

А. Нанданвар, Л.А. Рыбак, Д.А. Дьяконов

УПРАВЛЕНИЕ МУЛЬТИРОБОТИЗИРОВАННЫМИ СИСТЕМАМИ НА ОСНОВЕ СКОЛЬЗЯЩИХ РЕЖИМОВ ВЫСОКОГО ПОРЯДКА

Рассматривается задача управления мультиагентной роботизированной системой второго порядка с дискретным временем в условиях сетевых задержек. Предложен новый подход к управлению формированием агентов, основанный на скользящем режиме высшего порядка и облачных технологиях. Для описания взаимодействия между агентами используется теория графов, где матрица Лапласа \mathcal{L} представляет канал связи между агентами и лидером. Динамика системы описывается уравнениями движения для положения и скорости каждого агента. Особое внимание уделяется влиянию сетевых задержек, возникающих при передаче данных от датчиков к контроллеру и от контроллера к исполнительным механизмам. Разработан многоступенчатый предиктор состояния, использующий методы прогнозирования для компенсации случайных задержек в сети. Предложенный алгоритм управления обеспечивает быструю сходимость системы к желаемому образованию даже при наличии существенных сетевых задержек. Для каждого агента определяется поверхность скольжения и закон достижения, учитывающий несколько временных меток. Проведен детальный анализ устойчивости замкнутой системы, подтверждающий асимптотическую устойчивость разработанного алгоритма управления. Результаты моделирования в MATLAB демонстрируют высокую эффективность предложенного подхода: система из пяти последователей и одного лидера достигает желаемого формирования за 10.3 секунды и успешно поддерживает его при наличии случайных сетевых задержек. По сравнению с традиционными методами управления первого порядка, новый подход показывает значительно улучшенные характеристики, особенно в части снижения эффекта дребезжания в сигналах управления. Использование облачных технологий позволяет эффективно обрабатывать большие объемы данных в реальном времени и реализовывать сложные алгоритмы прогнозирования без перегрузки локальных вычислительных ресурсов агентов. Полученные результаты подтверждают перспективность применения предложенного подхода для управления мультиагентными системами в условиях реальных сетевых ограничений. Работа также демонстрирует возможность использования методов прогнозирования для компенсации случайных потерь пакетов и задержек связи, что обеспечивает надежное управление и связь в динамичных, непредсказуемых ситуациях.

Мультиагентная роботизированная система; облачный контроллер; скользящий режим; алгоритм.

A. Nandanwar, L.A. Rybak, D.A. Dyakonov

CONTROL OF A MULTI-ROBOT SYSTEM BASED ON HIGHER-ORDER SLIDING MODES

The article addresses the control problem of a second-order multi-agent robotic system with discrete time under network-induced delays. A novel approach to formation control is proposed, based on higher-order sliding mode control and cloud technologies. The interaction between agents is described using graph theory, where the Laplacian matrix \mathcal{L} represents the communication channel between agents and the leader. The system dynamics are modeled by motion equations for the position and velocity of each agent. Special attention is paid to the impact of network-induced delays that occur during data transmission from sensors to the controller and from the controller to actuators. A multi-stage state predictor is developed, utilizing prediction methods to compensate for random delays in the network. The proposed control algorithm ensures rapid convergence of the system to the desired formation even in the presence of significant network delays. For each agent, a sliding surface and a reaching law are defined, taking into account multiple timestamps. A detailed stability analysis of the closed-loop system confirms the asymptotic stability of the developed control algorithm. Simulation results in MATLAB demonstrate the high efficiency of the proposed approach: a system consisting of five followers and one leader achieves the desired formation in 10.3 seconds and successfully maintains it despite random network delays. Compared to traditional first-order control methods, the new approach shows significantly improved performance, particularly in reducing chattering effects in control signals. The use of cloud technologies enables efficient real-time processing of large data volumes and implementation of complex prediction algorithms without overloading the local computational resources of the agents. The obtained results con-

firm the potential of the proposed approach for controlling multi-agent systems under real-world network constraints. The work also demonstrates the feasibility of using prediction methods to compensate for random packet losses and communication delays, ensuring reliable control and communication in dynamic, unpredictable scenarios.

Multi-agent robotic system; cloud controller; sliding mode; algorithm.

Введение. Современные мультиагентные системы (MAS) играют ключевую роль в различных областях, таких как робототехника, транспортные системы, автоматизация и управление сложными процессами. Одной из основных задач таких систем является координация движения агентов для достижения общих целей, таких как следование за лидером, поддержание формаций или распределение ресурсов. Подобная задача рассматривается в статьях [1–7]. Однако реализация эффективного управления сталкивается с проблемой ограничения коммуникационных сетей. Решением проблемы является формирование эффективной математической модели. В статьях [8–18] предлагаются контроллер или математическая модель для управления мультиагентной системой. В статье рассматривается мультиагентная система второго порядка с дискретным временем, состоящая из N последователей и одного виртуального лидера. Основной целью является разработка алгоритма управления, способного обеспечить быстрое отслеживание положения лидера даже при наличии сетевых ограничений, таких как случайные задержки и потери данных. Для решения задачи используется метод скользящего режима высшего порядка (DHOSM), который позволяет минимизировать влияние внешних возмущений и внутренних ограничений на производительность системы. Для анализа и управления динамикой системы применяется матрица Лапласа, которая описывает структуру коммуникационной сети между агентами и лидером. Применение матрицы Лапласа показано в статьях [19–22]. Дополнительно предлагается интеллектуальная архитектура управления на основе облачных технологий, которая снижает вычислительную нагрузку на локальные устройства и обеспечивает прогнозирование случайных задержек в сети. Это позволяет достичь более устойчивого и точного управления системой даже в условиях непредсказуемых внешних воздействий. Разработанная схема управления включает многоступенчатый предиктор состояния, который использует данные о текущем положении и скорости агентов для прогнозирования их будущего поведения. Проверенное моделирование демонстрирует эффективность предложенного подхода: система достигает желаемой конфигурации за конечное время, минимизируя эффекты дрейфа и обеспечивая устойчивость к сетевым ограничениям.

Постановка задачи исследования. Матрица Лапласа \mathcal{L} представляет собой канал связи между каждым агентом и лидером, а архитектура коммуникационной сети мультиагентных систем (MAS) имеет форму прямого соединения. Матрица Лапласа может быть записана как $\mathcal{L} = \Delta - \mathcal{A}$, где $\Delta = \text{diag} \{\Lambda_1, \Lambda_2, \dots, \Lambda_N\}$ – диагональная матрица степеней, а $\Lambda_i = \sum_{j=1}^N a_{ij}$. Здесь $\mathcal{A} = [a_{ij}]$ является взвешенной матрицей смежности, а её элемент a_{ij} указывает на возможность обмена информацией между агентами: если $a_{ij} = 0$, то агент i не может получать информацию от агента j ; в противном случае $a_{ij} \neq 0$. Аналогично, $b_i = 1$ показывает, что агент способен получать информацию о состоянии лидера. Рассмотрим мультиагентную систему второго порядка с дискретным временем, состоящую из N последователей и виртуального лидера. Динамика лидера и последователей описывается следующими уравнениями:

Для последователей:

$$\begin{aligned} p_i(k+1) &= p_i(k) + \sigma v_i(k+1) \\ v_i(k+1) &= v_i(k) + \sigma u_i(k), \end{aligned} \quad (1)$$

где $p_i(k) \in \mathbf{R}$, $v_i(k) \in \mathbf{R}$, $u_i(k) \in \mathbf{R}$ представляют соответственно положение, скорость и управляющий вход агента i . Здесь σ – время выборки.

