

TRABAJO FIN DE MÁSTER

---

# Análisis y optimización de errores cinemáticos en una máquina CNC de 5 ejes

---

Realizado por  
**Yasser Zekalmi Richa**

Para la obtención del título de  
Máster en Ingeniería Mecánica

Dirigido por  
José Antonio Albajez  
María José Oliveros



**Universidad**  
Zaragoza



Escuela de  
Ingeniería y Arquitectura  
**Universidad** Zaragoza






## DECLARACIÓN DE AUTORÍA Y ORIGINALIDAD

(Este documento debe remitirse a [seceina@unizar.es](mailto:seceina@unizar.es) dentro del plazo de depósito)

D./D<sup>a</sup>. YASSER ZEKALMI RICHA ,

en aplicación de lo dispuesto en el art. 14 (Derechos de autor) del Acuerdo de 11 de septiembre de 2014, del Consejo de Gobierno, por el que se aprueba el Reglamento de los TFG y TFM de la Universidad de Zaragoza,

Declaro que el presente Trabajo de Fin de Estudios de la titulación de Máster Universitario en Ingeniería Mecánica

 (Título del Trabajo)

ANÁLISIS Y OPTIMIZACIÓN DE ERRORES CINEMÁTICOS EN UNA MÁQUINA CNC DE 5 EJES

es de mi autoría y es original, no habiéndose utilizado fuente sin ser citada debidamente.

Zaragoza, 31/05/2023

Firmado por ZEKALMI RICHA, YASSER  
(AUTENTICACIÓN) el día 31/05/2023  
con un certificado emitido por AC DNIE  
005

Fdo:

# Índice

<b>1. Introducción</b>	<b>8</b>
<b>2. Equipo y elementos disponibles</b>	<b>9</b>
2.1. Máquinas CNC de 5 ejes . . . . .	9
2.2. Máquina analizada HAAS VF3 TRT160 . . . . .	11
2.3. Software CAM utilizado . . . . .	13
<b>3. Metodología</b>	<b>14</b>
3.1. Diseño CAD de la superficie . . . . .	17
3.2. Ajustes del CAM . . . . .	19
3.2.1. Geometría de herramienta . . . . .	19
3.2.2. Sistemas de coordenadas . . . . .	19
3.2.3. Estrategia de mecanizado . . . . .	20
3.3. Parámetros de mecanizado . . . . .	21
3.4. Extracción de trayectorias CAM . . . . .	22
3.5. Modelo cinemático de la máquina . . . . .	24
3.6. Errores geométricos en la máquina . . . . .	26
3.6.1. Errores debidos a las traslaciones de la máquina . . . . .	26
3.6.2. Errores debidas a las rotaciones de la máquina . . . . .	28
<b>4. Estudio de los errores</b>	<b>30</b>
4.1. Análisis de los errores de diseño CAD/CAM frente a la superficie Teórica . . . . .	30
4.2. Errores del postprocesador. Comprobación del modelo cinemático y extracción de decimales. . . . .	33
4.3. Métricas de medida de error . . . . .	34
4.4. Análisis de los errores geométricos . . . . .	35
4.4.1. Errores según ejes de la máquina . . . . .	36
<b>5. Descripción del algoritmo de enjambre de partículas</b>	<b>39</b>
<b>6. Resolución del problema</b>	<b>40</b>
6.1. Funcionamiento del algoritmo . . . . .	40
6.2. Función coste u objetivo . . . . .	42
6.3. Lectura de la herramienta en Matlab y detección de colisiones. . . . .	44
<b>7. Resultados y discusión</b>	<b>47</b>
7.1. Situación inicial . . . . .	47
7.2. Calibración del algoritmo . . . . .	48
7.2.1. Parámetros del algoritmo y hardware . . . . .	48
7.2.2. Soluciones iniciales de las partículas . . . . .	48
7.3. Control de trayectoria . . . . .	49
7.3.1. Cálculo del error mínimo sin control de la trayectoria . . . . .	49
7.3.2. Cálculo del error mínimo con control de la trayectoria . . . . .	51
7.4. Análisis de métricas . . . . .	55
7.5. Elección de la mejor estrategia . . . . .	56
<b>8. Conclusiones y trabajo futuro</b>	<b>58</b>

## Índice de figuras

1.	(Modelos de máquinas 5 ejes. a) tipo mesa giratoria/inclinable; (b) tipo husillo giratorio/inclinable; (c) tipo mesa/husillo inclinable. [8] . . . . .	10
2.	Máquina HAAS de 5 ejes con el Trunnion. Fuente HAAS . . . . .	11
3.	Características de la máquina y del trunnion. Fuente HAAS . . . . .	12
4.	Esquema de comparación de posición y orientación de la herramienta, objetivo, real y compensada. Elaboración propia. . . . .	14
5.	Esquema general de la metodología seguida, (Elaboración propia) . . . . .	16
6.	Diseño del paraboloide en SolidWorks. (Elaboración propia). . . . .	17
7.	Sistemas de coordenadas existentes para el mecanizado de la pieza. (Elaboración propia). . . . .	19
8.	Sistemas de coordenadas utilizados. (Elaboración propia). . . . .	20
9.	Estrategia de mecanizado CAM. (Elaboración propia). . . . .	21
10.	Esquemas de la máquina estudiada (HAAS) . . . . .	24
11.	Errores geométricos debidos a traslaciones XYZ. (Elaboración propia). . . . .	27
12.	Errores geométricos debidos a giros AC. (Elaboración propia). . . . .	28
13.	Metodología seguida y errores a tener en cuenta. (Elaboración propia). . . . .	30
14.	Esquema del error de diseño CAD comparado con la superficie teórica. (Elaboración propia). . . . .	31
15.	Comparación del diseño CAD/CAM con la superficie teórica. (Elaboración propia). . . . .	32
16.	Comparación de puntos teóricos vs Superficie CAM, en GeoMagic. (Elaboración propia). . . . .	32
17.	Diferencias entre los GOTO y el resultado de pasar los POST por el modelo cinemático de la máquina. (Elaboración propia). . . . .	33
18.	Trayectoria zig-zag para el mecanizado del paraboloide. (Elaboración propia). . . . .	35
19.	Cuadrantes del paraboloide a analizar. (Elaboración propia). . . . .	35
20.	Errores debidos a movimientos de máquina en X-Y-Z. (Elaboración propia) . . . . .	36
21.	Errores debidos a movimientos de máquina en X-Y-Z-A. (Elaboración propia) . . . . .	36
22.	Errores debidos a movimientos de máquina en X-Y-Z-C. (Elaboración propia) . . . . .	37
23.	Errores debidos a movimientos de máquina en X-Y-Z-A-C. (Elaboración propia) . . . . .	37
24.	Bandada de aves, comportamiento de la naturaleza en el que se basa el algoritmo. Fuente PSO . . . . .	39
25.	Esquema del movimiento de las partículas en el algoritmo. [27] . . . . .	41
26.	Esquema del algoritmo implementado. (Elaboración propia) . . . . .	42
27.	Herramienta diseñada en CAD y remallada en Rhino finalmente. (Elaboración propia). . . . .	44
28.	Herramienta leída como conjunto de puntos en Matlab. (Elaboración propia). . . . .	45
29.	Esquema de la punta de herramienta optimizada frente a la objetivo, distancia punta optimizada y punta objetivo (rojo), entrada en Z (amarillo). (Elaboración propia). . . . .	46
30.	Posiciones de máquina y trayectoria extraídos del CAM. (Elaboración propia). . . . .	47
31.	Errores geométricos (mm) inicialmente, previo a la optimización. (Elaboración propia). . . . .	47
32.	Resultados con solución inicial post (CAM) sin control de incrementos. (Elaboración propia). . . . .	49
33.	Resultados con solución inicial gbest sin control de incrementos. (Elaboración propia). . . . .	50

34.	Resultados de posiciones de máquina con solución inicial el <b>POST</b> del CAM, controlando incrementos de los ejes lineales. (Elaboración propia).	51
35.	Resultados de posiciones de máquina con solución inicial el <b>POST</b> del CAM, controlando incrementos de los ejes de rotación. (Elaboración propia).	52
36.	Resultados de posiciones de máquina con solución inicial el <b>POST</b> del CAM, controlando incrementos de todos los ejes. (Elaboración propia).	52
37.	Resultados de posiciones de máquina con solución inicial el <b>GBEST</b> , controlando incrementos de los ejes lineales. (Elaboración propia).	53
38.	Resultados de posiciones de máquina con solución inicial el <b>GBEST</b> , controlando incrementos de los ejes de rotación. (Elaboración propia).	53
39.	Resultados de posiciones de máquina con solución inicial el <b>GBEST</b> , controlando incrementos de todos los ejes. (Elaboración propia).	54
40.	Comparación de consignas CAM vs Optimizada. (Elaboración propia).	56
41.	Comparación de errores geométricos CAM vs Optimizada. (Elaboración propia).	57

## Índice de tablas

1.	Comparación del código extraído del CAM, antes de realizar la preparación. (Elaboración propia).	23
2.	Comparación del código extraído del CAM, después de la preparación. (Elaboración propia).	23
3.	Tabla con métricas sobre los 4 cuadrantes, estando todos los errores en mm. (Elaboración propia).	38
4.	Resultados de las métricas elegidas	55
5.	Comparación de errores por cuadrante CAM vs Optimizada. (Elaboración propia).	57

## Resumen

La fresadora de 5 ejes es una herramienta clave en la industria manufacturera moderna, ya que permite realizar operaciones de mecanizado de alta precisión en componentes complejos. Sin embargo, debido a la naturaleza inherente de los sistemas mecánicos, se producen errores geométricos que afectan la precisión y calidad de los mecanizados realizados por la fresadora CNC de 5 ejes. Estos errores pueden deberse a diversas fuentes, como por ejemplo, la rigidez del sistema, desalineaciones de los ejes, holguras mecánicas o errores en los sistemas de medición, entre otros.

El presente trabajo se centra en la determinación de la influencia de los diferentes errores geométricos presentes en una fresadora de 5 ejes a partir de las trayectorias propuestas por un software de mecanizado CAM. Estas presentan la peculiaridad de no tener una solución única, al verse influenciado el error de cada una de las coordenadas [XYZ] por la combinación de los valores de los parámetros de movimiento de los 5 ejes de la máquina [XYZAC].

De manera adicional a este objetivo principal se va a desarrollar también una metodología de optimización que permita reducir los errores geométricos mejorando así la precisión de la fresadora. Esto es, se realiza un análisis de los errores geométricos presentes en el sistema y se calculan unas posiciones optimizadas de la máquina que logren el menor error de mecanizado en la realidad.

En contraste con las máquinas CNC de 3 ejes, en las que la relación cinemática es directa y es más fácil corregir éstos errores, las máquinas CNC de 5 ejes, debido a los dos giros adicionales, aportan una mayor complejidad cinemática presentando una combinación no lineal y entrelazada con los distintos parámetros de la máquina y sus diferentes errores geométricos.

Para corregir los errores es necesario utilizar técnicas de modelado, simulación y optimización, logrando encontrar la combinación óptima de parámetros de máquina que minimice los errores geométricos en la pieza. Los resultados obtenidos proporcionan una base sólida para mejorar la precisión de la máquina, lo cual tiene un impacto significativo en la calidad de los mecanizados y la eficiencia de los procesos de fabricación en la industria. En los resultados obtenidos, se logra reducir los errores geométricos más de cuatro órdenes de magnitud, obteniendo así con ello una metodología eficaz y completamente automatizada para la mejora de la calidad de mecanizado en la máquina.

La estrategia de trabajo desarrollada empieza por el diseño de la geometría de la pieza en [SolidWorks](#) y se usa [Rhino](#) para mallar la herramienta de mecanizado. Se definen los ajustes del proceso de mecanizado de la pieza en [NX CAM](#), obteniendo información del movimiento que hace la máquina para mecanizar dicha pieza. Después se programan subrutinas en python (entorno [Jupyter](#)) que sirven para preparar la toma de datos automática de la información extraída del CAM. El último paso es el análisis y optimización de los errores geométricos que tiene la máquina al mecanizar la pieza diseñada, lo cuál se hace en [Matlab](#).

## 1. Introducción

Las fases de la fabricación de una pieza comienzan normalmente por el diseño en CAD (Computer Aided Design) de la pieza a mecanizar. Después se importa a un software especializado en fabricación CAM (Computer Aided Manufacturing), en éste se definen una gran cantidad de parámetros inherentes al proceso de fabricación y máquina. Como pueden ser, escoger una herramienta, definición de una estrategia de mecanizado, profundidades de corte, al igual que avances y velocidades del husillo.

A continuación se extraen del CAM unas instrucciones que se llevan a la máquina de mecanizado CNC, dónde se indica cuales son las posiciones que debe mover la máquina para seguir la trayectoria del mecanizado. Hay una gran variedad de tipos de máquinas en el mecanizado convencional y se clasifican principalmente por el número de posiciones que pueden mover sus ejes. Las más comunes son las máquinas de 3 ejes que permiten las traslaciones en los tres ejes del sistema de coordenadas cartesianas. Existen en el mercado máquinas de mayor número de ejes, como es la máquina estudiada en este proyecto, la cual tiene un total de 5 ejes, permitiendo así dos giros adicionales alrededor de dos ejes cartesianos. Ésta máquina permite una mayor flexibilidad, reducción de tiempos y permite mecanizar piezas muy complejas. Las cuáles no habrían sido posibles sin esos dos grados de libertad adicionales.

Las máquinas CNC, al igual que cualquier mecanismo que se mueve, presentan un conjunto de errores. Éstos pueden ser principalmente dinámicos y geométricos. Los dinámicos surgen del exceso de vibraciones producidas por las diferentes fuerzas que experimenta la máquina. Los geométricos son los que ocurren cuando los ejes de la máquina CNC no se mueven a las posiciones exactas deseadas. Estos pueden ser causados por desgastes en los sistemas de transmisión, falta de rigidez en las estructuras, desalineaciones de los ejes, falta de perpendicularidad o paralelismo, entre otros. Es necesario estudiar éstos últimos por su magnitud y especialmente en máquinas de 5 ejes, debido a que el incremento de los giros adicionales proporciona errores adicionales a tener en cuenta. Los errores geométricos serán objeto de estudio del presente trabajo.

En máquinas CNC de tres ejes la posición de XYZ de cada punto está asociada directamente con el eje de movimiento X, Y, Z de la máquina. Sin embargo, en MH de 5 ejes para llegar a un punto X Y Z, se puede llegar de diferentes formas combinando los diferentes ejes de movimiento, lo que hace que existan múltiples opciones para llegar a un mismo punto.

De entre todos los parámetros que permite ajustar el CAM, en ninguno se incluye la posibilidad de corrección de estos errores geométricos. Con el presente trabajo se obtiene un post-procesador posterior propio que corrige los movimientos de la máquina teniendo en cuenta sus errores geométricos y minimizándolos.

## 2. Equipo y elementos disponibles

### 2.1. Máquinas CNC de 5 ejes

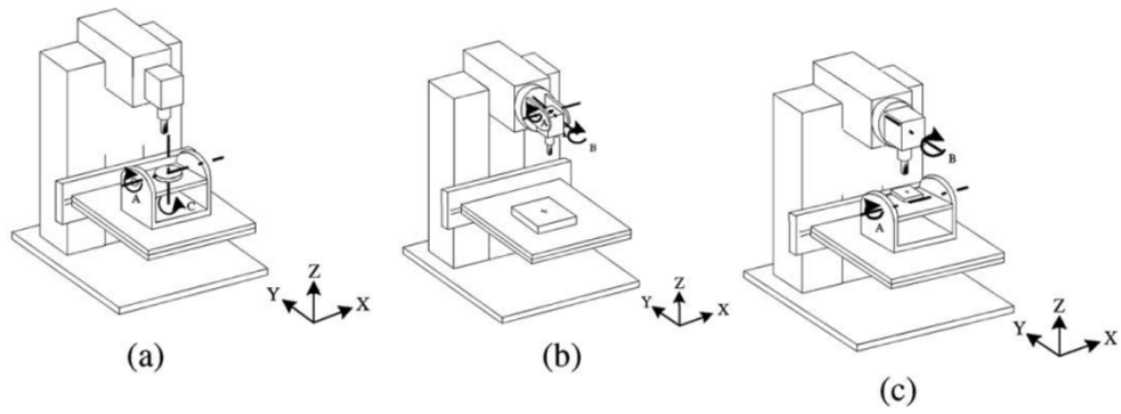
Las máquinas CNC de 5 ejes representan una evolución significativa en la industria del mecanizado. Estas máquinas son capaces de realizar operaciones complejas en cinco direcciones diferentes, lo que les proporciona una gran flexibilidad y versatilidad. Los tres ejes lineales (X, Y, Z) permiten el movimiento tradicional hacia adelante y hacia atrás, hacia los lados y hacia arriba y abajo, mientras que los dos ejes rotativos alrededor de cualquiera de los ejes lineales, permiten la rotación de la pieza y/o la inclinación de la herramienta en ángulos específicos. Los ángulos girados se denominan normalmente  $[A, B, C]$  para los giros en los ejes  $[X, Y, Z]$  respectivamente.

Esta capacidad de movimiento en múltiples direcciones simultáneamente ofrece numerosas ventajas. En primer lugar, reduce los tiempos de mecanizado al eliminar la necesidad de reposicionar la pieza de trabajo. Esto se traduce en una mayor eficiencia de producción y una reducción de los tiempos de mecanizado.

Además, el mecanizado en cinco ejes permite alcanzar áreas de difícil acceso y mecanizar características complejas en una sola operación, lo que contribuye a una mayor precisión dimensional y un mejor acabado superficial de las piezas mecanizadas. A continuación se discuten los principales tipos de máquinas de 5 ejes que hay en la actualidad:

- Modelo de tipo **mesa giratoria/inclinable**, figura 1 (a): Este tipo de máquina CNC de 5 ejes utiliza una mesa giratoria que puede inclinarse en diferentes ángulos para permitir el mecanizado en cinco ejes. La pieza de trabajo se sujeta en la mesa giratoria, que puede rotar alrededor de dos ejes, generalmente denominados eje A y eje C. Este tipo de máquinas suelen tener más precisión, al poder controlar los giros más cerca del sistema pieza.
- Modelo de tipo **husillo giratorio/inclinable**, figura 1 (b): Este modelo de máquina CNC de 5 ejes utiliza un husillo giratorio que puede inclinarse en diferentes ángulos. La mesa fija proporciona una base sólida para el mecanizado, lo que puede ayudar a mantener una mayor estabilidad y rigidez durante las operaciones de corte. En este tipo el cambio de herramienta suele ser más rápido.
- Modelo de tipo **mesa/husillo inclinable**, figura 1 (c): Este modelo combina las características de los dos modelos anteriores al tener tanto una mesa giratoria/inclinable como un husillo giratorio/inclinable. Posee amplio rango de movimientos con la combinación de ventajas e inconvenientes de los dos anteriores.

Los modelos de mesa giratoria/inclinable, husillo giratorio/inclinable y mesa/husillo inclinable son opciones populares que se adaptan a diferentes necesidades y aplicaciones en la industria del mecanizado. En los artículos [1-7] se habla del análisis y la optimización de errores geométricos en distintos tipos de fresadoras de 5 ejes y la importancia de ello para garantizar la precisión y calidad del mecanizado.



*Figura 1:* (Modelos de máquinas 5 ejes. a) tipo mesa giratoria/inclinable; (b) tipo husillo giratorio/inclinable; (c) tipo mesa/husillo inclinable. [8]

## 2.2. Máquina analizada HAAS VF3 TRT160

Este estudio tiene como objetivo mejorar la precisión y la calidad de mecanizado de la máquina [HAAS](#), disponible en el taller de fabricación de la Escuela de Ingeniería y Arquitectura. La máquina por sí sola es una fresadora de 3 ejes, es la versión [HAAS VF3](#), véase figura 2 (a). A ésta se le incorpora el Trunnion [TRT160](#), que es el elemento que le proporciona los dos giros adicionales, véase figura 2 (b). Las especificaciones de ambos elementos, como los planos y los diseños 3D de la máquina y trunnion, se pueden consultar en la página oficial de la [HAAS](#).

A continuación se describen las coordenadas de posicionamiento de la máquina. El cabezal de la herramienta lleva únicamente el movimiento en **Z** y gira entorno a él a una velocidad máxima de  $8100rpm$ , véase las especificaciones en la figura 3. Mientras que el resto de movimientos los lleva la pieza, en dónde la mesa posee dos carriles de movimiento de traslación en los ejes cartesianos **X** e **Y**.

Añadido a éstos los movimientos de rotación los proporciona el Trunnion que gira alrededor del eje **X** y **Z**, estando representados por los ángulos **A** y **C**, respectivamente. Se puede nombrar al movimiento cinemático que hace la máquina como **XYACFZ**, en el que los ejes a la izquierda de la letra **F** los lleva la pieza y los de la derecha los lleva la herramienta. En este trabajo todas las unidades de distancia están en milímetros y los ángulos se representan en grados.



(a) Máquina [HAAS VF3](#)



(b) Trunnion [TRT160](#)

Figura 2: Máquina HAAS de 5 ejes con el Trunnion. Fuente [HAAS](#)

MESA	METRIC
Longitud	1219 mm
Anchura	457 mm
Anchura de ranuras en T	15.90 mm to 16.00 mm
Distancia entre ejes con ranura en T	80 mm
Número de ranuras en T estándares	5
Peso máx. en mesa (distribuido uniformemente)	1588 kg

SPINDLE	METRIC
Potencia máx.	22.4 kW
Velocidad máx.	8100 rpm
Par máx.	122.0 Nm @ 2000 rpm
Par máx. con caja de engranajes opc.	339 Nm @ 450 rpm
Sistema de accionamiento	Inline Direct-Drive
Cono	CT40   BT40   HSK-A63

RECORRIDOS	METRIC
Eje X	1016 mm
Eje Y	508 mm
Eje Z	635 mm
Nariz del husillo a la mesa (~ máx.)	742 mm
Nariz del husillo a la mesa (~ mín.)	107 mm

DIMENSIONS - SHIPPING	METRIC
paleta doméstica	387 cm x 251 cm x 269 cm
paleta para exportación	333 cm x 233 cm x 257 cm
Peso	5316.0 kg
paleta doméstica con cinta transportadora	387 cm x 251 cm x 269 cm
paleta para exportación con cinta transportadora	389 cm x 233 cm x 257 cm
Peso con cinta transportadora	6260 kg

(a) Especificaciones máquina [HAAS VF3](#)

INDEXING - A (TILT)	METRIC
Rotación máx./paso	$\pm 120^\circ$
Resolución	$0.001^\circ$
Precisión ( $\pm$ )	$\pm 15$ arc-sec
Repetibilidad	10 arc-sec

INDEXING - B (ROTARY)	METRIC
Rotación/paso máximo (con control giratorio)	$999.999^\circ$
Resolución	$0.001^\circ$
Precisión ( $\pm$ )	$\pm 15$ arc-sec
Repetibilidad	10 arc-sec

PLATTER	METRIC
Diámetro de la plataforma	160 mm
Peso máx. en plataforma	36.3 kg
Volteo de pieza máx.	445 mm
Número de ranuras en T estándares	6 @ $60^\circ$

(b) Especificaciones [TRT160](#)

Figura 3: Características de la máquina y del trunnion. Fuente [HAAS](#)

### 2.3. Software CAM utilizado

El [NX CAM Pro v2022](#) es un software de fabricación asistida por computadora desarrollado por Siemens PLM Software. NX hace referencia a la plataforma de diseño y fabricación integrada y CAM es la abreviatura de Computer-Aided Manufacturing (Fabricación Asistida por Ordenador).

