

Generación de trayectorias de mecanizado con un número mínimo de puntos

J. L. Huertas Talón^(1 y 2), R. Gella Marín⁽¹⁾, C. García-Hernández⁽¹⁾, L. Berges-Muro⁽¹⁾, C. López-Gómez⁽¹⁾, J.J. Marín Zurdo⁽¹⁾, J. J. Garde Barace⁽²⁾, F. Valdivia Calvo⁽²⁾, J.C. Cisneros Ortega⁽³⁾

*(1) Dpto. de Ingeniería de Diseño y Fabricación de la Universidad de Zaragoza
jhuertas@Unizar.es*

(2) Dpto. de Fabricación Mecánica del Centro Público Integrado de Formación Profesional "Corona de Aragón" de Zaragoza

(3) Universidad Popular Autónoma del Estado de Puebla. México

Después de digitalizar un perfil o una superficie o generar puntos de la función que define el perfil o la superficie (si admiten una ecuación), se dispone de un elevado número de coordenadas de puntos agrupados según la estrategia seguida para la digitalización o la generación. Pero no todos los puntos tienen el mismo peso: en las zonas donde la curvatura es grande, se precisan más puntos que donde es pequeña. Aquí se presenta un método, basado en el error cordal máximo permitido, el cual facilita la selección de los puntos que aportan información significativa para la reconstrucción de la forma mediante un programa de CAD o su mecanizado en una máquina de CN con la menor información o número de puntos o líneas de programa CN. Para ello, es posible utilizar cualquier lenguaje de programación, habiendo utilizado en nuestro caso una macro de Excel y MATLAB.

Esto es igualmente interesante en máquinas de control numérico de alta velocidad como en equipos antiguos. Las primeras disponen de prestaciones del tipo "look ahead", puesto que el número de bloques de programa CN que se leen por adelantado, cuanto menor cantidad de líneas se dispongan para la misma precisión requerida, la velocidad de proceso será mayor. Los segundos se ven beneficiados del menor número de líneas a procesar. En este sentido hay que recordar que un sistema es tan rápido como el más lento de sus elementos.

1. INTRODUCCIÓN

Hay varios métodos para digitalizar un perfil o una superficie:

- a) Utilizando una fresadora de CN con un palpador acoplado a su cabezal.
- b) Con la máquina de medir de coordenadas. El sistema es semejante al anterior, pero con más precisión.
- c) Sustituyendo el palpador mecánico de la máquina de coordenadas por un láser [3].
- d) Utilizando directamente un mecanismo Láser Tracker.

También, hay varios métodos para evaluar los puntos que se deben enviar a una máquina CN para mecanizar la geometría en cuestión. En las figuras siguientes se muestran las diversas posibilidades aplicadas a la evolvente del flanco de un engranaje:

- a) Se trata de obtener puntos equidistantes en cuanto a la longitud de arco. O sea, todos los arcos tienen la misma longitud, figura 1. Es el método que se suele utilizar para obtener los puntos de verificación, figura 2.

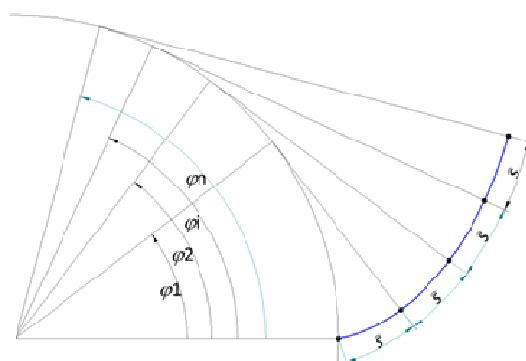


Figura 1. Igual longitud de arco

En efecto, la figura 2 muestra la posición del centro del palpador de una máquina de coordenadas tras escanear un engranaje para su verificación. Se observa que, prácticamente la distancia, medida sobre la línea de palpado, es la misma (excepto en los cambios bruscos de dirección).

