Improving Safety of Vertical Manoeuvres in a Layered Airspace with Deep Reinforcement Learning

Groot, D.J.; Ribeiro, M.J.; Ellerbroek, J.; Hoekstra, J.M.

Publication date
2022

Document Version
Final published version

Published in
International Conference on Research in Air Transportation (ICRAT) 2022

Improving Safety of Vertical Manoeuvres in a Layered Airspace with Deep Reinforcement Learning

Jan Groot, Marta Ribeiro, Joost Ellerbroek and Jacco Hoekstra
Control and Simulation, Faculty of Aerospace Engineering
Delft University of Technology (TU Delft)
Delft, The Netherlands

***Abstract*—Current estimates show that the presence of unmanned aviation is likely to grow exponentially over the course of the next decades. Even with the more conservative estimates, these expected high traffic densities require a re-evaluation of the airspace structure to ensure safe and efficient operations. One structure that scored high on both the safety and efficiency metrics, as defined by the Metropolis project, is a layered airspace, where aircraft with an intended heading are assigned to a specific altitude layer. However, a problem arises once aircraft start to vertically traverse between these layers, leading to a large number of conflicts and intrusions. One way to potentially reduce the number of intrusions during these operations is by using conventional conflict resolution algorithms. These algorithms however have also been shown to lead to instabilities at higher traffic densities. As recent years have shown tremendous growth in the capabilities of Deep Reinforcement Learning, it is interesting to see how well these methods perform in the field of conflict resolution. This research investigates and compares the performance of multiple Soft Actor Critic models with the Modified Voltage Potential algorithm during vertical manoeuvres in a layered airspace. The final obtained performance of the trained models is comparable to that of the Modified Voltage Potential algorithm and in certain scenarios, the trained models even outperform the MVP algorithm. Overall, the results show that DRL can improve upon the current state of conflict resolution algorithms and provide new insight into the development of safe operations.**

***Keywords*—Conflict Detection and Resolution (CD&R), Deep Reinforcement Learning (DRL), Modified Voltage Potential (MVP), Unmanned Traffic Management (UTM), Self-Separation, BlueSky ATC Simulator**

I. Introduction

With the current rise in market demands for faster parcel delivery combined with the incentive to reduce the cost of delivery services, more and more companies have started researching the viability of using drones for these so-called last-mile delivery operations [1], [2]. Estimates for the drone delivery market in Paris range widely between 110 k–275 k drones operating per hour in the city by 2035 [3]. Even at the lowest estimates, this far surpasses the traffic densities of the current aviation standards. Thus, Federal Aviation Administration (FAA) and the International Civil Aviation Organisation (ICAO) have required drones to be capable of detect and avoid manoeuvres without the need of a human controller [4].

Constantly requiring avoidance manoeuvres, which likely move aircraft away from their nominal path, is inefficient and can lead to instabilities. Therefore, it is worthwhile to research the effect of airspace structures on the intrinsic safety of the airspace. The Metropolis project researched a variety of different structures differing in complexity and restrictions. It showed that a layered airspace, separating traffic vertically based on their heading, leads to a high intrinsic safety without (heavily) impacting the efficiency of the air traffic operations [5]. This increase in safety can be attributed to the separation and alignment effect [6]. The problem that arises, however, is that vertically manoeuvring aircraft do not benefit from this separation and alignment effect. This results in these vertical operations leading to a large increase in conflicts and intrusions, as demonstrated in different studies [7], [8].

One potential solution to this rise in conflicts would be to use conventional conflict resolution algorithms. However, at higher traffic densities, these algorithms might potentially lead to instabilities [9]. Conflict resolution at high traffic densities essentially is a multi-agent coordination problem. Deep Reinforcement Learning (DRL) has been shown to successfully learn how to operate in these environments by adapting to emergent behaviour that follows from these continuous interactions. Furthermore, studies have also shown that DRL can be used for lane changing and merging of cars in highway scenarios, which can be considered a 2D version of the layer transition problem [10]. Because of this, this research will investigate the capabilities of DRL for improving the safety of vertical manoeuvres in a layered airspace structure through direct control of the drones.

The DRL models will be trained for a variety of degrees of freedom in large scale simulations, simulating both package deliveries and take-offs. The performance of the final converged models will then be compared to the Modified Voltage Potential (MVP) conflict resolution algorithm to see if there is a benefit to using DRL over conventional methods. It is decided to use the MVP algorithm as previous research has shown that it is optimal at resolving conflicts whilst minimizing additional travel distance [11].

II. Problem Formulation

To compare the effectiveness of DRL and the MVP algorithm at resolving vertical conflicts and improving the overall safety of vertical manoeuvres, vertical operations will be simulated in the BlueSky Open Air Traffic Simulator [12]. In this simulation environment, drones will be tasked with either climb or descent commands to a specific target layer within this airspace. During these vertical manoeuvres, the goal of the model is to safely control the drone to the target layer while avoiding the other aircraft. From now on, individually controlled aircraft will be referred to as agents, whereas the model is used to define which policy is used by these individual agents.

