# Generation of Optimal Smooth Trajectories using Bezier Curves for Transport Aircraft

Hector Escamilla Nuñez, Felix Mora-Camino

# Generation of Optimal Smooth Trajectories using Bezier Curves for Transport Aircraft.

**H. Escamilla Núñez**

Laboratoire ENAC-Optim. E-mail: hector.hen91@gmail.com. Tel:(33)622416300
7 avenue Edouard Belin, Toulouse, 31055, France.

**F. Mora Camino**

Laboratoire ENAC-Optim. E-mail: felix.mora@enac.fr. Tel:(33)562174000
7 avenue Edouard Belin, Toulouse, 31055, France.

**Abstract**

With the increase in air traffic, new concepts have been introduced to organize and better manage air traffic flows (free flight, air-corridors, airstreams) with the view of increasing traffic safety and airspace capacity by solving air traffic conflicts. Hence the need of ad-hoc descriptions and parametrization of more complex and flexible transport aircraft trajectories with new characteristics, allowing high traffic densities as well as limiting environmental impact (noise).
In the first part of this paper the motivation and trajectory generation problem for transportation aircraft is introduced, and a state of the art with respect to transportation aircraft trajectory generation techniques is proposed. Then a smooth path generation based on Bezier curves is proposed.

***Keywords:*** *Flight trajectory, trajectory generation, Bezier curves, optimization, 4D guidance.*

## I. Introduction

Nowadays, commercial airliners aircraft even with operating Airborne Collision Avoidance Systems (ACAS) or Traffic Alert and Collision Avoidance Systems (TCAS), are exposed to near mid air collisions, or in worst case scenarios, to events like the one occurred on july 1 over Überlingen, Germany in 2002, where two transport-category aircraft collided after TCAS instructed one pilot to climb, but the pilot descent in compliance to air traffic controller instructions.
Moreover, even if only 1 in 10 Traffic Advisory (TAs) provided by TCAS result in a Resolution Advisory (RAs), situations like:

- January 2001, a Boeing 747 followed the instruction of the ATC to descend instead of his TCAS RA to climb, intersecting course with a DC-10. The collision was avoided when the 747 was put into a steep descent after visual contact with the other aircraft. However, about 100 crew and passengers aboard the 747 sustained injuries due the emergency maneuver.
- Switzerland, June 2011, Raytheon 390 crew followed an ATC descent clearance during their TCAS climb RA, creating a conflict with an Airbus 319. Both aircraft passed in very close proximity (0.6 nm horizontally and 50 feet vertically) without either sighting the other.
- Switzerland, May 2012, an Airbus A320 departing Zurich in a climbing turn received a TCAS RA climb caused by an AW 139 only equipped with TCAS 1 also departing Zurich. The conflict in Class 'C' airspace was attributed to inappropriate clearance issue by the TWR controller and their inappropriate separation monitoring.
- May 2013, an A319 in Swiss Class 'C' airspace received a TCAS 'Level Off' RA against a 737 above after being inadvertently given an incorrect climb clearance by ATC.

and other [1], are clear examples of human error and lack of homogeneity in TCAS.
In spite of the fact that up-to-date TCAS (III and IV) versions were capable to give Vertical and/or Horizontal direction as RAs, both were abandoned due to new arising issues. As both SESAR and NextGen projects plan to implement new operational concepts which will reduce the spacing between aircraft. TCAS in its current form is not compatible with such concepts and would alert too frequently to be useful.
Consequently, in order to allow aircraft to fly closer, quick and efficient 4D trajectory generation is crucial. In this manner, as position and time of aircraft during fly is properly handled, air traffic conflicts can be avoided or addressed if present, taking into account load factor limits to look after passengers well-being.

The generation of flyable and efficient trajectories has been considered by several authors [1], [2], [3], [4], [5], [6]. From a general view, reference trajectories are essential for flight plans which meet a large set of overfly or profile constraints, which vary in general from flight to flight. On the other hand, the generation of reference trajectories at short term should allow the implementation of procedures in the case of potential conflicts.
Current path generation for transport aircraft is based on a sequence of objective points in a 2D or 3D space given to connect two geographical locations [7]. Then, using diagrams like Voronoi, or Delaunay triangulation, or any other method, a piecewise path is constructed using straight lines denominated *legs*. This path may be offered by a high-level path planner from techniques such as Dijkstra's, A*, probabilistic roadmaps, genetic algorithms [8], or Rapidly exploring random tree star (RRT*) [9], just to mention a few.
However, as path differential requirements have to be satisfied, the initial trajectory needs to be reshaped in order to provide a flyable trajectory for transport aircraft. The problem to design a flyable path over or close the objective points while satisfying constraints such as maximum curvature and/or $G^2$ continuity

[1] https://www.skybrary.aero/index.php/Accident_and_Serious_Incident_Reports:_LOS

(curvature continuity), has been addressed using different approaches.

