

Curvature Continuous Path Generation for UAVs based on Generalized Expo-Rational B-Splines

Tom Stian Andersen

Department of Electrical Engineering
UiT The Arctic University of Norway
NO-8514 Narvik, Norway
Email: tom.s.andersen@uit.no

Ghada Bouzidi

Department of Computer Science
and Computational Engineering
UiT The Arctic University of Norway
NO-8514 Narvik, Norway
Email: gbo013@post.uit.no

Raymond Kristiansen

Department of Electrical Engineering
UiT The Arctic University of Norway
NO-8514 Narvik, Norway
Email: raymond.kristiansen@uit.no

Abstract—This paper proposes a new path-planning algorithm based on GERBS to generate curvature continuous paths. The algorithm blends between a set of local curves generating a global smooth path which also interpolates the midpoint of the curves. Numerical examples are provided which demonstrates the potential of the proposed algorithm.

I. INTRODUCTION

In the last decades there has been significant improvements in sensor technology and processing power of embedded systems which has opened up the possibility of designing small and cheap autonomous aircrafts. The potential applications for these autonomous vehicles are vast, but the increased autonomy means that robust and reliable algorithms need to be developed to ensure that they operate safely. An important part of an unmanned aerial system is the path generation subsystem, which should generate a continuous path that either approximates or interpolates a set of waypoints. The properties of the path such as curvature has a direct influence on the control forces applied to the UAV where a rapid change in curvature such as discontinuities will cause a large control effort that might be damaging for the UAV.

The simplest path between a set of waypoints is made by connecting straight lines between the waypoints which is often used in waypoint tracking applications, but this generates a curve that is only C^0 smooth which results in a path that can not be followed by non-holonomic vehicles such as aircrafts without large deviations from the path near the waypoints. An interpolating path that achieves C^1 continuity where circular arcs defined at the waypoint locations are connected together by straight lines is known as a Dubin's path and were studied in [1]. They are C^1 because the curvature is discontinuous at the transition between the straight lines and the arcs. It is however often used in UAV applications because of its simplicity and low computational cost, and the fact that it provides the shortest path between two poses for a non-holonomic vehicle [2].

Despite its wide usage there has been considerable efforts in developing curvature continuous paths to replace or com-

plement Dubin's paths for non-holonomic vehicles. In [3] single polar-polynomials were proposed to generate smooth arc turns and single cartesian-polynomials to generate lane changing curves. In [4] Fermat's Spirals were used to generate smooth arc turns with straight lines to create curvature continuous paths that approximate a set of waypoints. It was also demonstrated that Fermat's Spirals can be used to join together lines and constant curvature arcs resulting in Dubin's smoothed paths. In [5] several methods are shown for generating smooth paths between waypoints such as clothoids [6] and pythagorean hodographs [7]. Another path generation technique is B-splines which was implemented in [8] where an rapidly exploring random trees (RRT) algorithm is used to generate the control points for the B-spline. In [9] cubic B-splines were used to construct curvature continuous paths with upper bounded curvature. In [10] generalized dubin's paths are approximated by third order algebraic polynomial splines. In [11] Cubic Bézier Splines are used to generate G^2 paths with constrained curvature.

In this paper a new path generation algorithm is proposed based on Generalized Expo-Rational B-Splines (GERBS) which are a new type of B-splines first presented in [12]. The main motivation for applying GERBS to the path generation problem is the versatility in how they can be used together with the geometric properties they offer.

This paper is organized in the following way. Section II introduces the notion of Expo-Rational B-Splines and some of their properties, then a generalized version named Generalized Expo-Rational B-Splines (GERBS) is introduced where a basis function based on Beta functions is used in favour of the Expo-Rational basis function. Section III details the path generation algorithm using a second order GERBS with Beta function as its basis function. Section IV shows numerical examples of the proposed path generation algorithm using Beta function basis functions and Expo-Rational basis functions and it is compared to Dubin's paths. In Section V a brief conclusion is given highlighting some future work.

