

AUTOMATIZACIÓN DEL CORTE TRIDIMENSIONAL DE PIEZAS MEDIANTE UN ROBOT MANIPULADOR A PARTIR DE UN DISEÑO CAD

M. Gómez Langley

Dpto. Ing. Sistemas y Automática, Universidad de Sevilla, email: mglangley@supercable.es

M. Vargas y F. R. Rubio

Dpto. Ing. Sistemas y Automática, Universidad de Sevilla, email: {vargas,rubio}@cartuja.us.es

Resumen

En este artículo se expone el desarrollo de una aplicación informática que permite la automatización del corte de piezas en tres dimensiones mediante un brazo robótico industrial a partir de un diseño de las mismas realizado mediante una herramienta CAD/CAM. Partiendo de un programa en lenguaje simbólico y de la posición inicial de la herramienta de corte, el software desarrollado permite simular la trayectoria de corte y determinar si todos los puntos de la misma son alcanzables, en cuyo caso se transmiten al controlador del manipulador en un formato comprensible por éste y se ejecuta la trayectoria.

Palabras Clave: Automatización, CAD/CAM, programación automática en máquinas de control numérico, simulación.

1 INTRODUCCIÓN

Este proyecto está encaminado a automatizar el corte de piezas de dimensiones relativamente reducidas (aproximadamente de 0.5m^2 de superficie) del fuselaje de aviones. Actualmente, este proceso es realizado manualmente con una fresa neumática. Nuestro objetivo es realizarlo de forma automática mediante la utilización de un robot manipulador Staübli de la serie RX-90 [7] de seis grados de libertad junto con un sistema de corte adecuado al tipo de material del que está compuesta la pieza de partida o stock.

Fundamentalmente, existen dos razones por las que se ha optado por la utilización de robots manipuladores industriales en lugar de máquinas de control numérico (CN) específicas: un menor coste y una mayor flexibilidad. Por una parte, para poder realizar el corte de una pieza de unas dimensiones determinadas, el coste de un equipo compuesto por un manipulador industrial junto con el sistema de corte conveniente al tipo de material del que está

formado es sustancialmente menor que el de una máquina de CN del tamaño adecuado. Esta diferencia se acentúa enormemente al aumentar el tamaño de los diseños y al aumentar el número de grados de libertad necesarios. Por otra parte, el uso de un manipulador de seis grados de libertad nos aporta una flexibilidad mucho mayor que una máquina de CN en tanto en cuanto existen, generalmente, muchas posibilidades a la hora de atacar la pieza para realizar el corte.

En la programación manual de mecanizados de piezas por CN, el llamado “programa pieza” se escribe únicamente por medio de razonamientos y cálculos que realiza el programador. Estos cálculos no suelen presentar excesivas dificultades para una persona especializada. Sin embargo, cuando las piezas son complejas, los cálculos aumentan y las posibilidades de error por parte del programador aumentan consecuentemente. La *programación automática* consiste en utilizar la ayuda del ordenador para la programación de un CN.

Frente a la programación manual, la programación automática presenta las siguientes diferencias: el operador no ha de preocuparse de realizar cálculos geométricos; no tiene que conocer los códigos de funciones de la máquina concreta, pero sí es necesario conocer el lenguaje simbólico APT (Automatically Programmed Tooling) [5] o similar; finalmente, la estructura de programación es distinta. Estas diferencias implican tres ventajas: un menor tiempo de preparación del programa, menores errores de cálculo y transcripciones, y una sustancial reducción de los costes de preparación y programación.

Por lo tanto, para ciertas aplicaciones como la que nos ocupa, consistente en fresados tridimensionales para generar superficies curvas de cierto espesor y contornos poligonales, es imposible prescindir de la programación automática.

Si utilizamos la programación automática, el papel del programador se limita a las siguientes operaciones:

- Definir la geometría de la pieza objetivo en tres dimensiones. Esta operación puede ser realizada mediante un sistema de *CAD/CAM* adecuado.
- Especificar la secuencia de operaciones a ejecutar en la generación de la trayectoria que debe seguir el robot. Tanto esta secuencia como la definición de las características de la herramienta (diámetro de la fresa...) se pueden fijar mediante el módulo de CN de la herramienta *CAD/CAM*, obteniendo como resultado final el correspondiente programa en lenguaje simbólico como APT o similar.

Por su parte, el computador lleva a cabo las operaciones que es capaz de realizar a mucha más velocidad y con una probabilidad mínima de cometer errores, como son: interpretar las órdenes recibidas en el módulo de CN necesarias para la generación del programa pieza, detectando y corrigiendo posibles errores; realizar los cálculos aritméticos necesarios para construir las cotas características del mecanizado teniendo en cuenta las correcciones derivadas de la geometría de la herramienta; procesar y adaptar el resultado al lenguaje máquina característico del CN utilizado. Este proceso se realiza mediante el correspondiente postprocesador incorporado en el módulo de CN.

