

А. Ш. Джачаев, С. А. Воротников

**МЕТОД РОЯ ЧАСТИЦ ДЛЯ РЕШЕНИЯ ОБРАТНОЙ ЗАДАЧИ КИНЕМАТИКИ
МАНИПУЛЯТОРОВ С ИЗБЫТОЧНЫМ ЧИСЛОМ СТЕПЕНЕЙ ПОДВИЖНОСТИ**

Московский государственный технический университет им. Н. Э. Баумана

ajachaev@yandex.ru, s_vorotnikov@mail.ru

В статье рассматривается применение метода роя частиц в качестве алгоритма решения обратной задачи кинематики манипулятора с избыточным числом степеней подвижности. Показано, что использование этого метода для поиска оптимальной конфигурации манипулятора позволяет минимизировать ошибку позиционирования схвата. Проведено моделирование работы алгоритма для ряда тестовых задач в условиях различных начальных конфигураций манипулятора и целевых позиций и определена эффективность предложенного подхода.

Ключевые слова: система управления манипулятором, обратная задача кинематики, метод роя частиц, методы оптимизации, избыточные степени подвижности.

A. S. Jachaev, S. A. Vorotnikov

PARTICLE SWARM OPTIMIZATION METHOD FOR SOLVING THE REDUNDANT MANIPULATORS INVERSE KINEMATICS

Bauman Moscow State Technical University

The article discusses the use of the Particle Swarm Optimization method as an algorithm for solving the inverse problem of manipulator kinematics with an excessive number of degrees of freedom. It is shown that using this method to find the optimal manipulator configuration allows minimizing the gripper positioning error. The algorithm operation was simulated for a number of test problems under conditions of various initial manipulator configurations and target positions, and the efficiency of the proposed approach was determined.

Keywords: Manipulator control system, inverse kinematics problem, particle swarm optimization, optimization methods, excessive number of degrees of freedom.

Введение

В настоящее время манипуляторы находят применение в различных отраслях промышленности и сферах услуг. В рамках широкого спектра задач существуют такие, где предъявляются особые требования к области достижимости манипуляторов с учетом существующих в условиях данной задачи пространственных ограничений. Это привело к появлению манипуляторов, способных обеспечить данные требования – главным образом, *манипуляторов с избыточным числом степеней подвижности* (МИСП). Наличие дополнительных степеней подвижности позволяет МИСП корректно позиционировать схват по положению и ориентации на тех участках рабочей области, доступ к которым затруднен ввиду особенностей выполняемой операции [1]. Например, при выполнении сварочных операций такой особенно-

стью может являться форма свариваемых изделий. На рис. 1 показано преимущество 7-степенного робота-манипулятора (рис. 1, б) перед обычным 6-степенным роботом-манипулятором (рис. 1, а) – наличие дополнительной степени подвижности позволяет обрабатывать весь периметр сварного шва [2].

Однако наличие дополнительных степеней подвижности приводит к изменениям в синтезе системы управления манипулятором, в частности, в решении *обратной задачи кинематики* (ОЗК). Решение ОЗК заключается в нахождении обобщенных координат сочленений манипулятора по заданным значениям положения и ориентации схвата [3] (рис. 2, а). В отличие от нее прямая задача кинематики (ПЗК) определяет по заданным значениям обобщенных координат соответствующие значения положения и ориентации схвата (рис. 2, б).

а



б



Рис. 1. Демонстрация сварочной операции:

а – 6-степенной робот-манипулятор; б – 7-степенной робот-манипулятор

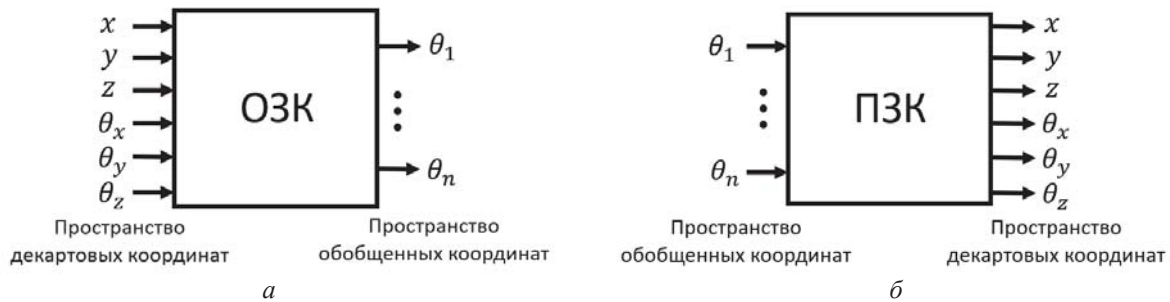


Рис. 2. Две задачи кинематики манипулятора:
а – обратная (OЗК) и б – прямая (ПЗК)

Положение OЗК в общей структуре системы управления манипулятором представлено на рис. 3.

