



А ВТОМАТИКА И ТЕЛЕМЕХАНИКА

AUTOMATION AND REMOTE CONTROL

Журнал основан в 1936 году

Выходит 12 раз в год



ноябрь

Москва

2025

Учредители журнала:

Отделение энергетики, машиностроения, механики и процессов управления РАН,
Институт проблем управления им. В.А. Трапезникова РАН (ИПУ РАН),
Институт проблем передачи информации им. А.А. Харкевича РАН (ИППИ РАН)

Главный редактор:

Галяев А.А.

Заместители главного редактора:

Рубинович Е.Я., Хлебников М.В.

Ответственный секретарь:

Самохин А.С.

Редакционный совет:

Васильев С.Н., Желтов С.Ю., Каляев И.А., Кузнецов О.П., Кулешов А.П.,
Куржанский А.Б., Мартынюк А.А. (Украина), Пешехонов В.Г., Попков Ю.С.,
Федосов Е.А., Черноусько Ф.Л.

Редакционная коллегия:

Алескеров Ф.Т., Арутюнов А.В., Бахтадзе Н.Н., Бобцов А.А., Виноградов Д.В.,
Вишневский В.М., Воронцов К.В., Граничин О.Н., Жилякова Н.Ю.,
Каравай М.Ф., Кибзун А.И., Краснова С.А., Крищенко А.П., Кузнецов Н.В.,
Кушнер А.Г., Лазарев А.А., Ляхов А.И., Маликов А.И., Матасов А.И.,
Меерков С.М. (США), Мещеряков Р.В., Миллер Б.М., Михальский А.И.,
Моржин О.В., Мунасыпов Р.А., Назин А.В., Немировский А.С. (США),
Новиков Д.А., Олейников А.Я., Пакшин П.В., Пальчунов Д.Е.,
Поляков А.Е. (Франция), Протасов В.Ю., Рапопорт Л.Б., Родионов И.В.,
Сельвесюк Н.И., Соболевский А.Н., Степанов О.А., Филимонюк Л.Ю.,
Фрадков А.Л., Хоров Е.М., Цыбаков А.Б. (Франция), Чеботарев П.Ю.,
Чхартишвили А.Г., Щербаков П.С.

Адрес редакции: 117997, Москва, Профсоюзная ул., 65

Тел./факс: 8 (495) 198-17-20, доб. 1443

Электронная почта: ait@pran.ru, redacsia@ipu.ru

Сайт: <https://aitras.ru>

Зав. редакцией *Е.А. Мартехина*

Москва

ФГБУ «Издательство «Наука»

© 2025 г. Д.Н. ИБРАГИМОВ, канд. физ.-мат. наук (rikk.dan@gmail.com)
(Московский авиационный институт
(национальный исследовательский университет)),
К.А. ЦАРЬКОВ, канд. физ.-мат. наук (k6472@mail.ru)
(Институт проблем управления им. В.А. Трапезникова РАН, Москва)

ЧИСЛЕННЫЙ МЕТОД РЕШЕНИЯ ЗАДАЧИ БЫСТРОДЕЙСТВИЯ ДЛЯ ЛИНЕЙНЫХ НЕСТАЦИОНАРНЫХ ДИСКРЕТНЫХ СИСТЕМ ОБЩЕГО ВИДА

Построена программно реализуемая численная процедура решения задачи быстрогодействия для линейных дискретных систем с произвольными переменными матрицами системы и выпуклыми множествами геометрических ограничений на управление. Доказана сходимость выстраиваемой алгоритмом последовательности процессов управления к решению задачи. Эффективность продемонстрирована на ряде примеров.

Ключевые слова: задача быстрогодействия, линейные дискретные системы, последовательное глобальное улучшение, метод Кротова.

DOI: 10.31857/S0005231025110018

1. Введение

Задача быстрогодействия представляет собой одну из классических проблем теории оптимального управления. В наибольшей общности она ставится как задача перевода заданной динамической системы из фиксированного начального состояния в некоторое известное множество в фазовом пространстве (целевую область) за наименьшее возможное время. При этом сама система может функционировать как в непрерывном, так и в дискретном времени, может быть линейной или нелинейной, а в качестве целевой области может выступать как заданная гладкая поверхность, так и одноточечное множество (сингльтон). В настоящей работе под задачей быстрогодействия будем понимать задачу перевода линейной динамической системы, действующей в дискретном времени, из заданного начального состояния в начало координат за наименьшее возможное число шагов. За исключением некоторых специальных частных случаев основным инструментом исследования задач такого рода всегда был и остается метод прямого перебора моментов времени возможного окончания переходного процесса и непосредственного решения получающихся при этом конечномерных задач терминальной стабилизации. В силу этих причин непосредственно задаче быстрогодействия в указанном выше смысле посвящено сравнительно небольшое число работ. Среди таковых выделим достаточно общие рассуждения в [1–3], а также продолжающие их [4–8].

В настоящей статье разрабатывается принципиально отличный алгоритм решения задачи быстродействия, в основу которого положен метод построения последовательных глобальных улучшений процессов управления, предложенный в работах В.Ф. Кротова [9, 10]. Этот метод также применяется к задачам терминальной стабилизации заданной линейной дискретной системы при различных фиксированных моментах времени окончания переходного процесса. Однако в отличие от классического подхода указанные задачи не предполагается решать полностью. Улучшения строятся до тех пор, пока не будет либо найден оптимальный по быстродействию процесс, либо доказана невозможность попадания в начало координат за заданное число шагов. В этих целях строится оценка оптимального значения функционала качества по текущему приближению. Аналогичный подход может быть реализован и при использовании других численных методов условной конечномерной оптимизации и улучшения, в том числе разработанных непосредственно для исследования дискретных задач оптимального управления. К последним относятся методы из [11, гл. 6] и некоторые методы из [12, часть 2 и часть 4]. Отличительной особенностью метода Кротова является скорость улучшения, а также то обстоятельство, что в рамках рассматриваемых задач терминальной стабилизации линейных дискретных систем для него удается установить наличие ключевого свойства строгого улучшения неоптимальных процессов. Общие результаты о нестрогом улучшении в задачах оптимального управления для дискретных и непрерывных систем были получены В.Ф. Кротовым. Некоторые частные результаты о строгом улучшении, близкие к обсуждаемым здесь, но, в основном, в случае непрерывных систем, излагаются в [13].

Настоящая работа непосредственно продолжает исследования, начатые в [14], где был предложен подход к построению двусторонних оценок времени быстродействия для стационарных линейных систем с невырожденной диагонализированной матрицей системы, а также разработан численный метод улучшения верхней оценки и построения гарантирующего процесса. Здесь результаты, полученные в [14], развиваются в следующих направлениях. Во-первых, будет показано, что итерационная процедура из [14] позволяет не только улучшать верхние оценки времени быстродействия и строить гарантирующие процессы, но и в действительности способна без каких-либо дополнительных предположений приближенно находить оптимальные управления в соответствующей задаче быстродействия. Во-вторых, будет предложена новая процедура, позволяющая решать задачу быстродействия и строить оптимальные процессы в отсутствие каких бы то ни было известных оценок времени быстродействия. В-третьих, все результаты будут обобщены на случай нестационарных линейных дискретных систем общего вида. В частности, все они применимы к стационарным системам с вырожденной матрицей.

2. Постановка задачи

Рассмотрим линейную нестационарную систему с дискретным временем

$$(1) \quad x(k+1) = A(k)x(k) + u(k), \quad k = 0, 1, \dots,$$

где $x(k) \in \mathbb{R}^n$ – состояние системы, $u(k) \in U(k)$ – управление, $U(k)$ – выпуклые компактные множества в $\mathbb{R}^n$, содержащие начало координат, $A(k) \in \mathbb{R}^{n \times n}$ – произвольные заданные матрицы. Начальное условие для системы (1) фиксировано:

$$(2) \quad x(0) = x_0 \in \mathbb{R}^n.$$

Требуется вычислить минимальное число шагов $N_{\min}$, за которое возможно перевести систему (1) из заданного начального состояния x_0 в начало координат и построить оптимальный процесс $\{x^*(k), u^*(k-1)\}_{k=1}^{N_{\min}}$, удовлетворяющий условию $x^*(N_{\min}) = 0$. Число $N_{\min}$ будем далее называть временем быстрогодействия системы (1) с начальным условием (2) и будем предполагать, что поставленная задача разрешима, т.е. $N_{\min} < \infty$.

В стационарном случае $A(k) \equiv A \in \mathbb{R}^{n \times n}$ и $U(k) \equiv U \subset \mathbb{R}^n$ хорошо известными достаточными условиями разрешимости задачи быстрогодействия для системы (1)–(2) являются устойчивость по Шуру матрицы A и включение $0 \in \text{int}U$ или управляемость по Калману системы (1) и включение $0 \in \text{ri}U$.

3. Задача с фиксированным временем и условия оптимальности

В [14] в случае стационарных систем с невырожденной матрицей $A(k) \equiv A$ был предложен алгоритм, позволяющий при известных оценках на время быстрогодействия строить гарантирующие процессы и улучшать верхнюю оценку. При этом последовательно решается несколько задач с фиксированным временем функционирования системы. Обсудим их.

Предварительно введем следующие обозначения. Пусть $N > 0$ – некоторое фиксированное значение дискретного времени и $k = 0, \dots, N-1$ в системе (1). Положим

$$\mathcal{U}_N := \{k \mapsto u(k) : \{0, \dots, N-1\} \rightarrow \mathbb{R}^n \mid u(k) \in U(k)\},$$

$$\mathcal{X}_N := \{k \mapsto x(k) : \{0, \dots, N\} \rightarrow \mathbb{R}^n\}.$$

Множество $\mathcal{X}_N$ будем отождествлять с евклидовым пространством $\mathbb{R}^{n(N+1)}$, а элементы множества $\mathcal{U}_N$ – с соответствующими векторами евклидова пространства $\mathbb{R}^{nN}$.

Рассмотрим для системы (1)–(2) задачу

$$(3) \quad J_N(u) = \|x(N)\|^2 \rightarrow \min_{u \in \mathcal{U}_N},$$

где $x \in \mathcal{X}_N$ – решение (1)–(2) при $k = 0, \dots, N - 1$ и фиксированном $u \in \mathcal{U}_N$. Здесь и далее через $\|\cdot\|$ обозначена евклидова норма в $\mathbb{R}^n$.

Задача (3) для системы (1)–(2) удовлетворяет условиям [11, с. 124, условия 1–4], которые гарантируют [11, теорема 5.6.2], что для оптимальности управления $\hat{u} \in \mathcal{U}_N$ в этой задаче необходимо и достаточно потребовать выполнения соотношений классического дискретного принципа максимума [11, теоремы 5.3.1 и 5.6.1]. Поскольку система (1) линейна, эти соотношения сводятся к проверке N условий максимума

$$(4) \quad \langle \hat{\psi}(k+1), \hat{u}(k) \rangle = \max_{u \in U(k)} \langle \hat{\psi}(k+1), u \rangle,$$

при $k = 0, \dots, N - 1$, где $\langle \cdot, \cdot \rangle$ обозначает скалярное произведение в $\mathbb{R}^n$, а значения двойственной переменной $\hat{\psi}(k)$ однозначно определяются формулами

$$(5) \quad \hat{x}(k+1) = A(k)\hat{x}(k) + \hat{u}(k), \quad \hat{x}(0) = x_0,$$

$$(6) \quad \hat{\psi}(k) = A(k)^T \hat{\psi}(k+1), \quad \hat{\psi}(N) = -2\hat{x}(N).$$

Нетрудно видеть, что при $N \geq N_{\min}$ оптимальные управления в задаче (3) являются особыми [15], поскольку в этом случае $\hat{\psi}(k) \equiv 0$. Для дальнейшего потребуется альтернативная форма записи условий оптимальности в этой задаче, относительно которой оптимальные управления уже не будут особыми ни при каком значении N .

Теорема 1. Пусть $\hat{u} \in \mathcal{U}_N$. Тогда для оптимальности $\hat{u}$ в задаче (1)–(3) необходимо и достаточно, чтобы при $k = 0, \dots, N - 1$ выполнялись включения

$$(7) \quad \hat{u}(k) \in \text{Arg} \min_{u \in U(k)} \|\mathcal{A}_N(k+1)u + \mathcal{A}_N(k)\hat{x}(k) - \hat{\xi}(k+1)\|,$$

где

$$(8) \quad \mathcal{A}_N(k) := A(N-1) \dots A(k), \quad k = 0, \dots, N-1, \quad \mathcal{A}_N(N) := I,$$

I – единичная матрица размеров $n \times n$, а значения $\hat{x}(k)$ и $\hat{\xi}(k)$ определяются равенствами (5) и

$$(9) \quad \hat{\xi}(k) = \hat{\xi}(k+1) - \mathcal{A}_N(k+1)\hat{u}(k), \quad \hat{\xi}(N) = 0.$$

Для удобства читателя доказательства приводимых в работе утверждений отнесены в Приложение 1.

Необходимые условия в теореме 1 с точностью до замен переменных и переобозначений были получены в [14] для случая $A(k) \equiv A$, $\det A \neq 0$, $U(k) \equiv U$. Достаточные условия получены здесь впервые.

Совокупность условий (5), (7) и (9), в отличие от (4)–(6), не вырождается в случае $N \geq N_{\min}$. Так, например, при $k = N - 1$ из (7) имеем

$$\hat{u}(N-1) \in \text{Arg} \min_{u \in U(N-1)} \|A(N-1)\hat{x}(N-1) + u\|,$$

а это условие всегда содержательно и отражает геометрический смысл оптимального управления в задаче (1)–(3).

В качестве прямого следствия из теоремы 1 можно получить утверждение о том, что алгоритм, предложенный в [14], позволяет приближенно находить оптимальные процессы в задаче быстрогодействия для стационарных невырожденных линейных систем. Это усиливает теоретические результаты, представленные в указанной работе. Предлагаемый ниже алгоритм решения задачи быстрогодействия также существенно опирается на утверждение теоремы 1.

4. Теоремы об улучшении

Пусть $\hat{u} \in \mathcal{U}_N$ – некоторое произвольное управление. Зададимся целью построить новое управление $\tilde{u} \in \mathcal{U}_N$, улучшающее $\hat{u}$ в смысле значения функционала J_N в задаче (3).

Теорема 2. Пусть $\hat{u} \in \mathcal{U}_N$ – произвольное управление, а $\hat{x}, \hat{\xi} \in \mathcal{X}_N$ – соответствующие ему решения уравнений (5) и (9). Определим управление $\tilde{u} \in \mathcal{U}_N$ из условия

$$(10) \quad \tilde{u}(k) \in \text{Arg} \min_{u \in U(k)} \|\mathcal{A}_N(k+1)u + \mathcal{A}_N(k)\tilde{x}(k) - \hat{\xi}(k+1)\| \quad \forall k \in \{0, \dots, N-1\},$$

где $\mathcal{A}_N(k)$ определяется из (8), а $\tilde{x}(k)$ удовлетворяет равенствам

$$(11) \quad \tilde{x}(k+1) = A(k)\tilde{x}(k) + \tilde{u}(k), \quad \tilde{x}(0) = x_0.$$

Тогда относительно управлений $\hat{u}, \tilde{u} \in \mathcal{U}_N$ имеет место нестрогое улучшение в задаче (3), т.е.

$$J_N(\tilde{u}) \leq J_N(\hat{u}).$$

Теорема 2 с точностью до обозначений была доказана в [14] в предположениях $A(k) \equiv A$, $\det A \neq 0$, $U(k) \equiv U$, которые несущественны. Доказательство, приведенное в Приложении 1, полностью основано на конструкциях глобального улучшения В.Ф. Кротова.

Теорема 3. Пусть $\hat{u} \in \mathcal{U}_N$, $\hat{x} \in \mathcal{X}_N$ удовлетворяет уравнению (5) и для пары $(\hat{x}, \hat{u})$ выполняются условия (10), (11). Тогда равенство $J_N(\tilde{u}) = J_N(\hat{u})$ имеет место в том и только том случае, когда управление $\hat{u}$ является оптимальным в задаче (1)–(3).

Теорема 3 утверждает, что любое неоптимальное управление в задаче (1)–(3) можно строго улучшить относительно значений функционала J_N , построив новое управление по соотношениям (10), (11). Поскольку все множества $U(k)$ предполагаются компактными, эти соотношения разрешимы всегда (возможно, не единственным образом).

5. Оценка оптимального значения функционала качества

При исследовании задачи (3) в контексте исходной задачи быстрогодействия для системы (1)–(2) важно уметь достаточно точно оценить снизу оптимальное значение функционала J_N . Для этого можно использовать информацию, полученную в процессе построения улучшений. Приведем одну такую оценку.

Теорема 4. Пусть $\hat{u} \in \mathcal{U}_N$, значения $\hat{x}(k)$ и $\hat{\psi}(k)$ определены из (5) и (6), а J_N^* – оптимальное значение функционала J_N в задаче (3). Тогда

$$(12) \quad J_N^* \geq \|\hat{x}(N)\|^2 - \sum_{k=0}^{N-1} \max_{u \in U(k)} \langle \hat{\psi}(k+1), u - \hat{u}(k) \rangle.$$

Из теоремы 4 вытекает следующее: если при некотором $N > 0$ для какого-то управления $\hat{u} \in \mathcal{U}_N$ правая часть неравенства (12) оказывается строго положительной, то гарантированно имеет место $N_{\min} > N$, поскольку $J_N^* = 0$ при всех $N \geq N_{\min}$ ввиду условий $0 \in U(k)$. В частности, подставляя в (12) управление $\hat{u} = 0 \in \mathcal{U}_N$, получаем такую оценку времени быстрогодействия:

$$N_{\min} \geq \min \left\{ N \geq 0 \mid \|\mathcal{A}_N(0)x_0\|^2 + 2 \sum_{k=0}^{N-1} \min_{u \in U(k)} \langle \mathcal{A}_N(k+1)^T \mathcal{A}_N(0)x_0, u \rangle \leq 0 \right\}.$$

Следует отметить, что аналогичное (12) неравенство имеется в [11, с. 166].

6. Итерационный алгоритм решения задачи быстрогодействия

Перейдем к построению общего алгоритма решения исходной задачи быстрогодействия для системы (1)–(2). Для этого будем последовательно улучшать процессы управления в задаче (3) при значениях $N = 1, 2, \dots$, применяя результаты раздела 4, и использовать неравенство (12) для быстрой оценки теоретической возможности попадания в начало координат при текущем значении N . Исключая тривиальный случай, будем предполагать, что $N_{\min} > 0$.

Имеем следующий алгоритм:

0. Задать величину допустимой погрешности вычислений $\varepsilon > 0$, положить $N = 1$.
1. Задать/дополнить начальное приближение равенством $u^{(0)}(N-1) = 0$, положить $l = 0$ и подсчитать матрицы $\mathcal{A}_N(k) = A(N-1) \dots A(k)$, $k = 0, \dots, N-1$, матрицу $\mathcal{A}_N(N)$ задать единичной, размером $n \times n$.
2. Найти решение $\xi^{(l)}$ системы уравнений

$$\xi(k) = \xi(k+1) - \mathcal{A}_N(k+1)u^{(l)}(k), \quad \xi(N) = 0.$$

3. Для каждого $k \in \{0, \dots, N-1\}$ последовательно найти и зафиксировать некоторое решение $u^{(l+1)}(k)$ экстремальной задачи

$$\|\mathcal{A}_N(k+1)u + \mathcal{A}_N(k)x^{(l+1)}(k) - \xi^{(l)}(k+1)\| \rightarrow \min_{u \in U(k)},$$

где значения $x^{(l+1)}(k)$ вычисляются по формулам

$$x^{(l+1)}(k+1) = A(k)x^{(l+1)}(k) + u^{(l+1)}(k), \quad k = 0, \dots, N-1, \quad x^{(l+1)}(0) = x_0.$$

4. Проверить условие внешней остановки

$$\|x^{(l+1)}(N)\| < \varepsilon,$$

при выполнении закончить расчеты с ответом $N_\varepsilon = N$, $u_\varepsilon = u^{(l+1)}$.

5. Найти решение $\psi^{(l+1)}$ системы уравнений

$$\psi(k) = A(k)^T \psi(k+1), \quad k = 0, \dots, N-1, \quad \psi(N) = -2x^{(l+1)}(N).$$

6. Вычислить оценку $E_N^{(l+1)}$ возможного приближения к нулю по формуле

$$E_N^{(l+1)} = \|x^{(l+1)}(N)\|^2 - \sum_{k=0}^{N-1} \max_{u \in U(k)} \langle \psi^{(l+1)}(k+1), u - u^{(l+1)}(k) \rangle.$$

7. Проверить условие внутренней остановки

$$E_N^{(l+1)} > 0,$$

при выполнении зафиксировать найденное $u^{(l+1)}$ в качестве нового начального приближения, т.е. положить $u^{(0)}(k) = u^{(l+1)}(k)$, $k = 0, \dots, N-1$, увеличить N на единицу и перейти к шагу 1, иначе увеличить l на единицу и перейти к шагу 2.

Обозначим через $\mathcal{U}^*$ множество всех оптимальных по быстродействию управлений для системы (1)–(2). В силу сделанных предположений выполняются включения $\mathcal{U}^* \subset \mathcal{U}_{N_{\min}} \subset \mathbb{R}^{nN_{\min}}$. Напомним, что расстоянием от точки до множества в пространстве $\mathbb{R}^{nN_{\min}}$ называется величина

$$\text{dist}(z, Z) := \inf_{\zeta \in Z} \|z - \zeta\|, \quad z \in \mathbb{R}^{nN_{\min}}, \quad Z \subset \mathbb{R}^{nN_{\min}}.$$

Теорема 5. Пусть $N_{\min} < \infty$. Тогда при любом значении $\varepsilon > 0$ построенный выше алгоритм завершает свою работу через конечное число итераций с ответом N_ε , u_ε . При этом $N_\varepsilon \leq N_{\min}$, а если число $\varepsilon > 0$ достаточно мало, то $N_\varepsilon = N_{\min}$. Более того, при таких значениях ε выполняется $u_\varepsilon \in \mathbb{R}^{nN_{\min}}$ и имеет место сходимость

$$\text{dist}(u_\varepsilon, \mathcal{U}^*) \rightarrow 0, \quad \varepsilon \rightarrow 0.$$

Таким образом, построенный алгоритм позволяет приближенно находить решения задачи быстродействия для системы (1)–(2).

7. Обсуждение и комментарии

Для реализации предложенного алгоритма необходимо определить способ решения двух типов конечномерных выпуклых задач оптимизации на шагах 3 и 6.

Задачи на шаге 3 по существу представляют собой задачи метрического проектирования заданных векторов на заданные выпуклые компактные множества в $\mathbb{R}^n$. В качестве последних выступают множества $\mathcal{A}_N(k+1)U(k)$, где все $U(k)$ известны заранее, а линейные преобразования $\mathcal{A}_N(k+1)$ вычисляются по известным матрицам системы $A(k)$ и зависят от текущих значений k и N . Таким образом, при любых значениях k, N множество $\mathcal{A}_N(k+1)U(k)$ можно считать заранее известным. При этом зачастую в приложениях $U(k)$ являются линейными преобразованиями некоторого базового множества U (например, U – единичный шар, а $U(k)$ – эллипсоид; U – единичный куб, а $U(k)$ – зонотоп и т.д.), и в этом случае $\mathcal{A}_N(k+1)U(k)$ имеют ту же самую структуру. Для многих таких множеств задача метрического проектирования крайне широко исследована, а для ее решения разработаны различные скоростные алгоритмы [16, 17]. Важно также отметить, что погрешности вычислений, связанные с тем или иным способом приближенного решения задач на шаге 3, не накапливаются при последующих итерациях, поскольку каждый раз заново выполняется нелокальное улучшение полученной ранее (неточной) программы управления. Заметим еще, что задачи оптимизации на шаге 3 можно переписать в терминах траекторий. А именно, вместо выполнения шагов 2 и 3 для каждого $k \in \{0, \dots, N-1\}$ могут быть последовательно решены следующие оптимизационные задачи:

$$x^{(l+1)}(k+1) \in \operatorname{Arg} \min_{x \in A(k)x^{(l+1)}(k) + U(k)} \|\mathcal{A}_N(k+1)(x - x^{(l)}(k+1)) + x^{(l)}(N)\|,$$

где для $k = 0$ имеем

$$x^{(l+1)}(0) = x_0.$$

В таком случае несколько сокращается количество промежуточных вычислений, необходимых для построения улучшенной траектории. При этом соответствующие значения управления определяются равенствами

$$u^{(l+1)}(k) = x^{(l+1)}(k+1) - A(k)x^{(l+1)}(k), \quad k = 0, \dots, N-1.$$

Более существенным является точное решение задач максимизации на шаге 6. Вычисление величины оценки $E_N^{(l+1)}$ с минимально возможной погрешностью важно для корректной проверки условия $E_N^{(l+1)} > 0$ на шаге 7, которое гарантирует, что текущее значение времени N строго меньше времени быстрогодействия $N_{\min}$. Наличие вычислительных погрешностей для $E_N^{(l+1)}$ может привести к тому, что время быстрогодействия будет найдено неверно даже в том случае, когда величина $\varepsilon > 0$ достаточно мала. В связи с этим актуально использование опорных функций ко множествам $U(k)$, которые могут

быть определены заранее, поскольку структура всех $U(k)$ предполагается известной. При таком подходе вместо решения задач максимизации линейной функции на выпуклом компактном множестве будет достаточно подсчитать значения известных опорных функций в заданных точках. Последнее позволяет повысить точность вычислений и заодно сократить общее время работы.

Предложенный в разделе 6 алгоритм может быть реализован в любой программной среде на произвольном языке программирования. Для демонстрации результатов была выполнена реализация в виде кода на свободно распространяемом высокоуровневом языке Python3. Для численного решения экстремальных задач на шагах 3 и 6 была использована обособленная библиотека выпуклой оптимизации *cvxpy* и методы *cvxpy.SCS* и *cvxpy.CLARABEL*. Отметим, что в том случае, когда множество U является зонотопом, целесообразно использовать метод *cvxpy.CLARABEL* как на шаге 6, так и на шаге 3. Подробная документация и исходные коды указанных методов могут быть найдены на официальной интернет-странице библиотеки <https://www.cvxpy.org/>.

Листинг программы отнесен в Приложение 2. Для подачи исходных данных из нижеследующих примеров на вход программы можно использовать код в формате json, приведенный в Приложении 3.

8. Примеры

Пример 1. Рассмотрим одномерную стационарную систему

$$x(k+1) = x(k) + u(k), \quad k = 0, 1, \dots, \quad x(0) = x_0 \in (2; 3]$$

с ограничением на управление $|u(k)| \leq 1$ и решим задачу быстродействия для этой системы, используя алгоритм из раздела 6.

Поскольку $n = 1$, то все экстремальные задачи на шаге 3 имеют единственное решение, при этом $\mathcal{A}_N(k) = 1$ при всех N и k . Будем считать $\varepsilon > 0$ настолько малым, что условие полной остановки на шаге 4 заменяется условием $|x^{(l+1)}(N)| = 0$.

Положим $N = 1$ и $u^{(0)}(0) = 0$. На шаге 2 имеем

$$\xi^{(0)}(0) = \xi^{(0)}(1) = 0.$$

На шаге 3 находим

$$u^{(1)}(0) = \arg \min_{u \in [-1; 1]} |u + x_0| = -1, \quad x^{(1)}(1) = x_0 - 1 \in (1; 2].$$

Проверка на шаге 4 показывает, что $|x^{(1)}(1)| > 0$, поэтому на шаге 5 получаем

$$\psi^{(1)}(0) = \psi^{(1)}(1) = -2x^{(1)}(1) = 2 - 2x_0 < 0.$$

Отсюда на шаге 6 вычисляем

$$E_1^{(1)} = |x^{(1)}(1)|^2 - \max_{u \in [-1; 1]} \psi^{(1)}(1)(u - u^{(1)}(0)) = |x^{(1)}(1)|^2 = (x_0 - 1)^2 > 0.$$

На шаге 7 фиксируем найденное $u^{(1)}(0) = -1$ и переходим к случаю $N = 2$.

При $N = 2$ имеем $u^{(0)}(0) = -1$, $u^{(0)}(1) = 0$. Далее последовательно получаем

$$\begin{aligned}\xi^{(0)}(1) &= \xi^{(0)}(2) = 0, \quad \xi^{(0)}(0) = 1; \\ u^{(1)}(0) &= \arg \min_{u \in [-1;1]} |u + x_0| = -1, \quad x^{(1)}(1) = x_0 - 1 \in (1; 2]; \\ u^{(1)}(1) &= \arg \min_{u \in [-1;1]} |u + x^{(1)}(1)| = -1, \quad x^{(1)}(2) = x_0 - 2 \in (0; 1].\end{aligned}$$

Поскольку $|x^{(1)}(2)| > 0$, находим

$$\begin{aligned}\psi^{(1)}(0) &= \psi^{(1)}(1) = \psi^{(1)}(2) = -2x^{(1)}(2) = 4 - 2x_0 < 0; \\ E_2^{(1)} &= |x^{(1)}(2)|^2 - \max_{u \in [-1;1]} \psi^{(1)}(1)(u - u^{(1)}(0)) - \max_{u \in [-1;1]} \psi^{(1)}(2)(u - u^{(1)}(1)) = \\ &= |x^{(1)}(2)|^2 = (x_0 - 2)^2 > 0.\end{aligned}$$

Фиксируем найденное $u^{(1)}(0) = u^{(1)}(1) = -1$ и переходим, наконец, к случаю $N = 3$.

При $N = 3$ имеем $u^{(0)}(0) = u^{(0)}(1) = -1$, $u^{(0)}(2) = 0$. Последовательно получаем

$$\begin{aligned}\xi^{(0)}(2) &= \xi^{(0)}(3) = 0, \quad \xi^{(0)}(0) = \xi^{(0)}(1) = 1; \\ u^{(1)}(0) &= \arg \min_{u \in [-1;1]} |u + x_0 - 1| = -1, \quad x^{(1)}(1) = x_0 - 1 \in (1; 2]; \\ u^{(1)}(1) &= \arg \min_{u \in [-1;1]} |u + x^{(1)}(1)| = -1, \quad x^{(1)}(2) = x_0 - 2 \in (0; 1]. \\ u^{(1)}(2) &= \arg \min_{u \in [-1;1]} |u + x^{(1)}(2)| = -x^{(1)}(2) = 2 - x_0, \quad x^{(1)}(3) = 0.\end{aligned}$$

Проверка на шаге 4 показывает, что работа алгоритма завершена и найдено решение задачи быстродействия: $N_{\min} = 3$, $u^*(0) = u^*(1) = -1$, $u^*(2) = 2 - x_0$, $x^*(1) = x_0 - 1$, $x^*(2) = x_0 - 2$, $x^*(3) = 0$.

Пример 2. Пусть в (1)–(2) $n = 2$, система стационарна и известно, что

$$A(k) \equiv \begin{pmatrix} \frac{4}{5}(\cos(1) + \sin(1)) & -\frac{8}{5}\sin(1) \\ \frac{4}{5}\sin(1) & \frac{4}{5}(\cos(1) - \sin(1)) \end{pmatrix};$$

$$U(k) \equiv \{v \in \mathbb{R}^2 \mid (2v_1 + v_2)v_1 + (v_1 + 3v_2)v_2 \leq 1\};$$

$$x_0 = (-37,79, -26,1); \quad \varepsilon = 0,0001.$$

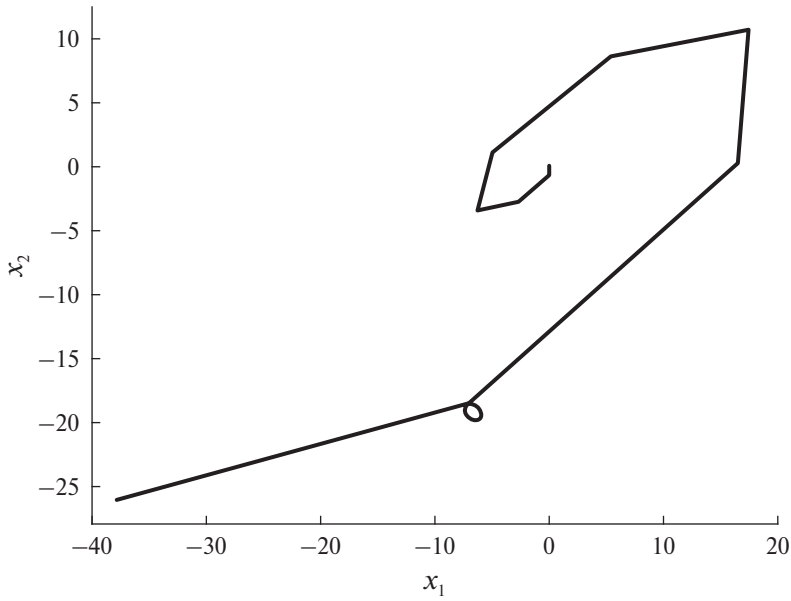


Рис. 1. Оптимальная траектория в примере 2.

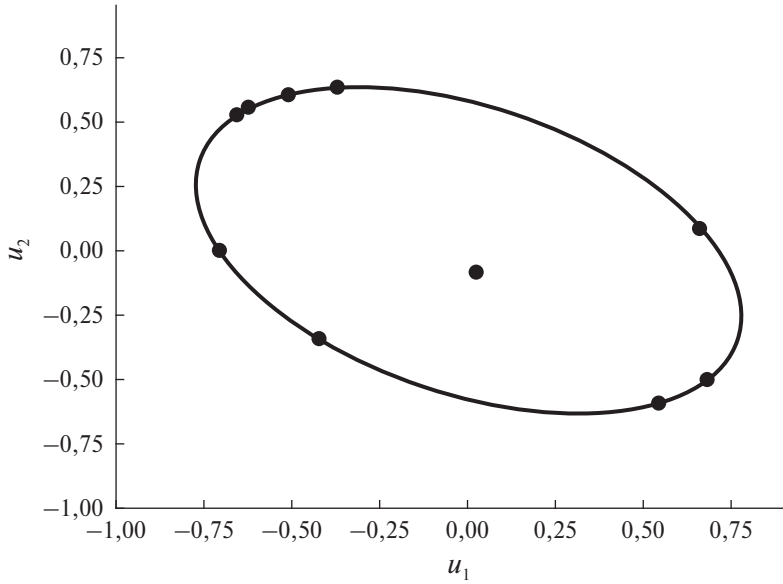


Рис. 2. Оптимальное управление в примере 2.

Применяя предложенный алгоритм, получим следующие результаты: $N_{\min} = 10$, оптимальный по быстродействию процесс имеет вид, представленный на рис. 1, 2. В табл. 1 приведены подробные сведения о сходимости алгоритма. В этой и последующих таблицах N и l соответствуют обозначениям в статье, $\|x(N)\|$ следует читать как $\|x^{(l+1)}(N)\|$, а $\sqrt{E}$ – как $\sqrt{\max\{E_N^{(l+1)}, 0\}}$.

Таблица 1. Сходимость алгоритма в примере 2

N	$l + 1$	$\ x(N)\ $	$\sqrt{E}$
1	1	19,72	19,72
2	1	16,24	16,24
3	1	20,4	20,4
4	1	8,91	8,9
5	1	4,58	4,54
6	1	7,07	7,07
7	1	2,52	2,51
8	3	0,55	0,25
9	1	0,22	0,19
10	1	10^{-5}	0

Пример 3. Пусть в системе (1)–(2) $n = 2$ и известно, что

$$A(k) \equiv \begin{pmatrix} \cos\left(\frac{\pi}{4}\right) & -\sin\left(\frac{\pi}{4}\right) \\ \sin\left(\frac{\pi}{4}\right) & \cos\left(\frac{\pi}{4}\right) \end{pmatrix};$$

$$x_0 = (9,33, 0,2); \quad \varepsilon = 0,0001.$$

Предположим, что множества $U(k)$ изменяются по следующему правилу:

$$U(k) = \left\{ v \in \mathbb{R}^2 \mid \max\{|v_1|, |v_2|\} \leq (0,9)^k \frac{\sqrt{3}}{2}, \quad \|v\| \leq \frac{3}{4} \right\}.$$

Применяя предложенный алгоритм, получим следующие результаты, представленные в табл. 2 и на рис. 3, 4.

Таблица 2. Сходимость алгоритма в примере 3

N	$l + 1$	$\ x(N)\ $	$\sqrt{E}$
1	1	8,58	8,58
2	1	7,83	7,83
3	1	7,08	7,08
4	1	6,45	6,45
5	1	5,7	5,7
...	...	...	...
28	1	0,13	0,13
29	1	0,07	0,07
30	1	0,03	0,03
31	1	10^{-6}	0

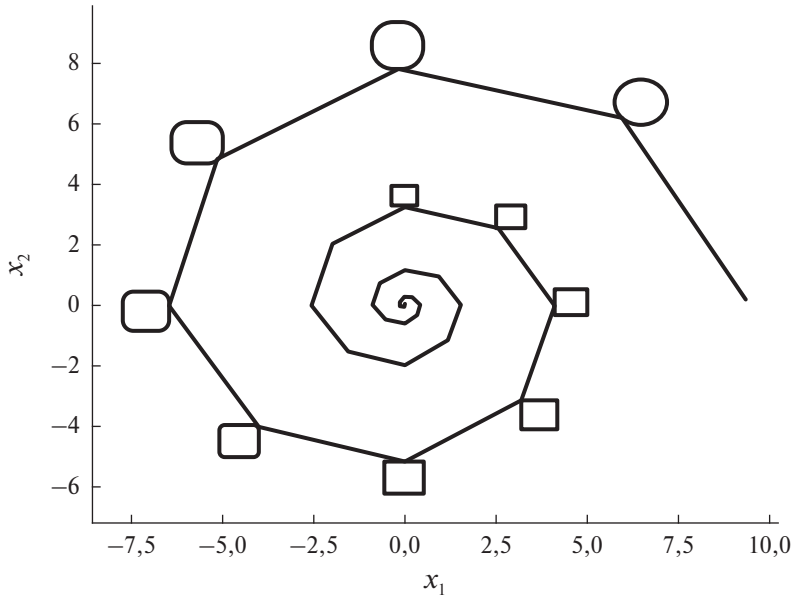


Рис. 3. Оптимальная траектория в примере 3.

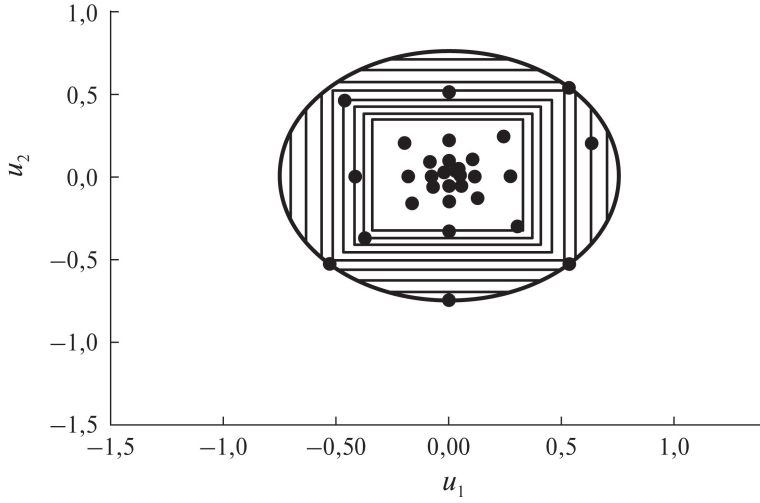


Рис. 4. Оптимальное управление в примере 3.

Пример 4. В условиях предыдущего примера предположим, что одновременно матрицы $A(k)$ и множества $U(k)$ изменяются по следующим правилам:

$$A(k) = (-1)^k \begin{pmatrix} \cos\left(\frac{\pi}{4}\right) & -\sin\left(\frac{\pi}{4}\right) \\ \sin\left(\frac{\pi}{4}\right) & \cos\left(\frac{\pi}{4}\right) \end{pmatrix};$$

$$U(k) = \left\{ v \in \mathbb{R}^2 \mid \max\{|v_1|, |v_2|\} \leq \frac{\sqrt{3}}{2}, \|v\| \leq (0,9)^k \right\}.$$

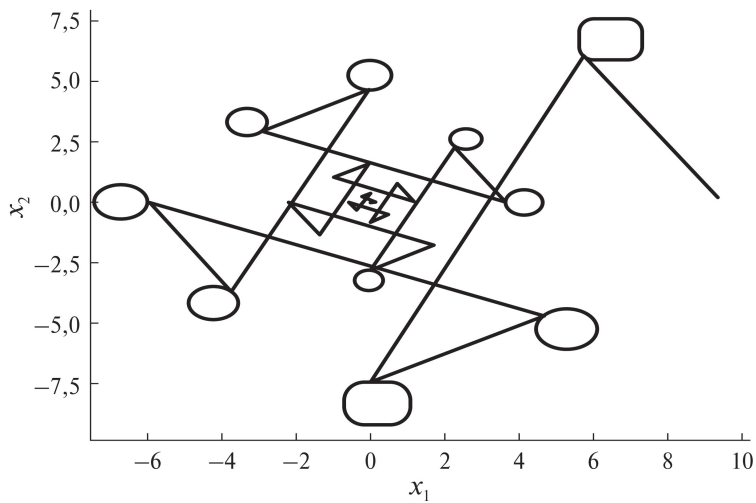


Рис. 5. Оптимальная траектория в примере 4.

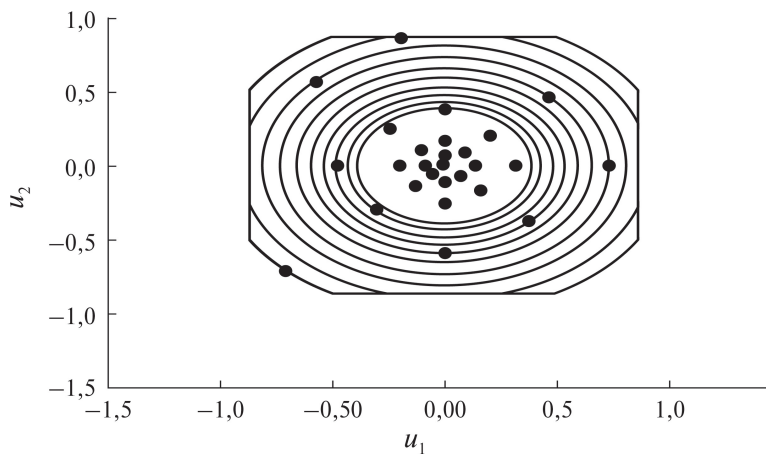


Рис. 6. Оптимальное управление в примере 4.

Значения x_0 и ε возьмем из примера 3. Применяя предложенный алгоритм, получим следующие результаты, представленные в табл. 3 и на рис. 5, 6.

Таблица 3. Сходимость алгоритма в примере 4

N	$l + 1$	$\ x(N)\ $	$\sqrt{E}$
1	1	8,33	8,33
2	1	7,46	7,46
3	1	6,65	6,65
4	1	5,92	5,92
5	1	5,27	5,27
...	...	...	...
24	1	0,16	0,16
25	1	0,08	0,08
26	1	0,1	0,1
27	1	10^{-5}	0

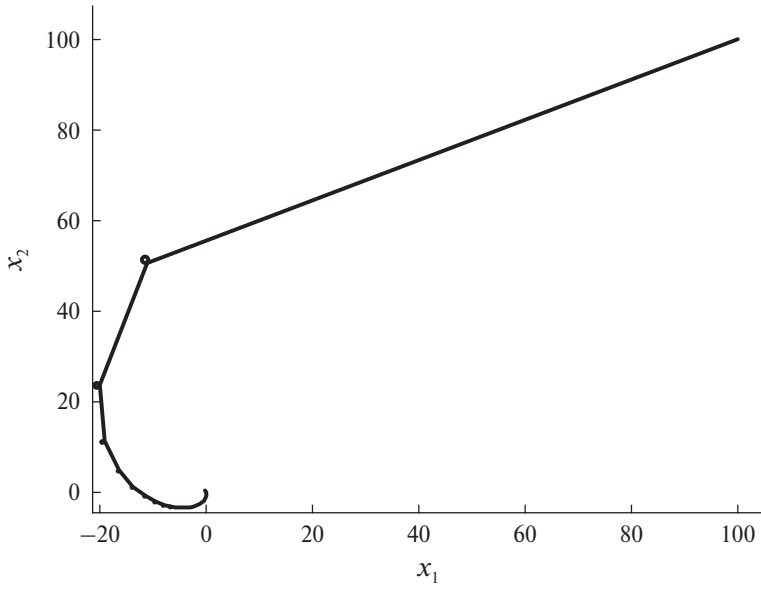


Рис. 7. Оптимальная траектория в примере 5.

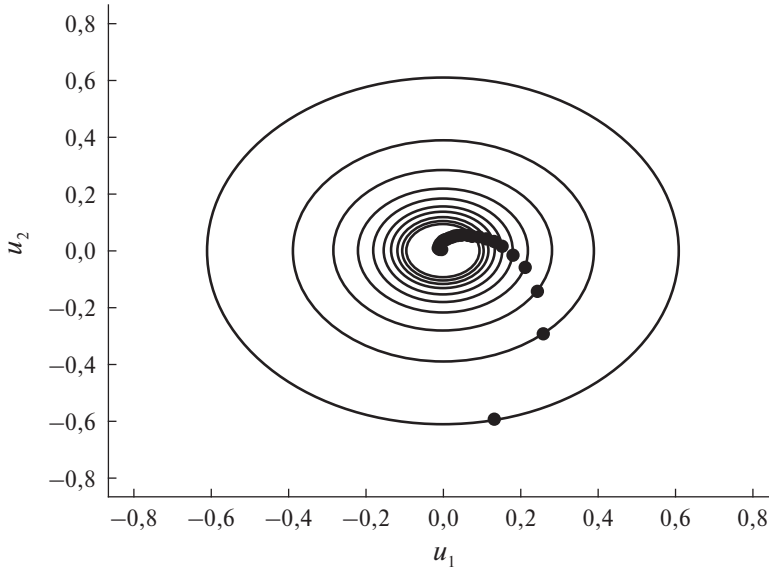


Рис. 8. Оптимальное управление в примере 5.

