при решении другой задачи может сгенерироваться совсем другой многометодный алгоритм, который будет более адекватно учитывать особенности новой задачи.

Следует также отметить, что выбор метода по приращению функционала будет работать правильно только тогда, когда всеми методами будет использоваться один и тот же способ аппроксимации управления и, кроме того, во всех методах будет обеспечиваться одинаковая точность интегрирования систем. В этом случае все методы оптимизации будут применяться к одной и той же полученной в результате дискретизации конечномерной задаче и значения параметров, по которым осуществляется выбор метода, будут давать более точную оценку эффективности каждого из методов.

Для поиска численного решения краевой задачи, возникающей после применения принципа максимума [2], в статье предлагается многометодная схема, которая состоит из итераций градиентного метода, приводящего решение в окрестность сходимости метода квазилинеаризации (см. [3]), и непосредственно итераций метода квазилинеаризации, обеспечивающего вычисление оптимального управления с заданной точностью.

Далее рассматривается многокритериальная задача оптимального управления и для поиска ее численного решения также предлагается многометодная схема, состоящая из итераций прямого и двойственного методов, обеспечивающая получение “идеального” управления.

Современные информационные технологии и многопроцессорная вычислительная техника позволяют создать программное обеспечение, реализующее многометодную технологию с параллельными вычислениями (см. [4]) для оптимизации сложных систем управления.

## 1. МНОГОМЕТОДНЫЙ АЛГОРИТМ ДЛЯ РЕШЕНИЯ ЗАДАЧИ ПЕРЕВОДА НЕЛИНЕЙНОГО ОБЪЕКТА

Одной из наиболее распространенных задач оптимального управления является задача перевода нелинейного объекта из одного состояния в другое по некоторому заданному критерию качества:

$$I(u) = \int_{t_0}^{t_1} F(x, u, t) dt \rightarrow \min, \quad \dot{x} = f(x, u, t), \quad (1.1)$$

$$x(t_0) = x^0, \quad x(t_1) = x^1, \quad x(t) \in R^n, \quad u(t) \in R^r, \quad t \in [t_0, t_1].$$

В виде (1.1) формулируются, например, задачи управления маневром и стабилизацией космического аппарата (см. [5]).

Для частного класса задач, допускающего явное выражение управления из условия  $H_u = 0$ , построим эффективный алгоритм метода квазилинеаризации [3], в котором хорошее начальное приближение отыскивается специальным градиентным методом. После применения принципа максимума будем иметь следующую краевую задачу:

$$\begin{aligned} \dot{x} &= X(x, \psi, t), & \dot{\psi} &= \Psi(x, \psi, t), \\ x(t_0) &= x^0, & x(t_1) &= x^1. \end{aligned} \tag{1.2}$$

Пусть известно достаточно хорошее начальное приближение, построенное, например, из физических или содержательных соображений. Тогда краевую задачу (1.2) можно решить методом квазилинеаризации, обеспечивающим высокую точность выполнения краевых условий. На итерациях этого метода рассматривается линеаризованная система

$$\dot{x}^{k+1} = X_x(x^k, \psi^k, t)x^{k+1} + X_\psi(x^k, \psi^k, t)\psi^{k+1} + \rho^k(t), \tag{1.3}$$

$$\dot{\psi}^{k+1} = \Psi_x(x^k, \psi^k, t)x^{k+1} + \Psi_\psi(x^k, \psi^k, t)\psi^{k+1} + \eta^k(t), \tag{1.4}$$

где  $k$  – номер итерации;

$$\rho^k = X(x^k, \psi^k, t) - X_x(x^k, \psi^k, t)x^k(t) - X_\psi(x^k, \psi^k, t)\psi^k(t),$$

$$\eta^k = \Psi(x^k, \psi^k, t) - \Psi_x(x^k, \psi^k, t)x^k(t) - \Psi_\psi(x^k, \psi^k, t)\psi^k(t).$$

С помощью фундаментальной матрицы

$$\Phi(t) = \begin{bmatrix} \Phi_{11}(t) & \Phi_{12}(t) \\ \Phi_{21}(t) & \Phi_{22}(t) \end{bmatrix},$$

удовлетворяющей  $2n$ -мерному уравнению

$$\dot{\Phi} = \begin{bmatrix} X_x(x^k, \psi^k, t) & X_\psi(x^k, \psi^k, t) \\ \Psi_x(x^k, \psi^k, t) & \Psi_\psi(x^k, \psi^k, t) \end{bmatrix} \Phi, \quad \Phi(t_0) = E, \tag{1.5}$$

решение системы (1.3), (1.4), исходящее из начальной точки  $(x^{k+1}(t_0), \psi^{k+1}(t_0))$ , записывается по формулам

$$\begin{aligned} x^{k+1}(t) &= \Phi_{11}(t)x^{k+1}(t_0) + \Phi_{12}(t)\psi^{k+1}(t_0) + \bar{x}^{k+1}(t), \\ \psi^{k+1}(t) &= \Phi_{21}(t)x^{k+1}(t_0) + \Phi_{22}(t)\psi^{k+1}(t_0) + \bar{\psi}^{k+1}(t), \end{aligned} \tag{1.6}$$

где  $(\bar{x}^{k+1}(t_0)', \bar{\psi}^{k+1}(t_0)')$  – частное решение системы (1.3), (1.4), соответствующее начальным условиям  $\bar{x}^{k+1}(t_0) = 0, \bar{\psi}^{k+1}(t_0) = 0$ .

