UAB
Universitat Autònoma de Barcelona

# Tactical Deconfliction Model for Safe Integration of Small UAS into Civil Airspace

**Marsel Omeri**

# Tactical Deconfliction Model for Safe Integration of Small UAS into Civil Airspace

**Marsel Omeri**

Doctoral Thesis

***supervised by Dr. Romualdo Moreno Ortiz***

*Presented in Partial Fulfillment of the Requirements for the PhD Degree Doctorat en Enginyeria Electrònica i de Telecomunicació*

Universitat Autònoma
de Barcelona

January 2022, Barcelona

*To my family*

# Summary

Unmanned aircraft systems (UAS), also known as drones, are aircraft that are operated remotely or autonomously without a human pilot on board. UAS *conflict management* systems, also known as *deconfliction systems*, are designed to address and resolve conflicts that may arise during the operation of drones. These systems can be used to mitigate risks (e.g. collision risk) and ensure safe and efficient operation of drones in various environments and scenarios.

At tactical level, drone deconfliction systems – referred in this thesis as Tactical Deconfliction (TD) systems – may be used to:

- Monitor and control the movements of individual drones in real-time to avoid collisions and other types of conflicts.
- Facilitate communication and coordination between different drones and other aircraft, as well as between drone operators and air traffic controllers.
- Assess and manage potential risks associated with individual drone operations in real-time.

Despite the ongoing research in UAS traffic management and in particular deconfliction systems, the up-to-date models, concepts and methods are not adequate for all UAS types and operations. In this work we identify two main reasons explaining challenges and current situation. First reason is related to the complexity of the problem: different types of UAS (e.g. small UAS, multicopter), different performance capabilities, airspace constrains and regulation restrictions. The second reason, is related to the approach different stakeholder are following to solve the aforementioned challenges. The present picture depicts a lack of common understanding, with several entities (e.g. research groups, related stakeholders) working in silos without acknowledging the existing confusion and proposing new concepts without considering context and relevance of the systems of interest.

To remedy these issues, we proposed a general framework based on systems engineering principles, to provide a structure for organizing and guiding the design of a TD systems. These frameworks specifies various components or subsystems that make up the system, as well as the relationships and interactions between these components. In addition, the framework is used to provide a separation method and minima adequate for small UAS operations, at low altitude airspace.

**Remark**

This thesis was written with the intention to reach a common viewpoint that drone safe integration requires a holistic, coherent and consistent framework. Such a framework should analyze current systems, identify challenges and ”wrong doings”, and pinpoint future directions.

# Resumen

Los sistemas de aeronaves no tripuladas (UAS), también conocidos como drones, son aeronaves que se operan de forma remota o autónoma sin un piloto humano a bordo. Los sistemas de *gestión de conflictos* de UAS, también conocidos como *sistemas de resolución de conflictos*, están diseñados para abordar y resolver los conflictos que puedan surgir durante la operación de drones. Estos sistemas se pueden utilizar para mitigar los riesgos (por ejemplo, el riesgo de colisión) y garantizar el funcionamiento seguro y eficiente de los drones en diversos entornos y escenarios.

A nivel táctico, los sistemas de eliminación de conflictos con drones, denominados en esta tesis como sistemas de eliminación de conflictos tácticos (TD), pueden usarse para:

- Supervise y controle los movimientos de drones individuales en tiempo real para evitar colisiones y otros tipos de conflictos.
- Facilitar la comunicación y coordinación entre diferentes drones y otras aeronaves, así como entre operadores de drones y controladores aéreos.
- Evalúe y gestione los riesgos potenciales asociados con las operaciones de drones individuales en tiempo real.

A pesar de la investigación en curso en la gestión del tráfico de UAS y, en particular, en los sistemas de eliminación de conflictos, los modelos, conceptos y métodos actualizados no son adecuados para todos los tipos y operaciones de UAS. En este trabajo identificamos dos razones principales que explican los desafíos y la situación actual. La primera razón está relacionada con la complejidad del problema: diferentes tipos de UAS (por ejemplo, UAS pequeños, multicópteros), diferentes capacidades de rendimiento, restricciones del espacio aéreo y restricciones reglamentarias. La segunda razón está relacionada con el enfoque que están siguiendo las diferentes partes interesadas para resolver los desafíos antes mencionados. La imagen actual muestra una falta de entendimiento común, con varias entidades (por ejemplo, grupos de investigación, partes interesadas relacionadas) trabajando en silos sin reconocer la confusión existente y proponiendo nuevos conceptos sin considerar el contexto y la relevancia de los sistemas de interés.

Para remediar estos problemas, propusimos un marco general basado en principios de ingeniería de sistemas, para proporcionar una estructura para

organizar y guiar el diseño de sistemas TD. Estos marcos especifican varios componentes o subsistemas que componen el sistema, así como las relaciones e interacciones entre estos componentes. Además, el marco se utiliza para proporcionar un método de separación y mínimos adecuados para operaciones de UAS pequeños, en espacio aéreo de baja altitud.

**Nota**

Esta tesis se escribió con la intención de llegar a un punto de vista común de que la integración segura de drones requiere un marco holístico, coherente y consistente. Dicho marco debería analizar los sistemas actuales, identificar los desafíos y los ”actos incorrectos”, y señalar las direcciones futuras.

# Resum

Els sistemes d'aeronaus no tripulades (UAS), també coneguts com a drones, són aeronaus que s'operen de manera remota o autònoma sense un pilot humà a bord. Els sistemes de *gestió de conflictes* d'UAS, també coneguts com a *sistemes de resolució de conflictes*, estan dissenyats per abordar i resoldre els conflictes que puguin sorgir durant l'operació de drones. Aquests sistemes es poden utilitzar per mitigar els riscos (per exemple, el risc de col · lisió) i garantir el funcionament segur i eficient dels drones en diversos entorns i escenaris.

A nivell tàctic, els sistemes d'eliminació de conflictes amb drones, anomenats en aquesta tesi com a sistemes d'eliminació de conflictes tàctics (TD), poden utilitzar-se per a:

- Supervisa i controla els moviments de drones individuals en temps real per evitar col · lisions i altres tipus de conflictes.
- Facilitar la comunicació i coordinació entre diferents drones i altres aeronaus, així com entre operadors de drones i controladors aeris.
- Avalueu i gestioneu els riscos potencials associats amb les operacions de drones individuals en temps real.

Tot i la recerca en curs en la gestió del trànsit d'UAS i, en particular, en els sistemes d'eliminació de conflictes, els models, conceptes i mètodes actualitzats no són adequats per a tots els tipus i les operacions d'UAS. En aquest treball identifiquem dues raons principals que expliquen els desafiaments i la situació actual. La primera raó està relacionada amb la complexitat del problema: diferents tipus d'UAS (per exemple, UAS petits, multicòpters), diferents capacitats de rendiment, restriccions de l'espai aeri i restriccions reglamentàries. La segona raó està relacionada amb lenfocament que estan seguint les diferents parts interessades per resoldre els desafiaments abans esmentats. La imatge actual mostra una manca d'enteniment comú, amb diverses entitats (per exemple, grups de recerca, parts interessades relacionades) treballant en sitges sense reconèixer la confusió existent i proposant conceptes nous sense considerar el context i la rellevància dels sistemes d'interès.

Per posar remei a aquests problemes, vam proposar un marc general basat en principis d'enginyeria de sistemes, per proporcionar una estructura per organitzar i guiar el disseny de sistemes TD. Aquests marcs especifiquen diversos

components o subsistemes que componen el sistema, així com les relacions i interaccions entre aquests components. A més, el marc s'utilitza per proporcionar un mètode de separació i mínims adequats per a operacions de petites UAS, en espai aeri de baixa altitud.

Nota

Aquesta tesi es va escriure amb la intenció darribar a un punt de vista comú que la integració segura de drones requereix un marc holístic, coherent i consistent. Aquest marc hauria d'analitzar els sistemes actuals, identificar els desafiaments i els "actes incorrectes" i assenyalar les adreces futures.

# Acknowledgements

I would like to express my sincere gratitude to everyone who has supported and helped me throughout the course of my PhD journey.

First and foremost, I would like to thank my supervisor, Dr. Romualdo Moreno Ortiz, for their guidance, support, and encouragement throughout the process.

I would also like to thank my colleagues of the research group who have provided support and encouragement throughout my research. In particular, I would like to thank Dr. Ralvi Isufaj, internally addressed as Niku, for his concrete feedback in development of several ideas and technical support; Dr. Thimjo Koca, internally addressed as ”the complete guy”, for his support in understanding complex concepts and translating them into simple ones; Dr. Marko Radanovic for being the first one guiding me in the world of aviation; and lastly Dr. Miquel Angel Piera for sharing his experience and continuous assistance to facilitate all my endeavours during PhD ”sentence”.

Finally, I would like to thank my family and loved ones for their unwavering support and encouragement throughout my studies. Their love and encouragement have been a constant source of strength and motivation for me.

I am deeply grateful to everyone who has contributed to my PhD journey, and I could not have completed this work without their help and support.

# Contents

# Publications, Conferences & Summer Schools

### Journal Articles

- Radanovic, M., Omeri, M.,  Piera, M. A. (2019). Test analysis of a scalable UAV conflict management framework. Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering, 233(16), 6076-6088.
- Isufaj, R., Omeri, M.,  Piera, M. A. (2022). Multi-UAV Conflict Resolution with Graph Convolutional Reinforcement Learning. Applied Sciences, 12(2), 610.
- Omeri, M., Isufaj, R.,  Ortiz, R. M. (2022). Quantifying Well Clear for autonomous small UAS. IEEE Access, 10, 68365-68383.
- Isufaj, R., Omeri, M., Piera, M. A., Saez Valls, J.,  Verdonk Gallego, C. E. (2022). From Single Aircraft to Communities: A Neutral Interpretation of Air Traffic Complexity Dynamics. Aerospace, 9(10), 613.

### Conference Articles

- Omeri, M., Radanovic, M., Cruz, E. E. S.,  Ortiz, R. M. (2022). Simulation Processes for Onboard State Estimation in a Small UAV Environment. Simul. Notes Eur., 32(1), 9-14.

### Summer Schools and Study Groups

- Advanced Air Mobility Summer School, September 2022. Naples, Italy.

# Chapter 1

# Introduction

The global market of commercial and civil applications of unmanned aircraft systems (UAS), commonly known as drones[1], is projected to grow significantly, with the European UAS industry expected to exceed 10 billion annually by 2035, and over €15 billion by 2050 [1]. This growing trend, particularly at low altitudes, will be accompanied by increased density and new challenges related to the safety, capacity, and efficiency of air traffic and airspace. To meet these challenges novel initiatives are being developed to improve and extend current air traffic management (ATM) systems and introduce new operational capabilities.

In the United States, the Federal Aviation Administration (FAA), the National Aeronautics and Space Administration (NASA), and other stakeholders are collaborating to develop and implement UAS Traffic Management (UTM) Systems enabling UAS operations, along with other airspace users (AUs) [2,3]. Similar efforts are undertaken in Europe through Eurocontrol and Single European Sky ATM Research Joint Undertaking (SESAR JU) leading efforts to develop an equivalent UTM concept, referred to as U-Space [4,5]. The recommended concepts are systemic, in the sense of comprising multiple elements interacting over large physical scale, with critical procedural and regulative changes to produce new capabilities.

A significant requirement in implementing UTM systems is the assurance of *safe* UAS operations. This is achieved, in part, through *conflict management* (CM) systems[2]. Given the importance, the vast amount of proposed concepts and the absence of standardization, this work seeks to develop a framework, needed to understand and overcome key challenges tailored to the design and implementation of such systems.

---

[1] In this dissertation drone and UAS will be used interchangeably

[2] Defined in section 1.2.2

## 1.1 Safety and Risk

The notion of *safety* refers to the state of being protected from harm or danger. In a broader sense, safety refers to the absence of risks or hazards that could cause injury, loss, or damage. In this work, *risk* refers to the *safety risk* concept, defined according to ICAO Safety Management Manual [6] as "the predicted probability and severity of the consequences or outcomes of a hazard". A *hazard* refers to a potential source (i.e. set of conditions or object) of harm or danger. Formally defined in ICAO Doc 9859 [6] as" a condition or an object with the potential to cause or contribute to an aircraft incident or accident".

In systems safety (i.e. UAS), a hazard is defined as an entity that exists in an intermediate state between a set of potential initiating events and their harmful outcomes [7]. Since it represents an intermediate state, for each hazard present in the operation of the system, several potential risks can follow. Therefore, is important to identify and mitigate potential risks in a systematic manner.

*Severity* on the other hand can be understood as the magnitude of harm. Severity is defined through discrete categories (e.g. hazardous, catastrophic) based on its consequences. For instance, an *accident* is considered a catastrophic hazard.

The hazard of interest in this work is mid-air collision (MAC), which is a type of aviation accident when two or more aircraft collide while in flight. The risk of a mid-air collision is the likelihood that two or more aircraft will collide with each other in the air. MAC has been a major concern of air transportation, and has motivated the development of ATM and collision avoidance systems, to control and prevent these accidents from happening. Similarly, assuring that UAS operations are as safe as possible, aviation authorities and industry stakeholders are working to establish safety standards and procedures and novel systems that are designed to minimize the risk of accidents and incidents occurring.

In addition to MAC, collision risk is linked also to another key concept of ATM, known as *conflict*. Essentially, a conflict refers to a state or situation where a predetermined *separation minima* may be compromised by two or more aircraft [8]. The event when the separation minima is infringed is called a *loss of separation* (LoS).

A core concept on determining a possible LoS lies on the estimation of closest point of approach (CPA) between two aircraft. CPA is characterized by time to CPA and distance to CPA. The time to CPA is referred to as the range tau ($\tau$) and is an approximation of the time, in seconds, to CPA. $\tau$ is defined as the ratio of the slant range with the closing speed between two aircraft. While this definition is quite simple, it leads into some issues, especially when there are encounters with very low and/or very high closure rates [9]. For instance, in a slow overtaking encounter, the intruder can get very close in distance, without any tau range infringement. To remedy these type of problems, a modified version of tau is used, referred to as tau mod ($\tau_{mod}$) and defined as following [10]:

$$\tau_{mod} \equiv -\frac{r^2 - DMOD^2}{r\dot{r}} \tag{1.1}$$

DMOD is a modified distance designed to comply with manned aviation collision avoidance systems; $r$ and $\dot{r}$ corresponds to slant range and closure rate respectively. These metrics are commonly used to define a conflict and to determine alerting level and resolution maneuvers.

To minimize the risk of a mid-air collision involving a UAS, operators are required to follow these regulations and procedures set forth by aviation authorities, which may include requirements for maintaining a certain distance from other aircraft, maintaining a certain altitude, and staying within certain designated areas. UAS operators will make use of UTM services and onboard capabilities to comply and assure the required level of safety.

**Observation**

Conflict, LoS and MAC are states that are linked to a certain level of collision risk describing the encounter of two or more aircraft.

## 1.2 UTM systems

Integration of UAS into civil airspace is tightly coupled with the development and implementation of UTM systems. The design and implementation of the UTM systems is an ongoing process and has attracted a lot of attention in national and international initiatives, industry related stakeholders (e.g., Airbus) and academic research communities. The preliminary proposals are derived based on experience with manned aviation concepts and on the lessons learned from the existing ATM systems.

At the core of the UTM systems are a set of collaborative services that would enable safety, handle demand and capacity, improve efficiency and comply with regulations of air traffic management. The services are provided by specialized systems, refereed to as UAS Service Supplier (USS in UTM concept) and U-Space Service Provider (USP in U-space concept). Furthermore both concepts consider supplemental services that would provide additional information and/or complementary services (e.g. weather information), referred to as Supplemental Data Service Provider (SDSP). While UTM systems are still under development and facing several issues [11], they all share the requirement to provide CM services.

For illustrative purposes, the interactions between USS-UAS-SDSP components are shown in the Figure 1.1. The figure depicts the structural and relationships between each component. The presence of arrows indicates that communication is required and the arrowhead indicates direction of the information flow. Noting CM service, it is worth mentioning that drones may be capable of managing conflicts independent of UTM services, using onboard systems.

The proposed components can be understood as following:

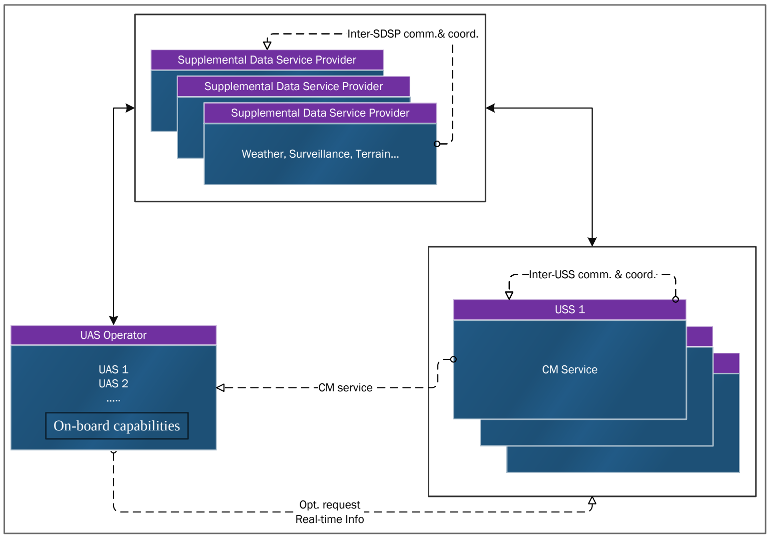


Figure 1.1: Block Diagram of Drone Conflict Management Systems

- *UAS operator* — the goal is to develop such systems that will enable a single operator to control and ideally just monitor several UAS operations at the same time.
- *UAS Service Supplier* — the goal is to develop a set of distributed services, with different capabilities (e.g. assuring safety) through inter-communication and coordination among themselves.
- *Supplemental Data Service Provider* — the idea behind this component, lies with the fact that some services (e.g. weather) can be offered from third party entities. In doing so, there will be a competition between different stakeholders, with the goal to increase fairness and quality of service.

**Observation**

UTM paradigm is based on the idea that the safe and efficient operation of drones in the national airspace system can be achieved through cooperation and coordination among different stakeholders.

### 1.2.1 ATM Conflict Management Systems

Given that UAS are new entrants into an existing air transportation system, it stands to reason that the primary approach for their accommodation and integration is a derivation of existing systems, such as ATM systems in manned aviation. As a result, having a thorough understanding of concepts that we already use in practise is a good practise. For this purpose, we will refer to ICAO documents, under the presumption of general acceptance and serving predominately as a reference guide for the air traffic services providers.

A good starting point for describing CM is ICAO Doc9854/AN458 [8], which provides a global strategy for how ATM systems will provide services and benefits to airspace users through 2025 and beyond. According to the aforementioned operational concept, the purpose of conflict management is to reduce to an acceptable level the risk of aircraft collisions with hazards. In a systemic context, this definition describes the system's mission or objective, which should be understood as *what* the system does – manages the mid-air collision risk. In this context, the *how* can be thought as a process or set of processes applied such that the operation is considered safe – a comprehensive material on what is referred as safe can be further explored here [12, 13].

Conforming to ICAO concept, CM is applied in three layers: *Strategic CM*, *Tactical CM / Separation Provision* and *Collision Avoidance*. Prior to elaborate our observation on the CM notion, let's describe the definitions of the identified layers. The former addresses mainly pre-flight procedures to mitigate conflicts based on the flight plans and aims to reduce the workload for tactical interventions. Strategic CM is achieved through airspace organization management (AOM), demand and capacity balancing (DCB) and traffic synchronization (TS) components of ATM. However, there are cases that strategic actions might be required after take-off, particularly in long-duration flights. The tactical level is responsible for mitigating midterm conflicts through gentle maneuvers in a timely fashion, also known as the separation provision function. In case that separation provision is compromised, CA is activated, which identifies short-term (imminent) intruders and performs last-resort maneuvers to prevent mid-air collisions. For visualization purposes, a simplified model is depicted in Figure 2.1.

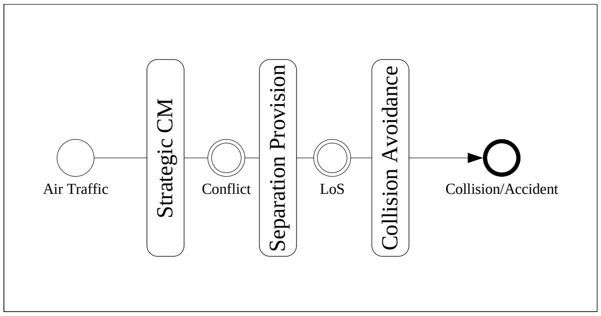


Figure 1.2: Simplified ICAO Conflict Management Model

This concept can be thought as a structured model, comprised of barriers, transition of which would happen if and only if, the previous barrier has failed. The barriers operate from left to right, in different time horizons – known as look ahead time (LAT) – and are characterized by distinct attributes and properties. A representation of CM time-frames and the corresponding functions are illustrated in Figure 2.2.

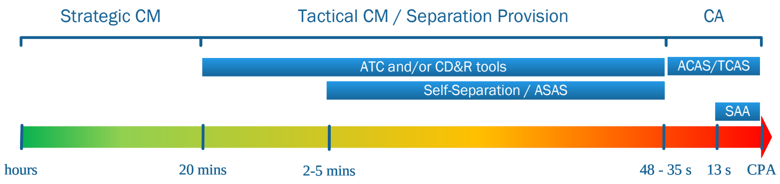


Figure 1.3: ATM Conflict Management time-frames

In manned aviation, tactical conflict management is issued by Air Traffic Control (ATC), a centralized ground-based system that provides guidance and information to the pilots through Air Traffic Control Operators (ATCo). In the event of an emerging collision, Collision Avoidance System (CAS) is enabled seconds before closest point of approach (CPA). Traffic Collision Avoidance System (TCAS) [14]and Airborne Collision Avoidance System (ACAS) are standard CAS systems mandatory for most commercial aircraft, and their main objective is pairwise collision avoidance [15]. The closure rate of aircraft, encounter geometry, and flight level are the primary factors that affect their performance. In addition, the *See and Avoid* principle serves as a CA method, particularly for operations in uncontrolled airspace and general aviation, which might not be equipped with TCAS or similar systems. In such cases, the pilots are fully responsible for assuring safety, under certain flight rules [16].

### 1.2.2 UAS Conflict Management Systems

Some possible areas of focus for UAS conflict management systems could include the following:

- Traffic Management – UAS conflict management systems may be used to monitor and control the movements of drones in order to avoid collisions and other types of conflicts.

- Communication and Coordination – UAS conflict management systems may be used to facilitate communication and coordination between different drones and other aircraft, as well as between drone operators and air traffic controllers.

- Safety Risk Management – UAS conflict management systems may be used to assess and manage potential risks associated with drone operations, such as the risk of collisions, interference with other aircraft, or damage to infrastructure.

- Surveillance and Monitoring – UAS conflict management systems may be used to monitor and track the movements and activities of drones in order to detect and prevent potential conflicts.

At the time of writing this thesis there are no standards of CM systems adequate for UAS operations. Nevertheless, there is a common alignment from different stakeholders with the concepts proposed by ICAO [17], NASA [18] and U-Space [19]. Based on these documents, we derived a framework [20] describing an end to end process, which involves the following stages:

*Stage 1* - Strategic Conflict Mitigation (CM): conflicts are detected and resolved before take-off based on their flight plans submitted to the UTM. This process invokes removing intersecting trajectories on spatio-temporal basis and engaging re-planning to align with various constraints such as no-fly zones (e.g., airports), weather, and other obstacles.

*Stage 2* – Separation Provision Service(SPS): Similar to the ATC functionalities, UTM has to offer in-flight separation as a service if the flight plans approved in Stage 1 are not conflict-free during the flight. The sUAS subscribed to this service [21], gets early awareness (i.e., alarms) for possible loss of separation between other aircraft(manned/unmanned) and guidance for safe and efficient resolutions for planned operations.

*Stage 3* – Self-Separation (SS): Derived from the Free Flight concept [22], relies on the sUAS capabilities to maintain a safe separation minima from other airspace users. This functionality can be carried manually by the remote pilot (RP), assisted, or fully automated. Still, it removes the responsibility of conflict mitigation from the UTM and delegates it to the sUAS.

*Stage 4* – Collision Avoidance (CA): provides a final safety layer to prevent mid-air collisions. It is characterized by imminent and sharp maneuvers (or getting into a hovering state) and can be managed by the remote pilot or autonomously as well [23].

It is worth noticing that systems in Stage 2 and Stage 3 deal with conflicts at tactical level. For simplicity, we will refer to these systems as *Tactical Deconfliction* (TD) systems.

## 1.3 Dissertation Scope

The purpose of this thesis is to provide a coherent and comprehensive framework to facilitate the future development of UAS conflict management systems by employing a systemic approach, that takes into account both the constituent parts and the system as a whole. By taking a systemic approach, this thesis seeks to uncover the complexity of UAS conflict management systems and develop an understanding of their role in assuring safe and efficient operations.

In this work, we focus only on sUAS operating in a UTM system environment. This choice was made given the demands of the application areas (e.g., delivery, inspection) and mission profile characteristics characteristics (e.g., low altitude, short range) [24,25]. Operating in this environment provides the sUAS with greater autonomy, as well as tools and services to safely coexist with other airspace users. In addition, due to the complexity and volume of information regarding CM systems, we will only discuss TD systems that deal with mid-air collision risk.

It is beyond the scope of this thesis to address UTM systems and capabilities other than TD systems (e.g., airspace management, demand, and capacity balancing). Nevertheless, the reader will find the necessary contextual information and reference materials for topics such as communication systems, surveillance systems and information systems, which we deem to be worthy of separate work.

## 1.4 Dissertation Motivation

During a long and systematic process, it has been observed that the literature on integration of UAS into civil airspace is full of academic papers, project deliverables and studies describing frameworks, technologies, methods, CM systems and related topics. Despite ongoing research, the up-to-date TD systems, separation methods and criteria (i.e. separation minima) are not adequate for operations of sUAS at low altitude airspace [26–29].

A good indicator that the literature is flooding with research papers regarding CM, CA and related topics, may be considered the large number of the review articles. For instance, a still cited article is an extensive review of Conflict Detection and Resolution (CD&R) methods, published in 2000 by Kuchar and Yang [30]. Given the high level of interest and the introduction of UAS into the airspace, more recent reviews, such as [31] and [32], focusing on the applications of UAS, were introduced. In addition, a number of papers have been published on UAS CM systems, including a recent systematic review on CA approaches by Tang et al. [33], a review on CA approaches and systems [34], a review on CA systems only [35], a review on Sense and Avoid technologies and applications [36], and a combination of separation management and CA approaches [37].

Given the amount of research done, one can only conclude that it is straightforward to pick up concepts and answer questions such as: ”What is a conflict?”; ”See and Avoid is exactly this...”; ”CA systems are a mandatory requirement because...”; ”Why CA systems are not considered as part of risk assessment methodologies?”; and so on (different related disciplines might have different specific questions). It is our understanding that not only is this not the case, but that the community is proposing, publishing, and building new ”things” on top of this unacknowledged confusion.

