ТЕОРЕТИЧЕСКИЙ И ПРИКЛАДНОЙ НАУЧНО-ТЕХНИЧЕСКИЙ ЖУРНАЛ

<u> ШЕХАТРОНИКА</u> ГОМАТИЗАЦИЯ, АВЛЕНИЕ



Издается с ноября 2000 года

ISSN 1684-6427

DOI 10.17587/issn.1684-6427

Главный редактор: ФИЛИМОНОВ Н. Б., л.т.н

Заместители главного редактора: БОЛЬШАКОВ А. А., л.т.н. ПОДУРАЕВ Ю. В., д.т.н. ЮЩЕНКО А. С., д.т.н.

Ответственный секретарь: БЕЗМЕНОВА М.Ю

Редакционный совет: АНШАКОВ Г. П., чл.-корр. РАН БОЛОТНИК Н. Н., чл.-корр. РАН ВАСИЛЬЕВ С. Н., акад. РАН ЖЕЛТОВ С. Ю., акад. РАН КАЛЯЕВ И.А., акад. РАН КУЗНЕЦОВ Н. А., акад. РАН КУРЖАНСКИЙ А. Б., акад. РАН МИКРИН Е. А., акад. РАН ПЕШЕХОНОВ В. Г., акад. РАН РЕЗЧИКОВ А. Ф., чл.-корр. РАН СЕБРЯКОВ Г. Г., чл.-корр. РАН СИГОВ А. С., акад. РАН СОЙФЕР В. А., акад. РАН СОЛОМЕНЦЕВ Ю. М., чл.-корр. РАН ФЕДОРОВ И. Б., акад. РАН ЧЕНЦОВ А. Г., чл.-корр. РАН ЧЕРНОУСЬКО Ф. Л., акад. РАН ЩЕРБАТЮК А. Ф., чл.-корр. РАН ЮСУПОВ Р. М., чл.-корр. РАН

Редколлегия:

DANIELE Z., PhD, Италия DORANTES D. J., PhD, Турция GROUMPOS P. P., PhD, Греция ISIDORI A., PhD, Италия KATALINIC B., PhD, Австрия LIN CH.-Y., PhD, Тайвань MASON O. J., PhD, Ирландия ORTEGA R. S., PhD, Франция SKIBNIEWSKI M. J., PhD, США STRZELECKI R. M., PhD, Польша SUBUDHI B. D., PhD, Индия АЛИЕВ Т. А., д.т.н., Азербайджан ГАРАЩЕНКО Ф. Г., д.т.н., Украина ТРОФИМЕНКО Е. Е., д.т.н., Беларусь БОБЦОВ А. А., д.т.н. БУКОВ В. Н., д.т.н. ЕРМОЛОВ И. Л., д.т.н. ИЛЬЯСОВ Б. Г., д.т.н. КОРОСТЕЛЕВ В. Ф., д.т.н. ЛЕБЕДЕВ Г. Н., д.т.н. ЛОХИН В М ЛТН ПАВЛОВСКИЙ В. Е., д.ф.-м.н. ПУТОВ В. В., д.т.н. ПШИХОПОВ В. Х., д.т.н. РАПОПОРТ Э. Я., д.т.н. СЕРГЕЕВ С. Ф., д.пс.н. ФИЛАРЕТОВ В. Ф., д.т.н. ФРАДКОВ А. Л., д.т.н. ФУРСОВ В. А., д.т.н. ЮРЕВИЧ Е. И., д.т.н.

Релакция: БЕЗМЕНОВА М. Ю. Директор издательства: АНТОНОВ Б. И.

СИСТЕМНЫЙ АНАЛИЗ. УПРАВЛЕНИЕ И ОБРАБОТКА ИНФОРМАЦИИ

СОДЕРЖАНИЕ

Дегтярев Г. Л., Файзутдинов Р. Н., Спиридонов И. О. Многокритериальный синтез

Рустамов Г. А., Фархадов В. Г., Рустамов Р. Г. Исследование К_с-робастных систем при

Волков В. Г., Демьянов Д. Н. Синтез и нейросетевая реализация ПИ регулятора

РОБОТЫ, МЕХАТРОНИКА И РОБОТОТЕХНИЧЕСКИЕ СИСТЕМЫ

Ibrahim I. N. A Comparative Study for an Inverse Kinematics Solution of an Aerial Manipulator Based on the Differential Evolution Method and the Modified Shuffled Frog-Leaping Algo-

ДИНАМИКА, БАЛЛИСТИКА И УПРАВЛЕНИЕ ДВИЖЕНИЕМ ЛЕТАТЕЛЬНЫХ АППАРАТОВ

Петрищев В. Ф. Энергосберегающий алгоритм автоматического управления принуди-725

Ван Гуоянь, Фомичев А. В. Алгоритм планирования безопасного маршрута движения

Солдаткин В. М., Солдаткин В. В., Никитин А. В., Арискин Е. О. Система измерения воздушных параметров вертолета с неподвижным приемником потока, ионно-

Журнал входит в Перечень рецензируемых научных изданий, в которых должны быть опубликованы основные результаты диссертаций на соискание ученой степени кандидата и доктора наук; журнал включен в систему Российского индекса научного цитирования, а также в БД RSCI на платформе Web of Science.

Информация о журнале доступна по сети Internet по адресу: http://novtex.ru/mech, e-mail: mech@novtex.ru

THEORETICAL AND APPLIED SCIENTIFIC AND TECHNICAL JOURNAL

MECHATRONICS, AUTOMATION, CONTROL No. 11 MEKHATRONIKA, AVTOMATIZATSIYA, UPRAVLE

Published since 2000

Editor-in-Chief FILIMONOV N. B.

Deputy Editors-in-Chief: BOLSHAKOV A. A. PODURAEV Yu. V. YUSCHENKO A. S.

Responsible Secretary: BEZMENOVA M. Yu.

Editorial Board: ANSHAKOV G. P. BOLOTNIK N. N. CHENTSOV A. G. CHERNOUSKO F. L. FEDOROV I. B. KALYAEV I. A. KURZHANSKI A. B. KUZNETSOV N. A. MIKRIN E. A PESHEKHONOV V G REZCHIKOV A. F. SCHERBATYUK A. F. SEBRYAKOV G. G. SIGOV A. S. SOJFER V. A SOLOMENTSEV Yu. M. VASSILYEV S. N. YUSUPOV R. M. ZHELTOV S. Yu.

Editorial Council:

ALIEV T. A., Azerbaijan DANIELE Z., PhD, Italy DORANTES D. J., PhD, Turkey GARASCHENKO F. G., Ukraine GROUMPOS P. P., PhD, Greece ISIDORI A., PhD, Italy KATALINIC B., PhD, Austria LIN CH.-Y., PhD, Taiwan MASON O. J., PhD, Ireland ORTEGA R. S., PhD, France SKIBNIEWSKI M. J., PhD, USA STRZELECKI R. M., PhD, Poland SUBUDHI B. D., PhD, India TROFIMENKO Ye. Ye., Belarus BOBTSOV A. A. BUKOV V. N. ERMOLOV I. L. FILARETOV V. F. FRADKOV V. L. FURSOV V. A. ILYASOV B. G. KOROSTELEV V. F. LEBEDEV G. N. LOKHIN V.M. PAVLOVSKY V. E. PUTOV V. V. PSHIKHOPOV V. Kh. RAPOPORT E. Ya. SERGEEV S F YUREVICH E. I.

Editorial Staff: BEZMENOVA M. Yu.

Director of the Publishing House: ANTONOV B. I. ISSN 1684-6427

DOI 10.17587/issn.1684-6427

The mission of the Journal is to cover the current state, trends and prospectives development of *mechatronics*, that is the priority field in the technosphere as it combines mechanics, electronics, automatics and informatics in order to improve manufacturing processes and to develop new generations of equipment. Covers topical issues of development, creation, implementation and operation of mechatronic systems and technologies in the production sector, power economy and in transport.

CONTENTS

SYSTEM ANALYSIS, CONTROL AND INFORMATION PROCESSING

ROBOT, MECHATRONICS AND ROBOTIC SYSTEMS

DYNAMICS, BALLISTICS AND CONTROL OF AIRCRAFT

Information about the journal is available online at: http://novtex.ru/mech.html, e-mail: mech@novtex.ru

СИСТЕМНЫЙ АНАЛИЗ, УПРАВЛЕНИЕ И ОБРАБОТКА ИНФОРМАЦИИ

УДК 681.5

DOI: 10.17587/mau.19.691-698

Г. Л. Дегтярев, д-р техн. наук, зав. кафедрой, gldegtyarev@mail.ru,

Р. Н. Файзутдинов, канд. техн. наук, доц., rustemfn@gmail.com, И. О. Спиридонов, аспирант,

Казанский национальный исследовательский технический университет им. А. Н. Туполева — КАИ, г. Казань

Многокритериальный синтез робастного регулятора нелинейной механической системы*

Рассматривается задача многокритериального синтеза робастного закона управления нелинейной механической системой, динамика которой описывается уравнениями Лагранжа второго рода. Подобные задачи имеют многочисленные практические приложения, например, при проектировании регуляторов роботехнических систем и гиростабилизированных платформ.

На практике при проектировании регуляторов часто приходится использовать неточные математические модели объектов управления. Поэтому важным требованием к проектируемой системе является обеспечение ее робастности при неопределенности параметров самой системы и внешних возмущений. В современной теории робастного управления наиболее разработаны методы синтеза регуляторов для линейных систем. При использовании этих методов для нелинейных систем нелинейности обычно включают в описание неопределенностей объекта. В результате синтезированные регуляторы получаются слишком консервативными, особенно при значительных неопределенностях. Развитием теории линейных робастных систем является нелинейная теория H_{∞} -оптимального управления, разработанная на базе теории дифференциальных игр. Методы нелинейной теории позволяют обеспечить робастную устойчивость синтезируемых систем управления. Однако для синтеза нелинейного H_{∞} -управления необходимо решить дифференциальное уравнение в частных производных, что является достаточно трудной задачей. Кроме того, при использовании данного метода трудно обеспечить робастное качество процессов управления.

В статье для синтеза робастного закона управления использованы методы теории линейных систем, зависящих от параметров. Показано, что лагранжева система может быть адекватно представлена в виде квазилинейной параметрической модели. С вычислительной точки зрения процедура синтеза сводится к технике выпуклой оптимизации при ограничениях, выраженных в форме линейных матричных неравенств (ЛМН). Измеряемые параметры включаются в закон управления, что позволяет обеспечить непрерывную подстройку коэффициентов регулятора к текущей динамике объекта управления и лучшее качество процессов управления по сравнению с H_{∞} -регулятором. Кроме того, использование аппарата ЛМН позволяет учесть при синтезе регулятора требования к качеству переходных процессов.

Так как квазилинейная параметрическая система непрерывно зависит от вектора параметров, то система ЛМН получается бесконечномерной. Эта бесконечномерная система уменьшается до конечного числа ЛМН с использованием политопного представления параметрической модели.

Приведен пример многокритериального синтеза робастного закона стабилизации и наведения линии визирования оптико-электронной системы, закрепленной на двухосной стабилизированной платформе.

Ключевые слова: многокритериальный синтез, робастный регулятор, нелинейная механическая система, политопная модель, линейные матричные неравенства

Введение

Движение множества технических систем, используемых на практике, описывается нелинейными уравнениями Лагранжа второго рода. Примерами таких систем являются роботы-манипуляторы, гиростабилизированные платформы, мостовые краны. Поэтому задача управления лагранжевыми системами привлекает внимание множества исследователей [1, 2]. При проектировании законов управления такими системами, как правило, предполагают, что параметры объекта известны точно. Однако на практике большинство параметров может быть определено только с той или иной степенью неопределенности. Поэтому одним из основных требований к проектируемой системе управления является обеспечение робастности системы в условиях неопределенности параметров самой системы и внешних возмущений [3, 4]. Большая часть результатов современной теории робастного управления получена на основе линейных методов, т. е. эти результаты основаны на предположении, что номинальный объект — линейный. Если ре-

^{*} Статья подготовлена при финансовой поддержке Министерства образования и науки РФ в рамках государственного задания подведомственных образовательных организаций (шифр проекта 8.3974.2017/4.6).

альный объект имеет нелинейную динамику, эти нелинейности обычно включают в описание неопределенностей объекта. Недостатком такого подхода является то, что он игнорирует доступную информацию о нелинейностях объекта, и синтезированные регуляторы получаются слишком консервативными, особенно при значительных неопределенностях.

Естественным развитием теории линейных робастных систем является нелинейная теория Н_∞-оптимального управления. Теоретической базой для разработки методов синтеза законов нелинейного H_{∞} -управления являлась теория дифференциальных игр [5]. В рамках данной теории были развиты методы, позволяющие обеспечить робастную устойчивость систем стабилизации и слежения за командным сигналом. Однако для синтеза нелинейного Н₂управления необходимо решить дифференциальное уравнение в частных производных, что является достаточно трудной задачей. При значительных неопределенностях параметров системы этот метод дает слишком консервативные результаты. Кроме того, при использовании данного метода трудно обеспечить робастное качество процессов управления.

Альтернативой нелинейной теории Н_∞-управления является теория линейных систем, зависящих от параметров [6]. В рамках данного подхода объект управления описывается множеством линейных моделей, непрерывно зависящих от вектора параметров. Многие нелинейные модели можно адекватно описать с помощью квазилинейных параметрических моделей. Достоинством метода является то, что при синтезе закона управления можно использовать эффективный аппарат линейной теории синтеза робастных систем. С вычислительной точки зрения применение метода сводится к широко используемой в настоящее время технике выпуклой оптимизации при ограничениях, выраженных в форме линейных матричных неравенств [7]. Измеряемые параметры включаются в закон управления, что позволяет обеспечить непрерывную подстройку коэффициентов регулятора к текущей динамике объекта управления и лучшее качество процессов управления по сравнению с Н_∞-регулятором. Кроме того, использование аппарата линейных матричных неравенств (ЛМН) позволяет учесть при синтезе регулятора требования к качеству переходных процессов.

В данной работе предложена методика многокритериального синтеза робастного регулятора лагранжевой системы с использованием методов теории линейных параметрических систем. В первом разделе приведена постановка задачи. Во втором разделе описана методика расчета параметров регулятора с использованием квазилинейной параметрической модели лагранжевой системы. В третьем разделе приведен пример синтеза регулятора.

Постановка задачи

Рассмотрим нелинейную механическую систему, движение которой описывается уравнениями Лагранжа второго рода в векторно-матричном виде [2]:

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = \tau + \tau_e, \qquad (1)$$

где $q \in \mathbb{R}^m$ — вектор обобщенных координат; M(q) — матрица инерции; $C(q, \dot{q})$ — матрица кориолиосовых и центростремительных сил; G(q) — *m*-мерный вектор потенциальных (гравитационных) сил; τ — *m*-мерный вектор управляющих обобщенных сил; τ_e — *m*-мерный вектор внешних возмущающих обобщенных сил.

Зададим желаемую траекторию $q_d(t)$ системы (1) и введем векторы ошибки $e_q(t)$ и скоростной ошибки $\dot{e}_q(t)$:

$$e_{q}(t) = q(t) - q_{d}(t);$$
 (2)

$$\dot{e}_q(t) = \dot{q}(t) - \dot{q}_d(t). \tag{3}$$

Для обеспечения отслеживания системой (1) желаемой траектории $q_d(t)$ необходимо синтезировать закон управления, обеспечивающий выполнение соотношений $e_a(t) = 0$, $\dot{e}_a(t) = 0$.

Для оценки робастности системы управления представим матрицы уравнения (1) в виде

$$\begin{split} M(q) &= M_0(q) + \Delta M(q);\\ C(q,\dot{q}) &= C_0(q,\dot{q}) + \Delta C(q,\dot{q});\\ G(q) &= G_0(q) + \Delta G(q), \end{split}$$

где $M_0(q)$, $C_0(q, \dot{q})$, $G_0(q)$ — номинальные матрицы, определенные при точных значениях параметров системы; $\Delta M(q)$, $\Delta C(q, \dot{q})$, $\Delta G(q)$ — неопределенности модели, определяемые неопределенностями параметров системы. В результате уравнение (1) приобретает следующий вид:

$$M_0(q)\ddot{q} + C_0(q,\dot{q})\dot{q} + G_0(q) = \tau + \delta(q,\dot{q},\ddot{q}), \quad (4)$$

где

$$\delta(q, \dot{q}, \ddot{q}) = -(\Delta M(q)\ddot{q} + \Delta C(q, \dot{q})\dot{q} + \Delta G(q)) + \tau_e.$$

Используя соотношения (2), (3), запишем уравнения, описывающие динамику ошибок системы (4) в форме пространства состояния:

$$\begin{cases} \frac{de_q}{dt} = \dot{e}_q; \\ \frac{d\dot{e}_q}{dt} = -M_0^{-1}C_0(q,\dot{q})\dot{e}_q + u + w, \end{cases}$$
(5)

где $u = M_0^{-1}(\tau - M_0(q)\ddot{q}_d - C_0(q,\dot{q})\dot{q}_d - G_0(q))$ — вектор управления; $w = M_0^{-1}\delta(q,\dot{q},\ddot{q})$ — вектор возмущающих воздействий.

Введем вектор состояния $x = [e_q^{\mathsf{T}} \quad \dot{e}_q^{\mathsf{T}}]^{\mathsf{T}}$ и перепишем уравнения (5) в матричном виде:

$$\dot{x} = A(\dot{q}, q)x + Bu + Bw, \tag{6}$$

где

$$A(\dot{q},q) = \begin{bmatrix} 0_{m \times m} & I_m \\ 0_{m \times m} & -M_0^{-1}C_0(q,\dot{q}) \end{bmatrix}; B = \begin{bmatrix} 0_{m \times m} \\ I_m \end{bmatrix}; (7)$$

 $I_m - m \times m$ -мерная единичная матрица; $0_{m \times m} - m \times m$ -мерная нулевая матрица.

Вектор управляющих сил т вычисляется через управление *и* как

$$\tau = M_0(q)(u + \ddot{q}_e) + C_0(q, \dot{q})\dot{q}_e + G_0(q).$$

Матрица $A(\dot{q},q)$ зависит только от обобщенных координат q(t) и обобщенных скоростей $\dot{q}(t)$. Поэтому систему (6) можно рассматривать как квазилинейную параметрическую систему [6], зависящую от вектора переменных параметров $\rho(t) = [q^{T}(t) \ \dot{q}^{T}(t)]^{T}$.

Представим систему (6) в виде

$$\begin{cases} \dot{x} = A(\rho(t))x + Bu + Bw; \\ z = Cx + D_1 u + D_2 w, \end{cases}$$
(8)

где $z \in \mathbb{R}^s$ — вектор выходных переменных; C, D_1 , D_2 — постоянные матрицы соответствующих размерностей. Выходные переменные zмогут включать ошибки отслеживания командных сигналов, затраты на управление и другие сигналы, которые необходимо минимизировать в результате применения управления u(t).

Предполагается, что параметры $\rho(t)$ можно вычислить в реальном масштабе времени на основании сигналов, поступающих с датчиков. При "замораживании" параметров $\rho(t)$ система (8) становится линейной стационарной системой.

Будем предполагать, что все состояния x(t) доступны измерению. Регулятор возьмем в виде функции от переменных состояния u = u(x).

Пусть на вход объекта (8) действует ограниченное по норме L_2 возмущение w(t), т. е.

$$\left\|w\right\|_{L_2} = \left(\int_0^\infty \left|w(t)\right|^2 dt\right)^{1/2} < \infty.$$

Тогда, если регулятор асимптотически стабилизирует замкнутую систему и обеспечивает выполнение неравенства

$$\sup_{w\neq 0} \frac{\|z\|_{L_2}}{\|w\|_{L_2}} < \gamma, \, \gamma > 0, \tag{9}$$

то уровень гашения возмущений в замкнутой системе меньше заданного числа γ [7]. Величина $\|z\|_{L_2} / \|w\|_{L_2}$ представляет собой коэффициент усиления системы между входом w и выходом z. Максимальный коэффициент усиления нелинейной системы для всех L_2 -ограниченных возмущений (сигналов с ограниченной энергией) представляет собой H_{∞} -норму системы [3, 4]. Задача синтеза регулятора, обеспечивающего асимптотическую устойчивость замкнутой системы и гашение возмущений в соответствии с условием (9), называется задачей синтеза субоптимального H_{∞} -регулятора [5]. H_{∞} -регулятор с минимальным значением γ считается [7] оптимальным H_{∞} -регулятором.

Субоптимальный H_{∞} -регулятор обеспечивает только асимптотическую устойчивость и гашение возмущений в замкнутой системе. Для обеспечения качества переходных процессов потребуем размещение полюсов системы в область $S(\alpha, r, \theta)$ (рис. 1). Размещение полюсов в область $S(\alpha, r, \theta)$ гарантирует [8], что переходные процессы в замкнутой системе имеют степень устойчивости не меньше α и колебательность не больше tg θ . Степень устойчивости характеризует быстродействие в системе. Значение полюсов ограничено сверху константой *r*.



Рис. 1. Область $S(\alpha, r, \theta)$ для размещения полюсов Fig. 1. The pole-placement region $S(\alpha, r, \theta)$

Задачу многокритериального синтеза закона управления линейной параметрической системой (8) поставим следующим образом. Найти регулятор в форме полной обратной связи по состоянию

$$u(t) = K(\rho(t))x(t), \qquad (10)$$

такой что замкнутая система (8), (10) устойчива, удовлетворяет H_{∞} -критерию (9), и полюса замкнутой системы находятся в области $S(\alpha, r, \theta)$.

Методика расчета параметров робастного многокритериального закона управления на основе линейных матричных неравенств

При управлении (10) уравнение замкнутой системы имеет вид

$$\begin{cases} \dot{x} = A_c(\rho(t))x + B_c w; \\ z = C_c x + D_c w, \end{cases}$$
(11)

где

$$A_{c}(\rho(t)) = A(\rho(t)) + BK(\rho(t)), B_{c} = B;$$

$$C_{c}(\rho(t)) = C + D_{2}K(\rho(t)), D_{c} = D_{1}.$$

Известно [7, 9], что при фиксированном $\rho(t)$ система (11) удовлетворяет H_{∞} -критерию (9) тогда и только тогда, когда существуют матричные переменные $Y = Y^{T} > 0$ и Z = KY, удовлетворяющие ЛМН

$$\begin{bmatrix} YA^{\mathsf{T}}(\rho) + A(\rho)Y + Z^{\mathsf{T}}B^{\mathsf{T}} + BZ & B & YC^{\mathsf{T}} + Z^{\mathsf{T}}D_{2}^{\mathsf{T}} \\ B^{\mathsf{T}} & -\gamma I & D_{1}^{\mathsf{T}} \\ CY + D_{2}Z & D_{1} & -\gamma I \end{bmatrix} < 0. (12)$$

Задачу размещения полюсов в область $S(\alpha, r, \theta)$ можно свести к решению следующих матричных неравенств [7, 9]:

$$A(\rho)Y + YA^{T}(\rho) + BZ + Z^{T}B^{T} + 2\alpha Y < 0; \quad (13)$$

$$\begin{bmatrix} -rY & A(\rho)Y + BZ\\ YA^{\mathsf{T}}(\rho) + Z^{\mathsf{T}}B^{\mathsf{T}} & -rY \end{bmatrix} < 0; \qquad (14)$$

$$\begin{bmatrix} N_{11} & N_{12} \\ N_{12} & N_{11} \end{bmatrix} < 0, \tag{15}$$

где

$$N_{11} = (A(\rho)Y + YA^{\mathsf{T}}(\rho) + BZ + Z^{\mathsf{T}}B^{\mathsf{T}})\sin\theta;$$

$$N_{12} = (A(\rho)Y + BZ - YA^{\mathsf{T}}(\rho) - Z^{\mathsf{T}}B^{\mathsf{T}})\cos\theta.$$

Если существуют матричные переменные $Y = Y^{T} > 0$ и Z, удовлетворяющие ЛМН (12)—(15), то параметры регулятора K определяются

как $K = ZY^{-1}$. Для линейной параметрической системы ЛМН должны выполняться для всех допустимых значений вектора $\rho(t)$. Так как вектор $\rho(t)$ непрерывен, то система ЛМН получается бесконечномерной. При использовании политопного представления модели (8) бесконечномерную систему ЛМН можно уменьшить до конечного множества ЛМН.

Пусть значения параметров принадлежат диапазонам

$$\rho_{j\min} \leq \rho_j \leq \rho_{j\max}, j = 1, 2, ..., l, l = 2m.$$
 (16)

Тогда вектор параметров $\rho(t)$ изменяется внутри гиперкуба пространства R^l с $N = 2^l$ вершинами Π_i , i = 1, 2, ..., N.

Матричным политопом [10] называют выпуклую оболочку конечного числа матриц A_i одинаковой размерности:

$$Co(A_i, i = 1, 2, ..., N) = \left\{ \sum_{i=1}^N \alpha_i A_i : \alpha_i \ge 0, \sum_{i=1}^N \alpha_i = 1 \right\}.$$

Политопное представление линейной параметрической модели (8) внутри гиперкуба (16) имеет вид [10]

$$\begin{cases} \dot{x} = \left(\sum_{i=1}^{N} \alpha_i A_i\right) x + Bu + Bw;\\ z = Cx + D_1 u + D_2 w, \end{cases}$$
(17)

где $A_i = A(\Pi_i)$ — значение матрицы $A(\rho(t))$ в вершине политопа Π_i .

С использованием политопной модели (17) решение бесконечномерной системы ЛМН (12)—(15) сводится к решению следующей конечной системы ЛМН [10]:

$$\begin{array}{ccc} YA_i^{\mathsf{T}} + A_iY + Z_i^{\mathsf{T}}B^{\mathsf{T}} + BZ_i & B & YC^{\mathsf{T}} + Z_i^{\mathsf{T}}D_2^{\mathsf{T}} \\ B^{\mathsf{T}} & -\gamma I & D_1^{\mathsf{T}} \\ CY + D_2Z_i & D_1 & -\gamma I \end{array} \right] < 0; (18)$$

$$A_iY + YA_i^{\mathrm{T}} + BZ_i + Z_i^{\mathrm{T}}B^{\mathrm{T}} + 2\alpha Y < 0; \qquad (19)$$

$$\begin{bmatrix} -rY & A_iY + BZ_i \\ YA_i^{\mathsf{T}} + Z_i^{\mathsf{T}}B^{\mathsf{T}} & -rY \end{bmatrix} < 0;$$
(20)

$$\begin{bmatrix} N_{11i} & N_{12i} \\ N_{12i} & N_{11i} \end{bmatrix} < 0, \ i = 1, 2, \dots, N,$$
 (21)

где

$$N_{11i} = (A_i Y + Y A_i^{\mathsf{T}} + B Z_i + Z_i^{\mathsf{T}} B^{\mathsf{T}}) \sin \theta;$$

$$N_{12i} = (A_i Y + B Z_i - Y A_i^{\mathsf{T}} - Z_i^{\mathsf{T}} B^{\mathsf{T}}) \cos \theta.$$

Если существуют матричные переменные $Y = Y^{T} > 0$ и Z_i , i = 1, 2, ..., N, удовлетворяющие системе ЛМН (18)—(21), то матрицу $K(\rho(t))$ регулятора (10) можно вычислить по формуле

$$K(\rho(t)) = \sum_{i=1}^{N} \alpha_i K_i, \sum_{i=1}^{N} \alpha_i = 1, \alpha_i \ge 0, \qquad (22)$$

где $K_i = Z_i Y^{-1}$ — значения матриц коэффициентов регуляторов в вершинах политопа Π_i .

Таким образом, задача многокритериального синтеза закона управления (10) линейной параметрической системой (8) сведена к решению системы ЛМН (18)—(21).

Параметры регулятора (10) определяются в зависимости от непрерывно изменяющегося вектора параметров $\rho(t)$. Траектория $\rho(t)$ заранее не известна, но вычисляется на основании сигналов датчиков в процессе функционирования системы. За счет этого в процессе реального времени происходит непрерывная настройка параметров регулятора, что обеспечивает высокое качество процессов управления.

Решение системы ЛМН (18)—(21) и вычисление матриц коэффициентов K_i можно провести заранее. Единственные вычисления, которые необходимо выполнить в процессе реального времени управления объектом (8), — это вычисления по формуле (22).

Пример расчета

Двухосная стабилизированная (СП) (рис. 2) состоит из трех тел: основания (тело 1), жестко прикрепленного к несущему телу, внешнего подвеса рысканья (тело 2), внутреннего тангажного подвеса (тело 3) и полезной оптикоэлектронной нагрузки (ПОЭН), установлен-



Рис. 2. Двухосная стабилизированная платформа Fig. 2. The two-axis inertially stabilized platform

ной в тангажной рамке внутреннего подвеса. С несущим телом связана базовая система координат $Ox_0y_0z_0$. С карданом рысканья связана система координат $Ox_1y_1z_1$, с тангажным карданом — система координат $Ox_2y_2z_2$. Будем предполагать, что начала всех трех систем координат совпадают. Угловые положения рам карданова подвеса (тел 2, 3) по отношению к предыдущим телам определяются соответственно углами ϕ_1 , ϕ_2 (рис. 2). Ориентация линии визирования определяется относительно базовой системы координат $Ox_0y_0z_0$. Эта ориентация определяется с помощью последовательности углов поворота φ₁ и φ₂ вокруг осей карданова подвеса, совмещающих базовую систему координат с системой координат $Ox_2y_2z_2$, связанной с карданом, на котором прикреплена ПОЭН.

Моменты инерции тел 2, 3 (все величины в кг \cdot м²):

$$J_{2} = \begin{bmatrix} J_{2x} & 0 & 0 \\ 0 & J_{2y} & 0 \\ 0 & 0 & J_{2z} \end{bmatrix} = \begin{bmatrix} 0,065 & 0 & 0 \\ 0 & 0,069 & 0 \\ 0 & 0 & 0,07 \end{bmatrix};$$
$$J_{3} = \begin{bmatrix} J_{3x} & 0 & 0 \\ 0 & J_{3y} & 0 \\ 0 & 0 & J_{3z} \end{bmatrix} = \begin{bmatrix} 0,018 & 0 & 0 \\ 0 & 0,024 & 0 \\ 0 & 0 & 0,025 \end{bmatrix}.$$

Уравнение динамики СП имеет вид [11]

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} = \tau + \tau_{e},$$

где

$$M(q) = \begin{bmatrix} J_{2z} + J_{3x} \sin^2 q_2 + J_{3z} \cos^2 q_2 & 0\\ 0 & J_{3y} \end{bmatrix};$$

$$C(q, \dot{q}) = \begin{bmatrix} 0 & -(J_{3z} - J_{3x})\dot{q}_1 \sin 2q_2\\ \frac{1}{2}(J_{3z} - J_{3x})\dot{q}_1 \sin 2q_2 & 0 \end{bmatrix}.$$

В качестве обобщенных координат примем следующий вектор: $q = [q_1(t) \ q_2(t)]^T = [\phi_1(t) \ \phi_2(t)]^T$. Запишем уравнения динамики ошибок СП в форме квазилинейной параметрической модели (8):

$$\begin{cases} \dot{x} = A(\rho(t))x + Bu + Bw; \\ z = Cx + D_1 u + D_2 w, \end{cases}$$
(23)

где матрица $A(\rho(t))$ определяется формулой (7): $\rho(t) = \begin{bmatrix} q^{T}(t) & \dot{q}^{T}(t) \end{bmatrix}^{T}; \quad C = \begin{bmatrix} I_{2} & I_{2} \end{bmatrix}; \quad B = \begin{bmatrix} 0_{2\times 2} \\ I_{2} \end{bmatrix}; \quad D_{1} = I_{2}; \quad D_{2} = 0_{2\times 2}; \quad x = \begin{bmatrix} e_{q}^{T} & \dot{e}_{q}^{T} \end{bmatrix}^{T}.$ Будем предполагать, что обобщенные координаты и скорости ограничены:

$$q_{1\min} \leq q_1 \leq q_{1\max}, q_{2\min} \leq q_2 \leq q_{2\max}; \quad (24)$$

$$\left|\dot{q}_{1}\right| \leq \dot{q}_{\max}, \left|\dot{q}_{2}\right| \leq \dot{q}_{\max}, \tag{25}$$

где $q_{1\min} = -\pi/4$, $q_{2\min} = -\pi/4$, $q_{1\max} = \pi/4$, $q_{2\max} = \pi/4$, $\dot{q}_{2\max} = 10$ рад/с.

