

Общероссийский математический портал

Н. В. Балашевич, Р. Габасов, Ф. М. Кириллова, Построение оптимальных обратных связей по математическим моделям с неопределенностью, *Ж. вычисл. матем. и матем. физ.*, 2004, том 44, номер 2, 265–286

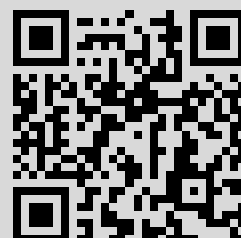
Использование Общероссийского математического портала Math-Net.Ru подразумевает, что вы прочитали и согласны с пользовательским соглашением

<http://www.mathnet.ru/rus/agreement>

Параметры загрузки:

IP: 134.17.36.144

7 марта 2020 г., 14:22:16



УДК 519.626.2

ПОСТРОЕНИЕ ОПТИМАЛЬНЫХ ОБРАТНЫХ СВЯЗЕЙ ПО МАТЕМАТИЧЕСКИМ МОДЕЛЯМ С НЕОПРЕДЕЛЕННОСТЬЮ¹⁾

© 2004 г. Н. В. Балашевич*, Р. Габасов**, Ф. М. Кириллова*

(*220072 Минск, ул. Сурганова, 11, Ин-т матем. НАН Беларуси;

**220080 Минск, пр. Скорины, 4, Белгосуниверситет)

e-mail: balash@im.bas-net.by

Поступила в редакцию 01.10.2002 г.

Рассматривается задача оптимального управления динамическими системами в условиях неопределенности. В отличие от классического подхода, в котором оптимальные обратные связи строятся по детерминированным моделям, в работе используются математические модели с множественной неопределенностью. Допустимые и оптимальные управления гарантируют определенный результат. В зависимости от используемой информации вводится несколько типов обратных связей (размыкаемые, замыкаемые). По мере усложнения обратной связи (ОС) увеличивается ее эффективность. Для каждого типа оптимальной ОС предложены методы ее реализации на современных вычислительных устройствах. Результаты иллюстрируются примерами. Библ. 18. Фиг. 8. Табл. 2.

1. ВВЕДЕНИЕ

Обратные связи (одно из величайших открытий человечества) используются в системах управления уже более 4000 лет (см. [1]). Они представляют собой один из эффективных способов управления динамическими системами в условиях неопределенности, которые характерны практически для всех реальных систем управления. В классическую эпоху теории управления (до 40-х гг. XX в.) использовались в основном линейные ОС. С целью повышения качества систем управления, особенно в связи с первыми задачами оптимального управления (ОУ) (см. [2], [3]), стали систематически конструировать нелинейные ОС. Характерной особенностью классического подхода к синтезу оптимальных ОС являлось то, что ОС строились по детерминированным моделям, хотя предназначались для управления системами в условиях неопределенности. Один из подходов к эффективной реализации оптимальных классических ОС описан в [4].

Инженеры понимали (см. [5]), что адекватный учет любой информации о неизвестных возмущениях может повысить эффективность конструируемых ОС. Однако для реализации такого подхода до 40-х гг. не существовало необходимого математического аппарата. В теории управления информацию о возмущениях стали использовать сначала (см. [6], [7]) в рамках стохастических моделей, опираясь на созданную в те годы теорию случайных процессов [8]. Стохастические модели налагали жесткие ограничения на необходимую информацию и не полностью отражали все особенности современных моделей неопределенности. Со временем появились нестохастические (множественные) модели неопределенности (см. [9], [10]), в которых информация о возмущениях ограничивалась (исчерпывалась) множествами их возможных значений. Новые модели неопределенности позволяли естественным образом исследовать важные в приложениях задачи гарантированного (минимаксного) управления, но они требовали применения нового математического аппарата, связанного с негладким анализом. Такой аппарат был создан в теории оптимального управления (см. [11], [12]) и в дифференциальных играх [13].

Следует, однако, подчеркнуть, что основные результаты новых теорий носили качественный характер, т.е. относились к анализу решений соответствующих задач. Хотя при этом выявлялась структура решений, потенциальные возможности систем управления, но не были построены эффективные методы нахождения решений. Существующие математические методы решения задач ОУ нацелены, как правило, на вычисление программных управлений. Такая форма решений родственна традиционным представлениям вариационного исчисления и других разделов математики. Задачи же управления динамическими системами принципиально отличаются от стан-

¹⁾Работа выполнена при финансовой поддержке Белорусского республиканского фонда фундаментальных исследований (грант Ф02Р-008) и INTAS (грант YSF 2001/2-79).

дартных математических задач тем, что они требуют построения позиционных решений, конструирования управляющих воздействий по ходу процессов управления в зависимости от возникающих ситуаций, которые невозможно предвидеть заранее. В потоке работ по исследованию программных решений эта особенность задач ОУ фактически не учитывалась. Одно общеизвестное исключение касается линейно-квадратичных задач Летова–Калмана и примыкающих к ним задач H_∞ -теории. Но эти задачи очень специфичны и по существу являются задачами классического вариационного исчисления. В них не учитываются прямые (геометрические) ограничения на управления, что позволяет получить решение в виде линейных ОС. Теперь, после пятидесяти лет развития теории ОУ, пора выйти за рамки линейных ОС и приступить к решению задач ОУ в той постановке, которая содержалась в первых инженерных формулировках (см. [2], [3]).

Причины тех трудностей, которые возникают при попытках использования для синтеза оптимальных ОС классических результатов Понтрягина и Беллмана, можно объяснить следующим образом. В основе упомянутых результатов лежат сопряженные уравнения и уравнения Беллмана. Для каждого из этих уравнений “естественным” временем является обратное время, т.е. нормальный процесс их решения осуществляется во времени, которое течет в направлении обратном к направлению реального (фактического) времени, в котором формулируются задачи управления. Для построения программных решений это обстоятельство не играет никакой роли, так как программные решения в реальном процессе управления не используются. Однако если строить управление в реальном времени (в частности, реализовывать оптимальную ОС), то нужно обратить время, участвующее в сопряженном уравнении и в уравнении Беллмана. Классический способ обращения времени – найти явные формулы для ОУ типа ОС для всех возможных позиций. Другой способ – запомнить (затабулировать) значения ОУ для всех возможных позиций. Оба способа утопичны, поскольку нереализуемы современными математическими и вычислительными средствами из-за сложности оптимальных ОС и из-за “проклятия” размерности. Предлагаемый в данной работе подход к синтезу оптимальных ОС можно трактовать как способ обращения обратного времени в процессе реального управления, основанный на непрерывном решении в процессе управления специальных экстремальных задач. Получаемый при этом результат можно назвать оптимальным замыканием системы управления на позициях, которые возникают в процессе реального управления. С этой точки зрения оптимальное программное управление осуществляет замыкание ОУ до начала процесса управления только на начальной позиции, что исключает возможность учета действия возмущений, которые сказываются на последующих состояниях системы. Оптимальные ОС, если их удастся построить с помощью принципа максимума или динамического программирования, замыкают систему до начала процесса управления на всех возможных позициях (в том числе и на основной массе позиций, которые не встретятся в реальном процессе управления).

Цель настоящей работы – обобщить результаты работ [4], [14] по синтезу оптимальных классических ОС на случай, когда математические модели систем управления содержат множественные неопределенности. Подобная задача ранее исследовалась в [15], но используемые там методы трудно перенести на системы высокого порядка. Предлагаемый ниже подход соответствует основной идее, лежащей в основе всех работ авторов по синтезу оптимальных систем. Согласно этой идее, до начала реального процесса управления выполняется предварительная работа по получению “скелета” решения, а затем в процессе управления предварительные результаты непрерывно уточняются в соответствии с получаемой информацией о реализующихся состояниях. Усилия на обработку и хранение результатов обработки других возможных, но не реализовавшихся в данном процессе состояний не тратятся. В конечном счете, такой подход позволяет обойти трудности классического подхода по поиску ОС в замкнутой форме и проклятие размерности динамического программирования не только в детерминированных задачах ОУ, но и в рассматриваемых ниже задачах ОУ, которые намного сложнее задач [4], [14], поскольку являются по существу задачами теории дифференциальных игр.

В нашей работе в разд. 2 дается постановка задачи синтеза оптимальных систем по недетерминированным моделям. Рассматриваемая линейная модель в отличие от классического случая содержит аддитивно входящую неизвестную функцию возмущений. Последняя вводится для учета неточностей математической модели задачи и неизвестных возмущений, возникающих в процессе реального управления. Информация о возмущении ограничивается заданием множества его возможных значений. В этой ситуации естественной оказывается задача оптимального гарантирующего управления, которую можно трактовать как одну из задач теории дифференциальных игр. Описывается метод динамического программирования для построения оптимальной ОС, и показывается, как проявляется проклятие размерности, из-за которого метод оказывается

практически нереализуем. В разд. 3 вводится понятие оптимальной размыкаемой ОС и предлагается реализуемый метод ее синтеза. Возможности оптимальной размыкаемой ОС расширяются в разд. 4, где вводится первая фаза, позволяющая начать процесс управления и в тех случаях, когда оптимальной размыкаемой ОС для начальной позиции не существует. Начиная с разд. 5, исследуются оптимальные замыкаемые ОС. При их построении учитывается информация о том, что ОС замкнута не только на текущей позиции, но будет замкнута и в будущем. Сначала исследуется случай одного момента замыкания. В разд. 7 описывается метод синтеза оптимальной ОС для нескольких моментов замыкания. Оптимальная замыкаемая ОС, моменты замыкания которой состоят из всех моментов, в которых возможно управление, называется оптимальной замкнутой ОС. Ее реализация не связана с табулированием функций многих переменных и поэтому может быть осуществлена на практике с помощью современных вычислительных устройств. Результаты иллюстрируются примерами в разд. 6, 8.

2. ПОСТАНОВКА ЗАДАЧИ

Рассмотрим линейную задачу ОУ

$$h'_{(0)}x(t^*) \rightarrow \max, \quad \dot{x} = A(t)x + b(t)u + d(t)w, \quad x(t_*) = x_0, \\ x(t^*) \in X^* = \{x \in \mathbb{R}^n : h'_{(i)}x \geq g_i, i = \overline{1, m}\}, \quad (2.1)$$

$$|u(t)| \leq 1, \quad w(t) \in W = \{w \in \mathbb{R} : w_* \leq w \leq w^*\}, \quad t \in T = [t_*, t^*],$$

где $x = x(t)$ есть n -вектор состояния динамической системы в момент t , $u = u(t)$ – значение скалярного управления, $w = w(t)$ ($t \in T$) – кусочно-непрерывная функция возмущения.

Функцию

$$u = u(t, x), \quad x \in X_t, \quad t \in T, \quad (2.2)$$

с областями определения X_t ($t \in T$) назовем ОС в задаче (2.1).

Замкнем ею систему управления (2.1):

$$\dot{x} = A(t)x + b(t)u(t, x) + d(t)w, \quad x(t_*) = x_0. \quad (2.3)$$

Будем считать, что каждому возможному возмущению $w(\cdot) = (w(t), t \in T)$ соответствует единственное решение $x(t | w(\cdot))$ ($t \in T$) уравнения (2.3). Пусть $X(t) = \{x : x = x(t | w(\cdot)), w(s) \in W, t_* \leq s \leq t\}$, $w_t(\cdot) = (w(s), t_* \leq s \leq t)$.

ОС (2.2) назовем допустимой, если

$$|u(t, x)| \leq 1, \quad x \in X_t, \quad t \in T,$$

и система (2.3) при любом возможном возмущении попадает в момент $t = t^*$ на терминальное множество $X^* : X(t^*) \subset X^*$.

Качество ОС (2.2) оценим числом

$$J(u) = \min h'_{(0)}x, \quad x \in X(t^*).$$

Допустимая ОС $u^0 = u^0(t, x)$ ($x \in X_t, t \in T$) называется оптимальной, если $J(u^0) = \max J(u)$, где максимум вычисляется по всем допустимым ОС.

