Analysis, Assessment and Development of Online Path Planning Algorithms for Helicopters under Threat

Andre Nahrwold, Philippe Petit (DLR Braunschweig), Jürgen Plorin (HENSOLDT Sensors GmbH)

Abstract

This paper investigates the scenario, in which a helicopter is attacked by a Small Arms and Light Weapons (SALW) threat, and tries to automatically find an escape path which minimizes the encountered threat on the corresponding escape trajectory. To achieve this goal, well-established path planning algorithms are used in conjuncture with an elementary flight simulation and a threat metric, in order to evaluate different path planning algorithms and their ability to minimize the exposure to a sensed SALW source. Additionally, a genetic optimization algorithm is used to optimize the hyper parameters of the individual path planning algorithms to strengthen their ability to minimize the threat and improve execution time. The optimized path-planning algorithms show a significant improvement in the minimization of the SALW threat of the planned trajectory compared to the original trajectory.

Keywords

Aviation; Helicopter; Path planning; Escape trajectories; Evasion; Hostile fire indication;

1 INTRODUCTION

During combat missions, helicopters enable the operation in difficult to access and poorly developed areas. Areas likes these are often occupied with hostile troops when the operation takes place in a war zone. Because of the low altitude and velocity of a helicopter, these troops and their use of SALW were a serious threat for rotorcraft in combat action during the Iraq War.

FIG 1 displays the number of fatalities of U.S. soldiers in rotorcraft during the Iraq War from October 2001 to September 2009. SALW threats contributed heavily to the losses by causing more then 40% of hostile combat losses [1].

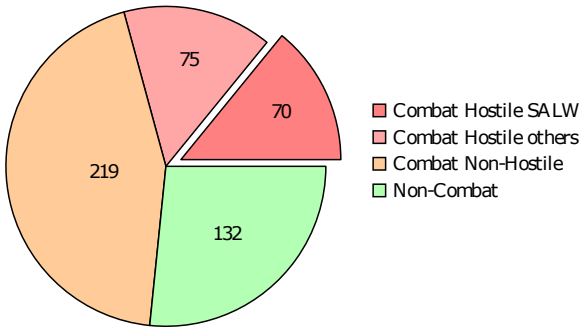

FIG 1. Number of fatalities of U.S. soldiers in rotorcraft during the Iraq War [1]

1.1 Motivation

Today most of the military helicopters can detect and defend themselves against guided missiles. These missiles have an unique signature in e.g. infrared or ultraviolet spectrum in contrast to the projectiles from SALW which are more quiet when fired and on impact onto the helicopter. These factors may lead to a missing awareness of the pilots to the on-going threat of hostile SALW fire. Because of this, pilots may not be able to localize the shooter or react accordingly to this threat.
The company Hensoldt Sensors GmbH (Hensoldt) developed a sensor for hostile fire indication for this purpose. The sensor can detect individual projectiles and predict the position of the shooter based on the trajectory of the projectile. A more detailed description can be found in chapter 1.2. Information about the shooter and especially its position shall then be used to analyze, assess and develop escape trajectories for helicopters.

1.2 Sensor for Hostile Fire Indication

FIG 2 shows the new radar sensor, designed by Hensoldt, which ensures a continuous surveillance of the helicopters surrounding area in order to detect SALW fire. It distinguishes between single shot and salvo fire and delivers an indication of the caliber which can be used to determine the size of the corresponding danger zone. The optimized operating frequency is defined by the Radar Cross Section (RCS) of the bullet to be detected. This results in an operating frequency

©2020

doi:10.25967/490079

of cm-waves in the Ku-Band which also conforms with the relevant airworthiness regulations.
The radar system operates in continuous wave (cw) mode. The cw-signal is used without any frequency modulation in order to provide precise velocity information which leads to missing information about the distance.

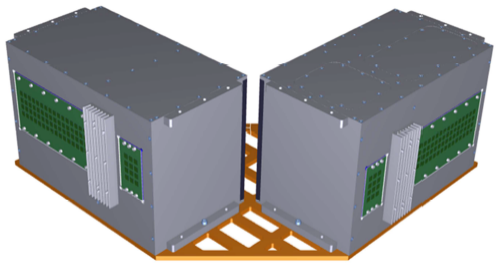

FIG 2. Sensor for Hostile Fire Indication (HFI)

The phased array sensor with digital beam forming has a field of view of 90 deg in azimuth and 45 deg in elevation. For testing, two sensors were mounted at 45 deg and 135 deg with respect to the longitudinal axis on the left side of a helicopter which can be seen in FIG 3. The evaluation processor unit was placed inside of the helicopter.
The system was tested in three different flight test campaigns with intense improvements of the system in between campaigns. At a range of less then 100 m and four different calibers ranging from 5.56 mm up to 20 mm, single shots were detected with 98.7% and salvo fire with 100% at a false alarm rate of about 0.1%.

FIG 3. Sensor System Mounted on the Test Helicopter

The goal for future development is to reduce the size of the sensor to about a quarter of the current volume and to integrate the processing unit with the sensor.

