A NEW WAKE VORTEX ENCOUNTER PREDICTION SYSTEM BASED ON SPATIAL DATA STRUCTURES

Bachelor's Degree in Aeronautical Management
Written by Laura Hernández Herrera
Directed by Dr. Sergio Ruiz Navarro
Sabadell, July 8th 2015

Títol del Treball Fi de Grau

A new Wake Vortex Encounter Prediction System based on Spatial Data Structures

Autor[a]: Laura Hernández Herrera

Data: 08 de Juliol de 2015

Tutor[a]/s[es]: Dr. Sergio Ruiz Navarro

Titulació: Grau en Gestió Aeronàutica

Paraules clau

- **Català:** Trobada amb Estela Turbulenta, Trajectòria 4D, Estructures de Dades Espacials, Granularitat, Gestió del Tràfic Aeri, Detecció de Conflictes, Distàncies de seguretat.
- **Castellà:** Encuentro con Estela Turbulenta, Trayectoria 4D, Estructuras de Datos Espaciales, Granularidad, Gestión del Tráfico Aéreo, Detección de Conflictos, Distancias de seguridad.
- **Anglès:** Wake Vortex Encounter, 4D Trajectory, Spatial Data Structures, Granularity, Air Traffic Management, Conflict Detection, Safety distances.

Resum del Treball Fi de Grau

- **Català:** El projecte proposat es basa en el disseny i desenvolupament conceptual d'un sistema predictor de trobades amb esteles turbulentes, per tal de garantir una futura gestió del tràfic aeri més segura i eficient. Adaptant l'ús de les tecnologies existents per a la detecció de conflictes de manera adequada, el sistema ha de ser capaç de contemplar tot l'espai aeri Europeu (tot l'espai d'estats), fent possible la seva implementació a nivell computacional, en termes de memòria requerida.

- **Castellà:** El proyecto propuesto se basa en el diseño y desarrollo conceptual de un sistema predictor de encuentros con estelas turbulentas, a fin de garantizar una futura gestión del tráfico aéreo más segura y eficiente. Adaptando el uso de las tecnologías existentes para la detección de conflictos de manera adecuada, el sistema debe ser capaz de contemplar todo el espacio aéreo Europeo (todo el espacio de estados), haciendo posible su implementación a nivel computacional, en términos de memoria requerida.

- **Anglès:** The proposed project is based on the conceptual design and development of a wake vortex encounter predictor system, in order to ensure a safer and more efficient future air traffic management. Adapting the use of current technologies to detect conflicts properly, the system should be able to consider all European airspace (all state space), making possible its computational implementation in terms of memory required.

CONTENTS

FIGURES

1. INTRODUCTION

1.1 Background Information

Given the spectacular growth of air transport in recent years and considering future forecasts, it is essential the consolidation of a more unified and efficient European air traffic management.

The SESAR program (Single European Sky ATM Research) is the technological support and the means of implementation of the European project SES (Single European Sky). One of its most important goals is the operations management based on trajectories, in order to obtain, the optimal trajectories through a collaborative decision process (Collaborative Decision Making) to obtain conflict-free trajectories, considering the interests of all stakeholders, and with enough time-ahead to ensure the highest possible safety and efficiency.

Since forecasts show a growth factor of 150% of air traffic up to 2020, and therefore, there will be an increased traffic condensation, it will be necessary overturn the efforts not only to detect conflicts and collisions, but also to avoid the hazards of wake vortex encounters (WVE). Although they rarely cause accidents with consequences of great magnitude, it may cause the destabilization of the aircraft and injury the crew and its passengers.

Actually, in TMA the dangers of wake vortex encounters are well known and airborne separation minima defined by ICAO are specifically designed to minimize the underlying hazards, especially near the runway (with the aircraft closer to the ground and closer to their flight envelope limits). Nowadays wake vortex hazard is one of the most limiting factors in airport capacity, thus many efforts have been done to find new ways of re-categorizing aircraft according to their wake vortex (RECAT project is an example) and new procedures such as Time-Based Separations (TBS) have been designed considering wake vortex behaviour and duration to allow ATC (Air Traffic Control) and flights being more precise with the synchronization of the landing traffic.

In en-route operations the dangers have been traditionally considered negligible because the probability of a wake vortex encounter has been up to now low and the severity of a possible encounter has been often found not worrying. Nevertheless, several factors that may cause the increase of both the probability of encounter and the severity are changing in the context of the new ATM proposed by SESAR: for the next decades more traffic density is expected, as well as more heterogeneity in the aircraft mix (there is a tendency to bigger and smaller aircraft coexisting in the same airspace) and new concepts of operations may be introduced such as the concept of Free-Routing, or the potential introduction of continuous operations for all the phases of flight, referred sometimes as 3D Free-Routing (which implies the relaxation of Flight Level separation standards, with an expected gain in the capacity and efficiency performances of the network).

Figure 1 illustrates an example of potential hazardous wake encounters at en-route airspace with changes in aircraft mix even when the current safety standards are applied.

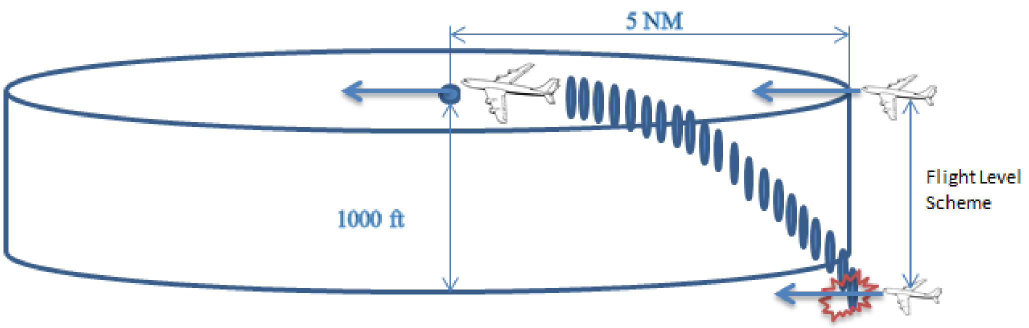


Figure 1: Potential hazardous wake encounters at en-route airspace

Figure 2 illustrates the risk of dangerous encounters if 3D Free-Route operations (i.e. continuous operations) were introduced in the ATM. Note that the relaxation of the flight level schemes might cause severe encounters even if aircraft current separation is maintained and that these identified hazards could become even worse if the current separation standards are reduced on behalf of the airspace capacity.

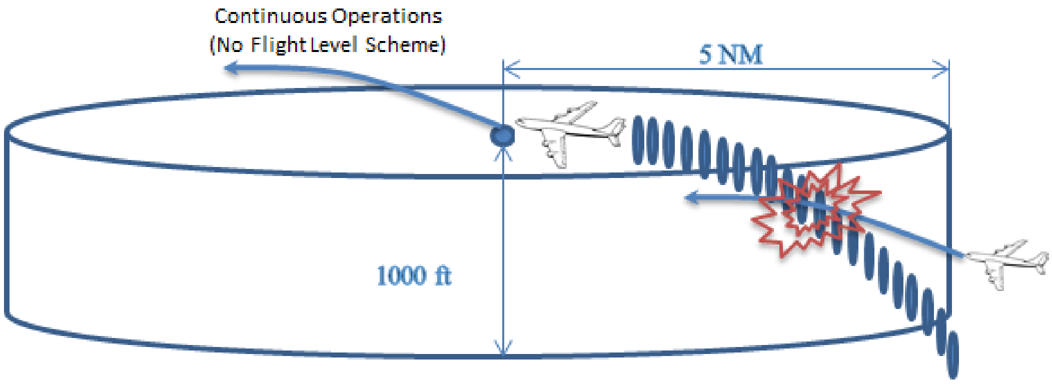


Figure 2: Risk of dangerous encounters if 3D Free-Route operations are applied

Thus, in the context of the future ATM proposed by SESAR for 2020+ [21], i.e., a Trajectory Based Operations (TBO) system, some new wake vortex separation rules and new strategic separation methods for all phases of flight (in particular for en-route) will be required.

In particular, new strategies have to be found in order to store information about wake vortex, and consequently, to predict wake vortex encounters, achieving Conflict Resolution (CR) systems safer and more efficient since the potential risks of this phenomenon are taken into account and the responsiveness to uncertainties arising is improved.

Therefore, the main objective of this project is to extend and enhance previous technologies used in TMA to be used in ECAC, since SDS current technology has been successful in MTCD (Mid-Term Conflict Detection) and SD (Stategic-deconfliction) due to its capability to efficiently store and manage information about all European state-space.

This project is based mainly in the fields of information management and computer science, since what we want to achieve is a way to join these two fields in order to be able to handle micro-level state-space information of the traffic and associated wake vortices at entire European ATM level.

1.2 Research Objectives

The main purpose of the project is to develop a system capable to predict encounters with wake vortices generated by aircraft flying over the airspace, considering all the European air traffic (all state space), and reducing the amount of memory necessary to store all the information required.

However, in order to achieve this objective successfully, it has been necessary the definition of specific sub-goals, which are presented below:

- Obtain a wider knowledge of the nature and behaviour of wake vortices.
- Know the derivative risks and consequences of wake vortex encounters during flight.
- Become familiar with current conflict detection systems based on spatial data structures (SDS) and identify gaps they may present.
- Design a mechanism able to provide information about the position and strength of wake vortices, in a certain period of time, caused by those aircraft that fly over the modelled space.
- Adapt this mechanism to consider the entire state space, i.e. the entire European air traffic, reducing the amount of memory needed to facilitate their implementation.
- Design a high level architecture that contains this mechanism, and therefore, is able to provide information about the severity in the time-space of this phenomenon to guarantee a more efficient global conflict resolution.

1.3 Research Methodology

The research methodologies followed in order to achieve each of the objectives set out in Section 1.2 are listed below:

- Analyze the related literature and preceding projects documentation.
- Identify the shortcomings in the systems presented in the previous researches.
- Develop a brainstorming about potential solutions to implement.
- Identify the strengths and the weaknesses of each one.
- Select the best option to be included in the system’s development.
- Perform calculations to obtain results.
- Analyze the results in order to identify the improvements obtained with respect to the current systems.

1.4 Structure of the document

The structure of the document is composed by the following chapters:

Chapter 2: includes information about SESAR’s objectives and innovations as well as its influence in future ATM. A part from that, data about the origin and behavior of wake vortex are presented, also explaining the derived potential hazards. Finally, SDS architectures used in current conflict detection systems are deeper explained due to their importance in this project.

Chapter 3: presents the extension of the current SDS technology, describing each one of the functional modules that configures its operational structure.

Finally, the results and improvements obtained using the proposed high-level architecture are shown in chapter 4 while the conclusions are presented in chapter 5.

2. STATE OF THE ART

The contents presented in this chapter are the following ones:

- **SESAR's objectives and future ATM** to understand the environment in which the project is contextualized.
- **Wake vortex and potential hazards** because it is important to know the WV's behaviour and dangers of encounters in order to extend the current technologies in such way that the derived uncertainties of this phenomenon can be contemplated.
- **Previous conflict detection systems based on Spatial Data Structures (SDS)**, because they are the technological starting point to conceptually develop the proposed high-level architecture.

2.1 Single European Sky ATM Research (SESAR) and Trajectory Based Operations (TBO)

The European ATM system currently handles approximately 28,000 commercial flights daily, with peaks up to 33,000 flights, aside from other types of traffic. The forecast of growth to 2020 points to a traffic growth of 150% [12] what is a challenging task in reference to their management and the necessary infrastructure. This implies that should undertake operational and technological developments in various fields in order to optimize both the use of airport infrastructure and the airspace, to reduce the operating costs, and to improve the safety standards by minimizing the environmental impact [12].

In this context the European Union launched the "Single European Sky" initiative that involves the development and implementation of a common transport policy in order to achieve an effective air traffic management system able to allow the safe and regular operation of its services, so as to facilitate the free movement of goods, people and services. These goals are achieved through a more rational use of airspace, increasing its capacity and ensuring a high degree of safety throughout the European level airspace organization.

Contrary to what happens in the US, Europe has a fragmented sky where the air traffic management is conducted at a national level. This situation greatly complicates the management of air traffic in our continent. The Europe's Single European Sky initiative aims to solve this fragmentation, which is a major bottleneck for airspace capacity, structuring the air navigation services at pan-European level instead of at national level.

The implementation of the "Single European Sky", which is based on the intensive use of modernize technologies and higher degree of automation, will allow to, progressively, have more direct flight trajectories, reduce flight times, achieve fuel savings and thus reduce pollutant emissions into the atmosphere, while increasing punctuality and reducing costs [12].

The answer to the challenge of increasing air traffic in Europe is growing in two different ways, but closely related: first the seamless management of the airspace with global perspective (Single European Sky) and, on the other hand, operational tools and technologies necessary for its proper management (SESAR).

SESAR (Single European Sky ATM Research) is a joint project of the European air transport community whose aim is the implementation in 2020 of a European ATM network of high performance, born from the need to establish an integrated and shared vision on the current state and evolution of air traffic management system. Its aim is to address, through the implementation of new operational concepts, procedures and technologies, the demand increasing and the requirements planned for the coming years.

The SESAR JU’s general objectives can be summarized as:

- Manage an increasing volume of air traffic until triple the current.
- Reduce by 10% the environmental impact of each flight.
- Reduce by 50% the costs associated with the management of air traffic.
- Multiply by 10 the security of air traffic.

The achievement of these goals will enable a number of benefits in economic terms:

a) Social benefits (by the year 2020). Contributions to European gross domestic product (sum of direct, indirect and induced contributions) equivalent to 470.000 M € a year.

b) Operational Benefits quantified (cumulative figures between 2008 and 2020):

- In cost-effectiveness: 8,000 M €.
- In increasing capacity and reducing delays: 9,000 M €.
- In fuel savings: 8,000 M €.
- In service's quality: 2,000 M €.

One of the most important challenges of SESAR with respect the current ATM, which establishes the basis of this project and will be present in the system's development, is the introduction of the Trajectory Based Operations (TBOs), implying the use of 4D gate-to-gate precision trajectories (trajectories defined in the 3 spatial dimensions together with a time-stamp), also known as Business Trajectories (BT) in the SESAR's terminology for civil flights. The idea behind this concept is that both airlines and providers of Air Navigation Services (ANSP) and airport operators come to an agreement, in a collaborative way, in order to find the optimal door to door flight plan for each of the scheduled flights [1].

During the negotiation process between controllers and airlines, the Business Trajectory is called Shared Business Trajectory (SBT). In this process, the company will propose the more efficient trajectory for them to cover a certain distance between two airports. Then the ANSP accepts or denies the trajectory depending on the constraints of airspace and airports. This negotiation process is repeated until a trajectory is found and accepted by both agents (except in critical situations where there is no time for negotiation, in which case will prevail the trajectory imposed by the ANSP) [5].

