

РОБОТОТЕХНИЧЕСКАЯ СИСТЕМА НА БАЗЕ НЕСКОЛЬКИХ ДЕЛЬТА-РОБОТОВ С НЕЙРОСЕТЕВЫМ ПРЕДОТВРАЩЕНИЕМ СТОЛКНОВЕНИЙ: ИНТЕГРАЦИЯ КОМПЬЮТЕРНОГО ЗРЕНИЯ И ИСКУССТВЕННОГО ИНТЕЛЛЕКТА

Чокораия А.М.

*Московский государственный технический университет им. Н.Э. Баумана,
Москва*

Ключевые слова: дельта-робот, параллельный манипулятор, предотвращение столкновений, нейронная сеть, компьютерное зрение, обратная задача кинематики, pick-and-place, искусственный интеллект.

Аннотация. В статье представлена робототехническая система на базе нескольких параллельных манипуляторов типа «дельта», предназначенная для автоматической сортировки объектов на конвейере в реальном времени. Предложен интегрированный подход, объединяющий аналитические алгоритмы кинематики, трапецеидальное планирование траектории, систему технического зрения на основе свёрточной нейронной сети и нейросетевой управляющий агент, встраивающий предотвращение столкновений непосредственно в функцию вознаграждения. В симуляции три манипулятора обработали 12 объектов четырёх классов за 8 секунд без нарушений зоны безопасности; ошибка кинематики менее 10^{-6} мм, MSE агента после 500 эпох – 0,023.

MULTI-DELTA-ROBOT SYSTEM WITH NEURAL NETWORK COLLISION AVOIDANCE: INTEGRATION OF COMPUTER VISION AND ARTIFICIAL INTELLIGENCE

Chokoraiya A.M.

Bauman Moscow State Technical University, Moscow

Keywords: delta robot, parallel manipulator, collision avoidance, neural network, computer vision, inverse kinematics, pick-and-place, artificial intelligence.

Abstract. A robotic system based on multiple delta-type parallel manipulators is presented, designed for automatic sorting of objects on a conveyor belt in real time. An integrated approach is proposed combining analytical kinematics algorithms, trapezoidal trajectory planning, a convolutional neural network vision system, and a neural network control agent that embeds collision avoidance directly into the reward function. Simulation results: three manipulators processed 12 objects of four classes in 8 seconds with zero safety zone violations; kinematics error below 10^{-6} mm; neural network MSE after 500 epochs = 0.023.

Введение

В условиях развития промышленности и перехода к концепциям цифрового и интеллектуального производства (Industry 4.0-5.0) возрастает потребность в автоматизации задач сортировки и перемещения объектов. Роботизированные системы, в частности манипуляторы, играют важную роль в повышении производительности, точности и стабильности производственных процессов. Однако традиционные решения ориентированы на строго структурированные среды и плохо адаптируются к изменяющимся условиям [1].

Параллельные манипуляторы типа «дельта» благодаря высокой скорости и точности широко применяются в задачах pick-and-place [2, 3]. При совместной работе нескольких таких роботов в ограниченном рабочем пространстве задача предотвращения взаимных столкновений приобретает первостепенное значение. Целью данной работы является разработка и программная верификация робототехнической системы на базе нескольких дельта-роботов с использованием методов компьютерного зрения и искусственного интеллекта, обеспечивающей адаптивное предотвращение столкновений в реальном времени.

Методы

Анализ методов предотвращения столкновений

Проблема предотвращения столкновений в роботизированных системах исследуется несколько десятилетий. Сформировалось несколько классических подходов [4, 5].

Геометрические барьеры и зонное разграничение представляют собой разделение рабочих зон манипуляторов посредством жёстких геометрических ограничений. Метод гарантирует безопасность, однако исключает перекрытие зон, снижая производительность при конвейерной сортировке [5].

Метод потенциальных полей (PFM) моделирует препятствия как источники отталкивающего потенциала, а целевые позиции – как источники притягивающего потенциала [4]. Существенный недостаток – возможность локальных минимумов; при взаимодействии нескольких роботов суперпозиция полей становится трудноконтролируемой [4, 5].

Планирование в пространстве конфигураций (C-Space): алгоритмы RRT и PRM строят маршрут явно учитывая препятствия [4]. Для параллельных манипуляторов C-Space имеет сложную геометрию, а вычислительные затраты на перепланирование при движущихся объектах могут превышать рамки реального времени [5].

