



Universidad
Rey Juan Carlos

ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA DE TELECOMUNICACIÓN

INGENIERÍA DE ROBÓTICA SOFTWARE

TRABAJO FIN DE GRADO

PLANIFICACIÓN DE MANIOBRAS DE
MOVIMIENTO DE AERONAVES EN TIERRA
MEDIANTE TRACTORES ROBÓTICOS
towbarless

AUTORA: BLANCA SORIA RUBIO

TUTOR: ERNESTO STAFFETTI GIAMMARIA

Curso Académico 2023-2024

©2023 Blanca Soria Rubio

Algunos derechos reservados

Este documento se distribuye bajo la licencia Atribución- CompartirIgual 4.0 Internacional de Creative Commons, disponible en:

<https://creativecommons.org/licenses/by-sa/4.0/deed.es>

Índice general

1. Introducción	9
1.1. <i>Pushback</i>	9
1.1.1. Accidentes y peligros	11
1.1.2. Motivos de los incidentes	12
1.1.3. Target Level of Safety	13
1.1.4. Métodos de reducción de riesgo	13
1.2. Modelos de tractor existentes	14
1.3. Limitaciones	15
1.4. Objetivos	16
2. Modelado y simulación de un sistema	17
2.1. Modelo	17
2.1.1. Modelado dinámico	17
2.1.2. Modelado cinemático	18
2.1.3. Restricciones holonómicas	18
2.1.4. Restricciones no holonómicas	18
2.1.5. Servo motores	18
2.2. Simulación	19
2.2.1. Simulaciones con animación gráfica	19
2.2.2. Algoritmo de simulación	21
3. Planificación de maniobras para un vehículo tractor	26
3.1. Modelo <i>car-like</i>	26
3.2. <i>Chained form</i>	31
3.2.1. Modelo de tractor en <i>chained form</i>	32
3.3. Métodos de <i>steering</i> para un tractor de tipo <i>car-like</i> en <i>chained form</i>	34
3.3.1. Método 1: <i>Asymptotic Sinusoidal Inputs</i>	34

3.3.2.	Método 2: <i>Piecewise Constant Inputs</i>	41
3.3.3.	Método 3: <i>Polynomial Inputs</i>	46
3.4.	Resultados	48
3.4.1.	Experimento 1: <i>Asymptotic Sinusoidal Inputs</i>	48
3.4.2.	Experimento 2: <i>Piecewise Constant Inputs</i>	55
3.4.3.	Experimento 3: <i>Polynomial Inputs</i>	59
4.	Planificación de maniobras para un tractor con remolque	62
4.1.	Modelo de un sistema tractor-remolque	63
4.2.	Modelo tractor-remolque en <i>chained form</i>	64
4.3.	Métodos de <i>steering</i> para un sistema tractor-remolque en <i>chained form</i> . .	67
4.3.1.	Método basado en <i>Aymptotic Sinusoidal Inputs</i>	68
4.3.2.	Método basado en <i>Polynomial Inputs</i>	70
4.4.	Resultados	72
4.4.1.	Experimento 1: maniobra de desplazamiento paralelo	72
4.4.2.	Experimento 2: maniobra de <i>pushback</i>	75
4.5.	Singularidades	79
4.5.1.	Solución 1: Planificación de trayectoria en 2 pasos	80
4.5.2.	Solución 2: Rotación de todo el camino	82
5.	Planificación de maniobras para aeronaves en tierra	84
5.1.	Modelo un sistema tractor-avión	84
5.1.1.	Modelo de un avión	84
5.1.2.	Modelo de un tractor	86
5.1.3.	Modelo completo tractor-avión	88
5.2.	Resultados	89
5.2.1.	Experimento 1: maniobra de <i>pushback</i>	89
5.2.2.	Experimento 2: maniobra de desplazamiento paralelo	92
6.	Conclusiones	95
	Bibliografía	97

Índice de figuras

1.1.	Fases e involucrados en un <i>pushback</i>	10
2.1.	Sketch del chasis de un vehículo tractor de tipo <i>car-like</i>	20
3.1.	Distintas configuraciones de un vehículo <i>car-like</i>	27
3.2.	Modelo de un vehículo <i>car-like</i>	28
3.3.	Maniobra recta de un vehículo tractor con <i>Asymptotic Synusoidal Inputs</i> .	49
3.4.	Maniobra de desplazamiento paralelo de un vehículo tractor con <i>Asymptotic Synusoidal Inputs</i>	50
3.5.	Maniobra de desplazamiento paralelo de un vehículo tractor con orientación final distinta y <i>Asymptotic Synusoidal Inputs</i>	51
3.6.	Maniobra de desplazamiento paralelo de un vehículo tractor con cambio de orientación y <i>Asymptotic Synusoidal Inputs</i>	52
3.7.	Maniobra de desplazamiento paralelo de un vehículo tractor con <i>Asymptotic Synusoidal Inputs</i> y $a_1 = 10$	53
3.8.	Maniobra de desplazamiento paralelo de un vehículo tractor con <i>Asymptotic Synusoidal Inputs</i> y $a_1 = 20$	54
3.9.	Comparación de caminos en el plano XY que toma el método <i>Asymptotic Synusoidal Inputs</i> en desplazamientos diagonales.	55
3.10.	Maniobra recta de un vehículo tractor con <i>Piecewise Constant Inputs</i>	56
3.11.	Maniobra de desplazamiento diagonal de un vehículo tractor con <i>Piecewise Constant Inputs</i>	57
3.12.	Maniobra de retroceso de un vehículo tractor con <i>Piecewise Constant Inputs</i>	58
3.13.	Maniobra de desplazamiento diagonal de un vehículo tractor con <i>Polynomial Inputs</i>	60
3.14.	Maniobra de retroceso de un vehículo tractor con <i>Polynomial Inputs</i>	61
4.1.	Modelo de un tractor de tipo <i>car-like</i> con un remolque.	63

4.2. Maniobra de desplazamiento paralelo con <i>Asymptotic Synusoidal Inputs</i> y $a_1 = 10$	73
4.3. Maniobra de desplazamiento paralelo con <i>Asymptotic Synusoidal Inputs</i> y $a_1 = 20$	74
4.4. Comparación de maniobras paralelas con distintas orientaciones en un sistema tractor-remolque.	75
4.5. Maniobra de desplazamiento diagonal con un sistema tractor-remolque con <i>Polynomial Inputs</i>	76
4.6. Maniobra de retroceso con <i>Polynomial Inputs</i>	77
4.7. Maniobra de desplazamiento cualquiera con un sistema tractor-remolque y <i>Polynomial Inputs</i>	78
4.8. Maniobra propuesta para abordar las singularidades.	80
4.9. Comparación de maniobras análogas.	81
4.10. Representación estroboscópica de trayectorias sin singularidades de un sistema tractor-remolque en 2 pasos.	82
4.11. Comparación de rotaciones de $\gamma = \frac{\pi}{4} $	83
4.12. Representación estroboscópica de trayectorias sin singularidades de un sistema tractor-remolque rotado.	83
5.1. Medidas características para representar un avión.	85
5.2. Sketch de un avión <i>Airbus A320</i>	86
5.3. Medidas características de un tractor.	87
5.4. Sketch de un tractor.	87
5.5. Sketch de un sistema tractor-avión completo.	88
5.6. Movimiento estroboscópico de una maniobra <i>pushback</i>	90
5.7. Experimento 1: Variables de una maniobra <i>pushback</i>	90
5.8. Experimento 2: variables de una maniobra de <i>parking</i> con parámetro $a_1 = 10$	93
5.9. Experimento 2: variables de una maniobra de <i>parking</i> con parámetro $a_1 = 20$	94

Resumen

Las operaciones aeroportuarias de tierra abarcan una amplia gama de actividades dentro de la gestión en aeropuertos, como el movimiento de aeronaves, el repostaje de las mismas, la manipulación de equipajes y los servicios generales de plataforma. Las operaciones de movimiento en tierra de aeronaves comprenden la rodadura y las maniobras en la plataforma, que incluyen el *pushback*, y las operaciones de desplazamiento.

La fase de rodadura es el proceso de guiar una aeronave en tierra de un lugar a otro, por ejemplo de la posición de estacionamiento a la pista de despegue o de la pista de aterrizaje a la plataforma. En las operaciones de desplazamiento las aeronaves se mueven de un lugar de estacionamiento a otro o a un hangar para el mantenimiento y suelen incluir movimientos de retroceso.

Actualmente, las operaciones de rodadura se ejecutan mediante los motores de las aeronaves, así como las maniobras de avance en la plataforma, mientras que las maniobras de *pushback* y de desplazamiento se ejecutan utilizando equipos especializados, conocidos como tractores de remolque ya que no existe la posibilidad de hacer movimientos de retroceso usando los motores de las aeronaves.

Todas las operaciones de movimiento en tierra de aeronaves ejecutadas mediante tractores conllevan muchos riesgos debido a las dimensiones de las aeronaves comerciales y a que las ejecutan operadores humanos con visibilidad limitada en un entorno complejo, con obstáculos estáticos y en movimiento.

De hecho, los análisis de los datos históricos de accidentes muestran cómo el *Target Level of Safety* de estas maniobras no cumple con los estándares de seguridad establecidos por la International Civil Aviation Organization ¹.

Para contribuir a paliar este problema, en este trabajo de Fin de Grado se propone estudiar la posibilidad de ejecutar maniobras de asistencia en tierra de aeronaves mediante un vehículo tractor que sea un robot móvil.

Se asumen modelos cinemáticos tanto para el tractor como para el avión y un modelo de tipo *car-like* para el tractor. Por tanto, el modelo del sistema compuesto por el tractor y el avión puede asimilarse a un *car-like with trailer*, un sistema muy estudiado en robótica móvil.

En particular, se estudia el problema de la planificación de estas maniobras, que se resuelve como un problema denominado *steering* del estado para mover el sistema de un

¹The need for a collision prevention system for the pushback of aircraft to minimize hazards caused by narrow spaces. Franziska Dieke-Meier and Hartmut Fricke[1]

estado a otro, que es particularmente difícil de resolver por la presencia de restricciones no holonómicas.

El problema del *steering* del estado de un sistema *car-like with trailer* se resuelve mediante la transformación del modelo a *chained form*. El sistema propuesto se enfoca únicamente en la tarea del *steering* del estado, por lo que no considera la evasión de obstáculos y en consecuencia no emplea ningún tipo de retroalimentación ni información sensorial del entorno.

Se han implementado varios métodos de *steering* para sistemas en *chained form*: *Asymptotic Sinusoidal Inputs*, *Piecewise Constant Inputs* y *Polynomial Inputs*. El método *Asymptotic Sinusoidal Inputs* es ideal en maniobras de tipo *parallel parking* y *Polynomial Inputs* es el método ideal para el resto de maniobras.

Capítulo 1

Introducción

Dentro de las operaciones para aviones, las operaciones de tierra, encargadas de los servicios que recibe una aeronave en el suelo, son consideradas de las más potencialmente peligrosas [1]. Los datos en 2004 provistos por la Airport Council International (ACI) muestran que los accidentes que involucran una aeronave abarcan el 30% de todos los accidentes dentro de un aeropuerto. En el contexto de Prevención de Accidentes en Tierra (GAP) la Flight Safety Foundation estima que los costes materiales de incidentes en plataforma suponen una media de 10 billones de dólares al año a nivel mundial, una cifra no menos alarmante que el riesgo que suponen para el personal de tierra que se involucra en las mismas.

1.1. *Pushback*

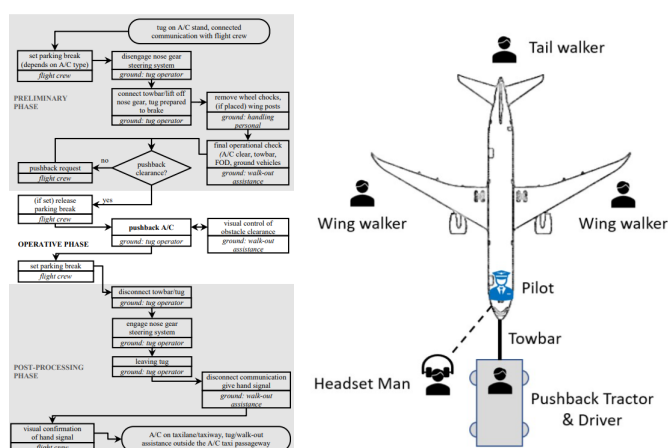
Una de estas operaciones, que implica un riesgo especialmente notable, es el proceso de *pushback*. Esta maniobra se aplica a una aeronave desde su posición en la plataforma de embarque hacia las calles de rodaje (también llamadas pistas de *taxiing*), donde posteriormente será dirigida con sus propios motores a la pista de despegue para volar. Este desplazamiento se hace con un vehículo tractor que empuja el avión a modo de remolque marcha atrás, ya que el sentido de los motores de un avión no permite un movimiento de retroceso por sí solo.

En el mercado se pueden encontrar 2 clases de vehículos tractores: unos tienen un *tow-bar* [6], una barra que conecta a este con el remolque, y otros que no lo tienen. Esta barra otorga al modelo cinemático de una complejidad mayor para planificar los movimientos y además la barra es susceptible de romperse si se le aplican fuerzas mal dirigidas o demasiado grandes. La ventaja que concede al sistema es la longitud segura que proporciona al operario del vehículo, dado que se encontrará a una distancia mayor al morro del avión en

comparación a un tractor de *pushback* sin *towbar*. La naturaleza automática del *steering* que se propone en este trabajo asume que no existe ningún conductor dentro del vehículo, por lo que escoger un tractor sin barra será la mejor opción para el sistema.

El personal encargado de manejar esta maniobra consta del conductor del tractor y varios operarios de tierra que se encargan de visualizar cada una de las partes del avión que pueden ser susceptibles de choque y de informar al conductor de cada movimiento que realiza, unos en las alas (*sidewalkers* o *wingwalkers*) y otros en la cola (*tailwalkers*). Estos además se ocupan de las operaciones de anclaje y desenganche del tractor al tren de aterrizaje. A estos trabajadores se les añaden el piloto de la aeronave y las personas involucradas en la torre de control.

La maniobra de *pushback*, como toda operación segura, consta de una serie de pasos detalladamente planificados [1]. La autorización para comenzar la maniobra la da el piloto del avión, el conductor del tractor, un miembro de la torre de control o un encargado específico de la maniobra según la normativa de cada aeropuerto. Tras el proceso de enganche del tractor al tren de aterrizaje y todos los pasos previos de seguridad, el conductor dirige el avión a una velocidad suficientemente lenta como para que los operarios puedan caminar junto al vehículo e informar de la estabilidad de sus movimientos. Estos operarios se posicionan junto al tractor, las alas y la parte trasera de la aeronave, cumpliendo el principio de los 4 ojos (que constata que al menos dos personas deben supervisar una operación para que esta se considere segura).



(a) Plan de acción de un pushback (b) Personal involucrado en un pushback

Figura 1.1: Fases e involucrados en un *pushback*

1.1.1. Accidentes y peligros

Los accidentes ligados a la maniobra de *pushback* se clasifican en 4 categorías:

- **Categoría 1:** daño generado a/por la aeronave que está siendo empujada causado por/a otra aeronave u objeto estático de su entorno.
- **Categoría 2:** daño generado a/por la aeronave que está siendo empujada causado por/a el tractor.
- **Categoría 3:** daño generado a/por la aeronave que está siendo empujada causado por/a algún vehículo en tierra que no sea el tractor.
- **Categoría 4:** daños generados a los operarios durante la operación.

Solo en EEUU, ocurre un accidente de categoría 1 una media de 3 veces al año [1]. Los escenarios en los que esto ocurre se dividen en 5 situaciones:

- Escenario 1 : que el avión colisione con otra aeronave u otro objeto estacionario a su lado.
- Escenario 2 : que el avión colisione con otro que esté previamente posicionado en la calle de rodadura.
- Escenario 5 : que ocurra una colisión durante dos maniobras de *pushback* simultáneas.

Los escenarios 3 y 4 se corresponden con una situación similar a la del Escenario 1 pero con una disposición distinta de los aviones en el entorno.

Dentro de las alas es dónde se almacena el combustible de los aviones; así que esta categoría de accidentes, que involucra la posible colisión entre ellas, alberga un peligro considerable.

La categoría 2 de accidentes suele darse por un mal direccionamiento del tractor generado por el conductor o por una superficie resbaladiza. Es en este momento cuando se puede producir un efecto navaja, efecto V, de acodillamiento de remolque o *jackknife* que consiste en la adopción de un ángulo respecto al avión superior al adecuado. Esto provoca un estancamiento del sistema, desgaste del equipo, e incluso en los peores casos,

la necesidad de cambio del tren de aterrizaje

Los accidentes de categoría 3 se deben mayoritariamente a errores del conductor del tractor y se han visto reducidos en gran parte con la introducción de mapas en movimiento que muestran las posiciones de los aviones y vehículos en pista.

Finalmente los accidentes que involucran el daño a las personas tienen que ver con la pérdida de la conexión entre el avión y el tractor, que lleva al descontrol de la aeronave y al arrollamiento de las personas al rededor de la misma.

1.1.2. Motivos de los incidentes

Los motivos de los incidentes son varios y se suman para generar situaciones potencialmente peligrosas.

Por un lado, el espacio del que se dispone para realizar la maniobra de *pushback* es mínimo y no existen marcas en el suelo para esta operación. Como el conductor del tractor tiene una visibilidad muy reducida, tanto para divisar el trayecto que debe llevar hasta su situación final, como para ver ciertas partes del avión, la maniobra se convierte en una operación muy susceptible de errores que lleven a una colisión. De este modo se necesitan operarios de tierra que se pongan en riesgo para poder tener una visión completa de la situación.

Por otro lado, la comunicación es uno de los aspectos clave de la operación y por ello también de los principales causantes de accidentes.

Los errores de este tipo son puramente de carácter humano. Muchas veces los accidentes se causan por el fallo en estas comunicaciones y sus malas interpretaciones, por ejemplo por errores en las instrucciones de comienzo de la maniobra en los que el material o los trabajadores no están completamente preparados. Además, la jerarquía que hace que el equipo de *pushback* normalmente tenga menos rango de decisión que los pilotos y personal de la torre de control, crea una falta de intercambio de opiniones, donde deberían ser el conductor y sus compañeros los que tomen la decisión de parar la maniobra si así se requiere.

El encendido de los motores es también un motivo de peligro en el *pushback*. Para que el avión, cuando haya llegado a la calle de rodadura, pueda comenzar a avanzar por sí

solo hasta la pista de despegue, los motores deben haberse calentado previamente. En la práctica, los motores se encienden antes de esta operación o incluso durante la misma. La fuerza que ejercen los motores en contra del tractor hace que su esfuerzo sea mayor y, si el tractor tiene barra, esta pueda quebrarse o incluso que se pueda perder el control sobre el avión.

En muchos casos los errores se deben a simples despistes de los operarios debidos a la presión, cansancio acumulado o falta de precaución. También existen una multitud de casos que tienen que ver con las condiciones meteorológicas del momento, como puede ser el mal estado de la pista o la falta de visibilidad.

1.1.3. Target Level of Safety

El concepto de *Target Level of Safety* (TLS) refleja las consecuencias de fallo, filosofía de la seguridad, acceso a inspección y reparación y comportamiento de los elementos estructurales de un sistema.

Los datos históricos de los accidentes [1] durante la maniobra de *pushback* muestran que el nivel de seguridad adecuado para este tipo de operaciones no se alcanza. Concretamente, la probabilidad establecida por el TLS de A-SMGCS de que ocurra un accidente no debe sobrepasar el $1,0 \cdot 10^{-8}$. Los datos de tráfico aéreo provistos por los ATADS de 1991 a 2010 muestran un ratio de accidentes de $6,0 \cdot 10^{-7}$ y un $2,2 \cdot 10^{-8}$ de accidentes fatales.

El número de interacciones del personal de tierra en pista junto con las distancias mínimas con los objetos del entorno y la limitación visual que tiene el conductor del tractor, combinado con el estrés y cansancio de los trabajadores hacen pensar que se necesita algún sistema que control y prevención de colisiones para la maniobra de *pushback*.

1.1.4. Métodos de reducción de riesgo

Existen métodos de reducción de estos peligros a lo largo de muchos aeropuertos del mundo. En aeropuertos pequeños y con poco movimiento se ejecutan trayectorias de *pushback* que implican únicamente al conductor, ya que la visibilidad y espacio disponible en los mismos es más segura y se puede prescindir del personal de tierra. Hay muchos otros que ya están usando un tipo de tractor teleoperable ¹, que aporta la ventaja de que el

¹<https://www.mototok.com>

conductor no esté innecesariamente cerca del avión.

También existen multitud de estudios que abordan mecanismos de seguridad alrededor de esta maniobra: unos abordan trayectorias seguras para los aviones [2], otros el diseño óptimo del espacio en la plataforma ² [5] y otros nuevos sistemas de *pushback* y *taxiing* para aviones. Existe también un último tipo de estudios que tratan de reducir la cantidad de combustible que se usa en los movimientos por las plataformas y pistas implantando tractores con motores eléctricos ³. En este escenario se elimina el riesgo que supone el encendido de los motores durante la operación de *pushback*, dejando que estos se enciendan y calienten durante el desplazamiento hasta la pista de despegue ya que los motores no se necesitarían para los desplazamientos.

En [1] se propone la forma que deberá tener un sistema de prevención de riesgo de colisiones para un *pushback*. Esta propuesta incluye los siguientes aspectos que debe incluir este sistema:

1. Sistema de localización de la aeronave y los objetos a su alrededor en tiempo real.
2. Trayectoria de *pushback* ideal.
3. Previsión de colisiones en función a cada posible parámetro de la maniobra (velocidades, ángulos etc.).
4. Sistema de aviso en caso de conflicto.
5. Sistema de parada de emergencia en caso de colisión inminente.

En este documento se expone una solución para los puntos 2 y 3 de forma parcial, dónde se estudia un modelo de control que proporcione una trayectoria óptima de un sistema tractor-avión, que podrá usarse para analizar situaciones hipotéticas con distintos parámetros de la operación y determinar sus consecuencias.

1.2. Modelos de tractor existentes

Actualmente existen modelos de tractor de aviones en el mercado que se usan a diario en operaciones de tierra.

²<https://www.mitma.gob.es/el-ministerio/sala-de-prensa/noticias/mar-15062021-1650>

³<https://airportindustry-news.com/all-electric-aircraft-pushback-tug-in-operation-at-munich/>

El modelo *Mototok*⁴ es un tractor eléctrico empleado en maniobras de *pushback*. Este sistema está planteado como un tractor de remolque compacto que se puede controlar de forma remota, mediante un mando a distancia que maneja un operador autorizado. Este reduce gran parte de los riesgos humanos que conlleva esta maniobra, ya que solo necesita de un operador que puede mantenerse a una distancia prudente de la aeronave, aunque deba mantenerse relativamente cerca para mantener la consciencia de las distancias y movimientos que aplica al tractor para desplazar el avión. En este trabajo de fin de grado no se utiliza este modelo debido a que la configuración de su estructura no coincide con el modelo cinemático que se estudia.

*Taxibot*⁵ es un tractor semi-autónomo usado para maniobras de rodadura y *pushback*. Este sistema contribuye a la reducción de consumo de combustible y ruido ya que los motores de los aviones remolcados se mantienen apagados durante el trayecto hasta la pista de despegue. Es el piloto de la aeronave quien tiene el control del vehículo durante las maniobras de rodadura, lo que mejora la regulación y la seguridad de estas operaciones. Además, como puede emplearse para las maniobras de *pushback* y rodadura, elimina los cuellos de botella que se forman en las pistas de rodadura generados al desacoplar el tractor de *pushback* del tren de aterrizaje.

*WheelTug*⁶ es un proyecto enfocado en motorizar las ruedas del tren de aterrizaje de las aeronaves comerciales para poder así desplazar los aviones sin necesidad de usar sus motores ni tractores de remolque auxiliares. Este proyecto aun no se está usando en los aeropuertos, pero aporta diversas ventajas. Por un lado todas las correspondientes al ahorro de combustible que consumirían los motores del avión. Por otro lado, todas las problemáticas que conllevan los tractores, como son los peligros por la cercanía de los trabajadores o las dificultades en el desarrollo de los mismos.

1.3. Limitaciones

En la práctica se demuestra que los controladores de tipo *open-loop* no son adecuados para controlar robots móviles con remolques cuando están dirigidos hacia atrás, porque si se usan las entradas de control calculadas se manifiesta un comportamiento inestable y se produce el fenómeno denominado *jackknifing*. Para resolver el problema del *jackknifing* es necesario implementar un sistema de control de seguimiento de la trayectoria planificada (*tracking*).

⁴<https://www.mototok.com>

⁵<https://taxibot-international.com>

⁶<https://www.wheeltug.com/>

Por otro lado, la realización de un sistema completo tractor-avión involucraría un planteamiento robótico del problema, donde se deberían introducir sensores de cercanía en las alas, fuselaje y cola del avión, en el tractor y en el tren de aterrizaje. Con este set de sensores se podría diseñar un algoritmo *closed-loop* que reportase el *feedback* necesario para la evasión de colisiones. Además de esto, se deberían plantear mecanismos de acople y desacople del tractor al tren de aterrizaje y de parada de emergencia en situación de colisión inminente.

1.4. Objetivos

El objetivo final de este trabajo es determinar un control de *steering* capaz de crear trayectorias para un sistema constituido por un tractor y un remolque.

A razón de estos propósitos se establecen los siguientes sub-objetivos:

- Encontrar un modelo del sistema idóneo para ser controlado (*car-like with trailer*).
- Comprender los métodos de *steering* propuestos en la literatura científica para un *car-like with trailer* e implementarlos en simulación.
- Analizar los resultados de las simulaciones y escoger el método más adecuado para cada maniobra en base a las simulaciones.
- Aplicar el método elegido a un sistema específico para un avión y un tractor existentes en el mercado e implementarlo en simulación para demostrar que el método es aplicable al *steering* de un sistema tractor-avión.

Capítulo 2

Modelado y simulación de un sistema

2.1. Modelo

El primer paso para generar un control sobre un cuerpo consiste en encontrar un modelo matemático del mismo. Este modelo puede ser de tiempo continuo o discreto. El modelado de un sistema consiste en representar su evolución en el tiempo mediante una ecuación diferencial:

$$\dot{x}(t) = f(x(t), u(t)), \quad (2.1)$$

para un sistema con tiempo continuo t o:

$$x(k+1) = f(x(k), u(k)), \quad (2.2)$$

para un sistema de tiempo discreto k . En estas ecuaciones x representa el estado y u los inputs/entradas del sistema.

Existen distintos tipos de modelos de sistemas mecánicos en función a las características del sistema a controlar: modelos dinámicos y modelos cinemáticos ¹.

2.1.1. Modelado dinámico

Los modelos dinámicos se utilizan cuando el sistema mecánico a controlar tiene como inputs fuerzas y pares. Los modelos dinámicos de sistemas mecánicos se obtienen a partir

¹<https://www.ensta-bretagne.fr/jaulin/automoc.html>

de las leyes de la dinámica y contienen parámetros inerciales del sistema como la masa y el momento de inercia y variables que representan su posición, velocidad y aceleración. Es importante destacar que los parámetros inerciales de un sistema no son fáciles de determinar.