Универсальным методом построения оптимальных ОС является динамическое программирование [12]. Чтобы не иметь дело с вопросами обоснования динамического программирования для оптимальных ОС общего вида (2.2), введем понятие дискретной ОС, которая отличается от (2.2) только способом воздействия на динамическую систему (2.3). Пусть $h = (t^* - t_*)/N$ ($N > 0$) – период квантования времени. ОС (2.2) называется дискретной (с периодом квантования h), если траектория замкнутой ею системы (2.3) совпадает с траекторией линейного уравнения

$$\dot{x} = A(t)x + b(t)u(t) + d(t)w, \quad x(t_*) = x_0,$$

с управлением $u(t) = u(t_* + kh, x(t_* + kh))$, $t \in [t_* + kh, t_* + (k+1)h]$, $k = \overline{0, N-1}$.

Понятно, что в случае дискретной ОС отпадают вопросы определения и существования решения уравнения (2.3) с разрывной по x правой частью.

Дискретным ОС соответствуют дискретные (программные) управления

$$u(t) = u(t_* + kh), \quad t \in [t_* + kh, t_* + (k+1)h[, \quad k = \overline{0, N-1}. \quad (2.4)$$

Задачу (2.1) в классе дискретных программных управлений погрузим в семейство

$$B(\tau, z) = \max_u \min_w h'_{(0)} x(t^*), \quad \dot{x} = A(t)x + b(t)u + d(t)w, \quad x(\tau) = z, \\ x(t^*) \in X^*, \quad |u(t)| \leq 1, \quad w(t) \in W, \quad t \in T(\tau) = [\tau, t^*], \quad (2.5)$$

зависящее от скаляра $\tau \in T_h = \{t_*, t_* + h, \dots, t^* - h\}$ и n -вектора $z \in X_\tau$.

Используя стандартные рассуждения динамического программирования, для функции Беллмана $B(\tau, z)$ ($z \in X_\tau, \tau \in T_h$) получаем рекуррентное уравнение

$$B(\tau, z) = \max_{|v| \leq 1} \min_{w(s) \in W, s \in [\tau, \tau+h[} B(\tau+h, x(\tau+h | \tau, z, v, w_h(\tau))), \quad (2.6)$$

где $x(\tau+h | \tau, z, v, w_h(\tau))$ – состояние системы (2.5), в которое она попадет в момент $\tau+h$ из состояния z в момент τ под действием управления $u(t) = v$ ($t \in [\tau, \tau+h[$) и возмущения $w_h(\tau) = (w(t), t \in [\tau, \tau+h[)$.

Начальное условие для рекуррентного уравнения (2.6) имеет вид

$$B(t^*, z) = \begin{cases} h'_{(0)} z, & z \in X^*, \\ -\infty, & z \notin X^*. \end{cases}$$

Уравнение (2.6) может быть “в принципе” решено по шагам справа налево, и попутно будет построена оптимальная дискретная ОС $u^0(t, x)$, $x \in X_t, t \in T_h$ ($X_t = \{x \in \mathbb{R}^n : B(\tau, x) > -\infty\}$) из соотношения

$$\min_{w(s) \in W, s \in [\tau, \tau+h[} B(\tau+h, x(\tau+h | \tau, z, u^0(\tau, z), w_h(\tau))) = \\ = \max_{|u| \leq 1} \min_{w(s) \in W, s \in [\tau, \tau+h[} B(\tau+h, x(\tau+h | \tau, z, u, w_h(\tau))).$$

Приведенная красивая схема решения задачи (2.1) методом динамического программирования не реализуема из-за проклятия размерности (см. [12]), т.е. из-за невозможности табулирования функции Беллмана и ОС для задач (2.1) при больших n .

В связи с этим в [15] предложены другие типы оптимальной ОС, которые проще оптимальной ОС, получаемой методом динамического программирования (но могут быть и менее эффективны). Впервые упрощенные ОС для сетевой экстремальной задачи со стохастической моделью неопределенности были описаны в [16]. Цель настоящей работы – изложить новый (по сравнению с [15]) метод реализации управлений типа ОС для задачи (2.1), который позволяет реализовать и оптимальную ОС, получаемую методом динамического программирования.

3. ОПТИМАЛЬНАЯ РАЗМЫКАЕМАЯ ОБРАТНАЯ СВЯЗЬ

Рассмотрим вновь задачу (2.1) в классе дискретных управлений (2.4). Погрузим ее в семейство задач (2.5). Обозначим через $u^0(t | \tau, z)$ ($t \in T(\tau)$) программное решение для позиции (τ, z) , X_τ – множество $z \in \mathbb{R}^n$, для которых задача (2.5) имеет решение.

Функцию

$$u^0(\tau, z) = u^0(\tau | \tau, z), \quad z \in X_\tau, \quad \tau \in T_h, \quad (3.1)$$

назовем оптимальной размыкаемой дискретной ОС.

Прежде чем описать алгоритм реализации ОС (3.1), рассмотрим задачу построения оптимального программного управления (ОПУ) $u^0(t | \tau, z)$ ($t \in T, (\tau, z)$).

Дискретное управление $u(t)$ ($|u(t)| \leq 1, t \in T(\tau)$) допустимо тогда и только тогда, когда при всех $w(t) \in W$ ($t \in T(\tau)$) соответствующие траектории

$$x(t) = F(t)F^{-1}(\tau)z + \int_{\tau}^t F(t)F^{-1}(s)[b(s)u(s) + d(s)w(s)]ds,$$

$$\dot{F} = A(t)F, \quad F(t_*) = E,$$

попадают в момент t^* на терминальное множество X^* , т.е.

$$h'_{(i)}x(t^*) \geq g_i, \quad i = \overline{1, m}.$$

Понятно, что это имеет место, если

$$h'_{(i)}F(t^*)F^{-1}(\tau)z + \int_{\tau}^{t^*} h'_{(i)}F(t^*)F^{-1}(s)b(s)u(s)ds \geq g_i - \gamma_i(\tau), \quad (3.2)$$

где

$$\gamma_i(\tau) = \min_{w(t) \in W, t \in T(\tau)} \int_{\tau}^{t^*} h'_{(i)}F(t^*)F^{-1}(s)d(s)w(s)ds.$$

Таким образом, соотношения (3.2), $|u(t)| \leq 1$ ($t \in T(\tau)$) описывают допустимые дискретные управления $u(t)$ ($t \in T(\tau)$).

По аналогии с (3.2) подсчитаем качество допустимого управления:

$$\begin{aligned} J(u) = & h'_{(0)}F(t^*)F^{-1}(\tau)z + \int_{\tau}^{t^*} h'_{(0)}F(t^*)F^{-1}(s)b(s)u(s)ds + \\ & + \min_{w(t) \in W, t \in T(\tau)} \int_{\tau}^{t^*} h'_{(0)}F(t^*)F^{-1}(s)d(s)w(s)ds. \end{aligned} \quad (3.3)$$

Отсюда следует, что ОПУ задачи (2.5) в условиях неопределенности является решением экстремальной детерминированной задачи

$$\begin{aligned} & \int_{\tau}^{t^*} h'_{(0)}F(t^*)F^{-1}(s)b(s)u(s)ds \rightarrow \max, \\ & h'_{(i)}F(t^*)F^{-1}(\tau)z + \int_{\tau}^{t^*} h'_{(i)}F(t^*)F^{-1}(s)b(s)u(s)ds \geq g_i - \gamma_i(\tau), \quad i = \overline{1, m}, \\ & |u(t)| \leq 1, \quad t \in T(\tau). \end{aligned} \quad (3.4)$$

Динамическая форма задачи (3.4) имеет вид

$$\begin{aligned} & h'_{(0)}x(t^*) \rightarrow \max, \quad \dot{x} = A(t)x + b(t)u, \quad x(\tau) = 0, \\ & x(t^*) \in \bar{X}^*(\tau, z) = \{x \in \mathbb{R}^n : h'_{(i)}x \geq \bar{g}_i(\tau), i = \overline{1, m}\}, \quad |u(t)| \leq 1, \quad t \in T(\tau), \end{aligned} \quad (3.5)$$

где $\bar{g}_i(\tau) = g_i - \gamma_i(\tau) - h'_{(i)}F(t^*)F^{-1}(\tau)z$, $i = \overline{1, m}$.

Алгоритм построения программных решений $u^0(t | \tau, z)$ ($t \in T(\tau)$) в задачах типа (3.5) описан в [14]. Он представляет собой динамическую реализацию адаптивного метода линейного программирования (ЛП, см. [17]), учитывающую особенности задачи ОУ.

Для введения понятия реализации оптимальной ОС (2.2) проанализируем поведение замкнутой системы (2.3) в некотором конкретном процессе управления.

Пусть в рассматриваемом процессе реализуются некоторое начальное состояние x_0^* и возмущение $w^*(t)$ ($t \in T$). Они породят в системе (2.3) переходный процесс $x^*(t)$ ($t \in T$), вдоль которого выполняется тождество

$$\dot{x}^*(t) \equiv A(t)x^*(t) + b(t)u^0(t, x^*(t)) + dw^*(t), \quad t \in T, \quad x(t_*) = x_0^*. \quad (3.6)$$

Из (3.6) видно, что в конкретном процессе управления ОС (2.2) целиком не используется, нужны

лишь ее значения

$$u^*(t) = u^0(t, x^*(t)), \quad t \in T_h, \quad (3.7)$$

вдоль изолированной непрерывной кривой $x^*(t)$ ($t \in T$). При этом достаточно в каждый текущий момент $\tau \in T_h$, зная текущее состояние $x^*(\tau)$, уметь вычислять текущее значение $u^*(\tau) = u^0(\tau, x^*(\tau))$ за время, не превосходящее h , т.е. в режиме реального времени.

Функцию (3.7) назовем реализацией оптимальной ОС в конкретном процессе управления, а устройство, способное в каждом конкретном процессе управления (3.6) вычислять ее значения в режиме реального времени, – оптимальным регулятором (ОР), который работает по следующему алгоритму. До начального момента $\tau = t_*$ он решает задачу (3.5) для $\tau = t_*$, $z = x_0^*$ и на промежутке времени $[t_*, t_* + h[$ подает на вход системы (2.3) сигнал $u^*(t_*) = u^0(t_* | t_*, x_0^*)$. Поскольку вычисления выполняются до начала процесса фактического управления системой, то можно считать, что имеется достаточно времени для построения программного решения задачи (3.5) и формирования вспомогательной информации согласно алгоритму из [14].

Затем в каждый текущий момент $\tau \in T_h$ процесса управления регулятор строит программное решение задачи (3.5) для позиции $(\tau, x^*(\tau))$, используя в качестве начального приближения для сопровождающих элементов [14] оптимального управления $u^0(t | \tau, x^*(\tau))$ ($t \in T(\tau)$) оптимальные элементы, построенные в предыдущий момент $\tau - h$. В [14] приведены оценки трудоемкости метода, позволяющие сформулировать требования к быстродействию микропроцессоров, реализующих ОР, чтобы время вычисления сигнала $u^*(\tau)$ для оптимизируемой системы не превосходило h единиц реального времени. Эти требования к используемым вычислительным устройствам можно существенно ослабить, если оптимальную ОС (3.1) реализовать с помощью нескольких вычислительных устройств заданной (не обязательно очень большой) мощности. Хотя при этом возникают задержки с передачей сигналов, но процесс управления можно организовать так, чтобы влияние задержек было незначительным (см. [18]).

4. ДВУХФАЗНАЯ ОПТИМАЛЬНАЯ РАЗМЫКАЕМАЯ ОБРАТНАЯ СВЯЗЬ

Как следует из определения оптимальной размыкаемой ОС (3.1), ее реализация $u^*(\tau)$ ($\tau \in T$) в конкретном процессе управления может быть построена, если начальное состояние системы удовлетворяет условию $x_0^* \in X_{t_*}$.