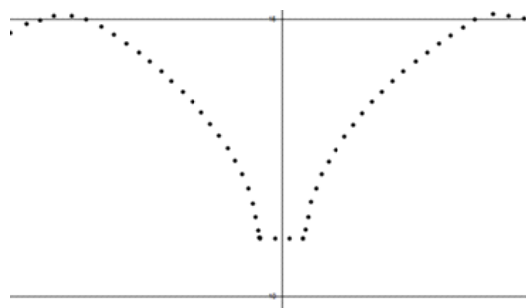


Figura 2. Posición del centro de un palpador tras verificar un engranaje (los puntos fueron exportados a una hoja electrónica)

b) La flecha o error cordal es el mismo. Este método es interesante cuando lo que se desea es la fabricación. Este procedimiento se ilustra en la figura 3.

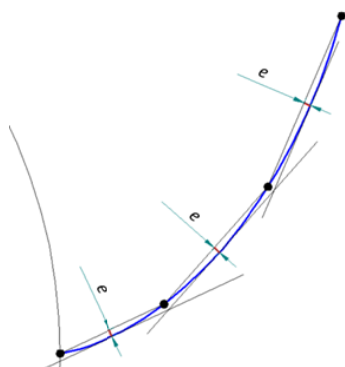


Figura 3. Igual error cordal

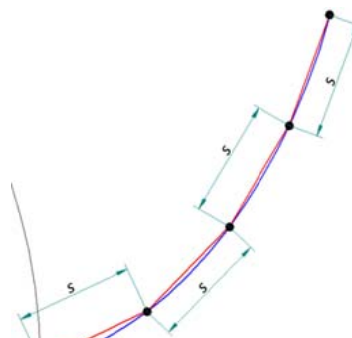


Figura 4. Igual error cordal

c) La longitud de la cuerda es la misma, figura 4. Cuando no hay mucha diferencia de curvatura el método es semejante al primero y puede sustituirle. No se va a utilizar.

d) Utilización de las ecuaciones paramétricas o explícitas para obtener puntos que, por ejemplo, faciliten la trayectoria que debe seguir una máquina CN. No es aconsejada, pues precisa un incremento muy pequeño del parámetro o de la variable independiente para obtener una precisión aceptable. Sin embargo, es un método muy utilizado por su sencillez, pero tiene el inconveniente de que en zonas donde la curvatura es grande se pierde precisión y, al contrario, en zonas en que la curvatura es pequeña hay un exceso de

precisión. Las figuras 5 y 6 ilustran un modelo con ecuación paramétrica y otro con ecuación explícita. Menciona aparte, merece la figura 7, en que se representa una ecuación paramétrica en forma de cociente de polinomios.

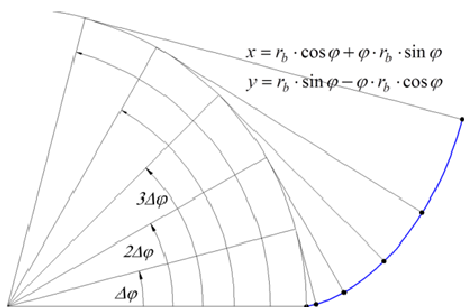


Figura 5. Aproximación de la evolutiva de círculo por incrementos iguales del parámetro

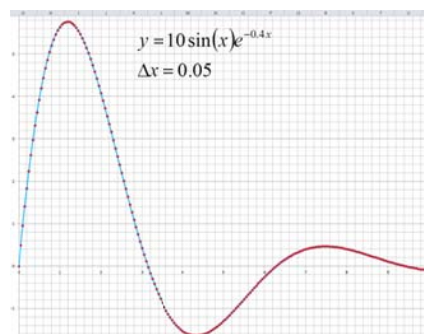


Figura 6. Aproximación de una ecuación explícita

En todos los casos, cuando el número de puntos escaneado es muy alto, conviene realizar un filtrado de los mismos para descartar aquellos que aportan menor peso para la identificación de la forma digitalizada.

