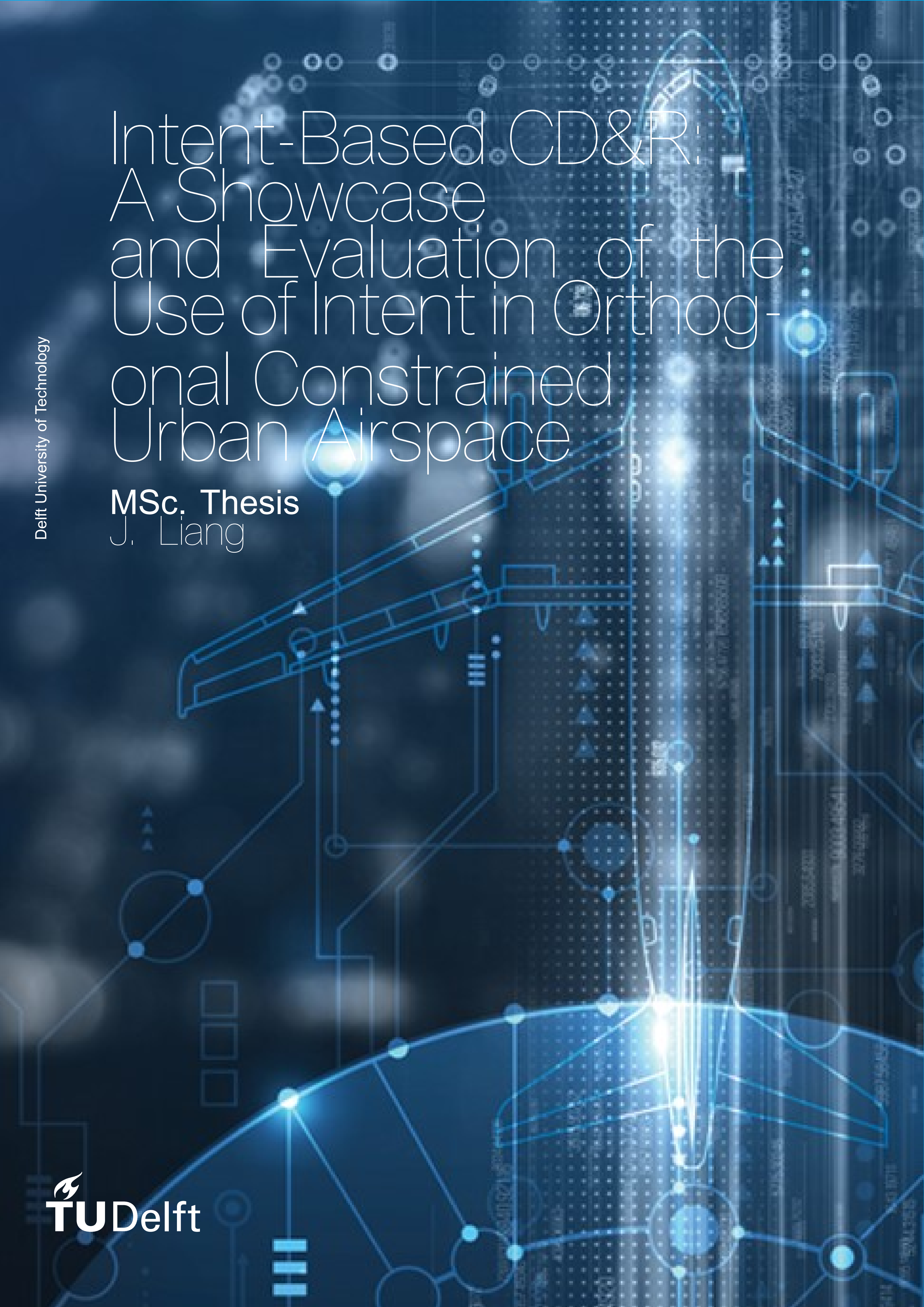


Intent-Based CD&R: A Showcase and Evaluation of the Use of Intent in Orthogonal Constrained Urban Airspace

MSc. Thesis

J. Liang

Delft University of Technology

TU Delft

Intent-Based CD&R: A Showcase and Evaluation of the Use of Intent in Orthogonal Constrained Urban Airspace

MSc. Thesis

by

J. Liang

to obtain the degree of
Master of Science in Aerospace Engineering
at the Delft University of Technology.

Student number:	4846222	
Project duration:	January 9, 2023 – February 2, 2024	
Thesis committee:	Prof. dr. ir. J. M. Hoekstra,	TU Delft, supervisor
	Dr. Ir. J. Ellerbroek,	TU Delft, supervisor
	Dr. O. A. Sharpans'kykh,	TU Delft, additional member
	Ir. C. A. Badea,	TU Delft, external member

An electronic version of this thesis is available at http://repository.tudelft.nl/.
Cover image obtained from: (https://www.flightsafetyaustralia.com/2020/05/pick-a-number-big-data-artificial-intelligence-and-aviation/)

TU Delft

Preface

This MSc. thesis is written to conclude my Master of Science in Aerospace Engineering at the Delft University of Technology. The project concerns the development and evaluation of an intent-based method for usage in orthogonal constrained urban airspace. This thesis consists of: 1) a scientific paper, 2) appendices to the scientific paper, and 3) a preliminary report.

Although it was at times hard, I am happy to have been able to work on this topic. I have learned a lot throughout my thesis, not only on the topic but about myself as well. I would like to thank all three of my supervisors, Andrei, Joost and Jacco for all the help that I have received throughout the whole project. I would like to thank Andrei specifically for all the help received in dealing with Bluesky and lending an ear to all the problems that I encountered.

Moreover, I would like to thank everyone who has supported and accompanied me during my times of being a MSc student. This includes my friends, my family and all the other amazing people that I have had the pleasure of meeting over the years. Now my time as a MSc. student can be concluded and I am excited to see what future awaits.

J. Liang
Delft, February 2024

Contents

List of Figures vii

List of Tables viii

Nomenclature ix

I Scientific Paper 1

II Appendices to Scientific Paper 16

A Conflict Plots 18
- A.1 Resolution Disabled . . . 18
- A.2 Resolution Active . . . 21

B Losses of separation Plots 25

C False Positive Rate Plots 29

D Average Detection Time Before Conflict Plots 33

III Preliminary Report (Already Graded) 36

1 Introduction 38
- 1.1 Research Questions . . . 38
- 1.2 Report Outline . . . 39

2 Literature Review 40
- 2.1 Basic Concepts & Definitions . . . 40
  - 2.1.1 UAV . . . 40
- 2.2 The Future of Airspace . . . 40
  - 2.2.1 Airspace. . . . 40
  - 2.2.2 U-space . . . 42
  - 2.2.3 Unmanned Aircraft System Traffic Management . . . 43
  - 2.2.4 Urban Airspace Designs . . . 44
  - 2.2.5 Constrained Airspace Designs. . . . 45
  - 2.2.6 Urban Airspace Centralization . . . 47
  - 2.2.7 Conclusion . . . 48
- 2.3 CD&R . . . 48
  - 2.3.1 Conflict . . . 48
  - 2.3.2 CD&R Taxonomy . . . 49
  - 2.3.3 State-Based Conflict Detection . . . 51
  - 2.3.4 State-Based Conflict Resolution . . . 52
  - 2.3.5 Solution Space Diagram . . . 52
  - 2.3.6 Modified Voltage Potential . . . 55
  - 2.3.7 Conclusion . . . 55
- 2.4 Intent . . . 56
  - 2.4.1 The Use of Intent Information in Conflict Detection and Resolution Models Based on Dynamic Velocity Obstacles . . . 56
  - 2.4.2 The Effect of Intent on Conflict Detection and Resolution at High Traffic Densities . . . 57
  - 2.4.3 Unifying Tactical Conflict Prevention, Detection, and Resolution Methods in Non-Orthogonal Constrained Urban Airspace . . . 58
  - 2.4.4 Conclusion . . . 59
- 2.5 Conclusion . . . 59

3 Experimental Setup **61**
3.1 Framework . . . 61
3.2 Background . . . 61
3.3 List of Assumptions . . . 62
3.4 Methodology . . . 63
3.4.1 Airspace design . . . 63
3.4.2 Conflict Detection . . . 63
3.4.3 Trajectory calculation . . . 64
3.4.4 Conflict Resolution . . . 66
3.5 Experiment Setup . . . 66
3.5.1 Simulation . . . 66
3.5.2 Independent Variables . . . 67
3.5.3 Dependent Variables . . . 68
3.5.4 Hypotheses . . . 68

4 Project Timeline **69**
4.1 Work packages . . . 69
4.1.1 Airspace Design . . . 69
4.1.2 Conflict Detection . . . 69
4.1.3 Conflict resolution . . . 70
4.1.4 Experiment . . . 70
4.1.5 Reporting . . . 71
4.2 Gantt Chart . . . 71

List of Figures

A.1 Number of conflicts per traffic density with resolution disabled . . . 18
A.2 Number of conflicts per traffic density with resolution disabled, Intent-based method . . . 19
A.3 Number of conflicts per traffic density with resolution disabled, State-based method . . . 19
A.4 Number of conflicts per traffic density with resolution disabled and a look-ahead time of 10 seconds . . . 20
A.5 Number of conflicts per traffic density with resolution disabled and a look-ahead time of 20 seconds . . . 20
A.6 Number of conflicts per traffic density with resolution disabled and a look-ahead time of 30 seconds . . . 21
A.7 Number of conflicts per traffic density with resolution active . . . 21
A.8 Number of conflicts per traffic density with resolution active, Intent-based method . . . 22
A.9 Number of conflicts per traffic density with resolution active, State-based method . . . 22
A.10 Number of conflicts per traffic density with resolution active and a look-ahead time of 10 seconds . . . 23
A.11 Number of conflicts per traffic density with resolution active and a look-ahead time of 20 seconds . . . 23
A.12 Number of conflicts per traffic density with resolution active and a look-ahead time of 30 seconds . . . 24

B.1 Number of losses of separation per traffic density . . . 25
B.2 Number of losses of separation per traffic density, Intent-based method . . . 26
B.3 Number of losses of separation per traffic density, State-based method . . . 26
B.4 Number of losses of separation per traffic density with a look-ahead time of 10 seconds . 27
B.5 Number of losses of separation per traffic density with a look-ahead time of 20 seconds . 27
B.6 Number of losses of separation per traffic density with a look-ahead time of 30 seconds . 28

C.1 False positive rate per traffic density . . . 29
C.2 False positive rate per traffic density, Intent-based method . . . 30
C.3 False positive rate per traffic density, State-based method . . . 30
C.4 False positive rate per traffic density with a look-ahead time of 10 seconds . . . 31
C.5 False positive rate per traffic density with a look-ahead time of 20 seconds . . . 31
C.6 False positive rate per traffic density with a look-ahead time of 30 seconds . . . 32

D.1 Average detection time per traffic density with a look-ahead time of 10 seconds . . . 33
D.2 Average detection time per traffic density with a look-ahead time of 20 seconds . . . 34
D.3 Average detection time per traffic density with a look-ahead time of 30 seconds . . . 34

2.1 Overview airspace classification. From [40] . . . 41
2.2 Graphical Overview Airspace. From [12] . . . 42
2.3 Roadmap of the U-Space Project. From [39] . . . 43
2.4 Technical Capability Levels. From [24] . . . 44
2.5 Top-down view of the Zones topology, which is designed to take into account the layout of a city. From [43] . . . 45

2.6 Schematic view for the complete set of altitude bands for the two urban airspace designs, where each altitude layer corresponds to its respective travel direction. Each concept has 40 altitude layers (which consist of 20 altitude layers allocated for through traffic and 20 altitude layers for turn traffic) from 75 to 1050 ft with a vertical spacing of 25 ft, respectively. The turn-layers are used for transitory flights, that is, drones that need to make turns/change direction at intersections. While the through-layers are utilised by through traffic, that is traffic passing through at least one intersection. (a) The two-way concept layer system for which traffic is allocated to the cardinal directions with respect to flight headings (ψ): 315◦ < ψ ≤ 045◦ headings assigned to north; 045◦ < ψ ≤ 135◦ headings to east; 135◦ < ψ ≤ 225◦ headings to south; and 225◦ < ψ ≤ 315◦ to west bound traffic. (b) One-way layer system where traffic with flight headings: 315◦ < ψ ≤ 045◦ and 135◦ < ψ ≤ 225◦ are assigned to north and south layers; and, traffic with flight headings between 045◦ < ψ ≤ 135◦ and 225◦ < ψ ≤ 315◦ are allocated to the east and west bound altitude layers. Of note, recognise that, when compared to the two-way concept, the one-way concept has north/south and east/west traffic assigned to one altitude layer, for which the separation of this traffic is assured by the spatial geometry of the urban street network. From [11] 46
2.7 The layer configurations used within constrained airspace. (a) Isometric view of layers at intersections, configured such that cruising aircraft do not interfere with each other, and turn layers are always at the same altitude; (b) Layer configuration in function of street categorization. Certain altitudes are reserved for cruising either in the east/west or north/south directions. Each layer is 30 ft in height. From [6] 47
2.8 Conflict Scenario adapted from [19] 51
2.9 Velocity Obstacle from [19] 52
2.10 In a situation with a foreseen turn, the SSD can be approximated using three phantom aircraft. From [9] 53
2.11 Constructing the tangent-based SSD, showing the effect of a turn. From [9] 54
2.12 The construction of the SSD in (c) by combining the sets of FV in (a) and the RV in (b) from [7] 55
2.13 Geometry of modified voltage potential resolution method 56
2.14 Projection-based conflict detection method: if intersecting paths are detected, the intruder is projected linearly from the intersection point, and a linear detection algorithm is applied: (a) Intersecting intent information is detected between two aircraft in constrained airspace; (b) Aircraft states are projected linearly from the intersection point (dashed lines) such that a state-based conflict detection method can be applied. From [6] 58
2.15 Velocity obstacle methods calculate the required change (Δv) in relative speed between two aircraft (v_{rel}) in order for the minimum separation distance (defined by the radius of the protection zone R_{pz}) to not be breached. From [6] 59

3.1 Research Framework 61
3.2 The layer configurations used within constrained airspace. Each layer is 30 ft in height. Grey are unused layers. (a) Red are layers reserved for travelling in east/west direction. (b) Blue are layers reserved for travelling in North/South direction. 63
3.3 A horizontal conflict scenario of drone 1 (green) and drone 2 (red). As can be seen the trajectory of drone 2 can be divided up into characteristic sections and predict the position of the drone for each measurement. A potential conflict is detected at measurement T4 as the drones are located to close to each other. 64
3.4 The conflict resolution algorithm 66

List of Tables

2.1 Definitions according to Granshaw[17] and SKYbrary[42] 40
2.2 Definitions CD&R 49
2.3 Conflict Detection Categories. From [33] 49
2.4 Conflict Resolution Categories. From [33] 49
2.5 Coordination Rulesets with labels and descriptions from [7] 55

3.1 Specification drones 67
3.2 Traffic density characteristics of the three demand scenarios for the simulation area of Manhattan, New York City, network consisting of an area of 59.1 km^2. From [11]. 67

List of Symbols

The next list describes several symbols and abbreviations that will be later used within the body of the document

Latin Symbols

a acceleration

g acceleration due to gravity

r turn radius

s distance

$s_{acelerate}$ length of acceleration sub-leg

$s_{decelerate}$ length of deceleration sub-leg

s_{leg} leg length

s_{turn} length of turn

t time

V_{cr} cruise speed

V_{turn} turn speed

Greek Symbols

$\Delta\Psi$ change in heading angle

ϕ bank angle

Abbreviations

ADS-B Automatic Dependent Surveillance–Broadcast

ANS Air Navigation Service

ARV Allowed Reachable Velocities

ATC Air Traffic Control

ATM Air Traffic Management

CD&R Conflict Detection and Resolution

CPA Closest Point of Approach

FAA Federal Aviation Administration

FRV Forbidden Reachable Velocities

FV Forbidden Velocities

H High

L Low

LOS Loss of Separation

MH Medium high

ML Medium low

MVP Modified Voltage Potential

NASA National Aeronautics and Space Administration

RTT Research Transition Team

RV Reachable Velocities

SESAR JU Single European Sky ATM Research Joint Undertaking

SSD Solution Space Diagram

TCL Technical Capability Level

TCP Trajectory Change Points

UAV Unmanned Aerial Vehicle

UTM Unmanned Aircraft System Traffic Management

VH Very high

VL Very low

VO Velocity Obstacle

I

Scientific Paper

Intent-Based CD&R: A Showcase and Evaluation of the Use of Intent in Orthogonal Constrained Urban Airspace

J. Liang (*Author*), C.A. Badea (*Supervisor*), J. Ellerbroek (*Supervisor*), J.M. Hoekstra (*Supervisor*)
Control and Simulation, Faculty of Aerospace Engineering
Delft University of Technology, Delft, The Netherlands

***Abstract*- The usage of drones in urban environments is expected to grow rapidly in the coming decades. To ensure the safe operations of drones, conflict detection and resolution are vital. Currently, a lot of research has gone into state-based CD&R, which has proven effective in unconstrained airspace but suffers from a large number of false positive conflicts in constrained airspace. The use of intent in constrained CD&R has the potential to reduce the number of false positive conflicts and improve the safety of drone operations significantly. In this paper, an intent-based detection and resolution method for orthogonal constrained very low-level urban airspace is presented and evaluated against a state-based method. The intent-based method calculates the future position along the trajectory at a time interval of 3 seconds for each aircraft, and conflicts are then detected by comparing these positions. The conflicts are solved utilizing a rule-based algorithm. The results show that the intent-based method has a much lower false positive rate for all traffic densities, as well as a higher average detection time before conflict for larger look-ahead times compared to the state-based method. The resolution of the state-based method, however, shows better performance with fewer losses of separation occurrences. With improvements, the intent-based method's low false positive rate, combined with the use of a larger look-ahead time, allows conflicts to be detected more reliably and earlier than the state-based method, thereby facilitating earlier conflict resolution and enhancing safety.**

I. INTRODUCTION

The utilization of drones in civilian airspace is expected to grow at a substantial rate in the upcoming decades [1]. Projections show that approximately 7 million recreational drones and 400,000 commercial or government-operated drones will be occupying European airspace by 2050. These drones will be used in a wide range of tasks in urban airspace, such as urban surveillance [2], urban air mobility [3], or package delivery [4]. To support the increasing number of drones and the tasks that these perform, it is crucial that focus is put on the safety and efficiency in the operation of drones, especially in complex and crowded urban environments.

To ensure safe operations, unmanned aerial vehicles (UAVs) need to be able to detect conflicts with other UAVs and resolve these conflicts. A lot of research has already been performed on conflict detection and resolution (CD&R) in urban airspace [5], [6]. However, these have mainly been focused on the usage of states and extrapolating these to find conflicts and resolve them. Although this method has proved very efficient in non-constrained airspace [7], the performance of this method in constrained airspace is lackluster [8]. An alternative that shows a lot of potential is the use of intent in constrained urban airspace [9], [10], [11]. By using intent, the future trajectory of aircraft is taken into account in the conflict detection process, potentially reducing the number of false-positive detections. This allows for better route efficiency as aircraft do not have to resolve false-positive conflicts, avoiding unnecessary delays. In addition, the use of intent can also help avoid late detections after UAVs make a last-second turn into the path of another UAV, increasing the safety of UAV operations [11].

The aim of this paper is to present and evaluate an intent-based detection and resolution method for orthogonal constrained very low-level urban airspace. In this paper, the implementation of an intent-based CD&R method is demonstrated and compared to the performance of a baseline state-based CD&R method. In section II, relevant background information on the topic is given. section III shows the methodology behind the intent-based CD&R method. In section IV, the design of the experiment is elaborated on. The results of the experiment can be found in section V. section VI shows a discussion of the results, and in section VII, a conclusion is made.

II. BACKGROUND

In this section the relevant background information for this topic are described.

A. Basic Concepts

The relevant concepts on this topic are described in this subsection. The first important definition is that of a conflict. A conflict can be described as a predicted loss of separation in the future between two aircraft. The goal of a conflict detection and resolution system is then "to predict that a conflict is going to occur in the future, communicate the detected conflict to a human operator, and, in some cases, assist in the resolution of the conflict situation" [12]. In the case of automated unmanned aerial vehicles, no human pilots are present and the resolution is performed by the system. The protected zone of an aircraft

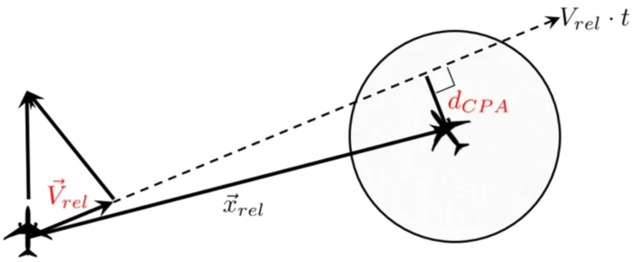


Fig. 1. Calculating closest point of approach. The circle indicates the separation minimum as a protected zone. Taken from [7]

can be described as a zone around an aircraft described by horizontal and vertical separation minima which may not be breached by any other aircraft [13]. If this protected zone is breached, a loss of separation (LoS) has occured. Lastly, intent can be defined as ”the flight path and associated flight data describing the planned trajectory of a flight to its destination, as updated at any moment” [14].

B. State-based detection

To understand the concept of state-based detection, the term closest point of approach (CPA) first needs to be explained. CPA can be described as ”an estimated point in which the distance between the own ship and another object target will reach the minimum value.” [15]. A state-based detection system takes the aircraft's current states and uses those to calculate the CPA to other aircraft as displayed in Figure 1 [7]. If the CPA is smaller than the separation minima, the aircraft is in conflict.

III. Methodology

A. Conflict detection

1) Trajectory calculations: Firstly, each aircraft calculates its trajectory with a certain look-ahead time. The position of the aircraft along its trajectory is calculated via a heuristic algorithm. The trajectory is divided up into legs between each waypoint. Based on the characteristics of the leg, the time to traverse the leg is found. Due to the fact that speed changes are only given before and after turns, there are in total four different types of legs. A visualization of each leg is shown in Figure 2.

1) Cruise leg
2) Pre-turn leg
3) Post-turn leg
4) In-between-turns leg

Case 1 is a leg where the aircraft comes into the leg at cruising speed and keeps travelling at cruising speed until exiting the leg. The time it takes to cover the cruise leg is calculated using Equation 1.

$$t = \frac{s_{leg}}{V_{cr}} \tag{1}$$

Case 2 involves an aircraft entering the leg at cruising speed and ending the leg before the turn. To find the time to cover

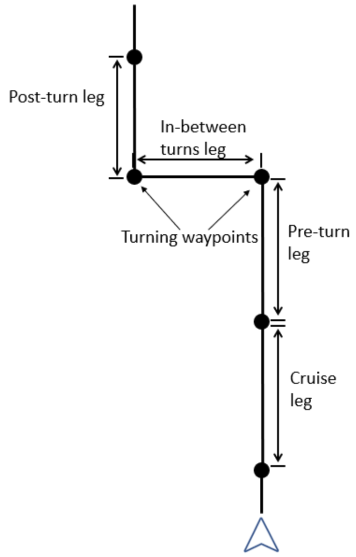


Fig. 2. Example of the four different leg types

this leg it is vital to find the distance before the waypoint where the aircraft starts turning. This can be determined using the parameters ϕ and $\Delta\Psi$ which are the banking angle and the heading change respectively.

$$\begin{aligned} r &= \frac{V_{turn}^2}{gtan(\phi)} \\ s_{turn} &= rtan(0.5\Delta\Psi) \end{aligned} \tag{2}$$

The leg can then further be divided into two sub-legs, namely decelerating and cruising. The time it takes to complete each sub-leg can then be calculated as well as the total time.

$$\begin{aligned} s_{decelerate} &= \frac{0.5|V_{turn}^2 - V_{cr}^2|}{a_{max}} \\ t_{decelerate} &= \frac{V_{turn} - V_{cr}}{a_{max}} \\ t_{cruise} &= \frac{s_{leg} - s_{decelerate} - s_{turn}}{V_{cr}} \\ t_{total} &= t_{decelerate} + t_{cruise} \end{aligned} \tag{3}$$

In case 3, the drone makes the turn and accelerates to reach cruising speed before it exits the leg. This leg can once again be divided into sub-legs. These are the turn, the acceleration and the cruise sub-legs. The calculations of the turn radius are the same as presented in Equation 2. The time to cover the turn can be found using Equation 4.

$$\begin{aligned} s_{turning} &= 2\pi r\frac{\Delta\Psi}{360} \\ t_{turn} &= \frac{s_{turn}}{V_{turn}} \end{aligned} \tag{4}$$

The acceleration can be calculated as follows

$$s_{accelerate} = \frac{0.5|V_{cr}^2 - V_{turn}^2|}{a_{max}}$$
$$t_{accelerate} = \frac{V_{cr} - V_{turn}}{a_{max}} \tag{5}$$

Lastly, the cruising time is found. Of note is the fact that the turning speed is used for the cruise part instead of the cruising speed. This is due to the fact that speed change commands are only given at the waypoints, hence the aircraft will only start accelerating when the distance between the waypoint and the aircraft is equal to $s_{accelerate}$. In addition, s_{turn} is once again used as the leg does not start from the waypoint, but from the point at which the aircraft ends the turn which is a certain distance away from the starting waypoint.

$$t_{cruise} = \frac{s_{leg} - s_{accelerate} - s_{turn}}{V_{turn}}$$
$$t_{total} = t_{turn} + t_{accelerate} + t_{cruise} \tag{6}$$

Case 4 is a rare case where the leg is situated between two turns which are happening back to back. The leg can once again be subdivided into two sub-legs which are the turn and cruise sub-leg.

$$s_{turn} = 2\pi r \frac{\Delta\Psi}{360}$$
$$t_{turn} = \frac{s_{turn}}{V_{turn}} \tag{7}$$

Once again, the cruise consists of the turn speed, due to the fact that the speed change commands are only given at the waypoints. Considering that the next waypoint is also a turn waypoint, the speed will remain turn speed and hence will not change. Furthermore, since another turn is performed at the next waypoint, the distance to the waypoint where the aircraft starts turning needs to be subtracted from the total leg distance.

$$t_{cruise} = \frac{s_{leg} - s_{turn1} - s_{turn2}}{V_{turn}}$$
$$t_{total} = t_{turn} + t_{cruise} \tag{8}$$

To find the position of the aircraft after a certain time, the leg times are added together until it matches the time or exceeds it. If it exceeds it, the last leg type is taken and a backwards calculation is performed based on by how much the time was overshot. These calculations are different for each leg type.

The calculations for case 1 are demonstrated in Equation 9 where t_{over} is the overshoot time.

$$s = V_{cr} \cdot t_{over} \tag{9}$$

For case 2, depending on the overshoot time, the aircraft may be in the deceleration part or the cruise part as displayed in Figure 3. To find out in which part the drone is located, the overshoot time is used and compared to the sub-leg times. The following calculations can then be determined based on these times.

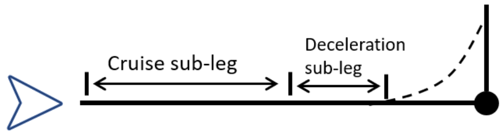


Fig. 3. Cruise and decleration phase of a pre-turn leg

$$s = \begin{cases} 0.5a_{max}t_{over}^2 + V_{turn}t_{over}, & \textit{if } t_{over} \leq t_{dec} \\ V_{cr}(t_{over} - t_{dec}) + & \textit{if } t_{over} > t_{dec} \\ 0.5a_{max}t_{dec}^2 + V_{turn}t_{dec} & \end{cases}$$

Case 3 is once again similar to case 2 except for an additional sub-leg being present leading to three conditions in total. The division of sub-legs is visualized in Figure 4. It is rather difficult to calculate the position of the drone during the turn due to issues in determining the position of the circle which the turn is made up of as well as finding the position of the aircraft on this circle during the turn. This is why for simplicity the latitude and longitude of the turning waypoint is taken as the position in case the aircraft is located within the turn sub-leg. Then, the following two conditions remain.

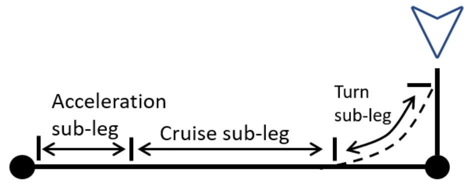


Fig. 4. Cruise, acceleration and turn phase of a post-turn leg

$$s = \begin{cases} 0.5a_{max}t_{over}^2 + V_{turn}t_{over}, & \textit{if } t_{over} \leq t_{acc} \\ V_{cr}(t_{over} - t_{acc}) + & \textit{if } t_{acc} \leq t_{over} \\ 0.5a_{max}t_{acc}^2 + V_{turn}t_{acc} & < t_{acc} + t_{cr} \end{cases}$$

The equation for case 4 is displayed in Equation 10. If the aircraft is located within the turn sub-leg, the latitude and longitude position of the first turning waypoint is taken. Then there is only one case remaining.

$$s = V_{turn}t_{over}, \textit{if } t_{over} \leq t_{cr} \tag{10}$$

After obtaining all the distances, the last waypoint is taken and the latitude and longitude are obtained by subtracting the calculated distance from the last waypoint along the leg.

2) Trajectory comparison: The detection algorithm makes use of a look-ahead time and time intervals. The look-ahead time is the time within which the detection method is able to detect conflicts. If a conflict happens at a time later than the look-ahead time, the conflict is not detected. The position of the aircraft is calculated for a regular time interval within the look-ahead time. Since a protected zone radius is used of 50 meters and the maximum speed at which aircraft travel during the experiment is 10 m/s, a time interval of 2 seconds is chosen. This time interval ensures that no conflicts go by unnoticed as the maximum distance the aircraft is able to travel within this time is less than half the protected zone radius, meaning that in the worst case scenario where two aircraft are going towards each other, the conflict is still detected.

The position of the aircraft is then calculated at an interval of 2 seconds. The total amount of data points for each aircraft will then be the look-ahead time divided by the interval time. Each data point consists of the predicted latitude and longitude position of the aircraft at the specified time in the future. The data of an aircraft with a look-ahead time of 10 seconds and with the specified time interval would then look like this. Where the position of Lat_0 and Lon_0 indicate the position of the aircraft 2 seconds in the future and the position of Lat_4 and Lon_4 10 seconds in the future.

$$data: [[Lat_0, Lon_0], [Lat_1, Lon_1], [Lat_2, Lon_2], [Lat_3, Lon_3], [Lat_4, Lon_4]]$$

The first data point is then used to compute the distance between the ownship to the position of all other aircraft at the first data point. If the distance between the aircraft falls below the radius of the protected zone, the two aircraft are considered in conflict. This process is repeated for all data points until all conflicts are marked.

One issue that arises when using this method to detect conflicts, is the fact that aircraft travelling on parallel roads might be within each others protected zone radius, but are technically not in conflict because they are on different roads. To prevent these false conflicts from being detected, the trajectories of the aircraft for each conflict are analysed. Their trajectories are propagated 300 meters ahead and if both trajectories at one point intersect, the conflict is considered real. If not, the conflict is removed and both aircraft are not in conflict anymore.

B. Conflict Resolution

1) Resolution Manoeuvre: The conflict resolution method is a rule-based algorithm. The resolution algorithm checks for multiple conflict conditions to determine which resolution manoeuvre to apply and which aircraft has priority. Since each conflict pair is handled separately, multiple resolution manoeuvres may have been assigned to one aircraft. This is why a resolution array is created which saves all the resolution manoeuvres of an aircraft. The lowest speed in the array is selected to attempt to resolve all conflicting situations.

