

Метод приближенного построения границы области достижимости многосекционного манипулятора типа «хобот»

01, январь 2011

авторы: Карпенко А. П., Семенихин А. С., Черная Л. А.

УДК 519.6

МГТУ им. Н.Э. Баумана

Введение

Построение области достижимости манипулятора (ОДМ) является одной из базовых задач робототехники. Знание этой области позволяет определить такие важные параметры робота-манипулятора, как объем области достижимости, показатель Ли-Янга, пределы достижимости схвата и т.д. [1].

Аналитически задача построения ОДМ может быть решена только в простейших случаях. Для многосекционного манипулятора типа «хобот» [2-4] возможны лишь приближенные численные методы решения этой задачи.

Простейшим из приближенных методов построения ОДМ является метод, идея которого состоит в следующем: покрываем область допустимых значений вектора обобщенных координат манипулятора некоторой сеткой; во всех узлах этой сетки решаем прямую позиционную задачу робототехники; полученное дискретное множество возможных положений схвата манипулятора принимаем за дискретную аппроксимацию ОДМ. При высокой размерности вектора обобщенных координат и высокой требуемой точности построения границы ОДМ, такой метод может требовать неприемлемо высоких вычислительных затрат.

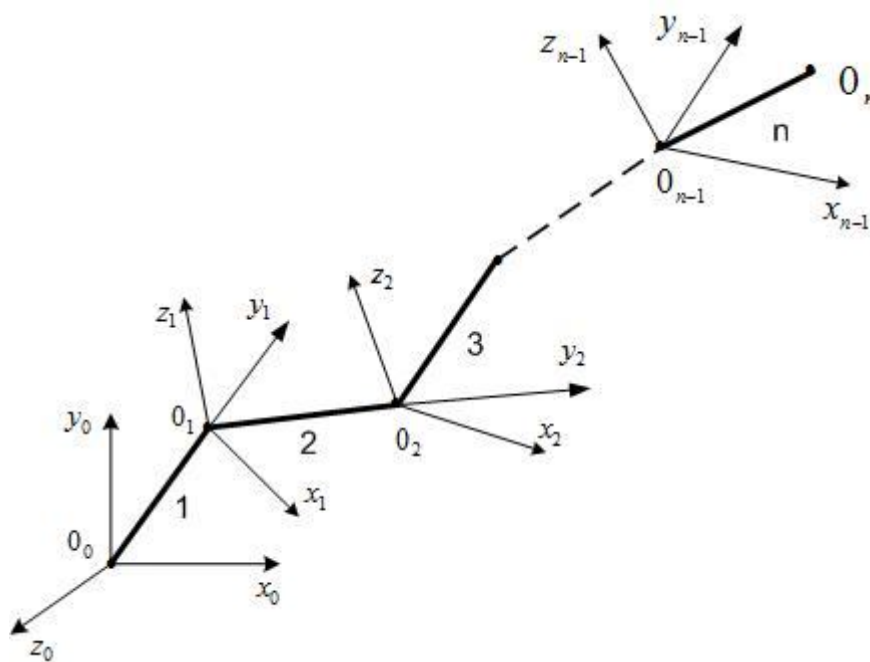
В работе для построения границы ОДМ предлагается *SPSO*-метод, источником идеи которого является метод *MOPSO* приближенного построения

фронта Парето (а тем самым, и множества Парето) в задаче многокритериальной оптимизации [5-7].

В первом разделе работы приведена постановка задачи. Во втором разделе рассмотрены метод *MOPSO* и σ -алгоритм. Третий раздел содержит описание метода *SPSO*, четвертый раздел – краткое описание программной системы, реализующей данный метод, пятый раздел – некоторые экспериментальные результаты исследования его эффективности. В заключении сформулированы основные результаты работы и перспективы ее развития.

1. Постановка задачи

Рассмотрим многосекционный манипулятор типа «хобот», схема которого представлена на рисунке 1.



1, 2, ..., n – номера секций; 0_n – схват манипулятора

Рисунок 1 - Схема манипулятора

Свяжем с i -ой секцией манипулятора систему координат $0_i x_i y_i z_i$, в которой ось $0_i y_i$ совпадает с продольной осью симметрии этой секции, а оси $0_i x_i$, $0_i z_i$ расположены таким образом, что образуют с осью $0_i y_i$ правую тройку; $i \in [1:n]$. Инерциальную систему координат, связанную с неподвижным основанием манипулятора, обозначим $0_0 x_0 y_0 z_0$. Там, где это не может вызвать неверного чтения, будем для простоты письма для этой системы координат использовать также обозначение $0xyz$.

Пусть L_i - длина i -ой секции манипулятора, а также сама эта секция; $i \in [1:n]$. Положим, что каждая кинематическая пара (L_{i-1}, L_i) имеет две степени свободы и ориентацию i -ой секции относительно $(i-1)$ -ой секции определяют углы Эйлера $\varphi_{i,1}$, $\varphi_{i,2}$ (рисунок 2).

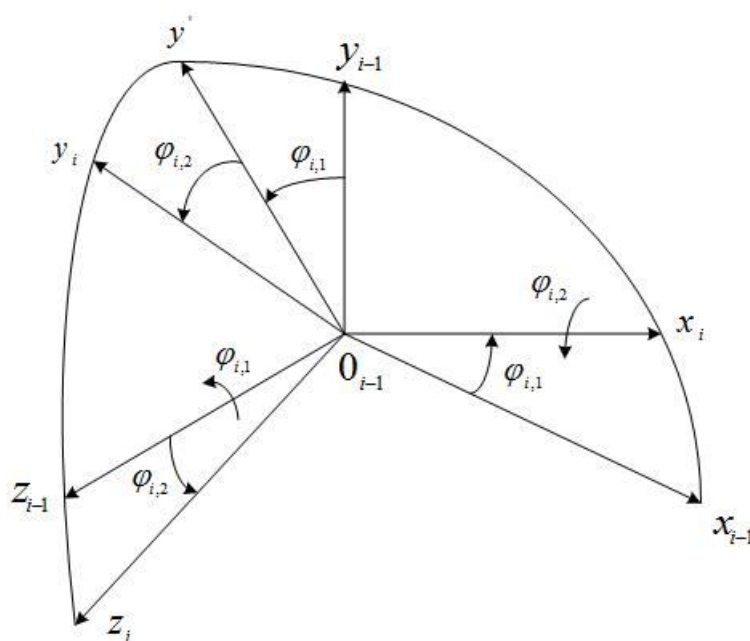


Рисунок 2 – Системы координат $0_{i-1} x_{i-1} y_{i-1} z_{i-1}$, $0_i x_i y_i z_i$