As a result, this thesis began with ”answers” (a prerequisite for developing a TD model) to primitive ”questions” that are necessary to adequately define the *problem domain*. Only then can someone come up with a solution to the problem, build new systems (like CM systems), test new methods (like AI-based CDR algorithms), and make it possible for safe UAS integration into civil airspace. Note that, *problem* and *solution domain* are technical terms used in sytems engineering processes [38].

We have evidenced two possible reasons why we are in the current situation. The first reason, which is widely accepted, is mainly related to the complexity of the problem itself: the heterogeneity of small UAS types (i.e., multi-rotor, fixed-wing), their different performance capabilities (i.e., size, maximum takeoff weight, maximum airspeed), airspace structure, and unreliability in Communication, Navigation, and Surveillance (CNS) systems [39, 40]. The second reason, which is the driving force behind this work, relates to "how" various stakeholders define problems and find solutions. In our opinion, there is a lack of shared knowledge and understanding, the presence of out-of-date terminology, disjointed concepts, a lack of context and relevance, and a general lack of clarity in defining the problems and their solutions from a holistic perspective. Both academia and industry have identified these problems, but they have not been adequately addressed.

In this thesis we categorize the aforementioned issues into three key arguments:

- Common and shared understanding — refers to the ability of different stakeholders in a project, such as engineers, managers, and customers, to have a clear and consistent understanding of the system's goals, requirements, and design. This means that all stakeholders are aware of the system's purpose, capabilities, and limitations and that they have a shared understanding of how the system will be used and how it will perform. This is related to the aspect of using isolated, local terminologies that are not well accepted and can take different meaning when people out of that organization access the information. For instance, should see and avoid and/or sense and avoid perform self-separation function?

- Context and relevance — Context refers to the environment in which the system will be operating, the stakeholders who will be using the system, and the goals and objectives of the system. It is related to the aspect of introducing new methods and systems that can perform very well on their own but lack interoperability and usability. whereas relevance refers to the degree to which something is related or applicable to a particular situation or context. For instance, the idea that TD systems have been traditionally seen as *reactive*.

- Comprehension and coherence — this aspect is related to the way how the information is provided. In our experience, it is very difficult to get involved and work in the domain of air traffic management. It is common that in literature, state of the art and background information, is a procedural repetition which in the best scenario is a rigid mechanical form of conveying information, and in the worst case is just wrong. The absence of holistic systems understanding and analysis, results into contradicting or unexplained information. We will bring a series of cases to familiarize the reader, with the attempt that this manuscript will remedy some of the issues.

It is worth mentioning that the use of the word *system* is present in many related works, but the content is typically tailored to only one aspect of the system (e.g. algorithms and methods aspect). It is in our view that integration of drones into civil airspace should lean more into a *systemic* context, which requires a hard paradigm shift towards *systems engineering* (SE). In doing so, we can effectively monitor all the characteristics from the design to the implementation of systems of interest, through well defined methods and processes. This approach brings definitonal rigor and common terminology, which increases the chances for practical implementation and the engagement of other research communities [23, 41]. Additionally, it would create common (or similar) frameworks to compare different systems alternatives, which is an important factor not only for TD systems, but for the design and implementation of the whole UTM systems.

## 1.5 Dissertation Objective

The main objective of this work is to identify the necessary structure elements for tactical deconfliction systems and defining their functional behavior. In doing so, we intent to give a holistic, comprehensive and coherent framework describing decisions, methods and metrics adequate for safe integration of sUAS. The intent is to progress state of the art and to ease the work for other researchers (and involving stakeholders) and prevent further miss-directions. More specifically, the objectives of this dissertation are the following:

- Identifying main reasons why there is a lack of standard TD systems adequate for sUAS operations.
- Developing a framework based on systems engineering principles to tackle the identified challenges (i.e., issues).
- Give a formal definition of current TD Systems, through modeling constituent elements/subsystems and their corresponding functions/capabilities.
- Systemic analysis of current TD systems – understanding structure and behavior (e.g. reactive and emergent behavior).
- Proposal of a feasible TD model for sUAS operating under UTM systems
- Proposal of self-separation minima for sUAS operations

## 1.6 Research Approach

Given the scope and the objectives of our work, we decided to review a vast amount of related materials that include not only research articles but studies, related project deliveries, and technical documents. The authors believe that a scoping methodology is adequate for the purpose of the manuscript. A scoping methodology is defined and used as “to map rapidly the key concepts underpinning a research area and the main sources and types of evidence available, and can be undertaken as standalone projects in their own right, especially where an area is complex or has not been reviewed comprehensively before” [42]. This approach fits very well with the complex situation of CM systems, given the lack of definitions, multitude of terminologies and the lack of meaningful mapping among all these components. In doing so we intent to identify potential research gaps and future research needs, by following a systemic analysis to determine structure and functional behavior of a possible CM system. From this analysis was possible to synthesize the outcomes into a general framework, which was used to conduct further studies on proposing a conceptual TD model and adequate separation minima.

The review process comprises four stages, similar to [43], as illustrated in Figure 2.3.

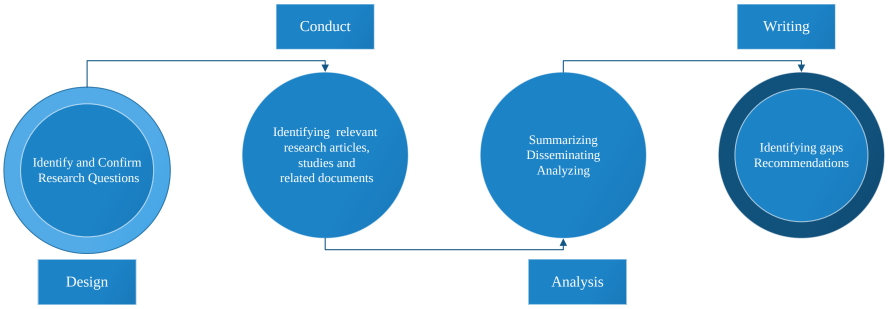


Figure 1.4: Scoping Review Process

### 1.6.1 Design

The first step in designing the scope review was identifying and defining the search terms and concepts of interest. Subject to our research topic and motivation, the selected terms and concepts are showed in Figure 2.2. At this point we do not attempt to map meanings or give any definition, but we just display a collection of ”key concepts” that are typically utilized in the literature. In total we filtered out a total of 28 terms and concepts, which bare the responsibility to keep drones safe while they are performing an in-flight operation.

Figure 1.5: Terms and key concepts related to Tactical Deconfliction

The second step was a broad understanding on how these notions fit together in the context of tactical deconfliction systems. We attempted to evaluate the coherence, relevance and the importance they have, and used these attributes as an inclusion criteria for the conducting phase of the review. This step was important to identify the gaps and formalize the research questions that this work is attempting to answer.

### 1.6.2 Conducting Research

In this phase, we narrowed down the sources used, based on number of citations and their impact that have played in leading the design and development of tactical conflict management systems. After careful consideration, we decided to focus on four main sources:

- Published research (i.e. journals and conference papers) that explicitly focus on tactical conflict management systems or related subsystems (e.g. collision avoidance system)
- NASA UTM technical documents and publications [44]
- SESAR U-SPACE technical documents and publications [45]
- Published research and technical documents that focus on specific and reference Detect and Avoid systems: DAIDALUS [46] and ACAS sXu [47].

The selected literature was read in details and when possible was complemented by following corresponding seminars, events and presentations. For example, Deconfliction and Separation Management—ICAO Drone Enable Symposium [48]. In doing so we attempted to get a deep understanding on the topics and prepare for an objective analysis in the next phase.

### 1.6.3 Analysis

The analysis phase was done through two steps:

- Summarise the state of the art and attempt to identify any redundant concept (e.g. Tactical Conflict Management and Tactical Deconfliction)
- Define each abstract concept as a component of a tactical deconfliction system. In doing so, we had to extract the properties and attributes per each component, characterize the component functions and provide a contextual structure of the TD system and how all the components interact with other.

### 1.6.4 Synthesizing Framework

This thesis is written in the context of systems engineering (SE), and all the terminology and definitions were synthesized from well accepted academic books [49, 50], technical handbooks [51–53] and related articles [54–56].

Figure 1.6 depicts the tasks comprising the proposed framework. Each component depicted below will be explained in details in the second chapter 2.

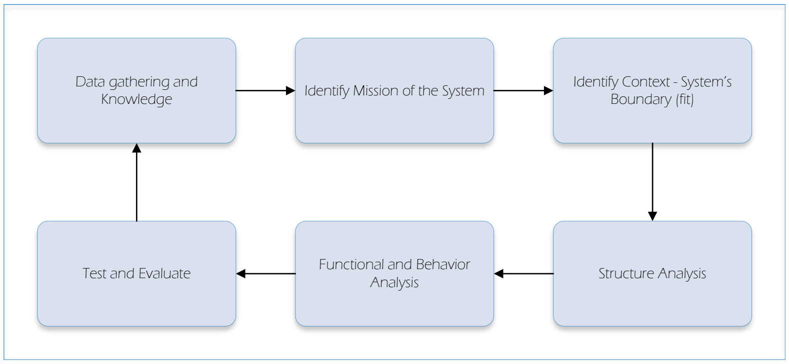


Figure 1.6: Framework for developing TD model for sUAS operations

## 1.7 Dissertation Outline

The rest of this work is organized as follows: Chapter 2 will attempt to equip the reader with the necessary concepts and definitions required to understand the upcoming analysis. First we discuss about systems and notions of systems thinking. Then we focus on identifying high-level concepts of TD systems based on state of the art and analyze their definitions and objectives. Chapter 3 provides the state of the art on TD systems, giving an overview on methods, separation minima and metrics. In addition, a summary of dependent parameters and current limitations are given. After that, in Chapter 4 and Chapter 5 we present the main contribution of this thesis, which is the proposal of TD model and adequate separation minima for sUAS operations. Once we have a feasible model and the required parameters, in Chapter 6 we show a use-case how our work can be applied in practice, in the form of discussion and recommendations. Finally, in Chapter 7 we summarize the contributions of this thesis and give suggestions for future work.

# Chapter 2

# Key Concepts and Background

In this chapter we will attempt to provide the reader the necessary concepts and definitions required to understand the upcoming analysis. In addition, it does not include all concepts related to systems and SE, but we have made an effort to capture key concepts to build a framework with a similar intent as in [57] - ”building on this framework will not be as difficult as construction of the framework itself has been ”. Note that, different from [57], we present up-to-date and practical-oriented concepts tailored to ATM/UTM systems. After we introduce the reader with this line of thought, we discuss high-level concepts of (focused on) TD systems and analyze their definitions and objectives.

## 2.1 Systems Engineering

Systems engineering (SE) is an interdisciplinary field of engineering that focuses on the design, development, and operation of complex systems. It is a holistic approach to engineering that takes into account the entire life cycle of a system, from the early stages of planning and design, through development and integration, to operation and maintenance [52, 58]. It provides the methodologies to design, develop, operate and maintain complex systems.

It is worth mentioning that it is out the scope to discuss the different categories and approaches that have evolved with systems design and implementation, from classical systems engineering to novel approaches such as Model Based Systems Engineering (MBSE) [58]. We do not attempt to compare frameworks, systems architectures and methodologies. While there are different characterization of systems, different categories and specific architectures and life cycle procedures, the fundamental concepts remain the same. It is also true that different working frameworks have common concepts at the core of their principles. Therefore, systems engineering can serve as a guidance for design and implementation of the UTM systems and subsystems (i.e. TD). It also means, that the methodical approach is clear, holistic, identifies requirements, finds solutions, and provides a shared knowledge and common understanding among all the actors and stakeholders.

In context of SE, the definitions of *systems behavior* (see subsection 2.1.3) and *systems architecture* (see subsection 2.1.2) are key components in designing processes. This involves developing a detailed plan for the system's objectives, including the selection of components and interfaces, and the definition of the relationships between those components.

### 2.1.1 What is a system?

Systems are a way of thinking about everything and in almost every domain, therefore there are many definitions, subject to the expertise and the specific domain. The literature is rich in attempts to end the discussion for the "best" definition and conceptualization of the *system* and related notions (e.g. properties, attributes, states, modes, etc.). Nevertheless, for simplicity, we will attempt to put all the concepts and terminologies in the context of UTM and CM — without loss of information and without critiquing other definitions of the literature.

In this thesis we adopt this definition [50], since we find it adequate and more informative to characterize UTM and CM systems:

> The *system* is considered an integrated set of interoperable *elements* or *entities* with specified and bounded *capabilities*, organized in various configurations that enable specific behavior to *emerge* for *command and control* (C2) by user, in order to achieve performance-based mission outcomes, in a prescribed operating environment with a chance of success.

The emphasized terms are deemed important by the authors to capture the essence of a system and are defined as follows:

- *entity* and *element*—are general terms referring to products (e.g. hardware, software, firmware, middle-ware), processes, stakeholders, information, methods, services and other composing components [50, 51].

- *capability*—can be understood as an inherent feature of the system, that is characterized by a function (i.e. action) and a level of performance. Note that a function refers to an action to be performed by the system as in—UAS communicates—but is unitless and does not express a level of performance [50].

- *command and control* — can be thought as a closed-loop process that: (1) continuously monitors systems performance; (2) perform situational assessment for decision making purposes; and (3) issues commands to achieve the required performance [50]. We do prefer using the notion of C2 a system, instead of "utilizing" or "using", since it emphasizes the notion of control — which we will use to describe systems behavior — and the paradigm that everything starts and ends with the user.

- *emergence*— can be understood as a phenomena deriving from the configuration and integration of elements (i.e. components), such that their interactions result in properties, capabilities and behaviours of the whole (i.e. system) that may not be attributed to either of the constituent elements or components [50, 54].

It is noteworthy, this chapter does not discuss and provide a comprehensive understanding of emergence as a concept. We do not argue to what degree it can be managed or designed, but we acknowledge its existence and consider it as an important factor for systems thinking. Nonetheless, we do recommend the reader to pursue a comprehensive understanding of the notion, since it directly effects the essence of a system and may lead into different and (hopefully) better viewpoints. The authors align the concept of emergence with what is referred as *weak emergence* [59], which emphasizes the significance of the context — a CA system may behave differently with respect to different CM models; and modeling and simulation (M&S) as means to understand emergence — the need to know exactly what you are modeling (i.e. what is a conflict?).

Furthermore, the notion of a system would not be complete without considering its hierarchical nature, which is an essential part of the modelling aim to capture this hierarchy and ensure that all the constituent elements adhere to the captured hierarchy [60]. If we consider a two-level hierarchy of a system, the lower level is referred to as *subsystem.* Typically, subsystems interact together to emerge a behavior for a higher-level system. For instance, a TD system (i.e. higher-level) may be comprised of several subsystems such as Surveillance and Conflict Detection and Resolution (CD&R) subsystems. The relationships between system, subsystem and components (i.e. element) are relative, meaning that a system in one level of hierarchy can be the subsystem or the component in other levels.

In addition, throughout this manuscript, we will address systems and system's instances — which can be *products* or *services.* Products are typically employed as *tools* by higher-level systems. For instance, Medium Term Conflict Detection (MTCD) is a tool employed by ATC system. Service will be considered an activity provided and performed by a system, as such it produces a specific outcome that benefits its user (e.g. TD service). For simplicity, we will use the general term system, except of the cases when the terms tool and service add more clarity and meaning.

Now that we "know" what a system is, some concepts related to *systems analysis* (SA) are in order. One may think of SA as a process that assesses solutions – systems objective(s) – to some defined problem (e.g. mid-air collision risk). In order to define the objective(s) of a system, one should identify and study parts/subsystems of the system, its relationships and the overall behavior. Given the fact that most of the engineered systems do not exist standalone but interacting with some other, one should draw a border of interest, and focus in a particular part of the whole. This particular part, is known as *system of* *interest* (SoI). Formally, a SoI refers to the system under the boundary, scoped for contextual analysis and research studies. The essence here lies with the fact to determine what is and/or is not part of *our* system. This is typically depicted via a *context diagram* as shown in Figure 2.1.

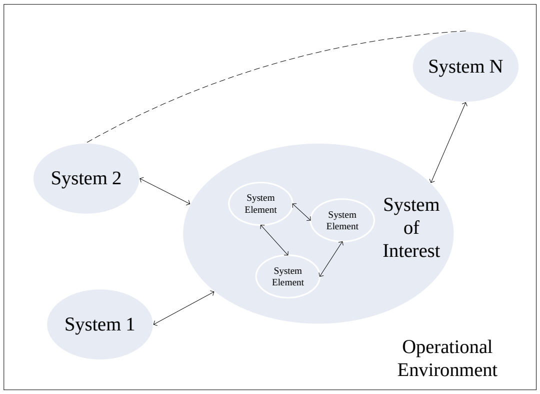


Figure 2.1: System's Context Diagram

In more concrete terms, consider the TD system as SoI, which is part of a higher order system, the ATM/UTM system. In order to differentiate the TD system from another system (e.g. SCM system), one should be able to describe it as a whole and its constituent parts. Analytically, a SoI can be thought as a whole of parts that are characterized by *attributes* and *properties*. Note that, these particular properties and attributes, emerge into different properties and attributes while configured and interacting together (i.e. the system). The term *attribute* refer to the features of the system — which can be observable and quantified — or to subjective qualitative traits. For instance, the mission or objective of a system is considered as attribute [50]. The term *property* refers to unique observable and measurable features that are related to the physical state or logical state of the system. The properties describing a physical state such as size, shape and mass, are known as *physical properties*. Whereas, the properties describing a logical state — a situation (condition and location) at a point in time of the system, or its components (i.e. parts) [49] — are known as *emergent properties* [50]. The state of the system may change over time only in certain ways, due to a significant occurrence, referred to as *event*. A set of connected state changes over time, comprise the systems *behavior*. Thus, one might argue that emergent properties attempt to describe the systems emergent behavior [50].

### 2.1.2 Systems Architecture

In systems engineering, systems architecture refers to the high-level structure of a system (i.e. form), including the components, interfaces, and relationships between those components [49, 61, 62]. It is an important concept because it provides a holistic view and helps to ensure that the system is designed and built in a consistent and logical way (i.e. fit).

From a systems thinking point of view, the design process has typically three main architectures: conceptual, logical (also referred to as functional) and physical. Conceptual architecture refers to the translation of needs of different stakeholders into requirements. This level is commonly regarded as Concept of Operations (ConOps), and represents the problem from the stakeholders and users point of view, independent from technological aspects [51, 52]. Moreover, it defines clearly the capabilities that the SoI should perform and how the stakeholders (also customers) interact and use the system.

The logical architecture of a system interprets models and views of the functionality and behavior of the system [52, 63]. Typically it includes a functional architecture and behavioral architecture. The former refers to the set of functions and sub-functions that defines the transformation of input flows to output flows, in order to achieve the objective. The latter refers to the organization of functions and interfaces that defines the execution sequencing, control and data flow constrains in order to satisfy the performance requirements [63].

The physical architecture of a system orchestrates the physical elements, interfaces and physical links (i.e. radio link) that provides the design solution for a product, service or enterprise, and is intended to satisfy logical architectural elements and system requirements [64]. At this level, the technological aspect plays an important role.

### 2.1.3 Systems behavior

There is a variety of systems behavior that can be analyzed, depending on the domain of the expertise and the type of the system. For example, some general type of systems behavior include: *dynamic behavior*, *emergent behavior*, *adaptive behavior* and *control behavior*. Overall, studying systems behavior can help us to better comprehend complex systems and can give insights into how these systems can be managed and improved. At the same time, understanding systems behavior can help to make better decisions along the process of developing and integrating new systems (e.g. CM systems).

It is common and straight forward to describe systems behavior in terms of states, events and conditions, such that the latter two cause transition of the former [57, 65]. Nevertheless, a simpler and practical approach is to give some insights from *control behavior* perspective. In this context, control — also referred to as decision making or computational architecture — is the process of taking information (e.g. stimuli, cue, event, etc. ) about the environment through the sensors, processing it in order to make decisions and executing the decision [66]. For better comprehension on this concept, Figure 2.2 depicts a model similar to [67], which serves as a baseline of our analysis. This approach is akin to Boyd's OODA loop [68] and RCS architecture [69] which have been a reference on developing C2 and intelligent systems [70–72].

Referring to the Figure 2.2, one can identify the fundamental control components, the definitions of which are as following:

*Sensing* – refers to the ability of the system to receive input (e.g. stimuli, cue) from the environment. For instance, a radar is a sensing equipment, therefore it has the ability to receive electromagnetic waves as input.

*Perception* – refers to the sensory processing mechanism (e.g. filtering, computation, interpreting) such that the derivation of useful information from the environment can be achieved. Note that perception can use different mode of sensory inputs for better performance. Following the previous example, a radar in *sensing* phase can receive radio waves reflected by an aircraft, and in the *perception* phase can provide useful information such as its position.

*Plan* – refers to a set of tasks and sub-tasks object to specific goals, that enables meaningful sequencing of task execution such that drive the system from initial state to the goal state.

*Decision* – refers to all cognitive processes that affect which possible action or set of actions need to be executed.

*Action* – refers to goal-oriented behavior of the system. It can be thought as the ability that systems have to produce the indented outcomes in their environment

Certain aspects of systems behavior can be determined by examining the sequence of interactions between control components. The simplest form of control behavior is *reactive* behavior and comprises a tight coupling of *sense - act* components. These systems are referred as *reactive systems* and are designed to respond in timely manner – often in real-time – therefore the use of explicit abstract representation of knowledge (i.e useful information) is avoided [73]. In this thesis, we will refer to the systems with this specific control behavior as L1 systems.

The L2 systems depicted in the Figure 2.2, involve the interaction of *sense-perceive-decide-act* components and describe a higher level of composure and behavior. These systems are characterized by a degree of variability in their actions, thus requiring *perception* and *decision* elements. L2 systems are typical example of *adaptive behavior*, which adjust their behavior based on the performance of the system or the environment in which they are operation.

When all components are engaged - sensing, perceiving, planning, deciding, and acting - this is known as *deliberative control* and is seen as a higher level layer, necessitating the incorporation of a *planning* element. The authors in [66], characterize deliberative control as "Think, then Act", which can serve as a good analogy for differentiating between other control behaviors.

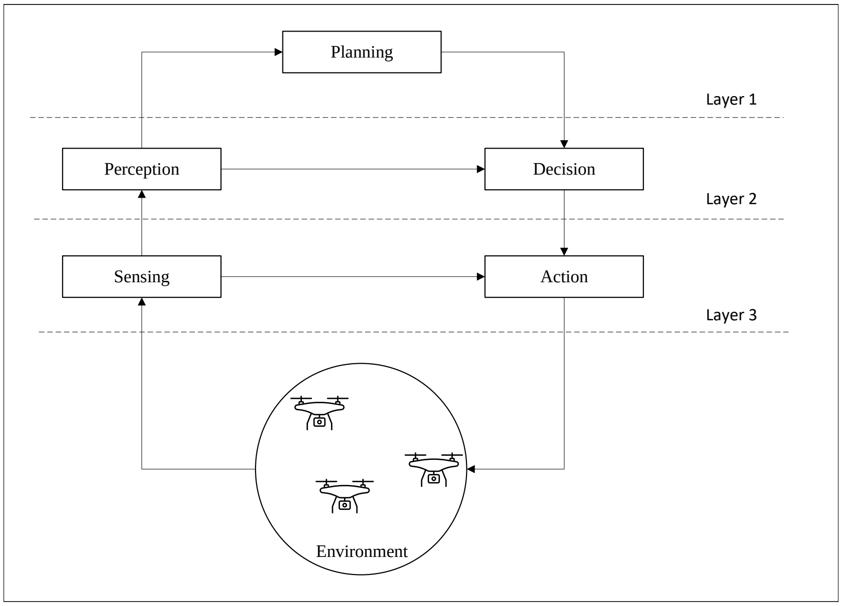


Figure 2.2: Model of Systems Control Behavior

## 2.2 UTM Systems Architecture

Unmanned Traffic Management (UTM) systems are designed to facilitate the safe and efficient operation of unmanned aircraft systems (UAS), also known as drones. UTM systems are typically composed of various components that work together to provide a comprehensive framework for managing drone traffic.

The architecture of a UTM system typically includes the following components:

- UAS – Unmanned Aircraft Systems consists of Unmanned Aircraft (UA), Ground Control Station (GCS), and Communication and Navigation Systems (e.g. onboard systems).
- UTM service provider – The organization that are responsible for providing UTM services, such as conflict management, communication and coordination, and surveillance and monitoring.
- UTM infrastructure – The hardware and software components that are used to support UTM services, such as air traffic control systems, communication networks, and databases.
- UTM data – The data that is collected and processed by the UTM system, including information about the movements and activities of UAS, as well as weather conditions, airspace restrictions, and other relevant factors.

Following the description in section 2.1.2, the architecture of UTM systems is needed to support the safe and efficient operation of UAS by providing a framework for managing drone traffic and addressing potential risks (i.e. mid-air collision) that may arise. For the purpose of this work, we have adopted a comprehensive high level architecture proposed in [7] and depicted in Figure 2.3. The diagram-notation is based on a decision framework [74] developed to represent human-automation function allocation.

In systems architecture depiction, rounded rectangles represent systems that offer a particular function in the sense of controller capabilities (e.g. Tactical Deconfliction). Rectangles represent surveillance capabilities split into specific characteristics (e.g. non-cooperative) and dashed round rectangles represent boundaries of a systems comprised of two or more subsystems (e.g. GCS). The triangle is used to show the controlled systems, i.e. the system whose state is intentionally altered due to some event (e.g. aircraft). Information flow is illustrated with arrows and the small blue rectangles are used to denote interfaces between information flow.