2.1.2. Modelado cinemático

Los modelos cinemáticos se utilizan cuando el sistema mecánico a controlar tiene como inputs velocidades. Los modelos cinemáticos de sistemas mecánicos se obtienen a partir de las leyes de la cinemática y por tanto no contienen parámetros inerciales del sistema. Estos modelos contienen únicamente parámetros geométricos y variables que representan su posición y velocidad.

Ambos modelos permiten especificar restricciones de movimiento que pueden ser holonómicas y no holonómicas.

2.1.3. Restricciones holonómicas

Las restricciones holonómicas son relaciones entre las variables del sistema que se pueden integrar y permiten eliminar alguna variable del modelo. Por ejemplo, para un robot móvil, es holonómica la restricción de seguir un camino preestablecido en el plano.

2.1.4. Restricciones no holonómicas

Las restricciones no holonómicas son relaciones entre las variables del sistema que no pueden integrarse. Por ejemplo, para las ruedas de un robot móvil, es no holonómica la restricción de moverse sin deslizar [7], es decir de moverse únicamente en la dirección de la rueda misma [Ver apartado 3.1].

2.1.5. Servo motores

Es posible controlar un sistema que admite solo fuerzas y pares como inputs de control en base a su modelo cinemático usando servo-motores. Un servo-motor es un motor con un sensor de posición, velocidad o aceleración y un circuito de control interno que se encarga de aplicar al sistema la velocidad deseada para alcanzar un *setpoint* concreto.

El sistema que se estudia en este TFG es un sistema no-holonómico cuyo modelo dinámico es muy complejo ². Así, usar un modelo dinámico, complicaría mucho la planificación de las maniobras de desplazamiento en tierra. Por tanto, finalmente se decide elegir un modelo cinemático del sistema.

2.2. Simulación

En esta sección se presenta la manera de simular la evolución de un sistema mecánico en un ordenador ya sea controlado o no.

En el caso de sistemas controlados, la simulación es importante para verificar que el sistema cumple con las especificaciones. Esta comprobación es una fase previa a la implementación del controlador del sistema real.

Para obtener la representación del cuerpo podemos por un lado mostrar cada variable de estado en una gráfica donde el eje de abscisas sea el tiempo, así se verá la evolución de cada una de ellas por cada instante temporal, sus puntos iniciales y también los finales. Asimismo, tener una representación visual del mecanismo puede ser más útil para comprobar los movimientos, distancias recorridas y posiciones tomadas en el espacio.

2.2.1. Simulaciones con animación gráfica

Para generar animaciones gráficas de la evolución de un sistema mecánico durante una simulación se necesitarán varios elementos detallados a continuación:

- **Sketch:** Es una representación gráfica simplificada del sistema que se está simulando. Esta es un conjunto de polígonos que representan los elementos del sistema mecánico. El modelo de cada uno de los polígonos es una matriz de 3 filas y n columnas que representa los n vértices del polígono en coordenadas homogéneas (ver siguiente apartado). Por cada columna se tiene un punto en el plano y cada vértice se une con el de sus posiciones anterior y posterior creando líneas poligonales.

En el problema que se está estudiando tendremos 4 sketches distintos para los 4 elementos del sistema. Estos son:

- el chasis del tractor junto con sus ruedas traseras,
- cada una de las ruedas delanteras por separado

²<https://doi.org/10.1017/S0263574700019858>

- el remolque/avión

Un ejemplo de estos es la representación del chasis del tractor:

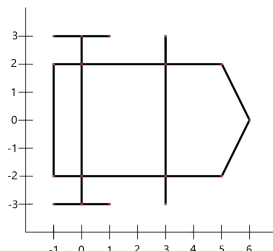


Figura 2.1: Sketch del chasis de un vehículo tractor de tipo *car-like*

- **Coordenadas homogéneas:** Para dibujar elementos en el plano se requiere hacer transformaciones (rotaciones, translaciones y escalados) sobre las matrices que conforman los sketches. Las coordenadas homogéneas tienen la ventaja de que las coordenadas de los puntos, incluidos los puntos en el infinito, se pueden representar mediante coordenadas finitas. De esta manera las coordenadas homogéneas permiten darles coordenadas a los puntos, ecuaciones a las curvas y expresiones algebraicas a las transformaciones proyectivas. Para transformar un punto a coordenadas homogéneas se le debe añadir un último elemento que represente el escalado. Así para \mathbb{R}^2 , el punto $(x, y)^T$ en el espacio cartesiano se corresponde con $(x, y, w)^T$ en coordenadas homogéneas.

Habitualmente le daremos a w el valor de 1, sin ningún escalado, y 0 cuando el punto se encuentra en el infinito. La capacidad de este tipo de coordenadas de expresar rectas paralelas que se cruzan el infinito y geometrías en el espacio tridimensional es esencial en campos como la visión artificial. Por ejemplo, son necesarias en el caso de la calibración de cámaras.

En las matrices que conforman los sketches, la tercera fila, y por tanto el tercer elemento de todos los vectores es igual a 1 para que los puntos estén expresados en coordenadas homogéneas. Posteriormente esto será necesario para poder utilizar las matrices de transformación homogéneas.

- **Matrices de transformación homogéneas:** a lo largo de la simulación se necesitará rotar y/o trasladar los vértices de los polígonos de forma que mantengan la

posición oportuna en cada instante. Para ello multiplicaremos el sketch/matriz por una matriz de transformación, que para \mathbb{R}^2 tiene la forma:

$$R = \begin{pmatrix} \cos(\theta) & -\sin(\theta) & t_x \\ \sin(\theta) & \cos(\theta) & t_y \\ 0 & 0 & w \end{pmatrix}$$

Esta matriz aplicará al elemento que lo multiplique una rotación de θ radianes, un desplazamiento de magnitud t_x en el eje x y t_y en el eje y . El vector $(x, y, 1)^T$ quedará transformado en $(u, v, w)^T$ aplicando la matriz R de la siguiente manera:

$$\begin{pmatrix} u \\ v \\ w \end{pmatrix} = \begin{pmatrix} \cos(\theta) & -\sin(\theta) & t_x \\ \sin(\theta) & \cos(\theta) & t_y \\ 0 & 0 & w \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$$

A la hora de aplicar distintas transformaciones a un mismo vector o matriz, podemos concatenar las transformaciones ordenando las matrices de transformación de derecha a izquierda. En el siguiente ejemplo se le aplica al vector $(x, y, 1)^T$ una rotación de γ radianes y una translación $t1_x$ y $t1_y$ y despues una rotación de ϕ radianes y una translación $t2_x$ y $t2_y$.

$$\begin{pmatrix} u \\ v \\ w \end{pmatrix} = \begin{pmatrix} \cos(\theta) & -\sin(\theta) & t2_x \\ \sin(\theta) & \cos(\theta) & t2_y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \cos(\phi) & -\sin(\phi) & t1_x \\ \sin(\phi) & \cos(\phi) & t1_y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$$

2.2.2. Algoritmo de simulación

La simulación de un sistema trata de mostrar la evolución en el tiempo del mismo, representada en la ecuación diferencial 2.1 anteriormente descrita. En cuanto al tiempo, es una variable continua, pero a la hora de utilizarlo en nuestro algoritmo, estará discretizado de modo que la duración total de cada maniobra se divida en cierta cantidad de pequeños intervalos de tamaño $\delta(t)$. Esta fracción temporal deberá ser suficientemente precisa desde el punto de vista numérico.

El algoritmo debe resolver esta ecuación diferencial para cada instante en el tiempo y una vez resuelta mostrar el estado del sistema. Por este motivo se necesitará un método de integración para resolver la ecuación y así obtener el estado x a partir del valor de \dot{x} en

cada momento. En la práctica, resolver esta ecuación significará que a partir del estado del sistema en el instante t_i se podrá encontrar el estado en t_{i+1} .

A continuación se presentan 2 métodos de integración capaces de estimar este estado: el método de Euler y el método de Runge-Kutta.

Método de Euler

Este es un procedimiento de integración numérica para resolver ecuaciones diferenciales a partir de un valor inicial dado, que se corresponde con un método de Taylor de orden 1. Este método supone un espacio temporal discreto dividido en intervalos de tamaño $\delta(t)$, como es nuestro caso, y demuestra que la ecuación de evolución de un sistema se puede aproximar de la siguiente forma:

$$\frac{x(t + \delta(t)) - x(t)}{\delta(t)} \approx f(x(t), u(t))$$

y por tanto

$$x(t + \delta(t)) \approx x(t) + f(x(t), u(t))\delta(t).$$

Método de Runge-Kutta

Este es otro método de integración numérica, que en este caso consigue estimar la solución con una exactitud mayor, ya que el error local del método es de orden 2. En las simulaciones será más preciso y por ende más útil que Euler. La diferencia fundamental que supone este método respecto al anterior es :

$$x(t + \delta(t)) \approx x(t) + \delta(t)\left[\frac{1}{4}f(x(t), u(t)) + \frac{3}{4}f\left(x(t) + \frac{2}{3}f(x(t), u(t)), u(t)\right)\right].$$

Implementación en MATLAB

Para implementar el algoritmo de simulación existen ciertas herramientas destinadas a la simulación de sistemas de control. En concreto se tiene *Simulink*, un entorno de diagramas de bloque que se utiliza para diseñar sistemas con modelos multidominio, simular antes de implementar en hardware y desplegar sin necesidad de escribir código. Este soft-

ware simplifica el desarrollo de las simulaciones y esconde los detalles mas complejos de los sistemas de control, además de hacer muy visual la composición de los mismos. Dentro de esta herramienta existe otra llamada *Simscape* que sirve para crear modelos de sistemas físicos y representarlos visualmente de una forma realista.

En este TFG no se utilizan estas tecnologías debido a varios motivos. Por un lado conviene tener bien identificadas y controladas todas las variables que componen el problema. Además, el sistema propuesto no es muy extenso ni complejo, por lo que la simplificación que conlleva *Simulink* llevaría a no comprender el conocimiento de sus elementos. Por último la representación de los modelos que se necesitan para este problema se satisface con las animaciones simples que proporcionan los sketches y matrices de rotación, por lo que el realismo que aporta *Simscape* se puede omitir. Por estas razones la implementación del algoritmo se aborda con scripts en MATLAB, que permiten hacer cálculos a bajo nivel, como puede ser la integración numérica de una ecuación, que sustituiría a un bloque integrador de *Simulink*.

Para todos los casos el algoritmo se compone de 4 ficheros distintos:

- *main.m*: es el programa principal dónde se describe la lógica de la simulación, se estima el estado siguiente en función del actual y se calculan los inputs para cada método.
- *init.m*: se usa para inicializar el canvas dónde se hará la animación del modelo, definiendo el tamaño y posición del sistema de referencia
- *draw.m*: función que se encarga de dibujar en el canvas la animación del modelo y por tanto dependerá de la forma de cada sistema. Contiene los sketches del modelo en cuestión y opciones para que la animación sea estroboscópica y de ciertos colores.
- *system.f.m*: función que representa $f(x(t), u(t))$ y que por tanto, a partir del estado actual del sistema $x(t)$ y los inputs $u(t)$, devuelve \dot{x} .

A continuación se expone una forma general del programa principal en la que podemos observar 2 bloques diferenciados: una parte secuencial de inicialización y un bucle para todo el tiempo que dura la maniobra.

La primera es la parte secuencial dónde se inicializan las variables del bucle principal y se calculan los valores de las constantes que conforman los inputs u , que en este caso hemos llamado $a0$ y $b0$ como ejemplo. La segunda parte es un bucle que itera el número de veces necesario para que el tiempo total t_F quede dividido en intervalos de tamaño dt .

Dentro del mismo se calcula y dibuja el estado del sistema:

1. - Primero se calcula el valor de u , cuya fórmula dependerá del método de control que se esté aplicando (en este caso se ha escogido una forma arbitraria sin ninguna validez de control real).
2. - Posteriormente se hace una sustitución de los inputs que transforman u en v . Esta transformación será común para todo problema que comparta el mismo sistema/modelo.
3. - Tras haber obtenido v se calcula el nuevo estado del sistema que resultará de haber aplicado los inputs llamando a *system-f.m*. Es aquí donde se aplica el método de Euler o Runge-Kutta.
4. - Se parará la ejecución del bucle durante dt segundos, lo que aparentará ser el paso del tiempo real. Esta sentencia se podrá modificar para que la ejecución sea más rápida o incluso se omitirá para obtener una solución inmediata.
5. - Finalmente se dibujará el sistema en el canvas llamando a *draw.m*. Solo se esbozará cada cierto número de iteraciones con la intención de reducir el coste computacional del programa y de conseguir una representación más útil y vistosa.

File main.m

```
% llamada a init.m: inicialización del canvas
init;
% Inicialización de variables
xI = [...];
xF = [...];
dt = 0.001;
...

% Cálculo de constantes necesarias para los inputs, que están en función
% de los estados iniciales finales y otros parámetros definidos encima
a0 = ...;
b0 = ...;
...

% Primera representación del sistema en su estado inicial (llamada a draw.m)
system_draw(x);

for t=0:dt:tF
    % cálculo de inputs
    u1=a0 * t;
    u2=b0 * t;
    u=[u1;u2];
    %sustitución de inputs
    v1 = c1 *u(1);
    v2 = c2* u(2);
    v = [v1;v2];

    % Métodos para obtener el estado siguiente:
    %x = x + system_f(x,v)*dt % Euler
    x = x + dt*(0.25*system_f(x,v)
        + 0.75*(system_f(x + dt*(2/3)*system_f(x,v),v))) % Runge-Kutta

    pause(dt);

    % Dibujo del sistema
    frame_counter = frame_counter+1;
    if frame_counter == 200
        system_draw(x);
        frame_counter = 0;
    end
end
end
```

Capítulo 3

Planificación de maniobras para un vehículo tractor

Para poder controlar un sistema completo compuesto por un vehículo tractor y un remolque, que en este caso es un sistema tractor-avión, comenzaremos buscando un control más sencillo sobre el tractor únicamente. Esto nos servirá como prueba para analizar los métodos de control con un sistema sencillo, pero también como parte de la maniobra ya que el tractor deberá posicionarse bajo el puente de aterrizaje del avión para elevar su estructura y así posteriormente desplazarlo.

3.1. Modelo *car-like*

Existen varios tipos de configuraciones para las ruedas de vehículos *car-like*. En este caso se ha escogido una configuración de Ackerman; esta es la que corresponde a los coches comunes, con dos ruedas traseras fijas con la misma dirección de giro y dos delanteras que rotan de forma paralela. Las ruedas delanteras por ende son las que dirigen el vehículo.

Otra configuración común es la que tiene ruedas diferenciales. En este caso la dirección del modelo viene dada por la combinación de las rotaciones de las ruedas de cada lado del vehículo. Las rotaciones se consiguen dando velocidades distintas a cada lado del vehículo, si se le dan velocidades opuestas el sistema podrá girar en torno a sí mismo girando las ruedas de la derecha hacia un lado, y las de la izquierda hacia el otro. Esta configuración es común en sistemas con dos ruedas y para modelos con 4 ruedas utiliza dos ruedas diferenciales y otras dos de cabeza giratoria/*caster wheels*.

Otro ejemplo de configuración es el llamado *crab steering* o configuración de cangrejo. En esta configuración las 4 ruedas giran con un mismo ángulo, consiguiendo generar

movimientos en todas las direcciones, incluso en paralelo.

El último ejemplo es el de la configuración de círculo. En este último la rotación del sistema depende del giro de las ruedas traseras frente al de las delanteras, que tendrá un ángulo opuesto entre ellas.

En las siguientes imágenes podemos ver estas configuraciones, que no son las únicas que existen:

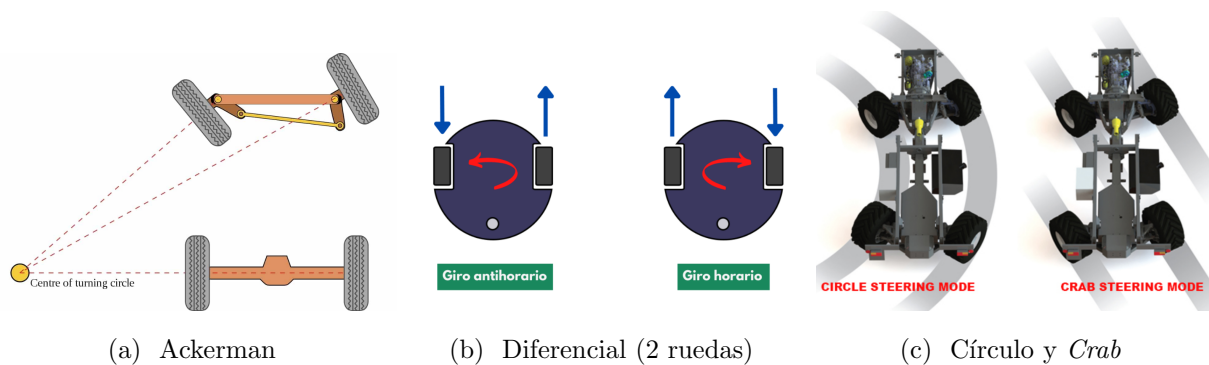
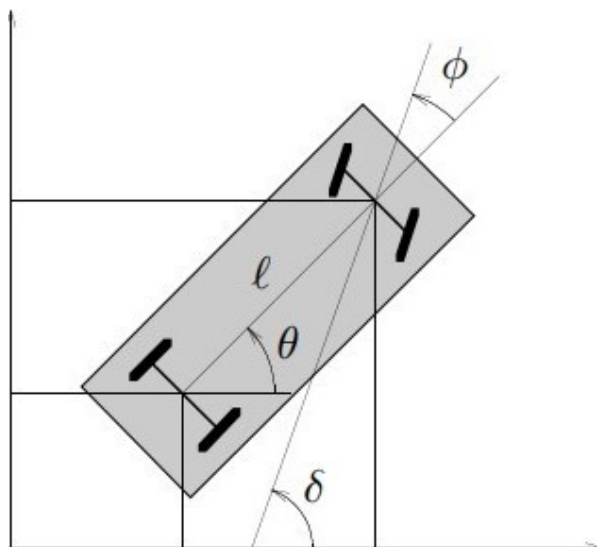


Figura 3.1: Distintas configuraciones de un vehículo *car-like*

Para generar el control deberemos empezar definiendo los modelos cinemáticos. La configuración escogida es la de Ackerman, que tiene varias distribuciones: de tracción trasera y de tracción delantera.

La configuración del tractor estará descrita por un vector q de 4 coordenadas: las posiciones x e y en el espacio cartesiano, que denotan el centro del eje de las ruedas traseras y por tanto del tractor; la orientación del tractor θ respecto al eje de coordenadas y la dirección de las ruedas motrices (*steering wheels*) respecto del tractor, que en este caso serán las delanteras, representadas por ϕ .

Figura 3.2: Modelo de un vehículo *car-like*

Para averiguar el modelo cinemático del tractor se debe cumplir la restricción no holonómica. Esta restricción se puede resumir en que las ruedas deben rodar pero no resbalar; lo que significa que la velocidad lineal de las ruedas solo puede recaer sobre la línea que forma cada rueda bajo su perímetro, es decir, no puede existir un movimiento lateral sobre ellas. En un vehículo *car-like*, al tener 2 ejes, se tienen 2 restricciones: una para las ruedas delanteras y otra para las traseras. Estas dos restricciones se presentan de la siguiente forma:

$$\begin{aligned}\dot{x}_f \sin \theta + \dot{y}_f \cos \theta + \dot{\phi} &= 0, \\ \dot{x} \sin \theta - \dot{y} \cos \theta &= 0.\end{aligned}$$

Donde x_f y y_f denotan las coordenadas del centro del eje de las ruedas delanteras, que definimos en relación a la posición del tractor como:

$$\begin{aligned}x_f &= x + l \cos \theta, \\ y_f &= y + l \sin \theta.\end{aligned}$$

Siendo l la distancia entre los dos ejes. Sustituyendo estas expresiones en las restricciones

anteriores se obtiene:

$$\dot{x} \sin(\theta + \phi) - \dot{y} \cos(\theta + \phi) - \dot{\theta} l \cos \phi = 0.$$

A partir de la ecuación previa podremos derivar el modelo cinemático de nuestro sistema *car-like* [7].

El modelo que se obtiene con tracción trasera será:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} \cos \theta \\ \sin \theta \\ \frac{\tan(\phi)}{l} \\ 0 \end{bmatrix} v_1 + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} v_2. \quad (3.1)$$

Si en cambio derivamos la expresión con tracción delantera:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} \cos \theta \cos \phi \\ \sin \theta \cos \phi \\ \frac{\sin(\phi)}{l} \\ 0 \end{bmatrix} v_1 + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} v_2, \quad (3.2)$$

donde v_1 es la velocidad lineal del centro del eje de las ruedas traseras y v_2 la velocidad angular o de *steering* de las ruedas directrices, que conformarán los inputs del sistema. Podemos observar que, en el modelo de tracción trasera, el $\cos(\phi)$ en la ecuación de $\dot{\theta}$ se puede anular y por tanto generar una singularidad en $\phi = \pm\pi/2$. Esta situación se corresponde con una posición de las ruedas delanteras perpendiculares a las traseras, pero debido a que el *steering* de las ruedas suele estar delimitado, en casos prácticos no es relevante.

Teniendo en cuenta la aplicación del sistema escogeremos el modelo de tracción trasera, ya que este tipo de vehículo tendrá mas agarre al suelo porque tiene que soportar un peso muy grande (el tren de aterrizaje de un avión). Además la complejidad del sistema asegura que es el más apropiado para este caso. Para los posteriores cálculos utilizaremos

la siguiente manera de escribir este modelo, con una forma reducida y más sencilla:

$$\dot{x} = \begin{bmatrix} \cos \theta v_1 \\ \sin \theta v_1 \\ \frac{\tan(\phi)}{l} v_1 \\ v_2 \end{bmatrix} \quad (3.3)$$

3.2. Chained form

Para poder diseñar el *steering* sobre modelos cinemáticos para robots o sistemas no holonómicos existen formas canónicas que simplifican y sistematizan su desarrollo. Una de las más útiles es la llamada *chained form* o forma encadenada, estudiada para casos similares [3], [4], [10], [7]. Su forma, en su versión de 2 inputs, abarca la mayoría de problemáticas para vehículos de este tipo.

Un sistema expresado en *chained form* se identifica con 2 dígitos (m,n): el primero (m) indica el número de inputs que tiene el sistema y el segundo (n) el número de variables que lo representan.

$$\begin{aligned}\dot{x}_1 &= u_1, \\ \dot{x}_2 &= u_2, \\ \dot{x}_3 &= x_2 u_1, \\ \dot{x}_4 &= x_3 u_1, \\ &\vdots \\ \dot{x}_n &= x_{n-1} u_1.\end{aligned}$$

Estos sistemas, aun siendo no lineales, tienen una estructura lineal subyacente que aporta una controlabilidad mayor que en otras representaciones. De hecho, la forma (2,n) ha sido demostrada como completamente controlable [8] mediante cálculos de *Lie Algebra*.

En nuestro problema estamos tratando un sistema (2,4), con 2 inputs que representan velocidades (lineal y angular) y un vector q de 4 variables, que resulta en:

$$\begin{aligned}\dot{x}_1 &= u_1, \\ \dot{x}_2 &= u_2, \\ \dot{x}_3 &= x_2 u_1, \\ \dot{x}_4 &= x_3 u_1.\end{aligned}$$

Para convertir el modelo 3.3 a esta forma deberemos hacer una sustitución del valor de los inputs. Transformaremos las velocidades v_1 y v_2 a los nuevos inputs característicos de un *chained form* u_1 y u_2 respectivamente. En consecuencia, también se hará un cambio de

variables que sustituyan a x , y , θ y ϕ por x_1, x_2, x_3 y x_4 . De esta manera podremos ver el sistema como una abstracción que contenga las variables reales pero estaremos trabajando con su *chained form*.

3.2.1. Modelo de tractor en *chained form*

Establecemos el cambio de inputs de la siguiente forma [7]:

$$\begin{aligned} v_1 &= \frac{u_1}{\cos \theta}, \\ v_2 &= \frac{-3 \sin(\theta) \sin^2(\phi) u_1}{l \cos^2(\theta)} + l \cos^3(\theta) \cos^2(\phi) u_2. \end{aligned} \quad (3.4)$$

Lo que resultará en el siguiente cambio de variables:

$$\begin{aligned} x_1 &= x, \\ x_2 &= \frac{\tan \phi}{l \cos^3 \theta}, \\ x_3 &= \tan \theta, \\ x_4 &= y. \end{aligned} \quad (3.5)$$

Para demostrar que este cambio de variables e inputs es correcto y se corresponde con el modelo que previamente habíamos escogido 3.3, primero sustituiremos los valores de los inputs dentro del vector \dot{q} :

$$\begin{aligned} \dot{x} &= \cos \theta v_1 = \frac{\cos \theta u_1}{\cos \theta} = u_1, \\ \dot{y} &= \sin \theta v_1 = \frac{\sin \theta u_1}{\cos \theta} = \tan \theta u_1, \\ \dot{\theta} &= \frac{\tan \phi v_1}{l} = \frac{\tan \phi u_1}{l \cos \theta}, \\ \dot{\phi} &= v_2 = \frac{-3 \sin(\theta) \sin^2(\phi) u_1}{l \cos^2(\theta)} + l \cos^3(\theta) \cos^2(\phi) u_2. \end{aligned}$$

Una vez sustituidos los inputs iniciales por los correspondientes a la *chained form*, se podrá calcular si los valores de las nuevas variables instanciadas anteriormente son

correctas. Para ello iremos derivando cada valor \dot{x}_n en *chained form* y despejando cada sustitución x_n .

Comenzamos la demostración asumiendo las sustituciones de la primera y última variable: $x_1 = x$ y $x_4 = y$.

$$\dot{x}_1 = \dot{x} = u_1, \quad (3.6)$$

$$\dot{x}_4 = \dot{y} = \tan \theta u_1. \quad (3.7)$$

De 3.7 podemos asegurar que como $\dot{x}_4 = x_3 u_1$, entonces $x_3 = \tan \theta$. Seguimos el procedimiento con \dot{x}_3 :

$$\dot{x}_3 = \frac{d}{dt} \tan \theta = \frac{1}{\cos^2 \theta} \dot{\theta} = \frac{1}{\cos^2 \theta} \frac{\tan \phi u_1}{l \cos \theta} = \frac{\tan \phi}{l \cos^3 \theta} u_1. \quad (3.8)$$

De la misma forma que en la variable anterior, al saber que $\dot{x}_3 = x_2 u_1$, se demuestra el cambio $x_2 = \frac{\tan \phi}{l \cos^3 \theta}$, concluyendo con la demostración.

3.3. Métodos de *steering* para un tractor de tipo *car-like* en *chained form*

En esta sección se presentan 3 métodos de *steering* que se pueden aplicar a sistemas representados en *chained form*.

3.3.1. Método 1: *Asymptotic Sinusoidal Inputs*

El método de *steering Sinusoidal Inputs* [8] se aplica a modelos muy sencillos o puramente en *chained form*. En él se usan entradas sinusoidales que tienen una relación integral entre sus frecuencias. Se puede deducir que es una buena opción para manejar un sistema expresado en *chained form* ya que también relaciona sus variables de forma integral.