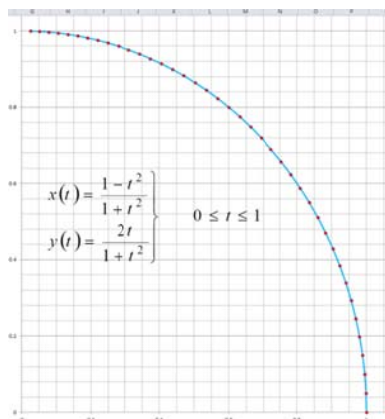


Figura 7. Aproximación de una ecuación paramétrica (círculo) de en forma de fracción de polinomios

Los puntos que aportan más peso son los situados en las zonas de más curvatura y los que menos, los situados en las zonas de menor curvatura. En la figura 8, se muestra un perfil aerodinámico que pone de manifiesto lo comentado.

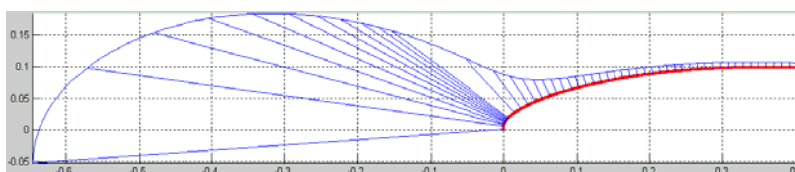


Figura 8. Viendo el peine de curvatura, los puntos de tramo próximos al borde de ataque aportan más peso a la gráfica

Seguidamente se describe un algoritmo para utilizar solamente los puntos más significativos o que aportan más peso al trazado de una curva. Con ello, además de la limpieza del proceso de reconstrucción, se va a ganar en velocidad de procesamiento en el sistema informático y en el control numérico que debe realizar el mecanizado.

2. CONSIDERACIONES INICIALES

En el desarrollo de nuestras experiencias, mecanizando hélices de aeromodelos, midiendo engranajes, escaneando superficies, observamos que sería interesante disponer de un

algoritmo que seleccionara la cantidad y calidad de los puntos obtenidos tras escanear con la máquina de coordenadas o el sistema de visión artificial, a la vez que también se precisan menos puntos para enviar a la máquina CN (especialmente en controles más antiguos) o que no disponen de las prestaciones “look ahead”.

Observando la figura 8, correspondiente a un perfil aerodinámico [4], se llega a la conclusión de que en las proximidades del borde de ataque serán precisos más puntos para su definición que en puntos del extradós. Ello es debido a que la curvatura en el primero es mayor que en el segundo, como se aprecia al observar al peine de curvatura representado. A la misma conclusión se llega analizando las figuras 1 a 5, correspondientes a un perfil de engranaje, y la figura 6 que representa una función explícita, así como la figura 7 donde la ecuación paramétrica es un cociente de dos polinomios.

En la figura 9, se muestran dos circunferencias, con dos aproximaciones poligonales. Cuantos más lados tiene el polígono más aproximado será éste a la circunferencia. Este hecho es utilizado en diversos programas de CAD/CAM para indicar la exactitud en la representación de curvas [5]

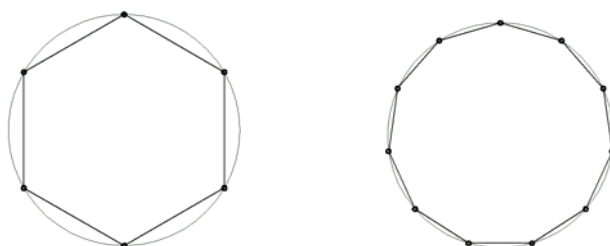


Figura 9. El polígono de la derecha se aproxima mejor a la circunferencia.

