Dedicated to innovation in aerospace

Aerospace and Defense

NLR-CR-2021-050-RevEd-1 | November 2021

D(emo)-CRAT Demonstration and Final Report

Demonstrator Drone Collision Risk Assessment Tool

CUSTOMER: EUROCONTROL

NLR – Royal Netherlands Aerospace Centre

D(emo)-CRAT Demonstration and Final Report

Demonstrator Drone Collision Risk Assessment Tool

Problem area

Assuring safe integration of Unmanned Aircraft Systems (UAS) or drones in all airspace classes is an important element of European research in SESAR. A central element in the European development is U-space, which is a set of new services and specific procedures designed to support safe, efficient and secure access to airspace for large numbers of drones in U-space airspaces. Safety risk assessment of drone operations and U-Space services are essential for their introduction.

For quantitative assessment of drone collision risks there is a strong need for simulation approaches that can represent a variety of drone operations and the types of uncertainty and hazards that can affect drone operations. These simulations should be able to achieve results up to the level of collision risk, meaning that they must support rare event estimation.

Description of work

This project develops a demonstrator tool for modelling and Monte Carlo simulation of drone traffic, called D(emo)-CRAT (Demonstrator Drone Collision Risk Assessment Tool). The Monte Carlo (MC) simulation uses acceleration by Interacting Particle Systems (IPS) to efficiently achieve results up to the collision risk level. The purpose of the demonstrator tool is to show that useful collision risk results can be achieved by the modelling and simulation, and as such may set the scene for further development of a full scale Drone Collision Risk Assessment Tool (D-CRAT).

This is the final report which presents a high-level overview of the models and the software tool, and which provides the detailed results of a demonstration of the tool to a use case for drone and air taxi traffic in an urban area south of Paris.

REPORT NUMBER
NLR-CR-2021-050-RevEd-1

AUTHOR(S)
S.H. Stroeve
G.J. Bakker
M.H.C. Everdij
M. Trezza
C.V. Cañizares
I.M. Calle
D.R. García

REPORT CLASSIFICATION
UNCLASSIFIED

DATE
November 2021

KNOWLEDGE AREA(S)
Safety

DESCRIPTOR(S)
drone operations
collision risk
urban air mobility
detect and avoid

Results and conclusions

The results of the project show that agent-based dynamic risk modelling can be effectively applied to assess close proximity safety events of drone operations. The IPS MC simulation approach and the risk decomposition for global failure conditions are advanced techniques for rare event risk assessment that could be effectively combined with the agent-based modelling of drone operations. The demonstration of the tool has shown that close proximity probabilities can be attained for various types of drone missions. The sensitivity analyses show how the tool can be used to tune settings of the DAA system, to attain insight in the safety impact of airspace design and traffic density, and to understand the safety impact of PIC behaviour. A considerable set of recommendations for extension of the models and their simulation, scope of simulation functionalities, and extension of the functionalities of the GUI are presented. An initial assessment of timescale and effort has been made to help prioritization of future developments of D-CRAT. The models, simulation approaches and software developed in D(emo)-CRAT have shown to be at the heart of the needs for new safety modelling and assessment methodologies for U-space such as identified in the Strategic Research and Innovation Agenda “Digital European Sky”.

Applicability

Collision risk assessment of drone operations and urban air mobility, including evaluation of detect and avoid systems.

NLR

Anthony Fokkerweg 2

1059 CM Amsterdam, The Netherlands

p) +31 88 511 3113

e) info@nlr.nl i) www.nlr.nl

Dedicated to innovation in aerospace

NLR-CR-2021-050-RevEd-1 | November 2021

D(emo)-CRAT Demonstration and Final Report

Demonstrator Drone Collision Risk Assessment Tool

CUSTOMER: EUROCONTROL

AUTHOR(S):

S.H. Stroeve	NLR
G.J. Bakker	NLR
M.H.C. Everdij	NLR
M. Trezza	everis Aerospace and Defense
C.V. Cañizares	everis Aerospace and Defense
I.M. Calle	everis Aerospace and Defense
D.R. García	everis Aerospace and Defense

NLR – Royal Netherlands Aerospace Centre

CUSTOMER	EUROCONTROL
CONTRACT NUMBER	19-220550-C
OWNER	NLR
DIVISION NLR	Aerospace Operations
DISTRIBUTION	Limited
CLASSIFICATION OF TITLE	UNCLASSIFIED

APPROVED BY :		
AUTHOR	REVIEWER	MANAGING DEPARTMENT
Sybert Stroeve Digitally signed by Sybert Stroeve Date: 2021.06.29 15:17:41 +02'00'	Yuk Shan Cheung Digitally signed by Yuk Shan Cheung Date: 2021.06.29 16:24:46 +02'00'	Alex Rutten Digitally signed by Alex Rutten Date: 2021.06.29 16:03:39 +02'00'

Contents

Abbreviations 5

Glossary of terms 7

1 Introduction 9
1.1 D(emo)-CRAT objective and structure 9
1.2 Objective and structure of this report 10

2 Context and motivation 11
2.1 Safe drone operations 11
2.2 DAA and collision avoidance systems 12
2.3 Modelling and simulation of UAS 14
2.4 Motivation: agent-based dynamic risk modelling for drone risk assessment 15

3 Agent-based model and MC simulation 17
3.1 Urban area and mission types 17
3.2 Environment of drone operations 18
3.3 Operator and flight planning 20
3.4 Aircraft and flight performance 20
3.5 Flight management system 21
3.6 Aircraft CNS systems 22
3.7 CNS region 22
3.8 DAA system 22
3.9 Remote pilot station 23
3.10 Pilot in command 23
3.11 Variability and failure modes in the agent-based model 23
3.12 Monte Carlo simulation methods 24

4 D(emo)-CRAT software 26
4.1 Software architecture 26
4.2 User experience 28
4.2.1 Configuration settings 28
4.2.2 Simulation mode 29
4.2.3 Visualization of results 30

5 Illustrative simulation and risk results 33
5.1 Mission types results 33
5.2 IPS, risk decomposition and computational load 36
5.3 Sensitivity analysis 40
5.3.1 DAA zones 40
5.3.2 Flight level planning, airspace configuration and traffic density 41
5.3.3 PIC response to DAA 43

6 Recommendations 45

6.1 Extended HMI 45

6.2 Conflict management functions 46

6.3 Extended DAA systems 47

6.4 Other types of risk 49

6.5 Extended modelling 50

6.6 Generic tool for agent-based dynamic risk modelling and simulation 51

7 Conclusions 53

8 References 55

Appendix A Generic tool for agent-based dynamic risk modelling and simulation 59

Appendix A.1 Overview of the initiative 59

Appendix A.2 Using the toolset for agent-based modelling and simulation 61

Appendix A.3 Example local Petri net in extended toolset 66

Abbreviations

ACRONYM	DESCRIPTION
ACAS	Airborne Collision Avoidance System
ADS-B	Automatic Dependent Surveillance - Broadcast
ATC	Air Traffic Control
ATM	Air Traffic Management
BVLOS	Beyond Visual Line Of Sight
C2	Command and Control
CA	Collision Avoidance
CNPC	Command and Non-Payload Communications
CNS	Communication Navigation Surveillance
CP	Contingency Plan
CR	Collision Risk
CS	Control Station
DAA	Detect And Avoid
DAIDALUS	Detect and Avoid Alerting Logic for Unmanned Systems
D-CRAT	Drone Collision Risk Assessment Tool
D(emo)-CRAT	Demonstrator Drone Collision Risk Assessment Tool
DLL	Dynamic Link Library
DOF	Degree Of Freedom
DWC	DAA Well Clear
EASA	European Union Aviation Safety Agency
EUROCONTROL	European Organisation for the Safety of Air Navigation
EVLOS	Extended Visual Line Of Sight
FAA	Federal Aviation Administration
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
GSHP	Generalised Stochastic Hybrid Process
GSHS	Generalised Stochastic Hybrid System
GUI	Graphical User Interface
HMI	Human-Machine Interface
IPS	Interacting Particle System
JARUS	Joint Authorities for Rulemaking of Unmanned Systems
MAC	Mid-Air Collision
MALE RPAS	Medium Altitude Long Endurance Remotely Piloted Aircraft System
MASPS	Minimum Aviation System Performance Standards
MC	Monte Carlo
MOPS	Minimum Operational Performance Standards

ACRONYM	DESCRIPTION
NASA	National Aeronautics and Space Administration
NLR	Royal Netherlands Aerospace Centre
NMAC	Near Mid-Air Collision
OS	Operating System
OSED	Operational Services and Environment Definition
PIC	Pilot In Command
R&D	Research & Development
RPA	Remotely Piloted Aircraft
RPAS	Remotely Piloted Aircraft System
RPS	Remote Pilot Station
RWC	Remain Well Clear
SESAR	Single European Sky ATM Research
STM	Surveillance and Tracking Module
sUAS	Small Unmanned Aircraft System
TRM	Threat Resolution Module
UA	Unmanned Aircraft
UAM	Urban Air Mobility
UAS	Unmanned Aircraft System
UAV	Unmanned Aerial Vehicle
UTM	UAS Traffic Management
VLL	Very Low Level
VLOS	Visual Line Of Sight
VTOL	Vertical Take-Off and Landing

Glossary of terms

TERM	DEFINITION
Advisory	An alert for conditions that require flightcrew awareness and may require subsequent flightcrew response for collision avoidance [1].
Airborne Collision Avoidance System (ACAS)	An avionics system onboard aircraft that performs collision avoidance [1].
Beyond Visual Line-of-Sight (BVLOS)	Operations where the pilot is not capable of using his or her vision to determine the location or orientation of the UA, hazards in the airspace, or potential of the UA to endanger life or property of another [2].
C2 Link (Command & Control Link)	The data link between the remotely piloted aircraft and the remote pilot station for the purposes of managing the flight [3]. This is equivalent to Control and Non-Payload Communications (CNPC) Link in [2]. The C2 Link is the logical connection, however physically realised, used for the exchange of information between the remote pilot station (RPS) and the remotely piloted aircraft (RPA). It enables the remote pilot's manipulation of the flight controls in the RPS to be sent to the RPA and for the RPA to return its status to the remote pilot. The C2 Link also enables the remote pilot to manage the safe integration of the remotely piloted aircraft system into the global aviation, communications, navigation and surveillance operational environment [3].
Collision Avoidance (CA)	A method of conflict management, whereby aircraft in imminent risk of colliding are given traffic situation awareness and possibly manoeuvring guidance to avoid a mid-air collision [1].
Collision Avoidance System (CAS)	The combination of aircraft, pilot (when pilot is present), avionics, and procedures working together to perform the function of collision avoidance [1].
Control Station (CS)	The equipment used to command, communicate with, or otherwise pilot an unmanned aircraft [2]. This is equivalent to a remote pilot station (RPS).
Detect and Avoid (DAA)	A means to surveil traffic and remain a safe distance from nearby aircraft so as to not create a collision hazard [1].
DAA Corrective Alert	A caution-level aural and visual annunciation intended to draw immediate pilot attention to traffic and make the pilot aware that action may be needed [1].
DAA Warning Alert	A warning-level aural and visual annunciation intended to notify the pilot that immediate awareness and immediate action is required to Remain Well Clear [1].
DAA Well Clear	A temporal and/or spatial boundary around an aircraft intended to be an electronic means of avoiding conflicting traffic [1].
Directive Guidance	A specific recommended resolution to avoid a hazard with manual or automated execution. An algorithm informs the pilot when and how to perform a recommended manoeuvre [1].
Interacting Particle System	Continuous-time Markov jump processes describing the collective behaviour of stochastically interacting components. They can be effectively applied for acceleration of rare event estimation.
Pilot In Command (PIC)	The person who has the final authority and responsibility for the operation and safety of flight [2].

TERM	DEFINITION
Near Mid-Air Collision (NMAC)	Two aircraft simultaneously coming within 100 ft vertically and 500 ft horizontally [1].
Remain Well Clear (RWC)	The ability to detect, analyse and manoeuvre to avoid potential conflicting traffic by applying adjustments to the current flight path in order to prevent the conflict from developing into a collision hazard [1].
Resolution Advisory (RA)	The combination of alerting and guidance given to a pilot that recommends a vertical or horizontal manoeuvre to either increase or maintain the existing vertical or horizontal separation relative to an intruding aircraft. The term 'Resolution Advisory' is equivalent to a 'DAA Warning Alert plus directive guidance' in RTCA/DO-365 [1].
Suggestive Guidance	A range of potential resolution manoeuvre provided in order to avoid a hazard with manual execution. An algorithm provides the pilot with manoeuvre decision aiding regarding advantageous or disadvantageous manoeuvres [1].
Traffic Advisory (TA)	Information given to the pilot pertaining to the position of another aircraft in the immediate vicinity. The information contains no suggested manoeuvre [1].
Unmanned Aircraft (UA)	An aircraft operated without the possibility of direct human intervention from within or on the aircraft [1].
Unmanned Aircraft System (UAS)	An unmanned aircraft and associated elements (including communication links and the components that control the UA) that are required for the remote PIC to operate safely and efficiently [2].

1 Introduction

1.1 D(emo)-CRAT objective and structure

Assuring safe integration of Unmanned Aircraft (UA) in all airspace classes is an important element of European research in the SESAR program. A central element in the European development is U-space, which is a set of new services and specific procedures designed to support safe, efficient and secure access to airspace for large numbers of drones[1]. Safety risk assessment of drone operations and the impact of the availability of U-space services on the safety risk are essential for their introduction.

For quantitative assessment of drone collision risks there is a strong need for simulation approaches that can represent a variety of drone operations and the types of uncertainties and hazards that can affect drone operations. These simulations should be able to achieve results up to the level of collision risk, meaning that they must support rare event estimation.

This project has developed a demonstrator tool for modelling and Monte Carlo (MC) simulation of drone traffic, called D(emo)-CRAT (Demonstrator Drone Collision Risk Assessment Tool). The Monte Carlo simulation uses acceleration by Interacting Particle Systems (IPS) and risk decomposition to efficiently achieve results up to the level of collision risk. The purpose of the demonstrator tool is to show that useful collision risk results can be achieved by the modelling and simulation, and as such may set the scene for further development of a full scale Drone Traffic Collision Risk Assessment Tool (D-CRAT).

The D(emo)-CRAT project is organised along the following tasks [4]:

1. Requirements for the Demonstrator Drone Collision Risk Assessment Tool
2. Model specification of element I (Drone traffic generator)
3. Model specification of element II (Drone Operations)
4. Model specification of element III (Monte Carlo simulation accelerator)
5. Implementation/testing of the Demonstrator Drone Collision Risk Assessment Tool
6. Demonstration
7. Project management

The results of the D(emo)-CRAT project are documented in the following reports:

- D(emo)-CRAT Requirements, which describes the missions, types of drones, U-space services, and hazards that are to be included in the agent-based modelling and MC simulation of drone operations, as well as D(emo)-CRAT software environment and the high-level human machine interface [5].
- D(emo)-CRAT Model Specification, which describes the agent-based model of the drone operations, including uncertainty and hazards, and MC simulation techniques, incorporating Interacting Particle Systems (IPS) and risk decomposition [6].
- D(emo)-CRAT Software Design, which describes the design of the backend and frontend of the D(emo)-CRAT software [7].
- D(emo)-CRAT User Manual, which describes how to use the D(emo)-CRAT software [8].

[1] The terms 'drones' and 'UA' are used interchangeably in this context.

- D(emo)-CRAT Demonstration and Final Report, which demonstrates the application of the D(emo)-CRAT software to a illustrative use case south of Paris and provides recommendations for future developments [9] (this report).

1.2 Objective and structure of this report

This is the final report of the D(emo)-CRAT project. The objective of this report is to provide an overview over the modelling and the tool development pursued, to demonstrate application of the tool to a use case for drone and air taxi traffic in an urban area south of Paris, and to give recommendations for future development of the tool.

This report describes the results towards this objective along the following structure:

- Section 2 describes the context and motivation for the development of D(emo)-CRAT.
- Section 3 describes the agent-based model and the rare event Monte Carlo simulation approach.
- Section 4 describes the design and user experience of the D(emo)-CRAT software.
- Section 5 presents illustrative simulation and risk results for a use case for drone traffic south of Paris.
- Section 6 presents recommendations for further development towards a full-scale Drone Traffic Collision Risk Assessment Tool.
- Section 7 presents conclusions of the project.
- Section 8 provides a list of references.

2 Context and motivation

2.1 Safe drone operations

Assuring safe integration of Unmanned Aircraft (UA) in all airspace classes is an important element of European research in SESAR [10]. A central element in the European development is U-space [11], which is a set of new services and specific procedures designed to support safe, efficient and secure access to airspace for large numbers of drones. These services rely on a high level of digitalisation and automation of functions, whether they are on board the drone itself, or are part of the ground-based environment. U-space provides an enabling framework to support routine drone operations, as well as a clear and effective interface to manned aviation, ATM/ANS service providers and authorities. It addresses the needs to support all types of missions (including VLOS and BVLOS operations) and may concern all drone users (commercial, leisure, state, military) and categories of drones. A comprehensive overview of SESAR exploratory research projects on U-space is provided in [12]. A key result is the U-space Concept of Operations as developed in the CORUS project [13-15]. It defines various U-space services, which are envisioned to be gradually implemented over three development phases U1 to U3. Future R&D needs for drone operations identified in [12] include urban air mobility (UAM), air traffic management (ATM)/U-space convergence, and advanced U-space services and technologies (e.g. miniaturisation and automated detect and avoid capabilities).

As part of NextGen research, the Federal Aviation Administration (FAA) developed a Concept of Operations for UAS Traffic Management (UTM) [16]. It focuses on very low level operations (below 400 ft AGL), but also addresses operations in other airspaces. In [17] FAA provides a ConOps for UAM, which describes roles and responsibilities and key elements of architectures and scenarios.

EASA uses a regulatory framework for UAS operations [18], including three types of categories: Open, Specific and Certified. The Open category involves a combination of limitations, operational rules, requirements for the competency of the remote pilot, as well as technical requirements for UAS, such that the UAS operator may conduct the operation without prior authorisation by the competent authority, or without submitting a declaration. The Specific category involves a risk assessment being conducted by the UAS operator before starting an operation, or an operator complying with a standard scenario, or an operator holding a certificate with privileges. The Certified category considers a category of UAS operation that, considering the risks involved, requires the certification of the UA (based on appropriate airworthiness standards, comparable to manned aviation) and its operator, as well as licensing of the flight crew. A safety risk analysis of UAS operations based on occurrence data by EASA [19] indicates that key risk areas include airborne conflicts, aircraft upset, system failures, and third party conflict.

Safety risk criteria for drone operations are considered by EUROCAE [20] and in a study by the National Academy of Sciences in an advice to the FAA [21].They advise that it is necessary to consider the increase in risk to people in manned aircraft and on the ground as well as safety benefits, and provide specific recommendations for the development of UAS categorisation, quantitative probability requirements, severity definitions, fail-safe criteria, and assurance levels. Approaches for safety risk management of UAS in [22] are based on standard risk management techniques applied to the UAS context. The application of barrier bow tie models for safety cases of UAS operations and the consequences of mid-air collisions are presented in [23, 24]. The Joint Authorities for Rulemaking of Unmanned Systems (JARUS) developed guidelines for safety risk assessment of Specific Operations, which focus on risks for third parties in the air, on the ground, and critical infrastructure [25].

In the recent Strategic Research and Innovation Agenda “Digital European Sky” [26] it is expressed that for safety assurance of U-Space and urban air mobility: “New safety modelling and assessment methodologies applicable to U-space are needed. Tools are required to analyse and quantify the level of safety of U-space operations involving high levels of automation and autonomy, where multiple actors automatically make complex, interrelated decisions under uncertainty (e.g. weather-related uncertainty). Research is needed to ensure that the distributed decision-making protocols implemented in U-space achieve the required level of safety while catering for differing levels of experience of participants. Examples of approaches that could be leveraged for this purpose include greater use of simulation and machine learning applications such as stress-testing.”

2.2 DAA and collision avoidance systems

Detect And Avoid (DAA) systems and Collision Avoidance Systems (CAS) are important constituents for safety of aircraft operations. For manned aviation, Traffic Alert and Collision Avoidance System II (TCAS II) [27] is the current airborne safety net used in commercial operations and ACAS Xa [28] is a next generation system. Both these systems provide advisories to the pilot to detect and vertically manoeuvre around nearby intruders. For UAS, various DAA systems have been developed and are being developed. DAA systems may include a Remain Well Clear (RWC) function as well as a collision avoidance function, but they may also include only one of these functionalities [29].

MOPS for DAA of larger drones flying at higher altitudes have been developed in RTCA DO-365 [2]. Here the DAA system environment consists of the unmanned aircraft (UA), the associated remote pilot station (RPS), intruder aircraft, navigation systems, Global Navigation Satellite Systems (GNSS) and communication systems. Onboard the UAS, the DAA system includes othership measurements and ownship state estimation for navigation and surveillance, which provide input for the DAA alerting and guidance processing. Command and Non-Payload Communications (CNPC) by data link packages enable a Pilot in Command (PIC) in the RPS to maintain situation awareness of the traffic situation and control the UAS. The basic DAA functionality of [2] includes a RWC functionality, which gives guidance to PICs to maintain well-clear and to regain separation in case of a well-clear violation. It does not include a collision avoidance functionality.

A reference implementation of DAA logic in Appendix G of [2] is DAIDALUS (Detect and Avoid Alerting Logic for Unmanned Systems) developed by NASA [30]. The DAIDALUS software library is released under the NASA Open Source Agreement at https://github.com/nasa/wellclear in Java and C++. The software implementation has been verified against the formal models and validated against multiple stressing cases jointly developed by the US Air Force Research Laboratory, MIT Lincoln Laboratory, and NASA [30]. DAIDALUS includes the following algorithms:

- *Detection*: Determine the current, pairwise well-clear status of the ownship and all aircraft inside its surveillance range. DAIDALUS does not include sensor processing functionalities (filtering, data fusion), but assumes the availability of 3D position and speed of ownship and otherships.

- *Determine Processing*: Compute manoeuvre guidance in the form of ranges of manoeuvres that a pilot-in-command may take that will cause the aircraft to maintain or increase separation from the well-clear violation volume, or allow for recovery from loss of separation in a timely manner within the performance limits of the ownship aircraft.
- *Alerting Logic*: Determine the corresponding alert type, based on a given alerting schema, corresponding to the level of threat to the well-clear volume.

The algorithms are rule-based and can be tuned by adaptation of a set of configuration parameters (see also Section 9.2 of [6]). DAIDALUS provides the RWC functionality only, but it has no CA functionality.

As part of the ACAS X development program [31, 32], ACAS Xu is a version dedicated to large UAS [1, 33] in line with RTCA DO-365. ACAS Xu includes both RWC and CA functionalities, meaning that it provides RWC warning alerts, recovery guidance to regain well-clear, and resolution advisories for collision avoidance. Its architecture is composed of a Surveillance and Tracking Module (STM), which provides filtering and fusion of sensor data for ownship and otherships, and a Threat Resolution Module (THM), which provides RWC and CA alerts/guidance/advisories on the basis of an off-line optimised logic table.

A comparison between DAIDALUS and an early version of ACAS Xu is presented in [34]. Results indicate comparable timelines and outcomes between ACAS-Xu’s RWC alert and guidance and DAIDALUS’s corrective alert and guidance, although ACAS-Xu’s guidance appears to be more conservative. ACAS-Xu’s CA alert and guidance occurs later than DAIDALUS’s warning alert and guidance, and overlaps with DAIDALUS’s timeline of manoeuvre to RWC.

In support of the development of MASPS and MOPS for DAA systems of UAS operating at Very Low Level (VLL), an Operational Services and Environment Definition (OSED) has been published [35]. In the OSED, VLL is defined as the airspace where it is reasonable to expect manned aviation will not operate except by permission from the competent authority, e.g. up to 1000 ft above highest obstacles in an urban area, or up to 500 ft above highest obstacles outside of urban areas. A broad definition of DAA is adhered to, including detect and avoid to avoid conflicts/collisions with manned aircraft, unmanned aircraft, terrain, fixed and mobile obstacles, hazardous weather, and people and animals. UA capable of carrying passengers, operations in the vicinity of airports, and highly automated UAS with possibility of PIC intervention are included in the OSED.

As part of the ACAS X development program, ACAS sXu is a DAA capability for small UAS, which provides an autonomous and decentralised DAA capability against manned aircraft, UAS, and other sUAS [36]. In ACAS sXu, the RWC and CA functions have been combined in one level of alerting and guidance, with the separation volume scaled based on the intruder type. Simulation results show risk reductions by ACAS sXu for scenarios with manned aircraft and with high traffic density of sUAS. Also an initial implementation of an interface for obstacle awareness has been included, where obstacles are represented as stationary point intruders.

2.3 Modelling and simulation of UAS

Modelling and simulation is a prime means to study, assess and validate operational scenarios of UAS operations. Some literature on relevant tools and projects for UAS modelling and simulation is provided in this section.

An evaluation tool for low-altitude air traffic operations called "Flexible engine for Fast-time evaluation of Flight environments" Fe^3 is presented in [37]. The Fe^3 core simulation engine is composed of two main functions: trajectory generation and collision avoidance. The trajectory models are 6 degrees of freedom (DOF) and are differentiated in several types, such as quad-rotor, fixed-wing, and hybrid. Three types of algorithms for DAA and CA are included in the simulator:

- Trajectory-projection based, predicting an intruder's trajectory and identifying resolution with predefined manoeuvre rules, e.g. DAIDALUS [30];
- Off-line table based, using off-line optimised tables for resolution manoeuvres based on intruder's state estimates, e.g. ACAS Xu;
- Force field based, using attractive forces to stay on an original path and repulsive forces to avoid potential conflicts.

Fe^3 includes models for vehicle communication, such as Dedicated Short Range Communications (DSRC), and on-board sensors such as LIDAR and Echodyne radar; these models include uncertainty due to sensor noise and communication reliability. The wind model in Fe^3 uses a spatially discretised database with turbulence intensity/uncertainty associated with each location. The evaluation of the Fe^3 models is done using Monte Carlo simulation on a system of graphical processing units. Some examples of simulation results are presented in [37], e.g. for loss of separation as function of communication capabilities and traffic densities, and as function of wind and traffic density. A sensitivity analysis using Fe^3 simulations is presented in [38] for the impact of intruder position estimation accuracy, communication latencies, wind, and separation buffer. In [39] Fe^3 was used to evaluate UAS traffic complexity metrics.

