

6. Hoffmeister F. Genetic algorithms and evolution strategies: similarities and differences / Hoffmeister F., Back T. // Parallel Problem Solving from Nature - Proceedings of 1st Workshop, ed. par Scheffel H.P. et Manner R. Dortmund, Germany, 1-3 October 1991. - P.455-469.
  7. Sefioui M. Fast convergence thanks to diversity / Sefioui M., Periaux J., Ganascia J.-G. // Evolutionary Programming V. Proc. of the 5th Annual Conference on Evolutionary Programming. L.J.Fogel, P.J.Angeline and T.Back editors. -MIT Press, 1996. P.92-118
  8. Michalewicz Z. Genetic algorithms + data structures = evolution programs. / Michalewicz Z.- New York, Springer-Verlag, 1992, Artificial Intelligence. – 264 p.
  9. Петухов И.В. Численный расчет двумерных течений в пограничном слое / Петухов И.В. // Численные методы решения дифференциальных и интегральных уравнений и квадратурные формулы. - М.: Наука, - 1964. - С.305-325.
- Д.т.н., с.н.с. Тимченко С.В. – зав.каф. прикладной математики и информатики ТУСУР (г.Томск), tsv@tcde.ru, (3822) 41-33-06*

УДК 519.876.5: 621.865.8

## МОДЕЛИРОВАНИЕ РОБОТОВ-МАНИПУЛЯТОРОВ С УЧЕТОМ ВНЕШНЕЙ СРЕДЫ

А.Н. Горитов

Рассматриваются задачи, связанные с проблемой моделирования роботов-манипуляторов, функционирующих в сложной внешней среде. Движения строятся таким образом, чтобы избежать соударений звеньев манипулятора и не допустить столкновения с объектами внешней среды. Предложенный метод позволяет строить оптимальные траектории перемещения манипулятора по заданным критериям. Для более плавного перемещения манипулятора рассматривается алгоритм сглаживания табличных значений изменения обобщенных переменных манипулятора.

**Ключевые слова:** робототехника, мехатроника, моделирование, планирование траектории

### 1. Введение

При создании программного обеспечения для роботов-манипуляторов одной из задач является обеспечение функционирования роботов в средах с препятствиями. Это вызывает необходимость разработки алгоритмов и программ, позволяющих оперативно контролировать и учитывать ограничения внешней среды, при этом в качестве основных целей ставится задача избегать соударений звеньев роботов между собой и с объектами внешней среды. Моделирование функционирования робота-манипулятора при выполнении таких операций приводит к комплексу взаимосвязанных задач, таких как моделирование робота и его внешней среды, планирование траектории перемещения робота-манипулятора, определение точек соударения звеньев манипулятора, как между собой, так и с объектами внешней среды, синтез законов управления.

В настоящее время разработано большое количество комплексов программ, которые обладают широкими возможностями моделирования механизмов и механических систем. Среди них можно отметить такие комплексы программ, как Euler [1], UM (Уни-

версальный механизм) [2], CATIA [3] и другие. В то же время в этих пакетах не решаются такие задачи, как планирование траектории перемещения схвата манипулятора, синтез законов управления приводами роботов-манипуляторов. Не смотря на то, что эти задачи уже неоднократно ранее рассматривались [4, 5], однако, в силу разнообразия используемых методов при их решении они далеко не исчерпаны. В данной работе предлагаются методы и алгоритмы учета влияний внешней среды при моделировании сложных робототехнических систем.

### 2. Постановка задачи

Пусть задан произвольный робот-манипулятор  $R$  и внешняя среда  $D$ . В начальный момент времени  $t_0$  схват манипулятора находится в точке  $M_0(x_0, y_0, z_0)$ . Необходимо построить план траектории перемещения схвата манипулятора из точки  $M_0$  в целевую точку  $M_k(x_k, y_k, z_k)$  так, чтобы не допустить столкновений звеньев робота, как между собой, так и с объектами внешней среды. На основе построенного плана траектории синтезировать законы управления приводами, обеспечивающие безопасное функциониро-

### РАЗДЕЛ III. МОДЕЛИРОВАНИЕ В ИНФОРМАЦИОННЫХ СИСТЕМАХ

вание робота-манипулятора в заданной внешней среде.