NX CAM se utiliza en el ámbito de la ingeniería mecánica para programar y controlar máquinas herramienta CNC (Control Numérico Computarizado) con el objetivo de fabricar piezas y componentes mecánicos de alta precisión. Este software permite a los ingenieros y fabricantes crear y optimizar procesos de mecanizado para producir piezas de manera eficiente y confiable. Las principales características y funcionalidades del programa son:

- Librerías de herramientas: Permite escoger entre una gran variedad de herramientas y portaherramientas de la librería disponible. También permite editar y crear herramientas completamente personalizadas.
- Estrategias de mecanizado: Ofrece una variedad de estrategias de mecanizado, como fresado, torneado, taladrado y electroerosión, que se pueden adaptar a diferentes tipos de piezas y materiales.
- Generación automática de trayectorias: Puede generar automáticamente trayectorias de herramientas eficientes a través de algoritmos avanzados. En los que permite simular la cantidad de material que se está mecanizando y calcular las trayectorias en función del material que falta por mecanizar.
- Control de máquinas múltiples: NX CAM admite la programación y control de máquinas herramienta CNC de 2 a 5 ejes, lo que lo hace adecuado para una amplia gama de aplicaciones.
- Postprocesamiento: Genera códigos G y M que son compatibles con la máquina CNC específica, lo que permite una fácil transferencia del programa al controlador de la máquina previo paso por el postprocesador.
- Simulación de mecanizado: Proporciona una función de simulación para verificar la trayectoria de la herramienta y asegurarse de que no haya colisiones entre la herramienta y la pieza, lo que ayuda a prevenir errores costosos.

En resumen, NX CAM es una potente herramienta de software que facilita la planificación, simulación y optimización de los procesos de mecanizado, lo que ayuda a los ingenieros y fabricantes a producir piezas de alta calidad con mayor eficiencia y precisión. Su integración con la plataforma de diseño NX también facilita el flujo de trabajo entre el diseño y la fabricación, lo que contribuye a una mayor productividad en el proceso de desarrollo de productos.

### 3. Metodología

La orientación de la herramienta frente a la pieza para máquina 3 ejes es única. Sin embargo, en máquinas de 5 ejes existen múltiples posibilidades a la hora de establecer la relación de posicionamiento relativo entre pieza y herramienta. Los programas CAM, a partir de estrategias definidas por el usuario, establecen de manera automática esta relación a partir de las trayectorias generadas.

Sin embargo, las herramientas CAM comerciales no permiten compensar los errores geométricos propios de cada máquina. Estos errores hacen que la posición real a la que llega la herramienta no sea la misma que la posición programada por el CAM. Actualmente los sistemas de control de máquina herramienta incorporan software de compensación de errores. Estos corrigen directamente la diferencia entre la coordenada teórica y la real sin tener en cuenta la trayectoria seguida por la herramienta ni la interferencia que hay entre pieza y herramienta.

En la figura 4 se representa la posición y orientación de la herramienta respecto de la pieza, en verde la posición objetivo, que es la que extrae el CAM, en rojo la posición y orientación real de la herramienta debido a los errores geométricos de la máquina y en morado la compensación que hace el control de máquina. Esta compensación aunque mejor que la posición real sin compensación, se puede traducir en la generación de defectos en la pieza mecanizada al no tener en cuenta la trayectoria y el contacto entre pieza y herramienta.

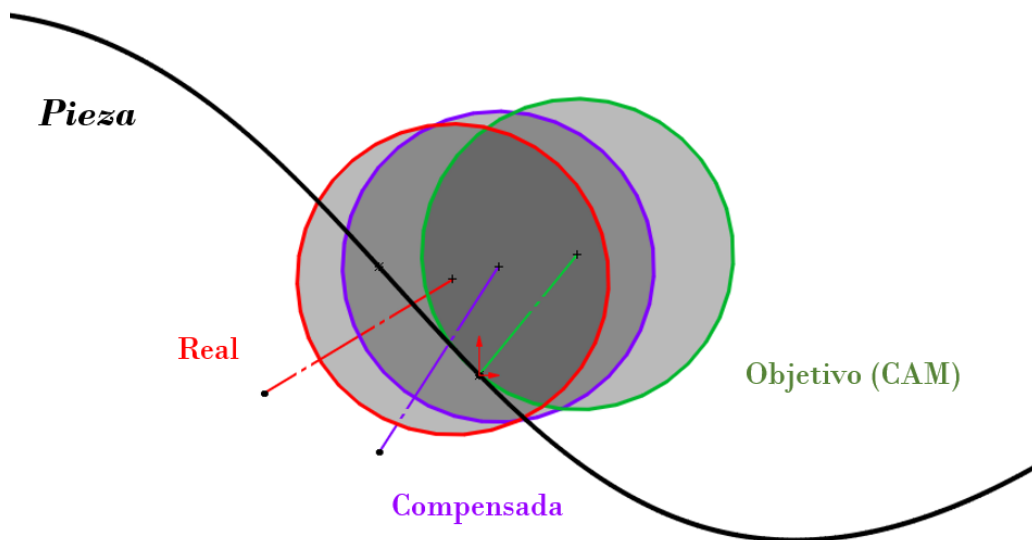


Figura 4: Esquema de comparación de posición y orientación de la herramienta, objetivo, real y compensada. Elaboración propia.

En este trabajo se estudia este problema y se introduce un nuevo método en su resolución. La nueva metodología proporciona un CNC corregido manteniendo lo máximo posible las trayectorias y cosignas del CNC creado por el CAM. Para ello se ha diseñado un post-procesador entre el CAM y la máquina que teniendo en cuenta su modelo cinemático y errores geométricos de la misma minimice su influencia mediante un procedimiento de optimización basado en un algoritmo de enjambre de partículas.

En primer lugar, se inicia el trabajo generando el modelo **CAD** de una superficie. Este modelo servirá como base para las simulaciones y análisis posteriores. A continuación, se emplea el programa de **CAM** para definir una estrategia CNC de 5 ejes, al igual que otros parámetros propios del mecanizado que se pueden ajustar. De lo que se obtiene un conjunto de puntos que forman la trayectoria de mecanizado, es lo que se llamará **GOTO**. También se extrae información sobre los movimientos que debe de realizar la máquina al mecanizar, que es el código que se lleva a la máquina HAAS, es lo que se llamará **POST**. Se relacionan mediante el postprocesador instalado en el CAM, el cuál es específico para cada máquina. Más adelante se explicarán en profundidad estos conceptos y como se relacionan matemáticamente.

Para automatizar la toma de datos del CAM se preparan una serie de scripts programados en python tanto el GOTO como el POST e información adicional necesaria, como pueden ser los puntos de contacto de herramienta-pieza. De esta manera se puede sacar información de cualquier estrategia, ajuste o cambios que se vayan realizando en el CAM de forma completamente automatizada.

Una vez preparados los datos, se procede a estudiar en **Matlab** los errores geométricos presentes en la máquina HAAS y otros debidos a la combinación del diseño CAD y CAM. Los errores geométricos se representan en forma de múltiples matrices de transformación, cuyas aportaciones combinadas hacen que sea un problema no lineal complejo, no habiendo solución analítica directa.

Para abordar este problema, se propone la implementación de un algoritmo de optimización basado en enjambres de partículas. Como se ha comentado previamente, esto permitirá crear un postprocesador propio diseñado matemáticamente en Matlab, el cuál aplicará una corrección posterior de los movimientos de la máquina, minimizando su error geométrico real. Se obtendrán diversas soluciones para distintos casos de estudio y se evaluará la opción más óptima para mejorar la calidad superficial de la pieza a estudiar. En la figura 5 se puede ver un esquema general de la metodología seguida.

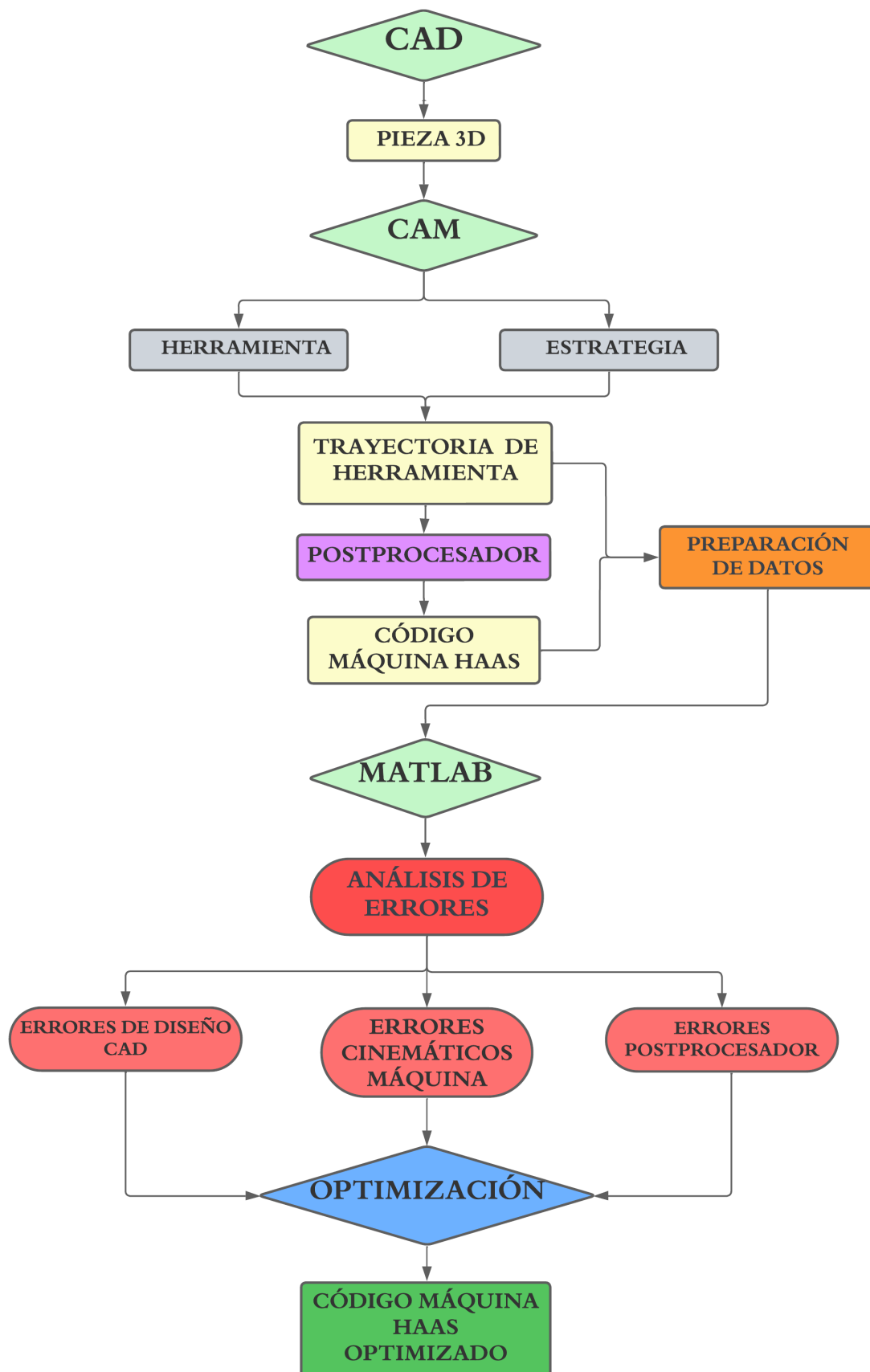


Figura 5: Esquema general de la metodología seguida, (Elaboración propia)

### 3.1. Diseño CAD de la superficie

La superficie elegida es un paraboloide hiperbólico, se ha escogido ésta ya que es una geometría controlada y que se puede diseñar fácilmente con 4 puntos de control. Añadir que se tiene la superficie matemáticamente definida, pudiendo obtener cualquier punto y vector normal en la misma. Se han sacado las ecuaciones de la superficie del artículo [9]. La ecuación paramétrica de la superficie en coordenadas propias es la 1, a la que hay que añadir la limitación en el plano  $\mathbf{XY}$ , dónde los puntos válidos son aquellos que están a una distancia menor de 40mm del centro, siendo éste el que se representa en el sistema de coordenadas de la superficie  $X'Y'Z'$ . La superficie se moverá 150mm y este centro pasará a estar en las coordenadas  $[0, 0, 150]mm$ , del sistema de coordenadas de pieza  $SCP$ , véase figura 6 (b). Los parámetros  $u$  y  $v$  indican un sistema de coordenadas referido a la superficie, véase figura 6 (a). Toman valores entre 0 y 1, con la ecuación 1 se devuelven coordenadas  $[X, Y, Z]$  de puntos en la superficie.

$$\mathbf{S}(u, v) = \begin{pmatrix} 96u - 48 \\ 96v - 48 \\ 144uv - 84u - 84v + 73 \end{pmatrix} \quad (1)$$

Para diseñar la superficie, se ha usado [SolidWorks](#). Para el diseño de la pieza se crean los 4 puntos de control recomendados por el artículo [9] y se unen mediante líneas siguiendo el orden en la figura 6 (a). Después se hace una operación de superficie relleno la que se eligen las 4 líneas y finalmente se recorta por la superficie lateral de un cilindro de diámetro 40mm centrado en el origen, véase figura 6 (b).

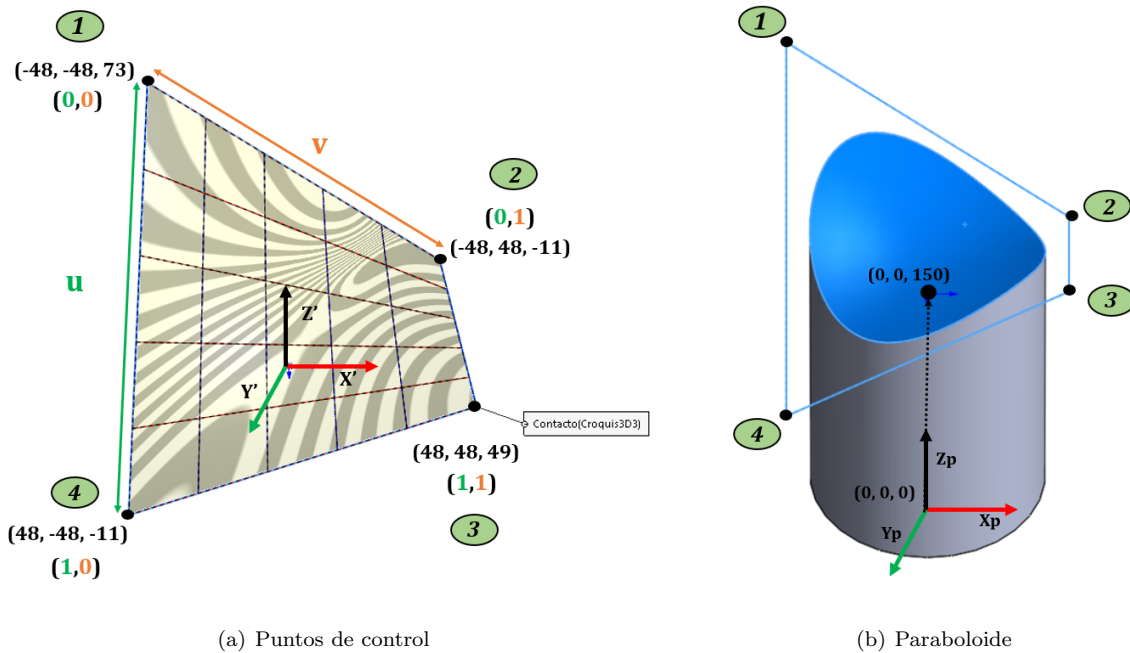


Figura 6: Diseño del paraboloide en SolidWorks. (Elaboración propia).

Para programar el problema se va a usar un sistema de coordenadas cartesianas. Para ello es necesario pasar la ecuación 1 a una función dependiente de los ejes  $\mathbf{X}$ ,  $\mathbf{Y}$ , manteniendo la restricción  $R = \sqrt{X^2 + Y^2} \leq 40mm$ . Se obtiene así la ecuación 2 del paraboloide en función de  $x$  y  $y$ . Y para el cómputo de la normal en cualquier punto de la superficie se hace uso de la

definición de normal, la cuál es el gradiente de la función de  $z$ , ecuación 3, respecto de  $x$  e  $y$ , ecuación 4.

$$\mathbf{S}(x, y) = \begin{pmatrix} x \\ y \\ 150 + (\frac{3x}{2} + 72)(\frac{y}{96} + \frac{1}{2}) - \frac{7y}{8} - \frac{7x}{8} - 11 \end{pmatrix} \quad (2)$$

$$z = f(x, y) = 150 + (\frac{3x}{2} + 72)(\frac{y}{96} + \frac{1}{2}) - \frac{7y}{8} - \frac{7x}{8} - 11 \quad (3)$$

$$\vec{n} = \nabla f(x, y) = \left\{ \frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}, \frac{\partial f}{\partial z} \right\} = \left\{ \frac{y-8}{64}, \frac{x-8}{64}, 1 \right\} \quad (4)$$

### 3.2. Ajustes del CAM

Para este proyecto se ha hecho uso del programa de [NX CAM Pro 2022](#). En éste, generalmente se introduce la pieza a mecanizar, previamente diseñada en CAD.

Los principales ajustes que se pueden escoger en el programa son los siguientes: Definición de las herramientas de mecanizado, operaciones de desbaste y acabado para múltiples estrategias. También el número de ejes de la máquina a usar en la estrategia, profundidades de corte, avances y velocidades del husillo de la herramienta, entre otros. Finalmente se obtiene la trayectoria y las órdenes que se le darán a la máquina para realizar el mecanizado. En el presente proyecto se han escogido los siguientes ajustes:

#### 3.2.1. Geometría de herramienta

La herramienta escogida es una herramienta de punta esférica de longitud 50mm, diámetro 10mm y de 4 filos. Se ha escogido este tipo de herramienta ya que la forma esférica de la herramienta permite el mecanizado de superficies curvas y contornos complejos con mayor precisión. Puede realizar cortes suaves y continuos sin dejar marcas o arañazos abruptos. El material de la herramienta es un acero de dureza 50 HRC, con código U5140.300, sacada del catálogo de herramientas de [Fraisa](#).

#### 3.2.2. Sistemas de coordenadas

Los sistemas de coordenadas a tener en cuenta a la hora de mecanizar en la máquina utilizada son: El sistema de coordenadas de la máquina herramienta global, que se denota por  $SC_{MH}$  y está localizado lejos del Trunnion. También se tiene en cuenta el sistema de coordenadas de la herramienta ( $SC_H$ ), el de la pieza diseñada en CAD ( $SC_P$ ) el cuál se hace coincidir con el sistema de coordenadas del Trunnion ( $SC_{Trunnion}$ ), que se encuentra en el centro del plato del mismo y es el mismo que el de mecanizado ( $SC_{Mecanizado}$ ), véase la figura 7.

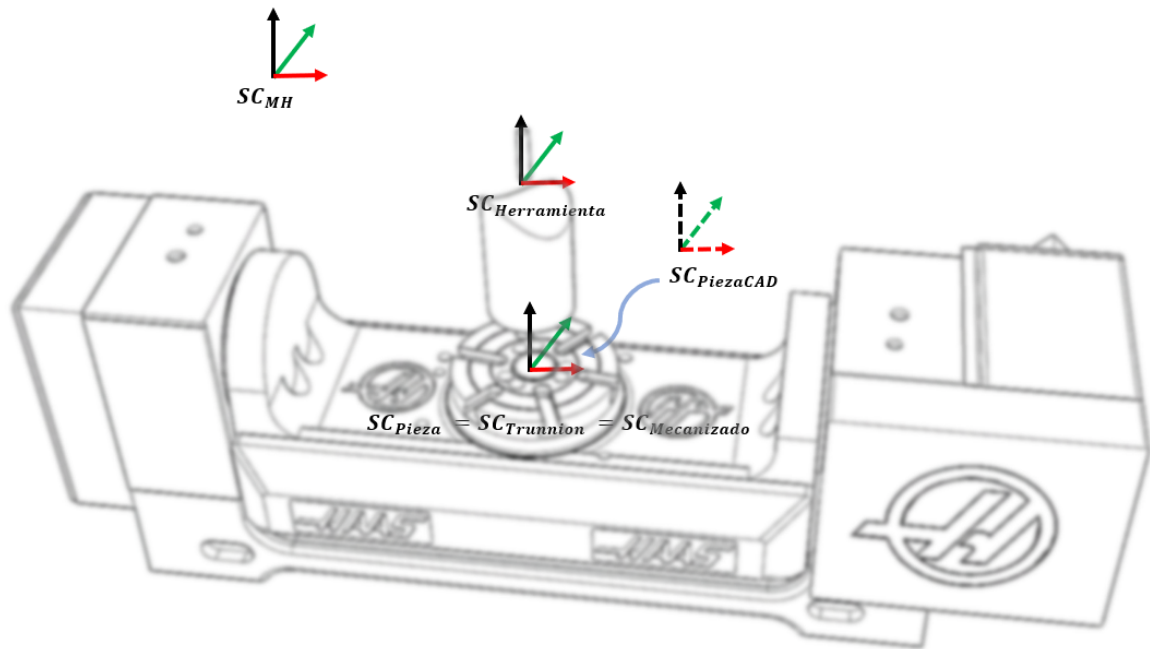


Figura 7: Sistemas de coordenadas existentes para el mecanizado de la pieza. (Elaboración propia).

Los movimientos de la máquina son:

- La mesa o Trunnion lleva los movimientos lineales en  $X_p$  e  $Y_p$ .
- El otro movimiento de traslación lo lleva la herramienta en  $Z_H$ .
- Los giros de los ejes los realiza el trunnion de manera que el giro del eje  $X_p$  se representa por el ángulo  $A$  ( $^\circ$ ) y  $C$  ( $^\circ$ ) representa el giro en el eje  $Z_p$ .

Simplificando, los sistemas de coordenadas a tener en cuenta son, el de la pieza  $SC_p$  y el de la herramienta  $SC_H$ . El de la pieza se encuentra en el centro del plato del Trunnion y el sistema de coordenadas de la herramienta está en la punta de la misma, siendo éste el punto de intersección del eje de la herramienta con la bola esférica de la misma, véase figura 8.

Si se conoce la relación entre ambos sistemas de coordenadas se puede simular los movimientos de la máquina, como movimientos de la herramienta relativos a la pieza, dejando la misma quieta con respecto del sistema de coordenadas de la pieza.