As a pioneer, Dubins assured that the shortest path between two points involves circles and straight line path segments [10]. However, curvature continuity at the joints of lines and circular arcs is not satisfied. Techniques to solve this track transition problem are addressed using Clothoids [11], but as they don't have closed-form expressions, the computation complexity is increased. Circular arcs as transitions tracks have been also proposed [2], where a 2D real-time trajectory is generated satisfying curvature and velocity constraints. Also, the deviation between the generated circular path and the associated control points is minimized using a parameter $\kappa \in [0, 1]$. If $\kappa = 0$, the distance is zero, so the plane will fly exactly over the waypoint, and if $\kappa = 1$, minimum-time transitions between control points are achieved. However, the election of this parameter $\kappa$ becomes an issue when a range of distance wants to be elected as a permitted deviation from the generated path.

Furthermore, Dubins like approaches generate paths limited to straight lines and arcs of circles without parametrization, thus, the method shows disadvantages being able to generate several arcs without curvature continuity or with high computation complexity.

Another tool to generate smooth flyable paths are splines, defined by series of low order polynomials. In [3], after finding the best path from a UAV position location to a target location using a Voronoi diagram and a Dijkstra's algorithm, the 2D path is smoothed using cubic splines. It is worth to say that the optimal locations of the middle knots of a spline are crucial for the shape of the segments. However, the optimal computation of these points is very time-consuming unless a set of cases are defined apriori.

As an attractive approach, Bezier curves are able to generate continuous-curvature paths, having the advantage of passing through initial and final points while the whole curve always lies within the convex hull that is constructed by the control points. Also, the initial and final piecewise straight lines of the control polygon are always tangent to the Bezier curve at the starting and ending control points. An example is given in [12], where the authors present the generation of a 3D path obtained from a combination of Rapidly exploring random tree star (RRT*) using a $C^2$ class Bezier curve, and Dynamic Movement Primitives (DMP's), that allow cooperative aerial manipulators to avoid known and unknown obstacles.

As the final Bezier curve is a barycentric combination of the polygon vertices formed by the control points, one may think that a trajectory cannot be accurately generated if the control points positions are zig-zags like, however, this aspect can be handled by stitching several Bezier curves to form a bigger path while assuring $G^2$ continuity at the joints. In [4], after a Guiding Attraction based Random Tree (GART) is used to obtain control points in a 2D plane, a kinematic smoother based on sixth order Bezier curves to achieve second derivative continuity (curvature continuity) is proposed. Finally, a local optimal reshaping of the path, minimizing length and curvature cost is performed. In [5], it is presented an algorithm based on cubic Bezier curves for 3D path smoothing, satisfying $G^2$ continuity and maximum curvature constraints, where instead of addressing a direct solution for the 3D path smoothing, a 2D path smoothing for consecutive triplets of control points is applied, seeing each triplet as a 2D plane thanks to the Frenet frame; Nevertheless, deviation from the resulting trajectory w.r.t. the control points is not controlled at all.

In this paper, assuming velocity or time constraints, a time-parametrized smooth trajectory valid for 4D guidance is developed by stitching several Bezier curves while assuring $G^2$ continuity at the joints, also, the Euclidian distance between a certain control point and the proposed trajectory is controlled, yielding an optimal 4D path as a reference trajectory.

The paper is organized as follows, Section II gives some theory on Bezier curves and conditions for $G^2$ continuity. Then the path generation is explained in Section III. In Section IV, the optimal reshaping of the path is described. Also, the obtention of the load factor for independent vertical or lateral trajectories is obtained in Section V Finally, conclusions are given in Section VI.