Пусть решение (1.6) удовлетворяет краевому условию

$$\Phi_{11}(t_1)x^0 + \Phi_{12}(t_1)\psi^{k+1}(t_0) + \bar{x}^{k+1}(t_1) = x^1.$$

Тогда новое приближение для вектора начальных условий сопряженной системы  $\psi(t_0)$  найдем по формуле

$$\psi^{k+1}(t_0) = \Phi_{12}^{-1}(t_1)[x^1 - \Phi_{11}(t_1)x^0 - \bar{x}^{k+1}(t_1)]. \tag{1.7}$$

Интегрируя систему (1.2) при начальных условиях  $x(t_0) = x^0, \psi(t_0) = \psi^{k+1}(t_0)$ , получаем точку  $x^{k+1}(t_1)$ , которая удовлетворяет неравенству

$$\|x^{k+1}(t_1) - x^1\| < \|x^k(t_1) - x^1\|.$$

Для получения первого приближения  $\psi(t_0) = \psi^k$ , достаточно близкого к искомому вектору  $\psi^*(t_0)$ , построим градиентную процедуру, не требующую хорошего начального приближения для сходимости итерационного процесса.

Введем функционал  $I_1(v) = \frac{1}{2}\|x(t_1, v) - x^1\|^2$ , который является мерой отклонения решения системы (1.2) с начальными условиями  $x(t_0) = x_0, \psi(t_0) = v$  от заданной точки  $x^1$ , и поставим задачу

поиска  $v^* = \arg \min_{v \in R^*} I_1(v)$ . Поскольку теперь будет решаться задача со свободным правым концом, решение сопряженной системы  $\psi(t)$  должно удовлетворять условию трансверсальности

$$\psi(t_1) = x^1 - x(t_1, v). \quad (1.8)$$

Построим функционал невязок

$$I(v) = I_1(v) + \frac{1}{2} \|\psi(t_1) + x(t_1, v) - x^1\|^2 \quad (1.9)$$

и будем рассматривать задачу минимизации функционала (1.9) на решениях системы (1.2) с начальными условиями  $x(t_0) = x^0$ ,  $\psi(t_0) = v$ . Градиент функционала (1.9) по параметрам  $(v_1, \dots, v_n)$ , определяющим начальные условия системы (1.2), вычисляется через решение системы, сопряженной к (1.2):

$$\begin{aligned} \dot{\xi} &= -X'_x(x, \psi, t)\xi - \Psi'_x(x, \psi, t)\zeta, \\ \dot{\zeta} &= -X'_\psi(x, \psi, t)\xi - \Psi'_\psi(x, \psi, t)\zeta, \\ \xi(t_1) &= 2(x^1 - x(t_1, v) - \psi(t_1)), \\ \zeta(t_1) &= x^1 - x(t_1, v) - \psi(t_1), \end{aligned} \quad (1.10)$$

и равен  $\nabla I(v) = -\zeta(t_0)$ .

### Алгоритм 1

Для улучшения вектора  $v_k$  начальных условий сопряженной системы можно применить градиентную процедуру, состоящую из следующих операций:

**Шаг 1.** Интегрируется система (1.2) при  $x(t_0) = x^0$ ,  $\psi(t_0) = v^k$ , в узлах интегрирования запоминается решение  $x^k(t)$ ,  $\psi^k(t)$ ,  $t \in T$ .

**Шаг 2.** В обратном времени интегрируется система линейных дифференциальных уравнений (1.10), матрица коэффициентов которой вычисляется на решениях  $x^k(t)$  и  $\psi^k(t)$ ; при  $t = t_0$  находится градиент  $\nabla I(v) = -\zeta(t_0)$ .

**Шаг 3.** С помощью процедуры одномерного поиска отыскивается параметр

$$\alpha_k = \arg \min_{\alpha \geq 0} I(v^k + \alpha \zeta(t_0)).$$

При этом с каждым значением  $\alpha$  решается система (1.2) при  $x(t_0) = x^0$ ,  $\psi(t_0) = v^k + \alpha \zeta(t_0)$ .

**Шаг 4.** Строится новое приближение  $v^{k+1} = v^k + \alpha \zeta(t_0)$ . Если  $I(v^k) - I(v^{k+1}) > \varepsilon$ , то при  $k = k + 1$  повторяются операции ш. 1–3. В противном случае с ш. 5 выполняются операции метода квазилинеаризации.

**Шаг 5.** Решается матричное уравнение (1.5), и запоминается  $\Phi(t_1)$ .

**Шаг 6.** По формуле (1.7) находится вектор  $\psi^{k+1}(t_0)$ .

**Шаг 7.** Выполняются операции на ш. 1 при  $\psi(t_0) = v^k = \psi^{k+1}(t_0)$ , в результате находится  $x^{k+1}(t_1)$ .

**Шаг 8.** Если выполняются неравенства  $\|x^{k+1}(t_1) - x^1\| < \|x^k(t_1) - x^1\|$  и  $\|x^{k+1}(t_1) - x^1\| > \varepsilon$ , то повторяются операции ш. 5–7.

**Шаг 9.** Если  $\|x^{k+1}(t_1) - x^1\| < \|x^k(t_1) - x^1\|$ , то уменьшается шаг интегрирования, повышается точность одномерного поиска и вновь повторяются операции градиентной процедуры 1–4, а затем операции метода квазилинеаризации 5–7.

**Шаг 10.** Если  $\|x^{k+1}(t_1) - x^1\| \leq \varepsilon$ , т.е. точность выполнения краевых условий достигнута, то итерационный процесс прекращается.

Оптимальное управление вычисляется из условия  $H_u = 0$  путем интегрирования системы (1.2) с найденными начальными условиями  $x(t_0) = x^0$ ,  $\psi(t_0) = \psi^{k+1}(t_0)$ .

Трудоемкость выполнения операций ш. 1–3 оценивается трудоемкостью интегрирования  $2n(2 + \nu)$  скалярных уравнений, где  $\nu$  – оценка сверху числа итераций одномерного поиска. Тру-

доемкость итерации метода квазилинеаризации эквивалентна интегрированию  $2n + 4n^2$  скалярных уравнений.