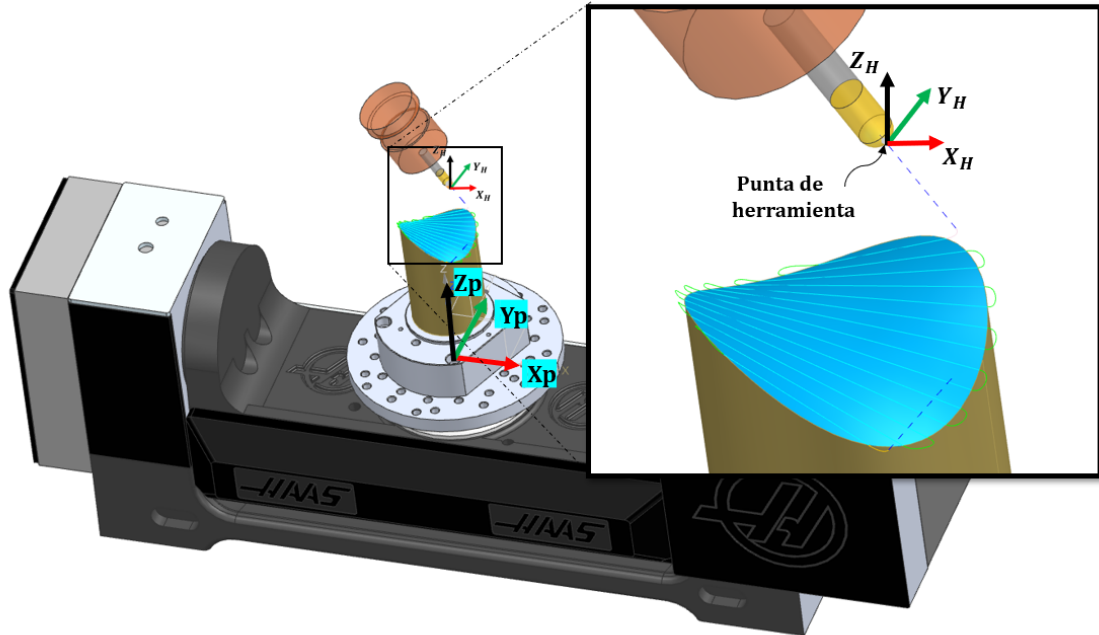


Figura 8: Sistemas de coordenadas utilizados. (Elaboración propia).

### 3.2.3. Estrategia de mecanizado

Se va a simular únicamente la operación de acabado, ya que es dónde más interesa controlar los errores geométricos y su influencia sobre la calidad superficial. Hay unas creces de mecanizados de 1mm en dirección  $Z_p$  sobre la superficie del paraboloide.

Se ha escogido un acabado de 5 ejes, de contorno variable con la herramienta normal a la superficie guía, véase la figura 9. Esto ha sido una aproximación inicial, ya que de esta forma se logra que la punta de la herramienta coincida con el punto de contacto herramienta-pieza al ser la herramienta completamente normal a la superficie. Más adelante se generalizará la forma de extrapolarlo a estrategias que no sean normales a la superficie.

La forma en la que se ha elegido que se mueva la herramienta es en zig zag con 15 pasadas paralelas al eje  $X$ , con una distancia de 5,33mm entre cada una. Se ha elegido dejar una de-

masía de 0mm, es decir un acabado final. Para dicha estrategia el CAM ha generado un total de 580 puntos que forman en conjunto la trayectoria que debe seguir la herramienta, veáse la figura 9.

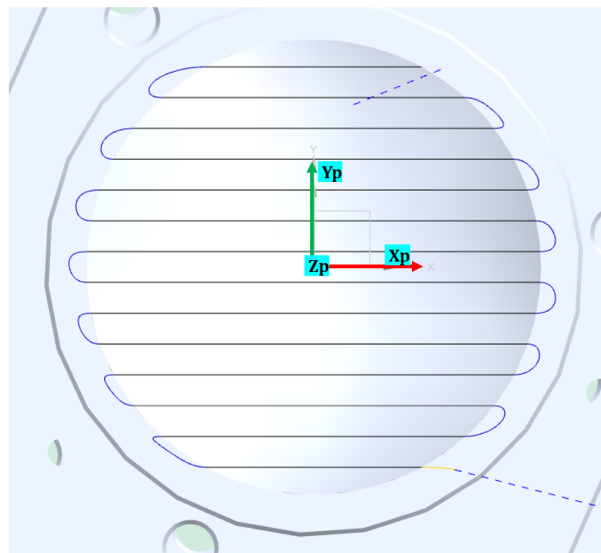
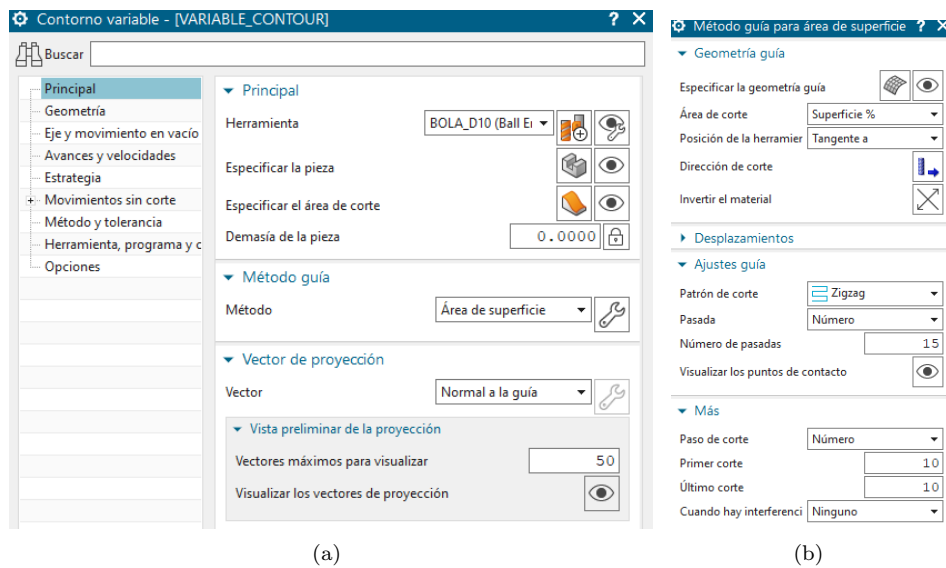


Figura 9: Estrategia de mecanizado CAM. (Elaboración propia).

### 3.3. Parámetros de mecanizado

Para la velocidad del husillo se ha escogido una de  $1000rpm$ , un avance por diente de  $0,0625mm$  y una profundidad de corte máxima del 30 % del diámetro de la bola.

### 3.4. Extracción de trayectorias CAM

Introducidos parámetros y estrategias de mecanizado junto con la pieza, NX CAM Pro propone un conjunto de trayectorias. El programa presenta los resultados en dos formatos **GOTO** y **POST**, en los cuales las posiciones se encuentran en mm y los ángulos girados en grados:

- **GOTO**: Representa la trayectoria hecha por la herramienta y las coordenadas están referidas al sistema de coordenadas pieza ( $SC_P$ ). Consta de un vector **P** que indica la posición de la punta de herramienta, y de otro vector **N** que son los cosenos directores del eje de la herramienta respecto del  $SC_P$ .
- **POST**: Se saca del post-procesador de la máquina, llamado también código G. Es información que verá la máquina, ya que son las posiciones de los ejes que tiene que mover para realizar la trayectoria. Está formado por un vector **T** que indica las traslaciones en cada uno de los ejes **X**, **Y**, **Z** y ángulos **A** y **C** que representan el giro alrededor de X y Z, respectivamente.

$$\mathbf{GOTO}=[\mathbf{P}, \mathbf{N}]=[xyz, ijk]_{SC_P}$$

$$\mathbf{POST}=[\mathbf{T}, \mathbf{A}, \mathbf{C}]=[XYZ, A, C]$$

Las diferencias son que el **GOTO** es información global e indica como se sitúa la herramienta respecto de la pieza. Mientras que el **POST** es información específica de cada máquina que tiene en cuenta su diseño estructural, en el cual XYZ están asociados al sistema de coordenadas máquina y el AC están asociados al sistema de coordenadas trunnion. Se encontrará la relación matemática entre ambas formas de representación de trayectorias.

Es necesario encontrar la relación matemática entre ambos, para lo cual se usará Matlab y como la información no se puede extraer directamente del CAM, se preparará como matrices en formato de excel (*.xlsx*). Serán filas para cada punto de la trayectoria, siendo en el caso del **POST** 5 columnas con los movimientos de la máquina y 6 columnas con la información de la herramienta para el **GOTO**.

Por ello se han creado scripts en python que automaticen la extracción y preparación de datos. Se puede observar la diferencia entre ambos archivos, antes de la preparación en la tabla 1 y después de realizarla, en la tabla 2.

(a) GOTO

```

TOOL PATH/VARIABLE_CONTOUR,TOOL,BOLA_D10
TLDATA/MILL,10.0000,5.0000,55.0000,0.0000,0.0000
MSYS/0.0000,0.0000,0.0000,1.0000000,0.0000000,0.0000000,0.0000000,0.0000000,0.0000000
$$ centerline data
PAINT/PATH
PAINT/SPEED,10
LOAD/TOOL,7,ADJUST,7
PAINT/COLOR,186
RAPID
GOTO/49.6707,-41.9729,203.0414,0.5540418,-0.1410988,0.8204443
PAINT/COLOR,211
RAPID
GOTO/24.8124,-35.6422,166.2304
PAINT/COLOR,42
FEDRAT/MMPM,250.0000
GOTO/24.3259,-35.5423,165.7134
GOTO/23.7051,-35.4517,165.3657
GOTO/23.0106,-35.3768,165.2002
GOTO/22.2943,-35.3193,165.1978
GOTO/21.5875,-35.2778,165.3236
GOTO/20.9035,-35.2494,165.5450
GOTO/20.2460,-35.2314,165.8368
GOTO/19.6144,-35.2218,166.1817
GOTO/19.0066,-35.2189,166.5673
PAINT/COLOR,31
GOTO/18.9750,-35.2189,166.5886,0.5540734,-0.1407014,0.8204912
GOTO/18.7220,-35.2189,166.7595,0.5543235,-0.1375200,0.8208616
GOTO/18.2160,-35.2189,167.1012,0.5548072,-0.1311444,0.8215778
GOTO/16.1920,-35.2189,168.4680,0.5565183,-0.1054863,0.8241117
GOTO/12.1440,-35.2189,171.2016,0.5588367,-0.0535835,0.8275448
GOTO/8.0960,-35.2189,173.9352,0.5596402,-0.0012431,0.8287347
GOTO/4.0480,-35.2189,176.6688,0.5589093,0.0511075,0.8276523
GOTO/0.0000,-35.2189,179.4024,0.5566618,0.1030405,0.8243241
GOTO/-4.0480,-35.2189,182.1360,0.5529520,0.1541449,0.8188305
GOTO/-8.0960,-35.2189,184.8695,0.5478670,0.2040421,0.8113005
GOTO/-12.1440,-35.2189,187.6031,0.5415213,0.2523992,0.8019036
GOTO/-16.1920,-35.2189,190.3367,0.5340498,0.2989374,0.7908396
GOTO/-18.2160,-35.2189,191.7035,0.5299380,0.3214534,0.7847506
GOTO/-18.7220,-35.2189,192.0452,0.5288740,0.3270001,0.7831751
GOTO/-18.9750,-35.2189,192.2161,0.5283369,0.3297608,0.7823797
GOTO/-19.0066,-35.2189,192.2374,0.5282695,0.3301053,0.7822799
PAINT/COLOR,36
GOTO/-20.1635,-35.1143,193.0603,0.5221399,0.3391422,0.7825296
GOTO/-21.2382,-34.8307,193.8824,0.5161214,0.3478883,0.7826828
GOTO/-22.2425,-34.4131,194.6692,0.5102171,0.3563499,0.7827472
GOTO/-23.1884,-33.9064,195.3860,0.5044872,0.3644525,0.7827305

```

(b) POST

```

(OP1 VARIABLE_CONTOUR)
(Desbloqueo de frenos)
M11
M13
(HTA: BOLA_D10)
T07 M06
G54
G00 X-28.416 Y68.095
G00 A-34.871 C75.712
G43 H07 Z200.026
S1000 M03
Z155.159
(Movimientos de entrada)
G93 G01 X-28.439 Y68.206 Z154.451 F250
X-28.505 Y68.519 Z153.809 F250
X-28.604 Y68.992 Z153.278 F250
X-28.725 Y69.572 Z152.871 F250
X-28.859 Y70.214 Z152.577 F250
X-29. Y70.89 Z152.375 F250
X-29.145 Y71.583 Z152.248 F250
X-29.292 Y72.285 Z152.179 F250
X-29.439 Y72.989 Z152.159 F250
X-29.465 Y73.03 Z152.154 A-34.866 C75.751 F250
X-29.675 Y73.364 Z152.108 A-34.829 C76.067 F250
X-30.084 Y74.043 Z152.012 A-34.757 C76.701 F250
X-31.587 Y76.908 Z151.563 A-34.501 C79.267 F250
X-33.899 Y83.328 Z150.35 A-34.153 C84.523 F250
X-35.201 Y90.567 Z148.721 A-34.031 C89.873 F250
X-35.441 Y98.473 Z146.682 A-34.142 C95.225 F250
X-34.631 Y106.847 Z144.257 A-34.48 C100.487 F250
X-32.838 Y115.489 Z141.471 A-35.032 C105.577 F250
X-30.179 Y124.208 Z138.363 A-35.777 C110.427 F250
X-26.791 Y132.843 Z134.974 A-36.688 C114.99 F250
X-22.823 Y141.269 Z131.35 A-37.736 C119.238 F250
X-20.665 Y145.375 Z129.466 A-38.302 C121.24 F250
X-20.11 Y146.391 Z128.986 A-38.448 C121.728 F250
X-19.83 Y146.896 Z128.747 A-38.521 C121.97 F250
X-19.795 Y146.958 Z128.718 A-38.53 C122. F250
X-18.464 Y148.401 Z128.639 A-38.507 C123.005 F250
X-17.011 Y149.697 Z128.67 A-38.493 C123.982 F250
X-15.477 Y150.848 Z128.766 A-38.487 C124.932 F250
X-13.906 Y151.855 Z128.878 A-38.489 C125.845 F250
X-12.342 Y152.722 Z128.965 A-38.497 C126.709 F250
X-10.829 Y153.45 Z128.985 A-38.51 C127.508 F250
X-9.398 Y154.034 Z128.935 A-38.526 C128.236 F250
X-8.078 Y154.474 Z128.814 A-38.545 C128.886 F250
X-6.896 Y154.761 Z128.626 A-38.564 C129.454 F250

```

Tabla 1: Comparación del código extraído del CAM, antes de realizar la preparación. (Elaboración propia).

(a) GOTO

	1	2	3	4	5	6
1	24.8124	-35.6422	166.2304	0.5540	-0.1411	0.8204
2	24.3259	-35.5423	165.7134	0.5540	-0.1411	0.8204
3	23.7051	-35.4517	165.3657	0.5540	-0.1411	0.8204
4	23.0106	-35.3768	165.2002	0.5540	-0.1411	0.8204
5	22.2943	-35.3193	165.1978	0.5540	-0.1411	0.8204
6	21.5875	-35.2778	165.3236	0.5540	-0.1411	0.8204
7	20.9035	-35.2494	165.5450	0.5540	-0.1411	0.8204
8	20.2460	-35.2314	165.8368	0.5540	-0.1411	0.8204
9	19.6144	-35.2218	166.1817	0.5540	-0.1411	0.8204
10	19.0066	-35.2189	166.5673	0.5540	-0.1411	0.8204
11	18.9750	-35.2189	166.5886	0.5541	-0.1407	0.8205
12	18.7220	-35.2189	166.7595	0.5543	-0.1375	0.8209
13	18.2160	-35.2189	167.1012	0.5548	-0.1311	0.8216
14	16.1920	-35.2189	168.4680	0.5565	-0.1055	0.8241
15	12.1440	-35.2189	171.2016	0.5588	-0.0536	0.8275
16	8.0960	-35.2189	173.9352	0.5596	-0.0012	0.8287
17	4.0480	-35.2189	176.6688	0.5589	0.0511	0.8277
18	-9.5320e-08	-35.2189	179.4024	0.5567	0.1030	0.8243
19	-0.0480	-35.2189	182.1360	0.5530	0.1541	0.8188
20	-8.0960	-35.2189	184.8695	0.5479	0.2040	0.8113

(b) POST

	1	2	3	4	5
1	-28.4160	68.0947	155.1587	-34.8710	75.7120
2	-28.4393	68.2061	154.4508	-34.8710	75.7120
3	-28.5048	68.5193	153.8088	-34.8710	75.7120
4	-28.6035	68.9920	153.2777	-34.8710	75.7120
5	-28.7247	69.5718	152.8707	-34.8710	75.7120
6	-28.8589	70.2141	152.5765	-34.8710	75.7120
7	-29.0002	70.8902	152.3751	-34.8710	75.7120
8	-29.1450	71.5834	152.2478	-34.8710	75.7120
9	-29.2915	72.2847	152.1794	-34.8710	75.7120
10	-29.4387	72.9890	152.1586	-34.8710	75.7120
11	-29.4649	73.0295	152.1535	-34.8660	75.7510
12	-29.6746	73.3641	152.1076	-34.8290	76.0670
13	-30.0841	74.0434	152.0116	-34.7570	76.7010
14	-31.5873	76.9078	151.5633	-34.5010	79.2670
15	-33.8990	83.3278	150.3501	-34.1530	84.5230
16	-35.2008	90.5672	148.7207	-34.0310	89.8730
17	-35.4412	98.4727	146.6822	-34.1420	95.2250
18	-34.6306	106.8471	144.2567	-34.4800	100.4870
19	-32.8383	115.4891	141.4714	-35.0320	105.5770
20	-30.1786	124.2083	138.3626	-35.7770	110.4270

Tabla 2: Comparación del código extraído del CAM, después de la preparación. (Elaboración propia).

### 3.5. Modelo cinemático de la máquina

Para hallar la relación que hay entre el **GOTO** y el **POST** es necesario conocer el modelo cinemático de la máquina, que es justamente lo que se necesita para poder hacer el postprocesador del CAM. En la siguientes referencias se crean varios postprocesadores de máquinas CNC de 5 ejes, veáse los artículos [8, 10-17]. La máquina a analizar es la HAAS de mesa móvil (giratoria e inclinable) con el esquema cinemático **XYACFZ** explicado anteriormente, en la figura 10 se ve el esquema de los movimientos de la máquina.

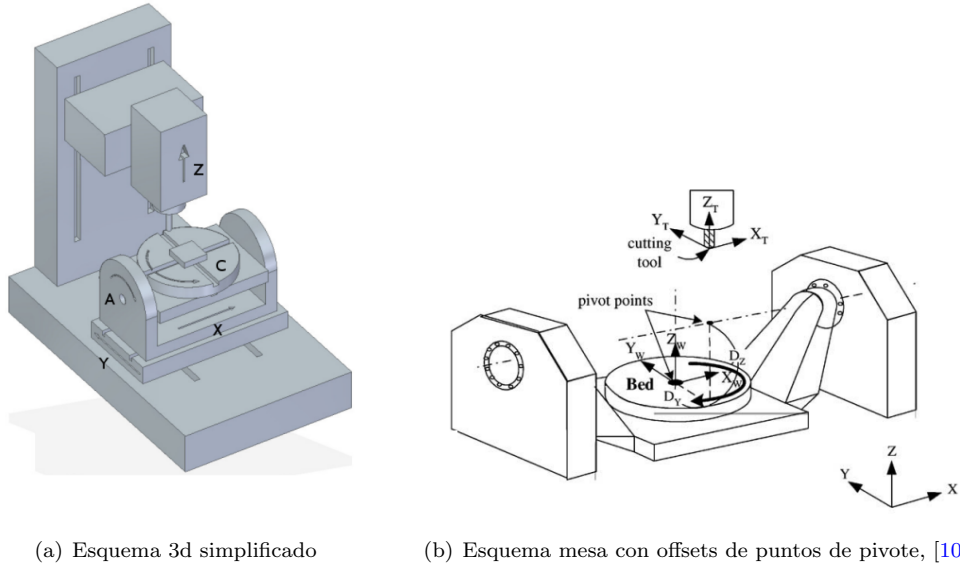


Figura 10: Esquemas de la máquina estudiada (HAAS)

La secuencia de multiplicación de las matrices de transformación que representan el movimiento siguen la secuencia siguiente: Movimiento de traslación en los tres ejes lineales **Z**, **X** e **Y** y después el giro en **X** y en **Z**, que se representan con los ángulos **A** y **C** respectivamente.

Se obtiene finalmente la ecuación del modelo cinemático de la máquina, ecuación 10. Dadas las posiciones  $[X, Y, Z, A, C]_M$  de la máquina, si se coge la cuarta columna de la multiplicación de matrices, se obtiene la posición  $[x, y, z]_H$  de la punta de herramienta, ecuación 12. Mientras que si se coge la tercera columna, se consigue la orientación de la herramienta  $[i, j, k]_H$ , de la ecuación 13. Todo ello en sistema de coordenadas de pieza.

Dónde **T** indica traslación con el subíndice del eje en el que se mueve y **R** denomina rotación, en X o Z, ángulos A o C. Dicha ecuación, es el modelo cinemático puro, sin implementación de los errores de la máquina.

$$T_X = \begin{bmatrix} 1 & 0 & 0 & X \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5)$$

$$T_Y = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & Y \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (6)$$

$$T_Z = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & Z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (7)$$

$$R_A = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(A) & -\sin(A) & 0 \\ 0 & \sin(A) & \cos(A) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (8)$$

$$R_C = \begin{bmatrix} \cos(-C) & -\sin(-C) & 0 & 0 \\ \sin(-C) & \cos(-C) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (9)$$

$$\begin{Bmatrix} x \\ y \\ z \\ 1 \end{Bmatrix} = R_C^{-1} \cdot R_A^{-1} \cdot T_Y \cdot T_X \cdot T_Z \cdot \begin{Bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{Bmatrix} \quad (10)$$

$$\begin{Bmatrix} i \\ j \\ k \\ 0 \end{Bmatrix} = R_C^{-1} \cdot R_A^{-1} \cdot T_Y \cdot T_X \cdot T_Z \cdot \begin{Bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{Bmatrix} \quad (11)$$

$$\begin{Bmatrix} x \\ y \\ z \\ 1 \end{Bmatrix} = \begin{Bmatrix} X \cos(C) - Y \cos(A) \sin(C) - Z \sin(A) \sin(C) \\ X \sin(C) + Y \cos(A) \cos(C) + Z \cos(C) \sin(A) \\ Z \cos(A) - Y \sin(A) \\ 1 \end{Bmatrix} \quad (12)$$

$$\begin{Bmatrix} i \\ j \\ k \\ 0 \end{Bmatrix} = \begin{Bmatrix} -\sin(A) \sin(C) \\ \cos(C) \sin(A) \\ \cos(A) \\ 0 \end{Bmatrix} \quad (13)$$

### 3.6. Errores geométricos en la máquina

Los errores geométricos en una máquina CNC de 5 ejes se refieren a las discrepancias o desviaciones entre los movimientos planificados y los movimientos reales de los ejes de la máquina. Estos errores pueden surgir debido a diversas fuentes, como imperfecciones en la estructura mecánica, flexibilidad de los componentes, desgaste, holguras, errores de calibración, entre otros.

Comprender y compensar estos errores permite lograr resultados más precisos y consistentes, lo que se traduce en una mejora en la calidad de las piezas producidas. Si no se tienen en cuenta, estos errores pueden generar piezas fuera de tolerancia, desperdicio de material y costos adicionales. Dentro de los errores cinemático se distinguen los geométricos que son los que están dentro de las matrices de transformación.