Для лидера:

$$\begin{aligned} p_0(k+1) &= p_0(k) + \sigma v_0(k+1) \\ v_0(k+1) &= v_0(k) + \sigma u_0(k), \end{aligned} \quad (2)$$

где $p_0(k) \in \mathbf{R}$, $v_0(k) \in \mathbf{R}$ – это положение и скорость лидера.

Сетевые задержки, такие как передача данных от датчика к контроллеру или от контроллера к исполнительному механизму, иногда неизбежны в системах сетевого управления. Эти задержки оказывают влияние на производительность и устойчивость системы. Данные, полученные контроллером, могут относиться к предыдущим шагам выборки или быть утерянными в канале передачи данных от контроллера к исполнительному механизму. Важно найти способ минимизировать влияние этих ограничений без ущерба для производительности системы. Поэтому целью данной работы является разработка алгоритма управления, который обеспечивает быстрое отслеживание позиции лидера в условиях таких ограничений.

Разработка системы управления на основе скользящего режима. Для разработки системы управления мультиагентной роботизированной системы рассматривается подход с использованием линейризованных моделей агентов и расширенного уравнения состояния, учитывающего ошибки положения и скорости. Для компенсации случайных сетевых задержек в облачной среде предлагается контроллер скользящего режима высшего порядка, который прогнозирует состояние агентов с помощью многоступенчатых предикторов. Устойчивость системы обеспечивается за счет корректной настройки параметров управления и применения функции Ляпунова, что гарантирует асимптотическую сходимость замкнутой системы. Прогнозирующий контроллер, размещенный в облаке, позволяет минимизировать влияние задержек и обеспечивать надежное взаимодействие между агентами в реальном времени. На рис. 1 показана схема облачного прогнозирующего контроллера.

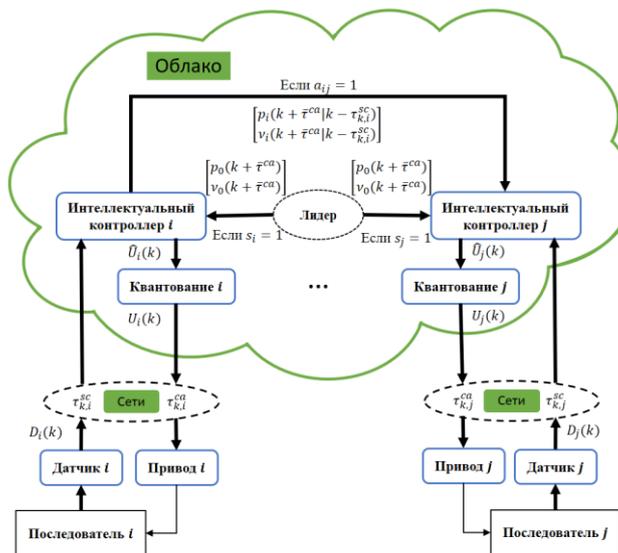


Рис. 1. Схема облачного прогнозирующего управления с временной вариацией

Разработка контроллера без задержек. На основе линейризованной модели отдельного агента формируется расширенное уравнение состояния полной системы MAS.

$$\begin{aligned} p(k + 1) &= p(k) + v(k + 1) \\ v(k + 1) &= v(k) + u(k), \end{aligned} \tag{3}$$

где $v(k) = [v_1^T(k), v_2^T(k), \dots, v_N^T(k)]$, $p(k) = [p_1^T(k), p_2^T(k), \dots, p_N^T(k)]$ и $u(k) = [u_1^T(k), u_2^T(k), \dots, u_N^T(k)]$. Время выборки $\sigma = 1$.

Погрешности положения и скорости могут быть представлены в виде:

$$e_{pi}(k) \& = p_i(k) - p_0(k) - \delta_{pi}(k) \tag{4}$$

$$e_{vi}(k) \& = v_i(k) - v_0(k) - \delta_{vi}(k), \tag{5}$$

где $\delta_{p_i}(k) \in \mathbf{R}$, $\delta_{v_i}(k) \in \mathbf{R}$ отражают желаемые значения относительного положения и скорости между агентом i и лидером. Для описания ошибок всей системы используется компактная форма:

$$e_p(k) = p(k) - P_0(k) - \delta_p(k) \quad (6)$$

$$e_v(k) = v(k) - V_0(k) - \delta_v(k), \quad (7)$$

где $\delta_p(k) = [\delta_{p,1}^T(k), \delta_{p,2}^T(k), \dots, \delta_{p,N}^T(k)]^T$ и $\delta_v(k) = [\delta_{v,1}^T(k), \delta_{v,2}^T(k), \dots, \delta_{v,N}^T(k)]^T$.

Поверхность скольжения для агента i задается формулой

$$s_i(k) = \eta_i \left(\sum_{j \in N} a_{ij} \left((v_j(k) - v_i(k)) + (p_j(k) - p_i(k)) \right) + b_i (e_{v_i}(k) + e_{p_i}(k)) \right). \quad (8)$$

Компактная форма формулы (8) представлена ниже

$$\begin{aligned} s(k) &= \eta((\mathcal{L} + \mathcal{B}) \otimes I_n)(e_v(k) + e_p(k)) \\ &= \Psi(e_v(k) + e_p(k)), \end{aligned} \quad (9)$$

где $e_v(k) = [e_{v1}^T(k), e_{v2}^T(k), \dots, e_{vN}^T(k)]^T$ и $e_p(k) = [e_{p1}^T(k), e_{p2}^T(k), \dots, e_{pN}^T(k)]^T$,

$s(k) = [s_1^T(k), s_2^T(k), \dots, s_N^T(k)]^T$, $\eta = \text{diag}(\eta_1, \eta_2, \dots, \eta_N)$, $\Psi = \eta((\mathcal{L} + \mathcal{B}) \otimes I_n)$.