Пример 5. Рассмотрим следующий пример нестационарной системы (1)–(2). Пусть $n = 2$ и известно, что

$$A(k) = \begin{pmatrix} e^{-1/(k+1)} \cos(1/(k+1)) & -e^{-1/(k+1)} \sin(1/(k+1)) \\ e^{-1/(k+1)} \sin(1/(k+1)) & e^{-1/(k+1)} \cos(1/(k+1)) \end{pmatrix};$$

$$U(k) = \left\{ v \in \mathbb{R}^2 \mid 2v_1^2 + 2v_2^2 \leq 1 + e^{-2/(k+1)} - 2e^{-1/(k+1)} \cos(1/(k+1)) \right\};$$

$$x_0 = (100, 100); \quad \varepsilon = 0,0001.$$

Здесь матрицы $A(k)$ и множества $U(k)$ соответствуют системе вида (1)–(2), полученной путем дискретизации системы с непрерывным временем вида

$$\dot{z}(t) = A_c z(t) + w(t), \quad z(0) = x_0,$$

где $t \in [0; +\infty)$,

$$A_c = \begin{pmatrix} -1 & -1 \\ 1 & -1 \end{pmatrix},$$

функция управления $w(t)$ удовлетворяет геометрическому ограничению $\|w(t)\| \leq 1$, $t \geq 0$, и является кусочно-постоянной на интервалах времени $[t_k; t_{k+1})$, $k = 0, 1, \dots$, причем $t_0 = 0$ и шаг дискретизации $\Delta_k := t_{k+1} - t_k$ не является постоянным, а изменяется по правилу $\Delta_k = 1/(k+1)$.

Применяя предложенный алгоритм, получим следующие результаты, представленные в табл. 4 и на рис. 7, 8.

Таблица 4. Сходимость алгоритма в примере 5

N	$l+1$	$\ x(N)\ $	$\sqrt{E}$
1	1	51,42	51,42
2	1	30,78	30,78
3	1	21,79	21,79
4	1	16,75	16,75
5	1	13,53	13,53
...	...	...	...
77	1	0,03	0,03
78	1	0,02	0,02
79	1	0,01	0,01
80	1	10^{-5}	0

9. Заключение

Разработанный в настоящей статье алгоритм представляет собой метод последовательного глобального улучшения программы управления в задаче быстрогодействия для линейной дискретной системы общего вида. В отсутствие дополнительных предположений о задаче и о свойствах ее решения установлена сходимость к оптимальному по быстродействию процессу. Следует подчеркнуть, что несмотря на присутствие в структуре алгоритма перебора моментов времени возможного окончания процесса, информация, полученная на более ранних этапах, не теряется при переходе к следующему моменту времени, а используется для построения нового приближения.

Примеры показывают, что существуют задачи, в которых такой подход является более эффективным в сравнении с классическим переборно-оптимизационным подходом. Однако на текущий момент не доказано, что

алгоритм будет работать настолько же эффективно в общем случае. Более того, вычислительная практика показывает, что скорость сходимости напрямую зависит как от способа программной реализации, так и от исходных данных задачи. В то же время различные дополнительные сведения, которые могут возникнуть при решении конкретных прикладных задач (например, какие-либо оценки времени быстроедействия и/или известное приближение к оптимальному процессу), естественным образом встраиваются в структуру алгоритма и могут быть использованы для увеличения скорости сходимости к решению. Теоретическое исследование свойств сходимости разработанной итерационной процедуры представляется актуальным направлением для дальнейших изысканий.

Отметим еще, что результаты настоящей работы могут быть без особого труда обобщены на более содержательный с практической точки зрения случай нестационарных линейных систем вида $x(k+1) = A(k)x(k) + B(k)u(k)$, где вектор $u(k)$ имеет размерность m , причем, вообще говоря, $m \neq n$, а матрицы $B(k) \in \mathbb{R}^{n \times m}$ считаются известными.

ПРИЛОЖЕНИЕ 1

Лемма 1. Пусть $\hat{u} \in \mathcal{U}_N$ – произвольное управление и $\hat{x} \in \mathcal{X}_N$ найдено из (5). Пусть также $\hat{\psi}, \hat{\xi} \in \mathcal{X}_N$ определены формулами (6) и (9) соответственно. Тогда при каждом $k \in \{0, \dots, N\}$ имеет место равенство

$$(П.1) \quad \hat{\psi}(k) = 2\mathcal{A}_N(k)^T \left(\hat{\xi}(k) - \mathcal{A}_N(k)\hat{x}(k) \right).$$

Доказательство. Действительно, при $k = N$ равенство (П.1) вытекает из начальных условий в правых частях формул (6) и (9), с учетом $\mathcal{A}_N(N) = I$. Предположим, что (П.1) выполняется для некоторого $k \in \{1, \dots, N\}$. Тогда

$$\begin{aligned} \hat{\psi}(k-1) &\stackrel{(6)}{=} A(k-1)^T \hat{\psi}(k) \stackrel{(П.1)}{=} 2\mathcal{A}_N(k-1)^T \left(\hat{\xi}(k) - \mathcal{A}_N(k)\hat{x}(k) \right) \stackrel{(5)}{=} \\ &\stackrel{(5)}{=} 2\mathcal{A}_N(k-1)^T \left(\hat{\xi}(k) - \mathcal{A}_N(k-1)\hat{x}(k-1) - \mathcal{A}_N(k)\hat{u}(k-1) \right) \stackrel{(9)}{=} \\ &\stackrel{(9)}{=} 2\mathcal{A}_N(k-1)^T \left(\hat{\xi}(k-1) - \mathcal{A}_N(k-1)\hat{x}(k-1) \right). \end{aligned}$$

Следовательно, (П.1) имеет место при всех $k \in \{0, \dots, N\}$. Лемма доказана.

Доказательство теоремы 1. Поскольку соотношения (4)–(6) являются необходимым и достаточным условием оптимальности в задаче (1)–(3), то достаточно установить их эквивалентность соотношениям (5), (7) и (9). Соотношение (5) присутствует в обоих наборах. При этом для фиксированного $\hat{u} \in \mathcal{U}_N$ оно однозначно определяет траекторию $\hat{x}$, а соотношения (6) и (9) однозначно определяют значения двойственных переменных $\hat{\psi}(k)$ и $\hat{\xi}(k)$, которые связаны формулой (П.1) в силу леммы 1. Таким образом, остается проверить эквивалентность условий (4) и (7).

Условие (7) эквивалентно условию

$$\hat{u}(k) \in \text{Arg} \min_{u \in U(k)} \left(\|\mathcal{A}_N(k+1)u\|^2 + 2\langle \mathcal{A}_N(k+1)u, \mathcal{A}_N(k)\hat{x}(k) - \hat{\xi}(k+1) \rangle \right).$$

С учетом (5) это также эквивалентно

$$\begin{aligned} \hat{u}(k) \in \text{Arg} \min_{u \in U(k)} & \left(\|\mathcal{A}_N(k+1)u\|^2 + \right. \\ & \left. + 2\langle \mathcal{A}_N(k+1)u, \mathcal{A}_N(k+1)(\hat{x}(k+1) - \hat{u}(k)) - \hat{\xi}(k+1) \rangle \right), \end{aligned}$$

а в силу (П.1) эквивалентно

$$\hat{u}(k) \in \text{Arg} \min_{u \in U(k)} \left(\|\mathcal{A}_N(k+1)u\|^2 - \langle u, \hat{\psi}(k+1) + 2\mathcal{A}_N(k+1)^T \mathcal{A}_N(k+1)\hat{u}(k) \rangle \right).$$

Поэтому условие (7) эквивалентно условию

$$\begin{aligned} & \langle \hat{\psi}(k+1), \hat{u}(k) \rangle + \|\mathcal{A}_N(k+1)\hat{u}(k)\|^2 \geq \\ & \geq \langle \hat{\psi}(k+1), u \rangle + 2\langle \mathcal{A}_N(k+1)^T \mathcal{A}_N(k+1)\hat{u}(k), u \rangle - \|\mathcal{A}_N(k+1)u\|^2 \quad \forall u \in U(k) \end{aligned}$$

или, что то же самое,

$$\langle \hat{\psi}(k+1), \hat{u}(k) - u \rangle + \|\mathcal{A}_N(k+1)(\hat{u}(k) - u)\|^2 \geq 0 \quad \forall u \in U(k).$$

Ясно, что последнее выполняется, если имеет место (4). Предположим, что (4) не выполняется. Тогда существует $u' \in U(k)$, для которого

$$\langle \hat{\psi}(k+1), \hat{u}(k) - u' \rangle < 0.$$

Но тогда и для $v_\varepsilon = (1 - \varepsilon)\hat{u}(k) + \varepsilon u'$ при $\varepsilon > 0$ выполняется

$$\langle \hat{\psi}(k+1), \hat{u}(k) - v_\varepsilon \rangle = \varepsilon \langle \hat{\psi}(k+1), \hat{u}(k) - u' \rangle < 0.$$

При этом для значений $\varepsilon \in (0; 1]$ имеет место $v_\varepsilon \in U(k)$, поскольку множество $U(k)$ выпукло. Кроме того, $v_\varepsilon \rightarrow \hat{u}(k)$ при $\varepsilon \rightarrow 0$. Поскольку

$$\begin{aligned} & \langle \hat{\psi}(k+1), \hat{u}(k) - v_\varepsilon \rangle + \|\mathcal{A}_N(k+1)(\hat{u}(k) - v_\varepsilon)\|^2 = \\ & = \varepsilon \langle \hat{\psi}(k+1), \hat{u}(k) - u' \rangle + \varepsilon^2 \|\mathcal{A}_N(k+1)(\hat{u}(k) - u')\|^2 \leq \\ & \leq \varepsilon \left(\langle \hat{\psi}(k+1), \hat{u}(k) - u' \rangle + \varepsilon \|\mathcal{A}_N(k+1)\|^2 \|\hat{u}(k) - u'\|^2 \right), \end{aligned}$$

то отсюда следует, что при достаточно малом значении $\varepsilon > 0$ будет справедливо

$$\langle \hat{\psi}(k+1), \hat{u}(k) - v_\varepsilon \rangle + \|\mathcal{A}_N(k+1)(\hat{u}(k) - v_\varepsilon)\|^2 < 0,$$

где $v_\varepsilon \in U(k)$, так что условие (7) в этом случае также не выполняется. Окончательно устанавливаем, что условия (4) и (7) эквивалентны. Теорема доказана.

Для доказательства теорем 2, 3 и 5 понадобятся следующие конструкции (см. также [9, 10, 14]). Зафиксируем $\hat{u} \in \mathcal{U}_N$, определим значения $\hat{\xi}(k)$ из уравнения (9) и положим для произвольных $k \in \{0, \dots, N-1\}$ и $x, u \in \mathbb{R}^n$

$$\begin{aligned}\hat{\varphi}(k, x) &= 2\langle \mathcal{A}_N(k)^T \hat{\xi}(k), x \rangle - \|\mathcal{A}_N(k)x\|^2, \\ \hat{R}(k, x, u) &= \hat{\varphi}(k+1, A(k)x + u) - \hat{\varphi}(k, x).\end{aligned}$$

Лемма 2. Пусть $\hat{u} \in \mathcal{U}_N$ – произвольное управление, а $\hat{x}, \hat{\xi} \in \mathcal{X}_N$ – соответствующие ему решения уравнений (5) и (9). Тогда условие (10) выполняется в том и только том случае, если

$$(П.2) \quad \hat{R}(k, \tilde{x}(k), \tilde{u}(k)) = \max_{u \in U(k)} \hat{R}(k, \tilde{x}(k), u) \quad \forall k \in \{0, \dots, N-1\},$$

где $\tilde{x}$ удовлетворяет (11). В частности, управление $\hat{u}$ оптимально в точности тогда, когда имеет место

$$(П.3) \quad \hat{R}(k, \hat{x}(k), \hat{u}(k)) = \max_{u \in U(k)} \hat{R}(k, \hat{x}(k), u) \quad \forall k \in \{0, \dots, N-1\}.$$

Доказательство. По определению имеем

$$\begin{aligned}\hat{R}(k, \tilde{x}(k), u) &= \hat{\varphi}(k+1, A(k)\tilde{x}(k) + u) - \hat{\varphi}(k, \tilde{x}(k)) = \\ &= 2\langle \mathcal{A}_N(k+1)^T \hat{\xi}(k+1), A(k)\tilde{x}(k) + u \rangle - \\ &\quad - \|\mathcal{A}_N(k+1)(A(k)\tilde{x}(k) + u)\|^2 - \hat{\varphi}(k, \tilde{x}(k)) = \\ &= -\|\mathcal{A}_N(k)\tilde{x}(k) + \mathcal{A}_N(k+1)u\|^2 + 2\langle \hat{\xi}(k+1), \mathcal{A}_N(k)\tilde{x}(k) + \mathcal{A}_N(k+1)u \rangle - \\ &\quad - \hat{\varphi}(k, \tilde{x}(k)) = \\ &= -\|\mathcal{A}_N(k)\tilde{x}(k) + \mathcal{A}_N(k+1)u - \hat{\xi}(k+1)\|^2 + \|\hat{\xi}(k+1)\|^2 - \hat{\varphi}(k, \tilde{x}(k)).\end{aligned}$$

Поскольку второе и третье из полученных слагаемых не зависят от u , приходим к первому из доказываемых утверждений. Второе утверждение непосредственно следует из первого и теоремы 1. Лемма доказана.

Доказательство теоремы 2. В силу введенных обозначений имеем

$$\begin{aligned}J_N(\tilde{u}) &= -\hat{\varphi}(N, \tilde{x}(N)) = -\hat{\varphi}(0, x_0) + \hat{\varphi}(0, x_0) - \hat{\varphi}(N, \tilde{x}(N)) = \\ &= -\hat{\varphi}(0, x_0) - \sum_{k=0}^{N-1} \left(\hat{\varphi}(k+1, \tilde{x}(k+1)) - \hat{\varphi}(k, \tilde{x}(k)) \right) = \\ &= -\hat{\varphi}(0, x_0) - \sum_{k=0}^{N-1} \hat{R}(k, \tilde{x}(k), \tilde{u}(k))\end{aligned}$$

и, аналогично,

$$J_N(\hat{u}) = -\hat{\varphi}(0, x_0) - \sum_{k=0}^{N-1} \hat{R}(k, \hat{x}(k), \hat{u}(k)).$$

Поскольку согласно (9)

$$\begin{aligned}\hat{R}(k, x, \hat{u}(k)) &= \hat{\varphi}(k+1, A(k)x + \hat{u}(k)) - \hat{\varphi}(k, x) = \\ &= 2\langle \mathcal{A}_N(k+1)^T \hat{\xi}(k+1), A(k)x + \hat{u}(k) \rangle - \|\mathcal{A}_N(k+1)(A(k)x + \hat{u}(k))\|^2 - \\ &- 2\langle \mathcal{A}_N(k)^T \hat{\xi}(k), x \rangle + \|\mathcal{A}_N(k)x\|^2 \equiv 2\langle \mathcal{A}_N(k+1)^T \hat{\xi}(k), \hat{u}(k) \rangle + \|\mathcal{A}_N(k+1)\hat{u}(k)\|^2\end{aligned}$$

не зависит от x , то в силу (10) и леммы 2 при каждом $k \in \{0, \dots, N-1\}$ выполняется

$$(П.4) \quad \hat{R}(k, \tilde{x}(k), \tilde{u}(k)) \geq \hat{R}(k, \tilde{x}(k), \hat{u}(k)) = \hat{R}(k, \hat{x}(k), \hat{u}(k)).$$

Следовательно, $J_N(\tilde{u}) \leq J_N(\hat{u})$. Теорема доказана.

Проведем еще одно общее построение, используемое ниже при доказательстве теорем 3 и 5.

Рассмотрим вспомогательную систему

$$(П.5) \quad y(k+1) = y(k) + v(k), \quad k = 0, \dots, N-1, \quad y(0) = y_0 := \mathcal{A}_N(0)x_0$$

с геометрическими ограничениями $v(k) \in V(k) := \mathcal{A}_N(k+1)U(k)$. Ясно, что все множества $V(k)$ выпуклы, компактны и содержат начало координат. Пусть $\hat{u} \in \mathcal{U}_N$ задано. Положим в системе (П.5) $v(k) = \hat{v}(k) = \mathcal{A}_N(k+1)\hat{u}(k)$. Тогда поскольку $\hat{u}(k) \in U(k)$, то $\hat{v}(k) \in V(k)$ и решение $\hat{y}$ системы (П.5) удовлетворяет соотношению

$$\hat{y}(k) = \mathcal{A}_N(k)\hat{x}(k), \quad k = 0, \dots, N.$$

Запишем для системы (П.5) конструкции, аналогичные построенным выше. А именно, положим для $k \in \{0, \dots, N-1\}$ и $y, v \in \mathbb{R}^n$

$$\begin{aligned}\hat{\phi}(k, y) &= 2\langle \hat{\xi}(k), y \rangle - \|y\|^2, \\ \hat{\mathcal{R}}(k, y, v) &= \hat{\phi}(k+1, y+v) - \hat{\phi}(k, y),\end{aligned}$$

где значения $\hat{\xi}(k)$ найдены из (9). Тогда выполняется следующее соотношение:

$$(П.6) \quad \hat{\mathcal{R}}(k, \mathcal{A}_N(k)x, \mathcal{A}_N(k+1)u) = \hat{R}(k, x, u) \quad \forall x, u \in \mathbb{R}^n, \quad k = 0, \dots, N-1.$$

Действительно,

$$\begin{aligned}\hat{\mathcal{R}}(k, \mathcal{A}_N(k)x, \mathcal{A}_N(k+1)u) &= \\ &= \hat{\phi}(k+1, \mathcal{A}_N(k)x + \mathcal{A}_N(k+1)u) - \hat{\phi}(k, \mathcal{A}_N(k)x) = \\ &= 2\langle \hat{\xi}(k+1), \mathcal{A}_N(k)x + \mathcal{A}_N(k+1)u \rangle - \|(\mathcal{A}_N(k)x + \mathcal{A}_N(k+1)u)\|^2 - \\ &- 2\langle \hat{\xi}(k), \mathcal{A}_N(k)x \rangle + \|\mathcal{A}_N(k)x\|^2 = 2\langle \mathcal{A}_N(k+1)^T \hat{\xi}(k+1), A(k)x + u \rangle - \\ &- \|\mathcal{A}_N(k+1)(A(k)x + u)\|^2 - 2\langle \mathcal{A}_N(k)^T \hat{\xi}(k), x \rangle + \|\mathcal{A}_N(k)x\|^2 = \\ &= \hat{\varphi}(k+1, A(k)x + u) - \hat{\varphi}(k, x) = \hat{R}(k, x, u).\end{aligned}$$

Лемма 3. Пусть $\hat{u} \in \mathcal{U}_N$ и значения $\hat{\xi}(k)$ найдены из (9). Тогда существует единственная пара $(\tilde{y}, \tilde{v})$, удовлетворяющая условиям

$$(П.7) \quad \hat{\mathcal{R}}(k, \tilde{y}(k), \tilde{v}(k)) = \max_{v \in V(k)} \hat{\mathcal{R}}(k, \tilde{y}(k), v), \quad k = 0, \dots, N-1,$$

$$(П.8) \quad \tilde{y}(k+1) = \tilde{y}(k) + \tilde{v}(k), \quad k = 0, \dots, N-1, \quad \tilde{y}(0) = y_0.$$

При этом условие (П.7) эквивалентно условию

$$(П.9) \quad \tilde{v}(k) \in \text{Arg} \min_{v \in V(k)} \|v + \tilde{y}(k) - \hat{\xi}(k+1)\| \quad \forall k \in \{0, \dots, N-1\}$$

и для любой пары $(\tilde{x}, \tilde{u})$, удовлетворяющей соотношениям (10)–(11), выполняется

$$(П.10) \quad \begin{aligned} \tilde{y}(k) &= \mathcal{A}_N(k) \tilde{x}(k), \quad k = 0, \dots, N, \\ \tilde{v}(k) &= \mathcal{A}_N(k+1) \tilde{u}(k), \quad k = 0, \dots, N-1. \end{aligned}$$

Доказательство. Рассмотрим условие (П.7). По определению имеем

$$\begin{aligned} \hat{\mathcal{R}}(k, \tilde{y}(k), v) &= \hat{\phi}(k+1, \tilde{y}(k) + v) - \hat{\phi}(k, \tilde{y}(k)) = \\ &= 2\langle \hat{\xi}(k+1), \tilde{y}(k) + v \rangle - \|(\tilde{y}(k) + v)\|^2 - \hat{\phi}(k, \tilde{y}(k)) = \\ &= -\|\tilde{y}(k) + v\|^2 + 2\langle \hat{\xi}(k+1), \tilde{y}(k) + v \rangle - \hat{\phi}(k, \tilde{y}(k)) = \\ &= -\|\tilde{y}(k) + v - \hat{\xi}(k+1)\|^2 + \|\hat{\xi}(k+1)\|^2 - \hat{\phi}(k, \tilde{y}(k)). \end{aligned}$$

Как и при доказательстве леммы 2, находим, что (П.7) эквивалентно (П.9). В случае $k=0$ условие (П.9) означает, что значение $\tilde{v}(0)$ находится путем решения экстремальной задачи

$$\|v + \mathcal{A}_N(0)x_0 - \hat{\xi}(1)\| \rightarrow \min_{v \in V(0)}.$$

Ясно, что для непустого выпуклого компактного множества $V(0)$ решение этой задачи существует и единственно. Кроме того, в силу (П.8) отсюда однозначно определено значение $\tilde{y}(1)$. В то же время любое $\tilde{u}(0)$, удовлетворяющее (10), является одним из решений задачи

$$\|\mathcal{A}_N(1)u + \mathcal{A}_N(0)x_0 - \hat{\xi}(1)\| \rightarrow \min_{u \in U(0)},$$

откуда $\tilde{v}(0) = \mathcal{A}_N(1)\tilde{u}(0)$, поскольку множества $U(0)$ и $V(0)$ связаны соотношением $V(0) = \mathcal{A}_N(1)U(0)$. Для $\tilde{x}(1)$ из (11) получаем $\tilde{y}(1) = \mathcal{A}_N(1)\tilde{x}(1)$, поскольку выполняется $y_0 = \mathcal{A}_N(0)x_0$.

Пусть для некоторого $k \in \{1, \dots, N-1\}$ значения $\tilde{v}(k-1)$ и $\tilde{y}(k)$ уже известны. Тогда $\tilde{v}(k)$ определяется как единственное решение задачи

$$\|v + \tilde{y}(k) - \hat{\xi}(k+1)\| \rightarrow \min_{v \in V(k)},$$

где $V(k)$ непусто, выпукло и компактно. При известных $\tilde{v}(k)$ и $\tilde{y}(k)$ значение $\tilde{y}(k+1)$ однозначно определяется из (П.8). Таким образом, пара $(\tilde{y}, \tilde{v})$, удовлетворяющая условиям (П.7) и (П.8), полностью определена единственным образом.

Проводя аналогичные приведенным выше сопоставления значений $\tilde{v}(k)$ и $\tilde{u}(k)$, $\tilde{y}(k+1)$ и $\tilde{x}(k+1)$ при $k=1, \dots, N-1$ с учетом $V(k) = \mathcal{A}_N(k+1)U(k)$, устанавливаем справедливость (П.10). Лемма доказана.

Доказательство теоремы 3. Предположим, что управление $\hat{u}$ не является оптимальным в задаче (1)–(3). Тогда в силу леммы 2 найдется $r \in \{0, \dots, N-1\}$ такое, что

$$\hat{R}(r, \hat{x}(r), \hat{u}(r)) < \max_{u \in U(r)} \hat{R}(r, \hat{x}(r), u).$$

Возьмем наименьшее такое r . Тогда в силу (П.6) для любого $k \in \{0, \dots, r-1\}$ выполняется

$$\begin{aligned} \hat{\mathcal{R}}(k, \hat{y}(k), \hat{v}(k)) &= \hat{R}(k, \hat{x}(k), \hat{u}(k)) = \\ &= \max_{u \in U(k)} \hat{R}(k, \hat{x}(k), u) = \max_{v \in V(k)} \hat{\mathcal{R}}(k, \hat{y}(k), v), \end{aligned}$$

где

$$\hat{y}(k) = \mathcal{A}_N(k)\hat{x}(k), \quad \hat{v}(k) = \mathcal{A}_N(k+1)\hat{u}(k).$$

Поскольку в силу леммы 3 существует единственная пара $(\tilde{y}, \tilde{v})$, удовлетворяющая условиям (П.7) и (П.8), то имеет место

$$\tilde{v}(k) = \hat{v}(k), \quad k = 0, \dots, r-1,$$

и, следовательно, $\tilde{y}(r) = \hat{y}(r)$. Отсюда в силу (П.6) и (П.10) для любой пары $(\tilde{x}, \tilde{u})$, удовлетворяющей (10)–(11), справедливо

$$\begin{aligned} \hat{R}(r, \hat{x}(r), \hat{u}(r)) &< \max_{u \in U(r)} \hat{R}(r, \hat{x}(r), u) = \max_{v \in V(r)} \hat{\mathcal{R}}(r, \hat{y}(r), v) = \\ &= \max_{v \in V(r)} \hat{\mathcal{R}}(r, \tilde{y}(r), v) = \max_{u \in U(r)} \hat{R}(r, \tilde{x}(r), u) = \hat{R}(r, \tilde{x}(r), \tilde{u}(r)). \end{aligned}$$

Возвращаясь теперь к доказательству теоремы 2, находим, что при $k=r$ неравенство в (П.4) является строгим и поэтому

$$J_N(\tilde{u}) < J_N(\hat{u}).$$

Если же управление $\hat{u}$ оптимально в задаче (1)–(3), то согласно лемме 2 и (П.6) при всех $k \in \{0, \dots, N-1\}$ имеет место

$$\begin{aligned} \hat{\mathcal{R}}(k, \hat{y}(k), \hat{v}(k)) &= \hat{R}(k, \hat{x}(k), \hat{u}(k)) = \\ &= \max_{u \in U(k)} \hat{R}(k, \hat{x}(k), u) = \max_{v \in V(k)} \hat{\mathcal{R}}(k, \hat{y}(k), v), \end{aligned}$$

а значит, пара $(\tilde{y}, \tilde{v}) = (\hat{y}, \hat{v})$ удовлетворяет условиям (П.7)–(П.8). Следовательно, в силу леммы 3 и (П.6) для произвольной пары $(\tilde{x}, \tilde{u})$, удовлетворяющей (10)–(11), выполняется

$$\hat{R}(k, \tilde{x}(k), \tilde{u}(k)) = \hat{\mathcal{R}}(k, \hat{y}(k), \hat{v}(k)) = \hat{R}(k, \hat{x}(k), \hat{u}(k)) \quad \forall k \in \{0, \dots, N-1\}.$$

Но тогда, возвращаясь к доказательству теоремы 2, находим, что все неравенства в (П.4) при значениях $k = 0, \dots, N-1$ выполняются в виде равенств и, следовательно, $J_N(\tilde{u}) = J_N(\hat{u})$. Теорема доказана.

Доказательство теоремы 4. Пусть $\hat{u} \in \mathcal{U}_N$ – произвольное управление, а значения $\hat{x}(k)$ и $\hat{\psi}(k)$ определены из (5) и (6). Рассмотрим функцию $\hat{L} : \mathbb{R}^n \times \mathcal{U}_N \rightarrow \mathbb{R}$ вида

$$\hat{L}(x, u) = \|x\|^2 - 2\langle \hat{x}(N), x \rangle - \langle \hat{\psi}(0), x_0 \rangle - \sum_{k=0}^{N-1} \langle \hat{\psi}(k+1), u(k) \rangle.$$

Подчеркнем, что в этой записи x – вектор из $\mathbb{R}^n$, а u – элемент множества $\mathcal{U}_N$. Покажем, что для вектора $x(N)$, найденного из (1)–(2) при фиксированном $u \in \mathcal{U}_N$, выполняется

$$\hat{L}(x(N), u) = \|x(N)\|^2 = J_N(u).$$

Действительно, в силу (1) и (6) получаем

$$\begin{aligned} \hat{L}(x(N), u) &= \|x(N)\|^2 - 2\langle \hat{x}(N), x(N) \rangle - \langle \hat{\psi}(0), x_0 \rangle - \sum_{k=0}^{N-1} \langle \hat{\psi}(k+1), u(k) \rangle = \\ &= \|x(N)\|^2 + \langle \hat{\psi}(N), x(N) \rangle - \langle \hat{\psi}(0), x_0 \rangle - \\ &\quad - \sum_{k=0}^{N-1} \left(\langle \hat{\psi}(k+1), A(k)x(k) + u(k) \rangle - \langle \hat{\psi}(k+1), A(k)x(k) \rangle \right) = \\ &= \|x(N)\|^2 + \langle \hat{\psi}(N), x(N) \rangle - \langle \hat{\psi}(0), x_0 \rangle - \\ &\quad - \sum_{k=0}^{N-1} \left(\langle \hat{\psi}(k+1), x(k+1) \rangle - \langle \hat{\psi}(k), x(k) \rangle \right) = \|x(N)\|^2. \end{aligned}$$

В частности, для любого оптимального процесса (x^*, u^*) в задаче (3) имеет место

$$\hat{L}(x^*(N), u^*) = J_N^* := J_N(u^*).$$

С другой стороны, в точке своего глобального минимума функция $\hat{L}$ принимает значение

$$\hat{L}^* = \min_{\substack{x \in \mathbb{R}^n \\ u \in \mathcal{U}_N}} \hat{L}(x, u) = -\|\hat{x}(N)\|^2 - \langle \hat{\psi}(0), x_0 \rangle - \sum_{k=0}^{N-1} \max_{u \in U(k)} \langle \hat{\psi}(k+1), u \rangle.$$

Следовательно, в силу (5) имеет место неравенство

$$\begin{aligned}
J_N^* = \hat{L}(x^*(N), u^*) &\geq \hat{L}^* = -\|\hat{x}(N)\|^2 - \langle \hat{\psi}(0), x_0 \rangle - \sum_{k=0}^{N-1} \max_{u \in U(k)} \langle \hat{\psi}(k+1), u \rangle = \\
&= \|\hat{x}(N)\|^2 - 2\|\hat{x}(N)\|^2 - \langle \hat{\psi}(0), x_0 \rangle - \\
&\quad - \sum_{k=0}^{N-1} \max_{u \in U(k)} \left(\langle \hat{\psi}(k+1), A(k)\hat{x}(k) + u \rangle - \langle \hat{\psi}(k+1), A(k)\hat{x}(k) \rangle \right) = \\
&= \|\hat{x}(N)\|^2 + \langle \hat{\psi}(N), \hat{x}(N) \rangle - \langle \hat{\psi}(0), x_0 \rangle - \\
&\quad - \sum_{k=0}^{N-1} \max_{u \in U(k)} \left(\langle \hat{\psi}(k+1), A(k)\hat{x}(k) + u \rangle - \langle \hat{\psi}(k), \hat{x}(k) \rangle \right) = \\
&= \|\hat{x}(N)\|^2 - \sum_{k=0}^{N-1} \max_{u \in U(k)} \left(\langle \hat{\psi}(k+1), A(k)\hat{x}(k) + u \rangle - \langle \hat{\psi}(k+1), \hat{x}(k+1) \rangle \right) = \\
&= \|\hat{x}(N)\|^2 - \sum_{k=0}^{N-1} \max_{u \in U(k)} \langle \hat{\psi}(k+1), u - \hat{u}(k) \rangle.
\end{aligned}$$

Теорема доказана.

Сделаем еще одно построение, необходимое для доказательства теоремы 5. Пусть $N > 0$ задано. Для произвольного $\hat{u} \in \mathcal{U}_N$ определим два числа $f_I(\hat{u})$ и $f_E(\hat{u})$ следующим образом. Положим

$$f_I(\hat{u}) = \|\tilde{y}(N)\|^2,$$

где $\tilde{y}$ определяется условиями (П.9) и (П.8), в которых $\hat{\xi}$ найдено из (9). Кроме того, положим

$$f_E(\hat{u}) = \|\hat{x}(N)\|^2 - \sum_{k=0}^{N-1} \max_{u \in U(k)} \langle \hat{\psi}(k+1), u - \hat{u}(k) \rangle,$$

где $\hat{x}$ и $\hat{\psi}$ – решения уравнений (5) и (6).

Лемма 4. Функции $f_I, f_E : \mathcal{U}_N \rightarrow \mathbb{R}$ однозначно определены и непрерывны. При этом

$$f_I(\hat{u}) = J_N(\tilde{u})$$

для любого $\tilde{u}$, удовлетворяющего условиям (10)–(11), и каждое из двух равенств

$$f_I(\hat{u}) = J_N(\hat{u}) = f_E(\hat{u})$$

имеет место в точности тогда, когда $\hat{u}$ является оптимальным управлением в задаче (1)–(3).

Доказательство. Однозначная определенность функции f_I вытекает из леммы 3. Функция f_E корректно определена в силу компактности всех множеств $U(k)$.

Решение $\hat{\xi}$ уравнения (9) непрерывно зависит от параметров $\hat{u}(k)$, а решение $\tilde{y}$ уравнения (П.8) непрерывно зависит от параметров $\tilde{v}(k)$. Кроме того, при каждом $k \in \{0, \dots, N-1\}$ значение $\tilde{v}(k)$ определяется условием (П.9) как метрическая проекция точки $\hat{\xi}(k+1) - \tilde{y}(k)$ на непустое выпуклое компактное множество $V(k)$ в пространстве $\mathbb{R}^n$. Как известно, оператор метрической проекции на выпуклое и замкнутое множество в конечномерном евклидовом пространстве однозначно определен и непрерывен. Следовательно, функция f_I непрерывна. Пусть $\tilde{u} \in \mathcal{U}_N$ и $\tilde{x} \in \mathcal{X}_N$ удовлетворяют условиям (10)–(11). Тогда в силу леммы 3 имеет место

$$f_I(\hat{u}) = \|\tilde{y}(N)\|^2 = \|\mathcal{A}_N(N)\tilde{x}(N)\|^2 = \|\tilde{x}(N)\|^2 = J_N(\tilde{u}).$$

В частности, согласно теореме 3 равенство $f_I(\hat{u}) = J_N(\hat{u})$ выполняется тогда и только тогда, когда $\hat{u}$ – оптимальное управление в задаче (1)–(3).

Решение уравнения (5) непрерывно зависит от параметров $\hat{u}(k)$, решение уравнения (6) непрерывно зависит от параметров $\hat{x}(k)$, а величина $\max_{u \in U(k)} \langle \hat{\psi}(k+1), u \rangle$ представляет собой значение опорной функции к непустому компактному множеству $U(k)$ в точке $\hat{\psi}(k+1)$, которая также непрерывна. Следовательно, функция f_E непрерывна. При этом, если $\hat{u}$ – оптимальное управление в задаче (1)–(3), то выполняется условие максимума (4), откуда

$$f_E(\hat{u}) = \|\hat{x}(N)\|^2 = J_N(\hat{u}).$$

Наконец, в силу теоремы 4 для произвольного управления $\hat{u} \in \mathcal{U}_N$ справедлива двусторонняя оценка

$$f_E(\hat{u}) \leq J_N^* \leq J_N(\hat{u}),$$

поэтому в случае равенства левой и правой частей управление $\hat{u}$ является оптимальным в задаче (1)–(3). Лемма доказана.

Доказательство теоремы 5. Поскольку по условию задачи $0 \in U(k)$ при всех k , то при любом фиксированном $N > 0$ управление $u^{(0)}$, построенное по правилу

$$u^{(0)}(k) \in U(k), \quad k = 0, \dots, N-2, \quad u^{(0)}(N-1) = 0,$$

удовлетворяет геометрическим ограничениям, а шаг 1 алгоритма тем самым определен корректно. Шаги 3 и 6 корректно определены в силу компактности всех множеств $U(k)$.

Пусть $N > 0$ произвольно. Обозначим через J_N^* оптимальное значение функционала J_N в задаче (1)–(3), а через $\mathcal{U}_N^* \subset \mathcal{U}_N$ – множество всех оптимальных управлений в ней. Рассмотрим последовательность $\{u^{(l)}\} \subset \mathcal{U}_N$,

выстраиваемую алгоритмом на шагах 2–5. Согласно теореме 2 последовательность неотрицательных чисел $J_N(u^{(l)})$ монотонно не возрастает. Поэтому она имеет некоторый предел J^0 . В силу компактности множества $\mathcal{U}_N$ найдется подпоследовательность $\{u^{(l_m)}\}$ и элемент $u^* \in \mathcal{U}_N$ такие, что $u^{(l_m)} \rightarrow u^*$ при $m \rightarrow \infty$. Покажем, что $u^* \in \mathcal{U}_N^*$. Рассмотрим построенную выше функцию f_I . Для нее в силу леммы 4 имеет место равенство

$$f_I(u^{(l_m)}) = J_N(u^{(l_m+1)}),$$

причем функция f_I непрерывна. Переходя к пределу по $m \rightarrow \infty$, получаем

$$f_I(u^*) = J^0.$$

С другой стороны, функция J_N также непрерывна в силу непрерывности функции квадрата нормы и непрерывной зависимости решения уравнений (1)–(2) от параметров $u(k)$. Поэтому имеет место

$$J_N(u^*) = \lim_{m \rightarrow \infty} J_N(u^{(l_m)}) = J^0.$$

Согласно лемме 4 полученные равенства означают, что u^* – оптимальное управление в задаче (1)–(3). В частности, $J^0 = J_N(u^*) = J_N^*$.

Рассмотрим теперь функцию f_E , построенную выше, а также функцию $\text{dist}(\cdot, \mathcal{U}_N^*)$. Обе эти функции однозначно определены на $\mathcal{U}_N$. В силу леммы 4 функция f_E непрерывна и $f_E(u^*) = J_N^*$ для любого $u^* \in \mathcal{U}_N^*$. Функция $\text{dist}(\cdot, \mathcal{U}_N^*)$ непрерывна по определению и $\text{dist}(u^*, \mathcal{U}_N^*) = 0$ для любого $u^* \in \mathcal{U}_N^*$. Покажем, что последовательности чисел $f_E(u^{(l)})$ и $\text{dist}(u^{(l)}, \mathcal{U}_N^*)$ имеют пределы J_N^* и 0 соответственно. Пусть это не так и, для определенности, последовательность $\{f_E(u^{(l)})\}$ не стремится к J_N^* . Тогда для некоторого $\delta > 0$ найдется подпоследовательность $\{u^{(l_m)}\}$ такая, что $|f_E(u^{(l_m)}) - J_N^*| > \delta$ при всех m . Переходя еще раз к подпоследовательности, в силу компактности множества $\mathcal{U}_N \subset \mathbb{R}^{nN}$ можем считать, что найдется элемент $u^* \in \mathcal{U}_N$ такой, что $u^{(l_m)} \rightarrow u^*$ при $m \rightarrow \infty$. По доказанному выше имеет место включение $u^* \in \mathcal{U}_N^*$, т.е. $f_E(u^*) = J_N^*$ по лемме 4. Но в то же время функция f_E непрерывна, а значит, $|f_E(u^*) - J_N^*| \geq \delta > 0$. Полученное противоречие показывает, что для последовательности чисел $E_N^{(l)} = f_E(u^{(l)})$, построенной на шагах 2–5 алгоритма, имеет место сходимость

$$E_N^{(l)} \rightarrow J_N^*, \quad l \rightarrow \infty.$$

Аналогичные рассуждения приводят к тому, что

$$(II.11) \quad \text{dist}(u^{(l)}, \mathcal{U}_N^*) \rightarrow 0, \quad l \rightarrow \infty.$$

Поскольку $N > 0$ было выбрано произвольно, то последнее верно и для $N = N_{\min}$.

Итак, показано, что для любого $N > 0$ последовательность чисел $J_N(u^{(l)})$ монотонно сходится к J_N^* сверху и, кроме того, последовательность чисел $E_N^{(l)}$ сходится к J_N^* . При этом в силу теоремы 4 имеет место $E_N^{(l)} \leq J_N^*$ при всех l . Таким образом, доказано, что

$$(П.12) \quad J_N(u^{(l)}) \downarrow J_N^* \quad \text{и} \quad E_N^{(l)} \uparrow J_N^* \quad \text{при} \quad l \rightarrow \infty.$$

Пусть задано произвольное $\varepsilon > 0$. Для любого $N < N_{\min}$ вне зависимости от значения ε имеем $J_N^* > 0$, и в силу (П.12) найдется такое $l' = l'(N)$, что $E_N^{(l')} > 0$. Поэтому для любого такого N через конечное число внутренних итераций оказывается выполнено условие останова на шаге 7 (если до этого не оказалось выполнено условие на шаге 4). Если же $N = N_{\min}$, то $J_N^* = 0$ и опять в силу (П.12) найдется l^* , для которого $J_N(u^{(l^*)}) < \varepsilon^2$, а значит, через конечное число итераций будет выполнено условие полной остановки, проверяемое на шаге 4. Следовательно, алгоритм гарантированно завершает свою работу за конечное совокупное число итераций.

Пусть алгоритм завершил работу с ответом $N_\varepsilon, u_\varepsilon$. Из предыдущего рассуждения вытекает, что $N_\varepsilon \leq N_{\min}$. Но если число ε выбрано настолько малым, что выполняется

$$0 < \varepsilon < \varepsilon^* := \min_{0 < N < N_{\min}} J_N^*,$$

то тогда строгое неравенство $N_\varepsilon < N_{\min}$ невозможно, а значит, $N_\varepsilon = N_{\min}$.

В силу (П.11) последовательность $u^{(l)}$, построенная на шагах 2–5 при $N = N_{\min}$, сходится ко множеству $\mathcal{U}_{N_{\min}}^* = \mathcal{U}^*$ в смысле расстояния dist . Более того, эта последовательность не меняется в зависимости от выбора значения $\varepsilon \in (0; \varepsilon^*)$, поскольку условие внутренней остановки на шаге 7 от ε не зависит. При этом значение номера $l^* = l^*(\varepsilon)$, для которого по результатам работы алгоритма имеет место $u_\varepsilon = u^{(l^*)}$, не убывает с уменьшением ε . В предположении, что существует l_0 такое, что $l^*(\varepsilon) = l_0$ при всех достаточно малых ε , получаем $J_{N_{\min}}(u^{(l_0)}) < \varepsilon^2$ при всех малых $\varepsilon > 0$, т.е. $J_{N_{\min}}(u^{(l_0)}) = 0$ и $u^{(l_0)} \in \mathcal{U}^*$. В противном случае $l^*(\varepsilon) \rightarrow \infty$ с уменьшением ε . В каждом из этих случаев $\text{dist}(u_\varepsilon, \mathcal{U}^*) \rightarrow 0$ при $\varepsilon \rightarrow 0$. Теорема полностью доказана.

ПРИЛОЖЕНИЕ 2

Исходный код исполняемой программы.

```
import numpy as np
import cvxpy as cp
from tkinter import filedialog
import pathlib
import json
```

```
MAX_ITER_NUM = 200
```

```
MAX_NORM = 1e3
```

```

ex_file_path = filedialog.askopenfilename(
    initialdir = pathlib.Path(__file__).parent.resolve())

with open(ex_file_path, 'r') as file:
    ex_data = json.load(file)
n = ex_data['n']
A = ex_data['A']
x0 = np.array(ex_data['x0'])
epsilon = ex_data['epsilon']
U = ex_data['U']

Id = np.identity(n)
Zn = np.array([0] * n)

u = list()
A_ = list()
x = list()
xi = list()
psi = list()
v = cp.Variable(n)

N = 1
x.append(x0)
nrm = cp.norm(x0).value
E = 0
iter = 1

while nrm >= epsilon and nrm < MAX_NORM
    and iter < MAX_ITER_NUM:
        u.append(Zn)
        iter = 1

        A_.clear()
        A_.append(Id)
        for k in range(N):
            A_.insert(0, A_[0]@np.array(eval(A.replace
                ('k', str(N-k-1)))))

        while iter < MAX_ITER_NUM:
            xi.clear()
            xi.append(Zn)
            for k in range(N):
                xi.insert(0, xi[0]-A_[N-k]@u[N-k-1])

```

```

x.clear()
x.append(x0)
for k in range(N):
    prob = cp.Problem(cp.Minimize(
        cp.norm(A_[k+1]@v+A_[k]@x[k]-xi[k+1])),
        [eval(constr.replace('k',str(k)))
         for constr in U])
    prob.solve(solver=cp.SCS)
    u[k] = v.value
    x.append(np.array(eval(A.replace('k',str(k))))
              @x[k]+u[k])

nrm = cp.norm(x[N]).value
if nrm < epsilon: break

psi.clear()
psi.append(-2*x[N])
for k in range(N):
    psi.insert(0,np.array(eval(
        A.replace('k',str(N-k-1))))).T@psi[0])

E = nrm**2
for k in range(N):
    prob = cp.Problem(cp.Maximize(
        psi[k+1].T@(v-u[k])),
        [eval(constr.replace('k',str(k)))
         for constr in U])
    E -= prob.solve(solver=cp.CLARABEL)

print (N, iter , x[N] , nrm , np.sqrt(np.max([E,0])))

if E > 0:
    N += 1
    break
    iter += 1

if nrm >= MAX_NORM:
    print ("No_convergence!_Try_another_example.")
elif iter >= MAX_ITER_NUM:
    print ("Too_many_inner_iterations!_Break.")
else :
    print (N, x[N] , nrm)
    print ("Nmin=_", N)

```

```

ex_data[ 'Nmin' ] = N
ex_data[ 'u_opt' ] = [[u[k][j] for j in range(n)]
    for k in range(N)]
ex_data[ 'x_opt' ] = [[x[k+1][j] for j in range(n)]
    for k in range(N)]

with open(ex_file_path, 'w') as file:
    json.dump(ex_data, file, indent=4, sort_keys=True)

```

ПРИЛОЖЕНИЕ 3

Исходные данные для примера 1.

```

{
    "n": 1,
    "A": "[[1]]",
    "U": ["cp.norm(v) <= 1"],
    "x0": [2.5],
    "epsilon": 0.0001
}

```

Исходные данные для примера 2.

```

{
    "n": 2,
    "A": "[[4/5*(np.cos(1)+np.sin(1)), -8/5*np.sin(1)],
        [4/5*np.sin(1), 4/5*(np.cos(1)-np.sin(1))]]",
    "U": [
        "cp.quad_form(v, np.array([[2, 1], [1, 3]])) <= 1"
    ],
    "x0": [-37.79, -26.1],
    "epsilon": 0.0001
}

```

Исходные данные для примера 3.

```

{
    "n": 2,
    "A": "[[np.cos(np.pi/4), -np.sin(np.pi/4)],
        [np.sin(np.pi/4), np.cos(np.pi/4)]]",
    "U": [
        "cp.norm(v, \"inf\") <= 0.9**k*cp.sqrt(3)/2",
        "cp.norm(v) <= 3/4"
    ],
    "x0": [9.33, 0.2],
    "epsilon": 0.0001
}

```

Исходные данные для примера 4.

```
{
  "n": 2,
  "A": "[[(-1)**k*np.cos(np.pi/4),
          -(-1)**k*np.sin(np.pi/4)],
        [(-1)**k*np.sin(np.pi/4),
          (-1)**k*np.cos(np.pi/4)]]",
  "U": [
    "cp.norm(v, \"inf\") <= cp.sqrt(3)/2",
    "cp.norm(v) <= (0.9)**k"
  ],
  "x0": [9.33, 0.2],
  "epsilon": 0.0001
}
```

Исходные данные для примера 5.

```
{
  "n": 2,
  "A": "[[np.exp(-1/(k+1))*np.cos(1/(k+1)),
          -np.exp(-1/(k+1))*np.sin(1/(k+1))],
        [np.exp(-1/(k+1))*np.sin(1/(k+1)),
          np.exp(-1/(k+1))*np.cos(1/(k+1))]]",
  "U": [
    "cp.norm(v) <= np.sqrt((1+np.exp(-2*1/(k+1))
          -2*np.exp(-1/(k+1))*np.cos(1/(k+1)))/2)"
  ],
  "x0": [100, 100],
  "epsilon": 0.0001
}
```

СПИСОК ЛИТЕРАТУРЫ

1. *Blanchini F.* Polyhedral Set Constrained Control for Discrete-Time Systems with Unknown Additive Disturbances // IFAC Proceedings Volumes. 1991. V. 24. No. 8. P. 95–100.
2. *Blanchini F., Ukovich W.* Linear Programming Approach to the Control of Discrete-Time Periodic Systems with Uncertain Inputs // Journal of Optimization Theory and Applications. 1993. V. 78. No. 3. P. 523–539.
3. *Keerthi S., Gilbert E.* Computation of Minimum-Time Feedback Control Laws for Discrete-Time Systems with State-Control Constraints // IEEE Transactions on Automatic Control. 1987. V. 32. No. 5. P. 432–435.
4. *Abdelhak A., Rachik M.* The Linear Quadratic Minimum-Time Problem for a Class of Discrete Systems // J. Math. Program. Oper. Res. 2010. V. 59. No. 4. P. 575–587.