Firstly, the resolution method checks for back-to-back conflicts. These are conflicts where one aircraft is ahead of the other while travelling on the same road. Two ways are implemented to check whether conflict happens back-to-back. The first way is to check the ownship's forward trajectory up to a distance of 300 meters. If the intruder is located somewhere along that trajectory, the conflict is a back-to-back conflict with the ownship being in the back. The second way is heading-based where the ownship first checks whether the aircraft is in front. If the intruder falls within a heading of -20 to 20 degrees relative to the front of the ownship, the intruder is ahead. The next step is then to check the difference in heading of the ownship and intruder to see if they are going to the same direction. If this heading difference check falls under 20 degrees, then the conflict is a back-to-back conflict with the ownship in the back.

If one of these cases occur, a resolution through a change in speed is given. The speed command is only given to the trailing aircraft in a back-to-back conflict. If the trailing aircraft is located at a distance of more than 2.5 times the radius of the protected zone, the speed command passes a speed equal to the aircraft in front of the trailing aircraft. The distance of 2.5 times is chosen as a safety margin to ensure that no LoS happens in case the front aircraft suddenly comes to a stop. This distance gives enough time for the trailing aircraft to first detect that the aircraft in front has slowed down and for the aircraft to slow itself down as well without causing a LoS. If the distance between the two conflicting aircraft is less than 2.5 times the protected zone radius, the aircraft is slowed down until the distance between the aircraft is 2.5 times the protected zone radius.

If there is no back-to-back conflict, the resolution methods determines who has priority based on the distance to the point of conflict. This distance is found by looking at the waypoint the aircraft was travelling to when a conflict was found. The distance to this waypoint via the aircraft's trajectory is calculated for each aircraft and the aircraft with a lower distance has priority as it is closer to the conflict resulting in less delay time. The aircraft without priority slows down until the conflict is resolved.

Lastly, if no back-to-back conflict is detected, and the distance to the point of conflict is equal, the resolution method gives priority to the aircraft which has been in the air the longest aircraft as it is at higher risk of running out of fuel or energy. The aircraft without priority slows down until the conflict is resolved.

An issue observed in test runs involves the occurrence of deadlocks, particularly in higher traffic densities. These arose due to multi aircraft conflicts where all involved aircraft slow down to a halt and wait for each other to pass before continue flying.For this reason a bypass was implemented to get rid of these deadlocks and to ensure that the simulation does not come to a stop at the cost of LoSs. Whenever two aircraft in conflict have both stopped, the aircraft which has been the longest in the air can resume its autopilot.

IV. Experiment Design

A. Simulator

The experiment is simulated making use of BlueSky Open Air Traffic Simulator. The simulator is built using Python 3 and uses open-source data for its navaids, aircraft performance and geography [16]. The proposed conflict detection and resolution methods have been implemented as plugins on this platform.

B. Aircraft Models

The aircraft model that is used in the simulations is the DJI 600 Matrice Pro hexacopter. The relevant specifications are displayed in Table I. These specifications are taken from [9] and [17] with the exception of the turning speed.

TABLE I
AIRCRAFT SPECIFICATIONS

Specification	Value
Max horizontal speed (m/s)	18
Average cruise speed (m/s)	10
Average turning speed (m/s)	2.5
Max take-off mass (kg)	15
Max acceleration/deceleration (m/s^2)	3.5
Maximum flight time without load (minutes)	32

C. Airspace Design

The airspace design is a 2D design where aircraft are only allowed to fly above the streets. The authors of [18] argue that this is the best travel method due to lower privacy risks and the ability to combine different transport modes like drones and trucks for last mile delivery. In addition, the streets are uni-directional meaning traffic is only allowed to travel one way for every street. This choice is made based on the higher safety provided by uni-directional streets compared to bi-directional streets [18].

D. Street Network Design

The street network that is chosen is that of Manhattan, New York City, United States of America with a surface area of $59,1km^2$. This street network has been chosen for its mainly orthogonal design which allows for less complicated trajectory calculations. Moreover, orthogonal designs gives more certainty and less variation and is ideal for state-based detection methods. This gives a fairer comparison between intent-based and state-based conflict detection.

The street network is obtained using OSMNx which is a Python package that allows geospatial data to be downloaded [19]. This street network is then further processed to fit the experiment. The first thing that is done is removing complicated sections of the street network as well as designing it such that the traffic is spread evenly across the city instead of having bottleneck streets.

The streets are then divided into long strokes similar to the method used in [8] and will be used as input for the genetic algorithm. The strokes are created using the COINS algorithm [20]. After the COINS algorithm has been implemented, some small manual changes are made to ensure that roads do not become too long, making it harder to find routes between nodes. Furthermore, strokes which at one point make a turn of $90°$ or more are also split at the turn.

A genetic algorithm is then applied to obtain the most efficient street network by switching around the direction of the strokes. The genetic algorithm that is used is taken from Badea et al. [8]. This genetic algorithm uses cost minimization to ensure the highest level of connectivity within the street network.

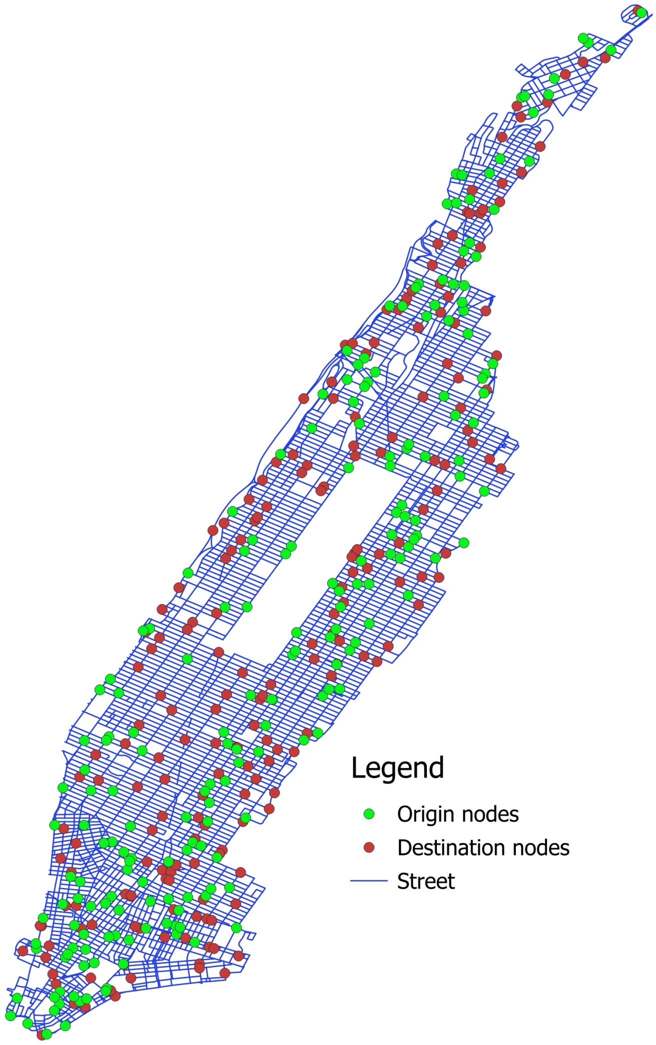


Fig. 5. Street network of Manhattan with the origin and destination nodes used in the experiment

E. Missions

The mission type that is flown are point to point. There are in total 200 origin nodes and 200 destination nodes which have been randomly generated. The map of the street network of Manhattan with the origin and destination nodes can be viewed in Figure 5. The routes between each origin and destination node has been generated pre-flight using the Dijkstra algorithm from [21] to find the shortest path possible. The minimum distance of a route is 1 kilometer and the farthest distance between an origin and destination node is 6 kilometers. This does not mean that the maximum distance is 6 kilometers as the aircraft follows the street network and does not directly go to the destination node. A limit of 6 kilometers has been chosen as it is a realistic estimate of the maximum flight time and distance that can safely be flown with the specified aircraft. An aircraft is deleted from the simulation once it reaches its destination. The simulation itself will run for an hour.

F. State-based method

In the experiment the intent-based method is compared to a state-based method. The state-based method that is used is one designed by Badea, Ellerbroek, and Hoekstra [22]. This state-based method resolves conflict by making speed changes or altitude changes. The altitude resolution manoeuvres have been disabled as this experiment is performed in a 2D environment.

G. Independent Variables

There are in total three different independent variables.

1) **Detection Method**: In this paper, the trajectory-based detection method will be compared to a baseline which utilizes a state-based detection method.
2) **Traffic Density**: Since traffic density can have a significant impact on the performance of the detection methods, this has also been chosen as an independent variable. The different traffic densities are displayed in Table II and have been deducted from [23] adjusting for the difference in size between Paris and Manhattan. The traffic density indicates the number of aircraft that are simultaneously in the air at any given time.
3) **Look-ahead Time**: Another independent variable is the look-ahead time. This parameter also has a big influence on the performance. Especially for state-based detection methods a bigger look-ahead time usually corresponds to more false positives. The different look-ahead times are displayed in Table III

Each combination of independent variables is simulated a total of 10 times. In addition, each simulation is run once with resolution active, and once with resolution disabled. This is done to obtain information surrounding LoSs, if they occur and when they occur. This comes to a total of 720 simulations.

TABLE II
NUMBER OF AIRCRAFT CONCURRENTLY IN FLIGHT FOR EACH TRAFFIC DENSITY

	No. Aircraft
Very low (VL)	40
Low (L)	60
Medium low (ML)	80
Medium high (MH)	100
High (H)	120
Very high (VH)	140

TABLE III
LOOK-AHEAD TIMES USED IN THE EXPERIMENT

Look-ahead time	Time in seconds
Small	10
Medium	20
Large	30

H. Dependent variables

In total four dependent variables are measured during the simulations.

- **Number of Conflicts**: is the number of conflicts that have been detected over the whole simulation. The number of conflicts is logged for simulations with and without the resolution active.
- **Number of LoSs**: is the number of LoSs that have been detected over the whole simulation. This number is logged for simulations where the resolution is active
- **False Positive Rate**: is the number of false positive conflicts divided by the total number of conflicts. The number of False postive conflicts are found by running the simulation without resolution and comparing the number of conflicts that are flagged and the number of LoSs that have been experienced.
- **Average detection time before conflict**: is the time it takes for the conflict to be detected by the detection method before the conflict is supposed to happen. It is found by subtracting the time the conflict was detected from the time that LoS occured.

I. Hypotheses

Two hypotheses are made, one surrounding the performance based on different traffic densities and the other on different look-ahead times.

*1) **Traffic densities**: It is expected that the intent-based method has a better performance for all traffic densities compared to the state-based method.*

*a) **Conflict Detection**: It is expected that the intent-based detection method has a lower false positive rate and a higher average detection time before conflict for all traffic densities compared to the state-based detection method*

The advantage of the intent-based method comes from the fact that it removes false positive conflicts. This is a major advantage as the main issue of the state-based detection method is the amount of false positive conflicts it detects, making it unsuitable for constrained urban airspace as described by Badea et al. [8]. This is why it is expected that the false positive rate is lower for the intent-based detection method. It is also expected that the intent-based method shows a higher average detection time before conflict as the intent-based method is able to look ahead in its trajectory and detect conflicts reliably as compared to the state-based method where the states have to line up before a conflict is detected. This makes the average detection time less consistent resulting in a lower average detection time before conflict.

*b) **Conflict Resolution**: It is expected that the intent-based resolution method has a lower number of losses of separation for all traffic densities compared to the state-based resolution method*

Due to the decrease in the number of conflicts detected by the intent-based method, there will also be a decrease in losses of separation compared to the state-based method.

*2) **Look-ahead time**: It is expected that the intent-based method has a better performance for all look-ahead times compared to the state-based method.*

*a) **Conflict Detection**: It is expected that the intent-based detection method has a lower false positive rate and a higher* *average detection time before conflict for all look-ahead times compared to the state-based detection method*

The look-ahead time is of significance more so for the state-based model compared to the intent-based model. State-based predictions are less accurate the further ahead the method looks for conflicts since a lot of changes can happen to the state in between the detection and the conflict. The look-ahead time is less relevant for the intent-based method as there are no significant ways that could make the prediction inaccurate between the look-ahead time and the time of the conflict. Therefore, it is expected that the false positive rate is lower for the intent-based method. For the same reason as mentioned in the traffic density hypothesis, the average detection time before conflict is expected to be higher for the intent-based method due to the methods reliability of detecting conflicts early on compared to that of the state-based method where the states must be lined up for the conflicts to be detected.

b) ***Conflict Resolution****: It is expected that the intent-based resolution method has a lower number of losses of separation for all look-ahead times compared to the state-based resolution method*

Once again the intent-based method is expected to result in less losses of separation due to the fact that less conflicts are detected by this method compared to the state-based method.

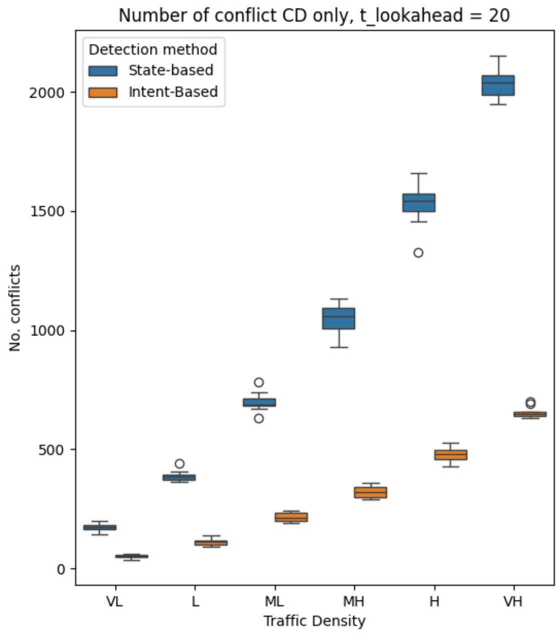


Fig. 6. Number of conflicts per traffic density using a look-ahead time of 20 seconds with resolution disabled

V. Results

In this section the results of the simulation are displayed. First the results dealing with the conflict detection performance are displayed. After that the results relating to the conflict resolution performance are shown. For these two performances the results are only shown for the medium look-ahead time as in the last subsection the results related to the effect of look-ahead time on CD&R performance are shown.

A. Conflict detection performance

1) Number of conflicts detected: First the number of conflicts detected by either detection methods is displayed in Figure 6

Since conflict resolution is not active, both intent-based and state-based are exposed to the same amount of conflicts. Despite of this, the number of conflicts detected by the state-based method is significantly more than the intent-based method as shown in Figure 6. Both methods show an increase in conflicts detected with a higher traffic density as expected. The rate at which the state-based method increases is a lot larger than the intent-based method, showing that the intent-based method scales better with higher traffic densities.

2) False positive rate: The false positive rate is a great indication of the efficiency of the detection method. The false positive rate of both methods for a medium look-ahead time is displayed in Figure 7

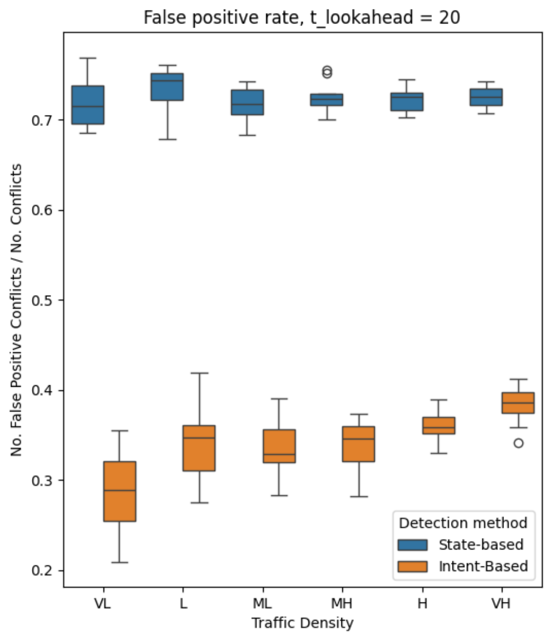


Fig. 7. False positive rate per traffic density using a look-ahead time of 20 seconds

As can be seen in Figure 7 the false positive rate is a lot higher for the state-based method compared to the intent-based method with the state-based rate being more than double for some traffic densities. This shows that the intent-based method is a lot more accurate in detecting conflicts compared to the state-based method. An explanation for this high difference in false positive rate is the fact that the intent-based is able to detect whether aircraft are travelling in parallel streets. The state-based method detects those cases as a conflict due to the aircraft intercepting each other's protected zones, but it is actually not a conflict as both aircraft are travelling in separate streets with most likely a building in between them. This results in a lot more false positive conflicts and hence influences the false positive rate, giving the intent-based method a major advantage over the state-based method.

In addition, it appears that the traffic density does not have an influence on the false positive rates of the state-based method as it is more or less consistent. The traffic density does have an influence on the intent-based method, which shows an increase in false positive rate with a higher traffic density. There is no clear explanation for why this occurs, a possible reason might be that there are certain conditions at which false conflicts are detected at a higher rate and these conditions happen more often in higher traffic densities.

3) Average detection time before conflict: From the initial results it was observed that the average detection time of the state-based method surpassed the look-ahead time in some cases, as is displayed in Figure 8. A possible cause for this is that the conflict was detected early on, but due to state changes, the LoS occured at a time later than the expected 10 seconds. Another reason for the high detection time is that the detected conflict is paired with a LoS in the future due to repeat conflicts. This can skew the average by quite a lot as some of the detection times are more than 100 seconds. This is why all results are recreated, but removing detection times higher than 50 seconds. The corrected average detection time for a medium look-ahead time is is displayed in Figure 9.

It is observed from Figure 9 that the intent-based method consistently detects conflicts earlier than the state-based method. To add on to that, the state-based method still suffers from the phenomena described earlier. Considering this phenomena, the much lower average detection time from the state-based method compared to the look-ahead time can then be explained by a large number of late detections being present to bring the average down. Since the phenomena can not happen for the intent-based method due to the way this method detects its conflicts, it shows that the intent-based method consistently detects conflicts early on to maintain the high average detection time.

The intent-based method and the state-based method both show no real changes due to the traffic densities, except for the very low traffic density, where the intent-based method shows a higher average detection time before conflict. The reason why this occurs is unknown.

B. Conflict resolution performance

1) Number of conflicts detected: The number of conflicts detected differs with resolution active, as the resolution ma-

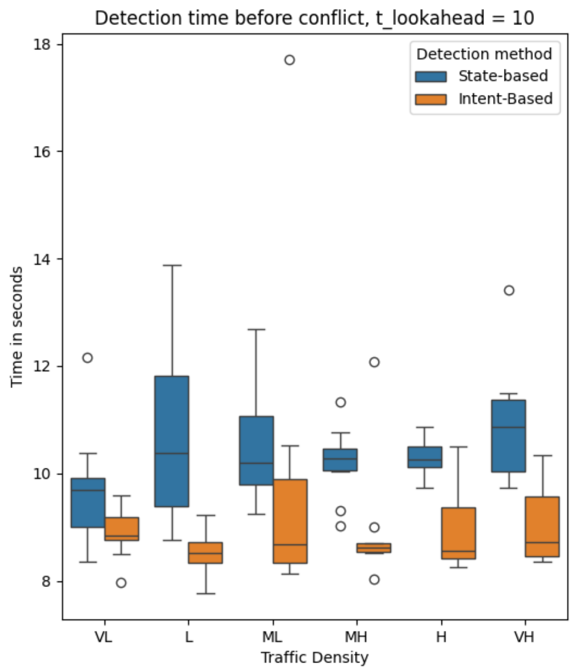


Fig. 8. Average detection time before conflict with a look-ahead time of 10 seconds

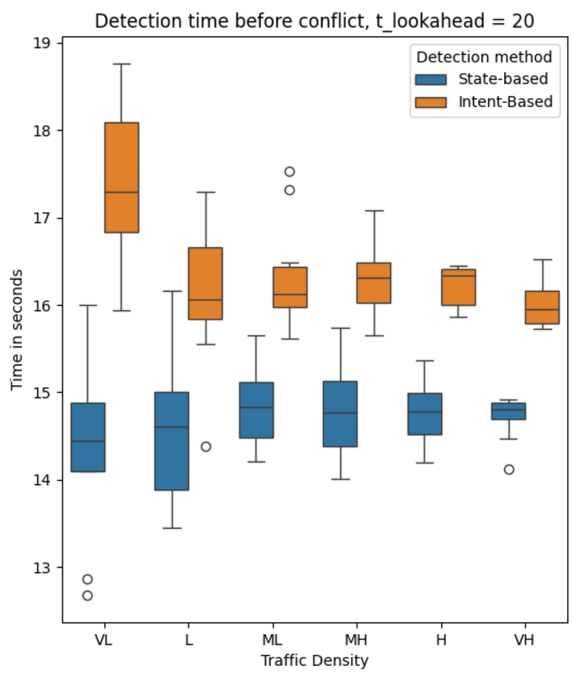


Fig. 9. Corrected average detection time before conflict with a look-ahead time of 20 seconds

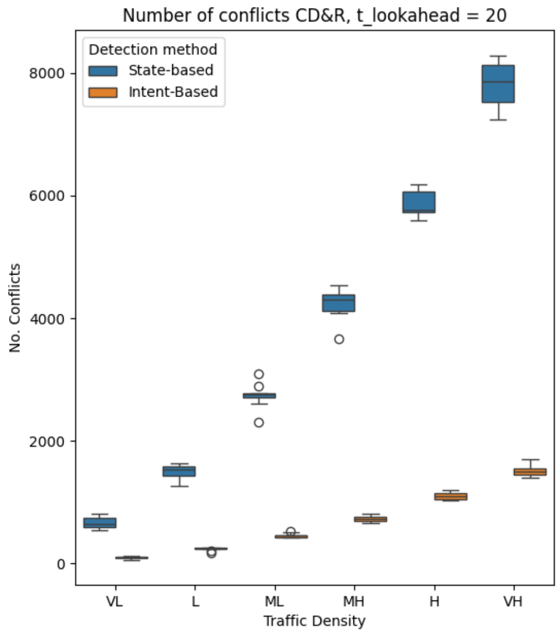


Fig. 10. Number of conflicts per traffic density using a look-ahead time of 20 seconds with resolution active

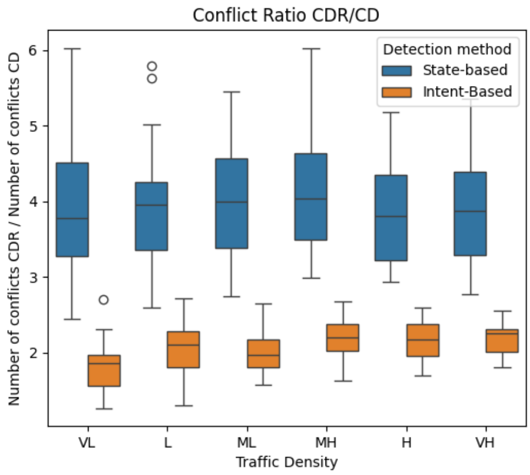


Fig. 11. Conflict ratio between CD with and without resolution active

noeuvres themselves will create secondary conflicts. The impact of these secondary conflicts can be observed in Figure 10

As can be seen from Figure 10 the amount of conflicts detected has increased substantially for both detection methods. This is the result of the secondary conflicts that are created due to the resolution manoeuvres. Although Figure 6 and Figure 10 look very similar, the number of secondary conflicts created by the state-based method is more compared to the intent-based method. This is observed in Figure 11 and shows that the number of conflicts with resolution active is approximately four times higher than without resolution active for the state-based method. The intent-based method shows an average ration between 1.9 and 2.5 meaning it creates fewer secondary conflicts.

2) Number of losses of separation: The number of LoS gives a solid idea on the performance of the conflict resolution. The result of the simulations can be found in Figure 12.

Although there are more conflicts detected by the state-based method, more LoSs occured for the intent-based method compared to the state-based method. The difference is initially small for lower traffic densities but becomes quite large for higher traffic densities. The likely cause of the difference between the two methods is the bypass that was implemented for the rule-based algorithm to get rid of deadlocks. This also explains the bigger increase in LoSs for higher traffic densities as the likelihood of deadlocks occuring increases with traffic densities.

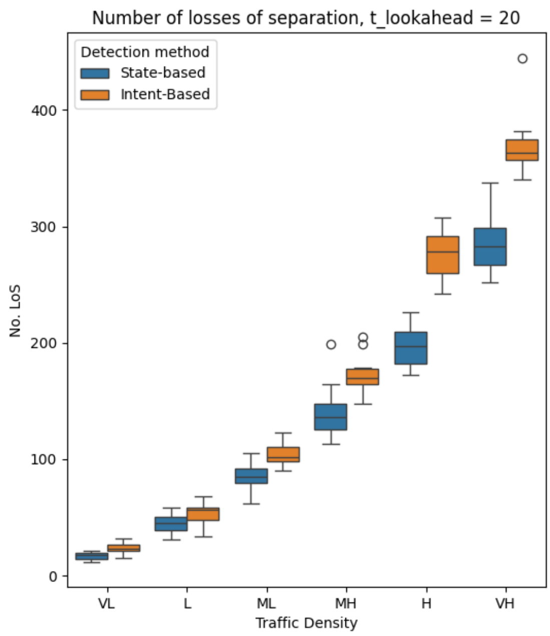


Fig. 12. Number of losses of separation per traffic density using a look-ahead time of 20 seconds with resolution active

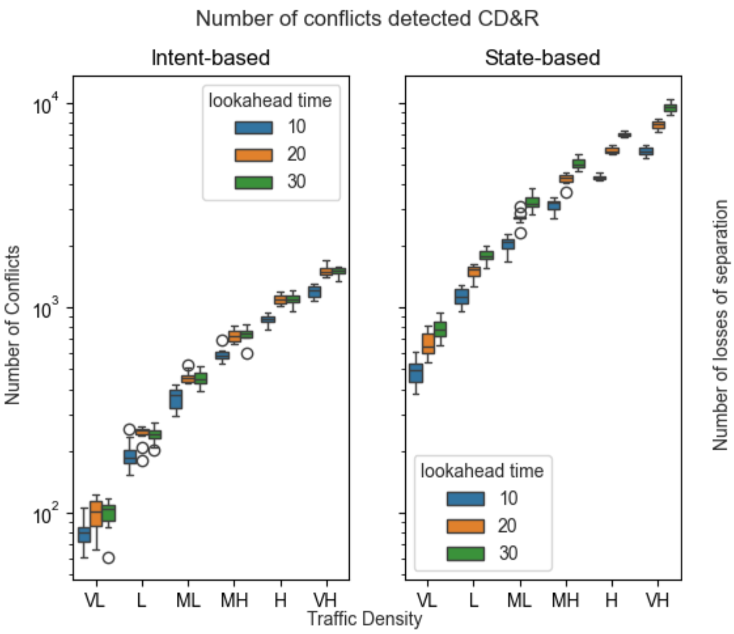


Fig. 13. Number of conflicts per look-ahead time with resolution active on a logarithmic scale

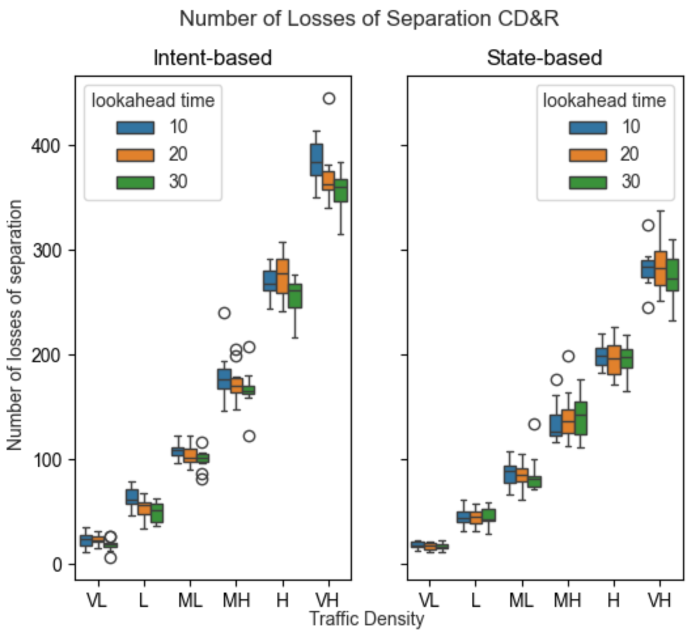


Fig. 15. Number of losses of separation per look-ahead time

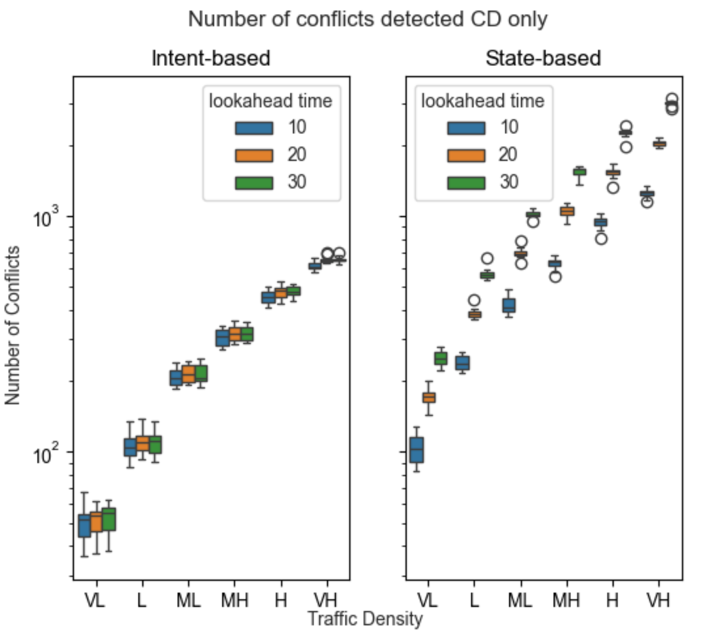


Fig. 14. Number of conflicts per look-ahead time with resolution disabled on a logarithmic scale

C. *Effect of look-ahead time on CD&R performance*

1) Number of conflicts: The number of conflicts detected with and without the resolution active are plotted in Figure 13 and Figure 14

A very large difference can be observed for the state-based method where the amount of conflicts detected increased rapidly with a larger look-ahead time while the intent-based method stayed around the same level as depicted in Figure 13 and Figure 14. This can be explained by the state-based method extrapolating its current state for the whole look-ahead time, not taking into account that these states can change in the time it takes for the conflict to occur. This results in a higher amount of conflicts detected. From this it can be concluded that a higher look-ahead time favours the intent-based method as the state-based method performs very poorly in that region. Moreover, it is shown for the intent-based method that the amount of conflicts detected by the medium and large look-ahead time is a bit bigger compared to the small look-ahead time, especially in Figure 13. This is probably due to the intent-based method's prediction being more accurate the closer it is to the conflict. This is why a 10 second look-ahead time may show less conflicts as it may record less false positive conflicts. A possible reason why the difference between medium and large look-ahead time might be small is that the accuracy of the intent-based method drops sharply somewhere between a look-ahead time of 10 and 20 seconds resulting in the 20 and 30 second look-ahead time detecting the same number of false positive conflicts.

2) Number of losses of separation: The number of losses of separation for each look-ahead time are displayed in Figure 15.