En cierto sentido, podríamos decir que la aplicación informática que se ha desarrollado no es más que una especie de postprocesador para el manipulador Staübli RX-90 en concreto. Este razonamiento se debe a que, partiendo del programa en lenguaje simbólico que contiene la información acerca de la localización y la orientación del extremo de la herramienta para todos los puntos que componen la trayectoria de corte, el software creado debe ejecutar las siguientes operaciones para conseguir la pieza objetivo deseada a partir de la pieza matriz o stock:

- Adquirir los puntos iniciales de la pieza matriz en el espacio real para poder calcular la posición y orientación iniciales donde debe situarse el extremo de la herramienta.
- Componer los puntos de la trayectoria contenidos en el programa en lenguaje simbólico respecto a dicho punto.
- Simular la ejecución de la trayectoria para determinar si todos los puntos de la misma serán alcanzables por el robot.
- En caso afirmativo, transmitir la secuencia de puntos que componen la trayectoria al controlador en un formato comprensible por éste y ejecutarla. Si, por el contrario, alguno de los puntos de la trayectoria no es alcanzable, mostrar un aviso para que se cambie o bien la

localización inicial del extremo de la herramienta o bien la trayectoria de corte.

La estructura del artículo es la siguiente. En la sección 2 se revisan brevemente algunos conceptos básicos referentes a aspectos geométricos a tener en cuenta. A continuación, en la sección 3 se detalla extensamente la funcionalidad de la aplicación informática desarrollada. En la sección 4, por su parte, se lleva a cabo la descripción de la implementación del sistema. Finalmente, en la sección 5 se presentan las conclusiones y los posibles desarrollos futuros.

2 CONSIDERACIONES GEOMÉTRICAS

Para representar la posición y orientación del extremo de la herramienta de corte en el espacio se han empleado distintas representaciones según conviniese:

- Coordenadas articulares: consiste en un vector de seis componentes que refleja el valor que toman las seis variables articulares.
- Coordenadas cartesianas: consiste en un vector de seis componentes en el que las tres primeras representan la posición en el espacio en coordenadas cartesianas XYZ y las tres últimas nos informan sobre la orientación. Hemos empleado dos tipos de representaciones de la orientación:
 - Mediante los ángulos ZYZ de Euler [1].
 - Mediante un vector unitario tridimensional que define la posición del extremo de la herramienta respecto a un sistema de referencia fijado en la muñeca del manipulador.
- Matriz de transformación homogénea: consiste en una matriz 4x4 compuesta por una matriz de rotación y un vector de translación que representan, respectivamente, la orientación y la posición.

Las coordenadas articulares son empleadas internamente en el programa para chequear la alcanzabilidad. Como ya hemos dicho, consisten en un vector de seis componentes de la forma:

$$\theta = [\theta_1 \ \theta_2 \ \theta_3 \ \theta_4 \ \theta_5 \ \theta_6]^T \quad (1)$$

donde θ_i con $i=\{1,2, \dots, 6\}$ son las variables correspondientes a cada una de las seis articulaciones de las que consta el manipulador empleado.

Los dos tipos de coordenadas cartesianas empleadas se utilizan como formatos de entrada y de salida de información en la aplicación. Por una parte, la representación formada por las coordenadas XYZ y

los ángulos ZYZ de Euler, que nosotros llamaremos coordenadas *XYZeulerZYZ*, son utilizadas por el controlador del robot. Por otra, la representación compuesta por las coordenadas XYZ y un vector, que hemos denominado coordenadas *XYZvector* para diferenciarlas de las anteriores, es empleada en el programa en lenguaje simbólico APT.

Como ya hemos dicho, las coordenadas *XYZeulerZYZ* emplean los ángulos ZYZ de Euler para representar la orientación. Estos tres ángulos denominados α , β y γ definen la orientación mediante tres giros consecutivos en los tres ejes coordenados solidarios al cuerpo: primero respecto al eje \hat{Z} un ángulo α , luego un ángulo β respecto al nuevo eje \hat{Y} , y finalmente respecto al nuevo eje \hat{Z} un ángulo γ .

$$P_{\alpha\beta\gamma} = [X \ Y \ Z \ \alpha \ \beta \ \gamma]^T \quad (2)$$

Por su parte, las coordenadas *XYZvector* representan la orientación mediante un vector unitario de tres componentes que definen la posición del extremo de la herramienta de corte respecto a un sistema de referencia fijo situado en la muñeca del manipulador.

$$P_{ijk} = [X \ Y \ Z \ i \ j \ k]^T \quad (3)$$

donde i , j y k son las coordenadas cartesianas del extremo de la herramienta respecto al sistema de referencia fijo en la muñeca del manipulador, como puede observarse en la figura 1.

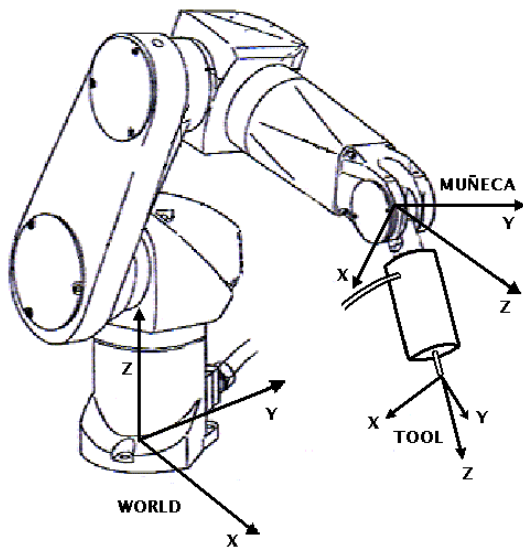


Figura 1: Representación de la posición y orientación mediante el formato *XYZvector*.

