UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA

Generación de maniobras suaves en el espacio 3D.

PhD Dissertation

Autor Gloria Vanegas Zabala.

Director Leopoldo Armesto Ángel
Vicent Girbés Juan

Instituto de Diseño y Fabricación
Departamento de Ingeniería de Sistemas y Automática
Universitat Politècnica de València

5 de diciembre de 2023

Este trabajo ha sido parcialmente financiado por el Gobierno Español a través del Ministerio de Economía y Competitividad bajo el proyecto de Investigación $DPI2016-81002-R$, por la Administración local de la Generalitat Valenciana a través de los proyectos $GV/2017/029$, $GV/2021/074$ y por el seguimiento postdoctoral $APOSTD/2017/055$, además, las subvenciones $PID2020-116585GB-I00, PID2020-118071GB-I00$ y $PID2020-113785RB-100$ financiadas por $MCIN/AEI/10{,}13039/501100011033$.

Agradecimientos

Deseo extender mis más sincera muestra de gratitud y estima hacia mis tutores, Leopoldo Armesto y Vicent Girbés, por haberme dado la oportunidad y guiarme en la realización de mi tesis doctoral, sin ellos, este sueño no se hubiera logrado realizar.

Quiero dedicar este trabajo a toda mi familia, en especial a mis padres que con su apoyo, paciencia y cariño en todo momento me han ayudado a crecer como persona y a culminar las metas que me he planteado. También a mis hermanos que me han motivado a ser una profesional y no dejarme vencer, así como a mis sobrinos, por todo el amor que me han brindado. Y por último y no menos importante, quiero agradecer a Franklin, que a pesar de las dificultades que la vida nos ha presentado, siempre estuvo allí apoyándome y creyendo en mí.

Sinceramente
Gloria Isabel Vanegas Zabala

Resumen

El desarrollo tecnológico en la creación de trayectorias que permitan navegación libre de colisiones de Vehículos Autónomos (AVs) ha sido un objetivo constante de estudio debido a su fuerte interés científico y tecnológico en las últimas tres décadas. Las diferentes clases de AVs, ya sean Vehículos Aéreos no Tripulados (UAVs), Vehículos Terrestres no Tripulados (UGVs) o Vehículos Submarinos no Tripulados (UUVs), fomentan el desarrollo e implementación de trayectorias en el espacio tridimensional (3D). Un grupo especial de tecnología UAV está caracterizado por su ala fija, lo cual destaca características particulares en los AVs, debido a las restricciones no-holonómicas (un sistema que se describe mediante un conjunto de parámetros sujetos a restricciones diferenciales que no permiten que un vehículo se mueva de forma instantánea en cualquier dirección). En este sentido, las trayectorias navegables para estos UAVs no deben ser construidas como un conjunto de líneas rectas y círculos como en la gran mayoría de planificadores basados en primitivas, ya que no se garantiza una continuidad en su curvatura. Por lo tanto, las trayectorias construidas para esta rama tecnológica deben ser resueltas considerando las diferentes restricciones de maniobrabilidad del UAV, además de criterios de continuidad de curvas (el problema de continuidad se refiere principalmente a la continuidad geométrica, en términos de continuidad tangencial o de curvatura), suavidad en las curvas (una curva es suave si sus derivadas son continuas en el intervalo definido) y la seguridad en el vuelo (el control de seguridad garantiza que una trayectoria suave esté suficientemente lejos de los obstáculos). Finalmente, la cinemática del movimiento de los vehículos es otro factor que debe ser considerado mientras se suavizan las trayectorias.

El presente trabajo está enfocado en la creación de trayectorias navegables en el espacio 3D, para UAVs de características no-holonómicas. La principal dificultad al solventar este problema se debe a la movilidad de esta clase de UAVs, pues se ven obligados a avanzar sin la posibilidad de detenerse a través de trayectorias 3D, realizando curvas con curvaturas limitadas (una máxima capacidad de giro a una velocidad definida). En consecuencia, se han desarrollado las herramientas necesarias para proporcionar una completa caracterización de trayectorias óptimas (con un radio de giro limitado) para UAVs que se mueven en el espacio 3D a una velocidad constante.

Esta tesis se centra en la generación de caminos con trayectorias navegables en el espacio Euclídeo 3D, que contenga curvas con curvatura continua, considerando de esta manera las restricciones cinemáticas de los UAVs. Por tal motivo el objetivo principal es el desarrollo de la matemática necesaria para definir curvas clotoides en el espacio tridimensional, de modo que puedan ser utilizadas como primitivas en la generación de trayectorias. Finalmente, culminado el desarrollo de esta herramienta básica, y en función de los obstáculos del entorno, se puede completar una planificación y replanificación activa de movimientos.

Para complementar la investigación, la verificación de las herramientas de planificación de trayectorias y del sistema, se han realizado simulaciones con la ayuda del entorno de desarrollo integrado (IDE) Matlab. De la misma forma, se ha preparado una plataforma de simulación de vuelo, tomando las virtudes del simulador de vuelo FlightGear 2018, y el modelo dinámico del avión de ala fija con restricciones no-holonómicas (*Kadett 2400*). En cuanto a la generación de trayectorias 3D, se han desarrollado simulaciones off-line, donde las acciones de control que debe ejecutar el avión para que siga la trayectoria calculada son definidas por: acceleración, brusquedad de curvatura y brusquedad de torsión.

Por último, el enfoque de revisión bibliográfica presente en este documento se ha centrado en trabajos realizados que buscan cumplir con las tareas de planificación de trayectorias, planificación de movimiento y construcción de curvas suaves para AVs. Se ha tenido un especial cuidado en las metodologías usadas, la variedad de técnicas, además de las ventajas y desventajas presentadas a lo largo de la revisión literaria.

Abstract

The technological development in the creation of trajectories that allow collision-free navigation of Autonomous Vehicles (AVs) has been a continuous target of study due to its strong scientific and technological interest in the last three decades. Different classes of AVs, whether, Unmanned Aerial Vehicles (UAVs), Unmanned Ground Vehicles (UGVs) or Unmanned Underwater Vehicles (UUVs), encourage the development and implementation of paths in three-dimensional (3D) space. A special group of UAV technology is characterized by its fixed wing, which emphasizes particular characteristics in UAVs, due to non-holonomic constraints (a system that is described by a set of parameters subject to differential constraints that do not allow a vehicle to move instantaneously in any direction). In this sense, navigable paths for these UAVs should not be built as a set of straight lines and circles as in the vast majority of primitive-based planners, since no continuity in their curvature is guaranteed. Therefore, the paths built for this technology branch must be solved considering the different maneuverability constraints of the UAV, in addition to curve continuity criteria (the continuity problem refers mainly to geometric continuity, in terms of tangential or curvature continuity), curve smoothness (a curve is smooth if its derivatives are continuous in the defined interval) and flight safety (safety control ensures that a smooth path is sufficiently far away from obstacles). Finally, the kinematics of vehicle motion is another factor to be considered while smoothing paths.

This thesis work is focused on the creation of navigable paths in 3D space for UAVs with non-holonomic characteristics. The main difficulty in solving this problem is due to the mobility of this kind of UAVs, since they are forced to

move without the possibility of stopping through 3D paths, performing curves with limited curvatures (a maximum turning capacity at a defined speed). Consequently, the needed tools have been developed to provide a complete characterization of optimal paths (with a limited turning radius) for UAVs moving in the 3D plane at a constant velocity.

This thesis focuses on the generation of paths with navigable trajectories in 3D Euclidean space, containing curves with continuous curvature, thus considering the kinematic constraints of UAVs. Therefore, the main aim is the development of the necessary mathematics to define clothoid curves in the three-dimensional space, so that they can be used as primitives in the generation of paths. Finally, once the development of this basic tool has been completed, and depending on the obstacles in the environment, an active planning and replanning of movements can be completed.

To complement the research, the verification of the path planning tools and the system, simulations have been performed with the help of the integrated development environment (IDE) Matlab. In the same way, a flight simulation platform has been prepared, taking the virtues of the FlightGear 2018 flight simulator, and the dynamic model of the fixed-wing aircraft with non-holonomic constraints (*Kadett 2400*). Regarding the generation of 3D paths, off-line simulations have been developed, where the control actions to be executed by the aircraft to follow the calculated path are defined by: acceleration, curvature sharpness and torsion sharpness.

Finally, the literature review approach presented in this document has focused on works that address the tasks of path planning, motion planning and construction of smooth curves for AVs. Special care has been taken in the methodologies used, the variety of techniques, in addition to the advantages and disadvantages presented throughout the literature review.

Resum

El desenvolupament tecnològic en la creació de trajectòries que permeten navegació lliure de col·lisions de Vehicles Autònoms (AVs) ha estat un objectiu constant d'estudi a causa del seu fort interés científic i tecnològic en les últimes tres dècades. Les diferents classes d'AVs, ja siguen Vehicles Aeris no Tripulats (UAVs), Vehicles Terrestres no Tripulats (UGVs) o Vehicles Submarins no Tripulats (UUVs), fomenten el desenvolupament i la implementació de trajectòries a l'espai tridimensional (3D). Un grup especial de tecnologia UAV està caracteritzat per la seua ala fixa, cosa que destaca característiques particulars en els AVs, a causa de les restriccions no-holonòmiques (un sistema que es descriu mitjançant un conjunt de paràmetres subjectes a restriccions diferencials que no permeten que un vehicle es menege de forma instantània en qualsevol direcció). En aquest sentit, les trajectòries navegables per a aquests UAVs no han de ser construïdes com un conjunt de línies rectes i cercles com a la gran majoria de planificadors basats en primitives, ja que no es garanteix una continuïtat en la seua curvatura. Per tant, les trajectòries construïdes per a aquesta branca tecnològica han de ser resoltes considerant les diferents restriccions de maniobrabilitat de l'UAV, a més de criteris de continuïtat de corbes (el problema de continuïtat es refereix principalment a la continuïtat geomètrica, en termes de continuïtat tangencial o de curvatura), suavitat a les corbes (una corba és suau si les seves derivades són contínues en l'interval definit) i la seguretat en el vol (el control de seguretat garanteix que una trajectòria suau estiga prou lluny dels obstacles). Finalment, la cinemàtica del moviment dels vehicles és un altre factor que cal considerar mentre se suavitzen les trajectòries.

Aquest treball està enfocat a la creació de trajectòries navegables a l'espai 3D, per a UAVs de característiques no-holonòmiques. La principal dificultat en solucionar aquest problema es deu a la mobilitat d'aquesta classe de UAVs, ja que es veuen obligats a avançar sense la possibilitat d'aturar-se a través de trajectòries 3D, fent corbes amb curvatures limitades (una màxima capacitat de gir a una velocitat definida). En conseqüència, s'han desenvolupat les ferramentes necessàries per proporcionar una completa caracterització de trajectòries òptimes (amb un radi de gir limitat) per a UAVs que es mouen al pla 3D a una velocitat constant.

Aquesta tesi se centra en la generació de camins amb trajectòries navegables a l'espai Euclidià 3D, que continguen corbes amb curvatura contínua, considerant així les restriccions cinemàtiques dels UAVs. Per aquest motiu, l'objectiu principal és el desenvolupament de la matemàtica necessària per definir corbes clotoides a l'espai tridimensional, de manera que puguen ser utilitzades com a primitives en la generació de trajectòries. Finalment, culminat el desenvolupament d'aquesta ferramenta bàsica, i en funció dels obstacles de l'entorn, es pot completar una planificació i una replanificació activa de moviments.

Per complementar la investigació, la verificació de les ferramentes de planificació de trajectòries i del sistema, s'han fet simulacions amb l'ajuda de l'entorn de desenvolupament integrat (IDE) Matlab. De la mateixa manera, s'ha preparat una plataforma de simulació de vol, prenent les virtuts del simulador de vol FlightGear 2018 i el model dinàmic de l'avió d'ala fixa amb restriccions no-holonòmiques (Kadett 2400). Pel que fa a la generació de trajectòries 3D, s'han desenvolupat simulacions off-line, on les accions de control que ha d'executar l'avió perquè seguisca la trajectòria calculada són definides per: acceleració, brusquedat de curvatura i brusquedat de torsió.

Finalment, l'enfocament de revisió bibliogràfica present en aquest document s'ha centrat en treballs realitzats que busquen complir les tasques de planificació de trajectòria, planificació de moviment i construcció de corbes suaus per a AVs. S'ha tingut una cura especial en les metodologies usades, la varietat de tècniques, a més dels avantatges i desavantatges presentats al llarg de la revisió literària.

Índice general

Capítulo 1

Introducción

Resumen: *Este primer capítulo tiene como objetivo realizar un resumen introductorio acerca de las principales ideas de la tesis, la cual ha sido desarrollada dentro del marco de la planificación y navegación en vehículos autónomos AVs. En este sentido, con el propósito de hacer un acercamiento hacia la motivación del tema de tesis, se ha realizado un breve resumen sobre la relevancia en el estudio actual de la tecnología de los AVs y su impacto económico. En la siguiente instancia se describen los objetivos alcanzados durante el transcurso del trabajo de tesis, para finalmente concluir con la descripción de la estructura general del documento de tesis.*

1.1 Motivación

Los Vehículos Autónomos (AV, del acrónimo *Autonomous Vehicles*), por décadas han tenido como objetivo el reemplazar a los pilotos humanos en diversas misiones, facilitando de esta manera el desenvolvimiento cotidiano en diferentes tareas. Diversos autores presentan un conjunto importante de desarrollo en las diferentes tecnologías de los AVs, ya sean Vehículos Terrestres Autónomos (UGV, del acrónimo *Unmanned Ground Vehicle*) como se destaca en Wang y col. 2009. De forma similar los trabajos realizados por Gafurov y Klochkov 2015; Wynn y col. 2014; Phillips y col. 2017 se enfocan en Vehículos Submarinos Autónomos (AUV, del acrónimo *Autonomous Underwater Vehicle*) y, finalmente, en vehículos Aéreos no Tripulados (UAV, del acrónimo *Unmanned Aerial Vehicle*) los avances son trascendentes como se destaca en Stöcker y col. 2017; Gupta, Ghonge y Jawandhiya 2013; Carrio y col. 2017. La constante evolución y desarrollo tecnológico continúa en busca de mejoras, por lo que el estudio de los AVs es cada vez más profundo, sin importar las soluciones que se van presentando sobre determinados problemas.

En este sentido, un objetivo de estudio dentro de la tecnología de los AVs, se enfoca en la reducción de accidentes ya sean terrestres, marinos o aéreos como se describe en Claesson y col. 2017; Giese, Carr y Chahl 2013; Asim, Ehsan y Rafique 2005. En consecuencia, un AV tiene que poseer la capacidad de escoger entre minimizar el daño propio o el de los pasajeros. Por lo tanto, en función de las tareas que han sido asignadas a determinado AV, para brindar apoyo en la toma de decisiones se debe implementar una base algorítmica apropiada sobre tal tecnología.

Por otra parte, Bonnefon, Shariff y Rahwan 2016 afirman que un AV no requiere ninguna intervención humana (o tal vez mínima) durante su misión, ya sea civil, según Zhou y Zang 2007; Maza. I Caballero. F y A 2011, o militar de acuerdo con Anderson y col. 2008; Pinkney, Hampel y DiPierro 1996. Por lo que, a partir del estudio realizado por Yang y col. 2017, se puede definir su autonomía como la capacidad de un sistema para sensar, transmitir información, planificar y tomar decisiones. Por ello, el objetivo de la autonomía es facultar al vehículo a obrar según algún criterio y con independencia. En específico en esta tesis, los criterios a cumplir se enfocan en: seguir, planificar y encontrar trayectorias desde un punto inicial hasta otro punto final denominado meta. Esto significa un reto desafiante dentro del campo de los AVs. Sin embargo, a diferencia de la planificación en 2D, según Kavraki y LaValle 2016; Ilari y Torras 1990, las dificultades aumentan exponencialmente al considerar las restricciones cinemáticas, tal como se describe en Cai, Zhang y Zheng 2017, y más aún sobre entornos espaciales 3D como se destaca en Carsten, Ferguson y Stentz 2006; Yang y col. 2016.

El presente trabajo de tesis se enfoca en el estudio, desarrollo y generación de trayectorias suaves continuas. Diversos autores han abordado esta problemática, como se describe en Yang y Sukkarieh 2008a; Ravankar y col. 2018. No obstante, esta tesis se basa en curvas clotoidales o espirales de Cornu, conceptos detallados por Boissonnat, Cerezo y Leblond 1994; Meek y Walton 2004a, quienes garantizan la aeronavegabilidad para UAVs que se desplazan en el espacio de trabajo Euclídeo 3D. Otros autores como Frego 2022; Blasi y col. 2023 se enfocan en el estudio y desarrollo de curvas clotoidales tridimensionales. Sin embargo, esta tesis busca alcanzar una adecuada transición con curvas clotoidales hasta construir nuevas trayectorias navegables por UAVs de ala fija, que poseen restricciones con características no-holonómicas como las descritas en Girbés, Armesto y Tornero 2014; Marzbani y col. 2015a. Por otra parte, en la literatura existe una gran variedad de trabajos, en los que se aborda el problema de generación de trayectorias suaves, como se describe en Ho y Liu 2009; Neto, Macharet y Campos 2013; Huh y Chang 2014; Fraichard y Scheuer 2004b. No obstante, la base metodológica adoptada para la resolución en planificación y navegación 3D sigue siendo objeto de estudio en la actualidad, como se destaca en Ravankar y col. 2018.

Finalmente, la gran demanda de estudio, desarrollo e investigación en el campo de los UAVs, se debe a su alto impacto económico. Por ejemplo, en los Estados Unidos de Norteamérica (EE.UU.), se hace una importante inversión económica cada año, superando los 13.6 mil millones de dólares, beneficio que se pronostica hasta el año 2025 y que, además, se prevé un incremento de $100,000$ puestos de trabajo con un futuro impacto económico de \$82 mil millones de dólares, como se menciona en Jenkins y Vasigh 2013.

1.2 Objetivos

El principal objetivo para el desarrollo de la tesis se ha centrado en el estudio y generación de la matemática necesaria para definir una nueva trayectoria en el espacio tridimensional, denominada Trayectoria 3D basada en Clotoide (Cb3D, del acrónimo *Clothoid based 3D curve*), de modo que pueda ser utilizada como primitiva en la generación de trayectorias. A continuación, se describen otros objetivos planteados y alcanzados durante el desarrollo de la tesis:

- Inferir y comparar el rendimiento de Cb3D con otras herramientas de planificación espacial (spline, espirales cúbicas, curvas Bézier, Rational Bézier, etc.).

- Aproximar la trayectoria Clotoide 3D (C3D, del acrónimo *Clothoid 3D*) mediante Rational Bézier, para lograr un rendimiento en tiempo real.

- Implementar plataformas de simulación de vuelo para la generación de trayectorias 3D, para Vehículos Aéreos no Tripulados (UAV) con restricciones no-holonómicas.

- Implementar metodologías de trayectorias suaves para el movimiento continuo de Vehículos Aéreos no Tripulados UAVs de ala fija, teniendo en cuenta limitaciones como la curvatura factible de vuelo y mínima distancia.

- Establecer una metodología de planificación de trayectoria suave basada en curvas clotoidales 3D con curvatura y torsión no nula.

- Extender el concepto de trayectoria de Doble Curvatura Continua (DCC, del acrónimo *Double Continuous Curvature*) del espacio 2D al 3D, tomando como punto de partida el concepto de la curva Cb3D.

- Implementar la trayectoria de Doble Curvatura Continua (DCC3D, del acrónimo *Double Continuous Curvature 3D Curve*) como una herramienta de planificación de trayectorias que permita alcanzar una configuración arbitraria en posición y orientación en el espacio Euclídeo 3D.

Dentro del desarrollo de la tesis, la etapa cero ubicada en el centro de la Figura 1.1, presenta el aporte más significativo, el cual propone y completa la matemática para la generación de Curvas 3D basadas en Clotoides (Cb3D). El análisis matemático ha sido usado como primitiva en la generación de trayectorias suaves. Las siguientes etapas son el complemento del desarrollo de la tesis, donde se han realizado diferentes estudios sobre los que se han implementado simulaciones, obteniendo resultados, y a través de ellos se han determinado diversas conclusiones.

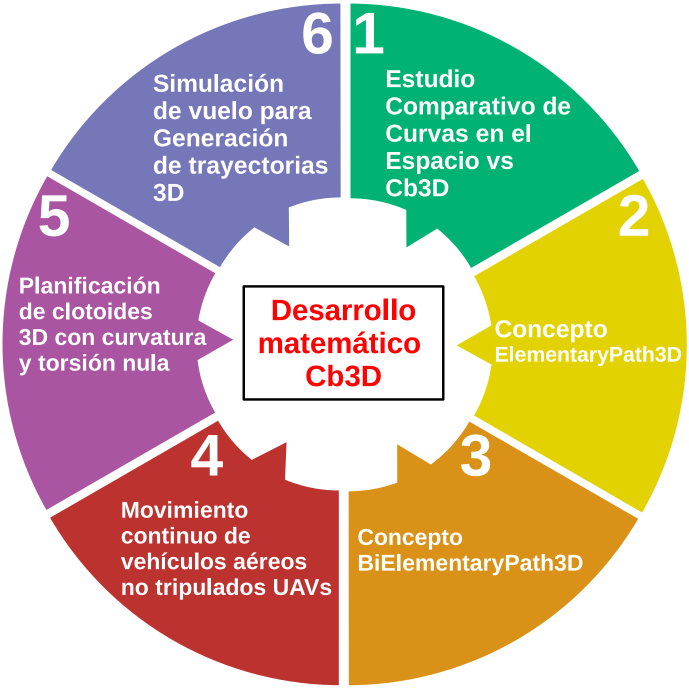

6
Simulación de vuelo para Generación de trayectorias 3D
1
Estudio Comparativo de Curvas en el Espacio vs Cb3D
2
Concepto ElementaryPath3D
3
Concepto BiElementaryPath3D
4
Movimiento continuo de vehículos aéreos no tripulados UAVs
5
Planificación de clotoides 3D con curvatura y torsión nula
Desarrollo matemático Cb3D

Figura 1.1: Fases de trabajo realizadas durante la tesis.

1.3 Estructura de la Tesis

El presente documento de tesis contiene un resumen de las 4 publicaciones realizadas durante su desarrollo, las cuales exponen los resultados alcanzados como parte de este estudio de doctorado.

A continuación, se presenta un resumen del material compilado en este documento, que se encuentra organizado en 2 grandes partes. En la primera parte se presenta un estudio del estado del arte, enfocado en la definición de curvas suaves en el espacio Euclídeo.

- En el capítulo 2 se describe la base matemática, además de diversas características de un conjunto de diferentes curvas en el espacio, realizando un enfoque especial sobre las curvas Clotoides 2D y Clotoides 3D.

En la segunda parte del documento de tesis se detallan las publicaciones resultantes del trabajo, resumido en 4 capítulos, en lo cuales se describen los diferentes conceptos aportados a lo largo de la investigación, siendo: Trayectoria 3D basada en Clotoide (Cb3D), Trayectoria Elemental 3D basada en Clotoide (ECb3D), Trayectoria 3D de Doble Curvatura Continua basada en Clotoide (DCC3D) y la aplicación de estos conceptos en la generación de trayectorias aeronavegables. A continuación, se realiza una descripción de las contribuciones publicadas en cada trabajo y cómo se relacionan entre ellas. Adicionalmente, se presentan las conclusiones finales y los posibles trabajos futuros en relación a esta tesis.

- El capítulo 3, busca realizar una primera aproximación hacia las trayectorias suaves, para lo cual se ha tomado como punto de partida la base metodológica de las curvas clotoidales 3D (C3D), aproximadas mediante curvas Bézier-Racional (curva muy estudiada dentro de la matemática Euclídea). El resultado es aprovechado como una curva de transición para un planificador de vuelo suave 3D. El objetivo del estudio ha sido enfocado en la definición de las características más relevantes, así como los alcances de la C3D.

- En el capítulo 4, se describe la solución analítica matemática de una nueva curva en el espacio, definida como Trayectoria 3D basada en Clotoides (Cb3D). Esta curva proporciona diferentes propiedades como escalabilidad, simetría, suavidad y monotonicidad. Dichas propiedades la hacen adecuada para resolver el problema de Planificación de trayectorias suaves en el espacio 3D.

- En el capítulo 5, se presenta una nueva trayectoria suave 3D denominada Trayectoria Elemental 3D basada en Clotoide (ECb3D), que usa como primitiva la trayectoria Cb3D, explota sus propiedades y permite hacer una concatenación sin pérdida de continuidad con valores acotados en curvatura y torsión.

- En el capítulo 6, se alcanza un objetivo adicional no incluido en los capítulos anteriores, puesto que las trayectorias suaves propuestas sólo llegan a una orientación objetivo en un espacio de trabajo 3D. En este capítulo se describe una nueva trayectoria suave que consigue llegar al mismo tiempo a una posición y orientación 3D. Esta tarea ha sido completada al definir la nueva trayectoria, denominada Trayectoria 3D de Doble Curvatura Continua basada en Clotoide (DCC3D).

- El capítulo 7 presenta las conclusiones finales del trabajo desarrollado a lo largo de la tesis, para así finalizar mencionando los posibles trabajos futuros sobre el estudio abordado.

Finalmente, se presenta una sección adicional donde se describen 2 anexos, necesarios para completar la descripción y entendimiento de la tesis.

- Con objetivo de aclarar los diferentes conceptos matemáticos propuestos durante el trabajo de investigación, en el Anexo A se describen propiedades, teoremas y demostraciones que complementan las contribuciones de la tesis.

- Dado que se han realizado simulaciones y experimentos de vuelo explorando las capacidades del modelo *UAV Kadett 2400*, el Anexo B inicia realizando un breve resumen de la tecnología del modelo UAV, hasta finalizar con la descripción cinemática y dinámica del UAV.

Publicaciones

El trabajo desarrollado a lo largo de esta tesis ha logrado varias publicaciones que se detallan a continuación:

Artículos de Revista

* Girbés, V., **Vanegas, G**., & Armesto, L. (2019). *Clothoid-Based Three-Dimensional Curve for Attitude Planning.* Journal of Guidance, Control, and Dynamics, 42(8), 1886-1898., JCR: Q1, Impact Factor: 2.486, SJR: Q1, Impact Factor: 1.34, DOI: https://doi.org/10.2514/1.G003551.

* Armesto, L., **Vanegas, G**., & Girbés-Juan, V. (2022). Elementary Clothoid-Based Three-Dimensional Curve for Unmanned Aerial Vehicles. Journal of Guidance, Control, and Dynamics, 45(12), 2421-2431. JCR: Q1, Impact Factor: 2.486, SJR: Q1, Impact Factor: 1.34, DOI: https://doi.org/10.2514/1.G006935.

* **Vanegas, G**., Armesto, L., Girbés-Juan, V., & Pérez, J. (2022). Smooth Three-Dimensional Route Planning for Fixed-Wing Unmanned Aerial Vehicles With Double Continuous Curvature. IEEE Access, 10, 94262-94272. JCR: Q1, Impact Factor: 3.476, SJR: Q2, Impact Factor: 0.93, DOI: https://10.1109/ACCESS.2022.3203069.

Artículo de Congreso

* **Vanegas, G**., Samaniego, F., Girbés, V., Armesto, L., & Garcia-Nieto, S. (2018, October). *Smooth 3D path planning for non-holonomic UAVs.* In 2018 7th International Conference on Systems and Control (ICSC) (pp. 1-6). IEEE.

Del mismo modo, durante el transcurso de la tesis, se ha asistido a un curso especializado:

* European Embedded Control Institute (EECI) - International Graduate School on Control. M12 - Nonlinear Observers: Applications to Aerial Robotic Systems. Genova - Italia. Abril 2019.

Capítulo 2

Curvas en el espacio y su aproximación a trayectorias suaves

Resumen: *Este capítulo realiza un breve estudio matemático y metodológico en la construcción de diversas curvas en el espacio. En especial, se ha enfocado en la definición de las curvas Clotoides, sus características y alcances. En específico, el objetivo es definir el punto de partida y las bases para el posterior desarrollo del trabajo de tesis. Finalmente, se describe de forma breve el acercamiento hacia la generación de trayectorias de curvatura continua, a partir de las curvas suaves.*

2.1 Introducción

Hace aproximadamente 3.000 años los griegos comenzaron a buscar explicaciones racionales a fenómenos naturales y sentaron las bases de la geometría y la aritmética. En ese ámbito de las matemáticas destacaron diversos personajes históricos como Pitágoras (c. 569 a. C. - c. 475 a. C.), Euclides (ca. 325 a. C. - ca. 265 a. C.) o Diofanto, Galois(1811 - 1832), matemáticos como Fermat(1601 - 1665), Euler (1707 - 1783), Lagrange (1736 - 1813) o Gauss (1777 - 1855), sin dejar atrás a Téano, la primera mujer matemática de la historia (y esposa de Pitágoras).

En específico, el estudio matemático de las curvas se remonta a los trabajos de la antigua Grecia, dónde se resolvieron cuestiones de lo que en la actualidad se conocen como curvas algebraicas, siendo ejemplo de éstas, las curvas cónicas, la curva cisoide de Diocles, la curva concoide de Nicomedes, entre otras; curvas estudiadas de forma amplia por Cordero, Fernández y Gray 1995; Gómez y Torres 2006.

Alexander y Koeberlein 2014 describen la curva inicialmente estudiada en geometría elemental. De forma similar, Kühnel 2015, detalla de forma rigurosa la curva en geometría diferencial, la que se define como una línea continua de una dimensión y que varía de dirección paulatinamente. Ejemplos sencillos de curvas cerradas simples son la elipse o la circunferencia (el óvalo o el cicloide). Mientras que ejemplos de curvas abiertas, son la parábola, la hipérbola, la catenaria y una infinidad de curvas que son estudiadas en la geometría analítica plana por varios autores, entre ellos Aguirre 2007.

Diversas curvas han sido utilizadas como base para la generación de trayectorias suaves y navegables por AVs de características no-holonómicas. En este sentido, a continuación, se realiza un breve resumen de trabajos que utilizan curvas suaves como punto de partida, tales como: curvas Bézier, curvas Clotoides, etc.

El trabajo realizado por Choi, Curry y Elkaim 2008, propone dos algoritmos de planificación de trayectorias que utilizan las curvas Bézier, modificando la posición adecuada de los puntos de control para un robot móvil. En Kawabata y col. 2015, se propone una generación de trayectorias suaves para un vehículo automatizado, por medio de la curva Bézier en 2D y 3D. Los trabajos realizados por Choi, Curry y Elkaim 2008; Rastelli, Lattarulo y Nashashibi 2014, discuten la ubicación optimizada de los puntos de control para la posterior generación de curvas Bézier. Los autores, Walton, Meek y Ali 2003, utilizan las curvas Bézier como curvas de transición para unir segmentos rectos con segmentos curvos.

Mientras que los autores Han y col. 2010; Pérez y col. 2013; González y col. 2014, discuten la generación de trayectorias para los vehículos en entornos urbanos. Finalmente, algunos investigadores han utilizado las curvas Bézier para el estacionamiento automático de vehículos como es detallado por Liang, Zheng y Li 2012.

Es importante destacar que existen diversas curvas geométricas que generan trayectorias con curvatura continua. En particular, las curvas clotoides poseen propiedades geométricas particulares, incluyendo su cercana relación entre los fenómenos físicos (aceleración, normal y jerks) con el parámetro de escalado clotoidal, lo que implica beneficios al trabajar con esta curva, propiedad resaltada en Montes, Mora y Tornero 2007; Montés y col. 2008, donde se plantea la generación de trayectorias clotoidales, demostrando su modularidad y extensibilidad.

Como se puede apreciar, existe una amplia literatura con respecto a trabajos de trayectorias suaves con base en curvas geométricas. No obstante, antes de asumir una metodología geométrica de curvas, a continuación, se realiza la descripción formal de curvas en el espacio y de dos tipos de curvas que son utilizadas y que brindan buenos resultados en la generación de trayectorias suaves.

2.2 Curvas en el espacio

Con frecuencia se considera una curva en el plano como una línea trazada sobre un papel, tal como puede ser una línea recta, una circunferencia o una curva parabólica. Se puede realizar una definición analítica de una curva en el plano, al definir los puntos que forman la curva. Para lo cual se pueden utilizar las coordenadas cartesianas de los puntos $P(x, y)$ de la curva expresados como una función de $x[y = F(x)]$, ó x como una función de $y[x = G(y)]$, o simplemente dar una relación entre x e y que defina de forma implícita una variable en términos de la otra $[H(x, y) = 0]$. Por otra parte, existen curvas que se describen de una mejor forma cuando las coordenadas x e y están dadas en términos de una tercera variable t (llamada parámetro) $[x = f(t)$ e $y = g(t)]$. Finalmente, con base en el trabajo expuesto por Stewart y col. 2012; Aguirre 2007, se puede indicar cada punto de una curva, haciendo uso de la asociación de P con el punto final del vector $\overrightarrow{\mathbf{r}} = \overrightarrow{OP}$ ubicado en su posición canónica.

Con el objetivo de continuar con el desarrollo de este capítulo, se asume que toda curva en el espacio $\mathbb{R}^n$ se puede considerar como la imagen de una función vectorial

$$\mathbf{r} : [a, b] \to \mathbb{R}^n, \mathbf{r}(t) = (x_1(t), \ldots, x_n(t)), \tag{2.1}$$

que recibe el nombre de parametrización de la curva. Los puntos $\mathbf{r}(a)$ y $\mathbf{r}(b)$ son los extremos inicial y final de la curva. En el caso de que $\mathbf{r}(a) = \mathbf{r}(b)$, se dice que la curva es cerrada.

Por otra parte, dos funciones $\boldsymbol{\varphi} : [a, b] \to \mathbb{R}^n$ y $\boldsymbol{\psi} : [\alpha, \beta] \to \mathbb{R}^n$ son equivalentes si existe una función $\boldsymbol{\lambda} : [a, b] \to [\alpha, \beta]$ biyectiva y continua, tal que $\boldsymbol{\psi} = \boldsymbol{\varphi}$. Entonces, la función $\boldsymbol{\lambda}$ recibe el nombre de *cambio de parámetro*. Por lo tanto, dos funciones equivalentes representan parametrizaciones distintas de la misma curva y la función $\boldsymbol{\lambda}$ representa un cambio en la rapidez del movimiento.

- Si $\boldsymbol{\lambda}$ es creciente, se dice que las parametrizaciones $\boldsymbol{\varphi}$ y $\boldsymbol{\psi}$ conservan la orientación de la curva.
- Si $\boldsymbol{\lambda}$ es decreciente, las parametrizaciones $\boldsymbol{\varphi}$ y $\boldsymbol{\psi}$ invierten la orientación de la curva.

La curva C es simple cuando $\boldsymbol{\varphi}$ es inyectiva. Entonces, la curva definida por la función $\boldsymbol{\varphi}(t) = (\cos t, \operatorname{sen} t), t \in [0, 2\pi]$, es simple solo si se define $\boldsymbol{\varphi_2}(t) = (\cos 2t, \operatorname{sen} 2t), t \in [0, 2\pi]$, entonces, la curva resultante no es simple.

Dada una curva C con vector de posición $\mathbf{r}(t)$, se define la longitud de arco de curva entre los puntos $\mathbf{r}(a)$ y $\mathbf{r}(b)$, a las longitudes de las poligonales inscritas a la curva entre tales puntos. Lo cual indica que la curva es rectificable, por tanto, se asume que:

Dada una función $\boldsymbol{\varphi} : [a, b] \to \mathbb{R}^n$, se llama variación de $\boldsymbol{\varphi}$ con respecto a una partición $p = \{t_0, t_1, \ldots, t_m\}$ de $[a, b]$ a

$$\mathbf{V}(\boldsymbol{\varphi}, p) = \sum_{i=1}^{m} |\boldsymbol{\varphi}(t_i) - \boldsymbol{\varphi}(t_i - 1)| \tag{2.2}$$

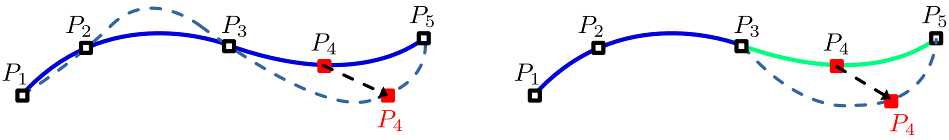
Se llama variación total de $\boldsymbol{\varphi}$ en $[a, b]$ a

$$\mathbf{V}(\varphi) = \sup_p V(\boldsymbol{\varphi}, p) \tag{2.3}$$

Entonces, la función $\boldsymbol{\varphi}$ es de variación acotada cuando $\mathbf{V}(\boldsymbol{\varphi}) < \infty$.

2.2.1 Descripciones de continuidad de una curva

Una curva puede descomponerse en cadenas de curvas, definidas por medio de un conjunto de puntos de control[1] P_i, y denominadas como curvas globales, como se puede ver en la Figura 2.1. La variación de las propiedades geométricas de los P_i a lo largo de la curva o por partes, no presenta discontinuidades. Es decir, la curva se puede descomponer en un conjunto de segmentos rectilíneos o curvilíneos encadenados. Por tanto, los cambios en los P_i afectan de forma local a las partes que están desacopladas.



(a) Un solo segmento de curva que atraviesa 5 puntos de control P_1, P_2, P_3, P_4 y P_5.

(b) Dos segmentos de curva, el primer segmento (línea azul continua) definido por 3 puntos de control P_1, P_2 y P_3, mientras que el segundo segmento (línea verde contínua) está definido por los puntos de control P_3, P_4 y P_5.

Figura 2.1: Tipos de cadenas de curvas, atravesando un conjunto de puntos de control.

La Figura 2.1(a) muestra una curva global (línea azul continua), donde la acción de movimiento sobre el punto de control P_4 (cuadro rojo) implica la deformación de toda la curva, causando una nueva curva (línea azul discontinua). Mientras que en la Figura 2.1(b) el desplazamiento del punto de control P_4 (cuadro rojo) solo deforma el segmento correspondiente de curva (línea verde), ocasionando un nuevo segmento de curva (línea azul discontinua).

El análisis de continuidad de curvas implica, un estudio profundo de continuidad (también conocido como el encadenamiento de las curvas). De esta forma, en la práctica las condiciones de encadenamiento de las curvas se define como el grado de continuidad.

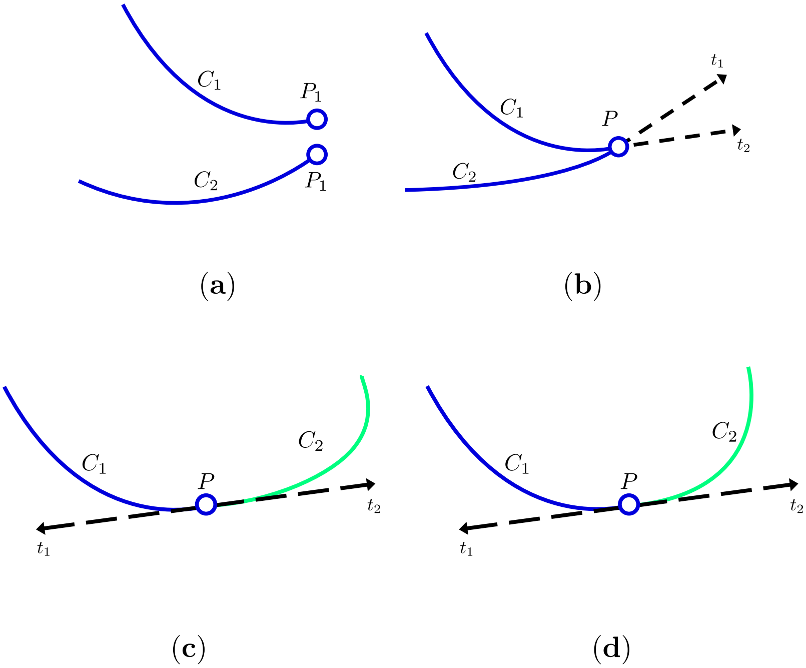
Entonces, a partir de 2 curvas, se consideran 4 niveles o grados crecientes de continuidad, siendo:

[1] Un conjunto de puntos espaciales p_k, a los que se aproxima a una curva c, siendo: $p_k = (x_k, y_k)$ si $C \in \mathbb{R}^2$ o $p_k = (x_k, y_k, z_k)$ si $C \in \mathbb{R}^3$.

- **Nula** C^0
  - Si 2 curvas no tienen un punto en común (ver Figura 2.2(a)).
- **Posicional** C^1
  - Si 2 curvas tienen un punto en común (ver Figura 2.2(b)).
- **Tangencial** C^2
  - Siempre que exista continuidad posicional, y además, la dirección de la tangente es la misma para el punto común, tanto si se le considera como último punto de la primera curva, como si se le considera primer punto de la siguiente curva (ver Figura 2.2(c)).
- **De curvatura** C^3
  - Siempre que exista continuidad tangencial, y además el radio de curvatura sea el mismo para un punto común, tanto si se le considera como último punto de la primera curva, como si se le considera primer punto de la segunda curva (ver Figura 2.2(d)).

Existen otras clasificaciones más complejas, que distinguen entre continuidad geométrica y paramétrica, siendo:

- **Posición** (G^0).
  - Sólo calcula la posición, es decir, si los puntos finales de cada curva se encuentran en la misma posición en el espacio, las curvas tienen continuidad de posición G^0 en los puntos finales. En consecuencia, las dos curvas se tocan en sus puntos finales.
- **Tangencia** (G^1).
  - Una curva tiene continuidad geométrica G^1, cuando las primeras derivadas (tangentes) de las dos curvas son proporcionales en el punto de unión. La continuidad G^1, calcula la posición y la dirección de la curva en los puntos finales. En otras palabras, las dos curvas no solo se juntan, sino que van en la misma dirección en el punto donde se juntan. La dirección se determina por el primer y el segundo punto en cada curva, si estos dos puntos se encuentran en una línea, las dos



(**a**) (**b**)

(**c**) (**d**)

Figura 2.2: Grados crecientes de continuidad de Curvas.

curvas son tangentes G^1 en los puntos finales. Es decir, la primera derivada de las dos curvas es igual en el punto en el que se juntan.

- **Curvatura** (G^2).

  - La continuidad de curvatura (continuidad G^2) entre dos curvas calcula la posición, la dirección y el radio de curvatura en los puntos finales. Si el radio de curvatura es el mismo en el punto final común, las curvas tendrán continuidad de curvatura G^2. En otras palabras, las curvas no solo van en la misma dirección cuando coinciden, sino que también tienen el mismo radio en ese punto. Tanto la primera como la segunda derivada de las ecuaciones son iguales en ese punto.

- G^3
  - La continuidad G^3 añade la aceleración plana. Las curvas que son continuas G^3 tienen la misma dirección y el mismo radio. Las curvas con continuidad G^3 poseen terceras derivadas iguales.
- G^4
  - La continuidad G^4 poseen los mismos requisitos que las curvas G^3, pero su aceleración de curvatura es igual en tres dimensiones.

2.2.2 Propiedades geométricas

Las propiedades geométricas de una curva pueden describirse mediante las propiedades de la función que la describe. A continuación, se enuncian propiedades y operaciones básicas con funciones vectoriales, que se aplican al estudio de las curvas en el espacio.

Partiendo del hecho de que toda función vectorial $f : \mathbb{R} \to \mathbb{R}^n$ se puede descomponer en n funciones escalares, entonces, definen un conjunto de operaciones algebraicas con dichas funciones de manera análoga a las correspondientes con funciones escalares. Por lo tanto, dadas $f, g : \mathbb{R} \to \mathbb{R}^n$, y $u : \mathbb{R} \to \mathbb{R}$ se definen las siguientes operaciones:

- Suma: $f + g : \mathbb{R} \to \mathbb{R}^n$, como $(f + g)(t) = f(t) + g(t)$.
- Multiplicación por una función escalar: $uf : \mathbb{R} \to \mathbb{R}^n$, como $(uf)(t) = u(t) \cdot f(t)$.
- Producto escalar: $f \cdot g : \mathbb{R} \to \mathbb{R}^n$, como $(f \cdot g)(t) = f(t) \cdot g(t)$.
- Producto vectorial (para $n = 3$): $f \times g : \mathbb{R} \to \mathbb{R}^n$, como $(f \times g)(t) = f(t) \times g(t)$.
- Composición: $f\ o\ u : \mathbb{R} \to \mathbb{R}^n$, como $(f\ o\ u)(t) = f(u(t))$.

2.2.3 Límites y continuidad de funciones vectoriales

Si $f(f_1, \ldots, f_n) : \mathbb{R} \to \mathbb{R}^n$ es una función vectorial que se define como

$$\lim_{t\to t_0} f(t) = \left(\lim_{t\to t_0} f_1(t), \ldots, \lim_{t\to t_0} f_n(t) \right) \tag{2.4}$$

por tanto, una función vectorial es continua en t_0 si $\lim_{t\to t_0} f(t) = f(t_0)$.

2.2.4 Derivación de funciones vectoriales

Una función vectorial $f(t) : \mathbb{R} \to \mathbb{R}^n$ es derivable en t_0 si existe

$$f'(t_0) = \lim_{t\to t_0} \frac{f(t_0 + h) - f(t_0)}{h} \tag{2.5}$$

si f es variable en t, entonces

$$\frac{df}{dt} = f'(t) = (f_1'(t), \ldots, f_n'(t)) \tag{2.6}$$

En este sentido, dada una curva $\mathbf{r}(t) = (x_1(t), \ldots, x_n(t))$, el valor de $\mathbf{r}'(t)$[2] recibe el nombre de *vector tangente* a la curva. Si $\mathbf{r}'(t) = 0$, no se define el vector tangente[3]. Entonces, se denota como $\mathbf{T}(t) = \frac{\mathbf{r}'(t)}{|\mathbf{r}'(t)|}$ al *vector tangente unitario.*

De la misma forma, se define como *recta tangente* a la curva $\mathbf{r}$ en p_0, a la recta que pasa por el punto $p_0 = \mathbf{r}(t_0)$ y tiene la dirección del vector $\mathbf{r}'(t_0)$. Por tanto, $f(\boldsymbol{\lambda}) = \mathbf{r}(t_0) + \lambda \cdot \mathbf{r}'(t_0)$.

En consecuencia, el vector unitario tangente no depende de la parametrización, pues si $\boldsymbol{\varphi}$ y $\boldsymbol{\psi}$ son parametrizaciones distintas de la misma curva, entonces, $\boldsymbol{\psi}$ o $\boldsymbol{\lambda} = \boldsymbol{\varphi}$, tal que

[2] En caso de ser no nulo.

[3] En este caso el móvil está en reposo y puede haber un cambio brusco de dirección

$$\begin{aligned}\boldsymbol{\varphi}'(t) = \boldsymbol{\psi}'(\boldsymbol{\lambda}(t)) \cdot \boldsymbol{\lambda}'(t) \Longrightarrow & \frac{\boldsymbol{\varphi}'(t)}{|\boldsymbol{\varphi}'(t)|} \\ &= \frac{\boldsymbol{\psi}'(\boldsymbol{\lambda}(t)) \cdot \boldsymbol{\lambda}'(t)}{|\boldsymbol{\psi}'(\boldsymbol{\lambda}(t))| \cdot |\boldsymbol{\lambda}'(t)|} \\ &= \pm \frac{\boldsymbol{\psi}'(\boldsymbol{\lambda}(t))}{|\boldsymbol{\psi}'(\boldsymbol{\lambda}(t))|}\end{aligned} \tag{2.7}$$

donde el signo indica solo si las parametrizaciones mantienen o invierten la orientación de la curva.

Por tanto, si $f, g : \mathbb{R} \to \mathbb{R}^n$ y $u : \mathbb{R} \to \mathbb{R}$ son derivables, entonces, además de las propiedades análogas a las correspondientes funciones escalares, se puede aplicar las propiedades y reglas de las derivadas, siendo:

- $\frac{d}{dt}(f(t) \cdot g(t)) = f(t) \cdot g'(t) + f'(t) \cdot g(t)$
- $\frac{d}{dt}(f(t) \times g(t)) = f(t) \times g'(t) + f'(t) \times g(t)$
- $\frac{d}{dt}(f(u(t))) = u'(t) \cdot f'(u(t))$
- Si f es derivable y tiene longitud constante en un intervalo abierto I, entonces $f(t) \cdot f'(t) = 0, \forall t \in I$.

2.2.5 Integración de funciones vectoriales

Una función vectorial $f : \mathbb{R} \to \mathbb{R}^n$ es integrable cuando todos sus componentes lo son, y se define como

$$\int_a^b f(t)dt = \left(\int_a^b f_1(t)dt, \ldots, \int_a^b f_n(t)dt \right) \tag{2.8}$$

y por tanto cumple con

- Si $f : \mathbb{R} \to \mathbb{R}^n$ es continua en $\mathbb{R}$ y $g(t) = \int_a^t f(s)ds$, entonces g es derivable y $g'(t) = f(t), \forall t$.
- Si f y $|f|$ son integrables en $[a, b]$, entonces $|\int_a^b f(t)dt| \leq \int_a^b |f(t)|dt$.

2.2.6 Longitud del arco de la curva

Una función vectorial $\boldsymbol{\varphi}[a,b] \to \mathbb{R}^n$ se dice que es regular si $\boldsymbol{\varphi} \in C^1([a,b])$ y $\boldsymbol{\varphi}'(t) \neq 0, \forall t \in [a,b]$.

Es decir, una curva a trozos admite una parametrización $\boldsymbol{\varphi}$ regular a trozos, a partir de una partición p de $[a,b]$, tal que la restricción de $\boldsymbol{\varphi}$ a cada subintervalo abierto de p es regular.

Por lo tanto, una aplicación $\boldsymbol{\lambda} : [a,b] \to [\alpha,\beta]$ es un cambio regular de parámetro si

- $\boldsymbol{\lambda}$ es biyectiva.
- $\boldsymbol{\lambda} \in C^{(1)}[a,b]$.
- $|\boldsymbol{\lambda}'(t)| > 0, \forall t \in (a,b)$.

Entonces, una curva C con vector de posición $\mathbf{r}(t)$, define la longitud de arco de curva entre los puntos $\mathbf{r}(a)$ y $\mathbf{r}(b)$ a las longitudes de las poligonales inscritas a la curva entre dichos puntos, de esta forma, se dice que la curva es rectificable. Por lo tanto, dada una función $\boldsymbol{\varphi} : [a,b] \to \mathbb{R}^n$, se llama variación de $\boldsymbol{\varphi}$ con respecto a una partición $p = \{t_0, t_1, \ldots, t_m\}$ de $[a,b]$.

2.3 Ecuaciones de Frenet-Serret

En el año 1847, el matemático, astrónomo y meteorólogo francés Jean Frédéric Frenet escribió en su tesis, seis de las nueve fórmulas de las ahora conocidas *Fórmulas de Frenet-Serret*, que en ese momento no estaban expresadas en notación vectorial. Mientras que en el año de 1851, el matemático francés Joseph Alfred Serret completó el trabajo de la notación vectorial (álgebra lineal que se utiliza en la actualidad en la teoría de curvas).

Aguirre 2007, realiza un profundo estudio de las ecuaciones de Frenet-Serret. Dentro del cual, se describen las derivadas de sus campos, expresadas a través de $n-1$ funciones (es decir, las curvaturas).

Si se asume $\boldsymbol{\alpha} : I \to \mathbb{R}^n$ como una curva tal que $0 \in I$ y $\boldsymbol{\alpha}(0) = p$, se dice entonces que $\boldsymbol{\alpha}$ es una curva que pasa por el punto $p \in \mathbb{R}^n$, siendo $\boldsymbol{\alpha} : I \to \mathbb{R}^n$ una curva. Entonces, la variable en I se denomina parámetro de $\boldsymbol{\alpha}$.

Por tanto, partiendo de $\boldsymbol{\alpha} : I \rightarrow \mathbb{R}^n$ como una curva en el plano y sea $(\mathbf{R}_1, \ldots, \mathbf{R}_n)$ la referencia de Frenet, entonces, si $\mathbf{V}$ es un campo a lo largo de $\boldsymbol{\alpha}$, se define $\mathbf{V} = \sum_{i=1}^{n} (\mathbf{R}_i, \mathbf{V})\mathbf{R}_i$; en particular $(\forall = 1, \ldots, n)$.

$$\frac{d\mathbf{R}_j}{dt} = \sum_{i=1}^{n} \omega_{ij} \mathbf{R}_i, \text{ con, } \omega_{ij} := \left(\mathbf{R}_i, \ \frac{d\mathbf{R}_j}{dt} \right) \tag{2.9}$$

siendo $(\forall j = 1, \ldots, n-1)$, con

$$\begin{cases} \mathbf{R}_j \in sub(\boldsymbol{\alpha}', \ldots, \boldsymbol{\alpha}^{(j)}); \Rightarrow \dfrac{d\mathbf{R}_j}{dt} \in sub(\boldsymbol{\alpha}', \ldots, \boldsymbol{\alpha}^{(j+1)}); \Rightarrow \omega_{ij} = 0, \texttt{si } i > j+1 \\ < \mathbf{R}_i, \mathbf{R}_j >= cte, \forall i; \Rightarrow 0 = \dfrac{d\left(\mathbf{R}_i, \mathbf{R}_j\right)}{dt} = \omega_{ji} + \omega_{ij}, \forall i \end{cases} \tag{2.10}$$

Por lo tanto, a partir de estas expresiones se deduce que para $\frac{d\mathbf{R}_j}{dt}$ la forma matricial se expresa como

$$\left(\frac{d\mathbf{R}_1}{dt}, \ldots, \frac{d\mathbf{R}_n}{dt} \right) = (\mathbf{R}_1, \ldots, \mathbf{R}_n) \begin{pmatrix} 0 & -\omega_{21} & \ldots & 0 \\ \omega_{21} & 0 & \ddots & \vdots \\ \vdots & \ddots & 0 & -\omega_{n,n-1} \\ 0 & \ldots & \omega_{n,n-1} & 0 \end{pmatrix} \tag{2.11}$$

definido como las fórmulas de Frenet.

Por otra parte, toda curva regular es reparametrizable por la longitud de arco. Es importante distinguir, que dentro de una curva regular se distingue dos clases de parámetros, siendo: los que dependen de la trayectoria y los que dependen de la parametrización regular concreta. Entonces, el campo $\boldsymbol{\alpha}'$ depende de la parametrización, mientras que el campo $\boldsymbol{\alpha}'/|\boldsymbol{\alpha}'|$ depende sólo de la trayectoria orientada, y el conjunto de rectas afines tangentes a $\boldsymbol{\alpha}$ depende sólo de la trayectoria.

Por lo tanto, una curva regular $\boldsymbol{\alpha} : I \rightarrow \mathbb{R}^n$ está parametrizada por la longitud de arco si verifica la condición $|\boldsymbol{\alpha}'| = 1$. Entonces, $L(\boldsymbol{\alpha}|_{[a,b]}) = b - a, \forall a, b \in I(a < b)$, lo cual verifica que $(\boldsymbol{\alpha}', \boldsymbol{\alpha}'') = 0$ (lo que significa que la velocidad y la aceleración son mutuamente ortogonales).

2.3.1 Diedro de Frenet

Una curva en el plano está determinada por su curvatura (salvo reparametrizaciones que preservan la orientación y salvo movimientos directos).

Entonces, una curva en el plano definida por $\alpha : I \to \mathbb{R}^2$ produce una curvatura $\kappa_1 := \frac{\omega_{21}}{|\boldsymbol{\alpha}'|}$. Siendo, la curvatura de α denotada por κ determinada a partir de $(\mathbf{R}_1, \mathbf{R}_2)$ (diedro de Frenet), y definida por los términos $\mathbf{T} \equiv \mathbf{R}_1$ vector **tangente** y $\mathbf{N} \equiv \mathbf{R}_2$ vector **normal**.

Por lo tanto, las fórmulas de Frenet (2.11) se reducen a la siguiente expresión:

$$\left(\frac{d\mathbf{T}}{dt}, \frac{d\mathbf{N}}{dt}\right) = (\mathbf{T}, \mathbf{N}) = \begin{pmatrix} 0 & -|\boldsymbol{\alpha}'|\kappa \\ |\boldsymbol{\alpha}'|\kappa & 0 \end{pmatrix}, \tag{2.12}$$

siendo

$$\kappa := \frac{1}{|\boldsymbol{\alpha}'|} \tag{2.13}$$

Marcando una relevancia significativa al parametrizar la longitud de arco s (es decir, cuando $|\boldsymbol{\alpha}'| = 1$). Entonces, a partir de una referencia euclídea con origen en el punto $\boldsymbol{\alpha}(0)$ y con una base ortonormal dada por $[\overrightarrow{T}(0), \overrightarrow{N}(0)]$, la curva se define a través de coordenadas $\boldsymbol{\alpha}(s) \equiv (x(s), y(s))$, cuyo desarrollo en serie de Taylor en torno a $s = 0$, determina los valores de la curvatura y sus sucesivas derivadas en el 0, siendo

$$\boldsymbol{\alpha}(s) = \boldsymbol{\alpha}(0) + \underbrace{\frac{d\boldsymbol{\alpha}}{ds}(0)}_{\overrightarrow{T}(0)}s + \frac{1}{2}\underbrace{\frac{d^2\boldsymbol{\alpha}}{ds^2}(0)}_{\frac{d\overrightarrow{T}}{ds}(0)}s^2 + \frac{1}{3!}\underbrace{\frac{d^3\boldsymbol{\alpha}}{ds^3}(0)}_{\frac{d^2\overrightarrow{T}}{ds^2}(0)}s^3 + \bigcirc(s^4), \tag{2.14}$$

donde, $\bigcirc(s^4) = \frac{d^n\overrightarrow{T}}{ds^n}(0)$, entonces, partiendo de

$$\begin{cases} \dfrac{d\overrightarrow{T}}{ds} \overset{(2,12)}{=} \kappa\overrightarrow{N}, \\ \dfrac{d^2\overrightarrow{T}}{ds^2} = \dfrac{d\kappa}{ds}\overrightarrow{N} + \kappa\dfrac{d\overrightarrow{N}}{ds} \overset{(2,12)}{=} -\kappa^2\overrightarrow{T} + \dfrac{d\kappa}{ds}\overrightarrow{N}, \\ \dots \end{cases} \tag{2.15}$$

con lo que se alcanza la expresión

$$\begin{cases} x(s) = s - \dfrac{1}{3!}\kappa^2(0)s^3 + \bigcirc(s^4) \\ y(s) = \dfrac{1}{2}\kappa(0)s^2 + \dfrac{1}{3!}\dfrac{d\kappa}{ds}(0)s^3 + \bigcirc(s^4) \end{cases} \tag{2.16}$$

y sobre esto, se deduce la ecuación geométrica

$$\kappa(0) = \lim_{s\to 0} \frac{2y(s)}{s^2} = \lim_{s\to 0} \frac{2y(s)}{L(\mathbf{r}|_{[0,s]})^2} \tag{2.17}$$

misma que muestra diferentes propiedades geométricas, siendo

- el signo de κ positivo, si la curva "se dobla" en el sentido de la normal.
- la curvatura cambia de signo cuando cambia el sentido de recorrido (la normal se invierte, pero el contacto de la curva con el eje x en el punto es de orden 2);
- el módulo de la curvatura no depende de la parametrización.

Por lo tanto, para una curva plana parametrizada por la longitud de arco, las expresiones anteriores son válidas para valores de la curvatura y sus sucesivas derivadas en un punto ($\forall s \in I$).

2.3.2 Triedro de Frenet

Dentro del estudio realizado por Gray, Abbena y Salamon 2017, se elabora una descripción de la rama matemática conocida como geometría diferencial. En particular, se realiza un breve énfasis en las fórmulas de Frenet-Serret, a través de la descripción de las propiedades geométricas y cinemáticas de una partícula que se mueve a lo largo de una curva diferenciable continua en el espacio Euclídeo tridimensional $\mathbb{R}^3$. En específico, las fórmulas de Frenet-Serret describen las derivadas de los llamados *vectores unitarios tangente, normal y binormal* entre sí.