## 2. МНОГОКРИТЕРИАЛЬНАЯ ЗАДАЧА ОПТИМАЛЬНОГО УПРАВЛЕНИЯ

Пусть управляемый процесс описывается системой

$$\dot{x} = f(x, u, t), \quad x(t_0) = x_0, \quad t \in T = [t_0, t_1], \quad (2.1)$$

где  $x(t) \in R^n$ ,  $u(t) \in R^r$ ; вектор-функция  $f(x, u, t)$  непрерывно дифференцируема по  $x$  и  $u$  и непрерывна по  $t$ , а управление стеснено ограничениями  $u(t) \in U$ ,  $U$  – ограниченное замкнутое множество. На правых концах траекторий системы (2.1) определим непрерывно дифференцируемые выпуклые скалярные функции  $\varphi_j(x(t_1))$ ,  $j = \overline{1, m}$ , которые будут играть роль критериев качества управления:

$$I_j(u) = \varphi_j(x(t_1)), \quad j = \overline{1, m}. \quad (2.2)$$

Многоцелевое управление не может быть оптимальным по всем заданным критериям (2.2) одновременно, поэтому речь будет идти о поиске только эффективных (или слабо эффективных) решений [6].

### 2.1. Алгоритмы прямого поиска численного решения в многокритериальной задаче

Наиболее простой способ для нахождения таких решений состоит в решении последовательности однокритериальных задач с функционалом вида свертки

$$I(\lambda, u) = \sum_{i=1}^m \lambda_i I_i(u) \rightarrow \min_{u \in U} \quad (2.3)$$

при разных значениях  $\lambda = (\lambda_1, \dots, \lambda_m)$ :  $\lambda_i \geq 0$ ,  $\sum_{i=1}^m \lambda_i = 1$ . Известно (см. [6]), что любое оптимальное решение однокритериальной задачи (2.1), (2.3) будет парето-оптимальным решением.

В соответствии с заданными функционалами (2.2) введем сопряженные системы

$$\dot{\psi}_j = -f'_x(x, u, t)' \psi_j(t), \quad \psi_j(t_1) = -\varphi'_j(x(t_1)), \quad j = \overline{1, m}, \quad (2.4)$$

и с помощью их решений  $\psi_j = \psi_j(t)$  определим функции Гамильтона  $H^j(\psi_j, x, u, t) = \psi'_j(t) f(x, u, t)$ ,  $j = \overline{1, m}$ . Далее сформулируем принцип максимума для однокритериальной задачи (2.1), (2.3).

Пусть  $u^*(t)$ ,  $x^*(t)$ ,  $t \in T$ , – оптимальное управление и оптимальная траектория. Тогда существует вектор-функция  $\psi_\lambda = \psi_\lambda(t)$  – решение сопряженной системы

$$\dot{\psi}_\lambda = -f'(x^*, u^*, t)' \psi_\lambda(t), \quad \psi_\lambda(t_1) = -\sum_{j=1}^m \lambda_j \varphi'_j(x^*(t_1)) \quad (2.5)$$

такая, что при каждом  $t \in T$  для функции  $H^\lambda(\psi_\lambda, x, u, t) = \psi'_\lambda(t) f(x, u, t)$  выполняется условие

$$H^\lambda(\psi_\lambda, x^*, u^*, t) = \max_{u \in U} H^\lambda(\psi_\lambda, x^*, u, t). \quad (2.6)$$

Рассмотрим алгоритм поиска управления, удовлетворяющего принципу максимума (2.6) при некотором заданном  $\lambda$ .

### Алгоритм 2

Предположим, что имеем некоторое заданное управление  $u^k(t) \in U$ ,  $t \in T$ , как начальное приближение.

**Шаг 1.** Интегрируется уравнение (2.1) при  $u = u^k(t)$ , в узлах интегрирования запоминается траектория  $x^k(t)$ .

**Шаг 2.** В обратном времени интегрируется сопряженная система  $\Psi_\lambda = -f'_x(x^k, u^k, t) \Psi_\lambda(t)$ ,  $\Psi_\lambda(t_1) = -\sum_{j=1}^m \lambda_j \varphi'_x(x^k(t_1))$ , при этом в каждом узле интегрирования вычисляются и запоминаются управление

$$\bar{u}^k(t) = \arg \max_{u \in U} H^\lambda(\Psi^k, x^k, u, t), \quad t \in T,$$

и скалярная функция

$$w_k(\bar{u}^k(t), t) = H^\lambda(\Psi^k, x^k, \bar{u}^k, t) - H^\lambda(\Psi^k, x^k, u^k, t), \quad t \in T.$$

**Шаг 3.** Находится точка  $\tau_k$ :  $w_k(\bar{u}^k(\tau_k), \tau_k) = \max_{t \in T} w_k(\bar{u}^k(t), t)$ . Если  $w_k(\bar{u}^k(\tau_k), \tau_k) = 0$ , то  $u = u^k(t)$  удовлетворяет принципу максимума и итерационный процесс прекращается.

**Шаг 4.** Методом одномерной минимизации решается задача

$$I(\lambda, u_\varepsilon) \rightarrow \min_{\varepsilon \in [0,1]}, \quad u_\varepsilon(t) = \begin{cases} \bar{u}^k(t), & \text{если } w_k(t) \geq \varepsilon w_k(\tau_k), \\ u^k(t), & \text{если } w_k(t) < \varepsilon w_k(\tau_k), \end{cases}$$

где  $w_k(t) = w_k(\bar{u}^k, t)$ .

**Шаг 5.** При найденном оптимальном значении  $\varepsilon_k$  построим новое приближение для управления:

$$u^{k+1}(t) = u_{\varepsilon_k}(t), \quad (2.7)$$

которому соответствует наименьшее значение функционала на данной итерации.