The margin by which other aircraft must be avoided is based on a minimum horizontal and vertical separation. If any two aircraft are within these margins of each other, then an intrusion occurs. In this research a conflict between two aircraft means that the distance at the predicted closest point of approach between these aircraft is smaller than the required separation margins, indicating a potential future intrusion.

How these conflicts are resolved depends on which model the agent is based. For the MVP models, when the agent is in conflict, the shortest way out of the conflict is determined and translated to a set of actions that should be taken by the agent. The DRL based agents on the other hand are able to select actions at any time, independent of whether or not the agent is in conflict.

For this research the DRL and MVP models are further subdivided into different models based on the freedom they have in their actions, as it is currently unknown which set of actions will result in the optimal performance. In total three individual actions can be isolated: a change in vertical speed, a change in horizontal speed and a change in heading. These actions are combined to obtain the following 'sub-models':

- 'vs', control of the vertical speed only.
- 'v+vs', control of the vertical and horizontal speed.
- 'v+hdg', control of the horizontal speed and heading.
- 'full', or 3 degrees of freedom, control of all motions.

III. Methods

Here, the methods used in the experiments will be presented. First, the Markov Decision Processes (MDPs) are formulated to allow the usage of DRL methods. Then the employed DRL algorithm, Soft Actor-Critic, will be further elaborated. Finally, an overview of the used resolution baseline, MVP, will be given.

A. Markov Decision Process

To ensure that DRL can be used for the defined problem, this problem must first be formulated as an MDP. An MDP is a mathematical framework that can be used for decision making in systems with uncertainty. An important element of the MDP is the so-called Markov-property, which entails that the future states of the system should only be dependent on the current state of the system. For the scenario of conflict resolution with MVP this Markov-property holds, as for a specific conflict, the used resolution manoeuvre, and therefore future states, are independent of how these aircraft came to be in conflict. It is therefore assumed that this property also holds for DRL. This allows the problem to be formulated as an MDP, described by the quadruple (S,A,P,R): [13]

1) S, the state space of the system.
2) A, the action space of the system.
3) $P([s,a],s')$, the state transition function.
4) $R(s,a,s')$, the reward function.

The goal of the model is to learn which action $a \in A$ given a state $s \in S$ maximizes the total reward $\sum r \in R$ over all the state transitions $s,a \rightarrow s'$, where s' indicates the new state.

1) State: The state vector is a combination of the ownship states and the (relative) states of the intruders. Note that the number of aircraft in the vicinity of the ownship is variable, but the proposed method required the state representation to be constant in size. This means that the problem has to be converted to a partial observable MDP (POMDP). For this research, it is decided to include 5 aircraft in the state representation sorted by time until the closest point of approach (T_{cpa}) with a maximum distance at the closest point of approach (D_{cpa}) of 250m, which is 5 times the minimum horizontal separation (PZ_h) between 2 aircraft. This ignores aircraft that are moving away from the agent and only includes the aircraft with the smallest T_{cpa} in the state, e.g, the aircraft that require the most imminent action. An exception to this is made for aircraft that are in conflict, these are prioritized over other aircraft and are always included in the state, again sorted by T_{cpa}. All the horizontal states considered for state inclusion are given in Fig. 1. CPA stands for the closest point of approach, which is the point at minimum horizontal distance. Apart from this also the vertical distance, D_z, and relative vertical velocity, V_w, are considered.

For the ownship state, the height difference with the target layer (Δ_h), vertical speed (V_s), horizontal speed (V_{own}) and heading difference with the current layer (Δhdg_{layer}) are used for the state vector. The final state vectors for all the models are given in table I. Not all models are given the same state vector as it is assumed that a too-large state vector containing non-relevant information will negatively impact the required training time of the models.

Finally, all states are normalized using equation 1 before usage, which makes the distribution of all states have a zero mean and unit variance. This ensures that the initial weights for all states are of a similar magnitude. In this equation, μ_s refers to the mean value of this state and σ_s to the standard deviation. The values for σ_s and μ_s are determined by observing 100.000

state transitions. An exception for the normalization of the state vector is made for the conflict boolean parameter, which is kept as a boolean.

$$S = \frac{s_i - \mu_s}{\sigma_s} \tag{1}$$

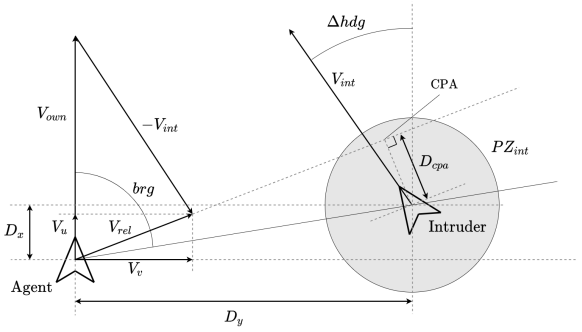


Figure 1. Visualization of the (horizontal) states related to an intruder.