1.3 Goals

The main goal of this paper is to develop an online path planning to reduce the threat posed by a SALW shooter to the vehicle and the crew, detected by this new radar sensor. For this purpose, a scenario is used where the autopilot of a single helicopter follows an automatic waypoint mission when encountering a single SALW threat. The computation unit of the helicopter is then tasked to plan an escape path which minimizes the damage received by the sensed SALW threat and then let the autopilot automatically follow these waypoints to generate the corresponding escape trajectory. Reacting to the encounter of the threat must be done in about two seconds due to the criticality of the threat.

2 METHODICAL APPROACH

In order to achieve the stated goals, three path planning algorithms were evaluated which were choosen according to given limitations. These limitations demand, that the algorithm must be deterministic which would enable a certification for the use in helicopters and rules out some optimization based or artificial intelligence path planning algorithm. Other path planning algorithms may not produce the optimal path but they will deliver one possible path to the pilot or autopilot in a deterministic time-frame.
Because the helicopter does not only interact with the SALW threat at the position of the waypoints which form the planned escape path, an elementary flight simulation was implemented. This flight simulation emulates an autopilot which follows the waypoints within constraints and dynamics of a simplified real helicopter to generate flight trajectories.
Additionally, a self-developed threat metric was used in order to compare different trajectories in respect to their ability to minimize the threat to the crew and helicopter, while the helicopter is inside of the effective range of the SALW threat.
Every path planning algorithm is configurable by a set of parameters. In order to tune these parameters a genetic optimization algorithm was employed to tune each algorithm for execution speed and its ability to minimize the threat metric.

2.1 Overview

This chapter describes the basic concepts for the path planning, optimization, flight simulation and the threat metric. For a better understanding, a few terms will be introduced in the following paragraphs.
A **waypoint** is defined as a state of the helicopter in the environment, containing its 3D position and the scalar velocity.
A **path** is defined as a sequence of waypoints with equal distances, often allocated over larger distances.
A **trajectory** is defined as a time quantized sequence of simulation steps with position and velocity information of an object following a path.
A trajectory is called **original** when it is the trajectory of the planned path before encountering the threat. Therefore an **escape trajectory** belongs to a planned escape path.

©2020

2.1.1 Desktop simulation

To test and compare the algorithms against each other a common basis for the execution is necessary which was implemented in a MATLAB based desktop simulation. The desktop simulation reads all the inputs, containing information about the environment, the helicopter and the shooter. After processing the inputs and the simulation itself, the escape trajectory and the threat metric is set as the output. The simulation starts by planning the escape path, followed by generating the escape and original trajectory via the flight simulation. Based on these two trajectories, the threat metric is calculated, indicating their threat level.

2.1.2 Helicopter damage model

In order to quantify the damage a shooter imposes on the helicopter on a given flight trajectory, a damage model was developed. This model defines areas of triangular shape on the outside of the helicopter with equal damage points. These damage points are a representation of the amount of damage within a range from one (uncritical damage) to one hundred (critical damage).
Triangular shapes were used because it simplifies the calculation of the intersection between the areas and the Line of Sight (LOS).

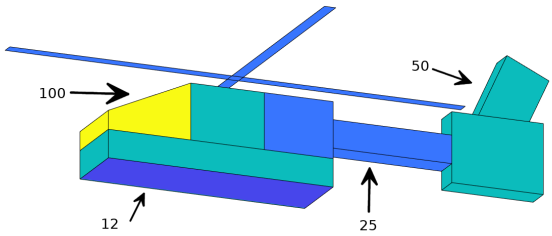

FIG 4. Damage points distribution

The helicopter damage model can be seen in FIG 4 whereby the helicopter dimensions were derived from the Airbus H135 helicopter. The distribution of damage points among the different parts was based on general assumptions and not on the specific characteristics of this helicopter. For example the pilots cabin is considered to be very critical because of weak materials (100 damage points) and the bottom of the helicopter is less critical because it is more likely to be equipped with armour (12 damage points).

2.1.3 Environment and scenarios

The urban area taken from a simulator scenario displayed in FIG 5 is used as the environment basis for this study.

FIG 5. Urban area of the simulator scenario

This environment basis was then transformed into the internal representation which can be seen in FIG 6. The white rectangles and the TV tower represent the obstacles which are enlarged by the size of the helicopter model to simplify the path planning. A red dot indicates the origin of the threat (shooter position) which is surrounded by two spheres defining its influence, here shown as circles. The threat trigger distance marks the area where the shooter will begin to shoot at the rotorcraft. On the other hand, the threat mean range defines the maximum distance where the shooter can effectively hit a rotorcraft.

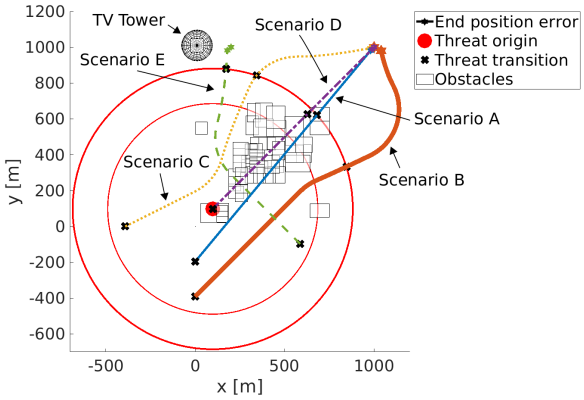

FIG 6. Top view onto the scenarios

The differently plotted lines in FIG 6 and FIG 7 represent trajectories that the helicopter planned to fly prior to encountering the SALW threat. Five different scenarios were developed which describe different confrontations between the shooter and the helicopter. The shooter remains at the same position in every scenario and only the entry angle and flight altitude of the helicopter is changed within the original paths. In both figures, **x** marks a transition into the threat trigger distance and out of the threat mean range. These marks show that the displayed trajectories always start when the threat is encountered, which is manually achieved by correctly placing one waypoint of the original path on this point. All escape trajectories therefore start when the original path is crossing the sphere of the threat trigger distance. The end of the planned path and the last position of the trajectory are each marked with a hexagon, connected by a straight line visualizing the horizontal or vertical error to the end of the planned path.