#### Реализация

В процессе моделирования функционирования робота-манипулятора в среде с препятствиями необходимо контролировать возможные соударения звеньев манипулятора – как между собой, так и с объектами внешней среды. Дополнение моделей компонентов геометрическими характеристиками позволяет решить эту задачу в рамках метода компонентных цепей. Простым методом определения соударений является алгоритм поточечной развертки компонентов [6].

Исходными данными для алгоритма являются описание исследуемого объекта и его внешней среды, линейные преобразования координат между смежными компонентами.

Пусть  $K$  – множество компонентов моделируемой цепи,  $P_i$  – геометрическое место точек компонента  $k_i \in K$ . Тогда алгоритм проверки на отсутствие геометрического противоречия будет следующим.

Шаг 1. Положить  $i$  равным 1.

Шаг 2. В соответствии со списковым описанием геометрической цепи выбрать  $k_i$  компонент. Задать  $j$  равным 1.

Шаг 3. Выдать точку  $p_{ij}$ , принадлежащую множеству  $P_i$ . Задать  $h$  равным  $i+1$ .

Шаг 4. С помощью линейных преобразований пересчитать координаты точки  $p_{ij}$  в локальную систему координат  $k_h$ -го компонента.

Шаг 5. Проверить на принадлежность  $p_{ij}$  точки геометрическому месту точек компоненту  $k_h$ . Увеличить номер компонента, выполняющего геометрическую проверку ( $h$  положить равным  $h+1$ ). Если  $h > |K|$ , то перейти на шаг 6. Иначе перейти на шаг 4.

Шаг 6. Увеличить номер точки, принадлежащей геометрическому месту точек  $k_i$  компонента ( $j$  положить равным  $j+1$ ). Если  $j > |P_i|$ , то перейти на шаг 7. Иначе перейти на шаг 3.

Шаг 7. Увеличить номер компонента  $i$  для которого будут выполняться геометрические проверки ( $i$  положить равным  $i+1$ ). Если  $i < |K|$ , то перейти на шаг 2. Иначе перейти на шаг 8.

Шаг 8. Алгоритм проверки на геометрическое противоречие работу закончил.

Результатом работы алгоритма будет сообщение о том, имеет ли место соударение в моделируемом объекте и его внешней среде.

Оценим вычислительную сложность алгоритма. Верхняя оценка числа операций умножения равна  $9 \cdot |K|^2 \cdot |P|_{\max}$ , где  $|K|$  – количество элементов во множестве  $K$ ,  $|P_i|$  – количество элементов во множестве  $P_i$ ,  $|P|_{\max}$  – максимальное значение, выбранное из  $|P_i|$ ,  $i = 1, \dots, |K|$ . Это позволяет сделать вывод, что для проверки на геометрическое противоречие потребуется  $O(|K|^2 \cdot |P|_{\max})$  операций умножения. Такой же функцией оценивается число операций типа сложения и операций сравнения.

Одной из важных задач является задача построения плана траектории исполнительного звена манипулятора. Рассмотрим один из вариантов решения этой задачи.

Пусть рабочее пространство задано в виде трехмерной рабочей сцены с размещенными на ней роботом, трехмерными препятствиями и целевой точкой, в которую необходимо переместить схват манипулятора робота. Требуется определить множество состояний, через которые манипулятор робота должен пройти для того, чтобы переместиться из начального состояния в конечное.

Для решения этой задачи воспользуемся методом поиска в пространстве состояний [7]. Для сокращения числа анализируемых состояний в методе полного перебора введем эвристическую функцию. Будем оценивать каждое состояние с помощью функции.

