Strategies for the Implementation of a Sense and Avoid System for Unmanned Aerial Vehicles

J.-B.Park[1] , P.Vörsmann[2]
Technische Universität Braunschweig, Braunschweig, Germany, 38108

The objective of this paper is to present methods appropriate for the practical implementation of a *Sense and Avoid* system for Unmanned Aerial Vehicles (UAV). The Institute of Aerospace Systems of the Technische Universität Braunschweig has been developing diverse UAVs varying in size and weight since 2001. These aircrafts can be applied for geosciences, surveillance and further various missions. In addition to present features, it is one of the major aspects of ongoing researches to realize the autonomy of such UAVs. For this purpose, information about the environment has to be transmitted to the UAV by means of adequate sensors. In this regard, a laser range finder is designated to be installed to an experimental aircraft along its longitudinal axes. With information about the distances toward detected obstacles, various methods are investigated to let the aircraft decide its collision-free path amidst obstacles autonomously. These are first the generation of feasible trajectories by means of Bézier curves. Secondly, the distance transformation for generating a map of the current environment giving an overview of endangered regions and less ones is subject of further investigations.

Nomenclature

b	=	coefficients of the Bézier curve
d	=	predefined waypoints
D	=	distance between UAV and obstacle, metric
E	=	edge of a graph
G	=	graph
k	=	number of segments of the Bézier curve
$\vec{n}$	=	normal vector of the trajectory
N	=	node of a graph
P(ν)	=	Parametric Bézier curve
$\vec{t}$	=	tangent vector of the trajectory
κ	=	curvature of the trajectory
dθ	=	drift from the original trajectory
ν	=	parameter for generating the Bézier curve

I. Introduction

THE Institute of Aerospace Systems of the Technische Universität Braunschweig has developed numerous UAVs since 2001. The developed system consists of the experimental aircraft equipped with an on-board computer for flight control and the adequate ground station software. In addition to the flight control, the on-board computer in which the navigation algorithm runs enables the aircraft to fly a predefined path automatically.[10] Parallel to this, the need for a *Sense and Avoid* system for UAVs becomes more important. The successful implementation of such systems will grant the *autonomy* of these aircrafts. This autonomy implies that static as well as dynamic obstacles can be detected and avoided independently during the mission.

[1] Visiting Scientist, Dr.-Ing. candidate, Institute of Aerospace Systems, Hermann-Blenk-Str. 23, jb.park@tu-bs.de.
[2] Head of Institute, Prof. Dr.-Ing., Institute of Aerospace Systems, Hermann-Blenk-Str. 23, p.voersmann@tu-bs.de.

Generally, in the planning phase of a mission, the operator will normally exclude routes containing obstacles. However, it is possible that the aircraft encounters unanticipated obstacles while executing its mission. In order to avoid such danger, the aircraft should be able to detect obstacles and generate a feasible trajectory.
In this regard, the main focus of this paper is the investigation of the potential to implement a *Sense and Avoid* system whose sensing instrument relies on a laser range finder solely. Subsequent sections give an overview of strategies based on the information of the laser range finder.

II. The UAV and Measurement System

The designated experimental aircraft for the investigation is the P 200 shown in Fig. 1. It has a wing span of 2 m with a maximum takeoff mass of 5 kg of which a maximum of 1 kg is planned as payload, e.g. camera for surveillance. It can operate with an average velocity of approximately 55 km/h and is able to fly up to 60 min.

In analyzing the terrain, diverse measurement systems such as radar, 2D/3D laser scanner[1,5,7,9] or ultrasonic instruments provide the airplanes or robots with information about the ranges and orientations of detected objects. Unlike conventional aircrafts or mobile robots where the capacity for sensing instruments is relatively high, the P 200 is limited in its capacity to carry sensors due to its maximum payload of 1 kg. Therefore, a laser range finder of Opti-Logic Inc. with a mass of 230 g is chosen. Besides the advantage of less weight, the sensor has the disadvantage that it measures only points instead of entire surfaces meaning that the UAV is given only one value of distance for each measurement. This characteristic will be the challenging aspect in implementing a *Sense and Avoid* system based on such laser range finder. The specification of the laser range finder is listed in Table 1.