Формула закона достижения агента i в скользящем режиме управления высшего порядка выглядит следующим образом:

$$\begin{aligned} s_i(k+1) &= -k_1 |s_i(k)|^{\frac{1}{2}} \text{sign}(s_i(k)) + \xi_i(k) \\ \xi_i(k+1) &= -k_3 \xi_i(k) - \sigma k_2 \text{sign}(s_i(k)), \end{aligned} \quad (10)$$

где $\xi_i(k)$ – вспомогательная переменная контроллера DHOSM, k_1 и k_2 – коэффициенты усиления настраиваемого контроллера, k_3 – фиксированный коэффициент усиления при $0 < k_3 < 1$. Упрощенная форма (10) представлена формулой

$$\begin{aligned} s_i(k+1) &= -k_1 |s_i(k)|^{\frac{1}{2}} \text{sign}(s_i(k)) - k_3 \xi_i(k-1) \\ &\quad - \sigma k_2 \text{sign}(s_i(k-1)). \end{aligned} \quad (11)$$

Для обеспечения глобального консенсуса в системе применяется следующий закон:

$$\begin{aligned} s(k+1) &= -k_1 |s(k)|^{\frac{1}{2}} \text{sign}(s(k)) - k_3 \xi(k-1) \\ &\quad - \sigma k_2 \text{sign}(s(k-1)). \end{aligned} \quad (12)$$

Более того, во время $(k+1)$, поверхность скольжения, представленная в уравнение (9), принимает следующий вид:

$$s(k+1) = \Psi(e_v(k+1) + e_p(k+1)). \quad (13)$$

Используя компактные выражения для ошибок положения и скорости, заданные в (6) и (7), а также применяя (12), получаем:

$$\begin{aligned} &-k_1 |s(k)|^{\frac{1}{2}} \text{sign}(s(k)) + k_3 \xi(k-1) - \sigma k_2 \text{sign}(s(k-1)) \\ &= \Psi(e_v(k+1) + e_p(k+1)) \\ &= \Psi(V_0(k+1) - v(k+1) - \delta_v(k+1) \\ &\quad + P_0(k+1) - p(k+1) - \delta_p(k+1)) \\ &= \Psi(V_0(k+1) + P_0(k+1) - p(k) \\ &\quad - \delta_p(k+1) - \delta_v(k+1) - 2v(k) - 2u(k)), \end{aligned} \quad (14)$$

где $V_0(k) = [v_0^T(k), v_0^T(k), \dots, v_0^T(k)]^T$, $P_0(k) = [p_0^T(k), p_0^T(k), \dots, p_0^T(k)]^T$.

Результатом дальнейших вычислений является закон управления скользящим режимом более высокого порядка, который может быть выражен следующим образом:

$$u(k) = \frac{k_3}{2} \xi(k-1) - \frac{k_1}{2} |s(k)|^{\frac{1}{2}} \text{sign}(s(k)) - \sigma \frac{k_2}{2} \text{sign}(s(k-1)) + V_0(k+1) + \frac{1}{2} e_p(k) - \delta_v(k+1) - v(k). \quad (15)$$

Выполним анализ устойчивости. Для предложенного закона управления (15) и скользящей переменной (9) траектории замкнутых систем (1) и (2) демонстрируют асимптотическую устойчивость при выполнении условия:

$$\gamma_s < s^T(k) s(k). \quad (16)$$

Доказательство:

Рассмотрим функцию Ляпунова в виде:

$$V_s(k) = s(k)^T s(k). \quad (17)$$

Очевидно, что $V_s(k) > 0$. Изменение функции Ляпунова на шаге k определяется как:

$$\begin{aligned} \Delta V_s(k) &= V_s(k+1) - V_s(k) \\ &= s(k+1)^T s(k+1) - s(k)^T s(k) \\ &= [\Psi(e_v(k+1) + e_p(k+1))]^T \\ &\quad [\Psi(e_v(k+1) + e_p(k+1))] - s(k)^T s(k). \end{aligned} \quad (18)$$

Подставляя выражения для скользящей переменной (9), уравнений ошибок (4), (5) и закона управления (15), упрощаем (18) до:

$$\Delta V_s(k) = \gamma_s - s(k)^T s(k), \quad (19)$$

где $\gamma_s = -k_1 |s(k)|^{\frac{1}{2}} \text{sign}(s(k)) + k_3 \xi_i(k-1) - \sigma k_2 \text{sign}(s(k-1))$.

При корректной настройке параметров k_1 и k_2 величина γ_s может быть минимизирована, обеспечивая выполнение неравенства $s(k)^T s(k) > \gamma_s$. Таким образом, для любого малого значения $\Gamma' > 0$ выполняется:

$$\Delta V_s(k) < \Gamma' s(k)^T s(k). \quad (20)$$

Это подтверждает конечную сходимости системы за время $\Delta V_s(k)$

Разработка интеллектуального контроллера на основе облака. Для реализации совместного управления в дискретных мультиагентных системах предлагается архитектура интеллектуального управления, основанная на облачных технологиях. Этот подход предоставляет ряд уникальных преимуществ. Благодаря использованию распределенных вычислений облачные системы обеспечивают обработку и хранение данных в реальном времени, что позволяет эффективно управлять задержками и снижать вычислительную нагрузку на систему. Кроме того, такие системы поддерживают применение современных алгоритмов прогнозирования для минимизации случайных потерь данных и задержек, обеспечивая надежное управление и связь даже в динамичных и непредсказуемых условиях.

В облачной сети возникают случайные сетевые задержки, например, при передаче данных от датчиков к контроллеру и от контроллера к исполнительным устройствам каждого агента. Эти задержки обозначаются как $\tau_{k,i}^{sc}$ и $\tau_{k,i}^{ca}$ соответственно. Целочисленные значения $\bar{\tau}_{k,i}^{sc}$ и $\bar{\tau}_{k,i}^{ca}$ представляют собой верхние границы для $\tau_{k,i}^{sc}$ и $\tau_{k,i}^{ca}$. Для компенсации влияния случайных сетевых задержек разработан контроллер скользящего режима высшего порядка, который обеспечивает достижение формирования, изменяющегося во времени. При проектировании системы каждый агент оснащается компенсатором в приводе и прогнозирующим контроллером, размещенным в облачном узле. Датчики фиксируют скорость и местоположение агентов в момент времени k , после чего эти данные передаются в сеть для дальнейшей обработки контроллером. Сеть использует метод прогнозирования для оценки общей случайной задержки в каждый момент времени на основе временной метки k . Этот расчет учитывает влияние таких факторов, как потеря пакетов и случайные задержки в канале связи, обеспечивая более точное управление.

Разработка многоступенчатого предиктора состояния. Последний доступный пакет данных для прогнозирующего контроллера каждого агента в момент времени k определяется как: $\zeta(k - \tau_{k,i}^{sc}) = \{p_i(k - \tau_{k,i}^{sc}), v_i(k - \tau_{k,i}^{sc})\}$, где $k - \tau_{k,i}^{sc}$ – временная мет-

ка пакета, связанная с задержкой в сети $\tau_{k,i}^{sc}$ от датчика к контроллеру. На основе этих данных выполняется прогнозирование положения, скорости и поверхности скольжения для интервала от $k - \tau_{k,i}^{sc} + 1$ до $k + \bar{\tau}_{k,i}^{ca}$.

