

MODELO DE SIMULACIÓN BASADO EN AGENTES PARA LA MEJORA DEL SISTEMA TCAS

Memòria del Treball Fi de Grau
Gestió Aeronàutica
realitzat per
Stanislav Ilkov Pergelov
i dirigit per
Miquel Àngel Piera Eroles
Sabadell, Julio de 2016

El sotasignat, **Miquel Àngel Piera Eroles**,
professor de l'Escola d'Enginyeria de la UAB,

CERTIFICA:

Que el treball a què correspon aquesta memòria ha estat realitzat sota la seva supervisió per en **Stanislav Ilkov Pergelov**

I per tal que consti firma la present.

Signat:

Sabadell,de.....de 2016

FULL DE RESUM – TREBALL FI DE GRAU DE L'ESCOLA D'ENGINYERIA

Títol del Treball Fi de Grau (obligatori en tres idiomes: Català, Castellà, Anglès)	
Modelo de simulación basado en agentes para la mejora del sistema TCAS	
Autor[a]: Stanislav Ilkov Pergelov	Data: <i>Julio, 2016</i>
Tutor[a]/s[es]: Miquel Àngel Piera Eroles	
Titulació: Gestió Aeronàutica	
Paraules clau (mínim 3)	
<ul style="list-style-type: none">• Català: Sistema Multi-Agent, conflicte, detecció, evasió, col·lisió, Netlogo, aeronau• Castellà: Sistema Multi-Agente, conflicto, detección, evación, colisión, Netlogo, aeronave• Anglès: Multi Agent System, conflicte, detection, avoidance, collision, Netlogo, airplane	
Resum del Treball Fi de Grau (extensió màxima 100 paraules)	
<ul style="list-style-type: none">• Català: Aquest projecte explorarà la introducció de les Sistemes Multi-Agents (MAS) en la detecció i resolució de conflictes amb el objectiu de millorar la seguretat aèria i la coordinació entre les aeronaus en el espai aeri. Es pretén desenvolupar una simulació basada en agents per avaluar la eficàcia i millorar el sistema TCAS considerant un escenari real o sintètic de tràfic en conflicte. Per resoldre aquest conflicte es tindran en consideració diferents kpi's de les companyies aèries involucrades.• Castellà: Este proyecto explorará la introducción de los Sistemas Multi-Agente (MAS) en la detección y resolución de conflictos con el objetivo de mejorar la seguridad aérea y la coordinación entre las aeronaves en el espacio aéreo. Se pretende desarrollar una simulación basada en agentes para evaluar la eficacia y mejorar el sistema TCAS considerando un escenario real o sintético de tráfico en conflicto. Para resolver este conflicto se tendrán en cuenta diferentes kpi's de las compañías aéreas involucradas.• Anglès: This project will explore the introduction of Multi-Agent System (MAS) in the detection and resolution of conflicts in order to improve aviation safety and coordination between aircraft in airspace . It aims to develop an agent-based simulation to evaluate the effectiveness and improve the TCAS system considering a real or synthetic traffic scenario in conflict. To resolve this conflict different kpi 's of the involved airlines will be considered.	

Índice

Capítulo 1	1
1.1 Introducción	3
1.2 Motivación.....	7
1.3 Objetivos	10
1.3.1. Objetivo principal.....	10
1.3.2. Objetivos específicos	10
1.4 Estudio de viabilidad.....	11
1.5 Netlogo	12
Capítulo 2. Estado del arte.....	15
2.1. TCAS	17
2.2. Modelado basado en agentes.....	19
Capítulo 3. Modelo MAS de un nuevo TCAS.....	23
3.1 Escenario considerado	25
3.2 Descripción de Agentes y variables.....	26
3.3 Flujo del modelo	29
3.3.1 Algoritmo de detección	30
3.3.2 Actualización del algoritmo de detección	32
3.3.3 Validación del algoritmo de detección	33
3.3.4 Negociación.....	35
3.3.5 Recuperación de la trayectoria inicial – procedimiento Volver	35
Capítulo 4. Resultados obtenidos	37
4.1 Escenario 1.....	39
4.2 Escenario 2.....	41
4.3 Escenario 3.....	42
Capítulo 5. Conclusiones y líneas de trabajo futuras	45
5.1. Conclusiones	47
5.2. Líneas futuras de trabajo.....	47
Bibliografía	51

Índice de figuras

Figura 1. Horas de vuelo anuales vs ratio de colisiones.	3
Figura 2. Separación estándar.	4
Figura 3. Criterios de vigilancia de TCAS.	5
Figura 4. Modelo conceptual de TCAS.....	6
Figura 5. Escenario de colisión inducida entre 4 aeronaves.....	7
Figura 6. Crecimiento anual por estado (2022 vs 2015).....	7
Figura 7. Número de vuelos extra por día (2035 vs 2012).....	8
Figura 8. Captura del escenario inicial	26
Figura 9. Monitor de turtles con la simulación ejecutada	28
Figura 10. Variables globales	29
Figura 11. Captura de la ventana de ejecución del modelo	30
Figura 12. Código del algoritmo de detección.	31
Figura 13. Agentes generados para la detección de intrusos.	32
Figura 14. Captura del escenario 1 de la validación del algoritmo de detección.....	33
Figura 15. Captura del escenario 1 de la validación del algoritmo de detección.....	34
Figura 16. Captura de la simulación del escenario 1.....	40
Figura 17. Captura del escenario 1 con las trayectorias libres de conflicto.....	40
Figura 18. Captura de la simulación del escenario 2.....	41
Figura 19. Captura del escenario 2 con las trayectorias libres de conflicto.....	42
Figura 20. Captura de la simulación del escenario 3.....	43
Figura 21. Captura de las trayectorias del escenario 3	43
Figura 22. CPU time vs distancia de detección.....	48

Índice de tablas

Tabla 1. Tabla 1. Criterios de TCAS para TA y RA	6
Tabla 2. Catalogación de los riesgos	12
Tabla 3. Plan de contingencia.....	12

Capítulo 1

- 1.1 Introducción
- 1.2 Motivación
- 1.3 Objetivos
 - 1.3.1 Objetivo principal
 - 1.3.2 Objetivos específicos
- 1.4 Estudio de viabilidad
- 1.5 Netlogo

Capítulo 1. Introducción

1.1 Introducción

Cuando ocurre un accidente de tráfico aéreo las consecuencias suelen ser catastróficas. Raras veces hay supervivientes y el impacto social y económico es enorme. Sin embargo el transporte aéreo sigue siendo el más seguro comparado con el resto de tipos de transporte. Como se puede ver en la Figura 1 el número de horas de vuelo se ha cuadruplicado en las últimas décadas, sin embargo el ratio de accidentes aéreos ha disminuido considerablemente.

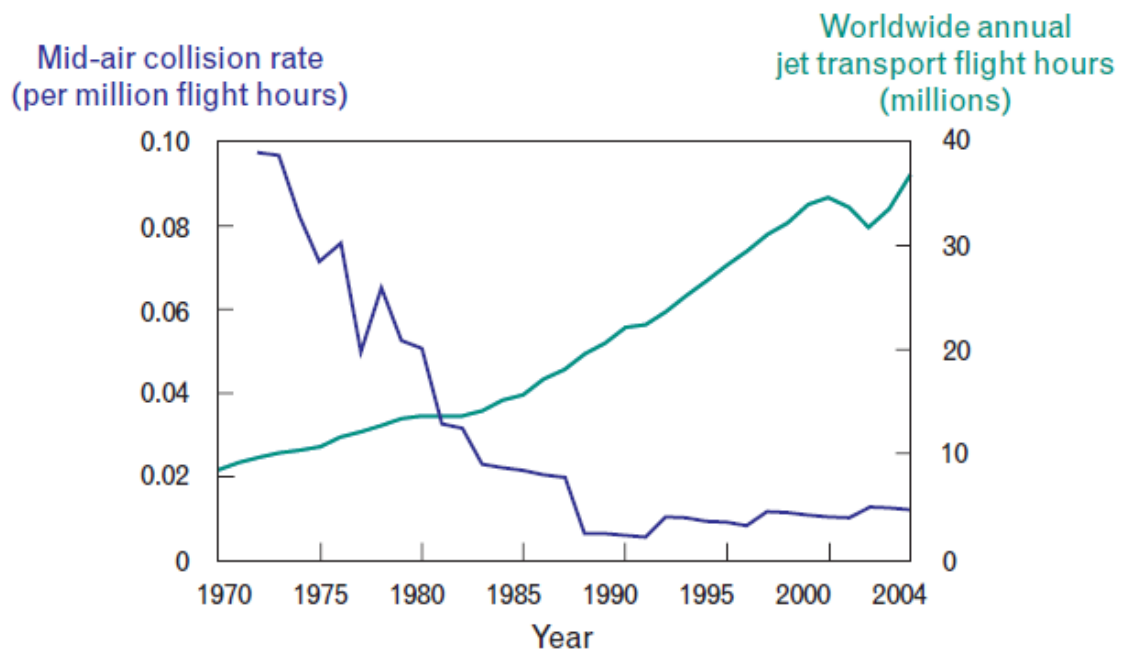


Figura 1. Horas de vuelo anuales vs ratio de colisiones. Fuente [15]

Para prevenir que ocurran este tipo de accidentes el espacio aéreo está organizado de manera que queda protegido a través de varias capas de seguridad. El flujo del tráfico está organizado por aerovías con una separación vertical de mil pies y una separación lateral de cinco millas náuticas. Los controladores aéreos aseguran que las separaciones sean respetadas guiando los pilotos con suficiente antelación. TCAS es un componente de la seguridad aérea que actúa cuando los demás han fallado.

TCAS (*Traffic Alert and Collision Avoidance System* en inglés) también conocido como el Sistema anticollisión de a bordo (ACAS) es un sistema diseñado para alertar a los pilotos sobre el tráfico cercano y es la última capa

de seguridad contra las colisiones en el aire. El sistema monitorea el espacio aéreo alrededor para detectar otras aeronaves que pueden presentar una amenaza. TCAS opera independientemente del equipamiento terrestre y proporciona a los pilotos orientación sobre la forma de evitar un posible conflicto. En este contexto consideramos que dos o más aeronaves están en conflicto cuando se pierde la separación arriba mencionada. La Figura 2 muestra la zona de seguridad que no debe ser violada.

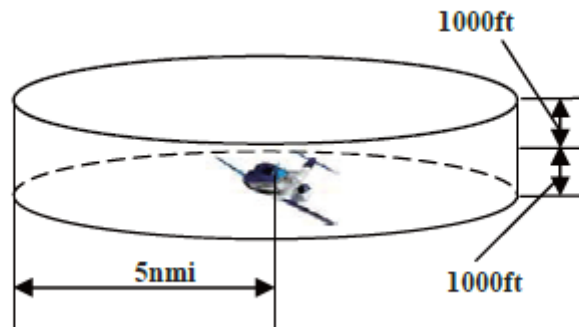


Figura 2. Separación estándar. (Fuente [14])

Para su funcionamiento es necesario que las aeronaves estén equipadas con *transponders* que respondan a las interrogaciones del radar. El TCAS usa a los *transponders* y antenas propias para poder ubicar una aeronave cercana, analizar si representa peligro y, dependiendo del modo de *transponder* (A, C o S) que se esté usando, poder coordinar una resolución conjunta de una colisión. Una vez que el procesador identifica un problema, sigue al intruso, evalúa el riesgo para la aeronave y de manera coordinada da una solución para resolver el conflicto. Basado en la posición, altitud y distancia de los demás aviones, TCAS clasifica a las aeronaves intrusas como: Non Threat, Proximity, Traffic Advisory (TA), Resolution Advisory (RA).

El algoritmo de detección de colisiones de TCAS consiste en proyectar la posición de la aeronave en el futuro e identificar posibles conflictos basándose en varios parámetros como separación horizontal y vertical y tiempo hasta el punto de encuentro. Es necesario que ambas aeronaves estén equipadas con TCAS II para poder coordinar una resolución (RA). En la Figura 3 se pueden ver los criterios de vigilancia del TCAS y el tiempo en segundos en el cual se da un aviso de tráfico (TA) o una resolución de conflicto (RA).