5. *Amato F., Cosentino C., Tommasi G.D., Pironti A., Romano M.* Input-Output Finite-Time Stabilization of Linear Time-Varying Discrete-Time Systems // IEEE Transactions on Automatic Control. 2022. V. 67. No. 9. P. 4438–4450.
6. *Chen D., Bako L., Lecoecuche S.* The Minimum-Time Problem for Discrete-Time Linear Systems: A Non-Smooth Optimization Approach // Proceedings of the IEEE International Conference on Control Applications. 2012. P. 196–201.
7. *Lee J., Haddad W.M.* Fixed Time Stability and Optimal Stabilisation of Discrete Autonomous Systems // International Journal of Control. 2022. V. 96. No. 9. P. 2341–2355.
8. *Yang H., Xia Y., Geng Q.* Stabilization on Null Controllable Region / In: Analysis and Synthesis of Delta Operator Systems with Actuator Saturation. Studies in Systems, Decision and Control. 2019. V. 193. P. 39–65.
9. *Кротов В.Ф., Гурман В.И.* Методы и задачи оптимального управления. М.: Наука, 1973.
10. *Коннов А.И., Кротов В.Ф.* О глобальных методах последовательного улучшения управляемых процессов // АиТ. 1999. № 10. С. 77–88.
11. *Пропой А.И.* Элементы теории оптимальных дискретных процессов. М.: Наука, 1973.
12. *Габасов Р.Ф., Кириллова Ф.М., Тятюшкин А.И. и др.* Конструктивные методы оптимизации (в 5 частях). Минск: Университетское издание, 1984–1998.
13. *Срочко В.А.* Итерационные методы решения задач оптимального управления. М.: Физматлит, 2000.
14. *Ибрагимов Д.Н., Царьков К.А.* Об одном подходе к решению задачи быстродействия на основе метода Кротова // АиТ. 2024. № 11. С. 3–35.
15. *Габасов Р.Ф., Кириллова Ф.М.* Качественная теория оптимальных процессов. М.: Наука, 1981.
16. *Jia Z., Cai X., Han D.* Comparison of several fast algorithms for projection onto an ellipsoid // J. Comp. Appl. Math. 2017. V. 319. P. 320–337.
17. *Kitahara T., Sukegawa N.* A Simple Projection Algorithm for Linear Programming Problems // Algorithmica. 2018. V. 81. P. 167–178.

Статья представлена к публикации членом редколлегии Н.В. Кузнецовым.

Поступила в редакцию 10.04.2025

После доработки 06.06.2025

Принята к публикации 21.07.2025

© 2025 г. М.В. ХЛЕБНИКОВ, д-р физ.-мат. наук (khlebnik@ipu.ru)
(Институт проблем управления им. В.А. Трапезникова РАН, Москва)

СИНТЕЗ ПИД-РЕГУЛЯТОРА ДЛЯ ПОДАВЛЕНИЯ ОГРАНИЧЕННЫХ ВНЕШНИХ ВОЗМУЩЕНИЙ¹

Предлагается новый подход к задаче подавления неслучайных ограниченных внешних возмущений в линейных системах управления при помощи ПИД-регулятора. Подход предполагает сведение исходной проблемы к задаче невыпуклой матричной оптимизации. Выписан градиентный метод для отыскания параметров ПИД-регулятора и дано его обоснование. Предлагаемая рекуррентная процедура отличается простотой реализации и приводит ко вполне удовлетворительным по инженерным критериям качества регуляторам.

Ключевые слова: линейная система, внешние возмущения, ПИД-регулятор, оптимизация, уравнение Ляпунова, градиентный метод.

DOI: 10.31857/S0005231025110024

1. Введение

ПИД-регуляторы являются наиболее распространенным типом автоматических регуляторов; по различным оценкам к ним относится более 90% используемых в настоящее время регуляторов. В 2019 г. IFAC Industry Committee провел [1] опрос своих членов на тему, какие технологии управления наиболее востребованы в современной промышленности; можно было указывать несколько вариантов ответов. Первое место с большим отрывом заняло ПИД-управление – 91% опрошенных. При этом согласно прогнозам на среднесрочный период, эта доля по-прежнему будет исключительно велика, составляя около 80%. Столь обширное применение ПИД-регуляторов в промышленности обусловлено рядом обстоятельств. Помимо пригодности для решения большинства практических задач, невысокой стоимости, их востребованность связана с простотой: для настройки ПИД-регулятора необходимо правильно выбрать всего три коэффициента.

Вместе с тем, сами процедуры их практической настройки во многом остаются эвристичными: в реальных установках ПИД-регуляторы часто настраиваются вручную, исходя из интуитивного понимания технологического процесса и влияния на него отдельных составляющих ПИД-управления.

¹ Результаты исследований, представленные в разделах 4 и 5, получены за счет средств Российского научного фонда (грант № 25-29-20062), <https://rscf.ru/project/25-29-20062/>.

Известные аналитические подходы к синтезу ПИД-регуляторов рассматривают “фиксированные” модели объектов управления, т.е. не являются универсальными, они также часто не учитывают влияние неопределенностей. Соответственно проблема разработки регулярных подходов к их настройке по-прежнему сохраняет свою содержательность и актуальность.

Среди публикаций, посвященных этой проблематике, отметим [2], в которой синтез ПИД-регуляторов производится на основе генетических алгоритмов, и [3], где для этой цели привлекаются эволюционные алгоритмы. Следует упомянуть и работы [4, 5], связанные с так называемым активным управлением с подавлением возмущений (ADRC), относящимся к безмодельному управлению, и [6, 7] по применению в этих целях метода оптимального значения (magnitude optimum method). Для настройки ПИД-регуляторов активно применяется и метод внутренней модели [8–10]. Обширную библиографию читатель также найдет в недавнем обзоре [11].

Общий тренд последних десятилетий – это переход к численным методам синтеза ПИД-регуляторов на основе решения задач оптимизации. В его русле находится предложенный в [12] и восходящий к [13] оптимизационный подход к задаче подавления ограниченных внешних возмущений. Напомним, что классическая задача подавления неслучайных ограниченных внешних возмущений ставится следующим образом. Рассмотрим линейную систему управления

$$\begin{aligned}\dot{x} &= Ax + Bu + Dw, \quad x(0) = x_0, \\ y &= C_1 x, \\ z &= C_2 x + B_1 u\end{aligned}$$

с состоянием $x(t) \in \mathbb{R}^n$, измеряемым выходом $y(t) \in \mathbb{R}^\ell$, регулируемым выходом $z(t) \in \mathbb{R}^r$, управлением $u(t) \in \mathbb{R}^p$ и внешним возмущением $w(t) \in \mathbb{R}^m$ таким, что $\|w(t)\| \leq \bar{w}$ для всех $t \geq 0$. Задача состоит в поиске стабилизирующей обратной связи так, чтобы уменьшить “пиковое” значение выхода z , т.е. величину $\sup_{t \geq 0} \max_{\|w\| \leq \bar{w}} \|z(t)\|$. В [12] эта задача была сведена к задаче невыпуклой матричной оптимизации, был выписан градиентный метод для отыскания статической линейной обратной связи по состоянию ($u = Kx$) или по выходу ($u = Ky$) системы и дано его обоснование.

С другой стороны, в [14] оптимизационный подход был применен к задаче синтеза ПИД-регулятора. А именно, был предложен регулярный подход к отысканию его параметров, предполагающий решение задачи невыпуклой матричной оптимизации. При этом качество регулятора оценивалось по квадратичному критерию от выхода системы: регулятор настраивался против неопределенности в начальных условиях так, чтобы выход системы был равномерно малым. Предложенная рекуррентная процедура оказалась весьма эффективной и приводящей к регуляторам, вполне удовлетворительным по инженерным критериям качества. Позже, в [15] оптимизационный подход был использован при решении задачи подавления ограниченных внешних возмущений при помощи ПИД-регулятора. Настоящая статья продолжает

и развивает эту линию исследований: в ней решается задача синтеза ПИД-регулятора для подавления ограниченных внешних возмущений. Важно отметить, что развиваемый подход может быть распространен и на различные робастные постановки задачи.

Всюду далее $\|\cdot\|$ – евклидова норма вектора и спектральная норма матрицы, $\|\cdot\|_F$ – фробениусова норма матрицы, T – символ транспонирования, tr – след матрицы, I – единичная матрица соответствующей размерности, $\lambda_i(A)$ – собственные значения матрицы A , а $\sigma(A) \doteq -\max_i \text{Re}(\lambda_i(A)) > 0$ – степень устойчивости гурвицевой матрицы A .

2. Постановка задачи

Рассмотрим линейную SISO-систему управления

$$(1) \quad \begin{aligned} \dot{x} &= Ax + bu + Dw, & x(0) &= x_0, \\ y &= c^T x, \\ z &= Cx, \end{aligned}$$

где $A \in \mathbb{R}^{n \times n}$, $b \in \mathbb{R}^n$, $D \in \mathbb{R}^{n \times m}$, $c \in \mathbb{R}^n$, $C \in \mathbb{R}^{r \times n}$, с состоянием $x(t) \in \mathbb{R}^n$, управлением $u(t) \in \mathbb{R}$, измеряемым выходом $y(t) \in \mathbb{R}$, регулируемым выходом $z(t) \in \mathbb{R}^r$ и внешним возмущением $w(t) \in \mathbb{R}^m$, удовлетворяющим ограничению

$$(2) \quad \|w(t)\| \leq \bar{w} \quad \text{для всех } t \geq 0.$$

Пара (A, D) предполагается управляемой, пара (A, C) – наблюдаемой.

Будем искать управление в виде ПИД-регулятора

$$(3) \quad u(t) = -k_P y(t) - k_I \int_0^t y(\tau) d\tau - k_D \dot{y}(t),$$

который стабилизирует замкнутую систему и подавляет воздействие внешних возмущений w , минимизируя размер ограничивающего эллипсоида для выхода z .

Напомним концепцию метода инвариантных эллипсоидов (подробнее см. [17]). Рассмотрим линейную стационарную динамическую систему

$$(4) \quad \begin{aligned} \dot{x} &= Ax + Dw, & x(0) &= x_0, \\ z &= Cx \end{aligned}$$

с состоянием $x(t) \in \mathbb{R}^n$, выходом $z(t) \in \mathbb{R}^r$ и измеримым по t внешним возмущением $w(t) \in \mathbb{R}^\ell$, ограниченным в каждый момент времени: $\|w(t)\| \leq 1$ для всех $t \geq 0$. Будем полагать, что система (4) устойчива (т.е. матрица A гурвицева), пара (A, D) управляема.

Эллипсоид с центром в начале координат будем называть *инвариантным* для динамической системы (4), если любая ее траектория, исходящая из точки, лежащей в эллипсоиде, в любой момент времени будет оставаться в этом эллипсоиде при всех допустимых внешних возмущениях, действующих на систему.

Оценивая влияние внешних возмущений на выход системы, естественно интересоваться минимальными (в том или ином смысле) эллипсоидами, содержащими этот выход. Нетрудно видеть, что если эллипсоид

$$(5) \quad \mathcal{E} = \{x \in \mathbb{R}^n : x^T P^{-1} x \leq 1\}$$

с положительно определенной матрицей P ($P \succ 0$) является инвариантным, то выход системы (4) при $x_0 \in \mathcal{E}$ принадлежит так называемому *ограничивающему* эллипсоиду

$$(6) \quad \mathcal{E}_z = \{z \in \mathbb{R}^r : z^T (CPC^T)^{-1} z \leq 1\}.$$

В литературе в качестве критерия его минимальности часто рассматривается линейная функция $f(P) = \text{tr } CPC^T$, равная сумме квадратов полуосей ограничивающего эллипсоида.

В [16] был установлен критерий инвариантности эллипсоида в терминах линейных матричных неравенств; приведем его в следующей формулировке (см. [17]).

Теорема 1. Пусть матрица A гурвицева, пара (A, D) управляема, а матрица $P(\alpha) \succ 0$ удовлетворяет уравнению Ляпунова

$$\left(A + \frac{\alpha}{2}I\right)P + P\left(A + \frac{\alpha}{2}I\right)^T + \frac{1}{\alpha}DD^T = 0$$

на интервале $0 < \alpha < 2\sigma(A)$.

Тогда задача о минимальном ограничивающем эллипсоиде для системы (4) сводится к минимизации функции $f(\alpha) = \text{tr } CP(\alpha)C^T$ на интервале $0 < \alpha < 2\sigma(A)$.

В [12] (в предположениях, которые заведомо можно ослабить) была показана строгая выпуклость функции $f(\alpha)$ на интервале $0 < \alpha < 2\sigma(A)$. Заметим также, что если α^* – точка минимума в задаче из теоремы 1 и $x(0) = x_0$ удовлетворяет условию $x_0^T P^{-1}(\alpha^*)x_0 \leq 1$, то гарантируется очевидная равномерная оценка

$$\|z(t)\| \leq \sqrt{\|CP(\alpha^*)C^T\|} \leq \sqrt{f(\alpha^*)}$$

для всех $t \geq 0$.

3. Подход к решению

Введем в рассмотрение вспомогательную скалярную переменную ξ следующим образом:

$$\dot{\xi} = y, \quad \xi(0) = 0.$$

Тогда, введя расширенный вектор состояния

$$g = \begin{pmatrix} x \\ \xi \end{pmatrix} \in \mathbb{R}^{n+1},$$

систему (1) можно записать в виде

$$(7) \quad \begin{aligned} \dot{g} &= \begin{pmatrix} A & 0 \\ c^T & 0 \end{pmatrix} g + \begin{pmatrix} b \\ 0 \end{pmatrix} u + \begin{pmatrix} D \\ 0 \end{pmatrix} w, \quad g(0) = \begin{pmatrix} x_0 \\ 0 \end{pmatrix}, \\ y &= (c^T \ 0) g, \\ z &= (C \ 0) g. \end{aligned}$$

При этом согласно (1), (3) имеем:

$$\begin{aligned} u &= -k_P y(t) - k_I \int_0^t y(\tau) d\tau - k_D \dot{y}(t) = \\ &= -k_P c^T x - k_I \xi - k_D c^T \dot{x} = -k_P c^T x - k_I \xi - k_D c^T (Ax + bu + Dw) = \\ &= -k_P (c^T \ 0) g - k_I (0 \ 1) g - k_D (c^T A \ 0) g - k_D c^T b u - k_D c^T D w, \end{aligned}$$

откуда

$$(1 + k_D c^T b) u = -k_P (c^T \ 0) g - k_I (0 \ 1) g - k_D (c^T A \ 0) g - k_D c^T D w$$

или

$$(8) \quad \begin{aligned} u &= -\frac{k_P}{1 + k_D c^T b} (c^T \ 0) g - \frac{k_I}{1 + k_D c^T b} (0 \ 1) g - \\ &\quad - \frac{k_D}{1 + k_D c^T b} (c^T A \ 0) g - \frac{k_D}{1 + k_D c^T b} c^T D w. \end{aligned}$$

Если ввести новые переменные

$$k_1 = \frac{k_P}{1 + k_D c^T b}, \quad k_2 = \frac{k_I}{1 + k_D c^T b}, \quad k_3 = \frac{k_D}{1 + k_D c^T b},$$

то (8) примет вид

$$(9) \quad u = - (k_1 c^T + k_3 c^T A \ k_2) g - k_3 c^T D w,$$

при этом исходные параметры ПИД-регулятора восстанавливаются единственным образом:

$$k_P = \frac{k_1}{1 - k_3 c^T b}, \quad k_I = \frac{k_2}{1 - k_3 c^T b}, \quad k_D = \frac{k_3}{1 - k_3 c^T b}.$$

Итак, замыкая систему (7) обратной связью (9), приходим к замкнутой системе

$$\begin{aligned}\dot{g} &= \begin{pmatrix} A - k_1 bc^T - k_3 bc^T A & -k_2 b \\ c^T & 0 \end{pmatrix} g + \begin{pmatrix} (I - k_3 bc^T)D \\ 0 \end{pmatrix} w, \quad g(0) = \begin{pmatrix} x_0 \\ 0 \end{pmatrix}, \\ z &= (C \ 0) g,\end{aligned}$$

которой можно придать вид

$$(10) \quad \begin{aligned}\dot{g} &= (\mathcal{A}_0 + k_1 \mathcal{A}_1 + k_2 \mathcal{A}_2 + k_3 \mathcal{A}_3)g + (\mathcal{D}_0 + k_3 \mathcal{D}_3)w, \quad g(0) = \begin{pmatrix} x_0 \\ 0 \end{pmatrix}, \\ z &= \mathcal{C}g,\end{aligned}$$

где

$$\begin{aligned}\mathcal{A}_0 &= \begin{pmatrix} A & 0 \\ c^T & 0 \end{pmatrix}, \quad \mathcal{A}_1 = \begin{pmatrix} -bc^T & 0 \\ 0 & 0 \end{pmatrix}, \quad \mathcal{A}_2 = \begin{pmatrix} 0 & -b \\ 0 & 0 \end{pmatrix}, \quad \mathcal{A}_3 = \begin{pmatrix} -bc^T A & 0 \\ 0 & 0 \end{pmatrix}, \\ \mathcal{D}_0 &= \begin{pmatrix} D \\ 0 \end{pmatrix}, \quad \mathcal{D}_3 = \begin{pmatrix} -bc^T D \\ 0 \end{pmatrix}, \quad \mathcal{C} = (C \ 0).\end{aligned}$$

Замечание 1. Обратим внимание, что пара $(\mathcal{A}_0, \mathcal{D}_0)$ управляема. В самом деле, в противном случае существует вектор $0 \neq v \in \mathbb{C}^{n+1}$ такой, что при некотором $\lambda \in \mathbb{C}$ выполняются условия

$$v^* \begin{pmatrix} A & 0 \\ c^T & 0 \end{pmatrix} = \lambda v^*, \quad v^* \begin{pmatrix} D \\ 0 \end{pmatrix} = 0.$$

Представим вектор v в виде $v = \begin{pmatrix} v_1 \\ v_2 \end{pmatrix}$, $v_1 \in \mathbb{C}^n$, $v_2 \in \mathbb{C}$. Тогда имеем

$$\begin{pmatrix} v_1 \\ v_2 \end{pmatrix}^* \begin{pmatrix} A & 0 \\ c^T & 0 \end{pmatrix} = \lambda \begin{pmatrix} v_1 \\ v_2 \end{pmatrix}^*, \quad \begin{pmatrix} v_1 \\ v_2 \end{pmatrix}^* \begin{pmatrix} D \\ 0 \end{pmatrix} = 0$$

или $v_1^* A = \lambda v_1^*$, $v_1^* D = 0$, что противоречит управляемости пары (A, D) .

Таким образом, “номинальная” система

$$(11) \quad \begin{aligned}\dot{g} &= \mathcal{A}_0 g + \mathcal{D}_0 w, \\ z &= \mathcal{C}g\end{aligned}$$

является управляемой.

Аналогичным образом пара $(\mathcal{A}_0, \mathcal{C})$ является наблюдаемой. В самом деле, в противном случае существует вектор $0 \neq v \in \mathbb{C}^{n+1}$ такой, что при некотором $\lambda \in \mathbb{C}$ выполняются условия

$$\begin{pmatrix} A & 0 \\ c^T & 0 \end{pmatrix} v = \lambda v, \quad (C \ 0) v = 0.$$

Вновь представляя вектор v в виде $v = \begin{pmatrix} v_1 \\ v_2 \end{pmatrix}$, $v_1 \in \mathbb{C}^n$, $v_2 \in \mathbb{C}$, получим

$$\begin{pmatrix} A & 0 \\ c^T & 0 \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \end{pmatrix} = \lambda \begin{pmatrix} v_1 \\ v_2 \end{pmatrix}, \quad (C \ 0) \begin{pmatrix} v_1 \\ v_2 \end{pmatrix} = 0$$

или $Av_1 = \lambda v_1$, $Cv_1 = 0$, что противоречит наблюдаемости пары (A, C) . Таким образом, “номинальная” система (11) является наблюдаемой.

Продолжим: следуя методу инвариантных эллипсоидов, заключим состояние g системы (10) в инвариантный эллипсоид (5), порожденный матрицей $0 \prec P \in \mathbb{S}^{n+1}$, и будем минимизировать размер соответствующего ограничивающего эллипсоида (6) по выходу $z = Cg$.

Учитывая условие (2) путем масштабирования соответствующим образом матрицы D , согласно теореме 1 приходим к задаче минимизации $\text{tr } CPC^T$ при ограничении

$$\begin{aligned} & \left(\mathcal{A}_0 + k_1 \mathcal{A}_1 + k_2 \mathcal{A}_2 + k_3 \mathcal{A}_3 + \frac{\alpha}{2} I \right) P + P \left(\mathcal{A}_0 + k_1 \mathcal{A}_1 + k_2 \mathcal{A}_2 + k_3 \mathcal{A}_3 + \frac{\alpha}{2} I \right)^T + \\ & + \frac{\bar{w}^2}{\alpha} (D_0 + k_3 D_3)(D_0 + k_3 D_3)^T = 0. \end{aligned}$$

Введя для удобства обозначения

$$\begin{aligned} A_k & \doteq \mathcal{A}_0 + \{\mathcal{A}, k\} \doteq \mathcal{A}_0 + k_1 \mathcal{A}_1 + k_2 \mathcal{A}_2 + k_3 \mathcal{A}_3, \\ D_k & \doteq \begin{pmatrix} (I - k_3 bc^T)D \\ 0 \end{pmatrix}, \end{aligned}$$

придадим ему вид

$$(12) \quad \left(\mathcal{A}_0 + \{\mathcal{A}, k\} + \frac{\alpha}{2} I \right) P + P \left(\mathcal{A}_0 + \{\mathcal{A}, k\} + \frac{\alpha}{2} I \right)^T + \frac{\bar{w}^2}{\alpha} D_k D_k^T = 0.$$

При этом оптимизация проводится по матричной переменной $0 \prec P \in \mathbb{S}^{n+1}$, векторной переменной $k \in \mathbb{R}^3$ и скалярному параметру $\alpha > 0$.

Окончательно, в качестве критерия качества примем величину

$$(13) \quad f(k, \alpha) = \text{tr } CPC^T + \rho \|k\|^2, \quad \rho > 0,$$

в которую введен штраф за величину управления (коэффициент $\rho > 0$ регулирует его важность).

Таким образом, исходная задача о синтезе ПИД-регулятора, подавляющего внешние возмущения, свелась к невыпуклой матричной оптимизационной задаче (13)–(12). Обратим внимание, что при заданных k и α матрица P находится из уравнения Ляпунова (12); тем самым, независимыми переменными являются k и α .

Отсутствие выпуклости в полученной оптимизационной задаче обусловлено наличием билинейных слагаемых вида $A_i k_i P$, $i = 1, \dots, 3$, в уравнении

Ляпунова (12), задающем ограничение в пространстве параметров. Фактически приходится иметь дело с билинейным матричным уравнением, решение которого (как и в случае с билинейными матричными неравенствами), является NP-трудной задачей [18–20]. В некоторых частных случаях билинейные матричные уравнения (или неравенства) могут быть линеаризованы путем специальных замен переменных: например, в задаче стабилизации линейной стационарной системы вида $\dot{x} = Ax + Bu$ пропорциональным регулятором $u = Kx$. Однако уже в задаче стабилизации этой системы по выходу $y = Cx$ (т.е. с помощью регулятора $u = KCx$) такой подход принципиально невозможен, и для ее решения требуются иные методы; один из них сопряжен с наложением более сильных условий, предъявляемых к системе, например, в духе условий теоремы о пассивации (частотной теоремы с обратной связью), см. [19]. Несмотря на разницу между постановками задач стабилизации и подавления внешних возмущений, в них возникают сходные проблемы, что демонстрирует сложность решения рассматриваемой задачи. В следующем разделе, после изучения свойств минимизируемой функции, прибегнем к прямой оптимизации в пространстве параметров для ее решения.

4. Свойства функции $f(k, \alpha)$

Минимизацию функции $f(k, \alpha)$ по α можно осуществлять, например, при помощи метода Ньютона, как было предложено в [12]. А именно, рассмотрим задачу

$$\min f(\alpha), \quad f(\alpha) = \text{tr } PC^T C,$$

при ограничении

$$\left(A + \frac{\alpha}{2}I\right)P + P\left(A + \frac{\alpha}{2}I\right)^T + \frac{1}{\alpha}DD^T = 0$$

относительно матричной переменной $P \in \mathbb{S}^n$ и скалярного параметра $0 < \alpha < 2\sigma(A)$; матрица A предполагается устойчивой (гурвицевой).

Зададимся начальным приближением $0 < \alpha_0 < 2\sigma(A)$ и применим итерационный процесс

$$\alpha_{j+1} = \alpha_j - \frac{f'(\alpha_j)}{f''(\alpha_j)},$$

где

$$f'(\alpha) = \text{tr } Y \left(P - \frac{1}{\alpha^2} DD^T \right), \quad f''(\alpha) = 2 \text{tr } Y \left(X + \frac{1}{\alpha^3} DD^T \right),$$

а матрицы Y и X – решения уравнений Ляпунова

$$\left(A + \frac{\alpha}{2}I\right)^T Y + Y \left(A + \frac{\alpha}{2}I\right) + C^T C = 0$$

и

$$\left(A + \frac{\alpha}{2}I\right) X + X \left(A + \frac{\alpha}{2}I\right)^T + P - \frac{1}{\alpha^2} DD^T = 0.$$

Метод будет сходиться глобально (быстрее, чем геометрическая прогрессия с коэффициентом $1/2$), причем в окрестности решения сходимость квадратичная.

С другой стороны, в силу выпуклости (см. [12]) функции $f(\alpha)$ ее минимизацию на интервале $(0, 2\sigma(A))$ можно весьма эффективно проводить и более простыми средствами, например методом золотого сечения.

Далее, введем в рассмотрение функцию

$$f(k) \doteq \min_{\alpha} f(k, \alpha).$$

Легко видеть, что $f(k)$ определена и неотрицательна на множестве $\mathcal{S}$ стабилизирующих регуляторов. При этом множество $\mathcal{S}$ может быть невыпуклым и несвязным, причем его границы могут быть негладкими.

Сделаем следующее предположение.

Предположение. Известен стабилизирующий ПИД-регулятор с коэффициентами

$$k^{(0)} = \begin{pmatrix} k_1^{(0)} \\ k_2^{(0)} \\ k_3^{(0)} \end{pmatrix},$$

т.е. такой, что матрица $A_{k^{(0)}} = \mathcal{A}_0 + \{\mathcal{A}, k^{(0)}\}$ гурвицева.

Перейдем к свойствам градиента минимизируемой функции.

Лемма 1. Функция $f(k, \alpha)$ определена на множестве стабилизирующих обратных связей k и для $0 < \alpha < 2\sigma(A_k)$. На этом допустимом множестве она дифференцируема, причем градиент дается выражениями

$$(14) \quad \begin{aligned} f'_{\alpha}(k, \alpha) &= \text{tr} Y \left(P - \frac{\bar{w}^2}{\alpha^2} D_k D_k^T \right), \\ \frac{1}{2} \frac{\partial f(k, \alpha)}{\partial k_i} &= \text{tr} P Y \mathcal{A}_i + \rho k_i - \delta_{i3} \frac{\bar{w}^2}{\alpha} \text{tr} Y D_k \begin{pmatrix} bc^T D \\ 0 \end{pmatrix}^T, \quad i = 1, \dots, 3, \end{aligned}$$

где δ_{ij} – символ Кронекера, матрица P является решением уравнения (12), а матрица Y – решение уравнения Ляпунова

$$(15) \quad \left(\mathcal{A}_0 + \{\mathcal{A}, k\} + \frac{\alpha}{2} I \right)^T Y + Y \left(\mathcal{A}_0 + \{\mathcal{A}, k\} + \frac{\alpha}{2} I \right) + C^T C = 0.$$

Минимум $f(k, \alpha)$ достигается во внутренней точке допустимого множества и определяется условиями

$$f'_{\alpha}(k, \alpha) = 0, \quad f'_{k_i}(k, \alpha) = 0, \quad i = 1, \dots, 3.$$

При этом $f(k, \alpha)$ как функция от α строго выпукла на $0 < \alpha < 2\sigma(A_k)$ и достигает минимума во внутренней точке этого интервала.

Свойства якобиана функции $f(k)$ представлены следующим утверждением.

Лемма 2. Функция $f(k)$ дважды дифференцируема, при этом

$$(16) \quad \begin{aligned} \frac{1}{2}(f''(k)v, v) &= 2 \operatorname{tr} PY' \{ \mathcal{A}, v \} + \rho(v, v) - \\ &- \frac{\bar{w}^2}{\alpha} v_3 \operatorname{tr} \left[2Y' D_k - v_3 Y \begin{pmatrix} bc^T D \\ 0 \end{pmatrix} \right] \begin{pmatrix} bc^T D \\ 0 \end{pmatrix}^T, \end{aligned}$$

где $v \in \mathbb{R}^3$, матрица P удовлетворяет уравнению (12), матрица Y удовлетворяет уравнению (15), а матрица Y' является решением уравнения Ляпунова

$$(17) \quad \begin{aligned} \left(\mathcal{A}_0 + \{ \mathcal{A}, k \} + \frac{\alpha}{2} I \right)^T Y' + Y' \left(\mathcal{A}_0 + \{ \mathcal{A}, k \} + \frac{\alpha}{2} I \right) + \\ + \{ \mathcal{A}, v \}^T Y + Y \{ \mathcal{A}, v \} = 0. \end{aligned}$$

Градиент функции $f(k)$ не является липшицевым на множестве $\mathcal{S}$ стабилизирующих регуляторов, однако он обладает этим свойством на его подмножестве $\mathcal{S}_0$. Соответствующий результат будет представлен ниже.

Далее для получения простых количественных оценок в леммах 3 и 4 в оптимизационную задачу (13), (12) вводятся регуляризующие добавки ε и δ , и она принимает следующий вид:

$$\min f(k, \alpha), \quad f(k, \alpha) = \operatorname{tr} P(\mathcal{C}^T \mathcal{C} + \varepsilon I) + \rho \|k\|^2, \quad 0 < \varepsilon \ll 1,$$

при ограничении

$$(18) \quad \left(A_k + \frac{\alpha}{2} I \right) P + P \left(A_k + \frac{\alpha}{2} I \right)^T + \frac{\bar{w}^2}{\alpha} (D_k D_k^T + \delta I) = 0, \quad 0 < \delta \ll 1.$$

Требование необходимости их введения можно существенно ослабить, но цель сейчас состоит в получении наиболее простых и наглядных результатов.

Лемма 3. На множестве $\mathcal{S}$ стабилизирующих регуляторов функция $f(k)$ коэрцитивна (т.е. стремится к бесконечности на границе области), причем справедливы следующие оценки:

$$(19) \quad \begin{aligned} f(k) &\geq \frac{\bar{w}^2}{4\sigma(A_k)} \frac{\varepsilon}{\|A_k\| + \sigma(A_k)} \|D_k\|_F^2, \\ f(k) &\geq \rho \|k\|^2. \end{aligned}$$

Введем в рассмотрение множество уровня

$$\mathcal{S}_0 = \{ k \in \mathcal{S} : f(k) \leq f(k^{(0)}) \}.$$

Из леммы 3 вытекает очевидное следствие 1.

Следствие 1. Для любого регулятора $k^{(0)} \in \mathcal{S}$ множество $\mathcal{S}_0$ ограничено.

С другой стороны, у функции $f(k)$ на множестве $\mathcal{S}_0$ существует точка минимума (как у непрерывной функции на компактном множестве), но множество $\mathcal{S}_0$ не имеет общих точек с границей $\mathcal{S}$ в силу (19). Выше было показано, что $f(k)$ дифференцируема на $\mathcal{S}_0$. Следовательно, справедливо следствие 2.

Следствие 2. Существует точка минимума k_ на множестве $\mathcal{S}$, причем $f'(k_*) = 0$.*

Лемма 4. На множестве $\mathcal{S}_0$ градиент функции $f(k)$ липшицев с константой

$$\begin{aligned} L = & 2\sqrt{3(n+1)} \frac{f(k_0)}{\varepsilon} \frac{2(n+1)^{3/2} f(k_0)}{\bar{w}^2 \varepsilon \delta} \times \\ & \times \left[\frac{4f^2(k_0)}{\bar{w}^4 \delta^2} \left(\|\mathcal{A}_0\| + \max_i \|\mathcal{A}_i\| \sqrt{\frac{3}{\rho} f(k_0)} \right)^2 + 3 \max_i \|\mathcal{A}_i\|^2 \right] \times \\ & \times \left(\|\mathcal{A}_0\| + \max_i \|\mathcal{A}_i\| \sqrt{\frac{3}{\rho} f(k_0)} \right) \max_i \|\mathcal{A}_i\| + 2\rho + \\ & + 2\bar{w}^2 \left(2 \frac{(n+1)f(k_0)}{\bar{w}^2 \varepsilon \delta} \left[\frac{4f^2(k_0)}{\bar{w}^4 \delta^2} \left(\|\mathcal{A}_0\| + \max_i \|\mathcal{A}_i\| \sqrt{\frac{3}{\rho} f(k_0)} \right)^2 + 3 \max_i \|\mathcal{A}_i\|^2 \right] \times \right. \\ & \left. \times \left(1 + \sqrt{\frac{f(k_0)}{\rho}} \|bc^T\| \right) \|D\|_F + \frac{f(k_0)}{\bar{w}^2 \delta} \|bc^T D\|_F \right) \|bc^T D\|_F. \end{aligned}$$

Полученные свойства минимизируемой функции и ее производных позволяют построить оптимизационную процедуру и обосновать ее сходимость.

5. Алгоритм оптимизации

Предлагается итеративный подход к решению этой задачи, в основе которого лежит применение градиентного метода по переменной k и выпуклой минимизации по α . Приведем принципиальную схему соответствующего алгоритма.

- 1) Задаемся параметрами $\varepsilon > 0$, $\gamma > 0$, $0 < \tau < 1$ и начальным стабилизирующим приближением $k^{(0)}$. Вычисляем величину

$$\alpha_0 = \sigma(\mathcal{A}_0 + \{\mathcal{A}, k^{(0)}\}).$$

- 2) На j -й итерации заданы $k^{(j)}$, α_j . Вычисляем $A_{k^{(j)}} = \mathcal{A}_0 + \{\mathcal{A}, k^{(j)}\}$ и находим матрицы P , Y как решения уравнений (12) и (15). Вычисляем градиент

$$H_j = \nabla_k f(k^{(j)}, \alpha_j)$$

из соотношений (14). Если $\|H_j\| \leq \varepsilon$, то $k^{(j)}$ принимаем за приближенное решение и завершаем работу алгоритма.

3) Делаем шаг градиентного метода

$$k^{(j+1)} = k^{(j)} - \gamma_j H_j.$$

При этом длину шага $\gamma_j > 0$ подбираем дроблением γ до выполнения условий:

а. матрица $\mathcal{A}_0 + \{\mathcal{A}, k^{(j+1)}\} + \frac{\alpha_j}{2}I$ гурвицева;

б. $f(k^{(j+1)}) \leq f(k^{(j)}) - \tau \gamma_j \|H_j\|^2$.

4) Для полученного $k^{(j+1)}$ решаем задачу минимизации $f(k^{(j+1)}, \alpha)$ по α (см. начало раздела 4), получая α_{j+1} . Переходим к п. 2.

Предлагаемый метод сходится в следующем смысле.

Теорема 2. В алгоритме 1 на каждой итерации реализуется лишь конечное число дроблений γ_j , функция $f(k^{(j)})$ монотонно убывает и градиент стремится к нулю

$$\lim_{j \rightarrow \infty} \|H_j\| = 0$$

со скоростью геометрической прогрессии.

В самом деле, алгоритм 1 определен корректно в начальной точке, так как $k^{(0)}$ является стабилизирующим регулятором в силу предположения. При этом при достаточно малых γ_j в алгоритме происходит монотонное уменьшение $f(k)$ (движение по антиградиенту); такой способ подбора шага в алгоритме гарантирует, что величины $k^{(j)}$ остаются в области $\mathcal{S}_0$, для которой лемма 4 гарантирует липшицевость градиента. Таким образом, применимы результаты о сходимости градиентного метода для безусловной минимизации [21]. В частности, условие б на шаге 3 алгоритма 1 будет выполнено после конечного числа дроблений, а в градиентном методе будет иметь место сходимость по градиенту с линейной скоростью.

Естественно, что трудно рассчитывать на сходимость к глобальному минимуму, поскольку область определения $f(k)$ может быть даже несвязной.

Сделаем важное

Замечание 2. Введение в функционал (13) вместо евклидовой нормы взвешенной

$$\|x\|_\rho \doteq \sqrt{\sum_i \rho_i x_i^2}, \quad \rho_i > 0$$

дает возможность несколько более гибкой настройки параметров ПИД-регулятора, позволяя в разной мере влиять на их величину.

6. Примеры

Рассматриваемый в этом разделе пример взят из статьи [22]. Рассмотрим передаточную функцию

$$G(s) = \frac{1}{(1+s)(1+\alpha s)(1+\alpha^2 s)(1+\alpha^3 s)}, \quad \alpha = 0,5.$$

МАТЛАВ-процедура **tf2ss** доставляет матрицы

$$A = \begin{pmatrix} -15 & -70 & -120 & -64 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}, \quad b = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \quad c = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 64 \end{pmatrix}$$

системы (4). В качестве матрицы D возьмем

$$D = \begin{pmatrix} 10 \\ 0 \\ 0 \\ 0 \end{pmatrix},$$

а в качестве матрицы регулируемого выхода

$$C = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}.$$

Положим $\rho = 0,1$ и выберем в качестве начального некоторый стабилизирующий ПИД-регулятор с коэффициентами

$$k_0 = \begin{pmatrix} 0,6389 \\ 0,9853 \\ 0,3243 \end{pmatrix}.$$

Итерационный процесс, соответствующий предложенному алгоритму, завершился нахождением ПИД-регулятора с коэффициентами

$$k_* = \begin{pmatrix} 0,6670 \\ 0,5466 \\ 0,1081 \end{pmatrix},$$

и соответствующего ограничивающего эллипса с матрицей

$$CP_*C^T = \begin{pmatrix} 4,0336 & -0,1999 \\ -0,1999 & 0,3259 \end{pmatrix}, \quad \text{tr } CP_*C^T = 4,3595.$$

Динамика изменения критерия $f(k)$ показана на рис. 1.

Система, замкнутая ПИД-регулятором с коэффициентами k_* , является устойчивой по критерию Найквиста; ее минимальный запас устойчивости по модулю составляет 23,3 дБ, а по фазе $70,3^\circ$, см. рис. 2.

Положив теперь $\rho = 10$, приходим к ПИД-регулятору с коэффициентами

$$k'_* = \begin{pmatrix} 0,1232 \\ 0,2410 \\ -0,0710 \end{pmatrix}$$

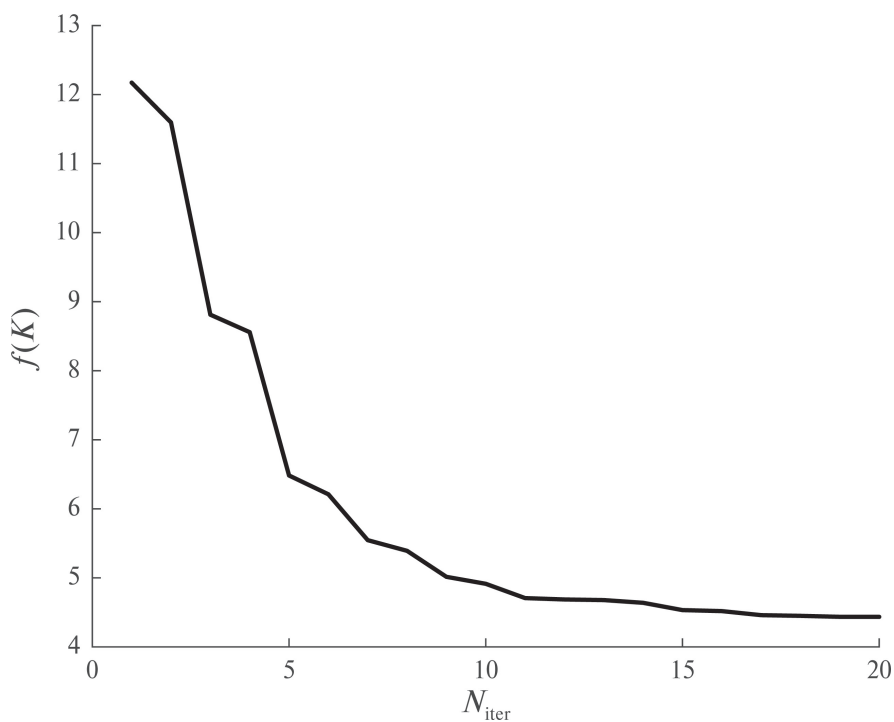


Рис. 1. Оптимизационная процедура.

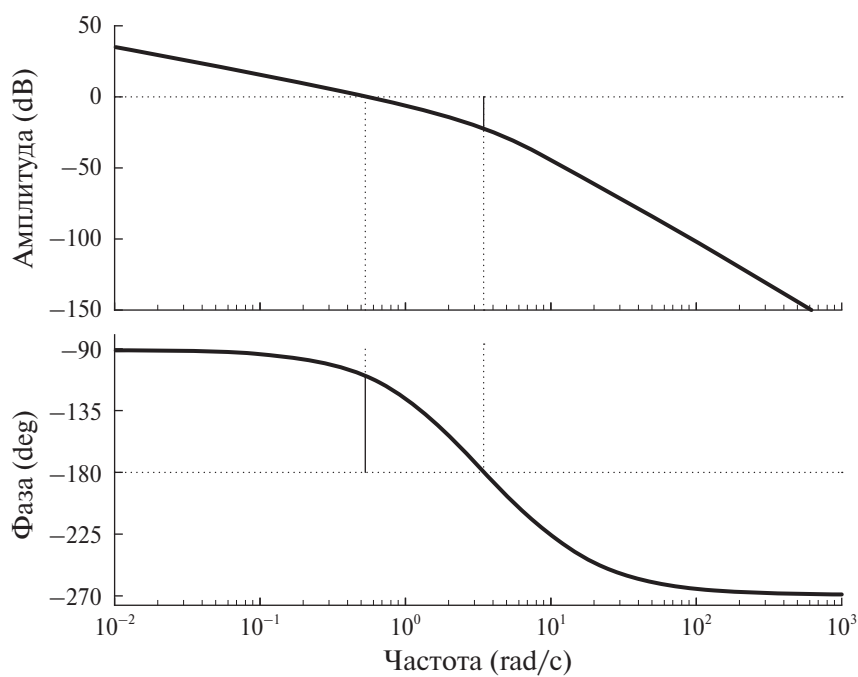


Рис. 2. ЛАФЧХ замкнутой системы.

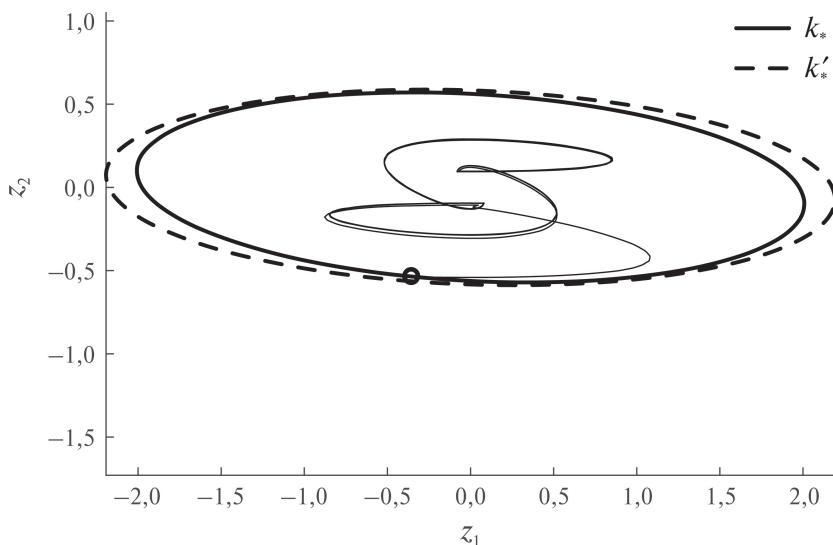


Рис. 3. Ограничивающие эллипсы.

и соответствующему ограничивающему эллипсу с матрицей

$$CP_*'C^T = \begin{pmatrix} 4,8090 & -0,1560 \\ -0,1560 & 0,3453 \end{pmatrix}, \quad \text{tr} CP_*'C^T = 5,15437.$$

При этом норма вектора коэффициентов ПИД-регулятора уменьшилась втрое; “платой” за это явилось увеличение ограничивающего эллипса менее чем на 20%.

На рис. 3 показаны полученные ограничивающие эллипсы и траектория системы, замкнутой ПИД-регулятором с коэффициентами k_* при некотором допустимом внешнем возмущении.

