

УДК 681.5

АВТОМАТИЗАЦИЯ ПРОГРАММИРОВАНИЯ КОНТУРНОГО ДВИЖЕНИЯ РОБОТА ПРИ ЛАЗЕРНОЙ РЕЗКЕ

А.П. ПАШКЕВИЧ, О.А. ЧУМАКОВ

Белорусский государственный университет информатики и радиоэлектроники
П. Бровки, 6, Минск, 220013, Беларусь

Поступила в редакцию 9 января 2003

Предложены методы, позволяющие преобразовать трехмерную модель контура резки, полученную из САПР изделия, в программу управления роботом. При этом учитываются ограничения промышленных систем управления, обусловленные особенностями алгоритмов планирования движений манипулятора, а также оптимально используется кинематическая избыточность шестистепенного робота для операций резки. В результате генерируется плавная траектория движения инструмента манипулятора, которая может быть воспроизведена в реальном масштабе времени с заданной точностью.

Ключевые слова: робот, off-line программирование, автоматизация проектирования.

Введение

В настоящее время в промышленности все шире применяются робототехнологические комплексы (РТК) лазерной и плазменной резки, обеспечивающие прецизионную обработку деталей и высокую производительность в мелкосерийном производстве. Однако их широкое применение сдерживается сложностью программирования, которое может выполняться в ручном или автоматическом (автономном) режимах. Наиболее перспективным является второй метод, не требующий полной остановки оборудования и его исключения из производственного процесса [1].

Существующие системы автономного программирования [2] ориентированы, как правило, на технологию дуговой либо точечной контактной сварки и не могут быть использованы при роботизации резки трехмерных объектов сложной формы. Поэтому в данной работе рассматривается проблема автоматизации программирования РТК резки, в которых традиционные манипуляторы обладают кинематической избыточностью, обусловленной возможностью поворота инструмента вокруг режущего луча. Эта избыточность существенно усложняет программирование, но ее можно использовать для оптимизации движений робота. Другие проблемы, возникающие при программировании таких РТК, связаны с особенностями алгоритмов планирования движений, реализованных в системах управления промышленных роботов. Они проявляются в снижении скорости при малом шаге квантования контура, а также в отклонении траектории в узловых точках.

Предлагаемый подход учитывает указанные особенности и включает в себя два этапа. На первом из них производится преобразование исходного контура в оптимальный набор линейных и круговых сегментов, которые могут быть реализованы системой управления. На втором этапе осуществляется оптимизация движений робота и формирование соответствующей управляющей

программы. В результате генерируется плавная траектория, которая может быть воспроизведена в реальном масштабе времени с заданной точностью.

Сегментация траектории

Предположим, что исходные данные о траектории резки представлены в виде “обобщенной полилинии” (*augmented line*), созданной на основе каркасной или поверхностной модели изделия, которая описывается с помощью массива вертексов

$$\Omega = \{\omega_1, \omega_2, \omega_3, \dots, \omega_k\}, \quad (1)$$

где $\omega_i = (p_i, a_i)$, $p_i = (p_{xi}, p_{yi}, p_{zi})$ — вектор положения i -й точки, $a_i = (a_{xi}, a_{yi}, a_{zi})$ — соответствующий вектор ориентации инструмента, причем расстояние между соседними точками удовлетворяет ограничению $\|p_i - p_{i-1}\| \leq \Delta S_{min}$. Для отработки этой траектории роботом требуется построить новую последовательность вертексов Ω' , состоящую из минимального количества узлов m и удовлетворяющую краевым условиям $\omega'_0 = \omega_0$; $\omega'_m = \omega_k$, а также двум ограничениям по точности, обусловленным особенностями алгоритмов управления движениями роботов, реализованных в промышленных контроллерах [4]:

$$\max_{\mu \in [0,1]} \rho(\mu p'_{i-1} + (1-\mu)p'_i, \Omega) \leq \delta; \quad \forall i = 1 : m, \quad (2)$$

$$\frac{S_T}{2} \cdot \cos \frac{\alpha_i}{2} \leq \delta; \quad \forall i = 1 : m-1, \quad (3)$$

где δ — требуемая точность контурного движения; $\rho(p, \Omega) = \min_i \|p_i - p\|$ — расстояние от точки p до траектории Ω ; α_i — угол изгиба траектории Ω' в узле ω'_i : $\cos \alpha_i = (p'_i - p'_{i-1}) \cdot (p'_{i+1} - p'_i) / |p'_i - p'_{i-1}| \cdot |p'_{i+1} - p'_i|$; S_T — путь торможения при заданной скорости резки; $\Delta S_{min} = vT_0$ — минимальный интервал между вертексами, зависящий от скорости резки v и периода квантования системы управления T_0 .