©2020

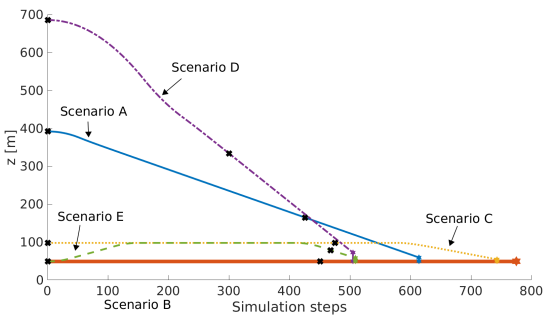

FIG 7. Heights profile of the scenarios

2.1.4 Shooter model

The shooter, operating the SALW threat, can roughly be characterized by its physical aptitude and its used weapon. Using a specific weapon defines the maximum range of the shooter and the corresponding firing rate. The weight of the weapon and the physical aptitude form the parameter of maximum angular velocity a shooter can achieve.
To achieve a mean value for all these parameters, six weapons from the SALW category were considered that were used in a study of a similar hostile fire indication sensor [2]. In summary, the mean range is about 800 m, the mean weight is 5.4 kg and the mean firing rate of the shooters weapons is 471 shots per minute. Using the mean weight of the weapons, an empirical measurement lead to the maximum angular velocity a shooter could achieve of $\omega_{shooter}$ = 1.97 rad/s.

2.2 Path planning

Combining the information of the environment and the shooter shall provide a safer path for the helicopter. Three well-established path planning algorithms which are modified for this special situation of threat are used: A* algorithm, Follow The Gap Method (FGM) and Artificial Potential Field (APF).
As described in a previous chapter, one waypoint of the original path is placed directly at the transition into the threat zone. This waypoint is used as the start of the planning. The end of the path planning is defined by the last waypoint of the original path. To determine whether the end is reached or not and to secure a finite end of the planning, a fictional line is tied between the start and the end of the path planning. The position of the current planning step can then be projected onto this line to measure the progress of the planning procedure.

2.2.1 A* algorithm

A* was chosen because it is established and often used to find optimal paths in a graph. To find the optimal path, the algorithm combines the ideas of the Dijsktra algorithm and the Greedy Best-First-Search. While the Dijsktra algorithm only monitors the costs to get to a position in the graph, Greedy Best-First-Search only monitors the heuristic, an esteemed distance between the current node and the end. Therefore the heuristic acts like a direction beam to the goal, whereas the costs include the appearance of the environment. Combining the monitoring of the costs and the heuristic is the main idea of A* which provides fast and focused optimal path planning. [3]

Because the shooter and its area of threat is a part of the environment, the cost calculation must include the information of the SALW threat. Just treating this area like an obstacle would mean to start the path planning inside of an obstacle. This would lead to costs with a magnitude of infinity right from the beginning and an undirected propagation of the search in every direction. Therefore, the costs of nodes in the SALW threat area are calculated depending on the distance to the shooter. This dependency is transformed into the three different shaped equations (1) - (3) which are plotted in FIG 8.

(1) $$g_{red}(d) = g_s + g_f - g_s^{\frac{d}{d_{smr}}}$$

(2) $$g_{green}(d) = \frac{g_f - g_s}{d_{smr}} * d + g_s$$

(3) $$g_{blue}(d) = g_s^{1-\frac{d}{d_{smr}}}$$

The variable d describes the euclidean distance between any point and the shooter and the constant d_{smr} defines the threat mean range of the shooter. The constants g_s and g_f define the maximum costs at the shooter position and the costs of a free node.

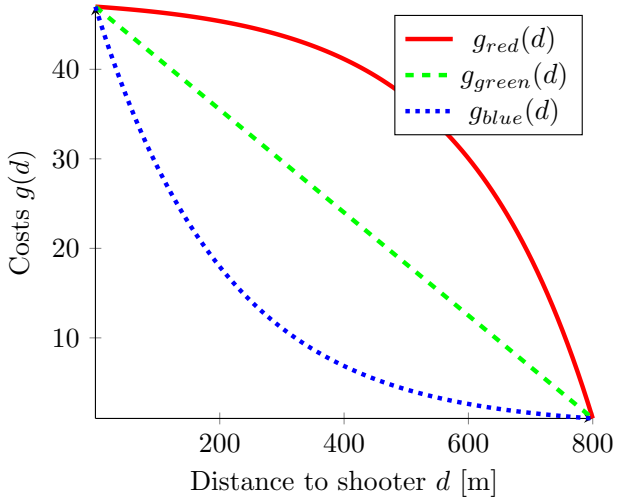

FIG 8. Cost model for shooter area (optimized parameters)