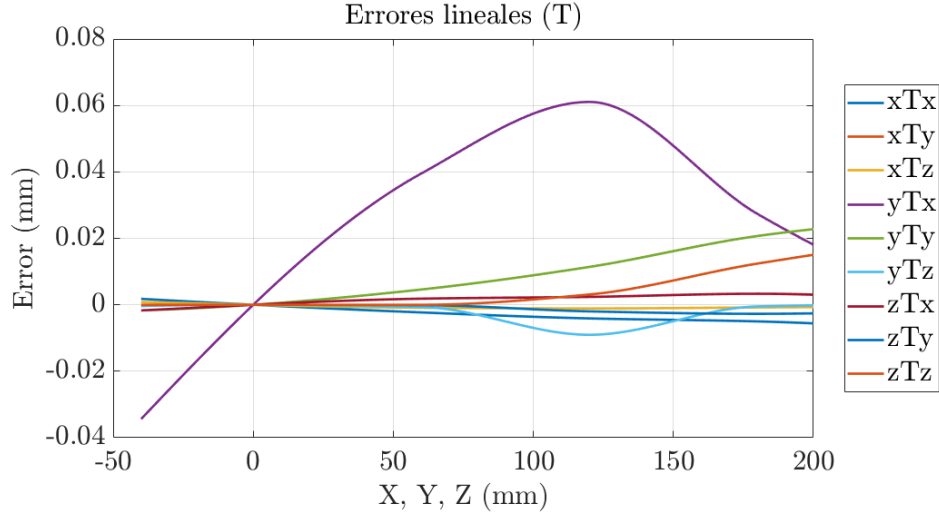
Conocida la importancia que tienen los errores geométricos y conseguido el modelo cinemático de la máquina. Se procede a implementar los errores geométricos, siguiendo lo propuesto en [18, 19]. Se distinguen dos grupos de errores, los que son debidos a los movimientos de traslación en XYZ, son en total 21 errores [1], y añadiendo los debidos a los movimientos de rotación AC de la máquina, en total son 41 errores para máquinas CNC de 5 ejes. Es más complicado corregir estos errores en las máquinas de 5 ejes, debido a estos dos giros adicionales.

Estos errores tienen su efecto sobre la posición y orientación de la herramienta relativos a la pieza. Es necesario modelarlos y compensarlos para mejorar la precisión de la máquina en el mecanizado de piezas, véase [20]. Dentro de cada grupo existen los siguientes:

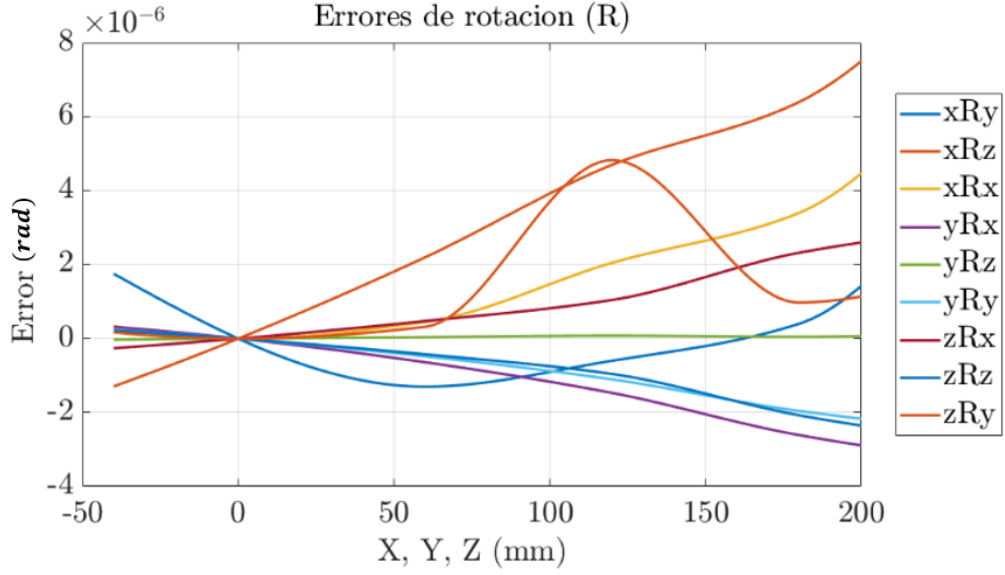
#### 3.6.1. Errores debidos a las traslaciones de la máquina

- Error de perpendicularidad -  $iP_j$  error entre eje  $i$  y  $j$
- Errores lineales -  $iT_j$  error en  $j$  al moverse el eje  $i$
- Errores de rotación -  $iR_j$  error en  $j$  al moverse el eje  $i$

Quedando las matrices que implementan los errores de rotación  $iR_\alpha$  y matrices de traslación  $T_\alpha$  que llevan incluidos los errores de perpendicularidad y lineales.



(a) Errores lineales



(b) Errores de rotación

Figura 11: Errores geométricos debidos a traslaciones XYZ. (Elaboración propia).

$$T_X = \begin{bmatrix} 1 & 0 & 0 & X + xT_x \\ 0 & 1 & 0 & xTy - X \cdot xP_y \\ 0 & 0 & 1 & xT_z - X \cdot xP_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (14)$$

$$T_Y = \begin{bmatrix} 1 & 0 & 0 & yT_x \\ 0 & 1 & 0 & Y + yT_y \\ 0 & 0 & 1 & yT_z - Y \cdot yP_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (15)$$

$$T_Z = \begin{bmatrix} 1 & 0 & 0 & zT_x \\ 0 & 1 & 0 & zT_y \\ 0 & 0 & 1 & Z + zT_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (16)$$

$${}^iR_x = \begin{bmatrix} 1 & xR_z & -xR_y & 0 \\ -xR_z & 1 & xR_x & 0 \\ xR_y & -xR_x & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (17)$$

$${}^iR_y = \begin{bmatrix} 1 & yR_z & -yR_y & 0 \\ -yR_z & 1 & yR_x & 0 \\ yR_y & -yR_x & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (18)$$

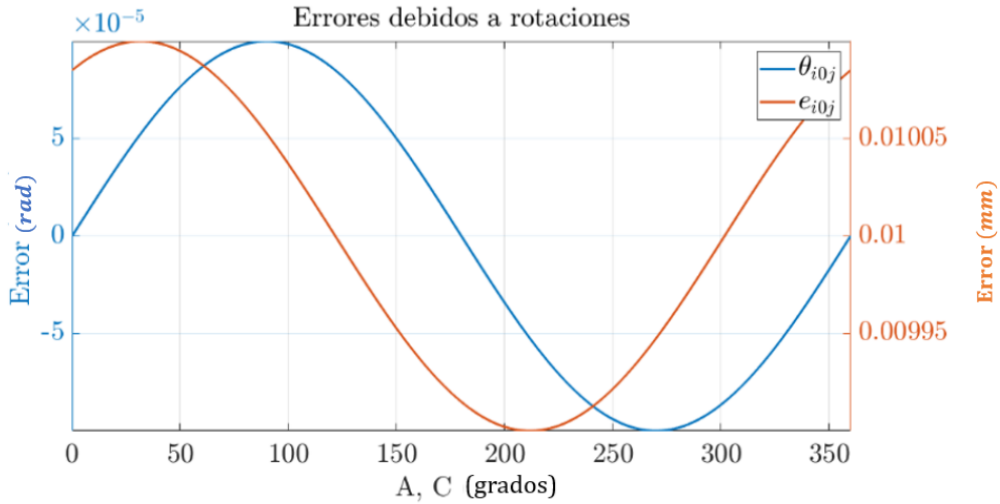
$${}^iR_z = \begin{bmatrix} 1 & zR_z & -zR_y & 0 \\ -zR_z & 1 & zR_x & 0 \\ zR_y & -zR_x & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (19)$$

### 3.6.2. Errores debidas a las rotaciones de la máquina

En las referencias [21, 22] señalan que 2 errores de desplazamiento y 2 errores angulares para cada eje rotatorio (A,C) son suficientes para definir los errores geométricos en máquinas de 5 ejes con 2 articulaciones de giro. Se han definido los siguientes:

- $e_{x0c}$  y  $e_{y0c}$  son los errores de posición del eje C en X e Y respectivamente.
- $\theta_{x0c}$  y  $\theta_{y0c}$  son los errores angulares del eje C sobre los ejes X e Y respectivamente.

Se hace de la misma manera para el giro A del eje X. Dónde la matrices  $E_A$  y  $E_C$  son matrices que incluyen los errores para los giros en el eje X y Z.



(a) Errores lineales

Figura 12: Errores geométricos debidos a giros AC. (Elaboración propia).

$$R_A = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(A) & -\sin(A) & 0 \\ 0 & \sin(A) & \cos(A) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (20)$$

$$R_C = \begin{bmatrix} \cos(-C) & -\sin(-C) & 0 & 0 \\ \sin(-C) & \cos(-C) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (21)$$

$$E_A = \begin{bmatrix} 1 & -\theta_{z0a} & \theta_{y0a} & 0 \\ \theta_{z0a} & 1 & 0 & e_{y0a} \\ -\theta_{y0a} & 0 & 1 & e_{z0a} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (22)$$

$$E_C = \begin{bmatrix} 1 & 0 & \theta_{y0c} & e_{x0c} \\ 0 & 1 & -\theta_{x0c} & e_{y0c} \\ -\theta_{y0c} & \theta_{x0c} & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (23)$$

Quedando así la matriz final con todos los errores, que representa el modelo cinemático con los errores incluidos:

$$\mathbf{M} = E_C \cdot \mathbf{R}_C^{-1} \cdot E_A \cdot \mathbf{R}_A^{-1} \cdot iR_y \cdot \mathbf{T}_Y \cdot iR_x \cdot \mathbf{T}_X \cdot iR_z \cdot \mathbf{T}_Z \quad (24)$$

## 4. Estudio de los errores

Se analizarán los errores del propio diseño CAD, comparando los **GOTO** con la superficie teórica dada por las ecuaciones del paraboloide teóricos, siendo del orden de **1e – 5mm**. Después el **POST** se pasará por el modelo cinemático sin errores M(POST), siendo los errores del postprocesador del orden de **1e – 10mm**. Al introducir los errores geométricos de la máquina, pasan a ser del orden de **5e – 2mm**, que comparados con el resto de errores son los predominantes, y son los que se van a optimizar. La metodología seguida para el análisis de errores ha sido la que se representa en la figura 13.

De la optimización se obtienen unas posiciones de la máquina  $[X, Y, Z, A, C]_{OP}$  para cada punto de la trayectoria, que en conjunto será el código de la máquina HAAS optimizado, que logra unos errores de mecanizado real mucho menores que los que se habrían obtenido mecanizando únicamente con el postprocesador del CAM, mejorando así la precisión y acabado de la pieza.

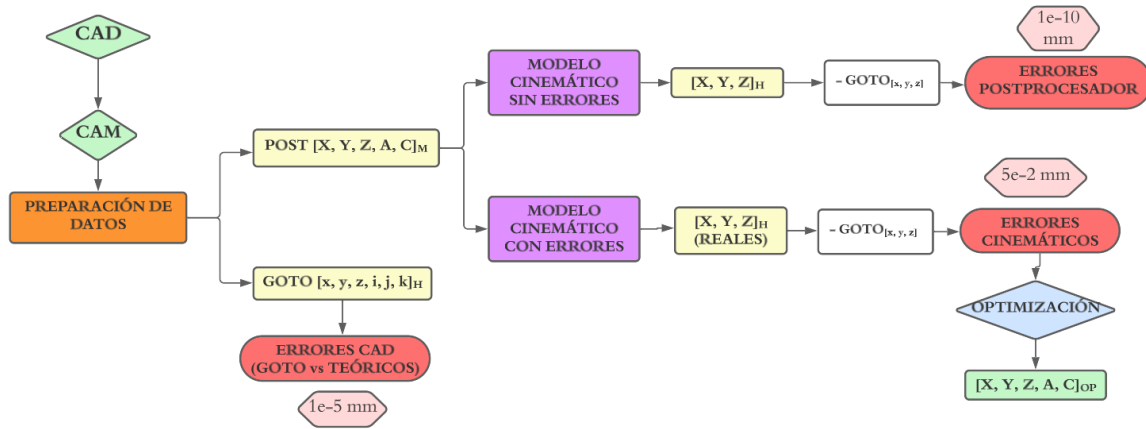


Figura 13: Metodología seguida y errores a tener en cuenta. (Elaboración propia).

### 4.1. Análisis de los errores de diseño CAD/CAM frente a la superficie Teórica

En primer lugar se van a analizar los errores debidos al propio diseño computacional CAD/CAM. Para ello se va a comparar la trayectoria que genera por el CAM en punta de herramienta y orientación de la misma, con la superficie y normales teóricas. Es decir, el **GOTO** con las posiciones y normales teóricas de la superficie. Se retoman de la ecuación 3 las posiciones de los puntos en la superficie y de la ecuación 4 las normales de los mismos.

Se comprobará la distancia de punta teórica a punta de CAD con la distancia mínima entre la punta de herramienta del CAD a la superficie teórica, veáse figura 14.

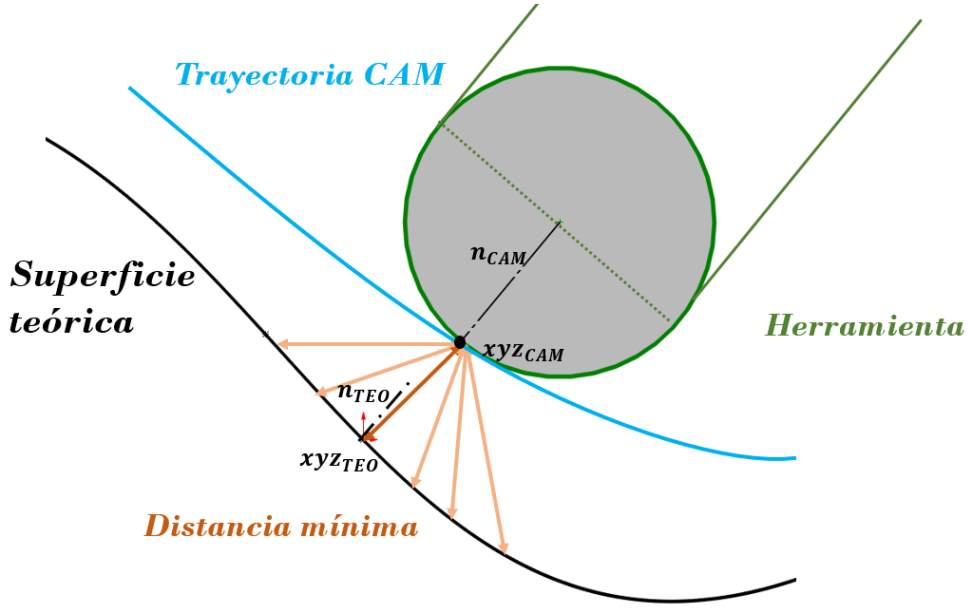


Figura 14: Esquema del error de diseño CAD comparado con la superficie teórica. (Elaboración propia).

Para poder medir el error que hay de los puntos GOTO comparados con la superficie teórica, hay que encontrar la distancia euclídea mínima entre los puntos y la superficie paraboloide. Como la ecuación a resolver no posee de solución directa, debido a la no linealidad de la ecuación de la superficie, se resolverá de manera iterativa con el algoritmo de enjambre de partículas.

Queda como resultado por un lado el error en distancia de la punta de herramienta del CAM a la superficie teórica, figura 15 (a), siendo del orden de  $6e - 5mm$ . Mientras que el error en orientación de la herramienta viene dado por las diferencias en valor absoluto de la normal teórica en el punto a distancia mínima y la orientación dada por el CAM ( $[ijk]_{SC_P}$  del GOTO) al ser la estrategia normal a la superficie.

Esto da como resultado errores del orden de  $1e - 7$  para las diferencias de normales, figura 15 (b). En ésta se observa un patrón cíclico que representa las pasadas que se hacen en la trayectoria, en los arcos como no existe superficie teórica las normales tienen error nulo y van creciendo a medida que se entra dentro de la superficie, siendo el mayor error de orientación el término  $dn_x$ .

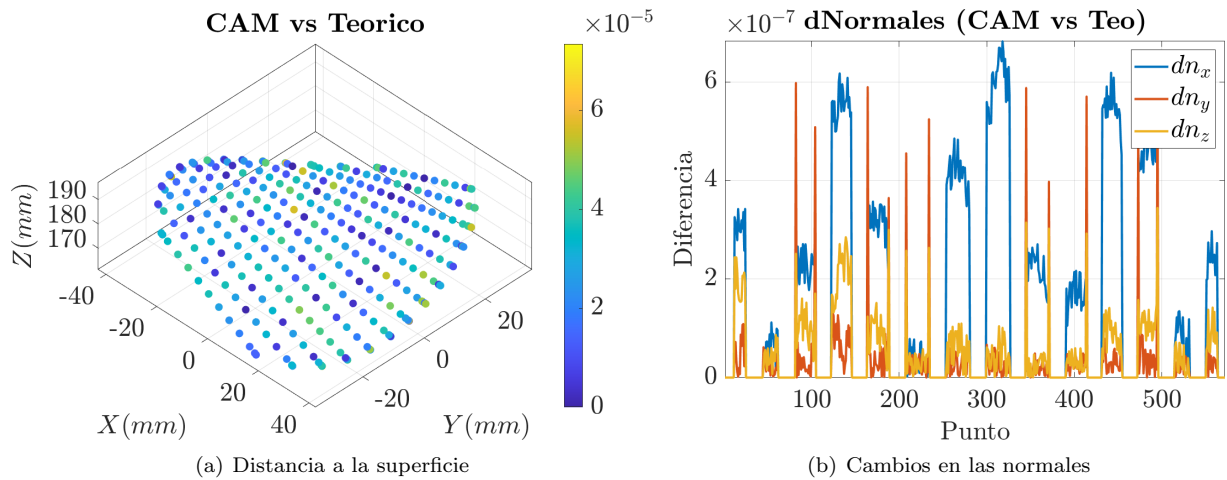


Figura 15: Comparación del diseño CAD/CAM con la superficie teórica. (Elaboración propia).

Esta misma comparación se hará en el software de [GeoMagic Studio](#), en el que se comparará la superficie teórica (conjunto de puntos calculados matemáticamente) con toda la superficie diseñada en CAD. Para ello se extraen las coordenadas de una gran cantidad de puntos de la superficie teórica como matrices y con un script en python se exportan como nubes de puntos en formato ".ply" para que puedan ser leídos por el software. Haciendo esto sale un patrón aleatorio con resultados similares a lo obtenido en Matlab, con órdenes de magnitud de  $1e - 5mm$ , figura 16.

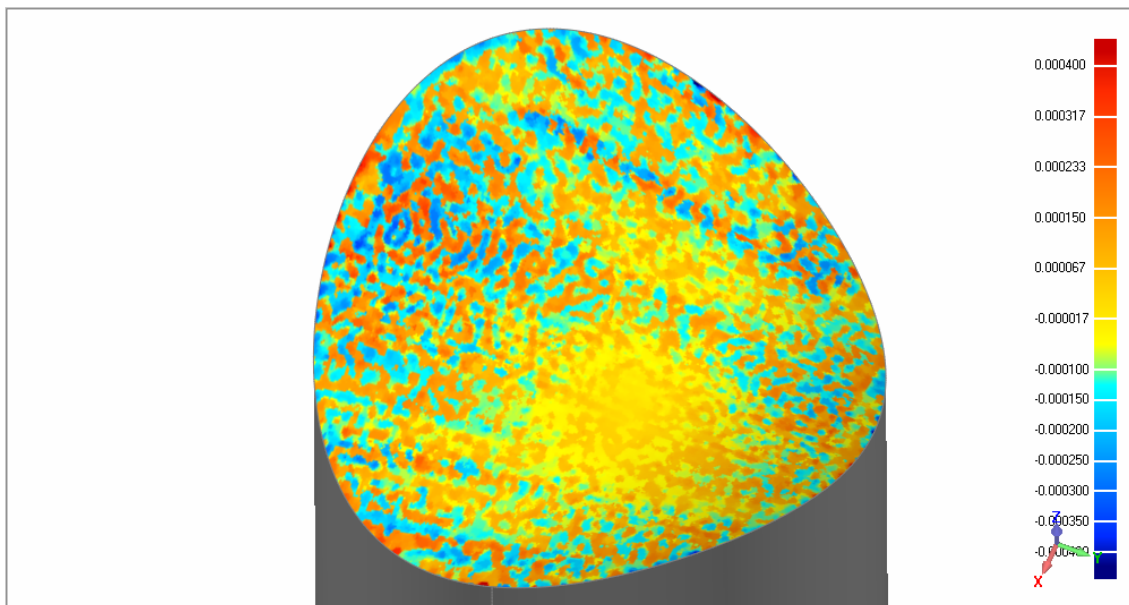


Figura 16: Comparación de puntos teóricos vs Superficie CAM, en GeoMagic. (Elaboración propia).

#### 4.2. Errores del postprocesador. Comprobación del modelo cinemático y extracción de decimales.

Es necesario evaluar la validez del modelo cinemático para ello lo que se hará es comparar la diferencia en valor absoluto ( $|d_\alpha|$ ) entre la información del GOTO ( $[x, y, z, i, j, k]_{SC_P}$ ) y lo que devuelve el modelo cinemático al pasar las posiciones de la máquina del POST, es lo que se denomina M(POST) ( $[x, y, z, i, j, k]_{SC_P}$ ). Esto dará 6 diferenciales en total y se denominará error del postprocesador.

Una problemática que surge es la tolerancia en decimales que proporciona el postprocesador. Ya que si se observa la figura 17 (b), el **POST** genera únicamente 3 decimales para las 5 articulaciones de la máquina. Mientras que el **GOTO** permite como mínimo 4 decimales. Esto hace que tengamos un error por defecto del orden de  $1e - 3mm$ , lo cuál no justifica que el modelo cinemático sea el correcto.

Se encontró una herramienta denominada **Post Review Tool** que escribía un fichero en formato *.out* en el directorio de trabajo. Es un fichero de texto que se puede visualizar, y contiene las variables que usa el postprocesador del CAM para realizar la transformación de **GOTO** a **POST**.

Para extraer la información del fichero se creo un script de toma y preparación de datos también. En este fichero la ventaja principal es que permite visualizar hasta 15 decimales para todos los parámetros. Exceptuando los ángulos de giro, para los cuáles permite 3 decimales, debido a que la tolerancia angular es mucho más fina que la de las traslaciones ( $1^\circ/1000$ ). Es la tolerancia que se especifica para los ángulos, en la especificaciones técnicas de la **HAAS**.

En la gráfica 17 se presentan las diferencias entre el GOTO y el M(POST) para los 580 puntos de la trayectoria, en la subfigura (a) están los diferenciales de las posiciones de punta de herramienta ( $dXYZ$ ) y en la subfigura (b) se representa el error de orientación de la herramienta  $dijk$ . El error para los ejes lineales es del orden de  $1e - 10mm$  para los diferenciales en  $[x, y, z]$  (error de posición). Y un error del orden de  $1e - 6$  para los diferenciales de las normales  $[i, j, k]$  o vectores directores de la herramienta (error de orientación).

Esto valida el modelo cinemático que se ha propuesto para la máquina **HAAS**, ecuación 10, permite relacionar correctamente toda la información extraída del CAM para su posterior uso en la optimización.