Для сегментации исходной траектории (1) в соответствии с ограничениями (2)–(3) разработан эвристический алгоритм, приведенный ниже. Он обеспечивает последовательное формирование линейных и круговых сегментов максимальной длины с фиксированной начальной точкой. При этом более высоким приоритетом обладают линейные сегменты, требующие меньших вычислительных затрат при реализации в системе управления роботом.

Входные данные:

- массив равноудаленных точек, описывающих исходную траекторию в виде “обобщенной полилинии” $\{p_i, a_i\}$, $i = 1 : k$;
- допустимое отклонение синтезируемой траектории δ_{max} ;
- предельный радиус дугового сегмента R_{max} ;

Выходные данные:

- массив точек $\{p'_i, a'_i\}$, $i = 1 : k'$, определяющий концы сегментов;
- массив $\{^m p'_i\}$, $i = 1 : k'$, определяющий дополнительные точки дуговых сегментов.

Основные шаги:

Инициализация: $p'_1 = p_1$, $^m p'_1 = p_1$; флаг окончания массива $endofpath=0$; указатели границ сегмента $i=1$, $i'=1$.

While(endofpath=0) – выполняем цикл до тех пор, пока не будут обработаны все точки исходного массива.

 Полагаем **done=0** (флаг окончания сегмента)

Аппроксимация дуговым сегментом

$j = i + 2$

While($done=0 \ \& \ j \leq n$) - выполняем цикл до тех пор, пока дуга с точками i, m, j отклоняется от траектории не более, чем на δ_{max} и имеет радиус не более R_{max} .

- Полагаем $m = \text{int}((i + j) / 2)$; $R = 0$,

- **If** $\|(\mathbf{p}_j - \mathbf{p}_m) * (\mathbf{p}_m - \mathbf{p}_i)\| < \varepsilon$

then полагаем $done=1$, т.е. считаем, что точки лежат на одной прямой,
else находим центр \mathbf{p}_0 и радиус R окружности, проходящей через точки

$\mathbf{p}_i, \mathbf{p}_m, \mathbf{p}_j$. Для всех точек \mathbf{p}_k , при $k=i+1$ до $j-1$ вычисляем расстояние до окружности $\rho_k = \left| \|\mathbf{p}_k - \mathbf{p}_0\| - R \right|$ и сравниваем с δ_{max} ; если $\rho_k > \delta_{max}$ или $R > R_{max}$, полагаем $done=1$.

- **If** $done=0$, **then** $j=j+1$; т.е. увеличиваем длину сегмента,

- **If** $j \geq n$ **then** $endofpath=1$; т.е. найден конец траектории,

End of While

If $j > i + 2$ **then** полагаем $j = j - 1$; $i' = i' + 1$ и запоминаем найденный сегмент: $p'_{i'} = p_j$; ${}^m p'_{i'} = p_m$; $i = j$

else полагаем $done=0$.

Аппроксимация линейным сегментом

$j = i + 1$

While($done=0 \ \& \ j \leq n$) — выполняем цикл до тех пор, пока отрезок с точками i, j отклоняется от траектории не более, чем на δ_{max}

- Строим уравнение прямой, проходящей через точки $\mathbf{p}_i, \mathbf{p}_j$.

- Для всех точек \mathbf{p}_k при $k=i+1$ до $j-1$ вычисляем расстояние до прямой ρ_k и сравниваем с δ_{max} ; если $\rho_k > \delta_{max}$, полагаем $done=2$.

- **If** $done=0$ **then** $j=j+1$; т.е. увеличиваем длину сегмента,

- **If** $j > n$ **then** $done=2$; т.е. найден конец траектории,

End of While

If $done=2$ **then** полагаем $j = j - 1$; $i' = i' + 1$ и запоминаем найденный сегмент: $p'_{i'} = p_j$; $i = j$.

End of While

Оптимизация движений робота

Современные РТК лазерной и плазменной резки строятся, как правило, на базе манипуляторов с шестью степенями подвижности, которые применительно к данной операции обладают кинематической избыточностью. Это обусловлено тем, что допускается произвольное вращение инструмента вокруг режущего луча. Поэтому при разработке управляющих программ для робота целесообразно использовать избыточность для оптимизации движений манипулятора в пространстве обобщенных координат.