Once one SBT is accepted by both agents, it passes to be called Reference Business Trajectory (RBT). From that moment, this trajectory is considered optimal, because it takes into account the constraints of the air traffic controllers and airline preferences (assuming the choice of the least cost trajectory).

The Business trajectory (BT) requires a precise definition in the 4 dimensions, i.e., in the three spatial dimensions and in the time (see Fig. 3). For this reason his name is often replaced by 4D trajectory.

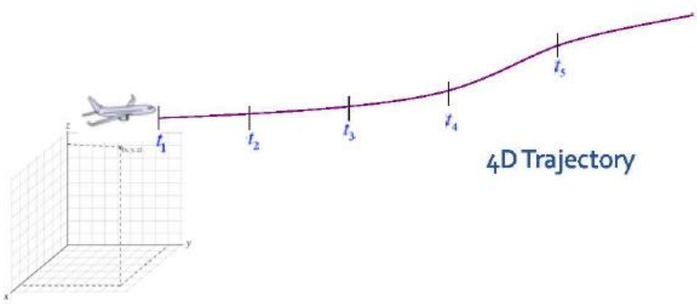
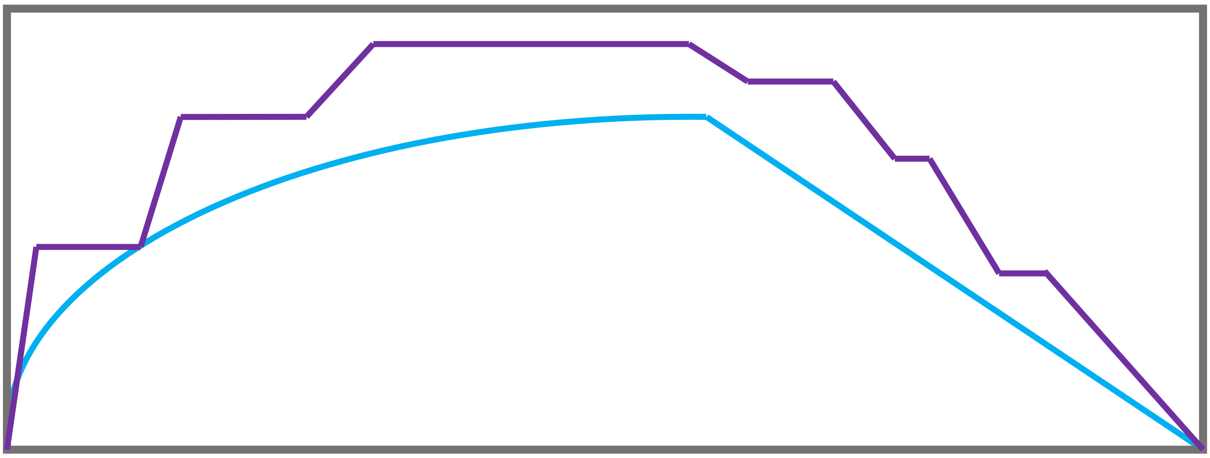


Figure 3: 4D Trajectory Dimensions

Source: Ruiz, 2010, [5]

In the absence of airspace or traffic constraints, “an optimal trajectory” basically consists of following the most direct route possible and maintaining a uniform acceleration during take-off and up to the maximum desired altitude where the aircraft can start a smooth and continuous descent up to the runway, also using a uniform maneuver without accelerations until get into contact with soil.

In this way, by applying the optimum vertical profile it is achieved that the planes are less time in the air (thus increasing the capacity of the sky), whereas progressive acceleration allows a significant fuel savings and trip-time, as well as a significant decrease of the CO2 emitted [5].



Optimal profile: constant and progressive elevation and descent (no steps).

Traditional profile: staggered elevation and descent (with acceleration steps).

Figure 4: Comparison between the optimal vertical profile and the traditional one

Source: Ruiz, 2010, [5]

Another important concept inside the SESAR context is the term called “Strategic de-confliction”. This term encompasses those actions whose objective is to minimize the amount of conflicts, appearing because of a loss of separation between two aircraft. These actions are taken by Network Manager (NM) which are always executed when the takeoff time is known with sufficient accuracy, or when the flight is airborne, i.e. always with enough time to guarantee a previous Collaborative Decision Making process (i.e. Collaborative Flight Planning). It excludes tactical instructions and clearances that need an immediate response, but includes activities such as dynamic route allocation [1].

In this context, strategic conflict management and traffic synchronization, would lead to pre-deconflicted 3D routes subject to dynamic refinement or adjustment during flight (i.e. 4D contracts), as a result of the introduction of automation support to conflict detection, situation monitoring and conflict resolution, the main initiatives for increasing airspace capacity in the period up to 2020. Thus, the SESAR's role to achieve these capacity perspectives (extra capacity is required since an increment of the current air traffic levels in a factor of x2 is forecasted by 2030) is to deploy tools to assist the controller with complex situations and to reduce complexity by strategic deconfliction measures.

A progressive improvement in the accuracy of ground-based trajectory prediction through reduced flight uncertainty will lead to improved performance of controller support tools. The consideration of 4D trajectories together with the representation of wake vortex with ECAC-Wide perspective is considered necessary in this work to allow 4D trajectories, to be planned safely with regards the potential hazards of WVE during all phases of flight.

2.2 Wake Vortex Phenomenon

2.2.1 Origin and stochastic behaviour

All aircraft produce wake turbulence, also known wingtip vortices or wake vortices when they are airborne during a flight. Wake vortices are formed any time an aerofoil is producing lift. Lift is an upward force generated by the creation of a pressure differential over the wing surfaces. The lowest pressure occurs over the upper surface of the wing, and the highest pressure is formed under the wing. Air will always move towards the area of lower pressure. This causes it to move outwards under the wing towards the wingtip and curl up and over the upper surface of the wing. This starts the wake vortex (see Fig. 5) [15, 17].

The same pressure differential also causes air to move inwards over the wing. Small trailing edge vortices, formed by outward and inward moving streams of air meeting at the trailing edge, move outwards to the wingtip and join the large wingtip vortex. Swirling air masses trail downstream of the wingtips. Viewed from behind, the left vortex rotates clockwise and the right vortex rotates counter-clockwise (see Fig. 5).

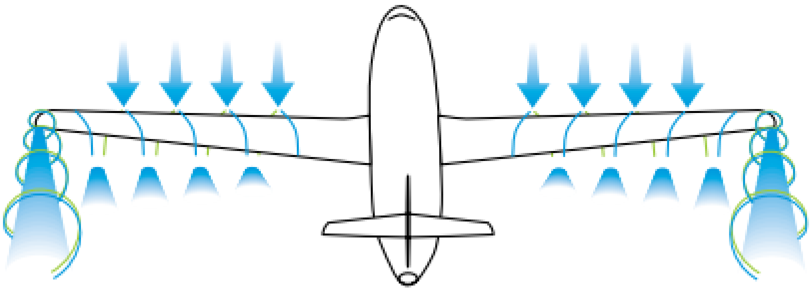


Figure 5: Wake vortex roll-up process

Source: [15]

Typically, a vortex develops a circular motion around a core region. The core size can vary in size from only a few centimetres in diameter to a metre or more, depending on the type of aircraft. From larger aircraft, the speed of the air inside this core can be up to 100 metres per second. The core is surrounded by an outer region of the vortex, as large as 30 metres in diameter, with air moving at speeds that decrease as the distance from the core increases. Wake vortices can persist in the air for three minutes, or longer, in certain conditions [15, 17].

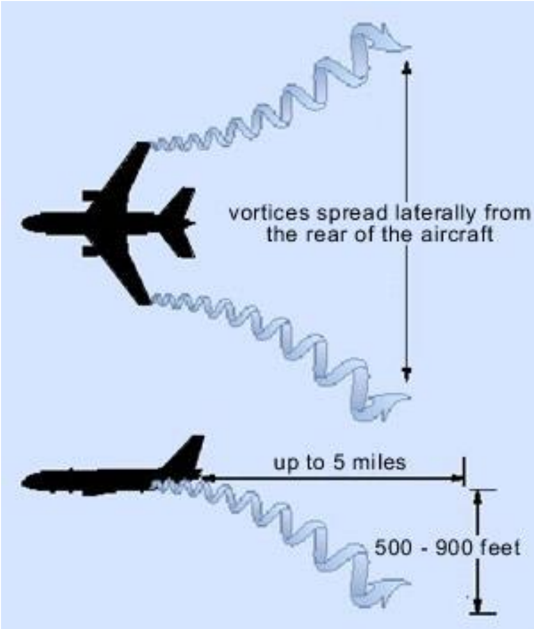


Figure 6: Plan view and lateral view of wake vortex

Source: [17]

In order to obtain a deeper knowledge about the nature of this phenomenon for the purpose of this work, it is important to analyse its characteristics and its development and evolution over time. These characteristics are:

- **Circulation**

The strength of the vortex generated after this roll-up process, as a result of the combination of same direction vortices, is expressed by the circulation or rotation velocity.

$$\Gamma_0 = \frac{W}{\rho\, V_L b_L \frac{\pi}{4}} \quad (1)$$

Equation (1) is the KuttaJoukowsky relation, which is used to determine the initial circulation, Γ_0, of a single vortex, with m2/s as unit. If we observe the previous expression, the circulation depends only on air density (ρ) and aircraft parameters as weight (W), velocity (V_L) and wingspan (b_L). The vortices are initially positioned at a distance of about $b_0 = b_L \frac{\pi}{4}$ from each other, which is less than the aircraft's wingspan [4].

- **Descent**

Another important characteristic of the vortices is their tendency to descent from their initial altitude, phenomenon influenced by buoyancy effects of the atmosphere that causes a constant upward pushing force. This effect is called rebound. So, we can conclude that with a stably stratified atmosphere, the buoyancy force additionally reduces the descent velocity. Equation (2) describes the initial descent (ω_0) of a vortex pair.

$$\omega_0 = \frac{\Gamma_0}{2\pi b_0} \tag{2}$$

It is important to mention that buoyancy force depends on the stratification of the atmosphere and, at the same time, this last one, is dependent of the temperature gradient. Thus, a positive, or small negative temperature gradient will induce a large buoyancy force, making the wake rebound, while a large negative temperature gradient will decelerate the wake's descent. The typical pattern of wake vortex in standard atmosphere is shown in the Fig. 7 [4].

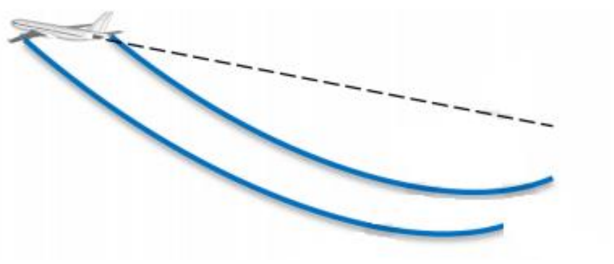


Figure 7: Typical pattern of wake vortex in standard atmosphere

Source: DE CLEYN, 2011, [4]

- **Decay**

After this so-called roll-up process, the main trend is that the vortex strength is reduced by the time. This process depends on the internal characteristics of the vortex and external characteristics of the surrounding atmosphere. This phenomenon is produced when the stably rotating airflow is broken down, as a result of the interaction with internal and external vorticities. The decay of the vortices occurs in two phases, as is shown in Fig. 8.

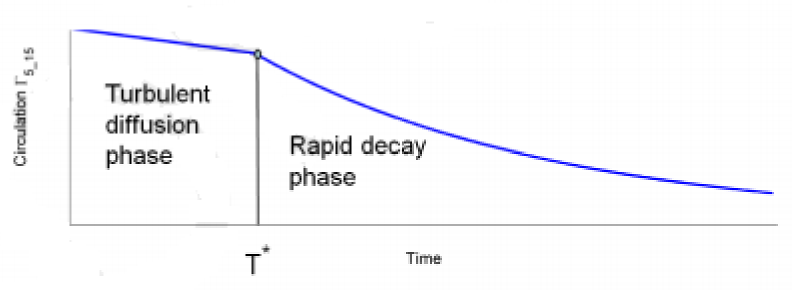


Figure 8: Two phase decay

Source: DE CLEYN, 2011, [4]

In the first phase, called turbulent diffusion phase, the circulation decays rather slowly. However, after a certain time T*, which mainly depends on the atmospheric conditions, the rapid decay phase starts and it is characterized by a higher rate of the vortex decay.

- **Transport**

Considering an atmosphere with non-zero wind, the wakes will move with the wind after their generation. This is a phenomenon called wake vortex transport. The position with respect to each other does not vary during the transport, maintaining both in parallel. The transport of the vortex pair depends on the combination of the velocity field of the vortex particles and those of the atmospheric velocity components. These include wind, but also wind shear and stratification [4].

2.2.2 Potential hazards and safety distances

The International Civil Aviation Organization (ICAO) is an organization that develops regulations and recommendations for international air navigation. In ICAO's 6108873-DOC-4444 documentation different safety distances are described to vertical, horizontal and temporal level that should be preserved between aircraft.

Horizontal safety distances include the lateral and longitudinal separations. A way to simplify the ICAO regulation is to assume that all aircraft will be controlled by radar, so you can take a sufficiently conservative separation measure, which is considered to be 5NM (9,3 km) as horizontal safety distance between aircraft for en-route airspace, which can be reduced to 3NM (5.5 km) in the case of TMAs [5].

For vertical safety distances a minimum separation of 1000ft (300 m) is currently defined for those airspaces where a vertical separation called RVSM (Reduced Vertical Separation Minimum) is applied. This is the case in Europe, where the rules of RVSM for all levels of civil aviation flight were implemented since 2002, with the idea of increasing airspace capacity while maintaining the same level of safety [5].

The RVSM procedure was extensively investigated on potential safety risks before implementation. The probability of an aircraft wake descending 1000 ft below its generation altitude, while being of sufficient strength to cause major turbulence was concluded to be very low, but this investigation did not consider a thorough analysis of real atmospheric circumstances, and the impact of new weight classes such as the very heavy aircraft.

By modelling the wake vortex transport and decay taking into account a meteorological forecast, it could be demonstrated that at cruise altitudes, heavy aircraft may produce a wake that can sink as low as 1000 ft below its generation altitude and remain sufficiently strong to cause severe turbulence. This statement was verified by a number of wake vortex encounter incident reports, where aircraft encountered severe turbulence caused by the wake of a heavy aircraft passing overhead, one flight level above. Among the meteorological conditions that can favour this phenomenon are a negative temperature gradient, low atmospheric turbulence and wind velocity.