Finalmente, el último formato empleado son las matrices de transformación homogénea, que se utilizan internamente en el programa para realizar cálculos necesarios. Estas matrices son de la forma:

$$M = \begin{bmatrix} R & T \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4)$$

donde R es una matriz 3x3 que representa la orientación respecto al sistema de referencia fijado en la base del manipulador, y $T=[X \ Y \ Z]^T$ es un vector 3x1 que representa una traslación respecto al origen de coordenadas del mismo.

Obviamente, al emplear diversas representaciones ha sido necesario incorporar también los algoritmos necesarios para convertir de un formato a otro. En el siguiente apartado se volverá sobre este tema a medida que sea necesario.

3 APLICACIÓN INFORMÁTICA

La aplicación informática desarrollada tiene como objetivo automatizar el proceso de corte de piezas ligeras de aviación. Para ello, su funcionalidad básica debe ser leer el fichero de salida del software CAD/CAM que contiene la información de la trayectoria de corte a seguir, simularla y, si todos los puntos de la misma son alcanzables, transmitírsela al controlador del robot en un formato comprensible por éste para que pueda ejecutarla. A modo de resumen, en la figura 2 se presenta un diagrama de flujo de las operaciones realizadas sobre los datos.

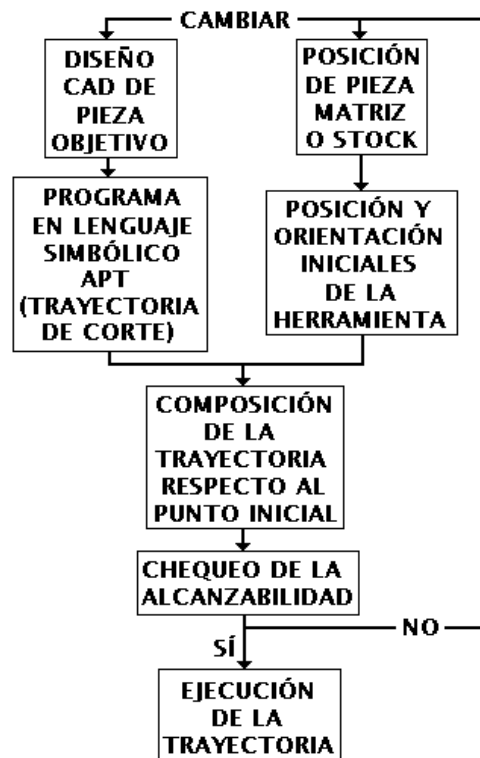


Figura 2: Diagrama de flujo de datos y procedimiento.

Otras características interesantes de las que dispone el software creado son la existencia de un emulador de terminal simple que permite que el programa sirva como interfaz entre el manipulador y el usuario, y de una pantalla de simulación en la que puede observarse un modelo virtual en tres dimensiones del robot.

Para que todas las funciones incluidas en el programa se ejecuten correctamente es imprescindible que exista una correcta comunicación entre éste y el armario de control del robot. Para resolver el problema de la comunicación entre ambos a través del puerto serie se optó por dotar al programa de una estructura multihilo. Es decir, además del hilo principal del programa, que se encarga de manejar los eventos asociados a la interfaz gráfica, se ha programado un proceso paralelo que se encarga de gestionar de forma transparente al usuario la lectura y la escritura de datos por el puerto serie.

En los siguientes subapartados se comentarán las distintas operaciones implementadas en el programa: obtención de la posición y orientación iniciales de la herramienta de corte, composición de la trayectoria respecto ese punto, confirmación de que todos los puntos son alcanzables y, si procede, ejecución de la trayectoria.

3.1 CÁLCULO DE LA POSICIÓN Y LA ORIENTACIÓN INICIALES DEL EXTREMO DE LA HERRAMIENTA

A partir del programa en lenguaje simbólico que se obtiene como resultado del diseño de la pieza y de la definición de la trayectoria de corte mediante el software CAD/CAM, disponemos de la información necesaria acerca de la representación geométrica de los puntos que componen la trayectoria. Sin embargo, todos estos puntos están referidos a un sistema de referencia propio del diseño, por lo que no son útiles tal cual, sino que es necesario referirlos al sistema de referencia WORLD asociado al robot. Para ello, estos puntos deben componerse respecto al punto inicial donde estará colocada la herramienta antes de empezar a realizar el corte, que será un punto perteneciente al espacio alcanzable por el manipulador.

El proceso seguido para calcular la posición y la orientación iniciales de la herramienta se basa en la técnica conocida como *enseñanza por demostración* (“*teach by showing*”). Como puede apreciarse en la figura 3, sobre la pieza matriz o stock definimos tres puntos *A*, *B* y *C* fácilmente identificables (como, por ejemplo, tres de los vértices de una de las caras) y los memorizamos posicionando el extremo de la herramienta sobre ellos.

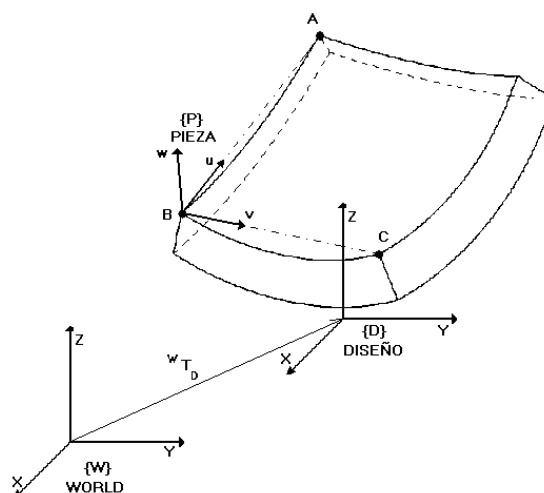


Figura 3: Obtención de la posición y orientación iniciales de la herramienta.