El trasfondo matemático indica que si $u_1 = \sin(t)$ y $u_2 = \cos(kt)$, entonces $\dot{x}_1 = u_1$ tendrá componentes en frecuencias $k - 1$ y \dot{x}_2 las tendrá en $k - 2 \dots$. Por tanto cada $\dot{x}_k \neq \dot{x}_2$ tendrá alguna componente a frecuencia 0 y así se obtendrá movimiento en x_k mientras que las variables con $n < k$ se mantendrán invariantes. Por otro lado las variables dependientes del segundo input tendrán la forma $nk \pm 1$ lo que hará que para $k > 1$ no haya movimiento relativo.

Dado un sistema de n estados en *chained form*, se puede observar como x_1 y x_2 pueden ser posicionados (*steered*) en sus configuraciones finales usando inputs constantes dentro de un periodo T (ya que estas variables se corresponden con los propios inputs del sistema). Esto hará que los estados posteriores (x_3, \dots, x_n) se muevan en consecuencia. Si se establecen los inputs:

$$\begin{aligned} u_1 &= \alpha \sin(\omega t), \\ u_2 &= \beta \cos(\omega t), \end{aligned} \tag{3.9}$$

en x_3 se generará un movimiento durante $T = \frac{2\pi}{\omega}$ de la forma:

$$\begin{aligned} x_1(T) &= x_1(0), \\ x_2(T) &= x_2(0), \\ x_3(T) &= x_3(0) + \frac{\alpha\beta}{2\omega}. \end{aligned}$$

En las componentes posteriores (x_4, \dots, x_n) se generará un movimiento análogo.

En este método se propone dirigir x_1 y x_2 con inputs constantes hasta sus configuraciones finales, y una vez allí, con inputs sinusoidales 3.9, dirigir x_3 hacia su posición final, manteniendo x_1 y x_2 inalterados. Después se irá destinando cada componente hacia su estado final con esta clase de inputs, hasta llegar a los n niveles que componen el sistema encadenado.

En [9] se instancian ciertos problemas para este método:

1. Al haber relaciones integrales entre las frecuencias usadas, pueden aparecer interferencias en forma de DC-offset (error de desplazamiento de la señal).
2. Por otro lado hay que tener en cuenta que, debido a el encadenamiento de las variables, si nos equivocamos en el valor de cualquier x_k los valores de las variables anteriores $(x_{k-1}, x_{k-2}, \dots)$ se mantendrán sin alterar, pero las que le suceden $(x_{k+1}, x_{k+2}, \dots, x_n)$ variarán en conformidad y el error será acumulativo.
3. Finalmente este método se compone de muchos pasos, uno por cada componente del sistema, lo que supone un trabajo tedioso para el que lo estudie y/o calcule.

En [9] se propone un método análogo al anterior pero que resume el cálculo de todas las frecuencias reuniéndolas en u_2 .

Para ello la forma general de los inputs tendrá que ser:

$$\begin{aligned} u_1 &= a_0 + a_1 \sin(\omega t), \\ u_2 &= b_0 + b_1 \cos(\omega t) + b_2 \cos(2\omega t) + \dots + b_{n-2} \cos((n-2)\omega t). \end{aligned} \quad (3.10)$$

Como el modelo que se trata de controlar tiene $n=4$ variables, los inputs tendrán la forma:

$$\begin{aligned} u_1 &= a_0 + a_1 \sin(\omega t), \\ u_2 &= b_0 + b_1 \cos(\omega t) + b_2 \cos(2\omega t). \end{aligned} \quad (3.11)$$

Entonces, para hallar $u_1(t)$ y $u_2(t)$ bastará con calcular los valores de las constantes a_0, a_1, b_0, b_1, b_2 .

La integral de cada \dot{x}_n en *chained form* representa el desplazamiento de dicha variable (x_n) entre sus extremos. Para poder despejar los parámetros a_0, a_1, b_0, b_1, b_2 deberemos conformar un sistema de ecuaciones de la siguiente manera:

$$x_n^{t_F} = x_n^{t_I} + \int_{t_I}^{t_F} \dot{x}_n dt. \quad (3.12)$$

Por tanto, para deducir los inputs bastará con conocer 2 estados de la maniobra: el estado inicial del sistema $x(0) = x^0$ y el final $x(T) = x^{t_F}$. Si se establece que el tiempo de la maniobra sea $T = \frac{2\pi}{\omega}$, tras integrar las n ecuaciones 3.12, todos los elementos sinusoidales se convertirán en 0 ó 1 y así se podrán evaluar las constantes: a_0, b_0, b_1, b_2 . No obstante al tener n ecuaciones y $n + 1$ incógnitas se deberá establecer arbitrariamente una de ellas. En este caso se escogerá a_1 y deberá ser no nula.

Se asume que t_I es el principio de la maniobra y por tanto igual a 0. El instante final es el tiempo en el cual se completa la maniobra, y por tanto el periodo T . De esta manera, tras resolver las integrales definidas se harán las sustituciones:

$$\begin{aligned} t_I &= 0, \\ t_F &= T, \\ w &= \frac{2\pi}{T} = \frac{2\pi}{t_F}. \end{aligned} \quad (3.13)$$

Aplicación del método

Se comenzarán integrando los valores de nuestro sistema en *chained form* en función a la nueva forma 3.11 de los inputs. Para x_1 :

$$\begin{aligned} x_1(t_F) &= x_1(t_I) + \int_{t_I}^{t_F} u_1 dt \\ &= x_1(t_I) + a_0 t \Big|_{t_I}^{t_F} - \frac{a_1}{\omega} \cos(\omega t) \Big|_{t_I}^{t_F} \\ &= x_1(t_I) + a_0(t_F - t_I) - \frac{a_1}{\omega} (\cos(\omega t_F) - \cos(\omega t_I)) \end{aligned}$$

Tras hacer las sustituciones 3.13:

$$\begin{aligned} x_1(t_F) &= x_1(t_I) + a_0(t_F - 0) - \frac{a_1}{\omega}(1 - 1) \\ &= x_1(t_I) + a_0 t_F \end{aligned}$$

La segunda componente x_2 :

$$\begin{aligned}
x_2(t_F) &= x_2(t_I) + \int_{t_I}^{t_F} u_2 dt \\
&= x_2(t_I) + b_0 t \Big|_{t_I}^{t_F} + \frac{b_1}{\omega} \sin(\omega t) \Big|_{t_I}^{t_F} + \frac{b_2}{2\omega} \sin(2\omega t) \Big|_{t_I}^{t_F} \\
&= x_2(t_I) + b_0(t_F - t_I) + \frac{b_1}{\omega} (\sin(\omega t_F) - \sin(\omega t_I)) + \frac{b_2}{2\omega} (\sin(2\omega t_F) - \sin(2\omega t_I))
\end{aligned}$$

Y tras sustituir con 3.13 se obtiene:

$$\begin{aligned}
x_2(t_F) &= x_2(t_I) + b_0(t_F - 0) + \frac{b_1}{\omega}(0 - 0) + \frac{b_2}{2\omega}(0 - 0) \\
&= x_2(t_I) + b_0 t_F
\end{aligned}$$

Con la tercera componente:

$$\begin{aligned}
x_3(t_F) &= x_3(t_I) + \int_{t_I}^{t_F} x_2 u_1 dt \\
&= x_3(t_I) + \int_{t_I}^{t_F} (x_2(t_I) + b_0 t + \frac{b_1}{\omega} \sin(\omega t) + \frac{b_2}{2\omega} \sin(2\omega t))(a_0 + a_1 \sin(\omega t)) dt \\
&= x_3(t_I) + a_0 x_2(t_I) t \Big|_{t_I}^{t_F} - \frac{a_1 x_2(t_I)}{\omega} \cos(\omega t) \Big|_{t_I}^{t_F} + \frac{a_0 b_0}{2} t^2 \Big|_{t_I}^{t_F} + \frac{a_1 b_0}{\omega^2} (\sin(\omega t) \Big|_{t_I}^{t_F} \\
&\quad - \omega t \Big|_{t_I}^{t_F} \cos(\omega t) \Big|_{t_I}^{t_F}) - \frac{a_0 b_1}{\omega^2} \cos(\omega t) \Big|_{t_I}^{t_F} + \frac{a_1 b_1}{4\omega^2} (2\omega t \Big|_{t_I}^{t_F} - \sin(2\omega t) \Big|_{t_I}^{t_F}) \\
&\quad - \frac{a_0 b_2}{4\omega^2} \cos(2\omega t) \Big|_{t_I}^{t_F} + \frac{a_1 b_2}{3\omega^2} \sin^3(\omega t) \Big|_{t_I}^{t_F} \\
&= x_3(t_I) + a_0 x_2(t_I) (t_F - t_I) - \frac{a_1 x_2(t_I)}{\omega} (\cos(\omega t_F) - \cos(\omega t_I)) + \frac{a_0 b_0}{2} (t_F^2 - t_I^2) \\
&\quad + \frac{a_1 b_0}{\omega^2} ((\sin(\omega t_F) - \sin(\omega t_I)) - \omega(t_F - t_I)(\cos(\omega t_F) - \cos(\omega t_I))) \\
&\quad - \frac{a_0 b_1}{\omega^2} (\cos(\omega t_F) - \cos(\omega t_I)) + \frac{a_1 b_1}{4\omega^2} (2\omega(t_F - t_I) - (\sin(2\omega t_F) - \sin(2\omega t_I))) \\
&\quad - \frac{a_0 b_2}{4\omega^2} (\cos(2\omega t_F) - \cos(2\omega t_I)) + \frac{a_1 b_2}{3\omega^2} (\sin^3(\omega t_F) - \sin^3(\omega t_I))
\end{aligned}$$

$$\begin{aligned}
x_3(t_F) &= x_3(t_I) + a_0 x_2(t_I) t_F - \frac{a_1 x_2(t_I) t_F}{2\pi} (1 - 1) + \frac{a_0 b_0}{2} t_F^2 + \frac{a_1 b_0 t_F}{4\pi^2} (0 - 2\pi(1 - 1)) \\
&\quad - \frac{a_0 b_1 t_F}{4\pi^2} (1 - 1) + \frac{a_1 b_1 t_F}{16\pi^2} (4\pi - 0) - \frac{a_0 b_2 t_F}{16\pi^2} (1 - 1) + \frac{a_1 b_2}{12\pi^2} 0 \\
x_3(t_F) &= x_3(t_I) + a_0 x_2(t_I) t_F + \frac{a_0 b_0}{2} t_F^2 + \frac{a_1 b_1 t_F}{4\pi}
\end{aligned}$$

Por último x_4 :

$$\begin{aligned}
x_4(t_F) &= x_4(t_I) + \int_{t_I}^{t_F} u_1 x_3 dt \\
&= x_4(t_I) + \int_{t_I}^{t_F} (a_0 + a_1 \sin(\omega t))(x_3(t_I) + a_0 x_2(t_I) t \\
&\quad - \frac{a_1 x_2(t_I)}{\omega} \cos(\omega t) + \frac{a_0 b_0}{2} t^2 + \frac{a_1 b_0}{\omega^2} (\sin(\omega t) - \omega t \cos(\omega t)) \\
&\quad - \frac{a_0 b_1}{\omega^2} \cos(\omega t) + \frac{a_1 b_1}{4\omega^2} (2\omega t - \sin(2\omega t)) \\
&\quad - \frac{a_0 b_2}{4\omega^2} \cos(2\omega t) + \frac{a_1 b_2}{3\omega^2} \sin^3(\omega t)) dt \\
&= x_4(t_I) + \frac{a_0 x_3(t_I)}{2} t^2 \Big|_{t_I}^{t_F} \\
&\quad + \frac{a_0^2 x_2(t_I)}{2} t^2 \Big|_{t_I}^{t_F} - \frac{a_0 a_1 x_2(t_I)}{\omega^2} \sin(\omega t) \Big|_{t_I}^{t_F} + \frac{a_0^2 b_0}{6} t^3 \Big|_{t_I}^{t_F} - \frac{a_0 a_1 b_0}{\omega^3} \cos(\omega t) \Big|_{t_I}^{t_F} \\
&\quad - \frac{a_0 a_1 b_0}{\omega^3} (\omega t \Big|_{t_I}^{t_F} \sin(\omega t) \Big|_{t_I}^{t_F} + \cos(\omega t) \Big|_{t_I}^{t_F}) + \frac{a_0^2 b_1}{\omega^2} \sin(\omega t) \Big|_{t_I}^{t_F} \\
&\quad + \frac{a_0 a_1 b_1}{4\omega} t^2 \Big|_{t_I}^{t_F} - \frac{a_0 a_1 b_1}{8\omega^3} \cos(2\omega t) \Big|_{t_I}^{t_F} - \frac{a_0^2 b_2}{8\omega^3} \sin(2\omega t) \Big|_{t_I}^{t_F} \\
&\quad + \frac{a_0 a_1 b_2}{36\omega^3} (\cos(3\omega t) \Big|_{t_I}^{t_F} - 9 \cos(\omega t) \Big|_{t_I}^{t_F}) - \frac{a_1 x_3(t_I)}{\omega} \cos(\omega t) \Big|_{t_I}^{t_F} \\
&\quad + \frac{a_0 a_1 x_2(t_I)}{\omega^2} (\sin(\omega t) \Big|_{t_I}^{t_F} - \omega t \Big|_{t_I}^{t_F} \cos(\omega t) \Big|_{t_I}^{t_F}) - \frac{a_1^2 x_2(t_I)}{2\omega^2} \cos^2(\omega t) \Big|_{t_I}^{t_F} \\
&\quad + \frac{a_0 a_1 b_0}{2\omega^3} (2\omega t \Big|_{t_I}^{t_F} \sin(\omega t) \Big|_{t_I}^{t_F} + 2 \cos(\omega t) \Big|_{t_I}^{t_F} - \omega^2 t^2 \Big|_{t_I}^{t_F} \cos(\omega t) \Big|_{t_I}^{t_F}) \\
&\quad + \frac{a_1^2 b_0}{4\omega^3} (2\omega t \Big|_{t_I}^{t_F} - \sin(2\omega t) \Big|_{t_I}^{t_F}) - \frac{a_1^2 b_0}{8\omega^3} (\sin(2\omega t) \Big|_{t_I}^{t_F} - 2\omega t \Big|_{t_I}^{t_F} \cos(\omega t) \Big|_{t_I}^{t_F}) \\
&\quad - \frac{a_0 a_1 b_1}{2\omega^3} \cos^2(\omega t) \Big|_{t_I}^{t_F} + \frac{a_1^2 b_1}{4\omega^2} (\sin(\omega t) \Big|_{t_I}^{t_F} - \omega t \Big|_{t_I}^{t_F} \cos(\omega t) \Big|_{t_I}^{t_F}) \\
&\quad - \frac{a_1^2 b_1}{6\omega^3} \sin^3(\omega t) \Big|_{t_I}^{t_F} + \frac{a_0 a_1 b_2}{24\omega^3} (\cos(3\omega t) \Big|_{t_I}^{t_F} - 3 \cos(\omega t) \Big|_{t_I}^{t_F}) \\
&\quad - \frac{a_1^2 b_2}{96\omega^2} (\sin(2\omega t) \Big|_{t_I}^{t_F} + \sin(4\omega t) \Big|_{t_I}^{t_F} + 12\omega t \Big|_{t_I}^{t_F})
\end{aligned}$$

$$\begin{aligned}
x_4(t_F) = & x_4(t_I) + \frac{a_0 x_3(t_I)}{2} (t_F^2 - t_I^2) + \frac{a_0^2 x_2(t_I)}{2} (t_F^2 - t_I^2) - \frac{a_0 a_1 x_2(t_I)}{\omega^2} (\sin(\omega t_F) \\
& - \sin(\omega t_I)) + \frac{a_0^2 b_0}{6} (t_F^3 - t_I^3) - \frac{a_0 a_1 b_0}{\omega^3} \\
& (\cos(\omega t_F) - \cos(\omega t_I)) - \frac{a_0 a_1 b_0}{\omega^3} \\
& (\omega(t_F - t_I)(\sin(\omega t_F) - \sin(\omega t_I)) + (\cos(\omega t_F) - \cos(\omega t_I))) \\
& + \frac{a_0^2 b_1}{\omega^2} (\sin(\omega t_F) - \sin(\omega t_I)) + \frac{a_0 a_1 b_1}{4\omega} (t_F^2 - t_I^2) \\
& - \frac{a_0 a_1 b_1}{8\omega^3} (\cos(2\omega t_F) - \cos(2\omega t_I)) - \frac{a_0^2 b_2}{8\omega^3} (\sin(2\omega t_F) - \sin(2\omega t_I)) \\
& + \frac{a_0 a_1 b_2}{36\omega^3} ((\cos(3\omega t_F) - \cos(3\omega t_I)) - 9(\cos(\omega t_F) - \cos(\omega t_I))) \\
& - \frac{a_1 x_3(t_I)}{\omega} (\cos(\omega t_F) - \cos(\omega t_I)) + \frac{a_0 a_1 x_2(t_I)}{\omega^2} \\
& ((\sin(\omega t_F) - \sin(\omega t_I)) - \omega(t_F - t_I)(\cos(\omega t_F) - \cos(\omega t_I))) \\
& - \frac{a_1^2 x_2(t_I)}{2\omega^2} (\cos^2(\omega t_F) - \cos^2(\omega t_I)) + \frac{a_0 a_1 b_0}{2\omega^3} (2\omega(t_F - t_I)(\sin(\omega t_F) \\
& - \sin(\omega t_I)) + 2(\cos(\omega t_F) - \cos(\omega t_I)) - \omega^2(t_F^2 - t_I^2)(\cos(\omega t_F) \\
& - \cos(\omega t_I))) + \frac{a_1^2 b_0}{4\omega^3} (2\omega(t_F - t_I) - (\sin(2\omega t_F) - \sin(2\omega t_I))) \\
& - \frac{a_1^2 b_0}{8\omega^3} ((\sin(2\omega t_F) - \sin(2\omega t_I)) - 2\omega(t_F - t_I)(\cos(\omega t_F) - \cos(\omega t_I))) \\
& - \frac{a_0 a_1 b_1}{2\omega^3} (\cos^2(\omega t_F) - \cos^2(\omega t_I)) + \frac{a_1^2 b_1}{4\omega^2} ((\sin(\omega t_F) \\
& - \sin(\omega t_I)) - \omega(t_F - t_I)(\cos(\omega t_F) - \cos(\omega t_I))) - \frac{a_1^2 b_1}{6\omega^3} \\
& (\sin^3(\omega t_F) - \sin^3(\omega t_I)) + \frac{a_0 a_1 b_2}{24\omega^3} ((\cos(3\omega t_F) - \cos(3\omega t_I)) \\
& - 3(\cos(\omega t_F) - \cos(\omega t_I))) - \frac{a_1^2 b_2}{96\omega^2} ((\sin(2\omega t_F) - \sin(2\omega t_I)) \\
& + (\sin(4\omega t_F) - \sin(4\omega t_I)) + 12\omega(t_F - t_I))
\end{aligned}$$

Finalmente, sustituimos con 3.13:

$$\begin{aligned}
x_4(t_F) = & x_4(t_I) + \frac{a_0 x_3(t_I)}{2} t_F^2 + \frac{a_0^2 x_2(t_I)}{2} t_F^2 \\
& + \frac{a_0^2 b_0}{6} t_F^3 + \frac{a_0 a_1 b_1}{8\pi} t_F^3 + \frac{a_1^2 b_0}{8\pi^2} t_F - \frac{a_1^2 b_2}{16\pi} t_F
\end{aligned}$$

Para evaluar las constantes se ha establecido un sistema de ecuaciones en función a las posiciones iniciales $x_n(t_I)$ y finales $x_n(t_F)$ de cada variable y el tiempo que queremos que dure la maniobra (t_F). Tras resolver este sistema se obtienen los parámetros que componen los inputs:

$$\begin{aligned}
a_0 &= \frac{x_1(t_F) - x_1(t_I)}{t_F} \\
b_0 &= \frac{x_2(t_F) - x_2(t_I)}{t_F} \\
b_1 &= \frac{2}{a_1 t_F^2} (2\pi x_3(t_F) - 2\pi x_3(t_I) - \pi x_1(t_F)x_2(t_F) - \pi x_1(t_I)x_2(t_I) \\
&\quad + \pi x_2(t_F)x_1(t_I) + \pi x_1(t_I)x_2(t_I) + a_1 t_F x_2(t_F) - a_1 t_F x_2(t_I)) \\
b_2 &= -(96x_4(t_F)\pi^2 - 96x_4(t_I)\pi^2 + 8x_1(t_F)^2 x_2(t_F)\pi^2 + 8x_2(t_F)x_1(t_I)^2 \pi^2 \\
&\quad - 8x_1(t_F)^2 x_2(t_I)\pi^2 - 8x_1(t_I)^2 x_2(t_I)\pi^2 + 6a_1^2 t_F^2 x_2(t_F) - 6a_1^2 t_F^2 x_2(t_I) \\
&\quad - 48x_1(t_F)x_3(t_F)\pi^2 - 48x_1(t_F)x_3(t_I)\pi^2 + 48x_3(t_F)x_1(t_I)\pi^2 \\
&\quad + 48x_1(t_I)x_3(t_I)\pi^2 - 16x_1(t_F)x_2(t_F)x_1(t_I)\pi^2 + 16x_1(t_F)x_1(t_I)x_2(t_I)\pi^2 \\
&\quad + 48\pi a_1 t_F x_3(t_F) - 48\pi a_1 t_F x_3(t_I) - 24\pi a_1 t_F x_1(t_F)x_2(t_F) \\
&\quad + 24\pi a_1 t_F x_1(t_F)x_2(t_I) + 24\pi a_1 t_F x_2(t_F)x_1(t_I) - 24\pi a_1 t_F x_1(t_I)x_2(t_I)) \\
&\quad (-3a_1^2 t_F^3 + 6t_F x_1(t_F)^2 - 12t_F x_1(t_F)x_1(t_I) + 6t_F x_1(t_I)^2)
\end{aligned}$$

3.3.2. Método 2: *Piecewise Constant Inputs*

El segundo método de *steering* para sistemas en *chained form* es *Piecewise Constant Inputs*. Este método mantiene constante el valor de los inputs a lo largo de un cierto periodo de tiempo δ :

$$\begin{aligned} u_1(\tau) &= u_{1,1}, \\ u_2(\tau) &= u_{2,1}, \end{aligned}$$

con

$$\tau \in [0, \delta].$$

Si integramos las ecuaciones en *chained form* de la misma forma que en el método anterior 3.12, al haber establecido valores constantes para los inputs, estos salen naturalmente de las integrales. Aplicado al intervalo $[0, \delta]$ se obtiene la forma:

$$\begin{aligned} x_1(\delta) &= x_1(0) + u_{1,1}\delta \\ x_2(\delta) &= x_2(0) + u_{2,1}\delta \\ x_3(\delta) &= x_3(0) + x_2(0)u_{1,1}\delta + u_{1,1}u_{2,1}\frac{\delta^2}{2} \\ &\vdots \\ x_n(\delta) &= x_n(0) + x_{n-1}(0)u_{1,1}\delta + \dots + u_{1,1}^{n-2}u_{2,1}\frac{\delta^{n-1}}{n-1} \end{aligned} \tag{3.14}$$

Que depende de $x(0)$, $u_{1,1}$ y $u_{2,1}$.

Si ahora consideramos el siguiente intervalo de tiempo $\tau \in [\delta, 2\delta]$ y su nuevo par de inputs constantes:

$$\begin{aligned} u_1(\tau) &= u_{1,2} \\ u_2(\tau) &= u_{2,2} \end{aligned}$$

obtendremos las ecuaciones para $x(2\delta)$ en función a $x(\delta)$, $u_{1,2}$ y $u_{2,2}$. Ahora, si se sustituyen los valores de $x(\delta)$ por las ecuaciones 3.14, se obtiene la expresión de $x(2\delta)$ en función a

$x(0)$, $u_{1,1}$, $u_{2,1}$, $u_{1,2}$ y $u_{2,2}$.

Este proceso de cálculo y sustitución de las variables se puede ampliar las veces que sea necesario, obteniendo en cada paso la expresión del estado $x(k\delta)$ en función al estado inicial y los inputs constantes de cada tramo.

En el estudio de un problema de planificación de trayectorias se escoge un u_1 constante, o lo que es lo mismo, una velocidad lineal constante para todo el periodo T de la maniobra. El valor de δ se puede escoger de forma arbitraria, pero para conseguir el mejor resultado trataremos de buscar el intervalo δ mas grande posible. En caso contrario los valores de u deberán ser mayores para ejecutar la misma trayectoria. Por tanto, manteniendo u_1 constante, se deberá iterar $n - 1$ veces para poder resolver los n parámetros: $u_{1,1}$, $u_{2,1}$, $u_{2,2}$, ..., $u_{2,n-1}$.

Se establecerá que $x(0) = x(t_I)$ y $x(T) = x(t_F)$ y, a razón de que u_1 sea constante para todo T , su expresión será:

$$u_1 = \frac{x_1(t_F) - x_1(t_I)}{T} \quad (3.15)$$

Las $n - 1$ ecuaciones restantes pueden ser resueltas para $u_{1,1}$, $u_{2,1}$, ..., $u_{2,n-1}$ con facilidad.

Observando la forma de un sistema en *chained form* y la del input constante u_1 3.15 se puede apreciar que $x_1(t_F)$ y $x_1(t_I)$ deben ser distintos. En caso contrario todas las componentes que contienen u_1 se anularán y solo será útil x_2 . Podemos deducir que este método no podrá devolver una trayectoria válida si solo hay movimiento en la dirección de las ruedas directrices. Una manera de resolver un caso en el que queramos que la primera coordenada comience y termine en el mismo lugar será realizar 2 trayectorias, una que lleve el sistema a un estado intermedio con x_1 s distintos y otra que lo lleve de este al estado final. Un caso que refleja esta circunstancia es el de un aparcamiento en paralelo dónde la coordenada x sea la misma tanto en el estado inicial como en el final.