Pero ¿qué hacer con los puntos obtenidos? Seguidamente se verá que se pueden filtrar con un algoritmo, que utiliza el coseno del ángulo de dos vectores [6] y que se materializará en Excel o MATLAB para dejar un número muy inferior de puntos, que serán más fáciles de manipular en la reconstrucción o mecanizado CN de la pieza. A la vez que sirve, una vez obtenidos multitud de puntos, para mecanizar una pieza utilizando solamente aquellos que aportan un peso significativo en cuanto a la curvatura se refiere, y que como se ha experimentado se obtiene una disminución de los tiempos de mecanizado.

3. PRINCIPIO MATEMÁTICO

La relación que hay entre el número de segmentos de la poligonal que aproxima la circunferencia y el error cometido en la figura 9 se puede deducir de la figura 10, donde e representa el error cordal.

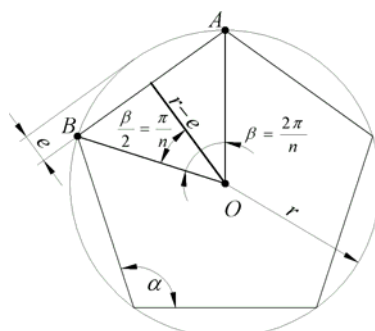


Figura 10. Principio matemático

El error está relacionado con el radio de la circunferencia por la expresión:

$$\cos \frac{\beta}{2} = \frac{r - e}{r} \Rightarrow e = \left(1 - \cos \frac{\beta}{2}\right)r \tag{1}$$

Obtenida a partir del triángulo OAB de la figura 10.

Por otro lado, el ángulo que forman los lados del polígono está relacionado con el ángulo α que forman los radios por:

$$\beta + \frac{\alpha}{2} + \frac{\alpha}{2} = \pi \Rightarrow \alpha = \pi - \beta \tag{2}$$

Con lo que:

$$\cos \alpha = \cos(\pi - \beta) = -\cos \beta = 2 \cos^2 \frac{\beta}{2} - 1 \tag{3}$$

Es decir, el error cordal finalmente está relacionado con el coseno del ángulo α , el cual, a su vez, se puede obtener a partir del ángulo que forman las cuerdas o lados del polígono que aproximan la circunferencia.

$$\cos \alpha = 2 \left(1 - \frac{e}{r} \right)^2 - 1 \tag{4}$$

Si ahora los lados del polígono se convierten en vectores (figura 6), resulta que el ángulo formado por los lados está definido por el producto escalar de los vectores:

$$\cos \alpha = \frac{\mathbf{v}_{12} \cdot \mathbf{v}_{23}}{|\mathbf{v}_{12}| \cdot |\mathbf{v}_{23}|} \tag{5}$$

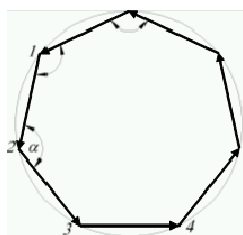


Figura 11. Vectores delimitados por puntos de escaneo o generados por cualquier método analítico.

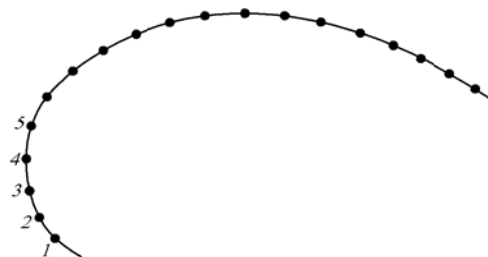


Figura 12. Sucesión de puntos escaneados

Como se ha indicado en la figura 11, los puntos que delimitan los vectores pueden corresponder a puntos escaneados. Así, pues, a la vista de las figuras 13 y 14, si los puntos escaneados están lo suficientemente próximos, cada tres puntos determinan una circunferencia, cuyo centro coincide con el centro de curvatura y su radio es el de curvatura en el entorno próximo a los puntos. De este modo, tendremos las circunferencias definidas por los puntos (1, 2, 3), (2, 3, 4) y así sucesivamente.