Для решения этой задачи воспользуемся методом поиска в пространстве состояний [7]. Для сокращения числа анализируемых состояний в методе полного перебора введем эвристическую функцию. Будем оценивать каждое состояние с помощью функции.

Для решения этой задачи воспользуемся методом поиска в пространстве состояний [7]. Для сокращения числа анализируемых состояний в методе полного перебора введем эвристическую функцию. Будем оценивать каждое состояние с помощью функции.

$$f(n) = A(n-1) + \sum_{k=1}^m p_k(n)F_k + R_1(n)F_{\min} + \delta(n), \quad (1)$$

где  $A(n-1)$  – работа, выполняемая при переходе из начального состояния в  $(n-1)$ -е;

$\sum_{k=1}^m p_k(n)F_k$  – работа, выполняемая при переходе из  $(n-1)$ -го в  $n$ -е состояние;

$d_k(n)$  – величина перемещения, выполненное  $k$ -м приводом при переходе из  $(n-1)$ -го в  $n$ -е состояние;  $m$  – количество приводов исполнительного механизма робота;  $F_k$  – сила  $k$ -го привода, затрачиваемая на единицу перемещения;  $R_1(n)$  – расстояние от рабочего инструмента манипулятора, находящегося в  $n$ -м состоянии, до целевой точки по прямой;  $F_{\min}$  – минимальная сила, развиваемая приводами манипулятора на единицу измерения;  $\delta(n) = \infty$ , если  $n$ -е состояние является запрещенным, и равно нулю в противном случае.

Под запрещенным состоянием понимается недопустимое состояние с точки зрения физических и геометрических ограничений.

Оценки состояний, полученные с помощью такой функции, позволяют сократить

число анализируемых состояний за счет того, что из всех возможных состояний для дальнейшего исследования выбираются состояния с минимальными (максимальными) значениями эвристической функции.

Наличие избыточных степеней подвижности робота-манипулятора относительно требуемого для выполнения заданной операции порождает кинематическую и динамическую неопределенность. Наличие этой неопределенности позволяет определить закон движения робота-манипулятора таким образом, чтобы движение было оптимальным в некотором смысле. Например, управление может быть оптимальным в смысле быстродействия или с точки зрения минимальных энергетических затрат на перемещение схвата из одного положения в другое. Эвристическая функция (1) позволяет строить план траектории с минимизацией энергетических затрат.

Для построения оптимальной по времени траектории перемещения манипулятора будем использовать следующую оценочную функцию

$$f(n) = T(n) + R1(n) / V_{min} + \delta(n), \quad (2)$$

где  $T(n)$  – время, затрачиваемое при переходе из начального состояния в  $n$ -е;  $R1(n)$  – расстояние схвата манипулятора робота до целевой точки из  $n$ -го состояния по прямой;  $V_{min}$  – минимальная скорость линейного перемещения, возникающая в кинематических узлах робота;  $\delta(n) = \infty$ , если  $n$ -е состояние является запрещенным, и равно нулю в противном случае.

Применение метода перебора в пространстве состояний позволяет планировать траекторию в условиях отсутствия информации о расположении препятствий во внешней среде. В этом случае используется информация о возможных столкновениях, которая формируется на основе геометрического анализа положения манипулятора и препятствий внешней среды. В процессе построения плана траектории накапливается информация о запрещенных состояниях. Таким образом, алгоритм позволяет строить план траектории в априори неизвестной внешней среде.

Оценочная функция состоит из двух частей. Первая часть включает работу, выполненную при переходе из начального состояния в  $n$ -е состояние. Вторая часть является оценкой работы, которую предстоит выполнить при переходе из  $n$ -го состояния в конечное.