Представим уравнение (23) в политопной форме:

$$\dot{x} = \left(\sum_{i=1}^{N} \alpha_i(t) A_i\right) x + Bu + Bw, \qquad (26)$$

где $\sum_{i=1}^{N} \alpha_i = 1$, $\alpha_i(t) \ge 0$, A_i — значение матрицы $A(\rho(t))$ в вершине политопа Π_i . Следуя работе [12], для уменьшения объема вычислений при

[12], для уменьшения объема вычислений при синтезе регулятора матрицу $A(q, \dot{q})$ приближенно заменим на $A(q) = A(q, \dot{q}_{max})$. Это позволяет уменьшить число вершин политопа (24), (25), в которых решается система ЛМН (18)— (21). Политоп определяется четырьмя вершинами: $\Pi_1 = [q_{1min}, q_{2min}], \Pi_2 = [q_{1max}, q_{2max}], \Pi_3 =$ $= [q_{1min}, q_{2max}], \Pi_4 = [q_{1max}, q_{2min}]$. Коэффициенты $\alpha_i(t)$ в политопном представлении (26) имеют следующий вид:

$$\begin{aligned} \alpha_1(t) &= \frac{(q_{1\max} - q_1(t))(q_{2\max} - q_2(t))}{(q_{2\max} - q_{2\min})(q_{1\max} - q_{1\min})};\\ \alpha_2(t) &= \frac{(q_1(t) - q_{1\min})(q_2(t) - q_{2\min})}{(q_{2\max} - q_{2\min})(q_{1\max} - q_{1\min})};\\ \alpha_3(t) &= \frac{(q_{1\max} - q_1(t))(q_2(t) - q_{2\min})}{(q_{2\max} - q_{2\min})(q_{1\max} - q_{1\min})};\\ \alpha_4(t) &= \frac{(q_1(t) - q_{1\min})(q_{2\max} - q_2(t))}{(q_{2\max} - q_{2\min})(q_{1\max} - q_{1\min})}.\end{aligned}$$

Для размещения полюсов замкнутой системы в область $S(\alpha, r, \theta)$ зададим $\alpha = 5, r = 13, \theta = \pi/4$. Уровень гашения возмущения возьмем $\gamma = 0,1$. Тогда матрицы коэффициентов регуляторов в вершинах политопа равны

$$\begin{split} K_1 &= \begin{bmatrix} -75,88 & 0 & -17,47 & -0,79 \\ 0 & -75,88 & 1,46 & -17,47 \end{bmatrix}; \\ K_2 &= \begin{bmatrix} -75,88 & 0 & -17,47 & 0,79 \\ 0 & -75,88 & -1,46 & -17,47 \end{bmatrix}; \\ K_3 &= \begin{bmatrix} -75,88 & 0 & -17,47 & 0,79 \\ 0 & -75,88 & -1,46 & -17,47 \end{bmatrix}; \end{split}$$

$$K_4 = \begin{bmatrix} -75,88 & 0 & -17,47 & -0,79 \\ 0 & -75,88 & 1,46 & -17,47 \end{bmatrix}.$$

Промоделируем два режима работы СП. Первый режим: приведение угловых положений платформы в нулевые значения: $q_d(t) = [0 \ 0]^{\text{T}}$. В качестве начальных значений приняты $\varphi_{10} = 0^\circ$, $\varphi_{20} = 6^\circ$. Результаты моделирования замкнутой системы с синтезированным регулятором показаны на рис. 3, 4.

На рис. 5 показаны результаты моделирования замкнутой системы при варьировании значений элементов тензора инерции J_2 . Взяты три значения тензора инерции $J_2 = J_2 + \delta$ при $\delta = -0,3; 0; 0,3$. Видно, что синтезированный регулятор обеспечивает достаточно хорошую робастность замкнутой системы управления в присутствии параметрических неопределенностей.





t, c

8

-3

-4

-5

-6 L 0

Рис. 4. Ошибка по углу ф2



Рис. 5. Ошибка по углу φ_1 при варьировании элементов матрицы J_2

Fig. 5. Graph on the φ_1 angle tracking error in the presence of matrix J_2 perturbations



Рис. 6. Ошибка по углу ϕ_1 при подавлении возмущающего момента

Fig. 6. Graph on the ϕ_1 angle tracking error in the presence of disturbance torque

Второй режим — это стабилизация угловой скорости СП при действии на систему возмущающих моментов. На рис. 6 показаны результаты применения субоптимальных H_{∞} -регуляторов по углу φ_1 для гашения возмущающего момента $\tau_e(t) = -0.05 \sin 2\pi (t - t_0)$ при $\gamma = 0.01$; 0.1; 0.25. Возмущающий сигнал подается в момент времени $t_0 = 3$ с.

Заключение

В работе предложена методика многокритериального синтеза робастного закона управления нелинейной механической системы,

описываемой уравнениями Лагранжа второго рода. Показано, что лагранжева система может быть адекватно представлена в виде квазилинейной параметрической модели. Процедура синтеза сведена к технике выпуклой оптимизации при ограничениях, выраженных в форме ЛМН, выполнение которых обеспечивает устойчивость замкнутой системы, гашение внешних возмущений и желаемое качество переходных процессов. Так как квазилинейная параметрическая система непрерывно зависит от вектора параметров, то система ЛМН получается бесконечномерной. Эта бесконечномерная система сведена к конечному числу ЛМН с использованием политопного представления параметрической модели.

Приведен пример синтеза системы стабилизации и наведения линии визирования оптико-электронной системы (ОЭС), закрепленной на двухосной стабилизированной платформе. Синтезированная система стабилизации и наведения линии визирования ОЭС промоделирована в системе MATLAB. Результаты моделирования подтвердили робастность и достаточно высокое качество синтезированного закона управления.

Список литературы

1. Зенкевич С. Л., Ющенко А. С. Управление роботами. Основы управления манипуляционными роботами. М.: Изд-во МГТУ им. Н. Э. Баумана, 2000.

2. Ortega R., Loria A., Nicklasson P. J., Sira-Ramirez H. Passivity-based Control of Euler-Lagrange Systems. London: Springer Verlag, 1998.

3. Поляк Б. Т., Щербаков П. С. Робастная устойчивость и управление. М.: Наука, 2002.

4. **Методы** робастного, нейронечеткого и адаптивного управления / Под ред. Н. Д. Егупова. М.: Изд-во МГТУ им. Н. Э. Баумана, 2002.

5. Basar T., Bernhard P. H_{∞} -Optimal Control and Related Minimax Problems. Berlin: Birkhauser, 1990.

6. **Briat C.** Linear parameter-varying and time delay systems. Analysis, observation, filtering & control. Berlin: Springer Verlag, 2015.

7. Баландин Д. В., Коган М. М. Синтез законов управления на основе линейных матричных неравенств. М.: Физматлит, 2006.

8. Бесекерский В. А., Попов Е. П. Теория систем автоматического регулирования. М.: Наука, 1972.

9. Chilali M., Gahinet P. H_{∞} design with pole placement constraints: an LMI approach // IEEE Trans. Aut. Contr, 1996. Vol. 41, N. 3. P. 358–367.

10. Apkarian P., Gahinet P., Becker G. Self-scheduled H_{∞} control of linear parameter-varying systems: a design example // Automatica, 1995. Vol. 31, N. 9. P. 1251–1261.

11. Бородин В. М., Спиридонов И. О., Файзутдинов Р. Н. Анализ динамики системы пассивной стабилизации линии визирования с четырехосным кардановым подвесом // Изв. вузов. Авиационная техника. 2016. Т. 59, № 4. С. 38–45.

12. Yu Z., Chen H., Woo P. Gain scheduled LPV H_{∞} control based on LMI approach for a robotic manipulator // Journal of Robotic Systems. 2002. Vol. 19, N. 12. P. 585–593.

Multiobjective Robust Controller Synthesis for Nonlinear Mechanical System

G. L. Degtyarev, gldegtyarev@mail.ru, R. N. Faizutdinov, rustemfn@gmail.com,

I. O. Spiridonov, igor.georg.2@gmail.com,

Kazan National Research Technical University named after A. N. Tupolev, 420111, Kazan, Russian Federation

Corresponding author: Degtyarev Gennadii L., D. Sc., Head of Automation and Control Department, Kazan National Research Technical University named after A. N. Tupolev,

420111, Kazan, Russian Federation, e-mail: gldegtyarev@mail.ru

Accepted on August 07, 2018

Abstract

In the paper multiobjective robust controller synthesis problem for nonlinear mechanical system described by Lagrange's equations of the second kind is considered. Such tasks have numerous practical applications, for example in controller design of robotic systems and gyro-stabilized platforms. In practice, we often have to use uncertain mathematical plant models in controller design. Therefore, ensuring robustness in presence of parameters perturbations and unknown external disturbances is an important requirement for designed systems. Much of modern robust control theory is linear. When the actual system exhibits nonlinear behavior, nonlinearities are usually included in the uncertainty set of the plant. A disadvantage of this approach is that resulting controllers may be too conservative especially when nonlinearities are significant. The nonlinear H_{x} optimal control theory developed on the basis of differential game theory is a natural extension of the linear robust control theory. Nonlinear theory methods ensure robust stability of designed control systems. However, to determine nonlinear H_{x} -control law, the partial differential equation have to be solved which is a rather complicated task. In addition, it is difficult to ensure robust performance of controlled processes when using this method. In this paper, methods of linear parameter-varying (LPV) systems are used to synthesize robust control law. It is shown, that Lagrange system may be adequately represented in the form of quasi-LPV model. From the computational point of view, the synthesis procedure is reduced to convex optimization techniques under constraints expressed in the form of linear matrix inequalities (LMIs). Measured parameters are incorporated in the control law, thus ensuring continuous adjustment of the controller parameters to the current plant dynamics and better performance of control processes in comparison with H_{∞} -regulators. Furthermore, the use of the LMIs allows to take into account the transient performance requirements in the controller synthesis. Since the quasi-LPV system depends continuously on the parameter vector, the LMI system is infinite-dimensional. This infinitedimensional system is reduced to a finite set of LMIs by introducing a polytopic LPV representation. The example of multiobjective robust control synthesis for electro-optical device's line of sight pointing and stabilization system suspended in two-axes inertially stabilized platform is given.

Keywords: multiobjective synthesis, robust controller, nonlinear mechanical system, polytopic model, linear matrix inequialities

Acknowledgements: This article was financially supported by the Ministry of education and science of the Russian Federation within the framework of the state task of subordinate educational institutions (project code 8.3974.2017/4.6)

For citation:

Degtyarev G. L., Faizutdinov R. N., Spiridonov I. O. Multiobjective Robust Controller Synthesis for Nonlinear Mechanical System, *Mekhatronika, Avtomatizatsiya, Upravlenie,* 2018, vol. 19, no. 11, pp. 691–698.

DOI: 10.17587/mau.19.691-698

References

1. Zenkevich S. L., Yushchenko A. S. Upravleniye robotami. Osnovy upravleniya manipulyatsionnymi robotami (Foundation of Control of Robot Manipulators), Moskow, Publishing house of MSTU named after N. E. Bauman Publ., 2000 (in Russian).

2. Ortega R., Loria A., Nicklasson P. J., Sira-Ramirez H. Passivity-based Control of Euler-Lagrange Systems, London, Springer Verlag, 1998.

3. **Polyak B. T., Shcherbakov P. S.** *Robastnaya ustoychivost' i upravleniye.* (Robust stability and control), Moscow, Nauka, 2002 (in Russian).

4. Egupov N. D. (ed.). *Metody robastnogo, neyro-nechetkogo i adaptivnogo upravleniya* (Methods of Robust, Neuro-fuzzy and Adaptive control), Moskow, Publishing house of MSTU named after N. E. Bauman Publ., 2002 (in Russian).

5. **Basar T., Bernhard P.** H_{∞} -Optimal Control and Related Minimax Problems, Berlin, Birkhauser, 1990.

6. **Briat C.** Linear parameter-varying and time delay systems. Analysis, observation, filtering & control, Berlin, Springer Verlag, 2015.

7. Balandin D. V., Kogan M. M. Sintez zakonov upravleniya na osnove lineinykh matrichnykh neravenstv (Synthesis of Control Laws on the Basis of Linear Matrix Inequalities), Moscow, Fizmatlit, 2006 (in Russian).

8. **Besekerskij V. A., Popov E. P.** *Teoriya sistem avtomaticheskogo regulirovaniya* (The Theory of Automatic Control Systems), Moscow, Nauka, 1975 (in Russian).

9. **Chilali M., Gahinet P.** H_{∞} design with pole placement constraints: an LMI approach, *IEEE Trans. Aut. Contr.*, 1996, vol. 41, no. 3, pp. 358–367.

10. Apkarian P., Gahinet P., Becker G. Self-scheduled H_{∞} control of linear parameter-varying systems: a design example, *Automatica*, 1995, vol. 31, no. 9, pp. 1251–1261.

11. Borodin V. M., Spiridonov I. O., Faizutdinov R. N. Analysis of Dynamics of a Passive Line-of-sight Stabilization System with Four-axis Gimbal Suspension, *Izv. Vuz. Av. Tekhnika*, 2016, vol. 59, no. 4, pp. 38–45. [Russian Aeronautics (Engl. Transl.), vol. 59, no. 4, pp. 480–488].

12. Yu Z., Chen H., Woo P. Gain scheduled LPV H_{∞} control based on LMI approach for a robotic manipulator, *Journal of Robotic Systems*, 2002, vol. 19, no. 12, pp. 585–593.

Г. А. Рустамов, д-р техн. наук, проф., gazanfar.rustamov@gmail.com, **В. Г. Фархадов**, канд. техн. наук, доц., **Р. Г. Рустамов**, докторант, Азербайджанский технический университет, г. Баку, Азербайджан

Исследование К_∞-робастных систем при ограниченном управлении

Исследуется влияние эффекта насыщения регулятора на расход энергии управления и робастные свойства систем, синтезированных без учета ограничения на управление при постановке задачи. В статье эта задача рассматривается применительно к K_{∞} -робастным системам управления.

Показано, что в следящих робастных системах в начальный момент времени управление принимает чрезмерно большое значение, что обеспечивает робастность динамического (переходного) режима. Это связано с тем, что в начальный момент времени начальные условия имеют большое значение. Основной причиной ухудшения робастных свойств связано со стеснением управления именно в начальном интервале времени. Обеспечение же робастности статического режима не требует больших усилий управления.

Для предельных систем характерны следующие динамические процессы. В пределе, когда коэффициент усиления стремится к бесконечности, все траектории мгновенно попадают на предельную плоскость s = 0. Дальнейшее движение в положение равновесия происходит по этой плоскости. При достаточно большом конечном коэффициенте, а следовательно, ограниченном управлении, все траектории стремятся к предельной плоскости и через заданное время установления достигают ее малой окрестности, определяемой заданной точностью слежения. Дальнейшее движение в положение равновесия происходит в пределах этой окрестности. Стеснение управления может привести к нарушению заданных показателей и устойчивости системы. В статье предельная плоскость названа "разомкнутым аттрактором" с правым концом в устойчивом положении равновесия.

Впервые с помощью компьютерной графики в трехмерном пространстве показаны траектории различных режимов движения: быстрое и медленное движения; установившееся движение в пределах заданной точности. Предложен интегральный критерий оценки показателя робастности.

Достоверность теоретических рассуждений подтверждена путем решения модельной задачи в блочно-визуальной среде имитационного моделирования Simulink.

Ключевые слова: неопределенность, робастная система управления, функция Ляпунова, большой коэффициент усиления, разомкнутый аттрактор, ограничение, насыщение, оценка робастности

Введение

Построение систем управления при наличии ограничения на управление с точки зрения практики является весьма актуальным. Тем не менее наличие ограничения значительно усложняет решение задач синтеза, в которых отыскивается управление с обратной связью. Основное замечание экспертов к задачам синтеза связано с указанной проблемой [см., например, 1, 2, с. 33—46]. Игнорирование ограничения часто обосновывается предположением о том, что система функционирует в малой окрестности рабочей точки, где вариации управления и переменных состояния малы.

В настоящее время, в отличие от методов нахождения программного управления, не имеется удовлетворительных инженерных решений, связанных с синтезом робастных регуляторов в условиях ограниченного управления. Вмешательство в естественный ход развития процесса путем введения различных ограничений затрудняет использование стандартных методов теории автоматического управления. При синтезе преобладают эвристические методы с множеством допущений, часть которых является не адекватными реальному процессу, который по определению является неопределенным.

В робастных следящих системах в начальный момент времени наблюдается высокое значение сигнала управления [2—7]. Эта особенность связана с требованиями робастности, заложенными в основу постановки задачи. Поскольку на практике имеются физические ограничения, то возникает задача выбора компромисса между предельным значением управления и показателем робастности. Ограниченность управления в начальном интервале времени в основном ухудшает робастность динамического (переходного) режима.

Впервые о K_{∞} -робастных системах управления упоминается в работах [3—7]. Эти системы относятся к предельным системам управления с большим коэффициентом усиления (СУБКУ). Начало данного направления заложено в классических работах М. Меерова [8, 9] и развито А. Востриковым, Г. Французовой [10—13], А. Филимоновым, Н. Филимоновым [1, 14] и др.

Основу метода синтеза СУБКУ составляет зависимость статической ошибки от коэффициента усиления разомкнутой цепи. Решение этой простой на первый взгляд задачи наталкивается на принципиальную трудность — увеличение коэффициента усиления приводит к потере устойчивости замкнутой системы. Имеющиеся исследования в этом направлении сконцентрированы вокруг решения этой проблемы. Первичной задачей в указанном направлении является синтез структуры, позволяющей беспредельное увеличение коэффициента усиления разомкнутой цепи без нарушения устойчивости. При этом задача сводится к структурному синтезу и в настоящее время не имеет строгого математического решения. Преобладают эвристические рассуждения.

Принципиальным отличием СУБКУ является то, что в них не используются номинальная модель неопределенного объекта и средства адаптации, в частности алгоритмы идентификации неопределенностей. К основным недостаткам можно отнести отсутствие аналитической формулы для определения коэффициента усиления *К* и усиление высокочастотных помех, имеющих доступ в регулятор. Кроме того, при синтезе СУБКУ, как и во многих других задачах синтеза, на управление ограничение не налагается.

Отличительной особенностью K_{∞} -робастных систем является тот факт, что основой синтеза являются не эвристические приемы, а применяется метод функции Ляпунова. Построенная таким образом система имеет методологическую основу, кроме того, регулятор является не астатическим, а робастным. Как известно, астатический регулятор обеспечивает нулевую статическую ошибку $\Delta_s = 0$ для определенного класса возмущений без их измерения. При изменении типа возмущений требуется вновь настраивать регулятор.

В статье исследуется случай, когда выходной сигнал регулятора ограничивается из-за выхода сигнала управления за пределы физических ограничений, т. е. при насыщении регулятора. Анализируется влияние этой особенности на потребление энергии управления и робастные показатели K_{∞} -робастных систем управления.

Постановка и решение задачи

В работах [3—7] рассматривается одномерный по входу и выходу (SISO) неопределенный объект, заданный скалярным уравнением

$$y^{(n)} = f(\mathbf{y}, t) + b(\mathbf{y}, t)u + \upsilon(t),$$

где $\mathbf{y} = (y, \dot{y}, ..., y^{(n-1)})^{\mathsf{T}} \in \mathbb{R}^n$ — доступный измерению или оценке вектор состояния; $y \in \mathbb{R}$ — регулируемый выход; $u \in \mathbb{R}$ — скалярное управление; $f(\mathbf{y}, t), b(\mathbf{y}, t) > 0$ — неизвестные ограниченные функции; $\upsilon(t)$ — неконтролируемое ограниченное внешнее возмущение.

Рассматривается задача слежения, состоящая в выборе такого управления с обратной связью u, которое после завершения переходной составляющей обеспечивает движение выхода y(t) объекта по эталонной траектории $y_d(t)$ с заданной точностью

$$|y_d(t) - y(t)| = |e(t)| \le \delta_s, \ t \ge t_s,$$

где t_s — заданное время установления (*settling time*) переходной составляющей для $\delta_s = (1...5)$ %-ной ошибки слежения.

С использованием функции Ляпунова $V = 1/2s^2$ получен робастный П закон управления:

$$u = Ks = K(c_1e + c_2\dot{e} + \dots + e^{(n-1)}) = K\mathbf{c}^{\mathsf{T}}\mathbf{e}, \quad (1)$$

где K > 0 —коэффициент усиления регулятора (достаточно большое число); $\mathbf{c}^{\mathrm{T}} = (c_1, c_2...1);$ $\mathbf{e}^{\mathrm{T}} = (e, \dot{e}, ..., e^{(n-1)}); K, \mathbf{c}^{\mathrm{T}}$ — параметры настройки регулятора. После некоторого конечного значения *K* достигаются заданные показатели качества.

Можно показать, что условию асимптотической устойчивости $dV/dt = s\dot{s} < 0$ также отвечают релейное u = -Ksigns и линейно-релейное управления u = -Ksat (s, Δ) , где Δ — ширина зоны линейности.

Относительно ошибки e(t) регулятор (1) является пропорционально-дифференциальным регулятором (n-1)-го порядка (ПДⁿ⁻¹ регулятором). Подобный регулятор в определенном смысле можно отнести к полиномиальным регуляторам [15] с передаточной функцией W = R(s)/C(s) при deg $C(s) = 0 \Rightarrow C(s) = c_0$, где $c_0 = 1/K$.

При стремлении *К* к бесконечности в пределе система описывается уравнением гиперплоскости, характеризующем ее свободное движение:

$$s = c_1 e + c_2 \dot{e} + \ldots + e^{(n-1)} = 0, \ s(0) = 0.$$

Для устойчивости движения угловые коэффициенты *c_i* характеристического полинома

$$H(p) = p^{n} + c_{n-1}p^{n-1} + \dots + c_{1}$$

должны удовлетворять условию $\operatorname{Re} p_i < 0$.

В пределе приведенная линейная система приобретает инвариантное свойство с устойчивым положением равновесия. При решении прикладных задач использование достаточно большего конечного значения коэффициента усиления *К* позволяет получить удовлетворительные показатели качества. При этом гиперплоскость s = 0 выступает в роли разомкнутого аттрактора (назовем это *аттракторной воронкой*), к которому притягиваются все траектории приведенной линейной системы.

В работе [16] управление (1) под названием "робастное эквивалентное управление" по аналогии с работой [17] использовано для синхронизации неидентичных динамических систем. Отметим, что структура системы синхронизации для многомерного случая не раскрыта и приведем некоторое разъяснение. При последовательной структуре выход ведущей системы подается на вход первой ведомой системы, а выход каждой ведомой системы является входом последующей ведомой системы. При параллельной структуре выход ведущей системы подается на входы ведомых систем, составляющих цепочку, а при смешанной структуре используются всевозможные комбинации указанных соединений.

В статье ограничение на управление реализуется релейно-линейным блоком $u_{sat} =$ = sat($u, u_0, \overline{u}_{sat}$) (saturation — насыщение), включенным на выход робастного регулятора. sat(•) выполняет функцию П регулятора с учетом интервальных ограничений $u_{sat} \leq |\overline{u}_{sat}|$.

В данном случае функция *sat*(·) описывается выражением

$$u_{sat} = \begin{cases} ku, \, \text{если} \, |u| \leq u_0; \\ \overline{u}_{sat} \text{sign} u, \, \text{если} \, |u| > u_0, \end{cases}$$

где u_0 , \overline{u}_{sat} — параметры настройки; $k = \overline{u}_{sat}/u_0$; u = Ks — входной сигнал.



Рис. 1. Статическая характеристика (*a*) и схема реализации (δ) функции sat (•) Fig. 1. The static characteristic (*a*) and the realization scheme (*b*) of the function sat (•)

На рис. 1 показаны статическая характеристика (*a*) и схема реализации (δ) функции *sat* (\cdot) в пакете *Simulink*.

Здесь $k = tg\alpha = 1 \Rightarrow u_0 = \overline{u}_{sat}$. При этом на интервале линейности $u \in [-u_0; +u_0]$ входной сигнал проходит на выход без изменения $u_{sat} =$ = u = Ks.

Робастный регулятор с учетом ограничения имеет n + 1 параметров настройки: n - 1 угловых коэффициентов c_i , i = 1, 2, ..., n - 1, плоскости s = 0, коэффициент усиления K и предельное значение $\pm \overline{u}_{sat}$. При n = 2 имеем всего три параметра настройки. Методика определения параметров c_i и K изложена в работах [3, 4].

Результаты моделирования

Пример. Рассмотрим особенности управления математическим маятником [18, с. 27] в случае ограниченности выходного сигнала регулятора при параметрической и сигнальной неопределенности:

$$d^{2}\theta/dt^{2} = -k_{1}\sin\theta + k_{2}u + \upsilon(t);$$

$$k_{1} = g/l; \quad k_{2} = 2/(ml^{2}),$$

где m — масса груза; θ — угол отклонения от положения равновесия (вертикаль); l — длина стержня; $\upsilon(t)$ — неконтролируемое ограниченное внешнее возмущение.

Генеральные составляющие модели:

$$f(\cdot) = -k_1 \sin\theta + \upsilon(t), \ b(\cdot) = k_2.$$

Обозначая $y = x_1 = \theta, x_2 = \dot{y} = \dot{\theta}$, уравнение маятника запишем в координатах состояния:

$$\dot{x} = x_1;$$

 $\dot{x}_2 = -k_1 \sin x + k_2 u + v(t);$
 $v = x_1.$

Пусть возмущение v(t) = 1 + sin(10t) + cos(4t), эталонная траектория (неявная эталонная модель) $y_d = 0,5sin(0,5t) + 0,5cos(t)$. Показатели качества: $t_s = 2$ с, $\delta_s = 2$ %. Начальные условия: $x(0) = (x_{10}, x_{20})^{T} = (2, 0)^{T}$. Соответствующее начальное условие по ошибке слежения: $e(0) = y_d(0) - y(0) = 0,5 - 2 = -1,5$.

Сначала рассмотрим особенности функционирования K_{∞} -робастной системы, когда не учитывается ограничение на управление. При n = 2 уравнение робастного регулятора $u = Ks = K(ce + \dot{e})$. Согласно работам [3, 4] при заданных показателях качества и начальных условиях для монотонной переходной характеристики угловой коэффициент (параметр настройки) линии $s = ce + \dot{e} = 0$ равен c = 2,16.

Номинальное значение параметров m = l = 1. Пусть модель имеет параметрическую неопределенность по m и l на довольно широких интервалах $0,5 \le m \le 1,5; 0,5 \le l \le 1,5$.

В работах [10, 11] динамика управляемого процесса в робастных системах с большим коэффициентом усиления разбита на два интервала: быстрое движение, когда траектории за короткое время достигают малой окрестности плоскости s = 0 и быстро затухают по экспоненте $\exp(-Kt)$; медленное движение по s = 0в положение равновесия. Эта особенность авторами названа эффектом локализации движения. В K_{∞} -робастных системах также наблюдается аналогичная закономерность.

На рис. 2, a-e показаны динамические характеристики системы при K = 120 без уче-

та ограничения на управляющий сигнал *и*. Моделирование выполнено при дискретных значениях параметров m = [0,5; 1,0; 1,5], l == [0,5; 1,0; 1,5]. Соответственно, $k_1 = [19,6; 9,8; 6,5],$ $k_2 = [16; 2; 0,6].$

На рис. 2 изображены пучки, состоящие из трех кривых, которые соответствуют различным комбинациям параметров модели. Таким образом демонстрируется, что при достаточно большом коэффициенте усиления (K = 120) концентрация пучков велика, и это свидетельствует об удовлетворительных робастных свойствах системы. На рис. 2, *е* показано, что при малых K (K = 5) разброс траекторий внутри пучка увеличивается, и это говорит об ухудшении робастности.

Установившийся режим — движение по коридору шириною $2|\delta_s|$ — наступает после момента времени t_s . При таком движении фазовая траектория не покидает малую окрестность положения равновесия (здесь — начало координат, рис. 2, *е*).

В правильности указанной последовательности движений в СУБКУ можно убедиться, если построить фазовый портрет, задавая время си-



Рис. 2. Характеристики K_∞-робастной системы управления:

a — переходные характеристики по выходу; *б* — ошибка слежения; *в* — сигнал управления; *г* — фазовый портрет при *T* = 10 с; *d*, *e* — графики изменения *s*(*t*) при *K* = 120 и *K* = 5

Fig. 2. Characteristics of the K_{∞} -robust control system:

a - transition characteristics for the output; δ - tracking error; e - control signal; e - phase portrait at T = 10 s; ∂ , e - graphs of the variation of s(t) for K = 120 and K = 5



Рис. 3. Характер движения системы в трехмерном пространстве Fig. 3. The nature of the system motion in three-dimensional space

муляции равным времени установления: $T = t_s = 2$ с. Поскольку полученная при этом картинка по виду похожа на фазовый портрет, показанный на рис. 2, c (T = 10 с), то ее здесь не будем демонстрировать.

Более наглядное представление различных участков движения можно получить в трехмерной системе координат (e, \dot{e}, t) (рис. 3).



Рис. 4. Динамические характеристики K_{∞} -робастной системы при ограниченном управлении

Fig. 4. The dynamic characteristics K_{∞} -robust system with limited control

На рис. З 1 — траектория быстрого движения, 2 — траектория медленного движения по линии s = 0, 3 — установившийся режим. Рисунок получен путем ввода функции *plot3*(e_1 , e_2 , t) из командного окна *MATLAB*. Здесь e_1 , e_2 обозначены ошибка слежения и ее производная.

Рассмотрим закономерности влияния ограничения на расход энергии управления. Вычислим энергию управления для трех комбинаций параметров k_1 , k_2 за t = 20 с (время симуляции) при отсутствии ограничения:

$$E_K = \int_{0}^{20} u^2 dt = [83,9; 384,1; 1365,0].$$

Основной причиной возникновения высоких значений E_K является то, что в начальный момент времени управляющий сигнал принимает чрезмерно большое значение u(0) = [377; 270; 125].

Построим характеристики системы при наличии ограничения $u_{sat} \in [+20 - 20]$. На рис. 4 показаны динамические характеристики K_{∞} робастной системы.

Данная схема отличается от схемы, используемой для моделирования системы без учета

> ограничения, лишь тем, что в ней после робастного ПД регулятора включен блок Saturation. В этом случае предельное значение управляющего сигнала для всех трех комбинаций параметров составляет $u_{sat} = \bar{u}_{sat}(0) = 20$. Из рис. 4, г видно, что ограниченность управления в основном влияет на робастность (здесь отклонение траектории от предельной точки s = 0) переходного (быстрое движение + медленное движение) режима.

При ограничении $u_{sat} \in [+20; -20]$ энергия управления за время t = 20 с

$$E_M = [9,6; 116,5; 542,8].$$

Сравним затраченную энергию управления до и после введения ограничения:

$$\Delta E = E_K - E_M = [41,6; 267,6; 722,2].$$

Как видно, при каждой комбинации параметров имеется



Рис. 5. Характеристики свободного движения при различных ограничениях Fig. 5. Characteristics of free movement under various restrictions

значительная экономия энергии управления, обусловленная наличием ограничения. Однако в переходном режиме (интервал $0 \le t \le t_s = 2$ с) наблюдается ослабление робастности: происходит некоторое расхождение траекторий от линии s = 0 (рис. 5).

На рис. 5, *а* и *б* влияние ограничения на управление показано для свободного движения системы при $y_d(t) = 0$, v(t) = 0 и трех значений $\overline{u}_{sat} = \pm [1; 5; 20]$ при $k_1 = 9,8$; $k_2 = 2$.

Графические построения позволяют еще раз констатировать известную истину, что для улучшения показателей системы (в том числе и робастных) следует расширить интервал ограничений. В данном примере $u_{sat} \in [+\overline{u}_{sat}; -\overline{u}_{sat}]$.

При построении робастных систем управления одной из важных задач является определение и оценка показателя робастности. В настоящее время не имеется инженерных методик оценки качественных показателей робастности неопределенных нелинейных систем. Оценка робастности на основе теории чувствительности, инвариантности и др. для инженерных приложений является обременительной и часто невостребованной на практике задачей.

В статье для оценки робастности динамического режима (быстрое движение + медленное движение) СУБКУ предложен интегральный критерий, характеризующий "сгущение траекторий" в окрестности гиперплоскости *s*(*t*) = 0:

$$J_{j} = \int_{0}^{t_{s}} |s_{j}(t)| dt, \quad j = 1, 2, \dots, N,$$
 (2)

где *t*_s —время установления.

Здесь N — число точек разбиения реквизитов объекта, например неопределенных параметров, на дискретные значения внутри известных интервалов. Разбиение проводится в целях более наглядного представления понятия робастности. Можно показать, что в пределе, когда $K \to \infty$, значения $J_j = 0$. При конечном K значение J_j отличается от нуля (см. рис. 2, *e*).

Концентрацию пучка по s(t) (см. рис. 2, ∂ , e) определим путем усреднения по ансамблю:

$$\overline{J} = (J_1 + J_2 + \dots + J_N)/N.$$

Без учета ограничения при $K = 200, t_s = 2$ с имеем пучок из трех траекторий (N = 3). Значения $J_1 = 0,01012;$ $J_2 = 0,03898; J_3 = 0,09266 \Rightarrow \overline{J} = 0,04725.$ При ограничении $u_{sat} \in [+20; -20]$ значения $J_1 = 0,03461; J_2 = 0,1227; J_3 = 0,3174 \Rightarrow \overline{J} = 0,1582.$

Соотношение $\lambda = 0,1582/0,04725 = 3,3481$. Как и следовало ожидать, наложение ограничения на управление ухудшает робастное свойство, в данном примере приблизительно в 3,35 раза.