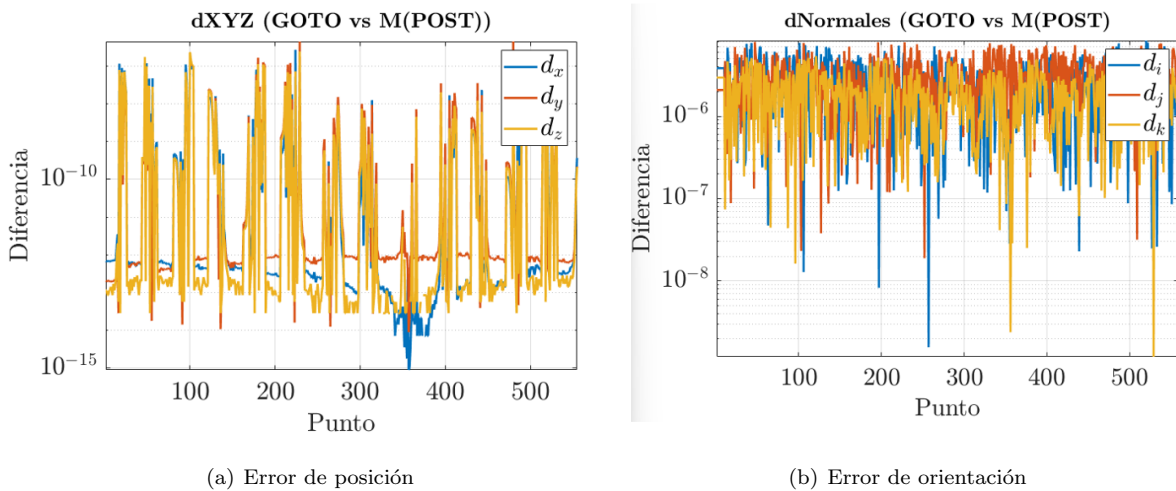


Figura 17: Diferencias entre los GOTO y el resultado de pasar los POST por el modelo cinemático de la máquina. (Elaboración propia).

### 4.3. Métricas de medida de error

Para poder tomar decisiones que tengan que ver con el error producido en toda la trayectoria de mecanizado, hacen falta métricas que evalúen todo el conjunto de puntos con un sólo valor. Para ello se van a usar las siguientes métricas, dónde  $\mathbf{y}$  indica los valores de error para todos los puntos,  $y_j$  indica el valor del error en un único punto e  $\hat{y}_j$  indica la media del error sobre todos los puntos. En general se va a usar la medida del error como distancia euclídea de diferentes formas:

- **Máximo:** Es el error máximo obtenido de todo el conjunto de puntos que forma la trayectoria.

$$MAX = \max(\mathbf{y}) \quad (25)$$

- **Mínimo:** Es el error mínimo obtenido de todo el conjunto de puntos que forma la trayectoria.

$$MIN = \min(\mathbf{y}) \quad (26)$$

- **Error absoluto medio (MAE):** MAE mide la magnitud promedio de los errores en un conjunto de predicciones, sin considerar su dirección. Es el promedio sobre la muestra de prueba de las diferencias absolutas entre la predicción y la observación real donde todas las diferencias individuales tienen el mismo peso.

$$MAE = \frac{1}{n} \sum_{j=1}^n |y_j - \hat{y}_j| \quad (27)$$

- **Error cuadrático medio (RMSE) :** RMSE es una regla de puntuación cuadrática que también mide la magnitud media del error. Es la raíz cuadrada del promedio de diferencias cuadradas entre la predicción y la observación real.

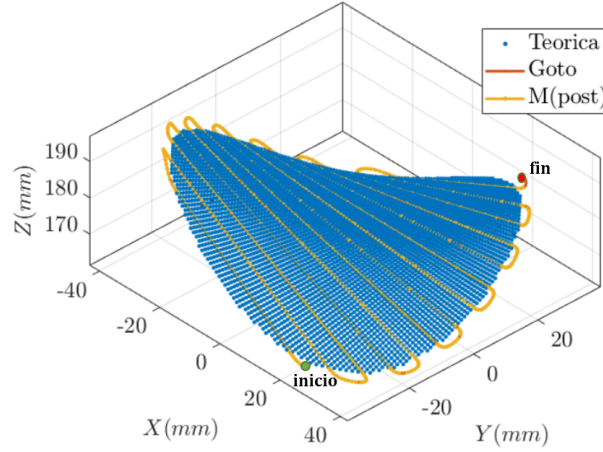
$$RMSE = \sqrt{\frac{1}{n} \sum_{j=1}^n (y_j - \hat{y}_j)^2} \quad (28)$$

- **Desviación estándar  $\sigma$  (DEST):** La desviación estándar es una medida estadística que indica la dispersión o variabilidad de un conjunto de datos con respecto a su media. Es utilizada para cuantificar qué tan alejados están los valores individuales del promedio.

$$\sigma = \sqrt{\frac{\sum_i^N (y_j - \hat{y}_j)^2}{N - 1}} \quad (29)$$

#### 4.4. Análisis de los errores geométricos

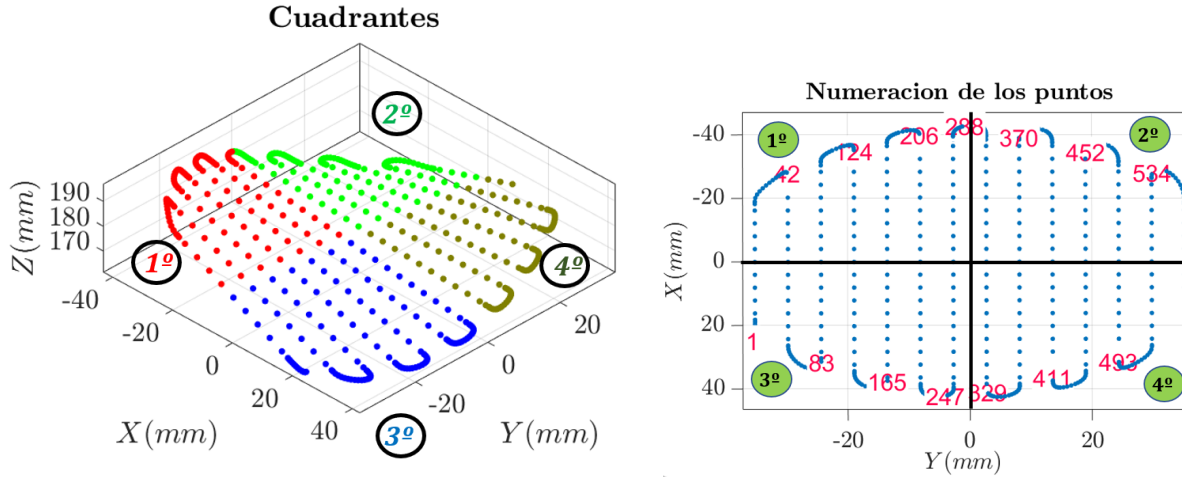
Se van a analizar a continuación los errores geométricos en la pieza para la trayectoria elegida. Estos errores se representan como distancia entre la punta de herramienta del CAM ( $[x, y, z]_{SC_P}$  del GOTO) y la punta de herramienta real ( $[x, y, z]_{SC_P}$  del MPOST) que se obtiene del modelo cinemático con errores, según ecuación 24. Estos errores se medirán para cada punto de la trayectoria, en la gráfica 18 se observa la distribución de éstos puntos, en azul se representa la superficie teórica, en amarillo el MPOST y en naranja el GOTO.



(a) Trayecto obtenido de la estrategia CAM

Figura 18: Trayectoria zig-zag para el mecanizado del paraboloide. (Elaboración propia).

Para hacer un análisis más exhaustivo se va a dividir la superficie en 4 cuadrantes, que siguen los cambios de curvatura del paraboloide, véase la figura 19 (a). Mientras que en la subfigura 19 (b) se muestra la numeración de los puntos de la trayectoria.



(a) Cuadrantes de la trayectoria

(b) Orden de numeración

Figura 19: Cuadrantes del paraboloide a analizar. (Elaboración propia).

#### 4.4.1. Errores según ejes de la máquina

En este apartado se va a estudiar el efecto que tienen los ejes de la máquina al combinarlos de distintas formas, con el objetivo de ver su influencia en los errores geométricos. Para ello se estudian las siguientes casos para la estrategia de perpendicularidad elegida:

- **XYZ:** En este caso se tienen en cuenta únicamente las matrices que introducen los errores en los ejes de traslación. Se pueden ver los errores geométricos de este caso en la figura 20, en la cuál la subfigura (a) se representan los errores en todos los puntos de la trayectoria como distancia euclídea, ecuación 30, mientras que en la subfigura (b) se representan desglosados por diferenciales en cada eje cartesiano ( $[d_x, d_y, d_z]$ ). Esto se hace introduciendo en la matriz del modelo cinemático con errores, ecuación 24, únicamente los errores aportados por estos ejes.

$$Distancia_{12} = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2 + (z_1 - z_2)^2} \quad (30)$$

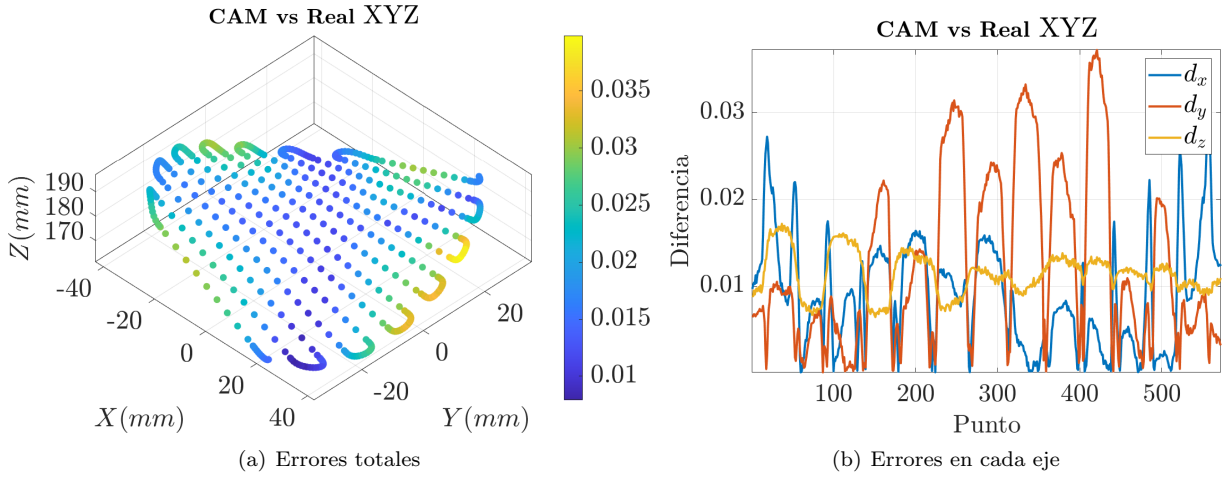


Figura 20: Errores debidos a movimientos de máquina en X-Y-Z. (Elaboración propia)

- **XYZA:** Aquí se añade el error que aporta el giro en X representado por el ángulo A. Se añade la matriz que tiene en cuenta los errores de este giro, que es la matriz  $E_A$ , en la ecuación 24. Los resultados se representan en la figura 21.

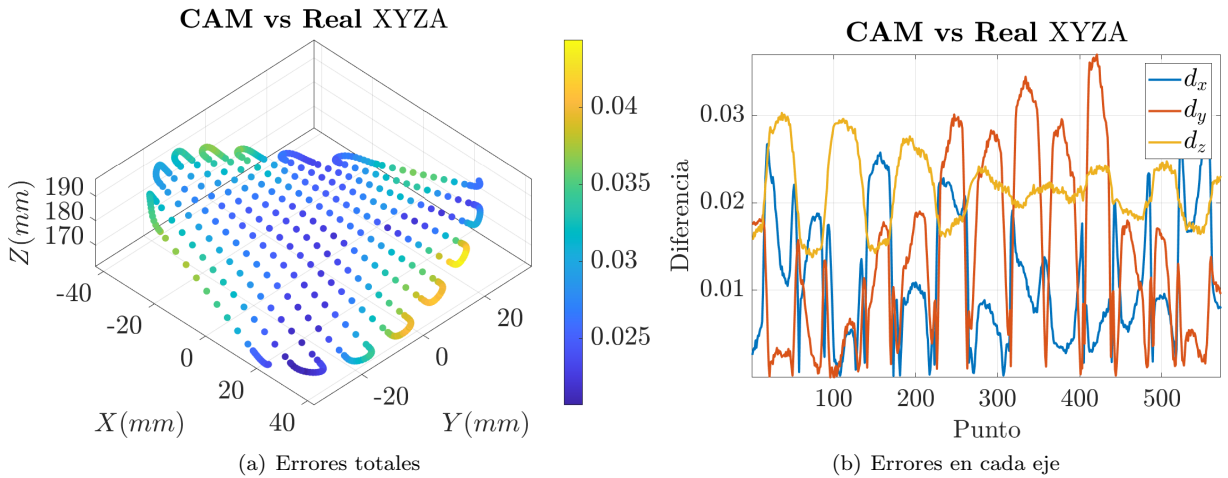


Figura 21: Errores debidos a movimientos de máquina en X-Y-Z-A. (Elaboración propia)

- **XYZC:** Aquí se añade a los errores de traslación el error que aporta el giro en Z representado por el ángulo C. Se añade la matriz que tiene en cuenta los errores de este giro, que es la matriz  $E_C$ , en la ecuación 24. Los resultados se representan en la figura 22.

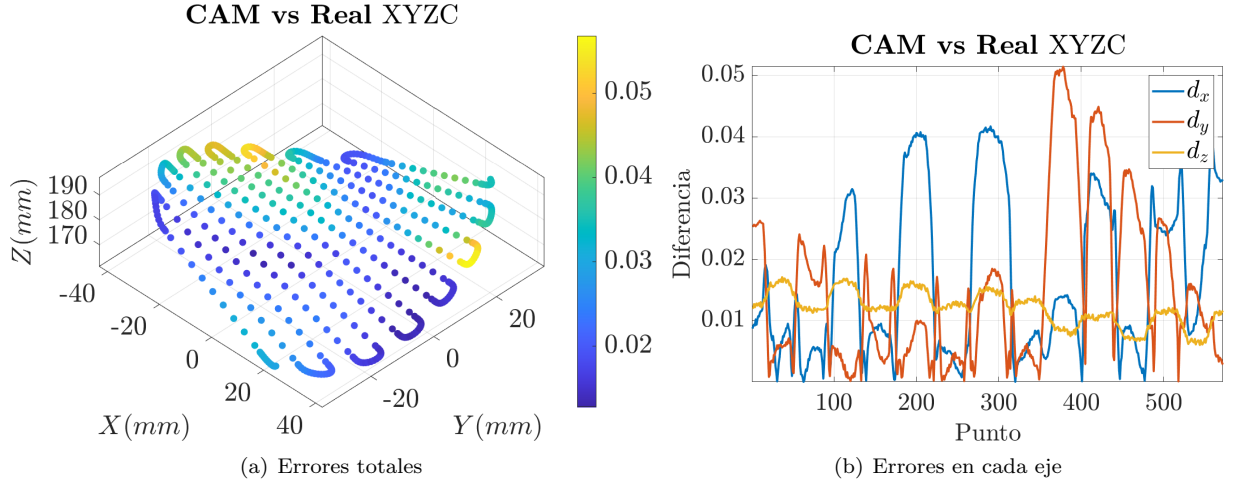


Figura 22: Errores debidos a movimientos de máquina en X-Y-Z-C. (Elaboración propia)

- **XYZAC:** Para este caso se tienen en cuenta todos los errores geométricos de la máquina. Los resultados se representan en la figura 23.

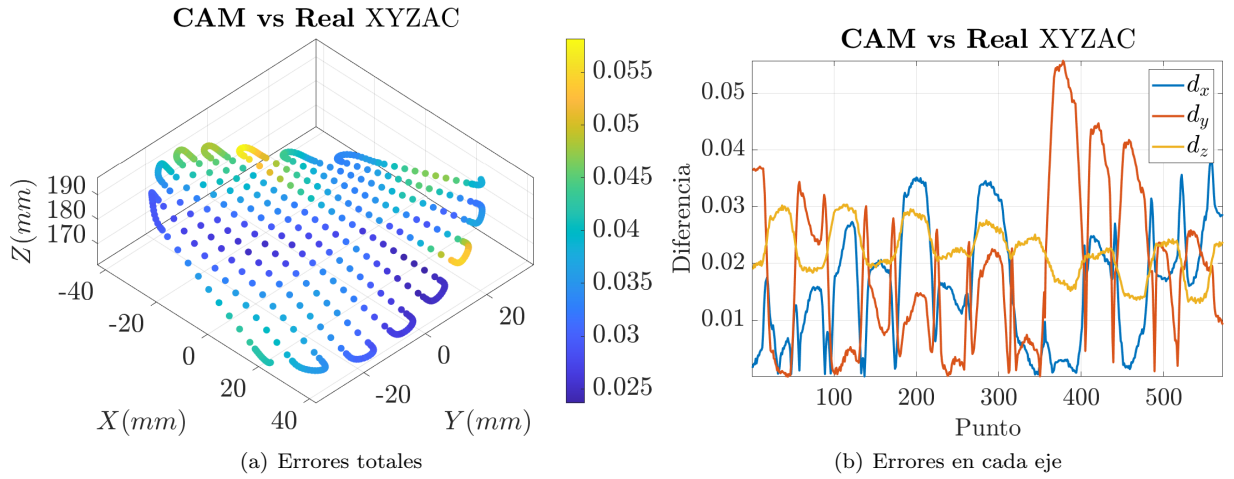


Figura 23: Errores debidos a movimientos de máquina en X-Y-Z-A-C. (Elaboración propia)

En las gráficas anteriores se observa como el menor error está en general localizado en el centro de la pieza, esto es porque por definición, los errores geométricos son nulos en la coordenada 0 y por ello también las zonas que mayor error representan son las situadas más lejos del centro de la pieza.

La mayor contribución al error total la están aportando los ejes de traslación (XYZ), con un máximo de  $0,035mm$  frente a los  $0,055mm$  teniendo en cuenta todos los errores (XYZAC). Para ver cuál de los giros es el que provoca mayor error en la pieza, no es suficiente con ver el valor máximo que se obtiene para los casos XYZA y XYZC, ya que para XYZC este valor es mayor pero también hay mayor cantidad de puntos en los que el error es menor.

Por ello a continuación se van a analizar los errores geométricos dividiendo la superficie en

los 4 cuadrantes mencionados. Se van a tomar las métricas del error absoluto medio (MAE) y la desviación estándar (DEST) para los distintos conjuntos de puntos de los 4 cuadrantes de la pieza. En la tabla 3 se representan dichos resultados para los 4 casos.

Cuadrante	XYZ (a)		XYZA (b)		XYZC (c)		XYZAC (d)	
	MAE	DEST	MAE	DEST	MAE	DEST	MAE	DEST
1°	7.570E-03	1.487E-02	9.270E-03	1.814E-02	6.745E-03	1.338E-02	7.396E-03	1.462E-02
2°	5.831E-03	1.176E-02	7.543E-03	1.494E-02	3.080E-03	6.674E-03	4.828E-03	9.578E-03
3°	5.378E-03	1.060E-02	7.215E-03	1.409E-02	8.850E-03	1.723E-02	1.004E-02	1.953E-02
4°	5.773E-03	1.242E-02	7.209E-03	1.512E-02	7.773E-03	1.678E-02	9.581E-03	2.022E-02

Tabla 3: Tabla con métricas sobre los 4 cuadrantes, estando todos los errores en mm. (Elaboración propia).

Con ello se sacan las siguientes conclusiones:

- Se observa que en general, aumentar el número de errores hace que aumente el error en la pieza. Pero como se observa el caso XYZAC, metiendo todos los errores, hace que el cuadrante 1 y 2 reduzcan su error en comparación con el caso XYZ, a costa de aumentar el error en el cuadrante 3 y 4. Esto es porque los errores llevan signo y hay zonas del paraboloide en el que se suman y otras en las que se anulan entre ellos.
- Los errores obtenidos de los movimientos de giro A y C no son iguales. Aún siendo de la misma forma y magnitud sus errores, véase figura 12. En el caso XYZC salen mejores métricas para el cuadrante 1 y 2, y para el XYZA se obtienen mejores en el cuadrante 3 y 4. Esto es debido a que el resultado en el error no se debe únicamente a los errores geométricos, sino también al aporte que tienen éstos en el modelo cinemático.

En conclusión la mejor forma de realizar el mecanizado teniendo en cuenta los errores geométricos de la máquina sería realizar una estrategia XYZC en el primer y segundo cuadrante seguido de una estrategia XYZ en el tercer y cuarto cuadrante. El mecanizado empieza en el 3° va al 1° y vuelve al 3° unas 7 veces, después pasa al 2° hacia el 4° otras 7 veces. Además recalcar que esto es una aproximación muy grosera, dividiendo la superficie en 4 parches o zonas, y realizando aproximaciones sobre ello.

En cambio los errores y la forma de la ecuación cinemática no lineal, hace que realizar esto exactamente y teniendo en cuenta el mecanizado tridimensional herramienta-pieza sea algo inviable para los cálculos tradicionales. Por ello se va a continuar con la resolución del problema programando un algoritmo para lograr el menor error en mecanizado.

## 5. Descripción del algoritmo de enjambre de partículas

Como se ha visto, las ecuaciones que surgen en el problema del presente proyecto son no lineales complejas, con los errores geométricos entrelazados. Esto hace que sea necesario hacer uso de un algoritmo de optimización para encontrar una solución al problema.

El algoritmo implementado en este proyecto es el enjambre de partículas (Particle Swarm Optimization, PSO). Es una técnica de optimización global inspirada en el comportamiento de los enjambres de aves o peces, figura 24. Fue propuesto por James Kennedy y Russell Eberhart en 1995, en el artículo [23].

Se basa en la idea de que las partículas en un enjambre pueden aprender de sus experiencias pasadas y de las de sus compañeras para buscar soluciones óptimas en un espacio de búsqueda. Cada partícula en el enjambre representa una solución potencial en el espacio de búsqueda y se mueve a través de este espacio ajustando su posición y velocidad.

Este algoritmo ha sido ampliamente utilizado en problemas de optimización en diversas áreas, como la optimización de funciones matemáticas, el diseño de redes neuronales artificiales, la planificación de rutas, la optimización de parámetros, entre otros. Su simplicidad y eficacia han contribuido a su popularidad en la comunidad científica, véase [24-26].



*Figura 24:* Bandada de aves, comportamiento de la naturaleza en el que se basa el algoritmo. Fuente [PSO](#)

## 6. Resolución del problema

### 6.1. Funcionamiento del algoritmo

El funcionamiento básico del algoritmo de enjambre de partículas implica lo siguiente, ver figura 26:

- **Inicialización:** Se crea un enjambre de partículas, donde cada partícula representa una posible solución en el espacio de búsqueda. Se parte de una solución inicial, que puede ser aleatoria o guiada y se randomiza alrededor de ella para que haya variedad, para todas las partículas.
- **Evaluación del coste u objetivo:** Se evalúa la calidad de cada partícula mediante una función objetivo predefinida. En el contexto de las máquinas CNC de 5 ejes, esta función objetivo puede estar relacionada con la eficiencia de mecanizado, la precisión y criterios adicionales que se detallarán más adelante.
- **Actualización de la mejor posición:** Cada partícula mantiene un registro de su mejor posición personal hasta el momento (PBEST). Además, se registra la mejor posición global del enjambre (GBEST), que es la mejor solución encontrada por cualquier partícula en el grupo.
- **Actualización de la velocidad y la posición:** El algoritmo se basa en partículas que se van moviendo de una iteración a otra para alcanzar nuevas posiciones. Para que todas las partículas se muevan, es necesario ajustar su velocidad y posición de acuerdo con fórmulas que consideran tanto su experiencia personal como la experiencia global del enjambre. Estas fórmulas están diseñadas para guiar a las partículas hacia mejores soluciones y/o a espacios de factibilidad no visitados hasta el momento.