Самыми простыми методами одномерной минимизации являются метод деления отрезка пополам и метод золотого сечения. Меньшее число пробных значений переменной требует метод парабол. Но в некоторых случаях необходимо применение методов поиска глобального минимума, которые требуют существенно большего числа пробных значений  $\varepsilon$ . При поиске  $\varepsilon_k$  можно использовать несколько потоков для одновременного интегрирования системы (2.1) с управлениями  $u_\varepsilon(t)$ ,  $t \in T$ , соответствующими разным значениям  $\varepsilon$ .

**Шаг 6.** При незначительном уменьшении функционала на ш. 4–5 (плохая сходимость) новое приближение для управления строится по формуле

$$u^{k+1}(t) = u^k(t) + \alpha[\bar{u}^k(t) - u^k(t)], \quad \alpha \in [0, 1], \quad (2.8)$$

где для определения параметра  $\alpha \in [0, 1]$  аналогично, как на ш. 4 для  $\varepsilon$ , применяется одномерный поиск. Для этого с выбранным  $\alpha$  строится управление (2.8), затем с этим управлением интегрируется система (2.1), а в момент  $t = t_1$  находится значение функционала. Меняя значение  $\alpha$ , метод одномерной минимизации находит его оптимальное значение, при котором на данной итерации функционал  $I(\lambda, u^{k+1})$  достигает наименьшего значения.

**Шаг 7.** Если на ш. 4–6 будет найдено новое приближения для управления, на котором достигается меньшее значение функционала, чем на предыдущей итерации, то выполняется новая итерация на ш. 1.

В противном случае итерационный процесс прекращается.

Как уже отмечалось, в силу структуры управлений, генерируемых итерационной формулой (2.7), погрешность численного интегрирования системы может стать причиной ухудшения сходимости еще до получения управления, удовлетворяющего принципу максимума. Тогда для продолжения процесса оптимизации необходимо применить аналог метода условного градиента (2.8), на итерациях которого конструируются управления не только с граничными, но и с внутренними относительно множества допустимых управлений значениями. Вычисления по формулам (2.7) и (2.8) проводятся параллельно с выбором такого  $u^{k+1}$ , которому соответствует меньшее значение функционала. Если в итерационной формуле (2.8) управление  $\bar{u}^k(t)$  будет вычисляться из линейризованного принципа максимума:  $\bar{u}^k(t) = \arg \max_{u \in U} H_u(\Psi^k, x^k, u^k, t) u(t)$ ,  $t \in T$ , то будем иметь итерации метода условного градиента. Для систем, линейных по управлению,  $\bar{u}^k(t)$  будет совпадать с управлением, найденным из принципа максимума.

Итерационные процессы (2.7), (2.8) должны завершиться получением управления, для которого выполнится необходимое условие оптимальности  $w_k(\bar{u}^k(\tau_k), \tau_k) = 0$ . При этом будет найдено одно из парето-оптимальных решений, которое соответствует заданному вектору  $\lambda$ .

Поиск приближенного решения многокритериальной задачи, которое было бы приемлемым по нескольким критериям сразу, можно вести и среди парето-оптимальных решений, получаемых в результате применения описанного алгоритма 2 к последовательности однокритериальных задач (2.1), (2.3) при некоторых значениях вектора  $\lambda$ , заданных экспертами. При небольшом числе критериев перебор значений  $\lambda$  — достаточно простая процедура.

Другой подход к поиску решения многокритериальной задачи состоит в нахождении “идеального” управления, на котором достигается минимум функционала

$$R(u) = \sum_{i=1}^m (I_i(u) - I_i^0)^2, \quad (2.9)$$

где значения  $I_i^0$ ,  $i = \overline{1, m}$ , найдены путем решения однокритериальной задачи  $I_i(u) \rightarrow \min_{u \in U}$ . Функционал (2.9) обеспечивает приближение, наиболее близкое к однокритериальным решениям.

Заметим, что до минимизации функционала (2.9) необходимо решить  $m$  однокритериальных задач оптимального управления для системы (2.1) с функционалами (2.2), а затем найти решение однокритериальной задачи с функционалом (2.9). Для решения многокритериальной задачи (2.1), (2.2), (2.9) можно использовать следующий алгоритм. Организовать  $m$  параллельных процессов для минимизации функционалов (2.2) на решениях системы (2.1) при допустимых управлениях из  $U$ . При этом для минимизации каждого из функционалов можно использовать алгоритм 2.

После удачного завершения процесса оптимизации получим наименьшие значения функционалов (2.2):  $I_i^0$ ,  $i = \overline{1, m}$ . Далее применим алгоритм 2 для минимизации функционала (2.9). При этом сопряженная система будет иметь следующее конечное условие:

$$\psi(t_1) = -2 \sum_{j=1}^m (\varphi^j(x(t_1)) - I_j^0) \varphi_x^j(x(t_1)),$$

т.е. являться линейной комбинацией градиентов критериев (2.2), а в ш. 4–6 алгоритма 1 будет отыскиваться минимум функционала (2.9). Итерационные процессы (2.7), (2.8), реализованные в алгоритме 2, должны обеспечить получение приближенного решения многокритериальной задачи: оптимального управления и соответствующей ему траектории, на которых функционал (2.9) достигает минимума. Другими словами, полученное управление будет обеспечивать перевод системы (2.1) в точку, которая наиболее близка к точкам минимума критериев (2.2), и поэтому носит название идеального решения.

