for journal papers

Optimization of 4D Splines for Unmanned Aerial System (UAS) Trajectories under Obstacle, Kinematic, and Time Constraints

Matthew D. Osburn*, Cameron K. Peterson†, and John L. Salmon‡

Unmanned aerial systems (UAS) operating in dynamic environments must ensure safety for the duration of their flight. This paper presents an optimization method for planning safe, kinematically constrained, and time-constrained UAS trajectories in the presence of static and dynamic obstacles. Vehicle constraints are applied to a novel 4D 5^{th}-order polynomial spline formulation and the results are graphed to show meaningful relationships between constraints, optimality, and feasibility. This 4D 5^{th}-order polynomial spline is shown to be a special subset of the Bézier spline, which ensures that our approach inherits the advantages and convex hull properties of Bézier splines. The connection with Bézier splines will allow algorithms designed for Bézier splines to be used with 4D-polynomial splines in future work. Numerical experiments show the success of this approach in providing optimal trajectories for UAS navigating dynamic environments. The experiments further illustrate when gradient descent optimization will result in feasible trajectories, sub-optimal solutions, or distributions of optimal trajectories.

I. Introduction

The use of unmanned aerial systems (UAS) by governments and commercial entities has increased dramatically in recent years [1–3]. This surge in UAS air traffic has raised concerns about potential collisions between UAS and other aircraft, as well as with obstacles in their surrounding environment [4]. The risk of collisions is apparent in scenarios such as UAS colliding with vehicles within the same group, UAS with different tasks being directed on trajectories that pass dangerously close to each other, or UAS navigating through urban environments. Understanding how and when collisions may occur helps engineers choose the right collision avoidance method for their specific application.

Collision avoidance solutions have been well-researched and integrated into all levels of the UAS management and control systems. In [5], control barrier functions were integrated into the low-level flight controller to avoid collisions. Model predictive control methods were used in [6] to continuously optimize safe trajectories around obstacles. Artificial intelligence was used in [7] to detect other aircraft and avoid them as well as communicate the UAS's intent through radio communication. Pre-planned trajectories define the desired position of a UAS at every point in time. In [8–11], the trajectory that the low-level controller follows was optimized to avoid collisions while minimizing a cost function.

Collision-free trajectories can be generated in several ways. Some methods, like the one proposed in [12], plan trajectories that avoid the entire geometric space that an obstacle moves through to guarantee that the trajectory is safe for all time. In [13], an offline genetic algorithm is used to optimize a spline-based trajectory. In [14] and [15], prior knowledge of the expected movement of other agents and obstacles was leveraged to increase the safety of the planned

derivatives are defined so that the trajectory and its derivative functions are associated with physically meaningful units without any additional bookkeeping.

In the paper by Klinefelter et al., trajectories were optimized with the requirement that the endpoints remain fixed in space and time. While velocity and acceleration constraints were mentioned, they were not implemented in the numerical optimization. This is important because many applications require the trajectory to obey certain kinematic constraints so that a controller can successfully follow the trajectory as in [8–11]. This paper uses 4D 5^{th}-order polynomial splines to create feasible trajectories that adhere to kinematic constraints while maintaining the fixed endpoint constraints of [17].

A disadvantage of the 4D 5^{th}-order polynomial implementation is its lack of connection with more commonly used spline formulations. For example, Bézier splines are often used in UAS path-planning approaches because they provide collision avoidance guarantees, describe an entire trajectory with a limited number of variables, and are compatible with many trajectory planning algorithms and methods. We show that 4D 5^{th}-order polynomial splines are a special subset of Bézier splines. This mathematical equivalence ensures that the advantages inherent in Bézier splines are also assured in 4D polynomial splines.

In summary, this paper provides collision-free trajectories that adhere to time and kinematic constraints in the presence of static and dynamic obstacles. To achieve this objective we optimize 4D 5^{th}-order splines using a gradient descent-based optimization and create safe trajectories with fixed endpoints. The optimization method is set to minimize the UAS' path length. However, kinematic constraints and obstacles (static and dynamic) may make achieving the true minimum impossible. We explore the relationship between path minimization and the required constraints with a set of studies showing how these affect the minimum trajectory distance. To the best of our knowledge, this is the first time that trajectory-level collision avoidance has been achieved when there are both kinematic constraints and constraints on the time the vehicle must achieve its waypoints.

In this paper, we make the following three claims:

1) 4D 5^{th}-order polynomial spline trajectories with fixed time waypoints and vehicle constraints can be successfully optimized to avoid collisions.
2) Multi-modal distributions of optimal trajectories may result from gradient-descent optimization when the environment is dynamic. Many of the resulting trajectories within these distributions can be considered optimal.
3) 4D 5^{th}-order polynomial splines are equivalent to a special subset of Bézier splines.