Figure 1. Experimental UAV P 200

Power	7~9 Volts DC
Protocol	RS232
Laser Wavelength	905 nm +/- 10 nm (IR)
Laser Divergence	5 x 5 cm at 100 m
Accuracy	+/- 1 m
Measuring Range	max. 360 m
Measuring Frequency	10 ~ 200 Hz

Table 1. Specification of the laser range finder RS 400

III. Simulation Model

In order to anticipate the flight characteristic of the UAV properly, a simulation model is programmed in MATLAB.[11] On the basis of this model, the operating mode of the laser range finder and diverse strategies are to be investigated.

In most scenarios, the operator will have the information about the terrain in which the UAV is planned to operate. In such cases, the mission planning is offline and methods like potential field method, rapidly-exploring random tree (RRT), distance transformation or search algorithms are preferable. However, it is possible that unpredicted obstacles will emerge while executing the mission. For avoiding such suddenly emerging obstacles, the mission planning has to be adapted online. Therefore the program tries to model not only the offline but also the online aspect of mission planning.

The simulation model is structured as shown in Fig 2. At the beginning, a matrix including the obstacles as well as the initial flight path is generated. It is assumed that the UAV will follow the predefined trajectory while sensing its environment with the laser range finder continuously. This predefined flight path is generated by means of Bézier curves. Because the applied sensor performs rather point measurements than surface measurements as common in mobile robotics, a steady drift maneuver is initiated to acquire further information of the terrain. Once a potential obstacle is sensed, strategies like the distance transformation and the adapted flight path by means of Bézier curves are implemented to find a feasible trajectory amidst obstacles to reach the goal.