При нарушении этого условия нельзя априорно гарантировать выполнение терминальных ограничений задачи (2.1) при любом возможном возмущении. В этом случае целесообразно на начальном этапе (первая фаза) организовать управление системой таким образом, чтобы к некоторому моменту $\bar{t} \in T_h$ привести траекторию системы на множество $X_{\bar{t}}$, а затем перейти к построению реализации оптимальной ОС путем решения задачи (3.5) (вторая фаза). Такая схема управления реализует двухфазную оптимальную размыкаемую ОС.

Задачей первой фазы назовем задачу попадания терминального состояния системы (2.5) в минимальную γ -окрестность терминального множества X^* :

$$\begin{aligned} \gamma^0(\tau, z) &= \min \gamma, \quad \dot{x} = A(t)x + b(t)u, \quad x(\tau) = 0, \\ h'_{(i)}x(t^*) &\geq \bar{g}_i(\tau) - \gamma, \quad i = \overline{1, m}, \quad |u(t)| \leq 1, \quad t \in T(\tau). \end{aligned} \quad (4.1)$$

Оптимальное программное управление $u_\gamma^0(t | \tau, z)$ ($t \in T(\tau)$) задачи (4.1) существует для любых $z \in \mathbb{R}^n$.

Двухфазная оптимальная размыкаемая ОС определяется соотношением

$$u^0(\tau, z) = \begin{cases} u_\gamma^0(\tau | \tau, z), & z \notin X_\tau, \\ u^0(\tau | \tau, z), & z \in X_\tau, \quad \tau \in T_h. \end{cases}$$

Условие $z \in X_\tau$ эквивалентно неравенству $\gamma^0(\tau, z) \leq 0$. Таким образом, если задача (3.5) не имеет решения для позиции (t_*, x_0^*) , то вдоль реализующейся траектории $x^*(\tau)$ ($\tau \geq t_*$) регулятор строит

решения задачи (4.1) до тех пор, пока $\gamma^0(\tau, x^*(\tau)) > 0$, и подает на вход системы сигнал $u^*(\tau) = u_\gamma^0(\tau | \tau, x^*(\tau))$.

Если неравенство $\gamma^0(\tau, x^*(\tau)) > 0$ выполняется на всем T_h , то траектория системы в момент t^* достигнет $\gamma^0(t^* - h, x^*(t^* - h))$ -окрестности множества X^* .

В случае, когда существует такой момент $\bar{t} \leq t^* - h$, что $\gamma^0(\bar{t}, x^*(\bar{t})) \leq 0$, регулятор переходит с момента \bar{t} ко второй фазе управления, на которой реализация $u^*(\tau)$ ($\tau \geq \bar{t}$) строится с помощью решения задачи (3.5).

Для того чтобы осуществить переход от первой фазы ко второй (выработать сигнал $u^*(\bar{t})$) в режиме реального времени, или, другими словами, избежать задержек при переходе от решения задачи (4.1) к решению задачи (3.5), достаточно к моменту \bar{t} заготовить приближение к решению задачи (3.5). Для этого в каждый момент $\tau = t_*, \dots, \bar{t} - h$ после вычисления для текущей позиции $(\tau, x^*(\tau))$ значения $\gamma^0(\tau, x^*(\tau))$ подставляем эти данные в задачу

$$\begin{aligned} h'_{(i)}x(t^*) &\rightarrow \max, \quad \dot{x} = A(t)x + b(t)u, \quad x(\tau) = 0, \\ h'_{(i)}x(t^*) &\geq \bar{g}_i(\tau) - \gamma^0(\tau, z) - \varepsilon, \quad |u(t)| \leq 1, \quad t \in T(\tau). \end{aligned} \quad (4.2)$$

Здесь $\varepsilon > 0$ – малый параметр, обеспечивающий, с одной стороны, неединственность допустимого управления в задаче (4.2), а с другой стороны, близость ОПУ задач (4.2) и (3.5) для позиции $(\bar{t}, x^*(\bar{t}))$. Параметр ε подбираем таким образом, чтобы, имея сопровождающие элементы оптимального управления задачи (4.2) для позиции $(\bar{t} - h, x^*(\bar{t} - h))$, построить двойственным методом [14] оптимальное управление задачи (3.5) для позиции $(\bar{t}, x^*(\bar{t}))$ за время, не превосходящее h единиц реального времени.

5. ОПТИМАЛЬНАЯ ЗАМЫКАЕМАЯ ОБРАТНАЯ СВЯЗЬ С ОДНИМ МОМЕНТОМ ЗАМЫКАНИЯ

При реализации размыкаемой ОС (разд. 3) система управления (2.1) замыкалась в каждый текущий момент времени $\tau \in T_h$, но не использовалась информация о том, что она может быть замкнута в какие-то будущие моменты времени. В случае детерминированных моделей (см. [4], [14]) использование этой дополнительной информации не улучшает (классическую) оптимальную ОС. Однако если при построении оптимальной ОС опираться на математические модели с неопределенностью, то дополнительная информация о замыкании системы управления в будущем может существенным образом повысить эффективность ОС.

Начнем с простейшего случая. Будем считать, что еще до начала процесса управления известно, что система управления будет замкнута в момент времени $t^1 \in T_h$. Это означает, что для обеспечения выполнения терминальных ограничений траектория системы должна оказаться в момент t^1 на множестве X^1 , для каждой точки которого существует свое доступное управление, переводящее систему при любом возмущении на множество X^* в момент t^* . Считаем, что доступные технические средства позволяют обрабатывать эту информацию наряду с текущей информацией, получаемой в процессе управления. Как и в разд. 3, предположим, что в каждый текущий момент $\tau \in T_h$ доступно текущее состояние $x^*(\tau)$.

5.1. Оптимальное программное управление

Сначала опишем алгоритм построения ОПУ в задаче (2.5) с одним моментом замыкания. Для этого введем необходимые вспомогательные конструкции.

Выберем момент $t^1 \in T_h$ ($t_* < t^1 < t^* - h$) и будем считать, что $\tau \in T_h^0 = \{t_*, \dots, t^1 - h\}$ в задаче (2.5). Используя момент замыкания t^1 , определим множество замыкания X^1 , включив в него все начальные состояния x в момент t^1 , которые с гарантией можно перевести в момент t^* на терминальное множество X^* .

Вычислим

$$\alpha_{\min} = \min_{x \in X^*} h'_{(0)}x, \quad \alpha_{\max} = \max_{x \in X^*} h'_{(0)}x.$$

Для любого $\alpha \in [\alpha_{\min}, \alpha_{\max}]$ определим множество

$$X_{\alpha}^* = X^* \cap \{x \in \mathbb{R}^n : h'_{(0)}x \geq \alpha\}.$$

Дискретное управление $u(t)$ ($t \in T^1 = [t^1, t^*]$) назовем доступным на промежутке T^1 , если оно удовлетворяет ограничению $|u(t)| \leq 1$ ($t \in T^1$).

Множество $X_{\alpha}^1 \subset \mathbb{R}^n$ назовем α -множеством замыкания в момент t^1 , если для любого $x \in X_{\alpha}^1$ существует доступное на промежутке T^1 управление $u^{\alpha}(t | t^1, x, 1)$ ($t \in T^1$), переводящее при любой реализации возмущения $w(t) \in W$ ($t \in T^1$) траекторию системы (2.5) из состояния $x(t^1) = x$ на множество X_{α}^* в момент t^* . Управление $u^{\alpha}(t | t^1, x, 1)$ ($t \in T^1$) будем называть α -допустимым управлением для позиции (t^1, x) .

Чтобы определить множество X_{α}^1 , используя детерминированную модель

$$\dot{x} = A(t)x + b(t)u, \quad (5.1)$$

введем сужение множества X_{α}^* , в котором учитывается наихудшее возможное влияние возмущений на промежутке T^1 :

$$\bar{X}_{\alpha}^* = \{x \in \mathbb{R}^n : h'_{(i)}x \geq g_i - \gamma_i(t^1), i = \overline{1, m}; h'_{(0)}x \geq \alpha - \gamma_0(t^1)\}.$$

Теперь множество X_{α}^1 определим как множество состояний $x \in \mathbb{R}^n$, для каждого из которых существует доступное на промежутке T^1 управление, переводящее траекторию системы (5.1) из состояния $x(t^1) = x$ на множество \bar{X}_{α}^* в момент t^* .

Таким образом, согласно определению, если траектория системы (2.5) из начального состояния $x(\tau) = z$ попадает в момент t^1 на множество X_{α}^1 , то выбором подходящего управления на промежутке T^1 можно обеспечить гарантированное значение критерия качества задачи (2.5), не меньшее α .

Доступное управление $u^{\alpha}(t | \tau, z, 0)$ ($t \in T^0(\tau) = [\tau, t^1]$) назовем α -допустимым для позиции (τ, z) , если при любых реализациях возмущения $w(t) \in W$, $t \in T^0(\tau)$, оно переводит траекторию системы (2.5) из состояния $x(\tau) = z$ на множество X_{α}^1 в момент t^1 .

Максимальное значение α^0 , для которого существуют множество $X_{\alpha^0}^1$ и α^0 -допустимое управление $u^{\alpha^0}(t | \tau, z, 0)$ ($t \in T^0(\tau)$), является оптимальным гарантированным значением критерия качества задачи (2.5) для момента замыкания t^1 .

Множество состояний, в которых может оказаться система (2.5) под действием управления $u^{\alpha}(t | \tau, z, 0)$ ($t \in T^0(\tau)$) и любых реализаций возмущения $w(t) \in W$ ($t \in T^0(\tau)$), обозначим через $X_{\alpha}^1(\tau, z)$. ОПУ задачи (2.5) с моментом замыкания t^1 представляет собой α^0 -допустимое управление $u^{\alpha^0}(t | \tau, z, 0)$ ($t \in T^0(\tau)$) и семейство α^0 -допустимых управлений $u^{\alpha^0}(t | t^1, x, 1)$ ($t \in T^1, x \in X_{\alpha^0}^1(\tau, z)$).

Опишем алгоритм вычисления α^0 и построения управления $u^{\alpha^0}(t | \tau, z, 0)$ ($t \in T^0(\tau)$).

Поскольку в общем случае множества X_{α}^1 ($\alpha \in [\alpha_{\min}, \alpha_{\max}]$) имеют сложную структуру, то не будем тратить усилия на точное описание этих множеств. Учитывая прикладную направленность задачи, опишем метод, позволяющий получить решение с заданной точностью. Для этого выберем некоторое начальное $\alpha \in [\alpha_{\min}, \alpha_{\max}]$, по $m + 1$ граням множества X_{α}^* найдем нормальные векторы к $m + 1$ "каркасным" граням многогранника $X_{\alpha}^{1,1}$, аппроксимирующего множество X_{α}^1 , и запишем уравнения граней этого многогранника. Затем построим последовательность значений $\alpha_1, \dots, \alpha_{k^*}$, сходящуюся к α^0 , и последовательность многогранников $X_{\alpha_k}^{1,k}$, $k = \overline{1, k^*}$, уточня-

ющих структуру множества $X_{\alpha_k}^1$ в окрестности множества $X_{\alpha_k}^1(\tau, z)$ возможных состояний системы (2.5) в момент t^1 .

Полагаем $k = 1, M(k) = m$.

Итерация алгоритма включает две процедуры: 1) вычисление значения α_k и соответствующего этому значению субоптимального управления на промежутке $T^0(\tau)$ для заданного количества $M(k) + 1$ граней аппроксимирующего многогранника $X_{\alpha}^{1,k}$; 2) увеличение числа граней аппроксимирующего многогранника и нахождение нормальных векторов к новым граням.