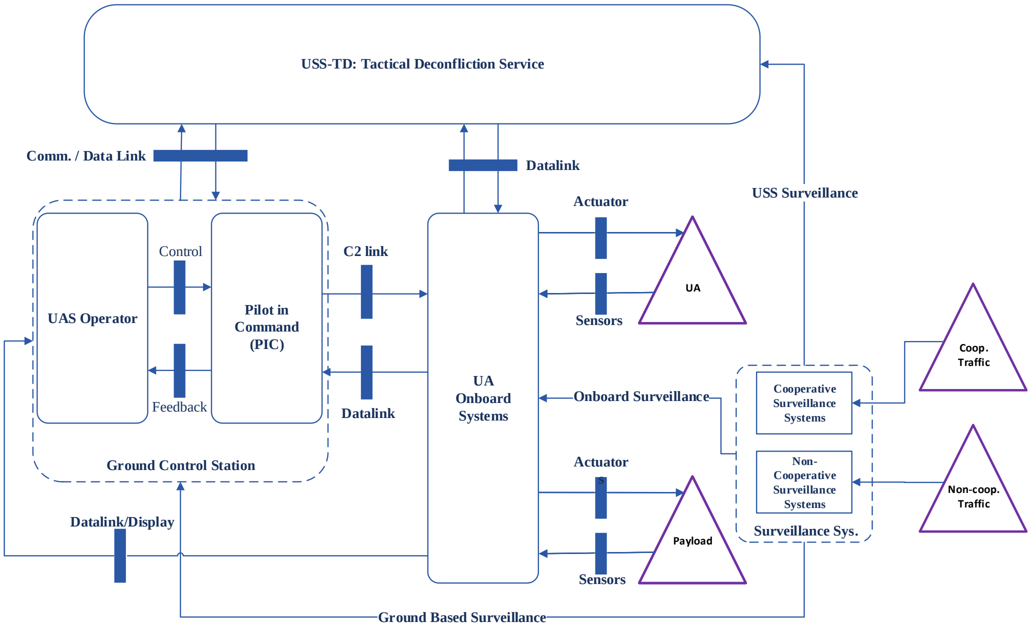


Figure 2.3: High-level UTM systems architecture

### 2.2.1 Systems of Systems Approach

Unmanned Traffic Management (UTM) systems can be considered as a type of "system of systems" (SoS). A system of systems is a complex, interdependent network of systems that work together to achieve a common goal. Each individual system within the network is autonomous and may have its own goals and objectives, but the overall SoS is designed to coordinate and integrate the capabilities of these individual systems to achieve a larger, more complex mission.

UTM systems are a good example of an SoS because they involve the integration of a wide range of systems and technologies, including communication systems, sensors, and software, that work together to enable the safe and efficient operation of drones in the national airspace system. These individual systems may be developed and operated by different organizations, but they are all connected and integrated through the UTM system to achieve a common goal.

In this work, CM is considered a constituent system of a larger UTM SoS, and TD is considered a subsystem of CM. For detailed and comprehensive information on analysis of ATM as SoS, we suggest this PhD thesis [75].

### 2.2.2 System of Interest: Tactical Deconfliction

As discussed in section 1.2.2, to prevent a LoS, UAS make use of *separation provision*(i.e. a possible UTM service), which is a *tactical process* for keeping aircraft away from hazards by at the least the appropriate separation minima [8]. According to ICAO, separation method refers to a set of approved rules, procedures and conditions associated with the separation minima [8], also referred to as separation thresholds in this work. While estimating threshold values, various factors are considered, such are UAS characteristics (e.g. size), performance (e.g. ground speed, turning rate) and an acceptable collision risk level [76]. There are three types of separation thresholds applied in UAS missions: distance-based, time-based, and a combination of both, time-distance-based.

- Distance-based separation threshold – is the simplest and can be seen as a spacial boundary around the aircraft, e.g., a cylindrical volume with height $H$ and radius $R$, if which is infringed by an intruder, a loss of separation has occurred. A drawback of this approach is not taking in consideration the intruder speed, in an explicit way.

- A time-based separation on the other hand, takes into account the relative speed of UAS (i.e. closure rate) by calculating time to the closest point of approach ($T_{CPA}$ ). If the estimated $T_{CPA}$ is less than a predetermined time threshold, it is considered as a loss of separation event [77, 78].

- hybrid – a time and distance based separation, combines the advantages of both metrics and has become the tendency of defining safe separations in UAS.

#### Well Clear Concept and Detect and Avoid

When UTM services fail and a UAS determines it is still in conflict, a *self-separation* maneuver is taken. The self-separation is a function carried out by the UAS Detect and Avoid (DAA) system and intended as means of compliance with regulatory requirements to remain *well clear* of other airborne hazard [28]. The Detect and Avoid capabilities are illustrated in Fig.5.1. Nominal DAA capabilities comprise three main modules: Conflict Detection, Alerting and Conflict Resolution. Once a conflict is determined the DAA capabilities offer three resolution functions (services): Remain Well Clear (RWC), Well Clear Recovery (WCR) and Collision Avoidance (shown with yellow, orange and red respectively). The difference between this functions is based on the objective, triggering event and maneuver behaviour. The triggering events can be thought as thresholds when a particular function should be activated. For instance, if two sUAS are closer than a specified RWC threshold, a maneuver should initiated as soon as possible to prevent an infringement of Well Clear boundary. In addition, DAA offers situational awareness in form of cascade alert levels depending on the risk severity.

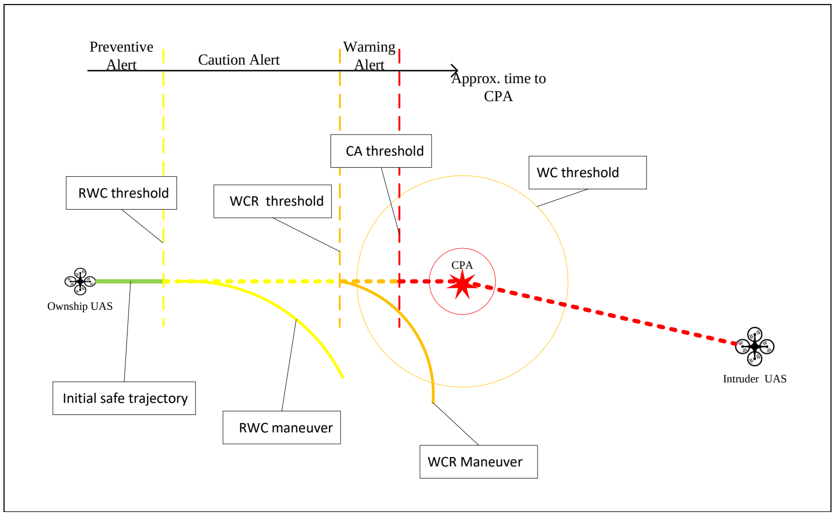


Figure 2.4: DAA capabilities, based on [28, 79]

A typical DAA system is composed of CN subsystems, sensors, conflict detection module, alerting and guidance algorithms, ground control station and command and control (C2C) subsystems. In Fig.5.2 we show a block diagram for a plausible autonomous DAA system for sUAS. The DAA system can be implemented on-board of sUAS and/or on the ground.

In case of autonomous flights, the navigation and maneuvers are made possible by the use of a flight computer, referred to as the autopilot (AP). Each one of these components adds a delay lag in the overall time response of a particular DAA system, which directly effects the quantification of the separation minima(e.g. remotely guided sUAS must take in consideration human factor, which adds a specific t seconds delay).

The concept of well clear has been proposed as an airborne separation stan-

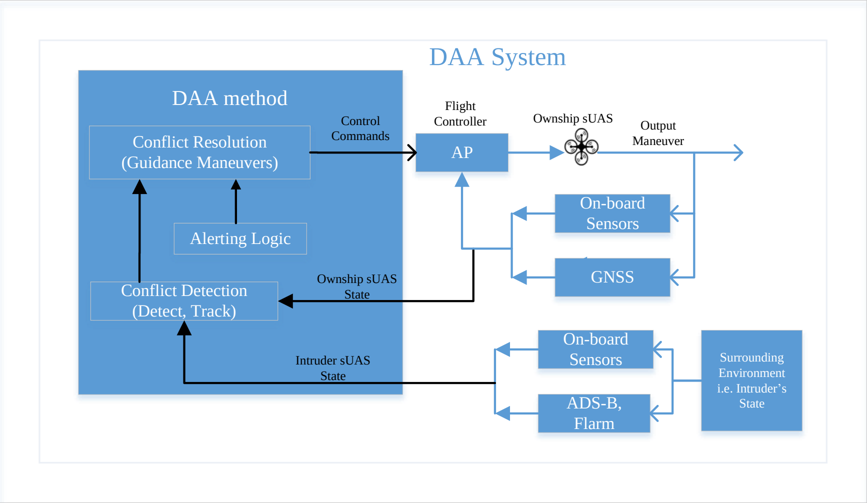


Figure 2.5: sUAS DAA system block diagram based on [46, 81]

dard to which an DAA system must adhere [80]. This notion is mentioned by FAA-defined Vision Flight Rules (VFR) and used in ICAO Annex 2 Rules of Air, but neither of them provides an exact definition for the concept, nor specifies any minimum separation threshold. Defining well clear for UAS is challenging because of the need to quantify a separation standard that is determined subjectively by pilots. If WC threshold is too small, unacceptable collision risks could arise. On the other hand a large threshold could impact the airspace system in various ways (e.g. capacity). Therefore, the challenge is to find an acceptable definition and quantification that ensures safety while minimizing operational impacts.

While there is no standard definition of well clear, two main functions are associated with this state: *Remain Well Clear* (RWC) and *Well Clear Recovery* (WCR). In terms of tactical conflict management, RWC is equivalent to the self-separation function, which aims to prevent a loss of WC to occur through smooth maneuvers that consider several factors (e.g. safety, operational, mission) [82]. *Well Clear Recovery* (WCR) is a function activated seconds before an unavoidable loss of WC and/or when an actual loss of WC occurs. In this situation DAA systems should give directive maneuvers, such that the sUAS regains its previous state. Both of these functions are related to a well clear notion, which is mainly viewed as protection volume around UAS, referred to as well clear volume (WCV) [28, 83, 84]. This volume can be specified by spatial thresholds, temporal thresholds, or both at the same time, referred to as separation minima in this work.

In addition, the near-mid-air-collision (NMAC)[1] represents the last safety volume. As the name suggests, a distance smaller than NMAC represents a

---

[1] here referred as small NMAC(sNMAC) to indicate the UA category

very severe loss of well clear that could result in a collision in the worst case. This distance is usually defined based on the dimensions of the UAS and its navigation performance [85].

### DAIDALUS as a reference DAA system

In this thesis, we decided to use Detect and Avoid Alerting Logic for Unmanned Systems (DAIDALUS)[2], as a reference system for modeling and simulation purposes. DAIDALUS is a formally verified software library, that implements a configurable DAA concept intended to support the integration of UAS into civil airspace [46, 84]. The core services provided by DAIDALUS are situational awareness through alerting logic, conflict detection (CD) and maneuver guidance. It is intended in aiding to maintain WC status via RWC maneuvers, and WCR in case the WC status is lost [46]. To do so, DAIDALUS uses linear projections of both ownship and intruder in a given look-ahead time $T$.

The DAA alerting logic is to provide critical timing information to the RP and/or to an autonomous system, regarding a potential loss of WC with a conflicting aircraft [86,87]. The alerting algorithms utilized in DAIDALUS span three level of redundancy based on the projected time to loss of WC (LoWC), within $T$:

- Predictive, intended for monitoring and situational awareness. No actions are taken at this level.
- Corrective, requires immediate awareness and a subsequent response from RP and/or autonomously to prevent a loss of WC. In this work it is consider as a time-based self-separation threshold (see Fig.2.1) and is associated with RWC function.
- Warning, indicates a loss of WC, therefore an immediate response is required. In our model it corresponds to WCR function.

In the CD logic, DAIDALUS uses parametric WC volumes to determine well clear status between pair of aircraft. The WC volumes are easily configured and serve as separation minima for computing maneuver guidance. Maneuvers can be suggestive to help the remote pilot and/or directive in more severe situations, i.e. WC recovery.

---

[2] https://github.com/nasa/daidalus/tree/v2.0.1

# Chapter 3

# State of the Art

This chapter is divided into two sections. In section 3.1 will give an overview of systemic analysis on TD systems (e.g., functional analysis), while in section 3.2 we summarize methods and standards applied in UAS/sUAS operations.

## 3.1 Systems-based approach to TD

The need for systemic understanding and analysis is recognized as a fundamental step in the process of designing, developing, and integrating ATM/UTM systems. In aviation this need has been acknowledged and there is a considerable work done in this direction, particularly in functional analysis and allocation.

A good start of understanding separation provision and comprising functions is a survey written by Karl.D.Bilimoria [88]. The author defines a comprehensive taxonomy for the allocation of separation provision functions with respect to locus of control (e.g. ground-based) and level of automation (e.g. ATC controller is responsible). The intent was to identify trends and gaps of separation provision, to better guide future works. The authors in [89] propose a framework which provides a hierarchical functional structure and allocation of these functions to corresponding system elements. This work can be seen as an extension of [88], with a particular focus on UAS operations and the intent to identify key functions and capabilities by decomposing high-level system goals into smaller functions. In addition, it also identifies primary system elements to perform the identified functions by decomposing the whole system into smaller systems hierarchically.

In a more recent work [90], the authors proposed a framework for decomposing the functions required to ensure safety in UAS operations, while assessing all the agents (i.e. subsystems) in the system and classifying the levels of autonomy (i.e. control modes). In a later paper [91], the authors use the aforementioned autonomy framework to provide a functional analysis with the aim of exploring the separation provision function for UAS operations. Note that the authors use Remotely Piloted Aircraft (RPA) and Separation Assurance (SA), instead of UAS and SP, nevertheless, we decided to use same terminology throughout all this thesis.

## 3.2 Self-Separation Methods

In order to quantify a self-separation standard (i.e. self-separation minima and method) for sUAS two main approaches are identified based on an extensive literature review. The typical method, which is adopted for the development of RTCA and ASTM DAA standards, is based on unmitigated collision risk analysis. A well clear separation is defined as relative separation where a desired unmitigated risk threshold is achieved. The evaluation is done based on simulated Monte Carlo encounters that take in consideration representative flight trajectories and environmental uncertainties (e.g., wind). The other method is based on defining safe separation boundaries around UAS, generally characterized by the UAS performance, operational constrains and related uncertainties. UAS performance includes aircraft maneuver capabilities, CNS uncertainties and other associated systems performance such as a DAA system. This method tends to model the behavior of each component and requires unique separation boundaries with respect to the UAS. For instance, a fixed-wing UAS would have a different safety boundary compared to a quad-rotor. Each component affecting this safe boundary has different impacts in horizontal and vertical separation criteria. A good example is the difference in the dynamics between horizontal and vertical maneuvers. Moreover, sensor accuracy, flight controller behavior, wind influence and so on, change on how they affect the aircraft in the horizontal and vertical dimensions. Therefore, to quantify a WC volume the horizontal and vertical criteria are studied separately.

### 3.2.1 Well Clear as Self-Separation Standard based on unmitigated collision risk analysis

In the last decade there has been a lot of effort to define and quantify Well Clear as self-separation standard that can be applied to UAS, sUAS, and other Advanced Air Mobility (e.g. Urban Air Mobility). Wiebel et al. make an important case of using and defining WC based on an acceptable collision risk value [92]. According to this work, the separation standard may vary according to what the regulator entities consider an acceptable risk level of a NMAC occurring, given the relative state of a pairwise encounter. The model takes in consideration TCAS alerting criteria and recommends a 8000 ft threshold for head-on encounters and 3000 ft for track-crossing and/or overtaking encounters.

In 2013, the Second Caucus of the FAA Sense and Avoid Workshop endorsed the idea that WC for UAS is a separation standard and recommended for it to be time-based (i.e., number of seconds prior to near mid-air collision) in the horizontal plane, and distance-based in the vertical plane [93]. The workshop considered three UAS well clear concepts by NASA, MIT LL, and Air Force Research Laboratory [76]. Based on the conclusions and recommendations of the workshop, in August 2014 Unmanned Aircraft System (UAS) Executive Committee Science and Research Panel (SaRP) and the Radio Technical Commission for Aeronautics (RTCA) Special Committee 228, defined well clear as a volume that relates a modified tau ($\tau_{\text{mod}}$) value of 35 seconds with a distance threshold (both a minimum distance and horizontal miss distance filter) of 4000 feet in the horizontal plane. The vertical Well Clear definition was determined by a fixed distance from the ownship of 700 feet [94].

Munoz. et. al. [95] brought a formal definition for Well Clear, by giving a mathematical foundation for the concept, based on TCAS II logic and ICAO principles. His work progressed with the implementation of a set of DAA algorithms, known as DAIDALUS [46]. In conjunction with RTCA, a standard for DAA systems was released in 2017, DO-365 [96]. It uses a $\tau_{\text{mod}}$ value of 35 seconds with a distance threshold (both a minimum distance (or, distance modification - DMOD) and horizontal miss distance filter) of 4000 feet in the horizontal plane. The vertical component of the Well Clear definition was determined by either a distance from ownship UAS of 450 feet or a time-to-co-altitude value of 20 seconds. This recommendation was not adequate for small UAS and VLL operations. Hence some scaling of the parameters was done.In addition, a Well Clear boundary of 2200 feet laterally and 450 feet vertically is proposed for non-cooperative encounters. The selection was based on findings in [97, 98] and published in later review of DO-365B [99]. Note that the terms of reference and scope of the standard currently does not include sUAS-sUAS encounters.

A more recent recommendation is defined by MIT LL, considering small UAS in VLL operations [26]. In distinction from the first RTCA DO-365 MOPS recommendation, it uses only spatial metrics, using a “hockey-pock-shaped” volume with a distance threshold (both a minimum distance (or, distance modification - DMOD) and horizontal miss distance filter) of 2000 feet in the horizontal plane. The vertical Well Clear component is determined by a fixed distance from the ownship sUAS of 250 ft. These metrics were adapted and published as part of American Society for Testing and Materials (ASTM) Standard Specification for Detect and Avoid System Performance Requirements (ASTM F38) [100]. It is worth mentioning, ASTM F38 DAA performance standard is only applicable to avoidance of manned aircraft by sUAS and not sUAS to sUAS.

### 3.2.2 Self-Separation based on sUAS performance

Michael. M. et. al. [101] proposed a time-based separation method, applicable for small UAS operations. Using worst-case analysis regarding UAS maneuverability and ground speed range, they show how to generate dynamic separation thresholds. However since their time metrics were based on the recommendations of Second Caucus Workshop (i.e. suitable for large UAS and UAS-manned aviation encounters), the values resulted way too large to be considered adequate for sUAS operations.

In this work [102], the authors focus on sUAS operations in urban environment. They propose a preliminary WC volume with radius of 20 ft and half-height of 24 ft. Given the low sUAS speeds and high maneuverability (i.e.

high turning rate), they demonstrated that safety might be acquired with much less conservative thresholds. The authors argue that these parameters are proposed due to sUA small size and the capability to do turn maneuvers with 30 degrees per second. Nevertheless, no explicit methodology is given in how to define well clear for sUAS.

McLain. et.al [27] analyzed high density sUAS operations and proposed a methodology to define well clear based on the limitations of an ADS-B dependant airspace and sUA maneuver capabilities. The authors calculate spatio-temporal self-separation thresholds by determining minimum distance and time between an intruder and maneuvering sUAS. Similar to Michael. M. et. al. [101] this separation minima depend on horizontal maneuver capability in stressing head-on encounters. The standard definitions recommended by SARP are analyzed to demonstrate their method. Results showed that the recommendations were too conservative for sUAS operations and recommend for horizontal WC definition to be 3200 ft distance or a a modified tau ($\tau_{\text{mod}}$) of 25 seconds.

Considering a service oriented airspace (i.e. UTM/U-space), this work [103] successfully simulates safe sUAS delivery missions. Each sUAS is subscribed to CD&R services which help the vehicles to keep a self-separation distance. The authors recommend horizontal thresholds varying from 30 to 45 meters based on Total System Error of the sUAS and an arbitrary safe separation minimum.

There is no (up to date) definitive well clear concept or another alternative approach recommended from U-Space. Nevertheless, CORUS as part of the initial projects, proposes some minimal distances to be considered at VLL operations [104]. In the case of beyond visual line of sight encounters (BVSOL) between two sUAS, a horizontal distance of 250 ft and 150 ft vertical, is considered as separation minima. To the best of our knowledge no explanations whatsoever are provided to the open public regarding the methodology.

## 3.3 Gaps and Limitations

To give a comprehensible overview on Well Clear standard for sUAS, we extracted the main processes(i.e. activities) — which refers to the method of choice, model assumptions, simulation and experimental set-ups — that each work in literature review has considered to define WC (see Table 3.1).The authors recognize all activities as complementary elements that need to be taken into account, and do not compare the weight of their importance.

As evidenced, a lot of work has been done related to safety of UAS and recently small UAS. Section 3.2.1 summarizes works that base their contributions on principles 1,3,4,8 and partially 2 (since unmitigated collision risk analysis does not require any type of DAA systems). While this approach has been proved to have significant contributions on the standardization process of Well Clear for medium to large UAS, it is difficult to be adapted and define adequate self separation metrics for sUAS operations. One main concern that the authors have, relies on the compatibility with the DAA systems. In our understanding, the DAA system role is far more important in sUAS environments, than in case

Table 3.1: sUAS WC processes

| Extracted processes from Literature Review | Nr. |
|---|---|
| Unmitigated Collision Risk Method and analysis | 1 |
| sUAS (aircraft and systems) performance behavior | 2 |
| Fast simulations / Monte Carlo | 3 |
| Operational Acceptability | 4 |
| Simulations in the Loop / Hardware in the loop | 5 |
| Real Flight Tests | 6 |
| Scripted Encounters (head-on, crossing...) | 7 |
| Representative sUAS trajectories | 8 |
| sUAS-sUAS encounters | 9 |
| Environmental Uncertainties (e.g. wind) | 10 |

of large UAS or manned aviation due to access of Separation Provision Services (see section 1.2.2). Furthermore, this approach requires static separation thresholds, which has negative impacts on the airspace capacity.

In our opinion, the primary contribution of the Section 3.2.1 lies on the identification of the encounter geometries between sUAS, rather than in direct quantification of self-separation thresholds. It is worth mentioning that none of the reviewed works, considered sUAS-sUAS encounters while attempting to define WC criteria.

Section 3.2.2 summarizes the works that follow principles 2,3,4,5,7,9 and 10. Since this approach is mainly based on the sUAS performance, typically *Scripted Encounters* are used to create stressing situations(e.g. head-on) and evaluate the performance of each system. This might be one of the trade-offs that this method has to consider, which can be remedied by the work derived from Section 3.2.1. In addition, it was observed that the quite often authors quantify the criteria based on expert's experience [102,103,105] and/or deriving from manned aviation standards(e.g TCAS metrics) [27, 101]. While we do agree that *scaling factor* presumptions can be used to evaluate DAA systems and methods, it appears not to be very rigorous when it comes on determining Well Clear.

Given this picture of sUAS Well Clear separation standard, the authors attempt to overcome the limitations mentioned above, by formally justifying their assumptions and utilizing fast simulations to verify and give the most adequate recommendations based on severity of loss of separation and operational considerations.

## 3.4 Effects of External Parameters on DAA Systems

The effects of a number of factors and parameters have been evaluated to understand the influence they have on ensuring safe separation. Lee et. al. [106], provide two analyses regarding effects of the well clear threshold. Firstly, they give a study in dependencies of well clear metrics on the rate well clear violation occurrence. Secondly, a relationship between ATC separation and well clear volume definition. As part of a work from NASA Ames Research Center [86], a detailed evaluation of alerting logic and pilot response delay is shown. Three main parameters of DAA systems were checked as independent variables: trajectory prediction, alerting time threshold, and alerting distance threshold. Results indicated DAA alerting distance has a greater effect on DAA system performance than alerting time or ownship trajectory prediction.

Consiglio et. al. [28], investigate different performance parameters such as a variety of well clear volumes, initial conditions, and encounter geometries. Kim et. al [107], suggest a methodology to assess the conflict risk of sUAS traffic. It is shown that conflict risk is affected by the flow rate, the speed of sUAS, the intersection angle, and the number of sUAS. More research investigating other attributes of DAA systems such as Speed Range [108] [109], Turn Performance [110] [109], Limited Surveillance Volume [111] has been done, giving different aspects and propositions that would be of interest for RTCA, ICAO, and other interested organizations.

## 3.5 Alternative DAA systems

Aircraft Collision Avoidance System X (ACAS-X) [112] is projected to play a key role in the safety of the Next-Generation Air Transportation System (NextGen) and replace the currently deployed TCAS-II [14]. Based on this concept a new version ACAS-Xu [113] has been developed to provide DAA capabilities to UAS. It meets the functional requirements proposed defined by Minimum Operational Performance Standards (MOPS) and provides alert and guidance logic. Recent research has extended ACAS-Xu into ACAS-sXu , which takes into consideration the challenges raised by sUAS operations [47]. Based on similar approach systems like JADEM [114], SAFIT [102], and CPDS [115] are used to evaluate and test DAA systems that comply with the recommended MOPS.

# Chapter 4

# Systemic Analysis of sUAS Tactical Deconfliction Systems

In this chapter we present state of the art of the main approaches regarding CM at tactical level and provide a systemic analysis about their service based capabilities. Systemic analysis is a holistic approach to understanding and analyzing a system and its components, interactions, and environments. In the context of systems engineering, it involves examining the system and its components in terms of their functions, requirements, and performance, as well as the relationships and dependencies between them. Typically it includes the following steps:

- Define the system – Clearly define the boundaries and scope of the system, and identify its key components and functions.
- Analyze the system and its components – Identify and analyze the relationships and dependencies between the system and its components, as well as the interfaces between the system and its environment.
- Identify and analyze the requirements – Identify the system's functional and performance requirements, and analyze how these requirements will be satisfied by the system and its components.
- Develop and validate the system design – Based on the results of the systemic analysis, develop a conceptual design for the system and its components, and validate the design through simulations, prototyping, and testing.

## 4.1 Object Process Methodology

To conduct this analysis we use Object Modeling Methodology (OPM) [116], which is considered a holistic approach for conceptual modeling of complex systems. OPM uses diagrams to represent the components, processes, and interactions of a system. OPM diagrams are used to analyze and design systems, as well as to document and communicate their structure and behavior.

An OPM Systems Diagram (SD) is a graphical representation of a system that shows the relationships and dependencies between the system's components and processes. It typically includes the following elements:

- Objects – Objects represent the physical (shadow colored) or abstract components of the system, such as sensors, actuators, or data stores.
- Processes – Processes represent the activities or transformations that take place within the system, such as data processing or control functions.
- Links – Links represent the interactions and dependencies between objects and processes, such as the flow of data or control signals. OPM defines four essential relation categories: specialization, combination, exemplification, and representation. Relationships can be either one-directional or bidirectional.
- Environments – Environments represent the external factors that impact the system, such as the physical or social context in which the system operates. Environmental objects are not part of the SoI and are denoted with dashed line rectangles.

## 4.2 Structure and Behavior Analysis

*Systems structure* analysis is a method of analyzing the structure and organization of a system and its components, in order to understand how the system functions and performs. It involves examining the relationships and dependencies between the various components of the system, as well as the interfaces between the system and its environment. Whereas, *systems behavior* analysis is a method of analyzing the behavior of a system and its components in order to understand how the system functions and performs. It involves examining the inputs, outputs, and processes of the system, as well as the interactions and dependencies between the system and its environment.

In Figure 4.1 is shown a high level OPM SD [1], depicting the comprising structure of a *safe sUAS Operation*. There are three main components nested inside the operation behavior (e.g. packet deliver): 1) UAS, 2) TDS and 3) Operational Environment.