Por lo tanto, disponemos de las coordenadas cartesianas de estos tres puntos *A*, *B* y *C* tanto en el entorno real como en el entorno virtual. En el virtual, conocemos las coordenadas cartesianas respecto al sistema de referencia del diseño $\{D\}$, que denotaremos por el subíndice *D*. Por su parte, en el entorno real conocemos la posición de los tres puntos respecto al sistema de coordenadas $\{W\}$ (también conocido como WORLD) asociado al robot, que denotaremos con el subíndice *W*.

A partir de estos puntos podemos definir un sistema de referencia ligado a la pieza y con origen uno de los puntos, tanto en el entorno real como en el virtual. A estos sistemas de referencia los denotaremos como $\{P_W\}$ y $\{P_D\}$ respectivamente. Para calcularlos nos basta con definir en cada caso tres vectores *u*, *v* y *w* que formen una base ortogonal, lo cual es posible a partir de tres puntos: dos vectores podemos calcularlos como los vectores diferencia entre los puntos, mientras que el tercero lo calculamos como el producto vectorial de ambos.

Una vez que hemos calculado estos sistemas de referencia ligados a la pieza, tenemos determinada la transformación ${}^W T_P$ de $\{W\}$ a $\{P_W\}$ en el entorno real, y la transformación ${}^D T_P$ de $\{D\}$ a $\{P_D\}$ en el entorno virtual. Por lo tanto, conocida la posición de un punto p_i cualquiera de la pieza respecto a los sistemas de referencia WORLD $\{W\}$ o de diseño $\{D\}$, podemos determinar su posición respecto a los sistemas de referencia ligados a la pieza en sendos entornos. Es decir:

$${}^W p_i = ({}^W T_P)^{-1} \cdot {}^W p_i \quad (5)$$

$${}^D p_i = ({}^D T_P)^{-1} \cdot {}^D p_i \quad (6)$$

El último paso consiste en ligar ambas cosas: puntos en la pieza virtual y puntos en la pieza real. Para ello hemos desarrollado un modelo virtual en 3D de la pieza matriz de dimensiones y forma idéntica a las de la pieza real. Teniendo en cuenta este hecho, dado cualquier punto p_i genérico por el cual deba pasar la pieza en el entorno virtual (es decir, dado ${}^D p_i$), podemos calcular su correspondiente ${}^{P_w} p_i$. Y como este punto coincide con el de la pieza real, es decir:

$${}^{P_w} p_i = {}^{P_r} p_i \quad (7)$$

resulta que, igualando las ecuaciones (5) y (6) y despejando, obtenemos que:

$${}^w p_i = {}^w T_p \cdot ({}^D T_p)^{-1} \cdot {}^D p_i \quad (8)$$

En resumen, mediante esta ecuación tan simple podemos convertir los puntos por los que pasa la fresa en el entorno CAD a puntos donde posicionar la herramienta del robot en la realidad.

Es fundamental realizar estos cálculos con mucha precisión para que exista una equivalencia total entre la posición y la orientación iniciales de la herramienta en la realidad y las definidas en el diseño, de forma que la trayectoria de corte definida en el diseño se ejecute en la realidad sobre la pieza de partida, obteniéndose como resultado final el corte deseado. Ésta localización inicial virtual es determinada al trazar la trayectoria de corte en el módulo de CN del sistema CAD/CAM. Aunque es totalmente configurable, generalmente suele ser un punto separado unos 10 cm. del centro de la pieza en dirección normal a la misma.

3.2 COMPOSICIÓN DE LA TRAYECTORIA RESPECTO AL PUNTO INICIAL

Como se ha comentado en el punto anterior, la trayectoria definida en el programa en lenguaje simbólico está referida a un sistema de referencia propio del diseño. Para que sea útil es necesario referir la trayectoria respecto a la posición y orientación iniciales de la herramienta. Esto se consigue utilizando las matrices de transformación homogénea, que permiten realizar una traslación y una rotación mediante una simple operación de multiplicación de matrices. En concreto, si llamamos ${}^w T_H$ a la matriz de transformación homogénea que contiene la posición y orientación iniciales de la herramienta, referida al sistema de referencia WORLD ligado al robot, y ${}^O T_{P_j}$ con $j=\{1,2, \dots, N\}$ a cada uno de los N puntos que componen la trayectoria de corte, referidos respecto al sistema de referencia del diseño, la nueva trayectoria de corte referida al sistema de referencia WORLD viene dada por:

$${}^w T_{P_j} = {}^w T_H \cdot {}^O T_{P_j} \quad (9)$$

Nótese que para realizar esta operación ha sido necesario convertir previamente la representación geométrica de cada punto desde el formato que tuviera a su expresión equivalente mediante una matriz de transformación homogénea.