Вычисления производились в среде MATLAB с использованием программного пакета `cvx` [23].

Проведение обширного численного моделирования с обсуждением вычислительных аспектов автор предполагает осуществить в будущем; здесь же важна принципиальная возможность нового подхода к задаче подавления возмущений.

7. Заключение

Предложен новый легкорезализуемый подход к задаче синтеза ПИД-регуляторов, подавляющих неслучайные ограниченные внешние возмущения в линейных системах управления. Подход основан на сведении исходной задачи к невыпуклой задаче матричной оптимизации, которая далее решается градиентным методом. Построен соответствующий алгоритм и дано его обоснование.

Несмотря на то, что в работе рассматриваются только SISO-системы, развиваемый подход полностью переносится и на многомерный случай; при этом

выкладки становятся несколько более громоздкими, в то время как идейная сторона меняется мало.

Важным направлением дальнейших исследований является распространение полученных результатов на системы с неопределенностями.

ПРИЛОЖЕНИЕ

Лемма П.1 [12]. Пусть X и Y – решения двойственных уравнений Ляпунова с гурвицевой матрицей A :

$$A^T X + X A + W = 0 \quad \text{и} \quad A Y + Y A^T + V = 0.$$

Тогда $\text{tr}(XV) = \text{tr}(YW)$.

Лемма П.2 [24]. 1. Для вещественных матриц A и B соответствующих размерностей справедливы соотношения

$$\begin{aligned} \|AB\|_F &\leq \|A\|_F \|B\|, \\ |\text{tr} AB| &\leq \|A\|_F \|B\|_F, \\ \|A\| &\leq \|A\|_F, \\ AB + B^T A^T &\leq \varepsilon A A^T + \frac{1}{\varepsilon} B^T B \quad \text{для любого } \varepsilon > 0. \end{aligned}$$

2. Для неотрицательно определенных матриц A и B справедливы соотношения

$$0 \leq \lambda_{\min}(A) \lambda_{\max}(B) \leq \lambda_{\min}(A) \text{tr} B \leq \text{tr} AB \leq \lambda_{\max}(A) \text{tr} B \leq \text{tr} A \text{tr} B.$$

Доказательство леммы 1. Продифференцируем уравнение (12) по α :

$$(П.1) \quad \left(\mathcal{A}_0 + \{\mathcal{A}, k\} + \frac{\alpha}{2} I \right) \frac{\partial P}{\partial \alpha} + \frac{\partial P}{\partial \alpha} \left(\mathcal{A}_0 + \{\mathcal{A}, k\} + \frac{\alpha}{2} I \right)^T + P - \frac{\bar{w}^2}{\alpha^2} D_k D_k^T = 0.$$

Применяя лемму П.1 к двойственным уравнениям Ляпунова (П.1) и (15), получаем

$$f'_\alpha(k, \alpha) = \text{tr} \mathcal{C} \frac{\partial P}{\partial \alpha} \mathcal{C}^T = \text{tr} \frac{\partial P}{\partial \alpha} \mathcal{C}^T \mathcal{C} = \text{tr} Y \left(P - \frac{\bar{w}^2}{\alpha^2} D_k D_k^T \right).$$

Для дифференцирования по k дадим ему приращение Δk и обозначим соответствующее приращение P через ΔP :

$$\begin{aligned} &\left(\mathcal{A}_0 + \{\mathcal{A}, k + \Delta k\} + \frac{\alpha}{2} I \right) (P + \Delta P) + \\ &+ (P + \Delta P) \left(\mathcal{A}_0 + \{\mathcal{A}, k + \Delta k\} + \frac{\alpha}{2} I \right)^T + \frac{\bar{w}^2}{\alpha} D_{k+\Delta k} D_{k+\Delta k}^T = 0, \end{aligned}$$

где

$$D_{k+\Delta k} = \begin{pmatrix} (I - (k_3 + \Delta k_3)bc^T)D \\ 0 \end{pmatrix} = D_k - \Delta k_3 \begin{pmatrix} bc^T D \\ 0 \end{pmatrix}.$$

После линеаризации и вычитания этого и предыдущего уравнений, имеем

$$(П.2) \quad \left(\mathcal{A}_0 + \{\mathcal{A}, k\} + \frac{\alpha}{2} I \right) \Delta P + \Delta P \left(\mathcal{A}_0 + \{\mathcal{A}, k\} + \frac{\alpha}{2} I \right)^T + \\ + \{\mathcal{A}, \Delta k\} P + P \{\mathcal{A}, \Delta k\}^T - \Delta k_3 \frac{\bar{w}^2}{\alpha} \left[D_k \begin{pmatrix} bc^T D \\ 0 \end{pmatrix}^T + \begin{pmatrix} bc^T D \\ 0 \end{pmatrix} D_k^T \right] = 0.$$

Вычислим приращение $f(k)$, линеаризуя соответствующие величины:

$$\begin{aligned} \Delta f(k) &= \text{tr } \mathcal{C}(P + \Delta P) \mathcal{C}^T + \rho \|k + \Delta k\|^2 - (\text{tr } \mathcal{C} P \mathcal{C}^T + \rho \|k\|^2) = \\ &= \text{tr } \Delta P \mathcal{C}^T \mathcal{C} + 2\rho k^T \Delta k. \end{aligned}$$

Применяя лемму П.1 к двойственным уравнениям Ляпунова (П.2) и (15), имеем

$$\begin{aligned} \Delta f(k) &= 2 \text{tr } Y \left[\{\mathcal{A}, \Delta k\} P - \Delta k_3 \frac{\bar{w}^2}{\alpha} D_k \begin{pmatrix} bc^T D \\ 0 \end{pmatrix}^T \right] + 2\rho k^T \Delta k = \\ &= 2 \text{tr } P Y \{\mathcal{A}, \Delta k\} - 2 \frac{\bar{w}^2}{\alpha} \text{tr } Y D_k \begin{pmatrix} bc^T D \\ 0 \end{pmatrix}^T \Delta k_3 + 2\rho k^T \Delta k. \end{aligned}$$

Таким образом,

$$df(k) = 2 \text{tr } P Y \sum_{i=1}^3 \mathcal{A}_i dk_i - 2 \frac{\bar{w}^2}{\alpha} \text{tr } Y D_k \begin{pmatrix} bc^T D \\ 0 \end{pmatrix}^T dk_3 + 2\rho \sum_{i=1}^3 k_i dk_i.$$

Лемма 1 доказана.

Доказательство леммы 2. Пусть $v = \begin{pmatrix} v_1 \\ v_2 \\ v_3 \end{pmatrix} \in \mathbb{R}^3$. Вычислим вели-

чину $(f''(k)v, v)$. Для этого, линеаризуя соответствующие величины и введя для удобства обозначение

$$[\text{tr } P Y \mathcal{A}] \doteq \begin{pmatrix} \text{tr } P Y \mathcal{A}_1 \\ \text{tr } P Y \mathcal{A}_2 \\ \text{tr } P Y \mathcal{A}_3 \end{pmatrix},$$

вычислим приращение $f'(k)$ по направлению v :

$$\begin{aligned} &\frac{1}{2} \Delta f'(k)v = \\ &= [\text{tr } (P + \Delta P)(Y + \Delta Y) \mathcal{A}] + \rho(k + \delta v) - \frac{\bar{w}^2}{\alpha} \text{tr } (Y + \Delta Y) D_{k+\delta v} \begin{pmatrix} bc^T D \\ 0 \end{pmatrix}^T e_3 - \\ &\quad - \left([\text{tr } P Y \mathcal{A}] + \rho k - \frac{\bar{w}^2}{\alpha} \text{tr } Y D_k \begin{pmatrix} bc^T D \\ 0 \end{pmatrix}^T e_3 \right) = \\ &= [\text{tr } (P + \delta P'(k)v)(Y + \delta Y'(k)v) \mathcal{A}] + \rho(k + \delta v) - \end{aligned}$$

$$\begin{aligned}
& -\frac{\bar{w}^2}{\alpha} \operatorname{tr} (Y + \delta Y'(k)v) D_{k+\delta v} \begin{pmatrix} bc^T D \\ 0 \end{pmatrix}^T e_3 - \\
& - \left([\operatorname{tr} PY \mathcal{A}] + \rho k - \frac{\bar{w}^2}{\alpha} \operatorname{tr} Y D_k \begin{pmatrix} bc^T D \\ 0 \end{pmatrix}^T e_3 \right) = \\
& = \delta [\operatorname{tr} (PY'(k)v + P'(k)vY) \mathcal{A}] + \delta \rho v - \\
& - \frac{\bar{w}^2}{\alpha} \operatorname{tr} \left[(Y + \delta Y') \left(D_k - \delta v_3 \begin{pmatrix} bc^T D \\ 0 \end{pmatrix} \right) - Y D_k \right] \begin{pmatrix} bc^T D \\ 0 \end{pmatrix}^T e_3 = \\
& = \delta [\operatorname{tr} (PY'(k)v + P'(k)vY) \mathcal{A}] + \delta \rho v - \\
& - \delta \frac{\bar{w}^2}{\alpha} \operatorname{tr} \left[Y' D_k - v_3 Y \begin{pmatrix} bc^T D \\ 0 \end{pmatrix} \right] \begin{pmatrix} bc^T D \\ 0 \end{pmatrix}^T e_3,
\end{aligned}$$

где $e_3 = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$, а

$$\begin{aligned}
\Delta P &= P(k + \delta v) - P(k) = \delta P'(k)v, \\
\Delta Y &= Y(k + \delta v) - Y(k) = \delta Y'(k)v.
\end{aligned}$$

Таким образом, обозначая $P' = P'(k)v$ и $Y' = Y'(k)v$, имеем

$$\begin{aligned}
& \frac{1}{2} (f''(k)v, v) = \\
& = \operatorname{tr} (PY' + P'Y) \{\mathcal{A}, v\} + \rho(v, v) - \frac{\bar{w}^2}{\alpha} \operatorname{tr} \left[Y' D_k - v_3 Y \begin{pmatrix} bc^T D \\ 0 \end{pmatrix} \right] \begin{pmatrix} bc^T D \\ 0 \end{pmatrix}^T v_3.
\end{aligned}$$

Далее, $P = P(k)$ есть решение уравнения (12); запишем его в приращениях по направлению v :

$$\begin{aligned}
& \left(\mathcal{A}_0 + \{\mathcal{A}, k + \delta v\} + \frac{\alpha}{2} I \right) (P + \delta P') + \\
& + (P + \delta P') \left(\mathcal{A}_0 + \{\mathcal{A}, k + \delta v\} + \frac{\alpha}{2} I \right)^T + \frac{\bar{w}^2}{\alpha} D_{k+\delta v} D_{k+\delta v}^T = 0
\end{aligned}$$

или

$$\begin{aligned}
& \left(\mathcal{A}_0 + \{\mathcal{A}, k + \delta v\} + \frac{\alpha}{2} I \right) (P + \delta P') + (P + \delta P') \left(\mathcal{A}_0 + \{\mathcal{A}, k + \delta v\} + \frac{\alpha}{2} I \right)^T + \\
& + \frac{\bar{w}^2}{\alpha} \left[D_k - \delta v_3 \begin{pmatrix} bc^T D \\ 0 \end{pmatrix} \right] \left[D_k - \delta v_3 \begin{pmatrix} bc^T D \\ 0 \end{pmatrix} \right]^T = 0,
\end{aligned}$$

откуда после динеаризации

$$\begin{aligned}
& \left(\mathcal{A}_0 + \{\mathcal{A}, k\} + \frac{\alpha}{2} I \right) (P + \delta P') + (P + \delta P') \left(\mathcal{A}_0 + \{\mathcal{A}, k\} + \frac{\alpha}{2} I \right)^T + \\
& + \delta \left(\{\mathcal{A}, v\} P + P \{\mathcal{A}, v\}^T \right) + \\
& + \frac{\bar{w}^2}{\alpha} \left[D_k D_k^T - \delta v_3 D_k \begin{pmatrix} bc^T D \\ 0 \end{pmatrix}^T - \delta v_3 \begin{pmatrix} bc^T D \\ 0 \end{pmatrix} D_k^T \right] = 0.
\end{aligned}$$

Почленно вычитая из полученного соотношения уравнение (12), приходим к уравнению

$$(П.3) \quad \left(\mathcal{A}_0 + \{\mathcal{A}, k\} + \frac{\alpha}{2} I \right) P' + P' \left(\mathcal{A}_0 + \{\mathcal{A}, k\} + \frac{\alpha}{2} I \right)^T + \\ + \{\mathcal{A}, v\} P + P \{\mathcal{A}, v\}^T - v_3 \frac{\bar{w}^2}{\alpha} \left[D_k \begin{pmatrix} bc^T D \\ 0 \end{pmatrix}^T + \begin{pmatrix} bc^T D \\ 0 \end{pmatrix} D_k^T \right] = 0.$$

Аналогично, $Y = Y(k)$ есть решение уравнения Ляпунова (15); запишем его в приращениях по направлению v :

$$\left(\mathcal{A}_0 + \{\mathcal{A}, k + \delta v\} + \frac{\alpha}{2} I \right)^T (Y + \delta Y') + \\ + (Y + \delta Y') \left(\mathcal{A}_0 + \{\mathcal{A}, k + \delta v\} + \frac{\alpha}{2} I \right) + \mathcal{C}^T \mathcal{C} = 0$$

или

$$\left(\mathcal{A}_0 + \{\mathcal{A}, k\} + \frac{\alpha}{2} I \right)^T (Y + \delta Y') + (Y + \delta Y') \left(\mathcal{A}_0 + \{\mathcal{A}, k\} + \frac{\alpha}{2} I \right) + \\ + \delta (\{\mathcal{A}, v\}^T Y + Y \{\mathcal{A}, v\}) + \mathcal{C}^T \mathcal{C} = 0.$$

Почленно вычитая из полученного соотношения уравнение (15), приходим к соотношению (17).

Из (П.3) и (17) имеем соотношение

$$\text{tr } P' Y \{\mathcal{A}, v\} = \text{tr } Y' \left[\{\mathcal{A}, v\} P - v_3 \frac{\bar{w}^2}{\alpha} D_k \begin{pmatrix} bc^T D \\ 0 \end{pmatrix}^T \right] = \\ = \text{tr } P Y' \{\mathcal{A}, v\} - v_3 \frac{\bar{w}^2}{\alpha} \text{tr } Y' D_k \begin{pmatrix} bc^T D \\ 0 \end{pmatrix}^T,$$

так что

$$\frac{1}{2} (f''(k) v, v) = \\ = \text{tr } (P Y' + P' Y) \{\mathcal{A}, v\} + \rho(v, v) - \frac{\bar{w}^2}{\alpha} v_3 \text{tr} \left[Y' D_k - v_3 Y \begin{pmatrix} bc^T D \\ 0 \end{pmatrix} \right] \begin{pmatrix} bc^T D \\ 0 \end{pmatrix}^T = \\ = 2 \text{tr } P Y' \{\mathcal{A}, v\} + \rho(v, v) - v_3 \frac{\bar{w}^2}{\alpha} \text{tr } Y' D_k \begin{pmatrix} bc^T D \\ 0 \end{pmatrix}^T - \\ - \frac{\bar{w}^2}{\alpha} v_3 \text{tr} \left[Y' D_k - v_3 Y \begin{pmatrix} bc^T D \\ 0 \end{pmatrix} \right] \begin{pmatrix} bc^T D \\ 0 \end{pmatrix}^T = \\ = 2 \text{tr } P Y' \{\mathcal{A}, v\} + \rho(v, v) - \frac{\bar{w}^2}{\alpha} v_3 \text{tr} \left[2 Y' D_k - v_3 Y \begin{pmatrix} bc^T D \\ 0 \end{pmatrix} \right] \begin{pmatrix} bc^T D \\ 0 \end{pmatrix}^T.$$

Лемма 2 доказана.

Доказательство леммы 3. Рассмотрим последовательность стабилизирующих регуляторов $\{k^{(j)}\} \in \mathcal{S}$ такую, что $k^{(j)} \rightarrow k \in \partial\mathcal{S}$, т.е. $\sigma(A_k) = 0$. Это означает, что для любого $\epsilon > 0$ найдется число $N = N(\epsilon)$ такое, что неравенство

$$|\sigma(A_{k^{(j)}}) - \sigma(A_k)| = \sigma(A_{k^{(j)}}) < \epsilon$$

справедливо для всех $j \geq N(\epsilon)$.

Пусть P_j – решение уравнения Ляпунова (12), ассоциированного с регулятором $k^{(j)}$:

$$\left(A_{k^{(j)}} + \frac{\alpha_j}{2}I\right)P_j + P_j\left(A_{k^{(j)}} + \frac{\alpha_j}{2}I\right)^T + \frac{\bar{w}^2}{\alpha_j}(D_{k^{(j)}}D_{k^{(j)}}^T + \delta I) = 0,$$

а Y_j – решение двойственного к нему уравнения Ляпунова

$$\left(A_{k^{(j)}} + \frac{\alpha_j}{2}I\right)^T Y_j + Y_j \left(A_{k^{(j)}} + \frac{\alpha_j}{2}I\right) + C^T C + \varepsilon I = 0.$$

Тогда

$$\begin{aligned} f(k^{(j)}) &= \text{tr } P_j(C^T C + \varepsilon I) + \rho \|k^{(j)}\|^2 \geq \\ &\geq \text{tr } P_j(C^T C + \varepsilon I) = \text{tr } Y_j \frac{\bar{w}^2}{\alpha_j} (D_{k^{(j)}} D_{k^{(j)}}^T + \delta I) \geq \\ &\geq \frac{\bar{w}^2}{\alpha_j} \lambda_{\min}(Y_j) \text{tr } (D_{k^{(j)}} D_{k^{(j)}}^T + \delta I) \geq \frac{\bar{w}^2}{\alpha_j} \frac{\lambda_{\min}(C^T C + \varepsilon I)}{2 \|A_{k^{(j)}} + \frac{\alpha_j}{2}I\|} \|D_{k^{(j)}}\|_F^2 \geq \\ &\geq \frac{\bar{w}^2}{4\sigma(A_{k^{(j)}})} \frac{\varepsilon}{\|A_{k^{(j)}}\| + \sigma(A_{k^{(j)}})} \|D_{k^{(j)}}\|_F^2 \geq \frac{\bar{w}^2}{4\epsilon} \frac{\varepsilon}{\|A_{k^{(j)}}\| + \epsilon} \|D_{k^{(j)}}\|_F^2 \xrightarrow{\epsilon \rightarrow 0} +\infty, \end{aligned}$$

поскольку

$$0 < \alpha_j < 2\sigma(A_{k^{(j)}})$$

и

$$\|A_{k^{(j)}} + \frac{\alpha_j}{2}I\| \leq \|A_{k^{(j)}}\| + \frac{\alpha_j}{2} \leq \|A_{k^{(j)}}\| + \sigma(A_{k^{(j)}}).$$

С другой стороны,

$$f(k^{(j)}) = \text{tr } P_j(C^T C + \varepsilon I) + \rho \|k^{(j)}\|^2 \geq \rho \|k^{(j)}\|^2 \xrightarrow{\|k^{(j)}\| \rightarrow +\infty} +\infty.$$

Лемма 3 доказана.

Доказательство леммы 4. Установим несколько полезных для дальнейшего оценок. Прежде всего,

$$(П.4) \quad \|\{\mathcal{A}, v\}\| = \left\| \sum_i \mathcal{A}_i v_i \right\| \leq \sum_i \|\mathcal{A}_i\| |v_i| \leq \max_i \|\mathcal{A}_i\| \|v\|_1 \leq \sqrt{3} \max_i \|\mathcal{A}_i\| \|v\|,$$

поскольку $\|a\|_1 \leq \sqrt{n}\|a\|$ для $a \in \mathbb{R}^n$.

С учетом (П.4) оценка для α устанавливается следующим образом:

$$(П.5) \quad \alpha < 2\sigma(\mathcal{A}_0 + \{\mathcal{A}, k\}) \leq 2\|\mathcal{A}_0 + \{\mathcal{A}, k\}\| \leq 2(\|\mathcal{A}_0\| + \|\{\mathcal{A}, k\}\|) \leq \\ \leq 2(\|\mathcal{A}_0\| + \sqrt{3} \max_i \|\mathcal{A}_i\| \|k\|) \leq 2\left(\|\mathcal{A}_0\| + \max_i \|\mathcal{A}_i\| \sqrt{\frac{3}{\rho} f(k_0)}\right),$$

так как

$$\|k\| \leq \sqrt{\frac{f(k)}{\rho}} \leq \sqrt{\frac{f(k_0)}{\rho}}.$$

Наконец,

$$(П.6) \quad \|D_k\|_F = \left\| \begin{pmatrix} (I - k_3 bc^T)D \\ 0 \end{pmatrix} \right\|_F = \|(I - k_3 bc^T)D\|_F \leq \|I - k_3 bc^T\| \|D\|_F \leq \\ \leq (1 + |k_3| \|bc^T\|) \|D\|_F \leq \left(1 + \sqrt{\frac{f(k_0)}{\rho}} \|bc^T\|\right) \|D\|_F.$$

Применяя к (16) лемму П.2, имеем

$$(П.7) \quad \frac{1}{2} \|f''(k)\| = \frac{1}{2} \sup_{\|v\|=1} (f''(k)v, v) \leq 2 \sup_{\|v\|=1} |\operatorname{tr} PY' \{\mathcal{A}, v\}| + \rho \sup_{\|v\|=1} |(v, v)| + \\ + \frac{\bar{w}^2}{\alpha} \sup_{\|v\|=1} \left| v_3 \operatorname{tr} \left[2Y'D_k - v_3 Y \begin{pmatrix} bc^T D \\ 0 \end{pmatrix} \right] \begin{pmatrix} bc^T D \\ 0 \end{pmatrix}^T \right| \leq \\ \leq \|P\|_F \|Y'\|_F \sup_{\|v\|=1} \|\{\mathcal{A}, v\}\| + \rho + \bar{w}^2 \sup_{\|v\|=1} |v_3| \sup_{\|v\|=1} \left\| \frac{2}{\alpha} Y'D_k - \frac{v_3}{\alpha} Y \begin{pmatrix} bc^T D \\ 0 \end{pmatrix} \right\|_F \times \\ \times \left\| \begin{pmatrix} bc^T D \\ 0 \end{pmatrix} \right\|_F \leq \sqrt{3} \|P\|_F \|Y'\|_F \max_i \|\mathcal{A}_i\| + \rho + \\ + \bar{w}^2 \left(2 \frac{1}{\alpha} \|Y'D_k\|_F + \sup_{\|v\|=1} |v_3| \frac{1}{\alpha} \left\| Y \begin{pmatrix} bc^T D \\ 0 \end{pmatrix} \right\|_F \right) \|bc^T D\|_F \leq \\ \leq \sqrt{3} \underbrace{\|P\|_F}_{(П.8)} \underbrace{\|Y'\|_F}_{(П.12)} \max_i \|\mathcal{A}_i\| + \rho + \\ + \bar{w}^2 \left(2 \frac{1}{\alpha} \underbrace{\|Y'\|}_{(П.11)} \underbrace{\|D_k\|_F}_{(П.6)} + \frac{1}{\alpha} \underbrace{\|Y\|}_{(П.9)} \|bc^T D\|_F \right) \|bc^T D\|_F.$$

Имеем следующую оценку для $\|P\|$:

$$\varepsilon \|P\| \leq \lambda_{\min}(\mathcal{C}^T \mathcal{C} + \varepsilon I) \|P\| \leq \operatorname{tr} P(\mathcal{C}^T \mathcal{C} + \varepsilon I) = f(k) - \rho \|k\|^2 \leq f(k) \leq f(k_0),$$

откуда

$$\|P\| \leq \frac{f(k_0)}{\varepsilon}$$

и

$$(II.8) \quad \|P\|_F \leq \sqrt{n+1} \frac{f(k_0)}{\varepsilon}.$$

Далее, из уравнений (18) и (15) в силу леммы II.1

$$\text{tr } Y(D_k D_k^T + \delta I) = \text{tr } P(\mathcal{C}^T \mathcal{C} + \varepsilon I),$$

откуда

$$\begin{aligned} \frac{\bar{w}^2 \delta}{\alpha} \|Y\| &\leq \frac{\bar{w}^2}{\alpha} \lambda_{\min}(D_k D_k^T + \delta I) \text{tr } Y \leq \text{tr } Y \frac{\bar{w}^2}{\alpha} (D_k D_k^T + \delta I) = \\ &= \text{tr } P(\mathcal{C}^T \mathcal{C} + \varepsilon I) = f(k) - \rho \|k\|^2 \leq f(k) \leq f(k_0), \end{aligned}$$

так что

$$(II.9) \quad \frac{1}{\alpha} \|Y\| \leq \frac{f(k_0)}{\bar{w}^2 \delta}$$

и с учетом (II.5)

$$(II.10) \quad \|Y\| \leq \frac{\alpha}{\bar{w}^2 \delta} f(k_0) \leq \frac{2f(k_0)}{\bar{w}^2 \delta} \left(\|\mathcal{A}_0\| + \max_i \|\mathcal{A}_i\| \sqrt{\frac{3}{\rho} f(k_0)} \right).$$

При этом, с учетом леммы II.2 и оценок (II.4), (II.10) для $\|v\| = 1$ имеем

$$\begin{aligned} &\lambda_{\max}(\{\mathcal{A}, v\}^T Y + Y \{\mathcal{A}, v\}) = \\ &= \|\{\mathcal{A}, v\}^T Y + Y \{\mathcal{A}, v\}\| \leq \|Y^2 + \{\mathcal{A}, v\}^T \{\mathcal{A}, v\}\| \leq \\ &\leq \|Y\|^2 + \|\{\mathcal{A}, v\}\|^2 \leq \frac{4f^2(k_0)}{\bar{w}^4 \delta^2} \left(\|\mathcal{A}_0\| + \max_i \|\mathcal{A}_i\| \sqrt{\frac{3}{\rho} f(k_0)} \right)^2 + 3 \max_i \|\mathcal{A}_i\|^2. \end{aligned}$$

Из уравнений (18) и (17) в силу леммы II.1 имеем

$$\text{tr } Y' \frac{\bar{w}^2}{\alpha} (D_k D_k^T + \delta I) = \text{tr } P(\{\mathcal{A}, v\}^T Y + Y \{\mathcal{A}, v\}),$$

откуда

$$\begin{aligned} \frac{\bar{w}^2 \delta}{\alpha} \|Y'\| &\leq \frac{\bar{w}^2}{\alpha} \lambda_{\min}(D_k D_k^T + \delta I) \text{tr } Y' \leq \text{tr } Y' \frac{\bar{w}^2}{\alpha} (D_k D_k^T + \delta I) = \\ &= \text{tr } P(\{\mathcal{A}, v\}^T Y + Y \{\mathcal{A}, v\}) \leq \lambda_{\max}(\{\mathcal{A}, v\}^T Y + Y \{\mathcal{A}, v\}) \text{tr } P \leq \\ &\leq \left[\frac{4f^2(k_0)}{\bar{w}^4 \delta^2} \left(\|\mathcal{A}_0\| + \max_i \|\mathcal{A}_i\| \sqrt{\frac{3}{\rho} f(k_0)} \right)^2 + 3 \max_i \|\mathcal{A}_i\|^2 \right] (n+1) \|P\| \leq \\ &\leq \frac{(n+1)f(k_0)}{\varepsilon} \left[\frac{4f^2(k_0)}{\bar{w}^4 \delta^2} \left(\|\mathcal{A}_0\| + \max_i \|\mathcal{A}_i\| \sqrt{\frac{3}{\rho} f(k_0)} \right)^2 + 3 \max_i \|\mathcal{A}_i\|^2 \right], \end{aligned}$$

так что

$$(П.11) \quad \frac{1}{\alpha} \|Y'\| \leq \frac{(n+1)f(k_0)}{\bar{w}^2 \varepsilon \delta} \left[\frac{4f^2(k_0)}{\bar{w}^4 \delta^2} \left(\|\mathcal{A}_0\| + \max_i \|\mathcal{A}_i\| \sqrt{\frac{3}{\rho} f(k_0)} \right)^2 + 3 \max_i \|\mathcal{A}_i\|^2 \right].$$

Соответственно с учетом (П.5)

$$\begin{aligned} \|Y'\| &\leq \alpha \frac{(n+1)f(k_0)}{\bar{w}^2 \varepsilon \delta} \left[\frac{4f^2(k_0)}{\bar{w}^4 \delta^2} \left(\|\mathcal{A}_0\| + \max_i \|\mathcal{A}_i\| \sqrt{\frac{3}{\rho} f(k_0)} \right)^2 + 3 \max_i \|\mathcal{A}_i\|^2 \right] \leq \\ &\leq \frac{2(n+1)f(k_0)}{\bar{w}^2 \varepsilon \delta} \left[\frac{4f^2(k_0)}{\bar{w}^4 \delta^2} \left(\|\mathcal{A}_0\| + \max_i \|\mathcal{A}_i\| \sqrt{\frac{3}{\rho} f(k_0)} \right)^2 + 3 \max_i \|\mathcal{A}_i\|^2 \right] \times \\ &\quad \times \left(\|\mathcal{A}_0\| + \max_i \|\mathcal{A}_i\| \sqrt{\frac{3}{\rho} f(k_0)} \right) \end{aligned}$$

и

$$(П.12) \quad \begin{aligned} \|Y'\|_F &\leq \sqrt{n+1} \|Y'\| \leq \\ &\leq \frac{2(n+1)^{3/2} f(k_0)}{\bar{w}^2 \varepsilon \delta} \left[\frac{4f^2(k_0)}{\bar{w}^4 \delta^2} \left(\|\mathcal{A}_0\| + \max_i \|\mathcal{A}_i\| \sqrt{\frac{3}{\rho} f(k_0)} \right)^2 + 3 \max_i \|\mathcal{A}_i\|^2 \right] \times \\ &\quad \times \left(\|\mathcal{A}_0\| + \max_i \|\mathcal{A}_i\| \sqrt{\frac{3}{\rho} f(k_0)} \right). \end{aligned}$$

Возвращаясь к (П.7), с учетом полученных оценок окончательно приходим к соотношению

$$\begin{aligned} \frac{1}{2} \|f''(k)\| &\leq \sqrt{3(n+1)} \frac{f(k_0)}{\varepsilon} \frac{2(n+1)^{3/2} f(k_0)}{\bar{w}^2 \varepsilon \delta} \times \\ &\times \left[\frac{4f^2(k_0)}{\bar{w}^4 \delta^2} \left(\|\mathcal{A}_0\| + \max_i \|\mathcal{A}_i\| \sqrt{\frac{3}{\rho} f(k_0)} \right)^2 + 3 \max_i \|\mathcal{A}_i\|^2 \right] \times \\ &\times \left(\|\mathcal{A}_0\| + \max_i \|\mathcal{A}_i\| \sqrt{\frac{3}{\rho} f(k_0)} \right) \max_i \|\mathcal{A}_i\| + \rho + \\ &+ \bar{w}^2 \left(2 \frac{(n+1)f(k_0)}{\bar{w}^2 \varepsilon \delta} \left[\frac{4f^2(k_0)}{\bar{w}^4 \delta^2} \left(\|\mathcal{A}_0\| + \max_i \|\mathcal{A}_i\| \sqrt{\frac{3}{\rho} f(k_0)} \right)^2 + 3 \max_i \|\mathcal{A}_i\|^2 \right] \times \right. \\ &\quad \left. \times \left(1 + \sqrt{\frac{f(k_0)}{\rho}} \|bc^T\| \right) \|D\|_F + \frac{f(k_0)}{\bar{w}^2 \delta} \|bc^T D\|_F \right) \|bc^T D\|_F. \end{aligned}$$

Лемма 4 доказана.

СПИСОК ЛИТЕРАТУРЫ

1. The IFAC Newsletter. No. 2. April, 2019.

https://www.ifac-control.org/newsletter_archive/IFAC_Newsletter_2019_2_April.pdf

2. *Krohling R.A., Rey J.P.* Design of Optimal Disturbance Rejection PID Controllers Using Genetic Algorithms // IEEE Transactions on Evolutionary Computation. 2001. V. 5. No. 1. P. 78–82.
3. *Kim D.H., Cho J.H.* Robust Tuning for Disturbance Rejection of PID Controller Using Evolutionary Algorithm // Proc. 2004 IEEE Annual Meeting of the North American Fuzzy Information Processing Society (NAFIPS'04). Banff, Canada, 2004. V. 1. P. 248–253.
4. *Han J.* From PID to Active Disturbance Rejection Control // IEEE Transactions on Industrial Electronics. 2009. V. 56. No. 3. P. 900–906.
5. *Teppa-Garran P.A., Garcia G.* Optimal Tuning of PI/PID/PID⁽ⁿ⁻¹⁾ Controllers in Active Disturbance Rejection Control // J. Control Engineer. Appl. Inform. 2013. V. 15. No. 4. P. 26–36.
6. *Vrančić D., Strmčnik S., Juričić D.* A Magnitude Optimum Multiple Integration Tuning Method for Filtered PID Controller // Automatica. 2001. V. 37. Iss. 9. P. 1473–1479.
7. *Vrančić D., Strmčnik S., Kocijan J., de Moura Oliveira P.B.* Improving Disturbance Rejection of PID Controllers by Means of the Magnitude Optimum Method // ISA Transactions. 2010. V. 49. No. 1. P. 47–56.
8. *Rivera D.E., Morari M., Skogestad S.* Internal Model Control: PID Controller Design // Industrial and Engineering Chemistry Process Design and Development. 1986. V. 25. No. 1. P. 252–265.
9. *Saxena S., Hote Y.V.* Simple Approach to Design PID Controller via Internal Model Control // Arab. J. Sci. Engineer. 2016. V. 41. P. 3473–3489.
10. *Zhou C., Shen Y.* A PID Control Method Based on Internal Model Control to Suppress Vibration of the Transmission Chain of Wind Power Generation System // Energies. 2022. V. 15. No. 16. P. 5919.
11. *Borase R.P., Maghade D.K., Sondkar S.Y., Pawar S.N.* A Review of PID Control, Tuning Methods and Applications // Int. J. Dynam. Control. 2021. V. 9. P. 818–827.
12. *Поляк Б.Т., Хлебников М.В.* Синтез статического регулятора для подавления внешних возмущений как задача оптимизации // АиТ. 2021. № 9. С. 86–115.
Polyak B.T., Khlebnikov M.V. Static Controller Synthesis for Peak-to-Peak Gain Minimization as an Optimization Problem // Autom. Remote Control. 2021. V. 82. No. 9. P. 1530–1553.
13. *Fatkhullin I., Polyak B.* Optimizing Static Linear Feedback: Gradient Method // SIAM J. Control Optimiz. 2021. V. 59. Iss. 5. P. 3887–3911.
14. *Поляк Б.Т., Хлебников М.В.* Новые критерии настройки ПИД-регуляторов // АиТ. 2022. № 11. С. 62–82.
Polyak B.T., Khlebnikov M.V. New Criteria for Tuning PID Controllers // Autom. Remote Control. 2022. V. 83. No. 11. P. 1724–1741.
15. *Хлебников М.В.* Синтез ПИ-регулятора для подавления внешних возмущений // АиТ. 2023. № 8. С. 3–23.
Khlebnikov M.V. PI Controller Design for Suppressing Exogenous Disturbances // Autom. Remote Control. 2023. V. 84. No. 8. P. 901–917.
16. *Boyd S., El Ghaoui L., Feron E., Balakrishnan V.* Linear Matrix Inequalities in System and Control Theory. Philadelphia: SIAM, 1994.

17. Поляк Б.Т., Хлебников М.В., Щербаков П.С. Управление линейными системами при внешних возмущениях: Техника линейных матричных неравенств. М.: ЛЕНАНД, 2014.
18. Чурилов А.Н., Гессен А.В. Исследование линейных матричных неравенств. Путеводитель по программным пакетам. СПб.: Изд-во Санкт-Петербургского ун-та, 2004.
19. *Frادkov A.* Passification of Non-square Linear Systems and Feedback Yakubovich-Kalman-Popov Lemma // Eur. J. Control. 2003. V. 9. Iss. 6. P. 577–586.
20. *Toker O., Ozbay H.* On the NP-hardness of Solving Bilinear Matrix Inequalities and Simultaneous Stabilization with Static Output Feedback // Proc. 1995 American Control Conference (ACC'95). Seattle, USA, June 21–23, 1995. V. 4. P. 2525–2526.
21. Поляк Б.Т. Введение в оптимизацию. 2-е изд. М.: УРСС, 2014.
22. *Åström K.J., Hägglund T.* Benchmark Systems for PID Control // IFAC Proceedings Volumes. 2000. V. 33. Iss. 4. P. 165–166.
23. *Grant M., Boyd S.* CVX: Matlab Software for Disciplined Convex Programming, version 2.1. URL <http://cvxr.com/cvx>
24. Хорн Р., Джонсон Ч. Матричный анализ. М.: Мир, 1989.

Статья представлена к публикации членом редколлегии А.Л. Фрадковым.

Поступила в редакцию 06.05.2025

После доработки 23.07.2025

Принята к публикации 15.08.2025

© 2025 г. В.А. МОЗЖЕЧКОВ, д-р техн. наук (v.a.moz@yandex.ru)
(Тульский государственный университет)

СИНТЕЗ РЕЛЕЙНЫХ РЕГУЛЯТОРОВ АВТОКОЛЕБАТЕЛЬНЫХ СИСТЕМ УПРАВЛЕНИЯ НА ОСНОВЕ ОБУЧАЮЩИХ ПРИМЕРОВ

Рассматривается задача синтеза релейных регуляторов в составе автоколебательной системы с линейным объектом управления. Необходимо обеспечить наличие в системе автоколебаний с заданными параметрами и приблизить ее поведение к желаемому, определяемому набором законов изменения во времени выхода системы, выступающих в качестве обучающих примеров. Учитываются ограничения на структуру регулятора и требование к степени устойчивости автоколебаний. Получены соотношения и предложен основанный на их применении итерационный метод решения рассматриваемой задачи, позволяющий синтезировать релейные регуляторы с простой структурой. Приведены примеры реализации предложенного метода.

Ключевые слова: релейный регулятор, автоколебания, обучающие примеры.

DOI: 10.31857/S0005231025110034

1. Введение

Релейное управление с обратной связью, анализ и синтез основанных на его применении автоколебательных систем управления – классические темы теории управления [1–9].

Актуальность задачи синтеза автоколебательных систем управления с релейным регулятором подтверждается наличием большого числа примеров практического использования таких систем различного назначения. К ним относятся: автоколебательные системы регулирования разнообразных физических величин [1], следящие системы [1, 2, 4, 5, 8], самонастраивающиеся и адаптивные системы, использующие автоколебания в режиме автонастройки регулятора [6–8], автоколебательные конверторы электрических напряжений и автогенераторы сигналов [1, 10], вибрационные гироскопы и акселерометры [11, 12], сенсоры на основе нанорезонаторов, резонисторов и кантилеверов [13, 14], вибрационные машины [15], системы управления неполноприводных многосвязных механизмов и роботов, использующие автоколебания [16], и многие другие устройства. Автоколебания в указанных системах являются рабочим режимом, в котором регулируемая величина колеблется с заданной частотой в допустимом диапазоне ее значений.

Длительное время преобладающим подходом к анализу и синтезу автоколебательных систем был подход, основанный на использовании частотных методов [1–3, 8]. Среди них наибольшую популярность, благодаря простоте применения, получил метод гармонической линеаризации [2, 3]. Существенно более точным и универсальным является частотный метод анализа автоколебаний, разработанный Я.З. Цыпкиным [1], основанный на использовании предложенного им понятия годографа релейной системы. Развитие методов анализа релейных систем в пространстве состояний позволило обеспечить более высокую эффективность процедур их анализа и синтеза, а также более детальный учет возможных режимов функционирования релейной системы в сравнении с частотными методами. В [4, 5] был предложен метод синтеза автоколебательных систем, основанный на использовании понятия фазового годографа, представляющего собой линию в пространстве состояний объекта управления, каждая точка x^* которой соответствует переключению реле с минуса на плюс при некотором свойственном ей периоде автоколебаний. Фазовый годограф остается неизменным при выборе коэффициентов обратной связи, что существенно упрощает синтез системы и позволяет обеспечить высокую точность реализации назначенных параметров автоколебаний. В [6, 7] получена аналитическая зависимость вектора x^* от периода автоколебаний, матриц и векторов параметров, присутствующих в описании линейного объекта управления в пространстве состояний, а также представлены алгебраические условия существования автоколебаний заданной частоты, включая условие их локальной устойчивости.

В [9] решена задача синтеза релейных регуляторов, обеспечивающих наличие в системе с линейным объектом управления автоколебаний заданной частоты и амплитуды и приближающих ее поведение к желаемому, определяемому заданной передаточной функцией. Предложенный в [9] метод синтеза основан на полиномиальном представлении гармонически линеаризованной синтезируемой системы и сведении процедуры синтеза к решению систем линейных алгебраических уравнений и неравенств. Следствием использования в методе [9] гармонической линеаризации является в некоторых случаях значительное отклонение частоты автоколебаний от ее заданного значения, что иногда приводит к существенному отличию поведения синтезированной системы от желаемого.

В настоящей работе с целью исключения недостатков метода [9] при решении задачи синтеза релейной автоколебательной системы используются результаты исследований [4–7], обеспечивающие более высокую точность предсказания параметров автоколебаний в сравнении с методом гармонической линеаризации. Синтез осуществляется согласно предлагаемому в статье методу, основанному на применении соотношений, полученных с использованием описания объекта управления в пространстве состояний, и являющемуся результатом развития и распространения алгоритма, изложенного в [17], на класс релейных автоколебательных систем управления. Желаемое поведение системы определяется набором законов изменения во времени ее выхо-

да, выступающих в качестве обучающих примеров, а также требованием к степени ее устойчивости. Регулятор реализует статическую обратную связь с релейным выходом. Предложен метод, позволяющий находить начальное приближение искомых значений коэффициентов обратной связи и осуществлять их итерационное уточнение. В общем случае, когда вектор выхода объекта управления содержит несколько измеряемых линейно независимых величин, решается задача синтеза регуляторов, имеющих простую структуру [18, 19], т.е. регуляторов, в которых отличны от нуля только те коэффициенты обратной связи, выбор значений которых, отличных от нуля, необходим и достаточен для придания системе желаемых свойств.

2. Постановка задачи

Объект управления задан системой уравнений

$$(1) \quad \dot{x} = Ax + Bu,$$

$$(2) \quad y = Cx,$$

где x – вектор состояния, $\dot{x} = dx/dt$, t – время, u – скалярное управляющее воздействие, y – вектор выхода, все его компоненты могут быть измерены и использованы в регуляторе, матрицы A , C и вектор B считаем заданными, их элементы, а также значения величины u и компонент векторов x , y – действительные числа, пара (A, B) является управляемой, а пара (A, C) – наблюдаемой.

Математическое описание регулятора имеет вид

$$(3) \quad \tilde{u} = K^T y,$$

$$(4) \quad u = \text{sgn}(\tilde{u}),$$

где скалярная величина $\tilde{u}$ является выходом линейной части (3) регулятора; вектор K коэффициентов обратной связи подлежит определению, его элементы – действительные числа, функция $\text{sgn}(\cdot)$ описывает реле, формирующее управляющее воздействие u , ее значением является -1 , $+1$ или действительное число из отрезка $[-1, 1]$ соответственно, если значение ее аргумента отрицательно, положительно или равно нулю. При этом равенство единице амплитудных значений u не снижает общности рассматриваемой задачи, поскольку реальные амплитуды управляющего воздействия можно учесть в (1) при определении вектора B .

Примем во внимание ограничения, накладываемые на структуру линейной части регулятора, которые обычно сводятся [20] к требованию равенства нулю некоторых компонент в векторе $K = (k_i)$, поэтому будем учитывать условие

$$(5) \quad k_i = 0, \quad \forall i \notin S,$$

где S – набор значений индексов тех коэффициентов обратной связи k_i , которые не стеснены требованием равенства их значений нулю. Набор S в описании регулятора (3)–(5) определяет перечень используемых в нем сигналов

обратных связей и с учетом (3)–(5) определяет его структуру. Полагаем, что структура регулятора при решении данной задачи может изменяться только в результате изменения набора S . Поэтому далее выбор структуры регулятора отождествляется с выбором набора S , а термины “структура регулятора” и “набор S ” используются как синонимы.

Далее решение разрывной системы (1)–(5) понимается в смысле Филиппова [21].

В системе (1)–(5) установившимся режимом функционирования должно быть асимптотически орбитально устойчивое периодическое движение, представляющее собой автоколебания с заданной частотой ω_0 и заданной амплитудой $\tilde{U}$ колебаний сигнала $\tilde{u}$. Автоколебаниям в фазовом пространстве соответствует изолированная замкнутая фазовая траектория – предельный цикл. В результате решения задачи синтеза регулятора предельный цикл L должен быть устойчивым, симметричным (каждой точке $x \in L$ должна соответствовать точка $-x \in L$) и простым (сигнал $\tilde{u}$ должен менять знак только дважды за период автоколебаний).

Требование наличия запаса устойчивости автоколебаний опишем условием

$$(6) \quad \eta(K) \geq \check{\eta},$$

где $\eta(K)$ – запас устойчивости автоколебаний, $\check{\eta}$ – заданный нижний предел допустимых значений запаса устойчивости. Математическое описание функции $\eta(K)$ представлено в разделе 3.