Первая процедура начинается с построения аппроксимирующего многогранника $X_{\alpha}^{1,k}$ при некотором значении $\alpha \in [\alpha_{\min}, \alpha_{\max}]$. Он строится с помощью решения $M(k) + 1$ задач ОУ

$$\begin{aligned} \beta_i^{1\alpha} &= \min_{y, u} h_{(i)}^{1'} y, \quad \dot{x} = A(t)x + b(t)u, \quad x(t^1) = y, \\ x(t^*) &\in \bar{X}_{\alpha}^*, \quad |u(t)| \leq 1, \quad t \in T^1, \quad i = \overline{0, M(k)}, \end{aligned} \quad (5.2)$$

где $h_{(i)}^{1'} = h_{(i)}' F(t^* - t^1) / \|F'(t^* - t^1) h_{(i)}\|_2$ ($i = \overline{0, m}$), в результате получаем множество

$$X_{\alpha}^{1,k} = \{x \in \mathbb{R}^n : h_{(i)}^{1'} x \geq \beta_i^{1\alpha}, i = \overline{0, M(k)}\}.$$

Управление $u^{\alpha, k}(t | \tau, z, 0)$ ($t \in T^0(\tau)$) назовем αk -субдопустимым для позиции (τ, z) , если $X_{\alpha}^1(\tau, z) \subset X_{\alpha}^{1,k}$, а αk -субдопустимое управление $u^{\alpha, k}(t | \tau, z, 0)$ ($t \in T^0(\tau)$) с максимальным значением $\alpha = \alpha_k$ назовем k -субоптимальным управлением задачи (2.5) для позиции (τ, z) .

Чтобы перейти к детерминированной системе, введем сужение $\bar{X}_{\alpha}^{1,k}$ множества $X_{\alpha}^{1,k}$ как

$$\bar{X}_{\alpha}^{1,k} = \{x \in \mathbb{R}^n : h_{(i)}^{1'} x \geq \bar{\beta}_i^{1\alpha}, i = \overline{0, M(k)}\},$$

где

$$\bar{\beta}_i^{1\alpha} = \beta_i^{1\alpha} - \min_{w(t) \in W, t \in T^0(\tau)} \int_{\tau}^{t^1} h_{(i)}^{1'} F(t^1 - s) d(s) w(s) ds.$$

Задача отыскания k -субоптимального управления $u^{\alpha, k}(t | \tau, z, 0)$ ($t \in T^0(\tau)$) представляет собой элемент семейства задач ОУ

$$\begin{aligned} \rho_{\alpha} &= \min_p, \quad \dot{x} = A(t)x + b(t)u, \quad x(\tau) = z, \\ x(t^1) &\in \bar{X}_{\alpha}^{1,k}, \quad |u(t)| \leq \rho, \quad t \in T^0(\tau), \end{aligned} \quad (5.3)$$

зависящих от параметра $\alpha \in [\alpha_{\min}, \alpha_{\max}]$, при $\alpha = \alpha_k$.

Допустимым управлением $u_{\alpha}(t)$ ($t \in T^0(\tau)$) в задаче (5.3) при фиксированном α является дискретное управление, обеспечивающее выполнение терминального ограничения $x(t^1) \in \bar{X}_{\alpha}^{1,k}$, а оптимальное управление $u_{\alpha}^0(t)$ ($t \in T^0(\tau)$) помимо этого доставляет минимальное значение критерию качества.

ОУ $u_{\alpha_k}^0(t)$ ($t \in T^0(\tau)$) задачи (5.3), соответствующее значению $\alpha = \alpha_k$, при котором $\rho_{\alpha_k} = 1$, является k -субоптимальным управлением $u^{\alpha, k}(t | \tau, z, 0)$ ($t \in T^0(\tau)$) задачи (2.5).

Таким образом, алгоритм построения k -субоптимального управления задачи (2.5) состоит в следующем.

Алгоритм 1

Шаг 1. Полагаем $\bar{\rho} = 0$.

Шаг 2. Решаем $M(k) + 1$ задачу (5.2).

Шаг 3. Формируем множество $\bar{X}_\alpha^{1,k}$.

Шаг 4. Если $\bar{X}_\alpha^{1,k} = \emptyset$, то проверяем, можно ли, уменьшая α , получить $\bar{X}_\alpha^{1,k} \neq \emptyset$. Если да, полагаем $\alpha_{\max} = \alpha$, уменьшаем α , переходим к шагу 2. Если нет, то в задаче (2.5) не существует гарантированного управления для момента замыкания t^1 .

Шаг 5. Решаем задачу (5.3), находим ρ_α .

Шаг 6. Если $\alpha = \alpha_{\max}$ и $\rho_\alpha \leq 1$ или $1 - \varepsilon \leq \rho_\alpha \leq 1$ (ε – заданная точность), то решение завершаем, полагаем $\alpha_k = \alpha$, и $u^{\alpha_k, k}(t | \tau, z, 0) = u_{\alpha_k}^0(t) (t \in T^0(\tau))$.

В случае $\rho_\alpha = \bar{\rho} > 1$ задача (2.5) не имеет допустимых управлений.

При $\rho_\alpha > 1$ уменьшаем α . Если $\rho_\alpha < 1 - \varepsilon$, то увеличиваем α . Полагаем $\bar{\rho} = \rho_\alpha$, переходим к шагу 2.

Приращение α уменьшаем при приближении значения ρ_α к 1.

Задачи (5.2) решаются параллельно при помощи несложной модификации быстрого алгоритма решения линейной нестационарной задачи ОУ, описанного в [14]. Поскольку при каждом следующем обращении к процедуре решения задач (5.2) значение α меняется незначительно, то числа $\beta_i^\alpha, i = \overline{0, M(k)}$, также меняются незначительно и сохранение вспомогательной информации [14], соответствующей оптимальному решению для текущего значения α , позволяет очень быстро построить оптимальное решение для следующего значения α .

Для решения задачи (5.3) также используется быстрый алгоритм (см. [14]). Задача (5.3) приводится к виду, исследованному в [14], с помощью замены $\xi_0 = 1/\rho, \xi(t) = u(t)/\rho (t \in T^0(\tau))$. Опять же использование вспомогательной информации, сохраненной для текущего значения α , позволяет существенно сократить время на решение задачи (5.3) при следующем значении α .

Пусть найдены значение α_k и k -субоптимальное управление $u^{\alpha_k, k}(t | \tau, z, 0) (t \in T^0(\tau))$. Перейдем ко второй процедуре алгоритма.

Введем следующие обозначения: $x^{\alpha_k}(t^1)$ – точка, в которую в момент t^1 попадает траектория системы (5.1) под действием управления $u^{\alpha_k, k}(t | \tau, z, 0) (t \in T^0(\tau))$:

$$x^{\alpha_k}(t^1) = F(t^1)F^{-1}(\tau)z + \int_{\tau}^{t^1} F(t^1)F^{-1}(s)bu^{\alpha_k, k}(s | \tau, z, 0)ds, \quad t \in T^0(\tau),$$

I_a – множество активных ограничений задачи (5.3) на управлении $u^{\alpha_k, k}(t | \tau, z, 0) (t \in T^0(\tau))$:

$$I_a = \{i \in \{0, 1, \dots, M(k)\} : h_{(i)}^{1, \alpha_k}(t^1) = \bar{\beta}_i^{\alpha_k}\},$$

$v_i^0 (i \in I_a)$ – множители Лагранжа, найденные в результате решения задачи (5.3) (см. [14], $v_i^0 = 0, i \in \{0, 1, \dots, M(k)\} \setminus I_a$).

Случай 1. Если $|I_a| > 1$, то положим $M(k+1) = M(k) + 1$ и систему нормальных векторов, определяющих многогранник $X_{\alpha_k}^{1,k}$, дополним вектором

$$h_{(M(k)+1)}^1 = - \frac{\sum_{i \in I_a} v_i^0 h_{(i)}^1}{\left\| \sum_{i \in I_a} v_i^0 h_{(i)}^1 \right\|_2}. \quad (5.4)$$

Случай 2. Если $I_a = \{i_0\} (i_0 > m)$, то найдем точку $z_{i_0}^0$ касания множества замыкания $X_{\alpha_k}^1$ и многогранника $X_{\alpha_k}^{1,k}$. В этой точке достигается минимум критерия качества задачи (5.2) при $\alpha = \alpha_k, i = i_0$. Построим вектор $s = z_{i_0}^0 - x^{\alpha_k}(t^1)$ и найдем векторы

$$h_{(M(k)+i)}^1 = \left(h_{(i_0)}^1 + \frac{i}{p}s \right) \left(\left\| h_{(i_0)}^1 + \frac{i}{p}s \right\|_2 \right)^{-1}, \quad i = \overline{1, p}$$

(p – параметр метода, в численных экспериментах полагался равным 2). Положим $M(k+1) = M(k) + p$.

Случай 3. Если $I_a = \{i_0\}$ ($i_0 < m$), то построим векторы

$$h_{(M(k)+i)}^1 = \frac{h_{(i_0)}^1 + \delta h_{(l)}^1}{\|h_{(i_0)}^1 + \delta h_{(l)}^1\|_2}, \quad i = \overline{1, m}, \quad l = \begin{cases} i-1, & i < i_0, \\ i, & i > i_0, \end{cases} \quad (5.5)$$

где $\delta > 0$ – малое число. Положим $M(k+1) = M(k) + m$.

Положим $k := k+1$, выполним первую процедуру описанного алгоритма, начиная с ранее найденного значения $\alpha = \alpha_{k-1}$. При этом для новых задач (5.2) в качестве начального приближения сопровождающих элементов оптимального управления в случаях 2 и 3 берем оптимальные сопровождающие элементы i_0 -й задачи (5.2), построенные для α_{k-1} . В случае 1 простейший способ построения начального приближения сопровождающих элементов заключается в нахождении такого $i_0 \in I_a$, что $|v_{i_0}^0| = \max_{i \in I_a} |v_i^0|$, и в использовании оптимальных сопровождающих элементов i_0 -й задачи (5.2).

Если окажется, что для вновь найденного α_k выполняются неравенства

$$\alpha_{k-1} - \alpha_k \leq \varepsilon, \quad \|x^{\alpha_k}(t^1) - x^{\alpha_{k-1}}(t^1)\| \leq \varepsilon, \quad (5.6)$$

где $\varepsilon \geq 0$ – заданная точность, то решение завершаем и полагаем $\alpha^0 = \alpha_k$, $u^{\alpha^0}(t | \tau, z, 0) = u^{\alpha_k}(t | \tau, z, 0)$ ($t \in T^0(\tau)$).

Если же $\alpha_{k-1} - \alpha_k > \varepsilon$, то выполняем вторую процедуру.

Замечание 1. В ходе итераций алгоритма может оказаться целесообразным избавляться от промежуточных вспомогательных векторов $h_{(i)}^1$ ($i > m$), чтобы ограничить рост количества пассивных ограничений задачи (5.3). Для этого выбирается такое число $M_0 > m$, что при $M(k) > M_0$ из системы векторов $h_{(i)}^1, i = \overline{0, M(k)}$, удаляются некоторые векторы с номерами из множества $\{m+1, \dots, M(k)\} \setminus I_a$, что приводит к уменьшению числа задач (5.2) и количества ограничений в задаче (5.3).

5.2. Реализация оптимальной замыкаемой обратной связи

Обозначим через X_τ^α множество всех $z \in \mathbb{R}^n$, для которых в задаче (2.5) существуют оптимальные программные управления с моментом замыкания t^1 .

Функцию

$$u^0(\tau, z) = \begin{cases} u^{\alpha^0}(\tau | \tau, z, 0), & z \in X_\tau^\alpha, \quad \tau \in T_h^0(t_*), \\ u^0(\tau | \tau, z), & z \in X_\tau, \quad \tau \in T_h^1, \end{cases} \quad (5.7)$$

назовем оптимальной замыкаемой ОС с моментом замыкания t^1 .

Как и в разд. 3, функцию $u^*(\tau) = u^*(\tau, x^*(\tau))$ ($\tau \in T_h$) будем называть реализацией оптимальной ОС (5.7) в конкретном процессе управления.