Therefore, it is very important to revise these safety distances, because wake vortex encounters can cause serious consequences. Reported wake vortex encounter incidents indicate that trailing aircraft may suffer very short period vertical accelerations between -1 and 3g. These often result in altitude excursions and violent roll accelerations, which may bring an aircraft in conflict with other traffic on higher or lower flight levels. The combination of vertical, lateral and roll accelerations can bring crew or passengers that are not strapped down by their seatbelt out of balance and can cause severe injuries. If these rolling motions occur in close proximity of the ground, there may be no more time to recover and avoid impact with the ground.

The hazards of WVE (Wake Vortex Encounter) in en-route phase of flight can become much severe in the context of the modernization of the ATM (SESAR 2020+), in which a deep revision of the procedures and separation standards is going on. Also much more demand and less homogenous aircraft mix is expected (i.e. heavier aircraft sharing airspace with lighter ones), thus increasing the probability of encounter and the severity, parameter that depends on the rotation velocity of vortex, the stability factors of the affected aircraft, and the space geometry where these ones occur, being very important for the purpose of this project [22].

For instance, see Fig. 2 in which airspace and flight operations could be not constrained by the current Flight Level Scheme that is used to separate traffic (which is in line with future SESAR ATM).

For this reason, and considering the ATM’s future capacity and safety requirements, we need new technology and systems capable to know, in every moment, the wake vortex position and its strength, because it can be very variable depending on the aircraft’s characteristics and specially on the meteorological conditions.

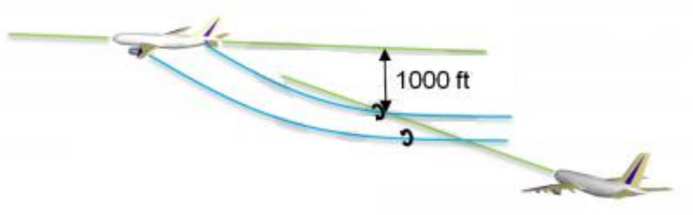


Figure 9: Wake Vortex Encounter

Source: DE CLEYN, 2011, [4]

2.3 Spatial Data Structures (SDS) and their use in to tools for ATM

In order to develop the Wake Vortex Encounter Predictor System proposed in this project, we will use the current conflict detection technology based on Spatial Data Structures (SDS).

A SDS is a tool capable to represent a particular spatial region (in this case an air sector) being a very easy way in terms of computational treatment. Actually, a SDS is a database in which every memory position represents each of the discrete coordinates of the modelled sector. From a conceptual point of view it can be thought as a mesh of discrete points distributed throughout the space region, in which the spatial information are stored (associated with such a coordinate) making easy the access to this data, applying simple mathematical formulas.

Fig.10 illustrates this concept. Note that inside this three-dimensional SDS, represented by a cube, a discretized 4D trajectory is stored (different 3D positions of an aircraft in different discrete time steps).

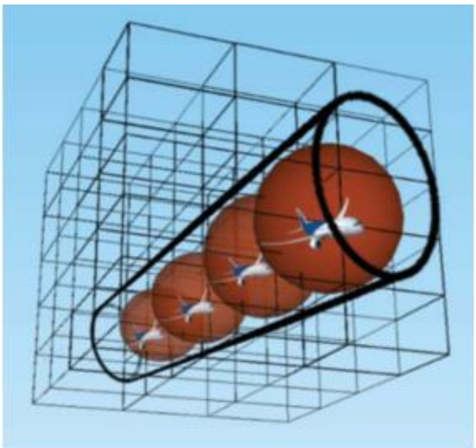


Figure 10: SDS conceptual representation

Source: Ruiz, Piera, 2013, [1]

The tube that envelops the trajectory establishes the space that should be reserved for each trajectory in the SDS, considering the safety distances between aircraft and the behavior of wake vortex that can cause instability if there is an encounter. Given that a wake vortex is usually propagated down 1000ft (300m), the radius of the tube is 300 meters.

To understand properly the system, it is necessary to know the *granularity* concept. The *granularity* or *resolution* of a SDS is the distance between the constituent discrete points of the SDS.

Depending on the granularity chosen we will have a specific quality referring to the state-space stored and a certain degree of efficiency at processing and managing the spatial data. Therefore, an excess of resolution may lead to a loss of computer performance as well as to an inoperable amount of memory requirements, whereas a lack of resolution may lead to lose some important objects of the space [1].

So, to determine the appropriate resolution, taking into account the region that we want to model, we have to consider the next factors among others:

- The size of the physical airspace to model
- The size of the objects to be stored in the database
- The speed at what these objects move
- The quantity of memory available in the computer
- The speed of execution of the algorithms

The SDS should be implemented as a large one-dimensional array stored in a RAM memory, which can be conceptually drawn as a table that contains as many rows as coordinates are in the modelled airspace and as many columns as aircraft / processing trajectories are. See Fig.11 to observe an example of spatial data and a specific SDS configuration.

The data stored in the SDS consists of both spatial information, like the discrete representation of either the trajectory or its corresponding enveloping safety tube, and non-spatial information like the identification number of the flight and the timestamps associated with each discrete trajectory or tube, which can be stored explicitly in one or several fields in the same database record.

The representation of the discrete airspace coordinates by each row/record and their sequential classification in the database, make easy the access for writing/reading applying the following equation:

$$\mathbf{SDSrow\ (x, y, z) = x \cdot Y + y \cdot Z + z + 1} \qquad (3)$$

being SDSrow the record position inside the database that stores the information of a particular coordinate (x, y, z), with $x \in [0 , X - 1]$, $y \in [0 , Y - 1]$ and $z \in [0 , Z - 1]$, and being X, Y and Z the maximum number of discrete coordinates of each spatial dimension, respectively.

Note that in Fig. 11, the information stored in each record is about the aircraft/flight that will use the corresponding coordinate and the time period in which this one will be occupied. In the fourth record it can be found that aircraft with id = 2 is expected to make use of the coordinate (0, 0, 3) in the time window from second 520 until second 640.

	Reservation 1				Reservation 2				Reservation 3				…	Reservation n			
Coords	Aircraft	TWon	TWoff	5D, 6D…	Aircraft	TWon	TWoff	5D, 6D…	Aircraft	TWon	TWoff	5D, 6D…	…	Aircraft	TWon	TWoff	5D, 6D…
(0,0,0)	0	0	0	0	0	0	0	0	0	0	0	0	…	0	0	0	0
(0,0,1)	0	0	0	0	0	0	0	0	0	0	0	0	…	0	0	0	0
(0,0,2)	15	50	170	5D, 6D…	0	0	0	0	0	0	0	0	…	0	0	0	0
(0,0,3)	2	520	640	5D, 6D…	4	76	196	5D, 6D…	0	0	0	0		0	0	0	0
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	…	⋮	⋮	⋮	⋮
(0,1,0)	8	99	219	5D, 6D…	0	0	0	0	0	0	0	0	…	0	0	0	0
(0,1,1)	**3**	**34**	**154**	**5D, 6D…**	**4**	**32**	**152**	**5D, 6D…**	7	879	999	5D, 6D…	…	0	0	0	0
(0,1,2)	0	0	0	0	0	0	0	0	0	0	0	0	…	0	0	0	0
(0,1,3)	0	0	0	0	0	0	0	0	0	0	0	0	…	0	0	0	0
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	…	⋮	⋮	⋮	⋮
(0,2,0)	6	565	685	5D, 6D…	15	233	353	5D, 6D…	0	0	0	0	…	0	0	0	0
(0,2,1)	0	0	0	0	0	0	0	0	0	0	0	0	…	0	0	0	0
(0,2,2)	76	12320	12440	5D, 6D…	78	800	920	5D, 6D…	0	0	0	0	…	0	0	0	0
(0,2,3)	0	0	0	0	0	0	0	0	0	0	0	0	…	0	0	0	0
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	…	⋮	⋮	⋮	⋮
(1,0,0)	0	0	0	0	0	0	0	0	0	0	0	0	…	0	0	0	0
(1,0,1)	**4**	**590**	**710**	**5D, 6D…**	15	88	208	5D, 6D…	**19**	**680**	**800**	**5D, 6D…**	…	0	0	0	0
(1,0,2)	1	79	199	5D, 6D…	2	800	920	5D, 6D…	3	550	670	5D, 6D…	…	n	10098	10118	5D, 6D…
(1,0,3)	0	0	0	0	0	0	0	0	0	0	0	0	…	0	0	0	0
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	…	⋮	⋮	⋮	⋮

Figure 11: SDS content example

Source: Ruiz, Piera, 2013, [1]

As you can see, the coordinate (0, 1, 1) is used by two different aircraft in overlapped time periods (TWon refers to the moment when the aircraft occupies the coordinate and TWoff to the time in which the aircraft leaves) information that allows the detection of a conflict between both in this time-space. This is the mechanism used by current systems based on SDS to detect conflicts.

However, there are coordinates that are not booked for any aircraft, and this system requires as much space for these coordinates as N reservations are done, causing an unnecessary increase in memory use (Examples: (0,0,0) and (0,0,1)).

The use of SDSs allows the storage of the entire State-Space description of the air traffic at a given time and its evolution over the time. All the processed trajectories will remain stored as a “4D snapshot” of the ATM system which makes the SDSs to be an interesting tool for traffic separation purposes since:

• The SDS acts like a powerful spatial pruning filter which it makes available a continuous updated air traffic state due to its capability to process a large number of trajectories within a large time-window look-ahead (e.g. 2 hours) in few seconds, which allows to tackle some ATM sources of uncertainty.

• The SDS can be used through causal models in order to efficiently access and explore the SS information stored inside, which enables an improvement of the decision-making with a global optimization scope due to the capacity to evaluate all the potential network emergent dynamics.

2.3.1 SDS applications for ATM purposes

Spatial Data Structures have been explored in different previous researches as a highly configurable technique easily used in different ATM applications, whose implementation to store the relevant state-space presents excellent results in collision avoidance algorithms, in time-based MTCD prototype for Terminal Maneuvering Areas and in the strategic conflict detection and resolution process of thousands of trajectories at European ATM level.

2.3.1.1 Collision Avoidance

A common method to detect conflicts between trajectories is the application of pairwise CD algorithms, which consist on distance calculations between the point-mass positions occupied by the aircraft at each given time-step, comparing each different pair of 4D trajectories. In Fig.12 we can observe how the dotted lines are representing the 4D coordinates that belong to the same time instant.

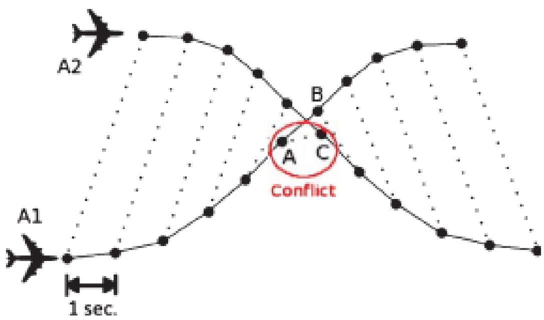


Figure 12: Representation of a pairwise algorithm

Source: Ruiz, Piera, 2013, [1]

However, these pairwise algorithms are not efficient because of its computational complexity of quadratic order, o (n^2), thus it is necessary the use of some kind of filters (e.g. flight level pre-filter, time-skipping strategies. . .) in order to avoid unnecessary comparisons, improving the time performance of the algorithms.

As an alternative to this method, we can use SDSs as a powerful spatial pruning filter that linearizes the runtime/complexity, o (n), of the collision avoidance algorithm. This one, considers the fact that collisions can only occur when two boids are at a certain short distance, i.e., when agents are geographically correlated.

Thus, the characters “pre-sorted” are kept in the SDS depending on their location in space, allowing a quick identification of the neighbourhood in which they are at a given time-step, achieving a reduction of the amount of these pairwise computations between trajectories. Therefore, firstly, the neighbours are identified at each time step (i.e. boids geographically correlated at close locations) in order to filter the amount of pairs to be compared with and, after that, the spatial distances among those still-remaining pairs of trajectories are checked, performing the detection of conflicts for each boid [1].

Fig. 13 shows this concept. A boid is circled representing its safety area. Its associated neighbourhood are formed by the grid-cells overlapped with the circle (shadowed) at the time of simulation. If there are boids inside the neighbourhood at the same time instant of simulation they will be candidates for a pairwise distance comparison. Note that Fig. 13 represents a top-view of a 2D scenario, but the same concept can be extended to 3D.

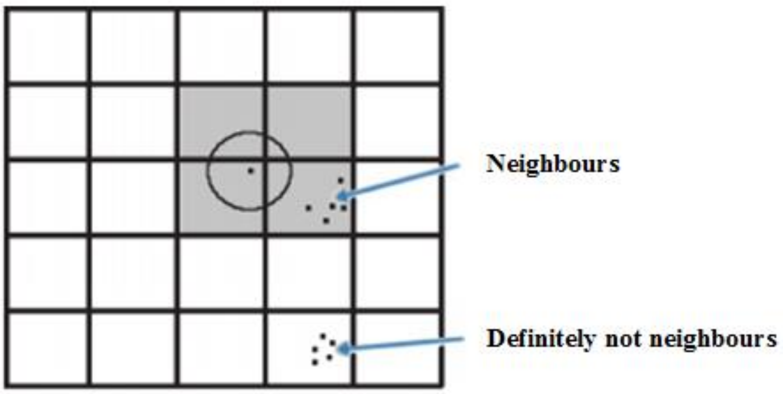


Figure 13: Neighbours search to filter some of the pairwise comparisons

Source: Ruiz, Piera, 2013, [1]

This algorithm has the capability to update the content of the SDS at each simulation time-step so that the temporal information is not stored. Therefore, the algorithm considers a SDS configured to store only the id information of the boids that are located over the surface of the grid cells (the bin-volume in case of a 3D scenario), determining the next trajectory-position. Consequently, if at current time of simulation a conflict/potential collision is detected, the boids react by changing their trajectories at next time of simulation.

However, when a large amount of boids is contemplated, the emerging dynamics effects can make the avoidance of collisions unfeasible. Therefore, the whole algorithm and SDS architecture must be reconfigured for ATM planning applications, such as strategic de-confliction and/or tactical management of conflicts in the European airspace, in which a look-ahead planning horizon is required and domino effects must be considered, together with ATM uncertainties.

2.3.1.2 MTCD based on SDS

The technique explained above based on SDS has been applied to obtain a MTCD algorithm for Terminal Maneuvering Area (TMA). This one has the function of check time-distance separations between approaching/landing aircraft according to the wake vortex generated by other aircraft. So, the SDS is used in this algorithm in order to take a 4D snapshot of the scenario to observe the trajectories executed by aircrafts that are flying over the modelled area and the vortices generated.

In this particular CD algorithm, the SDS' logical architecture allows the booking of spatial discrete cells by each aircraft/flight according to their expected vortex dimension, concept shown in Fig.11 as the storage of the reservations of resources at each database record.