Aplicación del método

Para el caso que se está tratando, con $n = 4$ variables, se establecerán $n - 1 = 3$ intervalos de tamaño δ :

$$\begin{aligned}\text{tramo 1} &= \tau \in [0, \delta], \\ \text{tramo 2} &= \tau \in [\delta, 2\delta], \\ \text{tramo 3} &= \tau \in [2\delta, 3\delta] \equiv \tau \in [2\delta, T].\end{aligned}$$

Dado que la forma de cada tramo depende únicamente del intervalo del que se trate, se puede calcular un tramo genérico dentro de un intervalo $\tau \in [a, b]$ que resuma todos:

$$\begin{aligned}x_1(b) &= x_1(a) + \int_a^b u_1(\tau) dt \\ &= x_1(a) + u_1(\tau)\tau \Big|_a^b \\ x_2(b) &= x_2(a) + \int_a^b u_2(\tau) dt \\ &= x_2(a) + u_2(\tau)\tau \Big|_a^b \\ x_3(b) &= x_3(a) + \int_a^b u_1(\tau)x_2(b) dt \\ &= x_3(a) \int_a^b u_1(\tau)(x_2(a)u_2(\tau)\tau) dt \\ &= x_3(a) + \int_a^b u_1(\tau)x_2(a) dt + \int_a^b u_1(\tau)u_2(\tau)\tau dt \\ &= x_3(a) + x_2(a)u_1(\tau)\tau \Big|_a^b + \frac{u_1(\tau)^2 u_2(\tau)\tau^2 \Big|_a^b}{2} \\ x_4(b) &= x_4(a) + \int_a^b u_1(\tau)x_3(b) dt \\ &= x_4(a) + \int_a^b u_1(\tau)(x_3(a) + x_2(a)u_1(\tau)\tau + \frac{u_1(\tau)^2 u_2(\tau)\tau^2}{2}) dt \\ &= x_4(a) + \int_a^b u_1(\tau)x_3(a) dt + \int_a^b u_1(\tau)(x_2(a)u_1(\tau)\tau) dt + \\ &\quad \int_a^b u_1(\tau)(\frac{u_1(\tau)^2 u_2(\tau)\tau^2}{2}) dt \\ &= x_4(a) + x_3(a)u_1(\tau)\tau \Big|_a^b + \frac{x_2(a)u_1(\tau)^2 \tau^2 \Big|_a^b}{2} + \frac{u_1(\tau)^2 u_2(\tau)\tau^3 \Big|_a^b}{6}\end{aligned}$$

Por tanto, después de sustituir a y b por cada respectivo intervalo, se obtienen las ecuaciones para cada tramo:

Tramo 1 : $\tau \in [0, \delta)$

$$\begin{aligned} x_1(\delta_-) &= x_1(0) + u_1(\tau)\delta_- \\ x_2(\delta_-) &= x_2(0) + u_2(\tau)\delta_- \\ x_3(\delta_-) &= x_3(0) + x_2(0)u_1(\tau)\delta_- + \frac{u_1(\tau)^2 u_2(\tau)\delta_-^2}{2} \\ x_4(\delta_-) &= x_4(0) + x_3(0)u_1(\tau)\delta_- + \frac{x_2(0)u_1(\tau)^2 \delta_-^2}{2} + \frac{u_1(\tau)^2 u_2(\tau)\delta_-^3}{6} \end{aligned}$$

Tramo 2: $\tau \in [\delta, 2\delta)$

$$\begin{aligned} x_1(2\delta_-) &= x_1(\delta_+) + u_1(\tau)(2\delta_- - \delta_+) \\ x_2(2\delta_-) &= x_2(\delta_+) + u_2(\tau)(2\delta_- - \delta_+) \\ x_3(2\delta_-) &= x_3(\delta_+) + x_2(\delta_+)u_1(\tau)(2\delta_- - \delta_+) + \frac{u_1(\tau)^2 u_2(\tau)(2\delta_- - \delta_+)^2}{2} \\ x_4(2\delta_-) &= x_4(\delta_+) + x_3(\delta_+)u_1(\tau)(2\delta_- - \delta_+) \\ &\quad + \frac{x_2(\delta_+)u_1(\tau)^2(2\delta_- - \delta_+)^2}{2} + \frac{u_1(\tau)^2 u_2(\tau)(2\delta_- - \delta_+)^3}{6} \end{aligned}$$

Tramo 3: $\tau \in [2\delta, 3\delta]$

$$\begin{aligned} x_1(3\delta) &= x_1(2\delta_+) + u_1(\tau)(3\delta - 2\delta_-) \\ x_2(3\delta) &= x_2(2\delta_+) + u_2(\tau)(3\delta - 2\delta_-) \\ x_3(3\delta) &= x_3(2\delta_+) + x_2(2\delta_+)u_1(\tau)(3\delta - 2\delta_-) + \frac{u_1(\tau)^2 u_2(\tau)(3\delta - 2\delta_-)^2}{2} \\ x_4(3\delta) &= x_4(0) + x_3(2\delta_+)u_1(\tau)(3\delta - 2\delta_-) + \frac{x_2(2\delta_+)u_1(\tau)^2(3\delta - 2\delta_-)^2}{2} \\ &\quad + \frac{u_1(\tau)^2 u_2(\tau)(3\delta - 2\delta_-)^3}{6} \end{aligned}$$

Para acabar con el cálculo de los inputs $u_{2,k}$ deberemos sustituir los valores de $x(\delta)$ del tramo 1 en el tramo 2 y los de $x(2\delta)$ en el tercer tramo, obteniendo así el intervalo

completo, que conformará el sistema de ecuaciones en función de δ , $T, x(0)$ y $x(T)$:

$$\begin{aligned}
x_1(T) &= x_1(0) + u_1(\tau)\delta + u_1(\tau)(2\delta - \delta_+) + u_1(\tau)(3\delta - 2\delta) \\
x_2(T) &= x_2(0) + u_{2,1}(\tau)\delta + u_{2,2}(\tau)(2\delta - \delta) + u_{2,3}(\tau)(3\delta - 2\delta) \\
x_3(T) &= x_3(0) + x_2(0)u_1(\tau)\delta + \frac{u_{2,1}u_1\delta}{2} + (x_2(0) + u_{2,1}\delta)u_1(2\delta - \delta) + \frac{u_1u_{2,2}(2\delta - \delta)}{2} \\
&\quad + x_2(0) + u_{2,1}\delta + u_{2,2}(2\delta - \delta)u_1(3\delta - 2\delta) + \frac{u_{2,3}u_1(3\delta - 2\delta)}{2} \\
x_4(T) &= x_4(0) + x_3(0)u_1\delta + \frac{x_2(0)u_1^2\delta^2}{2} + \frac{u_1^2u_{2,1}\delta^3}{6} + (x_3(0) + x_2(0)u_1\delta \\
&\quad + \frac{u_1u_{2,1}\delta}{2})u_1(2\delta - \delta) + \frac{x_2(0) + u_{2,1}\delta}{2}u_1^2(4\delta^2 - \delta^2) + \frac{u_1^2u_{2,2}(8\delta^3 - \delta^3)}{6} \\
&\quad + (x_3(0) + x_2(0)u_1\delta + \frac{u_{2,1}u_1\delta}{2} + (x_2(0) + u_{2,1}\delta)u_1(2\delta - \delta) \\
&\quad + \frac{u_1u_{2,2}(2\delta - \delta)}{2})u_1(3\delta - 2\delta) + (x_2(0) + u_{2,1}\delta + u_{2,2}(2\delta - \delta))\frac{u_1^2(9\delta^2 - 4\delta^2)}{2} \\
&\quad + \frac{u_1^2u_{2,3}(27\delta^3 - 8\delta^3)}{6}
\end{aligned}$$

Introduciendo este sistema en una herramienta de cálculo obtenemos el valor de los inputs constantes:

$$\begin{aligned}
u_1 &= \frac{x_1(t_F) - x_1(t_I)}{3\delta} \\
u_{2,1} &= \left(\frac{9}{\delta(x_1(t_F) - x_1(t_I))^2} \right) x_4(t_F) - 9x_4(t_I) - 6x_1(t_F)x_3(t_F) - 3x_1(t_F)x_3(t_I) + \\
&\quad 6x_3(t_F)x_1(t_I) + 3x_1(t_I)x_3(t_I) + x_1(t_F)^2x_2(t_F) + x_2(t_F)x_1(t_I)^2 - x_1(t_F)^2x_2(t_I) \\
&\quad - x_1(t_I)^2x_2(t_I) - 2x_1(t_F)x_2(t_F)x_1(t_I) + 2x_1(t_F)x_1(t_I)x_2(t_I) \\
u_{2,2} &= \left(-\frac{3}{\delta(x_1(t_F) - x_1(t_I))^2} \right) (6x_4(t_F) - 6x_4(t_I) - 5x_1(t_F)x_3(t_F) - x_1(t_F)x_3(t_I) + \\
&\quad 5x_3(t_F)x_1(t_I) + x_1(t_I)x_3(t_I) + x_1(t_F)^2x_2(t_F) + x_2(t_F)x_1(t_I)^2 \\
&\quad - 2x_1(t_F)x_2(t_F)x_1(t_I)) \\
u_{2,3} &= \left(\frac{3}{\delta x_1(t_F)^2 - 2\delta x_1(t_F)x_1(t_I) + \delta x_1(t_I)^2} \right) x_2(t_F)x_1(t_F)^2 - 6x_2(t_F)x_1(t_F)x_1(t_I) \\
&\quad - 9x_3(t_F)x_1(t_F) + 3x_2(t_F)x_1(t_I)^2 + 9x_3(t_F)x_1(t_I) + 9x_4(t_F) - 9x_4(t_I)
\end{aligned}$$

El mayor inconveniente que tiene este método recae en la discontinuidad de u_2 . Esto es porque estamos usando un modelo cinemático, donde los inputs son velocidades, y en la práctica, sobre un robot móvil, lo que se querrá introducir son aceleraciones/fuerzas. Estas aceleraciones se calculan derivando las velocidades de las que disponemos, y al ser constantes, no son diferenciables en sus discontinuidades.

3.3.3. Método 3: *Polynomial Inputs*

Por último, se presenta el método de *steering* de *Polynomial Inputs*, que presenta la ventaja que obteníamos en el método anterior con el input u_1 constante y añade la utilidad de un u_2 polinomial. Este cambio elimina el problema de la diferenciableidad que encontrábamos en el caso anterior.

Derivada de la forma constante de u_1 3.15, el tiempo total de la maniobra (T) viene determinado por el desplazamiento de la primera coordenada:

$$T = \frac{x_1(t_F) - x_1(t_I)}{u_1}.$$

Los inputs polinomiales que se proponen son:

$$\begin{aligned} u_1 &= b_1, \\ u_2 &= c_0 + c_1 t + c_2 t^2 + \dots + c_{n-2} t^{n-2}. \end{aligned} \tag{3.16}$$

Donde el valor de la constante b_1 estará en función de la duración de la maniobra y el desplazamiento de x , siguiendo la ecuación 3.15.

Del mismo modo que en los métodos anteriores se podrán calcular los valores c_i con la ecuación 3.12:

$$x_n^{t_F} = x_n^{t_I} + \int_{t_I}^{t_F} \dot{x}_n dt.$$

Ampliando esta ecuación para los inputs polinomiales, la forma general del sistema de

ecuaciones del que sacaremos los inputs es:

$$\begin{aligned}
 x_1(t_F) &= x_1(t_I) + t \\
 x_2(t_F) &= x_2(t_I) + c_0 t + \frac{c_1 t^2}{2} + \frac{c_{n-2} t^{n-1}}{n-1} \\
 &\vdots \\
 x_i(t_F) &= x_i(t_I) + \sum_{k=0}^{n-2} \frac{k! c_k^{i+k-1}}{(i+k-1)!} + \sum_{k=2}^{i-1} \frac{t^{i-k}}{(i-k)!} x_k(t_I)
 \end{aligned} \tag{3.17}$$

Aplicación del método

Con $n = 4$ variables, obtendremos el siguiente sistema dependiente de 3 parámetros: c_0, c_1 y c_2 :

$$\begin{aligned}
 x_1(t_F) &= x_1(t_I) + t \\
 x_2(t_F) &= x_2(t_I) + c_0 t + \frac{c_1 t^2}{2} + \frac{c_2 t^3}{3} \\
 x_3(t_F) &= x_3(t_I) + \frac{c_1^3}{6} + \frac{c_2^4}{12} + x_2(t_I) t \\
 x_4(t_F) &= x_4(t_I) + \frac{c_1^4}{24} + \frac{c_2^5}{60} + \frac{t^2}{2} x_2(t_I) + x_3(t_I) t
 \end{aligned}$$

El resultado del mismo es:

$$\begin{aligned}
 c_0 &= -\frac{3(20x_4(t_I) - 20x_4(t_F) - b_1^2 t_F^2 x_2(t_F) + 3b_1^2 t_F^2 x_2(t_I) + 8b_1 t_F x_3(t_F) + 12b_1 t_F x_3(t_I))}{b_1^2 t_F^3} \\
 c_1 &= \frac{2(30x_4(t_I) - 30x_4(t_F) - 2b_1^2 t_F^2 x_2(t_F) + 3b_1^2 t_F^2 x_2(t_I) + 14b_1 t_F x_3(t_F) + 16b_1 t_F x_3(t_I))}{b_1^2 t_F^4} \\
 c_2 &= -\frac{30(12x_4(t_I) - 12x_4(t_F) - b_1^2 t_F^2 x_2(t_F) + b_1^2 t_F^2 x_2(t_I) + 6b_1 t_F x_3(t_F) + 6b_1 t_F x_3(t_I))}{b_1^2 t_F^5}
 \end{aligned}$$

3.4. Resultados

Una vez se han calculado todos los valores de los inputs para cada método, es el momento de introducirlos en un programa MATLAB para simular su comportamiento y analizar los resultados. En general todos los algoritmos son análogos al presentado en el capítulo 2.2.2.

Los experimentos que se muestran a continuación analizan los métodos de planificación estudiados en los apartados anteriores en situaciones sin singularidades. Debido a la presencia de cosenos en los denominadores de las ecuaciones de estado en *chained form* 3.4 y 3.5, se tendrán singularidades que llevarán al sistema hacia comportamientos indeterminados. Para eludir las singularidades se tendrá que evitar introducir ángulos que anulen los elementos sinusoidales, que como en este caso son cosenos, serán todos los ángulos cercanos a $\frac{k\pi}{2}$. Si escogiésemos algún ángulo de estas magnitudes se generaría una singularidad de $\frac{expr}{0} = \pm\infty$ y el sistema fallaría, devolviendo resultados incoherentes.

Para poder realizar planificaciones que pasen o contengan estos ángulos críticos se estudian dos métodos en el apartado 4.5, tras haber estudiado todos los métodos con ángulos ideales para el sistema de tractor *car-like* y de tractor-remolque.

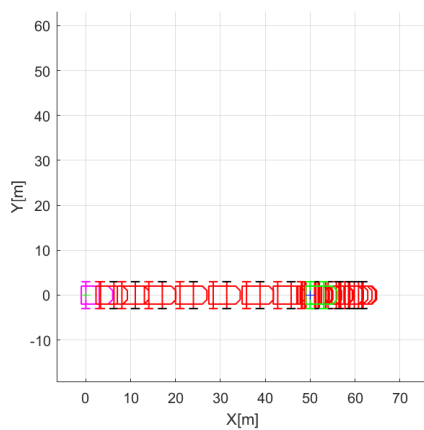
En los siguientes experimentos se presentan los estados iniciales y finales previos a una serie de figuras. En las figuras aparecen: una representación estroboscópica de la animación del sistema en el plano XY, donde se dibujan el estado inicial de color rosa y el final de color verde; una representación del camino que ha llevado el sistema en el plano XY, donde se dibuja el estado inicial con una cruz verde y el final con una cruz azul; y los valores que han tomado las variables del sistema a lo largo del tiempo $T = 10s$, que son los valores de x , y , θ , ϕ , u y v . Todas las distancias están representadas en metros y los ángulos en radianes.

3.4.1. Experimento 1: *Asymptotic Sinusoidal Inputs*

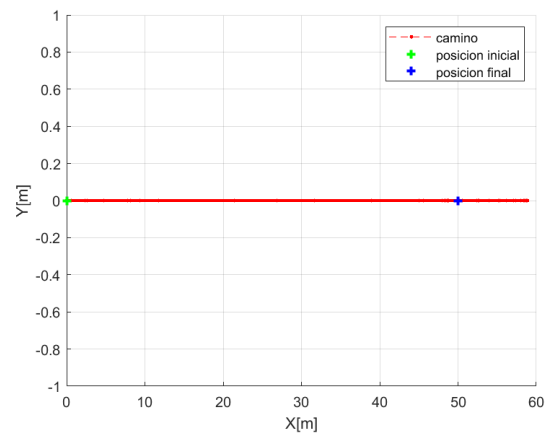
La primera maniobra presentada estudia el método *Asymptotic Sinusoidal Inputs*. En esta trayectoria se busca averiguar el comportamiento del primer input de control u_1 : la velocidad lineal. Para esto se establecen en los estados inicial y final el mismo valor para la componente y y una orientación acorde al propósito. Por tanto, los estados introducidos al sistema, en la forma $(x, y, \theta, \phi)^T$ son:

$$\text{Estado inicial} = \begin{bmatrix} 50 & 0 & 0 & 0 \end{bmatrix}^T$$

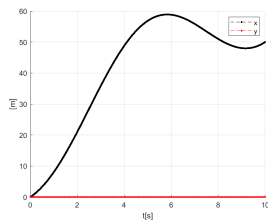
$$\text{Estado final} = \begin{bmatrix} 0 & 0 & 0 & 0 \end{bmatrix}^T$$



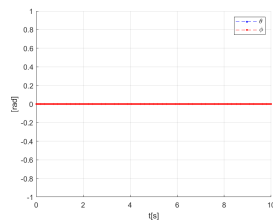
(a) Representación estroboscópica del sistema



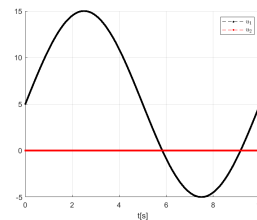
(b) Camino en el plano XY



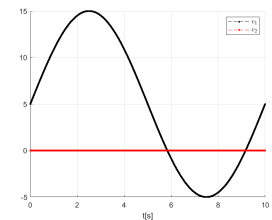
(c) Variables de estado X e Y



(d) Variables de orientación θ y ϕ



(e) input U



(f) input V

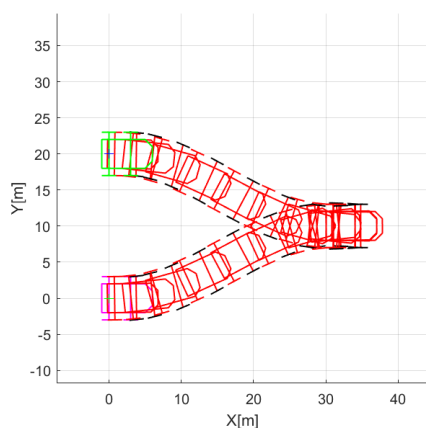
Figura 3.3: Maniobra recta de un vehículo tractor con *Asymptotic Sinusoidal Inputs*

En las figura 3.3 podemos apreciar como el segundo input u_2 y v_2 y las variables de orientación θ y ϕ son nulas, ya que esta trayectoria no hace ningún giro. Por otro lado, en las gráficas de las entradas del sistema se puede comprobar el comportamiento sinusoidal característico del método *Asymptotic Sinusoidal Inputs*. Este método hace que el vehículo continúe avanzando pasado el estado final y posteriormente retroceda hasta pararse en la posición indicada.

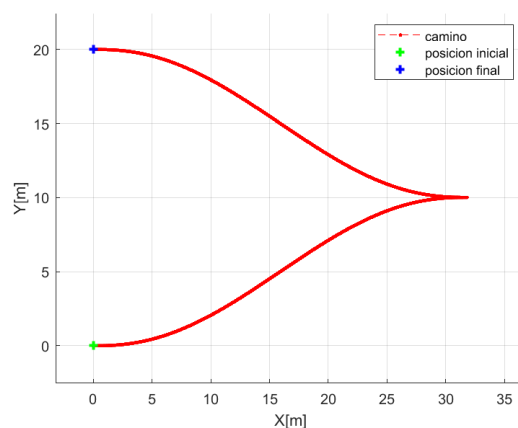
A continuación se planifica un movimiento en paralelo. Para ello se establece una misma coordenada x en los estados inicial y final y una posición distinta para la componente y . Las orientaciones iniciales y finales también se mantienen idénticas. Los estados en forma $(x, y, \theta, \phi)^T$ son:

$$\text{Estado inicial} = \begin{bmatrix} 0 & 0 & 0 & 0 \end{bmatrix}^T$$

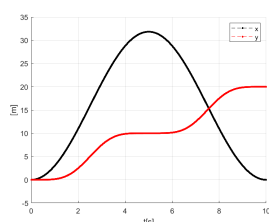
$$\text{Estado final} = \begin{bmatrix} 0 & 20 & 0 & 0 \end{bmatrix}^T$$



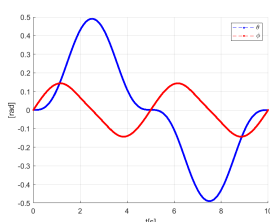
(a) Representación estroboscópica del sistema



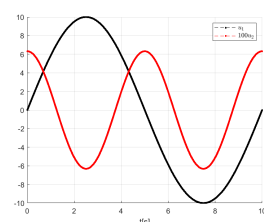
(b) Camino en el plano XY



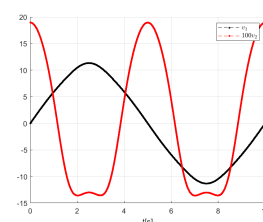
(c) Variables de estado X e Y



(d) Variables de orientación θ y ϕ



(e) input U



(f) input V

Figura 3.4: Maniobra de desplazamiento paralelo de un vehículo tractor con *Asymptotic Sinusoidal Inputs*

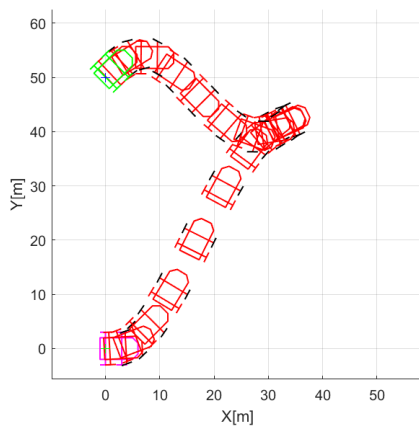
En la figura 3.4 se puede observar como el movimiento de desplazamiento en paralelo se ejecuta de una manera correcta, llevando al sistema del estado inicial al final con una exactitud adecuada. Este tipo de movimientos son para los que este método está desti-

nado, ya que su comportamiento con vaivenes es el más adecuado para desplazarse en paralelo.

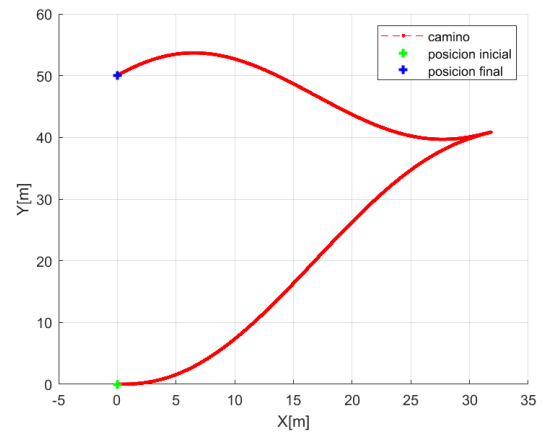
En la siguiente figura se mostrará un desplazamiento paralelo que lleve al sistema hacia un ángulo distinto al del estado inicial. Los estados en forma $(x, y, \theta, \phi)^T$ son:

$$\text{Estado inicial} = \begin{bmatrix} 0 & 0 & 0 & 0 \end{bmatrix}^T$$

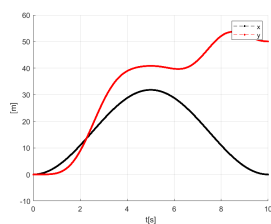
$$\text{Estado final} = \begin{bmatrix} 0 & 50 & \frac{\pi}{4} & 0 \end{bmatrix}^T$$



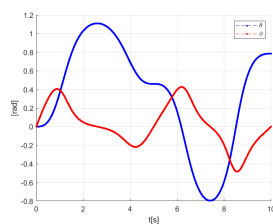
(a) Representación estroboscópica del sistema



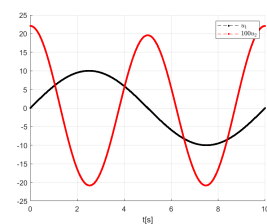
(b) Camino en el plano XY



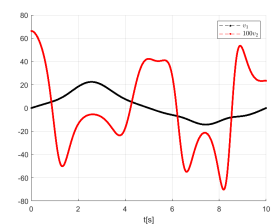
(c) Variables de estado X e Y



(d) Variables de orientación θ y ϕ



(e) input U



(f) input V

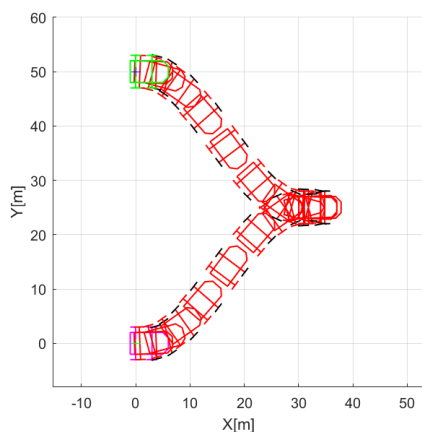
Figura 3.5: Maniobra de desplazamiento paralelo de un vehículo tractor con orientación final distinta y *Asymptotic Sinusoidal Inputs*

En la figura 3.5 podemos ver como el sistema llega al estado final de forma correcta. Tras varias pruebas con distintos valores en las orientaciones finales nos damos cuenta

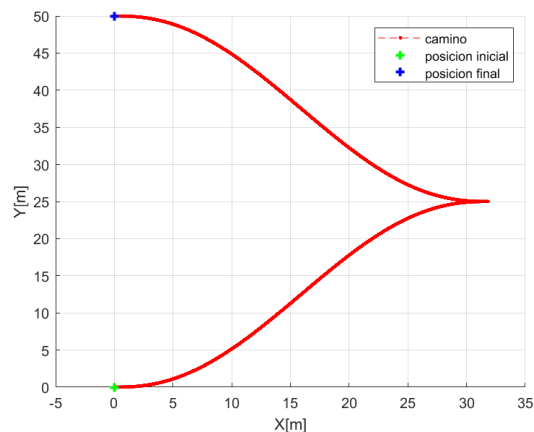
de que el método *Asymptotic Sinusoidal Inputs* es capaz de mover el sistema en un rango de ángulos de $(-\frac{\pi}{2}, \frac{\pi}{2})$. Esto quiere decir que si introducimos ángulos fuera de ese rango se generará un desplazamiento hasta su ángulo simétrico respecto al eje x y el eje y. En el siguiente ejemplo 3.6 se introduce un ángulo final de $\theta = \pi rad$, pero el sistema llega hasta $\theta = 0 rad$.

$$\text{Estado inicial} = \begin{bmatrix} 0 & 0 & 0 & 0 \end{bmatrix}^T$$

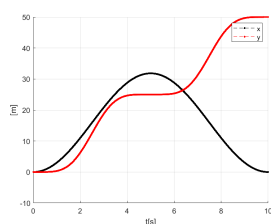
$$\text{Estado final} = \begin{bmatrix} 0 & 50 & \pi & 0 \end{bmatrix}^T$$



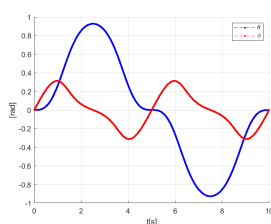
(a) Representación estroboscópica del sistema



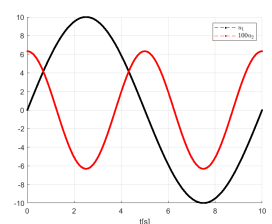
(b) Camino en el plano XY



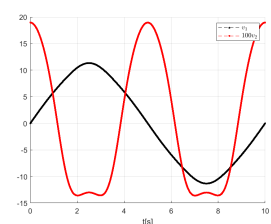
(c) Variables de estado X e Y



(d) Variables de orientación θ y ϕ



(e) input U



(f) input V

Figura 3.6: Maniobra de desplazamiento paralelo de un vehículo tractor con cambio de orientación y *Asymptotic Sinusoidal Inputs*

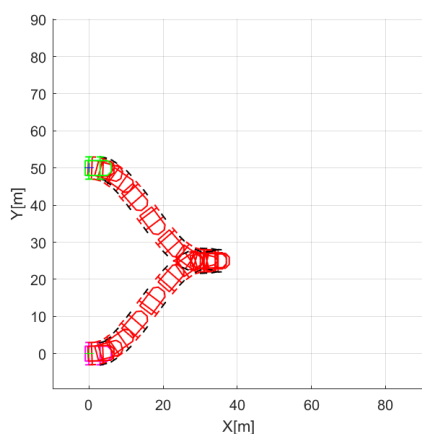
Con este experimento se puede asegurar que el intervalo de trabajo de este método es $(-\frac{\pi}{2}, \frac{\pi}{2})$, y por tanto que todos los movimientos que se generan mueven el sistema hacia

el lado derecho del eje X, siempre iniciando el desplazamiento hacia x mayores a la actual.