II. PRELIMINARIES

A. Expo-Rational B-Splines

A parametric curve can be represented using ERBS as [12]

$$\mathbf{p}(t) = \sum_{i=1}^n \mathbf{c}_i(t) B_i(t) \quad (1)$$

where $\mathbf{c}_i(t)$ is a set of local curves, $B_i(t)$ denotes the exponential basis functions defined later and $t \in (t_1, t_{n+2}]$ with $\{t_j\}_{j=1}^{n+2}$ defining the knot sequence. This differs from classical B-splines where \mathbf{c}_i usually denotes the control points that are blended and in this context ERBS can be viewed as a blending of local curves. A second order ERBS can be written in B-spline form as

$$\mathbf{p}(t) = [1 - (B \circ w_{1,i})(t) \quad (B \circ w_{1,i})(t)] \begin{bmatrix} c_{i-1}(t) \\ c_i(t) \end{bmatrix} \quad (2)$$

where indices are defined through the relation $t_i < t < t_{i+1}$. By introducing the basis functions in the B-spline formulation it is possible to increase the continuity of the B-spline without increasing the support. This is in contrast to normal B-splines where a second order B-spline is C^1 continuous, while a second order ERBS is C^∞ continuous. The continuity of a B-spline is also dependent on the multiplicity of knots in the knot sequence in addition to the continuity of the basis function. For a B-spline of degree d with knot multiplicity r at a knot t_k the continuity of the B-spline at t_k will be C^{d-r} . Since it is not necessary to increase the degree of an ERBS to increase the curves continuity it is therefore possible to keep the interpolating property of second order B-splines while still achieving high continuity. The Expo-Rational Basis Function is defined as [12]

$$B_k(t) = \begin{cases} \int_{t_{k-1}}^t \Psi_{k-1}(s) ds & t_{k-1} < t \leq t_k \\ 1 - \int_{t_k}^t \Psi_k(s) ds & t_k < t \leq t_{k+1} \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

where

$$\Psi_k(t) = S_k^{-1} \cdot e^{-\beta_k \frac{|t - (1-\lambda_k)t_k + \lambda_k t_{k+1}|^{2\sigma_k}}{((t-t_k)(t_{k+1}-t)^{\gamma_k})^{\alpha_k}}} \quad (4)$$

and

$$S_k = \int_{t_k}^{t_{k+1}} e^{-\beta_k \frac{|s - (1-\lambda_k)t_k + \lambda_k t_{k+1}|^{2\sigma_k}}{((s-t_k)(t_{k+1}-s)^{\gamma_k})^{\alpha_k}}} ds \quad (5)$$

with $\alpha_k > 0$, $\beta_k > 0$, $\gamma_k > 0$, $1 \geq \lambda_k \geq 0$, $\sigma_k \geq 0$ and $k = 1, \dots, n$ as intrinsic parameters that can be used to tune the basis function. Given an increasing knot vector $\{t_j\}_{j=1}^{n+2}$ the support for $B_k(t)$ lies in $[t_{k-1}, t_{k+1}]$ and $B_k(t) \in C^\infty$. If the internal knots are simple without multiplicity then the curve $\mathbf{p}(t)$ will be C^∞ smooth while interpolating the local curve $\mathbf{c}_k(t)$ at $t = t_k$. Calculating the value of $B_k(t)$ requires approximating it using numeric integration. This initially increases the computational burden of ERBS with regards to B-splines, but several optimizations can be utilized to still make the evaluation of ERBS efficient [13].

The choice of local curves are restricted in the sense that all derivatives of $\mathbf{c}_k(t)B_k(t)$ is zero when $t \in \{t_{k-1}, t_{k+1}\}$ which is a very weak restriction since most curves will satisfy it, for instance straight lines, circular arcs and Bézier curves. Note that from (3) the basis function is only non-zero over a three knot interval which gives ERBS strong local support which implies that changing the position of a local curve has only local effect on the entire curve $\mathbf{p}(t)$. This property is advantageous in cases where changes to the location of a waypoint is needed or a new waypoint needs to be added. The locality property ensures that these type of changes to the path does not require complete reevaluation of the entire path.

