# HELICOPTER LOW LEVEL FLIGHT USING TRAJECTORY PLANNING AND OBSTACLE AVOIDANCE

**Volker Gollnick* and Torsten Butter,**
**Eurocopter Deutschland GmbH,**

**Bernhard Reppelmund+**
**RWTH Aachen, Inst. Of Flight Dynamics**

**Keywords**: *Trajectory Planning, Obstacle Avoidance, Flight Guidance, Flight Management*

## Abstract

*Emergency Medical Services and Military Missions require helicopter capabilities to operate in unknown areas with reduced visibility and the risk of obstacle collision.*
*This paper presents a 4D-trajectory algorithm developed at Eurocopter Deutschland, which allows an online optimized trajectory planning considering also operational constraints. A second algorithm is introduced to avoid short term unknown obstacles during a mission while following a pre-planned route. Simulation results are presented verifying the applicability of the algorithm.*

## 1 Introduction

More and more Emergency Medical Services (EMS) are required to operate helicopters for their missions also under adverse weather conditions. For military helicopter missions also robustness against adverse environmental conditions including regional threats is still a demanding requirement. In order to improve the mission capabilities Eurocopter Deutschland performed several research projects about all weather capabilities for future rescue missions, [1]. Fig. 1 presents the system architecture of the All Weather Rescue Helicopter (AWRH) experimental project, which was performed within several national German Research programmes.
An advanced flight management system (AFMS), a new Data Link system, an active obstacle detection sensor and new Flight Management and Guidance Displays are the main features of the system especially addressing the 4D-flight guidance aspects for low level flight.

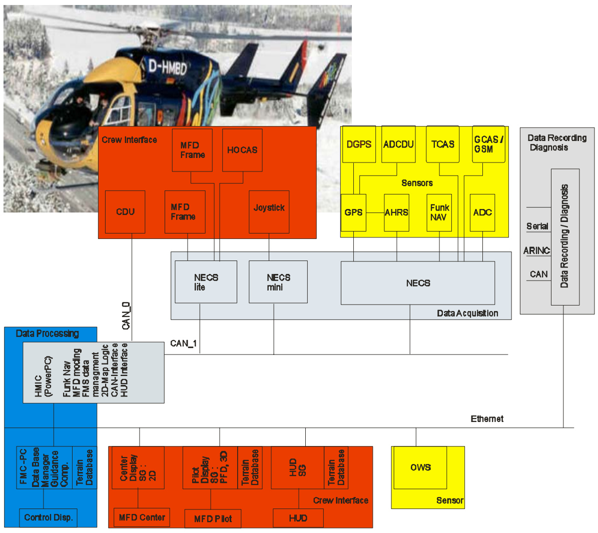


**Fig. 1: System Architecture of the All Weather Helicopter Experimental System**

The AFMS allows the crew during flight to modify start, approach and landing phases of a created route. The exchange of weather, airspace and route information between the airborne system and a simulated air traffic control and weather stations was realised by a SatCom system.
This feature provides the capability to adapt the flight profile during the mission in a cooperative manner.
In a next step the system should be able to calculate optimised trajectories in flight

*now: EADS Corporate Research Centre, Munich
+now: German Forces Flight Test Center, Manching

considering also actual and local limitations and constraints like restricted airspaces, new obstacles or local adverse weather conditions.

## 2 Flight Management System

The Flight Management System (FMS) presented here shall fulfil these requirements. In the following the principle architecture and functionality of the FMS is described, [2], [3].

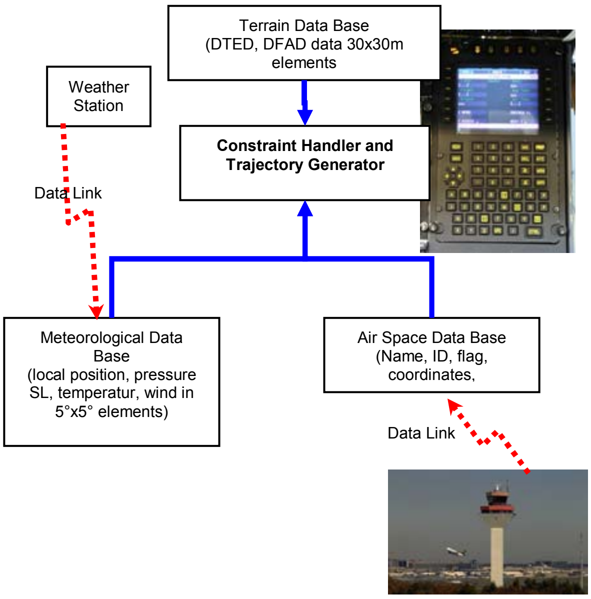


**Fig. 2: FMS Architecture and Data Bases**

All necessary data used for the calculation of the trajectory is provided and stored in separate data bases of the FMS.