---

[1] All the modeling is done using OPCLOUD software: https://www.opcloud.tech/

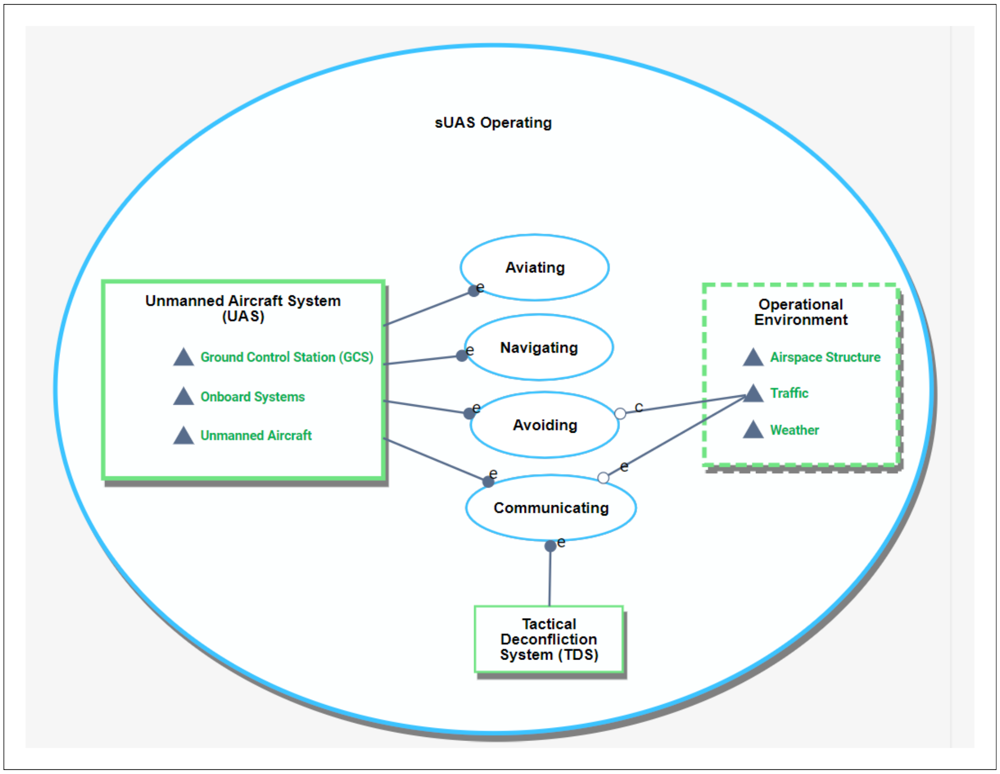


Figure 4.1: High-Level System Diagram

### 4.2.1 ICAO approach

According to the ICAO Doc9854/AN458, TD is responsible for mitigating midterm conflicts through gentle maneuvers in a timely fashion, also known as the SP function. In case that separation provision is compromised, CA is activated, which identifies short-term (imminent) intruders and performs last-resort maneuvers to prevent mid-air collisions. It is worth noticing that CA is considered as an independent layer and not included as part of calculation the target level of safety of separation provision. This model is reinforced with the publication of ICAO UTM Common Framework document [17]. In this document it is stated that the principles used in manned aviation can be applied to UAS operations, however there is a need for new methods, standards and new technological requirements. The nature of the document is suggestive and technological agnostic. In the Figure 4.2, a summary of TD related services are depicted and defined as following:

1. Tactical Separation: should be a real-time service that provides information about manned aircraft, such that UAS remain well clear of manned aircraft.

2. Conflict Advisory and Alert: should be a real-time service that provides to the remote pilots alerting through suggestive or directive information with regard to the proximity other airspace users (manned and/or unmanned).
3. Dynamic Reroute Service: should be a real-time service that provides adjustments to the intended operation volume, route or trajectory, taking into account the minimization of the likelihood of an airborne conflict and maximizing the likelihood to conform airspace restrictions.
4. Conformance Monitoring Service: should be a real-time service that provides to UAS operator or remote pilot monitoring data and alerting of non-conformance to intended operational volumes, routes, or trajectories.
5. Collision Avoidance: No additional information is included in the document (see ICAO Doc9854).

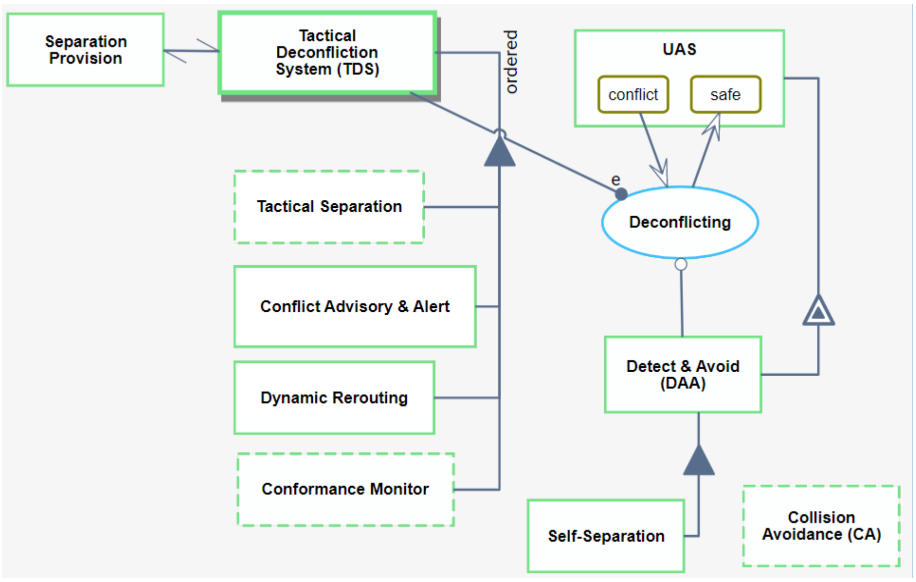


Figure 4.2: ICAO TD System Diagram

The provision of a safe distance or safe time between UAS in flight can be accomplished by a coordination between TD service and GCS, and/or an automated RWC function on board the UA. According to ICAO, separation provision (i.e. TD) should not to be confused with the UA's CA function as the final safety barrier. This CA feature is not taken into account while calculating the safety level.

It is worth noticing the document argues the TD provided as a service by UTM systems is considered as *reactive* conflict management. Moreover, ICAO suggests that for low density traffic, drones may be permitted for self-separation

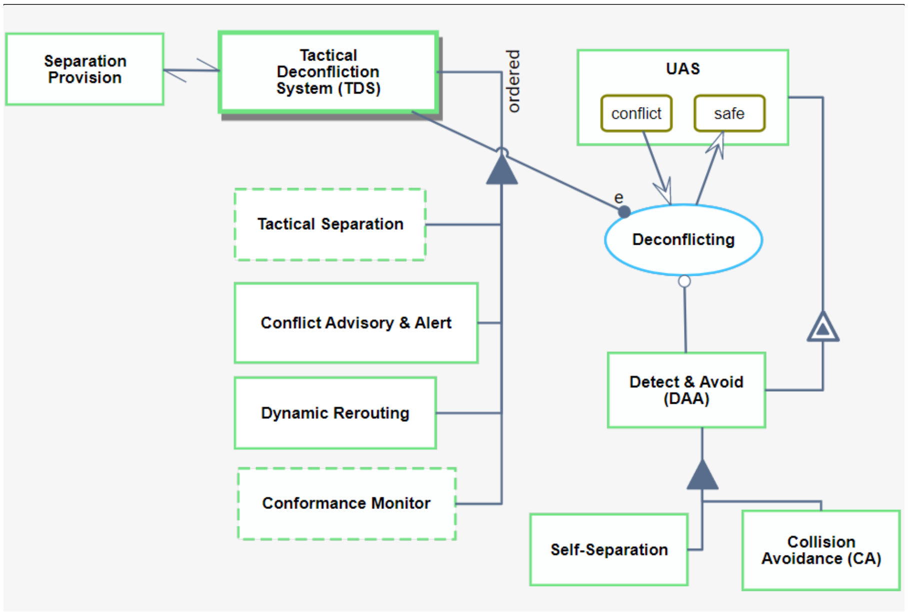


Figure 4.3: NASA TD System Diagram

using on-board DAA capabilities, or alternatively remote pilots may navigate according to information received by the UTM or visual acquisition [17].

Based on the proposed framework1.6.4 and the model for determining systems behavior2.1.3, the ICAO argument on describing TD system as reactive is incorrect. As it is illustrated above, all comprising components should at least take a decision before providing a response to the environment (or input to another system). Therefore these type of subsystems belong to Level 2 and have an adaptive nature. In case of Dynamic Reroute, depending on the degree of optimization, it can be classified even as Level 1, given a planning step takes place.

### 4.2.2 FAA-NASA approach

In a recent publication, NASA-FAA proposed a new and detailed CM model that will support UAS operations under the UTM system of systems [18]. This model is based on ICAO ICAO Doc9854/AN458, but it is more comprehensive, cohesive and concrete in terms of deconfliction functions suitable for UAS operations. Figure 4.3 depicts all the services comprising this model.

In the context of separation provision dealing with airborne risk three subfunctions are identified:

1. Conflict Advisory and Alerting Service
2. Dynamic Rerouting Service

3. Detect and Avoid

According to this model, TD can be accomplished by functions provided only by UTM services or a USS plays a supporting role and separation is accomplished by UAS operator and functions onboard UAS. In this approach no characterization of behavior type is given. Nonetheless, the description per each model component is comprehensive and we use it as a baseline for the proposed TD model in section.

### 4.2.3 U-Space approach

There is no definitive model proposed by U-Space regarding CM systems. U-Space advancements are done through a set of different SESAR research projects, that serve as a primary example of the issues we raise in this work. In an initial ConOps document [117], there was no detailed information regarding CM, apart the general ”saying” – CM is applied in two layers Strategic CM and Tactical CM. In addition, services as Emergency Management and Monitoring were considered as separate services and not under CM.

In the latest ConOps update [118], a more structured information was provided; the services are depicted in Figure 4.4.

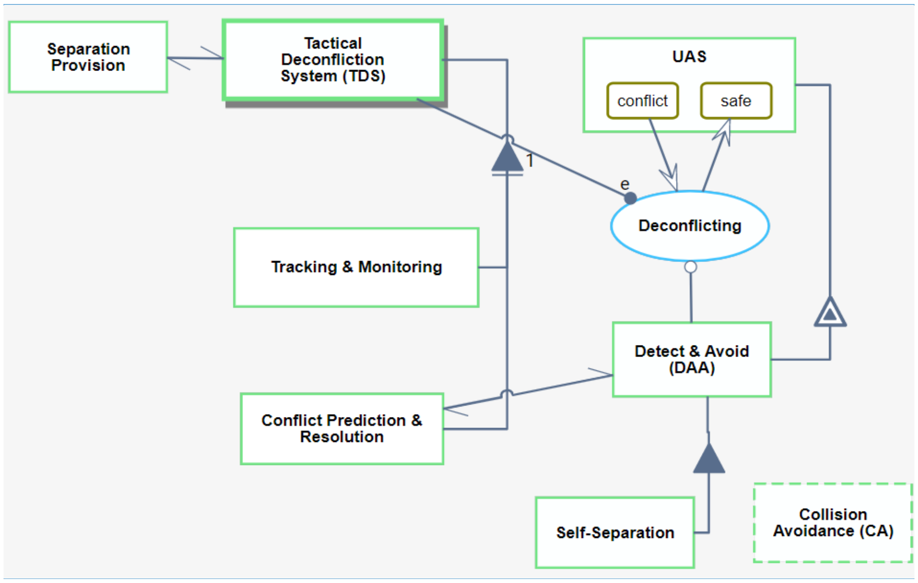


Figure 4.4: U-Space TD System Diagram

In the context of managing airborne risk at tactical level the following functions are identified:

1. Emergency Management

2. Tracking and Monitoring
3. Conflict Prediction and Resolution

It is worth noticing that Conflict Prediction and Resolution is utilized in the same context as Detect and Avoid. It is unclear why the authors have chosen to use different terminology and there is a lack of further clarification. Similar to NASA approach no characterization of systems behavior is mentioned in the document.

## Chapter 5

# Dynamic Self-Separation Thresholds for Autonomous sUAS

In this section we specify and recommend adequate values of well clear threshold & DAA alerting times for sUAS-only encounters. For this purpose, we adopt and extend a dynamic protection zone concept as separation method, based on [101], and use it to characterize dynamic well clear (WC) boundaries suitable for sUAS. The well clear boundaries define a safety volume (e.g. cylinder) such that sUAS pairs not occupying this volume simultaneously, are said to be well clear. This approach requires dynamic thresholds based on the performance of the aircraft, for instance, UAS with high maneuverability require smaller separation minima. Nevertheless, in their work [101] they do not consider the affects coming from the DAA systems and other uncertainties that influence sUAS operations, to which we consider as key components.

To verify the well clear threshold specifications and to study the effects that uncertainties (such communication delays, wind estimation errors, and navigation errors) have in the relationships of our metrics, we run closed-loop fast simulations in ICAROUS[1]. We assume that ownship sUAS is equipped with DAIDALUS [46] as a DAA method and the intruder traffic continues in straight line through the encounter, i.e. $V_i = constant$ and turn rate $\omega_m = 0$ rad/s. We found out that the dynamic well clear thresholds can ensure safety and be more efficient compared to previously adapted well clear definitions for medium-to-large UAS.

---

[1] https://github.com/nasa/icarous

## 5.1 Methodology

Based on the framework derived in chapter 4, we propose a generic methodology to quantify well clear (self-separation) based on both the unmanned aircraft and systems performance. Given the fact that to define a WC volume two separate studies are required, and since the application of the methodology is similar for both the horizontal and vertical criteria, in this work we choose to focus only on one of the two criteria. In particular, we focus on the horizontal criteria, which are preferred in sUAS operations since: (a) Horizontal conflict resolution maneuvers are more preferable and a two dimensional approach is a common assumption in CDR works [119]; (b) sensor accuracy is higher in horizontal dimension and performance of UAS is affected by flight level; (c) sUAS operate mostly in low altitudes, and flight-level regulations or constraints may cause sUAS to maintain flight level during their operations [120]. This might be to decrease the risk of collision with high buildings in urban areas. (d) It is also considered as conservative assumption; any method that operates adequately in two dimensions is likely to be able to perform adequately in three [121]. In any case, to the best of our understanding, two dimensional studies are useful and sufficient for preliminary investigations.

In this study, we propose a separation method comprised in two layers of safety zones, as illustrated in Fig. 5.1. The inner layer is a fixed circle with radius $\mathrm{R}_{\mathrm{sNMAC}}$, modeled after the Near Mid Air Collision (NMAC) concept, also referred as small NMAC (sNMAC) when applied to sUAS [122]. The outer layer is characterized by dynamic thresholds, which serve for sUAS to maintain self-separation. Here after we will refer to this area as Dynamic Well Clear Area (DWA).

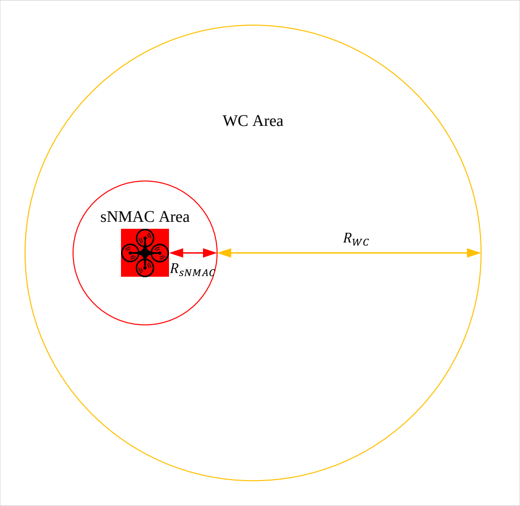


Figure 5.1: Well Clear and small NMAC area

### 5.1.1 small NMAC area

To determine $\mathrm{R}_{\mathrm{sNMAC}}$ we follow the method proposed in [123, 124], which considers the size of the UAS and an estimation of the total system error (TSE):

$$R_{sNMAC} = 2 \times MSW + TSE \tag{5.1}$$

, where in this case Maximum Wing Span (MWS) is the diagonal distance of sUAS and Total System Error (TSE) is composed of: Navigation System Error, Flight Technical Error, and Path Definition Error. For a more comprehensive discussion of TSE and its applications on small UAS, we suggest these papers [125, 126]. The sNMAC threshold is used to evaluate WC thresholds such that, WC threshold should be larger than sNMAC by an appropriate value that would prevent sUAS traffic getting to an unacceptable proximity (i.e. sNMAC cannot be evaded). In this work, we model a sUAS according to the characteristics of DJI Inspire 2 Quadcopter, which has MWS = 0.6 meters. To calculate TSE, we first need to assign the values per each component. Navigation System Error (NSE) is considered 2 meters, since it is the GPS standard accuracy. The values for Flight Technical Error and Path Deviation Error are obtained from [126], which under a normal distribution model for TSE, suggest a value of 3.58 meters. Therefore, an approximate value of sNMAC = 4 meters is used during our simulations.

### 5.1.2 Dynamic Well Clear Area

The Dynamic Well Clear Area is acquired from an early concept developed by US Air Force (USAF) [127,128] and a later work adapted for sUAS [101], referred to as Dynamic Protection Zone (DPZ). It is defined by a circle representing the maximum reach set of the projected trajectory of the sUAS, as shown in Fig.5.2. Note that that the center of the circle has an offset distance from the UAS track position. The overall size of DWCA is adjusted based on the UAS heading and ground speed.

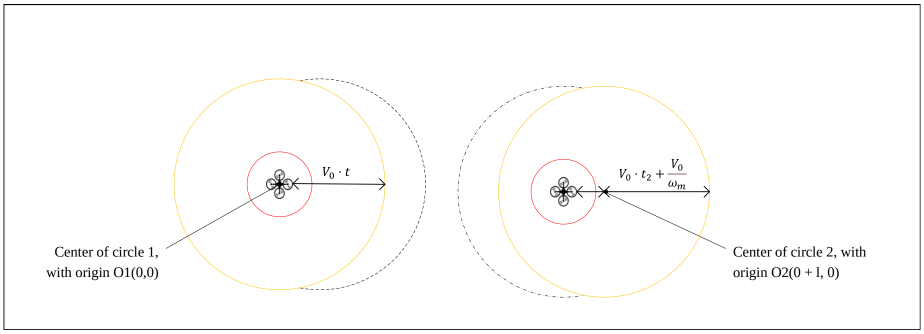


Figure 5.2: Non-maneuver reaching area (on left) and maneuvering reaching area (right)

The core idea behind this concept relies on the maximum range of maneuver that a sUAS can reach in a predefined time $t$. According to aforementioned papers, the heading change maneuvers can be grouped into three main modes: 1) sUAS turns at a turning rate until $t$ is reached; 2) sUAS starts turning until a heading change $\theta$, and then flies straight until $t$ is reached; and 3) sUAS first flies straight and then turns at a given turning rate. Utilizing basic turning flight dynamics, it was shown that the maximum displacement from the original track, in a given time $t$, is achieved by the mode 2. This mode creates a kidney-bean like geometric boundary and the widest point is reached when the sUAS turns at maximum turning rate $\omega_m$ and spends as much as possible time at level flight approximately 1.6 radians (i.e. 90 degrees) with respect to the original track [101].

The relationship between the estimated positions and turning mode is given in 5.2. Lets assume that ownship sUAS has a constant ground speed $V_o$ and a maximum turning rate (i.e yaw rate) $\omega_m$, then the the whole maneuver would consists of *turning* with $\omega$ for $t_1$ and *flying straight* with $V_o$ for $t2$, where $t1 + t2 = t$. Supposing that sUAS is a point in a Cartesian reference frame with coordinates $O1(0,0)$, then in case of a maneuver, all possible positions of (x,y) can be expressed as:

$$\begin{cases} \theta = \omega t_1 \\ x = R\sin(\theta) + vt_2\cos(\theta) \\ y = R + R\cos(\theta) + vt_2\sin(\theta) \end{cases} \tag{5.2}$$

, where $\theta$ is the yaw angle (rad) (i.e. heading change with respect to original track), and $R$ is the minimum *turning radius*, i.e. $R = \frac{V_o}{\omega_m}$. Note that, the original heading of sUAS is inline with x-axis and y represents lateral position of sUAS after $t$.

As mentioned above, all reaching points in mode 2, form an irregular boundary(i.e. kidney-bean) which would not be preferable as a separation standard. Therefore, a circle that encompasses this boundary is considered acceptable as separation boundary, without increasing its radius to sizes not acceptable for operational use. The radius of the circle is equivalent with the sum of maximum value of $y$ and $R_{sNMAC}$ as in 5.3:

$$R_t = R_{sNMAC} + V_o \times t_2 + \frac{V_o}{\omega_m} \tag{5.3}$$

, where $y_m = R + R\cos(\theta) + vt_2\sin(\theta)$, $\theta = 1.6$ radians, and $R$ is the minimum turning radius.

A visual description is given in Fig. 5.2, as it is shown in the right, the DWCA is modelled as a circle with radius $R_t$ with center $O_2(0{+}\mathrm{l},0)$, where $l$ is an offset from origin $O_1(0,0)$. The offset $l$ can be expressed as $l = |x - x_m|$ and can be determined by simulations or analytically. In here, $x$ is a random point and $x_m$ is the maximum reaching point along x-axis, calculated under the same conditions as $y_m$, using equation 5.2. We give an analytical solution for the value of $l$, which serves as a constrain to determine $R_t$:

$$\begin{cases} R_t = y_m + R_{NMAC} \\ (x - x_m)^2 + y_m^2 \geq |V_o \times t - x| \end{cases} \tag{5.4}$$

In this thesis, $R_{WC}$ is considered as self-separation minimum, by which the WC area is determined(see Fig.5.1). This threshold is directly proportional to $t$, which in our approach is the total time that ownship sUAS requires to autonomously (i.e. no RP in the loop) maintain and/or regain Well Clear state. Note that $R_{WC}$ is different from RWC. The former describes a distance-based threshold, while the latter is a function, i.e. perform a maneuver to avoid a loss of WC from occurring. We do not study RWC function in this work. The following subsection describes the evaluation of the separation minima.

### 5.1.3 Separation Minima

To quantify $R_{WC}$, we have to determine $t$. We compute adequate values of $t$ by considering it as sum of $t_1$, the time a sUAS needs to alter its heading by 90 degrees with respect to its original track; $t_2$ which is the time the sUAS flies straight at level flight after altering its heading; and $t_{TRT}$ which is the time of the system's total response time (i.e. the time between the moment of conflict detection to the moment the execution of a conflict free maneuver begins)

$$t = t_1 + t_2 + t_{TRT} \tag{5.5}$$

In our approach $t_{TRT}$ is considered an added safety buffer, to compensate the time lag of an on-board DAA system. It is composed of $t_{sens}$, the time the ownship needs to estimate the intruder's state (also referred to as sensors update rate); $t_{DAA}$, the time the DAA method needs to detect a loss of WC, generate a conflict-free trajectory and send a command to the autopilot; and the autopilot response time $t_{ap}$, that is the time lag the on-board system requires to generate the right parameters to execute the maneuver received from DAA:

$$t_{TRT} = t_{sens} + t_{DAA} + t_{ap} \tag{5.6}$$

Furthermore, in this study we focus on specifying warning alert time-thresholds $t_{al}$, required for as a time-threshold that would prevent an intruder sUAS to enter ownship's WCA by generating recovery maneuvers in case that loss of separation is unavoidable. We determine its value by using fast simulations and considering the following constraint:

$$T > t_{al} \geq t \tag{5.7}$$

## 5.2 Modeling and Simulation

Modeling and simulations are the chosen means to determine proper time values for $t$, which will be translated to spatial WC thresholds and serve as adequate separation minimum. To attain this, we generate sUAS-sUAS encounters such that they would result in a loss of WC, t-seconds after the run of simulation, unless an avoidance maneuver is initiated. The analysis are focused on the severity of loss of WC results and WCR maneuver performance. For this purpose two metrics are introduced and an analysis method that can be used to derive proper recommendations.

### 5.2.1 Simulation Environment

In this work, we utilize Independent Configurable Architecture for Reliable Operations of Unmanned Systems (ICAROUS) [2] as the simulation environment. ICAROUS is a software architecture that is designed for building autonomous unmanned aircraft applications. It is made of several core modules that include formally verified algorithms for detection, monitoring control of safety criteria. Furthermore, it comes with algorithms that avoid stationary obstacles and other airspace users. These algorithms calculate resolution and recovery maneuvers which are executed by the autopilot. ICAROUS incorporates DAIDALUS as DAA method. For our purposes, we rely on Pycarous, which is a Python wrapper for the core ICAROUS modules written in C++. As such, Pycarous allows for faster than real time. closed-loop simulation i.e. including a DAA system to mitigate possible conflicts. Furthermore it allows the implementation of near-realistic operational environment by adding uncertainty in several factors. More specifically, the positions of ownship and intruder are uncertain according to a Gaussian distribution $\mathcal{N}(0, 2)$ (i.e. mean 0 and variance 2 meters, set according to GPS technology parameters). Regarding the sensors update rate, we have assumed that both sUAS are equipped with ADS-B like type of sensors. In the simulation environment, the sensors update rate is modelled as a communication delay. Based on the current development and recommendation a reliable ADS-B update rate is considered between 1 and 2 Hz [129]. However, it is suggested that for sUAS operation under a UTM ecosystem, the update rate might be further increased as part of the requirements of DAA systems [130]. Therefore, in this study we model communication delay(i.e., vehicle-to-vehicle) as a triangular distribution with minimum 0s, most likely value 1s and maximum 2s. Finally, wind speed is modelled by a Weibull distribution with shape 3.2 and scale 2.2, while wind direction is modelled by a uniform distribution between $[0, 2\pi]$ [126, 131]. In order to get statistically meaningful results, we utilize Monte Carlo simulations with 400 simulations per scenario setting. One simulation corresponds to a random sample of each variable modelled by a probabilistic distribution.

### 5.2.2 Assumptions

The ownship and intruder UAS are modelled as point-mass. The ownship has constant ground speed $v$, and turning rate $\omega$. On the other hand, the intruder

---

[2] https://github.com/nasa/icarous

cannot make maneuvers to change its speed or heading (i.e. fly through encounters). The reason the authors do not consider a maneuvering intruder is to enforce a worst-case scenario that comprises not only the encounter geometry but systems behavior as well. Based on the ICAO. Annex 2 (Rules of Air) a head-on is considered a high-risk situation and both aircraft should diverge from the original flight track to the right until a safe separation minimum is achieved. However, in our assumptions of the worst-case scenario, the systems behavior is taken in consideration as well. In other words, despite that we assume vehicle to vehicle communication is available, not all the sUAS can do a conflict resolution maneuver (i.e., not equipped with a DAA system). Another practical situation is considered for non-conforming sUAS as described here in [132]. In this manuscript, we use DAIDALUS as DAA reference method, which uses a linear state-based approach to detect and resolve conflicts. The results of a state-based predictions are only valid for the time that the state of the involved sUAS does not change (i.e., it behaves linearly within the look-ahead time). In case of maneuvering intruders, the DAA performance would not be acceptable due to a relatively high number of false positives (predicted loss of separations that will not actually occur). However, this is not true for a cooperative ecosystem (i.e., continuous exchange of sUAS state space). If the sUAS intent information is available, state-based prediction performs better, and the false positives are filtered out [133–135]. The only remaining issue would be the uncertainties in communication delays, which could effect the intent information. Our model does consider these delays for the definition of the Well Clear separation minima, which can be thought as an added safety buffer to the Well Clear area. Therefore, theoretically speaking, if we would consider a maneuvering intruder, we expect that the change on the results would be very likely insignificant compared to the current results.

