

УДК 681.5.015

**АЛГОРИТМЫ УПРАВЛЕНИЯ СВАРОЧНЫМИ РОБОТАМИ-МАНИПУЛЯТОРАМИ  
НА ОСНОВЕ СТАТИСТИЧЕСКОЙ МОДЕЛИ КОНФИГУРАЦИОННОГО ПРОСТРАНСТВА**

**И.Э. ИЛЮШИН, кан. техн. наук, доц. М.М. КОЖЕВНИКОВ**  
(Могилевский государственный университет продовольствия)

*Предложены новые эффективные алгоритмы управления сварочными роботами-манипуляторами на основе статистической модели конфигурационного пространства. Предложенные алгоритмы, в отличие от известных, позволяют эффективно учесть ограничения на ориентацию технологического инструмента, сложную форму элементов роботизированного комплекса, а также ограничения, накладываемые на угловые скорости движения звеньев манипулятора. Эффективность разработанных алгоритмов подтверждается результатами тестирования в экспериментальной среде моделирования роботов.*

**Ключевые слова:** роботы-манипуляторы, алгоритмы управления, конфигурационное пространство.

**Введение.** Расширение применения роботов в промышленном производстве обусловлено как стремлением к повышению производительности, так и необходимостью обеспечить высокое качество продукции при больших партиях или при частых изменениях объектов производства. Кроме того, серьезными стимулами применения промышленных роботов являются снижение их стоимости на фоне удорожания рабочей силы, недостаток квалифицированного рабочего персонала, освобождение работающих на производстве от тяжелого, интенсивного и монотонного труда, а также возможность снижения вредного влияния производства (особенно сварочного) на здоровье персонала [1].

Большинство роботов, применяемых в промышленности, представляют собой манипуляторы, управляемые микропроцессорным контроллером. Для реализации управления промышленным роботом-манипулятором необходимо решить ряд задач, в частности, необходимо совместить зоны обслуживания робота с рабочим пространством, в котором выполняется технологическая операция, таким образом, чтобы технологический инструмент мог быть доставлен в любое требуемое положение с необходимой ориентацией. Другой важной задачей является планирование перемещений: необходимо выбрать траекторию движения технологического инструмента, при этом не только траектория движения, но и законы изменения скоростей должны, с одной стороны, соответствовать требованиям технологического процесса, а с другой – возможностям робота [2].

В сварочных роботизированных технологических комплексах (РТК) используются в основном позиционно-контурные системы управления, обеспечивающие реализацию траекторий, заданных последовательностью опорных точек. В процессе управления вычисляются промежуточные точки траектории и формируются задающие воздействия для исполнительных приводов. В результате реализуется согласованное движение степеней подвижности, приводящее к требуемым изменениям положения и ориентации технологического инструмента.

В известных работах предложен ряд алгоритмов управления сборочно-сварочными роботами-манипуляторами в среде с препятствиями [3, 4], которые основаны на дискретной модели свободного от столкновений конфигурационного пространства. Задача управления при этом сводится к поиску кратчайшего пути на графе, ребрам которого ставятся в соответствие простейшие фрагменты траектории робота, при движении по которым отсутствуют столкновения и удовлетворяются конструктивные ограничения. В ряде последних исследований [5, 6] реализованы алгоритмы управления, выполняющие поиск нескольких решений, что увеличивает вероятность нахождения траектории движения за фиксированное время. В частности, в работе [7] управление реализовано с использованием концепции мультиграфа задачи движения (ТММ).

Необходимо отметить, что известные алгоритмы управления сварочными роботами-манипуляторами позволяют эффективно обойти проблему размерности конфигурационного пространства роботов [8–11], но не учитывают форму препятствий и форму звеньев манипулятора, что приводит к реализации траекторий низкого качества с большим объемом движений [8]. При использовании таких алгоритмов траектория робота может быть найдена за конечное время также лишь с определенной вероятностью, т.е. свойство «полноты» решения теряется [8, 12].

**Постановка задачи.** Рассмотрим робот-манипулятор (рис. 1), который описывается как кинематическая цепь из  $n$  поворотных сочленений  $g_i$  и звеньев  $l_i$  ( $i = 1:n$ ), в его рабочей зоне расположено некоторое множество препятствий  $B = \{B_1, B_2, \dots, B_m\}$ .