Скорость агентов оценивается с использованием формул (21)-(23):

$$\hat{p}(k - \tau_k^{sc} + 1 | k - \tau_k^{sc}) = \hat{p}(k - \tau_k^{sc} | k - \tau_k^{sc}) + \hat{v}(k - \tau_k^{sc} + 1 | k - \tau_k^{sc}) \quad (21)$$

$$\hat{p}(k - \tau_k^{sc} + 2 | k - \tau_k^{sc}) = \hat{p}(k - \tau_k^{sc} + 1 | k - \tau_k^{sc}) + \hat{v}(k - \tau_k^{sc} + 2 | k - \tau_k^{sc}) \quad (22)$$

⋮

$$\hat{p}(k + \bar{\tau}_k^{ca} | k - \tau_k^{sc}) = \hat{p}(k + \bar{\tau}_k^{ca} - 1 | k - \tau_k^{sc}) + \hat{v}(k + \bar{\tau}_k^{ca} | k - \tau_k^{sc}). \quad (23)$$

Аналогично, положение агентов вычисляется по формулам (24)-(26):

$$\hat{v}(k - \tau_k^{sc} + 1 | k - \tau_k^{sc}) = \hat{v}(k - \tau_k^{sc} | k - \tau_k^{sc}) + \hat{u}_i(k - \tau_k^{sc} | k - \tau_k^{sc}) \quad (24)$$

$$\hat{v}(k - \tau_k^{sc} + 2 | k - \tau_k^{sc}) = \hat{v}(k - \tau_k^{sc} + 1 | k - \tau_k^{sc}) + \hat{u}_i(k - \tau_k^{sc} + 1 | k - \tau_k^{sc}) \quad (25)$$

⋮

$$\begin{aligned} \hat{v}(k + \bar{\tau}_k^{ca} | k - \tau_k^{sc}) &= \hat{v}(k + \bar{\tau}_k^{ca} - 1 | k - \tau_k^{sc}) \\ &+ \hat{u}(k + \bar{\tau}_k^{ca} - 1 | k - \tau_k^{sc}) \end{aligned} \quad (26)$$

Прогнозируемая поверхность скольжения для момента времени k определяется формулами (27)-(30).

$$\begin{aligned} \hat{s}(k - \tau_k^{sc} | k - \tau_k^{sc}) &= \Psi(\hat{v}(k - \tau_k^{sc} | k - \tau_k^{sc}) - V_0(k - \tau_k^{sc} | k - \tau_k^{sc}) - \delta_v(k - \tau_k^{sc} | k - \\ &\tau_k^{sc}) + \hat{p}(k - \tau_k^{sc} | k - \tau_k^{sc}) - P_0(k - \tau_k^{sc} | k - \tau_k^{sc}) - \delta_p(k - \tau_k^{sc} | k - \tau_k^{sc})) \end{aligned} \quad (27)$$

$$\begin{aligned} \hat{s}(k - \tau_k^{sc} + 1 | k - \tau_k^{sc}) &= \Psi(2\hat{v}(k - \tau_k^{sc} + 1 | k - \tau_k^{sc}) - V_0(k - \tau_k^{sc} + 1 | k - \tau_k^{sc}) - \\ &\delta_v(k - \tau_k^{sc} + 1 | k - \tau_k^{sc}) + \hat{p}(k - \tau_k^{sc} + 1 | k - \tau_k^{sc}) - P_0(k - \tau_k^{sc} + 1 | k - \tau_k^{sc}) - \delta_p(k - \tau_k^{sc} + \\ &1 | k - \tau_k^{sc})) \end{aligned} \quad (28)$$

$$\begin{aligned} \hat{s}(k - \tau_k^{sc} + 2 | k - \tau_k^{sc}) &= \Psi(3\hat{v}(k - \tau_k^{sc} + 2 | k - \tau_k^{sc}) + 3\hat{u}(k - \tau_k^{sc} + 2 | k - \tau_k^{sc}) + 2\hat{u}(k - \tau_k^{sc} + \\ &2 | k - \tau_k^{sc}) - V_0(k - \tau_k^{sc} + 2 | k - \tau_k^{sc}) - \delta_v(k - \tau_k^{sc} + 2 | k - \tau_k^{sc}) + \hat{p}(k - \tau_k^{sc} + 2 | k - \\ &\tau_k^{sc}) - P_0(k - \tau_k^{sc} + 2 | k - \tau_k^{sc}) - \delta_p(k - \tau_k^{sc} + 2 | k - \tau_k^{sc})) \end{aligned} \quad (29)$$

⋮

$$\begin{aligned} \hat{s}(k + \bar{\tau}_k^{ca} | k - \tau_k^{sc}) &= \Psi(2\hat{v}(k + \bar{\tau}_k^{ca} - 1 | k - \tau_k^{sc}) + 2\hat{u}(k + \bar{\tau}_k^{ca} - 1 | k - \tau_k^{sc}) - \\ &V_0(k + \bar{\tau}_k^{ca} | k - \tau_k^{sc}) - \delta_v(k + \bar{\tau}_k^{ca} | k - \tau_k^{sc}) + \hat{p}(k + \bar{\tau}_k^{ca} - 1 | k - \tau_k^{sc}) - P_0(k + \bar{\tau}_k^{ca} | k - \\ &\tau_k^{sc}) - \delta_p(k + \bar{\tau}_k^{ca} | k - \tau_k^{sc})). \end{aligned} \quad (30)$$

Компактная форма этой поверхности представлена уравнением (31):

$$\begin{aligned} \hat{S}(k + 1 | k - \tau_k^{sc}) &= \Psi(-V_{0\kappa}(k + 1 | k - \tau_k^{sc}) - P_{0\kappa}(k + 1 | k - \tau_k^{sc}) \\ &- \delta_{v\kappa}(k | k - \tau_k^{sc}) - \delta_{p\kappa}(k | k - \tau_k^{sc}) \\ &+ \Pi \hat{V}_\kappa(k | k - \tau_k^{sc}) + \hat{P}_\kappa(k | k - \tau_k^{sc}) + \Upsilon \hat{u}_\kappa(k | k - \tau_k^{sc})) \end{aligned} \quad (31)$$

где

$$\begin{aligned} \hat{S}(k + 1) &= [\hat{s}(k - \tau_k^{sc} + 1)^T, \hat{s}(k - \tau_k^{sc} + 2)^T, \dots, \hat{s}(k + \bar{\tau}_k^{ca})^T]^T, \\ V_{0\kappa}(k + 1) &= [V_0(k - \tau_k^{sc} + 1)^T, V_0(k - \tau_k^{sc} + 2)^T, \dots, V_0(k + \bar{\tau}_k^{ca})^T]^T, \\ P_{0\kappa}(k + 1) &= [P_0(k - \tau_k^{sc} + 1)^T, P_0(k - \tau_k^{sc} + 2)^T, \dots, P_0(k + \bar{\tau}_k^{ca})^T]^T, \\ \hat{\delta}_{p\kappa}(k + 1) &= [\hat{\delta}_p(k - \tau_k^{sc} + 1)^T, \hat{\delta}_p(k - \tau_k^{sc} + 2)^T, \dots, \hat{\delta}_p(k + \bar{\tau}_k^{ca})^T]^T, \\ \hat{\delta}_{v\kappa}(k + 1) &= [\hat{\delta}_v(k - \tau_k^{sc} + 1)^T, \hat{\delta}_v(k - \tau_k^{sc} + 2)^T, \dots, \hat{\delta}_v(k + \bar{\tau}_k^{ca})^T]^T, \\ \Pi &= \text{diag}\{2, 3, \dots, \bar{\tau}_k^{ca}\}, \\ \hat{V}_\kappa(k) &= [\hat{v}(k - \tau_k^{sc})^T, \hat{v}(k - \tau_k^{sc} + 1)^T, \dots, \hat{v}(k + \bar{\tau}_k^{ca} - 1)^T]^T, \\ \hat{u}_\kappa(k) &= [\hat{u}(k - \tau_k^{sc})^T, \hat{u}(k - \tau_k^{sc} + 1)^T, \dots, \hat{u}(k + \bar{\tau}_k^{ca} - 1)^T]^T, \\ \hat{P}_\kappa(k) &= [\hat{p}(k - \tau_k^{sc})^T, \hat{p}(k - \tau_k^{sc} + 1)^T, \dots, \hat{p}(k + \bar{\tau}_k^{ca} - 1)^T]^T, \end{aligned}$$