Геометрические соотношения между системами координат $0_{i-1} x_{i-1} y_{i-1} z_{i-1}$, $0_i x_i y_i z_i$ можно представить в виде (4×4) -матрицы однородных преобразований

$$V_i = \begin{bmatrix} \cos \varphi_{i,1} & -\sin \varphi_{i,1} & 0 & 0 \\ \sin \varphi_{i,1} \cos \varphi_{i,2} & -\cos \varphi_{i,1} \cos \varphi_{i,2} & \sin \varphi_{i,2} & L_i \\ \sin \varphi_{i,1} \sin \varphi_{i,2} & -\cos \varphi_{i,1} \sin \varphi_{i,2} & \cos \varphi_{i,2} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

При этом координаты точки 0_i , $i \in [1:n]$ в системе координат $0_0 x_0 y_0 z_0$ определяются формулой

$$\begin{pmatrix} x_{0,i} & y_{0,i} & z_{0,i} & 1 \end{pmatrix}^T = V_{0,i} \begin{pmatrix} 0 & 0 & 0 & 1 \end{pmatrix}^T,$$

где (4×4) -матрица $V_{0,i} = V_1 V_2 \dots V_i [1]$.

Положим, что секция манипулятора с номером $i \in [1:n]$ представляет собой прямой круговой усеченный конус высотой L_i , диаметр большого основания которого равен d_{i-1} , а меньшего основания – d_i (рисунок 3).

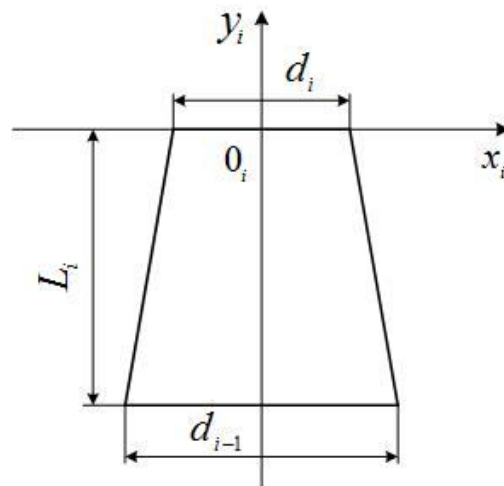


Рисунок 3 – Форма секций манипулятора

Введем в рассмотрение $(2n \times 1)$ -вектор обобщенных координат манипулятора $q = (\varphi_{1,1}, \varphi_{1,2}, \varphi_{2,1}, \dots, \varphi_{n,2})^T$ и множество допустимых значений его компонентов

$$D_q = \{\varphi_{i,j} \mid -\varphi_{i,j}^- \leq \varphi_{i,j} \leq \varphi_{i,j}^+, i \in [1:n], j = 1, 2\}, \quad (1)$$

где $\varphi_{i,j}^-$, $\varphi_{i,j}^+$ - заданные константы.

Областью достижимости манипулятора D называется множество всех возможных положений его схвата $(x_{0,n}, y_{0,n}, z_{0,n})$, удовлетворяющих ограничению

$$q \in D_q. \quad (2)$$

Границу области D обозначим Γ .

Ставится *задача 1 (без препятствий)* – задача приближенного построения границы Γ области D .

В области D может находиться некоторое число неподвижных препятствий $B_1, B_2, \dots$. При этом ОДМ некоторым образом трансформируется. Обозначим $\tilde{D}$ трансформированную препятствиями ОДМ и определим следующим образом:

- область $\tilde{D}$ удовлетворяет ограничениям (1), (2);
- каждая из секций манипулятора отстоит от каждого из препятствий B_j , $j = 1, 2, \dots$ не менее чем на заданную величину $\varepsilon_B > 0$.

Границу области $\tilde{D}$ обозначим $\tilde{\Gamma}$.

Ставится *задача 2 (задача с препятствиями)* – задача приближенного построения границы $\tilde{\Gamma}$ области $\tilde{D}$.

Отметим, что отображения $D_q \rightarrow D$, $D_q \rightarrow \tilde{D}$ являются, вообще говоря, однозначными, но не взаимно-однозначными.

2. Оптимизация роем частиц

2.1. Рассмотрим сначала задачу глобальной безусловной *однокритеральной оптимизации*

$$\min_X f(X) = f(X^*), \quad X \in R^m. \quad (3)$$

где $f(X)$ – целевая функция, определенная в m -мерном арифметическом пространстве R^m (пространстве параметров).

Множество частиц (рой, популяцию) обозначим $\{P_i, i \in [1:N]\}$, где N – число частиц в рое (размер популяции). Частица P_i определяется m -мерными векторами координат $X_{i,t}$ и скоростей $V_{i,t}$. Здесь $t \in [0:T]$ – дискретный момент времени (номер итерации); T – общее число итераций. Начальные координаты и скорости частицы P_i известны и равны $X_{i,0} = X_i^0$, $V_{i,0} = V_i^0$ соответственно.

В каноническом алгоритме роя частиц (PSO) итерации выполняются по схеме

$$X_{i,t+1} = X_{i,t} + V_{i,t+1},$$

$$V_{i,t+1} = \phi_1 V_{i,t} + U[0, \phi_2] \otimes (X_{i,t}^b - X_{i,t}) + U[0, \phi_3] \otimes (X_{i,t}^g - X_{i,t}).$$

Здесь $U[a,b]$ – m -мерный вектор псевдослучайных чисел, равномерно распределенных в интервале $[a,b]$; $\otimes$ – символ покомпонентного умножения векторов; $X_{i,t}^b$ – вектор координат частицы P_i с наилучшим (в смысле формулы (3)) значением целевой функции $f(X)$ за все время поиска $[0:t]$; $X_{i,t}^g$ – вектор координат частицы, которая является соседней с данной частицей P_i и имеет наилучшее за то же время значение целевой функции $f(X)$; ϕ_1, ϕ_2, ϕ_3 – свободные параметры алгоритма. Соседство частиц определяется используемой топологией соседства. Позиция $X_{i,t}^b$ частицы P_i называется локально лучшей позицией этой частицы, а $X_{i,t}^g$ – глобально лучшей позицией соответствующей соседней частицы [8].

2.2. Перейдем к рассмотрению задачи *многокритериальной оптимизации* (МКО-задачи). Положим, что множеством допустимых значений вектора варьируемых параметров X является ограниченное и замкнутое множество $D_X \subset R^m$. На множестве D_X определен векторный критерий оптимальности $F(X) = (f_1(X), f_2(X), \dots, f_k(X))$ со значениями в пространстве R^k (пространстве критериев). Обозначим D_F – множество достижимости задачи [9].

Рассмотрим две следующие МКО-задачи.

1) *МКО-задача 1.* Лицо, принимающее решения (ЛПР), стремится минимизировать каждый из частных критериев оптимальности $f_1(X), f_2(X), \dots, f_k(X)$, что условно записывается в виде

$$\min_X F(X) = F(X^I), \quad X \in D_X, \quad (4)$$

где X^I - решение задачи.