## 2.2. Двойственный метод для решения многокритериальной задачи

Изложенные выше способы решения многокритериальной задачи относятся к классу прямых методов. Рассмотрим теперь двойственный алгоритм, в основе которого лежит следующее

**Утверждение.** *Задача минимизации функционала (2.3) является двойственной по отношению к задаче минимизации функционала (2.9).*

Действительно, если в  $m$ -мерном пространстве критериев рассматривать вектор  $\lambda$ ,  $\|\lambda\| \leq 1$ , и вектор  $y$  с координатами  $y_j = \varphi^j(x(t_1)) - I_j^0$ ,  $j = 1, 2, \dots, m$ , то задачу (2.9) можно представить в виде  $\|y\|^2 \rightarrow \min$ , что эквивалентно задаче  $\|y\| \rightarrow \min$ . По определению нормы имеем  $\|y\| = \max_{|\lambda| \leq 1} (\lambda, y)$ . Тогда  $\min \|y\| = \min_{|\lambda| \leq 1} \max (\lambda, y)$ . Так как функции  $\varphi^j(x)$  выпуклые, то и определяемое ими множество в пространстве критериев (на множестве достижимости) выпукло. Множество  $\|\lambda\| \leq 1$ , очевидно, также выпукло. Следовательно, операции  $\min$  и  $\max$  допускают перестановку:  $\min \|y\| = \min_{|\lambda| \leq 1} \max (\lambda, y) = \max_{|\lambda| \leq 1} \min (\lambda, y)$ . Таким образом, вместо минимизации функционала (2.9) можно решать задачу максимизации минимума функционала (2.3) на единичной сфере  $\|\lambda\| \leq 1$ . В результате получим  $\lambda^*$ , при котором решение задачи (2.1), (2.2) будет обеспечивать минимум

функции (2.9). Эффективный алгоритм нахождения единичного по норме вектора  $\lambda^*$  рассматривается в [1] и носит название метода выпуклых оболочек. Здесь рассмотрим двойственный алгоритм 3, на внешних итерациях которого будет вестись максимизация по  $\lambda$ , а на внутренних итерациях – минимизация по  $u$ :

$$\max_{\|\lambda\| \leq 1} \min_{u \in U} I(\lambda, u) = \max_{\|\lambda\| \leq 1} \min_{x(t_1)} \sum_{j=1}^m \lambda_j (\varphi^j(x(t_1)) - I_j^0).$$

### Алгоритм 3

**Шаг 1.** Задаем начальное приближение для коэффициентов свертки, например,  $\lambda_j = 1/m$ ,  $j = \overline{1, m}$ .

**Шаг 2.** Применяя алгоритм 2 для минимизации функционала  $I(\lambda, u) = \sum_{j=1}^m \lambda_j (\varphi^j(x(t_1)) - I_j^0)$  по  $u$  при фиксированном векторе  $\lambda$ , найдем решение внутренней задачи. При этом на ш. 4–6 при поиске оптимальных значений параметров  $\varepsilon$  и  $\alpha$  будет минимизироваться данный функционал

$$I(\lambda, u) = \sum_{j=1}^m \lambda_j (\varphi^j(x(t_1)) - I_j^0).$$

**Шаг 3.** По окончании работы алгоритма 2 будет найдено управление, приводящее систему (2.1) в точку  $x(t_1)$ , в которой для заданного  $\lambda$  функционал  $I(\lambda, u)$  будет минимальным, а полученные при этом значения  $\varphi_j(x(t_1)) - I_j^0$  будут компонентами градиента функционала  $I(\lambda, u)$  по  $\lambda$ :

$$I(\lambda, u) = \varphi^j(x(t_1)) - I_j^0, \quad j = \overline{1, m}.$$

**Шаг 4.** Применяя метод одномерного поиска для максимизации по  $\alpha$ , найдем оптимальный шаг  $\alpha$  вдоль градиента для изменения коэффициентов вектора  $\lambda$  с целью увеличения значения  $I(\lambda, u)$ :  $\lambda_j^{k+1} = \lambda_j^k + \alpha(\varphi^j(x(t_1)) - I_j^0)$ ,  $j = \overline{1, m}$ ; затем полагаем  $\lambda_j^{k+1} = \lambda_j^k / \|\lambda\|$ ,  $j = \overline{1, m}$ . Заметим, что на практике из-за трудоемкости поиска шаг  $\alpha$  иногда выбирают достаточно малым (необязательно оптимальным), лишь бы увеличить значение функционала.

**Шаг 5.** Если на ш. 4 достигнуто выполнение неравенства  $I(\lambda^{k+1}, u) > I(\lambda^k, u)$ , то, полагая  $\lambda = \lambda^{k+1}$ , переходим на ш. 2 для продолжения процесса оптимизации. В противном случае итерационный процесс прекращается и найденное на последней итерации управление будет решением многокритериальной задачи в смысле минимума функционала (2.9).

## 3. ЧИСЛЕННОЕ РЕШЕНИЕ ПРИКЛАДНЫХ ЗАДАЧ

### 3.1. Оптимальное управление положением асимметричного космического аппарата

На основе предложенной многометодной технологии было найдено решение следующей нелинейной краевой задачи. Рассмотрим модель, описывающую вращательную динамику космического аппарата (КА), которая приведена в статье [5]:

$$J(u) = \frac{1}{2} \int_0^{100} [u_1^2(t) + u_2^2(t) + u_3^2(t)] dt \rightarrow \min,$$

$$\dot{\beta}_0 = k(-\omega_1\beta_1 - \omega_2\beta_2 - \omega_3\beta_3),$$

$$\dot{\beta}_1 = k(\omega_1\beta_0 - \omega_2\beta_3 + \omega_3\beta_2),$$

$$\dot{\beta}_2 = k(\omega_1\beta_3 + \omega_2\beta_0 - \omega_3\beta_1),$$

$$\dot{\beta}_3 = k(-\omega_1\beta_2 + \omega_2\beta_1 + \omega_3\beta_0), \quad k = 0.5,$$

$$\dot{\omega}_1 = -\bar{I}_1\omega_2\omega_3 + u_1/I_1,$$

$$\dot{\omega}_2 = -\bar{I}_2\omega_1\omega_3 + u_2/I_2,$$

$$\dot{\omega}_3 = -\bar{I}_3\omega_1\omega_2 + u_3/I_3,$$

где  $\bar{I}_1 = (I_3 - I_2)/I_1$ ,  $\bar{I}_2 = (I_1 - I_3)/I_2$ ,  $\bar{I}_3 = (I_2 - I_1)/I_3$ ,  $(I_1, I_2, I_3)$  – моменты инерции космического аппарата;  $(u_1, u_2, u_3)$  – компоненты управляющего вектора (вращающего момента);  $(\beta_0, \beta_1, \beta_2, \beta_3)$  – параметры Эйлера,  $(\omega_1, \omega_2, \omega_3)$  – компоненты угловой скорости.