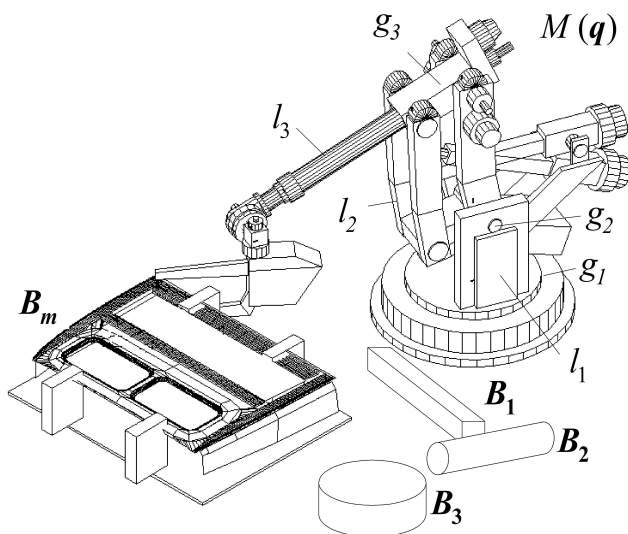


Рисунок 1. – Робот-манипулятор в среде с препятствиями

Конфигурацию этого робота зададим в виде вектора  $\mathbf{q} = [q_i]^T$ , где  $q_i$  – величины углов в сочленениях. Ограничения на изменения углов в сочленениях зададим в виде

$$\mathbf{q}_{\min} \leq \mathbf{q} \leq \mathbf{q}_{\max}, \quad (1)$$

где  $\mathbf{q}_{\min}$ ,  $\mathbf{q}_{\max}$  – векторы, определяющие нижнее и верхнее конструктивные ограничения на значения углов в сочленениях робота-манипулятора.

Каждой конфигурации ставится в соответствие вектор скоростей  $\dot{\mathbf{q}}_i = [\dot{q}_i]^T$ , где  $\dot{q}_i$  – величины угловых скоростей движения звеньев. На скорости накладываются ограничения

$$\|\dot{\mathbf{q}}\| \leq \dot{\mathbf{q}}_{\max}, \quad (2)$$

где  $\dot{\mathbf{q}}_{\max}$  – вектор, определяющий ограничения на значения угловых скоростей движения звеньев робота-манипулятора.

Зададим следующие ограничения на углы ориентации технологического инструмента  $\alpha$ ,  $\beta$ ,  $\gamma$ :

$$\alpha \in [\alpha^- \alpha^+], \quad \beta \in [\beta^- \beta^+], \quad \gamma \in [\gamma^- \gamma^+]. \quad (3)$$

Геометрическую модель робота-манипулятора, установленного в конфигурацию  $\mathbf{q}$ , зададим в виде множества  $M(\mathbf{q})$ . Тогда свободное от столкновений конфигурационное пространство определяется как

$$C_f = \{\mathbf{q} \in C \mid M(\mathbf{q}) \cap \mathbf{B} = \emptyset\}. \quad (4)$$

Задача определения пересечения множеств  $M(\mathbf{q})$  и  $\mathbf{B}$  решается путем проведения теста столкновения [3] между роботом и препятствиями по соотношению

$$\min \|M(\mathbf{q}) - \mathbf{B}\| \leq \delta, \quad (5)$$

где  $\delta$  – точность позиционирования робота-манипулятора.

Рассмотрим систему управления сборочно-сварочным роботом-манипулятором (рис. 2). Она состоит из следующих структурных элементов: элемент, формирующий заданную траекторию, контроллер, обеспечивающий её реализацию, т.е. вырабатывающий управляющие воздействия для привода робота, датчики положения и скорости, а также робот-манипулятор. Принцип работы системы следующий: текущие векторы углов в сочленениях  $\mathbf{q}_i$  и угловых скоростей  $\dot{\mathbf{q}}_i$  определяются с помощью датчиков положения и датчиков скорости соответственно, после чего сравниваются с заданными значениями  $q_{si}$  и  $\dot{q}_{si}$  соответственно ( $i = 1:g$ ). Таким образом происходит формирование сигналов рассогласования по положению  $\Delta q_i$  и по угловой скорости  $\Delta \dot{q}_i$ , и в зависимости от их значения контроллером вырабатываются соответствующие управляющие воздействия  $\tau_i$  на приводы сочленений робота.