De esta forma, es posible preparar un algoritmo que estudie los puntos (1, 2, 3). Si el coseno evaluado por la expresión (5), es superior a un valor el cual determina (4), podemos descartar el punto intermedio 2. Si no se descarta este punto, se pasa a verificar los puntos (2, 3, 4). En el caso de desechar el punto 2, se pueden dar varias posibilidades de continuar con el algoritmo.

- a) Repetir lo anterior con los puntos (1, 3, 4) y así sucesivamente. El punto que se puede ahora desechar es el 3.
- b) Realizar el algoritmo con los puntos (3, 4, 5), siendo el objetivo desechar o no el punto 4.
- c) Utilizar el método primero, pero el punto eliminado desaparece de la serie. O sea, el punto 3 pasa a ser el punto 2, el punto 4 pasa a ser el 3.

Tras diversas comprobaciones, la variante que mejor resultado ha dado ha sido la última, especialmente al emplear una macro realizada para Excel en la que la fila que contiene las coordenadas del punto desechado se elimina. Cuando el número de puntos es elevado resulta más rápido utilizar MATLAB, pues basta eliminar el elemento correspondiente en el vector que los contiene.

Lo anteriormente comentado, permite no solamente hacer que el algoritmo sea recursivo, además facilita ir avanzando sin tener en cuenta si el punto fue o no eliminado (pues de ello se ha encargado Excel o MATLAB).

Así mismo, el algoritmo es válido tanto en 2D como en 3D:

En el primer caso, 2D, solamente hay que trabajar con dos coordenadas. Los tres puntos determinan la circunferencia sin más condición que la ecuación de la misma, siendo las incógnitas las coordenadas del centro y el radio.

En el caso de 3D, hay 4 incógnitas, las coordenadas del centro (x_c, y_c, z_c) y el radio. La cuarta condición es que el centro pertenece al plano determinado por los tres puntos en estudio, siempre que no estén alineados.

También, se considera el caso de que los puntos están sensiblemente alineados. En tal circunstancia, el punto 2 es eliminado sin entrar en el algoritmo.

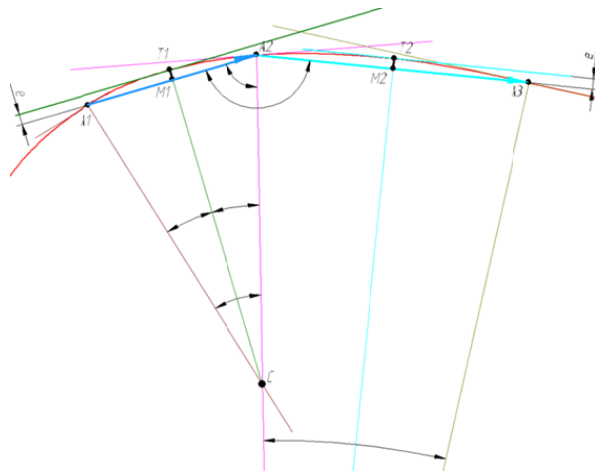


Figura 13. Si los puntos son próximos, error cordal e pequeño, tres puntos seguidos forman una circunferencia (su centro de curvatura está muy próximo)

El algoritmo se muestra comentado en el siguiente organigrama.

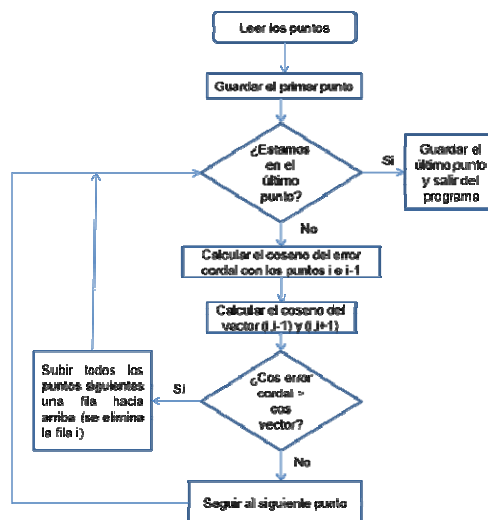


Figura 14. Organigrama del algoritmo de eliminación de un punto.