The remainder of this paper proceeds as follows. Section II provides an overview of the 4D 5^{th}-order splines as well as its mathematical connection to Bézier splines. In addition to this, the trajectory optimization problem is formulated. Results are given in Section III, along with an explanation of the experimental design and a discussion of their meaning. Section IV concludes our paper and suggests some future work items.

II. Methods

In this section, we utilize 4D splines to provide a trajectory optimization technique with both vehicle and time constraints. This technique works in environments with both static and dynamic obstacles. The rest of this section provides details on the 4D splines, the derivation of our cost function and constraints, and the resulting optimization formulation.

A. 4D Spline Overview

This subsection provides an overview of the 4D splines used in this paper. For more details and applications of this formulation see [17]. A 4D spline is a piecewise function of position, velocity, and acceleration with respect to time, between a set of waypoints. Waypoints are the building blocks of 4D splines. They define the state that the UAS must reach at specific points in time. Interpolating between waypoints creates piecewise functions of 5^{th}-order polynomials describing the UAS position, velocity, and acceleration through time.

To construct a 4D spline we first define the waypoints. Let $w_i = [\vec{p}_i{}^T, \vec{v}_i{}^T, \vec{a}_i{}^T, t_i]^T$ be the ith spline waypoint and $w_{i+1} = [\vec{p}_{i+1}, \vec{v}_{i+1}, \vec{a}_{i+1}, t_{i+1}]$ be the connecting $i+1$ waypoint. The column vectors $\vec{p}_i$, $\vec{v}_i$, and $\vec{a}_i$ are the 3D position, velocity, and acceleration at the ith waypoint and the scalar t_i defines the time at which the spline passes through state $\vec{p}_i$, $\vec{v}_i$, and $\vec{a}_i$. The normalized time function τ is defined as $\tau_i(t) = (t - t_i)/T_i$ so that $\tau \in [0, 1]$. The normalization coefficient T_i is the difference between the time of w_i and w_{i+1} such that $T_i = t_{i+1} - t_i$. We use two special matrices B_i and C to ensure that the trajectory traverses between waypoints correctly and is twice differentiable. These matrices are

$$B_i = \begin{bmatrix} \vec{p}_{i+1} - \vec{p}_i, & \vec{v}_i T_j, & (\vec{v}_{i+1} + \vec{v}_i)T_i, & \vec{a}_i T_i^2, & (\vec{a}_{i+1} - \vec{a}_i)T_i^2 \end{bmatrix} \tag{1}$$

and

$$C = \begin{bmatrix} 0 & 0 & 10 & -15 & 6 \\ 1 & 0 & -2 & 1 & 0 \\ 0 & 0 & -4 & 7 & -3 \\ 0 & 1 & -2 & 1 & 0 \\ 0 & 0 & 1 & -2 & 1 \end{bmatrix}. \tag{2}$$

Let $\vec{\mathrm{p}}_i(t), \vec{\mathrm{v}}_i(t)$ and $\vec{\mathrm{a}}_i(t)$ be the vector-valued polynomial functions defining the position, velocity, and acceleration between waypoints w_i and w_{i+1} at time t. The equations describing the trajectory between w_i and w_{i+1} are

$$\vec{\mathrm{p}}_i(t) = \vec{p}_i + B_i C \left[\tau_i(t), \tau_i(t)^2, \tau_i(t)^3, \tau_i(t)^4, \tau_i(t)^5\right]^T, \tag{3}$$

$$\vec{\mathrm{v}}_i(t) = \frac{d\vec{\mathrm{p}}_i(t)}{dt} = B_i C \left[1, 2\tau_i(t), 3\tau_i(t)^2, 4\tau_i(t)^3, 5\tau_i(t)^4\right]^T \frac{1}{T_i}, \tag{4}$$

$$\tag{5}$$

and

$$\vec{\mathrm{a}}_i(t) = \frac{d\vec{\mathrm{v}}_i(t)}{dt} = B_i C \left[0, 2, 6\tau_i(t), 12\tau_i(t)^2, 20\tau_i(t)^3\right]^T \frac{1}{T_i^2}. \tag{6}$$

Using these equations, we can define piecewise functions composed of the $n-1$ individual polynomials that interpolate between n waypoints. The position over the entire 4D trajectory is

$$\vec{P}(t) = \begin{cases} \vec{\mathrm{p}}_1(t) & \text{if } t_1 \le t < t_2 \\ \vec{\mathrm{p}}_2(t) & \text{if } t_2 \le t < t_3 \\ & \vdots \\ \vec{\mathrm{p}}_{n-1}(t) & \text{if } t_{n-1} \le t \le t_n \end{cases}, \tag{7}$$

which is a vector-valued piecewise function that contains each equation connecting the spline's waypoints.