Желаемое поведение системы (1)–(5) определим, задав для начальных условий x_0^γ , $\gamma \in \{\overline{1, q}\}$ соответствующие желаемые траектории изменения во времени выхода (2) системы (1)–(5), представленные векторами $Y_\gamma = (y_k^\gamma)$, $k \in \{\overline{1, N}\}$ его значений в дискретные моменты времени $t = k\Delta t$, где Δt – шаг выборки, k – натуральное число. Величины N и Δt определяют отрезок времени $[0, N\Delta t]$, на котором заданы желаемые траектории. Задаваемые начальные состояния x_0^γ , $\gamma \in \{\overline{1, q}\}$ должны быть достаточно удаленными от траектории установившегося движения, чтобы переходный процесс имел длительность по времени, существенно превосходящую период автоколебаний. Кроме того, задаваемые начальные состояния должны существенно различаться между собой. Целесообразно назначать $q \in [1, n]$. Набор $Q = \{(x_0^\gamma, Y_\gamma)\}$, $\gamma \in \{\overline{1, q}\}$ является набором обучающих примеров и должен определять желаемое поведение синтезируемой системы при переходе системы от возмущенного к установившемуся режиму движения. Требование соответствия системы (1)–(5) желаемому поведению запишем в виде

$$(7) \quad \varepsilon_k^{\gamma-} \leq y(x_0^\gamma, K)_k - y_k^\gamma \leq \varepsilon_k^{\gamma+}, \quad \forall k \in \{\overline{1, N}\}, \quad \forall \gamma \in \{\overline{1, q}\},$$

где $\varepsilon_k^{\gamma-}$, $\varepsilon_k^{\gamma+}$ – заданные постоянные векторы, $y(x_0^\gamma, K)_k$ – вектор выхода системы (1)–(5) в k -ый момент времени, соответствующий начальному условию $x(0) = x_0^\gamma$ и выбранному вектору K , символ $\leq$ здесь и далее применительно к векторам означает, что каждый элемент вектора, указанного слева

от него, меньше или равен соответствующему элементу вектора, указанного справа от него, т.е. обозначает покомпонентное неравенство векторов. Векторы $\varepsilon_k^{\gamma-}, \varepsilon_k^{\gamma+}$ можно задать, вычислив их координаты $\varepsilon_{ki}^{\gamma-}, \varepsilon_{ki}^{\gamma+}$, например, по формулам $\varepsilon_{ki}^{\gamma-} = -\delta_i m_i^{\gamma}, \varepsilon_{ki}^{\gamma+} = +\delta_i m_i^{\gamma}$, где δ_i – допустимое значение абсолютной величины относительной погрешности $(y(x_0^{\gamma}, K)_{ki} - y_{ki}^{\gamma})/m_i^{\gamma}$ воспроизведения в системе i -ой координаты желаемой траектории, m_i^{γ} – максимальное значение абсолютной величины i -ой координаты желаемой траектории y^{γ} . В более общем случае можно использовать при вычислении $\varepsilon_{ki}^{\gamma-}, \varepsilon_{ki}^{\gamma+}$ значения δ_i , различающиеся для различных моментов времени $k\Delta t$ и траекторий y^{γ} . Целесообразно назначать $\delta_i \in [1, 20]10^{-2}$.

Выбор вектора K осуществим из условия наилучшего приближения поведения системы (1)–(5) к желаемому в смысле минимизации евклидовой нормы вектора $\Delta y(K)$, составленного из разностей $y(x_0^{\gamma}, K)_k - y_k^{\gamma}$, т.е. из условия

$$(8) \quad |\Delta y(K)| \rightarrow \min_K,$$

где $|\cdot|$ – евклидова норма [22].

Рассматриваемая задача в случае, когда структура регулятора задана (указан фиксированный набор S , определяющий совместно с (3)–(5) структуру регулятора), состоит в нахождении вектора K , обеспечивающего выполнение системой управления (1)–(5) требований (6)–(8) для заданных значений ω_0 и $\tilde{U}$.

В общем случае, когда набор S не задан и вектор выхода y содержит несколько измеряемых линейно независимых величин, будем решать задачу структурного синтеза, состоящую в определении тех наборов S и соответствующих им векторов K , для которых при заданных ω_0 и $\tilde{U}$ выполняются условия (6)–(8) и структура регулятора (3), (5) является простой, что означает [18–19] отличие от нуля только тех коэффициентов k_i , выбор значений которых, отличных от нуля, необходим и достаточен для придания системе (1)–(5) желаемых свойств. Формально задача поиска множества Ω простых структур регулятора состоит в нахождении таких приемлемых структур $S \in \zeta$, для которых нельзя указать менее сложную приемлемую структуру $S' \in \zeta$ (структуру S' считаем проще, чем S , если $S' \subset S$), т.е. требуется найти

$$(9) \quad \Omega = \{S \in \zeta \mid \{S' \in \zeta \mid S' \subset S\} = \emptyset\},$$

где ζ – множество приемлемых структур, т.е. структур, для которых существует вектор K , обеспечивающий выполнение системой управления (1)–(5) требований (6)–(8), формула $\{S' \in \zeta \mid S' \subset S\} = \emptyset$ указывает на отсутствие приемлемой структуры S' , которая проще, чем структура $S \in \Omega$.

3. Анализ задачи

Согласно [5, 6] для существования в системе (1)–(5) простого симметричного предельного цикла с периодом $2h = 2\pi/\omega_0$ необходимо выполнение сле-

дующих условий:

$$(10) \quad x^* = -(I + e^{Ah})^{-1} \int_0^h e^{A(h-\tau)} B d\tau,$$

$$(11) \quad K^\top C x^* = 0,$$

$$(12) \quad K^\top C \left(e^{At} x^* + \int_0^t e^{A(t-\tau)} B d\tau \right) > 0, \quad 0 < t < h,$$

где x^* – точка предельного цикла, соответствующая смене знака $\tilde{u}$ с минуса на плюс и, следовательно, переключению реле (4) с -1 на $+1$, I – единичная матрица.

Предельный цикл локально устойчив [5–7] тогда и только тогда, когда меньше единицы абсолютные величины всех собственных чисел матрицы

$$(13) \quad W = (I - v K^\top C / (K^\top C v)) e^{Ah},$$

$$(14) \quad v = -A x^* + B.$$

Справедливость данного утверждения доказана в [6, 7] применительно к системам вида (1)–(5) посредством классического подхода к анализу устойчивости периодических движений на основе отображения Пуанкаре.

Требование (6) наличия заданного запаса устойчивости автоколебаний будем считать выполненным, если имеет место неравенство

$$(15) \quad \eta(K) = 1 - \rho(W(K)) \geq \check{\eta},$$

где $\rho(W(K))$ – спектральный радиус матрицы $W(K)$, при этом целесообразно назначать $\check{\eta} \in [1, 10]h/(N\Delta t)$.

Процессы в релейных системах, протекающие существенно медленнее в сравнении с процессом автоколебаний, принято называть [1, 2, 5] медленными процессами. В качестве приближенной модели таких процессов в случае, когда диапазон значений сигнала на входе реле незначительно превышает амплитуду $\tilde{U}$, часто используют модель, в которой описание реле (4) в случае устойчивого, простого симметричного предельного цикла заменяют линеаризованным описанием [1, 2, 5]

$$(16) \quad u = R\tilde{u}, \quad R = 4/(\pi\tilde{U}),$$

где R – коэффициент линеаризации. Уравнения (1)–(3), (5), (16) будем использовать как упрощенную модель медленных процессов в автоколебательной системе (1)–(5).

Полагаем, что набор обучающих примеров содержит описания процессов, которые для заданных параметров $\omega_0, \tilde{U}$ можно считать медленными. Тогда

задачу обучения системы (1)–(5) желаемому поведению можно заместить задачей обучения системы (1)–(3), (5), (16). С целью применения метода обучения [17] заместим систему (1)–(3), (16) ее аналогом с дискретным временем

$$(17) \quad x_{k+1} = \hat{A}x_k + \hat{B}u_k, \quad y_k = Cx_k, \quad u_k = RKy_k,$$

где k – дискретное время (натуральное число), $t = k\Delta t$, шаг дискретизации времени Δt выбираем с учетом условия $\Delta t \ll h$, векторы x_k, y_k, u_k являются дискретной аппроксимацией векторов x, y, u , при этом $\hat{A} = e^{A\Delta t}$; $\hat{B} = \int_0^{\Delta t} e^{At} B dt$.

Согласно [17] условия (7), (8) наилучшего приближения поведения системы (17) к желаемому, заданному набором обучающих примеров Q , эквивалентны условиям

$$(18) \quad \sum_{\gamma=1}^q |G_\gamma(K)_S K_S - \hat{Y}_\gamma|^2 \rightarrow \min_K,$$

$$(19) \quad \hat{Y}_\gamma + \varepsilon_\gamma^- \leq G_\gamma(K)_S K_S \leq \hat{Y}_\gamma + \varepsilon_\gamma^+, \quad \gamma \in \{\overline{1, q}\},$$

где $\hat{Y}_\gamma = Y_\gamma - Y_{0\gamma}$, при этом $Y_\gamma, Y_{0\gamma}, G_\gamma(K)$ – столбцы, составленные соответственно из блоков $y_k^\gamma, C\hat{A}^k x_0^\gamma, G_{k\gamma}(K) = C \sum_{i=0}^{k-1} \left(y(x_0^\gamma, K)_i^\top \otimes \hat{A}^{k-i-1} \hat{B} R \right)$, $k \in \{\overline{1, N}\}$; $\otimes$ – произведение Кронекера [23, с. 83], матрица $G_\gamma(K)_S$ и вектор K_S составлены соответственно из столбцов матрицы $G_\gamma(K)$ и координат вектора K , номера которых указаны в наборе S .

Система (17) отличается от системы управления, рассмотренной в [17], только наличием множителя R в выражении, определяющем u_k . Это отличие учтено в представленном выше выражении, определяющем $G_{k\gamma}(K)$.

Условие (11) с учетом (5) принимает вид линейного уравнения

$$(20) \quad (Cx^*)_S^\top K_S = 0,$$

в котором вектор $(Cx^*)_S$ составлен из координат вектора Cx^* , номера которых указаны в наборе S .

Дискретным аналогом требования (12) с учетом (5) является система линейных неравенств

$$(21) \quad H_{kS} K_S > 0, \quad \forall k \in \{\overline{1, \hat{h}}\},$$

где матрица H_{kS} составлена из указанных в S столбцов матрицы $H_k = (Ce^{Ak\Delta t} x^* + C \int_0^{k\Delta t} e^{A(k\Delta t - \tau)} B d\tau)^\top$, $\hat{h} = \text{fix}(h/\Delta t - 1)$, где $\text{fix}(\cdot)$ – функция округления до ближайшего целого числа.

Левая часть неравенства (12) определяет для режима автоколебаний значения сигнала $\tilde{y}$ в пределах полувольтны его колебаний, поэтому амплитуда $\tilde{U}$

сигнала $\tilde{u}$ близка значению величины $H_{kS}K_S$, когда k находится в середине отрезка $[1, \hat{h}]$. Для реализации заданной амплитуды $\tilde{U}$ колебаний сигнала $\tilde{u}$ в режиме простых симметричных автоколебаний потребуем выполнение условия

$$(22) \quad H_{kS}K_S = \tilde{U}, \quad k = \text{fix}((\hat{h} + 1)/2).$$

Таким образом, рассматриваемая задача нахождения вектора K , обеспечивающего выполнение системой управления (1)–(5) требований (6)–(8) для заданных значений ω_0 , $\tilde{U}$, сводится к задаче (18)–(22), (15). В ней условия (20)–(22) существования автоколебаний с заданными параметрами и условие (15) их устойчивости получены из нелинейной модели релейной системы (1)–(5) без привлечения упрощающих допущений, и только условия (18), (19) близости поведения синтезируемой системы к желаемому основаны на ее упрощенной линеаризованной модели (17).

4. Метод решения

Пусть структура регулятора задана, т.е. указан набор S . Искомый вектор K можно определить, решив, как показано выше, задачу (18)–(22), (15), которая эквивалентна задаче обучения статического регулятора, рассмотренной в [17]. Для ее решения воспользуемся предложенным в [17] методом, на каждой итерации которого решается задача о наименьших квадратах с линейными ограничениями [24, с. 225] (задача НКЛ), относящаяся к классу задач выпуклого программирования [25], для которого разработаны эффективные оптимизационные процедуры, гарантирующие получение решения либо констатацию его отсутствия (в системе Matlab для решения задач НКЛ предназначена функция `lsqlin`). Переход на очередной итерации в методе [17] к задаче НКЛ при решении рассматриваемой задачи основан на замене в (18), (19) матрицы $G_\gamma(K)_S$ на неизменную в пределах итерации матрицу $G_\gamma(\dot{K})_S$, соответствующую вектору $\dot{K}$, найденному на предыдущей итерации, а также на замене в (15) функции $\rho(W(K))$ ее линейной аппроксимацией $r_0 + r_1\dot{K}_S$ вблизи $\dot{K}$, т.е. на замене условия (15) линейным неравенством

$$(23) \quad r_0 + r_1\dot{K}_S \leq 1 - \tilde{\eta},$$

где r_0 , r_1 – коэффициенты линеаризации.

С целью определения начального приближения решения рассматриваемой задачи выделим ее важный частный случай, в котором ограничимся выполнением только условий (18), (20), (22). При этом упростим условие (18) на основе допущения о том, что все желаемые траектории в наборе Q принадлежат множеству решений синтезируемой системы. Тогда [17] можно заменить матрицу $G_\gamma(K)_S$ в (18) матрицей $\bar{G}_{\gamma S}$, не зависящей от K , и записать условие (18) в виде

$$(24) \quad \sum_{\gamma=1}^q |\bar{G}_{\gamma S}K_S - \hat{Y}_\gamma|^2 \rightarrow \min_K,$$

где матрица $\bar{G}_{\gamma S}$ составлена из указанных в S столбцов матрицы $\bar{G}_\gamma$, представляющей собой столбец блоков $\bar{G}_{k\gamma} = C \sum_{i=0}^{k-1} (y_i^{\gamma\top} \otimes \hat{A}^{k-i-1} \hat{B} R)$, $k \in \{\overline{1, N}\}$.

Известно [24], что задача квадратичного программирования с линейными ограничениями в виде равенств, частным случаем которой является задача (20), (22), (24), сводится к решению системы линейных уравнений, имеющей в рассматриваемом случае следующий вид:

$$(25) \quad \begin{pmatrix} F & L^\top \\ L & 0 \end{pmatrix} \begin{pmatrix} K_S \\ \lambda \end{pmatrix} = \begin{pmatrix} c \\ d \end{pmatrix},$$

где $F = \bar{G}_S^\top \bar{G}_S$, $\bar{G}_S$ – матрица системы уравнений $\bar{G}_{\gamma S} K_S = \hat{Y}_\gamma$, $\forall \gamma \in \{\overline{1, q}\}$, $L = ((Cx^*)^\top, H_{kS})^\top$, $c = \bar{G}_S^\top \hat{Y}_\gamma$, $d = (0, \tilde{U})^\top$, λ – вектор множителей Лагранжа.

Вектор K_S , получаемый в результате решения системы (25), может служить начальным приближением решения задачи (18), (22), (15).

Если с использованием начального приближения, определенного согласно (25), не удалось решить рассматриваемую задачу (18)–(22), (15), можно воспользоваться начальным приближением, рассчитанным с учетом требований, описываемых системой равенств (20), (22), (24) и неравенств (19), (21), (23). В таком случае, заменив (19) его приближенным аналогом – условием

$$(26) \quad \hat{Y}_\gamma + \varepsilon_\gamma^- \leq \bar{G}_{\gamma S} K_S \leq \hat{Y}_\gamma + \varepsilon_\gamma^+, \quad \gamma \in \{\overline{1, q}\},$$

находим начальное приближение, решив задачу НКЛ (20)–(24), (26).

Найденное начальное приближение уточняется на последующих итерациях изложенного выше метода без привлечения допущения о принадлежности траекторий, составляющих набор Q , множеству решений синтезируемой системы.

Задача поиска множества Ω простых структур регулятора может быть решена методом, предназначенным для задач синтеза простых структур общего вида [19], в котором в качестве процедуры оценки приемлемости структуры регулятора и нахождения соответствующего ей вектора K может использоваться описанная выше процедура решения задачи (18)–(22), (15).

5. Примеры

Пример 1. Решим задачу синтеза релейного регулятора автоколебательной системы управления электроприводом. Объект управления, включающий в себя двигатель постоянного тока с неизменным потоком возбуждения, редуктор и инерционную нагрузку, описывается уравнениями (1)–(2), в которых:

$$A = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & C_m/J_m \\ 0 & -C_e/L_e & -R_e/L_e \end{pmatrix}, \quad B = \begin{pmatrix} 0 \\ 0 \\ U/L_e \end{pmatrix}, \quad C = \begin{pmatrix} 1/k_r & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix},$$

где R_e , L_e – сопротивление и индуктивность якорной обмотки двигателя; C_e , C_m – коэффициент противо ЭДС и коэффициент момента двигателя; U – напряжение питания; k_r – коэффициент передачи редуктора; J_m – момент инерции подвижных частей, приведенный к валу двигателя. В международной системе единиц: $R_e = 0,475$; $L_e = 5,7 \times 10^{-4}$; $C_e = C_m = 6,83 \times 10^{-2}$; $U = 27$; $k_r = 2$; $J_m = 9,43 \times 10^{-5}$. Проверка показывает, что пара (A, B) является управляемой, а пара (A, C) – наблюдаемой. Вектор состояний $x = (x_1, x_2, x_3)$, где x_1 , x_2 – угловое положение и угловая скорость вращения вала двигателя, x_3 – ток в якорной обмотке. Вектор выхода $y = (y_1, y_2, y_3) = (x_1/k_r, x_2, x_3)$, в нем y_1 – регулируемая величина (угловое положение выходного вала электропривода), $S = \{1, 2, 3\}$. Назначаем $\tilde{U} = 0,01$ В. Считаем, что амплитуда автоколебаний регулируемой величины y_1 должна быть не более 0,1 мрад. Для выполнения этого требования, как следует из анализа амплитудно-частотных характеристик объекта управления, частота автоколебаний должна быть не менее 1820 Гц, поэтому принимаем $f = 2000$ Гц, при этом $\omega_0 = 2\pi f = 12\,566$ рад/с.

Требуется определить значения коэффициентов обратных связей, составляющих искомый вектор K .

Сформируем набор обучающих примеров, определяющих желаемое поведение синтезируемой системы при переходе системы от возмущенного к установившемуся режиму движения. Согласно представленной в разделе 2 рекомендации назначать $q \in [1, n]$ принимаем число обучающих примеров $q = 2$. В качестве примеров желаемого поведения системы укажем оптимальные по критерию минимальной энергии управления траектории движения объекта управления (1), (2) из существенно различных начальных состояний $x_0^1 = (1; 0; 0)$ и $x_0^2 = (0; 10; -10)$ в начало координат за время 0,025 с. Расчет таких траекторий можно осуществить, используя известные зависимости [22, с. 128]. Указанные траектории Y_1, Y_2 в паре с соответствующими начальными условиями на интервале времени от 0 до 0,05 с составляют набор обучающих примеров $Q = \{(x_0^1, Y_1), (x_0^2, Y_2)\}$. При расчете траекторий Y_1, Y_2 был выбран согласно представленной в разделе 3 рекомендации, шаг дискретизации времени $\Delta t = 2,5 \times 10^{-5} \ll h = 2,5 \times 10^{-4}$, ему соответствует значение $N = 2000$. Задача вычисления этих траекторий имеет решение в силу управляемости и наблюдаемости рассматриваемого объекта управления.

Вначале решим рассматриваемую задачу синтеза, не используя ограничений (6), (7), что эквивалентно решению задачи (20), (22), (24). В качестве начального приближения принимаем решение системы (25). Воспользовавшись методом, изложенным в разделе 4.1, после трех итераций находим $K = (-5,11; -3,73 \times 10^{-2}; 1,06 \times 10^{-4})$, при этом запас устойчивости $\eta(K) = 0,017$, значение целевой функции (18) равно $2,1 \times 10^{-6}$. Полученные значения $\tilde{U}$, ω_0 равны соответственно 0,010 В и 12 566 рад/с. Графики переходных процессов в системе (1)–(5), соответствующие найденному вектору K , представлены на рис. 1 для $x(0) = x_0^1$ и на рис. 2 для $x(0) = x_0^2 = (-0,5; -10; +10)$. Рисунок 2 демонстрирует поведение системы, соответ-

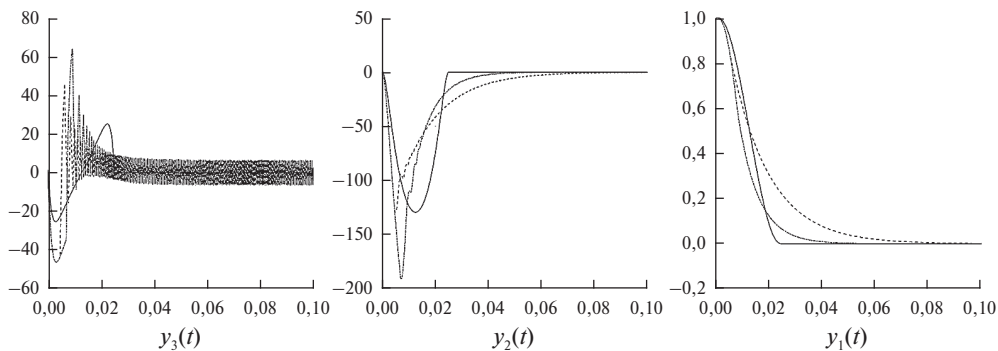


Рис. 1. Графики процессов для $x(0) = x_0^1$ (желаемый процесс и процессы, полученные в примерах 1, 2, показаны соответственно сплошной, пунктирной и штрихпунктирной линией).

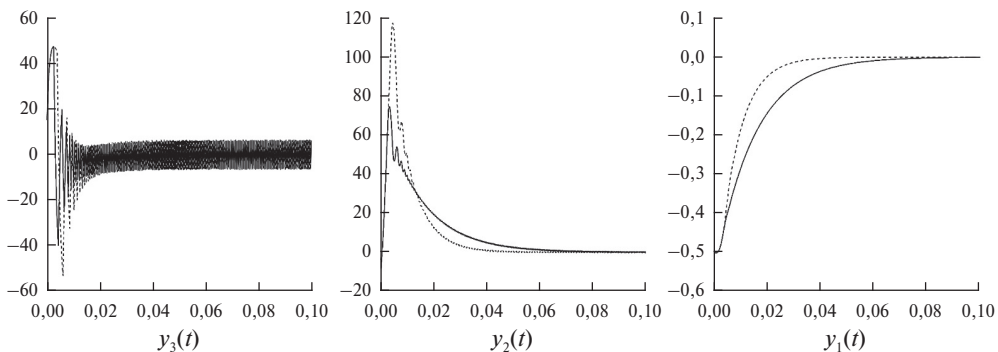


Рис. 2. Графики процессов для $x(0) = x_0^3$ (процессы, полученные в примерах 1, 2, показаны соответственно сплошной и пунктирной линией).

ствующее начальному состоянию x_0^3 , существенно удаленному от начальных состояний x_0^1 и x_0^2 , указанных в исходных данных рассматриваемого примера.

Пример 2. Решим задачу примера 1, в исходных данных которой увеличим запас устойчивости $\eta(K)$ не менее, чем до $\tilde{\eta} = 0,030$ и потребуем уменьшить отклонение координат y_1, y_2 вектора выхода от их желаемых значений, назначив для них $\delta_1 = 0,05, \delta_2 = 0,15$ на интервале времени от 0,025 до 0,05 с и $\delta_1 = 0,1, \delta_2 = 0,2$ на интервале времени от 0 до 0,025 с. Расчет векторов $\varepsilon_k^{\gamma-}, \varepsilon_k^{\gamma+}$, соответствующих указанным значениям δ_1, δ_2 , произведен согласно пояснениям к (7). Выбор значения $\tilde{\eta} = 0,030$ осуществлен согласно пояснениям к (15) по формуле $\tilde{\eta} = 6h/(N\Delta t)$, где $h = 1/(2f) = 1/4000$ с, $N\Delta t = 0,05$ с.

С учетом указанных требований решаем задачу (18)–(22), (15). В качестве начального приближения принимаем решение задачи НКЛ (20)–(24), (26).

Методом раздела 4.1 в результате трех итераций находим

$$K = (-9,81; -3,76 \times 10^{-2}; 9,75 \times 10^{-5}).$$

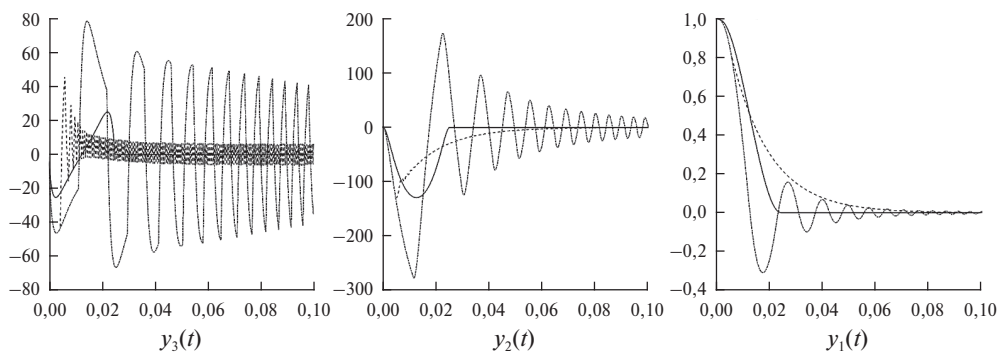


Рис. 3. Графики процессов для $x(0) = x_0^1$ (желаемый процесс и процессы, полученные в примерах 1, 4, показаны соответственно сплошной, пунктирной и штрихпунктирной линией).

При этом $\eta(K) = 0,0323$, условия (6), (7) выполняются, значение целевой функции (18) составило $4,1 \times 10^{-6}$.

Графики переходных процессов, соответствующие найденному вектору K , представлены на рис. 1 для $x(0) = x_0^1$ и на рис. 2 для $x(0) = x_0^3$.

Пример 3. Изменим постановку задачи примера 1: решим задачу структурного синтеза, состоящую в определении тех наборов S и соответствующих им векторов K , для которых при заданных $\omega_0, \tilde{U}$ структура регулятора согласно (9) является простой. Искомое множество Ω найдем методом синтеза простых структур, представленным в [19, с. 19] для “задач общего вида”. На первом шаге проверяем приемлемость структур $S \subseteq \{1, 2, 3\}$. Структура S приемлема (см. пояснения к (9)), если для нее удастся найти вектор K , обеспечивающий выполнение системой управления (1)–(5) требований (6)–(8). Проверку наличия такого вектора K выполняем согласно процедуре решения задачи (18)–(22), (15), изложенной в разделе 4. Проверка структуры $S = \{1, 2, 3\}$ показала ее приемлемость. Проверка всех структур $S \subseteq \{1, 2, 3\}$, соответствующих наборам из двух компонент, т.е. наборам $\{1, 2\}$, $\{1, 3\}$, $\{2, 3\}$, показала их неприемлемость. Как следствие, заведомо неприемлемы и не требуют соответствующей проверки все наборы $\{1\}$, $\{2\}$, $\{3\}$, являющиеся подмножествами неприемлемых наборов $\{1, 2\}$, $\{1, 3\}$, $\{2, 3\}$ [19, утверждение 8]. Таким образом, в данном случае приемлемой и простой оказалась структура $S = \{1, 2, 3\}$.

Пример 4. Решим задачу примера 3, в исходных данных которой уменьшим заданный нижний предел запаса устойчивости до $\check{\eta} = 2,5 \times 10^{-3}$. В результате получаем множество Ω , содержащее единственный набор $S = \{1, 2\}$, определяющий структуру регулятора, использующего сигналы обратной связи по положению y_1 и скорости вращения y_2 выходного вала привода. При этом структура $S = \{1, 2, 3\}$ оказалась избыточной, а остальные структуры неприемлемыми. Найденной простой структуре $S = \{1, 2\}$ соответству-

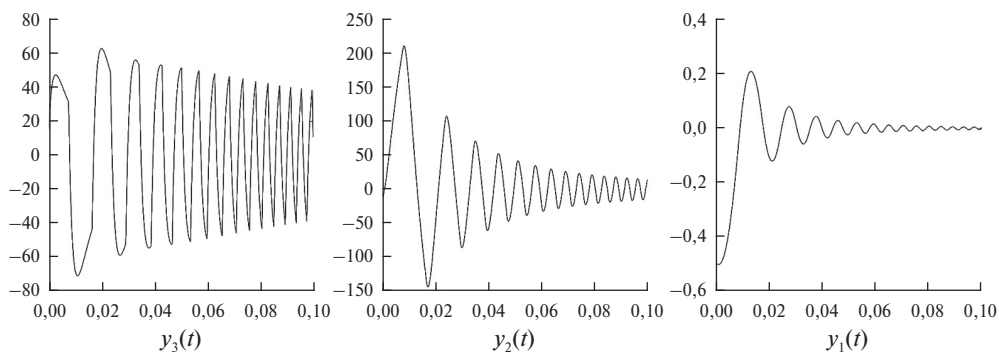


Рис. 4. Графики процессов для $x(0) = x_0^3$, полученные в примере 4.

ет $K = (-61,45; -3,73 \times 10^{-2}; 0)$. Графики переходных процессов в системе (1)–(5), соответствующие найденному вектору K , представлены на рис. 3 для $x(0) = x_0^1$ и на рис. 4 для начального состояния $x(0) = x_0^3$, существенно удаленного от начальных состояний x_0^1 и x_0^2 , указанных в исходных данных рассматриваемого примера.

6. Заключение

В статье поставлена и решена задача синтеза релейных регуляторов в составе автоколебательной системы с линейным объектом управления. Новизна рассматриваемой задачи состоит в совместном выполнении условий существования в системе асимптотически устойчивых автоколебаний с заданными параметрами, ограничений на структуру регулятора и требований приближения поведения системы к желаемому. Для определения желаемого поведения системы задается набор законов изменения во времени выхода системы, выступающих в качестве обучающих примеров. Предложен метод решения сформулированной задачи, являющийся результатом развития алгоритма, изложенного в [17], с целью его распространения на класс релейных автоколебательных систем управления. В математической модели регулятора используется функция идеального реле. Степень влияния различий характеристик реального и идеального реле на поведение синтезируемой системы можно оценить в результате моделирования синтезируемой системы, заменив в (1)–(5) функцию идеального реле более полным и подробным математическим описанием реального реле.

СПИСОК ЛИТЕРАТУРЫ

1. Цыпкин Я.З. Релейные автоматические системы. М.: Наука, 1974.
2. Попов Б.П. Теория нелинейных систем автоматического регулирования и управления. М.: Наука, 1974.
3. Atherton D.P. Nonlinear Control Engineering – Describing Function Analysis and Design. London: Van Nostrand Company Limited, 1975.

4. Фалдин Н.В., Руднев С.А. Синтез релейных систем методом фазового годографа // Изв. Вузов. Приборостроение. 1982. № 7. С. 32–36.
5. Фалдин Н.В. Релейные системы автоматического управления. Математические модели, динамические характеристики и анализ систем автоматического управления / Под ред. К.А. Пупкова, Н.Д. Егунова. М.: Изд-во МГТУ, 2004.
6. Astrom K.J. Oscillations in systems with relay feedback. Adaptive Control, Filtering and Signal Processing / K.J. Astrom, G.C. Goodwin, P.R. Kumar. New York: Springer-Verlag, 1995.
7. Astrom K.J., Hagglund T. Automatic tuning of simple regulators // 9th IFAC World Congress. Budapest, Hungary. 1984. P. 267–272.
8. Boiko I. Discontinuous Control Systems. Boston: Birkhauser, 2009.
9. Мозжечков В.А. Синтез простых релейных регуляторов автоколебательных систем управления // АиТ. 2022. № 9. С. 81–93.
Mozzhechkov V.A. Synthesis of simple relay controllers in self-oscillating control systems // Autom. Remote Control. 2022. V. 83. No. 9. P. 1393–1403.
10. Varigonda S., Georgiou T.T. Dynamics of Relay Relaxation Oscillators // IEEE Trans. Autom. Control. 2001. V. 46. No. 1. P. 65–77.
11. Han T., Wang G., Dong C. A Self-Oscillating Driving Circuit for Low-Q MEMS Vibratory Gyroscopes. // Micromachines. 2023. V. 14. No. 5. P. 1057–1072.
12. Малетин А.Н., Хатанзейская М.А. Исследование фильтрующих свойств маятникового акселерометра, функционирующего в режиме автоколебаний // Авиакосмическое приборостроение. 2021. № 10. С. 3–12.
13. Индейцев Д.А., Лобода О.С., Морозов Н.Ф. Автоколебательный режим нанорезонатора // Физическая мезомеханика. 2016. № 5. С. 23–28.
14. Urasaki S., Yabuno H. Amplitude control for sensorless self-excited oscillation of cantilever based on a piezoelectric device // Nonlinear Dynamics. 2022. V. 108. P. 15–25.
15. Красовский А.А. Синтез автоколебательных систем с применением к ветроэнергетической установке нового класса // Изв. РАН. Теория и системы управления. 1996. № 2. С. 21–30.
16. Aguilar L.T., Boiko I., Fridman L., Iriarte R. Generating Self-excited Oscillations for Underactuated Mechanical Systems via Two-Relay Controller // International Journal of Control. 2009. V. 82. No. 9. P. 1678–1691.
17. Мозжечков В.А. Синтез статической обратной связи в линейных дискретных системах управления на основе обучающих примеров // АиТ. 2023. № 9. С. 68–81.
Mozzhechkov V.A. Static Feedback Design in Linear Discrete-Time Control Systems Based on Training Examples // Autom. Remote Control. 2023. V. 84. No. 9. P. 1065–1074.
18. Мозжечков В.А. Синтез линейных регуляторов с простой структурой // АиТ. 2003. № 1. С. 27–41.
Mozzhechkov V.A. Design of Simple-Structure Linear Controllers // Autom. Remote Control. 2003. V. 64. No. 1. P. 23–36.
19. Мозжечков В.А. Простые структуры в задачах теории управления: формализация и синтез // Известия РАН. Теория и системы управления. 2022. № 3. С. 3–20.

20. *Syrmos V.L., Abdallah C.T., Dorato P., Grigoriadis K.* Static output feedback – a survey // *Automatica*. 1997. V. 33. No. 2. P. 125–137.
21. *Филиппов А.Ф.* Дифференциальные уравнения с разрывной правой частью. М.: Наука, 1985.
22. *Поляк Б.Т., Хлебников М.В., Рапопорт Л.Б.* Математическая теория автоматического управления. М.: Ленанд, 2019.
23. *Икрамов Х.Д.* Численное решение матричных уравнений. М.: Наука, 1984.
24. *Гилл Ф., Мюррей У., Райт М.* Практическая оптимизация. М.: Мир, 1977.
25. *Поляк Б.Т.* Введение в оптимизацию. М.: Наука, 1983.

Статья представлена к публикации членом редколлегии С.А. Красновой.

Поступила в редакцию 13.03.2024

После доработки 10.06.2025

Принята к публикации 20.06.2025

Стохастические системы

© 2025 г. А.А. МАСЛОВ, канд. техн. наук (maslov.aa@mipt.ru),
Г.В. СЕБЕКИН, канд. техн. наук (sebekin.gv@mipt.ru)
(Московский физико-технический институт
(национальный исследовательский университет)),
М.С. СТЕПАНОВ, канд. техн. наук (m.s.stepanov@mtuci.ru),
С.Н. СТЕПАНОВ, д-р техн. наук (s.n.stepanov@mtuci.ru)
(Московский технический университет связи и информатики),
А.О. ШУРКОВ (shchurkov.ao@mipt.ru)
(Московский физико-технический институт
(национальный исследовательский университет))

МОДЕЛЬ ОБСЛУЖИВАНИЯ МУЛЬТИСЕРВИСНОГО ТРАФИКА В УЗЛЕ ДОСТУПА СПУТНИКОВОЙ СЕТИ СВЯЗИ С ДИНАМИЧЕСКИ ИЗМЕНЯЕМОЙ СКОРОСТЬЮ ПРЕДОСТАВЛЕНИЯ СЕРВИСОВ

В мультисервисных спутниковых сетях связи, как правило, существует возможность предоставления того или иного сервиса с различным качеством. При этом используются разные скорости передачи трафика. В лицензионных соглашениях может оговариваться, что конкретный сервис предоставляется с некоторой скоростью в течение основной части времени и допустимо снижение скорости до предельного порога в оставшееся время. При этом операторам сетей необходим математический аппарат, позволяющий оценить выполнение указанных в соглашениях требований, чтобы иметь представление, до каких пределов возможно расширение абонентской емкости сети. В статье разработана математическая модель совместного обслуживания в узлах доступа таких сетей трафика сервисов реального времени и эластичного трафика данных на основе формализации процесса функционирования сети с использованием аппарата многомерных ступенчатых марковских процессов. Приведены примеры решения задач определения требуемого ресурса на этапе планирования сети и оценки возможности расширения абонентской емкости сети при имеющемся ресурсе.

Ключевые слова: космические аппараты, каналный ресурс, мультисервисный трафик, трафик реального времени, эластичный трафик, многомерные ступенчатые марковские процессы.

DOI: 10.31857/S0005231025110044

1. Введение

В мультисервисных спутниковых сетях связи (рис. 1), где роль узлов доступа абонентов к сервисам наземных сетей и Интернета играют центральные земные станции (ЦЗС) совместно с космическими аппаратами (КА), как

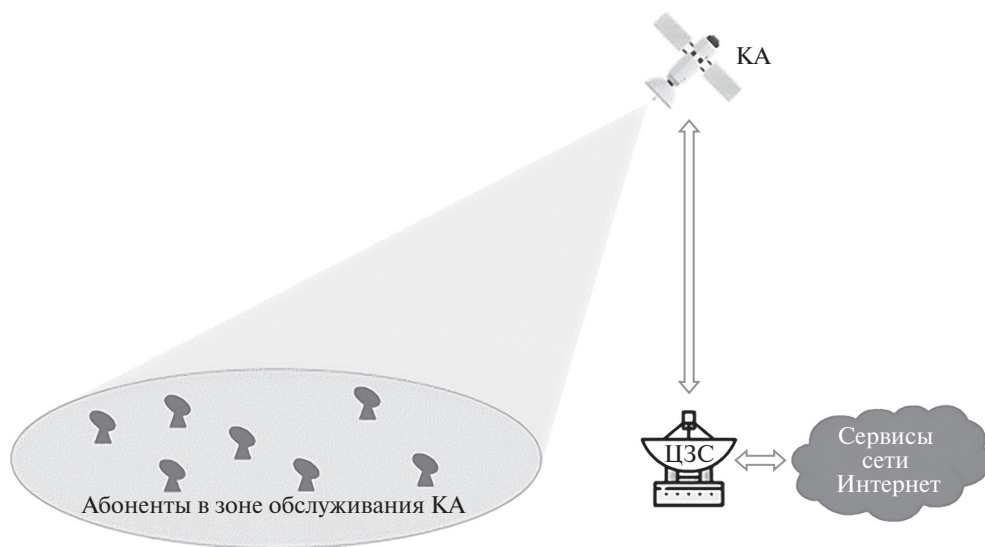


Рис. 1. Упрощенная схема мультисервисной спутниковой сети связи.

правило, существует возможность предоставления того или иного сервиса с различными скоростями передачи трафика. Чем выше скорость, тем лучше качество, например четче видеоизображение и т.п.

Поскольку пропускная способность спутниковых каналов ограничена, то при большом количестве одновременно обслуживаемых абонентов использование максимально высоких скоростей становится невозможным, и для части абонентов требуется снижение скорости. В лицензионных соглашениях (Service Level Agreement, SLA) может оговариваться, что конкретный сервис предоставляется с высокой скоростью в течение основной части времени и допустимо снижение скорости до минимального порога в оставшееся время. В этом случае операторам сетей необходим математический аппарат, позволяющий оценить выполнение указанных в SLA требований, чтобы понимать, до каких пределов возможно расширение абонентской емкости сети при заданных требованиях или когда дальнейшее расширение невозможно и требуется модернизация сети, корректировка нормативов, изменение тарифной политики и т.п.

В статье для решения этой задачи рассматривается узел доступа мультисервисной сети, в которой качество предоставления сервисов (Quality of Service, QoS) оценивается следующими показателями:

- высокие скорости передачи трафика каждого сервиса для обеспечения наилучшего качества;
- доли времени, в течение которых трафики сервисов передаются с высокими скоростями;
- минимально допустимые скорости передачи трафика для каждого сервиса в оставшееся время;

- доли потерянных заявок каждого типа вследствие недостаточности ресурса;
- среднее время доставки файла.

Полагается, что в сети обслуживается K типов трафика сервисов реального времени (РВ) и эластичный трафик сервисов передачи данных. Ресурс для обслуживания заявок на получение сервисов выделяется в виде целого числа квантов скорости. Квант Δv измеряется в бит/с. Для организации работы сети в составе ЦЗС есть система управления связью, которая реализует правила, позволяющие однозначно определить, каким образом назначается и перераспределяется ресурс при поступлении и завершении обслуживания заявок с учетом текущего состояния сети. Считается, что прием заявки на обслуживание осуществляется, если ресурса будет достаточно для обслуживания всех заявок хотя бы на минимальных скоростях. Оставшийся ресурс используется для назначения высоких скоростей обслуживания для части или всех заявок с учетом важности соответствующих типов сервисов и объема указанного оставшегося ресурса. Остатки ресурса, при их наличии, направляются на повышение скорости эластичного трафика. Поэтому для эластичного трафика заданные в SLA высокие и минимальные скорости фактически являются гарантированными и могут быть превышены. Кроме того, если для трафика РВ продолжительность сессии не зависит от скорости, то время доставки файлов с увеличением скорости снижается.

Целью настоящей статьи является разработка математической модели узла доступа мультисервисной спутниковой сети, учитывающей возможности изменения скоростей передачи трафика отдельных сервисов и позволяющей оценить перечисленные выше показатели качества обслуживания трафика каждого типа.

Моделирование мультисервисных сетей, как правило, осуществляется в предположении, что потоки заявок в сети являются пуассоновскими, а продолжительности обслуживания распределены по экспоненциальному закону. Это позволяет строить аналитические модели с использованием аппарата многомерных ступенчатых марковских процессов. Разработке моделей мультисервисных сетей посвящен широкий круг работ. При этом фундаментальные результаты приведены, например, в [1–5]. Получено большое количество результатов для сетей различного назначения [6–23], в том числе, и для мультисервисных спутниковых сетей [16–23], включая сети на базе КА на геостационарной и высокоэллиптической орбитах [16–19, 22], а также на низких круговых орбитах [20, 21]. Хотя фактор изменения скорости передачи в большинстве работ учитывался только для эластичного трафика, но и для трафика реального времени возможности перестройки скорости в некотором диапазоне также рассматривались, например в [15]. Однако при этом учитывались отличные от перечисленных выше показатели качества. Так в [15] целью является анализ средней скорости передачи данных видеоконференций. Поэтому задача моделирования узла доступа сети является актуальной.

Для достижения поставленной цели в разделе 2 дается описание модели, в разделе 3 формируется и решается система уравнений равновесия марковского процесса, описывающего динамику изменения состояний сети, и получаются соотношения для расчета показателей качества. В разделе 4 проводится численный анализ характеристик модели.

2. Описание модели

В рассматриваемой сети передача трафика сервисов реального времени (речевой информации, данных видеоконференцсвязи и др.) в зависимости от типа сервиса может осуществляться на высоких скоростях V_k , $k = 1, \dots, K$, предусмотренных в SLA, и на отдельных интервалах времени при большом количестве активных абонентов на минимально допустимых скоростях V'_k . Передача файлов производится в рамках эластичного трафика и может выполняться в течение основной части времени со скоростью не ниже гарантированно высокой V_e и в остальное время – со скоростью не ниже минимально допустимой V'_e .

Пусть $b_k = \lceil V_k / \Delta v \rceil$, $k = 1, \dots, K$ и $b_e = \lceil V_e / \Delta v \rceil$ – ресурсы, выраженные целым числом используемых квантов скорости и требуемые для передачи трафика РВ и эластичного трафика на высоких скоростях V_k или V_e . Отметим, что ресурс на передачу данных одним абонентом может быть и более b_e . Пусть $b'_k = \lceil V'_k / \Delta v \rceil$, $k = 1, \dots, K$ и $b'_e = \lceil V'_e / \Delta v \rceil$ – ресурсы, соответствующие минимальным скоростям V'_k или V'_e . Здесь также ресурс на передачу данных одним абонентом может быть и более b'_e .

Сервис РВ с номером $k = 1, \dots, K$ получают N_k абонентов. Время обслуживания заявки на получение сервиса является экспоненциально распределенной случайной величиной с параметром μ_k . После завершения обслуживания заявки новая заявка от этого абонента возникает также через случайное время, имеющее экспоненциальное распределение с параметром β_k . Соответствующие параметры для эластичного трафика обозначим через N_e и β_e , а μ_e – параметр экспоненциального распределения времени передачи файла при выделении ресурса b_e .

Введем обозначения для показателей качества. Пусть P_k , $k = 1, \dots, K$, и P_e – доли времени, в течение которого трафик каждого сервиса РВ передается на высокой скорости, а эластичный трафик данных на скорости, не менее гарантированной высокой. Пусть π_k , $k = 1, \dots, K$, и π_e – доли потерянных заявок на предоставление сервисов, которые получили отказ ввиду недостаточности свободного ресурса при их поступлении, W – среднее время доставки файла и v – общий ресурс сети, которым управляет узел доступа.

Динамика изменения состояния сети описывается случайным процессом $r(t) = (i_1(t), \dots, i_K(t), d(t))$, где $i_k(t)$ – число обслуживаемых заявок на предоставление сервиса k и $d(t)$ – число обслуживаемых заявок на передачу данных в момент времени t . Процесс определен на пространстве состояний S , ку-

да входят состояния $s = (i_1, \dots, i_K, d)$ с целочисленными неотрицательными компонентами, каждая из которых не превышает количество абонентов, получающих соответствующий сервис. Пространство S можно записать в виде

$$(1) \quad S = \left\{ (i_1, \dots, i_K, d) : N_k \geq i_k \geq 0, k = 1, \dots, K; N_e \geq d \geq 0; \sum_{k=1}^K i_k b'_k + db'_e \leq v \right\}.$$

В этом пространстве можно выделить подмножества состояний U_k , $k = 1, \dots, K$, и U_e , в которых поступающие заявки на передачу каждого типа трафика реального времени и трафика данных получают отказ из-за недостаточности ресурса на их обслуживание. Указанные подмножества определяются следующим образом:

$$(2) \quad U_k = \left\{ (i_1, \dots, i_K, d) : (i_1, \dots, i_K, d) \in S, \left(\sum_{k=1}^K i_k b'_k + db'_e > v - b'_k \right) \cup (i_k = N_k) \right\},$$

$$(3) \quad U_e = \left\{ (i_1, \dots, i_K, d) : (i_1, \dots, i_K, d) \in S, \left(\sum_{k=1}^K i_k b'_k + db'_e > v - b'_e \right) \cup (d = N_e) \right\}.$$

Для управления работой сети оператору необходимо иметь правило $f(s)$, однозначно определяющее для каждого состояния сети $s \in S$ количество заявок каждого типа $i_k^{(h)}$ из i_k , $k = 1, \dots, K$, и $d^{(h)}$ из d , которое будет обслуживаться на высокой скорости.

Здесь необходимо иметь в виду, что в разных сетях свои перечни предоставляемых сервисов, и важность каждого сервиса определяется оператором с учетом целевых задач, решаемых сетью. Поэтому выбор правила $f(s)$ назначения ресурса имеет качественный характер, и в статье не рассматривается. Будем лишь полагать, что:

- прерывание обслуживания заявок недопустимо;
- все заявки одного типа имеют равную ценность;
- при применении правила ресурс для передачи трафика на высокой скорости назначается для обслуживания заявок в порядке убывания важности их типов, т. е. сначала для заявок наиболее важного типа, потом следующего по важности типа и т. д.

