Capítulo 2

Aspectos geométricos

Los trabajos de investigación recientes están dedicados a la generación de trayectorias de herramienta complejas. La interpolación y el cálculo de la trayectoria de la herramienta se basa en la creación de curvas paralelas en donde la herramienta la sigue tangencialmente, observando que la dirección de la herramienta concuerde con el vector normal de la superficie, moviendo para ello los ejes que la máquina tenga disponibles.

En este Capítulo se detalla la problemática del manejo de curvas complejas en una máquina-herramienta, lo cual deja en claro la insuficiencia de la cuantificación del maquinado en términos de vida de herramienta, y esboza un panorama en el que es necesario involucrar la velocidad de avance en las evaluaciones de maquinabilidad.

El modelo geométrico a maquinar puede ser generado por diferentes medios de construcción [LOP00]. Ya sea a partir de la salida de una máquina de coordenadas o directamente del modelo conceptual implementado y mejorado con un sistema de diseño por computadora [FOL96].

Los algoritmos de generación de caminos de maquinado parten de la construcción del modelo geométrico en base a curvas determinadas matemáticamente. La mayoría de los algoritmos tienen 3 fases (ver figura 2.1):

1. Generación de puntos “CC” de contacto (del inglés *cutter contact*)
2. Generación de la posición “CL” y ángulo del elemento cortante “CO” (del inglés *cutter location* y *cutter orientation*)
3. Postprocesamiento

La generación de los puntos CC de la fase 1 consiste en el trazado de líneas a través de la geometría para encontrar a intervalos el punto de contacto con la superficie. Para definir los puntos CL y los ángulos de orientación CO de la fase 2 se utilizan los puntos CC y se calculan los vectores normales en la geometría a maquinar. Para máquinas con más de 3 ejes puede existir más de un punto CL y un ángulo CO para cada CC, debido a que la herramienta puede tener más de una orientación por punto.

La última fase, llamada procedimiento de *postprocesamiento* convierte los CL y CO en código NC. Para diferentes máquinas con diferentes cadenas cinemáticas debe generarse código NC diferente y, básicamente, el postprocesamiento es una transformación de coordenadas.

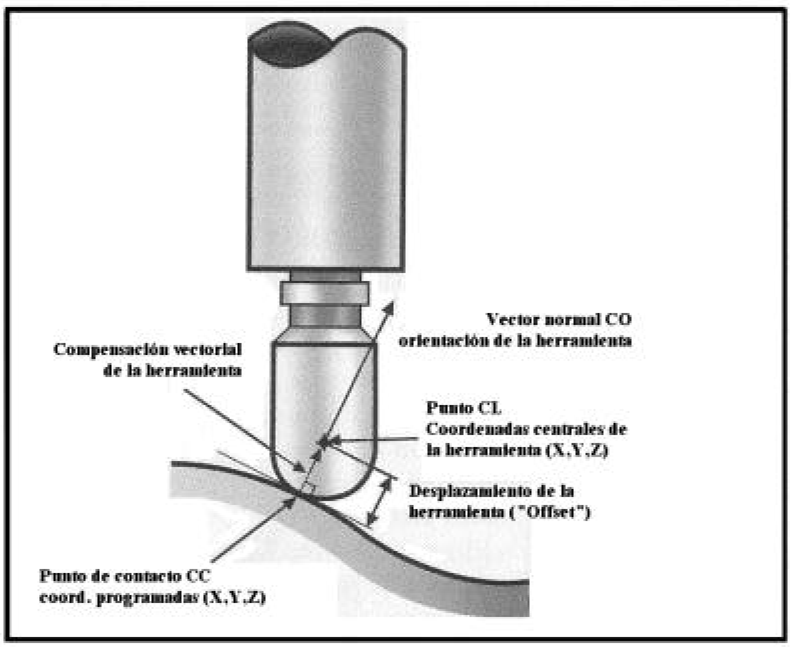

Fig. 2.1.- Puntos necesarios para la generación de trayectorias de herramienta.

Ya que solamente los controles modernos son capaces de procesar curvas complejas [JEO99a, JEO99b, LOP01b] (esencialmente las llamadas NURBS [PIE97], cuyo nombre proviene de las siglas en inglés *Non-Uniform Rational B-Splines,* curvas matemáticas parametrizadas definidas por puntos de control y que forman parte de la técnica moderna para el diseño de geometrías en computadora) es necesario encontrar un método que permita obtener una calidad similar utilizando un control que no pueda manejar esa matemática compleja [LOP01a].

A menudo ocurre que las características de procesamiento de una computadora sobrepasan las del control de una máquina-herramienta. Esto significa que, para la mayoría de los programas de diseño mecánico, se supone una máquina-herramienta de características adecuadas a él, lo cual normalmente no sucede [LOP00], ya que el desarrollo del ámbito computacional es de mucho mayor velocidad que el de las máquinas-herramienta. Esto sucede básicamente por las siguientes razones:

- La computadora se ha convertido en un producto casero, por lo que el número de personas a nivel mundial trabajando sobre desarrollos y mejoras es mucho mayor al de las máquinas-herramienta.
- Tanto la construcción como los códigos de programación de una máquina-herramienta están casi en su totalidad estandarizados mundialmente (salvo un grupo pequeño de códigos específicos). Lo cual no ocurre con las computadoras cuyos estándares son determinados prácticamente por el momento del mercado.
- El costo de venta de una máquina-herramienta es mucho mayor que el de una computadora y, en consecuencia, las capacidades innovativas en las nuevas generaciones de máquinas-herramienta suceden a intervalos mayores.
- El atraso tecnológico de algunos países obliga a muchas empresas a trabajar con máquinas-herramienta parcial o totalmente obsoletas.