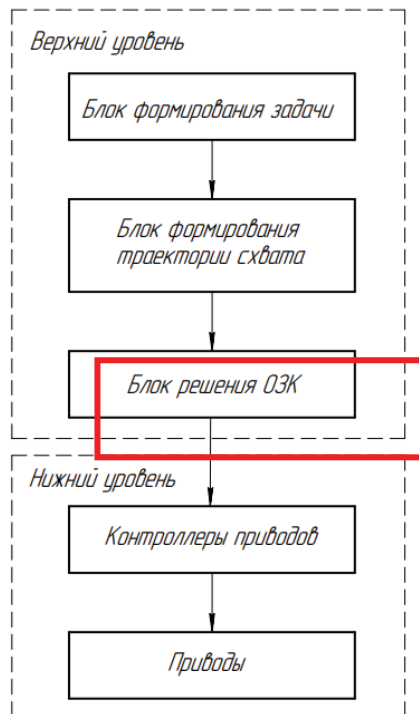


Рис. 3. Обобщенная иерархия задач управления манипулятором

В случае МИСП число степеней подвижности манипулятора превышает число пространственных степеней свободы схвата. Такая избыточность увеличивает область достижимости манипулятора, но приводит к появлению множественности возможных решений, что накладывает ограничения на выбор подходящего метода решения OЗК [4].

Так, традиционные *аналитические методы* решения OЗК основываются на выводе явных математических выражений для переменных обобщенных координат, что возможно только в сравнительно простых системах с ограниченным числом степеней подвижности [5]. В слу-

чае избыточных степеней подвижности аналитические методы не подходят для практического применения, поскольку сложность вывода уравнений значительно возрастает из-за множественности возможных решений [6]. Более того, такой подход не позволяет эффективно учитывать дополнительные критерии оптимизации, такие как отсутствие коллизий [7].

Численные методы, включая градиентный спуск и алгоритмы Ньютона, часто применяются для поиска решений в многомерных пространствах [8]. Однако и они имеют свои ограничения. Во-первых, данные методы сильно зависят от выбора начальной точки, что может привести к застреванию в локальном оптимуме [9]. Во-вторых, вычислительная сложность этих методов возрастает экспоненциально с увеличением размерности задачи, что существенно снижает их эффективность в случае МИСП [10].

Вопрос выбора оптимального подхода к решению OЗК стоит особенно остро в тех случаях, когда робот-манипулятор функционирует в динамичной среде, поскольку в этом случае неопределенность рабочей области может потребовать от системы управления проводить непрерывные вычислительные операции в режиме реального времени для нахождения актуального в текущий момент решения OЗК [11]. В качестве целевого примера предлагается рассмотреть семиосевой МИСП «KUKAiwa 14 R820», ассистирующий человеку при выполнении сборочно-монтажных операций (рис. 4) [12].

Как следует из рис. 4, в рабочей области МИСП присутствует ряд динамических объектов, что требует при поиске оптимальной конфигурации манипулятора учесть еще один критерий – *отсутствие коллизий*, где под коллизиями понимается столкновение с объектами окружающего пространства. Геометрия данных объектов выступает дополнительным фактором, формирующим пространство поиска возможных решений OЗК.



Рис. 4. Совместное выполнение сборочно-монтажных операций с помощью МИСП

Таким образом, приведенные особенности обуславливают необходимость поиска альтернативных подходов, которые способны эффективно справляться с многокритериальными оптимизационными задачами в рамках управления МИСП. Данная проблема была исследована в ряде работ, где авторы предложили использовать нейросетевые и эвристические методы оптимизации для повышения эффективности решения ОЗК [13–15]. Однако и в этих публикациях не учитывалась возможность возникновения коллизий при движении манипулятора; кроме того, для самих методов характерно существенное снижение вычислительной эффективности при увеличении размерности пространства поиска, а также невозможность достижения глобального оптимума для ряда случаев [16; 17].