As evident from Figure 15, there are no large differences in the amount of LoS experienced between the different look-ahead times. There is a slight pattern that can be observed for the intent-based method where the large look-ahead time shows the smallest number of LoSs followed by the medium look-ahead time and the small look-ahead time having the

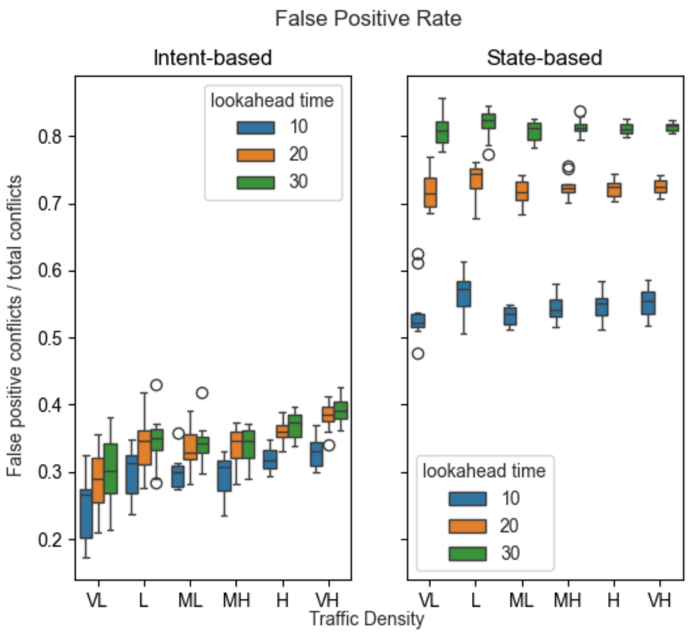


Fig. 16. False positive rate per look-ahead time

most LoSs. A possible explanation for this is that there is less time to solve the conflict with a shorter look-ahead time and hence results in more LoSs.

3) False positive rate: It is observed in Figure 16 that the false positive rate increases by a lot for the state-based method when using a higher look-ahead time, once again confirming that indeed the accuracy of the state-based method worsens with a higher look-ahead time. Meanwhile, the intent-based method shows similar amounts of false positive conflicts for the medium and large look-ahead time and a lot less for the small look-ahead time. This once again shows that the shorter look-ahead time provides more accurate predictions and results in less false positive conflicts. Moreover, the small spread of the false positive rate shows that the influence of look-ahead time is a lot smaller for the intent-based method compared to the state-based method.

VI. Discussion

A. Discussion on conflict detection performance

The results show that the detection performance of the intent-based method surpasses that of the state-based method for all traffic densities. The amount of conflicts detected are far lower for the intent-based method. More importantly, the amount of false positive conflicts detected are also much lower thanks to the intent-based method being able to distinguish whether the conflicting aircraft's paths intersect. This gives the intent-based method a definitive edge over the state-based method and is the main that its performance is so much better in constrained very low-level urban airspace. Although the traffic density has a small impact on the performance of the intent-based method, in terms of false positive rate, it is not very significant making the use of the intent-based method acceptable for all traffic densities.

B. Discussion on conflict resolution performance

Unlike conflict detection, the results rule clearly in favour of the state-based method for conflict resolution. Although the intent-based resolution increased the amount of conflicts by roughly the same factor as the state-based method, the number of LoSs are higher, especially for higher traffic densities. This performance is especially concerning, considering that the number of conflicts are sometimes more than 4 times higher. Another aspect that should be taken into account when explaining the difference in LoSs is the fact that the number of false positive conflicts that are detected are a lot higher for the state-based method and hence a lot more conflicts are solved. Although an effect of this is less efficient routing of aircraft due to resolution manoeuvres, a benefit of resolving the false conflicts is that it enhances safety by creating more distance between aircraft.

C. Discussion on the effect of look-ahead time on CD&R performance

From the results of the look-ahead time it can be observed that an increasing look-ahead time results in a very big drop in performance for the state-based method. The number of conflicts detected increases by a lot when using a larger look-ahead time. Not only does the number of conflicts increase but the false positive rate skyrockets when higher look-ahead times are used. The intent-based method on the other hand does suffer from a drop in efficiency for higher look-ahead times but is not very significant as is demonstrated by the slight increase of false positive rate for a large look-ahead time. A larger look-ahead time does help the performance of the intent-based resolution as a slight drop in LoS is observed. Looking at these two factors, the look-ahead time does not significantly impact the performance of the intent-based CD&R, whereas the state-based CD&R is negatively impacted by an increase in look-ahead time. Being able to use a larger look-ahead time without a significant penalty in performance allows the intent-based method to detect conflicts earlier and resolving them earlier, enhancing safety. Hence, it is advised to use a look-ahead time of 30 seconds for the intent-based method.

D. Hypotheses

From the results it can be determined that hypothesis 1 is partially correct as the intent-based method indeed has a better performance

1) Traffic densities: From the results it can be concluded that Hypothesis 1a is accepted as the false positive rate of the intent-based method is lower for all traffic densities. Furthermore, the average detection time before conflict is also larger for all traffic densities.

Hypothesis 1b on the other hand is discarded, as the number of losses of separation is higher for the intent-based method, eventhough the number of conflicts detected is considerably lower.

This means that hypothesis 1 is partially accepted as the intent-based detection showed a much better performance compared to the state-based method. However, the opposite is true for the resolution method.

2) Look-ahead time: The results show that hypothesis 2a is partially accepted as the false positive rate for all look-ahead times is much lower for the intent-based method compared to the state-based method. The average detection time before conflict is a bit more complicated due to the look-ahead time of 10 seconds being in favour of the state-based method and the 20 and 30 second look-ahead time being in favour of the intent-based method. However, the result of the detection time for a look-ahead time of 10 seconds is skewed due to the detection time sometimes being larger than 10 seconds which is not supposed to happen.

Hypothesis 2b is rejected as for all look-ahead times, the number of losses of separation is lower for the state-based method compared to the intent-based method.

With this, hypothesis 2 can be partially accepted, as hypothesis 2a is partially accepted and hypothesis 2b is rejected.

E. Recommendations for future work

The results of this paper show that indeed the intent-based CD&R method is viable, but still has some areas for improvements. The most obvious improvement that can be made is improving the resolution method. The main problem with the resolution is the creation of deadlocks, especially in higher traffic densities. These deadlocks form when multi-aircraft conflicts occur. Due to the rule-based algorithm, multiple if not all aircraft are put to a stop, and each aircraft is then waiting on the other before it continues flying. A major improvement can be made to the rule-based algorithm to reduce the number of LoSs. After the improvements are made and the resolution performs optimally, a comparison to a state-based method can then once again be made. If the state-based method still comes out on top regarding LoSs, an analysis can be made regarding safety and route efficiency. Using this analysis a trade-off can then be made to decide which one is superior.

Another point for future work is researching the effect of disturbances on the performance of the intent-based method. The current research assumes there are no outside disturbances meaning that the predictions made by the intent-based method are fairly accurate. With the introduction of disturbances, the intent-based predictions become less accurate and may result in more false positive conflicts detected or more false negative conflicts. These inaccuracies can potentially be negated by incorporating the disturbance into the trajectory calculation. This may result in inaccuracies for a longer look-ahead time but it is predicted that position estimations will still be accurate for a lower look-ahead time.

Moreover, the intent-based method has been tested in a mostly orthogonal environment which eases the predictability as almost every turn that is made is at a 90 degree angle, simplifying the calculations and the variety of speeds the drone travels at. The implementation in a non-orthogonal environment may pose some challenges due to a higher variation of navigation manoeuvres which makes accurately predicting future positions more complex.

VII. Conclusion

This paper demonstrated the implementation and the evaluation of an intent-based detection and resolution method in orthogonal constrained airspace. The results show that the intent-based detection method performs better than the state-based detection method in terms of having a lower false positive rate for very low to very high traffic densities. In addition, the false positive rate of the intent-based method is lower for all look-ahead times. The average detection time before conflict is also evaluated for both methods and shows that the intent-based method is superior for all traffic densities using a look-ahead time of 20 and 30 seconds. The state-based method performed better for all traffic densities using a look-ahead time of 10 seconds. Moreover, the state-based resolution method showed better performances for all look-ahead times and traffic densities compared to the intent-based method.

The paper shows that there is room for improvements for the intent-based resolution as it underperformed compared to the state-based method. The issue for this poor performance is the creation of deadlocks which originates from the poor handling by the rule-based algorithm for multi-aircraft conflicts. Another area that should be explored further is the effect of disturbances to the performance of the intent-based method. Introducing disturbances would reduce the accuracy of the predictions made by the detection method. An interesting point to research is whether the drop in performance makes the intent-based method worse compared to other detection methods. Lastly, another research point could be researching the performance of the intent-based method in a non-orthogonal street network as this would introduce more complicated navigation manoeuvres increasing the difficulty to make accurate position predictions.

All in all, the results show that an intent-based method might be viable for the use in constrained airspace after making some improvements. The false positive rate is lower for the intent-based method and combining it with a higher look-ahead time would make it possible to detect conflicts reliably, and earlier enhancing the safety of drone operations.

References

[1] SESAR JU. *European Drones Outlook Study: Unlocking the value for Europe*. 2018.

[2] David Gallacher. "Drones to manage the urban environment. Risks, rewards, alternatives." In: 4 (Feb. 2016), pp. 115–124. DOI: 10.1139/juvs-2015-0040.

[3] Emanuele Luigi de Angelis et al. "Toward Smart Air Mobility: Control System Design and Experimental Validation for an Unmanned Light Helicopter". In: *Drones* 7.5 (2023). ISSN: 2504-446X. DOI: 10.3390/drones7050288. URL: https://www.mdpi.com/2504-446X/7/5/288.

[4] Ang Li, Mark Hansen, and Bo Zou. "Traffic management and resource allocation for UAV-based parcel delivery in low-altitude urban space". In: *Transportation Research Part C: Emerging Technologies* 143 (2022), p. 103808. ISSN: 0968-090X. DOI: https://doi.org/10.1016/j.trc.2022.103808. URL: https://www.sciencedirect.com/science/article/pii/S0968090X22002339.

[5] Yazdi I. Jenie et al. "Taxonomy of Conflict Detection and Resolution Approaches for Unmanned Aerial Vehicle in an Integrated Airspace". In: *IEEE Transactions on Intelligent Transportation Systems* 18.3 (2017), pp. 558–567. DOI: 10.1109/TITS.2016.2580219.

[6] Marta Ribeiro, Joost Ellerbroek, and Jacco Hoekstra. "Review of Conflict Resolution Methods for Manned and Unmanned Aviation". In: 7 (June 2020). DOI: 10.3390/aerospace7060079.

[7] Jacco Hoekstra and Joost Ellerbroek. "Aerial Robotics: State-based Conflict Detection and Resolution (Detect and Avoid) in High Traffic Densities and Complexities". In: *Current Robotics Reports* 2 (Sept. 2021). DOI: 10.1007/s43154-021-00061-6.

[8] C. Badea et al. "Limitations of Conflict Prevention and Resolution in Constrained Very Low-Level Urban Airspace". English. In: *11th SESAR Innovation Days*. 11th SESAR Innovation Days, SIDs 2021 ; Conference date: 07-12-2021 Through 09-12-2021. 2021.

[9] Calin Andrei Badea et al. "Unifying Tactical Conflict Prevention, Detection, and Resolution Methods in Non-Orthogonal Constrained Urban Airspace". In: *Aerospace* 10 (Apr. 2023). DOI: 10.3390/aerospace10050423.

[10] Jimmy Krozel and Dominick Andrisani. "Intent Inference with Path Prediction". In: *Journal of Guidance Control and Dynamics - J GUID CONTROL DYNAM* 29 (Mar. 2006), pp. 225–236. DOI: 10.2514/1.14348.

[11] Marta Ribeiro, Joost Ellerbroek, and Jacco Hoekstra. "The Effect of Intent on Conflict Detection and Resolution at High Traffic Densities". In: Sept. 2020.

[12] J.K. Kuchar and L.C. Yang. "A review of conflict detection and resolution modeling methods". In: *IEEE Transactions on Intelligent Transportation Systems* 1.4 (2000), pp. 179–189. DOI: 10.1109/6979.898217.

[13] SKYbrary. *Separation standards*. URL: https://www.skybrary.aero/articles/separation-standards.

[14] *COMMISSION REGULATION No 2150/2005: laying down common rules for the flexible use of airspace*$_2$005. 2005.

[15] Ling-zhi Sang et al. "CPA Calculation Method based on AIS Position Prediction". In: *Journal of Navigation* -1 (Apr. 2016), pp. 1–18. DOI: 10.1017/S0373463316000229.

[16] Jacco Hoekstra and Joost Ellerbroek. "BlueSky ATC Simulator Project: an Open Data and Open Source Approach". In: June 2016.

[17] Marta Ribeiro, Joost Ellerbroek, and Jacco Hoekstra. "Improvement of Conflict Detection and Resolution at High Densities Through Reinforcement Learning". In: July 2020.

[18] Malik Doole et al. "Constrained Urban Airspace Design for Large-Scale Drone-Based Delivery Traffic". In: *Aerospace* 8.2 (Feb. 2021), p. 38. ISSN: 2226-4310. DOI: 10.3390/aerospace8020038. URL: http://dx.doi.org/10.3390/aerospace8020038.

[19] Geoff Boeing. "OSMNX: New Methods for Acquiring, Constructing, Analyzing, and Visualizing Complex Street Networks". In: *Computers Environment and Urban Systems* 65 (July 2017), pp. 126–139. DOI: 10.1016/j.compenvurbsys.2017.05.004.

[20] Pratyush Tripathy et al. "An open-source tool to extract natural continuity and hierarchy of urban street networks". In: *Environment and Planning B: Urban Analytics and City Science* 48.8 (2021), pp. 2188–2205. DOI: 10.1177/2399808320967680. eprint: https://doi.org/10.1177/2399808320967680. URL: https://doi.org/10.1177/2399808320967680.

[21] Aric Hagberg, Pieter Swart, and Daniel S Chult. *Exploring network structure, dynamics, and function using NetworkX*. Tech. rep. Los Alamos National Lab.(LANL), Los Alamos, NM (United States), 2008.

[22] C.A. Badea, Joost Ellerbroek, and Jacco Hoekstra. "The Benefits of Using Intent Information in Tactical Conflict Resolution for U-space/UTM Operations". In: (unpublished).

[23] Malik Doole, Joost Ellerbroek, and Jacco Hoekstra. "Estimation of traffic density from drone-based delivery in very low level urban airspace". In: *Journal of Air Transport Management* 88 (2020), p. 101862. ISSN: 0969-6997. DOI: https://doi.org/10.1016/j.jairtraman.2020.101862. URL: https://www.sciencedirect.com/science/article/pii/S0969699719304004.

II

Appendices to Scientific Paper

A

Conflict Plots

In this appendix the number of conflict plots are shown. Firstly, the plots with the resolution disabled are shown and afterwards the plots with resolution active are shown.

A.1. Resolution Disabled

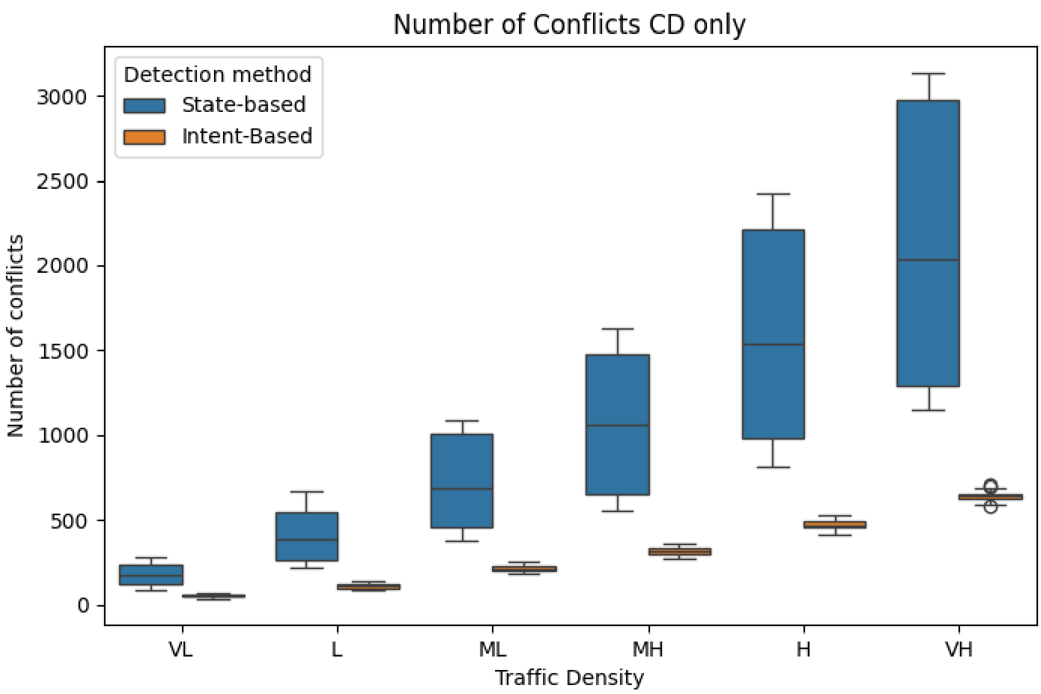


Figure A.1: Number of conflicts per traffic density with resolution disabled

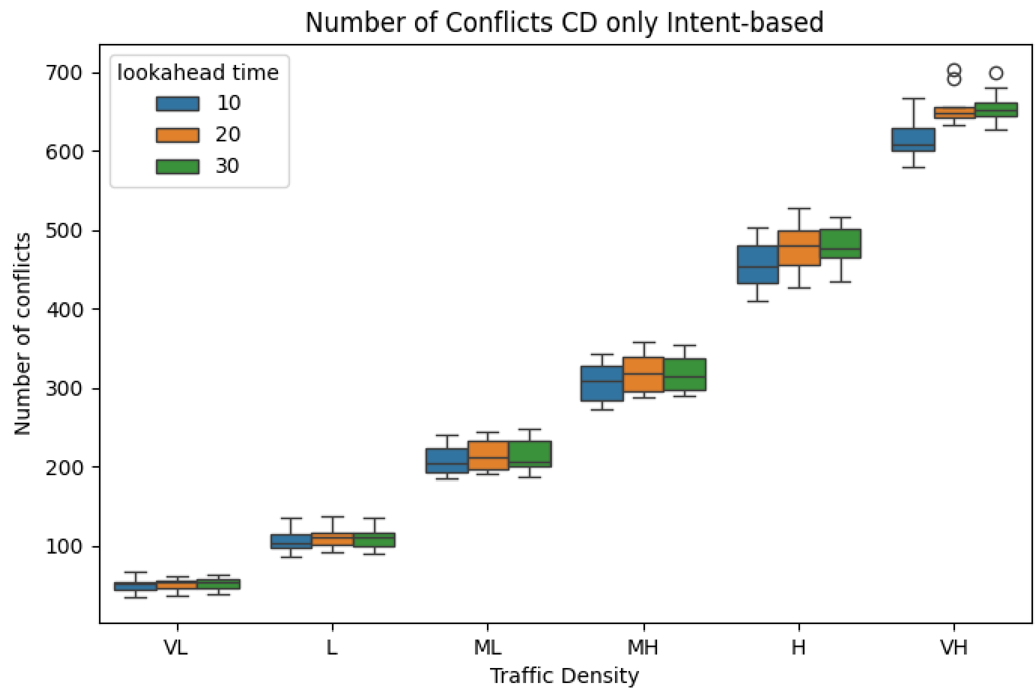


Figure A.2: Number of conflicts per traffic density with resolution disabled, Intent-based method

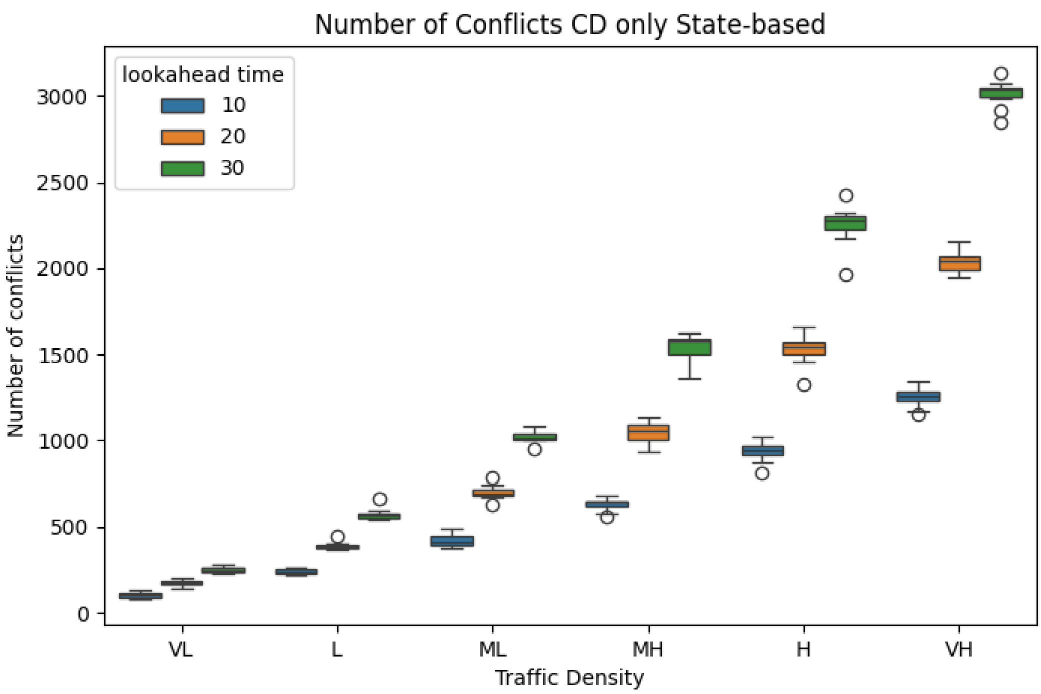


Figure A.3: Number of conflicts per traffic density with resolution disabled, State-based method

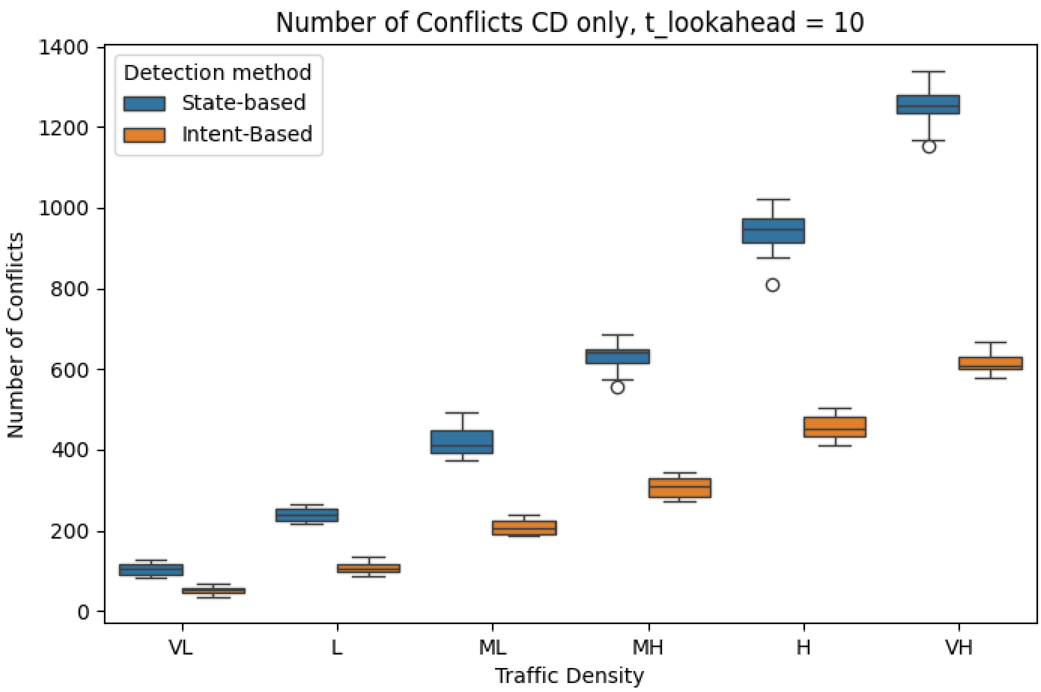


Figure A.4: Number of conflicts per traffic density with resolution disabled and a look-ahead time of 10 seconds

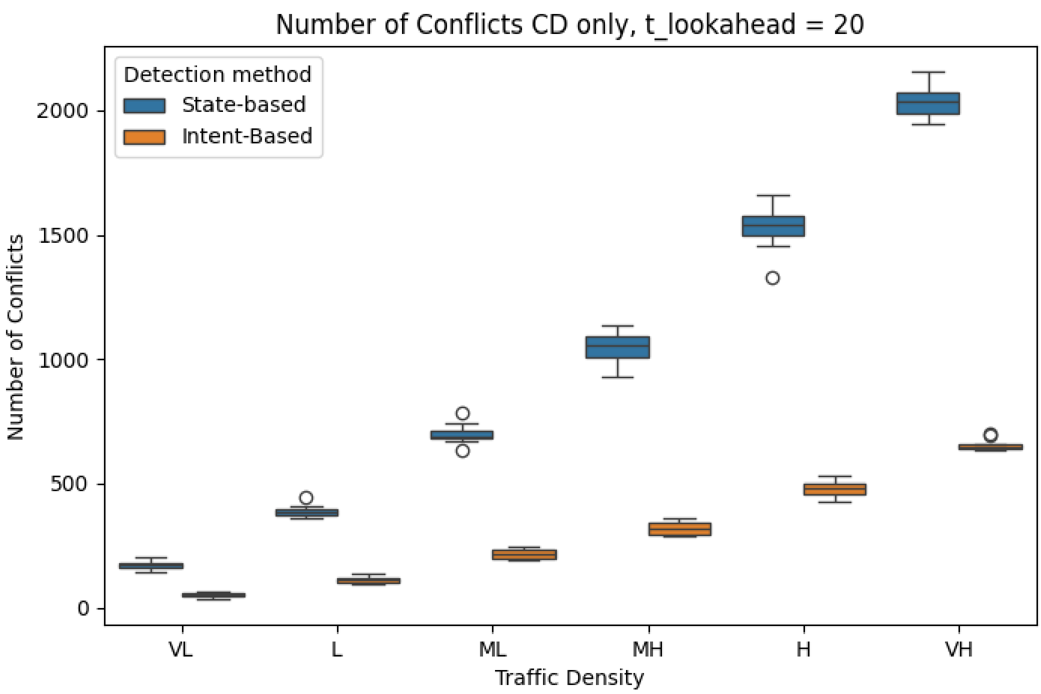


Figure A.5: Number of conflicts per traffic density with resolution disabled and a look-ahead time of 20 seconds

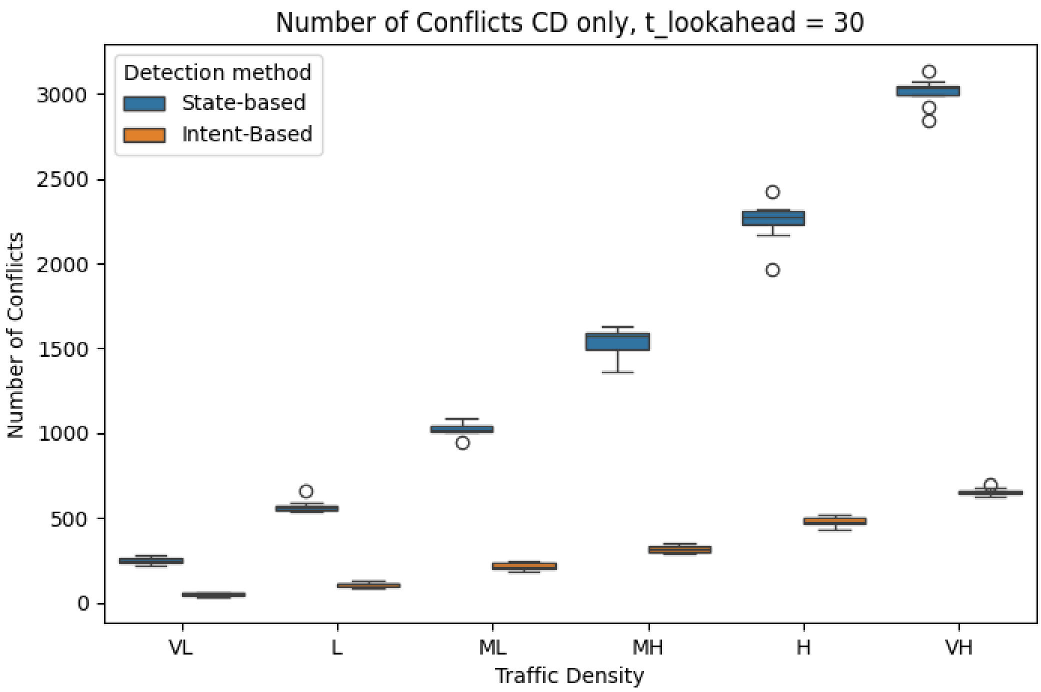


Figure A.6: Number of conflicts per traffic density with resolution disabled and a look-ahead time of 30 seconds

A.2. Resolution Active

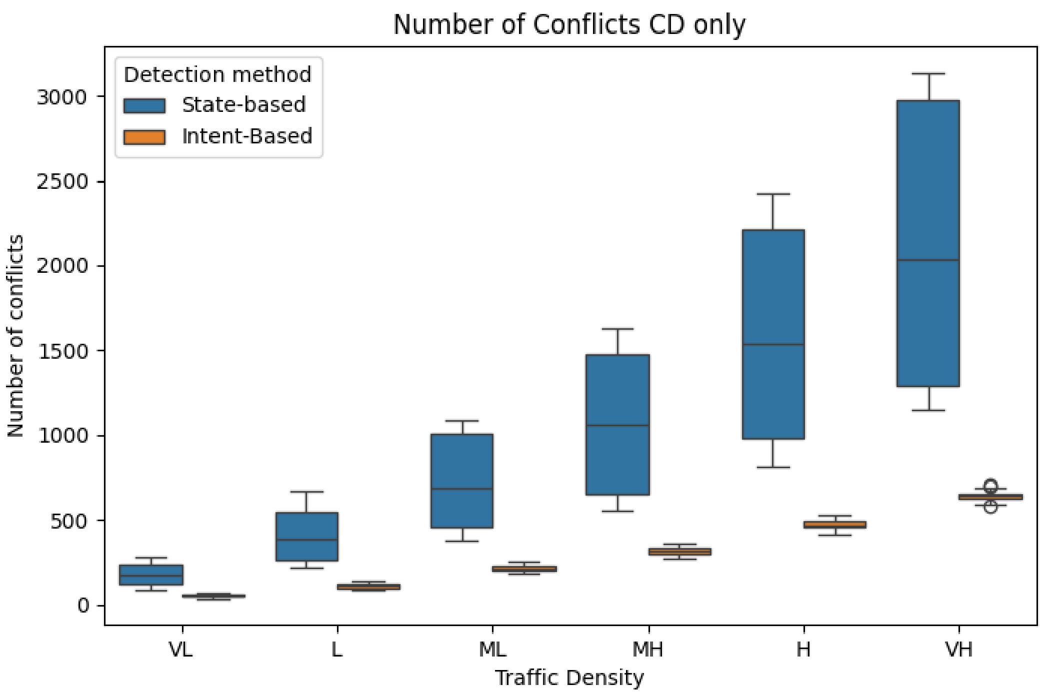


Figure A.7: Number of conflicts per traffic density with resolution active

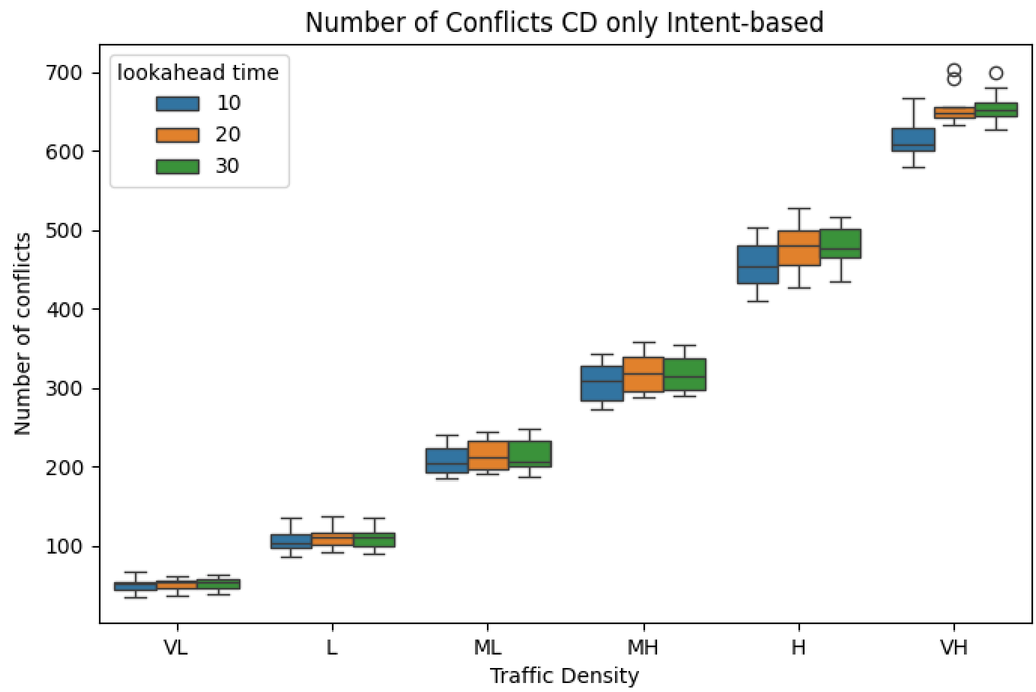


Figure A.8: Number of conflicts per traffic density with resolution active, Intent-based method