Одна из характеристик оценочной функции связана с понятием целенаправленности [7]. Под целенаправленностью будем пони-

мать отношение длины пройденного пути до цели  $L$  к количеству раскрытых вершин  $P$ . Целенаправленность перебора позволяет узнать, в какой мере перебор идет в направлении цели, а не ведется по нежелательным направлениям. Данное соотношение показывает, насколько дерево состояний, построенное в процессе планирования траектории перемещения исполнительного звена робота-манипулятора вытянуто, а не кустисто.

Модифицируем оценочную функцию (1) и преобразуем ее к виду

$$f(n) = w \left( A(n-1) + \sum_{k=1}^m p_k(n) F_k \right) + (1-w) R1(n) F_{min} + \delta(n), \quad (3)$$

где  $w$  – безразмерный коэффициент, принимающий значение из интервала (0, 1). Коэффициент  $w$  позволяет регулировать вклад каждой из двух частей в оценку  $n$ -й конфигурации робота-манипулятора. При уменьшении  $w$  оценка работы, которую предстоит выполнить для достижения целевой точки будет играть преобладающую роль. Если  $w$  задавать ближе к единице, ведущую роль приобретает оценка выполненной работы. В этом случае основное внимание будет уделено поиску наилучшего решения.

Вычислительные эксперименты, выполненные при различных значениях  $w$  на тестовых задачах, позволяют сделать вывод о резком снижении вычислительных затрат при выборе численного значения коэффициента  $w$  близким к нулю. Результаты экспериментов в виде графика зависимости величины целенаправленности  $L/P$  от коэффициента  $w$  приведены на рисунке 1.

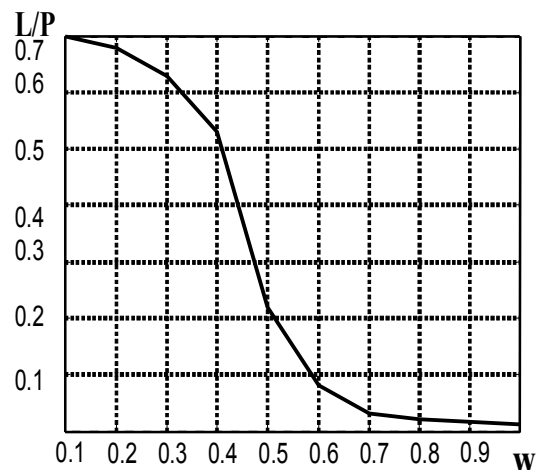


Рисунок 1. График изменения целенаправленности как функции от  $w$

Как видно на графике, при  $w < 0,5$  значительно снижаются вычислительные затраты на построение плана траектории. Повышение

### РАЗДЕЛ III. МОДЕЛИРОВАНИЕ В ИНФОРМАЦИОННЫХ СИСТЕМАХ

эвристической силы алгоритма за счет отказа от построения оптимальной траектории позволяет значительно снизить вычислительные затраты на планирование траектории перемещения исполнительного звена робота. Одновременно это позволяет решать существенно более трудные задачи.

Результатом построения плана траектории будет набор (таблица) дискретных значений обобщенных переменных, которые определяют положение манипулятора в пространстве. Для того, чтобы перейти от дискретного набора значений к непрерывным функциям управления необходимо выполнить их аппроксимацию. Учитывая, что шаг дискретизации, с которым строился план траектории, может быть большим, что в свою очередь может отрицательно отразиться на движении манипулятора, полученный дискретный дискретных значений обобщенных переменных необходимо сгладить.

При сглаживании и аппроксимации необходимо учитывать ограничения по скорости и ускорению. Рассмотрим решение этой задачи.