2.1 Código de control numérico

El método tradicional de programación de las máquinas-herramienta se basa en códigos estandarizados en forma de instrucciones de máquina (ver Tabla II.I). Estos códigos son suministrados al control de la máquina-herramienta en forma manual por medio del teclado del control de la máquina o a través de una computadora. El significado de estos códigos forma parte de un estándar internacional para los controles de las máquinas, y la ejecución de cada código se encuentra en la memoria del control de la máquina herramienta.

Cuando se desea maquinar piezas con geometrías que contengan algo más que algunas líneas rectas y arcos es necesario usar un programa de diseño por computadora.

Actualmente existe un buen número de ellos en el mercado. Generan código NC automáticamente a partir de geometrías que el usuario suministra. Sin embargo, existen situaciones de mercado en donde las empresas o negocios requieran de una inversión alta en la adquisición de programas de computadora de diseño para la fabricación con máquinas-herramienta, y el entrenamiento del personal en ello, lo cual no siempre se justifica en términos de tiempo, producción y ganancia.

La generación automática de código NC tiene una complejidad elevada y una curva de aprendizaje prolongada, que son independientes del grado de complejidad del producto a fabricar; esto tiene como consecuencia que, en ambientes productivos, el recurso humano es costoso en factores de tiempo y economía. Las instituciones de educación se enfrentan al problema adicional de tener que decidir cómo modular el aprendizaje, pues la duración de un curso -normalmente semestral- no contempla todo el proceso de fabricación en detalle. Esto repercute en la forma de interacción de los objetivos entre cursos.

El código NC se compone de instrucciones en forma de “palabras” (ver tabla II.I) que modifican el estado de la máquina, como por ejemplo movimientos de la herramienta, velocidades, repeticiones, etc. Salvo la nueva generación de máquinas-herramienta modernas, en la mayoría de los controles están implementados solamente los códigos que manejan entidades geométricas sencillas como movimientos lineales y arcos. Elaborar código NC por medio del método tradicional para geometrías de mediana complejidad podría llevarle a un programador de máquinas-herramienta tanto tiempo que no sería ni práctico ni rentable, pues tendría que hacer cálculos de continuidad geométrica e interpolaciones para adaptar los códigos de programación; en consecuencia puede decirse que la programación manual en código NC está encaminada a la fabricación de piezas de geometría sencilla.

Tabla II.I. Funciones estándar según la norma DIN66025.

G00	Posicionamiento en desplazamiento rápido
G01	Interpolación lineal
G02	Interpolación circular en el sentido de las agujas del reloj
G03	Interpolación circular en el sentido contrario al de las agujas del reloj
G04	Retardo
G17	Elección de plano XY
G18	Elección de plano XZ
G19	Elección de plano YZ
G33	Roscado con avance constante
G34	Roscado con avance creciente
G35	Roscado con avance decreciente
G40	Borrado de todas las correcciones de herramienta invocadas
G41	Corrección del radio de la herramienta, inclinación hacia la izquierda
G42	Corrección del radio de la herramienta, inclinación hacia la derecha
G43	Corrección del radio de la herramienta, positiva
G44	Corrección del radio de la herramienta, negativa
G53	Borrado del desplazamiento del punto cero invocado
G54-G59	Desplazamiento del punto cero
G60	Primera tolerancia de entrada
G61	Segunda tolerancia de entrada, también realizar un bucle
G62	Posicionamiento rápido
G63	Ajustar el avance al 100% para el roscado
G64	Cambio de avance y/o velocidad de giro
G70	Aproximar el eje Z a la posición de salida
G73	Avance programado = avance de eje
G74	Aproximar puntos de referencia de primer y segundo ejes
G75	Aproximar puntos de referencia de tercer y cuarto ejes
G80	Borrado de los ciclos invocados
G81-G89	Ciclos de taladrado fijados
G90	Introducción de medidas de referencia
G91	Introducción de medidas relativas
G92	Desplazamiento programado del punto de referencia
G94	Avance en mm/min
G95	Avance en mm/rev

En contraste con el método tradicional, los programas de computadora para dibujo y diseño mecánico permiten la elaboración parametrizada [SEN98] de entidades geométricas elementales tales como puntos, líneas y radios hasta entidades geométricas complejas como curvas splines. Si se requiere el maquinado de dichas geometrías, la interpretación de estas curvas de precisión requiere de programas de computadora apropiados para la generación de archivos CLF (del inglés *cutting location file*) en donde se consideran las características geométricas de pieza, materiales, máquina y herramental, así como de programas postprocesadores para la generación de código NC específico para una máquina determinada. Este procesamiento de información requiere por parte del usuario conocimiento acerca de:

- el tipo de máquina usada (torno, fresadora, etc),
- el número de ejes motrices,
- el controlador destino,
- los datos geométricos de herramientas,
- los parámetros de maquinado (profundidad de corte, velocidad, acabado).