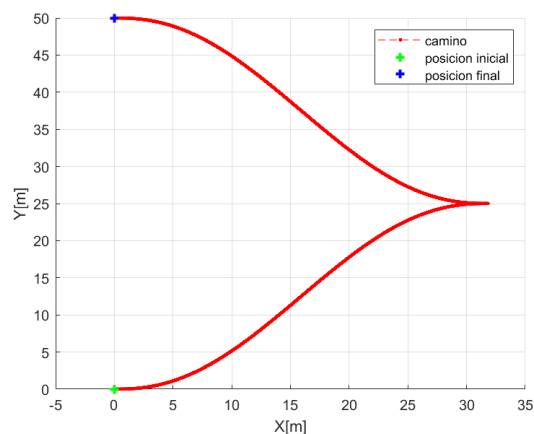
La última maniobra de desplazamiento paralelo estudiará el comportamiento del sistema al cambiar el parámetro a_1 , que hemos establecido hasta ahora de forma arbitraria. Las maniobras ejecutadas a continuación tienen como estados inicial y final en forma $(x, y, \theta, \phi)^T$:

$$\begin{aligned} \text{Estado inicial} &= \begin{bmatrix} 0 & 0 & 0 & 0 \end{bmatrix}^T \\ \text{Estado final} &= \begin{bmatrix} 0 & 50 & 0 & 0 \end{bmatrix}^T \end{aligned}$$

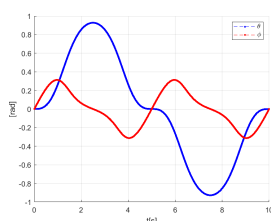
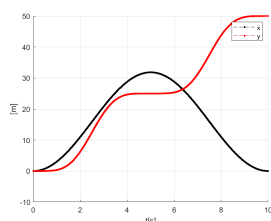
El primer ejemplo tiene $a_1 = 10$:



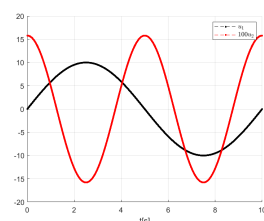
(a) Representación estroboscópica del sistema



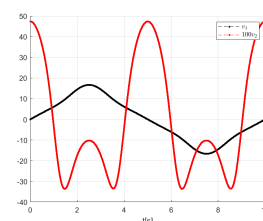
(b) Camino en el plano XY



(c) Variables de estado X e Y (d) Variables de orientación θ y ϕ



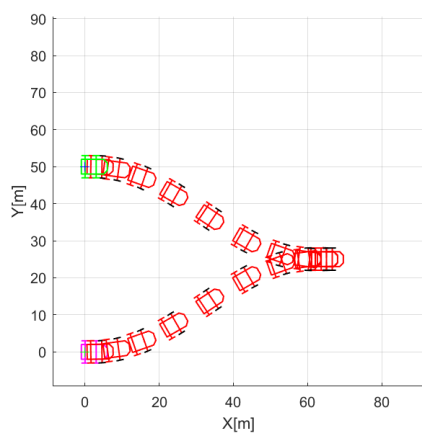
(e) input U



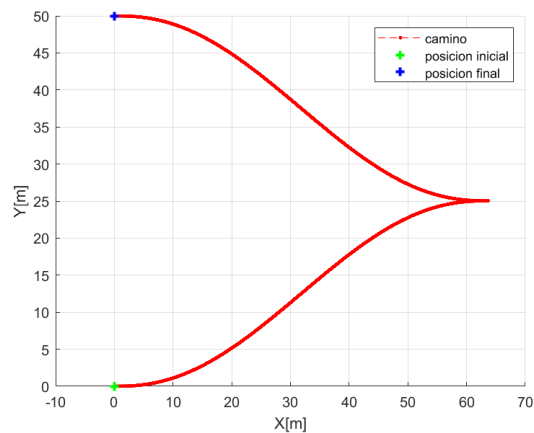
(f) input V

Figura 3.7: Maniobra de desplazamiento paralelo de un vehículo tractor con *Asymptotic Sinusoidal Inputs* y $a_1 = 10$.

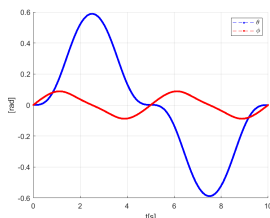
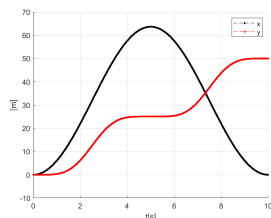
El segundo ejemplo tiene $a_1 = 20$



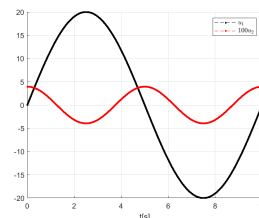
(a) Representación estroboscópica del sistema



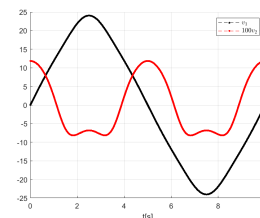
(b) Camino en el plano XY



(c) Variables de estado X e Y e (d) Variables de orientación θ y ϕ



(e) input U



(f) input V

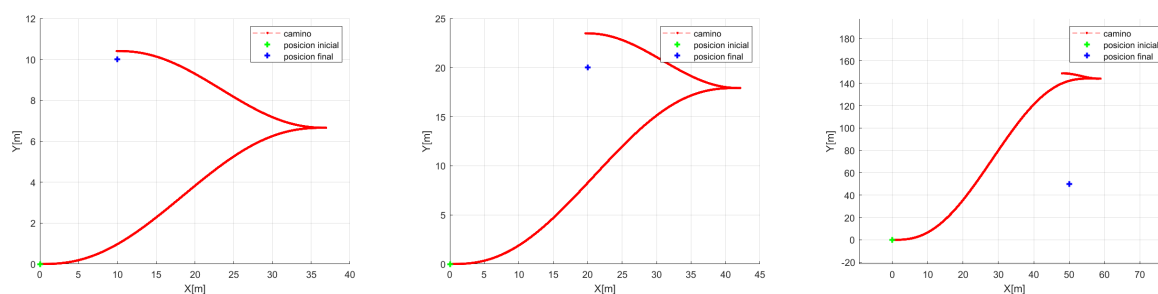
Figura 3.8: Maniobra de desplazamiento paralelo de un vehículo tractor con *Asymptotic Sinusoidal Inputs* y $a_1 = 20$.

Ambos ejemplos llevan al sistema hacia el punto final de forma correcta, pero cuanto más grande es el valor del parámetro a_1 más se separa de la x inicial para llegar a la final.

Por último se estudian las maniobras de desplazamiento diagonal, que llevan al sistema hacia un estado con una x y una y distintas. El análisis de varias trayectorias de este tipo demuestra como el método *Asymptotic Sinusoidal Inputs* no es el más adecuado para estas, ya que no lleva al sistema hasta la posición final de forma precisa. Cuanto más alejamos la posición final de la inicial, el método falla más y desplaza al tractor hasta una posición más errónea. En las siguientes figuras se ven únicamente los caminos que toma el sistema si se le introducen distintas posiciones finales. Todas ellas comienzan el estado

$(x, y, \theta, \phi)^T$:

$$\text{Estado inicial} = \begin{bmatrix} 0 & 0 & 0 & 0 \end{bmatrix}^T$$



(a) estado final = $(10, 10, 0, 0)^T$ (b) estado final = $(20, 20, 0, 0)^T$ (c) estado final = $(50, 50, 0, 0)^T$

Figura 3.9: Comparación de caminos en el plano XY que toma el método *Asymptotic Sinusoidal Inputs* en desplazamientos diagonales.

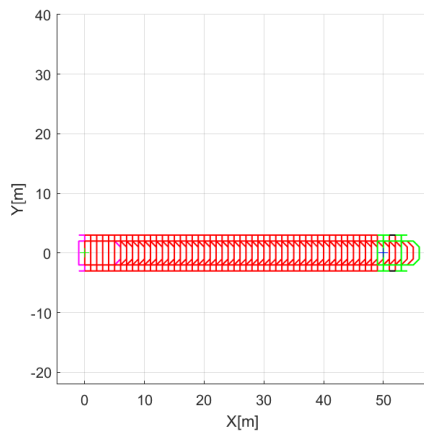
En la figura podemos ver como cuanto más alejado está el punto final, más erra el método. Por ejemplo en la figura 3.9a el sistema falla por unos centímetros, pero en 3.9c se aleja del estado final indicado muchos metros más de los que debería.

3.4.2. Experimento 2: *Piecewise Constant Inputs*

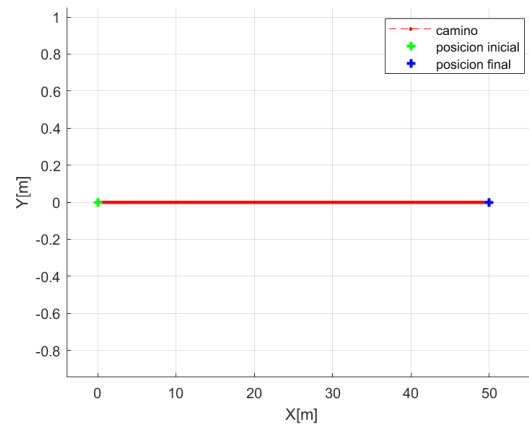
El algoritmo usado para el método *Piecewise Constant Inputs* ha sido modificado respecto al expuesto en 2.2.2; en él en vez de ejecutar un único bucle principal se han anidado 3 bucles, para los cuales se ha utilizado sus respectivos inputs u_i para cada caso.

La primera maniobra es un desplazamiento rectilíneo, que ejecutamos para compararlo con el método *Asymptotic Sinusoidal Inputs*. Los estados de la maniobra en forma $(x, y, \theta, \phi)^T$ son:

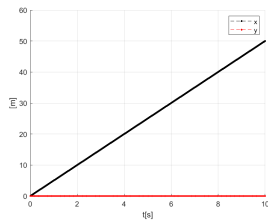
$$\begin{aligned} \text{Estado inicial} &= \begin{bmatrix} 0 & 0 & 0 & 0 \end{bmatrix}^T \\ \text{Estado final} &= \begin{bmatrix} 50 & 0 & 0 & 0 \end{bmatrix}^T \end{aligned}$$



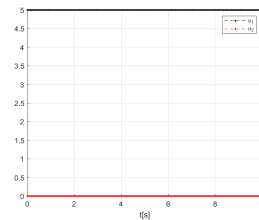
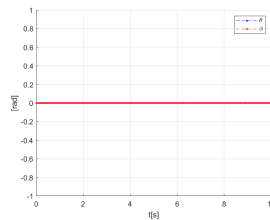
(a) Representación estroboscópica del sistema



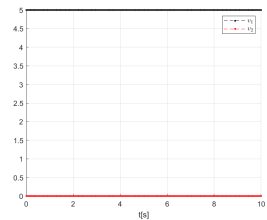
(b) Camino en el plano XY



(c) Variables de estado X e(d) Variables de orientación θ y ϕ



(e) input U



(f) input V

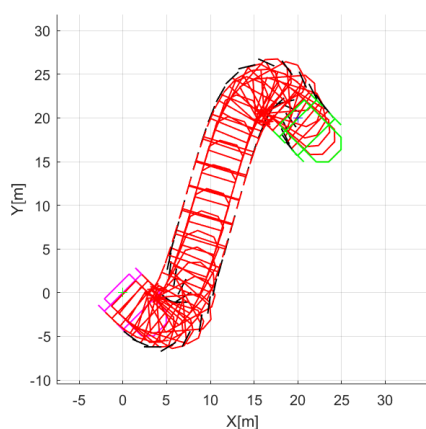
Figura 3.10: Maniobra recta de un vehículo tractor con *Piecewise Constant Inputs*.

En contraposición al caso anterior, *Piecewise Constant Inputs* es capaz de llegar a la posición final asignada sin ningún tipo de retroceso. En este caso, como los inputs son iguales, no se aprecia la diferenciación entre ellos.

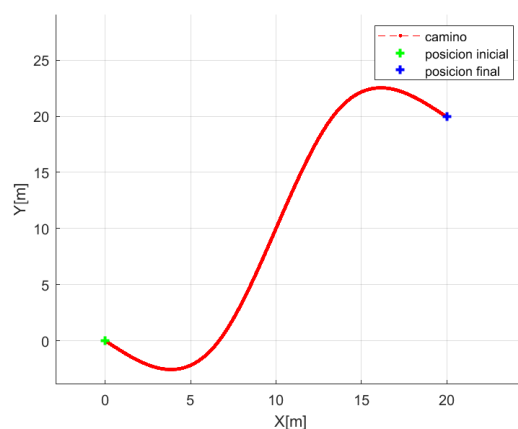
La siguiente maniobra es de desplazamiento diagonal, por lo que establece 2 puntos con coordenadas x e y distintas y posiciones angulares de $\theta = -\frac{\pi}{4}$:

$$\text{Estado inicial} = \begin{bmatrix} 0 & 0 & -\frac{\pi}{4} & 0 \end{bmatrix}^T$$

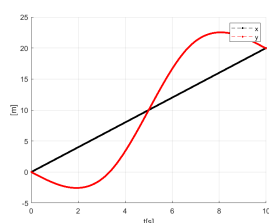
$$\text{Estado final} = \begin{bmatrix} 20 & 20 & -\frac{\pi}{4} & 0 \end{bmatrix}^T$$



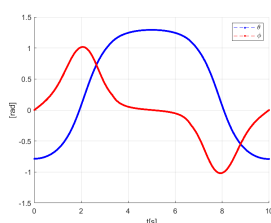
(a) Representación estroboscópica del sistema



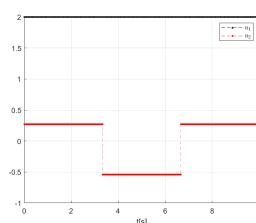
(b) Camino en el plano XY



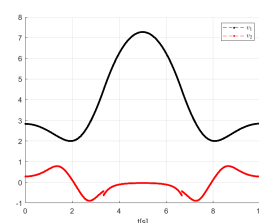
(c) Variables de estado X e Y



(d) Variables de orientación θ y ϕ



(e) input U



(f) input V

Figura 3.11: Maniobra de desplazamiento diagonal de un vehículo tractor con *Piecewise Constant Inputs*.

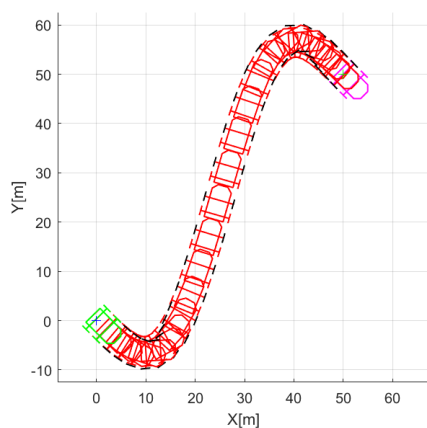
En esta maniobra se pueden observar los distintos inputs u_2 diferenciados a lo largo del tiempo y cómo cambian con un salto finito al transcurrir un tiempo δ , que en este caso se corresponde a $\frac{1}{3}$ del tiempo total de la maniobra. Por otro lado podemos comprobar como el input u_1 se mantiene constante durante toda la trayectoria.

Como se ha comentado anteriormente, este método no es capaz de ejecutar trayectorias en paralelo, cosa que *Asymptotic Sinusoidal Inputs* sí conseguía, por lo que se deberán establecer 2 sub-rutas que al complementarse consigan completar un desplazamiento de este tipo.

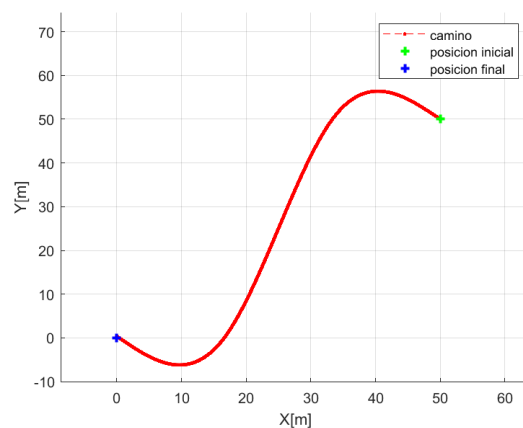
Finalmente comprobamos que este método es capaz de mover el sistema marcha atrás. Los estados de la maniobra en forma $(x, y, \theta, \phi)^T$ son:

$$\text{Estado inicial} = \begin{bmatrix} 50 & 50 & -\frac{\pi}{4} & 0 \end{bmatrix}^T$$

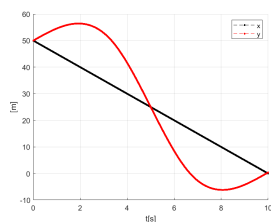
$$\text{Estado final} = \begin{bmatrix} 0 & 0 & -\frac{\pi}{4} & 0 \end{bmatrix}^T$$



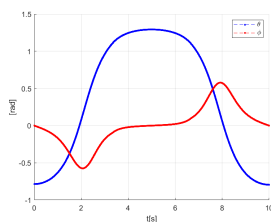
(a) Representación estroboscópica del sistema



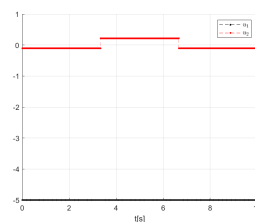
(b) Camino en el plano XY



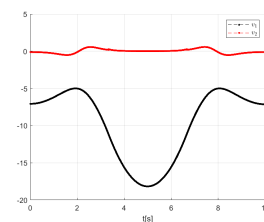
(c) Variables de estado X e Y



(d) Variables de orientación θ y ϕ



(e) input U



(f) input V

Figura 3.12: Maniobra de retroceso de un vehículo tractor con *Piecewise Constant Inputs*.

El método *Piecewise Constant Inputs* consigue ejecutar trayectorias de retroceso para

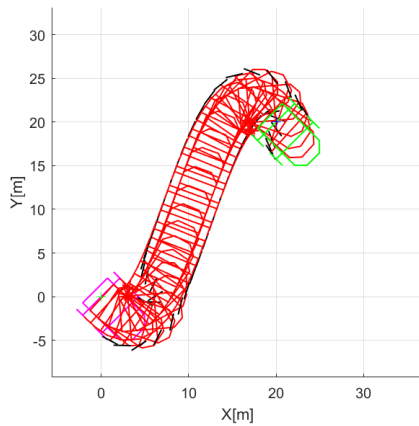
un vehículo tractor de tipo *car-like* con tracción trasera. Asimismo se puede examinar que el problema que había surgido con el método *Asymptotic Sinusoidal Inputs* en el que las distancias largas eran inabarcables queda resuelto en este otro método, que sí es capaz de encontrar velocidades correctas para desplazamientos grandes.

3.4.3. Experimento 3: *Polynomial Inputs*

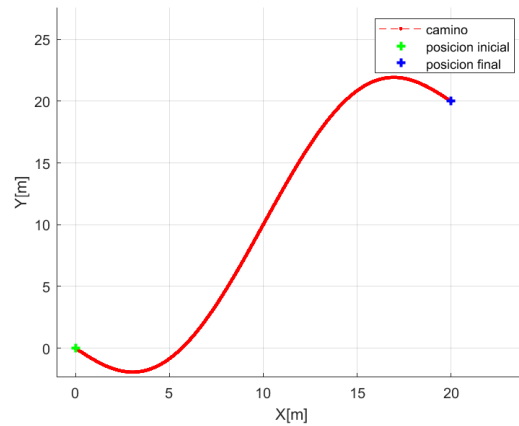
Para finalizar el estudio de los 3 métodos de *steering* se abordará el método basado en *Polynomial Inputs*, que obtiene unos resultados muy similares al caso anterior, pero otorga a sus inputs la ventaja de ser diferenciados y en consecuencia al sistema de ser transformado a un modelo dinámico.

En el caso de una trayectoria recta el resultado es idéntico al obtenido con *Piecewise Constant Inputs*, por lo que no se muestran en este caso. La primera maniobra para *Polynomial Inputs* es de tipo diagonal. Los estados de la maniobra en forma $(x, y, \theta, \phi)^T$ son:

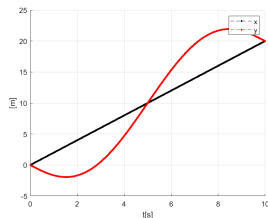
$$\begin{aligned}\text{Estado inicial} &= \begin{bmatrix} 0 & 0 & -\frac{\pi}{4} & 0 \end{bmatrix}^T \\ \text{Estado final} &= \begin{bmatrix} 20 & 20 & -\frac{\pi}{4} & 0 \end{bmatrix}^T\end{aligned}$$



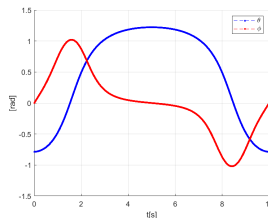
(a) Representación estroboscópica del sistema



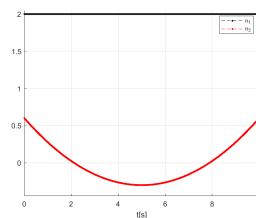
(b) Camino en el plano XY



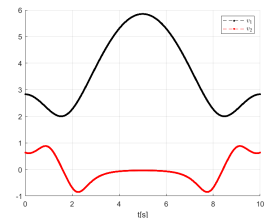
(c) Variables de estado X e Y



(d) Variables de orientación θ y ϕ



(e) input U



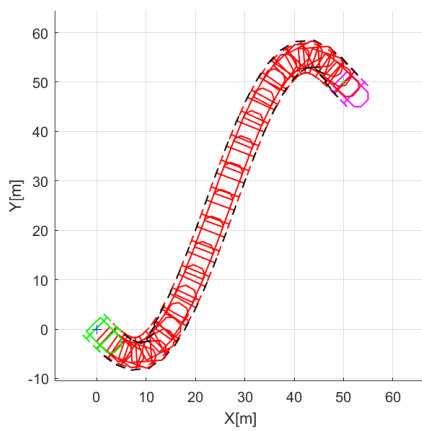
(f) input V

Figura 3.13: Maniobra de desplazamiento diagonal de un vehículo tractor con *Polynomial Inputs*.

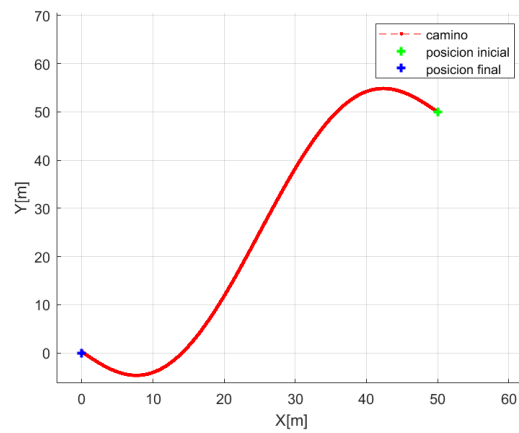
El resultado que se aprecia en 3.13 es el mismo que el que aparece en 3.11 con *Piecewise Constant Inputs*, pero los inputs de ambos son distintos. En este caso el input de velocidad lineal u_1 también es constante, pero el de velocidad angular u_2 no presenta discontinuidades.

Por último se muestra una maniobra de retroceso con *Polynomial Inputs*. Los estados de la maniobra en forma $(x, y, \theta, \phi)^T$ son:

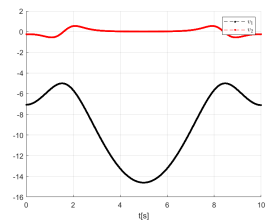
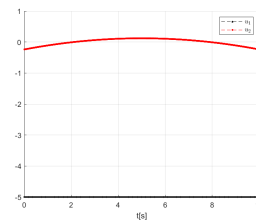
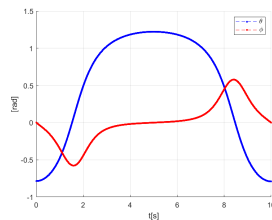
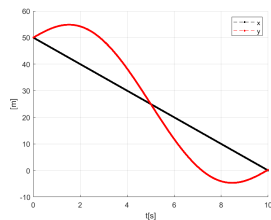
$$\begin{aligned} \text{Estado inicial} &= \begin{bmatrix} 50 & 50 & -\frac{\pi}{4} & 0 \end{bmatrix}^T \\ \text{Estado final} &= \begin{bmatrix} 0 & 0 & -\frac{\pi}{4} & 0 \end{bmatrix}^T \end{aligned}$$



(a) Representación estroboscópica del sistema



(b) Camino en el plano XY



(c) Variables de estado X e Y
(d) Variables de orientación θ y ϕ

(e) input U

(f) input V

Figura 3.14: Maniobra de retroceso de un vehículo tractor con *Polynomial Inputs*

Se puede concluir entonces con que este método obtiene buenos resultados, prácticamente idénticos a los del método anterior con la peculiaridad de utilizar inputs sin discontinuidades.

Capítulo 4

Planificación de maniobras para un tractor con remolque

Una vez estudiados los métodos de planificación que se pueden aplicar a un sistema expresado en *chained form*, podrán ser empleados en cualquier sistema de este tipo de un modo similar. Para ello habrá que modificar el modelo cinemático que lo represente. A continuación se estudiará el modelo de un sistema compuesto por un vehículo *car-like* no holonómico con un remolque sencillo y la aplicación de los métodos de *steering* anteriores para el mismo. Este modelo es el que representa al tractor como vehículo *car-like* y al avión como remolque.

El modelo que representa al tractor será el mismo que se ha estudiado en el capítulo anterior pero no es el único modelo que representa un tractor de maniobras de movimiento para aviones. En el mercado se pueden encontrar 2 clases de vehículos tractores: unos tienen un *towbar* [6], una barra que conecta a este con el remolque y que tiene una articulación más, y otros que no lo tienen. Esta barra otorga al modelo cinemático de una dificultad mayor en cuestión a la transformación del sistema a *chained form* y además es susceptible de romperse si se le aplican fuerzas mal dirigidas o demasiado grandes. La ventaja que concede al sistema es la longitud segura que proporciona al operario del vehículo, dado que se encontrará a una distancia mayor del morro del avión en comparación a un tractor sin *towbar*. La naturaleza automática del *steering* que se propone en este trabajo asume que no existe ningún conductor dentro del vehículo, por lo que escoger un tractor sin barra será la mejor opción para el sistema.