The SDS configuration used in this algorithm is different that the used in the conflict avoidance algorithm mentioned previously. In this last one each SDS record represents a grid area/volume, also called bin, while MTCD algorithm treats every record as a single discrete coordinate, i.e. a spatial resource that only can be used by one aircraft at a given time, considering a fixed time window of 120 seconds for each flight wake vortex's duration to simplify.

The process followed when we are using the SDS to detect conflicts, is the next one [1]:

- Firstly, at the moment of storing the surface tube-point, we have to identify its corresponding coordinates, and then, the algorithm checks if the coordinates are already occupied. Thus the first column is read and if its value is equal to zero, i.e. is empty, it means that no other aircraft intend to use such a coordinate, allowing the storage of this spatial resource as a conflict free position (e.g. in Fig. 11 aircraft #15 could freely reserve (0, 0, 2) since there was no previous reservation).

- If the first column is not empty, the coordinate will be used by another aircraft in a certain period of time. For this reason, the algorithm compares the time windows to check if they overlap, in which case a conflict is detected and the Conflict Resolution (CR) system is informed (e.g. in coordinate (0, 1, 1) there is a conflict between aircraft #3 and #4 because their utilization time-windows overlap). If the time windows are not in conflict, it means that the coordinate might be booked in the following column without informing the CR (e.g. in (0, 2, 0) aircraft #15 could make a reservation with no conflict with previous reservation of aircraft #6).

In next columns the same procedure is applied sequentially (e.g. firstly, the coordinate (1, 0, 1) was booked by aircraft #4 and later by aircraft #15. They were not in conflict because their utilization time windows did not overlap. However, when a third aircraft #19 made a reservation for the same coordinate, the algorithm detected a conflict with aircraft #4, without affecting the second one #15, which was still conflict-free).

The implementation of SDSs as efficient spatial information managers, capable to provide a 4D snapshot to analyze the scenario evolution, makes it an interest tool for ATM planning purposes. This is a conclusion obtained through several simulations in Gran Canaria TMA, validated with B738. However, the initial logical architecture of SDS (shown in Fig.11) is not prepared to consider wide airspaces and/or large amount of trajectories, due to its lack of efficiency in the computer main memory management, which makes it impractical for strategic European ATM dynamic planning.

2.3.1.3 Strategic Deconfliction based on SDS

SESAR WP-E launched a project called STREAM (Strategic Trajectory de-confliction to Enable seamless Aircraft conflict Management) whose objective is the development of innovative computational-efficient Conflict Detection and Resolution (CD&R) algorithms in order to apply strategic de-confliction considering thousands of trajectories and taking into account Airspace Users (AUs) preferences and network constraints, within just few seconds or minutes. This is aimed at enabling traffic to be de-conflicted for wide airspace regions and permitting large look-ahead times of order of hours (e.g. two or three hours) [1].

Strategic de-confliction STREAM algorithms make possible the building of a micro-model framework that represents all the traffic at European airspace scale which allows its efficient management through a set of individual 4D business trajectories and by suggesting strategically de-conflicted trajectories which closely match AUs preferred ones in a free-route environment, i.e. not constrained by pre-structured routes as occurring today.

Under the STREAM project, Spatial Data Structures were explored as a tool with capability to efficiently store spatial data (e.g. 4D trajectory information) at the time when the conflict detection among all SBTs/RBTs is performed, allowing an effective Conflict Detection (CD) process, with excellent results in terms of time performance.

A part from that, the efficient database access methods, have been a key factor for the development of new tools to analyze the entire ATM State-Space information under a global scope.

This enables for example a complete and precise identification of the domino effects that the emergent dynamics may cause in the network, because all the processed trajectories will remain stored as a "4D snapshot" of the European ATM [1].

2.3.2 SDS enhancements to reduce the memory and time

In spite of the benefits of SDS-based CD algorithm in terms of information management, this one presents some shortages in its scalability to manage the whole European ATM. For this reason, new SDS concepts were created considering two innovations explained below [1]:

• **Relational SDS (RSDS):** allows a reduction of approximately 98% in terms of main memory required by original SDS architecture, being a key factor to enable the storage of the overall European ATM 4D state-space with a look-ahead of several hours. In addition, the computational time performance and the rest of the advantages of the original algorithm are maintained.

• **Time-Space Data Structure (TSDS):** adding a fourth dimension (i.e. the time) to the original SDS architecture, an efficient management of 4D data and an improvement of the run-time performance of the CD process is achieved when a large amount of trajectories is considered. In order to efficiently adapt the ATM planning to uncertainties, perturbations and system disruptions, the fast processing time achieved referring to the CD process becomes to a key factor, since the whole state-space can be updated in real-time.

2.3.2.1 Relational SDS (RSDS)

The SDS-based CD algorithm presents an immense growth rate of memory required when the number of trajectories to be processed is increased, constituting an important shortage that drastically restricts the size of the modelled sector and/or the granularity of the SDS and/or the amount of trajectories that can be analyzed. Therefore, it seems necessary the use of the memory in a more efficient way.

In order to calculate the necessary amount of memory to store the logical SDS architecture, we have to know the number of constituent database records (i.e. rows), which is illustrated in Fig.11. The total number of rows is the result of multiply the number of records corresponding to each spatial dimension, X · Y · Z, which is obtained dividing its length (i.e. lengthX, lengthY and lengthZ) by the size of each grid cell or bin, i.e. SDS granularity [1]. A part from that, the SDS's configuration allows the storage of N potential reservations for each record, which occupy B bytes each in the main memory (B= 4 bytes in the MTCD application above). Thus, the total amount of memory required to allocate this SDS architecture is:

$$\text{total Memory SDS} = X \cdot Y \cdot Z \cdot B \cdot N = \frac{\text{lengthX}}{\text{sizeBinsX}} \cdot \frac{\text{lengthY}}{\text{sizeBinsY}} \cdot \frac{\text{lengthZ}}{\text{sizeBinsZ}} \cdot B \cdot N \qquad (4)$$

Since a new trajectory is introduced into the problem at each time instant, the SDS' memory positions (of size B each) are increasing continuously in an amount equal to the number of rows required to represent the airspace scenario of interest, which is usually huge due to the airspace sizes needed for ATM applications.

Moreover, note that when we are working with three-dimensional granularity, the amount of memory required grows cubically (e.g. reducing the size of the bins in a factor of 10 in each spatial dimension implies a memory growth of order 103), which makes impractical the allocation of the SDS in the current computers' main memory.

In addition, the SDS architecture presents another important shortage which is the impossibility that an aircraft can use a same coordinate more than once (e.g. during holding trajectories), due to the logical architecture's configuration that only allows a reservation per aircraft at each SDS record.

A part from that, there is another source of inefficiencies: the amount of empty SDS' memory positions, i.e. that ones that are not storing any king of useful information. The result is a SDS with a lot of fields setting to zero that are consuming memory needlessly, since the amount of memory required for a potential reservation of any coordinate/record has been already allocated.

A way to achieve more efficient information management is by using different databases, one optimized to store the basic SDS' structure, i.e. the memory positions that configure the modelled airspace to manage the spatial data efficiently, and another one optimized to store the associated non-spatial information of the trajectories/reservations. Using this method we are minimizing the immense growth of the SDS due to the fact that the information of those databases can relate each other through database pointers to outside database records, becoming into relational databases (see Fig.14) [1].

Base-SDS (BS)

Coords	Pointer to a STI position
(0,0,0)	0
(0,0,1)	0
(0,0,2)	12295
(0,0,3)	0
(0,1,0)	530
(0,1,1)	5
(0,1,2)	0
(0,1,3)	0
(0,2,0)	43
(0,2,1)	0
(0,2,2)	12
(0,2,3)	0
(0,2,2)	0
(0,2,3)	1000443
(0,3,0)	0
(0,3,1)	0
(0,3,2)	63
(0,3,3)	969
(1,0,0)	0

Stacked Trajectory Information (STI)

Position	Aircraft ID	Twmin	Twmax	Pointer to a STI position
1	4	5	125	0
2	2	6565	6685	4234
3	2	2534	2654	54
4	4	76	196	764
5	3	34	154	10
6	3	2565	2685	75
7	67	753	873	99
8	8	56313	56433	456
9	6	565	685	0
10	4	32	152	14
11	76	56567	56687	786
12	345	2346	2466	0
13	6	7	127	0
14	7	879	999	0
15	9	9976	10096	0
16	1	6	126	867

Figure 14: Relational SDS architecture

Source: Ruiz, Piera, 2013, [1]

To achieve the equivalent functionality of non-relational SDS (Fig.11) these two databases are required:

- **The Base SDS (BS):** is a database formed by an amount of records equal to the number of discrete coordinates of the airspace, i.e. with the same amount of records than in Fig. 11 (i.e. one row for each discrete coordinate). However, in this case the BS uses only 1 field/column per record (usually occupying 4 bytes) instead of N fields/columns for N trajectories ($4 \cdot N$ bytes). The content of this unique column may be zero if the coordinate will not be occupied by any aircraft or may store a pointer to a record position of the second database in the contrary case.

- **The Stacked Trajectory Information (STI):** the advantage of this database is that only stores that coordinates going to be used by all the trajectories, eliminating one of the most important sources of inefficiencies. However, the most important aspect to mention about this one, is its particularity to store all the information about trajectories in a stack (i.e. FIFO order), allowing the optimization of the information storage since no empty records/memory-positions are present at this database (saving lot of memory with respect the non-relational SDS). Note that in Fig. 14 the STI is constituted for 4 columns. In this example, 3 of them are used to store the non-spatial information of a booking, as it happens in the non-relational one. However, the column number 4 stores a pointer to another STI position, if later reservations for the same coordinate are made by same or different trajectories. Therefore, is a method to establish the order of the coordinates' use and a way to know how many aircrafts will use each one of them.

If you compare the information included in both SDSs (Fig. 11 and 14), you will note that they store the same spatial and non-spatial information. If the pointer to STI stored in BS is not zero (see Fig.14), at least one booking was done for this coordinate. The value of the pointer indicates the position of the previous booking stored in STI (e.g. for coordinate (0, 1, 1) a pointer is stored to position 5 of STI). In order to know if there is a conflict between the current and the previous booking you only have to access to this record/row in the STI and check if their utilization time-windows overlap.

After the checking performed to know if there is a conflict with such a previous booking, the next step is to search if another pointer to STI is present in the fourth column. If the pointer is set to zero, it means that it does not exist another previous booking for such a coordinate, allowing the storage of the current booking in the last free position of the STI, always following the FIFO order. Moreover, the pointer to that record (the new booking) is stored in the fourth column of the actual current record of the STI. However, if the pointer is not zero, the algorithm is executed sequentially as many times as it is necessary until a free position is found to complete the booking (e.g. in Fig. 14 positions 10 and 14 are sequentially checked) [1].

The main advantage of the RSDS design is its capability to adjust the amount of memory required used to configure the BS to the size of the airspace sector to model and the desired granularity, since this architecture allows unlinking its size to the amount of trajectories considered in the problem. Therefore, with a larger amount of trajectories, its size does not increase. Thus, the total memory required for the BS construction is:

$$\textbf{total Memory BS} = \mathbf{X \cdot Y \cdot Z \cdot P} = \frac{\textbf{lengthX}}{\textbf{sizeBinsX}} \cdot \frac{\textbf{lengthY}}{\textbf{sizeBinsY}} \cdot \frac{\textbf{lengthZ}}{\textbf{sizeBinsZ}} \cdot \mathbf{P} \tag{5}$$

being P the amount of bytes required to store a pointer to a record (typically P = 4 or P = 8) and the rest of the parameters with the same meaning as in Eq. (4). On the other hand, the amount of memory needed for the STI is calculated by:

$$\textbf{total Memory STI} = \mathbf{N \cdot L \cdot (B + P)} \tag{6}$$

being N the number of trajectories to be processed, L the average amount of time-steps per trajectory and B and P with the same meaning as in Eq. (4) and Eq. (5). Therefore, the total memory space needed to store the STI when N increase grows with a constant linear rate much lower than in the case of the original logical SDS architecture.

The results obtained through several RSDS configurations adapted to different ATM scenarios, have demonstrated the memory management improvement in terms of important reductions in the quantity of necessary memory. This reduction are comprised between 95-99%, as the most ATM practical cases show, which implies a much more efficient way to manage the main memory in comparison with the use of non-relational SDS for the same scenario.

The 4D snapshot of the ATM is still available in the RSDS showing all the state space of the problem, being possible to extract and summarize useful information to use the new CR algorithms in the necessary cases in order to find efficient and optimal conflict-free trajectories.

2.3.2.2 Time-Space Data Structure (TSDS)

The kind of SDS for collision avoidance introduced in section 2.3.1.1 had the capability of storing the trajectories' information depending on the grid cell/bin that they were occupying at a given time instant of simulation. This information was useful to detect how many pairwise comparisons were necessary to perform, acting as a pruning filter that improved the performance and the efficiency of the CD process.

However, for ATM planning purposes a similar CD algorithm could be contemplated, as an algorithm capable to refresh the SDS at each time of simulation and to create a registry composed by the detected conflicts that the CR module will have to solve.

Nevertheless, in order to have a network global view of the user's flight intentions and to have more capability to predict the potential emergent dynamics, a historical record of the evolution of all trajectories over the time (like a 4D snapshot) is highly beneficial and necessary to guarantee a more efficient management of the European air traffic.

Figure 15: TSDS conceptual representation

Source: Ruiz, Piera, 2013, [1]

To obtain the 4D ATM snapshot, we can modify the SDS' configuration to store the expected time-window in which each coordinate will be used at each cell/bin reservation, in a similar way than seen in Fig.11. However, this option has a shortage since the benefits of using such SDS configuration decrease when there is a large amount of trajectories willing to use the same SDS cells/bins, even in different time instants. This shortage proceeds from the computational cost that the comparisons with previous trajectories suppose, even if they are filtered by the time-window of utilization.

By adding the temporal dimension (the 4th one) to the SDS' structure, we can achieve a cost reduction of comparing with previous trajectories, since that reservations for different expected times of utilization, even for the same spatial resource, are considered as different reservations for different time-space regions. Therefore, this method becomes to a more powerful filter than only using spatial queries, since by the utilization of time-spatial ones we can achieve a higher reduction of the pairwise comparisons [1].

As a result, a new SDS configuration is obtained named Time-Space Data Structure (TSDS). Conceptually, a TSDS can be thought as a set of T different SDSs, one for each discrete portion of time (see Fig. 15). The TSDS configuration has the capability to support time-windows as discrete portions of time, not only considering time-instants (Fig. 15 shows this concept, since a set of T different SDSs, each one storing 4 time-steps of different 4D trajectories executed in different time-windows, is illustrated).