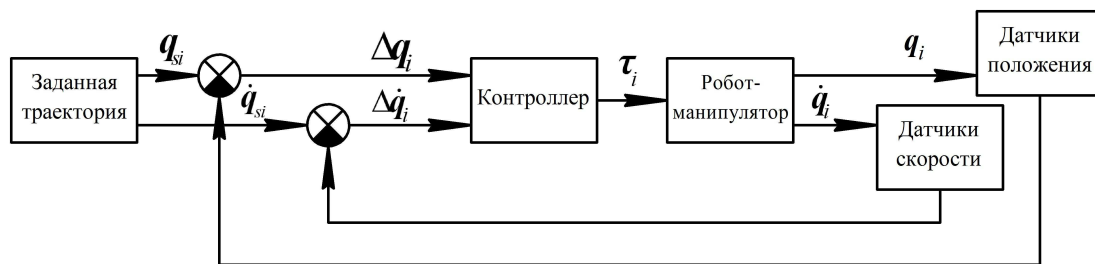


Рисунок 2. – Обобщенная блок-схема системы управления роботом-манипулятором

Стоит отметить, что контроллер, роботы-манипуляторы, а также датчики скоростей и положений имеют ряд готовых реализаций [13–15]. В связи с этим основной интерес представляет именно поиск траектории с учетом ограничений (1)–(3), поскольку это наиболее сложная и трудоемкая задача. Поэтому в данной работе предложены алгоритмы формирования заданной траектории с учетом ограничений и наличия препятствий в рабочей зоне.

**Алгоритмы управления.** Исходной информацией для алгоритмов управления сварочными роботами-манипуляторами является геометрическая модель РТК –  $M(q)$ ,  $B$ , а также координаты стартовой и целевой точек. Используя эти данные осуществляется поиск заданной траектории робота-манипулятора. Необходимо отметить, что конфигурационное пространство робота имеет достаточно сложную форму. Поэтому на практике обычно применяется приближенное описание пространства конфигураций вместо явного его расчета. В связи с этим для описания конфигурационного пространства робота применена статистическая модель, которая представляется в виде неориентированного графа  $R = (V, E)$ . Вершины  $V$  этого графа представляют собой множество свободных от столкновений конфигураций робота  $q_i$ , а также векторов допустимых угловых скоростей движения звеньев  $\dot{q}_i$ , соответствующих этим конфигурациям. Формирование множества  $V$  осуществляется следующим образом: генерируется случайная конфигурация робота-манипулятора и выполняется тест столкновения робота с препятствиями. Если столкновения нет, то генерируется соответствующий вектор допустимых скоростей  $\dot{q}_i$ , координаты которого также случайные величины, после чего конфигурация  $q_i$  и соответствующий ей вектор  $\dot{q}_i$  добавляются в множество  $V$ . Если же столкновение есть, то конфигурация отбрасывается. Ребрам  $E$  графа ставятся в соответствие прямолинейные участки траекторий между свободными от столкновений конфигурациями.

Прямолинейный участок траектории между двумя конфигурациями  $q_a$  и  $q_b$  ( $a \neq b$ ,  $q_a, q_b \in C_f$ ) задается в виде множества векторов

$$d_{ab} = \{d_k | M(d_k) \cap B = \emptyset\}, \quad (6)$$

где  $d_k = q_a + (h/Nh)(q_b - q_a)$ ,  $h = 0:Nh$ ,  $Nh > N$  – параметр дискретизации прямолинейного участка траектории.

Дискретная конфигурация робота  $q_b \in C_f$  является соседней с конфигурацией  $q_a \in C_f$ , если между ними существует прямолинейный участок траектории  $d_{ab}$  и  $D(q_a, q_b) \leq d$ , где  $D(\bullet)$  – симметричная функция, характеризующая расстояние между двумя локациями робота-манипулятора. Кроме того, осуществляется проверка прямолинейного участка на возможность перемещения робота из конфигурации  $q_a$  в конфигурацию  $q_b$  с заданной скоростью  $\dot{q}_a$ . Траектория, соединяющая стартовую  $q_{s1}$  и целевую  $q_{sg}$  конфигурации робота, представляет собой последовательность, состоящую из соседних конфигураций  $q_{s1}, q_{s2}, \dots, q_{sg} \in C_f$ , каждой из которых поставлен в соответствие вектор допустимых скоростей  $\dot{q}_{s1}, \dot{q}_{s2}, \dots, \dot{q}_{sg}$ , и прямолинейных участков, соединяющих эти конфигурации  $d_{s1s2}, d_{s2s3}, \dots, d_{(sg-1)sg}$ .