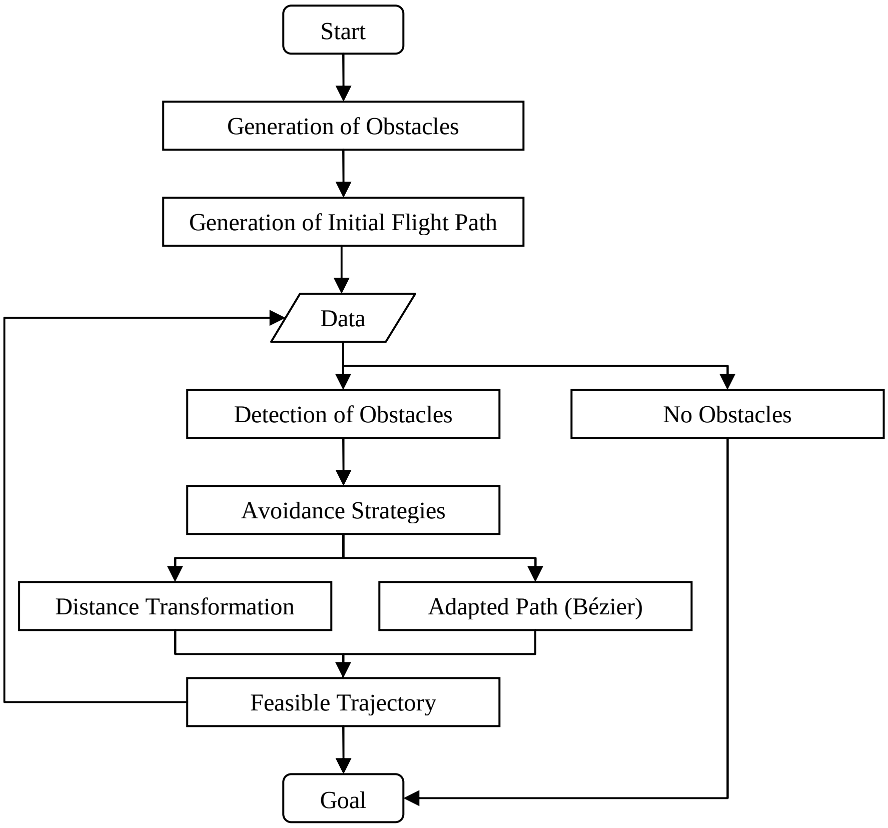

Figure 2. Structure of the simulation model

IV. Avoiding Strategies

A. Adapted Flight Path

The main aspect in applying an adaptive trajectory based on Bézier curves is to reduce the complexity in online path generation, especially when encountering dynamic obstacles. The following concept borrows the ideas from effective methods as Traffic Alert Collision Avoidance System (TCAS) and Ground Proximity Warning System (GPWS) which has been already applied in common air traffic for avoidance of collision with other aircrafts. The TCAS is conceived to alert the pilot of approaching airplanes while the GPWS alerts of crashing into the ground.

As common air traffic warning systems give the pilot recommendations, e.g. Traffic Advisory (TA) or Resolution Advisory (RA), in order to avoid possible collisions, similar recommendations can be given to the UAV in the form of adapted flight path depending on each case. Similar ideas were subjects of earlier investigation.[8]

Bézier curves were conceived in the 1960's primarily in the automobile industry for smooth shape designing.[3] There exist a number of methods for building various types of these curves. Among these types, the cubic Bézier curve is applied. The accordant equation is formulated in parametric equation (1) with $v \in [0,1]$. The coefficients b_i, $i \in [0,3]$ are the coordinates with which the curve is constructed. Similar to the different types of Bézier curves, there exist various methods in determining the coefficients b_i.[3]

$$P(v) = (1-v)^3 b_0 + 3(1-v)^2 v b_1 + 3(1-v) v^2 b_2 + v^3 b_3 \tag{1}$$

Once predefined points d_k are set, the coefficients of the Bézier curve can be determined with the equations in (2). The index *k* in equation (2) stands for the number of spline segments in the whole Bézier curve.[3]

$$
\begin{aligned}
b_0 &= \text{Initial point of the Bézier curve} \\
b_{3k-2} &= \frac{1}{3}\left(2d_{k-1}+d_k\right), \qquad k \in [1, \mathrm{m}] \\
b_{3k-1} &= \frac{1}{3}\left(d_{k-1}+2d_k\right), \qquad k \in [1, \mathrm{m}] \\
b_{3k} &= \frac{1}{6}\left(d_{k-1}+4d_k+d_{k+1}\right), \quad k \in [1, \mathrm{m\text{-}1}] \\
b_{3m} &= \text{Ending point of the Bézier curve}
\end{aligned}
\tag{2}
$$

Fig. 3 shows an example of constructing a Bézier curve. The predefined waypoints d_k, here k ∈ [1,2] standing for two consecutive segments, are represented as circles. The resulting coefficients b_k of this curve are depicted as the black filled markers.

Based on this principle, an exemplary flight path can be generated as shown in Fig. 4. The flight path begins at the coordinate (x_b,y_b) = (100,100) and ends at the coordinate (x_e,y_e) = (1000,700). The predefined waypoints are represented as circles similar to Fig 3. The resulting flight path is given as the solid line following the waypoints.

It is evident that the flight path will be generated on the basis of the information of the terrain with all *known* possible obstacles. However, it is also possible that an unpredicted obstacle may be recognized by the UAV while it fulfills its mission. For the case that such unexpected obstacles are detected along the longitudinal axes of the aircraft, various methods in constructing a maneuver for avoidance are introduced subsequently.

Firstly, depending on the actual distance D to the sensed obstacle, different maneuver can be classified as shown in Table 2. According to these distances the deviation from the original heading can be set between 5° and 15°. The result of this variation is presented in Fig.5. Depending on the distances from the actual position B (200, 200) to the sensed obstacles O1, O2 and O3, various curves denoted as I, II and III are proposed.

Secondly, the curvature at the actual coordinates of the generated trajectory can be applied for the determination of the direction of the curve. The curvature is a vector pointing to the center of a circle and its reciprocal is the radius of an osculating circle. For a parametrically defined plane curve, the curvature is determined as in equation (3).