Пусть дана последовательность из  $n$  узловых точек спланированной (исходной) траектории (4) в 3-х мерном пространстве и сетка, по узлам которой происходило планирование траектории ( $r$  – расстояние между соседними узлами сетки):

$$(x_i, y_i, z_i), \quad i = 0, \dots, (n-1) \quad (4)$$

Необходимо заменить узловые точки (4), на узловые точки (5):

$$(x'_i, y'_i, z'_i), \quad i = 0, \dots, (n-1), \quad (5)$$

таким образом, чтобы первая и последняя точки траектории, после сглаживания остались прежними, т. е. выполнялось условие (6):

$$x'_0 = x_0, \quad y'_0 = y_0, \quad z'_0 = z_0, \quad (6)$$

$$x'_{n-1} = x_{n-1}, \quad y'_{n-1} = y_{n-1}, \quad z'_{n-1} = z_{n-1}.$$

Кроме того, необходимо, чтобы новая (сглаженная) траектория была не только максимально гладкой, но и достаточно близкой к исходной траектории для обхода всех препятствий.

Сглаживание траектории происходит путем её замены на полином порядка  $m$ :

$$x'_i = F_m(i) = \sum_{j=0}^m ax_j i^{m-j}, \quad (7)$$

где коэффициенты полинома  $ax_j$  находятся из системы линейных алгебраических уравнений (СЛАУ):

$$\left\{ \begin{array}{l} \dots\dots\dots \\ \sum_{i=0}^{n-1} [(F_m(i) - x_i) \cdot i^{m-j}] = 0, \quad j = 0 \dots m. \quad (8) \\ \dots\dots\dots \end{array} \right.$$

Здесь и далее формулы для координат  $y$  и  $z$  аналогичные.

Для выполнения сглаживания с учетом условия (6), составим СЛАУ:

$$\left\{ \begin{array}{l} \dots\dots\dots \\ \sum_{i=1}^{n-2} [(F_m(i) - x_i) \cdot i^{m-j}] = 0, \quad j = 0 \dots (m-2); \quad (9) \\ \dots\dots\dots \\ F_m(n-1) - x_{n-1} = 0, \quad (m-1); \\ F_m(0) - x_0 = 0, \quad m. \end{array} \right.$$

СЛАУ (7) в матричной форме:

$$Bx \cdot ax = Cx,$$

где  $Bx$  – матрица СЛАУ,  $Cx$  – вектор свободных членов СЛАУ,  $ax$  – вектор искомых коэффициентов полинома (7).

Максимально допустимое отклонение от исходной траектории не должно превышать половины от длины диагонали кубика:  $0,5\sqrt{3}r$ . Этот критерий можно использовать для автоматического подбора необходимой степени сглаживающего полинома.

#### Алгоритм сглаживания

Вход: последовательность точек в 3-х мерном пространстве, представляющих план траектории  $(x_i, y_i, z_i)$ , размер сетки планирования  $r$ .

Выход: функция  $f(t)$  сглаженной траектории.

1. Берется степень сглаживающего полинома  $m := 1$ .

2. Цикл

2.1. Поиск  $X$ -составляющей траектории: формирование и решение системы уравнений (9).

2.2. Поиск  $Y$ -составляющей траектории: формирование и решение системы уравнений (9).

2.3. Поиск  $Z$ -составляющей траектории: формирование и решение системы уравнений (9).

2.4. Если максимальное отклонение  $< 0,5\sqrt{3}r$ , то переход на пункт 3.

2.5. Увеличение степени сглаживающего полинома  $m := m + 1$ .

2.6. Переход на пункт 2.1.

3. Конец.

Система уравнений (8) строится по формулам:

$$Bx_{j,k} = \begin{cases} \sum_{it=1}^{n-1} it^{2m-j-k}, & j = 0 \dots (m-2), \quad k = 0 \dots m; \\ n^{m-k}, & j = m-1, \quad k = 0 \dots m; \\ 0, & j = m, \quad k = 0 \dots (m-1); \\ 1, & j = m, \quad k = m, \end{cases}$$

$$Cx_j = \begin{cases} \sum_{it=1}^{n-1} x_{it} it^{m-j}, & j = 0 \dots (m-2); \\ x_0, & j = m-1; \\ x_n, & j = m, \end{cases}$$

где  $it$  – номер узла,  $x_{it}$  – значение координаты.