В качестве критерия качества траектории движения роботов-манипуляторов при сварке шва предлагается использовать время перемещения технологического инструмента вдоль сварного шва:

$$\tau = \rho/v, \quad (7)$$

где  $\rho$  – длина пути технологического инструмента при его движении в направлении от начальной к конечной локации;

$v$  – скорость движения технологического инструмента.

Тогда основная задача управления сварочным роботом-манипулятором может быть сформулирована следующим образом: среди всех последовательностей дискретных конфигураций  $q_{s1}, q_{s2}, \dots, q_{sg} \in C_f$ , координаты которых лежат внутри области ограниченной предельно допустимыми значениями углов в сочленениях (1), а соответствующие векторы скоростей  $\dot{q}_{s1}, \dot{q}_{s2}, \dots, \dot{q}_{sg}$  удовлетворяют ограничениям (2), найти последовательность, на которой достигается минимума критерий качества (7).

В предыдущей работе авторов данной статьи [16] предложен комбинированный алгоритм управления, основанный на синтезе траекторий роботов-манипуляторов в рабочей среде с препятствиями с использованием статистической модели и решетчатой дискретизации «насыщенных» препятствиями зон конфигурационного пространства. В соответствии с этим подходом синтез свободной от столкновений траектории осуществляется путем поиска кратчайшего расстояния на графе, ребрам которого ставятся в соответствие линейные движения робота, между промежуточными конфигурациями найденными, как посредством рандомизированной дискретизации (в зонах с малым количеством препятствий), так и решетчатой дискретизации (в зонах с большим количеством препятствий сложной формы). На основе этого алгоритма управления разработан новый алгоритм, который в отличие от известных учитывает ограничения на угловые скорости движения звеньев (2), а также предусматривает проверку прямолинейных участков на возможность прохождения по нему из одной вершины в другую с заданной скоростью. Разработанный алгоритм управления представлен в таблице 1.

Таблица 1. – Алгоритм управления сварочным роботом манипулятором

<b>Исходные данные:</b> Геометрическая модель РТК – $M(q), B$ , начальная и целевая конфигурации робота – $q_s, q_g$ .	
1:	$i \leftarrow 1; V \leftarrow q_s, q_g;$
2:	<b>повторять</b>
3:	$q_i \leftarrow Randq;$
4:	<b>если</b> $q_i \in C_f$
5:	<b>то</b> $\dot{q}_i \leftarrow Randqq;$
6:	$V \leftarrow (q_i, \dot{q}_i)$
7:	$i \leftarrow i + 1;$
8:	<b>до тех пор пока</b> $i \leq N_{max};$
9:	$k \leftarrow 1;$
10:	<b>повторять</b>
11:	$(i, j) \leftarrow Rand(1, N_{max});$
12:	<b>если</b> $((D(q_i, q_j) \leq d) \text{ и } ((EPath(q_i, q_j) \neq \emptyset) \text{ и } (STest(q_i, q_j, \dot{q}_i) = 1)))$
13:	<b>то</b> $E \leftarrow EPath(q_i, q_j)$
14:	<b>иначе</b>
15:	<b>если</b> $(D(q_i, q_j) \leq d \text{ и } (TTest(q_i) = 1 \text{ или } TTest(q_j) = 1))$
16:	<b>то</b> $(q_i^s, q_j^g) \leftarrow Connect(q_i, q_j, G);$
17:	$R \leftarrow SearchGridPath(q_i^s, q_j^g);$
18:	$V \leftarrow (EPath(q_i, q_i^s), EPath(q_j, q_j^g));$
19:	$k \leftarrow k + 1;$
20:	<b>до тех пор пока</b> $k \leq K_{max};$
21:	$path \leftarrow GraphSearch(R, q_s, q_g).$

Алгоритм использует следующие обозначения.  $Randq$  – функция генерации вектора конфигурации в допустимом диапазоне (1).  $Randqq$  – функция генерации вектора скорости с координатами с учетом ограничения (2).  $Rand$  – функция генерации случайных целых чисел  $i \neq j$  в диапазоне от 1 до  $N_{max}$ .  $EPath$  – функция расчета прямолинейного участка траектории между двумя конфигурациями  $q_i$  и  $q_j$ .  $STest$  – функция, выполняющая проверку ребра графа на возможность перемещения из конфигурации  $q_i$  в кон-