2.2.2 Follow The Gap Method

The FGM algorithm promises an online trajectory generation for 2D robots with safe trajectories and the absence of local minima. These abilities are partly achieved by taking the limitations of maneuverability and the Field of View (FOV) of the robot into account, which can be seen in FIG 9 as pink and green dash dotted lines. All obstacles in the FOV are modelled as circles with a given radius and the first step is to find the biggest gap between the resulting circles. After the biggest gap is found, the bearing to the middle of the gap and to the goal are calculated into the control bearing of the robot. [4]

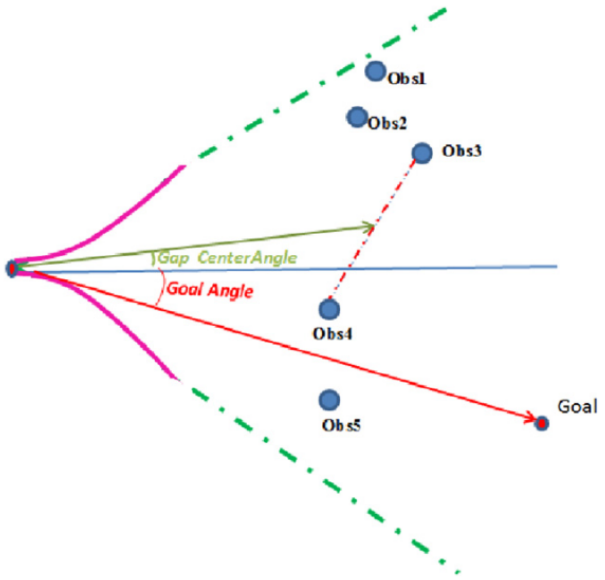

FIG 9. Operative environment FGM [4]

Normally the algorithm would be processed in every execution cycle of the robot to calculate the next move, generating a trajectory. Using this algorithm for path planning was done by emulating this execution cycle to calculate the next waypoint in the path.
To give the path planning the ability to work in a 3D environment, unit vectors were used as bearings instead of angles and the FOV was defined as a fixed-sized cube in front of the helicopter.
Finding the biggest gap in this new FOV is done by firstly creating a depth map of the area containing the distances to every obstacle inside of it. The vertical and horizontal gaps are then determined out of the depth map by combining elements of the FOV horizontally or vertically with the same distance to the helicopter. Lastly, the gap sizes and the depth are used to rate each element of the FOV and choosing the element with the highest score as the biggest gap.
When encountering a shooter, its relative position to the helicopter (right or left) is determined and fed into the rating process. All elements on the side of the shooter get the worst rating possible which leads to a higher probability that the path planning will choose the biggest gap on the opposite side.
This modified data can then be used to calculate the final bearing $\vec{e}_{final}$ which determines the direction of the next extension of the path planning. When the shortest distance to the nearest obstacle d_{min} is equal to zero, the bearing to the best gap $\vec{e}_{bestGap}$ is directly used as final bearing. In the other case, the bearing to the goal $\vec{e}_{goal}$ is additionally used for the calculation. In order to configure the influence between these both bearings, the factor α was introduced.

$$\vec{e}_{final} = \begin{cases} \vec{e}_{bestGap} & d_{min} == 0 \\ \frac{\frac{\alpha}{d_{min}} * \vec{e}_{bestGap} + \vec{e}_{goal}}{\frac{\alpha}{d_{min}} + 1} & d_{min} > 0 \end{cases} \tag{4}$$

2.2.3 Artificial Potential Field algorithms

The original APF algorithm is presented in “Real-time obstacle avoidance for manipulators and mobile robots” [5] and led to many altered versions over time which all follow the same basic principle, like [6] and [7]. This principle uses an artificial potential field consisting of elements which can reject and attract the robot through the environment. Calculating the rejection and attraction is done using geometric dependencies and logical assumptions derived from the rules presented in “Potential Fields Tutorial” [8].
To work in the proposed situation of SALW threat, adding the shooter is done by modelling another rejective spheric obstacle into the existing environment. All rejections $\overrightarrow{rej}$ and attractions $\overrightarrow{attr}$ can then be used to calculate the unit vector $\|\vec{n}_{trans}\|$ which determines the direction of the next extension/translation of the path planning. The corresponding calculation rule is derived from the FGM algorithm which can be seen in the following equation. With γ, a configuration variable was introduced to manipulate the influence between the attraction and rejection.

$$\|\vec{n}_{trans}\| = \left\| \frac{\gamma * \overrightarrow{attr} + \overrightarrow{rej}}{\gamma + 1} \right\| \tag{5}$$

2.3 Flight simulation

The flight simulation provides realistic flight trajectories by emulating an elementary autopilot of a helicopter which follows a given path of waypoints. First step of the procedure is to determine the initial state of the simulation from the first waypoints of the given path. The state vector $\vec{X}$ is defined by a position (x, y, z), a velocity (v), the azimuth of the helicopter (χ) and its rate of climb $(\dot{z})$.

$$\vec{X}^{\intercal} = \begin{pmatrix} x & y & z & v & \chi & \dot{z} \end{pmatrix} \tag{6}$$

Moving along the given path, is done by inspecting the fictional line between two following waypoints, called a section, in comparison to the current state of the simulation. In case the end waypoint is not reached the value of the errors between the current state of the simulated helicopter and the current section are calculated. These error values describe the deviation of the velocity e_v, the crosstrack error e_{hor} and the vertical position error e_{vert}. Calculating the deviations and their velocities $\dot{e}_v$, $\dot{e}_{hor}$ and $\dot{e}_{vert}$ is done via (7) - (12).