The sUAS parameters are based on a DJI Inspire 2 Quadcopter[3]. Its characteristics are summarized in Table 5.1. While our experiments are based on the DJI Inspire 2 characteristics, our model is generic and can be updated according to different parameters. For instance, if we would use a sUAS with lower performance like DJI MAVIC3[4], ground speed and turning rate parameters would be changing accordingly, and therefore the safe separation boundaries around the sUAS. An illustrative case scenario is given in the discussion section.

Weinert et al. [26] have pointed out that often the advertised maximum and/or cruise airspeed normally do not match with the real-life achievable sUAS airspeed. For this reason, we alternate both sUAS ground speeds, by limiting the closure rate, $max(CR) \leqq 35m/s$. The time parameter values regarding the systems behavior (DAA, Autopilot and Sensors Update Rate) are based on literature review [131, 136, 137]. It is very common that for sUAS having onboard decision making, the processing time is neglected, since it is typically less than 1 second. However, given the fact we assume a worst-case modeling, $t_{DAA} + t_{ap}$ is considered as 1 second. Regarding the sensors update rate, we

---

[3] https://www.dji.com/nl/inspire-2

[4] https://www.dji.com/es/mavic-3/specs

Table 5.1: Characteristics of DJI Inspire 2 Quadcopter

| Characteristics | Values |
|---|---|
| Dimensions | 60.5 cm |
| Maximum gross take-off weight | 4 kg |
| Maximum flight time/endurance | 27 minutes |
| Maximum airspeed | 26 m/s |
| Maximum altitude | 2500 m ASL (Above Sea Level) |
| Maximum pitch | 90 °/s |
| Maximum yaw | 90 °/s |
| Maximum roll | 90 °/s |

have assumed that both sUAS are equipped with ADS-B like type of sensors and take the maximum value of the triangular distribution(i.e. 2 seconds) as described in the section V.A. Furthermore, we suppose that while maneuvering the ownship, sUAS can perform a heading change with maximum turn rate $\omega_m \in [30°/s, 45°/s, 60°/s, 90°/s]$ and fly straight at level flight for at least 1 second, $min(t_2) = 1s$. Given the aforementioned assumptions, $t$ will be only dependant on turning rate. The formal definition is given in 5.8. To determine an upper limit for $t_2$ a systematic evaluation was done based on the severity level. Preliminary results stated that a $t_2 > 3$ seconds has little or no effect on the system's behavior.

$$\begin{cases} t_{TRT} = max(t_{sens}) + max(t_{DAA} + t_{ap}) \\ f(t_1) = t_1 + t_2 + t_{TRT} \end{cases} \tag{5.8}$$

, where $t_{TRT} = 3s$ and $t_2 \in [1, 2, 3]s$.

### 5.2.3 Scenario Generation

We define a scenario as a particular ownship-intruder scripted encounter. In order to create a comprehensive set of scenarios, we formalize a scenario configuration as a tuple $(V, \Omega, D, \alpha, t_{al})$, where $V$ is speed, $\Omega$ is turning rate, $D$ is the WC threshold and $\alpha$ is the encounter geometry. The intruder is generated based on the particular configuration of the ownship. More specifically, the initial position of the intruder is calculated by the relative range and bearing to ownship, where those values are in turn calculated by the angle $\alpha$ and time to loss of WC, held always constant at 15 sec. The time to LoWC is set relatively small in order to induce situations that are more likely to require WC recovery maneuvers. Generally, to quantify safety in the context of aviation, conservative approaches are followed. In our case, following a conservative approach means analyzing the worst-case encounters during short-time windows (i.e., less than 15 seconds). Since no other geometry can be riskier than the head-on encounter, we base our analysis on that. Moreover, the short-time windows comply with the requirements on communication and surveillance systems. Similar approaches

Table 5.2: sUAS encounter matrix

| Parameter Type | # | Values |
|---|---|---|
| Ownship ground speed | 2 | 10 m/s, 20 m/s |
| Intruder ground speed | 2 | 10 m/s, 15 m/s |
| Encounter geometry | 3 | Head-on, Crossing, Over-taking |
| Maximum turning rate | 4 | 30°/s, 45°/s, 60°/s, 90°/s |
| Flying straight time $t_2$ | 3 | 1s, 2s, 3s |
| Look-ahead time $T$ | 1 | 10 s |
| Alerting time $t_{al}$ | 3 | $t$, $t+1$, $t+2$ |

are followed in various research works [138] [139] [140], that are used to evaluate DAA methods, system requirements and separation thresholds.

A total of 216 scenarios are generated by combining sUAS and encounter parameters as documented in Table 5.2.

### 5.2.4 Metrics

Two metrics are analyzed in this study:

1. Maximum Severity of Loss of Well Clear [107].This metric captures LoWC events and gives information about the proximity between the sUAS per each encounter. In this context, a score of 0 means aircraft remained WC and a score of 1 a mid-air collision has occurred. A low separation severity is preferred. Formally, it is expressed as:

$$S_{max} = max(0, 1 - \frac{d_{WC} - d(t)}{d_{WC}}) \tag{5.9}$$

   where, $d_{WC}$ is the well clear minimum separation distance and $d(t)$ is the distance between the ownship and the intruder at time $t$. Low values of $S_{max}$ indicate that sUAS DAA system is more successful in regaining WC and preventing NMAC situations.

2. Average time between the time of LoWC and WC recovery time, denoted as $T_{\neg WC}$. This metric is utilized for operational reasons, in which $T_{\neg WC}$ shouldn't be too large, since a loss of WC represents a risky situation and sUAS has a limited time to regain WC. It is assumed that this time should be approximately less or equal to the maneuver time [141].

## 5.3 Simulation Results

In this section, Friedman's test [142] has conducted to analyze the impact of the parameters defining the WC threshold such as closure rate, encounter geometry and environment uncertainty (e.g. wind), on LoWC severity. We utilize an alpha level 0.05 to show that the results are statistically significant.

To concisely present the results, we provide bar plots showing the maximum mean values considering only critical scenario sets. In the next subsection, we explain what we consider critical scenarios and how they serve best to the scope of this work. Moreover, each bar plot is associated with a error bar, to give better comprehension of the results. Focused analysis on the specific scenario sets is provided in subsections 5.3.2 to 5.3.5. The authors base their discussion and recommendations on the outcomes of this analysis. Three data sets ( https://dx.doi.org/10.21227/0d10-nm73 ) are provided for the reader corresponding to analysis found on this manuscript for generating same results or for further investigations.

### 5.3.1 Data filtering: Critical Scenarios

In this thesis we follow a worst case analysis to quantify the Well Clear Area and determine adequate Warning Alert time-thresholds. Keeping this in mind, the preliminary results served as a filtering process, to select and further analyse scenarios that fit best to the scope of this chapter. In this regard, the following analysis focus only on critical scenario sets. Critical scenario set are considered the scenarios in the experiment, complying with the following constrains: 1) high risk encounter, i.e. high value of mean severity; 2) Sensitive towards influencing factors, e.g. WC threshold, Warning Alert time. Based on these two conditions, we exempt from further analysis overtaking scenarios and focus more into head-on encounters. Few exceptions are done. For instance, while showing the effect of the encounter geometry, we give a comparison between head-on and crossing scenarios. Note that while in the overtaking scenario the maximum severity tends to have high values (Fig 5.3) , it is more a matter of the self-separation method and experiment design, rather than a high risk situation. More specifically,overtaking cases have smaller WC threshold, $R_{WC} = R_t$ - $l$, and lower airspeed for ownship sUAS. This creates a long tail-chase situations, no matter the variance of parameters. For this reason, it is not considered as good indicator for our recommendations. However, we use the insight from the preliminary results for the general conclusions and the future work.

### 5.3.2 Evaluation of mean Severity

In this study, WC threshold and Alerting Logic objective is to prevent high risk situations, that might lead into a NMAC event. In this context, low mean severity values are preferred and any occurrence of NMAC would indicate a failure of our self-separation approach for sUAS-sUAS encounters. The simulation results demonstrated no such situations, verifying the model assumptions. The bar plot in Fig.5.4 illustrates the average maximum severity for head-on encounters and high maneuvering sUAS. The categorization on the maneuverability is based on the data presented in [26]. The bar plots are grouped by the values of $t_2$, where greater values imply larger WC threshold. In Fig.5.5, lower performance sUAS are shown. The lower values of severity compared to the previous plot, attributes to the fact that low performance sUAS have larger

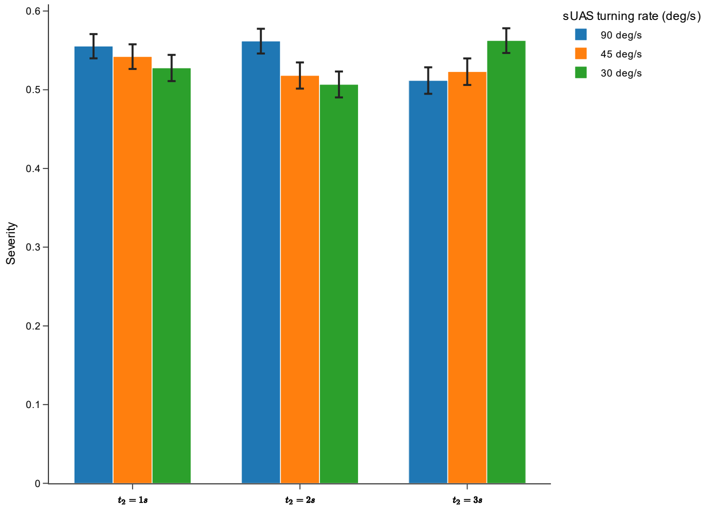


Figure 5.3: Mean Severity for Overtaking scenarios

WC thresholds. The encounters with $t_2 = 1$ s, experience the highest average $S_{max}$. The minimum values of $S_{max}$, are obtained for $t_2 = 3s$. Among the parameters included for the statistical analysis, it is observed that the results are statistically significant, with p-value, $p < 0.05$ and standard error, $SE < 0.02$.

It is worth noticing that higher performance sUAS and head-on encounters have higher difference in $S_{max}$ values, thus are used in the next subsections to see the effects of the warning alert times, encounter geometry and closure rate on the maximum severity metric.

### 5.3.3 Evaluating Warning Alert Time-Thresholds

In this subsection we attempt to analyze the effects of Warning Alert time-thresholds on the average severity behavior. In Fig.5.6, it can be seen that for larger Warning Alert time-thresholds, average $S_{max}$ is lowered. For instance, for $t_{al} = t = 5$ s (sUAS has a TR = 90 deg/s and $t_2 = 1s$, the red bar shows the mean value $S_{max}$=0.51 (p¡0.05,SE¡0.01); for $t_{al} = t + 1 = 6$ s, this value drops to 0.35($p < 0.05$,$SE < 0.01$); whereas for $t_{al} = t + 2 = 7$ s, mean value of $S_{max}$=0.24($p < 0.05$,$SE < 0.01$). During result analysis was noted that the Warning Alert times reduces the severity significantly when we increase alerting time with 1 second. In the scenarios when alerting time is increased 2 s, the changes in severity are smaller and not that significant as for 1 second increment. This especially noticed in the encounters with lower turning rates. In Fig.5.7, the performance of Warning Alert times on low maneuverable sUAS is shown. Since those sUAS have larger WC thresholds, they have lower severity and as such, the impact of warning alert times is even smaller. For the group with TR = 30 deg/s, increasing $t_{al}$ with 2 seconds (yellow bar), has an insignificant

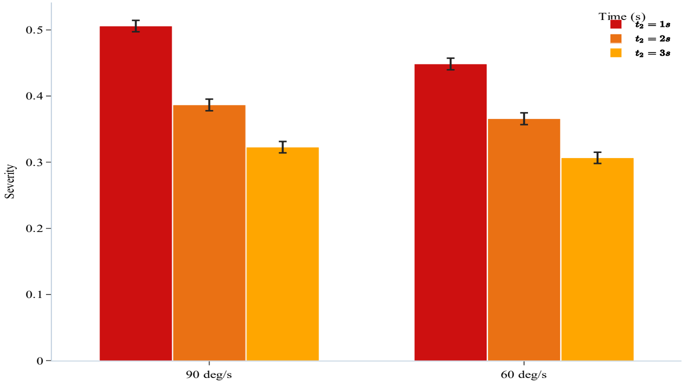


Figure 5.4: Mean Severity for high maneuverable sUAS

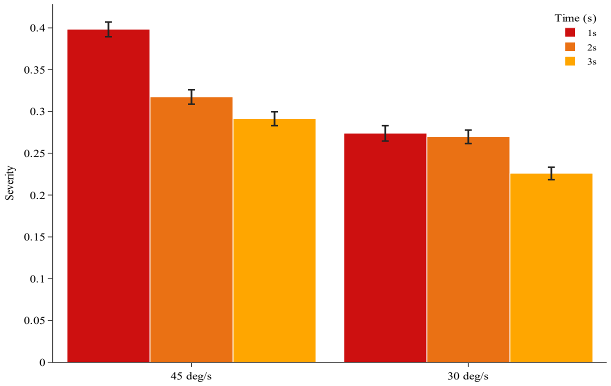


Figure 5.5: Mean Severity for low maneuverable sUAS

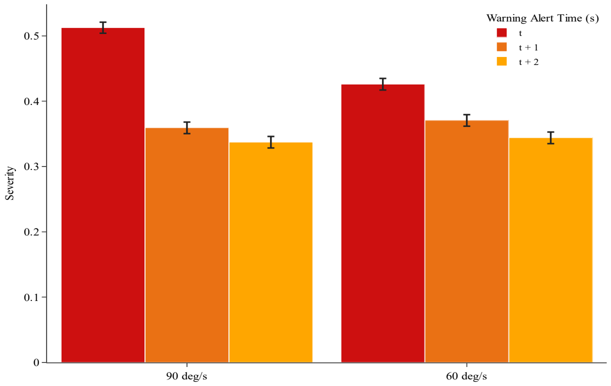


Figure 5.6: Mean Severity for different $t_{al}$ values

change on severity compared to $t_{al}$ + 1 seconds.

### 5.3.4 Encounter Geometry, Closure Rate and Uncertainties

**Influence of Encounter Geometry**

To see the encounter geometry affects, the evaluation of mean severity was studied with respect to minimum and maximum WC threshold. In the Fig.5.8, mean $S_{max}$ is shown for two combinations of the parameters. The first group shows a head-on scenario (the darker color) and crossing scenario (the lighter color) with respect to minimum WC threshold $R_{WC}$, i.e. $t = 5$s. In the second group, the same parameters are computed with respect to maximum WC threshold, i.e. $t = 7$s. The ground speed (GS) of sUAS and turning rate (TR) are kept constant, with ownship GS = 20 m/s, intruder GS = 15 ms/ and TR = 90 deg/s.

We observe that for minimum WC threshold, the encounter geometry influences severity significantly; a value of $S_{max} = 0.5$ ($p < 0.05$,$SE < 0.01$) in head-on encounter is reduced to $S_{max} = 0.22$ ($p < 0.05$,$SE < 0.01$) for crossing geometry. In the other hand, for maximum WC threshold the difference can be considered neglectable with a difference in severity of 0.01. This is attributed to the fact that large WC thresholds create low risk situations, and are less sensitive toward different factors.

**Influence of speed**

To understand the impact of the ownship sUAS speed and relative speed during the encounter. Since the WC area around the ownship depends on the sUAS performance, it results in smaller thresholds for lower performances. Due to this fact, sUAS with low speeds are expected to have higher severity. As it can be seen in Fig.5.9, $S_{max}$ has peak values for encounters in which ownship has minimum speed (here, 10 m/s) and maximum intruder speed (here, 15 m/s). This is attributed to self-separation model, which does not take in consideration the intruder sUAS speed in an explicit way. In our approach we make use of the warning alert time-thresholds to reduce the risk in such scenarios. Fig. 5.10, illustrates that for intruders with maximum velocity (here, 15 m/s), larger warning alert time-thresholds reduce mean $S_{max}$ value. It is worth noticing that low ownship sUAS speed has more impact on the severity, rather than high relative speed. For instance, in Fig.5.9, the case of ownship GS = 10 m/s and relative speed 20 m/s, has higher severity than when GS=20 m/s and relative speed is 30 m/s.

**Influence of Communication Delay and Wind**

For this analysis, a critical scenario with minimum WC threshold ($t = 5$ s), head-on encounter, ownship GS=10m/s and intruder GS=15 m/s. The results are displayed in Fig.5.11. In the first run of simulation, both parameter values were assigned to 0, to create a deterministic scenario (light orange bar). The scenario was run only once, and $S_{max}$ scored a value of 0.26 ($SE = 0$). Then in the environment we added communication uncertainty (see section V.A). In this case, 1000 runs were done and the red bar (delay) shows the severity value. Lastly the same procedure was done to evaluate the impact of wind. It is evident that the most influencing factor is the delay in communication system, with a value of $S_{max} = 0.73$ ($SE < 0.01$). This is attributed to the fact that DAIDALUS utilizes a deterministic model to project future states of sUAS. Therefore, it requires subsequent, in-time state information (e.g. position, speed), to accurately predict LoWC states and generate WCR maneuvers. Wind as well can influence severity$S_{max} = 0.65$ ($SE < 0.01$), but once the data about wind is provided to the DAA system, the state estimation can be done by considering airspeed instead of ground speed. Consequently reducing the error of prediction.

### 5.3.5 Average time between LoWC and WCR time

Figure 5.12 shows an overview of mean time that sUAS spend in LoWC or the time it was not well clear $T_{\neg WC}$, with the intruder sUAS. We illustrate different combination of turning rate and ground speed, to have a better insight on this value. We observed that for all the scenarios, $T_{\neg WC}$ is less then minimum turning maneuver, indicating good initial assumptions for our model time input parameters. The maximum value, $T_{\neg WC} = 3.86$ seconds ($SE = 0.25$), is reached for low performance sUAS, such that GS=10 m/s and TR=30 deg/s. This is an expected result, given the fact that low performance sUAS need more time to perform a WCR maneuver.

## 5.4 Discussion

The findings of our study suggest that a sUAS performance based Well Clear standard can be safe and efficient for sUAS operations. The described methodology is a function of sUAS types, UTM capabilities and environmental uncertainties.To the best of our knowledge, the paradigm of sUAS ecosystems is different from the standard aviation and requires a system's thinking approach. In other words, we think that each component performance is directly measurable and can be quantified with statistical significance (comprising Aleatoric and Epistemic uncertainty). One may follow a worst-case analysis to model each composing system or develop a probabilistic model. The process underlies the same principles to determine a time threshold, which can be translated into spatial separation thresholds. These thresholds are dynamic with respect to the sUAS performance and environment, which contributes to a better management of airspace capacity.

In doing so, a better understanding of each system and their effects on the overall behavior can be studied. We think this is an important consideration, since there is a lack of "experience" in UAS operations and especially in case of sUAS. Therefore, a self-separation standard which is less dependent on expert assumptions or arbitrary choices, can lead into compelling and complementary outcomes.

The results in 5.3.2. indicate no occurrence of NMAC and recovery of WC status in a timely manner. For sUAS with high maneuverability and in head-on encounters the mean severity is the highest for $t_2 = 1$ seconds ($S_{max} = 0.507$, $SE = 0.008$). In this scenario the method requires the minimum possible WCA, and further improvement can be considered. One solution, is to increase $t_2 = 3$ seconds, which would result in bigger WCA, and lower severity into ($S_{max} = 0.32$, $SE = 0.008$).

The results in 5.3.3 show how the severity can be lowered by changing the Warning Alert Thresholds. For the same scenario described in the paragraph above, increasing alerting time with 1 second reduces mean severity from ($S_{max} = 0.507$, $SE$=0.008) to ($S_{max} = 0.357$, $SE$=0.008); and if alerting time is increased with 2 seconds, we have a better performance ($S_{max} = 0.337$, $SE = 0.008$). However as noted in the Results section, the change is not that significant. This might be attributed to the fact that we have a constant sensor update rate during the encounter and DAA has no use of early situation awareness to provide a recovery maneuver. In our opinion, larger alerting time thresholds would be more robust in a Remain WC event.

The results in 5.3.4 show the sensitivity of the severity with respect to encounter geometry, closure rate and uncertainties i.e. communication delay and wind. It was indicated that severity is effected highly from the encounter geometry and communication delay. Two main points can be inferred from this analysis. Firstly, if a self-separation performs well in a head-on scenario, it is highly likely to perform at least as good in crossing encounters, under the same conditions. Secondly, once the intent information is available in encountering scenarios, communication delays have dominant effects on safe separation assurance.

## 5.5 Recommendations

In this section, we give recommendations related to adequate WC thresholds and Warning Alert Times. In subsection 5.5.1 we explain the reasoning behind our recommendations and give numerical values to quantify WCA and Warning Alert Threshold. Furthermore, in section 5.5.2 a hypothetical use case is given to illustrate how these recommendations can be used in a practical way.

### 5.5.1 Well Clear Area and Warning Alert Thresholds

The focus of this study is in airborne safety and the use of Well Clear standard to assure safe separation among sUAS encounters. Final recommendations based on this study consider the following principles:

- Group the sUAS based on their ground speed and maneuverability similar to [143]. We group sUAS into *high-maneuverable sUAS* when their turning rate is greater or equal to 60 deg/s and *fast sUAS* when their ground speed greater or equal to 15 m/s. The rest is considered as *slow sUAS* and *low maneuverable sUAS*.

- Select combination of parameters (i.e. $t_1$ and $t_2$ that have the lowest severity.
- Evaluate operation suitability, (i.e. average time between LoWC and WCR time should be less than 5 seconds)
- Approximate the value to be multiple of 5, as it is common for use in aviation standards [122]

In Table 5.3 recommendations for *high-manoeuvrable sUAS* are shown. The data corresponds to sUAS with turning rate 60 deg/s and severity level less than 0.5. Note that, for sUAS with 90 deg/s the threshold can be smaller, while keeping the same value of severity. However, to avoid other unforeseen uncertainties in the systems, and considering that a WC maneuver tends not to be as sharp as a CA maneuver, a 60 deg/s maneuver is the best fit.

Table 5.3: Recommendations for high maneuverable sUAS.

| | WC threshold (m) | | Warning Alert Time (s) | |
|---|---|---|---|---|
| | Fast Intruder | Slow Intruder | Fast Intruder | Slow Intruder |
| Fast ownship sUAS | 155 | 115 | 8 | 7 |
| Slow ownship sUAS | 70 | 60 | 8 | 7 |

In the table 5.4 we present recommendation values for low maneuverable sUAS. We follow the same previous reasoning, and extract the data from sUAS with 30 deg/s turning rate. These criteria are compatible for sUAS with a turning rate up to 45 deg/s as shown by the results.

Table 5.4: Recommendations for low maneuverable sUAS.

| | WC threshold (m) | | Warning Alert Time (s) | |
|---|---|---|---|---|
| | Fast Intruder | Slow Intruder | Fast Intruder | Slow Intruder |
| Fast ownship sUAS | 165 | 145 | 8 | 8 |
| Slow ownship sUAS | 85 | 75 | 9 | 8 |

Regarding Warning Alert time-thresholds, based on the results analysis, it was observed that an alert value $t_{al} = t + 1$ s is the suitable case, considering that larger thresholds can cause false positive alerts and effect the performance [111]. The only exception in our recommendations, was the case of slow ownship and fast intruder for low maneuverable sUAS. In-there, $t_{al} = t + 2$ s compensates the relatively shorter WC threshold (75 m), to maintain a low level of mean severity.

### 5.5.2 Use Case scenario

In this hypothetical scenario, we assume that a delivery company similar to Uber Eats, has received an order for delivery. The company has several sUAS types in their fleet and for this particular case, is going to use a DJI Phantom 4 Quad-copter. The characteristics of sUAS are shown in Table 5.5 and all the sUAS are equipped with a DAIDALUS like DAA system.

Table 5.5: Characteristics of DJI Phantom 4 Quadcopter

| Characteristics | Values |
|---|---|
| Dimensions | 30.5 cm |
| Maximum gross take-off weight | 1.4 kg |
| Maximum flight time/endurance | 27 minutes |
| Maximum airspeed | 20 m/s |
| Maximum altitude | 2500 m ASL (Above Sea Level) |
| Maximum pitch | 45 °/s |
| Maximum yaw | 45°/s |
| Maximum roll | 45 °/s |

Remote pilot (RP) has access to an UTM like ecosystem and before he starts the mission, a flight plan, together with DAA Well Clear parameters need to be uploaded. A simple process can be as following:

- Generate a flight plan for sUAS to autonomously go from point A to B.
- Specify sUAS nominal ground speed and flight level. In this case, we assume it is 10 m/s and 250 ft AGL as in [144]
- Specify sUAS nominal turning rate. In this case we assume 30 deg/s.
- Share the information with the UTM to establish an updated situation awareness.
- Define WCA and Warning Alert Thresholds.

Given the fact that Uber Eats ownship sUAS belongs to the category of *low-maneuverable* and *slow sUAS* the thresholds should be taken from Table 5.4. Based on the situational awareness of surrounding traffic the DAA system should select the thresholds corresponding to Fast Intruders or Slow Intruders. A conservative case would be assuming that all the time there are Fast Intruders. Therefore, the RP should update the parameters of the DAA system with a WCA of 85 meters and Warning Alert Time of 9 seconds. In this condition, the mission should proceed safely in an autonomous way. In case that there is a demand of airspace capacity, and the nominal speed of sUAS falls into the category of *slow sUAS*, less conservative thresholds might be used. For instance, 75 meters WCA and 8 seconds Warning Alert Time.