The meteorological data base (MDB) is regularly updated also in flight through a data link. It contains weather information about local position, atmospheric pressure levels, locally measured temperature and wind conditions in a specified format. These data are stored in elements of *5°x5°* size.

The air space data base (ASDB) contains defined airspaces based on ARINC424 specification. In the ASDB each air space is defined by its name, an ID and a flag indicating its kind. In addition also the coordinates, upper and lower surface and the related communication frequency are listed for each airspace. Air Traffic Control (ATC) could update the ASDB through the data link, e.g. in order to activate/deactivate temporary air spaces.

A Terrain Data Base (TDB) provides Digital Terrain Elevation Data (DTED) and Digital Feature Anaylsis Data (DFAD) terrain information including the maximum height of known obstacles in a merged form. For the purpose of this project the maximum height of obstacles in a field of *30x30* meters is generated. Using this information and taking into account additional distances for safety reasons of about *100-300* meters the trajectory planning system is able to calculate the most efficient and safe route at the beginning and during a dedicated mission.

## 3 Trajectory Planning

Dynamic pre- and in-flight trajectory planning requires the accurate calculation of the most efficient 4D-route, where constraints are taken in to account. For this purpose the trajectory planning module is composed of two parts, a trajectory generator and a constraint handler.

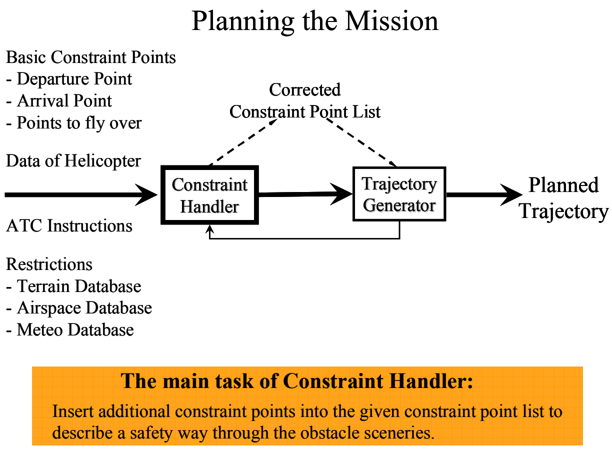


**Fig. 3: Process of Constraint Handling and Trajectory Planning**

The task to be performed by the trajectory planning can be described as the following.

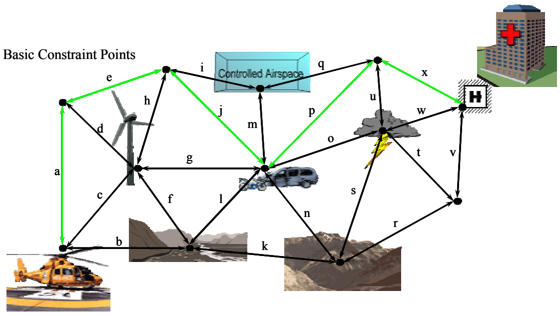


**Fig. 4: The Practical Environment of Trajectory Planning Including Constraints**

For a rescue mission the shortest way to the scene of accident and afterwards the shortest way to the suitable hospital have to be found. Those routes are very often disturbed by constraints.

This figure describes the real situation of an emergency rescue mission. For the planning at the beginning of a flight the target destination, some obstacles, an actual air traffic situation as well as the regional weather situation are known.

The mathematical task is to find the cheapest way from departure to arrival by calculating fly over points. In this example the route elements *a,e,j,p,x* define the cheapest route taking into account existing constraints.

## 3.1 Trajectory Generator

The task of the Trajectory Generator (TJG) is to create from the prepared constraint point list a feasible trajectory (list of trajectory points). It calculates all manoeuvres, which may occur during flight. These manoeuvres are acceleration, deceleration, climbs, descents, level flight, turns (fly over and fly by) and every possible combination of them (e.g. an accelerated turning climb). For the best result, it processes the forecasted weather conditions and the available helicopter performance and calculates the values for distance, time, performance, all kinds of weight, fuel, all speed types, heading, track and so on for each trajectory point. If the TJG cannot process a feasible trajectory for whatever reason, it will inform the constraint handler about the problem, which then will update the corresponding constraint points.