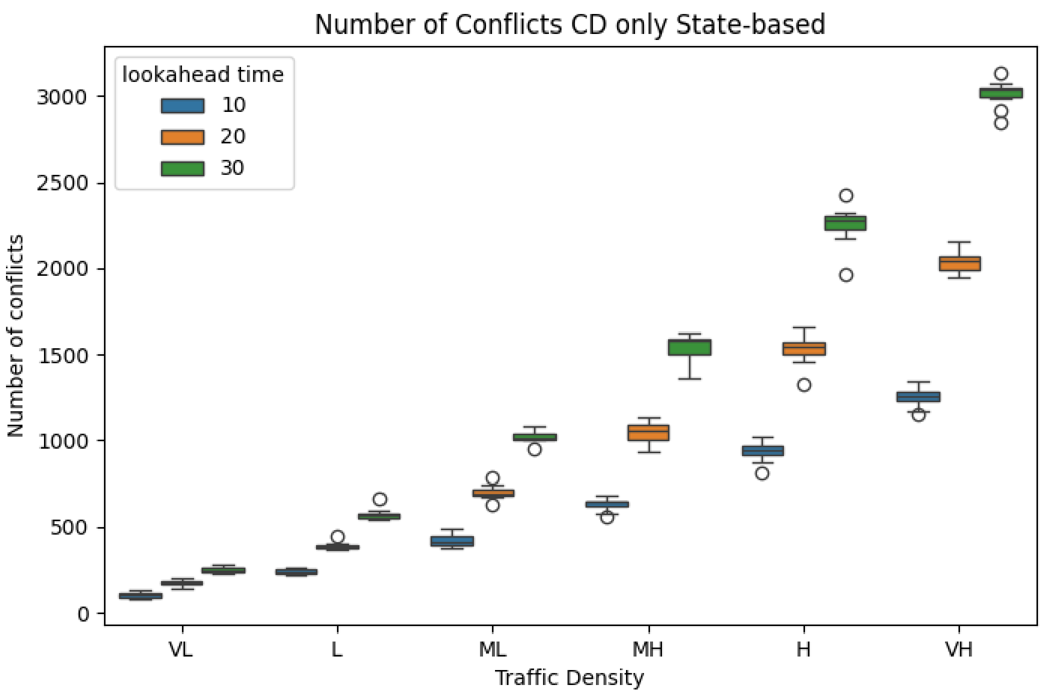


Figure A.9: Number of conflicts per traffic density with resolution active, State-based method

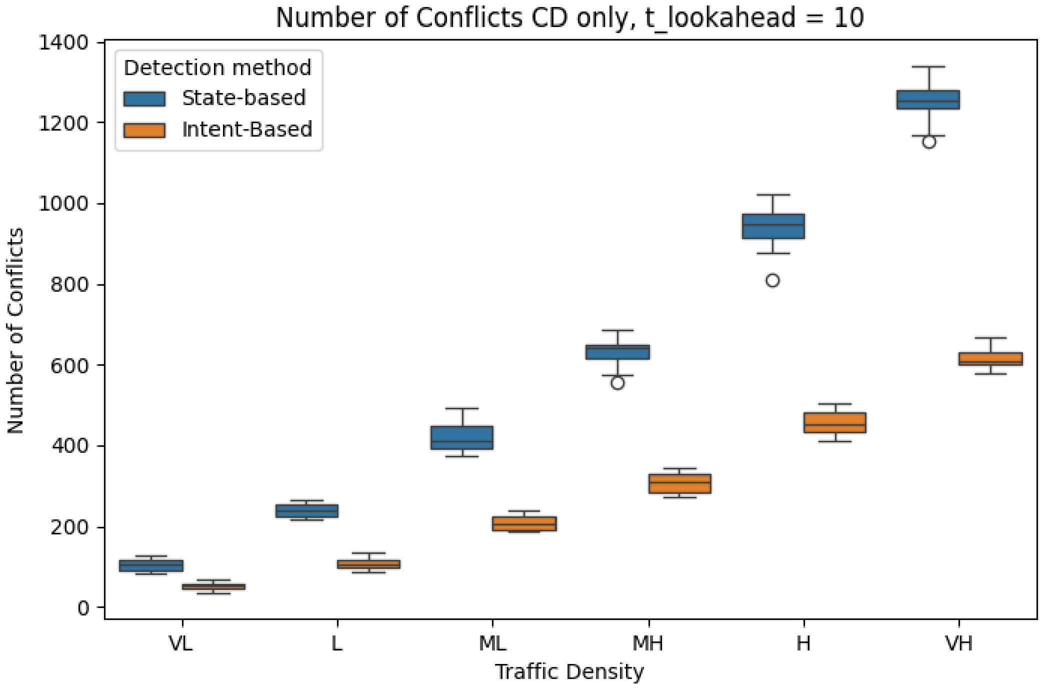


Figure A.10: Number of conflicts per traffic density with resolution active and a look-ahead time of 10 seconds

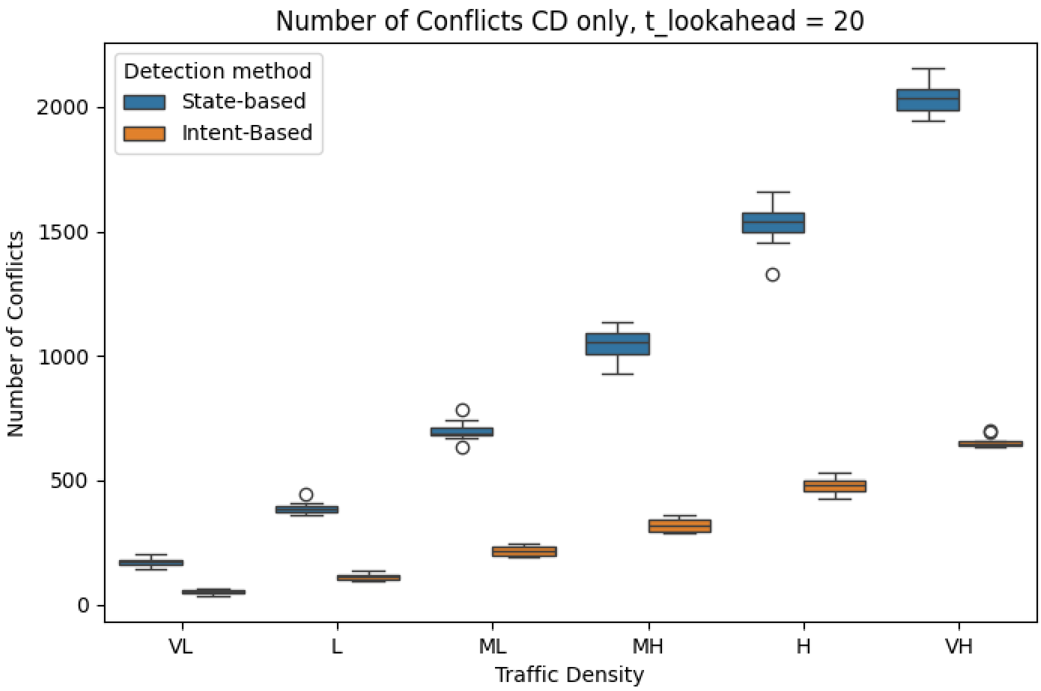


Figure A.11: Number of conflicts per traffic density with resolution active and a look-ahead time of 20 seconds

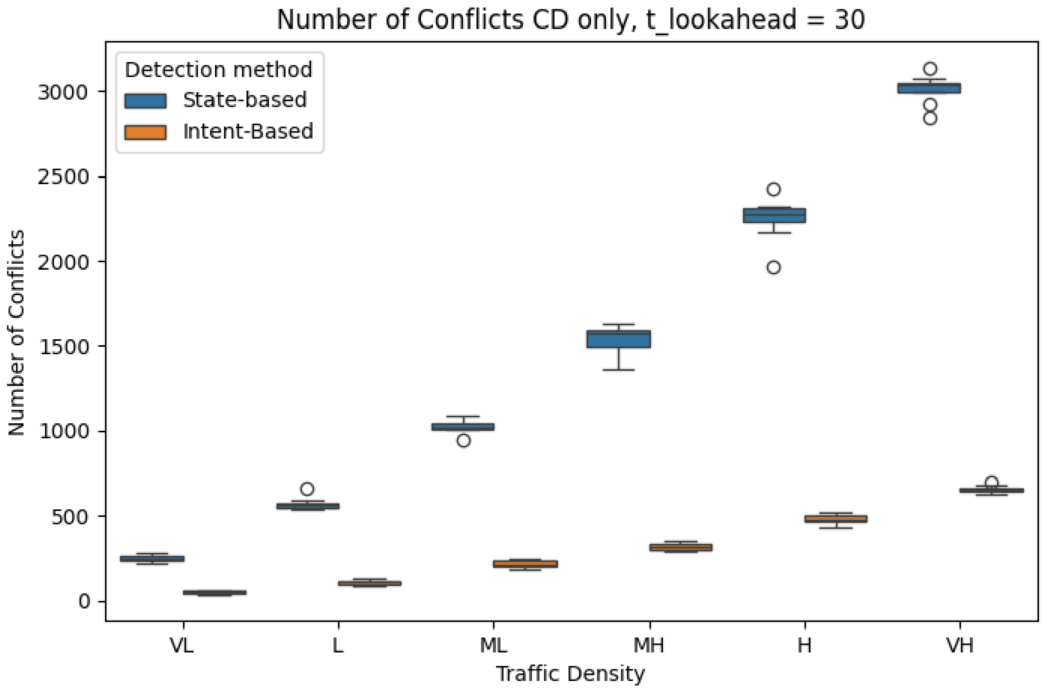


Figure A.12: Number of conflicts per traffic density with resolution active and a look-ahead time of 30 seconds

B

Losses of separation Plots

The losses of separation plots are displayed in this appendix.

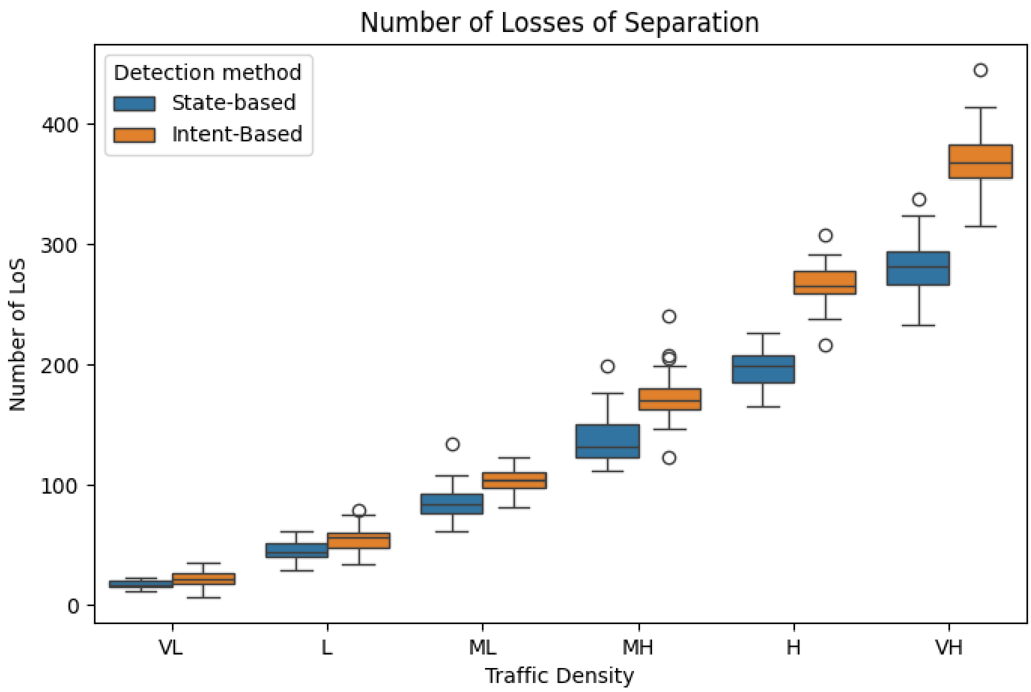


Figure B.1: Number of losses of separation per traffic density

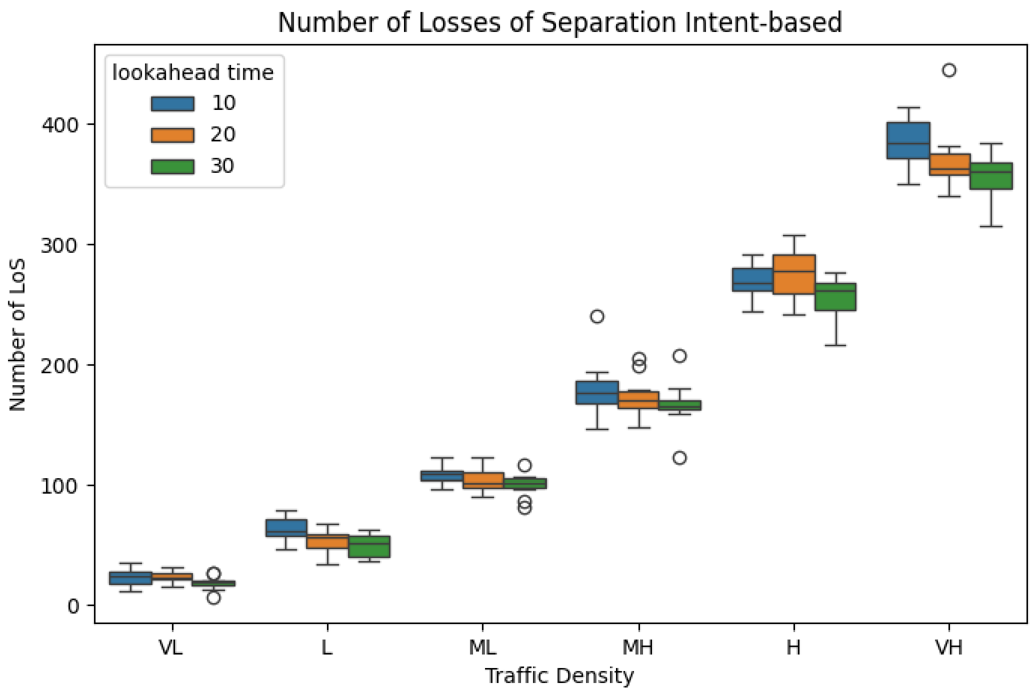


Figure B.2: Number of losses of separation per traffic density, Intent-based method

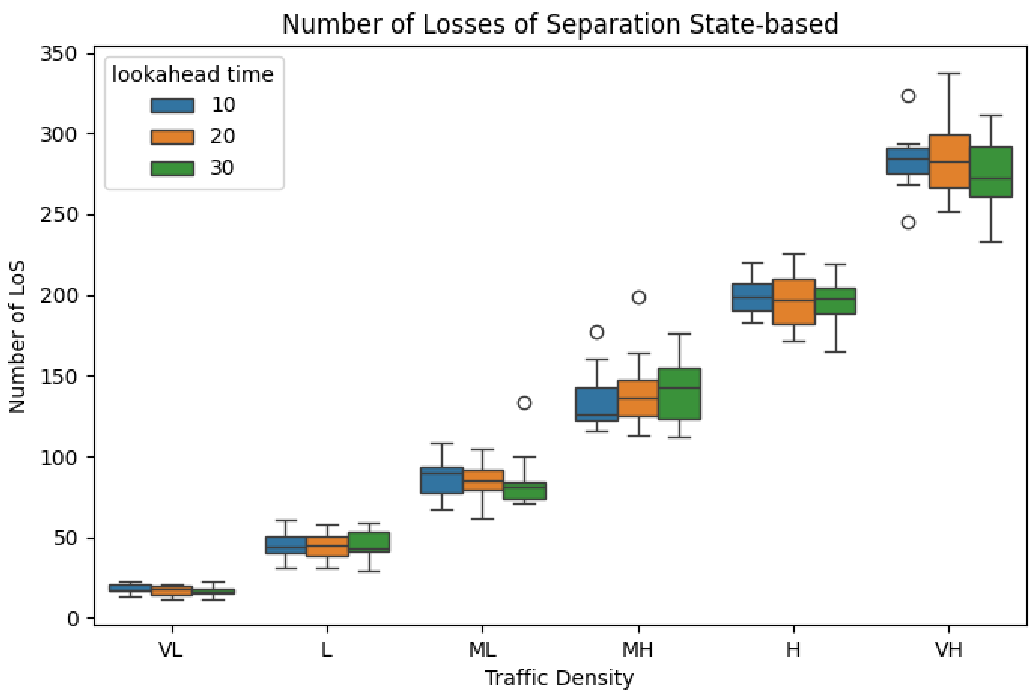


Figure B.3: Number of losses of separation per traffic density, State-based method

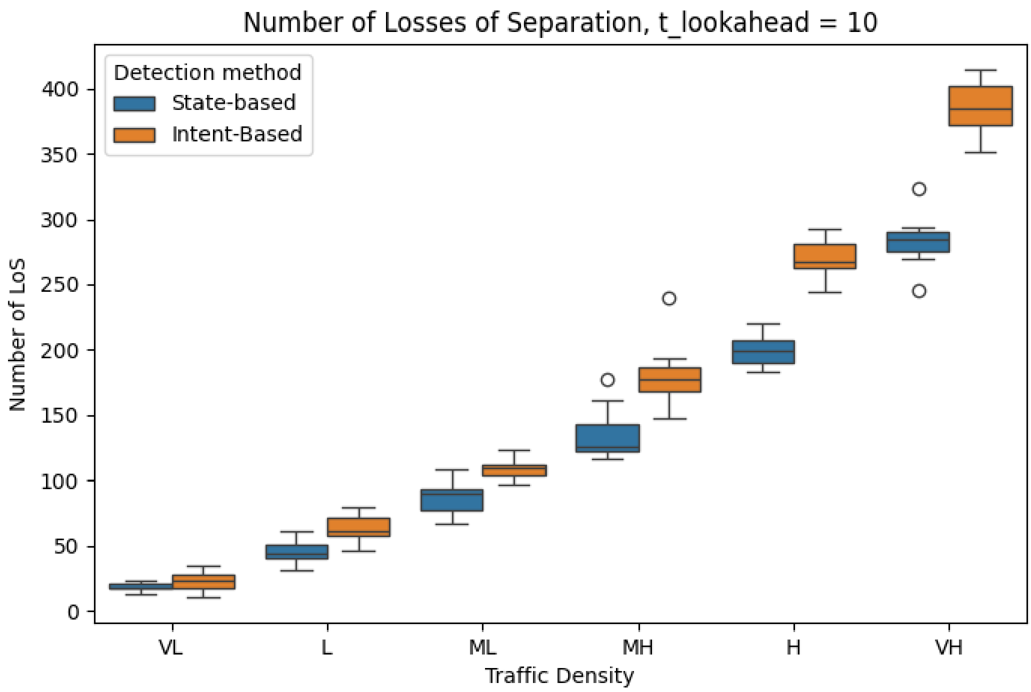


Figure B.4: Number of losses of separation per traffic density with a look-ahead time of 10 seconds

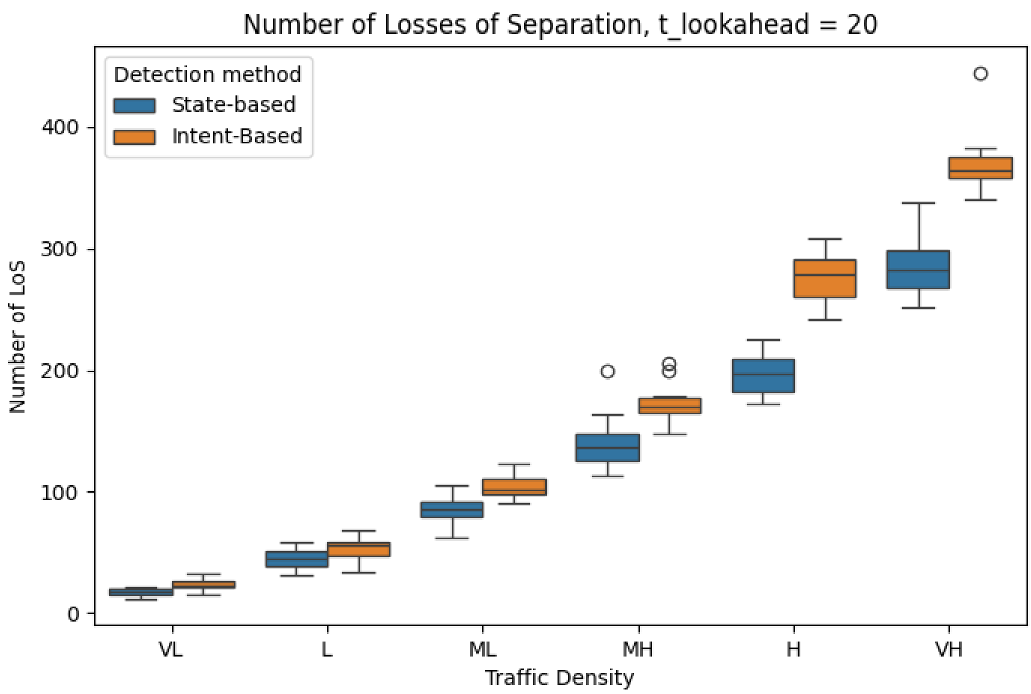


Figure B.5: Number of losses of separation per traffic density with a look-ahead time of 20 seconds

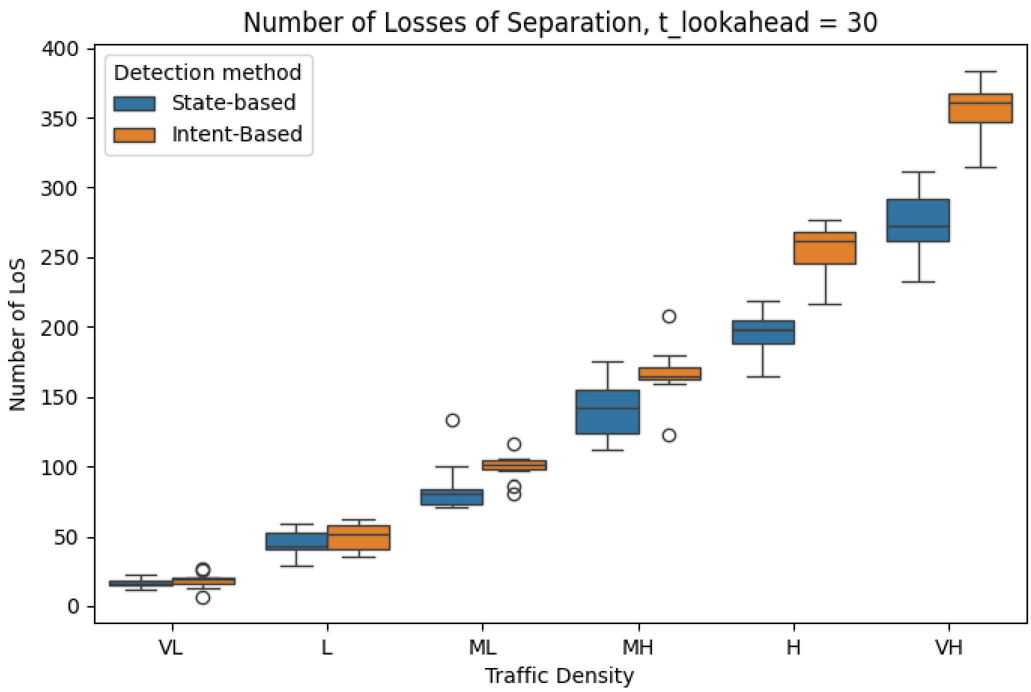


Figure B.6: Number of losses of separation per traffic density with a look-ahead time of 30 seconds

C

False Positive Rate Plots

The false positive rate plots are shown in this appendix

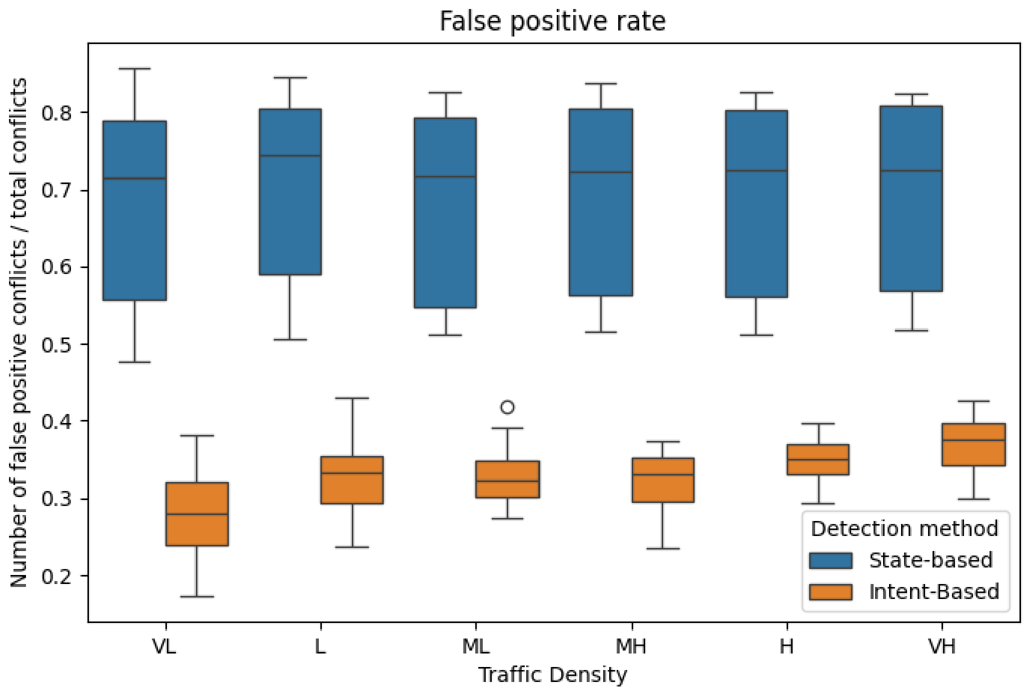


Figure C.1: False positive rate per traffic density

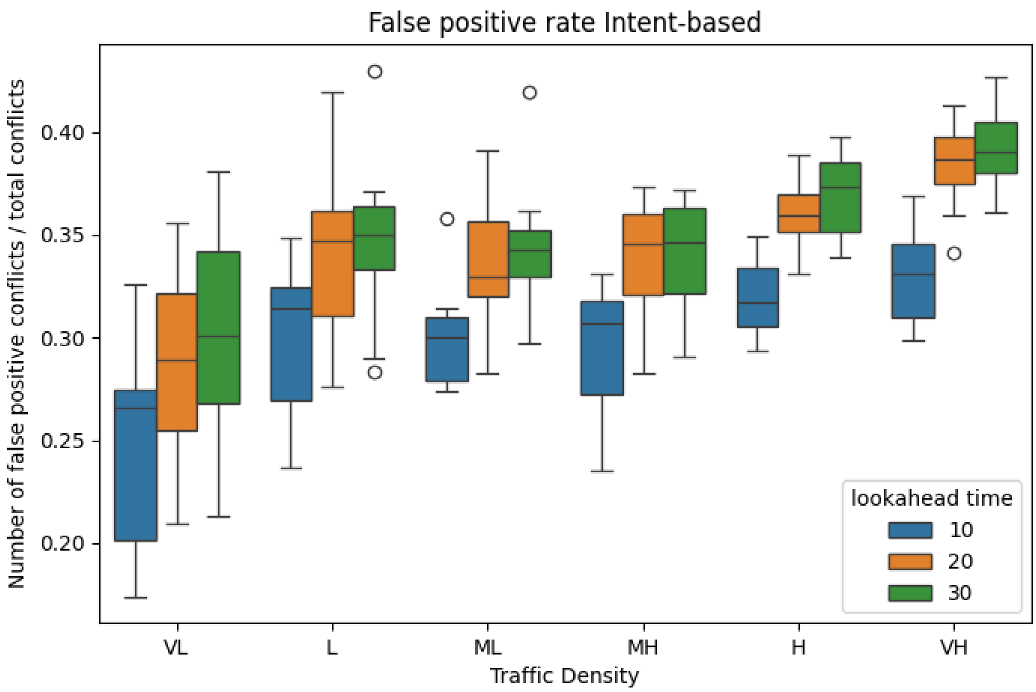


Figure C.2: False positive rate per traffic density, Intent-based method

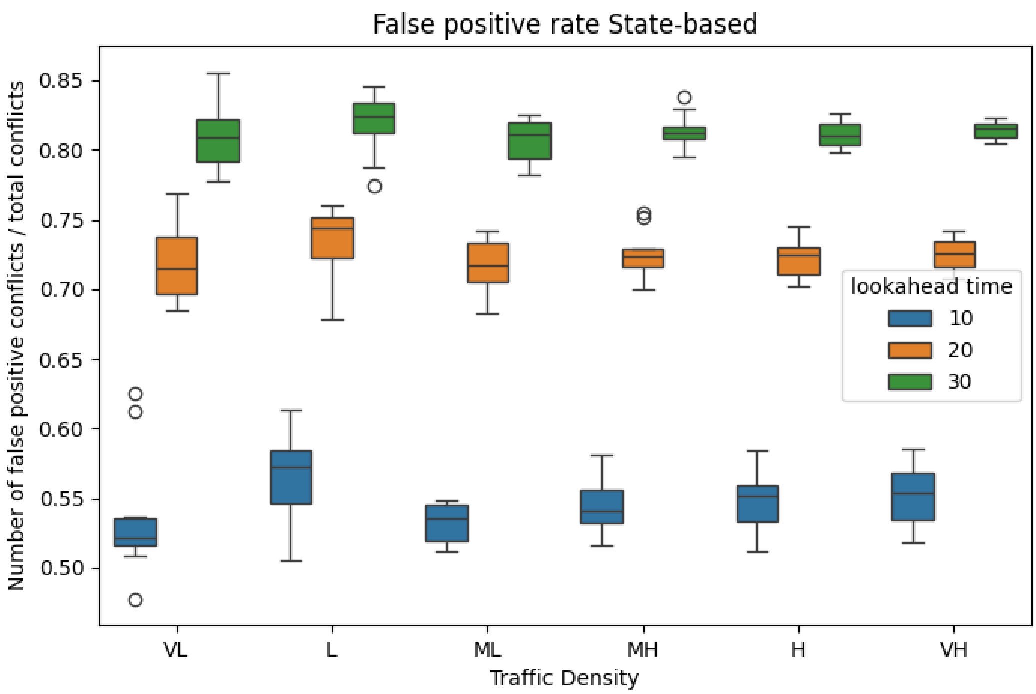


Figure C.3: False positive rate per traffic density, State-based method

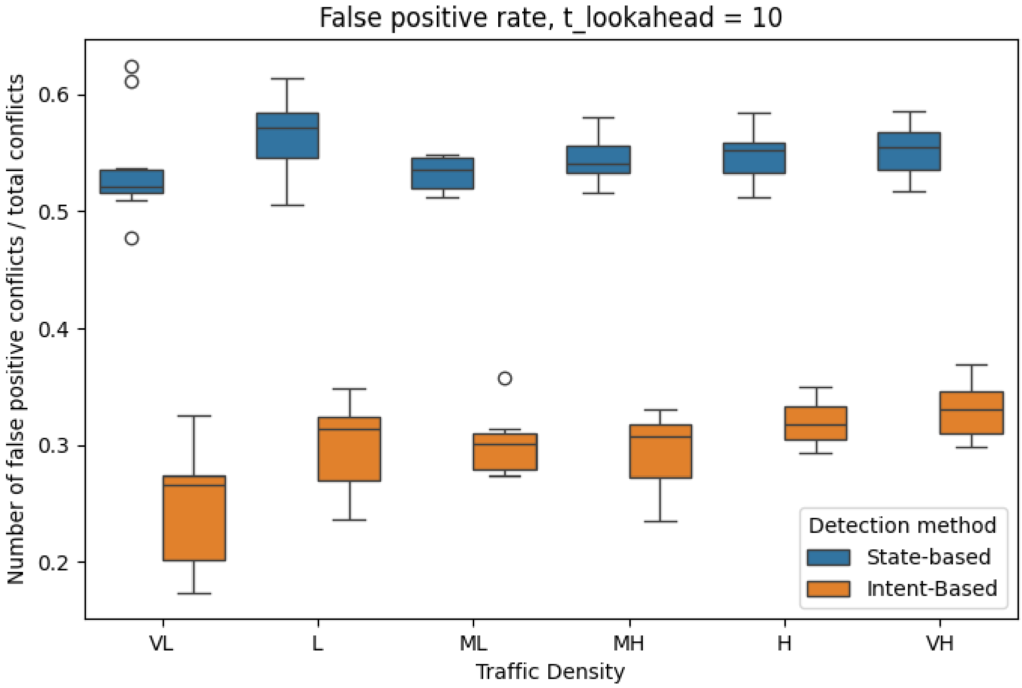


Figure C.4: False positive rate per traffic density with a look-ahead time of 10 seconds

