*ISSN 1814-1196*
*Научный вестник НГТУ*
*том 69, № 4, 2017, с. 33–46*

*http://journals.nstu.ru/vestnik*
*Science Bulletin of the NSTU*
*Vol. 69, No. 4, 2017, pp. 33–46*

МОДЕЛИРОВАНИЕ ПРОЦЕССОВ И УСТРОЙСТВ

MODELING OF PROCESSES AND DEVICES

УДК 519.7

# Алгоритм движения манипулятора в неизвестной статической среде*

**П.К. ЛОПАТИН**

*660014, РФ, г. Красноярск, пр. им. газеты «Красноярский рабочий», 31, Сибирский государственный университет науки и технологий имени академика М.Ф. Решетнева, кандидат технических наук, доцент. E-mail: robot-2006@yandex.ru*

Рассмотрена следующая задача управления манипуляционным роботом (МР) в неизвестной среде: МР, выдвинувшись из стартовой конфигурации и двигаясь в среде с неизвестными статическими препятствиями только по разрешенным состояниям, должен либо захватить своим схватом заданный объект в какой-либо разрешенной конфигурации, либо выдать обоснованное заключение о том, что объект не может быть захвачен ни в одной разрешенной конфигурации. Представлен алгоритм решения задачи для непрерывного конфигурационного пространства, сформулирована и доказана теорема, утверждающая, что, исполняя алгоритм, МР решит задачу за конечное число шагов. Показано, что решение данной задачи сводится к решению конечного числа задач ПИ – планирования пути в среде с известными запрещенными состояниями с последующим его исполнением. Процедура ПИ вызывается в точках смены пути – в точках, за которыми обнаруживается ранее неизвестное запрещенное состояние. Показано, что число точек смены пути будет конечным. Манипуляционный робот снабжен сенсорной системой (СС), которая при нахождении МР в каждой точке смены пути поставляет информацию об окрестности текущей точки. В настоящей статье предполагается, что окрестность может иметь произвольную форму и размер, а не только гипершар, как это предполагалось в предыдущих наших работах. Определены минимальные объемы надежной информации, которую должна поставлять СС в точке смены пути. Указаны пути использования ненадежной информации, поставляемой СС с тем, чтобы задача была решена за конечное число шагов. Объект и препятствия могут иметь произвольные формы, размеры и расположение.

**Ключевые слова:** робот, манипуляционный робот, планирование пути, траектория, управление, сенсорная система, неизвестная среда, захват объекта, препятствия, достижимость, недостижимость

DOI: 10.17212/1814-1196-2017-4-33-46

## ВВЕДЕНИЕ

При управлении манипуляционными роботами типичной является следующая проблема: МР должен выдвинуться из стартовой конфигурации $\mathbf{q}^0$ и захватить своим схватом некоторый объект Obj. При этом Obj может

* *Статья получена 01 ноября 2017 г.*

быть захвачен не в одной, а в нескольких целевых конфигурациях $\mathbf{q}_i^{\mathrm{T}}$, все эти конфигурации образуют целевое множество $\mathrm{B_T}$. В данной статье рассмотрен случай, когда все конфигурации из $\mathrm{B_T}$ вычислены заранее и $\mathrm{B_T}$ во время движения МР не пополняется.

МР представляется в пространстве конфигураций (пространстве обобщенных координат) как точка. Функционирование МР должно происходить в пределах ограниченной области $X$ конфигурационного пространства. Будем считать, что область $X$ имеет такой вид, что для любого $\mathbf{q} \in X$ выполняются неравенства:

$$\mathbf{a}^1 \le \mathbf{q} \le \mathbf{a}^2, \tag{1}$$

где $\mathbf{a}^1 = (\mathrm{a}_1^{\ 1}, \mathrm{a}_2^{\ 1}, \ldots, \mathrm{a}_n^{\ 1})$ – вектор нижних ограничений на значения обобщённых координат, $\mathbf{a}^2 = (\mathrm{a}_1^{\ 2}, \mathrm{a}_2^{\ 2}, \ldots, \mathrm{a}_n^{\ 2})$ – вектор верхних ограничений на значения обобщенных координат, $\mathbf{q} = (\mathrm{q}_1, \mathrm{q}_2, \ldots, \mathrm{q}_n)$ – вектор обобщенных координат МР. Таким образом, область $X$ представляет собой гиперпараллелепипед. При этом:

1) точки, находящиеся вне (1), квалифицируются как запрещенные;

2) внутри $X$ также могут присутствовать запрещенные состояния, которые можно вычислить заранее. Например, это те состояния (конфигурации), в которых происходит недопустимое взаимопересечение звеньев.

Кроме того, запрещенной является та конфигурация, в которой МР налегает на препятствия. В условиях неизвестной среды все такие конфигурации вычислить заранее невозможно. Перед началом движения информации о запрещенных состояниях в $X$ нет или она неполна. Точки из $X$, про которые нет достоверной информации о том, что они запрещенные, считаем разрешенными.

Рассмотрим теперь точки из $\mathrm{B_T}$. Достижимой точкой $\mathbf{q}^{\mathrm{T}} \in \mathrm{B_T}$ будем считать только ту точку, которая удовлетворяет следующим двум критериям: 1) она не является запрещенной, 2) в нее можно попасть за конечное число шагов из $\mathbf{q}^0$, двигаясь в $X$ по разрешенным состояниям. Точки из $\mathrm{B_T}$, не удовлетворяющие хотя бы одному из двух таких критериев, считаем недостижимыми.

## 1. ПОСТАНОВКА ЗАДАЧИ

Сформулируем следующую задачу управления МР в неизвестной статической среде: даны стартовая конфигурация МР $\mathbf{q}^0$ и целевое множество конфигураций $\mathrm{B_T}$. Требуется предложить алгоритм, который за конечное число шагов либо передвинет МР из $\mathbf{q}^0$ в хотя бы одно достижимое состояние из множества $\mathrm{B_T}$, либо выдаст обоснованный ответ о том, что ни одно состояние из целевого множества $\mathrm{B_T}$ не является достижимым.

Примем следующие допущения.

1.1. Считаем, что $\mathrm{B_T}$ состоит из конечного числа $NBT$ точек, т. е. $\mathrm{B_T} = \{\mathbf{q}^{\mathrm{T1}}, \mathbf{q}^{\mathrm{T2}}, \ldots, \mathbf{q}^{\mathrm{TNBT}}\}$; $\mathrm{B_T}$ не пополняется, и поэтому $NBT$ не возрастает. Каждая точка в $\mathrm{B_T}$ встречается только один раз. Координаты каждой точки из $\mathrm{B_T}$ определены достоверно. Множество $X$ непрерывно.