Правило $f(s)$ отрабатывается при изменениях состояний сети и включает три этапа:

- на первом этапе определяется, достаточно ли ресурса для обслуживания всех заявок хотя бы на минимальных скоростях;

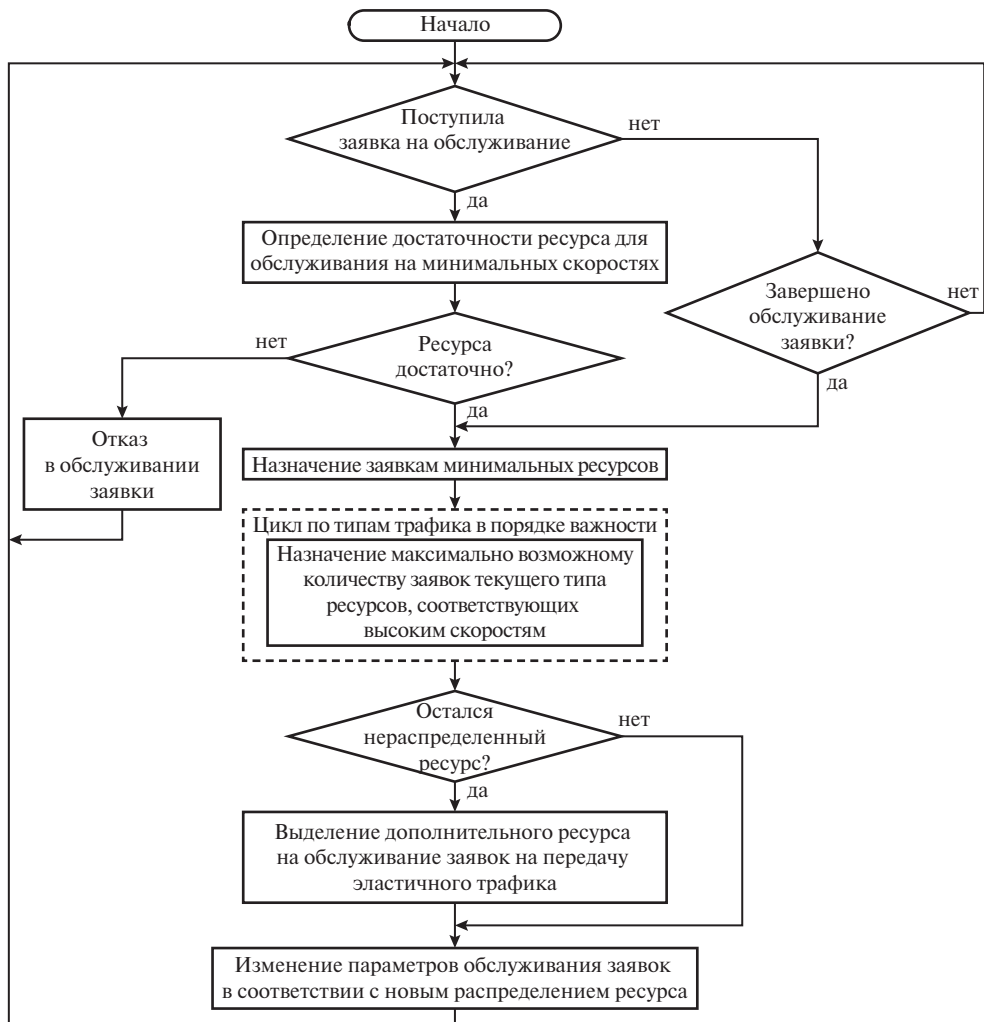


Рис. 2. Укрупненный алгоритм применения правила назначения ресурса.

- если ресурса достаточно, на втором этапе производится назначение высоких скоростей для обслуживания заявок с учетом объема свободного ресурса и важности типов сервисов;
- если имеются остатки свободного ресурса, они направляются на увеличение скорости передачи эластичного трафика.

Укрупненный алгоритм применения правила иллюстрируется рис. 2. Комментируя алгоритм, отметим, что, если после выполнения второго этапа остались заявки, обслуживаемые на минимальных скоростях, ни для одной из них невозможно увеличение скорости до высокой. Выделение дополнительного ресурса для обслуживания заявок, трафик которых уже передается на высокой скорости, не имеет смысла. Поэтому остатки ресурса, если они есть, на третьем этапе направляются на повышение скорости передачи эластично-

го трафика. Отметим также, что в случае поступления заявок на передачу трафика важных типов назначение ресурса для них будет осуществляться раньше, и для заявок менее важных типов при недостаточности ресурса может быть снижена скорость с высокой до минимальной, т.е. будет иметь место перераспределение ресурса, используемого для обслуживания разных типов трафика.

Возможный пример такого правила рассматривается в разделе 4 при проведении численного анализа.

Введем вектор $s^{(h)} = (i_1^{(h)}, \dots, i_K^{(h)}, d^{(h)})$. Тогда $s^{(h)} = f(s)$. Очевидно, после назначения высоких скоростей обслуживания заявок должно выполняться ограничение по ресурсу, более жесткое по сравнению с соответствующим ограничением в выражении (1):

$$(4) \quad v_{\min} = \sum_{k=1}^K \left[(i_k - i_k^{(h)}) b'_k + i_k^{(h)} b_k \right] + (d - d^{(h)}) b'_e + d^{(h)} b_e \leq v.$$

Если в сети обслуживаются заявки на передачу данных, т.е. $d > 0$, и в выражении (4) $v_{\min} < v$, то на третьем этапе отработки правила $f(s)$ остаток ресурса в объеме $v - v_{\min}$ направляется на обслуживание этих заявок. В итоге суммарный ресурс, выделяемый на обслуживание заявок на передачу данных при условии $d > 0$, равен

$$(5) \quad v_e(s, s^{(h)}) = v - v_{\min} + (d - d^{(h)}) b'_e + d^{(h)} b_e,$$

а суммарная интенсивность обслуживания таких заявок равна

$$(6) \quad \mu_{et}(s, s^{(h)}) = \frac{v_e(s, s^{(h)}) \mu_e}{b_e}.$$

При наступлении событий, изменяющих состояние сети, для нового состояния s осуществляется вычисление $s^{(h)} = f(s)$ и производится перераспределение ресурса между заявками, а выражение (6) определяет интенсивность $\mu_{et}(s, s^{(h)})$ выхода из состояния s по событию завершения передачи файла.

3. Система уравнений равновесия и показатели обслуживания заявок

Обозначим через $p(s)$ вероятность нахождения сети в состоянии $s \in S$, а через $P(s)$ ненормированную вероятность указанного состояния. Ненормированные вероятности используются в итерационных методах решения систем уравнений равновесия (СУР). Связь $p(s)$ и $P(s)$ с учетом условия нормировки следующая:

$$(7) \quad p(s) = \frac{P(s)}{\sum_{s \in S} P(s)}.$$

СУР имеет вид:

$$\begin{aligned}
 & P(i_1, \dots, i_K, d) \left[\sum_{k=1}^K (\beta_k (N_k - i_k) I((i_1, \dots, i_K, d) \in S \setminus U_k) + i_k \mu_k) + \right. \\
 & + \beta_e (N_e - d) I((i_1, \dots, i_K, d) \in S \setminus U_e) + \mu_{et} \left(s, s^{(h)} \right) I(d > 0) \left. \right] = \\
 (8) \quad & = \sum_{k=1}^K [P(i_1, \dots, i_k - 1, \dots, i_K, d) \beta_k (N_k - i_k + 1) I(i_k > 0) + \\
 & + P(i_1, \dots, i_k + 1, \dots, i_K, d) (i_k + 1) \mu_k I((i_1, \dots, i_K, d) \in S \setminus U_k)] + \\
 & + I(d > 0) P(i_1, \dots, i_K, d - 1) \beta_e (N_e - d + 1) + \\
 & + P(i_1, \dots, i_K, d + 1) \mu_{et} \left(s_d, s_d^{(h)} \right) I((i_1, \dots, i_K, d) \in S \setminus U_e), \\
 & (i_1, \dots, i_K, d) \in S.
 \end{aligned}$$

Система уравнений (8) является однородной, и единственность решения обеспечивается благодаря добавлению к ней условия нормировки (7). Здесь $s_d = (i_1, \dots, i_K, d + 1)$ и $s_d^{(h)} = f(s_d)$. Интенсивности $\mu_{et}(s, s^{(h)})$ и $\mu_{et}(s_d, s_d^{(h)})$ вычисляются в соответствии с соотношениями (5) и (6). Также используется индикаторная функция I (условие A), равная единице, если условие выполняется, и нулю в противном случае.

Система уравнений (8) может быть решена численно. Хорошо зарекомендовал себя при решении подобных СУР метод Гаусса–Зейделя, описанный в [1] и использованный в ряде работ, например в [16–20]. Указанный метод применяется также для получения численных результатов в разделе 4.

Далее перейдем к получению соотношений для вычисления показателей качества обслуживания трафика, интерпретируя вероятности состояний $p(s)$, $s \in S$, долей времени, в течение которого сеть пребывает в соответствующих состояниях, т.е. за достаточно большой интервал времени T сеть пребывает в состоянии s примерно время $Trp(s)$. Суммарное время обслуживания i_k , $k = 1, \dots, K$, заявок на предоставление k -го сервиса РВ в этом состоянии составляет $i_k Trp(s)$. При этом вектор количеств заявок, обслуживавшихся на высоких скоростях, определяется в виде $s^{(h)} = f(s)$, и суммарное время обслуживания $i_k^{(h)} \leq i_k$ заявок составляет $i_k^{(h)} Trp(s)$. Следовательно, доли времени, в течение которого трафик каждого сервиса передается на высокой скорости, определяются в виде

$$(9) \quad P_k = \frac{\sum_{s \in S} i_k^{(h)} p(s)}{\sum_{s \in S} i_k p(s)}, \quad k = 1, \dots, K.$$

Аналогично для заявок на передачу данных имеем:

$$(10) \quad P_e = \frac{\sum_{s \in S} d^{(h)} p(s)}{\sum_{s \in S} d p(s)}.$$

Поскольку входной трафик в рассматриваемой сети зависит от ее состояния s , то долю заявок на передачу трафика каждого типа, потерянных из-за отсутствия свободного канального ресурса, следует оценивать как отношение интенсивности потерянных заявок соответствующего потока к интенсивности поступивших заявок этого потока [1]. Для заявок на обслуживание трафика РВ k -го типа получаем

$$(11) \quad \pi_k = \frac{\sum_{s \in U_k} (p(s) (N_k - i_k))}{\sum_{s \in S} (p(s) (N_k - i_k))}, \quad k = 1, \dots, K,$$

и для заявок на передачу данных

$$(12) \quad \pi_e = \frac{\sum_{s \in U_e} (p(s) (N_e - d))}{\sum_{s \in S} (p(s) (N_e - d))}.$$

Среднее время доставки файла W может быть определено с использованием формулы Литтла как отношение среднего количества одновременно обслуживаемых в сети заявок на передачу данных y_e к интенсивности λ_e потока таких заявок, принятых на обслуживание, и равно

$$(13) \quad W = \frac{y_e}{\lambda_e},$$

$$(14) \quad y_e = \sum_{s \in S} dp(s),$$

$$(15) \quad \lambda_e = \beta_e \sum_{s \in S \setminus U_e} p(s) (N_e - d).$$

4. Численный анализ характеристик модели

В качестве примера для численного анализа рассмотрим трехсервисную сеть маломощных мобильных абонентских терминалов на базе КА с высокой пропускной способностью [16, 17], обслуживающую два типа трафика РВ и эластичный трафик данных. Характеристики трафика представлены в таблице.

Правило $f(s)$ назначения высоких скоростей исходит из того, что обеспечение высокой скорости при обслуживании заявок на передачу данных важнее, чем для трафика РВ второго типа, а трафик РВ первого типа не нуждается в повышении скорости, поэтому $b_1 = b'_1$. При этом $f(s)$ предусматривает:

- определение, достаточно ли ресурса для обслуживания всех заявок при выделении им ресурсов b'_1 , b'_2 и b'_e ;
- если ресурса достаточно и есть остаток, назначение ресурсов b_e по возможности большему количеству заявок на передачу данных;
- если всем заявкам на передачу данных назначен ресурс b_e и есть остаток, назначение ресурсов b_2 по возможности большему количеству заявок на передачу трафика РВ второго типа;

Параметры системы

Параметр	Значение
Ресурс на обслуживание заявки на сервис РВ первого типа, b_1/b'_1	1/1
Ресурс на обслуживание заявки на сервис РВ второго типа, b_2/b'_2	8/4
Ресурс на обслуживание заявки на передачу данных, b_e/b'_e	8/4
Количество пользователей сервиса РВ первого типа, N_1	150
Количество пользователей сервиса РВ второго типа, N_2	60
Количество пользователей сервисов передачи данных, N_e	120
Интенсивность обслуживания заявок на сервис РВ первого типа, μ_1	0,3 мин ⁻¹
Интенсивность обслуживания заявок на сервис РВ второго типа, μ_2	0,15 мин ⁻¹
Интенсивность обслуживания заявок на ПД при выделении ресурса b_e, μ_e	1,2 мин ⁻¹
Параметр экспоненциального распределения времени до поступления заявок первого типа, β_1	0,06 мин ⁻¹
Параметр экспоненциального распределения времени до поступления заявок второго типа, β_2	0,009 мин ⁻¹
Параметр экспоненциального распределения времени до поступления заявок на ПД, β_e	0,3 мин ⁻¹
Доля времени, в течение которого трафик сервисов РВ передается с заданной высокой скоростью, $P_k, k = 1, 2$, не менее	0,9
Доля времени, в течение которого трафик сервисов передачи данных передается со скоростью, не ниже заданного порога, P_e , не менее	0,9
Предельно допустимые доли потерянных заявок каждого типа вследствие недостаточности ресурса	0,01
Среднее время доставки файла, не более	0,5 мин

- если снова имеется остаток свободного ресурса и есть заявки на передачу данных, назначение указанного свободного ресурса для повышения скорости обслуживания этих заявок.

Использование описанного правила $f(s)$ для $v = 33$ и $s = (2, 2, 2)$ иллюстрируется рис. 3. На первом шаге установлено, что ресурса достаточно для обслуживания всех заявок на минимальной скорости и есть некоторый остаток ресурса, т.е. $2b'_1 + 2b'_2 + 2b'_e = 18$ и остаток ресурса равен 15 квантам. На втором шаге осуществляется проверка, можно ли обеспечить повышение скорости для заявок на передачу данных, и подтверждается такая возможность. Каждой заявке на передачу данных выделяется ресурс b_e . Остаток ресурса уменьшается на восемь и составляет теперь семь квантов. На третьем шаге определено, что повышение скорости возможно только для одной заявки второго типа, и остатка ресурса недостаточно для повышения скорости для второй заявки. Соответственно, ресурс одной заявки второго типа увеличится до восьми квантов, а остаток ресурса уменьшится до трех квантов. Этот оставшийся ресурс отдается на обслуживание заявок на передачу данных на четвертом шаге. В итоге суммарный ресурс заявок на передачу данных

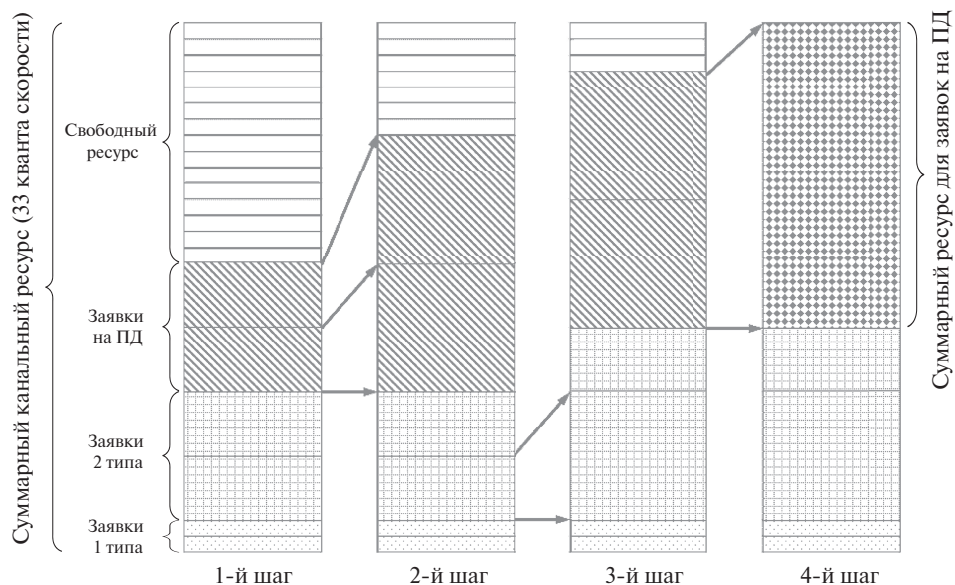


Рис. 3. Пример применения правила назначения ресурса для трехсервисной сети.

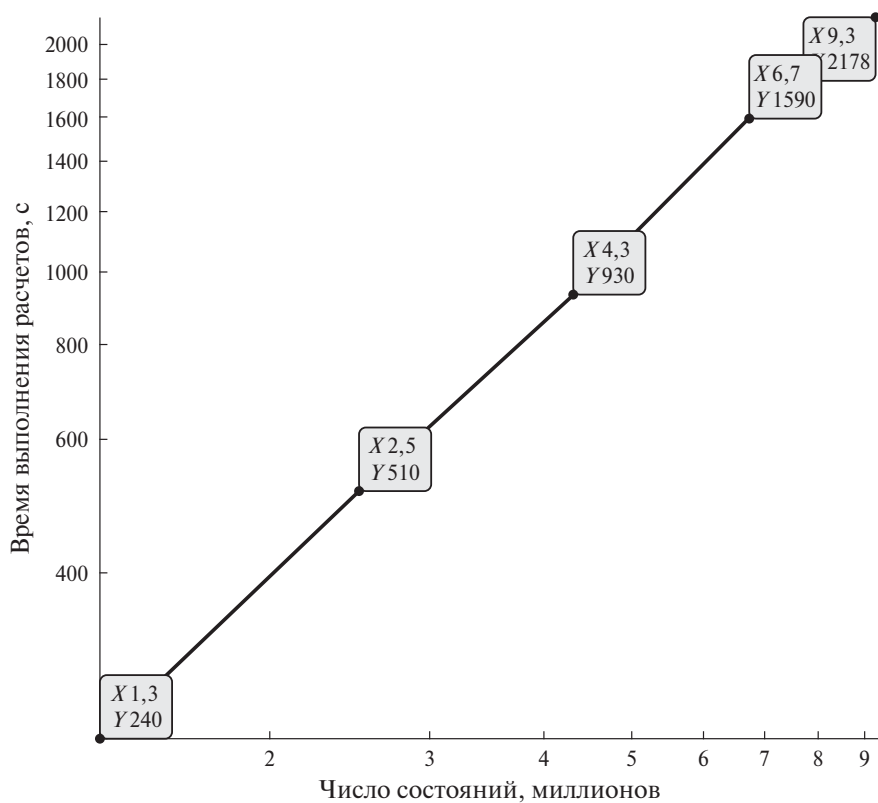


Рис. 4. Зависимость времени выполнения расчетов от числа состояний сети $N_{\text{сост}}$.

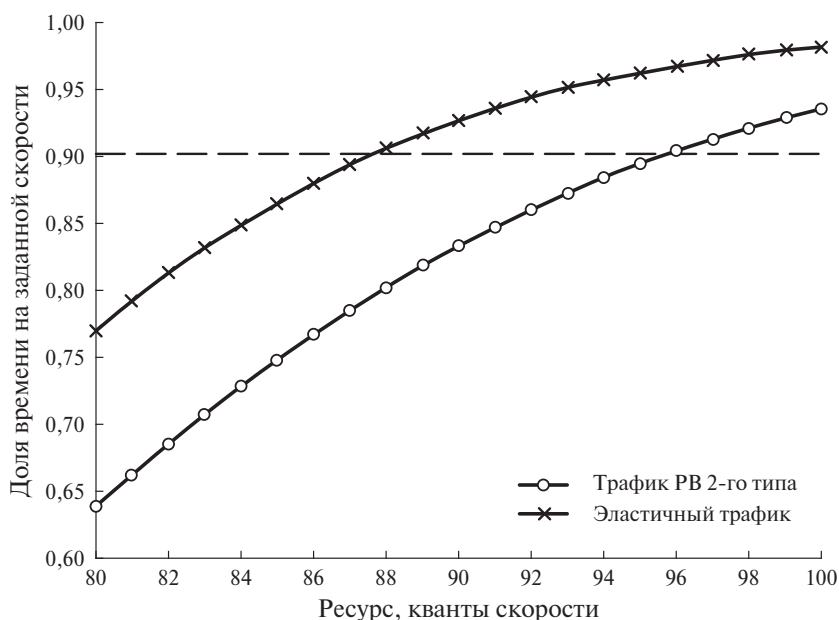


Рис. 5. Зависимости от ресурса v доли времени P_2 , в течение которого трафик РВ второго типа передается с заданной высокой скоростью, и доли времени P_e , в течение которого трафик сервисов передачи данных передается со скоростью, не ниже заданного порога.

составит $2b_e + 3 = 19$ квантов скорости. Для распределения ресурса между этими заявками могут использоваться известные подходы, описанные в [1], например водоналивной алгоритм.

Следует отметить, что предложенное правило является лишь частным примером. В некоторых источниках, например в [24], рассматриваются и другие правила.

Для проведения численного анализа разработан сценарий для среды Matlab, обеспечивающий по набору характеристик сети формирование СУР, ее решение методом Гаусса–Зейделя, а также вычисление показателей качества. Проведена оценка возможности решения СУР большой размерности, иллюстрируемая рис. 4.

Зависимость времени выполнения расчетов от числа состояний $N_{\text{сост}}$ является монотонно возрастающей, практически линейной. При $N_{\text{сост}} \approx 9,3$ миллиона время составило около 36 мин.

Как отмечено в разделе 1, интерес для оператора могут представлять как задача определения требуемого ресурса на этапе планирования сети, так и задача оценки возможности расширения абонентской емкости сети при имеющемся ресурсе.

При решении первой задачи строятся зависимости от ресурса показателей качества обслуживания трафика, иллюстрируемые рис. 5, 6, 7.

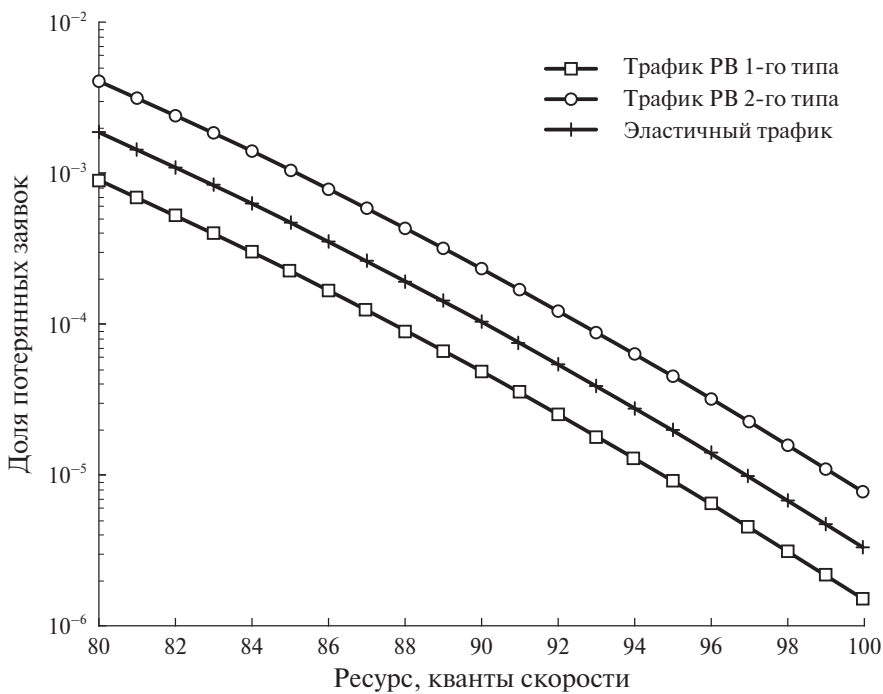


Рис. 6. Зависимости от ресурса v доли потерянных заявок каждого типа вследствие недостаточности ресурса π_k , $k = 1, 2$ и π_e .

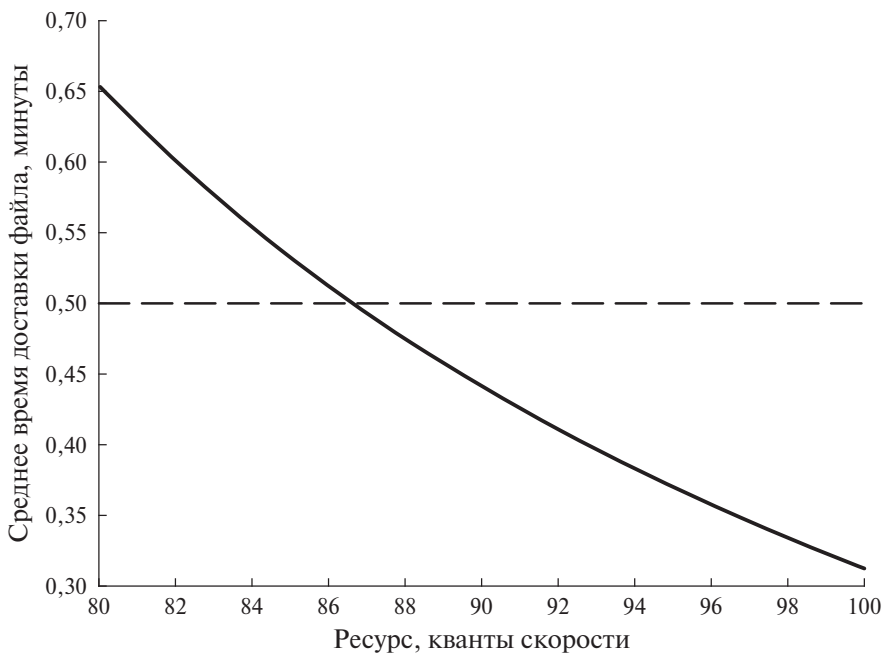


Рис. 7. Зависимости от ресурса v среднего времени доставки файла W .

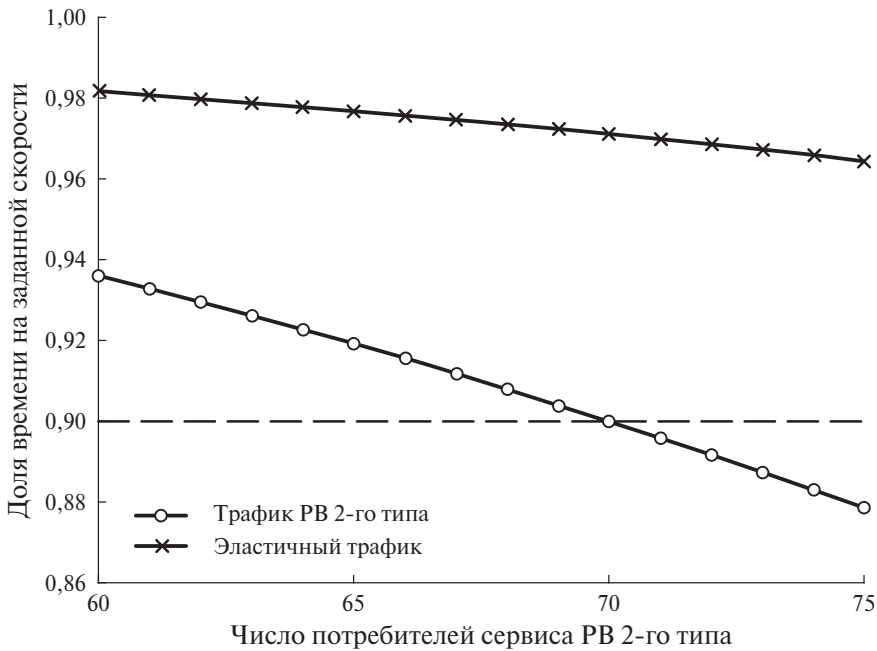


Рис. 8. Зависимости от количества потребителей сервиса второго типа N_2 доли времени P_2 , в течение которого трафик РВ второго типа передается с заданной высокой скоростью, и доли времени P_e , в течение которого трафик сервисов передачи данных передается со скоростью, не ниже заданного порога.

Из рис. 5 видно, что значение P_2 не ниже 0,9 обеспечивается при ресурсе $v \geq 96$, а значение P_e – при ресурсе $v \geq 88$. Для всех значений v в диапазоне от 80 до 100, как следует из рис. 6, ограничение по предельной доле потерянных заявок выполняется с большим запасом. В соответствии в рис. 7 для выполнения ограничения по среднему времени доставки файла необходим ресурс $v \geq 87$. Таким образом, все ограничения выполняются при ресурсе сети $v \geq 96$. Минимально необходимый объем ресурса сети равен 96 квантам скорости.

Оценку возможности расширения сети проведем на примере увеличения количества потребителей сервиса РВ второго типа. Примем, что сеть – с ресурсом $v = 100$ и параметры трафика – согласно таблице с тем отличием, что оценивается возможность увеличения параметра N_2 и он варьируется в диапазоне от 60 до 75. Результаты расчета показателей P_2 и P_e представлены на рис. 8. Видно, что с увеличением N_2 значения показателей P_2 и P_e уменьшаются. При этом величина P_e остается в заданных пределах, а для P_2 ограничение выполняется лишь при $N_2 \leq 70$.

Таким образом, сеть допускает увеличение количества потребителей сервиса РВ второго типа до 70 без изменения объема ресурса. Если количество потребителей превысит 70, то для выполнения заданных ограничений по значениям показателей качества потребуется наращивание ресурса сети.

5. Заключение

Для узла доступа спутниковой мультисервисной сети связи с динамически изменяемой скоростью передачи, где одним из показателей качества является доля времени, в течение которого каждый сервис предоставляется с высокой скоростью, построена модель совместного обслуживания трафика сервисов реального времени и эластичного трафика данных на основе формализации процесса функционирования сети с использованием аппарата многомерных ступенчатых марковских процессов. Получены соотношения для расчета показателей QoS. Программная реализация модели основана на решении систем уравнений равновесия марковских процессов с использованием итерационного метода Гаусса–Зейделя и обеспечивает оценку показателей при большой размерности сети, что подтверждено численными экспериментами при количестве состояний сети до десяти миллионов. Приведены примеры использования модели для решения задачи определения требуемого ресурса на этапе планирования сети и задачи оценки возможности расширения абонентской емкости сети при имеющемся ресурсе. Модель может быть использована при разработке программного обеспечения систем управления связью в спутниковых сетях, а также при эксплуатации сетей для обоснования целесообразности проведения мероприятий по наращиванию их характеристик и модернизации.

СПИСОК ЛИТЕРАТУРЫ

1. *Степанов С.Н.* Теория телетрафика: концепции, модели, приложения. М.: Горячая линия – Телеком, 2015. 868 с.
2. *Вишневецкий В.М., Ефросинин Д.В.* Теория очередей и машинное обучение. М.: ИНФРА-М, 2024. 370 с.
3. *Вишневецкий В.М., Рыков В.В., Козырев Д.В. и др.* Моделирование надёжности привязных высотных беспилотных телекоммуникационных платформ. М.: Техносфера, 2022. 194 с.
4. *Вишневецкий В.М., Дудин А.Н., Клименок В.И.* Стохастические системы с коррелированными потоками. Теория и применение в телекоммуникационных сетях. М.: Рекламно-издательский центр «ТЕХНОСФЕРА», 2018. 564 с.
5. *Вишневецкий В.М., Ляхов А.И., Портной С.Л. и др.* Широкополосные беспроводные сети передачи информации. М.: Техносфера, 2005. 529 с.
6. *Stepanov M.S., Kanishcheva M.G., Malikova E.E., et.al.* The Development and Analysis of a Service Model for the Traffic of a Surveillance System Operator by a Dedicated Resource of an LTE Cell // Proc. of 2022 Systems of Signals Generating and Processing in the Field of on Board Communications. 2022. P. 1–6.
7. *Stepanov M.S., Stepanov S.N., Kroshin F.S.* Effective Algorithm of Estimation the Performance Measures of Group of Servers with Dependence of Call Repetition on the Type of Call Blocking DCCN 2022 / Lecture Notes Computer Science, Springer, Cham. 2022.
8. *Gibadullina E.E., Viskova E.V., Stepanov S.N.* Automated Service Configuration Management in IP/MPLS Networks // 4th International Science and Technology Conference «Modern Network Technologies – 2022», MoNeTec – 2022.

9. *Andrabi U.M., Kanishcheva M., Stepanov S.N.* Observation system resource planning in presence of access control based on volume of resource occupied by traffic flows // T-Comm. 2022. V. 16. No. 8. P. 54–62.
10. *Dawood T., Stepanov M.S., Naoussi C., et.al.* The Mathematical Model of the Internet of Things Traffic Servicing in Case of its Impulse Nature // Proc. of 2023 Systems of Signals Generating and Processing in the Field of on Board Communications. 2023. P. 1–8.
11. *Ndimumahoro F., Stepanov M.S., Muzata A.R., et.al.* Using the Principles of Mobile Systems Modeling for LoRaWAN Characteristics Estimation // Proc. of 2022 Systems of Signals Generating and Processing in the Field of on Board Communications. 2022. P. 1–8.
12. *Степанов С.Н., Степанов М.С.* Приближенный метод оценки характеристик совместного обслуживания трафика реального времени и эластичного трафика данных в мультисервисных узлах доступа // АиТ. 2023. № 11. С. 93–114.
13. *Степанов С.Н., Степанов М.С.* Методы оценки необходимого объема ресурса мультисервисных узлов доступа // АиТ. 2020. № 12. С. 129–152.
14. *Макеева Е.Д., Кочеткова И.А., Шоргин В.С.* Модель для выбора уровней скорости широкополосного трафика eMBB в условиях приоритетной передачи трафика URLLC в сети 5G // Системы и средства информатики. 2023. Т.33. № 4. С. 60–68.
15. *Borodakiy V.Y., Samouylov K.E., Gudkova I.A., et.al.* Analyzing mean bit rate of multicast video conference in LTE network with adaptive radio admission control scheme // J. Math. Sci. 2016. V. 218. No. 3. P. 257–268.
16. *Maslov A.A., Sebekin G.V., Stepanov S.N., et.al.* Model of processes for joint maintenance of real-time multiservice traffic and elastic data traffic in a network of low-power mobile subscriber terminals based on high-throughput satellites // T-COMM. 2024. V. 18. No. 5. P. 41–49.
17. *Маслов А.А., Себекин Г.В., Степанов М.С. и др.* Модель резервирования канального ресурса при обслуживании разнородного трафика в сети маломощных мобильных абонентских терминалов на базе КА с высокой пропускной способностью // Информационные процессы. 2024. Т. 24. № 1. С. 1–15.
18. *Себекин Г.В., Маслов А.А., Щурков А.О.* Моделирование совместного обслуживания мультисервисного трафика реального времени и эластичного трафика данных в сетях на базе космических аппаратов с высокой пропускной способностью // Информационно-измерительные и управляющие системы. 2024. Т. 22. № 2. С. 11–22.
19. *Маков С.В., Маслов А.А., Себекин Г.В.* Оценка эффективности использования ресурса пропускной способности каналов ретрансляции космических аппаратов класса HTS на геостационарной и высокоэллиптической орбите для организации сети передачи трех типов трафика // Наноиндустрия. 2023. Т. 16. № S9-2 (119). С. 613–619.
20. *Маслов А.А., Себекин Г.В., Степанов М.С. и др.* Моделирование процессов обслуживания абонентов в сети передачи данных на базе космических аппаратов на низкой круговой орбите. I // Информационные процессы. 2024. Т. 24. № 4. С. 335–349.
21. *Маслов А.А., Себекин Г.В., Степанов М.С. и др.* Моделирование процессов обслуживания абонентов в сети передачи данных на базе космических аппаратов

на низкой круговой орбите. II // Информационные процессы. 2025. Т. 25. № 2. С. 151–168.

22. Антонович П.И., Маслов А.А., Себекин Г.В. Задача минимизации ресурса спутниковых каналов для организации функционирования сетей связи «Интернета вещей» // Системы синхронизации, формирования и обработки сигналов. 2021. Т. 12. № 6. С. 57–64.
23. Себекин Г.В., Щурков А.О., Маслов А.А. и др. Построение мультисервисной платформы спутниковой связи на основе решений для сетей передачи данных стандарта LTE (3GPP) // Успехи современной радиоэлектроники. 2024. Т. 78. № 2. С. 66–75.
24. Степанов С.Н. Модель обслуживания трафика сервисов реального времени и данных с динамически изменяемой скоростью передачи // АиТ. 2010. № 1. С. 18–33.

Статья представлена к публикации членом редколлегии В.М. Вишневым.

Поступила в редакцию 18.07.2025

После доработки 19.09.2025

Принята к публикации 22.09.2025

Стохастические системы

© 2025 г. А.П. ПОТАПОВ (potapov@ipu.ru),
А.А. ГАЛЯЕВ, д-р техн. наук, чл.-корр. РАН (galaev@ipu.ru)
(Институт проблем управления им. В.А. Трапезникова РАН, Москва)

ТРАЕКТОРНОЕ ПРОТИВОДЕЙСТВИЕ ЛИНЕЙНОМУ НАБЛЮДАТЕЛЮ¹

Рассматривается управляемая динамика трех объектов в n -мерном пространстве – Атакующего (А), Защитника (D) и Цели (Т). Атакующий осуществляет оценку относительного положения Цели при помощи фильтра Калмана–Бьюси, исходя из которой строит коллизийную траекторию. В качестве ответных действий Цель выпускает Защитника, который, воздействуя на приемный канал Атакующего, нарушает процесс построения оценки положения Цели, тем самым препятствуя перехвату. В результате формулируется задача построения оптимальной траектории движения Защитника, увеличивающей время до перехвата Цели Атакующим, и проводится численное моделирование динамики каждого из объектов с оценкой эффективности применения Защитника.

Ключевые слова: управление наблюдениями, фильтр Калмана–Бьюси, ADT-игра, противодействие наведению, траекторная оптимизация.

DOI: 10.31857/S0005231025110058

1. Введение

Групповые задачи управления несколькими игроками привлекают внимание исследователей не только с прикладной точки зрения, связанной с развитием технологий применения беспилотных аппаратов, но и с академической, поскольку аналитические конструкции позволяют проводить анализ разработанных решений и алгоритмов. Прикладные задачи нескольких игроков, как и игровые задачи противодействия двух игроков, делятся на два противоположных по своим целям подкласса: задачи перехвата [1] и задачи уклонения [2, 3] или противодействия перехвату [4, 5]. Когда количество игроков, играющих за одну или другую сторону, увеличивается, то расширяются и их возможности по достижению поставленных ими целей.

Естественное и простое расширение игры двух игроков (Атакующего и Цели) получается добавлением защитника. Возникает так называемая ADT (Атакующий–Защитник–Цель) игра [6]. Однако перед формализацией постановки задачи требуется выбрать политику каждого игрока. Одной из наиболее приближенных к приложениям постановкой является ситуация, в которой

¹ Работа выполнена при частичной поддержке гранта РФФИ № 23-19-00134.

Атакующий принимает решения, опираясь на измерения своих сенсоров. Для этого ему требуется сформировать оценку своего состояния и/или состояния Цели, что производится с использованием теории управления наблюдениями [7], в частности при помощи аппарата калмановской фильтрации [8, 9], который активно развивается уже долгое время [10], в том числе с использованием технологий искусственного интеллекта [11, 12]. В последующем для успешного перехвата Цели Атакующему необходимо выбрать алгоритм наведения, наиболее известным из которых является закон пропорционального наведения [13].

Задачей самого Защитника при этом может являться как перехват атакующего объекта [14] (в таком случае говорят о «жестком» противодействии), так и воздействие на его приемный канал [15, 16] (в таком случае говорят о «мягком» противодействии). При этом в качестве критерия задачи может использоваться как расстояние между целью и атакующим в терминальный момент времени, так и время, требуемое на перехват.

В настоящий момент можно найти большое количество работ, в которых получено аналитическое решение задачи о «жестком» противодействии в той или иной постановке [17]. В то же время для решения задач о «мягком» противодействии чаще всего используются методы Монте-Карло [18] или же методы искусственного интеллекта [19], что не позволяет анализировать полученное решение. При этом теория управления наблюдениями и теория случайных процессов позволяют исследовать сложные динамические системы, учитывая используемые объектами методы принятия решений.

Целью настоящей работы является постановка задачи противодействия простейшему алгоритму наведения и ее аналитическое решение при помощи описанного математического аппарата. Спецификой постановки является тот факт, что в силу использования Защитника уравнение наблюдаемого процесса, исходя из которого Атакующий строит оценку относительного положения Цели, не совпадает с уравнением, которому этот наблюдаемый процесс подчиняется.

Структура работы имеет следующий вид. В разделе 2 формализуется ADT-игра с неполной информацией, в которой каждый из игроков обладает линейной динамикой, а Атакующий получает информацию о внешнем мире из линейного канала наблюдения, зависящего от положения Цели и Защитника в относительной системе координат, связанной с Атакующим. Действуя в коалиции, посредством выбора своего направления движения Цель и Защитник должны отсрочить поимку Цели или же предотвратить ее, поэтому раздел 3 посвящен постановке задачи оптимального управления. В разделах 4 и 5 приводятся решение этой задачи и численное моделирование, показывающее эффективность применения Защитника. В Заключение предлагаются направления развития рассмотренной задачи.

2. Модель АДТ игры с неполной информацией

2.1. Описание исследуемой системы

Рассмотрим линейную систему, состоящую из трех типов игроков – Атакующего (А), Защитника (D) и Цели (Т). Уравнения движения такой системы запишем в виде уравнений Ито

$$(1) \quad \begin{cases} dx_A(t) = Fx_A(t)dt + B_A u(t)dt + \sigma_A dw_A(t), \\ dx_D(t) = Fx_D(t)dt + B_D v(t)dt + \sigma_D dw_D(t), \\ dx_T(t) = Fx_T(t)dt + \sigma_T dw_T(t) \end{cases}$$

с некоторыми начальными условиями в нулевой момент времени.

Будем считать, что Атакующий получает информацию о системе посредством канала наблюдений, который описывается уравнением

$$(2) \quad dz(t) = \beta_e e(t)dt + \beta_\varepsilon \varepsilon(t)dt + \sigma_z dw_z(t),$$

где

$$\begin{aligned} x_A, x_D, x_T \in \mathbb{R}^n, \quad F \in \mathbb{R}^{n \times n}, \quad B_D \in \mathbb{R}^{m \times n}, \quad v \in \mathbb{R}^m, \\ B_A \in \mathbb{R}^{r \times n}, \quad u \in \mathbb{R}^r, \quad \beta_e, \beta_\varepsilon \in \mathbb{R}^{n \times c}, \\ e(t) = x_A(t) - x_T(t), \quad \varepsilon(t) = x_A(t) - x_D(t). \end{aligned}$$

В уравнениях (1), (2) w_A, w_D, w_T, w_z – стандартные n -мерные винеровские процессы, $\sigma_A, \sigma_D, \sigma_T, \sigma_z \in \mathbb{R}^{n \times n}$. Таким образом, координаты всех игроков имеют размерность n , управляющие векторы Атакующего и Защитника имеют размерности r и m соответственно, а измеряемая величина z имеет размерность c .

При таком уравнении канала измерений оказывается удобнее перейти от уравнений (1) к уравнениям динамики относительных координат e, ε . Они имеют вид

$$(3) \quad \begin{cases} d\varepsilon(t) = F\varepsilon(t)dt + B_A u(t)dt - B_D v(t)dt + \sigma_\varepsilon dw_\varepsilon(t), \\ de(t) = Fe(t)dt + B_A u(t)dt + \sigma_e dw_e(t), \end{cases}$$

где

$$\sigma_\varepsilon = \sigma_A + \sigma_D, \quad \sigma_e = \sigma_A + \sigma_T.$$

Пусть атакующий игрок производит оценку относительного положения Цели. При этом считается, что он не знает о существовании защитника, а в качестве алгоритма оптимального оценивания использует фильтр Калмана [9]. Уравнения состояний и наблюдений, исходя из которых Атакующий производит оценку, запишем в виде

$$(4) \quad \begin{cases} de(t) = Fe(t)dt + B_A u(t)dt + \sigma_e dw_e(t), \\ dy(t) = \beta_e e(t)dt + \sigma_y dw_y(t). \end{cases}$$

Предложение 1. Для упрощения дальнейших рассуждений будем полагать, что в системе (4) пара (F, σ_e) управляема, пара (F, β_e) наблюдаема, а также $\det \sigma_q \neq 0$, где q – любой из символов A, D, T, z, y .

Замечание 1. Второе уравнение системы (4) не совпадает с уравнением наблюдений (2). Это объясняется тем фактом, что Атакующий не знает о существовании Защитника при формировании своей оценки. Именно с этим обстоятельством связано накопление ошибки в оценке относительного положения Цели и последующий промах Атакующего.

Для краткости записи зависимость от времени будем опускать.

Будем предполагать, что Атакующий для сближения с целью использует следующий закон управления (такой вид закона управления получается при решении некоторых линейно-квадратичных задач, см., например, [20]):

$$u = \lambda \hat{e}, \quad \lambda \in \mathbb{R}^{n \times r},$$

где $\hat{e}$ – оценка вектора e .

В таком случае оценка случайного вектора e описывается [7] решением уравнений

$$(5) \quad d\hat{e} = F\hat{e} dt + B_A u dt + \gamma \beta_e^T (\sigma_y \sigma_y^T)^{-1} (\beta_e e dt + \beta_e \varepsilon dt + \sigma_z dw_z - \beta_e \hat{e} dt),$$

$$(6) \quad \frac{d\gamma}{dt} = F\gamma + \gamma F^T + \sigma_e \sigma_e^T - \gamma \beta_e^T (\sigma_y \sigma_y^T)^{-1} \beta_e \gamma^T,$$

где γ – среднеквадратические ошибки фильтрации.