B. Generalized Expo-Rational B-Splines

In [14] the notion of ERBS was generalized as

$$G_j(t) = \begin{cases} F_j(t) & t \in (t_{j-1}, t_j) \\ 1 - F_{j+1}(t) & t \in (t_j, t_{j+1}) \\ 0 & \text{otherwise} \end{cases} \quad (6)$$

where $F_j(t)$ is monotonously increasing and left-continuous on the interval $[t_{j-1}, t_j]$, $F(t) = 0, \forall t < t_{j-1}$ and the left-hand limits satisfy $F(t_{j-1}+) = F(t_{i_1}) = 0$ and $F(t_i+) = F(t_i) = 1$. An interesting class of GERBS are the Beta-Function B-Splines (BFBS) which have most of the same properties of ERBS but do not need numerical integration for its evaluation, making them more computationally efficient. The Beta-Function basis function is defined as [15]

$$B_k(t) = \begin{cases} S_{k-1} \int_{t_{k-1}}^t \Psi_{k-1}(s) ds & t_{k-1} < t < t_k \\ S_k \int_t^{t_{k+1}} \Psi_k(s) ds & t_k < t < t_{k+1} \\ 1 & t = t_k \\ 0 & \text{otherwise} \end{cases} \quad (7)$$

where

$$S_k(t) = \left(\int_{t_k}^{t_{k+1}} \Psi_k(t) dt \right)^{-1} \quad (8)$$

and

$$\Psi_k(t) = \binom{i_k + i_{k+1}}{i_k} \frac{(t - t_k)^{i_k} (t_{k+1} - t)^{i_{k+1}}}{(t_{k+1} - t_k)^{i_k + i_{k+1}}} \quad (9)$$

where $i_l > 0, l \in \{k-1, k, k+1\}$. The smoothness of the BFBS is dependent on i_k meaning that the BFBS is C^m where $m = \min_{k \in [1, n]} i_k$, [16]. For the case when $i_k = 2, \forall k \in [1, n]$ a curvature continuous path can be constructed where

$$\Psi_k(t) = \frac{6(t - t_k)^2 (t_{k+1} - t)^2}{(t_{k+1} - t_k)^4} \quad (10)$$

which means that the integrals in (7) can be solved analytically avoiding the need for approximate algorithms. By evaluating

the integrals the Beta-Function basis function can be expressed as

$$B_k(t) = \begin{cases} \Phi(t_{k-1}, t_k, t) & t_{k-1} < t < t_k \\ \Theta(t_k, t_{k+1}, t) & t_k < t < t_{k+1} \\ 1 & t = t_k \\ 0 & \text{otherwise} \end{cases} \quad (11)$$

where

$$\begin{aligned} \Phi(t_{k-1}, t_k, t) &= \frac{(t_{k-1} - t)^3}{(t_{k-1} - t_k)^5} \\ ((t_{k-1} - 5t_k + 3t)t_{k-1} + (10t_k - 15t)t_k + 6t^2) \end{aligned} \quad (12)$$

and

$$\begin{aligned} \Theta(t_k, t_{k+1}, t) &= \frac{(t - t_{k+1})^3}{(t_k - t_{k+1})^5} \\ ((t_{k+1} - 5t_k + 3t)t_{k+1} + (10t_k - 15t)t_k + 6t^2). \end{aligned} \quad (13)$$

The derivatives of the Beta-Function basis function are

$$D\Phi = DA_\Phi B_\Phi + A_\Phi DB_\Phi \quad (14)$$

$$DD\Phi = DDA_\Phi B_\Phi + 2DA_\Phi DB_\Phi + A_\Phi DDB_\Phi \quad (15)$$

and

$$D\Theta = DA_\Theta B_\Theta + A_\Theta DB_\Theta \quad (16)$$

$$DD\Theta = DDA_\Theta B_\Theta + 2DA_\Theta DB_\Theta + A_\Theta DDB_\Theta \quad (17)$$

where

$$\begin{aligned} A_\Phi &= \frac{(t_{k-1} - t)^3}{(t_{k-1} - t_k)^5} \\ B_\Phi &= (t_{k-1} - 5t_k + 3t)t_{k-1} + (10t_k - 15t)t_k + 6t^2 \\ DA_\Phi &= \frac{-3(t_{k-1} - t)^2}{(t_{k-1} - t_k)^5} \\ DB_\Phi &= 3t_{k-1} - 15t_k + 12t \\ DDA_\Phi &= \frac{6(t_{k-1} - t)}{(t_{k-1} - t_k)^5} \\ DDB_\Phi &= 12 \\ A_\Theta &= \frac{(t - t_{k+1})^3}{(t_k - t_{k+1})^5} \\ B_\Theta &= (t_{k+1} - 5t_k + 3t)t_{k+1} + (10t_k - 15t)t_k + 6t^2 \\ DA_\Theta &= \frac{3(t - t_{k+1})^2}{(t_k - t_{k+1})^5} \\ DB_\Theta &= 3t_{k+1} - 15t_k + 12t \\ DDA_\Theta &= \frac{6(t - t_{k+1})}{(t_k - t_{k+1})^5} \\ DDB_\Theta &= 12. \end{aligned} \quad (18)$$