Ren et al. [40] presented a framework for the development and validation of trajectory modelling and prediction methods for diverse types of sUAS under nominal environment and under a variety of potential hazards, including adverse environmental conditions, and vehicle and system failures. It is recognised that the computation of trajectories under the effect of system failures include multiple sources of uncertainty, such as noise, uncertain parameters and mode switches [41]. The trajectory computation is presented as a hybrid state model, which calculates aircraft trajectories in nominal and off-nominal conditions. A related sUAS categorization framework is presented in [42]. A related framework for validation and verification of sUAS trajectories is provided in [40]. In these studies no details of implemented models or simulation results are presented, rather this is referred to as future work.

The Metropolis project aimed to study the relationship between airspace structure and airspace capacity for urban air mobility. Traffic scenarios are defined for UAM in the metropolitan region of Paris [43]. Various designs of the urban airspace are defined in [44]:

- Full mix design, where all vehicles share the same airspace, without any structure or non-physical constraints;
- Layers design, where every altitude band corresponds to a heading range in a repeating pattern;
- Zones design, having different zones for different types of vehicles, speed ranges as well as global directions have been defined to aid the separation by the structure of the airspace;

- Tubes design, where different directions, speeds and vehicle types will use different tubes ensuring safety by separating potentially conflicting traffic

A range of metrics for evaluation of UAM designs are presented in [45], including complexity metrics, operational metrics, and environmental metrics. Simulation results for the various UAM designs are presented in [46]. The simulations were performed using the Traffic Manager software, which had been developed for investigation of ATM concepts. For the purpose of Metropolis, parameters of existing vehicle models in Traffic Manager were adapted to match the performance specifications available for several current PAVs and UAVs. Additionally, VTOL aircraft were simulated using helicopter dynamics for take-off and landing, while fixed-wing models were used for cruising, climbing and descending flight phases.

A modelling, simulation and control framework for small unmanned multicopter platforms in urban environments is presented in [47]. The vehicle dynamics modelling includes a state vector of 13 kinematic variables (position, velocity, quaternion orientation, angular velocity), dynamic forces and moments, aerodynamics, and propulsive forces. It is combined with a flight management and flight control system, which receives input from vehicle sensors, vision hardware, and LIDAR hardware.

2.4 Motivation: agent-based dynamic risk modelling for drone risk assessment

As recognised in Digital European Sky [26] there is a need for new safety modelling and assessment methodologies for U-space, which can quantify levels of safety of U-space operations. Such safety risk assessment of new concepts and technologies needs to assess (1) how effective the new concepts and technologies are if they work as intended, as well as (2) what risks are induced if elements in the new concepts and technologies are failing. In the SESAR Safety Reference Material for assessment of changes in ATM [48, 49] these two perspective are referred to as success approach and failure approach, respectively. For new drone and U-space supported operations this means that a broad range of questions needs to be answered for the safety assessment:

- *When systems are working as intended in normal conditions*. What is the effectiveness of DAA systems and how can they be tuned optimally? What is the impact of traffic density and airspace design? What is the impact of normal sensor errors? What is the impact of normal communication delays? What is the impact of normal human reaction times? What is the impact of normal variability in speeds of drone operations? What is the impact of normal weather variability? Etc.
- *When there are failures or off-nominal conditions.* What is the impact of failures of technical systems, including drone propulsion, communication systems, surveillance systems, navigation systems, DAA system? What is the effectiveness of mitigating measures for failure conditions? What is the impact of humans errors, such as errors in planning and reaction to DAA advisories? What is the impact of adverse weather conditions? Etc.

Obstacles in effective safety risk assessment of drone operations include the novelty of drone types, drone systems, and procedures for their operations. The lack of experience by operators and the lack of data on safety occurrences imply that safety risk assessment methods that largely depend on expert judgement and safety occurrence data tend to be quite limited in achieving valid risk assessment results. For quantitative assessment of drone collision risks there is a strong need for simulation approaches that can represent a variety of drone operations and the types of uncertainty and hazards that can affect drone operations.

These simulations should be able to achieve results up to the level of collision risk, meaning that they must support rare event estimation.

It has been shown that agent-based dynamic risk modelling for safety risk assessment in ATM is an effective method to assess the risk of complex and novel operations [50]. Such models represent the dynamics and stochastic variability of operations involving complex interactions of technical systems, human operators and environmental conditions, both in normal conditions and in off-nominal/failure conditions. The models are used in rare event Monte Carlo simulation approaches to assess low probabilities of safety events. Agent-based dynamic risk modelling is included in the SESAR Safety Reference Material [48, 49] and it has been effectively used in various applications, including runway incursions, airborne self-separation, separation minima of conventional operations, ACAS evaluation, and others [51-56]. The safety assessment results include probabilities of rare safety events (collision, close proximity), the conditions contributing to key risks, and the sensitivity for settings in the operational concept.

Building on the evidence base and successful application of agent-based dynamic risk modelling for safety assessment of air traffic operations, it will be used in this study for the development of D(emo)-CRAT, the demonstrator tool for drone traffic collision risk assessment. The agent-based model and MC simulation methods for the drone operations are presented in the next chapter.

3 Agent-based model and MC simulation

The basis of the simulations of drone and air taxi operations in D(emo)-CRAT is an agent-based model, which describes the operations, including the uncertainty, performance variability, failures of technical systems, human operator errors, and abnormal conditions of the environment/context that may influence them. The details of the agent-based model and the applied Monte Carlo simulation methods for risk assessment are described in [6]. A high-level overview of the model is provided below.

3.1 Urban area and mission types

D(emo)-CRAT uses a set of generic models of drone operations. For the demonstration an urban area south of Paris has been chosen for the specific instantiation of the models [5]. The specific environment is an area along the line from Aerodrome Toussus-le-Noble (LFPN) to Aerodrome Brétigny-sur-Orge (LFPY). In this environment there are the following five areas of interest (Figure 1):

1. Aerodrome Toussus-le-Noble (LFPN), which is a regional airport in France supporting mainly general aviation. It has two parallel runways (heading 07/25) with lengths of about 1100 m;
2. Area of Orsay, which is a suburb at 21 km from the centre of Paris with an area of 8.0 km^2 and a population of about 16,400 (source Wikipedia);
3. Area of Montlhéry, which is a suburb at 26 km from the centre of Paris with an area of 3.3 km^2 and a population of about 7,600;
4. Area of Brétigny-sur-Orge, which is a suburb at 27 km from the centre of Paris with an area of 14.6 km^2 and a population of about 26,500;
5. Aerodrome Brétigny-sur-Orge (LFPY), which is a former French Air force base that will be used for civil drone operations. It has two runways: 05/23 of 3000 m and 11/29 of 2200 m.

The straight-line distances from Area 1 to the other areas are about: Area 2 is at 9 km, Area 3 is at 17 km, Area 4 is at 22 km, and Area 5 is at 24 km from Area 1.

In this environment, the following types of missions are included:

A. Vertical take-off and landing (VTOL) urban air mobility for transport of people between the suburb areas 2, 3, and 4 (green lines in Figure 1). It is assumed that flights are only between the suburbs, there are no flights between locations within one suburb.
B. Multicopter drone operations within the suburb areas 2, 3, and 4, including loitering and surveillance.
C. Fixed-wing drones for parcel delivery between the airports 1 and 5 (blue line).
D. Fixed-wing drones flying level and crossing the area between the airports 1 and 5, e.g. for parcel delivery to and from Paris centre (red lines).

In Mission types C and D, the parts of the flight in which the fixed-wing aircraft ascend from their origin location or descend to their destination location on the ground are not in the scope of D(emo)-CRAT. In Mission types A and B, the ascent and descent flight phases are in the scope of D(emo)-CRAT.

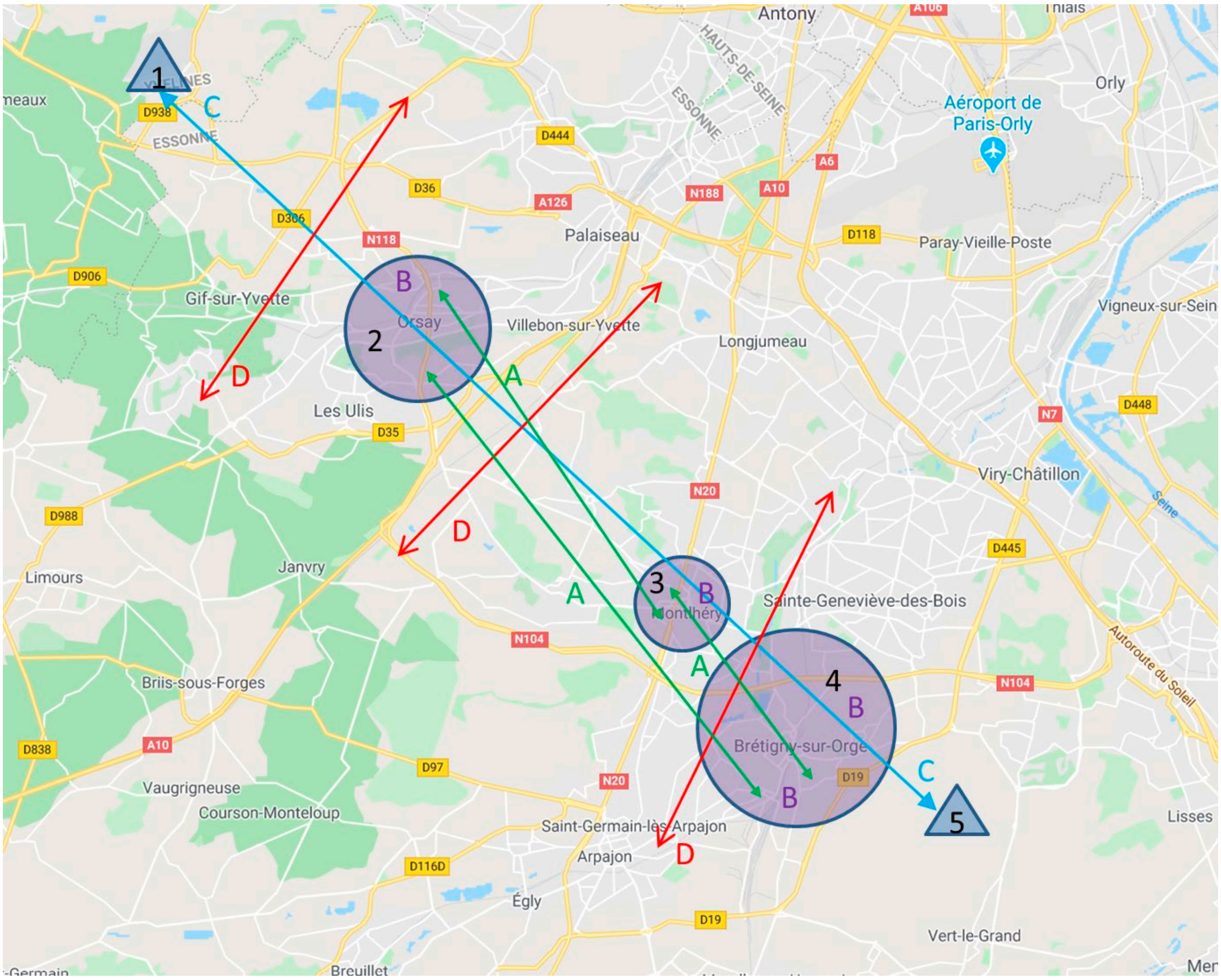


Figure 1: Map for use case of D(emo)-CRAT, including five areas (1-5), four mission types (A-D) and examples of missions (arrows).

3.2 Environment of drone operations

Urban areas and flight zones

A generic environment specification is provided that can represent the airports, urban areas, and entry/exit zones of drone operations such as in Figure 1. All terrain is flat and at zero altitude. Positions in the environment are provided in world geodetic coordinates and in a local tangent East-North-Up (ENU) coordinate frame; the latter is the prime coordinate frame used in the simulations.

Airspace design

Various airspace design options are included, which can be set by a user of D(emo)-CRAT for a simulation:

- *Free flight.* In this design, all aircraft and all missions use the same airspace without altitude restrictions.
- *Mission-dependent altitude layers.* In this design, there are altitude layers dedicated for types of missions. In particular, there are dedicated altitude layers for air taxies (mission A), surveillance and loitering (mission B), and for the fixed-wing drones (missions C and D). These altitude layers apply to

the main en-route part of the flight, meaning that during the climb and descend phases other altitude layers may be passed.

- *Mission and heading-dependent altitude layers*. In this design, there are altitude layers dedicated for types of missions and within these layers there may be heading-dependent sublayers. In particular, there are dedicated altitude layers for air taxies (mission A), surveillance and loitering (mission B), and for the fixed-wing drones (missions C and D). Within the layers for mission A and for mission C/D there are heading-dependent sublayers, which associate the route direction with an altitude layer. There are no such sublayers for mission B, since in surveillance and loitering the aircraft fly patterns with largely fluctuating headings.

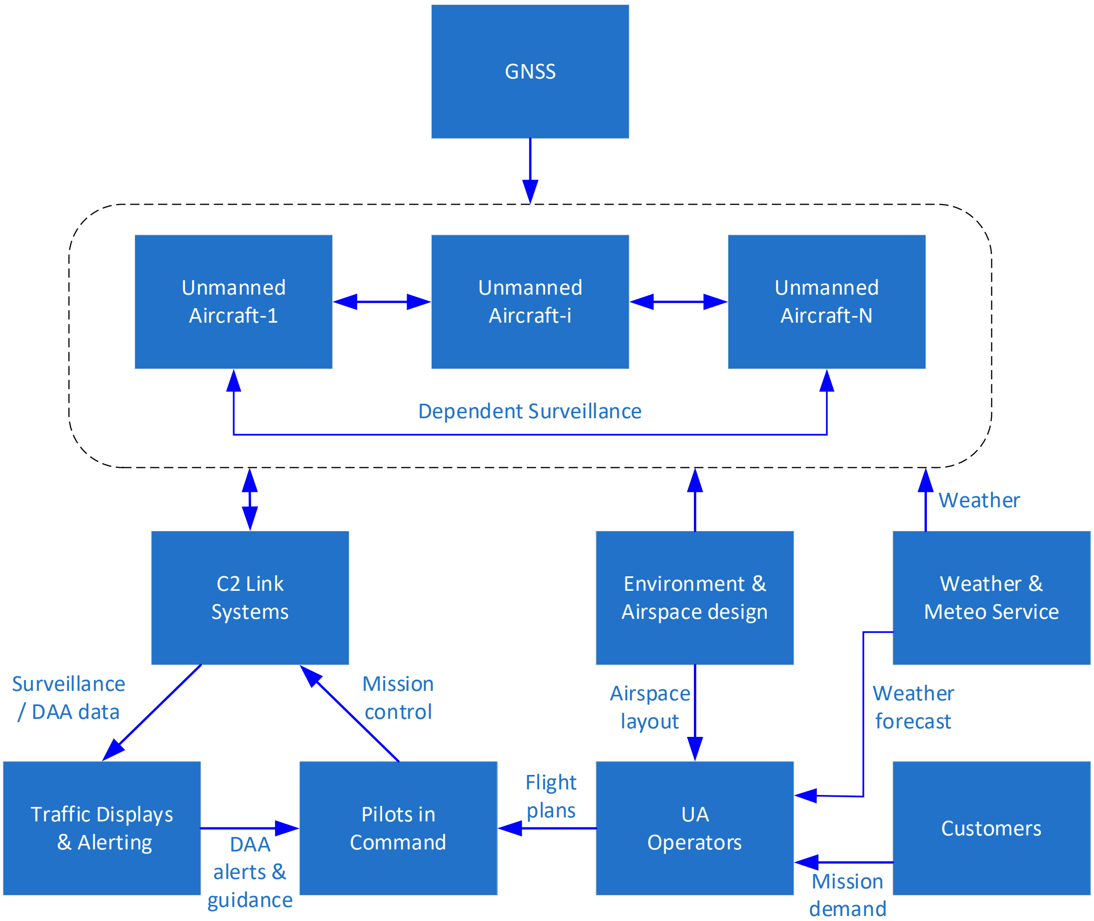


Figure 2: High-level overview of main interactions between agents influencing a number of UA.

Weather and meteorological services

The modelling of weather and meteorological services represents a constant and uniform wind that can be set and it includes a random possibility of an adverse weather condition in the region that is not predicted by the meteorological services. Two severity categories of adverse weather are considered, one having impact on VTOL operations (i.e. missions A and B) only, and one having impact on all operations. The consequence of adverse weather is that the drone trajectories exhibit oscillatory motion around a nominal flight path, thus representing turbulence.

Customers

The entity Customers represents the demand for services from the UAS operators for the mission types. This demand determines the traffic density in the simulations. The demand for the mission types has the following characteristics:

- *Mission A.* The demand for air taxi services between the urban areas is represented with a Poisson process between arbitrary positions in different urban areas.
- *Mission B.* The demand for surveillance and loitering is represented by a Poisson process for the starting time and by an exponentially distributed duration. The starting position is at an arbitrary location in an urban area.
- *Mission C*. The demand for fixed-wing drone operations between the airports is represented by a Poisson process between positions near the airports.
- *Mission D*. The demand for fixed-wing drone operations crossing the airspace is represented by a Poisson process between positions on the entry/exit edges.

3.3 Operator and flight planning

The operator performs the flight planning of the drone flights. The operator uses the airspace design and the customer demand as a basis for the flight planning. Flights are planned directly following the customer demand, there is no restriction in the number of available aircraft. The flight planning depends on the type of operation associated with the mission, namely air taxi for Mission A, surveillance & loitering for Mission B, and UAS en-route for Missions C and D. For Missions A, C and D the shortest routes between start and end points are planned, while for Mission B a start point and a flight duration is planned for manoeuvring within an urban area. The planning of flights sets a flight level that is between the altitude bounds of the airspace design for the type of operation considered. This can be done in two modes:

- *Middle*, planned flight level is exactly in the middle of the altitude bounds;
- *Random*, planned flight level is uniformly distributed between the altitude bounds.

The operator can make an altitude planning error, which implies that a flight is planned at an altitude layer above or below the layer according to the airspace design.

3.4 Aircraft and flight performance

A number of aircraft types are defined, which are associated to the operation types (air taxi, surveillance & loitering, and en-route UAS). The flight performance of the aircraft is specified, describing variables like position, heading, speed, and climb speed during manoeuvring. The input for the manoeuvring of the aircraft stems from the flight control system (which is a part of the flight management system), which uses mission control settings for various flight phases. The flight performance model includes non-nominal modes for uncommanded motion of the drone in adverse weather and for engine failure.

3.5 Flight management system

The flight management system (FMS) of the aircraft contains the flight plan, ownship and othership state estimates, output of the DAA system, and input from the Pilot In Command (PIC). This data is used for mission and flight control, as input of the DAA system, for ADS-B Out transmission to other aircraft, and to inform the PIC. Data transfer with the PIC is enabled by the C2 link. The mission control system uses settings by the PIC to control the various flight phases of nominal operations and to change the trajectories in response to DAA alerts and guidance. Furthermore the mission control system includes autonomous control modes for contingency plans in the case of a lost C2 link or in the case of lost GNSS ownship estimation. The mission control system sends commands to the flight control system for the control of heading, air speed and altitude.

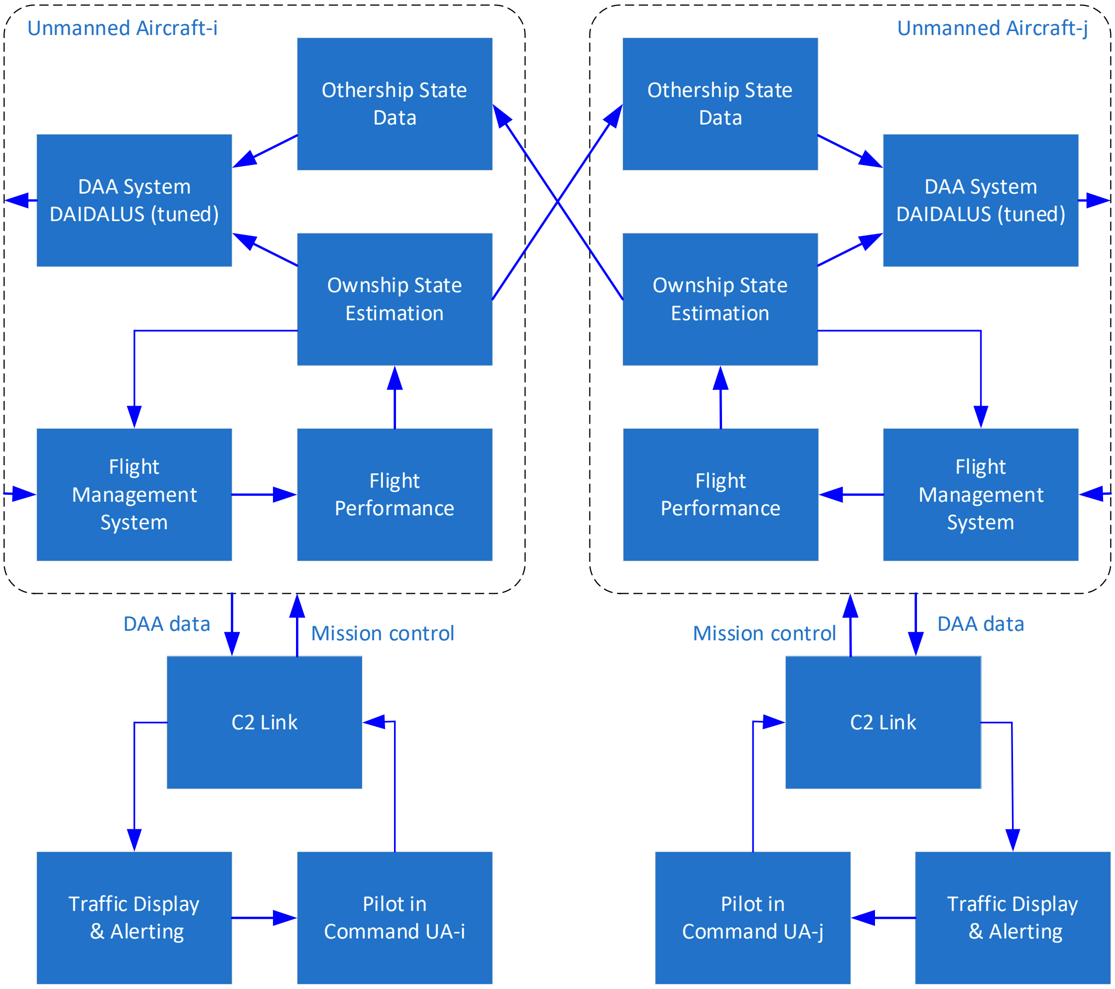


Figure 3: Main interactions between a pair of unmanned aircraft.

3.6 Aircraft CNS systems

Several models of aircraft Communication Navigation and Surveillance (CNS) systems are included in D(emo)-CRAT.

- A model for the pressure altimetry system is used, which includes modes for the altitude and vertical speed estimation in normal accuracy ranges and for reduced accuracy ranges. Altitude and vertical speed measurement errors are chosen from zero-mean Gaussian distributions with mode-dependent standard deviations.
- Horizontal position and speed estimation of the ownship are based upon GNSS data. The GNSS receiver on-board the aircraft includes modes for working or not. It includes modes for the GNSS-based state estimation in normal accuracy and reduced accuracy ranges. The modes are chosen randomly (with low probability for the non-nominal modes). Errors in the position and speed estimates are chosen from zero-mean Gaussian distributions with mode-dependent standard deviations.
- The C2 link is the logical connection used for the exchange of information between the remote pilot station (RPS) and the unmanned aircraft. A model for the availability of the C2 Link system at the aircraft is included, which represents the possibility of the system not being available. If C2 Link transmission is possible, durations for the uplink and the downlink of information are chosen from uniform distributions.
- ADS-B systems support the exchange of surveillance information between the aircraft. This surveillance information is a primary input source of the DAA system. A model for the availability of the ADS-B system at the aircraft is included, which represents the possibility of the system not being available. If ADS-B transmission is possible, its duration is chosen from a uniform distribution.

3.7 CNS region

CNS Region describes the functioning of the regional components of the C2 Link and the Global Navigation Satellite System (GNSS). The functioning of these components has influence on the navigation, communication and surveillance of all aircraft in the scenario. The primary role of these model components is to represent failure conditions of the C2 Link infrastructure or GNSS, in particular for the availability of the C2 Link and GNSS, and for the integrity of the GNSS-based state estimation. These modes are chosen randomly, based on user-specified probabilities.

3.8 DAA system

The Detect And Avoid (DAA) system is the prime means in D(emo)-CRAT to detect conflicts and to provide guidance and alerts for remaining well clear and avoiding collisions. D(emo)-CRAT integrates DAIDALUS (Detect and Avoid Alerting Logic for Unmanned Systems), which has been developed by NASA as a DAA reference system of the RTCA MOPS [2]. The alerting logic of DAIDALUS is rule-based, which has as major advantage that its alerting logic can be tuned to the operations considered in D(emo)-CRAT. The relationship between the DAIDALUS implementation and the surveillance data sources, separation standards, and pilot interface are described in detail in [6].

3.9 Remote pilot station

The Remote Pilot Station receives aircraft data and DAA data via the C2 Link and shows this data to the pilot in command (PIC) by traffic display and alerting (TDA). In the model the TDA includes post-processing of DAA data. In particular an M out of N filter is used to avoid false alerts. The model of the C2 Link of the RPS accounts for the possibility that the C2 Link system at the RPS is not working.

3.10 Pilot in command

The pilot in command (PIC) is the agent who sets the control of the various types of aircraft operations in nominal conditions and who is informed by the DAA system about conflicting aircraft and guidance to avoid close encounters. The PIC model consists of three components: (1) situation awareness model, (2) reaction to DAA alerts and guidance, and (3) nominal flight control settings.