$$Y(k) = \begin{bmatrix} 2 & 0 & \dots & 0 \\ 3 & 2 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots \\ k + \bar{\tau}_k^{ca} & \bar{\tau}_k^{ca} & \dots & 2 \end{bmatrix}.$$

Используя метод прогнозирования и закон достижения, поверхность скольжения для k -го временного шага определяется как:

$$S(k+1) = -K_{1,k}\Omega_1(k) - K_{3,k}\bar{\xi}(k) - K_{2,k}\Omega_2(k), \quad (32)$$

где $K_{1,k} = \text{diag}\{k_1, k_1, \dots, k_1\}$, $K_{2,k} = \text{diag}\{k_2, k_2, \dots, k_2\}$, $K_{3,k} = \text{diag}\{k_3, k_3, \dots, k_3\}$,

$$\Omega_1(k) = \begin{bmatrix} |s(k)|^{\frac{1}{2}} \text{sing}(s(k)) \\ |s(k+1)|^{\frac{1}{2}} \text{sing}(s(k+1)) \\ \vdots \\ |s(k + \bar{\tau}_k^{sa})|^{\frac{1}{2}} \text{sing}(s(k + \bar{\tau}_k^{sa})) \end{bmatrix},$$

$$\Omega_2(k) = [\text{sing}(s(k-1)) \text{sing}(s(k)) \dots \text{sing}(s(k + \bar{\tau}_k^{sa} - 1))]^T, \\ \bar{\xi}(k) = [\xi(k-1) \xi(k) \dots \xi(k + \bar{\tau}_k^{sa})]^T.$$

После дополнительных вычислений закон управления мультиагентной системой, учитывающий случайные задержки в сети, принимает вид:

$$\hat{u}_k(k) = \frac{1}{\gamma\varphi} \left(-K_{1,k}\Omega_1(k|k - \tau_k^{sc}) - K_{3,k}\bar{\xi}(k|k - \tau_k^{sc}) - \sigma K_{2,k}\Omega_2(k|k - \tau_k^{sc}) \right) + \\ V_{0k}(k+1|k - \tau_k^{sc}) + P_{0k}(k+1|k - \tau_k^{sc}) + \delta_{vk}(k|k - \tau_k^{sc}) + \delta_{vk}(k|k - \tau_k^{sc}) - \\ \Pi \hat{V}_k(k|k - \tau_k^{sc}) - \hat{P}_k(k|k - \tau_k^{sc}) \quad (33)$$

Система является асимптотически устойчивой, что подтверждается анализом устойчивости замкнутой системы, аналогичным расчетам, выполненным в разделе II-B.

Результаты моделирования. Рассмотрим дискретную систему второго порядка, состоящую из одного лидера (обозначенного как 0) и пяти последователей (обозначенных как 1, 2, ..., 5). Предполагается, что траектория движения лидера известна всем последователям. Начальные положения последователей и лидера заданы следующим образом: $p_1 = [-3 \ -3]^T$, $p_2 = [0 \ -3]^T$, $p_3 = [0 \ -1]^T$, $p_4 = [-3 \ 0]^T$, $p_5 = [-5 \ 0]^T$, и $p_0 = [0 \ 0]^T$. Начальная скорость лидера определяется как $v_0 = [0.7 \ 0.7]^T$. Для каждого последователя заданы желаемые расстояния разделения относительно лидера: $\delta_{p1} = [0 \ -2]^T$, $\delta_{p2} = [-4 \ 4]^T$, $\delta_{p3} = [-2 \ 2]^T$, $\delta_{p4} = [2 \ -2]^T$ и $\delta_{p5} = [4 \ -4]^T$. Также определены требуемые скорости разделения, зависящие от времени t : $\delta_{v1} = [0 \ 0]^T$, $\delta_{v2} = [-4 \ \cos(2t) \ 4 \ \cos(2t)]^T$, $\delta_{v3} = [-2 \ \cos(2t) \ 2 \ \cos(2t)]^T$, $\delta_{v4} = [2 \ \cos(2t) \ -2 \ \cos(2t)]^T$ и $\delta_{v5} = [4 \ \cos(2t) \ -4 \ \cos(2t)]^T$. Параметры контроллера выбраны следующим образом: $k_1 = 20$ и $k_2 = 0.1$. Период выборки составляет $\sigma = 1$ с. В системе присутствуют случайные задержки связи, которые для каждого последователя находятся в следующих диапазонах: $\tau_1^{sc} \in [1 \ 3]^T$, $\tau_2^{sc} \in [1 \ 2]^T$, $\tau_3^{sc} \in [2 \ 3]^T$, $\tau_4^{sc} \in [2 \ 3]^T$, $\tau_5^{sc} \in [2 \ 4]$, $\tau_5^{sc} \in [3 \ 4]^T$. Аналогично, для асинхронных задержек τ^{sa} : $\tau_1^{sa} \in [1 \ 3]$, $\tau_2^{sa} \in [1 \ 2]^T$, $\tau_3^{sa} \in [2 \ 3]^T$, $\tau_4^{sa} \in [2 \ 4]^T$, $\tau_5^{sa} \in [3 \ 4]^T$.

Моделирование системы выполняется в среде MATLAB.

На рис. 2 изображена траектория движения каждого робота. Пять роботов-последователей синхронизируют свои движения с траекторией лидера, формируя изменяющуюся во времени структуру после переходного периода длительностью 10,3 секунды. После того как желаемая траектория достигнута, предложенный алгоритм управления переводит роботов-последователей на фиксированную траекторию, сохраняя при этом заданное формирование. Благодаря использованию метода прогнозирования облачности система успешно поддерживает это формирование на протяжении всего моделирования, несмотря на влияние случайных задержек.