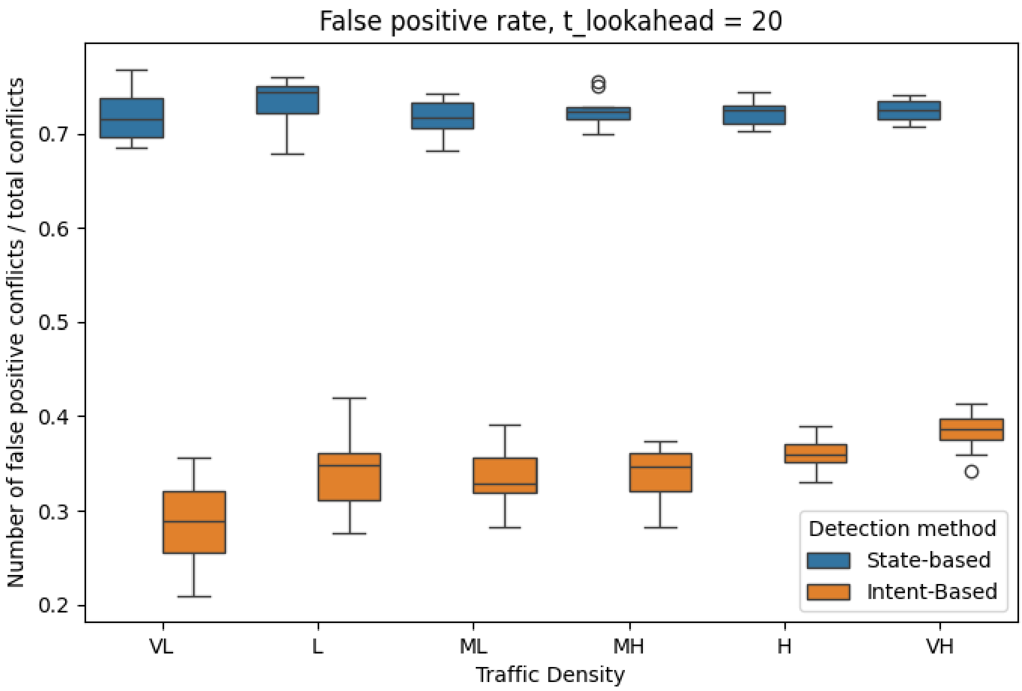


Figure C.5: False positive rate per traffic density with a look-ahead time of 20 seconds

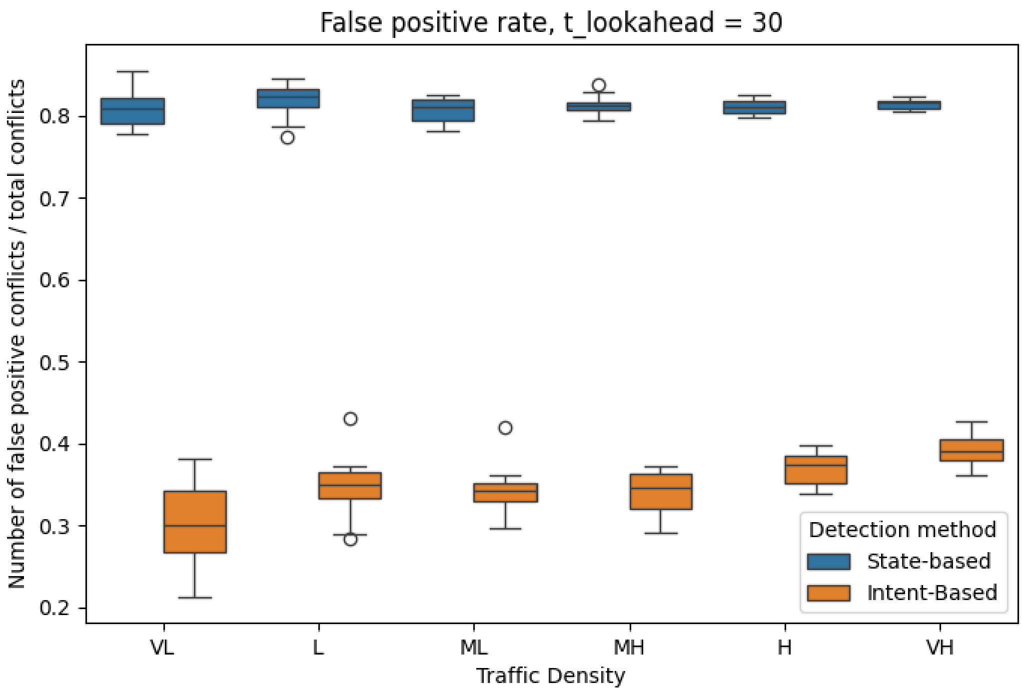


Figure C.6: False positive rate per traffic density with a look-ahead time of 30 seconds

D
Average Detection Time Before Conflict Plots

The average detection time before conflict plots are displayed in this appendix

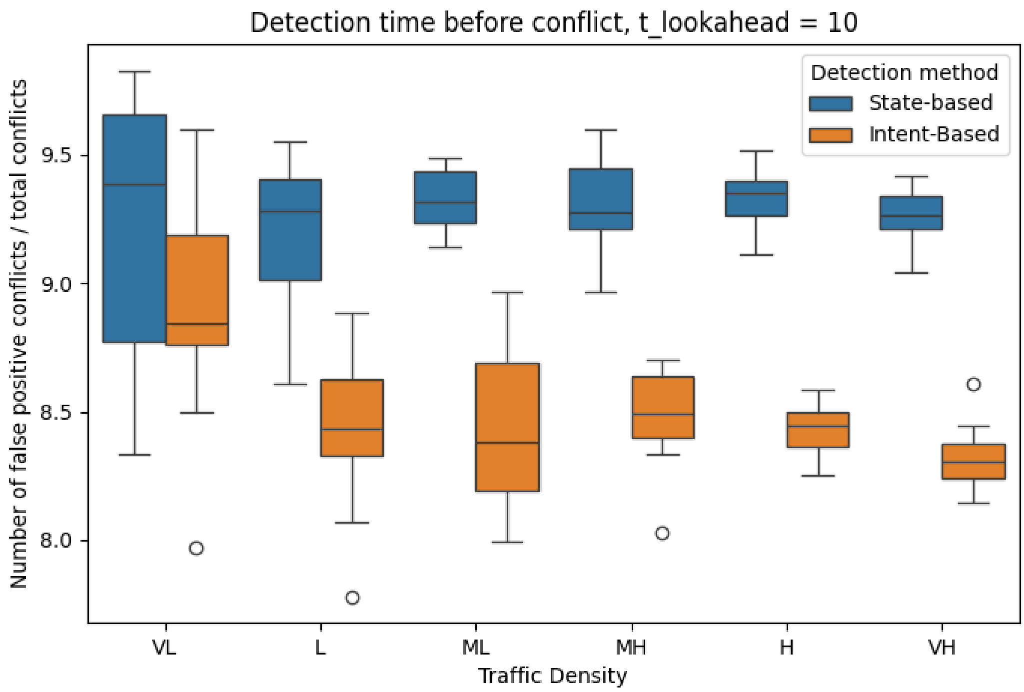


Figure D.1: Average detection time per traffic density with a look-ahead time of 10 seconds

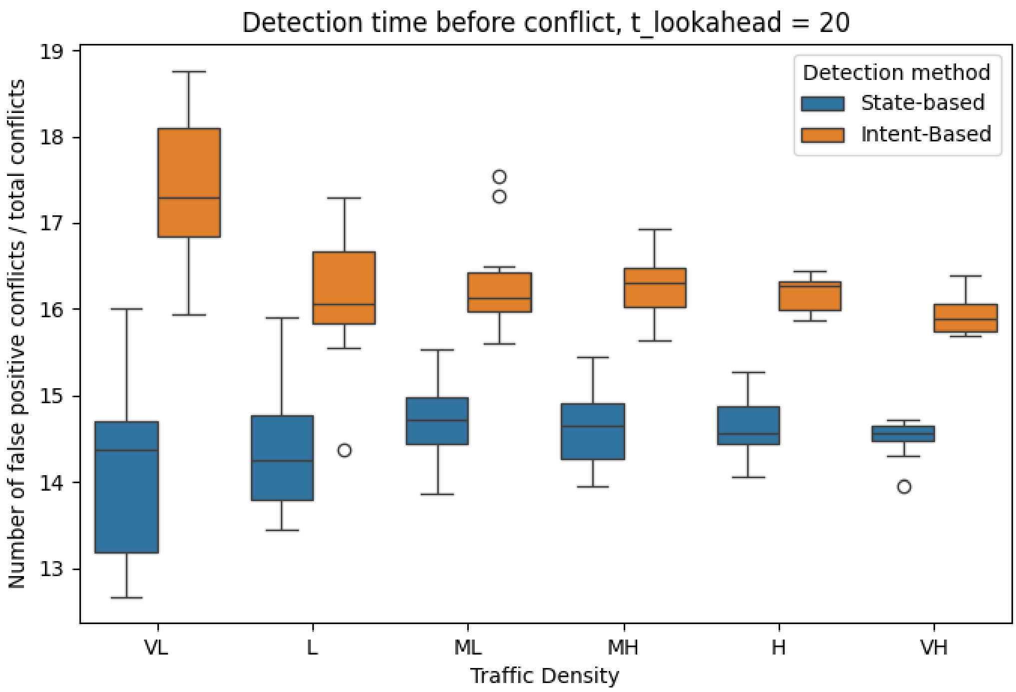


Figure D.2: Average detection time per traffic density with a look-ahead time of 20 seconds

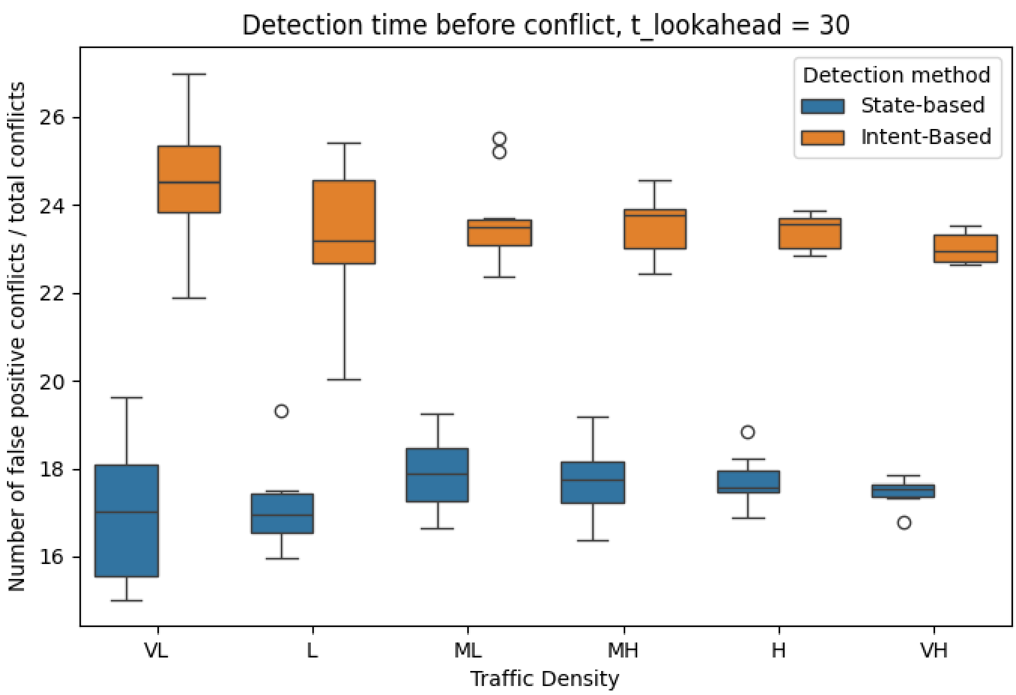


Figure D.3: Average detection time per traffic density with a look-ahead time of 30 seconds

III

Preliminary Report (Already Graded)

1
Introduction

According to SESAR JU the utilization of drones in civilian airspace is set to experience substantial growth in the upcoming decades [38]. Projections indicate that by 2050, Europe will witness the presence of approximately 7 million recreational drones and 400,000 commercial or government-operated drones. Unmanned aerial vehicles (UAVs) have become increasingly prevalent in a wide range of industries. Specifically in urban areas, the applications of drones are being considered for areas like urban air mobility [4], package delivery [27] and urban surveillance [16]. As these technologies continue to advance together with the popularity of drones, there is a growing need to ensure their safe and efficient operation, particularly in complex and crowded urban environments.

One of the key challenges in this area is the detection and resolution of conflicts between UAVs operating in very low-level urban airspace. While a lot of research has already been done surrounding this topic [23], [33], the main focus of most of these are based on detection using the current state of aircraft which has shown high effectivity [19]. An idea that has a lot of potential and has gained more traction recently [6], [25], [34], is the use of intent where the future trajectory of the aircraft is also considered in the conflict detection process. Through the use of intent, the number of false alarms can be reduced. This allows for better route efficiency as UAVs do not have to deviate from their course to avoid a conflict that does not exist. In addition, false negatives can also be detected through the use of intent, enhancing the safety of the drone [34].

The relevance of this work is significant both academically and in industry, as the development of intent-based Conflict Detection & Resolution (CD&R) techniques has the potential to significantly improve the safety of UAV operations. This in turn will allow for a easier integration of UAVs into urban environments.

In this project, we aim to investigate the effectiveness of intent-based conflict detection and resolution for UAVs operating in constrained very low-level urban airspace. The main goal is to develop a predictive intent-based model that can accurately identify potential conflicts and provide solutions to resolve the conflicts. This model will be tested and evaluated for its effectiveness and performance.

1.1. Research Questions

The objective of this research is *to create and evaluate a conflict detection and resolution method which incorporates intent for unmanned aerial vehicles operating in constrained very low-level urban airspace*.

From this research objective, the main research question is defined as follows: *"In what way can intent be used to further improve conflict detection and resolution for unmanned aerial vehicles operating in constrained very low-level urban airspace?"*

To answer this research question sub-questions are made to divide the research questions up into pieces which can individually be answered. These are defined as follows:

1. How can intent be used for CD&R?
2. How does the intent-based CD&R method perform?
3. In what environment is the intent-based CD&R method expected to work in?

4. Which metrics are important to evaluate the performance of intent-based CD&R Method?

Moreover, a literature study needs to be performed to answer questions relating to the research questions. To point the literature study into the right direction, questions are devised. These questions then serve as foundation of the literature review discussed in Appendix 2. The questions are the following:

- What is airspace?
- What kind of air services are available for urban airspace?
- What are characteristics of urban airspace?
- What is CD&R?
- How can CD&R be categorized?
- Which high performance CD&R methods can be used as baseline?
- What is intent?

1.2. Report Outline

The report is structured as follows. Firstly, a literature review is performed in Chapter 2. The literature review is performed on the topic of future airspace, CD&R and intent. In Chapter 3, a description is given of the experiment that will be performed. This description includes the resources that are used for the experiment, the experiment environment, the methodology and the hypotheses and variables that are used in the experiment. Lastly, in Chapter 4, the project timeline is shown. This timeline consists of work packages and is displayed in the form of a Gantt chart.

2

Literature Review

In this chapter, the literature review is discussed. The literature review is divided up into multiple broader topics which are then dived into deeper. section 2.1 goes over some basic definitions of the UAV. In section 2.2 the future airspace is discussed and what it may look like, including some concepts. CD&R methods are explained in section 2.3 as well as an elaboration on the state-of-the-art detection and resolution methods. In section 2.4, the definition of intent is formulated, and some papers are discussed which are related to intent in CD&R. Finally, a conclusion on the literature review is given in section 2.5

2.1. Basic Concepts & Definitions

This section goes over the basic concepts and definitions regarding this project.

2.1.1. UAV

The first definitions are given surrounding the word UAV. These definitions are given to these terms by Granshaw[17] and SKYbrary[42].

Table 2.1: Definitions according to Granshaw[17] and SKYbrary[42]

Term	Definition
Unmanned Aerial Vehicle	Airborne vehicle which is either remotely piloted or performing completely autonomous flight
Drone	Pilotless aircraft
Unmanned Aerial System	an unmanned aircraft and the equipment to control it remotely.

Do note that throughout the report UAV and drone are simultaneously used as synonyms.

2.2. The Future of Airspace

In this section, a literature study on airspace is performed regarding what it is and how it may look like in the future under an urban setting.

2.2.1. Airspace

In general, airspace is defined as the following: ”The section of Earth’s atmosphere which covers both land and sea and is regulated and administered by a specific state.” [2]. Each section can then be further classified as ”controlled” or ”uncontrolled” airspace. In controlled airspace, air traffic control is actively communicating, directing and separating all air traffic. This is not the case in uncontrolled airspace where air traffic is not directed or controlled [12].

Furthermore, airspace can be divided up into layers, each with their own rules regarding type of flight, separation, speed limit, communication requirements and ATC clearance. An overview of this division can be found in Figure 2.1.

Class	Type of flight	Separation Provided	Service Provided	Speed limitation*	Radio communication requirement	Subject to an ATC clearance
A	IFR only	All aircraft	Air traffic control service	Not applicable	Continuous two-way	Yes
B	IFR	All aircraft	Air traffic control service	Not applicable	Continuous two-way	Yes
	VFR	All aircraft	Air traffic control service	Not applicable	Continuous two-way	Yes
C	IFR	IFR from IFR IFR from VFR	Air traffic control service	Not applicable	Continuous two-way	Yes
	VFR	VFR from IFR	1) Air traffic control service for separation from IFR 2) VFR/VFR traffic information service (and traffic avoidance advice on request)	250 kts IAS below 10000 ft amsl	Continuous two-way	Yes
D (1)	IFR	IFR from IFR	Air traffic control service, traffic information about VFR flights (and traffic avoidance advice on request)	250 kts IAS below 10000 ft amsl	Continuous two-way	Yes
	VFR	Nil	IFR/VFR and VFR/VFR traffic information (and traffic avoidance advice on request)	250 kts IAS below 10000 ft amsl	Continuous two-way	Yes
E (2)	IFR	IFR from IFR	Air traffic control service and, as far as practical traffic information about VFR flights	250 kts IAS below 10000 ft amsl	Continuous two-way	Yes
	VFR	Nil	Traffic information as far as practical	250 kts IAS below 10000 ft amsl	No	No
F	IFR	IFR from IFR as far as practical	Air traffic advisory service; flight information service	250 kts IAS below 10000 ft amsl	Continuous two-way	No
	VFR	Nil	Flight information service	250 kts IAS below 10000 ft amsl	No	No
G	IFR	Nil	Flight information service	250 kts IAS below 10000 ft amsl	Continuous two-way	No
	VFR	Nil	Flight information service	250 kts IAS below 10000 ft amsl	No	No

* When the height of the transition altitude is lower than 10,000 ft amsl, FL100 should be used in lieu of 10000 ft

Figure 2.1: Overview airspace classification. From [40]

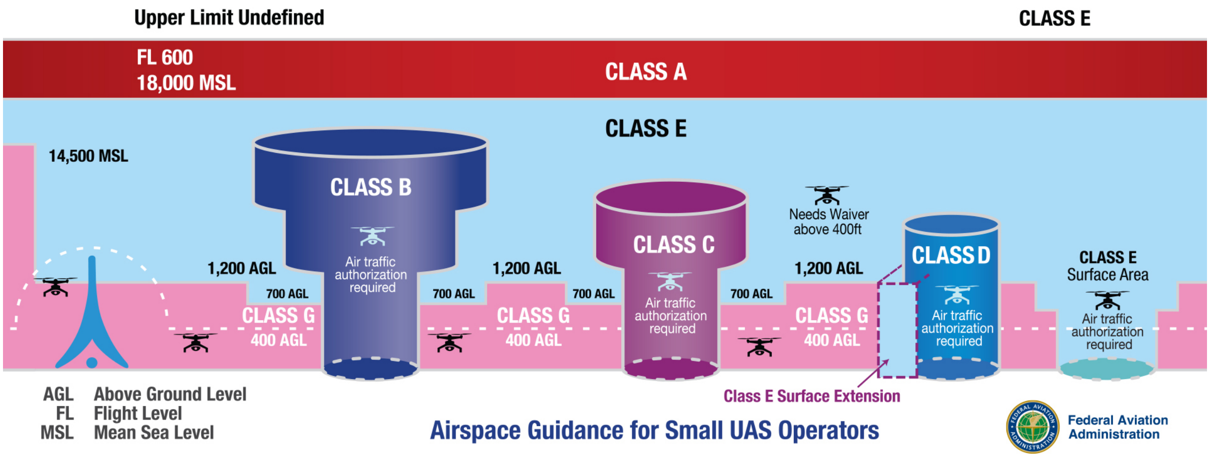


Figure 2.2: Graphical Overview Airspace. From [12]

A graphical overview of the division of airspace can be found in Figure 2.2.

With the rise in popularity of UAV's and the possibilities in urban environments, an additional airspace, meant specifically for the use of UAS. In Europe it is called U-space whereas in the United States of America it is dubbed Unmanned Aircraft System Traffic Management (UTM).

2.2.2. U-space

The U-Space project, initiated by the Single European Sky ATM Research Joint Undertaking (SESAR JU), is a project which goal is to support the increasing demand for drone services which in turn could result in significant economic growth and social benefits [39] [37]. More specifically, "U-space is a set of new services and specific procedures designed to support safe, efficient and secure access to airspace for large numbers of drones" [39]. Of note is the fact that U-space is not a defined volume of airspace specifically for drones but rather a framework which supports drone operations as well as be an effective interface between drones and manned aviation, ATM/ANS, service providers and authorities [39]. To ensure the reachability of the goal of the U-space project, a roadmap has been created for the development of services. This roadmap is illustrated in Figure 2.3. The project has been divided up into work packages with each work package expanding upon the developments of the preceding work package.

The foundation services (U1) serve as the groundwork of the U-space project [36]. The services that are the subject of this phase of the project are the registration, registration assistance, e-identification and geo-awareness. Through the registration service, drone owners are able to register their drones, its operator and its pilots. The e-identification allows information about the drone to be verified without physical access to the drone. The geo-awareness service provides a geo-fence and other flight restrictions to drone pilots at request before take-off. It also provides existing aeronautical information such as information from NOTAMS and temporary restrictions from the national airspace authority.

The initial services (U2) focus on supporting the management of drone operations [36]. The support is provided in multiple ways and are summarized below:

- Tactical geofencing, updates the operator with geofencing information during flight
- Tracking of individual drones
- Flight planning management, checking the flight plans before take-off
- Providing weather information
- Drone aeronautical information management, provides operators with relevant aeronautical information for drone operations and will be connect to the Aeronautical Information Service (AIS).
- Emergency management
- Provides a procedural interface with ATC

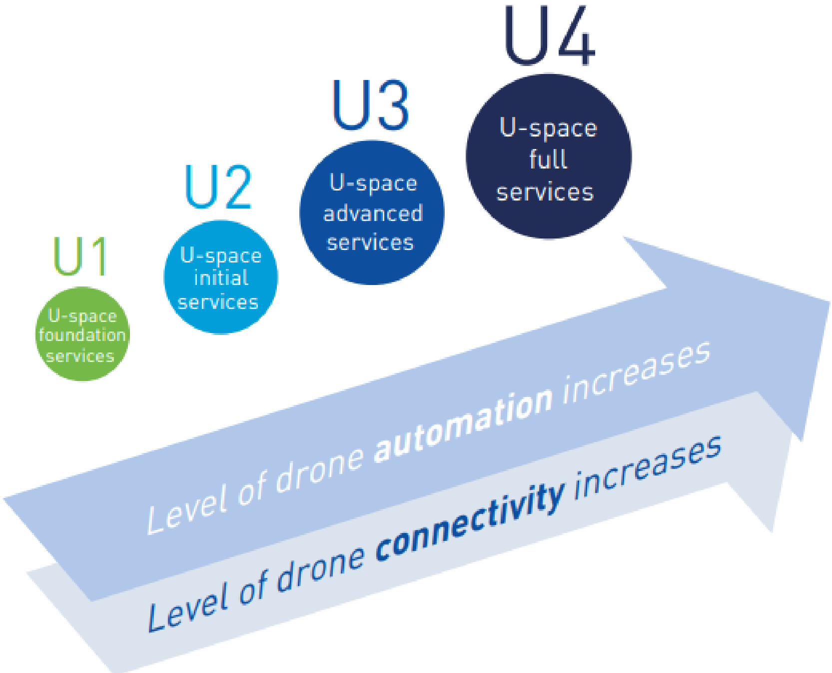


Figure 2.3: Roadmap of the U-Space Project. From [39]

- Strategic de-confliction between drones
- Monitoring of the air situation
- Providing Traffic Information

The advanced services (U3) uses the service of U2 and develops them even further as stated by SESAR JU [36]. To be precise, U3 puts its focus on geofencing, interface with ATC, deconfliction and capacity management. The service will provide dynamic geofencing which is drone-specific. Furthermore, a collaborative interface with ATC is developed to ensure proper and effective coordination with ATC. Moreover, a focus is put on tactical deconfliction to further enhance safety and lastly, the capacity is managed dynamically making use of drone density thresholds.

The full services (U4) will allow full integration with manned aviation and air traffic services, as well as supporting the full operational capability of U-space. The services that are required to reach this goal have not been set but are expected to appear once U3 is applied. [36]

Overall, the U-Space project strives to establish a comprehensive framework of services and procedures to ensure the safe and efficient integration of drones into European airspace. By gradually introducing increasingly sophisticated services, the project aims to support the growth and development of the drone industry while maintaining airspace security and effectiveness.

2.2.3. Unmanned Aircraft System Traffic Management

The Unmanned Aircraft System Traffic Management is a collaboration by the FAA, NASA and other federal and industry partners. The project focuses on developing concepts of operation, data exchange requirements and a supporting framework to enable multiple beyond visual line-of-sight operations at altitudes below 400 feet and above ground level [13]. Similar to the U-space project, UTM is divided into smaller goals called Technical Capability Levels (TCL) as stated in the Research Transition Team (RTT) plan [24]. The TCLs can be seen in Figure 2.4. The aim is to achieve a higher TCL in a time span of 12-18 months.

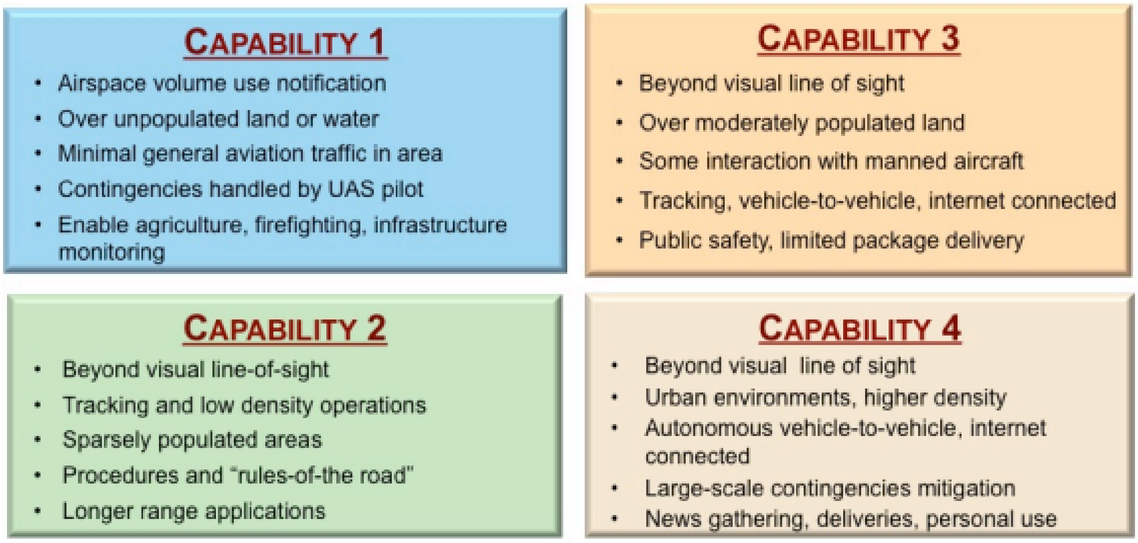


Figure 2.4: Technical Capability Levels. From [24]

Furthermore, the RTT has been subdivided into four distinct sub-groups, each focusing on a different aspect. These are the Concepts and Use Case, Data Exchange and Information Architecture, Sense and Avoid and Communication & Navigation [24]. Moreover, the RTT utilizes Joint Management Plans (JMPs) which asks for stakeholders and partners to develop alongside the FAA and NASA.