2) *МКО-задача 2.* ЛПР стремится максимизировать каждый из частных критериев оптимальности:

$$\max_X F(X) = F(X^A), \quad X \in D_X, \quad (5)$$

где X^A - решение.

Определим для МКО-задач 1, 2 отношения доминирования.

В МКО-задаче 1 вектор $F(X_1) = F_1$ доминирует вектор $F(X_2) = F_2$ ($F_1 \succ^I F_2$), если среди равенств и неравенств $f_i(X_1) \leq f_i(X_2)$, $i \in [1:k]$ имеется, хотя бы одно строгое неравенство. Аналогично, в МКО-задаче 2 вектор F_1 доминирует вектор F_2 ($F_1 \succ^A F_2$), если среди равенств и неравенств $f_i(X_1) \geq f_i(X_2)$, $i \in [1:k]$ имеется, хотя бы одно строгое неравенство [9].

Подмножества множества D_F , среди которых нет векторов F , доминируемых в смысле отношений $\succ^I$, $\succ^A$, представляют собой фронты Парето МКО-задач 1, 2 и обозначаются D_F^I , D_F^A соответственно. Множества $D_X^I \in D_X$, $D_X^A \in D_X$, соответствующие множествам D_F^I , D_F^A , представляют собой ни что иное, как множества Парето МКО-задач 1, 2 [9].

По аналогии с тем, что вектор $X_{i,t}$ определяет положение частицы P_i в пространстве параметров, будет говорить, что вектор $F_{i,t} = (f_{1,i,t}, f_{2,i,t}, \dots, f_{k,i,t}) = F(X_{i,t})$ определяет положение этой частицы в пространстве критериев.

Хорошо известно, что при отсутствии дополнительной априорной информации (по сравнению с информацией, присутствующей в постановке задач (4), (5)), во множествах Парето D_F^I , D_F^A невозможно выделить в качестве

решения этих задач одну из точек. Поэтому далее под решениями задач (4), (5) понимаются множества Парето D_X^I , D_X^A этих задач [5].

При использовании алгоритма *PSO* для решения МКО-задачи основной проблемой является то, какую позицию считать глобально лучшей позицией $X_{i,t}^g$ для частицы P_i . Существует несколько алгоритмов решения этой проблемы, одним из которых является σ -алгоритм [5, 6]. Рассмотрим схему этого алгоритма на примере двухкритериальной задачи ($k = 2$).

В случае двухкритериальной задачи σ -параметром частицы P_i , $i \in [1:N]$ с координатами X_i называется величина

$$\sigma_i = \sigma(F_i) = \frac{f_{1,i}^2 - f_{2,i}^2}{f_{1,i}^2 + f_{2,i}^2}, \quad (6)$$

где $F_i = (f_{1,i}, f_{2,i}) = F(X_i)$. Легко видеть, что векторы F_i , F_j , имеющие одно и то же значение σ -параметра, лежат на одной прямой, проходящей через начало координат системы координат $0f_1f_2$. Формулу (6) иллюстрирует рисунок 4, на котором одинаковые значения σ -параметра имеют пары векторов (F_1, F_2) , (F_3, F_4) , (F_5, F_6) .

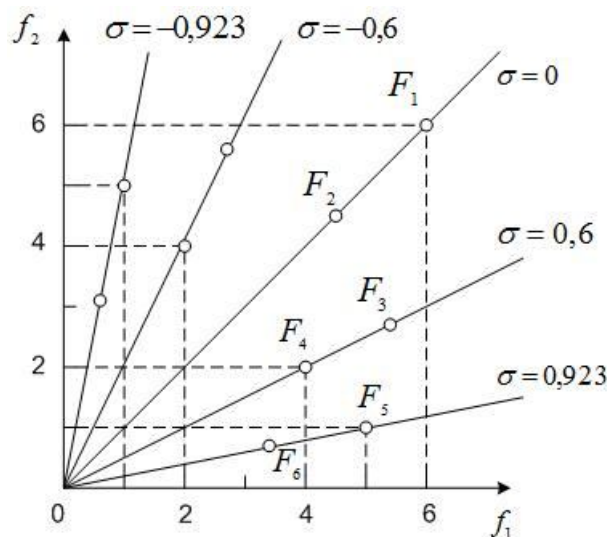


Рисунок 4 – К определению σ -параметра: $k = 2$

В случае, когда размерность пространства критериев больше двух ($k > 2$) σ -алгоритм использует k -мерный вектор σ -параметров $\sigma = (\sigma_1, \sigma_2, \dots, \sigma_k)$. Так для частицы P_i , $i \in [1:N]$ на итерации $t \in [1:T]$ этот вектор равен

$$\sigma_{i,t} = \sigma(F_{i,t}) = \frac{\left((f_{1,i,t}^2 - f_{2,i,t}^2) \quad (f_{2,i,t}^2 - f_{3,i,t}^2) \quad \dots \quad (f_{k,i,t}^2 - f_{1,i,t}^2) \right)^T}{f_{1,i,t}^2 + f_{2,i,t}^2 + \dots + f_{k,i,t}^2}.$$

Сигма-алгоритм существенно использует архив недоминируемых решений. Обозначим этот архив S^I для МКО-задачи 1 и S^A - для МКО-задачи 2.

Если на итерации $t \in [1:T]$ алгоритма *PSO* в архиве S^q находятся компоненты векторов $X_{i,t}^q$, $F_{i,t}^q$, соответствующих частице P_i^q , то для краткости письма будем далее говорить, что в архиве S^q на этой итерации находится частица P_i^q ; $q \in \{I; A\}$. Здесь $i \in [1:N_t^q]$, где N_t^q - число частиц в соответствующем архиве в момент времени t . Сигма-параметр частицы P_i^q в момент времени t обозначим $\sigma_{i,t}^q$.

В σ -алгоритме для частицы P_j на итерации t глобально лучшая позиция $X_{i,t}^g$ в архиве S^q , $q \in \{I; A\}$ определяется по следующему правилу.

1) Определяем σ -параметр частицы P_j , т.е. вычисляем величину $\sigma_{j,t} = \sigma(F_{j,t})$.

2) Находим в архиве S^q частицу P_l^q , у которой σ -параметр наиболее близок к величине $\sigma_{j,t}$:

$$\min_i \|\sigma_{j,t} - \sigma_{i,t}^q\| = \|\sigma_{j,t} - \sigma_{l,t}^q\|; i \in [1:N_t^q].$$

Здесь $\|\cdot\|$ - символ евклидовой нормы.

3) В качестве глобально лучшей позиции $X_{i,t}^g$ для частицы P_j принимаем вектор $X_{l,t}^q$, т.е. вектор координат частицы P_l^q .