PIC situation awareness model

The situation awareness of the PIC includes information of the ownship for nominal flight control actions, such as the flight plan, airspeed, altitude and heading. Furthermore, the situation awareness includes the interpretation of the DAA alert and guidance, such as the need to make changes in direction or altitude. The situation awareness updating by the PIC does not include errors, but the situation awareness may include erroneous/inaccurate information provided by others.

PIC reaction to DAA alerts and guidance

Based on the situation awareness, the PIC implements actions in response to the DAA alert and guidance information. Models for the following elements are included:

- PIC response mode, describing probabilities of response to horizontal and vertical guidance;
- Delay in the response to DAA alert and guidance information;
- PIC mission control actions, which describe the manoeuvring to selected directions/altitudes and returning back to the planned trajectory if there is no DAA alert remaining.

PIC nominal flight mission control settings

This part of the PIC model describes the settings for the control by the FMS of the various types of missions, namely the air taxi, surveillance and loitering, and UAS en-route operations. The models describe the choices of climb, descent and cruise speeds, acceleration and deceleration, and rate of turn during nominal operations.

3.11 Variability and failure modes in the agent-based model

As explained in Section 2.4, for safety risk assessment of operations it is needed to account both for the variability of operations in normal conditions and for failure modes that may occur. An overview of the normal variability and the failure modes in the agent-based model is presented in Table 1.

Table 1: Normal variability and failure modes represented in the D(emo)-CRAT agent-based model

Normal variability	Failure modes
Wind speed	Adverse weather not predicted
Customer demand: timing, locations	Wrong altitude in flight planning
Errors in altitude and vertical speed measurement	Engines failure
Errors in GNSS-based horizontal position and speed estimates	Reduced accuracy of pressure altimetry
Delay in ADS-B transmission (between drones)	GNSS-based estimation is not working (aircraft system or in whole region)
Delay in C2 link transmission (with RPS)	Reduced accuracy of GNSS-based estimation (single aircraft or whole region)
Delay in response to DAA alert by PIC	C2 Link not working (aircraft system, RPS, or whole region)
Rates of turn and climb/descent by PIC in DAA response	ADS-B system of aircraft is not working
Rates of climb, descent, turn, acceleration, deceleration, cruise speed during nominal flight	DAA system of aircraft is not working
	No/limited response to DAA alert by PIC

3.12 Monte Carlo simulation methods

The Monte Carlo (MC) simulation methods used for acceleration of the estimation of risk levels up to collisions between aircraft are Interacting Particle Systems (IPS) and risk decomposition.

IPS MC simulation

The IPS MC simulation uses a series of miss distance cycles $c = 1, \ldots, m$ with decreasing miss distance boundaries $d_c < d_{c-1}$, and a collision cycle. Each miss distance boundary d_c is composed of a horizontal distance r_c and a vertical distance h_c, which both have to be reached by a pair of drones for a miss distance boundary hit. For instance, if $r_c = 50\text{m}$ and $d_c = 15\text{m}$ then a miss distance boundary hit occurs if there is a pair of drones that comes closer than 50 m horizontally and 15 m vertically. In the collision cycle, a collision between a pair of drones occurs if cylinders encapsulating the drones overlap (here drones can have different sizes). In the IPS MC simulation, N simulation objects (or particles) of an air traffic scenario are executed over a finite time interval $(0, T)$ for m miss distance cycles and a collision cycle.

In each miss distance cycle, N particles are simulated and the cycle is ended if the simulation of each particle has ended, either because a particle has reached the next miss distance boundary, or because a particle has reached the end time of the simulation T . In miss distance cycle c , the number of simulation objects that hit miss distance boundary d_c is denoted as N_c^h and the fraction of miss distance boundary hits is $\gamma_c = N_c^h / N$. At the end of cycle c , independent copies are sampled from the N_c^h simulation objects that have reached d_c , such that there are N simulation objects that satisfy boundary condition d_c for continuing the simulation in the next cycle $c+1$. If during cycle c , no simulation object reaches d_c within (0,T), then $N_c^h = 0$, $\gamma_c = 0$ and the IPS MC simulation stops there. The probability of reaching the miss distance boundary at the end of a sequence of miss distance cycles d_m within a certain time T is estimated as

$$P\{\tau_m < T\} = \prod_{c=1}^{m} \gamma_c = \prod_{c=1}^{m} \frac{N_c^h}{N}.$$

For instance, consider an IPS MC with 1000 particles and two miss distance cycles, where the first cycle has a boundary at 100 m horizontally and 30 m vertically, and the second cycle has a boundary at 50 m horizontally and 15 m vertically. Now if in 50 of the simulations the first miss distance boundary is achieved, then $\gamma_1 = 50/1000 = 0.05$ and the complete state spaces of these 50 particles are resampled as a basis for the state spaces of the continuing simulation of 1000 particles at the start of the second miss distance cycle. Now if in 100 of these continued simulations the second miss distance boundary is achieved, then $\gamma_2 = 0.1$ and the complete state spaces of these 100 particles are resampled for the state spaces of the continuing simulation of 1000 particles at the start collision cycle. In combination the probability of attaining the second miss distance boundary is estimated as $P\{\tau_2 < T\} = \gamma_1 \gamma_2 = 0.005$.

In the collision cycle, N particles are simulated. Typically, the state space is based on resampling of the state space of particles that reached the last miss distance boundary d_m. However, a collision cycle may also be started without any miss distance boundary (i.e. for $m = 0$) and in this case the state space is based on the initial conditions of the scenario of the simulation. The collision cycle is ended if each particle has reached the end time of the simulation T. In the collision cycle, all collisions between drones are counted and drones involved in a collision are removed from the simulation. The collision cycle thus provides an estimate of the mean number of drone collisions. For instance, consider that in above example of 1000 particles and two miss distance cycles, there are 992 particles without a collision, 6 particles with a single collision, and 2 particles with collisions of 2 aircraft pairs. For these numbers the mean number of collisions per particle, given the start at the second miss distance boundary, is $(6+2\times 2)/1000 = 0.01$. In combination with the probability of attaining the second miss distance boundary, the estimate of the mean number of collisions per particle is $0.005 \times 0.01 = 5 \cdot 10^{-5}$.

Risk decomposition

A risk decomposition may be applied in support of the assessment of rare global failures. Herein the risk $R(d)$ is decomposed in a component $R_{NGF}(d)$ without any global failure modes and a component $R_{GF}(d)$ with one or more global failure modes: $R(d) = R_{NGF}(d) + R_{GF}(d)$. We will explain the risk decomposition for an agent-based model with global systems with modes $\kappa_{t,G,i} \in \{W, F\}$ with $i = 1 \ldots N_G$, where *W* denotes that the system is working and *F* denotes that the system is failing. We denote the number of global systems that are in a failure mode as n_t^{GF}. With this, above equation can also be written as $R(d) = R(d \mid n_t^{GF} = 0) P(n_t^{GF} = 0) + R(d \mid n_t^{GF} > 0) P(n_t^{GF} > 0)$. For rare failure conditions the term $P(n_t^{GF} = 0) \approx 1$ and the term $P(n_t^{GF} > 0)$ is some low value determined by the failure probabilities. The term $R(d \mid n_t^{GF} = 0)$ is the risk given that there are no global failure conditions. It is evaluated by IPS MC simulation with all global systems set as working, i.e. $\forall i : \kappa_{t,G,i} = W$. The term $R(d \mid n_t^{GF} > 0)$ is the risk given that there are one or more global failure conditions. It is evaluated by a set of IPS MC simulations with one or more global systems set as failing, i.e. $\exists i : \kappa_{t,G,i} = F$.

4 D(emo)-CRAT software

D(emo)-CRAT software has been developed for simulation of the agent-based models highlighted in Section 3. The software architecture is introduced in Section 4.1. The user experience is presented in Section 4.2.

4.1 Software architecture

The D(emo)-CRAT software is conceptually decoupled into two main blocks, a Frontend and a Backend. The Backend handles the simulation process by applying the models described in [6]. The Frontend provides an interface for the user to visualise and interact with the simulation configuration and results. This high level architecture is depicted in Figure 4. Details of the software design are presented in [7].

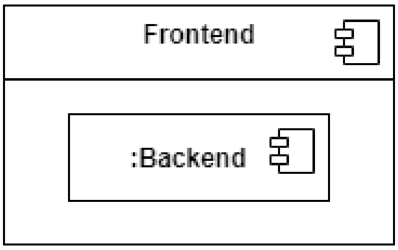


Figure 4: Highest level architectural design.

The Backend provides the whole set of functionalities that are needed for the execution of the D(emo)-CRAT simulation models. This includes:

- Data structures containing the parameters of a simulation (as well as validation functions for these parameters);
- Creation and execution of simulation objects that implement the agent-based modelling approach described in [6], as well as access to simulation results;
- Functions for interaction with more complex simulation constructs (e.g. IPS) that make use of simple simulation objects.

The Backend has been developed almost exclusively from the C++ standard library. Exceptions are the functions for parsing/writing of JSON files, which is based on Rapidjson. The Backend encapsulates a C++ implementation of DAIDALUS, which was developed by NASA and made available on GitHub [57].

The Frontend acts as an interface between the user and the Backend, as shown in Figure 5. It provides methods for definition of simulations, interaction with simulations and visualisation of the results (see use cases in Table 2). The graphics development environment used for the D(emo)-CRAT GUI is the C++ QT framework as adopted in [5]. The GUI low-level widgets (menus, buttons,…) were generated using QT base widgets. Higher level widgets (map, work tree) were generated with custom graphics classes. Finally, non-graphics classes use QT libraries as well as pure C++ code. The GUI is based on the QApplication QT class. In particular, all the graphics widgets communication are based on the QApplication loop and its signal/slot communication mechanism.

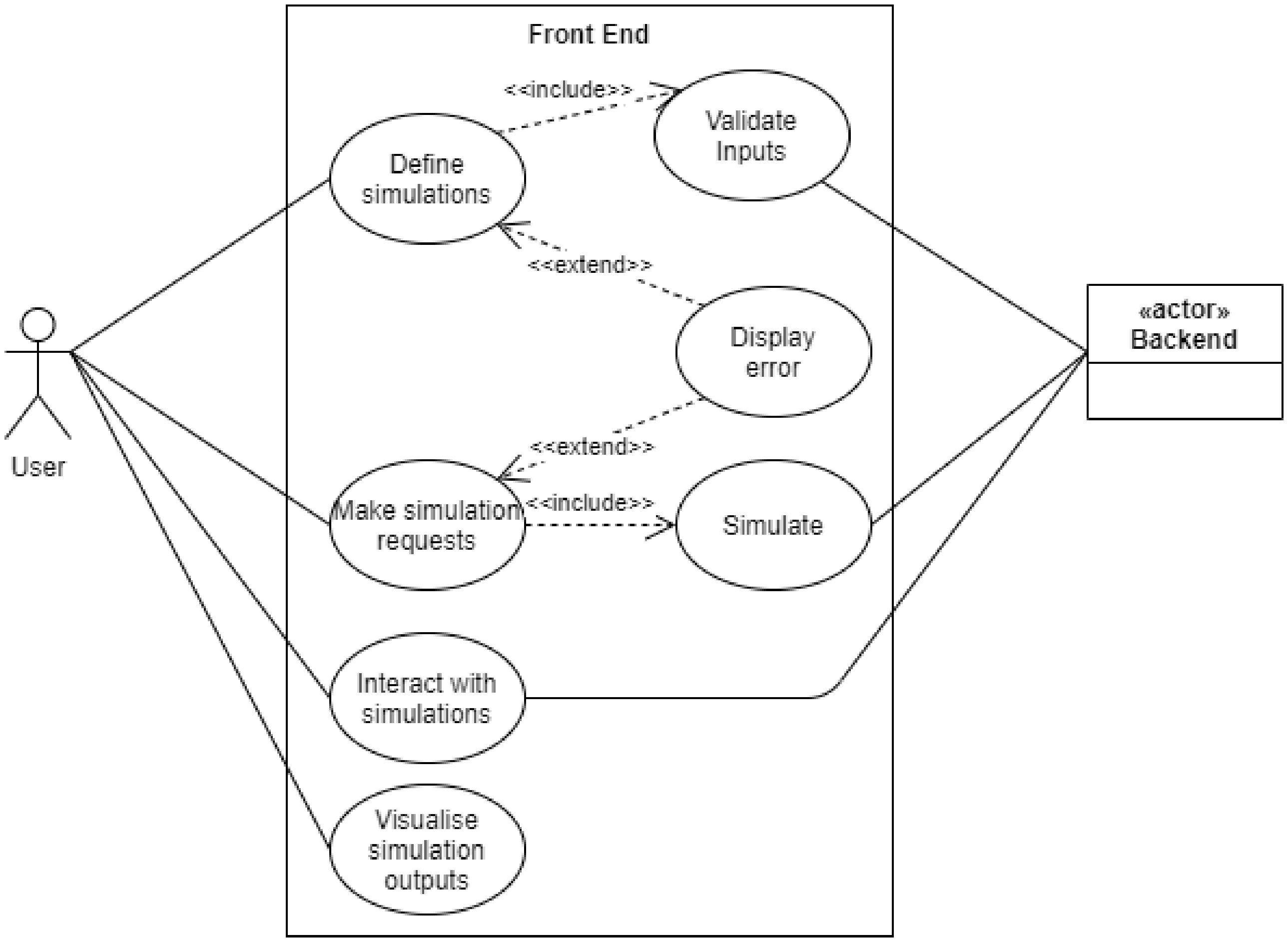


Figure 5: Use-case diagram for the Frontend.

Table 2: Use cases of the D(emo)-CRAT Frontend

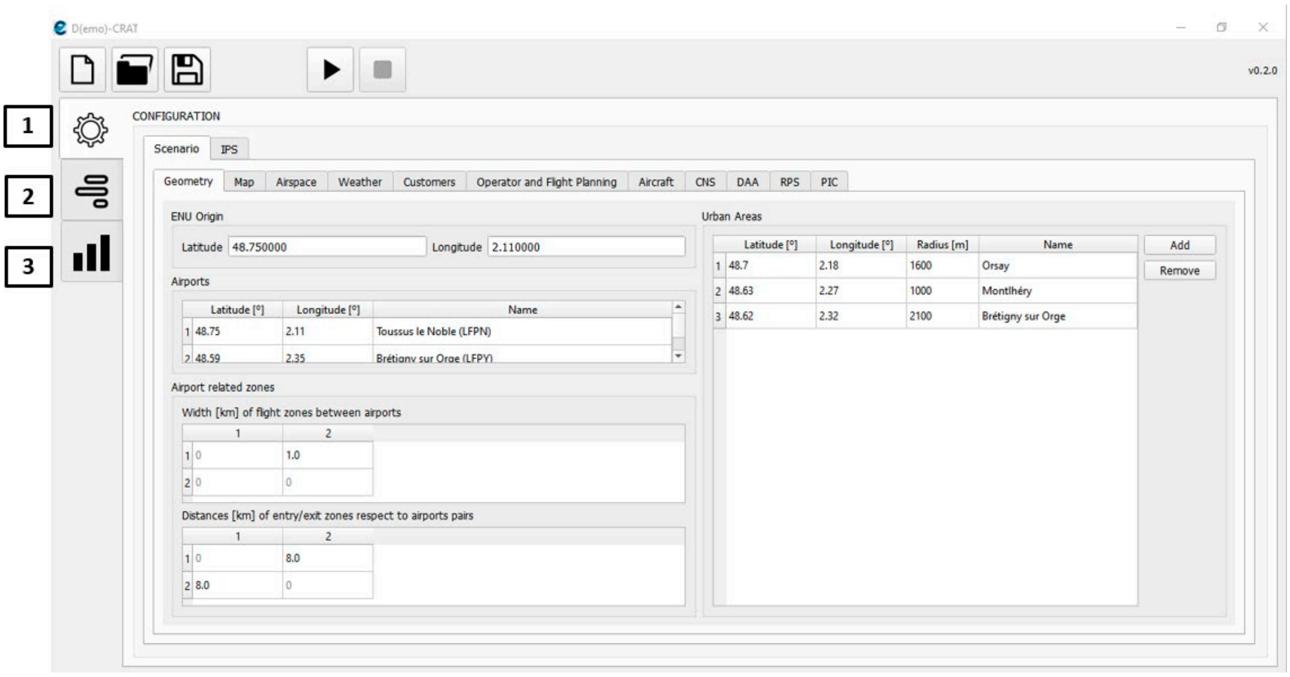
Use Case	Description
Define simulations	Describes the interaction of the user with the Frontend for the configuration of the parameters of a simulation. For these, there exists a number of restrictions that must be met, which are verified through use case *Validate Inputs*. This validation is done through the functionalities exposed by the Backend. If validation fails, an error message is displayed.
Make simulation requests	Describes the request by a user to initiate a simulation. This use case makes a request to the Backend to run the simulation models (use case *Simulate*). In the event of failure, the user is warned by an error message.
Interact with simulations	Represents user interaction during the time when the Backend is simulating. This includes visualisation of the simulation progress, the ability to stop a simulation in progress, or interaction with simulations with the IPS.
Visualise simulation outputs	This use case is associated to the user visualising and interpreting the simulation results provided by D(emo)-CRAT.

4.2 User experience

This section gives an introduction on the user experience of D(emo)-CRAT; a detailed user manual is provided separately in [8].

The main window of D(emo)-CRAT (Figure 6) presents of three modes of operation:

1. **Configuration**: Allows the user to define a simulation by setting the parameters of the agent-based models (e.g. define urban areas and flight zones, set customer demand, etc.), as well as to set the parameters of the IPS MC simulation.
2. **Simulation**: Allows the user to interact with the progress of a simulation. The GUI provides feedback of the current simulation status (execution progress and partial results). If a simulation finishes, a simulation folder with the results is created.
3. **Visualization**: Allows the user to visualise the results of a completed simulation. The GUI provides tools for loading and analysing the simulation results for quantitative risk assessment.



ENU Origin

Latitude 48.750000 Longitude 2.110000

Airports

	Latitude [°]	Longitude [°]	Name
1	48.75	2.11	Toussus le Noble (LFPN)
2	48.59	2.35	Brétigny sur Orge (LFPY)

Airport related zones

Width [km] of flight zones between airports

	1	2
1	0	1.0
2	0	0

Distances [km] of entry/exit zones respect to airports pairs

	1	2
1	0	8.0
2	8.0	0

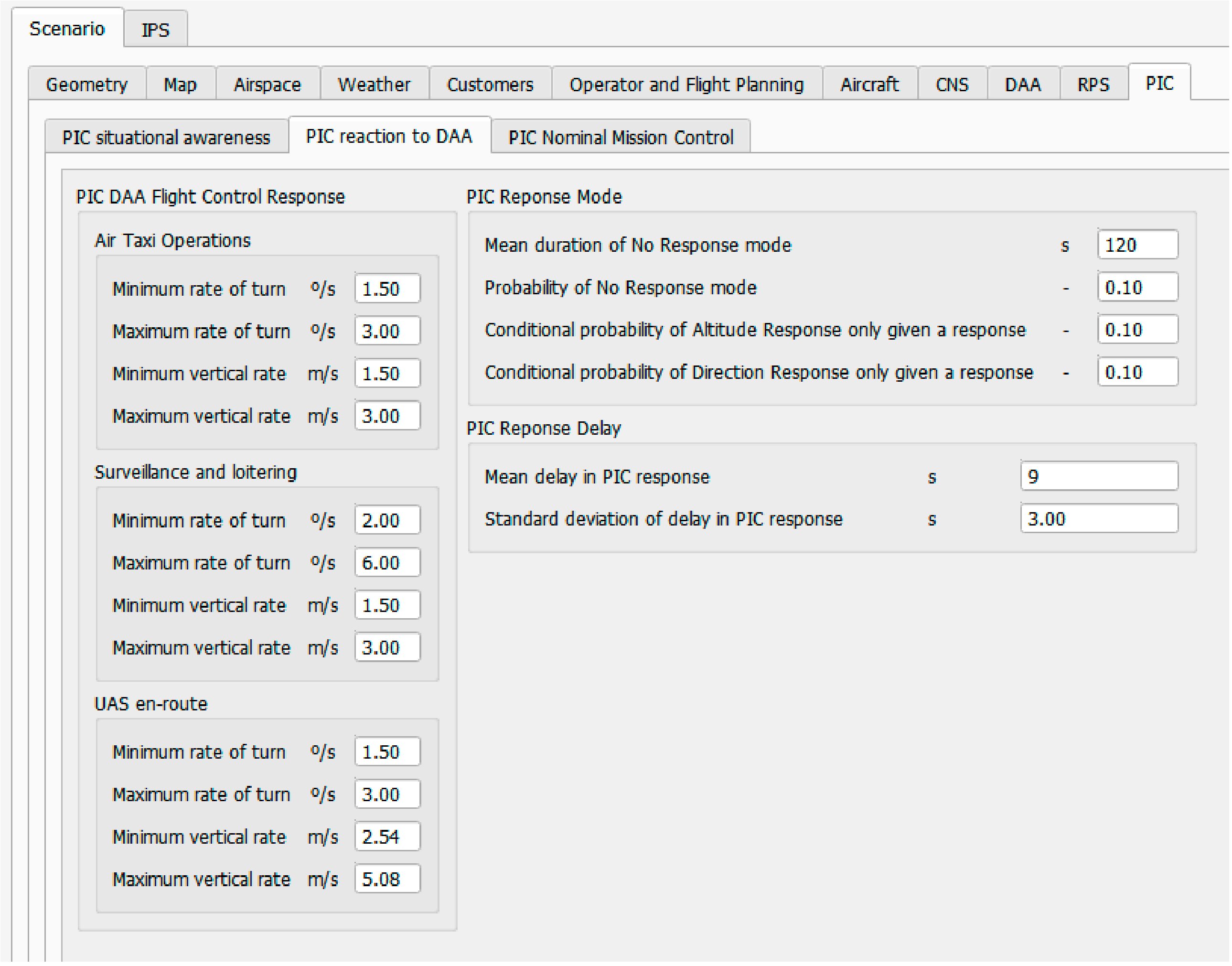
Urban Areas

	Latitude [°]	Longitude [°]	Radius [m]	Name
1	48.7	2.18	1600	Orsay
2	48.63	2.27	1000	Montlhéry
3	48.62	2.32	2100	Brétigny sur Orge

Figure 6: Main window.

4.2.1 Configuration settings

In the configuration mode, the user can adapt the default values of all parameters of the agent-based model and the IPS MC simulation. This is done via a series of tabs as illustrated in Figure 6. For instance, shown Geometry tab allows the user to set positions of the airports and urban areas and the overall flight zones employed in the simulations. As another example, Figure 7 shows the tab for defining parameters for the response by the PIC to DAA advisories, such as probabilities of response modes, moments of the probability density function for response delay, and extrema of rates of turn and climb/descent.



PIC DAA Flight Control Response

Air Taxi Operations

Minimum rate of turn	°/s	1.50
Maximum rate of turn	°/s	3.00
Minimum vertical rate	m/s	1.50
Maximum vertical rate	m/s	3.00

Surveillance and loitering

Minimum rate of turn	°/s	2.00
Maximum rate of turn	°/s	6.00
Minimum vertical rate	m/s	1.50
Maximum vertical rate	m/s	3.00

UAS en-route

Minimum rate of turn	°/s	1.50
Maximum rate of turn	°/s	3.00
Minimum vertical rate	m/s	2.54
Maximum vertical rate	m/s	5.08

PIC Reponse Mode

Mean duration of No Response mode	s	120
Probability of No Response mode	-	0.10
Conditional probability of Altitude Response only given a response	-	0.10
Conditional probability of Direction Response only given a response	-	0.10

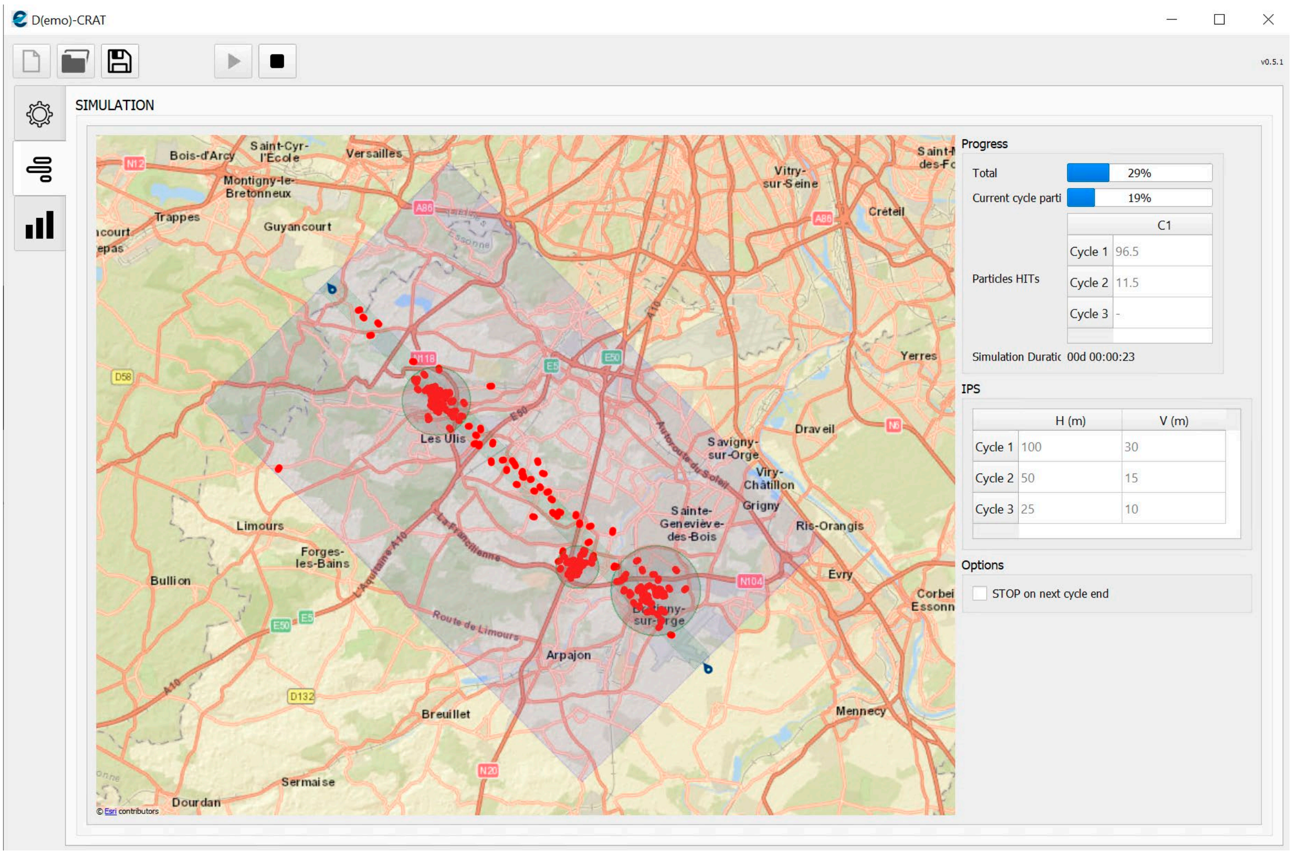
PIC Reponse Delay

Mean delay in PIC response	s	9
Standard deviation of delay in PIC response	s	3.00

Figure 7: Scenario configuration for PIC reaction to DAA.