4.1. Modelo de un sistema tractor-remolque

Partimos del modelo del tractor anterior, que estaba definido por sus coordenadas x , y en el espacio, el ángulo θ del mismo respecto al sistema de referencia y el ángulo ϕ de sus ruedas directoras respecto al coche. A este sistema se le añadirá un nuevo parámetro θ_r que representará la posición angular del remolque respecto al mundo. De este modo obtendremos un sistema completamente definido por 5 variables.

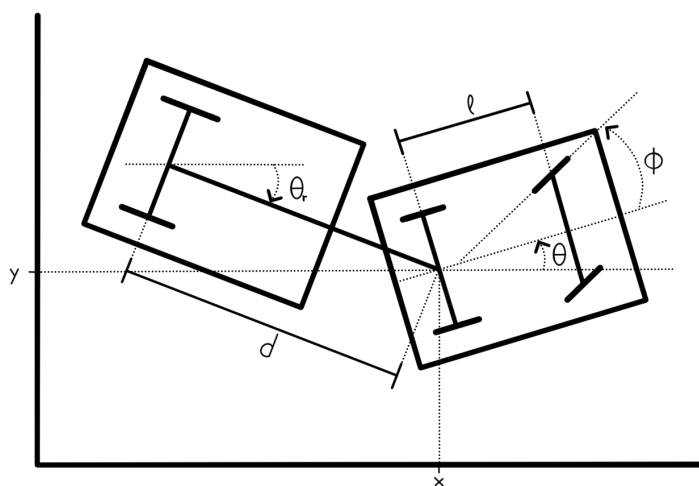


Figura 4.1: Modelo de un tractor de tipo *car-like* con un remolque.

Como en el caso del coche no holonómico, para hallar las ecuaciones del vehículo articulado mantenemos la asunción de que sus ruedas pueden girar sobre sus ejes pero no resbalar, por lo que no puede haber un movimiento lateral de las mismas. Así, el modelo cinemático [4],[3],[10] se describe por:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\phi} \\ \dot{\theta} \\ \dot{\theta}_r \end{bmatrix} = \begin{bmatrix} \cos \theta \\ \sin \theta \\ 0 \\ \frac{\tan(\phi)}{l} \\ \frac{\sin(\theta - \theta_r)}{d} \end{bmatrix} v_1 + \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} v_2.$$

$$\dot{x} = \begin{bmatrix} \cos \theta v_1 \\ \sin \theta v_1 \\ v_2 \\ \frac{\tan(\phi)}{l} v_1 \\ \frac{\sin(\theta - \theta_r)}{d} v_1 \end{bmatrix} \quad (4.1)$$

Donde l define la distancia entre los ejes del coche y d la longitud de la barra o elemento que conecta el coche con el remolque.

4.2. Modelo tractor-remolque en *chained form*

Hallar el modelo expresado en *chained form* es imprescindible para poder aplicar los métodos de *steering* estudiados en los capítulos anteriores, por lo que en esta sección se presenta el modelo en *chained form* del tractor con remolque.

La forma de este sistema encadenado será un (5,2), con 5 variables de estado y 2 inputs de control:

$$\begin{aligned} \dot{x}_1 &= u_1, \\ \dot{x}_2 &= u_2, \\ \dot{x}_3 &= x_2 u_1, \\ \dot{x}_4 &= x_3 u_1, \\ \dot{x}_5 &= x_4 u_1. \end{aligned}$$

Para este caso se han estudiado las transformaciones a *chained form* de los papers [4],[3],[10], con las cuales ha sido imposible encontrar un modelo que devuelva un buen resultado al aplicar los controles interesantes. Para conseguir la transformación correcta del modelo se han escogido únicamente los cambios en las variables de estado x_1 y x_5 , que sí han sido útiles para encontrar la forma correcta de los inputs v_1 y v_2 . Estas igualdades son:

$$\begin{aligned} x_1 &= x, \\ x_5 &= y - d \ln \left(\frac{1 + \sin(\theta_r)}{\cos(\theta_r)} \right). \end{aligned}$$

A partir de estas 2 transformaciones y la forma del modelo cinemático del sistema 4.1 seremos capaces de encontrar el resto de sustituciones del mismo modo que se demostraron las del modelo *car-like* en el capítulo anterior. Se comenzará con la variable x_5 y, paso a paso, se descubrirá que casi todas las sustituciones eran correctas excepto la última. Esta corresponde con x_2 y por tanto con el segundo input u_2 del sistema, es decir, la velocidad angular que domina los movimientos del vehículo articulado.

Como $x_1 = x$:

$$\dot{x}_1 = \dot{x} = v_1 \cos(\theta)$$

Como queremos que \dot{x}_1 tenga la forma $\dot{x}_1 = u_1$, podemos asegurar que $u_1 = v_1 \cos(\theta)$ y por tanto $v_1 = \frac{u_1}{\cos(\theta)} = \sec(\theta)u_1$. De este modo se ha obtenido el valor de la transformación del primer input.

Continuamos con: $x_5 = y - d \ln \left(\frac{1 + \sin(\theta_r)}{\cos(\theta_r)} \right)$:

$$\begin{aligned} \dot{x}_5 &= \frac{d}{dt} \left[y - d \ln \left(\frac{1 + \sin(\theta_r)}{\cos(\theta_r)} \right) \right] \\ &= \frac{\cos(\theta) \sin(\theta_r)}{\cos(\theta_r)} v_1 \\ &= \frac{\cos(\theta) \sin(\theta_r)}{\cos(\theta_r) \cos(\theta)} u_1 = \tan(\theta_r) u_1 \end{aligned}$$

Sabiendo que $\dot{x}_5 = x_4 u_1$, se ha obtenido la transformación de x_4 con valor $x_4 = \tan(\theta_r)$. Continuamos con la siguiente variable:

$$\begin{aligned} \dot{x}_4 &= \frac{d}{dt} [\tan(\theta_r)] \\ &= \frac{\sin(\theta - \theta_r)(\tan(\theta_r)^2 + 1)}{d} v_1 \\ &= \frac{\sin(\theta - \theta_r)(\tan(\theta_r)^2 + 1)}{d \cos(\theta)} u_1 = \frac{1}{d} \sin(\theta - \theta_r) \sec(\theta_r)^2 \sec(\theta) u_1 \end{aligned}$$

Por lo que se demuestra que $x_3 = \frac{1}{d} \sin(\theta - \theta_r) \sec(\theta_r)^2 \sec(\theta)$.

$$\begin{aligned}
 \dot{x}_3 &= \frac{d}{dt} \left[\frac{1}{d} \sin(\theta - \theta_r) \sec(\theta_r)^2 \sec(\theta) \right] \\
 &= \frac{2v_1 \sin(\theta - \theta_r)^2 \sin(\theta_r)}{d^2 \cos(\theta) \cos(\theta_r)^3} - \frac{\cos(\theta - \theta_r) v_1 (l \sin(\theta - \theta_r) - d \tan(\phi))}{d^2 l \cos(\theta) \cos(\theta_r)^2} \\
 &\quad + \frac{v_1 \sin(\theta - \theta_r) \tan(\phi) \sin(\theta)}{dl \cos(\theta)^2 \cos(\theta_r)^2} \\
 &= \frac{2u_1 \sin(\theta - \theta_r)^2 \sin(\theta_r)}{d^2 \cos(\theta)^2 \cos(\theta_r)^3} - \frac{\cos(\theta - \theta_r) u_1 (l \sin(\theta - \theta_r) - d \tan(\phi))}{d^2 l \cos(\theta)^2 \cos(\theta_r)^2} \\
 &\quad + \frac{u_1 \sin(\theta - \theta_r) \tan(\phi) \sin(\theta)}{dl \cos(\theta)^3 \cos(\theta_r)^2} \\
 &= u_1 \left[\frac{2 \sin(\theta - \theta_r)^2 \sin(\theta_r)}{d^2 \cos(\theta)^2 \cos(\theta_r)^3} - \frac{\cos(\theta - \theta_r) (l \sin(\theta - \theta_r) - d \tan(\phi))}{d^2 l \cos(\theta)^2 \cos(\theta_r)^2} \right. \\
 &\quad \left. + \frac{\sin(\theta - \theta_r) \tan(\phi) \sin(\theta)}{dl \cos(\theta)^3 \cos(\theta_r)^2} \right]
 \end{aligned}$$

Con lo que se podrá decir que:

$$\begin{aligned}
 x_2 &= \frac{2 \sin(\theta - \theta_r)^2 \sin(\theta_r)}{d^2 \cos(\theta)^2 \cos(\theta_r)^3} - \frac{\cos(\theta - \theta_r) (l \sin(\theta - \theta_r) - d \tan(\phi))}{d^2 l \cos(\theta)^2 \cos(\theta_r)^2} \\
 &\quad + \frac{\sin(\theta - \theta_r) \tan(\phi) \sin(\theta)}{dl \cos(\theta)^3 \cos(\theta_r)^2}
 \end{aligned}$$

y finalmente se podrá calcular el valor de del segundo input v_2 :

$$\begin{aligned}
 \dot{x}_2 &= \frac{d}{dt} \left[\frac{2 \sin(\theta - \theta_r)^2 \sin(\theta_r)}{d^2 \cos(\theta)^2 \cos(\theta_r)^3} - \frac{\cos(\theta - \theta_r) (l \sin(\theta - \theta_r) - d \tan(\phi))}{d^2 l \cos(\theta)^2 \cos(\theta_r)^2} \right. \\
 &\quad \left. + \frac{\sin(\theta - \theta_r) \tan(\phi) \sin(\theta)}{dl \cos(\theta)^3 \cos(\theta_r)^2} \right] \\
 &\quad \vdots \\
 &= \beta_1 u_1 + \beta_2 u_2
 \end{aligned}$$

Donde :

$$\begin{aligned}\beta_1 &= \frac{1}{2d^2l} \cos(\phi)^2 \sec(\theta) (2dl \cos(\theta) \sec(\theta_r) \tan(\phi) \\ &\quad - 6d \tan(\phi) (d \tan(\phi) \tan(\theta) + l \sin(\theta - \theta_r) \tan(\theta_r)) \\ &\quad - l^2 \sec(\theta_r)^2 (\sin(2\theta) \sin(\theta - \theta_r)^2 + \cos(\theta)^2 (\sin(2(\theta - \theta_r)) - 12 \sin(\theta - \theta_r)^2 \tan(\theta_r))) \\ \beta_2 &= ld \cos(\theta)^3 \cos(\theta_r) \cos(\phi)^2\end{aligned}$$

Tras elaborar estos cálculos se han conseguido encontrar las transformaciones de inputs y coordenadas adecuadas para transformar a *chained form* el sistema tractor-remolque, que se resumen en:

$$\begin{aligned}x_1 &= x, \\ x_2 &= \frac{2 \sin(\theta - \theta_r)^2 \sin(\theta_r)}{d^2 \cos(\theta)^2 \cos(\theta_r)^3} - \frac{\cos(\theta - \theta_r) (l \sin(\theta - \theta_r) - d \tan(\phi))}{d^2 l \cos(\theta)^2 \cos(\theta_r)^2} \\ &\quad + \frac{\sin(\theta - \theta_r) \tan(\phi) \sin(\theta)}{dl \cos(\theta)^3 \cos(\theta_r)^2}, \\ x_3 &= \frac{1}{d} \sin(\theta - \theta_r) \sec(\theta_r)^2 \sec(\theta), \\ x_4 &= \tan(\theta_r), \\ x_5 &= y - d \log\left(\frac{1 + \sin(\theta_r)}{\cos(\theta_r)}\right), \\ v_1 &= \sec(\theta) u_1, \\ v_2 &= \beta_1 u_1 + \beta_2 u_2.\end{aligned}\tag{4.2}$$

4.3. Métodos de *steering* para un sistema tractor-remolque en *chained form*

Finalmente, para controlar el sistema completo, se escogerá el método de *steering* más adecuado, teniendo en cuenta los resultados obtenidos en el control del tractor, la complejidad que cada uno supone y la maniobra que nos interesa realizar.

Como se ha visto anteriormente, el método de *Asymptotic Sinusoidal Inputs* será útil para maniobras de movimiento en paralelo, como puede ser el caso de aparcar un vehículo en línea. Además de ser el más adecuado para este tipo de situaciones, es el único que

permite establecer el mismo valor inicial y final en la primera coordenada del sistema (la posición en x), sin tener que dividir la trayectoria en dos sub-controles distintos. Para el problema de *pushback* que plantea este trabajo con distintas posiciones en x y sin necesidad de recolocaciones, este método no será el más útil de los que se han estudiado pero sí lo será para otro tipo de maniobras.

Los otros dos métodos de *steering* que se podrán utilizar son *Piecewise Constant Inputs* y *Polynomial Inputs*. En ambos se aprecia una complejidad en el cálculo de sus parámetros mucho más abordable, la ventaja de tener un input constante durante toda la maniobra y buenos resultados en la práctica. Estos dos métodos comparten muchas cualidades, pero por sus formas se ha visto que *Polynomial Inputs* es el más apropiado para poder ser usado posteriormente en el *steering* real de un vehículo.

El método más conveniente para maniobras de *pushback* es *Polynomial Inputs*. Por otra parte, también se estudiará el método *Asymptotic Sinusoidal Inputs* para tener una opción de *steering* ideal en maniobras paralelas. A continuación se calcularán las constantes que conforman los inputs del sistema de forma análoga al caso *car-like* del capítulo anterior.

4.3.1. Método basado en *Asymptotic Sinusoidal Inputs*

La forma de los inputs sinusoidales para el modelo de tractor *car-like* con un remolque propuestos en las fórmulas 3.10 es:

$$\begin{aligned} u_1 &= a_0 + a_1 \sin(\omega t), \\ u_2 &= b_0 + b_1 \cos(\omega t) + b_2 \cos(2\omega t) + b_3 \cos(3\omega t). \end{aligned}$$

A partir de esta forma de u_1 y u_2 se pueden integrar las ecuaciones del modelo en *chained form* con la estructura 3.10 para formar un sistema que devuelva los parámetros a_i y b_i . Manteniendo a_1 como parámetro, el resto de constantes resultan en:

$$\begin{aligned}
a0 &= \frac{x_1(t_F) - x_1(t_I)}{t_F} \\
b0 &= \frac{x_2(t_F) - x_2(t_I)}{t_F} \\
b1 &= \frac{1}{a_1 t_F^2} (2(2\pi x_3(t_F) - 2\pi x_3(t_I) - \pi x_1(t_F)x_2(t_F) - \pi x_1(t_F)x_2(t_I) \\
&\quad + \pi x_2(t_F)x_1(t_I) + \pi x_1(t_I)x_2(t_I) + a_1 x_2(t_F)t_F - a_1 x_2(t_I)t_F) \\
b2 &= \frac{1}{3a_1^2 t_F^3} (2(48x_4(t_F)\pi^2 - 48x_4(t_I)\pi^2 + 4x_1(t_F)^2 x_2(t_F)\pi^2 + 4x_2(t_F)x_1(t_I)^2 \pi^2 \\
&\quad - 4x_1(t_F)^2 x_2(t_I)\pi^2 - 4x_1(t_I)^2 x_2(t_I)\pi^2 + 3a_1^2 x_2(t_F)t_F^2 - 3a_1^2 x_2(t_I)t_F^2 \\
&\quad - 24x_1(t_F)x_3(t_F)\pi^2 - 24x_1(t_F)x_3(t_I)\pi^2 + 24x_3(t_F)x_1(t_I)\pi^2 + 24x_1(t_I)x_3(t_I)\pi^2 \\
&\quad - 8x_1(t_F)x_2(t_F)x_1(t_I)\pi^2 + 8x_1(t_F)x_1(t_I)x_2(t_I)\pi^2 + 24\pi a_1 x_3(t_F)t_F \\
&\quad - 24\pi a_1 x_3(t_I)t_F - 12\pi a_1 x_1(t_F)x_2(t_F)t_F + 12\pi a_1 x_1(t_F)x_2(t_I)t_F \\
&\quad + 12\pi a_1 x_2(t_F)x_1(t_I)t_F - 12\pi a_1 x_1(t_I)x_2(t_I)t_F)
\end{aligned}$$

$$\begin{aligned}
b3 = & \frac{1}{a_1^3 t_F^4} 384x_5(t_F)\pi^3 - 384x_5(t_I)\pi^3 + 32x_1(t_F)^2 x_3(t_F)\pi^3 + 32x_3(t_F)x_1(t_I)^2 \pi^3 \\
& - 32x_1(t_F)^2 x_3(t_I)\pi^3 - 32x_1(t_I)^2 x_3(t_I)\pi^3 + 2a_1^3 x_2(t_F)t_F^3 - \\
& 2a_1^3 x_2(t_I)t_F^3 + 96\pi x_1(t_F)^2 x_3(t_F) - 48\pi x_1(t_F)^3 x_2(t_F) \\
& - 192x_1(t_F)x_4(t_F)\pi^3 + 48\pi x_2(t_F)x_1(t_I)^3 + 96\pi x_3(t_F)x_1(t_I)^2 \\
& - 96\pi x_1(t_F)^2 x_3(t_I) - 48\pi x_1(t_F)^3 x_2(t_I) - 192x_1(t_F)x_4(t_I)\pi^3 \\
& + 192x_4(t_F)x_1(t_I)\pi^3 - 96\pi x_1(t_I)^2 x_3(t_I) + 48\pi x_1(t_I)^3 x_2(t_I) \\
& + 192x_1(t_I)x_4(t_I)\pi^3 - 192\pi x_1(t_F)x_3(t_F)x_1(t_I) + 192\pi x_1(t_F)x_1(t_I)x_3(t_I) \\
& + 192a_1 x_4(t_F)t_F\pi^2 - 192a_1 x_4(t_I)t_F\pi^2 - 144\pi x_1(t_F)x_2(t_F)x_1(t_I)^2 \\
& + 144\pi x_1(t_F)^2 x_2(t_F)x_1(t_I) - 64x_1(t_F)x_3(t_F)x_1(t_I)\pi^3 - 144\pi x_1(t_F)x_1(t_I)^2 x_2(t_I) \\
& + 144\pi x_1(t_F)^2 x_1(t_I)x_2(t_I) + 64x_1(t_F)x_1(t_I)x_3(t_I)\pi^3 \\
& + 48a_1 x_1(t_F)^2 x_2(t_F)t_F + 48a_1 x_2(t_F)x_1(t_I)^2 t_F - 48a_1 x_1(t_F)^2 x_2(t_I)t_F \\
& - 48a_1 x_1(t_I)^2 x_2(t_I)t_F + 36\pi a_1^2 x_3(t_F)t_F^2 - 36\pi a_1^2 x_3(t_I)t_F^2 \\
& + 16a_1 x_1(t_F)^2 x_2(t_F)t_F\pi^2 - 18\pi a_1^2 x_1(t_F)x_2(t_F)t_F^2 \\
& + 16a_1 x_2(t_F)x_1(t_I)^2 t_F\pi^2 - 16a_1 x_1(t_F)^2 x_2(t_I)t_F\pi^2 \\
& + 54\pi a_1^2 x_1(t_F)x_2(t_I)t_F^2 + 18\pi a_1^2 x_2(t_F)x_1(t_I)t_F^2 \\
& - 16a_1 x_1(t_I)^2 x_2(t_I)t_F\pi^2 - 54\pi a_1^2 x_1(t_I)x_2(t_I)t_F^2 \\
& - 96a_1 x_1(t_F)x_2(t_F)x_1(t_I)t_F + 96a_1 x_1(t_F)x_1(t_I)x_2(t_I)t_F \\
& - 96a_1 x_1(t_F)x_3(t_F)t_F\pi^2 + 96a_1 x_1(t_F)x_3(t_I)t_F\pi^2 \\
& + 96a_1 x_3(t_F)x_1(t_I)t_F\pi^2 - 96a_1 x_1(t_I)x_3(t_I)t_F\pi^2 \\
& - 32a_1 x_1(t_F)x_2(t_F)x_1(t_I)t_F\pi^2 + 32a_1 x_1(t_F)x_1(t_I)x_2(t_I)t_F\pi^2
\end{aligned}$$

4.3.2. Método basado en *Polynomial Inputs*

El modelo del vehículo articulado con un remolque tiene una forma en *chained form* (5,2), por lo que la forma general de sus inputs 3.16 con 4 parámetros c_i se deduce como:

$$\begin{aligned}
u_1 &= b_1 \\
u_2 &= c_0 + c_1 t + c_2 t^2 + c_3 t^3
\end{aligned}$$

b_1 continuará teniendo un valor proporcional al desplazamiento en x y a la duración de la maniobra: $b_1 = \frac{x_1(t_F) - x_1(t_I)}{t_F}$. Por ende necesitaremos 4 ecuaciones para resolver los parámetros c_0, c_1, c_2 y c_3 que seguirán las ecuaciones 3.17:

$$\begin{aligned}
 x_2(t_F) &= x_2(t_I) + c_0 t_F + \frac{c_1 t_F^2}{2} + \frac{c_2 t_F^3}{3} + \frac{c_3 t_F^4}{4} \\
 x_3(t_F) &= x_3(t_I) + b_1 t_f \left(\frac{3c_3 t_F^4 + 5c_2 t_F^3 + 10c_1 t_f^2 + 30c_0 t_F + 60x_2(t_I)}{60} \right) \\
 x_4(t_F) &= x_4(t_I) + b_1 t_f \left(\frac{b_1 c_3 t_F^5 + 2b_1 c_2 t_F^4 + 5b_1 c_1 t_F^3 + 20b_1 c_0 t_f^2 + 60b_1 x_2(t_I) t_F + 120x_3(t_I)}{120} \right) \\
 x_5(t_F) &= x_5(t_I) + x_4(t_I) b_1 t_F + \frac{c_3 b_1^3 t_F^7}{840} + \frac{c_2 b_1^3 t_F^6}{360} + \frac{c_1 b_1^3 t_F^5}{120} + \frac{c_0 b_1^3 t_F^4}{24} + \frac{x_2(t_I) b_1^3 t_F^3}{6} + \frac{x_3(t_I) b_1^2 t_F^2}{2}
 \end{aligned}$$

De las cuales se despejan los parámetros c :

$$\begin{aligned}
 c_0 &= -\frac{1}{b_1^3 t_F^4} 4(210x_5(t_I) - 210x_5(t_F) - 15b_1^2 t_F^2 x_3(t_F) + b_1^3 t_F^3 x_2(t_F) \\
 &\quad + 30b_1^2 t_F^2 x_3(t_I) + 4b_1^3 t_F^3 x_2(t_I) + 90b_1 t_F x_4(t_F) + 120b_1 t_F x_4(t_I)) \\
 c_1 &= \frac{1}{b_1^3 t_F^5} 60(168x_5(t_I) - 168x_5(t_F) - 14b_1^2 t_F^2 x_3(t_F) + b_1^3 t_F^3 x_2(t_F) \\
 &\quad + 20b_1^2 t_F^2 x_3(t_I) + 2b_1^3 t_F^3 x_2(t_I) + 78b_1 t_F x_4(t_F) + 90b_1 t_F x_4(t_I)) \\
 c_2 &= -\frac{1}{b_1^3 t_F^6} 60(420x_5(t_I) - 420x_5(t_F) - 39b_1^2 t_F^2 x_3(t_F) + 3b_1^3 t_F^3 x_2(t_F) \\
 &\quad + 45b_1^2 t_F^2 x_3(t_I) + 4b_1^3 t_F^3 x_2(t_I) + 204b_1 t_F x_4(t_F) + 216b_1 t_F x_4(t_I)) \\
 c_3 &= \frac{1}{b_1^3 t_F^7} 140(120x_5(t_I) - 120x_5(t_F) - 12b_1^2 t_F^2 x_3(t_F) + b_1^3 t_F^3 x_2(t_F) \\
 &\quad + 12b_1^2 t_F^2 x_3(t_I) + b_1^3 t_F^3 x_2(t_I) + 60b_1 t_F x_4(t_F) + 60b_1 t_F x_4(t_I))
 \end{aligned}$$

4.4. Resultados

Como en el capítulo anterior, en las figuras de los siguientes experimentos aparecen: una representación estroboscópica de la animación del sistema en el plano XY, donde se dibujan el estado inicial de color rosa y el final de color verde; una representación del camino que ha llevado el sistema en el plano XY, donde se dibuja el estado inicial con una cruz verde y el final con una cruz azul; y los valores que han tomado las variables del sistema a lo largo del tiempo $T = 10s$, que son los valores de x , y , θ , ϕ , u y v . Todas las distancias están representadas en metros y los ángulos en radianes.

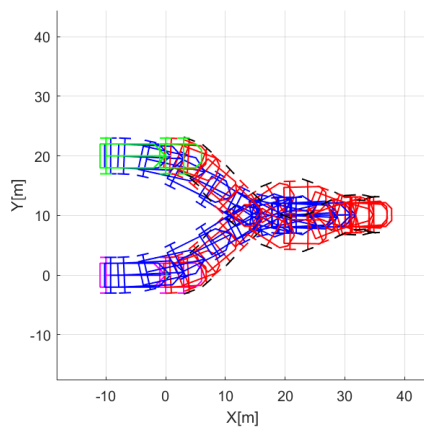
4.4.1. Experimento 1: maniobra de desplazamiento paralelo

Dado que el método *Asymptotic Sinusoidal Inputs* es el más adecuado para maniobras de *steering* en paralelo, se estudiará qué desplazamientos pueden aplicarse a un sistema de un tractor con remolque. Se analizan los casos en los que la posición en x es compartida en el estado inicial y final. No se estudia el caso en el que la coordenada compartida es la componente y ya que supone una orientación del sistema en $\frac{\pi}{2}rad$, lo que genera singularidades 4.5. Los estados correspondientes a las siguientes maniobras tienen la forma $(x, y, \phi, \theta, \theta_r)^T$ y son:

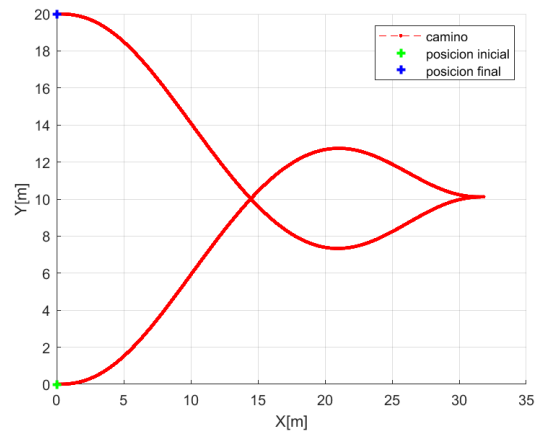
$$\begin{aligned} \text{Estado inicial} &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \\ \text{Estado final} &= \begin{bmatrix} 0 & 20 & 0 & 0 & 0 \end{bmatrix}^T \end{aligned}$$

Inicialmente se aplicarán los parámetros a_0 y b_k calculados en el apartado 4.3.1. Estos parámetros, al igual que en el caso de sistema *car-like*, están especificados en función a una variable arbitraria a_1 . La modificación de este valor cambiará el comportamiento del sistema. El valor de a_1 es proporcional a la distancia que recorre el sistema para recolocarse en la misma posición x . Cuanto más grande sea a_1 , más se alejará de la x inicial para llegar a la final y, por tanto, los giros serán mas suaves.