Esta información debe ser dominada por el usuario para generar código NC de mediano grado de complejidad.

En la siguiente sección se presenta el impacto de la filosofía de operación del control en la generación automática de código NC para controles modernos.

2.2 Posición, velocidad y aceleración

Los avances tecnológicos en la microelectrónica hacen posible que hoy en día pueda integrarse en una sola unidad el sistema de control para motores de las máquinas-herramienta. La interacción del motor, el convertidor y los componentes de control pueden optimizarse como un todo.

Funcionalmente, un control de máquina-herramienta consta de un procesador de bloques de código y un ejecutor de dichos bloques, el cual está dividido en tareas de diferentes prioridades. Las funciones preparatorias para el procesamiento del código se ejecutan en el fondo (modo de "background"), mientras que las secuencias de movimiento individual son controladas y reguladas por el ejecutor de bloques.

El procesador de bloques tiene asignadas las siguientes tareas:

- creación del programa en memoria,
- decodificación e interpretación (generación del código intermedio),
- preprocesador (transformación de coordenadas, compensaciones de herramienta, función previsoria “look-ahead”).

Mientras que al ejecutor de bloques le corresponden las funciones de:

- procesador principal (interpolación, transformación de coordenadas, control de movimiento),
- interpolación fina, control de posición,
- control de velocidad, control del par de torsión,
- modulación de pulsos.

La calidad del maquinado de una máquina-herramienta está directamente influida por la construcción mecánica, la interacción y el control del motor de cada eje. Las características deseables son:

- buen desempeño: movimientos suaves y precisos, emanados del control,
- estabilidad robusta: respuesta estable independiente de perturbaciones, como por ejemplo al ruido o a las cargas repentinas.

Tanto en los movimientos suaves y precisos, como la calidad de respuesta, el tipo de motor-convertidor juega un rol determinante. Para muchos casos, el motor es sincrónico, con un momento de inercia bajo, con capacidad de sobrecarga. El convertidor tiene tiempos de muestreo cortos para el sistema de lazo cerrado y características de control que compensan las no-linealidades como la fricción.

El seguimiento adecuado de la trayectoria se logra por medio de la combinación *convertidor/NC*. El control central de velocidad y par se asegura de que las acciones correctivas y la sincronización se lleven a cabo, mientras que otros módulos de control garantizan que la trayectoria definida sea seguida fielmente por el servomecanismo. Un arreglo especial de la estructura *preprocesador/procesador principal* asegura que el potencial de productividad de cada componente pueda ser explotado al máximo.

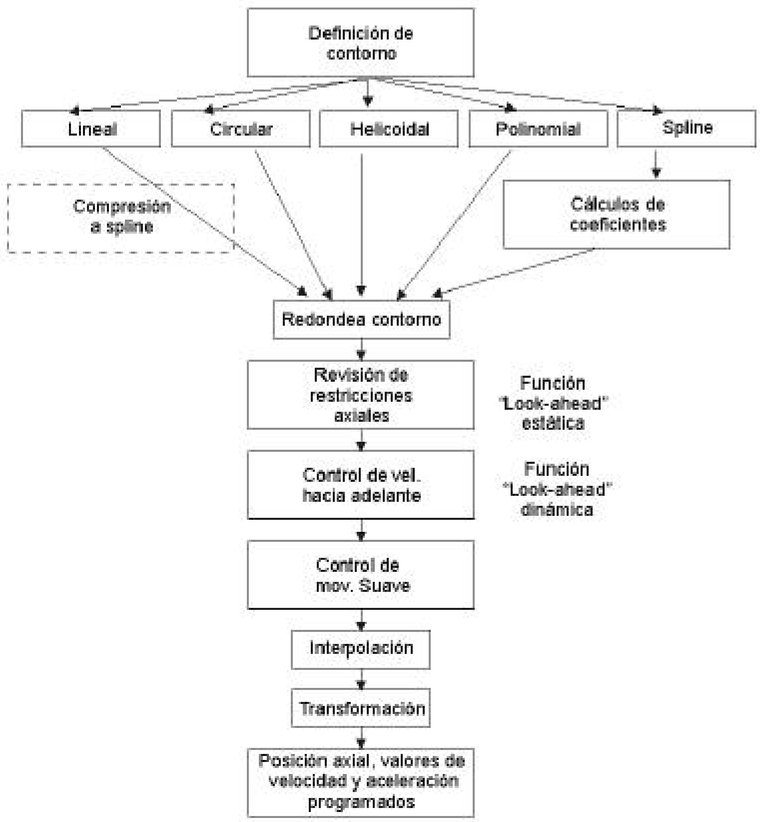

Fig. 2.2.- Procesamiento de datos geométricos y de control de movimiento para una máquina-herramienta con control completo [PAP96].