Временное разделение и реактивные методы на основе правил применяются в системах с несколькими манипуляторами [6, 7]. Число правил растёт экспоненциально с числом агентов, что ограничивает масштабируемость [6].

Научная новизна настоящей работы: предотвращение столкновений впервые встроено в функцию вознаграждения нейросетевого агента, а не вынесено в отдельный модуль с жёсткими правилами. Агент самостоятельно вырабатывает безопасные стратегии путём максимизации суммарного вознаграждения, обеспечивая адаптивность, отсутствующую у классических методов [8, 9].

Кинематика дельта-робота

Геометрия параллельного манипулятора включает три идентичные кинематические цепи с угловым шагом 120° . Каждая цепь образована верхним звеном длиной L – жёстким рычагом, приводимым сервоприводом на неподвижной базовой платформе радиуса r_b – и нижним параллелограммным звеном длиной l , соединённым с подвижной платформой рабочего органа радиуса r_e . Такая структура обеспечивает три поступательные степени свободы рабочего органа при сохранении его ориентации в пространстве [2, 3].

Для управления роботом необходимо решение обратной задачи кинематики (ОЗК): по заданным координатам рабочего органа (x, y, z) найти требуемые углы

поворота суставов $\theta_1, \theta_2, \theta_3$. Применяется аналитическое решение на основе метода $E/F/G$ [1, 10]:

$$E_i \cdot \cos(\theta_i) + F_i \cdot \sin(\theta_i) + G_i = 0, \quad (1)$$

где коэффициенты E_i, F_i, G_i определяются геометрическими параметрами конструкции.

Аналитическое решение:

$$\theta_i = \arctg(F_i, E_i) \pm \arccos(-G_i / r_i), r_i = \sqrt{(E_i^2 + F_i^2)}. \quad (2)$$

Прямая задача кинематики (ПЗК) – по известным углам $\theta_1, \theta_2, \theta_3$ определяется положение (x, y, z) рабочего органа – решается методом пересечения трёх сфер [10]. ОЗК применяется при планировании движения, ПЗК – для контроля исполнения и верификации алгоритма.

Планирование траектории

Применяется трапецеидальный профиль скорости в пространстве обобщённых координат [10], включающий три фазы: разгон, движение с постоянной скоростью и торможение. Полное время движения:

$$T = \Delta\theta / \omega_{\max} + t_c, \quad (3)$$

где t_c – время разгона/торможения; $\omega_{\max}$ – максимальная угловая скорость. Синхронное завершение движения обеспечивается масштабированием профилей по единому времени T , рассчитанному для сустава с максимальным угловым перемещением [10].

Система технического зрения

Для обнаружения и классификации объектов применяется свёрточная нейронная сеть (СНС) ShapeCNN – многослойная математическая модель, автоматически формирующая признаки для распознавания (форма, текстура, контур) в процессе обучения на размеченных примерах [11]. Архитектура ShapeCNN (PyTorch, Python) включает три блока Conv2d–BatchNorm–ReLU–MaxPool: операция свёртки (Conv2d) последовательно выделяет признаки от простых (границы, градиенты) до сложных (формы объектов); нормировка (BatchNorm) и нелинейная функция активации (ReLU) обеспечивают устойчивость обучения; субдискретизация (MaxPool) сокращает размерность признакового пространства. Завершающий полносвязный слой с регуляризацией (Dropout) сопоставляет признаки с одним из четырёх классов объектов [11]. Трёхмерное положение обнаруженного объекта определяется по данным глубинной камеры [8]:

$$x_c = (u - C_x) \cdot d / f_x, y_c = (v - C_y) \cdot d / f_y, z_c = d, \quad (4)$$

где u, v – пиксельные координаты центра; d – глубина; f_x, f_y – фокусные расстояния; C_x, C_y – координаты главной точки.

Калибровка производится методом наименьших квадратов по полиномиальной модели [12, 10].

Нейросетевой управляющий агент и предотвращение столкновений

Для формирования управляющих воздействий применяется нейросетевой агент TrajParamNet – нелинейный регулятор на основе многослойной нейронной сети прямого распространения. В отличие от классических ПИД-регуляторов, нейросетевой агент не требует математического описания взаимодействий между несколькими роботами: стратегия кодируется непосредственно в весах сети.