Entonces, se asume $\boldsymbol{\alpha} : I \to \mathbb{R}^3$ como una curva en el espacio, determinada por sus curvaturas $\kappa_1 := \frac{\omega_{21}}{\boldsymbol{\alpha}'} > 0$ y $\kappa_2 := \frac{\omega_{32}}{\boldsymbol{\alpha}'}$, definidas como curvatura κ y torsión τ de $\boldsymbol{\alpha}$, respectivamente. Por tanto, $(\mathbf{E}_1, \mathbf{E}_2, \mathbf{E}_3)$ define el triedro de Frenet de α, también conocido como $\mathbf{T} \equiv \mathbf{E}_1$ vector **tangente**, $\mathbf{N} \equiv \mathbf{E}_2$ vector **normal** principal y $\mathbf{B} \equiv \mathbf{E}_3$ vector **binormal**. Siendo, $(\mathbf{T}, \mathbf{N}, \mathbf{B})$ la referencia ortonormal positiva, resultante de $\mathbf{T} \times \mathbf{N} = \mathbf{B}$, consecuencia, las fórmulas de Frenet definidas en (2.11) se reducen a

$$\left(\frac{d\mathbf{T}}{dt}, \frac{d\mathbf{N}}{dt}, \frac{d\mathbf{B}}{dt}\right) = (\mathbf{T}, \mathbf{N}, \mathbf{B}) \begin{pmatrix} 0 & -|\boldsymbol{\alpha}'|\kappa & 0 \\ |\boldsymbol{\alpha}'|\kappa & 0 & -|\boldsymbol{\alpha}'|\tau \\ 0 & |\boldsymbol{\alpha}'|\tau & 0 \end{pmatrix} \tag{2.18}$$

siendo, $\kappa := \frac{1}{|\boldsymbol{\alpha}'|}$ y $\tau := \frac{1}{\boldsymbol{\alpha}'}$.

En consecuencia, al igual que en el caso de las curvas planas, se puede realizar el desarrollo de Taylor de la curva, expresada en referencia euclídea con origen el punto $\boldsymbol{\alpha}(0)$ y con base ortonormal dada por $[\overrightarrow{T}(0), \overrightarrow{N}(0), \overrightarrow{B}(0)]$. Por lo tanto, los primeros términos de dicho desarrollo, cuando α está parametrizada por la longitud de arco s (es decir, cuando $|\boldsymbol{\alpha}'| = 1$), son

$$\begin{cases} x(s) = s - \dfrac{1}{3!}\kappa^2(0)s^3 + \bigcirc(s)^4 \\ y(s) = \dfrac{1}{2}\kappa(0)s^2 + \dfrac{1}{3!}\dfrac{d\kappa}{ds}(0)s^3 + \bigcirc(s)^4 \\ z(s) = \dfrac{1}{3!}\kappa(0)\tau(0)s^3 + \bigcirc(s)^4 \end{cases} \tag{2.19}$$

de lo cual se deducen diferentes propiedades geométricas de la curva, tal que

$$\tau(0) = \lim_{s\to 0} \frac{3!z(s)}{\kappa(0)s^3} = \lim_{s\to 0} \frac{3!z(s)}{\kappa(0)L(\boldsymbol{\alpha}|_{[0,s]})^3} \tag{2.20}$$

por tanto, se mantiene una curvatura κ positiva, siendo:

- el signo de τ positivo, si la curva “cambia” en el sentido indicado por la binormal;
- la torsión no cambia de signo cuando cambia el sentido de recorrido (la binormal se invierte, con el plano xy en el punto es de orden 3);
- el módulo de la torsión no depende de la parametrización.

Entonces, para cada $t \in I$, las rectas que pasan por $\boldsymbol{\alpha}(t)$ tienen por direcciones $\overrightarrow{T}(t), \overrightarrow{N}(t)$ y $\overrightarrow{B}(t)$, los cuales se definen como **recta tangente**, **recta normal principal** y **recta binormal** de $\boldsymbol{\alpha}$ en t, respectivamente. Los planos afines que pasan por $\boldsymbol{\alpha}(t)$ y tienen por vectores directores $\overrightarrow{T}(t), \overrightarrow{N}(t)$ y $\overrightarrow{B}(t)$ se denominan **plano osculador**, **plano normal** y **plano rectificante** de $\boldsymbol{\alpha}$ en t, respectivamente. En consecuencia, la curvatura κ mide cuánto se desvía la curva contenida en su recta tangente y la torsión τ mide cuánto se desvía en su plano osculador.

2.3.3 Cálculo de la curvatura y la torsión

Si se asume que $\boldsymbol{\alpha} : I \to \mathbb{R}^3$ como una curva en el espacio tal que $\kappa > 0$, $\tau \in F(I)$ se definen como su curvatura y torsión respectivamente, entonces

$$\kappa = \frac{\boldsymbol{\alpha}' \times \boldsymbol{\alpha}''}{|\boldsymbol{\alpha}'|^3} \quad \text{y} \quad \tau = \frac{\det(\boldsymbol{\alpha}', \boldsymbol{\alpha}'', \boldsymbol{\alpha}''')}{|\boldsymbol{\alpha}' \times \boldsymbol{\alpha}''|^2} \tag{2.21}$$

por otra parte, si $|\boldsymbol{\alpha}| = 1$, entonces:

$$\kappa = |\boldsymbol{\alpha}''| \quad \text{y} \quad \tau = \frac{\det(\boldsymbol{\alpha}', \boldsymbol{\alpha}'', \boldsymbol{\alpha}''')}{|\boldsymbol{\alpha}''|^2} \tag{2.22}$$

2.4 Curvas Splines

En la década de 1940 el matemático estadounidense de origen rumano Isaac Jacob Schoenberg (1903-1990), realizó un trabajo de análisis numérico sobre curvas diferenciables definidas en porciones por medio de polinomios, trabajo denominado “curvas splines”, tal como se resalta en el estudio de Micula y Micula 2012. Siendo un trabajo que posee una representación completa y sencilla, para una representación informática.

El concepto de spline tiene una amplia gama de aplicaciones, siendo una de ellas el ajuste de curvas, a través de la interpolación de puntos n-dimensionales. En específico, la spline $S(x)$ definida en el intervalo $[a, b]$, parte de un conjunto de puntos $(x_0, x_1, \ldots, x_n)$ (los cuales se asumen ordenados de forma creciente es decir, $x_0 < x_1 \ldots < x_n$) con el objetivo de ajustar una función interpolante de grado k en cada intervalo de la partición de los $(N + 1)$ puntos.

En consecuencia, se denomina función spline de grado p, a una función polinómica definida a trozos, donde, $S(x)$ se define en $[x_0, x_n]$. Entonces, $S_i = S|_{x_i, x_{i+1}}$ se define como el polinomio de grado $p : i = 1, \ldots, (N - 1)$. Siendo, S diferenciable $p - 1$ veces en $[x_0, x_n]$, $C^{p-1}([x_0, x_n])$.

En función del grado, las splines se pueden tratar de diferente forma. En específico, las de menor grado (menor o igual a 1) se definen como lineales, parabólicos y cúbicos.

2.4.1 Spline lineal

Dado el conjunto de puntos $\Delta = a = x_0 < x_1 < \ldots < x_n = b$ del intervalo $[a, b]$, S es un spline lineal asociado a Δ siempre que S_i de S en cada intervalo $[x_i, x_{i+1}]$, para $i = 0, \ldots, (N-1)$, sea un polinomio de grado no superior a 1. En consecuencia, $S(x) \in C^0([a, b])$, es decir, S es una función continua en el intervalo $[a, b]$.

Entonces, si en cada intervalo de Δ se construye un polinomio de grado 1 que aproxime a la función, se deben calcular dos incógnitas por intervalo, con coeficientes $[a_i, b_i]$. Por tanto, la continuidad en los $(N-1)$ puntos interiores se expresa por

$$S_i(x_i) = S_{i+1}(x_i) \ \forall \ i = 1, \ldots, (N-1) \tag{2.23}$$

En consecuencia, se deben cumplir un total de $(N-1)$ condiciones, además de las $(N+1)$ de interpolación, lo cual produce $2n$ incógnitas y $(N-1)+(N+1)$ ecuaciones.

2.4.2 Spline parabólico

Las curvas splines construidas por polinomios de grado menor o igual a 2 exigen una continuidad de primer orden C^1. Es decir, las primeras derivadas (tangentes a la curva) de las curvas conservan las pendientes en todo el intervalo de definición.

Entonces, dado el conjunto de puntos $\Delta = a = x_0 < x_1 < \ldots < x_n = b$ del intervalo $[a, b]$, la función S define un spline parabólico asociado a Δ siempre que S_i de S en cada intervalo $[x_i, x_{i+1}]$, para $i = 0, \ldots, (N-1)$, sea un polinomio de grado no superior a 2. En consecuencia, $S(x) \in C^1([a, b])$, es decir, S es una función diferenciable, con derivada continua, en el intervalo $[a, b]$. Por lo tanto, S es un spline parabólico para el conjunto de puntos (x_i, y_i), para $i = 0, 1, \ldots, n$.

2.4.3 Splines cúbicos

La curvas splines construidas por polinomios de grado menor o igual a 3 exigen una condición de continuidad de segundo orden C^2. Es decir, cumplen con condiciones de segundas derivadas, por tanto, el punto de unión de las curvas es igual, lo que significa que la transición es suave de una sección de curva con la siguiente. En consecuencia, al exigir la condición C^2 en la curva spline, ésta puede ser tratada dependiendo de su clase C^1 o C^2.

- *Spline cúbico C^1 o Interpolación de Hermite.* Dado el conjunto de puntos $\Delta = a = x_0 < x_1 < \ldots < x_n = b$ del intervalo $[a, b]$, diremos que la función S es un Spline cúbico C^1 asociado a Δ siempre que la restricción S_i de S en cada intervalo $[x_i, x_{i+1}]$, para $i = 0, \ldots, n-1$, sea un polinomio de grado no superior a 3. Por tanto, $S(x) \in C^1([a, b])$, es decir, S es una función diferenciable, con derivada continua, en el intervalo $[a, b]$.

  Lo cual indica que S es una función cúbica, continua y diferenciable en los $n + 1$ puntos de control C^1 y los polinomios, en cada intervalo, se expresan tal que $S_i(x) = a_i x^3 + b_i x^2 + c_i x + d_i$.

- *Spline cúbico C^2*. Dado el conjunto de puntos $\Delta = \{a = x_0 < x_1 < \ldots x_n = b\}$ del intervalo $[a, b]$, entonces, la función S es un Spline cúbico C^2 asociado a Δ siempre que, la restricción S_i de S en cada intervalo $[x_i, x_{i+1}]$, para $i = 0, \ldots, n-1$, sea un polinomio de grado no superior a 3. Por tanto, $S(x) \in C^2([a, b])$, es decir, S es una función dos veces diferenciable, con las dos derivadas continuas, en el intervalo $[a, b]$.

  Lo cual indica que S sea un Spline cúbico C^2 interpolante para el conjunto de puntos (x_i, y_i), para $i = 0, 1, \ldots, n$, siempre que S es un Spline cúbico C^2 asociado a Δ y $S(x_i) = y_i$ para $i = 0, \ldots, n$. Por lo tanto, S es una función cúbica, continua y dos veces diferenciable en los $n + 1$ puntos base, C^2, y los polinomios, en cada intervalo, se expresan tal que $S_i(x) = a_i x^3 + b_i x^2 + c_i x + d_i$.

2.4.4 Curvas Bézier

Las curvas de Bézier realizan una descripción matemática de curvas numéricamente estables, tal como se detalla en Gordon y Riesenfeld 1974b; Segovia y col. 1991. Por una parte, en Hammouda y col. 2013 se estudia un caso especial de una curva spline, en que la curva tiene sólo un componente polinomial. Mientras que, Lorentz 2013 afirma que los polinomios de Bernstein de grado n (con $n > 0$) también pueden interpretarse como las funciones de base de la

B-spline de grado n sobre el dominio de esos polinomios, como se ratifica en Gordon y Riesenfeld 1974a.

La curva de Bézier del grado n puede ser expresada como

$$\begin{aligned}\mathbf{B}(t) &= \sum_{i=0}^{n} \binom{n}{i} \mathbf{P}_i (1-t)^{n-i} t^i \\ &= \mathbf{P}_0 (1-t)^n + \binom{n}{1} \mathbf{P}_1 (1-t)^{n-1} t + \cdots + \mathbf{P}_n t^n, \\ & t \in [0,1]\end{aligned} \tag{2.24}$$

donde $\mathbf{P} := [\mathbf{P}_0^T, \mathbf{P}_1^T, \ldots, \mathbf{P}_n^T] \in \mathbb{R}^{3(n+1)}$ son el conjunto de puntos de control y $\mathbb{R}^3$ es el espacio tridimensional donde se define la curva. En otras palabras, el grado n de la curva de Bézier es una interpolación entre los $(n+1)$ puntos de control. Por otra parte, las curvas de Bézier pueden ser expresadas en términos de polinomios de Bernstein de grado n, tal que

$$\mathbf{B}(t) = \sum_{i=0}^{n} \mathbf{P}_i \mathbf{b}_{i,n}(t), \ t \in [0,1] \tag{2.25}$$

donde los polinomios $\mathbf{b}_{i,n}(s) \in \mathbb{R}$

$$\mathbf{b}_{i,n}(t) := \binom{n}{i} t^i (1-t)^{n-i}, \ i = 0, \ldots, n \tag{2.26}$$

también conocidos como binomio de Newton. Por lo tanto, una curva de grado n tiene un conjunto de puntos de control de $n+1$ vértices.

2.5 Clotoide

La clotoide (también llamada espiral de Cornú o espiral Euler), es una curva plana en forma de espiral doble con simetría central (ver Figura 2.3), tangente al eje de las abscisas en el origen y cuyo radio de curvatura R disminuye de manera inversamente proporcional a la distancia recorrida sobre ella. Es por ello que en el punto origen de la curva, el radio es infinito, es decir, su curvatura crece linealmente desde $\frac{1}{R} = 0$, hasta $\frac{1}{R} = \infty$. Debido a esta propiedad, se ha aplicado como una curva de transición entre arcos, líneas rectas o un arco y una línea recta. Además, la curva clotoide y su desplazamiento han sido utilizadas en las últimas 7 décadas en el diseño de carreteras, en vías de ferrocarril y montañas rusas, como se ha descrito en una amplia variedad de trabajos, siendo ejemplo de estos, los realizados por BAASS 1984; Hartman 1957; Sadeghi y col. 2022; Lambert, Romano y Watling 2019; Afrianto y col. 2018.

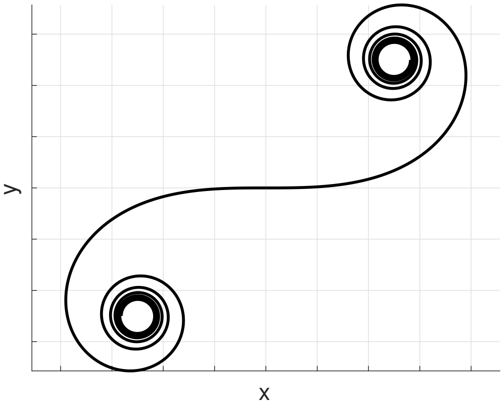


Figura 2.3: Curva Clotoide.

Por definición, la ecuación intrínseca de la Clotoide es:

$$C^2 = \rho \cdot s \tag{2.27}$$

donde, C es el parámetro característico de la clotoide, ρ es el radio de curvatura y s es la longitud del arco de la clotoide. Es decir, que la ley de curva en la clotoide se definie por cada valor de C, siendo el producto de la longitud s por el radio R el mismo en todos los puntos de la clotoide. En consecuencia, C expresa de forma unívoca el tamaño de la clotoide.

Por tanto, se define a la clotoide como una curva cuyas ecuaciones paramétricas vienen expresadas por $S(t)$ y $C(t)$, siendo las integrales de Fresnel, tal que:

$$C'(t)^2 + S'(t)^2 = \sin^2(t^2) + \cos^2(t^2) = 1 \tag{2.28}$$

siendo la longitud de vector tangente definida por la unidad, mientras que t define la longitud de arco medida a partir del origen, por lo que la curva tiene longitud infinita.

De esta forma, a partir de (2.27) se puede establecer una longitud de arco, que es proporcional al radio en cada punto de la curva (curvatura κ):

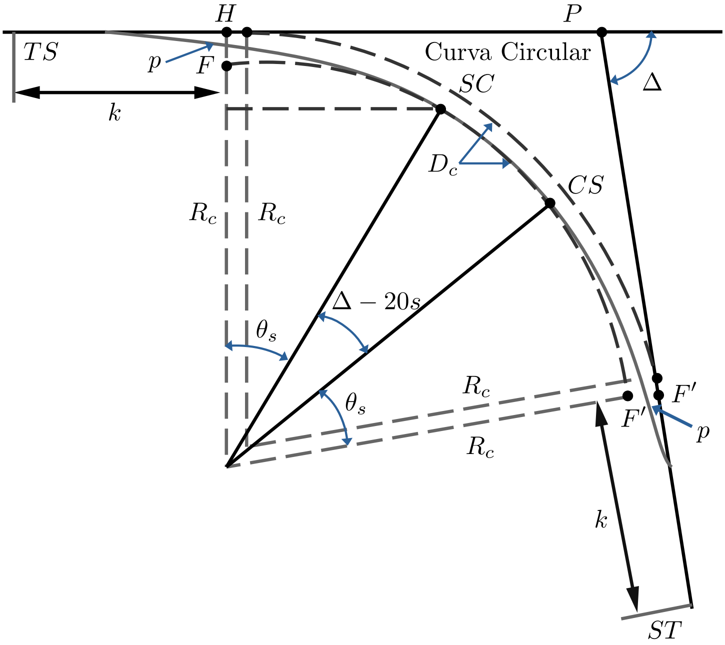


Figura 2.4: Términos y símbolos de la Clotoide.

$$\kappa = \frac{1}{\rho} = \frac{L}{A^2} \tag{2.29}$$

La Figura 2.4 y el listado de términos en el cuadro 2.1 muestran el significado de los símbolos y términos empleados en la construcción de la clotoide. Es importante, resaltar que varios de estos términos son variables genéricas y hacen referencia a ecuaciones futuras en este documento.

De esta forma, a partir de la ecuación fundamental de la clotoide (2.27), y puesto que en cualquier punto de la curva se cumple que:

$$\rho \cdot l = C^2 \rightarrow \rho \cdot l = R_c \cdot L_s \tag{2.30}$$

el área diferencial se define como:

$$dl = \rho \cdot d\theta \tag{2.31}$$

Cuadro 2.1: Símbolos y su significado en la construcción de una clotoide.

Simbolo	Significado
PI	Punto de intersección de las tangentes principales.
TS	Punto de cambio de tangente a espiral (punto de inicio de la espiral de entrada)
SC	Punto de cambio de espiral a curva circular.
CS	Punto de cambio de curva circular a espiral.
ST	Punto de cambio de espiral a tangente (punto final de la espiral de salida).
SS	Punto de cambio de una espiral a otra.
L_s	Longitud total de la espiral del TS al SC o bien del CS al ST.
L	Longitud desde TS o ST hasta un punto cualquiera de la espiral.

Cuando ρ es muy grande la variación es despreciable (ver Figura 2.5), mientras que al despejar ρ y sustituyendo en la ecuación anterior, se obtiene:

$$\frac{dl}{d\theta} \cdot l = \rho \cdot L_s \qquad d\theta = \frac{l \cdot dl}{\rho L_s} \tag{2.32}$$

siendo después de la integración

$$\theta = \frac{l^2}{2 \cdot \rho \cdot L_s} + C \tag{2.33}$$

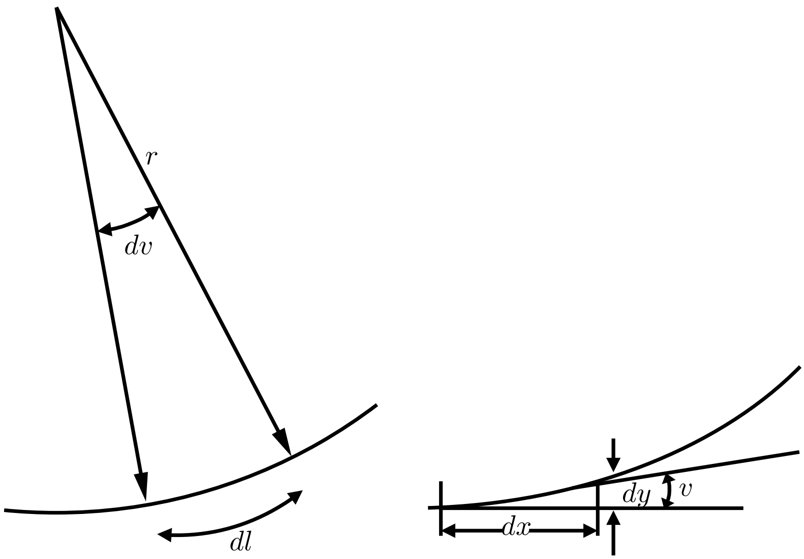
asumiendo $C = 0$ puesto que $\theta = 0$ con $l = 0$, la expresión se reduce

$$\theta = \frac{l^2}{2 \cdot \rho \cdot L_s} \tag{2.34}$$

siendo el valor de θ en radianes. Sin embargo, en el punto F, $\theta = \theta_s$ y $l = L_s$, por lo que

$$\theta_s = \frac{(L_s)^2}{2 \cdot R_c \cdot L_s} = \frac{L_s}{2 \cdot R_c} \tag{2.35}$$

Simbolo	**Significado**
L_1	Longitud entre cualquiera de dos puntos sobre la espiral.
θ	Ángulo central del arco en un punto cualquiera de la espiral.
θ_s	Ángulo central del arco de espiral L_s, llamado "ángulo de la espiral".
Φ	Ángulo de desviación de la espiral en el TS, desde la tangente inicial a un punto de la curva $R = radio$.
G	Grado de curvatura de la espiral en cada punto.
G_c	Grado de curvatura del círculo desplazado al que resulta tangente la espiral en el SC.
K	$\frac{D_c}{L_s}$ relación de cambio de grado de curvatura por metro de la espiral.
Δ	Ángulo central total de la curva circular.
Δ_s	Ángulo central del arco circular de longitud L_c que va del SC al CS.
y	Ordenada a la tangente de cualquier punto de la espiral, con referencia al TS y la tangente inicial.
y_s	Ordenada a la tangente del SC.
x	Distancia en la tangente de cualquier punto de la espiral, con referencia al TS y la tangente inicial.
x_s	Distancia en la tangente del SC.
p	Ordenada desde la tangente inicial al PC del círculo desplazado; $p = y_s - R_c(1 - \cos\theta_s)$.
k	Abscisa del PC desplazado, referido al TS; $k = x_s - R_c \operatorname{sen}\theta_s$.
a	Distancia del PC al TS de la espiral que se aplica a una curva circular existente (no desplazada).
T_s	Distancia total en la $tangente = distancia$ desde el PI al TS, o bien del PI al ST; $T_s = (R_c + p)\tan\frac{1}{2}\Delta + k$.
E_s	Externa de la curva total o distancia del medio del arco circular al PI. $Es = (R_c + p)(\sec\frac{1}{2}\Delta - 1) + p$.
LT	Longitud de la tangente larga de la espiral.
ST	Longitud de la tangente corta de la espiral.
LC	Cuerda de la espiral entre TS y SC.



(a) (b)

Figura 2.5: Diferencial de una clotoide.

Finalmente, sustituyendo L_s por su expresión en función de C y de R_c, se obtiene el ángulo central del arco de espiral

$$\theta_s = \frac{\frac{C^2}{R_c}}{2 \cdot R_c} = \frac{C}{2 \cdot (R_c)^2} \rightarrow R_c = \frac{C}{\sqrt{2 \cdot \theta_s}} \tag{2.36}$$

Por lo cual se mantiene la relación entre el parámetro C, el radio R_c y el desarrollo L_s con el valor angular θ_s.

Entonces, dado el espacio diferencial y de acuerdo con la Figura 2.5(b), se define dx, tal que:

$$\begin{gathered} \cos(\theta) = \frac{dx}{dl} \\ dx = \cos(\theta) dl \end{gathered} \tag{2.37}$$

que expresado en serie de cosenos

$$dx = \sum \frac{(-1)^{(n-1)} \cdot \theta^{2n-2}}{(2n-2)} dl \tag{2.38}$$

sustituida la ecuación (2.34) en (2.38) e integrada

$$\begin{aligned} \delta x &= \sum \frac{(-1)^{(n+1)} 1^{(4n-4)}}{(2n-2)!(2 \cdot R_c \cdot L_s)^{(2n-2)}} \delta l \\ \int 1^{(4n-4)} dl &= \frac{1^{(4n-3)}}{(4n-3)} + C \end{aligned} \tag{2.39}$$

dado que $x = 0$ cuando $l = 0$, entonces, $C = 0$, con lo que la expresión se define como:

$$X = \sum \frac{(-1)^{n+1} 1^{(4n-3)}}{(4n-3) - (2n-2)!(2 \cdot R_c \cdot L_s)^{(2n-2)}} \tag{2.40}$$

Por lo tanto, para los cuatro primeros términos se expresa

$$X = 1 - \frac{1^5}{5 \cdot 2!(2R_c \cdot L_s)^2} + \frac{1^9}{9 \cdot 4!(2R_c \cdot L_s)^4} + \frac{1^{13}}{13 \cdot 6!(2R_c \cdot L_s)^6} \tag{2.41}$$

Por otra parte, con el objetivo de deducir y, partiendo de la Figura 2.5(a), con

$$\begin{aligned} \sin(\theta) &= \frac{dy}{dl} \\ dy &= \sin(\theta) dl \end{aligned} \tag{2.42}$$

Expresado en serie de senos

$$dy = \sum \frac{(-1)^{(n-1)} \theta(2n-2)}{(2n-1)!} \tag{2.43}$$

reemplazando (2.34) en (2.43) e integrando

$$\begin{aligned} dy &= \sum \frac{(-1)^{(n+1)} 1^{(4n-2)}}{(2n-2)!(2 \cdot R_c \cdot L_s)^{(2n-1)}} \\ Y &= \sum \frac{(-1)^{(n+1)} l^{(4n-2)}}{(2n-1)!(4n-1)(2 \cdot R_c \cdot L_s)^{(2n-1)}} \end{aligned} \tag{2.44}$$

se definen los primeros cuatro términos como:

$$Y = \frac{l^3}{3(2R_c \cdot L_s)} - \frac{l^7}{7 \cdot 3!(2R_c \cdot L_s)^3} + \frac{l^{11}}{11 \cdot 5!(2R_c \cdot L_s)^5} - \frac{l^{15}}{15 \cdot 7!(2R_c \cdot L_s)^7} \tag{2.45}$$

Por lo tanto, a partir de las ecuaciones (2.40) y (2.44) se determinan los valores de X e Y en cualquier punto, siempre que se definan los parámetros C (o R_c y L_s) y el punto en concreto.

2.5.1 Aplicaciones de la curva Clotoide

En la práctica, la curva clotoide ofrece diferentes propiedades interesantes como se muestra en la Figura 2.6, donde la línea recta se muestra en negro sólido, la curva de transición en azul sólido, la curva circular en rojo discontinuo y los puntos de tangencia en verde. Siendo ejemplo de estas propiedades las siguientes:

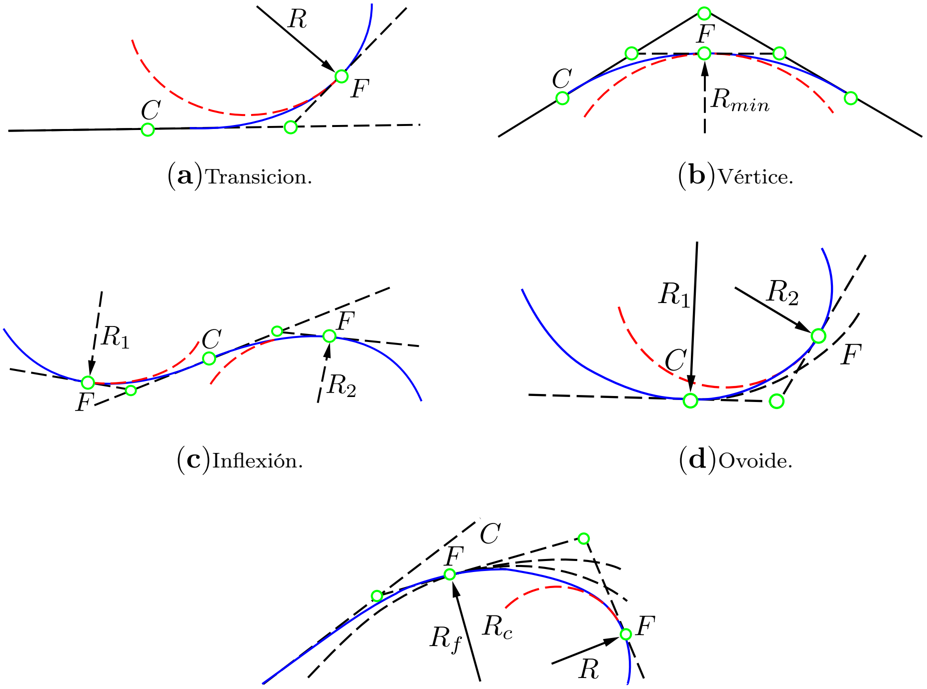
Transición Se define por un arco, desde el radio ∞ que se une con una recta, hasta el radio del arco circular siguiente (ver Figura 2.6(a)).

Vértice Describe la transición entre dos rectas de direcciones distintas. Se compone de dos ramas de clotoide con el mismo radio de curvatura κ y tangente común en su punto de contacto (ver Figura 2.6(b)).

Inflexión Definida por una curva en forma de S, que une dos círculos de curvaturas κ opuestas y sin segmento recto intermedio. Formada por dos ramas clotoidales, cuyo radio es infinito ∞, y el punto de origen para las ramas y la tangente es común (ver Figura 2.6(c)).

Ovoide Representada por una secuencia de curvas de tipo círculo - clotoide - círculo, donde la curvatura κ mantiene el mismo sentido. Mientras que el arco de clotoide intermedia mantiene tangentes comunes y radios iguales en los puntos de contacto con los círculos (ver Figura 2.6(d)).

Serie de clotoides Definida por una concatenación de arcos de clotoide con distintos parámetros, donde las curvaturas son dirigidas y crecientes en el mismo sentido, además de tangentes comunes con la misma curvatura κ, para cada dos arcos sucesivos en su punto de contacto (ver Figura 2.6(e)).



(**a**)Transicion. (**b**)Vértice.

(**c**)Inflexión. (**d**)Ovoide.

(**e**)Serie de clotoides.

Figura 2.6: Aplicaciones en las curvas clotoide.

2.6 Aproximación a trayectorias suaves

Una rama subsecuente de investigación y desarrollo que además, es el motivo de estudio del presente trabajo de tesis, es la construcción de trayectorias suaves (del inglés *smooth planning*) navegables por AVs, para lo cual se han explotado las cualidades de las curvas de transición.

En este sentido, las curvas de transición comúnmente utilizadas para los ferrocarriles y carreteras, se definen como curvas que proporcionan un cambio de curvatura gradual, a partir de un tramo recto a otro circular. Así, el concepto de transición obedece a la necesidad de adaptación de AVs que se desplazan a altas velocidades, a través de una trayectoria continua. Entonces, el objetivo es alcanzar comodidad y estabilidad en el viaje de los AVs sobre tales trayectorias.

El uso de las curvas de transición brinda ventajas en relación al coeficiente de seguridad en un AV, lo cual significa que se reduce la tendencia a desviarse la trayectoria. De hecho, una curva de transición se construye a través de la unión de un conjunto de líneas rectas y curvas circulares. Entonces, el objetivo de proporcionar suavidad en una curva de radio infinito ∞ (es decir, como es en una línea recta), con otra curva circular de radio determinado, se puede apreciar en la la Figura 2.7, donde la línea recta se muestra en negro, la curva de transición en azul, la curva circular en rojo y los puntos de tangencia en verde.

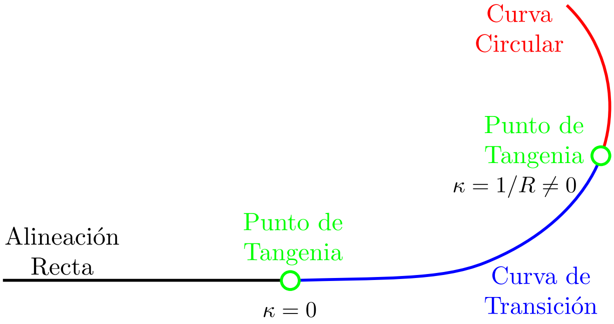


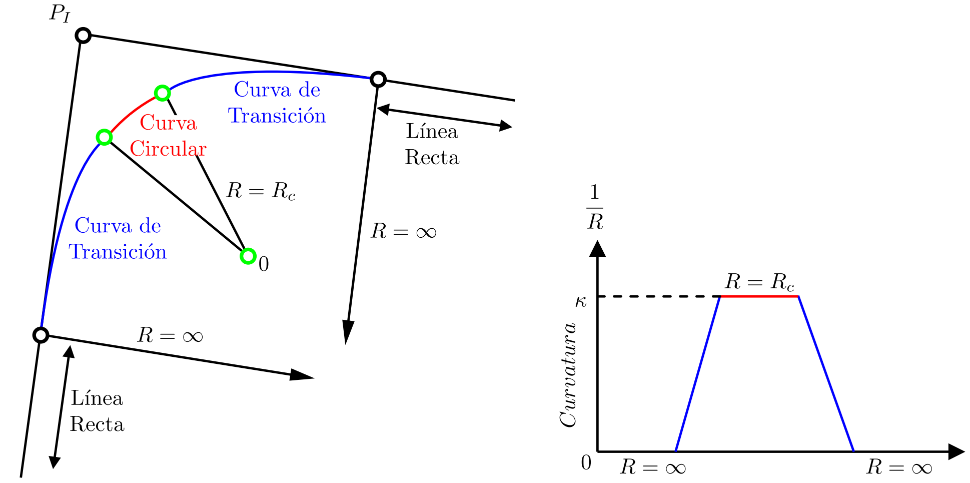
Figura 2.7: Unión entre líneas recta, curva circular y curva de transición.

Por otra parte, partiendo del concepto de fuerza centrífuga F_c, la cual se define como:

$$F_c = \frac{M \cdot v^2}{R} \tag{2.46}$$

donde, F_c representa la fuerza centrífuga, M la masa del AV, v la velocidad del mismo AV y R el radio de la curva circular. Entonces, asumiendo que la fuerza centrífuga es igual a cero en un tramo recto, es decir, $F_c = 0 : 1/\kappa = \infty; \kappa = 0$. Por otra parte, si el AV ingresa en una curva, la fuerza centrífuga adquiere un valor diferente de cero, es decir, $F_c \neq 0 : 0 < 1/\kappa < \infty$, lo cual significa que el AV podría sufrir un vuelque a la entrada de la curva, a menos que se realice una disminución de velocidad v, lo cual implica disminución en F_c, según se aprecia en 2.46.

Entonces, si un AV se desplaza a través de una curva de transición, al momento de cambiar la trayectoria desde la línea recta hacia la curva, se ejerce presión constante y creciente sobre el giro, lo cual implica desaceleración gradual con el objetivo de no perder la trayectoria (ver Figura 2.8).



(**a**)Curva de transición.

(**b**)Efecto de curvatura debido a la curva de transición.

Figura 2.8: Fuerza centrífuga debido al cambio de trayectoria.

Por lo tanto, explotar las cualidades de una curva de transición, significa una variación decreciente del radio R, lo que implica que la fuerza centrífuga F_c aumente de manera gradual. El objetivo es evitar que R decrezca hasta el punto que se produzca el vuelque del AV (debido a un exceso de velocidad), al ingresar en una curva.

En consecuencia, con el objetivo de que un AV recorra una trayectoria con velocidad uniforme (de tal forma que el AV se mantenga dentro de tal trayectoria), se han propuesto aproximaciones a partir de curvas de transición. Para ello, la longitud de la curva de transición debe permitir al AV desplazarse a la velocidad de diseño, y así pasar de una línea recta hasta una curva sin pérdida de sustentación.

Por otra parte, abordar el problema de curvas de transición en trayectorias 3D, implica la consideración en los tres ejes (x, y, z) al mismo tiempo, lo cual significa considerar tanto la curvatura como la torsión.

A continuación, se realiza un breve pero completo resumen de estudios compilados y analizados a lo largo del trabajo de investigación. Aunque este trabajo busca completar una generación de curvas navegables en el espacio 3D, un estudio y análisis 2D es necesario. Por lo tanto, el trabajo de investigación toma como punto de partida el estudio y desarrollo en el espacio 2D, para posteriormente hacer la extensión al espacio 3D.

2.6.1 Transición a Trayectorias 2D

El trabajo realizado por Yang y col. 2014a tiene como objetivo construir una trayectoria suave, partiendo de una trayectoria generada por el planificador Árbol aleatorio de exploración rápida (RRT, del acrónimo *Rapidly-exploring Random Tree*) que genera un conjunto de puntos de control p_i (también conocidos como nudos) que son utilizados para construir una trayectoria de tipo spline, parametrizada por limitaciones internas debidas a las restricciones cinemáticas y dinámicas del AV, con el objetivo de generar una continuidad en curvatura κ a lo largo de la trayectoria. RRT realiza una exploración por el entorno y posteriormente, se efectúa una operación de purga de nudos que no sean útiles para llegar al objetivo de mantener una curvatura con un radio circular definido. Finalmente, para llegar a la construcción completa de la curva spline, se toman grupos de tres nudos contiguos, sobre los cuales se insertan ocho o nueve nudos y así construir la curva spline que satisfaga la curvatura establecida, procedimiento que toma el nombre de Árbol aleatorio de exploración rápida suave (SRRT, del acrónimo *Smooth Rapidly-exploring Random Tree*).

En Elbanhawi, Simic y Jazar 2015 se busca mantener continuidad en una trayectoria, además de no sobrepasar una máxima curvatura κ en un AV. La curva se genera como una curva paramétrica, y dentro de esos parámetros se incluye la consideración de las restricciones del AV. Un conjunto de puntos de control P_i son generados y, con el objetivo de mantener la curvatura, el trabajo se realiza en base a segmentos de tres P_i, además de la longitud y el ángulo formado entre dichos pares de P_i. El objetivo es ubicar puntos de control intermedios de tal modo que κ se mantenga dentro de los límites establecidos. Entonces se genera una nueva curva de tipo B-spline a la cual se aplican pesos en sus P_i, lo cual permite que la curva se desplace hacia uno o varios de tales P_i.

Las Curvas Bézier introducidas en la sección 2.4.4, poseen diversas ventajas debido a su fácil implementación en 3D, además de las propiedades de seguimiento de los P_i. En este sentido, la Espiral de Bézier Cúbica Continua G^2 (G2CBS, del acrónimo G^2 *Continuous Cubic Bézier Spiral*) como se define en Yang y Sukkarieh 2008a, opera sobre entornos naturales y satisface las restricciones de máxima curvatura κ y curvatura continua. El trabajo parte de los resultados obtenidos por un planificador RRT al cual se aplica un proceso de purga con el objetivo de mejorar sus resultados y con ello la construcción de la curva Bézier cumpla mejor con sus objetivos.

El trabajo propuesto en Sahingoz 2014, construye una trayectoria partiendo del funcionamiento de un algoritmo genético, que junto con la definición de varias limitaciones (P_i mínimos y/o máximos, tiempo de vuelo y distancia de vuelo), se determina la trayectoria. De esta forma, la trayectoria resultante es utilizada para generar una curva suave Bézier, navegable por un UAV. Los puntos topológicos del entorno por el cual se construye la trayectoria son asumidos, es decir, provienen de un sistema satelital o de un grupo externo de UAVs que llevan sensores abordo.

En el artículo de Montes, Mora y Tornero 2007 se presenta un algoritmo que realiza una maniobra de adelantamiento en carretera basado en una curva de tipo clotoide. La maniobra se realiza escalando una curva paramétrica general basada en una aproximación (fuera de línea) de las integrales de Fresnel sobre curvas Bézier Racionales (RBC, del acrónimo *Rational Bézier Curve*), que después se utilizan en la generación de trayectorias clotoides (en línea). Las ecuaciones de RBC son expresadas en la forma de matriz $P = B \cdot w$ con el objetivo de aproximar a la integral de Fresnel $C(\gamma)$, al ajustar los P_i de inicio y final. Entonces, con el objetivo de aproximar el término de la integral de Fresnel $S(\gamma)$, los tres primeros puntos de control deben estar alineados con la recta, y por lo tanto, los pesos requeridos deben ser ajustados.

El trabajo presentado en González y col. 2014, describe un algoritmo que genera una trayectoria con una curvatura continua, la trayectoria considera las restricciones del AV. Esta aproximación de trayectoria se genera como una curva paramétrica de tipo Bézier polinómico, el cual permite determinar el control longitudinal dinámico de acuerdo al tipo de trayectoria. El punto de partida, son los resultados de un planificador global de tipo RRT que devuelve un conjunto de P_i, además del máximo radio de giro que puede realizar el AV. Para lo cual, se implementa una Bézier de grado 4 (siempre que una Bézier de grado 3 no cumpla con los requisitos previos definidos).

El aporte científico realizado por Choi, Curry y Elkaim 2008, propone dos algoritmos de planificación de trayectorias que utilizan las curvas Bézier, partiendo de la posición adecuada de los P_i para un AV con ciertas restricciones. En Kawabata y col. 2015, se propone una generación de trayectorias suaves para un AV, utilizando la curva Bézier en 2D y 3D. Los trabajos realizados por Choi, Curry y Elkaim 2008; Rastelli, Lattarulo y Nashashibi 2014, discuten la ubicación optimizada de los P_i. En Walton, Meek y Ali 2003, se utiliza las curvas Bézier como curvas de transición para unir un segmento recto con un segmento curvo, y también en la generación de trayectorias de tipo clotoide. Mientras que en Montes, Mora y Tornero 2007; Montés y col. 2008 se ha demostrando la modularidad y extensibilidad en curvas clotoides, a parir de curvas Bézier. La generación de trayectorias para los vehículos en entornos urbanos ha sido discutida en Han y col. 2010; Pérez y col. 2013; González y col. 2014. Finalmente, algunos investigadores como Liang, Zheng y Li 2012 han utilizado las curvas Bézier para el estacionamiento automático de vehículos.

El trabajo propuesto en Fraichard y Ahuactzin 2001, aborda el problema de generación de trayectorias con curvatura continua al plantear un planificador global que considera los obstáculos del entorno. La metodología propuesta, calcula una posible trayectoria en base a segmentos lineales y arcos en forma de curvas clotoides, con el objetivo de alcanzar una curvatura continua, misma que es utilizada en curvas Dubins. El punto de partida explota las cualidades de un planificador de trayectoria de nombre Planificador de hoja de Ruta Probabilística (del acrónimo, *Probabilistic Path Planner*), también conocido como el algoritmo Ariadne's Clew.

En Parlangeli y col. 2009, se propone un algoritmo de planificación de trayectorias suaves con curvatura limitada. La trayectoria realiza una aproximación a curvas Dubins, las cuales buscan alcanzar una solución óptima de longitud. El trabajo asume las configuraciones inicio y fin, información que incluye la ubicación espacial 2D y las orientaciones (ángulos) de partida y de llegada.

El artículo presentado en Lau, Sprunk y Burgard 2009, inicia con la determinación de una trayectoria generada por un planificador global. Partiendo de los resultados anteriores, se construye una curva Bézier la cual es utilizada con el objetivo de generar una trayectoria que posea continuidad en curvatura. La curva se construye a partir de segmentos de curva formados por grupos de 3 P_i contiguos, sobre los cuales se insertan 2 P_i adicionales, con lo que cada segmento termina con un total de 5 P_i. Así, el objetivo final al crear segmentos con 5 P_i, es optimizar la curva para generar movimiento continuo y suave en la trayectoria.

En Kim y Kim 2011 se busca alcanzar una minimización de tiempo en planificación de trayectoria, sobre un vehículo omnidireccional de tres ruedas. La aproximación se resuelve en base al modelo cinemático del vehículo junto con los actuadores (motores) y las entradas de control provenientes de las baterías. De acuerdo al límite de la curva alcanzable por el vehículo, se genera una curva basada en Clotoides. En consecuencia, y debido a la complejidad del modelo dinámico del AV (debido al análisis de optimalidad en tiempo) los autores se han centrado en un análisis sobre una trayectoria previamente definida.

Finalmente, pero más importante, los investigadores Girbés, Armesto y Tornero 2014, proponen la generación de trayectorias de doble curvatura continua (DCC) para AVs de ruedas. Así, un controlador híbrido en bucle cerrado se encarga del control cinemático, el cual además determina los valores máximos de curvatura y aceleración del AV. El enfoque de trayectoria se basa en el uso de clotoides como curvas de transición debido a su relación explicita con la brusquedad de movimiento y, por lo tanto, relacionadas con la seguridad.

2.6.2 Transición a Trayectorias 3D

El estudio literario relacionado con la temática de esta investigación muestra que existen diversos enfoques populares que buscan como objetivo alcanzar una trayectoria no-holonómica en 3D. Estos enfoques se centran en la aplicación de métodos que explotan las cualidades de las curvas Bézier, B-spline, Pitagóricas, Dubins, Clotoides, etc., en la generación de trayectorias, planificación de movimientos y construcción de curvas suaves 3D para los AVs, como se describe, a continuación.

Sampaio 2017, se presenta una extensión de las curvas de Bézier para diseñar complejas trayectorias 3D. La solución desarrollada es matemáticamente simple con dos grados de libertad. La metodología presentada realiza una descripción de la trayectoria a través de una sola expresión, que maneja las tres coordenadas espaciales (no trabaja con tres funciones de coordenadas separadas). Finalmente, la formulación para el cálculo de la trayectoria, presenta expresiones para calcular la inclinación, el azimut y la curvatura en cualquier punto de la trayectoria, proporcionando así un control considerable sobre las características de la trayectoria.

En trabajo descrito en McNaughton y col. 2011, presenta un planificador para navegación autónoma en entornos estructurados. El entorno se define como una malla discreta, sobre la cual se aplica un planificador con el objetivo de determinar una trayectoria. Entonces, con una trayectoria definida, se aplica un planificador de navegación autónomo, que se define como un conjunto en el espacio de estados de dimensión 6 ($[x, y, z, \theta, t, v]$, siendo (x, y) el punto de partida, θ el ángulo de movimiento, t el tiempo y v la velocidad de movimiento), el cual se encarga de obtener la trayectoria o conjunto de trayectorias por las cuales se puede navegar. Finalmente, una función de coste determina la mejor trayectoria navegable.

En Subchan y col. 2008 se propone la generación de trayectorias navegables por UAVs, basadas en curvas pitagóricas (PH, del Acrónimo *Pythagorean Hodograph*), con el objetivo de realizar un estudio, detección, modelización y seguimiento de contaminantes aéreos,

En Shanmugavel y col. 2006, se trabaja con un grupo de UAVs generando rutas seguras de vuelo continuo dentro de los límites de curvatura de vuelo definido y a una distancia de separación mínima y que además garantizan una llegada simultánea. Las trayectorias se generan con el objetivo de satisfacer la restricción de curvatura a lo largo de la longitud de la trayectoria (demostrando así que múltiples restricciones de la planificación de trayectoria pueden cumplirse variando solamente la curvatura del mismo). Para lo cual se utilizan curvas PH, explotando el principio de geometría diferencial de que una curva plana está completamente determinada por su curvatura (principio utilizado para cambiar la forma de la trayectoria).

El trabajo presentado en Yang, Zhou y Zhang 2016, propone un método de planificación de trayectoria colaborativa para múltiples UAVs, partiendo de la base de las curvas PH, las cuales se generan en función de las restricciones de tiempo y ángulo de movimiento del UAV. El desarrollo del trabajo estudia la relación entre la longitud de la trayectoria de la curva PH, los parámetros de control, y la restricción del ángulo de vuelo hasta alcanzar el punto objetivo. Todas las condiciones iniciales de la planificación colaborativa de la trayectoria son integradas, de modo que la cooperación en tiempo y ángulo para alcanzar el punto objetivo sea realizable mediante un control de velocidad.

El trabajo de Neto, Macharet y Campos 2013, presenta una generación de trayectoria suave partiendo de una variación de curva PH, calculada a partir de un problema de optimización. El objetivo es modelar un camino como una curva espacial 3D de Bézier de séptimo orden que se calcula de forma iterativa. Esta curva busca maximizar curvatura κ, torsión τ y ángulo de ascenso o descenso, y con ello garantizar maniobras continuas.

En Yang y col. 2015, se propone un método de generación de curva de Bézier de sexto orden (6OBC) entre un conjunto de cuatro nudos denominado árbol aleatorio basado en la atracción guía (GART, (del acrónimo, *Guiding Attraction based Random Tree*), descrito por el mismo Yang y col. 2014b. Como medio para restringir la curvatura, un ángulo de aproximación a la conexión de dos secciones subsiguientes se actualiza iterativamente hasta que la curva se ajusta a la restricción. Este enfoque permite manipular la curva hasta que satisface la restricción y reduce la longitud de la trayectoria generada al suavizar sus bordes. Sin embargo, estos enfoques limitan la trayectoria basándose en la curvatura κ de la curva, que no tiene en cuenta la cinemática en tres dimensiones completas. Para garantizar el cumplimiento de las restricciones, es necesario desacoplar las dimensiones y calcularlas por separado, o bien calcular la curvatura máxima κ_{max} en base únicamente a la menor velocidad de rotación.

El artículo desarrollado por Nikolos y col. 2003, propone un diseño de planificador que calcula una curva suave 3D (sobre un terreno irregular), que es navegable por un UAV. Para la construcción de la curva se utiliza una base de curva B-spline que parte de un conjunto de P_i generados por un algoritmo evolutivo. La trayectoria se construye por medio de la concatenación de segmentos de curvas B-spline.

El trabajo desarrollado en Cai, Zhang y Zheng 2017 está enmarcado en el campo de los AUV, donde se propone una metodología de curva 3D basada en curvas Dubins. En primera instancia se construye una curva Dubins 2D en el plano (x, y), y posteriormente se realiza una interpolación al eje z, con el objetivo de completar la curva 3D, trayectoria que a la vez se construye a partir de las restricciones dinámicas del AUV.

Hernández y col. 2019 trabajan con un AUV y construyen una trayectoria navegable partiendo de los resultados de un algoritmo de planificación de trayectoria RRT. Después, se implementa una curva basada en Dubins, la cual se adecúa a los requerimientos de movimiento del AUV. La algoritmia propuesta funciona cuando el AUV se encuentra en una profundidad constante 2D o cuando el AUV requiere cambiar su profundidad a medida que se desplaza hacia 3D.

Mittal y Deb 2007a parten de un entorno definido como una cuadricula sin restricciones adicionales de movimiento. Entonces, se propone un planificador fuera de línea que minimiza la longitud de la trayectoria y maximiza la seguridad, completando la tarea de planificación de trayectoria a través de una optimización multiobjetivo (el optimizador determina la mejor trayectoria), y su posterior construcción de trayectoria suave 3D en base a una curva B-spline.

El trabajo presentado por Singh y Hota 2017 propone la generación de un planificador de trayectorias para UAVs, partiendo del conocimiento de la posición inicial, posición final y las restricciones cinemáticas de dicho UAV. La metodología analiza una geometría apropiada para generación de trayectorias, que se basa en segmentos rectos y circulares (para los cuales se define el máximo radio de giro horizontal y el máximo radio de giro vertical). Entonces, se construye una curva en el plano (x, y) y posteriomente se construye una nueva curva en el plano (x, z), para finalmente realizar una interpolación de curvas con el objetivo de construir una curva final 3D.

En Wu y col. 2017 se propone un planificador de trayectoria suave para entornos complicados, a partir de curvas PH. El trabajo se divide en dos partes, en la primera parte, se define el entorno de movimiento sobre el que se encuentran definidas irregularidades (como un entorno discretizado y definido como una matriz 2D), elementos representados con diferentes alturas. En la siguiente, por una parte, se genera una trayectoria espacial PH, navegable por un grupo de UAVs, mientras que por otra parte, se satisface las restricciones cinemáticas y dinámicas de los UAVs.

De forma similar, en Yi y col. 2015, se propone un algoritmo de planificación de trayectoria 3D en-línea basado en curvas PH para UAVs. Los autores proponen la construcción de una trayectoria navegable dentro de un entorno dinámico, además, se consideran restricciones cinemáticas de curvatura κ, torsión τ y ángulo de ascenso del UAV. El trabajo adopta algoritmos de estimación de distribución para la optimización y selección de los parámetros de la trayectoria, de tal forma que sean adecuados para resolver el problema. Por otra parte, para la evitación de obstáculos, se utiliza el método denominado velocidad-obstáculo basado en la reducción dimensional. En este sentido, la trayectoria puede volver a planearse dando el punto de inserción de la evitación de obstáculos.

El trabajo desarrollado en Shah y Aouf 2010, presenta una planificación de trayectorias 3D para cooperación en entornos estáticos y obstáculos estacionarios. Las trayectorias se construyen con una base geométrica de curvas PH basadas en las curvas de Bézier. Estas curvas se adaptan para la planificación cooperativa de trayectorias continuas y suaves para varios UAVs que operan simultáneamente en el mismo entorno espacial.

El aporte propuesto en Su y col. 2018, describe un método de planificación de trayectorias en tiempo óptimo sobre un robot paralelo de tipo Delta. La trayectoria se genera a partir de segmentos de transición formados por curvas quíntuples PH, lo cual garantiza que las velocidades, aceleraciones y sacudidas en el espacio sean limitadas, suaves y continuas. Tanto los resultados de la simulación como los experimentos revelan que la planificación de trayectoria propuesta contribuye a la operación en alta velocidad del robot Delta.

En Shao y col. 2019, se propone la generación de un grupo de trayectorias cooperativas navegables para grupos de UAVs, mediante la creación de un algoritmo de optimización cooperativa distribuida de enjambre de partículas

(DCPSO). Esta clase de problemáticas presentan diversas dificultades debido a las restricciones de vuelo. Entonces, con el objetivo de satisfacer las restricciones cinemáticas (restricción de curvatura máxima κ_{max} y el requisito de curvatura continua del trayecto) de los UAVs, se ha adoptado una curva de tipo PH (como una aproximación a una trayectoria parametrizada) debido a su continuidad de curvatura y sus propiedades intrínsecas racionales.

2.7 Conclusiones del capítulo

Con el objetivo de dar inicio al trabajo de investigación propuesto en la tesis, se ha realizado un estudio teórico de curvas en el espacio 2D y 3D, descrito en la sección 2.2, donde se han resaltado las diferentes propiedades de las curvas continuas. En especial se ha realizado un enfoque sobre las ecuaciones de Frenet-Serret en la sección 2.3, ecuaciones que han servido como punto de partida para las propuestas de nuevas curvas suaves 3D presentadas en la tesis.

Una definición y exploración de la Clotoide, también conocida como la espiral de Cornú o espiral de Euler, fue realizada en la sección 2.5, donde además, se han definido sus diferentes símbolos y términos de construcción, así como su formulación fundamental y aplicaciones.

Después de la revisión literaria y el análisis de los resultados provenientes del breve resumen de aportes presentados en las secciones 2.6.1 y 2.6.2, se ha llegado a la conclusión que la mejor opción de partida para el desarrollo del trabajo de tesis son las curvas Clotoides. Debido a sus propiedades geométricas, además de sus alcances, los cuales no han llegado a su punto más alto, es decir, son curvas que aún pueden ser explotadas, maximizando su aplicabilidad y rendimiento, como se mostrará a lo largo del trabajo de tesis.

Capítulo 3

Smooth 3D path planning for non-holonomic UAVs

The first contribution of the thesis work has been presented at an international congress, where a Smooth 3D path planner is proposed, based on Chlotoid 3D curves, which are built by means of Rational Bézier curves. This approach has shown good computational efficiency.

Chapter Note: The contents of this chapter can be found in the paper:

> **Vanegas, G**., Samaniego, F., Girbés-Juan, V., Armesto, L., Garcia-Nieto, S. (2018, October). *Smooth 3D path planning for non-holonomic UAVs.* In 2018 7th International Conference on Systems and Control (ICSC) (pp. 1-6). IEEE.

Abstract: *This paper presents a path planning approach for Unmanned Aerial Vehicles (UAVs) with non-holonomic constraints, such as fixed-wing airplanes. The control scheme is divided in two stages: local planner and kinematic control. In the planning stage, a smooth curve is generated in 3D Euclidean space, using 3D clothoids approximated by Rational Bézier curves. Then, in a second stage, the autonomous flying vehicle is piloted to track references of velocity and orientation. Real flight simulation tests have been carried out to observe the behaviour of the UAV when following 3D clothoids, in order to analyze the properties and benefits of using such smooth curves for path planning of fixed-wing planes.*

3.1 Introduction

The area of Unmanned Aerial Vehicle (UAV) has been evolving drastically in recent years. UAVs were initially conceived for military purposes, but nowadays there is a large number of commercial applications [C. Goerzen y col. 2010; Sujit, Saripalli y Sousa 2014].

When referring to UAVs, it is important to recall this branch is not only about aerial vehicles but about systems which determine achievable trajectories for a correct autonomous navigation. One of the main disadvantages of UAVs is they are non-linear systems and some of them have non-holonomic navigation constraints under normal conditions.

To control these vehicles, space curves are often used to generate smooth trajectories, although most of these curves are not intuitive, do not take vehicle's constraints into account and/or require optimization procedures, which may be not feasible for real-time applications. In this sense, several types of curves have been extensively used, such as Bézier, Spline or Polynomial curves [Kan y col. 2011; Jung y Tsiotras 2013; Wang y col. 2017a], which allow reaching a determined position in the Euclidean space. Geometric analysis in curves context is a continuous map from 1 to n-dimensional space, which its domain has continuous derivatives up to the desired order. This concept is often used in continuous robot navigation besides of ensuring a dynamically feasible and steerable path for those kinds of UAVs.

On the other hand, it is a well-known fact that clothoids provide some interesting geometric, appearance, comfort, and safety aspects which are worth to be used in road, railway and roller coaster design [Meek y Walton 2004a; Marzbani, Jazar y Fard 2015] and to control non-holonomic vehicles [Girbés, Armesto y Tornero 2014; Marzbani y col. 2015b; Gim y col. 2017a]. Unfortunately, there is no closed-form solution to compute Fresnel integrals, although some approximations, with errors up to 10^{-20}, allow real time computation [Mielenz 2000; Montés y col. 2008; Brezak y Petrovic 2014; Narayan 2014; Chen y col. 2017a]. Despite this fact, there is no analytical solution but, it is possible to compute some geometric properties analytically, such as curvature and tangent angle as a function of arc-length.

Thus, this paper focuses on generating smooth 3D trajectories to pilot non-holonomic UAVs, such as fixed-wing airplanes. Due to their interesting properties and smoothness, the aim is to approximate 3D clothoids by Rational-Bézier curves to achieve real-time performance. The control scheme is divided in two stages: *local planner* and *kinematic control*, as depicted in 3.1.