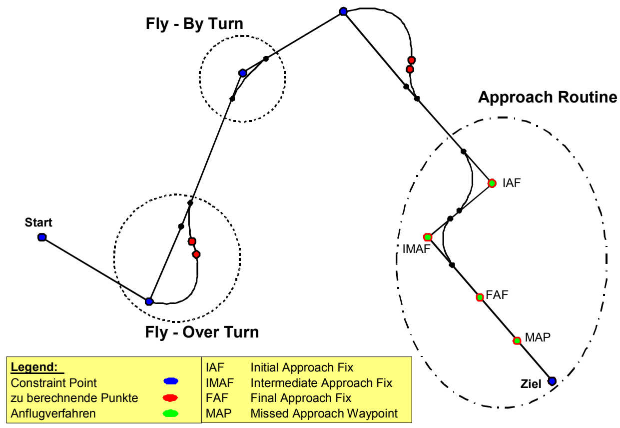


**Fig. 5: Result of a Calculated Trajectory**

The COH and TJG repeat this loop one time for each problem and, if they are not successful, they will inform the pilot through the Man-Machine-Interface (MMI) of the control and display unit (CDU) about the problem, which could not be resolved. Then it is pilot's responsibility to decide which way to go.

## 3.2 Handling of Constraints

The Constraint Handler (COH) prepares a constraint point list based on the constraints specified by the pilot or mission control. It will then pass on the constraint point list to the TJG for final trajectory generation. Its main task is to add additional constraint points if necessary to the given constraint points to establish a safe route through the obstacle sceneries of the terrain database, the airspace database and the meteorological database (with a defined upgrade path to military applications, e.g. also through the tactical enemy situation-database). This is achieved by the application of a well known mathematical algorithm, the *Dijkstra*-Algorithm, which is also used in car route planning programs. The second task of COH is to determine the take off, approach and landing flight path by using the available databases and the One-Engine-Inoperative (OEI)-Simulation performance calculation model. Finally, it will inform the pilot about irresolvable problems to achieve a solution in co-operation with him.

At the beginning of the planning the

- Departure point
- Rendezvous point (place of accident scene)
- Destination point
- Fly over points

are provided by the crew. Based on this initial data a first solution is calculated by the trajectory generator. In a second step conflicting constraints are managed by the COH. The solution to be found has to fulfil the requirement to find the direct way between two points. In order to fulfil this requirement the following aspects are considered:

- Is it possible to overcome the height difference between two way points with the performance available?
- Is there a terrain conflict on this direct route?
- Is there a restricted air space on this direct route?
- Are precalculated approach and departure procedures available for the way points considered?
- Are there additional restrictions, e.g. other areas to avoid?

The answers to these questions can be received by several strategies. In this algorithm terrain conflicts are set at highest priority. If the direct route will lead to a collision with the terrain, the flight level of the route segment will be adapted to avoid a collision with the terrain. If this is not possible, e.g. due to performance limitations of the helicopter, an additional way point is introduced to circle around the terrain obstacle. If the terrain conflict is resolved, the air space situation about restricted air spaces is investigated. Also in this case the sequence is to fly above, below or circle around. At last the weather situation is investigated following the same strategy. Further potential constraints could be added to the software in the same way, following the same inner strategy flying above, below or circle around. Regarding the type of threat itself the priority of each threat could be adapted to each principle type of mission, i.e. for military missions armed threats are assumed to be more important than airspace or weather constraints.

The overall planning starts at the critical decision point (CDP), which is defined by a minimum speed and a minimum height. For the landing segment of the route planning the landing decision point (LDP) is used as a boundary condition. For the planning of the departure phase following the starting phase the algorithm first checks a database, whether standard procedures are available. If not, the departure is planned using the principles of flight physics, the wind direction and the required heading. The route segment is checked according to obstacles stored in the terrain data base, and is adapted if necessary. For this purpose the calculated local height of each trajectory element is compared to the elevation information of the terrain data base.

If a constraint exists, the algorithm introduces a new constraint point *C* at the same level as the target point *B*. The position of this new constraint point is defined by the minimum distance to reach the same level a target point *B* before the terrain elevation.