These are used to evaluate the curvature of the path and as seen from the BFBS and its derivatives the curvature will mostly depend on the choice and parameterization of the local curves. The analytic expressions for the Beta-Function basis functions allows for precomputation and storing of

terms that only depend on the knot values which leads to good computational efficiency. Additionally since second order BFBS have strong local support changing the position of a waypoint only requires recomputation of the connected path segments. Another interesting property is the ability to have different continuities on path segments by having different basis functions for each k in (7).

III. GERBS BASED PATH GENERATION

Given a set of n waypoints $\{\mathbf{W}_i\}_{i=1}^n \in \mathbb{R}^2$ define a set of unit tangent vectors $\{\mathbf{m}_i\}_{i=1}^n \in \mathbb{R}^2$ and a set of turning directions $\{s_i\}_{i=1}^n \in \{-1, 1\}$ using Algorithm 1 corresponding to the desired orientation and turn at each waypoint. Define a set of circular arcs for each waypoint $\mathbf{W}_i = [W_x \ W_y]^T$, and tangent $\mathbf{m}_i = [m_x \ m_y]^T$ and turning direction s_i with $i = 2, \dots, n-1$ as

$$\mathbf{p}_{arc}(t) = \begin{bmatrix} x_c + R \cos((1-t)\theta_1 + t\theta_2) \\ y_c + R \sin((1-t)\theta_1 + t\theta_2) \end{bmatrix} \quad (19)$$

where $\mathbf{p}_c = [x_c \ y_c]^T$ is the center of the arc defined as

$$\mathbf{p}_c(t) = \begin{bmatrix} W_x + R \cos(\arctan 2(m_y, m_x) + s_i \frac{\pi}{2}) \\ W_y + R \sin(\arctan 2(m_y, m_x) + s_i \frac{\pi}{2}) \end{bmatrix} \quad (20)$$

with $s \in \{1, -1\}$ depending if its a left turning arc or right turning arc. The two angles θ_1 and θ_2 are defined as

$$\theta_1 = \arctan 2(m_y, m_x) - s_i \frac{\pi}{2} + \theta_o \quad (21)$$

$$\theta_2 = \arctan 2(m_y, m_x) - s_i \frac{\pi}{2} - \theta_o \quad (22)$$

where $\theta_o \in (-\pi, \pi)$ defines how much of the circle should be included and can be viewed as a tuning parameter together with the arc radius for the overall path. The circular arcs are the local curves to the interior waypoints, and for the first and last waypoints straight lines are used as local curves. The straight lines are parameterized as

$$\mathbf{p}_{line}(t) = \begin{bmatrix} W_x + lm_x t \\ W_y + lm_y t \end{bmatrix} \quad (23)$$

and

$$\mathbf{p}_{line}(t) = \begin{bmatrix} W_x - lm_x(1-t) \\ W_y - lm_y(1-t) \end{bmatrix} \quad (24)$$

for the first and last waypoints respectively and where l is the length of the straight lines. Together with the double multiplicity of the first and last knot it is ensured that the path starts at \mathbf{W}_1 and ends in \mathbf{W}_n . Let the set of circular arcs and straight lines be the local curves to the GERBS defined by

$$\mathbf{p}(t) = \sum_{i=1}^n \mathbf{c}_i(t) B_i(t) \quad (25)$$

and generate a strictly increasing knot vector $\{t_j\}_{j=1}^{n+2}$ with interior multiplicity of zero, where $t_1 = t_2 = 0$, $t_{n+1} = t_{n+2} = 1$ and the rest of the knots are defined as $t_j = t_{j-1} +$

$1/(n-1)$. Then the resulting GERBS spline will interpolate every waypoint and the overall path will be C^m smooth when Beta Function basis functions are used or C^∞ in the case when Expo-Rational basis functions are used.