The three different used velocities are the velocity of the current state v_a, the target velocity from the current section v_t and the velocity from the last simulation step v_{ao}.

$$e_v = v_a - v_t \tag{7}$$

$$\dot{e}_v = \frac{|v_a - v_{ao}|}{\Delta t} \tag{8}$$

$$e_{hor} = \overrightarrow{pos}_{hor,rot,y} \tag{9}$$

$$\dot{e}_{hor} = \overrightarrow{v}_{hor,rot,y} \tag{10}$$

$$e_{vert} = \overrightarrow{d}_{intersect,act,z} \tag{11}$$

$$\dot{e}_{vert} = \dot{z}_a \tag{12}$$

The differential equation $\frac{d}{dt}\overrightarrow{X}$ which is implemented by this simulation uses second order systems to calculate the derivatives of v, χ and $\dot{z}$. Derivatives of x, y and z are calculated using logical and geometrical dependencies. In the mentioned equation the term $minmax(...)$ stands for physical and logical boundaries of the corresponding state variable, like the maximum turning angular velocity.

$$\frac{d}{dt}\overrightarrow{X} = \begin{pmatrix} v * \cos(\chi) \\ v * \sin(\chi) \\ \dot{z} \\ k_1 * e_v + k_2 * \dot{e}_v \\ minmax(k_3 * e_{hor} + k_4 * \dot{e}_{hor}) \\ minmax(k_5 * e_{vert} + k_6 * \dot{e}_{vert}) \end{pmatrix} \tag{13}$$

To test this functionality, an exemplary reference path was constructed which was then fed to the flight simulation. The sample output of the flight simulation can be seen in FIG 10.

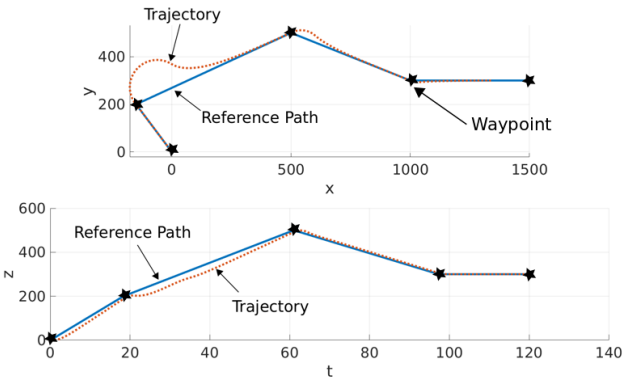

FIG 10. Examplary flight simulation

2.4 Threat metric

In order to compare different trajectories in respect to their exposure to SALW threat, a numeric metric was developed. This metric is called the threat metric Θ which consists of three parameters.

$$\Theta = \theta_{LOS} + \theta_h + \theta_d \tag{14}$$

The first parameter θ_{LOS} measures how many simulation steps are placed within the LOS and the range of the shooter. When a LOS is established, the maximum probability to be hit by a bullet is reached when the shooter can follow the helicopter, increasing the value of θ_h by one each simulation step. Following the helicopter is possible when the angular velocity of the LOS ω_{LOS} is lower than the maximum angular velocity of the shooter $\omega_{shooter}$. Combining the helicopter damage model, the shooter model and the LOS, an amount of damage per simulation step can be calculated which can be averaged into the value θ_d.
The composition of the threat metric is visualized via an exemplary scenario shown in FIG 11 where a LOS is established on the line $P1 - P2$ and $P3 - P4$.

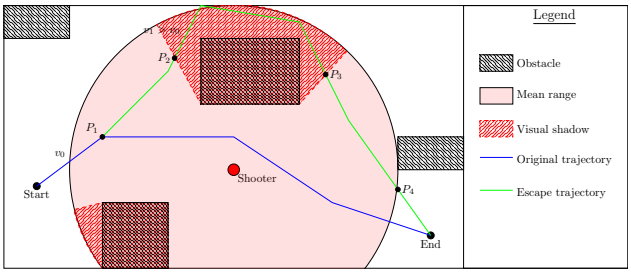

FIG 11. Composition of the threat metric

With a velocity of v_0 until the helicopter reaches the point $P2$, the helicopter is slow enough that the shooter can follow and hit it. But as the helicopter moves past the point $P2$, its velocity has increased, as a reaction to the threat, leading to a velocity v_1 which is too high as that the shooter can follow it again after passing point $P3$.