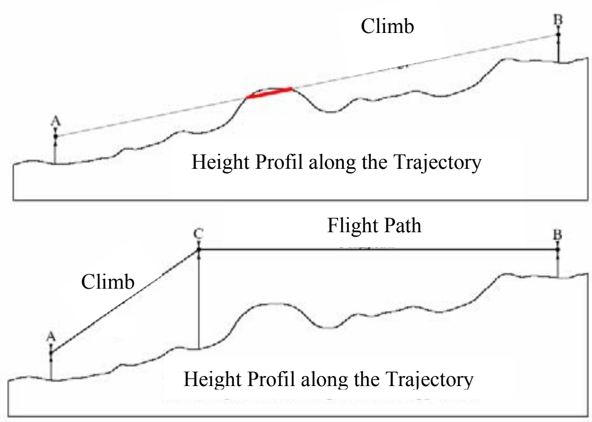


**Fig. 6: Example for Avoiding a Terrain Elevation in a Flight Path**

The process is completed by a performance calculation to ensure, that the helicopter is able to fly the segment.

There are three different cases to evade from obstacles as shown in Fig. 7.

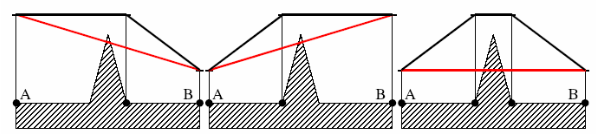


**Fig. 7: Example for Avoiding Obstacles in the Flight Path Planning**

The first two approaches consider descent and climb conditions, which are described in principle before. For the descent the new constraint point to be introduced is defined as the position behind the obstacle, the maximum descent rate and the target position *B*. The third situation is characterised by two new constraint points before and behind the obstacle. Their position and height are determined by the size of the obstacle including safety margins.
The algorithm performs these procedures to find all way points (predefined and new introduced constraints) and route segments to link the starting point, fly over and final destination point within a corridor of 10km. In the next step, the shortest route has to be identified.
In order to solve this optimization problem a geometric method is implemented based on the *Dijkstra*-algorithm to find the shortest route including the identified constraints. This kind of algorithm is used in automotive navigation systems and provides the shortest way within directed graphs. The graph *G* contains two elements: a set of nodes V and a set of edges *E*.

$$G = (V, E) \quad (1)$$

$$V = \{(i,j) : i = 1,\ldots n; j = 1,\ldots,m\}; |V| = n \cdot m \quad (2)$$

$$E = \{(v,w) : v,w \in V, v = (i_v, j_v), w = (i_w, j_w \quad (3)$$

The task to find the shortest respectively cheapest route through a well defined constraint situation is subdivided in two subtasks. At first set up the graph $G$ with its nodes $V$ and its edges $E$. A route between $u, v \in V$ is the union of sequential edges between *u* and *v*.
To jugde the edges E of a graph G the following rules are applied:

- edges between nodes with big altitude differences will get large values
- edges to nodes with forbidden obstacles will get the infinite value
- all the other edges will get the value of distance between their nodes
- if there is no connection between two nodes, the value for the edge is also infinite

All edges will get a positive real number by the function $c : E \rightarrow R^+$. This function $c(v,w)$, $v,w \in V$ describes the judging rules above between two nodes in a mathematical way.
At the first view the problem seems to be a two dimensional problem only. But the three dimensional problem can be solved with an similar graph in the same way.
Second the optimization (minimum) problem is solved using the *Dijkstra-Algorithm*. The *Dijkstra-Algorithm* based on the following definitions:

1) The distance $d(u,v)$ from node $u \in V$ to node $v \in V$ is the length of any shortest, directed way from $u \in V$ to $v \in V$. In general $d(u,v) \neq d(v,u)$. If no way exists from $u \in V$ to $v \in V$ $d(u,v) = \infty$. Furthermore $d(v,v) = 0, v \in V$.

2) The distance $d(s,T) = \min\{d(s,v) : v \in T\}, \quad T \subseteq V, s \in V \setminus T$ from node $s \in V \setminus T$ to a subset of nodes $T \subseteq V$.