El usuario define un contorno ya sea lineal, circular, helicoide, polinomial o del tipo spline. En la figura 2.2 se observa que para el caso spline el control calcula los coeficientes de los polinomios a partir de los puntos de control antes de procesar. Es posible también comprimir una secuencia de bloques lineales (la cual es la forma usual en que un sistema CAD/CAM se almacenan curvas complejas para garantizar su realización por un control incapaz de procesar curvas complejas) en bloques para splines. Suponiendo que se mantiene la misma exactitud, este proceso reduce drásticamente a) el número de bloques de ejecución en el control, b) el tamaño del programa en memoria y c) el tiempo de cálculo. Es necesario hacer notar que el uso de rectas para almacenar splines conlleva errores de exactitud, y somete al control a cambios abruptos de trayectoria. Como se verá en la siguiente sección, el uso de splines, en cambio, da como resultado una trayectoria de contorno continua, evitando momentos de inercia fuertes y golpes en las partes mecánicas debido a cambios abruptos de trayectoria, lo cual permite la operación a mayores velocidades de avance [PAP96].

Las características de velocidad y aceleración del movimiento de cada eje de la herramienta para un contorno dado de tres bloques (*N1*, *N2* y *N3*) se muestran en la figura 2.3, suponiendo que tanto en el punto inicial como en el punto final la herramienta se encuentra en reposo. Los valores de la aceleración para los ejes X y Y se representan como a_X y a_Y respectivamente; la suma vectorial de ambos es la resultante a_P. El vector Velocidad V_P tiene al inicio de la trayectoria programada una aceleración positiva, mientras que al final una negativa.

El criterio de redondeo para cambios abruptos de posición se muestra en la figura 2.4. El control de la máquina permite al usuario determinar el radio de redondeo de trayectoria para puntos críticos.

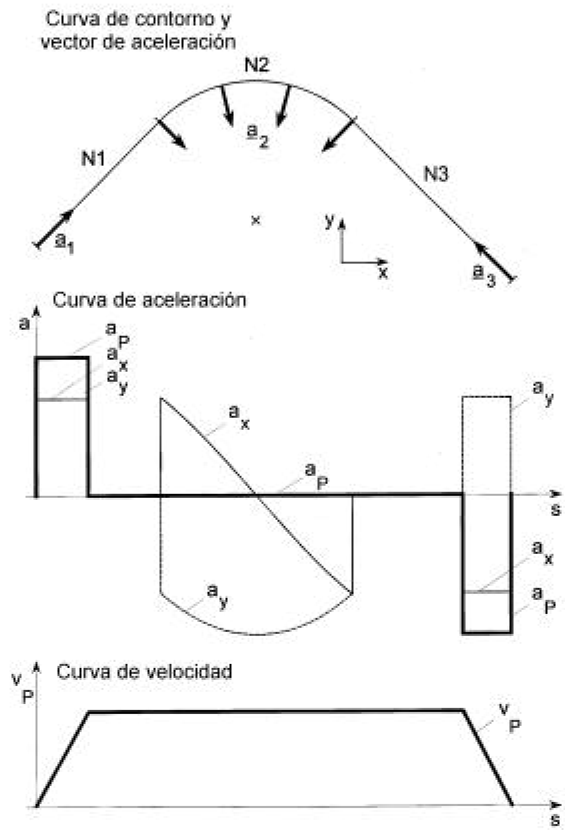

Fig. 2.3.- Comparación de la trayectoria y las aceleraciones de los ejes [PAP96].

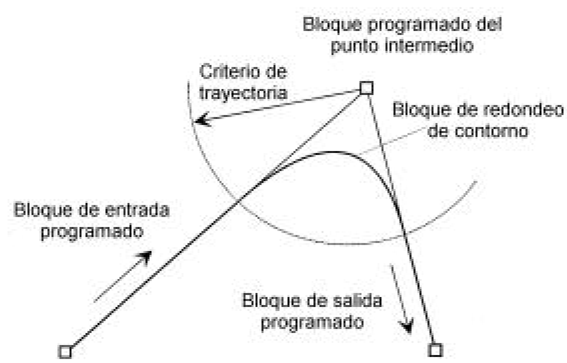

Fig. 2.4.- Redondeo del contorno de acuerdo a un criterio de trayectoria, para evitar errores de sobrecorrida del control ("over-run error").

La función previsoria ("look-ahead function") adelanta los cambios de movimiento y posición para el siguiente bloque, lo cual permite un cambio más suave, previniendo de esta manera un error potencial de velocidad o posición programada imposibles de alcanzar. Las limitantes de esta función están dadas, entre otros factores, por la velocidad máxima y la distancia geométrica de trayectoria requeridas por el usuario para ese bloque de código.

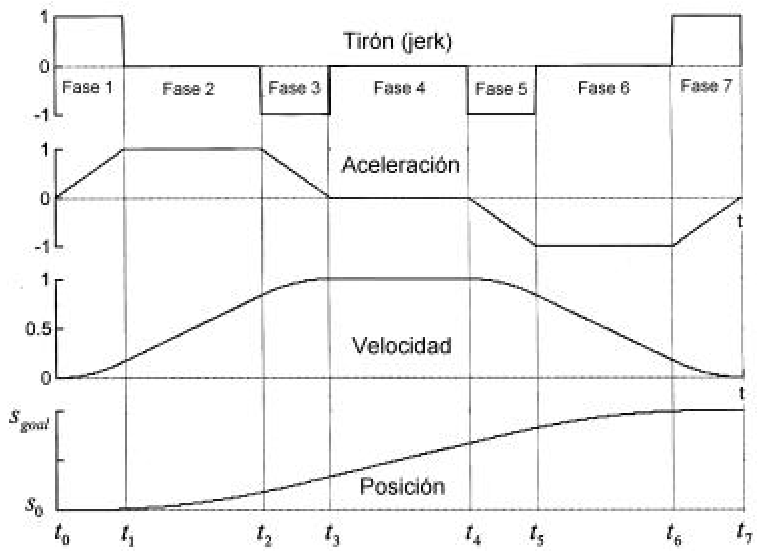

Fig. 2.5.- Curvas de tiempo normalizadas con un control de movimiento suave [PAP96].