## II. Bezier Curves

A Bezier curve $P(s)$ of degree $n$, obtained from $n+1$ control points $(P_0, \ldots, P_n)$, is described by

$$P(s) = \sum_{i=0}^{n} B_i^n(s) P_i \qquad s \in [0,1] \tag{1}$$

where $B_i^n(s)$ is the $i^{th}$ Bernstein polynomial of degree $n$, given by

$$B_i^n(s) = \binom{n}{i} s^i (1-s)^{n-i} \qquad i \in \{0, 1, \ldots, n\} \tag{2}$$

Let a Bezier curve $C(s)$ with $m+1$ control points $(C_0, \ldots, C_m)$, and a second Bezier curve $D(s)$ with $n+1$ control points $(D_0, \ldots, D_n)$ be joined. According to [13] on $C^n$ class and regularity, $C^n$ Continuity, and $G^n$ Continuity, $C^0$ continuity is guaranteed if:

$$C_m = D_0 \tag{3}$$

Then, knowing that the first curve is tangent to the last leg, and the second curve is tangent to the first leg, a smooth transition is assured if (3) is satisfied and $C_m, D_0, C_{m-1}, D_1$ are on the same line.
Furthermore, $C^1$ continuity is guaranteed if the tangent vector of the first curve at $s = 1$ is identical to the tangent vector of the second curve at $s = 0$, meaning that:

$$C'(1) = m(C_m - C_{m-1}) = n(D_1 - D_0) = D'(0) \tag{4}$$

This states that the ratio $\left(\frac{C_m - C_{m-1}}{D_1 - D_0}\right)$, involving the length of the last leg of the first curve $(\|C_m - C_{m-1}\|)$, and the length of the first leg of the second curve $(\|D_0 - D_1\|)$, must be $\frac{n}{m}$. Since $n$ and $m$ are fixed numbers, the positions of $C_{m-1}$ and $D_1$ can be rearranged to be not only at the same line, but also at the proper distance to assure $C^1$ continuity.
In the same tenor, $C^2$ and $G^2$ continuity are guaranteed at the joint if it is verified that

$$\begin{aligned} C''(1) &= m(m-1)(C_m - 2C_{m-1} + C_{m-2}) \\ &= n(n-1)(D_2 - 2D_1 + D_0) = D''(0) \end{aligned} \tag{5}$$

Hence, to assure $G^2$ continuity, the locations of $C_m$, $C_{m-1}$, $C_{m-2}$, $D_0$, $D_1$, $D_2$, where (3)-(5) are satisfied, need to be proposed.

## III. Trajectory Generation

### A. $G^2$ Continuity Path

In order to handle the curvature while the torsion (what distinguishes a circle from a helix) is zero, the 3D path is decomposed into several 2D planar trajectories laying on the tangent/normal plane of the frenet frame, as some authors have proposed [5]. Since three control points are required to form a plane, at least $n = 3$ control points $(P_1, \ldots, P_n)$ are defined, and then divided into $n-2$ triplets. Also, in order to assure an accurate generation of the path even if the objective points are zig-zags like, several quintic Bezier curves, one for each triplet of control points, are stitched together to form a bigger path while assuring $G^2$ continuity at the joints.
For a triplet of control points $(P_1, P_2, P_3)$, like the ones depicted in figure 1, $G^1$ continuity can be achieved by interpolating four points $(Q_0, Q_1, Q_4, Q_5)$, and for $G^2$ continuity, six points are interpolated $(Q_i; i \in \{0, \ldots, 5\})$. After the interpolation of these auxiliary control points is done, in the case of the $G^2$ continuity path, a quintic Bezier curve is adjusted to them.
The points are interpolated as follows.
$Q_0$ and $Q_5$ are defined to be at the middle point of $(\bar{P_1P_2})$ and $(\bar{P_2P_3})$ respectively, forcing a past