В работе для демонстрации робастности интервал изменения параметров разбит на несколько дискретных значений. Это позволяет оценить робастность как "сгущение траекторий" по s(t) в окрестности гиперплоскости s(t) = 0. Если при моделировании на этих интервалах задать некоторую непрерывную функцию времени, например гармоническую, то получим единственную траекторию s(t) и, следовательно, $\overline{J} = J_1$.

Значение *К* должно обеспечивать выполнение условий $\overline{J} \leq \gamma$ и $|e(t)| \leq \delta_s$, $t \geq t_s$, где $\gamma > 0$ — заданное малое число.

Заключение

На основе теоретических результатов и компьютерного моделирования можно прийти к следующим выводам.

1. В предельных следящих системах с большим коэффициентом усиления, в частности K_{∞} -робастных системах, в начальном интервале времени управление принимает чрезмерно большое значение, которое обеспечивает робастность динамического режима. Эта особенность связана с тем, что в начальный момент времени начальное условие $e^{T}(0)$ имеет относительно большое значение. Например, при e(0) = -1.5; $\dot{e}(0) = 0$; c = 2.16; K = 120 значение $s(0) = \dot{e}(0) + ce(0) = -3.14 \Rightarrow u(0) = |377|$. 2. Сужение ограничений главным образом ухудшает робастность динамического режима из-за стеснения управления в начальном интервале времени.

3. Впервые с помощью компьютерной графики в трехмерной координатной системе (e, \dot{e}, t) дано наглядное представление участков фазовых траекторий, относящихся к различным видам движений: быстрое движение из произвольного начального состояния, которое обеспечивает попадание изображающей точки на плоскость s = 0; медленное движение по s = 0 — локализация А. С. Вострикова; установившееся движение в пределах заданной точности $\pm \delta_s$.

4. Для предельных систем управления с большим коэффициентом усиления предложена интегральная оценка показателя робастности, заключающаяся в вычислении интеграла от |s(t)|с верхней границей, равной заданному времени установления. Этот показатель характеризует отклонение реальной траектории s(t) относительно предельной s(t) = 0. Показано, что в пределе при $K \rightarrow \infty$ интеграл равен нулю.

5. Сформулировано условие для определения коэффициента усиления *К*.

6. Компьютерное моделирование предельных робастных систем значительно отстает от теоретических исследований.

7. В среде имитационного моделирования Simulink построена схема моделирования K_{∞} -робастных систем управления, которая может составить основу моделирования других классов подобных систем.

8. Теоретические положения обоснованы путем решения модельной задачи на примере гармонического осциллятора с параметрической неопределенностью. Компьютерное исследование позволило сделать ряд положительных выводов, имеющих важное прикладное значение.

Список литературы

1. **Филимонов А. Б., Филимонов Н. Б.** Метод больших коэффициентов усиления и эффект локализации движения в задачах синтеза систем автоматического управления // Мехатроника, автоматизация, управление. 2009. № 2. С. 2—10.

2. **Фуртат И. Б.** Управление с компенсацией возмущений. СПб.: Университет ИТМО, 2017. 65 с.

3. Рустамов Г. А. Робастная система управления с повышенным потенциалом // Известия Томского Политехнического Университета. 2014. Т. 324, № 5. С. 13—20.

4. **Рустамов Г. А.** К_∞-робастные системы управления // Мехатроника, автоматизация, управление. 2015. Т. 16, № 7. С. 435—442.

5. Rustamov G. A. Absolutely robust control systems // Automatic Control and Computer Sciences. 2013. Vol. 47, N. 5. P. 227–241.

6. Рустамов Г. А., Рустамов Г. А. Система робастного управления. Евразийский патент № 025475 от 30.12.2016 г.

7. Рустамов Г. А. Анализ методов построения робастных систем управления с большим коэффициентом усиления // Мехатроника, автоматизация, управление. 2018. Т. 19, № 6. С. 363—373.

8. **Мееров М. В.** Системы автоматического управления, устойчивые при бесконечно больших коэффициентах усиления // Автоматика и телемеханика. 1947. Т. 8, № 4. С. 225–243.

9. Мееров М. В. Синтез структур систем автоматического управления высокой точности. М.: Наука, 1967. 424 с.

10. Востриков А. С. Синтез систем регулирования методом локализации. Новосибирск: Изд-во НГТУ, 2007. 252 с.

11. Востриков А. С. Старшая производная и большие коэффициенты усиления в задаче управления нелинейными нестационарными объектами // Мехатроника, автоматизация, управление. 2008. № 5. С. 2—7.

12. Востриков А. С. Проблема синтеза регуляторов для систем автоматики: состояние и перспективы // Автометрия. 2010. Т. 46, № 2. С. 3–19.

13. Востриков А. С., Французова А. Г. Синтез ПИДрегуляторов для нелинейных нестационарных объектов // Автометрия. 2015. Т. 51, № 5. С. 53—60.

14. **Филимонов А. Б., Филимонов Н. Б.** Робастная коррекция динамических объектов в системах автоматического управления // Автометрия. 2015. Т. 51, № 5. С. 61–68.

15. **Тарарыкин С. В., Аполонский В. В., Терехов А. И.** Исследование влияния структуры и параметров полиномиальных регуляторов "входа-выхода" на робастные свойства синтезированных систем // Мехатроника, автоматизация, управление. 2013. Т. 152, № 11. С. 2—9.

16. Рустамов Г. А. Синхронизация неидентичных динамических систем с помощью эквивалентного управления // Известия Томского Политехнического Университета. 2014. Т. 325, № 5. С. 33—42.

17. Utkin V. I. Sliding Mode in Optimization and Control Problems. New York, Springer-Verlag, 1992. P. 387.

18. **Арнольд В. И.** Обыкновенные дифференциальные уравнения. М.: Наука, 1984. 272 с.

Research of K_{∞} -Robust Systems with Constrained Control

G. A. Rustamov, gazanfar.rustamov@gmail.com, V. G. Farchadov, vahidf@mail.ru, R. G. Rustamov, rustamshrustamov@gmail.com, Azerbaijan Technical University

> Corresponding author: Rustamov Gazanfar A., D. Sc., Professor, Azerbaijan Technical University, AZ1073, G. Javid avenue 25, Baku, Azerbaijan, e-mail: gazanfar.rustamov@gmail.com

Abstract

At the present, there are no satisfactory engineering solutions related to the synthesis of robust regulators taking into account the constraints on control. In this connection, it is important to study the influence of saturation effect of the controller on the robust properties of systems. In this paper, this problem is considered in connection with K_{∞} -robust control systems with a high gain. It is shown that in control limit systems, in particular, K_{∞} -robust systems at the initial instant of time, the control assumes an excessively large value. This ensures the robustness of the dynamic mode. The reason for this feature is related to the fact that at the initial instant of time the initial conditions has a great importance. The main reason for the deterioration of robust properties is due to the tight control in the initial time interval. Provision of robustness of the static mode does not require great control efforts. For the first time, using computer graphics in a three-dimensional coordinate system, taking into account the time, a visual representation of the sections of phase trajectories pertaining to different types of movements is given: rapid motion from an arbitrary initial state that ensures hitting the imaging point into the degenerate trajectory; slow motion along this trajectory; steady-state motion within the specified accuracy. For the limit control systems, an integral robustness estimate is proposed, which consists in calculating the integral of the absolute value of the slow motion trajectory. This indicator characterizes the discrepancy (dispersion) of the real trajectory with respect to the limiting trajectory. The reliability of theoretical reasoning is confirmed by solving a model problem in the block-vision environment of Matlab/Simulink.

Keywords: uncertainty, robust control system, Lyapunov function, high gain coefficient, attractor, constraints, saturation, estimation robustness

For.citation:

Rustamov G. A., Farchadov V. G., Rustamov R. G. Research of K_{∞} -Robust Systems with Constrained Control, *Mekhatronika*, *Avtomatizatsiya*, *Upravlenie*, 2018, vol. 19, no. 11, pp. 699–706.

DOI: 10.17587/mau.19.699-706

References

1. Filimonov A. B., Filimonov N. B. Metod bolshih kojefficientov usilenija i jeffekt lokalizacii dvizhenija v zadachah sinteza sistem avtomaticheskogo upravlenija (Large gain coefficients method and effect of the motion localization in the problem of synthesis of the automatic control system), Mekhatronika, Avtomatizatsiya, Upravlenie, 2009, no. 2, pp. 2–10 (in Russian).

2. **Furtat I. B.** *Upravlenie s kompensaciej vozmushhenij* (Control with compensation of disturbances), SPb, Publishing house of Universitet ITMO, 2017, pp. 65 (in Russian).

3. **Rustamov G. A.** *Robastnaja sistema upravlenija c povyshennym potencialom* (Robust Control System with High Potential), *Izvestija Tomskogo Politehnicheskogo Universiteta*, 2014, vol. 324, no. 5, pp. 13–20 (in Russian).

4. **Rustamov G. A.** K_{∞} -robastnye sistemy upravlenija (K_{∞} -robust control systems), *Mekhatronika, Avtomatizatsiya, Upravlenie,* 2015, vol. 16, no. 7, pp. 435–442 (in Russian).

5. Rustamov G. A. Absolutely robust control systems, *Automatic Control and Computer Sciences*, 2013, vol. 47, no. 5, pp. 227–241.

6. Rustamov G. A., Rustamov G. A. Sistema robastnogo upravlenija (Robust control system), Evrazijskij patent \mathbb{N} 025475 ot 30.12.2016 g. (in Russian).

7. Rustamov G. A. Analiz metodov postroenija robastnyh sistem upravlenija s bol'shim kojefficientom usilenija (Analysis of methods of design of robust control systems with high gain coefficient), Mekhatronika, Avtomatizatsiya, Upravlenie, 2018, vol. 19, no. 6, pp. 363–373 (in Russian).

8. **Meerov M. V.** Sistemy avtomaticheskogo upravlenija, ustojchivye pri beskonechno bol'shih kojefficientah usilenija (Automatic control system, stable at infinitely large gain factors), Avtomatika i Telemehanika, 1947, vol. 8, no. 4, pp. 225–243 (in Russian). 9. **Meerov M. V.** Sintez struktur sistem avtomaticheskogo upravlenija vysokoj tochnosti (Synthesis of structures of automatic control systems with high precision), Moscow, Nauka, 1967, 424 p. (in Russian).

10. **Vostrikov A. S.** *Sintez sistem regulirovanija metodom lokalizacii* (Synthesis of control systems by localization method), Novosibirsk, Publishing house of NGTU, 2007, 252 p. (in Russian).

11. **Vostrikov A. S.** *Starshaja proizvodnaja i bol'shie kojefficienty usilenija v zadache upravlenija nelinejnymi nestacionarnymi obektami* (The highest derivative and large coefficients in controlling the linear non-stationary objects), *Mekhatronika, Avtomatizatsiya, Upravlenie*, 2008, no. 5, pp. 2–7 (in Russian).

12. Vostrikov A. S. Problema sinteza reguljatorov dlja sistem avtomatiki: sostojanie i perspektivy (The problem of the synthesis of regulators for automation systems: state and prospects), Avtometrija, 2010, vol. 46, no. 2, pp. 3–19 (in Russian).

13. **Vostrikov A. S., Francuzova A. G.** *Sintez PID-reguljatorov dlja nelinejnyh nestacionarnyh obektov* (Synthesis of PID regulators for nonlinear nonstationary objects), *Avtometrija*, 2015, vol. 51, no. 5, pp. 53–60 (in Russian).

14. Filimonov A. B., Filimonov N. B. Robastnaja korrekcija dinamicheskih obektov v sistemah avtomaticheskogo upravlenija (Robust correction of dynamic objects in automatic control systems), Avtometrija, 2015, vol. 51, no. 5, pp. 61–68 (in Russian).

15. Tararykin S. V., Apolonskij V. V., Terehov A. I. Issledovanie vlijanija struktury i parametrov polinomial'nyh regulyatorov "vhoda-vyhoda" na robastnye svojstva sintezirovannyh system (Research of the influence of the structure and parameters of polynomial "input-output" regulators on the robust features of synthesized systems), Mekhatronika, Avtomatizatsiya, Upravlenie, 2013, vol. 152, no. 11, pp. 2–9 (in Russian).

16. **Rustamov G. A.** Sinhronizacija neidentichnyh dinamicheskih sistem s pomoshh'ju jekvivalentnogo upravlenija (Synchronization of non-identical dynamic systems using equivalent control), *Izvestija Tomskogo Politehnicheskogo Universiteta*, 2014, vol. 325, no. 5, pp. 33–42 (in Russian).

17. Utkin V. I. Sliding Mode in Optimization and Control Problems, New York, Springer-Verlag, 1992, 387 p.

18. **Arnold V. I.** Obyknovennye differencial'nye uravnenija (Ordinary differential equations), Moscow, Nauka, 1984, 272 p. (in Russian).

В. Г. Волков, вед. инженер-конструктор¹, аспирант², vgvolkov93@mail.ru, **Д. Н. Демьянов,** канд. техн. наук, доц²,

¹НТЦ ПАО "КАМАЗ", г. Набережные Челны,

²Казанский (Приволжский) федеральный университет, г. Казань

Синтез и нейросетевая реализация ПИ регулятора адаптивного круиз-контроля грузового автомобиля*

Рассматривается проблема разработки алгоритма функционирования адаптивного круиз-контроля, работающего в условиях, изменяющихся в широких пределах передаточного отношения трансмиссии и скорости движения грузового автомобиля. Функционирование классической системы круиз-контроля, как правило, основывается на использовании ПИД регулятора с постоянными коэффициентами. Однако несмотря на простоту настройки и физической реализации, а также относительно высокую робастность данный класс управляющих устройств может не обеспечивать оптимальное функционирование системы круиз-контроля во всех режимах движения в силу нестационарности и нелинейности объекта управления. Для преодоления указанных недостатков в рамках данного исследования рассматривается возможность нейросетевой реализации алгоритма адаптивного круиз-контроля грузового автомобиля.

Предложена математическая модель продольного движения грузового автомобиля, предназначенная для решения задачи анализа и синтеза системы управления. Проведена настройка коэффициентов ПИ регулятора для управления продольной скоростью при различных режимах движения грузового автомобиля. Показано, что коэффициенты регулятора изменяются по достаточно сложному закону. Предложен алгоритм функционирования адаптивного круизконтроля, основанный на аппроксимации законов изменения коэффициентов ПИ регулятора с помощью искусственной нейронной сети. Используемая сеть представляет собой многослойный персептрон и для обеспечения высокого качества аппроксимации содержит десять нейронов на скрытом слое. Обучение нейронной сети осуществлялось методом Левенберга—Марквардта на основе выборки общим объемом 500 точек, полученной с помощью стандартных методов синтеза регуляторов. Корректность полученных результатов была подтверждена результатами компьютерного моделирования разгона автомобиля от 0 до 100 км/ч, показывающими, что в процессе движения коэффициенты ПИ регулятора, обеспечивающие требуемое качество переходных процессов, значительно изменяются в зависимости от текущего состояния транспортного средства. Предложенный метод аппроксимации законов изменения коэффициентов ПИ регулятора может быть в дальнейшем использован для построения адаптивных систем управления, способных эффективно функционировать в различных режимах.

Ключевые слова: адаптивный круиз-контроль, нейронные сети, ПИД регулятор, многослойный персептрон

Введение

В связи с постоянным ростом уровня урбанизации, резким увеличением числа транспортных средств (TC) на дорогах, а также ужесточающимися требованиями к комфорту, средней скорости и безопасности движения как на больших автомагистралях, так и в городских условиях в настоящее время активно развиваются системы активной помощи водителю.

Традиционный круиз-контроль представляет собой классическую систему автоматического управления, как правило, основанную на использовании ПИД регуляторов [1]. Несмотря на простоту настройки и физической реализации, данный класс управляющих устройств обладает существенным ограничением: в случае, когда объект управления имеет нелинейную природу, коэффициенты регулятора гарантированно обеспечивают цели управления лишь в некоторой окрестности точки пространства состояний [2]. В процессе функционирования система может выйти за пределы данной окрестности, тогда для поддержания желаемой динамики переходного процесса необходимо перенастроить коэффициенты регулятора для новых условий.

Для преодоления вышеописанного ограничения были разработаны различные варианты адаптивных ПИД регуляторов [2]. Например, для самонастройки коэффициентов регулятора в процессе его функционирования используются таблицы параметров (gain scheduling) для нескольких режимов работы объекта [2, 3]. В этом случае для определения коэффициентов в промежуточных режимах работы необходимо использовать различные методы аппроксимации и интерполяции. Развитием такого подхода может являться использование методов искусственного интеллекта, в частности нейронных сетей, позволяющих эффективно

^{*} Работа выполнена при финансовой поддержке РФФИ (грант № 16-38-00042).

решать задачу аппроксимации сильно нелинейных функций, какими часто являются зависимости коэффициентов ПИД регулятора от режимов работы объекта управления [4].

Таким образом, целью работы является синтез алгоритмов управления адаптивного круизконтроля грузового автомобиля, использующего нейронные сети для аппроксимации зависимости коэффициентов ПИД регулятора от режимов работы автомобиля.

Математическая модель грузового автомобиля как объекта управления

В настоящее время в автомобилестроении для решения большинства задач синтеза систем управления используются математические модели [5]. Система уравнений, описывающая работу двигателя внутреннего сгорания, имеет вид

$$\begin{cases} n_{\min}, \text{если } n_{\text{ДBC}} \leq n_{\min}; \\ J_{\text{ДBC}} \frac{dn_{\text{ДBC}}}{dt} = \alpha_{\text{ДBC}} M_{\text{ДBC}} - M_{\text{сц}}, \\ \text{если } n_{\min} \leq n_{\text{ДBC}} \leq n_{\max}; \\ n_{\max}, \text{если } n_{\text{ДBC}} \leq n_{\max}, \end{cases}$$
(1)

где $\alpha_{\rm дBC}$ — степень нажатия на педаль управления подачей топлива; $J_{\rm дBC}$ — момент инерции двигателя, кг·м²; $n_{\rm дBC}$ — частота вращения коленчатого вала двигателя, мин⁻¹; $n_{\rm min}$ — минимальная частота вращения коленчатого вала двигателя, мин⁻¹; $n_{\rm max}$ — максимальная частота вращения коленчатого вала, мин⁻¹; $M_{\rm дBC}$ — полезный крутящий момент, вырабатываемый двигателем, Н·м; $M_{\rm cu}$ — крутящий момент сцепления, затрачиваемый на преодоление инерции трансмиссии и автомобиля, Н·м.

Моделирование продольного движения грузового автомобиля основывается на следующем дифференциальном уравнении [6, 7]:

$$m_{\rm a}\frac{dv}{dt} = F_{\rm T} - m_{\rm a}g(f\cos\alpha + \sin\alpha) - k_{\rm B}Av^2.$$
 (2)

Здесь m_a — полная масса автомобиля, кг; v — скорость автомобиля, м/с; $F_{\rm T}$ — сила тяги, H; f — коэффициент сопротивления качению; α — угол уклона дороги, °; g — ускорение свободного падения, м/с²; A — площадь миделя, м²; $k_{\rm B}$ — коэффициент сопротивления воздуха, кг·м³. Для учета возможного проскальзывания дисков сцепления и колес автомобиля введем дополнительное уравнение, описывающее динамику узлов трансмиссии

$$(J_{\rm Tp} + J_{\kappa})\frac{d\omega}{dt} = M_{\rm cu}iu_{\rm pagg}u_{\rm r,\pi}\eta - M_{\rm T},\qquad(3)$$

где $J_{\rm тp}$ — приведенный к ведущим колесам момент инерции трансмиссии, кг·м²; $J_{\rm k}$ — суммарный момент инерции ведущих колес, кг·м²; ω — скорость вращения ведущих колес автомобиля, рад/с; *i* — текущее передаточное число коробки передач; $u_{\rm разд}$ — текущее передаточное число раздаточной коробки; $u_{\rm г.п}$ — передаточное число главной передачи; η — общий КПД трансмиссии; $M_{\rm T}$ — тяговый крутящий момент между шиной и дорогой, Н·м.

Крутящий момент, передаваемый сцеплением, и сила тяги вычисляются по формулам

$$M_{\rm cu} = (1 - \gamma_{\rm cu}) M_{\rm cu}^{\rm max} \tanh\left(2\frac{n_{\rm ДBC} - \frac{30\omega}{\pi}}{\Delta n}\right); \quad (4)$$
$$F_{\rm T} = \frac{1}{r} M_{\rm T}^{\rm max} \tanh\left(2\frac{\omega - \frac{v}{r}}{\Delta \omega}\right). \quad (5)$$

Здесь γ_{cu} — степень открытия сцепления; M_{cu}^{max} — максимальный крутящий момент, передаваемый сцеплением, Н·м; M_{T}^{max} — максимальный крутящий момент, передаваемый шинами, H·м; r — динамический радиус ведущих колес, м; Δn — относительная скорость проскальзывания дисков сцепления, при которой достигается 95 % максимального крутящего момента, передаваемого сцеплением, мин⁻¹; $\Delta \omega$ — относительная скорость проскальзывания колес, при которой достигается 95 % максимального тягового крутящего момента, мин⁻¹.

При анализе и синтезе алгоритмов функционирования адаптивного круиз-контроля использовались параметры автомобиля, представленные ниже:

Момент инерции двигателя, кг·м ² 0,7500
Минимальная поддерживаемая частота
вращения, мин ⁻¹
Максимальная поддерживаемая частота
вращения, мин ⁻¹
Масса автомобиля, кг
Радиус ведущих колес, м
Коэффициент сопротивления качению 0,0091
Площадь миделя, м ²

Коэффициент сопротивления воздуха, кг/м ³ 0,4816
Приведенный момент инерции трансмиссии,
$\kappa \Gamma \cdot M^2 \dots \dots$
Суммарный момент инерции ведущих колес,
$\kappa \Gamma \cdot M^2 \dots \dots$
Передаточное число раздаточной коробки 0,9170
Передаточное число главной передачи6,5300
Общий КПД трансмиссии
Максимальный крутящий момент,
передаваемый сцеплением, Н/м 1200
Максимальный крутящий момент,
передаваемый шинами, Н/м

На основе уравнений (1)—(5) в среде имитационного моделирования *Simulink* системы компьютерной математики *MATLAB* была реализована имитационная модель продольного движения грузового автомобиля. Уравнениям (1)—(3) в имитационной модели соответствуют блоки *Engine* (Двигатель), *Powertrain* (Трансмиссия) и *Chassis* (Шасси).

Блок Engine на основе нагрузки на двигатель со стороны сцепления и положения педали управления подачей топлива вычисляет частоту вращения коленчатого вала двигателя, решая дифференциальное уравнение (1). На основе частоты вращения коленчатого вала двигателя, скорости автомобиля, степени открытия сцепления, положения педали тормоза, а также выбранной передачи, блок Powertrain вычисляет нагрузку на двигатель и тяговый крутящий момент на ведущих колесах, передаваемые блокам Engine и Chassis, соответственно. Кроме того, благодаря введению в систему уравнений (4)—(5) модель учитывает возможность возникновения проскальзывания сцепления и ведущих колес при решении дифференциального уравнения (2). В свою очередь, блок Chassis вычисляет скорость движения транспортного средства согласно дифференциальному уравнению (3).

Из уравнений (1)—(5) видно, что модель прямолинейного движения автомобиля является существенно нелинейной. В частности, в уравнении (1) присутствует полезный крутящий момент двигателя $M_{\rm ДBC}$, как правило, определяемый по многопараметровой характеристике двигателя, которая является нелинейной функцией, связывающей крутящий момент и частоту вращения коленчатого вала двигателя. Также в уравнении (2) присутствует слагаемое, пропорциональное квадрату скорости автомобиля и характеризующее сопротивление воздуха. Кроме того, выражения (4)—(5) содержат функцию гиперболического тангенса.

Отдельно обратим внимание на тот факт, что при движении автомобиля происходит изменение передаточного числа трансмиссии, что делает систему нестационарной.

Линеаризация модели объекта управления

Многие методы синтеза систем управления, в том числе те, что используются в пакете *Control System Toolbox* системы компьютерной математики *MATLAB*, предназначены для работы с линейными моделями динамических систем [5]. Поэтому для линеаризации математической модели (1)—(5) были приняты следующие допущения:

1) частота вращения коленчатого вала двигателя находится в допустимых пределах, т. е. $n_{\min} \leq n_{\text{ДВС}} \leq n_{\max}$ и, следовательно, математическая модель двигателя упрощается:

$$J_{\text{ABC}} \frac{dn_{\text{ABC}}}{dt} = \alpha_{\text{ABC}} M_{\text{ABC}} - M_{\text{cu}}; \qquad (6)$$

2) дорога идеально ровная, тогда $\sin \alpha = 0$ и $\cos \alpha = 1$, а уравнение (2) примет вид

$$m_{\rm a}\frac{dv}{dt} = F_{\rm T} - fm_{\rm a}g - k_{\rm B}Av^2; \qquad (7)$$

3) проскальзывания сцепления и колес малы, тогда выражения (4) и (5) преобразуются следующим образом:

$$M_{\rm cu} = (1 - \gamma_{\rm cu}) M_{\rm cu}^{\rm max} \left(2 \frac{n - \frac{30\omega}{\pi}}{\Delta n} \right); \qquad (8)$$

$$F_{\rm T} = \frac{1}{r} M_{\rm TSFM}^{\rm max} \left(2 \frac{\omega - \frac{\nu}{r}}{\Delta \omega} \right). \tag{9}$$

Поскольку в уравнении движения (7) попрежнему присутствует слагаемое, пропорциональное квадрату скорости автомобиля, его дальнейшая линеаризация проводится программно, отдельно для каждой рабочей точки в пакете *Control System Toolbox*.

Синтез алгоритмов управления скоростью

Целями функционирования круиз-контроля являются: поддержание скорости движения



Рис. 1. Структурная схема алгоритма управления адаптивного круиз-контроля грузового автомобиля Fig. 1. Adaptive cruise control algorithm block diagram

на заданном уровне, установление желаемой скорости движения при небольшом перерегулировании и за короткий промежуток времени. В силу достаточно большой инерционности объекта управления скорость изменения ошибки управления невелика, и рассматривать дифференциальную составляющую нет необходимости, поэтому для реализации алгоритма управления выберем ПИ регулятор [8]. Общая структурная схема алгоритма адаптивного круиз-контроля грузового автомобиля представлена на рис. 1.

В качестве состояний, оказывающих влияние на функционирование круиз-контроля, были выделены скорость автомобиля и текущая передача. Коробка передач рассматриваемого грузового автомобиля является десятиступенчатой, поэтому номер передачи выбирается из множества

$$I = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}.$$
 (10)





Разобьем область значений возможной скорости автомобиля (в км/ч) на набор интервалов таким образом, чтобы границы интервалов были минимальными значениями скорости движения TC на каждой из передач:

 $V = \{5,52; 6,77; 10,67; 13,08; 17,21; 21,10; 28,13; 34,44; 43,05; 52,82\}.$ (11)

На основе данных множеств была сформирована "сетка" рабочих точек. С помощью стандартных методик, представленных, например, в работе [9], был осуществлен синтез ПИ регуляторов в каждой точке пространства состояний из множеств (10), (11). Результаты расчета приведены на рис. 2 и 3 (см. вторую сторону обложки) в виде карт изменения коэффициентов ПИ регулятора.

Переходные характеристики, полученные при разных значениях коэффициентов ПИ регулятора, представлены на рис. 4. Видно, что полученные алгоритмы управления удовлетворяют требованиям качества, а именно отсутствует статическая ошибка, и перерегулирование не превышает 25 %.

Аппроксимация коэффициентов ПИ регулятора с помощью многослойного персептрона

Ввиду того что алгоритм управления может быть настроен оптимально лишь для некоторого конечного числа рабочих точек, а скорость автомобиля является переменной состояния и описывается некоторой непрерывной функцией времени, при движении автомобиля он может оказаться неоптимальным. Для того чтобы сгладить негативные эффекты от перехода из одного режима работы системы в другой, необходимо обеспечить непрерывное изменение коэффициентов регулятора. Существует множество различных методов аппроксимации и интерполяции разной степени эффективности, однако, как упоминалось выше, одним из самых эффективных способов считается обучение для решения этой задачи нейронной сети. Для аппроксимации функций чаще всего используется многослойный персептрон.

Для описания нелинейной зависимости коэффициентов ПИ регулятора от текущего состояния автомобиля была использована нейронная сеть, имеющая два входных нейрона, два выходных нейрона с линейной функцией активации, а также десять нейронов в скрытом слое. Кроме того, сеть имеет по два входных нейрона для получения информации о скорости движения и текущей передаче и два выходных нейрона с линейной функцией активации для вычисления текущих значений пропорционального и интегрального коэффициентов.

Полученные в результате синтеза коэффициенты были использованы в качестве обучающей выборки. Обучение проводилось методом Левенберга—Марквардта в *MATLAB* с применением пакета *Neural Network Toolbox*. Для обучения, валидации и тестирования многослойного персептрона использовалась выборка общим объемом 500 точек. На валидацию и тестирование результатов обучения отводилось по 15 % данных исходной выборки, т. е. по 75 точек соответственно.

На рис. 5 (см. вторую сторону обложки) представлены результаты обучения нейронной сети для вычисления пропорционального коэффициента K_p . Можно видеть, что коэффициент линейной регрессии для каждой из выборок близок к единице, что говорит о высоком качестве обучения нейронной сети.

Результаты компьютерного моделирования

Для тестирования разработанного алгоритма адаптивного круиз-контроля использовалась имитационная модель, основанная на структурной схеме, представленной на рис. 1, и реализованная в среде имитационного моделирования *Simulink*. Для моделирования работы коробки передач применялся аппарат конечных автоматов. На рис. 6 представлены результаты моделирования разгона автомобиля от 0 до 100 км/ч.



Рис. 6. Результаты моделирования разгона автомобиля от 0 до 100 км/ч:

a — частота вращения коленчатого вала двигателя, мин⁻¹; δ — скорость, км/ч; s — выбранная передача; c — пропорциональный коэффициент ПИ регулятора; ∂ — интегральный коэффициент ПИ регулятора

Fig. 6. Simulation results of vehicle acceleration from 0 to 100 kmh: a — engine crankshaft rotational speed, rpm; δ — vehicle velocity, kmh; e — current gear; e — PI-controller proportional gain; ∂ — PI-controller integral gain

Основываясь на ошибке по скорости, алгоритм вычисляет степень нажатия на педаль управления подачей топлива, необходимую для обеспечения желаемой скорости движения. На рис. 6 показано, что в процессе разгона ТС коэффициенты K_n и K_i ПИ регулятора, вычисляемые многослойным персептроном, изменяются. Пока ТС движется на выбранной передаче, его скорость изменяется непрерывно, и коэффициенты изменяются вместе с ней. Поскольку событие переключения передачи является дискретным, коэффициенты K_n и K_i при этом также изменяются дискретно, вследствие этого на рис. 6 наблюдаются существенные перепады их значений, соответствующие моментам переключения передач.

По графикам рис. 6, г, д можно видеть, что коэффициенты ПИ регулятора, обеспечивающие требуемое качество переходного процесса, значительно изменяются в зависимости от текущего состояния ТС. Кроме того, по рис. 2 и 3 (см. вторую сторону обложки) видно, что их поведение является существенно нелинейным. Все это оправдывает использование многослойного персептрона для аппроксимации данной зависимости.

В целом компьютерное моделирование показывает, что разработанный алгоритм адаптивного круиз-контроля работает корректно, качество переходных процессов, частично представленных на рис. 6, говорит о том, что метод аппроксимации коэффициентов ПИД регулятора с помощью многослойного персептрона может быть применен для обеспечения свойства адаптивности алгоритма круиз-контроля.

Заключение

В рамках данной работы предложен алгоритм работы круиз-контроля грузового авто-

мобиля, адаптирующегося к текущим условиям движения: скорости и выбранной передаче. На основе известных методов синтеза ПИД регуляторов были получены коэффициенты регулятора для разных условий движения и обучена нейронная сеть для аппроксимации законов изменения данных коэффициентов. Корректность синтезированного алгоритма была проверена с помощью компьютерного моделирования.

Предложенный в статье подход может быть использован при решении и других прикладных задач, в которых объект управления является нелинейным и алгоритму управления требуется адаптироваться к меняющемуся состоянию объекта. Полученные результаты планируется использовать в дальнейшем для совершенствования грузовых автомобилей КАМАЗ.

Список литературы

1. **Ribbens W. B.** Understanding automotive electronics. Burlington: Elsiever Science, 2003. 470 p.

2. Vlacic L., Parent M., Harashima F. Intelligent Vehicle Tecnologies. Madras: Butterworth Heinemann, 2001. 498 p.

3. Александров А. Г., Паленов М. В. Состояние и перспективы развития адаптивных ПИД-регуляторов в технических системах // Тр. 3-й Всеросс. конф. с междунар. участием "Технические и программные средства систем управления, контроля и измерения". 2012. С. 1577—1587.

4. Васильев В.И., Ильясов Б. Г. Интеллектуальные системы управления. Теория и практика: учебное пособие. М.: Радиотехника, 2009. 392 с.

5. Волков В. Г., Демьянов Д. Н., Карабцев В. С. Разработка и исследование математической модели плоского движения автомобиля с полуприцепом // Математическое моделирование. 2017. Т. 29, № 7. С. 29—43.