Algorithm 1 Generate tangents and turning directions.

```

1: function GENERATE_TANGENTS( $\mathbf{W}$ )
2:    $n \leftarrow \text{length}(\mathbf{W})$ 
3:   %Generate internal tangents
4:    $\mathbf{m}[1] \leftarrow (\mathbf{W}[2] - \mathbf{W}[1]) / \|\mathbf{W}[2] - \mathbf{W}[1]\|$ 
5:   for  $i \in [2, n-1]$  do
6:      $\mathbf{m}_1 \leftarrow (\mathbf{W}[i] - \mathbf{W}[i-1]) / \|\mathbf{W}[i] - \mathbf{W}[i-1]\|$ 
7:      $\mathbf{m}_2 \leftarrow (\mathbf{W}[i+1] - \mathbf{W}[i]) / \|\mathbf{W}[i+1] - \mathbf{W}[i]\|$ 
8:      $\mathbf{m}[i] \leftarrow ((\mathbf{m}_1 + \mathbf{m}_2) / \|\mathbf{m}_1 + \mathbf{m}_2\|$ 
9:     %The  $\wedge$  operator denotes the wedge product
10:     $s[i] = \text{sign}(\mathbf{m}_1 \wedge \mathbf{m}_2)$ 
11:   $\mathbf{m}[n] \leftarrow (\mathbf{W}[n] - \mathbf{W}[n-1]) / \|\mathbf{W}[n] - \mathbf{W}[n-1]\|$ 
12:   $s[1] = -s[2]$ 
13:   $s[n] = -s[n-1]$ 
14:  return  $\mathbf{m}$ 

```

Algorithm 2 GERBS path generation

```

1: function GENERATE_GERBS_PATH( $\mathbf{W}, \mathbf{m}, \mathbf{s}$ )
2:    $n \leftarrow \text{length}(\mathbf{W})$ 
3:   %Generate local curves
4:   for  $i \in [1, n]$  do
5:      $\mathbf{c}[i] \leftarrow \text{generate\_local\_arc}(R, \theta_o, \mathbf{W}_i, \mathbf{m}_i, \mathbf{s}_i)$ 
6:   %Generate knot vector
7:    $\mathbf{T} \leftarrow \text{generate\_knots}(n)$ 
8:   %Interpolate
9:   for  $t \in (\mathbf{T}[1], \mathbf{T}[n+2])$  do
10:     $\mathbf{p}[t] \leftarrow \mathbf{0}$ ;
11:    for  $k \in [2, n+1]$  do
12:       $\mathbf{c}_k \leftarrow \text{evaluate\_local\_curve}(\mathbf{c}[k-1], \mathbf{T}, k, t)$ 
13:       $\mathbf{B}_k \leftarrow \text{evaluate\_basis\_function}(\mathbf{T}, k, t)$ 
14:       $\mathbf{p}[t] = \mathbf{p}[t] + \mathbf{B}_k \cdot \mathbf{c}_k$ 
15:  return  $\mathbf{p}$ 

```

IV. NUMERICAL EXAMPLES

In Figure 1 the basis functions for the BFBS is shown. It can be seen that the local support for B_k is zero when $t = t_k$ which means that the BFBS will interpolate the local curve when $t = t_k$. This is because the BFBS is only of order two which would not produce a smooth path in the B-spline case. By reducing the order of the beta function a smoother path is generated however the local support remains the same in contrast to B-splines where the local support increases with the order of the B-spline. In Figure 2 the basis function for the Expo-Rational B-Spline is shown which generates a C^∞ curve that also interpolates the local curve $\mathbf{c}(t)$ when $t = t_k$. The intrinsic parameters are chosen to give a smooth transition at the peaks which reduces the required curvature.