Для двухкритериальной МКО-задачи 1, изложенную схему σ -алгоритма иллюстрирует рисунок 5, где темным кружками показаны векторы F_i^I ,

соответствующие частицам P'_i , стрелками показаны глобально лучшие позиции для соответствующих частиц; $i \in [1 : N'_t]$.

3. SPSO-метод приближенного построения границы области достижимости

Ограничимся случаем симметричных ограничений на углы $\varphi_{i,j}$, $i \in [1 : n]$, $j = 1, 2$ - положим, что

$$D_q = \{\varphi_{i,j} \mid -\varphi_{i,j}^{\max} \leq \varphi_{i,j} \leq \varphi_{i,j}^{\max}, i \in [1 : n], j = 1, 2\},$$

где $\varphi_{i,j}^{\max}$ - заданные константы. Легко видеть, что в этих допущениях ОДМ в задаче без препятствий симметрична относительно оси Oy . Поэтому для построения ОДМ в данном случае достаточно построить лишь одно ее сечение (пусть это будет сечение $z = 0$) и, более того, - лишь половину этого сечения (пусть это будет часть области D , соответствующая $x \geq 0$).

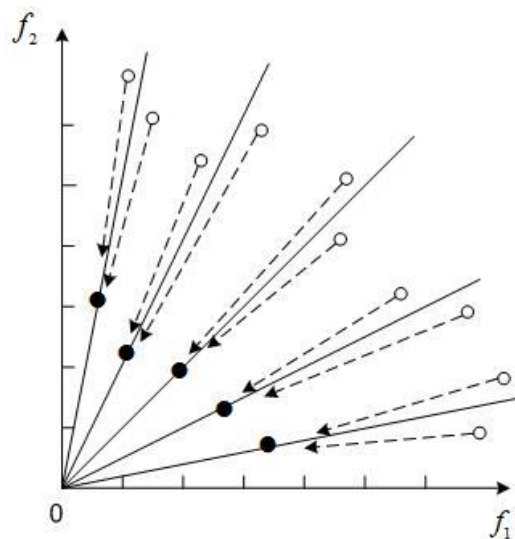


Рисунок 5 – К схеме σ -алгоритма: МКО-задача 1; $k = 2$

В задаче с препятствиями ОДМ, построенная в классе плоских допустимых конфигураций манипулятора, вообще говоря, не совпадает с ОДМ, полученной в классе его произвольных пространственных конфигураций.

В работе, однако, и в задаче с ограничениями рассматриваются только «плоские» конфигурации манипулятора.

SPSO-метод построения границы *ОДМ* основан на следующих аналогиях этой задачи с задачей многокритериальной оптимизации:

- пространству параметров $\{X\}$ соответствует пространство обобщенных координат манипулятора $\{q\}$;
- множеству допустимых значений вектора варьируемых параметров D_x - множество допустимых значений вектора обобщенных координат D_q ;
- пространству критериев $\{F\}$ - пространство геометрических координат схвата манипулятора $\{x, y, 0\}$;
- множеству достижимости D_F - область достижимости манипулятора D ;
- границе Γ_F множества достижимости D_F - граница Γ область достижимости манипулятора D .

Сигма-параметр некоторой точки $X = (x, y, 0)$ определим как

$$\sigma = \sigma(X) = \frac{\arctg\left(\frac{y}{x}\right)}{\pi/2} \in [-1, 1]. \quad (7)$$

Очевидно, что все точки, лежащие на прямой $y = \sigma x$, имеют одно и то же значение σ -параметра, равное σ .

В методе *SPSO* граница Γ условно разбивается на два фрагмента (не обязательно односвязных), которые называются ближней Γ^l и дальней Γ^d границами соответственно.

Рассмотрим точки $X_1 = (x_1, y_1, 0)$, $X_2 = (x_2, y_2, 0)$, принадлежащие области достижимости манипулятора D . При построении ближней границы Γ^l аналогом отношения доминирования $\succ^l$ в задаче многокритериальной оптимизации является отношение $\triangleright^l$, определяемое следующим образом: точка X_1 доминирует точку X_2 ($X_1 \triangleright^l X_2$), если среди равенств и неравенств

$$|x_1| \leq |x_2|, \quad |y_1| \leq |y_2| \quad (8)$$

имеется хотя бы одно строгое. Аналогично, при построении дальней границы Γ^A точка X_1 доминирует точку X_2 ($X_1 \succ^A X_2$), если среди равенств и неравенств

$$|x_1| \geq |x_2|, \quad |y_1| \geq |y_2| \quad (9)$$

имеется хотя бы одно строгое.

Замечание. Легко видеть, что в принятых в начале данного раздела соглашениях, величины x_1, x_2 всегда неотрицательны. Поэтому для этих величин знак модуля в формулах (8), (9) можно было бы опустить. Однако ниже эти формулы будут использоваться в ситуации, когда величины x_1, x_2 могут принимать и отрицательные значения •

Основная процедура *SPSO*-метода использует разбиение полуплоскости $x \geq 0$ на M равных секторов C_i с углом раскрыва $\varepsilon_c = \frac{180^\circ}{M}$; $i \in [1:M]$. Значения σ -параметров верхних границ этих секторов обозначим σ^i , $i \in [1:(M-1)]$ (рисунок 6). Нижнюю σ -границу сектора C_1 , совпадающую с отрицательным направлением оси Oy обозначим $\sigma^0 = -1$, а верхнюю σ -границу сектора C_M , совпадающую с осью Ox обозначим $\sigma^M = 1$.

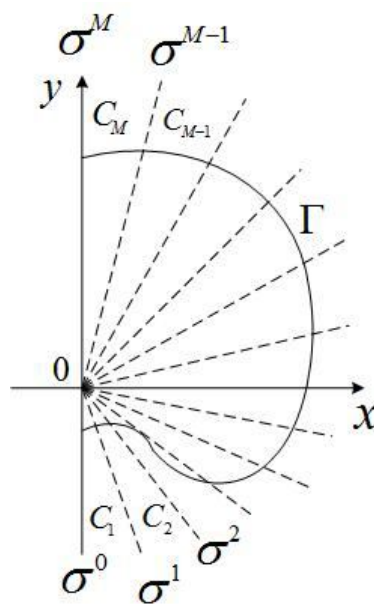


Рисунок 6 – К схеме базовой процедуры метода *SPSO*

Схема основной процедуры метода *SPSO* при построении аппроксимации границы Γ^q , $q \in \{I; A\}$ имеет следующий вид.

1) Создаем M начальных популяций частиц таким образом, чтобы в каждом из секторов C_i оказалось N частиц $P_{i,j}$; $i \in [1:M]$, $j \in [1:N]$.