The granularity of the temporal dimension (i.e. lengthT sizeBinsT), will depend on both the order of such dimension (i.e. length of t axis) and the amount of discrete portions of time (i.e. size of cells/bins in the temporal dimension), requiring the following amount of memory to store the TSDS:

$$\textbf{total Memory SDS} = \mathbf{X \cdot Y \cdot Z \cdot T \cdot B \cdot N} = \frac{\textbf{lengthX}}{\textbf{sizeBinsX}} \cdot \frac{\textbf{lengthY}}{\textbf{sizeBinsY}} \cdot \frac{\textbf{lengthZ}}{\textbf{sizeBinsZ}} \cdot \frac{\textbf{lengthT}}{\textbf{sizeBinsT}} \cdot \mathbf{B \cdot N} \quad (7)$$

The meaning of parameters in Eq. (7) is the same as in Eq. (4). The combination of the concepts of the RSDS and TSDS is also possible (RTSDS), thus reducing the memory needs to support the TSDS (similar to Eq. (5)):

$$\textbf{total Memory BS} = \mathbf{X \cdot Y \cdot Z \cdot T \cdot P} = \frac{\textbf{lengthX}}{\textbf{sizeBinsX}} \cdot \frac{\textbf{lengthY}}{\textbf{sizeBinsY}} \cdot \frac{\textbf{lengthZ}}{\textbf{sizeBinsZ}} \cdot \frac{\textbf{lengthT}}{\textbf{sizeBinsT}} \cdot \mathbf{P} \quad (8)$$

In the logical structure of the TSDS each 4D coordinate is represented by a single database record/row, sorted sequentially to ease the reading/writing content access. The access method to records of a (3D) SDS is extended to take into account the 4th dimension:

$$\mathbf{SDSpos = x \cdot Y \cdot Z \cdot T + y \cdot Z \cdot T + z \cdot T + t} \quad (9)$$

Thus, inside the TSDS, the SDSpos represents an univocal record/row position that stores the information relative to a given 4D certain coordinate (x,y,z,t), while X, Y, Z and T are the total amount of different discrete values that the constituent variables of such a 4D coordinate can adopt, according to the order/size of each respective dimension.

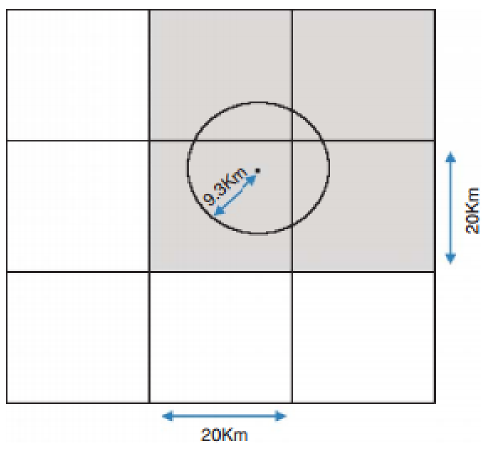


Figure 16: Neighbourhood (shadowed) defined by geometrical arguments

Source: Ruiz, Piera, 2013, [1]

2.3.2.3 RTSDS: Combination of RSDS and TSDS

The combination of the two SDS configuration introduced above, i.e. RTSDS configuration (RSDS + TSDS), was tested in the STREAM project [1] in order to build a strategic Conflict Detection & Resolution tool. The tests have only taken in consideration the route phase and the granularity used is 20 km x 20 km x 600 m, which is in reality the dimension of the constituent bins. The utilization of these dimensions is because these ones are the double of the minimum safety en-route separation defined in the current ATM, i.e. 5NM (~9.3 Km) in the horizontal plane and 1000ft (~300 m) in the vertical one. Since the second is the time unit used to discretize the 4D trajectories of the aircraft, the resolution of the temporal dimension has also been set to 1 second, facilitating the construction and manipulation of the RTSDS [1].

The size of these bins has been established considering a trade-off between the resulting benefits of the RTSDS trajectory pruning (too big bins means a less powerful "trajectory pruning") and the amount of RTSDS accesses required for neighbourhood queries which affects directly to the time consuming (too small bins require searching for neighbours in more bins).

Taking into account the bins dimension in the horizontal plane (20 km x 20 km), we have to consider only 4 bins to be accessed in order to complete the neighbour search at each trajectory time-step (as Fig. 16 shows), while a considerable amount of pairwise comparisons is filtered due to the relatively narrow dimensions of the neighbourhood (40 km x 40 km). In the vertical plane occurs something similar since 2 bins are necessary to ensure the detection of any conflict due to the 600 meters fixed as the bin-size in this dimension. In total, the algorithm checks $4 \times 2 = 8$ adjacent bins, looking for neighbours at each time-step of a given trajectory.

When another aircraft is found inside the neighbourhood formed by these 8 bins (in the same time instant, since TSDS neighbourhoods are time-spatial regions), in order to detect if there is a conflict between them, the algorithm performs a direct distance comparison between their expected point-mass positions. In this case we will use an airspace sector of 5.000 x 5.000 km^2 of surface, i.e. the most part of the European ATM, as an example to show the amount of memory required using this kind of SDS configuration (RTSDS). Therefore, 20 flight levels (6000 meters) and a strategic look ahead of 5 hours with a resolution of 1 second will be considered in the CD process. Then, the memory size occupied by the BS is (consider 4 bytes per row) [1]:

$$\mathbf{X = Y = 5000km/20km = 250} \tag{10}$$

$$\mathbf{Z = 6000m/600m = 10} \tag{11}$$

$$\mathbf{T = 5h \cdot 3600s/h = 18000} \tag{12}$$

$$\mathbf{BS= X{\cdot}Y{\cdot}Z{\cdot}T{\cdot}4 = 250 \cdot 250 \cdot 10 \cdot 18000 \cdot 4 = 45GB} \tag{13}$$

Let consider a maximum of 30.000 different trajectories within the 5 hours look-ahead of the scenario, with average flight duration of 2 hours (and resolution of 1 second). Then, the STI will occupy (consider 8 bytes per row)

$$\mathbf{STI = N \cdot L \cdot 8 = 30000 \cdot 2 \cdot 3600 \cdot 8 = 1.8GB} \tag{14}$$

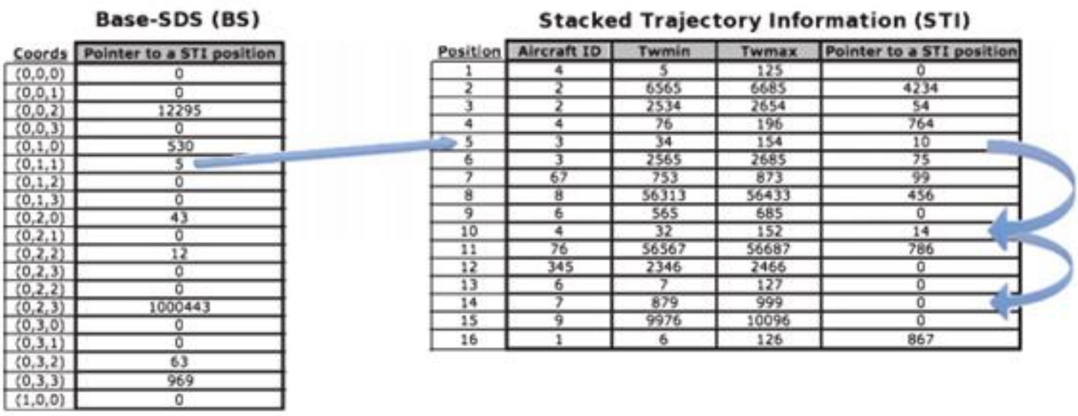
As you can observe, the total memory required using the RTSDS and considering this relatively wide scenario described before would occupy less than 47GB, an amount easily supportable by the current commercial 64-bit computer in RAM main memory (instead of using external hard drivers that are much slower at reading and writing the information). A CR module can take advantage to the capability of the RTSDS configuration since it makes possible the storage of the ATM 4D state-space within 5 hours strategic look ahead. It means that the content of the RTSDS can be updated in order of seconds, a strategic key factor that can be used to dynamically adapt the flight routes allocation in response to network changes, which allows a balance in real time of the actual available airspace capacity and demand.

3. WAKE VORTEX ENCOUNTER PREDICTION SYSTEM

The SDS architectures introduced in the state of art have only been implemented to enable wake vortex encounter predictions in reduced regions like TMAs, but the future ATM necessities require the development of a system capable to consider the traffic of all European airspace. A part from that, it is necessary to introduce information about the origin and behaviour of the wake vortex in time-space to achieve an efficient wake vortex encounter prediction system. These two aspects represent the development of new models and the use of a larger amount of memory, which is currently bounded by technological levels achieved in computer industry.

Therefore, taking into account the improvements achieved by using Relational SDSs, it is logical to consider this system as a starting point.

Analysing the RSDS to find a way to include more information without increasing the necessary amount of memory, we observed the existence of a large set of regions in BS (memory positions) that were actually empty (i.e. 0 reservations like in coordinates (0, 0, 0) and (0, 0, 1)) during the representation of traffic at ECAC level (more than 4000 trajectories).



Base-SDS (BS)

Coords	Pointer to a STI position
(0,0,0)	0
(0,0,1)	0
(0,0,2)	12295
(0,0,3)	0
(0,1,0)	530
(0,1,1)	5
(0,1,2)	0
(0,1,3)	0
(0,2,0)	43
(0,2,1)	0
(0,2,2)	12
(0,2,3)	0
(0,2,2)	0
(0,2,3)	1000443
(0,3,0)	0
(0,3,1)	0
(0,3,2)	63
(0,3,3)	969
(1,0,0)	0

Stacked Trajectory Information (STI)

Position	Aircraft ID	Twmin	Twmax	Pointer to a STI position
1	4	5	125	0
2	2	6565	6685	4234
3	2	2534	2654	54
4	4	76	196	764
5	3	34	154	10
6	3	2565	2685	75
7	67	753	873	99
8	8	56313	56433	456
9	6	565	685	0
10	4	32	152	14
11	76	56567	56687	786
12	345	2346	2466	0
13	6	7	127	0
14	7	879	999	0
15	9	9976	10096	0
16	1	6	126	867

Therefore, the main goal of this section is to propose a method in order to allocate in memory only the regions in which there is a considerable probability that a wake vortex encounter occurs, and only declare these ones in BS. This concept is the base of the system proposed in this project to perform Wake Vortex Encounter Prediction (WVEP) at ECAC-level and with thousands of trajectories. The proposed system architecture is shown in Fig. 17. SDS will be used in two different processes (HRRI and WVE Risk Analysis) with different configurations to achieve the goals of this project.

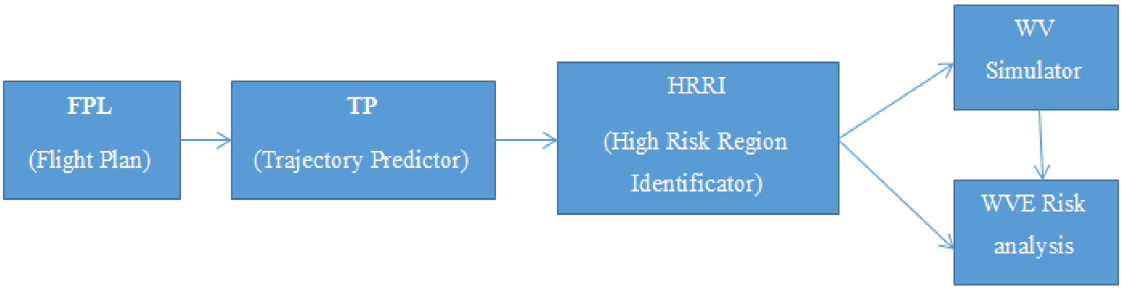


Figure 17: Wake Vortex Encounter Prediction System's Architecture

Firstly, the system will process the Flight Plan (FPL) of all the flights scheduled to provide information about the origin, destination and the route preferences for each flight. These flight plans are used by the Trajectory Predictor (TP), where they will be processed in order to obtain the associated 4D trajectories with their corresponding waypoints. This TP module is assumed to provide optimal trajectories for the airline operator.

Then, the High Risk Region Identificator (HRRI) will be fed with the outputs of the Trajectory Predictor to detect the regions or neighbourhoods where there is a high probability of a severe wake vortex encounter. The identification of these HRRs (High Risk Regions) will be performed with the help of SDSs.

To avoid excessive and unnecessary computational effort, the wake vortex will be simulated by the WV Simulator module only in those regions considered dangerous.

Once the wake vortex is simulated for each of the identified regions, it will be processed by the WVE Risk Analysis module. Here a configuration with SDSs will be also used in order to efficiently and accurately detect potential wake vortex encounters. If a wake vortex encounter is predicted then the associated severity will be calculated, to determine the actual risk of such encounter.

So, to summarize the procedure that the system follows, in the next figure you can observe the associated flowchart of the proposed system, which includes all functions explained above and the way in which they are interconnected.