В данной работе в качестве альтернативного подхода предлагается использовать *метод роя частиц* (Particle Swarm Optimization – PSO), осно-

ванный на применении роевого интеллекта и имитирующий поведение природных популяций, который демонстрирует высокую эффективность при решении оптимизационных задач в высокоразмерных пространствах поиска [18]. Благодаря этому алгоритм метода роя частиц уже получил применение в различных прикладных научных областях, включая робототехнику [19].

Применение метода роя частиц для решения ОЗК МИСП

Следует подробнее рассмотреть метод роя частиц. В основе этого метода, предложенного Р. Эберхартом и Дж. Кеннеди в 1995 г., лежит процесс, схожий с поведением птиц в стае и рыб в косяке: группа частиц (аналогично стае птиц) совершает поиск оптимального состояния в пространстве решений. Каждая частица делится своей текущей лучшей позицией с одной или несколькими другими частицами в рое после смещения в пространстве решений. В результате каждая частица определяет свой следующий шаг в пространстве решений на основе траектории каждой другой частицы (*социальный компонент*) и инерциальной составляющей (*инерциальный компонент*), зависящей от своего состояния на предыдущем шаге (*когнитивный компонент*). После того как все частицы обновляют свои позиции в текущей итерации, начинается следующая итерация процесса поиска, исследующая области, близкие к оптимуму. Результатом всего процесса является нахождение глобального оптимума целевой функции, куда сходится весь рой. Блок-схема алгоритма с пояснениями представлена на рис. 5.

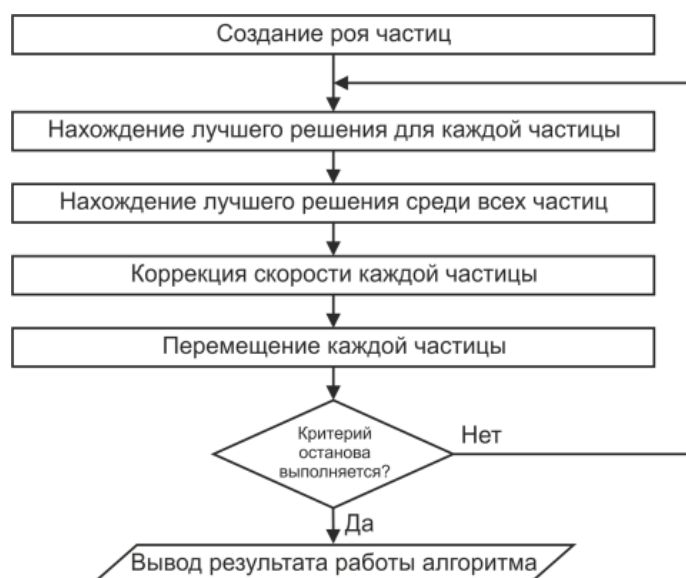


Рис. 5. Блок-схема алгоритма PSO

Применим этот метод при постановке оптимизационной задачи в рамках решения ОЗК манипулятора. Для наглядности представления рассмотрим плоский двухзвенный манипулятор,

схват которого необходимо переместить в целевую точку (рис. 6). Помимо манипулятора в рабочей области также присутствует препятствие, столкновение с которым нужно избежать.