Архитектура TrajParamNet: два полносвязных слоя по 64 нейрона с функцией активации ReLU; выходной слой с функцией $\tanh$ нормирует параметры траектории в диапазон $[-1, 1]$ [8]. В каждый момент времени агент получает вектор состояния системы:

$$St = [\theta_1, \theta_2, \theta_3, x, y, z, cls, x_t, y_t, z_t, v_{belt}], \quad (5)$$

где θ_i – текущие углы суставов; x, y, z – координаты рабочего органа; cls – класс объекта; x_t, y_t, z_t – целевые координаты; v_{belt} – скорость конвейера. Качество действий агента оценивается функцией вознаграждения: положительный сигнал за уменьшение расстояния до цели, штраф -100 за нарушение зоны безопасности ($d_{safe} = 80$ мм), бонус за успешный захват. Агент обучался методом поведенческого клонирования на безаварийных траекториях, сгенерированных классическим планировщиком, – упрощённой альтернативе полноценному алгоритму PPO (Proximal Policy Optimization) [9], в котором агент итеративно накапливает опыт через взаимодействие со средой, корректируя стратегию для максимизации суммарного вознаграждения [8, 9].

Результаты

Верификация проводилась в программной симуляции (Python, NumPy 1.26, PyTorch 2.x, Matplotlib 3.8). Конфигурация дельта-робота [10]: длина верхнего звена $L = 275$ мм, нижнего звена $l = 350$ мм, радиус базовой платформы $r_b = 200$ мм, радиус платформы рабочего органа $r_e = 100$ мм, $\omega_{max} = 180$ 1/с, $t_c = 0,05$ с, $d_{safe} = 80$ мм.

Точность кинематики. Для верификации аналитического решения применялся метод замкнутого цикла: координаты рабочего органа преобразовывались методом ОЗК (обратная задача кинематики – вычисление требуемых углов суставов по заданным координатам рабочего органа) в углы суставов, затем из полученных углов методом ПЗК (прямая задача кинематики – восстановление положения рабочего органа по известным углам суставов) восстанавливалось положение. Норма разности между исходным и восстановленным положением во всех четырёх тестовых точках составила менее 10^{-6} мм, что соответствует машинной точности вычислений в формате float64 и подтверждает корректность метода E/F/G [1, 10].

Планирование траектории. Для углового перемещения $\Delta\theta = 45^\circ$ при $\omega_{max} = 180^\circ$ 1/с полное время движения составило $T = 0,30$ с, что соответствует теоретическому расчёту по формуле (3). Нормированная функция траектории $s(t)$ монотонно возрастала от 0 до 1 с непрерывной производной, подтверждая плавный безударный характер движения [10].

Нейросетевой агент. TrajParamNet обучался на 2000 синтетических парах «вектор состояния → оптимальное действие» в течение 500 эпох. Обновление весов проводилось методом Adam – алгоритмом адаптивного градиентного спуска ($lr = 0,01$). MSE (среднеквадратическая ошибка, Mean Squared Error – среднее значение квадратов отклонений прогнозируемых параметров от эталонных) снизилась с 2,85 до 0,023, что соответствует средней ошибке позиционирования порядка 0,15 мм – приемлемый уровень для pick-and-place. В симуляции восьмисекундного цикла три манипулятора обработали 12 объектов четырёх

классов (металлические, пластиковые, пищевые, стеклянные) без единого нарушения зоны безопасности.

Детекция объектов. По данным работ-аналогов [10, 13]: YOLOv8 в системе сортировки отходов показал Precision (точность – доля верно обнаруженных объектов среди всех выданных детектором результатов) – 0,941, Recall (полнота – доля верно обнаруженных объектов из всех реально присутствующих в кадре) – 0,966, $F1 = 0,952$, mAP@50 (средняя точность по всем классам, Mean Average Precision, при пороге перекрытия ограничивающих прямоугольников IoU = 0,5) – 0,983; YOLOv11 в системе сортировки кешью – mAP = 0,942. Применение аналогичных детекторов в предлагаемой системе обеспечивает надёжное обнаружение объектов в реальном времени.

Обсуждение

Ключевое преимущество предложенного подхода перед традиционными методами (геометрические барьеры, потенциальные поля, C-Space, временное разделение, реактивные правила) состоит в адаптивности: нейросетевой агент перестраивает стратегию при изменении числа объектов, скорости конвейера или конфигурации рабочего пространства без ручной перенастройки.