1.2. Расположение препятствий, их форма и количество не изменяются в течение всего времени движения МР.

1.3. МР имеет сенсорную систему (СС), которая, будучи включена в точке смены пути $\mathbf{q}^n$, $n = 0, 1, 2, \ldots$, способна поставить информацию об $r$-окрестности этой точки. Точка смены пути $\mathbf{q}^n$, $n = 0, 1, 2, \ldots$ – это текущая точка МР, в которой происходит планирование нового пути. Текущая точка МР – это точка, в которой МР в настоящий момент времени находится; $r$-окрестность $\mathbf{q}^n$ – это гипершар в $X$ с центром в $\mathbf{q}^n$ и радиусом $r > 0$. Заметим, что $r$-окрестность, состоящая только из точек, непосредственно прилегающих к $\mathbf{q}^n$ (рис. 1), слишком мала. Наименьшая допустимая $r$-окрестность $\mathbf{q}^n$ состоит не только из точек $\mathbf{q}$, непосредственно прилегающих к $\mathbf{q}^n$, но и из всех точек, непосредственно прилегающих к каждой из точек $\mathbf{q}$ (рис. 2).

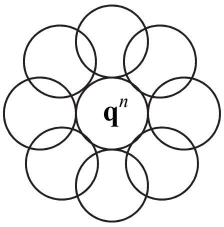


*Рис. 1.* Такая $r$-окрестность $\mathbf{q}^n$ слишком мала

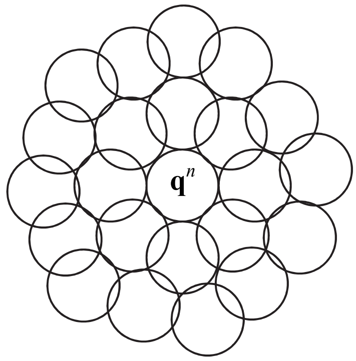


*Рис. 2.* Наименьшая возможная $r$-окрестность $\mathbf{q}^n$

Иначе говоря, мы требуем, чтобы радиус $r$ был таков, чтобы в $r$-окрестность можно было вписать ε-окрестность – гипершар с центром в $\mathbf{q}^n$ и радиусом $0 < \varepsilon < r$ (рис. 3).

Обозначим через $\mathrm{Y}(\mathbf{q}^n)$ множество всех точек, образующих $r$-окрестность $\mathbf{q}^n$. Считаем, что СС поставляет информацию о каждой точке $\mathbf{q}^* \in \mathrm{Y}(\mathbf{q}^n)$ – разрешенная она или запрещенная, и эта информация является точной и надежной. Точная информация означает, что если МР запросил СС поставить информацию о точке $\mathbf{q}^*$, то СС поставляет информацию именно о точке $\mathbf{q}^*$, а не о какой-нибудь $\mathbf{q}^{**} \neq \mathbf{q}^*$. Надежная информация означает, что если СС сообщает, что точка $\mathbf{q}^*$ является запрещенной, то она и в действительности является запрещенной, а если СС сообщает, что $\mathbf{q}^*$ – разрешенная, то она и в действительности разрешенная. Далее точную и надежную информацию будем называть просто надежной. Разрешенные точки из $\mathrm{Y}(\mathbf{q}^n)$ образуют множество $\mathrm{Z}(\mathbf{q}^n)$, запрещенные точки из $\mathrm{Y}(\mathbf{q}^n)$ образуют множество $\mathrm{Q}(\mathbf{q}^n)$, $\mathrm{Z}(\mathbf{q}^n) \cup \mathrm{Q}(\mathbf{q}^n) = \mathrm{Y}(\mathbf{q}^n)$. Считаем, что у нас есть метод для записи множеств $\mathrm{Y}(\mathbf{q}^n)$, $\mathrm{Z}(\mathbf{q}^n)$, $\mathrm{Q}(\mathbf{q}^n)$ (например, с помощью списков, формул и т. п.).

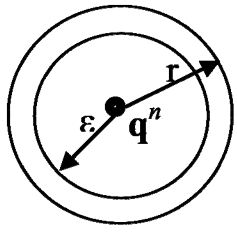


*Рис. 3.* Размер $r$ должен давать возможность вписать в $r$-окрестность ε-окрестность с центром в $\mathbf{q}^n$ и $0 < \varepsilon < r$

Кроме того, допустим, что, когда МР находится в произвольной точке смены пути $\mathbf{q}^n$, $n = 0, 1, 2,\dots$, его СС может поставить информацию о множестве точек Y1($\mathbf{q}^n$)$\in X$. Число точек в множестве Y1($\mathbf{q}^n$), $n = 0, 1, 2,\dots$, его форма и размеры могут быть, вообще говоря, произвольными для любой $\mathbf{q}^n$, $n = 0, 1, 2,\dots$, определяются конструкцией СС, условиями ее использования, потребностями пользователя СС и сообщаются для СС в каждой $\mathbf{q}^n$. Такая ситуация возникает, например, когда при нахождении МР в $\mathbf{q}^n$ система управления МР получает фотоснимок окружения МР в декартовом пространстве, иначе говоря, МР получает информацию о множестве Y1CS точек декартова пространства. Затем Y1CS преобразуется в множество Y1($\mathbf{q}^n$), например, путем решения обратной задачи кинематики. Считаем, что все точки из Y1($\mathbf{q}^n$), про которые СС поставила надежную информацию о том, что они разрешенные, записываются в множество Z1($\mathbf{q}^n$). Все точки из Y1($\mathbf{q}^n$), про которые СС поставила надежную информацию о том, что они запрещенные, записываются в множество Q1($\mathbf{q}^n$). Если МР включил СС, находясь в конфигурации $\mathbf{q}^n$, $n = 0, 1, 2,\dots$ то он остается в $\mathbf{q}^n$ до тех пор, пока не будут сформированы множества Y($\mathbf{q}^n$), Z($\mathbf{q}^n$), Q($\mathbf{q}^n$), Y1($\mathbf{q}^n$), Z1($\mathbf{q}^n$), Q1($\mathbf{q}^n$).

Все точки, про которые есть надежная информация о том, что они запрещенные, записываются в множество FRBDN, т. е.

$$\mathrm{FRBDN} = \bigcup_{i=0}^{n} \mathrm{Q}(\mathbf{q}^i),$$

$n = 0, 1, 2,\dots$

Структура СС здесь не рассматривается.