©2020

2.5 Optimization

To ensure the safety of the helicopter and the crew, the path planning algorithms must be configured to act in the threat induced environment as good as possible. As manual tuning of these parameters would be cumbersome, an automatic optimization will be executed using a genetic algorithm. The basic idea of such an algorithm is to replicate the three main processes in nature proposed by the darwinism: *reproduction/heredity*, *variation* and *selection*. When these principles are applied to a population of entities, their genes will develop with each generation to solve problems better then the generation before. [9] Defining a parameters as a gene, a population of path planning parameter sets (entities) can be generated to solve the problem of finding the best trajectory through the threat zone.

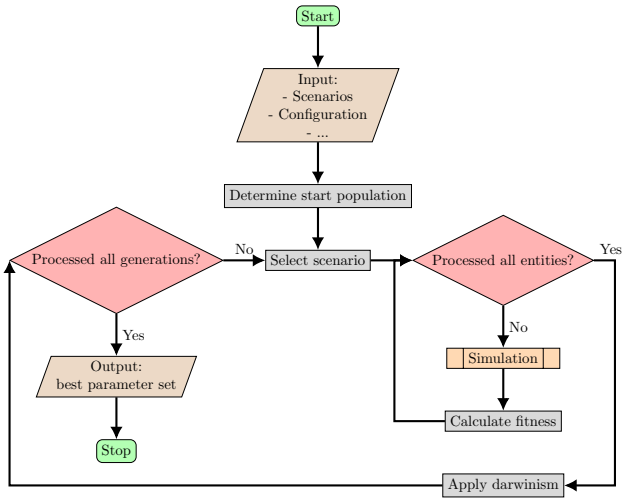

FIG 12. Optimization procedure

The optimization process is shown in FIG 12 where the start population is determined at the beginning. Random values or a snapshot of a previous population can be used as a start population. With an established start population, a scenario is selected for every generation out of the five developed scenarios described in chapter 2.1.3. For each entity in a generation, the simulation sub module from the desktop simulation is executed, followed by the calculation of the fitness of the corresponding entity. When all entities in a generation are processed, the darwinism is applied onto the current population, generating a new population and indicating the best entity corresponding to the currently best parameter set. The process is terminated after the defined number of generations has been reached.

Rating the solutions is done by eq. (15) which first of all consists of the developed threat metric Θ to implement the requirements of the escape. But to also implement the requirements for the path planning itself, the amount of simulation steps that lie inside of obstacles θ_{block} and the error of the end of the trajectory relative to the desired end $e_{end,hor}$ $e_{end,vert}$ are measured. Since an escape trajectory should be planned fast enough, the corresponding time to plan the escape path t_p is also used as part of the fitness.

$$fit = \Theta + \theta_{block} + t_p + e_{end,hor} + e_{end,vert} \tag{15}$$

Applying the darwinism is done by firstly selecting two entities from the old population to be the parents of one entity in the new population. The selection of the parents depends on their fitness value, the lower the fitness the more likely it is that an entity is selected. These parents form a new entity by randomly selecting which parameter from which parent will be passed to the child which is done with a 50% probability. Following the birth of the new child, each parameter may now be again randomly reset with the probability of a given mutation rate.

3 EXECUTION AND RESULTS

This section describes the execution of the optimization, the desktop simulation and the collection of the resulting data.

To have the optimized parameter sets available at the start of the desktop simulation, the optimization itself was executed before the desktop simulation. The optimization is configured with a population size of 400 entities which can develop itself within 40 generations with a mutation rate of 1%. Consequential 16,000 solutions are examined per path planning algorithm which altogether had a processing time of about four days. The execution took place on a desktop PC with an Intel(R) Core(TM) i5-6200U CPU @ 2.30GHz and 8GB RAM.

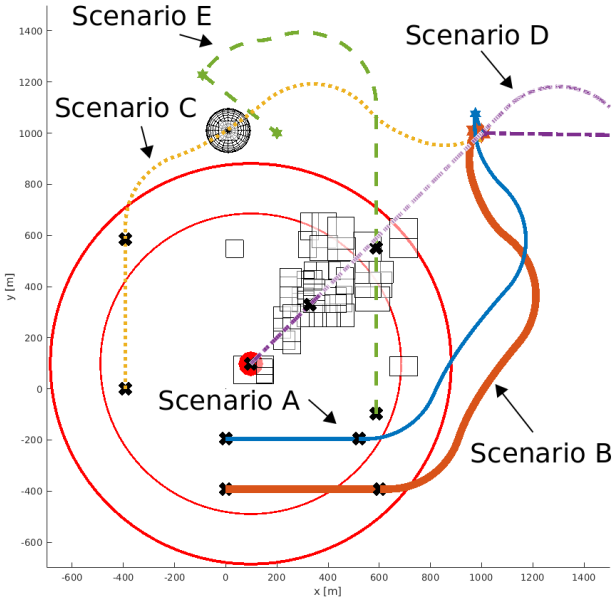

FIG 13. Top view A* trajectories (original parameters)

©2020

The following details of the desktop simulation results are focused on the A* algorithm. FIG 13 and FIG 14 show the results of the desktop simulation execution with the original parameters. Characteristics of these trajectories in scenario A, B and C are long curves from the transition into the threat to the goal while Scenario D missed the goal by more than two kilometres. The heights profile shows that all trajectories tend to firstly rise high above the shooters range before rapidly decreasing the height again.