Особенность алгоритма поиска максимального отклонения.

После нахождения всех составляющих траектории оценивается максимальное отклонение сглаженной траектории от исходной траектории, это происходит следующим образом:

1. Сглаженная траектория разбивается на последовательность узлов, равномерно по времени. Количество узлов равно количеству узлов исходной траектории.

2. Ищется максимальное расстояние между соответствующими узлами исходной и сглаженной траектории.

Результат такого алгоритма не может быть заниженным, так как расстояние между соответствующими узлами не меньше расстояния взятого узла исходной траектории до сглаженной траектории. Кроме того, расстояние между двумя точками в пространстве найти значительно легче и быстрее, чем найти расстояние от узла исходной траектории до сглаженной траектории. В результате такой замены ужесточается критерий максимально допустимого отклонения и уменьшается время поиска этой оценки. Ужесточение критерия может привести к выбору более высокой степени сглаживающего полинома и к сближению сглаженной траектории к исходной траектории.

Оценим вычислительную сложность приведенного алгоритма. Формирование СЛАУ потребует примерно:

$$\sim (n-1) \left[ m^3 + \frac{m^2}{2} - \frac{3m}{2} \right]$$

операций умножения. Учитывая, что при сглаживании используются обычно полиномы не высокой степени, то вычислительные затраты этого этапа есть  $O(n)$ . Решения СЛАУ потребует примерно:

$$\sim \frac{m^3}{2} - \frac{m^2}{2} - m + \frac{3}{2}$$

операций умножения. Вычислительная сложность этого этапа есть  $O(m^3)$ .

### Заключение

В статье рассмотрен комплекс задач, обеспечивающих функционирование роботов-манипуляторов в произвольной среде с препятствиями. Предложены алгоритмы построения плана траектории и сглаживания полученных табличных значений изменения обобщенных переменных робота-манипулятора. Приведенные оценки вычислительной эффективности, позволяют сделать заключение о возможности практического использования предложенных алгоритмов.

### СПИСОК ЛИТЕРАТУРЫ

1. Бойков, В. Программный комплекс автоматизированного динамического анализа многокомпонентных механических систем EULER/ В. Бойков // САПР и графика. – 2000. – № 8. – С. 17 – 20.
2. Ковалев Р.В., Даниленко Д.В. Введение в моделирование динамики механических систем/ Р.В.Ковалев, Д.В.Даниленко// САПР и Графика, № 4, 2008. – С. 26-31.
3. Голдовский, П. Проектирование и моделирование промышленных роботов в системе САТИА/П.Голдовский// САПР и графика. – 2005. – № 2. – С. 82 – 84.
4. Булгаков, А.Г. Промышленные роботы. Кинематика, динамика, контроль и управление/ А.Г.Булгаков, В.А.Воробьев. – М.: СОЛОН-ПРЕСС, 2007. – 488 с.
5. Корендясев, А.И. Теоретические основы робототехники/ А.И. Корендясев, Б.Л.Саламандра, Л.И.Тывес- В 2 кн. – М.: Наука, 2006. – 384 с.
6. Горитов, А.Н., Дмитриев В.М. Анализ управляемых механических систем с геометрической интерпретацией рабочего пространства. Геометрический и кинематический анализ./ А.Н. Горитов, В.М.Дмитриев - Томск: Изд-во Томского государственного ун-та систем управления и радиоэлектроники, 1998. - 120 с/
7. Нильсон, Н. Искусственный интеллект. Методы поиска решений/ Н. Нильсон. – М.: Мир, 1973. – 272 с.

*Д.т.н., с.н.с. Горитов А.Н. – профессор кафедры АСУ ТУСУР (г.Томск), ang@asu.tusur.ru, (3822) 70-15-36*