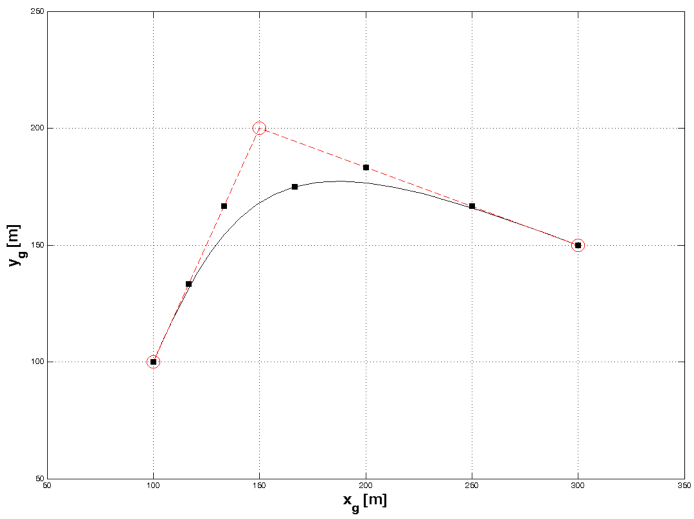

Figure 3. Scheme of constructing a Bézier curve

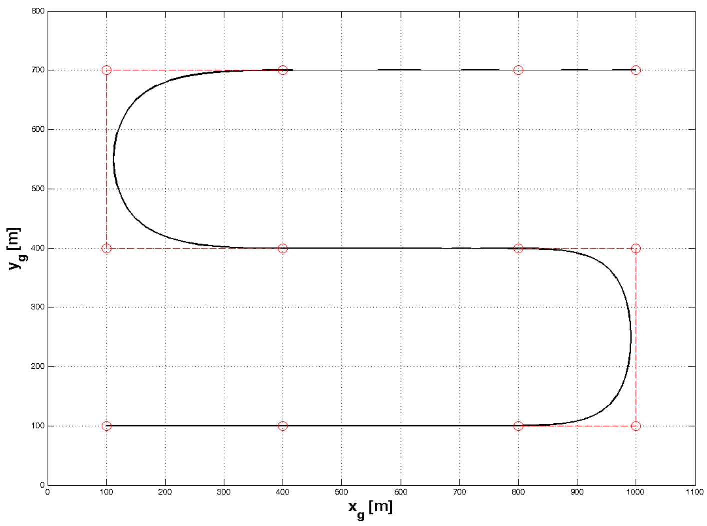

Figure 4. Exemplary generation of a flight path

Category	Distance	dθ [°]
I	D < 150 m	15
II	150 m ≤ D < 300 m	10
III	300 m ≤ D	5

Table 2. Category for various maneuver

$$\kappa = \frac{\dot{x}\,\ddot{y} - \dot{y}\,\ddot{x}}{(\dot{x}^2 + \dot{y}^2)^{3/2}} \tag{3}$$

Depending on the direction of the curve or flight trajectory, the sign of the curvature κ is either positive or negative which is shown in Fig. 6. Positive κ indicates that the concave side of the curve is in the same direction as the normal vector at this point while negative κ implies an opposite direction to the normal vector.

Based on this notation, a curve is generated to the right hand side if $\kappa > 0$ and to the left hand side if $\kappa < 0$, respectively.

Considering the above mentioned criteria, a simulated flight is presented in Fig.7. The UAV starts from its initial position (x,y) = (100,100) denoted as A. It is assumed that the measuring frequency of the laser ranger finder is set to 10 Hz and the velocity is 20 m/s. The aircraft follows its original flight path which is presented as a dashed line. The solid line represents the actual trajectory. At (x_B, y_B) = (349,100) the aircraft detects an obstacle at the position (x_{Ob1}, y_{Ob1}) = (700,100) which results in a distance of D = 351 m between the UAV and the obstacle. According to Table 2, this is the category III meaning that the aircraft has to drift 5° from its original flight path. Because the curvature κ is positive, the UAV initiate a drift of 5° to the right hand side. After moving further toward next waypoints it detects at (x_C, y_C) = (583,55) again a point at (x_{Ob2}, y_{Ob2}) = (700,50) which is declared as an obstacle. Since the curvature κ is positive and the distance toward this obstacle is D = 117 m, the drift angle is chosen to 15°. The solid line shows that the UAV is able to adapt its flight path depending on the distance toward the obstacle and the curvature of the trajectory.