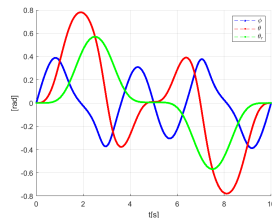
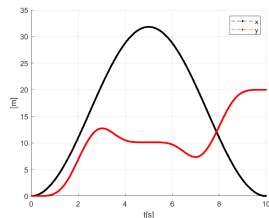
En el primer ejemplo se escoge una a_1 igual a 10:



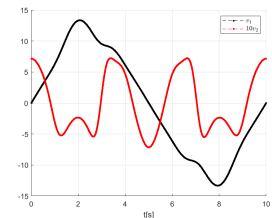
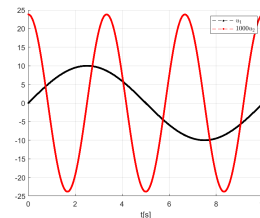
(a) Representación estroboscópica del sistema



(b) Camino en el plano XY



(c) Variables de estado X e(d) Variables de orientación ϕ, θ, θ_r

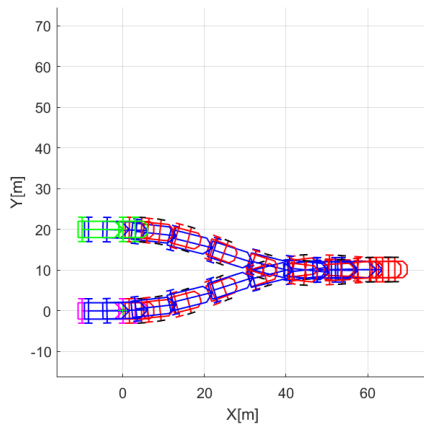


(e) input U (f) input V

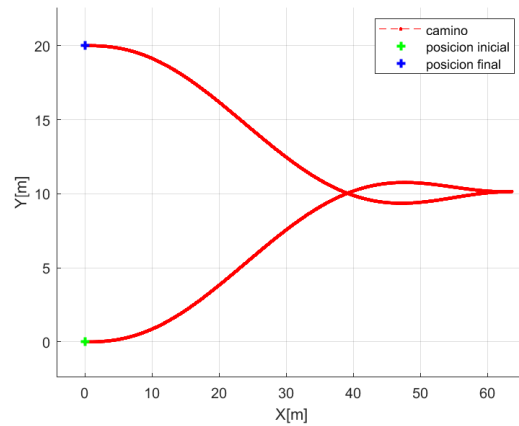
Figura 4.2: Maniobra de desplazamiento paralelo con *Asymptotic Sinusoidal Inputs* y $a_1 = 10$.

Ahora, si establecemos un a_1 mayor obtendremos un comportamiento distinto del sistema. Se puede observar como la posición x intermedia que alcanza el sistema está mas alejada de la x inicial y final.

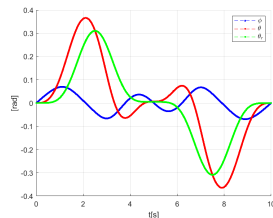
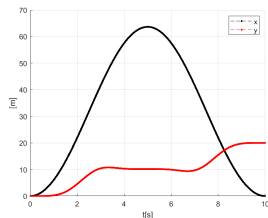
Con a_1 establecida a 20:



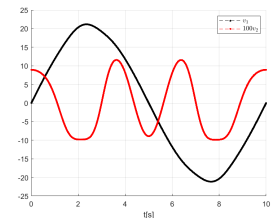
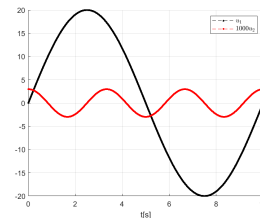
(a) Representación estroboscópica del sistema



(b) Camino en el plano XY



(c) Variables de estado X e Y
(d) Variables de orientación ϕ , θ , θ_r



(e) input U
(f) input V

Figura 4.3: Maniobra de desplazamiento paralelo con *Asymptotic Sinusoidal Inputs* y $a_1 = 20$.

Con esta prueba podemos concluir con que se puede cambiar el comportamiento del *steering* del sistema tractor-remolque modificando el parámetro a_1 .

Todos los movimientos paralelos que se generan (con la misma posición x para el estado inicial y final) comienzan hacia el lado positivo del eje x . Por este motivo a la hora de hacer los experimentos y pruebas posteriores con este método, el escenario en el que se aplique el movimiento debe estar modificado para que el desplazamiento que se busca esté siempre iniciado hacia la derecha.

En la siguiente figura podemos comprobar cómo, estableciendo distintas orientaciones iniciales y finales, el método siempre se dirige inicialmente hacia el lado derecho del eje

x , al igual que con el sistema tractor de tipo *car-like*. Los estados iniciales y finales son:

$$\begin{aligned} \text{Estado inicial} &= \begin{bmatrix} 0 & 20 & 0 & \theta & \theta_r \end{bmatrix}^T \\ \text{Estado final} &= \begin{bmatrix} 0 & 0 & 0 & \theta & \theta_r \end{bmatrix}^T \end{aligned}$$

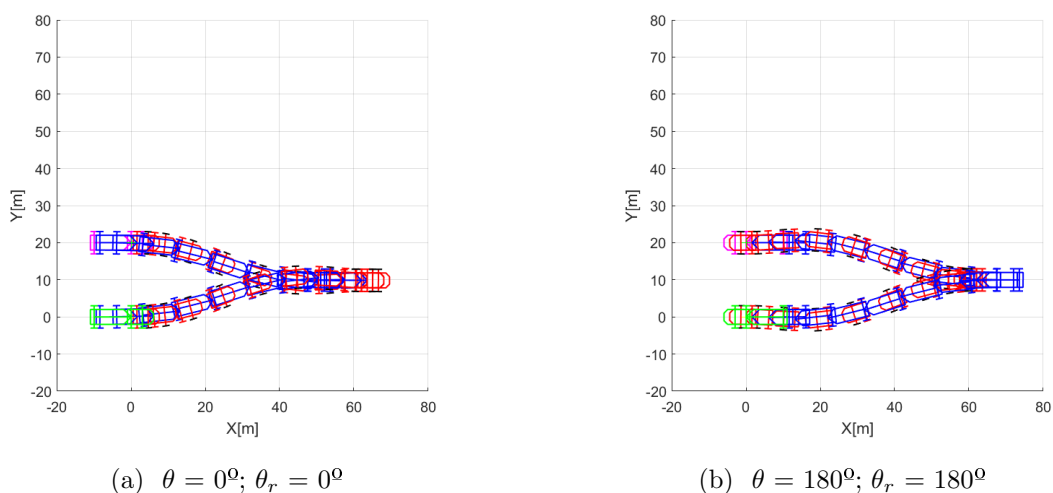


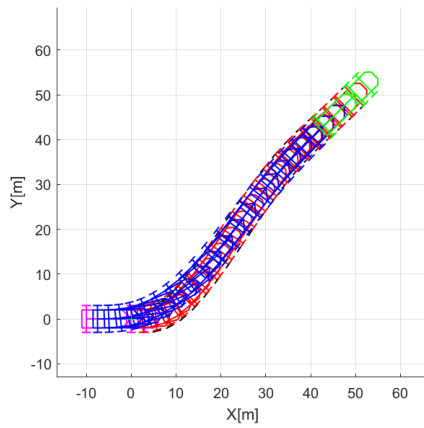
Figura 4.4: Comparación de maniobras paralelas con distintas orientaciones en un sistema tractor-remolque.

4.4.2. Experimento 2: maniobra de *pushback*

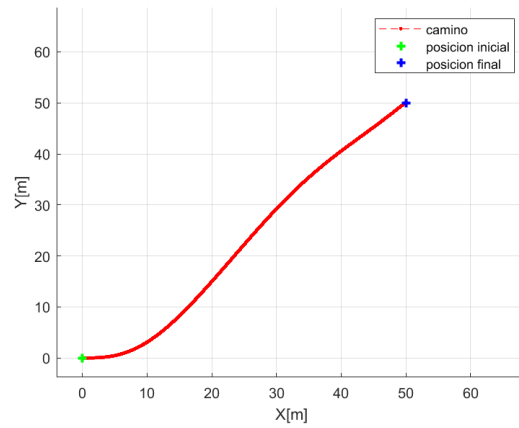
En el caso de una maniobra de *pushback*, se plantean 3 maniobras distintas, donde se demuestran las capacidades y dificultades que plantea este método aplicado a un sistema tractor-remolque.

La primera corresponde a un desplazamiento diagonal, que tiene los siguientes estados en la forma $(x, y, \phi, \theta, \theta_r)^T$:

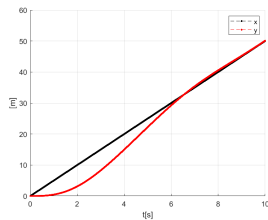
$$\begin{aligned} \text{Estado inicial} &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \\ \text{Estado final} &= \begin{bmatrix} 50 & 50 & 0 & \frac{\pi}{4} & \frac{\pi}{4} \end{bmatrix}^T \end{aligned}$$



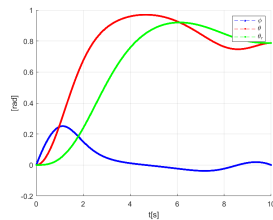
(a) Representación estroboscópica del sistema



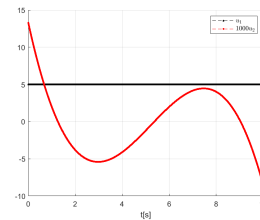
(b) Camino en el plano XY



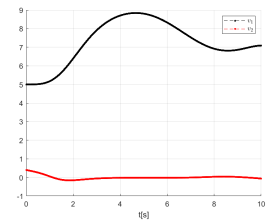
(c) Variables de estado X e Y



(d) Variables de orientación ϕ, θ, θ_r



(e) input U



(f) input V

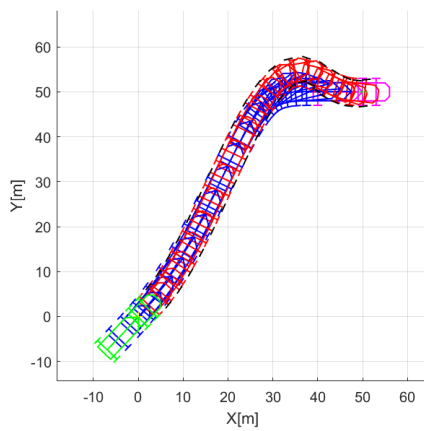
Figura 4.5: Maniobra de desplazamiento diagonal con un sistema tractor-remolque con *Polynomial Inputs*

El método de *steering* es capaz de llegar al estado final con una exactitud apropiada, ya que las posiciones finales x e y se alcanzan con un margen de 0,2m de media. La gráfica de los inputs u muestra lo suaves con son los inputs empleados en el método.

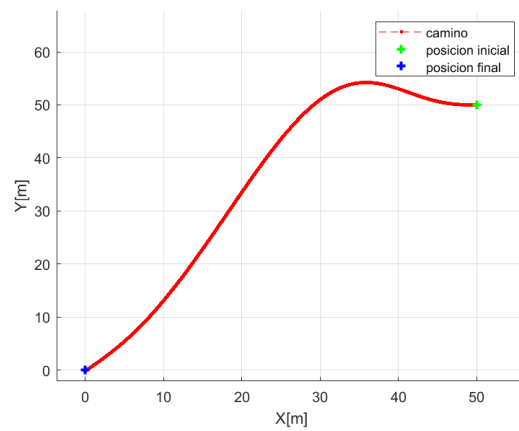
La siguiente maniobra corresponde a un desplazamiento marcha atrás, con los estados en la forma $(x, y, \phi, \theta, \theta_r)^T$:

$$\text{Estado inicial} = \begin{bmatrix} 50 & 50 & 0 & 0 & 0 \end{bmatrix}^T$$

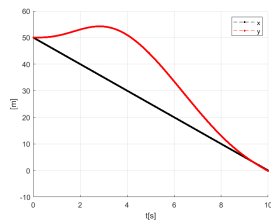
$$\text{Estado final} = \begin{bmatrix} 0 & 0 & 0 & \frac{\pi}{4} & \frac{\pi}{4} \end{bmatrix}^T$$



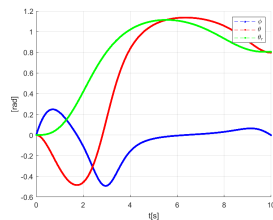
(a) Representación estroboscópica del sistema



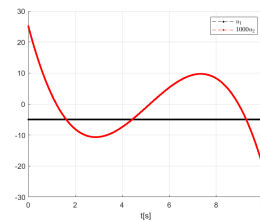
(b) Camino en el plano XY



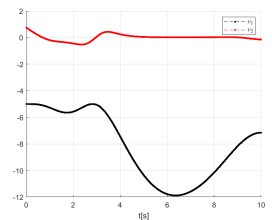
(c) Variables de estado X e Y



(d) Variables de orientación ϕ, θ, θ_r



(e) input U



(f) input V

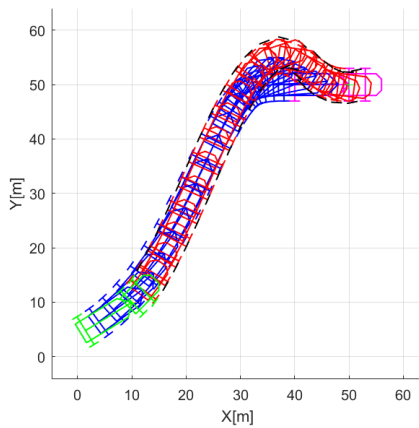
Figura 4.6: Maniobra de retroceso con *Polynomial Inputs*.

De nuevo, el sistema ha llegado a la posición indicada con un margen de error pequeño.

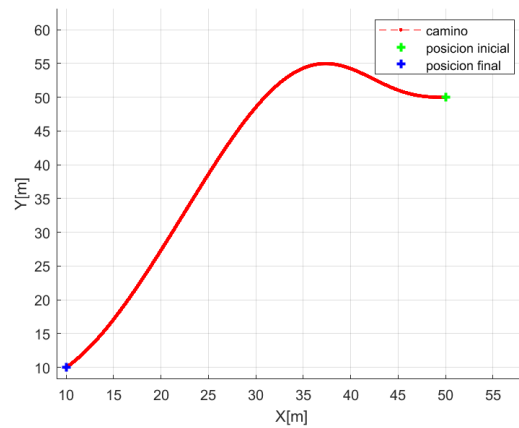
Para demostrar la robustez del método se establecerán dos estados bastante dispares entre sí. Esta maniobra será también de retroceso y tiene los siguientes estados iniciales y finales:

$$\text{Estado inicial} = \begin{bmatrix} 50 & 50 & 0 & 0 & 0 \end{bmatrix}^T$$

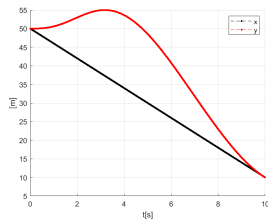
$$\text{Estado final} = \begin{bmatrix} 10 & 10 & \frac{10\pi}{4} & \frac{\pi}{4} & \frac{\pi}{6} \end{bmatrix}^T$$



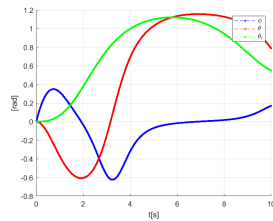
(a) Representación estroboscópica del sistema



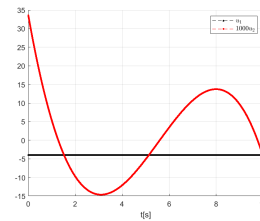
(b) Camino en el plano XY



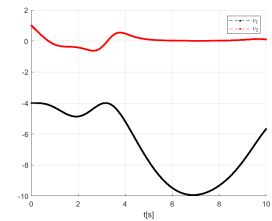
(c) Variables de estado X e Y



(d) Variables de orientación ϕ , θ , θ_r



(e) input U



(f) input V

Figura 4.7: Maniobra de desplazamiento cualquiera con un sistema tractor-remolque y *Polynomial Inputs*.

4.5. Singularidades

Este sistema esconde un problema que será fundamental resolver a la hora de establecer la maniobra final. Las ecuaciones de los inputs y las variables del vehículo en *chained form* 4.2 contienen denominadores sinusoidales que pueden ser anulados al introducir ciertos ángulos. Concretamente, varias de estas contienen cosenos en sus denominadores, por lo que si aparece algún ángulo con valor $\frac{k\pi}{2}$, aparecerán divisiones entre 0 y generarán indeterminaciones matemáticas que llevarán a un comportamiento indefinido del sistema. Por esta razón se deberá evitar para todos los casos la introducción de ángulos iguales o cercanos a los $\frac{k\pi}{2}rad$.

El gran problema que surge a continuación es que una de las maniobras para las que se está diseñando el *steering* es en esencia una rotación de $\frac{k\pi}{2}rad$: el *pushback*. Esta maniobra asume que deberá o bien comenzar o terminar en una posición que contenga este valor en al menos el ángulo θ , que corresponde con la rotación del coche respecto al escenario. Para diseñar la trayectoria final deberemos plantear alguna solución que evite pasar por este valor crítico, ya sea rotando el sistema o planteando varias rutas que al combinarse alcancen el estado final sin complicaciones.

A continuación se plantean 2 posibles soluciones que resuelven esta dificultad de manera similar para un problema concreto. Se propone una trayectoria que comienza en el estado $(x, y, \phi, \theta, \theta_r)^T = (50, 50, 0, 0, 0)^T$ y lleva al sistema hasta $(0, 0, 0, \frac{k\pi}{2}, \frac{k\pi}{2})^T$. Si introduyésemos directamente en el algoritmo estos valores, se produciría un comportamiento indeseado e indeterminado en el vehículo.

La siguiente figura muestra la maniobra propuesta para abordar las singularidades del sistema:

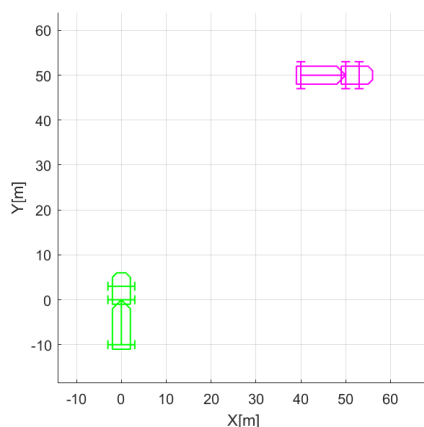


Figura 4.8: Maniobra propuesta para abordar las singularidades.

4.5.1. Solución 1: Planificación de trayectoria en 2 pasos

La primera solución propuesta plantea una separación de la maniobra en 2 o más tramos que al combinarse lleven al vehículo hasta la posición final. La idea fundamental de esta separación es que la sub-trayectoria que contenga el ángulo $\frac{\pi}{2}$ sea sustituida por una análoga en otra posición del espacio, que estará rotada $\frac{\pi}{2}rad$.

Primero, mediante una sucesión de pruebas, se estudiará el rango alrededor del ángulo crítico para el que el sistema tiene un comportamiento adecuado. En nuestro caso se ha comprobado como el sistema puede llegar a obtener movimientos correctos hasta unos $\frac{85\pi}{180}rad$, pero en este límite los desplazamientos comienzan a tomar rumbos poco intuitivos y por tanto poco útiles e inseguros. Por esta razón se decidirá escoger un ángulo máximo de $\frac{70\pi}{180}rad$ que asegure el comportamiento ideal del sistema. Una vez se ha establecido el límite práctico, podremos calcular el punto intermedio más cercano al que debe llegar el vehículo.

En el ejemplo sobre el que estamos trabajando, el tramo final es el que contiene el ángulo de $\frac{\pi}{2}rad$, por lo que trataremos de mover el remolque hasta una posición cercana al destino que esté inclinada $\frac{70\pi}{180}rad$. Una vez allí tendremos que ejecutar una segunda maniobra que gire el sistema $\frac{20\pi}{180}rad$ más y lo lleve a las coordenadas indicadas.

El estado intermedio que he escogido en este caso es $(x_F + 5, y_F + 10, 0, \frac{70\pi}{180}, \frac{70\pi}{180})^T$; las velocidades que habrá que tomar para alcanzar $(x_F, y_F, 0, \frac{\pi}{2}, \frac{\pi}{2})^T$ serán las mismas que las que llevan este mismo sistema de $(x_F + 5, y_F + 10, 0, \frac{-20\pi}{180}, \frac{-20\pi}{180})^T$ a $(x_F, y_F, 0, 0, 0)^T$. Podemos ver que esta transformación corresponde a un giro de 90° , por lo que bastará con rotar todo el sistema para averiguar el valor de las velocidades que necesitamos, todo

ello sin haber introducido como ángulo en ningún caso $\frac{\pi}{2}$.

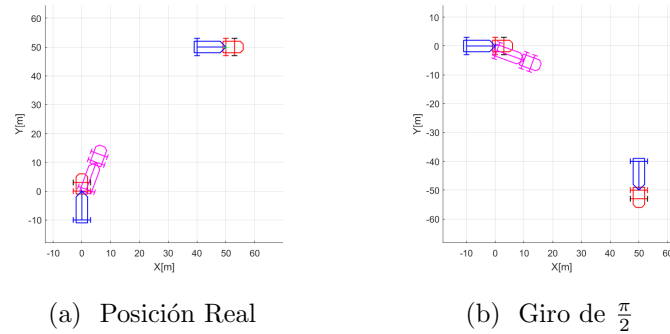


Figura 4.9: Comparación de maniobras análogas.

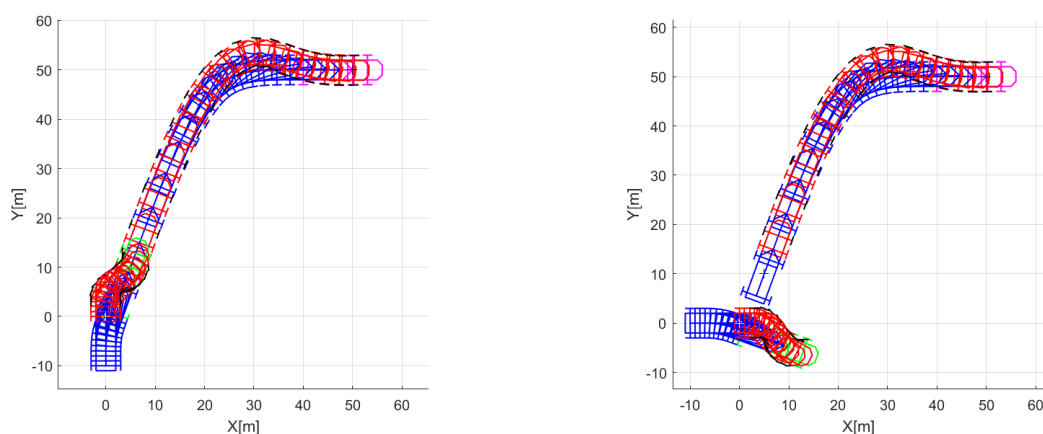
Para rotar el sistema se han hecho 2 cambios: a ϕ , θ y θ_r se les ha sumado $\frac{\pi}{2}$ y a las posiciones x e y se les ha aplicado una matriz de rotación de la forma:

$$R = \begin{pmatrix} \cos(\gamma) & -\sin(\gamma) & 0 \\ \sin(\gamma) & \cos(\gamma) & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

Cuando se introduzca en el algoritmo esta trayectoria auxiliar:

1. se rotará $\frac{\pi}{2}$ el sistema
2. se ejecutará el algoritmo con el sistema rotado, lo que no ocasionará singularidades
3. se mostrará la animación, rotada de nuevo $-\frac{\pi}{2}$

De esta manera visualizaremos el comportamiento real sin generar números infinitos ni imaginarios en el cálculo de las entradas del sistema.



(a) Trayectoria real del sistema sin singularidades

(b) Trayectoria usada por el algoritmo

Figura 4.10: Representación estroboscópica de trayectorias sin singularidades de un sistema tractor-remolque en 2 pasos.

Esta solución es válida para llegar a la posición final pero tiene el inconveniente de necesitar un punto intermedio que hay que estudiar con detenimiento tras una sucesión de pruebas y errores. Además, el hecho de que la maniobra no sea completa, hace que el movimiento no sea tan limpio ni continuo como se esperaría. Por otro lado, crear una pequeña sub-trayectoria y girar el sistema $\frac{\pi}{2}$ para evitar este mismo ángulo es una solución que puede ser trasladable a otras circunstancias que lo requieran y que podríamos tener en cuenta para otros problemas.

4.5.2. Solución 2: Rotación de todo el camino

Esta segunda opción plantea de igual manera aplicar una rotación al sistema, pero en este caso se hace a lo largo de todo el camino.

El movimiento que buscamos es un giro de $\frac{\pi}{2}$, por lo que queremos hacer una rotación de esta magnitud, pero dentro de otro rango; por ejemplo, cambiaremos el giro de $0rad$ a $\frac{\pi}{2}rad$ por uno de $-\frac{\pi}{4}rad$ a $\frac{\pi}{4}rad$.

Del mismo modo que en el caso anterior deberemos aplicar al estado correspondiente a cada instante una matriz de rotación. En este caso la magnitud de giro adecuada serán $\frac{\pi}{4}$. Se deberá tener en cuenta la restricción que tiene el propio método de *steering*, dónde se especifica que la posición x inicial y final debe ser distinta. En este caso, cuando giramos

$-\frac{\pi}{4}$ los estados del problema, encontramos que la x se iguala en el origen y destino, por lo que queremos hacer un giro de $\frac{\pi}{4}rad$ (en sentido positivo) para evitar que el control falle. En las figuras se pueden ver los estados inicial (verde) y final (rojo) de la maniobra original que contiene singularidades y los estados inicial (rosa) y final (azul) de la maniobra rotada cierto ángulo γ .

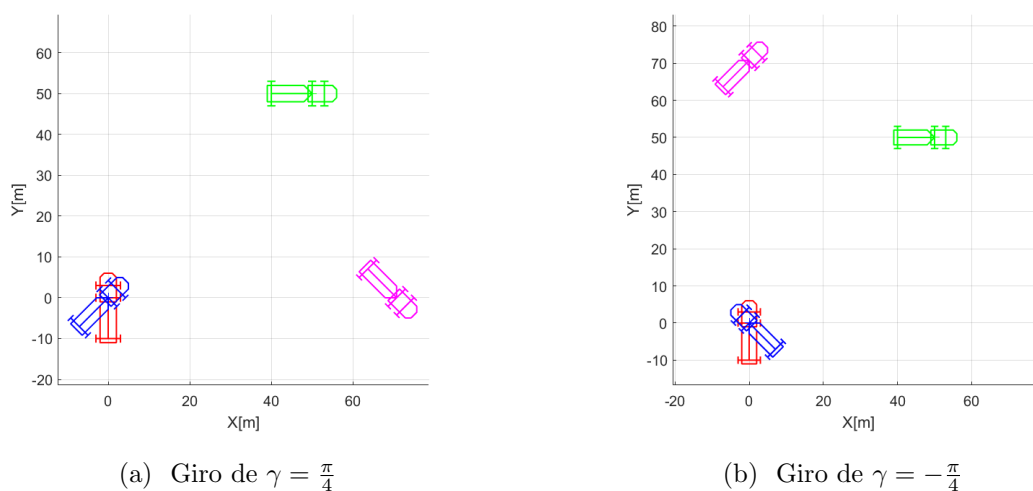


Figura 4.11: Comparación de rotaciones de $\gamma = |\frac{\pi}{4}|$.