Las ecuaciones de velocidad y posición son respectivamente 31 y 32, en las cuáles;  $\mathbf{v}$ , indica velocidad;  $\mathbf{x}$ , posición; el subíndice  $\mathbf{i}$ , la partícula; el subíndice  $\mathbf{t}$ , indica el número de la iteración actual y  $\mathbf{p}$  indica su mejor posición personal (PBEST), mientras que  $\mathbf{g}$  indica la mejor posición global de todas las partículas (GBEST).

$$\mathbf{v}_i(t+1) = w \cdot \mathbf{v}_i(t) + c_1 \cdot \mathbf{r}_1 \cdot (\mathbf{p}_i - \mathbf{x}_i(t)) + c_2 \cdot \mathbf{r}_2 \cdot (\mathbf{g} - \mathbf{x}_i(t)) \quad (31)$$

$$\mathbf{x}_i(t+1) = \mathbf{x}_i(t) + \mathbf{v}_i(t+1) \quad (32)$$

Así, la  $i$ -partícula que se encuentra en la posición  $\mathbf{x}_i(\mathbf{t})$  en la iteración  $\mathbf{t}$  pasará, en la siguiente iteración, a la posición  $\mathbf{x}_i(\mathbf{t}+1)$  sumando el término de velocidad,  $\mathbf{v}_i(\mathbf{t}+1)$ , en el que se tiene en cuenta tanto la velocidad de la iteración anterior junto con las mejores posiciones de la partícula,  $\mathbf{p}_i$ , y de su vecindario,  $\mathbf{g}$ .

El algoritmo se maneja con los siguientes parámetros:

- **N:** Número de partículas de las que va estar compuesta la población.
- **w:** Coeficiente de inercia, que se puede establecer como un coeficiente estático o variable a lo largo de las iteraciones.
- **$c_1$ :** Coeficiente asociado a la componente cognitiva. Indica lo mucho que una partícula tiene que fiarse de sí misma.

- $c_2$ : Coeficiente asociado a la componente social. Indica cuánto tiene que tener en cuenta al resto de partículas.
- $r_1$  y  $r_2$ : Son dos valores aleatorios que varían entre 0 y 1, y sirven para que haya variedad de velocidad para las distintas partículas. Le quita o le da importancia a los términos de búsqueda cognitivo y social, ver figura 25.

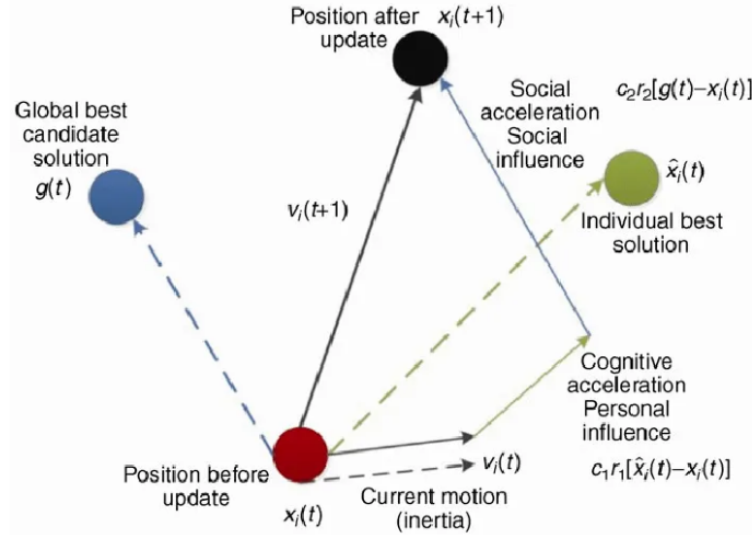


Figura 25: Esquema del movimiento de las partículas en el algoritmo. [27]

- **Iteración:** El proceso de actualización de velocidad y posición se repite iterativamente. Cada partícula busca mejorar su posición personal y, al mismo tiempo, se beneficia de la información compartida por el enjambre para buscar soluciones globales más prometedoras.
- **Convergencia y salida:** El algoritmo continúa iterando hasta que se cumple alguna condición de parada, como alcanzar un número máximo de iteraciones o lograr una convergencia satisfactoria. La mejor solución encontrada se considera la solución óptima o una aproximación cercana a ella.

El problema se va a resolver optimizando punto a punto de la trayectoria y retro alimentando el enjambre de partículas para que tenga en cuenta el resultado en el punto anterior. En cada momento, las partículas están formadas por las variables independientes a optimizar, que son las posiciones de la máquina optimizadas ( $[X, Y, Z, A, C]_{OP}$ ). Además cada partícula llevará información sobre su posición, velocidad y un valor coste u objetivo que indicará lo cerca que está de minimizar la función objetivo. En el esquema 26, se puede seguir el proceso del algoritmo programado.

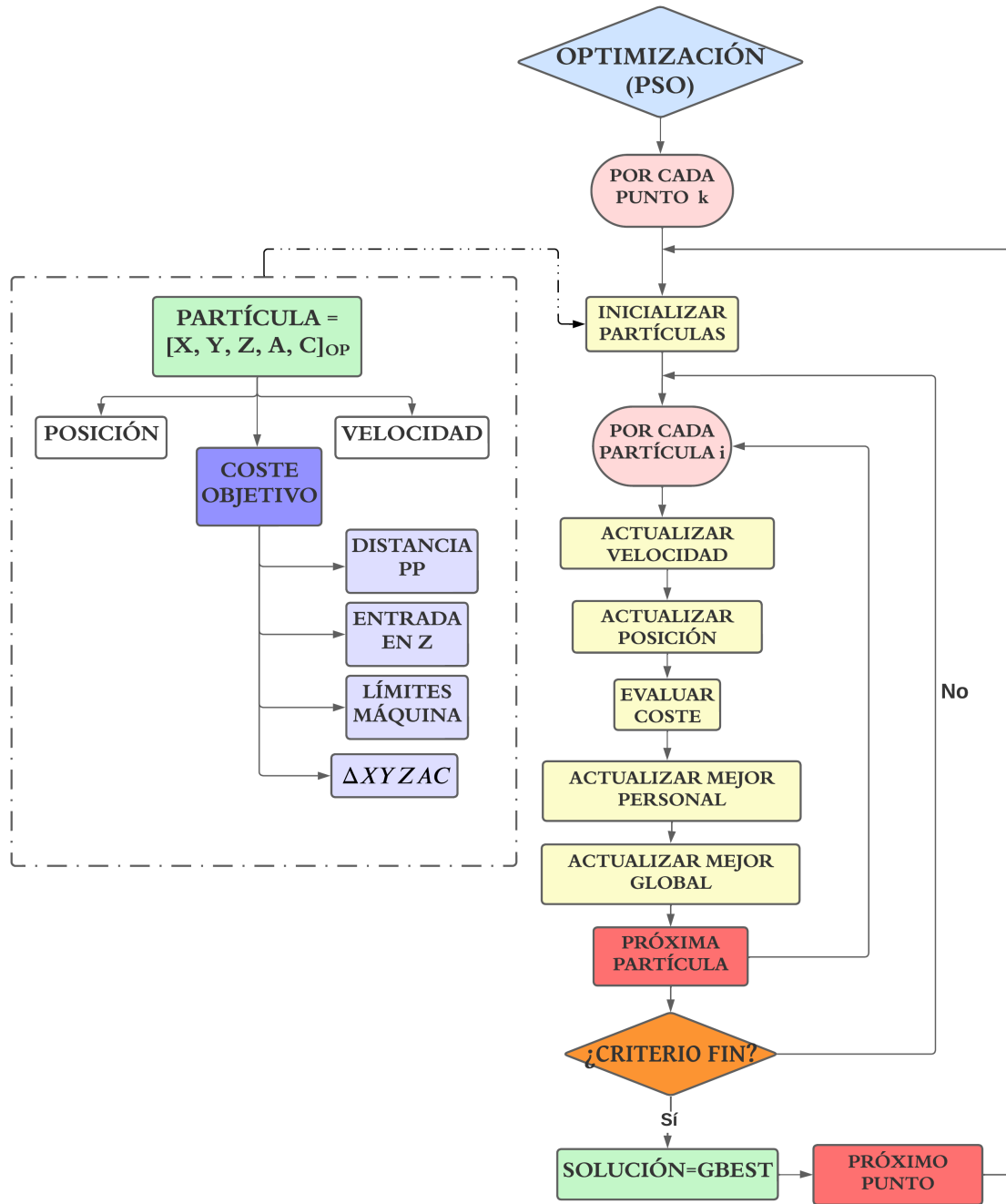


Figura 26: Esquema del algoritmo implementado. (Elaboración propia)

## 6.2. Función coste u objetivo

El principal objetivo de la optimización es hallar unos valores de posiciones de máquina  $[X, Y, Z, A, C]_{OP}$ , que logren una punta de herramienta  $P_{OP}$  lo más cercana al punto objetivo  $P_{OBJ}$ , que es la punta de la herramienta extraída del CAM ( $[x, y, z]_{SCP}$  del GOTO).

Para evaluar lo buena que es una solución, es necesario definir la función objetivo correctamente, con términos que representen lo que se quiere minimizar. Las partículas se encargarán de buscar distintas soluciones que puedan ser factibles en la resolución del problema.

Para evitar soluciones sin sentido físico hay que tener en cuenta el contacto que se produce entre herramienta y pieza, lo cuál es un problema de comprobación de colisiones entre objetos tridimensionales. También es necesario controlar transición con la que se producen los cambios de las posiciones de la máquina para evitar cambios bruscos en las mismas, ya que no se tiene control sobre las aceleraciones ni las velocidades de la máquina.

Estas dos cuestiones se pueden añadir como restricciones al algoritmo, pero es más sencillo introducirlas como costes adicionales en la función objetivo. Para que el enjambre consiga encontrar soluciones dentro de estas limitaciones, se dispara el coste si se sale de ellas y se aumenta la capacidad de búsqueda haciendo que jueguen todas las opciones y penalizando a las que no cumplen los costes.

En el presente trabajo se han definido cuatro aportaciones al coste, que están ponderados por importancia por un factor  $F_\alpha$  debido a la diferencia de magnitud de los distintos costes. Se ha escogido la métrica para la función coste de distancia euclídea, ecuación 33. Se han elegido los siguientes términos a minimizar:

- **Distancia PP:** Este término representa la distancia euclídea entre punta de herramienta optimizada ( $P_{OP}$ ) y punta de herramienta objetivo ( $P_{OBJ}$ ).
- **Entrada Z:** Representa la distancia máxima que entra la herramienta dentro de la pieza, en el eje Z. Se podría hablar de un término relacionado con la rugosidad, pero al ser ésta un problema no trivial se decide calcular la distancia en Z únicamente, se explica más adelante como se ha resuelto.
- **Límites:** Controla que los movimientos optimizados  $[X, Y, Z, A, C]_M$  no se salgan de los rangos físicos de los ejes de la máquina. El límite a tener en cuenta ha sido el del giro en **A** que debe oscilar entre  $\pm 120^\circ$ , mientras que el giro en **C** es completamente libre.
- **$\Delta XYZAC$ :** Indica el incremento de las posiciones de la máquina en un punto respecto de la mejor solución del anterior punto de la trayectoria. Permite controlar la transición de los movimientos, suavizando los mismos.

$$f_{COSTE} = \sqrt{F_D \cdot DistanciaPP^2 + F_R \cdot EntradaZ^2 + F_\Delta \cdot \Delta XYZAC^2 + F_{LIM} \cdot Limites^2} \quad (33)$$

Finalmente se ha añadido un reset del enjambre (reinicialización de todas las partículas, reducción de coeficiente cognitivo  $c1$  y aumento de la inercia  $w$ ) en cuanto se detectaba un atascamiento en un mínimo local. Esto es, si el valor de la función coste no varía durante un número determinado de iteraciones, se activa el reset y se reinicializa el algoritmo.

### 6.3. Lectura de la herramienta en Matlab y detección de colisiones.

La manera más fácil para detectar las colisiones entre objetos tridimensionales, es que ambos objetos se puedan aproximar por esferas. Si la distancia entre sus centros es menor que la suma de los radios de las esferas, eso indica que hay colisión. Sin embargo en el caso del presente proyecto no es posible aproximar la superficie paraboloidal por una esfera.

Para detectar colisiones entre objetos de geometría complicada existen muchos algoritmos en la literatura, [28-30]. Todos ellos se basan en el principio de representación de los objetos mediante puntos, mallado de facetas y búsqueda por distancias entre puntos de un objeto y otro.

En cambio en el presente proyecto no es computacionalmente viable usar un algoritmo que a su vez esté ejecutando otro algoritmo de chequeo de interferencias. Esto es porque en cada partícula se chequeará la colisión, para todas las partículas y después durante varias iteraciones se repetirá el proceso.

Por ello se ha encontrado una solución mucho más rápida, que implica lo siguiente: Se van a mallar ambos objetos (herramienta y pieza) como nubes de puntos. Esto se puede realizar ya que como la superficie está definida matemáticamente se puede obtener cualquier punto de la misma sin necesidad de mallarla.

Para mallar la herramienta, se ha dibujado en CAD y se importa a Rhino, donde se hará un mallado cuadrangular del NURBS de la herramienta definiendo así el grado de refinamiento, esto con el fin de exportarla como STL.

El STL aporta información variada tal como las coordenadas de los vértices, el orden de unión entre ellos y la conectividad. Para chequear las colisiones será necesario únicamente lo primero. En la figura 27 (a) se muestra este proceso de diseño y lectura de los puntos en Matlab, la subfigura 28 (b) muestra la herramienta movida y orientada satisfactoriamente según la normal en un punto de la superficie.

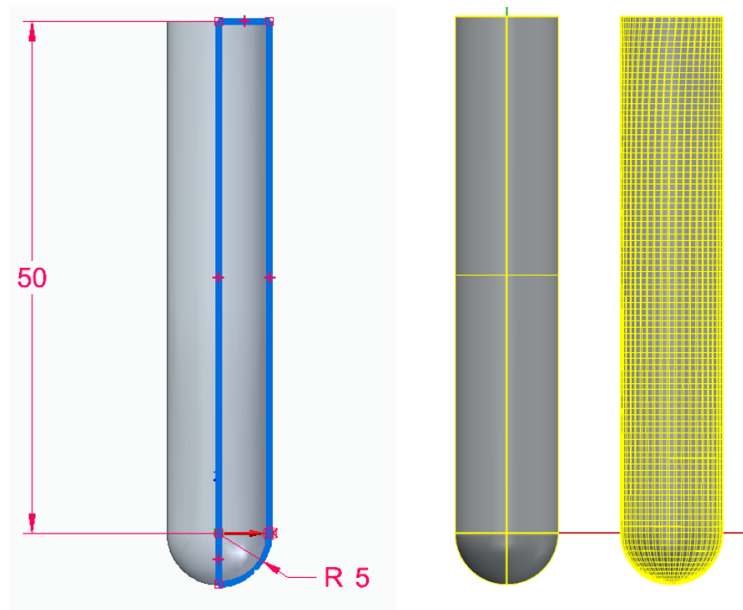


Figura 27: Herramienta diseñada en CAD y remallada en Rhino finalmente. (Elaboración propia).

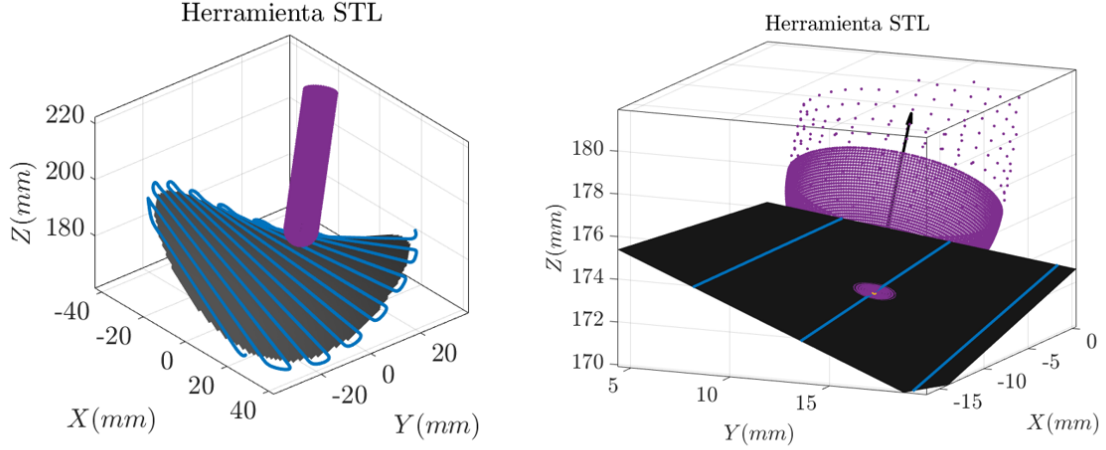


Figura 28: Herramienta leída como conjunto de puntos en Matlab. (Elaboración propia).

La función que realiza el movimiento de la herramienta, primero gira ésta A y después C grados con las matrices de rotación 10. Efectuando el giro alrededor del centro de la bola esférica, por esto se ha diseñado en CAD con el origen aquí. Después se desplaza con las matrices de traslación de la ecuación 10 al punto que representa su centro, el cuál se calcula a partir de la punta de la herramienta desplazándolo el radio de la misma en la dirección de su eje, ecuación 34.

$$\mathbf{Centro}_H = \mathbf{Punta}_H + \mathbf{normal}_H \cdot \text{Radio}_H \quad (34)$$

Como la superficie no tiene zonas convexas se puede obtener la Entrada en Z como la distancia que hay entre los puntos de la herramienta y esos mismos proyectados sobre la superficie teórica con la ecuación del paraboloide 3. A efectos prácticos, si es negativa significa que la herramienta ha penetrado en la superficie, y positiva si se ha quedado por encima, ecuación 35.

$$\text{Entrada}Z = Z_{STL} - Z_{\text{paraboloide}}(X_{STL}, Y_{STL}) \quad (35)$$

A los propios puntos de la herramienta importada como STL, se crea una esfera teórica de radio 5mm, y se calculan unos puntos alrededor del punto objetivo, lo cuál permitirá un refinamiento mucho mayor en las cercanías de la zona de contacto herramienta-pieza, de lo que permite el remallado cuadrangular en Rhino, ecuación 36. Se calculan éstos puntos en la esfera teórica y en la superficie del paraboloide y se comprueba la interferencia con la misma ecuación que en 35.

$$(X_{ESF} - X_{\text{centro}})^2 + (Y_{ESF} - Y_{\text{centro}})^2 + (Z_{ESFERA} - Z_{\text{centro}})^2 = R^2 \quad (36)$$

En la figura 29, se puede ver un esquema en 2D de lo que se hace en la optimización y los términos de la función coste a optimizar (Distancia PP y Entrada Z). La orientación de la herramienta optimizada se deja libre con la condición de que la entrada en Z no sea mayor que una determinada tolerancia, por encima y debajo de la superficie. Eso sí, se penaliza más que la herramienta entre en la superficie, ya que algo que se ha mecanizado no se puede recuperar.

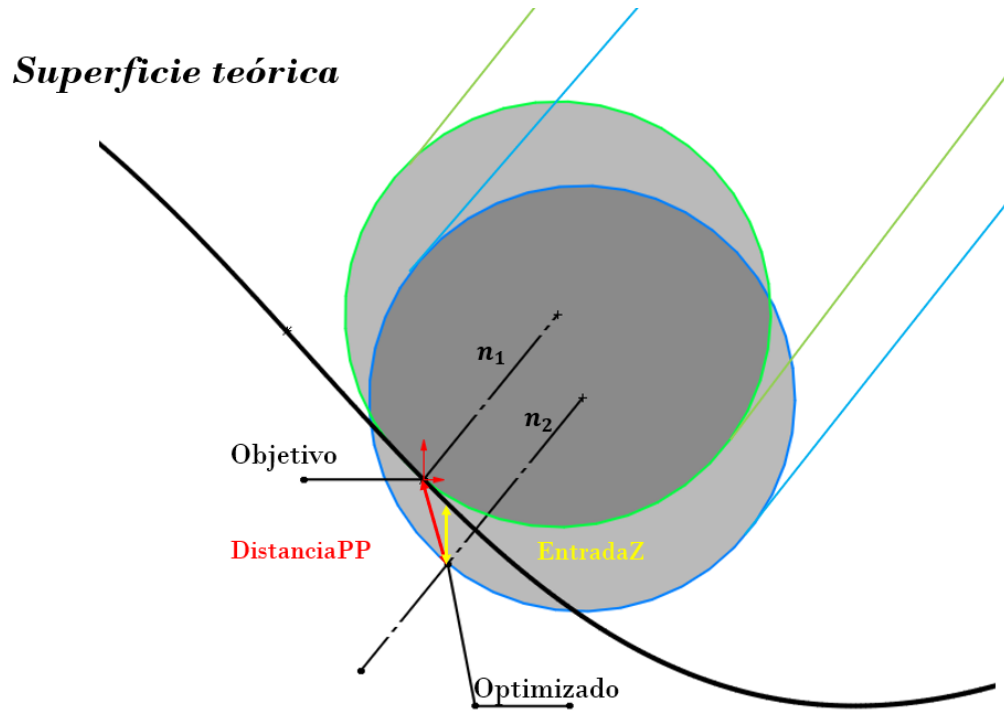


Figura 29: Esquema de la punta de herramienta optimizada frente a la objetivo, distancia punta optimizada y punta objetivo (rojo), entrada en Z (amarillo). (Elaboración propia).

# 7. Resultados y discusión

## 7.1. Situación inicial

En la gráfica 30 (a) se representan los valores del postprocesador del CAM (POST), con el eje izquierdo representando las posiciones de traslaciones (XYZ) en mm, y el derecho representa las rotaciones (AC) en grados, a lo largo de la trayectoria (580 puntos). En dicha figura se puede contar el número de pasadas, por ejemplo contando los picos del parámetro X, se ve que son 13, iguales que el número de arcos que hay en la trayectoria 30 (b).

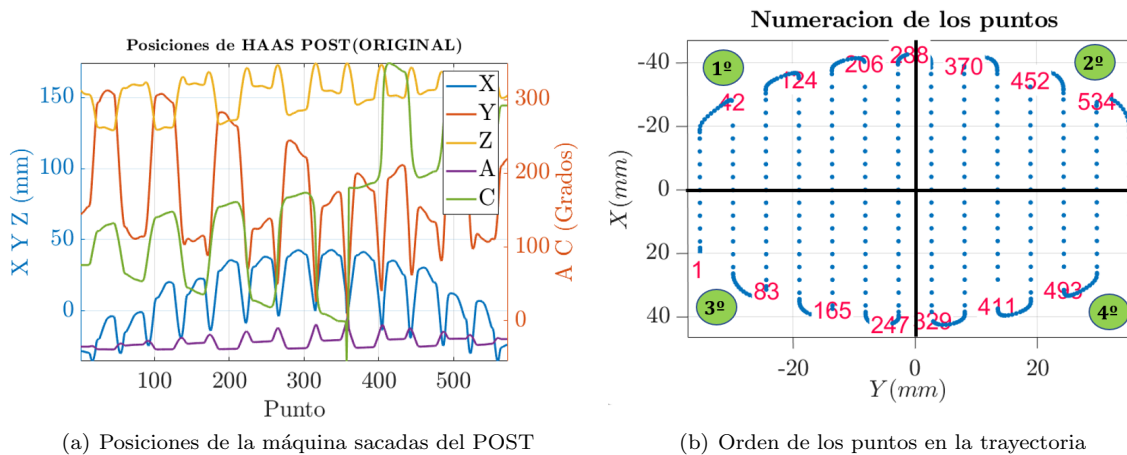


Figura 30: Posiciones de máquina y trayectoria extraídos del CAM. (Elaboración propia).