Тогда, обозначая $\varphi(\gamma) = \gamma \beta_e^T (\sigma_y \sigma_y^T)^{-1}$ и затем подставляя выражение для управления u в уравнения фильтрации (5), (6), получим уравнение динамики оценки $\hat{e}$ в следующем виде

$$d\hat{e} = \hat{F}(\gamma) \hat{e} dt + \varphi(\gamma) \beta_e \varepsilon dt + \varphi(\gamma) \beta_e e dt + \hat{\sigma}_e(\gamma) dw_z,$$

где

$$\begin{aligned} \hat{F}(\gamma) &= F + B_A \lambda - \varphi(\gamma) \beta_e, \\ \hat{\sigma}_e(\gamma) &= \varphi(\gamma) \sigma_z. \end{aligned}$$

В результате уравнения динамики всех игроков и уравнения фильтра Калмана составляют систему

$$(7) \quad \begin{cases} d\varepsilon = F\varepsilon dt + B_A \lambda \hat{e} dt - B_D v dt + \sigma_\varepsilon dw_\varepsilon, \\ de = Fe dt + B_A \lambda \hat{e} dt + \sigma_e dw_e, \\ d\hat{e} = \hat{F}(\gamma) \hat{e} dt + \varphi(\gamma) \beta_e \varepsilon dt + \varphi(\gamma) \beta_e e dt + \hat{\sigma}_e(\gamma) dw_z, \\ \dot{\gamma} = F\gamma + \gamma F^T + \sigma_e \sigma_e^T - \gamma \beta_e^T (\sigma_y \sigma_y^T)^{-1} \beta_e \gamma^T. \end{cases}$$

2.2. Постановка задачи

Будем считать, что известны математические ожидания переменных системы (7) в начальный момент времени:

$$(8) \quad \mathbb{E} q(0) = \mu_q(0),$$

где q – любой из символов ε , e , $\hat{e}$. Начальные значения ковариационных матриц всех векторов будем, в свою очередь, считать неизвестными.

Замечание 2. Фактически задание начальных условий в таком виде означает, что коалиции Цель–Защитник из каких-либо соображений (например, по собственным измерениям) известно начальное положение Атакующего с точностью до некоторой ошибки измерения. Также предполагаем, что коалиции считается известным алгоритм оценки положения Цели Атакующим. Это приводит к следующей постановке задачи противодействия перехвату.

Задача 1. Для системы, динамика которой описывается уравнениями (7), математическое ожидание случайных процессов в начальный момент времени описывается (8), время движения ограничено t_* , требуется найти управление $v(t)$, ограниченное по модулю $|v| \leq \varkappa$, которое доставляет максимум критерию

$$(9) \quad J[e] = \mathbb{E} (e^T(t_*)e(t_*)).$$

Замечание 3. По своей сути критерий (9) является квадратичным промахом. Обратим также внимание на вид начальных условий (8) – дисперсия случайных векторов считается неизвестной. Как будет показано далее, в силу вида критерия (9) и управляющего воздействия сведения о дисперсии не понадобятся для решения задачи.

3. Приведение задачи к детерминированному виду

Несложно заметить, что критерий (9) можно преобразовать к виду

$$(10) \quad J[e] = \mathbb{E} (e^T(t_*)e(t_*)) = \text{tr} (\Sigma_e(t_*)) + \mu_e^T(t_*)\mu_e(t_*),$$

где $\Sigma_e(t) = \text{Var } e(t)$ – дисперсия случайного вектора $e(t)$.

Очевидно также, что уравнения динамики математических ожиданий имеют вид

$$(11) \quad \begin{cases} \dot{\mu}_\varepsilon = F\mu_\varepsilon + B_A\lambda\hat{\mu}_e - B_D v, \\ \dot{\mu}_e = F\mu_e + B_A\lambda\hat{\mu}_e, \\ \dot{\hat{\mu}}_e = \hat{F}(\gamma)\hat{\mu}_e + \varphi(\gamma)\beta_\varepsilon\mu_\varepsilon + \varphi(\gamma)\beta_e\mu_e, \\ \dot{\gamma} = F\gamma + \gamma F^T + \sigma_e\sigma_e^T - \gamma\beta_e^T(\sigma_y\sigma_y^T)^{-1}\beta_e\gamma^T. \end{cases}$$

Если говорить об уравнениях динамики дисперсий, то важной является следующая лемма, которая приводится без доказательства в силу ее общеизвестности.

Лемма 1. Динамика дисперсии случайного процесса $x(t)$, заданного уравнением Ито вида

$$dx(t) = A(t)x(t)dt + B(t)v(t)dt + \sigma(t)dw(t)$$

с детерминированными матрицами $A(t)$, $B(t)$, $\sigma(t)$ и управлением $v(t)$, описывается уравнением Рикатти

$$\dot{\Sigma}(t) = A(t)\Sigma(t) + \Sigma(t)A^T(t) + \sigma(t)\sigma^T(t).$$

Применим лемму 1 к первым трем уравнениям (7). В таком случае дисперсия случайного вектора $x = (\varepsilon^T \ e^T \ \hat{e}^T)^T$ описывается уравнением

$$(12) \quad \dot{\Sigma} = \Phi(\gamma)\Sigma + \Sigma\Phi^T(\gamma) + \sigma_{\text{total}}\sigma_{\text{total}}^T,$$

где

$$\Phi(\gamma) = \begin{pmatrix} F & 0 & B_A\lambda \\ 0 & F & B_A\lambda \\ \varphi(\gamma)\beta_\varepsilon & \varphi(\gamma)\beta_e & \hat{F}(\gamma) \end{pmatrix}, \quad \sigma_{\text{total}} = \text{diag}(\sigma_\varepsilon, \sigma_e, \hat{\sigma}(\gamma)).$$

Лемма 2. $\text{tr}(\Sigma_e(t_))$ не зависит от выбора управления $v(t)$.*

Доказательство. Несложно заметить, что уравнения системы (12) вместе с последним уравнением системы (7) образуют замкнутую систему, в которой отсутствует управление, что и приводит к утверждению леммы. Таким образом, максимизация критерия (10) по управлению эквивалентна максимизации только его второго слагаемого.

Замечание 4. По существу, в критерии (10) содержится 2 слагаемых, отвечающих за «промах» наблюдателя. Первое из них характеризует влияние на критерий зашумленности канала измерений, а значит, влияние излучаемого Защитником сигнала. Второе же слагаемое характеризует влияние на критерий собственно траекторий Защитника. Лемма 2 фактически утверждает, что отсутствие возможности управлять излучаемым Защитником сигналом приводит к невозможности влияния на первое слагаемое в критерии (10). Это слагаемое имело бы важное значение в случае, если бы можно было управлять излучаемым сигналом, а именно коэффициентом β_ε . Однако в исследуемой постановке это недоступно, а потому максимизация критерия будет происходить только с использованием второго слагаемого, отвечающего за траекторию наблюдателя и защитника.

С другой стороны, такой вид критерия позволяет на каждом шаге или при оценке всей миссии сравнить вклады каждого слагаемого в общий критерий задачи. При этом, рассматривая каждое слагаемое в отдельности, можно установить возможности Защитника по противодействию как при помощи маневрирования, так и при помощи воздействия на измерительные каналы.

Далее обратим внимание, что уравнение Риккати в системе (11) представляет собой независимое дифференциальное уравнение. В таком случае оставшиеся три уравнения системы, по сути, составляют линейную неавтономную систему дифференциальных уравнений. Это, в свою очередь, означает, что вместо исходной динамической системы (7) можно рассматривать систему математических ожиданий, описываемую уравнениями

$$\begin{cases} \dot{\mu}_\varepsilon = F\mu_\varepsilon + B_A\lambda\hat{\mu}_e - B_D v, \\ \dot{\mu}_e = F\mu_e + B_A\lambda\hat{\mu}_e, \\ \dot{\hat{\mu}}_e = \hat{F}(\gamma)\hat{\mu}_e + \varphi(\gamma)\beta_\varepsilon\mu_\varepsilon + \varphi(\gamma)\beta_e\mu_e, \end{cases}$$

которую запишем более кратко в виде

$$(13) \quad \dot{\mu} = \Phi(\gamma)\mu + Bv,$$

где

$$\mu = \begin{pmatrix} \mu_\varepsilon \\ \mu_e \\ \hat{\mu}_e \end{pmatrix}, \quad \Phi(\gamma) = \begin{pmatrix} F & 0 & B_A\lambda \\ 0 & F & B_A\lambda \\ \varphi(\gamma)\beta_\varepsilon & \varphi(\gamma)\beta_e & \hat{F}(\gamma) \end{pmatrix}, \quad B = \begin{pmatrix} -B_D \\ 0 \\ 0 \end{pmatrix},$$

а $\gamma(t)$ – решение дифференциального уравнения

$$\dot{\gamma} = F\gamma + \gamma F^T + \sigma_e \sigma_e^T - \gamma \beta_e^T (\sigma_y \sigma_y^T)^{-1} \beta_e \gamma^T$$

с начальным условием $\gamma(0) = \gamma_0$.

Вместо критерия (9) с учетом леммы 2 для системы (13) запишем новый интегральный критерий

$$J'[e] = -\mu_e^T(t_*)\mu_e(t_*) = -\frac{1}{2} \int_0^{t_*} \langle \dot{\mu}, Q\mu \rangle dt = -\frac{1}{2} \int_0^{t_*} \langle A\mu, Q\mu \rangle dt,$$

где

$$Q = \begin{pmatrix} 0 & 0 & 0 \\ 0 & E_n & 0 \\ 0 & 0 & 0 \end{pmatrix},$$

и сформулируем следующую задачу.

Задача 2. Для системы, динамика которой описывается уравнениями (13), начальные условия задаются равенствами (8), время движения ограничено t_ , требуется найти управление $v(t)$, ограниченное по модулю $|v| \leq \varkappa$, которое доставляет минимум критерию*

$$(14) \quad J'[\mu] = -\frac{1}{2} \int_0^{t_*} \langle A\mu, Q\mu \rangle dt.$$

Предложение 2. Решение задачи 2 является решением задачи 1.

Доказательство. Доказательством, по сути, являются рассуждения, проведенные выше.

4. Решение детерминированной задачи оптимального управления

Для решения задачи 2 воспользуемся принципом максимума Понтрягина [21]. Для этого составим функцию Понтрягина

$$H = \langle \psi, \Phi(\gamma)\mu + Bv \rangle + \frac{1}{2} \langle \Phi(\gamma)\mu, Q\mu \rangle.$$

Условие максимума

$$\langle \psi, Bv \rangle = \langle B^T \psi, v \rangle \longrightarrow \max_{|v| \leq \varkappa}$$

дает почти всюду следующий вид оптимального управления:

$$(15) \quad v^*(t) = -\varkappa |B_D^T \psi_1|^{-1} B_D^T \psi_1,$$

где ψ_i – компоненты вектора $\psi = (\psi_1^T \ \psi_2^T \ \psi_3^T)^T$.

Уравнения сопряженной системы при этом записываются в виде

$$(16) \quad \dot{\psi} = -\Phi^T(\gamma)\psi - \frac{1}{2} (\Phi^T(\gamma)Q + Q\Phi(\gamma)) \mu.$$

Начальные условия для системы (13) задаются в виде (8). При этом для сопряженных переменных из условий трансверсальности известны [21] значения на правом конце траектории в виде $\psi(t_*) = 0$.

Решение такой двухточечной задачи будем искать методом последовательных приближений, описанным в [22], решив перед этим отдельно дифференциальное уравнение Рикатти. Его суть заключается в следующем: в качестве начального приближения выбирается любое допустимое управление $v_1(t)$. Далее, на k -й итерации метода требуется:

- 1) решить задачу Коши для уравнения (13) с начальными условиями (8) и управлением $v_k(t)$. Тем самым получим траекторию $\mu_k(t)$ на $[0, t_*]$;
- 2) решить сопряженную систему (16) с терминальными условиями $\psi(t_*) = 0$ от t_* до 0 при $v(t) = v_k(t)$, $\mu(t) = \mu_k(t)$. Тем самым получим сопряженные переменные $\psi_k(t)$ на $[0, t_*]$;
- 3) согласно равенству (15) определить управление $v_{k+1}(t)$ на $[0, t_*]$.

Алгоритм останавливается, когда величина

$$(17) \quad \text{Err} = \sum_{t=0}^{t_*} |v_{k+1}(t) - v_k(t)|$$

становится меньше заранее выбранного числа $\bar{\delta}$: $\text{Err} < \bar{\delta}$. Сумма в (17) при этом берется по всем точкам вводимой пользователем временной сетки на отрезке $[0, t_*]$.

Замечание 5. Решение такой краевой задачи может занимать значительное время. А именно, выполнение функции, написанной на языке MATLAB для расчета управления в разделе 6, в среднем занимает на процессоре Apple M1 0,0967 с. В случае высоких скоростей объектов такое время решения краевой задачи не позволит успешно выполнить защиту Цели.

5. Исследование линеаризованной системы

Заметим, что для линейных систем известно, что в случае наблюдаемости и управляемости системы ошибка фильтра Калмана γ , а следовательно, и сам коэффициент обратной связи $\varphi(\gamma)$ сходятся при $t \rightarrow \infty$ при любой начальной матрице $\gamma(0)$ [23]. В таком случае, учитывая терминальный вид функционала (10), будет разумно рассматривать систему (13) начиная с некоторого большого момента времени t при постоянной матрице γ .

Согласно предложению 1 выполнены условия теоремы 3.7 [23, с. 237] о сходимости решения дифференциального уравнения Риккати. Следовательно, предельное значение ковариационной матрицы удовлетворяет алгебраическому уравнению Риккати

$$F\mathcal{P} + \mathcal{P}F^T + \sigma_e\sigma_e^T - \mathcal{P}\beta_e^T(\sigma_y\sigma_y^T)^{-1}\beta_e\mathcal{P}^T = 0.$$

Предельное значение коэффициента обратной связи находится согласно уравнению

$$\varphi = \mathcal{P}\beta_e^T(\sigma_y\sigma_y^T)^{-1}.$$

В таком случае при больших t вместо динамической системы (13) с нелинейным уравнением Риккати можно рассматривать линейную систему математических ожиданий, описываемую уравнениями

$$\begin{cases} \dot{\mu}_\varepsilon = F\mu_\varepsilon + B_A\lambda\hat{\mu}_e - B_Dv, \\ \dot{\mu}_e = F\mu_e + B_A\lambda\hat{\mu}_e, \\ \dot{\hat{\mu}}_e = \hat{F}\hat{\mu}_e + \varphi\beta_\varepsilon\mu_\varepsilon + \varphi\beta_e\mu_e, \end{cases}$$

которую запишем более кратко в виде

$$(18) \quad \dot{\mu} = A\mu + Bv,$$

где

$$\mu = \begin{pmatrix} \mu_\varepsilon \\ \mu_e \\ \hat{\mu}_e \end{pmatrix}, \quad A = \begin{pmatrix} F & 0 & B_A\lambda \\ 0 & F & B_A\lambda \\ \varphi\beta_\varepsilon & \varphi\beta_e & \hat{F} \end{pmatrix}, \quad B = \begin{pmatrix} -B_D \\ 0 \\ 0 \end{pmatrix}.$$

Задача оптимального управления для системы (18) формулируется с использованием критерия (14) следующим образом:

Задача 3. Для системы, динамика которой описывается уравнениями (18), начальные условия задаются равенствами (8), время движения ограничено t_ , требуется найти управление $v(t)$, ограниченное по модулю $|v| \leq \varkappa$, которое доставляет минимум критерию*

$$J'[\mu] = -\frac{1}{2} \int_0^{t_*} \langle A\mu, Q\mu \rangle dt.$$

В задаче 3 закон оптимального управления совпадает с (15), однако уравнения сопряженной системы в отличие от уравнений (16) записываются в виде

$$\dot{\psi} = -A^T \psi - \frac{1}{2} (A^T Q + Q A) \mu.$$

6. Численное моделирование

6.1. Параметры моделирования

Рассмотрим систему, которая представлена двойным интегратором. Динамика подобной системы описывается в простых движениях. Будем обозначать символом E_k единичную матрицу размера $k \times k$, а символом 0_k – матрицу $k \times k$, каждый элемент которой равен 0. Далее предположим, что матрицы системы (1) имеют вид

$$(19) \quad F = \begin{pmatrix} 0_2 & E_2 \\ 0_2 & 0_2 \end{pmatrix}, \quad B_A = B_D = \begin{pmatrix} 0_2 \\ E_2 \end{pmatrix}, \quad n = 4, \quad m = r = 2.$$

Другие постоянные зададим следующим образом:

$$(20) \quad \lambda = -6 \begin{pmatrix} E_2 & E_2 \end{pmatrix}, \quad \kappa = 6 \cdot 10^{-3}, \quad \beta_e = \beta_\varepsilon = E_4, \quad t_* = 50,$$

$$(21) \quad \sigma_A = \sigma_D = \sigma_T = \sigma_z = \sigma_y = 10^{-3} \cdot E_4,$$

$$(22) \quad \mathbb{E} x_A(0) = x_A^0 = \begin{pmatrix} 0 \\ -3 \\ 0 \\ 0 \end{pmatrix}, \quad \mathbb{E} x_T(0) = \mathbb{E} x_D(0) = x_D^0 = x_T^0 = \begin{pmatrix} 0 \\ 0 \\ -0,1 \\ 0,2 \end{pmatrix}.$$

Замечание 6. Для системы (4) с матрицами, выбираемыми согласно (19)–(22), выполнено предложение 1.

6.2. Результаты моделирования при успешной работе защитника на всем временном отрезке

Далее будем говорить о математических ожиданиях соответствующих случайных процессов, так как данная характеристика полностью отражает суть задачи.

Рассмотрим случай, когда Защитник оказывает расчетное воздействие на приемный канал, влияя на траекторию движения Атакующего в целом. При этом вплоть до терминального момента времени t_* считаем, что Атакующий не меняет тактику наведения, даже несмотря на то, что оценка в его приемном канале может ухудшаться со временем. Траектории, по которым объекты движутся при рассчитанном согласно равенству (15) и уравнениям (18) управлении в плоскости математических ожиданий их координат, представлены на рис. 1 в случае использования Защитника, а также на рис. 2 без использования Защитника. В первом случае значение критерия составляет

$$J'[\mu_1] = -53,06,$$

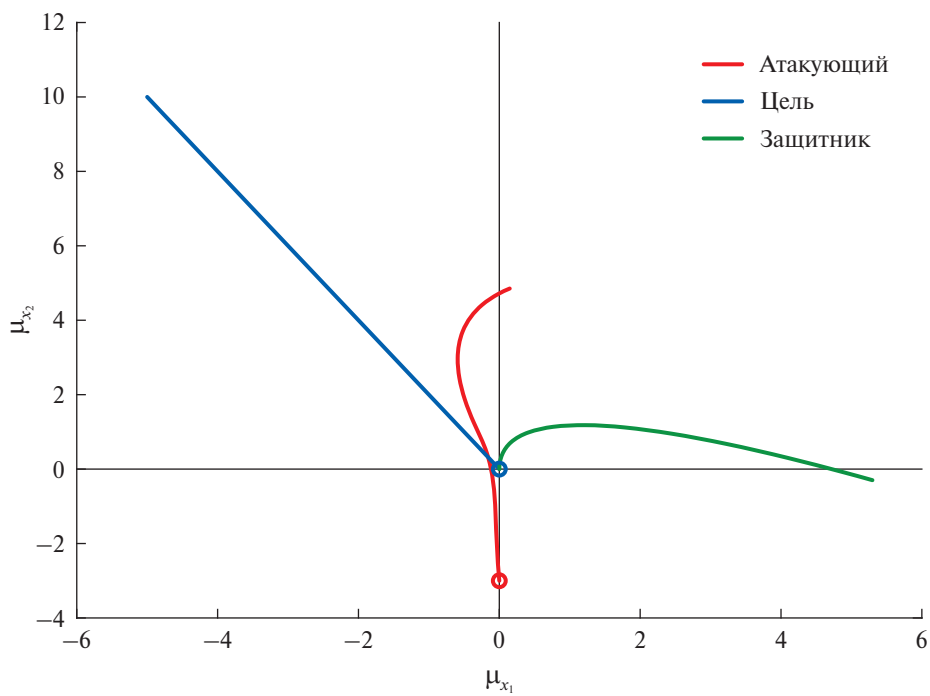


Рис. 1. Траектории движения объектов при использовании Защитника.

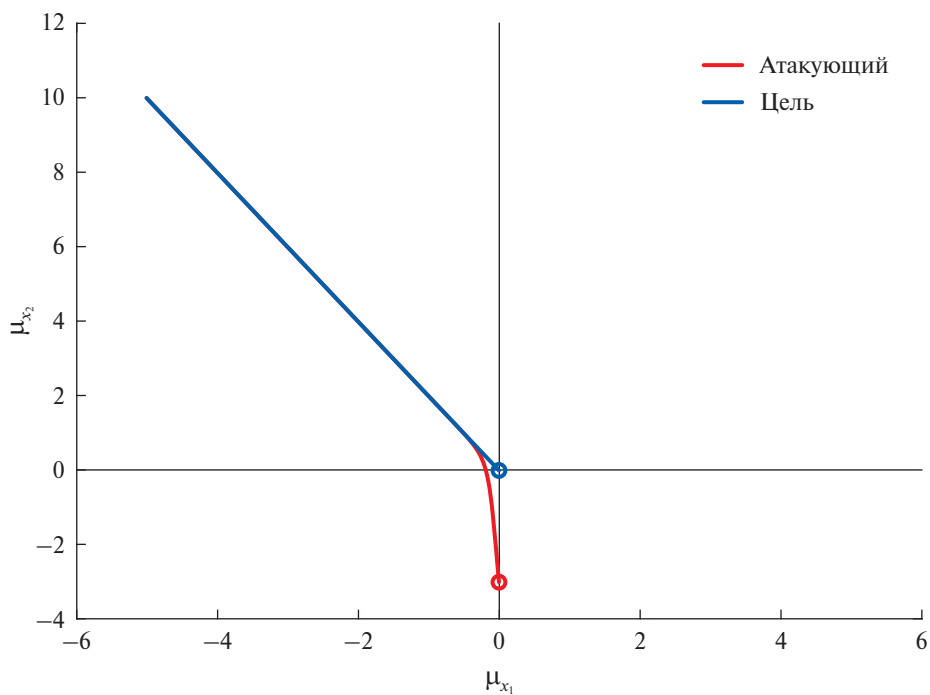


Рис. 2. Траектории движения объектов без использования Защитника.

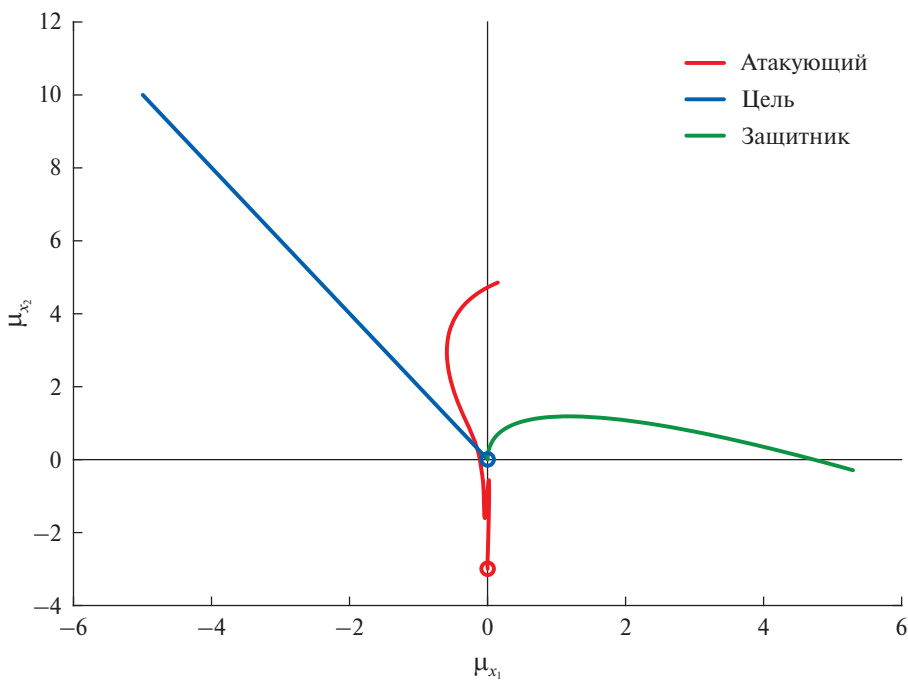


Рис. 3. Траектории движения объектов при использовании Защитника, рассчитанные согласно (13).

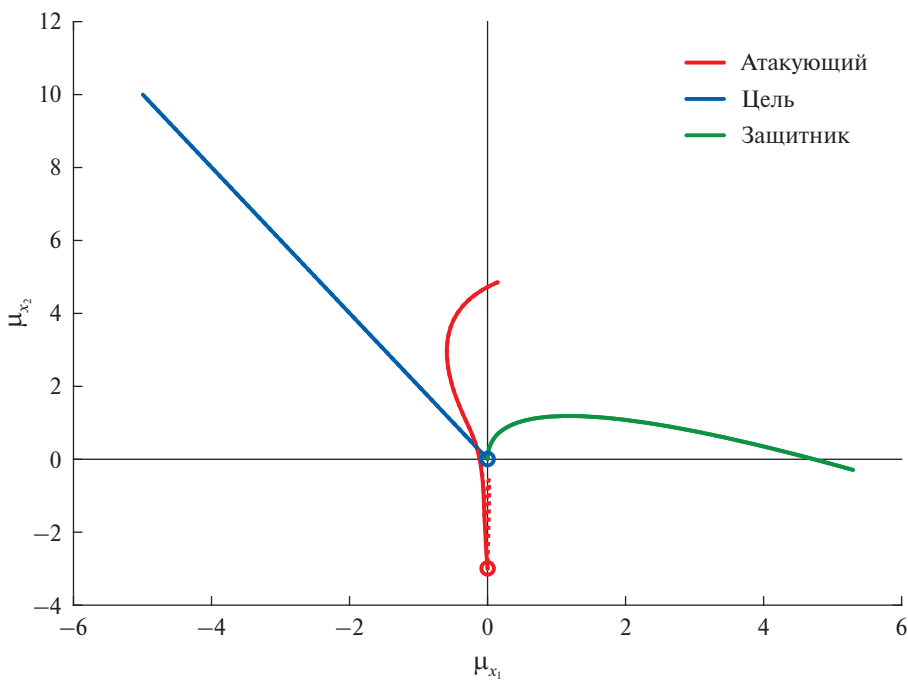


Рис. 4. Траектории движения объектов при использовании Защитника, рассчитанные согласно (18) и (13).

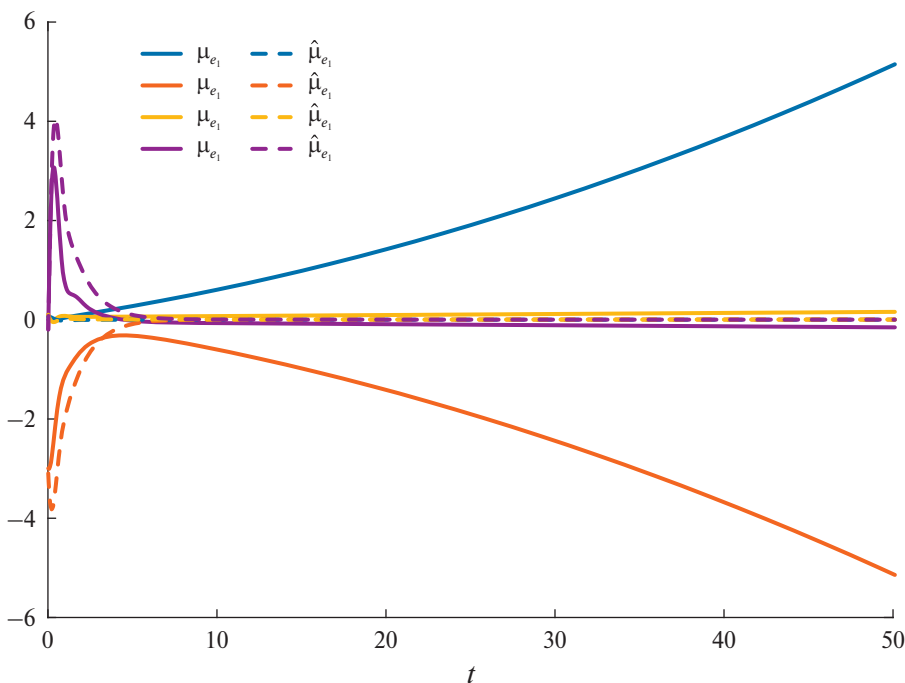


Рис. 5. Изменение компонент векторов μ_e и $\hat{\mu}_e$ с течением времени.

в то время как во втором

$$J'[\mu_2] \approx 0.$$

При этом траектории, по которым объекты движутся при рассчитанном согласно равенству (15) и уравнениям (13) управлению в плоскости математических ожиданий их координат, представлены на рис. 3 (при $\gamma_0 = 4 \cdot 10^{-4} E_4$). Как можно заметить, по прошествии некоторого времени, которое требуется для сходимости решения уравнения Риккати, математические ожидания координат объектов практически не отличаются. Это особенно хорошо видно на рис. 4, на котором сплошными линиями представлены траектории объектов, рассчитанные согласно уравнениям (18), а пунктирными – согласно (13).

Приведем также на рис. 5 графики изменения компонент математических ожиданий вектора относительных координат и его оценки.

6.3. Результаты моделирования при успешной работе защитника на части временного отрезка

Предположим теперь, что в некоторый момент времени $t = \tau$ сигнал, излучаемый Защитником, перестал влиять на выбор направления движения Атакующего. На практике такая ситуация может быть связана с отказом генерирующего сигнал элемента на борту Защитника или коррекцией работы алгоритмов наведения Атакующего. Для описания такого сценария, который назовем сценарием с коррекцией алгоритмов наведения, введем понятие мо-

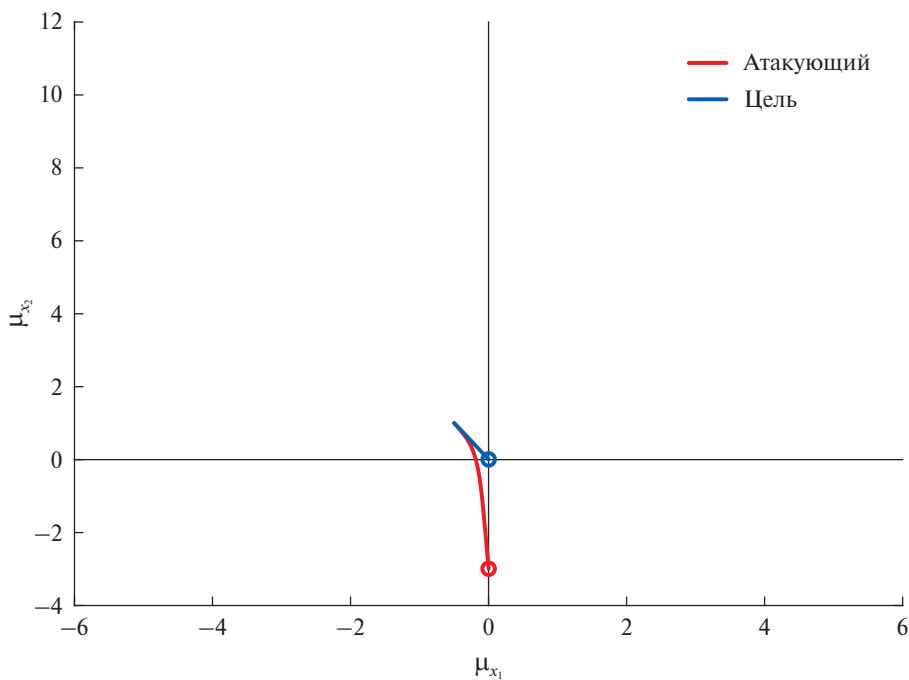


Рис. 6. Траектории движения объектов без использования Защитника до момента перехвата.

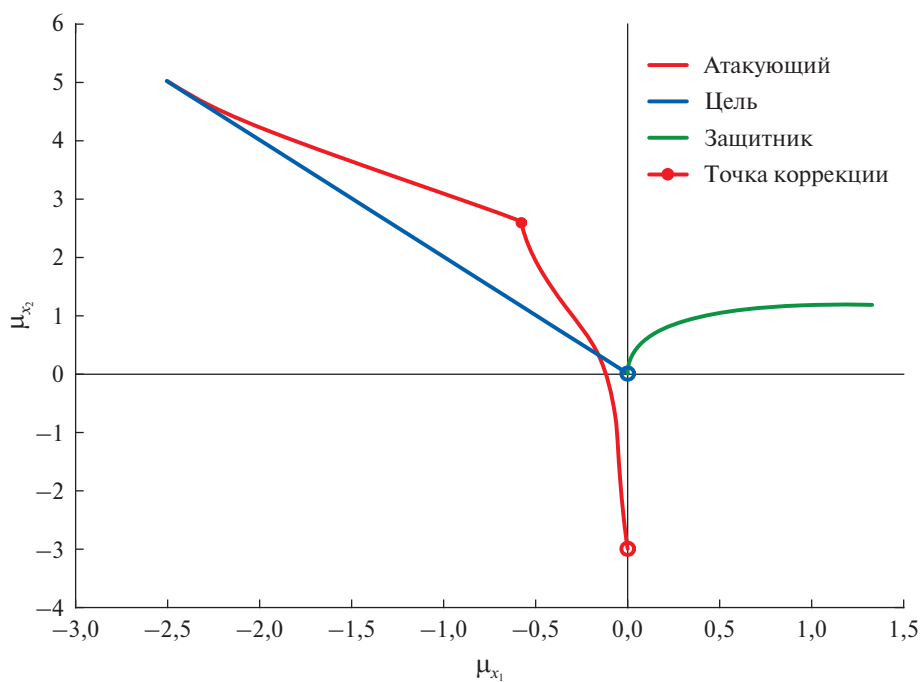


Рис. 7. Траектории движения объектов с использованием Защитника до момента перехвата. Момент коррекции алгоритмов наведения Атакующего обозначен сплошной красной точкой.

мента (времени) перехвата t^* согласно равенству

$$t^* = \min\{t : |\mu_e(t)| < r_0\}.$$

Выберем $r_0 = 10^{-4}$. Траектории движения всех объектов в плоскости математических ожиданий до момента перехвата в случае, если Защитник не используется, показаны на рис. 6. Время перехвата при этом равно $t^* = 5,0$.

Предположим, что при использовании Защитника коррекция алгоритмов наведения произошла при $t = \tau = 20$. Траектории движения объектов в плоскости математических ожиданий до момента перехвата в таком случае представлены на рис. 7. Время перехвата при этом равно $t^* = 25,05$.

Значительное увеличение времени перехвата делает использование даже одного Защитника целесообразным в случае, если алгоритмы наведения Атакующего предполагают коррекцию работы с учетом использования Целью Защитника. На практике это позволяет обеспечить Цель временным запасом для выполнения маневра уклонения, построения последующей стратегии защиты или выпуска других Защитников.

Замечание 7. Предлагаемый метод построения траектории Защитника предполагает также использование в случае, если известными являются не уравнения движения объектов в лабораторной системе отсчета (т.е. уравнения (1)), а уравнения движения объектов в связанной с Атакующим системой координат (т.е. уравнения (3)).

6.4. Сравнение предлагаемой и альтернативных стратегий

Сравним значение критерия на оптимальной траектории, полученной в разделе 6.2, и на траекториях, получаемых при использовании более простых эвристических стратегий:

- 1) движение в сторону, противоположную Цели;
- 2) максимальное сближение с Атакующим.

Первую стратегию описывает закон управления

$$(23) \quad \tilde{v}(t) = -\kappa \left| (x_T^0)_{3,4} \right|^{-1} (x_T^0)_{3,4} = \text{const},$$

а вторую –

$$(24) \quad \bar{v}(t) = -0,02 \left((x_D(t))_{1,2} - (x_A(t))_{1,2} \right).$$

Траектории движения всех объектов в плоскости математических ожиданий при использовании Защитником закона управления (23) представлены на рис. 8. Значение критерия J' при этом составляет

$$J'[\mu_3] = -51,6962 > J'[\mu_1].$$

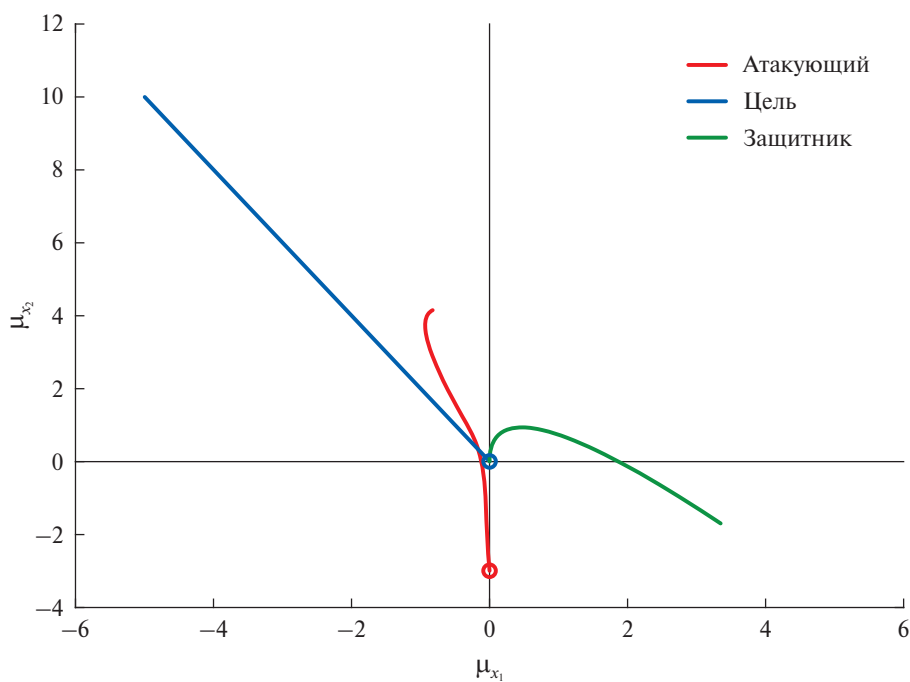


Рис. 8. Траектории движения объектов при использовании Защитником закона управления (23).

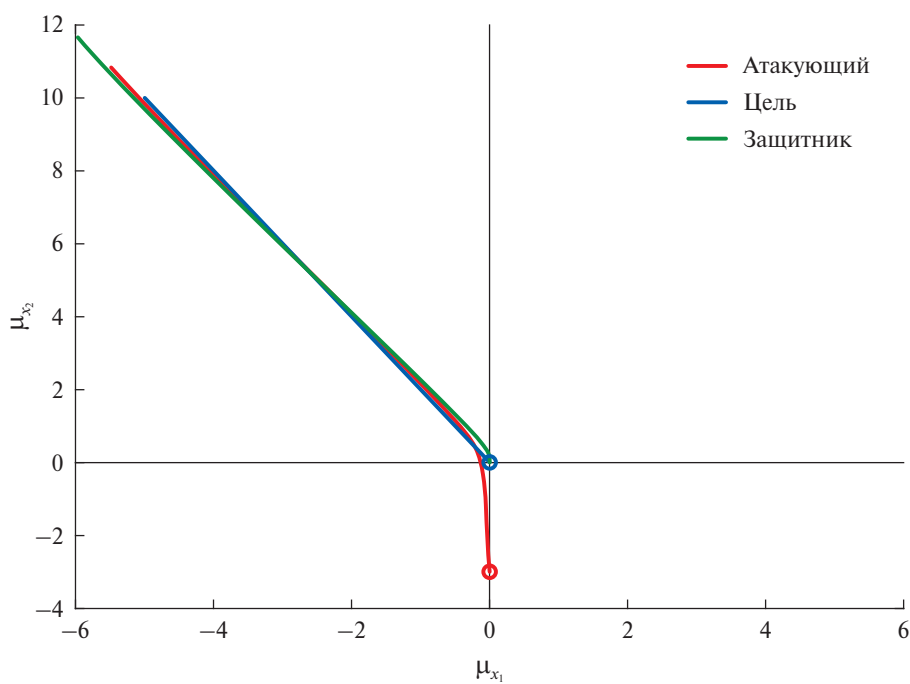


Рис. 9. Траектории движения объектов при использовании Защитником закона управления (24).

Траектории же движения объектов в случае использования Защитником закона управления (24) представлены на рис. 9. Значение критерия J' при этом составляет

$$J'[\mu_4] = -0,93297 > J'[\mu_1].$$

Как несложно заметить, предлагаемое оптимальное решение позволяет получить траекторию, обеспечивающую значительное уменьшение критерия по сравнению с более простыми стратегиями и, как следствие, улучшение тактической обстановки для Цели.

7. Заключение

В работе была показана эффективность использования Защитника в задаче отвлечения Атакующего от перехвата Цели. Предложенная в работе модель применения Защитника при воздействии на приемный канал Атакующего позволяет качественно моделировать и предсказывать изменения в алгоритмах наведения Атакующего, а также формализовывать постановки задач оптимизации траекторного уклонения Цели от перехвата в игре с неполной информацией трех игроков.

Дальнейшая работа будет направлена на формализацию и исследование задач перехвата с более сложными алгоритмами наведения, в том числе учитывающими возможность использования одного или нескольких Защитников с определением способов их применения.

СПИСОК ЛИТЕРАТУРЫ

1. *Galyaev A.A., Lysenko P.V., Rubinovich E.Y.* Optimal Stochastic Control in the Interception Problem of a Randomly Tacking Vehicle // *Mathematics*. 2021. V. 19. No. 9. P. 2386.
2. *Leitmann G.* A differential game of pursuit and evasion // *Int. J. Non-Linear Mech.* 1969. V. 4. No. 1. P. 1–6.
3. *Рубинович Е.Я., Андреев К.В.* Траекторное управление наблюдателем за мобильной целью по угломерной информации // *АиТ*. 2016. № 1. С. 134–162.
4. *Васильев С.Н., Галяев А.А., Залетин В.В. и др.* Совместное использование мехатронных систем для организации эффективного противодействия скоординированному действию торпед противника // *Мехатроника. Автоматизация. Управление*. 2022. Т. 23. № 4. С. 197–208.
5. *Бузиков М.Э., Васильев С.Н., Галяев А.А. и др.* Модель группового противодействия системе самонаведения // *Материалы конференции «Управление в морских системах» (УМС-2022)*. СПб, АО «Концерн «ЦНИИ «Электронприбор», 2022. С. 95–97.
6. *Галяев А.А., Самохин А.С., Самохина М.А.* Моделирование отсрочки поимки цели в АДТ-игре с использованием одного или двух защитников // *Проблемы управления*. 2024. № 2. С. 83–94.
7. *Григорьев Ф.Н., Кузнецов Н.А., Серебровский А.П.* Управление наблюдениями в автоматических системах. М.: Наука, 1986.

8. *Kalman R. E.* A New Approach to Linear Filtering and Prediction Problems // Transactions of the ASME-Journal of Basic Engineering. 1960. P. 35–45.
9. *Лунцер П.Ш., Ширяев А.Н.* Статистика случайных процессов. М.: Наука, 1974.
10. *Julier S.J., Uhlmann J.K.* Unscented Filtering and Nonlinear Estimation // Proceedings of the IEEE. 2004. V. 92. No. 3. P. 401–422.
11. *Song F., Li Y., Cheng W. et al.* An Improved Kalman Filter Based on Long Short-Memory Recurrent Neural Network for Nonlinear Radar Target Tracking // Wireless Communications and Mobile Computing. 2022. P. 10.
12. *Coskun H., Achilles F., DiPietro R. et al.* Long Short-Term Memory Kalman Filters: Recurrent Neural Estimators for Pose Regularization // 2017 IEEE International Conference on Computer Vision (ICCV). 2017. P. 5525–5533.
13. *Girard A.R., Kabamba P.T.* Proportional Navigation: Optimal Homing and Optimal Evasion // SIAM Review. 2015. V. 57. No. 4. P. 611–624.
14. *Потапов А.П., Рубинович Е.Я.* Программная 3D-траектория защитника в ADT-игре при неполной априорной информации у цели // Проблемы управления. 2024. № 5. С. 57–63.
15. *Потапов А.П., Галеев А.А.* Противодействие алгоритму наведения атакующего в игре трех игроков // Мехатроника. Автоматизация. Управление. 2024. Т. 25. № 11. С. 575–584.
16. *Потапов А.П., Галеев А.А.* Модель группового противодействия системе самонаведения // Материалы 27-й Всероссийской научно-практической конференции «Актуальные проблемы защиты и безопасности». СПб, ФГБУ «Российской академии ракетных и артиллерийских наук», 2024. С. 71–73.
17. *Garcia Eloy, Casbeer David W., Pachter Meir.* The Complete Differential Game of Active Target Defense // J. Optim. Theor. Appl. 2021. V. 191. No. 2–3. P. 675–699.
18. *Akhil K.R., Ghose D., Rao S. Koteswara.* Optimizing deployment of multiple decoys to enhance ship survivability // 2008 American Control Conference. 2008.
19. *Chen Y.C., Guo Y.H.* Optimal Combination Strategy for Two Swim-Out Acoustic Decoys to Countermeasure Acoustic Homing Torpedo // 2017 4th International Conference on Information Science and Control Engineering (ICISCE). 2017. P. 1061–1065.
20. *Поляк Б.Т., Хлебников М.В., Рапопорт Л.Б.* Математическая теория автоматического управления. ЛЕНАНД, 2019.
21. *Иоффе А.Д., Тихомиров В.М.* Теория экстремальных задач. М: Наука, 1974.
22. *Чернуосько Ф.Л., Баничук Н.В.* Вариационные задачи механики и управления. Численные методы. М: Наука, 1973.
23. *Kwaternaak H., Sivan R.* Linear optimal control system. Wiley, 1979.

Статья представлена к публикации членом редколлегии Б.М. Миллером.

Поступила в редакцию 17.04.2025

После доработки 11.08.2025

Принята к публикации 18.08.2025

Оптимизация, системный анализ и исследование операций

© 2025 г. О.П. КУЗНЕЦОВ, д-р техн. наук (olpkuz@yandex.ru)
(Институт проблем управления им. В.А. Трапезникова РАН, Москва)

РАСПРЕДЕЛЕННЫЙ АЛГОРИТМ НУМЕРАЦИИ ВЕРШИН ГРАФА, СОВМЕЩЕННЫЙ С ПОСТРОЕНИЕМ ДЕРЕВА ОБХОДА В ШИРИНУ

Предлагается новый распределенный алгоритм нумерации вершин корневого неориентированного графа, который в процессе нумерации строит остовное дерево, являющееся при этом деревом обхода в ширину. Приведены оценки сложности алгоритма.

Ключевые слова: неориентированный граф, распределенный алгоритм, нумерация вершин, остовное дерево, обход в ширину.