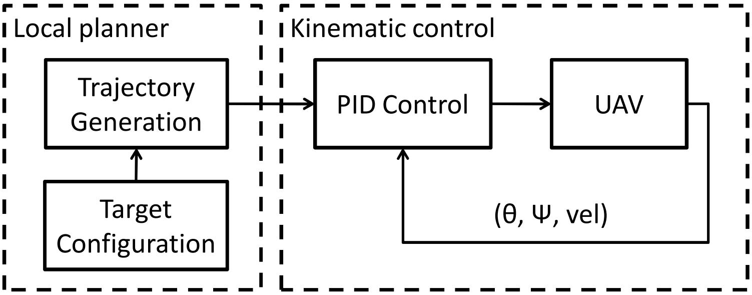


Figura 3.1: Planning and control diagram.

The paper is organized as follows. Section 3.2 introduces some preliminaries about smooth curves, such as clothoids and Bézier curves. A methodology to approximate 3D clothoids by Rational Bézier curves is explained in section 3.3. Then, in section 3.4 some flight simulation tests are performed, whose results are shown and discussed. Finally, in Section 3.5 some conclusions and future work are drawn.

3.2 Smooth curves definition

3.2.1 Space Curves

Any space curve $\mathbf{C}(s)$, parametrized by its arc-length s in a three-dimensional space $\mathbb{R}^3$, is determined by its curvature $\kappa(s)$ and its torsion $\tau(s)$, according to the space curves fundamental theorem [Banchoff y Lovett 2010]. Intuitively, a curve can be obtained from a straight line by bending (curvature) and twisting (torsion). Furthermore, for $|\kappa(s)| > 0$ and $\tau(s)$, there exists a unique space curve defined by its Frenet-Serret equations as follows

$$\begin{bmatrix}\mathbf{T}'(s)\\ \mathbf{N}'(s)\\ \mathbf{B}'(s)\end{bmatrix} = \begin{bmatrix}0 & \kappa(s) & 0\\ -\kappa(s) & 0 & \tau(s)\\ 0 & -\tau(s) & 0\end{bmatrix}\begin{bmatrix}\mathbf{T}(s)\\ \mathbf{N}(s)\\ \mathbf{B}(s)\end{bmatrix} \tag{3.1}$$

being $\mathbf{T}(s)$, $\mathbf{N}(s)$ and $\mathbf{B}(s)$ the tangent, normal and binormal vectors of a right-handed unit orthogonal system, respectively. While $\mathbf{T}'(s) = d\mathbf{T}(s)/ds$, $\mathbf{N}'(s)=d\mathbf{N}(s)/ds$, $\mathbf{B}'(s)=d\mathbf{B}(s)/ds$ are the first derivatives of such vectors.

Thus, this orthogonal system is defined as $\mathbf{R}(s) := [\mathbf{T}(s)\ \mathbf{N}(s)\ \mathbf{B}(s)]$ and it can be integrated from (3.1) based on a specific pair of functions of curvature $\kappa(s)$ and torsion $\tau(s)$, with a given initial value $\mathbf{R}(0):=[\mathbf{T}(0)\ \mathbf{N}(0)\ \mathbf{B}(0)]$. The position is obtained by integrating the tangent vector:

$$\mathbf{C}(s) := \mathbf{C}(0) + \int_0^s \mathbf{T}(\xi)\, d\xi \tag{3.2}$$

On the other hand, based on a curve $\mathbf{C}(s)$, we can obtain the tangent $\mathbf{T}(s)$, normal $\mathbf{N}(s)$ and binormal $\mathbf{B}(s)$ vectors. Hence, taking into account the theory of differential geometry of curves [Banchoff y Lovett 2010], the following equalities for the mobile trihedron of any curve $\mathbf{C}(s)$ are defined

$$\mathbf{T}(s) := \frac{\mathbf{C}'(s)}{\|\mathbf{C}'(s)\|} \tag{3.3}$$

$$\mathbf{N}(s) := \frac{\mathbf{T}'(s)}{\|\mathbf{T}'(s)\|} = \frac{[\mathbf{C}'(s) \times \mathbf{C}''(s)] \times \mathbf{C}'(s)}{\|[\mathbf{C}'(s) \times \mathbf{C}''(s)] \times \mathbf{C}'(s)\|} \tag{3.4}$$

$$\mathbf{B}(s) := \mathbf{T}(s) \times \mathbf{N}(s) = \frac{\mathbf{C}'(s) \times \mathbf{C}''(s)}{\|\mathbf{C}'(s) \times \mathbf{C}''(s)\|} \tag{3.5}$$

being $\mathbf{C}'(s) = d\mathbf{C}(s)/ds$, $\mathbf{C}''(s) = d^2\mathbf{C}(s)/ds^2$, $\mathbf{C}'''(s) = d^3\mathbf{C}(s)/ds^3$ the derivatives of position vector $\mathbf{C}(s)$.

3.2.2 Planar 2D clothoids

An Euler spiral or planar clothoid is defined in $\mathbb{R}^2$ as the curve whose curvature varies linearly with respect to arc-length

$$\kappa(s) := \sigma_\kappa s, \tag{3.6}$$

where $\sigma_\kappa := d\kappa(s)/ds$ is the clothoid's sharpness, which is related to its homothety factor or scale K by $\sigma_\kappa := \pi/K^2$ and clothoid's tangent angle is

$$\beta(s) := \frac{\sigma_\kappa}{2} s^2, \tag{3.7}$$

In addition to this, the tangent vector, can be expressed as

$$\mathbf{T}(s) := \begin{bmatrix} \cos(\beta(s)) \\ \sin(\beta(s)) \end{bmatrix} \tag{3.8}$$

From equations (3.2), (3.7) and (3.8), the well-known equations for a planar clothoid (contained in plane XY) [Levien 2008a], noted as $\mathbf{C}(s)$, can be derived from Fresnel integrals as follows:

$$\mathbf{C}(s) = \begin{bmatrix} \mathcal{C}(s) \\ \mathcal{S}(s) \end{bmatrix} := \begin{bmatrix} \int_0^s \cos(\frac{\sigma_\kappa}{2}\xi^2)\, d\xi \\ \int_0^s \sin(\frac{\sigma_\kappa}{2}\xi^2)\, d\xi \end{bmatrix} \tag{3.9}$$

being $\mathcal{C}(s)$ and $\mathcal{S}(s)$ the Fresnel's cosine and sine integrals. Note that without loss of generality, the clothoid is assumed to be origin-centered, that is $\mathbf{C}(0) = \mathbf{0}$ in (3.2).

The authors Harary y Tal 2012a introduced 3D Euler spiral or clothoid (C3D), defined in $\mathbb{R}^3$, as a curve whose curvature varies as in (3.6) and its torsion varies as:

$$\tau(s) := \sigma_\tau s \tag{3.10}$$

where $\sigma_\tau := \delta\tau/\delta s$ its first geometric derivative or torsion sharpness.

The main drawback of 3D clothoids, as presented in Harary y Tal 2012a requires numerical integration to be solved. More specifically, to reach a 3D position or orientation, the 3D clothoid parameters must be optimized to the clothoid satisfies the constraints, which is time-consuming and not suitable for real-time operations.

3.2.3 Bézier Curves

Bézier curves are a mathematical description method of curves numerically stable [Gordon y Riesenfeld 1974b; Segovia y col. 1991]. A Bézier curve (B-curve) is a special case of a spline curve, in which the curve has only one polynomial component in the piecewise. The Bernstein polynomials of degree n (with $n > 0$) can also be interpreted as the B-spline basis functions of degree n over the domain of those polynomials.

The Bézier curve of degree n can be expressed as

$$\begin{aligned}\mathbf{B}_B(s, \mathbf{P}) &= \sum_{i=0}^{n} \binom{n}{i} \mathbf{P}_i \left(\frac{s_f - s}{s_f}\right)^{n-i} \left(\frac{s}{s_f}\right)^{i} \\ &= \mathbf{P}_0(1-t)^n + \binom{n}{1}^{n-1} t + \cdots + \mathbf{P}_n t^n, \\ & t \in [0,1]\end{aligned} \tag{3.11}$$

where $\mathbf{P} := [\mathbf{P}_0^T, \mathbf{P}_1^T, \ldots, \mathbf{P}_n^T] \in \mathbb{R}^{3(n+1)}$ are the set of control points, s_f is the curve's length and $\mathbb{R}^3$ is the 3D space where the curve is defined. In other words, the degree n of the Bézier curve is an interpolation between the $(n+1)$ 3D *control points*. Bézier curves can be expressed in terms of Bernstein polynomials of degree n:

$$\mathbf{B}_B(s, \mathbf{P}) = \sum_{i=0}^{n} b_{i,n}(s) \mathbf{P}_i, \; s \in [0, s_f] \tag{3.12}$$

where the polynomials $b_{i,n}(s) \in \mathbb{R}$

$$b_{i,n}(s) := \binom{n}{i} \left(\frac{s_f - s}{s_f}\right)^{n-i} \left(\frac{s}{s_f}\right)^{i}, \; i = 0, \ldots, n \tag{3.13}$$

The main advantage of Bézier curves with respect to clothoids is that they have scaling and rotation properties [Montes, Mora y Tornero 2007]. Thus, we can fit *control points* of a unitary Bézier and then used them to generated a huge variety of curves due to their scaling and rotation properties. This is one of the main motivations behind this preliminary paper, in which we aim to fit a 3D Clothoid curve using a Bézier curve.

3.3 Heading-Attitude Control

Heading and attitude control in UAVs is a very important problem to solve [Adiprawita, Suwandi Ahmad y Sembiring 2012]. This is particularly relevant in airplanes (fixed-wing flying vehicles), because they have non-holonomic constraints and therefore heading and attitude control is the "natural" choice to pilot a plane.

3.3.1 Smooth Trajectory Generation

Press, William H., Teukolsky, Saul A., Vetterling, William T., and Flannery 2007, have used an optimization procedure using the Gradient-descent algorithm to generate a curve joining two arbitrary point-vector configurations, to get the optimum values of initial curvature κ_0 and torsion τ_0, as well as sharpness of curvature σ_κ and torsion σ_τ. Therefore, their solution could take any initial and final values of curvature and torsion. However, when planning curves for non-holonomic vehicles such as aerial or underwater robots moving on a 3D space, due to their non-holonomic constraints, such vehicles cannot follow arbitrary curves with instantaneous changes in either, curvature, torsion or orientation. Therefore, in many practical cases, the initial curvature and torsion are given, based on current's robot curvature and torsion. In that case, using the solution proposed in [Harary y Tal 2012a] only the final position or final orientation can be achieved, but not both simultaneously.

In this paper, a curve parameterized by a generic parameter vector $\mathbf{p}$ is proposed, noted as $\mathbf{C}(s, \mathbf{p})$. Actually, the parameter vector, will be $\mathbf{p} = (\sigma_\kappa, \sigma_\tau)$. The aim is to compute $\mathbf{p}^*$ such as $\mathbf{T}(s, \mathbf{p}) = \mathbf{T}^*$ for a given $s > 0$. Without loss of generality it can be assumed that the local frame of the curve is coincident to the global coordinate frame, which means that the start point is $\mathbf{C}(0, \mathbf{p}) = [0, 0, 0]^T$.

Using the chlotoid properties described by Girbes, Armesto y Tornero 2011, we can provide an initial solution for the optimization procedure. The idea is to individually compute the sharpness required by two clothoids on the XY-plane and YZ-plane so that the sharpness of curvature and torsion can be obtained, separately, for a given pitch angle $\theta(s)$ and yaw angle $\psi(s)$.

$$\gamma_\tau = \sqrt{\frac{|\theta(s)|}{\pi/2}},\ \gamma_\kappa = \sqrt{\frac{|\psi(s)|}{\pi/2}} \tag{3.14}$$

$$\sigma_{\kappa_0} = \text{sign}(\theta(s))\frac{\pi\gamma_\kappa^2}{s^2} \tag{3.15}$$

$$\sigma_{\tau_0} = \text{sign}(\psi(s))\frac{\pi\gamma_\tau^2}{s^2} \tag{3.16}$$

Therefore, the challenge to be solved can be stated as an orientation minimization problem with initial guess $\mathbf{p}_0 = (\sigma_{\kappa_0}, \sigma_{\tau_0})$:

$$\mathbf{p}^* = \arg_{\mathbf{p}} \text{mín} \left\|\mathbf{T}^* - \mathbf{T}(s, \mathbf{p})\right\| \tag{3.17}$$

3.3.2 3D Clothoid Approximation

Now, we describe a method to approximate 3D Clothoids using Rational Bézier curves (RB-curves) with a given number of points. We extend the formulation provided in [Montes, Mora y Tornero 2007] by considering independent weights on each control point of a RB-curve. RB-curves can be expressed as:

$$\mathbf{B}_{RB}(s, \mathbf{P}, \mathbf{W}) := \left[\sum\nolimits_{i=0}^{n} b_{i,n}(s)\mathbf{W}_i\right]^{-1} \sum\nolimits_{i=0}^{n} b_{i,n}(s)\mathbf{W}_i\mathbf{P}_i, \tag{3.18}$$

where $\mathbf{W}_i \in \mathbb{R}^3$ is a diagonal of a weight matrix. The parameters to be learned are the position of the *control points* and the weights. Since they have a non-linear relation, we propose to learn them in a two-step method: first learn the position of the *control points* assuming $\mathbf{W}_i = \mathbf{I}$, which corresponds to the expression of a conventional Bézier curve; then, the weights of the RB-curve are learned considering that the *control points* are fixed, to refine the estimation, by including a greater flexibility to fit the original 3D clothoid curve. In order to estimate the conventional Bézier curve, we will set the position of the first and last control points to the curve initial and final values, that is, $\mathbf{P}_0 = \mathbf{C}(0, \mathbf{p})$ and $\mathbf{P}_n = \mathbf{C}(s_f, \mathbf{p})$, being s_f the overall curve's length. Since we assume $\mathbf{W}_i = \mathbf{I}$ the curve points can be obtained via Least-squares fitting $\mathbf{Y} = \mathbf{X}\tilde{\mathbf{P}}$ as:

$$\mathbf{Y} := \begin{bmatrix} \mathbf{C}(0,\mathbf{p}) - \frac{b_{0,n}(0)}{\sum_i^n b_{i,n}(0)}\mathbf{P}_0 - \frac{b_{n,n}(0)}{\sum_i^n b_{i,n}(0)}\mathbf{P}_n \\ \vdots \\ \mathbf{C}(s_f,\mathbf{p}) - \frac{b_{0,n}(1)}{\sum_i^n b_{i,n}(1)}\mathbf{P}_0 - \frac{b_{n,n}(1)}{\sum_i^n b_{i,n}(1)}\mathbf{P}_n \end{bmatrix} \tag{3.19}$$

$$\mathbf{X} := \begin{bmatrix} \frac{b_{1,n}(0)}{\sum_i^n b_{i,n}(0)} & \cdots & \frac{b_{n-1,n}(0)}{\sum_i^n b_{i,n}(0)} \\ \vdots & \ddots & \vdots \\ \frac{b_{1,n}(0)}{\sum_i^n b_{i,n}(1)} & \cdots & \frac{b_{n-1,n}(0)}{\sum_i^n b_{i,n}(1)} \end{bmatrix} \tag{3.20}$$

$$\tilde{\mathbf{P}} := \begin{bmatrix} \mathbf{P}_1 \\ \vdots \\ \mathbf{P}_{n-1} \end{bmatrix} \tag{3.21}$$

With the estimated position of the control points, we can now estimate the values of the weights for each coordinate. First we need to express the equality $\mathbf{C}(s,\mathbf{p}) = \mathbf{B}_{RB}(s,\mathbf{P},\mathbf{W})$ in terms of the weights as:

$$\sum_{i=0}^{n} b_{i,n}(s)\mathbf{C}(s,\mathbf{p})\mathbf{W}_i = \sum_{i=0}^{n} b_{i,n}(s)\mathbf{P}_i\mathbf{W}_i, \tag{3.22}$$

considering that $\mathbf{W}_0 = \mathbf{I}$ and $\mathbf{W}_n = \mathbf{I}$ in order to ensure that the estimated curve starts and ends at the initial and final position of the original curve, we have the following linear equation $\hat{\mathbf{Y}} = \hat{\mathbf{X}}\hat{\mathbf{W}}$ that can be solved via Least Squares “LS”:

$$\hat{\mathbf{Y}} := \begin{bmatrix} b_{0,n}(0)\hat{\mathbf{P}}_0(0) + b_{n,n}(0)\hat{\mathbf{P}}_n(0) \\ \vdots \\ b_{0,n}(s_f)\hat{\mathbf{P}}_0(s_f) + b_{n,n}(s_f)\hat{\mathbf{P}}_n(s_f) \end{bmatrix} \tag{3.23}$$

$$\hat{\mathbf{X}} := \begin{bmatrix} -b_{1,n}(0)\hat{\mathbf{P}}_1(0) & \ldots & b_{n-1,n}(0)\hat{\mathbf{P}}_{n-1}(0) \\ \vdots & \ddots & \vdots \\ -b_{1,n}(s_f)\hat{\mathbf{P}}_1(s_f) & \ldots & b_{n-1,n}(s_f)\hat{\mathbf{P}}_{n-1}(s_f) \end{bmatrix} \tag{3.24}$$

$$\hat{\mathbf{W}} := \begin{bmatrix} \mathbf{W}_1^T & \ldots & \mathbf{W}_{n-1}^T \end{bmatrix}^T \tag{3.25}$$

where $\hat{\mathbf{P}}_i(s) := (\mathbf{C}(s,\mathbf{p}) - \mathbf{P}_i)$ and abusing of notation $\hat{\mathbf{W}}$ is indeed a vectorized version of it when solving the LS problem.

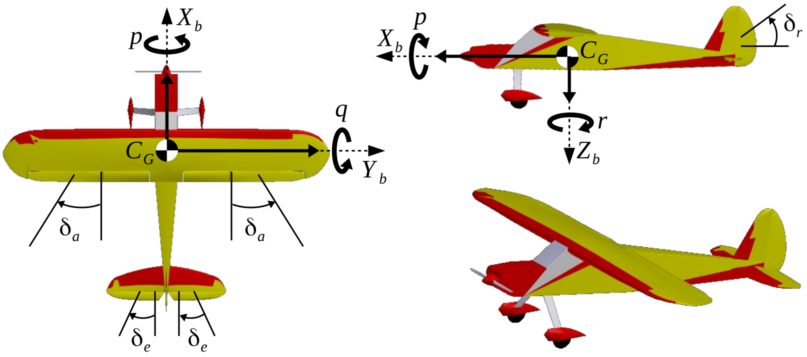


Figura 3.2: Kadett UAV model and variables definition.

3.4 Experiments and Results

The experimentation has been carried out using a flight simulator *FlightGear 2018*, the mathematical software tool *Matlab R2017b* and a computer with Intel Core i7-6700HQ 2.60GHz processor and 16GiB DDR4 RAM memory. The airplane model used for the dynamic simulations is based on *kadett 2400* [Velasco-Carrau y col. 2015] (depicted in Figure 3.2).

The systems has 4 inputs (see Figure 3.3) to control the aerodynamic surfaces and speed: δ_e (elevator), δ_a (aileron), δ_r (rudder) and δ_{th} (throttle). As any vehicle moving in a 3D Cartesian space, it can be represented by 6 states $\{x, y, z, \varphi, \theta, \psi\}$, where the first three states define the position vector of the center of gravity C_G w.r.t. a Global Coordinate System CS_g placed at the origin. The latter three are the Euler angles roll, pitch and yaw, respectively, which define the orientation of body's local coordinate system CS_b w.r.t. CS_g. Notice that CS_b is defined by 3 unitary orthogonal vectors $\{X_b, Y_b, Z_b\}$ aligned with the three axis of the vehicle and centered at C_G, with Z_b pointing downwards, as can be seen in Figure 3.2. Angular velocities along axis X_b ,Y_b and Z_b are represented by p, q and r, respectively, see [Velasco-Carrau y col. 2015] for details.

Figure 3.3 shows the control structure used to drive the aircraft. Given some initial conditions (current tangent vector $\mathbf{T}$) and a target configuration (final tangent vector $\mathbf{T}^*$), a reference trajectory is generated. The goal is to keep a constant velocity vel along the trajectory, which is accomplished by a PID controller that affects the throttle. Besides, the profiles θ and ψ of the generated trajectory are used as a reference input for two more PID controllers. The first PID controls the elevator, which affects the pitch orientation. Whilst the output of the second PID is multiplied by a gain K to control the aileron, which affects the roll orientation, but also makes the aircraft turn and therefore affects the yaw orientation. The control action is also applied to the rudder, which affects slightly the yaw orientation and is used to compensate little errors when tracking the reference yaw ψ.

As explained in Section 3.3.2, a 3D clothoid (C3D) can be approximated by a Rational Bézier curve (RBC). The main drawback of using C3D is that these curves are computed using the Fresnel integrals, which for a small discretization step and long distances (upper integration limit) can be very time-consuming. Besides, another problem is the fact that the final result depends on the selection of the discretization step. So, for the same sharpness of curvature σ_κ and torsion σ_τ the generated path and final orientation will differ depending on such integration step. On the contrary, a RBC is parametrized by n control points and weights, depending on the order n, and can be discretized using any step producing always the same curve. Another drawback of using C3D curves directly is that to solve the planning problem they use an optimization algorithm to find the values $\sigma^*_{\kappa_0}$ and $\sigma^*_{\tau_0}$ to reach a given final tangent vector $\mathbf{T}^*$, for a particular integration step. This process is in fact highly time-consuming and is not appropriate for real-time planning and control of flying vehicles.

For the simulations performed in this section, different cases of C3D has been approximated by a RBC, which different orders were have been used $n = \{11 - 15\}$, we found that $n = 13$ was the best option taking into account accuracy and computation time, since $n > 13$ order of approach results in an overlap of regressors, depicted in Table 3.1.

Cuadro 3.1: Error: RCB approximation with different order.

Control points	**11**	**12**	**13**	**14**	**15**
$\log_{10}(e)$	-7.3511	-8.2492	-9.0989	-7.9992	-7.4833

For this particular case we generate a C3D using 10^4 discrete points. The curvature sharpness is set to $\sigma_\kappa = 1{,}796$ rad/m^2, the torsion sharpness is $\sigma_\tau = 1{,}983$ rad/m^2, the overall length is $l = 1$ m, with final angles $\theta = -\pi/8$ rad and $\psi = \pi/4$ rad. The average computation time for this particular case is $t_c = 15{,}97$ ms, whilst the approximation error integrated along the path is $e = 7{,}9635 \cdot 10^{-10}$ m. Figure 3.4 shows the C3D (blue) and the approximated RBC (dashed red), as well as the control points (green circles).

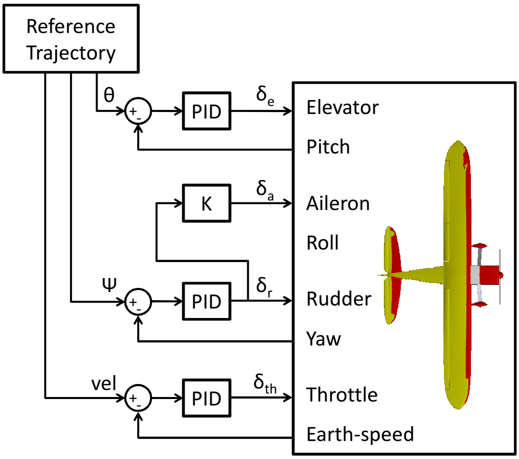


Figura 3.3: Scheme used to control the Kadett UAV model.

Some results of the experiments carried out in FlightGear simulator and Matlab are shown in Figure 3.5. The reference velocity has been set to $vel = 18$ m/s, with a sampling period of $T_s = 20$ ms to control the aircraft. Since the length of the reference trajectory is set to $l = 180$ m, the simulation time is $t = 10$ s.

In Figure 3.5(a) it can be observed that the aircraft (red) follows very closely the reference trajectory (blue), even when only pitch and yaw references have been tracked by the UAV. On the other hand, Figure 3.5(b) shows the tracking errors for some variables. It can be observed that the tracking error of angles pitch and yaw is very slow. This is mainly due to the fact that the RBC is an approximation of a C3D, which has the property of smoothness that allows an easy tracking. Regarding the velocity, the error is very low throughout almost the whole trajectory. However, in the last part (around $l = 180$ m), due to the fact that the pitch angle is high, the UAV has some difficulties to follow the reference velocity and the error e_v increases. As a consequence, the distance e_d between vehicle and path also increases considerably in the second half of the experiment.

Finally, Figure 3.6 depicts a sequence of screen-shots of the simulation at different time instants $t = \{0, 2, 4, 6, 8, 10\}$ s. It can be seen that the UAV starts at $t = 0$ s with height $h = 0$, roll $\varphi = 0$, pitch $\theta = 0$ and yaw $\psi = 0$. Then, it starts to rotate in roll, which makes the airplane turn to the left, while progressively increasing its pitch angle and therefore its height ($t = \{2 - 8\}$ s). The last image shows the UAV in the last configuration at $t = 10$ s, with pitch $\theta = -\pi/8$ rad and yaw $\psi = \pi/4$ rad.

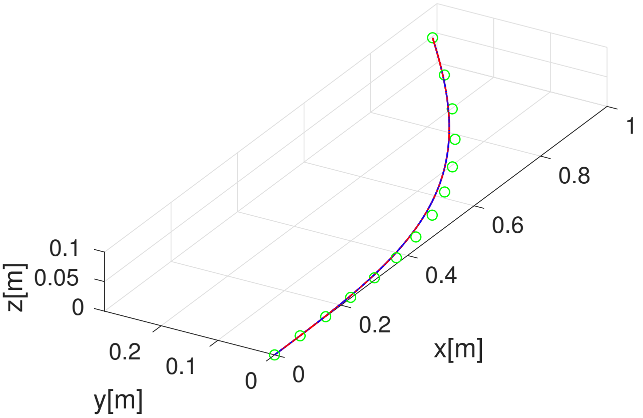


Figura 3.4: Approximation of C3D to RBC: final orientation $\theta^* = -\pi/8$ and $\psi^* = \pi/4$.

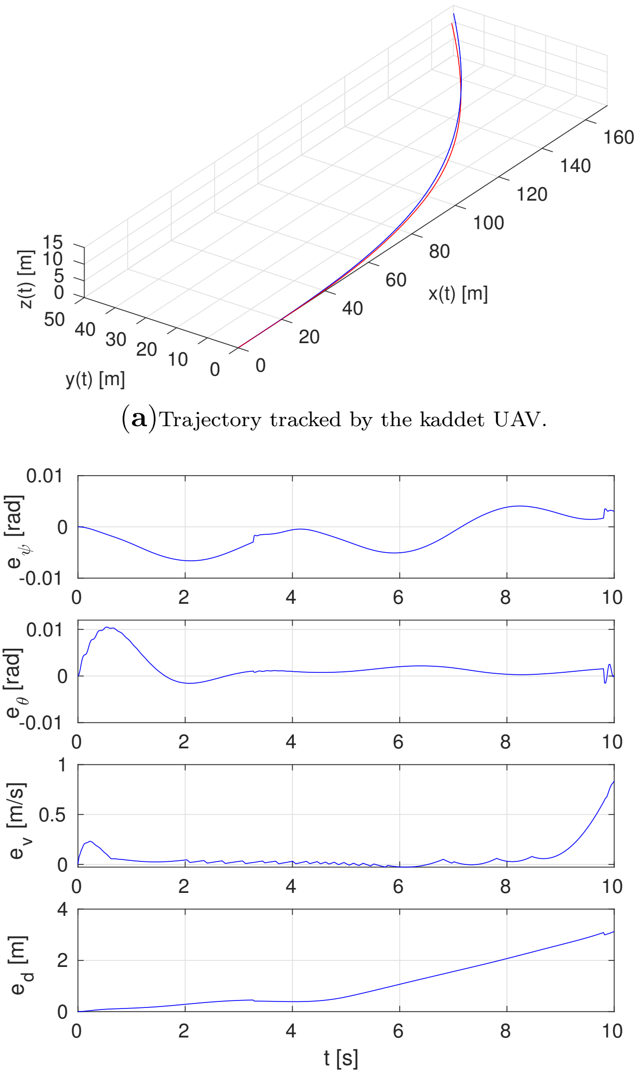
3.5 Conclusions

A smooth path planner for fixed-wing airplanes has been introduced. The control scheme is divided in two stages: local planner and kinematic control. In the planning stage, 3D clothoids are approximated by Rational Bézier curves in order to generate smooth trajectories in 3D Euclidean space. Then, this trajectory is used as a reference to pilot autonomously the UAV in order to track references of velocity and orientation.

Using RBC of order 13 to approximate 3D clothoids has been proven to be efficient computationally (around 16 ms to compute 10^4 discrete points) and with high accuracy (approximation error of around $e = 8 \cdot 10^{-10}$ m along a trajectory with length 1 m.

Flight simulation tests have been performed using the FlightGear simulator and Matlab. After analyzing the results, some interesting benefits of using 3D clothoids for path planning have been observed. A non-holonomic vehicle following such smooth trajectories is able to track easily the reference of pitch and yaw angles. In this sense, even though the position reference is not taken into account, the results have shown that a non-holonomic UAV is able to trace a trajectory close to the one designed by the path planner.

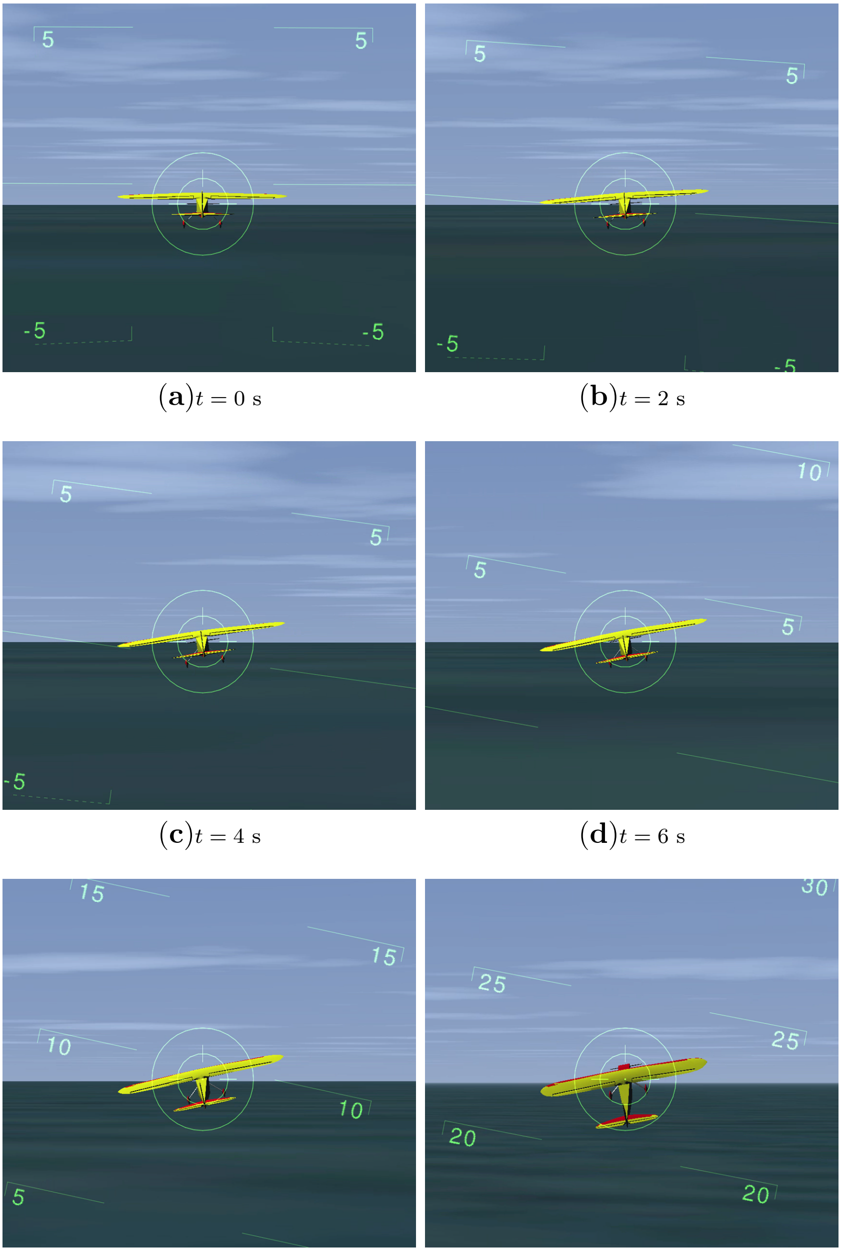
As further work, we aim to use machine learning techniques in order to generate a family of Rational Bézier curves approximating unitary 3D clothoids in a very efficient way. So, the control points and weights of the RBC will be learned as a function of the final orientation, described by a target tangent vector in 3D space.



(**a**)Trajectory tracked by the kaddet UAV.

(**b**)Tracking errors in roll, pitch and yaw movements.

Figura 3.5: (a) Trajectory traced by the UAV (dashed red) following a C3D approximated by a RBC (blue) with length $l = 180$ m and final orientation $\theta^* = -\pi/8$ rad and $\psi^* = \pi/4$ rad. (b) Tracking error of angles yaw and pitch, velocity and position distance to the reference trajectory.



(a)$t = 0$ s

(b)$t = 2$ s

(c)$t = 4$ s

(d)$t = 6$ s

(e)$t = 8$ s

(f)$t = 10$ s

Figura 3.6: Some screen shots of the simulation in FlightGear.

3.6 Acknowledgment

The authors are grateful to the financial support of Spanish Ministry of Economy and European Union, grant DPI2016-81002-R (AEI/FEDER, UE). This work was also supported by the postdoctoral fellowship “APOSTD/2017/055” and the local administration “GV/2017/029” (Generalitat Valenciana, Conselleria d’Educació) Valencia - Spain.

Capítulo 4

A clothoid-based three-dimensional curve for attitude planning

This chapter describes the second contribution of the thesis, where a new analytical solution in the building of clothoid curves has been presented. As a result of this first study, interesting properties of the curve have been proven, properties that have given rise to the following contributions. The applications of the new clothoid curve (clothoid based 3D curve) are extensive, nevertheless and given the aims of the thesis, the new curve has been exploited in order to use it to solve the Smooth 3D Path Planning problem.

Chapter Note: The contents of this chapter can be found in the paper:

> Girbés, V., **Vanegas, G.**, & Armesto, L. (2019). *Clothoid-Based Three-Dimensional Curve for Attitude Planning.* Journal of Guidance, Control, and Dynamics, 42(8),1886-1898., JCR: Q1, Impact Factor: 2.024, SJR: Q1, Impact Factor: 1.57, DOI: https://doi.org/10.2514/1.G003551.

4.1 Introduction

Interest in flying robots, also known as unmanned aerial vehicles (UAVs), has grown during last years in both military and civil fields [C. Goerzen y col. 2010; Sujit, Saripalli y Sousa 2014]. The same happens to autonomous underwater vehicles (AUVs) [Zeng y col. 2015]. These vehicles, UAVs and AUVs, offer a wide variety of possible applications and challenges, such as control, guidance or navigation [Sujit, Saripalli y Sousa 2014; Zeng y col. 2015].

In this sense, heading and attitude control in UAVs is very important [Adiprawita, Suwandi Ahmad y Sembiring 2012], particularly relevant in airplanes (fixed-wing flying vehicles), because they are strongly non-linear, coupled, and tend to be underactuated systems with non-holonomic constraints. Hence, designing a good attitude controller is a difficult task [Zhai y col. 2014; Rubio Hervas y col. 2016; Bhandari y col. 2016; Nguen, Putov y Nguen 2017; Prabhakar y col. 2018], where stability must be taken into account by the controller [Eren y col. 2017]. Indeed, if the reference is too demanding for the controller or non-achievable because its dynamics is too fast, the vehicle might become unstable. In order to address this issue, autonomous navigation systems usually include a high-level path planner to generate smooth reference trajectories to be followed by the vehicle using a low-level controller. Usually a set of waypoints is given in GPS coordinates, normally from a map, in order to apply a smooth point-to-point control trajectory [Liu, Liu y Wei 2014; Liang, Zhong y Chunhui 2015].

There are plenty of space curves commonly used for UAV and AUV navigation, such as Bézier, B-spline, NURBS or other parametric curves [Kan y col. 2011; Jung y Tsiotras 2013; Choe y col. 2016; Dilectis, Mortari y Zanetti 2016; Wang y col. 2017a]. But none of them are intuitive for attitude planning, because they are very sensitive to parameter initialization (control points and weights) [Pérez y col. 2018]. In addition to this, they do not take vehicle's constraints into account and/or require optimization procedures, which may be not feasible for real-time applications. For all these reasons, these parametric curves (polynomial and rational) are out of the scope of the paper and, therefore, are not used in our experimentation. Another drawback of these types of parametric curves is the absence of physical meaning, and it might be unclear to get the relationship between their design parameters and system's dynamic variables. In other words, vehicle's constraints or control limitations (maximum input/output values) cannot be taken into account *a priori* in the trajectory generation, because it is difficult or not possible to do so. For that reason, other solutions propose the use of kinematic and dynamic models to generate a trajectory [Mittal y Deb 2007b; Hota y Ghose 2010b; Rikovitch y Sharf 2013; Ataei y Yousefi-Koma 2015], although optimization procedures are required to minimize/maximize functions such as time, distance, curvature, velocity, acceleration, jerk, etc. [Gonzalez y col. 2016].

In two-dimensional (2D) planning, most authors concatenate straight lines, circular arcs, and some transition curves [Shin y Singh 1990; Delingette, Hebert y Lkeuchi 1991; Tounsi y Le˜Corre 1996; amd Alonzo Kelly 2001]. Among all proposed 2D curves in the bibliography, clothoids (also known as Euler or Cornu spirals [Levien 2008a]) play an important role, since it is proven to be the shortest path [Fraichard y Scheuer 2004a; Girbés, Armesto y Tornero 2014]. Besides, clothoids have “nice” geometric properties, including a close relation between their scaling parameter and physical phenomena (angular velocity, normal acceleration and jerk). Clothoids have been studied in depth to control non-holonomic ground mobile robots [Fraichard y Scheuer 2004a; Armesto y col. 2012; Girbés, Armesto y Tornero 2014; Brezak y Petrovic 2014], proving that clothoid’s parameters have a direct effect in safety and comfort.

In order to guarantee C^2-continuity in three-dimensional (3D) planning, the easiest and most intuitive approach is setting null curvature and torsion in the joining points, as done in 2D planning [Fraichard y Scheuer 2004a; Girbés, Armesto y Tornero 2014]. This way, straight segments and turnings (made of transition curves) can be easily concatenated, while guaranteeing curvature and torsion continuity. Thus, borrowing the idea from planar movement, space clothoids can also be used in path-planning and navigation applications. However, even though the benefits of planar clothoids have been proven, researchers have paid little attention to the use of such curves in 3D space. A few works proposed the use of clothoids in path-planning for UAVs [Yang y Sukkarieh 2008a; Wan y col. 2011; Wilburn, Perhinschi y Wilburn 2013]. Others instead, approached a similar problem from the fields of road design [Kommer y Weidner 2007] and computer graphics [Harary y Tal 2012a; Bertails-Descoubes 2012; Çelik, Semra S.; Yayli, Yusuf; Güler 2016].

Nevertheless, only [Harary y Tal 2012a] seems to match the exact definition of a 3D clothoid (C3D), which is a curve whose curvature and torsion evolve linearly with respect to arc-length. But, the problem is that this approach is based on a Frenet-Serret frame [Banchoff y Lovett 2010] and requires both a numerical integration and an optimization procedure [Harary y Tal 2012a]. Moreover, it is not possible to know *a priori* the increment in position and orientation with respect to curve’s origin.

In particular, the paper focuses on the problem of attitude planning, where an aircraft flying straight needs to perform a smooth maneuver to change its direction and height, for instance to avoid an imminent collision. In fact, we seek for a curve that can be used in a local planning framework as smooth reference trajectory for obstacle avoidance of fixed-wing planes and other UAVs, as well as for planning more complex maneuvers, combining straight lines and other three-dimensional curves to generate smooth paths in a global planning framework. In this paper we present a novel clothoid-based three-dimensional curve (coined as Cb3D), which has an analytical solution, allows real-time computation for most of applications and provides some *interesting* properties for path-planning problems, such as scalability, symmetry, smoothness, monotonicity, etc. The integration of this curve within either local or global planning algorithms is out of the scope of this paper.

The paper is organized as follows. Section 4.2 introduces some preliminaries about differential geometry of space curves, as well as 2D and 3D clothoids. The problem statement is explained in this section too. The proposed Cb3D curve is defined in Section 4.3, where some lemmas and properties are also proved in Appendix A. In order to evaluate the proposed Cb3D curve against a C3D, a comparative analysis is performed in Section 4.4, with some simulations using FlightGear in Section 4.5. Finally, the paper ends with a discussion of results in Section 4.6.

4.2 Preliminaries

4.2.1 Space Curves

Any space curve $\mathbf{C}(s)$, parametrized by its arc-length s in a three-dimensional space $\mathbb{R}^3$, is determined by its curvature $\kappa(s)$ and its torsion $\tau(s)$, according to the fundamental theorem of space curves [Banchoff y Lovett 2010]. Intuitively, a curve can be obtained from a straight line by bending (curvature) and twisting (torsion). So, for $|\kappa(s)| > 0$ and torsion $\tau(s)$, there exists a unique space curve defined by its Frenet-Serret equations as follows

$$\begin{bmatrix} \mathbf{T}'(s) \\ \mathbf{N}'(s) \\ \mathbf{B}'(s) \end{bmatrix} = \begin{bmatrix} 0 & \kappa(s) & 0 \\ -\kappa(s) & 0 & \tau(s) \\ 0 & -\tau(s) & 0 \end{bmatrix} \begin{bmatrix} \mathbf{T}(s) \\ \mathbf{N}(s) \\ \mathbf{B}(s) \end{bmatrix} \tag{4.1}$$

being $\mathbf{T}(s)$, $\mathbf{N}(s)$, and $\mathbf{B}(s)$ the tangent, normal, and binormal vectors of a right-handed unit orthogonal system, respectively. While $\mathbf{T}'(s) = d\mathbf{T}(s)/ds$, $\mathbf{N}'(s) = d\mathbf{N}(s)/ds$, $\mathbf{B}'(s) = d\mathbf{B}(s)/ds$ are the first derivatives of such vectors.

Thus, this orthogonal system is defined as $\mathbf{R}(s) := [\mathbf{T}(s)\ \mathbf{N}(s)\ \mathbf{B}(s)]$ and it can be integrated from Eq. (4.1) based on a specific pair of functions of curvature $\kappa(s)$ and torsion $\tau(s)$, with a given initial value $\mathbf{R}(0) := [\mathbf{T}(0)\ \mathbf{N}(0)\ \mathbf{B}(0)]$. The position is obtained by integrating the tangent vector:

$$\mathbf{C}(s) := \mathbf{C}(0) + \int_0^s \mathbf{T}(\xi)\, d\xi \tag{4.2}$$

On the other hand, based on a curve $\mathbf{C}(s)$, we can obtain the tangent $\mathbf{T}(s)$, normal $\mathbf{N}(s)$, and binormal $\mathbf{B}(s)$ vectors. Hence, taking into account the theory of differential geometry of curves [Banchoff y Lovett 2010], the following equalities for the mobile trihedron of any curve $\mathbf{C}(s)$ are defined

$$\mathbf{T}(s) := \frac{\mathbf{C}'(s)}{\|\mathbf{C}'(s)\|} \tag{4.3}$$

$$\mathbf{N}(s) := \frac{\mathbf{T}'(s)}{\|\mathbf{T}'(s)\|} = \frac{[\mathbf{C}'(s) \times \mathbf{C}''(s)] \times \mathbf{C}'(s)}{\|[\mathbf{C}'(s) \times \mathbf{C}''(s)] \times \mathbf{C}'(s)\|} \tag{4.4}$$

$$\mathbf{B}(s) := \mathbf{T}(s) \times \mathbf{N}(s) = \frac{\mathbf{C}'(s) \times \mathbf{C}''(s)}{\|\mathbf{C}'(s) \times \mathbf{C}''(s)\|} \tag{4.5}$$

being $\mathbf{C}'(s) = d\mathbf{C}(s)/ds$, $\mathbf{C}''(s) = d^2\mathbf{C}(s)/ds^2$, $\mathbf{C}'''(s) = d^3\mathbf{C}(s)/ds^3$ the derivatives of position vector $\mathbf{C}(s)$.

4.2.2 Planar and spatial clothoids

An Euler spiral or planar clothoid (C2D) is defined in $\mathbb{R}^2$ as the curve whose curvature varies linearly with respect to arc-length

$$\kappa(s) := \kappa_0 + \sigma_\kappa s \tag{4.6}$$

where κ_0 is the initial curvature and $\sigma_\kappa := d\kappa(s)/ds$ is the clothoid's sharpness, which is related to its homothety factor or scale K by $\sigma_\kappa := \pi/K^2$, and clothoid's tangent angle is

$$\beta(s, \mathbf{p}) := \beta_0 + \kappa_0 s + \frac{\sigma_\kappa}{2} s^2 \tag{4.7}$$

where β_0 is the initial tangent angle and $\mathbf{p} = \{\beta_0, \kappa_0, \sigma_\kappa\}$ is the parameter vector (for notation compactness). In addition to this, the tangent vector can be expressed as

$$\mathbf{T}(s, \mathbf{p}) := \begin{bmatrix} \cos(\beta(s, \mathbf{p})) \\ \sin(\beta(s, \mathbf{p})) \end{bmatrix} \tag{4.8}$$

From Eqs. (4.2), (4.7), and (4.8), well-known equations for a planar clothoid (contained in plane XY) [Levien 2008a], noted as $\mathbf{C}(s, \mathbf{p})$, can be derived from Fresnel integrals as follows

$$\mathbf{C}(s, \mathbf{p}) = \begin{bmatrix} \mathcal{C}(s, \mathbf{p}) \\ \mathcal{S}(s, \mathbf{p}) \end{bmatrix} := \begin{bmatrix} \int_0^s \cos(\beta_0 + \kappa_0 \xi + \frac{\sigma_\kappa}{2} \xi^2)\, d\xi \\ \int_0^s \sin(\beta_0 + \kappa_0 \xi + \frac{\sigma_\kappa}{2} \xi^2)\, d\xi \end{bmatrix} \tag{4.9}$$

being $\mathcal{C}(s, \mathbf{p})$ and $\mathcal{S}(s, \mathbf{p})$ the Fresnel's cosine and sine integrals. Note that without loss of generality, the clothoid is assumed to be origin-centered, that is $\mathbf{C}(0) = \mathbf{0}$ in Eq. (4.2).

It is a well-known fact that clothoids provide some interesting geometric, aspect-appearance, comfort, and safety advantages, which are worth to be used in road, railway, and roller coaster design [Meek y Walton 2004a; Marzbani, Jazar y Fard 2015]. Clothoids are also good to control non-holonomic vehicles [Girbés, Armesto y Tornero 2014; Marzbani y col. 2015b; Gim y col. 2017a]. Unfortunately, there is no closed-form solution to compute Fresnel integrals, although it is still possible to analytically compute some geometric properties such as curvature and tangent angle. Despite of clothoid not having analytical solution, some approximations allow real-time computation [Mielenz 2000; Montés y col. 2008; Brezak y Petrovic 2014; Narayan 2014; Chen y col. 2017a]. Thus, due to their good accuracy to compute clothoids, these approximations are considered, henceforth, as analytical solution. Hence, there is no loss of generality by assuming that clothoids can be computed analytically with errors up to 10^{-20} [Montés y col. 2008].

The authors of Harary y Tal 2012a introduced the 3D Euler spiral or clothoid (C3D), defined in $\mathbb{R}^3$, as a curve whose curvature varies as in Eq. (4.6) and its torsion evolves as

$$\tau(s) := \tau_0 + \sigma_\tau s \tag{4.10}$$

where τ_0 is the initial torsion, and $\sigma_\tau := d\tau/ds$ its first geometric derivative or torsion sharpness.

The main contribution of Harary y Tal 2012a is the development of a smooth curve to join two arbitrary point-vector configurations in the space for 3D curve completion. The authors used Frenet-Serret formulas (Eq. (4.1)) to compute a space curve as in Eq. (4.2). To find a solution to the problem, they implemented an optimization procedure using Gradient-descent algorithm [Press, William H., Teukolsky, Saul A., Vetterling, William T., and Flannery 2007] to get the optimum values of initial curvature κ_0 and torsion τ_0, as well as sharpness of curvature σ_κ and torsion σ_τ. Therefore, their solution could take any initial and final values of curvature and torsion.

However, aerial or underwater robots moving on a 3D space cannot follow arbitrary curves with instantaneous changes in orientation, curvature, or torsion. Therefore, in many practical cases, apart from taking into account non-holonomic constraints, trajectory planners use robot's current state to set initial values of curvature and torsion. In these cases, the algorithm in [Harary y Tal 2012a] only allows to reach either an arbitrary target position or orientation at a time, but not both simultaneously.

4.2.3 Problem Statement

As stated in section 4.2, it is possible to compute a curve $\mathbf{C}(s)$ using Frenet-Serret matrix, Eqs. (4.1) and (4.2). However, the main drawback is that, apart from some particular cases such as straight lines, helices, and some geometric curves, it is not possible to know *a priori* the position and orientation of the final configuration based on generic curvature and torsion functions. In order to compute them, numerical methods need to be used [Press, William H., Teukolsky, Saul A., Vetterling, William T., and Flannery 2007], which have a high computation cost for small integration steps. Besides, accuracy is inversely proportional to the integration step size. In these cases, an optimization procedure is required to solve planning problems to reach final Cartesian position and/or orientation [Harary y Tal 2012a], which is highly computationally demanding.

In this sense, it is of particular relevance that the proposed curve can provide not only analytical expressions and closed-form implementation and synthesis (curve design), with low computation time and high accuracy, but also some interesting properties/operations such as scaling, computing symmetric curves, ensuring smoothness, computing orientation, generalizing other simplistic curves, and ensuring that position and orientation increase monotonically [Knuth 1979].

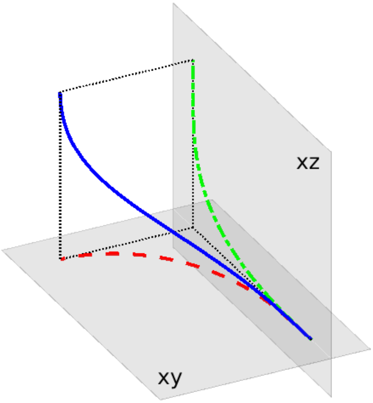


Figure 4.1: Cb3D curve (blue) and its projections onto XY plane (dashed red) and XZ plane dash-dotted green).

In particular, we seek for a curve parametrized by a generic vector $\mathbf{p}$, noted as $\mathbf{C}(s, \mathbf{p})$. Actually, the parameter vector can be divided into two subsets $\mathbf{p} := [\mathbf{p}_0 \; \tilde{\mathbf{p}}]$, where $\mathbf{p}_0$ is a list of parameters defining initial conditions, and $\tilde{\mathbf{p}}$ is the design parameter vector. The goal is to compute $\tilde{\mathbf{p}}^*$ such that the tangent vector of the curve $\mathbf{T}(s, \mathbf{p})$ equals an arbitrary target tangent vector $\mathbf{T}^*$ for a given $s > 0$.

Equivalently to the problem statement in Harary y Tal 2012a, the problem to solve can be stated as an orientation minimization problem

$$\tilde{\mathbf{p}}^* = \arg_{\tilde{\mathbf{p}}} \text{mín} \, \|\mathbf{T}^* - \mathbf{T}(s, \mathbf{p})\| \tag{4.11}$$

Without loss of generality, it can be assumed that the local frame of the curve is coincident to the global coordinate frame, which implies that the start point is $\mathbf{C}(0, \mathbf{p}) = [0 \; 0 \; 0]^T$.

4.3 Clothoid-based three-dimensional curve

A clothoid-based three-dimensional curve (Cb3D) is defined as a composition of two planar clothoids. As depicted in Fig. 4.1, each clothoid is contained in a plane orthogonal to each other (clothoids are defined in planes XY and XZ). However, the clothoid contained in plane XZ is indeed a clothoid whose length depends on the clothoid on plane XY, as explained later.

4.3.1 Curve definition and properties

Lemma 4.3.1 *Let* $\mathbf{C}_1(q, \mathbf{p}_1) := \begin{bmatrix} \mathcal{C}(q, \mathbf{p}_1) \\ \mathcal{S}(q, \mathbf{p}_1) \end{bmatrix} \in \mathbb{R}^2$ *be a first planar clothoid contained in plane* XY *with arc-length* q *and parameters* $\mathbf{p}_1 = \{\psi_0, \epsilon_0, \mu\}$*, and* $\mathbf{C}_2(w, \mathbf{p}_2) := \begin{bmatrix} \mathcal{C}(w, \mathbf{p}_2) \\ \mathcal{S}(w, \mathbf{p}_2) \end{bmatrix} \in \mathbb{R}^2$ *a second planar clothoid contained in plane* XZ *with arc-length* w *and parameters* $\mathbf{p}_2 = \{\theta_0, \upsilon_0, \rho\}$*, both defined as in Eq. (4.9). A Cb3D curve can be computed as a composition of both planar clothoids as follows*

$$\mathbf{C}(s, \mathbf{p}) \equiv \mathbf{C}(s, \mathbf{p}_1, \mathbf{p}_2) := \left\{ \begin{array}{c} \mathcal{C}(\mathcal{C}(s, \mathbf{p}_2), \mathbf{p}_1) \\ \mathcal{S}(\mathcal{C}(s, \mathbf{p}_2), \mathbf{p}_1) \\ -\mathcal{S}(s, \mathbf{p}_2) \end{array} \right. \tag{4.12}$$

where the spatial curve has the following properties/operations:

A ***Scalability***: $\lambda \mathbf{C}(s, \mathbf{p}_1, \mathbf{p}_2) = \mathbf{C}(\lambda s, \boldsymbol{\lambda}^{-1} \cdot \mathbf{p}_1, \boldsymbol{\lambda}^{-1} \cdot \mathbf{p}_2)$*, with* $\boldsymbol{\lambda}^{-1} := [1, \frac{1}{\lambda}, \frac{1}{\lambda^2}]$.

B ***Symmetry***: $\mathbf{C}(s, \mathbf{q} \cdot \mathbf{p}_1, \mathbf{q} \cdot \mathbf{p}_2) = -\mathbf{C}(-s, \mathbf{q} \cdot \mathbf{p}_1, \mathbf{q} \cdot \mathbf{p}_2)$*, with* $\mathbf{q} := [\pm 1, \mp 1, \pm 1]$.

C ***Smoothness***: *A Cb3D curve is* C^∞ *smooth with* $\epsilon_0 \neq 0$ *or* $\mu \neq 0$*;* $\upsilon_0 \neq 0$ *or* $\rho \neq 0$.

D ***Orientation***: *Pitch* $\theta(s, \mathbf{p}_2)$*, yaw* $\psi(s, \mathbf{p}_1, \mathbf{p}_2)$*, and roll* $\phi(s, \mathbf{p}_1, \mathbf{p}_2)$ *angles defined as*

$$\theta(s, \mathbf{p}_2) := \theta_0 + \upsilon_0 s + \frac{\rho}{2} s^2 \tag{4.13}$$

$$\psi(s, \mathbf{p}_1, \mathbf{p}_2) := \psi_0 + \epsilon_0 \mathcal{C}(s, \mathbf{p}_2) + \frac{\mu}{2} \mathcal{C}^2(s, \mathbf{p}_2) \tag{4.14}$$

$$\phi(s, \mathbf{p}_1, \mathbf{p}_2) := -\arcsin\left(\frac{\theta'(s, \mathbf{p}_2)}{\sqrt{\omega(s, \mathbf{p}_1, \mathbf{p}_2)^2 + \theta'(s, \mathbf{p}_2)^2}} \right) \tag{4.15}$$

being $\omega(s, \mathbf{p}_1, \mathbf{p}_2) := \cos(\theta(s, \mathbf{p}_2)) \psi'(s, \mathbf{p}_1, \mathbf{p}_2)$, $\theta'(s, \mathbf{p}_2) := d\theta(s, \mathbf{p}_2)/ds$*, and* $\psi'(s, \mathbf{p}_1, \mathbf{p}_2) := d\psi(s, \mathbf{p}_1, \mathbf{p}_2)/ds$*; where tangent, normal, and binormal vectors can be computed as:*

$$\mathbf{T}(s,\mathbf{p}_1,\mathbf{p}_2) := \begin{bmatrix} \cos(\psi(s,\mathbf{p}_1,\mathbf{p}_2))\cos(\theta(s,\mathbf{p}_2)) \\ \sin(\psi(s,\mathbf{p}_1,\mathbf{p}_2))\cos(\theta(s,\mathbf{p}_2)) \\ -\sin(\theta(s,\mathbf{p}_2)) \end{bmatrix} \tag{4.16}$$

$$\mathbf{N}(s,\mathbf{p}_1,\mathbf{p}_2) := \frac{\mathbf{T}'(s,\mathbf{p}_1,\mathbf{p}_2)}{\|\mathbf{T}'(s,\mathbf{p}_1,\mathbf{p}_2)\|} \tag{4.17}$$

$$\mathbf{B}(s,\mathbf{p}_1,\mathbf{p}_2) := \mathbf{T}(s,\mathbf{p}_1,\mathbf{p}_2) \times \mathbf{N}(s,\mathbf{p}_1,\mathbf{p}_2) \tag{4.18}$$

E ***Straight line****: A Cb3D curve is a straight line iff* $\mathbf{p}_1 = \{\forall\psi_0, \epsilon_0 = 0, \mu = 0\}$ *and* $\mathbf{p}_2 = \{\forall\theta_0, \upsilon_0 = 0, \rho = 0\}$*.*

F ***Circular arc****: A Cb3D curve is a circular arc iff* $\mathbf{p}_1 = \{\forall\psi_0, \epsilon_0 \neq 0, \mu = 0\}$ *and* $\mathbf{p}_2 = \{\theta_0 = 0, \upsilon_0 = 0, \rho = 0\}$*, or* $\mathbf{p}_1 = \{\forall\psi_0, \epsilon_0 = 0, \mu = 0\}$ *and* $\mathbf{p}_2 = \{\forall\theta_0, \upsilon_0 \neq 0, \rho = 0\}$*.*

G ***Planar clothoid****: A Cb3D curve is a planar clothoid iff* $\mathbf{p}_1 = \{\psi_0 = 0, \epsilon_0 = 0, \mu = 0\}$ *and* $\forall\mathbf{p}_2$*, or* $\mathbf{p}_2 = \{\theta_0 = 0, \upsilon_0 = 0, \rho = 0\}$ *and* $\forall\mathbf{p}_1$*.*

H ***Circular helix****: A Cb3D curve is a circular helix iff* $\mathbf{p}_1 = \{\forall\psi_0, \epsilon_0 \neq 0, \mu = 0\}$ *and* $\mathbf{p}_2 = \{\theta_0 \neq 0, \upsilon_0 = 0, \rho = 0\}$*.*

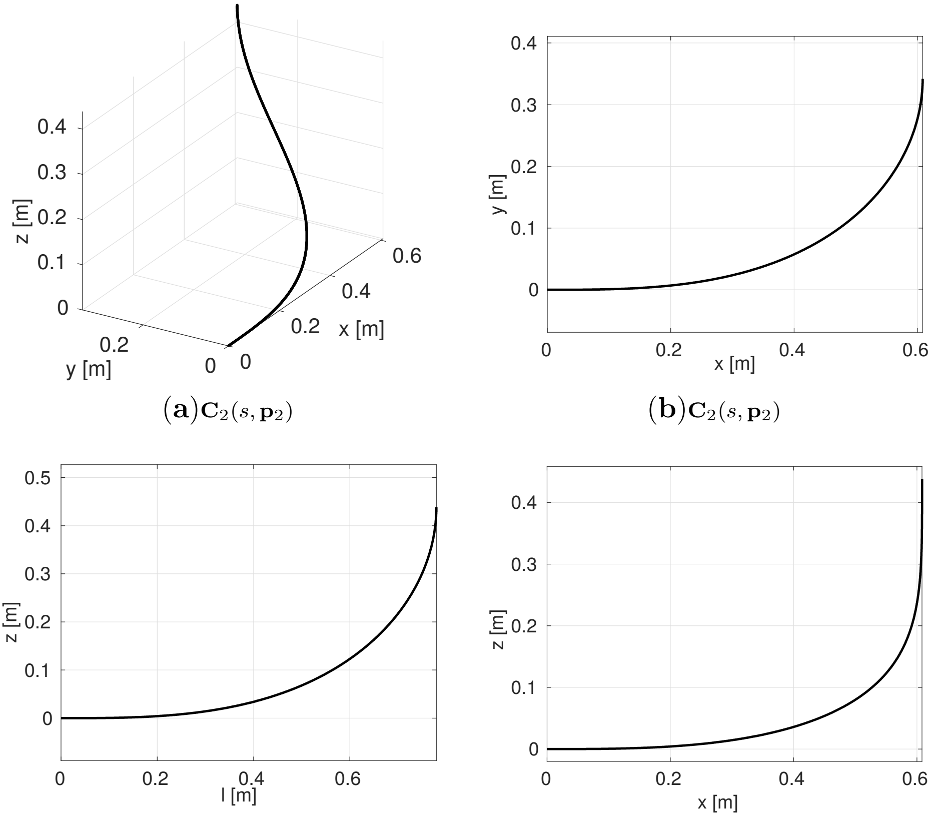
I ***Monotonic increasing in position*** *of* $\mathbf{C}(s,\mathbf{p}_1,\mathbf{p}_2)$ *for any* $s>0$*, with* $\theta(s,\mathbf{p}_2) \in [-\pi/2, 0]$ *and* $\psi(s,\mathbf{p}_1,\mathbf{p}_2) \in [0,\pi/2]$*, iff* $-\pi/2 \leq \theta_0 \leq 0$*,* $0 \leq \psi_0 \leq \pi/2$*,* $\upsilon_0 \leq 0$*,* $\epsilon_0 \geq 0$*,* $\rho \leq 0$ *and* $\mu \geq 0$*.*

J ***Monotonic increasing pitch and yaw angles*** *for any* $s > 0$*, with* $\theta(s,\mathbf{p}_2) \in [0,\pi/2]$ *and* $\psi(s,\mathbf{p}_1,\mathbf{p}_2) \in [0,\pi]$*, iff* $0 \leq \theta_0 \leq \pi/2$*,* $0 \leq \psi_0 \leq \pi$*,* $\upsilon_0 \geq 0$*,* $\epsilon_0 \geq 0$*,* $\rho \geq 0$ *and* $\mu \geq 0$*.*

Proof 4.3.1 *Appendix A contains a detailed list of proofs for the properties stated in Lemma 4.3.1.* ■

Fig. 4.2 shows a Cb3D curve $\mathbf{C}(s,\mathbf{p}_1,\mathbf{p}_2)$ with the following parameters $\mathbf{p}_2 = \{0, 0, -\pi\}$ and $\mathbf{p}_1 = \{0, 0, \pi/\mathcal{C}^2(1,\mathbf{p}_2)\}$, with arc-length $s \in [0,1]$ m. The final tangent vector is $\mathbf{T}(1,\mathbf{p}_1,\mathbf{p}_2) = [0,0,1]^T$, which is equivalent to final orientation angles yaw $\psi(1,\mathbf{p}_1,\mathbf{p}_2) = \pi/2$ rad and pitch $\theta(1,\mathbf{p}_2) = -\pi/2$ rad, as stated in Eq. (4.16). Notice that the projection of a Cb3D curve onto XY plane (horizontal plane $Z=0$) is the planar clothoid $\mathbf{C}_1(s,\mathbf{p}_1,\mathbf{p}_2)$, as can be seen in Fig. 4.2(b).