In addition, the UTM Pilot Program (UPP) is developed through which the results of prototypes can be evaluated for UTM [14]. In phase one, UPP was able to demonstrate services such as the exchange of intent, the generation of notifications to UAS operators and the ability to share UVRs with stakeholders. The UPP is currently in phase two where it focuses on the deployment of Remote Identification technologies in increasingly complex environments.

2.2.4. Urban Airspace Designs

Some early concepts have been developed to see what the future urban airspace might look like. An example of such is the Metropolis project where the influence of airspace structure is analyzed to see its impact on capacity, complexity, safety, and efficiency [43].

In the paper by Sunil et al. [43], four different airspace structure concepts are compared to each other. They however, do have some similarities when it comes to Airborne Separation Assurance System (ASAS) and Airspace Limits. Each individual drone is responsible for CD&R and is handled with the help of the ASAS implementation of Hoekstra, Gent, and Ruigrok [21]. The conflict detection method used is a state-based extrapolation of traffic positions with a look-ahead time of 60 seconds. The conflict resolution is done through the Modified Voltage Potential (MVP) algorithm. For the airspace limit all concepts assume that the drones are flying above buildings at an altitude between 1100 ft and 6500 ft.

The first concept is a Full Mix or unstructured airspace [43]. In this concept there are no restrictions on the path of aircraft, heading, speed and altitude. The idea behind this concept is that structuring results in an overall decrease in traffic efficiency and safety is enhanced by spreading traffic.

The second concept is a layered airspace design [43]. In this concept, the airspace is divided up into layers through which horizontal travel is possible. Moreover, at each travel is only allowed into a certain heading range. Of note is that in this concept, two complete sets of layers are used to offset the efficiency loss of traffic.

In the third concept, the airspace is divided into different sectors corresponding to the layout of the city. The city structure that is used in the paper is displayed in Figure 2.5. The structure is a circular design consisting of radial and circular zones. The radial zones take care of traffic going towards and away from the city center as well as connecting the different rings. Furthermore, altitude is selected flexibly based on the distance between origin and destination.

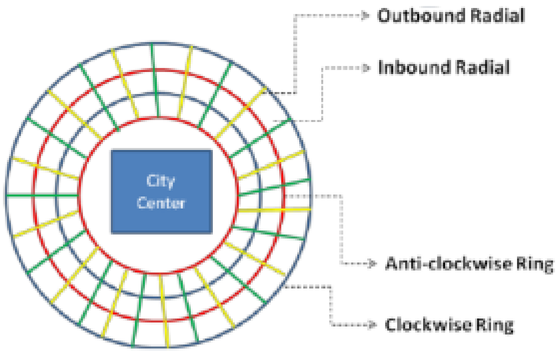


Figure 2.5: Top-down view of the Zones topology, which is designed to take into account the layout of a city. From [43]

The fourth concept is a tube concept where the airspace is structured as much as possible [43]. This is done to increase the predictability of traffic flows by ensuring pre-planned conflict-free routes. The airspace is divided up into layers of nodes, with each layer having a finer or coarser distribution of nodes. The nodes serve as connection points of the tubes which go from node to node. The tube concept, unlike the other concept, also takes into account time to keep separation between aircraft. This is done by keeping track of the occupation of nodes as only one aircraft can pass a node at a certain time.

The result of the paper showed that the layered airspace design has the best balance between efficiency and safety. Further recommendations to progress with this concept include investigating how to optimize CD&R or flight rules to further improve the concept.

2.2.5. Constrained Airspace Designs

The design of constrained urban airspace has a big influence on the performance of CD&R methods. Hence, a lot of designs have been tried to ensure the safety and efficiency of urban traffic.

In the paper by Doole et al., multiple designs of constrained urban airspace are proposed [11]. The airspace is designed in terms of the travel method of drones, the directionality of streets, and layering of airspace.

The paper argues for travelling above the street as travel method. The paper argues that this is the best option due to a lower privacy risk as well as allowing ground delivery modes such as trucks to work in tandem with drones and optimize the last-mile delivery [1].

Furthermore, the paper tests and compares two different layering and directionality concepts. Two designs are created, one utilizes a one-way street network whereas the other makes use of a two-way street network.

The two-way layered airspace consists of 40 layers in total organised into 2 distinct type of layers. There are turn layers and through layers. At turn layers, aircraft who need to make a turn can do so without slowing down surrounding aircraft and turn in a safer manner. In addition, the layer system contains North, East, South and West bound layers. The distinction between these layers are the headings that traffic are allowed to travel towards. The distinction is made to prevent interactions of different traffic flows at intersections. A drone is assigned to a certain layer, based on the distance and time the drone will take to reach its destination. Drones who travel further will fly at a higher altitude than drones who do not travel as far.

Similarly, the one-way concept consists of 40 layers with the same division as the two-way concept. The difference is that due to the uni-directionality of the streets, the one way concept allows twice as many layers to be used per cardinal direction. This amounts to 20 layer each direction. The layers are sorted as displayed in Figure 2.7.

The paper compared both one-way and two-way concepts and concluded that in terms of safety, the one-way concept performed better. In terms of efficiency, both concepts had a similar performance.

Another constrained airspace concept is that of Badea et al. Badea et al. Likewise, the urban airspace is designed with the directionality of streets and layers in mind. In this paper, the roads have been made unidirectional which is done to prevent head-on collisions as well as structuring the airspace resulting in less conflicts. Moreover, global road directions are assigned. These are either in a North-South

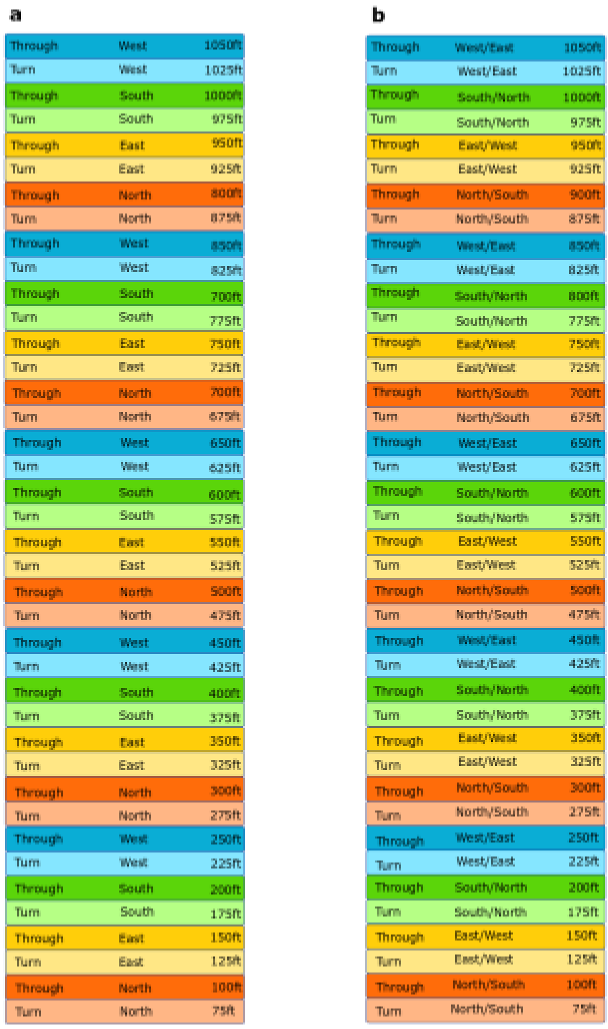


Figure 2.6: Schematic view for the complete set of altitude bands for the two urban airspace designs, where each altitude layer corresponds to its respective travel direction. Each concept has 40 altitude layers (which consist of 20 altitude layers allocated for through traffic and 20 altitude layers for turn traffic) from 75 to 1050 ft with a vertical spacing of 25 ft, respectively. The turn-layers are used for transitory flights, that is, drones that need to make turns/change direction at intersections. While the through-layers are utilised by through traffic, that is traffic passing through at least one intersection. (a) The two-way concept layer system for which traffic is allocated to the cardinal directions with respect to flight headings (ψ): 315◦ < ψ ≤ 045◦ headings assigned to north; 045◦ < ψ ≤ 135◦ headings to east; 135◦ < ψ ≤ 225◦ headings to south; and 225◦ < ψ ≤ 315◦ to west bound traffic. (b) One-way layer system where traffic with flight headings: 315◦ < ψ ≤ 045◦ and 135◦ < ψ ≤ 225◦ are assigned to north and south layers; and, traffic with flight headings between 045◦ < ψ ≤ 135◦ and 225◦ < ψ ≤ 315◦ are allocated to the east and west bound altitude layers. Of note, recognise that, when compared to the two-way concept, the one-way concept has north/south and east/west traffic assigned to one altitude layer, for which the separation of this traffic is assured by the spatial geometry of the urban street network. From [11]

direction or an East-West direction. This is used in the layering of the street network. The directionality is determined using the COINS algorithm [44]. After that, the direction of each road is assigned using a genetic algorithm such that all areas are accessible.

The paper also designed airspace layers through which the drones travel. The layers are made up of cruising layers and turning layers. Furthermore, the cruising layers are different for North-South direction and East-West direction. This layer design is made to ensure that aircraft who slow down to make a turn, do not affect other aircraft in the same layer. The lay-out of the layers is visualized in Figure 2.7.

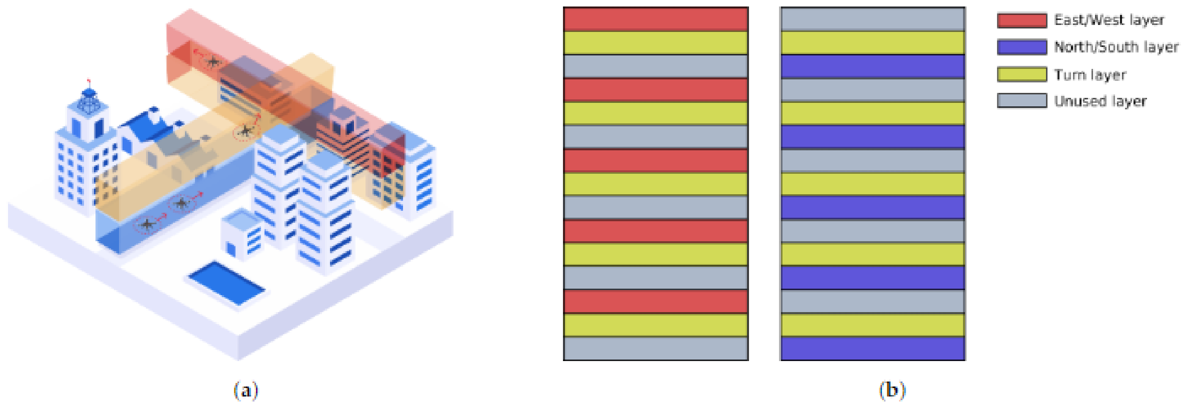


Figure 2.7: The layer configurations used within constrained airspace. (a) Isometric view of layers at intersections, configured such that cruising aircraft do not interfere with each other, and turn layers are always at the same altitude; (b) Layer configuration in function of street categorization. Certain altitudes are reserved for cruising either in the east/west or north/south directions. Each layer is 30 ft in height. From [6]

As discussed by the paper, a lot of conflicts stem from drones vertically merging into the turn layer. To solve this problem, an improved maneuvering concept is applied. This concept entails that the drone first checks whether the layer above or below are free of conflicts if it decided to merge. If that is not the case, the drone will perform its turn in the cruise layer itself. The drone will than merge back into the cruising layer once it has finished its turn.

The paper compared the improved and the baseline version of the layering system as part of conflict prevention. It concludes that the improved concept without the necessity to merge has a better safety performance compared to the baseline version. From the results it was observed that most aircraft made turns in the cruise layer which resulted in less complex conflict situations and more stability.

2.2.6. Urban Airspace Centralization

Other than the design of urban airspace, there is also discussion surrounding the degree of centralization in the management of separation. This matter is discussed in the Metropolis 2 project where the impact of centralized separation management is studied [29]. In the paper three different levels of centralization are compared. These are between fully centralized design, a hybrid design where there is a degree of centralization and a degree of tactical separation and a fully decentralized design.

In the centralized concept, the separation is done through pre-flight deconflicting [29]. The concept makes use of the layered airspace design as described in the paper of Sunil et al. [43]. Conflicts are minimized through a two-step process. First the number of conflicts are minimized through assigning flights to certain layers. Then, the remaining conflicts are solved by delaying the departure slot of flights.

In the hybrid concept, there are both elements of strategic deconfliction and tactical deconfliction [29]. In this concept a central system takes care of strategic horizontal separation making use of flight planning. The vertical separation is performed through vertical and speed maneuvers on a tactical basis. Furthermore, the airspace is divided up into constrained and unconstrained airspace. In unconstrained airspace there are no limitations to where you can fly while in constraint airspace there are limitations. In this case the constrained airspace is the city center where the aircraft have to fly above the street network. The layout of the hybrid concept consists of radial unidirectional rings in unconstrained airspace which are then connected to the constrained street network. Furthermore, a penalty is given when traveling through the city center to reduce traffic within this area.

The decentralized concept has not strategic planning and relies on tactical deconfliction between aircraft [29]. The tactical separation is done through speed and altitude maneuvers. In the constrained airspace, layers are once against used in addition to the street network only employing one-directional streets. Moreover, each layer has a heading range where each aircraft is only allowed to travel in that layer when its heading falls within that range. To perform turns, the concept makes use of turn layers located between cruising layers. Lastly, a cost function is constantly updated based on traffic, allowing aircraft to take a different route if the cost is too high.

The results of the paper concluded that the hybrid concept has the best performance regarding safety, but this came at a cost of route efficiency and delay [29]. The paper proposes to continue with the hybrid concept and to make further improvements by reducing the strictness of strategic planning and more sophisticated path planning.

2.2.7. Conclusion

To conclude, research towards urban airspace has come to life in recent years. A lot of different designs have been tested and tried, each with their own performance in terms of safety, efficiency. It can be concluded that not only, the CD&R method has a role in avoiding conflicts but also the design of airspace. A takeaway from the literature study on airspace is that the use of layers in airspace has potential as it has the best combination between safety and efficiency. Furthermore, one-way streets also show to be safer while having similar performance to two-way streets. Both these concept will be considered when designing the airspace for the experiment.

An interesting take-away from the paper by Badea et al. is the fact that turn layers cause a destabilizing effect on traffic. An area to explore further in is then the disregard of turning layers and creating a layered system without turning layers.

2.3. CD&R

In this section, a taxonomy on CD&R is discussed as well as some of the staple CD&R algorithms/methods. First a definition is given to conflict detection and conflict resolution. After that a taxonomy is discussed and lastly some CD&R algorithms are elaborated on.

2.3.1. Conflict

To be able to define what exactly conflict detection and resolution is, the term conflict first needs to be defined. According to Kuchar and Yang a conflict can be described as ”an event in which two or more aircraft experience a loss of minimum separation. In other words, the distance between aircraft violates a criterion defining what is considered undesirable.” [26]. With this definition it is easy to understand what conflict detection is and conflict resolution. The goal of a CD&R system is then to ”predict that a conflict is going to occur in the future, communicate the detected conflict to a human operator, and, in some cases, assist in the resolution of the conflict situation” [26]. In the case of unmanned aerial vehicles without pilots, this would mean that the resolution part itself is automated and no human interference is required.

Moreover, there are other key terms that need to be understood relating to CD&R. First of all, is the closest point of approach (CPA). This is defined as ”an estimated point in which the distance between the own ship and another object target will reach the minimum value.” [35].

Another key term is the protected zone of an aircraft. By ICAO standards [41], an aircraft should be separated from other aircraft horizontally with a minimum distance of 5 nautical miles when using surveillance systems such as radar and ADS-B. Vertically the minima is 1000 feet below an altitude of FL290. Together these minima form the protected zone of the aircraft. For UAVs, there are no specified protected zone requirements.

Table 2.2: Definitions CD&R

Term	Definition
Conflict	An event in which two or more aircraft experience a loss of minimum separation.
Conflict Detection	Predict if a conflict is going to occur in the future.
Conflict Resolution	Resolve a conflict situation
Closest Point of Approach	An estimated point in which the distance between the own ship and another object target will reach the minimum value.
Protected Zone	A zone around an aircraft defined by horizontal and vertical separation minima which may not be breached by other aircraft

2.3.2. CD&R Taxonomy

CD&R has been extensively researched in aviation. This was first done for manned flights but with the rise of UAVs, a lot of research has gone into CD&R for unmanned flights. The taxonomy helps to categorize and order the different CD&R methods based on certain criteria.

According to Jenie et al. [23], the three major factors that contribute to a multi-layered safety CD&R system are the type of surveillance, coordination and maneuver. Ribeiro, Ellerbroek, and Hoekstra [33] proposes a different taxonomy with more factors. This gives more distinguished differences between CD&R methods. On top of that [33] is published more recently and hence includes the methods that have been developed since the writing of [23]. For these reasons, the taxonomy of [33] is used to provide an overview of CD&R methods for unmanned aviation. In Table 2.3 and Table 2.4 the taxonomy categories are displayed for Conflict Detection and Conflict Resolution respectively.

Table 2.3: Conflict Detection Categories. From [33]

Conflict Detection Categories		
Surveillance	**Trajectory Propagation**	**Predictability Assumption**
Centralized Dependent	State-Based	Nominal
Distributed Dependent	Intent-Based	Probabilistic
Independent		Worst-Case

Table 2.4: Conflict Resolution Categories. From [33]

Conflict Resolution Categories						
Control	**Method Categories**	**Multi-actor Conflict Resolution**	**Avoidance Planning**	**Avoidance maneuver**	**Obstacle Types**	**Optimization**
centralized	Exact	Sequential	Strategic	Heading	Static	Flight Path
Distributed	Heuristic	Concurrent	Tactical	Speed	Dynamic	Flight Time
	Prescribed	Pairwise Sequential	Escape	Vertical	All	Fuel/Energy Consumption
	Reactive	Pairwise Summed		Flight Plan		
	Explicitly Negotiated	Joint Solution				

Surveillance The surveillance category indicates whether the drone uses an external or on-board systems for surveillance. The centralized dependent surveillance systems are able to be located by ground sensors through interrogation. The distributed dependent systems make use of ADS-B and are able to communicate their position, velocity, altitude, etc. to other aircraft without the use of ground systems. The independent surveillance systems are often referred to as Sense and Avoid and are reliant on sensors to detect static and dynamic objects.

Trajectory Propagation Trajectory Propagation takes into account how the future position of aircraft is determined. State-based trajectory propagation extrapolates the current states in a straightforward fashion. Intent-based trajectory propagation takes into account future changes in speed, heading, altitude, etc. creating a more accurate future prediction but in turn being more complex and requiring more computational power. The type of trajectory propagation is of importance as intent could reduce the number of false positives and false negatives that would not be detected when using state-based trajectory propagation.

Predictability Assumption There is always a certain degree of uncertainty present when estimating the future position of aircraft. The predictability assumption deals with the degree of uncertainty that is used in the CD&R method. A nominal assumption does not take into account uncertainty at all which makes for a simpler method. A probabilistic method finds the likelihood of a trajectory change and the method assumes the trajectory with the highest likelihood. The worst-case method considers all trajectory changes possible due to uncertainties. Adding uncertainties to your method is a way to create more accurate results, but in exchange requires more computational power and an increase in false positives.

Control Control, or separation management, defines in what manner decisions surrounding conflict resolution and trajectory changes are managed. This can be centralized where the decisions are computed in a centralized location for multiple drones. Another option is distributed where each individual drone makes their own decisions.

Method Categories The methods that are used in CD&R can be divided into roughly 5 method categories. The two methods that are used in centralized approaches are exact and heuristic methods. Exact methods try to find the best global optimum solution when optimizing conflict resolution. This however, requires a lot of computing time and hence heuristic methods are used to reduce computing time in exchange for not guaranteeing the most optimal solution. For decentralized methods, prescribed and reactive both resolve conflicts implicitly, which means a pre-defined set of rules or maneuver strategy is set in place which drones adhere to. Explicitly negotiated methods require communication between drones to resolve their conflict. The movements of the drones in conflict are exchanged and a solution is drawn up between the two.

Multi-Actor Conflict Resolution Multi-Actor Conflict Resolution methods are once again different for centralized and distributed methods. The centralized methods make use of sequential and concurrent resolution. In sequential resolution, conflicts are solved one-by-one whereas in concurrent the conflicts are solved all in one go. In distributed methods, individual conflict resolutions may cause secondary conflicts and result in global sub-optimal solutions, especially in higher traffic densities. Hence, it is crucial to see how distributed methods deal with multi-actor conflicts and resolve them. Pairwise methods resolve one-on-one conflicts. In pairwise sequential, the conflicts are solved one after the other starting with the highest priority. In pairwise summed, the resolution vector of each maneuver is summed giving one single maneuver to be performed. A joint solution considers multiple aircraft, and the conflicts are then solved in one solution.

Avoidance Planning Avoidance planning methods mainly determines the nature of the avoidance maneuver. Strategic methods usually involve a long-range action which may change the flightpath significantly. Tactical is a mid-range action and changes the flightpath slightly. An escape maneuver is a short-term action which does not take into account flight path changes as long as conflict is avoided.

Avoidance maneuver Avoidance maneuver deals with the change that is made to the aircraft to avoid conflict. One or more changes can be made to avoid conflicts. The changes that are made are changes in heading, speed or altitude. Also changes to the flight plan can be made to force the aircraft to take a different set of waypoints.

Obstacle Types This defines the type of obstacles the drone may encounter and is pretty straight forward. Dynamic obstacles define whether the obstacles are in motion or static when they are not moving.

Optimization The prime goal of CD&R is to ensure safety. However, once that goal is reached, the CD&R method may be able to be further optimized to make the operation more efficient whilst ensuring safety. The variables for which CD&R methods are optimized, are the flight path, flight time and fuel/energy consumption.

2.3.3. State-Based Conflict Detection

As described in Section 2.3.1, a conflict is an event in which two or more aircraft experience a loss of separation. The loss of separation happens when the CPA of the aircraft falls within the protected zone of another aircraft, to calculate if and when this happens is the goal of conflict detection.

To demonstrate the calculations for a state-based detection method, a scenario is taken where two aircraft are going in the same direction and will cross paths at some distant time. A visualization of this scenario is given in Figure 2.8. The aircraft with a circle around it will be called the ownship and the other aircraft will be referred to as the intruder. Some key variables are the relative velocity, relative distance and the CPA.

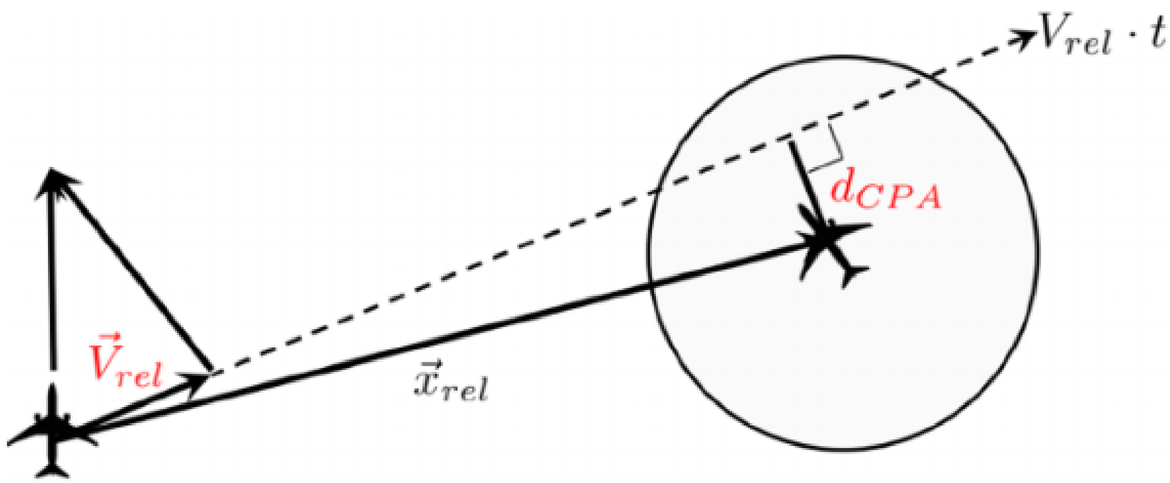


Figure 2.8: Conflict Scenario adapted from [19]

V_{rel} is the relative velocity vector of the intruding aircraft compared to the ownship. This is obtained by taking the velocity vector of the intruder and subtract it by the velocity vector of the ownship as displayed in Equation 2.1

$$\mathbf{V}_{rel} = \mathbf{V}_{intruder} - \mathbf{V}_{ownship} \tag{2.1}$$

The relative distance is the distance vector between the intruder and ownship. It is easily calculated by taking the position vector of the ownship and subtracting the position vector of the intruder.

$$\mathbf{X}_{rel} = \mathbf{X}_{ownship} - \mathbf{X}_{intruder} \tag{2.2}$$

To find the CPA first the time to CPA must be calculated. This is done as displayed in Equation 2.3. The CPA distance to the ownship can then easily be calculated using Pythagoras theorem. If this distance is then smaller than the separation minima, the aircraft are in conflict.

$$\begin{aligned} \mathbf{V}_{rel} &\perp (\mathbf{X}_{rel} - \mathbf{V}_{rel} \cdot t_{CPA}) \\ \mathbf{V}_{rel} \cdot (\mathbf{X}_{rel} - \mathbf{V}_{rel} \cdot t_{CPA}) &= 0 \\ t_{CPA} &= \frac{\mathbf{V}_{rel} \cdot \mathbf{X}_{rel}}{|\mathbf{V}_{rel}^2|} \end{aligned} \tag{2.3}$$

$$d_{CPA} = \sqrt{\mathbf{X}_{rel}^2 - (\mathbf{V}_{rel} \cdot t_{CPA})^2} \tag{2.4}$$

2.3.4. State-Based Conflict Resolution

A concept that is used often in state-based conflict resolution is the concept of velocity obstacles [15]. This algorithm utilizes collision cones to establish regions where the velocity vector of an object should not reside, in order to prevent conflicts.

The collision cone is constructed by drawing two lines who are tangent to the protected zone circle as can be seen in Figure 2.9. If the relative velocity falls within this cone, the aircraft will at some point intrude the protected zone and cause a conflict.

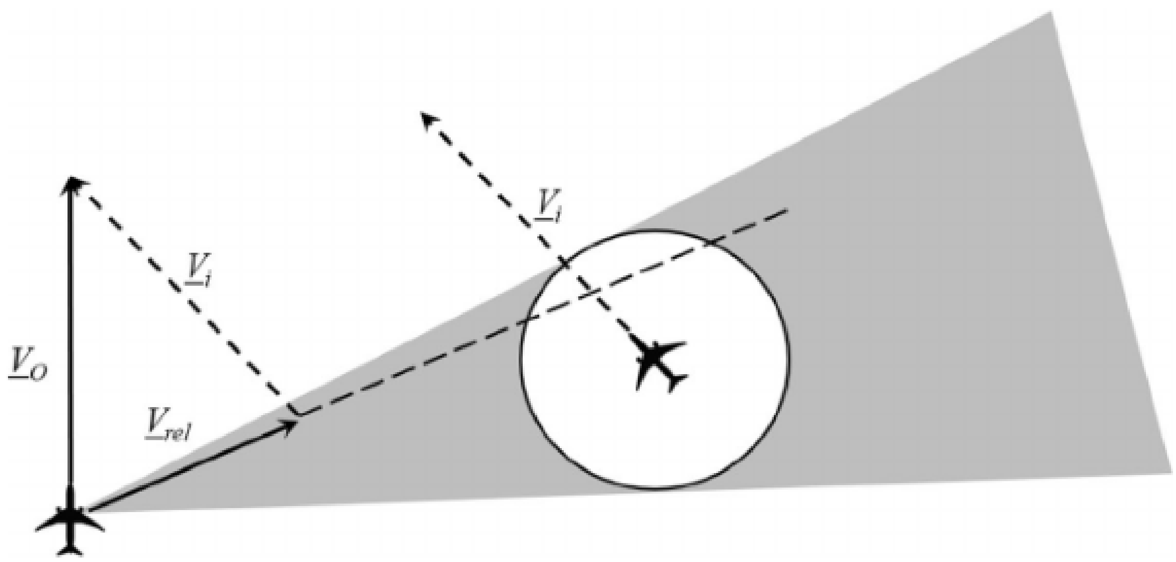


Figure 2.9: Velocity Obstacle from [19]

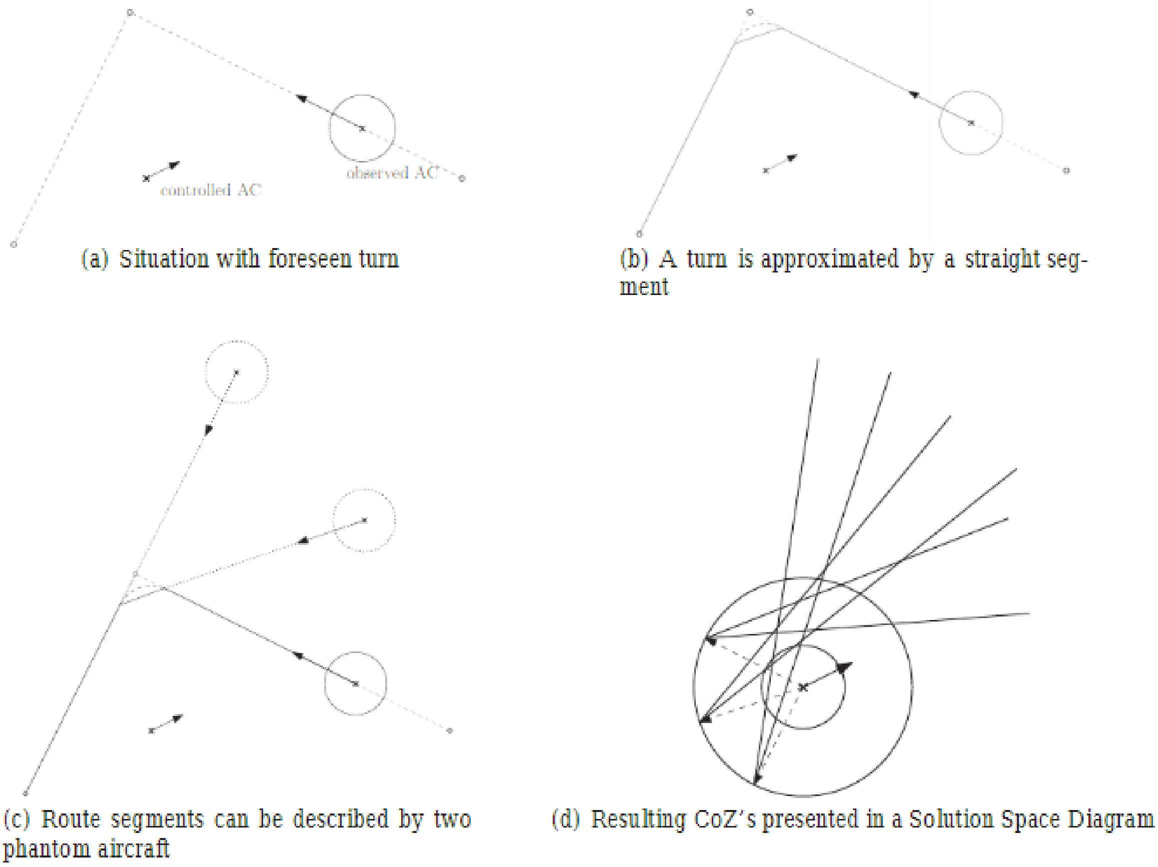
The velocity cone can then be translated onto the true velocity vector to give an area in which the true velocity vector should not be in. From this visualization it is then easily seen what velocity the aircraft should adapt to avoid conflict.