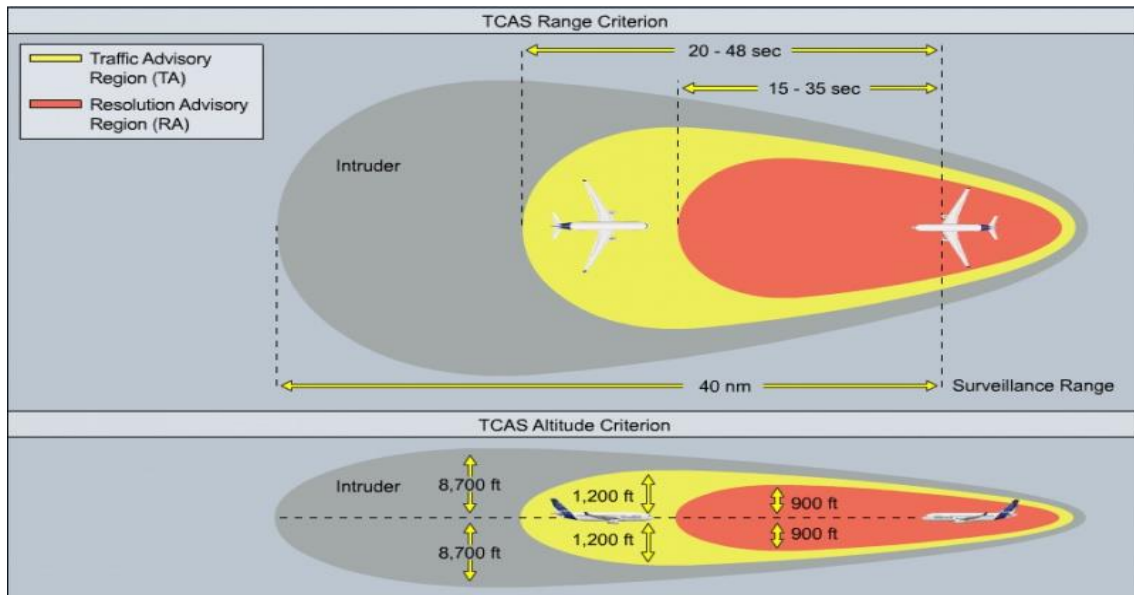


Figura 3. Criterios de vigilancia de TCAS. Fuente (1)

Si existe una amenaza en la zona TA (20-48 segundos dependiendo de la altitud) se dispara un aviso acústico en cabina. Así, los pilotos están avisados del posible conflicto y pueden hacer una búsqueda visual y prepararse para una maniobra de evasión.

En cambio si el intruso está a (15 – 35 segundos dependiendo de la altitud) de la colisión se dispara un RA con un aviso acústico tipo *Climb, Climb* indicando la maniobra necesaria para evitar el encuentro. La maniobra de evasión recomendada es coordinada entre las dos aeronaves. Los pilotos deben reaccionar inmediatamente ante un RA para conseguir la separación mínima necesaria.

Cuando la amenaza haya pasado TCAS avisa a ambas aeronaves con un mensaje que están libres de conflicto.

En la Figura 4 la aeronave i recibe un RA de descenso mientras que la aeronave j recibe orden de ascenso. De esta manera se consigue la separación mínima en altitud (ALIM) en el CPA entre las dos aeronaves.

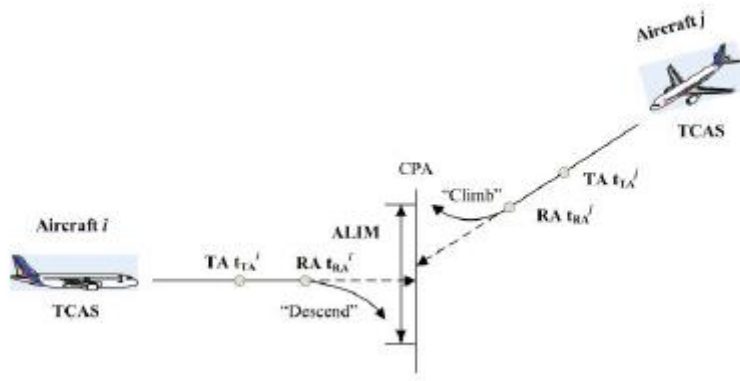


Figura 4. Modelo conceptual de TCAS. Fuente [13]

En la Tabla 1 se muestran los criterios para TA y RA.

Own altitude (feet)	Tau (seconds)		DMOD (nmi)		Altitude Threshold (feet)	
	TA	RA	TA	RA	TA	RA(ALIM)
< 1000	20	N/A	0.30	N/A	850	N/A
1000 ~ 2350	25	15	0.33	0.20	850	300
2350 ~ 5000	30	20	0.48	0.35	850	300
5000 ~ 10000	40	25	0.75	0.55	850	350
10000 ~ 20000	45	30	1.00	0.80	850	400
20000 ~ 42000	48	35	1.30	1.10	850	600
> 42000	48	35	1.30	1.10	1200	700

Tabla 1. Tabla 1. Criterios de TCAS para TA y RA

Cuando el conflicto es entre más de dos aeronaves, TCAS resuelve los conflictos indicando una maniobra coordinada (RA) que proporciona la separación necesaria a cada pareja en conflicto sin considerar el resto de tráfico alrededor. Resolviendo de esta manera es posible que la RA indicada evite la colisión con la aeronave en conflicto pero al mismo tiempo genere una colisión inducida con otra aeronave. Para evitar estas colisiones inducidas es imprescindible tener en consideración el tráfico colindante.

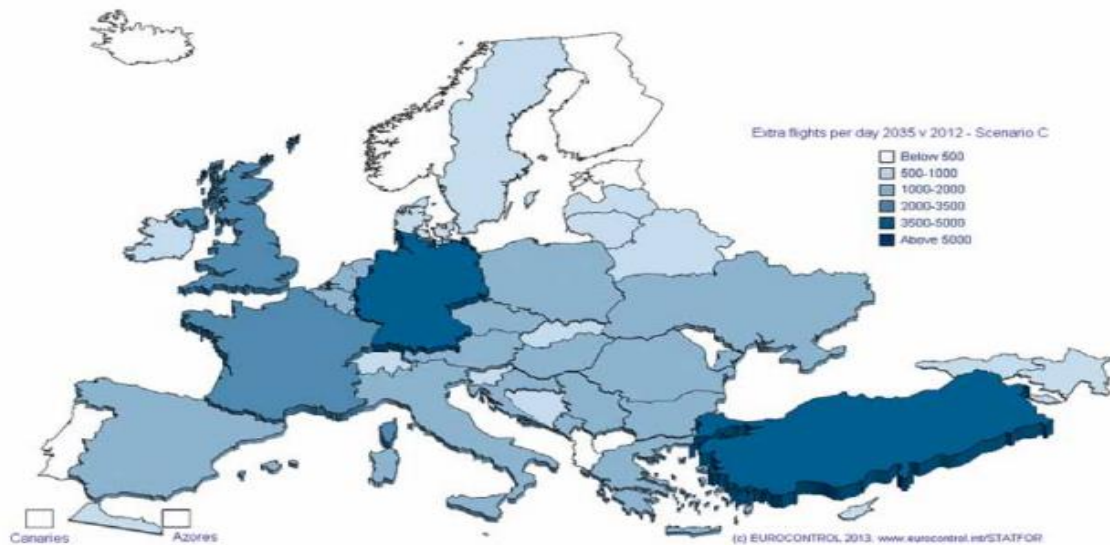


Figura 7. Número de vuelos extra por día (2035 vs 2012). Fuente (Eurocontrol 2016)

Por otro lado en la Figura 7 se observa el crecimiento de vuelos extra por día a largo plazo (2012 - 2035). Se espera que los vuelos pasen de 10 a cerca de 17 millones al año. El número máximo de vuelos en un día actualmente está en 33 000 con una previsión de alcanzar los 50 000 en el año 2030.

Para poder dar respuesta a esta futura demanda se están desarrollando nuevos conceptos en ATM (*Air Traffic Management*).

SES (*Single European Sky*) es un programa propuesto por Comisión Europea para eliminar las fronteras en el espacio aéreo entre los países. Los objetivos más importantes de este programa son:

- Incrementar la capacidad del espacio aéreo
- Reestructurar el espacio aéreo
- Incrementar la eficiencia del ATM

Para conseguir los objetivos del SES se ha puesto en marcha el programa SESAR (*Single European Sky ATM Research*). Es la dimensión tecnológica del SES que coordina la investigación y el desarrollo de los mecanismos necesarios para conseguir los objetivos arriba mencionados.

Otro concepto relacionado es *Free Route* que ofrece a los usuarios una ruta más cercana al plan de vuelo preferido. En un espacio aéreo *Free Route* se define un punto de entrada y un punto de salida del dicho espacio pero los usuarios pueden planear la ruta que más les convenga sin seguir rutas preestablecidas. En todo momento los vuelos son sujetos al control de tráfico aéreo.

El *Free routing* no es solo una manera de mejorar la eficiencia y la capacidad del espacio aéreo. Es el paso intermedio hacia *4D Business Trajectory*, uno de los retos más importantes para conseguir los objetivos del programa SESAR. Estas trayectorias son expresadas en espacio y tiempo y son compartidas con antelación con el resto de usuarios con el fin de planificar y negociar el encaje de las mismas en el espacio aéreo. Dichas trayectorias son las *Shared Business Trajectory* (SBT) que posteriormente evolucionan hacia las *Reference Business Trajectory* (RBT) implementadas momentos antes del vuelo. Son las trayectorias acordadas entre los usuarios y los proveedores de servicios de navegación aérea.

Un proyecto similar llamado NEXTGEN se está llevando a cabo en EE. UU. Los objetivos de este programa no distan mucho de los de SESAR: incrementar la capacidad del espacio aéreo, incrementar la seguridad y reducir las emisiones de CO₂.

Estos nuevos conceptos permitirán a las aeronaves más flexibilidad para cambiar la trayectoria inicial con lo cual la responsabilidad de separación en algunos casos será delegada a cada aeronave. Como consecuencia se necesitaría especial énfasis en el incremento de la capacidad de cada aeronave de detectar y evitar colisiones. Surge la necesidad de mejorar los algoritmos de detección y evasión de conflictos en ruta para poder mantener y/o mejorar la seguridad en vuelo y dar cabida a la demanda prevista en el futuro.

El ATM actual está llegando a su capacidad máxima. Las normas y procedimientos actuales en la gestión del tráfico aéreo no podrán hacer frente al futuro crecimiento de la demanda. Es de vital importancia que todos los actores de la industria del transporte aéreo colaboren para identificar medidas que puedan mejorar la capacidad y la eficiencia del espacio aéreo. Para medir la eficiencia del ATM existen una serie de indicadores llamados kpi's. Algunos de ellos son:

- La comparación del plan de vuelo con la distancia de la trayectoria óptima. Este indicador es influenciado por las decisiones de los controladores para modificar las trayectorias y por el diseño del espacio aéreo.
- Capacidad del espacio aéreo. El máximo número de vuelos que un sector puede asumir en condiciones normales. Se utiliza para comprobar que los proveedores de servicios de navegación aseguran una capacidad que pueda asumir la demanda actual.
- Retraso en ruta. Retrasos atribuidos a las restricciones que presenta el ATM.

Las restricciones del espacio aéreo introducen costes adicionales a las compañías aéreas. Por ejemplo cuando los controladores indican una maniobra para asegurar la distancia mínima de separación desvían a la aeronave de su ruta inicial con el consecuente consumo de combustible. También pueden introducir retrasos que se traducen en costes de tripulación.

Este proyecto explorará la introducción de los Sistemas Multi-Agente (MAS) en la detección y resolución de conflictos con el objetivo de mejorar la seguridad aérea y la coordinación entre las aeronaves en el espacio aéreo y al mismo tiempo reducir los costes adicionales que pueden ser introducidos por las restricciones que presenta el ATM. La diferencia principal con el TCAS convencional es que las aeronaves negociaran el cambio de trayectorias antes de la necesidad de hacer una maniobra brusca para evasión de conflicto y al mismo tiempo evitar que se generen colisiones inducidas.

1.3 Objetivos

1.3.1. Objetivo principal

Se pretende desarrollar una simulación basada en agentes para evaluar la eficacia y mejorar el sistema TCAS considerando un escenario real o sintético de tráfico con conflicto. Para resolver este tipo de escenarios es necesario proponer maniobras a las trayectorias originales considerando diferentes indicadores económicos de las aerolíneas involucradas. De entre todas las posibles maniobras libres de conflicto, se trata de encontrar una solución óptima para todos los agentes minimizando la desviación de las trayectorias iniciales y el impacto económico y medioambiental. Así tendrán que negociar entre ellos para encontrar las trayectorias factibles que eviten colisiones y penalicen al mínimo a cada agente y al mismo tiempo eviten generar colisiones inducidas. La plataforma utilizada para desarrollar la simulación será Netlogo.

1.3.2. Objetivos específicos

Para poder cumplir el objetivo principal se definirán objetivos parciales que aportaran la información necesaria para la consecución del objetivo principal. Estos objetivos a corto plazo permitirán determinar una metodología de

estudio y análisis de los ámbitos necesarios para la realización de este proyecto.

- Estudiar el funcionamiento del sistema TCAS y su evolución histórica. Resulta necesario conocer cómo ha ido cambiando el sistema a lo largo de los años para poder determinar cuál es la mejor estrategia para la resolución de un conflicto.
- Estudiar el funcionamiento de los modelos de simulación basados en agentes y en concreto la plataforma Netlogo.
- Desarrollo de los algoritmos para la resolución del problema
- Diseñar y ejecutar un conjunto de experimentos que permitan evaluar y validar el potencial del modelo desarrollado.

1.4 Estudio de viabilidad

Para determinar la viabilidad de un proyecto es imprescindible analizar los objetivos definidos y ver si estos se pueden cumplir con éxito. Por otro lado se definirán una serie de riesgos, se catalogaran en función de la probabilidad de que sucedan y se elaborara un plan de contingencia para determinar cual el la mejor manera de solucionarlos.