1.4. Считаем, что у нас есть программа ПИ($\mathbf{q}^n$, $\mathbf{q}^{\mathrm{T}}$, FRBDN, X), которая решает задачу ПИ. Задача ПИ такова: за конечное число шагов сгенерировать путь (последовательность точек, непрерывно следующих одна за другой) L($\mathbf{q}^n$, $\mathbf{q}^{\mathrm{T}}$), удовлетворяющий следующим условиям:

1.4.1. L($\mathbf{q}^n$, $\mathbf{q}^{\mathrm{T}}$) соединяет $\mathbf{q}^n$ и $\mathbf{q}^{\mathrm{T}}$;

1.4.2. L($\mathbf{q}^n$, $\mathbf{q}^{\mathrm{T}}$) $\cap$ FRBDN = Ø;

1.4.3. L($\mathbf{q}^n$, $\mathbf{q}^{\mathrm{T}}$) $\in X$ в случае, если в $X$ существует хотя бы один путь, удовлетворяющий условиям 1.4.1–1.4.3 либо обнаружено, что в $X$ нет пути, удовлетворяющего условиям 1.4.1–1.4.3. Для решения задачи ПИ предлагается использовать какой-либо существующий алгоритм (см., например, [1]) или специально разработанный. Условие 1.4.2 означает, что ни одна точка из L($\mathbf{q}^n$, $\mathbf{q}^{\mathrm{T}}$) не должна совпадать ни с одной точкой из FRBDN. Условие 1.4.3 означает, что каждая точка из L($\mathbf{q}^n$, $\mathbf{q}^{\mathrm{T}}$) должна удовлетворять (1). Если ПИ() удается сгенерировать L($\mathbf{q}^n$, $\mathbf{q}^{\mathrm{T}}$) – возвращается единица, в противном случае – ноль.

1.5. Считаем, что у нас есть программа Процедура1($\mathrm{B_T}$, *NBT*, FRBDN). Процедура1() в момент вызова получает множество $\mathrm{B_T}$, число *NBT* точек в $\mathrm{B_T}$, множество FRBDN запрещенных точек. Процедура1() выбрасывает из $\mathrm{B_T}$ точки, совпадающие с точками из FRBDN, после чего перенумеровывает оставшиеся в $\mathrm{B_T}$ точки непрерывной нумерацией начиная с единицы, и в *NBT* записывается оставшееся число точек.

1.6. Считаем, что у нас есть программа Процедура2(). Процедура2() работает в соответствии со следующим псевдокодом:

| Процедура2($\mathbf{q}^n$) | |
|---|---|
| 1 | МР находится в точке $\mathbf{q}^n$. СС поставляет информацию об Y($\mathbf{q}^n$) и Y1($\mathbf{q}^n$) и формирует FRBDN |
| 2 | if (QBT:=FRBDN$\cap$B$_T$$\neq$0)<br>/*выбросить запрещенные точки из B$_T$*/<br>*NBT*:=Процедура1(B$_T$, *NBT*, FRBDN);<br>endif |
| 3 | return(*NBT*) |

1.7. МР исполняет любой путь L($\mathbf{q}^n$, $\mathbf{q}^T$) следующим образом. Пусть МР находится в точке $\mathbf{q}^* \in$ L($\mathbf{q}^n$, $\mathbf{q}^T$). МР получает надежную информацию о точке $\mathbf{q}^{**} \in$ L($\mathbf{q}^n$, $\mathbf{q}^T$), следующей за $\mathbf{q}^*$. Если $\mathbf{q}^{**}$ разрешенная, то МР переходит в $\mathbf{q}^{**}$; если $\mathbf{q}^{**}$ запрещенная, то МР остается в $\mathbf{q}^*$.

Поскольку B$_T$ состоит из конечного числа точек, то для решения задачи в принципе можно использовать уже существующие подходы, описанные, например, в [2–5]. В [6] проведено сравнение этих подходов с нашим. В настоящей статье мы даем алгоритм для исследования достижимости объекта. Сформулированы также требования к минимальным объемам надежной информации, которую должна поставлять СС, и пути использования ненадежной информации.

Ниже приведен плгоритм для решения задачи, где $\mathbf{q}^c$ – это текущая конфигурация МР, перед началом движения $\mathbf{q}^c = \mathbf{q}^0$, во время движения алгоритм 1 может быть вызван из других текущих конфигураций МР.

## 2. АЛГОРИТМ ДЛЯ НЕИЗВЕСТНОЙ СРЕДЫ

| Алгоритм | |
|---|---|
| 1 | $n$:=0. FRBDN:= Ø |
| 2 | while (*NBT*$\neq$0) |
| 3 | $\mathbf{q}^c$ – текущая конфигурация МР. В качестве $\mathbf{q}^T$ рассматриваем первую точку из B$_T$ |
| 4 | if (объект_захвачен:=Алгоритм1($\mathbf{q}^c$, $\mathbf{q}^T$) =ДА)<br>Obj захвачен в $\mathbf{q}^T$;<br>go to 7;<br>endif |
| 5 | endwhile |
| 6 | Obj не может быть захвачен ни в одной разрешенной конфигурации |
| 7 | Конец алгоритма |

Алгоритм1($\mathbf{q}^c$, $\mathbf{q}^T$) определяет, достижима ли точка $\mathbf{q}^T$ из $\mathbf{q}^c$.

| Алгоритм1($\mathbf{q}^c$, $\mathbf{q}^T$) | |
|---|---|
| 1 | $\mathbf{q}^n$:=$\mathbf{q}^c$ |
| 2 | /*Включить СС и модифицировать $B_T$ */<br>*NBT*:=Процедура2($\mathbf{q}^n$) |
| 3 | /*Если $\mathbf{q}^T$ была выброшена из $B_T$, то в качестве $\mathbf{q}^T$ надо рассматривать другую точку из $B_T$ */<br>if ($\mathbf{q}^T \notin B_T$)<br>return(объект_захвачен:=НЕТ);<br>endif |
| 4 | /*МР пытается сгенерировать путь L($\mathbf{q}^n$, $\mathbf{q}^T$) в X*/<br>if (ПИ($\mathbf{q}^n$,$\mathbf{q}^T$,FRBDN,X)=0)<br>/*Если попытка оказалась неуспешной, то $\mathbf{q}^T$ – недостижима*/<br>FRBDN:=FRBDN$\cup\mathbf{q}^T$;<br>*NBT*:=Процедура1($B_T$, *NBT*, FRBDN);<br>return(объект_захвачен:=НЕТ);<br>endif<br>/*Если попытка была успешной, перейти на шаг 5*/ |
| 5 | МР начинает исполнять путь L($\mathbf{q}^n$, $\mathbf{q}^T$). Может быть два исхода исполнения: 1) МР придет в разрешенную точку $\mathbf{q}^T \in B_T$, return (объект_захвачен:=ДА); 2) МР придет в такую точку $\mathbf{q}^c \in$ L($\mathbf{q}^n$, $\mathbf{q}^T$), что следующая за $\mathbf{q}^c$ точка $\mathbf{q}^* \in$ L($\mathbf{q}^n$, $\mathbf{q}^T$) является запрещенной. В этом случае МР исполняет: $n$:=$n$+1; go to 1 |