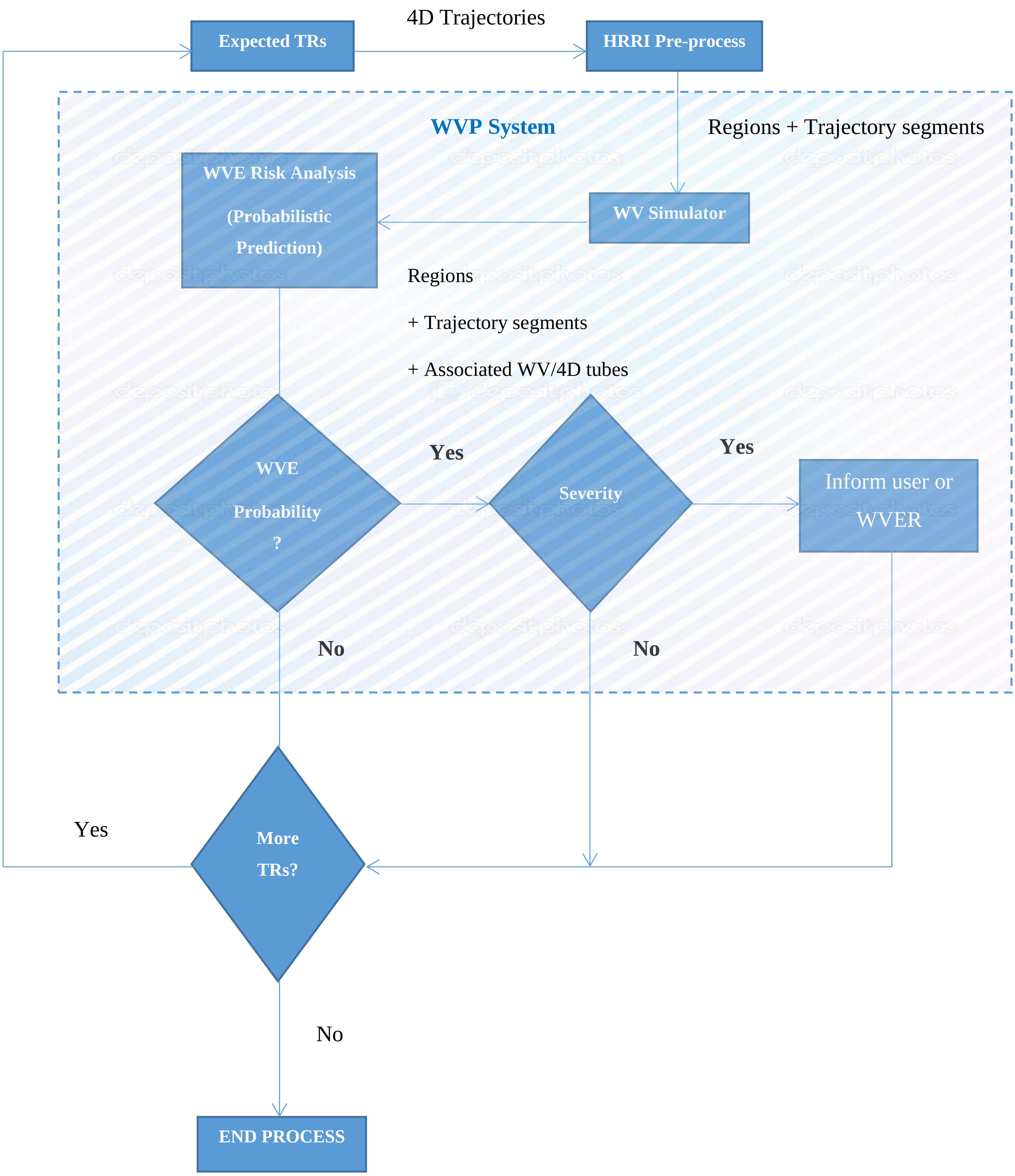


Figure 18: System's Flowchart

The modules that use SDS configurations to perform their functions are those explained in the next sections (3.1 and 3.2), which are the base of the proposed high-level architecture.

3.1 High Risk Region Identificator (HRRI)

To perform this function properly and efficiently, it is necessary to analyze the future ATM situation and to assess if the current safety standards are suitable for the future necessities. Forecasts indicate an increasing traffic density, whereby the prediction of encounters with enough time ahead will be essential.

This module is necessary to filter unnecessary computations when the process to identify the regions of interest is being performed, improving the system’s efficiency and its time-performance.

The current safety standards are 5NM in the horizontal plane and 1000ft (300m) in the vertical one, as Fig. 19 shows. However, these distances are sometimes not enough to protect flights against wake vortex encounters [4]. Therefore, in this project we will use 10 NM and 2000ft as conservative distances which contemplate all the possibilities referring to the behavior of wake vortex including the uncertainties that can be present, thus allowing a prediction with enough time ahead that guarantees the effectiveness of the system in terms of safety.

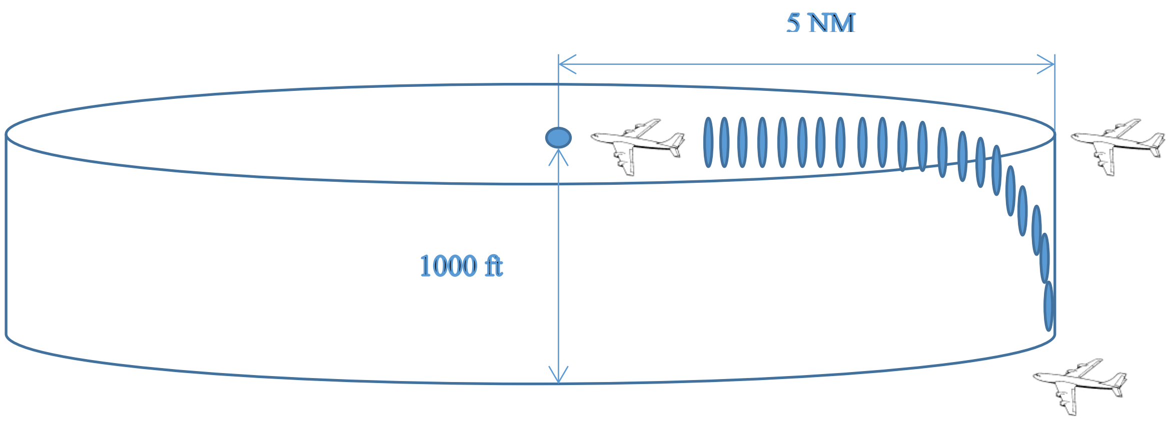


Figure 19: Current Safety Standards

The process to filter unnecessary computations consists in performing a pre-process considering the same SDS configuration (i.e., RSDS and TSDS) than the used for Strategic De-confliction explained in Section 2, whose objective is obtaining a microscale snapshot of all European air traffic. Thus, the SDS's granularity will be 20 NM x 20NM x 2000ft (37km x 37kmx 600m) and we will use the conflict detection function in order to detect the trajectories that are spaced apart less than 10NM horizontally and 2000ft vertically (the SDS' granularity is the double of the distance considered as safety standard).

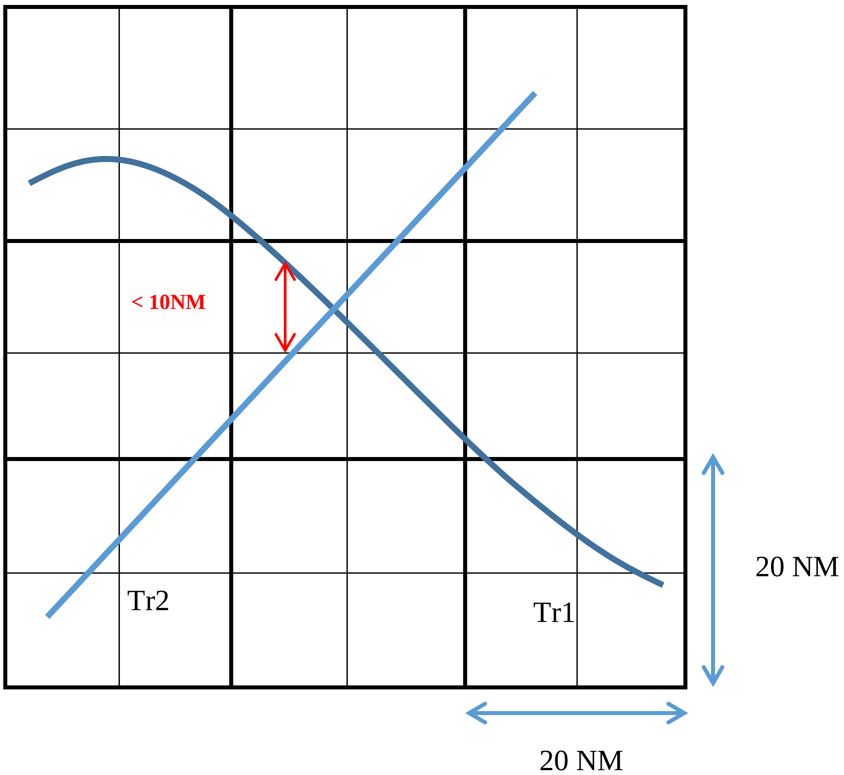


Figure 20: Pre-process scenario

In Fig. 20 you can see how each cell has four quadrants, so the first step in the pre-process will be the identification of the quadrant in which the separation between both trajectories is less than 10NM, i.e. the affected quadrant. The way to obtain this information is applying the next equations given a coordinate (x', y', z'), each one for each dimension:

$$Qx = \frac{x' - x0}{xf - x0}$$

$$Qy = \frac{Y' - y0}{yf - y0}$$

$$Qz = \frac{z' - z0}{zf - z0}$$

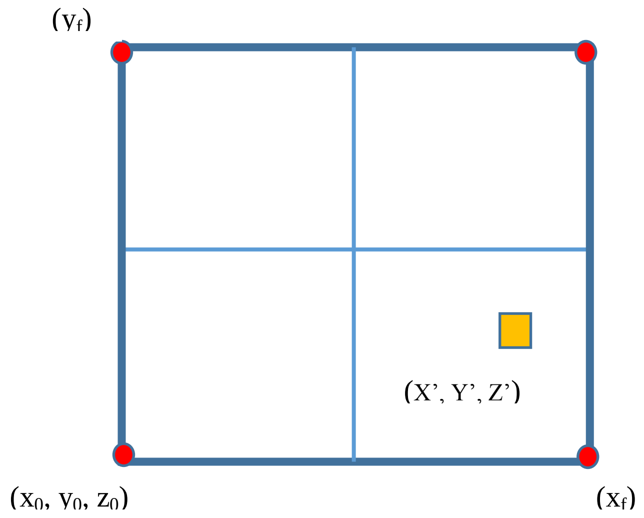


Figure 21: Method to determine the affected quadrant

Substituting the coordinate values in these equations a value between 0 and 1 will be obtained, which will give information about the quadrant of the bin occupied by a certain object (e.g., an aircraft). If the value is less than 0.5, the affected quadrant will be the first that forms the grid on that axis, while if it is equal or greater than 0.5, will be the next one.

Once the affected quadrants have been detected, we have to identify the 8 neighboring quadrants in which the wake vortex will be present (4 horizontally and 2 vertically, forming a cube).

This method will allow the identification of the regions with a significant risk probability that an encounter occurs, and it is similar than the used in collision avoidance applications introduced in section 2.3.1.1. Therefore, the output obtained through this pre-process is the following one:

Pre-process Output			
1	**2**	**Affected Quadrant**	**Neighboring Quadrants**
2	**5**	**Affected Quadrant**	**Neighboring Quadrants**
...	**...**	**...**	**...**

Figure 22: Pre-process Output

The output shows the involved trajectories (first and second columns) and the affected quadrants as well as neighboring ones, which will configure the High Risk Regions (HRR). This output is stored in the main memory together with the scenario (to analyze the changes in the network and the corresponding updated information).

Therefore the TSDS configuration is applied to continuously obtain updated information of the state-space when the pre-process is performed, following the trajectories evolution at each given time step in order to identify the HRRs. Apart from that, the RSDS is also used to create the scenario for the pre-process with all the necessary information about trajectories, and time-windows associated with each one of the coordinates occupied. Thus, the union of these two configurations (TSDS + RSDS) enables the effective and efficient identification of the HRRs, which is the first step to achieve an architecture capable to contemplate all European airspace, focusing the efforts in the regions of interest to reducing the necessary amount of memory and to make it computationally feasible.

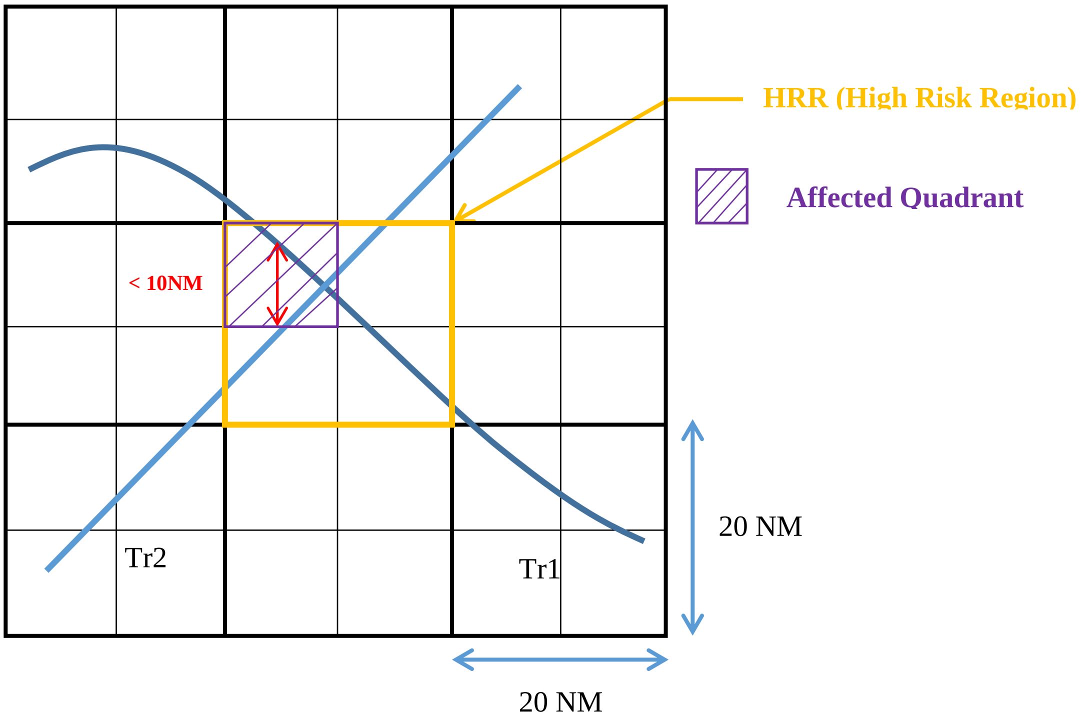


Figure 23: 2D High Risk Region

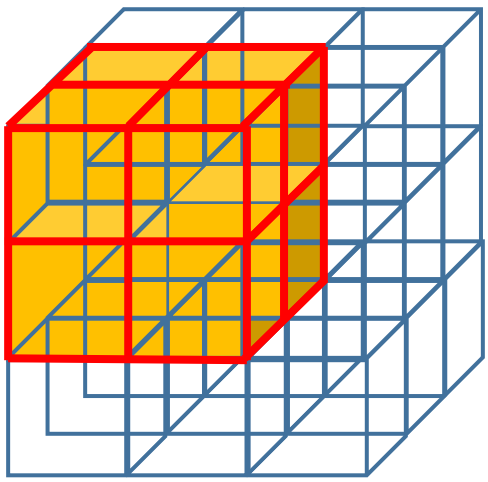


Figure 24: 3D High Risk Region (cube)

Once, a HRR is detected, the corresponding SDS is created to represent the scenario. As a result, we will have N High Risk Regions identified, i.e. N SDS with a granularity of 100 meters to accurately reproduce the dynamics of the wake vortices and the severity of a potential encounter (if this one occurs), a process that will occur in the WVE Risk Analysis module. Therefore, there will be N BSs and N STIs, whose role will be played by Wake Vortex Information (WVI), concept explained below.

3.2 WVE Risk Analysis

The information generated by the HRRI module, i.e., the High Risk Regions identified together with the involved trajectory segments, will be transferred to the WV Simulator. Its function is to generate the wake vortex in volume, using a discretization of a second to achieve a representation with optimum resolution for wake vortex prediction and severity quantification.