Analizando los objetivos, este trabajo será viable si se puede realizar un estudio detallado del sistema TCAS, los sistemas multi-agente y el entorno de simulación Netlogo. También será necesario conseguir información real referente a los indicadores económicos que utilizan las compañías aéreas y las trayectorias a simular.

Riesgos del proyecto:

- R1: Planificación temporal errónea
- R2: No conseguir datos reales para la simulación
- R3: No realizar alguna de las tareas planificadas
- R4: Falta de capacidad para modelar el sistema
- R5: Que el entorno de simulación no tenga capacidad de escalabilidad
- R6: Que el entorno de simulación no soporte la definición de algunas dinámicas
- R7: Falta de información referente al indicadores económicos de las compañías aéreas
- R8: Abandono del proyecto por algún motivo

Catalogación de los riesgos:

RIESGO	PROBABILIDAD	IMPACTO
R1	Media	Medio
R2	Media	Medio
R3	Baja	Alto
R4	Baja	Alto
R5	Media	Medio
R6	Media	Alto
R7	Alta	Medio
R8	Baja	Critico

Tabla 2. Catalogación de los riesgos

Plan de contingencia:

Riesgo	Solución
R1	Planificación nueva del proyecto y plantearse de eliminar algunas tareas con poca importancia e impacto bajo sobre el proyecto
R2	Trabajar con datos sintéticos(académicos)
R3	Valorar la importancia de la tarea y si se puede prescindir de la misma
R4	Buscar asesoramiento externo y por parte del tutor
R5	Cambiar el entorno de simulación por uno con más capacidad
R6	Cambiar el entorno de simulación por uno con más capacidad
R7	Trabajar con datos recomendados por parte de los fabricantes de aeronaves
R8	Intentar reconducir la situación para poder llevarlo a cabo

Tabla 3. Plan de contingencia

1.5 Netlogo

Netlogo es un entorno para la simulación de fenómenos naturales y sociales suficientemente avanzado para utilizarlo como herramienta potente para investigaciones en diferentes áreas [12]. Es un software adecuado para modelar sistemas complejos y dinámicos como la gestión de tráfico aéreo y ver cómo se comportan los agentes en un entorno cambiante con la evolución del tiempo. En este contexto se define a un agente como un individuo capaz de tomar decisiones de manera autónoma e interactuar con otros agentes y el entorno que le rodea.

En la interfaz de Netlogo encontremos tres pestañas principales con las secciones que nos permiten definir y ejecutar los modelos.

- Pestaña Ejecutar: es la que nos permite visualizar el modelo y ejecutar la simulación. También nos permite añadir controles adicionales para poder controlar variables globales para modificar el comportamiento global del sistema. Por otro lado se pueden introducir gráficas y diferentes tipos de monitores que actualizan la información con la ejecución del modelo y ayudan a entender los datos que el modelo está generando.
- Pestaña Información: en esta pestaña el creador del modelo puede explicar cómo funciona la simulación y los aspectos importantes que hay que tener en cuenta al ejecutarlo.
- Pestaña Código: aquí se dónde se escriben los procedimientos del modelo que definen el comportamiento de los agentes.

Los agentes presentes en Netlogo son los siguientes:

- *Turtles*: son los agentes móviles.
- *Patches*: representas una porción del entorno.
- *Links*: son agentes que pueden conectar las turtles.
- *Observador*: este agente puede dar instrucciones sobre los demás agentes.

Netlogo permite modelar en 2D y en 3D. Este proyecto se modelara en 3D. También existe una extensa biblioteca de modelos que ayudan al modelador a entender mejor el funcionamiento del software (Netlogo, 2016).

Capítulo 2. Estado del arte

2.1. TCAS

2.2. Modelado basado en agentes

Capítulo 2. Estado del arte

2.1. TCAS

La creciente demanda de movilidad en el siglo XXI pone a los investigadores ante el reto de encontrar herramientas y técnicas más eficientes para solucionar los problemas del tráfico y del transporte (Bazzan y Klugl, 2014). A continuación se realizara un análisis de la investigación sobre el Sistema de Alerta de Tráfico y Evasión de Colisión (TCAS) que se utiliza como última capa de seguridad para disminuir el riesgo de colisiones en el aire (NMAC), y forma parte del equipamiento obligatorio en todos los aviones de transporte comercial (Tang et al., 2015). Posteriormente se hará un estudio sobre los enfoques basados en agentes destinados a evitar colisiones.

Kuchar y Yang (2000) aportan un resumen y debate sobre los principales enfoques utilizados para solucionar problemas de detección y resolución de conflictos (CD&R). En su estudio, los autores citan más de 60 métodos diferentes propuestos por distintos investigadores para solucionarlos. Estos métodos se han desarrollado no solo para el espacio aéreo, sino también para vehículos terrestres, robótica y aplicaciones marítimas ya que los aspectos fundamentales relativos a evitar conflictos son similares en todos los medios de transporte. Kuchar y Yang (2000), definen un conflicto como un acontecimiento en el que dos o más aviones pierden la separación mínima. En consecuencia, el objetivo del sistema de CD&R es predecir la ocurrencia de un conflicto en el futuro, comunicar el conflicto detectado a un operador humano y en algunos casos, colaborar en la resolución de la situación de conflicto. Específicamente respecto a TCAS, Kuchar y Yang (2000) observan que los casos de contradicción entre la detección y solución de conflictos por parte del controlador de tráfico aéreo y las alertas TCAS han tenido lugar desde la introducción de TCAS. Los autores sugieren que en futuros enfoques, será importante tener en cuenta más aspectos, entre ellos los efectos de la incertidumbre, la capacidad de gestionar conflictos múltiples, coordinación, requisitos computacionales, problemas de implementación y aceptación por parte de los pilotos y los controladores.

Según Billingsley (2009), en los métodos de análisis de seguridad utilizados inicialmente para apoyar el desarrollo del TCAS se simularon un gran número de encuentros entre dos aviones y se analizó el comportamiento de la lógica de TCAS en la solución de estos encuentros. Sin embargo, como observa Billingsley (2009), utilizando estos métodos no se ha analizado rigurosamente el comportamiento de la lógica multiamenaza de TCAS y no se

entienden muy bien los mecanismos internos de la lógica. TCAS resuelve los encuentros multiamenaza seleccionando un aviso de solución (RA) que proporciona la separación vertical adecuada en cada amenaza. Puede hacerlo enviando el mismo RA de un encuentro de amenaza única o una combinación de RA de sentido ascendente y descendente (p.e., No Ascender y No Descender). Según Billingsley (2009), la preocupación principal en los encuentros multiamenaza es que TCAS puede enviar una maniobra inadecuada que solucione el encuentro con la primera amenaza pero que induzca a colisión con una amenaza secundaria, o que sea incompatible (p.e., RA de Ascenso y Descenso simultáneos). En el estudio de Billingsley (2009), se utilizó el modelo de encuentro correlacionado para generar un encuentro uno a uno entre dos aviones y para posteriormente utilizar una versión modificada del modelo para generar un tercer avión involucrado en el encuentro. Los resultados indican que en encuentros multiamenaza, la lógica TCAS duplica las colisiones en el aire y aproximadamente las quintuplica respecto a los encuentros uno a uno. Billingsley (2009) observó que TCAS proporciona más seguridad en encuentros multiamenaza mediante avisos de solución que comportan una mayor separación vertical entre el avión equipado y el primer intruso. Además, el estudio evidenció que la Versión 7.1 de TCAS en encuentros multiamenaza aumenta en un 0,45% la seguridad en comparación con la Versión 7.0. Los cambios de lógica de la Versión 7.1 trabajan conjuntamente con la lógica multiamenaza para resolver geometrías relativas a encuentros.

En el estudio de Tang et al. (2015), los autores trataron el problema de que en casos de tráfico intenso TCAS puede inducir a una colisión potencial y que faltan mecanismos para mejorar la función de evasión de TCAS en tráfico aéreo denso. Su estudio se basa en un modelo CPN causal teniendo en cuenta las operaciones de evasión basadas en TCAS para generar todos los posibles estados futuros en los que deberían destacarse los potenciales estados de colisión. El modelo causal de Tang et al. (2015) puede mejorar el comportamiento de TCAS a nivel operativo en escenarios de tráfico de alta densidad para facilitar un control preciso de todo el tráfico y para asegurar operaciones seguras y eficientes.

Para verificar la viabilidad de la técnica propuesta, Tang et al. (2015) realizaron varios experimentos de simulación en un escenario multiamenaza que reunía las siguientes características: (1) Un escenario bastante complejo con varios aviones, (2) Implica interrelación (también denominada efecto dominó) entre amenazas cercanas. Por ejemplo, una nueva amenaza secundaria puede emerger en el proceso de resolución de sus respectivas amenazas anteriores. (3) Permite a cada uno de los aviones implicados seleccionar diferentes cambios de rumbos (ascender/sin

variaciones/descender). La principal contribución de Tang et al. (2015) es una mejora innovadora del comportamiento de evasión de TCAS utilizando un modelo causal de CPN basado en el espacio de estado generado que ayuda a comprender los casos de colisión potencial para evaluar riesgos mediante la representación de la relación causa efecto de cada acción y también ayuda a la tripulación del avión implicada a encontrar una opción cooperativa y viable.

Liang (2015) sigue un enfoque distinto. Su trabajo presenta un sistema de razonamiento basado en casos para la solución automatizada de conflictos de corto alcance en ruta en situaciones de Alerta de Conflicto a Corto Plazo (STCA). El estudio consta principalmente de dos aspectos: representación y recuperación de casos. En primer lugar, se presentan las experiencias del controlador en solución de conflictos como un conocimiento específico organizado en el marco y características del caso que se analizan detalladamente. En segundo lugar, se realiza la recuperación de casos para obtener el mejor caso histórico coincidente con el nuevo caso mediante cálculos de similitud basados en el algoritmo de cercanía más parecido.

2.2. Modelado basado en agentes

Durante la última década, ha habido grandes progresos en la ingeniería de transporte y tráfico debido al uso de tecnología de agentes (Bazzan y Klugl, 2014). Debido a la creciente complejidad de los sistemas de transporte y de tráfico ocasionada por el modo de vida moderna y nuevos medios de organización del transporte, no solo hay que desarrollar nuevas técnicas sino también comprender mejor las opciones individuales para que todo el sistema sea más eficiente. Se ha aceptado que los enfoques basados en agentes se adaptan muy bien a la gestión de tráfico y transporte considerando la distribución de datos y el control geográfico, funcional y temporal y también la interacción frecuente y flexible entre los participantes y su entorno (Bazzan y Klugl, 2014). Por ello, los enfoques basados en agentes pueden contribuir al esfuerzo relacionado con el diseño y control de sistemas de transporte inteligentes. Según Shah et al. (2005), los sistemas de gestión de tráfico aéreo tiene las siguientes características que los hacen adecuados para la simulación basada en agentes: 1. Implicar varios agentes en diversos roles y con diversos y distintos fines y capacidades; 2. Orientado a finalidades u objetivos; 3. Disponer de conocimientos, cultura y procesos establecidos y 4. Capacidad de interactuar con su entorno.

Según Chen y Cheng (2010), el concepto de agentes inteligentes ya se aplica en una gran variedad de aplicaciones en fabricación, sistemas de control en tiempo real, comercio electrónico, gestión de redes, sistemas de transporte,

gestión de la información, computación científica, asistencia sanitaria y espectáculos. Chen y Cheng (2010) manifiestan que desde el punto de vista de la gestión del tráfico y del transporte, las características más interesantes de los agentes son autonomía, colaboración y reactividad. El hecho de que los agentes pueden operar sin la intervención directa de humanos u otros ayuda a implementar sistemas automatizados de control y gestión de tráfico. Además, los agentes son colaborativos. En un sistema multiagente (MAS), los agentes se comunican con otros agentes en un entorno para alcanzar un objetivo global. Además, los agentes también pueden percibir su entorno y responder oportunamente a los cambios ambientales. Chen y Cheng (2010) explican que la distribución geográfica y funcional y la naturaleza altamente dinámica del control del tráfico aéreo (ATC) lo convierten en un candidato ideal con muchas aplicaciones potenciales que se pueden modelar con MAS, como la evasión de colisión y la gestión del flujo de tráfico aéreo. La secuenciación óptima de aviones utilizando la planificación inteligente (OASIS) presentada en Ljungberg y Lucas (1992) es un sistema en tiempo real orientado a agentes desarrollado para dar soporte a la gestión del tráfico aéreo. OASIS distribuye las tareas de gestión de tráfico aéreo entre dos clases de agentes autónomos y cooperativos: agentes aéreos y agentes globales.