Para el caso de ${}^w T_H$, como su fuente era el controlador del robot, el formato origen era *XYZeulerZYZ* y la transformación a aplicar viene dada por las ecuaciones:

$${}^w T_H = \begin{bmatrix} & X \\ {}^w R_H & Y \\ & Z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (10)$$

donde la matriz de rotación viene dada por:

$${}^w R_H = \begin{bmatrix} c_\alpha \cdot c_\beta \cdot c_\gamma - s_\alpha \cdot s_\gamma & -c_\alpha \cdot c_\beta \cdot s_\gamma - s_\alpha \cdot c_\gamma & c_\alpha \cdot s_\beta \\ s_\alpha \cdot c_\beta \cdot c_\gamma + c_\alpha \cdot s_\gamma & -s_\alpha \cdot c_\beta \cdot s_\gamma - c_\alpha \cdot c_\gamma & s_\alpha \cdot c_\beta \\ -s_\beta \cdot c_\gamma & s_\beta \cdot s_\gamma & c_\beta \end{bmatrix} \quad (11)$$

siendo $s_X = \sin(X)$ y $c_X = \cos(X)$.

En el caso de ${}^O T_{P_j}$ con $j=\{1,2, \dots, N\}$, la fuente era el programa en lenguaje simbólico, por lo que los puntos estaban en formato *XYZvector*. En este caso, la conversión había que realizarla en 2 pasos: primero debíamos pasar del formato *XYZvector* al formato *XYZeulerZYZ* usando las siguientes ecuaciones.

$${}^O T_p = [X \ Y \ Z \ \alpha \ \beta \ \gamma]^T \quad (12)$$

donde los ángulos de la representación ZYZ de Euler valen, respectivamente:

$$\alpha = A \tan 2(j,i) \quad (13)$$

$$\beta = \begin{cases} j / \sin(\alpha) & \text{si } \sin(\alpha) > 0.5 \\ i / \cos(\alpha) & \text{si } \sin(\alpha) \leq 0.5 \end{cases} \quad (14)$$

$$\gamma = 0 \quad (15)$$

Posteriormente, usando (10) obteníamos la representación en forma de matriz de transformación homogénea. Nótese en (15) que el valor γ de la rotación respecto al nuevo eje \hat{Z} resultante de hacer los giros anteriores se toma igual a cero puesto que, al coincidir con la rotación de la fresa, este grado de libertad no se emplea.

3.3 CHEQUEO DE LA ALCANZABILIDAD

Una funcionalidad fundamental de la que debía disponer la aplicación informática desarrollada era la posibilidad de simular la trayectoria de corte en el PC antes de ejecutarla, de tal forma que si alguno de los puntos de la misma no fuera alcanzable, ésta no se trazara, consiguiéndose así un ahorro de tiempo considerable y evitando posibles complicaciones con el manipulador. En este caso, si la falta de alcanzabilidad no se debe a que la pieza diseñada es de unas dimensiones excesivas para el manipulador empleado, bastaría con cambiar la posición inicial de la pieza matriz -y por tanto el punto de partida de la herramienta- y recalculer la alcanzabilidad de la nueva trayectoria tantas veces como fuera necesario, pero sin necesidad de ejecutarla cada vez.

El criterio seguido para discriminar si un punto de la trayectoria es alcanzable o no consiste en comprobar si el valor de cada una de las seis variables articulares se encuentran dentro de su rango alcanzable. Estos datos, que se conocen de las especificaciones cinemáticas del robot, se muestran a continuación en la tabla 1.

Coordenada articular	Valor mínimo (°)	Valor máximo (°)
θ_1	-160	160
θ_2	-200	35
θ_3	-52.5	232.5
θ_4	-270	270
θ_5	-105	120
θ_6	-270	270

Tabla 1: Rango de valores alcanzables para las variables articulares del manipulador RX-90.

Por consiguiente, para poder comprobar si un punto de la trayectoria era alcanzable necesitamos obtener el valor de las variables articulares del robot conocidas la posición y la orientación de la herramienta en el espacio cartesiano. Para ello debemos resolver el problema cinemático inverso en ese punto empleando como datos los valores de la representación en forma de matriz de transformación homogénea. En concreto, resolveremos el modelo cinemático inverso para un robot manipulador tipo PUMA [2], que son un caso singular debido a que poseen una muñeca (articulaciones de la cadena cinemática que unen el brazo y la herramienta) esférica, que se caracteriza porque los ejes de las articulaciones de la muñeca se cortan en un punto. Esta configuración permite simplificar el análisis cinemático ya que desacopla el problema del posicionamiento de la garra del problema de la orientación de la misma.

Resolver el problema cinemático inverso consiste, por tanto, en resolver la siguiente transformación:

$${}^0T = \begin{bmatrix} r_{11} & r_{12} & r_{13} & X \\ r_{21} & r_{22} & r_{23} & Y \\ r_{31} & r_{32} & r_{33} & Z \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{matrix} {}^0T(\theta_1) \cdot {}^1T(\theta_2) \cdot {}^2T(\theta_3) \cdot {}^3T(\theta_4) \cdot {}^4T(\theta_5) \cdot {}^5T(\theta_6) \end{matrix} \quad (16)$$