On the other hand, $\mathbf{C}_2(s,\mathbf{p}_2)$ is a second clothoid produced by unrolling the Cb3D curve onto a virtual vertical plane defined by the length of the horizontal



(**a**)$\mathbf{C}_2(s, \mathbf{p}_2)$ (**b**)$\mathbf{C}_2(s, \mathbf{p}_2)$

(**c**)$\mathbf{C}_2(s, \mathbf{p}_2)$ (**d**)Projection onto XZ plane

Figure 4.2: Cb3D curve $\mathbf{C}(s, \mathbf{p}_1, \mathbf{p}_2)$, for $s \in [0, 1]$ m, $\mathbf{p}_2 = \{0, 0, -\pi\}$ and $\mathbf{p}_1 = \{0, 0, \pi/\mathcal{C}^2(1, \mathbf{p}_2)\}$, with $l \equiv \mathcal{C}(s, \mathbf{p}_2)$.

clothoid $l \equiv \mathcal{C}(s, \mathbf{p}_2)$ as base and $z \equiv \mathcal{S}(s, \mathbf{p}_2)$ as height, shown in Fig. 4.2(c). It is interesting to remark that the projection of a Cb3D curve onto XZ plane is not strictly a planar clothoid, but a 2D clothoid flattened along its x component, as can be observed in Fig. 4.2(d). Based on the symmetry property (C) of Lemma 4.3.1, a Cb3D curve can be defined for $s \in \mathbb{R}$ in any of the 8 octants in $\mathbb{R}^3$ space, as depicted in Fig. 4.3, where a Cb3D curve parametrized as in Fig. 4.2 has been used.

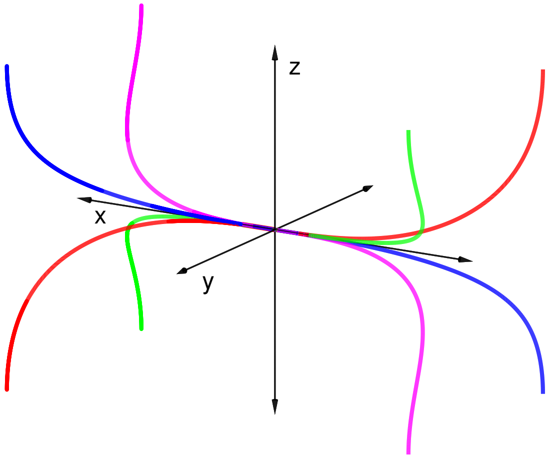


Figure 4.3: Family of symmetric unitary Cb3D curves covering all 8 octants, parametrized as in Fig. 4.2.

4.3.2 Synthesis

As explained in the problem statement, the parameter vector $\mathbf{p}$ is composed by two sets of parameters: $\mathbf{p}_0$ related to the initial conditions of the curve and $\tilde{\mathbf{p}}$ the actual parameters to design. In particular, the Cb3D curve uses $\mathbf{p}_0 := \{\psi_0, \theta_0, \epsilon_0, \upsilon_0\}$ and $\tilde{\mathbf{p}} := \{\mu, \rho\}$. So, let's recall that our aim is to find $\tilde{\mathbf{p}}^* := \{\mu^*, \rho^*\}$ such that Eq. (4.11) is minimized. Thus, by abusing of notation we denote $\mathbf{C}(s, \mathbf{p}^*) \equiv \mathbf{C}(s, \mathbf{p}_0, \tilde{\mathbf{p}}^*) \equiv \mathbf{C}(s, \mathbf{p}_1^*, \mathbf{p}_2^*)$, with $\mathbf{p}_1^* := \{\psi_0, \epsilon_0, \mu^*\}$ and $\mathbf{p}_2^* := \{\theta_0, \upsilon_0, \rho^*\}$.

Lemma 4.3.2 Given an arbitrary target tangent vector $\mathbf{T}^*$, defined by final angles $\theta^* \in [-\pi/2, \pi/2]$ rad and $\psi^* \in [-\pi, \pi]$ rad, and given values of arc-length $s > 0$, and parameters $\mathbf{p}_0 = \{\psi_0, \theta_0, \epsilon_0, \upsilon_0\}$ which can be obtained from initial orientation, curvature, and torsion, there is only one pair of parameters $\tilde{\mathbf{p}}^* = \{\mu^*, \rho^*\}$ that synthesize a Cb3D curve $\mathbf{C}(s, \mathbf{p}_1^*, \mathbf{p}_2^*)$ reaching $\mathbf{T}(s, \mathbf{p}_1^*, \mathbf{p}_2^*) = \mathbf{T}^*$.

Proof 4.3.2 *For a given value of arc-length $s > 0$, and final angles $\{\theta^*, \psi^*\}$, using Eqs.* (4.13) *and* (4.14)*, the parameters in $\tilde{\mathbf{p}}^*$ can be computed as*

$$\rho^* = \frac{2\,(\theta^* - \theta_0 - \upsilon_0 s)}{s^2} \tag{4.19}$$

$$\mu^* = \frac{2\,(\psi^* - \psi_0 - \epsilon_0 \mathcal{C}(s, \mathbf{p}_2^*))}{\mathcal{C}^2(s, \mathbf{p}_2^*)} \tag{4.20}$$

Therefore, the solution is unique and synthesizes a Cb3D curve expressed as in Eqs. (A.10) *and* (A.11) *of Appendix A*

$$\mathbf{C}(s, \mathbf{p}_1^*, \mathbf{p}_2^*) := \begin{cases} \int_0^{\mathcal{C}(s,\mathbf{p}_2^*)} \cos(\psi_0 + \epsilon_0 \xi + \frac{\mu^*}{2}\xi^2)\, d\xi \\ \int_0^{\mathcal{C}(s,\mathbf{p}_2^*)} \sin(\psi_0 + \epsilon_0 \xi + \frac{\mu^*}{2}\xi^2)\, d\xi \\ -\int_0^{s} \sin(\theta_0 + \upsilon_0 \xi + \frac{\rho^*}{2}\xi^2)\, d\xi \end{cases} \tag{4.21}$$

with

$$\mathcal{C}(s, \mathbf{p}_2^*) := \int_0^s \cos(\theta_0 + \upsilon_0 \xi + \frac{\rho^*}{2}\xi^2)\, d\xi \tag{4.22}$$

■

Lemma 4.3.3 Given an arbitrary final tangent vector $\mathbf{T}^*$, and any pair of arc-lengths $s_1 > 0$ and $s_2 > 0$, the solutions to reach $\mathbf{T}^*$, computed as in Lemma 4.3.2 (Eqs. (4.19) and (4.20)), are $\tilde{\mathbf{p}}_1^* := \{\mu_1^*, \rho_1^*\}$ and $\tilde{\mathbf{p}}_2^* := \{\mu_2^*, \rho_2^*\}$ for each arc-length. Then, the ratio $\eta^* := \rho_1^*/\mu_1^* = \rho_2^*/\mu_2^*$ is a constant value for $\forall s_1 \in \mathbb{R}^+$ and $\forall s_2 \in \mathbb{R}^+$ iff $\upsilon_0 = 0$ and $\epsilon_0 = 0$.

Proof 4.3.3 *From property (A) of Lemma 4.3.1 (Eqs.* (A.1) *and* (A.2) *in Appendix A), it is straightforward to see that scaling arc-length by a factor λ, so that $s_2 = \lambda s_1$, requires a new pair of parameters $\{\rho_2^*, \mu_2^*\} = \{\rho_1^*/\lambda^2, \mu_1^*/\lambda^2\}$, whose ratio η^* remains constant.*

$$\rho_2^* = \frac{2(\theta^* - \theta_0)}{s_2^2} = \frac{2(\theta^* - \theta_0)}{\lambda^2 s_1^2} = \frac{\rho_1^*}{\lambda^2} \tag{4.23}$$

$$\mu_2^* = \frac{2(\psi^* - \psi_0)}{\mathcal{C}^2(s_2, \mathbf{p}_2^*)} = \frac{2(\psi^* - \psi_0)}{\mathcal{C}^2(\lambda s_1, \boldsymbol{\lambda}^{-1} \cdot \mathbf{p}_1^*)} = \frac{2(\psi^* - \psi_0)}{\lambda^2 \mathcal{C}^2(s_1, \mathbf{p}_1^*)} = \frac{\mu_1^*}{\lambda^2} \tag{4.24}$$

■

Figure 4.4(a) shows a Cb3D curve reaching an arbitrary final tangent vector $\mathbf{T}^*$. It can be observed that final angles θ^* and ψ^* are the deflection angles of

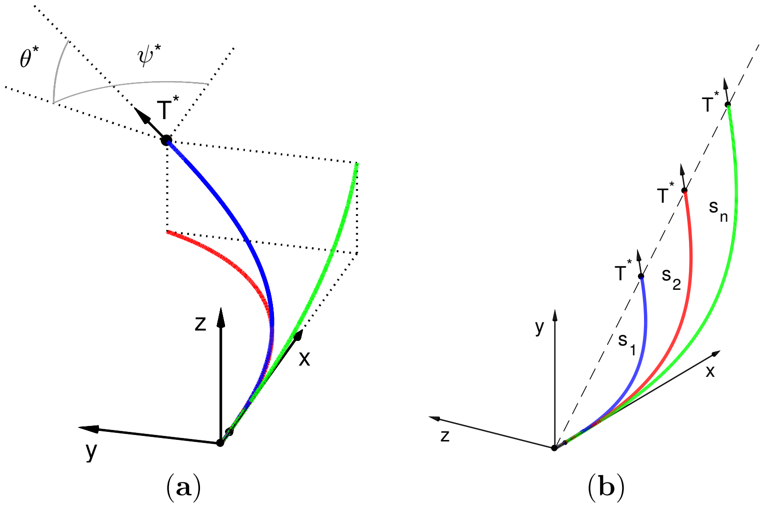


Figure 4.4: Examples of Cb3D curves reaching arbitrary final tangent vector $\mathbf{T}^*$: (a) Cb3D curve (blue) and its projections onto XY plane (red) and XZ plane (green). (b) Family of Cb3D curves with different arc-lengths.

the tangent vector with respect to planes XY and XZ, as stated in the orientation property (D) of Lemma 4.3.1. On the other hand, Fig. 4.4(b) illustrates Lemma 4.3.3, closely related to the scaling property (A) of Lemma 4.3.1, where multiplying a Cb3D curve by a scale factor does not modify its shape and final orientation.

Lemma 4.3.4 Given an arbitrary target position vector $\mathbf{C}^* := \{x^*, y^*, z^*\}$, with arc-length $s > 0$ and initial parameters $\mathbf{p}_0 = \{0, 0, 0, 0\}$, there is only one pair of parameters $\tilde{\mathbf{p}}^* = \{\mu^*, \rho^*\}$ that synthesize a Cb3D curve as a solution to the following position minimization problem

$$\tilde{\mathbf{p}}^* = \arg_{\tilde{\mathbf{p}}} \text{mín} \, \|\mathbf{C}^* - \mathbf{C}(s, \mathbf{p}_0, \tilde{\mathbf{p}})\| \tag{4.25}$$

where, for $\mathbf{C}^*$ to be reachable by the Cb3D curve, the following conditions are required

$$|y^*| \leq 1{,}765042236 \cdot x^* \tag{4.26}$$

$$|z^*| \leq 0{,}561947501 \cdot l^* \tag{4.27}$$

being l^* the arc-length of a planar clothoid $\mathbf{C}_1(l^*, \mathbf{p}_1) = \{x^*, y^*\}$, so that the condition that constrains component z^* depends on the solution of the projection of $\mathbf{C}^*$ onto XY plane.

Proof 4.3.4 *If a solution for* $\mathbf{C}^* = \mathbf{C}(s, \mathbf{p}_0, \tilde{\mathbf{p}}^*)$ *can be found, therefore such solution will minimize* (4.25)*. So, let's define a unitary planar clothoid* $\mathbf{C}_1(s_1, \mathbf{p}_1)$ *as in Eq.* (4.9)*, with* $\mathbf{p}_1 = \{0, 0, \pi\}$*. The goal is to solve the following problem*

$$\frac{|y^*|}{x^*} = \frac{\mathcal{S}(s_1, \mathbf{p}_1)}{\mathcal{C}(s_1, \mathbf{p}_1)} \tag{4.28}$$

being s_1 *the arc-length of* $\mathbf{C}_1(s_1, \mathbf{p}_1)$ *contained in the horizontal plane, where the solution* s_1^* *can be found using any root finding algorithm, and it is unique iff condition* (4.26) *is satisfied. Such condition establishes the maximum ratio between coordinates* X *and* Y*, which is produced when the line tangent to the planar clothoid passes through the origin Kostov y Degtiariova-Kostova 1995, where for a unitary clothoid it happens at* $s_1 = 1{,}634577$ *m, so that* $\mathbf{C}_1(1{,}634577, \{0, 0, \pi\}) \approx \{0{,}345860, 0{,}610458\}$ *m.*

From property (A) of Lemma 3.1 (Eqs. (A.1) *and* (A.2) *in Appendix A), the horizontal unitary clothoid contained in plane* XY*, found as a solution to Eq.* (4.28)*, can be scaled to the original target position coordinates, so that* $\{x^*, |y^*|\} = K_1\mathbf{C}(s_1^*, \mathbf{p}_1)$*, with scaled arc-length* $l^* = K_1 s_1^*$*, being* K_1 *the homothety factor of the first planar clothoid, which can be easily computed as*

$$K_1 = \frac{x^*}{\mathcal{C}(s_1^*, \mathbf{p}_1)} = \frac{|y^*|}{\mathcal{S}(s_1^*, \mathbf{p}_1)} \tag{4.29}$$

Now, let's repeat the previous steps for a second unitary planar clothoid $\mathbf{C}_2(s_2, \mathbf{p}_2)$ *in the vertical plane, with* $\mathbf{p}_2 = \{0, 0, \pi\}$*. We aim to find a solution to the following equality*

$$\frac{|z^*|}{l^*} = \frac{\mathcal{S}(s_2, \mathbf{p}_2)}{\mathcal{C}(s_2, \mathbf{p}_2)} \tag{4.30}$$

being s_2 *the arc-length of* $\mathbf{C}_2(s_2, \mathbf{p}_2)$*, where the solution* s_2^* *can be found by any root finding algorithm, being unique iff condition* (4.27) *is satisfied, where for a unitary clothoid it happens at* $s_2 = 1$ *m, so that* $\mathbf{C}_2(1, \{0, 0, \pi\}) \approx \{0{,}779893, 0{,}438259\}$ *m. Again, using property (A) of Lemma 3.1, the unitary clothoid can be scaled* $\{l^*, |z^*|\} = K_2\mathbf{C}(s_2^*, \mathbf{p}_2)$*, where* $s^* = K_2 s_2^*$ *is the arc-length of the vertical planar clothoid and* K_2 *its homothety factor*

$$K_2 = \frac{l^*}{\mathcal{C}(s_2^*, \mathbf{p}_2)} = \frac{|z^*|}{\mathcal{S}(s_2^*, \mathbf{p}_2)} \tag{4.31}$$

Using planar clothoids properties, $\tilde{\mathbf{p}}^* = \{\mu^*, \rho^*\}$ *can be computed as* $\mu^* = \operatorname{sgn}(y^*)\frac{\pi}{K_1^2}$ *and* $\rho^* = -\operatorname{sgn}(z^*)\frac{\pi}{K_2^2}$*, where* $\operatorname{sgn}(\bullet)$ *is the signum function.* ■

As a conclusion, Lemmas 4.3.2, 4.3.3 and 4.3.4 imply that a Cb3D curve can reach a target position $\mathbf{C}^*$ in the space with one and only one orientation $\mathbf{T}^*$. On the contrary, a target tangent vector $\mathbf{T}^*$ can be reached by a Cb3D curve in multiple positions, depending on the arc-length $s > 0$. However, the final position of such solutions lies in a straight line, since any solution to the problem has the same shape and is related by a scaling factor λ.

Finally, it is obvious that the solution to the orientation minimization problem to reach target tangent vector $\mathbf{T}^*$, which is stated in Eq. (4.11) and given by the pair of design parameters $\tilde{\mathbf{p}}^* = \{\mu^*, \rho^*\}$, is the same as the solution obtained when solving the position minimization problem in Eq. (4.25), iff arc-length $s = s^*$, where s^* is the length of the Cb3D curve designed to reach target position $\mathbf{C}^*$. Therefore, in path planning problems, due to geometric constraints only final arbitrary orientation or position can be reached by a Cb3D curve, but not both simultaneously.

4.4 Comparison: C3D vs. Cb3D

In this section we will compare the proposed clothoid-based 3D curve (Cb3D) and the pure 3D clothoid (C3D) introduced in [Harary y Tal 2012a]. Both curves are considered geometric curves and therefore can be directly compared considering not just only their geometric properties, but also their kinematic properties as well as their computational time requirements. An example of singular case for C3D is also shown and explained in order to compare both curves.

The authors of [Harary y Tal 2012a] used a fixed step Euler integration method [Press, William H., Teukolsky, Saul A., Vetterling, William T., and Flannery 2007]. In this sense, to speed up computation, such method has been improved by implementing an algorithm to solve ordinary differential equations with variable step [Press, William H., Teukolsky, Saul A., Vetterling, William T., and Flannery 2007]. Besides, values of curvature sharpness σ_κ and torsion sharpness σ_τ required to join initial and final orientations cannot be known *a priori*, so an optimization procedure using Gradient-descent algorithm is required [Harary y Tal 2012a]. Regarding the parameter initialization of the optimizer, initial sharpness pair (σ_{κ_0} and σ_{τ_0}) is set to the solution proposed in our method. Although they do not represent the same variables, we could verify that this initialization is better than random or *naive* initializations, in terms of computation time and final accuracy. On the other hand, initial values of curvature and torsion are set to $\kappa_0 = 0$ and $\tau_0 = 0$, and they are not modified during optimization procedure, which differs from [Harary y Tal 2012a]. This modification is due to the fact that we want to solve the problem of a fixed-wing airplane following a straight line, which in a certain time instant needs to perform a maneuver to turn and change its height and direction of movement. Therefore, $\kappa_0 = \tau_0 = 0$ rad/m is required as a boundary condition for the problem to be solved.

It is interesting to remark that, other approaches, based on parametric curves, such as Bézier, Splines or B-spline curves were considered as well, however as stated by [Pérez y col. 2018], we found they were very sensitive to control points initialization and in order to be comparable with clothoid-based approaches they were required to minimize the jerk and to impose specific constraints on the curves. The results were not satisfactory and are not shown for brevity.

The test performed consists of planning a curve joining the initial tangent vector $\mathbf{T}(0) = [1\ 0\ 0]^T$, equivalent to $\mathbf{p}_0 = \{0,0,0,0\}$, to a final arbitrary $\mathbf{T}^*$, whose pitch and yaw angles are randomly drawn from a uniform distribution $\theta^* \sim \mathcal{U}(0,\pi/2)$ rad and $\psi^* \sim \mathcal{U}(0,\pi/2)$ rad. A total of $n = 10^3$ random combinations of samples have been generated. For all cases the curve length is set to $s = 1$ m.

Computation Time and Accuracy

As shown in Table 4.1, computation time of a Cb3D curve is much lower (around 10^{-4} times smaller) than for the case of C3D. The main reason is that Cb3D curves can be computed in closed-form based on C2D approximations [Mielenz 2000; Montés y col. 2008], while C3D computation requires numerical integration of Eqs. (4.1) and (4.2) [Harary y Tal 2012a], which is much slower.

Now, analyzing the worst case of computation time, for C3D it is around 0,4 s, while the maximum computation time of a Cb3D curve is approximately 700 μs. Therefore, our method can be used to compute 3D smooth curves in real-time. Note that all processing has been done in a computer with Intel Core i7-6700HQ 2.60GHz processor and 16GiB DDR4 memory, using an implementation of both methods in MATLAB® R2017a.

Table 4.1: Computation time and accuracy: Cb3D vs. C3D.

Performance Metric	Space curve	Mean	Standard deviation	Worst case
Computation time [ms]	**C3D**	282.241	27.670	413.251
	Cb3D	$32.991{\cdot}10^{-3}$	$28.499{\cdot}10^{-3}$	0.731
Accuracy [m]	**C3D**	$7.595{\cdot}10^{-15}$	$49.053{\cdot}10^{-15}$	$8.982{\cdot}10^{-13}$
	Cb3D	$1.764{\cdot}10^{-16}$	$1.399{\cdot}10^{-16}$	$8.006{\cdot}10^{-16}$

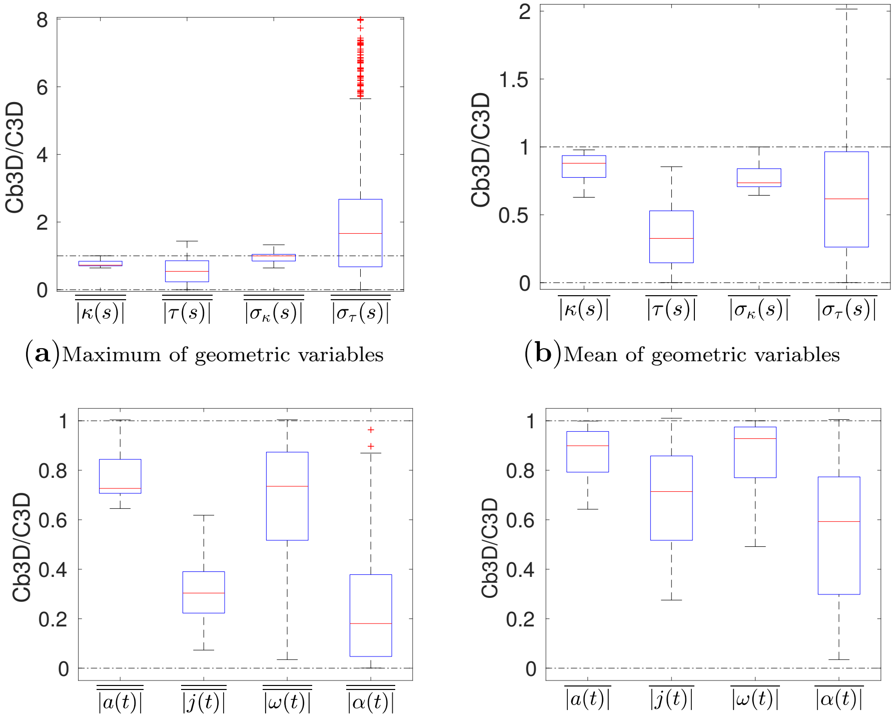
Regarding the accuracy, a Cb3D curve has the same order of magnitude as planar clothoids [Mielenz 2000], whilst the accuracy of C3D depends on the stop conditions of the optimization method. Thus, the more accurate, the higher the computation time, and *vice versa*. In order to obtain the results of Table 4.1, the stop condition was intentionally set to $e = 10^{-15}$ m to do a fair comparison, where $e = \sqrt{(\theta^* - \theta)^2 + (\varphi^* - \varphi)^2}$ is the error between the target orientation $\{\theta^*, \varphi^*\}$ and the final orientation $\{\theta, \varphi\}$ reached by the generated C3D.

Geometry and Kinematics

Some geometric and kinematic metrics are evaluated in this section. On the one hand, geometric metrics, which only depend on arc-length s, comprise mean and maximum values of curvature $\kappa(s)$, torsion $\tau(s)$, and their derivatives, curvature sharpness $\sigma_\kappa(s)$ and torsion sharpness $\sigma_\tau(s)$. On the other hand, kinematic variables, which depend on time t, are mean and maximum values of acceleration $a(t)$, jerk $j(t)$, angular velocity $\omega(t)$, and angular acceleration $\alpha(t)$. Kinematic variables have been computed assuming a constant linear velocity $vel = 1$ m/s, so that curve length is time-dependent $s(t) = t$. Besides, since all four kinematic variables are vectors, metrics have been computed as the 2-norm of such vectors. Note that $\overline{\overline{|\bullet|}}$ means maximum values, while $\overline{|\bullet|}$ means average values.

For each random case, metrics have been computed as a ratio between absolute values of the variables obtained when computing Cb3D with respect to C3D, so that $r_i := |m_{i_{Cb3D}}|/|m_{i_{C3D}}|$ is the ratio of the i-th metric, where $m_{i_{Cb3D}}$ is the metric obtained with Cb3D and $m_{i_{C3D}}$ is the metric obtained with C3D.

From Fig. 4.5(a), it can be concluded that Cb3D tends to have lower maximum values of curvature $\overline{\overline{|\kappa|}}$ (100 % of cases) and torsion $\overline{\overline{|\tau|}}$ (86,2 % of cases). Moreo-



(**a**)Maximum of geometric variables

(**b**)Mean of geometric variables

(**c**)Maximum of kinematic variables

(**d**)Mean of kinematic variables

Figure 4.5: Ratio Cb3D/C3D of metrics related to geometry (a,b) and kinematics (c,d), with maximum absolute values (a,c) and mean absolute values (b,d).

ver, the maximum value of curvature sharpness $\overline{\overline{|\sigma_\kappa|}}$ is very similar according to the box-whiskers diagram, although slightly better for Cb3D computation (50,8 % of cases). However, the maximum value of torsion sharpness $\overline{\overline{|\sigma_\tau|}}$ is worse for Cb3D (only in 34,3 % of cases Cb3D is better). This phenomenon is explained later in a graphical example of a particular case representing this situation.

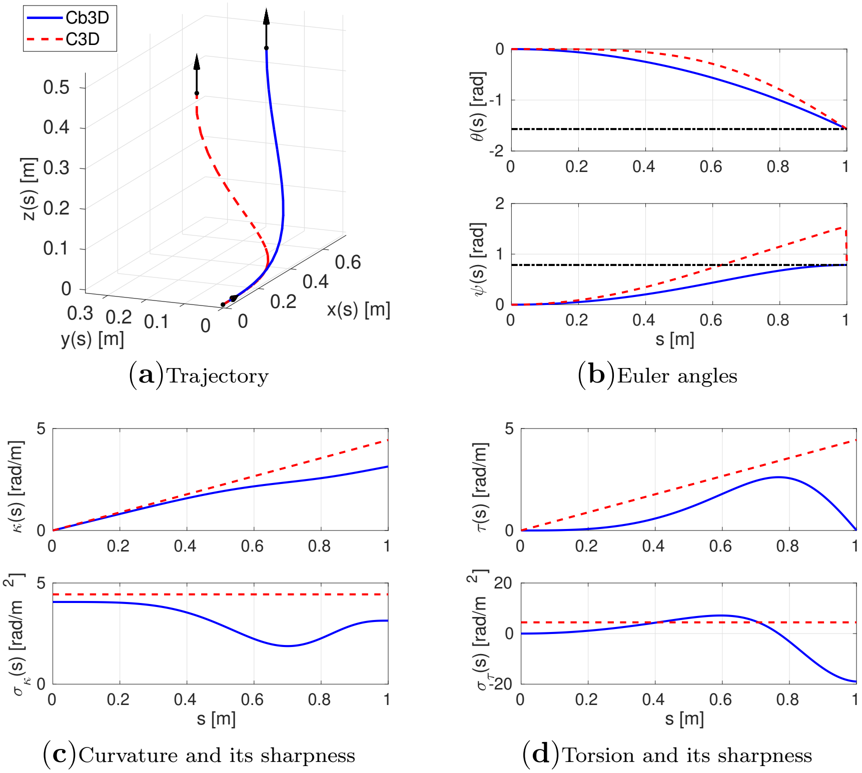
In Fig. 4.5(b), the average values of geometric variables show that Cb3D tends to have lower mean values in all four metrics: $\overline{|\kappa|}$ (100 % of cases), torsion $\overline{|\tau|}$ (100 % of cases), curvature sharpness $\overline{|\sigma_\kappa|}$ (100 % of cases), and torsion sharpness $\overline{|\sigma_\tau|}$ (77,1 % of cases). This happens because, as explained later through a graphical example, the Cb3D has lower values most of the time. However, in some particular cases the maximum value of torsion sharpness increases very much, which has 3 different effects: 1) maximum torsion sharpness is much higher in Cb3D than C3D ($r_i > 1$ for $\overline{\overline{|\sigma_\tau|}}$), 2) which might imply that the average torsion sharpness of Cb3D surpass the value of C3D ($r_i > 1$ for $\overline{|\sigma_\tau|}$), and 3) the maximum torsion of Cb3D might also increase to be higher than C3D ($r_i > 1$ for $\overline{\overline{|\tau|}}$).

Regarding kinematic variables, Fig. 4.5(c) shows that maximum values tend to be lower in Cb3D curve, since most ratios are $r_i < 1$: acceleration $\overline{\overline{|a|}}$ (96,6 % of cases), jerk $\overline{\overline{|j|}}$ (100 % of cases), angular velocity $\overline{\overline{|\omega|}}$ (96,5 % of cases), and angular acceleration $\overline{\overline{|\alpha|}}$ (100 % of cases). Something similar happens for ratios of mean values: acceleration $\overline{|a|}$ (100 % of cases), jerk $\overline{|j|}$ (95,6 % of cases), angular velocity $\overline{|\omega|}$ (100 % of cases), and angular acceleration $\overline{|\alpha|}$ (97,3 % of cases).

Singular case

Figure 4.6 shows a singular case in which both curves converge to final orientation $\theta^* = -\pi/2$ and $\psi^* = \pi/4$ (gimbal effect of Euler angles with singularity $\theta^* \to \pm\pi/2$ rad), as depicted in Fig. 4.8(b). In this particular case, it can be observed that C3D (dashed red line) needs to correct the yaw angle very abruptly, while Cb3D solves the same problem with a smoother solution based on properties (j) and (k) of Lemma 4.3.1.

Regarding curvature and torsion, Figs. 4.6(c) and 4.6(d) show that for a Cb3D curve these variables are non-linear, since their sharpness are not constant values. This result implies that the proposed clothoid-based three-dimensional curve (Cb3D) is not claimed to be a pure three-dimensional clothoid (C3D), but a smooth space curve based on a composition of two-dimensional clothoids (C2D).



(a) Trajectory

(b) Euler angles

(c) Curvature and its sharpness

(d) Torsion and its sharpness

Figure 4.6: Singular case to reach final orientation $\theta^* = -\pi/2$ rad and $\psi^* = \pi/4$ rad: Cb3D (blue) vs. C3D (dashed red).

4.5 Simulation results

Our curve proposal has been validated in a UAV attitude planning application using the simulator *FlightGear 2018*, with the dynamic model of the fixed-wing airplane *Kadett 2400* Velasco-Carrau y col. 2015, depicted in Fig. 4.7. The system has 4 inputs to control the aerodynamic surfaces and speed: δ_e (elevator), δ_a (aileron), δ_r (rudder) and δ_{th} (throttle). Fig. 4.7 also shows the control structure used to drive the aircraft. Given some initial conditions (current tangent vector $\mathbf{T}$) and a target configuration (final tangent vector $\mathbf{T}^*$), a reference trajectory is generated in a local planning framework for heading-attitude control architectures, which can be used for obstacle avoidance.

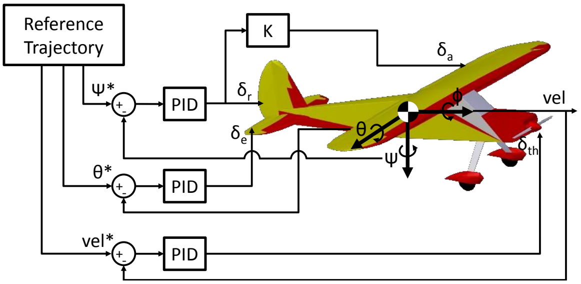
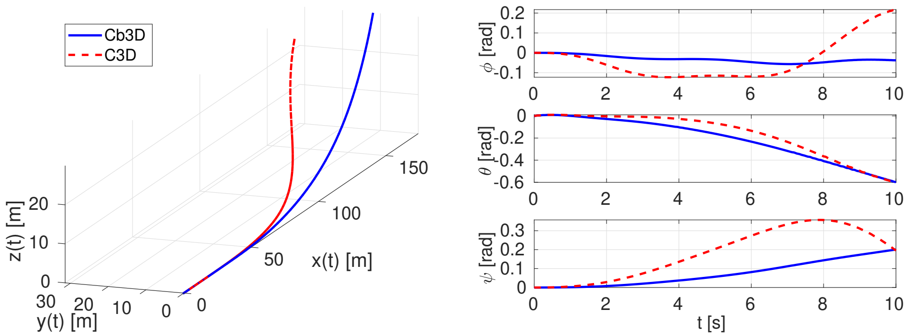


Figure 4.7: Model of Kadett 2400, with variables definition and control scheme.

The goal is to keep a constant velocity vel^* along the trajectory, which is accomplished by a PID controller that affects the throttle. Besides, the profiles θ^* and ψ^* of the generated trajectory are used as a reference input for two more PID controllers. The first PID controls the elevator, which affects the pitch orientation. Whilst the output of the second PID is applied to the rudder, which affects the yaw orientation and is used to compensate little errors when tracking the reference yaw ψ^*. Besides, the output of such PID controller is multiplied by a gain K to control the aileron, which directly affects the roll orientation, but also makes the aircraft turn in yaw.

Some results of the experiments carried out in FlightGear simulator are shown in Figs. 4.8 and 4.9. The reference velocity is set to $vel^* = 18$ m/s, with a sampling period of $T_s = 20$ ms to control the airplane. Since the time to perform the maneuver is set to $t = 10$ s, the length of the reference trajectory is $s = 180$ m. For this particular case, the target tangent vector $\mathbf{T}^*$ is defined by final angles pitch $\theta^* = -0{,}6$ rad and yaw $\psi^* = 0{,}2$ rad.

Figure 4.8(a) shows that the aircraft follows a completely different trajectory depending on the curve used as reference. In Fig. 4.8(b) it can be seen that, in both cases, the UAV starts at $t = 0$ s with initial roll $\varphi(0) = 0$, pitch $\theta(0) = 0$ and yaw $\psi(0) = 0$ (relative to its current local frame, used as reference coordinate system for the planning task). The Cb3D reference (blue line) is much smoother and more intuitive, since it avoids unnecessary turnings. The main drawback of using a C3D as reference (dashed red line) is the fact that in order to follow such a curve the airplane has to overshoot and turn excessively in yaw, and then compensate by counter turning to reach the target orientation, using roll to stabilize itself. This problem is intrinsic of pure 3D clothoid, since it is a spiral with some particular properties and limitations. On the contrary, all these problems are avoided by the way a clothoid-based 3D curve is generated, because for any target configuration both orientation and position increase monotonically and smoothly, which helps fixed-wing planes track such a reference.

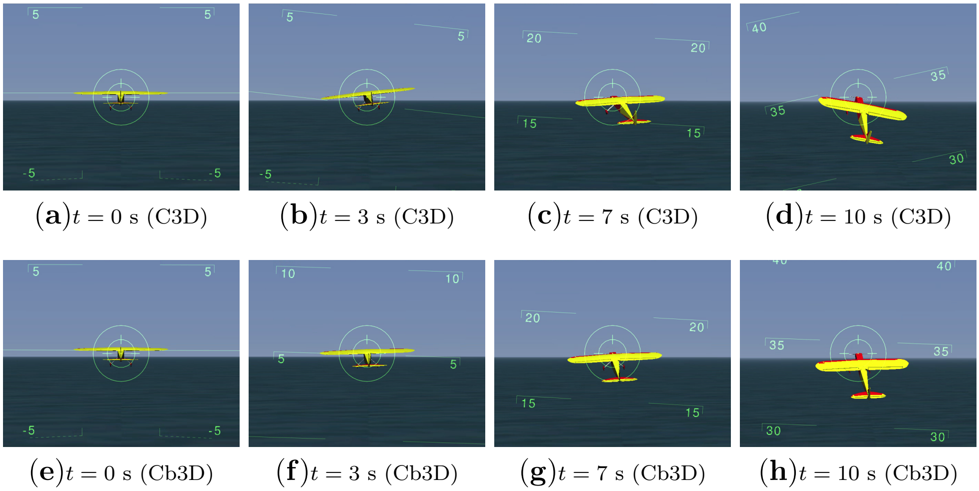


(a) Trajectory **(b)** Pitch and yaw angles

Figure 4.8: Trajectories followed by the UAV with different references to reach final orientation $\theta^* = -0{,}6$ rad and $\psi^* = 0{,}2$ rad: Cb3D (blue) vs. C3D (dashed red).

Figure 4.9 depicts a sequence of screenshots of the previous simulation at different time instants $t = \{0, 3, 7, 10\}$ s. It can be seen that the UAV starts at $t = 0$ s completely horizontal, moving straight in both cases (see Figs. 4.9(a) and 4.9(e)). Then, it starts to turn to the left by rotating in roll (see Figs. 4.9(b) and 4.9(f)), while progressively increasing its height (see Figs. 4.9(c) and 4.9(g)). The last pair of images (see Figs. 4.9(d) and 4.9(h)) show the UAV in the last configuration at $t = 10$ s, with pitch $\theta(10) = -0{,}6$ rad and yaw $\psi(10) = 0{,}2$ rad. Even though both trajectories guide the vehicle to the same target tangent vector, notice that the C3D planning forces the vehicle to turn much more in roll and yaw, than for the case of Cb3D planning. This unnecessary turning might difficult the control task and produce some undesired oscillatory or even unstable behavior in the airplane. For instance, UAV's final attitude for the C3D reference has a positive roll (see Figs. 4.8(b) and 4.9(d)), which makes the airplane turn to the right, but the target yaw was a positive value so the turning tendency should be to the left. This situation is one of the potential cases where a Cb3D curve outperforms a pure 3D clothoid in motion planning problems for fixed-wing UAVs.

The previous example and some more target orientations are shown and explained in the following video. The first case is for a final orientation $\theta^* = 0$ rad and $\psi^* = \pi$ rad, where both planners generate the same reference, a planar clothoid, so the UAV traces the same trajectory both times. The second example is for target angles $\theta^* = -0{,}2$ rad and $\psi^* = 0{,}6$ rad, where the trajectories followed by the airplane are not exactly the same, although they have similar profiles of roll, pitch and yaw. The third case, for $\theta^* = -0{,}6$ rad and $\psi^* = 0{,}2$ rad, has been already explained before in detail. Finally, the fourth example is for target orientation $\theta^* = -0{,}9$ rad and $\psi^* = 0$ rad, and it can be seen that for high values of θ^* with low ψ^* the C3D planner generates poor references, whilst the Cb3D trajectories generated by the planner are smoother and very intuitive, since they increase monotonically in both position and orientation, which helps the low-level controller track the reference, guiding and stabilizing easily the flying vehicle. So, as a conclusion, the lower the value of θ^* the more similar C3D and Cb3D trajectories are. On the contrary, the higher the value of θ^* the more different C3D and Cb3D trajectories are, being the Cb3D curve more intuitive and easy to follow from UAV's navigation perspective.



(a) $t = 0$ s (C3D) **(b)** $t = 3$ s (C3D) **(c)** $t = 7$ s (C3D) **(d)** $t = 10$ s (C3D)

(e) $t = 0$ s (Cb3D) **(f)** $t = 3$ s (Cb3D) **(g)** $t = 7$ s (Cb3D) **(h)** $t = 10$ s (Cb3D)

Figure 4.9: Screenshots of the simulation in FlightGear with final orientation $\theta^* = -0{,}6$ rad and $\psi^* = 0{,}2$ rad, using different references: C3D (a-d) and Cb3D curve (e-h).

4.6 Conclusions

Navigation of unmanned aerial vehicles (UAVs) and autonomous underwater vehicles (AUVs) requires generation of reference trajectories that need to be achievable and fast to compute. Space curves can be used to control such vehicles, but most of them are not very intuitive and/or require optimization procedures, which might be not feasible for real-time applications. Specially for the case of an airplane moving forward that needs to turn in order to change its attitude and altitude, for instance to avoid a collision.

A new space curve, coined as clothoid-based three-dimensional curve (Cb3D), has been presented as a composition of two planar clothoids. As proven in the paper, the proposed Cb3D curve can be seen as a generalization of other curves such as a straight line, a circular arc, a circular helix or a planar clothoid. They can be scaled and symmetric curves can be easily computed too. More importantly, it has been shown that the Cb3D curve is C^∞ smooth, and its position and orientation vectors are monotonic increasing. This property is very helpful for motion-planning, specially in obstacle avoidance problems. In fact, the shape of a Cb3D curve avoids unnecessary turnings, allowing its inclusion as a primitive for more complex path planning problems, where arbitrary initial and final configurations (position and orientation) could be joined smoothly.

Cb3D curves have been compared to pure 3D clothoids (C3D). Our results show that Cb3D curves can be several order of magnitude faster, because they can be approximated in a closed-form with high accuracy and can be computed analytically, which has benefits in terms of computation time for real-time planning. Besides, in the problem of joining two arbitrary orientations in the space, most of geometric and kinematic properties tend to be lower for Cb3D than for C3D, both in mean and maximum values. Besides, since the solution is unique, a Cb3D curve is not dependent on parameter initialization, contrary to Bézier or B-spline curves, and does not require optimization procedures as 3D clothoids and parametric curves.

Contrary to C3D, in the case of a Cb3D curve its curvature and torsion evolve non-linearly with respect to curve length. This might look inappropriate from the point of view of differential geometry, but from the perspective of planning and navigation of UAVs and AUVs it is not that relevant, as it has been shown. In addition to this, we used the FlightGear simulator to show some of the benefits of our proposal in motion-planning methods, since the generated trajectory is smooth and has not overshoot on the orientation.

Funding Sources

This work was supported by Generalitat Valenciana under the postdoctoral grant APOSTD/2017/055. The authors are also grateful to the financial support of Spanish Ministry of Economy and European Union, grant DPI2016-81002-R (AEI/FEDER, UE).

Capítulo 5

Elementary Clothoid-based 3D curve for Unmanned Aerial Vehicles

This chapter performs a mathematical analysis of a new Clothoid 3D curve, which is based on the Clothoid-based 3D Curve (Cb3D), Different properties are highlighted, which show a good applicability towards a Smooth 3D Path Planning.

Chapter Note: The contents of this chapter can be found in the paper:

> Armesto, L., **Vanegas, G**., & Girbés-Juan, V. (2022). Elementary Clothoid-Based Three-Dimensional Curve for Unmanned Aerial Vehicles. Journal of Guidance, Control, and Dynamics, 45(12), 2421-2431. JCR: Q1, Impact Factor: 2.486, SJR: Q1, Impact Factor: 1.34, DOI: https://doi.org/10.2514/ 1.G006935.

5.1 Introduction

Unmanned Aerial Vehicles (UAVs), either multi-rotor or fixed-wing UAVs, can be used in many fields to solve complex problems in safety [Afman y col. 2018], communications [Baek y Lim 2018; Varshosaz y col. 2020], military applications [Hartmann y Giles 2016; Orfanus, Freitas y Eliassen 2016], civilian applications [Mualla y col. 2019; Löbl y col. 2021], protection of nuclear plants [Islam, Ahmed e Islam 2018], energy efficiency [Liu y col. 2018], nonlinear control [Mo y Farid 2019; Nicotra, Naldi y Garone 2017] and path planning [Roberge, Tarbouchi y Labonte 2018], among others. Vertical Take Off and Landing (VTOL) is by far the most used UAV's configuration [Finger, Braun y Bil 2017]. Probably the main reason is that, nowadays, there are lots of low cost multi-rotor models. VTOL strategies can be applied to fixed-wing airplanes [Çakici y Leblebicioğlu 2016], however in normal operation they usually fly horizontally to keep a constant altitude [Liu y Chen 2016; Guclu, Kurtulus y Arikan 2016], but they need to perform changes in their heading and altitude very often, specially in cluttered environments [Floreano y col. 2017].

One approach to deal with navigation problems of UAVs is from the point of view of optimization and control theory. In [Hong y col. 2019], a model predictive convex programming (MPCP) for constrained vehicle guidance was developed. Compared to model predictive static programming methods, it offers the advantage of considering state and input constraints as well as choosing the cost function. The same authors have recently used trigonometric series for UAV path planning [Hong y col. 2021]. In this work a computationally efficient trajectory is generated for smooth aircraft flight level changes. It uses convex-optimization based on linearizations of the dynamics.

A second approach to solve navigation applications is from the trajectory planning point of view. In [Wang y col. 2017b], the authors proposed a curvature continuous and bounded path planning method for UAVs considering kinematic and dynamic constraints such as maximum velocity and curvature. Indeed, they proposed the use of Bézier curves as transition curves, but this type of spatial curve has no physical meaning and it is hard to get the relationship between its parameters and vehicle's physics (kinematics and dynamics). Pythagorean Hodographs are also well extended for UAV planning, but they require to solve an optimization problem in order to take constraints into account [Neto, Macharet y Campos 2013; Vinokursky, Mezentceva y Samoylov 2020]. Among other spatial curves, clothoids (also known as Euler or Cornu Spirals) play an important role, because of their "nice" geometric properties, including a close relation between physical phenomena (normal acceleration and jerks) with the clothoid scaling parameter. Therefore, borrowing the idea from the planar movement, spatial clothoids could be used in Path Planning [Celik, Yayli y Guler 2016; Girbés, Vanegas y Armesto 2019].

Clothoids have been studied deeply in ground mobile robotics [Fraichard y Scheuer 2004a; Alia y col. 2015; Girbés, Armesto y Tornero 2014; Lima y col. 2015; Girbés, Vanegas y Armesto 2019; Arshad y col. 2020]. For instance, in [Kim y col. 2017; Sedighi, Nguyen y Kuhnert 2019; Horváth y Pozna 2021] the authors showed that the use of clothoids as transition curves not only guaranteed continuous curvature (G^2 continuity) but also a bound on its derivative, the sharpness. So, when the tracking speed is constant, the normal acceleration is bounded if the maximum curvature and its derivative are bounded. In order to increase driving safety and comfort in road design, many studies have been done to determine appropriate values for clothoid sharpness in transition curves [Wilde 2009; Gim y col. 2017b]. Indeed, clothoids are usually used in highway, railway and roller coasters design [McCrae y Singh 2008], among other applications. Recent works proved that including comfort and safety requirements in the path planning process can avoid tip-overs in forklifts carrying heavy loads at high speeds [Girbés, Armesto y Tornero 2014; Lima y col. 2015].

Even though the benefits of planar clothoids have been proven, researchers have paid little attention to the use of such curves in the 3D space [Xiao y col. 2020]. A few works proposed the use of clothoids in path planning for UAVs [Wan y col. 2011]. Others instead, approached a similar problem from the field of computer graphics [Celik, Yayli y Guler 2016]. Nevertheless, only one seems to match with the exact definition of a 3D clothoid [Harary y Tal 2012a]. But, the problem is that this approach is based on a Frenet-Serret frame and requires a numeric integration to be computed. Moreover, since they are generated from the linear equations of curvature and torsion, it is not possible to know *a priori* the increment in position and orientation with respect to the origin of the curve.

A new spatial clothoid-based 3D curve is proposed in this Note. The new curve is built with two clothoid-based three-dimensional curves (Cb3D) [Girbés, Vanegas y Armesto 2019] and inherits many of their interesting properties, such as smoothness, scaling and symmetry, for path planning methods. The main contribution with respect to our previous work, with the new curve definition, we can join line segments pointing towards arbitrary target directions. Thus, it can be used as a transition curve for many path planning methods for UAVs [Fu y col. 2018]. In addition to this, we show how to compute the parameters of the new curve, and analyze the conditions in which the monotonicity property holds. It is important to remark that constraints on the maximum sharpness of curvature and torsion are considered in this approach. In summary, Cb3D curves represent the tool on which our new proposal is based on, while this Note proposes the proper combination of two of them to actually build a transition curve with application to UAVs, resembling a 3D extension to Elementary paths [Scheuer y Fraichard 1996a] used in car-like vehicles.

The note is organized as follows. Section 5.2 provides some preliminaries as well as states the problem to be solved. In Section 5.3 the proposed curve is defined, and it shows curve properties such as smoothness, scalability, and symmetry, among others. Section 5.4 conducts an analysis of the target angles region where the curve increases monotonically in position and orientation. Section 5.5 describes the proposed design methodology to obtain the shortest possible curve satisfying parameter constraints and shows a case study, a comparison with 3D pure clothoids and its application to fixed-wing UAVs in orientation heading and altitude change problem. At the end of the Note, we discuss the application and advantages of the proposed method in Section 5.6 and draw some conclusions in Section 5.7.

5.2 Preliminaries

5.2.1 Clothoid-Based Three-Dimensional Curve (Cb3D)

In Girbés, Vanegas y Armesto 2019 a new clothoid-based three-dimensional curve (Cb3D) is defined as a composition of two planar clothoids. The authors proposed that each clothoid is contained in a plane orthogonal to each other (clothoids are defined in planes XY and XZ). The clothoid contained in plane XY is indeed a clothoid whose length depends on the clothoid on plane XZ, curve used as a reference trajectory to control a fixed-wing UAV.

Let $\mathbf{C}_1(q,\mu) := [\mathcal{C}(q,\mu)\ \ \mathcal{S}(q,\mu)]^T \in \mathbb{R}^2$ be a planar clothoid contained in plane XY with arc-length q, being μ the clothoid's curvature sharpness; and $\mathbf{C}_2(w,\rho) := [\mathcal{C}(w,\rho)\ \ \mathcal{S}(w,\rho)]^T \in \mathbb{R}^2$ another planar clothoid contained in plane XZ with arc-length w and torsion sharpness ρ, both defined by Fresnel integrals $\mathcal{C}(s,\sigma) = \int_0^s \cos(\frac{\sigma}{2}\xi^2)\,d\xi$ and $\mathcal{S}(s,\sigma) = \int_0^s \sin(\frac{\sigma}{2}\xi^2)\,d\xi$.

A Cb3D curve can be generated as a composition of both planar clothoids Girbés, Vanegas y Armesto 2019 as follows:

$$\mathbf{C}(s,\mu,\rho) := \left[\mathcal{C}(\mathcal{C}(s,\rho),\mu) \quad \mathcal{S}(\mathcal{C}(s,\rho),\mu) \quad -\mathcal{S}(s,\rho)\right]^T, \tag{5.1}$$

while its pitch $\theta(s,\rho)$ and yaw $\Psi(s,\mu,\rho)$ angles can be computed as

$$\theta(s,\rho)=\frac{\rho}{2}s^2,\quad \Psi(s,\mu,\rho)=\frac{\mu}{2}\mathcal{C}^2(s,\rho), \tag{5.2}$$

and its tangent vector is

$$\mathbf{T}(s,\mu,\rho):=\begin{bmatrix}\cos(\Psi(s,\mu,\rho))\cos(\theta(s,\rho)) & \sin(\Psi(s,\mu,\rho))\cos(\theta(s,\rho)) & -\sin(\theta(s,\rho))\end{bmatrix}^T. \tag{5.3}$$

Thus, given an arbitrary target tangent vector $\mathbf{T}^*$, defined by final angles $\theta^*\in[-\pi/2,\pi/2]$ rad and $\Psi^*\in[-\pi,\pi]$ rad, and arc-length $s>0$, there is only one pair of parameters μ^* and ρ^* that synthesize a Cb3D curve [Girbés, Vanegas y Armesto 2019]:

$$\rho^*=\frac{2\theta^*}{s^2} \tag{5.4}$$

$$\mu^*=\frac{2\Psi^*}{\mathcal{C}^2(s,\rho^*)} \tag{5.5}$$

5.2.2 Problem Statement

The aim is to generate a path $\mathbf{E}(s,\mathbf{p})$, where $\mathbf{p}=\{\mu,\ \rho\}$ are curve specific parameters (curvature and torsion sharpness) and $s\in[0,S]$ is the curve length. Parameters $\mathbf{p}$ are assumed to be constrained, i.e.: $\mu\in[-\mu_{max},\mu_{max}]$ and $\rho\in[-\rho_{max},\rho_{max}]$. For a given initial configuration with null curvature and torsion, the path must point towards a given relative direction $\mathbf{T}^*$, that is, $\mathbf{T}^*=\frac{\mathbf{E}'(S,\mathbf{p})}{\|\mathbf{E}(S,\mathbf{p})\|}$. Without loss of generality, it can be assumed that the UAV is centered at the origin and its initial orientation is also zero (local and global coordinate frames are coincident), otherwise we can always apply a transformation to obtain a local target direction. We also assume that a low-level controller ensures that the UAV follows that path and thus, the design of such low-level controller is out of the scope of this note.

5.3 Elementary Clothoid-based 3D curve: Definition and Properties

In the planar case, the concept of *Elementary Path* was introduced by Fraichard and Scheuer y Fraichard 1996a, as a combination of two symmetrical 2D clothoids with the same sharpness and length. In this sense, the work presented in this note extends this idea to 3D space, combining two Cb3D curves. An Elementary Clothoid-based 3D curve (ECb3D or Elementary for short) can be defined, for $s \in [-S, S]$ and $\tilde{s} > 0$ as:

$$\mathbf{E}(s, \tilde{s}, \mathbf{p}) := \begin{cases} \mathbf{C}(s, \mathbf{p}) & if\ |s| \leq \tilde{s} \\ \tilde{\mathbf{C}}(s, \tilde{s}, \mathbf{p}) & \text{otherwise} \end{cases} \tag{5.6}$$

with $\tilde{s}$ the length of each Cb3D curve (referred here as *primary* and *secondary* curves), S is the overall Elementary length and thus, by definition, $S := 2\tilde{s}$, and Cb3D sharpness parameters are $\mathbf{p} = \{\mu, \rho\}$. The *secondary* Cb3D is indeed a rotated and translated Cb3D curve as follows:

$$\tilde{\mathbf{C}}(s, \tilde{s}, \mathbf{p}) = \mathbf{R}_{\mathbf{T}(\tilde{s},\mathbf{p})}(\pi) \left(\mathbf{C}(s - \operatorname{sign}(s)2\tilde{s}, \mathbf{p}) + \operatorname{sign}(s)\mathbf{C}(\tilde{s}, \mathbf{p})\right) + \operatorname{sign}(s)\mathbf{C}(\tilde{s}, \mathbf{p}), \tag{5.7}$$

being $\mathbf{R}_{\mathbf{T}(\tilde{s},\mathbf{p})}(\pi)$ a rotation of π around vector $\mathbf{T}(\tilde{s}, \mathbf{p})$, that is, based on Rodrigues formula [Dai 2015] we have:

$$\begin{aligned} \mathbf{R}_{\mathbf{T}(\tilde{s},\mathbf{p})}(\pi) &= \cos\pi \mathbf{I}_{3\times3} + (1 - \cos\pi)\mathbf{T}(\tilde{s}, \mathbf{p})\mathbf{T}^T(\tilde{s}, \mathbf{p}) + \sin(\pi) \left[\mathbf{T}(\tilde{s}, \mathbf{p})_\times\right] \\ &= 2\mathbf{T}(\tilde{s}, \mathbf{p})\mathbf{T}^T(\tilde{s}, \mathbf{p}) - \mathbf{I}_{3\times3}, \end{aligned} \tag{5.8}$$

being $\left[\mathbf{T}(\tilde{s}, \mathbf{p})_\times\right]$ a skew-symmetric matrix.

Figure 5.1 shows a representative ECb3D curve with its 4 possible cases, i.e., combinations of positive and negative values of the clothoid parameters. The *primary clothoid* $\mathbf{C}(s, \mathbf{p})$, painted in solid line, is a Cb3D curve evaluated for $|s| \leq \tilde{s}$ and it can be computed for positive or negative s, while $\tilde{s}$ is the length of such primary clothoid. On the other hand, the *secondary clothoid*, $\tilde{\mathbf{C}}(s, \tilde{s}, \mathbf{p})$, painted in a dashed line, is another Cb3D curve properly rotated and translated to ensure position continuity as well as coincident tangent vectors for $s = \tilde{s}$, i.e. the curve has G^1 continuity by definition. Note that in Fig. 5.1 the colored frame axis at the origin have been depicted for reference. By convenience, and abusing of notation, we will drop the last arguments of expressions if it is clear from the context, i.e.: $\mathbf{E}(s, \tilde{s}, \mathbf{p})$ might be expressed as $\mathbf{E}(s, \tilde{s})$ or even $\mathbf{E}(S)$ for the particular case when $s = S = 2\tilde{s}$. Similarly, $\mathbf{C}(s, \mathbf{p})$ can be expressed as $\mathbf{C}(s)$.