In Figure 3 an ERBS is used with the intrinsic parameters of Figure 2 to interpolate a set of waypoints and in Figure

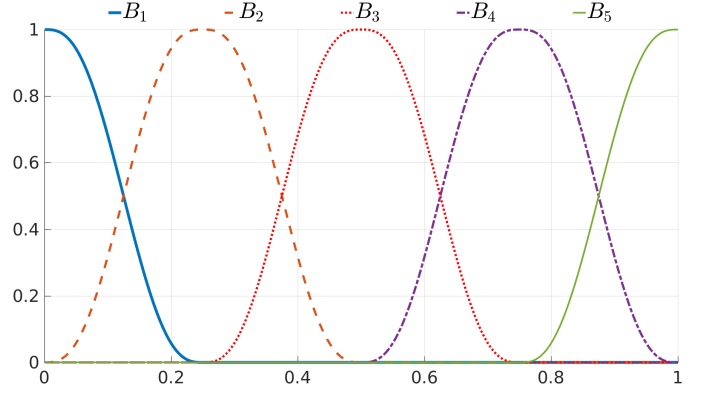


Fig. 1. BFBS basis functions for the knot vector $\mathbf{t} = \{0, 0, 1/4, 2/4, 3/4, 1, 1\}$.

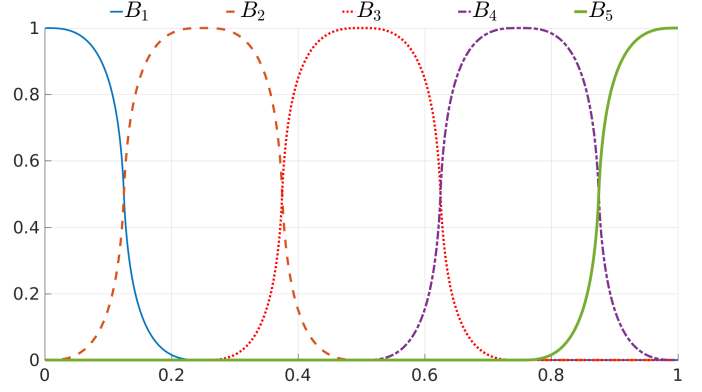


Fig. 2. Expo-Rational basis functions for the knot vector $\mathbf{t} = \{0, 0, 1/4, 2/4, 3/4, 1, 1\}$ with $\alpha = 0, \beta = 3, \gamma = 1, \sigma = 0.2$ and $\lambda = 0.5$.

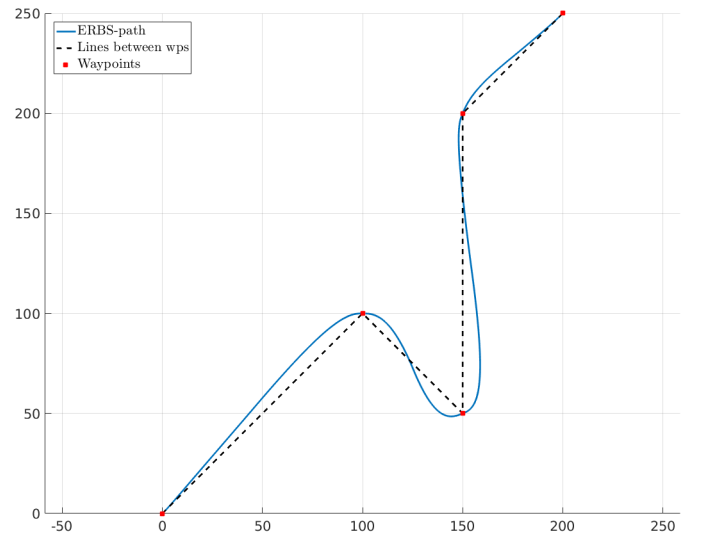


Fig. 3. Path generated using ERBS with intrinsic parameters from Figure 2.

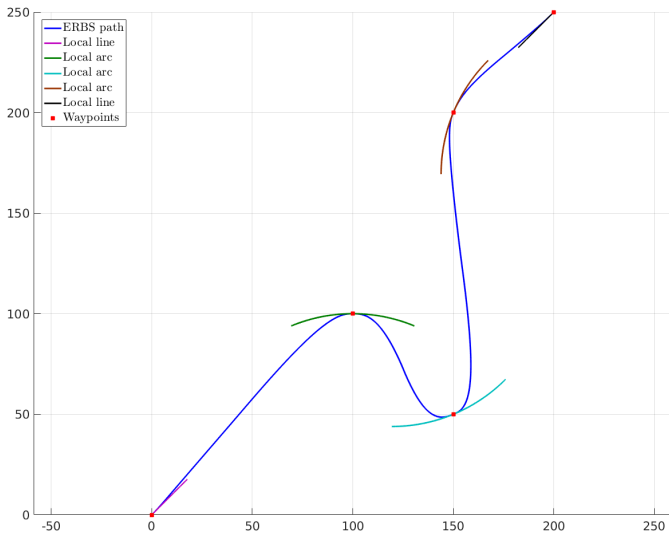


Fig. 4. Path generated using ERBS showing the local curves. Straight lines are used for the start and end local curve.