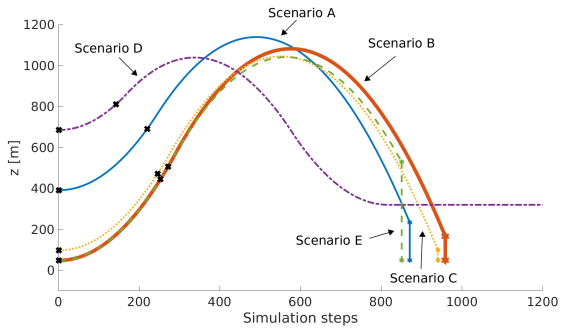

FIG 14. Heights profile A* trajectories (original parameters)

The figures FIG 15 and FIG 16 show the results of the desktop simulation execution with the optimized parameters. The top view reveals that all trajectories are now more focused to move directly towards the goal. Also the heights profile shows, that most of the trajectories do not tend to rise up after encountering the threat.

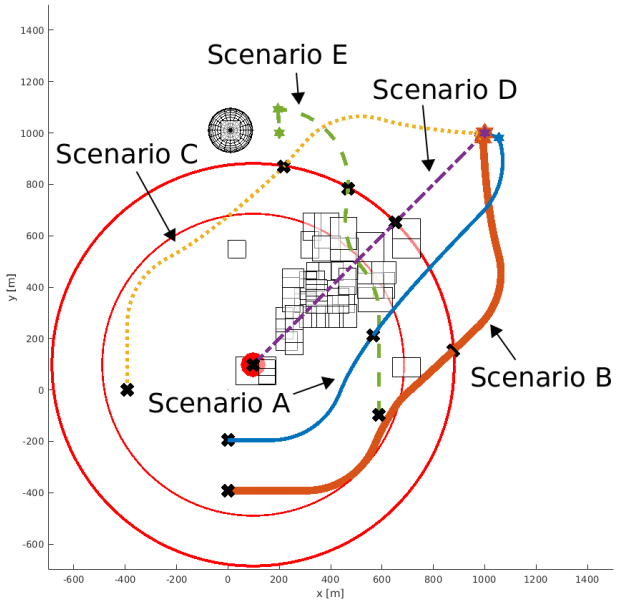

FIG 15. Top view A* trajectories (optimized parameters)

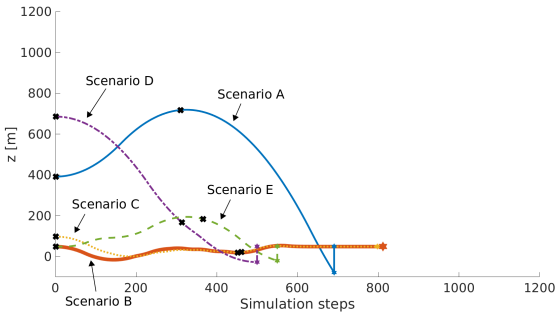

FIG 16. Heights profile A* trajectories (optimized parameters)

The execution of the optimization increased the mean threat metric Θ in a range from 0.7% to 8.7% depending on the algorithm. In contrast to the threat metric, the mean planning time of every algorithm was decreased significantly, e.g. mostly for A* with 5.15 s to 1.18 s. Besides the planning time, also the horizontal and vertical error to the end decreased from a few hundred metres to a few tenth of metres.
Besides the optimization results, TAB 1 shows the simulation results averaged over all scenarios and for the different algorithms with their optimized parameters compared to the original trajectories of the developed scenarios. The value Θ_{rel} is defined as the relative threat metric compared to the threat metric of the original trajectory which shows that all algorithms achieve an increase of safety for the helicopter and its crew. Even tough the threat metric is the lowest for the APF algorithm, the FGM algorithms for example achieves a better precision reaching the end of the path planning.

	Original	A*	FGM	APF
$t_p[s]$	—	1.18	0.09	0.17
$e_{end,hor}[m]$	22.08	44.49	25.4	58.31
$e_{end,vert}[m]$	9.61	54.29	28.23	21.05
Θ	1479.02	1255.76	1201.71	963.26
$\Theta_{rel}[\%]$	100	85	81	65

TAB 1. Mean values of simulation results (optimized parameters)

4 SUMMARY AND CONCLUSION

The main goal of this study was to develop online path planning algorithms to minimize the SALW threat encountered by a helicopter. In order to meet this objective, a simulation workflow was implemented which enabled us to evaluate different path planning algorithms for this objective. Several shortcomings or optimization opportunities in the simulation workflow were identified:
As a basic part to solve the proposed problem, the helicopter damage model was introduced in chapter 2.1.2 which can be used to measure the potential

©2020

damage of a projectile hitting the helicopter. However, the damage model was only aligned with the trajectory; roll and pitch angles were not taken into account. In order to increase the fidelity of the damage model simulation, these attitude angles should be considered in future work.

In chapter 2.1.4 a shooter model was introduced, which also features a property indicating the maximum angular velocity a shooter can achieve during tracking of a target. This limit is however never achieved. At 35 meters distance, the helicopter would have to fly at approximately 120 knots in order to exceed the maximum angular velocity the shooter model can achieve.