До момента $\tau = t_*$ ОР строит управление $u^{\alpha^0}(t | t_*, x_0^*, 0)$ ($t \in T^0(t_*)$) согласно алгоритму и на промежутке $[t_*, t_* + h[$ подает на вход системы (2.3) сигнал $u^*(t_*) = u^{\alpha^0}(t_* | t_*, x_0^*, 0)$. В произвольный момент $\tau \in T_h^0$ регулятор строит $u^{\alpha^0}(t | \tau, x^*(\tau), 0)$ ($t \in T^0(\tau)$) также с помощью алгоритма. При этом в качестве начального значения α берется оптимальное значение α^0 , полученное в момент $\tau - h$, а в качестве начальных приближений сопровождающих элементов оптимальных управлений задач (5.2), (5.3) берутся оптимальные сопровождающие элементы, построенные в момент $\tau - h$. Построив управление $u^{\alpha^0}(t | \tau, x^*(\tau), 0)$ ($t \in T^0(\tau)$), получаем, что регулятор на промежутке $[\tau, \tau + h[$ подает на вход системы (2.3) сигнал $u^*(\tau) = u^{\alpha^0}(\tau | \tau, x^*(\tau), 0)$.

Начиная с момента t^1 , для выработки управления $u^*(\tau)$ ($\tau \in T_h^1$) регулятор использует решение задачи (3.5) для позиции $(\tau, x^*(\tau))$. Чтобы осуществить построение управления $u^0(t | t^1, x^*(t^1))$ ($t \in T^1$) в режиме реального времени, параллельно с вычислением $u^*(\tau)$, $\tau \in T_h^0$, вычисляются и уточняются приближения для сопровождающих элементов этого управления. Для этого для каждой реализовавшейся позиции $(\tau, x^*(\tau))$ ($\tau \in T_h^0$) находим состояние

$$x(t^1 | \tau, x^*(\tau)) = F(t^1)F^{-1}(\tau)x^*(\tau) + \int_{\tau}^{t^1} F(t^1)F^{-1}(s)b(s)u^{\alpha^0}(s | \tau, x^*(\tau), 0)ds$$

и строим ОУ $u^0(t | t^1, x(t^1 | \tau, x^*(\tau)))$ ($t \in T^1$) задачи (3.5) для позиции $(t^1, x(t^1 | \tau, x^*(\tau)))$. Действуя таким образом, к моменту $\tau = t^1$ будем иметь оптимальные сопровождающие элементы задачи (3.5) для позиции $(t^1, x(t^1 | t^1 - h, x^*(t^1 - h)))$. Поскольку различие в состояниях $x(t^1 | t^1 - h, x^*(t^1 - h))$ и $x^*(t)$ определяется только действием ограниченного возмущения на промежутке длины h , то по имеющимся данным регулятор сможет построить управление $u^0(t | t^1, x^*(t^1))$ ($t \in T^1$) за время, не превосходящее h единиц реального времени.

6. ПРИМЕР 1

Предложенные методы гарантированной оптимизации проиллюстрируем на задаче максимального разгона колебательной системы. Математическая модель задачи имеет вид

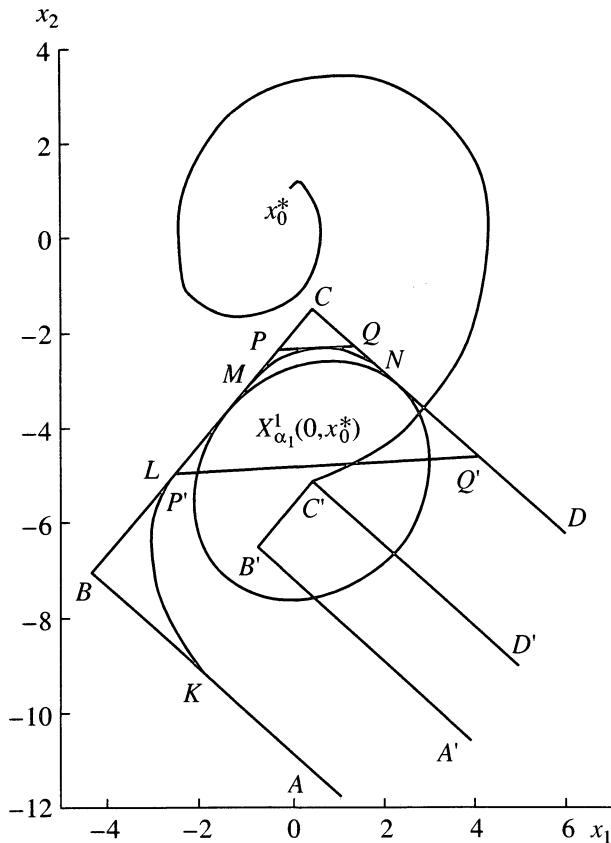
$$\begin{aligned} x_2(t^*) &\rightarrow \max, \quad \dot{x}_1 = x_2, \quad \dot{x}_2 = -x_1 + u + w, \\ x(0) &= x_0^*, \quad x_1(t^*) \in X^* = \{x_1 \in \mathbb{R} : x_{1*} \leq x_1 \leq x_1^*\}, \\ |u(t)| &\leq 1, \quad w(t) \in W = \{w \in \mathbb{R} : |w| \leq w^*\}, \quad t \in T. \end{aligned} \quad (6.1)$$

Параметрам задачи присвоены значения $x_0^* = (0, 1)$, $t^* = 12$, $x_{1*} = 2$, $x_1^* = 7$, $w^* = 0.5$.

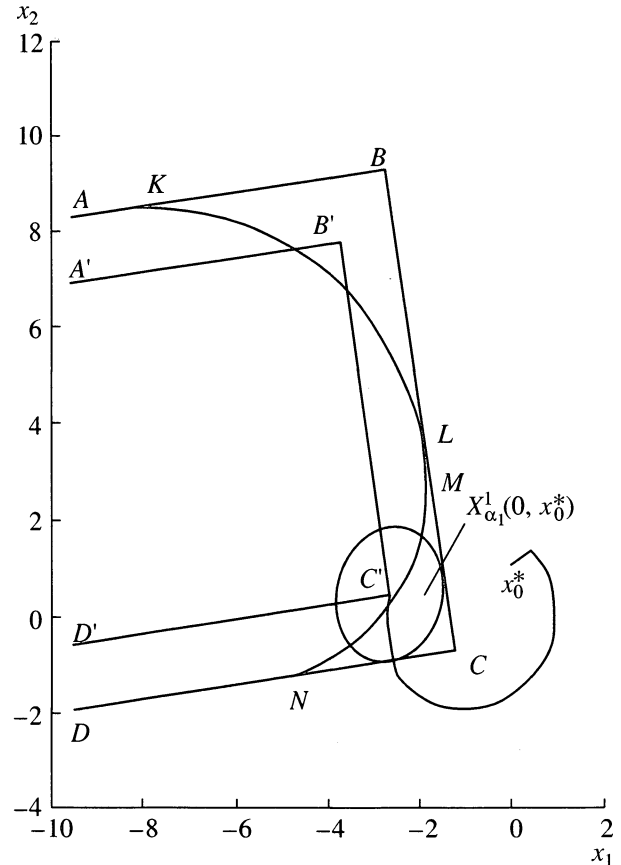
Вначале проиллюстрируем построение ОПУ $u^{\alpha^0}(t | 0, x_0^*)$ ($t \in T^0$). В качестве момента замыкания возьмем $t^1 = 8$. После выполнения первой процедуры алгоритма 1 получено значение $\alpha_1 = 2.751587$ и управление

$$u^{\alpha_1, 1}(t | 0, x_0^*, 0) = \begin{cases} 1, & t \in [0, 0.16[\cup [3.36, 6.48[, \\ -0.074369, & t \in [0.16, 0.24[, \\ -1, & t \in [0.24, 3.36[\cup [6.48, 8]. \end{cases} \quad (6.2)$$

На фиг. 1 α_1 -множество замыкания $X_{\alpha_1}^1$ ограничено линией $AKLMND$. Аппроксимирующее множество $X_{\alpha_1}^{1,1}$ задается линией $ABCD$. Линия $A'B'C'D'$ ограничивает суженное множество $\bar{X}_{\alpha_1}^{1,1}$, на которое в момент t^1 должна попадать траектория детерминированной системы (5.1). Действительно (см. фиг. 1), под действием управления $u^{\alpha_1, 1}(t | 0, x_0^*, 0)$ ($t \in T^0$) траектория системы (5.1) из состояния x_0^* приходит в точку $x^{\alpha_1}(t^1) = C'$. На фиг. 1 приведено также множество $X_{\alpha_1}^1(0, x_0^*)$ состояний, в которых может оказаться система (6.1) в момент t^1 под действием управления $u^{\alpha_1, 1}(t | 0, x_0^*, 0)$ ($t \in T^0$) и любых возмущений $w(t) \in W$ ($t \in T^0$). Как видно из фиг. 1, $X_{\alpha_1}^1(0, x_0^*) \subset X_{\alpha_1}^1$, т.е. управление (6.2) является ОПУ задачи (2.5). Чтобы это распознать, выполним вторую процедуру алгоритма. Множество активных ограничений задачи (5.3) $I_a = \{0, 1\}$, оптимальные множители Лагранжа $v_0^0 = -0.145131$, $v_1^0 = -0.139710$. Находим новый нормальный вектор $h_{(3)}^1 = (0.053955, -0.998543)$ и формируем множество $X_{\alpha_1}^{1,1}$. На фиг. 1 оно представлено линией $ABPQD$. Сужение множества $X_{\alpha_1}^{1,1}$ в направлении $h_{(3)}^1$ дает линию $P'Q'$. Таким образом, точка C'



Фиг. 1.



Фиг. 2.

по-прежнему принадлежит суженному множеству $\bar{X}_{\alpha_1}^{1,1}$. Поэтому, решив задачи (5.2), (5.3), получим $\alpha_2 = \alpha_1$, $x^{\alpha_2}(t^1) = x^{\alpha_1}(t^1)$, т.е. $\alpha^0 = \alpha_1$ – оптимальное гарантированное значение критерия качества, а управление (6.2) – ОПУ.

Возьмем теперь $t^1 = 4$. Выполнив первую процедуру алгоритма, получим $\alpha_1 = 3.865088$,

$$u^{\alpha_1,1}(t \mid 0, x_0^*, 0) = \begin{cases} 1, & t \in [0, 0.4] \cup [3.56, 4], \\ 0.870709, & t \in [3.52, 3.56], \\ -1, & t \in [0.4, 3.52]. \end{cases} \quad (6.3)$$

На фиг. 2 представлены множества $X_{\alpha_1}^1$, $X_{\alpha_1}^{1,1}$, $\bar{X}_{\alpha_1}^{1,1}$, $X_{\alpha_1}^1(0, x_0^*)$ и траектория детерминированной системы (5.1), соответствующая управлению (6.3) (обозначения те же, что и на фиг. 1). В отличие от предыдущего случая, теперь $X_{\alpha_1}^1(0, x_0^*) \not\subset X_{\alpha_1}^1$. Поэтому для решения задачи потребовалось выполнить 9 итераций алгоритма, в результате чего построено ОПУ

$$u^{\alpha^0}(t \mid 0, x_0^*, 0) = \begin{cases} 1, & t \in [0, 0.16] \cup [3.32, 4], \\ -0.724368, & t \in [0.16, 0.20], \\ -1, & t \in [0.2, 3.32]. \end{cases} \quad (6.4)$$

В табл. 1 для каждой итерации представлены значения α_k и $x^{\alpha_k}(t^1)$, $k = \overline{1, 9}$.