The fundamental idea of *Dijkstra*-Algorithm is derived from these considerations:

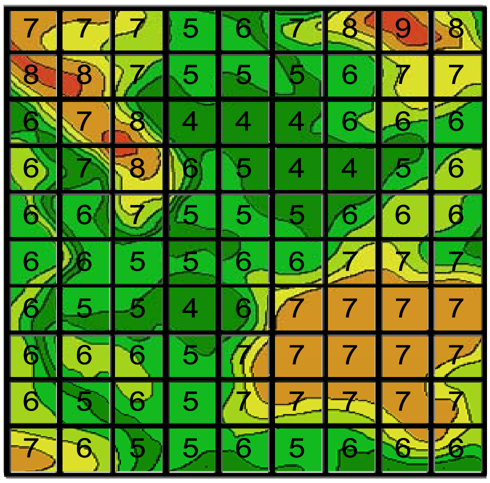


**Fig. 8: Jugded Constraint Grid**

$$d(s,T)=\min\{d(s,v)+c(v,w): v\in V\setminus T, w\in T, (v,w)\in E\} \quad (4)$$

The specification of the *Dijkstra*-Algorithm is to look at the distance between two nodes over a node its distance to the start node is well known and not to look at the distance between a node and a subset of nodes. The fact that unknown distances to nodes exist will be defined by $c(v,w)=\infty,\ \ (v,w)\notin E$. The following equation is derived:

$$\left\{\begin{array}{l} d(s,s)=\min\left\{\min\limits_{v\in V}\{d(s,v)+c(v,s)\},0\right\} \\ d(s,w)=\min\left\{\min\limits_{v\in V}\{d(s,v)+c(v,w): w\neq s\},\infty\right\} \\ s,w\in V \end{array}\right\} \quad (4)$$

For numerical calculations it advises to transfer the judged edges of the graph $G$ by function $c$ into a matrix of following form:

$$C=(c(u,v))_{u,v\in V}=\left\{\begin{array}{ccc} 0 & , & if\ u=v \\ c(u,v) & , & if\ (u,v)\in E \\ \infty & , & else \end{array}\right\} \quad (5)$$

As a result a vector of points is generated containing all way points along the shortest route. In the same way also alternate routes are calculated, which may result from such confined areas.

The planning results in a grid of judged elements, when running the process for all elements within a corridor along the route.

In the following figure a result of such a constraint analysis is presented.

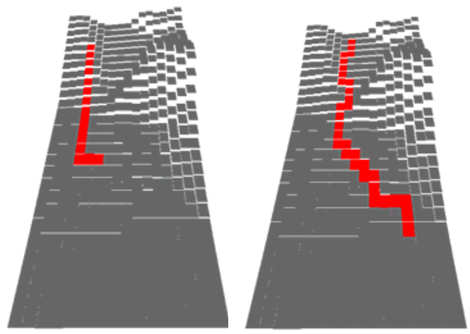

**Fig. 9: Example of a Constraint Analysis**

The Trajectory Generator (TG) generates the 4D trajectory used for flight guidance and the AFCS taking the optimised way points of the Constraint Handler. Flight performance data like minimum turn radius, weight, remaining fuel, etc. are taken into account for the final route calculation.

The mission safety is significantly improved by avoiding pre-known constraints and also time and fuel consumption are reduced, when these apriori information is used for mission planning. If new constraints may occur during a mission, such information coming from the weather station or ATC is transmitted through a data link on board. In this project the InmarSat C data link system was used. An online re-planning during the mission allows considering also such new situations, which could be adverse weather areas or new threats.

## 3 Online Obstacle Avoidance

However, due to the limited actuality of map and obstacle data bases, the potential risk to collide with new obstacles is still significant. In order to reduce this, obstacle detection sensors like "Hellas" or "MilOWS" from EADS Defence Electronics have been developed to improve the online obstacle situation awareness. If a new obstacle is detected it shall be identified and presented to crew. In addition, the flight guidance shall give support to avoid these obstacles, without loosing the original track.

This is the purpose of a second algorithm developed at Eurocopter Deutschland within the All Weather Rescue Helicopter (AWRH) project [4].
After online detection of a new obstacle an evasion route is calculated by setting first an avoidance point (AP). The position of this avoidance point is dependent from the obstacle scene to minimize the distance to travel and set close to obstacle. For obstacles like trees or windmills it is set beside the basic trajectory, for power lines a vertical manoeuvre is set. Starting point of this online calculation of the new route is the planned helicopter position 3.5 seconds ahead to take the pilots response time into account.