The flight simulation implemented the simplified approximated flight dynamics of an autopilot enabled helicopter, described in chapter 2.3. Because the limits of the flight simulation were not included in the path planning algorithms, some discrepancies between the planned path and the executed trajectory were found. As can be seen in FIG 17, the planned path exceeded the imposed limits on the vertical acceleration $\ddot{z}$, which led to large deviations in the height. In future work, the capability of the simulated helicopter will have to be accounted for more thoroughly in order to avoid such issues.

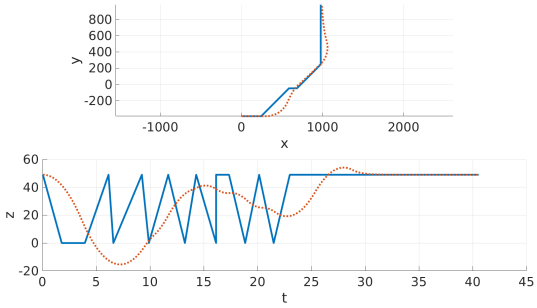

FIG 17. Flight simulation discrepancies (Scenario B)

For the path planning algorithms some conclusions can also be drawn:

One of the analyzed path planning algorithm was the A* algorithm described in chapter 2.2.1 which calculates the costs of a node in the zone of threat depending on the distance to the shooter. This led the A* algorithm to command a climb in a lot of scenarios, in order to increase the distance between the shooter and the helicopter. The manually tuned A* algorithm is especially prone to this effect as can be seen in FIG 14, as the red cost from equation (1) was used. As the optimized algorithm used the blue cost function, the effect is slightly less as can be seen in FIG 16. In reality, gaining height after a SALW threat is detected, is the worst possible action as helicopters only have a limited capability to climb. A much better decision is to fly is to loose height, in order to trade height for speed and escape the threat faster. This coupling is however not reflected in the flight dynamics nor in the cost function, which should be adressed in future work.

The basic idead of the FGM algorithm was maintained while many things had to be changed to work in the 3D environment under the proposed circumstances. Changes included the construction of the FOV, the calculation of the best gap, the definition of the bearing and the introduction of the shooter which were explained in chapter 2.2.2. The optimization of its hyper parameters led to a minimal improvement of the trajectories and their ability to minimize the SALW threat which can be linked to small amount of possible hyper parameters of this algorithm. Even though the optimization could not achieve any significant improvement, the algorithm itself provided better escape trajectories than the A* algorithm in a shorter period of time.

To modify the APF algorithm, we included another rejective element at the position of the shooter and extend the equations for a 3D environment which is explained in chapter 2.2.3. Shorter and more focused trajectories where the outcome of the optimization of this algorithm which can be connected to a higher ratio between the rejections and the attractions forcing the path planning to aim more for the goal. While this algorithm needs twice the planning time as the FGM algorithm, it could achieve the most improvement of the threat metric compared to the original trajectory.

Optimizing the hyper parameters of the path planning algorithms, led to an unintended increase of the threat metric, as the influence of the threat metric on the fitness was too small compared to the other parameters. As described in chapter 2.5, the best parameter set was determined out of the last population which may lead to missing the optimal parameter set of all examined parameter sets. Nevertheless, the optimization decreased the error to the end point and the planning time by a significant factor.

In order to apply our results to systems which can be practically applied, future research should focus on multiple helicopters which show a coordinated response to a SALW threat as well as pilot studies, in order to increase the acceptability of such a system.

REFERENCES

[1] Mark Couch and Dennis Lindell. “Study on Rotorcraft Survivability”. In: *Rotorcraft Survivability* (2010) (cit. on p. 1).

[2] Charles Shepard. “Hostile Fire Detection for Parapublic Operations”. In: *American Helicopter Society 68th Annual Forum 2012* (May 1, 2012) (cit. on p. 4).

©2020

[3] Amit Patel. *Introduction to A**. Jan. 8, 2019. URL: http://theory.stanford.edu/~amitp/GameProgramming/AStarComparison.html (visited on 02/01/2019) (cit. on p. 4).

[4] Volkan Sezer and Metin Gokasan. “A novel obstacle avoidance algorithm: “Follow the Gap Method””. In: *Robotics and Autonomous Systems* 60.9 (2012), pp. 1123–1134. DOI: 10.1016/j.robot.2012.05.021 (cit. on p. 5).

[5] O. Khatib. “Real-time obstacle avoidance for manipulators and mobile robots”. In: *Proceedings. 1985 IEEE International Conference on Robotics and Automation*. Institute of Electrical and Electronics Engineers, 1985. DOI: 10.1109/robot.1985.1087247 (cit. on p. 5).

[6] Kyoung-Youb Kwon, Jeongmok Cho, and Joongseon Joh. “Collision Avoidance of Moving Obstacles for Underwater Robots”. In: 2013 (cit. on p. 5).

[7] Lei Tang et al. “A novel potential field method for obstacle avoidance and path planning of mobile robot”. In: *2010 3rd International Conference on Computer Science and Information Technology*. IEEE, 2010. DOI: 10.1109/iccsit.2010.5565069 (cit. on p. 5).

[8] Michael A. Goodrich. “Potential Fields Tutorial”. In: 2002 (cit. on p. 5).

[9] Daniel Shiffman. *The Nature of Code: Simulating Natural Systems with Processing*. The Nature of Code, 2012. ISBN: 978-0985930806 (cit. on p. 7).

©2020