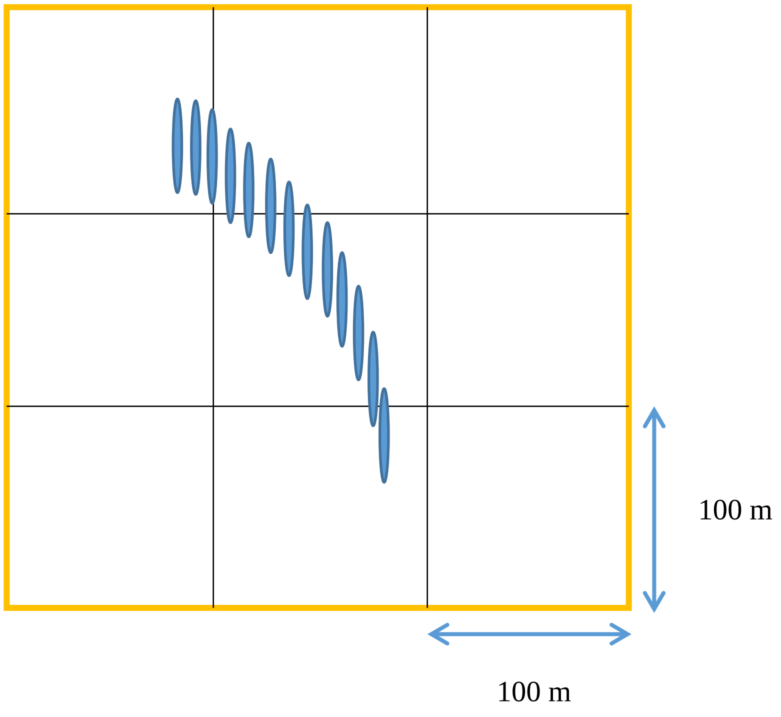


Figure 25: Wake Vortex's Discretization

Based on the same simplified models used in [1] and explained in section 2.3, the wake vortices of each flight will be modeled as a 4D tube used as envelope of the worst-case behavior of the vortex (as Fig. 10 shows) . However, the model will be improved to represent more precisely the actual dynamics of position and strength.

The new wake vortex model can be represented as a sequence of shorter tubes enveloping each one of the velocity vectors that configure the wake vortex. Therefore, for each segment comprised between a pair of coordinates, a tube is built covering completely the corresponding part of the wake vortex, which will be centered inside the tube (this one will be as long as the velocity vector formed by the two coordinates is).

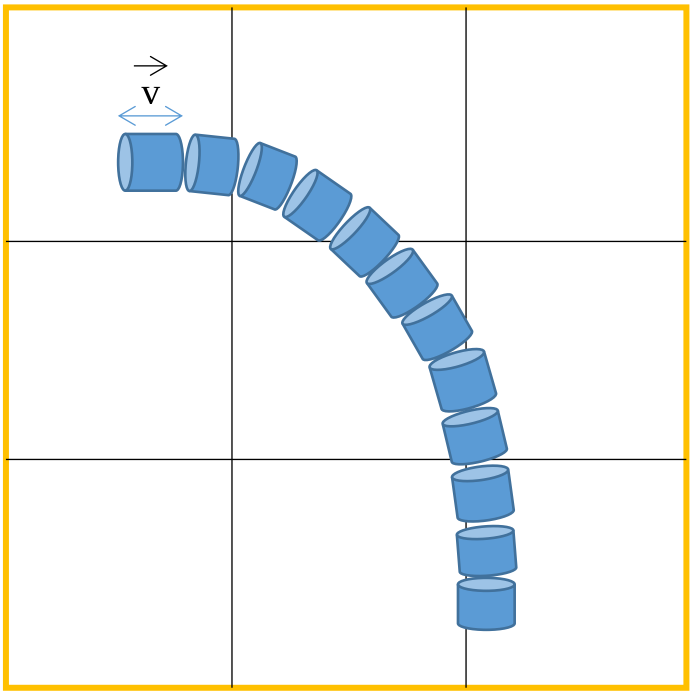


Figure 26: WV's Enveloping Tube

Not to scale

Thus, as Fig. 26 shows, a tube can be represented as a succession of circumferences situated one in front of the other. Therefore, the construction of a discrete tube can be simplified by forming a single circumference with a certain radius, in order to do as many copies as we need to achieve the total length of such the velocity vector. Actually, it constitutes a discrete model of wake vortex, having as a surface, the worst case of a probabilistic envelop. So the radius considered in the discrete model will be determined analyzing the outputs obtained doing simulations.

There are several computer algorithms used to create circumferences. One of the most popular is the Bresenham algorithm, which uses the symmetries and antisymmetries that occur between 8 octants in which a circumference can be divided. Thus, it is only necessary the construction of one of the octants in order to shape the rest of octants by sign changes in its coordinates, obtaining the circumference shown in Fig. 27 [5].

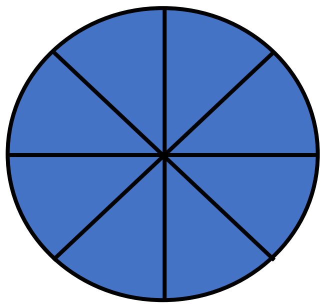


Figure 27: Circumference with symmetrical and antisymmetric segments

Source: Ruiz, 2010, [5]

Fig. 29 shows the discretization of a circumference in a grid of pixels to allow its reproduction on a computer screen. A circumference can be represented by a serie of separate pixels together with a minimum separation between them equal to the chosen for the discretization of the space model (in this case, 100 meters). Thus, each point on the circumference can be thought as a position (coordinate) inside the SDS [5].

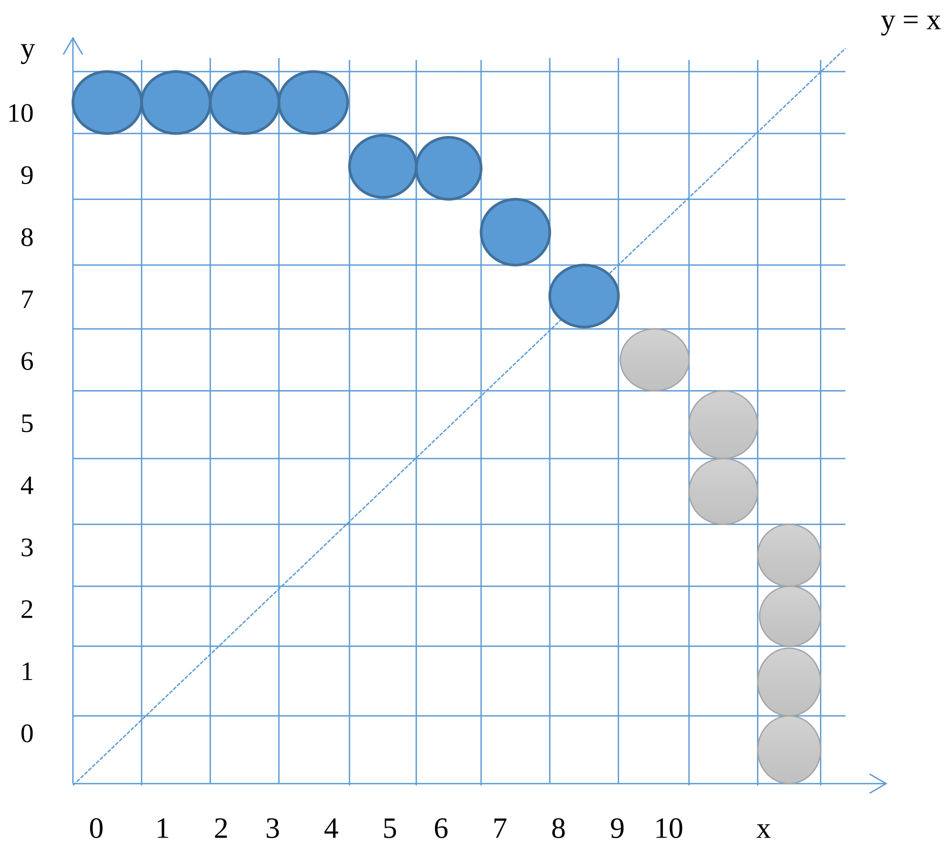


Figure 28: Octant formed by joint pixels

Source: Ruiz, 2010, [5]

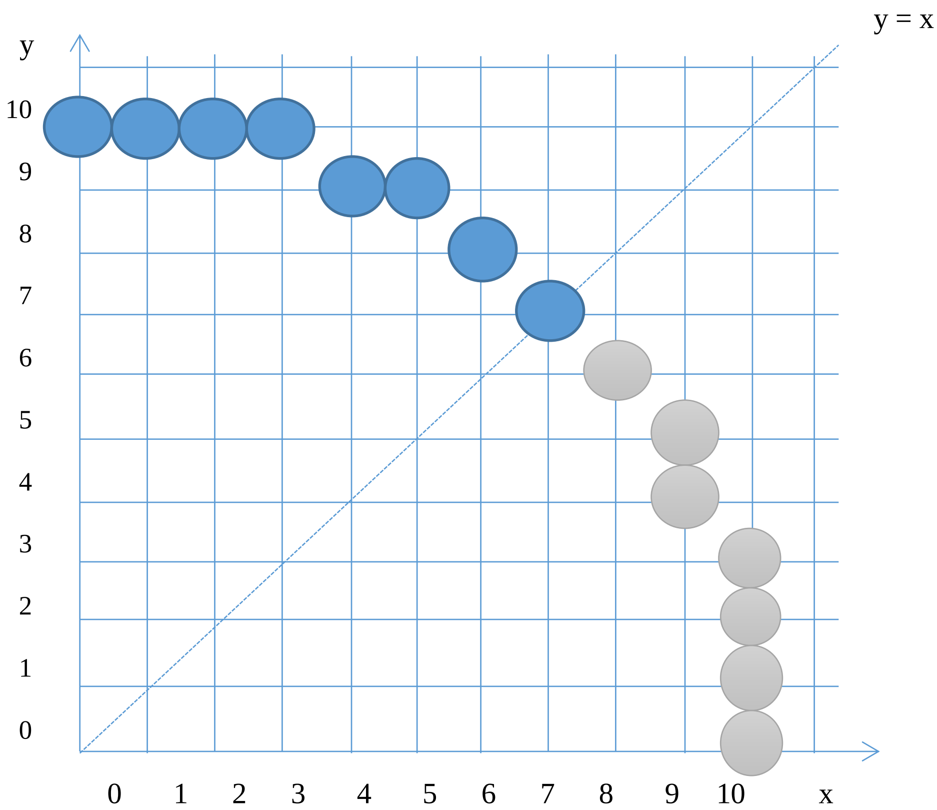


Figure 29: Octant formed by disjoint pixels

Source: Ruiz, 2010, [5]

Once all octants are calculated on the XY plane, a circumference can be obtained as shown in Fig. 30. And as Fig. 31 shows, a cylinder can be constructed centered on the z axis by repeating pre-calculated circumferences separated by a distance of 100 meters to equal or exceed the length of the corresponding speed vector.

This way of constructing a cylinder is very simple to implement in a computer and it has the great advantage that the circumferences are pre-calculated, so it is totally efficient from a computational point of view.

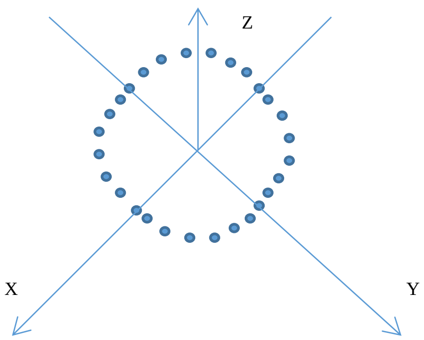


Figure 30: Discrete Circumference

Source: Ruiz, 2010, [5]

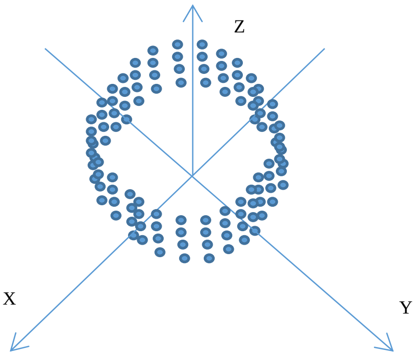
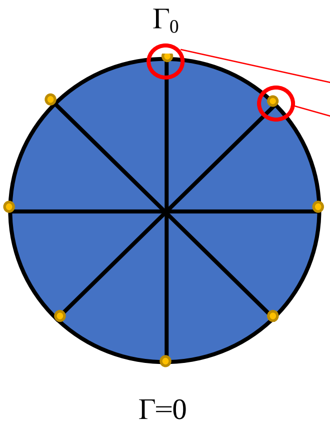


Figure 31: Construction of discrete cylinder

Source: Ruiz, 2010, [5]

These techniques were already used in [5], however in this project a new model enhancement is proposed. We want to achieve a discrete model of wake vortex represented in a single circumference in order to identify at each coordinate, the wake vortex generator aircraft's id, the period of time (Δt) ranging from the instant in which the wake vortex leaves a coordinate (ti) until the instant in which it enters in the next one (tf) and the circulation of the wake vortex associated to each time-space position, thus having a more precise wake vortex dynamic model than in [5] but still being much more computationally efficient than precise fluid dynamic computations, which are not fully required for wake vortex encounter predictions and hazards analyses. This concept is shown in Fig. 32.



Wake Vortex Information (WVI)				
(x, y, z)	**id**	**ti**	**tf**	**Γ0**
(x1, y1, z1)	**id**	**ti1**	**tf1**	**Γ1**
(x2, y2, z2)	**id**	**ti2**	**tf2**	**Γ2**
…	**…**	**…**	**…**	**…**
(xi, yi, zi)	**id**	**tii**	**tfi**	**0**

Δt

Figure 32: Wake Vortex's Discrete Model

Therefore, each coordinate represents a position of the wake vortex at each time instant, considering the first one (centered above) as the position with initial circulation $\Gamma_{0,}$ and the coordinate located at 180 º respect this one, as the position in which the circulation (Γ) is equal to zero. Thus, we can know at each position the severity of the wake vortex to decide if it is dangerous overfly the area or not.

Finally, we have to understand the last function called WVE Risk analysis. This one works with information from HRRI and with the data originated in WV Simulator, i.e. the tubes as well as the severity corresponding to each time-space.

Its objectives are basically to determine the risk of flying over a certain area with a specific wake vortex and detect a potentially affected aircraft and quantify the severity of a potential encounter. In the last case the information is completely known thanks to the WV Simulator function, but to determine the affected aircraft it is necessary a representation of the wake vortex's evolution at micro-scale as well as the trajectories involved in the HRR. Thus the system will display the state of the airspace (HRR) in order to avoid wake vortex encounters that may endanger flight's safety, to inform the Wake Vortex Encounter Resolution (WVER) system or the user, and to take more efficient and safer decisions. An illustration of the information stored in the SDS used at a given HRR can be represented as in Fig. 33. In this case the RSDS configuration is applied again to store all the wake vortex information (stored in WVI) corresponding to the discrete vortex model.