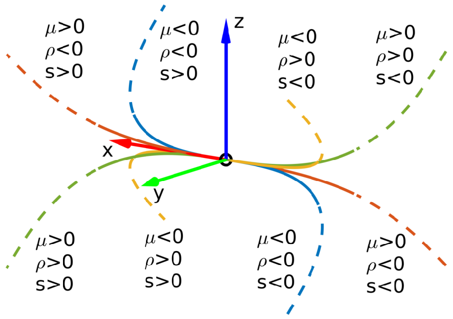


Figure 5.1: Elementary Clothoid-based 3D curves, all 4 possible cases based on clothoid parameters, with *Primary clothoids* in solid lines and *secondary clothoids* in dashed lines.

Lemma 5.3.1 (Smoothness) *An Elementary clothoid-based 3D curve is C^∞ smooth with $\mu \neq 0$ or $\rho \neq 0$ for $s \in [-2\tilde{s}, 2\tilde{s}]$ and $\tilde{s} > 0$. Indeed, the n-th derivative can be computed as a piece-wise function as follows:*

$$\mathbf{E}^{(n)}(s, \tilde{s}, \mathbf{p}) := \begin{cases} \mathbf{C}^{(n)}(s, \mathbf{p}) & \textit{if } |s| \leq \tilde{s} \\ \mathbf{R}_{\mathbf{T}(\tilde{s},\mathbf{p})}(\pi)\mathbf{C}^{(n)}(s - 2\tilde{s}, \mathbf{p}) & \textit{otherwise} \end{cases}, \tag{5.9}$$

where $\mathbf{C}^{(n)}(s, \mathbf{p})$ is a vector with trigonometric expressions:

$$\mathbf{C}^{(n)}(s, \mathbf{p}) = \begin{bmatrix} \tilde{d}_{1,n} \cos\Psi(s)\cos\theta(s) + \tilde{d}_{2,n} \sin\Psi(s)\cos\theta(s) \\ \hfill + \tilde{d}_{3,n} \cos\Psi(s)\sin\theta(s) \\ \hfill + \tilde{d}_{4,n} \sin\Psi(s)\sin\theta(s) \\ \bar{d}_{1,n} \cos\Psi(s)\cos\theta(s) + \bar{d}_{2,n} \sin\Psi(s)\cos\theta(s) \\ \hfill + \bar{d}_{3,n} \cos\Psi(s)\sin\theta(s) \\ \hfill + \bar{d}_{4,n} \sin\Psi(s)\sin\theta(s) \\ \hat{d}_{1,n} \sin\theta(s) + \hat{d}_{2,n} \cos\theta(s) \hfill \end{bmatrix}, \tag{5.10}$$

for some polynomials $\tilde{d}_{1,n}$, $\tilde{d}_{2,n}$, $\tilde{d}_{3,n}$, $\tilde{d}_{4,n}$, $\bar{d}_{1,n}$, $\bar{d}_{2,n}$, $\bar{d}_{3,n}$, $\bar{d}_{4,n}$, $\hat{d}_{1,n}$ and $\hat{d}_{2,n}$; and pitch and yaw angles are given by Eq. (5.2), where parameters μ and ρ are omitted to simplify notation.

Proof 5.3.1 *The proof is made for $s > 0$, but equivalent proof can be made for $s < 0$. A C^∞ function is differentiable for all degrees of differentiation; that is, its position vector can be derived ∞ times. Since an Elementary curve is a piece-wise function, its derivatives can be computed for $s < \tilde{s}$ and $s > \tilde{s}$ and recalling the smoothness property of a Cb3D curve in [Girbés, Vanegas y Armesto 2019], we can clearly see that $\mathbf{E}^{(n)}(s, \tilde{s}) = \mathbf{C}^{(n)}(s)$ is C^∞ for $s < \tilde{s}$ and also is C^∞ for $s > \tilde{s}$ due to $\mathbf{E}^{(n)}(s, \tilde{s}) = \tilde{\mathbf{C}}^{(n)}(s, \tilde{s})$, where $\tilde{\mathbf{C}}^{(n)}(s, \tilde{s}) = \mathbf{R}_{\mathbf{T}(\tilde{s})}(\pi)\mathbf{C}^{(n)}(s - 2\tilde{s})$ is a transformed Cb3D curve and thus the same property holds.*

So, to complete the proof we need to check that the following limits exist:

$$\lim_{s\to\tilde{s}^-} \mathbf{E}^{(n)}(s, \tilde{s}) = \lim_{s\to\tilde{s}^+} \mathbf{E}^{(n)}(s, \tilde{s}). \tag{5.11}$$

In this sense, we can see that $\lim_{s\to\tilde{s}^-} \mathbf{E}^{(n)}(s, \tilde{s}) = \mathbf{C}^{(n)}(\tilde{s}, \tilde{s})$ and on the other hand:

$$\lim_{s\to\tilde{s}^+} \mathbf{E}^{(n)}(s, \tilde{s}) = \lim_{s\to\tilde{s}^+} \frac{\tilde{\mathbf{C}}^{(n-1)}(s, \tilde{s}) - \tilde{\mathbf{C}}^{(n-1)}(\tilde{s}, \tilde{s})}{s} = \mathbf{C}^{(n)}(\tilde{s}), \tag{5.12}$$

because $\lim_{s\to\tilde{s}^+} \frac{\tilde{\mathbf{C}}^{(n-1)}(s,\tilde{s}) - \tilde{\mathbf{C}}^{(n-1)}(\tilde{s},\tilde{s})}{s} = \mathbf{R_T} \lim_{s\to\tilde{s}^+} \frac{\mathbf{C}^{(n-1)}(s-2\tilde{s}) - \mathbf{C}^{(n-1)}(\tilde{s}-2\tilde{s})}{s} = \mathbf{R_T}\mathbf{C}^{(n)}(-\tilde{s})$. Noting that $\theta(-\tilde{s}) = \theta(\tilde{s})$, $\Psi(-\tilde{s}) = \Psi(\tilde{s})$ and $\theta^{(n)}(-\tilde{s}) = -\theta^{(n)}(\tilde{s})$, $\Psi^{(n)}(-\tilde{s}) = -\Psi^{(n)}(\tilde{s})$ for $n = 1, 3, \ldots$, while $\theta^{(n)}(-\tilde{s}) = \theta^{(n)}(\tilde{s})$, $\Psi^{(n)}(-\tilde{s}) = \Psi^{(n)}(\tilde{s})$ for $n = 2, 4, \ldots$; and the fact that coefficients in Eq. (5.10) depend on polynomials of such derivatives of order $n - 1$ that are affected in such a way that also $\tilde{d}_{1,n}(-\tilde{s}) = -\tilde{d}_{1,n}(\tilde{s})$, $\tilde{d}_{2,n}(-\tilde{s}) = -\tilde{d}_{2,n}(\tilde{s})$, etc... for $n = 1, 3, \ldots$ and $\tilde{d}_{1,n}(-\tilde{s}) = \tilde{d}_{1,n}(\tilde{s})$, $\tilde{d}_{2,n}(-\tilde{s}) = \tilde{d}_{2,n}(\tilde{s})$, etc... for $n = 2, 4, \ldots$. As a consequence, we can express $\mathbf{C}^{(n)}(-\tilde{s}) = (-1)^{n-1}\mathbf{C}^{(n)}(\tilde{s})$ and therefore, $\mathbf{R_T}\mathbf{C}^{(n)}(-\tilde{s}) = (-1)^{n-1}\mathbf{R_T}\mathbf{C}^{(n)}(\tilde{s}) = \mathbf{C}^{(n)}(\tilde{s})$, because $\mathbf{C}^{(n)}(\tilde{s}) \parallel \mathbf{T}(\tilde{s})$ for $n = 1, 3, \ldots$ and $\mathbf{C}^{(n)}(\tilde{s}) \perp \mathbf{T}(\tilde{s})$ for $n = 2, 4, \ldots$ and thus $\mathbf{R_T}$ does not affect to cases $n = 1, 3, \ldots$ and changes the sign of $\mathbf{C}^{(n)}(\tilde{s})$ for $n = 2, 4, \ldots$, additional details have been omitted for brevity ■.

Thus, Lemma 5.3.1 states that the proposed curve is smooth, that is, their derivatives are differentiable. This is a desirable property in many path planning methods to ensure that control actions are also smooth [Wang y col. 2017b]. In [Girbés, Vanegas y Armesto 2019], the authors show that some kinematic-based and dynamic-based metrics such as linear and angular velocities, linear and angular accelerations, jerks, etc. are improved using this type of curves.

Lemma 5.3.2 (Symmetry and Scalability) *An Elementary clothoid-based 3D curve is a symmetric and a scalable curve with scaling factor $\lambda > 0$ as follows:*

$$\mathbf{E}(s, \tilde{s}, \mathbf{p}) = -\mathbf{E}(-s, \tilde{s}, \mathbf{p}) \tag{5.13}$$

$$\lambda\mathbf{E}(s, \tilde{s}, \mathbf{p}) = \mathbf{E}(\lambda s, \lambda\tilde{s}, \lambda^{-2}\mathbf{p}). \tag{5.14}$$

Proof 5.3.2 *Let's recall the symmetry property of a Cb3D curve [Girbés, Vanegas y Armesto 2019], that is $\mathbf{C}(s, \mathbf{p}) = -\mathbf{C}(-s, \mathbf{p})$, we have:*

$$\begin{aligned} -\mathbf{E}(-s, \tilde{s}, \mathbf{p}) &= \left\{ \begin{array}{cc} -\mathbf{C}(-s, \mathbf{p}) & if\ |s| \le \tilde{s} \\ -\tilde{\mathbf{C}}(-s, \tilde{s}, \mathbf{p}) & otherwise \end{array} \right. \\ &= \left\{ \begin{array}{cc} \mathbf{C}(s, \mathbf{p}) & if\ |s| \le \tilde{s} \\ \tilde{\mathbf{C}}(s, \tilde{s}, \mathbf{p}) & otherwise \end{array} \right. = \mathbf{E}(s, \tilde{s}, \mathbf{p}). \end{aligned} \tag{5.15}$$

Now, recalling the scalability property of a Cb3D curve in [Girbés, Vanegas y Armesto 2019], a Cb3D curve is scalable as follows $\lambda\mathbf{C}(s, \mathbf{p}) = \mathbf{C}(\lambda s, \lambda^{-2}\mathbf{p})$. Thus,

$$\begin{aligned} \lambda\mathbf{E}(s, \tilde{s}, \mathbf{p}) &= \left\{ \begin{array}{cc} \lambda\mathbf{C}(s, \mathbf{p}) & if\ |s| \le \lambda\tilde{s} \\ \lambda\tilde{\mathbf{C}}(s, \lambda\tilde{s}, \mathbf{p}) & otherwise \end{array} \right. \\ &= \left\{ \begin{array}{cc} \mathbf{C}(\lambda s, \lambda^{-2}\mathbf{p}) & if\ |s| \le \lambda\tilde{s} \\ \tilde{\mathbf{C}}(\lambda s, \lambda\tilde{s}, \lambda^{-2}\mathbf{p}) & otherwise \end{array} \right. \\ &= \mathbf{E}(\lambda s, \lambda\tilde{s}, \lambda^{-2}\mathbf{p})\blacksquare. \end{aligned} \tag{5.16}$$

The scalability property of Lemma 5.3.2 allows us to design a curve for the unitary case $\tilde{s} = 1$ and then scale it, if necessary, by multiplying with the factor λ (to extend its length, which implies that curve parameters are indeed multiplied with a factor $\mu \leftarrow \lambda^{-2}\mu$ and $\rho \leftarrow \lambda^{-2}\rho$). As a consequence, the unitary curve can be approximated with a look-up table (LUT) up to a desired accuracy and evaluated in a *zero-cost* time. On the other hand, the symmetry property of Lemma 5.3.2 allows us to implement only the half of the LUT, to save some memory resources.

Lemma 5.3.3 (Tangent vector) *Tangent vector $\mathbf{T}_E(s, \tilde{s}, \mathbf{p})$ can be computed as follows:*

$$\mathbf{T}_E(s, \tilde{s}, \mathbf{p}) := \begin{cases} \mathbf{T}(s, \mathbf{p}) & \textit{if } |s| \leq \tilde{s} \\ \underbrace{\mathbf{R}_{\mathbf{T}(\tilde{s},\mathbf{p})}(\pi)\mathbf{T}(s - 2\tilde{s}, \mathbf{p})}_{\tilde{\mathbf{T}}(s,\tilde{s},\mathbf{p})} & \textit{otherwise} \end{cases}. \tag{5.17}$$

In particular, the tangent vector evaluated at the end of the curve, that is $\mathbf{T}_E(\pm S, \rho, \mu)$ with $S = 2\tilde{s}$, can be computed as:

$$\begin{aligned} \mathbf{T}_E(\pm S, \rho, \mu) &= \begin{bmatrix} 2\cos^2(\theta(\tilde{s}, \rho))\cos^2(\Psi(\tilde{s}, \rho, \mu)) - 1 \\ \cos^2(\theta(\tilde{s}, \rho))\sin(2\Psi(\tilde{s}, \rho, \mu)) \\ -\sin(2\theta(\tilde{s}, \rho))\cos(\Psi(\tilde{s}, \rho, \mu)) \end{bmatrix} \\ &= \begin{bmatrix} 2\cos^2(\frac{\rho\tilde{s}^2}{2})\cos^2(\frac{\mu}{2}\mathcal{C}^2(\tilde{s}, \rho)) - 1 \\ \cos^2(\frac{\rho\tilde{s}^2}{2})\sin(\mu\mathcal{C}^2(\tilde{s}, \rho)) \\ -\sin(\rho\tilde{s}^2)\cos(\frac{\mu}{2}\mathcal{C}^2(\tilde{s}, \rho)) \end{bmatrix}. \end{aligned} \tag{5.18}$$

Proof 5.3.3 *Tangent vector can be obtained from the vector corresponding to the first derivative of Eq.* (5.9)*. It's unitary expression is simply computed as $\mathbf{T}(s, \mathbf{p}) = \frac{\mathbf{C}'(s,\mathbf{p})}{\|\mathbf{C}'(s,\mathbf{p})\|}$ for $|s| \leq \tilde{s}$ and for $|s| > \tilde{s}$ the tangent vector is computed as $\tilde{\mathbf{T}}(s, \tilde{s}, \mathbf{p}) := \frac{\tilde{\mathbf{C}}'(s,\tilde{s},\mathbf{p})}{\|\tilde{\mathbf{C}}'(s,\tilde{s},\mathbf{p})\|} = \frac{\mathbf{R}_{\mathbf{T}(\tilde{s},\mathbf{p})}(\pi)\mathbf{C}'(s-2\tilde{s},\mathbf{p})}{\|\mathbf{R}_{\mathbf{T}(\tilde{s},\mathbf{p})}(\pi)\mathbf{C}'(s-2\tilde{s},\mathbf{p})\|} = \mathbf{R}_{\mathbf{T}(\tilde{s},\mathbf{p})}(\pi)\frac{\mathbf{C}'(s-2\tilde{s},\mathbf{p})}{\|\mathbf{C}'(s-2\tilde{s},\mathbf{p})\|} = \mathbf{R}_{\mathbf{T}(\tilde{s},\mathbf{p})}(\pi)\mathbf{T}(s - 2\tilde{s}, \mathbf{p})$.*

On the other hand, Eq. (5.7) *is indeed a transformed Cb3D curve, where the term $\mathbf{C}(s - \text{sign}(s)2\tilde{s}, \mathbf{p})$ is the piece of the* original *curve. It can clearly be seen that for $s = \pm 2\tilde{s}$ the tangent vector is indeed coincident with X-axis $\mathbf{T}(0) = \mathbf{X} = [1\ 0\ 0]^T$. Thus, the rotation of $\mathbf{X}$ around $\mathbf{T}(\tilde{s}) = [T_x(\tilde{s})\ T_y(\tilde{s})\ T_z(\tilde{s})]^T$ is the vector:*

$$\mathbf{T}_E(\pm 2\tilde{s}) = \mathbf{R}_{\mathbf{T}(\tilde{s})}(\pi)\mathbf{X} = \begin{bmatrix} 2T_x^2(\tilde{s}) - 1 & 2T_x(\tilde{s})T_y(\tilde{s}) & 2T_x(\tilde{s})T_z(\tilde{s}) \end{bmatrix}^T. \tag{5.19}$$

Therefore, replacing the elements of the tangent vector in Eq. (5.3) *gives:*

$$\mathbf{T}_E(\pm S, \rho, \mu) = \begin{bmatrix} 2\cos^2(\theta(\tilde{s}))\cos^2(\Psi(\tilde{s})) - 1 \\ \cos^2(\theta(\tilde{s}))\sin(2\Psi(\tilde{s})) \\ -\sin(2\theta(\tilde{s}))\cos(\Psi(\tilde{s})) \end{bmatrix}^T. \tag{5.20}$$

After some trivial trigonometric manipulations and substituting pitch $\theta(\tilde{s})$ and yaw $\Psi(\tilde{s})$ angles at the intermediate point, given by Eq. (5.2), we get to Eq. (5.18) $\blacksquare$.

Lemma 5.3.3 allows to compute the pointing direction of the curve. Particularly, Eq. (5.18) provides the direction that UAV following that curve will point given a set of parameters $\tilde{s}$, ρ and μ. Indeed, this is a supporting lemma for the following two lemmas, as well as for the design of the curve parameters.

Lemma 5.3.4 (Pitch and yaw angles) *Pitch $\Psi_E(s, \tilde{s}, \mathbf{p})$ and yaw $\theta_E(s, \tilde{s}, \mathbf{p})$ can be computed as follows:*

$$\begin{aligned} \theta_E(s, \tilde{s}, \mathbf{p}) &:= \begin{cases} \theta(s, \mathbf{p}) & if\ |s| \leq \tilde{s} \\ -\arcsin \tilde{T}_z & otherwise \end{cases}, \\ \Psi_E(s, \tilde{s}, \mathbf{p}) &:= \begin{cases} \Psi(s, \mathbf{p}) & if\ |s| \leq \tilde{s} \\ \arctan \frac{\tilde{T}_y}{\tilde{T}_x} & otherwise \end{cases}, \end{aligned} \tag{5.21}$$

where $\tilde{T}_x$, $\tilde{T}_y$ and $\tilde{T}_z$ are the elements of the tangent vector $\tilde{\mathbf{T}}(s, \tilde{,}\mathbf{p})$ in Lemma 5.3.3.

In particular, for intermediate angles $\theta(\tilde{s}) \in [-\frac{\pi}{4}, 0]$ and $\Psi(\tilde{s}) \in [0, \frac{\pi}{4}]$, yaw and pitch angles at the end of an ECb3D are:

$$\theta(\tilde{s}) \geq \theta_E(2\tilde{s}) := \arcsin\left(\cos\Psi(\tilde{s})\sin(2\theta(\tilde{s}))\right) \in [-\frac{\pi}{2}, 0] \tag{5.22}$$

$$\Psi(\tilde{s}) \leq \Psi_E(2\tilde{s}) := \arctan\left(\frac{\cos^2(\theta(\tilde{s}))\sin(2\Psi(\tilde{s}))}{2\cos^2(\theta(\tilde{s}))\cos^2(\Psi(\tilde{s})) - 1)}\right) \in [0, \frac{3\pi}{4}]. \tag{5.23}$$

Also, the inverse relation can be obtained. Given pitch $\theta_E(2\tilde{s}) \in [-\frac{\pi}{2}, 0]$ and yaw $\Psi_E(2\tilde{s}) \in [0, \frac{\pi}{2}]$ angles of an Elementary Clothoid-based 3D curve, the angles, that is $\theta(\tilde{s})$ and $\Psi(\tilde{s})$, of the primary Cb3D curve are:

$$\begin{aligned}\theta_E(2\tilde{s}) \leq \theta(\tilde{s}) &:= \arctan\left(\frac{\sin(\theta_E(2\tilde{s}))}{\sqrt{\cos^2(\theta_E(2\tilde{s})) + 2\cos(\theta_E(2\tilde{s}))\cos(\Psi_E(2\tilde{s})) + 1}}\right)\\ &\in [-\frac{\pi}{4}, 0]\end{aligned} \tag{5.24}$$

$$\begin{aligned}\Psi_E(2\tilde{s}) \geq \Psi(\tilde{s}) &:= \arctan\left(\frac{\cos(\theta_E(2\tilde{s}))\sin(\Psi_E(2\tilde{s}))}{\cos(\Psi_E(2\tilde{s}))\cos(\theta_E(2\tilde{s})) + 1}\right)\\ &\in [0, \frac{\pi}{4}]\end{aligned} \tag{5.25}$$

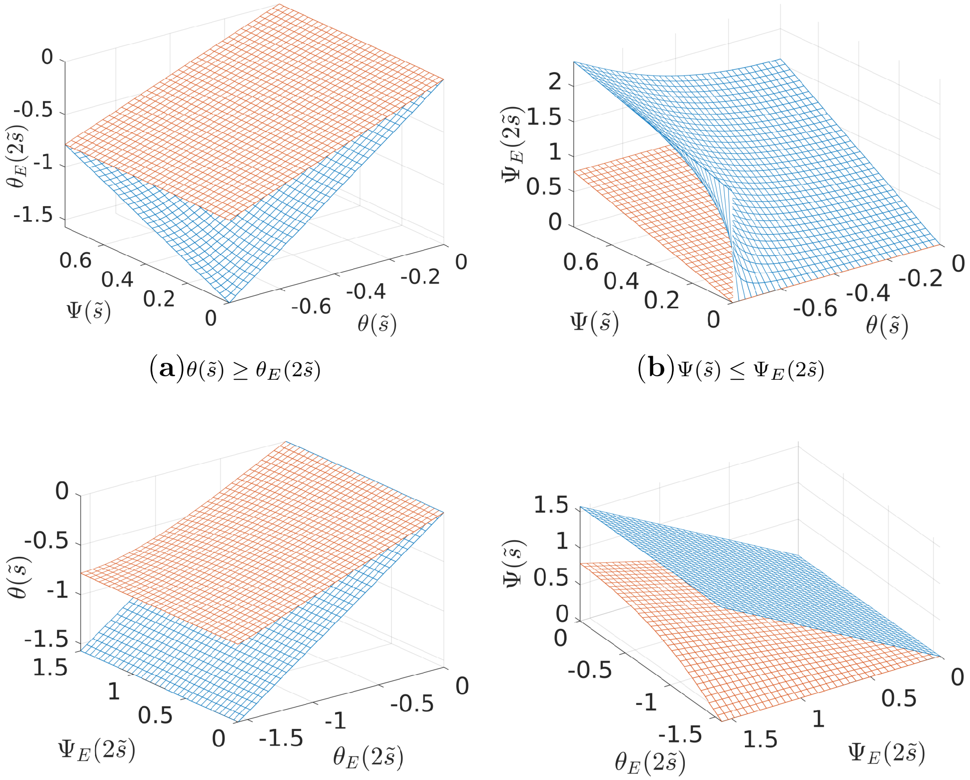
Proof 5.3.4 *The orientation in $\mathcal{S}^3$ defined by the* **TNB** *basis of any space curve in $\mathbb{R}^3$ can be defined by Euler angles roll $\phi_E(s,\tilde{s})$, pitch $\theta_E(s,\tilde{s})$, yaw $\Psi_E(s,\tilde{s})$, expressed as $\mathbf{R}_E(s,\tilde{s}) = \mathbf{R}_z(\Psi_E(s,\tilde{s}))\mathbf{R}_y(\theta_E(s,\tilde{s}))\mathbf{R}_x(\phi_E(s,\tilde{s}))$. The corresponding X-axis of the local frame (first column of $\mathbf{R}_E(s,\tilde{s})$) is indeed the tangent vector, which depends on $\theta_E(s,\tilde{s})$ and $\Psi_E(s,\tilde{s})$ (similar to Eq. (5.3)):*

$$\mathbf{T}_E(\theta_E(s,\tilde{s}), \Psi_E(s,\tilde{s})) := \begin{bmatrix} \cos(\Psi_E(s,\tilde{s}))\cos(\theta_E(s,\tilde{s})) \\ \sin(\Psi_E(s,\tilde{s}))\cos(\theta_E(s,\tilde{s})) \\ -\sin(\theta_E(s,\tilde{s})) \end{bmatrix}. \tag{5.26}$$

Thus, combining Eqs. (5.17) and (5.26), we get Eq. (5.21).

Now, from Eq. (5.18) we can get the angles for $s = 2\tilde{s}$, thus comparing Eqs. (5.18) and (5.26), and after some trivial manipulations, we get Eqs. (5.22) and (5.23). It is worth noting that, for $\theta(\tilde{s}) \in [-\frac{\pi}{4}, 0]$ and $\Psi(\tilde{s}) \in [0, \frac{\pi}{4}]$, Eqs. (5.22) and (5.23) provide values within ranges $\theta_E(2\tilde{s}) \in [-\frac{\pi}{2}, 0]$ and $\Psi_E(2\tilde{s}) \in [0, \frac{3\pi}{4}]$, because $\sin(\theta_E(2\tilde{s})) \in [-1, 0]$ (the minimum is given for $\theta(\tilde{s}) = -\frac{\pi}{4}$ and $\Psi(\tilde{s}) = 0$). On the other hand, the maximum for $\tan(\Psi_E(2\tilde{s}))$ is produced when $\theta(\tilde{s}) = -\frac{\pi}{4}$ and $\Psi(\tilde{s}) = \frac{\pi}{4}$, being $\Psi_E(2\tilde{s}) = \frac{3\pi}{4}$ for those values, see Fig. 5.2, in its (a) and (b) plots. Also note that $\theta(\tilde{s}) \geq \theta_E(2\tilde{s})$ and $\Psi(\tilde{s}) \leq \Psi_E(2\tilde{s})$. A similar prove could be done for $s = -2\tilde{s}$, since the curve is symmetric (Lemma 5.3.2).

On the other hand, from Eq. (5.20) we can get primary Cb3D angles for a given target tangent vector $\mathbf{T}_E(2\tilde{s}) = [T_{E,x}\ T_{E,y}\ T_{E,z}]^T$ as follows:



(**a**)$\theta(\tilde{s}) \geq \theta_E(2\tilde{s})$ (**b**)$\Psi(\tilde{s}) \leq \Psi_E(2\tilde{s})$

(**c**)$\theta_E(2\tilde{s}) \leq \theta(\tilde{s})$ (**d**)$\Psi_E(2\tilde{s}) \geq \Psi(\tilde{s})$

Figure 5.2: (a)-(b) Pitch and yaw angles of an ECb3D curve (in blue) and (c)-(d) its inverse relations for a Cb3D curve (in red).

$$\Psi(\tilde{s}) = \arctan\left(\frac{T_{E,y}}{T_{E,x}+1}\right) \tag{5.27}$$

$$\begin{aligned}\theta(\tilde{s}) &= \arctan\left(\frac{-T_{E,z}}{T_{E,y}}\sin\Psi(\tilde{s})\right)\\ &= \arctan\left(\frac{-T_{E,z}}{\sqrt{(T_{E,x})^2+(T_{E,y})^2+2T_{E,x}+1}}\right)\end{aligned} \tag{5.28}$$

Now, using Eq. (5.26) *and after some basic trigonometric manipulations, we can get to Eqs.* (5.24) *and* (5.25). *Figure 5.2, in its (c) and (d) plots, shows that pitch and yaw angles of the primary Cb3D are within the indicated ranges in Eqs.* (5.24) *and* (5.25), *and also that* $\theta_E(2\tilde{s}) \leq \theta(\tilde{s})$ *and* $\Psi_E(2\tilde{s}) \geq \Psi(\tilde{s})$ ■.

Lemma 5.3.4 states that for the given range of $\theta(\tilde{s}) \in [-\frac{\pi}{4}, 0]$ and $\Psi(\tilde{s}) \in [0, \frac{\pi}{4}]$ the value of final pitch angle of the Elementary curve is always smaller (also the opposite relation occurs, given $\theta_E(2\tilde{s}) \in [-\frac{\pi}{2}, 0]$ and $\Psi_E(2\tilde{s}) \in [0, \frac{3\pi}{4}]$ the intermediate pitch angle is always greater). A similar reasoning can be applied to the yaw angle.

5.4 Analysis of ECb3D Monotony

The aim of this section is to analyse the regions of space where an ECb3D curve is monotonic in position and/or orientation (pitch and yaw angles). First, we study the necessary conditions such that the curve is monotonic in position. Afterwards, we focus on the orientation problem, which is more difficult to prove. Without loss of generality the study is made for the case where target angles $\theta_E^* \in [-\frac{\pi}{2}, 0]$ and $\Psi_E^* \in [0, \frac{\pi}{2}]$, that is the red curve in Fig. 5.1, but a similar proof can be made for the other cases.

Lemma 5.4.1 (Monotonically increasing in position) *An Elementary Clothoid-based 3D curve monotonically increases its position over s if* $\theta_E(s, \tilde{s}, \mathbf{p}) \in [-\frac{\pi}{2}, 0]$ *and* $\Psi_E(s, \tilde{s}, \mathbf{p}) \in [0, \frac{\pi}{2}]$.

Proof 5.4.1 *The position vector of* $\mathbf{E}(s, \tilde{s}, \mathbf{p})$ *is monotonically increasing iff the elements of the tangent vector, as stated in Eq.* (5.26) *of Lemma 5.3.3, are positive:*

$$\mathbf{T}_E(s, \tilde{s}, \mathbf{p}) = \left\{ \begin{array}{c} \cos(\Psi_E(s, \tilde{s}, \mathbf{p})) \cos(\theta_E(s, \tilde{s}, \mathbf{p})) \geq 0 \\ \sin(\Psi_E(s, \tilde{s}, \mathbf{p})) \cos(\theta_E(s, \tilde{s}, \mathbf{p})) \geq 0 \\ -\sin(\theta_E(s, \tilde{s}, \mathbf{p})) \geq 0 \end{array} \right. \tag{5.29}$$

We can clearly see that all elements of the tangent vector are always zero or greater than 0 for the given ranges $\theta_E(s, \tilde{s}, \mathbf{p}) \in [-\frac{\pi}{2}, 0]$ *and* $\Psi_E(s, \tilde{s}, \mathbf{p}) \in [0, \frac{\pi}{2}]$ ■.

Lemma 5.4.1 provides necessary conditions to ensure that the position increases monotonically. However, given arbitrary clothoid parameters ρ and μ, angles

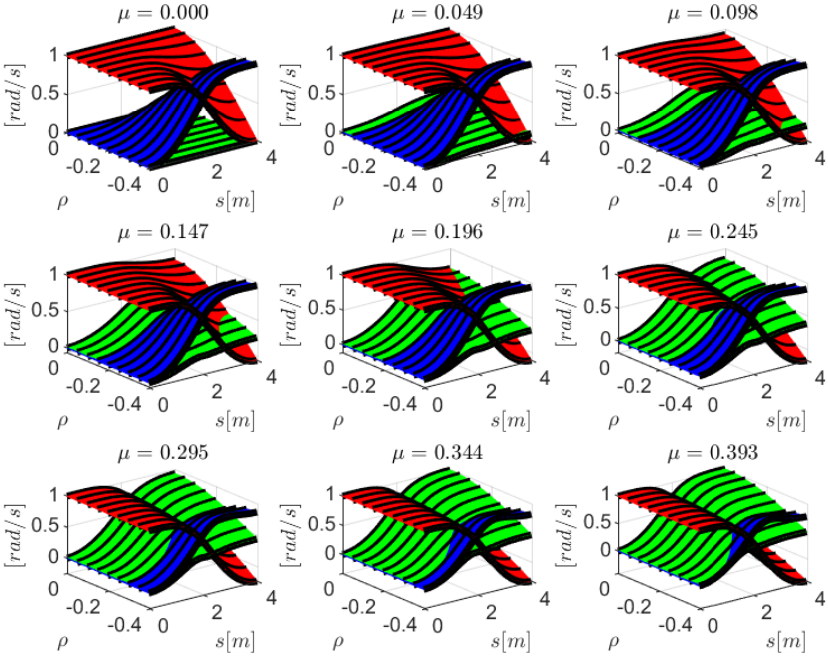
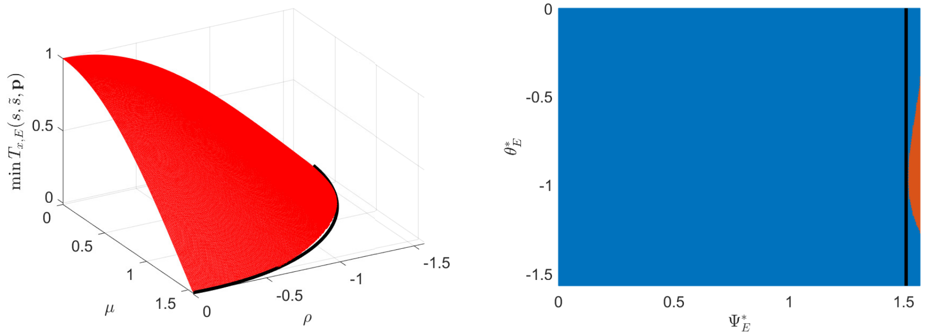


Figure 5.3: Elements of $\mathbf{T}_E(s, \tilde{s}, \mathbf{p})$ ($T_{x,E}(s)$ in red, $T_{y,E}(s)$ in green and $T_{z,E}(s)$ in blue) over a parametric sweep of $\rho \in [-\frac{\pi}{2}, 0]$ and $\mu \in [0, \frac{\pi}{2}]$ with $\tilde{s} = 1$.

in Eq. (5.21) might outside the bounds indicated in Lemma 5.3.4 for some s. This is shown in Fig. 5.3, where the elements of Eq. (5.17) have been obtained over s and a parametric sweep of $\rho \in [-\frac{\pi}{2}, 0]$ and $\mu \in [0, \frac{\pi}{2}]$ with $\tilde{s} = 1$. It can be seen that the X coordinate of the tangent vector can take negative values for some combinations of s, ρ and μ, while Y and Z coordinates are always positive. This implies that Ψ_E in Eq. (5.21) can take values greater than $\frac{\pi}{2}$ and thus the curve is not monotonic in X for those cases.

Indeed, if we plot the minimum of the X element of the tangent vector (as shown in Fig. 5.4(a)), we can see that the following constraint (depicted in black in the figure) includes all the cases where the curve is monotonic in X:

$$\sqrt{\left(\frac{\rho^* \tilde{s}^2}{\pi/2}\right)^2 + \left(\frac{\mu^* \tilde{s}^2}{\pi/2}\right)^2 - \phi \left|\frac{\rho^* \tilde{s}^2}{\pi/2}\right| \left|\frac{\mu^* \tilde{s}^2}{\pi/2}\right|} < 1, \tag{5.30}$$



(**a**)Minimum of the X coordinate of the tangent vector.

(**b**)Configuration region with monotonicity.

Figure 5.4: Analysis of the monotonicity of X.

being $\phi = 0{,}15$ a parameter that has been experimentally adjusted by trial and error until all represented cases provide positive values for all elements of the tangent vector. Constraint in Eq. (5.30) can be expressed in terms of intermediate angles using Eqs. (5.4) and (5.5) independently of the value of $\tilde{s}$:

$$\sqrt{\left(\frac{2\theta^*}{\pi/2}\right)^2 + \left(\frac{2\Psi^*}{\mathcal{C}^2(1,2\theta^*)\pi/2}\right)^2} - \phi\left|\frac{2\theta^*}{\pi/2}\right|\left|\frac{2\Psi^*}{\mathcal{C}^2(1,2\theta^*)\pi/2}\right| < 1. \qquad (5.31)$$

Figure 5.4(b) represents the region of the target configuration space where the solution obtained increases monotonically in position (blue region), which basically covers the entire configuration space. This region has been obtained by replacing Eqs. (5.24) and (5.25) in Eq. (5.31). Based on this result, in a conservative option, we could bound Eq. (5.31) with $\Psi^*_E < 1{,}51$ rad (depicted as a black line in the figure) to ensure monotonicity.

Lemma 5.4.2 (Monotonically increasing in yaw) *An Elementary clothoid-based 3D curve monotonically increases its yaw angle over s if the final pitch and yaw angles are within the ranges* $\theta_E(2\tilde{s}) \in [-\frac{\pi}{2}, 0]$ *and* $\Psi_E(2\tilde{s}) \in [0, \frac{\pi}{2}]$.

Proof 5.4.2 *Yaw angle monotonically increases iff its derivative is positive, thus, recalling that* $\Psi'(s) \geq 0$ *for* $|s| \leq \tilde{s}$ *as demonstrated in [Girbés, Vanegas y Armesto 2019], we need to check that the secondary Cb3D yaw angle also* *increases monotonically (Eq.* (5.21) *from Lemma 5.3.4). Its derivative, for* $|s| > \tilde{s}$*, is computed as:*

$$\tilde{\Psi}'_E(s,\tilde{s}) = \frac{\tilde{T}'_y(s,\tilde{s})\tilde{T}_x(s,\tilde{s}) - \tilde{T}'_x(s,\tilde{s})\tilde{T}_y(s,\tilde{s})}{\tilde{T}^2_x(s,\tilde{s}) + \tilde{T}^2_y(s,\tilde{s})}, \tag{5.32}$$

So if $\chi(s,\tilde{s}) := \tilde{T}'_y(s,\tilde{s})\tilde{T}_x(s,\tilde{s}) - \tilde{T}'_x(s,\tilde{s})\tilde{T}_y(s,\tilde{s}) \geq 0$*, implies that* $\tilde{\Psi}'_E(s,\tilde{s}) \geq 0$*, which is true for the given ranges:*

$$\begin{aligned}\chi(s,\tilde{s}) = &(\sin(2\theta(\tilde{s}))\sin(\Psi(s-2\tilde{s}') - \Psi(\tilde{s}))\theta'(s-2\tilde{s})\\ &-\cos(2\theta(\tilde{s}))\cos^2(\theta(s-2\tilde{s}))\Psi'(s-2\tilde{s})\\ &-\frac{1}{2}\sin(2\theta(\tilde{s}))\sin(2\theta(s-2\tilde{s}))\cos(\Psi(s-2\tilde{s}')\\ &-\Psi(\tilde{s}))\Psi'(s-2\tilde{s})) \geq 0,\end{aligned} \tag{5.33}$$

since $\theta(\tilde{s}) \geq \theta_E(2\tilde{s})$*,* $\Psi(\tilde{s}) \leq \Psi(2\tilde{s})$ *and the fact that* $\sin(2\theta(\tilde{s})) \leq 0$, $\cos(2\theta(\tilde{s})) \geq 0$, $\sin(\Psi(s-2\tilde{s})-\Psi(\tilde{s})) \leq 0$*,* $\theta'(s-2\tilde{s}) \geq 0$*,* $\Psi'(s-2\tilde{s}) \leq 0$*,* $\sin(2\theta(s-2\tilde{s})) \leq 0$*,* $\cos(\Psi(s-2\tilde{s}) - \Psi(\tilde{s})) \geq 0$*.* ■

Lemma 5.4.2 provides necessary conditions to obtain solutions where the rotation given by the yaw angles increases monotonically. However, we cannot guarantee that pitch angle decreases monotonically for the specified ranges. Indeed, there are cases, depending on ρ and μ, where the derivative of the angle for the secondary Cb3D takes positive values. This can be seen from the derivative of Eq. (5.22):

$$\begin{aligned}\tilde{\theta}'_E(s,\tilde{s}) &= -\frac{\tilde{T}'_z(s,\tilde{s})}{\sqrt{1-\tilde{T}^2_z}},\\ \tilde{T}'_z(s,\tilde{s}) &= \left(\begin{array}{l}\left(\begin{array}{l}\sin(\theta(s-2\tilde{s}))\sin(2\theta(\tilde{s}))\cos(\Psi(s-2\tilde{s})-\Psi(\tilde{s}))-\\ -\cos(\theta(s-2\tilde{s}))(2\sin^2(\theta(\tilde{s}))-1)\end{array}\right)\theta'(s-2\tilde{s})+\\ +\left(\sin(\Psi(s-2\tilde{s})-\Psi(\tilde{s}))\sin(2\theta(\tilde{s}))\cos(\theta(s-2\tilde{s}))\right)\Psi'(s-2\tilde{s})\end{array}\right) \geq 0,\end{aligned} \tag{5.34}$$

where the terms multiplying $\theta'(s-2\tilde{s})$ and $\Psi'(s-2\tilde{s})$ are both positive, however $\theta'(s-2\tilde{s}) \geq 0$ and $\Psi'(s-2\tilde{s}) \leq 0$. As a consequence, there is no guarantee that $\tilde{T}'_z(s,\tilde{s})$ is positive for all s. Indeed, by definition, $\tilde{T}'_z(\tilde{s},\tilde{s}) \geq 0$ because the primary Cb3D increases monotonically for those ranges, and $\tilde{T}'_z(2\tilde{s},\tilde{s}) = 0$

since the ECb3D path's curvature and torsion are both zero when $s = 2\tilde{s}$. Therefore, depending on ρ and μ the derivative could take positive values.

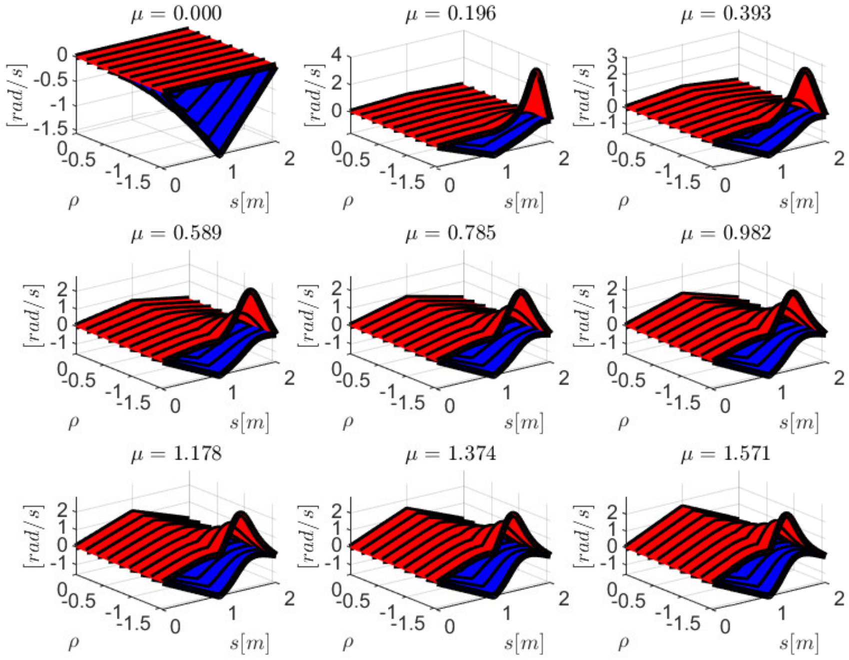
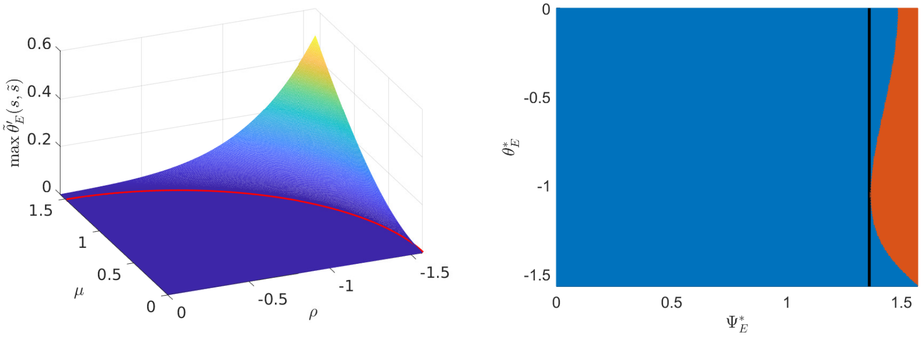


Figure 5.5: Derivatives $\tilde{\Psi}'_E(s, \tilde{s})$ (in red) and $\tilde{\theta}'_E(s, \tilde{s})$ (in blue) over a parametric sweep of $\rho \in [-\frac{\pi}{2}, 0]$ and $\mu \in [0, \frac{\pi}{2}]$ with $\tilde{s} = 1$.

This aspect is shown in Fig. 5.5, where a study has been conducted to represent the derivatives over s of yaw and pitch angles, $\tilde{\Psi}'_E(s, \tilde{s})$ and $\tilde{\theta}'_E(s, \tilde{s})$ respectively, and a parametric sweep of $\rho \in [-\frac{\pi}{2}, 0]$ and $\mu \in [0, \frac{\pi}{2}]$ with $\tilde{s} = 1$. It can be seen that there are combinations, particularly with large $|\rho|$ and $|\mu|$, that generate a slight increase of $\tilde{\theta}_E(s, \tilde{s})$ (positive values of its derivative depicted in blue in Fig. 5.5), while the derivative of $\Psi_E(s, \tilde{s})$ is always positive (as already proved in Lemma 5.4.2). Varying the clothoid length would provide similar results. Indeed, if we use the scaling property of Lemma 5.3.2, clothoid's parameter ranges need to be reduced by a factor of λ^{-2}, being λ the scaling factor, which produces a derivative scaled with λ^{-1} factor.

Inspired by this result, we seek now a region where pitch angle decreases monotonically. Figure 5.6(a) shows the maximum of $\tilde{\theta}'_E(s, \tilde{s})$ over a parametric



(a) Maximum of the derivative $\tilde{\theta}'_E(s,\tilde{s})$

(b) Configuration region where pitch angle decreases monotonically

Figure 5.6: Analysis of the monotonicity of pitch angle.

sweep of $\rho \in [-\frac{\pi}{2\tilde{s}^2}, 0]$ and $\mu \in [0, \frac{\pi}{2\tilde{s}^2}]$ with $\tilde{s} = 1$. It can be seen that there is a wide region where the maximum is zero (which implies that pitch angle decreases monotonically). Due to the shape of this region, we have experimentally adjusted an ellipsoid that constrains solutions to ensure a monotonicity of pitch angle:

$$\sqrt{\left(\frac{\rho^*}{r_\rho}\right)^2 + \left(\frac{\mu^*}{r_\mu}\right)^2} < 1. \tag{5.35}$$

with $r_\rho = \frac{\pi}{2\tilde{s}^2}$ and $r_\mu = \frac{\varphi\pi}{2\tilde{s}^2}$, being $\varphi = 0{,}943277178$. The value for φ has been experimentally obtained by analysing all the cases depicted in Fig. 5.6(a) where the maximum derivative of the angle is zero. Replacing ρ^* and μ^* with Eqs. (5.4) and (5.5), respectively, then Eq. (5.35) can be equivalently expressed as:

$$\sqrt{\left(\frac{2\theta^*}{\pi/2}\right)^2 + \left(\frac{2\Psi^*}{\varphi\frac{\pi}{2}\mathcal{C}^2(1, 2\theta^*)}\right)^2} < 1. \tag{5.36}$$

It is important to remark that Eq. (5.36) is independent of the clothoid arc length and sharpness parameters. The configuration region where this condition is satisfied is depicted in blue in Fig. 5.6(b). A conservative constraint would be to impose that $\Psi^*_E < 1{,}36$ rad (depicted as a black line in Fig. 5.6(b)).

5.5 Design Methodology Proposal

In this section we describe a design methodology to compute clothoid parameters ρ^* and μ^* based on target angles $\theta_E^* \in [-\frac{\pi}{2},\ 0]$ rad and $\Psi_E^* \in [0,\ 1{,}36]$ rad, where admissible regions to ensure monotonicity have been taken into account as a result of the analysis performed in Section 5.4. An important aspect to consider is that we seek for solutions with the minimum possible curve length S, given admissible clothoid parameters $\rho^* \in [-\rho_{max}, \rho_{max}]$ and $\mu^* \in [-\mu_{max}, \mu_{max}]$. Thus, the length of the primary Cb3D, $\tilde{s}$, needs to be computed together with the clothoid parameters.

The methodology is based on the following steps:

1. Compute primary clothoid angles θ^* and Ψ^* from target angles θ_E^* and Ψ_E^* using Eqs. (5.24) and (5.25).

2. Then, assume that maximum torsion sharpness is applied, that is, $\rho^* = \text{sign}(\theta^*)\rho_{max}$. Thus, from Eq. (5.4), we compute the primary Cb3D length as:

$$\tilde{s}_{min} = \sqrt{\frac{2|\theta^*|}{\rho_{max}}}, \tag{5.37}$$

   where $\tilde{s}_{min}$ is the shortest possible length (considering only the torsion sharpness constraint).

3. Compute μ^* using Eq. (5.5) assuming such length and previously computed yaw angle Ψ^*. If $|\mu^*| \leq \mu_{max}$, it means that the assumption about the maximum torsion solution was correct and then $\tilde{s}^* = \tilde{s}_{min}$. However, if $|\mu^*| > \mu_{max}$, it means that the primary Cb3D is too short, since it violates the admissible curvature sharpness, i.e.: torsion sharpness needs to be reduced. In that case, we now assume the maximum curvature sharpness is applied $\mu^* = \text{sign}(\Psi^*)\mu_{max}$. Using Eq. (5.5) we can compute the primary Cb3D length $\tilde{s}^*$, by replacing ρ^* of Eq. (5.4) into Eq. (5.5), that is $\mathcal{C}(\tilde{s}, \rho^*) = \mathcal{C}(\tilde{s}, \frac{2\theta^*}{\tilde{s}^2})$. Then, recalling the scalability property of Fresnel cosine, this term can be computed as $\mathcal{C}(\tilde{s}, \frac{2\theta^*}{\tilde{s}^2}) = \mathcal{C}(1, 2\theta^*)\tilde{s}$. As a consequence, a proper clothoid length can be chosen to ensure that both sharpness constraints are satisfied:

$$\tilde{s}^* = \frac{\sqrt{\frac{2|\Psi^*|}{\mu_{max}}}}{\mathcal{C}(1, 2\theta^*)}. \tag{5.38}$$

4. Compute torsion sharpness ρ^* using Eq. (5.4) and the previously computed length.

5.5.1 Case study

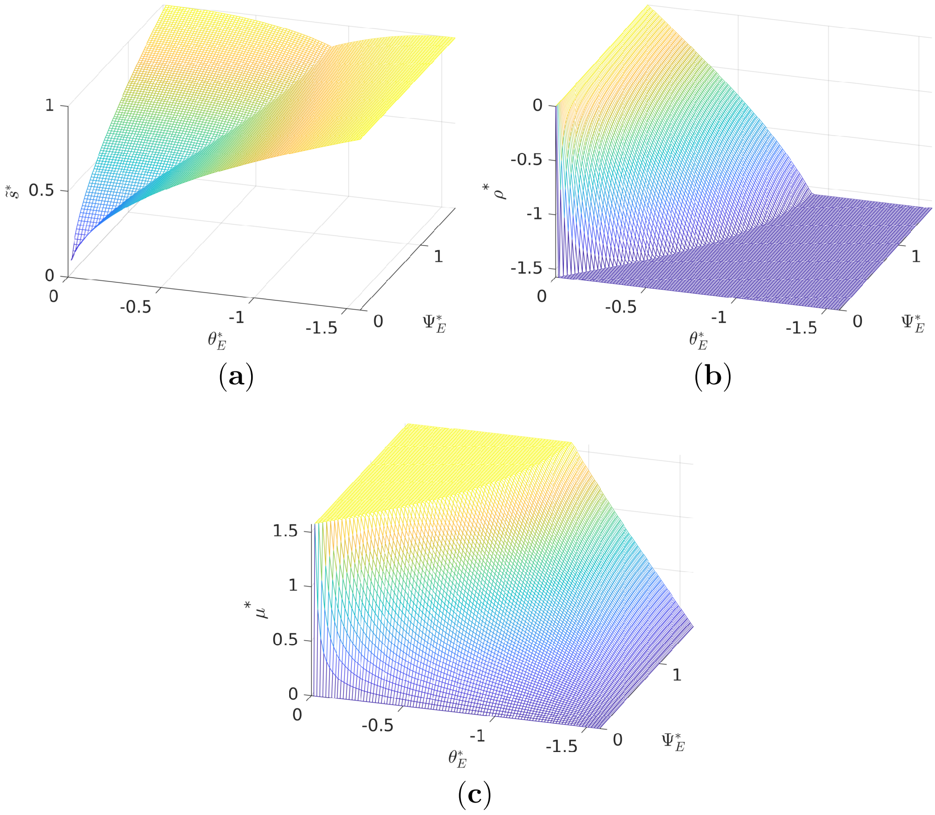
The proposed method finds clothoid parameters with the shortest possible length because it finds solutions trying to maximize clothoid sharpness (it selects the solution that first violates the maximum clothoid sharpness constraints). Figure 5.7 shows the computed values for $\tilde{s}^*$, ρ^* and μ^* using the proposed method, assuming that $\rho_{max} = \frac{\pi}{2}$ and $\mu_{max} = \frac{\pi}{2}$. These results were compared with a nonlinear optimization procedure (using `fmincon` function in Matlab®) where the curve with the shortest length, satisfying clothoid sharpness constraints, was found. As a result, the maximum error between both alternative procedures in the whole surface was 10^{-6}, but the optimization procedure takes about 100 times more to produce the same results.

Two particular cases of ECb3D curves are analyzed next. In the first one the solution obtained reaches the maximum torsion sharpness, whereas the second one provides the maximum curvature sharpness. In particular, if $\theta^* = -\frac{\pi}{4}$ and $\Psi^* = \frac{\pi}{4}$, we obtain $\rho^* = -\frac{\pi}{2}$, $\mu^* = 1{,}24511$ and $\tilde{s}^* = 0{,}731738$; while if $\theta^* = -\frac{\pi}{8}$ and $\Psi^* = 3\frac{\pi}{8}$, we obtain $\rho^* = -0{,}64818$, $\mu^* = \frac{\pi}{2}$ and $\tilde{s}^* = 0{,}85105$. Figure 5.8 represents these two curves, where it can clearly be seen that the parameters obtained for the curve depicted in solid blue imply a solution where torsion sharpness is maximum, while the curve depicted in dashed red provides a solution with maximum curvature sharpness. In both cases, the curves reach the specified angles.

It's worth mentioning that the average computational time to compute 200 points of the curve parameters is approximately 170μs (including clothoid parameters computation) based on our code implemented in Matlab® 2019b running on computer with an Intel® Core i7-8750H processor and 16 GB of RAM memory and Windows 10.

5.5.2 Benchmarking

To complete the analysis of the proposed methodology, we have generated 10^4 random target orientations, Ψ^* and θ^*, comprised in the ranges $\Psi^* \in [0, \frac{\pi}{2}]$ rad and $\theta^* \in [0, \frac{\pi}{2}]$ rad, respectively. The aim is to compare some geometric properties, such as abruptness, maximum bending energy, as well as accuracy and computational time between our proposed curve (ECb3D) and an elementary curve consisting of two symmetric pure 3D clothoids (EC3D) [Harary y Tal



(a)

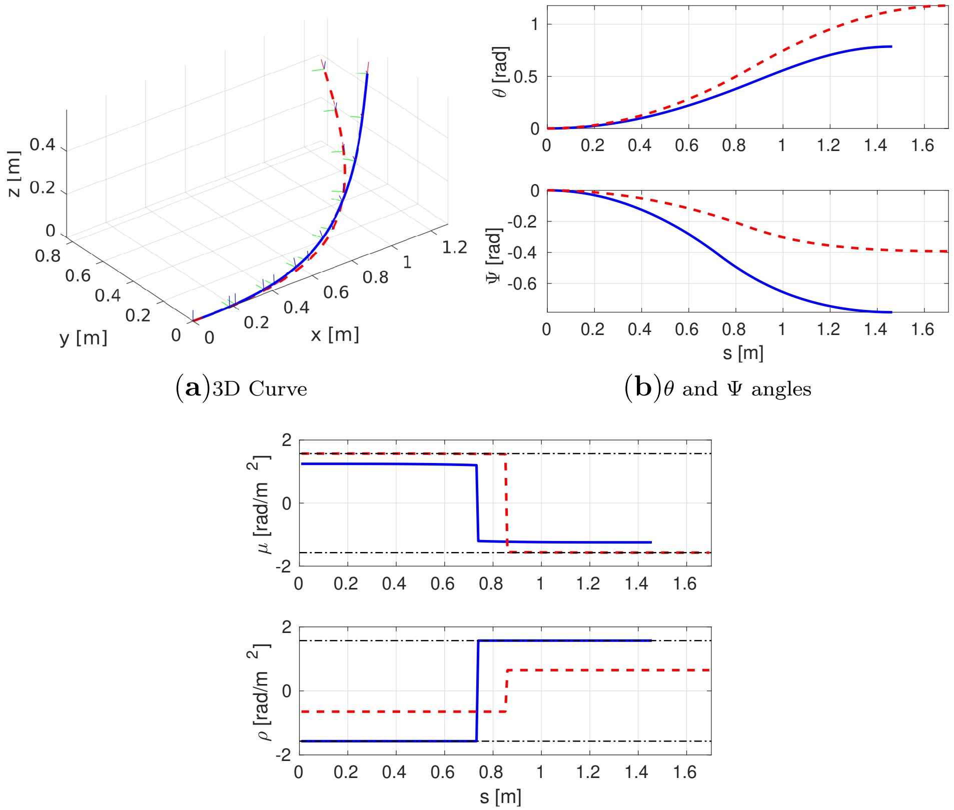
(b)

(c)

Figure 5.7: Computed clothoid parameters using the proposed methodology.

2012a]. EC3D are obtained after an optimization procedure where the curve sharpness parameters for curvature and torsion are adjusted so that the curve reaches the target orientations. In both cases, the length of the curves have been set to $S = 2$ m.

On the one hand, Fig. 5.9(a) depicts the histogram of the maximum bending energy of each curve $\text{máx}(\mathcal{B}(s))$, being $\mathcal{B}(s) := \kappa'^2(s) + \tau'^2(s)$ and $\kappa'(s)$ and $\tau'(s)$ the geometric derivatives of curve's curvature and torsion, respectively. In both cases, it can be seen that, in most cases, the proposed curve provides smaller bending energy (histogram cases are concentrated on the lower part of the chart). On the other hand, Fig. 5.9(b) shows the results obtained for the



(**a**)3D Curve

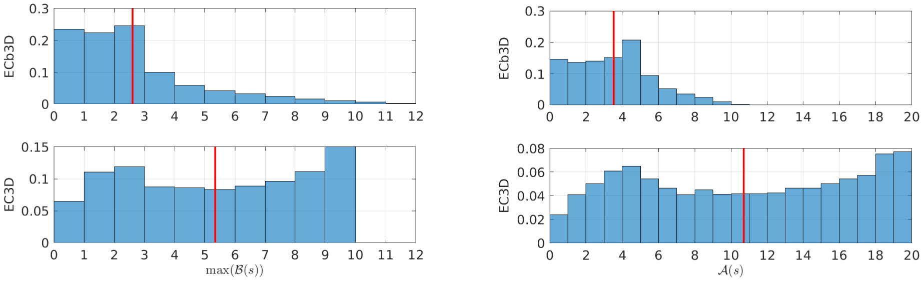
(**b**)θ and Ψ angles

(**c**)Curvature and torsion sharpness

Figure 5.8: Two representative Elementary Clothoid-based 3D curves, with final angles $\theta^* = -\frac{\pi}{4}$ and $\Psi^* = \frac{\pi}{4}$ (solid blue), and with final angles $\theta^* = -\frac{\pi}{8}$ and $\Psi^* = 3\frac{\pi}{8}$ (dashed red). Sharpness bounds appear in dash-dotted black.

curve abruptness $\mathcal{A}(S) = \int_0^S \mathcal{B}(s)ds$, where it can be seen that ECb3D curve provides a solution with smaller abruptness.