6. Стуканов В. А. Основы теории автомобильных двигателей и автомобиля. М: ФОРУМ — ИНФРА-М, 2005. 367 с.

7. **Тарасик В. П.** Теория движения автомобиля: учебник для вузов. СПб.: БХВ-Петербург, 2006. 478 с.

8. Воронов А. А. Основы теории автоматического управления: Автоматическое регулирование непрерывных линейных систем. М.: Энергия, 1980. 312 с.

9. Гилл Ф., Мюррей У., Райт М. Практическая оптимизация. М.: Мир, 1985. 509 с.

Synthesis and Approximation of Control in Adaptive Cruise Control Systems of Commercial Vehicles

V. G. Volkov^{1,2}, vgvolkov93@mail.ru, D. N. Demyanov²,
 ¹Scientific Technical Centre of Kamaz PTC, Naberezhnye Chelny, Russia
 ²Kazan (Volga Region) Federal University, Kazan, Russia

Corresponding author: Volkov V. G., Design Engineer of Scientific Technical Centre of Kamaz PTC, Russia, graduate student of Kazan (Volga Region) Federal University, Kazan, Russia, e-mail: vgvolkov93@mail.ru

Abstract

In this paper, we consider the problem of the development of an algorithm of the adaptive cruise control functioning operating in the conditions of powertrain gear ratio varying in a wide range and vehicle velocity changing. The functioning of a classical cruise control system is generally based on the usage of a PID-controller with constant coefficients. However, despite the easiness of its tuning and physical realization and also its relatively high robustness this class of control devices cannot guarantee the cruise control system optimal functioning in all driving conditions because the plant is not time-invariant and linear. To overcome the above shortcomings, in this research we consider the possibility of neural network realization of a commercial vehicle adaptive cruise control algorithm.

In this paper, we propose the mathematical model of a commercial vehicle longitudinal motion designed for the control system analysis and synthesis. We carry out the PI-controller coefficients tuning to control the vehicle longitudinal velocity in various driving conditions of a commercial vehicle. We show that the controller coefficients vary according to a rather complex law. Therefore, we propose the algorithm of the adaptive cruise control functioning based on the approximation of the controller coefficients by the artificial neural network. The network used is the multilayer perceptron and it has ten neurons in the hidden layer to provide the high quality of the approximation. We carry out the training of the neural network by the Levenberg-Marquardt method with a sample of a total volume of 500 points, obtained using standard methods of controller synthesis. We verify the correctness of the obtained results through the computer simulations of the vehicle acceleration from 0 to 100 km/h, proving that the PI-controller coefficients, providing the required transient responses, significantly vary depending on the current state of the vehicle. The approach of the PI-controller coefficients approximation presented in this paper may be further used in the design of adaptive control systems able to function effectively in various operating modes.

Keywords: adaptive cruise control, neural networks, PID-controller, multilayer perceptron

For citation:

Volkov V. G., Demyanov D. N. Synthesis and Approximation of Control in Adaptive Cruise Control Systems of Commercial Vehicles, *Mekhatronika, Avtomatizatsiya, Upravlenie,* 2018, vol. 19, no. 11, pp. 707–713.

DOI: 10.17587/mau.19.707-713

References

1. **Ribbens W. B.** Understanding automotive electronics, Burlington, Elsiever Science, 2003, 470 p.

2. Vlacic L., Parent M., Harashima F. Intelligent Vehicle Tecnologies, Madras, Butterworth Heinemann, 2001, 498 p.

3. Aleksandrov A. G., Palenov M. V. Sostoyanie I perspektivy razvitiya adaptivnyh PID-regulyatorov v tehnicheskih sistemah (Actual state and perspectives of PID-controllers evolution in technical systems), Trudy 3 Vserossijskoj konferencii s mezhdunarodnym uchastiem "Tehnicheskiye i programmye sredstva system upravleniya, kontrolya I izmereniya", 2012, pp. 1577–1587 (in Russian). 4. Vasilyev V. I., Ilyasov B. G. Intellektual'nye sistemy upravleniya. Teoriya i praktika. (Intelligent Control Systems. Theory and practice), Moscow, Radiotehnika, 2009, 392 p. (in Russian).

 Volkov V. G., Demyanov D. N., Karabtsev V. S. Razrabotka i issledovaniye matematicheskoj modeli ploskogo dvizheniya avtomobilya s polupricepom (Development and research of the mathematical model of planar motion of a vehicle with a semitrailer), Matematicheskoe Modelirovaniye, Moscow, Nauka, 2017, vol. 29, no. 7, pp. 29–43 (in Russian).
 6. Stukanov V. A. Osnovy teorii avtomobil'nyh dvigatelej i av-

6. Stukanov V. A. Osnovy teorii avtomobil'nyh dvigatelej i avtomobilya (Basic theory of vehicle engines and vehicles), Moscow, FORUM – INFRA-M, 2005, 367 p. (in Russian).

7. **Tarasik V. P.** *Teoriya dvizheniya avtomobilya: uchebnik dlya vuzov* (Vehicle motion theory), SPb., BHV-Peterburg, 2006, 478 p. (in Russian).

 Voronov A. A. Osnovy teorii avtomaticheskogo upravleniya: Avtomaticheskoye regulirovaniye nepreryvnyh linejnyh system (Basics of Control Systems: Automatic regulation of continuous linear systems), Moscow, Energiya, 1980, 312 p. (in Russian).
 Gill F., Murray W., Right M. Prakticheskaya optimizaciya

9. **Gill F., Murray W., Right M.** Prakticheskaya optimizaciya (Practical Optimization), Moscow, Mir, 1985, 509 p. (in Russian).



17-19 июня 2019 года в Москве состоится

XIII Всероссийское совещание по проблемам управления,

посвященное 80-летию Института проблем управления имени В. А. Трапезникова РАН

Направления работы Совещания:

- Теория систем управления
- Управление подвижными объектами и навигация
- Интеллектуальные системы в управлении
- Управление в промышленности и логистике
- Управление системами междисциплинарной природы
- Средства измерения, вычислений и контроля в управлении
- Системный анализ и принятие решений в задачах управления
- Информационные технологии в управлении
- Проблемы образования в области управления: современное содержание и технологии обучения

Подробную информацию о Совещании смотрите на сайте http://vspu2019.ipu.ru

РОБОТЫ, МЕХАТРОНИКА И РОБОТОТЕХНИЧЕСКИЕ СИСТЕМЫ

УДК 621.865:004.896

DOI: 10.17587/mau.19.714-724

I. N. Ibrahim, Ph. D. Student, ibrncfe@gmail.com, http://orcid.org/0000-0001-9544-3020, Department of Mechatronics and Robotics, Kalashnikov Izhevsk State Technical University, Izhevsk, 426069, Russian Federation

A Comparative Study for an Inverse Kinematics Solution of an Aerial Manipulator Based on the Differential Evolution Method and the Modified Shuffled Frog-Leaping Algorithm

Accepted on July 16, 2018

Abstract

This paper focuses on the real-time kinematics solution of an aerial manipulator mounted on an aerial vehicle, the vehicle's motion isn't considered in this study. Robot kinematics using Denavit-Hartenberg model was presented. The fundamental scope of this paper is to obtain a global online solution of design configurations with a weighted specific objective function and imposed constraints are fulfilled. Acknowledging the forward kinematics equations of the manipulator; the trajectory planning issue is consequently assigned to on an optimization issue. Several types of computing methods are documented in the literature and are well-known for solving complicated nonlinear functions. Accordingly, this study suggests two kinds of artificial intelligent techniques which are regarded as search methods; they are differential evolution (DE) method and modified shuffled frog-leaping algorithm (MSFLA). These algorithms are constrained metaheuristic and population-based approaches. moreover, they are able to solve the inverse kinematics problem taking into account the mobile platform additionally avoiding singularities since it doesn't demand the inversion of a Jacobian matrix. Simulation results are carried out for trajectory planning of 6 degree-of-freedom (DOF) kinematically aerial manipulator and confirmed the feasibility and effectiveness of the supposed methods.

Keywords: Inverse Kinematics, Degree-of-Freedom (DOF), Human-like Aerial Manipulator, Optimization Algorithms, metaheuristic and Revolutionary Methods, differential evolution (DE), Shuffled Frog Leaping Algorithm

1. Introduction

The inverse kinematics (IK) solver is a primary problem in robotic manipulation, particularly when demand real-time and precision in calculations. Mathematically, the numerical solution of kinematics is intricate because of the high degree of nonlinearity, furthermore, the Linear and dynamic programming techniques usually fail or reach local optimum in solving NP-hard problems with a large number of variables and non-linear objective functions, moreover, Traditionally Jacobian-based solutions are identified to scale inadequately with the high number of degrees of freedom (DOF) [1] in addition to singularities existence. In contrast, [2] presented a comparative study of several methods based on the Jacobian matrix, clarifying that the modified Levenberg-Marquardt method is much better for a quite large set of random configurations than others but may lose convergence compared to Jacobian transpose and Pseudocode inverse methods. Recently many researchers [3] proposed a new method

for solving real-time IK without using the Jacobian matrix based on the position of end-effector (ee). using numerical and analytical mathematical tools but not mentioned exactly the performance as the time consuming to get the solution, in [4] also applied alike method for hyper-redundant manipulator arm. [5] combined two methods as a real-time IK solver for a human-like arm manipulator based on closed-form analytical equations for a given position while others [6] presented an on-line adaptive strategy based on the Lyapunov stability theory in addition to Radial Basis Function Network (RBFN) and quadratic programming which requires a complex hardware resources, the simulation was done for the position of ee in addition to avoid obstacles and was conducted on the 7-DOF PA-10 robot manipulator. In [7] a kinematic and time-optimal trajectory planning was considered for redundant robots, two approaches were presented, joint space decomposition and a numerical null-space method for a given pose. they were tested by 7-DOF industrial robots and demand high consuming time for resolving IK.

Metaheuristic optimization algorithms are an encouraging alternative approach to traditional IK techniques due to their strong performance on challenging and high-DOF problems in many various domains, the solution can be solved by minimizing an objective function allowing the end-effector to follow the desired path avoiding dynamics singularities and obstacles [8] was explained and proved that DE has emerged as one of the most powerful and versatile global numerical optimizers for nondifferential and multimodal problems, they showed challenges of the variants of DE which may provide less time and more robustness in solving IK. In [9] presented a quadratic programming with branching idea with a weighted multi-objective function which gave a short-time response about seconds while [10] showed a comparative research of four different heuristic optimization algorithms GA, PSO, QPSO and GSA for 4-DOF manipulator in order to reach the target as a position. they proved that the Quantum PSO is the best with average execution time of 1.65 seconds. In [11] investigated the performance of many PSO variants to resolve two DOF IK problem for a given position, they proved that PSO-VG is the fast variant compared with others which took a less average convergence iteration about 740 for 15 particles. [12] derived and minimized a fitness function to resolve the pose IK problem based on PSO for multiple DOF up to 180, they concluded that the runtime and iteration are 4.22 seconds and 118 respectively for a 9-DoF. In [13] a hybrid method called DEMPSO based on differential evolution (DE) and Modified PSO algorithms was developed in order to minimize the solution time for the pose. moreover presented a comparative study for several swarm intelligent optimization algorithms as ABC and ACO algorithms, based on their results, the DEMPSO had a great advantages at execution time for reaching the position while similar performance with DE for the orientation aim, the simulation was conducted with population size 30 for 10-DoF serial-parallel robot, furthermore, [14] a comparison of three evolutionary algorithms as GA, PSO, and DE was discussed. In [15] presented a comparative study of IK solver for a mobile manipulator using DE algorithm, they concluded that hybrid DE and biogeography-based optimization called HBBO provides good results but a higher computational cost for weighted fitness function and pose target, in contrast, DE proved to be superior to PSO, CS, and TLBO, additionally the PSO algorithm verified that it does not solve the inverse kinematic problems correctly. In [16] a developed methodology

was applied to synthesize of six-bar mechanism, it used DE with geometric centroid of precision positions technique (GCCP). [17] used DE to improve the design of a fuzzy controller for a wall-following hexapod robot. In addition [18] proposed a modified self-adaptive DE in order to improve the static force of humanoids robot, showing robust, safe, reliable performance compared with other metaheuristics. While [19] presented an approximation tool for the inverse model of the industrial robot based on an adaptive neural model optimized by advance DE. [20] proposed an optimal joint trajectory planning method using forward kinematics of 7-DoF freefloating space robot based on DE method, also they depict the general aspect of equality and inequality constraints which govern each joint in the manipulator. On another-side, [21] introduced an algorithm shuffled frog-leaping algorithm SFLA which is a population-based collaborative search metaphor inspired by natural memetics, it relies on a knowledge called 'meme' which causes someone to replicate it or to repeat it to someone else, spreading from brain to brain. All transmitted knowledge is memetic and spreading is much faster than a gene, the effectiveness, suitability, and global optimal resolving have been demonstrated in addition to short processing time. Additionally, [22] proposed an MSFLA for a high dimensional continuous function optimization. this method yields a strong robustness and best convergence also presented a comparative study for PSO, SFLA, MSFLA, and MSFLA-EO which designated that MSFLA is better than others. In [23] a modified SFLA was assumed for obtaining the optimum preventive maintenance scheduling of generating units in power system. While [24] presented a comparative study among five evolutionary-based optimization algorithms as GA, MA, PSW, ACO, and SFLA, they showed the processing time for solving the F8 function and concluded that the SFLA is the best.

In this work, we applied two types of metaheuristic algorithms in order to solve the inverse kinematics of a mobile manipulator [25] as a constrained optimization problem the proposed algorithms are the differential evolution (DE) method and modified shuffled frog-leaping algorithm (MS-FLA), which characterized as an accurate and fast convergence in discovering the solution based on the previous literature study. Initially, we define an objective function to minimize the error between the desired and the actual end-effector pose. The objective function takes into account the minimal movement between the previous and the actual joint configurations. To overcome the constrained problems, we use a penalty function to penalize all those manipulator configurations that violate the allowed joint boundary. Hence, the proposed approach estimates the feasible manipulator configuration needed to reach the desired end-effector pose. The remainder of this paper is organized as follows: Section 2 and 3 present the architecture and kinematics of a robotic manipulator. Section 4 introduces metaheuristic optimization algorithms and the weighted objective function. Section 5 shows the simulation results of the proposed trajectory planning methods applied to a 6-DOF kinematically manipulator. The conclusive observations are listed in the last section.

2. The architecture of the aerial manipulator

The aerial manipulator was designated in [25] as a human-like arm and consists of 20 degrees of freedom. It was proposed that each joint has one DOF with revolute type while the links are a type of rigid-body which organized from the base-frame located in the center of aerial vehicle's gravity with a constant displacement based on a design consideration for the mass distribution. While the other main-frame called hand-frame located in the center of the hand part 7th frame (in the Fig. 1 has a number 7, see the 3rd side of cover) that was considered as a reference node for combining and moving the fingers in order to perform specific object-oriented tasks [25]. Furthermore, the realized prototype of the manipulator as shown in Fig. 1 was divided into two parts are arm-part and hand-part, the arm-part architecture consists of 6 joints originated from shoulder joints (J_1, J_2) to the forearm joint (J_3) and finally, wrist joints (J_4, J_5, J_6) along to center hand's frame. The specifications of the arm-part are given in the Table 1 which consists of six joints and four 3d-printed links in addition to two metal links as metal horns and brackets, besides, the scheme of this part is designated to be anti-resistance of the air friction in design as possible as shown in the figure. the last link is the hand-part which consists of five fingers driven by five micro-servo motors based on linkage mechanism out of scope in this paper. the total number of degrees-of-freedom for the intended arm-part is six.

Table 1 Specification of the Arm-Part

l _i	Between J_i and J_{i+1}	Length (cm)
l_0	link (Between J_0 and J_1)	3.5
l_1	link (Between J_1 and J_2)	6.5
l_2	link (Between J_2 and J_3)	28.2
l_3	link (Between J_3 and J_4)	21
l_4	link (Between J_4 and J_5)	2.5
l_5	link (Between J_5 and J_6)	7.5
l_6	link (Between J_6 and J_7)	5

The navigation process of the arm-part is performed by controlling the motion of hand-part located in the 7th frame in the workspace related to the base frame. Furthermore, this process is computed by analyzing the forward and inverse kinematics. the goal of this paper is to study the solutions of the inverse kinematics in order to realize this movement precisely within a short time. Hence, each of links was described by some of the properties as material type, stiffness, toughness, bearing gear, shape, weight, inertia, lubrication in addition to aerodynamic parameters. The values of motion range for the joints are manifested in the following Table 2 which readjusted to be more fitting for accomplishing more tasks compared to real joints of the humanlike arm [25].

3. Manipulator Kinematics

Now, in order to determine the relationship between the coordinate frames, that are assigned to the links and joints of the robot, homogeneous transformations are required. Three parameters are employed to describe the rotation while other three parameters are used to define the translation. accordingly, the Denavit Hartenberg (DH) convention was used to describe kinematically the rigid motion by assigning the values of four quantities for each link. Two describe the link itself, and two describe the link's connection to a neighboring link. Where θ , *a*, *d* and α are the joint angle, link length, link offset and link twist between joints. Considering T_i is the homogeneous transformation matrix between the frames that is a function of θ while the other three parameters are constant. The data in Table 2

The	motion	range	for	the	joints	of	the	Arm-Part
-----	--------	-------	-----	-----	--------	----	-----	----------

Arm-part	angle	θ_1	θ2	θ_3	θ_4	θ_5	θ_6
	range	$-90 \rightarrow +90$	$0 \rightarrow +180$	$-90 \rightarrow +90$	$-90 \rightarrow +90$	$-90 \rightarrow +90$	$-70 \rightarrow +90$

Link parameters of the arm-part of manipulator

Modified Denavit Hartenberg					Standard Denavit Hartenberg					
α_{i-1}	α_{i-1} [cm]	<i>d_i</i> [cm]	θ _i	Initial Value of θ_i	α _i	<i>a_i</i> [cm]	<i>d_i</i> [cm]	Θ_i	Initial Value of θ_i	Joint Offset
$-\pi/2$	l ₀	0	θ1	π/2	$-\pi/2$	6.4	0	θ1	0	0
π/2		0	θ2	$-\pi/2$	0	30.2	0	θ2	$-\pi/2$	$-\pi/2$
0	l_2	0	θ3	$-\pi/2$	π/2	0	0	θ ₃	π/2	π/2
$-\pi/2$	0	$l_3 + l_4$	θ ₄	0	π/2	0	23.5	θ_4	0	0
π/2	0	0	θ_5	$-\pi/2$	$-\pi/2$	5.3	0	θ5	π/2	π/2
$-\pi/2$	<i>l</i> ₅	0	θ ₆	0	0	5.6	-2	θ ₆	0	0

the Table 3 present link parameters of the arm-part based on DH strategy in two formulas: standard and modified DH. Since the standard form was used in simulation by LabVIEW Robotics module in order to validate the design. The position of all links of an arm-part manipulator can be specified with a set of 6 joint variables from the shoulder's joints till wrist's joints. This set of variables is often referred to as a 6×1 joint vector [25].

The space of all joint variables is referred to as joint-space $\Theta = [\theta_1, \theta_2, ..., \theta_6]^T$, here we have been concerned with computing the Cartesian space representation from knowledge of the joint-space information. hence the homogeneous transformations of links are as following. these transformations, ${}^{i-1}_{i}T$, will be a function of all joint variables. If the robot's joint-position sensors are estimated by servo mechanisms, the Cartesian position and orientation of the hand-part can be computed by ${}^{0}_{7}T$ [25].

$${}^{0}_{1}T = \begin{bmatrix} C_{1} & -S_{1} & 0 & l_{0} \\ 0 & 0 & 1 & 0 \\ -S_{1} & -C_{1} & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, {}^{1}_{2}T = \begin{bmatrix} C_{2} & -S_{2} & 0 & l_{1} \\ 0 & 0 & -1 & 0 \\ S_{2} & C_{2} & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

$${}^{2}_{3}T = \begin{bmatrix} C_{3} & -S_{3} & 0 & l_{2} \\ S_{3} & C_{3} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, {}^{3}_{4}T = \begin{bmatrix} C_{4} & -S_{4} & 0 & 0 \\ 0 & 0 & 1 & l_{3} + l_{4} \\ -S_{4} & -C_{4} & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

$${}^{4}_{5}T = \begin{bmatrix} C_{5} & -S_{5} & 0 & 0 \\ 0 & 0 & -1 & 0 \\ S_{5} & C_{5} & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, {}^{5}_{6}T = \begin{bmatrix} C_{6} & -S_{6} & 0 & l_{5} \\ 0 & 0 & 1 & 0 \\ -S_{6} & -C_{6} & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

$${}^{6}_{7}T = \begin{bmatrix} 1 & 0 & 0 & l_{6} \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

Where the computations of the transformation as follows:

$${}^{0}_{7}T = \begin{bmatrix} b_{11} & b_{12} & b_{13} & b_{11}l_6 \\ b_{21} & b_{22} & b_{23} & b_{21}l_6 \\ b_{31} & b_{32} & b_{33} & b_{31}l_6 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\begin{split} b_{11} &= [(C\theta_1 C\theta_{23} C\theta_4 - S\theta_1 S\theta_4) C\theta_5 - C\theta_1 S\theta_{23} S\theta_5] C\theta_6 + \\ &+ (-C\theta_1 C\theta_{23} C\theta_4 S\theta_5 + S\theta_1 S\theta_4 S\theta_5 - S\theta_5 S\theta_1 S\theta_{23}) S\theta_6; \\ b_{12} &= -[(C\theta_1 C\theta_{23} C\theta_4 - S\theta_1 S\theta_4) C\theta_5 - C\theta_1 S\theta_{23} S\theta_5] S\theta_6 + \\ &+ (-C\theta_1 C\theta_{23} S\theta_4 + S\theta_1 S\theta_4) C\theta_6; \\ b_{13} &= -(C\theta_1 C\theta_{23} C\theta_4 - S\theta_1 S\theta_4) S\theta_5 - C\theta_1 S\theta_{23} S\theta_5; \\ b_{21} &= (C\theta_1 S\theta_{23} C\theta_4 + C\theta_{23} S\theta_5) C\theta_6 - S\theta_{23} S\theta_4 S\theta_6; \\ b_{22} &= -(S\theta_{23} C\theta_4 C\theta_5 + C\theta_{23} S\theta_5) S\theta_6 - S\theta_{23} S\theta_4 C\theta_6; \\ b_{31} &= [(S\theta_1 C\theta_{23} - C\theta_1 S\theta_4) C\theta_5 + S\theta_1 S\theta_{23} S\theta_5] C\theta_6 + \\ &+ (S\theta_1 C\theta_{23} S\theta_4 + C\theta_1 C\theta_4) S\theta_6; \\ b_{32} &= -[(S\theta_1 C\theta_{23} - C\theta_1 S\theta_4) C\theta_5 + S\theta_1 S\theta_{23} S\theta_5] S\theta_6 - \\ &- (S\theta_1 C\theta_{23} S\theta_4 + C\theta_1 C\theta_4) C\theta_6; \\ b_{33} &= (S\theta_1 C\theta_{23} - C\theta_1 S\theta_4) S\theta_5 + S\theta_1 S\theta_{23} C\theta_5. \end{split}$$

4. Proposed Optimization Techniques or solving kinematics

The evolutionary optimization algorithms can solve the complicated nonlinear equations completely and efficiently. The solution of the inverse kinematics for the manipulator results is a very difficult problem to solve by traditional approaches. Besides, the suggested strategies do not require the inversion of any Jacobian matrix, and then it avoids singularities configurations. In this paper, two algorithms used to optimize this problem are differential evolution and modified shuffled frog-leaping algorithm. In general, this solution is based on the forward kinematics equations which always produces a solution in addition to the objective function. Hence, the general aspect of the problem may express as minimizing $J(\Theta)$, constrained by $\Theta_{\min} \leq \Theta \leq \Theta_{\max}$, furthermore, the objective function could be defined as the weighted sum of the errors as follows

$$J(\Theta) = \sigma P_{error}(\Theta) + \beta O_{error}(\Theta) = = \sigma \|P_G - P_E(\Theta)\| + \varepsilon \|O_G - O_E(\Theta)\|,$$
(1)

where $P_{error}(\Theta)$ and $O_{error}(\Theta)$ represent the position and orientation errors respectively and could be computed as a difference in distance between the target and current position, in this work we used Euclidean formula as a representation of distance. While the parameters σ and ε are the weights of the position and the orientation, respectively. Let $G = (P_G, O_G)$ be a given target end-effector pose while $E(\Theta) = (P_E(\Theta), O_E(\Theta))$ is the current endeffector pose in the workspace corresponding to configuration $\Theta = [\theta_1, \theta_2 \dots \theta_6]^T$ which can be calculated using forward kinematics — where P refers to the 3D position vector of pose while O refers to vector of Roll-Pitch-Yaw Euler Angles of pose (in radians), respectively, that optimization algorithms are exploring directly in the configuration space of the manipulator. Hence, each individual $\Theta_i = [\Theta_{i,1}, \Theta_{i,2}...\Theta_{i,j}...\Theta_{i,6}]^T$ represents an *i*-th candidate set of joint angles. henceforward, at each iteration, we evaluate each candidate configuration Θ_i by passing it through the forward kinematics model and measuring the position and orientation error between where the end-effector would be at configuration Θ_i and the target end-effector pose. In order to enforce joint limits, each dimension *j* of element Θ_i should be limited to searching in the range of valid joint angles $\Theta_i \in [\Theta_{\min}, \Theta_{\max}]$. This can be realized by clamping each dimension *j* within these bounds at each iteration immediately after it is updated.

4.1. Differential Evolution Algorithm

The first study on DE algorithm was introduced by Storn and Price [8], [18], [26]. It is one of the most powerful stochastic population-based optimization algorithms. It was invented to optimize functions in an n-dimensional continuous domain. moreover, it occupies several benefits such as simple implementation, good performance, global optimization, robust, low space complexity, converges fast, and has a good balance between exploration and exploitation. In particular, DE is relevant to stan-

dard evolutionary algorithms in which a population of candidate solutions, initialized to a uniform sampling of the instance space, are continuously enhanced by periodically adding a scaled variant of the difference vector to a third individual to generate a new candidate solution and then producing the succeeding generation. DE consists of four stages: initialization, mutation, crossover, and selection. The last three of these are iterated until a termination condition such as the maximum number of generations is reached. Nevertheless, unlike other evolutionary algorithms before-mentioned as evolution strategies, mutation is performed by applying the scaled difference between members of the population. This has the impact of adjusting the step size to the fitness aspect over time. The implementation of this method is illustrated in Algorithm 1.

Algorithm 1: The pseudo-code of the differential evolution algorithm

Initialization: $Population^{(1)} \leftarrow \{\Theta_{1}^{(1)}, \Theta_{2}^{(1)}, ..., \Theta_{i}^{(1)}, ..., \Theta_{NP}^{(1)}\}, g \leftarrow 1, g_{max}$ **Evolution Process:** While Termination criteria not met do for $i \leftarrow 1, NP$ do Mutation Process: $v_i^{(g)} \leftarrow mutate(\Theta_i^{(g)})$ Crossover Process: $u_i^{(g)} \leftarrow crossover(\Theta_i^{(g)}, v_i^{(g)})$ Selection Process: $f(u_i^{(g)}) \leq f(\Theta_i^{(g)})$ then insert $u_i^{(g)}$ into population^(g+1) if else insert $\Theta_i^{(g)}$ into population^(g+1) end if end for $g \leftarrow g + 1$ end while

The trajectory planning strategy can be transformed into an optimization issue with multiple constraints. Firstly, it demands to determine the dimension of the population NP, the generation number g with maximum g_{max} , the dimension real-valued of the individual is equal to the configuration space of the manipulator, the scale factor F, and the crossover factor C_r . Then Individuals in the population are expressed by: $\Theta_i^{(g)} = (\theta_{i,1}^{(g)}, \theta_{i,2}^{(g)}, \dots, \theta_{i,6}^{(g)});$ i = 1, 2, ..., NP, represents the design variable of the *i*-th individual in generation g DE begins by initializing a population of NP to cover as much as possible of the exploration space constrained by the minimum and maximum bounds $\Theta_{\min} =$ $\left[\theta_{\min,1}, \theta_{\min,2}, \dots, \theta_{\min,i}, \dots, \theta_{\min,6}\right]_{T}^{T}$ and $\Theta_{\max} =$ = $[\theta_{\max,1}, \theta_{\max,2}, ..., \theta_{\max,i}, ..., \theta_{\max,6}]^T$. Hence, the *i*-th individual may then be initialized as: $\theta_{i,j}^{(1)}$ = $= \theta_{\min,j} + H(0,1)[\theta_{\max,j} - \theta_{\min,j}], \text{ with } H = rand(0,1)$

being a uniformly random value between 0 and 1. Henceforward, the mutant strategy is adopted after initialization to generate a donor vector $v_i^{(g)} = (v_{i,1}^{(g)}, v_{i,2}^{(g)}, \dots, v_{i,6}^{(g)})$ by its corresponding target vector $\Theta_i^{(g)}$ [8, 20] have been proposed:

$$\begin{aligned} & \text{DE/rand/1:} \\ & \nu_i^{(g)} = \Theta_{r_1}^{(g)} + F_i(\Theta_{r_2}^{(g)} - \Theta_{r_3}^{(g)}) \\ & \text{DE/best/2:} \\ & \nu_i^{(g)} = \Theta_{best}^{(g)} + F_i(\Theta_{r_1}^{(g)} - \Theta_{r_2}^{(g)}) + F_i(\Theta_{r_3}^{(g)} - \Theta_{r_4}^{(g)}) \\ & \text{DE/current-to-best/1:} \\ & \nu_i^{(g)} = \Theta_i^{(g)} + F_i(\Theta_{best}^{(g)} - \Theta_i^{(g)}) + F_i(\Theta_{r_1}^{(g)} - \Theta_{r_2}^{(g)}) \end{aligned}$$

either-or: this strategy merges two methods to generate the donor vector [26].

 $p_{f} \leftarrow mutation \ probability \in [0, 1],$ $a \leftarrow random \ number \in [0, 1]$ $if \ a < p_{f} \ then$ $use \ DE/rand/1: \ v_{i}^{(g)} = \Theta_{r_{1}}^{(g)} + F_{i}(\Theta_{r_{2}}^{(g)} - \Theta_{r_{3}}^{(g)})$ else $use \ DE/rand/2:$ $v_{i}^{(g)} = \Theta_{r_{1}}^{(g)} + K(\Theta_{r_{2}}^{(g)} - \Theta_{r_{1}}^{(g)} + \Theta_{r_{3}}^{(g)} - \Theta_{r_{1}}^{(g)})$ $end \ if$

Where, F_i is the scaling factor within 0 and 1, indices r_1 , r_2 , r_3 and r_4 are randomly selected integers from the range [1, NP], such that $r_1 \neq r_2 \neq r_3 \neq i$. $\Theta_{best}^{(g)}$ is the best individual in the current population, also p_f and a are the mutation probability and random number, respectively. At that point, a crossover between $v_i^{(g)}$ and $\Theta_i^{(g)}$ is performed to generate a trial vector $u_i^{(g)} = (\mu_{i,1}^{(g)}, \mu_{i,2}^{(g)}, \dots, \mu_{i,6}^{(g)})$, two methods were used in this paper, A binomial and Exponential crossover procedures [8]. The binomial crossover provides a trial vector by selecting an element from the donor vector whenever a randomly produced value formed from a uniform distribution is below the crossover rate C_r . Additionally, an element h is randomly taken per iteration to always come from a donor vector as follows:

$$\mu_{i,j}^{(g)} = \begin{cases} \upsilon_{i,j}^{(g)} & \text{if } i = h \text{ or } rand(0,1) \leq C_r, \\ \theta_{i,j}^{(g)} & \text{otherwise.} \end{cases}$$

Exponential crossover tries to exploit relationships between adjacent elements. It works by choosing a random starting element and selecting the next L consecutive elements in a circular manner from the donor vector. The number of elements L is calculated as follows:

Algorithm 2: Exponential crossover

 $\begin{array}{l} L \leftarrow 0 \\ repeat \\ L \leftarrow 0 \\ until \ rand(0,1) > C_r \ or \ L > D \end{array}$

After crossover, the objective function as explained in Eq. 1 is evaluated for the trial vector $u_i^{(g)}$. According to the greedy selection only, as shown in algorithm 1. Afterward, the better of $u_i^{(g)}$ and $\Theta_i^{(g)}$ will be picked to remain into the next generation.

4.2. Modified Shuffled Frog-Leaping Algorithm

The shuffled frog-leaping algorithm (SFLA) was developed by Eusuff and Lansey in 2003 [21]. It is a member of the Memetic algorithm family, a particular kind of a meta-heuristic optimization approach and a type of evolutionary algorithms which based on population. It is inspired by the memetic evolution of frogs exploring food in a lake, which consolidates the benefits of the genetic-based MA and the social behavior-based particle swarm optimization [21]. In general, the SFLA incorporates two alternating processes: a local exploration in the sub-memeplex and global information exchange among all memeplexes.