De este modo se habrá generado un giro de 0 a $\frac{\pi}{2}$ sin alcanzar ninguna singularidad, además de conseguir que el movimiento sea continuo y sin interrupciones.

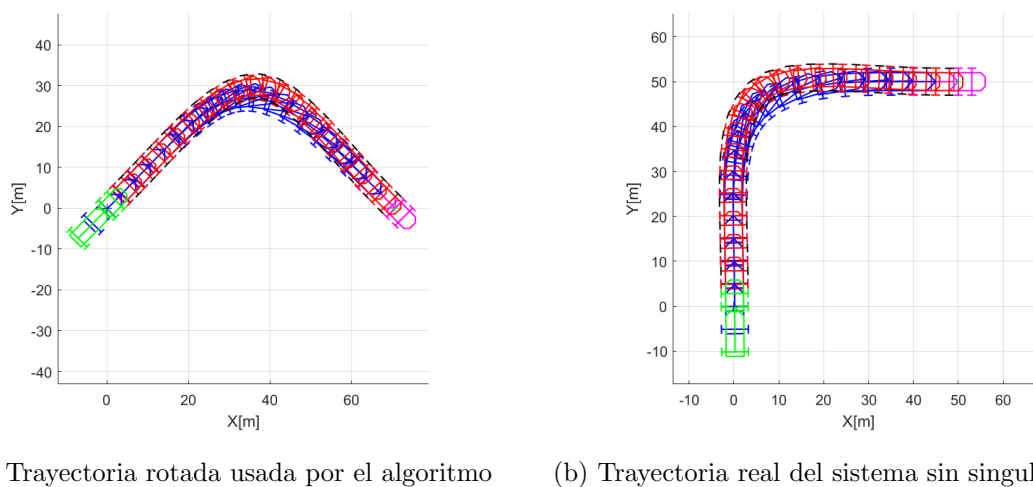


Figura 4.12: Representación estroboscópica de trayectorias sin singularidades de un sistema tractor-remolque rotado.

Capítulo 5

Planificación de maniobras para aeronaves en tierra

En capítulos anteriores se ha llegado a un control satisfactorio para el problema de *steering* de un vehículo articulado formado por un tractor con un remolque. En este capítulo se exponen varias trayectorias a propósito de una maniobra *pushback* y otras similares que pueden resultar útiles dentro de un aeropuerto.

5.1. Modelo un sistema tractor-avión

Para representar el sistema completo tractor-avión se han modificado los modelos que representaban al remolque y al tractor, y se han diseñado unos dibujos/sketches en función a ciertas medidas características de sus formas. El remolque ahora tendrá forma de avión y dependerá de medidas que se escogerán de fichas técnicas de un modelo de avión: el *Airbus A320*, muy utilizado en vuelos comerciales.

5.1.1. Modelo de un avión

El modelo del avión se basa en el modelo cinemático descrito en el capítulo anterior, dónde las medidas características son d y θ_r . Para representar un avión se necesitarán más medidas que caractericen la aeronave para así poder determinar por completo todas sus partes. Las medidas que se toman para hacer un sketch del avión son:

- a : la distancia entre las dos ruedas principales
- b : la distancia entre el tren de aterrizaje y el morro del avión

- c : el largo de la aeronave
- d : la distancia entre el tren de aterrizaje y las ruedas principales
- e : la envergadura de la aeronave
- f : la envergadura de los elevadores
- g : el ancho del fuselaje

Adicionalmente se tienen otras medidas útiles para tener una representación fiel de un avión como son varias distancias del ancho de las alas (i,j,k), que se pueden establecer arbitrariamente si no aparecen en los diagramas del avión a representar.

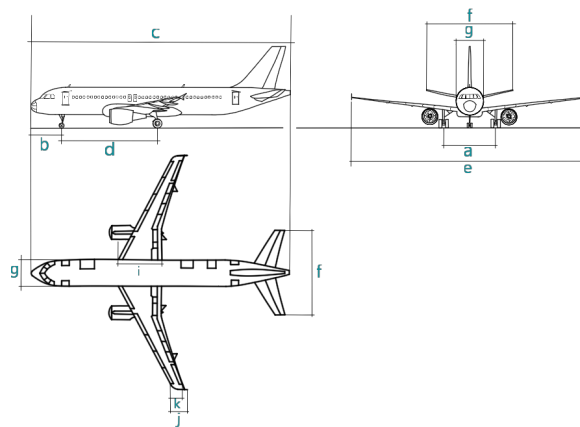


Figura 5.1: Medidas características para representar un avión.

El modelo que queremos representar en concreto es un *Airbus A320*, por lo que las medidas concretas que han de usarse son las siguientes:

Medidas	Valor (m)
a	7.59
b	5.07
c	37.57
d	12.64
e	35.8
f	12.45
g	3.95
i	6.07
j	3.22
k	1.64

Al introducir estos parámetros en la función `system_draw.f` que se ha creado en MATLAB, se obtiene el siguiente sketch que representa el avión:

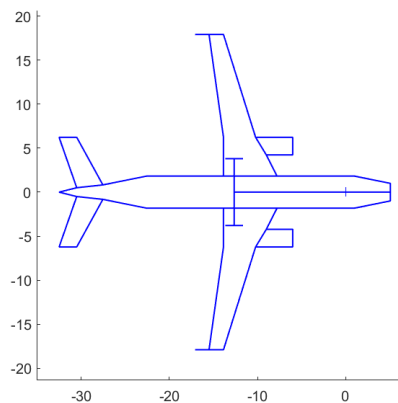


Figura 5.2: Sketch de un avión *Airbus A320*.

5.1.2. Modelo de un tractor

Respecto al modelo del tractor se ha preferido diseñar uno genérico, con una apariencia similar al modelo *car-like* propuesto en los apartados anteriores pero teniendo en cuenta el ancho y largo del vehículo. Se han omitido todas las medidas del modelo que no pertenecen al plano XY, por lo que el tractor en los diagramas es una plataforma plana. En este caso las medidas que caracterizan al tractor son:

- l : la distancia entre las ruedas delanteras y traseras
- m : el ancho del tractor
- n : el largo del tractor
- o : ancho del hueco dónde se ancla en tren de aterrizaje
- p : la distancia entre las ruedas de la derecha y la izquierda
- r : el radio máximo de las ruedas

Como las ruedas directrices del tractor son muy pequeñas respecto al tamaño del conjunto del sistema tractor-avión, también se dibujará una línea adicional que muestre el ángulo ϕ que representa la dirección de las mismas. Así se tendrá una representación completa y visual de todas las variables.

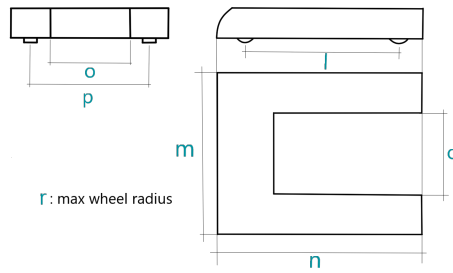


Figura 5.3: Medidas características de un tractor.

Las medidas que se han escogido para representar el tractor de este sistema han sido arbitrarias, teniendo en cuenta únicamente que las magnitudes sean proporcionales al tamaño del avión. Estas han sido:

Medidas	Valor (m)
l	7
m	3
n	10
o	1
p	2
r	0.5

Con estas medidas se ha generado el siguiente sketch:

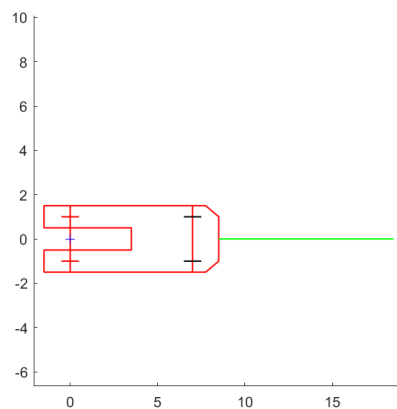


Figura 5.4: Sketch de un tractor.

5.1.3. Modelo completo tractor-avión

Ambos sketches conforman el sistema completo y tendrán el siguiente aspecto:

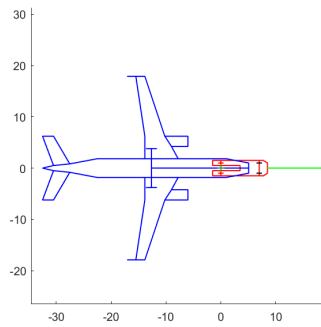


Figura 5.5: Sketch de un sistema tractor-avión completo.

5.2. Resultados

A continuación se detallan los resultados de la aplicación de los métodos estudiados en anteriores capítulos a un sistema tractor-avión. En concreto se harán 2 experimentos: una maniobra *pushback* y una maniobra de desplazamiento en paralelo.

Los escenarios para cada una de estas maniobras han sido adaptados a MATLAB en base a imágenes tomadas por satélite ¹ del aeropuerto Adolfo Suarez Madrid-Barajas (LEMD). Con estas fotografías se han introducido las distancias entre *aprons*, tamaño de las pistas y separación entre aviones.

Al igual que en los capítulos anteriores, en las figuras aparecen: una representación estroboscópica de la animación del sistema en el plano XY, donde se dibujan el estado inicial de color rosa y el final de color verde; una representación del camino que ha llevado el sistema en el plano XY, donde se dibuja el estado inicial con una cruz verde y el final con una cruz azul; y los valores que han tomado las variables del sistema a lo largo del tiempo $T = 10s$, que son los valores de x , y , θ , ϕ , u y v . Todas las distancias están representadas en metros y los ángulos en radianes.

5.2.1. Experimento 1: maniobra de *pushback*

La primera maniobra es la de *pushback*. Esta maniobra mueve el sistema desde su posición en la plataforma de desembarco a una en la pista de *taxiing*, todo ello marcha atrás. El método que se utilizará en este experimento es *Polynomial Inputs*, que anteriormente habíamos modificado para generar rotaciones de $\frac{\pi}{2}rad$.

En las figuras que representan la animación del sistema, el estado inicial está pintado de magenta. Los estados inicial y final de esta maniobra en la forma $(x, y, \phi, \theta, \theta_r)^T$ son los siguientes:

$$\begin{aligned} \text{Estado inicial} &= \begin{bmatrix} 87,23 & 38 & 0 & 0 & 0 \end{bmatrix}^T \\ \text{Estado final} &= \begin{bmatrix} 0 & 0 & 0 & \frac{\pi}{2} & \frac{\pi}{2} \end{bmatrix}^T \end{aligned}$$

¹<https://www.google.es/intl/es/earth/index.html>

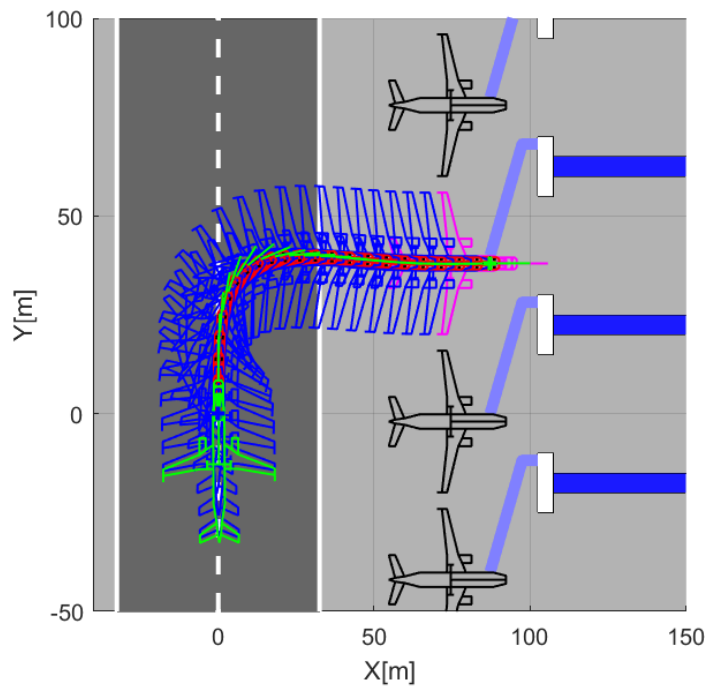
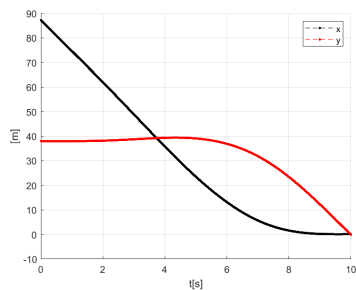
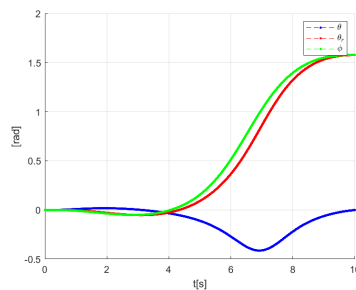


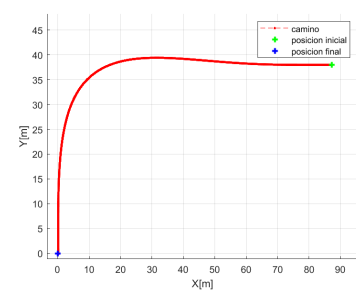
Figura 5.6: Movimiento estroboscópico de una maniobra *pushback*.



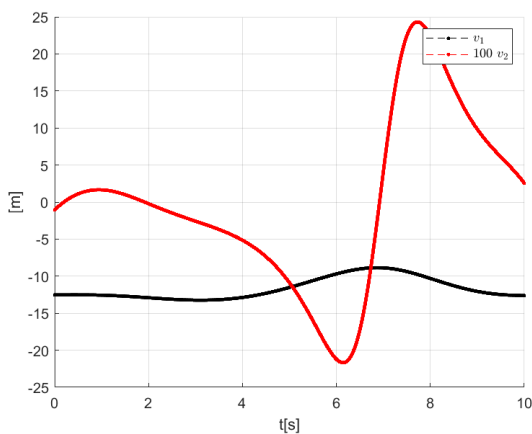
(a) Variables de estado X e Y en el tiempo



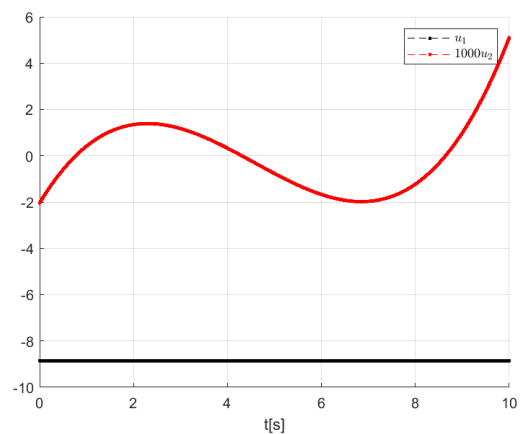
(b) Variables de orientación θ , θ_r , y ϕ



(c) Camino en el plano XY



(d) input V



(e) input U

Figura 5.7: Experimento 1: Variables de una maniobra *pushback*.

La figura representa de forma estroboscópica el movimiento generado por el control *Polynomial Inputs* en el sistema tractor-avión. De esta manera se muestra un dibujo del sistema cada cierto tiempo, pudiendo ver el movimiento que ha llevado durante toda la trayectoria. Se puede ver cada punto de colisión por el que pasa la aeronave, donde podemos apreciar que se genera una trayectoria limpia sin peligro de choque con los elementos del escenario. En este caso el método de *steering* no permite modificar parámetros para cambiar el comportamiento de la trayectoria.

5.2.2. Experimento 2: maniobra de desplazamiento paralelo

Este movimiento se utiliza a menudo en aeropuertos para hacer maniobras de *parking* y así mover aviones desde unas plataformas de aparcamiento a otras. En esta maniobra el avión está alejado de los *aprons*, por lo que los obstáculos más significativos son los aviones al rededor del objetivo.

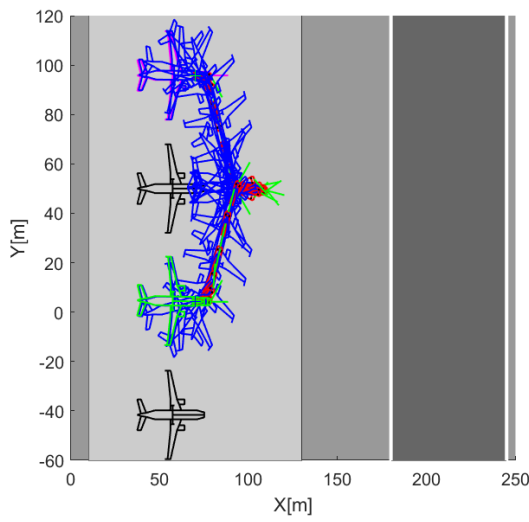
En este caso el experimento utiliza el método *Asymptotic Sinusoidal Inputs* para generar una trayectoria que mueve el avión de forma paralela. Los estados inicial y final en la forma $(x, y, \phi, \theta, \theta_r)^T$ de la maniobra son:

$$\begin{aligned} \text{Estado inicial} &= \begin{bmatrix} 70 & 95,8 & 0 & 0 & 0 \end{bmatrix}^T \\ \text{Estado final} &= \begin{bmatrix} 70 & 4,2 & 0 & \frac{\pi}{2} & \frac{\pi}{2} \end{bmatrix}^T \end{aligned}$$

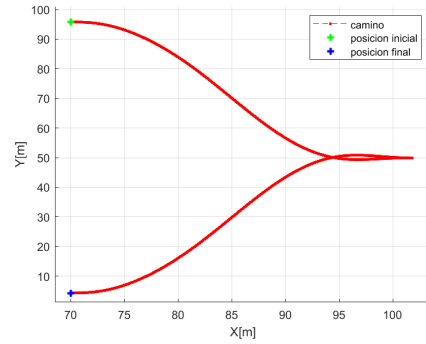
Como se ha visto en la aplicación teórica de este método para un sistema de un vehículo tractor con un remolque 4.4.1, la única dirección en la que mueve el sistema de forma paralela es inicialmente hacia la derecha y después hacia la izquierda. Por esta razón deberemos establecer el escenario de forma que los aviones queden a la izquierda de la pista de *taxiing* y a su vez a las puertas de embarque.

Por otro lado se deberá modificar el parámetro a_1 para conseguir que el sistema esquive correctamente los aviones que se interpongan en el camino del sistema. Este parámetro permitirá hacer más o menos larga la distancia que avanza hacia delante (en nuestro caso, desde el punto de vista del escenario, el avión avanzará hacia la derecha).

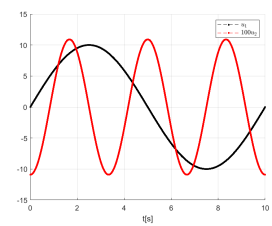
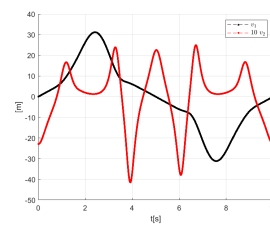
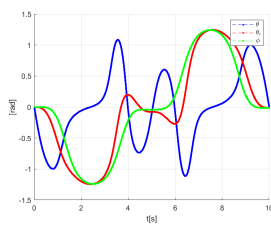
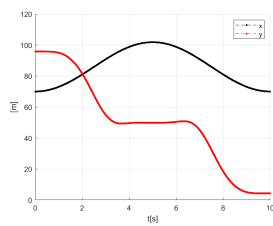
Inicialmente se prueba con un $a_1 = 10$, que no consigue un resultado satisfactorio ya que generaría una colisión con el avión mas cercano [1]:



(a) Animación estroboscópica



(b) Camino en el plano XY



(c) Variables de estado X e Y en el tiempo

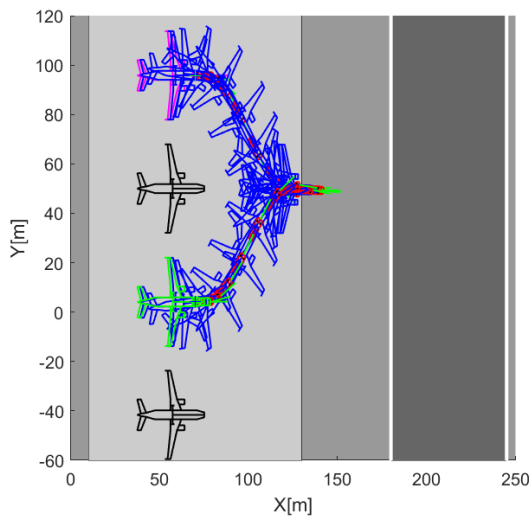
(d) Variables de orientación θ , θ_r y ϕ

(e) input V

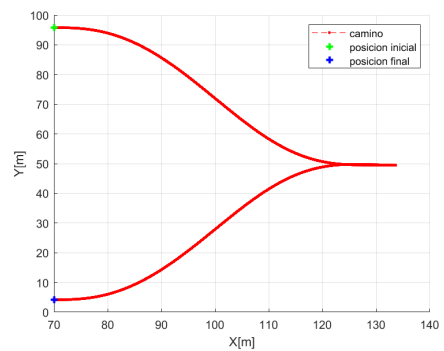
(f) input U

Figura 5.8: Experimento 2: variables de una maniobra de *parking* con parámetro $a_1 = 10$.

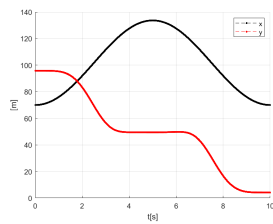
Si modificamos este parámetro para que consiga alejarse del avión y a su vez no se aparte demasiado de la pista de *parking* hallaremos un resultado más adecuado:



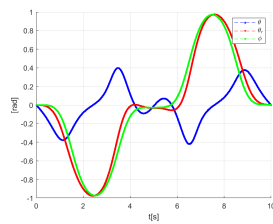
(a) Animación estroboscópica



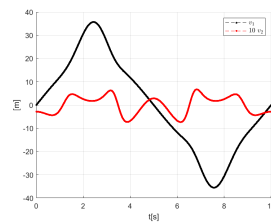
(b) Camino en el plano XY



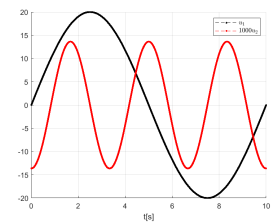
(c) Variables de estado X e Y en el tiempo



(d) Variables de orientación θ , θ_r y ϕ



(e) input V



(f) input U

Figura 5.9: Experimento 2: variables de una maniobra de *parking* con parámetro $a_1 = 20$.

Capítulo 6

Conclusiones

En este trabajo de fin de grado se ha estudiado el problema de la planificación de movimientos de aeronaves en tierra a través de tractores robóticos sin barra de remolque.

Primero se ha definido el modelo cinemático que representa el sistema tractor-avión. El modelo propuesto es un robot tipo *car-like* con un remolque. Dado que los métodos de planificación para un sistema tractor-remolque presentes en la literatura se basan en la llamada *chained form*, se ha estudiado la transformación del modelo a esta forma. Posteriormente se han estudiado los métodos de *steering*/planificación del sistema tractor-remolque en *chained form*, para los que se han seleccionado los procedimientos: *Asymptotic Sinusoidal Inputs* y *Polynomial Inputs*. Finalmente se han implementado estos dos métodos para la planificación de trayectorias en maniobras de aviones típicas de un aeropuerto: el *pushback* y el desplazamiento en paralelo. En estos últimos experimentos se ha usado un sistema con un tractor y un avión *Airbus A320*.

Tras el estudio de las simulaciones de estos métodos de *steering* se ha podido concluir con que *Polynomial Inputs* ha permitido crear trayectorias que consiguen mover marcha atrás un sistema tractor-remolque de un punto a otro, pudiendo aplicarse así a maniobras de *pushback* para aviones en tierra. Por otro lado *Asymptotic Sinusoidal Inputs* ha conseguido generar trayectorias correctas aplicables a una maniobra de parking o desplazamiento paralelo en un sistema de un tractor *car-like* con un remolque. Este procedimiento permite además ajustar las trayectorias que genera para poder así esquivar obstáculos o suavizar sus giros.

Los métodos de *steering* son procedimientos de planificación que calculan la trayectoria que debe llevar un sistema de un estado inicial a uno final. En este tipo de procedimientos no se tienen en cuenta los posibles obstáculos que puedan bloquear el movimiento generado. A pesar de los buenos resultados, usando herramientas avanzadas de simulación, estos métodos de *steering* no se pueden aplicar a modelos reales directamente. Además

hay que resolver el problema del seguimiento de las trayectorias planificadas, que no se ha estudiado en este Trabajo de Fin de Grado. Este es un aspecto importante debido al fenómeno del *jack-knifing*, que ocurre cuando un sistema tractor-remolque es empujado hacia atrás y deja al mismo inmovilizado.

Como posibles continuaciones de este trabajo se podrían abordar estos dos últimos aspectos: la planificación de trayectorias para un sistema tractor-avión con evasión de obstáculos y el diseño del control de seguimiento de las trayectorias generadas en este trabajo.

Bibliografía

- [1] Franziska Dieke-Meier y Hartmut Fricke. The need for a collision prevention system for the pushback of aircraft to minimize hazards caused by narrow spaces. En *28th International Congress of Aeronautical Sciences. Brisbane, Australia*. September 23-28, 2012.
- [2] Franziska Dieke-Meier, Thomas Kalms, Hartmut Fricke, y Michael Schultz. Modeling aircraft pushback trajectories for safe operations. Inf. téc., Chair of Air Transport Technology and Logistics Technische Universität Dresden, 2013.
- [3] Fan Hong, Shuzhi Sam Ge, Chee Khiang Pang, y Tong Heng Lee. Robust adaptive neuro-fuzzy control of uncertain nonholonomic systems. *Journal of Control Theory and Applications*, Vol.8(Number 2):125–138, 2010.
- [4] Zhong Ping Jiang y Henk Nijmeijer. A recursive technique for tracking control of nonholonomic systems in chained form. *IEEE Transactions on Automatic Control*, Vol. 44(Number 2):265–279, 1999.
- [5] Sergio Llopis Juan. *Optimización de plataformas y puertas de embarque en aeropuertos*. Tesis Doctoral, Escuela Técnica Superior de Ingeniería del Diseño de Valencia, 2021.
- [6] Oskar Ljungqvist, Niclas Evested, Daniel Axehill, y Marcello Cirillo and Henrik Pettersson. A path planning and path following control framework for a general 2-trailer with a car-like tractor. Inf. téc., Department of Automatic Control, Linköping University, Linköping, Sweden and Embark Trucks Inc. San Francisco, USA, 2019.
- [7] A. De Luca, G. Oriolo, y C. Samson. *Robot motion planning and control*, cap. 4: Feedback Control of a Nonholonomic Car-like Robot, págs. 171–250. 1998. ISBN 3-540-76219-1.

-
- [8] Richard M. Murray y S. Shankar Sastry. Steering nonholonomic systems in chained form. En *30th Conference on Decision and Control. Brighton, England*. December 11-13, 1991.
- [9] D. Tilbury, J-P. Laumond, R. Murray, S. Sastry, y G. Walsh. Steering car-like systems with trailers using sinusoids. En *IEEE International Conference on Robotics and Automation. Nice, France*. May 12-14,1992.
- [10] Liang Zhen-Ying y Wang Chao-Li. Robust stabilization of nonholonomic chained form systems with uncertainties. *Acta Automatica Sinica*, Vol. 37(Number 2):129 – 142, 2011.