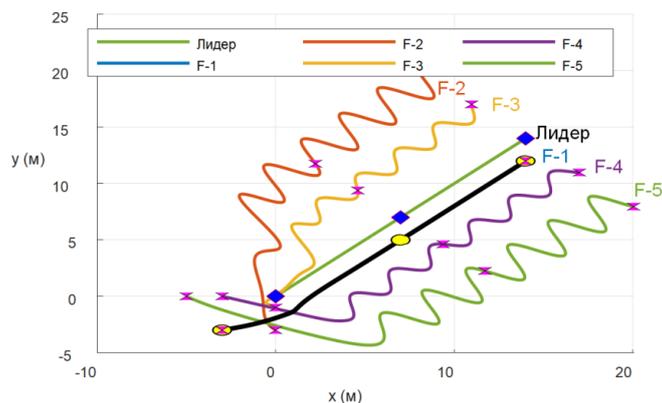


Рис. 2. Формирование времени с использованием предложенного подхода

Желаемые различия в расстояниях между агентами $\delta_{p,i}$ наглядно представлены на рис. 3. Это демонстрирует, что система "лидер-последователи" успешно достигла целевой конфигурации. Изначально между лидером и последователями наблюдались значительные расстояния, однако благодаря предлагаемым контроллерам агенты постепенно сближаются до необходимых интервалов и сохраняют их на протяжении оставшейся части пути, учитывая возникающие задержки.

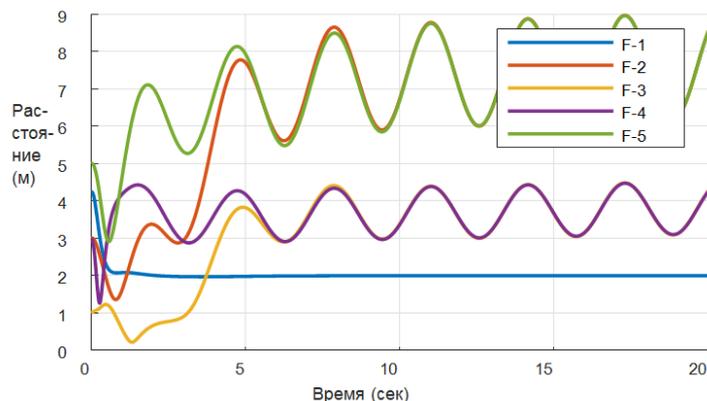


Рис. 3. Относительное расстояние последователей от лидера

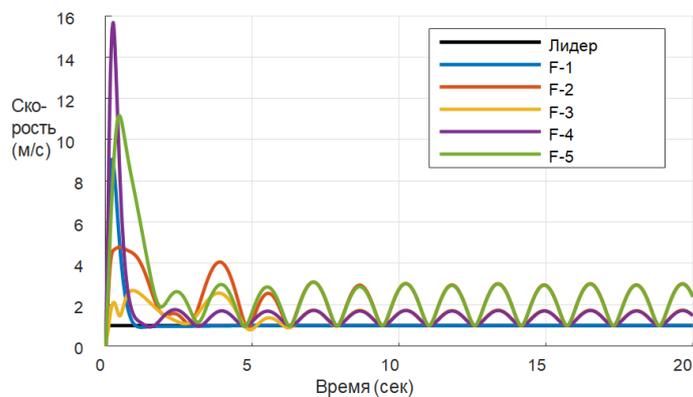


Рис. 4. Управляющий ввод последователей

На рис. 4 представлены управляющие входные данные лидера и последователей в виде скоростей. Несмотря на наличие задержек, выходные сигналы контроллера не демонстрируют признаков дребезжания. Это подтверждает, что управление скользящим режимом более высокого порядка эффективно снижает дребезжание по сравнению с управлением скользящим режимом первого порядка. Предложенный контроллер был протестирован при различных скоростях и маневрах, включая повороты. Поверхности скольжения для последователей показаны на рис. 5. Результаты демонстрируют, что характерным точкам агентов требуется около 10,3 секунды для достижения поверхностей скольжения. Достижение режима скольжения сигнализируется выполнением условия $s_i(k) = 0$ за конечное время.

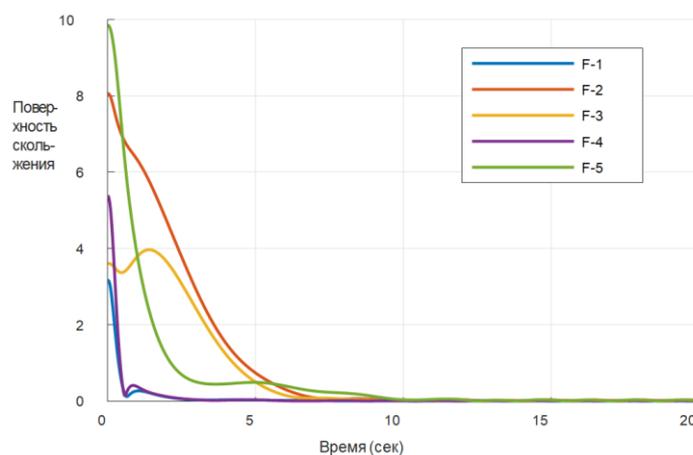


Рис. 5. Поверхность скольжения

Результаты моделирования показывают, что формирование изменяющейся во времени структуры достигается за конечное время благодаря использованию контроллера скользящего режима более высокого порядка. Предложенный подход обеспечивает более быстрое достижение целевой конфигурации, а также эффективно справляется с явлением дребезжания, улучшая качество управления.

Заключение. В статье рассмотрена задача управления мультиагентной роботизированной системой второго порядка с дискретным временем в условиях сетевых задержек. Результаты моделирования в MATLAB демонстрируют высокую эффективность предложенного подхода: система из пяти последователей и одного лидера достигает желаемого формирования за 10.3 секунды и успешно поддерживает его при наличии случайных сетевых задержек. По сравнению с традиционными методами управления первого порядка, новый подход показывает значительно улучшенные характеристики, особенно в части снижения эффекта дребезжания в сигналах управления. Использование облачных технологий позволяет эффективно обрабатывать большие объемы данных в реальном времени и реализовывать сложные алгоритмы прогнозирования без перегрузки локальных вычислительных ресурсов агентов. Полученные результаты подтверждают перспективность применения предложенного подхода для управления мультиагентными системами в условиях реальных сетевых ограничений. Работа также демонстрирует возможность использования методов прогнозирования для компенсации случайных потерь пакетов и задержек связи, что обеспечивает надежное управление и связь в динамичных, непредсказуемых ситуациях. Предложенный подход к управлению мультиагентными системами показал свою эффективность и перспективность для применения в условиях реальных сетевых ограничений. Полученные результаты создают основу для дальнейшего развития методов управления распределенными системами в условиях неопределенности и сетевых возмущений.