Iordanova (2003) propone la planificación de un tráfico aéreo sin conflictos apoyado en arquitectura de DDS integrados para aeropuertos, líneas aéreas y ATC. La coordinación y el intercambio adecuado de conocimientos del espacio aéreo por estas tres partes garantizan la planificación del uso del mismo libre de conflictos antes de que el avión despegue, durante el control en ruta y aseguran un uso eficiente del espacio aéreo. La mayoría de sistemas ATC basados en agentes se diseñan e implementan en arquitectura MAS. En estos sistemas los agentes no actúan solos. Wangermann y Stengel (1996, 1998) proponen utilizar negociación ejemplar en los sistemas inteligentes de aviones/espacio aéreo para que tanto los aviones como las líneas aéreas optimicen sus operaciones. Wollkind et al. (2004) proponen utilizar negociación multiagente cooperativa para la solución de conflictos de tráfico aéreo. La capacidad de coordinación de agentes en MAS es prometedora en muchos sistemas de gestión ATC.

Chen y Cheng (2010) avisan que aunque MAS ofrece muchas ventajas, como la descentralización y colaboración, también aumenta la complejidad del sistema. No obstante, concluyen que los resultados de la investigación demuestran claramente el potencial de utilizar tecnologías basadas en agentes para mejorar el rendimiento de los sistemas de tráfico y transporte. Sin embargo, Chen y Cheng (2010) afirman que la mayoría de aplicaciones basadas en agentes se centran en el diseño y la simulación y que se implementan y desarrollan pocas aplicaciones del mundo real. Por ello, Chen

y Cheng (2010) sugieren que el diseño, la implementación y la aplicación de enfoques basados en agentes en el ámbito de tráfico y transporte todavía necesitan madurar y seguir siendo estudiados.

Shah et al. (2005) explican que la simulación basada en agentes puede integrar modelos cognitivos de conductas humanas, modelos físicos de comportamiento tecnológico y descripción de su entorno operativo. Según Shah et al. (2005), aunque las simulaciones basadas en agentes no pueden incluir todos los aspectos del comportamiento del sistema, pueden proporcionar conocimientos rápidos y rentables que pueden complementar otras formas de análisis. En Shah et al. (2005), las simulaciones basadas en agentes no se basan en modelos de alto nivel de un sistema de tráfico aéreo sino que sitúan modelos de agente en un entorno enriquecido, los simulan en un escenario realista y observan cuál es el resultante comportamiento del sistema. En este caso los niveles de abstracción son (1) los agentes (normalmente humanos como pilotos y controladores) y (2) el comportamiento emergente general del sistema. Además del comportamiento general del sistema, en simulaciones basadas en agentes los agentes responden a su entorno y a los demás. Shah et al. (2005) especifican que se ha utilizado el término agente en el sentido de cualquier cosa entre una mera subrutina u objeto y una entidad adaptable, autónoma e inteligente; pero en su trabajo utilizan la definición de Hayes de un agente como entidad con (1) autonomía, es decir la capacidad de efectuar algunas operaciones locales y (2) interactividad, es decir la necesidad y capacidad de interactuar con otros agentes para realizar sus propias tareas y objetivos. Además, Shah et al. (2005) especifican que la inclusión de un modelo de entorno en una simulación basada en agentes requiere una concepción ligeramente distinta de "entorno" de la utilizada generalmente en ingeniería de sistemas. Más que considerar el entorno como algo exterior a los límites del sistema, en la simulación basada en agentes el entorno abarca todos los elementos pasivos del sistema que sitúan el funcionamiento de los agentes proactivos. El entorno de los agentes puede tener un impacto radical en su comportamiento individual y por ello, en la actuación del sistema emergente; sus elementos (incluyendo espacio físico, nuevas tecnologías y procedimientos y reglamentaciones) son frecuentemente los medios mediante los cuales se realiza el cambio en todo el sistema. Shah et al. (2005) explican que la simulación basada en agentes requiere dos nuevos desarrollos además de la especificación conceptual de agente y modelos ambientales: (1) codificar estos modelos conceptuales con suficiente exactitud para que puedan ser implementados como objetos de software computacional y (2) colocar estos objetos de software en una estructura de software mayor que cree y mantenga sus correctas interacciones.

En resumen, la simulación basada en agentes es un método comparativamente nuevo para analizar sistemas de gestión de tráfico aéreo. Se basa en aquellos aspectos de sistemas de gestión del tráfico aéreo que se pueden observar o especificar directamente– las prácticas de trabajo de los propios agentes – y a partir de ello, predice el comportamiento del sistema en conjunto, y las demandas correspondientes que el entorno presentará a sus agentes, generalmente pilotos y controladores (Shah et al. (2005).

Capítulo 3. Modelo MAS de un nuevo TCAS

3.1 Escenario considerado

3.2 Descripción de Agentes y variables

3.3 Flujo del modelo

3.3.1 Algoritmo de detección

3.3.2 Actualización del algoritmo de detección

3.3.3 Validación del algoritmo de detección

3.3.4 Negociación

3.3.5 Recuperación de la trayectoria inicial –
procedimiento Volver

Capítulo 3. Modelo MAS de un nuevo TCAS

3.1 Escenario considerado

El objetivo principal de este proyecto es evaluar la eficacia y el potencial de los Sistemas Multi-Agente aplicados en el ámbito de la detección y resolución de conflictos en el espacio aéreo. Se presentara la simulación del sistema TCAS donde los agentes negociaran las trayectorias con suficiente antelación para evitar maniobras bruscas de *Climb/Descend* y desviarse lo mínimo posible de sus trayectorias iniciales. El objetivo último de la mejora del sistema TCAS simulado va en línea con los objetivos principales del programa SESAR de incrementar la capacidad del espacio aéreo y la seguridad en ruta y disminuir los efectos sobre el medio ambiente.

Debido a la dificultad para conseguir datos reales el modelo utilizara datos sintéticos (académicos) referentes a las trayectorias simuladas y los indicadores económicos de las compañías aéreas.

Inicialmente se considera un escenario de 4 aeronaves en el cual existen 2 conflictos, uno por cada pareja de aeronaves. Posteriormente se añaden más aeronaves (hasta 6) generando escenarios de multi-amenaza donde los conflictos aumentan de manera considerable y la resolución de los mismos se hace más compleja.

Aunque el objetivo de este proyecto es encontrar soluciones colaborativas entre los agentes, se simulara un escenario donde los agentes compiten entre sí para encontrar la mejor solución a nivel individual. Esta simulación se hará con 6 aeronaves y tiene como objetivo comparar la diferencia de coste entre los escenarios colaborativo y competitivo.

El tiempo en Netlogo se mide en pasos discretos llamados *ticks*. Cada vez que la simulación avanza un paso el número de *ticks* se incrementa en una unidad.

Se asume que cada *patch* tiene 1Nm, la velocidad de las aeronaves es constante a 360 *knot* y cada *tick* equivale a 1 segundo.

En la Figura 8 se muestra el escenario inicial de la simulación.

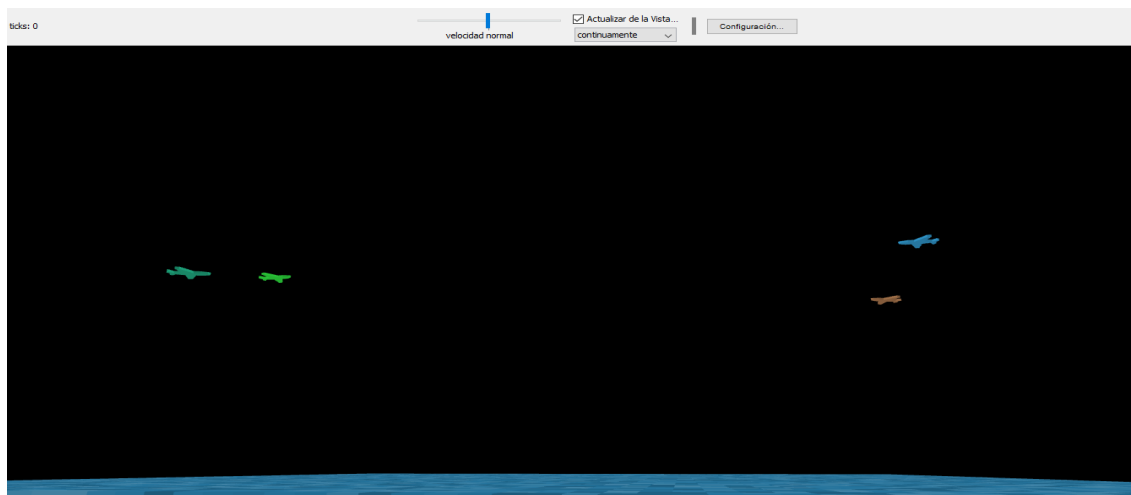


Figura 8. Captura del escenario inicial

3.2 Descripción de Agentes y variables

Los agentes utilizados en este proyecto se representan a través de *turtles* y *patches* según la terminología de Netlogo. Las aeronaves están representadas por las *turtles* y los *patches* representan las trayectorias y los puntos de conflicto que existen entre las aeronaves.

Las *turtles* tienen variables inherentes como el número identificador, las coordenadas xyz, heading, pitch, etc. que juegan un papel importante en esta simulación. Se ha predefinido la forma de las tortugas visibles como "airplane".

A parte de estas se han definido otras variables necesarias para el funcionamiento del modelo que se detallan a continuación:

- **n_pitch**; esta variable conserva el pitch inicial de cada aeronave necesario para poder volver a su trayectoria una vez resuelto el conflicto
- **m_distance**; esta variable es de tipo lista y guarda la distancia hasta el punto de conflicto con cada una de las aeronaves (en el caso de exista más de un conflicto)
- **lp**; es el conjunto de *patches* que conforman la trayectoria inicial de la aeronave
- **new-lp**; esta conjunto de *patches* representa la trayectoria modificada de la aeronave después de efectuar un *Climb/Descend*
- **conflict**; esta variable cuenta el número de conflictos que tiene la aeronave

- **id-conflict**; contiene la lista de identificadores de las aeronaves con las que hay un conflicto
- **movements**; cuenta los movimientos que ha efectuado cada aeronave
- **col1**; contiene la lista de *patches* en los que hay un conflicto
- **col2**; contiene la lista de tiempos hasta llegar al col1
- **last-p**; es el último *patch* de la trayectoria
- **cost**; es el coste acumulado por los movimientos efectuados para resolver los conflictos.

En la siguiente Figura se muestran dos *turtles* con la ventana de variables desplegada y con el modelo ejecutado. Se puede observar que los agentes 0 y 5 están en conflicto ya que en ambas listas de id-conflict aparece el agente contrario. Al mismo tiempo ambos agentes están en conflicto con el agente 1.

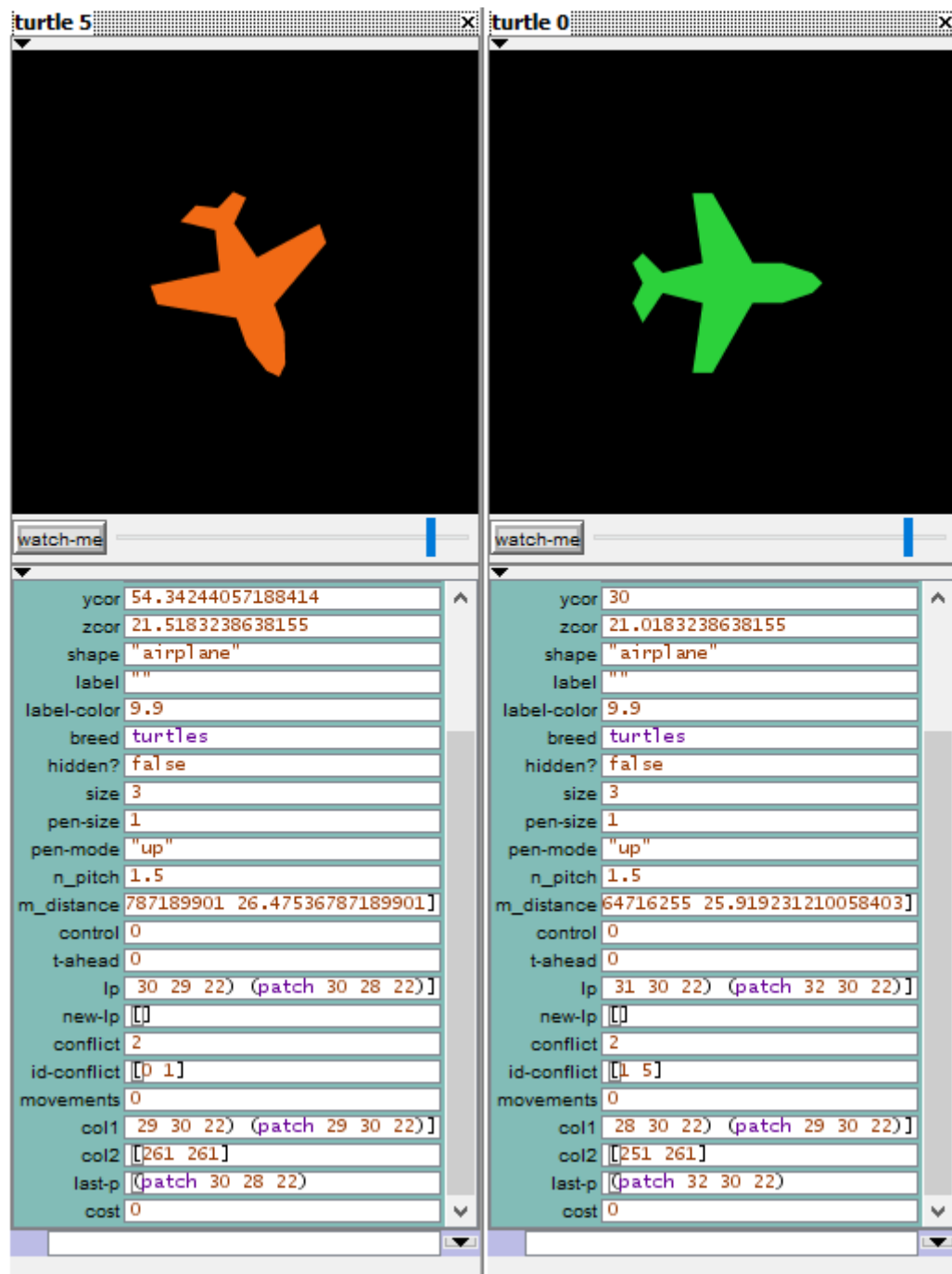


Figura 9. Monitor de turtles con la simulación ejecutada

Para poder controlar el flujo de la simulación se han definido una serie de variables globales. Estas variables se pueden utilizar por todos los procedimientos definidos.

```

globals
[
  lp1          ; lista patches trayectorias tortugas
  c-id         ; lista de numero de tortuga por orden de deteccion
  col         ; index colision
  tiempog      ; timepo global
  total-conflicts ; numero total de conflictos
  stres       ; limite para negociar
  w1          ; penalizacion por movimiento
  w2          ; penalizacion por grado de pitch
  w3          ; penalizacion por tiempo
]

```

Figura 10. Variables globales

3.3 Flujo del modelo

Para ejecutar la simulación se ha habilitado el botón *Setup* que limpia todos los datos de simulaciones anteriores e inicializa las variables globales con los valores predeterminados y reinicia los *ticks* a cero.