Esto es así, porque la estrategia ha sido perpendicular a la superficie y la cinemática inversa, tanto para la punta de herramienta como su orientación están relacionadas con la geometría de la superficie. Pero puede haber múltiples formas de combinar las distintas posiciones de la máquina  $([X, Y, Z, A, C])$  y lograr el mismo punto final de herramienta sin que las gráficas de las posiciones sigan la forma de la trayectoria.

La situación inicial con los errores geométricos sin optimizar, es la que se representa en la figura 31, tanto los valores de error como distancia en cada punto de la trayectoria (a) como su análisis por cuadrantes (b). Estos errores serían los que se obtendrían en el mecanizado de la pieza si se mandasen a la máquina las consignas del CAM directamente.

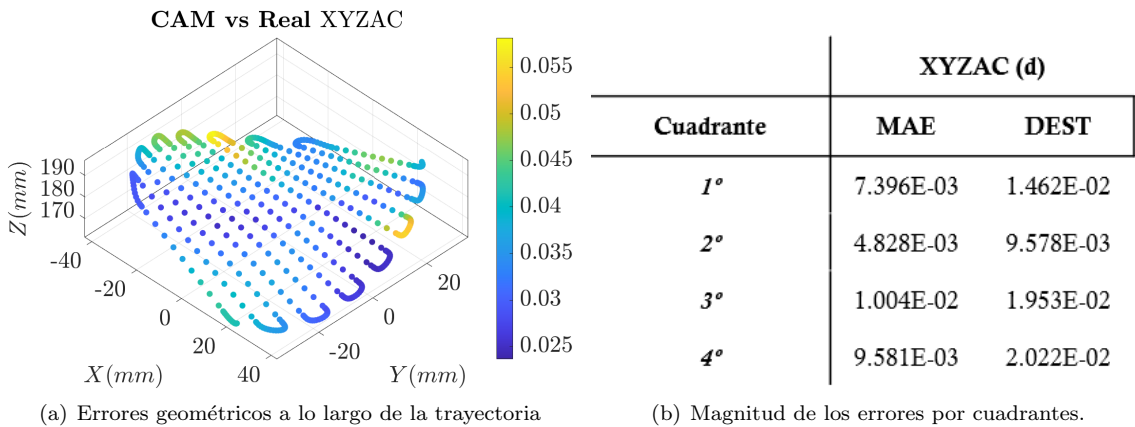


Figura 31: Errores geométricos (mm) inicialmente, previo a la optimización. (Elaboración propia).

## 7.2. Calibración del algoritmo

Para comprobar la sensibilidad del algoritmo y los parámetros que obtienen un mejor resultado se van a llevar a cabo varias simulaciones cambiando condiciones del algoritmo. Tales como pueden ser las soluciones iniciales, la inercia de las partículas, los coeficientes cognitivo y social. Las siguientes ajustes del algoritmo se han elegido a estudiar:

### 7.2.1. Parámetros del algoritmo y hardware

- Reducción de la inercia (**w**) y el parámetro cognitivo (**c1**) a valores iniciales de 0,3 y (**c2**) en 2, para todos los casos. Se ha comprobado que valores de inercia y coeficiente cognitivo altos hacían que el algoritmo fuese menos eficaz.
- Se han escogido un total de 70 **partículas** para formar el enjambre, irse mucho más de este número no ofrecía ventajas y aumentaba el tiempo de computación.
- El número de **iteraciones** por defecto era de 100, pero si se lograba obtener la tolerancia requerida se daba por finalizada la búsqueda, en cambio si se detectaba un atascamiento en algún mínimo local se aumentaba el número de iteraciones.
- Por cada caso simulado se repetía la simulación, generando varias réplicas y se selecciona una de ellas para analizar.
- El **hardware** usado para este proyecto ha sido un portátil con un *i7 – 9750H* a *2,58Ghz* y con *16Gb* de RAM.
- Con todos los parámetros anteriores el **tiempo** de computación es aproximadamente 30 minutos para cada caso simulado.

### 7.2.2. Soluciones iniciales de las partículas

Otro factor importante que influencia al algoritmo es la solución inicial que se le da a las partículas, lo cuál influenciará el movimiento que tenga el enjambre. Se han decidido analizar de entre las posibles soluciones iniciales:

1. **si post (CAM)**: En este caso se le daba como solución inicial soluciones aleatorizadas alrededor de los parámetros de máquina que sacaba el postprocesador del CAM.
2. **si gbest (mejor del punto anterior)**: Aquí en el primer punto de la trayectoria se le daba como solución inicial código NC del CAM (POST) y para los siguientes puntos se cogía la mejor solución en el punto anterior ( $gbest_{i-1}$ ) de la trayectoria.

### 7.3. Control de trayectoria

Para controlar la brusquedad de los movimientos de la máquina existen varias formas. Una de ellas es alinear el vector de velocidad lineal en la punta de la herramienta de manera que sea tangente con la trayectoria. Esto no se va a realizar en este proyecto ya que no se tienen en cuenta las velocidades y aceleraciones de la máquina.

Para ello se han elegido controlar la trayectoria, ajustando la transición de los parámetros  $[X, Y, Z, A, C]$  de la máquina, con incrementos de las mismas de un punto respecto del anterior en la trayectoria.

#### 7.3.1. Cálculo del error mínimo sin control de la trayectoria

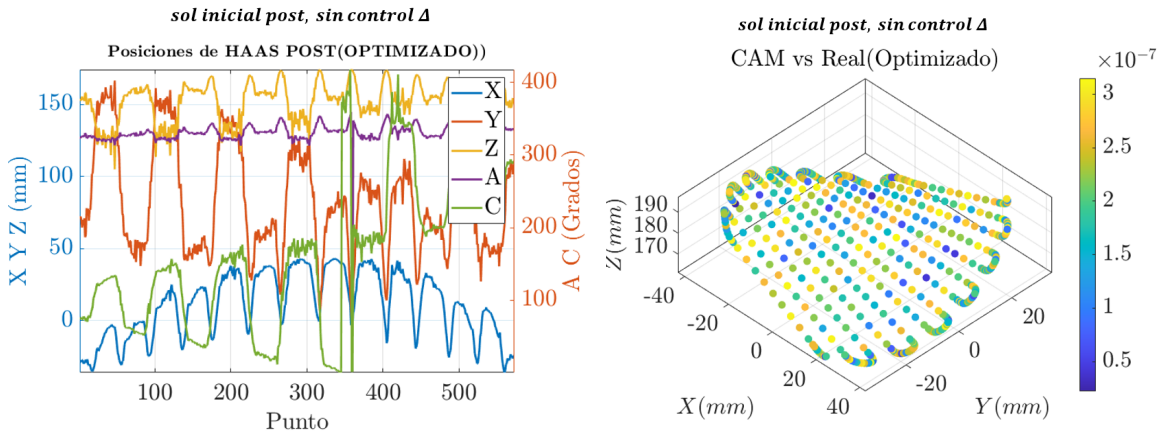
En primer lugar se va a estudiar los casos sin aplicación del control de trayectoria. Esto se hace quitando del término función coste la influencia de la trayectoria, dejando la función coste únicamente en función de la distancia PP, la EntradaZ y los límites, ecuación 37.

$$f_{COSTE} = f(\text{DistanciaPP}, \text{EntradaZ}, \text{Límites}) \quad (37)$$

Esto va a resultar en el error mínimo que se va a lograr de entre todas las simulaciones ya que se le da mayor libertad de búsqueda al algoritmo y no se le restringen las posibles soluciones con los incrementos  $(\Delta XYZAC)$ . Se van a evaluar los siguientes casos sin estos incrementos:

- Solución inicial el CAM (**POST**): En la figura 32 se muestra en (a) las posiciones de la máquina optimizadas a lo largo de la trayectoria y en (b) se muestra la magnitud de los errores en cada punto de la misma.

Los perfiles de las posiciones de la máquina de la figura 32 (a) son similares a la situación inicial del CAM figura 30 (a). Esto es porque se le ha dado a la optimización soluciones, que están cerca de las consignas del CAM y lo que hace el algoritmo es encontrar soluciones alrededor de las mismas sin tener en cuenta los anteriores movimientos ni la trayectoria.



(a) Posiciones de la máquina optimizadas a lo largo de la trayectoria (b) Magnitud de los errores en cada punto de la trayectoria

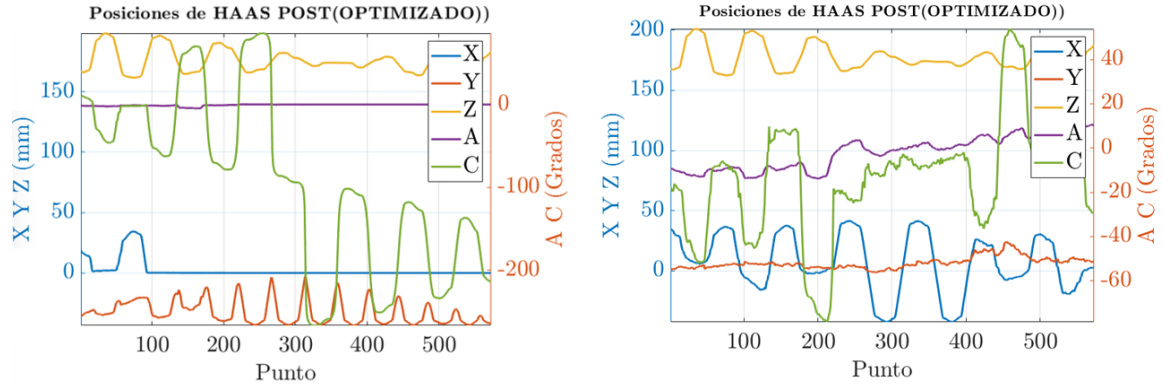
Figura 32: Resultados con solución inicial post (CAM) sin control de incrementos. (Elaboración propia).

- Solución inicial el mejor punto anterior (**gbest**): Se han hecho varias simulaciones para este caso, se ha escogido mostrar dos soluciones más diferentes entre sí, véase figura 33 (a). En la de la izquierda se mantienen prácticamente constantes X y A, y en dónde no es

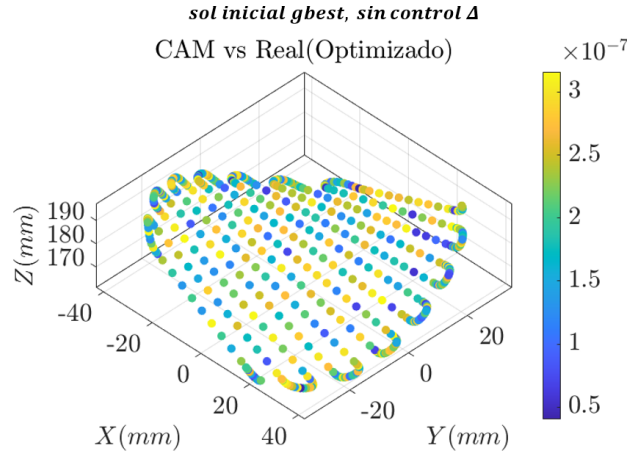
suficiente para reducir el error, el algoritmo elige mover X. En cambio en la de la derecha, encuentra una solución moviendo constantemente todas las posiciones de la máquina.

Esto muestra que es posible realizar movimientos de la trayectoria de formas muy distinta logrando errores de un mismo orden de magnitud. En la figura 33 (b) se muestra el resultado de los errores para el caso de posiciones 33 (a) izquierdo.

***sol inicial gbest, sin control  $\Delta$***



(a) Posiciones de la máquina optimizadas a lo largo de la trayectoria



(b) Magnitud de los errores en cada punto de la trayectoria

Figura 33: Resultados con solución inicial gbest sin control de incrementos. (Elaboración propia).

En ambas simulaciones con distintas soluciones iniciales se han obtenido errores del orden de  $1e-7mm$  en distancia. En cuanto a la forma que tienen los perfiles de las posiciones de la máquina, las que mejor se comportan son las que tienen solución inicial el gbest incluso sin haber metido el control de trayectoria, ya que tiene menos picos bruscos, véase figura 33 (a).

### 7.3.2. Cálculo del error mínimo con control de la trayectoria

Ahora se va a introducir el término que ayuda con el control de la trayectoria y evita los picos bruscos de las posiciones de la máquina a lo largo de la trayectoria. Es decir, la función coste tendrá en cuenta también el término adicional de control de los incrementos de las posiciones ( $\Delta XYZAC$ ), ecuación 38.

$$f_{COSTE} = f(\text{DistanciaPP}, \text{EntradaZ}, \text{Limites}, \Delta XYZAC) \quad (38)$$

Se van a analizar en total 6 simulaciones, variando la solución inicial del enjambre y variando los deltas a controlar. El control de trayectoria se basa en evitar que el punto actual de optimización esté un determinado incremento por encima o debajo del valor de ese eje en el punto anterior de la trayectoria. Cuanto más pequeño sea este incremento más difícil será encontrar una solución. Se han simulado los siguientes casos:

- Solución inicial **POST**:

- Lineales ( $\Delta XYZ$ ): Aquí se controlaban únicamente los incrementos en los ejes de traslación, en la figura 34 se representan las consignas optimizadas (a) y la magnitud del error en cada punto de la trayectoria (b).

Comparando este caso con el caso de misma solución inicial (CAM) pero sin control de trayectoria, se observa que los movimientos ahora son más suaves, figura 32. Pero los errores son mayores, debido a la restricción adicional de los incrementos, se llega con ello a errores máximos del orden de  $1e - 5mm$  en contraste con la estrategia sin control de trayectoria con errores del orden de  $1e - 7mm$ .

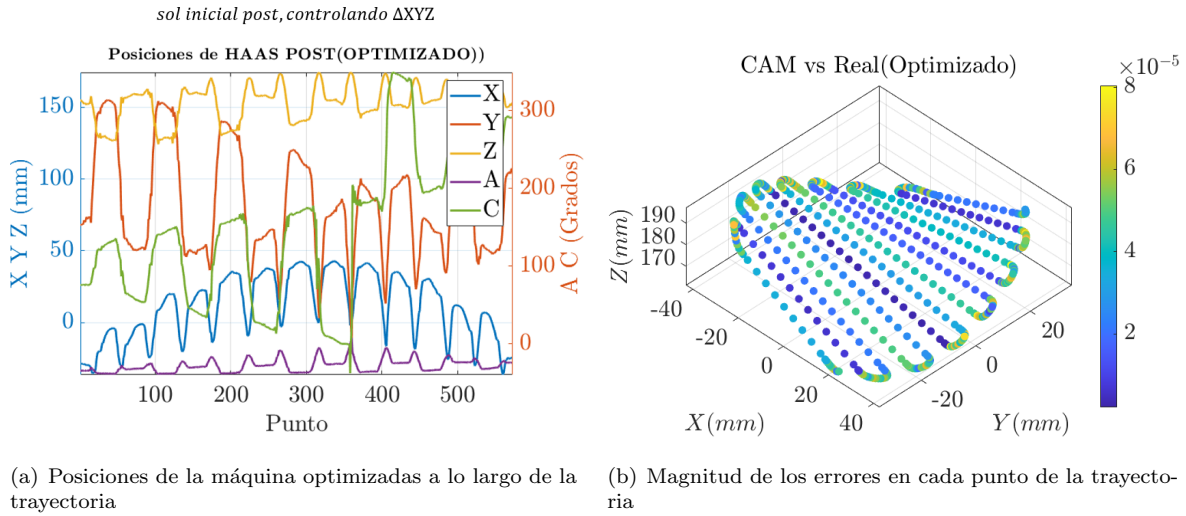


Figura 34: Resultados de posiciones de máquina con solución inicial el **POST** del CAM, controlando incrementos de los ejes lineales. (Elaboración propia).

- Los ejes de rotación ( $\Delta AC$ ): En esta se controlan los giros con los ángulos A y C, obteniendo los resultados de la figura 35.

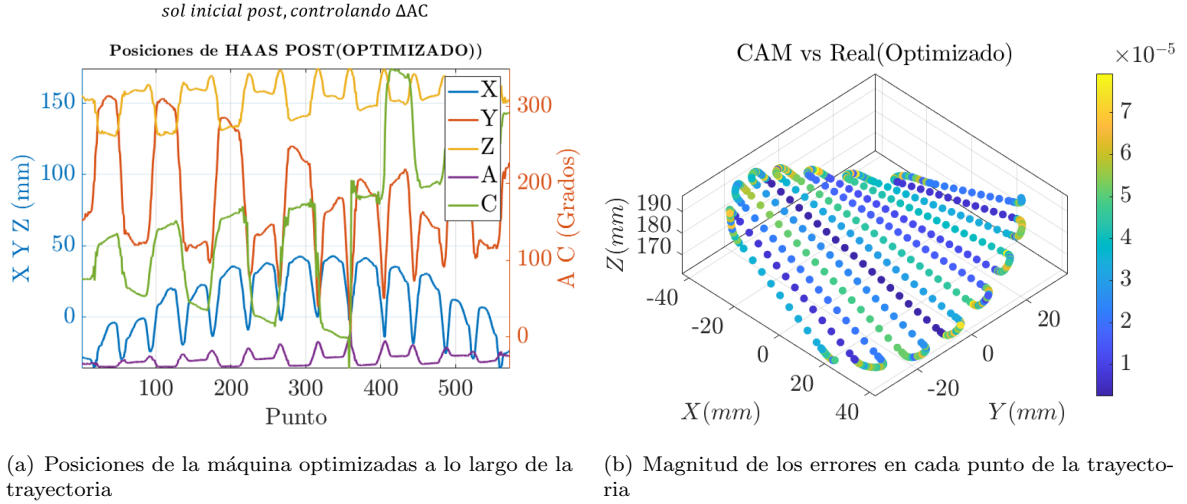


Figura 35: Resultados de posiciones de máquina con solución inicial el **POST** del CAM, controlando incrementos de los ejes de rotación. (Elaboración propia).

- Todos ( $\Delta XYZAC$ ): Aquí se controlan los incrementos de todos los ejes o posiciones de la máquina, resultados en la figura 36.

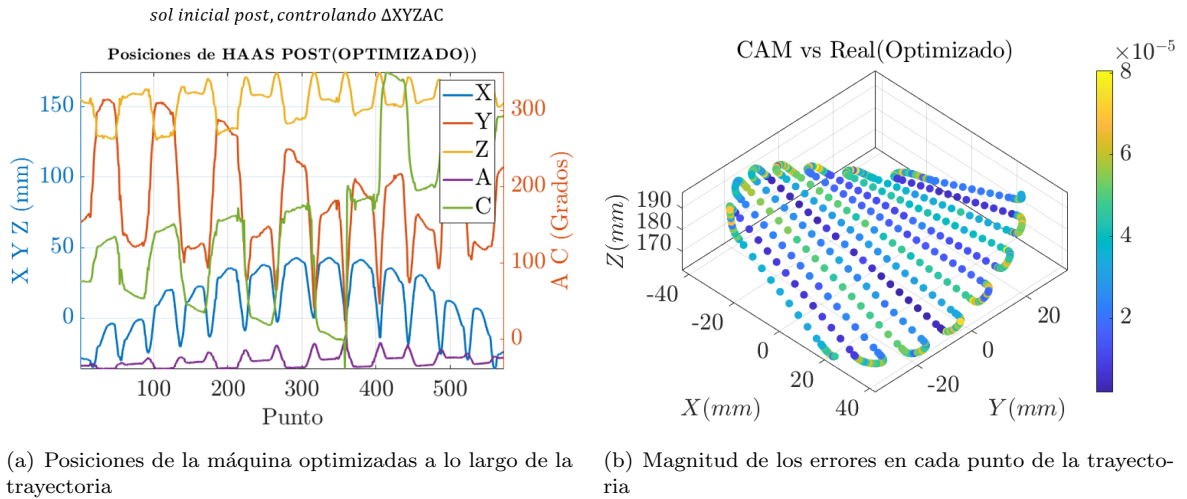


Figura 36: Resultados de posiciones de máquina con solución inicial el **POST** del CAM, controlando incrementos de todos los ejes. (Elaboración propia).

Los errores y la forma de las posiciones de la máquina son muy similares para los tres casos con esta misma solución inicial. Lo que está haciendo el algoritmo es, meter pequeñas perturbaciones a las distintas posiciones de la máquina en las que no se está controlando su incremento, para con ello lograr un error mínimo de mecanizado.

- Solución inicial **GBEST**: Ahora se empiezan a ver soluciones más variadas, ya que se le da mayor capacidad de búsqueda al algoritmo al dar como solución inicial la mejor del punto anterior de la trayectoria.
  - Lineales ( $\Delta XYZ$ ): En la figura 37 se muestran los resultados para este caso. Se observa que hay zonas en las que el algoritmo decide dejar algunos ejes quietos, esto ocurre porque dichas posiciones pasan por el 0, que es donde los errores por definición son nulos, gráfica 11 y 12.

En cambio, hay zonas por ejemplo en 37 (b), en el tramo de puntos de 340 a 400, donde no es posible corregir los errores manteniendo la posición X quieta en 0.

Después en el punto 400 la tendencia del ángulo C hace que pase por el 0 y este reduce su error lo deja quieto y entonces es necesario empezar a mover X acompañado del cambio de tendencia que llevaba Y, el cuál pasa de un movimiento de vaivén a ir aumentando su valor gradualmente.

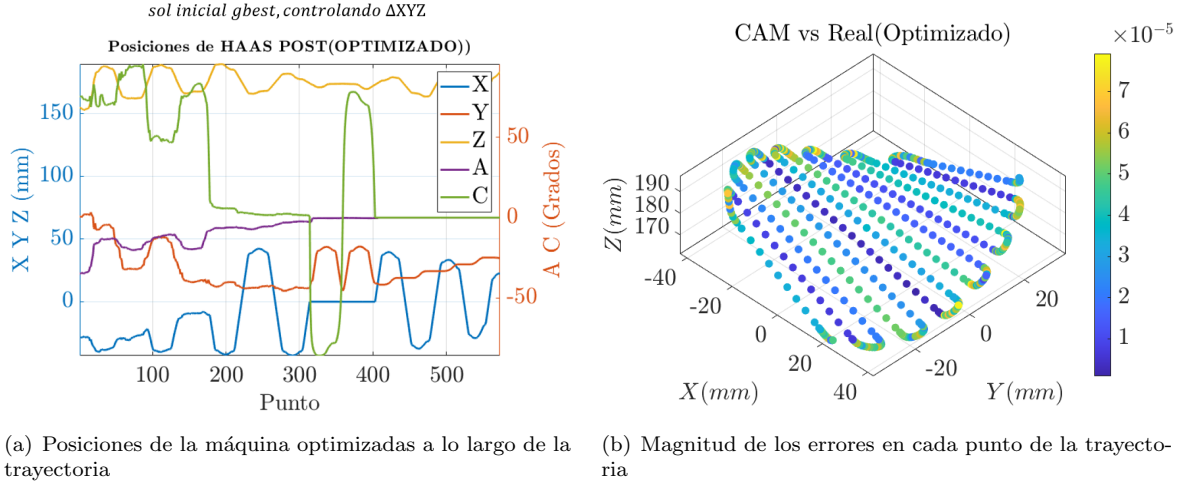


Figura 37: Resultados de posiciones de máquina con solución inicial el **GBEST**, controlando incrementos de los ejes lineales. (Elaboración propia).