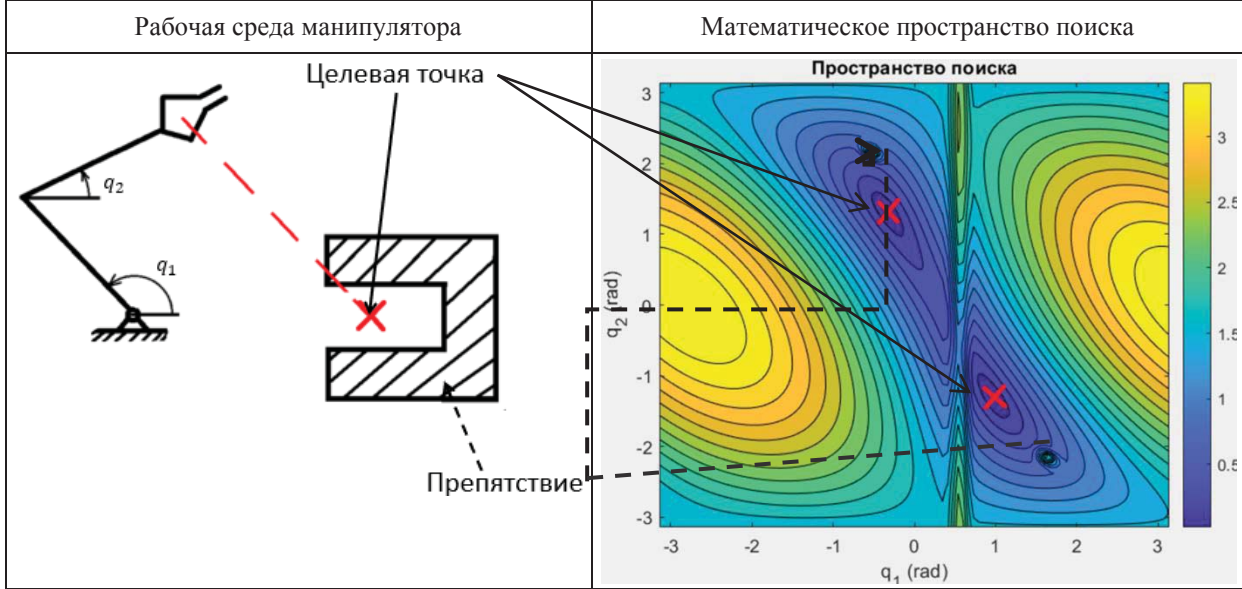


Рис. 6. Иллюстрация постановки оптимизационной задачи в рамках решения ОЗК

В методе роя частиц положение $x_{i,j}^{t+1}$ каждой частицы на текущем шаге обновляется согласно выражению $x_{i,j}^{t+1} = x_{i,j}^t + v_{i,j}^{t+1}$, $x_{i,j}^t$ – положение частицы на предыдущем шаге, $v_{i,j}^{t+1}$ – вектор перемещения частицы [20]. В рамках решения ОЗК манипулятора с двумя степенями подвижности каждая частица $x_i = (q_1, q_2)^T$ – это вектор из двух элементов, соответствующий

значениям обобщенных координат соединений манипулятора.

Далее необходимо определить целевую функцию $f(\mathbf{x}_i)$. Как правило, для этой цели используются уравнения ошибки по положению $e_p(\mathbf{q})$ и ориентации $e_o(\mathbf{q})$ схвата [21].

Уравнение ошибки по положению $e_p(\mathbf{q})$ выглядит следующим образом:

$$e_p(\mathbf{q}) = \|\mathbf{p}_d - \mathbf{p}(\mathbf{q})\| = \sqrt{(x_d - x(\mathbf{q}))^2 + (y_d - y(\mathbf{q}))^2 + (z_d - z(\mathbf{q}))^2}, \quad (1)$$

где $\mathbf{p}_d = (x_d, y_d, z_d)$ – желаемое положение схвата манипулятора.

Уравнение ошибки по ориентации $e_o(\mathbf{q})$ можно записать аналогично:

$$e_o(\mathbf{q}) = \|\mathbf{o}_d - \mathbf{o}(\mathbf{q})\| = \sqrt{(y_d - y(\mathbf{q}))^2 + (p_d - p(\mathbf{q}))^2 + (r_d - r(\mathbf{q}))^2}, \quad (2)$$

где $\mathbf{o}_d = (y_d, p_d, r_d)$ – желаемая ориентация схвата манипулятора.

Кроме того, поскольку требуется не учитывать те конфигурации манипулятора, при которых он может пересечь препятствие, к общей величине ошибки необходимо добавить соответствующий множитель $\xi = \begin{cases} 0 \\ 1 \end{cases}$, который будет принимать значения 0 и 1 при наличии и отсутствии коллизий соответственно.

Таким образом, итоговое выражение для целевой функции $f(\mathbf{x}_i) = e(q_j)$ примет следующий вид:

$$f(\mathbf{x}_i) = e(q_j) = \frac{\xi}{1 + (\alpha e_p(\mathbf{q}) + \beta e_o(\mathbf{q}))} \rightarrow 1, \quad (3)$$

где α, β – весовые коэффициенты, учитывающие влияние ошибки по положению и по ориентации схвата соответственно.

В рассматриваемом случае пространство поиска возможных решений ОЗК будет двух-

мерным, где каждому вектору $\mathbf{q} = \begin{pmatrix} q_1 \\ q_2 \end{pmatrix}$ соответствует некоторое значение целевой функции $e(\mathbf{q})$.