The average computational time for computing curve parameters for a ECb3D is 28 μs, while computing an EC3D curve takes 121 ms in average. The superior performance is clearly influenced by the fact that the EC3D requires an optimization procedure in order to synthesize the curve. In addition to this, the proposed method is exact (errors are in the order of numerical accuracy), while the EC3D provides an approximated solution with an average error of 1 mrad, measured as $\epsilon := \sqrt{(\Psi^* - \Psi(S))^2 + (\theta^* - \theta(S))^2}$.



(**a**)Maximum bending energy

(**b**)Abruptness

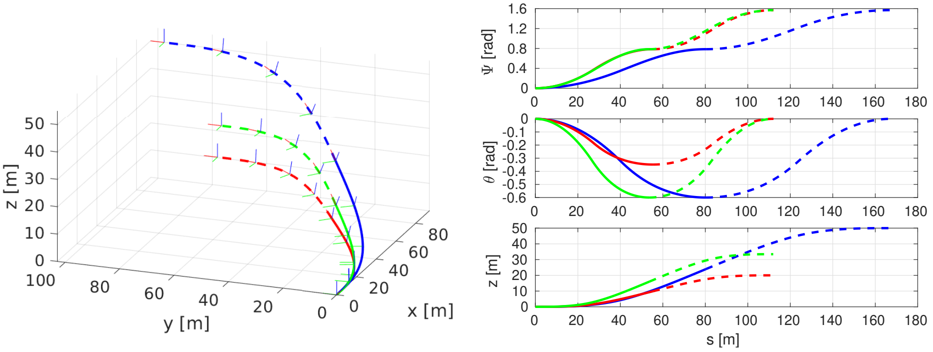
Figure 5.9: Maximum bending energy and abruptness histograms to compare ECb3D and EC3D curves. Vertical axis represents parts per unit, where red lines show mean values for each metric.

5.5.3 Fixed-wing UAV manoeuvres generation

The aim is to find a path for an UAV moving on a horizontal plane that manoeuvres to reach a target heading orientation $\Psi^* \in [-\pi, \pi]$ (relative to its current heading direction) on another horizontal plane ($\theta^* = 0$) at a different altitude z^*, without exceeding the maximum allowed pitch angle $|\theta| < \theta_{max}$, and also considering maximum allowed sharpness in curvature and torsion, that is, $\rho \in [-\rho_{max}, \rho_{max}]$ and $\mu \in [-\mu_{max}, \mu_{max}]$. In order to solve this problem, we can use two concatenated ECb3D curves with intermediate heading orientation $\Psi^*_m = \frac{\Psi^*}{2}$ and $\theta^*_m = -\operatorname{sign}(z^*)\theta_{max}$. These angles will be the actual target angles to reach with the first ECb3D curve (as explained at the beginning of this section). Then, we compute relative pitch and yaw angles required for the second ECb3D by computing the relative rotation matrix $\mathbf{R}^*_m = (\mathbf{R}_z(\Psi^*_m)\mathbf{R}_y(\theta^*_m))^{-1}(\mathbf{R}_z(\Psi^*)\mathbf{R}_y(\theta^*))$, details left to the reader.

Due to the assumption that the first ECb3D produces a curve with a maximum or minimum pitch angle, i.e.: $\theta^*_m = \theta_{max}$ or $\theta^*_m = -\theta_{max}$, this implies that the curve reaches a minimum altitude change z_{min}. Thus, if $|z^*| > z_{min}$, then the curve can be scaled by a factor $\lambda = (\frac{z_{min}}{z^*})^2$. On the contrary, if $|z^*| < z_{min}$, then we need to find the pitch angle that produces the required height (using any optimization method). The resulting solution produces a curve that reaches z^* with the maximum sharpness parameters.

Figure 5.10(a) shows three cases of two concatenated ECb3D curves generating manoeuvres with $\Psi^* = \frac{\pi}{2}$ heading at different altitudes with $\rho_{max} = \mu_{max} = 0{,}001$ rad/m^2 and $\theta_{max} = 0{,}6$ rad. In particular, we show in green the case where, due to the given constraints, the target altitude is equal to z_{min}. The curve plotted in red represents a case where $z^* < z_{min}$, while the curve plotted in blue represents a case where $z^* > z_{min}$. It can be seen that minimum pitch angle for green and blue curves is $min(\theta) = -0{,}6$ rad, expected by design (the blue one reaches a higher altitude by scaling the ECb3D parameters). The minimum pitch angle for the red curve is bigger than $-\theta_{max}$ as a consequence of an optimization procedure that makes the curve reach the target altitude.



(a) Position

(b) Orientation and altitude

Figure 5.10: Three paths with two concatenated ECb3D curves each, with $\rho_{max} = \mu_{max} = 0{,}001$ rad/m^2 and $\theta_{max} = 0{,}6$, reaching final orientation $\Psi^* = \frac{\pi}{2}$ rad and different altitudes $z^* = 20$ m (red), $z^* = 33{,}44 = z_{min}$ m (green) and $z^* = 50$ m (blue).

5.6 Discussion

Smooth spatial paths for autonomous aerial vehicles are usually generated using Bezier curves, splines or Pythagorean Hodographs, among other curves [Wang y col. 2017b; Neto, Macharet y Campos 2013; Vinokursky, Mezentceva y Samoylov 2020; Pfeifle y Fichter 2021]. However, they do not have a clear physical meaning or require to solve an optimization problem in order to take geometric, kinematic or dynamic constraints into account [Neto, Macharet y Campos 2013; Vinokursky, Mezentceva y Samoylov 2020]. The curve proposed in this Note inherits the properties from the Clothoid-based 3D curves introduced in Girbés, Vanegas y Armesto 2019 and, therefore, it has a physical meaning as its design parameters are related to curvature and torsion sharpness. ECb3D paths can be computed using optimization-free approximation methods [Sánchez-Reyes y Chacón 2003; Montés y col. 2008; Meek y Walton

2004b] based on Bezier, splines or polynomial curves, among others. ECb3D paths are smooth and scalable, which make them suitable for many UAV navigation applications.

Some potential applications of the proposed Elementary Clothoid-based 3D curve are the following. Firstly, ECb3D curves can be used as transition curves in global path planning problems defining a poly-line in the workspace. Using ECb3D as a transition curve between segments will avoid the intrinsic $\mathcal{C}^1$ discontinuity. Secondly, ECb3D curves can be integrated in local path planning methods to avoid obstacles by generating random motion primitives pointing to different directions. Collision tests could be performed and the collision-free ones would be selected as candidate solutions, where the final decision would depend on specific optimization criteria. Finally, ECb3D curves can also be used as a primitive in UAVs planning problems, combining straight segments, arcs and transition curves as part of their set of primitives. For instance, specific planning problems, such as changing the direction and altitude of a fixed-wing UAV, can benefit from the use of ECb3D curves by combining two of them as shown in section 5.5.3.

5.7 Conclusions

In this note, we have proposed a new curve by concatenating two symmetric Clothoid-based 3D curves (Cb3D). We have proved that the new curve keeps the same properties regarding smoothness, scalability and symmetry as the original Cb3D curves. One of the main advantages of the new curve, compared to Cb3D, is that curvature and torsion are zero, both at the beginning and at the end of the path, which makes it easier to combine with straight line segments and use them as transition curves to change the direction, while satisfying curvature and torsion sharpness constraints, i.e.: to ensure that transitions are smooth and compliant with some higher-level kinematic and dynamic constraints.

This note also shows how to obtain curve parameters given the curve length, as well as how to compute pitch and yaw angles at the end of the curve and at the intermediate position. We also analyze the range of target pitch and yaw angles that ensures that the proposed curve increases monotonically in position and orientation, which is a desirable property for transition curves.

A method to obtain the shortest possible curve satisfying parameter constraints has also been proposed in section 5.5. We also show a case study where fundamental variables of the curve are depicted and a comparison between the proposed curve and the curve in [Harary y Tal 2012a] has been conducted. The benchmarking concludes that the proposed curve generates smoother paths in the majority of the cases, with reduced bending energy and abruptness. The computational time and error are clearly smaller due to the closed-form solution presented.

At the end of the Note, we discuss why ECb3D paths are suitable for many UAV navigation applications. The main conclusions are that smoothness and scalability properties, as well as accuracy and reduced computation time, allow the use of ECb3D paths as primitive curves for higher level 3D path planners.

Further work will be oriented to generalize the problem statement to design curves that join two arbitrary positions and orientations in the configuration space, for instance, by combining ECb3D curves with line segments and helix curves extending the ideas of [Scheuer y Fraichard 1997] to 3D.

Funding Sources

This research was funded by Generalitat Valenciana, grant $GV/2021/074$, and by Agencia Española de Investigación (European Union), grants $PID2020-116585GB-I00$ and $PID2020-118071GB-I00$.

Capítulo 6

Smooth three-dimensional route planning for fixed-wing unmanned aerial vehicles with double continuous curvature

In this chapter a new path planner is described, which is able to generate a smooth trajectory that starts from an arbitrary 3D position and orientation at the same time. In addition the path planner is also able to concatenate several curves of the same type, generating a trajectory without loss of continuity.

Chapter Note: The contents of this chapter can be found in the paper:

> **Vanegas, G**., Armesto, L., Girbés-Juan, V., & Pérez, J. (2022). Smooth Three-Dimensional Route Planning for Fixed-Wing Unmanned Aerial Vehicles With Double Continuous Curvature. IEEE Access, 10, 94262-94272. JCR: Q1, Impact Factor: 3.476, SJR: Q1, Impact Factor: 0.93, DOI: https://10.1109/ACCESS.2022.3203069.

Abstract: *This paper presents a smooth flight path planner for maneuvering in a 3D Euclidean space, which is based on two new space curves. The first one is called "Elementary Clothoid-based 3D Curve (ECb3D)", which is built by concatenating two symmetric Clothoid-based 3D Curves (Cb3D). The combination of these curves allows to reach an arbitrary orientation in 3D Euclidean space. This new curve allows to generate continuous curvature and torsion profiles that start and finish with a null value, which means that they can be concatenated with other curves, such as straight segments, without generating discontinuities on those variables. The second curve is called "Double Continuous Curvature 3D Curve (DCC3D)" which is built as a concatenation of three straight line segments and two ECb3D curves, allowing to reach an arbitrary configuration in position and orientation in the 3D Euclidean space without discontinuities in curvature and torsion. This trajectory is applied for autonomous path planning and navigation of unmanned aerial vehicles (UAVs) such as fixed-wing aircrafts. Finally, the results are validated on the FlightGear 2018 flight simulator with the UAV kadett 2400 platform.*

Keywords: Nonholonomic Motion Planning, Motion and Path Planning, Constrained Motion Planning, Autonomous Vehicle Navigation.

6.1 Introduction

In the last decades, the aeronautics industry has maintained a continuous and vertiginous development, particularly in the military field [Chaurasia y Mohindru 2021]. The literature in the different fields of research is extensive. A relevant field of study focuses on path planning, which has been approached from different perspectives, such as communication networks [Zhang y col. 2019b; Zhang y col. 2019a] or computational intelligence based on path planning algorithms [Zhao, Zheng y Liu 2018; Cabreira, Brisolara y Ferreira Paulo 2019]. Additionally, applications for civilian missions are currently in high demand [Wargo y col. 2014], such as rescue missions [Hayat y col. 2017; Ryan y col. 2004], or work in agriculture [Xiang y Tian 2011].

The aim of flight planning for aerial vehicles is to generate a path joining initial and final configurations, while passing through several intermediate target points. In case of potential collisions with static or dynamic objects, such as other UAVs flying nearby, the initial route must be replanned to guarantee collision-free paths [Huang, Teo y Tan 2019; Huang y Teo 2019].

In particular, this paper addresses the 3D path planning task for UAVs with nonholonomic characteristics [Velasco-Carrau y col. 2016; Shen y col. 2018], i.e., fixed-wing aircrafts. In this sense, it should be highlighted that one of the most relevant particularities of this kind of paths is the continuity of the curve, since a fixed-wing UAV cannot perform abrupt maneuvers during its flight time. Therefore, the starting point to create a smooth flight path must consider the particular maneuverability capabilities of the UAV. Several studies have been proposed in this context, where 3D path planning algorithms use cubic Bezier spiral curves to satisfy the curvature constraint are presented in [Yang y Sukkarieh 2008b; Yang y Sukkarieh 2008c; González y col. 2014], while [Neto, Macharet y Campos 2010] proposes a seventh-order Bézier curve as a continuous curvature path approximation, which does not exceed the kinematic constraints of an aerial vehicle. The authors of [Liu y col. 2021] have performed a fusion between two heuristic methodologies, with the aim of solving the smooth path planning problem in a mountainous environment, for which the characteristics of B-spline curves are exploited.

The authors of [Wang, Li y Li 2014] study curvature constraints in path planning and solve this problem through Dubins curves. In Tan y col. 2022, the smooth trajectory planning problem with continuous curvature is solved through an optimization algorithm based on Pythagorean curves, which satisfies the kinematic constraints of the UAV, in a similar approach, the authors of Davoodi y col. 2015 propose a path planning generated through a multiobjective optimization problem operating with standard genetic operators. In Huang y col. 2016 coordinated path planning for multiple UAVs is performed, starting from an ant colony optimization algorithm smoothed through a k-degree smoothing method. While in Wu y col. 2018 a Rauch-Tung-Striebel (RTS), smoothing is used, a procedure that permits smoothing the path produced by a Particle Swarm Optimization (PSO) algorithm, while in Armesto y col. 2015 an RTS smoothing is also used for the control of mobile robots with nonholonomic wheels. A particular approach is presented in Liu y Hu 2021, where a smooth path planning algorithm based on a Gaussian spectrum function is built, which aims to optimize the smooth path. Finally, in Kaiser y Verl 2021 model-based smooth paths are proposed for the estimation of the optimal geometric parameters, through polynomial spline curves, the results are used in industrial robots, with the aim of improving the productivity.

Focusing on works with good results based on clothoid curves, Gim y col. 2017b solves the problem of generating continuous curvature paths by composing multiple clothoids. The relevance of clothoid curves and their application to nonholonomic vehicles can also be appreciated in Lima y col. 2015; Silva y Grassi 2018; Kim y Chung 2021. In Chen y col. 2017b; Vanegas y col. 2018, clothoids are approximated using Bézier curves to minimize curvature profiles and thus guarantee higher-order geometric continuity while minimizing error. In Lambert, Romano y Watling 2021, smooth paths based on clothoid curves are proposed for planning high-speed wheeled vehicle paths, for which a numerical optimization is performed within the constraints of convex regions. In Sui y col. 2020, autonomous valet parking service path planning is performed. Finally, the authors of [Huang y col. 2021] propose a path smoothing, based on clothoid curves, parameterized by the arc length.

The aim of this work is to generate a smooth flight path in the context of fixed-wing UAV autonomous navigation. A new 3D smooth curve, called Elementary Clothoid-based 3D Curve or ECb3D, is proposed, which is built by combining two symmetric Cb3D [Girbés, Vanegas y Armesto 2019]. An ECb3D is capable of reaching an arbitrary direction in 3D space, being its curvature and torsion profiles equal to zero at both the beginning and the end of the curve. That property allows to build more complex curves combining them with straight line segments or other ECb3D curves. In this sense, a second 3D smooth curve is introduced, coined as Double Continuous Curvature 3D Curve (DCC3D), which is a concatenation of two ECb3D curves and three straight line segments. A DCC3D curve can reach any arbitrary position and orientation in 3D Euclidean space. Finally, it should be emphasized that collision avoidance is out of the scope of this paper, although the proposed path could be used as a primitive in both global and local planners to generate collision-free paths.

This paper is organized as follows: in section 6.2, the formulations of the preliminary works related to this article are discussed. Section 6.3 describes the problem to be solved. In Section 6.4, the methodology to design smooth curves to reach arbitrary target orientation is explained in depth, whereas Section 6.5 describes how to generate 3D Double Continuous Curvature Curves that allow to reach arbitrary position and orientation. Section 6.6 presents the results that validate the application of this new curve through flight simulations performed on a fixed-wing UAV. Finally, conclusions and further work are described in Section 6.7.

6.2 Preliminaries

A nonholonomic constraint cannot be expressed only in position variables but includes the time derivative of one or several variables. In direct reference to fixed-wing UAVs, these constraints are directly related to their maneuverability in flight.

There are different approaches used for the construction of smooth paths, whether they are heuristic [Mac y col. 2016] or geometrical [Hota y Ghose 2010a; Hota y Ghose 2010c; Upadhyay y Ratnoo 2015]. Thus, this work takes a geometric approach for constructing smooth paths as a starting point, based on the criterion of 3D continuous curves. Thereafter, a set of concepts necessary for the development of this article are defined.

6.2.1 Curves in space

A curve in space $\mathbb{R}^n$, can be defined as a vector function [Do Carmo 2016] such that:

$$\mathbf{C} : [a, b] \rightarrow \mathbb{R}^n, \mathbf{C}(s) = (x_1(s), \ldots, x_n(s)) \tag{6.1}$$

where, the points $\mathbf{C}(a)$ and $\mathbf{C}(b)$ are the initial and final boundaries of the curve. In particular, a curve in the three-dimensional space $\mathbb{R}^3$ can be defined as $\mathbf{C}(s) = (x(s)_{\mathbf{i}} + y(s)_{\mathbf{j}} + z(s)_{\mathbf{k}})$, where **i**, **j**, **k** refer to the unit vectors of the global reference frame.

Tangent, normal and *binormal* vectors are defined as:

- $\mathbf{T}(s)$ is the unit vector tangent to the curve, pointing tp the direction of movement:

$$\mathbf{T}(s) = \frac{\mathbf{C}'(s)}{\|\mathbf{C}'(s)\|}. \tag{6.2}$$

- $\mathbf{N}(s)$ is the unit normal vector, given by the ratio of the derivative of $\mathbf{T}(s)$ to its length:

$$\mathbf{N}(s) = \frac{[\mathbf{C}'(s) \times \mathbf{C}''(s)] \times \mathbf{C}'(s)}{\|[\mathbf{C}'(s) \times \mathbf{C}''(s)] \times \mathbf{C}'(s)\|}. \tag{6.3}$$

- $\mathbf{B}(s)$ is the binormal unit vector, represented by the cross product of $\mathbf{T}(s)$ and $\mathbf{N}(s)$:

$$\mathbf{B}(s) = \mathbf{T}(s) \times \mathbf{N}(s) = \frac{\mathbf{C}'(s) \times \mathbf{C}''(s)}{\|\mathbf{C}'(s) \times \mathbf{C}''(s)\|}, \tag{6.4}$$

where $\mathbf{C}'(s) = \mathrm{d}\mathbf{C}(s)/\mathrm{d}s$, $\mathbf{C}''(s) = \mathrm{d}^2\mathbf{C}(s)/\mathrm{d}s^2$, and $\mathbf{C}'''(s) = \mathrm{d}^3\mathbf{C}(s)/\mathrm{d}s^3$ are the derivatives of the position vector $\mathbf{C}(s)$.

On the other hand, $\kappa(s)$ defines the curvature of the curve, such that:

$$\kappa(s) = \frac{\|\mathbf{C}'(s) \times \mathbf{C}''(s)\|}{\|\mathbf{C}'(s)\|^3}, \tag{6.5}$$

and $\tau(s)$ is the torsion, defined as:

$$\tau(s) = \frac{\mathbf{C}'(s) \cdot (\mathbf{C}''(s) \times \mathbf{C}'''(s))}{\|\mathbf{C}''(s)\|^2} \tag{6.6}$$

Therefore, for a continuous curve defined in $\mathbb{R}^3$, as the one shown in Figure 6.1, $\mathbf{T}(s)$ is a unit vector defining the direction of the curve, $\mathbf{N}(s)$ is perpendicular to $\mathbf{T}(s)$, while $\mathbf{B}(s)$ forms a right-handed system between $\mathbf{T}(s)$ and $\mathbf{N}(s)$. Based on Frenet-Serret frame, the derivative of equations (6.2)-(6.4) depends on (6.5) and (6.6), and can be computed as follows:

$$\begin{bmatrix} \mathbf{T}'(s) \\ \mathbf{N}'(s) \\ \mathbf{B}'(s) \end{bmatrix} = \begin{bmatrix} 0 & \kappa(s) & 0 \\ -\kappa(s) & 0 & \tau(s) \\ 0 & -\tau(s) & 0 \end{bmatrix} \begin{bmatrix} \mathbf{T}(s) \\ \mathbf{N}(s) \\ \mathbf{B}(s) \end{bmatrix}, \tag{6.7}$$

where $\mathbf{T}'(s) = \mathrm{d}\mathbf{T}(s)/\mathrm{d}s$, $\mathbf{N}'(s) = \mathrm{d}\mathbf{N}(s)/\mathrm{d}s$ and $\mathbf{B}'(s) = \mathrm{d}\mathbf{B}(s)/\mathrm{d}s$ are the first derivatives of such vectors.

Then, the orthogonal basis of the system is defined as $\mathbf{R}(s) := [\mathbf{T}(s)\ \mathbf{N}(s)\ \mathbf{B}(s)]$, which can be integrated from (6.7), based on the functions of $\kappa(s)$ and $\tau(s)$, from an initial value, such that, $\mathbf{R}(0) := [\mathbf{T}(0)\ \mathbf{N}(0)\ \mathbf{B}(0)]$. Hence, the position can be determined by integrating the tangent vector.

$$\mathbf{C}(s) := \mathbf{C}(0) + \int_0^s \mathbf{T}(\xi)\mathrm{d}\xi \tag{6.8}$$

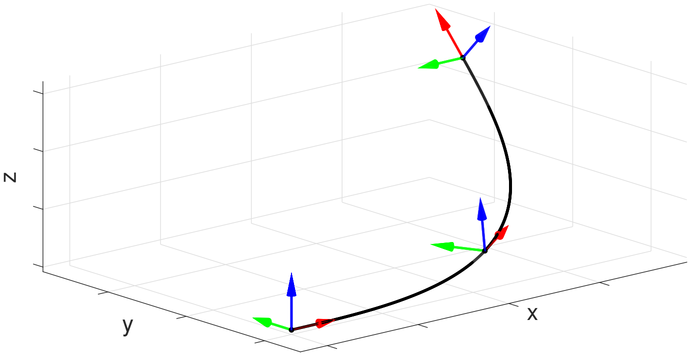


Figure 6.1: Curve in the space $\mathbb{R}^3$, where three orthogonal local systems are shown, which are defined by **T** vectors (red arrows), **N** vectors (green arrows) and **B** vectors (blue arrows).

6.2.2 Clothoid in space $\mathbb{R}^2$ (C2D)

A planar clothoid (C2D) [Levien 2008b], also known as the Euler Spiral, defined in $\mathbb{R}^2$ (see Figure 6.2), is a curve whose curvature varies linearly with respect to the arc length, being:

$$\kappa(s) := \sigma_\kappa s \tag{6.9}$$

where, $\sigma_\kappa := \mathrm{d}\kappa(s)/\mathrm{d}s$ is referred to as the curvature sharpness, which is related to the homotopy factor K, being $\sigma_\kappa := \pi/K^2$. Hence, the tangent angle of the clothoid is defined as:

$$\beta(s, \sigma_\kappa) := \frac{\sigma_\kappa}{2} s^2 \tag{6.10}$$

The C2D is a curve that has contributed in various aspects, both in development and construction of roads and/or railroads [Lambert, Romano y Watling 2019; Vázquez-Méndez y Casal 2016], and also in research performed by Girbés, Armesto y Tornero 2014, giving nonholonomic vehicles a good tracking control, due to its various geometric properties such as curvature and tangent angle. Being the tangent vector computed as:

$$\mathbf{T}(s, \sigma_\kappa) := \begin{bmatrix} \cos(\beta(s, \sigma_\kappa)) \\ \sin(\beta(s, \sigma_\kappa)) \end{bmatrix} \tag{6.11}$$

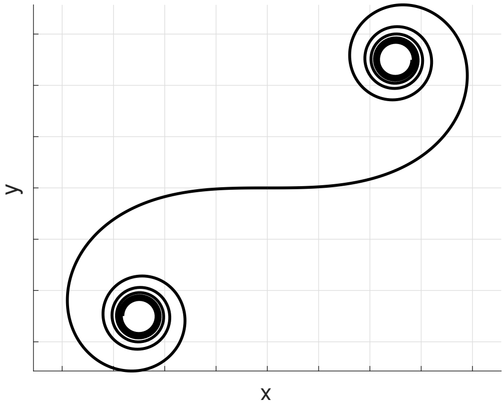


Figure 6.2: Clothoid curve in space $\mathbb{R}^2$.

Consequently, a planar clothoid curve contained in the plane XY, is defined by the equations (6.8), (6.10) and (6.11). Hence, $\mathbf{C}(s, \sigma_\kappa)$ can be solved using the Fresnel integrals, as follows:

$$\mathbf{C}(s, \sigma_\kappa) := \begin{bmatrix} \mathcal{C}(s, \sigma_\kappa) \\ \mathcal{S}(s, \sigma_\kappa) \end{bmatrix} := \begin{bmatrix} \int_0^s \cos(\frac{\sigma_\kappa}{2}\xi^2)\mathrm{d}\xi \\ \int_0^s \sin(\frac{\sigma_\kappa}{2}\xi^2)\mathrm{d}\xi \end{bmatrix} \tag{6.12}$$

where $\mathcal{C}(s, \sigma_\kappa)$ and $\mathcal{S}(s, \sigma_\kappa)$ are the Fresnel integrals in cosine and sine, respectively. Finally, it should be noted that without loss of generality, it is assumed that the clothoid starts from the origin of coordinates, being $\mathbf{C}(0) = 0$ according to the equation (6.8).

6.2.3 Clothoid in space $\mathbb{R}^3$ *(C3D)*

The concept of Euler Spiral defined in $\mathbb{R}^3$, also known as a 3D clothoid (C3D), was introduced by Harary y Tal 2012b. The curvature of a C3D varies as a function of the equation (6.9), while its torsion is defined as:

$$\tau(s) := \sigma_\tau s \tag{6.13}$$

where $\sigma_\tau := \mathrm{d}\tau/\mathrm{d}s$ is the first geometric derivative (also known as torsion sharpness). The development of this 3D smooth curve allows arbitrary configurations to be achieved, either in position or orientation in 3D space, but not both at the same time.

6.3 Problem definition

Let us assume $\mathcal{R}$ as a UAV of nonholonomic characteristics, such as a fixed-wing aircraft, whose state space $\mathbf{q}_{\mathcal{R}} = [x_{\mathcal{R}}\ y_{\mathcal{R}}\ z_{\mathcal{R}}\ \theta_{\mathcal{R}}\ \Psi_{\mathcal{R}}]^T \in \mathbb{R}^3 \times \mathbb{S}^2$, is composed of the coordinates position $\mathbf{p}_{\mathcal{R}} = [x_{\mathcal{R}}\ y_{\mathcal{R}}\ z_{\mathcal{R}}]^T$, and orientation coordinates $\angle\mathbf{q}_{\mathcal{R}} = [\theta_{\mathcal{R}}\ \Psi_{\mathcal{R}}]^T$, pitch and yaw angles. Whereas the input parameters to the system are given by the curvature sharpness $\sigma_{\kappa_{\mathcal{R}}}$ and the torsion sharpness $\sigma_{\tau_{\mathcal{R}}}$. It is important to remark that the roll angle of the curve is not relevant for computing the curve geometrically and, for this reason, it is not considered as part of the configuration.

Assuming that $\mathcal{R}$ can perform motions within its kinematic boundaries of maneuverability (boundaries set by the particular aerodynamic constraints of the UAV) and that the values of the geometric derivatives of curvature and torsion (defined as curvature sharpness and torsion sharpness, respectively), are within the set boundaries, being $\sigma_{\kappa_{\mathcal{R}}} \in [\sigma_{\kappa_{min}}\ \sigma_{\kappa_{max}}]$ and $\sigma_{\tau_{\mathcal{R}}} \in [\sigma_{\tau_{min}}\ \sigma_{\tau_{max}}]$. Then, the aim is to build a new smooth curve G^1 to join two arbitrary configurations in position and orientation, starting from $\mathbf{q}_S = [x_S\ y_S\ z_S\ \theta_S\ \Psi_S]^T$ to $\mathbf{q}_G = [x_G\ y_G\ z_G\ \theta_G\ \Psi_G]^T$. In addition to this, curvature and torsion at start and goal configurations must be zero, i.e., $\kappa_S = \kappa_G = 0$ and $\tau_S = \tau_G = 0$.

6.4 Elementary Clothoid-based 3D Curve (ECb3D)

The aim is to construct a new three-dimensional curve, in the space $\mathbb{R}^3\times\mathbb{S}^2$, able to reach an arbitrary configuration in orientation with continuous curvature and torsion (CC). This new curve will be composed of two segments of the Clothoid-based 3D curve (Cb3D) [Girbés, Vanegas y Armesto 2019].

The procedure starts with the description of the Cb3D curve. Afterwards, the generation process of the new Elementary Clothoid-based 3D Curve (ECb3D), generated from the concatenation of two symmetrical Cb3D curves is detailed. Specifically, the ECb3D will allow reaching an arbitrary orientation, but the position will be given according to the shape and size of the constructive clothoid parameters.

6.4.1 Clothoid-based 3D Curve (Cb3D)

The authors of Girbés, Vanegas y Armesto 2019, propose a new Clothoid-based 3D Curve (Cb3D), capable of achieving an arbitrary configuration in position or orientation in 3D space. The Cb3D is built from two C2D curves generated in the orthogonal XY and XZ planes. Thus, Cb3D projects a C2D curve in the XY plane with arc length s, while the clothoid in the orthogonal XZ plane depends on the length of the C2D curve in the XY plane (see Figure 6.3). The curve Cb3D achieves relevant results due to its analytical solution. Moreover, the curve presents a set of interesting properties/operations such as scalability, symmetry, monotonicity, and smoothness along the curve. Appendix A includes the most relevant aspects of the curve [Girbés, Vanegas y Armesto 2019], which have been included in this work to justify the computations of the proposed methodology.

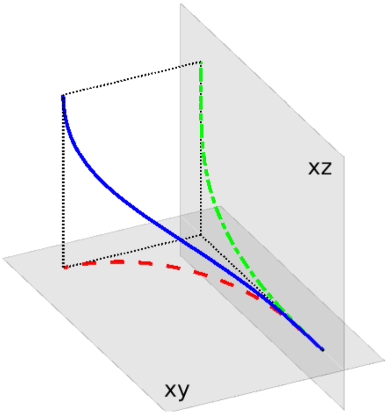


Figure 6.3: Clothoid-based 3D Curve (Cb3D). The dotted red line shows the C2D construction in the XY plane, the green line shows the C2D in the XZ plane while the blue line shows the Cb3D curve. Image taken from Girbés, Vanegas y Armesto 2019

6.4.2 ECb3D Curve Generation

The concept of Elementary Curve in $\mathbb{R}^2$ (E2D) was introduced by Scheuer y Fraichard 1996b, a curve developed to build appropriate paths for mobile wheeled vehicles. In that sense, the E2D is built by combining two symmetric C2D curves that have the same homotopy factor and the same length. The goal is to create smooth trajectories that do not exceed certain physical limits associated with comfort and safety in mobile vehicles [Girbés, Armesto y Tornero 2014], leading to proper path following.

In particular, the ECb3D curve seeks to extend the original concept of E2D curve to the space $\mathbb{R}^3$, by concatenating two symmetric Cb3D curves [Girbés, Vanegas y Armesto 2019]. Continuity and smoothness are guaranteed due to the properties of the original Cb3D curve (see Appendix A for a brief description of Cb3D curves). An ECb3D curve has zero curvature and torsion values at start and goal configurations, that is $\kappa_S = \kappa_G = \tau_S = \tau_G = 0$.

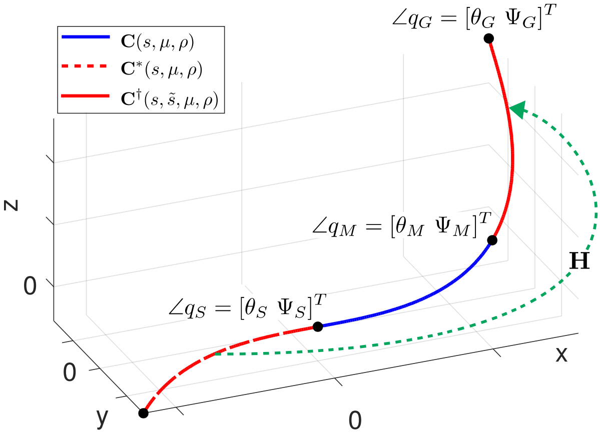

$\mathbf{C}(s,\mu,\rho)$
$\mathbf{C}^*(s,\mu,\rho)$
$\mathbf{C}^\dagger(s,\tilde{s},\mu,\rho)$
$\angle q_G = [\theta_G \;\; \Psi_G]^T$
$\angle q_M = [\theta_M \;\; \Psi_M]^T$
$\angle q_S = [\theta_S \;\; \Psi_S]^T$
H
z
0
0
y
0
x

Figure 6.4: ECb3D curve generation from the symmetry of two concatenated Cb3D curves, through an intermediate deflection angle $\angle\mathbf{q}_M$. It is assumes an initial orientation configuration $\angle\mathbf{q}_S = [\theta_S \;\; \Psi_S]^T = [0 \;\; 0]^T$, while the goal orientation configuration is $\angle\mathbf{q}_G = [\theta_G \;\; \Psi_G]^T$.

Figure 6.4 shows an example of ECb3D curve, where $\mathbf{C}(s,\mu,\rho)$ (in solid blue) is the first Cb3D curve, $\mathbf{C}^*(s,\mu,\rho) = \mathbf{C}(-s,\mu,\rho)$ (in dashed red) is symmetrical to $\mathbf{C}$, and $\mathbf{C}^\dagger(s,\tilde{s},\mu,\rho)$ (in solid red) is the second Cb3D curve, which is $\mathbf{C}^*$ rotated and translated. Consequently, the curve $\mathbf{C}^\dagger$ starts at configuration $\mathbf{q}_M$ and ends at configuration $\mathbf{q}_G$. Note that in Figure 6.4 the start configuration $\mathbf{q}_S$ is at the origin of coordinates aligned with x-axis, so $\angle\mathbf{q}_S = [\theta_S \;\; \Psi_S]^T = [0 \;\; 0]^T$. In a generic case, where start configuration with respect to the global frame is different from zero, that is, $\angle\mathbf{q}_S^0 \neq [0 \;\; 0]^T$, we can compute the *local* tangent vector of $\angle\mathbf{q}_G$, by considering the following rotation:

$$\mathbf{T}_G \equiv \begin{bmatrix} \cos(\theta_G)\cos(\Psi_G) \\ \cos(\theta_G)\sin(\Psi_G) \\ -\sin(\theta_G) \end{bmatrix} := \mathbf{R}^T(0,\theta_S^0,\Psi_S^0) \begin{bmatrix} \cos(\theta_G^0)\cos(\Psi_G^0) \\ \cos(\theta_G^0)\sin(\Psi_G^0) \\ -\sin(\theta_G^0) \end{bmatrix} \tag{6.14}$$

with $\angle\mathbf{q}_S^0 := [\theta_S^0 \;\; \Psi_S^0]^T$ and $\angle\mathbf{q}_G^0 := [\theta_G^0 \;\; \Psi_G^0]^T$ being init and goal angles expressed with respect to a global frame and $\mathbf{R}$ is a rotation matrix

$$\mathbf{R}(\phi, \theta, \Psi) := \mathbf{R}_z(\Psi)\mathbf{R}_y(\theta)\mathbf{R}_x(\phi) \tag{6.15}$$

where $\mathbf{R}_x(\bullet)$, $\mathbf{R}_y(\bullet)$ and $\mathbf{R}_z(\bullet)$ are basic rotation matrices around each axis of the global reference frame; being roll ϕ, pitch θ and yaw Ψ the Euler angles.

Then, the deflection angle $\angle\mathbf{q}_G = [\theta_G\ \Psi_G]^T$ can be computed from (6.14) as follows

$$\theta_G := \arctan\left(\frac{\mathbf{T}_{G,z}}{\sqrt{(\mathbf{T}_{G,x})^2 + (\mathbf{T}_{G,y})^2}}\right) \tag{6.16}$$

$$\Psi_G := \arctan\left(\frac{\mathbf{T}_{G,y}}{\mathbf{T}_{G,x}}\right) \tag{6.17}$$

where subscripts x, y and z refer to each component of the tangent vector $\mathbf{T}_G$.

On the other hand, for the computation of the intermediate deflection angle $\angle\mathbf{q}_M$, the following rotations must be applied to ensure that the curve is continuous (in position, curvature and torsion):

$$\begin{aligned}[\mathbf{T}_G, \mathbf{N}_G, \mathbf{B}_G] =& \mathbf{R}_z(\Psi_M) \cdot \mathbf{R}_y(\theta_M) \cdot \mathbf{R}_y(\theta_M) \cdot \mathbf{R}_z(\Psi_M) \\ \mathbf{T}_G =& \begin{bmatrix} 2\cos^2(\theta_M)\cos^2(\Psi_M) - 1 \\ 2\cos^2(\theta_M)\cos(\Psi_M)\sin(\Psi_M) \\ -2\cos(\theta_M)\cos(\Psi_M)\sin(\theta_M) \end{bmatrix}.\end{aligned} \tag{6.18}$$

where the first column of the right-side expression corresponds to the symbolic expression for computing $\mathbf{T}_G$. Thus, with equations (6.14) and (6.18) we can compute the intermediate deflection angle $\angle\mathbf{q}_M = [\theta_M\ \Psi_M]^T$:

$$\begin{aligned}\theta_M :=& \arctan\left(\frac{\tan(\theta_G)\sin(\Psi_M)}{\sin(\Psi_G)}\right) \\ \Psi_M :=& \arctan\left(\frac{\cos(\theta_G)\sin(\Psi_G)}{\cos(\theta_G)\cos(\Psi_G) + 1}\right).\end{aligned} \tag{6.19}$$

Therefore, the parameters necessary to build Cb3D curves are computed by means of the following expressions (see Appendix):

$$\rho^* := \frac{2\theta_M^*}{\tilde{s}^2} \tag{6.20}$$

$$\mu^* := \frac{2\Psi_M^*}{\mathcal{C}^2(\tilde{s},\rho^*)}, \tag{6.21}$$

being $\tilde{s}$ the length of one Cb3D, that is, the half length of the ECb3D curve.

On the other hand, the transformation governing the rotation and translation of the curve $\mathbf{C}^*(s,\mu,\rho)$ to obtain $\mathbf{C}^\dagger(s,\mu,\rho)$ (see Figure 6.4) is expressed as:

$$\begin{gathered}\mathbf{H} = \begin{bmatrix} 2\mathbf{T}(\tilde{s},\mu,\rho)\mathbf{T}^T(\tilde{s},\mu,\rho) - \mathbf{I} & \mathbf{0}^T \\ 2\mathbf{T}(\tilde{s},\mu,\rho)\mathbf{T}^T(\tilde{s},\mu,\rho)\mathbf{C}(\tilde{s},\mu,\rho) & 1 \end{bmatrix}^T \\ \begin{bmatrix} \boldsymbol{C}^\dagger(s,\tilde{s},\mu,\rho) \\ 1 \end{bmatrix} = \mathbf{H}\cdot \begin{bmatrix} \boldsymbol{C}(-s,\mu,\rho) \\ 1 \end{bmatrix}^T, \end{gathered} \tag{6.22}$$

where, it can be seen that, as the $\mathbf{C}^\dagger$ curve has been defined, when $s = 0$ the curve is in the $\mathbf{q}_M$ configuration, while when $s = \tilde{s}$, the curve ends in the $\mathbf{q}_G$ configuration.

Finally, as a result of the concatenation of the curves $\mathbf{C}$ and $\mathbf{C}^\dagger$, the ECb3D curve is defined as:

$$\mathbf{E}(s,\tilde{s},\mu,\rho) = \begin{cases} \mathbf{C}(s,\mu,\rho) & \text{if } s < \tilde{s} \\ \mathbf{C}^\dagger(2\tilde{s} - s,\tilde{s},\mu,\rho) & \text{otherwise} \end{cases} \tag{6.23}$$

Hence, the ECb3D curve is evaluated for $s \in [0 \;\; 2\tilde{s}]$, with no discontinuity.

Figure 6.5, shows a comparative example of the ECb3D curve (blue line) and the Cb3D curve (dashed red line), both of which point to a goal orientation of $\theta_G = -\pi/4$ and $\Psi_G = \pi/2$ with $\tilde{s} = 0{,}5$. In Figure 6.5(b), the variation of the pitch θ and yaw Ψ Euler angles is displayed, while in Figures 6.5(c) and 6.5(d) the behavior in the curvature profiles and their sharpness in the orthogonal XY and XZ planes can be appreciated. It can be highlighted that the ECb3D curve starts and ends with curvature and torsion equal to zero ($\kappa = \tau = 0$), characteristic that allows the ECb3D to link with another curve without loss of continuity, a property lacking in the Cb3D.

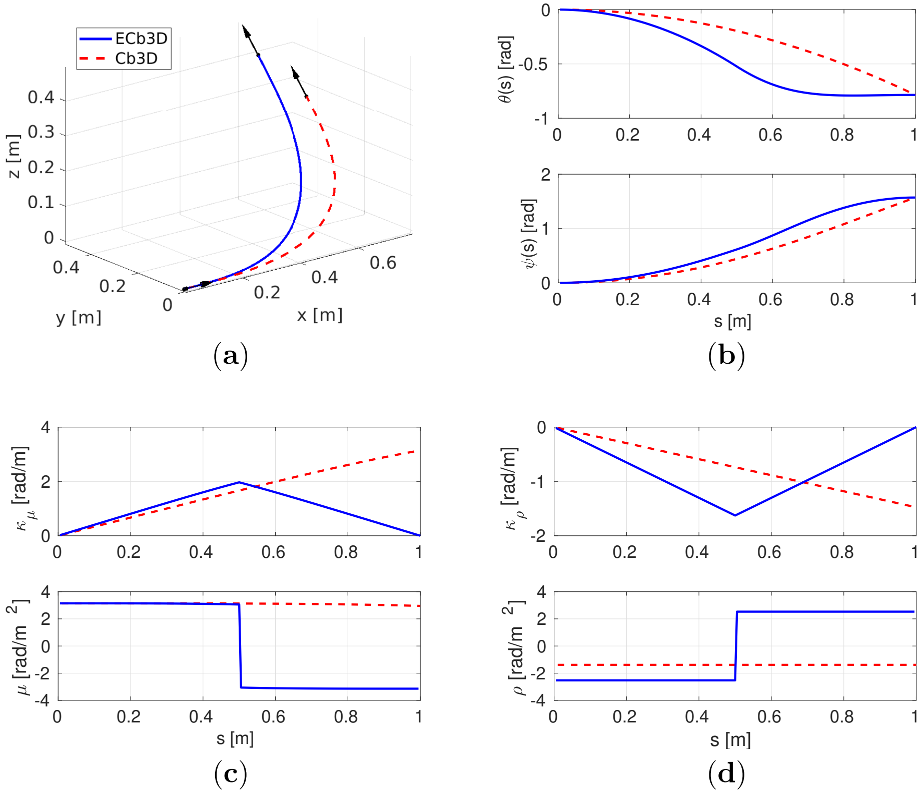


Figure 6.5: Case study with final orientation $\theta_G = -\pi/4$ rad and $\Psi_G = \pi/2$ rad, with $\tilde{s} = 0{,}5$. ECb3D (blue) vs. Cb3D (dashed red): **(a)** position, **(b)** Euler angles, **(c)** curvature profile and its sharpness in the orthogonal plane XY, **(d)** curvature profile and its sharpness in the orthogonal plane XZ.

6.5 Double Continuous Curvature 3D Curve (DCC3D)

This section defines a curve that reaches an arbitrary configuration in position and orientation in the 3D Euclidean space. The new Double Continuous Curvature 3D Curve (DCC3D) is built by three straight line segments and two ECb3D curves, as can be seen in Figure 6.6.

In this sense, since there are two ECb3D paths in a DCC3D, a subscript has been added in order to refer to the corresponding ECb3D. Therefore, configurations $\mathbf{q}_S$, $\mathbf{q}_M$ and $\mathbf{q}_G$ of an ECb3D have been renamed as $\mathbf{q}_{S,i}$, $\mathbf{q}_{M,i}$ and $\mathbf{q}_{G,i}$,

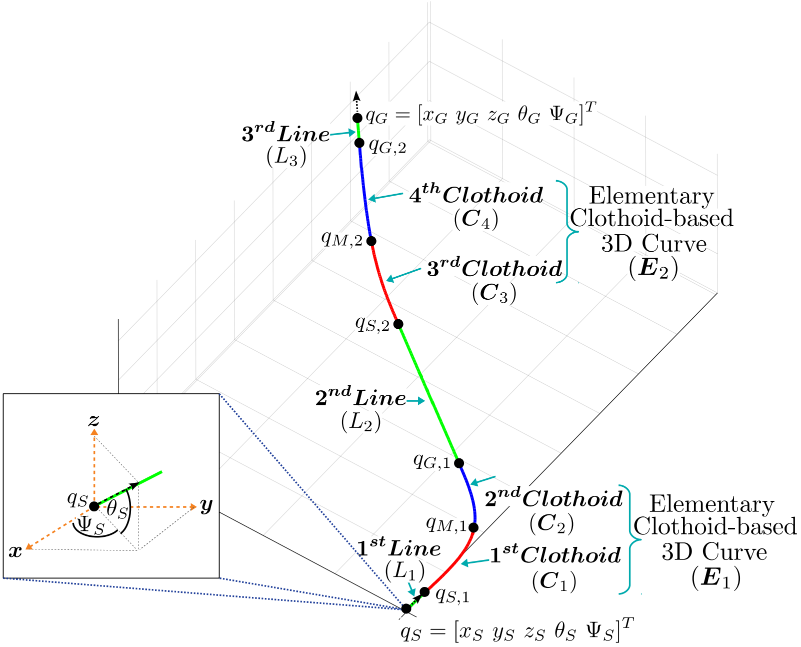


Figure 6.6: Double Continuous Curvature 3D Curve (DCC3D)

respectively, where $i=1$ refers to the first ECb3D, E_1, and $i=2$ to the second one, E_2. Thus, the following equalities relative to configurations' orientations hold: $\angle \mathbf{q}_S \equiv \angle \mathbf{q}_{S_1}$, $\angle \mathbf{q}_{G_1} \equiv \angle \mathbf{q}_{S_2}$, $\angle \mathbf{q}_{G_2} \equiv \angle \mathbf{q}_G$.

Since there are multiple solutions to the problem, the deflection angle $\angle \mathbf{q}_{G,1}$ is assumed to be known, which allows determining the straight lines and clothoids' lengths. That is possible because derivatives of curvature and torsion are performed at the maximum admissible sharpness, that is $|\mu| \leq \mu_{max}$ and $|\rho| \leq \rho_{max}$. This allows to obtain the shortest path for the given deflection angle in a preliminary step. Thus, this states a numerical optimization problem that aims to obtain the shortest curve by finding the intermediate orientation $\angle \mathbf{q}_{G,1}$ that minimizes the overall length, while ensuring that the path satisfies some constraints to avoid abrupt changes in curvature and torsion. As shown later on, the lengths of the first and third straight lines act as *relaxation variables* in order to satisfy position constraints for a given $\angle \mathbf{q}_{G,1}$.

6.5.1 DCC3D curve generation

Let us assume that the intermediate orientation of the DCC3D curve is given, that is, $\angle \mathbf{q}_{G,1}$. Let us also assume that in order to satisfy position constraints imposed by $\mathbf{q}_G$, we allow maneuvers, that is, changes in the direction of the curve that would force a robot stop to avoid abrupt changes in curvature and torsion. The steps to generate the DCC3D curve are described next.

1. **Compute the first ECb3D**: the intermediate deflection angle $\angle \mathbf{q}_{M,1} = [\theta_{M,1}\ \Psi_{M,1}]^T$ is determined from equation (6.19) using the orientation of the goal configuration $\angle \mathbf{q}_{G,1} = [\theta_{G,1}\ \Psi_{G,1}]^T$ (assuming that the start configuration $\angle \mathbf{q}_{S,1} = [0\ 0]^T$). Otherwise, a rotation must be applied to compute $\angle \mathbf{q}_{G,1}$ with respect to the local frame of $\angle \mathbf{q}_{S,1}$, as in equation (6.14).

   Assuming that the curve is computed with the maximum allowable torsion sharpness $\rho_1 := \rho_{max}$, the arc-length of each Cb3D curve (of the ECb3D), and the curvature sharpness to reach the deflection angle $\angle \mathbf{q}_{M,1}$ can be obtained from (6.20) and (6.21) as follows:

$$\tilde{s}_1 := \sqrt{\frac{2|\theta_{M,1}|}{\rho_{max}}}, \quad \mu_1 := \frac{2\Psi_{M,1}}{\mathcal{C}^2(\tilde{s}_1, \rho_{max})} \tag{6.24}$$

   If $|\mu_1| > \mu_{max}$, then it means that the assumption of the previous point was incorrect and, therefore, the curve with the maximum admissible curvature sharpness $\mu_1 := \mu_{max}$ must be computed as:

$$\tilde{s}_1 := \frac{\sqrt{\frac{2|\Psi_{M,1}|}{\mu_{max}}}}{\mathcal{C}(1, 2|\theta_{M,1}|)}, \quad \rho_1 := \frac{2\theta_{M,1}}{\tilde{s}_1^2}. \tag{6.25}$$

2. **Compute the second ECb3D**: repeat the previous step to obtain curvature μ_2 and torsion ρ_2 sharpness parameters, as well as the arc-length $\tilde{s}_2$ based on deflection angle between $\angle \mathbf{q}_{S,2} \equiv \angle \mathbf{q}_{G,1}$ and $\angle \mathbf{q}_{G,2} \equiv \angle \mathbf{q}_G$.

3. **Compute line segments**: it is interesting to remark that the resulting curve, as a consequence of concatenating the two ECb3D, satisfies orientation constraints, but it does not satisfy position constraints. Indeed, the position displacement imposed by the clothoid arcs is:

$$\mathbf{p}_{E_1,E_2} := \mathrm{E}_1(2\tilde{s}_1) + \mathrm{E}_2(2\tilde{s}_2) \tag{6.26}$$

Thus, line segments must generate the remainder position displacement in order to satisfy position constraints:

$$\Delta \mathbf{p} = \mathbf{p}_G - \mathbf{p}_S - \mathbf{p}_{E_1,E_2} \tag{6.27}$$

As a consequence, lengths of line segments can be computed as:

$$\mathbf{L} = [\mathbf{T}_S \ \mathbf{T}_{G,1} \ \mathbf{T}_G]^{\dagger} \, \Delta \mathbf{p} \tag{6.28}$$

where † denotes the Moore-Penrose pseudo-inverse, $\mathbf{T}_S := [1\ 0\ 0]^T$ and

$$\mathbf{T}_{G,1} := \begin{bmatrix} \cos(\theta_{G,1})\cos(\Psi_{G,1}) \\ \cos(\theta_{G,1})\sin(\Psi_{G,1}) \\ -\sin(\theta_{G,1}) \end{bmatrix} \tag{6.29}$$

Obviously, when the rank of $[\mathbf{T}_S \ \mathbf{T}_{G,1} \ \mathbf{T}_G]$ is not 3, i.e., the matrix has one or two singular values close to zero, it implies that line segments are redundant. In those cases, we can force to zero one of the line segments and compute the others. In particular, we propose to force to zero L_1 and L_3 before forcing to zero L_2, details left to the reader.

6.5.2 Shortest curve

The aim now is to compute the curve with the shortest length, being $\angle \mathbf{q}_{G,1}$ decision variables of an optimization process:

$$\begin{gathered} \angle \mathbf{q}^*_{G,1} = \arg \min_{\angle \mathbf{q}_{G,1}} L_1 + 2\tilde{s}_1 + L_2 + 2\tilde{s}_2 + L_3, \\ \text{subject to } L_1, L_2, L_3 \geq 0 \end{gathered} \tag{6.30}$$

Note that values of L_1, L_2, L_3, $\tilde{s}_1$ and $\tilde{s}_2$ depend on the deflection angle as discussed in the previous section. Since this procedure looks for solutions of maximum sharpness in torsion or curvature of the clothoid arcs, the curve resulting after solving (6.30) will produce the shortest length that joins the initial and goal configurations without discontinuities. Also note that the positiveness constraints imposed to L_1, L_2 and L_3 ensure that the obtained curve will imply no maneuvers, and thus are suitable for fixed-wing UAVs.

Lemma 6.5.1 *The optimal solution will imply that lengths for the first and third lines are zero.*

Proof 6.5.1 *For the unbounded problem, where $\rho_{max} = \infty$ and $\mu_{\text{máx}} = \infty$, the clothoid lengths have zero length, i.e.: $\tilde{s}_1 = 0$ and $\tilde{s}_2 = 0$, which implies an instantaneous change of orientation between the line segments. In that case, it is obvious that the optimal solution joining two points in the Cartesian space is $\angle \mathbf{q}^*_{G,1} = [\arctan(\frac{\Delta \mathbf{p}_y}{\Delta \mathbf{p}_x}) \ \arctan(\frac{-\Delta \mathbf{p}_z}{\sqrt{\Delta \mathbf{p}_x^2 + \Delta \mathbf{p}_y^2}})]^T$, which implies that the lengths of the line segments are $L_1 = 0$, $L_2 = |\Delta \mathbf{p}|$ and $L_3 = 0$. The optimal solution of the bounded problem will generate clothoids with the shortest possible length such as the angle between $\mathbf{T}_{G,1}$ and $\Delta \mathbf{p}$ is minimum, which implies that numerically $L_1 \to 0$ and $L_3 \to 0$, details left for.* ■

6.5.3 Computation time

In order to test the computational effort to generate a DCC3D curve, a total of 10^4 arbitrary goal positions and orientations have been used to analyse time performance. The computational time has been divided into two stages. In the first stage, the optimization process to find the intermediate configuration angle $\angle \mathbf{q}^*_{G,1}$ that minimizes the overall length, obtaining an average time of $\bar{t}_{\mathbf{q}^*_{G,1}} =$ 0,110 s. The second timing analysis focuses on determining the computational time to get μ and ρ values for each Elementary segment, $E1$ and $E2$. The mean time to compute such parameters is $\bar{t}_E = 0{,}0024$ s.

The computer used for this analysis has the following specifications: CPU Intel i5-9400F 4.100GHz, GPU NVIDIA GeForce GT 610 and 8 GB DDR4 memory, under an Ubuntu 18.04.4 LTS x86_64 OS with Kernel 5.4.0-120-generic.

6.5.4 Case study

To achieve a better understanding of the results obtained, a specific case study with three variants is described next. Figure 6.7 shows an example of DCC3D curves joining the initial configuration $\mathbf{q}_S = [0\ 0\ 0\ 0\ 0]^T$ with a goal configuration $\mathbf{q}_G = [170\ 120\ 90\ \pi/4\ \pi/6]^T$. The aim is to describe the behavior of the DCC3D curve by setting three different values for the maximum curvature sharpness and torsion sharpness. Figure 6.7(a) shows the curves obtained in 3D space. In Figure 6.7(b), the Euler orientation angles along the curves length are shown, while in Figures 6.7(c) and 6.7(d), the curvature and torsion profiles together with their sharpness are depicted.

As expected, lower values of maximum curvature sharpness and torsion sharpness yield longer and smoother ECb3D curves. That implies shorter straight lines and, consequently, longer DCC3D curves. It is interesting to remark that,

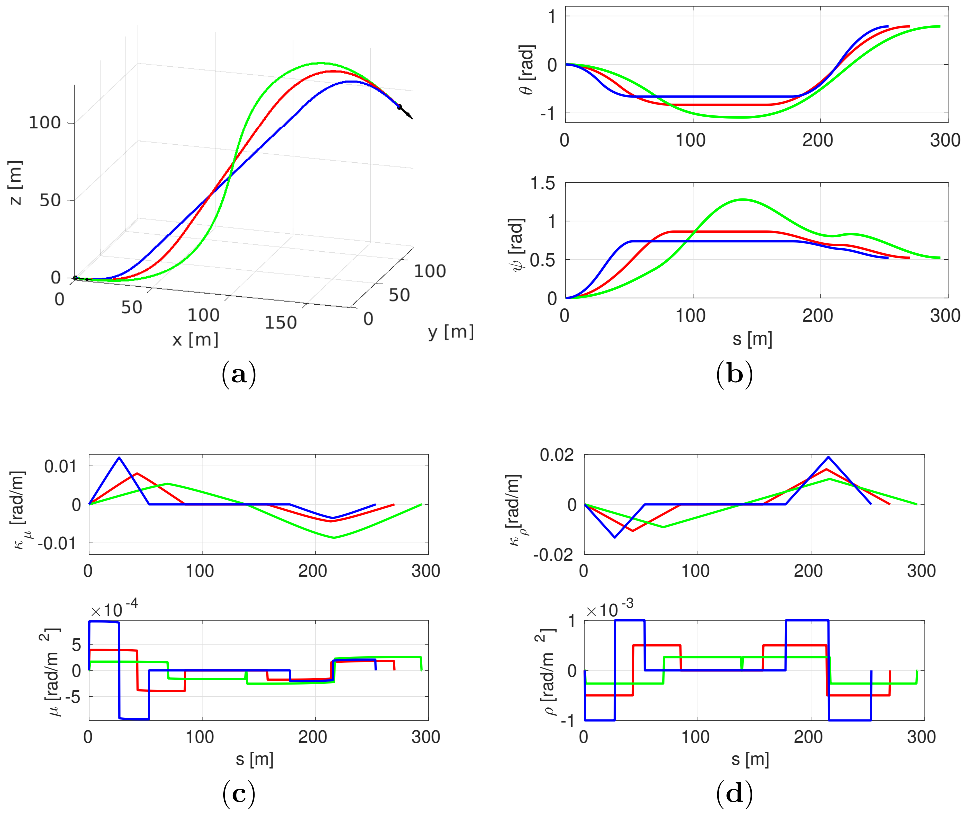


Figure 6.7: DCC3D curve with final orientation $\theta_G = \pi/4$ rad and $\Psi_G = \pi/6$ rad, final position $\mathbf{p_G} = [170\ 120\ 90]^T$ m, with $\rho_{max} = \mu_{max} = 0{,}001$ (blue), $\rho_{max} = \mu_{max} = 0{,}0005$ (red), and $\rho_{max} = \mu_{max} = 0{,}00025$ (green). **(a)** Smooth path 3D, **(b)** Euler angles, **(c)** curvature profile and its sharpness in the orthogonal plane XY, **(d)** Curvature profile and its sharpness in the orthogonal plane XZ.

reaching positions too close would imply curvy trajectories with a very pronounced S shape (or even an 8 shape) or a complete loop, i.e. turning to the opposite direction. If we do not seek for such type of curves, then the target configuration must be consistent with the sharpness constraints and the initial deflection angle can be set to $\angle \mathbf{q}_{G,1}^{initial} = [\arctan(\frac{y_G}{x_G})\ \ \arctan(\frac{-z_G}{\sqrt{x_G^2+y_G^2}})]^T$.

6.6 Flight Simulation Experiments and Results

In order to realistically reproduce the Kadett 2400 fixed-wing UAV dynamic model, a setup with the Flight-Gear 2019 simulator and Matlab R2020b has been used for the low-level control of the different subsystems shown in Figure 6.8. For this experimentation the same computer as in Section 6.5.2 has been used.