Similarly, the velocity, $\vec{V}(t)$, and acceleration, $\vec{A}(t)$, are defined as

$$\vec{V}(t) = \begin{cases} \vec{\mathrm{v}}_1(t) & \text{if } t_1 \le t < t_2 \\ \vec{\mathrm{v}}_2(t) & \text{if } t_2 \le t < t_3 \\ & \vdots \\ \vec{\mathrm{v}}_{n-1}(t) & \text{if } t_{n-1} \le t \le t_n \end{cases} \tag{8}$$

and

$$\vec{A}(t) = \begin{cases} \vec{\mathrm{a}}_1(t) & \text{if } t_1 \le t < t_2 \\ \vec{\mathrm{a}}_2(t) & \text{if } t_2 \le t < t_3 \\ & \vdots \\ \vec{\mathrm{a}}_{n-1}(t) & \text{if } t_{n-1} \le t \le t_n \end{cases}. \tag{9}$$

B. Connection between 4D splines and Bézier splines

In this subsection, we will show how to convert the trajectory connecting two 4D waypoints into an equivalent set of 5^{th}-order Bézier curve control points. This process will provide the control points for an equivalent piecewise Bézier spline when applied to each polynomial function in $\vec{P}(t)$. The Bézier spline will also be twice differentiable as is the original 4D 5^{th}-order polynomial spline.

Expanding Equation (3) and grouping terms to form a polynomial in τ yields

$$\begin{aligned} \vec{\mathrm{p}}_i(\tau) = {} & \vec{p}_i \\ & + (T\vec{v}_i)\tau \\ & + (\frac{1}{2}\vec{a}_i T^2)\tau^2 \\ & + (10(\vec{p}_{i+1} - \vec{p}_i) + \frac{1}{2}(\vec{a}_{i+1} - \vec{a}_i)T^2 - \vec{a}_i T^2 - 2T\vec{v}_i - 4T(\vec{v}_{i+1} + \vec{v}_i))\tau^3 \\ & + (-15(\vec{p}_{i+1} - \vec{p}_i) - (\vec{a}_{i+1} - \vec{a}_i)T^2 + \frac{1}{2}\vec{a}_i T^2 + T\vec{v}_i + 7T(\vec{v}_{i+1} + \vec{v}_i))\tau^4 \\ & + (6(\vec{p}_{i+1} - \vec{p}_i) + \frac{1}{2}(\vec{a}_{i+1} - \vec{a}_i)T^2 - 3T(\vec{v}_{i+1} + \vec{v}_i))\tau^5. \end{aligned} \tag{10}$$

The equation for a 5^{th}-order Bézier curve segment is

$$B(\vec{\tau}) = \sum_{k=0}^{5} \binom{5}{k} \tau^k (1-\tau)^{5-k} \vec{c}_k, \qquad 0 \le \tau \le 1, \tag{11}$$

where $\vec{c}_k$ is the kth control point of the curve. Expanding this equation and also grouping terms to form a polynomial in τ gives

$$
\begin{aligned}
\vec{B}(\tau) = \vec{c}_0 & \\
&+ 5(\vec{c}_1 - \vec{c}_0)\tau \\
&+ 10(\vec{c}_0 - 2\vec{c}_1 + \vec{c}_2)\tau^2 \\
&+ 10(-\vec{c}_0 + 3\vec{c}_1 - 3\vec{c}_2 + \vec{c}_3)\tau^3 \\
&+ 5(\vec{c}_0 - 4\vec{c}_1 + 6\vec{c}_2 - 4\vec{c}_3 + \vec{c}_4)\tau^4 \\
&+ (-\vec{c}_0 + 5\vec{c}_1 - 10\vec{c}_2 + 10\vec{c}_3 - 5\vec{c}_4 + \vec{c}_5)\tau^5.
\end{aligned} \tag{12}
$$

Writing the Bézier spline as a fifth-order polynomial allows us to select control points that yield an equivalent curve to that of the 4D polynomial. We solve for the six control points that make up a 5^{th}-order Bézier curve by setting Equation (12) equal to Equation (10) and matching coefficients. This gives the control points

$$
\begin{aligned}
\vec{c}_0 &= \vec{p}_i \\
\vec{c}_1 &= \vec{p}_i + \frac{1}{5}\vec{v}_i T \\
\vec{c}_2 &= \vec{p}_i + \frac{2}{5}\vec{v}_i T + \frac{1}{20}\vec{a}_i T^2 \\
\vec{c}_3 &= \vec{p}_{i+1} - \frac{2}{5}\vec{v}_{i+1} T - \frac{1}{20}\vec{a}_{i+1} T^2 \\
\vec{c}_4 &= \vec{p}_{i+1} - \frac{1}{5}\vec{v}_{i+1} T \\
\vec{c}_5 &= \vec{p}_{i+1}.
\end{aligned} \tag{13}
$$