El siguiente paso es el botón *Go* que ejecuta los procedimientos de creación de las aeronaves, detección de colisiones, evasión y negociación para la evasión de los conflictos. En las siguientes líneas se explicaran cada uno de los procedimientos del modelo. También está la posibilidad de avanzar la simulación por pasos a través del botón *Step*. Cada vez que se pulsa este botón la simulación avanza un *tick*.

Para poder visualizar los datos que genera la simulación se genera una gráfica con los costes que tiene que asumir cada aeronave al negociar la nueva trayectoria libre de conflictos. Por otro lado se han creado monitores para poder ver los costes individuales y el coste total que generan los movimientos.

El monitor *total-conflicts* indica los conflictos que existen en cada escenario.

El interruptor *mostrar-trayectorias?* nos permite ver las trayectorias libres de conflicto. Al seleccionar la posición *on* los *patches* que conformas dichas trayectorias cambiaran de color.

Por último el deslizador *numt* permite elegir el número de aeronaves que conforman el escenario.

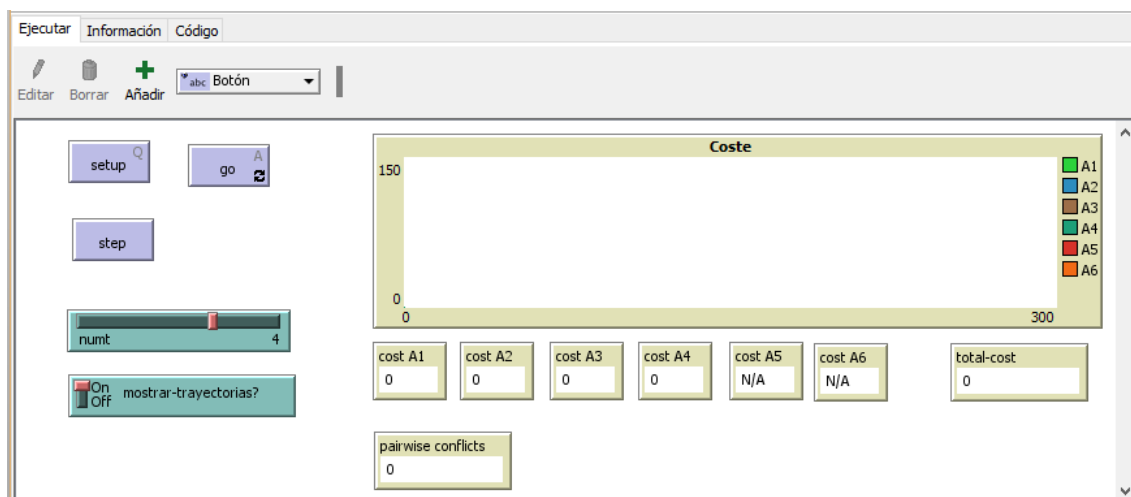


Figura 11. Captura de la ventana de ejecución del modelo

3.3.1 Algoritmo de detección

Es uno de los procedimientos con más peso de la simulación. La correcta detección de las posibles amenazas y la proyección de sus trayectorias en el futuro son esenciales para el funcionamiento del modelo.

El primer paso es preguntar a todos los agentes si existen otras aeronaves a una distancia de 30 Nm. Si es así, se almacena el número identificador de cada una en una lista global.

La siguiente acción que se ejecuta es almacenar la trayectoria del propio agente en la lista *lp* y copiarla en una lista global de trayectorias de todos los agentes. También se guarda el último *patch* de la trayectoria en la variable *last-p* que servirá como referencia para volver a la trayectoria inicial una vez resuelto el conflicto.

Por otro lado cada *turtle* crea un agente “invisible” y del mismo color en cada *patch* de la trayectoria. Estos agentes “invisibles” serán necesarios en otro procedimiento que se explicara más adelante.

Después se comparan las trayectorias almacenadas en la lista global con las listas de cada agente para comprobar si existe un conflicto. Cabe destacar que en esta comprobación con la lista global hay que excluir el propio agente ya que si no se producirían coincidencias que no son reales.

Si se encuentra un *patch* común para dos agentes, este es almacenado en la variable *col1* de cada uno de ellos y la variable *conflict* se incrementa en uno. También se actualizan las variables *id-conflict*, *col2*, *m_distance* y la variable global *total-conflicts*.

Esta serie de pasos del algoritmo implica que desde el primer *tick* de la simulación todos los agentes que se encuentran a 30 Nm uno de otro serán conscientes de los conflictos que tienen y el tiempo que tardan en llegar hasta el CPA ¹ con cada una de las aeronaves que están en la lista *id-conflict*.

A continuación se muestra el código del procedimiento:

```

to detect-plane

  ask turtles with [shape = "airplane"]
  [
    if (any? other turtles in-cone 30 180) and (control = 0) and not(member? who c-id)
    [ set col 1
      set control 1
      let x 0
      let y 0
      let c color
      let p pitch
      let h heading
      while [x < 30]
        [set lp lput patch-at-heading-pitch-and-distance heading pitch x lp

          ask patch-at-heading-pitch-and-distance heading pitch x[sprout 1[ set hidden? true
                                                                           set color c
                                                                           set pitch p
                                                                           set heading h
                                                                           ]]

          set x (x + 1) ]
          set last-p item 29 lp
          while [y < 30]
            [set lp1 lput (item y lp) lp1
              set y (y + 1) ]

          set c-id lput who c-id
          show c-id
          set control 0
        ]
    ]

  if col = 1
  [
    let d 0 ; variable contador para el bucle mientras busca el patch comun
    let k 1 ; variable contador para el bucle que cuenta la totalidad de las tortugas
    let j 1
    let i 0
    while [k < length c-id ]
    [
      set j k
      while [j < length c-id ]
      [
        set d 0

        while [d < 30]
        [
          set i 0
          while [i < 30]
          [
            if item (d + ((k - 1) * 30)) lp1 = item (i + (j * 30)) lp1
            [ type "Collision Avion" type " " type item (k - 1) c-id type " con Avion " type item j c-id type " \n "

              ask turtles with [who = item (k - 1) c-id ] [
                if not(member? (item j c-id) id-conflict)
                [set id-conflict lput (item j c-id) id-conflict
                  set conflict (conflict + 1)
                  set total-conflicts (total-conflicts + 1)
                  set col1 lput (item (d + ((k - 1) * 30)) lp1) col1
                  set m_distance lput distance (item (d + ((k - 1) * 30)) lp1) m_distance
                  set col2 lput ((d * 10) + tiempog) col2]
                ]

              ask turtles with [who = item j c-id ] [
                if not(member? (item (k - 1) c-id) id-conflict)
                [set id-conflict lput (item (k - 1) c-id) id-conflict
                  set conflict (conflict + 1)
                  set total-conflicts (total-conflicts + 1)
                  set col1 lput (item (d + ((k - 1) * 30)) lp1) col1
                  set m_distance lput distance (item (d + ((k - 1) * 30)) lp1) m_distance
                  set col2 lput ((i * 10) + tiempog) col2]
                ]

            show col1
            show col2
          ]
        ]
      ]
    ]
  ]

```

Figura 12. Código del algoritmo de detección.

¹ CPA (Closest point of approach).

3.3.2 Actualización del algoritmo de detección

Para asegurar que en cada momento la distancia mínima de separación no sea violada se ha creado el procedimiento *update-detection*. Aquí es donde cumplirán su función los agentes invisibles creados en el procedimiento anterior.

El funcionamiento de este procedimiento es el siguiente: los agentes invisibles que conforman la trayectoria de cada aeronave tienen un color diferente de los demás agentes. Se preguntan a todos estos agentes si en un radio de 5 Nm y una altura 1000 ft existen agentes de otro color visibles. Si es así comparan el tiempo de llegada de cada aeronave hasta el patch de conflicto y en caso de coincidencia llaman al procedimiento *avoid-collision*. En la siguiente figura se observan los agentes en 30Nm de la trayectoria de cada aeronave visibles a través de la propiedad *hidden?* que tienen los agentes en Netlogo.

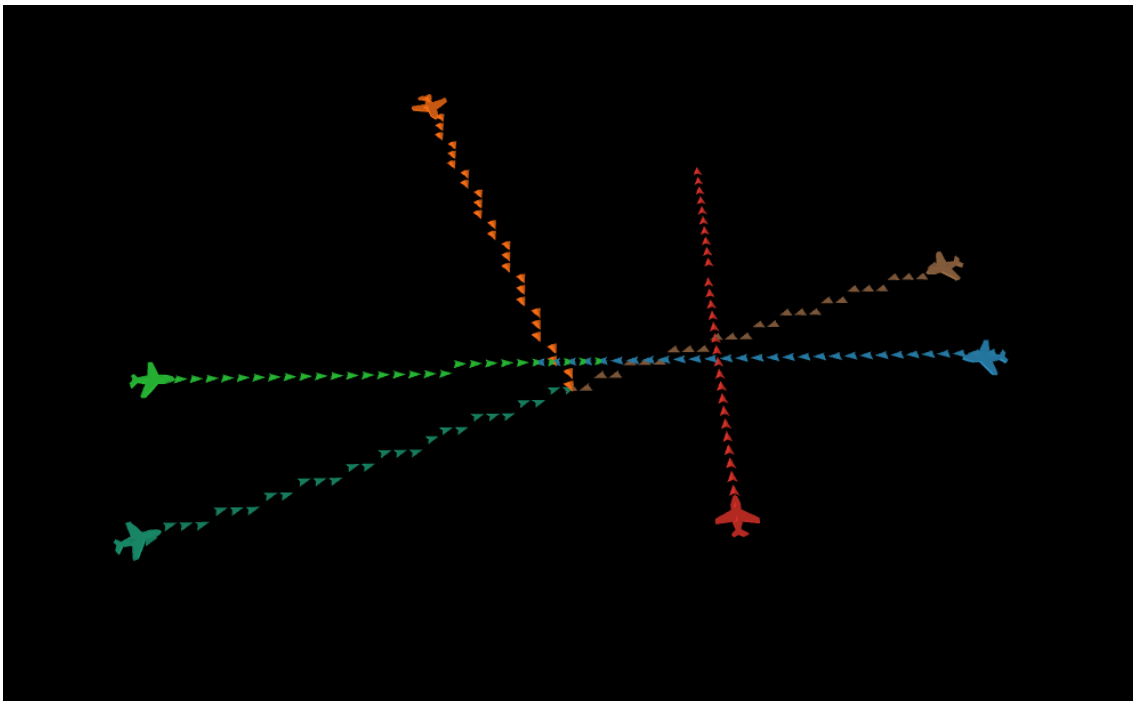


Figura 13. Agentes generados para la detección de intrusos.

3.3.3 Validación del algoritmo de detección

Para validar el algoritmo de detección se han realizado dos simulaciones de escenarios extremos con dos aeronaves volando a la misma altitud.

Se ha habilitado un botón que modifica la coordenada “y” de una de las aeronaves y de esta manera disminuye la distancia lateral entre ambas a menos de 5 Nm. De esta manera el algoritmo debería detectar los *patches* donde la distancia es menor que 5 Nm.

Debido a que son escenarios de validación, la acción que se pide a los agentes que realicen es cambiar el color del *patch* en el que se pierde la separación lateral mínima.