TABLE I. The resulting state vector for the different experiments.

	vert	vert+	hor	full
Ownship States				
V_s	x	x	x	x
V_{own}		x	x	x
Δhdg_{layer}			x	x
Δ_h	x	x	x	x
Intruder States (x5) $\downarrow$				
T_{cpa}	x	x	x	x
D_{cpa}	x	x	x	x
Conflict with ownship (boolean)	x	x	x	x
D_z	x	x	x	x
D_x		x	x	x
D_y		x	x	x
V_u		x	x	x
V_v			x	x
brg			x	x
Δhdg_{int}			x	x

2) Action Space: For the action space, the allowable actions and their limits have to be defined. The allowable actions are dependent on the different models, defined in section II. The limits for the different actions are given in table II. In this table, the increment column indicates the maximum change in action per time-step of the simulation. Note that the sign of allowed vertical speed is bounded to the objective of the agent.

3) State Transition Function: The state transition function is fully determined by the underlying dynamics implemented in the BlueSky Open Air Traffic Simulator

TABLE II. Allowed range and increments per time-step for each of the different actions.

Action	Range	Increment
Vertical Speed (m/s)	[-5, 5]	[-5, 5]
Horizontal Speed (m/s)	[5, 15]	[-1.5, 1.5]
Heading (deg)	[0, 360]	[-45, 45]

4) Reward: It is preferred to keep the reward function as simple as possible while encompassing all the requirements of the solution to the problem [14]. This leads to the reward function given in equation 2. Here s_{target} refers to a state in which the agent is in the corresponding target layer and s_{LoS} is a state in which an intrusion with the agent is present.

$$r = \begin{cases} 1 & \text{s = s}_{target} \\ -1 & \text{s = s}_{LoS} \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

B. Deep Reinforcement Learning: Soft Actor Critic

To solve the (PO)MDP defined in section III-A use is made of the Soft Actor-Critic (SAC) DRL algorithm. SAC is an off-policy, model-free, DRL algorithm, which means that it can learn from past experiences without explicitly knowing the environment dynamics or reward function. Off-policy is preferred over on-policy methods because on-policy methods have a lower sample efficiency, which would result in slower learning. SAC also has shown to be very stable during training, even in environments with sparse rewards, which makes it a prime candidate for this research. The hyperparameters used for this research are the same as the ones used by the original authors with a reward scale of 10 [15].

C. Baseline Resolution Algorithm: Modified Voltage Potential

To provide a reference for the performance of the DRL models, all scenarios are also simulated with the MVP conflict resolution algorithm [16]. MVP determines the closest point of approach of two aircraft, and, if the distance between the two aircraft at CPA is smaller than the minimum separation distance, a repelling 'force' is determined which changes the velocity vector such that the shortest way out of the conflict is determined. This is visually presented in Fig. 2. This also entails that, unlike the DRL agents, MVP agents will only change the current course if the aircraft is in conflict. To ensure a fair comparison between the MVP and the DRL models, the MVP model will have the same constraints on their degrees of freedom imposed as their DRL counterpart.

IV. Experimental Setup

A. Experimental Scenario

For all conducted experiments, the goal of the agent is to traverse through the different layers in a layered airspace and reach the target layer without intrusions.

The layered airspace in question consists of 2 sets of 8 altitude layers, each having an allowed heading range of 45

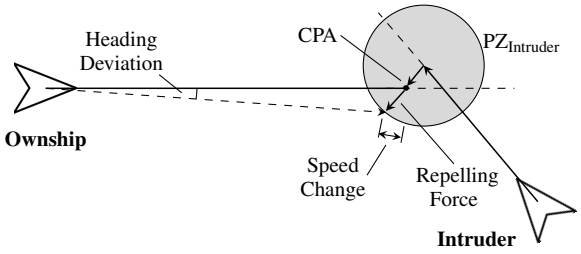


Figure 2. Graphical representation of the MVP algorithm, adapted from [16].

degrees, covering all the possible heading angles twice. The purpose of having 2 sets of layers is that long-distance travel can be done at higher speeds in the top layers, whereas short-distance commute is allocated to the slower bottom layers [5]. For this research, however, the different layers function solely as a way to artificially generate the need for vertical manoeuvres. A transition layer is placed between each layer that can only be accessed by aircraft conducting vertical manoeuvres, which allows the agent to adapt to the new layer before merging. All layers are 25ft in height.

Within this airspace, aircraft operating in the top 8 layers will have a certain probability to obtain a descent command to one of the 8 bottom layers, simulating the delivery of a package. Similarly, aircraft flying in the bottom 8 layers have a probability to get a climb command, simulating the return to a warehouse or place outside of the city. This probability is selected such that on average 5% of the aircraft in the airspace are conducting vertical manoeuvres at any given time. This means that at any given time roughly 5% of the aircraft in the airspace will be controlled by either DRL or MVP.