**Теорема.** Алгоритм управления манипуляционным роботом решает поставленную задачу за конечное число шагов.

**Доказательство.** Рассмотрим сначала случай, когда $Y1(q^n) = \varnothing$, $\forall n = 0, 1, 2,\ldots$ и $r$-окрестность точки $q^n$ $\forall n = 0, 1, 2,\ldots$ имеет минимальный размер, показанный на рис. 2.

Алгоритм исследует достижимость конечного числа точек из $B_T$. Для определения достижимости произвольной точки $q_i^T \in B_T$ алгоритм 1 вызывается один раз. Таким образом, исполнение алгоритма сводится к конечному числу вызовов алгоритма 1. Поэтому, чтобы показать, что алгоритм будет исполнен за конечное число шагов, необходимо показать, что алгоритм 1 будет исполнен за конечное число шагов для произвольных $\mathbf{q}^c$ и $\mathbf{q}^T$.

Перед тем как рассматривать алгоритм 1, сформулируем следующий вопрос: «Получена ли информация о том, что $\mathbf{q}^T$ достижима или недостижима?» Может быть три ответа на вопрос: 1) Да, мы получили информацию о том, что $\mathbf{q}^T$ достижима; 2) Да, мы получили информацию о том, что $\mathbf{q}^T$ недостижима; 3) Нет, мы еще не получили информации о том, достижима или недостижима $\mathbf{q}^T$.

На шаге 1 алгоритма 1 можно видеть точки смены пути $\mathbf{q}^n$, $n = 0, 1, 2,\ldots$ Когда МР находится в такой точке, на шаге 2 вызывается процедура 2(), и после исполнения шага 3 мы получаем один из двух ответов на вопрос: если мы получили информацию о том, что $\mathbf{q}^{\mathrm{T}} \in$FRBDN или все $\mathrm{B_T} \subset$FRBDN, то мы получаем ответ 2, во всех остальных случаях получаем ответ 3. Если получен ответ 3, то алгоритм 1 переходит на шаг 4, где вызывается процедура ПИ(). Если ПИ() не удается сгенерировать путь, мы получаем ответ 2; если ПИ() сгенерировала путь $\mathrm{L}(\mathbf{q}^n, \mathbf{q}^{\mathrm{T}})$, мы получаем ответ 3 и алгоритм 1 переходит на шаг 5. На шаге 5 МР начинает исполнять $\mathrm{L}(\mathbf{q}^n, \mathbf{q}^{\mathrm{T}})$ в соответствии с пунктом 1.7. Если исполнение пути завершается первым исходом, т. е. МР, исполняя $\mathrm{L}(\mathbf{q}^n, \mathbf{q}^{\mathrm{T}})$ и не встретив запрещенных точек, придет в $\mathbf{q}^{\mathrm{T}}$ и обнаружит, что и она разрешенная, то мы получим ответ 1, и исполнение алгоритма завершится с выводом «Obj захвачен в $\mathbf{q}^{\mathrm{T}}$». Исполнение любого пути $\mathrm{L}(\mathbf{q}^n, \mathbf{q}^{\mathrm{T}})$ происходит за конечное число шагов потому, что длина любого $\mathrm{L}(\mathbf{q}^n, \mathbf{q}^{\mathrm{T}})$ конечна. Но может случиться так, что МР придет в такую точку $\mathbf{q}^* \in \mathrm{L}(\mathbf{q}^n, \mathbf{q}^{\mathrm{T}})$, что следующая за $\mathbf{q}^*$ точка $\mathbf{q}^{**} \in \mathrm{L}(\mathbf{q}^n, \mathbf{q}^{\mathrm{T}})$ является запрещенной. В этом случае МР исполняет $n{:=}n{+}1$; $\mathbf{q}^n{:=}\mathbf{q}^*$ (т. е. $\mathbf{q}^*$ становится новой точкой смены пути) и алгоритм 1 переходит на шаг 1. Покажем, что число NPATH точек смены пути $\mathbf{q}^n$, $n = 0, 1, 2,\ldots$, NPATH-1 будет конечным, и все они будут различны.

Покажем, что все точки смены пути различны. Предположим, что МР сменил путь в точке $\mathbf{q}^s$, а затем вновь сменил путь в точке $\mathbf{q}^p$, т. е. $s < p$. Покажем, что $\mathbf{q}^s \neq \mathbf{q}^p$. Сначала предположим, что $\mathbf{q}^s = \mathbf{q}^p$. В таком случае будет $\mathrm{Q}(\mathbf{q}^s) = \mathrm{Q}(\mathbf{q}^p)$. Как в точке $\mathbf{q}^s$, так и в последующих точках смены пути МР генерировал путь, ни одна точка которого не совпадала ни с одной точкой каждого из множеств $\mathrm{Q}(\mathbf{q}^i)$, $i = 0, 1,..., s$. Когда МР попал в точку $\mathbf{q}^p$, он обнаружил, что необходимо менять путь, поскольку следующая за $\mathbf{q}^p$ точка $\mathbf{q}^*$ оказалась ранее неизвестной запрещенной точкой, т. е. $\mathbf{q}^* \in \mathrm{Q}(\mathbf{q}^p)$. Но поскольку $\mathrm{Q}(\mathbf{q}^s) = \mathrm{Q}(\mathbf{q}^p)$, то $\mathbf{q}^* \in \mathrm{Q}(\mathbf{q}^s)$. То есть возникает такая ситуация, что путь МР налег на запрещенную точку, которая должна была быть учтена, а значит, была учтена при планировании этого пути. Такая невозможная ситуация возникла при допущении $\mathbf{q}^s = \mathbf{q}^p$ для произвольных $s$ и $p$. Следовательно, $\mathbf{q}^s \neq \mathbf{q}^p$, т. е. все точки смены пути различны.