2) В соответствии с σ -алгоритмом, независимо в каждом из секторов строим архив недоминируемых частиц S_i^q , содержащий дискретную аппроксимацию i -го фрагмента Γ_i^q соответствующей границы ОДМ.

3) В качестве искомой аппроксимации границы Γ^q используем объединение частиц из архивов S_i^q , т.е. полагаем

$$S^q = \bigcup_{i=1}^M S_i^q,$$

или уточняем решение (см. ниже).

Вариант метода *SPSO*, использующий рассмотренную процедуру назовем базовым методом *SPSO*.

Возможные основные варианты поведения границы ОДМ в пределах сектора иллюстрирует рисунок 7а, а производные варианты, представляющие собой все возможные комбинации двух основных вариантов, - рисунок 7б. Здесь $\alpha, \beta, \gamma, \delta$ - некоторые целые числа из интервала $[1:M]$. Возможные комбинации более чем двух указанных основных вариантов в методе *SPSO* не используются.

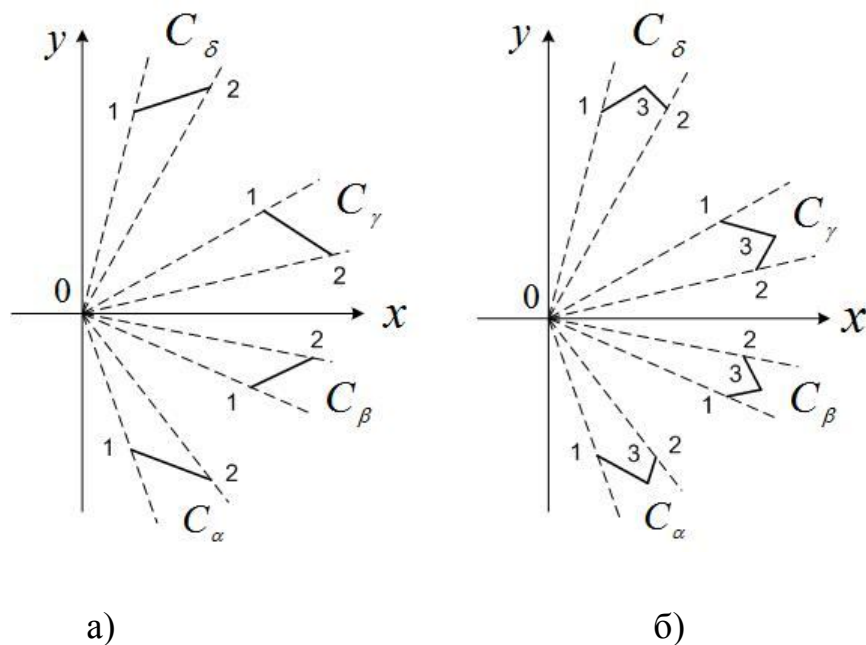


Рисунок 7 – Варианты поведения границы ОДМ в пределах одного сектора

Рассмотрим прежде ситуации, представленные на рисунке 7а. Формально, эти ситуации описываются следующим образом (граница ОДМ полагается всюду дифференцируемой): всюду в секторе C_α , C_γ производные функций, определяющих границы Γ_α , Γ_γ , отрицательны; всюду в секторах C_β , C_δ аналогичные производные не отрицательны.

Из определения отношения доминирования $\triangleright^q$ следует, что при построении с помощью базового метода *SPSO* границ Γ_β^q , Γ_γ^q частицы в архивах S_β^q , S_γ^q будут примерно равномерно концентрироваться вдоль этих границ; $q \in \{I; A\}$. Из свойств того же отношения следует, что при построении границ Γ_α^I , Γ_δ^I частицы в архивах S_α^I , S_δ^I будут сосредотачиваться вблизи точек “1”, а при построении границ Γ_α^A , Γ_δ^A частицы в архивах S_α^A , S_δ^A будут «стекаться» к точкам “2”.

В ситуациях, представленных на рисунке 7б, имеют место комбинации рассмотренных вариантов поведения частиц в архивах. Рассмотрим для примера сектор C_α . При построении с помощью базового метода *SPSO* границы Γ_α^q , $q \in \{I; A\}$ частицы в архиве S_α^q будут примерно равномерно

распределяться вдоль части “2-3” этой границы и концентрироваться вблизи точки “1” при построении границы Γ_α^I и со стороны точки “2” вблизи точки “3” – при построении границы Γ_α^A .

Введем в качестве меры плотности покрытия границы Γ_i^q величину

$$\rho_i^q = \frac{\Delta\sigma_i^q}{2/M}, \quad i \in [1:M], \quad q \in \{I; A\},$$

где $2/M$ - угол раскрытия сектора C_i в σ -пространстве, $\Delta\sigma_i^q$ - максимальное σ -расстояние между соседними частицами в архиве S_i^q , предварительно отсортированными по возрастанию их σ -параметра:

$$\Delta\sigma_i^q = \max_{j,l} |\sigma_{i,j}^q - \sigma_{i,l}^q|; \quad j, l \in [1:N].$$

С целью обеспечения равномерности покрытия границ вида Γ_α^q , Γ_δ^q , $q \in \{I; A\}$, представленных на рисунке 76, базовый метод *SPSO* модифицируется следующим образом (рисунок 8).

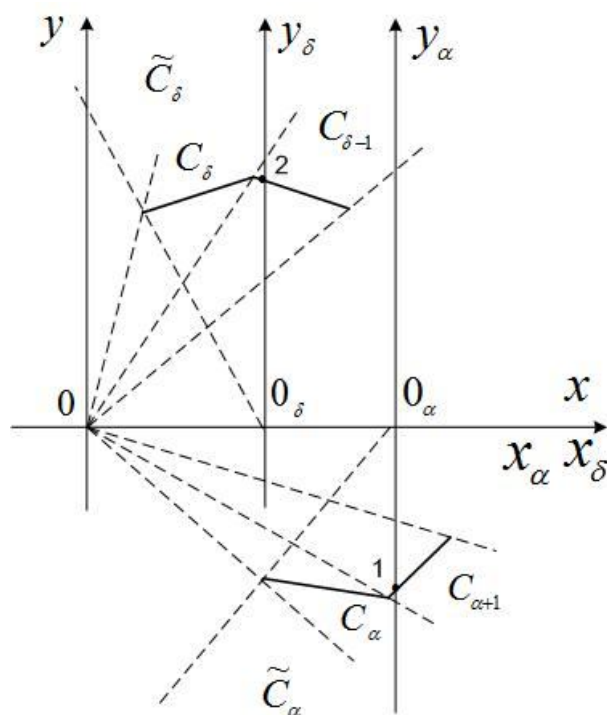


Рисунок 8 – К схеме модифицированного метода *SPSO*: 1 – первая частица в архиве $S_{\alpha+1}^q$; 2 – последняя частица в архиве $S_{\delta-1}^q$