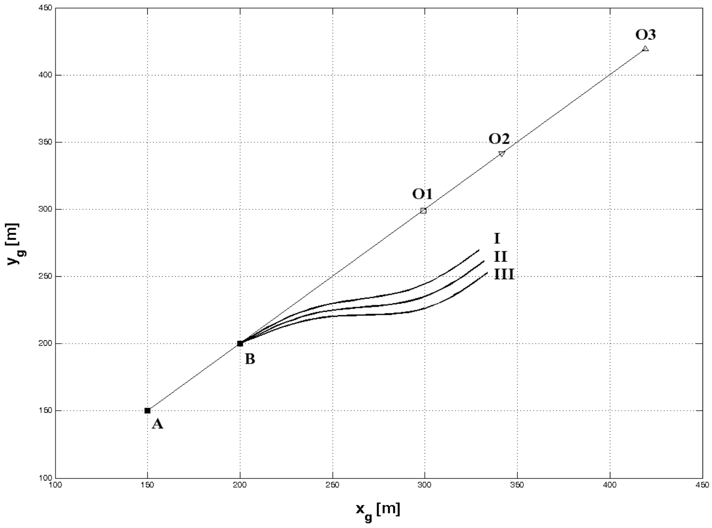

Figure 5. Types of maneuver

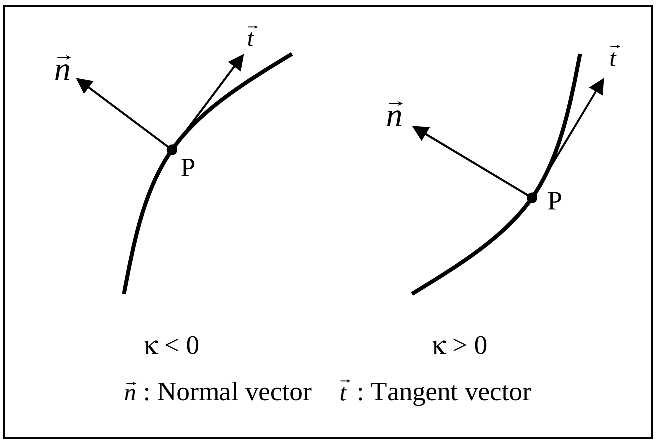

Figure 6. Sign of the curvature κ

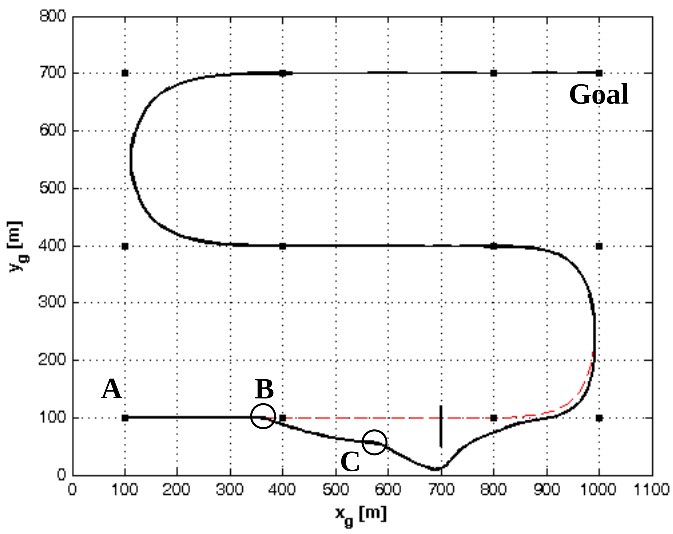

Figure 7. Adapted flight path

B. Distance Transformation

In addition to the adapted flight path generation, the distance transformation can be applied for finding a feasible trajectory. This kind of transformation is widely applied in medical image processing and informatics to build contrasting sections of investigated subjects.[6] Its main idea is to construct a reference map. For example, the mapping of bones is the subject in medical image processing while the mission environment is the main subject of the transformation in informatics and robotics. This reference map is also named graph G consisting of nodes and edges – denoted as N and E.[2,4]