Объясним, почему мы потребовали от СС, чтобы она поставляла надежную информацию о каждой точке из $r$-окрестности $\mathbf{q}^n$, $n = 0, 1, 2,\ldots$ (ниже через $\mathbf{q}$ будем обозначать точки, непосредственно прилегающие к $\mathbf{q}^n$, через $\mathbf{qq}$ – точки, непосредственно прилегающие к $\mathbf{q}$). Если мы получили ложную информацию о том, что $\mathbf{q}$ – разрешенная, то $\mathbf{q}^n$ может вновь стать точкой смены пути. И мы не можем гарантировать, что $\mathbf{q}^n$ не будет становиться точкой смены пути бесконечное число раз: если СС поставила ложная информация о $\mathbf{q}$, когда МР был в $\mathbf{q}^n$ в первый раз, то у нас нет гарантии, что не будет поставляться ложная информация о $\mathbf{q}$, когда МР попадет в $\mathbf{q}^n$ в следующий раз. Ситуация может стать еще хуже в случае, если СС будет поставлять ложную информацию не об одной, а о нескольких (многих) точках $\mathbf{q}$ о том, что они разрешенные. Такая же проблема может возникнуть, если СС поставит ложную информацию о том, что точка(-и) $\mathbf{qq}$ – разрешенная(-ые).

Теперь рассмотрим ситуацию, когда СС поставила ложную информацию о том, что некоторая точка $\mathbf{q}$ – запрещенная. Может получиться так, что $\mathbf{q}$ –

единственная разрешенная точка из всех точек, непосредственно прилегающих к $\mathbf{q}^n$, и тогда будет сделано ложное заключение о том, что путь из $\mathbf{q}^n$ не может быть сгенерирован. Такое заключение приведет либо к преждевременному, либо к ложному выводу «Obj не может быть захвачен». Такая же проблема может возникнуть, если СС поставит ложную информацию о том, что некоторая **qq** – запрещенная.

Покажем, что число точек смены пути конечно. Предположим, что оно бесконечно. Все точки смены пути должны удовлетворять неравенствам (1). Это означает, что последовательность этих точек ограничена. Согласно теореме Больцано–Вейерштрасса из этой последовательности можно извлечь сходящуюся подпоследовательность $\mathbf{q}^i$, $i = 1, 2,\ldots$ В соответствии со свойством Коши сходящихся последовательностей для любого ε можно найти такой номер $s$, что все точки $\mathbf{q}^i$, $i > s$, будут лежать в ε-окрестности $\mathbf{q}^s$. Возьмем $\varepsilon < r$. Рассмотрим произвольную точку смены пути $\mathbf{q}^i$, лежащую в ε-окрестности $\mathbf{q}^s$. Поскольку в $\mathbf{q}^i$ МР вынужден был сменить путь, то это означает, что среди точек, непосредственно прилегающих к $\mathbf{q}^i$, МР встретил ранее неизвестную запрещенную точку. Но все точки, непосредственно прилегающие к $\mathbf{q}^i$, принадлежат $r$-окрестности $\mathbf{q}^s$, и непосредственно прилегающие к $\mathbf{q}^i$ запрещенные точки принадлежат $Q(\mathbf{q}^s)$. Но $Q(\mathbf{q}^s)$ должно было быть, а значит, было принято во внимание, когда происходила генерация путей $L(\mathbf{q}^j, \mathbf{q}^T)$ для каждого $j \geq s$. Итак, предположив, что число точек смены пути бесконечно, мы получим невозможную ситуацию – некоторые точки смены пути будут лежать в ε-окрестности другой точки смены пути. Отсюда получаем, что число NPATH точек смены пути конечно.

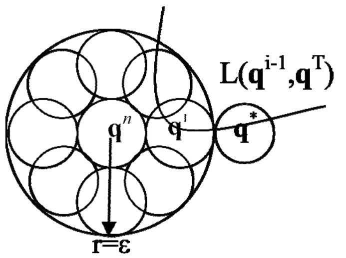


*Рис. 4.* Если $r$-окрестность $\mathbf{q}^n$ слишком мала, то любая прилегающая к ней точка может стать точкой смены пути

Поясним, почему $r$-окрестность точки смены пути $\mathbf{q}^n$, $n = 0, 1, 2,\ldots$, не может быть настолько малой, какой показана на рис. 1. Если бы $r$-окрестность имела такой размер, то в качестве ε-окрестности пришлось бы рассматривать либо точку $\mathbf{q}^n$, либо саму $r$-окрестность. Тогда любая точка $\mathbf{q}^i$, непосредственно прилегающая к $\mathbf{q}^n$, может стать точкой смены пути (рис. 4) по следующей причине.

Пусть МР, будучи в точке смены пути $\mathbf{q}^n$, $n = 0, 1, 2,\ldots$, получил информацию об $r$-окрестности, размер которой показан на рис. 1. После этого момента МР имеет информацию о каждой точке $\mathbf{q}^n$ из $r$-окрестности. Впоследствии был сгенерирован путь $L(\mathbf{q}^{i-1}, \mathbf{q}^T)$, $i - 1 \geq n$, и МР исполняет этот путь.

$L(\mathbf{q}^{i-1}, \mathbf{q}^T)$ содержит точки $\mathbf{q}^i$ и $\mathbf{q}^*$ ($\mathbf{q}^*$ – следующая точка после $\mathbf{q}^i$ в пути $L(\mathbf{q}^{i-1}, \mathbf{q}^T)$); $\mathbf{q}^i$ является разрешенной, поскольку СС поставила информацию о ней, когда МР был в $\mathbf{q}^n$ и $L(\mathbf{q}^{i-1}, \mathbf{q}^T)$ содержит ее, но если $\mathbf{q}^*$ запрещенная, то $\mathbf{q}^i$ становится точкой смены пути, и $\mathbf{q}^i$ принадлежит ε-окрестности $\mathbf{q}^s$ (см. рис. 4). Можно видеть, что любая точка, непосредственно прилегающая к $\mathbf{q}^n$ может стать точкой смены пути. Таким образом, если $r$-окрестность будет настолько мала, как это показано на рис. 1, то может получиться, что в ε-окрестности точки смены пути $\mathbf{q}^n$ будет бесконечное число точек смены пути. Но если минимальный размер $r$-окрестности точек смены пути $\mathbf{q}^n$, $n = 0, 1, 2,\ldots$, будет таким, каким показан на рис. 2, и мы будем выбирать $0 < \varepsilon < r$, то МР будет иметь информацию о каждой точке, непосредственно прилегающей к каждой точке $\mathbf{q}$ из $\varepsilon$-окрестности $\mathbf{q}^n$, и благодаря этому среди точек, непосредственно прилегающих к $\mathbf{q}$, не появится ранее неизвестных запрещенных точек.