The optimization achievement of the SFLA basically relies on two facts, the first one is the evolution process on each memeplex which embraces different cultures of frogs, this culture stimulates fitness values and the evolution process serves as a local search within memeplex analogous to PSO algorithm which imitates the social behavior of the leaping action of frogs searching for food. Additionally, the extra fact is an idea held within each frog which can be influenced by the ideas of other frogs from other memeplexes throughout the shuffling rule, this animates the cooperation process which it implies an adaptation idea and improves the success rate of discovering the solution in optimization puzzle. In this process, a modification was applied to the frog-leaping action that it enhances the exploration manner in the space [22], [23]. Moreover, the randomization strategy in the evolution process proffers the algorithm the ability to discover the local best solution within search space stochastically in addition to the communication process that possibly finds a global optimum solution in shorter time. The local search and the shuffling processes continue until defined convergence criteria are satisfied. The pseudocode of the algorithm was presented in Algorithm 3.

Algorithm 3: The pseudo-code of the Shuffled Frog-Leaping Algorithm

Initialization:

Population $\leftarrow \{\Theta_1, \Theta_2, ..., \Theta_i, ..., \Theta_{NP}\};$ m \leftarrow number of memeplexes;

 $n \leftarrow number of memorphexes,$ $n \leftarrow quantity of frogs in each memoplex;$

 $l \leftarrow 1, iN$

while (convergence criteria is satisfied Or ubtil met iN) do

Rank Step: Evaluate each frog Θ_i using a fitness function;

Partition Step:

Construct an array U of frogs and their fitness's values; Sort the array U in descending order based on the fitness column;

Construct $(Y^k; k = 1, ..., m)$ memeplexes each including *n* frogs;

Evaluation Step:

for $l \leftarrow 1$, iM do

for $k \leftarrow 1$, m do

Determine the worst and best frogs based on their fitness's values;

Improve the worst frog position using a leaping distance;

end for end for

Shuffle Memeplexes Step: combine the evolved memeplexes;

Check Convergence: Update the population best frog's position Θ_g ;

 $l \leftarrow l + 1;$ end while

The MSFLA meta-heuristic strategy is summarized in the following steps:

a. Initialization step, construct the population *NP* of frogs randomly similar to the first step in DE algorithm, then Select *m*, and *n*, where *m* is the number of memeplexes and *n* is the number of frogs in each memeplex. Therefore, the total amount of frogs *NP* can be calculated as NP = mn, additionally, the *i*-th frog is expressed as a vector with the dimension is equal to the configuration space as follows $\Theta_i = (\theta_{i,1}, \theta_{i,2}, ..., \theta_{i,6}); i = 1, 2, ..., NP.$

b. Rank step, compute the performance value f_i for each frog Θ_i . Sort the *NP* frogs in a descending order according to their fitness. Save them in an array $U = \{f_i, \Theta_i; i = 1, 2, ..., NP\}$, so that i = 1 denotes the frog with the best performance value and could save it as a Θ_g in each iteration while algorithm running.

c. Partition Step, partition array U into m memeplexes $Y_1, Y_2, ..., Y_m$, each including n frogs, such that $Y^k = [\Theta_i^k, f_i^k | \Theta_i^k = \Theta_{(k+m(i-1))}, f_i^k = f_{(k+m(i-1))}, i = 1, ..., n]; k = 1, ..., m$. In this process, the first frog goes to the first memeplex, the second frog goes to the second memeplex, frog *m* goes to the *m*-th memeplex, and frog m + 1 goes back to the first memeplex, etc.

d. Memetic Evaluation step, evolve each memeplex Y^k ; k = 1, ..., m according to the frog-leaping algorithm as follow. Within each memeplex, the frogs with the best and the worst fitness values are defined as Θ_b and Θ_w , respectively. Furthermore, the frog with the global best fitness is defined as Θ_{o} . Next, an improvement process is applied to only the frog with the worst fitness in each cycle. Hence, the position of the frog with the worst fitness is modified which emulates the leaping process as follows: leaping distance $D = C_L rand(0, 1)[\Theta_b - \Theta_w]$, then new position $\Theta_w = \Theta_w + D$; $D \in]-D_{\max}$, $D_{\max}[$. Where, rand(0, 1) is a random number between 0 and 1, D_{max} is the maximum allowed change in a frog's position and C_L is the modification of the algorithm which it is a constant indicates the amount of frog-leaping in each memeplex. The evaluation process for all memeplexes is repeated by an adaptable number of iterations called iM until no improvement becomes possible.

e. Shuffle Memeplexes Step, shuffle frogs and replace all memeplexes Y^k ; k = 1, ..., m into U, such that $U = \{f_i, \Theta_i; i = 1, 2, ..., NP\}$ similar to initialization phase, afterward sort U in order of decreasing performance value, Update the population best frog's position Θ_g .

f. Check the convergence criteria if satisfied then stop otherwise return to the partition step and continue for a specific quantity of iterations which called iN, finally after each iteration the first frog in the sorted list represents a global solution. The number of iteration iM specifies the depth of search within memeplexes while the iN governs the solution producing process.

5. Simulation Results and Discussions

In this work, we solve inverse kinematics of the redundant manipulator with six joints to follow a destination pose. The manipulator's joints correspond to the variable θ_j , J = 1, 2, ..., 6 had been constrained by a Table 2. The DH table is presented in Table 3. In the inverse kinematics experiments, the desired end-effector pose for the arm-part of the manipulator was determined as a variable $G = (P_G, O_G) = (x, y, z, roll, pitch, yaw) = (-20, 3, 40, 0, 10, 15)$. Moreover, the

	Table 4							
Setting of the DE Algorithm								
Mutation Method	Random							
Scale Factor	0.9							
Crossover Method	Uniform							
Crossover Probability	0.95							

parameters of the objective function were adjusted as follows $\varepsilon = 1 - \beta = 0.7$ so there is a balance between position and orientation to be optimized. In case of DE algorithm, Table 4 shows DE settings while Table 5 presents the results of utilizing DE for some scenarios. As presented in Table 6, it is obvious that the execution time depends on the size of the population and the iterations, respectively. Further, the population size achieves the diversity feature which let the algorithm explores more solutions in the workspace while the high iteration gives a solution much closer to the target. Hereafter, a Fig. 2 presents the values of the objective function, while the Fig. 4 and Fig. 5 illustrate the position and orientation of end-effector for the manipulator after applying the solutions to validate IK solver. Forthwith, the Fig. 3 (see the 3rd side of cover) demonstrates the configuration joints for reaching the setpoint G, which shows multiple solutions after each new iteration be-

Table 5

Inverse Kinematics Results of Differential Evolution Algorithm



Fig. 2. Displays the values of the objective function after applying IK-DE solver

Table 6

Setting of the MSFL Algorithm

т	Number of memeplexes	3
n	Number of frogs within memeplexes	NP/m
C_L	Amount of Leaping	1.3

cause of the redundant nature of our manipulator, in addition, each new solution is considered as a global solution within its iteration, new high iteration grants algorithm an ability to explore a new global solution. therefore, it's important to alter the settings of DE to conclude one solution based on the objective function in shorter time.

By analyzing the results in the Table 5 and Fig. 3, it was explicit that the test number 9 is the proper solution with a convergence time equal to 835 ms and a total error equal to -0.001729 taking into consideration that the adaptation of DE parameters nearby setting of this result may improve the solution to be more fitting but longer convergence time.

In case of MSFLA, the parameters of the algorithm were introduced in the Table 6 additionally, the summary of the results of utilizing the algorithm for multiple scenarios was represented in the Table 7.

As presented in the results, the amounts of execution time are longer than those in the experimental results of DE algorithm. The population of the DE algorithm is created randomly and the size of the population should be selected based on the problem and dimension of the workspace in order to cover all the space and consequently obtain the global solution quickly. Besides, the maximum number of iteration depends on the accuracy required on the problem.

Table 7

Inverse Kinematics Results of MSFL Algorithm

Tests Population	Demolation	Itera	tions	I(O)	Total Eases	Execution	Desching Transfer or a still witch start	
	Population	iN	iМ	J(B)	lotal Error	Time [ms]	Reaching Target (x, y, z, roll, pitch, yaw)	
1	20	30	10	11.618	29.7156	729	(-15.7365, 5.43, 52.57, 6.63, 12.66, 16.164)	
2	30	30	10	7.6614	12.0878	1045	(-21.1832, 2.91528, 50.77, -0.201742.10.0158, 14.8583)	
3	40	30	15	10.5382	19.2097	1685	(-25.08, 8.56, 46.818, -6.2, 9.6, 5.4251)	
4	40	40	30	18.4625	18.4625	4526	(-25.2365, 8.34134, 47.591, -2.537.8.6258.14.1301)	
5	60	40	30	8.2925	8.2925	6645	(-24.4625, 0.04208, 44.589, 1.655, 11.1161, 14.054)	
6	80	50	40	11.0236	11.0236	13 540	(-26.9977, 3.59399, 42.871, -0.0686, 9.816, 15.6762)	
7	100	60	60	29.7744	29.7744	24 191	(-20.0331, 30.03867, 39.9706, -7.709, 3.6786, -0.7192)	
8	130	70	60	0.151062	0.648529	46 282	(-20.09, 2.988, 40.004, 0.20727, 9.642, 14.894)	
9	170	60	50	0.616846	2.16791	40 459	(-20.1505, 2.8357, 40.0911, 0.8872, 10.3554, 16.149)	
10	200	90	40	0.113833	0.297961	57 362	(-19.9268, 3.00718, 39.98, -0.1344, 10.1049, 14.894)	
11	200	100	60	0.0728863	0.378871	92 779	(-20.0018, 2.99737, 39.9906, -0.13745, 10.1506, 14.9178)	
12	200	120	80	2.76717	5.81648	150 246	(-20.4796, 4.15287, 40.4894, 0.80744, 4.08227, 13.7867)	
13	200	200	100	2.67129	1.93396	318 481	(-19.2649, 2.23482, 41.9365, -0.978994, 10.8776, 11.4859)	
14	250	90	40	0.003134	0.0165418	69 818	(-19.9995, 3.00023, 39.9998, 0.00495, 10.0061, 14.995)	
15	250	140	80	1.26635	6.55357	215 027	(-20, 3, 10, -7.01976e-10, 10, 15)	
16	250	140	100	4.64762e-9	1.05135e-8	260 325	(-20.0982, 2.9671, 40.0008, -1.68742, 6.68997, 13.5741)	
17	300	140	80	1.01476e-9	3.3374e-9	255 989	(-20, 3, 40, 1.16487e-9, 10, 15)	
18	500	90	40	5.4912e-10	9.9601e-10	136 888	(-20, 3, 40, -1.66261e-11, 10, 15)	
19	500	200	100	3.02894e-15	1.5664e-14	68 1646	(-20, 3, 40, 3.22962e-15, 10, 15)	
20	1000	30	45	0.0968137	0.0255	95197	(-20.0305, 3.0454, 40.0451, -0.0037, 10.957, 14.8752)	

Fig. 6. Displays the values of the objective function after applying IK-MSFLA solver

Hereafter, the Fig. 6 displays the values of the objective function, while the Fig. 7 and Fig. 8 (see the 3^{rd} side of cover) represent the position and orientation of end-effector for the manipulator after applying the solutions to validate IK solver. Forthwith, the Fig. 9 (see the 3^{rd} side of cover) shows the configuration joints for reaching the setpoint G.

6. Conclusion

The design and kinematics of a human-size aerial manipulation robot have been introduced. this design has been separated into two parts: arm-part and hand part in order to be easier in the navigation process and control system. the purpose of this study is to observe a perception of actuating the manipulator so fast; to reach a target point in the 3D workspace, robotically investigate a kinematics solution. therefore, in this paper, we have presented a methodology to solve the inverse kinematics of a 6-DOF lightweight manipulator based on evolutionary algorithms. Furthermore, the metaheuristic algorithms included in the simulations were the differential evolution (DE) and modified shuffled frogleaping (MSFL). These algorithms were designated as multi-objective, multi-dimensional, stochastic and diversity methods. The simulation and experiments were conducted to prove the effectiveness of each approach. Consequently, the DE had the best performance over SFLA with the high success rate and the fast convergence as well, the simulation was

carried out through PC with following specifications (Core i7, Q740, 1.73GHz). Subsequently, all the optimization techniques of population-based algorithms depend on the concept of generating a population randomly then relocate or migrate individuals in the space to obtain the local/global solution. besides, it is not arduous to promote those two algorithms and proffer them an adaptive feature as we fulfilled with MSFLA in this research to deliver more stabilization plus little run-time. Finally, the aim of this study was acknowledged as one from multiple objectives of our project, which are controlling the aerial manipulator integrated into an aerial vehicle, in the space additionally, studying of the disturbances effects on the vehicle and stabilizing the motion of the manipulator.

References

1. **Buss S. R.** (2004). Introduction to inverse kinematics with jacobian transpose, pseudoinverse and damped least squares methods. *IEEE Journal of Robotics and Automation*, 17 (1-19), 16.

2. Dulęba I., & Opałka M. (2013). A comparison of Jacobianbased methods of inverse kinematics for serial robot manipulators, *International Journal of Applied Mathematics and Computer Science*, 23(2), 373–382.

3. Wang X., Zhang D., Zhao C. (2017). The inverse kinematics of a 7R 6-degree-of-freedom robot with non-spherical wrist, *Advances in Mechanical Engineering*, 9(8), 1687814017714985.

4. Ananthanarayanan H., & Ordóñez R. (2015). Real-time Inverse Kinematics of (2n + 1) DOF hyper-redundant manipulator arm via a combined numerical and analytical approach, *Mechanism and Machine Theory*, *91*, 209–226.

5. Tolani D., Badler N. I. (1996). Real-time inverse kinematics of the human arm, *Presence: Teleoperators & Virtual Environments*, 5(4), 393–401.

6. Toshani H., & Farrokhi M. (2014). Real-time inverse kinematics of redundant manipulators using neural networks and quadratic programming: a Lyapunov-based approach, *Robotics and Autonomous Systems*, 62(6), 766–781.

7. Reiter A., Müller A., Gattringer H. (2016, October). Inverse kinematics in minimum-time trajectory planning for kinematically redundant manipulators, In *Industrial Electronics Society, IECON 2016 – 42nd Annual Conference of the IEEE* (pp. 6873–6878). IEEE.

8. Geitle M. (2017). Improving differential evolution using inductive programming (Master's thesis).

9. Bodily D. M., Allen T. F., Killpack M. D. (2017, May). Motion planning for mobile robots using inverse kinematics branching, *In Robotics and Automation (ICRA), 2017 IEEE International Conference on* (pp. 5043–5050). IEEE.

10. Ayyıldız M., Çetinkaya K. (2016). Comparison of four different heuristic optimization algorithms for the inverse kinematics solution of a real 4-DOF serial robot manipulator, *Neural Computing and Applications*, 27(4), 825–836.

11. Rokbani N., Alimi A. M. (2013). Inverse kinematics using particle swarm optimization, a statistical analysis, *Procedia Engineering*, 64, 1602–1611.

12. Collinsm T. J., Shen W. M. (2017, April). Particle swarm optimization for high-DOF inverse kinematics, *In Control, Automation and Robotics (ICCAR), 2017 3rd International Conference on* (pp. 1–6). IEEE.

13. Mao B., Xie Z., Wang Y., Handroos H., Wu H., Shi S. (2017). A hybrid differential evolution and particle swarm optimization algorithm for numerical kinematics solution of remote maintenance manipulators, *Fusion Engineering and Design*, 124, 587–590.

14. Kachitvichyanukul V. (2012). Comparison of three evolutionary algorithms: GA, PSO, and DE, *Industrial Engineering and Management Systems*, 11(3), 215–223.

15. López-Franco C., Hernández-Barragán J., Alanis A. Y., Arana-Daniel N., López-Franco M. (2018). Inverse kinematics of mobile manipulators based on differential evolution, *International Journal of Advanced Robotic Systems*, 15(1), 1729881417752738.

16. Shiakolas P. S., Koladiya D., Kebrle J. (2005). On the optimum synthesis of six-bar linkages using differential evolution and the geometric centroid of precision positions technique, *Mechanism and Machine Theory*, 40(3), 319–335.

17. Juang C. F., Chen Y. H., Jhan Y. H. (2015). Wall-following control of a hexapod robot using a data-driven fuzzy controller learned through differential evolution, *IEEE Transactions on Industrial electronics*, 62(1), 611–619.

18. Pierezan J., Freire R. Z., Weihmann L., Reynoso-Meza G., dos Santos Coelho L. (2017). Static force capability optimization of humanoids robots based on modified self-adaptive differential evolution, *Computers & Operations Research*, *84*, 205–215.

19. Ngoc Son N., Anh H. P. H., Thanh Nam N. (2016). Robot manipulator identification based on adaptive multiple-input and multiple-output neural model optimized by advanced differential evolution algorithm, *International Journal of Advanced Robotic Systems*, 14(1), 1729881416677695.

20. **Wang M., Luo J., Fang J., Yuan J.** (2018). Optimal Trajectory Planning of Free-Floating Space Manipulator Using Differential Evolution Algorithm, *Advances in Space Research*.

21. Eusuff M., Lansey K., Pasha F. (2006). Shuffled frogleaping algorithm: a memetic meta-heuristic for discrete optimization, *Engineering optimization*, 38(2), 129–154.

22. Li X., Luo J., Chen M. R., Wang N. (2012). An improved shuffled frog-leaping algorithm with extremal optimisation for continuous optimization, *Information Sciences*, *192*, 143–151.

23. Samuel G. G., Rajan C. C. A. (2014). A modified shuffled frog leaping algorithm for long-term generation maintenance scheduling, *In Proceedings of the Third International Conference on Soft Computing for Problem Solving* (pp. 11–24). Springer, New Delhi.

24. Afzalan E., Taghikhani M. A., Sedighizadeh M. (2012). Optimal placement and sizing of DG in radial distribution networks using SFLA, *International Journal of Energy Engineering*, 2(3), 73–77.

25. **Ibrahim I. N.** (2018). Ultra Light-Weight Robotic Manipulator. *Bulletin of Kalashnikov ISTU*, 2018, vol. 21, no. 1, pp. 12–18, DOI: 10.22213/2413-1172-2018-1-12-18 (in Russian).

26. **Simon D.** (2013). *Evolutionary optimization algorithms*. John Wiley & Sons.

ГЛАВНОЕ СОБЫТИЕ В ОБЛАСТИ ПРИБОРОСТРОЕНИЯ, ТОЧНЫХ ИЗМЕРЕНИЙ, МЕТРОЛОГИИ И ИСПЫТАНИЙ

МОСКОВСКИЙ МЕЖДУНАРОДНЫЙ ИННОВАЦИОННЫЙ ФОРУМ

ТОЧНЫЕ ИЗМЕРЕНИЯ – ОСНОВА КАЧЕСТВА И БЕЗОПАСНОСТИ

Спешите забронировать стенд www.metrol.expoprom.ru

ДИНАМИКА, БАЛЛИСТИКА И УПРАВЛЕНИЕ ДВИЖЕНИЕМ ЛЕТАТЕЛЬНЫХ АППАРАТОВ

УДК 629.7.05

DOI: 10.17587/mau.19.725-733

В. Ф. Петрищев, д-р техн. наук, вед. науч. сотр., mail@samspace.ru, АО "Ракетно-космический центр "Прогресс", г. Самара

Энергосберегающий алгоритм автоматического управления принудительной посадкой пассажирского самолета. Часть І

Предложена задача создания системы автоматической посадки (САП) пассажирского самолета, инициируемой извне и исключающей возможность влияния экипажа на режим посадки, например, при изменении курса самолета и отсутствии связи с экипажем. В истории авиакатастроф имеется много случаев, которые можно было бы предотвратить, если бы на борту самолетов имелась САП, а технические средства аэропортов имели бы возможность инициируемой инициируемой устоять, если бы на борту самолетов имелась САП, а технические средства аэропортов имели бы возможность инициировать эту систему и отключать экипаж от процесса управления. Одним из таких памятных примеров явлются трагические события 11 сентября 2001 года в Нью-Йорке. Современный уровень техники позволяет решить задачу автоматической посадки самолета. Ярчайшим примером успешного решения этой задачи является посадка космического самолета (КС) "Буран" в автоматическом режиме 30 лет назад 15 ноября 1988 года.

Статья состоит из двух частей.

В первой части статьи рассмотрены условия успешного решения задачи автоматической посадки самолета. Кратко изложен перечень режимов полета самолета при автоматическом управлении посадкой. Для решения задачи автоматического управления самолетом в продольной плоскости на самом ответственном заключительном режиме посадки предложен энергосберегающий алгоритм управления, обеспечивающий управление в режиме отрицательной обратной связи. Приведено краткое описание энергосберегающего алгоритма управления. Алгоритм управления разработан применительно к пассажирскому самолету ТУ-154М. В состав вектора состояния системы включены шесть параметров: дальность, высота, угол тангажа и их первые производные по времени. При разработке алгоритма использованы следующие допущения: а) линейная модель зависимости аэродинамических характеристик самолета от угла атаки; б) линейная модель программного перевода тяги двигателей в режим малого газа на интервале 3 с от начала этапа выравнивания; в) использование углового ускорения самолета по каналу тангажа, возникающего при перекладке руля высоты, в качестве управляющего сигнала; г) частота работы алгоритма управления, равная 200 Гц.

Во второй части статьи на основе энергосберегающего алгоритма автоматического управления посадкой пассажирского самолета на заключительном участке посадки, разработанного в первой части, продолжены работы по анализу характеристик этого алгоритма. Разработана модельная программа управления посадкой применительно к самолету TУ-154M. Проведено математическое моделирование этапов режима посадки самолета. При переходе от одного этапа к другому проводилась конкатенация (сшивка) параметров движения, в результате которой конечные параметры движения предыдущего этапа становились начальными параметрами движения последующего этапа. Исследовано влияние погрешностей в аэродинамических характеристиках на условия посадки. В результате моделирования выявлено, что если для определения этапов использовать направление изменения угла тангажа, то режим посадки в общем случае складывается не из двух, традиционно определяемых, а из трех этапов: увеличения угла тангажа (выравнивание), его уменьшения (выдерживание) и вновь увеличения угла (этап назван поддерживанием). Необходимость введения третьего этапа обусловлена наличием погрешностей в аэродинамических характеристиках самолета. В целом подтверждено, что энергосберегающий алгоритм управления обеспечивает успешное решение задачи автоматической посадки пассажирского самолета на заключительном этапе его полета. При этом установлено, что длительность режима посадки не превышает 5 с.

Ключевые слова: автоматическая посадка, высота, глиссада, дальность, пассажирский самолет, руль высоты, скорость, тяга двигателя, угол атаки, угол тангажа, энергосберегающий алгоритм

Введение

В настоящей статье делается попытка обосновать необходимость иметь на борту современных самолетов гражданской авиации систему автоматической посадки (САП) в интересах повышения безопасности авиапассажиров и предлагается алгоритм управления этой системой на заключительном этапе посадки. В истории авиакатастроф имеется много случаев, которые можно было бы предотвратить, если бы на борту самолетов имелась САП, а технические средства аэропортов имели бы возможность инициировать эту систему и отключать экипаж от процесса управления. Одним из таких памятных примеров являются трагические события 11 сентября 2001 г. в Нью-Йорке.

Современный уровень техники позволяет решить задачу автоматической посадки самолета. Ярчайшим примером успешного решения этой задачи является посадка космического самолета (КС) "Буран" в автоматическом режиме 30 лет назад 15 ноября 1988 года. Понятно, что эта задача решалась в интересах Министерства Обороны СССР с соответствующими этой задаче экономическими затратами. В то время еще не были развернуты глобальные навигационные спутниковые системы "ГЛОНАСС" и GPS. Поэтому были разработаны специальные наземные радиотехнические средства измерения дальности от взлетно-посадочной полосы (ВПП) до КС с ретрансляцией этой информации на его борт, а также бортовые цифровые вычислительные средства автоматического управления на всей траектории движения КС от спуска с орбиты и до остановки на ВПП аэродрома посадки [1, 2].

В настоящее время при наличии глобальных навигационных спутниковых систем САП самолета может быть разработана со сравнительно малыми экономическими затратами по отношению, например, к стоимости самолета. Имеются косвенные сведения, что на ряде типов самолетов такие системы уже созданы. Например, в работе [3] отмечено, что в режиме автопилота аэробуса А-380 "доступны все режимы полета". При этом алгоритм управления при выравнивании и выдерживании не приводится и остается недоступным для анализа. Такие режимы могут применяться по усмотрению экипажа в экстренных ситуациях на борту.

Представляется особо важным создание САП, инициируемой диспетчерской службой аэропорта при определенных, строго оговоренных условиях без возможности вмешательства в ее работу со стороны экипажа, например, при явном отклонении самолета от заданного курса и отсутствии связи с экипажем. Это позволит снизить вероятность дальнейшего аварийного развития ситуации на борту. Знание того, что на данном типе самолета имеется САП, также может заставить террористов отказаться от его использования.

Анализ показывает, что на всех этапах движения возможно применение автоматического управления самолетом, если на борт будут закладываться необходимые данные об аэропорте с наилучшими погодными условиями для посадки, текущие параметры полета будут определяться бортовыми средствами в полном объеме, а характеристики самолета, необходимые для расчета режимов полета, будут заложены в память бортовых компьютеров. При этом недостающая информация о текущих параметрах самолета (вес, запас и расход топлива, тяга двигателей) может быть получена в ходе дальнейшего автоматического полета с использованием известных алгоритмов идентификации параметров.

Условия успешного решения задачи автоматической посадки самолета

Для решения задачи автоматической посадки самолета необходимо, прежде всего, сформулировать положения или условия, которые позволят конкретизировать саму постановку этой задачи. Вот некоторые из них:

- должны быть определены и, возможно, даже законодательно подтверждены условия перевода самолета диспетчерской службой аэропорта в режим автоматической посадки.
 Этот перевод может осуществляться только с Земли и применительно к самолетам, оборудованным САП;
- для применения САП должен быть создан специальный закрытый канал включения режима и получения необходимой контрольной информации от бортовых компьютеров самолета о параметрах работы двигателей и параметрах полета;
- должна быть исключена возможность вмешательства экипажа в управление самолетом после перевода самолета в режим автоматической посадки. Это достаточно жесткое условие означает, например, что на самолете, оснащенном САП, не может быть механических (рычажных) схем управления какими-либо элементами и все управление самолетом должно осуществляться с использованием бортовой дистанционной системы управления;
- при переводе самолета в режим САП в бортовые компьютеры заносятся данные о взлетно-посадочной полосе (ВПП) аэропорта посадки (геодезические широта, долгота и превышение над уровенной поверхностью Земли средней точки торцевой кромки ВПП, а также азимут средней линии ВПП);
- состав бортовых систем самолета и их датчиков должен быть достаточным для решения задачи САП. Это условие также является

достаточно жестким, поскольку в процессе проектирования режима посадки, возможно, потребуется иметь доступ к параметру какой-либо системы, ранее отсутствующему в бортовых компьютерах;

- в бортовых компьютерах самолета должны находиться все необходимые проектные параметры самолета;
- сведения о техническом состоянии самолета (вес, тяга двигателей, запас и расход топлива, положение рулевых органов и др.) в момент перевода самолета в режим автоматической посадки полагаются неизвестными. Из этого необходимо исходить при проектировании начального этапа режима посадки.

Приведенные условия не являются исчерпывающими и должны применяться для укрупненного планирования режимов автоматической посадки.

Планируемые режимы полета самолета при автоматическом управлении посадкой

Все рассматриваемые ниже планируемые режимы полета при использовании САП соответствуют принятым в гражданской авиации [4]. При их рассмотрении будем отмечать только необходимое различие в задачах и методах исполнения. При этом собственно аварийные ситуации самолета и двигателей не рассматриваются. Приводимые в работе данные соответствуют использованному в качестве примера самому массовому в СССР среднемагистральному самолету ТУ-154М.

Режим горизонтального автоматического пилотирования (режим автопилота). Основная задача этого режима состоит в идентификации параметров систем самолета и параметров его полета, без знания и прогнозирования которых невозможна дальнейшая автоматическая посадка. В этом режиме должны поддерживаться постоянными параметры горизонтального полета: высота полета, заданная горизонтальная скорость и азимут полета. Учитывая, что в момент перевода самолета в режим автоматической посадки он мог находиться в другом динамическом режиме, можно ожидать некоторого переходного процесса выхода на режим автопилотирования. В этом режиме с использованием проектных характеристик самолета должны быть определены такие параметры, как тяга двигателей, вес самолета, запас и расход топлива, работоспособность и эффективность органов управления самолетом и тягой двигателей. Длительность режима определяется длительностями решения перечисленных задач идентификации. Реализация режима автопилота обеспечивается на всех типах судов гражданской авиации и не вызывает вопросов.

Режим снижения по отечественной системе посадки СП-50 [3] начинается с разворота в целях изменения курса полета в первую контрольную точку, находящуюся на продолжении оси ВПП и отстоящую от посадочной торцевой кромки ВПП на 10 км и находящуюся на высоте круга над Землей 500 м. В режиме снижения должны согласованно выдерживаться расчетные значения угла тангажа, вертикальной скорости снижения и тяги двигателей. Управление и контроль за полетными параметрами в этом и в последующих режимах осуществляется бортовыми компьютерами с использованием информации от бортовых инерциальных средств и от глобальных спутниковых навигационных систем. Режим заканчивается моментом пролета над первой контрольной точкой.

Режим захода на посадку предусматривает круговое движение на постоянной высоте 500 м в целях выхода на ту же первую контрольную точку с азимутом осевой линии ВПП. В этом режиме по заданной программе последовательно осуществляются: выпуск шасси, выпуск предкрылков и закрылков в положение 1, выпуск закрылков в положение 2 (посадочное). При этом в соответствии с рассчитанной программой осуществляется снижение горизонтальной скорости полета до минимальной (для текущего веса самолета), а также увеличение угла атаки самолета для поддержания этой скорости.

Режим снижения по глиссаде осуществляется в посадочной конфигурации с непрерывным уменьшением высоты при постоянной горизонтальной скорости и реализуется также в соответствии с расчетной программой и показаниями датчиков глиссады. Пролет над дальним приводным радиомаяком (вторая контрольная точка) за 4000 м от торцевой кромки ВПП должен осуществляться на высоте 200 м, пролет над ближним приводным радиомаяком (третья контрольная точка) за 1000 м от торцевой кромки ВПП — на высоте 60 м с постоянной и минимальной скоростью.

Режим посадки начинается на высоте 6...8 м с момента пересечения торцевой кромки ВПП

(четвертая контрольная точка) и заканчивается касанием ВПП колесами основных стоек шасси. Это наиболее ответственный режим из всего процесса посадки. Традиционно он складывается из двух этапов: выравнивания и выдерживания. На этапе выравнивания двигатели переводятся в режим минимальной тяги с плавным увеличением угла атаки. На этом этапе осуществляется уменьшение вертикальной скорости снижения за счет увеличения угла атаки и, как следствие, увеличения подъемной силы. Угол атаки возрастает, а с ним и лобовое сопротивление. В результате продольная скорость снижается. На этапе выдерживания наоборот, несколько снижается угол атаки, продолжает снижаться продольная скорость, и, как следствие, снижается подъемная сила, самолет мягко касается ВПП колесами основных стоек шасси.

Режим пробега по ВПП с этого момента до полной остановки самолета обеспечивается следующими управляющими воздействиями: включением спойлеров (интерцепторов), уменьшением угла атаки самолета до соприкосновения с ВПП колесами передней стойки шасси, включением реверса тяги двигателей, последующим торможением колес основных стоек шасси и, наконец, выключением реверса при определенной скорости пробега.

Современное состояние аппаратурной базы бортовых и наземных средств позволяет обеспечить автоматическое управление на всех режимах посадки самолета.

Задача автоматического управления посадкой самолета и способы ее решения

В режиме автоматической посадки наиболее важным является рассмотрение управления в продольном канале, т. е. управление поступательным и вращательным движением самолета в плоскости тангажа. При этом в поперечных каналах крена и рыскания (скольжения) обеспечивается лишь стабилизация движения. Именно продольный канал наиболее детально исследован в технической литературе.

Традиционно решение терминальной задачи управления самолетом в режиме посадки связывается с использованием методов программного управления. Приведем известные примеры построения простейших программ управления при посадке самолета. Так, в работе [4] предложено кривую выравнивания при автоматической посадке задавать экспонентой. При этом единственным варьируемым параметром оказывается высота на момент начала выравнивания. По данным работы [5] в этом варианте при малых значениях вертикальной скорости самолета в момент касания точка касания колесами ВПП будет значительно удалена от начала ВПП, а при приземлении ближе к началу ВПП будет большая по модулю вертикальная скорость касания.

В работе [5] кривую выравнивания предложено задавать полиномом третьего порядка, определяющим зависимость высоты полета от пройденного пути вдоль оси ВПП от ее начала. Четыре коэффициента полинома определяются заданными краевыми условиями. Алгоритм разработан для военных самолетов типа МИГ-29 с отличной от принятой в гражданской авиации логикой управления в режиме посадки. Так, например, в режиме выравнивания осушествляется выпуск закрылков и шасси. Предложено по виртуальной кривой выравнивания определять программные значения продольной и вертикальной перегрузок, реализация которых должна обеспечиваться тягой двигателя и углом поворота стабилизатора соответственно.