Исследование выполнено за счет гранта Российского научного фонда № 23-79-01326 <https://rscf.ru/project/23-79-01326/> с использованием оборудования Центра высоких технологий БГТУ им. В.Г. Шухова.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Назарова А.В., Рыжова Т.П. Методы и алгоритмы мультиагентного управления робототехнической системой // Инженерный журнал: наука и инновации. – 2012. – № 6 (6). – С. 93-105.
2. Naserian M., Ramazani A., Khaki A., Moarefiانpour A. Leader–follower consensus control for a nonlinear multi-agent robot system with input saturation and external disturbance // Systems Science & Control Engineering. – 2021. – Vol. 9 (1). – P. 260-271.
3. Xiong F., Zhang Y., Kuang X., He L., Han X. Multi-agent dual actor-critic framework for reinforcement learning navigation // Applied Intelligence. – 2024. – Vol. 55 (2). – P. 104-124.
4. Ahmed S., Karsiti M., Loh R. Multiagent Systems // InTech. – 2009. – P. 428.
5. Mousavi A., Davaie Markazi A. H. A new control method for leader-follower consensus problem of uncertain constrained nonlinear multi-agent systems // Journal of the Franklin Institute. – 2024. – Vol. 361 (9).
6. Behera L., Rybak L., Malyshev D.I., Khalapyan S. Numerical simulation of the workspace of robots with moving bases in the multi-agent system // Procedia Computer Science. – 2021. – Vol. 186 (6). – P. 431-439.
7. Nandanwar A., Dhar N.K., Malyshev D., Rybak L., Behera L. Finite-Time Robust Admissible Consensus Control of Multirobot System under Dynamic Events // IEEE Systems Journal. – 2021. – Vol. 15 (1). – P. 780-790.
8. Kim J. Three-dimensional multi-robot control to chase a target while not being observed // International Journal of Advanced Robotic Systems. – 2019. – Vol. 16 (1). – P. 1-11.
9. Azid S., Raghuwaiya K., Javed A., Kumari E. Autonomous Leader–Follower Formation of Vehicular Robots Using the Lyapunov Method // Unmanned Systems. – 2022. – Vol. 12 (01). – P. 75-85.
10. Yang H., Li S., Yang L., Ding Z. Leader-Following Consensus of Fractional-Order Uncertain Multi-Agent Systems with Time Delays // Neural Processing Letters. – 2022. – Vol. 54 (6). – P. 4829-4849.
11. Chen B., Qi X., Li C., Qi X., Ma H. Observer-Based Distributed Adaptive Consensus Tracking of Nonlinear Multi-agent Systems on Directed Graphs // IEEE Access. – 2022. – Vol. PP (99). – P. 1-1.
12. Ma J., Sun D., Haibo J., Feng G. Leader-following consensus of multi-agent systems with limited data rate // Journal of the Franklin Institute. – 2016. – Vol. 354 (1). – P. 184-196.
13. Kim J. Three dimensional motion camouflage guidance utilizing multiple leaders and one interceptor // IET Radar, Sonar & Navigation. – 2021. – Vol. 16 (3). – P. 617-631.
14. Ramachandran R., Fronda N., Preiss J., Dai Z., Sukhatme G. Resilient Multi-Robot Multi-Target Tracking // IEEE Transactions on Automation Science and Engineering. – 2024. – Vol. 21 (3). – P. 1-17.
15. Rehak B., Lynnyk A., Lynnyk V. Synchronization of Multi-Agent Systems Composed of Second-Order Underactuated Agents // Mathematics. – 2024. – Vol. 12 (21). – P. 3424.
16. Kou L., Huang Y., Zuo G., Jian L., Dou Y. Fixed-time rotating consensus control of second-order multi-agent systems // International Journal of Robust and Nonlinear Control. – 2024. – Vol. 34 (18). – P. 12031-12049.
17. Chen J., Yang Y., Qin S. A Distributed Optimization Algorithm for Fixed-Time Flocking of Second-Order Multiagent Systems // IEEE Transactions on Network Science and Engineering. – 2023. – Vol. 11 (1). – P. 152-162.
18. Yin Y, Shi Y, Liu F, [et al.]. Second-order consensus for heterogeneous multi-agent systems with input constraints // Neurocomputing. – 2019. – Vol. 351. – P. 43-50.
19. Nataliia Y., Zababurin K. Matrix Laplace transform // Boletín de la Sociedad Matemática Mexicana. – 2023. – Vol. 29 (3). – P. 1-21.
20. Olson N., Andrews J. A Matrix Exponential Generalization of the Laplace Transform of Poisson Shot Noise // IEEE Transactions on Information Theory. – 2024. – Vol. 71 (1). – P. 396-412.
21. Bouchenak A., Horani M. Al H., Younis J. [et al.]. Fractional Laplace transform for matrix valued functions with applications // Arab Journal of Basic and Applied Sciences. – 2022. – Vol. 29 (1). – P. 330-336.
22. Rani D., Mishra V. Laplace Transform Inversion using Bernstein Operational Matrix of Integration and its Application to Differential and Integral Equations // Proceedings Mathematical Sciences. – 2020. – Vol. 130 (1). – P. 60-89.

REFERENCES

1. Nazarova A.V., Ryzhova T.P. Metody i algoritmy multiagentnogo upravleniya robototekhnicheskoy sistemoi [Methods and algorithms of multi-agent control of a robotic system], *Inzhenernyy zhurnal: nauka i innovatsii* [Engineering Journal: Science and Innovation], 2012, No. 6 (6), pp. 93-105.
2. Naserian M., Ramazani A., Khaki A., Moarefianpour A. Leader-follower consensus control for a nonlinear multi-agent robot system with input saturation and external disturbance, *Systems Science & Control Engineering*, 2021, Vol. 9 (1), pp. 260-271.
3. Xiong F., Zhang Y., Kuang X., He L., Han X. Multi-agent dual actor-critic framework for reinforcement learning navigation, *Applied Intelligence*, 2024, Vol. 55 (2), pp. 104-124.
4. Ahmed S., Karsiti M., Loh R. Multiagent Systems, *InTech*, 2009, pp. 428.
5. Mousavi A., Davaie Markazi A. H. A new control method for leader-follower consensus problem of uncertain constrained nonlinear multi-agent systems, *Journal of the Franklin Institute*, 2024, Vol. 361 (9).
6. Behera L., Rybak L., Malyshev D.I., Khalapyan S. Numerical simulation of the workspace of robots with moving bases in the multi-agent system, *Procedia Computer Science*, 2021, Vol. 186 (6), pp. 431-439.
7. Nandanwar A., Dhar N.K., Malyshev D., Rybak L., Behera L. Finite-Time Robust Admissible Consensus Control of Multirobot System under Dynamic Events, *IEEE Systems Journal*, 2021, Vol. 15 (1), pp. 780-790.
8. Kim J. Three-dimensional multi-robot control to chase a target while not being observed, *International Journal of Advanced Robotic Systems*, 2019, Vol. 16 (1), pp. 1-11.
9. Azid S., Raghuvaiya K., Javed A., Kumari E. Autonomous Leader-Follower Formation of Vehicular Robots Using the Lyapunov Method, *Unmanned Systems*, 2022, Vol. 12 (01), pp. 75-85.
10. Yang H., Li S., Yang L., Ding Z. Leader-Following Consensus of Fractional-Order Uncertain Multi-Agent Systems with Time Delays, *Neural Processing Letters*, 2022, Vol. 54 (6), pp. 4829-4849.
11. Chen B., Qi X., Li C., Qi X., Ma H. Observer-Based Distributed Adaptive Consensus Tracking of Nonlinear Multi-agent Systems on Directed Graphs, *IEEE Access*, 2022, Vol. PP (99), pp. 1-1.
12. Ma J., Sun D., Haibo J., Feng G. Leader-following consensus of multi-agent systems with limited data rate, *Journal of the Franklin Institute*, 2016, Vol. 354 (1), pp. 184-196.
13. Kim J. Three dimensional motion camouflage guidance utilizing multiple leaders and one interceptor, *IET Radar, Sonar & Navigation*, 2021, Vol. 16 (3), pp. 617-631.
14. Ramachandran R., Fronda N., Preiss J., Dai Z., Sukhatme G. Resilient Multi-Robot Multi-Target Tracking, *IEEE Transactions on Automation Science and Engineering*, 2024, Vol. 21 (3), pp. 1-17.
15. Rehak B., Lynnyk A., Lynnyk V. Synchronization of Multi-Agent Systems Composed of Second-Order Underactuated Agents, *Mathematics*, 2024, Vol. 12 (21), pp. 3424.
16. Kou L., Huang Y., Zuo G., Jian L., Dou Y. Fixed-time rotating consensus control of second-order multi-agent systems, *International Journal of Robust and Nonlinear Control*, 2024, Vol. 34 (18), pp. 12031-12049.
17. Chen J., Yang Y., Qin S. A Distributed Optimization Algorithm for Fixed-Time Flocking of Second-Order Multiagent Systems, *IEEE Transactions on Network Science and Engineering*, 2023, Vol. 11 (1), pp. 152-162.
18. Yin Y, Shi Y, Liu F, [et al.]. Second-order consensus for heterogeneous multi-agent systems with input constraints, *Neurocomputing*, 2019, Vol. 351, pp. 43-50.
19. Natalia Y., Zababurin K. Matrix Laplace transform, *Boletin de la Sociedad Matemática Mexicana*, 2023, Vol. 29 (3), pp. 1-21.
20. Olson N., Andrews J. A Matrix Exponential Generalization of the Laplace Transform of Poisson Shot Noise, *IEEE Transactions on Information Theory*, 2024, Vol. 71 (1), pp. 396-412.
21. Bouchenak A., Horani M. Al H., Younis J. [et al.]. Fractional Laplace transform for matrix valued functions with applications, *Arab Journal of Basic and Applied Sciences*, 2022, Vol. 29 (1), pp. 330-336.
22. Rani D., Mishra V. Laplace Transform Inversion using Bernstein Operational Matrix of Integration and its Application to Differential and Integral Equations, *Proceedings Mathematical Sciences*, 2020, Vol. 130 (1), pp. 60-89.