фигурацию  $q_j$  с заданной скоростью  $\dot{q}_i$ . В случае если перемещение с заданной скоростью вдоль ребра от одной конфигурации к другой возможно, функция  $STest$  возвращает 1, в противном случае возвращает 0.  $TTest$  – функция, выполняющая оценку «насыщенности» окрестности конфигурации  $q$  препятствиями за счет генерации случайных конфигураций в ее окрестности и тестирования их на столкновения. В случае, когда количество столкновений превышает допустимый предел, функция  $TTest$  возвращает 1, что означает «насыщенность» окрестности конфигурации  $q$  препятствиями.  $Connect$  – функция, выполняющая поиск конфигураций  $(q_i^s, q_j^g)$  на решетке дискретизации  $G$ , являющихся соседними с конфигурациями  $(q_i, q_j)$ .  $GraphSearch$  – функция нахождения кратчайшего пути на графе  $R$ , исходя из критерия качества (7).

Отдельного рассмотрения заслуживает процедура  $SearchGridPath$ , разработанная на основе методики, предложенной в предыдущих работах авторов данной статьи [16, 17]. Процедура используется для поиска криволинейной траектории между двумя конфигурациями робота-манипулятора  $(q_i^s, q_j^g)$  путем дискретизации окрестности данных конфигураций решеткой. Соответственно, в результате выполнения процедуры в граф  $R$  добавляются вершины и ребра, принадлежавшие криволинейному участку траектории между  $(q_i^s, q_j^g)$ . При этом скорости движения звеньев задаются постоянными  $\dot{q}_{greed}$  для каждой конфигурации. Алгоритм процедуры  $SearchGridPath$  приведен в таблице 2.

Таблица 2. – Алгоритм процедуры  $SearchGridPath$

<b>Исходные данные:</b> Геометрическая модель РТК – $M(q)$ , $B$ , начальная и целевая конфигурации робота – $q_{s1} = q_i^s, q_{sg} = q_j^g$ , вектор скорости $\dot{q}_{greed}$ .	
1:	Установить начальное значение параметра дискретизации $N \leftarrow N_0$ ;
2:	<b>повторять</b>
3:	Вычислить $V_a$ для параметра дискретизации $N$ ;
4:	Установить весовые коэффициенты для ребер решетки $T_{ab_k} \leftarrow 1/3n$ ( $k = 1:d$ );
5:	<b>повторять</b>
6:	Вычислить весовую функцию $\phi_a$ ( $a=1:N^n$ );
7:	$a \leftarrow s1$ ;
8:	<b>повторять</b>
9:	$\phi \leftarrow \maxf(\phi_{b_k})$ ;
10:	$b \leftarrow \maxb(\phi_{b_k})$ ;
11:	$p \leftarrow \text{explore}(q(\phi_a), q(\phi_b), \dot{q}_{greed})$ ;
12:	$P \leftarrow \{q(\phi_a), q(\phi_b)\}$ ;
13:	если $b = sg$ то вернуть фрагмент траектории $P$ ;
14:	$a \leftarrow b$ ;
15:	<b>до тех пор пока</b> $p=0$ ;
16:	$T_{ab_k} \leftarrow 0$ ;
17:	$P \leftarrow 0$ ;
18:	<b>до тех пор пока</b> $\phi_{s1} = 0$ ;
19:	$N \leftarrow N + N_s$ ;
20:	<b>до тех пор пока</b> $N \leq N_{max}$ .

Алгоритм использует следующие обозначения:  $N_0$  – начальное значение параметра дискретизации конфигурационного пространства;  $N_{max}$  – максимально допустимое значение параметра дискретизации конфигурационного пространства;  $N_s$  – шаг изменения параметра дискретизации;  $P$  – криволинейная траектория робота. Алгоритм использует следующие функции:  $\maxf(\phi_{b_k})$  – функция, возвращающая максимальное значение весовой  $\phi$  из множества  $\phi_{b_k}$ ;  $\maxb(\phi_{b_k})$  – функция, возвращающая индекс  $b$  мак-

симального значения весовой функции из множества  $\phi_{b_k}$ ;  $q(\phi)$  – функция, возвращающая конфигурацию робота, соответствующую значению весовой функции  $\phi$ ;  $explore(q(\phi_a), q(\phi_b), \dot{q}_{greed})$  – функция проверки существования прямолинейного участка траектории между двумя конфигурациями робота  $q(\phi_a)$  и  $q(\phi_b)$  в соответствии с формулой (6) и заданным вектором угловых скоростей  $\dot{q}_{greed}$ . Если такой участок траектории существует, функция возвращает 1, в противном случае она возвращает 0.