4.2.2 Simulation mode

The simulation mode allows the user to manage and see the progress of a simulation. Figure 8 shows the simulation mode panel. The simulation progress is indicated by the percentage of overall simulation done, the percentage of particles simulated in the current cycle, and by the miss distance hits in the simulated cycles. The red dots in the map shows the positions of aircraft that have reached a miss distance boundary. It is possible to adjust miss distance boundaries of cycles that have not yet been completed (this leads to a restart of the current cycle).



Progress

Total	29%
Current cycle parti	19%

Particles HITs

	C1
Cycle 1	96.5
Cycle 2	11.5
Cycle 3	-

Simulation Duratic 00d 00:00:23

IPS

	H (m)	V (m)
Cycle 1	100	30
Cycle 2	50	15
Cycle 3	25	10

Options

□ STOP on next cycle end

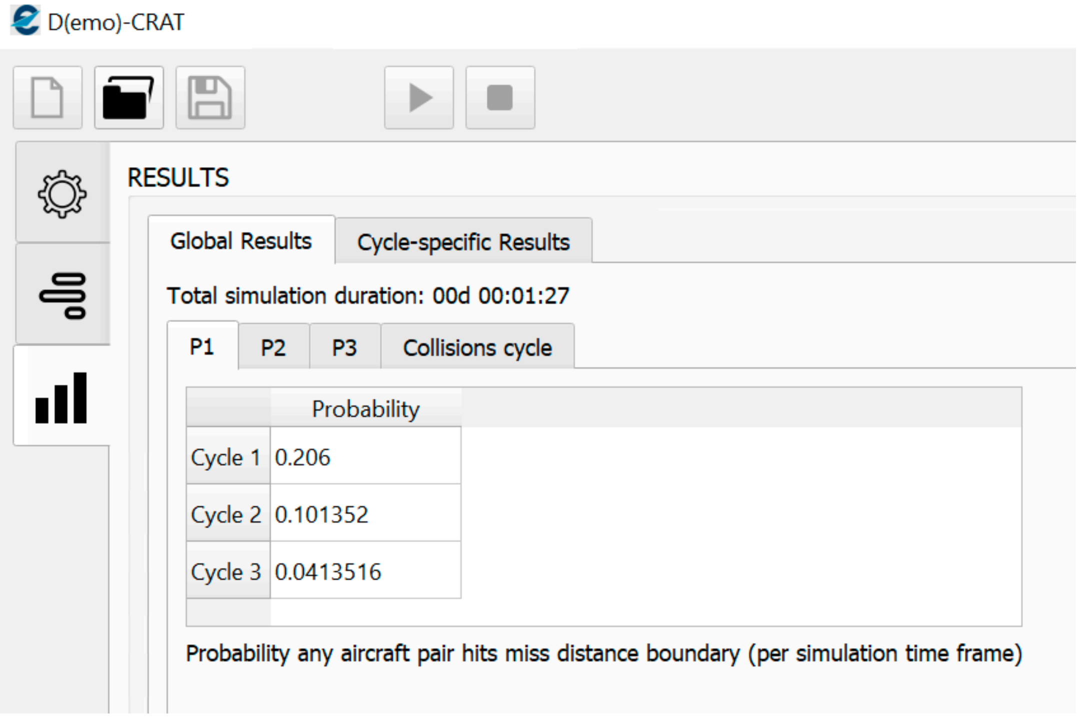
Figure 8: Simulation mode panel, showing simulation progress and close proximity locations (red dots).

4.2.3 Visualization of results

The visualization mode allows the user to study the results of a completed simulation. A full simulation is the result of the execution of a set of (zero to five) IPS cycles followed by a final collision cycle. For each type of cycle, specific cycle events are defined: for IPS cycles, the events are hits of the corresponding miss distances, whereas for collision cycles, events are collisions. The simulation outputs in the visualisation mode are classified in global results and cycle-specific results.

In the tab Global Results, the following statistics are provided:

- *P1*. Probability per simulation time frame that any aircraft pair hits the miss distance boundary of a cycle, which is shown for each of the miss distance cycles (Figure 9).
- *P2*. Probability per flight hour that an aircraft hits the miss distance boundary of a cycle.
- *P3*. Probability per flight that an aircraft hits the miss distance boundary of a cycle.
- *Collision cycle*: Average numbers of aircraft pair collisions per simulation time frame, average number of aircraft colliding per flight hour, average number of aircraft colliding per flight.



D(emo)-CRAT

RESULTS

Global Results | Cycle-specific Results

Total simulation duration: 00d 00:01:27

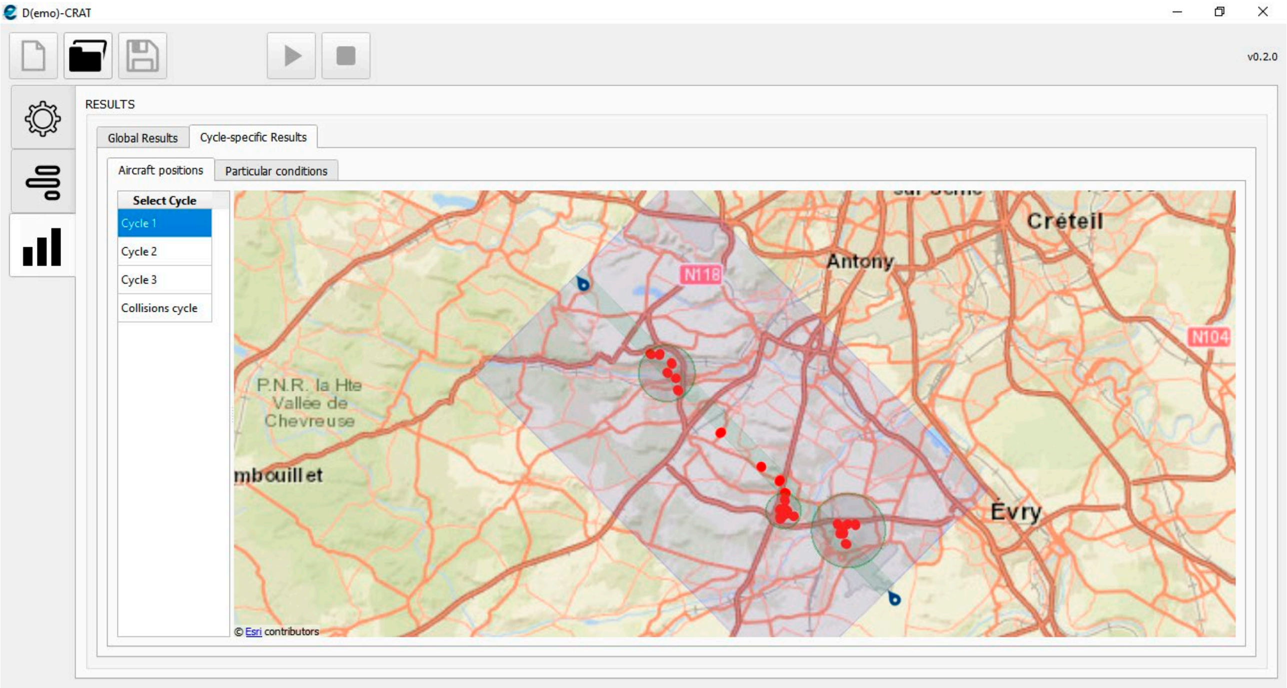
P1 | P2 | P3 | Collisions cycle

	Probability
Cycle 1	0.206
Cycle 2	0.101352
Cycle 3	0.0413516

Probability any aircraft pair hits miss distance boundary (per simulation time frame)

Figure 9: Global results P1: probability any aircraft pair hits miss distance boundary of a cycle (per simulation time frame).

The tab Cycle-specific results includes maps of the region with points for the positions of aircraft that are involved in miss distance hit in an IPS cycle, or that are involved in a collision (Figure 10). As such it provides insight into the areas that are most safety-critical.



D(emo)-CRAT

v0.2.0

RESULTS

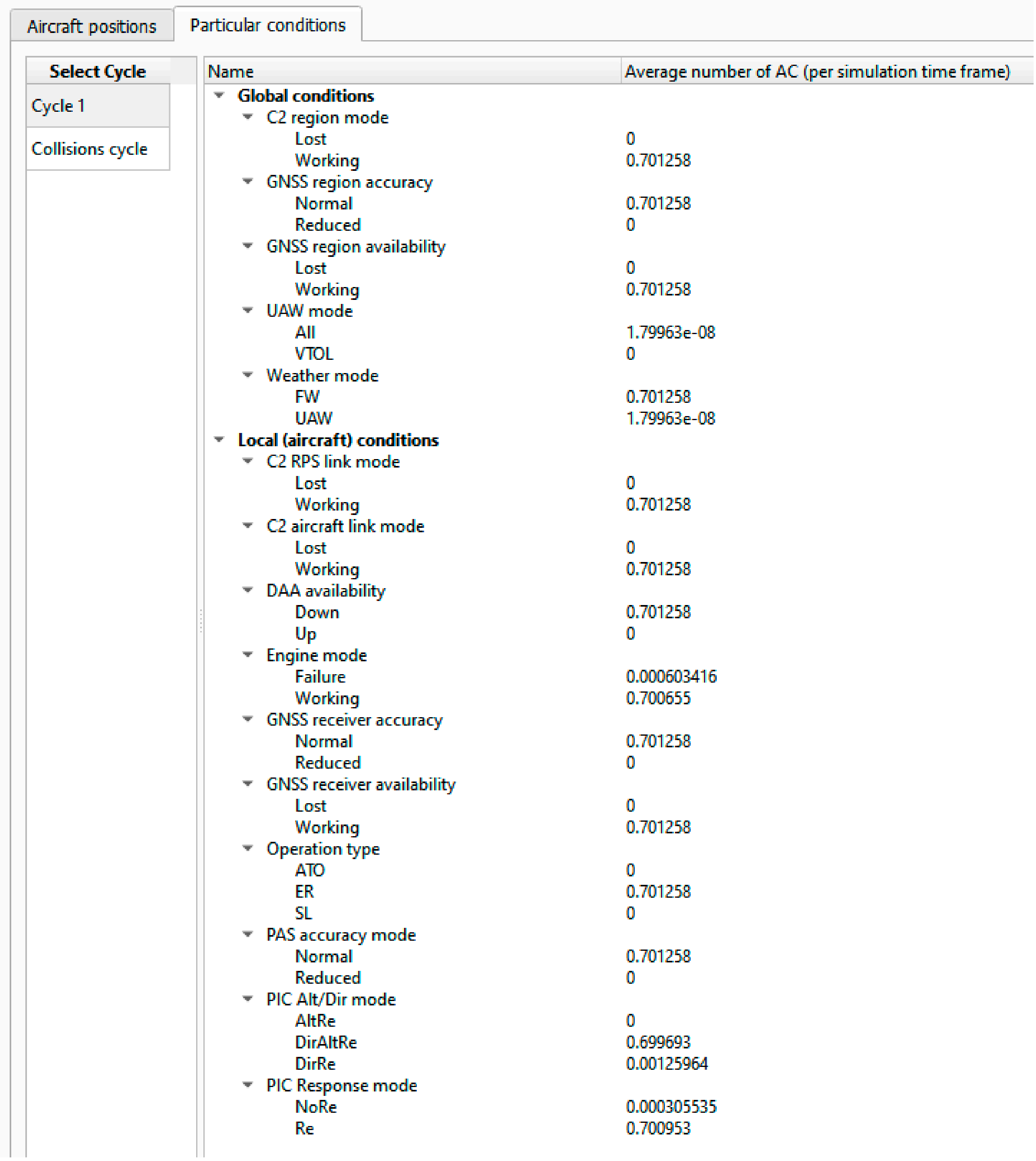
Global Results | Cycle-specific Results

Aircraft positions | Particular conditions

Select Cycle
- Cycle 1
- Cycle 2
- Cycle 3
- Collisions cycle

Figure 10: Visualization of the aircraft positions in miss distance hits or collisions.

The Particular conditions tab (Figure 11) shows statistics of modes/conditions of global variables (affecting all aircraft) or local variables (affecting a single aircraft) in combination with the event of attaining a miss distance boundary (in an IPS cycle), or a collision.



Name	Average number of AC (per simulation time frame)
Global conditions	
C2 region mode	
Lost	0
Working	0.701258
GNSS region accuracy	
Normal	0.701258
Reduced	0
GNSS region availability	
Lost	0
Working	0.701258
UAW mode	
All	1.79963e-08
VTOL	0
Weather mode	
FW	0.701258
UAW	1.79963e-08
Local (aircraft) conditions	
C2 RPS link mode	
Lost	0
Working	0.701258
C2 aircraft link mode	
Lost	0
Working	0.701258
DAA availability	
Down	0.701258
Up	0
Engine mode	
Failure	0.000603416
Working	0.700655
GNSS receiver accuracy	
Normal	0.701258
Reduced	0
GNSS receiver availability	
Lost	0
Working	0.701258
Operation type	
ATO	0
ER	0.701258
SL	0
PAS accuracy mode	
Normal	0.701258
Reduced	0
PIC Alt/Dir mode	
AltRe	0
DirAltRe	0.699693
DirRe	0.00125964
PIC Response mode	
NoRe	0.000305535
Re	0.700953

Figure 11: Average numbers of aircraft (per simulation time frame) for miss distance / collision events and specific values of global or local (aircraft specific) conditions in the agent-based model.

5 Illustrative simulation and risk results

This section shows simulation and risk results for missions in the chosen urban area south of Paris. Section 5.1 provides illustrative results for each of the mission types. Section 5.2 gives illustrative results for cycles in the IPS MC simulation, for the risk decomposition, and for the computational load of the MC simulations. Section 5.3 gives results for a sensitivity analysis of airspace characteristics, air traffic density and of manners of response by the pilot in command to DAA advisories.

5.1 Mission types results

D(emo)-CRAT includes four types of mission and each of them are illustrated in this section for a scenario without DAA and a scenario with DAA. In each of these cases the probability of attaining a close proximity event with horizontal and vertical miss distances within 50 m and 15 m, respectively, is compared and the locations of the close proximity events are illustrated.

Figure 12 illustrates the results for UAS en-route crossing operations, flying at random levels between 2000 and 3000 ft, with a mean time between the flights of 900 s. The duration of each simulated period (end time of customer demand) is 12 hours and 10,000 simulation particles are used. This implies that the total number of expected flights in the simulation is 480,000. Horizontal DAA zones are set at 1500 m, vertical DAA zones are set at 75 m, and the distance filtering factor is set at 1.5 (see also Section 5.3). Without DAA, the traffic leads to a close proximity probability of P=4.0e-4 per flight or P=2.9e-3 per flight-hour. With DAA, the close proximity probability is reduced by a factor 10 to P=5.0e-5 per flight or P=3.0e-4 per flight-hour. It can be recognised in the right pane of Figure 12 that many of the remaining close proximity events are close to the boundaries of the overall flight zone. At these boundaries the drones enter and exit the simulations and the possible DAA effectiveness is impacted negatively due to the sudden appearance of intruders. Without this model boundary effect, the risk reduction by the DAA system would thus be higher.

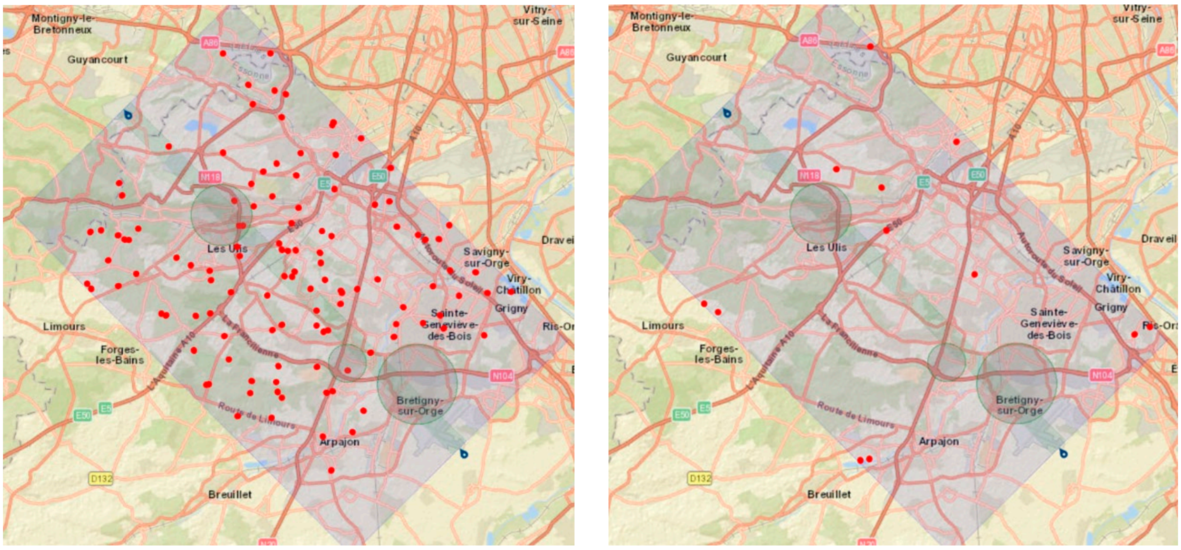


Figure 12: Close proximity events (HMD≤50m, VMD≤15m) for UAS en-route crossing operations (random flight levels). Left figure: without DAA, P=5.0e-4/fl, P=2.9e-3/fl-hr. Right figure: with DAA, P=5.0e-5/fl, P=3.0e-4/fl-hr.

Figure 13 illustrates the results for UAS en-route airport operations flying at random levels between 2000 and 3000 ft, with a mean time between the flights of 900 s. The duration of each simulated period (end time of customer demand) is 12 hours and 1000 simulation particles are used. This implies that the total number of expected flights in the simulation is 48,000. Horizontal DAA zones are set at 1500 m, vertical DAA zones are set at 75 m, and the distance filtering factor is set at 1.5 (see also Section 5.3). Without DAA, the traffic leads to a close proximity probability of P=1.3e-2 per flight or P=5.9e-2 per flight-hour. With DAA, the close proximity probability is reduced by a factor 22 to P=6.0e-4 per flight or P=2.7e-3 per flight-hour. It can be recognised in the right pane of Figure 13 that many of the remaining close proximity events are close to the model boundaries at the airports. This is a same model boundary effect as recognised in Figure 12. The risk levels in Figure 13 are higher than in Figure 12 as a result of the larger traffic density in the narrow flight zone between the airports. Interestingly, the risk reduction by the DAA system is also higher in Figure 13, presumably because it is easier to reduce the higher initial risk levels (without DAA).

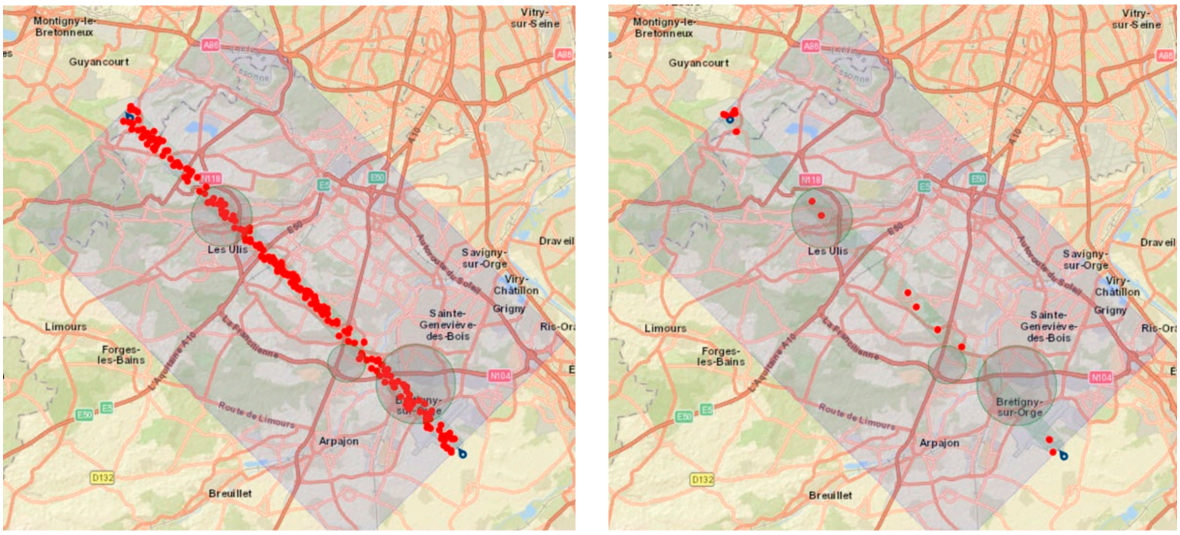


Figure 13: Close proximity events (HMD≤50m, VMD≤15m) for UAS en-route airport operations (random flight levels). Left figure: without DAA, P=1.3e-2/fl, P=5.9e-2/fl-hr. Right figure: with DAA, P=6.0e-4/fl, P=2.7e-3/fl-hr.

Figure 14 illustrates the results for air taxi operations between Orsay and Brétigny-sur-Orge flying at random levels between 400 and 2000 ft, with a mean time between the flights of 600 s. The duration of each simulated period (end time of customer demand) is 12 hours and 1000 simulation particles are used. This implies that the total number of expected flights in the simulation is 72,000. Horizontal DAA zones are set at 1500 m, vertical DAA zones are set at 100 m, and the distance filtering factor is set at 1.5 (see also Section 5.3). Without DAA, the traffic leads to a close proximity probability of P=6.7e-3 per flight or P=2.2e-2 per flight-hour. With DAA, the close proximity probability is reduced by a factor 4.5 to P=1.5e-3 per flight or P=4.8e-3 per flight-hour. It can be recognised in the right pane of Figure 14 that in this simulation all of the remaining close proximity events are in the urban areas. Several factors contribute to this phenomenon: (1) during vertical take-off and landing (VTOL) the DAA system is not active, since it was not designed for VTOL operations; (2) aircraft approaching an urban area can suddenly encounter air taxies that are in VTOL without sufficient time to react; (3) air taxies may use nearby arrival or departure positions and thus come in close proximity. These factors are likely to contribute to the lower risk reduction for the air taxi operations in comparison with UAS en-route operations in Figure 12 and Figure 13.

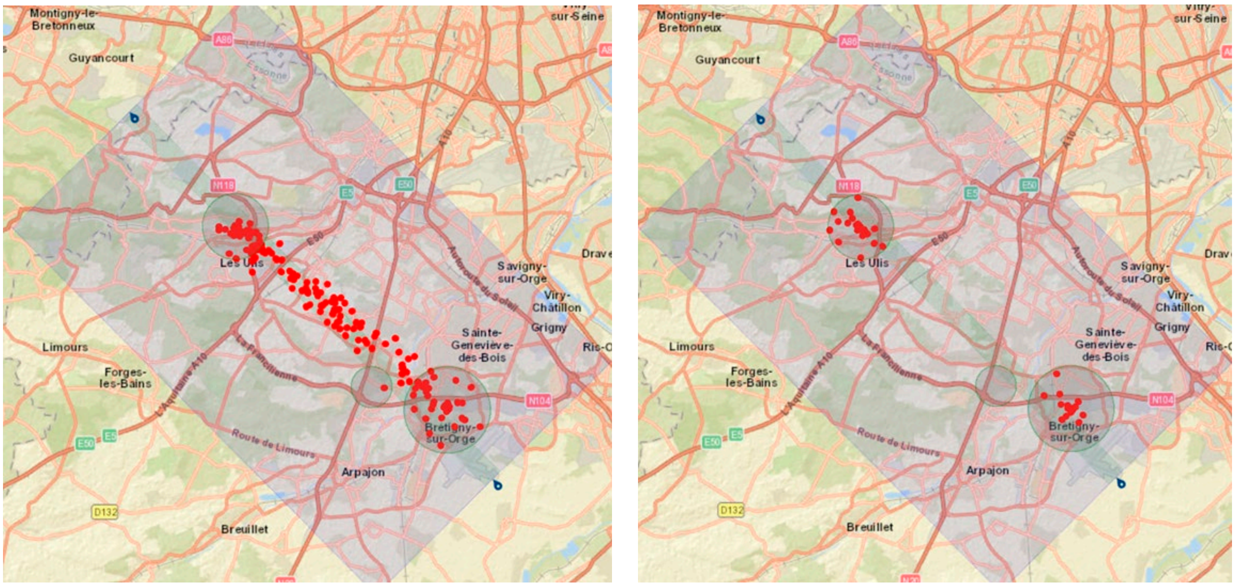


Figure 14: Close proximity events (HMD≤50m, VMD≤15m) for air taxi operations between Orsay and Brétigny-sur-Orge (random flight levels). Left figure: without DAA, P=6.7e-3/fl, P=2.2e-3/fl-hr. Right figure: with DAA, P=1.5e-3/fl, P=4.8e-3/fl-hr.