4. REALIZACIÓN PRÁCTICA

A continuación se muestran una serie de ejemplos de aplicación práctica. Se puede utilizar el procedimiento tanto para representar en CAD con una menor cantidad de puntos para una resolución dada, como para el mecanizado con el error cordal admisible. A continuación se presentan ejemplos 2D y 3D mecanizados con CN como para su representación en programa de CAD.

4.1. Función amortiguada 2D.

Se representa la función amortiguada en el intervalo 0 a 80 mm.

$$y = 50 \sin\left(\frac{x}{4}\right) e^{-0.05x} \tag{6}$$

Si se realiza a incrementos $\Delta x=0.0001$ hacen falta 800001 puntos, en un fichero binario que guarde ambas coordenadas con precisión doble son necesarios 8 bytes por cada número. Ello hace que el fichero adquiera un tamaño de 1242kb (dados por el explorador de Windows). La misma función con un error de arco de 0.001 mm MATLAB facilita 144 puntos, figura 15, el fichero de los mismos solamente ocupa 4kb (siempre según el explorador, pero es sencillo realizar el cálculo exacto).

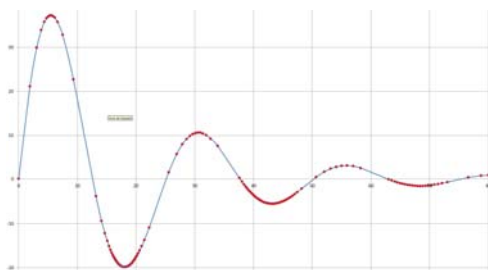


Figura 15. Representación de la función $y=50*\sin(x/4)*\exp(-0.05*x)$ con un error cordal de 1 μm

4.2. Espiral de Arquímedes 2D

En este caso se realizó con MATLAB la obtención de los puntos de la espiral separados por una longitud de arco de 0.05 mm, siendo el paso de la espiral 10 mm, la longitud total de la espiral que se ha mecanizado fue de 500 mm. La tabla siguiente muestra los resultados obtenidos en una fresadora FANUC Robodrill del Centro Integrado de F.P. “Crona de Aragón” (se desactivó la función de suavizado). Con equipos menos modernos, los resultados fueron más divergentes.

Avance programado mm/min	Tiempo/avance real		
	Error cordal 1 μm (210 líneas CN)	Arco de 0.05 mm (10000 líneas CN)	Diferencia de tiempo
8000	4/7500	45/667	41
4000	8/3750	45/667	37
2000	15/2000	44/682	29

Tabla 1. Posición, velocidad y aceleración angulares del sólido 1

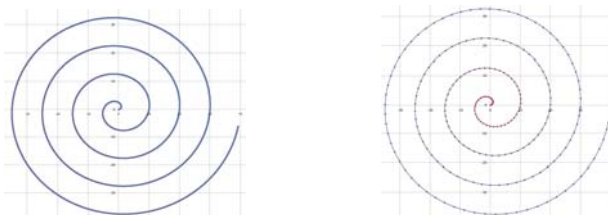


Figura 16. Comparación entre espirales con longitud de arco de 0.05 mm y error cordal 0.001 mm

La figura 16 a la izquierda muestra la espiral de Arquímedes realizada con una longitud de arco de 0.05 mm. Se observa que los puntos de representación se acumulan. A la derecha está la misma espiral pero con los puntos seleccionados para un error cordal inferior a 1 μm . En la figura 17 se observa el mecanizado de la espiral sobre un bloque de POM y una herramienta de punta esférica.



Figura 17. Mecanizado de un surco en espiral con las condiciones de la tabla 1

4.3. Superficie de un sillín de bicicleta

En la figura al pie se muestra una superficie: a la izquierda se han generado un total de 230911 puntos que se corresponden con otras tantas líneas de programa CN (además de la cabecera y pie de programa). A la derecha, para un error cordal de 0.01 mm los puntos se han reducido a solamente 74268 puntos, una tercera parte. Para un avance programado de 2000 mm/min ha supuesto una reducción de tiempo de mecanizado de 2,5 minutos estando activada la función de suavizado, cuando no se activa esta función el tiempo se reduce a la mitad.