Вектор  $\beta$  удовлетворяет условию  $\|\beta(t)\|^2 = \text{const}$ . Выбором начальных условий обеспечивается равенство  $\|\beta(t)\| = 1$ .

Требуется найти управление, обеспечивающее минимум функционалу  $J(u)$  и переводящее КА из заданного начального положения в заданное конечное состояние.

В [5] заданы следующие начальное и конечное состояния:  $x^0 = (1, 0, 0, 0, 0.01, 0.005, 0.001)$ ,  $x^1 = (0.43047, 0.70106, 0.09230, 0.56098, 0, 0, 0)$ , моменты инерции:  $I_1 = 10^6$ ,  $I_2 = 833333$ ,  $I_3 = 916667$ , а начальное приближение для управления, с которого метод квазилинеаризации начинал сходиться к оптимальному, находилось аналитически из механических свойств системы.

Управление, на котором достигается максимум зависящей от  $u$  части функции Понтрягина,  $Z(u) = \psi_5 u_1 / I_1 + \psi_6 u_2 / I_2 + \psi_7 u_3 / I_3 - u_1^2 - u_2^2 - u_3^2$ , определяется по формулам:  $u_1 = \psi_5 / (2I_1)$ ,  $u_2 = \psi_6 / (2I_2)$ ,  $u_3 = \psi_7 / (2I_3)$ .

Краевая задача (1.2) для данной системы имеет следующий вид:

$$\begin{aligned} \dot{x}_1 &= k(-x_5 x_2 - x_6 x_3 - x_7 x_4), \\ \dot{x}_2 &= k(x_5 x_1 + x_7 x_3 - x_6 x_4), \\ \dot{x}_3 &= k(x_6 x_1 - x_7 x_2 + x_5 x_4), \\ \dot{x}_4 &= k(x_7 x_1 + x_6 x_2 - x_5 x_3), \\ \dot{x}_5 &= -0.08333 x_6 x_7 + \psi_5 / (2I_1^2), \\ \dot{x}_6 &= -0.1 x_5 x_7 + \psi_6 / (2I_2^2), \\ \dot{x}_7 &= 0.18182 x_5 x_6 + \psi_7 / (2I_3^2), \\ \dot{\psi}_1 &= k(\psi_2 x_5 + \psi_3 x_6 + \psi_4 x_7), \\ \dot{\psi}_2 &= k(\psi_1 x_5 + \psi_3 x_7 - \psi_4 x_6), \\ \dot{\psi}_3 &= k(\psi_1 x_6 - \psi_2 x_7 + \psi_4 x_5), \\ \dot{\psi}_4 &= k(\psi_1 x_7 + \psi_2 x_6 - \psi_3 x_5), \\ \dot{\psi}_5 &= k(\psi_1 x_2 - \psi_2 x_1 - \psi_3 x_4 + \psi_4 x_3) + 0.1 \psi_6 x_7 - 0.18182 \psi_7 x_6, \\ \dot{\psi}_6 &= k(\psi_1 x_3 + \psi_2 x_4 - \psi_3 x_1 - \psi_4 x_2) + 0.08333 \psi_5 x_7 - 0.18182 \psi_7 x_5, \\ \dot{\psi}_7 &= k(\psi_1 x_4 - \psi_2 x_3 + \psi_3 x_2 - \psi_4 x_1) + 0.08333 \psi_5 x_6 + 0.1 \psi_6 x_5. \end{aligned}$$

Краевые условия были заданы такими же, как и в [5]. Они приведены выше. Максимальное нарушение краевого условия на правом конце при нулевом управлении было равно 0.41622. Метод квазилинеаризации с этого управления не сходилась. Градиентная процедура (алгоритм 1), начиная с  $v^0$ , соответствующего нулевому управлению, сократила невязку выполнения краевых условий на порядок, после чего с полученного  $v^k$  метод квазилинеаризации стал сходиться и нарушение краевых условий уменьшилось до  $10^{-9}$ . График полученного при этом управления практически совпадает с графиком управления, приведенного в [5].

Таким образом, многометодная схема в данной краевой задаче состоит из итераций двух методов: итераций градиентного метода, которые приводят в окрестность сходимости метода квазилинеаризации, и непосредственно итераций метода квазилинеаризации, которые обеспечивают вычисление оптимального управления с заданной точностью. Заметим, что ни одним из этих методов при его автономном использовании от начала и до конца невозможно было получить решение краевой задачи с удовлетворительной для практики точностью.

### 3.2. Оптимальное управление в задаче слежения за маневрирующей целью

Рассматривается прикладная задача управления маневром самолета при слежении за ним средствами наблюдения, например, РЛС. Необходимость применения нескольких критериев и многометодной технологии возникает, например, в задачах слежения за пилотируемой маневрирующей целью [7]. В силу особенности движения целей этого класса (самолет, корабль, подводная лодка), которая состоит в их способности к маневру – разворотам и быстрым изменениям скорости и высоты, модель оценивания движения в реальном времени при слежении за такой управляемой пилотируемой целью должна гарантировать необходимую точность слежения. Надежность расчета оптимального управления в этой сложной нелинейной модели обеспечивается применением арсенала эффективных математических методов для построения многометодных алгоритмов поиска численного решения [1].