Вместе с тем использование заранее разработанных "жестких" программ управления требует применения дополнительного стабилизирующего управления, задача которого состоит в компенсации отклонений от программы, возникающих при посадке от действия возмущений. Исследование методов формирования стабилизирующего управления привело специалистов в области автоматического управления к разработке принципа "гибких" траекторий. "Гибкая" программа управления находится путем решения двухточечной краевой задачи с заданными начальным состоянием в момент времени ее обновления и заданным конечным состоянием. В предельном случае обновление начального состояния и решение двухточечной краевой задачи может осуществляться на каждом шаге управления. В этом предельном случае получаем аналог системы с отрицательной обратной связью.

Существенный вклад в разработку алгоритмов управления самолетом в режиме посадки на основе концепции "гибких" траекторий внесли А. Б. Филимонов и Н. Б. Филимонов [6—10]. Ими же в работе [10] приведена обширная библиография по методам терминального управления объектами авиационной и космической техники.

В настоящей работе предложено использовать алгоритм управления, обеспечивающий работу системы управления в форме отрицательной обратной связи.

Движение самолета в продольной плоскости описывается тремя парами переменных: дальностью и продольной скоростью, высотой и вертикальной скоростью, углом тангажа и угловой скоростью его изменения. Поскольку в режиме посадки двигатели программно переводятся в режим малого газа, управление самолетом в этом режиме обеспечивается одним управляющим параметром — угловой скоростью поворота руля высоты. Поворот руля высоты вызывает механический момент поворота самолета в канале тангажа. Этот момент вызывает соответствующее угловое ускорение самолета по каналу тангажа. Для простоты в настоящей работе в качестве управляющего параметра принят результат воздействия руля высоты — угловое ускорение самолета в канале тангажа. В этом случае задача разработки программы управления в режиме посадки сводится к разработке алгоритма определения углового ускорения самолета в канале тангажа на каждом шаге управления.

Для автоматического управления движением в продольной плоскости в режиме посадки пассажирского самолета типа ТУ-154М в данной работе предлагается использовать энергосберегающий алгоритм управления [11]. Здесь термин "энергосберегающий" относится к названию применяемого алгоритма, а не к постановке задачи экономии энергозатрат на управление.

Практика применения энергосберегающего алгоритма показывает, что для обеспечения высокой точности расчетов требуется большое число циклов его работы, измеряемое сотнями и тысячами. Поскольку длительность режима посадки самолета мала и составляет 5...6 с, выбрана частота работы алгоритма 200 Гц, что соответствует интервалу Δt дискретности работы алгоритма, равному 0,005 с.

Важной особенностью управления в режиме посадки самолета является необходимость увеличения угла атаки на этапе выравнивания и уменьшения на этапе выдерживания. Эта особенность приводит к необходимости использовать последовательность алгоритмов управления на каждом из этих этапов с конкатенацией (сшивкой) параметров движения, используемых при управлении. Как будет показано ниже, это обстоятельство гарантирует успешную посадку самолета при наличии погрешностей в аэродинамических характеристиках самолета.

Ниже приводится краткое описание разработанного автором энергосберегающего алгоритма управления.

Краткое описание энергосберегающего алгоритма управления

Пусть задана линейная по состоянию и управлению дискретная полностью управляемая система:

$$X_{i+1} = A_i X_i + B_i U_i, i = 0, 1, 2, 3, ...,$$

где $X_i \in \mathbb{R}^n$ — вектор состояния системы в *n*-мерном евклидовом пространстве в момент времени, соответствующий номеру шага *i*; $U_i \in \mathbb{R}^r$ — вектор управлений в *r*-мерном евклидовом пространстве в тот же момент времени; A_i — матрица состояния системы размерности $n \times n$, которая зависит от X_i ; B_i — матрица управлений размерности $n \times r$, также зависящая от X_i . Пусть также известно начальное состояние системы: $X_0 \neq 0$.

Требуется найти регулятор, вырабатывающий последовательность управлений U_0 , U_1 , U_2 , ..., U_i , U_{i+1} , ..., на значения которых не наложены ограничения, переводящую систему из произвольного начального состояния в заданное конечное состояние (начало координат) X = 0 за неограниченное число шагов и при этом минимизирующий заданный ниже показатель качества системы.

Пусть в нашем распоряжении имеется линейная по состоянию и управлению система, уравнение движения которой нам также известно:

$$\xi_{i+1} = C_i \xi_i + D_i U_i,$$

где $\xi_i \in \mathbb{R}^n$ — вектор состояния системы. Матрицы C_i и D_i имеют размерности матриц A_i и B_i соответственно и в общем случае могут зависеть от ξ_i . Назовем эту систему вспомогательной. Вспомогательная система управляется тем же вектором U_i , что и заданная система. Кроме того, потребуем, чтобы начальное состояние вспомогательной системы совпадало с начальным состоянием заданной системы: $\xi_0 = X_0$.

От матрицы C_i требуется, чтобы невозмущенное движение вспомогательной системы было асимптотически устойчиво в целом. Порядок и примеры выбора матрицы C_i приведены в работе [11]. Полагаем $D_i = -B_i$.

В целом вспомогательная система обеспечивает формирование желаемой программы движения заданной системы.

Закон управления с обратной связью выбирается в линейной форме:

$$U_{i+1} = U_i + P_{i+1}(\tilde{X}_{i+1} - \xi_{i+1}),$$

т. е. управление на текущем шаге определяется в виде алгебраической суммы управления на предыдущем шаге и взвешенной разности векторов состояния заданной и вспомогательной систем на текущем шаге. Здесь P_{i+1} — весовая матрица, оптимальным образом взвешивающая разность между векторами состояний заданной и вспомогательной систем. Вектор состояния заданной системы определяется в результате измерений:

$$\tilde{X}_i = X_i + e_i,$$

где X_i — истинное значение вектора состояния; e_i — вектор случайных аддитивных погрешностей измерений типа дискретного белого шума.

По определению вводится ковариационная матрица управления на текущем шаге управления (математическое ожидание от произведения вектора управления на его транспонированное значение на один и тот же момент времени):

$$K_{U,\,i+1} = M(U_{i+1}U_{i+1}^{\mathrm{T}}).$$

Здесь верхний индекс ^т означает операцию транспонирования.

Критерием (показателем) качества, как следует из названия алгоритма, является минимум энергозатрат на управление на каждом шаге. Определяется он через след ковариационной матрицы управления (сумму квадратов ее диагональных элементов), являющийся функцией весовой матрицы:

$$\Im(i) = M(U_{i+1}^{\mathsf{T}}U_{i+1}) = \operatorname{Tr}[K_{U,i+1}(P_{i+1})] \to \min,$$

где Tr[...] — операция вычисления следа ковариационной матрицы управления $K_{U, i+1}$.

При такой постановке вспомогательная система играет роль ведущей системы, а заданная система — ведомой, следующей за ведущей. Идея изложенной постановки задачи состоит в следующем. Если вспомогательная система благодаря выбору матрицы C_i под действием управлений движется в начало координат, то траектория движения заданной системы не будет значительно отклоняться от траектории движения вспомогательной системы благодаря весовой матрице P_{i+1} и, следовательно, также будет направлена в начало координат. При возрастании отклонения вектор управления будет возрастать и компенсировать это возросшее отклонение.

Решение приведенной оптимизационной задачи приводит к оптимальному выражению для весовой матрицы:

$$P_{i+1} = -K_{U,i}(B_i - D_i)^{\mathrm{T}} [A_i K_{X,i} A_i^{\mathrm{T}} + C_i K_{\xi,i} C_i^{\mathrm{T}} + (B_i - D_i) K_{U,i} (B_i - D_i)^{\mathrm{T}} + K_e]^{-1}$$

и для ковариационной матрицы управления

$$K_{U, i+1} = [P_{i+1}(B_i - D_i) + E]K_{U, i}$$

на каждом интервале управления. Здесь

$$K_{X,i} = M(X_i X_i^{\mathsf{T}}), K_{\xi,i} = M(\xi_i \xi_i^{\mathsf{T}}), K_e = M(ee^{\mathsf{T}})$$

ковариационные матрицы состояний заданной и вспомогательной систем, характеризующие дисперсии текущих отклонений состояний от поставленной цели движения, и ковариационная матрица погрешностей измерений. Их значения должны задаваться при проведении расчетов.

Здесь требуется сделать некоторое пояснение. При разработке энергосберегающего алгоритма не ставилась задача использования его для решения терминальных задач. При формулировании постановки задачи ставился лишь вопрос о достижении заданной системой начала координат за неограниченное время. Однако практическое использование энергосберегающего алгоритма показало его способность выполнять задачу за фиксированное время. Эту способность можно объяснить возможностью выбора элементов переходной матрицы вспомогательной системы. Вспомогательная система является ведущей по отношению к заданной ведомой системе и обеспечивает в итоге достижение заданных значений параметров движения. Поэтому на практике всегда можно указать время завершения переходного

процесса при наличии значений допустимых погрешностей решения задачи. Это позволило применить энергосберегающий алгоритм для решения задачи автоматического управления посадкой самолета, т. е. задачи, которая, по нашему мнению, является терминальной в силу необходимости обеспечения нулевой вертикальной скорости в момент, когда высота полета приближается к нулю.

Разработка энергосберегающего алгоритма автоматического управления посадкой самолета

Введем необходимую систему координат и систему сил, действующих на самолет, представленную на рисунке. Для разрабатываемой системы управления с обратной связью начало системы координат *ОХҮ* поместим в конечную точку участка управления в режиме посадки — в точку касания ВПП колесами основных стоек шасси. Направление осей выбрано таким образом, чтобы начальные координаты продольного движения были отрицательными.

Запишем дифференциальные уравнения движения самолета в продольной плоскости. Для этого введем вектор состояния системы, содержащий шесть элементов:

$$X = [x_1; x_2; x_3; x_4; x_5; x_6]^{\mathrm{T}}.$$

Здесь x_1 — дальность до точки O касания ВПП; x_2 — высота полета над ВПП; x_3 — угол тангажа самолета, положительный при кабрировании; x_4 — продольная скорость полета в направлении оси X; x_5 — вертикальная скорость полета в направлении оси Y; x_6 — угловая скорость вращения самолета по тангажу.

Forces influencing on a landing plane

Учитывается, что направление действия силы тяги двигателей *P* совпадает с осью самолета.

Тогда в соответствии с действующими силами дифференциальные уравнения движения самолета могут быть записаны в виде системы шести уравнений:

$$\frac{dx_1}{dt} = x_4; \quad \frac{dx_2}{dt} = x_5; \quad \frac{dx_3}{dt} = x_6;$$
$$\frac{dx_4}{dt} = l; \quad \frac{dx_5}{dt} = h; \quad \frac{dx_6}{dt} = \varepsilon_{\upsilon}.$$

Здесь

$$l = \frac{-X_a \cos \Theta + Y_a \sin \Theta + P \cos x_3}{mg};$$

$$h = \frac{mg - X_a \sin \Theta - Y_a \cos \Theta - P \sin x_3}{mg}$$

 ε_{0} — управляющее угловое ускорение самолета по каналу тангажа; X_a и Y_a — аэродинамические силы сопротивления движению и подъемная сила; Θ — угол наклона траектории полета к местному горизонту; G = mg — вес самолета; m — его масса; g — гравитационное ускорение у поверхности Земли; P — тяга двигателей. Следует еще раз отметить, что управляющий параметр ε_0 выбран в целях простоты расчетов, поскольку перекладка руля высоты в итоге вызывает угловое ускорение поворота самолета по каналу тангажа.

Аэродинамические силы определяются по известным формулам:

$$X_a = \frac{1}{2} \rho V^2 C_X S, \quad Y_a = \frac{1}{2} \rho V^2 C_Y S,$$

где р — плотность атмосферы на заданной высоте при заданной температуре; $V = (x_4^2 + x_5^2)^{1/2}$ скорость полета самолета; C_X и C_Y — безразмерные коэффициенты соответствующих аэродинамических сил; *S* — площадь крыла самолета.

Обозначим управляющий скаляр: $u = \varepsilon_{v}$.

Используя введенный вектор состояния системы и скаляр управления, проведем дискретизацию системы с использованием матрицы Коши [11]. В результате приведенные выше уравнения движения могут быть записаны в дискретной рекуррентной форме с малым шагом дискретизации по времени Δt :

$$X_{i+1} = A_i X_i + b_i u_i,$$

где

$$A_{i} = \begin{bmatrix} 1 & 0 & 0 & \Delta t & 0 & 0 \\ 0 & 1 & 0 & 0 & \Delta t & 0 \\ 0 & 0 & 1 & 0 & 0 & \Delta t \\ \frac{I\Delta t}{x_{1}} & 0 & 0 & 1 & 0 & 0 \\ 0 & \frac{h\Delta t}{x_{2}} & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}_{i}^{\mathrm{T}},$$

Назовем эту систему заданной.

Для использования энергосберегающего алгоритма управления посадкой самолета необходимо ввести вспомогательную систему

$$\xi_{i+1} = C_i \xi_i + d_i u_i,$$

в которой векторы и матрицы имеют ту же размерность, что и в заданной системе. Вспомогательная система управляется тем же вектором управления, ее начальное состояние совпадает с начальным состоянием заданной системы $\xi_0 = X_0, d_i = -b_i$, а матрица C_i обеспечивает выполнение условия асимптотической устойчивости невозмущенного движения вспомогательной системы (при $u_i \equiv 0$). Рекомендации по выбору элементов матрицы C_i приведены в работе [11].

Во второй части статьи разработанный алгоритм будет применен для построения модельной программы управления самолетом на заключительном участке посадки, проведения математического моделирования и оценки эффективности применения алгоритма по результатам моделирования.

Заключение

Предложена задача создания системы автоматической посадки пассажирского самолета, инициируемой извне и исключающей возможность влияния экипажа на режим посадки. Рассмотрены условия успешного решения задачи автоматической посадки самолета. Кратко изложен перечень режимов полета самолета при автоматическом управлении посадкой. Для решения задачи автоматического управления самолетом в продольной плоскости на самом ответственном заключительном режиме посадки предложен энергосберегающий алгоритм управления, обеспечивающий управление в режиме отрицательной обратной связи. Алгоритм управления разработан применительно к пассажирскому самолету ТУ-154М. При разработке алгоритма использованы следующие допущения:

- линейная модель зависимости аэродинамических характеристик самолета от угла атаки;
- линейная модель программного перевода тяги двигателей в режим малого газа на интервале 3 с от начала этапа выдерживания;
- использование углового ускорения самолета по каналу тангажа, возникающего при перекладке руля высоты, в качестве управляющего сигнала;
- частота работы алгоритма управления, равная 200 Гц.

В дальнейшем разработанный алгоритм будет использован для проведения математического моделирования в целях оценки его эффективности.

Список литературы

1. URL: www.buran.ru

2. Набойщиков Г. Ф. Авиационно-космические системы / Под ред. Лозино-Лозинского. М.: Изд-во МАИ, 1997. 416 с.

3. **Коптев А. Н.** Авиационное и радиоэлектронное оборудование воздушных судов гражданской авиации. Кн. 3. Учеб. пособ. Самара: Изд-во СГАУ, 2011. 392 с.

4. Котик М. Г. Динамика взлета и посадки самолета. М.: Машиностроение, 1984. 256 с.

5. **Кулифеев Ю. Б., Афанасьев Ю. Н.** Алгоритм автоматической посадки самолета // Электронный журнал "Труды МАИ". Вып. 62.

6. **Филимонов Н. Б.** Аналитическое конструирование квазиоптимальной системы терминального управления // Аналитические методы синтеза регуляторов: Межвуз. науч. сб. Вып. 3. Саратов: СПИ, 1978. С. 100–113.

7. Филимонов А. Б., Филимонов Н. Б. Циклические процессы регулирования в нелинейных объектах // Аналитические методы синтеза регуляторов: Межвуз. науч. сб. Саратов: СПИ, 1988. С. 90—96.

8. Пухов А. Л., Толокнов В. И., Филимонов Н. Б. Компьютерная сертификация посадочного маневра сверхзвукового пассажирского самолета Ту-144. М.: Изд. ОАО "Туполев", 2001.

9. Филимонов А. Б., Филимонов Н. Б. Принцип гибких кинематических траекторий управления терминальными маневрами летательных аппаратов // Системы управления беспилотными космическими и атмосферными летательными аппаратами: Тез. докл. II Всерос. науч.-техн. конф. М.: МОКБ "Марс", 2012. С. 51–53.

10. **Проблемы** управления сложными динамическими объектами авиационной и космической техники / С. Н. Васильев и др. / Под ред. акад. РАН С. Н. Васильева. М.: Машиностроение, 2015. 519 с.

11. **Петрищев В. Ф.** Энергосберегающее управление объектами ракетно-космической техники. Самара: Изд-во СамНЦ РАН, 2017. 140 с.

Energy-Saving Algorithm of Automatic Control of Compulsory Passenger Carrier Landing. Part 1

V. F. Petrishchev, mail@samspace.ru,

Progress Space-Rocket Centre JSC, Samara, 443122, Russion Federation

Corresponding auther: Petrishchev Vladimir F., Doctor of Engineering, Leading Researcher, Progress Space-Rocket Centre JSC, Samara, 443122, Russion Federation, e-mail: mail@samspace.ru

Accepted July 07, 2018

Abstract

The task was to develop an automatic landing system (ALS) for a passenger carrier that can be externally activated and excludes the possibility of the crew's interference into the landing process, for example, when a carrier alters its nominal course or there is no contact with the crew. The air crush history saw a lot of cases that could have been prevented if the planes had had an ALS system and airports had had possibilities to activate that system and suspend the crew from flight control. One of such unforgettable examples is the New-York tragedy of September 11, 2001. State-of-the-art technology allows solving the problem of automatic carrier landing. The most remarkable example demonstrating solution of this problem is the automatic landing of the Buran orbiter 30 years ago on November 15, 1988. The article consists of two sections. The first section of the article deals with conditions of effective solution of autoland problem. It describes in short, the flight modes during automatic landing control. To solve the problem of automatic longitudinal control in the most crucial final landing mode, the author proposes an energy-saving control algorithm that provides control in the mode of negative feedback. The system status vector comprises six parameters: range, altitude, pitch angle, and their first-order derivatives. The control algorithm is developed for the Tupolev TU-154M airliner. In development of the algorithm, the following assumptions were used: a) a linear model of dependence of aerodynamic data on the angle of attack; b) a linear model of programmed switch of engine thrust to the idle mode on the interval of 3 seconds from the beginning of the flareout; c) a pitch angular acceleration, occurring at elevator rate reversal, as a control signal; d) the frequency of the control algorithm operation equal to 200 Hz. The second section further analyzes characteristics of the energy-saving algorithm of automatic control of compulsory passenger carrier landing during the final landing phase, which was developed in the first section. The author developed a model program of control and mathematically modeled the carrier landing phases. When switching from one phase to another, the motion parameters were concatenated so that the final motion parameters of the previous phase became the initial motion parameters of the next phase. The author also studied the influence of errors in aerodynamic data on the landing conditions. The modeling revealed that if a pitch deflection direction is used for the determination of phases, then in a general case, the landing mode consists not of two traditionally determined phases, but of the following three: pitch angle increase (flareout), pitch angle decrease (float), and again, pitch angle increase (this phase is called 'maintenance'). The necessity to introduce the third phase is determined by the presence of errors in the aerodynamic data of the airplane. On the whole, it is confirmed that the energy saving control algorithm provides successful solution of the problem of automatic landing of a passenger carrier at its final flight phase. At that, it is determined that the landing mode does not exceed 5s.

Keywords: autoland, altitude, glissade, range, passenger carrier, elevator, velocity, engine thrust, angle of attack, pitch angle, energy saving algorithm

For citation:

Petrishchev V. F. Energy-Saving Algorithm of Automatic Control of Compulsory Passenger Carrier Landing. Part 1, *Mekhatronika, Avtomatizatsiya, Upravlenie,* 2018, vol. 19, no.11, pp. 725–733.

DOI: 10.17587/mau.19.725-733

References

1. Available at: www.buran.ru.

2. Naboishchikov G. F. Aerospace systems: Book of Technical Papers. Moscow, MAI publ., 1997, pp. 33-46 (in Russian).

3. **Koptev A. N.** Aviationnoye i padioelektronnoye oborudovanie vozdushnykh sudov grazhdanskoi aviatsii [Avionics of Civil Aircraft]. Vol. 3. Teaching aid. Samara: SGAU publ., 2011, 392 p. (in Russian).

4. Kotik M. G. *Dinamika vzlyota i posadki samolyota* [The Dynamics of Airplane Take-Off and Landing]. Moscow: Mashinostroyenie, 1984, 256 p. (in Russian).

5. Kulifeev Yu. B., Afanasyev Yu. N. Algoritm avtomaticheskoi posadki samolyota [The Algorithm of Automatic Aircraft Landing]. *Trudy MAI electronic journal*, vol. 62.

6. Filimonov N. B. Analiticheskoye konstruirovanie kvazioptimalnoy sistemy terminalnogo upravlenia [Analytic Design of a Quasioptimal Termination Control System]. *Analiticheskiye metody sinteza regulatorov: mezhvuz. nauchn. sb.* [Analytic Methods of Regulator Synthesis: Inter-institute collected articles]. Vol. 3, Saratov, SPI, 1978, pp. 100–113 (in Russian). 7. Filimonov A. B., Filimonov N. B. Tsiklicheskie protsessy regulirovania v nelineynykh obyektakh [Cyclic Control Processes in Non-Linear Objects]. *Analiticheskiye metody sinteza regulatorov: mezhvuz. nauchn. sb.* [Analytic Methods of Regulator Synthesis: Inter-institute collected articles]. Saratov, SPI, 1988, pp. 90–96 (in Russian).

8. Pukhov A. L., Toloknov V. I., Filimonov N. B. Kompyuternaya sertifikatsia posadochnogo manevra sverkhzvekovogo passazhirskogo samoleta Tu-144 [Computed Certification of the Landing Maneuver of the Tupolev Tu-144 Supersonic Airliner]. Moscow, OAO Tupolev, 2001 (in Russian).

9. Filimonov A. B., Filimonov N. B. Printsip gibkikh kinematicheskikh trayektoriy upravlenia terminalnymi manevrami letatelnykh apparatov [The Principle of Flexible Kinematic Trajectory of Control of Final Maneuvers of Airborne and Space Vehicles]. Sistemy upravlenia bespilotnymi kosmicheskimi i atmosfernymi letatelnymi apparatami: Tez. dokl. II Vseros. nauch.-tekhn. konf. [Control Systems of Unmanned Airborne and Space Vehicles: Proceedings of the II-nd All-Russian Scientific and Technical Conference]. Moscow, MOKB Mars, 2012, pp. 51–53 (in Russian).

10. **Vasilyev S. N. et al.** *Problemy upravlenia slozhnymi dinamicheskimi obyektami aviatsionnoy i kosmicheskoy tekhiki* [Control Problems of Sophisticated Airborne and Space Dynamic Objects] Moscow, Mashinostroyenie publ., 2015, 519 p. (in Russian).

11. Petrishchev V. F. Energosberegayushcheye upravlenie obyektami raketno-kosmicheskoi tekhniki [Energy Saving Control of Rocket and Space Equipment]. Samara: SamNTs RAN Publ, 2017, 140 p. (in Russian). Ван Гуоянь, аспирант, guoyan@yandex.ru, А. В. Фомичев, канд. техн. наук, доц., a.v.fomichev@bmstu.ru Московский государственный технический университет им. Н. Э. Баумана, г. Москва

Алгоритм планирования безопасного маршрута движения марсохода с учетом рельефа местности

Для успешного осуществления запланированной миссии полета на Марс необходимо решение проблемы планирования пути движения марсохода по поверхности планеты с учетом особенностей рельефа местности.

Традиционные алгоритмы поиска пути, такие как A^* , а также улучшенный алгоритм $A^* - D^*$ и Field D^* , которые успешно применялись на планетоходах в состоявшихся экспедициях на Луну и Марс, имеют проблему поиска кратчайшего оптимального пути.

В то же время такие улучшенные алгоритмы A*, как Basic Theta*, Lazy Theta* позволяют планировать движение в любом направлении и искать более короткие маршруты движения.

В данной статье на основе проведенного сравнительного анализа существующих алгоритмов A*, Basic Theta*, Lazy Theta* предложена новая модификация алгоритма планирования движения в любом направлении Lazy AT, позволяющая сократить время вычисления и получить более короткий путь от исходной до конечной точки маршрута.

На основе анализа поверхностных особенностей рельефа планеты можно судить о показателе опасности ее рельефа. Основной опасностью при движении по поверхности Марса для колесного марсохода являются наклоны и препятствия. Исходя из требований устойчивости положения марсохода при его движении и способности преодоления препятствий были выбраны следующие топографические коэффициенты для описания показателя опасности: угол наклона местности, шероховатость поверхности, размах рельефа.

В новой модификации алгоритма Lazy AT для получения безопасной траектории движения марсохода предлагается использовать индикатор опасности местности (индикатор риска), который учитывается в модификации алгоритма Risk Lazy AT.

Сравнительный анализ результатов моделирования алгоритмов Risk Lazy AT и Lazy Theta* показал, что предложенный новый алгоритм Risk Lazy AT может обеспечить безопасное движение подвижного объекта по поверхности планеты.

Основываясь на реальных данных о рельефе поверхности Марса, была разработана цифровая карта поверхности планеты и проведено моделирование пространственного маршрута движения марсохода с помощью нового алгоритма Risk Lazy AT.

Ключевые слова: алгоритм A*, алгоритм планирования движения в любом направлении, алгоритм Basic Theta*, алгоритм Lazy Theta*, алгоритм Lazy AT, алгоритм Risk Lazy AT, индикатор опасности местности

Введение

Одна из важнейших задач, требующих решения при планировании маршрута марсохода, заключается в том, чтобы его траектория движения от начальной до конечной (целевой) точки маршрута, в общем случае проходящая в малоизученной обстановке на поверхности планеты, обеспечивала его гарантированную безопасность. Поэтому разработка алгоритма планирования маршрута марсохода, который в автономном режиме обеспечивал бы поиск оптимальной и безопасной траектории, является в настоящее время весьма актуальной задачей.

Известно, что в уже состоявшихся экспедициях на Луну и Марс на планетоходах успешно применялись следующие подобные алгоритмы: алгоритмы А* и улучшенный алгоритм А*, например D*, а также Field D*, разработанный на базе алгоритма D* [1—5].

Алгоритмы А* и D* являются эвристическими алгоритмами. Как указано в работе [6], эвристический алгоритм (эвристика) — это алгоритм решения задачи, включающий практический метод, не являющийся гарантированно точным или оптимальным, но достаточный для решения поставленной задачи. Проще говоря, эвристика — это не полностью математически обоснованный (или даже иногда "не совсем корректный"), но при этом практически полезный алгоритм. Его преимущества заключаются в высокой эффективности поиска, простоте реализации и возможности ускоренного решения задачи в тех случаях, когда точное решение не может быть найдено.

Алгоритм планирования движения в любом направлении (алгоритм ПДЛН) основан на алгоритме А* [7—9]. С помощью данного алгоритма по сравнению с алгоритмами А*, D* возможно найти более короткие маршруты движения. Это является следствием того, что в алгоритмах А*, D* движение формируется, главным образом, либо вдоль границ ячейки сетки, либо по ее диагонали. Напротив, алгоритм ПДЛН

Рис. 1. Примеры траектории движения при фиксированном угле (*a*) и в любом направлении (*б*)

Fig. 1. Geometric interpretation of trajectory planning with fixed angle (a) and in any-angle (δ)

пути (Путь 2) Fig. 3. Option 2 Trajectory updating (Trajectory 2)

(алгоритмы Basic Theta* и Lazy Theta*) не имеет ограничений в выборе направления движения. Например, как видно из рис. 1, *а* и *б*, на траектории, реализованной с помощью алгоритма ПДЛН, удается избежать лишних изменений (отклонений, изгибов) при движении по направлению от начальной точки A4 к целевой точке C1 маршрута. Поэтому с помощью алгоритма ПДЛН возможно в значительной степени сократить результирующий путь. Таким образом, изучение свойств и особенностей данного алгоритма представляет особый интерес.

В алгоритмах ПДЛН траектории движения могут обновляться согласно двум вариантам:

- в первом варианте (рис. 2) путь обновляется между текущей вершиной и соседней вершиной ной расширения (будем называть его Путь 1);
- во втором варианте (рис. 3) путь обновляется между родительской вершиной (parent) и произвольной вершиной расширения (будем называть его Путь 2).

Описание эвристических алгоритмов поиска маршрута

С помощью эвристических алгоритмов поиска возможно находить маршруты движения от одной вершины (начальной) к другой вершине (целевой, конечной) с наименьшими затратами (стоимостью). Порядок обхода вершин при движении по маршруту определяется эвристической функцией, учитывающей показатели "расстояние + стоимость" и имеющей следующий вид:

$$f(n) = g(n) + h(n),$$
 (1)

где f(n) — значение эвристической функции; n — обозначение (номер) рассматриваемой вершины; g(n) — значение функции стоимости достижения рассматриваемой вершины из начальной вершины (может быть как эвристической функцией, так и нет); h(n) — значение функции эвристической оценки расстояния от рассматриваемой вершины до конечной (должна быть допустимой эвристической оценкой).

В общем случае функцию h(n) возможно вычислить различными методами. В данной работе используется один из наиболее распространенных методов оценки расстояния — от рассматриваемой вершины до конечной, а именно:

$$h(n) = \sqrt{(x_t - x_s)^2 + (y_t - y_s)^2},$$
 (2)

где x_t , x_s — значения абсцисс рассматриваемой и целевой вершин; y_t , y_s — значения ординат рассматриваемой и целевой вершин.

Рассмотрим свойства и особенности нескольких алгоритмов поиска маршрутов движения.

Алгоритм A^* — это алгоритм поиска, который позволяет найти во взвешенном графе маршрут с наименьшим значением эвристической функции f(n) при движении от начальной вершины до заданной конечной вершины. Алгоритм A^* можно упростить так, как показано на рис. 4.

Введем следующие обозначения: s — рассматриваемая вершина; s' — вершина расширения; *parent(s)* — родительская вершина; g(parent(s)) функция стоимости достижения рассматриваемой вершины (*parent(s*)) из начальной вершины; b(parent(s), s') — функция стоимости достижения рассматриваемой вершины (s') из вершины *parent(s*); Neighbour(s) — соседние вершины, которые можно видеть.

Рассмотрим основные этапы выполнения алгоритма А*.

1. Точка s_0 является начальной вершиной, а также рассматриваемой вершиной. Точки s'_0 (*Neighbour*(s_0)) являются ее соседними верши-

Рис. 4. Упрощенный алгоритм A* Fig. 4. Brief explanation of Algorithm A*

нами расширения, как показано на рис. 4, *а*. Вычислим все значения эвристической функции $f(s_0)$ для начальной (рассматриваемой) вершины s_0 .

2. Выберем вершину расширения s', соответствующую наименьшему значению функции $f(s_0)$. В качестве рассматриваемой вершины принимается вершина s_1 (т. е. $s'_0 = s_1$). Затем обновляются пути так, как показано на рис. 4, *б*. Расширив вершину s_1 в разных направлениях, получим вершины расширения s'_1 (*Neighbour*(s_1)

Рис. 5. Упрощенный алгоритм Basic Theta* Fig. 5. Brief explanation of Algorithm Basic Theta*

и s'_1 , которые не включают вершины расширения на прошлом этапе). Вычислим все значения эвристической функции $f(s_1)$ для рассматриваемой вершины s_1 .

3. Выберем вершину расширения s'_1 , соответствующую наименьшему значению $f(s_1)$. Далее в качестве рассматриваемой вершины принимается вершина s_2 (т. е. $s'_1 = s_2$). Обновим пути так, как показано на рис. 4, *в*. Расширив вершину s_2 , получим вершины расширения s'_2 (*Neighbour*(s_2)) и s'_2 , которые не включают вершины расширения на прошлом этапе) так, как показано на рис. 4, *в*.

4. Целевая вершина *g* находится в одной из вершин расширения s'_2 . Обновим пути так, как показано на рис. 4, *г*. Таким образом, ито-говый путь от начальной до конечной точки маршрута $s_0 - s_1 - s_2 - g$ показан на рис. 4, *г*.

Алгоритм Basic Theta* является модификацией алгоритма A* и имеет определенное сходство с алгоритмом D* [10, 11]. Отличительная особенность алгоритма Basic Theta* — способность обеспечивать движение по короткому пути с меньшим числом изменений в направлении движения. При этом траектория может распространяться в любом направлении, не обязательно через ближайшие вершины, но при условии, что между ними имеется прямая видимость. Основные этапы алгоритма Basic Theta* показаны на рис. 5.

Проанализируем основные этапы алгоритма Basic Theta* (рис. 5).

> 1. Содержание этапа аналогично этапу 1 алгоритма А* (см. рис. 4, *a* и 5, *a*).

> 2. Содержание этапа аналогично этапу 2 алгоритма А* (см. рис. 4, б и 5, б).