B. Traffic Density and Conflict Probability

The traffic density in the airspace is selected to be $55 AC/NM^2$, equally distributed over all of the heading layers. The conflict probability between an agent and any other aircraft, based on the equations in Sunil [9], equals 9.9%.

C. Control Variables

1) Simulation time-steps: The simulation is run with time-steps of 1.5 seconds. Thus, the DRL agent selects an action for the aircraft every 1.5 seconds. The MVP agent selects an action for the aircraft every 1.5 seconds only when in conflict.
2) Minimum Separation: The protected zone around all aircraft is set at 50m horizontally (R_{pz}) and 25 feet vertically (h_{pz}). These values are based on comparable work [17], as currently no standard for separation requirements has been specified for unmanned aviation.
3) Conflict Detection: For all experiments, instead of look-ahead time use is made of a 'search cylinder' with a radius of 500m, spanning from the agent's altitude to the altitude of the target layer. All aircraft within this cylinder with a $D_{cpa} < PZ_h$ are evaluated for potential conflicts. This is done by comparing the times in and out of the horizontal and vertical minimum separation. If there is overlap between these times the aircraft are labelled as in conflict. The choice for a look-ahead distance instead of look-ahead time is made to ensure that aircraft that are flying (almost) parallel to the agent, but that are very close in absolute distance, will not be overlooked for state inclusion. This has as a drawback that aircraft with a very high relative speed, and therefore a much smaller T_{cpa} than other aircraft, might initially be ignored.
4) Default Speeds: All cruising aircraft will be flying at a constant horizontal speed of 10m/s. The default vertical speed for the baseline and MVP during climb or descent is 4m/s.
5) Conflict Resolution: For all of the aircraft that are not conducting vertical manoeuvres the conflict resolution is turned off. Solely the agents are responsible for resolving conflicts.

D. Dependent Variables

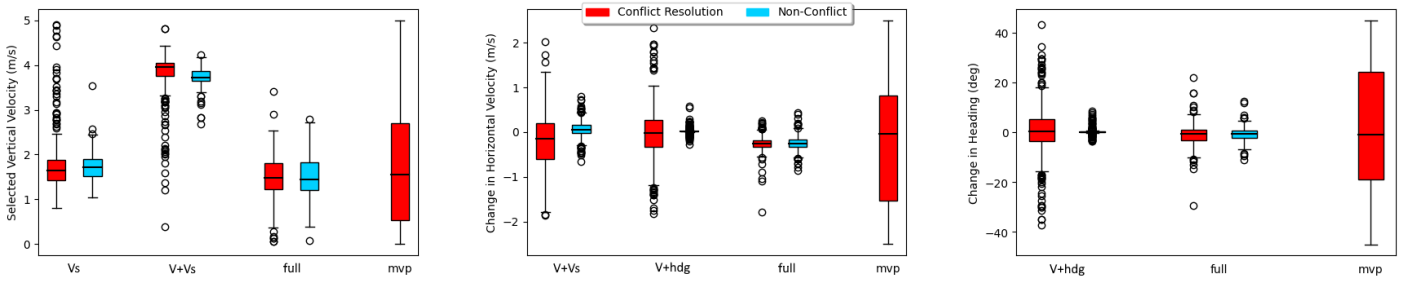
Three safety parameters are used: the average number of conflicts encountered during a vertical manoeuvre, average time spent in conflicts, and the average number of intrusions or losses of minimum separation. The latter is the most important as it directly relates to the safety of the operations. The number of conflicts encountered can give a good indication of the relative stability between the different methods and the percentage of time spent in conflict can be related to the efficacy of the performed resolution manoeuvres.

E. Experimental Hypotheses

1) Action Space Usage: It is hypothesized that all the models that can control the vertical speed will opt for a high mean vertical speed. This hypothesis is based on the findings of Sunil and Tra where it is shown that lower vertical speeds lead to more intrusions [7], [8].

For the horizontal speeds, it is expected that the mean horizontal speed will be equal to the mean cruise speed. This also implies that the mean horizontal speed change will be equal to zero. This hypothesis stems from the fact that having a horizontal speed equal to the cruise speed lowers the relative horizontal velocity between the aircraft.

Finally, the heading changes are expected to be small of magnitude, as large heading changes will also change the observed aircraft by the agent considerably. From a predictability point of view, this is unfavourable, as the agent has less control over the next state.
2) Performance Differences: For the performance differences, it is hypothesized that the models with more degrees of freedom will have fewer intrusions than the models with fewer degrees of freedom at the cost of higher training time. This is because it was shown that having more degrees of freedom increases the safety of a DRL model in a lane-changing and merging task on the highway [10]. Simultaneously it is hypothesized that the total number of conflicts will increase due to the Domino Effect of conflict resolution manoeuvres [18].



(a) (b) (c)

Figure 3. Boxplots for the selected actions during conflict resolution and during non-conflict situations. a) Selected (absolute) vertical speed. b) Selected horizontal speed changes. c) Selected heading changes. Note that these are selected actions per time-step.