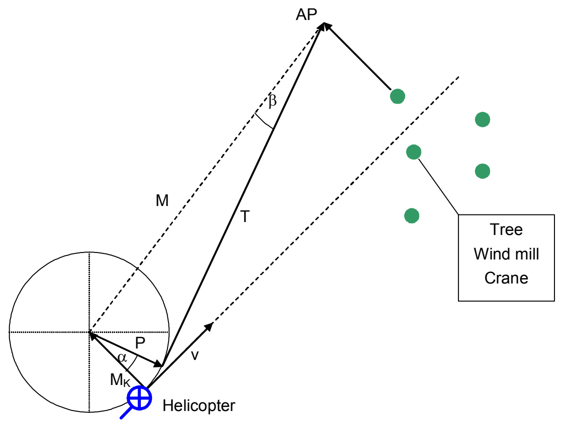


**Fig. 10: Online Obstacle Avoidance Calculation**

The algorithm to determine the avoidance route is based on a geometric method using only a circular arc and a straight line element, Fig. 10. The radius of the arc must be determined from the current speed for an average turn rate respectively an average climb rate.
First the center of the circle is determined left or right from the helicopter depending on the direction to evade.

$$\overrightarrow{M_K} = R_{\min}\left(\vec{v}^{\circ} \times \begin{pmatrix} 0 \\ 0 \\ \pm 1 \end{pmatrix}\right), \qquad (6)$$

v=vector of flight trajectroy

Secondly the tangent to the circle is calculated

$$\left|\vec{T}\right| = \sqrt{\left|\vec{M}\right|^2 + R_{\min}{}^2} \qquad (7)$$

In the third step the tangent is turned to the AP.

$$\beta = \arcsin\left(\frac{R_{\min}}{\left|\vec{M}\right|}\right) \qquad (8)$$

At last the length of the segment is determined, which has to be flown on the circle.

$$\alpha = 180° - \arccos\left(\frac{\overrightarrow{M_K} * \vec{P}}{\left|\overrightarrow{M_K}\right| * \left|\vec{P}\right|}\right) \qquad (9)$$

Having calculated the avoidance point (AP) the returning route back to the planned route is calculated in a second step defining a prediction point (PP) and the interception point (IP). The returning route contains always the same elements: Circular arc, straight line and another circular arc, Fig. 11 and Fig. 12. Here again, the current speed of the helicopter determines the turn rates or climb rates used to derive a route flyable without reducing speed and wide enough to allow smooth manoeuvres.

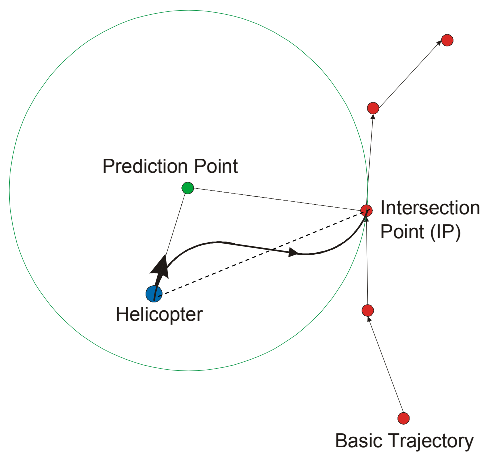


**Fig. 11: Determination of the Intersection Point**

The return to the original track could be realised by four circles $K_1$-$K_4$ theoretically. The algorithm analyses geometrically the different possible tangents, to find the most suitable. The direction of the ground track and the position of the helicopter reduce the amount of possible solutions to the feasible ones. For the remaining tangents the shortest one is selected for the generation of the intersection course.

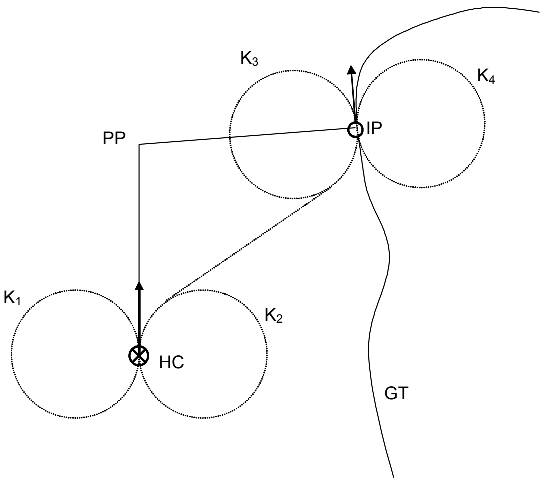


**Fig. 12: Determination of the Return Manoeuvre**

The approach to consider all possible solutions to evade and to return is very robust. An other advantage of this algorithm is the pilot´s possibility to deviate from the basic trajectory. If the algorithm recognizes the deviation it calculates a return route based on the same logic as mentioned above.