En resumen, la construcción del control de los ejes de una máquina-herramienta determina la capacidad de la máquina para ejecutar movimientos curvos con continuidad de velocidad de avance y exactitud posicional (ver figura 2.5). La función previsoria "look-ahead" permite adelantar la interpretación de código NC e intentar mantener la velocidad de avance constante a lo largo de las trayectorias de los bloques siguientes (ver figura 2.6), sin embargo esto no es posible para todos los bloques de código NC. Las características de procesamiento del control para trayectorias libres son un factor determinante para el proceso de maquinado y en consecuencia en las propiedades de acabado del material. Esto hace que se establezca una diferencia con el tipo de maquinado usado para las evaluaciones de maquinabilidad .

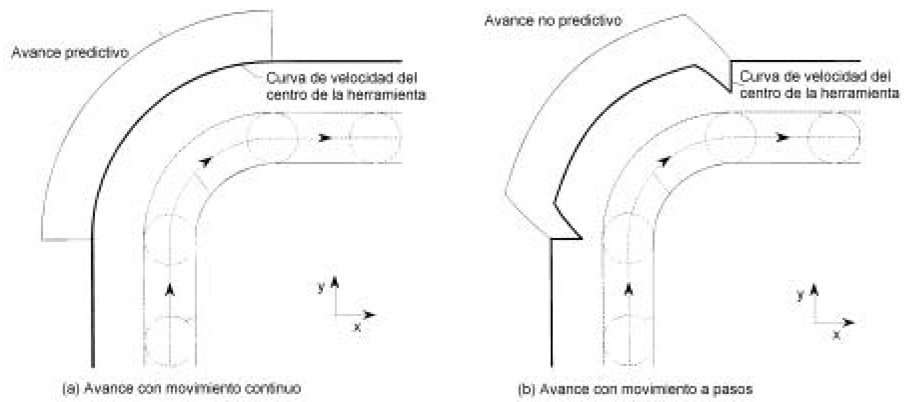

Fig. 2.6.- Diferencia de procesamiento de bloques de código a)usando la función previsoria "look-ahead" y b) sin ella [PAP96].

2.3 Curvas complejas

Un método de representación de curvas consistente debe observar que:

- sea posible representar exactamente todas las curvas que sean necesarias,
- sea fácil, eficiente y lo más exacto posible procesarlas en computadora,
- el cálculo de puntos y derivadas sea sencillo,
- los métodos numéricos sean robustos y acarreen el mínimo error de redondeo,
- se requiera poca memoria computacional para procesar,
- sean sencillas y matemáticamente fáciles de entender.

Los métodos usados para la representación de geometrías son las ecuaciones *implícitas* y las *paramétricas*. La función $f(x,y) = 0$ es una función implícita. Un ejemplo (ver figura 2.7) es la clásica ecuación de un círculo de radio unitario y centro en el origen:

$$f(x,y) = x^2 + y^2 - 1 = 0$$

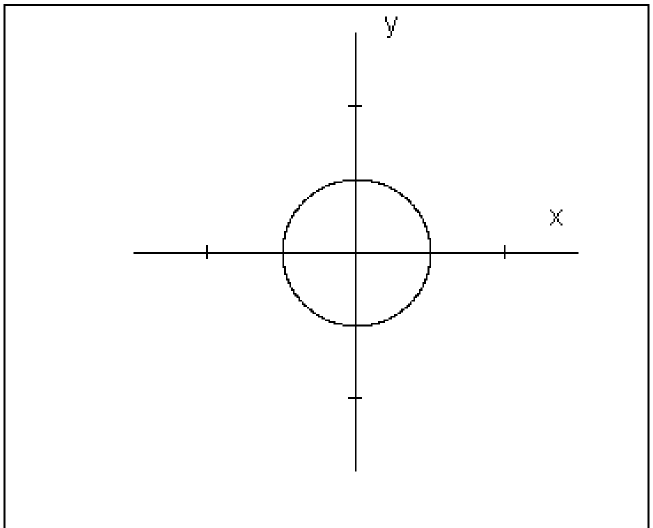

Fig. 2.7.- Forma implícita de un círculo con radio unitario.

Sin embargo, ésta no es la única forma de representar un círculo. Está ampliamente demostrado que un círculo puede representarse por medio de las funciones trigonométricas seno y coseno:

$$x(t) = \text{sen}\ (t)$$

$$y(t) = \cos\ (t)$$

en donde $0 < t < 2\pi$

Con lo que se tiene el círculo en forma paramétrica de la figura 2.8. Es importante hacer notar que puede existir más de una forma paramétrica para una curva. Sin embargo, la selección de esta forma está determinada por los criterios de sencillez que se deseen tener en un caso determinado.