Итак, число точек смены пути $\mathbf{q}^n$, $n = 0, 1, 2,\ldots$, NPATH-1 конечно, и все они различны. Пусть МР пришел в последнюю точку смены пути $\mathbf{q}^{\text{NPATH-1}}$. Это означает, что МР сделал конечное число шагов, двигаясь из $\mathbf{q}^0$, и все еще имеет ответ 3 на вопрос. В точке $\mathbf{q}^{\text{NPATH-1}}$ вызывается Процедура 2(). Если мы получаем ответ 2, то алгоритм 1 заканчивает свою работу. Если мы получили ответ 3, то вызывается процедура ПИ(). Если процедуре ПИ() не удается сгенерировать путь, то получаем ответ 2, если удается, то получаем ответ 3. МР начинает исполнять путь $L(\mathbf{q}^{\text{NPATH-1}}, \mathbf{q}^{\text{T}})$. Поскольку точка смены пути была последней, МР не встретит ранее неизвестной запрещенной точки, достигнет $\mathbf{q}^{\text{T}}$ за конечное число шагов, поскольку длина $L(\mathbf{q}^{\text{NPATH-1}}, \mathbf{q}^{\text{T}})$ конечна, обнаружит, что точка $\mathbf{q}^{\text{T}}$ – разрешенная, будет получен ответ 1, и исполнение алгоритма 1 и алгоритма будет закончено с выводом о том, что Obj захвачен в $\mathbf{q}^{\text{T}}$.

Итак, ответ на вопрос, достижима ли произвольная точка $\mathbf{q}^{\text{T}}$, будет получен за конечное число шагов. Другими словами, алгоритм 1 будет исполнен за конечное число шагов для произвольных $\mathbf{q}^{\text{c}}$ и $\mathbf{q}^{\text{T}}$. Исполнение алгоритма сводится к конечному числу вызовов алгоритма 1. Отсюда видно, что, исполняя алгоритм, МР решит задачу за конечное число шагов.

Выше мы показали, что даже если $\text{Y1}(\mathbf{q}^n)= \varnothing$, $\forall n = 0, 1, 2,\ldots$, т. е.

$$\text{FRBDN} = \bigcup_{i=0}^{n} \text{Q}(\mathbf{q}^i),$$

и $r$-окрестность $\mathbf{q}^n$ $\forall n = 0, 1, 2,\ldots$ имеет минимальный размер, показанный на рис. 2, то задача тем не менее будет решена и можно не собирать дополнительную информацию. Но если она поставляется, то ее можно использовать: можно видеть, что доказательство будет действительным и для случая, когда

$$\text{FRBDN} = \bigcup_{i=0}^{n} \text{Q}(\mathbf{q}^i) \bigcup (\bigcup_{i=0}^{n} \text{Q1}(\mathbf{q}^i)),$$

$n = 0, 1, 2,\ldots$

Главное, чтобы информация о каждой точке из FRBDN была надежной, и если ПИ() вызывается при нахождении МР в точке $\mathbf{q}^n$, $n = 0, 1, 2,\ldots$, то FRBDN должно содержать все точки множества

$$\bigcup_{i=0}^{n} \text{Q1}(\mathbf{q}^i).$$

Поскольку задача может быть решена без сбора информации о множестве $\bigcup_{i=0}^{n} Q(\mathbf{q}^i)$ для $\forall n = 0, 1, 2,\ldots$, то FRBDN может содержать ноль, несколько точек или все точки этого множества. Но если ПИ() получает информацию о точках этого множества, то эта информация должна быть надежной. Выше мы уже показывали, что если будет поставлена ложная информация о том, что какая-то точка является запрещенной, то может быть сделан ложный вывод о том, что путь не может быть спланирован. Если же о какой-то точке

$$\mathbf{q}^* \in \bigcup_{i=0}^{n} Q1(\mathbf{q}^i)$$

не будет сообщено процедуре ПИ(), то такая точка будет считаться разрешенной, МР может прибыть в точку $\mathbf{q}$, предшествующую $\mathbf{q}^*$, и ему придется планировать новый путь, но будет собрана информация о $Q(\mathbf{q})$ и, как мы показали выше, задача будет решена, если информация о множествах $Q1(\mathbf{q}^i)$, $i = 0, 1, 2,\ldots$, не будет собираться и, соответственно, не будет поставляться для процедуры ПИ(). Тем не менее мы рекомендуем включать в FRBDN все известные запрещенные точки – это сократит число перепланирований пути. **Теорема доказана**.

## 3. СЛЕДСТВИЕ

Как мы видели, алгоритм использует только надежную информацию о запрещенных точках. Но возникает вопрос: можно ли и каким образом для решения задачи использовать ненадежную информацию? Рассмотрим следующий пример (рис. 5).

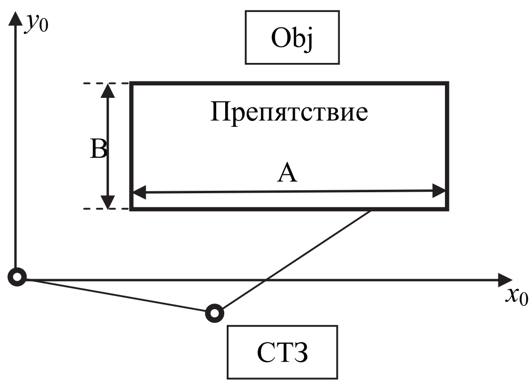


*Рис. 5.* МР может использовать вспомогательные алгоритмы для облегчения решения задачи

Пусть МР, решая задачу в соответствии с алгоритмом и исполняя путь, столкнулся с препятствием. МР имеет искусственную «чувствительную кожу», и также имеется система технического зрения (СТЗ) в фиксированном положении, которая способна только сообщить ширину A препятствия, и, поскольку МР коснулся препятствия, имеется возможность вычислить координаты ($x_0$, $y_0$) каждой точки нижнего ребра препятствия. Предположим, что МР имеет дополнительную информацию о том, что препятствие – прямоугольник. Если бы у нас была информация о высоте B препятствия, то мы бы имели полную информацию о запрещенных точках в декартовом и конфигурационном пространствах. Высоту B МР может также определить в соответствии с некоторыми алгоритмами, ощупывая препятствие. МР может также «предположить» величину B, но такая информация будет ненадежной. Если B достаточно мала, и препятствие не «поглощает» Obj, то в случае, если величина B будет предположена правильно, МР будет способен обойти препятствие и захватить Obj. Но если «предположение» ошибочно (например, препятствие на самом деле «поглощает» Obj) и МР использует какой-то алгоритм, не гарантирующий решения задачи, то МР может совершить множество бесполезных шагов, пытаясь захватить Obj, после которых возникнет необходимость в обращении к алгоритму. Можно ли все-таки, исполняя алгоритм, использовать другие алгоритмы, которые будут поставлять некоторую (в общем случае – ненадежную) информацию о форме и размерах препятствия и каким образом? Ответ даст следующее.