Complex trajectories, such as turns, or velocity changes, are a more complicated matter. d'Engelbronner et al. developed a method to also allow VOs to be created for complex trajectories [9]. The complex trajectory is segmented into multiple linear parts for which individual VOs are created. Give for instance a turn, the turn is then segmented into multiple aircraft each with a different heading which is tangent to the turn as demonstrated in Figure 2.10.

These so-called "phantom" aircraft each have their own VOs which are plotted. This gives a messy group of VOs going into different directions without a clear area to avoid. The overlapping area of all VOs is considered the VO of the turning aircraft and this VO should then be avoided to avoid conflict. An illustration is given in Figure 2.11.

2.3.5. Solution Space Diagram

The Solution Space Diagram (SSD) method makes use of the concept of velocity obstacles as a means for conflict resolution as applied in [7]. The velocity obstacles of all neighbouring aircraft are created which are the forbidden velocities (FV). Also, an area around the aircraft is marked which are the velocities that the aircraft can reach (RV). Applying this new area to the FV, a new area is obtained called the forbidden reachable velocities (FRV). The FRV should be avoided and everything that remains of the RV is labeled as allowed reachable velocities (ARV) and can be used to avoid conflicts. A visualization is given in Figure 2.12.



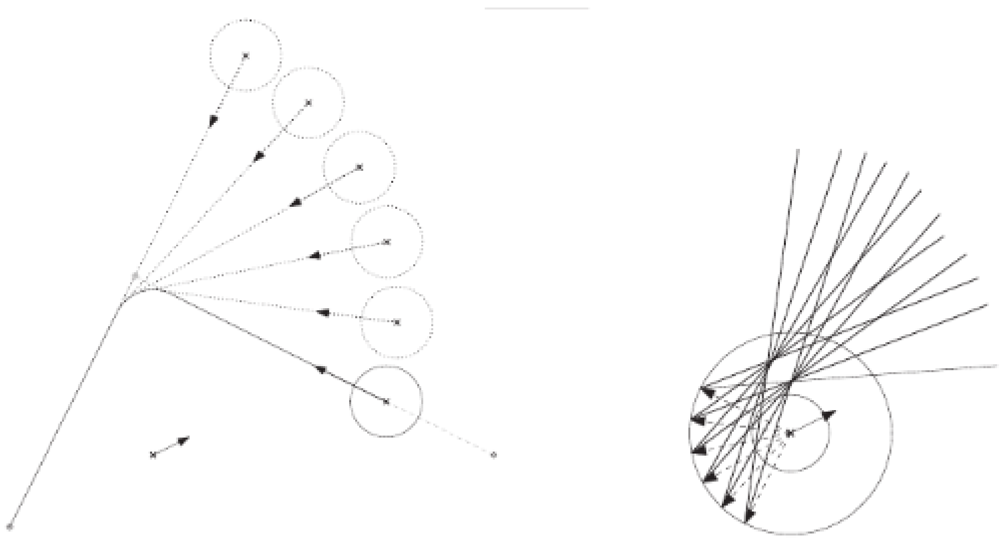
(a) Situation with foreseen turn

(b) A turn is approximated by a straight segment

(c) Route segments can be described by two phantom aircraft

(d) Resulting CoZ's presented in a Solution Space Diagram

Figure 2.10: In a situation with a foreseen turn, the SSD can be approximated using three phantom aircraft. From [9]



(a) A turn is approximated by multiple route segments and thus by multiple phantom aircraft

(b) The turn can be represented by multiple (phantom) conflict zones

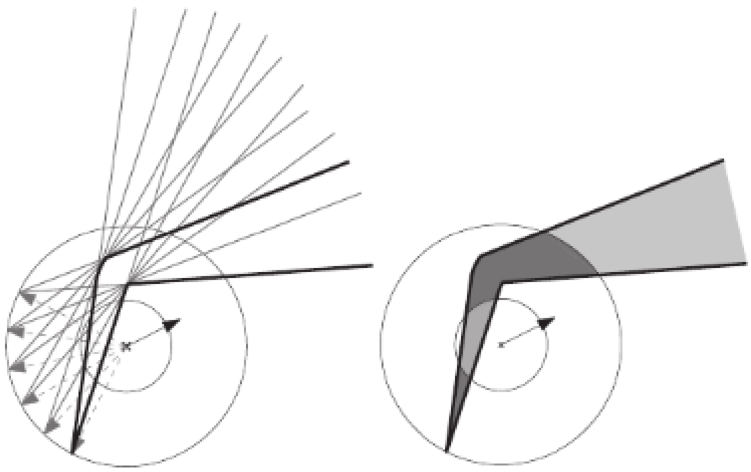


Figure 2.11: Constructing the tangent-based SSD, showing the effect of a turn. From [9]

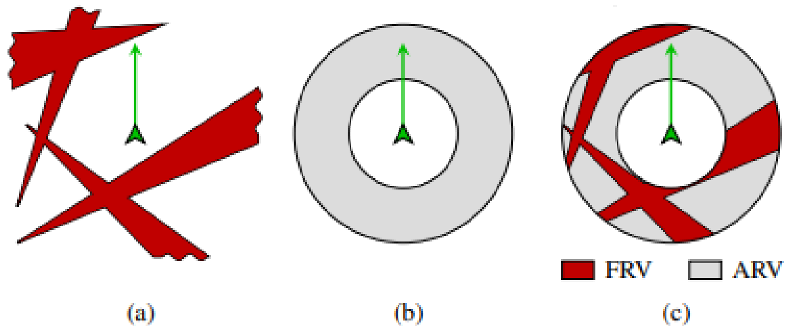


Figure 2.12: The construction of the SSD in (c) by combining the sets of FV in (a) and the RV in (b) from [7]

Rules are then set in place to coordinate between the aircraft in conflict. An overview of the rules is given in Table 2.5

Table 2.5: Coordination Rulesets with labels and descriptions from [7]

Label	Priority	Description
1. OPT	x	Resolve by taking the shortest way out.
2. RIGHT	x	Resolve by only turning right.
3. HDG	x	Resolve by only changing heading.
4. SPD	x	Resolve by only changing speed.
5. DEST	x	Resolve towards the target heading.
6. ROTA	□	Resolve by adhering to the rules of the air.
7. OPT+	□	Resolve sequentially while adhering to OPT.
8. DEST+	□	Resolve sequentially while adhering to DEST.

Balasooriyan concluded that resolving conflict by taking the shortest way out resulted in the best efficiency for the SSD conflict resolution method. However, the method itself performed worse than the Modified Voltage Potential method it was compared to.

2.3.6. Modified Voltage Potential

The Modified Voltage Potential (MVP) algorithm is a conflict resolution method which has shown promising results [19]. The algorithm takes the idea from two electrically charged particles repelling each other and applies it to conflicts.

The algorithm described by Hoekstra works by predicting the CPA of the intruder and the ownship [18]. The minimum distance vector then points from the ownship to the intruder. The avoidance vector is found by taking a vector, parallel to the minimum distance vector and with a length from the CPA to the edge of the protected zone. The process is visually depicted in Figure 2.13.

The ownship also creates an avoidance vector with regards to the intruder using the same method. The result is an avoidance vector which is opposite to the avoidance maneuver from the intruder. This opposite action is what is meant by the charged particle analogy and results in implicit coordination between the two aircraft.

In multi-aircraft conflicts, MVP makes use of pairwise summing where the avoidance vectors are summed into one vector. This allows the algorithm to perform better at de-conflicting multi-aircraft conflicts compared to other algorithms [7] [19] and provide an effective global solution with a high efficiency.

2.3.7. Conclusion

To conclude, a lot of research has been performed on CD&R methods. The most often used, and best performing methods have been listed. There is one conflict detection method in particular that is often used and efficient and that is the state-based conflict detection method. This method will be considered as baseline method.

Moreover, it is found that a lot of conflict resolution methods use VOs. These are however, not really applicable to an urban airspace due to the restrictions imposed by flying over the street network,

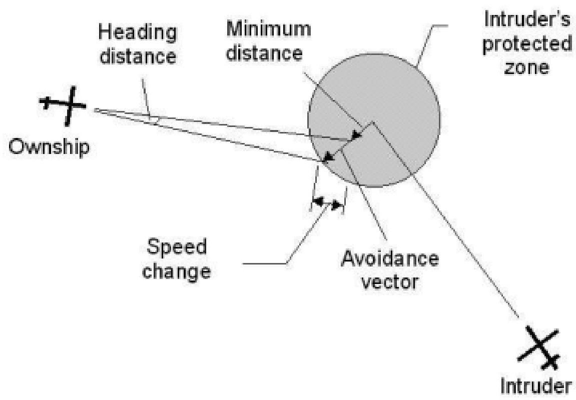


Figure 2.13: Geometry of modified voltage potential resolution method

restricting the heading changes that the drones can perform leaving only velocity changes as the only option.

2.4. Intent

First, it is important to know what intent is. Intent can be defined as "the flight path and associated flight data describing the planned trajectory of a flight to its destination, as updated at any moment" [8]. The use of intent in manned aviation has been tried in numerous times [25], [45], [22]. Research has moved to intent for unmanned aviation with the more recent increase in popularity of drones for civil use. In this section, a few methods are discussed where intent is used for CD&R purposes for unmanned aircraft.

2.4.1. The Use of Intent Information in Conflict Detection and Resolution Models Based on Dynamic Velocity Obstacles

In this paper a method is discussed which incorporates intent into velocity obstacle theory. In this subsection, the proposed method by Mercado Velasco et al. is discussed [28].

A key concept to understand in the proposed method is the use of time to collision. This time varies depending on the velocity vector along the bisector of the VO. To calculate the velocity required for a collision between intruder A and ownship B at time t_c the following equation is used.

$$\mathbf{v}_A\left(\mathbf{p}_A\left(t_c\right)=\mathbf{p}_B\left(t_c\right)\right)=\frac{\mathbf{p}_B\left(t_c\right)-\mathbf{p}_A\left(t_0\right)}{t_c-t_0}=\frac{\mathbf{d}\left(t_c\right)}{t_c-t_0}=\mathbf{v_c}(t_c) \tag{2.5}$$

Furthermore, it follows from this equation that for any position $\mathbf{P}_i$ inside the protected zone exists a set of velocity vectors that reaches that position at $t=t_c$. This set of velocity vectors also defines a circle with its center at $\mathbf{v_c}(t_c)$. The radius of each circle can be determined from geometric relations and is displayed in Equation 2.6 with R being the radius of the protected zone. The result is then a family of circular curves which decrease in radius over a longer time.

$$r\left(t_c\right)=R\cdot\frac{|\mathbf{v}_c\left(t_c\right)|}{|\mathbf{d}\left(t_c\right)|}=\frac{R}{t_c-t_0} \tag{2.6}$$

An envelope is developed which contains the family of circles and is tangent to them. For aircraft with no changes in state, the result is the regular velocity obstacle cone. For moving objects it is a different case. First the family of circles is expressed in Cartesian coordinates where θ is the angular coordinate along a circle.

$$\begin{bmatrix} v_x \\ v_y \end{bmatrix}=v_c\left(t_c\right)+r_c\left(t_c\right)\begin{bmatrix} \cos(\theta) \\ \sin(\theta) \end{bmatrix}, \quad \forall\theta\in[-\pi,\pi],\quad t_c\in[t_c,\infty], \tag{2.7}$$

The envelope can then be described as the following. Making use of the substitutions in Equation 2.9, the partial differential equation is obtained in Equation 2.10.

$$\begin{vmatrix} \frac{\partial v_x}{\partial t_c} & \frac{\partial v_x}{\partial \theta} \\ \frac{\partial v_y}{\partial t_c} & \frac{\partial v_y}{\partial \theta} \end{vmatrix} = 0 \tag{2.8}$$

$$\dot{v}_{c_x} = \frac{\partial V_{c_x}}{\partial t_c}, \quad \dot{v}_{c_y} = \frac{\partial V_{c_y}}{\partial t_c}, \quad \dot{r} = \frac{\mathrm{d}r}{\mathrm{d}t_c} = \frac{-R}{(t_c - t_0)^2} \tag{2.9}$$

$$\dot{v}_{c_x} \cos\theta + \dot{v}_{c_y} \sin\theta + \dot{r} = 0 \tag{2.10}$$

By further substituting $\Theta = tan(\theta/2)$ and solving the resulting second order polynomial the following results are obtained.

$$\theta = 2arctan(\Theta) \tag{2.11}$$

$$\Theta = \frac{-\dot{v}_{c_y} \pm \sqrt{|\dot{\mathbf{v}}_c|^2 - \dot{r}^2}}{-\dot{v}_{c_x} + \dot{r}} \tag{2.12}$$

The original VO can then be traced, making use of Equation 2.7, Equation 2.11 and Equation 2.12. This is valid for all except $|\dot{\mathbf{v}}_c| \geq \dot{r}$.

2.4.2. The Effect of Intent on Conflict Detection and Resolution at High Traffic Densities

Ribeiro, Ellerbroek, and Hoekstra argue that state projection into the future is necessary to prevent very short-term conflicts [34]. This is especially trivial for unmanned aircraft, due to the expected high density traffic in an urban environment. For this, Ribeiro, Ellerbroek, and Hoekstra have developed a conflict detection and resolution algorithm using intent and state.

The CD&R algorithm makes use of trajectory change points (TCPs). By using TCPs, false positives and false negatives are removed. False positives are removed as LoSs after the intruder's TCP can be neglected as well as LoSs after the ownship's TCP. False negatives are removed by incorporating the expected LoSs after the intruder's and ownship's TCP.

Intent is Incorporated in the conflict detection by dividing the trajectory into leg segments. Each leg is a segment between two TCPs. For each leg the CPA is calculated and if a conflict is detected it is first looked at whether the conflict occurs before or after the expected time of starting the next leg. If the conflict occurs after the start of the next leg, the conflict is deemed a false positive and the conflict is removed. If the conflict happens before the start of the next leg, the aircraft performs a conflict resolution.

The conflict resolution is done through using VO with intent as explained in the previous subsection. A solution space diagram is then created for the VOs to solve the conflicts. The paper discusses two ways to solve the conflict. The first one is the shortest way out, where the smallest change in velocity vector is chosen to avoid conflict. This has as benefit that two aircraft will always make an opposite maneuver. The second option is the shortest from destination, where the velocity vector is chosen that results in the least deviation from the trajectory and destination. This does not necessarily result in opposite maneuvers.

Furthermore, the experiment is done using just state CD&R, state and intent CD&R and state or intent CD&R. State and intent will detect and resolve conflicts for both situations, whether the aircraft is following intent or not. This results in a very crowded SSD as both VOs are used. State or intent makes use of only intent when flying its normal trajectory. When it deviates from its normal trajectory because of a conflict it will switch to state CD&R.

The paper concludes that the shortest way out is the best conflict resolution strategy in this case as it resulted in lower chain reaction conflicts. Furthermore, a lack of implicit coordination makes the shortest way out strategy safer. The use of both state and intent resulted in the least LoSs and is hence deemed best. This is due to being able to catch drones who have ”missed” their trajectory change point.

2.4.3. Unifying Tactical Conflict Prevention, Detection, and Resolution Methods in Non-Orthogonal Constrained Urban Airspace

In the paper by Badea et al. a lot of concepts are discussed regarding U-space like for instance layering and directionality of roads [6]. Also intent is discussed within this paper in the form of a projection-based conflict detection method.

The conflict detection method that is used in the paper is a state-based method due to its robustness. However, intent is added to the state-based method by exchanging trajectories and scanning them for potential intersecting trajectories through a geometry-based search. For each detected trajectory intersection, further investigation is required to see if they are actual conflicts. The states of the aircraft are taken as well as the distance towards the intersection. A state-based conflict detection is then performed on a projected linear line with the distance as length, and the current drone states are used. A visualization of this can be seen in Figure 2.14. If there is a LoS, the conflict is resolved. Inaccuracies occur with this method when turns are involved as these change the state of an aircraft. The experiment that is done in the paper compares a purely state-based method, a purely projection-based method and a method where both are used simultaneously.

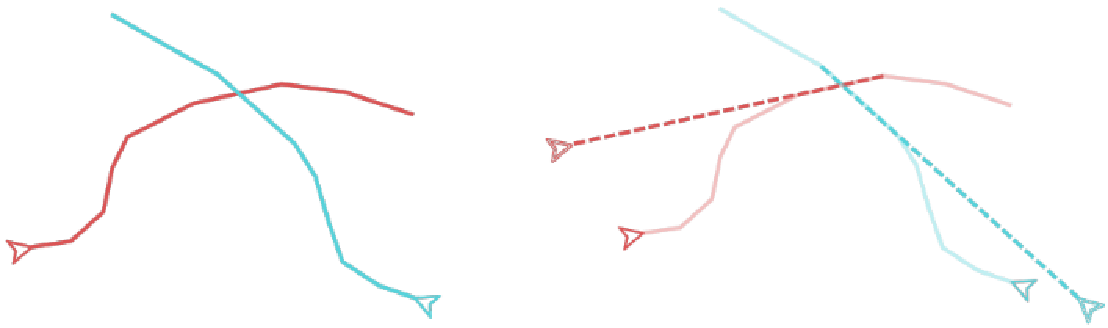


Figure 2.14: Projection-based conflict detection method: if intersecting paths are detected, the intruder is projected linearly from the intersection point, and a linear detection algorithm is applied: (a) Intersecting intent information is detected between two aircraft in constrained airspace; (b) Aircraft states are projected linearly from the intersection point (dashed lines) such that a state-based conflict detection method can be applied. From [6]

The conflict resolution is tried in two different ways, by resolving using only velocity changes and resolving using only heading changes. A rule-based algorithm is set-up to ensure separation between aircraft. This rule-based algorithm for velocity is displayed in Figure 2.15. A similar rule-based algorithm is set up for heading changes. A potential benefit that heading-only has over velocity-only, is the fact that aircraft do not have to slow down, or be stuck between slower aircraft creating repeat conflicts.

Algorithm 1 Conflict resolution algorithm of the decentralised concept of operations

```
for each intruder of this ownship do
    if intruder is behind then:                              ▷ ownship does not perform an action
        return Continue cruising
    else if intruder is in front then:
        if ownship can ascend then:                          ▷ ownship can ascend to next cruise layer
            return Ascent command                            ▷ ownship overtakes
        else                                                 ▷ ownship cannot overtake
            return Speed-based CR command
    else if intruder is directly above or below then:
        set ownship and intruder vertical speed to 0         ▷ stop vertical manoeuvring
        if ownship has priority then:                        ▷ ownship continues cruising
            return Maintain altitude, continue cruising
        else                                                 ▷ intruder has priority
            return Slow down to let intruder merge
    else                                                     ▷ intruder is coming from the side
        if ownship has priority then:                        ▷ ownship does not perform an action
            return Continue cruising
        else                                                 ▷ intruder has priority
            return Speed-based CR command
```

Figure 2.15: Velocity obstacle methods calculate the required change (Δv) in relative speed between two aircraft (v_{rel}) in order for the minimum separation distance (defined by the radius of the protection zone R_{pz}) to not be breached. From [6]

The paper concludes that using both detection methods simultaneously increases the performance of CD&R. Moreover, it is found that false positives provide a stabilizing effect as it results in aircraft with false positives slowing down, usually in areas with a high traffic density, which in turn helps detecting and preventing conflicts. A downside of the projection-based method is the inaccuracy of the detection due to turns. This resulted in the projection-based method detecting conflicts very late when the intent information is accurate. The heading-only and velocity-only resolution maneuvers performed similarly to each other.

2.4.4. Conclusion

Intent has been applied in multiple cases, each in their own way. First of all, intent in an urban environment has so far only been used in the paper by Badea et al. The other papers do use intent but are not used in an urban environment where constrained areas play a huge role in the movement ability of the drone. Hence, the paper by Badea et al. will be the best paper to move forward from.

Some potential improvements to this paper could be to make a full intent-based method instead of a state-based method which uses intent. By removing the state-based portion, the accuracy of the prediction can be potentially improved, leading to a safer and more efficient travel in urban airspace. A great starting point would be tackling the issue of turns, making the intent-state-based detection inaccurate.

2.5. Conclusion

In this literature review a lot of information related to the topic at hand has been discussed. Knowledge gaps are identified and a research direction can be distinguished.

From the literature review on airspace it is found that the layered system provide the best combination of safety and efficiency. Furthermore one-way streets has a better safety performance compared to two-way streets whilst having similar performances on efficiency. A knowledge gap that is found is the use of turning layers and the effects of removing them from the layering system.

From the literature review on CD&R methods it is found that state-based conflict detection is an efficient detection method which can be used as baseline for the experiment.

The literature review of intent, found that the usage of intent in constrained airspace is not that common. A state-based method using intent is used in the paper by Badea et al. and may be a good starting ground for the development of an intent-based conflict detection method. The paper already stated that turns make the results inaccurate. Making the predictions more accurate through taking into account turns and other trajectory features can be explored.

3

Experimental Setup

In this chapter, the experimental setup is described. Firstly the framework of the experiment is given in Figure 3.1. Secondly, background information is provided in section 3.2. A list of assumptions is shown in section 3.3. The methodology is elaborated on in section 3.4. And lastly, the experiment setup is described in section 3.5

3.1. Framework

To illustrate the steps that are taken to get to the experiment and its results, a framework is created. This framework is displayed in Figure 3.1.

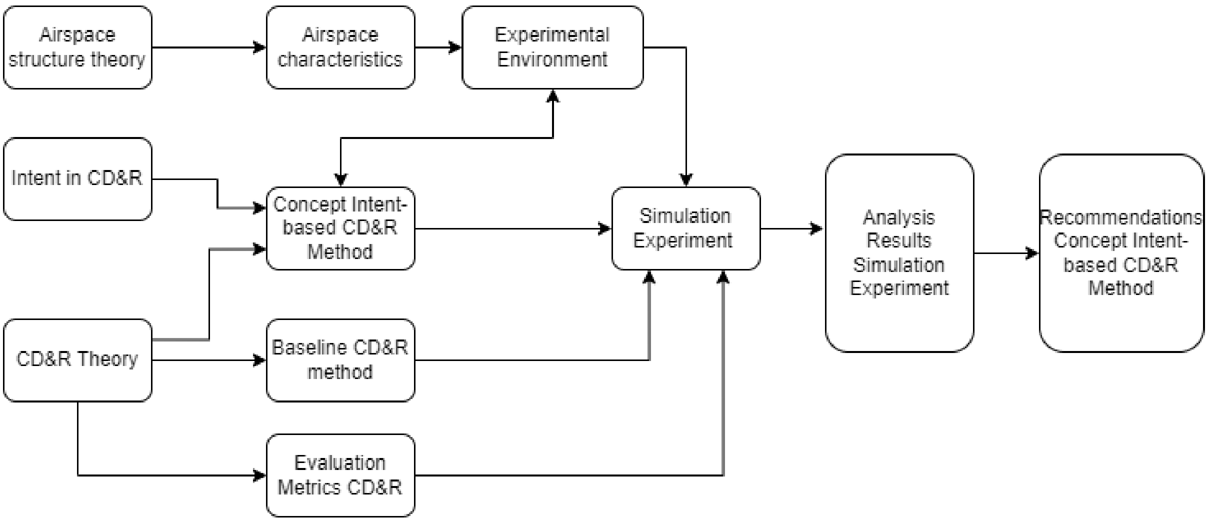


Figure 3.1: Research Framework

3.2. Background

In this section, background information is given regarding the experiment. The information provided is relevant to understand the experiment fully.

BlueSky Open Air Traffic Simulator

The BlueSky ATC Simulator Project is an initiative that focuses on the development of an air traffic control (ATC) simulator using an open data and open-source approach [20]. The project aims to create a realistic and accessible simulation environment for training and research purposes in the field of air traffic control. The simulator is built using Python 3 and uses opensource data for its navaids, aircraft performance and geography.

OSMnx

OSMnx is a python package which allows geospatial data, such as street networks to be downloaded. The data is obtained from OpenStreetMap which is an open source database [30]. Moreover, the package also includes built-in functions which are useful for navigating the street network, such as finding the shortest route to a destination.

Mission Types

From literature, in general two different types of missions can be distinguished. These are hub and spoke missions [11], [3], [6] and point-to-point mission [6], [31].

Hub to spoke mission are missions which start at a vertiport and travel to a random position in the city. This application is similar to how a drone would deliver items from a centralized distribution center to its customers.

Point-to-point missions are missions which start at a random point and end at a random point. These mission's real life applications are for instance mail delivery.

Drone Behaviour

The focus of drone operations in urban airspace mostly revolves around delivery. For this purpose, mostly multicopter drones have been used in past papers, thanks to their agility and maneuverability [11], [6], [5], [32].

It is expected that drones will be flying above the streets [11]. To navigate through these streets a turn procedure is required to ensure safety and consistency among drones. Some considerations need to be taken into account.

Firstly, the turn should be taken at a slower speed compared to cruising for safety. Doole et al. found that 50% slower, so a speed of $5m/s$ would make sure that the drone would not exceed its trajectory. In addition, a turn is made with a certain bank angle. This can be set to a fixed maximum bank angle such as $25°$ derived from the maximum pitch angle of the DJI Matrice 600. With these two variables known, the turn radius can be calculated and the drone then knows when to take the turn. The turn procedure is then as follows.

As the drone approaches the turn it will slow down from cruise speed to turn speed. The drone then takes the turn in a circular motion, and once it has the correct heading, leaves the circle and accelerates once again to cruising speed using a constant acceleration.

3.3. List of Assumptions

The following list of assumptions is made for this experiment. The assumptions can be categorized as movement assumptions and operational assumption.

Movement assumptions

- The drone accelerates vertically and horizontally with a constant acceleration.
- The drone decelerates vertically and horizontally with a constant deceleration.
- During a turn the turn is taken at a constant turning speed.
- The trajectory of the turn is described as a circle.
- The turn is made utilizing a fixed bank-angle.
- Drones behave perfectly and do not make mistakes.

Operational assumptions

- The conflict detection is executed at the same time for all drones.
- Drones are able to exchange information of their trajectory instantly with each other.
- No wind or other disturbances are taken into account.
- The vertical separation minima is 25 feet.
- The horizontal separation minima is 32 meters.

- The missions are flown point-to-point.
- Drones will navigate over the streets and will stay within the confines of the street network.

3.4. Methodology

In this section the methodology of the intent-based CD&R algorithm is described. subsection 3.4.1 goes over the airspace design. In subsection 3.4.2, the conflict detection algorithm is discussed. The trajectory calculations are demonstrated in subsection 3.4.3. Lastly, the conflict resolution algorithm is described in subsection 3.4.4.

3.4.1. Airspace design

The CD&R will be performed in a layered airspace with one-way directionality. The airspace lay-out will be similar to the one by Badea et al. discussed in subsection 3.4.1.However, changes to the design are made.

First of all, as concluded from the paper, turn layers have a destabilizing effect on traffic and safety was enhanced after removing the necessity to merge into the turn layer. Hence, it is opted to remove the turn layers as a whole and creating a zig-zag pattern as illustrated in Figure 3.2.

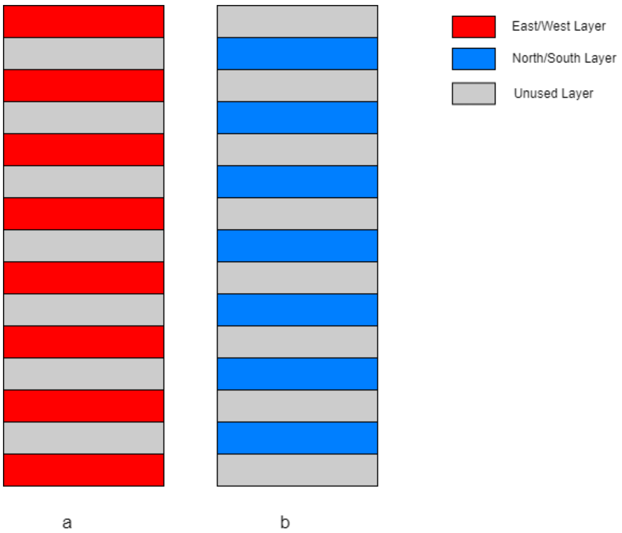


Figure 3.2: The layer configurations used within constrained airspace. Each layer is 30 ft in height. Grey are unused layers. (a) Red are layers reserved for travelling in east/west direction. (b) Blue are layers reserved for travelling in North/South direction.

The idea behind this system is that it creates simpler conflict at complex intersection and traffic can merge back into the cruising layer after the intersection when the situation is less complex.

Lastly, similar to the original concept, all new aircraft are assigned to the bottom layers and the upper layers are used for conflict resolution purposes.

3.4.2. Conflict Detection

For the intent-based conflict detection method it is assumed that the drones can communicate with each other and are able to exchange data regarding their trajectory. Moreover, it is required that the drones perform their conflict detection and trajectory extrapolation at the same time.

For each drone the trajectory is extrapolated in three dimensions based on a look-ahead time of 10.5 seconds. The horizontal position is then calculated as well as indicated in which turn layer the drone will be at for a measurement resolution of 3.5 seconds.

The resolution has been calculated for the worst case scenario where a conflict might occur undetected. The worst case scenario occurs when one drone is flying at a speed of 0 m/s and a drone flies towards it at a speed of 18 m/s. This is the worst case scenario as a head-on collision is not possible due to the one-way directionality of the streets. To still being able to detect a conflict in this scenario, the resolution must be small enough such that the intruder is not able to cross the protected zone of the ownship undetected.

Given a horizontal separation minima radius of 32 meters, to fully cross the zone a distance of 64 meters is crossed. Given a cruising speed of 18 m/s the following calculation can be performed to find the measurement resolution

$$t = \frac{64}{18} \approx 3.5s \tag{3.1}$$

The look-ahead time of 10.5 seconds is chosen based on the look-ahead time in [6] and slightly adjusted to fit the measurement resolution of 3.5 seconds.