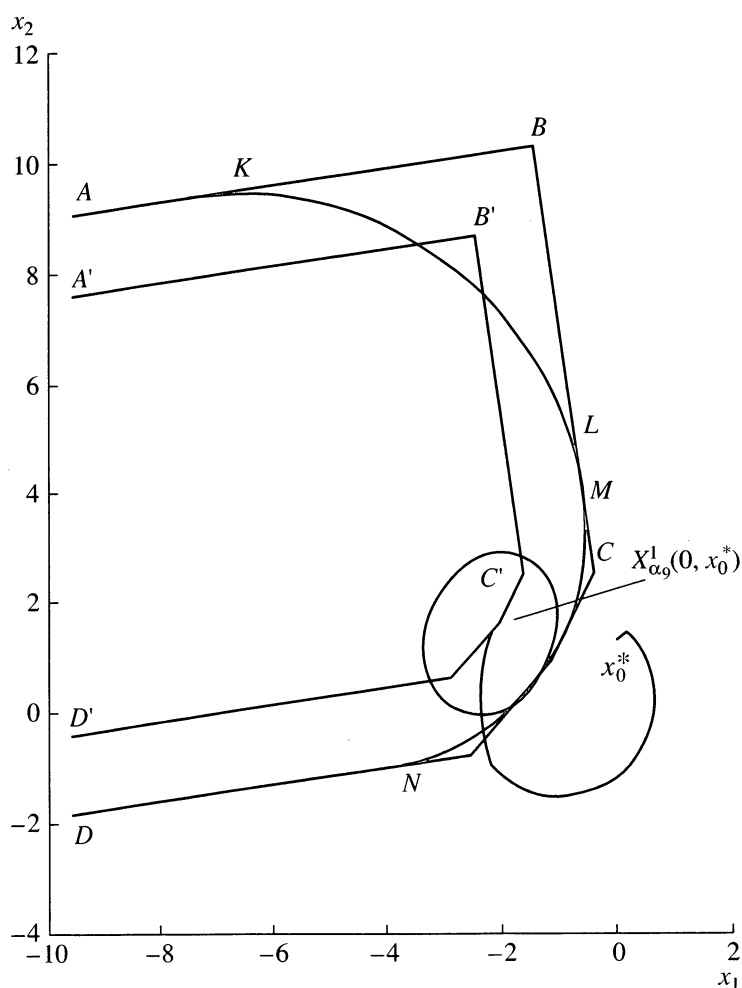
На фиг. 3 изображены множества $X_{\alpha_9}^1$, $X_{\alpha_9}^{1,9}$, $\bar{X}_{\alpha_9}^{1,9}$, $X_{\alpha_9}^1(0, x_0^*)$ и траектория детерминированной системы (5.1), соответствующая управлению (6.4).

Таблица 1

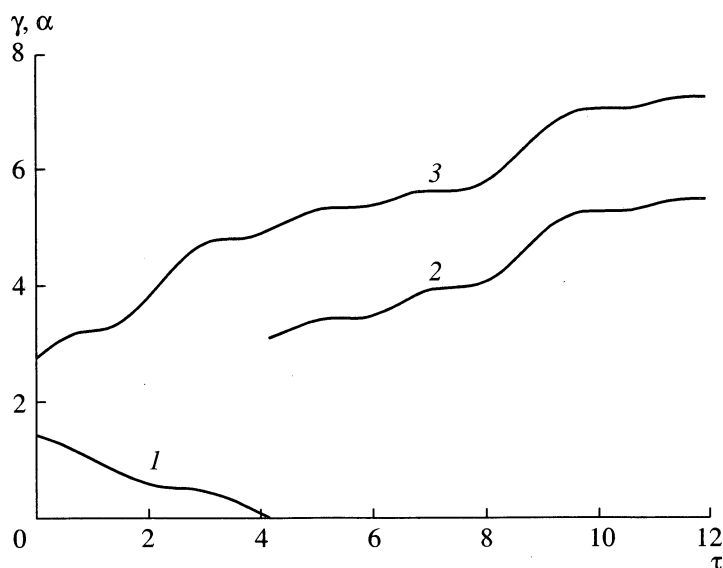
k	α_k	$x^{\alpha_k}(t^1)$
1	3.865088	-2.672962, 0.393520
2	2.772491	-2.670667, 0.398154
3	2.707208	-2.348972, 0.926592
4	2.598492	-2.336955, 0.943420
5	2.593512	-2.256832, 1.049475
6	2.583112	-2.273043, 1.028590
7	2.581439	-2.200702, 1.119911
8	2.579865	-2.196962, 1.124546
9	2.579865	-2.196957, 1.124554

Перейдем к решению задачи позиционной гарантированной оптимизации системы (6.1). Множество $\bar{X}^*(0, x_0^*)$ пусто, поэтому оптимальную размыкаемую ОС построить невозможно. Рассмотрим поведение системы, замкнутой двухфазной размыкаемой и замыкаемой ОС.

Пусть на систему действует неизвестное регулятору возмущение $w^*(t) = 0.5 \sin(2t)$ ($t \in T$). Значения реализации двухфазной размыкаемой ОС вычисляются с помощью решения задачи первой фазы (4.2) до момента $\bar{t} = 4.16$: $u^*(\tau) = u_\gamma^0(\tau | \tau, x^*(\tau))$ ($\tau \in T_h^0(0)$). На фиг. 4 линия 1 представ-



Фиг. 3.



Фиг. 4.

ляет реализовавшиеся значения $\gamma^0(\tau, x^*(\tau))$ ($\tau \in [0, \bar{t}]$). В момент \bar{t} начинается вторая фаза управления и для вычисления $u^*(\tau)$ ($\tau \geq \bar{t}$) используется решение задачи (3.5): $u^*(\tau) = u^0(\tau | \tau, x^*(\tau))$. Линия 2 изображает гарантированные значения критерия качества (3.3) вдоль оптимальных управлений $u^0(t | \tau, x^*(\tau))$ ($t \in T(\tau)$, $\tau \in [\bar{t}, t^*]$). Такая стратегия управления позволила к концу процесса достичь значения критерия качества $h'_{(0)} x^*(t^*) = 5.462913$. При этом управление, поданное на вход системы (6.1), имело вид

$$u^*(\tau) = \begin{cases} -1, & \tau \in [0, 2.56[\cup [7.2, 10.4[, \\ -0.55866, & \tau \in [7.12, 7.2[, \\ 1, & \tau \in [2.56, 7.12[\cup [10.4, 12[, \end{cases}$$

Эффективность замыкаемой ОС иллюстрируется на фиг. 4 линией 3, которая представляет оптимальные значения $\alpha^0(\tau)$ вдоль реализовавшейся траектории $x^*(\tau)$ ($\tau \in T$). В качестве момента замыкания взято $t^1 = 8$, построена реализация

$$u^*(\tau) = \begin{cases} 1, & \tau \in [0, 0.16[\cup [3.52, 6.88[\cup [10.4, 12[, \\ 0.651871, & \tau \in [0.16, 0.24[, \\ -1, & \tau \in [0.24, 3.44[\cup [6.88, 10.4[, \\ 0.515574, & \tau \in [3.44, 3.52[. \end{cases}$$

К моменту t^* достигнуто значение критерия качества $\alpha^0(t^*) = 7.249922$.

Отметим, что если построить ОПУ для детерминированной системы (6.1) с известным возмущением $w^*(t)$ ($t \in T$), то получим оптимальное значение критерия качества 7.793629.

7. ОПТИМАЛЬНАЯ МНОГОКРАТНО ЗАМЫКАЕМАЯ ОБРАТНАЯ СВЯЗЬ

Рассмотрим теперь задачу синтеза оптимальной замыкаемой ОС с несколькими моментами замыкания:

$$\{t^i \in T_h, i = \overline{1, p}\}, \quad t_* < t^1 < \dots < t^p < t^* = t^{p+1}. \quad (7.1)$$

Другими словами, при построении ОС будем принимать во внимание, что система управления будет замкнута не только в текущий момент $\tau \in T_h$, но и во все моменты (7.1). При этом предполо-

жим, что имеющиеся вычислительные средства позволяют обрабатывать всю доступную информацию в режиме реального времени. Моментам замыкания поставим в соответствие множества замыкания

$$X^1, \dots, X^p. \quad (7.2)$$

Множества (7.2) определим рекуррентно. Положим $X^{p+1} = X^*$. Множество X^i содержит состояния $x \in \mathbb{R}^n$, которые с гарантией можно перевести на множество X^{i+1} на промежутке $T^i = [t^i, t^{i+1}]$.

Оптимальная ОС, построенная с помощью p моментов замыкания, называется оптимальной многократно замыкаемой ОС. При $p = N$ оптимальная многократно замыкаемая ОС называется оптимальной замкнутой ОС. Именно такие обратные связи фигурируют в динамическом программировании.

Изменения, которые следует внести в конструкции разд. 5, поясним для случая двух моментов замыкания ($p = 2$). Обобщение на случай многих моментов замыкания проводится аналогичным образом, но из-за громоздкости описания здесь не приводится.

Выберем два момента замыкания $t^1, t^2 \in T_h$ ($\tau < t^1 < t^2 < t^* - h$) и, следуя разд. 5, введем α -множества замыкания X_α^2, X_α^1 и α -допустимые управления $u^\alpha(t | t^1, x, 1)$ ($t \in T^1, x \in X_\alpha^1$), $u^\alpha(t | t^2, x, 2)$ ($t \in T^2, x \in X_\alpha^2$).

Для удобства обозначений положим $t^0 = \tau, T^0 = T^0(\tau), h_{(i)}^3 = h_{(i)}$ ($i = \overline{0, m}$).

Максимальное значение α^0 , для которого существуют множества $X_{\alpha^0}^1, X_{\alpha^0}^2$ и α^0 -допустимое управление $u^{\alpha^0}(t | \tau, z, 0)$ ($t \in T^0(\tau)$), является оптимальным гарантированным значением критерия качества задачи (2.5) для моментов замыкания t^1, t^2 .

Через $X_\alpha^2(x)$ обозначим множество состояний, в которых может оказаться система (2.5), начав движение из позиции (t^1, x) , под действием управления $u^\alpha(t | t^1, x, 1)$ ($t \in T^1$) и любых реализаций возмущения $w(t) \in W$ ($t \in T^1$).

ОПУ задачи (2.5) с моментами замыкания t^1, t^2 представляет собой α^0 -допустимое управление $u^{\alpha^0}(t | \tau, z, 0)$ ($t \in T^0(\tau)$) и семейства α^0 -допустимых управлений $u^{\alpha^0}(t | t^1, x, 1)$ ($t \in T^1$), $u^{\alpha^0}(t | t^2, y, 2)$ ($t \in T^2, y \in X_{\alpha^0}^2(x), x \in X_{\alpha^0}^1(\tau, z)$).

Опишем, какие изменения нужно внести в алгоритм 1 для вычисления α^0 и построения управления $u^{\alpha^0}(t | \tau, z, 0)$ ($t \in T^0(\tau)$) в случае двух моментов замыкания.

Полагаем $k = 1, M(k) = m$.

Для построения аппроксимирующих многогранников $X_\alpha^{2,k}, X_\alpha^{1,k}$ решаем задачи (5.2) сначала на промежутке T^2 , затем на промежутке T^1 :

$$\begin{aligned} \beta_i^{l\alpha} &= \min_{y, u} h_{(i)}^{l_1} y, \quad \dot{x} = A(t)x + b(t)u, \quad x(t^l) = y, \\ x(t^{l+1}) &\in \bar{X}_\alpha^{l+1,k}, \quad |u(t)| \leq 1, \quad t \in T^l, \quad i = \overline{0, M(k)}, \quad l = 2, 1, \end{aligned} \quad (7.3)$$

где

$$h_{(i)}^{l_1} = h_{(i)}^{l+1_1} F(t^{l+1} - t^l) \|F'(t^{l+1} - t^l) h_{(i)}^{(l+1)}\|_2, \quad i = \overline{0, m}, \quad l = 2, \quad i = \overline{0, M(k)}, \quad l = 1. \quad (7.4)$$

В качестве множества $\bar{X}_\alpha^{3,k}$ берем множество \bar{X}_α^* . Множества $X_\alpha^{2,k}, \bar{X}_\alpha^{2,k}$ строим после решения задач (7.3) при $l = 2$:

$$X_\alpha^{l,k} = \{x \in \mathbb{R}^n : h_{(i)}^{l_1} x \geq \beta_i^{l\alpha}, i = \overline{0, M(k)}\}, \quad \bar{X}_\alpha^{l,k} = \{x \in \mathbb{R}^n : h_{(i)}^{l_1} x \geq \bar{\beta}_i^{l\alpha}, i = \overline{0, M(k)}\}, \quad (7.5)$$

где

$$\bar{\beta}_i^{l\alpha} = \beta_i^{l\alpha} - \min_{w(t) \in W, t \in T^{l-1}} \int_{t^{l-1}}^{t^l} h_{(i)}^{l'} F(t^l - s) d(s) w(s) ds.$$

Затем, решив задачи (7.3) при $l = 1$, найдем, согласно (7.5), множества $X_\alpha^{1,k}$, $\bar{X}_\alpha^{1,k}$.