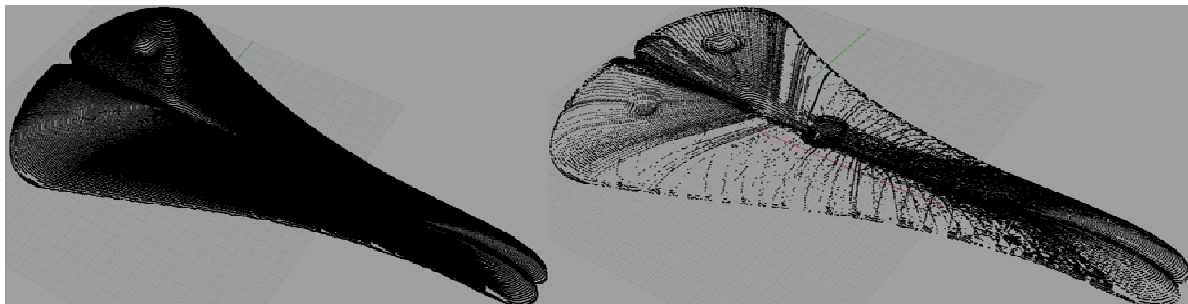


Figura 17. Superficies con alta y baja densidad de puntos para un mismo error cordal.

5. CONCLUSIONES

Se muestra un método que permite minimizar el número de puntos precisos para representar o para mecanizar tanto elementos 2D como 2D. En el caso de la representación, simplemente disminuyen los recursos precisos para “mover” los puntos en la pantalla. En el caso del mecanizado se obtiene una importante reducción de tiempo de mecanizado al aproximarse la velocidad de avance real a la programada. Por otro lado, el hecho de simplemente aumentar la velocidad de avance cuando se emplean los puntos sin optimizar, no supone una disminución de tiempo una vez sobrepasado un valor; es más, puede incluso suponer un incidente en la máquina.

6. AGRADECIMIENTOS.

El presente trabajo corresponde a parte del proyecto “El conocimiento como motor de la fabricación mecánica” subvencionado por el Ministerio de Educación Cultura y Deporte y el Fondo Social Europeo.

7. REFERENCIAS

- [1] M.W. Cho, T.I. Seo. Inspection Planning Strategy for the On-Machine Measurement Process Based on CAD/CAM/CAI Integration.
- [2] J.L. Huertas, F. Torres, F. Floren, V. Hernández. Obtaining a Program of Mechanizing of an Aerodynamic Profile Using Matlab. *The International Journal of Advanced Manufacturing Technology*. 2005, 19:607-617
- [3] F.J. Shiou, T. Pfeifer. Strategies and Processes to Measure the Dimensional Geometry of Sheet Metal Parts for Exact Laser Cutting. *The International Journal of Advanced Manufacturing Technology*. 2002, 19:106-116.
- [4] El de Metal Univers
- [5] Athulan Vijayaraghavan, Angela Sodemann, Aaron Hoover, J. Rhett Mayor, David Dornfeld. Trajectory generation in high-speed, high-precision micromilling using subdivision curves. *International Journal of Machine Tools & Manufacture* 50 (2010) 394-403.
- [4] E.N. Jacobs. K.E. Ward, R.M. Pinkerton. The characteristics of 78 related airfoil sections form tests in the variable-density wind tunnel. Report n° 460. NACA.
- [5] Solid Edge the CAD 3D of Siemens. Opciones de vista, suavidad de arco.
- [6] H.W. Corley, Jay Rosenberger, Wei-Chang Yeh, T.K. Sung. The cosine simplex algorithm. *The International Journal of Advanced Manufacturing Technology*. 2005, 10:106-116.