Данный алгоритм функционирует следующим образом. Первоначально конфигурационное пространство робота дискретизируется с низким разрешением  $N_0$  и предполагается, что между всеми соседними конфигурациями существуют прямолинейные участки траектории, т.е. всем весовым коэффициентам ребер решетки присваивается значение  $1/3n$ . Далее вычисляется весовая функция для каждого узла решетки  $\phi_a$  ( $a = 1:N_n$ ) и ищется траектория между стартовой  $q_{s1}$  и целевой  $q_{sg}$  конфигурациями путем подъема в направлении градиента весовой функции узлов решетки. Для этого отрезки, соединяющие соседние конфигурации  $q_a$  и  $q_{b_k}$ , дискретизируются с высоким разрешением  $N_h$  в соответствии с (6), и для каждой дискретной конфигурации робота  $d_k$  выполняется тест столкновения. Если столкновения отсутствуют, то выбираются две новые конфигурации на решетке низкого разрешения в направлении градиента. Если обнаружено столкновение или невозможно движение робота от конфигурации  $q_a$  и  $q_{b_k}$  с заданной скоростью  $\dot{q}_{greed}$ , то значение весового коэффициента ребра решетки  $T_{ab_k}$  устанавливается в 0.

Значения весовой функции  $\phi_a$  для каждого узла решетки вычисляются на основе следующей итерационной формулы:

$$\phi_a^{(l+1)} = f_a \left( \sum_{k=1}^d T_{ab_k} \phi_{b_k}^{(l)} + V_a \right), \quad (8)$$

где  $l$  – номер итерации;

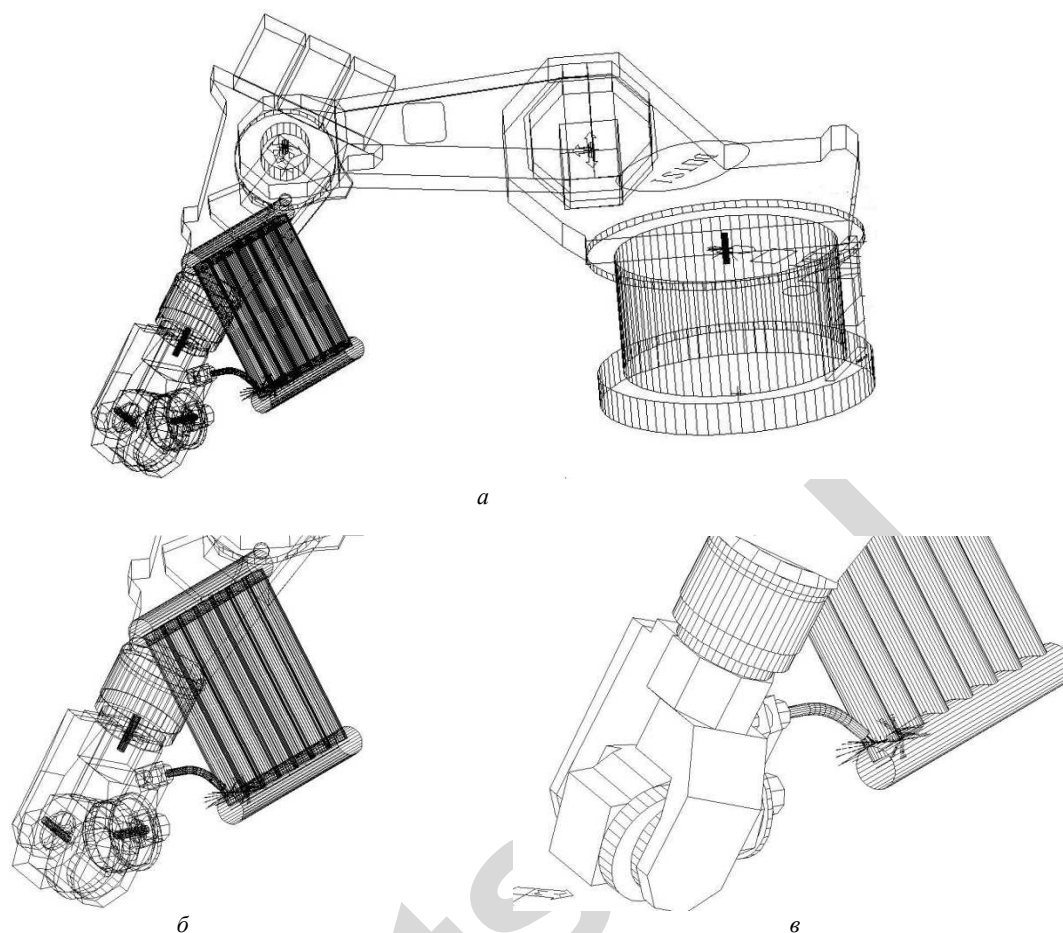
$V_a$  – параметр, значение которого определяется следующим образом:  $V_a = -1$ , если  $q_a \notin C_f$ ;  $V_a = 1$ , если  $q_a = q_{sg}$ ;  $V_a = 0$  во всех остальных случаях.