Очевидно, в случае МИСП, имеющего семь степеней подвижности, пространство поиска будет являться семимерным пространством, а искомый вектор – вектором $\mathbf{q} = \begin{pmatrix} q_1 \\ \dots \\ q_7 \end{pmatrix}$, содержащим семь значений обобщенных координат.

В остальном применение метода роя частиц для ОЗК МИСП не отличается от ранее рассмотренного случая двухзвенного манипулятора.

Как следует из (1)–(3), целевая функция

$$T_n^0(\mathbf{q}) = T_1(q_1)T_2(q_2) \dots T_n(q_n) = \begin{pmatrix} n_x & s_x & a_x & p_x \\ n_y & s_y & a_y & p_y \\ n_z & s_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} R_n^0 & p_n^0 \\ 0 & 1 \end{pmatrix},$$

здесь векторы $(n_x, n_y, n_z)^T$, $(s_x, s_y, s_z)^T$, $(a_x, a_y, a_z)^T$ определяют направления осей x_n, y_n, z_n относительно нулевой системы координат; R_n^0 – матрица вращения n -й системы координат; p_n^0 – вектор линейного смещения начала координат n -й системы относительно нулевой системы координат; $T_j(\mathbf{q})$ – матрица однородных преобразований для j -й системы координат относительно $(j-1)$ -й системы координат.

требует нахождения положения схвата манипулятора, соответствующего текущей конфигурации $\mathbf{q}$ его сочленений, то есть решения прямой задачи кинематики (ПЗК).

Формула ПЗК имеет следующий вид [22]:

$$\begin{pmatrix} x_0 \\ y_0 \\ z_0 \\ 1 \end{pmatrix} = T_n^0 \begin{pmatrix} x_n \\ y_n \\ z_n \\ 1 \end{pmatrix},$$

где $(x_0, y_0, z_0)^T, (x_n, y_n, z_n)^T$ – векторы начальной и конечной систем координат соответственно; T_n^0 – матрица однородных преобразований для n -й системы координат относительно нулевой системы координат, которая может быть вычислена по следующей формуле [22]:

Таким образом, оптимизационный алгоритм метода роя частиц применительно к решению ОЗК МИСП примет вид (рис. 7).

Полученный алгоритм был реализован в виде программного кода, где в ходе эмпирического анализа была выбрана численность роя, равная 500, а также коэффициенты α и β целевой функции, равные 1.