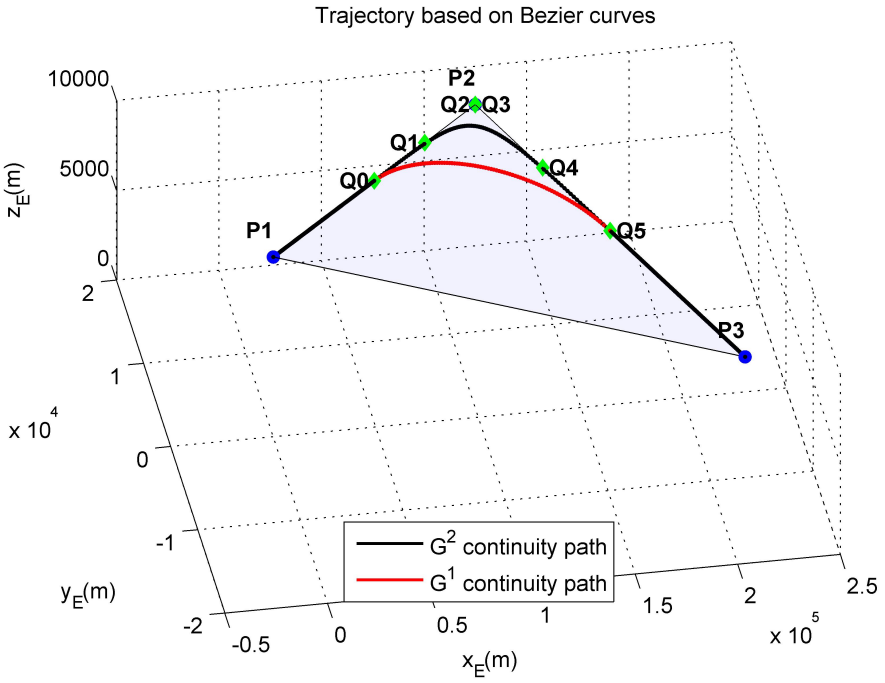


Fig. 1. Bezier curve completed with starting and ending straight lines.

and following Bezier curve formed by the past and next triplet of control points, to finish and start at $Q_0$ and $Q_5$ control points, in other words, a next triplet of points conformed by $(P_2, P_3, P_4)$ will have a Bezier curve starting at $Q_5$, so (3) is fulfilled.
For $Q_1$ and $Q_4$, they are computed to be also in the same line of $(P_1\bar{P}_2)$ and $(P_2\bar{P}_3)$ respectively, but separated by a $\delta_1$ distance from $Q_0$ in the case of $Q_1$, and a distance $\delta_2$ from $Q_5$ in the case of $Q_4$.
Note that the number of auxiliary interpolated points to compute the Bezier curves are the same for all the triplet of points, so $m$ and $n$ from (4) are equal, hence, to fulfill $G^1$ continuity, only remains to guarantee that the distance $\delta_2$ of a certain triplet of control points is equal to the distance $\delta_1$ of the next triplet of control points.
Finally, if the positions of $Q_2$ and $Q_3$ are computed also in the same line of $(P_1\bar{P}_2)$ and $(P_2\bar{P}_3)$ respectively, separated by the same distances $\delta_1$ from $Q_1$, and $\delta_2$ from $Q_4$, if it is proposed: $\delta_1 = \frac{P_1\bar{P}_2}{4}$, and $\delta_2 = \frac{P_2\bar{P}_3}{4}$.
Equations (4), (5) are satisfied, assuring $G^2$ continuity as the curvature is forced to be zero at the joints between Bezier curves.
Besides, knowing that the first and last Bezier curves conforming the total path will start and end at the middle point of their correspondent control points, as straight lines have zero curvature, the path can be completed with straight lines without affecting the $G^2$ continuity (see figure 1).

### *B. Parametrization of the path*

As several Bezier curves conform the total path, and each Bezier curve is parametrized by $s \in [0, 1]$, a time re-parametrization needs to be done for the creation of a flyable path for transport aircraft.
First, the arc length of each Bezier curve is obtained by integrating from zero to one the norm of the first derivative of the Bezier curves, as well as the initial and final straight lines that conform the total path. Once the arc lengths are obtained, a velocity for the aircraft to follow the path can be chosen, or as an alternative way, the time in which the aircraft is supposed to fly over the joints of the Bezier curves is defined. For the numerical simulation, some control points were chosen, given in Table 1.

These points generate a $G^1$ and $G^2$ path depicted in figure 2.
Regarding to the joints of the $G^2$ Bezier curves, and assuming that the path is intended to be followed at a constant $200m/s$ velocity, figure 2 shows the arc lengths $(l_i; i \in [1, 6])$ and their times $(t_j; j \in [0, 6])$ assigned. Note that these values of time, are the times at which the Bezier curves are joined. The values of $l_i$ and $t_j$ are given in Table 2.