**3.2.1. Математическая модель задачи управления и слежения.** Предположим, что сопровождаемая цель обладает хорошими маневренными свойствами и ее движение можно описать уравнением

$$\dot{x} = Fx + Gw, \quad x(0) = M[x(0)], \quad (3.1)$$

где  $x = x(t)$  – расширенный вектор состояния цели, включающий координаты формирующего фильтра,  $w = w(t)$  – случайный процесс в виде белого шума с корреляционной матрицей  $M[w(t)w(t)'] = Q(t)\delta(t - \tau)$ ,  $F, G$  – матрицы, определяющие параметр фильтра, формирующего  $x(t)$ . Информация о текущем положении цели (3.1) поступает от измерителя (РЛС):  $z = Hx + \xi + v$ , где  $H$  – матрица коэффициентов измерителя,  $\xi$  – случайный шум измерения с корреляционной матрицей  $M[\xi(t)\xi(t)'] = R(t)\delta(t - \tau)$ ,  $v = v(t)$  – помеха, создаваемая целью. Тогда структура фильтра, обеспечивающего минимум среднеквадратической ошибки  $M[(x - y)(x - y)']$ , где  $y$  – оценка вектора состояния  $x$ , имеет вид

$$\dot{y} = Fy + K(z - Hy), \quad y(0) = M[x(0)], \quad K = PHR^{-1}, \quad (3.2)$$

$$\dot{P} = FP - PH'R^{-1}HP + PF' + GQG', \quad P(0) = M[x(0)x(0)']. \quad (3.3)$$

В отличие от предполагаемой модели (3.1) реальное движение самолета описывается уравнением

$$\dot{x}_g = F_g G_g u + \eta, \quad x_g(0) = M[x_g(0)], \quad (3.4)$$

где  $x_g \in X$  – действительный вектор состояния самолета,  $u \in V$  – вектор управления из заданного множества,  $\eta$  – формирующий белый шум модели.

Из уравнений (3.2)–(3.4) получим уравнение для ошибки фильтрации

$$\dot{\varepsilon} = x_g - y: \quad \varepsilon = (F - KH)\varepsilon + Gu - Kv + \Delta Fx_g + \eta - K\xi, \quad (3.5)$$

где  $\Delta F = F_g - F$ .

Используем соотношение  $M\|\varepsilon\|^2 = \|\bar{\varepsilon}\|^2 + TrP(t)$ , где  $\bar{\varepsilon}$  – математическое ожидание  $\varepsilon(t)$ ,  $TrP(t)$  – след ковариационной матрицы ошибок фильтрации. Учитывая, что  $TrP(t)$  не зависит от  $\eta$  и  $\xi$ , достаточно потребовать достижения  $\max_{u,v} \|\bar{\varepsilon}\|^2$ , где

$$\dot{\bar{\varepsilon}} = (F - KH)\bar{\varepsilon} + Gu - Kv + \Delta Fx_g. \quad (3.6)$$

**3.2.2. Возможные постановки задач оптимального управления с различными критериями.** Задачу синтеза оптимального противодействия процессу сопровождения движущейся цели можно сформулировать следующим образом [7]. Известны уравнения для ошибки фильтрации (3.5) и действительного движения самолета (3.4), заданы ограничения на траекторию:  $x_{gH} \leq x_g \leq x_{gB}$ , на управление:  $u_{\min} \leq u \leq u_{\max}$  и на помеху:  $v_{\min} \leq v \leq v_{\max}$ . Требуется найти такие значения для управления  $u$  и параметра  $v$ , при которых математическое ожидание квадрата ошибки фильтрации  $\|\varepsilon\|^2$  достигнет максимума за некоторое время  $T$ . Для описанной модели рассматривались следующие постановки задач оптимального управления.

1. Проверка управляемости системы, определяющая возможность перевода системы из начального состояния в некоторое заданное состояние за время  $T$ .

2. Задача оптимального быстрогодействия: вычисление управления, которое обеспечило бы достижение заданного уровня ошибки за минимально возможное время.

3. Нахождение управления, которое обеспечило бы достижение требуемого максимального уровня ошибки  $\epsilon_{\max}$  в конце заданного временного отрезка  $[0, T]$ .

4. Вычисление управления, которое обеспечивало бы в каждой точке  $t$  временного отрезка  $[0, T]$  требуемое максимальное значение ошибки  $\epsilon_{\max}$ .

Для перечисленных задач наиболее естественным является выбор двух критериев: дальности  $\epsilon_1$  и скорости цели  $\epsilon_2$  и, следовательно, решать задачу с двумя критериями.

**3.2.3. Численное решение.** В качестве иллюстрации рассмотрим конкретную задачу оптимального управления самолетом и помехой из [7]:

$$\begin{aligned}\dot{R} &= V, \\ \dot{V} &= a, \\ \dot{a} &= -\alpha a + u,\end{aligned}$$

где  $R$  – дальность,  $V$  – скорость,  $a$  – ускорение цели,  $u$  – управление,  $\alpha$  – предполагаемая интенсивность маневра.

Реальное движение самолета описывается уравнениями:

$$\begin{aligned}\dot{R} &= V, \\ \dot{V} &= a, \\ \dot{a} &= u.\end{aligned}\tag{3.7}$$

Перейдем к стандартной постановке задачи оптимального управления, объединив уравнения для ошибки слежения (3.6) движения (3.7) фильтра (3.3).