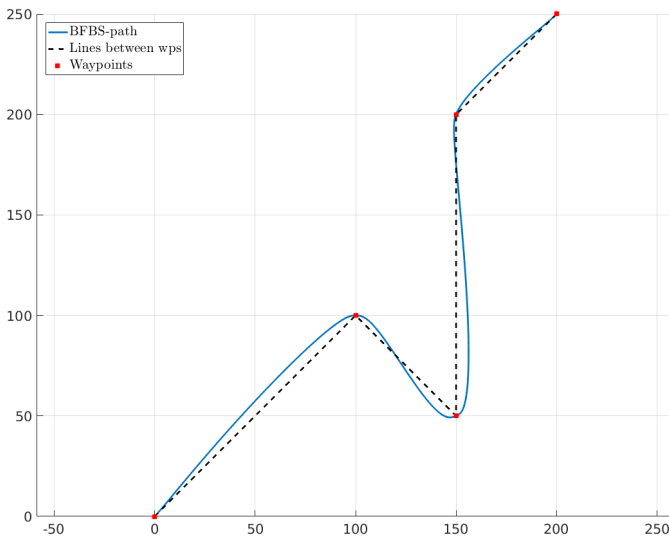


Fig. 5. Path generated using BFBS.

4 the local curves for the same path is shown. Notice how the start and end curves are straight lines while the remaining local curves are arcs with radius $R = 80$ and $\theta_o = \pi/8$, by changing the parameters of the local curves a different path can be obtained. This allows a lot of flexibility in designing the paths. In Figure 5 the path is generated by a BFBS and in Figure 6 it is compared to the ERBS path. It can be seen that the turns are tighter in the BFBS case since based on (10) there is less flexibility in tuning the basis functions. In Figure 7 the ERBS path is compared to a Dubin's path and it can be seen that the ERBS path follows the Dubin's path closely, but keeps the path smooth.

V. CONCLUSIONS

In this paper a new algorithm for continuous curvature path generation was presented based on GERBS. The desired

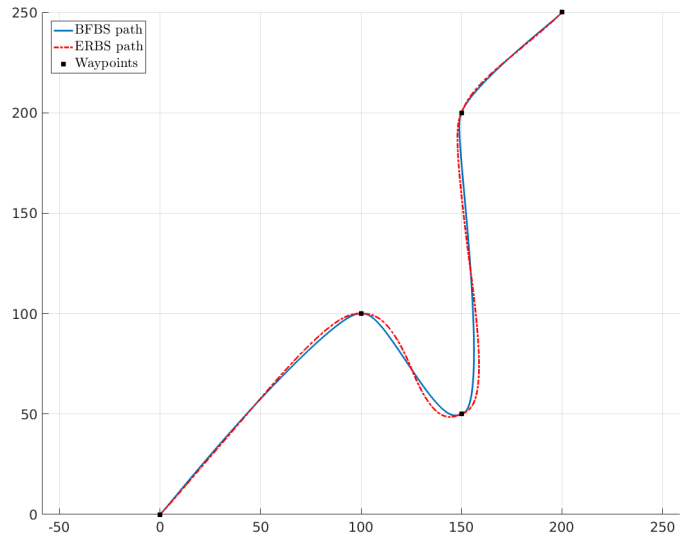


Fig. 6. Comparison between BFBS and ERBS.