La solución general de este problema para un manipulador tipo PUMA viene dada por:

$$\theta_1 = A \tan 2(Y, X) - A \tan 2(d_3, \pm \sqrt{X^2 + Y^2 + d_3^2}) \quad (17)$$

$$\theta_3 = A \tan 2(a_3, d_4) - A \tan 2(K, \pm \sqrt{a_3^2 + d_4^2 - K^2}) \quad (18)$$

siendo
$$K = \frac{X^2 + Y^2 + Z^2 - a_2^2 - a_3^2 - d_4^2}{2a_2}$$

$$\theta_2 = \theta_{23} - \theta_3 \quad (19)$$

siendo

$$\theta_{23} = A \tan 2[(-a_3 - a_2 \cdot c_3) \cdot Z - (-c_1 \cdot X + s_1 \cdot Y)(d_4 - a_2 \cdot s_3), (a_2 \cdot s_3 - d_4) \cdot Z - (a_3 + a_2 \cdot c_3)(c_1 \cdot X + s_1 \cdot Y)]$$

$$\theta_4 = A \tan 2(-r_{13} \cdot s_1 + r_{23} \cdot c_1, -r_{13} \cdot c_1 \cdot c_{23} - r_{23} \cdot s_1 \cdot c_{23} + r_{33} \cdot s_{23}) \quad (20)$$

$$\theta_5 = A \tan 2(s_5, c_5) \quad (21)$$

siendo
$$s_5 = -r_{13} \cdot (c_1 \cdot c_{23} \cdot c_4 + s_1 \cdot s_4) - r_{23} \cdot (s_1 \cdot c_{23} \cdot c_4 - c_1 \cdot s_4) + r_{33} \cdot (s_{23} \cdot c_4)$$

$$c_5 = -r_{13} \cdot (c_1 \cdot s_{23}) - r_{23} \cdot (s_1 \cdot s_{23}) - r_{33} \cdot (c_{23})$$

$$\theta_6 = A \tan 2(s_6, c_6) \quad (22)$$

siendo
$$s_6 = -r_{11} \cdot (c_1 \cdot c_{23} \cdot s_4 - s_1 \cdot c_4) - r_{21} \cdot (s_1 \cdot c_{23} \cdot s_4 + c_1 \cdot c_4) + r_{31} \cdot (s_{23} \cdot s_4)$$

$$c_6 = r_{11} [(c_1 \cdot c_{23} \cdot c_4 + s_1 \cdot s_4) \cdot c_5 - c_1 \cdot s_{23} \cdot s_5] + r_{21} [(s_1 \cdot c_{23} \cdot c_4 - c_1 \cdot s_4) \cdot c_5 - s_1 \cdot s_{23} \cdot s_5] - r_{31} \cdot (s_{23} \cdot c_4 \cdot c_5 + c_{23} \cdot s_5)$$

donde $s_j = \sin(\theta_j)$, $c_j = \cos(\theta_j)$ y a_2, a_3, d_3 y d_4 son los parámetros Denavit-Hartenberg [3] característicos del manipulador.

En nuestro caso debemos particularizar el modelo con los parámetros D-H del robot manipulador empleado, el Staübli RX-90. Estos valores se muestran en la tabla 2.

Articulación	α_{i-1} (°)	a_{i-1} (mm)	d_i (mm)	θ_i
1	0	0	0	θ_1
2	-90	0	0	θ_2
3	0	450	0	θ_3
4	-90	0	450	θ_4
5	90	0	0	θ_5
6	-90	0	0	θ_6

Tabla 2: Parámetros D-H del robot Staübli RX-90.

Una vez que hemos obtenido los valores de las variables articulares θ_i con $i=\{1,2,\dots,6\}$ y hemos verificado que están dentro del rango alcanzable para todos los puntos de la trayectoria, ya estamos en disposición de enviar la trayectoria al controlador, que se encargará de ejecutarla. Por el contrario, en el caso de que algún punto no fuese alcanzable, se mostraría un aviso y la trayectoria no sería enviada al controlador. En este caso, sería labor del operador reiniciar el proceso fijando una nueva posición inicial de la pieza matriz.

3.4 EJECUCION DE LA TRAYECTORIA

Antes de transmitir la trayectoria desde el PC hasta el controlador del robot a través del puerto serie es necesario adaptar los puntos desde su representación mediante matrices de transformación homogénea al formato *XYZeulerZYZ*, que es el único que entiende el controlador. Para ello es necesario resolver las siguientes ecuaciones:

$$\text{Si } 0 < \beta < 180^\circ \Rightarrow \begin{cases} \alpha = A \tan 2(r_{23}, r_{13}) \\ \beta = A \tan 2(\sqrt{r_{31}^2 + r_{32}^2}, r_{33}) \\ \gamma = A \tan 2(r_{32}, -r_{31}) \end{cases} \quad (23)$$

En caso de que $\beta=0$ ó $\beta=180^\circ$ la solución es degenerada, y sólo puede calcularse la suma o la diferencia de α y γ . Eligiendo $\alpha=0$ se obtienen las soluciones:

$$\alpha = 0, \quad \beta = 0, \quad \gamma = A \tan 2(-r_{12}, r_{11}) \quad (24)$$

$$\alpha = 0, \quad \beta = 180^\circ, \quad \gamma = A \tan 2(r_{12}, -r_{11}) \quad (25)$$

Finalmente, después de transmitir la trayectoria hacia el armario de control es necesario ejecutarla. De esto se encargará un software propio del controlador llamado V_TRAJSIG [8]. El operador sólo debe encargarse de configurarlo de forma adecuada y de pasarle datos válidos.

V_TRAJSIG es un programa para los robots Staübli de la serie RX creado en el lenguaje de programación VAL+. La función de este programa consiste básicamente en la generación y el seguimiento de trayectorias a partir de puntos de referencia en

coordenadas cartesianas. Además, permite controlar gran cantidad de parámetros relacionados con dichas trayectorias, como por ejemplo parámetros estáticos (tales como definiciones geométricas, posiciones...), parámetros dinámicos (velocidades, aceleraciones, paradas programadas...) e incluso las comunicaciones y periféricos asociados (principalmente constituidos por entradas y salidas tanto digitales como analógicas, control de ejes adicionales...).