Нанданвар Анудж – Индийский технологический институт Манди; e-mail: anujnandanwar@gmail.com; Парашар-роуд, Техсил-Садар, недалеко от Катаулы, Каманд, Химачал-Прадеш; тел.: +918989021051; доктор PhD; научный сотрудник.

Рыбак Лариса Александровна – Федеральный исследовательский центр «Информатики и управления» Российской академии наук; e-mail: rlbgtu@gmail.com; г. Москва, Россия; тел.: +79511307230; д.т.н.; г.н.с.

Дьяконов Дмитрий Алексеевич – Белгородский государственный технологический университет им. В.Г. Шухова; e-mail: furno.xl@yandex.ru; г. Белгород, Россия; тел.: +79606958167; инженер-исследователь.

Nandanwar Anuj – Indian Institute of Technology Mandi; e-mail: anujnandanwar@gmail.com; Parashar Road, Tehsil Sadar, Near Kataula, Kamand, Himachal Pradesh; phone: +918989021051; PhD; research associate.

Rybak Larisa Alexandrovna – Federal Research Center “Computer Science and Control” of the Russian Academy of Sciences (FRC CSC RAS); e-mail: rlbgtu@gmail.com; Moscow, Russia; phone: +79511307230; dr. of eng. sc.; chief researcher.

Dyakonov Dmitry Alekseevich – Belgorod State Technological University named after V.G. Shukhov; e-mail: furno.xl@yandex.ru; Belgorod, Russia; phone: +79606958167; research engineer.

УДК 007.51

DOI 10.18522/2311-3103-2025-5-83-93

Д.Г. Макоева, И.Р. Тлупов, А.О. Шогенов

ЕСТЕСТВЕННО-ЯЗЫКОВОЕ УПРАВЛЕНИЕ СТРОИТЕЛЬНЫМИ РОБОТЕХНИЧЕСКИМИ СИСТЕМАМИ

Исследование нацелено на исследование потенциала систем управления строительными роботами посредством естественного языка. Именно отсутствие надежных систем обработки естественного языка служит тем сдерживающим фактором, что не дает интеллектуальной робототехнике в полной мере раскрыть свои потенциал. Работа дает обзор современных роботизированных строительных систем, которые используются для облегчения и улучшения строительных и инженерных процессов и задач. Объединяет эти все системы отсутствие естественно-языкового управления. В настоящей статье мы представляем принципы, алгоритмы и методы, позволяющие интеллектуальному агенту проникать в суть контекста ситуации, разворачивающейся на поле строительных и инженерных задач. В основе подхода лежит мультиагентная нейрокогнитивная архитектура, служащая своеобразным инструментом для моделирования процесса автоматической интерпретации фраз, взятых из ограниченного подмножества естественного языка. Чтобы интеллектуальный агент смог верно интерпретировать входящее сообщение, ему необходимо безошибочно определить условия, действия, свойства и отношения, имеющие место в системе «интеллектуальный агент – окружающая среда». Только после этого агент обретает способность интерпретировать контекст текущего диалога и генерировать высказывания, необходимые для проектирования кооперативного поведения, направленного на совместное преодоление технических преград. Одной из наиболее распространенных задач, требующих своего решения в быстроразвивающейся области робототехники, является разработка диалоговой системы управления, способной координировать совместное человеко-машинное поведение и интерпретировать цели и условия миссий, изложенные на естественном языке. Система управления, опирающаяся на естественный язык, является неотъемлемой частью интеллектуальной системы, фундаментом которой служит самоорганизующаяся мультиагентная нейрокогнитивная архитектура. Ее главная цель – наладить беспрепятственное общение между человеко-машинными коллективами, для того чтобы они могли совместно ставить, описывать и успешно выполнять сложные строительные задачи. основополагающим элементом подхода является мультиагентность, позволяющая системе принятия решений робота быть гибкой, адаптивной и непрерывно расширять диапазон своих знаний, генерируя вопросы, необходимые для дальнейшей работы.

Мультиагентная система; нейрокогнитивная архитектура; естественно-языковое управление; робототехнические системы.

D.G. Makoeva, I.R. Tlupov, A.O. Shogenov

NATURAL LANGUAGE CONTROL OF CONSTRUCTION ROBOTIC SYSTEMS

The study aims to investigate the potential of natural language control systems for construction robots. It is the lack of reliable natural language processing systems that serves as a limiting factor that prevents intelligent robotics from fully realizing its potential. The work provides an overview of modern robotic construction systems that are used to facilitate and improve construction and engineering processes and tasks. What unites all these systems is the lack of natural language control. In this paper, we present principles, algorithms, and methods that allow an intelligent agent to penetrate the essence of the context of a situation unfolding in the field of construction and engineering tasks. The approach is based