TABLE 1
CONTROL POINTS.

| | *X(m)* | *Y(m)* | *Z(m)* |
|---|---|---|---|
| 1 | 0 | 0 | 10,000 |
| 2 | 120,843 | 16,983 | 9,300 |
| 3 | 210,332 | -14,779 | 9,000 |
| 4 | 272,744 | -759 | 8,200 |
| 5 | 388,920 | -11,130 | 9,500 |
| 6 | 478,501 | 12,964 | 9,800 |

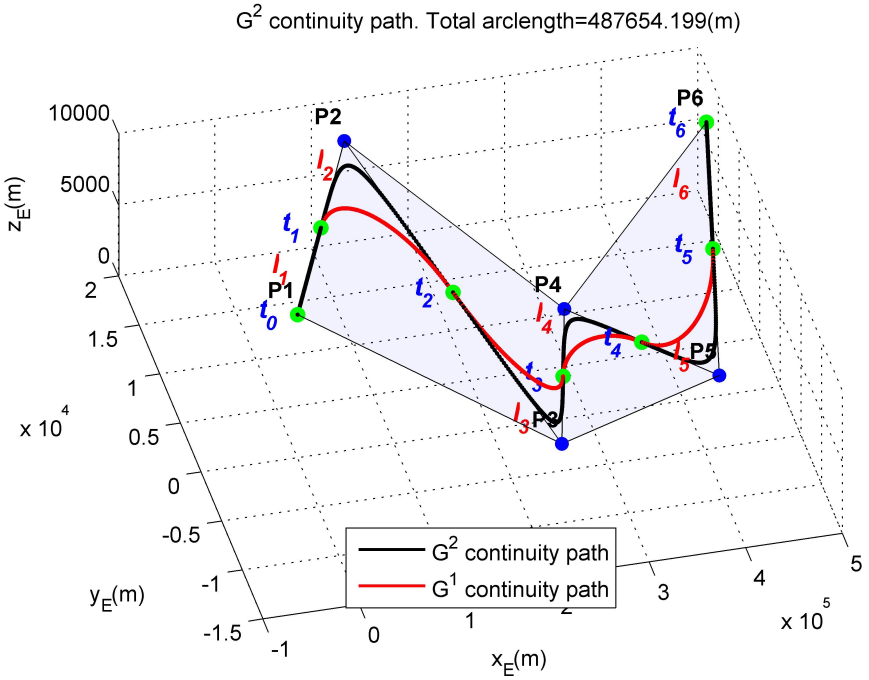


Fig. 2. Generated path showing arc lengths and joints of Bezier curves.

TABLE 2
TIMES AND ARC LENGTHS.

| *Arc length(m)* | | *Time(s)* | |
|---|---|---|---|
| | | $t_0$ | 0 |
| $l_1$ | 61,016 | $t_1$ | 305.1 |
| $l_2$ | 107,536 | $t_2$ | 842.8 |
| $l_3$ | 78,523 | $t_3$ | 1,235.4 |
| $l_4$ | 89,990 | $t_4$ | 1,685.3 |
| $l_5$ | 104,206 | $t_5$ | 2,206.4 |
| $l_6$ | 46,383 | $t_6$ | 2,438.3 |

However, as Bezier curves are not parametrized by time, an operation is performed such that:

$$s = \frac{t - t_i}{t_{i+1} - t_i} \qquad i \in [0, 1, ..., 5] \tag{6}$$

So that the multiple Bezier curves parametrized by $s \in [0, 1]$, can be used at the proper time intervals $[t_i, t_{i+1}]$. Finally, comparing the curvatures of the $G^1$ and $G^2$ continuity paths (see figure 3), it is clearly seen how the $G^2$ path has continuity in the curvature at the joints of the different Bezier curves, and that these joints occur at the assigned time, represented by asterisks in the correspondent axis.