Предположим, что расположение систем координат робота "Base" и изделия "Workpiece" относительно глобальной системы координат "World" описывается 4×4 -матрицами однородных преобразований 0T_b и 0T_w соответственно (рис. 1). Тогда кинематические свойства РТК резки могут быть описаны матричным уравнением

$${}^0T_i(q) \equiv {}^0T_b \cdot T_1(q_1) T_2(q_2) \dots T_5(q_5) T_6(q_6) \cdot T_i, \quad (4)$$

где T_i — матрица, описывающая положение и ориентацию системы координат инструмента "Tool" по отношению к системе координат крепежного фланца "Flange"; $T_i(q_i)$ — матрицы, описывающие геометрическую структуру манипулятора в зависимости от значений обобщенных координат звеньев q_i . Последние из указанных матриц существенно зависят от кинематической структуры робота, и для роботов антропоморфного типа их произведения могут быть представлены в виде [3]

$${}^0T_i = \begin{bmatrix} C_1 C_{23} & -S_1 & C_1 S_{23} & a_2 C_1 C_2 + a_3 C_1 C_{23} - d_2 S_1 \\ S_1 C_{23} & C_1 & S_1 S_{23} & a_2 S_1 C_2 + a_3 S_1 C_{23} + d_2 C_1 \\ -S_{23} & 0 & C_{23} & -a_2 S_2 - a_3 S_{23} \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} C_4 C_5 C_6 - S_4 S_6 & -C_4 C_5 S_6 - S_4 C_6 & C_4 S_5 & d_6 C_4 S_5 \\ S_4 C_5 C_6 + C_4 S_6 & -S_4 C_5 S_6 + C_4 C_6 & S_4 S_5 & d_6 S_4 S_5 \\ -S_5 C_6 & S_5 S_6 & C_5 & d_6 C_5 + a_4 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

где $C_i = \cos q_i$; $S_i = \sin q_i$; $C_{ij} \equiv \cos(q_i + q_j)$; $S_{ij} = \sin(q_i + q_j)$; a_i , d_i — параметры Денавита-Хартенберга, имеющие следующий смысл: d_2 - расстояние между осью первого сустава и плоскостью вращения третьего звена; a_2 , d_4 — длины второго и третьего суставов соответственно; a_3 — расстояние между осью вращения третьего сустава и осью симметрии второго звена; d_6 — расстояние от центра кисти до плоскости фланца. Для каждой узловой точки траектории ω'_i определим также систему координат и обозначим ее "Node $_i$ " (рис. 1). Начало координат этой системы разместим в точке p'_i , ось X направим вдоль траектории резки, ось Z — в направлении режущего луча a'_i , а ось Y — так, чтобы X, Y, Z образовывали правую тройку векторов.

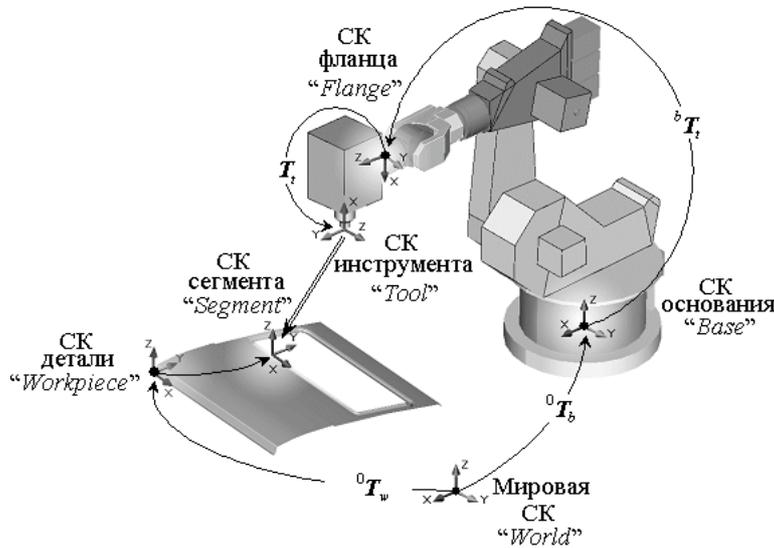


Рис. 1. Геометрическая модель РТК

Соответствующую матрицу однородных преобразований обозначим ${}^0T_{s_i}$. Тогда манипуляционную задачу робота можно представить как совмещение координатных осей фреймов ${}^0T_{s_i}$ и фреймов ${}^0T_i(q)$ таким образом, что ось X_i совпадает с осью Z_s , а взаимное расположение оставшихся осей определяется с точностью до произвольного скалярного параметра γ , описывающего поворот рассматриваемых фреймов вокруг оси резки:

$${}^0T_{s_i} R_z(\gamma) = {}^0T_i(q) \cdot H, \quad (5)$$

где постоянная матрица $H = R_x(\pi) \cdot R_y(\pi/2)$ учитывает порядок совмещения осей, а R_x , R_y , R_z — операторы вращения вокруг X, Y, Z соответственно. В результате, применяя соответствующий алгоритм решения обратной задачи кинематики

$$Q_i(\gamma) = \text{InvKin} \left[{}^0T_{s_i} R_z(\gamma) H^{-1} \right], \quad (6)$$

исходная совокупность сегментов траектории Ω' преобразуется в однопараметрическое семейство обобщенных координат $Q_i(\gamma)$. На практике целесообразно ограничиться дискретными значениями параметра $\gamma = 2\pi j/k$; $j = 0:k$ и множество $\{Q_i(\gamma); \gamma \in [0, \pi]; i = 1:m\}$ описать двумерной мат-

рицей, столбцы которой соответствуют одной и той же физической точке траектории резки, а элементами являются векторы обобщенных координат \mathbf{q} либо соответствующие им локации инструмента \mathbf{L} . При этом для каждого элемента этой матрицы целесообразно проверить существование решения обратной задачи кинематики и отсутствие столкновений робота и режущего инструмента с другими элементами РТК. Применяя указанные операции ко всем узлам траектории Ω' , получаем, что множество допустимых локаций инструмента и соответствующих обобщенных координат представляется в виде "разряженной" двумерной матрицы $\{\mathbf{L}_{ij}, \mathbf{q}_{ij}\}$, каждый элемент которой представляет собой совокупность фрейма \mathbf{L}_{ij} и шестимерного вектора \mathbf{q}_{ij} .

Набор таких матриц целесообразно представить в виде многослойного направленного графа с вершинами $V = \{\mathbf{L}_{ij}\}$ и ребрами $E = \{(\mathbf{L}_{ij}, \mathbf{L}_{kl}); i = k - 1; \forall i, j, k, l\}$, где каждый слой соответствует определенному индексу конфигурации M (рис. 2). Тогда для заданного индекса конфигурации функция $\gamma(t)$ определяет шесть траекторий суставов $q_k(t), k=1\dots 6$, каждая из которых может быть оценена скалярным критерием

$$J_s^{(k)}[\mathbf{q}(t)] = \int_0^T |\dot{q}_k(t)| dt, \quad (7)$$

который характеризует "объем движения" соответствующего сустава.

В результате рассматриваемая задача планирования движения робота сводится к следующей оптимизационной задаче на графе: для заданных множеств вершин V и ребер E , найти "наилучший" путь длины n $\Pi(\gamma_0, \dots, \gamma_n) = \langle \mathbf{L}_{0j_1} \rightarrow \mathbf{L}_{1j_2} \rightarrow \dots \mathbf{L}_{nj_n} \rangle$ с начальным состоянием $V_0 \in \{\mathbf{L}_{0j}\}$ и конечным состоянием $V_n \in \{\mathbf{L}_{nj}\}$, который обеспечивает парето-оптимальность векторного критерия качества (7).

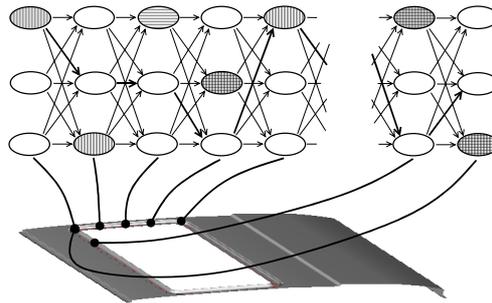


Рис. 2. Графовая модель пространства поиска

Для упрощения описания алгоритмов обобщенные координаты, соответствующие положению \mathbf{L}_{ij} , обозначим как $q_k(i, j)$, а траектории, соответствующие вектору решения Γ , — как $q_k(i, j_{\gamma_i})$. Используя эти обозначения, задачу минимизации аддитивного критерия (7) для каждой обобщенной координаты можно представить как

$$J_s^{(k)}(\Gamma) = \sum_i |q_k(i, j_{\gamma_i}) - q_k(i-1, j_{\gamma_{i-1}})| \rightarrow \min_{\Gamma}$$