Задача (5.3) остается без изменений, и алгоритм построения k -субоптимального управления задачи (2.5) принимает следующий вид.

Алгоритм 2

Шаг 1. Полагаем $\bar{\rho} = 0$.

Шаг 2. Для $l = 2, 1$ выполняем следующие шаги:

2.1. Решаем $M(k) + 1$ задачу (5.2).

2.2. Формируем множество $\bar{X}_\alpha^{l,k}$.

2.3. Если $\bar{X}_\alpha^{l,k} = \emptyset$, проверяем, можно ли, уменьшая α , получить $\bar{X}_\alpha^{l,k} \neq \emptyset$. Если да, полагаем $\alpha_{\max} = \alpha$, уменьшаем α , переходим к шагу 2.1. Если нет, то в задаче (2.5) не существует гарантированного управления для моментов замыкания t^1, t^2 .

Шаг 3. Решаем задачу (5.3), находим ρ_α .

Шаг 4. Если $\alpha = \alpha_{\max}$ и $\rho_\alpha \leq 1$ или $1 - \varepsilon \leq \rho_\alpha \leq 1$ (ε – заданная точность), то решение завершаем, полагаем $\alpha_k = \alpha$, и $u_{\alpha_k}^0(t | \tau, z, 0) = u_{\alpha_k}^0(t) (t \in T^0(\tau))$.

В случае $\rho_\alpha = \bar{\rho} > 1$ задача (2.5) не имеет допустимых управлений.

При $\rho_\alpha > 1$ уменьшаем α . Если $\rho_\alpha < 1 - \varepsilon$, то увеличиваем α , полагаем $\bar{\rho} = \rho_\alpha$. Переходим к шагу 2.

Приращение α уменьшаем при приближении значения ρ_α к 1.

Основные изменения в алгоритме, вызванные введением второго момента замыкания, относятся ко второй процедуре. Теперь для уточнения структуры множеств $X_\alpha^{1,k}$, $X_\alpha^{2,k}$ вначале по определенным правилам пополним систему нормальных векторов $h_{(i)}^2, i = \overline{0, M(k)}$, определяющих множество $X_\alpha^{2,k}$, а затем по формуле (7.4) подсчитаем соответствующие векторы $h_{(i)}^1, i = \overline{M(k) + 1, M(k + 1)}$.

Как и в разд. 5, после первой процедуры имеем точку $x^\alpha(t^1)$, множество I_a активных ограничений задачи (5.3) и оптимальные множители Лагранжа $v_i^0 (i \in I_a)$ задачи (5.3).

По правилам (5.4), (5.5) (подставляя h^2 вместо h^1 и используя правило (5.5) каждый раз, когда $|I_a| = 1$) находим новые нормальные векторы $h_{(i)}^2, i = \overline{M(k) + 1, M(k + 1)}$, используя множество I_a и множители Лагранжа $v_i^0 (i \in I_a)$.

Далее найдем разброс состояний системы (2.5) в момент t^1 относительно точки $x^\alpha(t^1)$ в направлениях $h_{(i)}^1 (i \in I_a)$ под действием возмущений $w(t) \in W (t \in T^0(\tau))$. Для этого решим $|I_a|$ задач ЛП относительно состояния x :

$$h_{(i)}^1 x \longrightarrow \max_x, \quad \int_{\tau}^{t^1} F(t^1 - s) d(s) w(s) ds - x = -x^\alpha(t^1), \quad w(t) \in W, \quad t \in T^0(\tau). \quad (7.6)$$

Оптимальный план i -й задачи (7.6) обозначим через x_w^i .

Для точек $x_w^i (i \in I_a)$ применим алгоритм 1, начав с вычисленного значения α_k и перенеся используемые обозначения с момента t^1 , соответственно, на момент t^2 . В результате выполнения первой процедуры алгоритма находим $\alpha(x_w^i)$. В отличие от п. 5.1, если на начальном $\alpha = \alpha_k$ значение критерия качества задачи (5.3) удовлетворяет неравенству $\rho_\alpha < 1$, то α не увеличиваем, а

завершаем первую процедуру, полагая $\alpha(x_w^i) = \alpha_k$. При переходе к следующей точке x_w^{i+1} уточняем значение $\alpha_k : \alpha_k := \min\{\alpha_k, \alpha(x_w^i)\}$.

Для каждой точки x_w^i ($i \in I_a$) по результатам первой процедуры выполняем вторую процедуру алгоритма 1, пополняя систему векторов $h_{(i)}^2, i = \overline{0, M(k+1)}$. Таким образом, получаем полный набор нормальных векторов $h_{(i)}^2, i = \overline{0, M(k+1)}$, и по формуле (7.4) находим соответствующие векторы $h_{(i)}^2, i = \overline{M(k+1), M(k+1)}$. Полагаем $k := k + 1$ и переходим к следующей итерации основного алгоритма, начиная выполнение первой процедуры с уточненного значения $\alpha = \alpha_{k-1}$.

Решение завершаем при выполнении условия (5.6).

Для определения оптимальной замыкаемой ОС обозначим через X_τ^α множество всех $z \in \mathbb{R}^n$, для которых в задаче (2.5) существуют ОПУ с моментами замыкания t^1, t^2 в случае $\tau \in T_h^0(t_*)$ и с моментом замыкания t^2 в случае $\tau \in T_h^1 = \{t^1, t^1 + h, \dots, t^2 - h\}$.

Функцию

$$u^0(\tau, z) = \begin{cases} u^{\alpha^0}(\tau | \tau, z, 0), & z \in X_\tau^\alpha, \quad \tau \in T_h^0(t_*), \\ u^{\alpha^0}(\tau | \tau, z, 1), & z \in X_\tau^\alpha, \quad \tau \in T_h^1, \\ u^0(\tau | \tau, z), & z \in X_\tau, \quad \tau \in T_h^2 = \{t^2, t^2 + h, \dots, t^* - h\}, \end{cases}$$

назовем оптимальной замыкаемой ОС с моментами замыкания t^1, t^2 .

Реализация оптимальной ОС $u^*(\tau) = u^*(\tau | \tau, x^*(\tau))$ ($\tau \in T_h$) строится по схеме п. 5.2. При этом параллельно с вычислением $u^*(\tau)$ ($\tau \in T_h^0(t_*)$) выполняются предварительные вспомогательные действия: подсчитывается состояние $x(t^1 | \tau, x^*(\tau))$, для него вычисляется оптимальное программное управление $u^{\alpha^0}(t | t^1, x(t^1 | \tau, x^*(\tau)), 1)$ ($t \in T_h^1$), находится точка

$$x(t^2 | t^1, x(t^1 | \tau, x^*(\tau))) = F(t^2)F^{-1}(t^1)x(t^1 | \tau, x^*(\tau)) + \\ + \int_{t^1}^{t^2} F(t^2)F^{-1}(s)b(s)u^{\alpha^0}(s | t^1, x(t^1 | \tau, x^*(\tau)), 1)ds$$

и строится ОПУ $u^0(t | t^1, x(t^2 | t^1, x(t^1 | \tau, x^*(\tau))))$ ($t \in T^2$). Использование этой вспомогательной информации позволяет осуществить в моменты t^1, t^2 переход на другой способ управления в режиме реального времени.

Замечание 2. Систему управления (2.1) с заданными ограничениями на управление и возмущение назовем эффективной, если при любых достаточно малых $h > 0$ и моментах замыкания $t_* = t^0 < \dots < t^p < t^*$ максимальный радиус r_k шара, который можно поместить в множество замыкания X^k , растет с уменьшением индекса k . Для эффективных систем управления можно для любых терминального множества X^* и числа $\varepsilon > 0$ построить такую оптимальную замыкаемую ОС, что замкнутая ею система переводит любое состояние $x(t_*) \in X^0$ (X^0 – множество замыкания в момент t^0 , построенное по множеству X^1 по указанным выше правилам) в ε -окрестность множества X^* при любой реализации возмущения $w(t) \in W$ ($t \in T$). В этом состоит основное преимущество замыкаемой ОС перед размыкаемой. Однако размыкаемым ОС присуще одно полезное качество, которым не обладают замыкаемые ОС: если в некоторый момент времени выйдет из строя вычислительное устройство ОР, то программное управление, с помощью которого в этот момент строилась реализация оптимальной размыкаемой ОС, переведет систему управления на терминальное множество вне зависимости от реализующегося в дальнейшем возмущения.

8. ПРИМЕР 2

Снова рассмотрим задачу (6.1). В качестве начального состояния возьмем $x_0^* = (-1, -1)$. Построим ОПУ $u^{\alpha^0}(t | 0, x_0^*, 0)$ ($t \in T^0(0)$) для двух моментов замыкания: $t^1 = 6$, $t^2 = 9$. На фиг. 5 представлены α_1 -множества замыкания $X_{\alpha_1}^1, X_{\alpha_1}^2$, ограниченные линиями $AKLMND$, аппроксимирующие множества $X_{\alpha_1}^{1,1}, X_{\alpha_1}^{2,1}$ (линии $ABCD$) и суженные множества $\bar{X}_{\alpha_1}^{1,1}, \bar{X}_{\alpha_1}^{2,1}$ (линии $A'B'C'D'$). По множеству $\bar{X}_{\alpha_1}^{1,1}$ построено управление

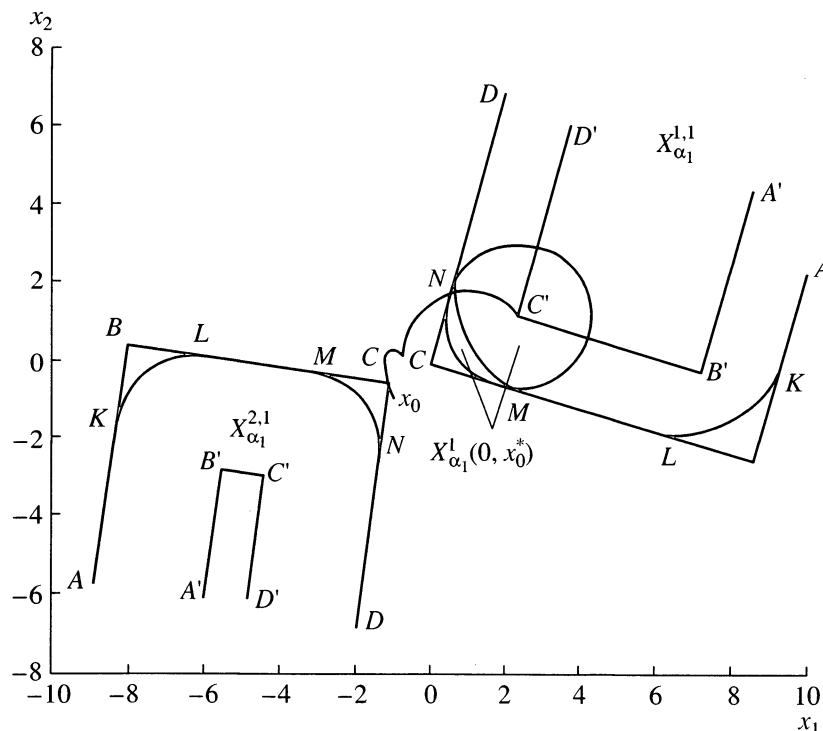
$$u^{\alpha_1,1}(t | 0, x_0^*, 0) = \begin{cases} 1, & t \in [0, 0.42[\cup [3.6, 6[, \\ -1, & t \in [0.42, 3.54[, \\ -0.778181, & t \in [3.54, 3.6[, \end{cases}$$

под действием которого траектория детерминированной системы (5.1) в момент t^1 приходит в точку C' . На фиг. 5 изображено также множество $X_{\alpha_1}^1(0, x_0^*)$ возможных состояний возмущенной системы (6.1) в момент t^1 . Видно, что $X_{\alpha_1}^1(0, x_0^*) \not\subset X_{\alpha_1}^1$.