Finally, it is not expected that the DRL models will outperform the MVP models. This is hypothesized as MVP has been shown to be very effective at resolving conflicts whilst simultaneously having minimal loss at the overall (path) efficiency. The used DRL models on the other hand are an initial attempt in terms of MDP formulation, model choice and hyperparameter selection and have not yet been extensively optimized and researched in regards of performance.

V. Results

For the results, the performance of the final trained DRL models will be shown next to the performance of the MVP conflict resolution algorithm with the same degrees of freedom. First, the usage of the action space will be shown to illustrate the final trained policies. Then, the results of the safety metrics will be shown for the different models. For the results, more than 10.000 vertical flight manoeuvres have been simulated per model with randomly generated surrounding traffic.

A. Model Policy Differences

To better understand the differences in performance in terms of safety and efficiency the differences in policies between the different models is shown. This is done through a selected action boxplot for all 3 actions, given in Fig. 3.

Analysis of the action space usage shows that the DRL model's policy is semi-independent on the Boolean conflict variable, which specifies if an agent is in conflict or not. Instead, a more notable correlation is observed with the T_{cpa} and D_{cpa} variables, as indicated in Fig. 4, which shows the mean magnitude of the actions for all DRL models against T_{cpa} and D_{cpa}. From this figure, it is noticeable that the most prevalent differences in the policy can be observed for $D_{cpa} < 100$m and $T_{cpa} < 20$s.

This shows that the agent changes course even if it is not necessarily in conflict. Therefore it is decided to define the

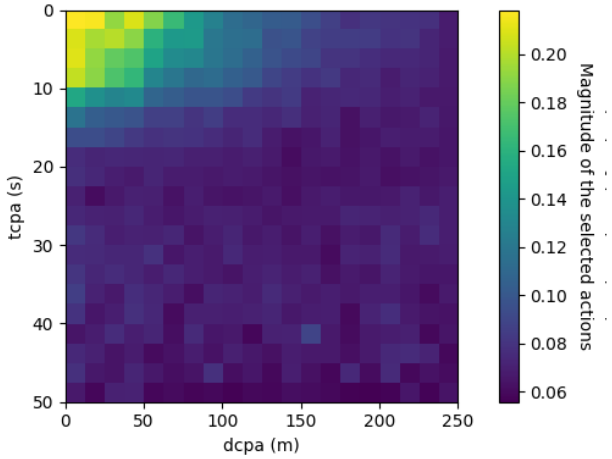


Figure 4. Color plot of the mean selected action magnitude, averaged for all the DRL models and different actions, normalized between 0 and 1. Plotted against T_{cpa} and D_{cpa}

"Conflict Resolution" category in Fig. 3 as all actions selected whilst T_{cpa} is smaller than 20 seconds and D_{cpa} smaller than 100 meters. The "Non-Conflict" category is then defined as all other actions. An exception to this definition is for the MVP model, which is called based on the Boolean conflict variable, for this model the category is simply: selected actions whilst in conflict.

The high number of outliers in Fig. 3 is also a direct result of highlighting only the differences based on T_{cpa} and D_{cpa}, as the influence of other state variables on the selected action can not be represented in the same figure.

From Fig. 3 it is visible that a broader area of the action space is utilized by the DRL models when resolving conflicts than during nominal flight, indicated by the larger whiskers and higher frequency of outliers. This shows that the DRL models successfully learn the concept of danger, and understand that continuing the current course of action might result in dangerous states or even intrusions.

Similarly, the narrow distribution during nominal flight

conditions shows that the agent understands that, in principle, there is no need to change the actions during safe operations.

Fig. 3a shows the selected vertical speeds by the different models. It is interesting to notice the differences in policy between the 'vs' and the 'v+vs' model. The 'vs' model utilizes a low vertical speed during nominal operations, during conflict resolution the mean vertical speed is slightly lower whilst simultaneously having more outliers in the direction of increasing vertical speed. The exact opposite policy is observed for the 'v+vs' model. The reason for the lower vertical speeds of the 'vs' and 'full' models is currently unknown and contradicts the hypothesis that the models would prefer a higher vertical speed.

For the horizontal speeds, it is visible that the mean speed change is centred around zero for all models except the 'full' model. This partially confirms the hypothesis that the models will prefer a horizontal speed equal to the horizontal cruise speed. As the 'full' model has a lower mean vertical speed it might be possible that the lower horizontal speed is used to increase the climb/descent angle, although more research is required to see if this behaviour is actually beneficial for the safety of the operations.

Overall it seems as if the differences in policy between "Conflict Resolution" and "Non-Conflict" are smaller for the 'full' model. It is possible that, because it can combine 3 different actions, the required magnitude of the actions to transition to a safer state is lower, indicated also by the relatively low spread in actions. Furthermore, as the DRL model does not necessarily resolve the conflict in a single time step this smaller spread in selected actions does not indicate that the DRL model is capable of resolving the conflict with smaller total deviations than MVP.