Figure 15 illustrates results for surveillance & loitering operations between in an area with a radius of 8000 m flying at random levels between 50 and 400 ft, with a mean time between the flights of 300 s and a mean flight duration of 1200 s. The duration of each simulated period (end time of customer demand) is 2 hours and 1000 simulation particles are used. This implies that the total number of expected flights in the simulation is 24,000. Horizontal DAA zones are set at 750 m, vertical DAA zones are set at 50 m, and the distance filtering factor is set at 1.5. Without DAA, the traffic leads to a close proximity probability of P=2.0e-2 per flight or P=7.1e-2 per flight-hour. With DAA, the close proximity probability is reduced by a factor 7 to P=2.6e-3 per flight or P=9.7e-3 per flight-hour. As explained for the air taxi operations, the effectiveness of the DAA system is limited by VTOL operations. So this can also be expected to limit the DAA effectiveness for the surveillance & loitering operations.

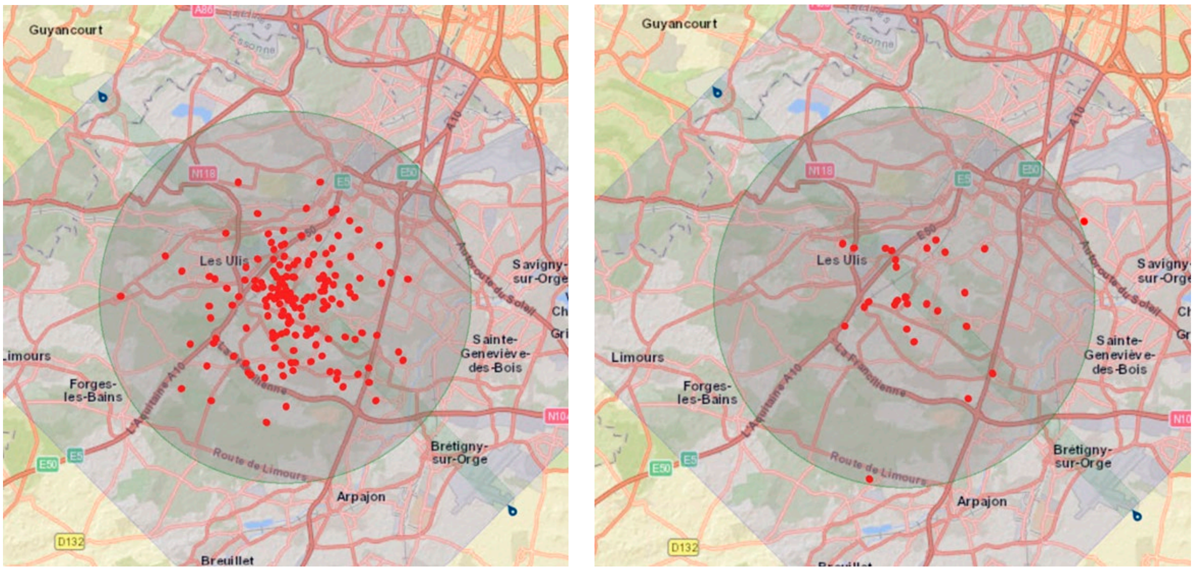


Figure 15. Close proximity events (HMD≤50m, VMD≤15m) for surveillance & loitering operations (random flight levels). Left figure: without DAA, P=2.0e-2/fl, P=7.1e-2/fl-hr. Right figure: with DAA, P=2.6e-3/fl, P=9.7e-3/fl-hr.

Figure 16 illustrates results for a scenario with mixed traffic consisting of UAS en-route crossing operations and air taxi operations between Orsay and Brétigny-sur-Orge. Both types of operations are at random levels between 200 and 2000 ft and the mean time between flight initiation is 600 s for each operation type. The duration of each simulated period (end time of customer demand) is 12 hours and 7000 simulation particles are used. This implies that the total number of expected flights in the simulation is about 1 million. Horizontal DAA zones are set at 1500 m for both operations, vertical DAA zones are set at 75 m for UAS en-route and at 100 m for air taxi operations, and the distance filtering factor is set at 1.5. Without DAA, the traffic leads to a close proximity probability of P=4.4e-3 per flight or P=1.9e-2 per flight-hour. With DAA, the close proximity probability is reduced by a factor 6.2 to P=3.0e-3 per flight or P=7.1e-4 per flight-hour. D(emo)-CRAT allows to inspect the average number of aircraft with some local condition, such as the operation type, at a miss distance boundary. While 50% of the flights are air taxi operations and 50% are en-route crossing operations, it follows from this data that without DAA 86% of the close proximities concern air taxi operations and 14% are en-route crossing. This can be explained by the larger density of air taxi operations. In the case with DAA, it follows that 93% are air taxi and 7% are en-route crossing. This is indicative of the larger effectiveness of the DAA system for the en-route crossing operations, as also recognised earlier in Figure 12 and Figure 14.

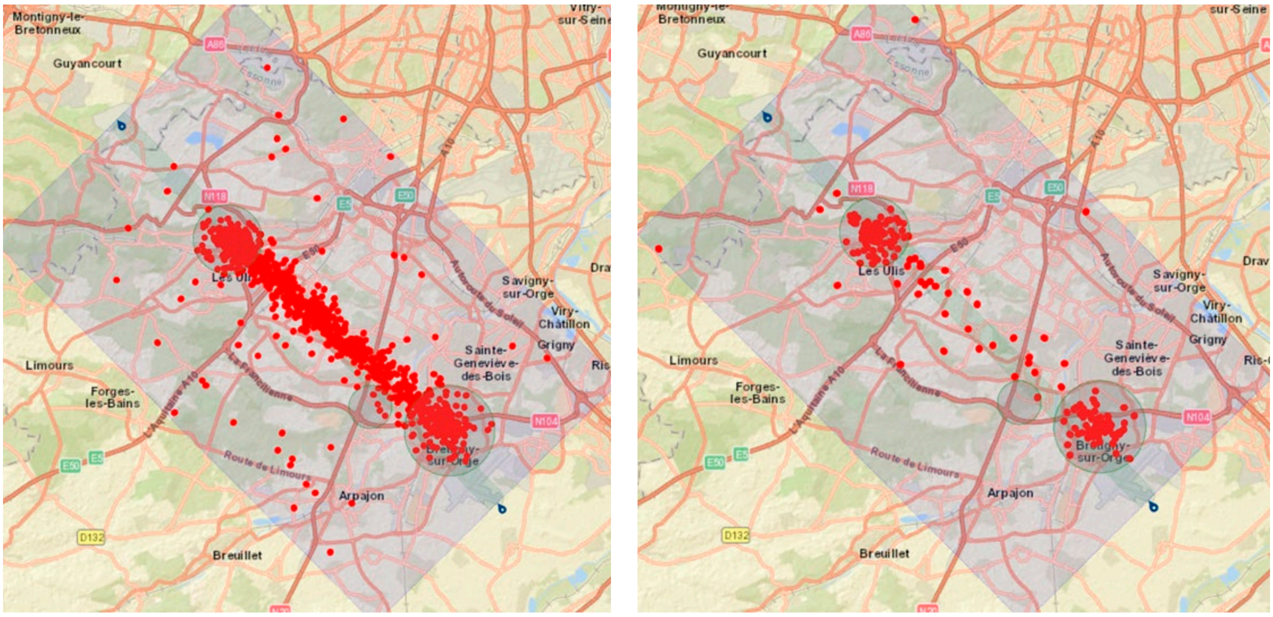


Figure 16: Close proximity events (HMD≤50m, VMD≤15m) for mixed traffic consisting of air taxies and en-route crossing operations (free flight with random flight levels). Left figure: without DAA, P=4.4e-3/fl, P=1.9e-2/fl-hr. Right figure: with DAA, P=7.1e-4/fl, P=3.0e-3/fl-hr.

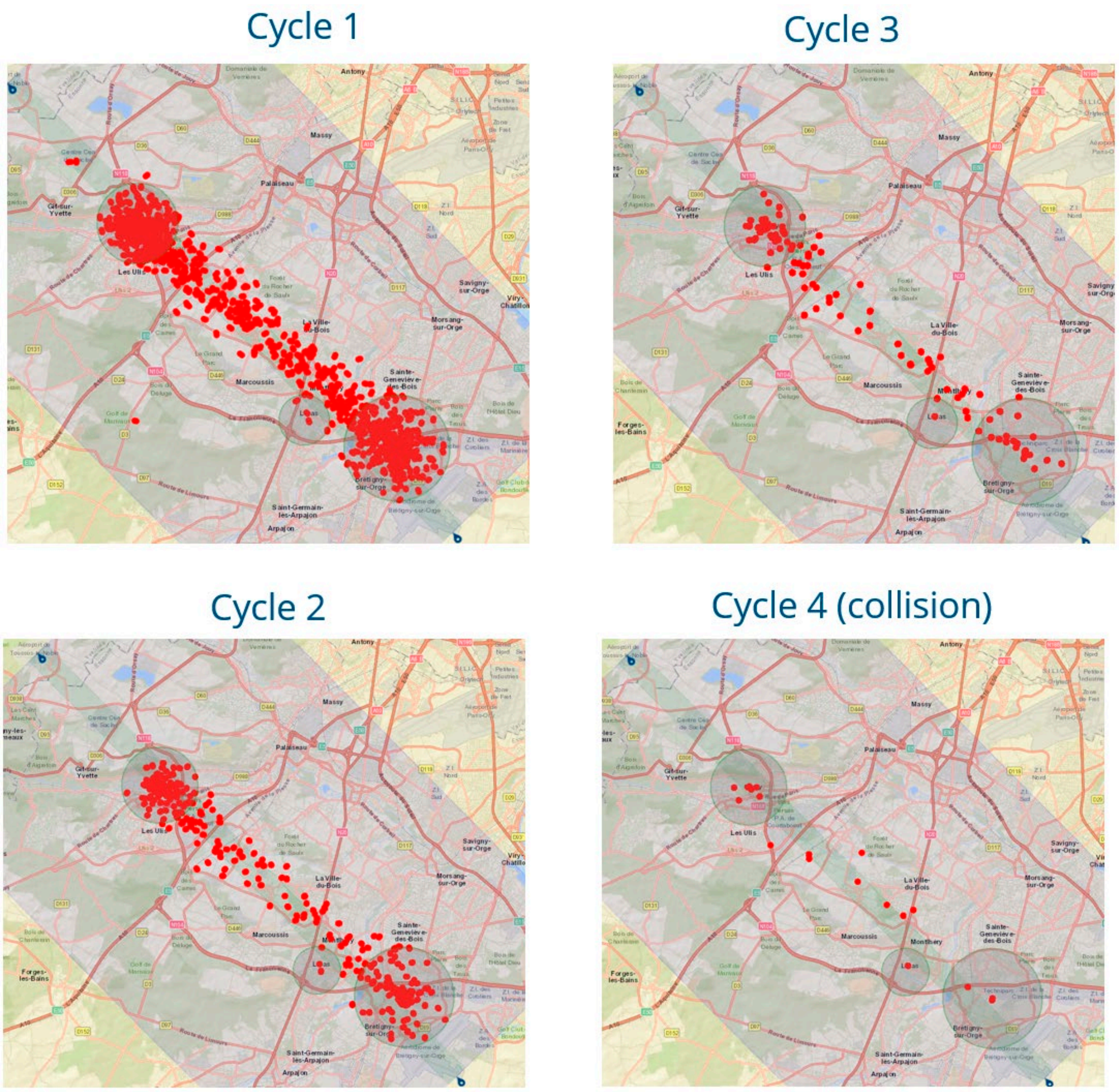
5.2 IPS, risk decomposition and computational load

Monte Carlo simulation with Interacting Particle Systems and risk decomposition of global failure conditions are characteristic features of D(emo)-CRAT, which are intended to accelerate the risk assessment process. This section illustrates these features and the computational load of the simulation process.

Figure 17 and Figure 18 illustrate results of an IPS MC simulation for a scenario with air taxi operations between Orsay and Brétigny-sur-Orge with a mean time between flights of 600 s. The duration of each simulated period (end time of customer demand) is 12 hours and 5000 simulation particles are used, implying a total of 360,000 flights. Horizontal DAA zones are set at 1500 m, vertical DAA zones are set at 100 m, and the distance filtering factor is set at 1.5 (see also Section 5.3). All flights are planned to use the same altitude of 1200 ft (thus inducing a large risk). Four IPS cycle were used with the following limits:

- Cycle 1: HMD≤100m, VMD≤30m,
- Cycle 2: HMD≤50m, VMD≤15m,
- Cycle 3: HMD≤25m, VMD≤10m,
- Cycle 4: Collision, implying HMD≤11.3m, VMD≤2.5m being the size of the air taxi.

Figure 17 shows the decrease in the density of proximity/collision events in the sequence of cycles and Figure 18 shows the decrease in event probability over the cycles, going down from 1.0e-2 to 1.0e-4 events per flight.



Cycle 1

Cycle 3

Cycle 2

Cycle 4 (collision)

Figure 17: Close proximity and collision events in four subsequent IPS cycles for air taxi operations between Orsay and Brétigny-sur-Orge (middle flight level). Cycle 1: HMD≤100m, VMD≤30m. Cycle 2: HMD≤50m, VMD≤15m. Cycle 3: HMD≤25m, VMD≤10m. Cycle 4: collisions.

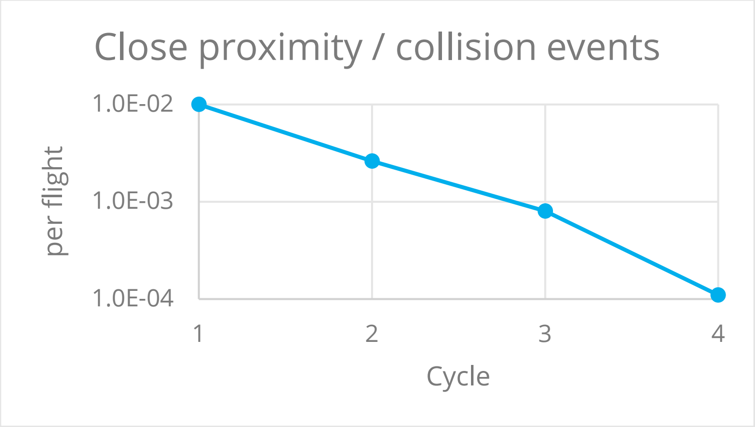


Figure 18: Probabilities of close proximity and collision events in four subsequent IPS cycles for air taxi operations between Orsay and Brétigny-sur-Orge (middle flight level) as shown in Figure 17.

Figure 19 and Figure 20 illustrate results of an IPS MC simulation with risk decomposition for global failure conditions concerning the availability of the C2 system in the region (*Working* or *Lost*) and the accuracy of GNSS in the region (*Normal* or *Reduced*). Decomposition for these two global failure modes implies four combinations of conditions. The scenario considers UAS en-route airport operations flying at an altitude of 2500 ft and a mean time between flights of 900 s. The duration of each simulated period is 12 hours and 1000 simulation particles are used for each risk decomposition condition, implying the simulation of about 36,000 flights per condition and about 144,000 flights in total. Horizontal DAA zones are set at 1500 m, vertical DAA zones are set at 75 m, and the distance filtering factor is set at 1.5 (see also Section 5.3). The locations of close proximities (Figure 19) are along the corridor and in particular near the airports, due to the lack of planning of arrivals and departures.

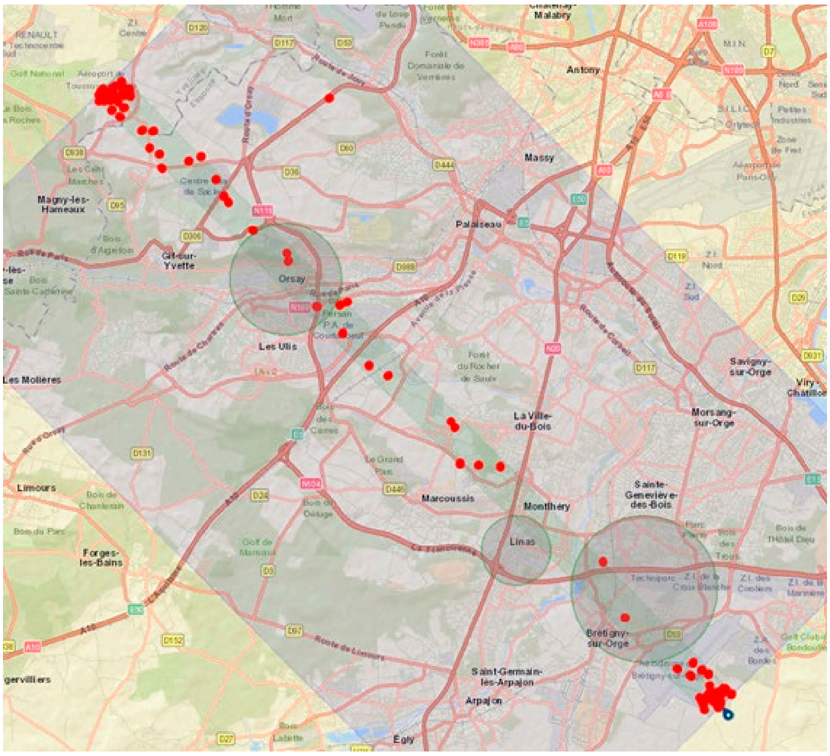


Figure 19: Close proximity events (HMD≤50m, VMD≤15m) for UAS en-route airport operations (middle flight levels), including DAA and risk decomposition.

Figure 20 shows the conditional probabilities of close proximities given each of the combinations of the risk decomposition as well as the combined probability for cases with and without DAA. There are no distinctive differences between the risk levels for the various conditions, but some risk increase can be observed for situations with DAA where both the C2 system is lost and the GNSS system has a reduced accuracy. The small differences indicate that the risk is now mostly driven by the nominal conditions, including the lack of departure and arrival planning. If the nominal risk would be lower, the risk increase due to the global failure conditions may be relatively higher. The total risk is almost completely determined by the situation without global failure conditions due to the low probability of the failure conditions and the small differences in the conditional risks.

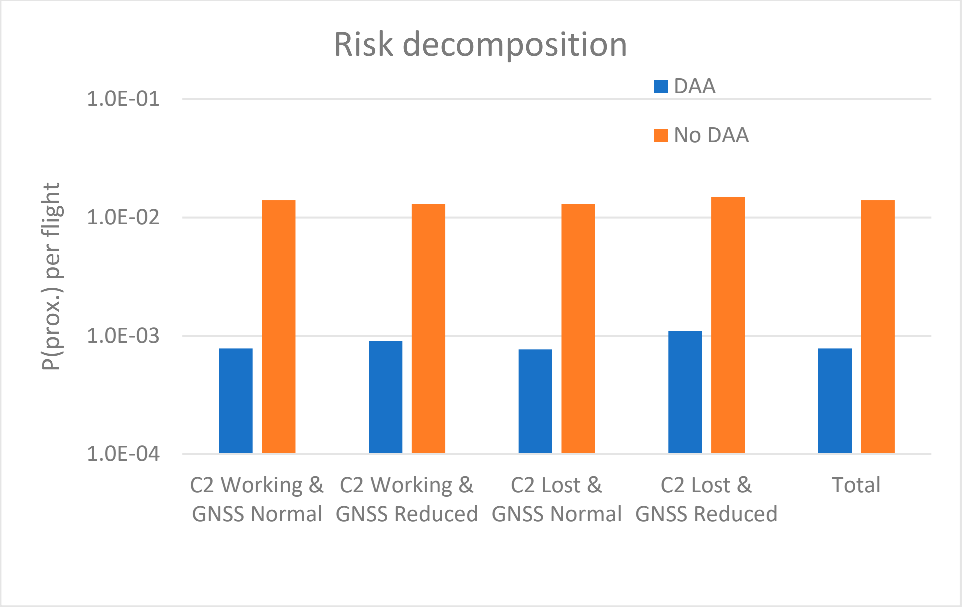


Figure 20: Conditional probabilities of close proximity events (HMD≤50m, VMD≤15m) given the combinations of the risk decomposition and the total (combined) probability for UAS en-route airport operations (middle flight levels) as shown in Figure 19.

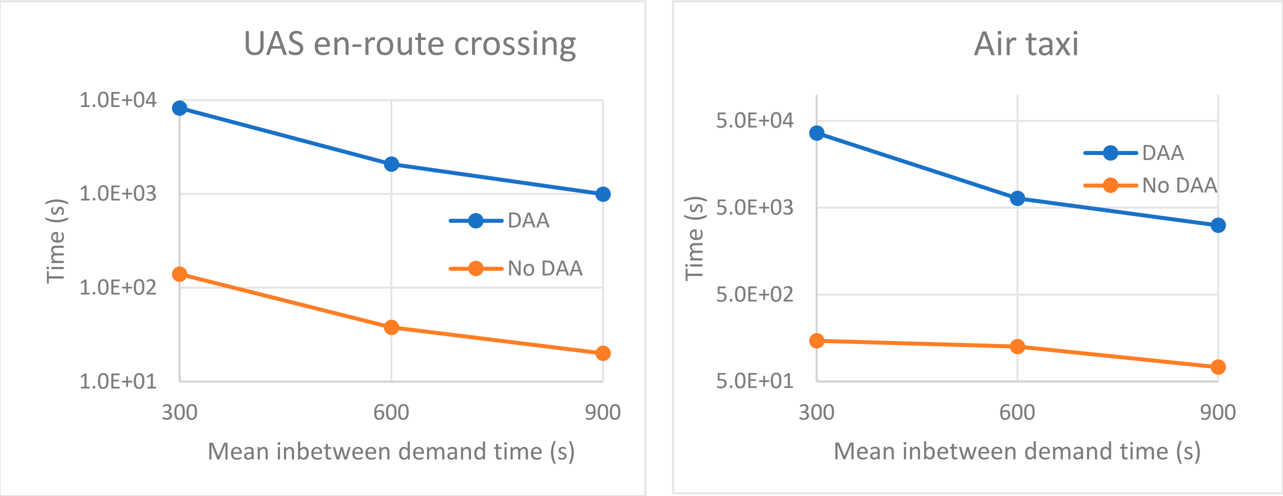


Figure 21: Computational load for 1000 particles with varying mean in between demand times for UAS en-route crossing operations (left figure) and air taxi operations (right figure).

Figure 21 provides results on the computational load of the IPS MC simulation for UAS en-route operations at 2500 ft and air taxi operations between Orsay and Brétigny-sur-Orge at 1200 ft. In all simulations the duration of each simulated period is 12 hours and 1000 simulation particles are used. The mean time between flights is set at 300, 600 or 900 s, implying that the expected number of flights is 144, 72, or 48 per simulated period. Horizontal DAA zones are set at 1500 m for both operations, vertical DAA zones are set at 75 m for UAS en-route and at 100 m for air taxi operations, and the distance filtering factor is set at 1.5. IPS MC simulation has been done with a first cycle of proximities with HMD≤50m and VMD≤15m, and a second cycle up to collisions. All simulations were done on a computer with a Xeon Silver 4114 CPU@2.2gHz (10-core, 20 threads). The computation times shown in Figure 21 make clear that there is a huge difference in the cases without DAA and with DAA system. Without DAA the computation time ranges from about 20 to 150 s, which is roughly about 1 ms per simulated flight. With DAA, the computation time ranges between 16 and 137 minutes for the UAS en-route scenario and between 52 minutes to 10 hours for the air taxi scenario. So the computation time is about 60 to 250 times higher with DAA. The largest factor is attained for the air taxi scenario with a mean time between flights of 300 s. In this scenario the traffic density is largest, implying that here the DAA system is called upon most frequently and is most likely to have multiple intruders.

5.3 Sensitivity analysis

A key advantage of a MC simulation-based approach for safety risk assessment is that is straightforward to assess the impact of changes in parameter values representing settings of technical systems, human operator performance, and environmental factors in the scenarios. Such sensitivity analysis provides a basis to evaluate the safety-relevance of components in the operation and it provides a basis to set requirements on minimum performance levels assure target levels of safety. In this section various examples of sensitivity analysis are provided; of course these are just a small sample of the sensitivity analyses that may be performed.

5.3.1 DAA zones

The key element for avoiding conflicts/collisions between drones in the operational concepts evaluated in D(emo)-CRAT is the DAA system DAIDALUS. The performance of DAIDALUS can be configured using a large set of configuration parameters (see overview in Section 9 of [6]). In D(emo)-CRAT nineteen of these configuration parameters can be tuned by the user for various types of operation. In addition, D(emo)-CRAT uses a “Distance filtering factor” that sets a minimum distance another aircraft such that it is incorporated in a DAIDALUS call. In the sensitivity analysis the following groups of parameters are varied:

- *Horizontal DAA zones*, which represent the parameters “DMOD of corrective/warning alert hazard zones” and “Minimum horizontal separation used in the computation of recovery manoeuvres”;
- *Vertical DAA zones*, which represent the parameters “Vertical separation threshold of corrective/warning alert hazard zones” and “Minimum vertical separation used in the computation of recovery manoeuvres”.

Figure 22 shows the impact of variation of the horizontal and vertical DAA zones for UAS en-route airport operations flying at a same level with mean time between flights of 300 s. Similarly, Figure 23 shows these relations for air taxi operations between Orsay and Brétigny-sur-Orge flying at a same level with mean time between flights of 300 s. The risk fraction is the probability of a close proximity event (HMD≤50m, VMD≤15m) attained by the DAA system in comparison to a scenario without DAA. The graphs shows that the risk fractions decrease for larger horizontal and vertical zones until they reach values where there is no or little gain.

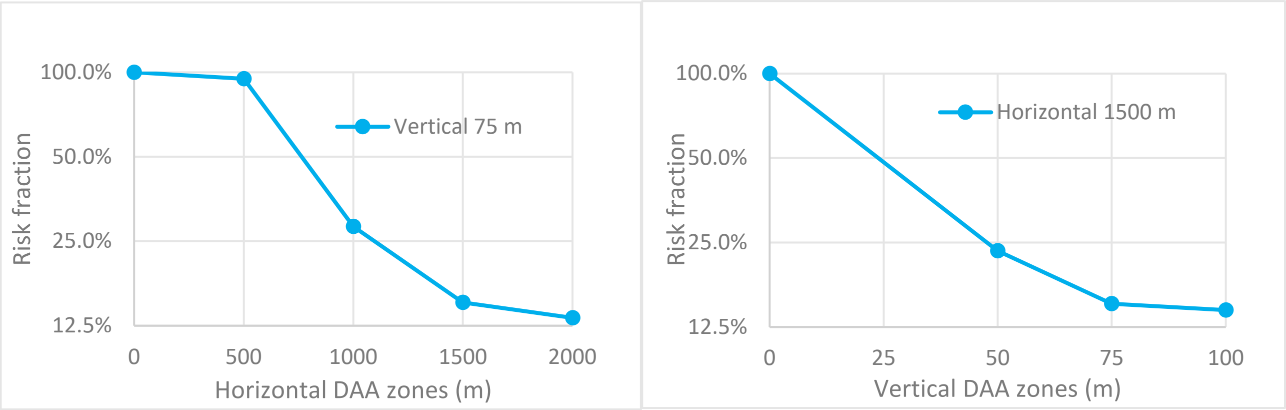


Figure 22: Relative risk reductions for close proximity events by DAA (with respect to scenarios without DAA) for various settings of horizontal and vertical DAA zones in UAS en-route airport operations.