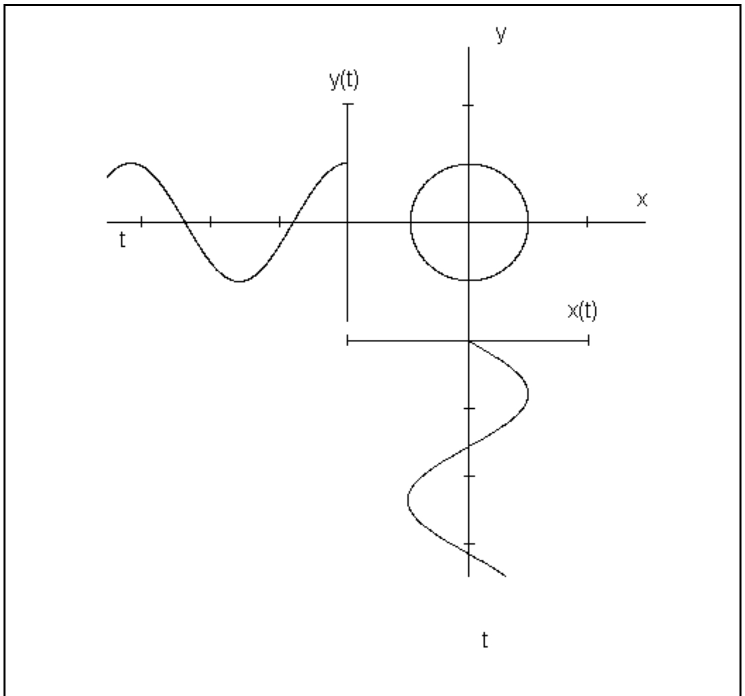

Fig. 2.8.- Forma paramétrica de un círculo con radio unitario.

El uso de la forma paramétrica se ha extendido mucho debido a las propiedades matemáticas que permiten su manejo sencillo y flexible [GLA97].

2.3.1 Representación de curvas de Bèzier

La parametrización de una curva por medio de la de sus componentes coordenadas *x(u)*, *y(u)* puede ser arbitraria, y en consecuencia es posible obtener un amplio espectro de curvas. El tipo de curva implementado tanto en los sistemas de CAD/CAM (ver [CHU97]), como en los controles modernos de máquinas-herramienta son las B-Splines. Para mantener sencillez y claridad, esta sección se centra en curvas de Bèzier (las cuales son un caso particular de splines) en el plano XY.

Las curvas de Bèzier se basan en los polinomios de Bernstein (ecuación 2.1), que se definen como:

$$B_{i,n}(u) = \frac{n!}{i!\cdot(n-i)!}u^{i}\cdot(1-u)^{n-i} \tag{2.1}$$

En donde $0 < u < 1$; i es el índice del vector de n polinomios.

Las ventajas que justifican su uso son:

- no negatividad: $B_{i,n}(u) \geq 0$ para toda i,n y $0<u<1$,
- partición unitaria,
- $B_{0,n}(0) = B_{n,n}(1) = 1$,
- simetría con respecto a: $u=i/n$,
- definición recursiva inclusive sus derivadas.

Las curvas de Bèzier se definen como la multiplicación de estos polinomios por un vector de puntos $\{\mathbf{P_i}\}$.

$$C(u) = \sum_{i=0}^{n} P_i B_{i,n}(u) \tag{2.2}$$

Las ecuaciones (2.3) para cada una de las coordenadas son:

$$C_x(u) = \sum_{i=0}^{n} Px_i \cdot B(i,n,u) \tag{2.3}$$

$$C_y(u) = \sum_{i=0}^{n} Py_i \cdot B(i,n,u)$$

Cuya curva graficada se muestra en la figura 2.9 para el caso de $\{\mathbf{P_i}\}$. Se debe hacer notar la posición de los puntos $\{\mathbf{P_i}\}$. El primer punto del vector es el punto de *inicio* de la curva; el último es el punto *final* de la curva. La curva *sigue* a los puntos intermedios, por lo que al vector $\{\mathbf{P_i}\}$ se le llama vector de **puntos de control**. Al polígono formado por el vector $\{\mathbf{P_i}\}$ se le conoce como polígono de control.

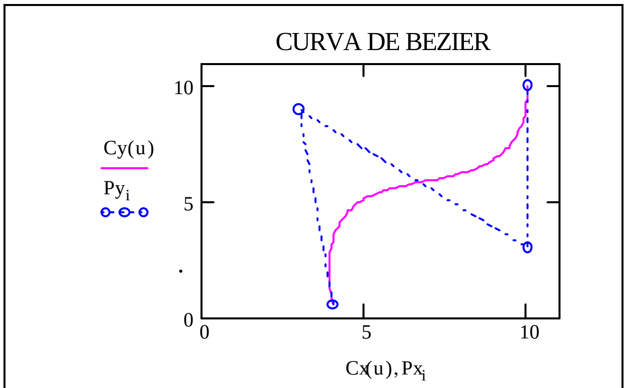

Fig. 2.9.- Ejemplo de una curva con n=3.