## 5 Simulation Results

Both the Trajectory Generator/Constraint Handler and the online avoidance algorithm are used to drive a “tunnel in the sky symbology” for guiding the pilot through a mission. This approach is considered to provide maximum situation awareness and controllability to the pilot, [5].
The Trajectory Generator and Constraint Handler were first implemented into the AFMS of the AWRH-Experimental helicopter. The handling was qualitatively assessed in flight by the pilots. The AFMS MMI is well suited to allow easy handling, [3]. The generation of the route was found to be robust and reliable concerning the coverage of terrain obstacles. Replanning of the route in flight was easy to handle, when the coordinates of bad weather areas or restricted airspaces were received through the data link.

The obstacle avoidance algorithm driving the “tunnel in the sky” was tested in a desktop simulation environment. The algorithm itself provides each *600ms* new data to update the tunnel guidance symbology. This update rate is mainly affected by the update rate of the OWS which is about *2Hz*. The algorithm itself provides an update rate of *25Hz (40ms)* for a maximum of 4 calculation runs per cycle.

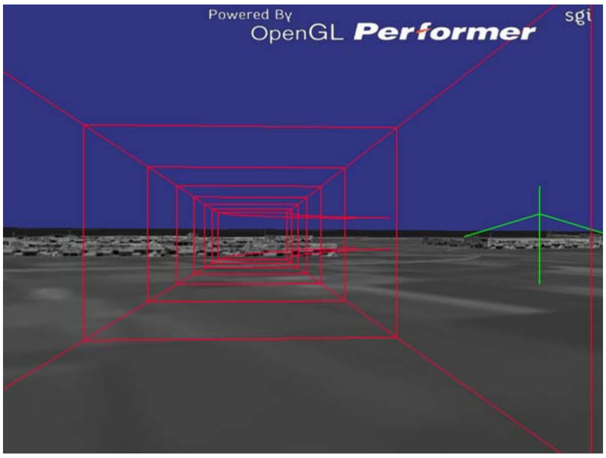


**Fig. 13: Tunnel Guidance on a Lateral Evasive Track**

The performance of the algorithm is accepted by the 3 test pilots, who evaluated the system in a first qualitative assessment.
For this simulation campaign the following mission task was given.

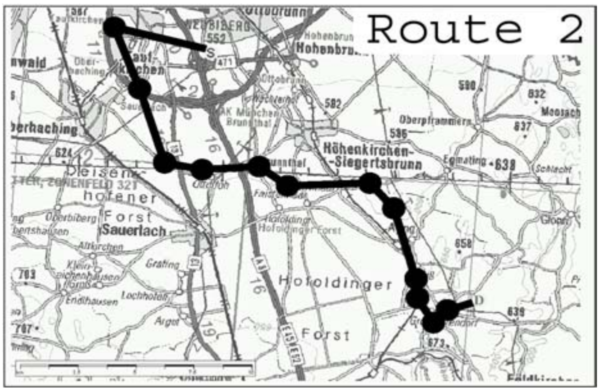


**Fig. 14: Example of a Simulation Task**

The pilot first planned the route, which is calculated by the TJG and COH. Afterwards the pilot flies the route, which is indicated by the tunnel. If a "new" obstacle occurs on the track, like the windmills indicated in the plot below, the obstacle avoidance algorithm starts.

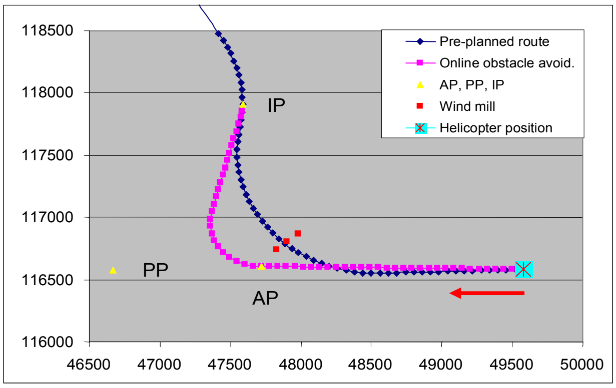


**Fig. 15: Avoidance Route**

First the avoidance point (AP) is determined to evade the obstacles. In the second step the prediction point (PP) is calculated to find the intersection point (IP) on the initial track. At last the returning track to the initial route is calculated.
The same sequence applies also, when a vertical evasion manoeuvre is to be performed.