Depending on the existence of obstacles, each node possesses different values. The goal of the mapping is to obtain the distances from each node or pixel to its neighbour pixels successively according to the information about occupied nodes. This transformation results in a so called occupancy map indicating endangered regions that an UAV should avoid. By applying the distance transformation, various metrics (distances) can be implemented. The difference between each metric lies in the definition of the distance between two distinct nodes. These metrics are the Chessboard distance, the City-Block distance and the Euclidean distance, given in equation (4) – (6) for two distinct nodes (x_1, y_1) and (x_2, y_2).

$$D_{chess}((x_1, y_1),(x_2, y_2)) = \max(|x_1 - y_1|,|x_2 - y_2|) \tag{4}$$

$$D_{cityblock}((x_1, y_1),(x_2, y_2)) = |x_1 - y_1| + |x_2 - y_2| \tag{5}$$

$$D_{euclidean}((x_1, y_1),(x_2, y_2)) = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2} \tag{6}$$

Compared to the Euclidean distance, the Chessboard and the City-Block distance do not seem to be optimal, because the Chessboard metric chooses the maximum of the absolute value between the horizontal and vertical distance of the nodes while the City-Block metric consider only rectangular movements between the nodes. In contrast to these metrics, the Euclidean distance applies the distance of line-of-sight which is more appropriate for the implementation.

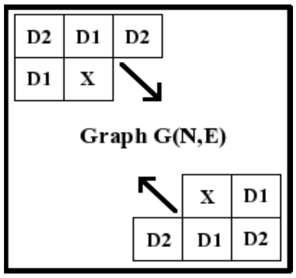

Figure 8. Scheme of the Distance Transformation

Once the metric is determined, an imaginary window which is called distance mask has to be defined. This window is built around the actual position or node X as shown in Fig.8. The scheme is to initiate a forward scan from the upper left corner to the lower right corner on the graph G firstly. Then, a backward scan in the opposite direction is carried out secondly. Thereby, D1 represents the distance in horizontal as well as the vertical direction while D2 accounts for the diagonal metric from the current node X towards its neighbours. The occupancy map for the Chessboard distance and Euclidean distance is exemplary shown in Fig. 9. The regions where an obstacle is located are represented as black areas having values 0 while the regions where no obstacles exist are marked as white areas possessing the value 1.