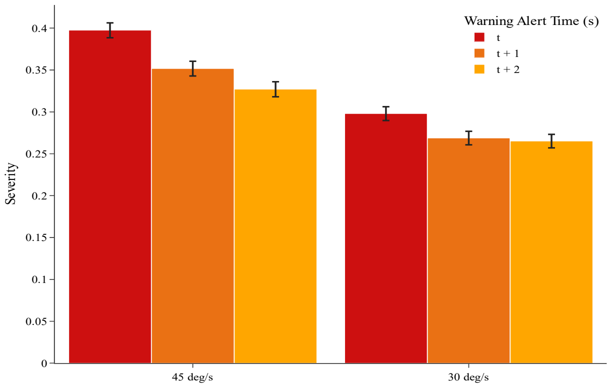


Figure 5.7: Mean Severity for different $t_{al}$ values

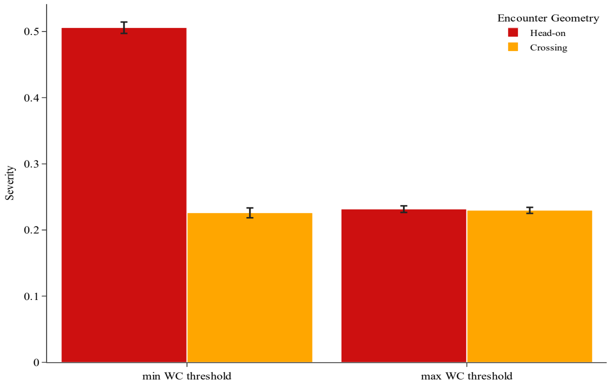


Figure 5.8: Influence of Encounter Geometry on Severity

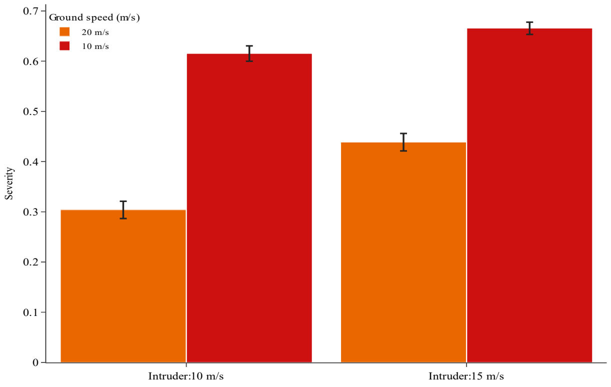


Figure 5.9: Influence of Speed on Severity

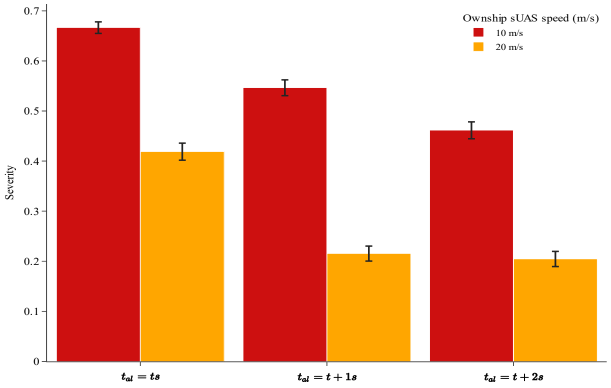


Figure 5.10: Influence of Speed on Severity for different $t_{al}$

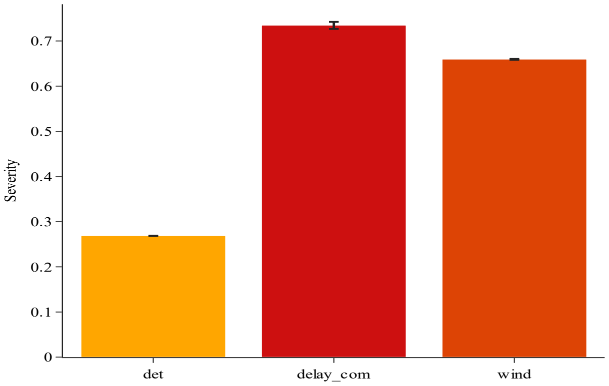


Figure 5.11: Influence of Comm. Delay and Wind on Severity

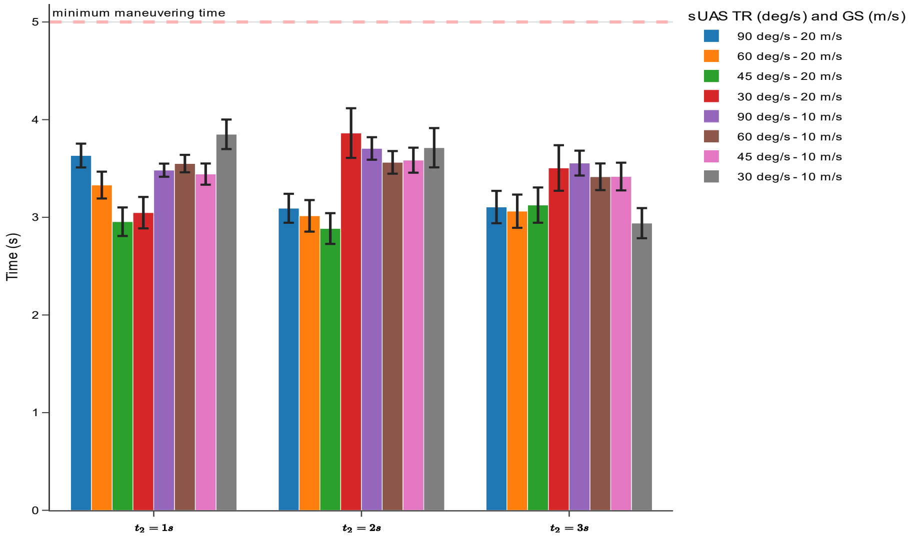


Figure 5.12: Average time in LoWC

# Chapter 6

# Conclusions & Future Work

In this chapter we first give a brief overview of this thesis and then list the main contributions. Lastly, some topics on how to extend or use this work, is given in the form of the future work.

## 6.1 Concluding Remarks

This research intended to overcome some of the barriers related to the safe integration of sUAS into civil airspace. Safety is the primary concern when it comes to air traffic. At the tactical level, the goal is to quantify en-route safety through minimal pairwise separation between sUAS. In doing so, significant system level changes should be designed, developed, and implemented. An extensive literature on TD systems, showed that despite the vast amount of work being done, there is a lack of a standard model that meets the requirements and complies with specific regulations. Moreover, there is a lack of shared knowledge and understanding, a presence of outdated terminology, disjointed concepts, lack of context and relevance and in general an ambiguity on a holistic view of defining the problems and solutions. In general, studies tend to focus on specific elements of TD and not evaluating the whole as a system. To bridge this gap and to remedy the misconception throughout literature, we proposed a framework based on SE principles, by providing a systemic analysis – studying constituent elements and the whole system – which is used to understand and evaluate systems behavior. Following this line of thought, some remarks were done, to remedy misconception by the academic and industrial community:

- TD should not be considered a reactive system, despite the fact some constituent elements can have reactive behavior.
- See and Avoid (SAA) is part of CA, and as such should not perform a SS function.
- DAA should not be considered a derivation of SAA. DAA are tightly coupled with the concept of decision support systems, widely used in aviation.

- The concept of *potential conflict* should be avoided. There can be only potential LoS with a high look-ahead time.
- DAA should be considered analogous with TD and should span both SP – SS included –, and CA.
- DAA systems should be analyzed systemically when evaluated, in the sense that CA should not be considered an independent system.

## 6.2 Contributions

This thesis has done the following contributions:

- Identification of challenges facing the design and development of TD systems
- Formal definition of TD Systems, through identification of constituent elements/subsystems and their corresponding functions/capabilities.
- Systemic analysis of current TD systems – understanding structure and behavior (e.g. reactive and emergent behavior).
- Developing a framework based on systems engineering principles to tackle the identified challenges (i.e., issues).
- Proposal of TD model by providing a logical system architecture
- Proposal of self-separation minima for sUAS operations

## 6.3 Future Works

There is a lot of work left to be done in the domain of conflict management and UTM systems. For the purpose of this thesis, we will structure the future work in three main topics. First, future work can be conducted on extending this work based on the gaps and limitations presented here. Then, there is an immediate need to understand information systems within the context of CM and UTM in general. Lastly, given the role that artificial intelligence (AI) is taking in aviation industry, a rigorous understanding and consequences (pros and cons) should be further investigated.

## 6.4 Gaps and Limitations

In our work we do various assumptions to facilitate the process and overall provide high level concepts. To further improve this work, the reader can take the following steps:

- The involvement of human as agent in the decision making process. While we mention that UAS can be remotely controlled, we remove the human factor from our problem. Further investigations can be done in modelling human behavior and analyzing the impacts on the systems behavior.
- Improve Encounter Generation model. In this work we used pair-wise only scenarios, with an intruder aircraft flying in straight line. This was done under the assumption that the intent of all aircraft was known during the flight duration. This approximation might not be true in all scenarios, given that sUAS mission profiles are irregular and off-nominal situation can occur.
- Test and evaluate another DAA reference system. In this work we used DAIDALUS as reference DAA system, given its capabilities and the flexibility to be tuned in to our needs. Nonetheless, to verify the proposed model is consistent and reliable, there is a need of evaluating different DAA systems.
- Validate our model via hardware in the loop and flight test scenarios. The ultimate goal of this work would be to analyzed the behavior of the model while performing real test scenarios. This would demonstrate its performance, but also evident hidden gaps and limitations.

### 6.4.1 Information Systems

Information systems are systems that are designed to collect, process, store, and disseminate information. These systems are key component of UTM and CM systems that are not taking the required importance yet. While there is work done regarding the communication systems, there is very limited knowledge on how the required information should be modelled, be reliable and secure. The very concept of UTM lies with the fact that it should be a set of digitized services, which should exchange information continuously, in order to coordinate and collaborate. Given information systems architectures (e.g. internet) are not adequate for safety critical UAS operations. In addition, information systems can be hacked and hijacked, consequences of which can be a lot more severe than hacking an internet network.

### 6.4.2 Artificial Intelligence (AI)

There is no doubt on the potential that AI has in aviation industry. Nevertheless, safety-critical domains such as aviation, needs high safety requirements as system failure could result in a loss of life or high cost of material losses. Despite its success, AI is not completely reliable as it is inherently non-deterministic. Furthermore, AI systems are often considered as “black boxes”, meaning that it is not easy to understand how it works and why it produces certain outputs.

It is argued that in order to increase the trust in AI models it is crucial to understand why it makes the decisions it does make. Furthermore, it is crucial that these explanations are tailored to the human who must interact with them. Additionally, these characteristics are important in understanding the system retrospectively, e.g., to understand a wrong or even harm-causing decision and proactively e.g., to predict and prevent any future harm-causing decisions.

The idea that ”AI is the second coming of Jesus” and will solve all the problems in integrating safe UAS operations, should be carefully considered.

# Appendices

# Appendix A

# The Effects of Encounter Geometries on the Separations Metrics for sUAS

In this chapter we will provide an extensive study of the effects that encounter geometries have on separation metrics and methods. This study, assumes that ownship sUAS has right of way and turns always right when a conflict is detected. As shown in the results, this typical behavior lowers the performance of system in some particular cases.

## A.1 Modeling a sUAS

sUAS are characterized by a Maximum Gross Take Off Weight (MGTOW) not bigger than 55 lbs (approx. 25 kg) and a Mean Cruise Airspeed not higher than to 60 kt (approx. 30 m/s).
Our onwship model is based on a DJI Inspire 2 Quadcopter[1]. Its characteristics are summarized in Table A.1. The intruder sUAV model is based on the latest Amazon delivery drone and its characteristics are summarized in Table A.2.

Weinert et al. [26] have pointed out that quite often the advertised maximum and/or cruise airspeed normally do not match with the real-life achievable sUAS airspeeds. For this reason we choose a cruising airspeed of 15m/s for our ownship. Nonetheless, since there are no publications related to the recent Amazon delivery drone, we decided to take the stated maximum air speed 30m/s for the intruder.

---

[1] https://www.dji.com/nl/inspire-2

Table A.1: Characteristics of DJI Inspire 2 Quadcopter.

| | |
|---|---|
| Dimensions | 60.5 cm |
| Maximum gross take-off weight | 4 kg |
| Maximum flight time/endurance | 27 minutes |
| Maximum airspeed | 26 m/s |
| Maximum altitude | 2500 m ASL (Above Sea Level) |
| Maximum pitch | 90 deg/sec |
| Maximum yaw | 90 deg/sec |
| Maximum roll | 90 deg/sec |

Table A.2: Characteristics of the Amazon Delivery Drone.

| | |
|---|---|
| Dimensions | 2.15 m |
| Maximum gross take-off weight | 40 kg |
| Maximum flight time/endurance | 15 minutes |
| Maximum airspeed | 30 m/s |
| Maximum altitude | 365 m |

Moreover, while the angular velocities of fixed-wing sUAS can be given analytically, the same does not hold true for multi-rotor type sUAs. The approximation we used that performed well for our ownship is a yaw rate of 45 deg/s. Given we have used a point-mass model for the sUAS, pitch and roll are not relevant.

## A.2 Separation Minima

We define the Well-Clear volume as in the MIT LL recommendation [26]. There, the authors propose a cylindrical volume for a sUAV. The radius of the cylinder (i.e. minimum horizontal distance) is 2000ft, while its height (i.e. minimum vertical distance) is 50ft.

For the NMAC volume, we follow a similar modeling approach to [29]. Firstly, the Mid Air Collision (MAC) volume is defined as the smallest cylinder that fully contains the sUAV. Then, the radius of NMAC is defined based on a time condition. Specifically, the radius of NMAC, $r_{NAMC}$, will be:

$$r_{NMAC} = 1.1(r_{MAC} + r_{rd}) \tag{A.1}$$

where $r_{MAC}$ is the MAC radius, and $r_{rd}$ is the relative distance that the sUAV needs to cover to cause a $MAC$. While calculating $r_{rd}$, we assume a head-on encounter and a traveling time of 1 sec.

In our scenarios, the diagonal distance of the ownship is 60.5cm [2]. Based on this, we choose a MAC radius of 35cm. Furthermore, the speed of the ownship is

---

[2] Propellers are not accounted.

15m/s and the cruising speed of the intruder is 30m/s. This results in a relative speed of 45m/s. Considering also a 10% safety buffer, we choose a NMAC radius of 50 m for this case.

## A.3 Metrics

There are three measures we use throughout the simulations. The first one counts the amount of time the pair of sUAS are not in well clear. The second checks the closest distance to the NMAC separation. Lastly, we measure the severity of the LoWC [29]. Differently from there, we define the severity as a quantity evolving over time. This is done in an attempt to make severity more informative. Formally, the severity is:

$$severity(t) = \frac{d_{WC} - d(t)}{d_{WC}} \tag{A.2}$$

where, $d_{WC}$ is the well clear minimum separation distance and $d(t)$ is the distance between the ownship and the intruder at time $t$.

## A.4 Modeling and Simulation

For modeling and simulation purposes, we used the open source ICAROUS[3] platform developed by NASA. It allows for building safety-centric UAS application. Furthermore, ICAROUS integrates DAIDALUS, which is our DAA system of choice.

In this work, we simulate ownship and intruder trajectories that intrude on angles from 0° to 180°, with a 10° increment. The intrusion angle is measured at the moment that the two vehicles loose their well clear separation. We take into consideration position uncertainty, which is represented as Gaussian noise with 0 mean and standard deviation of 2.4m for each dimension[4]. We simulate 100 trajectories for each intrusion angle and present aggregated results.

### A.4.1 Loss of Well Clear

In this section we illustrate how the angle of intrusion affects the time in which the ownship and intruder are not well clear. Figure A.1 shows the time spent in LoWC for each angle.

Results are the average of all the simulations per angle. As we can see, there are two general trends. The amount of time increases from 0°(head-on) to 90°, with the maximum being at 80°. The same happens from 90° to 180°. Such a behavior can be attributed to the heading of each vehicle. This can be made clear in the first and last angles. In the head-on case, the UAS will be

---

[3] https://github.com/nasa/icarous

[4] https://www.gsa.europa.eu/sites/default/files/uploads/drones_operations_whitepaper.pdf

flying in opposite directions when the DAA restores the necessary separation. This causes the separation to be restored quite fast, with LoWC only being around 30 seconds. In the case overtake case (180°), the opposite happens. The two UAS are moving in the same direction, Once a maneuver is issued by the DAA, the UAS will naturally take longer to restore the minimum separation. An additional support is the decrease of LoWC at 100°, at which the UAV start moving faster away from each other, once the regulating maneuver is issued.

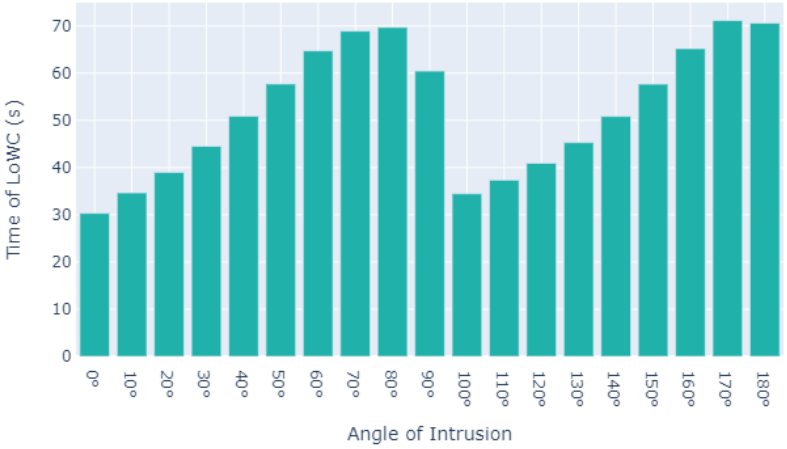


Figure A.1: Amount of time vehicles are not in well clear for each angle

### A.4.2 Severity of LoWC

We identify no cases where an NMAC infringement occurs, which indicates that the DAA is effective in restoring well clear. In Figure A.2 we show the closest the two UAS have been to reaching the NMAC distance. As we can see, the scenario with a 70°intruding angle is the one where the vehicles reach closest, with the distance being around 7m from NMAC (i.e. approximately 57m). We observe similar distances for the overtaking (and around the overtaking) cases. Under the given simulation configuration, where the intruder is faster than the ownship, cases where LoWC lasts longer, tend to cause more proximity between the UAS.

A higher severity level is also present for them, as illustrated by Figure A.3. We group angles in three groups. Group 1 consists of the head-on and close to head-on cases (0°-30°). Group 3 consists of the overtaking and cases around overtaking (150°-180°). Group 2 consists of the cases with the remaining angles

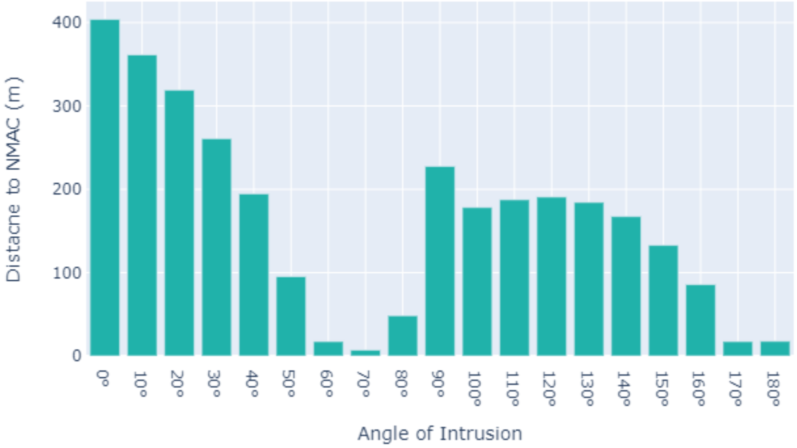


Figure A.2: Closest distance to NMAC in meters for each intrusion angle

(40°-140°). The figure contains the severity associated to a distance equal to the NMAC minimum separation distance, which is around 91.8%

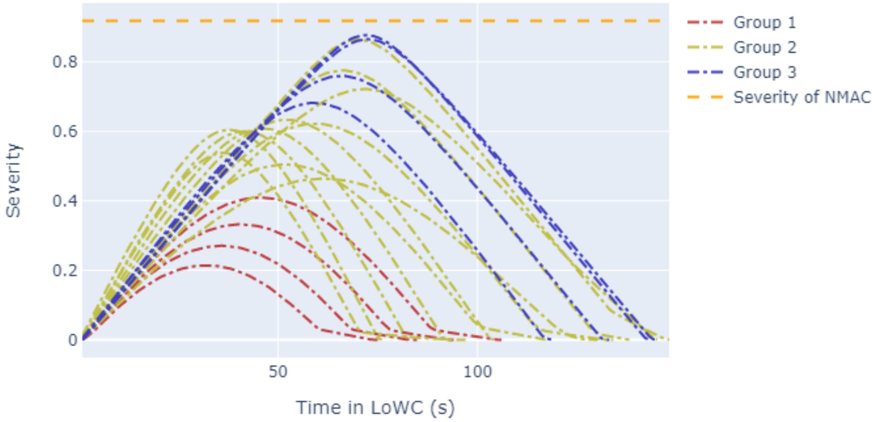


Figure A.3: Severity as a function of time for each intrusion angle

As we can see, intrusion angles in Group 1 have the lowest severity and the lowest LoWC time, while the opposite is true for intrusion angles in Group 3. Members of Group 2 have the biggest variation, a consequence of this group having more members. Nevertheless, angles that are not around 90° exhibit a similar behavior to Group 1 angles, while the angles around 90°exhibit a behavior similar to the Group 2 angles. Qualitatively, all severity plots have similar shapes. This is an expected consequence of the maneuverability constrains we have imposed (angular velocity of 45 deg/s).

## A.5 Conclusions

In this work, we investigate the effects of the intrusion angle on the time of loss of well clear for sUAS. We do so by simulating traffic with the intrusion angle ranging from 0° to 180°. We use DJI Inspire 2 as our ownship model and the latest Amazon delivery drone as our intruder model.

We observe an increase of time of LoWC with the increase of intrusion angle, up to a certain point ($\leq$90°). After that we observe a rapid decrease and an ensuing similar behavior ($\geq$100°). We believe this happens because of the heading of the vehicles in these case, with vehicles that have (close to) opposite headings having lower LoWC times. Furthermore, in our simulations we notice that intrusion angles with lower LoWC have lower severity.

These observations serve as evidence that the intrusion angle does indeed affect well clear. Therefore, we believe such an angle can be an important factor to consider when defining dynamic well clear metrics.

Furthermore, we define the severity of LoWC as a function of time, meaning that we calculate it for each timestep during which the two vehicles are not well clear. We believe this is an important contribution, as in this way severity is quite more informative than presenting a single value.

Such a definition can be useful in future research. We believe the shape that well clear is too rigid, and a parametrization of the shape according to our definition of severity can lead to a more efficient use of space. Moreover, given the vast variety of sUAS in terms of size and performance, it is important to investigate how our work here can be used in the characterization of dynamic safe volumes around sUAS . Finally, a limitation of our work is the narrow choice of sUAS type. This can be remedied in future studies, where more sUAS types, as well as a bigger variety of scenarios will be considered.

# Appendix B

# Reactive Tactical Deconfliction

In this chapter[1] we present a practical example of the use of an algorithm with reactive behaviour in the context of a separation provision (i.e., conflict detection and resolution). Our conceptual framework and the TD model lead us to believe that the overall performance of these kinds of systems should be quite poor. In order to achieve this goal, an investigation of applying Modified Voltage Potential (MVP) algorithm in high density sUAS operations has been conducted.

## B.1 Using Graph Theory to model Air Traffic

In this section an explanation on how the airspace and traffic have been adapted to the structure of a graph is given.

The airspace graph is defined at a certain time as an undirected weighed graph where each drone represents a vertex on it. The edges of the graph will represent interdependencies between UAS, more precisely, they will represent the *time to closest point of approach* (tCPA), which has already been defined in previous sections. We decided to use a time-base framework as most of the works on UAS has a time-based approach for conflict management. A threshold *thresh* will be defined so if the *tCPA* value between two aircraft is less than *thresh*, there will be an edge between them. The smaller the *tCPA* below the *thresh* value, the higher would be the edge weight value. Another threshold, *H*, is defined as the *tCPA* value which would result in the maximum weight value. If the *tCPA* is less than $H$ the weight of the edge would always be the maximum weight value. The weights will be normalized to be between 0 and 1. Their formal mathematical definition is,

---

[1] The method and results mentioned here, are part of a joint work with Javier Garcia Cañadillas and Ralvi Isufaj in the context of the former's master thesis

$$w_{i,j}(t) = \begin{cases} 1 \text{ if } tcpa_{i,j}(t) \leq H \\ 0 \text{ if } tcpa_{i,j}(t) \geq \text{ thresh} \\ \frac{\text{thresh } - tcpa_{i,j}(t)}{\text{thresh}} \text{ otherwise} \end{cases} \tag{B.1}$$

Where $tcpa_{i,j}(t)$ is the $tCPA$ value between aircrafts $i$ and $j$ at time $t$. The definition of the weights leaves the graph as an undirected graph as $tcpa_{i,j}(t) = tcpa_{j,i}(t)$. As the edges and their weights are defined at each time step, the graph is extended to the temporal domain, so we will be able to take into account information like the aircraft heading, as if two aircrafts are moving towards each other, the weight of the edge connecting them will increase over time.

Another usual way of computing the weights of the graph representing the air traffic is using the distance between aircraft instead of the tCPA. In Figure B.1 a comparison between the weights computing using formula (5) with tCPA and distance is shown. The situation is two aircraft going towards each other, beginning separated at a distance of 200 meters going at speeds of 40 m/s. The thresholds were chosen so both weighed were triggered in the beginning of the simulation and took their maximum values when the aircraft reach each other. In the computation of the weights, the absolute value of the tCPA was used. An advantage of using tCPA is that the value of the weights when decrease faster when the aircraft are moving away from each other than in the distance-based approach, as it can be shown in Figure B.1. This is an advantage of using tCPA for computing the weights because interdependencies should lose relevance when the aircraft are moving away as they can't no longer get in conflict with each other. Another benefit of using the tCPA is that it takes into consideration the speed of the aircraft, which is relevant to detect the severity of an intrusion and it is not contemplated by distance-based weights.

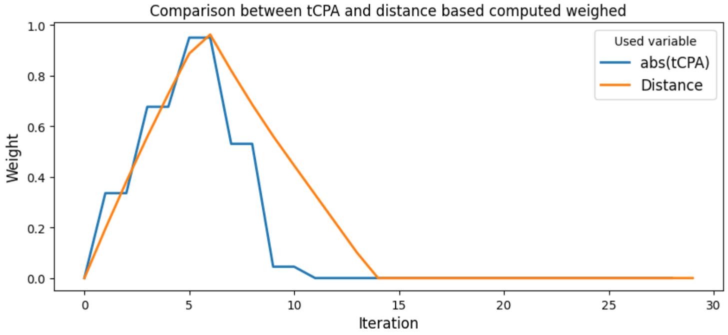


Figure B.1: Weights computed using tCPA and the distance between aircraft in a simulation consisting on two aircraft going towards each other. The velocities of the aircraft were 40 m/s, they were initially separated 200 meters. The thresholds for the tCPA weights were $thresh_{tcpa} = 2.5$ and $H_{tcpa} = 0$, and for the ones computed using the distance were $thresh_d = 200$ and $H_d = 0$

# B.2 Experimental Setup

## B.2.1 Simulation description

In this section, a description of the simulations carried out for the study is given. They were made in Python, using the air traffic simulator *BlueSky*. The results were extracted and store for further study and analysis using Python's library *Sacred*[2].