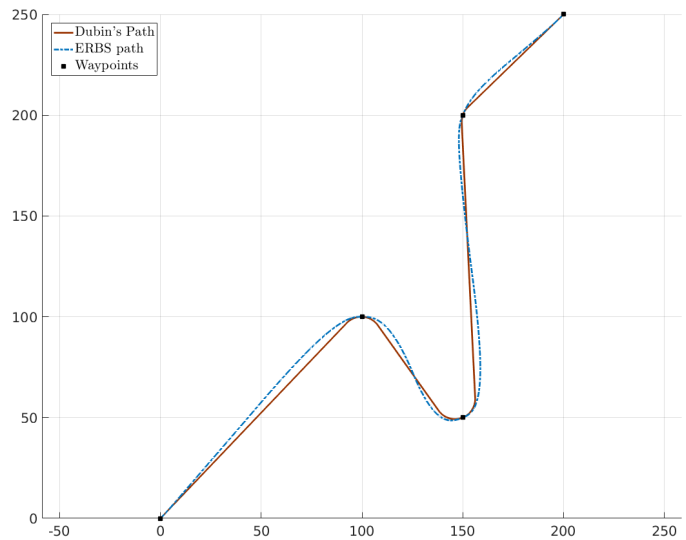


Fig. 7. Comparison between Dubin's Path and ERBS.

smoothness of the generated paths can be selected and can even be changed over different knot intervals. The path generated by the algorithm is compared to Dubin's paths and it can be seen that the path remains close to the Dubin's path while remaining smooth.

ACKNOWLEDGMENT

This research has been funded by the Norwegian Research Council and is part of the Arctic Earth Observation and Surveillance Technologies project 195142/I60.

REFERENCES

- [1] L. 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*, 1957.
- [2] R. W. Beard and T. W. McLain, *Small Unmanned Aircraft: Theory and Practice*. Princeton University Press, 2012.

- [3] W. Nelson, "Continuous-curvature paths for autonomous vehicles," in *Proceedings, 1989 International Conference on Robotics and Automation*, May 1989, pp. 1260–1264 vol.3.
- [4] A. M. Lekkas, A. R. Dahl, M. Breivik, and T. I. Fossen, "Continuous-Curvature Path Generation Using Fermat's Spiral," *Modeling, Identification and Control*, vol. 34, no. 4, pp. 183–198, 2013.
- [5] A. Tsourdos, B. White, and M. Shanmugavel, *Cooperative Path Planning of Unmanned Aerial Vehicles*. John Wiley & Sons, 2010.
- [6] T. Fraichard and A. Scheuer, "From Reeds and Shepp's to continuous-curvature paths," *IEEE Transactions on Robotics*, vol. 20, no. 6, pp. 1025–1035, Dec 2004.
- [7] R. T. Farouki and T. Sakkalis, "Pythagorean-hodograph space curves," *Advances in Computational Mathematics*, vol. 2, no. 1, pp. 41–66, Jan 1994.
- [8] E. Shan, B. Dai, J. Song, and Z. Sun, "A dynamic RRT path planning algorithm based on B-spline," in *2009 Second International Symposium on Computational Intelligence and Design*, vol. 2, Dec 2009, pp. 25–29.
- [9] T. Maekawa, T. Noda, S. Tamura, T. Ozaki, and K. ichiro Machida, "Curvature continuous path generation for autonomous vehicle using B-spline curves," *Computer-Aided Design*, vol. 42, no. 4, pp. 350 – 359, 2010.
- [10] F. Pinchetti, A. Joos, and W. Fichter, "Efficient continuous curvature path generation with pseudo-parametrized algebraic splines," *CEAS Aeronautical Journal*, May 2018.
- [11] K. Yang and S. Sukkarieh, "An analytical continuous-curvature path-smoothing algorithm," *Trans. Rob.*, vol. 26, no. 3, pp. 561–568, June 2010.
- [12] L. T. Dechevsky, A. Lakså, and B. Bang, "Expo-Rational B-Splines," *International Journal of Pure and Applied Mathematics*, vol. 27, no. 3, pp. 319–362, 2006.
- [13] A. Lakså, "Basic properties of Expo-Rational B-splines and practical use in computer aided geometric design," Ph.D. dissertation, University of Oslo, 2007.
- [14] L. T. Dechevsky, B. Bang, and A. Lakså, "Generalized Expo-Rational B-splines," *International Journal of Pure and Applied Mathematics*, vol. 57, no. 6, pp. 833–872, 2009.
- [15] A. R. Kristoffersen, L. T. Dechevsky, A. Lakså, and B. Bang, "Comparison between polynomial, Euler BetaFunction and ExpoRational BSpline bases," *AIP Conference Proceedings*, vol. 1410, no. 1, pp. 98–110, 2011.
- [16] L. T. Dechevsky, "Beta-Function B-Splines: definition and basic properties," *Int. J. Pure Appl. Math*, vol. 65, pp. 279–295.