Нелинейная функция  $f_a$  имеет следующий вид:

$$f_a(x) = \begin{cases} 0, & \text{если } x < 0, \\ \tanh(x), & \text{если } x \geq 0. \end{cases} \quad (9)$$

Величина весовых коэффициентов ребер решетки  $T_{ab_k}$  выбирается, исходя из достаточных условий сходимости итерационной процедуры (8), полученных в работах [16, 17]. В частности,  $T_{ab_k} = 1/3n$ , если существует прямолинейный участок траектории между конфигурациями  $q_a$  и  $q_{b_k}$ , и  $T_{ab_k} = 0$  в противном случае.

**Тестирование алгоритмов.** Исследование эффективности проводилось в экспериментальной среде моделирования РТК с применением эмулятора систем управления типа *RCM (Robot Control Multiprocessor)*. Предложенные алгоритмы управления реализованы на языке программирования C++ и интегрированы в систему моделирования РТК. В качестве объекта исследования использовалась роботизированная ячейка для дуговой сварки металлической конструкции, состоящей из 9 труб. Ячейка включает робот-манипулятор *KR125* со сварочной горелкой (рис. 3, а). На трехмерную модель металлоконструкции нанесено 14 сварных швов, имеющих форму эллипса, изогнутого в пространстве. Предложенные алгоритмы позволили обеспечить управление движением сварочной горелки вдоль каждого из 14 швов со скоростью, обеспечивающей минимизацию критерия качества (7). Заданные траектории реализованы на основе технологического языка программирования роботов *SRCL (Siemens Robot Control Language)* и отлажены с использованием эмулятора *RCM*. Тестирование предложенного алгоритма управления показало, что он обеспечил свободное от столкновений движение робота-манипулятора. Примеры позиционирования сварочной горелки на линии пространственного сварного шва при движении робота под управлением предложенного алгоритма приведены на рисунке 3, б, в. Анализ результатов данных экспериментов позволяет сделать вывод о том, что предлагаемый подход эффективен при управлении сварочными роботами-манипуляторами в рабочей среде с препятствиями.



**Рисунок 3. – Роботизированная ячейка для дуговой сварки и траектории движения сварочной горелки под управлением предложенного алгоритма**

**Заключение.** В данной работе предложены новые эффективные алгоритмы управления сварочными роботами-манипуляторами на основе статистической модели. Предложенные алгоритмы используют решетчатую дискретизацию «насыщенных» препятствиями зон конфигурационного пространства, что позволяет эффективно учесть сложную нерегулярную форму препятствий в РТК. Кроме того, предложенные алгоритмы, в отличие от известных, позволяют учесть ограничения на ориентацию технологического инструмента, а также ограничения, накладываемые на угловые скорости движения звеньев робота-манипулятора. Эффективность алгоритмов управления подтверждается результатами тестирования в экспериментальной среде моделирования РТК.