- 1) Выполняем шаги 1, 2 базовой процедуры *SPSO*-метода.
- 2) Вычисляем значения σ -параметра $\sigma_{i,j}^q$ всех частиц в архивах S_i^q ; $i \in [1:M]$, $j \in [1:N]$.
- 3) В каждом из секторов C_i определяем меру плотности покрытия ρ_i^q и из числа этих секторов выделяем сектора C_{i_l} , в которых $\rho_{i_l}^q > \varepsilon_\sigma$. Здесь ε_σ - заданная константа; $i_l \in [1:M]$, $l \in [1:\tilde{M}]$, где $\tilde{M}$ - число таких секторов.
- 5) Для каждого из указанных секторов C_{i_l} строим новую систему координат $0_{i_l} x_{i_l}, y_{i_l}$ и в этой системе координат - новый сектор $\tilde{C}_{i_l}$. Здесь $0_{i_l} = (\tilde{x}_{i_l}, 0)$, где координата $\tilde{x}_{i_l}$ определяется по правилу

$$\tilde{x}_{i_l} = \begin{cases} x_{(i_l+1),1}, & y_{i_l,N} \leq 0, \\ x_{(i_l-1),N}, & y_{i_l,1} > 0. \end{cases} \quad (10)$$

В формуле (10) приняты следующие обозначения: $x_{(i_l+1),1}$ - абсцисса первой частицы в архиве $S_{i_l+1}^q$, который соответствует сектору C_{i_l+1} , следующему за данным сектором C_{i_l} ; аналогично, $x_{(i_l-1),N}$ - абсцисса последней частицы в архиве $S_{i_l-1}^q$; $y_{i_l,1}$, $y_{i_l,N}$ - ординаты первой и последней частиц в архиве $S_{i_l}^q$, соответствующем данному сектору.

- 6) В каждом из секторов C_{i_j} по схеме базового метода *SPSO* строим аппроксимацию границы $\Gamma_{i_j}^q$, $q \in \{I; A\}$.

- 7) В качестве искомой аппроксимации границы Γ^q используем объединение частиц из все полученных архивов частиц.

В случае, когда граница ОДМ ведет себя в секторах C_α , C_δ подобно тому, как это изображено на рисунке 7б, в качестве аналогов рассмотренных секторов C_α , C_δ используются сектора, соответствующие частям “1–3” секторов C_α , C_δ частям “2–3” секторов C_β , C_γ соответственно.

Отметим также, что в случае необходимости получения тонкой структуры границы ОДМ, модифицированный метод *SPSO* в сектора типа C_α , C_δ (рисунок 7) может использоваться итерационно несколько раз.

4. Программная реализация метода *SPSO*

Программная система *SPSO++*, реализующая метод *SPSO*, написана на языке программирования. *C++*, основными преимуществами которого по сравнению с другими языками является кроссплатформенность и эффективность. Для реализации оконного интерфейса (*GUI – Graphical user interface*), использована встроенная *C++* библиотека *Microsoft Foundation Classes (MFC)*. Визуализация ОДМ реализована с помощью утилиты *Gnuplot*, которая является кроссплатформенной, свободно распространяемой, простой и удобной в использовании.

Основные классы программной системы *SPSO++* представлены на рисунке 9.

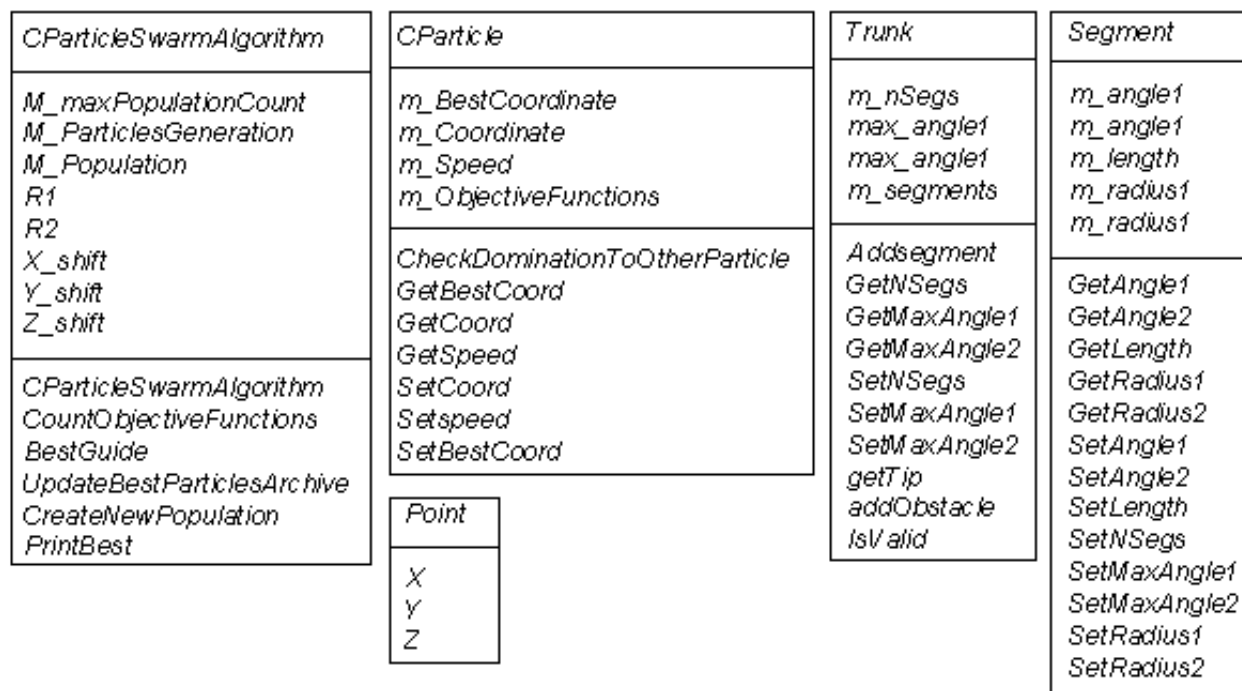


Рисунок 9 – Основные классы программной системы

Класс *Segment* описывает сегмент манипулятора. Методы этого класса позволяют задать такие параметры сегмента, как его длина L_i , диаметры d_{i-1} , d_i , углы поворота $\varphi_{i,1}$, $\varphi_{i,2}$ и т.д.

Основной класс системы *Trunk* соответствует манипулятору в целом и включает в себя следующие методы:

- метод *void addSegment (...)*, добавляющий в манипулятор новый сегмент;
- метод *bool IsValid (...)*, решающий задачу о допустимости данной конфигурации манипулятора при наличии препятствия;
- метод *Point getTip (...)*, обеспечивающий решение прямой позиционной задачи и возвращающий координаты точки схвата манипулятора.