- Escenario 1
Volando en la misma dirección

La figura muestra los *patches* de otro color donde la separación es por debajo de 5 Nm. Si la coordenada “y” se cambia en 0.1 los agentes no detectarían amenaza ya que la separación será respetada.

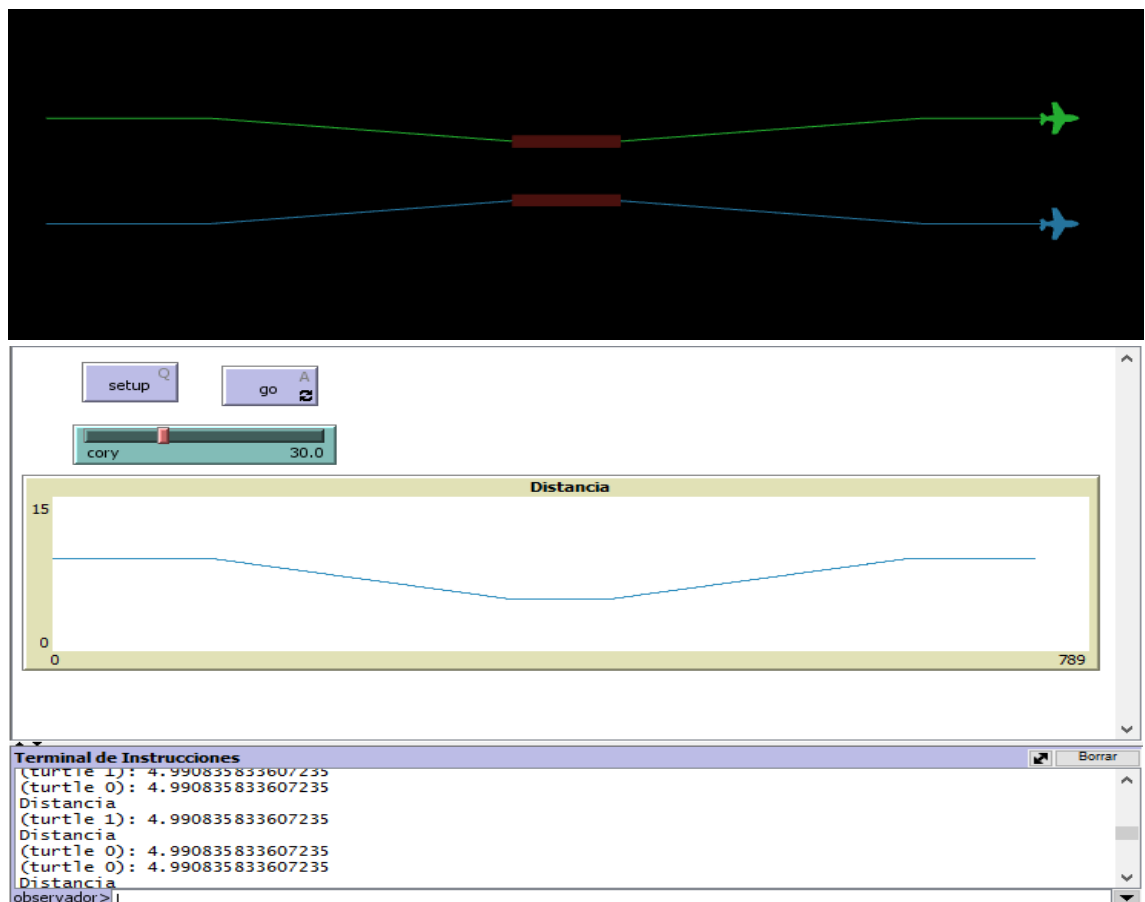


Figura 14. Captura del escenario 1 de la validación del algoritmo de detección.

- Escenario 2

Volando en direcciones opuestas

En este escenario de manera similar al primero los *patches* en los que no se guarda la separación mínima cambian de color. Se observa que la distancia entre las aeronaves es 4,9959 y por lo tanto el intruso es detectado.

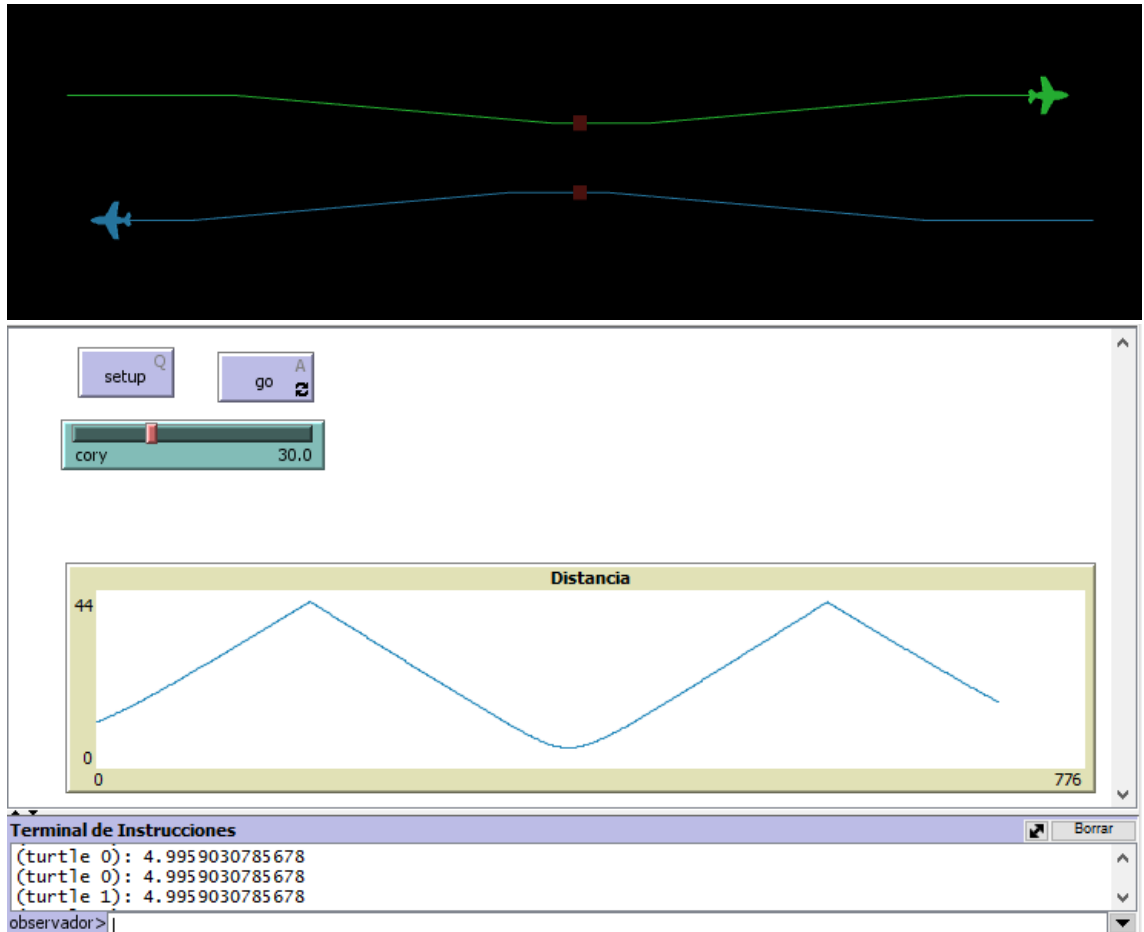


Figura 15. Captura del escenario 1 de la validación del algoritmo de detección.

3.3.4 Negociación

En este procedimiento los agentes que están en conflicto hacen un proceso de negociación para encontrar una solución viable para todos resolviendo los conflictos existentes. Cada agente compara su altitud con los agentes que está en conflicto inicialmente y les propone una maniobra vertical para solucionarlos.

También se comparan las trayectorias con el resto de agentes en el escenario para comprobar si el movimiento efectuado ha generado un nuevo conflicto al resolver el inicial. Si los conflictos resueltos son mayores que los generados se propone otra maniobra al siguiente agente hasta encontrar una solución en la que el escenario quede libre de conflictos.

Los parámetros en función de los cuales se hace la negociación son los siguientes:

- Numero de movimientos efectuados; si un agente ya ha hecho varias maniobras y en cambio el agente con el que está en conflicto no se ha movido, el primero se resiste a moverse de nuevo ya que otro movimiento aumentaría el coste total del vuelo.
- Penalización por cada grado de incremento/decremento de pitch; similar al parámetro anterior aumenta el coste total en función de los grados de inclinación que son necesarios para evitar la colisión.
- Variable stres; esta variable aumenta la presión sobre los agentes para negociar de la manera más rápida posible. El coste de cada maniobra incrementa de manera proporcional al incremento de la variable la cual aumenta en una unidad después de cada iteración. Es decir cuánto más tarden los agentes en ponerse de acuerdo más costosa será la maniobra a efectuar para resolver el conflicto.

3.3.5 Recuperación de la trayectoria inicial – procedimiento Volver

Una vez solucionados los conflictos existentes las aeronaves recuperan la trayectoria que tenían antes del comienzo de la negociación. El procedimiento volver se encarga de guiar los agentes hacia un punto de la trayectoria inicial que tenían asegurando que las separaciones mínimas sean respetadas. Así, después de la resolución de los conflictos las aeronaves vuelven a sus trayectorias óptimas.

Capítulo 4. Resultados obtenidos

4.1 Escenario 1

4.2 Escenario 2

4.3 Escenario 3

Capítulo 4. Resultados obtenidos

En este proyecto se ha desarrollado un modelo de simulación utilizando técnicas MAS para mejorar el sistema TCAS.

Aunque los datos para la simulación son académicos, se ha comprobado que la simulación basada en agentes permite una comunicación autónoma entre las aeronaves sin necesidad de un agente controlador. Las aeronaves tienen la información suficiente sobre el tráfico en el entorno y pueden negociar una resolución de los conflictos.

Para poder comparar los resultados y el impacto que supone la actitud de los agentes a la hora de negociar se han modelado dos escenarios que tienen los mismos agentes pero con comportamientos diferentes. En un caso los agentes colaboran para encontrar la solución más eficiente para todos como conjunto. En el otro caso los agentes tienen una actitud más “agresiva” y la solución que buscan es la más eficiente a nivel individual.

A continuación se realizarán varias simulaciones y se analizarán los datos obtenidos.

4.1 Escenario 1

El primer escenario es ejecutado con 4 aviones buscando una solución de los conflictos de manera cooperativa. Es un escenario en el cual se generaría una colisión inducida si fuera resuelto con la lógica del TCAS. Podemos observar como los costes que generan las maniobras efectuadas por cada aeronave son muy similares. Además la distancia de seguridad mínima es respetada entre cada una de ellas.

Por otro lado no se ha generado ninguna colisión inducida ya que los agentes han modificado sus trayectorias una vez comprobado que esta nueva trayectoria no genera un nuevo problema.

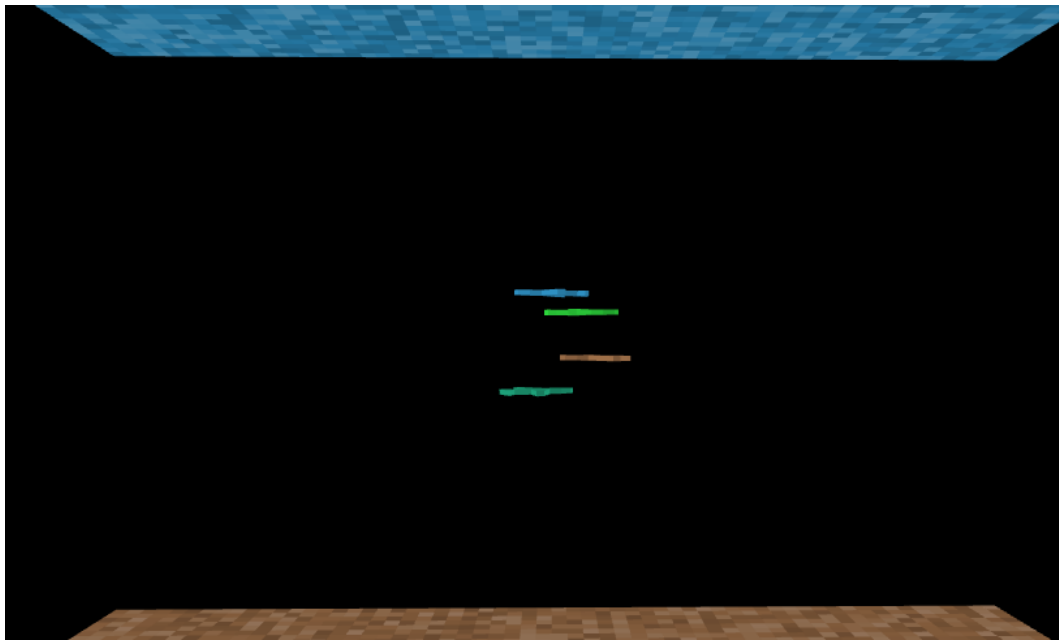
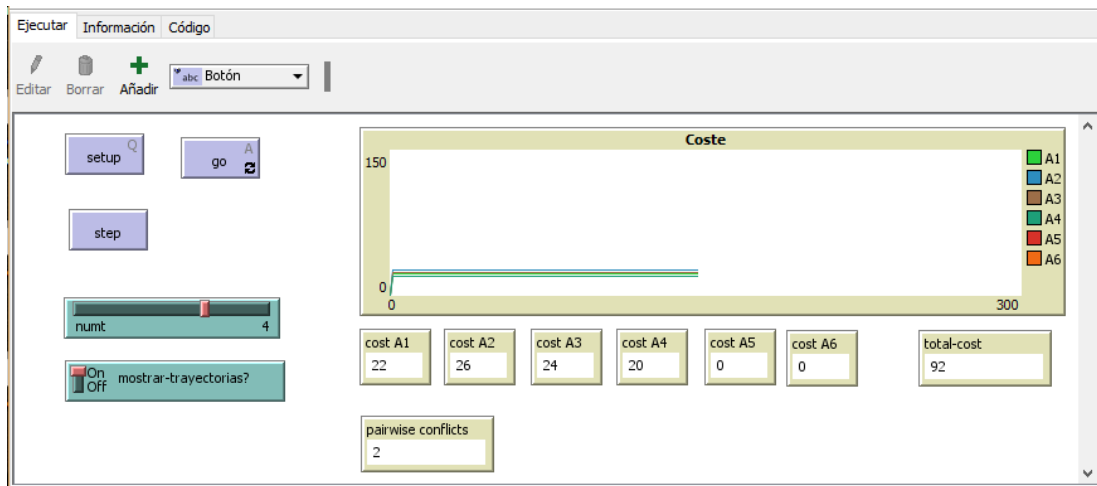