4 IMPLEMENTACIÓN

En la figura 4 puede observarse que el sistema empleado consta de un robot manipulador Staübli de la serie RX-90 con su correspondiente controlador CS7 [9], donde se encuentran las tarjetas encargadas de controlar la posición de los servomotores mediante distintos bucles de realimentación. También consta de un ordenador PC compatible donde se diseñan las piezas empleando una herramienta CAD/CAM y donde se ejecuta la aplicación desarrollada que sirve como interfaz. Por último, el sistema de corte está formado por una fresadora neumática, una electroválvula que permite el control de la fresadora desde el controlador del robot, y un grupo de filtraje que soluciona el problema de la acumulación de agua de condensación en la fresa.

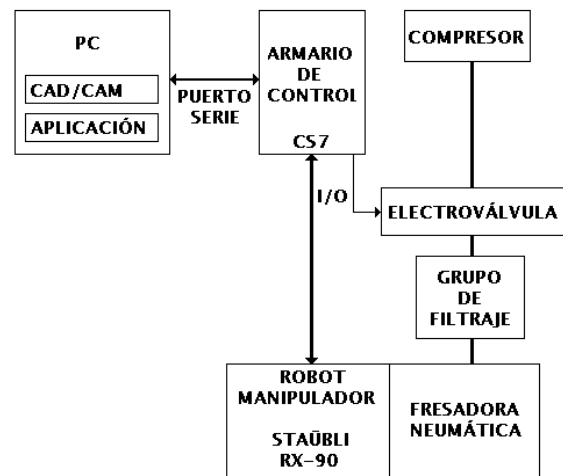


Figura 4: Esquema de los componentes del sistema utilizado.

El RX-90 es un robot de tipo brazo antropomórfico servocontrolado con posibilidad de especificar completamente la trayectoria a seguir por el extremo. Como se ha indicado anteriormente, este robot posee seis grados de libertad correspondientes a sus seis articulaciones de rotación, de las cuales tres son utilizadas para posicionarse en el espacio. Las otras tres se emplean para orientar la herramienta situada en el extremo y se encuentran localizadas en la muñeca.

Como puede apreciarse en la figura 5, para realizar el corte se ha colocado una fresadora neumática en el extremo sustituyendo la garra del manipulador. El útil de corte empleado debe adaptarse al material del que está compuesto el stock, que puede ser fibra de carbono, titanio o cualquier otro de los utilizados normalmente en el sector aeronáutico.



Figura 5: Robot manipulador con la fresadora neumática acoplada.

Con el objetivo de ser capaz de accionar la fresa directamente desde el controlador se ideó un sistema formado por una electroválvula controlada por medio de una de las salidas digitales de las que dispone el controlador del robot. Esta electroválvula se encuentra instalada a continuación del engrasador, y se conecta a través de un tubo flexible con la fresadora.

La elección de la electroválvula se realizó en base a varios criterios: la válvula debía ser del tipo cerrada en posición básica; el rango de presión del sistema de aire comprimido disponible en el laboratorio oscilaba entre los cinco y los ocho bares; el nivel y el tipo de tensión de la bobina se escogieron teniendo en cuenta que el armario de control disponía de una salida de tensión continua de 24V, así como de salidas digitales de tipo relé, por lo que se optó por una electroválvula activada por bobina de 24V de corriente continua; el caudal de aire consumido por la herramienta era inferior a los 1000 l/min. Teniendo en cuenta todas estas consideraciones, se escogió la electroválvula de 2/2 vías MN1H-2-1/4-MS de la marca FESTO.

Para adaptar la fresadora a la muñeca del robot ha sido necesario diseñar una estructura de acoplamiento [6]. Este sistema debía cumplir tres premisas básicas: debía ser ligero (ya que la carga máxima del manipulador es de 6 kg.) pero a su vez lo suficientemente robusto para que no se produjeran imprecisiones en el movimiento; debía ser simple y su coste no podría ser excesivamente alto; y finalmente, la transformación cinemática entre el

extremo del manipulador y el punto de operación de la herramienta debía ser lo más sencilla posible. La solución adoptada consistió en una estructura de tres piezas en aluminio con geometría cilíndrica, con lo que la transformación queda reducida a un desplazamiento (de unos 275mm. dependiendo del decalaje de la fresa) en el eje Z del sistema de referencia asociado al extremo del manipulador (sistema de referencia TOOL en la programación del robot).

Un último problema que tuvo que ser resuelto con respecto a la fresadora neumática fue la aparición de agua de condensación a la salida de la fresa, lo cual era muy perjudicial para la fresa y el sistema en general ya que provocaba una rápida oxidación del mismo. Para solucionarlo se añadió un grupo de filtraje formado por una válvula manual de entrada de 3/2 vías de la marca SMC modelo EVHS 4000-F04 que aislaba el conjunto de la instalación, un elemento deshumidificador modelo EAW 4000-F04 de la marca SMC encargado de eliminar el agua de condensación (que dispone además de un regulador de la presión y de un manómetro que nos permite ajustar y controlar la presión en función de la aplicación), y un elemento engrasador SMC EAL 4000-F04 cuya misión era lubricar el aire a la salida de éste con el objeto de mejorar la conservación de la herramienta.