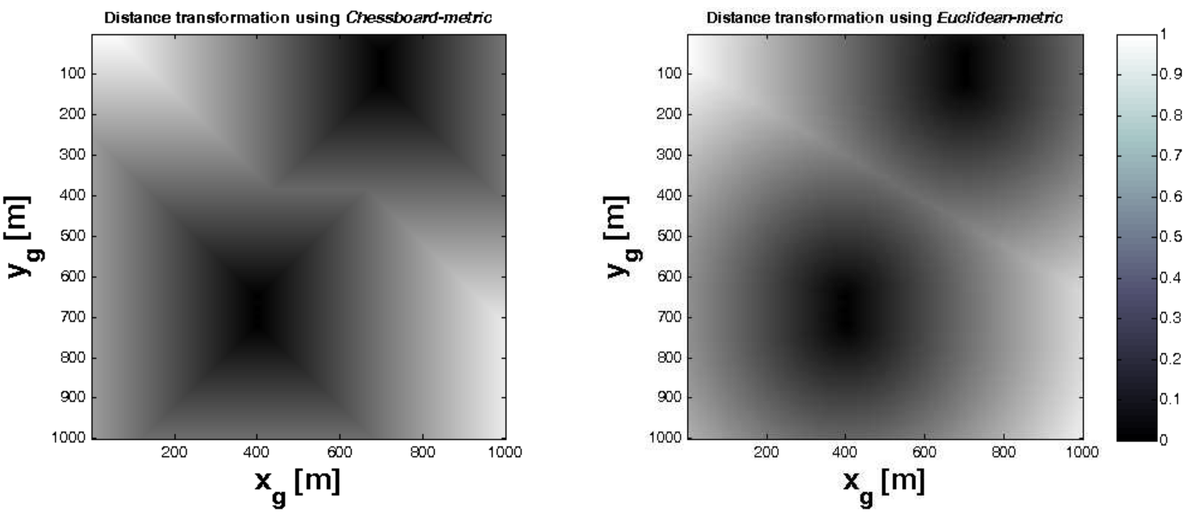

Figure 9. Occupancy maps for Chessboard and Euclidean metric

Beside this difference, the shade from the black to the white region can be interpreted as the influences of the obstacles. The darker the terrain in the occupancy map the higher is the danger of possible collision with obstacles. Therefore it is recommendable to follow the white marked areas or regions having values approximately 1 in order to avoid collision with obstacles.

In comparison to the occupancy map by means of the Euclidean metric, the mapping resulting from the Chessboard metric shows an overvaluation in the diagonal direction meaning a disadvantage in interpreting the terrain. Therefore, the distance transformation by means of Euclidean metric will be the subject of further investigation. Furthermore, it should be also investigated how fast the such transformation can be achieved during the flight to ensure the UAV not to collide with an obstacle.

V. Conclusion and Outlook

The applied methods show a potential for the implementation of a *Sense and Avoid* system. The flight path is generated initially by means of a Bézier curve. The simulation showed that the UAV detects obstacles along its longitudinal axes and reacts to this danger by selecting alternative curves depending on the distances to the obstacles and the curvature of the actual flight path.

Furthermore, the information about the locations of obstacles can be applied for generating the occupancy map. Because the distance transformation is time-consuming for the online flight path generation, an adaptive method should be investigated. This method should execute the transformation with a feasible time delay so that nodes possessing huge change in their values are considered only.

VI. Acknowledgment

J.-B. Park would like to thank Prof. Dr.-Ing. P. Vörsmann for thoroughly advising and supporting the current research at the Institute of Aerospace Systems of the Technische Universität Braunschweig and the DAAD (Deutscher Akademischer Austauschdienst) for granting the DAAD scholarship.

References

[1]Bräunl, T.: *Embedded Robotics*, Springer 2006

[2]Choset, H., et al.: *Principles of Robot Motion*, MIT Press, Cambridge, 2005, pp. 86-89, Chapter 7, Appendix H.

[3]Engeln-Müllges, et.al.: *Numerik-Algorithmen*, Springer, 2005

[4]Latombe, J.-C.: *Robot Motion Planning*, Kluwer Academic Publishers, Boston, MA, 1991, Appendix C.

[5]Langelaan, J., Rock, S.: “Navigation of Small UAVs Operating in Forests”, *AIAA Guidance, Navigation and Control Conference*, Providence, Rhode Island, 2004

[6]Cuisenaire, O.: “Distance Transformations: Fast Algorithms and Applications to Medical Image Processing”, Ph.D. Dissertation, Laboratoire de Telecomunications et Teledetection, Université Catholique de Louvain, Louvain-la-Neuve, Belgium, 1999.

[7]Beard, R.W., Saunders, J.B.: “Static and Dynamic Obstacle Avoidance in Miniature Air Vehicles”, AIAA 2005-6950, 2005.

[8]Richards, N., Sharma, M., Ward, D.: “A Hybrid A*/Automaton Approach to On-Line Path Planning with Obstacle Avoidance”, AIAA 2004-6229.

[9]Schulz, H.W., et.al.: Affordable Real Time cartography using the MAV Carolo – Limitations and Prospects, AIAA 3rd “Unmanned Unlimited” Technical Conference, Workshop and Exhibit, Chicago, 2004

[10]Schulz, H.W., et.al.: The Autonomous Micro and Mini UAVs of the CAROLO-Family, AIAA Infotech@Aerospace 2005, Virginia, 2005

[11]MATLAB, Software Package, Ver. 7.0.4.365, MathWorks Inc., Natick, MA, 2005