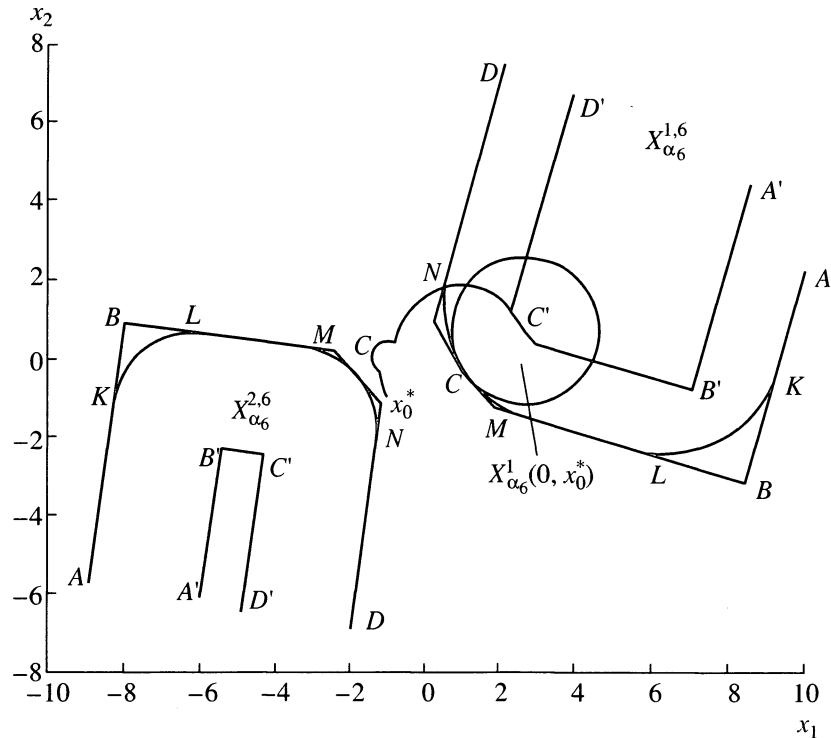
ОПУ $u^{\alpha^0}(t | 0, x_0^*, 0)$ ($t \in [0, 6[$) построено за 6 итераций и имеет вид

$$u^{\alpha^0}(t | 0, x_0^*, 0) = \begin{cases} 1, & t \in [0, 0.3[\cup [3.48, 6[, \\ -1, & t \in [0.3, 3.42[, \\ -0.388118, & t \in [3.42, 3.48[. \end{cases} \quad (8.1)$$

Табл. 2 содержит значения α_k и $x^{\alpha_k}(t^1)$, $k = \overline{1, 6}$, для каждой итерации.



Фиг. 5.



Фиг. 6.

Множества $X_{\alpha_6}^i$, $X_{\alpha_6}^{i,6}$, $\bar{X}_{\alpha_6}^{i,6}$ ($i = 1, 2$), $X_{\alpha_6}^1(0, x_0^*)$ и траектория детерминированной системы (5.1), соответствующая управлению (8.1), изображены на фиг. 6.

Таким образом, оптимальное гарантированное значение критерия качества $\alpha^0 = 1.175554$. Если вместо двух моментов замыкания использовать только один момент $t^1 = 6$, то получим $\alpha^0 = 0.646045$, в случае одного момента замыкания $t^1 = 9$ получаем $\alpha^0 = 0.733118$.

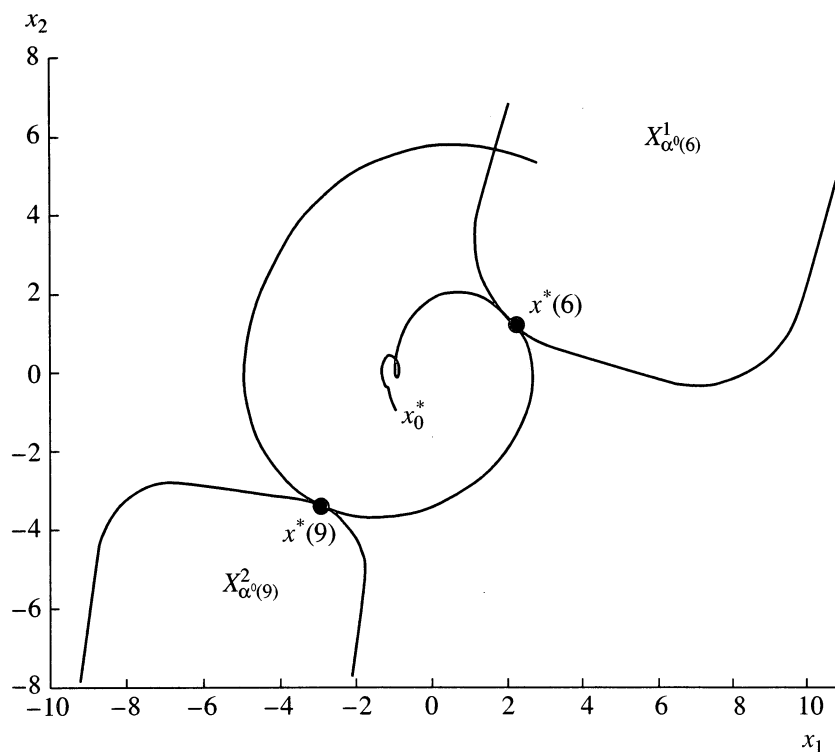
Построим теперь реализацию оптимальной замыкаемой обратной связи. Пусть на систему (6.1) действует неизвестное регулятору возмущение $w^*(t) = 0.5 \sin(2t)$ ($t \in T$). С использованием двух моментов замыкания $t^1 = 6$, $t^2 = 9$ регулятор построил реализацию ОС:

$$u^*(t) = \begin{cases} 1, & t \in [0, 0.24[\cup [3.84, 6.72[\cup [10.44, 12[, \\ -1, & t \in [0.24, 3.82[\cup [6.84, 10.44[, \\ -0.027471, & t \in [6.72, 6.84[. \end{cases}$$

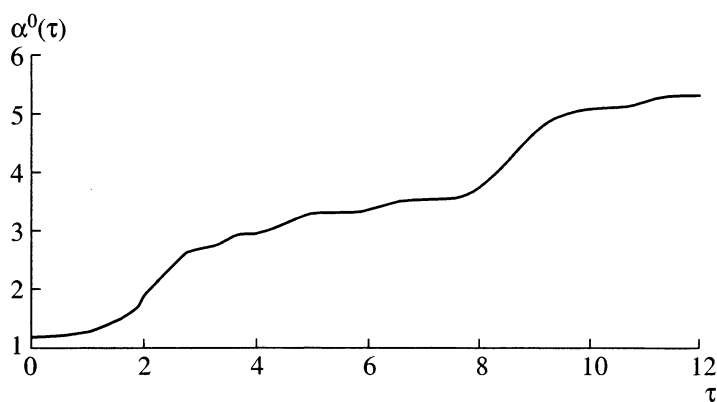
Соответствующая траектория $x^*(t)$ ($t \in T$) замкнутой системы (6.1) изображена на фиг. 7. Точками отмечены состояния $x^*(6)$, $x^*(9)$ и показаны множества $X_{\alpha^0(6)}^1$, $X_{\alpha^0(9)}^2$.

Таблица 2

k	α_k	$x^{\alpha_k}(t^1)$
1	1.720427	2.368126, 1.115149
2	1.221050	2.368125, 1.115145
3	1.184795	2.585360, 0.845156
4	1.175652	2.699952, 0.679233
5	1.175554	2.700842, 0.677907
6	1.175554	2.700842, 0.677907



Фиг. 7.



Фиг. 8.

К концу процесса достигнуто значение критерия качества $\alpha^0(t^*) = 5.316653$. Изменение оптимального гарантированного значения $\alpha^0(\tau)$ ($\tau \in T_h$) вдоль реализовавшейся траектории $x^*(\tau)$ ($\tau \in T$) представлено на фиг. 8. Как и в разд. 6, построим ОПУ для детерминированной системы (6.1) с заданным возмущением $w^*(t)$ ($t \in T$). При этом оптимальное значение критерия качества составит 5.638651.

СПИСОК ЛИТЕРАТУРЫ

1. Ньютон Дж.К., Гулд Л.А., Кайзер Дж.Ф. Теория линейных следящих систем. М.: Физматгиз, 1961.
2. Hopkin A.M. A phase plane approach to the design of saturating servomechanisms // Trans. AIEE. 1950. V. 70. № 3.
3. Фельдбаум А.А. О синтезе оптимальных систем с помощью фазового пространства // Автоматика и телемехан. 1955. Т. 16. № 1. С. 43–49.

4. Балашевич Н.В., Габасов Р., Кириллова Ф.М. Численные методы программной и позиционной оптимизации линейных систем управления // Ж. вычисл. матем. и матем. физ. 2000. Т. 40. № 6. С. 838–859.
5. Фельдбаум А.А. Основы теории оптимальных автоматических систем. М.: Физматгиз, 1963.
6. Лэнинг Дж., Бэттин Р. Случайные процессы в задачах автоматического управления. М.: Изд-во иностр. лит., 1958.
7. Пугачев В.С. Теория случайных функций и ее применение к задачам автоматического управления. М.: Физматгиз, 1960.
8. Дуб Дж. Вероятностные процессы. М.: Изд-во иностр. лит., 1956.
9. Witsenhausen H.S. A minimax control problem for sampled linear systems // IEEE Trans. Automat. Control. 1968. V. AC-13. № 1.
10. Schweppe F.C. Recursive state estimation: unknown but bounded errors and system inputs // IEE Trans. Automat. Control. 1968. V. AC-13. № 1.
11. Понтрягин Л.С., Болтянский В.Г., Гамкрелидзе Р.В., Мищенко Е.Ф. Математическая теория оптимальных процессов. М.: Наука, 1983.
12. Беллман Р. Динамическое программирование. М.: Изд-во иностр. лит., 1960.
13. Айзекс Р. Дифференциальные игры. М.: Мир, 1967.
14. Габасов Р., Дмитрук Н.М., Кириллова Ф.М. Оптимизация многомерных систем управления с параллелепипедными ограничениями // Автоматика и телемехан. 2002. № 3. С. 3–26.
15. Габасов Р., Кириллова Ф.М., Костина Е.А. Замыкаемые обратные связи по состоянию для оптимизации неопределенных систем управления // Автоматика и телемехан. 1996. № 7. С. 121–130; № 8. С. 90–99.
16. Dreyfus S.E. Some types of optimal control of stochastic systems // J. SIAM. Ser. A. Control. 1964. № 2. P. 120–134.
17. Альсевич В.В., Габасов Р., Глушенков В.С. Оптимизация линейных экономических моделей. Минск: БГУ, 2000.
18. Балашевич Н.В., Габасов Р., Кириллова Ф.М. Стабилизация динамических систем при наличии запаздываний в канале обратной связи // Автоматика и телемехан. 1996. № 6. С. 31–39.