The difference is that MVP will always compute the required state-change to resolve the individual conflicts at the current time-step, this paired with the summation of conflict resolutions in multi-aircraft conflict scenarios can lead to large initial deviations. The DRL model on the other hand can decide to wait before resolving or resolve a conflict in a series of small increments, which might still result in a large total deviation, not indicated by Fig. 3. The only exception to this is the selected vertical speed, as that does not reflect a change from the current state, but an absolute speed.

B. Safety Analysis

First, looking at the total number of conflicts shown in Fig. 5, it becomes apparent that for all cases the total number of conflicts encountered during operations increases with respect to the metrics for not using any conflict resolution. This is a common phenomenon often described as the Domino Effect [18], [9]. In essence, the resolving manoeuvres conducted by the agents result in a larger volume of airspace being used. This in turn increases the number of potential conflict pairs when

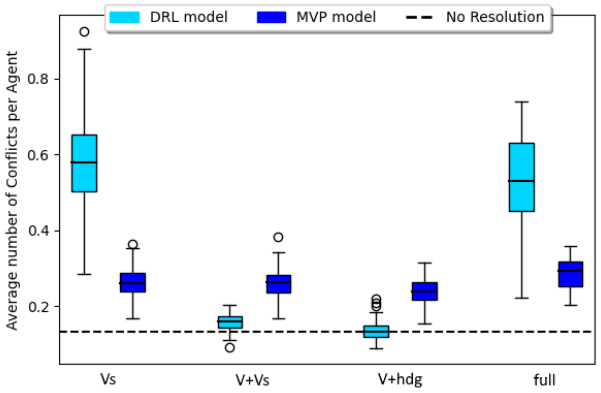


Figure 5. Average number of conflicts encountered during a vertical manoeuvre.

compared to flying in a straight line. The largest increase in the number of conflicts is observed in the 'vs' and 'full' models, and can be partially related to the overall lower vertical speed these models have during operations, as shown in Fig. 3a.

Because of these lower vertical speeds, the duration of vertical operations is also increased, which can result in an increase of conflicts encountered. It is interesting to note that this "Domino Effect" is less apparent in the 'v+vs' and 'v+hdg' DRL models than for their respective MVP models. It is hypothesized that this can be attributed to the ability of the DRL models to act when not in conflict. This, for example, allows the agent to delay returning to nominal conditions after a resolution if it is observed that this would result in a new conflict. Because of this conflicts are prevented and therefore not observed for this metric.

Looking at the percentage of time spent flying in conflicts, shown in Fig. 6, for most cases a decrease in comparison to no resolution is observed. This can be attributed to the conflict resolving actions both the MVP and DRL models use, which effectively shortens the duration of the conflicts. The only model for which this decrease is not observed is for the 'full' model. As the total number of conflicts is comparable to the 'v' model, this indicates that the duration of conflicts for the 'full' model, in general, is longer. This might be caused by either the postponing of resolution manoeuvres or due to less effective resolution manoeuvres.

Finally, with the total number of intrusions per flight given in Fig. 7, it can be seen that all of the models successfully reduce the total number of intrusions when compared to no resolution. Closer inspection of Fig. 7 also shows that increasing the degrees of freedom does not necessarily result in a safer policy for the DRL model. This is interesting, as the policy of the DRL 'vs' model is part of the solution space of the 'v+vs' and 'full' models. Similarly, the policy conducted by the MVP models is also part of the solution space of their respective DRL models. Because the performance of the 'v+vs' and 'full' models does not match the performance

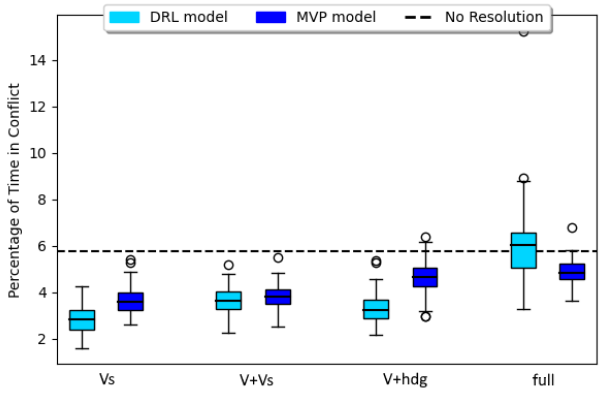


Figure 6. Average percentage of time spent in conflict while conducting vertical manoeuvres.

of better available policies (in this case the policy of the 'v+vs' MVP model), it can be concluded that these models are likely stuck in a local optimum. This highlights one of the drawbacks of using Deep Reinforcement Learning for higher-dimensional problems. With more actions, the required exploration increases exponentially, increasing the required training time whilst decreasing the guarantee of convergence to the global (or a more optimal local) optimum.