Como se mencionó anteriormente en la introducción, el objetivo de este proyecto es la automatización del corte de piezas ligeras de aviación. Generalmente, estas piezas son superficies curvas de un grosor de unos pocos centímetros. Para el diseño de estas piezas hemos empleado una herramienta CAD/CAM que suele utilizarse con asiduidad en el mundo del diseño aeronáutico.

La generación de las trayectorias necesarias para realizar el mecanizado de las mismas se realiza mediante el módulo de CN de dicha solución informática, configurado para trabajar con máquinas de CN de cinco grados de libertad, que es precisamente lo que necesitamos debido a que de los seis grados de libertad que nos ofrece el manipulador no utilizamos el que constituye la rotación de la fresa.

Éste módulo nos permite, además, definir tanto la geometría de la herramienta de corte, con lo que las correcciones por herramienta ya son tenidas en cuenta a la hora de generar los puntos de la trayectoria, como los puntos de aproximación y alejamiento, lo que nos evita tener que crear dichos puntos manualmente, además de asegurarnos un menor riesgo de colisión al determinar por donde comenzará a ejecutar la trayectoria el robot.

El resultado final del diseño de la pieza y de la definición de su trayectoria de corte es un programa de CN en lenguaje simbólico. En nuestro caso particular, se ha utilizado un postprocesador para APT en el módulo de CN de la herramienta CAD/CAM. Se ha preferido este lenguaje universal al código ISO de algún CN en particular puesto que de este modo sería posible partir de cualquier otra fuente que generase un archivo APT tratado con el mismo postprocesador.

El postprocesador elegido nos genera un archivo donde las trayectorias son guardadas como una secuencia de puntos, que están formados por las tres coordenadas del punto y tres coordenadas que representan la orientación de la herramienta en el espacio. Para obtener la trayectoria basta con interpolar tanto las coordenadas del punto como las de la orientación.

A partir de estos ficheros en lenguaje APT, la aplicación desarrollada se encarga de simular la trayectoria realizando los cálculos intermedios necesarios y, en caso de que sea totalmente viable, adaptar la representación geométrica de los puntos de la misma al formato entendible por el controlador y ejecutar el software de seguimiento de trayectorias V_TRAJSIG.

5 CONCLUSIONES

En este artículo se presenta una aplicación informática desarrollada con el objetivo de automatizar el corte de piezas en tres dimensiones. Este software tiene un amplio abanico de posibilidades de uso, pues no sólo se reduce al corte de piezas ligeras de aviación, sino que también es útil, por ejemplo, para el corte de cualquier otro tipo de material sin más que cambiar el útil de corte, o para la aplicación de adhesivos o siliconas a lo largo de un contorno sin más que cambiar la herramienta de trabajo.

Se pretende presentar el proyecto a una empresa dedicada al mecanizado de piezas para el sector aeronáutico. No obstante, los resultados obtenidos hasta el momento son bastante satisfactorios en cuanto a precisión y calidad de acabado de corte. Además, la aplicación se ejecuta a una velocidad elevada, de forma que el tiempo de cálculo necesario resulta despreciable frente al tiempo empleado en transmitir la información por el puerto serie y recibir las correspondientes confirmaciones por parte del controlador.

A pesar de estos logros, éste no es un proyecto cerrado. Debido a su alto interés práctico, actualmente se está trabajando en algunas mejoras del sistema. Por ejemplo, para el cálculo de la

posición inicial de la herramienta se está estudiando el empleo de un sistema de visión estéreo [4] que permita conocer la localización tanto de la pieza matriz sobre la que se va a realizar el corte como del extremo de la herramienta con un error inferior a 0.1mm. Con esta mejora se evitaría tener que “enseñar” al manipulador las posiciones necesarias para calcular la posición y orientación iniciales de la herramienta comentadas en el subapartado 3.1, acelerando todo el proceso de corte. Otra mejora importante consistiría en añadir a la aplicación un soporte para bases de datos de tal forma que pudieran almacenarse varias trayectorias de corte, con lo que no haría falta recalcularlas cada vez en caso de que el manipulador trabajara con distintos tipos de piezas. Finalmente, para el caso de piezas de titanio o de otro material más duro que la fibra de carbono, sería necesario realizar el corte en varias pasadas o incluso se podría incorporar una herramienta láser.

Agradecimientos

Los autores desean agradecer al MCyT la financiación de este trabajo con cargo al proyecto de investigación DPI2001-2424-C02-01.

Referencias

- [1] Barrientos, A. y otros (1997) “Fundamentos de Robótica”. Mc Graw-Hill.
- [2] Craig J.J. (1986) “Introduction to Robotics. Mechanics and Control”. Addison Wesley.
- [3] Fu K.S., González R.C. y Lee C.S.G (1988) “Robótica: control, detección, visión e inteligencia”. McGraw-Hill.
- [4] García Florido J.C. (2003) “Análisis teórico e implementación de un sistema de calibración estereoscópico”. Proyecto fin de carrera. Universidad de Sevilla.
- [5] González Núñez, J. (1986) “El control numérico y la programación manual de las máquinas herramienta con control numérico”. Urmo.
- [6] Vegas Cornejo A. (200X) “Automatización del corte de piezas en dos y tres dimensiones mediante el uso del robot Staübli RX-90”. Proyecto fin de carrera. Universidad de Sevilla.
- [7] Manual Staübli RX-90. Staübli Española, S.A.
- [8] Manual de V_TRAJSIG. Staübli Española, S.A.
- [9] Manual del armario de control CS7. Staübli Española, S.A.