### Simulation steps

The scenario for the simulations is a circle with a radius of half a nautical mile of longitude. The detailed steps of the simulations are the following:

1. **Waypoints creation**: a waypoint is a specific geographical point on a flight route that the aircraft is required to pass. In the beginning of the simulation, three waypoints (blue dots in Figure B.2) are generated: One at the centre of the circle and the other two aligned with it, separated from the centre by 10% of the radius of the circle. The reason the waypoints are created is to force the aircraft to go through some common points so conflicts can be generated. If the UAS were generated with completely random headings, it would be very unlikely to observe any conflict.

2. **Airspace initialization**: a chosen number of drones are created with random positions inside the circle perimeter. Each aircraft trace a route with two points: the first one is one of the three created waypoints, chosen randomly, and the second one is one random point chosen from the circle perimeter. The aircraft first go towards the center of the circle where the waypoints are, and then return somewhere in the circle's perimeter. BlueSky directs aircraft through waypoints with the autopilot system LNAV, which stands for lateral navigation. The system measure the aircraft's position and compares it to the desired flight path. If the aircraft strays from the desired path, the system automatically makes corrections to keep the drone on track. The drones were generated with randomly selected speeds extracted from a uniform distribution between 10 m/s and 20 m/s.

3. **Main loop**: the airspace began to be simulated inside the main loop, where all the necessary computations and logs are made and the airspace get updated right before the next iteration. The duration of the simulation was chosen depending on the radius of the circle. It was chosen so the slowest aircraft (with a speed of 10 m/s) could manage to travel a distance equivalent to the diameter of the radius. For example, in a scenario with a 1 nautic mile radius, the simulation time would be $simt = 2 * \frac{1nm * 1852m/nm}{10m/s} = 370.4$ seconds. The used timestep $dt$ was 1 second.

---

[2] https://github.com/IDSIA/sacred

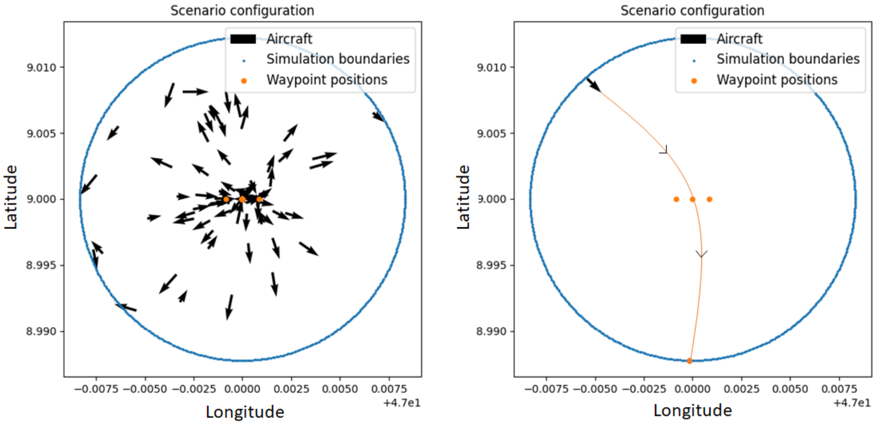


Figure B.2: In the first image of the airspace configuration in the middle of a simulation is shown. In the second one, an example of the path that an aircraft goes by in a simulation is displayed.

### Main loop

In this section, all the important operations carried out inside the simulation main loop are described. Figure B.3 shows an schematic flowchart of the hole process. In this part of the code BlueSky managed all the operations related to conflict detection and resolution and was also responsible of updating the state of the scenario after each iteration i.e. updating the positions and velocities of every aircraft based on their routes or the MVP instructions (if it is active). The rest of the computations were:

1. **Number of drones control**: to keep the number of aircraft constant, at every timestep a function checks if there is any aircraft beyond the circle limits. If so, this aircraft is erased and another one is randomly created inside the circle's perimeter in the same way that the original ones were created.
2. **Graph creation**: at each timestep, two different graphs were created using a `Graph` object from the *Networkx* library in Python. In both the nodes were the aircraft at that timestep and the difference was the way the edges were constructed:
    - *Complexity graph*: this is the main graph representing the airspace through which the complexity of the scenario is measured using the indicators described in section 2.3. The tCPA was computed using BlueSky conflict detection module and then formula (8) was apply to create the edges between nodes.

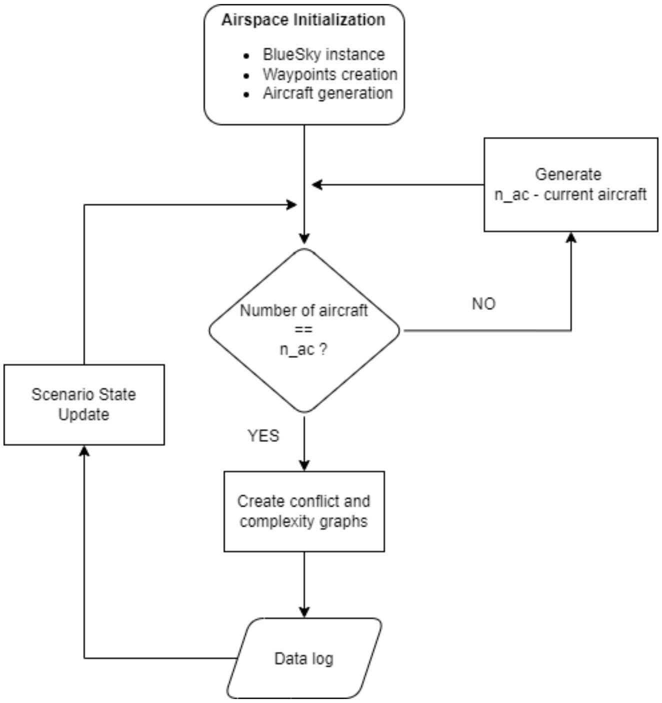


Figure B.3: Flowchart of the simulation code.

- *Conflict graph*: this is an auxiliary graph to find the compound conflicts. It is created when a conflict is detected. The edges connecting aircraft have no weights and they exist if there is a conflict between them. The way to compute the compound conflicts, as it were described in session 2.1.2, was to use `Networkx` library to find all the connected subgraphs in the graph. By selecting those with a size bigger than 2 we get the list of compound conflicts.

3. **Data logging**: to log the necessary data for the study, the Python library `sacred` has been used. It allowed us to log all the variables we needed at the corresponding time step. The variables that were extracted were:

    - Timestep
    - Whether conflict resolution method is active
    - Number of simulation
    - Circle radius
    - tCPA threshold
    - Number of aircraft
    - The four complexity indicators

- Number of conflicts detected
- Total number of conflicts
- Losses of separation detected
- Total number of losses of separation
- Number of compound conflicts
- Size of the biggest compound conflicts

The data was logged every timestep.

**Simulation runs**

The simulations were done varying two parameters: the number of aircraft (50, 70 and 100 and the tCPA threshold (25, 20 and 15 seconds). We did 500 simulations for each combination of parameters without any conflict resolution method and another 500 runs for each combination using MVP. Regarding the conflict detection parameters we have based the choice on the results shown in [**?**], where the safety for UAS and *separation minima* thresholds were study. The authors recommend to use an horizontal safety distance of 165 meters for sUAS. We have generated all the drones at the same height (300 ft) to reduce the simulations variability and execution time. Therefore, the choice of a vertical safety distance was not relevant. Based on sUAS typical size and speed we have chosen 15 seconds for the look-ahead time.

The radius of the simulations was 0.5 nautical miles as we wanted to simulate scenarios where drones were in close proximity, which will be the case in most of their promising applications. The simulation time was 185 seconds, as stated in the previous section.

## B.3 Results

In this section the simulation results are analysed. The machine where the simulation were made had 16 CPU cores so we could run 16 simulation at once. The parallelization didn't speed the program up 16 times because the processes managing slowed it down. However, the parallelized version took around 10 times less to do all the runs than the serial one: a remarkable improvement.

### B.3.1 Conflict distribution analysis

In this first part of the analysis of the results, a study of the amount and characteristics of the conflicts detected in the simulations has been done. Also, the effect that the resolution method has had on this values is studied, as well as the correlation between the conflicts and the number of aircrafts.

### Effect of CR on conflicts

In the logged data we can find the number of pair-wise conflicts, the number of Losses of Separation, the amount of compound conflicts and their sizes (the size of the biggest compound conflict if several were detected). Figure B.8a shows the number of conflicts depending on the number of aircraft and MVP. It can be seen that the total number of pair-wise conflicts is slightly higher in the case where we are using the Conflict Resolution method, which might seem counterintuitive but has several explanations. The first remark on this come from the definition of conflict. As it was stated in previous sections, a conflict is declared when a Loss of Separation is predicted to happen within a certain look-ahead time. The goal of a CR method is to solve conflicts in order to avoid a LoS, but it won't prevent drones from going into conflicts. In figure B.8b the total amount of detected Losses of Separation is plotted and it can be noticed how this number is lower when a conflict resolution method is used. The fact that we get a higher number of total conflicts when MVP is active in the simulations can be explained as follows: when a conflict is detected, both aircraft execute a manoeuvre in order to solve it. If the aircraft density is high, this can make the UAS to cause secondary conflict with surrounding drones. This increases the expected number of conflicts compared to the situation where MVP is not used, as the aircraft follows straight paths most of the time. This is a first indication of MVP poor performance in high-density airspace, as it is likely to create more conflicts after solving one.

We can also expect the number of compound conflicts to decrease when we are using a CR method. In Figure B.8c the percentage of time that we measure at least one conflict is shown. It can be seen that in the case without MVP this percentage is significantly higher. When no CR method is used, conflicts last till the involve drones are separated from each other beyond their safety distance. However, when MVP is used conflicts get solved as soon as they are detected, which is reflected in the low percentage of time during which conflicts are present. This also can be noticed in the amount of compound conflicts, as it can be seen in Figure B.8d. This number is significantly lower when a conflict resolution method is being used because conflicts are not prolonged long enough for their to become complex. For example, if two aircraft get into a conflict and a third one is going to get in conflict with one of them, a conflict resolution method would most likely solve the first conflict before the second one occurs, avoiding the creation of a compound conflict of size 3.

One more difference that can be found in the conflict distribution between the simulations that use MVP and the ones that don't is the size of the compound conflicts. Figure B.11a, shows the distribution of the compound conflict sizes. It can be seen that the compound conflicts tend to be smaller when using MVP, for the same reason stated before: the conflicts get resolved before they get the chance to evolve into more complex structures.

|  | $n_{ac} \sim n_{conf}$ | $n_{ac} \sim n_{compConf}$ | $n_{ac} \sim \sqrt{n_{compConf}}$ | $n_{ac} \sim confsize$ |
|---|---|---|---|---|
| Corr. coeff (No CR) | 0.99 | 0.98 | 0.99 | -0.13 |
| p-value (No CR) | 0.06 | 0.10 | 0.02 | 0.91 |
| Corr. coeff (With CR) | 0.99 | 0.98 | 0.99 | 0.39 |
| p-value (With CR) | 0.06 | 0.11 | 0.04 | 0.74 |

Table B.1: Results of the Correlation test performed between different conflict characteristics (Pearson Correlation Coefficient and p-value)

### Correlation between conflicts and UAS density

In Figure B.8 and Figure B.11 the different variables has been shown as a function of the number of aircraft. The effect of the number of drones present the airspace on the occurrence and characteristic of conflicts has been studied. To do so, a *Pearson Correlation test* has been carried out between the number of aircraft $n_{ac}$ and the number of pair-wise conflicts $n_{conf}$, the number of compound conflicts $n_{compConf}$ and the mean size of the compound conflicts $confsize$.

Looking at the figures, a monotonically increasing relation between $n_{ac}$ and both $n_{conf}$ and $n_{compConf}$ can be deduced. The Pearson correlation coefficient between these variables can be founded in Table 1. For the number of conflicts we got a coefficient of 0.99 for both cases (with and without CR), which mean that there exist a nearly perfect linear relation. As we are dealing only with three values (the total number of conflicts per number of aircraft) we should attend at the p-value of the test to check how certain can we be about this. In this case the p-value is 0.06, which under a typical 0.05 significance level would make us reject the hypothesis of both variables being linearly related. However, given the low number of data used in this test, we will accept it as true. When looking at the correlation between $n_{ac}$ and the number of compound conflicts we get a coefficient value of 0.98 with a p-value of 0.1. Again, a strong linear dependency is suggested by a Pearson coefficient value close to 1.0, but in this case the p-value is not small enough to accept it. However, it can be noticed in Figure B.8d that there might be a quadratic relationship between these variables, more than a linear one. To test this, we have repeated the Pearson test but taking the square root of $n_{compConf}$, because if they really follow a quadratic relation, taking the square root would cancel it out, leaving it a linear one. In this case, the correlation coefficient was 0.99 with a p-value of 0.02 for the case where no CR method is used and 0.04 for the case where MVP is active. We can accept the linear relation hypothesis under a 0.05 significance level, which allows us to accept as valid the idea that there is a quadratic relationship between the number of aircraft and the number of compound conflicts.

In Figure B.11b, the mean of the conflict sizes is shown. The correlation between $n_{ac}$ and the size of the compound conflicts is not clear in this figure. A Pearson test reveals a correlation coefficient of -0.13 with a p-value of 0.91 for the case without CR and 0.39 with a p-value of 0.74 for the case with CR. No linear correlation can be deducted between these variables from these tests.

This might be due to the fact that compound conflicts of sizes greater than 3 are very unlikely to happen, as it can be seen in Figure B.11a, giving us very few data to study this relation.

## B.4 Conclusions

In this work a large number of simulations were carried out with BlueSky to study the effect of different factors. The airspace (a 0.5 nautical miles radius circle) has been modelled as a graph with the interdependencies based on the Time of Closest Point of Approach. Following prior studies, the complexity has been defined and adapted to sUAS through 4 connectivity indicators: *Clustering Coefficient, Strength, Nearest Neighbor Degree* and *Edge Density*. The effect of a conflict resolution method (the Modified Voltage Potential) has been researched. Also, the number of drones has been varied in the simulation using 50, 70 and 100. The interdepencies threshold for the complexity graph were also varied, taking the values 25, 20, and 15 seconds.

In a first part, a study of conflicts has been made. The MVP method has proven to decrease the number of losses of separation. This is the result of conflicts being solved by the algorithm once detected. Also, the CR method of choice was observed to resolve conflicts in a quick way, which was reflected in two facts: the total time during which conflicts were detected was remarkably small and there were far fewer compound conflicts detected than when CR was not used. The reason for the latter is that conflicts were solved before they could evolve into more complex ones. However, it was noted that the total number of conflicts increased as a result of MVP, which can be explained by the fact that when it is applied in a high-density airspace, the aircraft involved in a conflict that the method is trying to solve cause secondary conflicts with surrounding drones when executing the avoidance manoeuvre. The results showed that under this conditions, MVP produces more conflicts that it solves.

The relation between the number of aircraft and the number of total conflicts turned out to be linear, as a Pearson Correlation test confirmed. Another correlation test demonstrated a linear relation between the number of aircraft and the square root of compound conflicts, which reveals a quadratic dependence between these kind of conflicts and the amount of drones. However no significant relation was found between the number of aircraft and the conflict sizes.

[b]0.475

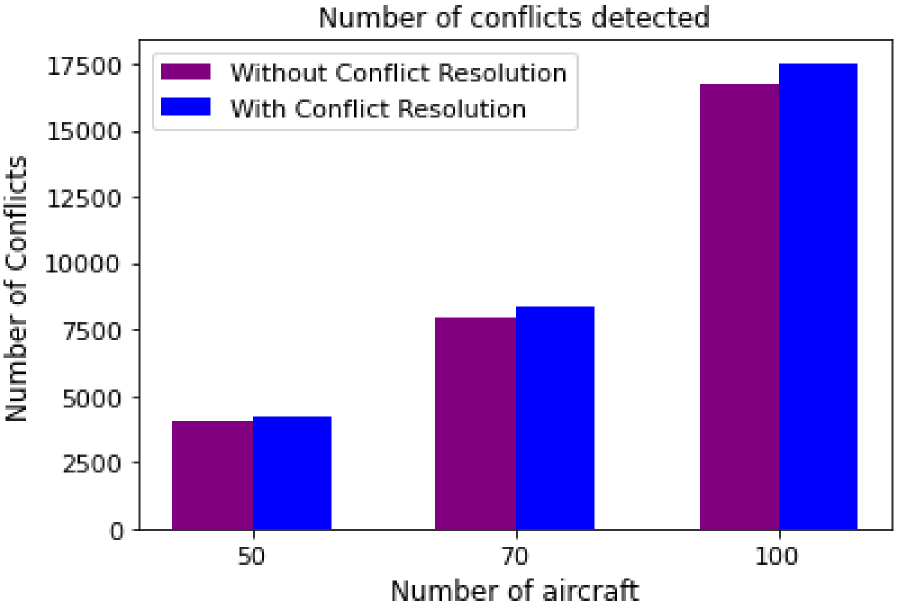


Figure B.4:

[b]0.475

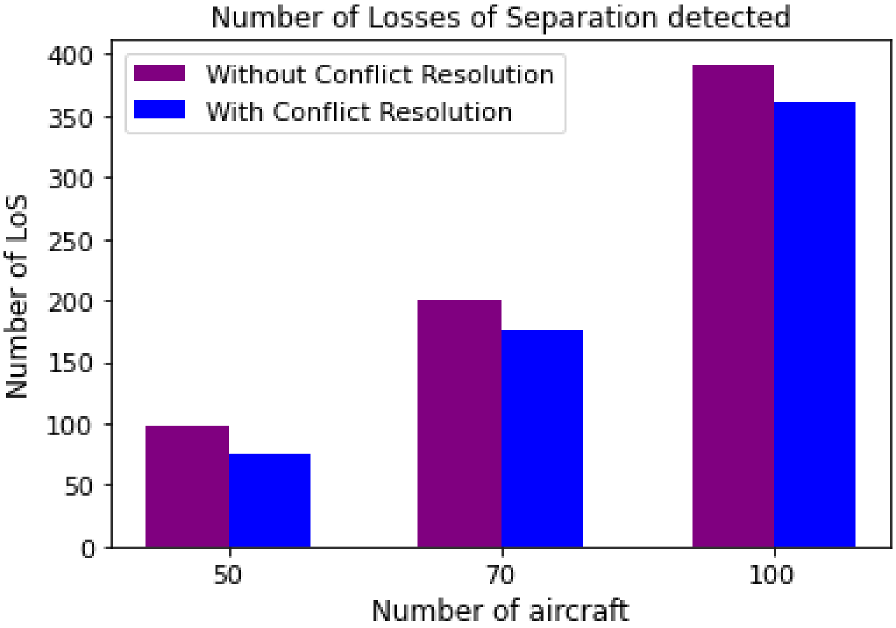
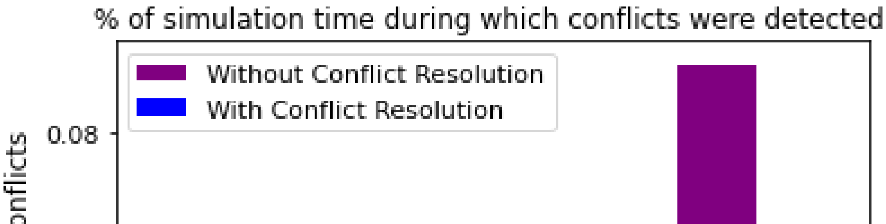


Figure B.5:

[b]0.475



[b]0.475

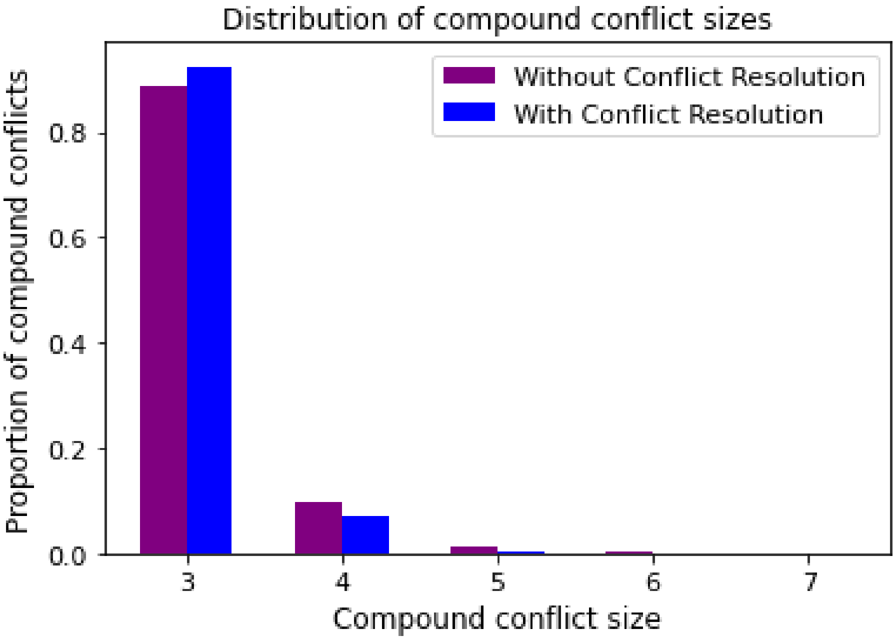


Figure B.9:

[b]0.475

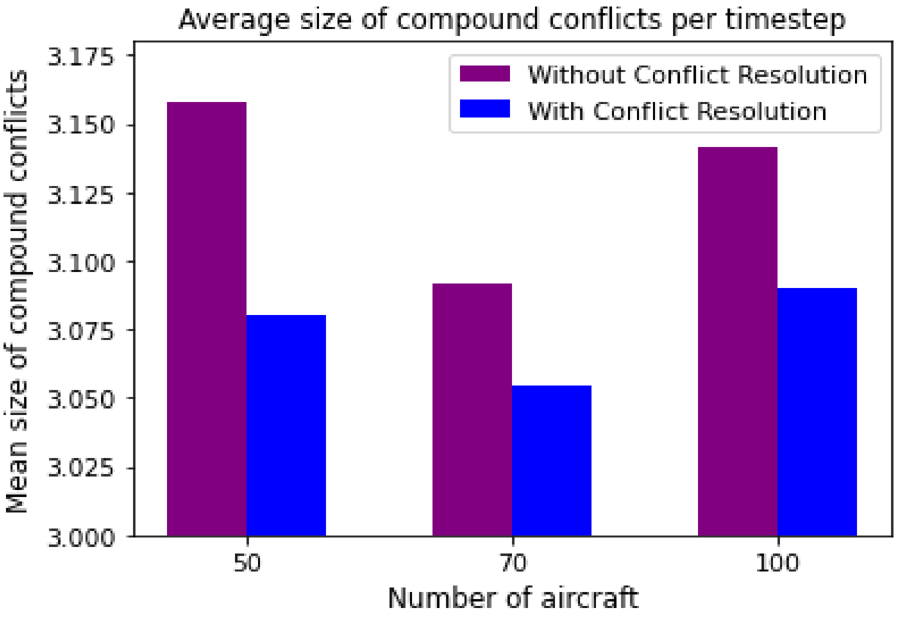


Figure B.10:

Figure B.11: Bar plot of the proportion of total conflicts that presented a certain size, separated by the present of Conflict Resolution methods in the simulations (a). Average size of the biggest compound conflict detected in every timestep depending on the number of aircraft (b)

# Appendix C

# List of Acronyms

- **AIS** → Aeronautical Information Services
- **AMAN** → Arrival Manager
- **ANAC** → Automated Negotiating Agents Competition
- **ANS** → Air Navigation services
- **AP** → Autopilot
- **AS** → Alerting Service
- **ASM** → Airspace Management
- **ATC** → Air Traffic Control service
- **ATCo** → Air Traffic Controller
- **ATFCM** → Air Traffic Flow and Capacity Management
- **ATFM** → Air Traffic Flow Management
- **ATM** → Air Traffic Management
- **ATS** → Air Traffic Services
- **CA** → Collision Avoidance
- **CD&R** → Conflict Detection & Resolution
- **CNS** → Communication, Navigation, and Surveillance services
- **CPA** → Closest Point of Approach
- **CR** → Conflict Resolution
- **CTA** → Controlled Areas

- **DAA** → Detect and Avoid
- **FAB** → Functional Airspace Blocks
- **FD** → Fractal Dimension
- **FDP** → Flight Data Processor
- **FIR** → Flight Information Region
- **FIS** → Flight Information Service
- **FL** → Flight Level
- **Flight Management System** → Flight Management System
- **ft** → feet
- **HMI** → Human-Machine Interface
- **IC** → Interval Complexity
- **IFR** → Instrumental Flight Rules
- **KPI** → Key Performance Indicator
- **MAS** → Multi-Agent System
- **NM** → Nautical Miles
- **OPM** → Object Process Methodology
- **RWC** → Remanin Well Clear
- **SAA** → See and Avoid
- **SA&A** → Sense and Avoid
- **SE** → Systems Engineering
- **SP** → Separation Provision
- **SORA** → Specific Operations Risk Assessment
- **STCA** → Short Term Conflict Alert
- **TBO** → Trajectory-Based Operations
- **TCAS** → Traffic alert and Collision Avoidance System
- **TD** → Tactical Deconfliction
- **TLS** → Target Level of Safety
- **TLS** → Target Level of Safety

- **UAM** → Urban Air Mobility
- **UAS** → Unmanned Aircraft System
- **UTM** → Unmanned Traffic Management
- **VFR** → Visual Flight Rules
- **WC** → Well Clear

# List of Figures

# List of Tables

# Bibliography

[1] European Drones Outlook Study Unlocking the value for Europe, 2016.

[2] Parimal Kopardekar, Thomas Prevot, Joseph Rios, John E. Robinson III, Jaewoo Jung, and Marcus Johnson. UAS Traffic Management (UTM) Concept of Operations to Safely Enable Low Altitude Flight Operations. *16th AIAA Aviation Technology, Integration, and Operations Conference*, 2016.

[3] Pamela Whitley. FAA UTM Concept of Operations v2.0, 2020.

[4] Cristina Barrado, Mario Boyero, Luigi Brucculeri, Giancarlo Ferrara, Andrew Hately, Peter Hullah, David Martin-Marrero, Enric Pastor, Anthony Peter Rushton, and Andreas Volkert. U-space concept of operations: A key enabler for opening airspace to emerging low-altitude operations. *Aerospace*, 7(3), mar 2020.

[5] Andrija Vidović, Tomislav Mihetec, Bo Wang, and Igor Štimac. Operations of drones in controlled airspace in europe. *International Journal for Traffic & Transport Engineering*, 9(1), 2019.

[6] Safety Management Manual, Doc9859, Fourth Edition, 2018.

[7] Roland E Weibel and R John Hansman. Safety considerations for operation of unmanned aerial vehicles in the national airspace system. Technical report, 2006.

[8] Global Air Traffic Management Operational Concept, Doc9854, AN/458, 2015.

[9] FAA. Introduction to tcas ii version 7.1. Technical report, February 2011.

[10] César Munoz, Anthony Narkawicz, and James Chamberlain. A tcas-ii resolution advisory detection algorithm. In *AIAA Guidance, Navigation, and Control (GNC) Conference*, page 4622, 2013.

[11] Tim McCarthy, Lars Pforte, and Rebekah Burke. Fundamental elements of an urban utm. *Aerospace*, 7(7):85, 2020.