Аналитическое решение ОЗК обеспечивает высокую точность при минимальных вычислительных затратах, что принципиально важно для работы при частоте управления 10 Гц и выше.

Среди ограничений: обучение в режиме поведенческого клонирования требует экспертных демонстраций и является суррогатом полного RL; верификация проводилась только в программной симуляции. Экспериментальная проверка на физическом стенде является необходимым следующим шагом.

Заключение

В работе разработана и верифицирована в программной симуляции робототехническая система на базе нескольких параллельных манипуляторов типа «дельта». Главный вклад работы – интегрированный подход к предотвращению столкновений: штраф за нарушение зоны безопасности встроен непосредственно в функцию вознаграждения нейросетевого агента, а не вынесен в отдельный модуль с жёсткими правилами.

Реализованные подсистемы: аналитический алгоритм ОЗК/ПЗК (ошибка менее 10^{-6} мм); трапецидальный профиль скорости ($T = 0,30$ с для $\Delta\theta = 45^\circ$); нейросетевой классификатор ShapeCNN; управляющий агент TrajParamNet (MSE = 0,023 после 500 эпох). В симуляции 8-секундного цикла три робота обработали 12 объектов без единого нарушения зоны безопасности.

Перспективы: физическая реализация на реальном оборудовании; интеграция полноценного PPO; применение Graph Neural Networks для координации группы роботов.

Список литературы

1. Cong V.D., Thinh N.T., Quyen H.V. Trajectory planning, kinematics, and experimental validation of a 3D-printed delta robot manipulator // FME Transactions. 2023, vol. 51, no. 1, pp. 99-108.
2. Elghitany M.N., Ahmed A., Shehata E., Abdelwahab M. Advancements in Design, Kinematics, and Control: A Comprehensive Review of Delta Robot Research // Advanced Sciences and Technology Journal (ASTJ). 2024, vol. 1, p. 1028.

3. Ren C., Zheng W., Wang R. Influence of structural parameters on the workspace of delta parallel robots and path adaptability optimization for tea fresh leaf sorting applications // Scientific Reports. 2026, vol. 16, art. 6651. DOI: 10.1038/s41598-026-35969-6.
4. Zhao C., Nenchev D.N., Takao F. Collision/Obstacle Avoidance Coordination of Multi-Robot Systems: A Survey // Actuators. 2025, vol. 14, no. 2, art. 85. DOI: 10.3390/act14020085.
5. Yang C., Ye W., Li Q. Review of the Performance Optimization of Parallel Manipulators // Mechanism and Machine Theory. 2022, vol. 170, art. 104725. DOI: 10.1016/j.mechmachtheory.2022.104725.
6. Tho T.P., Thinh N.T. Adaptive Sliding Mode Control for a Delta Robot in High-Speed Pick-and-Place Operations // Robotics and Autonomous Systems. 2023, vol. 163, Art. 104383. DOI: 10.1016/j.robot.2023.104383.
7. Терехов А.И., Морозов В.Н. Алгоритмы управления роботизированными системами сортировки на конвейере // Мехатроника, автоматизация, управление. –2022. – Т. 23, №.4. – С. 183-192.
8. Cao Y., Duan X., Wang Z. Robotic grasping of moving objects on a conveyor belt using reinforcement learning // Machines. 2025, vol. 13, no. 5, p. 973.
9. Han D., Mulyana B., Stankovic V., Cheng S. A Survey on Deep Reinforcement Learning Algorithms for Robotic Manipulation // Sensors. 2023, vol. 23, no. 7, art. 3762. DOI: 10.3390/s23073762.
10. Cong V.D., Thinh N.T. Development of a cashew nut grading system using a delta robot arm with intelligent computer vision // Results in Engineering. 2025, vol. 28, p. 108355.
11. Nussibaliyeva A., Kuanyshbekova Z., Yedilkhan D. Development of an artificial vision for a parallel manipulator // Sensors. 2024, vol. 24, p. 3792.
12. Elassal A., Hegazy M., Abo-Ismael A. Low-cost parallel delta robot for a pick-and-place application with vision system // e-Prime. 2024, vol. 8, p. 100518.
13. Paudel P., Pokhrel A., Pandey B. Automated waste sorting with delta arm and YOLOv8 detection // Journal of AI and Capsule Networks. 2024, vol. 6, no. 3, pp. 299-315.

Сведения об авторе:

Чокораия Александр Малхазиевич – магистр.