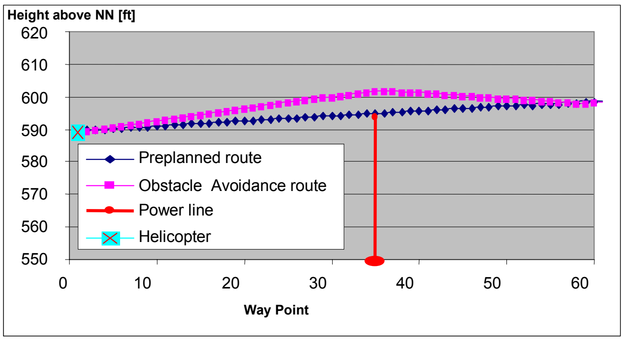


**Fig. 16: Vertical Evasion Course**

Because the online avoidance algorithm is driven by the active sensor the decision about flying over or around an obstacle is based on the lateral and vertical extension of the obstacle. If the obstacle has a defined adjustable aspect ratio (height/width) the evasion course is defined to be a lateral one.

A first campaign with 3 pilots involved has been performed to test and assess the system qualitatively.
The system is well accepted by all pilots, which have performed the trials. It was recognized, that new obstacles were quickly detected and a new trajectory is calculated in time. The pilots felt confident with the system. However, the reliability of the sensed environmental/obstacle data remains still as a critical issue.
It was also mentioned, that some "jumping effects" occurring in a complex obstacle scenario need to be resolved, see figure below.

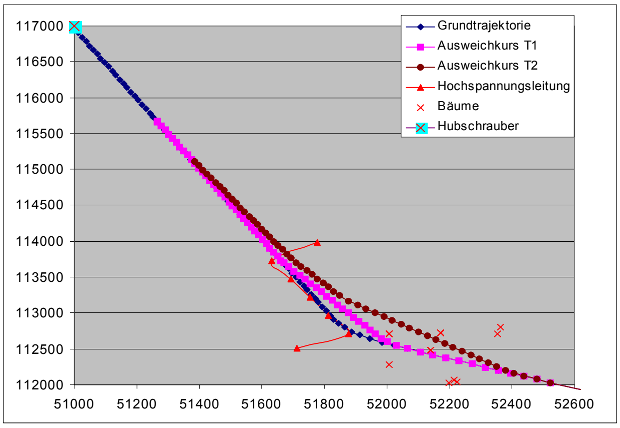


**Fig. 17: Evasion Course Sequence in a Complex Obstacle Scenario**

Also the optimisation of the avoidance trajectory with respect to the optimal routing must be mandatory: The decision in which direction the evasive manoeuvre is directed must be set with respect to the basic trajectory behind the detected obstacle.
Nevertheless first qualitative assessments have shown that the avoidance algorithm is well appropriate to leave the planned track smoothly without any jumping of the guidance aids. This is a minimum prerequisite to use such algorithms in operational systems also in flight.
In addition special focus must be set to the mandatory reliability of such calculation and visualisation systems. Also the reliability of the online obstacle detection using an active sensor must be taken into account.
The flight management and guidance investigations performed at ECD have demonstrated the potential and a useful way to

improve flight safety for modern civil and military helicopter and transport aircraft operations in an extended environment. However a reliable active sensor system to detect new obstacles along the course remains as a main technical issue.

## References

[1] V. Gollnick, K. Heidenreich, R.A. Schmidt: *Evaluation of an Advanced Display & Control Concept for Helicopter Adverse Weather Flight*, 27th European Rotorcraft Forum, Moscow, September 2001

[2] T. Butter *Development of a Way Point Managers for Automatic Trajectory Generation considering Air Traffic and Topographic Constraints*, Diplom-Thesis, University of Mittweida, September 2001, in German

[3] N.N. *AFMS Demonstrator Specification*, November 20th, 1996

[4] B. Reppelmund *Flight Guidance Algorithm for Obstacle Avoidance*, Diplom-Thesis, RWTH Aachen, November 2003, in German

[5] R. Gaffal, V. Gollnick *Comparison of Flight Guidance Symbologies for Helicopter Low Level Flight*, 29th European Rotorcraft Forum, Friedrichshafen, September 2003