#### ЛИТЕРАТУРА

1. Климов, А. С. Роботизированные технологические комплексы и технологические линии в сварке : учебное пособие / А. С. Климов, Н. Е. Машнин. – 2-е изд., испр. и доп. – СПб. : Лань, 2011. – 240 с.
2. Зенкевич, С. Л. Управление роботами. Основы управления манипуляционными роботами : учеб. для вузов / С. Л. Зенкевич, А. С. Ющенко. – М. : Изд-во МГТУ им. Н. Э. Баумана, 2000. – 400 с.
3. Principles of Robot Motion: Theory, Algorithms and Implementations / Н. Choset [et al.]. – Boston : MIT Press, 2005. – 680 p.
4. LaValle, S. M. Planning Algorithms / S. M. LaValle. – Cambridge : Cambridge University Press, 2006. – 1023 p.
5. Lahijanian, M. A. Sampling-Based Strategy Planner for Nondeterministic Hybrid Systems / M. A. Lahijanian, L. E. Kavraki, M. Y. Vardi // International Conference on Robotics and Automation. – Hong Kong, China, 2014. – P. 3005–3012.
6. Iterative Temporal Motion Planning for Hybrid Systems in Partially Unknown Environments / M. R. Maly [et al.] // ACM International Conference on Hybrid Systems: Computation and Control (HSCC). – Philadelphia, PA, USA, ACM, 2013. – P. 353–362.

7. Sucas, I. A. Accounting for Uncertainty in Simultaneous Task and Motion Planning Using Task Motion Multi-graphs / I. A. Sucas, L. E. Kavraki // IEEE International Conference on Robotics and Automation. – St. Paul, 2012. – P. 4822–4828.
8. Kavraki, L. E. Motion planning / L. E. Kavraki, S. M. LaValle // Springer Handbook of Robotics ; ed.: B. Siciliano and O. Khatib. – Berlin : Springer-Verlag, 2008. – 1628 p.
9. Geraerts, R. J. A comparative study of probabilistic roadmap planners / R. J. Geraerts, M. H. Overmars // Algorithmic Foundations of Robotics V. – Berlin : Springer-Verlag, 2003. – P. 43–58.
10. Geraerts, G. J. Reachability-based Analysis for Probabilistic Roadmap Planners / G. J. Geraerts, M. H. Overmars // Journal of Robotics and Autonomous Systems. – 2007. – № 55. – P. 824–836.
11. Geraerts, R. J. Sampling and Node Adding in Probabilistic Roadmap Planners / R. J. Geraerts, M. H. Overmars // Journal of Robotics and Autonomous Systems. – 2006. – № 54. – P. 165–173.
12. LaValle, S. M. On the relationship between classical grid search and probabilistic roadmaps / S. M. LaValle, M. Branicky, S. R. Lindemann // International Journal of Robotic Research. – 2004. – № 23(7/8). – P. 673–692.
13. Lewis, F. L. Robot Manipulator Control: Theory and Practice / F. L. Lewis, D. M. Dawson, Ch. T. Abdallah. – N.Y. : Merce Deccer, Inc., 2004. – 607 p.
14. Robotic Systems – Applications, Control and Programming / A. Dutta [et al.]. – 2012. – 628 p.
15. Юревич, Е. И. Основы робототехники / Е. И. Юревич. – 2-е изд., перераб. и доп. – СПб. : БВХ-Петербург, 2005. – 416 с.
16. Комбинированный метод синтеза траекторий сборочно-сварочных роботов-манипуляторов в рабочей среде с препятствиями / М. М. Кожевников [и др.] // Доклады БГУИР. – 2016. – № 1 (95). – С. 12–18.
17. Планирование траекторий роботов-манипуляторов в технологическом процессе точечной контактной сварки / М. М. Кожевников [и др.] // Вестн. ГГТУ им. П. О. Сухого. – 2016. – № 2 (65). – С. 3–10.

Поступила 05.08.2016

## ALGORITHMS FOR CONTROL OF WELDING ROBOTIC-MANIPULATORS BASED ON A STATISTICAL MODEL OF A CONFIGURATION SPACE

*I. ILIUSHYN, M. KOZHEVNIKOV*

*The new effective algorithms were proposed for control of welding robotic-manipulators. The proposed algorithms are based on a statistical model of a configuration space. In contrast to known they effectively take into account complex shape of the elements of the robotic technological cell, the limits for the orientation of technological tool, as well as limits for the angular velocity of the manipulator links. The effectiveness of the proposed algorithms is confirmed by the tests in the experimental space of robot modeling.*

**Keywords:** *robotic-manipulators, algorithms for control, configuration space.*