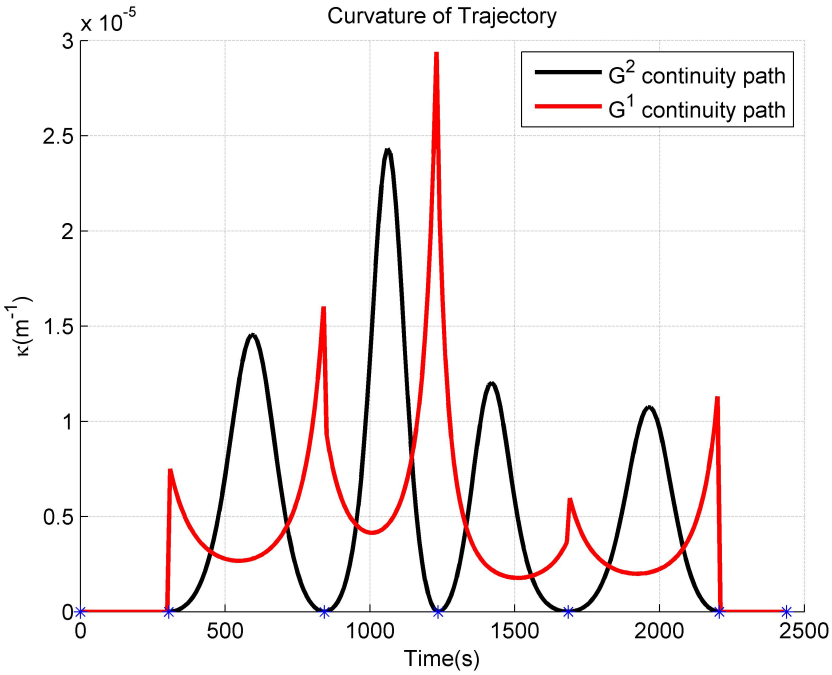


Fig. 3. Curvature of the $G^1$ and $G^2$ continuity paths.

## IV. Optimal Reshaping

Depending on the application, some aircraft may need to fly directly over the control points or at least near a defined range of distance from the control points. In order to achieve this demand, the Euclidian distance from the $G^2$ continuity path w.r.t. the nearest control point is controlled. For this special case, an extra auxiliary point ($Q_3$ in figure 4) is computed such that an optimum path based on sixth order Bezier curves is obtained. The other auxiliary control points are conserved as they are in order to do not affect curvature continuity, but the position of the heptic auxiliary point will be moved gradually until the generated path passes within a distance range defined by the user. The direction in which this point is moved will be denoted by

$$a\vec{u}x = \frac{(Q_2 \vec{-} Q_1) + (Q_4 \vec{-} Q_5)}{\|(Q_2 \vec{-} Q_1) + (Q_4 \vec{-} Q_5)\|} \tag{7}$$

In figure 4, a $100m$ maximum deviation for a triplet of control points is commanded, meaning that the optimized path will be generated such that it will pass no further than $100m$ away the control point $P_2$. In this case, the optimizing algorithm stopped at a distance of $92.48m$. The distance at which the algorithm stops depends on the step size in which $Q_3$ is moved away in the direction of $a\vec{u}x$. For a

small step size, the path will be closer to the distance defined by the user, in this case $100m$, but the computation time will be increased.

Now, for a complete trajectory, using the control points of Table 1, a $100m$ maximum deviation is

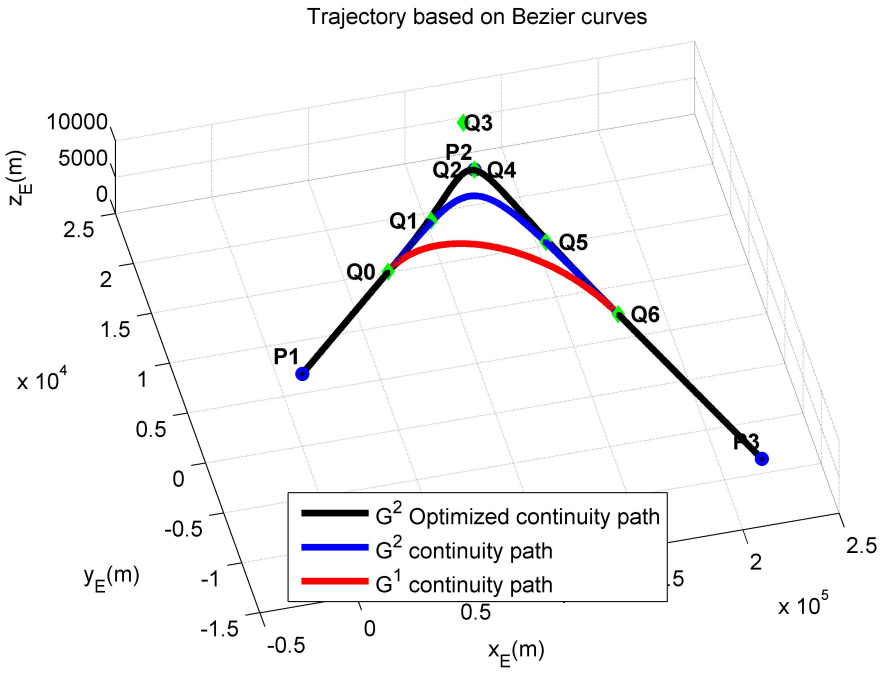


Fig. 4. $100m$ deviation optimized path for a triplet of control points using an auxiliary control point ($Q_3$).

commanded, the resultant path is shown in figure 5.