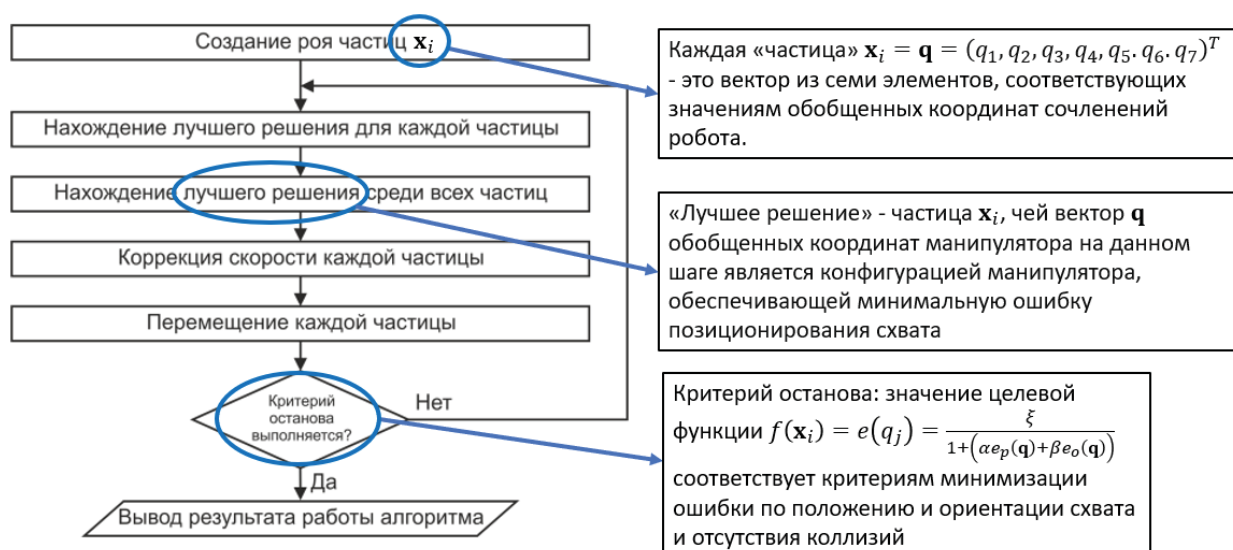


Рис. 7. Применение алгоритма метода роя частиц для решения ОЗК МИСП

Моделирование решения ОЗК для тестовых примеров

Как было упомянуто ранее, разработка оптимального алгоритмического решения ОЗК особенно актуальна для тех задач, которые робот-манипулятор выполняет в режиме реального времени. В связи с этим критически важным критерием оценки эффективности алгоритма является время решения им ОЗК. В данном случае решение ОЗК является итерационным процессом, поэтому это время соответствует длительности цикла сходимости целевой функции (3).

Длительность цикла сходимости целевой

функции изначально представляет собой количество итераций, которое должен выполнить алгоритм для того, чтобы найти значение аргумента, соответствующего оптимальному значению целевой функции. В рамках практического применения значение имеет временная оценка длительности. Соответствие числа итераций и времени выполнения всего расчета определяется аппаратным обеспечением, в частности, процессором, выполняющим вычисления. В табл. 1 представлен период выполнения одной итерации для целевого процессора блока управления МИСП и высокопроизводительного тестового процессора.

Таблица 1

Длительность итерационного цикла ОЗК

Процессор	Intel Pentium G850 (блок управления KRC2) [23]	Intel Xeon E3-1275 v5 (тестовый)
Продолжительность	2,6 мс [25]	1,5 мс [24]

В задачах реального времени цикл решения ОЗК манипулятора не должен превышать 100 мс [26]. Следовательно, за это время вычислительная мощность процессора блока управления МИСП позволит выполнить $\frac{100}{2.6} \sim 38$ итераций.

Это значение было использовано для последующей оценки результатов моделирования для двух тестовых примеров с различными конфигурациями МИСП и целевых точек (рис. 8, а, в).