[12] Eric Perrin, Barry Kirwan, and Ron Stroup. A systemic model of atm safety: the integrated risk picture. In *Proc. 7th USA/Europe ATM R&D Seminar, Barcelona*. Citeseer, 2007.

[13] Derek Fowler, Gilles Le Galo, E Perrin, and S Thomas. So it's reliable but is it safe? a more balanced approach to atm safety assessment. In *7th USA/Europe ATM R&D Seminar (ATM 2007), Paper*, volume 41, pages 02–05. Citeseer, 2007.

[14] James K. Kuchar and Ann C. Drumm. The traffic alert and collision avoidance system. 2007.

[15] Subramanian Ramasamy. Next generation flight management systems for manned and unmanned aircraft operations - automated separation assurance and collision avoidance functionalities. 2017.

[16] Rules of the Air Annex 2 to the Convention on International Civil Aviation International Civil Aviation Organization International Standards Tenth Edition, 2005.

[17] Unmanned aircraft systems traffic management (utm)-a common framework with core principles for global harmonization edition 3.

[18] Jarrett Larrow Marcus Johnson. UAS Traffic Management Conflict Management Model, 2020.

[19] Separation Management in the U-Space: a key enabler to boost U3 services take-off. Technical report, 2021.

[20] Marsel Omeri, Ralvi Isufaj, and Romualdo Moreno Ortiz. Quantifying well clear for autonomous small uas. *IEEE Access*, 10:68365–68383, 2022.

[21] Thomas Dubot and Antoine Joulia. Towards U-space conflict management services based on 4D protection bubbles. In *AIAA AVIATION 2021 FORUM*, VIRTUAL, United States, August 2021. American Institute of Aeronautics and Astronautics.

[22] Robert Schultz, Donald Shaner, Yiyuan Zhao, Robert Schultz, Donald Shaner, and Yiyuan Zhao. *Free flight concept*.

[23] Yazdi I. Jenie, Erik-Jan van Kampen, Joost Ellerbroek, and Jacco M. Hoekstra. Taxonomy of conflict detection and resolution approaches for unmanned aerial vehicle in an integrated airspace. *IEEE Transactions on Intelligent Transportation Systems*, 18(3):558–567, 2017.

[24] Small uav market size, share growth analysis report by product (fixed wing, helicopter, rotary blade), by application (defense and commercial and civil) - global industry analysis, trends, segment forecasts, regional outlook 2021 - 2027.

[25] Teal Group Corporation. Teal Group World Civil UAS Market Profile and Forecast, 2020.

[26] Andrew J. Weinert, Scot Edward Campbell, Adan Ernesto Vela, Dieter W. Schuldt, and Joel A. Kurucar. Well-clear recommendation for small unmanned aircraft systems based on unmitigated collision risk. *Journal of Air Transportation*, 2018.

[27] Timothy W. McLain and Matthew O. Duffield. A well clear recommendation for small uas in high-density, ads-b-enabled airspace. 2017.

[28] María Consiglio, César Muñoz, George Hagen, Anthony Narkawicz, and Swee Balachandran. ICAROUS INTEGRATED CONFIGURABLE ALGORITHMS FOR RELIABLE OPERATIONS OF UNMANNED SYSTEMS. Technical report.

[29] Eric Mueller and Mykel J. Kochenderfer. Simulation comparison of collision avoidance algorithms for multi-rotor aircraft. In *AIAA Modeling and Simulation Technologies Conference, 2016*. American Institute of Aeronautics and Astronautics Inc, AIAA, 2016.

[30] J.K. Kuchar and L.C. Yang. A review of conflict detection and resolution modeling methods. *IEEE Transactions on Intelligent Transportation Systems*, 1(4):179–189, 2000.

[31] Jun Tang. Conflict detection and resolution for civil aviation: A literature survey. *IEEE Aerospace and Electronic Systems Magazine*, 34(10):20–35, 2019.

[32] Marta Ribeiro, Joost Ellerbroek, and Jacco Hoekstra. Review of conflict resolution methods for manned and unmanned aviation. *Aerospace*, 7(6):79, 2020.

[33] Jun Tang, Songyang Lao, and Yu Wan. Systematic review of collision-avoidance approaches for unmanned aerial vehicles. *IEEE Systems Journal*, 2021.

[34] Jawad N. Yasin, Sherif A.S. Mohamed, Mohammad Hashem Haghbayan, Jukka Heikkonen, Hannu Tenhunen, and Juha Plosila. Unmanned Aerial Vehicles (UAVs): Collision Avoidance Systems and Approaches. *IEEE Access*, 8:105139–105155, 2020.

[35] Imen Mahjri, Amine Dhraief, and Abdelfettah Belghith. A review on collision avoidance systems for unmanned aerial vehicles. In *International Workshop on Communication Technologies for Vehicles*, pages 203–214. Springer, 2015.

[36] Xiang Yu and Youmin Zhang. Sense and avoid technologies with applications to unmanned aircraft systems: Review and prospects. *Progress in Aerospace Sciences*, 74:152–166, 2015.

[37] GUAN Xiangmin, LYU Renli, SHI Hongxia, and CHEN Jun. A survey of safety separation management and collision avoidance approaches of civil uas operating in integration national airspace system. *Chinese Journal of Aeronautics*, 33(11):2851–2863, 2020.

[38] Jeremy Dick, Elizabeth Hull, Ken Jackson, Jeremy Dick, Elizabeth Hull, and Ken Jackson. Requirements engineering in the problem domain. *Requirements Engineering*, pages 113–134, 2017.

[39] George Hunter and Peng Wei. Service-oriented separation assurance for small uas traffic management. In *2019 Integrated Communications, Navigation and Surveillance Conference (ICNS)*, pages 1–11, 2019.

[40] Subramanian Ramasamy, Roberto Sabatini, and Alessandro Gardi. A Unified Analytical Framework for Aircraft Separation Assurance and UAS Sense-and-Avoid. *Journal of Intelligent and Robotic Systems: Theory and Applications*, 2018.

[41] Evgenii Vinogradov, Franco Minucci, and Sofie Pollin. Wireless communication for safe uavs: From long-range deconfliction to short-range collision avoidance. *IEEE Vehicular Technology Magazine*, 15(2):88–95, 2020.

[42] Hilary Arksey and Lisa O'Malley. Scoping studies: towards a methodological framework. *International Journal of Social Research Methodology*, 8:19 – 32, 2005.

[43] Hannah Snyder. Literature review as a research methodology: An overview and guidelines. *Journal of business research*, 104:333–339, 2019.

[44] NASA Technical Documents, Papers and Presentations. `https://www.nasa.gov/aeroresearch/utm-tech-docs-papers-presentations`. Accessed: 2022-07-18.

[45] SESAR JU U-SPACE Ongoing, Closed Projects and Publications. `https://www.sesarju.eu/U-space`. Accessed: 2022-07-18.

[46] César Muñoz, Anthony Narkawicz, George Hagen, Jason Upchurch, Aaron Dutle, María Consiglio, and James Chamberlain. Daidalus: Detect and avoid alerting logic for unmanned systems. In *2015 IEEE/AIAA 34th Digital Avionics Systems Conference (DASC)*, pages 5A1–1–5A1–12, 2015.

[47] Luis E Alvarez, Ian Jessen, Michael P Owen, Joshua Silbermann, and Paul Wood. ACAS sXu : Robust Decentralized Detect and Avoid for Small Unmanned Aircraft Systems. 2019.

[48] Deconfliction and Separation Management. `https://www.icao.tv/drone-enable/season:2/videos/deconfliction-and-separation-management-continued`. Accessed: 2022-07-18.

[49] Benjamin S Blanchard, Wolter J Fabrycky, and Walter J Fabrycky. *Systems engineering and analysis*, volume 4. Prentice hall Englewood Cliffs, NJ, 1990.

[50] Charles S Wasson. *System engineering analysis, design, and development: Concepts, principles, and practices.* John Wiley & Sons, 2015.

[51] David D Walden, Garry J Roedler, and Kevin Forsberg. Incose systems engineering handbook version 4: Updating the reference for practitioners. In *INCOSE International Symposium*, volume 25, pages 678–686. Wiley Online Library, 2015.

[52] Robert Shishko and Robert Aster. Nasa systems engineering handbook. *NASA Special Publication*, 6105, 1995.

[53] James N Martin. *Systems engineering guidebook: A process for developing systems and products.* CRC press, 2020.

[54] Derek Hitchins. Emergence, hierarchy, complexity, architecture. 2008.

[55] Rick Adcock, Nicole Hutchison, and Claus Nielsen. Defining an architecture for the systems engineering body of knowledge. In *2016 Annual IEEE Systems Conference (SysCon)*, pages 1–7. IEEE, 2016.

[56] Niamat Ullah Ibne Hossain. A synthesis of definitions for systems engineering. In *Proceedings of the International Annual Conference of the American Society for Engineering Management.*, pages 1–10. American Society for Engineering Management (ASEM), 2018.

[57] Russell L Ackoff. Towards a system of systems concepts. *Management science*, 17(11):661–671, 1971.

[58] David Long and Zane Scott. *A primer for model-based systems engineering.* Lulu. com, 2011.

[59] Mark Bedau. Downward causation and the autonomy of weak emergence. *Principia: an international journal of epistemology*, 6(1):5–50, 2002.

[60] Jon Holt. *Systems Engineering Demystified: A practitioner's handbook for developing complex systems using a model-based approach.* Packt Publishing Ltd, 2021.

[61] Alan Faisandier and Garry Roedler. System architecture.

[62] Sheila R Conway and Maria C Consiglio. Towards a functionally-formed air traffic system-of-systems. In *2005 IEEE International Conference on Systems, Man and Cybernetics*, volume 4, pages 3167–3172. IEEE, 2005.

[63] Alan Faisandier and Garry Roedler. Logical architecture.

[64] Alan Faisandier and Rick Adcock. Physical architecture.

[65] David Harel and Amir Pnueli. On the development of reactive systems. In *Logics and models of concurrent systems*, pages 477–498. Springer, 1985.

[66] Maja J. Matarić and François Michaud. *Behavior-Based Systems*, pages 891–909. Springer Berlin Heidelberg, Berlin, Heidelberg, 2008.

[67] Mauricio Castillo-Effen and Nikita A Visnevski. Analysis of autonomous deconfliction in unmanned aircraft systems for testing and evaluation. In *2009 IEEE Aerospace conference*, pages 1–12. IEEE, 2009.

[68] Chet Richards. Boyd's ooda loop. 2020.

[69] James S. Albus and WG Rippey. Rcs: A reference model architecture for intelligent control. In *Proceedings of PerAc'94. From Perception to Action*, pages 218–229. IEEE, 1994.

[70] Erik Theunissen and Tessa Hope Veerman. Automated detect and avoid: Autonomy and ethics. In *2018 Integrated Communications, Navigation, Surveillance Conference (ICNS)*, pages 2A1–1. IEEE, 2018.

[71] James S Albus. 4d/rcs: a reference model architecture for intelligent unmanned ground vehicles. In *Unmanned Ground Vehicle Technology IV*, volume 4715, pages 303–310. SPIE, 2002.

[72] Tim Hutchings, Susan Jeffryes, and SJ Farmer. Architecting uav sense & avoid systems. In *2007 Institution of Engineering and Technology Conference on Autonomous Systems*, pages 1–8. IET, 2007.

[73] Ronald C Arkin, Ronald C Arkin, et al. *Behavior-based robotics*. MIT press, 1998.

[74] William N Kaliardos. *Semi-structured decision processes: a conceptual framework for understanding human-automation systems*. PhD thesis, Massachusetts Institute of Technology, 1999.

[75] Karthik Ramalingam. *System elements required to guarantee the reliability, availability and integrity of decision-making information in a complex airborne autonomous system*. PhD thesis, Loughborough University, 2017.

[76] Stephen P. Cook, Dallas Brooks, Rodney Cole, Davis Hackenberg, and Vincent Raska. Defining well clear for unmanned aircraft systems. In *AIAA Infotech at Aerospace*. American Institute of Aeronautics and Astronautics Inc., 2015.

[77] Xiangmin Guan, Renli Lyu, Hongxia Shi, and Jun Chen. A survey of safety separation management and collision avoidance approaches of civil uas operating in integration national airspace system. *Chinese Journal of Aeronautics*, 33:2851–2863, 2020.

[78] Dilan Amarasinghe and Siu O'Young. Estimation of alerting thresholds for sense-and-avoid. In *2010 IEEE International Workshop on Robotic and Sensors Environments*, 2010.

[79] Reece Clothier, Brendan Williams, and Neale Leslie Fulton. Structuring the safety case for unmanned aircraft system operations in non-segregated airspace. *Safety Science*, 79:213–228, 2015.

[80] Seung Man Lee, Chunki Park, Marcus A. Johnson, and Eric R. Mueller. Investigating Effects of Well Clear Definitions on UAS Sense-And-Avoid Operations in Enroute and Transition Airspace. In *2013 Aviation Technology, Integration, and Operations Conference*, Reston, Virginia, aug 2013. American Institute of Aeronautics and Astronautics.

[81] Yazdi Jenie, Erik-Jan Van Kampen, and Bart Remes. *Cooperative Autonomous Collision Avoidance System for Unmanned Aerial Vehicle*, pages 387–405. 01 2013.

[82] Guido Manfredi, Yannick Jestin, Guido Manfredi, Yannick Jestin, An Introduction, Ahead Dasc, and Guido Manfredi. An Introduction to ACAS Xu and the Challenges Ahead To cite this version : HAL Id : hal-01638049 An Introduction to ACAS Xu and the Challenges Ahead. 2017.

[83] Brandon Cook, Tim Arnett, and Kelly Cohen. Provisional chapter A Fuzzy Logic Approach for Separation Assurance and Collision Avoidance for Unmanned Aerial Systems. Technical report, 2017.

[84] César Muñoz and Anthony Narkawicz. Formal analysis of extended well-clear boundaries for unmanned aircraft. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 9690(Tcas Ii):221–226, 2016.

[85] Jaewoo Jung, Sarah N. D'Souza, Marcus A. Johnson, Abraham K. Ishihara, Hemil C. Modi, Ben Nikaido, and Hashmatullah Hasseeb. Applying required navigation performance concept for traffic management of small unmanned aircraft systems. In *30th Congress of the International Council of the Aeronautical Sciences, ICAS 2016*. International Council of the Aeronautical Sciences, 2016.

[86] Seung Man Lee, David P Thipphavong, Douglas R Isaacson, Confesor Santiago, Chunki Park, and volume = 219067 year = 2016 title = Evaluating Alerting and Guidance Performance of a UAS Detect-And-Avoid System, url = http://www.sti.nasa.gov.

[87] Evaluating the Preventive Alert Function for UAS Detect and Avoid Systems. Technical report, 2021.

[88] Karl Bilimoria. Survey of air/ground and human/automation functional allocation for separation assurance. In *12th AIAA Aviation Technology,*

*Integration, and Operations (ATIO) Conference and 14th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, page 5413, 2012.

[89] Seung Man Lee and Eric R Mueller. A systems-based approach to functional decomposition and allocation for developing uas operational concepts. In *2013 Aviation Technology, Integration, and Operations Conference*, page 4241, 2013.

[90] Jordan Sakakeeny, Husni R Idris, Devin Jack, and Vishwanath Bulusu. A framework for dynamic architecture and functional allocations for increasing airspace autonomy. In *AIAA AVIATION 2022 Forum*, page 3702, 2022.

[91] Gano Broto Chatterji, Vishwanath Bulusu, Husni R Idris, and Jordan Sakakeeny. Functional allocation approach for separation assurance for remotely piloted aircraft. In *AIAA AVIATION 2022 Forum*, page 3703, 2022.

[92] Roland E Weibel, Matthew W M Edwards, and Caroline S Fernandes. Establishing a Risk-Based Separation Standard for Unmanned Aircraft Self Separation. Technical report, 2011.

[93] Sense and Avoid (SAA) for Unmanned Aircraft Systems (UAS) – Second Caucus Workshop Report. Technical report, 2013.

[94] Stephen P. Cook and Dallas Brooks. A Quantitative Metric to Enable Unmanned Aircraft Systems to Remain Well Clear. *Air Traffic Control Quarterly*, 23(2-3):137–156, 2015.

[95] César A. Muñoz, Anthony Narkawicz, James Chamberlain, Jason Upchurch, María Consiglio, and Jason Upchurch. A family of well-clear boundary models for the integration of UAS in the NAS. *AIAA AVIATION 2014 -14th AIAA Aviation Technology, Integration, and Operations Conference*, (June), 2014.

[96] RTCA (Firm). SC-228. *Minimum Operational Performance Standards (MOPS) for Detect and Avoid (DAA) Systems.* RTCA, Incorporated, 2017.

[97] Christine Chen, Matthew W. Edwards, Bilal Gill, Samantha Smearcheck, Tony Adami, Sean Calhoun, Minghong G. Wu, Andrew Cone, and Seung Man Lee. Defining Well Clear Separation for Unmanned Aircraft Systems Operating with Noncooperative Aircraft. American Institute of Aeronautics and Astronautics (AIAA), jun 2019.

[98] Andrew Cone, Gilbert Wu, and Seung Lee. Detect-and-avoid alerting performance for high-speed uas and non-cooperative aircraft. 06 2019.

[99] RTCA-365B. SC-228. *Minimum Operational Performance Standards (MOPS) for Detect and Avoid (DAA) Systems.* RTCA, Incorporated, 2021.

[100] ASTM F3442/F3442M-20. *Standard Specification for Detect and Avoid System Performance Requirements.* 2020.

[101] Michael Mullins, Michael W. Holman, Kyle Foerster, Naima Kaabouch, and William Semke. Dynamic separation thresholds for a small airborne sense and avoid system. 2013.

[102] Sally C. Johnson, Alexander N. Petzen, and Dylan S. Tokotch. Exploration of detect-and-avoid and well-clear requirements for small uas maneuvering in an urban environment. 2017.

[103] Florence Ho, Ruben Geraldes, Artur Gonçalves, Marc Cavazza, and Helmut Prendinger. Improved conflict detection and resolution for service uavs in shared airspace. *IEEE Transactions on Vehicular Technology*, 68(2):1231–1242, 2019.

[104] Institute of Electrical and Electronics Engineers, Andreas Volkert space IEEE AeroNiklas Peinecke, Lennart Limmer, Electronic Systems Society, American Institute of Aeronautics and Astronautics, and American Institute of Aeronautics and Astronautics. Digital Avionics Technical Committee. Application of ”Well Clear” to Small Drones.

[105] Hao Yi Ong and Mykel Kochenderfer. Markov decision process-based distributed conflict resolution for drone air traffic management. *Journal of Guidance, Control, and Dynamics*, 40:1–12, 10 2016.

[106] Investigating effects of ”well clear” definitions on UAS sense-and-avoid operations. 2013.

[107] Sang Hyun Kim. Conflict Risk Assessment of Structured and Unstructured Traffic of Small Unmanned Aircraft Systems. *2018 Aviation Technology, Integration, and Operations Conference*, 2018.

[108] Devin P. Jack, Keith D. Hoffler, and Sally C. Johnson. Effects of basing UAS DAA requirements on intruder speeds lower than the statute speed limit. In *AIAA Atmospheric Flight Mechanics Conference*. American Institute of Aeronautics and Astronautics Inc, AIAA, 2016.

[109] Devin P. Jack, Jeremy Hardy, and Keith D. Hoffler. Analysis of influence of UAS speed range and turn performance on detect and avoid sensor requirements. *2018 Aviation Technology, Integration, and Operations Conference*, pages 1–10, 2018.

[110] Devin P. Jack, Keith D. Hoffler, and Sally Johnson. An Assessment of Unmanned Aircraft System Level-Turn Maneuver Performance Requirements In Relation to a Quantified Well-Clear Definition. American Institute of Aeronautics and Astronautics (AIAA), jun 2015.

[111] Minghong Wu, Seungman Lee, and Andrew C. Cone. Detect and avoid alerting performance with limited surveillance volume for non-cooperative aircraft. *AIAA Scitech 2019 Forum*, 2019.

[112] Jason T Davies and Minghong G Wu. Comparative Analysis of ACAS-Xu and DAIDALUS Detect-and-Avoid Systems. (February), 2018.

[113] Michael P. Owen, Adam Panken, Robert Moss, Luis Alvarez, and Charles Leeper. Acas xu: Integrated collision avoidance and detect and avoid capability for uas. In *2019 IEEE/AIAA 38th Digital Avionics Systems Conference (DASC)*, pages 1–10, 2019.

[114] Chester Gong, Minghong G Wu, and Confesor Santiago. UAS Integration in the NAS Project : Flight Test 3 Data Analysis of JADEM-Autoresolver Detect and Avoid System. (December 2016), 2020.

[115] Timothy Grebe. Detect and Avoid ( DAA ) Alerting Performance Comparison : CPDS vs . ACAS-Xu. 2019.

[116] Dov Dori. Object-process analysis: maintaining the balance between system structure and behaviour. *Journal of Logic and Computation*, 5(2):227–249, 1995.

[117] Andrew Hately, A Van Swalm, and A Volkert. U-space concept of operations. In *CORUS*, 2019.

[118] U-space ConOps (edition 3.10) , 2022.

[119] Marta Ribeiro, Joost Ellerbroek, and Jacco Hoekstra. Review of conflict resolution methods for manned and unmanned aviation. *Aerospace*, 7(6), 2020.

[120] Emmanuel Sunil, Ólafur órarson, Joost Ellerbroek, and Jacco M Hoekstra. Analyzing the Effect of Traffic Scenario Properties on Conflict Count Models. Technical report.

[121] Timothy B. Nysetvold and John L. Salmon. Deconfliction in high-density unmanned aerial vehicle systems. *Journal of Air Transportation*, 27(2):61–69, 2019.

[122] Andrew Weinert, Luis Alvarez, Michael Owen, and Benjamin Zintak. Near midair collision analog for drones based on unmitigated collision risk. *Journal of Air Transportation*, 30(2):37–48, 2022.

[123] Hemil C. Modi, Abraham K. Ishihara, Jaewoo Jung, Ben E. Nikaido, Sarah N. D'souza, Hashmatullah Hasseeb, and Marcus Johnson. Applying required navigation performance concept for traffic management of small unmanned aircraft systems. 2016.

[124] Improved Conflict Detection and Resolution for Service UAVs in Shared Airspace. *IEEE Transactions on Vehicular Technology*, 68(2):1231–1242, feb 2019.

[125] R Geister, L. Limmer, M. Rippl, and T. Dautermann. Total system error performance of drones for an unmanned pbn concept. 2018.

[126] Valtteri Kallinen, Terry Martin, and Aaron McFadyen. Required navigation performance specifications for unmanned aircraft based on utm flight trials. 2020.

[127] R. J. Stamm and J. Glaneuski. Unmanned aircraft sense and avoid: Leveraging atc infrastructure. *2015 IEEE Aerospace Conference*, pages 1–8, 2015.

[128] Donald Maki, Christine Parry, Kyle Noth, Matthew Molinario, and Raymond Miraflor. Dynamic protection zone alerting and pilot maneuver logic for ground based sense and avoid of unmanned aircraft systems. In *Infotech@Aerospace*, 2012.

[129] I Cao. Guidance material on comparison of surveillance technologies (gmst). *International Civil Aviation Organization*, 2007.

[130] Neno Ruseno, Chung-Yan Lin, and Shih-Cheng Chang. Uas traffic management communications: The legacy of ads-b, new establishment of remote id, or leverage of ads-b-like systems? *Drones*, 6(3), 2022.

[131] Jueming Hu, Heinz Erzberger, Kai Goebel, and Yongming Liu. Risk-based dynamic anisotropic operational safety bound for rotary uav traffic control. *Annual Conference of the PHM Society*, 11, 09 2019.

[132] Brendan Duffy, Swee Balachandran, Andrew Peters, Kyle Smalling, Maria Consiglio, Louis Glaab, Andrew Moore, and César Muñoz. Onboard autonomous sense and avoid of non-conforming unmanned aerial systems. pages 1–10, 10 2020.

[133] Jason W. Adaska. Computing risk for unmanned aircraft self separation with maneuvering intruders. In *2012 IEEE/AIAA 31st Digital Avionics Systems Conference (DASC)*, pages 8A4–1–8A4–10, 2012.

[134] Jochum Tadema, E. Theinissen, and Kevin Kirk. Self separation support for uas. 2010.

[135] Dilan Amarasinghe and Siu O'Young. Estimation of alerting thresholds for sense-and-avoid. In *2010 IEEE International Workshop on Robotic and Sensors Environments*, pages 1–6, 2010.

[136] Swee Balachandran, Anthony Narkawicz, César A. Muñoz, and Maria C. Consiglio. A path planning algorithm to enable well-clear low altitude uas operation beyond visual line of sight. 2017.

[137] Andrew J. Moore, Swee Balachandran, Steven D. Young, Evan Dill, Michael J. Logan, Louis J. Glaab, César A. Muñoz, and Maria C. Consiglio. Testing enabling technologies for safe uas urban operations. *2018 Aviation Technology, Integration, and Operations Conference*, 2018.

[138] Niklas Peinecke and Alexander Kuenz. Deconflicting the urban drone airspace. In *2017 IEEE/AIAA 36th Digital Avionics Systems Conference (DASC)*, pages 1–6, 2017.

[139] Swee Balachandran, Viren Bajaj, Marco A. Feliú, César A. Muñoz, and María C. Consiglio. A learning-based guidance selection mechanism for a formally verified sense and avoid algorithm. In *2019 IEEE/AIAA 38th Digital Avionics Systems Conference (DASC)*, pages 1–6, 2019.

[140] Víctor A. Carreño, María Consiglio, and César A. Muñoz. Analysis and preliminary results of a concept for detect and avoid in the cockpit. In *2019 IEEE/AIAA 38th Digital Avionics Systems Conference (DASC)*, pages 1–8, 2019.

[141] Michael Abramson, M. Refai, Minghong Wu, and Seungman Lee. Applying sensor uncertainty mitigation schemes to detect-and-avoid systems. *AIAA AVIATION 2020 FORUM*, 2020.

[142] Stephen W. Scheff. Chapter 8 - nonparametric statistics. In Stephen W. Scheff, editor, *Fundamental Statistical Principles for the Neurobiologist*, pages 157–182. Academic Press, 2016.

[143] Yazdi Ibrahim Jenie, Erik-Jan van Kampen, and Bart Remes. Cooperative autonomous collision avoidance system for unmanned aerial vehicle. In Qiping Chu, Bob Mulder, Daniel Choukroun, Erik-Jan van Kampen, Coen de Visser, and Gertjan Looye, editors, *Advances in Aerospace Guidance, Navigation and Control*, pages 387–405, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg.

[144] R. Arteaga, Mike Dandachy, Hong Linh Truong, Arun Aruljothi, Mihir Vedantam, Kraettli Epperson, and Reed McCartney. [micron]ads-b detect and avoid flight tests on phantom 4 unmanned aircraft system. 2018.