**Следствие.** Добавление конечного числа шагов (по другим алгоритмам) сохраняет конечное число шагов для этой задачи.

Приведем рекомендации к использованию следствия. Исполняя путь $L(\mathbf{q}^n, \mathbf{q}^T)$, $\forall n = 0, 1, \ldots$ на шаге 5 алгоритма 1, МР может сделать сходы с пути в произвольных точках $\mathbf{q}^d$, $d = 0, 1, ..., N_d$, где $N_d$ – конечное число, не большее, чем число точек в $L(\mathbf{q}^n, \mathbf{q}^T)$, а затем вернуться к исполнению алгоритма 1. Сход – это последовательность разрешенных точек, непрерывно следующих одна за другой. Число шагов на каждом сходе должно быть конечным. Число сходов из одной и той же точки $\mathbf{q}^d$ должно быть конечным. Таким образом, конечное число шагов на сходе с пути прибавляется к конечному числу шагов на самом пути. В каждой точке $\mathbf{q}^c$ схода МР может включать СС и получать как надежную, так и ненадежную информацию об окружении МР. Если МР включил СС, находясь в некоторой точке $\mathbf{q}^c$ схода, то он остается в $\mathbf{q}^c$ до тех пор, пока не будут сформированы множества $Y(\mathbf{q}^c)$, $Z(\mathbf{q}^c)$, $Q(\mathbf{q}^c)$, $Y1(\mathbf{q}^c)$, $Z1(\mathbf{q}^c)$, $Q1(\mathbf{q}^c)$.

При назначении точек, по которым МР будет перемещаться во время схода, может использоваться как надежная, так и ненадежная информация, но перемещение из точки в точку должно осуществляться в соответствии с п. 1.7. Если МР, будучи в какой-то точке $\mathbf{q}^c$ схода, получил надежную информацию о запрещенных конфигурациях, то он может пополнить ею множество FRBDN, хотя может и не пополнять. Выше было показано, что МР решит задачу, не осуществляя сходов и соответственно не пополняя FRBDN информацией, полученной во время сходов. Ненадежной информацией FRBDN не пополняется. Тем не менее для сокращения количества шагов мы, разумеется, рекомендуем пополнять FRBDN точками, в отношении которых получена надежная информация о том, что они запрещенные.

Рассмотрим ситуацию, когда на сходе МР получает надежную информацию о том, что $QBT := FRBDN \cap B_T \neq 0$, т. е. некоторые точки $\mathbf{q}_i^T$ из $B_T$ принадлежат FRBDN (при этом пока считаем, что среди точек $\mathbf{q}_i^T$ нет $\mathbf{q}^T$, т. е. точки, в которую был проложен путь L($\mathbf{q}^n$, $\mathbf{q}^{\mathrm{T}}$), с которого произошел сход). Такие точки можно выбросить из $B_{\mathrm{T}}$, но можно и не выбрасывать – сход произошел с пути L($\mathbf{q}^n$, $\mathbf{q}^{\mathrm{T}}$), т. е. исследуется достижимость точки $\mathbf{q}^{\mathrm{T}}$, за конечное число шагов она будет исследована. И если будет установлена ее недостижимость, то МР начнет исследовать достижимость другой точки из $B_{\mathrm{T}}$, и ее достижимость/недостижимость будет установлена за конечное число шагов. Но для сокращения числа шагов мы рекомендуем исключать из $B_{\mathrm{T}}$ точки, про которые получена надежная информация о том, что они запрещенные.

Теперь рассмотрим случай, когда в некоторой точке $\mathbf{q}^{\mathrm{c}}$ схода с пути L($\mathbf{q}^n$, $\mathbf{q}^{\mathrm{T}}$) получена надежная информация о том, что $\mathbf{q}^{\mathrm{T}}$ – запрещенная (например, ощупав препятствие, МР установил, что оно «поглощает» Obj). В этом случае рекомендуется, чтобы МР исполнил FRBDN:=FRBDN$\cup\mathbf{q}^{\mathrm{T}}$; $n$:=$n$+1; go to шаг 1 алгоритма 1. В то же время повторим, что поскольку МР решит задачу и не осуществляя сходов, то, даже если МР на сходе, обнаружив, что $\mathbf{q}^{\mathrm{T}}$ – запрещенная, не пополнит ею FRBDN или не выбросит ее из $B_{\mathrm{T}}$, все равно он за конечное число шагов обнаружит ее недостижимость, исполняя собственно алгоритм 1, совершив, правда, лишнее число шагов.

Если МР пришел в точку $\mathbf{q}^{\mathrm{T}}\in B_{\mathrm{T}}$ и имеет надежную информацию о том, что $\mathbf{q}^{\mathrm{T}}$ – разрешенная, то производится окончание работы алгоритма с выводом «Obj захвачен в $\mathbf{q}^{\mathrm{T}}$».

Совершив сход из некоторой точки $\mathbf{q}^d$, после конечного числа шагов на сходе прибыв в некоторую точку $\mathbf{q}^{\mathrm{c}}$, МР должен вернуться к исполнению алгоритма 1. При этом МР может выбрать один из двух вариантов:

1) вернуться в $\mathbf{q}^d$ (за конечное число шагов) только по разрешенным точкам (например, по пройденным) и продолжить исполнение L($\mathbf{q}^n$, $\mathbf{q}^{\mathrm{T}}$);

2) выполнить $n$:=$n$+1; $\mathbf{q}^n$:=$\mathbf{q}^{\mathrm{c}}$ и перейти на шаг 1 алгоритма 1, т. е. осуществить перезапуск алгоритма 1. Число таких перезапусков алгоритма 1 должно быть конечным.

Итак, конечное число шагов на сходах добавляется к конечному числу шагов при исполнении путей (и каждого шага алгоритма 1 и алгоритма), и в сумме мы получаем конечное число шагов. Поскольку перед каждым перезапуском алгоритма 1 выполняется конечное число шагов, в сумме мы получим конечное число шагов. Таким образом, МР может совершить конечное число шагов на основе ненадежной информации, но этой информацией нельзя пополнять FRBDN, и эти шаги должны быть совершены на сходах или до запуска алгоритма. Собственно алгоритм и алгоритм 1 должны исполнять свои шаги только на основе надежной информации. Разумеется, использование следствия и алгоритма должно быть направлено на эффективное решение задачи.