A final remark is that the DRL model found a horizontal resolution method that outperforms the MVP model in terms of safety. Fig. 3 already showed that the DRL models also acted when not in conflict. Because of this, the model encounters fewer conflicts during vertical manoeuvres, which in turn leads to fewer potential conflicts leading to an intrusion.

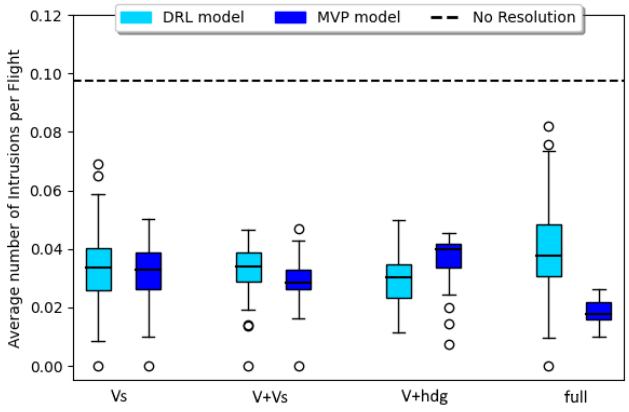


Figure 7. Number of intrusions per vertical manoeuvre.

VI. Discussion

The results of this research have shown that Deep Reinforcement Learning can be used to train a model that learns to reduce the number of intrusions during vertical manoeuvres in a layered airspace. It learned this from a simple reward function that only rewarded successful operations and penalized intrusions. This shows that simple reward structures can be used in complex environments, which has as an added benefit that it is easy to visualize what the desired behaviour of the agent is when compared to more complex reward structures.

During the safety analysis, the DRL model outperformed the MVP model in the horizontal scenarios. This performance difference can be attributed to the fact that the DRL model is able to perform conflict resolving manoeuvres at different moments. MVP on the other hand performs the resolution manoeuvre at the moment a conflict becomes apparent in the state vector, which is bounded by the look-ahead distance. DRL has the freedom to find the optimal moment for conflict resolution. In previous work, it has already been demonstrated that having a constant look-ahead time might not be optimal [19]. Furthermore, the DRL model has control over the selected actions during the recovery phase after resolving a conflict, allowing it to prevent new conflicts from happening.

Additionally, it is interesting to further investigate the fact that the model preemptively acted when aircraft would get close, but would not (yet) be in conflict. This behaviour effectively increases the minimum horizontal separation the model adheres to, which theoretically could decrease the stability of the manoeuvres by increasing the number of conflict resolution manoeuvres [9]. It is possible that the DRL model actively acts as a conflict prevention mechanism apart from resolving conflicts. This has two potential benefits, preventing conflicts requires smaller deviations from the current flight path than resolving conflicts and larger margins with other aircraft increases the available solution space in the case of new conflicts, potentially reducing the occurrence of multi-conflict scenarios where finding a solution is difficult. When looking at the total number of conflicts it seems as if this strategy does indeed lead to a minimal increase in secondary conflicts for both the 'vert+' and 'hor' models.

From the results, it is also visible that the DRL models are not converged to the global optimum. This means that the performance of the DRL model could potentially be further improved with either more training (unlikely if the model is stuck in a local optimum) or a better definition of the Markov Decision Process (MDP). One of the main problems with the current implementation of the MDP is the presence of partial observability in the state representation. Once a problem becomes partially observable, the theoretical guarantee of eventual convergence to the global optimum no longer holds. As the state-space in the simulation environment is continuous and the environment in real life would be unbounded, partial observability will always be a problem. However, expanding the state representation to include more aircraft, researching a more consistent representation and potentially also including historic states (k-th order history approach [14]) can all be done to decrease the impact of partial observability on the performance. Especially the inclusion of historic observations

in the state can prove to be of utmost importance, as currently, it is possible that the agent resolves a conflict and is no longer aware of the existence of the old conflict in the next time step, causing the agent to revert to the old state and back into conflict. This indicates that with the current implementation of the MDP the problem has areas in which the Markov Property does not hold. Another possibility would be to still include aircraft, with which the agent previously was in conflict, in the state, even though there is no imminent danger anymore. A final interesting experiment would be to research the effect of the number of aircraft in the state vector on the performance of the model. It might be that the performance gradually goes up due to reduction of the partial observability, at the same time however a larger state will lead to longer duration of the training time. Optimizing this trade-off can potentially increase the performance further.

Finally, there are some limitations to the results. For example, the traffic scenarios can be adapted to have higher or variable traffic densities and include aircraft flying at different cruise velocities. Furthermore, it is difficult to anticipate how the model would perform in more complex traffic where not all aircraft would adhere to the altitude layers. Apart from this, elements such as static obstacles or maximum horizontal distance travelled during the vertical operations can also influence the effectiveness of the trained model. To estimate the true effectiveness of DRL for safe manoeuvring, it should be trained and tested in a variety of different traffic scenarios consisting of operations during all stages of flight (potentially using different models/policies for different conditions). An initial step in this direction would be the activation of conflict resolution for cruising aircraft. This extra element will remove much of the stationarity and therefore the predictability from the environment, and will better show the ability of the DRL model to deal with emergent behaviour. This would however also lead to massive multi-agent operations, which will negatively impact the stability and duration of training. Overall, the results obtained however do show that DRL can potentially be used for improving the safety and can provide new insights into the understanding of safe operations.