> 3. Проверка поля зрения (пунктирная линия (рис. 5, *в*) между родительской вершиной (*parent*(s_1) = s_0) и всеми вершинами расширения s'_1 .

> Если между родительской вершиной (*parent*(s_1) = s_0) и вершиной расширения s'_1 имеется прямая видимость, то значение эвристической функции $f(s_1) = g(parent(s_1)) + b(parent(s_1), s'_1) + h(s'_1)$.

> Если между родительской вершиной (*parent*(s_1) = s_0) и вершиной расширения s'_1 не имеется

прямой видимости, то значение эвристической функции $f(s_1) = g(s_1) + b(s_1, s'_1) + h(s'_1)$.

Выберем вершину расширения s'_1 , соответствующую наименьшему значению эвристической функции $f(s_1)$, в качестве рассматриваемой вершины s_1 . На рис. 5, *в* показан пример, когда между родительской вершиной и соответствующей вершиной расширения s'_1 имеется прямая видимость.

4. Согласно варианту 2 обновления пути (Путь 2) принимаем $(parent(s_1) = s_0)$ и $s'_0 = s_1$. Расширяем вершину s_1 (рис. 5, e) и получаем вершины расширения s'_1 , которые не включают вершины расширения на предыдущем этапе (*Neighbour*(s_1) и s'_1).

5. Целевая вершина *g* находится в одной из вершин расширения s'_1 . Если между родительской вершиной *parent*(s_1) = s_0 и целевой вершиной *g* нет прямой видимости (рис. 5, ∂ , пунктирная линия), то дальнейшее обновление пути между текущей (рассматриваемой) вершиной s_1 и целевой вершиной *g* выполняется согласно варианту 1 (Путь 1).

6. На этом работа алгоритма заканчивается. Итоговый путь от родительской вершины до целевой вершины $s_0 - s_1 - g$ показан на рис. 5, *е*.

Алгоритм Lazy Theta* представляет собой модификацию алгоритма Basic Theta* [12], обеспечивающую более высокую скорость вычислений, поскольку в данном алгоритме реализован простой подход к выполнению проверки видимости вершин расширения, находящихся в данный момент в поле зрения из родительской вершины.

Алгоритм Lazy AT — это новый алгоритм, разработанный авторами статьи на основе алгоритма Lazy Theta* в результате оптимизации и комбинирования достоинств предыдущих алгоритмов. Данный алгоритм менее сложен по структуре и обладает более высоким быстродействием. Алгоритм Lazy AT отличается от рассмотренных алгоритмов процедурой поиска рассматриваемой вершины, которая позволяет уменьшить время вычислений. В алгоритме Lazy AT еще добавляется проверка поля зрения между родительской вершиной родителя *parent(parent(s))* и соответствующей вер-

Рис. 6. Упрощенный алгоритм Lazy Theta* и Lazy AT Fig. 6. Brief explanation of Algorithm Lazy Theta* and Lazy AT

шиной расширения *s*', чтобы получить более короткий путь маршрута.

Основные этапы алгоритмов Lazy Theta* и Lazy AT показаны на рис. 6.

Проанализируем основные этапы алгоритмов Lazy Theta* и Lazy AT (рис. 6).

1. Содержание этапа аналогично этапу 1 алгоритма А* (см. рис. 2, *а* и 6, *а*).

2. Содержание этапа аналогично этапу 2 алгоритма А* (см. рис. 2, б и 6, б).

3. Для алгоритмов Lazy Theta* и Lazy AT (рис. 6, *в*): заранее предполагается, что между родительской вершиной *parent*(s_1) = s_0 и вершиной расширения s'_1 должна быть прямая видимость.

Для алгоритма Lazy Theta*:

• Если условие

 $g(parent(s_1)) + b(parent(s_1), s') < g(s'_1)$

выполняется, то значение эвристической функции

$$f(s_1) = g(parent(s_1)) + b(parent(s_1), s'_1) + h(s'_1)$$
.

Если условие

 $g(parent(s_1)) + b(parent(s_1), s') < g(s'_1)$

не выполняется, то значение эвристической функции

 $f(s_1) = g(parent(s_1)) + b(parent(s_1), s'_1) + h(s'_1)$.

Выберем вершину расширения s'_1 , соответствующую наименьшему значению эвристических функций $f(s_1)$, в качестве рассматриваемой вершины s_1 .

Проверяется свойство прямой видимости (рис. 6, *в*) между родительской вершиной (*parent*(s_1) = s_0) и соответствующей вершиной расширения s'_1 (проверка выполняется только один раз). На рис. 6, *в* пунктирной линией по-казано, что между данными вершинами имеется прямая видимость.

Для алгоритма Lazy AT:

▶ Выберем вершину расширения *s*'₁, соответствующую наименьшему значению эвристических функций

$$f(s_1) = g(parent(s_1) + b(parent(s_1), s'_1) + h(s'_1),$$

в качестве рассматриваемой вершины s₁.

Проверяется свойство прямой видимости между родительской вершиной (*parent*(s_1) = s_0) и соответствующей вершиной расширения s'_1 (проверка выполняется только один раз). На рис. 6, *в* пунктирной линией показано, что между данными вершинами имеется прямая видимость.

Затем выполняется проверка свойства прямой видимости между родительской вершиной родителя *parent(parent(s₁))* и соответствующей вершиной расширения s'_1 . Если между родительской вершиной родителя *parent(parent(s₁))* и соответствующей вершиной расширения s'_1 имеется прямая видимость, то дальнейшее обновление пути между родительской вершиной родителя *parent(parent(s₁))* и соответствующей вершиной расширения s'_1 выполняется согласно варианту 2 (Путь 2).

4. Согласно варианту 2 обновления пути (Путь 2) принимаем *parent*(s_1) = s_0 и $s'_0 = s_1$. Расширяем вершину s'_1 (рис. 6, *г*) и получаем вершины расширения s'_1 , которые не включают вершины расширения на предыдущем этапе (*Neighbour*(s_1) и s'_1).

5. Целевая вершина *g* находится в одной из вершин расширения s'_1 . Если между родительской вершиной *parent*(s_1) = s_0 и целевой вершиной *g* нет прямой видимости (рис. 6, ∂ , пунктирная линия), то дальнейшее обновление пути между текущей (рассматриваемой) вершиной s_1 и целевой вершиной *g* выполняется согласно варианту 1 (Путь 1).

Для алгоритма Lazy AT:

Кроме этого, выполняется проверка условия прямой видимости между родительской вершиной родителя *parent*(*parent*(s_1)) = s_0 и целевой вершиной *g*.

Если между родительской вершиной родителя *parent*(*parent*(s_1)) и целевой вершиной *g* нет прямой видимости, то дальнейшее обновление пути между родительской вершиной родителя *parent*(*parent*(s_1)) и целевой вершиной *g* выполняется согласно варианту 1 (Путь 1). \blacktriangleleft

6. На этом работа алгоритма заканчивается. Итоговый путь от родительской вершины до целевой вершины $s_0 - s_1 - g$ показан на рис. 6, *е*.

Показатели опасности местности, по которой движется марсоход

При нахождении марсохода в какой-либо неизвестной неструктурированной среде (слабо изученной области планеты) особенно важно соблюдать правила безопасности его передвижения.

На основе анализа поверхностных особенностей рельефа планеты можно судить о показателе опасности ее рельефа. Показатель опасности рельефа является одним из оптимизируемых параметров алгоритма.

Анализ показателя опасности рельефа помогает найти для марсохода траекторию с низким уровнем риска.

Топографические особенности

В настоящее время обычно используют колесную конфигурацию марсохода. Основной опасностью при движении по поверхности Марса для колесного марсохода являются наклоны и препятствия. При значении угла наклона поверхности планеты больше его критического значения возможно опрокидывание аппарата. Преодоление препятствий — это одна из основных причин для существенного изменения положения марсохода. Данная ситуация часто возникает на поверхности планеты и может привести к снижению координации между колесами аппарата, а также к его застреванию в ямах, впадинах и других неровностях рельефа.

Прежде чем определить показатели опасности местности, необходимо провести анализ местности.

Исходя из требований устойчивости положения марсохода при его движении и преодолении препятствий были выбраны следующие топографические коэффициенты: угол наклона местности, шероховатость поверхности, размах рельефа.

• Анализ угла наклона местности

Угол наклона местности или крутизна склона (ската) — угол, образуемый направлением склона с горизонтальной плоскостью и выражаемый в угловых мерах или уклонах. Анализ угла наклона местности является одной из важных задач при анализе характеристик местности. Из литературы [13] известно, что в настоящее время наибольшее значение угла наклона местности, при котором марсоход может выполнить подъем, составляет 30°. Поэтому для оценки способности марсохода преодолевать склоны в работе было использовано именно это критическое значение данного показателя.

Введем следующие обозначения: $(i, j, Z_{i, j})$ — координаты положения ячейки на сетке (рис. 7, направление стрелки на сетке указывает направление движения), где i, j — координаты ячейки сетки (индексы); $Z_{i, j}$ — высота ячейки сетки над горизонтальной поверхностью; e — значение шага сетки (размер ячейки).

Рис. 7. Определение положения на координатной сетке Fig. 7. Given position on the coordinate grids

Введем два показателя, характеризующих изменение высоты поверхности:

• в направлении "восток-запад"

$$f_x(i,j) = \frac{Z_{i+1,j+1} + Z_{i+1,j} + Z_{i+1,j-1} - Z_{i-1,j+1} - Z_{i-1,j} - Z_{i-1,j-1}}{6e};$$
(3)

• в направлении "север-юг"

$$f_{y}(i,j) = \frac{Z_{i-1,j+1} + Z_{i,j+1} + Z_{i+1,j+1} - Z_{i-1,j-1} - Z_{i,j-1} - Z_{i+1,j-1}}{6e}.$$
 (4)

Тогда угол наклона местности вычисляется по следующему соотношению:

$$\theta(i,j) = \arctan \sqrt{f_x^2(i,j) + f_y^2(i,j)}.$$
 (5)

• Шероховатость поверхности (рельефа)

Шероховатость поверхности является отражением степени подъема и падений рельефа. Шероховатость может влиять на устойчивость движения марсохода, что важно для обеспечения эффективной работы его оборудования.

Шероховатость рельефа (r) обычно определяется как отношение площади поверхности ($S_1(i, j)$) к его проекции (S) на горизонтальную плоскость, т. е.

$$r(i, j) = S_1(i, j)/S.$$
 (6)

• Размах рельефа

Размах рельефа — разность абсолютных высот местности или относительное превышение максимальных высот над минимальными. Данный показатель отражает способность марсохода к преодолению препятствий. Из литературы [13] известно, что в настоящее время наибольшая высота препятствия, при которой марсоход может выполнить подъем, составляет 20 см.

Размах рельефа определяется в статье как разность высот двух соседних сеток по направлению движения. Например, на рис. 7 H(i, j) по направлению стрелки вычисляется по следующему соотношению

$$H(i, j) = | H(i + 1, j + 1) - H(i, j).$$
(7)

Оценка опасности рельефа

На основе полученных значений топографических коэффициентов возможно охарактеризовать степень опасности местности для движения марсохода.

Показатель опасности, связанный с углом наклона местности, определяется следующим образом:

$$R_{1}(i,j) = \begin{cases} +\infty \text{ при } \theta > \theta_{th}; \\ 1 + \frac{\theta(i,j)^{2}}{\theta_{th}^{2}}, \end{cases}$$
(8)

где θ_{th} — максимальный угол наклона местности. Показатель опасности, связанный с шероховатостью рельефа, определяется следующим выражением:

$$R_2(i, j) = 1 + \frac{r(i, j)}{r_{\max}},$$
(9)

где r(i, j) — шероховатость рельефа; r_{max} — максимальная шероховатость рельефа.

Показатель опасности, связанный с размахом рельефа, определяется следующим выражением:

$$R_{3}(i,j) = 1 + \frac{H(i,j)^{2}}{H_{\max}^{2}},$$
 (10)

где *H*(*i*, *j*) — размах рельефа; *H*_{max} — максимальный размах рельефа.

Комплексный показатель опасности местности R(i, j) определяется как максимальная величина одного из трех показателей опасности:

$$R(i, j) = \max(R_1(i, j), R_2(i, j), R_3(i, j)), \quad (11)$$

поскольку наибольший показатель опасности является тем фактором, который сильнее всего влияет на движение марсохода.

Данный комплексный показатель выступает в качестве одного из параметров эвристической функции, которая формируется для нового алгоритма Risk Lazy AT.

Алгоритм Risk Lazy AT

Основываясь на приведенной выше теории, в данной работе предлагается новая модификация алгоритма Lazy AT, а именно алгоритм Risk Lazy AT. Разница между ними заключается в том, что в алгоритме Risk Lazy AT в эвристической функции (см. (1)) учитывается фактор опасности рельефа местности.

Таким образом, новая эвристическая функция приобретает следующий вид:

$$f(n) = g(n) + h(n) + c(n),$$
 (12)

где *c*(*n*) — показатель опасности для маршрута марсохода.

Показатель опасности для маршрута марсохода от точки s_0 до точки s_m определяется следующим образом:

$$c(s_0, s_m) = \sum_{(i,j)=s_0}^{(i,j)=s_m} \alpha R(i,j),$$
(13)

Рис. 8. Пример определения показателя опасности для маршрута

Fig. 8. Explanation of the definition of the hazard indicator for trajectory

где s_0 — начальная точка маршрута; s_m — конечная точка маршрута; R(i, j) — показатель опасности рельефа местности на координатной сетке (i, j); α — соответствующие весовые коэффициенты.

Если длина пути через ячейку сетки меньше $\sqrt{2}/2e$, весовой показатель $\alpha = 0,5$. Если длина пути через сетку больше $\sqrt{2}/2e$, весовой показатель $\alpha = 1$. Здесь e — значение шага сетки. Например, для маршрута, показанного на рис. 8, показатель опасности имеет следующий вид:

$$c(s_0, s_2) = R(1, 1) + 1/2(R(2, 1) + R(2, 2)) + R(3, 2) + R(4, 3).$$
(14)

Результаты моделирования

На первом этапе был проведен сравнительный анализ следующих четырех алгоритмов: A*, Basic Theta*, Lazy Theta* и Lazy AT. Результаты моделирования алгоритмов для размеров области на координатной сетке 50×50 м приведены в табл. 1, а для размеров области 100×100 м — в табл. 2. Моделирование проводили при следующих условиях: шаг координатной сетки равен 1 м; число случайных препятствий в области — 10 %, 20 %, 30 %; L, м — длина пути; t, с — время вычисления.

Результаты моделирования показывают, что путь, полученный с помощью алгоритма Lazy AT, является самым коротким из всех. Хотя время вычисления для данного алгоритма больше, чем для алгоритма A*, однако по сравнению со временем вычисления с помощью

Число случайных препятствий	А	*	Basic	Theta*	Lazy '	Theta*	Lazy AT	
	<i>L</i> , м	<i>t</i> , c						
10 %	70,468	0,243	70,225	0,335	70,225	0,274	69,937	0,269
20 %	72,811	0,223	72,277	0,277	71,998	0,263	70,691	0,244
30 %	75,397	0,214	76,176	0,249	74,212	0,242	72,790	0,225

Таблица 2

Размер области на координатной сетке 100×100 м

Число случайных препятствий	А	*	Basic	Theta*	Lazy	Theta*	Lazy AT	
	<i>L</i> , м	<i>t</i> , c						
10 %	142,350	0,480	142,350	0,608	142,350	0,532	140,757	0,518
20 %	149,279	0,468	146,688	0,565	160,305	0,524	145,025	0,506
30 %	153,622	0,440	151,318	0,512	150,590	0,503	148,347	0,469

алгоритмов планирования движения в любом направлении (Basic Theta*, Lazy Theta*) оно оказывается меньше.

При увеличении числа препятствий время работы и путь алгоритма Lazy AT заметно меньше, чем у остальных алгоритмов ПДЛН, что еще раз доказывает эффективность данного алгоритма.

На втором этапе моделирования был проведен сравнительный анализ алгоритмов Lazy Theta* и разработанного алгоритма Risk Lazy AT.

Для моделирования данных алгоритмов в качестве примера была создана трехмерная карта произвольной области поверхности планеты размером 30×30 м. Для построения маршрута без столкновения с препятствиями необходимо увеличить размеры препятствий. Результаты работы алгоритмов Lazy Theta* и Risk Lazy AT с учетом препятствий, нанесенных на координатную сетку, показаны на рис. 9 (см. четвертую сторону обложки).

На рис. 9 обозначено: черный кружок — реальные препятствия; красные кружки — границы виртуальных препятствий (увеличение размеров препятствий необходимо для их гарантированного обхода и предотвращения столкновения); синий кружок — расширенная определенная точка (*s*); зеленая непрерывная линия — путь, построенный с помощью алгоритма Lazy Theta*; красная непрерывная линия — путь, построенный с помощью алгоритма Risk Lazy AT.

Результаты моделирования алгоритмов для размеров области на координатной сетке

	Таблица 3			
Число препятствий	Risk Lazy AT		Lazy Theta*	
	<i>L</i> , м	с	<i>L</i> , м	с
2 %	43,147	1,213	43,401	3,378

30×30 м приведены в табл. 3 (здесь *с* — показатель опасности маршрута).

Анализ полученных результатов показывает, что показатель опасности маршрута (с), построенного с помощью алгоритма Risk Lazy AT, меньше чем аналогичный показатель, полученный с помощью алгоритма Lazy Theta*. Следовательно, первый маршрут безопаснее второго.

На третьем этапе моделирования исследовали применение алгоритма Risk Lazy AT для планирования маршрута движения марсохода по поверхности Марса. На основе реальных данных о рельефе поверхности Марса, полученных из материалов USGS [14], была создана трехмерная карта участка поверхности планеты размером 330×330 м (рис. 10, *a*, см. четвертую сторону обложки) с размером пикселя 0,5 м.

Выберем на топографической карте поверхности Марса участок размером 30×30 м (см. на рис. 10, *а* участок, выделенный розовым цветом). Его укрупненный вид показан на рис. 10, *б*, см. четвертую сторону обложки).

На последнем этапе моделирования с помощью алгоритма Risk Lazy АТ были получены результаты планирования пространственного маршрута движения марсохода по поверхности планеты с учетом рельефа местности, наличия опасностей и препятствий (рис. 11, см. четвертую сторону обложки).

Заключение

Проведен сравнительный анализ известных алгоритмов планирования траектории движения подвижного объекта: A*, Basic Theta*, Lazy Theta*. На основе идеи уменьшения сложности алгоритма и протяженности маршрута предложен новый алгоритм планирования траектории движения подвижного объекта Lazy AT.

На основе особенностей топографии поверхности Марса введен показатель опасности, который включен в качестве эвристического параметра в модификацию нового алгоритма планирования траектории движения Risk Lazy AT.

Проведено моделирование алгоритмов A*, Basic Theta*, Lazy Theta*, Lazy AT и выполнен сравнительный анализ полученных результатов по следующим показателям: L — длина пути; t — время вычисления. Результаты моделирования показали, что алгоритм Lazy AT обеспечивает движение подвижного объекта между исходной и конечной точкой маршрута по кратчайшему пути. По сравнению с алгоритмами Basic Theta* и Lazy Theta* время вычислений с помощью алгоритма Lazy AT также меньше.

Сравнительный анализ результатов моделирования алгоритмов Risk Lazy AT и Lazy Theta* показал, что предложенный новый алгоритм Risk Lazy AT может обеспечить безопасное движение подвижного объекта по поверхности планеты.

На основе реальных данных о рельефе поверхности Марса была разработана цифровая карта поверхности планеты и проведено моделирование пространственного маршрута движения марсохода с помощью нового алгоритма Risk Lazy AT. Полученные результаты подтвердили эффективность предложенного алгоритма для планирования безопасного маршрута движения марсохода по пространственной траектории с учетом рельефа поверхности Марса.

Список литературы

1. Carsten J., Rankin A., Ferguson D., Stentz A. Global path planning on board the mars exploration rovers // Aerospace Conference. 2007. C. 1-11.

2. Тань Л., Фомичев А. Планирование пространственного маршрута полета беспилотного летательного аппарата с использованием методов частично целочисленного линейного программирования // Вестник Московского государственного технического университета им. Н. Э. Баумана. Сер. "Приборостроение". 2016. № 2 (107).

3. Ferguson D., Stentz A. Using interpolation to improve path planning: The Field D* algorithm // Journal of Field Robotics. 2006. Vol. 23, N. 2. P. 79–101.

4. Фомичев А. В., Тань Л. Разработка алгоритма быстрой компенсации погрешностей комплексированной инерциально-спутникой системы навигации малогабаритных беспилотных летательных аппаратов в условиях сложной среды // Наука и образование: научное издание МГТУ им. Н. Э. Баумана. 2015. № 10.

5. **Фомичев А. В., Ван Г.** Сравнительный анализ методов планирования маршрута движения для планетохода // Информационные системы и технологии. 2016.

6. URL: https://ru.wikipedia.org/wiki/Эвристический_алгоритм (дата обращения 25.04.2018).

7. Nash A., Koenig S. Any-angle path planning // AI Magazine. 2013. Vol. 34, N. 4. P. 85–107.

8. Choi S., Lee J. Y., Yu W. Fast any-angle path planning on grid maps with non-collision pruning // Robotics and Biomimetics (ROBIO), 2010 IEEE International Conference. 2010. P. 1051–1056.

9. Uras T., Koenig S. Speeding-Up Any-Angle Path-Planning on Grids // ICAPS. 2015. P. 234-238.

10. Nash A., Daniel K., Koenig S., Aelner A. Theta*: anyangle path planning on grids // AAAI. 2007. Vol. 7. P. 1177–1183.

11. **Mendonca P., Goodwin S.** C-Theta*: Cluster Based Path-Planning on Grids // Computational Science and Computational Intelligence (CSCI), 2015 International Conference. 2015. P. 605–608.

12. Nash A., Koenig S. and Tovey C. Lazy theta*: Any-angle path planning and path length analysis in 3d //Third Annual ymposium on Combinatorial Search. 2010.

13. URL: https://en.wikipedia.org/wiki/Spirit_(rover) (дата обращения 25.04.2018).

14. URL: https://astrogeology.usgs.gov/search/map/Phobos/ MarsExpress/HRSC/Phobos_ME_HRSC_DEM_Global_2ppd (дата обращения 25.04.2018).

Development of a Non-hazardous Path Planning Algorithm for Mars Rover in Real Terrain Enviroment

G. Wang, guoyan@yandex.ru, **A. V. Fomichev**, a.v.fomichev@bmstu.ru, Bauman Moscow State Technical University, Moscow, 105005, Russian Federation

> Corresponding author: Fomichev Aleksei V., Ph. D., Associate Professor, Bauman Moscow State Technical University, Moscow, 105005, Russian Federation, e-mail: a.v.fomichev@bmstu.ru

Abstract

In order to fulfill the corresponding task successfully, a crucial issue should be addressed is the path planning for the exploration of the Mars surface owing to the environmental features of the tough terrain. Traditional path planning algorithms, such as the A^* algorithm and the improved A^* algorithm — the algorithm D^* and the Field D^* , which have been successfully implemented on the planetary rover during the expeditions of the Moon and Mars, have the problem of finding the shortest optimal path. One of the more effective algorithms derived from the modified A* refers to the Basic Theta* or the Lazy Theta* algorithms, which are faster any-angle path planning. Additionally, the algorithms can find shorter routes. In this paper, derived from a comprehensive comparison of the existing algorithms (A*, Basic Theta* and Lazy Theta*), a novel modification of the Lazy AT methodology is proposed to reduce the calculation time and obtain a shorter path. Based on the analysis of the surface feature of the Mars topography, the corresponding safety indicator is discussed. The principal hazards of the wheeled vehicles during the exploration on the surface of the Mars are the slopes and the obstacles. According to the requirements for avoiding obstacles as well as the exploration stability of the Mars rover in the period of the exploration, the following topographic coefficients have been chosen to develop the hazard indicator, i.e., the inclination angle of the terrain, the surface roughness and the height difference of the terrain. In addition, to obtain a safe trajectory in algorithm Lazy AT on the Mars surface, the terrain hazard indicator (risk indicator) for the modification of the Risk Lazy AT algorithm is also proposed in this paper. The comparing analysis modeling results of the Risk Lazy AT and Lazy Theta* has shown that our proposed algorithm Risk Lazy AT can guarantee the safety movement of a mobile object during the exploration on the surface of the planet. In light of the real-world surface features of the Mars terrain, the digital map of the planet's surface has been developed and the spatial routing of the rover has been tested with our novel proposed algorithm, so-called Risk Lazy AT.

Keywords: algorithm A*, Any-Angle Path Planning, algorithm Basic theta*, algorithm Lazy Theta*, algorithm Lazy AT, algorithm Risk Lazy AT, terrain hazard indicator

For citation:

Wang G., Fomichev A. V. Development of a Non-hazardous Path Planning Algorithm for Mars Rover in Real Terrain Enviroment, *Mekhatronika, Avtomatizatsiya, Upravlenie,* 2018, vol. 19, no. 11, pp. 734–743.

DOI: 10.17587/mau.19.734-743

References

1. **Carsten J., Rankin A., Ferguson D., Stentz A.** Global path planning on board the mars exploration rovers. In Aerospace Conference, 2007 IEEE (pp. 1–11).

2. Tan Liguo, Fomichev A. V. Planirovanie prostranstvennogo marshruta poleta bespilotnogo letatelqnogo appalata s ispolqzovaniem metodov chactichno celochislennogo linejnogo programmirovaniya (The spatial flight route planning of unmanned aerial vehicles using the methods of mixed-integer linear programming), Vectnik moskovckogo gosudarstvennogo technicheckogo universiteta im. N. E. Baumana. Seriya "Priborostroenie", 2016 (in Russian).

3. Ferguson D., Stentz A. Using interpolation to improve path planning: The Field D* algorithm, *Journal of Field Robotics*, 23.2 (2006): 79–101.

4. Fomichev A. V., Tan Liguo. Razrabotka algoritma byhstroj kompensacii pogreshnostej kompleksirovannoj inercialqno-sputnikoj sistemyh navigacii malogabaritnyhch bespilotnyhch letatelqnyhch apparatov v usloviyach slozhnoj sredyh (Development of an algorithm for rapid error-compensation for small unmanned aerial vehicles with a complex inertial satellite navigation system in a complex environment), Nauka i obrazovanie: nauchnoe izdanie MGTU im. N. E. Baumana, 2015, no. 10 (in Russian). 5. Fomichev A. V., Wang Guoyan. Sravnitelqnyhq analiz metodov planirovaniya marshruta dvizheniya dlya planetochoda (Comparison analysis of path-planning methodologies for planetary rover), Informacionnyhe Sistemyh i Technologii, 2016 (in Russian).

6. Available at: https://ru.wikipedia.org/wiki/Эвристический_алгоритм/ (date of access 25.04.2018).

7. Nash A., Koenig S. Any-angle path planning, *AI Maga*zine, 2013, vol.34, no. 4., pp. 85–107.

8. Choi S., Lee J. Y., Yu W. Fast any-angle path planning on grid maps with non-collision pruning. Robotics and Biomimetics (ROBIO), *2010 IEEE International Conference on. IEEE*, 2010.

9. Uras T., Koenig S. Speeding-Up Any-Angle Path-Planning on Grids, *ICAPS*, 2015, pp. 234–238.

10. Nash A., Daniel K., Koenig S. Aelner A. Theta*: Any angle path planning on grids, *Proceedings AAAI Conference on Artificial Intelligence*, 2007, vol. 7, pp. 1177–1183.

11. Mendonca P., Goodwin S. C-Theta*: Cluster based Path-Planning on Grids. International Conference on Computational Science and Computational Intelligence.2015. Mendonca, Pramod, and Scott Goodwin. "C-Theta*: Cluster Based Path-Planning on Grids." Computational Science and Computational Intelligence (CSCI), 2015 International Conference on. IEEE, 2015, pp. 605–608.

12. Nash A., Koenig S., Tovey C. Lazy theta*: Any-angle path planning and path length analysis in 3d, *Third Annual Symposium on Combinatorial Search*, 2010.

13. Available at: https://en.wikipedia.org/wiki/Spirit_ (rover) / (date of access 25.04.2018).

14. **Available at:** https://astrogeology.usgs.gov/search/map/ Phobos/MarsExpress/HRSC/Phobos_ME_HRSC_DEM_ Global_2ppd / (date of access 25.04.2018). В. М. Солдаткин, д-р техн. наук, зав. кафедрой, w-soldatkin@mail.ru, В. В. Солдаткин, д-р техн. наук, доц., А. В. Никитин, канд. техн. наук, ст. преподаватель, Е. О. Арискин, аспирант,

Казанский национальный исследовательский технический университет им. А. Н. Туполева — КАИ

Система измерения воздушных параметров вертолета с неподвижным приемником потока, ионно-меточными и аэрометрическими измерительными каналами*

Показано, что известные ограничения на измерение воздушных параметров на борту вертолета обусловлены значительными аэродинамическими возмущениями, вносимыми индуктивными потоками вихревой колонны несущего винта. Это определяет необходимость создания средств измерений, учитывающих особенности аэродинамики и динамики полета вертолета.

Известным направлением преодоления этих ограничений является использование для измерения информации аэродинамического поля вихревой колонки несущего винта и ее восприятия с помощью неподвижного многофункционального аэрометрического приемника. Однако необходимость защиты большого числа трубок полного давления, установленных в проточном канале многофункционального аэрометрического приемника, и жесткие требования к идентичности и стабильности характеристик большого числа аэрометрических каналов усложняют конструкцию, снижают надежность, повышают стоимость, сдерживают использование системы измерения воздушных параметров на вертолетах различных классов и назначения.

Рассмотрены принципы построения, функциональная схема, особенности восприятия первичной информации системы измерения воздушных параметров вертолета с неподвижным приемником потока, ионно-меточными и аэрометрическими измерительными каналами. Приведены алгоритмы обработки первичной информации на различных этапах и режимах полета, в том числе: на стоянке до запуска силовой установки и при вращении несущего винта, при рулении и маневрировании по земной поверхности, на режимах взлета и посадки, при полетах на малых скоростях, а также на скоростях полета, когда неподвижный приемник первичной информации выходит из зоны вихревой колонны несущего винта, с использованием ионно-меточных и аэрометрических измерительных каналов.

Показано, что предложенные подходы к построению моделей и алгоритмов обработки первичной информации системы измерения воздушных параметров вертолета с ионно-меточными и аэрометрическими измерительными каналами позволяют определить скорость и направление ветра, а также высотно-скоростные параметры движения относительно окружающей среды и параметров атмосферы в широком диапазоне эксплуатации вертолета. Все это определяет конкурентные преимущества предлагаемой системы при решении задач эксплуатации и обеспечения безопасности полетов вертолетов различного класса и назначения.

Ключевые слова: вертолет, воздушные параметры, измерение, система, неподвижный приемник, ионно-меточные и аэрометрические измерительные каналы, функциональная схема, конструктивные особенности, алгоритмы обработки информации

Введение

В авиации и в различных отраслях промышленности, в энергетике и строительстве широкое применение находят вертолеты различных классов и назначения, эксплуатация которых происходит в приземном возмущенным слое атмосферы в условиях значительных ветровых возмущений. При пилотировании и решении задачи обеспечения безопасности полета вертолета необходима достоверная информация о величине и составляющих вектора истинной воздушной скорости, аэродинамических углах атаки и скольжения, барометрической высоте и приборной скорости, скорости и угле направления вектора ветра относительно продольной оси вертолета, атмосферном давлении и плотности, о других воздушных параметрах, определяющих аэродинамику и динамику движения вертолета относительно окружающей воздушной среды на всех этапах и режимах полета.

При работе силовой установки на стоянке, при рулении и маневрировании по земной поверхности (стартовые режимы), при взлете и наборе высоты, снижении, висении и посадке (взлетно-посадочные режимы), при полете с малыми скоростями на работу бортовых средств измерения воздушных параметров вертолета оказывают существенное влияние индуктивные потоки вихревой колонны несущего винта, ограничивающие использование традиционных средств измерения [1—3]. Это обусловливает необходимость разработки систем измерения воздушных параметров, максимально учитывающих аэродинамику и динамику полета вертолета.

^{*} Работа выполнена по гранту РФФИ № 18-08-00264.

Построение системы измерения воздушных параметров вертолета на основе неподвижного многофункционального аэрометрического приемника

Широкие возможности по преодолению указанных ограничений открываются при использовании для целей измерения воздушных параметров информации аэродинамического поля вихревой колонны несущего винта [4], воспринимаемой неподвижным многофункциональным аэрометрическим приемником, выходные сигналы которого регистрируются аэрометрическими измерительными каналами. Обработка выходных сигналов в соответствии с разработанными алгоритмами позволяет определить воздушные параметры вертолета в широком диапазоне рабочих скоростей и высот полета [5].

В работах [6, 7] раскрываются принципы построения и алгоритмы работы бортовой системы измерения воздушных параметров на стоянке, стартовых и взлетно-посадочных режимах, при полете вертолета на малых скоростях, в которой для целей измерения используется информация о параметрах вектора скорости результирующего воздушного потока вихревой колонны несущего винта, воспринимаемая неподвижным многофункциональным аэрометрическим приемником [5].