The Euclidian distance at which the trajectory is generated from the closest control points is given by Table 3. Assuring that the maximum deviation distance is achieved.

As close as the optimal path is to the control points, the curvature becomes bigger, such that a

TABLE 3

DISTANCE OF CURVE FROM CONTROL POINTS.

| *From* | $P_2$ | $P_3$ | $P_4$ | $P_5$ |
|---|---|---|---|---|
| *Distance(m)* | 92.4823 | 99.7277 | 96.0905 | 79.5305 |

maximum curvature could also be defined and controlled by the deviation distance defined by the user, establishing a tradeoff between deviation of the path from the control points and maximum curvature constraints. Furthermore, as the generated trajectory is time-parametrized, it could be used for 4D guidance problems.

## V. LOAD FACTOR OF THE TRAJECTORY

As transport aircraft are designed to flight smooth trajectories, some limitations on the load factor have to be respected for the well being and comfort of passengers. According to [14], on a typical flight, the load factor is limited to +2.5g and -1g for regular maneuvers, or up to 3.8g in maximum takeoff weight. The expressions for the load factor (in the body frame) of an aircraft are given by

$$n_x = -s_\theta + (rv - qw)/g \tag{8}$$

$$n_y = c_\theta s_\phi + (pw - ru)/g \tag{9}$$

$$n_z = c_\theta c_\phi + (qu - pv)/g \tag{10}$$

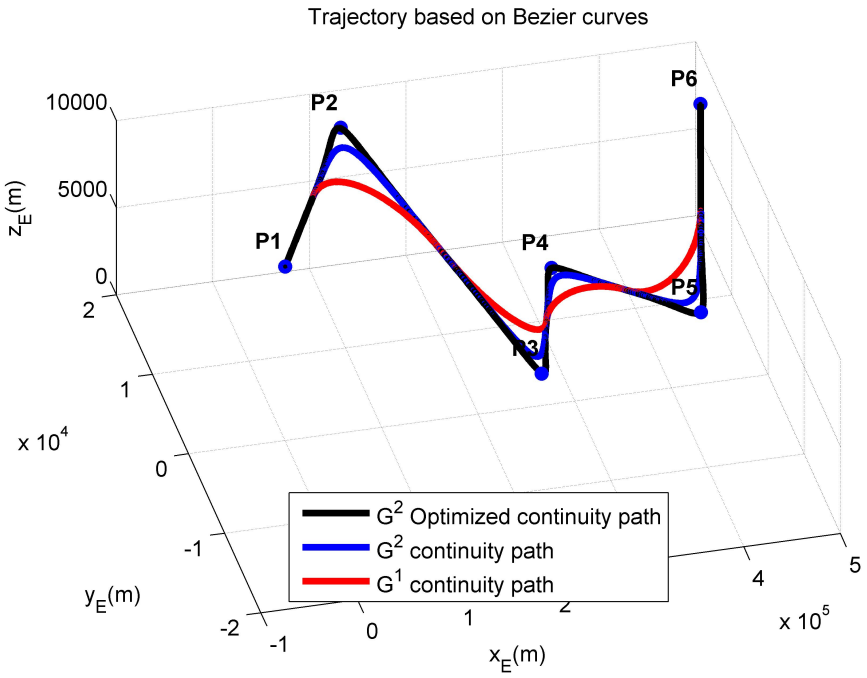


Fig. 5. $100m$ deviation optimized path.

Under the assumptions of a steady turn, it can be obtained that

$$n_z = \frac{c_\theta}{c_\phi} \tag{11}$$

Note that the bigger value of $n_z$ is obtained when $\theta = 0°$, so, for a 2.5g limit, the maximum bank angle is $66.5°$.
Also, an analysis involving the centrifugal acceleration yields

$$\frac{mV_a^2}{R} = Ls_\phi \tag{12a}$$

$$Lc_\phi = W \tag{12b}$$

thus, (12b) can be rewritten as

$$\frac{L}{W} = \frac{1}{c_\phi} = n_z \tag{13}$$

which is also when $\theta = 0°$ in (11), then, using this relation into (12a), the load factor can be related with the radius of horizontal curvature $R$ as