Figura 16. Captura de la simulación del escenario 1

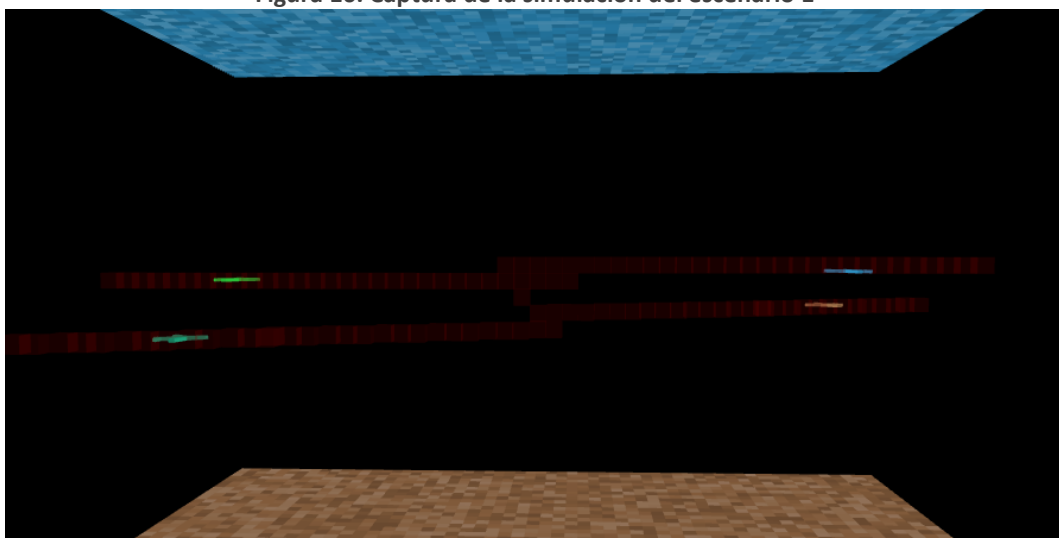


Figura 17. Captura del escenario 1 con las trayectorias libres de conflicto.

En la Figura 17 se muestran las trayectorias proyectadas de las aeronaves una vez resueltos los conflictos. Se observa claramente que las aeronaves volarán a niveles diferentes.

4.2 Escenario 2

En este escenario se han añadido dos aeronaves al escenario anterior. Los agentes han encontrado una solución global eficiente a pesar de que los conflictos han aumentado a 5. Aunque alguno de los agentes tiene un coste superior a los demás, ninguno de ellos ha tenido que efectuar una maniobra de evasión. Los cambios de las trayectorias son progresivos y las distancias mínimas son respetadas.

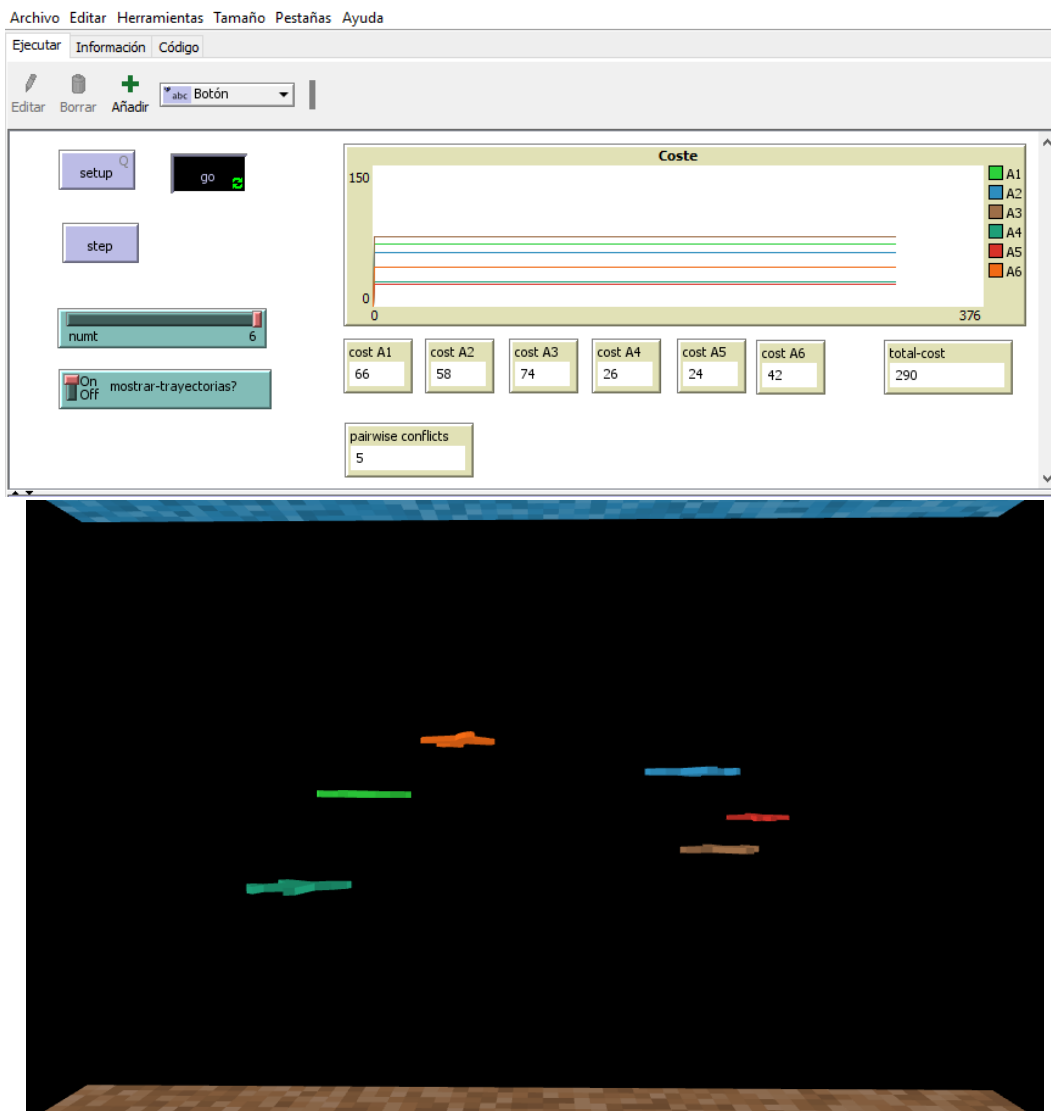


Figura 18. Captura de la simulación del escenario 2

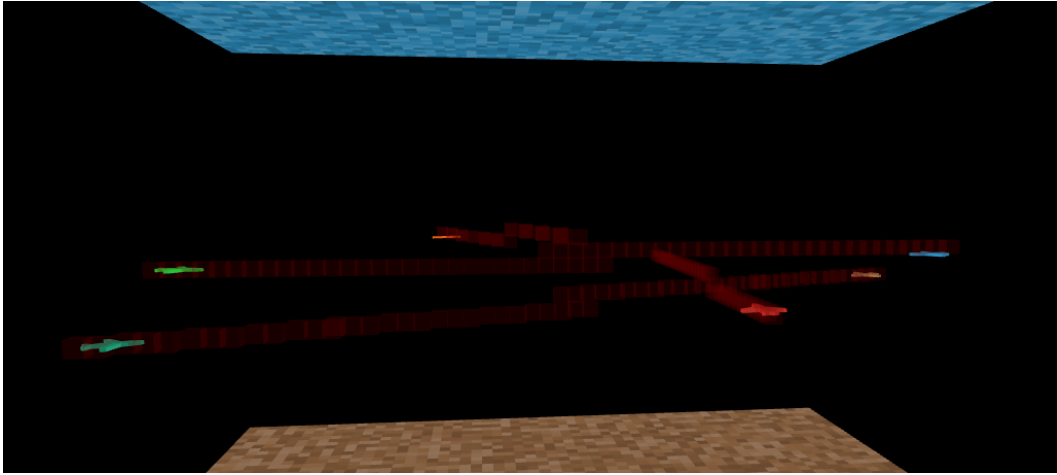


Figura 19. Captura del escenario 2 con las trayectorias libres de conflicto.

De manera similar al escenario 1 la Figura 19 muestra las trayectorias libres de conflicto de todas las aeronaves después de la negociación.

4.3 Escenario 3

Este escenario contiene las mismas 6 aeronaves que el escenario 2. La diferencia reside en que los agentes aquí tienen una actitud competitiva. Mientras en el escenario anterior los agentes intentan ponerse de acuerdo, en este cada uno intenta que se mueva otro agente sabiendo que puede llegar a una situación extrema de evasión de conflicto que puede ser mucho más costosa. Se observa en la figura siguiente como dos aeronaves cumplen un RA para evitar la colisión. También se puede ver de la gráfica que la aeronave de color azul ha hecho una maniobra brusca con el consecuente incremento de coste.

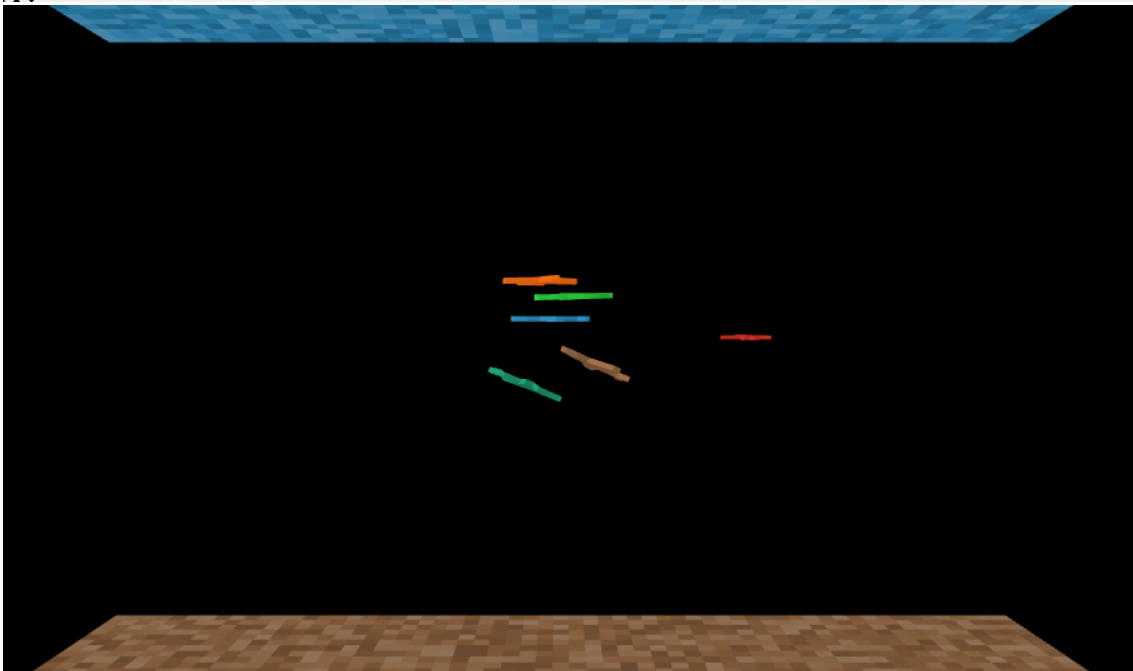
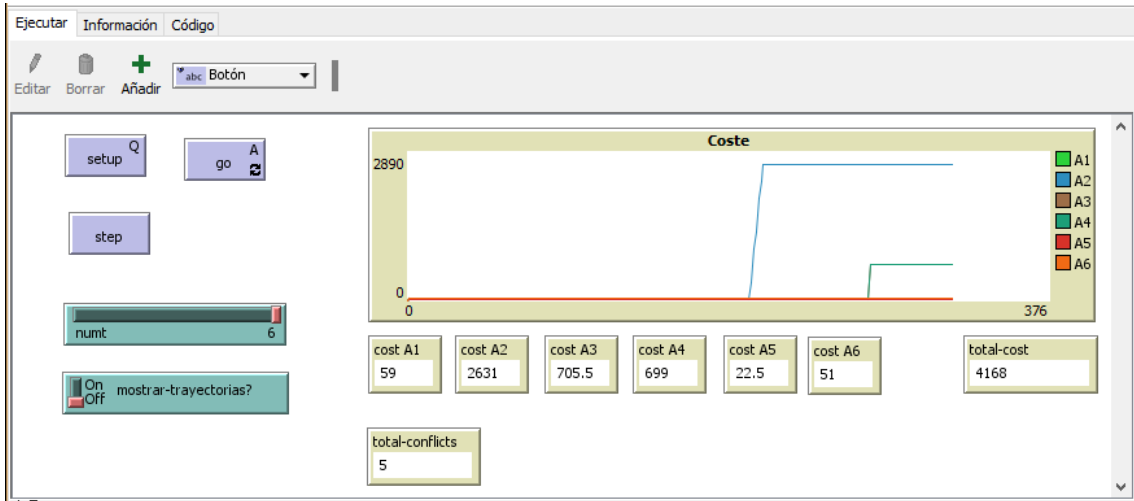