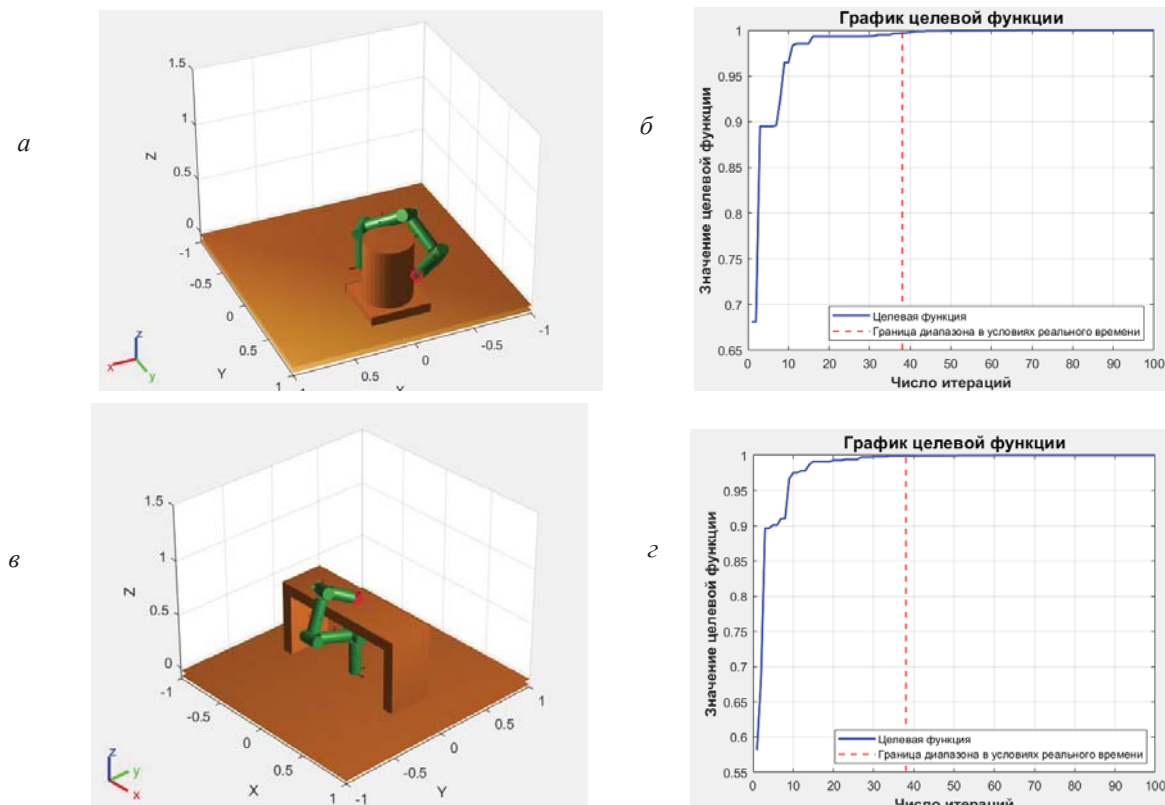


Рис. 8. Результаты моделирования алгоритма:
а, в – рабочие области манипулятора, б, г – графики сходимости целевой функции

Для каждого из примеров в ходе моделирования было проведено пять экспериментов. В табл. 2 представлены значения целевой функции по истечении 38 итераций для каждого из модельных экспериментов в обоих примерах. Поскольку метод роя частиц является

стохастическим методом, на основе результатов моделирования можно определить математическое ожидание $M_1(F) = 0,99913$, $M_2(F) = 0,99934$ и дисперсию $D_1(F) = 4,88 \cdot 10^{-7}$, $D_2(F) = 7,87 \cdot 10^{-7}$.

Таблица 2

Результаты экспериментов

№	Значение целевой функции f_i					$M(F)$	$D(F)$
	1	2	3	4	5		
1	0,99800	0,99915	0,99994	0,99976	0,99880	0,99913	$4,88 \cdot 10^{-7}$
2	0,99760	0,99976	0,99947	0,99989	0,99998	0,99934	$7,87 \cdot 10^{-7}$

Отсюда среднее значение математического ожидания и дисперсии составляет $M(F) = 0,99924$ и $D(F) = 6,38 \cdot 10^{-7}$ соответственно. Из формулы (3) можно вывести, что данное среднее значение математического ожидания соответствует ошибке положения схвата, равной $\frac{1}{0,99924} - 1 = 0,00076$ м = 0,7 мм, а также ошибке ориентации схвата, равной $\frac{1}{0,99924} - 1 = 0,00076$ рад = 0,044°.

Таким образом, полученные результаты подтверждают эффективность предложенного подхода для решения ОЗК МИСП.

Заключение

В данной статье оптимизационный метод роя частиц был применен для решения ОЗК МИСП, функционирующего в реальном времени в условиях недетерминированной рабочей среды, что позволило получить эффективное алгоритмическое решение, применимое для управления реальным МИСП. По результатам моделирования полученного решения было выявлено, что в условиях реального времени данный алгоритм позволяет решать ОЗК МИСП с точностью, соответствующей величине ошибки 0,7 мм по положению и 0,044° по ориентации схвата. Предложенный алгоритм отвечает критерию отсутствия коллизий, под которыми понимается возможность столкновения с объектами окружающего пространства, обеспечивая при этом глобальную сходимость за меньшее время.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Коротков, В. И. Управление мобильным манипуляционным роботом в задаче адаптивной обработки зеленых насаждений / В. И. Коротков, С. А. Воронников, Н. А. Выборнов // Прикаспийский журнал: управление и высокие технологии. – 2016. – № 2 (34). – С. 48–58.

2. Jean-Philippe, J. Top 9 Robotic Highlights from IREX and IROS 2013. [Электронный ресурс] – Режим доступа: <https://blog.robotiq.com/bid/68745/Top-9-Robotic-Highlights-from-IREX-and-IROS-2013>.

3. Sanz, P. Robotics: Modeling, planning, and control / P. Sanz // IEEE Robotics & Automation Magazine. – 2009. – V. 16. № 4. – P. 101.

4. Nakamura, Y. Advanced Robotics: Redundancy and Optimization. – Addison-Wesley, 1991. – 337 p.

5. Angeles, J. Fundamentals of Robotic Mechanical Systems. – Springer, 2003. – P. 523.

6. Братчиков, С. А. Решение обратной задачи кинематики манипулятора / С. А. Братчиков, Е. А. Абрамова, Ю. В. Федосов // Вестник Томского государственного университета. – 2021. – № 56. – С. 4–11.