VII. Conclusion

This paper analysed the capabilities of Deep Reinforcement Learning (DRL) for improving the safety of vertical manoeuvres in a layered airspace through direct control. It was shown that DRL is capable of learning policies that effectively reduce the number of intrusions for a variety of different degrees of freedom, even outperforming the Modified Voltage Potential algorithm in certain scenarios. This work shows that DRL can successfully be used for detect and avoid operations in high traffic density scenarios. More research is still required in the design of the Markov Decision Process as well as the DRL model selection to further improve on the obtained performance. Additionally, more analysis is required on the failure cases of the current DRL model to better understand the weaknesses and areas of improvement. Finally, future work should investigate the usage of DRL in more competitive and changing traffic scenarios such as non-uniform traffic densities and for different control tasks such as horizontal control.

References

[1] D. Pierce, "Delivery drones are coming: Jeff Bezos promises half-hour shipping with Amazon Prime Air."

[2] "DHL Express", ""DHL express launches its first regular fully-automated and intelligent urban drone delivery service."

[3] M. Doole, J. Ellerbroek, and J. Hoekstra, "Drone delivery: Urban airspace traffic density estimation," *8th SESAR Innovation Days*, 2018.

[4] "Organization, i.c.a. icao circular 328 - unmanned aircraft systems (UAS). technical report, icao, 2011."

[5] E. Sunil, J. Hoekstra, J. Ellerbroek, F. Bussink, D. Nieuwenhuisen, A. Vidosavljevic, and S. Kern, "Metropolis: Relating airspace structure and capacity for extreme traffic densities," in *Proceedings of the 11th USA/Europe Air Traffic Management Research and Development Seminar, Lisbon, 23-26 June, 2015*. FAA/Eurocontrol, 2015.

[6] J. M. Hoekstra, J. Ellerbroek, E. Sunil, and J. Maas, "Geovectoring: reducing traffic complexity to increase the capacity of UAV airspace," in *International conference for research in air transportation (ICRAT), Barcelona, Spain*, 2018.

[7] E. Sunil, J. Ellerbroek, J. M. Hoekstra, and J. Maas, "Three-dimensional conflict count models for unstructured and layered airspace designs," *Transportation Research Part C: Emerging Technologies*, vol. 95, pp. 295–319, 2018.

[8] M. Tra, E. Sunil, J. Ellerbroek, and J. Hoekstra, "Modeling the intrinsic safety of unstructured and layered airspace designs," in *ATM R&D Seminar*, 2017.

[9] E. Sunil, J. Ellerbroek, and J. M. Hoekstra, "Camda: Capacity assessment method for decentralized air traffic control," in *Proceedings of the 2018 International Conference on Air Transportation (ICRAT), Barcelona, Spain*, 2018, pp. 26–29.

[10] C.-J. Hoel, K. Wolff, and L. Laine, "Automated speed and lane change decision making using deep reinforcement learning," in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2018, pp. 2148–2155.

[11] M. Ribeiro, J. Ellerbroek, and J. Hoekstra, "Review of conflict resolution methods for manned and unmanned aviation," *Aerospace*, vol. 7, no. 6, p. 79, 2020.

[12] J. M. Hoekstra and J. Ellerbroek, "Bluesky atc simulator project: an open data and open source approach," in *Proceedings of the 7th International Conference on Research in Air Transportation*, vol. 131. FAA/Eurocontrol USA/Europe, 2016, p. 132.

[13] R. Bellman, "A markovian decision process," *Journal of mathematics and mechanics*, vol. 6, no. 5, pp. 679–684, 1957.

[14] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT press, 2018.

[15] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, "Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor," in *International conference on machine learning*. PMLR, 2018, pp. 1861–1870.

[16] J. M. Hoekstra, R. N. van Gent, and R. C. Ruigrok, "Designing for safety: the 'free flight'air traffic management concept," *Reliability Engineering & System Safety*, vol. 75, no. 2, pp. 215–232, 2002.

[17] M. Ribeiro, J. Ellerbroek, and J. Hoekstra, "Velocity obstacle based conflict avoidance in urban environment with variable speed limit," *Aerospace*, vol. 8, no. 4, p. 93, 2021.

[18] K. Bilimoria, K. Sheth, H. Lee, and S. Grabbe, "Performance evaluation of airborne separation assurance for free flight," in *18th Applied Aerodynamics Conference*, 2000, p. 4269.

[19] M. Ribeiro, J. Ellerbroek, and J. Hoekstra, "Determining optimal conflict avoidance manoeuvres at high densities with reinforcement learning," *10th SESAR Innovation Days*, 2020.