Las ventajas de representar curvas parametrizadas por este método se pueden concluir a través de sus propiedades matemáticas:

- Los polígonos de control aproximan la silueta de la curva,
- $P_0 = C(0)$ y $P_3 = C(1)$,
- las direcciones de tangencia de los puntos inicio y final son paralelas al segmento P_1-P_0 y P_3-P_2,
- al inicio (u=0) la curva toma la dirección de $P_0P_1P_2$. Al final (u=1) toma la dirección de $P_1P_2P_3$.

Vector de pesos

Es posible alterar la trayectoria de la curva de Bèzier “jalándola“ hacia los puntos de control intermedios. De esta manera puede modificarse sin alterar los puntos de inicio y final, conservando con esto la continuidad geométrica de la unión entre segmentos. Esto se realiza utilizando un *vector de pesos* $\{\mathbf{w_i}\}$ para obtener la curva racionalizada. Este vector es un valor de peso asignado a cada punto de control. El cálculo de las coordenadas será:

$$x(u) = \frac{X(u)}{W(u)} \qquad y(u) = \frac{Y(u)}{W(u)} \tag{2.4a}$$

En donde

$$X(u) = \sum_{i=0}^{n} Px_i \cdot B_{i,n}(u) \cdot w_i \tag{2.4b}$$

$$Y(u) = \sum_{i=0}^{n} Py_i \cdot B_{i,n}(u) \cdot w_i$$

y

$$W(u) = \sum_{i=0}^{n} w_i \cdot B_{i,n}(u) \tag{2.4c}$$

El requisito para poder formar una curva a través de la unión de una sucesión de curvas es la continuidad geométrica. Ésta expresa la suavidad de la unión de dos curvas. Significa que a lo largo de una trayectoria determinada en las uniones de las curvas no existen altibajos o cambios abruptos de trayectoria. Si se considera una curva compuesta por dos segmentos, la continuidad geométrica se definiría como sigue: si los dos segmentos de curva están unidos en un punto, entonces la curva tiene continuidad geométrica que se representa por G^0. Si las direcciones de las tangentes de cada segmento son iguales en el punto de unión, entonces se dice que la curva tiene continuidad geométrica G^1. Si los vectores tangente (primer derivada) de cada segmento son iguales tanto en magnitud como en dirección en el punto de unión, entonces se dice que la curva tiene continuidad geométrica C^1. Si los vectores resultado de la derivada n de ambos segmentos en el punto de unión son iguales, entonces se dice que la curva tiene C^n (continuidad geométrica *n*).

Para que exista continuidad geométrica en una cadena de curvas de Bèzier es necesario que el último punto del polígono de control de una curva sea el primero del polígono de control de la siguiente. Ya que la pendiente de la curva en sus extremos está determinada por el segmento de puntos de control inicial (o final, según sea el caso) la condición de continuidad geométrica está determinada por la posición del segmento final de la curva con respecto al inicial de la siguiente. Para cumplir con la condición de G^1 basta con que los tres puntos –que determinan la unión de los dos segmentos en cuestión- sean colineales, como se muestra en la figura 2.10.

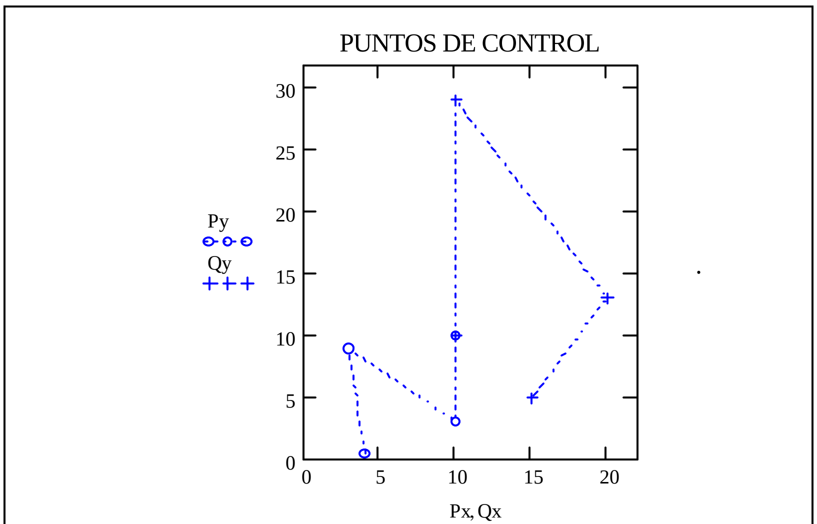

Fig. 2.10.- Ejemplo con vectores de puntos {P }y {Q} en el plano X-Y.