и решить методом динамического программирования. Для доказательства предположим, что на p -ом шаге были найдены все оптимальные последовательности $\Gamma^o(p, \chi) = \langle \gamma_0, \dots, \gamma_{p-1}, \chi \rangle$ с последним элементом $\chi \in \Gamma_p$ и соответствующими показателями качества $F_p(\gamma)$. Тогда для следующего шага оптимальная последовательность $\Gamma^o(p+1, \gamma) = \langle \gamma_0, \dots, \gamma_{p-1}, \chi, \gamma \rangle$ с последним элементом $\gamma \in \Gamma_{p+1}$ может быть найдена из следующей рекурсии:

$$F_{p+1}(\gamma) = \min_{\chi \in \Gamma_p} \{F_p(\chi) + |q_k(p+1, j_\gamma) - q_k(p, j_\chi)|\},$$

которая описывает алгоритм оптимизации каждой скалярной компоненты рассматриваемого векторного критерия. Для построения парето-оптимальных решений используется аналогичный рекурсивный алгоритм, в котором каждой компоненте критерия (7) назначаются весовые коэффициенты, изменяющиеся в процессе оптимизации.

Заключение

Разработанные методы позволяют преобразовать трехмерную модель контура резки, полученную из САПР изделия, в упорядоченную последовательность локаций, на основе которой генерируется программа управления роботом. При этом учитываются ограничения промышленных систем управления, обусловленные особенностями алгоритмов планирования движений манипулятора (искажения в узловых точках, снижение скорости при малых перемещениях, сегментация траектории на линейные и круговые участки), а также оптимально используется кинематическая избыточность шестистепенного робота для операций резки. Искомая траектория синтезируется в два этапа. На первом из них производится сегментация исходного контура, т.е. преобразование его в набор линейных и круговых сегментов максимальной длины. На втором этапе пространство поиска решений преобразуется в направленный граф и исходная задача формулируется в терминах теории комбинаторной оптимизации. Затем осуществляется линейная свертка векторного критерия качества, и методом динамического программирования формируется множество парето-оптимальных решений, соответствующих различным значениям весовых коэффициентов. В результате генерируется плавная траектория движения инструмента манипулятора, которая может быть воспроизведена в реальном масштабе времени с заданной точностью. Предложенные алгоритмы реализованы в подсистеме Laser САПР ROBOMAX и были использованы при решении реальных производственных задач по автономному программированию РТК резки для АМО ЗИЛ (г. Москва). Как следует из опыта эксплуатации, они позволяют эффективно синтезировать и оптимизировать программы роботизированной резки трехмерных деталей сложной формы и достичь существенного повышения экономической эффективности при мелкосерийном производстве.

COMPUTER AIDED PROGRAMMING OF CONTOUR MOTION FOR A LASER CUTTING ROBOT

A.P PASHKEVICH, O.A. CHUMAKOV

Abstract

The paper proposes a technique for transforming a CAD-model of a cutting contour into a robot control program. The design procedure takes into account constraints imposed by industrial robotic controls and their path planing algorithms. The kinematic redundancy of a robotic manipulator is used to optimise trajectory in the joint coordinate space. The technique yields a smooth path for a cutting tool, which can be implemented in real time with desired accuracy.

Литература

1. *Geiger. M., Otto. A.* Laser in der Elektronik produktion & Feinwerktechnik, Tagungsband des 3. Erlanger Seminars LEF 2000, Bamberg: Meisenbach, 2000.
2. *Пашкевич А.П.* // Справочник. Инженерный журнал. М., 1997. № 5, С. 27–33.
3. *Pashkevich A., Dolgui A., Chumakov O.* Optimal Control of Robotic Manipulator for Laser Cutting Applications. 15th Triennial World Congress of the International Federation of Automatic Control. Barcelona, SPAIN, 21th–26th July, 2002. Book of abstracts P. 273.
4. *Чумаков О.А.* // Тез. докл. II Междунар. науч.-практ. конф. "Моделирование. Теория, методы и средства". Новочеркасск, 2002. С. 45–49.
5. *Sciavicco L., Siciliano B.* Modelling and control of robot manipulators. Springer, New York, 2000.