Многофункциональный аэрометрический приемник (рис. 1) содержит неподвижный многоканальный проточный аэрометрический приемник l, выполненный в виде двух разнесенных по высоте экранирующих дисков 2 и 3, между внутренними профилированными поверхностями которых в азимутальной плоскости под одинаковыми углами расположены трубки полного давления 4 для забора давлений P_i , определяющих модуль W и угол направления ψ вектора ветра **W** или параметры вектора V_{Σ} истинной воздушной скорости в горизонтальной плоскости.

На внутренних поверхностях экранирующих дисков 2 и 3 расположены кольцевые каналы 6 для забора дросселированного статического давления $P_{\rm ст.д}$, формируемого в проточном канале приемника 1. На внутренних профилированных поверхностях экранирующих дисков 2 и 3 расположены отверстия 5 для забора давлений $P_{\alpha i}$ и $P_{\alpha i-1}$, определяющих угол ветра или вектора истинной воздушной скорости $\mathbf{V}_{\rm R} = -\mathbf{V}$ в вертикальной плоскости.

Таким образом, воспринимаемые неподвижным многоканальным проточным аэрометрическим приемником первичные информативные сигналы в виде давлений P_i , $P_{ct,R}$, P_{ai} , P_{ai-1} позволяют определить модуль W и угловое положение вектора скорости ветра **W** или вектора истинной воздушной скорости **V**_в относительно продольной оси вертолета или продольные и боковые составляющие вектора скорости ветра или вектора истинной воздушной скорости на стоянке до запуска силовой установки и раскрутки трансмиссии и на режимах полета, когда неподвижный многоканальный проточный аэрометрический приемник находится вне зоны действия вихревой колонны несущего винта.

Для восприятия аэрометрической информации на стоянке при запуске силовой установки и вращении несущего винта, на режиме руления и маневрирования вертолета по земле и на взлетно-посадочных режимах, при полете на малых скоростях, когда неподвижный многоканальный проточный аэрометрический приемник *1* находится в створе вихревой колонны несущего винта, на наружной поверхности экранирующего диска *3* установлен дополнительный аэрометрический приемник 7 в виде полусферы с диаметром, равным диаметру верхнего экра-

Рис. 1. Конструктивная схема неподвижного многофункционального аэрометрического приемника

Fig. 1. Constructive scheme of motionless multifunctional aerometric receiver

нирующего диска. На поверхности полусферы на оси симметрии расположено отверстие, являющееся приемником полного давления $P_{\Pi\Sigma}$ результирующего воздушного потока вихревой колонны. В плоскости, параллельной плоскости симметрии вертолета, под углом φ_{01} к оси симметрии симметрично расположены отверстия, являющиеся приемниками давлений P₁ и Р₂. В плоскости, перпендикулярной плоскости симметрии вертолета, под углом ϕ_{01} к оси симметрии расположены отверстия, являющиеся приемниками давлений Р₃ и Р₄. Перпендикулярно оси симметрии сферического приемника на его поверхности, например по окружности, расположены отверстия, объединенные в общем канале, являющемся приемником статического давления $P_{\text{ст}\Sigma}$ результирующего воздушного потока вихревой колонны.

При рулении и маневрировании по земле и на взлетно-посадочных режимах, при полете с малыми скоростями за меру величин составляющих вектора скорости ветра **W** и вектора истинной воздушной скорости $V_{\rm B}$ вертолета принимается угловое положение вектора скорости V_{Σ} результирующего воздушного потока вихревой колонны несущего винта, определяемое углами φ_1 и φ_2 , которые регистрируются ортогонально расположенными приемниками давлений P_1 и P_2 , P_3 и P_4 .

Давления P_1 и P_2 , P_3 и P_4 , $P_{\Pi\Sigma}$ и $P_{CT\Sigma}$, воспринимаемые дополнительным аэрометрическим приемником 3, подключены ко входам пневмоэлектрических преобразователей, выходы которых через последовательно соединенные мультиплексор и аналогово-цифровой преобразователь подключены к микропроцессору. На вход микропроцессора также подключены выходы приемника температуры торможения результирующего набегающего воздушного потока вихревой колонны несущего винта и выходы пневмоэлектрических преобразователей давлений P_i , $P_{\alpha i}$, $P_{\alpha i-1}$, $P_{\text{ст.д.}}$, воспринимаемых неподвижным многоканальным проточным аэрометрическим приемником 1. Выход микропроцессора является выходом бортовой системы по модулю и углу направления или продольной и боковой составляющим вектора скорости ветра W и истинной воздушной скорости V_в на стоянке, при рулении и маневрировании по земной поверхности, при взлете и посадке, в области малых и околонулевых скоростей полета и на режиме висения, когда неподвижный многофункциональный аэрометрический приемник находится в зоне вихревой колонны несущего винта.

Однако построение неподвижного многоканального аэрометрического приемника на основе многофункционального проточного аэрометрического приемника связано с необходимостью защиты большого числа (6 или 8) трубок полного давления, установленных в его проточном канале, от обледенения, попадания пыли и влаги в реальных условиях эксплуатации вертолета. Многоканальная схема преобразования воспринимаемых давлений обусловливает жесткие требования к идентичности и стабильности характеристик аэрометрических измерительных каналов. Все это усложняет конструкцию, снижает надежность, повышает стоимость, сдерживает применение системы измерения воздушных параметров на вертолетах различного класса и назначения.

Построение системы измерения воздушных параметров вертолета с неподвижным приемником потока, ионно-меточными и аэрометрическими измерительными каналами

Для устранения указанных недостатков системы измерения воздушных параметров вертолета на основе неподвижного многофункционального приемника и аэрометрических измерительных каналов предлагается система, функциональная схема которой приведена на рис. 2 [8].

В этом случае система измерения воздушных параметров выполняется по двухканальной схеме и включает ионно-меточный и аэрометрический измерительные каналы.

В качестве ионно-меточного канала используется панорамный ионно-меточный датчик аэродинамического угла и воздушной скорости [9], включающий плату 1 с системой приемных электродов 3, в центре которой установлен искровой разрядник 2, подключенный к генератору меток (ГМ) 4. При подаче на высоковольтный разрядник 2 высоковольтного импульса от генератора меток формируется ионная метка с явно выраженным электростатическим зарядом, которая движется совместно с набегающим воздушным потоком, приобретая его скорость V и направление α . При приближении заряженной ионной метки к приемным электродам 3, расположенным

Рис. 2. Функциональная схема системы измерения воздушных параметров вертолета с аэрометрическими и ионно-меточными измерительными каналами

Fig. 2. Functional scheme of measuring system of air parameters of helicopter with aerometric and ion-mark measuring channels

на окружности радиусом R, на них наводятся электростатические заряды, величина которых зависит от расстояния R от точки генерации ионной метки до приемного электрода и углового положения ψ траектории движения метки.

На рис. 3 показана конструктивная схема датчика первичной информации системы измерения воздушных параметров вертолета на основе ионно-меточных и аэрометрических измерительных каналов.

Перпендикулярно системе приемных электродов 1 панорамного ионно-меточного датчика аэродинамического угла и воздушной скорости на стойках 3 установлен неподвижный полусферический аэрометрический приемник 2, на поверхности которого расположены отверстия для восприятия давлений, несущих информацию о векторе скорости V_{Σ} результирующего воздушного потока вихревой колонны несущего винта. Воспринимаемые давления по пневмоканалам через стойки 3 передаются ко

Рис. 3. Конструктивная схема датчика первичной информации Fig. 3. Constructive scheme of the sensor primary informatio

входам пневмоэлектрических преобразователей, расположенных внутри панорамного ионно-меточного датчика аэродинамического угла и воздушной скорости.

Для восприятия информации результирующего воздушного потока вихревой колонны несущего винта на поверхности полусферического аэрометрического приемника 11 (см. рис. 2) расположены отверстия 12 для забора полного давления $P_{\Pi\Sigma}$ результирующего воздушного потока вихревой колонны несущего винта, отверстия 13, 14 и 15, 16 для забора давлений P₁, P₂ и P₃, P₄, определяющих углы φ₁ и φ_2 положения вектора скорости \mathbf{V}_{Σ} относительно оси симметрии полусферического приемника в плоскости, параллельной продольной оси вертолета, и в плоскости, перпендикулярной продольной оси вертолета, а также отверстия 17 для забора статического давления $P_{\text{ст}\Sigma}$ результирующего воздушного потока вихревой колонны несущего винта.

Воспринимаемые давления $P_{\Pi\Sigma}$, P_1 , P_2 и P_3 , P_4 , $P_{CT\Sigma}$ с помощью пневмоэлектрических преобразователей перепада давлений 18 и преобразователя абсолютного давления 19 преобразуются в электрические сигналы, которые через канал 20 аналогово-цифрового преобразования, включающего мультиплексор 21 и АЦП 22 подаются в вычислительное устройство 10, на выходе которого по разработанным алгоритмам формируются цифровые сигналы по воздушным сигналам вертолета.

Алгоритмы обработки информации измерительных каналов системы измерения воздушных параметров на различных этапах и режимах полета вертолета

На стоянке до запуска силовой установки входными сигналами ионно-меточного измерительного канала являются скорость $W_{\rm r}$ и угол направления ψ вектора горизонтального ветра $\mathbf{W}_{\rm r}$ относительно продольной оси вертолета, которые необходимы для предотвращения опрокидывания вертолета на бок или на хвостовую балку [10]. При попадании траектории ионной метки в *i*-й грубый канал текущее значение измеряемого угла ψ направления горизонтального ветра $W_{\rm r}$ определяется как

$$\psi = i\alpha_0 + \alpha_p, \tag{1}$$

где α_0 — угол, охватывающий рабочий сектор грубого канала отсчета (при $i_{max} = 4$, $\alpha_0 = 90^\circ$); i — номер рабочего сектора (i = 1, 4); α_p — значение измеряемого угла в пределах *i*-го рабочего сектора.

Сигналы $U(\alpha_i)$, пропорциональные синусу Usin α_i и косинусу Ucos α_i измеряемого угла ψ в *i*-м рабочем секторе грубого канала и снимаемые с блока предварительных усилителей, обрабатываются в вычислительном устройстве, на выходе которого формируется цифровой код, связанный со значением α_p измеряемого угла в пределах рабочего сектора точного канала соотношением

$$\alpha_{\rm p} = \arctan \frac{U \sin \alpha_i}{U \cos \alpha_i},\tag{2}$$

где α_i — текущее значение измеряемого угла в пределах *i*-го рабочего сектора.

При работе канала измерения скорости вектора горизонтального ветра \mathbf{W}_{Γ} формируется интервал времени τ_W пролета ионной метки расстояния R от точки генерации θ ионной метки до окружности с приемными электродами. В соответствии с интервалом времени τ_W в вычислительном устройстве вырабатывается цифровой код N_W , пропорциональный скорости горизонтального ветра

$$W_{\rm r} = \frac{R}{\tau_W}.$$
 (3)

На стоянке при запуске силовой установки и вращении несущего винта, при рулении и маневрировании по земной поверхности, на взлетно-посадочных режимах в работу включается второй — аэрометрический — канал, выполненный на основе неподвижного полусферического аэрометрического приемника, жестко установленного над системой приемных электродов ионно-меточного датчика аэродинамического угла и воздушной скорости.

По полному $P_{\Pi\Sigma}$ и статическому $P_{CT\Sigma}$ давлениям и температуре $T_{T\Sigma}$ торможения результирующего воздушного потока вихревой колонны несущего винта, воспринимаемой приемником температуры торможения, установленным на фюзеляже в зоне действия вихревой колонны несущего винта, можно определить модуль вектора скорости V_{Σ} результирующего воздушного потока вихревой колонны несущего винта, используя соотношение [6]

$$V_{\Sigma} = 44,826 \sqrt{\frac{T_{\Sigma} \left[\left(\frac{P_{\Pi\Sigma} - P_{CT\Sigma}}{P_{CT\Sigma}} + 1 \right)^{0,2857143} - 1 \right]}{\left(\frac{P_{\Pi\Sigma} - P_{CT\Sigma}}{P_{CT\Sigma}} + 1 \right)^{0,2857143}}, \quad (4)$$

где параметры, входящие в формулу (4), имеют размерности в единицах системы СИ.

Проекции вектора V_{Σ} скорости результирующего воздушного потока вихревой колонны несущего винта на оси связанной системы координат определяются как

$$V_{\Sigma x} = V_{\Sigma} \sin \varphi_{1}; V_{\Sigma y} = V_{\Sigma} \cos \varphi_{1} \cos \varphi_{2};$$

$$V_{\Sigma z} = V_{\Sigma} \sin \varphi_{2}, \qquad (5)$$

где φ_1 и φ_2 — углы, определяющие положение вектора скорости V_{Σ} результирующего воздушного потока вихревой колонны несущего винта относительно осей полусферического аэрометрического приемника, которые определяются по давлениям P_1 , P_2 и P_3 , P_4 .

Используя соотношения для определения давления на поверхности полусферического аэрометрического приемника, приведенные в работе [11], углы φ_1 и φ_2 , определяющие положение вектора скорости V_{Σ} результирующего набегающего воздушного потока вихревой колонны несущего винта, будут определяться соотношениями

$$\phi_{1} = \frac{1}{2} \arcsin\left(\frac{4}{9\sin 2\phi_{01}} \frac{P_{1} - P_{2}}{P_{\Pi\Sigma} - P_{CT\Sigma}}\right); \quad (6)$$

$$\phi_{2} = \frac{1}{2} \arcsin\left(\frac{4}{9\sin 2\phi_{02}} \frac{P_{3} - P_{4}}{P_{\Pi\Sigma} - P_{CT\Sigma}}\right),$$

где φ_{10} и φ_{20} — углы расположения отверстий для забора давлений P_1 , P_2 и P_3 , P_4 .

Поскольку неподвижный панорамный ионно-меточный датчик аэродинамического угла и истинной воздушной скорости с установленным над его системой приемных электродов полусферическим аэрометрическим приемником расположен на фюзеляже на определенном радиус-векторе \mathbf{R} от центра масс вертолета, то при вращении вертолета относительно центра масс имеет место кинематическое искажение вектора скорости \mathbf{V} воздушного потока, набегающего на аэрометрический приемник, которое определяется векторным уравнением вида

$$\mathbf{V}_{\kappa} = \mathbf{V} + \boldsymbol{\omega} \times \mathbf{R}(x, y, z), \tag{7}$$

где V_{κ} — кинематически искаженный вектор скорости набегающего воздушного потока в месте установки неподвижного комбинированного аэрометрического приемника; $\omega(\omega_x, \omega_y, \omega_z)$ вектор угловой скорости вращения вертолета относительно центра масс; ω_x , ω_y , ω_z — угловые скорости вращения вертолета относительно осей связанной системы координат; *x*, *y*, *z* — координаты места установки неподвижного панорамного меточного датчика аэродинамического угла и воздушной скорости с установленным над его системой приемных электродов полусферическим аэрометрическим приемником в связанной системе координат, центр которой находится в центре масс вертолета.

Проекции вектора скорости V_{κ} на оси связанной системы координат будут определяться системой уравнений вида

$$V_{\kappa x} = V_x + (\omega_y z - \omega_x y); V_{\kappa y} = V_y + (\omega_z x - \omega_x z);$$

$$V_{\kappa z} = V_z + (\omega_x y - \omega_y x), \qquad (8)$$

где V_x , V_y , V_z — проекции вектора истинной воздушной скорости $\mathbf{V}_{\rm B}$ на оси связанной системы координат.

Продольная и боковая составляющие V_x , V_z и модуль V_B вектора истинной воздушной скорости \mathbf{V}_B , углы атаки α и скольжения β вертолета, статическое давление P_H и барометрическая высота H будут определяться следующими соотношениями [6]:

$$\begin{split} V_{x} &= V_{\Sigma} \sin \left[\frac{1}{2} \arcsin \left(\frac{4}{9 \sin 2\varphi_{01}} \frac{P_{1} - P_{2}}{P_{\Pi\Sigma} - P_{\text{cT}\Sigma}} \right) \right] \times \\ &\times \cos \left[\frac{1}{2} \arcsin \left(\frac{4}{9 \sin 2\varphi_{02}} \frac{P_{3} - P_{4}}{P_{\Pi\Sigma} - P_{\text{cT}\Sigma}} \right) \right] - \\ &- K_{ix} V_{i0} - (\omega_{y} z - \omega_{x} y); \\ V_{z} &= V_{\Sigma} \sin \left[\frac{1}{2} \arcsin \left(\frac{4}{9 \sin 2\varphi_{02}} \frac{P_{3} - P_{4}}{P_{\Pi\Sigma} - P_{\text{cT}\Sigma}} \right) \right] - \\ &- K_{iz} V_{i0} - (\omega_{x} y - \omega_{y} x); \\ V_{B} &= \sqrt{V_{x}^{2} + V_{y}^{2} + V_{z}^{2}}; \\ (9) \\ \alpha &= \arctan \frac{V_{y}}{V_{x}}; \beta = \arcsin \frac{V_{z}}{2}; \\ P_{H} &= P_{\text{cT}\Sigma} - K_{p} \frac{\rho_{\Sigma} V_{\Sigma}^{2}}{2}; \\ \rho_{\Sigma} &= \frac{P_{\text{cT}\Sigma}}{287,05287T_{\text{T}\Sigma}} \left(\frac{P_{\Pi\Sigma} - P_{\text{cT}\Sigma}}{P_{\text{cT}\Sigma}} + 1 \right)^{0,2857143}; \\ H &= \frac{T_{0}}{\tau} \left[1 - \left(\frac{P_{H}}{P_{0}} \right)^{\tau R} \right], \end{split}$$

Мехатроника, автоматизация, управление, Том 19, № 11, 2018

где $P_0 = 101$ 325 Па и $T_0 = 288,15$ К — абсолютное давление и абсолютная температура воздуха на высоте H = 0 стандартной атмосферы; *R* — удельная газовая постоянная воздуха $(R = 287,05287 \ Дж/кг·К); \tau$ — температурный градиент ($\tau = 0,0065$ К/м); K_p — безразмерный коэффициент, определяющий связь P_{стΣ} и P_H, который определяется при летных испытаниях для конкретного типа вертолета и места установки на фюзеляже неподвижного панорамного ионно-меточного датчика аэродинамического угла и воздушной скорости с установленным на нем осесимметричным полусферическим аэрометрическим приемником; K_{iv} — безразмерный коэффициент, определяемый аналогично K_{ix} и K_{iz}.

Используя информацию о модуле $V_{\rm n}$ и угле сноса $\psi_{\rm c}$ вектора путевой скорости $\mathbf{V}_{\rm n}$ от доплеровского измерителя скорости и угла сноса (ДИСС) и о составляющих вектора истинной воздушной скорости $\mathbf{V}_{\rm B}$ от аэрометрического канала, продольную W_x , боковую W_z составляющие, модуль $W_{\rm r}$ и направление ψ горизонтального вектора ветра $\mathbf{W}_{\rm r}$ на взлетно-посадочных режимах вертолета будем определять соотношениями [6]:

$$W_{x} = |V_{x}| - V_{\pi} \cos \psi_{c}; \ W_{z} = |V_{z}| - V_{\pi} \sin \psi_{c};$$

$$\psi = \beta + \psi_{c}; \ W_{\Gamma} = \sqrt{W_{x}^{2} + W_{z}^{2}}.$$
(10)

Таким образом, на основе информации, воспринимаемой неподвижным датчиком первичной информации, выполненным в виде неподвижного панорамного ионно-меточного датчика аэродинамического угла и воздушной скорости с установленным над его системой приемных электродов полусферическим аэрометрическим приемником, в вычислительном устройстве, выполненном в виде вычислителя, реализующего алгоритмы (1)-(3), (5), (6), (9), (10), определяются параметры вектора скорости ветра и другие воздушные параметры вертолета, на стоянке при запуске силовой установки и вращении несущего винта, при рулении и маневрировании по земной поверхности, при взлете и наборе высоты, при полете на малых скоростях и на режимах снижения, на режимах висения и посадки вертолета.

При скоростях полета, когда неподвижный датчик первичной информации системы измерения воздушных параметров вертолета с аэрометрическими и ионно-меточными измерительными каналами выходит из зоны вихревой колонны несущего винта, для определения воздушных параметров вертолета используется информация аэрометрического и ионно-меточного измерительных каналов.

По информации ионно-меточного измерительного канала по соотношениям, аналогичным выражениям (1)—(3), определяются истинная воздушная скорость $V_{\rm B}$ и угол скольжения β :

$$V_{\rm B} = \frac{R}{\tau_V}; \beta = i\alpha_0 + \alpha_{\rm p}; \alpha_{\rm p} = \operatorname{arctg} \frac{U \sin \alpha}{U \cos \alpha}, (11)$$

где τ_V — время пролета ионной метки от разрядника до окружности с приемными электродами при данной истинной воздушной скорости $V_{\rm B}$.

Для определения барометрической высоты Hна наружной поверхности платы l с приемными электродами (см. рис. 2) ионно-меточного датчика аэрометрического угла и истинной воздушной скорости располагается отверстиеприемник для забора статического давления P_H набегающего воздушного потока. Тогда барометрическая высота H будет определяться известным соотношением [5]

$$H = \frac{T_0}{\tau} \left[1 - \left(\frac{P_H}{P_0} \right)^{\tau R} \right].$$

Вертикальная воздушная скорость V_y будет определяться путем вычисления производной по времени от барометрической высоты:

$$Vy = \frac{dH}{dt} = \frac{H(t_i) - H(t_{i-1})}{t_i - t_{i-1}},$$
 (12)

где t_i и t_{i-1} — текущий и предшествующие моменты времени, в которых проведены вычисления барометрической высоты. Возможно использование алгоритма вычисления вертикальной воздушной скорости по большему числу значений барометрической высоты, полученных через фиксированный интервал времени τ , например, вида

$$V_{y} = \frac{1}{4\tau} [H(t_{i}) - H(t_{i} + 2\tau) + H(t_{i} + \tau) - H(t_{i} + 3\tau)].$$
(13)

Тогда угол атаки вертолета можно определить, используя соотношение

$$\alpha = \operatorname{arctg} \frac{V_y}{V_x} = \frac{V_y}{V_B \cos\beta}.$$
 (14)

Таким образом, установка на наружной поверхности платы приемных электродов ионнометочного датчика аэродинамического угла отверстия-приемника статического давления набегающего воздушного потока позволяет вычислить воздушные параметры вертолета при выходе датчика первичной информации из зоны вихревой колонны несущего винта.

Заключение

Предложенные подходы к построению, модели и алгоритмы обработки первичной информации системы измерения воздушных параметров вертолета с аэрометрическими и ионно-меточными измерительными каналами позволяют определить параметры вектора ветра и высотно-скоростные параметры вертолета в широком диапазоне эксплуатации, в том числе на стоянке, стартовых и взлетно-посадочных режимах, при полете в диапазоне малых и полетных скоростях.

Применение рассмотренной системы измерения воздушных параметров на различных классах вертолетов позволяет повысить безопасность эксплуатации и эффективность решения полетных задач.

Список литературы

1. Руководство по летной эксплуатации вертолета Ми-8 (издание 4). М.: Авторитет, 1996. 554 с.

2. Козицин В. К., Макаров Н. Н., Порунов А. А., Солдаткин В. М. Анализ принципов построения систем измерения воздушных сигналов вертолета // Авиакосмическое приборостроение. 2003. № 10. С. 2—13.

3. **Солдаткин В. М.** Методы и средства измерения аэродинамических углов летательных аппаратов. Казань: Изд-во Казан. гос. техн. ун-та, 2001. 448 с.

4. Солдаткин В. В. Аэрометрическая система измерения малых воздушных скоростей вертолета на основе информации о положении вихревой колонны несущего винта // Известия вузов. Авиационная техника. 2009. № 4. С. 52—56.

5. Солдаткин В. В. Система воздушных сигналов вертолета на основе неподвижного аэрометрического приемника и информации аэродинамического поля вихревой колонны несущего винта. Казань: Изд-во Казан. гос. техн. ун-та, 2012. 284 с.

6. Никитин А. В., Солдаткин В. В., Солдаткин В. М. Система измерения параметров вектора ветра на стартовых и взлетно-посадочных режимах вертолета // Мехатроника, автоматизация, управление. 2013. № 6. С. 64—70.

7. Никитин А. В., Солдаткин В. М. Система измерения параметров вектора ветра и истинной воздушной скорости на борту вертолета // Датчики и системы. 2015. № 4. С. 48—54.

8. Никитин А. В., Арискин Е. О., Солдаткин В. В., Солдаткин В. М. Бортовая система измерения параметров вектора ветра на стоянке, стартовых и взлетно-посадочных режимах вертолета с аэрометрическими и ионно-меточными измерительными каналами // Известия вузов. Авиационная техника. 2015. № 4. С. 87—93.

9. Ганеев Ф. А., Солдаткин В. М. Ионно-меточный датчик аэродинамического угла и воздушной скорости с логометрическими информативными сигналами и интерполяционной схемой обработки // Известия вузов. Авиационная техника. 2010. № 3. С. 46—50.

10. Ерусалимский М. А., Егоров В. Н. Экипажам вертолета информационную поддержку // Авиасоюз. № 2. С. 24—26.

11. Петунин А. Н. Методы и техника измерения параметров газового потока (приемники давления и скоростного напора). М.: Машиностроение, 1972. 332 с.

Measuring System of Air Parameters of the Helicopter with a Stationary Receiver a Flow, Ion-Label and Aerometric Measurement Channels

V. M. Soldatkin, w-soldatkin@mail.ru, V. V. Soldatkin, w-soldatkin@mail.ru,

A. V. Nikitin, nikitin.rf@mail.ru, E. O. Ariskin, ariskineo@mail.ru,

Kazan National Research Technical University named after A. N. Tupolev — KAI, Kazan, 420111, Russian Federation

Corresponding author: Soldatkin Vladimir M., D. Sc., Professor, Head of Chair, Kazan National Research Technical University named after A. N. Tupolev – KAI, Kazan, 420111, Russian Federation, e-mail: w-soldatkin@mail.ru

Accepted on June 18, 2018

Abstract

It is shown that the known limitations on the measurement of air parameters on board the helicopter due to significant aerodynamic disturbances introduced by inductive flows of vortex column of main rotor. This determines the need to create the means of measurement, taking into account the aerodynamics and dynamics of the helicopter flight. The known direction of overcoming these limitations is the use for measuring the information of aerodynamic field of vortex column of main rotor and its perception by means of the stationary multi-functional aerometric receiver. However, the need to protect a large number of full-pressure tubes installed in the flow channel of the multifunctional aerometric receiver, strict requirements for the identity and stability of the characteristics of the large number aerometric channels, complicate the design, reduce reliability, increase cost, inhibit the use of the air parameters measurement system on helicopters of various classes and purposes. Principles of construction, functional scheme, features of perception of primary information of measuring system of air parameters of the helicopter with the stationary receiver of a stream, ion-label and aerometric measuring channels are showed. Algorithms for processing primary information at various stages and flight modes, including: in the parking lot before the launch of the power plant and when rotating the rotor, when taxiing and maneuvering on the earth's surface, on takeoff and landing modes and when flying at low speeds, at flight speeds, when the stationary receiver of primary information leaves the zone of the vortex column of the rotor using ion-label and aerometric measuring channels, are presented. It is shown that the proposed approaches to the construction, models and algorithms for processing the primary information of the measuring system air parameters of helicopter with ion-label and aerometric measurement channels allow to determine the speed and direction of the wind vector, altitude-velocity parameters of motion relative to the environment and atmospheric parameters in a wide range of helicopter operation, which determines its competitive advantages in solving problems of piloting and provide the flight safety of helicopters of different classes and purposes.

Keywords: helicopter, air parameters, measurement, system, stationary receiver, ion-label and aerometric measurement channels, functional diagram, design features, algorithms of information processing

Acknowledgements: Work performed under grant of the Russian Foundation for Basic Research № 18-08-00264.

For citation:

Soldatkin V. M., Soldatkin V. V., Nikitin A. V., Ariskin E. O. Measuring System of Air Parameters of the Helicopter with a Stationary Receiver a Flow, Ion-Label And Aerometric Measurement Channels, *Mekhatronica, Avtomatizatsiya, Upravlenie*, 2018, vol. 19, no. 11, pp. 744–752.

DOI: 10.17587/mau.19.744-752

References

1. **Rukovodstvo** *po letnoj jekspluatacii vertoleta Mi-8 (izdanie 4),* Moscow, Avtoritet, 1996, 554 p. (in Russian).

2. Kozicin V. K., Makarov N. N., Porunov A. A., Soldatkin V. M. Analiz principov postroenija sistem izmerenija vozdushnyh signalov vertoleta (Analysis of the principles of construction of measuringair data systems of the helicopter), Aviakosmicheskoe priborostroenie, 2003, vol. 10, pp. 2–13 (in Russian).

3. Soldatkin V. M. Metody i sredstva izmerenija ajerodinamicheskih uglov letatel'nyh apparatov (Methods and means of measurement of aerodynamic angles of the aircraft), Kazan, Publishing house of Kazan. gos. tehn. un-t, 2001, 448 p. (in Russian).

4. Soldatkin V. V. Ajerometricheskaja sistema izmerenija malyh vozdushnyh skorostej vertoleta na osnove informacii o polozhenii vihrevoj kolonny nesushhego vinta (Aerometric system of measurement of low air speeds of the helicopter based on information about the position of the vortex column of the main rotor), Izvestija vuzov. Aviacionnaja tehnika, 2009, no. 4, pp. 52–56 (in Russian).

5. Soldatkin V. V. Sistema vozdushnyh signalov vertoleta na osnove nepodvizhnogo ajerometricheskogo priemnika i informacii ajerodinamicheskogo polja vihrevoj kolonny nesushhego vinta (Air data system of helicopter on the based stationary aerometric receiver and information aerodynamic field of vortex column of the main rotor), Kazan Publishing house of Kazan. gos. tehn. un-ta, 2012, 284 p. (in Russian).

6. Nikitin A. V., Soldatkin V. V., Soldatkin V. M. Sistema izmerenija parametrov vektora vetra na startovyh i vzletno-posadochnyh rezhimah vertoleta (The measuring system of wind vector parameters at the starting and takeoff and landing modes of the helicopter), Mekhatronika, avtomatizatsiya, upravlenie, 2013, no. 6, pp. 64–70 (in Russian).

7. Nikitin A. V., Soldatkin V. M. Sistema izmerenija parametrov vektora vetra i istinnoj vozdushnoj skorosti na bortu vertoleta (The measuring system of wind vector parameters and true airspeed the helicopter), Datchiki i Sistemy, 2015, no. 4, pp. 48–54 (in Russian).

8. Nikitin A. V., Ariskin E. O., Soldatkin V. V., Soldatkin V. M. Bortovaja sistema izmerenija parametrov vektora vetra na stojanke, startovyh i vzletno-posadochnyh rezhimah vertoleta s ajerometricheskimi i ionno-metochnymi izmeritel'nymi kanalami (Onboard measuring system of wind vector parameters in the parking, starting and takeoff and landing modes of the helicopter with aerometric and ion-mark measuring channels), *Izvestija vuzov. Aviacionnaja tehnika*, 2015, no. 4, pp. 87–93 (in Russian).

9. Ganeev F. A., Soldatkin V. M. Ionno-metochnyj datchik ajerodinamicheskogo ugla i vozdushnoj skorosti s logometricheskimi informativnymi signalami i interpoljacionnoj shemoj obrabotki (Ionmark sensor of aerodynamic angle and air velocity with logometric informative signals and interpolation processing scheme), Izvestija vuzov. Aviacionnaja tehnika, 2010, no. 3, pp. 46–50 (in Russian).

vuzov. Aviacionnaja tehnika, 2010, no. 3, pp. 46–50 (in Russian). 10. Erusalimskij M. A., Egorov V. N. Jekipazham vertoleta informacionnuju podderzhku (The crew of the helicopter information support), Aviasojuz, no. 2, pp. 24–26 (in Russian).

11. **Petunin A. N.** *Metody i tehnika izmerenija parametrov gazovogo potoka (priemniki davlenija i skorostnogo napora)* (Methods and technique of measuring parameters of a gas (pressure receivers and dynamic head)), Moscow, Mashinostroenie, 1972, 332 p. (in Russian).

Издательство «НОВЫЕ ТЕХНОЛОГИИ»

107076, Москва, Стромынский пер., 4

Телефон редакции журнала: (499) 269-5510, (499) 269-5397

Технический редактор Е. В. Конова. Корректор Е. В. Комиссарова.

Сдано в набор 23.08.2018. Подписано в печать 15.10.2018. Формат 60×88 1/8. Бумага офсетная. Усл. печ. л. 8,86. Заказ МН1118. Цена договорная.

Журнал зарегистрирован в Комитете Российской Федерации по делам печати,

телерадиовещания и средств массовых коммуникаций

Свидетельство о регистрации ПИ № 77-11648 от 21.01.02

Учредитель: Издательство "Новые технологии"

Оригинал-макет ООО "Адвансед солюшнз". Отпечатано в ООО "Адвансед солюшнз".

119071, г. Москва, Ленинский пр-т, д. 19, стр. 1.