La condición de continuidad geométrica está determinada por la derivada de la curva y está dada por la ecuación 2.5. El resultado es la unión suave de dos curvas de la figura 2.11.

$$\frac{d}{du}B_{i,n}(u) = n\cdot\left[B_{i-1,n-1}(u) - B_{i,n-1}(u)\right] \qquad (2.5)$$

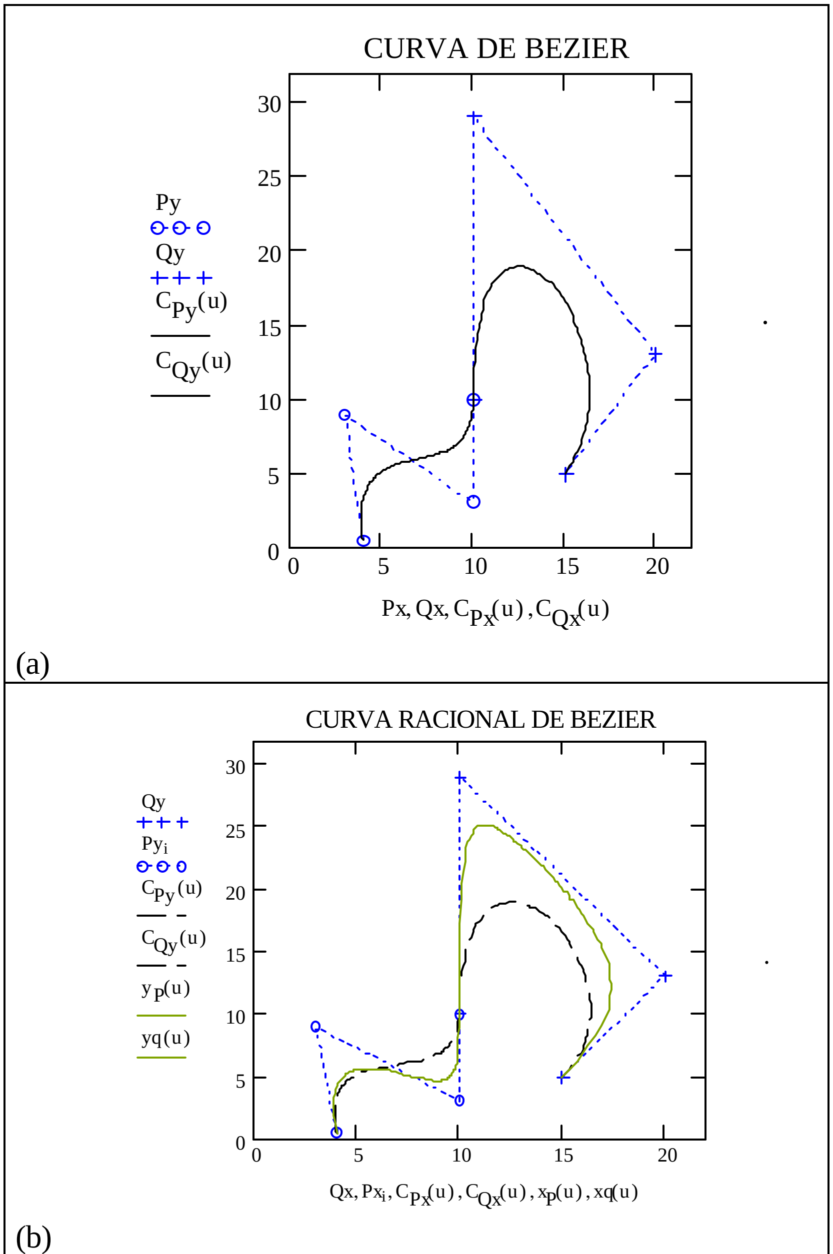

Fig. 2.11.- (a)Curvas de Bèzier de la figura 2.10. (b)curva resultante tras considerar el vector de pesos {$\mathbf{w_i}$}.

Condiciones de maquinado

Para efectos de la generación de código NC, el cálculo de puntos a lo largo de la curva se hace en función del radio de la herramienta a utilizar y considerando la longitud del arco del segmento. La longitud de arco *L* de cualquier curva paramétrica con límites 0<u<1 está definida por:

$$L = \int_0^1 \sqrt{\left(\frac{d}{du} Cx(u)\right)^2 + \left(\frac{d}{du} Cy(u)\right)^2}\, du \qquad (2.6)$$

Que es la distancia que tendrá que recorrer la herramienta a lo largo de la curva.

2.4 Resumen del Capítulo

En este Capítulo se expusieron los aspectos relativos a la estructura de control de una máquina-herramienta, los aspectos geométricos de la geometría de la pieza y las trayectorias de la herramienta que influyen sobre el maquinado. Se dejó en claro que la velocidad de avance y la capacidad del control de procesar bloques por adelantado deben ser un factor a considerar en los estudios de maquinado. Se dejó en evidencia que bajo estas condiciones, las pruebas de maquinabilidad simplificadas no garantizan la calidad del proceso. Esto sirve como justificante para que la atención de este estudio se concentre en la mejora de la velocidad programada de avance.

Se hizo énfasis en las diferencias generacionales de los controles en cuanto a su capacidad de hacer aproximaciones e interpolaciones para curvas complejas, exponiendo los errores como consecuencia de esto.

Se mostró una fase de cálculos para el maquinado de B-Splines enfocada en el caso de las curvas de Bèzier, ya que éstas son una clase de ellas. Esta matemática establece una de las principales diferencias en la velocidad de avance entre las generaciones de controles de máquinas-herramienta, por lo que la implementación de mejoras en términos de esta matemática resultará de una mejor competencia de los trenes productivos basados en máquinas-herramienta de la generación anterior.