$$R = \frac{1}{n_z}\frac{V_a^2}{gs_\phi} \tag{14}$$

Also, using (13) and the Pythagorean identity we obtain

$$s_\phi = \sqrt{1 - \frac{1}{n_z^2}} \tag{15}$$

Then (14) is expressed completely in terms of the load factor as

$$R = \frac{V_a^2}{g\sqrt{n_z^2 - 1}} \tag{16}$$

Arriving to an expression relating the load factor and speed of an aircraft with the horizontal path of radius $R$ that corresponds to these variables.
On the other hand, if a pitch up motion is assumed, the considerations of $\phi = p = 0$, $u = V_a$ and $\theta = 0^\circ$ for a maximum load factor, yields to

$$n_z = 1 + \frac{qV_a}{g} \tag{17}$$

which can be expressed as

$$q = \frac{(n_z - 1)\, g}{V_a} = \frac{V_a}{R'} \tag{18}$$

where $R'$ is the radius of vertical curvature.
In this manner, as the curvature is the inverse of the radius, a load factor for independent lateral or vertical maneuvers performed by the aircraft can be obtained.
In this way, following the load factor limits, and depending on the speed of the aircraft, a maximum radius path can be computed, and in case that the radius of the path needs to be increased to achieve more accuracy in the deviation of the desired path from the control points, speed of the aircraft can be decreased to attend the issue.

## VI. Conclusions

In this paper, a formal approach for the design of 4D smooth trajectories for commercial aircraft has been developed. The proposed approach is based on the use of a particular family of curves, the Bezier curves, which are built from control points. These control points are shown to be related with path constraints issues as well as flyability constraints.
The adoption of these trajectories should ease air traffic management in congested areas, as well as improve the performance of on-board guidance systems.

## References


[1] E. Bakolas, Y. Zhao, and P. Tsiotras, "Initial guess generation for aircraft landing trajectory optimization," *AIAA Guidance, Navigation, and Control Conference, Portland, Oregon, USA*, August 2011.
[2] E. P. Anderson, R. W. Beard, and T. W. McLain, "Real-time dynamic trajectory smoothing for unmanned air vehicles," *IEEE Transactions on Control Systems Technology*, vol. 13, no. 3, May 2005.
[3] K. B. Judd and T. W. McLain, "Spline based path planning for unmanned air vehicles," *AIAA Guidance, Navigation, and Control Conference and Exhibit, Montreal,Canada*, August 2001.
[4] L. Yang, D. Song, J. Xiao, J. Han, L. Yang, and Y. Cao, "Generation of dynamically feasible and collision free trajectory by applying six order bezier curve and local optimal reshaping," *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Hamburg, Germany*, 28-2 September-October 2015.
[5] K. Yang and S. Sukkarieh, "An analytical continuous-curvature path-smoothing algorithm," *IEEE Transactions on Robotics*, vol. 26, no. 3, June 2010.
[6] D. Delahaye, S. Puechmorel, P. Tsiotras, and E. Feron, *Mathematical Models for Aircraft Trajectory Design: A survey. In: Electronic Navigation Research Institute (eds) Air Traffic Management and Systems. Lecture Notes in Electrical Engineering*. Springer, 2014, vol. 290.
[7] R. Walter, *Digital Avionics Handbook. Chapter 24*, 3rd ed. CRC Press, 2015.
[8] O. K. Sahingoz, "Generation of bezier curve-based flyable trajectories for multi-uav systems with parallel genetic algorithm," *Journal of Intelligent and Robotic Systems*, vol. 74, no. 1-2, pp. 499–511, April 2014.
[9] D. J. Webb and J. van den Berg, "Kinodynamic rrt*: Asymptotically optimal motion planning for robots with linear dynamics," *2013 IEEE International Conference on Robotics and Automation (ICRA), Karlsruhe, Germany*, May 2013.
[10] L. E. Dubins, "On curves of minimal length with a constraint on average curvature, and with prescribed initial and terminal positions and tangents," *American Journal of Mathematics*, vol. 79, no. 3, pp. 497–516, July 1957.
[11] A. Scheuer and T. Fraichard, "Continuous-curvature path planning for car-like vehicles," *Proceedings of the 1997 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, September 1997.
[12] H. Lee, H. Kim, and H. J. Kim, "Planning and control for collision-free cooperative aerial transportation," *IEEE Transactions on Automation Science and Engineering*, vol. PP, no. 99, pp. 1–13, October 2016.
[13] B. A. Barsky and T. D. DeRose, "Geometric continuity of parametric curves. technical report no. ucb/csd 84/205," *University of Berkeley, USA*, October 1984.
[14] *Code of Federal Regulations (CFR)-Title 14 Aeronautics and Space*, Federal Aviation Administration (FAA), May 2017.