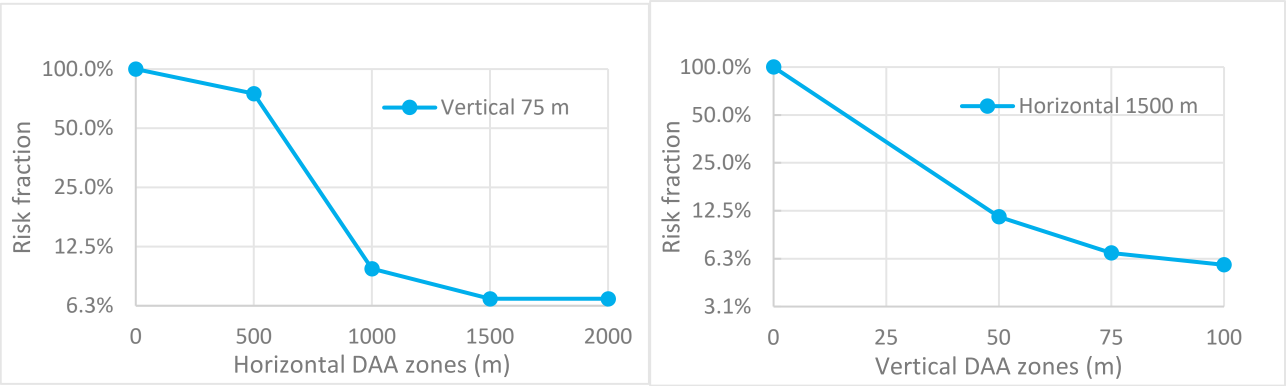


Figure 23: Relative risk reductions for close proximity events by DAA (with respect to scenarios without DAA) for various settings of horizontal and vertical DAA zones in air taxi operations.

5.3.2 Flight level planning, airspace configuration and traffic density

Next, close proximity probabilities are compared for simulations with variations in flight level planning, airspace configuration, and traffic density. The reference scenario considers UAS en-route airport operations with drones flying at 2500 ft and a mean interval between the flights of 900 s (i.e. 4 aircraft/hour). The duration of each simulated period is 12 hours and 1000 simulation particles are used. Horizontal DAA zones are set at 1500 m, vertical DAA zones are set at 75 m, and the distance filtering factor is set at 1.5. The locations of the close proximities in this case are similar to those for the scenario depicted in Figure 19.

Figure 24 compares the impact of drones flying randomly between 2000 and 3000 ft rather than all at 2500 ft. Without DAA, the risk is reduced by a factor 10 as a result of the vertically distributed flights. With DAA, the risk is reduced by a factor 12 due to combination of vertically dispersed flights and the DAA advisories.

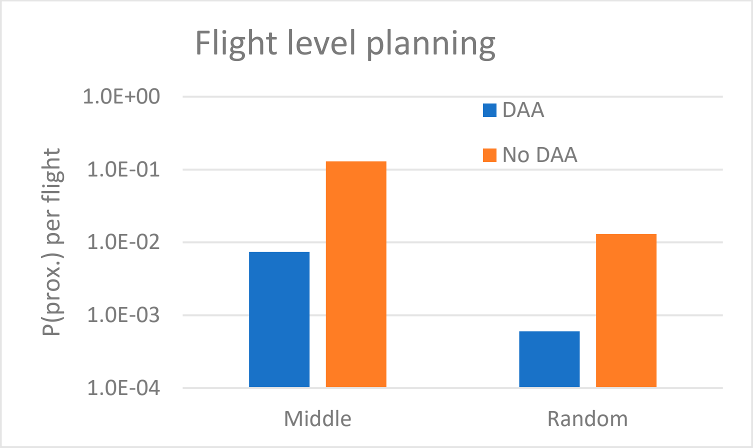


Figure 24: Probability of close proximity events (HMD≤50m, VMD≤15m) for UAS en-route airport operations employing middle flight levels versus random flight levels.

Figure 25 compares the impact of employing heading dependent altitude layers between the airports. This implies in this case that the flights moving to the south-east are planned to fly at 2250 ft and the flights moving to the north-west are planned to fly at 2750 ft altitude. Without DAA, this leads to a risk reduction of a factor 5. Here the remaining risk may be attributed to flights flying to the same airport with different speed. With DAA, the risk is reduced by a factor 10.

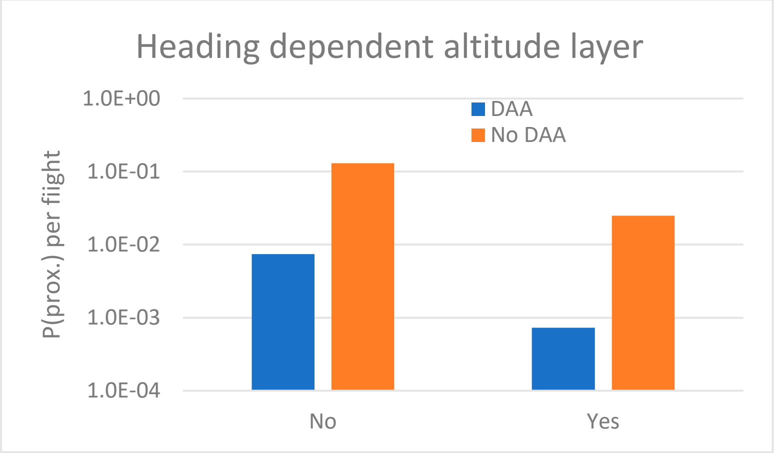


Figure 25: Probability of close proximity events (HMD≤50m, VMD≤15m) for UAS en-route airport operations with heading dependent altitude layers or not (middle flight levels).

Figure 26 compares the impact of the doubling the traffic density from 4 to 8 aircraft per hour (on average). Without DAA, this leads to increase in the risk of a close proximity by a factor 1.3. With DAA, the doubling of the traffic increases the risk by a factor 2.7. It could be that the seemingly lower DAA effectiveness is actually a result of the increase of the likelhood of a close proximity directly near one of the airports (where the DAA system cannot be effective due to the model boundary).

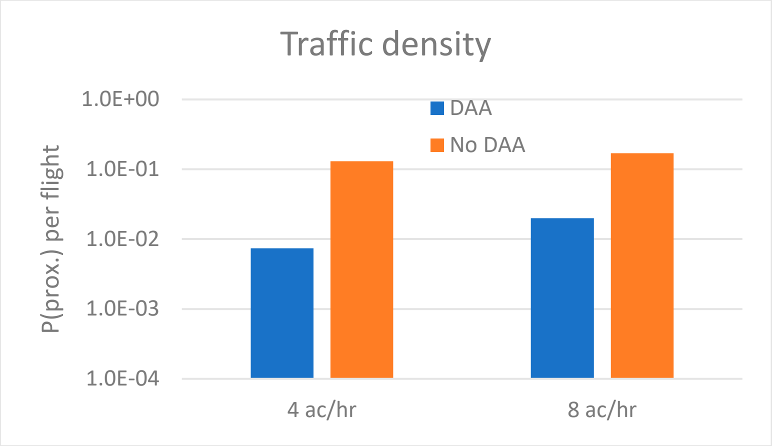


Figure 26: Probability of close proximity events (HMD≤50m, VMD≤15m) for UAS en-route airport operations (middle flight levels) with mean number of aircraft being 4 or 8 aircraft per hour.

5.3.3 PIC response to DAA

In the agent-based model each operation is controlled by a pilot-in-command (PIC) in remote pilot station (RPS) via a command and control (C2) datalink. The performance of the PIC in responding to the downlinked DAA advisories is an important factor in the effectiveness of the joint cognitive system consisting of the DAA system and the PIC. The model of the PIC performance includes a model for the delay in response and for the response mode.

- The response delay is chosen from a lognormal distribution. For the sensitivity analysis two settings are used:
  - *Slow*: mean delay is 9 s and standard deviation is 3 s (these settings are in line with observations in human-in-the-loop simulations in [62]);
  - *Quick*: mean delay is 3 s and standard deviation is 1 s.
- The response mode distinguishes between a mode where the PIC responds and a mode where the PIC does not respond to a DAA advisory. Given that the PIC responds, there can be three types of PIC response. which are studied in the sensitivity analysis:
  - *Altitude Response*: The PIC responds to DAA altitude guidance only;
  - *Direction Response*: The PIC responds to DAA direction guidance only;
  - *Both*: The PIC responds both to direction and altitude guidance.

Figure 27 shows risk reduction factors for close proximity events (HMD≤50m, VMD≤15m) for the PIC response options in UAS en-route crossing operations at an altitude of 2500 ft and mean time between the flights of 900 s. Horizontal DAA zones are set at 1500 m, vertical DAA zones are set at 75 m, and the distance filtering factor is set at 1.5. The risk reduction factor is the close proximity probability for a scenario without DAA divided by the close proximity probability for a scenario with DAA and a PIC response option. It follows that the largest risk reduction is attained of the PIC responds both to the direction and altitude guidance with a small delay. If the PIC would only respond to one of the guidance dimensions, it is more effective to only change altitude than to only change direction.

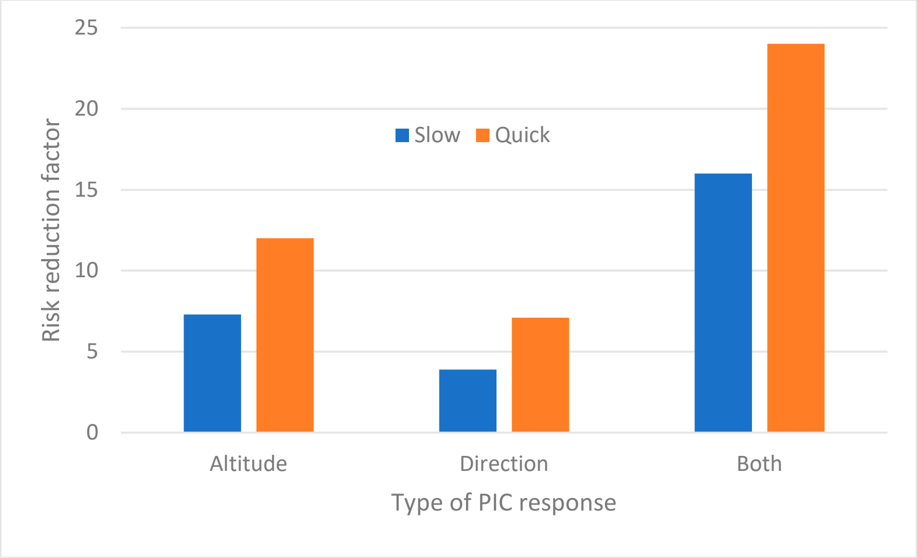


Figure 27: Close proximity risk reduction factors for types of PIC responses to DAA advisories in UAS en-route crossing operations: altitude response only, direction response only, both altitude & direction response, slow and quick response.

Similarly, Figure 28 shows the risk reduction factors for air taxi operations between Orsay and Brétigny-sur-Orge at an altitude of 1200 ft and a mean time between the flights of 600 s. Horizontal DAA zones are set at 1500 m, vertical DAA zones are set at 100 m, and the distance filtering factor is set at 1.5. Also here the largest risk reductions are achieved if the PIC responds to both the direction and altitude guidance, but remarkably there is almost no difference between the slow and quick responses. Responses to only altitude guidance are more effective than to only direction guidance for slow responses, but there is no difference in these risk reduction factors for quick responses. This limited impact of the response delay is not understood completely. It may be related to the smaller speeds of the air taxies.

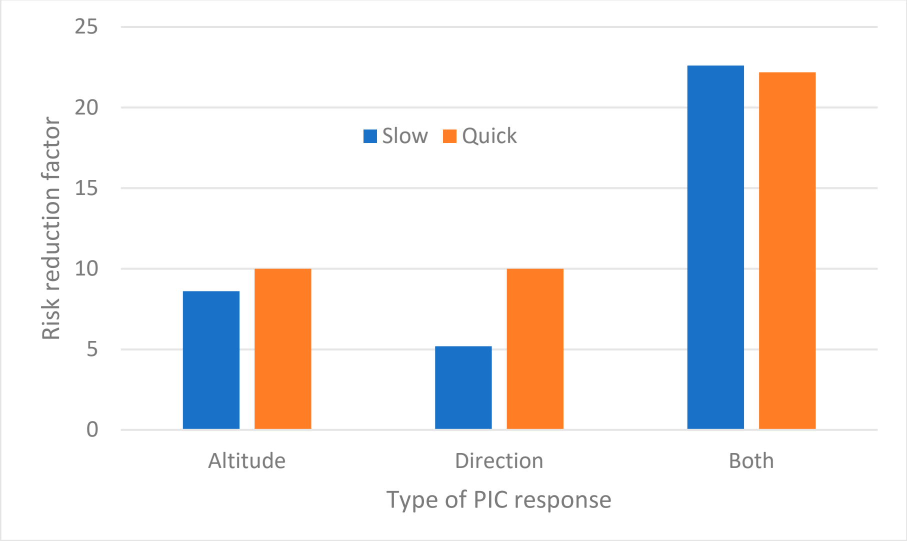


Figure 28: Close proximity risk reduction factors for types of PIC responses to DAA advisories in air taxi operations: altitude response only, direction response only, both altitude & direction response, slow and quick response.

6 Recommendations

This chapter provides recommendations for future developments of the Drone Traffic Collision Risk Assessment Tool (D-CRAT). The following types of recommendations are provided:

- Extended human machine interface (HMI)
- Conflict management functions
- Extended DAA systems
- Other types of risk
- Extended modelling
- Generic tool for agent-based dynamic risk modelling.

For each recommendation an initial assessment has been provided regarding the timescale and effort of execution of the recommendation.

- Timescale: short-term / mid-term / long-term. A shorter timescale indicates that a swift execution of the recommendation it expected to bring a considerable advantage for D-CRAT.
- Effort: small / medium / large. This is a rough indication of the expected effort for the execution of the recommendation.

6.1 Extended HMI

Visualization of selected flight data	
Timescale: short-term	Effort: medium
D(emo)-CRAT shows aircraft positions for miss distance events and collisions, in addition to statistics related to these events. In support of validation and verification of the models, as well as for enhanced analysis of scenarios, it is strongly advised to extend the visualization of simulation results. • To show the trajectories of selected drone flights in horizontal (x,y) and vertical (z,t) planes. Selection should be possible for particular types of operations, e.g. air taxi, surveillance & loitering, for particular conditions, e.g. a bad weather case or an engine failure, and for flights involved in miss distance events or collisions. • To show the DAA advisories of selected drone flights. • To show global system modes (e.g. *C2 Link Region Lost*, *Unexpected Adverse Weather*) and system modes of selected drone flights (e.g. *Engines Failure, ADS-B Lost*).	

Enhanced set of simulation metrics	
Timescale: short to medium-term	Effort: small - medium
The number of simulation metrics can be extended following feedback from users. Some examples are the following: • Allow the user to achieve statistics over specified regional zones, e.g. an area between urban zones. • Allow the user to achieve statistics for particular flight phases, e.g. during VTOL. The timescale and effort would depend on the feedback from users.	

Extension of the HMI & simulation functionalities	
Timescale: short-term	Effort: small
HMI and simulation functionalities can be extended following feedback from users. Some examples are the following: • To allow the user to discard the collision layer in the MC simulation, but to proceed until a miss distance boundary in a last simulation layer. In such last layer, it should be possible to proceed simulation after the first miss distance event, so as to possibly detect multiple miss distance events. • To analyse the possibility to allow the user to perform MC simulation with fixed replication (fixed seed of random number generator). • To shows the name of a scenario during a simulation. • To not visualize old simulation results after starting a new simulation. • To automatically load the configuration file of a simulation when displaying simulation results. • To assure that a “jamming” grey square in the map display is always removed. • Others...	

6.2 Conflict management functions

Strategic deconfliction of flight plans	
Timescale: short to mid-term	Effort: small - medium
Inclusion of strategic deconfliction algorithms in the drone traffic generator, which provide flight plans that are optimised for operational goals and priorities, while assuring strategically planned separation minima. Such extension increases the level of realism that can be attained by the simulations, as strategic deconfliction (CORUS phase U2) is expected to play an important role in safety management of drone operations. • In D(emo)-CRAT strategic deconfliction has been excluded completely and this can lead to peculiar conflicts, such as an air taxi and a drone taking-off from a very nearby location in an urban area. To avoid such conflicts, it is advised to include some basic strategic deconfliction rules, focusing on the start and end of flights, at a short stage (small effort). • In addition, developing concepts of more advanced strategic deconfliction strategies may be included in future developments (mid-term, medium effort).	

Dynamic capacity management	
Timescale: mid-term	Effort: medium
Inclusion of dynamic capacity management algorithms in the drone traffic generator in combination with setting of capacity-related scenario configuration parameters. For instance, rules could be added to restrict particular operations, depending on the overall demand for operations. Such rules would build on concepts for dynamic capacity management. D-CRAT simulations can thus support assessing the effectiveness of different dynamic capacity management strategies.	

Tactical conflict resolution by UTM	
Timescale: mid to long-term	Effort: medium - large
Inclusion of models for tactical conflict resolution by UTM in the agent-based modelling. Such models must be based on further development of the definition of such U-space service. This may include models for human operators, surveillance and communication systems of UTM providers. D-CRAT simulations can thus support assessing the effectiveness of different UTM concepts. The timescale and effort depend on the concept development and required level of detail of the models.	

Interaction with ATC	
Timescale: mid to long-term	Effort: medium
Inclusion of models for the interface of drone operators and UTM with ATC can support the assessment of procedures and systems for the interface with ATC. It is especially relevant for assessment of collision risk with manned aircraft. The foreseen type of interaction is diverting / holding of manned or unmanned aircraft in special (contingency) situations. Assuming that this may be represented by some basic models, the effort can be medium.	

Geofencing	
Timescale: mid-term	Effort: medium
Inclusion of the definition of no-fly zones as well as models for geofencing systems. D-CRAT simulations can thus support assessing the effectiveness of geofencing systems via the airspace infringement risk and possibly ground and collision risks.	

6.3 Extended DAA systems

D(emo)-CRAT has integrated DAIDALUS [30] developed by NASA as DAA system, because it is a reference implementation of the DAA MOPS [2], because its code is readily available under NASA's Open Source Agreement [57], and because it has easily tuneable parameters for adjustment to various types of operations. DAIDALUS was developed to provide manoeuvring guidance to pilots in command of large UAS to remain well clear and to recover from well-clear violations. The manoeuvring guidance in DAIDALUS was designed such that they are not in conflict with resolution advisories of airborne collision avoidance systems like TCAS II, but DAIDALUS itself does not incorporate a (short-term) collision avoidance functionality. DAIDALUS does not incorporate a specific surveillance and tracking module for filtering and fusion of surveillance data sources, but it assumes that position and speed of ownship and intruders are available. In D(emo)-CRAT simulations we found that the calculations by DAIDALUS tend to be a major component in the overall computational load of the MC simulations.

Enhanced tuning and modification of DAIDALUS	
Timescale: short to mid-term	Effort: small - medium
During the integration of DAIDALUS in D(emo)-CRAT we received valuable feedback from NASA developers. It is advised to extend the initial coordination with NASA. • To try to further improve the integration of DAIDALUS and the tuning of its parameters for the various types of operations. The objective is to improve the risk reduction and to reduce the computational load. This can be done at a short-term and for a small effort. • While DAIDALUS was developed for large UAS in higher airspaces, it may be possible to modify some of its algorithms to other types of operations, such as vertical take-off and landing phases by air taxies. The opportunity may be discussed with the NASA development team, and changes in the DAIDALUS algorithms could be made (mid-term, medium effort).	

Other DAA / ACAS systems for UAS

Timescale: mid to long-term	Effort: medium - large

It is advised to extend the library for simulation of additional DAA / ACAS systems, including the following.

- ACAS Xu is an airborne collision avoidance system for large UAS without hovering functionalities, which includes DAA RWC advisories as well as ACAS resolution advisories [1]. ACAS Xu includes a Surveillance and Tracking Module (STM) including fusion and filtering of various surveillance sources. This is an advantage, since no assumptions about surveillance and tracking, or separate surveillance and tracking algorithms have to be incorporated. ACAS Xu includes a Threat Resolution Module (TRM), which provides advisories based on an off-line optimised table. The use of such look-up table is memory intensive but supports a high computation speed. As the look-up table has been optimised for particular types of operations it cannot be easily used for other types of operations. Since response to resolution advisories is especially time-critical, responses to RAs can be handled most proficiently in an automatic manner by the flight management system of the UAS. Responses to DAA RWC advisories are expected to be handled by a PIC in a remote pilot station. The MOPS of ACAS Xu have been published [1] and it is expected that an implementation of ACAS Xu will be developed for EUROCONTROL within the coming year as part of the CAVEAT project. It is an opportunity to include this library in D-CRAT.
- ACAS sXu is an airborne collision avoidance system for small UAS with hovering functionalities, which unifies alerting and guidance for remaining well clear and avoiding collisions [36]. ACAS sXu MOPS are being developed. ACAS sXu includes STM and TRM modules. It is expected that its advisories are handled automatically by the FMS of the small UAS. It is advised to follow the development of ACAS sXu and to incorporate the libraries when they become available.
- Other DAA / ACAS algorithms

Collision avoidance of ground objects

Timescale: mid to long-term	Effort: medium - large

Systems for collision avoidance of ground objects are relevant for studying ground risk of UAS operations. Furthermore obstacle awareness can influence ACAS advisories. For instance in the development of ACAS sXu, obstacles are represented as stationary point intruders [36]. For studying ground risk and for studying UAS collision risk at low altitudes it is advised to incorporate obstacle awareness and avoidance functionalities.

6.4 Other types of risk

D(emo)-CRAT supports the assessment of probabilities of close encounters and collisions between various types of UAS operations, including air taxi operations, surveillance & loitering operations, and UAS en-route operations. Future extensions of the tool may support a broader variety of risks, which are discussed below.

Ground risk	
Timescale: long-term	Effort: large
Assessment of ground risk (fatalities and damage on the ground due to drone operations) may be provided by development of ground impact models, which estimate the severity and likelihood of the consequences of a drone hitting people or obstacles on the ground. Ground impact may be a consequence of a collision between two drones or of some failure condition of a single drone. Assessment of ground risk requires models for terrain, buildings and obstacles, as well as population densities. It requires detailed models for drone manoeuvring near obstacles and people, the sensors used, and the collision avoidance systems for ground objects and people.	

Airspace infringement risk	
Timescale: mid-term	Effort: medium
Airspace infringement risk (i.e. the likelihood that a drone enters a no-fly zone associated with a particular severity) can be assessed by definition of no fly zones, which may be dependent on the type of drone and mission. Such assessment can best be done in combination with modelling of geo-fencing systems. Furthermore, the airspace infringement risk may be coupled with the assessment of collision risk (manned, unmanned) and ground risk.	

Collision risk of drones with regular manned aircraft	
Timescale: mid-term	Effort: medium - large
Inclusion of flights by commercial air traffic (e.g. near airports) or general aviation. The agent-based modelling approach for the D(emo)-CRAT development allows for extension to other types of traffic and related technical systems and human operators. D-CRAT simulations can thus support assessing operational concepts and separation standards for drone operations in the vicinity of manned aircraft operations (commercial air transport and general aviation). • Collision risk may be evaluated under the assumption of no interaction between manned and unmanned operations, but just resulting from the trajectories in typical operations. Such evaluation can be done at mid-term with medium effort. • Alternatively, the collision risk may be evaluated including interaction between the manned flight (e.g. with ACAS Xa) and unmanned flight (e.g. with ACAS Xu). Such evaluation can be done at mid-term with a large effort, depending on the availability of relevant ACAS libraries.	

6.5 Extended modelling

Enhanced stochastic variability

Timescale: short to mid-term	Effort: small - medium

D(emo)-CRAT represents a limited set of hazards and types of errors. It is advised to extend the stochastic variability in the models, so as to extend the scope of modelling and the effectiveness of the IPS MC simulation. For instance, this could encompass jitter (dynamic noise) in sensors and more variability in PIC responses. Inclusion of jitter processes implies that models for filtering need to be incorporated.

Flight paths

Timescale: short to mid-term	Effort: small - medium

D(emo)-CRAT includes basic models of flight paths. The level of detail and scope can be extended in various ways:

- Arrival and departure procedures for fixed-wing drones.
- More complex routes and arrival/departure procedures.
- To include discrete sets of flight levels in flight planning.

Enhanced criteria for particle filtering

Timescale: short to mid-term	Effort: small - medium

D(emo)-CRAT uses distance-based criteria for particle filtering. The filtering criteria may be enhanced, e.g.

- Include relative speed as criterion.
- Exclude particles in indicated regions, e.g. near model boundaries.

Flight control and PIC

Timescale: short to mid-term	Effort: medium

D(emo)-CRAT has modelled the control of all flights in a same way. Each flight is controlled by a single PIC in a unique RPS and flight control actions in responses to DAA advisories are only done by the PIC. This generic control scheme can be extended in various ways:

- To allow for different flight control schemes, depending on type of operation.
- To include a mode for automatic response by the flight control systems of the UA to particular types of DAA advisories.
- To allow that a PIC can control multiple flights at the same time.
- To allow that there can be multiple PICs per RPS. As such there can be a shared dependency of various PICS for the functioning of RPS infrastructure (e.g. C2 Link systems).