Совместная система уравнений (3.6), (3.7), (3.3) после введения вектора общих переменных

$$z = (\epsilon_1, \epsilon_2, \epsilon_3, R, V, a, p_{11}, p_{12}, p_{13}, p_{22}, p_{23}, p_{33})$$

(в силу симметричности матрицы  $P$  остальные ее элементы не включены) будет иметь следующий вид:

$$\begin{aligned}\dot{z}_1 &= z_2 - z_7(z_1 + v)/r, \\ \dot{z}_2 &= z_3 - z_8(z_1 + v)/r, \\ \dot{z}_3 &= 0.1z_6 - z_9(z_1 + v)/r + u, \\ \dot{z}_4 &= z_5, \\ \dot{z}_5 &= z_6, \\ \dot{z}_6 &= u, \\ \dot{z}_7 &= 2z_8 - z_7^2/r + 0.5\alpha, \\ \dot{z}_8 &= z_9z_{10} + z_7z_8/r + 0.12\alpha, \\ \dot{z}_9 &= z_{11} - z_7z_9/r + 0.166\alpha, \\ \dot{z}_{10} &= 2z_{11} - z_8^2/r + 0.333\alpha, \\ \dot{z}_{11} &= z_{12} - z_8z_9/r + 0.5\alpha, \\ \dot{z}_{12} &= z_9^2 + \alpha, \quad t \in [0, 10].\end{aligned}$$

Были заданы следующие начальные условия из [7]:

$$z(0) = (0, 0, 0, -18.3, 0.3, 0, 0.033489, 0.033489, 0, 0.066978, 0, 0.00046).$$

Искомыми являются управление  $u(t)$  и параметр  $v$ .

В качестве функционалов были выбраны ошибки измерения дальности и скорости цели:  $I_1(u) = z_1(10)$ ,  $I_2(u) = z_2(10)$ , которые необходимо максимизировать.

Задано ограничение на управление:  $|u(t)| \leq 0.0075$ ,  $t \in [0, 10]$  и на параметр:  $|v| \leq 0.0015$ . Значения других параметров были заданы следующими:  $r = 0.033489$ ,  $\alpha = 0.000092$ .

Решение найдено с помощью алгоритма 3 с применением методов из [1]. Оптимальные значения коэффициентов  $\lambda_1$  и  $\lambda_2$  примерно равны 0.5, т.е. критерии оказались равновесными.

Найденное оптимальное управление имеет релейную структуру, т.е. переключается с одной границы на другую (порядка десяти раз), при этом наблюдается рост ошибок сопровождения по дальности и по скорости, что согласуется с заданными критериями.

### 3.3. Нелинейная задача об оптимальном управлении электроприводом

В [8] нелинейная задача об оптимальном управлении электроприводом имеет вид

$$\begin{aligned}\dot{x}_1 &= x_2, \\ \dot{x}_2 &= u - x_2^2, \\ \dot{x}_3 &= u^2, \\ T &= [0, 4.85], \\ x(0) &= (0, 0, 0), \\ |u| &\leq 1, \\ \varphi_1 &= (x_1(4.85) - 3.12)^2, \\ \varphi_2 &= x_2^2(4.85), \\ \varphi_3 &= x_3(4.85).\end{aligned}$$

На полученном оптимальном управлении, которое оказалось внутренним относительно заданных ограничений, функционалы  $I_1(u) = \varphi_1$  и  $I_2(u) = \varphi_2$  достигли нулевых значений, а функционал  $I_3(u) = \varphi_3$ , стал равным 2.465441.

## ЗАКЛЮЧЕНИЕ

В заключение отметим, что был решен целый ряд задач подобного типа (см., например, [9], [10]), которые различались видами функционалов, интервалами времени или другими параметрами. Заметим также, что компьютерная технология управления реальными процессами накладывает высокие требования к расчету законов управления в экстремальных ситуациях и для других технических объектов. Точный и надежный расчет оптимального управления необходим, например, при приведении электроэнергетической системы в рабочий режим при резких изменениях нагрузки. Таким образом, многометодная технология решения многокритериальных задач оптимального управления, реализованная в виде параллельных итерационных процессов оптимизации с выбором лучшего приближения, решает задачу с автоматическим применением разных методов оптимизации и тем самым существенно повышает эффективность поиска и надежность получения численного решения в прикладных задачах оптимального управления.

## СПИСОК ЛИТЕРАТУРЫ

1. Тятюшкин А.И. Многометодная технология оптимизации управляемых систем. Новосибирск: Наука, 2006. 343 с.
2. Васильев О.В., Тятюшкин А.И. Об одном методе решения задач оптимального управления, основанном на принципе максимума // Ж. вычисл. матем. и мат. физ. 1981. Т. 21. № 6. С. 1376–1384.
3. Беллман Р., Калаба Р. Квазилинеаризация и нелинейные краевые задачи. М.: Мир, 1968.
4. Тятюшкин А.И. Параллельные вычисления в задачах оптимального управления // Сиб. журн. вычисл. матем. 2000. Т. 3. № 2. С. 181–190.
5. Junkins J.L., Turner J.D. Optimal continuous torque attitude maneuvers // AIAA / AAS Astrodynamics conf. Palo Alto, Calif., 1978.
6. Макаров И.М., Виноградская Т.М., Рубчинский А.А., Соколов В.Б. Теория выбора и принятия решений. М.: Физматлит, 1982. 328 с.
7. Зингер Р. Оценка характеристик оптимального фильтра для слежения за пилотируемой целью // Зарубежная радиоэлектроника. 1971. № 8. С. 40–56.
8. Петров Ю.П. Оптимальное управление электроприводом // Энергия. 1978. № 1. С. 49–50.
9. Тятюшкин А.И., Федунцов Б.Е. Численное исследование свойств оптимального управления в одной задаче преследования // Изв. РАН, ТиСУ. 2005. № 3. С. 104–113.
10. Тятюшкин А.И., Федунцов Б.Е. Возможности защиты от атакующей ракеты задней полусферы самолета вертикальным маневром // Изв. РАН, ТиСУ. 2006. № 1. С. 111–125.