The UAV has four actuators δ_e (elevation), δ_a (aileron), δ_r (rudder) and δ_{th} (throttle), which are controlled by three PID regulators implementing low-level control of the yaw Ψ^* and pitch θ^* angles, as well as the ground speed vel^*. The yaw and pitch references are obtained from the yaw and pitch angles of the designed smooth curve, while the forward velocity reference is kept constant at $vel^* = 18$ m/s throughout the simulation. Notice that in our implementation the aileron and rudder angles are actuated by the same controller, which directly affects the yaw orientation.

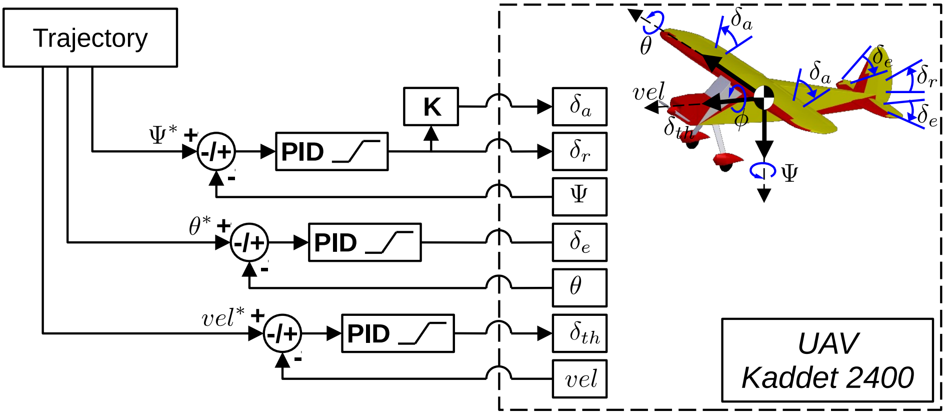


Figure 6.8: Kadett 2400 control model with 3 inputs to control roll, yaw and pitch which modify the aerodynamic surfaces δ_e (elevator), δ_a (aileron), δ_r (rudder).

As mentioned before in the article, a nice characteristic of the DCC3D curve is to maintain zero curvature and torsion values, both at the beginning and at the end of the path. Therefore, a concatenation of DCC3D curves can be easily performed, producing a smooth flight path without any in-flight lift loss. Figure 6.9 shows an example of a smooth flight, where three DCC3D paths have been concatenated, being the maximum values of Cb3D sharpness parameters $\mu_{max} = 0{,}001$ rad/m^2 and $\rho_{max} = 0{,}001$ rad/m^2. Without loss of generality, the path begins at initial configuration $\mathbf{q}_S$ (the UAV in horizontal flight), passes through two intermediate configurations $\mathbf{q}_A$ and $\mathbf{q}_B$, and finishes at goal configuration $\mathbf{q}_G$:

$$\begin{aligned}
\mathbf{q}_S &= \begin{bmatrix} 0{,}0 & 0{,}0 & 0{,}0 & 0{,}0 & 0{,}0 & 0{,}0 \end{bmatrix}^T \\
\mathbf{q}_A &= \begin{bmatrix} 480{,}0 & 200{,}0 & 20{,}0 & 0{,}0 & -0{,}4 & 0{,}3 \end{bmatrix}^T \\
\mathbf{q}_B &= \begin{bmatrix} 1000{,}0 & 440{,}0 & 28{,}0 & 0{,}0 & 0{,}2 & 0{,}2 \end{bmatrix}^T \\
\mathbf{q}_G &= \begin{bmatrix} 1400{,}0 & 600{,}0 & 56{,}0 & 0{,}0 & -0{,}6 & 0{,}1 \end{bmatrix}^T
\end{aligned}$$

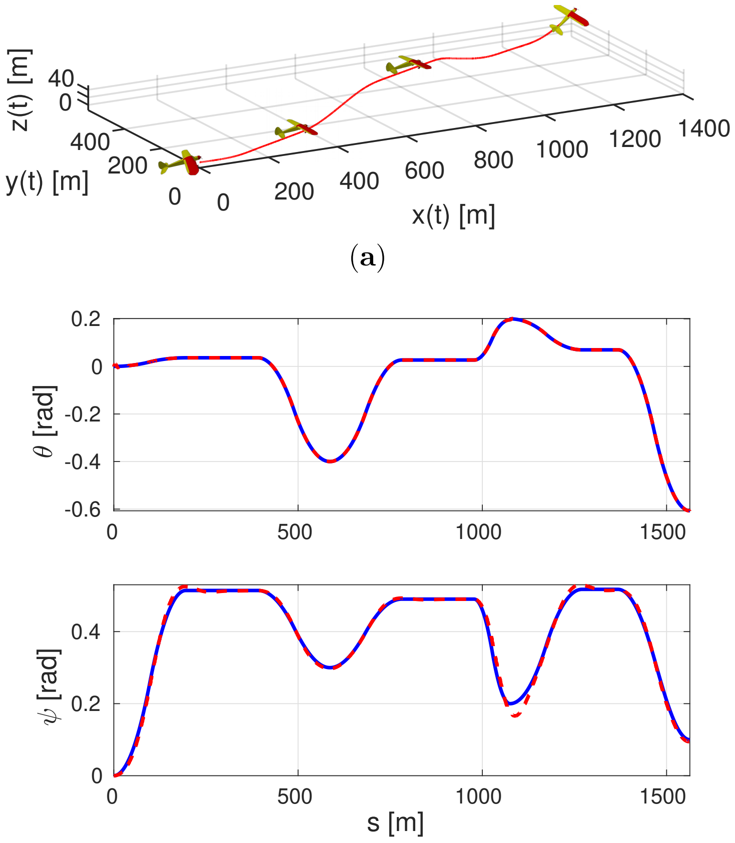
Figure 6.9(a) shows the reference path followed by the UAV. Figure 6.9(b) shows the Euler angles, where the reference angles of the DCC3D curve are the blue lines, and the UAV angles generated by the in-flight tracking system are the dashed red lines. In Figure 6.9 it can also be seen that the errors during the curve tracking in yaw and pitch angles are small, being the mean square error of $\epsilon_\theta = 0{,}00042$ rad and $\epsilon_\Psi = 0{,}006$ rad, respectively, where, $\epsilon_\theta = \frac{1}{s_T}\int_0^{s_T}(\theta^* - \theta)\mathrm{d}s$ and $\epsilon_\Psi = \frac{1}{s_T}\int_0^{s_T}(\Psi^* - \Psi)\mathrm{d}s$. Finally, it is important to mention that the flight time is $t = 86{,}68$ s with a full flight path of $s_T = 1560{,}28$ m.

6.7 Conclusions

The work developed in this paper presents a novel 3D smooth path planner based on a concatenation of Clothoid-based 3D curves (Cb3D) and straight line segments. The planner solves the problem of joining two arbitrary configurations (position and orientation) in 3D space (without loss of continuity and smoothness), making it suitable for navigation of fixed-wing UAVs.

To generate DCC3D paths, a new 3D smooth curve called Elementary Clothoid-based 3D Curve (ECb3D) has been presented. An ECb3D can achieve any orientation with zero curvature and torsion values both at initial and final configurations. That feature allows ECb3D paths to be used as primitives for path generation, since it is possible to smoothly concatenate several ECb3D paths and straight lines without loss of curvature and torsion continuity.

The computational time of the curve, including finding the curve with the shortest distance takes, in average about 0,1 on an Intel i5 (seed details in section 5). Hence, the fast computation of the proposed DCC3D, allows to generate 3D smooth paths in real-time.



(a)

(b)

Figure 6.9: DCC3D curve and flight simulation with four concatenated configurations.

As further work, we aim to include the proposed trajectory in global planners such as Randomized Path Planner (RPP), Probabilistic Road Map Method (PRM), Rapidly Exploring Random Tree (RRT). A different use of DCC3D curve would be the development of a local path planner for obstacle avoidance in 3D maneuvers. Finally, the same methodology could be implemented using other transition curves in order to compare the performance against the Cb3D curves used in this work.

Capítulo 7

Conclusiones y Trabajos Futuros

Resumen: *El trabajo de tesis se ha llevado a cabo con el objetivo de desarrollar un planificador de trayectorias suaves 3D, basado en curvas Clotoides. En particular, se ha diseñado un generador de trayectorias que permite unir 2 configuraciones en el espacio (posición y orientación), con una curvatura continua y de valor acotado. Finalmente, es importante resaltar que la nueva curva 3D basada en Clotoide (Cb3D) se puede integrar fácilmente en un planificador de trayectorias 3D, como se ha demostrado a lo largo de la experimentación detallada en los diferentes capítulos de la tesis y se describe a través de las conclusiones que se presentan a continuación.*

7.1 Conclusiones

La clotoide o espiral de Cornú, es una curva que posee resolución numérica, lo que implica un coste computacional elevado. En consecuencia, para resolver ésta ecuación es necesario definir sus particulares, traducidas como la longitud de arco y el radio de curvatura en cada curva, lo que posibilita su resolución a partir de la serie de Taylor. Por lo tanto, en cuanto a la generación de las trayectorias Cb3D, se han alcanzado varios órdenes de magnitud más rápidos, ya que al ser aproximadas de forma cerrada, su cálculo se completa de manera analítica, lo que significa ventajas en términos de tiempo de cálculo para la planificación en tiempo real.

Para la construcción de la Cb3D se han evaluado algunas métricas geométricas y cinemáticas. Por un lado, las métricas geométricas (que sólo dependen de la longitud de arco s) comprenden los valores medios y máximos de curvatura $\kappa(s)$, torsión $\tau(s)$ y sus derivadas, la brusquedad de curvatura $\sigma_\kappa(s)$ y la brusquedad de torsión $\sigma_\tau(s)$. Por otra parte, las variables cinemáticas (que dependen del tiempo t) que son los valores medios y máximos de la aceleración $a(t)$, la sacudida $j(t)$, la velocidad angular $\omega(t)$ y la aceleración angular $\alpha(t)$. Esto puede parecer poco relevante desde el punto de vista de la geometría diferencial, no obstante, desde la perspectiva de la planificación y navegación de UAVs sí es relevante, como se puede apreciar en los valores máximos de bruzquedad de torsión y bruzquedad de curvatura, descritos a partir de un ejemplo gráfico en el capítulo 5.

Una característica relevante alcanzada en la tesis ha sido enmarcada en la capacidad de concatenación curvas. Es decir, se generó la posibilidad de unir varias curvas, ya sean curvas de tipo Cb3D o rectas, proceso completado de forma correcta, sin pérdida de continuidad en la curva final.

Es importante destacar que todas y cada una de las contribuciones alcanzadas en el trabajo de tesis han sido descritas en cada capítulo, lo cual detalla cada conclusión parcial. En este sentido, a continuación, se ponen en evidencia diferentes conclusiones generales de la tesis como un todo, finalizando con las propuestas de los posibles trabajos futuros.

- *Estudio y descripción de un conjunto de diferentes curvas continuas y suaves en el espacio Euclídeo 2D y 3D.*

  En el capítulo 2 realiza un breve estudio del estado del arte (análisis enfocado en curvas en el espacio Euclídeo), se estudian diferentes tipos de curvas, sus propiedades, sus alcances y limitaciones. El objetivo del capítulo 2 es lograr definir la base metodológica que se usa como punto de partida en la construcción de curvas suaves 3D y que han contribuido en los diferentes aportes durante el transcurso de la tesis. De esta forma, a partir de las conclusiones definidas en el capítulo 2 ha sido posible la publicación más relevante del trabajo de tesis, la que se detalla en el capítulo 4. Ahí se propone la base matemática Cb3D, definida como primitiva de trabajo para los aportes de los capítulos 5 y 6.

- *Aproximación C3D mediante Bézier-Racional.*

  Las curvas spline, y en particular las curvas Bézier y Bézier Racionales han sido muy explotadas por la comunidad científica que se dedica a la construcción de planificadores de trayectorias suaves. En este sentido, el capítulo 3 realiza una aproximación hacia la curva C3D mediante Bézier-Racionales. El objetivo es medir el rendimiento de C3D contra una curva ya experimentada en la resolución del problema de planificación de trayectorias, resultados que muestran un desempeño aceptable, lo cual dio cabida a la continuación del trabajo de la presente tesis.

- *Definición del concepto Curva 3D basada en Clotoide (Cb3D).*

  A lo largo del capítulo 4 se describe el concepto de Curva 3D basada en Clotoide (Cb3D). Es importante resaltar que el capítulo describe diversas propiedades intrínsecas al nuevo concepto de curva suave y que la hace muy apropiada para completar la tarea de planificación de trayectorias suaves en el espacio 3D.

- *Curva Elemental 3D basada en Clotoide (ECb3D).*

  Un nuevo concepto de curva suave se propone en el capítulo 5. De esta forma, las diferentes propiedades alcanzadas y descritas en el capítulo 4 han sido ampliadas y demostradas en este capítulo, agregando además límites de curvatura κ y torsión τ cero, tanto al inicio como al final de la curva. Además, se garantiza que las transiciones entre segmentos de curva sean suaves y cumplan las restricciones cinemáticas del modelo UAV.

- *Curva 3D de Doble Curvatura Continua basada en Clotoide (DCC3D).*

  Por último, en el capítulo 6 se describe un concepto adicional, el cual une segmentos diferentes en una sola curva, que conserva el concepto de curvatura κ y torsión τ cero tanto al inicio como al final de la curva. Este concepto mantiene transiciones continuas entre segmentos, además de definir posición y orientación 3D tanto al inicio como al final de la curva. Características que parten de la primitiva definida en el capítulo 4, lo cual indica que el punto de partida posee un enfoque adecuado y ha permitido el desarrollo continuo de los aportes en la tesis.

7.2 Trabajos Futuros

Las contribuciones presentadas y aceptados en el trabajo de tesis obedecen a una secuencia de ideas que parten de las curvas clotoidales. Es así que el primer concepto denominado como *Curva 3D basada en Clotoide (Cb3D)* ha dado lugar a nuevos aportes, cada uno proponiendo un avance con respecto al anterior.

De esta forma, un primer posible trabajo a futuro, está centrado en la concatenación de curvas de tipo Curva 3D de *Doble Curvatura Continua basada en Clotoide (DCC3D)*, que toma los resultados de un planificador de trayectoria tipo RRT y alcanza una trayectoria suave en entornos con obstáculos estáticos.

Por otra parte, la tesis asume un solo UAV de características definidas, en un entorno de vuelo libre de obstáculos. Entonces, otro posible enfoque de trabajo se puede describir sobre entornos aéreos con varios UAV en vuelo simultáneo, además de la posibilidad de enfrentarse de forma simultánea a obstáculos estáticos y/o dinámicos.

Finalmente, retomando el punto de partida Cb3D, se debe recordar que los conceptos adicionales (ECb3D y DCC3D) pueden ser ampliados al agregar parámetros con incidencia de tiempo que mejoren las curvas, a través de procesos de optimización sobre diversos UAVs.

Appendix

Apéndice A

Proof of Lemma

A.1 Scalability

First, we can see that for the planar clothoid, the scalability property holds. For the tangent angle in Eq. (4.7), it can be seen that $\beta(s, \mathbf{p}) = \beta(\lambda s, \boldsymbol{\lambda}^{-1} \cdot \mathbf{p})$. Therefore, we can also see that $\mathbf{T}(s, \mathbf{p}) = \mathbf{T}(\lambda s, \boldsymbol{\lambda}^{-1} \cdot \mathbf{p})$ in Eq. (4.8) and so $\lambda \mathbf{C}(s, \mathbf{p}) = \mathbf{C}(\lambda s, \boldsymbol{\lambda}^{-1} \cdot \mathbf{p})$ in Eq. (4.9).

Now, we will proof that the same properties hold for Eq. (4.12). Following the same argumentations, it can clearly seen that $\lambda \mathbf{C}_1(s, \mathbf{p}_1) = \mathbf{C}_1(\lambda s, \boldsymbol{\lambda}^{-1} \cdot \mathbf{p}_1)$ and $\lambda \mathbf{C}_2(s, \mathbf{p}_2) = \mathbf{C}_2(\lambda s, \boldsymbol{\lambda}^{-1} \cdot \mathbf{p}_2)$. Therefore, it can also be asserted that

$$\lambda \mathcal{C}(\mathcal{C}(s, \mathbf{p}_2), \mathbf{p}_1) = \mathcal{C}(\lambda \mathcal{C}(s, \mathbf{p}_2), \boldsymbol{\lambda}^{-1} \cdot \mathbf{p}_1) = \mathcal{C}(\mathcal{C}(\lambda s, \boldsymbol{\lambda}^{-1} \cdot \mathbf{p}_2), \boldsymbol{\lambda}^{-1} \cdot \mathbf{p}_1) \tag{A.1}$$

$$\lambda \mathcal{S}(\mathcal{C}(s, \mathbf{p}_2), \mathbf{p}_1) = \mathcal{S}(\lambda \mathcal{C}(s, \mathbf{p}_2), \boldsymbol{\lambda}^{-1} \cdot \mathbf{p}_1) = \mathcal{S}(\mathcal{C}(\lambda s, \boldsymbol{\lambda}^{-1} \cdot \mathbf{p}_2), \boldsymbol{\lambda}^{-1} \cdot \mathbf{p}_1) \tag{A.2}$$

$$-\lambda \mathcal{S}(s, \mathbf{p}_2) = -\mathcal{S}(\lambda s, \boldsymbol{\lambda}^{-1} \cdot \mathbf{p}_2) \tag{A.3}$$

A.2 Symmetry

Let's first assert that a planar clothoid is a symmetric curve, such that $\mathbf{C}(s,\mathbf{p}) = -\mathbf{C}(-s,\mathbf{q}\cdot\mathbf{p})$, which can be easily proven from Eq. (4.9) by considering that$\beta(s,\mathbf{p}) = \beta(-s,\mathbf{q}\cdot\mathbf{p})$ in Eq. (4.7) and $\mathbf{T}(s,\mathbf{p}) = \mathbf{T}(-s,\mathbf{q}\cdot\mathbf{p})$ in Eq. (4.8) can be asserted too. Therefore, we can also prove that

$$\mathcal{C}(\mathcal{C}(s,\mathbf{p}_2),\mathbf{p}_1) = -\mathcal{C}(-\mathcal{C}(s,\mathbf{p}_2),\mathbf{q}\cdot\mathbf{p}_1) = -\mathcal{C}(\mathcal{C}(-s,\mathbf{q}\cdot\mathbf{p}_2),\mathbf{q}\cdot\mathbf{p}_1) \tag{A.4}$$

$$\mathcal{S}(\mathcal{C}(s,\mathbf{p}_2),\mathbf{p}_1) = -\mathcal{S}(-\mathcal{C}(s,\mathbf{p}_2),\mathbf{q}\cdot\mathbf{p}_1) = -\mathcal{S}(\mathcal{C}(-s,\mathbf{q}\cdot\mathbf{p}_2),\mathbf{q}\cdot\mathbf{p}_1) \tag{A.5}$$

$$\mathcal{S}(s,\mathbf{p}_2) = -\mathcal{S}(-s,\mathbf{q}\cdot\mathbf{p}_2) \tag{A.6}$$

A.3 Smoothness

A C^∞ function is differentiable for all degrees of differentiation, i.e. its position vector can be derived ∞ times. Let's first proof that the planar clothoid (Eq. (4.9)) is C^∞. Its n-th derivative is continuous and differentiable, and can be expressed as

$$\mathbf{C}^{(n)}(s,\mathbf{p}) = \begin{bmatrix} \tilde{c}_{1,n}(s,\mathbf{p})\sin(\beta(s,\mathbf{p})) + \tilde{c}_{2,n}(s,\mathbf{p})\cos(\beta(s,\mathbf{p})) \\ \bar{c}_{1,n}(s,\mathbf{p})\sin(\beta(s,\mathbf{p})) + \bar{c}_{2,n}(s,\mathbf{p})\cos(\beta(s,\mathbf{p})) \end{bmatrix} \tag{A.7}$$

for some polynomials $\tilde{c}_{1,n}(s,\mathbf{p})$, $\tilde{c}_{2,n}(s,\mathbf{p})$, $\bar{c}_{1,n}(s,\mathbf{p})$, $\bar{c}_{2,n}(s,\mathbf{p})$.

It is then clear that the n-th derivative of Eq. (4.12) can be expressed as

$$\mathbf{C}^{(n)}(s,\mathbf{p}_1,\mathbf{p}_2) = \begin{bmatrix} \tilde{d}_{1,n}(\star)\sin(\psi(\bullet)) + \tilde{d}_{2,n}(\star)\cos(\psi(\bullet)) \\ \bar{d}_{1,n}(\star)\sin(\psi(\bullet)) + \bar{d}_{2,n}(\star)\cos(\psi(\bullet)) \\ \hat{d}_{1,n}(\bullet)\sin(\theta(\bullet)) + \hat{d}_{2,n}(\bullet)\cos(\theta(\bullet)) \end{bmatrix} \tag{A.8}$$

where $\tilde{d}_{1,n}(\star)$, $\tilde{d}_{2,n}(\star)$, $\bar{d}_{1,n}(\star)$, and $\bar{d}_{2,n}(\star)$ denote some polynomials and trigonometric expressions related to $\theta(\bullet)$ and its derivatives (which are also continuous and differentiable); while $\hat{d}_{1,n}(\bullet)$ and $\hat{d}_{2,n}(\bullet)$ are some polynomials of s, as long as $\epsilon_0\neq 0$ or $\mu\neq 0$; $\upsilon_0\neq 0$ or $\rho\neq 0$. Details omitted for brevity.

A.4 Orientation

From fundamental theorem of calculus and using the chain rule, we get the following equality

$$\frac{d}{dx}\int_{a(x)}^{b(x)} f(t)dt = f(b(x))b'(x) - f(a(x))a'(x) \tag{A.9}$$

being $f(t)$ an integrable function over the range $[a(x), b(x)]$, with $a(x)$ and $b(x)$ derivable functions.

From Eq. (4.12) a Cb3D curve can be expressed explicitly as

$$\mathbf{C}(s, \mathbf{p}_1, \mathbf{p}_2) := \begin{cases} \int_0^{\mathcal{C}(s,\mathbf{p}_2)} \cos(\psi_0 + \epsilon_0 \xi + \frac{\mu}{2}\xi^2)\, d\xi \\ \int_0^{\mathcal{C}(s,\mathbf{p}_2)} \sin(\psi_0 + \epsilon_0 \xi + \frac{\mu}{2}\xi^2)\, d\xi \\ -\int_0^{s} \sin(\theta_0 + \upsilon_0 \xi + \frac{\rho}{2}\xi^2)\, d\xi \end{cases} \tag{A.10}$$

with

$$\mathcal{C}(s, \mathbf{p}_2) := \int_0^{s} \cos(\theta_0 + \upsilon_0 \xi + \frac{\rho}{2}\xi^2)\, d\xi \tag{A.11}$$

Hence, from Eqs. (4.3) and (4.12) the tangent vector can be derived from position vector, such that

$$\mathbf{C}'(s, \mathbf{p}_1, \mathbf{p}_2) := \begin{cases} \cos(\psi_0 + \epsilon_0 \mathcal{C}(s, \mathbf{p}_2) + \frac{\mu}{2}\mathcal{C}^2(s, \mathbf{p}_2))\mathcal{C}'(s, \mathbf{p}_2) \\ \sin(\psi_0 + \epsilon_0 \mathcal{C}(s, \mathbf{p}_2) + \frac{\mu}{2}\mathcal{C}^2(s, \mathbf{p}_2))\mathcal{C}'(s, \mathbf{p}_2) \\ -\sin(\theta_0 + \upsilon_0 s + \frac{\rho}{2}s^2) \end{cases} \tag{A.12}$$

where

$$\mathcal{C}'(s, \mathbf{p}_2) := \cos(\theta_0 + \upsilon_0 s + \frac{\rho}{2}s^2) \tag{A.13}$$

On the other hand, for any space curve in $\mathbb{R}^3$, orientation can be defined in $\mathcal{S}^3$ by Euler angles roll $\phi(s)$, pitch $\theta(s)$, and yaw $\psi(s)$. Using algebra of rotation matrices, orientation can be expressed as a product of rotation matrices so that

$$\mathbf{R}(\phi(s), \theta(s), \psi(s)) := \mathbf{R_z}(\psi(s))\mathbf{R_y}(\theta(s))\mathbf{R_x}(\phi(s)) \tag{A.14}$$

being $\mathbf{R_x}(\bullet)$, $\mathbf{R_y}(\bullet)$, and $\mathbf{R_z}(\bullet)$ three-dimensional rotation matrices around each axis of the global frame.

The first column of a rotation matrix represents the tangent vector $\mathbf{T}(s)$, computed as in Eq. (4.16), whose components are defined with respect to a global coordinate system.

Finally, the module of Eq. (A.12) can be computed easily to get $|\mathbf{C}'(s, \mathbf{p}_1, \mathbf{p}_2)| = 1$ and, as a consequence, Eq. (4.3) becomes $\mathbf{T}(s, \mathbf{p}_1, \mathbf{p}_2) = \mathbf{C}'(s, \mathbf{p}_1, \mathbf{p}_2)$. Therefore, from Eqs. (A.12) and (A.13) it is easy to proof the definitions of Eqs. (4.13) and (4.14).

A.5 Straight line

A straight line in $\mathbb{R}^3$ is defined as

$$\mathbf{C}(s,\mathbf{p}) := \begin{cases} x(s,\mathbf{p}) = as \\ y(s,\mathbf{p}) = bs \\ z(s,\mathbf{p}) = cs \end{cases} \tag{A.15}$$

where a, b, and c are constants.

For $\mathbf{p}_1 = \{\forall\psi_0, \epsilon_0 = 0, \mu = 0\}$ and $\mathbf{p}_2 = \{\forall\theta_0, \upsilon_0 = 0, \rho = 0\}$ it is straightforward to see that Eq. (4.12), expressed explicitly in Eqs. (A.10) and (A.11), becomes

$$\mathcal{C}(s,\mathbf{p}_2) = \int_0^s \cos(\theta_0)\, d\xi = \cos(\theta_0)s \tag{A.16}$$

$$-\mathcal{S}(s,\mathbf{p}_2) = -\int_0^s \sin(\theta_0)\, d\xi = -\sin(\theta_0)s \tag{A.17}$$

$$\mathcal{C}(\mathcal{C}(s,\mathbf{p}_2),\mathbf{p}_1) = \int_0^{\cos(\theta_0)s} \cos(\psi_0)\, d\xi = \cos(\psi_0)\cos(\theta_0)s \tag{A.18}$$

$$\mathcal{S}(\mathcal{C}(s,\mathbf{p}_2),\mathbf{p}_1) = \int_0^{\cos(\theta_0)s} \sin(\psi_0)\, d\xi = \sin(\psi_0)\cos(\theta_0)s \tag{A.19}$$

Thus, from Eqs. (A.15), (A.17), (A.18), and (A.19) we can state $a = \cos(\psi_0)\cos(\theta_0)$, $b = \sin(\psi_0)\cos(\theta_0)$, and $c = -\sin(\theta_0)$.

A.6 Circular arc

A circle in $\mathbb{R}^3$ can be defined as

$$\mathbf{C}(s,\mathbf{p}) := \begin{cases} x(s,\mathbf{p}) = a\cos(b+cs) \\ y(s,\mathbf{p}) = a\sin(b+cs) \\ z(s,\mathbf{p}) = d \end{cases} \tag{A.20}$$

being a the radius, b an initial angular offset, cs the turning angle traveled by the circle and d a plane parallel to $z = 0$.

For $\mathbf{p}_1 = \{\forall\psi_0, \epsilon_0 \neq 0, \mu = 0\}$ and $\mathbf{p}_2 = \mathbf{0}$ we can see that Eqs. (A.10) and (A.11) become

$$\mathcal{C}(s, \mathbf{p}_2) = \int_0^s \cos(0)\, d\xi = s \tag{A.21}$$

$$-\mathcal{S}(s, \mathbf{p}_2) = -\int_0^s \sin(0)\, d\xi = 0 \tag{A.22}$$

$$\mathcal{C}(\mathcal{C}(s, \mathbf{p}_2), \mathbf{p}_1) = \int_0^s \cos(\psi_0 + \epsilon_0\xi)\, d\xi = \frac{\sin(\psi_0 + \epsilon_0 s)}{\epsilon_0} \tag{A.23}$$

$$\mathcal{S}(\mathcal{C}(s, \mathbf{p}_2), \mathbf{p}_1) = \int_0^s \sin(\psi_0 + \epsilon_0\xi)\, d\xi = \frac{-\cos(\psi_0 + \epsilon_0 s)}{\epsilon_0} \tag{A.24}$$

From trigonometric relationships $\cos(w) = \sin(\pi/2 - w)$, $\sin(w) = \cos(\pi/2 - w)$, $\cos(-w) = \cos(w)$ and $\sin(-w) = -\sin(w)$, Eqs. (A.23) and (A.24) can be rewritten as

$$\mathcal{C}(\mathcal{C}(s, \mathbf{p}_2), \mathbf{p}_1) = \frac{\cos(\psi_0 - \pi/2 + \epsilon_0 s)}{\epsilon_0} \tag{A.25}$$

$$\mathcal{S}(\mathcal{C}(s, \mathbf{p}_2), \mathbf{p}_1) = \frac{\sin(\psi_0 - \pi/2 + \epsilon_0 s)}{\epsilon_0} \tag{A.26}$$

Then, from Eqs. (A.20), (A.22), (A.25), and (A.26) we get the following equalities $a = 1/\epsilon_0$, $b = \psi_0 - \pi/2$, $c = \epsilon_0$, and $d = 0$.

Proof for the case of $\mathbf{p}_1 = \{\forall\psi_0, \epsilon_0 = 0, \mu = 0\}$ and $\mathbf{p}_2 = \{\forall\theta_0, \upsilon_0 \neq 0, \rho = 0\}$ is omitted for brevity, although it can be easily obtained.

A.7 Planar clothoid

A planar clothoid in $\mathbb{R}^2$ is defined as in Eq. (4.9). But, for a planar clothoid contained in the XZ plane, equations can be expressed in $\mathbb{R}^3$ as follows

$$\mathbf{C}(s, \mathbf{p}) := \begin{cases} x(s, \mathbf{p}) = \mathcal{C}(s, \mathbf{p}) \\ y(s, \mathbf{p}) = 0 \\ z(s, \mathbf{p}) = -\mathcal{S}(s, \mathbf{p}) \end{cases} \tag{A.27}$$

For $\mathbf{p}_1 = \mathbf{0}$ and $\forall \mathbf{p}_2$, Eqs. (A.10) and (A.11) can be rewritten as

$$\mathcal{C}(s, \mathbf{p}_2) = \int_0^s \cos(\theta_0 + \upsilon_0 \xi + \frac{\rho}{2}\xi^2)\, d\xi \tag{A.28}$$

$$-\mathcal{S}(s, \mathbf{p}_2) = -\int_0^s \sin(\theta_0 + \upsilon_0 \xi + \frac{\rho}{2}\xi^{)}\, d\xi \tag{A.29}$$

$$\mathcal{C}(\mathcal{C}(s, \mathbf{p}_2), \mathbf{p}_1) = \int_0^{\mathcal{C}(s,\mathbf{p}_2)} \cos(0)\, d\xi = \mathcal{C}(s, \mathbf{p}_2) \tag{A.30}$$

$$\mathcal{S}(\mathcal{C}(s, \mathbf{p}_2), \mathbf{p}_1) = \int_0^{\mathcal{C}(s,\mathbf{p}_2)} \sin(0)\, d\xi = 0 \tag{A.31}$$

Therefore, it is straightforward to see that Eqs. (A.29), (A.30), and (A.31) compose a 2D clothoid contained in the XZ plane.

Proof for the case of $\mathbf{p}_2 = \mathbf{0}$ and $\forall \mathbf{p}_1$ is omitted for brevity, although it can be easily obtained following the same argumentation as in proof of circular arc's property.

A.8 Circular helix

A circular helix of radius a and slope d/a can be parametrized by arc-length s as follows

$$\mathbf{C}(s, \mathbf{p}) := \begin{cases} x(s, \mathbf{p}) = a\cos(b + cs) \\ y(s, \mathbf{p}) = a\sin(b + cs) \\ z(s, \mathbf{p}) = dcs \end{cases} \tag{A.32}$$

with b an initial angular offset, $c = 1/\sqrt{a^2 + d^2}$, and cs the turning angle traveled by the helix.

For $\mathbf{p}_1 = \{\forall\psi_0, \epsilon_0 \neq 0, \mu = 0\}$ and $\mathbf{p}_2 = \{\forall\theta_0, \upsilon_0 = 0, \rho = 0\}$ it is straightforward to see that Eqs. (A.10) and (A.11) can be expressed as

$$\mathcal{C}(s, \mathbf{p}_2) = \int_0^s \cos(\theta_0)\, d\xi = \cos(\theta_0)s \tag{A.33}$$

$$-\mathcal{S}(s, \mathbf{p}_2) = -\int_0^s \sin(\theta_0)\, d\xi = -\sin(\theta_0)s \tag{A.34}$$

$$\begin{aligned}\mathcal{C}(\mathcal{C}(s, \mathbf{p}_2), \mathbf{p}_1) = \int_0^{\cos(\theta_0)s} & \cos(\psi_0 + \epsilon_0\xi)\, d\xi = \\ & = \frac{\sin(\psi_0 + \epsilon_0 \cos(\theta_0)s)}{\epsilon_0}\end{aligned} \tag{A.35}$$

$$\begin{aligned}\mathcal{S}(\mathcal{C}(s, \mathbf{p}_2), \mathbf{p}_1) = \int_0^{\cos(\theta_0)s} & \sin(\psi_0 + \epsilon_0\xi)\, d\xi = \\ & = \frac{-\cos(\psi_0 + \epsilon_0 \cos(\theta_0)s)}{\epsilon_0}\end{aligned} \tag{A.36}$$

From trigonometric relationships, Eqs. (A.35) and (A.36) can be rewritten as

$$\mathcal{C}(\mathcal{C}(s, \mathbf{p}_2), \mathbf{p}_1) = \frac{\cos(\psi_0 - \pi/2 + \epsilon_0 \cos(\theta_0)s)}{\epsilon_0} \tag{A.37}$$

$$\mathcal{S}(\mathcal{C}(s, \mathbf{p}_2), \mathbf{p}_1) = \frac{\sin(\psi_0 - \pi/2 + \epsilon_0 \cos(\theta_0)s)}{\epsilon_0} \tag{A.38}$$

Finally, from Eqs. (A.32), (A.34), (A.37), and (A.38) we get the following equalities $a = 1/\epsilon_0$, $b = \psi_0 - \pi/2$, $c = \epsilon_0 \cos(\theta_0)$, and $d = -\sin(\theta_0)/c = -\tan(\theta_0)/\epsilon_0$.

A.9 Monotonic increasing position

Cb3D curves can be defined in all 8 octants using symmetries (see Fig. 4.3). Depending on each case the elements of position vector will be either monotonic increasing or decreasing. A monotonic decreasing function become monotonic increasing when using the absolute value. In this sense, using position vector's absolute value, all 8 cases can be reduced to only one. When $\mathbf{C}(s, \mathbf{p}_1, \mathbf{p}_2) = |\mathbf{C}(s, \mathbf{p}_1, \mathbf{p}_2)|$ the final angles are $\theta(s, \mathbf{p}_2) \in [-\pi/2, 0]$ and $\psi(s, \mathbf{p}_1, \mathbf{p}_2) \in [0, \pi/2]$, with $s > 0$. Hence, under these conditions the position vector $\mathbf{C}(s, \mathbf{p}_1, \mathbf{p}_2)$ of a Cb3D curve is monotonically increasing, iff

$$\mathbf{C}'(s, \mathbf{p}_1, \mathbf{p}_2) := \begin{cases} \cos\psi(s, \mathbf{p}_1, \mathbf{p}_2)\cos\theta(s, \mathbf{p}_2) \geq 0 \\ \sin\psi(s, \mathbf{p}_1, \mathbf{p}_2)\cos\theta(s, \mathbf{p}_2) \geq 0 \\ -\sin\theta(s, \mathbf{p}_2) \geq 0 \end{cases} \tag{A.39}$$

For equations in Eq. (A.39) to be true, the following conditions need to be true

$$\begin{array}{ll} \cos\psi(s, \mathbf{p}_1, \mathbf{p}_2) \geq 0, & \cos\theta(s, \mathbf{p}_2) \geq 0, \\ \sin\psi(s, \mathbf{p}_1, \mathbf{p}_2) \geq 0, & -\sin\theta(s, \mathbf{p}_2) \geq 0. \end{array} \tag{A.40}$$

which are true in the defined domain $\theta(s, \mathbf{p}_2) \in [-\pi/2, 0]$ and $\psi(s, \mathbf{p}_1, \mathbf{p}_2) \in [0, \pi/2]$, iff $-\pi/2 \leq \theta_0 \leq 0$, $0 \leq \psi_0 \leq \pi/2$, $\upsilon_0 \leq 0$, $\epsilon_0 \geq 0$, $\rho \leq 0$, and $\mu \geq 0$.

A.10 Monotonic increasing orientation

As in the previous property, depending on each case the angles $\theta(s, \mathbf{p}_2)$ and $\psi(s, \mathbf{p}_1, \mathbf{p}_2)$ can be either monotonically increasing or decreasing. However, all 8 cases can be reduced to only one, where $\theta(s, \mathbf{p}_2) = |\theta(s, \mathbf{p}_2)|$ and $\psi(s, \mathbf{p}_1, \mathbf{p}_2) = |\psi(s, \mathbf{p}_1, \mathbf{p}_2)|$. So, without loss of generality we only focus in such case to proof that the statement is true for $\theta(s, \mathbf{p}_2) \in [0, \pi/2]$ and $\psi(s, \mathbf{p}_1, \mathbf{p}_2) \in [0, \pi]$, with $s > 0$.

For angles yaw and pitch to be monotonic increasing, the following conditions in the derivatives of Eqs. (4.13) and (4.14) must be satisfied

$$\theta'(s, \mathbf{p}_2) = \upsilon_0 + \rho s \geq 0 \tag{A.41}$$

$$\psi'(s, \mathbf{p}_1, \mathbf{p}_2) = \epsilon_0 + \mu\mathcal{C}(s, \mathbf{p}_2)\mathcal{C}'(s, \mathbf{p}_2) \geq 0 \tag{A.42}$$

For Eqs. (A.41) and (A.42) to be true, the following conditions need to be true

$$\begin{array}{llll} \upsilon_0 \geq 0, & \rho \geq 0, & & \\ \epsilon_0 \geq 0, & \mu \geq 0, & \mathcal{C}(s, \mathbf{p}_2) \geq 0, & \mathcal{C}'(s, \mathbf{p}_2) \geq 0. \end{array} \tag{A.43}$$

which are all true in the defined range for $s > 0$, iff $\mathcal{C}'(s, \mathbf{p}_2) = \cos(\theta_0 + \upsilon_0 s + \frac{\rho}{2}s^2) \geq 0$, that is only true if $0 \leq \theta_0 \leq \pi/2$ since $\theta(s, \mathbf{p}_2) \in [0, \pi/2]$. Also considering that $\psi(s, \mathbf{p}_1, \mathbf{p}_2) \in [0, \pi]$, the condition $0 \leq \psi_0 \leq \pi$ must be also required.

Apéndice B

Plataforma de test: Kadett 2400

Un Vehículo Aéreo no Tripulado UAV, es un Vehículo Móvil que se desenvuelve en el espacio aéreo (a través del vuelo sostenido), el cual posee diversas habilidades que le otorgan las destrezas necesarias para realizar tareas específicas sin la intervención de un operador humano abordo (el operador puede interactuar a distancia).

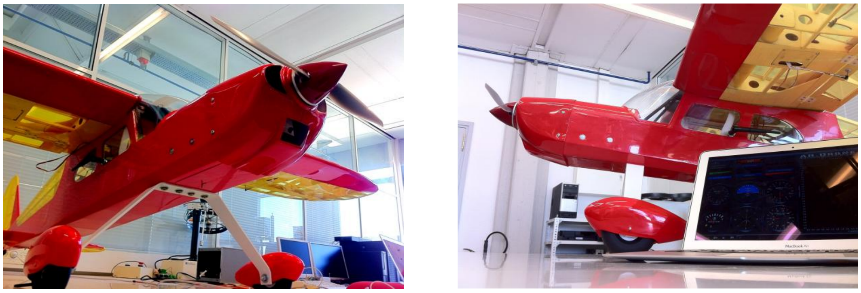
Nota: Los Vehículos balísticos o semibalísticos, misiles de crucero, proyectiles de artillería, torpedos, minas, satélites, y sensores desatendidos (sin forma de propulsión) no se consideran vehículos no tripulados [Clapper y col. 2007].

Con el objetivo de realizar un acercamiento hacia el UAV utilizado en el desarrollo de la experimentación del trabajo de tesis, a continuación, se describe el hardware y software referente al modelo UAV *Kadett 2400*, así como de la formalización matemática de su cinemática y dinámica.

Una vez más, es importante resaltar que el trabajo de tesis ha sido enfocado en un desarrollo de planificación de vuelo suave desde una perspectiva cinemática, no obstante, y debido al uso de las bondades brindadas por el UAV *Kadett 2400*, se ha considerado relevante realizar una breve descripción de la dinámica del modelo UAV.

B.1 Graupner Kadett 2400 RC aircraft

El modelo UAV *Graupner Kadett 2400 RC aircraft* (ver figura B.1) ha sido construido de forma convencional en su totalidad de madera, posee una envergadura aproximada de 2400 mm, longitud total aproximada de 1745 mm, superficie alar aproximada de 100 dm^2, un peso total según equipamiento 4900 g y un total de $1{,}65 \times 10^{-2}$ m^3 de volumen disponible para el hardware de control.



(**a**)Graupner Kadett 2400 RC aircraft (vista frontal).

(**b**)Graupner Kadett 2400 RC aircraft (vista lateral).

Figura B.1: Modelo UAV Graupner Kadett 2400 RC aircraft. Fuente: Velasco Carrau y col. 2012.

Equipado con todas las superficies de control necesarias para su desenvolvimiento (timón, rueda de cola acoplada, elevador, alerones, flaps de aterrizaje, acelerador). De esta forma, el fuselaje, alas y paneles de cola están recubiertos de plástico, el timón se maneja a través de un cable de tracción, mientras que elevador, alerón y flaps de aterrizaje son accionados por servos instalados cerca de las superficies de control. Sus diversos elementos hacen que el modelo esté Casi Listo para Volar (ARFT) y lo convierte en un dispositivo ideal para el vuelo, además que su velocidad de aterrizaje es extremadamente baja.

Este UAV *Kadett 2400* lleva embebidos todos los dispositivos necesarios para ejecutar vuelo continuo, mismo que puede ser controlado, tanto de forma manual, como automática. La propulsión es proporcionada por un motor de corriente alterna sin escobillas, el variador y los servomotores están controlados por señales de comando moduladas por ancho de pulso de (PWM). Entonces, con el objetivo de describir las conexiones internas del UAV *Kadett 2400*, en la Figura B.2 se muestran los diferentes dispositivos y el Sistema de Control de vuelo.

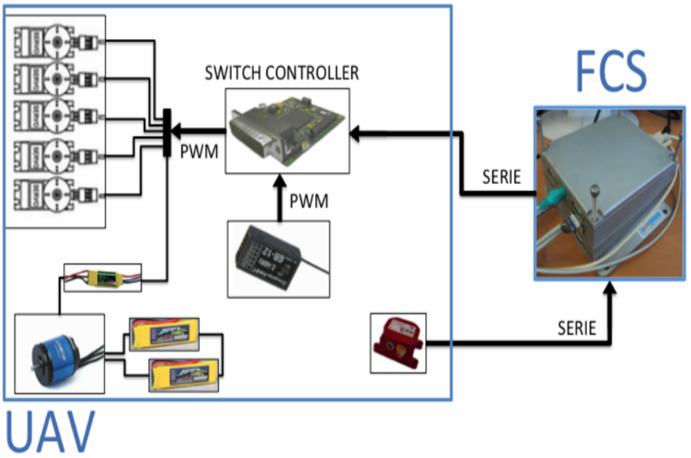


Figura B.2: Sistema de Control Completo del UAV *Kadett 2400*. Fuente: Velasco y García-Nieto 2014.

De forma independiente de las características detalladas sobre el modelo *Kadett 2400*, a continuación, se realiza una breve descripción del modelo cinemático y dinámico de una aeronave de 3 ejes y 6 Grados de Libertad (6 Dof).

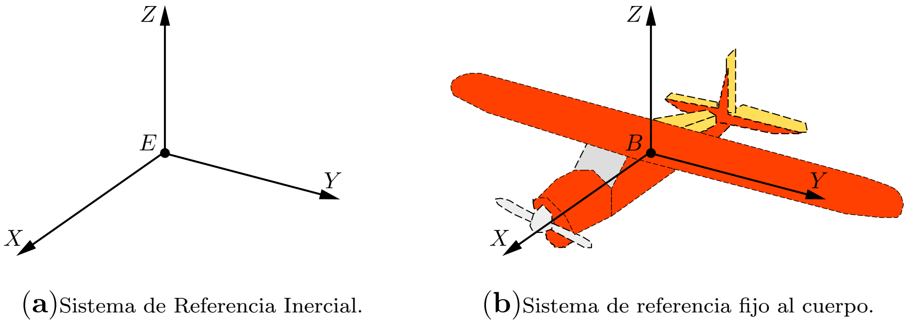
B.1.1 Modelo cinemático de la aeronave

En términos generales, la cinemática describe aspectos geométricos del movimiento sin considerar las fuerzas que las ocasionan. Entonces, con el objetivo de realizar un análisis de movimiento del UAV es necesario definir los diferentes sistemas de referencia, siendo:

Sistema de referencia inercial. Definido a partir de las leyes del movimiento, las cuales cumplen con las leyes de Newton. La Figura B.3(a) muestra un ejemplo, donde E representa el sistema de referencia inercial.

Sistema de referencia fijo al cuerpo. Sistema de referencia fijado al cuerpo del UAV, como se puede ver en la Figura B.3(b). La posición y orientación del UAV son medidas respecto al sistema de referencia inercial, representado por B.

Entonces, un modelo UAV de tipo aeronave de ala fija posee 6 DoF, traducidos como desplazamientos en x, desplazamientos en y, desplazamientos en z, es decir, desplazamientos sobre los ángulos de alabeo ϕ, cabeceo θ y guiñada ψ.



(a) Sistema de Referencia Inercial. (b) Sistema de referencia fijo al cuerpo.

Figura B.3: Cinemática del sistema UAV, sistemas de referencia.

En consecuencia, los movimientos del UAV se dividen en rotación y traslación, movimientos definidos en los 3 ejes fijos (X, Y, Z), y a través de la oscilación de los ángulos de navegación se determinan los ángulos de guiñada (eje Z), cabeceo (eje Y) y alabeo (eje X). Por tanto, los movimientos se traducen en 3 movimientos de traslación (x, y, z) y 3 movimientos de rotación (ϕ, θ, ψ).

Rotación Angular en alabeo. La rotación sobre el eje x, se obtiene al modificar el valor del ángulo de alabeo ϕ, lo cual produce movimientos hacia la derecha e izquierda.

Rotación Angular en cabeceo. La rotación sobre el eje y, se obtiene al modificar el ángulo de cabeceo θ, lo cual produce movimientos hacia arriba y abajo.

Rotación Angular en guiñada. La rotación sobre el eje z. El movimiento en el ángulo de guiñada ψ produce un desbalance aerodinámico que produce que el UAV realice un giro sobre su propio eje.

Sistema de Referencia. Con el objetivo de definir las ecuaciones de la dinámica del sistema UAV, es necesario expresar diversas fuerzas y momentos en el sistema de referencia inercial. En este sentido, si se asume que el UAV se desplaza en el transcurso del tiempo, entonces, el UAV cambia de posición respecto a un punto de referencia que se supone fijo. Por lo tanto, la posición del UAV se determina por medio de 2 puntos, uno fijo en tierra $E = [x_E, y_E, z_E]$, y el otro en el centro de masa del UAV $B = [x_B, y_B, z_B]$. La posición y traslación del UAV se describen mediante el movimiento de los ángulos (ϕ, θ, ψ), como se describe en la Figura B.4.

En el sistema de referencia B se emplean las ecuaciones cinemáticas de rotación y translación. Dicho punto está en el centro del UAV, sobre el que se refieren los giros y movimientos.

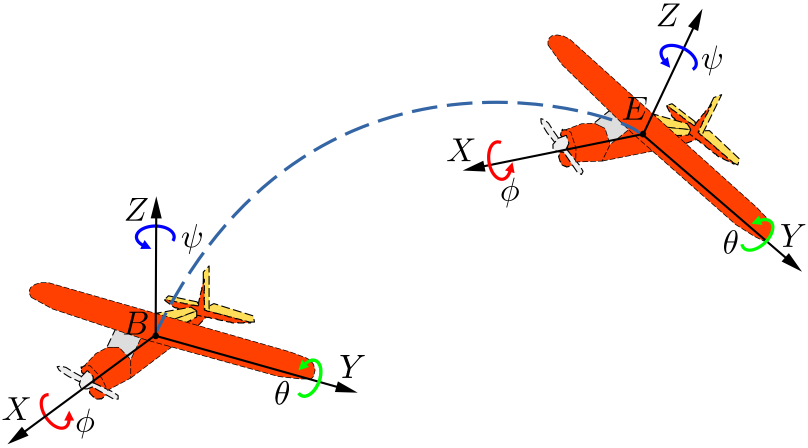


Figura B.4: Sistema de Referencia UAV.

En consecuencia, la posición del UAV en el punto E está definida por el vector $p = [x, y, z]^T$, mientras que la orientación esta dada por el vector $o = [\phi, \theta, \psi]$. Es así que, con base a estos ángulos se formula la orientación del UAV con respecto al sistema de referencia E.

Por lo tanto, para la definición de matrices de rotación se utiliza la convención $[z_\psi][y_\theta][x_\phi]$ [Markley y col. 2014], de movimiento alrededor de los ejes.

La matriz de rotación en alabeo se define por

$$R(x,\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\phi) & -\sin(\phi) \\ 0 & \sin(\phi) & \cos(\phi) \end{bmatrix} \tag{B.1}$$

La matriz de rotación en cabeceo se define por

$$R(y,\theta) = \begin{bmatrix} \cos(\theta) & 0 & \sin(\theta) \\ 0 & 1 & 0 \\ -\sin(\theta) & 0 & \cos(\theta) \end{bmatrix} \tag{B.2}$$

La matriz de rotación en guiñada se define por

$$R(z,\psi) = \begin{bmatrix} \cos(\psi) & -\sin(\psi) & 0 \\ \sin(\psi) & \cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{B.3}$$

Siendo la matriz de rotación completa definida por:

$$\begin{gathered} R(\phi,\theta,\psi) = R(x,\phi)\cdot R(y,\theta)\cdot R(z,\psi) = \\ \begin{bmatrix} c\theta c\psi & c\psi s\theta c\phi - s\psi c\phi & c\psi s\theta c\phi + s\psi s\phi \\ s\psi c\theta & s\psi s\theta s\phi + c\psi c\phi & s\psi s\theta c\phi - s\phi c\psi \\ -s\theta & c\theta s\phi & c\theta c\phi \end{bmatrix} \end{gathered} \tag{B.4}$$

donde $c = \cos$ y $s = \sin$, entonces, la ecuación B.4 expresa el movimiento de traslación del UAV.

B.1.2 Modelo dinámico de la aeronave

En general, las ecuaciones para el modelo dinámico de una aeronave se definen, a través de las ecuaciones de fuerza, tal que

$$\begin{aligned} \dot{u} &= rv - qw + \frac{\overline{q}S}{m}C_X(\delta[e,a,r]) - g\sin\theta + \frac{T}{m} \\ \dot{v} &= pw - ru + \frac{\overline{q}S}{m}C_Y(\delta[e,a,r]) + g\cos\theta\sin\phi \\ \dot{w} &= qu - pv + \frac{\overline{q}S}{m}C_Z(\delta[e,a,r]) + g\cos\theta\cos\phi \end{aligned} \tag{B.5}$$

donde g representa la intensidad del campo gravitacional cerca de la superficie de la Tierra, mientras que m representa la masa total del sistema [Chang y col. 2018].

Las ecuaciones de par de fuerzas se definen por:

$$\begin{aligned}
\dot{p} - \frac{I_{xz}}{I_x}\dot{r} &= \frac{\overline{q}Sb}{I_x}C_l(\delta[e,a,r]) - \frac{I_z - I_y}{I_x}qr + \frac{I_{xz}}{I_x}qp \\
\dot{q} &= \frac{\overline{q}S\overline{c}}{I_y}C_m(\delta[e,a,r]) - \frac{I_x - I_z}{I_y}pr - \frac{I_{xz}}{I_y}(p^2 - r^2) + I_p\Omega_p q \\
\dot{r} - \frac{I_{xz}}{I_z}\dot{p} &= \frac{\overline{q}Sb}{I_z}C_n(\delta[e,a,r]) - \frac{I_y - I_x}{I_z}pq + \frac{I_{xz}}{I_z}qr - I_p\Omega_p q
\end{aligned} \tag{B.6}$$

Las ecuaciones cinemáticas se definen como:

$$\begin{aligned}
\dot{\phi} &= p + \tan\theta(q\sin\phi + r\cos\phi) \\
\dot{\theta} &= q\cos\phi - r\sin\phi \\
\dot{\psi} &= \frac{q\sin\phi + r\cos\phi}{\cos\theta}
\end{aligned} \tag{B.7}$$

En este sentido, en la Figura B.5, se muestra el cuerpo de referencia $(X_bY_bZ_b)$, mientras que (u,v,w) representan las componentes de la velocidad de traslación. Las variables (p,q,r) representan las componentes de velocidad angular. Los momentos de inercia se describen a través de $(I_xI_yI_z)$. Donde, I_{xz} es el producto de inercia. Los productos de inercia I_{xy} y I_{yz} relacionados al plano longitudinal $(Y_b = 0)$ son nulos debido a la simetría del UAV con respecto al plano. I_p es la rotación inercial del motor y la hélice, Ω_p es la velocidad de rotación, y T es el empuje del motor. S, b y $\overline{c}$ son la superficie aerodinámica del modelo *Kadett 2400*, la envergadura y la cuerda alar respectivamente, y $\overline{q}$ es la presión dinámica[1]. Los coeficiente aerodinámicos (AC) C_X, C_Y, C_Z, C_l, C_m y C_n son variables del sistema. El símbolo δ entre paréntesis representa la dependencia de las deflexiones de las superficies de control (δ_e, δ_a y δ_r son las deflexiones de los elevadores, alerones y timón). Finalmente la orientación del UAV se representa por los ángulos de Euler alabeo θ, cabeceo ϕ y guiñada ψ.

A partir de un entorno de vuelo estable en el cual no se realizan maniobras abruptas [Klein y Morelli 2006], entonces, el sistema UAV se define por medio de un conjunto de las ecuaciones de primer y segundo orden. Por lo tanto, se asume que las variables simétricas (longitudinales) u, w y q, no afectan a las fuerzas y pares asimétricos (laterales) Y, L y N. Por otra parte, las variables

[1] Es una función de la la densidad del aire y la velocidad del aire en relación con el viento local

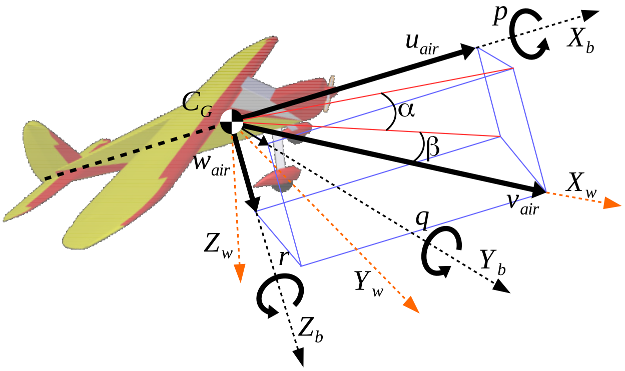


Figura B.5: Ejes del modelo UAV *Kadett 2400* (ejes de viento, y de la velocidad relativa del viento). Fuente:[Velasco y García-Nieto 2014]

asimétricas (laterales) v, p y r, no afectan a las fuerzas y pares simétricos (longitudinales) X, Z y M. Siendo las ecuaciones de los modelos aerodinámicos representadas como:

Modelos aerodinámicos longitudinales

$$\begin{aligned} C_D(t) = \mathbf{C_{D_0}} + \mathbf{C_{D_V}} \frac{1}{V_0} \Delta V(t) + \mathbf{C_{D_\alpha}} \Delta\alpha(t) \\ + \mathbf{C_{D_{\alpha^2}}} \Delta\alpha(t)^2 + \mathbf{C_{D_q}} \frac{\bar{c}}{2V_0} q(t) + \mathbf{C_{D_{\delta_d}}} \Delta\delta_e(t) \end{aligned} \tag{B.8}$$

$$\begin{aligned} C_L(t) = \mathbf{C_{L_0}} + \mathbf{C_{L_V}} \frac{1}{V_0} \Delta V(t) + \mathbf{C_{L_\alpha}} \Delta\alpha(t) \\ + \mathbf{C_{L_{\alpha^2}}} \Delta\alpha(t)^2 + \mathbf{C_{L_{\dot\alpha}}} \frac{\bar{c}}{2V_0} \dot\alpha(t) \\ + \mathbf{C_{L_q}} \frac{\bar{c}}{2V_0} q(t) + \mathbf{C_{L_{\delta_e}}} \Delta\delta_e(t) \end{aligned} \tag{B.9}$$

$$\begin{aligned} C_m(t) = \mathbf{C_{m_0}} + \mathbf{C_{m_V}} \frac{1}{V_0} \Delta V(t) + \mathbf{C_{m_\alpha}} \Delta\alpha(t) \\ + \mathbf{C_{m_{\alpha^2}}} \Delta\alpha(t)^2 + \mathbf{C_{m_{\dot\alpha}}} \frac{\overline{c}}{2V_0} \dot\alpha(t) \\ + \mathbf{C_{m_q}} \frac{\overline{c}}{2V_0} q(t) + \mathbf{C_{q_{\delta_e}}} \Delta\delta_e(t) \end{aligned} \tag{B.10}$$

Modelos aerodinámicos laterales

$$\begin{aligned} C_Y(t) = \mathbf{C_{Y_0}} + \mathbf{C_{Y_\beta}} \Delta\beta(t) + \mathbf{C_{Y_p}} \frac{b}{2V_0} p(t) \\ + \mathbf{C_{Y_r}} \frac{b}{2V_0} r(t) + \mathbf{C_{Y_{\alpha_{al}}}} \Delta\delta_{al}(t) + \mathbf{C_{Y_{\delta_r}}} \Delta\delta_r(t) \end{aligned} \tag{B.11}$$

$$\begin{aligned} C_l(t) = \mathbf{C_{l_0}} + \mathbf{C_{l_\beta}} \Delta\beta(t) + \mathbf{C_{l_p}} \frac{b}{2V_0} p(t) \\ + \mathbf{C_{l_r}} \frac{b}{2V_0} r(t) + \mathbf{C_{l_{\alpha_{al}}}} \Delta\delta_{al}(t) + \mathbf{C_{l_{\delta_r}}} \Delta\delta_r(t) \end{aligned} \tag{B.12}$$

$$\begin{aligned} C_n(t) = \mathbf{C_{n_0}} + \mathbf{C_{n_\beta}} \Delta\beta(t) + \mathbf{C_{n_p}} \frac{b}{2V_0} p(t) \\ + \mathbf{C_{n_r}} \frac{b}{2V_0} r(t) + \mathbf{C_{n_{\alpha_{al}}}} \Delta\delta_{al}(t) + \mathbf{C_{n_{\delta_r}}} \Delta\delta_r(t) \end{aligned} \tag{B.13}$$

donde α y β son el ángulo de ataque y de desplazamiento lateral, respectivamente. Mientras que V es el velocidad del aire. En particular, V_0 es la velocidad del aire medida en estado de vuelo estacionario. Estas variables dependen de la velocidad y se definen como:

$$\alpha = \arctan\left(\frac{w}{u}\right) \; ; \; \beta = \arcsin\left(\frac{v}{V}\right) \tag{B.14}$$

$$V = |\overrightarrow{V}| = \sqrt{u^2 + v^2 + w^2} \tag{B.15}$$

Finalmente, C_L y C_D describen los coeficientes de elevación y arrastre, en relación con C_X y C_Z, definidos como:

$$C_L(t) = -C_Z(t)\cos(\alpha(t)) + C_X(t)\sin(\alpha(t)) \tag{B.16}$$
$$C_D(t) = -C_X(t)\cos(\alpha(t)) + C_Z(t)\sin(\alpha(t)) \tag{B.17}$$

Por lo tanto, el modelo dinámico se ha definido desde B.8 hasta B.13. Lo cual da lugar al conjunto de constantes no dimensionales de estabilidad y control.