Geospatial models

Timescale: mid to long-term	Effort: medium - large

D(emo)-CRAT does not include models for terrain, buildings and obstacles. Inclusion of such models could support various aspects, such as:

- The planning and conduct of flights, including manoeuvring around obstacles and setting of suitable altitudes.
- They set the basis for evaluation of collision avoidance systems and ground risk.
- Obstacles may have impact on the coverage of navigation and communication systems.

Population density models	
Timescale: mid to long-term	Effort: medium
Inclusion of models for population density can support the level of realism of demand for drone flights and they support the assessment of ground risk (drones hitting people).	

Navigation coverage models	
Timescale: long-term	Effort: large
GNSS-based state estimation is used for UAS navigation. In D(emo)-CRAT, failure conditions of GNSS-based state estimation are included affecting a single UAS or all UAS in a region, using generic failure probabilities and mean failure times. In an urban environment satellite signals may well be blocked by man-made structures and buildings. In D-CRAT, inclusion of models of urban infrastructure may be combined with models describing the likelihood that satellite signs are blocked and GNSS-based state estimation is (temporarily) unavailable. Such modelling of navigation coverage could be applied in combination with detailed modelling of urban infrastructure.	

Communication coverage models	
Timescale: long-term	Effort: medium - large
C2 links are used to control a UAS by a PIC from a RPS. There exist various types of C2 link architectures, including Radio Line Of Sight (RLOS), Beyond Radio Line Of Sight (BRLOS), and various relay architectures. In D(emo)-CRAT, failure conditions of C2 links are included affecting a single UAS or all UAS in a region, using generic failure probabilities and mean failure times. In D-CRAT, more detailed models of C2 link architectures may be used to support assessing the likelihood of a drone being impacted by lack of communications or service coverage during flight.	

6.6 Generic tool for agent-based dynamic risk modelling and simulation

In D(emo)-CRAT an agent-based model for MC simulation in support of drone traffic risk assessment was developed [6] and next this model was implemented in a dedicated tool [7, 8]. A separate initiative detailed in Appendix A aims to develop a generic tool for dynamic risk modelling and simulation. Such generic tool could then be used for enhanced modelling and simulation in D-CRAT as explained below.

Development of toolset for agent-based modelling and simulation	
Timescale: long-term	Effort: large
To develop a generic user-friendly toolset that can be used for agent-based dynamic risk modelling and Monte Carlo simulation. Such toolset aims to give the user the graphical support to develop a dynamic risk model using Stochastically and Dynamically Coloured Petri Nets (SDCPN) that is automatically tested on satisfying all the modelling rules, to automatically generate the software for Monte Carlo simulations, and give the user the support to run this software to obtain simulation results. See details in Appendix A.	

Formulating the D(emo)-CRAT model as an SDCPN	
Timescale: long-term	Effort: medium-large
Once a toolset of agent-based modelling and simulation has been developed, the models such as those used for D(emo)-CRAT may be implemented in that toolset. This allows to update and extend the models to include other aircraft configurations, other procedures, other route structures, other agents, etc.	

<table>
<tr><th colspan="2">Using the toolset strategically for agent-based modelling and simulation</th></tr>
<tr><td>Timescale: long-term</td><td>Effort: medium-large</td></tr>
<tr><td colspan="2">The toolset of agent-based modelling and simulation can next be used on a wider scale for advanced agent-based modelling and simulation by a variety of users. With its further use, agent-based models and sub-models can be collected in a library, which allows the recycling of sub-models and save valuable resources. A way forward for ATM safety is to define critical areas and scenarios wherein agent-based modelling and simulation should be applied, and to feed such results into safety cases. The resulting expanding library of agent-based models for various applications will form a valuable resource in support of an increasing number of low-cost agent-based modelling and simulation applications.</td></tr>
</table>

7 Conclusions

In the recent Strategic Research and Innovation Agenda “Digital European Sky” [26] it is expressed that for safety assurance of U-Space and urban air mobility: “New safety modelling and assessment methodologies applicable to U-space are needed. Tools are required to analyse and quantify the level of safety of U-space operations involving high levels of automation and autonomy, where multiple actors automatically make complex, interrelated decisions under uncertainty”.

Drone operations are based on radically new operational concepts for which there are no or only little data and experience. The performance and safety of the operations are based upon systems involving complex dynamic interactions at multiple levels:

- A variety of drone operations, including surveillance and loitering, parcel delivery, air taxies, higher altitude operations, which are operating in various environments;
- A variety of communication, surveillance and navigation systems, both in the drones and on the ground, and including DAA systems;
- High levels of automation and varying levels of autonomy;
- Human oversight and interactions;
- A broad variety of performance variability in normal conditions, such as sensor errors, human reaction times, normal weather;
- A broad variety of performance variability in off-nominal/failure conditions, such as system failures, human errors, adverse weather.

Because of the lack of experience with drone operations, the lack of data on safety occurrences, and the complexity of the operations, there is a strong need for safety risk assessment of drone operations using simulation approaches. Such simulation approaches should be able to effectively represent various types of drone operations, the uncertainty and hazards that can affect these operations, and they must support estimation of rare safety events up to the level of collision risk. Building on an evidence base of successful applications of safety risk assessment for complex and novel operations, the D(emo)-CRAT project set out to use agent-based dynamic risk modelling and IPS MC simulation for the demonstrator tool on drone collision risk assessment.

The results of the project show that agent-based dynamic risk modelling can be effectively applied to assess close proximity safety events of drone operations. The models describe various types of operations, VTOL air taxi and surveillance & loitering along the whole flight, and fixed wing drones in the en-route phase, in a generic urban environment. The models account for customer demand, various types of airspace design, aircraft navigation and dependent surveillance systems, C2 link systems in the drones, RPS and region, and PIC behaviour, and they have an interface with the reference DAA system DAIDALUS. The models describe dynamics and stochastic variability of the agents in normal conditions as well as in off-nominal/failure conditions. The distinction between such modes is often represented by discrete systems with a nominal and a failure mode that determines the performance of a single associated agent (for a local system) or many associated agents (for a global system).

The agent-based modelling approach has effectively supported the systematic description of interactions between many actors and their components. The detailed model description of [6] was found to be an effective basis for the largely independent software development [7]. The background in aerospace engineering of members of the software development team supported independent evaluation of the suitability and validity of the models. The description of agents, their components and their interfaces with other agents and the environment, supports relatively straightforward additions of additional agents and components in future extensions.

The IPS MC simulation approach and the risk decomposition for global failure conditions are advanced techniques for rare event risk assessment. These could be effectively combined with the agent-based modelling of drone operations and they could be formulated in full detail, such that they formed an effective basis for inclusion in the software design and its implementation. The illustrative simulation results show that especially the probability estimates for close proximity events, which are attained in the IPS cycles, provide useful insights in the safety of the drone operations.

Although the focus of the project has been on demonstrating the principles of agent-based dynamic risk modelling and IPS MC simulation for drone safety risk assessment, the software development team has excelled in implementing an efficient software tool. The implementation of the agent models and the IPS MC simulation algorithms in C++ and the effective distribution of simulation threads support the computational efficiency of the simulations. Furthermore, the backend could effectively integrate the reference DAA system DAIDALUS developed by NASA. The GUI allows to easily tune the large sets of parameters in the agent-based model, to attain proficient insight in the locations of safety events, and to efficiently show large numbers of statistics of safety occurrences and contributions of global and local settings.

The demonstration of the tool by the simulations in Section 5 has illustrated the variety of results that can be attained for the various types of missions. The results illustrate the IPS cycles up to the level of collisions and the risk decomposition. The sensitivity analyses show how the tool can be used to tune settings of the DAA system, to attain insight in the safety impact of airspace design and traffic density, and to understand the safety impact of PIC behaviour. These are just a small set of the number of sensitivity analyses that may be performed to attain insight in the sensitivity of simulated safety indicators for elements in the drone operational concept and their detailed system settings. Nonetheless, before embarking on large sets of sensitivity analyses it is recommended to first include a number of extensions in the modelling and simulation.

A considerable set of recommendations for extension of the models and their simulation, scope of simulation functionalities, and extension of the functionalities of the GUI are presented in Section 6. An initial assessment of timescale and effort has been made to help prioritization of possible future developments of D-CRAT. We believe that the models, simulation approaches and software developed in D(emo)-CRAT have shown to be at the heart of the needs for new safety modelling and assessment methodologies for U-space such as identified in the Strategic Research and Innovation Agenda “Digital European Sky” [26].

8 References

[1] EUROCAE. Minimum operational performance standards for Airborne Collision Avoidance System Xu (ACAS Xu): Volume I. ED-275, December 2020.

[2] RTCA. Minimum Operational Performance Standards (MOPS) for Detect and Avoid (DAA) Systems. Radio Technical Commission for Aeronautics, RTCA DO-365, 31 May 2017.

[3] ICAO. Proposals for the amendment of Annex 10, Volume V, first edition of Volume VI and consequential amendments to Annexes 1 and 2 arising from the thirteenth meeting of the Remotely Piloted Aircraft Systems Panel (RPASP/13). International Civil Aviation Organization, Ref. AN 7/67.1.1-19/52, 23 August 2019.

[4] Stroeve SH. D(emo)-CRAT Project Plan. Netherlands Aerospace Centre NLR, NLR-CR-2019-556, 31 January 2020.

[5] Stroeve SH, Everdij MHC, Garcia Daroca C. D(emo)-CRAT Requirements: Demonstrator Drone Collision Risk Assessment Tool. Royal Netherlands Aerospace Centre NLR, NLR-CR-2020-013, June 2020.

[6] Stroeve SH, Bakker GJ, Cañizares CV. D(emo)-CRAT Model Specification: Demonstrator Drone Collision Risk Assessment Tool. Royal Netherlands Aerospace Centre NLR, NLR-CR-2020-411, June 2021.

[7] Cañizares CV, Trezza M, Marcos EG, Calle IM, García DR, Martinez AP, et al. D(emo)-CRAT Software Design: Demonstrator Drone Collision Risk Assessment Tool. Netherlands Aerospace Centre NLR, NLR-CR-2020-409, June 2021.

[8] Stroeve SH, Trezza M, Calle IM, Cañizares CV, García DR. D(emo)-CRAT User Manual: Demonstrator Drone Collision Risk Assessment Tool. Netherlands Aerospace Centre NLR, NLR-CR-2021-135, June 2021.

[9] Stroeve SH, Bakker GJ, Everdij MHC, Trezza M, Cañizares CV, Calle IM, et al. D(emo)-CRAT Demonstration and Final Report: Demonstrator Drone Collision Risk Assessment Tool. Netherlands Aerospace Centre NLR, NLR-CR-2021-050, June 2021.

[10] SESAR Joint Undertaking. European ATM Master Plan: Roadmap for the safe integration of drones into all classes of airspace. 19 May 2018.

[11] SESAR Joint Undertaking. U-space blueprint.

[12] SESAR Joint Undertaking. Supporting safe and secure drone operations in Europe: Consolidated report in SESAR U-space research and innovation results. November 2020.

[13] CORUS Consortium. U-Space concept of operations. SESAR Joint Undertaking, 03.00.02, 25 October 2019.

[14] CORUS Consortium. U-Space concept of operations: Enhanced overview. SESAR Joint Undertaking, Edition 01.01.03, 4/9/2019.

[15] Barrado C, Boyero M, Brucculeri L, Ferrara G, Hately A, Hullah P, et al. U-Space Concept of Operations: A Key Enabler for Opening Airspace to Emerging Low-Altitude Operations. Aerospace. 2020;7:24. https://www.mdpi.com/2226-4310/7/3/24

[16] FAA. Urban air mobility (UAM) concept of operations. Federal Aviation Administration, Version 1.0, 26 June 2020.

[17] FAA. Unmanned aircraft system (UAS) traffic management (UTM) concept of operations Federal Aviation Administration, Version 2.0, 2 March 2020.

[18] European Commission. Commission Implementing Regulation (EU) 2019/947 of 24 May 2019 on the rules and procedures for the operation of unmanned aircraft 2019.

[19] EASA. UAS Safety Risk Portfolio and Analysis. European Aviation Safety Agency.

[20] EUROCAE. UAS system safety assessment objectives and criteria inputs to "AMC 1309". ER-019, October 2018.

[21] NASEM. Assessing the Risks of Integrating Unmanned Aircraft Systems (UAS) into the National Airspace System. Washington, DC: National Academies of Sciences, Engineering, Medicine; The National Academies Press; 2018.

[22] Clothier RA, Walker RA. Safety risk management of unmanned aircraft systems. Handbook of Unmanned Aerial Vehicles: Springer; 2015. p. 2229-75.

[23] Clothier RA, Williams BP, Fulton NL. Structuring the safety case for unmanned aircraft system operations in non-segregated airspace. Safety Science. 2015;79:213-28.

[24] Clothier RA, Williams BP, Hayhurst KJ. Modelling the risks remotely piloted aircraft pose to people on the ground. Safety Science. 2018;101:33-47. doi:https://doi.org/10.1016/j.ssci.2017.08.008.

[25] JARUS. JARUS guidelines on Specific Operations Risk Assessment (SORA). Joint Authorities for Rulemaking of Unmanned Systems, JAR-DEL-WG6-D.04, Ed. 2.0, 30 January 2019.

[26] SESAR Joint Undertaking. Digital European Sky: Strategic Research and Innovation Agenda. Draft, September 2020.

[27] EUROCAE. Minimum operational performance standards for Traffic Alert and Collision Avoidance System II (TCAS II), Volume I. ED-143.

[28] EUROCAE. Minimum Operational Performance Standards for Airborne Collision Avoidance System X (ACAS X) (ACAS Xa AND ACAS Xo). ED-256, October 2018.

[29] EUROCAE. Minimum Aviation System Performance Standards (MASPS) for the Interoperability of Airborne Collision Avoidance Systems (CAS). ED-264, September 2020.

[30] Munoz C, Narkawicz A, Hagen G, Upchurch J, Dutle A, Consiglio M, et al. DAIDALUS: Detect and Avoid Alerting Logic for Unmanned Systems. Proceedings of the 34th Digital Avionics Systems Conference (DASC 2015), Prague, Czech Republic, 2015.

[31] Kochenderfer MJ, Chryssanthacopoulos JP, Weibel RE. A new approach for designing safer collision avoidance systems. Ninth USA/Europe ATM R&D Seminar, Berlin, Germany, 2011.

[32] FAA. Concept of operations for the Airborne Collision Avoidance System X. Federal Aviation Administration, Version 2, Revision 0, 18 April 2013.

[33] EUROCAE. Minimum operational performance standards for Airborne Collision Avoidance System Xu (ACAS Xu): Volume II Algorithm Design Description. ED-275, December 2020.

[34] Davies JT, Wu MG. Comparative analysis of ACAS-Xu and DAIDALUS Detect-and-Avoid Systems. NASA, NASA/TM-2018-219773, February 2018.

[35] EUROCAE. Operational services & environment definition (OSED) for detect & avoid in very low-level operations. ED-267, August 2020.

[36] Alvarez LE, Jessen I, Owen MP, Silbermann J, Wood P. ACAS sXu: Robust Decentralized Detect and Avoid for Small Unmanned Aircraft Systems. 2019 IEEE/AIAA 38th Digital Avionics Systems Conference (DASC)2019. p. 1-9.

[37] Xue M, Rios J, Silva J, Zhu Z, Ishihara AK. Fe3: An Evaluation Tool for Low-Altitude Air Traffic Operations. 2018 Aviation Technology, Integration, and Operations Conference2018.

[38] Xue M. Sensitivity Analysis of Key Factors in High Density Unmanned Aerial System Operations. AIAA Scitech 2019 Forum, San Diego, California, 2019. doi:10.2514/6.2019-0688.

[39] Xue M, Do M. Scenario Complexity for Unmanned Aircraft System Traffic. AIAA Aviation 2019 Forum, Dallas, Texas, 2019. doi:10.2514/6.2019-3513.

[40] Yu H, Ren L, Castillo-effen M. A Framework for small Unmanned Aircraft System(sUAS) Trajectory Validation. 2018 Aviation Technology, Integration, and Operations Conference: American Institute of Aeronautics and Astronautics; 2018.

[41] Castillo-Effen M, Ren L, Yu H, Ippolito C. Off-nominal trajectory computation applied to unmanned aircraft system traffic management. 2017 IEEE/AIAA 36th Digital Avionics Systems Conference (DASC), 2017. doi:10.1109/dasc.2017.8102075.

[42] Ren L, Castillo-Effen M, Yu H, Johnson E, Yoon Y, Nakamura T, et al. Small Unmanned Aircraft System (sUAS) Categorization Framework V2.0. General Electric, 2017GRC0488, 22 February 2019.

[43] Sunil E, Ellerbroek J, Hoekstra J. Scenario Definition Report. Metropolis project, Deliverable D1.2.

[44] Schneider O, Kern S, Knabe F, Gerdes I, Delahaye D, Vidosavlkevic A, et al. Concept Design. Metropolis project, Deliverable D2.2.

[45] Delahaye D, Vidosavljevic A, Sunil E, Hoekstra J, Ellerbroek J, Aalmoes R. Work Package 3: Development & Metrics Definition. Metropolis project, Deliverable D3.2.

[46] Sunil E, Hoekstra J, Ellerbroek J, Vidosavljevic A, Arntzen M, Aalmoes R. Simulation Results and Analysis. Metropolis project, Deliverable D5.2.

[47] Ippolito CA, Hening S, Sankararaman S, Stepanyan V. A Modeling, Simulation and Control Framework for Small Unmanned Multicopter Platforms in Urban Environments. 2018 AIAA Modeling and Simulation Technologies Conference, 2018. doi:10.2514/6.2018-1915.

[48] Rabiller B, Fota N, Carbo L. SESAR Safety Reference Material. SESAR Joint Undertaking, D4.0.060, Edition 00.04.01, 14 December 2018.

[49] Rabiller B, Fota N, Carbo L. Guidance to aplly SESAR Safety Reference Material. SESAR Joint Undertaking, Edition 00.03.01.

[50] Everdij MHC, Blom HAP, Stroeve SH, Kirwan B. Agent-based dynamic risk modelling for ATM: A white paper. EUROCONTROL, January 2014.

[51] Fota N, Everdij M, Stroeve SH, Krakenes T, Herrera IA, Quinones J, et al. Using dynamic risk modelling in Single European Sky Air Traffic Management Research (SESAR). European Safety and Reliability Conference ESREL Wroclaw, Poland, 2014.

[52] Everdij MHC. Compositional modelling using Petri nets with the analysis power of stochastic hybrid processes [PhD]: University of Twente; 2010.

[53] Stroeve SH, Blom HAP, Bakker GJ. Systemic accident risk assessment in air traffic by Monte Carlo simulation. Safety Science. 2009;47:238-49. doi:10.1016/j.ssci.2008.04.003.

[54] Blom HAP, Bakker GJ, Blanker PJG, Daams J, Everdij MHC, Klompstra MB. Accident risk assessment for advanced air traffic management. In: Donohue GL, Zellweger AG, editors. Air Transport Systems Engineering: AIAA; 2001. p. 463-80.

[55] Blom HAP, Bakker GJ. Safety Evaluation of Advanced Self-Separation Under Very High En Route Traffic Demand. Journal of Aerospace Information Systems. 2015;12:413-27. doi:10.2514/1.i010243.

[56] Stroeve SH, Blom HAP, Medel CH, Daroca CG, Cebeira AA, Drozdowski S. Modeling and simulation of intrinsic uncertainties in validation of collision avoidance systems. Journal of Air Transportation. 2020;28:173-83. doi:10.2514/1.d0187.

[57] NASA. DAIDALUS software repository. https://github.com/nasa/wellclear. 2020.

[58] Volocopter. Volocity design specifications. August 2019.

[59] DJI. Matrice 600 User Manual. V1.0, 10/2017.

[60] SCR. Atlantic: medium range and high performance UAS. Madrid, Spain2021.

[61] Munoz C. DAIDALUS configuration WC_SC_228_nom_b. NASA, 18 March 2017.

[62] Guendel RE, Kuffner MP, Maki DE. A model of unmanned aircraft pilot detect and avoid maneuver decisions. Massachusetts Institute of Technology Lincoln Laboratory, Project Report ATC-434, 24 January 2017.

[63] Zimmermann A, Knoke M. TimeNet 4.0 User Manual: A software tool for the performability evaluation with stochastic and colored Petri nets Technical University of Berlin, ISSN: 1436-9915.

[64] Zimmermann A. Modelling and performance evaluation with TimeNET 4.4. Quantitative Evaluation of Systems - 14th Int Conf (QEST 2017), Berlin, Germany, 2017.

[65] Everdij MHC, Blom HAP. Hybrid state Petri nets which have the analysis power of stochastic hybrid systems and the formal verification power of automata. In: Pawlewski P, editor. Petri Nets. Vienna, Austria: I-Tech Education and Publishing; 2010. p. 227-52.

[66] Everdij MHC, Klompstra MB, Blom HAP, Obbink BK. Compositional specification of a multi-agent system by stochastically and dynamically coloured Petri nets. In: Blom HAP, Lygeros J, editors. Stochastic hybrid systems: Theory and safety critical applications: Springer; 2006. p. 325-50.

Appendix A Generic tool for agent-based dynamic risk modelling and simulation

Appendix A.1 Overview of the initiative

When it comes to safety risk assessment of complex multi-agent operations such as those in air traffic, Monte Carlo simulation of an agent-based stochastic dynamic risk model has shown to provide valuable results. This methodology allows for quantitative assessment of rare events in current and future operations, parameter sensitivity analysis, identification of the main bottlenecks for improving the risk, and more. Main drawback is that the process of implementation of a dynamic risk model in a software environment for Monte Carlo simulations, and the testing and verification of both model and software takes time and is therefore relatively expensive. If this process needs to be repeated from scratch for every new application, the costs may not be worth the advantages.

As part of initiatives outside the scope of the D(emo)-CRAT project, a partnership including NLR, the University of Ilmenau and EUROCONTROL has been set up to develop a generic toolset that can be used for agent-based dynamic risk modelling and Monte Carlo simulation. Such toolset aims to give the user the graphical support to develop a dynamic risk model that is automatically tested on satisfying all the modelling rules, to automatically generate the software for Monte Carlo simulations, and give the user the support to run this software to obtain simulation results. Once the toolset itself has been validated regarding producing error-free software for any model introduced, the testing and verification phase does not have to be re-done for each new application. In the long run this saves valuable time and resources.

The modelling format selected by this partnership is Stochastically and Dynamically Coloured Petri Net (SDCPN), which is a Petri net extension selected for its main strengths [50]:

- SDCPN have a graphical representation, making sure that the models are readable and verifiable.
- Their modelling power admits all types of stochastic dynamic processes and interactions occurring in ATM operations, including causal dependencies, concurrent processes, synchronisation of events, continuous processes, discrete events, random events, etc.
- The models can be built in a hierarchical way, starting from models for local agent entities, and building up to multi-agent models including all interactions.
- Each agent entity model maintains its own state throughout the modelling and safety risk analysis process. This improves readability and allows recycling of agent entity models from previous modelling exercises.
- The parameters in an SDCPN-based model are largely of a physical nature, and are relatively easily quantified compared to the parameters of approaches that rely on probabilities of conditional events.
- SDCPN generated processes have the strong Markov property, which is a pre-requisite for speeding up the analysis.

Therefore, SDCPN support the agent-based dynamic risk modelling of elements like complete 3D trajectories of aircraft, human behaviour, weather influences, performance of technical systems, sudden occurrences, interactions between all these elements, etc.

The toolset will be based on an existing tool named TimeNET [63], which is a Petri net modelling and simulation tool selected for the following reasons:

- TimeNET is a software tool for the modelling and performability evaluation of Petri nets. The tool is free of charge for non-commercial use.
- TimeNET has been successfully applied in several modelling and performance evaluation projects and there are several hundred installations in universities and other organizations worldwide. Sample application areas are communication systems, embedded systems, reliability evaluation, train control systems, manufacturing, supply chains, and business processes.
- Its main characteristics include the ability to model and evaluate complex coloured stochastic Petri nets, and efficient simulation methods for models with rare events (using the RESTART method [64]). The Petri net classes it supports already cover most of the features of SDCPN.
- It is being regularly improved and extended, hence is inherently flexible regarding the extension with necessary new features. Its developers at the University of Ilmenau are also open to implementing extensions.

TimeNET will first be extended to allow for SDCPN features. With this, it can be used to graphically develop SDCPN-based models, and to verify these models regarding correctness and consistency of using SDCPN-based modelling rules. Next, the tool will be extended with layers that allow for Monte Carlo simulation of the models, including simulation speed-up techniques, which allow for assessment of rare events such as collisions between aircraft.

Once the toolset is completed, it may be used to support the modelling and simulation activities done by the D(emo)-CRAT project, as well as its follow-on activities in D-CRAT. At the moment, the project partners have completed the requirements identification phase, which includes the identification of SDCPN-based model features that are not yet included in TimeNET, and have started to implement the features one by one into the TimeNET tool. These activities are of interest for D(emo)-CRAT Task T1.5 “Initial specification allowing future evolution of the demonstrator to a full-scale, user-friendly tool. This task provides an initial specification of the evolution of D(emo)-CRAT into a future full-scale, user-friendly tool D-CRAT (Drone Collision Risk Assessment Tool), considering the long term goal of allowing future evolution of the demonstrator to a full-scale, user-friendly tool.” A future merging of activities can be taken into consideration. The following subsection describes what this could look like.