In this way we achieve the storage and management information at a micro level (WVE Risk analysis), through a pre-process performed at a macro level (HRRI) thanks to the suitable combination of these two configurations.

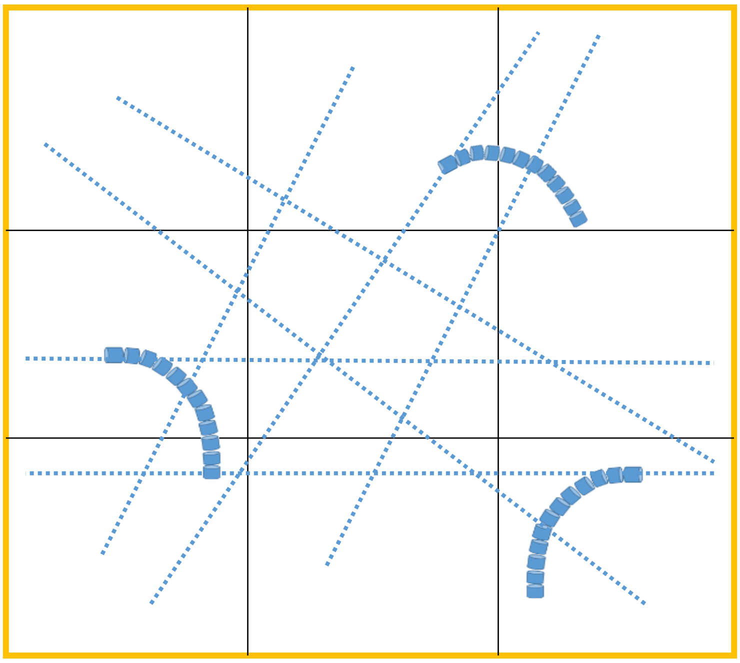


Figure 33: State space of the HRR

Not to scale

4. EXPECTED ALGORITHM PERFORMANCE

In this section it is anticipated the memory performance of the algorithms proposed for the Wake Vortex Encounter Prediction system at European scale. The time performance will be analyzed in future work when the proposed algorithms will be implemented.

It is important to point out, for comparison purposes, that trying to store the relevant micro-level model of wake vortices at European airspace scale with the RTSDS configuration seen in section 2.3.2.3, will lead to untreatable amounts of RAM memory. Considering a big European airspace of 5000Km2 and 20 flight levels (6000m), and applying the predictor system's concept that uses a granularity of 100 meters to have a better resolution of the European air traffic scenario, the memory requirements would be:

$$\mathbf{X = Y = 5000000m/100m = 50.000} \qquad (15)$$

$$\mathbf{Z = 6000m/100m = 60} \qquad (16)$$

$$\mathbf{T = 5h \cdot 3600s/h = 18000} \qquad (17)$$

$$\mathbf{BS = X{\cdot}Y{\cdot}Z{\cdot}T{\cdot}4 = 50.000 \cdot 50.000 \cdot 60 \cdot 18000 \cdot 4 = 10.800\ TB} \qquad (18)$$

$$\mathbf{STI = N \cdot L \cdot 8 = 30000 \cdot 2 \cdot 3600 \cdot 8 = 1.8GB} \qquad (19)$$

The necessary amount of memory to store only the BS database is impossible to support by the current commercial computers, thus the wake vortex encounter predictor system is considerably better in terms of resolution and in computational efficiency, being more effective and safer for the strategic management purposes.

To evaluate the proposed system's efficiency in terms of memory requirements, we have to consider first the memory required for performing the HRRI pre-process. Since the configuration of the SDS is based on RTSDS and similar configuration than the one seen in section 2.3.2.3 (RTSDS: combination of RSDS and TSDS), the maximum amount of memory required for the HRRI module is 45GB approximately. To this quantity we have to add the memory requirements of the WV Risk Analyzer, which is dependent on the number of HRR regions to be explored and the size of these.

The number of 2000 HRR will be considered as the maximum based on the approximation obtained through simulations using conflict detection technology and considering the absence of flight level constraints, the freedom of flight route and 10NM minimum separation to identify the HRR.

With regards the size of the HRRs, a deeper knowledge is necessary to determine with precision which can be the area of risk of a wake vortex generated. For this reason will perform calculations for the worst-case and the most-likely scenarios regarding the memory needs to store the entire state-space of the European airspace at micro-scale.

The worst-case scenario is considered to be a conservative configuration of the system that leads to the identification of larger areas of risk (i.e., HRR) that need to be zoomed in with a microscopic representation of the vortex and trajectories, thus requiring a lot of memory positions for their representation. In this case, the highly conservative HRR regions are considered to be of size 40NM x 40NM x 4000ft.

The most-likely scenario is considered to lead to conservative HRR sizes but much closer to the typical values of current safety distances, i.e., 20NM x 20NM x 2000ft, and therefore expected to be more realistic.

4.1 Most-likely Scenario

First of all we have to calculate how many cells there will be, knowing that the corresponding HRR's granularity is 100 meters.

Horizontally:

20 NM = 37.040 m

37.040m/100 m = 370,4 cells ~ **371 cells**

Vertically:

2000ft = 610 m

610m/100 m = 6,1 cells ~ **7 cells**

Therefore, the total number of cells is obtained multiplying the number of cells in each axis:

Number of cells for each HRR = 371 x 371 x 7 = **963.487 cells**

Then, since each cell occupies 4 bytes, and the expected number of HRR is 2000, the necessary amount of memory to store the BS will be:

$$\textbf{BS = 963.487 x 4 x 2000 = 7, 8 GB} \tag{20}$$

$$\textbf{WVI = TM x \# points x 4 x N} \tag{21}$$

A part from that, we have to calculate the memory occupied by the WVI (Wake Vortex Information) which is substituting the role of the STI in the CD process, using the following equation:

where TM refers to the time of simulation, #points to the number of point positions (i.e. coordinates) considered to configure the WV's discrete model and N to the number of trajectories that we have to model.

So, in this case, we will take into account the worst-case scenario in order to see if the system is capable to support the storage of all these information. Therefore, the number of trajectory's seconds to model for each region will be 3600, while amount of point positions will be 10 and the amount of trajectories 4000, obtaining the next result:

$$\textbf{WVI = 3600 x 10 x 4 x 4000 = 0, 6 GB} \tag{22}$$

However, as it is mentioned previously, this is the worst-case scenario. Thus, the expected one would occupy approximately a 5% of the result (i.e. 29 MB), since it depends on the number of regions, which is obtained through the interaction between trajectories, and the regions' size, which determines how many seconds of the trajectory we have to model for each region, obtaining values of the TM and N considerably lower. This result will be also used in the worst case, since the reasoning is the same.

So that, the total amount of memory required when the expected case is contemplated is:

Total Amount of Memory Required: BS + WVI = 7,8 GB + 0,6 GB = 8,4 GB (23)

4.2 Worst case scenario

The same process is repeated for the worst-case scenario, i.e. (40NM x 40NM x 4000ft) whose objective is to check the system's efficiency even considering very conservative distances.

Horizontally:

40 NM = 74.080 m

74.080 m/100 m = 740,8 cells ~ **741 cells**

Vertically:

4000ft = 1220 m

1220 m/100 m = 12,20 cells ~ **13 cells**

Therefore, the total number of cells is obtained multiplying the number of cells in each axis:

Number of cells for each HRR = 741 x 741 x 13 = **7.138.053 cells**

Then, since each cell occupies 4 bytes, and the expected number of HRR is 2000, the necessary amount of memory to store the BS will be:

BS = 7.138.053 x 4 x 2000 = 57, 2 GB (24)

Moreover, we have to add the amount of memory occupied by the WVI (Wake Vortex Information) as well as it was done in the expected case:

WVI = 3600 x 10 x 4 x 4000 = 0, 6 GB (25)

So that, the total amount of memory required when the worst case is contemplated is:

Total Amount of Memory Required: BS + WVI = 57,2 GB + 57,8 GB = 57,8 GB (26)

As we can observe the amount of memory in both the most-likely and the worst-case scenarios, together with the needs of memory of the HRRI module, is easily supportable by the capacity of current computers, which can accept up to 2TB with quasi-standard configurations, so the main objective has been achieved for both expected case and the worst case in which we are considering large scenarios.

5. CONCLUSIONS

The forecasts indicate a notable growth of the air traffic in the coming years, which implies more airspace congestion and, consequently, the necessity to carry out a more efficient management aimed at achieving an airspace free of conflicts and free of wake vortex encounter hazards.

The potential consequences of wake vortex encounters make it a phenomenon that requires special attention. For this reason, safety distances were defined to reduce the risk of encounter. However, nowadays, and considering the characteristics of current and future ATM, these distances are not enough, since it is a phenomenon whose behaviour and evolution in time and space depends on several factors, among which the meteorological conditions that brings a lot of uncertainty into the forecast, or the actual aircraft mass during the flight.

Thus, it is necessary to have a system able to predict with enough precision when and where a wake vortex encounter may occur, and quantify with accuracy the severity associated. The system architecture proposed in this project could meet with all these requirements with enough efficiency to process thousands of trajectories flying in the European airspace for look-ahead times of several hours.

In this sense, the main contributions of the proposed system have been to use the technology of Spatial Data Structures to identify regions with a high probability of wake vortex encounter, to filter and reduce the regions that are subject to a thorough micro-scale analysis of the evolution of the wake vortex phenomenon. Spatial Data Structures have been also proposed to support the efficient prediction of wake vortex encounters, to quantify their severity and to determine the affected aircraft.

The proposed method is based on the combined use of different pre-existing SDS configurations that have been enhanced and integrated to take full advantage of the strengths and diminish the shortcomings of each configuration. The resulting system shall allow to be implemented in a regular commercial computer to process large set of traffics at European airspace scale in timely manner, a problem that is not trivial due to the large amount of memory and/or time required.

In addition, in this project a better vortex's discretization has been proposed with respect previous research, achieving a more precise representation of the wake vortex dynamics. In previous works the tube built to model vortex and to perform wake vortex encounter detection had 300 meter of radius (1000ft) to take into account the worst-case dynamics of wake vortex, but in this system, with the discretization and model carried out it is possible to know the position of the vortex and strength for each discrete time step with much more precision.

Therefore, with the proposed system and algorithms it shall be possible to predict all the potential of wake vortex encounter and to predict the resulting consequences, thus enhancing the traffic planning processes and ATM decision-making processes with a system that can contribute to the current ATM and also fit perfectly into the SESAR's proposed ATM.

6. BIBLIOGRAPHY

[1] Ruiz, S., Piera, M.A., 2013. Relational time-space data structure to enable strategic de-confliction with a global scope in presence of large number of 4D trajectories. Journal of Aerospace Operations, 2, 53-78.

[2] Ruiz, S., Piera, M.A., del Pozo, I., 2012. A Medium Term Conflict Detection and Resolution system for Terminal Maneuvering Area based on Spatial Data Structures and 4D Trajectories. Transportation Research Part C, 26, 396-417.

[3] Ruiz, S., Piera, M.A., Nosedal, J., Rainieri, A., 2013. Strategic de-confliction in the presence of a large number of 4D trajectories using a causal modeling approach. Transportation Research Part C, 39, 129-147.

[4] DE CLEYN, K., 2011. Forecast based Wake Turbulence Separation. Master of Science Final Thesis Report. Delft University of Technology. EUROCONTROL.

[5] Ruiz, S., 2010. Sistema de Detección y Resolución de Conflictos basado en Estructuras de Datos Espaciales. Universitat Autònoma de Barcelona.

[6] Ruiz, S., 2013. Strategic Trajectory De-confliction to Enable Seamless Aircraft Conflict Management. PhD Thesis. Universitat Autònoma de Barcelona.

[7] EUROCONTROL. Single European Sky, 2015. https://www.eurocontrol.int/dossiers/single-european-sky.

[8] SESAR Joint Undertaking. Background on Single European Sky, 2015. http://www.sesarju.eu/discover-sesar/history/background-ses.

[9] European Comission. Transport Modes. Single European Sky, 2015. http://ec.europa.eu/transport/modes/air/single_european_sky/.

[10] SESAR Joint Undertaking. Discover SESAR. Objectives, 2015. http://www.sesarju.eu/discover-sesar/objectives //.

[11] Single European Sky ATM Research. SESAR in brief, 2008. http://www.euro-cdm.org/library/sesar_in_brief_2008_final_low.pdf.

[12] CDTI. SESAR. El futuro de la Gestión del Tránsito Aéreo en Europa: El Cielo Único Europeo y la JU SESAR. https://www.cdti.es/recursos/doc/Programas/Cooperacion_internacional/P.Marco_I%20D_de_la_UE/Transporte/31200_21112111200894122.pdf.

[13] EUROCONTROL. Wake Vortex, 2015. https://www.eurocontrol.int/articles/wake-vortex.

[14] SKYbrary. En-route Wake Vortex Hazard, 2013. http://skybrary.aero/index.php/En-route_Wake_Vortex_Hazard.

[15] Good Aviation Practice. Civil Aviation Authority of New Zeland. Wake Turbulence, 2008. file:///C:/Users/User/Downloads/Wake_Turbulence.pdf.

[16] FAA. Aircraft Wake Turbulence, 2002. http://www.faa.gov/documentLibrary/media/Advisory_Circular/AC90-23f.pdf.

[17] PilotFriend. Flight Safety. Wake turbulence. http://www.pilotfriend.com/safe/safety/wake_turb.htm.

[18] WETTER & FLIEGEN. DLR project. Wake Vortex Physics and Encounter Consequences. http://elib.dlr.de/76397/1/holz-1.pdf.

[19] Klaus-Uwe Hahn. Institute of Flight Research, German Aerospace Center (DLR). ICAS 2002 CONGRESS. Coping with Wake Vortex. http://wakenet.eu/fileadmin/wakenet2-europe/Wakenet2/Publications/Publications,%20Papers,%20Conferences/ICA0250P.pdf.

[20] Avstop. Section 3. Wake Turbulence. http://avstop.com/ac/aim/chap7/aim0703.html.

[21] SESAR. The SESAR Programme R&I 2020 future developments. http://www.iserd.org.il/_Uploads/dbsAttachedFiles/JU_SESAR_2020_v2.pdf

[22] ATM Seminar Severity Metric, 2015. http://www.atmseminar.org/seminarContent/seminar11/papers/425-De%20Visscher_0126151141-Final-Paper-5-5-15.pdf