DOI: 10.31857/S0005231025110064

1. Введение

Задачи о распределенных вычислениях на графах имеют давнюю историю. Первой такой задачей является задача о синхронизации цепи стрелков, предложенная Дж. Майхиллом в 1957 г. и решенная В. Левенштейном [1] и Ф. Муром [2]. В ней рассматривается цепь одинаковых автоматов, которые после активизации одного из них должны за минимальное время перейти в одно и то же состояние. Начиная с 1980-х гг. стало появляться большое количество распределенных алгоритмов решения различных графовых задач, среди которых отметим задачи нахождения минимального остовного дерева (задача MST) [3–7] и дерева поиска в ширину (задача BFS) [8–10]. Обзоры методов решения этих и многих других задач содержатся в [11–13].

Общая схема распределенных алгоритмов заключается в следующем. В вершинах графа располагаются процессоры, которые обмениваются сообщениями в синхронном или асинхронном режиме. Возможны два варианта начала процесса вычисления: 1) процессоры во всех вершинах начинают работать одновременно; 2) работа алгоритма начинается с активизации одной вершины, называемой корнем графа; остальные вершины активизируются после получения сообщений. Практически все эти алгоритмы предполагают, что вершины либо пронумерованы (см. обзоры [11–13]), либо имеют какие-то уникальные идентификаторы [10]. В [11] описан распределенный алгоритм нумерации вершин, основанный на известном алгоритме Тэрри обхода ребер графа, изложенном в [14].

В настоящей статье предлагается распределенный алгоритм нумерации вершин неориентированного графа, который в процессе нумерации строит остовное дерево, являющееся при этом деревом обхода в ширину (BFS-деревом) и, соответственно, деревом кратчайших путей из корня.

2. Описание алгоритма

Дан связный неориентированный граф с n вершинами и m ребрами, в котором выделена начальная вершина (корень), которую обозначим через v_0 . Такие графы будем называть корневыми; в дальнейшем по умолчанию будут рассматриваться именно такие графы. В каждой вершине находится процессор, который может выполнять один из локальных алгоритмов, в зависимости от состояния вершины. Все вершины могут находиться в одном из четырех состояний, которые будем обозначать цветами: белым, серым, черным и красным. Каждому состоянию (цвету) соответствует свой локальный алгоритм. В дальнейшем будем отождествлять процессор вершины с самой вершиной, т.е. говорить о действиях вершины, подразумевая действия ее процессора.

Каждая вершина имеет упорядоченный список инцидентных ей ребер. Ребра имеют признак, который принимает одно из четырех значений: входящее (верхнее), черное (нижнее), пунктирное (хорда) и красное (об этом ниже).

Вершины могут обмениваться сообщениями двух типов.

Тип 1 имеет вид (i, c) , где i – номер, т.е. $i \in \{1, \dots, n-1\}$, c – цвет, т.е. $c \in \{\text{черный, красный}\}$. Для краткости сообщения этого типа будем называть «черный (красный) i » или «черный (красный) номер». По умолчанию считаем, что номер черный, а сообщение имеет тип 1: выражение «отправить i » будет означать «отправить сообщение (i черный)».

Тип 2 имеет вид (i, j) , где i, j – черные номера; сообщение этого типа возникает, когда при попытке присвоить вершине номер $i+1$ она сообщает, что уже имеет номер j .

В обоих типах сообщений i – это последний присвоенный номер. Его будем называть текущим i или текущим номером.

Вершина белая, если она не пронумерована; ожидает сообщения, чтобы получить свой номер; после этого она становится либо серой (если у нее есть, кроме входящего ребра, другие инцидентные ей ребра), либо красной (если таких ребер у нее нет, т.е. она висятая);

– серая, если она пронумерована, но у нее есть непрономерованные соседи; ожидает сообщения типа 1, чтобы начать нумерацию соседних вершин, или сообщения типа 2, чтобы пометить ребро, по которому это сообщение пришло, как хорду;

– черная, если пронумерована и она, и все ее соседи; при получении текущего номера она передает его на следующий уровень;

– красная, если достижимые из нее вершины следующих уровней либо отсутствуют, либо пронумерованы. В обоих случаях об этом сообщается вверх: по входящему ребру отправляется красное i . Подробнее условия перехода в красный цвет будут описаны ниже.

Идея раскраски вершин в белый, серый и черный цвета заимствована из [15], но модифицирована в связи с распределенностью вычислений: каждому цвету соответствует свой алгоритм; кроме того, добавлен красный цвет, который используется, чтобы сообщить об окончании нумерации на некоторой ветке.

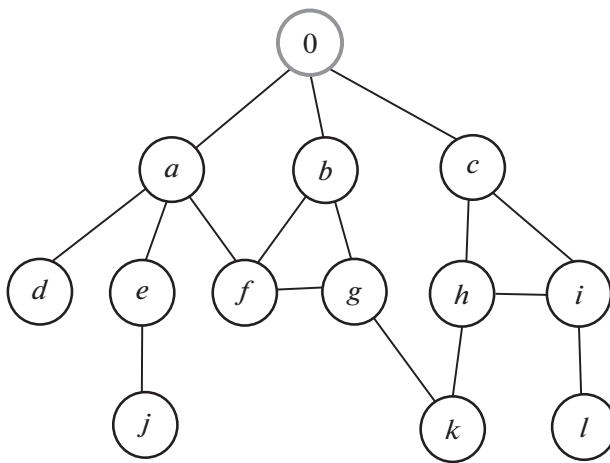


Рис. 1.

Предлагаемый алгоритм нумерации вершин основан на обходе графа в ширину [15] и является последовательным: в каждый момент активна только одна вершина. Передача сообщения фактически является передачей управления: получение сообщения активизирует вершину (запускает ее алгоритм, зависящий от текущего цвета вершины), отправление сообщения прекращает активность. В силу последовательного характера алгоритма в каждый период активности вершина может передать только одно сообщение по одному адресу.

Как будет показано ниже, предлагаемый алгоритм строит остовное дерево, состоящее из входящих ребер; каждая вершина этого дерева находится на минимальном расстоянии от корня. Это позволяет использовать при описании алгоритма два понятия: уровень и ветка. Ветка вершины – это поддереву, корнем которого она является; i -й уровень – это множество вершин, находящихся на расстоянии i от корня исходного графа. Обход в ширину означает, что пока не пронумерованы вершины очередного уровня, вершины следующих уровней не нумеруются; тем самым, если $i < j$, то номер любой вершины i -го уровня будет меньше номера любой вершины j -го уровня.

Напомним, что хорда – это ребро, не содержащееся в остовном дереве.

В качестве примера будем использовать граф на рис. 1 на различных этапах работы алгоритма. Буквы, которыми помечены вершины, введены для удобства описания и чтения. Буквы вершин неизвестны ни ее соседям, ни корню и потому не являются их идентификаторами в обычном смысле; в работе алгоритма они не используются.

Цвета вершин в черно-белой графике будем изображать следующим образом. Белые (ненумерованные) вершины – это вершины, помеченные буквами. Серые вершины имеют обычный (тонкий) контур. Черные вершины заштрихованы. Красные вершины и ребра имеют более толстый контур. Корень изображается наверху, поэтому выражение «отправить наверх» означает «отправить в сторону корня».

Схема общего алгоритма нумерации вершин:

Начальное состояние:

Все вершины, кроме корня, белые; у корня цвета нет; все ребра черные. Множество черных ребер (МЧР) упорядочено, поэтому имеет смысл говорить о первом ребре МЧР.

Начало:

корень присваивает себе номер 0;

формирует очередь из МЧР и посылает 0 по первому ребру очереди.

В дальнейшем он выполняет локальный алгоритм корня, который заключается в управлении нумерацией уровней. Общий процесс происходит следующим образом.

В первом цикле корень, отправляя 0 по первому ребру очереди, инициирует нумерацию вершин 1-го уровня. В конце этого цикла очередь оказывается пустой; это означает, что первый уровень пронумерован и его вершины стали серыми (висячие вершины первого уровня становятся красными). k -й цикл завершается тем, что вершины k -го уровня стали серыми или красными, все вершины предыдущих уровней – черные или красные, и сообщение с текущим номером приходит в вершину 0 по последнему ребру очереди.

В $(k + 1)$ -м цикле корень, получив этот текущий номер, заново формирует очередь из МЧР и отправляет сообщение с черным номером по первому ребру очереди. Это сообщение, пройдя $k - 1$ черных вершин, приходит в серую вершину v k -го уровня, которая, получив это сообщение, нумерует своих соседей, связанных с v черными ребрами (входящими в МЧР); по окончании нумерации отправляет последний номер вверх по входящему ребру (u, v) и становится черной. Если соседняя вершина w уже пронумерована, то ребро (v, w) помечается как хорда. Если после этого МЧР вершины v оказывается пустым, она становится красной и отправляет красный номер вверх в свою черную вершину u , которая делает ребро (u, v) красным. После того, как все нижние ребра черной вершины стали красными, она сама становится красной и отправляет красный номер вверх. Алгоритм заканчивается тогда, когда все ребра, инцидентные корню, стали красными.

Выражение «отправить сообщение» по умолчанию означает конец алгоритма и ожидание очередного сообщения.

Теперь опишем конкретные локальные алгоритмы: алгоритм корня и алгоритмы, соответствующие различным состояниям (цветам) вершин.

Локальный алгоритм корня

1а) Присвоить себе номер 0.

1б) запустить алгоритм нумерации соседей с текущим i ;

1в) сформировать очередь из МЧР;

1г) послать i по первому ребру очереди; ожидать сообщения.

2. Если по текущему ребру очереди пришло черное i (очередной уровень этой ветки пронумерован), то

- 2а) удалить это ребро из очереди;
 - 2б) если оставшаяся очередь непуста, послать i по первому ребру очереди; иначе (очередной уровень графа пронумерован)
 - 2в) сформировать очередь из черных ребер;
 - 2г) послать i по первому ребру очереди.
3. Иначе (по текущему ребру очереди пришло красное число i ; это означает, что на этой ветке ненумерованных вершин нет)
- 3а) сделать это ребро красным и удалить из очереди;
 - 3б) если очередь непуста, послать i по первому ребру;
 - 3в) если очередь пуста и МЧР непусто, то перейти к 2в;
 - 3г) если МЧР пусто (все ребра красные), **конец общего алгоритма.**

На первом уровне хорд нет, поэтому корень получает только сообщения типа 1.

Локальный алгоритм белой вершины

Белая вершина не имеет номера. Все ее ребра черные.

При получении сообщения типа 1 с числом i :

- 1) присвоить себе номер $i + 1$;
 - 2) пометить ребро, по которому пришло i , как входящее;
 - 3) если МЧР непусто, то
 - 3а) отправить черное $i + 1$ вверх по входящему ребру,
 - 3б) стать серой; конец алгоритма;
1. иначе
- 4а) отправить красное $i + 1$ вверх по входящему ребру;
 - 4б) стать красной; конец алгоритма.

Белая вершина, получив свой номер и став серой, «не знает», завершена ли нумерация ее уровня, и потому не нумерует своих соседей, которые находятся на следующем уровне, а лишь сообщает по входящему ребру свой номер «своей» верхней вершине, чтобы та продолжила нумерацию соседей. Но белая вершина может оказаться висячей, т.е. иметь единственного соседа, соединенного с ней входящим ребром. В этом случае (п. 4 алгоритма) она, минуя серый и черный этапы, сразу становится красной.

Локальный алгоритм серой вершины

- 1) если сообщение типа 1 пришло по входящему ребру, то
 - 1а) сформировать из МЧР очередь;
 - 1б) послать i по первому ребру очереди;
 - 1в) если по этому ребру вернулось черное $i + 1$, то
 - 1в1) удалить это ребро из очереди;
 - 1в2) если очередь непуста, перейти к 1б) с $i + 1$;
 - иначе
 - 1в3) отправить $i + 1$ по входящему ребру;

- 1в4) стать черной; конец локального алгоритма;
- 1г) если по этому ребру вернулось красное $i + 1$, то
 - 1г1) сделать это ребро красным;
 - 1г2) удалить его из очереди;
 - 1г3) если очередь не пуста, перейти к 1б) с $i + 1$;
 - иначе
 - 1г4) если МЧР не пусто, перейти к 1в3);
 - иначе (все ребра красные)
 - 1г5) отправить наверх красное i ;
 - 1г6) стать красной;
- 2) иначе (сообщение (x, y) типа 1 пришло по черному ребру)
 - 2а) сделать это ребро пунктирным (это ребро – хорда);
 - 2б) отправить по этому ребру сообщение типа 2 (i, j) , где j – номер этой вершины;
 - 2в) если МЧР не пусто, то конец алгоритма;
 - 2г) иначе
 - 2г1) отправить наверх красное i ;
 - 2г2) стать красной; конец алгоритма.

Пункт 2 соответствует ситуации, когда вершина уже пронумерована и является концом хорды. В результате шага 2б МЧР может стать пустым, и тогда (2г) вершина становится красной. Случай 2г нарушает последовательный характер алгоритма (см. ниже пример 4). Приход номера по черному ребру означает, что идет нумерация на другой ветке q , а обнуление МЧР вызывает параллельную передачу красного номера вверх по данной ветке p , которая может остановиться в любой вершине ветки. Когда красный номер достигнет корня (это произойдет, когда все вершины ветки p станут красными), то корень получит два текущих номера: от ветки p , пронумерованной раньше, и от ветки q , на которой прошла последняя нумерация. При этом: а) входящее ребро ветки p станет красным и будет удалено из МЧР корня; б) текущим останется номер, полученный от ветки q , поскольку он не может быть меньше номера, полученного от ветки p .

Пример 1 (рис. 2). Вершина 1 завершила нумерацию своих нижних соседей (при этом вершины 5 и 6 стали серыми, вершина 4 (висячая) стала красной); отправила последний номер 6 наверх и стала черной. Корень, получив номер 6, удаляет ребро $(0, 1)$ из очереди и отправляет номер 6 серой вершине 2.

Пример 2 (рис. 3). Вершина 2 (серая) запускает нумерацию соседей и пытается пронумеровать серую вершину 6, уже пронумерованную вершиной 1. У вершины 6 срабатывает п. 2 серого алгоритма: она помечает ребро $(2, 6)$ как пунктирное (хорда) и отправляет по этому ребру вершине 2 сообщение $(6, 6)$ типа 2. Вершина 2 также помечает у себя ребро $(2, 6)$ как

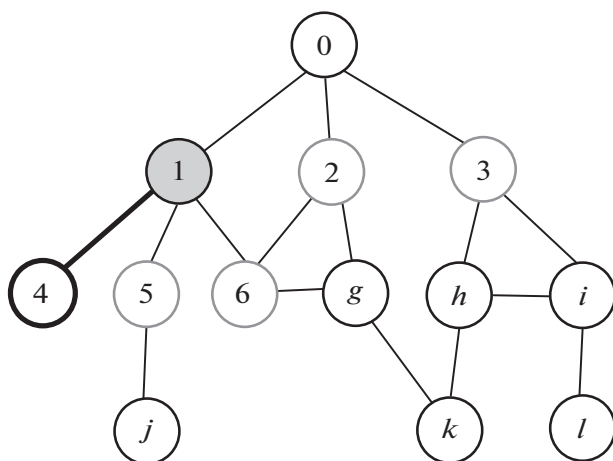


Рис. 2.

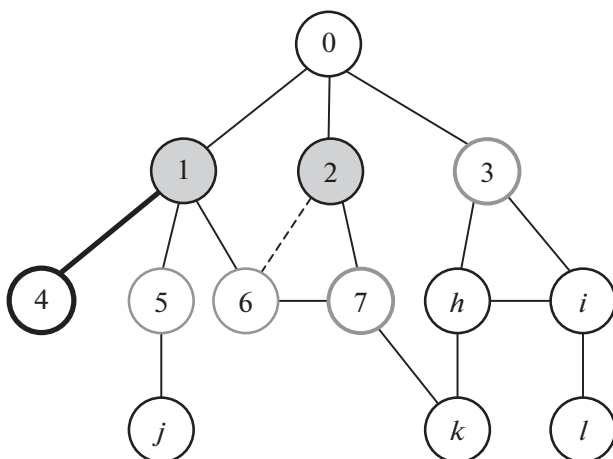


Рис. 3.

пунктирное и отправляет 6 в следующую вершину, которая получает номер 7 и отправляет его по входящему ребру вершине 2. Нумерация соседей вершины 2 на этом закончена (очередь пуста); она сообщает об этом наверх и становится черной.

Локальный алгоритм черной вершины

- 1) если по входящему ребру пришло черное число i , то
 - 1а) сформировать очередь из МЧР;
 - 1б) послать i по первому ребру очереди.
- 2) если снизу пришло черное число i , то
 - 2а) удалить это ребро из очереди;
 - 2б) если очередь не пуста, перейти к 1б) с i ;
 иначе отправить i по входящему ребру; конец локального алгоритма;

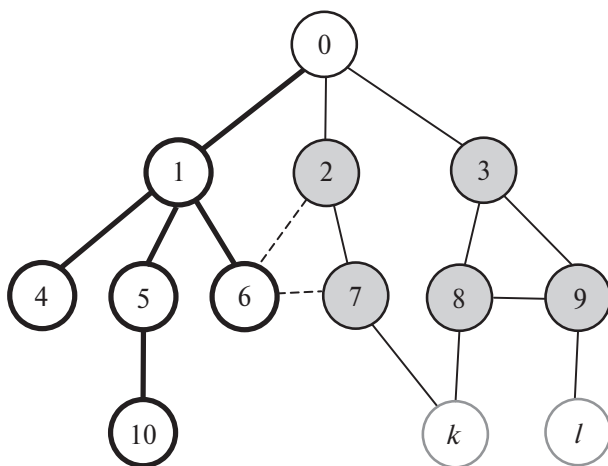


Рис. 4.

3) если снизу пришло красное число i , то

3а) сделать красным это ребро, удалить из очереди;

3б) если очередь не пуста, послать черное i по первому ребру;

3в) если очередь пуста, то

если все ребра красные, стать красной и послать вверх красное i ; конец локального алгоритма;

иначе послать вверх черное i ; конец локального алгоритма.

Пример 3. На рис. 4 представлен этап работы общего алгоритма, на котором завершена нумерация вершин 3-го уровня ветки 2. Рассмотрим, как происходила эта нумерация, приведшая граф к состоянию, изображенному на этом рисунке. Корень послал текущий номер 9 в вершину 1. Вершина 1 и ребро (1, 4) стали красными еще раньше, при нумерации 2-го уровня (см. пример 1), и потому ребро (1, 4) уже не входит в МЧР вершины 1. Вершина 5 пронумеровала вершину 10, которая обнаружила, что она – красная, отправила красный номер 10 вершине 5, которая тоже стала красной и отправила красный номер 10 вершине 1, которая делает ребро (1, 5) красным. Далее вершина 1 отправляет номер 10 в вершину 6, которая пытается пронумеровать вершину 7, получает от нее сообщение типа 1, помечает ребро (6, 7) как хорду, становится красной (п. 2г серого алгоритма) и отправляет красный номер 10 в вершину 1. Вершина 1 делает ребро (1, 6) красным, обнаруживает, что черных ребер у нее нет, становится красной и посылает красный номер 10 в корень. Ребро (0,1) становится красным, т.е. удаляется из МЧР корня, и потому ветка 1 в дальнейших нумерациях не участвует.

Отметим следующее. Уровни концов хорды могут либо совпадать (такие хорды будем называть горизонтальными; пример – хорда (6, 7)), либо отличаться не более чем на 1 (вертикальные хорды; пример – хорда (2, 6)). Действительно, пусть на ребре (x, y) вершина x пронумерована и находится на k -м уровне, а y – еще нет. Очевидно, что (x, y) в этот момент входит в

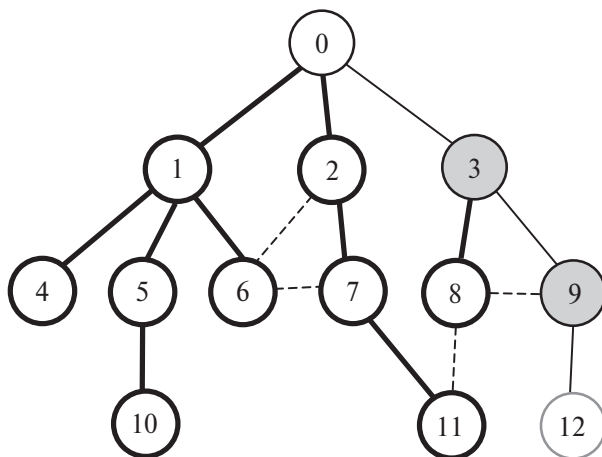


Рис. 5.

МЧР x . Поэтому при нумерации $(k + 1)$ -го уровня вершина x либо пронумерует y , либо обнаружит, что y уже пронумерована; в последнем случае (x, y) будет помечена как хорда. В любом случае, как видно на примере ребра $(6, 7)$, горизонтальная хорда k -го уровня будет обнаружена как хорда только при нумерации $(k + 1)$ -го уровня. Как увидим ниже, это может повлиять на общее время нумерации.

Пример 4 (продолжение примера 3). Получив номер 10 от вершины 1, корень начинает нумерацию ветки 2, которая заканчивается приходом в корень черного номера 11 и не дает новых красных вершин, потому что в алгоритме белой вершины 11 срабатывает п. 3 (МЧР еще содержит ребро $(8, 11)$), и она становится серой, а не красной. После этого начнется нумерация 3-го уровня ветки 3, в ходе которой вершина 8 обнаруживает хорды $(8, 9)$ и $(8, 11)$. При попытке вершины 8 пронумеровать уже пронумерованную (т.е. серую) вершину 11 в этой вершине срабатывает п. 2 серого алгоритма, она становится красной и отправляет вверх свой красный номер. Таким образом, какое-то время будут происходить **два параллельных процесса**: по ветке 2 передается в корень красный номер 11, а по ветке 3 продолжается нумерация. В общем случае возможны два варианта: а) процессы идут по разным веткам; б) процессы идут по разным подветкам одной ветки. В первом случае они сойдутся в корне; во втором случае они придут в некоторую вершину одной ветки, которая либо меняет свой цвет (если ее МЧР станет пустым), либо нет.

На рис. 5 представлен момент окончания нумерации. Вершина 12 только что получила свой номер. В следующий момент она станет красной; и через вершины 9 и 3 красный номер 12 придет в корень, на чем процесс нумерации завершится.

Пример 5. Предположим теперь, что в граф на рис. 5 добавлена горизонтальная хорда $(11, 12)$. Как отмечено в конце примера 3, горизонтальная хорда k -го уровня будет обнаружена как хорда только при нумерации $k + 1$ -го уровня. Поэтому процесс пойдет не так, как описано в примере 4: вер-

пины 11 и 12 не станут красными, после завершения нумерации 3-го уровня в корень придет черный номер 12 и корень запустит нумерацию 4-го уровня, не зная, что его нет. И только на 4-м цикле вершины 11 и 12 станут красными, о чем пойдут параллельно сообщения по веткам (11, 7, 2, 0) и (12, 9, 3, 0).

3. Основные результаты. Оценка сложности алгоритма

Теорема 1. Входящие ребра исходного графа G , помеченные в результате работы алгоритма, образуют дерево, которое является а) остовным, б) деревом кратчайших путей от корня, т.е. деревом обхода в ширину (BFS-деревом).

Рассмотрим граф G^* , образованный входящими ребрами. У каждой вершины только одно входящее ребро. Это следует из того, что ребро помечается как входящее, только когда вершина белая, которая после этого становится серой. Кроме того, если ребро (u, v) – входящее для v , то номер u меньше номера v , потому что v получила свой номер по ребру (u, v) , и вершина u при этом уже была пронумерована. Поэтому на любом пути из v по входящим ребрам номера вершин будут убывать. Отсюда следует, что 1) путь из входящих ребер из любой вершины v не может быть циклом; 2) он может закончиться только в корне, потому что только в корне нет входящего ребра. Таким образом, граф G^* является связным (любые две вершины либо находятся на одном пути к корню, либо связаны двумя путями, ведущими к корню), не содержит циклов и, следовательно, является деревом. Поскольку каждая вершина имеет входящее ребро, дерево G^* содержит все вершины исходного графа, т.е. является остовным.

Докажем теперь, что для любой вершины v ее путь к корню v_0 в графе G^* – кратчайший в графе G . Доказательство проведем индукцией по циклам нумерации.

1-й цикл нумерации – это нумерация корнем своих соседей. Очевидно, что после его окончания все соседи корня стали серыми, они пронумерованы, ребра, соединяющие их с корнем, – входящие, и их расстояние до корня равно 1. Будем считать, что они образуют уровень 1. Ясно, что других вершин с единственным расстоянием до корня нет.

Пусть теперь завершился k -й цикл нумерации, в результате чего появились новые серые вершины, расстояние которых до корня равно k , и других вершин с расстоянием k до корня нет. Эти вершины образуют уровень k . Тогда все вершины, имеющие расстояние $k + 1$ до корня, являются соседями вершин уровня k и, следовательно, в $k + 1$ -м цикле будут ими пронумерованы, станут серыми и образуют $k + 1$ -й уровень. При этом ребра, по которым им был доставлен номер, станут для них входящими и войдут в дерево G^* . Таким образом, для любого k вершина с расстоянием k до корня войдет в k -й уровень дерева G^* , что и доказывает, что граф G^* , образованный входящими ребрами исходного графа G , является деревом кратчайших путей до корня, т.е. деревом обхода в ширину (BFS-деревом).

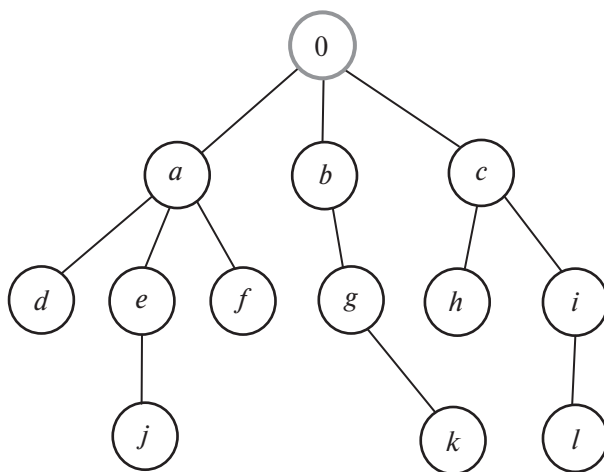


Рис. 6.

Итак, в результате работы алгоритма:

1. Пронумерованы все вершины графа и найдено общее число вершин.
2. Построено остовное дерево графа, состоящее из входящих ребер, найдены хорды графа и тем самым определено цикломатическое число графа.
3. Построенное остовное дерево является деревом обхода в ширину.

Таким образом, предложенный алгоритм решает одновременно две задачи: распределенную нумерацию вершин и распределенное построение дерева обхода в ширину. Поэтому этот алгоритм будем называть NBFS-алгоритмом.

Отметим, что локальный параллелизм, иногда возникающий при работе NBFS-алгоритма (см. пример 4), не требует синхронизации: конечное состояние вершины, в которую придут два параллельных сигнала, не зависит от того, в каком порядке они придут.

Построенное остовное дерево не является единственным и зависит от порядка ребер в очередях, формируемых локальными алгоритмами.

Дерево обхода в ширину для нашего примера приведено на рис. 6. Если на втором цикле очередь в корне имела бы вид bac , то вершина f получила бы номер 4, а ребро (b, f) не было бы хордой.

Сложность. Под сложностью NBFS-алгоритма в дальнейшем будем понимать его коммуникационную сложность, т.е. число сообщений, порождаемых NBFS-алгоритмом в ходе его работы. Введем следующие обозначения:

$C_{NBFS}(n)$ – сложность NBFS-алгоритма для корневых графов с n вершинами;

$C_{NBFS}(G)$ – сложность NBFS-алгоритма для конкретного графа G ;

$e(G)$ – эксцентриситет корня графа G , т.е. максимальное из расстояний от v_0 до остальных вершин;

$\gamma(G)$ – цикломатическое число (и соответственно число хорд) графа G ,

G_{BFS} – остовное дерево графа G . В дальнейшем будем писать просто e и γ .

Любой алгоритм нумерации вершин корневого графа должен совершить последовательный обход всех вершин начиная с корня. Распределенный алгоритм при этом должен обойти все ребра, поскольку при отсутствии исходной информации о глобальных свойствах графа нет гарантии, что посещены все вершины, пока не посещены все ребра. Известный алгоритм Тэрри [14] обхода всех ребер основан на том, что если расщепить каждое ребро на два «полуребра», то получим эйлеров граф, в котором возможен обход каждого полуребра по одному разу; это равносильно тому, что исходное ребро можно пройти два раза в разных направлениях, причем обход обязательно оканчивается в корне. Поэтому для распределенного алгоритма Тэрри [11] коммуникационная сложность равна числу посещений каждого ребра, т.е. $2m$. Очевидно, что в силу последовательного характера любого алгоритма нумерации оценку $O(m)$ понизить нельзя.

Заметим, что число уровней G и G_{BFS} равно e ; следовательно, число циклов нумерации также равно e . Очевидно, что из всех корневых неориентированных графов с n вершинами наибольший эксцентриситет имеет цепь Ch_n , т.е. связный граф с n вершинами, в котором имеются две концевые вершины со степенями 1, причем одна из этих вершин является корнем; все остальные вершины имеют степень 2. При этом

$$(1) \quad e(Ch_n) = n - 1.$$

Оценим сложность цепи Ch_n .

Лемма 1.

$$(2) \quad C_{NBFS}(Ch_n) = (n - 1)n.$$

Цепь Ch_n согласно (1) имеет $n - 1$ уровень; поэтому для ее нумерации потребуется $n - 1$ цикл. В k -м цикле ($k = 1, \dots, n - 1$) участвуют $k + 1$ вершин (включая корень). При этом сообщения, которые они порождают, можно представить в виде цепочки длины $k + 1$ вида $12 \dots 21$, где i -й элемент цепочки ($i \leq k + 1$) соответствует числу сообщений, порождаемых вершиной $(i - 1)$ -го уровня. Действительно, корень и последняя вершина цепочки порождают по одному сообщению, а остальные вершины – по два (одно нумерует следующую вершину, другое сообщает текущий номер в сторону корня). Общее число сообщений в одной такой цепочке равно $2k$; соответственно общее число сообщений, порождаемых процессом нумерации цепи длины n , равно сумме сообщений по всем $n - 1$ циклам, т.е. удвоенной сумме арифметической прогрессии $2 \sum_{k=1}^{n-1} k = (n - 1)n$.

Поскольку k -й цикл нумерации вовлекает в обмен сообщениями все черные ребра первых k уровней, ясно, что величина эксцентриситета существенно влияет на величину C_{NBFS} . Более точное утверждение выглядит так.

Лемма 2. Среди всех корневых деревьев с n вершинами наибольшей NBFS-сложностью обладает дерево с наибольшим эксцентриситетом, т.е. цепь Ch_n .

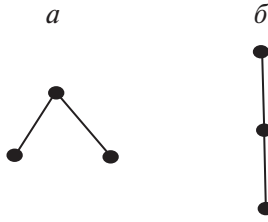


Рис. 7.

Это утверждение докажем по индукции.

Минимальное n , для которого это утверждение имеет смысл, равно 3. При $n = 3$ возможны два дерева, изображенные на рис. 7.

Дерево 7,а имеет один уровень и потому требует одного цикла нумерации, которому соответствует цепочка 211 (два сообщения от корня и по одному сообщению от остальных вершин), т.е. 4 сообщения. Дерево 7,б – это цепь, имеющая два уровня и два цикла нумерации: первому циклу соответствует цепочка 11, второму циклу – цепочка 121; итого $2 + 4 = 6$ сообщений. Таким образом, C_{NBFS} дерева 7,б больше, чем C_{NBFS} дерева 7,а, и, следовательно, для $n = 3$ лемма верна.

Предположим теперь, что утверждение леммы верно для всех деревьев с числом вершин, не превосходящим k , т.е. сложность $C_{NBFS}(Ch_k) = (k - 1)k$ максимальна для всех деревьев с k вершинами. Докажем, что оно верно для $k + 1$: сложность цепи $C_{NBFS}(Ch_{k+1}) = (k + 1)k$ максимальна для всех деревьев с $k + 1$ вершинами.

Любое дерево с $k + 1$ вершинами можно получить из некоторого дерева G_k с k вершинами путем присоединения к G_k ребра (x, y) таким образом, что вершина x отождествляется с некоторой вершиной G_k , а вершина y становится висячей вершиной нового дерева G_{k+1} . Возможны два варианта.

а) G_k – цепь. Если ребро (x, y) присоединяется к концу цепи, то получим цепь Ch_{k+1} со сложностью $(k + 1)k$. Если же ребро (x, y) присоединяется к любой другой вершине G_k уровня $i < k$, то уровень полученного дерева G_{k+1} не изменится, поэтому $(k + 1)$ -го цикла нумерации не будет, а при нумерации $(i + 1)$ -го уровня появятся два новых сообщения – от вершины y и обратно, причем при этом y станет красной и в дальнейшем сообщений генерировать не будет. Таким образом, $C_{NBFS}(G_{k+1}) = C_{NBFS}(Ch_k) + 2 < C_{NBFS}(Ch_{k+1})$ и, следовательно, в классе всех деревьев с $k + 1$ вершинами, полученных присоединением ребра к цепи Ch_k максимальную сложность имеет цепь Ch_{k+1} .

б) G_k – не цепь. Тогда $e(G_k) \leq k - 1$ и по предположению индукции

$$(3) \quad C_{NBFS}(G_k) \leq (k - 1)k.$$

Пусть ребро (x, y) присоединяется к висячей вершине v с уровнем $i \leq k - 1$. Любая висячая вершина является концом некоторой цепи, присоединенной другим концом к вершине со степенью > 2 , причем ее длина не пре-

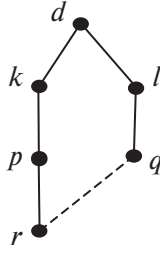


Рис. 8.

восходит $k - 1$. При нумерации дерева G_k вершина v (и вся цепь) становится красной на i -м цикле нумерации и в дальнейших циклах не участвует. В новом дереве G_{k+1} эта цепь удлиняется на одно ребро и ее концом вместо v становится вершина y с уровнем $i + 1 \leq k$. Поэтому эта цепь будет участвовать в $(i + 1)$ -м цикле нумерации, в котором к сложности $C_{NBFS}(G_k)$ добавится не более чем $2(i + 1) \leq 2k$ сообщений. Поэтому $C_{NBFS}(G_{k+1}) \leq C_{NBFS}(G_k) + 2k$, откуда, используя (3), получим $C_{NBFS}(G_{k+1}) \leq (k - 1)k + 2k = (k + 1)k = C_{NBFS}(G_{k+1})$. Таким образом, и в случае б) сложность цепи оказывается максимальной, что и доказывает лемму.

Рассмотрим теперь NBFS-сложность произвольного корневого неориентированного графа G с n вершинами и m ребрами. Цикломатическое число (оно же – число хорд) такого графа вычисляется по известной формуле $\gamma = m - n + 1$.

Теорема 2.

$$(4) \quad C_{NBFS}(n) = O(n^2 - n).$$

Сложность $C_{NBFS}(G)$ конкретного графа G можно представить в виде суммы двух слагаемых:

$$(5) \quad C_{NBFS}(G) = C_{NBFS}(G_{BFS}) + C_\gamma(G).$$

Первое слагаемое – это сложность остовного дерева графа G . Она определяется сообщениями, которые связаны с собственно нумерацией (присвоение номера и обратное сообщение о текущем номере) и проходят только по входящим ребрам, т.е. по будущему остовному дереву. Согласно леммам 1 и 2 $C_{NBFS}(G_{BFS}) \leq n^2 - n$.

Второе слагаемое – это число сообщений, порождаемых при обнаружении хорд. В общем случае с обнаружением хорды (x, y) связано два сообщения: вершина x пытается пронумеровать вершину y ; вершина y сообщает, что она уже пронумерована. После этого ребро (x, y) удаляется из МЧР вершин x и y ; последующие сообщения по нему не проходят. Однако возможна более сложная ситуация, которая описана в примере 4. Рассмотрим ее более подробно.

На рис. 8 представлен фрагмент некоторого графа, в котором вершины p, q находятся на i -м уровне и идет $(i + 1)$ -й цикл нумерации. По ветке $dkpr$ он

уже прошел, вершина r получила свой номер и отправила его обратно в вершину p ; при этом в МЧР вершины r ребро (r, q) сохраняется. При активизации вершины q она пытается пронумеровать вершину r и получает от нее сообщение, что она уже пронумерована. После этого обе вершины удаляют ребро (r, q) из своих МЧР. Если при этом оказывается, что МЧР вершины r пусто (это возможно, только если в будущем остовном дереве эта вершина – висячая), то она становится красной и посылает еще одно сообщение в вершину p . Вершина p , получив это сообщение, тоже может стать красной, что породит сообщение от нее в вершину k , и т.д. Таким образом, одновременно с процессом нумерации в ветке dlq возникнет дополнительная цепочка сообщений в ветке $dkpr$, которая и внесет свой вклад в слагаемое $C_\gamma(G)$. Все эти сообщения делают некоторые ребра красными, что приводит к их удалению из МЧР; поэтому по каждому ребру может пройти только одно такое сообщение. Следовательно, их общее число не превосходит m (общего числа ребер графа), для которого, как известно, справедливо $m \leq \frac{n^2-n}{2}$.

Другой вариант этой ситуации описан в примере 5, где горизонтальная хорда соединяет две висячие вершины дерева G_{BFS} . В этом случае возникает дополнительный цикл нумерации. Сообщения этого цикла, как максимум, пройдут не более двух раз по всем входящим ребрам графа. В этом случае вклад хорды в $C_\gamma(G)$ не превысит $2m \leq n^2 - n$.

Итак, оба слагаемых не превосходят величину $n^2 - n$, что и доказывает теорему.

Для разных классов графов соотношение между двумя слагаемыми в (5) может сильно отличаться. В частности, в разреженных графах (графах с малым числом хорд) основной вклад в C_{NBFS} вносит сложность остовного дерева, которая, в свою очередь, определяется величиной эксцентриситета корня. Поэтому даже в пределах одного графа сложность будет заметно меняться в зависимости от выбора корня; естественно ожидать, что минимальной она будет, когда корень – центр графа, а максимальной, когда корень – конец одного из диаметров. Верхняя оценка, согласно леммам 1, 2, достигается, когда граф является цепью с корнем на одном из ее концов.

Другой крайний случай – полный граф K_n , у которого при любом выборе корня $e = 1$, число ребер $m = \frac{n^2-n}{2}$; из этих ребер $n - 1$ – входящие, остальные – хорды: $\gamma = \frac{n^2-n}{2} - n + 1$, причем из $e = 1$ следует, что все хорды горизонтальны. Из этих формул видно, что основной вклад в $C_{NBFS}(K_n)$ вносит величина $C_\gamma(G)$, причем достижение верхней оценки обеспечивается числом хорд.

Работа NBFS-алгоритма на графе K_n состоит из двух циклов нумерации. В первом цикле все вершины получают свои номера, но благодаря горизонтальности хорд ни одна вершина не становится красной; поэтому корень об окончании нумерации не знает и запускает второй цикл, в котором обнаруживаются все хорды.

4. Заключение

Задача нумерации имеет универсальное значение для распределенных алгоритмов. Почти все распределенные алгоритмы на графах предполагают, что нумерация вершин уже проведена. Кроме того, наличие единой для всех нумерации позволяет корню после ее проведения запросить от каждой вершины списки инцидентных ей ребер и тем самым восстановить у себя полное описание графа.

Об эффективности NBFS-алгоритма можно судить, сравнив его с алгоритмом Тэрри. Для «плотных» графов, близких по числу ребер к K_n , оценки сложности обоих алгоритмов по порядку совпадают, и, следовательно, NBFS-алгоритм оказывается предпочтительным, поскольку он одновременно строит BFS-дерево. Напротив, для разреженных графов, близких к деревьям, NBFS-алгоритм применять не стоит: во-первых, для таких графов он слишком сильно проигрывает алгоритму Тэрри по сложности (ясно, что цепь можно пронумеровать за один проход), а во-вторых, главное достоинство NBFS-алгоритма (одновременное с нумерацией построение BFS-дерева) здесь не работает, поскольку всякое дерево само по себе уже является BFS-деревом.

Среди возможных применений NBFS-алгоритма, помимо обычных для распределенных алгоритмов приложений к сетям коммуникаций, следует указать групповую робототехнику. Одна из возможных ситуаций: группа роботов отправляется на задание по патрулированию местности, имея полный граф связей между роботами группы. К концу задания некоторые связи нарушены, и лидеру необходимо восстановить граф оставшихся связей. Процент нарушенных связей невелик, граф оставшихся связей близок к полному, и потому NBFS-алгоритм будет эффективен.

Еще один алгоритм, который строит BFS-дерево и нумерует вершины, описан в [16]. Он сначала строит BFS-дерево и уже на этом дереве проводит нумерацию вершин, используя обход дерева, похожий на алгоритм Тэрри, с той существенной разницей, что для дерева, как известно, $m = n - 1$. Вместе с использованием параллелизма (в алгоритме [16] вершины отправляют сигналы одновременно всем своим соседям) это дает оценку временной сложности $O(n)$, что лучше сложности NBFS-алгоритма. Однако за это улучшение приходится платить введением синхронизации, которая в реальных приложениях может либо потребовать дополнительного оборудования, либо просто затруднительна в реализации – например, в приведенном выше примере с групповой робототехникой. Кроме того, нумерация, описанная в [16], выглядит нерегулярно (величина номера вершины ничего не говорит о близости вершины к корню, т.е. числа – это всего лишь уникальные идентификаторы), тогда как в NBFS-алгоритме номера обладают указанным в начале раздела 2 свойством: если $i < j$, то номер любой вершины i -го уровня будет меньше номера любой вершины j -го уровня. Это свойство использовано при доказательстве теоремы 1 и может оказаться полезным в приложениях.

СПИСОК ЛИТЕРАТУРЫ

1. *Левенштейн В.И.* Об одном методе решения задачи синхронизации цепи автоматов за минимальное время // Пробл. передачи информ. 1965. Т. 1. Вып. 4. С. 20–32.
2. *Moore F.R., Langdon G.G.* A generalized firing squad problem // Information and Control. 1968. V. 12. P. 212–220.
3. *Gallager R.G., Humblet P.A., Spira P.M.* A distributed algorithm for minimum-weight spanning trees // ACM Transactions on Programming Languages and Systems. 1983. V. 5. No. 1. P. 66–77.
4. *Peleg D., Rubinfeld V.* A near-tight lower bound on the time complexity of distributed MST construction // Proc. 40 IEEE Symp. on Found. of Comp. Sci. (FOCS). 1999. P. 253–261.
5. *Вялый М.Н., Хузиев И.М.* Распределенная коммуникационная сложность построения остовного дерева // Пробл. передачи информ. 2015. Т. 51. № 1. С. 54–71.
6. *Вялый М.Н., Хузиев И.М.* Быстрые протоколы выбора лидера и построения остовного дерева в распределенной сети // Пробл. передачи информ. 2017. Т. 53. № 2. С. 91–111.
7. *Dinitz M., Halldórsson M., Izumi T., Newport C.* Distributed minimum degree spanning trees // Proceedings 2019 ACM Symposium Principles Distributed Computing, 2019. P. 511–520.
8. *Awerbuch B., Gallagher R.G.* Distributed BFS algorithms // Proc. 26 IEEE Symposium Foundations Computer Science (FOCS). 1985. P. 250–255.
9. *Park J., Tokura N., Masuzawa T., Hagihara K.* An efficient distributed algorithm for constructing a breadth-first search tree // Systems and Computers in Japan. 1989. V. 20. P. 15–30.
10. *Makki S.A.M.* Efficient distributed breadth-first algorithm // Computer Communications. 1996. V. 19. No. 8. P. 628–636.
11. *Бурдонов И., Косачев А.* Общий подход к решению задач на графах коллективным автоматов // Труды ИСП РАН. 2017. Т. 29. Вып. 2. С. 27–76.
12. *Бурдонов И., Косачев А., Сортов А.* Распределённые алгоритмы на корневых неориентированных графах // Труды ИСП РАН, 2017. Т. 29. Вып. 5. С. 283–310.
13. *Ghaffari M.* Distributed Graph Algorithms. 2022.
<https://people.csail.mit.edu/ghaffari/DA22/Notes/DGA.pdf>
14. *Оре О.* Теория графов. М.: Наука. 1968.
15. *Кормен Т., Лейзерсон Ч., Ривест Р.* Алгоритмы: Построение и анализ. М.: МЦНМО. 2001.
16. *Métivier Y., Robson J.M., Zemmari A.* A distributed enumeration algorithm and applications to all pairs shortest paths, diameter. . . // Information and Computation. 2016. V. 247. P. 141–151.

Статья представлена к публикации членом редколлегии П.Ю. Чеботаревым.

Поступила в редакцию 11.02.2025

После доработки 29.05.2025

Принята к публикации 12.09.2025

СОДЕРЖАНИЕ

Линейные системы

- Ибрагимов Д.Н., Царьков К.А.** Численный метод решения задачи быстрогодействия для линейных нестационарных дискретных систем общего вида 3
- Хлебников М.В.** Синтез ПИД-регулятора для подавления ограниченных внешних возмущений 35

Нелинейные системы

- Мозжечков В.А.** Синтез релейных регуляторов автоколебательных систем управления на основе обучающих примеров 60

Стохастические системы

- Маслов А.А., Себекин Г.В., Степанов М.С., Степанов С.Н., Щурков А.О.**
Модель обслуживания мультисервисного трафика в узле доступа спутниковой сети связи с динамически изменяемой скоростью предоставления сервисов 75
- Потапов А.П., Галаев А.А.** Траекторное противодействие линейному наблюдателю 92

Оптимизация, системный анализ и исследование операций

- Кузнецов О.П.** Распределенный алгоритм нумерации вершин графа, совмещенный с построением дерева обхода в ширину 110

C O N T E N T S

Linear Systems

Ibragimov D.N., Tsarkov K.A. Numerical Method for Solving the Time-Optimization Problem for Linear Non-Stationary Discrete-Time Systems of General Form	3
Khlebnikov M.V. PID Controller Design for Rejection of Bounded Exogenous Disturbances	35

Nonlinear Systems

Mozzhechkov V.A. Relay Controller Design in Self-Oscillating Control Systems Based on Training Examples	60
--	----

Stochastic Systems

Maslov A.A., Sebekin G.V., Stepanov M.S., Stepanov S.N., Shchurkov A.O. Model for Servicing Multiservice Traffic in an Access Node of a Satellite Communications Network with a Dynamically Variable Service Delivery Rate	75
Potapov A.P., Galyaev A.A. Trajectory Countermeasures Against a Linear Observer	92

Optimization, System Analysis, and Operations Research

Kuznetsov O.P. Distributed Graph Vertex Numbering Algorithm Combined with Breadth-First Tree Construction	110
--	-----