Bibliografía

Adiprawita, Widyawardana, Adang Suwandi Ahmad y Jaka Sembiring (2012). "Development of ahrs (attitude and heading reference system) for autonomous uav (unmanned aerial vehicle)". En: (vid. págs. 55, 66).

Afman, Juan-Pablo y col. (2018). "Towards a new paradigm of UAV safety". En: *arXiv preprint arXiv:1803.09026* (vid. pág. 92).

Afrianto, Fahrizal y col. (2018). *Studi Analisis Mekanika FIsika dalam Lintasan Loop the Loop Berbentuk Clothoid pada Roller Coaster* (vid. pág. 30).

Aguirre, Eduardo (2007). "Geometria diferencial de curvas y superficies". En: *Universidad Complutense de Madrid, España. Obtenido de http://www. mat. ucm. es/˜ edaguirr/cys06. pdf* (vid. págs. 12, 13, 21).

Alexander, Daniel C y Geralyn M Koeberlein (2014). *Elementary geometry for college students*. Cengage Learning (vid. pág. 12).

Alia, Chebly y col. (2015). "Local trajectory planning and tracking of autonomous vehicles, using clothoid tentacles method". En: *Institute of Electrical and Electronics Engineers (IEEE)*, págs. 674-679. DOI: 10.1109/IVS.2015.7225762 (vid. pág. 93).

amd Alonzo Kelly, Bryan Nagy (2001). "Trajectory Generation for Car-Like Robots Using Cubic Curvature Polynomials". En: *Field and Service Robots* (vid. pág. 67).

Anderson, Brian DO y col. (2008). "UAV formation control: Theory and application". En: *Recent advances in learning and control.* Springer, págs. 15-33 (vid. pág. 2).

Armesto, L y col. (2012). *Mobile robot obstacle avoidance based on quasi-holonomic smooth paths.* DOI: 10.1007/978-3-642-32527-4_22 (vid. pág. 67).

Armesto, Leopoldo y col. (2015). "Duality-based nonlinear quadratic control: Application to mobile robot trajectory-following". En: *IEEE Transactions on Control Systems Technology* 23.4, págs. 1494-1504. ISSN: 10636536. DOI: 10.1109/TCST.2014.2377631 (vid. pág. 121).

Arshad, Saba y col. (2020). "Clothoid: An Integrated Hierarchical Framework for Autonomous Driving in a Dynamic Urban Environment". En: *Sensors* 20.18. ISSN: 1424-8220. DOI: 10.3390/s20185053 (vid. pág. 93).

Asim, Muhammad, DN Ehsan y Khalid Rafique (2005). "Probable causal factors in uav accidents based on human factor analysis and classification system". En: *history* 1905, pág. 5 (vid. pág. 2).

Ataei, Mansour y Aghil Yousefi-Koma (2015). "Three-dimensional optimal path planning for waypoint guidance of an autonomous underwater vehicle". En: *Robotics and Autonomous Systems* 67, págs. 23-32. ISSN: 09218890. DOI: 10.1016/j.robot.2014.10.007 (vid. pág. 67).

BAASS, KG (1984). "The use of clothoid templates in highway design". En: *Transportation Forum.* Vol. 1, págs. 47-52 (vid. pág. 30).

Baek, Hoki y Jaesung Lim (2018). "Design of Future UAV-Relay Tactical Data Link for Reliable UAV Control and Situational Awareness". En: *IEEE Communications Magazine* 56.10, págs. 144-150. DOI: 10.1109/MCOM.2018.1700259 (vid. pág. 92).

Banchoff, Thomas. y Stephen (Stephen T.) Lovett (2010). *Differential geometry of curves and surfaces.* A K Peters/CRC Press, pág. 352. ISBN: 99781439894057. DOI: `10.1201/9781439894057` (vid. págs. 52, 67-69).

Bertails-Descoubes, Florence (2012). "Super-clothoids". En: *Computer Graphics Forum* 31.2, págs. 509-518. ISSN: 14678659. DOI: `10.1111/j.1467-8659.2012.03030.x` (vid. pág. 67).

Bhandari, Subodh y col. (2016). "Nonlinear Control of a Fixed-Wing UAV using Support Vector Machine". En: *AIAA Guidance, Navigation, and Control Conference.* Reston, Virginia: American Institute of Aeronautics y Astronautics. ISBN: 978-1-62410-389-6. DOI: `10.2514/6.2016-0107` (vid. pág. 66).

Blasi, Luciano y col. (2023). "Clothoid-Based Path Planning for a Formation of Fixed-Wing UAVs". En: *Electronics* 12.10, pág. 2204 (vid. pág. 3).

Boissonnat, Jean-Daniel, André Cerezo y Juliette Leblond (1994). "A note on shortest paths in the plane subject to a constraint on the derivative of the curvature". En: (vid. pág. 3).

Bonnefon, Jean-François, Azim Shariff e Iyad Rahwan (2016). "The social dilemma of autonomous vehicles". En: *Science* 352.6293, págs. 1573-1576 (vid. pág. 2).

Brezak, M e I Petrovic (2014). "Real-time Approximation of Clothoids With Bounded Error for Path Planning Applications". En: *Robotics, IEEE Transactions on* 30.2, págs. 507-515. ISSN: 1552-3098. DOI: `10.1109/TRO.2013.2283928` (vid. págs. 51, 67, 70).

C. Goerzen, Z Kong y col. (2010). "A Survey of Motion Planning Algorithms from the Perspective of Autonomous UAV Guidance". En: *Journal of Intelligent Robots and Systems* 57.1-4, págs. 65-100. ISSN: 09210296. DOI: `10.1007/s10846-009-9383-1` (vid. págs. 50, 66).

Cabreira, Tauã M., Lisane B. Brisolara y R. Ferreira Paulo (2019). "Survey on coverage path planning with unmanned aerial vehicles". En: *Drones* 3.1, págs. 1-38. ISSN: 2504446X. DOI: `10.3390/drones3010004` (vid. pág. 120).

Cai, Wenyu, Meiyan Zhang y Yahong Rosa Zheng (2017). “Task assignment and path planning for multiple autonomous underwater vehicles using 3D dubins curves”. En: *Sensors* 17.7, pág. 1607 (vid. págs. 2, 45).

Carrio, Adrian y col. (2017). “A review of deep learning methods and applications for unmanned aerial vehicles”. En: *Journal of Sensors* 2017 (vid. pág. 2).

Carsten, Joseph, Dave Ferguson y Anthony Stentz (2006). “3d field d: Improved path planning and replanning in three dimensions”. En: *2006 IEEE/RSJ international conference on intelligent robots and systems*. IEEE, págs. 3381 -3386 (vid. pág. 3).

Çakici, Ferit y M. Kemal Leblebicioğlu (2016). “Control System Design of a Vertical Take-off and Landing Fixed-Wing UAV”. En: *IFAC-PapersOnLine* 49.3. 14th IFAC Symposium on Control in Transportation Systems 2016, págs. 267-272. ISSN: 2405-8963. DOI: 10.1016/j.ifacol.2016.07.045 (vid. pág. 92).

Çelik, Semra S.; Yayli, Yusuf; Güler, Erhan (2016). “On Generalized Euler Spirals in E3”. En: *Differential Geometry* 5.1, págs. 5-14. arXiv: 1201.6528 (vid. pág. 67).

Celik, Semra S., Yusuf Yayli y Erhan Guler (2016). “ON GENERALIZED EULER SPIRALS IN E^3.” En: *International Journal of Geometry* 5.1, págs. 5-14 (vid. pág. 93).

Chang, Hao y col. (2018). “Design of Tracked Model Vehicle Measurement and Control System Based on VeriStand and Simulink”. En: *MATEC Web of Conferences*. Vol. 175. EDP Sciences, pág. 03047 (vid. pág. 160).

Chaurasia, Rohit y Vandana Mohindru (2021). “Unmanned Aerial Vehicle (UAV): A Comprehensive Survey”. En: *Unmanned Aerial Vehicles for Internet of Things (IoT) Concepts, Techniques, and Applications*, págs. 1-27 (vid. pág. 120).

Chen, Yong y col. (2017a). “Accurate and Efficient Approximation of Clothoids Using Bézier Curves for Path Planning”. En: *IEEE Transactions on Robotics* 33.5, págs. 1242-1247. ISSN: 15523098. DOI: 10.1109/TRO.2017.2699670 (vid. págs. 51, 70).

— (2017b). "Accurate and Efficient Approximation of Clothoids Using Bézier Curves for Path Planning". En: *IEEE Transactions on Robotics* 33.5, págs. 1242-1247. ISSN: 15523098. DOI: `10.1109/TRO.2017.2699670` (vid. pág. 122).

Choe, Ronald y col. (2016). "Cooperative Trajectory Generation Using Pythagorean Hodograph Bézier Curves". En: *Journal of Guidance, Control, and Dynamics* 39.8, págs. 1744-1763. ISSN: 0731-5090. DOI: `10.2514/1.G001531` (vid. pág. 66).

Choi, Ji-wung, Renwick Curry y Gabriel Elkaim (2008). "Path planning based on bézier curve for autonomous ground vehicles". En: *Advances in Electrical and Electronics Engineering-IAENG Special Edition of the World Congress on Engineering and Computer Science 2008*. IEEE, págs. 158-166 (vid. págs. 12, 41).

Claesson, A y col. (2017). "Drones may be used to save lives in out of hospital cardiac arrest due to drowning". En: *Resuscitation* 114, págs. 152-156 (vid. pág. 2).

Clapper, J y col. (2007). "Unmanned systems roadmap 2007-2032". En: *Office of the Secretary of Defense* 188 (vid. pág. 155).

Cordero, Luis A, M Fernández y A Gray (1995). *Geometria diferencial de curvas y superficies*. Addison-Wesley Iberoamericana (vid. pág. 12).

Dai, Jian S. (2015). "Euler-Rodrigues formula variations, quaternion conjugation and intrinsic connections". En: *Mechanism and Machine Theory* 92, págs. 144-152. ISSN: 0094-114X. DOI: `10.1016/j.mechmachtheory.2015.03.004` (vid. pág. 96).

Davoodi, Mansoor y col. (2015). "Clear and smooth path planning". En: *Applied Soft Computing Journal* 32, págs. 568-579. ISSN: 15684946. DOI: `10.1016/j.asoc.2015.04.017` (vid. pág. 121).

Delingette, H, M Hebert y K Lkeuchi (1991). "Trajectory Generation with Curvature Constraint based on Energy Minimization". En: *IEEE Int. Conf. on Intelligent Robots and Systems* (vid. pág. 67).

Dilectis, Francesco de, Daniele Mortari y Renato Zanetti (2016). “Bézier Description of Space Trajectories”. En: *Journal of Guidance, Control, and Dynamics* 39.11, págs. 2535-2539. ISSN: 0731-5090. DOI: 10.2514/1.G000719 (vid. pág. 66).

Do Carmo, Manfredo P (2016). *Differential geometry of curves and surfaces: revised and updated second edition.* Courier Dover Publications (vid. pág. 123).

Elbanhawi, Mohamed, Milan Simic y Reza N Jazar (2015). “Continuous path smoothing for car-like robots using B-spline curves”. En: *Journal of Intelligent & Robotic Systems* 80.1, págs. 23-56 (vid. pág. 40).

Eren, Utku y col. (2017). “Model Predictive Control in Aerospace Systems: Current State and Opportunities”. En: *Journal of Guidance, Control, and Dynamics* 40.7, págs. 1541-1566. ISSN: 0731-5090. DOI: 10.2514/1.G002507 (vid. pág. 66).

Finger, D Felix, Carsten Braun y Cees Bil (2017). “A review of configuration design for distributed propulsion transitioning VTOL aircraft”. En: *Asia-Pacific International Symposium on Aerospace Technology-APISAT*, págs. 3-5 (vid. pág. 92).

Floreano, Dario y col. (2017). “Aerial Locomotion in Cluttered Environments”. En: *Robotics Research : The 15th International Symposium ISRR*. Ed. por Henrik I. Christensen y Oussama Khatib. Cham: Springer International Publishing, págs. 21-39. ISBN: 978-3-319-29363-9. DOI: 10.1007/978-3-319-29363-9_2 (vid. pág. 92).

Fraichard, Th. y A Scheuer (2004a). “From Reeds and Shepp’s to continuous-curvature paths”. En: *Robotics and Automation, IEEE Transactions on.* 6, 1025-1035 vol. 20. DOI: 10.1109/TRO.2004.833789 (vid. págs. 67, 93).

Fraichard, Thierry y J-M Ahuactzin (2001). “Smooth path planning for cars”. En: *Proceedings 2001 ICRA. IEEE International Conference on Robotics and Automation (Cat. No. 01CH37164).* Vol. 4. IEEE, págs. 3722-3727 (vid. pág. 42).

Fraichard, Thierry y Alexis Scheuer (2004b). “From Reeds and Shepp’s to Continuous-Curvature Paths”. En: *IEEE Transactions on Robotics* 20.6, págs. 1025-1035. ISSN: 1552-3098. DOI: `10.1109/TRO.2004.833789` (vid. pág. 3).

Frego, Marco (2022). "Closed form parametrisation of 3D clothoids by arclength with both linear varying curvature and torsion". En: *Applied Mathematics and Computation* 421, pág. 126907 (vid. pág. 3).

Fu, Zhangjie y col. (sep. de 2018). "A Heuristic Evolutionary Algorithm of UAV Path Planning". En: *Wireless Communications and Mobile Computing* 2018, págs. 1-11. DOI: `10.1155/2018/2851964` (vid. pág. 93).

Gafurov, Salimzhan A y Evgeniy V Klochkov (2015). "Autonomous unmanned underwater vehicles development tendencies". En: *Procedia Engineering* 106, págs. 141-148 (vid. pág. 2).

Giese, Stefanie, David Carr y Javaan Chahl (2013). "Implications for unmanned systems research of military UAV mishap statistics". En: *2013 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, págs. 1191-1196 (vid. pág. 2).

Gim, Suhyeon y col. (2017a). "Clothoids Composition Method for Smooth Path Generation of Car-Like Vehicle Navigation". En: *Journal of Intelligent and Robotic Systems: Theory and Applications* 88.1, págs. 129-146. ISSN: 15730409. DOI: `10.1007/s10846-017-0531-8` (vid. págs. 51, 70).

Gim, Suhyeon y col. (2017b). "Clothoids composition method for smooth path generation of car-like vehicle navigation". En: *Journal of Intelligent & Robotic Systems* 88.1, págs. 129-146. DOI: `10.1007/s10846-017-0531-8` (vid. págs. 93, 122).

Girbes, V, L Armesto y J Tornero (2011). "On Generating Continuous-Curvature Paths for Line Following Problem with Curvature and Sharpness Constraints". En: *IEEE International Conference on Robotics and Automation*, págs. 6156-6161 (vid. pág. 55).

Girbés, Vicent, Leopoldo Armesto y Josep Tornero (2014). "Path following hybrid control for vehicle stability applied to industrial forklifts". En: *Robotics and Autonomous Systems* 62.6, págs. 910-922 (vid. págs. 3, 43, 93, 125, 128).

Girbés, Vicent, Leopoldo Armesto y Josep Tornero (2014). “Path Following Hybrid Control for Vehicle Stability Applied to Industrial Forklifts”. En: *Robotics and Autonomous Systems* 62.6, págs. 910-922 (vid. págs. 51, 67, 70).

Girbés, Vicent, Gloria Vanegas y Leopoldo Armesto (2019). “Clothoid-Based Three-Dimensional Curve for Attitude Planning”. En: *Journal of Guidance, Control, and Dynamics* 42.8, págs. 1886-1898. DOI: `10.2514/1.G003551` (vid. págs. 93-95, 98, 99, 106, 115, 122, 127-129).

Gómez, Carlos Munuera y Fernando Torres (2006). “Sobre curvas algebraicas y códigos correctores”. En: *Gaceta de la Real Sociedad Matematica Española* 9.1, págs. 203-222 (vid. pág. 12).

González, David y col. (2014). “Continuous curvature planning with obstacle avoidance capabilities in urban scenarios”. En: *17th International IEEE Conference on Intelligent Transportation Systems (ITSC).* IEEE, págs. 1430 -1435 (vid. págs. 13, 41, 42, 121).

Gonzalez, David y col. (2016). “A Review of Motion Planning Techniques for Automated Vehicles”. En: *IEEE Transactions on Intelligent Transportation Systems* 17.4, págs. 1135-1145. ISSN: 1524-9050. DOI: `10.1109/TITS.2015.2498841` (vid. pág. 67).

Gordon, William J y Richard F Riesenfeld (1974a). “B-spline curves and surfaces”. En: *Computer aided geometric design.* Elsevier, págs. 95-126 (vid. pág. 29).

— (1974b). “Bernstein-Bézier methods for the computer-aided design of free-form curves and surfaces”. En: *Journal of the ACM (JACM)* 21.2, págs. 293 -310 (vid. págs. 28, 54).

Gray, Alfred, Elsa Abbena y Simon Salamon (2017). *Modern differential geometry of curves and surfaces with Mathematica®*. Chapman y Hall/CRC (vid. pág. 24).

Guclu, Anil, Dilek F Kurtulus y Kutluk B Arikan (2016). “Attitude and altitude stabilization of fixed wing VTOL unmanned air vehicle”. En: *AIAA Modeling and Simulation Technologies Conference*, pág. 3378. DOI: `10.2514/6.2016-3378` (vid. pág. 92).

Gupta, Suraj G, Mangesh M Ghonge y PM Jawandhiya (2013). "Review of unmanned aircraft system (UAS)". En: *International journal of advanced research in computer engineering & technology (IJARCET)* 2.4, págs. 1646 -1658 (vid. pág. 2).

Hammouda, Laroussi y col. (2013). "Improving mobile robot robustness in visual servoing application". En: *10th International Multi-Conferences on Systems, Signals & Devices 2013 (SSD13)*. IEEE, págs. 1-6 (vid. pág. 28).

Han, Long y col. (2010). "Bezier curve based path planning for autonomous vehicle in urban environment". En: *2010 IEEE Intelligent Vehicles Symposium*. IEEE, págs. 1036-1042 (vid. págs. 13, 42).

Harary, Gur y Ayellet Tal (2012a). "3D Euler spirals for 3D curve completion". En: *Computational Geometry: Theory and Applications* 45.3, págs. 115-126. ISSN: 09257721. DOI: `10.1016/j.comgeo.2011.10.001` (vid. págs. 53-55, 67, 70-72, 80, 81, 93, 111, 117).

— (2012b). "3D Euler spirals for 3D curve completion". En: vol. 45. 3, págs. 393 -402. ISBN: 9781450300162. DOI: `10.1016/j.comgeo.2011.10.001` (vid. pág. 126).

Hartman, Paul (1957). "The highway spiral for combining curves of different radii". En: *Trans. Amer. Soc. Civil Engin.* 122, págs. 389-409 (vid. pág. 30).

Hartmann, Kim y Keir Giles (2016). "UAV exploitation: A new domain for cyber power". En: *Institute of Electrical and Electronics Engineers (IEEE)*, págs. 205-221. DOI: `10.1109/CYCON.2016.7529436` (vid. pág. 92).

Hayat, Samira y col. (2017). "Multi-objective UAV path planning for search and rescue". En: *Proceedings - IEEE International Conference on Robotics and Automation*, págs. 5569-5574. ISSN: 10504729. DOI: `10.1109/ICRA.2017.7989656` (vid. pág. 120).

Hernández, Juan David y col. (2019). "Online motion planning for unexplored underwater environments using autonomous underwater vehicles". En: *Journal of Field Robotics* 36.2, págs. 370-396 (vid. pág. 45).

Ho, Yi Ju y Jing Sin Liu (2009). “Collision-free curvature-bounded smooth path planning using composite bezier curve based on voronoi diagram”. En: *Proceedings of IEEE International Symposium on Computational Intelligence in Robotics and Automation, CIRA*, págs. 1-6. ISBN: 978-1-4244-4808-1. DOI: `10.1109/CIRA.2009.5423161` (vid. pág. 3).

Hong, Haichao y col. (2019). “Model Predictive Convex Programming for Constrained Vehicle Guidance”. En: *IEEE Transactions on Aerospace and Electronic Systems* 55.5, págs. 2487-2500. DOI: `10.1109/TAES.2018.2890375` (vid. pág. 92).

Hong, Haichao y col. (2021). “Computationally Efficient Trajectory Generation for Smooth Aircraft Flight Level Changes”. En: *Journal of Guidance, Control, and Dynamics* 44.8, págs. 1532-1540. DOI: `10.2514/1.G005529`. eprint: `https://doi.org/10.2514/1.G005529` (vid. pág. 92).

Horváth, Ernő y Claudiu Radu Pozna (2021). “Clothoid-based Trajectory Following Approach for Self-driving vehicles”. En: *Institute of Electrical and Electronics Engineers (IEEE)*, págs. 251 -254. DOI: `10.1109/SAMI50585.2021.9378664` (vid. pág. 93).

Hota, Sikha y Debasish Ghose (2010a). “Optimal geometrical path in 3D with curvature constraint”. En: 2. IEEE, págs. 113-118. ISBN: 9781424466757. DOI: `10.1109/IROS.2010.5653663` (vid. pág. 123).

— (2010b). “Optimal path planning for an aerial vehicle in 3D space”. En: *49th IEEE Conference on Decision and Control (CDC)*, págs. 4902-4907. ISSN: 0743-1546. DOI: `10.1109/CDC.2010.5717246` (vid. pág. 67).

— (2010c). “Optimal path planning for an aerial vehicle in 3D space”. En: págs. 4902-4907. ISBN: 978-1-4244-7745-6. DOI: `10.1109/CDC.2010.5717246` (vid. pág. 123).

Huang, Liwei y col. (2016). “A novel coordinated path planning method using k-degree smoothing for multi-UAVs”. En: *Applied Soft Computing Journal* 48, págs. 182-192. ISSN: 15684946. DOI: `10.1016/j.asoc.2016.06.046` (vid. pág. 121).

Huang, Sunan y Rodney Swee Huat Teo (2019). “Computationally efficient visibility graph-based generation of 3D shortest collision-free path among polyhedral obstacles for unmanned aerial vehicles". En: *2019 International Conference on Unmanned Aircraft Systems, ICUAS 2019*, págs. 1218-1223. DOI: `10.1109/ICUAS.2019.8798322` (vid. pág. 121).

Huang, Sunan, Rodney Swee Huat Teo y Kok Kiong Tan (2019). "Collision avoidance of multi unmanned aerial vehicles: A review". En: *Annual Reviews in Control* 48, págs. 147-164. ISSN: 13675788. DOI: `10.1016/j.arcontrol.2019.10.001` (vid. pág. 121).

Huang, Xiaoyong y col. (2021). "A newly developed corner smoothing methodology based on clothoid splines for high speed machine tools". En: *Robotics and Computer-Integrated Manufacturing* 70.September 2020, pág. 102106. ISSN: 07365845. DOI: `10.1016/j.rcim.2020.102106` (vid. pág. 122).

Huh, Uk-Youl y Seong-Ryong Chang (2014). "A G2 Continuous Path-smoothing Algorithm Using Modified Quadratic Polynomial Interpolation". En: *International Journal of Advanced Robotic Systems* 11.2, pág. 25. ISSN: 1729-8814. DOI: `10.5772/57340` (vid. pág. 3).

Ilari, Joan y Carme Torras (1990). "2D path planning: A configuration space heuristic approach". En: *The International Journal of Robotics Research* 9.1, págs. 75-91 (vid. pág. 2).

Islam, Md. Shafiqul, Md. Mobasher Ahmed y Saad Islam (2018). "A conceptual system architecture for countering the civilian unmanned aerial vehicles threat to nuclear facilities". En: *International Journal of Critical Infrastructure Protection* 23, págs. 139-149. ISSN: 1874-5482. DOI: `https://doi.org/10.1016/j.ijcip.2018.10.003` (vid. pág. 92).

Jenkins, Darryl y Bijan Vasigh (2013). "The Economic Impact of Unmanned Aircraft Systems Integration in the United States". En: *Association for Unmanned Vehicle Systems International* March, págs. 1-40 (vid. pág. 3).

Jung, Dongwon y Panagiotis Tsiotras (2013). "On-Line Path Generation for Unmanned Aerial Vehicles Using B-Spline Path Templates". En: *Journal of Guidance, Control, and Dynamics* 36.6, págs. 1642-1653. ISSN: 0731-5090. DOI: `10.2514/1.60780` (vid. págs. 50, 66).

Kaiser, Benjamin y Alexander Verl (2021). "Planning of Curvature-Optimal Smooth Paths for Industrial Robots Using Neural Networks". En: *2021 4th*

International Conference on Artificial Intelligence for Industries (AI4I), págs. 10-15. DOI: 10.1109/ai4i51902.2021.00011 (vid. pág. 121).

Kan, Ee-May y col. (2011). "Contour Based Path Planning with B-Spline Trajectory Generation for Unmanned Aerial Vehicles (UAVs) over Hostile Terrain". En: *Journal of Intelligent Learning Systems and Applications* 03.03, págs. 122-130. ISSN: 2150-8402. DOI: 10.4236/jilsa.2011.33014 (vid. págs. 50, 66).

Kavraki, Lydia E y Steven M LaValle (2016). "Motion planning". En: *Springer handbook of robotics*. Springer, págs. 139-162 (vid. pág. 2).

Kawabata, Kuniaki y col. (2015). "A path generation for automated vehicle based on Bezier curve and via-points". En: *Robotics and Autonomous Systems* 74, págs. 243-252 (vid. págs. 12, 41).

Kim, Dae Jung y Chung Choo Chung (2021). "Automated Perpendicular Parking System with Approximated Clothoid-Based Local Path Planning". En: *IEEE Control Systems Letters* 5.6, págs. 1940-1945. ISSN: 07431619. DOI: 10.23919/ACC50511.2021.9483409 (vid. pág. 122).

Kim, Ki Bum y Byung Kook Kim (2011). "Minimum-time trajectory for three-wheeled omnidirectional mobile robots following a bounded-curvature path with a referenced heading profile". En: *IEEE Transactions on Robotics* 27.4, págs. 800-808 (vid. pág. 42).

Kim, YH y col. (2017). "Modified turn algorithm for motion planning based on clothoid curve". En: *Electronics Letters* 53.24, págs. 1574-1576 (vid. pág. 93).

Klein, Vladislav y Eugene A Morelli (2006). *Aircraft system identification: theory and practice*. American Institute of Aeronautics y Astronautics Reston, VA (vid. pág. 161).

Knuth, Donald E (1979). "Mathematical typography". En: *Bull. Amer. Math. Soc. (N.S.)* 1.2, págs. 337-372 (vid. pág. 71).

Kommer, Andreas y Thomas Weidner (2007). “Some aspects of clothoids”. En: *Proceedings of 7th International Conference on Applied Informatics* 1, págs. 115-120 (vid. pág. 67).

Kostov, V.P. y E.V. Degtiariova-Kostova (1995). *Some Properties of Clothoids.* Inf. téc. RR-2752, INRIA (vid. pág. 79).

Kühnel, Wolfgang (2015). *Differential geometry.* Vol. 77. American Mathematical Soc. (vid. pág. 12).

Lambert, ED, Richard Romano y David Watling (2019). “Optimal path planning with clothoid curves for passenger comfort”. En: *Proceedings of the 5th International Conference on Vehicle Technology and Intelligent Transport Systems (VEHITS 2019).* Vol. 1. SciTePress, págs. 609-615 (vid. págs. 30, 125).

Lambert, Edward Derek, Richard Romano y David Watling (2021). “Optimal Smooth Paths Based on Clothoids for Car-like Vehicles in the Presence of Obstacles”. En: *International Journal of Control, Automation and Systems* 19.6, págs. 2163-2182. DOI: `10.1007/s12555-020-0179-1` (vid. pág. 122).

Lau, Boris, Christoph Sprunk y Wolfram Burgard (2009). “Kinodynamic motion planning for mobile robots using splines”. En: *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems.* IEEE, págs. 2427 -2433 (vid. pág. 42).

Levien, R (2008a). *Technical Report UCB/EECS-2008-111, The Euler spiral: a mathematical history.* Inf. téc. EECS Department, University of California, Berkeley (vid. págs. 53, 67, 70).

Levien, Raph (2008b). “The Euler spiral: a mathematical history”. En: *Rapp. tech*, págs. 1-14 (vid. pág. 125).

Liang, Hu, Wang Zhong y Zhao Chunhui (2015). “Point-to-point near-optimal obstacle avoidance path for the unmanned aerial vehicle”. En: *2015 34th Chinese Control Conference (CCC).* IEEE, págs. 5413-5418. ISBN: 978-9-8815-6389-7. DOI: `10.1109/ChiCC.2015.7260486` (vid. pág. 66).

Liang, Zhao, Guoqiang Zheng y Jishun Li (2012). “Automatic parking path optimization based on bezier curve fitting”. En: *2012 IEEE International Conference on Automation and Logistics*. IEEE, págs. 583-587 (vid. págs. 13, 42).

Lima, Pedro F. y col. (2015). “Clothoid-Based Speed Profiler and Control for Autonomous Driving”. En: *Institute of Electrical and Electronics Engineers (IEEE)*, págs. 2194-2199. DOI: `10.1109/ITSC.2015.354` (vid. págs. 93, 122).

Liu, Chi Harold y col. (2018). “Energy-Efficient UAV Control for Effective and Fair Communication Coverage: A Deep Reinforcement Learning Approach”. En: *IEEE Journal on Selected Areas in Communications* 36.9, págs. 2059-2070. DOI: `10.1109/JSAC.2018.2864373` (vid. pág. 92).

Liu, Cunjia y Wen-Hua Chen (2016). “Disturbance rejection flight control for small fixed-wing unmanned aerial vehicles”. En: *Journal of Guidance, Control, and Dynamics* 39.12, págs. 2810-2819. DOI: `10.2514/1.G001958` (vid. pág. 92).

Liu, M., H. Liu y X. Wei (2014). “Autonomous position holding and point to point flying control of quadrotor UAVs”. En: *Proceeding of the 11th World Congress on Intelligent Control and Automation*. IEEE, págs. 2389-2394. ISBN: 978-1-4799-5825-2. DOI: `10.1109/WCICA.2014.7053095` (vid. pág. 66).

Liu, Ping y Qingquan Hu (2021). “Gaussian pseudospectral optimization method with smoothing penalty function path constraint handling for UAV obstacle avoidance planning”. En: *Proceeding - 2021 China Automation Congress, CAC 2021*. IEEE, págs. 7530-7534. ISBN: 9781665426473. DOI: `10.1109/CAC53003.2021.9728229` (vid. pág. 121).

Liu, Qingli y col. (2021). “Multi-UAV Path Planning Based on Fusion of Sparrow Search Algorithm and Improved Bioinspired Neural Network”. En: *IEEE Access* 9, págs. 124670-124681. ISSN: 21693536. DOI: `10.1109/ACCESS.2021.3109879` (vid. pág. 121).

Löbl, David y col. (2021). “Cooperative Docking Guidance and Control with Application to Civil Autonomous Aerial Refueling”. En: *Journal of Guidance, Control, and Dynamics*, págs. 1-11. DOI: `10.2514/1.G004425` (vid. pág. 92).

Lorentz, George G (2013). *Bernstein polynomials.* American Mathematical Soc. (vid. pág. 28).

Mac, Thi Thoa y col. (2016). "Heuristic approaches in robot path planning: A survey". En: *Robotics and Autonomous Systems* 86, págs. 13-28. ISSN: 09218890. DOI: `10.1016/j.robot.2016.08.001` (vid. pág. 123).

Markley, F Landis y col. (2014). "Euler angles". En: *Fundamentals of Spacecraft Attitude Determination and Control*, págs. 361-364 (vid. pág. 159).

Marzbani, Hormoz, Reza N. Jazar y M. Fard (2015). *Better Road Design Using Clothoids.* Springer, Cham, págs. 25-40. DOI: `10.1007/978-3-319-17999-5_3` (vid. págs. 51, 70).

Marzbani, Hormoz y col. (2015a). "Better road design for autonomous vehicles using clothoids". En: *Intelligent Interactive Multimedia Systems and Services.* Springer, págs. 265-278 (vid. pág. 3).

Marzbani, Hormoz y col. (2015b). *Better Road Design for Autonomous Vehicles Using Clothoids.* Springer, Cham, págs. 265-278. DOI: `10.1007/978-3-319-19830-9_24` (vid. págs. 51, 70).

Maza. I Caballero. F, Capitán. J Martinez-de-Dios. J R y Ollero. A (2011). "Experimental Results in Multi-UAV Coordination for Disaster Management and Civil Security Applications". En: *Springer* 61, págs. 563-585 (vid. pág. 2).

McCrae, J y K Singh (2008). "Sketching Piecewise Clothoid Curves". En: *Workshop on Sketch-Based Interfaces and Modeling* (vid. pág. 93).

McNaughton, Matthew y col. (2011). "Motion planning for autonomous driving with a conformal spatiotemporal lattice". En: *2011 IEEE International Conference on Robotics and Automation.* IEEE, págs. 4889-4895 (vid. pág. 43).

Meek, D. S. y D. J. Walton (2004a). "A note on finding clothoids". En: *Journal of Computational and Applied Mathematics* 170.2-2, págs. 433-453. ISSN: 03770427. DOI: `10.1016/j.cam.2003.12.047` (vid. págs. 3, 51, 70).

Meek, DS y DJ Walton (2004b). “An arc spline approximation to a clothoid”. En: *Journal of Computational and Applied Mathematics* 170.1, págs. 59-77. DOI: 10.1016/j.cam.2003.12.038 (vid. pág. 115).

Micula, Gheorghe y Sanda Micula (2012). *Handbook of splines.* Vol. 462. Springer Science & Business Media (vid. pág. 26).

Mielenz, Klaus D. (2000). “Computation of Fresnel Integrals II”. En: *Journal of Research of the National Institute of Standards and Technology* 105.4, págs. 589-590. ISSN: 1044677X. DOI: 10.6028/jres.105.049 (vid. págs. 51, 70, 81, 82).

Mittal, Shashi y Kalyanmoy Deb (2007a). “Three-dimensional offline path planning for UAVs using multiobjective evolutionary algorithms”. En: *2007 IEEE Congress on Evolutionary Computation.* IEEE, págs. 3195-3202 (vid. pág. 45).

— (2007b). “Three-Dimentional Offline Path Planning for UAVs Using Multiobjective Evolutionary Algorithms”. En: *2007 IEEE Congress on Evolutionary Computation (CEC’2007)* i, págs. 3195-3202. ISSN: 1089-778X. DOI: 10.1109/CEC.2007.4424880 (vid. pág. 67).

Mo, Hongwei y Ghulam Farid (2019). “Nonlinear and adaptive intelligent control techniques for quadrotor UAV–a survey”. En: *Asian Journal of Control* 21.2, págs. 989-1008. DOI: 10.1002/asjc.1758 (vid. pág. 92).

Montés, N y col. (2008). “Real-time clothoid approximation by Rational Bezier curves”. En: *IEEE Int. Conf. on Robotics and Automation*, págs. 2246-2251. DOI: 10.1109/ROBOT.2008.4543548 (vid. págs. 51, 70, 81).

Montes, Nicolas, Marta C Mora y Josep Tornero (2007). “Trajectory generation based on rational bezier curves as clothoids”. En: *2007 IEEE Intelligent Vehicles Symposium.* IEEE, págs. 505-510 (vid. págs. 13, 41, 42, 54, 56).

Montés, Nicolás y col. (2008). “Real-time clothoid approximation by Rational Bezier curves”. En: *2008 IEEE International Conference on Robotics and Automation.* IEEE, págs. 2246-2251 (vid. págs. 13, 42, 115).

Mualla, Yazan y col. (2019). "Agent-based simulation of unmanned aerial vehicles in civilian applications: A systematic literature review and research directions". En: *Future Generation Computer Systems* 100, págs. 344-364. ISSN: 0167-739X. DOI: `https://doi.org/10.1016/j.future.2019.04.051` (vid. pág. 92).

Narayan, Smita (2014). "Approximating Cornu spirals by arc splines". En: *Journal of Computational and Applied Mathematics* 255, págs. 789-804. ISSN: 03770427. DOI: `10.1016/j.cam.2013.06.038` (vid. págs. 51, 70).

Neto, Armando A., Douglas G. Macharet y Mario F.M. Campos (2010). "Feasible RRT-based path planning using seventh order Bézier curves". En: *IEEE/RSJ 2010 International Conference on Intelligent Robots and Systems, IROS 2010 - Conference Proceedings*, págs. 1445-1450. DOI: `10.1109/IROS.2010.5649145` (vid. pág. 121).

Neto, Armando Alves, Douglas G Macharet y Mario FM Campos (2013). "Feasible path planning for fixed-wing UAVs using seventh order Bézier curves". En: *Journal of the Brazilian Computer Society* 19.2, págs. 193-203 (vid. págs. 3, 44, 92, 115).

Nguen, V. F., A. V. Putov y T. T. Nguen (2017). "Adaptive control of an unmanned aerial vehicle". En: *AIP Conference Proceedings*. Vol. 1798. 1. AIP Publishing LLC, pág. 020124. DOI: `10.1063/1.4972716` (vid. pág. 66).

Nicotra, Marco M., Roberto Naldi y Emanuele Garone (2017). "Nonlinear control of a tethered UAV: The taut cable case". En: *Automatica* 78, págs. 174-184. ISSN: 0005-1098. DOI: `https://doi.org/10.1016/j.automatica.2016.12.018` (vid. pág. 92).

Nikolos, Ioannis K y col. (2003). "Evolutionary algorithm based offline/online path planner for UAV navigation". En: *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* 33.6, págs. 898-912 (vid. pág. 45).

Orfanus, Dalimir, Edison Pignaton de Freitas y Frank Eliassen (2016). "Self-Organization as a Supporting Paradigm for Military UAV Relay Networks". En: *IEEE Communications Letters* 20.4, págs. 804-807. DOI: `10.1109/LCOMM.2016.2524405` (vid. pág. 92).

Parlangeli, G y col. (2009). “A motion planning algorithm for smooth paths of bounded curvature and curvature derivative”. En: *2009 17th Mediterranean Conference on Control and Automation*. IEEE, págs. 73-78 (vid. pág. 42).

Pérez, Joshué y col. (2013). “Trajectory generator for autonomous vehicles in urban environments”. En: *2013 IEEE International Conference on Robotics and Automation*. IEEE, págs. 409-414 (vid. págs. 13, 42).

Pérez, Lucía Hilario y col. (2018). “Path Planning Based on Parametric Curves”. En: *Advanced Path Planning for Mobile Entities*. InTech. DOI: `10.5772/intechopen.72574` (vid. págs. 66, 81).

Pfeifle, Ole y Walter Fichter (2021). “Cascaded Incremental Nonlinear Dynamic Inversion for Three-Dimensional Spline-Tracking with Wind Compensation”. En: *Journal of Guidance, Control, and Dynamics*, págs. 1-13. DOI: `10.2514/1.G005785` (vid. pág. 115).

Phillips, AB y col. (2017). “Understanding the power requirements of autonomous underwater systems, Part I: An analytical model for optimum swimming speeds and cost of transport”. En: *Ocean Engineering* 133, págs. 271-279 (vid. pág. 2).

Pinkney, Maj FJ, Dan Hampel y Stef DiPierro (1996). “Unmanned aerial vehicle (UAV) communications relay”. En: *Proceedings of MILCOM’96 IEEE Military Communications Conference*. Vol. 1. IEEE, págs. 47-51 (vid. pág. 2).

Prabhakar, Nirmit y col. (2018). “Trajectory-Driven Adaptive Control of Autonomous Unmanned Aerial Vehicles with Disturbance Accommodation”. En: *Journal of Guidance, Control, and Dynamics* 41.9, págs. 1976-1989. ISSN: 0731-5090. DOI: `10.2514/1.G003341` (vid. pág. 66).

Press, William H., Teukolsky, Saul A., Vetterling, William T., and Flannery, Brian P. (2007). *Numerical recipes: the art of scientific computing*. 3rd. Cambridge University Press, pág. 1235. ISBN: 9780521880688 (vid. págs. 55, 71, 80).

Rastelli, Joshue Perez, Ray Lattarulo y Fawzi Nashashibi (2014). “Dynamic trajectory generation using continuous-curvature algorithms for door to door assistance vehicles”. En: *2014 IEEE Intelligent Vehicles Symposium Proceedings*. IEEE, págs. 510-515 (vid. págs. 12, 41).

Ravankar, Abhijeet y col. (2018). “Path smoothing techniques in robot navigation: State-of-the-art, current and future challenges”. En: *Sensors (Switzerland)* 18.9, págs. 1-30. ISSN: 14248220. DOI: 10.3390/s18093170 (vid. pág. 3).

Rikovitch, Nir e Inna Sharf (2013). “Kinodynamic Motion Planning for UAVs: A Minimum Energy Approach”. En: *AIAA Guidance, Navigation, and Control (GNC) Conference*. Reston, Virginia: American Institute of Aeronautics y Astronautics. ISBN: 978-1-62410-224-0. DOI: 10.2514/6.2013-5231 (vid. pág. 67).

Roberge, Vincent, Mohammed Tarbouchi y Gilles Labonte (2018). “Fast Genetic Algorithm Path Planner for Fixed-Wing Military UAV Using GPU”. En: *IEEE Transactions on Aerospace and Electronic Systems* 54, págs. 2105 -2117. DOI: 10.1109/TAES.2018.2807558 (vid. pág. 92).

Rubio Hervas, Jaime y col. (2016). “Nonlinear control of fixed-wing UAVs in presence of stochastic winds”. En: *Communications in Nonlinear Science and Numerical Simulation* 33, págs. 57-69. ISSN: 1007-5704. DOI: 10.1016/J.CNSNS.2015.08.026 (vid. pág. 66).

Ryan, Allison y col. (2004). “An overview of emerging results in cooperative UAV control”. En: *Proceedings of the IEEE Conference on Decision and Control* 1, págs. 602-607. ISSN: 25762370. DOI: 10.1109/cdc.2004.1428700 (vid. pág. 120).

Sadeghi, Javad y col. (2022). “Investigation on optimum lengths of railway clothoid transition curves based on passenger ride comfort”. En: *Proceedings of the Institution of Mechanical Engineers, Part F: Journal of Rail and Rapid Transit*, pág. 09544097221143824 (vid. pág. 30).

Sahingoz, Ozgur Koray (2014). “Generation of bezier curve-based flyable trajectories for multi-UAV systems with parallel genetic algorithm”. En: *Journal of Intelligent & Robotic Systems* 74.1-2, págs. 499-511 (vid. pág. 41).

Sampaio, Jorge HB (2017). “Designing three-dimensional directional well trajectories using Bézier curves”. En: *Journal of Energy Resources Technology* 139.3 (vid. pág. 43).

Sánchez-Reyes, J. y J.M. Chacón (2003). “Polynomial approximation to clothoids via s-power series”. En: *Computer-Aided Design* 35.14, págs. 1305-1313. ISSN: 0010-4485. DOI: `10.1016/S0010-4485(03)00045-9` (vid. pág. 115).

Scheuer, A y T Fraichard (1997). “Collision-free and continuous-curvature path planning for car-like robots”. En: *IEEE Int. Conf. on Robotics and Automation*, 867-873 vol.1 (vid. pág. 117).

Scheuer, A y Th. Fraichard (1996a). “Planning continuous-curvature paths for car-like robots”. En: *IEEE Int. Conf. on Intelligent Robots and Systems*, 1304-1311 vol.3 (vid. págs. 94, 95).

Scheuer, Alexis y Th Fraichard (1996b). “Planning continuous-curvature paths for car-like robots”. En: *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems. IROS’96.* Vol. 3. IEEE, págs. 1304 -1311 (vid. pág. 128).

Sedighi, Saeid, Duong-Van Nguyen y Klaus-Dieter Kuhnert (2019). “A New Method of Clothoid-Based Path Planning Algorithm for Narrow Perpendicular Parking Spaces”. En: *Proceedings of the 5th International Conference on Mechatronics and Robotics Engineering.* ICMRE’19. Rome, Italy: Association for Computing Machinery, págs. 50-55. ISBN: 9781450360951. DOI: `10.1145/3314493.3314512` (vid. pág. 93).

Segovia, A y col. (1991). “Comparative study of the different methods of path generation for a mobile robot in a free environment”. En: *Advanced Robotics, 1991.’Robots in Unstructured Environments’, 91 ICAR., Fifth International Conference on.* IEEE, págs. 1667-1670 (vid. págs. 28, 54).

Shah, MA y N Aouf (2010). “3d cooperative pythagorean hodograph path planning and obstacle avoidance for multiple uavs”. En: *2010 IEEE 9th International Conference on Cyberntic Intelligent Systems.* IEEE, págs. 1-6 (vid. pág. 46).

Shanmugavel, Madhavan y col. (2006). “A solution to simultaneous arrival of multiple UAVs using Pythagorean hodograph curves”. En: *2006 American Control Conference*. IEEE, 6-pp (vid. pág. 44).

Shao, Zhuang y col. (2019). “Path planning for multi-UAV formation rendezvous based on distributed cooperative particle swarm optimization”. En: *Applied Sciences* 9.13, pág. 2621 (vid. pág. 46).

Shen, Jieliang y col. (2018). “Calculation and identification of the aerodynamic parameters for small-scaled fixed-wing UAVs”. En: *Sensors (Switzerland)* 18.1, págs. 1-18. ISSN: 14248220. DOI: 10.3390/s18010206 (vid. pág. 121).

Shin, Dong Hun y Sanjiv Singh (1990). *Path Generation for Robot Vehicles Using Composite Clothoid Segments.* Inf. téc. CMU-RI-TR-90-31. Pittsburgh, PA: Robotics Institute (vid. pág. 67).

Silva, Junior A.R. y Valdir Grassi (2018). “Clothoid-based global path planning for autonomous vehicles in urban scenarios”. En: *Proceedings - IEEE International Conference on Robotics and Automation*, págs. 4312-4318. ISSN: 10504729. DOI: 10.1109/ICRA.2018.8461201 (vid. pág. 122).

Singh, Nikhil Kumar y Sikha Hota (2017). “Optimal path planning for fixed-wing UAVs in 3D space”. En: *ICTACEM, IIT Kharagpur* (vid. pág. 45).

Stewart, James y col. (2012). *Cálculo de varias variables: trascendentes tempranas.* Inf. téc. Cengage learning (vid. pág. 13).

Stöcker, Claudia y col. (2017). “Review of the current state of UAV regulations”. En: *Remote sensing* 9.5, pág. 459 (vid. pág. 2).

Su, Tingting y col. (2018). “Time-optimal trajectory planning for delta robot based on quintic pythagorean-hodograph curves”. En: *IEEE Access* 6, págs. 28530-28539 (vid. pág. 46).

Subchan, S y col. (2008). “Pythagorean hodograph (ph) path planning for tracking airborne contaminant using sensor swarm”. En: *2008 IEEE Instrumentation and Measurement Technology Conference*. IEEE, págs. 501-506 (vid. pág. 44).

Sui, Liyang y col. (2020). "Path Planning based on Clothoid for Autonomous Valet Parking". En: *2020 17th International Computer Conference on Wavelet Active Media Technology and Information Processing, ICCWAMTIP 2020*, págs. 389-393. DOI: `10.1109/ICCWAMTIP51612.2020.9317391` (vid. pág. 122).

Sujit, P.B., Srikanth Saripalli y Joao Borges Sousa (2014). "Unmanned Aerial Vehicle Path Following: A Survey and Analysis of Algorithms for Fixed-Wing Unmanned Aerial Vehicless". En: *IEEE Control Systems* 34.1, págs. 42-59. ISSN: 1066-033X. DOI: `10.1109/MCS.2013.2287568` (vid. págs. 50, 66).

Tan, Li y col. (2022). "Optimization of UAV 3D trajectory smoothing based on improved PH curves". En: *2022 7th International Conference on Intelligent Computing and Signal Processing, ICSP 2022* 1, págs. 1219-1222. DOI: `10.1109/ICSP54964.2022.9778454` (vid. pág. 121).

Tounsi, M y J F Le˜Corre (1996). "Trajectory generation for mobile robots". En: *Mathematics and Computers in Simulation* 41.3-4, págs. 367-376. ISSN: 0378-4754. DOI: `http://dx.doi.org/10.1016/0378-4754(95)00085-2` (vid. pág. 67).

Upadhyay, Saurabh y Ashwini Ratnoo (2015). "Smooth path planning for passages with heading and curvature discontinuities". En: *IEEE International Conference on Intelligent Robots and Systems* 2015-Decem.1, págs. 2672-2677. ISSN: 21530866. DOI: `10.1109/IROS.2015.7353742` (vid. pág. 123).

Vanegas, Gloria y col. (2018). "Smooth 3D path planning for non-holonomic UAVs". En: *2018 7th International Conference on Systems and Control (ICSC)*. IEEE, págs. 1-6 (vid. pág. 122).

Varshosaz, Masood y col. (2020). "Spoofing Detection of Civilian UAVs Using Visual Odometry". En: *ISPRS International Journal of Geo-Information* 9.1. ISSN: 2220-9964. DOI: `10.3390/ijgi9010006` (vid. pág. 92).

Vázquez-Méndez, Miguel E. y G. Casal (2016). "The Clothoid Computation: A Simple and Efficient Numerical Algorithm". En: *Journal of Surveying Engineering* 142.3, págs. 1-18. ISSN: 0733-9453. DOI: `10.1061/(ASCE)SU.1943-5428.0000177` (vid. pág. 125).

Velasco, J y Sergio García-Nieto (2014). “Unmanned aerial vehicles model identification using multi-objective optimization techniques”. En: *IFAC Proceedings Volumes* 47.3, págs. 8837-8842 (vid. págs. 157, 162).

Velasco-Carrau, J y col. (2015). “Multi-objective optimization for wind estimation and aircraft model identification”. En: *Journal of Guidance, Control, and Dynamics* 39.2, págs. 372-389 (vid. págs. 58, 85).

Velasco Carrau, Jesús y col. (2012). “Desarrollo y evaluación de una estación de control de tierra para vehículos aéreos no tripulados”. En: *Actas de las XXXIII Jornadas de Automática* (vid. pág. 156).

Velasco-Carrau, Jesús y col. (2016). “Multi-objective optimization for wind estimation and aircraft model identification”. En: *Journal of Guidance, Control, and Dynamics* 39.2, págs. 372-389. DOI: `10.2514/1.G001294` (vid. pág. 121).

Vinokursky, D. L., O. S. Mezentceva y Ph. V. Samoylov (2020). “Trajectory Planning of UAV Group: Pythagorean Hodograph and Bernstein-Bezier Composite Curves in the Plane”. En: *Institute of Electrical and Electronics Engineers (IEEE)*, págs. 704-707. DOI: `10.1109/RusAutoCon49822.2020.9208121` (vid. págs. 92, 115).

Walton, DJ, DS Meek y JM Ali (2003). “Planar G2 transition curves composed of cubic Bézier spiral segments”. En: *Journal of Computational and Applied Mathematics* 157.2, págs. 453-476 (vid. págs. 12, 42).

Wan, Tao Ruan y col. (2011). “A real-time 3D motion planning and simulation scheme for nonholonomic systems”. En: *Simulation Modelling Practice and Theory* 19.1, págs. 423-439. DOI: `10.1016/j.simpat.2010.08.002` (vid. págs. 67, 93).

Wang, Xiaoliang y col. (2017a). “Curvature continuous and bounded path planning for fixed-wing UAVs”. En: *Sensors (Switzerland)* 17.9, págs. 1-21. ISSN: 14248220. DOI: `10.3390/s17092155` (vid. págs. 50, 66).

— (2017b). “Curvature Continuous and Bounded Path Planning for Fixed-Wing UAVs”. En: *Sensors* 17.9. ISSN: 1424-8220. DOI: `10.3390/s17092155` (vid. págs. 92, 98, 115).

Wang, Yongqiang y col. (2009). “Fault detection of networked control systems with packet based periodic communication”. En: *International Journal of Adaptive Control and Signal Processing* 23.8, págs. 682-698 (vid. pág. 2).

Wang, Zhong, Yan Li y Wenchao Li (2014). “An approximation path planning algorithm for fixed-wing UAVs in stationary obstacle environment”. En: *Proceedings of the 33rd Chinese Control Conference, CCC 2014*, págs. 664 -669. ISSN: 21612927. DOI: `10.1109/ChiCC.2014.6896704` (vid. pág. 121).

Wargo, Chris A. y col. (2014). “Unmanned Aircraft Systems (UAS) research and future analysis”. En: *IEEE Aerospace Conference Proceedings*, págs. 1 -16. ISSN: 1095323X. DOI: `10.1109/AERO.2014.6836448` (vid. pág. 120).

Wilburn, Jennifer N., Mario G. Perhinschi y Brenton K. Wilburn (2013). “Implementation of Composite Clothoid Paths for Continuous Curvature Trajectory Generation for UAVs”. En: *AIAA Guidance, Navigation, and Control (GNC) Conference*. Reston, Virginia: American Institute of Aeronautics y Astronautics. ISBN: 978-1-62410-224-0. DOI: `10.2514/6.2013-5230` (vid. pág. 67).

Wilde, Doran K. (2009). “Computing clothoid segments for trajectory generation”. En: *Institute of Electrical and Electronics Engineers (IEEE)*, págs. 2440-2445. DOI: `10.1109/IROS.2009.5354700` (vid. pág. 93).

Wu, Jiayang y col. (2017). “Cooperative path planning of multiple Uavs based on ph curves and harmony search algorithm”. En: *2017 IEEE 21St international conference on computer supported cooperative work in design (CSCWD)*. IEEE, págs. 540-544 (vid. pág. 46).

Wu, Xiande y col. (2018). “A hybrid algorithm of particle swarm optimization, metropolis criterion and RTS smoother for path planning of UAVs”. En: *Applied Soft Computing Journal* 73, págs. 735-747. ISSN: 15684946. DOI: `10.1016/j.asoc.2018.09.011` (vid. pág. 121).

Wynn, Russell B y col. (2014). “Autonomous Underwater Vehicles (AUVs): Their past, present and future contributions to the advancement of marine geoscience”. En: *Marine Geology* 352, págs. 451-468 (vid. pág. 2).

Xiang, Haitao y Lei Tian (2011). “Development of a low-cost agricultural remote sensing system based on an autonomous unmanned aerial vehicle

(UAV)”. En: *Biosystems Engineering* 108.2, págs. 174-190. ISSN: 15375110. DOI: `10.1016/j.biosystemseng.2010.11.010` (vid. pág. 120).

Xiao, Qun-Bao y col. (2020). “Space corner smoothing of CNC machine tools through developing 3D general clothoid”. En: *Robotics and Computer-Integrated Manufacturing* 64, pág. 101949. ISSN: 0736-5845. DOI: `10.1016/j.rcim.2020.101949` (vid. pág. 93).

Yang, Guang-Zhong y col. (2017). “Medical robotics-Regulatory, ethical, and legal considerations for increasing levels of autonomy”. En: *Science Robotics* 2.4, eaam8638. DOI: `10.1126/scirobotics.aam8638` (vid. pág. 2).

Yang, Kwangjin y Salah Sukkarieh (2008a). “3D smooth path planning for a UAV in cluttered natural environments”. En: *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS*, págs. 794-800. ISSN: 2153-0858. DOI: `10.1109/IROS.2008.4650637` (vid. págs. 3, 40, 67).

— (2008b). “3D smooth path planning for a UAV in cluttered natural environments”. En: *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS*, págs. 794-800. DOI: `10.1109/IROS.2008.4650637` (vid. pág. 121).

— (2008c). “Real-time continuous curvature path planning of UAVs in cluttered environments”. En: *2008 5th International Symposium on Mechatronics and Its Applications*. IEEE, págs. 1-6 (vid. pág. 121).

Yang, Kwangjin y col. (2014a). “Spline-based RRT path planner for nonholonomic robots”. En: *Journal of Intelligent & Robotic Systems* 73.1-4, págs. 763-782 (vid. pág. 40).

Yang, Liang y col. (2014b). “Guiding attraction based random tree path planning under uncertainty: Dedicate for UAV”. En: *2014 IEEE international conference on mechatronics and automation*. IEEE, págs. 1182-1187 (vid. pág. 44).

Yang, Liang y col. (2015). “Generation of dynamically feasible and collision free trajectory by applying six-order Bezier curve and local optimal reshaping”. En: *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, págs. 643-648 (vid. pág. 44).

Yang, Liang y col. (2016). “Survey of robot 3D path planning algorithms”. En: *Journal of Control Science and Engineering* 2016 (vid. pág. 3).

Yang, Xiuxia, Weiwei Zhou y Yi Zhang (2016). “On collaborative path planning for multiple UAVs based on pythagorean hodograph curve”. En: *2016 IEEE Chinese Guidance, Navigation and Control Conference (CGNCC)*. IEEE, págs. 971-975 (vid. pág. 44).

Yi, ZHANG y col. (2015). “STUDY OF THREE-DIMENSIONAL ON-LINE PATH PLANNING FOR UAV BASED ON PYTHAGOREAN HODOGRAPH CURVE.” En: *International Journal on Smart Sensing & Intelligent Systems* 8.3 (vid. pág. 46).

Zeng, Zheng y col. (2015). “A survey on path planning for persistent autonomy of autonomous underwater vehicles”. En: *Ocean Engineering* 110, págs. 303-313. ISSN: 00298018. DOI: `10.1016/j.oceaneng.2015.10.007` (vid. pág. 66).

Zhai, Ruiyong y col. (2014). “Control and navigation system for a fixed-wing unmanned aerial vehicle”. En: *AIP Advances* 4.3, pág. 031306. ISSN: 2158-3226. DOI: `10.1063/1.4866169` (vid. pág. 66).

Zhang, Long y col. (2019a). “A survey on 5G millimeter wave communications for UAV-assisted wireless networks”. En: *IEEE Access* 7, págs. 117460 -117504. ISSN: 21693536. DOI: `10.1109/ACCESS.2019.2929241` (vid. pág. 120).

Zhang, Shuhang y col. (2019b). “Cellular UAV-to-X communications: Design and optimization for multi-UAV networks”. En: *IEEE Transactions on Wireless Communications* 18.2, págs. 1346-1359 (vid. pág. 120).

Zhao, Yijing, Zheng Zheng y Yang Liu (2018). “Survey on computational-intelligence-based UAV path planning”. En: *Knowledge-Based Systems* 158, págs. 54 -64. ISSN: 09507051. DOI: `10.1016/j.knosys.2018.05.033` (vid. pág. 120).

Zhou, Guoqing y Deyan Zang (2007). “Civil UAV system for earth observation”. En: *International Geoscience and Remote Sensing Symposium (IGARSS)* 757, págs. 5319-5321. DOI: `10.1109/IGARSS.2007.4424063` (vid. pág. 2).