Класс *CParticleSwarmAlgorithm* предназначен для решения задачи методом роя частиц. Класс содержит следующие основные методы:

- метод *CParticleSwarmAlgorithm (...)*, реализующий алгоритм роя частиц;
- метод *void CountObjectiveFunctions (...)*, обеспечивающий вычисление положения точки схвата для данной частицы;
- метод *void BestGuide (...)*, предназначенный для определения направления оптимального движения каждой из частиц популяции;
- метод *void UpdateBestParticlesArchive (...)*, выполняющий обновления архивов лучших решений;
- метод *void CreateNewPopulation (...)*, создающий новое поколение частиц.

Объектом класса *CParticle* является частица метода роя частиц. Основным методом класса является функция *CparticleCheckDominationToOtherParticle (...)*, реализующая определение недоминируемых частиц.

5. Экспериментальные результаты

Все приведенные ниже результаты получены для семисекционного манипулятора ($n = 7$), в котором длина каждой последующей секции на 15% меньше длины предыдущей, а длина первой секции составляет 45 условных единиц. Углы $\varphi_{i,j}^{\max}$, $i \in [1:7]$, $j \in [1:2]$ приняты одинаковыми и равными $\varphi^{\max}$. Угол раскрытия секторов ε_c во всех случаях равен 5° .

Рассмотрим прежде задачу без препятствий. Эффективность базового метода *SPSO* иллюстрирует рисунок 10.

Рисунок 10 показывает, что, как и следовало ожидать, в данном случае на некоторых участках границы области достижимости имеет место высокая неравномерность плотности ее покрытия.

Аналогичные результаты, полученные с помощью модифицированного метода *SPSO*, приведены на рисунках 11 - 13.

Рисунки показывают, что по сравнению с базовым методом *SPSO* модифицированный метод *SPSO* обеспечивает существенно более равномерное покрытие границы области достижимости.

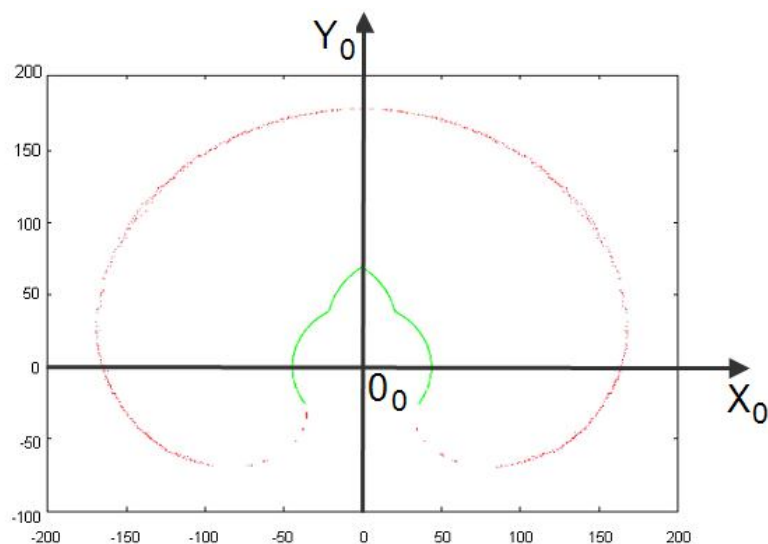


Рисунок 10 – Дискретная аппроксимация границы области достижимости Γ : задача без препятствий; базовый метод *SPSO*; $\varphi^{\max} = 45^\circ$

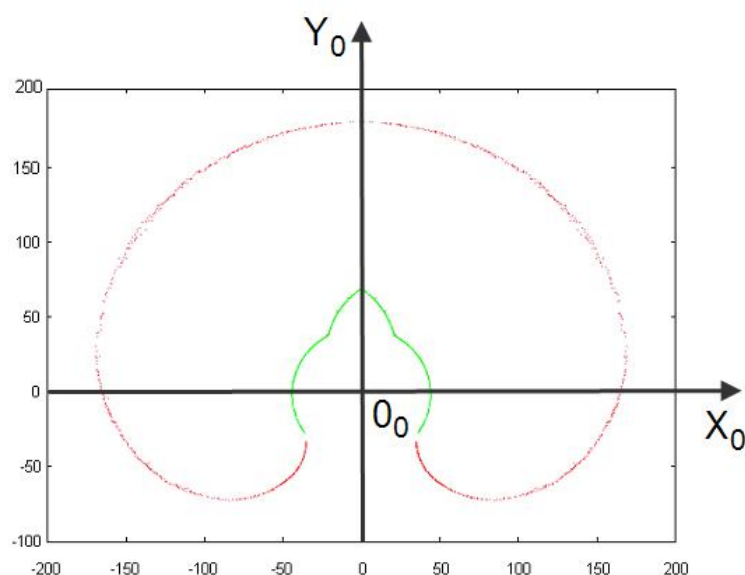


Рисунок 11 – Дискретная аппроксимация границы области достижимости Γ :
задача без препятствий; модифицированный метод *SPSO*; $\varphi^{\max} = 45^\circ$

Эффективность модифицированного метода *SPSO* в задаче с одним сферическим препятствием B_1 иллюстрирует рисунок 14. Прято, что радиус препятствия составляет 20 условных единиц, а его центр находится в точке с координатами (100,0,0).

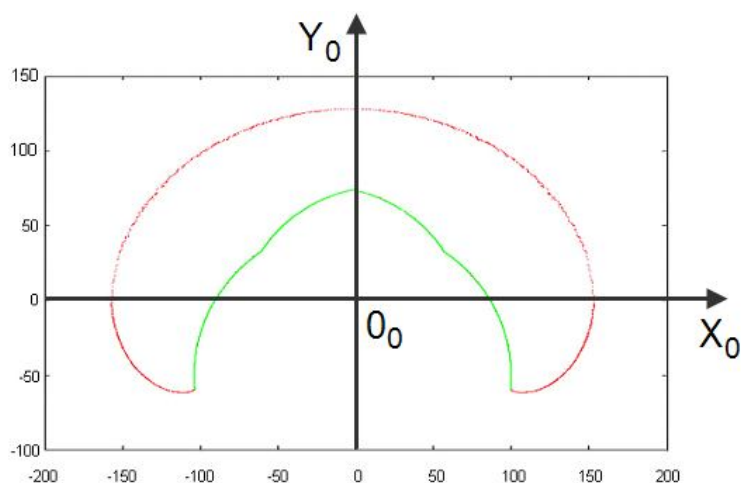


Рисунок 12 – Дискретная аппроксимация границы области достижимости Γ :
задача без препятствий; модифицированный метод *SPSO*; $\varphi^{\max} = 30^\circ$