The position calculations are done by considering the trajectory as individual pieces like turns and straights, each with their own characteristics. Based on these characteristics, the distance traveled within the measurement resolution can be calculated and the horizontal position found. Similarly, the vertical position can also be found. A measurement sample would then look like this. An example of a horizontal conflict is displayed in Figure 3.3.

$$data : [[X_0, Y_0, Z_0], [X_1, Y_1, Z_1], [X_2, Y_2, Z_2], [X_3, Y_3, Z_3], ..., [X_n, Y_n, Z_n]] \tag{3.2}$$

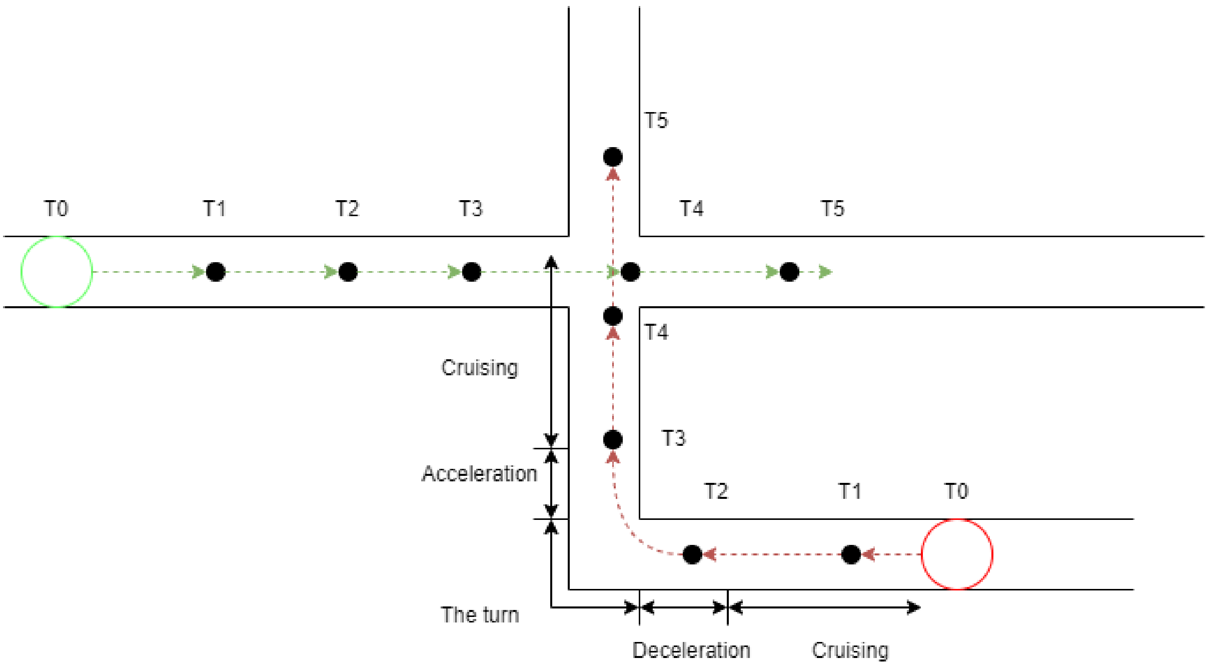


Figure 3.3: A horizontal conflict scenario of drone 1 (green) and drone 2 (red). As can be seen the trajectory of drone 2 can be divided up into characteristic sections and predict the position of the drone for each measurement. A potential conflict is detected at measurement T4 as the drones are located to close to each other.

For each measurement, the altitudes are compared of two aircraft. If for one or more measurements, the difference between the altitudes is smaller than the vertical separation minima, the horizontal distance of those measurement are calculated. If the horizontal distance is smaller than the horizontal separation minima, a conflict is detected.

3.4.3. Trajectory calculation

The total trajectory is the route that the drone will be following to get from its origin to its destination. The drone will only be looking ahead with a look-ahead time of 10.5 seconds. It will calculate its own position with a resolution of 3.5 seconds along its trajectory. To calculate its own position, the following cases are assumed:

1. The drone accelerates from motionless to cruising speed

2. The drone travels at cruising speed
3. The drone decelerates from cruising speed to turning speed
4. The drone takes a turn
5. The drone accelerates from turning speed to cruise speed
6. The drone decelerates from cruise speed to motionless
7. The drone changes altitude

Assumptions are made regarding the movement of drones as well as a turn procedure. With these assumptions we can accurately calculate the position the drone will be after a certain time. The equations that are used for each case are as follows. For Case 1, the position and time it takes is calculated simply using Equation 3.3

$$s = \frac{1}{2}a \cdot t^2 \ , \ t = \frac{V_{cr}}{a} \tag{3.3}$$

For Case 2, the distance and time can be simply found using Equation 3.4.

$$s = V_{cr} \cdot t, t = \frac{s}{V_{cr}} \tag{3.4}$$

Case 3 is similar to Case 1 with a slight difference.

$$s = V_{cr} \cdot t - \frac{1}{2}a \cdot t^2 \ , \ t = \frac{V_{cr} - V_{turn}}{a} \tag{3.5}$$

Case 4 is a bit more complex compared to the other cases. The first issue is that the turn radius should be found. This is done using Equation 3.6. For this equation the bank-angle is required as well as the turning velocity. These are both fixed with a bank-angle of $\phi = 25°$. Moreover, the turn velocity is also fixed as stated in subsection 3.5.1 with a value of $5m/s$. Through this equation the turn radius is found.

The last variable that is required for the distance and time calculations is $\Delta\Psi$, which is the change in heading. This variable is dependent on the turn that is made. The distance travelled during the turn and the time required to make the turn can then be derived.

$$r = \frac{V_{turn}^2}{gtan(\phi)} \tag{3.6}$$

$$s = 2\pi r \frac{\Delta\Psi}{360} = \frac{\pi r \Delta\Psi}{180} = \frac{\pi V_{turn}^2 \Delta\beta}{180 gtan(\phi)} \tag{3.7}$$

$$t = \frac{s}{V_{turn}} = \frac{\pi V_{turn}^2 \Delta\Psi}{180 gtan(\phi) V_{turn}} = \frac{\Delta\Psi \pi V_{turn}}{180 gtan(\phi)} \tag{3.8}$$

Case 5 and 6 are similar to case 1 and 3 and can be described by Equation 3.9 and Equation 3.10 respectively.

$$s = V_{turn} \cdot t + \frac{1}{2}a \cdot t^2, t = \frac{V_{cr} - V_{turn}}{a} \tag{3.9}$$

$$s = V_{cr} \cdot t - \frac{1}{2}a \cdot t^2, t = \frac{V_{cr}}{a} \tag{3.10}$$

Case 7 occurs whenever the drone needs to merge or resolve a conflict. The vertical acceleration and deceleration are similar to the horizontal acceleration and deceleration. Furthermore, it is known that to change layers a distance of 30 ft needs to be covered. Since the total vertical velocity needs to be 0 again at the end of the maneuvre, the equation can be split up in two parts. The acceleration and deceleration. The distance over time and total time taken for the acceleration can be found using Equation 3.11.

$$\Delta h = \frac{1}{2} a \cdot t^2, t = \sqrt{\frac{h_{layer}}{a}} \tag{3.11}$$

For the deceleration the end velocity of the first acceleration part needs to be found. This is done using Equation 3.12. Then finally the distance and time can be found using Equation 3.13.

$$V_{vertical} = a \cdot \sqrt{\frac{h_{layer}}{a}} \tag{3.12}$$

$$\Delta h = V_{vertical} - \frac{1}{2} a \cdot t^2, t = \sqrt{\frac{h_{layer}}{a}} \tag{3.13}$$

Depending on whether the drone is going up or down, the signs are switched of the equations.

3.4.4. Conflict Resolution

Once a conflict is detected, the drone goes into conflict resolution. A rule-based conflict resolution method is set-up using only altitude and velocity. Heading changes are not used as in constrained airspace, the aircraft have to follow the street network, which means there is not a lot of room for lateral maneuvering. The rule-based method is as follows. This conflict resolution method is able to take care of all scenarios where a breach of separation may occur.

[style = mystyle] for each detected conflict:
if intruder is behind then: continue trajectory
else if intruder is in front then: if ownship can ascend then: ascend to next cruising layer
if else ownship can descend then: descend to next cruising layer
else: compare distance between ownship-intruder and protected zone radius lower speed until ownship is outside protected zone radius + 10return to cruise speed

Figure 3.4: The conflict resolution algorithm

It is preferred for an intruder to first try to change cruising lane before altering speed. This is to reduce the amount of delay that the aircraft may encounter.

To see whether an intruder is in front of behind the ownship the predicted trajectory is looked at. If the ownship's predicted trajectory passes in front of the predicted position of the intruder, then then intruder is behind the ownship. If the ownship's predicted trajectory passes behind the prediced position of the intruder, then the intruder is in front of the ownship. Since in a conflict there is always one drone in front and one drone behind this means there is implicit coordination between the drones.

3.5. Experiment Setup

In this section, the experiment setup is described. In subsection 3.5.1 the details of the simulation are described. subsection 3.5.2 and subsection 3.5.3 presents the independent and dependent variables. Lastly, hypotheses are made for the experiment in subsection 3.5.4.

3.5.1. Simulation

Simulation Environment

The simulation is performed using BlueSky Open Air Traffic Simulator in combination with OSMnx. The CD&R method as well as the urban airspace design will be implemented in BlueSky. The street network will be generated using the OSMnx package. Using these two resources, the simulations can then be performed.

For the street network, Manhattan, New York City,has been chosen due to its orthogonal streets. The street network of Manhattan encourages the use of intent as it involves a lot points where a drone can change trajectory thanks to its grid design.

The missions that are flown are point-to-point and the origin and destination are selected randomly. It has been opted to use this mission type as it spreads out traffic over the whole simulation area.

In addition, point-to-point creates sporadic movements within the city, putting a bigger focus on the capabilities of conflict detection.

Drone behaviour

Two different drone types are chosen for this experiment. Both of these drones are based on the DJI 600 Matrice multicopter as this particular drone has been used often in other papers and has specifications suitable for delivery. The only difference between the two drone types are the cruise speed. The specifications are displayed in Table 3.1.

Table 3.1: Specification drones

Aircraft Name	MP20	MP30
Max horizontal Speed (m/s)	13	18
Average cruise speed (m/s)	10	15
Max vertical Speed (m/s)	5	5
Max take-off mass (kg)	15	15
Max acceleration/deceleration (m/s^2)	3.5	4.5
Turn Speed (m/s)	5	5

The constant acceleration and deceleration of the drones are set to be the max acceleration and deceleration of the drone. Thus these are $3.5m/s$ and $4.5m/s$ for MP20 and MP30 respectively. Lastly, the drones follow a turn procedure as described in section 3.2

3.5.2. Independent Variables

Two different independent variables are used for the experiment. These are the conflict detection method and the traffic density.

Detection Method

The first independent variable is the detection method. There are in total, three different methods that are used.

- *State-based only*, this serves as a baseline and to compare the result of the intent-based detection methods to
- *Intent-based only*, the conflict detection method only detects conflicts using only the intent-based method.
- *Intent-based and State-based simultaneously*, the intent and state-based detection methods are used in parallel, and all detected conflicts are resolved.

Traffic density

Traffic density may have a considerate influence on the performance of the detection method and how well the conflict resolution can perform. As intent-based algorithm look further in the future, the amount of conflicts detected and the resulting resolutions might create more chaos if a high traffic density is present.

An estimation of traffic density is made based on [10] and has been adapted from [11]. The values of the levels of traffic density can be seen in Table 3.2

Table 3.2: Traffic density characteristics of the three demand scenarios for the simulation area of Manhattan, New York City, network consisting of an area of 59.1 km^2. From [11].

Variable	Low	Medium	High
Traffic density ($drones/km^2$)	55	61	73
Inflow rate ($drones/min$)	54	60	72
Hourly demand ($drones/h$)	3240	3600	4320

3.5.3. Dependent Variables

To evaluate the performance, the dependent variables are looked at. The variables that are going to be tracked are the following.

- *Number of conflicts detected*, measures the amount of conflicts that have been detected.
- *Number of intrusions*, measures the amount of LoSs that have occured.
- *Average time delay*, measures the amount of delay that an aircraft had on average compared to the shortest full route time.
- *Average detection time before conflict*, measures the amount of time between the detection of the conflict and the predicted occurrence of the conflict.
- *False positive rate*, measures the amount of incorrect conflicts detected.
- *False negative rate*, measures the amount of conflicts missed.
- *Time in conflict*, measures the time a drone spent in conflict with another drone before being resolved.

3.5.4. Hypotheses

Three different conflict detection methods are evaluated at different traffic densities. The following hypotheses are made based on the three traffic scenarios.

Hypothesis 1: Low traffic densities
It is expected that in low traffic densities, the purely intent-based method has the best performance followed by the combined intent-state-based method and lastly the purely state-based method having the lowest performance.

It is expected for low traffic densities, that the intent-based methods are able to quickly identify conflicts having more time to solve them. The intent-based method has an advantage over the combined intent-state-based method as both methods remove false negatives, but purely intent also gets rid of false positives. This results in a more efficient travel time. The purely state-based method is expected to have a disadvantage as it is not able to filter out false positives and false negatives as well.

Hypothesis 2: Medium traffic densities
It is expected that in medium traffic densities, the purely intent-based method has the best performance followed by the purely state-based method and the combined intent-state-based method having similar performances.

It is expected that in medium traffic densities, the purely intent-based performs best compared to the other methods due to it being able to filter out false positives and false negatives. The remaining two methods should have similar performances as they both have drawbacks which should be limiting their performances in medium traffic densities.

Hypothesis 3: High traffic densities
It is expected that in high traffic densities, the purely intent-based method has the best performance followed by the purely state-based method and lastly the combined intent-state-based method having the lowest performance.

The combined intent-state-based method is expected to perform the worst as with the combined detection methods, a lot more conflicts are detected compared to the other two methods. This means a lot of resolutions are performed which will lower the route efficiency drastically in high-traffic scenarios. Intent-based is expected to still have an edge over state-based as its removal of false positives and false negatives will result in the best route efficiency and the least number of conflicts.

4 Project Timeline

In this chapter, the project timeline is discussed along with the work that is done within the timeframe of the project. In section 4.1 the work packages are described for this project. In section 4.2 a gantt chart is displayed to show the timeline of the project.

4.1. Work packages

The project has been divided up into work packages with a goal. Each work package is then divided into workable tasks. There are a total of 10 work packages subdivided into 4 groups.

4.1.1. Airspace Design

In this group of work packages, the airspace design is made, this includes the street network as well as the generation of aircraft.

Work package 1: Street Network
The goal of work package 1 is to make a start with the experiment environment. The main focus of this work package is the street network. The goals of the work package are as follows.

- Generate the street network of Manhattan in BlueSky
- Assign the roads of the street network to North-South and East-West directionality.
- Make the roads uni-directional

Work package 2: Traffic Generator
The focus of this task is to create a traffic generator through which the specified traffic density can be generated. The tasks are as follows.

- Create randomized shortest point-to-point paths
- Assign path to drone
- Assign lowest cruise layer to generated traffic
- Generate path to destination and make the traffic follow this path
- Once destination is reached delete the aircraft
- Create number of aircraft to fullfill the traffic density goal

4.1.2. Conflict Detection

In this group of work packages, the conflict detection method is created in BlueSky.

Work package 3: Trajectory Calculation
The third work package focuses around the trajectory calculations. Each case needs to be applied correctly and recognized by the drones. The following tasks are assigned to this work package.

- Recognize, which case should be applied where.
- Apply the position calculations for each case.
- Apply the time calculations for each case.
- Verification by calculating the time and distance of trajectory and individual stretches and comparing it to an actual flown trajectory.

Work package 4: Trajectory Comparison
The focus of the fourth work package is to create the data that is exchanged between drones and process them according to conflict detection. The tasks are as follows.

- Record position of aircraft at the resolution frequency
- Make sure all aircraft do it at the same time
- Compare horizontal distance of aircraft and mark cases where a LoS occurs
- Compare vertical distance of aircraft and mark cases where a LoS occurs
- Run small pre-defined scenario to make sure everything works correctly

Work package 5: Conflict detection
The focus of this work package is to apply the trajectory comparison and make sure it is able to detect conflicts on its own. The tasks are as follows.

- Create a conflict detection method in Bluesky using the trajectory comparison and calculations
- Generate aircraft with a pre-defined conflict and check if the conflict is detected

4.1.3. Conflict resolution

In this work package the conflict resolution algorithm is created.

Work package 6: Conflict Resolution
The goal of this work package is to create a work conflict resolution algorithm. The tasks are as follows.

- Create the rule-based algorithm in BlueSky
- Test it to see if it works using pre-defined scenarios

4.1.4. Experiment

In this group of work packages everything related to the experiment is set up. This includes the tracking of dependent variables and analysis of the results.

Work package 7: Tracker
The goal of this work package is to create the tracker for the dependent variables in the experiment. The tasks are as follows:

- Create a tracker for number of conflicts detected
- Create a tracker for number of intrusions
- Create a tracker which can calculate the average time delay
- Create a tracker for the average detection time before conflict
- For each individual tracker test them in custom scenarios.

Work package 8: Small scale testing
In this work package everything is put together that has been created so far. A small test scenario is created to see if everything is working together properly. The task are as follows.

- Use the traffic generator to generate traffic but do not randomize the point to point
- Use the conflict detection to detect the conflict in a pre-defined traffic scenario
- Check the resolution of the conflict and see if it is performed as expected
- Check if the street network is followed correctly
- Check if the trackers record everything and correctly

Work package 9: Full scale experiment
In this work package the experiment is executed.

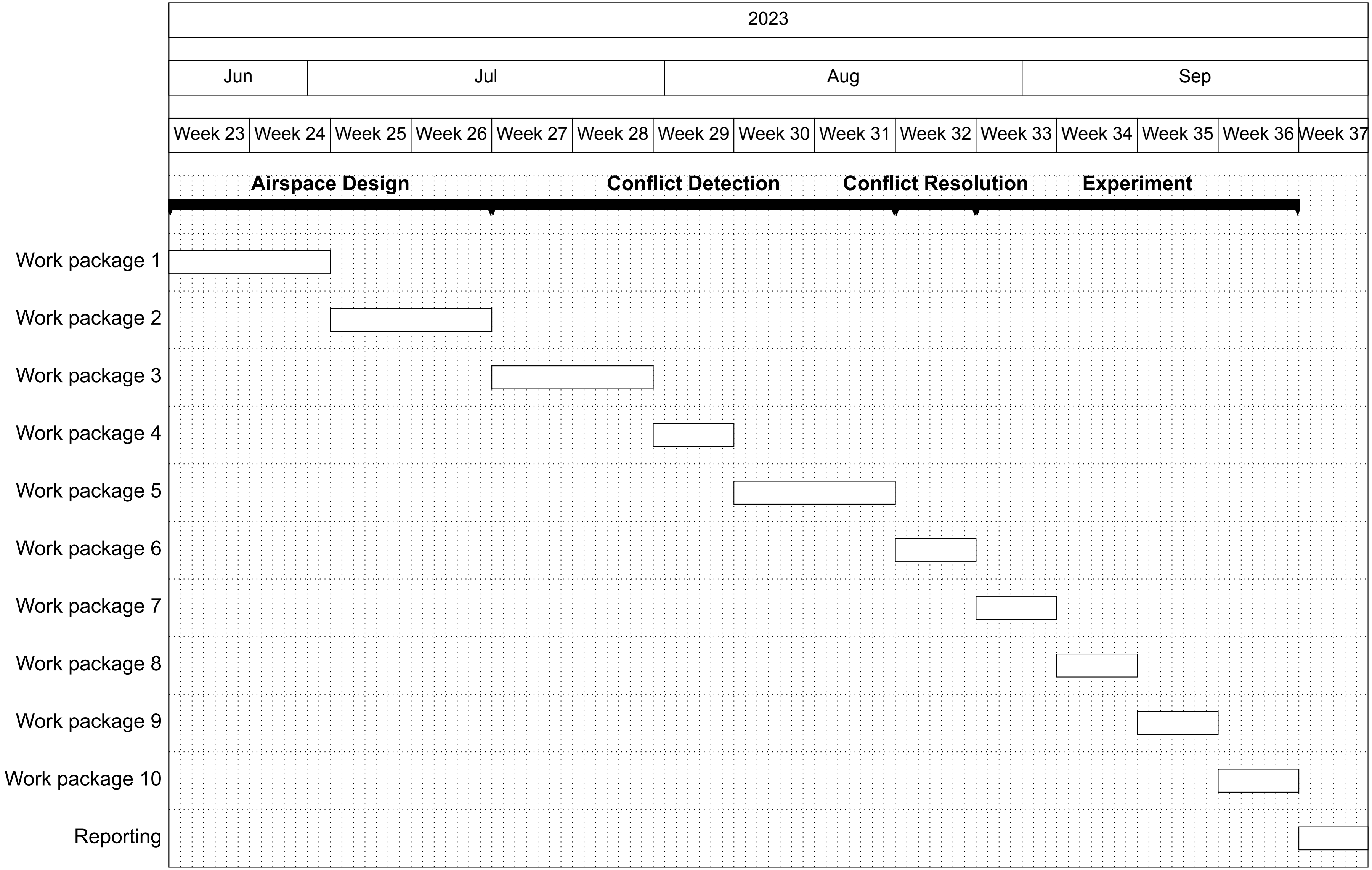
Work package 10: Analysis
After the full-scale experiment is completed, the data is analysed to evaluate the performance of each conflict detection method. From these analyses, the hypotheses can be concluded to be rejected or accepted. A final conclusion can then be made the research question answered.

4.1.5. Reporting

The final step of this project is then to start reporting all the work that has been done into a research article.

4.2. Gantt Chart

The Gantt chart provides an overview and duration of each work package. This provides a planning for the upcoming period.


2023
Jun
Jul
Aug
Sep
Week 23
Week 24
Week 25
Week 26
Week 27
Week 28
Week 29
Week 30
Week 31
Week 32
Week 33
Week 34
Week 35
Week 36
Week 37
Airspace Design
Conflict Detection
Conflict Resolution
Experiment
Work package 1
Work package 2
Work package 3
Work package 4
Work package 5
Work package 6
Work package 7
Work package 8
Work package 9
Work package 10
Reporting

Bibliography

[1] Niels A.H. Agatz, Paul Bouman, and Marie Schmidt. “Optimization Approaches for the Traveling Salesman Problem with Drone”. In: June 2017.

[2] One Air. *What is air space and how it is divided*. Dec. 2022. URL: `https://www.grupooneair.com/what-is-air-space/`.

[3] Amazon. “Amazon Prime Air prepares for drone deliveries”. In: *About Amazon* (). URL: `https://www.aboutamazon.com/news/transportation/amazon-prime-air-prepares-for-drone-deliveries` (visited on 06/09/2023).

[4] Emanuele Luigi de Angelis et al. “Toward Smart Air Mobility: Control System Design and Experimental Validation for an Unmanned Light Helicopter”. In: *Drones* 7.5 (2023). ISSN: 2504-446X. DOI: `10.3390/drones7050288`. URL: `https://www.mdpi.com/2504-446X/7/5/288`.

[5] Calin Andrei Badea, Joost Ellerbroek, and Jacco Hoekstra. “Conflict Prevention, Detection, and Resolution in Constrained Very Low-Level Urban Airspace”. In: June 2022.

[6] Calin Andrei Badea et al. “Unifying Tactical Conflict Prevention, Detection, and Resolution Methods in Non-Orthogonal Constrained Urban Airspace”. In: *Aerospace* 10 (Apr. 2023). DOI: `10.3390/aerospace10050423`.

[7] S Balasooriyan. “Multi-aircraft Conflict Resolution using Velocity Obstacles”. In: *Master’s Thesis Delft University of Technology* (Oct. 2017).

[8] *COMMISSION REGULATION No 2150/2005: laying down common rules for the flexible use of airspace*$_2$`005`. 2005.

[9] Jurriaan d’Engelbronner et al. “Solution-Space-Based Analysis of Dynamic Air Traffic Controller Workload”. In: *Journal of Aircraft* (Apr. 2015). DOI: `10.2514/1.C032847`.

[10] Malik Doole, Joost Ellerbroek, and Jacco Hoekstra. “Estimation of traffic density from drone-based delivery in very low level urban airspace”. In: *Journal of Air Transport Management* 88 (2020), p. 101862. ISSN: 0969-6997. DOI: `https://doi.org/10.1016/j.jairtraman.2020.101862`. URL: `https://www.sciencedirect.com/science/article/pii/S0969699719304004`.

[11] Malik Doole et al. “Constrained Urban Airspace Design for Large-Scale Drone-Based Delivery Traffic”. In: *Aerospace* 8.2 (Feb. 2021), p. 38. ISSN: 2226-4310. DOI: `10.3390/aerospace8020038`. URL: `http://dx.doi.org/10.3390/aerospace8020038`.

[12] FAA. *Airspace 101 – rules of the sky*. URL: `https://www.faa.gov/uas/getting_started/where_can_i_fly/airspace_101`.

[13] FAA. *Unmanned Aircraft System Traffic Management (UTM)*. Aug. 2022. URL: `https://www.faa.gov/uas/research_development/traffic_management`.

[14] FAA. *UTM pilot program (UPP)*. May 2022. URL: `https://www.faa.gov/uas/research_development/traffic_management/utm_pilot_program`.

[15] Paolo Fiorini and Zvi Shiller. “Motion planning in dynamic environments using velocity obstacles”. In: *The International Journal of Robotics Research* 17.7 (1998), pp. 760–772. DOI: `10.1177/027836499801700706`.

[16] David Gallacher. “Drones to manage the urban environment. Risks, rewards, alternatives.” In: 4 (Feb. 2016), pp. 115–124. DOI: `10.1139/juvs-2015-0040`.

[17] Stuart Granshaw. “RPV, UAV, UAS, RPAS … or just drone?” In: *The Photogrammetric Record* 33 (June 2018), pp. 160–170. DOI: `10.1111/phor.12244`.

[18] Jacco Hoekstra. “Designing for Safety: the Free Flight Air Traffic Management Concept””. In: *Ph.D Thesis Delft University of Technology* (2001).

[19] Jacco Hoekstra and Joost Ellerbroek. “Aerial Robotics: State-based Conflict Detection and Resolution (Detect and Avoid) in High Traffic Densities and Complexities”. In: *Current Robotics Reports* 2 (Sept. 2021). DOI: 10.1007/s43154-021-00061-6.

[20] Jacco Hoekstra and Joost Ellerbroek. “BlueSky ATC Simulator Project: an Open Data and Open Source Approach”. In: June 2016.

[21] Jacco Hoekstra, R.N.H.W Gent, and R.C.J Ruigrok. “Designing for safety: The ’free flight’ air traffic management concept”. In: *Reliability Engineering System Safety* 75 (Feb. 2002), pp. 215–232. DOI: 10.1016/S0951-8320(01)00096-5.

[22] Shifeng Huang et al. “Collision Detection for Cobots: A Back-Input Compensation Approach”. In: *IEEE/ASME Transactions on Mechatronics* 27.6 (2022), pp. 4951–4962. DOI: 10.1109/TMECH.2022.3169084.

[23] Yazdi I. Jenie et al. “Taxonomy of Conflict Detection and Resolution Approaches for Unmanned Aerial Vehicle in an Integrated Airspace”. In: *IEEE Transactions on Intelligent Transportation Systems* 18.3 (2017), pp. 558–567. DOI: 10.1109/TITS.2016.2580219.

[24] Parimal Kopardekar and Steve Bradford. *UAS Traffic Management (UTM): Research Transition Team (RTT) Plan*. Jan. 2017. URL: https://www.faa.gov/sites/faa.gov/files/2022-08/FAA_NASA_UAS_Traffic_Management_Research_Plan.pdf.

[25] Jimmy Krozel and Dominick Andrisani. “Intent Inference with Path Prediction”. In: *Journal of Guidance Control and Dynamics - J GUID CONTROL DYNAM* 29 (Mar. 2006), pp. 225–236. DOI: 10.2514/1.14348.

[26] J.K. Kuchar and L.C. Yang. “A review of conflict detection and resolution modeling methods”. In: *IEEE Transactions on Intelligent Transportation Systems* 1.4 (2000), pp. 179–189. DOI: 10.1109/6979.898217.

[27] Ang Li, Mark Hansen, and Bo Zou. “Traffic management and resource allocation for UAV-based parcel delivery in low-altitude urban space”. In: *Transportation Research Part C: Emerging Technologies* 143 (2022), p. 103808. ISSN: 0968-090X. DOI: https://doi.org/10.1016/j.trc.2022.103808. URL: https://www.sciencedirect.com/science/article/pii/S0968090X22002339.

[28] Gustavo Mercado Velasco et al. “The Use of Intent Information in Conflict Detection and Resolution Models Based on Dynamic Velocity Obstacles”. In: *IEEE Transactions on Intelligent Transportation Systems* 16 (Aug. 2015), pp. 1–6. DOI: 10.1109/TITS.2014.2376031.

[29] Andres Morfin Veytia et al. “Metropolis II: Benefits of Centralised Separation Management in High-Density Urban Airspace”. In: Dec. 2022.

[30] *OpenStreetMap*. https://www.openstreetmap.org/. Accessed: 2023-06-02.

[31] Roberto Pinto and Alexandra Lagorio. “Point-to-point drone-based delivery network design with intermediate charging stations”. In: *Transportation Research Part C: Emerging Technologies* 135 (2022), p. 103506. ISSN: 0968-090X. DOI: https://doi.org/10.1016/j.trc.2021.103506. URL: https://www.sciencedirect.com/science/article/pii/S0968090X21004927.

[32] Marta Ribeiro, Joost Ellerbroek, and Jacco Hoekstra. “Improvement of Conflict Detection and Resolution at High Densities Through Reinforcement Learning”. In: July 2020.

[33] Marta Ribeiro, Joost Ellerbroek, and Jacco Hoekstra. “Review of Conflict Resolution Methods for Manned and Unmanned Aviation”. In: 7 (June 2020). DOI: 10.3390/aerospace7060079.

[34] Marta Ribeiro, Joost Ellerbroek, and Jacco Hoekstra. “The Effect of Intent on Conflict Detection and Resolution at High Traffic Densities”. In: Sept. 2020.

[35] Ling-zhi Sang et al. “CPA Calculation Method based on AIS Position Prediction”. In: *Journal of Navigation* -1 (Apr. 2016), pp. 1–18. DOI: 10.1017/S0373463316000229.

[36] SESAR JU. *European ATM Master Plan: Roadmap for the safe integration of drones into all classes of airspace*. 2018.

[37] SESAR JU. *European Drones Outlook Study*. Nov. 2016. URL: https://www.sesarju.eu/sites/default/files/documents/reports/European_Drones_Outlook_Study_2016.pdf.

[38] SESAR JU. *European Drones Outlook Study: Unlocking the value for Europe*. 2018.

[39] SESAR JU. *U-space Blueprint*. 2017. DOI: 10.2829/335092.

[40] SKYbrary. *Classification of Airspace*. URL: https://www.skybrary.aero/articles/classification-airspace.

[41] SKYbrary. *Separation standards*. URL: https://www.skybrary.aero/articles/separation-standards.

[42] SKYbrary. *Unmanned Aerial Systems (UAS)*. URL: https://www.skybrary.aero/articles/unmanned-aerial-systems-uas.

[43] Emmanuel Sunil et al. "Metropolis: Relating Airspace Structure and Capacity for Extreme Traffic Densities". In: June 2015.

[44] Pratyush Tripathy et al. "An open-source tool to extract natural continuity and hierarchy of urban street networks". In: *Environment and Planning B: Urban Analytics and City Science* 48.8 (2021), pp. 2188–2205. DOI: 10.1177/2399808320967680. eprint: https://doi.org/10.1177/2399808320967680. URL: https://doi.org/10.1177/2399808320967680.

[45] Javier Yepes, Inseok Hwang, and Mario Rotea. "An Intent-Based Trajectory Prediction Algorithm for Air Traffic Control". In: vol. 1. Aug. 2005. ISBN: 978-1-62410-056-7. DOI: 10.2514/6.2005-5824.