Appendix A.2 Using the toolset for agent-based modelling and simulation

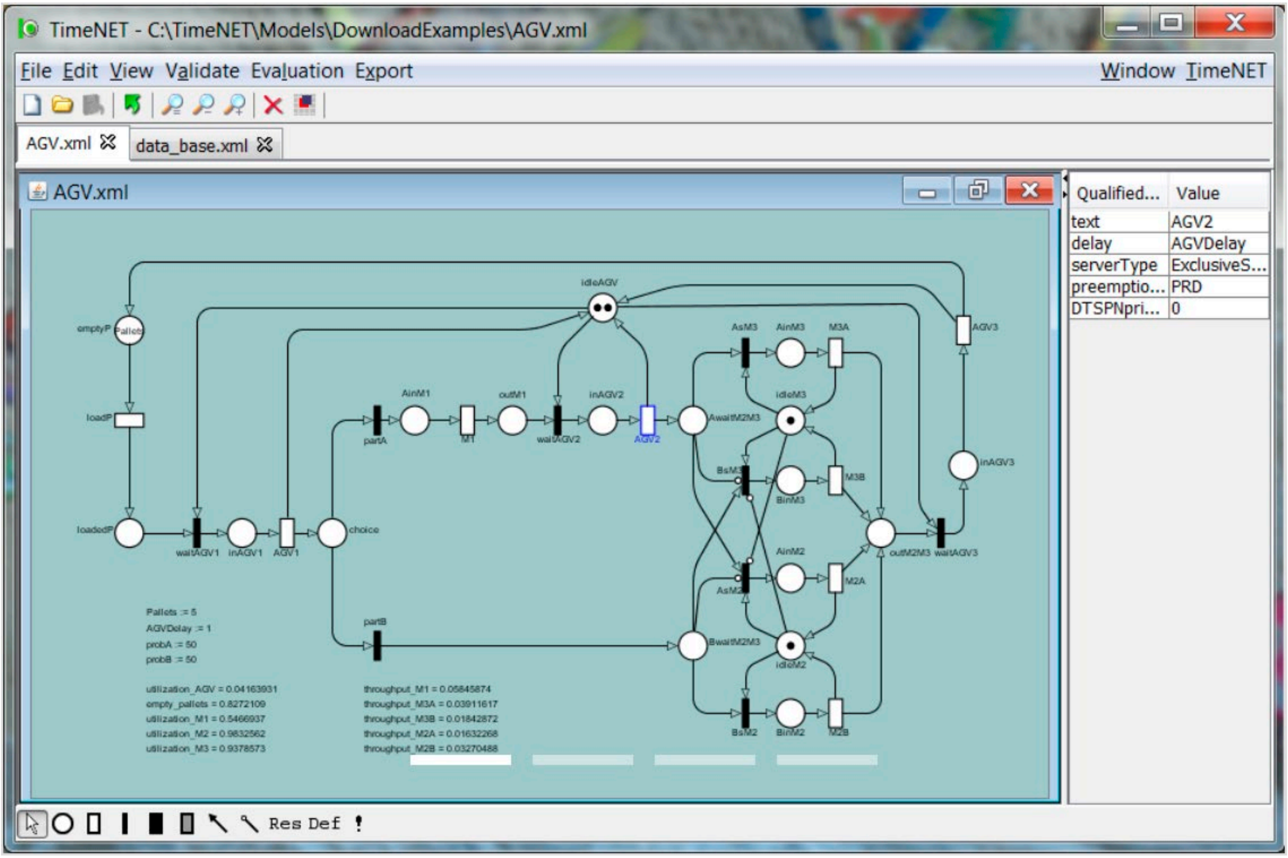
An ordinary Petri net, developed in 1962 by C.A. Petri, is a graphical and analytical formalism for modelling distributed systems. Discrete states are modelled by Places (depicted by circles). Tokens (depicted by black dots) inside the Places indicate whether the corresponding place is current. Switches between those discrete states are modelled by Transitions (depicted by rectangles). Places and transitions are connected by Arcs (arrows). Transitions remove tokens from their input places and produce them for their output places, thus modelling a discrete state change or mode switch. Numerous extensions have been proposed for this early formalism, including the notions of time, stochastics, coloured tokens. The Stochastically and Dynamically Coloured Petri Net (SDCPN) formalism was obtained through extensions that maintain the graphical elements and key properties of ordinary Petri nets, and add the notions of time, continuous-valued processes, various types of stochastics, and hierarchical modelling.

The process to develop an SDCPN-based model proceeds in several substeps. The first substep is to develop a local SDCPN-based model (referred to as Local Petri Net, LPN) for each agent entity identified for the ATM operation, by specifying all SDCPN elements, and how they work together. Next, the entities within one agent are coupled by modelling the interactions within each agent. Subsequently, all agent models are coupled by modelling the interactions between agents. Normally, there are iterations and loops between all substeps. Finally, all parameters are given a value. Once the SDCPN-based model for the operation is completed, it is implemented in a software environment for Monte Carlo simulation. This way, the operation can be played out millions of times to see how various event sequences may evolve under all conditions. Selected safety-critical events can be recorded, and their frequency of occurrence can be determined by counting how many times they happen in these millions of runs. Since air traffic is a very safe means of transport and the probability of a collision between two aircraft is extremely low, acceleration methods are applied to speed up the simulations.

SDCPN-based modelling and simulation has been used for safety risk assessment in numerous air traffic operations. The modelling was done on paper, using e.g. Microsoft Visio, and the software implementation was done in generic object oriented software languages such as Matlab or Turbo Pascal. Due to the agent-based nature of the modelling, many elements of both the models and the software could be re-used for each application, which saved valuable time. However, for the wider-spread usage of the formalism it was considered essential to develop a user-friendly toolset that was able to support the modelling, the generation of software for Monte Carlo simulations, and the running of those simulations to generate results. It was decided to extend an existing tool, i.e. TimeNET, for this purpose.

TimeNET is a software tool for the modelling and performability evaluation using stochastic Petri nets. TimeNET’s graphical user interface (GUI) allows the development of Petri net models by clicking on icons of places and transitions, positioning them in the drawing area on the screen, and connecting them by arcs, using mouse clicks only. Text such as names or parameter values can be inserted in small windows that open up if an icon for a place or transition is double-clicked. All these features are supported by user-friendly buttons and menus for moving objects, saving or re-using saved objects, aligning, editing, undoing, scaling and sizing, etc.

After generating and compiling software code, simulations of the behaviour of the tokens through time can be started and statistics on performance values can be collected in graphs. The simulation program has two working modes: normal simulation, which is intended for an efficient computation of performance measures; and a single-step mode used in conjunction with the GUI for an interactive visualization of the behaviour (token game). Results of the simulation run are graphically displayed on the user's screen during the simulation in a result monitor program. They are also stored in files that can be analysed after a completed simulation.



Pallets := 5
AGVDelay := 1
probA := 50
probB := 50

utilization_AGV = 0.04163931
empty_pallets = 0.8272109
utilization_M1 = 0.5466937
utilization_M2 = 0.9832562
utilization_M3 = 0.9378573

throughput_M1 = 0.05845874
throughput_M3A = 0.03911617
throughput_M3B = 0.01842872
throughput_M2A = 0.01632268
throughput_M2B = 0.03270488

Qualified...	Value
text	AGV2
delay	AGVDelay
serverType	ExclusiveS...
preemptio...	PRD
DTSPNpri...	0

Figure 29: Example GUI for TimeNET, taken from https://timenet.tu-ilmenau.de/#/features

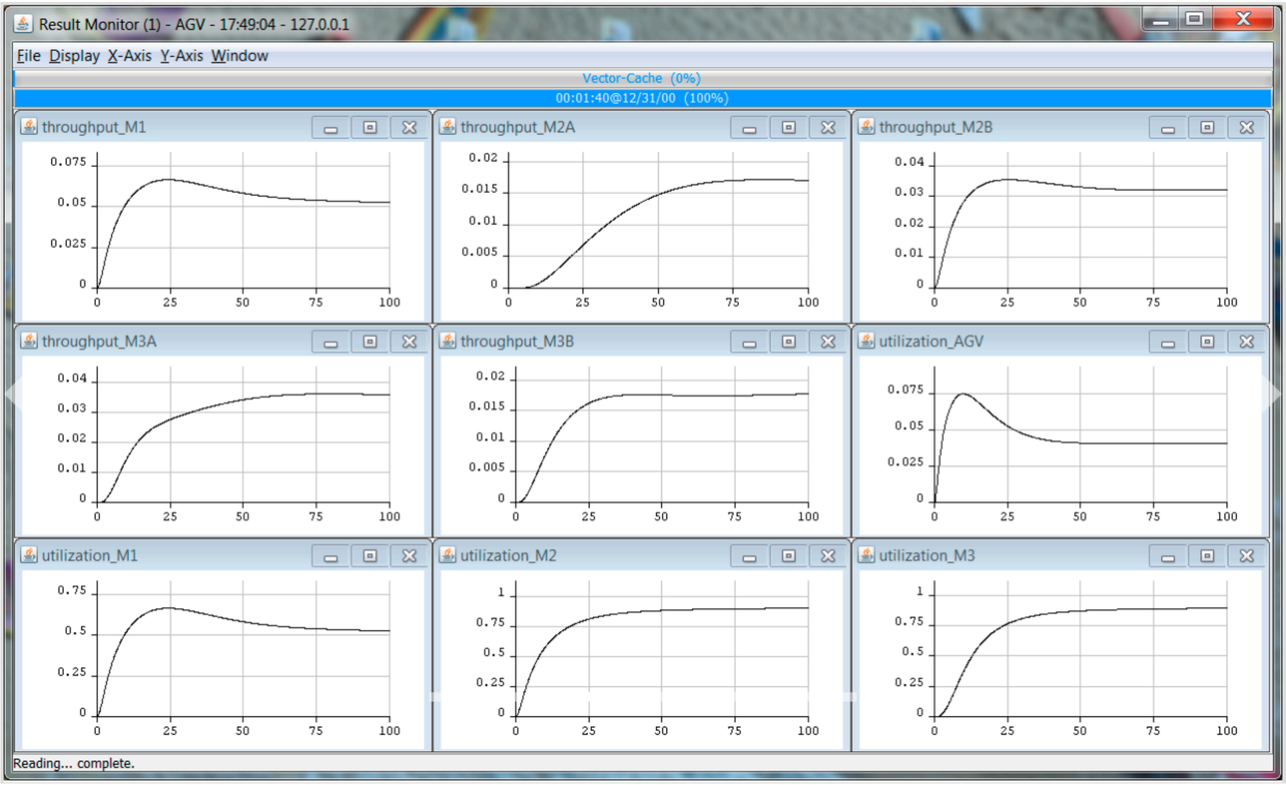


Figure 30: Example GUI for TimeNET, taken from https://timenet.tu-ilmenau.de/#/features

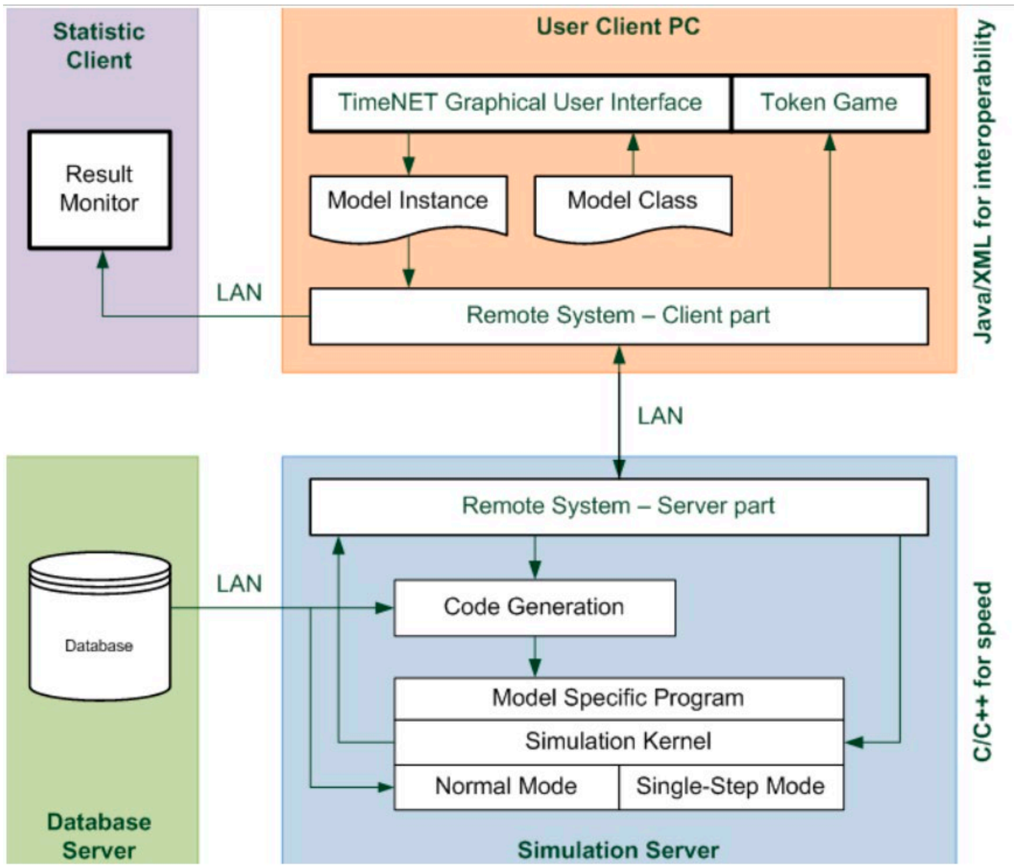


Figure 31: Software architecture from modelling in TimeNET, taken from http://www.eecs.tu-berlin.de/fileadmin/f4/TechReports/2007/2007-13.pdf

TimeNET can be used to model and evaluate implementations of two types of Petri net, i.e. Extended Deterministic and Stochastic Petri Nets (eDSPN) and Stochastic Coloured Petri Nets (SCPN), as well as a class of Stochastic Automata. The plan is to extend TimeNET to also address SDCPN. The main extensions necessary are:

- In SDCPN, the token colours satisfy the solution of a multi-dimensional stochastic differential equation (SDE) while they are residing in their place. A transition that has put the token in a place has also given it an initial multi-dimensional colour, and from that point on the colour follows the SDE without external influence, until the token is removed from the place by a transition.
- In SDCPN, some of the transitions are Guard transitions. From the moment they have at least one token per input place, they continuously look at the colours (values) of these input tokens. When the colours cross some boundary, the Guard is True and the transition is enabled and fires.
- In SDCPN, the delays of the Delay transitions are defined in terms of a firing rate, hence not as a delay time period as in TimeNET. A complementary issue is that TimeNET uses discrete-event simulation. Because SDCPN is continuous-time this requires a numerical approach in following the continuous-time line.
- SDCPN are used to develop Agent-based models. In principle, local SDCPN are developed for each Agent in the operation separately (e.g. aircraft, pilot, communication system, etc). Next the interactions between the agents are modelled by means of arcs. As a short hand notation, a box is drawn around each Agent SDCPN, and arcs may be drawn directly from the edge of a box to the edge of another box.

With these extensions in place, a user can open TimeNET, select the module for SDCPN, and start developing SDCPN-based models for air traffic (or other) applications. During and after the model development, the user will be able to verify if all necessary elements are completed, whether the developed model satisfies all SDCPN formalism rules, whether all parameters have been given values, etc.

In addition, the toolset needs to be extended with layers that allow Monte Carlo simulation of SDCPN-based models, including speed up techniques. An overview of the anticipated TimeNET modelling and simulation layers is given below:

1. The first layer allows the modelling of Local Petri Nets (LPN) according to the rules of SDCPN [52, 65]. Each LPN models a specific part of the air traffic operation, typically a part of an agent. It has a name that can be used as reference in further layers.

2. The second layer allows the modelling of interactions between LPN according to the rules of SDCPN^{imt}, i.e. SDCPN with interconnection mapping types[52, 66]. These interconnection mapping types are shorthand notations that draw a box around each LPN and draw enabling arcs from the edge of one box to the edge of another box, thus enabling the exchange of information between local agents without disturbing their intrinsic internal operation. An interaction may have an effect on the token colour types implemented in the first layer, so there will be iterations by the user between layer 1 and layer 2.

3. The third layer brings everything together as a set up to a simulation (at this stage without any risk decomposition or IPS), by creating the environment that allows to run the model during a particular time interval. For example, in the previous two layers there is an LPN for the behaviour of an aircraft, plus its interactions with other LPNs. But a full simulation run may need hundreds of

aircraft flying in an airspace. So this third layer includes setting a traffic generator that generates copies of the aircraft behaviour model and sends them into the airspace at certain instants, while interacting with other LPN. In this layer it is also determined which output is generated by the simulation. For example, the number of times that two aircraft come within a distance d of each other.

4. This fourth layer displays the result of the previous layer, i.e. a simulation code together with a list of all parameters that can be set by the user. This includes parameters for the start time and end time of a simulation, parameters for the number of aircraft to be simulated, etc. The user can give all parameters a value, and then run the simulation once at a time. This also allows testing and debugging of the model and simulation code.

5. Layers 5 and 6 take the step to Monte Carlo simulations, which means running the simulation of layer 4 many times with stochastic inputs, and collecting the results. For this, the simulation code is extended to enable the user to set the parameters for the simulation of layer 4, plus additional parameters that set the number of Monte Carlo simulation runs. In layer 5, the user is able to use Risk Decomposition by setting the parameters to create conditional circumstances (such as a satellite system being working all the time), to run the Monte Carlo simulation and collect results, and then repeat (e.g. with the satellite system *not* working all the time). The results can be weighted with the probabilities of the satellite system not working / working and summed to get the eventual risk results.

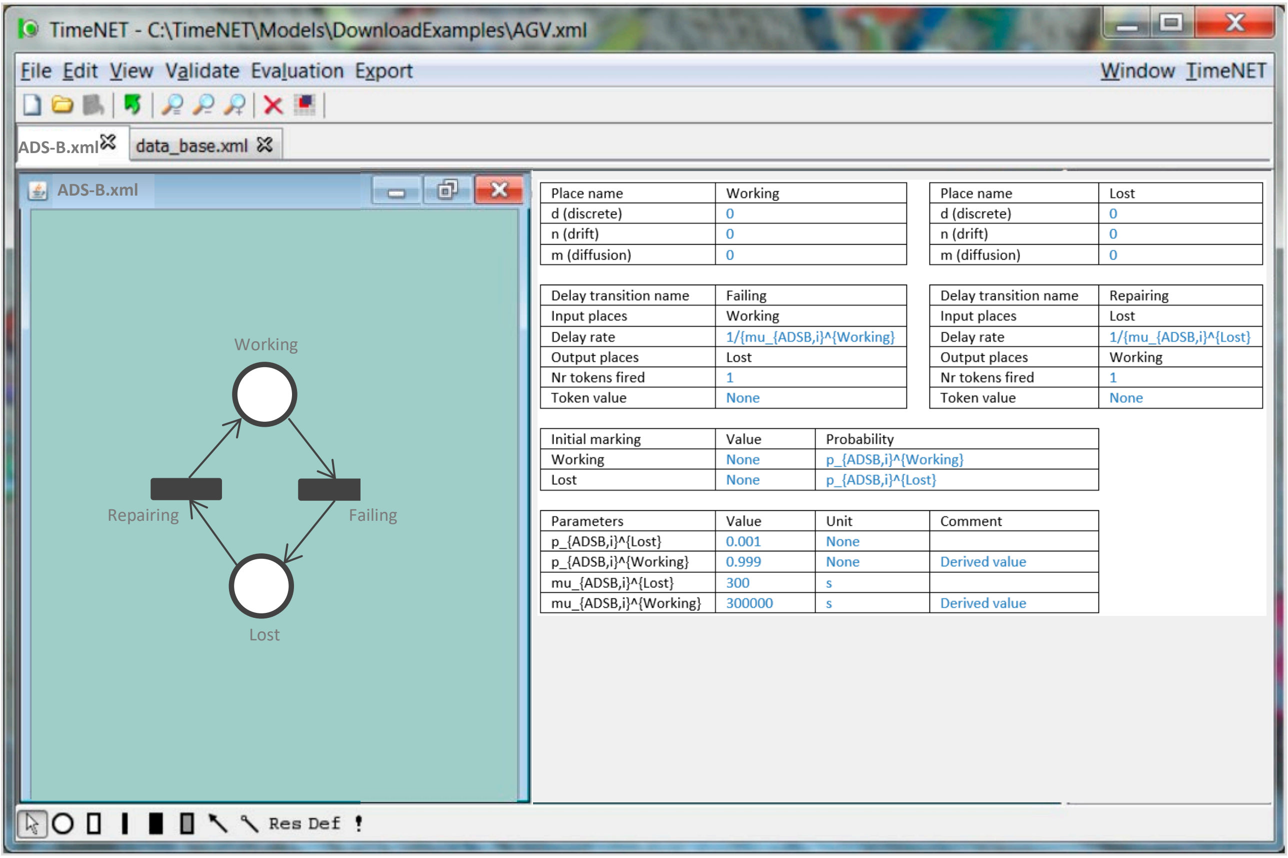
6. This sixth layer is the one for IPS, in which each complete simulation run of layer 4 is referred to as a “particle”. This IPS method makes use of a series of decreasing distances, $d_c < d_{c-1}$, c $= 1, .. m$, between aircraft. On a finite time interval $(0, T)$ the execution of N simulation objects of an air traffic scenario are conducted going through *m* sequential cycles, using the set-up of layer 5. In each cycle N particles are simulated and the cycle is ended if the simulation of each particle has ended, either because a particle has reached the next miss distance boundary, or because a particle has reached the end time of the simulation T (see also Section 12.1.2 of [6]).

7. If the toolset is going to be used by a team of air traffic operation experts (rather than by modelling and simulation experts) who are interested in the risk results, the whole package could be complemented by a final user-friendly layer than allows these experts to set parameter values and run simulations for one particular application, without having to bother with the underlying model.

Appendix A.3 Example local Petri net in extended toolset

The use of the extended TimeNET toolset for D-CRAT will start with the identification of the agents in the operation, together with their main interactions, similar to the description in Section 2.

Next, for each of the agents, a local SDCPN-based model will be developed. This starts with the identification of various discrete modes, the identification of possible switches between modes, and the specification of how and after which time delay the mode switches occur. Next, all continuous-valued elements are added such as flight paths, wind effects, navigation and surveillance data. The model currently developed for D(emo)-CRAT can be easily translated into these terms.


TimeNET - C:\TimeNET\Models\DownloadExamples\AGV.xml
File Edit View Validate Evaluation Export
Window TimeNET
ADS-B.xml
data_base.xml
ADS-B.xml
Working
Repairing
Failing
Lost
Res Def !

Place name	Working
d (discrete)	0
n (drift)	0
m (diffusion)	0

Place name	Lost
d (discrete)	0
n (drift)	0
m (diffusion)	0

Delay transition name	Failing
Input places	Working
Delay rate	1/{mu_{ADSB,i}^{Working}
Output places	Lost
Nr tokens fired	1
Token value	None

Delay transition name	Repairing
Input places	Lost
Delay rate	1/{mu_{ADSB,i}^{Lost}
Output places	Working
Nr tokens fired	1
Token value	None

Initial marking	Value	Probability
Working	None	p_{ADSB,i}^{Working}
Lost	None	p_{ADSB,i}^{Lost}

Parameters	Value	Unit	Comment
p_{ADSB,i}^{Lost}	0.001	None	
p_{ADSB,i}^{Working}	0.999	None	Derived value
mu_{ADSB,i}^{Lost}	300	s	
mu_{ADSB,i}^{Working}	300000	s	Derived value

Figure 32: Concept GUI design (not implemented yet) for specifying an SDCPN-based model in TimeNET

For example, a local Petri net (LPN) for the ADS-B system described in Section 7.7 of [6] can be graphically specified by two places (Working and Lost), two transitions (Failing, Repairing), and connecting arcs, see Figure 32, left-hand-side. If the user double-clicks on one of the places or transitions, a small window is opened (Figure 32, right-hand-side, windows with automatically generated text in black), in which the specifics for that node can be completed by the user (text in blue). In this example, both places in the LPN contain black tokens (the number of discrete, drift and diffusion components are all zero). In case of coloured tokens, additional rows are generated for specification of the details of those colours.

Each such LPN can be saved under a particular name, and recalled again where needed. Interactions between LPN can be specified by recalling two LPN on the screen, drawing a box around them, and drawing arrows from the edge of one box to the edge of another box. In one double clicks on one of the places or transitions in the original LPN, a small window appears like in the previous figure, but additional rows are automatically generated for the new input and output places and transitions due to the interconnections, and the user is able to complete the specifics.

Some SDCPN modelling elements are less conveniently implemented in table-like windows as above, such as complex stochastic differential equations for the behaviour of an aircraft. In such cases, the user fills in a function name in the table-like window, and specifies the specifics for that function in a separate window using, for example, C++ code. It might even be possible to make connections to external software, such as was done in D(emo)-CRAT and DAIDALUS; however, this needs to be confirmed by the TimeNET programmers.

Dedicated to innovation in aerospace

NLR - Royal Netherlands Aerospace Centre

Royal NLR operates as an unaffiliated research centre, working with its partners towards a better world tomorrow. As part of that, Royal NLR offers innovative solutions and technical expertise, creating a strong competitive position for the commercial sector.

Royal NLR has been a centre of expertise for over a century now, with a deep-seated desire to keep innovating. It is an organisation that works to achieve sustainable, safe, efficient and effective aerospace operations.
The combination of in-depth insights into customers' needs, multidisciplinary expertise and state-of-the-art research facilities makes rapid innovation possible. Both domestically and abroad, Royal NLR plays a pivotal role between science, the commercial sector and governmental authorities, bridging the gap between fundamental research and practical applications. Additionally, Royal NLR is one of the large technological institutes (GTIs) that have been collaborating since 2010 in the Netherlands on applied research as part of the TO2 federation.

From its main offices in Amsterdam and Marknesse plus two satellite offices, Royal NLR helps to create a safe and sustainable society. It works with partners on numerous (defence) programmes, including work on complex composite structures for commercial aircraft and on goal-oriented use of the F-35 fighter. Additionally, Royal NLR helps to achieve both Dutch and European goals and climate objectives in line with the Luchtvaartnota (Aviation Policy Document), the European Green Deal and Flightpath 2050, and by participating in programs such as Clean Sky and SESAR.

For more information visit: **www.nlr.org**

Postal address
PO Box 90502
1006 BM Amsterdam, The Netherlands
e) info@nlr.nl **i)** www.nlr.org

NLR Amsterdam
Anthony Fokkerweg 2
1059 CM Amsterdam, The Netherlands
p) +31 88 511 3113

NLR Marknesse
Voorsterweg 31
8316 PR Marknesse, The Netherlands
p) +31 88 511 4444

NLR is a registered trade name of Stichting Nationaal Lucht- en Ruimtevaartlaboratorium, Chamber of Commerce No. 41150373. VAT No. NL002760551B01