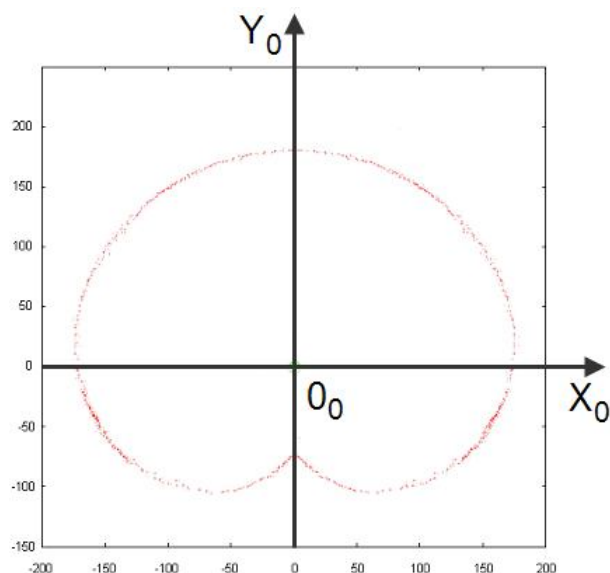


Рисунок 13 – Дискретная аппроксимация границы области достижимости Γ : задача без препятствий; модифицированный метод $SPSO$; $\varphi^{\max} = 60^\circ$

Заключение

В работе предложен метод $SPSO$ для приближенного построения границы области достижимости многосекционного манипулятора. Метод использует алгоритм роя частиц (PSO) для решения задачи глобальной оптимизации и σ -алгоритм для отбора недоминируемых решений. Эффективность метода $SPSO$ продемонстрирована на нескольких задачах построения области достижимости, как в случае отсутствия препятствий, так и в случае их наличия.

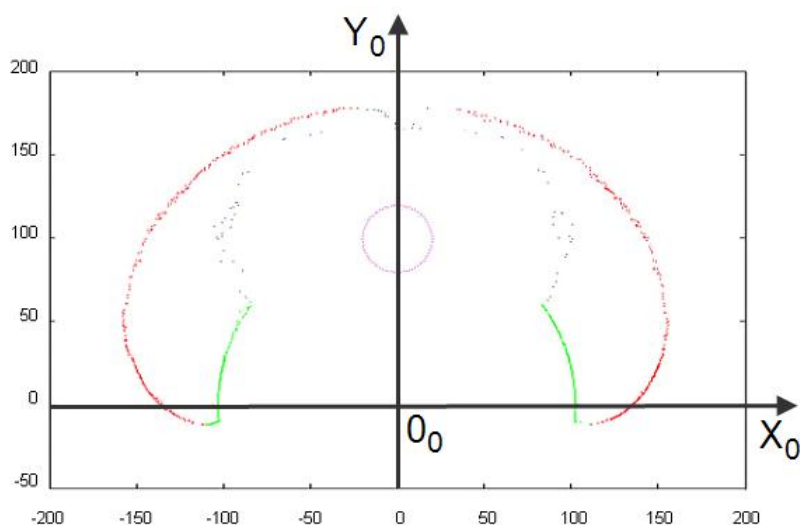


Рисунок 14 – Дискретная аппроксимация границы области достижимости Γ :
задача с одним сферическим препятствием B_1 ; модифицированный
метод *SPSO*; $\varphi^{\max} = 30^0$

Результаты исследования показывают, что метод обеспечивает вполне удовлетворительное качество построения границы области достижимости в задаче без препятствий. В задаче с препятствиями требуется разработка мер, обеспечивающих более равномерное покрытие этой границы.

Вариантом рассмотренной задачи построения области достижимости является задача, в которой в каждую из достижимых точек пространства схват манипулятора должен приблизиться с заданной ориентацией [1]. В развитие работы планируется модификация метода *SPSO* для решения данной задачи.

Как отмечалось выше, поскольку метод *SPSO* использует только плоские конфигурации манипулятора, в задаче с препятствиями этот метод позволяет корректно определить границу ОДМ только в случае одного центрально симметричного препятствия. Поэтому планируется развитие метода в направлении использования произвольных пространственных конфигураций манипулятора.

При большом числе секций манипулятора и высокой требуемой точности определения границ его области достижимости метод *SPSO* требует высоких вычислительных ресурсов. Поэтому в развитие работы планируется также разработка параллельных вариантов метода, ориентированных на параллельные вычислительные системы с различной архитектурой – *MIMD*-системы с общей и распределенной памятью, *SIMD*-системы, например графические процессорные устройства и т.д.

Работа выполнена в рамках аналитической ведомственной целевой программы «Развитие потенциала высшей школы (2009 – 2010 годы)», проект 2.1.2/1509.

СПИСОК ЛИТЕРАТУРЫ

1. Зенкевич С.Л., Ющенко А.С. Основы управления манипуляционными роботами.- М.: Изд-во МГТУ им. Н.Э. Баумана, 2004.- 480 с.
2. Карпенко А.П., Волкоморов С.В., Каганов Ю.Т. Моделирование и оптимизация некоторых параллельных механизмов // Информационные технологии, Приложение, 2010, №5, с. 1-32.
3. Карпенко А.П., Волкоморов С.В., Мартынюк В.А. Анализ кинематики параллельных механизмов средствами системы автоматизированного проектирования САПР // Информационные технологии, 2010, №11, с. 45-52.
4. Карпенко А.П., Волкоморов С.В. Планирование оптимальной целевой конфигурации робота-манипулятора типа «хобот» // Мехатроника, автоматизация, управление, 2010, №8, с. 30–36.
5. Hu X., Eberhart R. Multiobjective optimization using dynamic neighborhood particle swarm optimization / World Congress on Computational Intelligence. Proceeding, 2002, p. 1677–1681.
6. Srinivas N., Deb K. Multiobjective optimization using nondominated sorting in genetic algorithms // Evolutionary Computation, 1994, vol. 2, p. 221–248.
7. Антух А.Э., Карпенко А.П., Семенихин А.С., Хасанова Р.В. Построение множества Парето методом роя частиц на графических процессорах архитектуры CUDA / Научный сервис в сети Интернет: суперкомпьютерные центры и задачи. Труды международной суперкомпьютерной конференции. – М.: Изд-во МГУ, 2010.– с.274-280.
8. Карпенко А.П., Селиверстов Е.Ю. Глобальная оптимизация методом роя частиц. Обзор // Информационные технологии, 2010, № 2, с. 25-34.
9. Штойер Р. Многокритериальная оптимизация. Теория, вычисления и приложения.– М.: Радио и связь, 1992.- 504 с.
10. Подиновский В.В., Ногин В.Д. Парето-оптимальные решения многокритериальных задач.- М.: "Наука", 1982.- 256 с.