Figura 20. Captura de la simulación del escenario 3

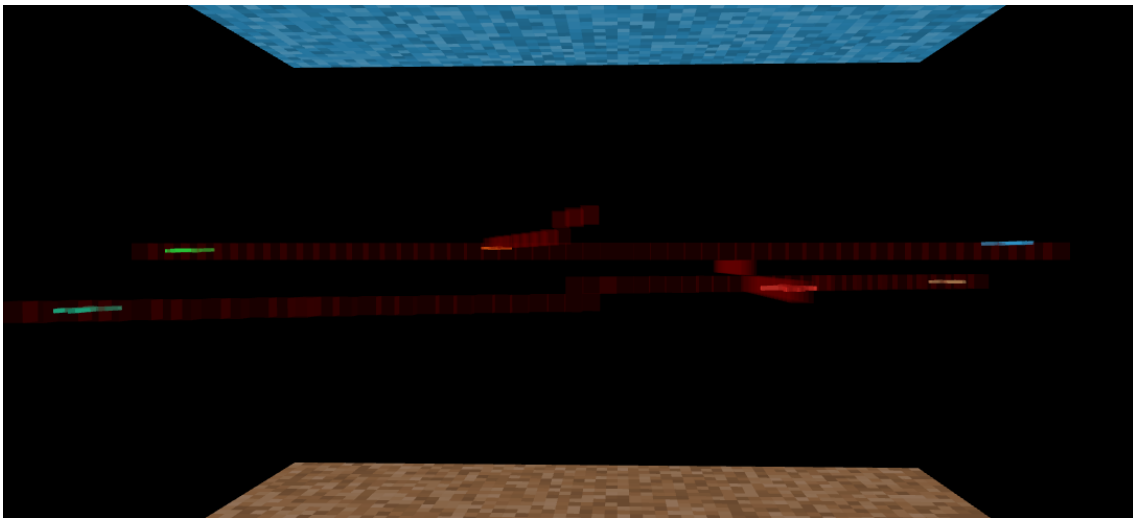


Figura 21. Captura de las trayectorias del escenario 3

En la Figura 21 se observa claramente que algunas de las trayectorias proyectadas no están libres de conflicto. En este caso los agentes tienen un comportamiento competitivo y no están dispuestos a modificar sus trayectorias.

Capítulo 5. Conclusiones y líneas de trabajo futuras

5.1. Conclusiones

5.2. Líneas futuras de trabajo

Capítulo 5. Conclusiones y líneas de trabajo futuras

5.1. Conclusiones

El propósito de este trabajo era evaluar la aplicación de los sistemas basados en agentes para mejorar el sistema TCAS. Aunque TCAS funciona de manera fiable en los encuentros uno a uno, presenta deficiencias en los encuentros multi-amenaza pudiendo generar colisiones inducidas. Se ha demostrado que MAS son una herramienta útil y eficaz aplicada a la resolución de conflictos en el espacio aéreo y en la mejora del sistema TCAS.

Para hacer frente a las previsiones de aumento de tráfico aéreo se tendrán que aplicar nuevas técnicas de gestión de tráfico basadas en la automatización y en la integración de los diferentes actores. Las técnicas MAS constituyen una nueva línea de investigación y pueden ser un enfoque diferente en la resolución de conflictos y evasión de colisiones.

5.2. Líneas futuras de trabajo

Para aproximar el modelo a la realidad hay algunas áreas que pueden ser mejoradas.

- En el algoritmo de detección de amenazas cada aeronave está creando un agente invisible por cada *patch* en su trayectoria en las 30 Nm siguientes. Se ha elaborado un gráfico para ver como degrada el algoritmo en tiempo de ejecución a medida que se aumenta la distancia de detección de intrusos. Podemos observar que el tiempo de ejecución aumenta considerablemente con la distancia de detección. Esto es debido a que el modelo tiene que crear tantos agentes como millas náuticas de detección multiplicados por el número de aeronaves. Es necesario llevar a cabo un estudio para ver cómo afectaría la fiabilidad del algoritmo y el tiempo de ejecución si en vez de crear un agente cada *patch* lo creamos cada 2 o 3 *patches*.

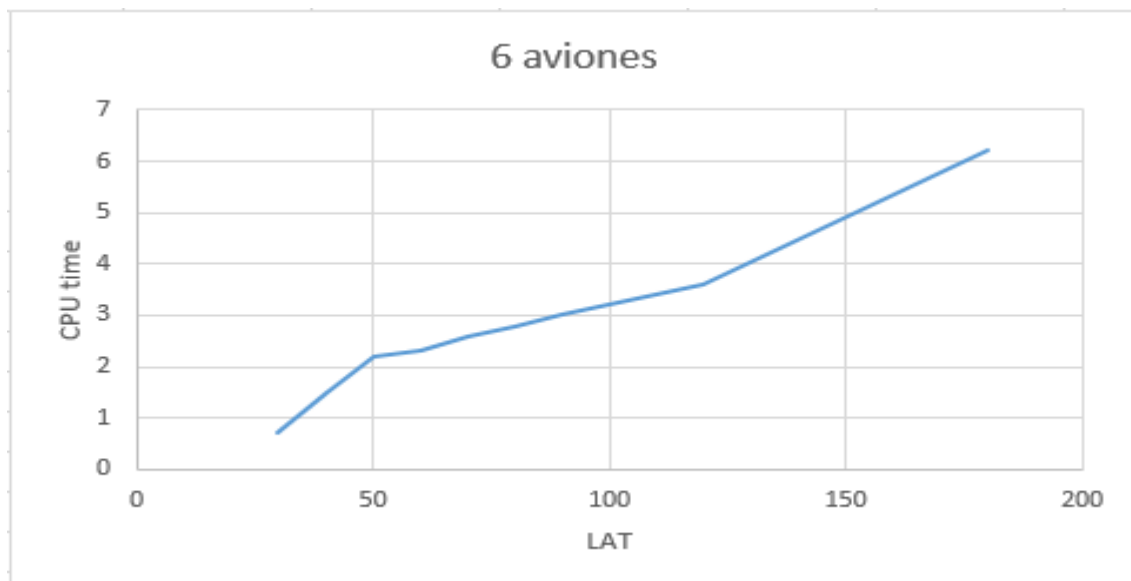


Figura 22. CPU time vs distancia de detección.

- Una función de coste que pueda proyectar el coste total en función de las decisiones actuales podría ser otro punto de mejora que influiría considerablemente en las decisiones que toman los agentes.
- En este modelo se asume que las aeronaves son de características similares y no se aplican cambios de velocidad. En la negociación se podría aplicar el cambio de pitch teniendo en cuenta las dimensiones de las aeronaves. Tampoco se ha contemplado el efecto de la estela turbulenta.
- En este proyecto se tienen en cuenta solo los conflictos del sector actual en el que están las aeronaves. Se podrían considerar todos los conflictos que han tenido las aeronaves a lo largo del trayecto desde el despegue a la hora negociar los movimientos.
- Otro aspecto a tener en consideración es si las aeronaves son de la misma compañía o de la competencia. Diferentes normas se pueden aplicar en este sentido ya que si dos aeronaves en conflicto son de la misma compañía lo podrían resolver según normas internas de la compañía. Sin embargo, si son aeronaves de diferentes compañías la negociación será más compleja.
- Por otro lado la negociación se puede llevar a nivel de compañías en vez de aeronaves. Esto significa que a la hora de resolver un conflicto a

través de la negociación entre aviones de dos compañías distintas se debería mirar el histórico de encuentros que han tenido a lo largo del día, semana, mes los aviones de sus flotas y tomar las decisiones de manera cooperativa o competitiva. En esta línea se podría tener en consideración el poder de negociación de las compañías aéreas en función del número de aeronaves que tienen, facturación, pertenencia a algún grupo, etc.

- Muchos estudios se están llevando a cabo para mejorar la capacidad del TCAS para evitar colisiones. TCAS III, que todavía no está implementado, considera la posibilidad de efectuar maniobras horizontales para la resolución de conflictos. Incorporar este enfoque al modelo actual podría mejorar la función del TCAS en la resolución de los conflictos.
- Por último se podría introducir un factor “competitividad” entre las compañías aéreas. Si una compañía quiere eliminar un competidor del mercado nunca negociara de manera cooperativa en la resolución de los conflictos e intentara resolver el conflicto en beneficio propio y perjudicando a la competencia en la mayor medida posible.

Bibliografía

Bibliografía

- [1] Ana L.C.Bazzan and Franziska Klügl, “A review on agent-based technology for traffic and transportation”, 2014.
- [2] James K. Kuchar and Lee C. Yang, “A Review of Conflict Detection and Resolution Modeling Methods”, IEEE Transactions on Intelligent Transportation Systems, Vol. 1, No. 4, December 2000, pp. 179-189.
- [3] T.B. Billingsley, L.P. Espindle and J.D. Griffith, “TCAS Multiple Threat Encounter Analysis”, October 2009.
- [4] Jun Tang, Miquel Angel Piera, Yunxiang Ling, and Linjun Fan, Extended Traffic Alert Information to Improve TCAS Performance by means of Causal Models, June 2015.
- [5] Shah et al. "Analyzing Air Traffic Management Systems Using Agent-based Modeling and Simulation." 6th USA/Europe ATM R&D Seminar. 27-30 June 2005, Baltimore, MD.
- [6] M. Ljungberg and A. Lucas, “The OASIS air traffic management system,” in Proc. 2nd Pacific Rim Int. Conf. Artif. Intell., Seoul, Korea, 1992.
- [7] B. N. Iordanova, “Air traffic knowledge management policy,” Eur. J. Oper. Res., vol. 146, no. 1, pp. 83–100, Apr. 1, 2003.
- [8] J. P. Wangermann and R. F. Stengel, “Distributed optimization and principled negotiation for advanced air traffic management,” in Proc. IEEE Int. Symp. Intell. Control Held Together With IEEE Int. Conf. Control Appl., IEEE Int. Symp. Comput.-Aided Control Syst. Des. (Cat. No. 96CH35855), New York, 1996, pp. 156–161.
- [9] J. P. Wangermann and R. F. Stengel, “Principled negotiation between intelligent agents: A model for air traffic management,” Artif. Intell. Eng., vol. 12, no. 3, pp. 177–187, Jul. 1998.
- [10] S. Wollkind, J. Valasek, and T. R. Ioerger, “Automated conflict resolution for air traffic management using cooperative multiagent negotiation,” in Proc. AIAA Guid., Navigat., Control Conf., 2004, pp. 1078–1088.
- [11] Bo Chen and Harry H. Cheng, “A Review of the Applications of Agent Technology in Traffic and Transportation Systems”, June 2010.
- [12] Juan Carlos García Vázquez, Fernando Sancho Caparrini, “Netlogo una herramienta de modelado”, 2015

- [13] Jun Tang, “Causal models for analysis of TCAS-Indused collisions”, PhD Thesis, 2015
- [14] Youngraee Kim, Sangchul Lee, Keumjin Lee and Ja-Young Kang, “A Development of 3-D Resolution Algorithm for Aircraft Collision Avoidance”, Int’l J. of Aeronautical & Space Sci. 14(3), 272–281, 2013
- [15] James K. Kuchar and Ann C. Drumm, “The Traffic Alert and Collision Avoidance System”, 2007

Paginas web:

- (1) <http://www.aviator.tk/2016/03/traffic-alert-and-collision-avoidance.html>
- (2) <https://ccl.northwestern.edu/netlogo/>
- (3) <http://www.eurocontrol.int/statfor>
- (4) [http://www.skybrary.aero/index.php/Airborne_Collision_Avoidance_System_\(ACAS\)](http://www.skybrary.aero/index.php/Airborne_Collision_Avoidance_System_(ACAS))