## ЗАКЛЮЧЕНИЕ

Рассмотрена задача управления МР в неизвестной статической среде: даны стартовая конфигурация МР $\mathbf{q}^0$ и целевое множество конфигураций $B_{\mathrm{T}}$. Требуется предложить алгоритм, который либо передвинет МР из $\mathbf{q}^0$ в хотя бы одно достижимое состояние из множества $B_{\mathrm{T}}$, либо выдаст обоснованный ответ о том, что ни одно состояние из целевого множества $B_{\mathrm{T}}$ не является достижимым. Представлен алгоритм для решения задачи и доказана теорема, утверждающая, что, исполняя алгоритм, МР решит задачу за конечное число шагов. Информация об окружающей среде поставляется для МР его СС в предопределенных объемах. Описаны минимальные объемы надежной информации, указаны пути использования ненадежной информации, поставляемой СС.

## СПИСОК ЛИТЕРАТУРЫ

1. *Canny J.* The complexity of robot motion planning. – Cambridge, Mass.: MIT Press, 1988.

2. Principles of robot motion: theory, algorithms and implementations / H. Choset, K.M. Lynch, S. Hutchinson, G. Kantor, W. Burgard, L.E. Kavraki, S. Thrun. – Cambridge, Mass.: MIT Press, 2005.

3. *LaValle S.M.* Planning algorithms [Electronic resource]. – Cambridge: Cambridge University Press, 2006. – URL: http://planning.cs.uiuc.edu/ (accessed: 10.12.2017).

4. *LaValle S.M.* Motion planning: the essentials // IEEE Robotics and Automation Magazine. – 2011. – Vol. 18 (1). – P. 79–89.

5. *Chaabani A., Bellamine M.S., Gasmi M.* Motion planning and controlling algorithm for grasping and manipulating moving objects in the presence of obstacles // International Journal on Soft Computing, Artificial Intelligence and Applications. – 2014. – Vol. 3, N 3/4. – P. 1–17.

6. *Lopatin P.* Investigation of a target reachability by a manipulator in an unknown environment // Proceedings of 2016 IEEE International Conference on Mechatronics and Automation, August 7–10, Harbin, China. – Harbin, 2016. – P. 37–42.

*Лопатин Павел Константинович*, кандидат технических наук, доцент кафедры информатики и вычислительной техники Сибирского государственного университета науки и технологий имени академика М.Ф. Решетнева. Основное направление научных исследований – интеллектуальная робототехника. Имеет более 60 публикаций, в том числе четыре учебных пособия. E-mail: robot-2006@yandex.ru

## *An algorithm for a manipulator motion in an unknown static environment*[*]

*P.K. LOPATIN*

*Reshetnev Siberian State University of Science and Technology, associate professor, 31 Krasnoyarskiy Rabochiy Prospekt, Krasnoyarsk, 640014, Russian Federation. E-mail: robot-2006@yandex.ru*

In this paper we consider the problem of manipulating robot (MR) control in an unknown environment. The MR moving only by allowed states from a start configuration in an environment with unknown static obstacles should either grasp the object by its gripper in an allowed configuration or come to the valid conclusion that the object may not be grasped in any allowed configuration. An algorithm of solving the problem in the continuous configuration space is proposed. We have formulated and proved the theorem stating that following the proposed algorithm the MR will solve the problem in a finite number of steps. It is shown that the solution of the problem is reduced to solving a finite number of PI problems, i.e. planning a path in an environment with known forbidden states with its subsequent execution. The PI procedure is called at path changing points, i.e. the points after which earlier unknown forbidden states are found. It is shown that the number of path changing points will be finite. The MR has a sensor system (SS) which in every path changing point provides information about a neighborhood of the current point. Compared with the previous author's papers where we presumed that the

[*] *Received 01 November 2017.*

neighborhood should have a hypersphere shape, in this paper we presume that the neighborhood may have arbitrary shapes and dimensions. We defined the minimal volumes of reliable information which the SS should provide at every path changing point. Ways of using unreliable information provided by the SS are defined in order to find a solution of the problem in a finite number of steps. The object and obstacles may have arbitrary shapes, dimensions and disposition.

**Keywords:** robot, manipulator, path planning, trajectory, control, sensor system, unknown environment, object grasp, obstacles, reachability, unreachability

DOI: 10.17212/1814-1196-2017-4-33-46

## REFERENCES

1. Canny J. *The complexity of robot motion planning*. Cambridge, Mass., MIT Press, 1988.
2. Choset H., Lynch K.M., Hutchinson S., Kantor G., Burgard W., Kavraki L.E., Thrun S. *Principles of robot motion: theory, algorithms and implementations*. Cambridge, Mass., MIT Press, 2005.
3. LaValle S.M. *Planning algorithms*. Cambridge, Cambridge University Press, 2006. Available at: http://planning.cs.uiuc.edu/ (accessed 10.12.2017).
4. LaValle S.M. Motion planning: the essentials. *IEEE Robotics and Automation Magazine*, 2011, vol. 18 (1), pp. 79–89.
5. Chaabani A., Bellamine M.S., Gasmi M., Motion planning and controlling algorithm for grasping and manipulating moving objects in the presence of obstacles. *International Journal on Soft Computing, Artificial Intelligence and Applications*, 2014, vol. 3, no. ¾, pp. 1–17.
6. Lopatin P. Investigation of a target reachability by a manipulator in an unknown environment. *Proceedings of 2016 IEEE International Conference on Mechatronics and Automation*, August 7–10, 2016, Harbin, China, pp. 37–42.

Для цитирования:
Лопатин П.К. Алгоритм движения манипулятора в неизвестной статической среде / П.К. Лопатин // Научный вестник НГТУ. – 2017. – № 4 (69). – С. 33–46. – doi: 10.17212/1814-1196-2017-4-33-46.

For citation:
Lopatin P.K. Algoritm dvizheniya manipulyatora v neizvestnoi staticheskoi srede [An algorithm for a manipulator motion in an unknown static environment]. *Nauchnyi vestnik Novosibirskogo gosudarstvennogo tekhnicheskogo universiteta – Science bulletin of the Novosibirsk state technical university*, 2017, no. 4 (69), pp. 33–46. doi: 10.17212/1814-1196-2017-4-33-46.