7. Kucuk, S. Inverse kinematics solutions for industrial robot manipulators with offset wrists / S. Kucuk, Z. Bingul // Appl. Math. Model. – 2014. – Vol. 38, № 7–8. – P. 1983–1999.

8. Колтыгин, Д. С. Аналитический и численный методы решения обратной задачи кинематики для робота Delta / Д. С. Колтыгин, И. А. Седельников, Н. В. Петухов // iPolytech Journal. – 2017. – Т. 21, № 5. – С. 87–96.

9. Борисов, О. И. Методы управления робототехническими приложениями: учебное пособие / О. И. Борисов, В. С. Громов, А. А. Пыркин. – СПб.: Университет ИТМО, 2016. – 108 с.

10. Cormen, T. H. Introduction to Algorithms / T. H. Cormen, C. E. Leiserson, R. L. Rivest, C. Stein. – MIT Press, 2009. – P. 1313.

11. Cortes, J. A random loop generator for planning the motions of closed kinematic chains using PRM methods / J. Cortes, T. Simeon, J. P. Laumond // Proceedings of the IEEE International Conference on Robotics and Automation (ICRA). – 2002. – P. 2141–2146.

12. Sebastian, S. Eenvoudig leren programmeren met een bouwsteenrobot. [Электронный ресурс] / S. Sebastian. – Режим доступа: <https://www.kuka.com/nl-be/onderneming/pers/news/2020/05/doe-het-zelf-robot>.

13. Wu, D. T-IK: an efficient multi-objective evolutionary algorithm for analytical inverse kinematics of redundant manipulator / D. Wu, G. Hou, W. Qiu, et al. // IEEE RobotAutom Lett. – 2021. – Vol. 6. – P. 8474–8481.

14. Koganezawa, K. A fast method of solving inverse kinematics of redundant manipulators / K. Koganezawa // Preprints of the Fourth IFAC Symposium on Robot Control – Capri, Italy, 1994. – P. 369–374.

15. Галемов, Р. Т. Комбинированный поисковой метод решения обратной задачи кинематики многозвенного манипулятора / Р. Т. Галемов, Г. Б. Масальский // Мехатроника, автоматизация, управление. – 2018. – Т. 19, № 7. – С. 464–473.

16. Goldberg, D. E. Genetic Algorithms in Search, Optimization, and Machine Learning / D. E. Goldberg. – Addison-Wesley, 1989. – 412 p.

17. Buss, S. R. Introduction to inverse kinematics with Jacobian transpose, pseudoinverse and damped least squares methods / S. R. Buss // IEEE Journal of Robotics and Automation. – 2004. – V. 17. – P. 1–19.

18. Kennedy, J. Particle Swarm Optimization / J. Kennedy, R. Eberhart // Proceedings of the IEEE International Conference on Neural Networks. – 1995. – V. IV. – P. 1942–1948.

19. Poli, R. Particle swarm optimization / R. Poli, J. Kennedy, T. Blackwell // Swarm Intelligence. – 2007. – V. 1. – P. 33–57.

20. Shi, Y. A modified particle swarm optimizer / Y. Shi, R. C. Eberhart // Proceedings of IEEE International Conference on Evolutionary Computation. – 1998. – P. 69–73.

21. Starke, S. Evolutionary multi objective inverse kinematics on highly articulated and humanoid robots / S. Starke, N. Hendrich, D. Krupke, J. Zhang // In Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. – 2017. – P. 6959–6966.

22. Зенкевич, С. Л. Основы управления манипуляционными роботами / С. Л. Зенкевич, А. С. Ющенко. – М.: Изд-во МГТУ им. Н. Э. Баумана, 2004. – 480 с.

23. <https://www.quora.com/What-type-of-processor-is-used-for-robots>.

24. Wang, C.-C. cuPSO: GPU Parallelization for Particle Swarm Optimization Algorithms / C.-C. Wang, C.-Y. Ho, C.-H. Tu, S.-H. Hung // SAC'22. – Brno, Czech Republic. – 2022. – 7 p.

25. https://www.cpubenchmark.net/compare/2672vs1116/Intel-Xeon-E3-1275-v5-vs-Intel-Pentium-G850#table_pp_ratio

26. Gomez, S. R. A faster algorithm for calculating the inverse kinematics of a general 6R manipulator for robot real-time control / S. R. Gomez, J. A. Cerrada, V. Feliu // 14th Triennial World Congress. – Beijing, China, 2015. – P. 47–62.