- Los ejes de rotación ( $\Delta AC$ ): En este caso, como se observa en los perfiles 38, se deja la posición X quieta cuando llega al punto 120 y lo vuelve a mover cuando no es posible reducir el error con los otros cuatro ejes. La magnitud de los errores es similar a las simulaciones anteriores con control de trayectoria.

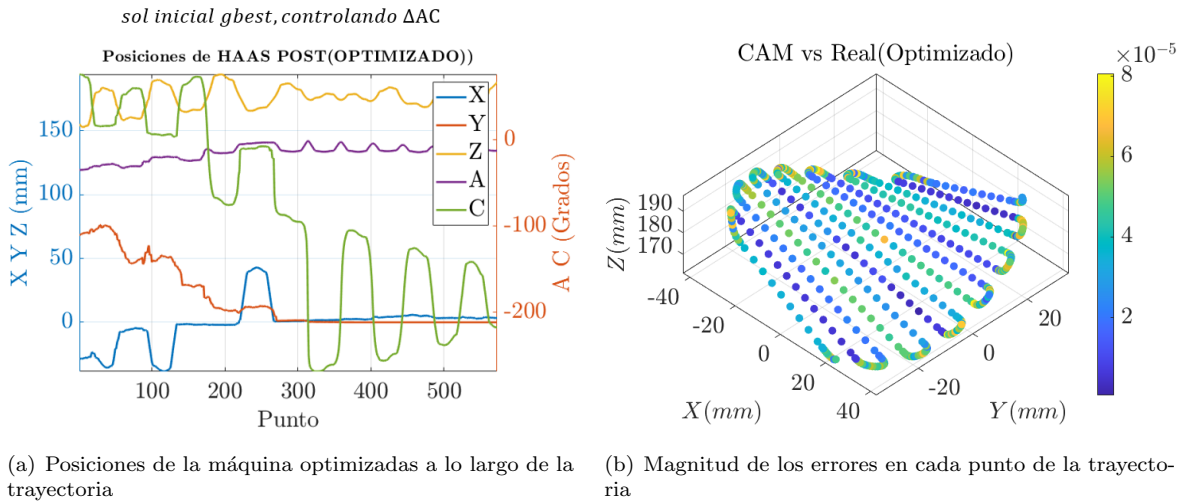
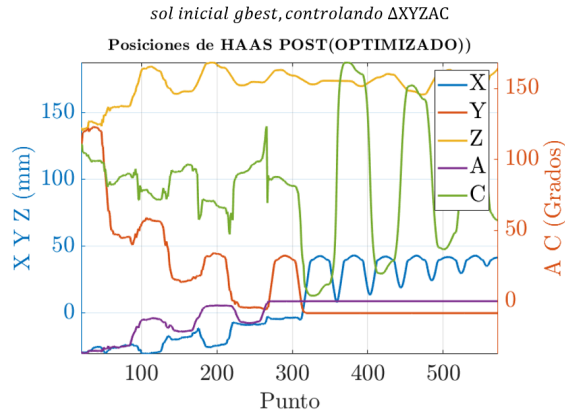
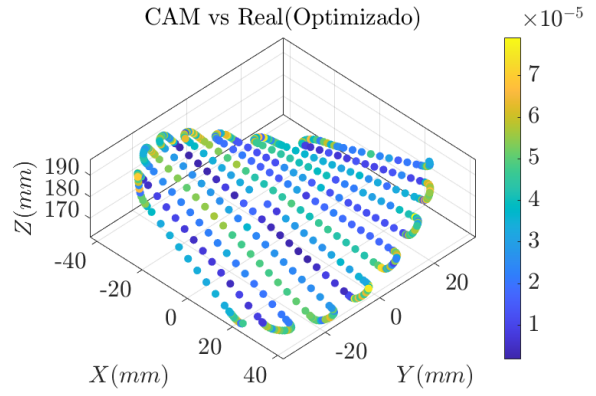


Figura 38: Resultados de posiciones de máquina con solución inicial el **GBEST**, controlando incrementos de los ejes de rotación. (Elaboración propia).

- Todos ( $\Delta XYZAC$ ): Aquí se controlan todos los incrementos con los resultados en la figura 39. En este caso el algoritmo decide dejar quietas las posiciones Y y A, ambos con valor nulo, a partir del punto 300. Lo cuál es acompañado con el cambio en tendencia que llevaban los ejes X y C, que pasan a moverse en vaivén.



(a) Posiciones de la máquina optimizadas a lo largo de la trayectoria



(b) Magnitud de los errores en cada punto de la trayectoria

Figura 39: Resultados de posiciones de máquina con solución inicial el **GBEST** , controlando incrementos de todos los ejes. (Elaboración propia).

En general en todos los casos se logra que las posiciones de la máquina sigan una trayectoria sin cambios bruscos, eso sí aumentando los errores dos órdenes de magnitud frente a los casos sin control de trayectoria.

#### 7.4. Análisis de métricas

Los dos términos que más importancia tienen para la medición de lo bien que se ha hecho el mecanizado son: La distancia entre punta de herramienta objetivo y optimizada (Distancia PP) y la entrada máxima en dirección Z de la herramienta en la superficie (Entrada Z), veáse el esquema 29. Son los términos que cuánto menor sean, mejor será el mecanizado real de la pieza.

Es por ello que se presentan a continuación los resultados de ambos términos para los distintos casos de simulación, con todas las métricas en milímetros, veáse las tablas 4. Se representa en la tabla (a) las distintas métricas sobre todos los puntos de la trayectoria y diferentes casos de simulación, para el término de **Distancia PP** y en la tabla (b) se representan sobre el término de entrada en Z.

La optimización ha sido satisfactoria en todos los casos. Ya que se han pasado de errores geométricos de la máquina del orden de  $1e - 2mm$  a unos errores del orden de  $1e - 6mm$  en las estrategias optimizadas para la distancia PP.

Además, la entrada en Z de la herramienta si se hubiese realizado el mecanizado con el código original que saca el CAM directamente y sin compensación, habría resultado en gran pérdida de precisión en la pieza, con entradas del orden de  $1e - 1mm$  y máxima de  $4mm$ . En cambio con el postprocesador adicional que se ha diseñado, ésta misma variable es del orden de  $1e - 7mm$ .

(a) Valores para la Distancia PP

<b>DistanciaPP</b>					
	<i>MAE</i>	<i>RMSE</i>	<i>MIN</i>	<i>MAX</i>	<i>DEST</i>
<b>Original (CAM)</b>	<b>5.391E-02</b>	<b>5.552E-02</b>	<b>2.179E-02</b>	<b>7.647E-02</b>	<b>5.557E-02</b>
<i>_post_sin_control</i>	3.369E-07	1.444E-06	7.920E-08	2.515E-05	1.446E-06
<i>_gbest_sin_control</i>	4.328E-07	2.259E-06	6.871E-08	4.778E-05	2.260E-06
<i>_gbest_control_ac_</i>	1.120E-05	2.098E-05	3.507E-07	1.880E-04	2.100E-05
<i>_gbest_control_xyz_</i>	5.566E-06	9.996E-06	1.244E-07	8.642E-05	1.000E-05
<i>_gbest_control_xyzac_</i>	1.049E-05	2.151E-05	2.527E-07	2.321E-04	2.153E-05
<i>_post_control_ac_</i>	8.894E-07	3.242E-06	5.781E-09	5.260E-05	3.245E-06
<i>_post_control_xyz_</i>	3.197E-06	5.353E-06	1.701E-07	4.615E-05	5.358E-06
<i>_post_control_xyzac_</i>	7.814E-06	2.606E-05	2.348E-07	4.831E-04	2.608E-05

(b) Valores para la Entrada en Z

<b>EntradaZ</b>					
	<i>MAE</i>	<i>RMSE</i>	<i>MIN</i>	<i>MAX</i>	<i>DEST</i>
<b>Original (CAM)</b>	<b>4.963E-01</b>	<b>9.882E-01</b>	<b>-7.395E-03</b>	<b>4.382E+00</b>	<b>9.891E-01</b>
<i>_post_sin_control</i>	7.261E-08	1.168E-07	4.431E-11	3.615E-07	1.169E-07
<i>_gbest_sin_control</i>	6.628E-08	1.076E-07	3.007E-10	3.494E-07	1.077E-07
<i>_gbest_control_ac_</i>	1.181E-06	2.698E-06	1.612E-11	1.240E-05	2.700E-06
<i>_gbest_control_xyz_</i>	5.418E-07	1.340E-06	2.785E-12	8.002E-06	1.341E-06
<i>_gbest_control_xyzac_</i>	1.001E-06	2.570E-06	1.663E-11	1.867E-05	2.572E-06
<i>_post_control_ac_</i>	7.681E-08	2.397E-07	2.842E-11	4.038E-06	2.399E-07
<i>_post_control_xyz_</i>	3.269E-07	7.694E-07	8.527E-14	6.290E-06	7.701E-07
<i>_post_control_xyzac_</i>	9.963E-07	2.636E-06	3.638E-12	2.135E-05	2.638E-06

Tabla 4: Resultados de las métricas elegidas

Ahora se procede a seleccionar la mejor estrategia de optimización teniendo en cuenta el análisis del capítulo anterior y las métricas para cada caso dispuestas en las tablas 4. Se observa lo siguiente:

- Los casos en los que no se controla la trayectoria se obtiene buenos resultados en comparación con el resto de simulaciones en las que se controlan los incrementos. Como se había explicado anteriormente, son los casos que definen la cota mínima de todas las simulaciones.
- De los casos con control de trayectoria, en general las que mejores resultados en término de magnitud del error son las que tienen de solución inicial el **post**.
- Respecto a los ejes en los que se controla el incremento, la que destaca es la simulación controlando los ejes de giro AC ( $\Delta AC$ ) y con solución inicial el **post**, ya que obtiene un orden de magnitud menor en comparación con el resto.
- Los valores de Entrada en Z para el CAM original son positivos y negativos, mientras que en la optimización son todos positivos, es decir, la herramienta se queda por encima de la superficie. Esto es porque se le penalizaba más si entraba dentro de la superficie, ya que algo mecanizado no se puede recuperar.

### 7.5. Elección de la mejor estrategia

La mejor estrategia ha sido sin duda aquella en la que no se controlaba la trayectoria con los incrementos, y se le daba de solución inicial la mejor del punto anterior. Esta estrategia es mejor por las siguientes razones:

- Las consignas son las que mejor transición logran de toda la optimización e incluso son más óptimas que la original del CAM, al no mover demasiado los ejes X y A, véase figura 40.

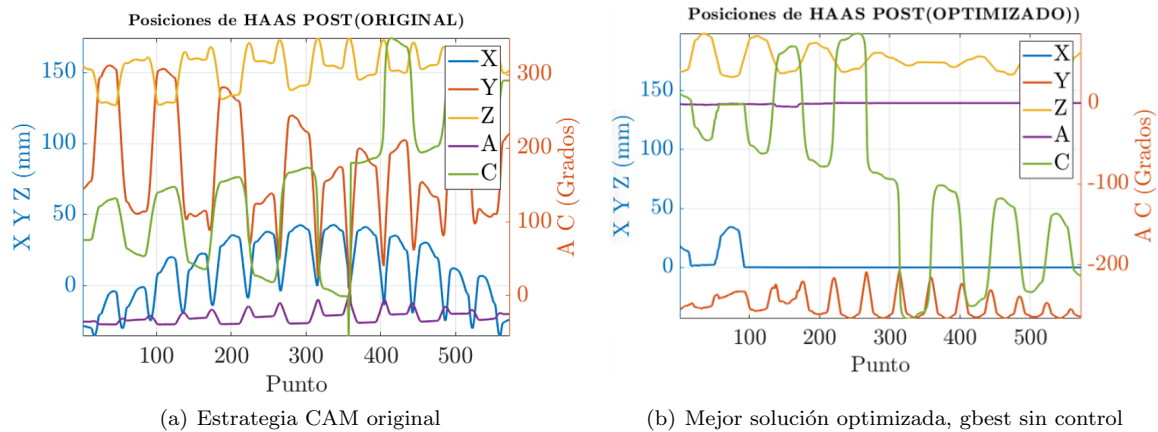


Figura 40: Comparación de consignas CAM vs Optimizada. (Elaboración propia).

- Los errores obtenidos son los menores entre todos los casos, junto con el caso sin control pero con solución inicial el post, éste no se elige por las transiciones con picos abruptos en los movimientos de los ejes. Se logra reducir los errores respecto del CAM cuatro órdenes de magnitud, véase figura 41.

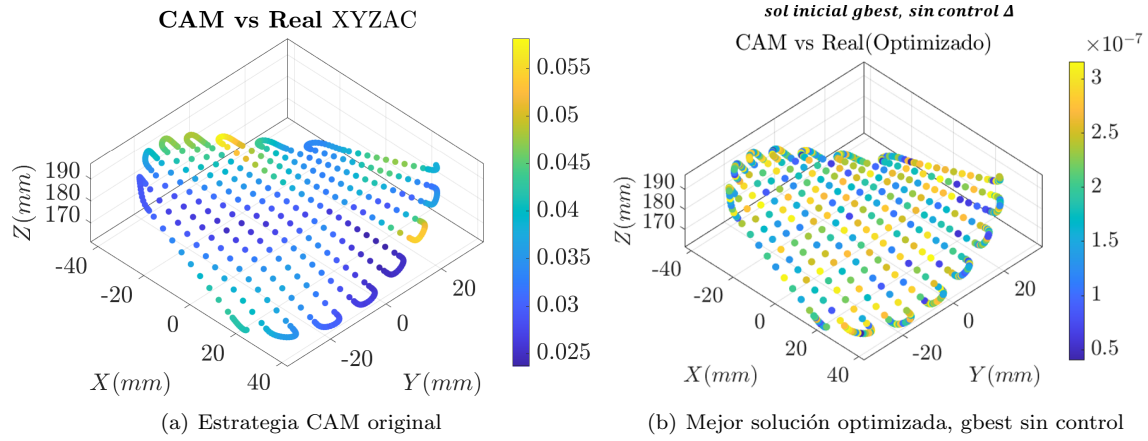


Figura 41: Comparación de errores geométricos CAM vs Optimizada. (Elaboración propia).

- Analizando los errores por cuadrantes para la estrategia de optimización elegida, tabla 5. Se obtiene que los cuadrantes con menor error son el primero y el cuarto, sin ser muy distintos del resto de cuadrantes, ya que queda la distribución de errores más homogénea en la optimizada, véase 41 (b). En cambio en la original del CAM (a), los errores son máximos en el cuadrante 2, pero en conjunto son mayores en el tercer cuadrante.

Cuadrante	CAM original		GBEST sin control	
	MAE	DEST	MAE	DEST
1°	7.396E-03	1.462E-02	4.938E-07	1.841E-06
2°	4.828E-03	9.578E-03	1.827E-06	2.101E-05
3°	1.004E-02	1.953E-02	1.117E-06	6.337E-06
4°	9.581E-03	2.022E-02	8.886E-07	5.093E-06

Tabla 5: Comparación de errores por cuadrante CAM vs Optimizada. (Elaboración propia).

## 8. Conclusiones y trabajo futuro

Las conclusiones y contribuciones según lo obtenido en el trabajo son:

- Se ha creado una metodología de optimización para una máquina herramienta HAAS de 3+2 ejes con la que se han logrado reducir sus errores geométricos en general más de 4 órdenes de magnitud. Dicha metodología se puede adaptar fácilmente para cualquier otra máquina para la que exista un postprocesador en el CAM y se sepa el modelo cinemático de la misma.
- Se ha automatizado el proceso de extracción de información relevante del CAM. Creando scripts para extraer todas las coordenadas relevantes contenidas en los ficheros GOTO y POST, pudiéndose incrementar el número de decimales con el que han sido generadas.
- Se ha creado una forma sencilla y computacionalmente eficiente para el chequeo del contacto entre herramienta pieza en Matlab. Permitiendo el diseño de cualquier herramienta y portaherramientas, con su lectura en Matlab y optimización del proceso de mecanizado. Para geometrías de pieza muy complejas se puede en un futuro programar un algoritmo específico para el chequeo de colisiones e interferencias.
- Se ha concluido que todas las estrategias que se han obtenido de la optimización logran reducir los errores y con los ajustes adecuados obtienen perfiles de los ejes de la máquina suaves. De las cuáles, la estrategia con menores cambios bruscos la presenta el resultado de la simulación sin control de incrementos, manteniendo los ejes X y A de movimiento prácticamente sin mover.
- El algoritmo de enjambre de partículas ha probado su eficacia para resolver este tipo de problemas, dando resultados con transiciones suaves y con métricas óptimas. Los tiempos de simulación han sido del orden de media hora por simulación y se ha modificado el algoritmo a medida sacándolo de mínimos locales fácilmente.
- Se estudiará como proyecto futuro, la creación de un simulador de la fresadora HAAS pudiendo simular el acabado tridimensional final previo al mecanizado. Conociendo y corrigiendo los errores geométricos se puede obtener una mayor aproximación al comportamiento real de la máquina, permitiendo así mejorar las estrategias de mecanizado.

## Referencias

- [1] Anthony Chukwuiekwu Okafor y Yalcin M Ertekin. «Derivation of machine tool error models and error compensation procedure for three axes vertical machining center using rigid body kinematics». En: *International Journal of Machine Tools and Manufacture* 40.8 (2000), págs. 1199-1213.
- [2] Shaowei Zhu et al. «Integrated geometric error modeling, identification and compensation of CNC machine tools». En: *International journal of machine tools and manufacture* 52.1 (2012), págs. 24-29.
- [3] Sitong Xiang y Yusuf Altintas. «Modeling and compensation of volumetric errors for five-axis machine tools». En: *International Journal of Machine Tools and Manufacture* 101 (2016), págs. 65-78.
- [4] Vladimir M Kvrđic et al. «Equivalent geometric errors of rotary axes and novel algorithm for geometric errors compensation in a nonorthogonal five-axis machine tool». En: *CIRP Journal of Manufacturing Science and Technology* 37 (2022), págs. 477-488.
- [5] Erik LJ Bohez. «Five-axis milling machine tool kinematic chain design and analysis». En: *International Journal of Machine Tools and Manufacture* 42.4 (2002), págs. 505-520.
- [6] Burak Sencer, Yusuf Altintas y Elizabeth Croft. «Modeling and control of contouring errors for five-axis machine tools—part I: modeling». En: (2009).
- [7] Stephen Mann et al. «Machine models and tool motions for simulating five-axis machining». En: *Computer-Aided Design* 42.3 (2010), págs. 231-237.
- [8] Hyoun-Chul Jung et al. «Development of practical postprocessor for 5-axis machine tool with non-orthogonal rotary axes». En: *Journal of Central South University of Technology* 18 (2011), págs. 159-164.
- [9] Vít Zelený, Ivana Linkeová y Pavel Skalník. «Calibration of freeform standard». En: (2016).
- [10] YH Jung et al. «NC post-processor for 5-axis milling machine of table-rotating/tilting type». En: *Journal of Materials Processing Technology* 130 (2002), págs. 641-646.
- [11] Chu A My y Erik LJ Bohez. «A novel differential kinematics model to compare the kinematic performances of 5-axis CNC machines». En: *International Journal of Mechanical Sciences* 163 (2019), pág. 105117.
- [12] Yaman Boz e Ismail Lazoglu. «A postprocessor for table-tilting type five-axis machine tool based on generalized kinematics with variable feedrate implementation». en. En: *Int. J. Adv. Manuf. Technol.* 66.9-12 (jun. de 2013), págs. 1285-1293.
- [13] Yaman Boz e Ismail Lazoglu. «A postprocessor for table-tilting type five-axis machine tool based on generalized kinematics with variable feedrate implementation». En: *The International Journal of Advanced Manufacturing Technology* 66 (2013), págs. 1285-1293.
- [14] Chu A My et al. «Transformation of CAM data for 5-axis CNC machine spinner U5–620». En: *Int. J. Mech. Eng. Robot. Res* 9.2 (2020), págs. 233-237.
- [15] Mihir Adivarekar y Frank Liou. «Developing a General Postprocessor for Multi-Axis CNC Milling Centers». En: *Computer-Aided Design & Applications* (2012), págs. 57-68.
- [16] Hong Sheng Zhou Yang, Jian Gang Li y Ze Xiang Li. «Kinematics analysis of non-orthogonal 5-axis machine tools». En: *Advanced Materials Research* 317 (2011), págs. 1826-1831.
- [17] Hélène Chanal, E Duc y A Chevalier. «Studying the influence of the machining process on the geometrical defects of the standardized S-shape test part». En: *Precision Engineering* 75 (2022), págs. 193-209.
- [18] Nuodi Huang et al. «Integrated post-processor for 5-axis machine tools with geometric errors compensation». En: *International Journal of Machine Tools and Manufacture* 94 (2015), págs. 65-73.
- [19] Xiaogeng Jiang. «Characterising geometric errors in rotary axes of 5-axis machine tools». Tesis doct. University of Birmingham, 2015.
- [20] Luis Uriarte et al. «Machine tools for large parts». En: *CIRP annals* 62.2 (2013), págs. 731-750.
- [21] Kwang-Il Lee y Seung-Han Yang. «Measurement and verification of position-independent geometric errors of a five-axis machine tool using a double ball-bar». En: *International Journal of Machine Tools and Manufacture* 70 (2013), págs. 45-52.
- [22] M Sharif Uddin et al. «Prediction and compensation of machining geometric errors of five-axis machining centers with kinematic errors». En: *Precision engineering* 33.2 (2009), págs. 194-201.
- [23] James Kennedy y Russell Eberhart. «Particle swarm optimization». En: *Proceedings of ICNN'95-international conference on neural networks*. Vol. 4. IEEE. 1995, págs. 1942-1948.

- [24] Federico Marini y Beata Walczak. «Particle swarm optimization (PSO). A tutorial». En: *Chemometrics and Intelligent Laboratory Systems* 149 (2015), págs. 153-165.
- [25] Anthony Carlisle, Gerry Dozier et al. «An off-the-shelf PSO». En: *Proceedings of the workshop on particle swarm optimization*. Vol. 1. Indianapolis, USA. 2001, págs. 1-6.
- [26] Ali B Hashemi y Mohammad Reza Meybodi. «Cellular PSO: A PSO for dynamic environments». En: *Advances in Computation and Intelligence: 4th International Symposium, ISICA 2009 Huangshi, China, October 23-25, 2009 Proceedings 4*. Springer. 2009, págs. 422-433.
- [27] Abhishek Banerjee et al. «Impacts of metaheuristic and swarm intelligence approach in optimization». En: *Cognitive Big Data Intelligence with a Metaheuristic Approach*. Elsevier, 2022, págs. 71-99.
- [28] E Gilbert, D Johnson y S Keerthi. «Fast collision detection using the gjk algorithm». En: *ACM SIGGRAPH Computer Graphics*. Vol. 22. 4. ACM. 1988, págs. 15-22.
- [29] Christer Ericson. «Real-time collision detection». En: *CRC press* 1.1 (2004), págs. 1-5.
- [30] Takahiro Harada, Yuji Kajiyama y Koji Koyamada. «Dynamic bounding volume trees for collision detection». En: *Proceedings of the 2005 ACM SIGGRAPH/Eurographics symposium on Computer animation*. ACM. 2005, págs. 131-140.