This resulting Bézier curve is a special subset of Bézier curves that shares the same properties of differentiability as the 4D spline. It is assumed that the Bézier curve's τ function is computed in the same ways as the 4D spline's τ function. Oftentimes the construction of Bézier splines must be constrained to have those properties, so this is a convenient way to guarantee they are incorporated into the Bézier spline. The equations for finding the waypoints of a 4D spline waypoint from a Bézier curve can be derived in a similar way. However, if the Bézier spline has not been constrained to be twice differentiable, the results will be contradictory and/or discontinuous.

C. Optimization Approach

In our experiments we use the geometric length of the trajectory as the optimization cost function. This cost function was chosen because reducing energy consumption or effort to get between points is a typical goal for UAS path planning. Additionally, with this cost function, we can use the geometric shortest distance as an analytical solution to determine if our optimization results should be classified as optimal or sub-optimal.

The length of a trajectory can be calculated using the 4D spline functions. We do this by breaking the vector valued position function $\vec{P}(t)$ into its x, y, and z scalar piecewise polynomial functions $P_x(t), P_y(t), P_z(t)$. Then we integrate the derivatives of these functions to find the total geometric length as

$$\begin{aligned} L &= \int \sqrt{\left(\frac{d}{dt}P_x(t)\right)^2 + \left(\frac{d}{dt}P_y(t)\right)^2 + \left(\frac{d}{dt}P_z(t)\right)^2}\, dt \\ &= \int \|\vec{V}(t)\|_2\, dt. \end{aligned} \tag{14}$$

Equation (14) can be approximated as

$$\begin{aligned} L &\approx \sum_{k=1}^{N} \left\| \frac{\vec{P}(kT_s) - \vec{P}((k-1)T_s)}{T_s} \right\|_2 T_s \\ &\approx \sum_{k=1}^{N} \|\vec{P}(kT_s) - \vec{P}((k-1)T_s)\|_2, \end{aligned} \tag{15}$$

where the trajectory is discretized into N points and T_s is the discretization time step.

We constrain the trajectory to limit a vehicle's velocity and acceleration to v_{max} and a_{max}, respectively. The constraints limit the maximum magnitude, as defined by the Euclidean norm, of the vector-valued piecewise functions $\vec{V}(t)$ and $\vec{A}(t)$. For any time t_p such that $t_0 \leq t_p \leq t_f$, the constraints can be written as $\|\vec{V}(t_p)\|_2 \leq v_{max}$ and $\|\vec{A}(t_p)\|_2 \leq a_{max}$. Additionally, we impose a separation distance between the UAS and any obstacles in the environment. The vector-valued function of a generic obstacle's position will be written as $\vec{P}_{obs}(t)$ and the minimum distance between the UAS and an obstacle will be constrained to $dist_{min} < \|\vec{P}(t) - \vec{P}_{obs}(t)\|_2$.

Combining the cost function (Equation (15)) with the velocity, acceleration, obstacle, and time constraints yields the optimization problem

$$\begin{aligned} \arg \min_{w \in [w_0, w_1, \ldots, w_n]} \quad & \sum_{k=1}^{N} \|\vec{P}(kT_s) - \vec{P}((k-1)T_s)\|_2 \\ \text{s.t} \quad & \|\vec{V}(t_p)\|_2 < v_{max} && \forall \{t_p : t_0 \leq t_p \leq t_f\} \\ & \|\vec{A}(t_p)\|_2 < a_{max} \\ & dist_{min} < \|\vec{P}(t_p) - \vec{P}_{obs}(t_p)\|_2 \text{ for all obstacles} \\ & w_0 = w_i \\ & w_n = w_f \\ & t_{w_i} \text{ remains fixed } \forall i \in [0, 1, \ldots, n] \end{aligned} \tag{16}$$

where the variable w refers to all the scalars that make up the positions, velocities, and accelerations of the intermediate

waypoints in the trajectory. While the positions, velocities, and accelerations of the intermediate waypoints are free to change, the time scalars of all waypoints in the trajectory remain fixed. Additionally, the first and last waypoints of the trajectory are fixed.

III. Results

Each of the following subsections will summarize the results of a different optimization experiment and give the related plots and analysis. The experiments start with a basic straight-line trajectory (no obstacles) to provide a baseline for comparison. We then show how the 4D trajectories behave when finding optimal solutions around static and dynamic obstacles. Optimization results are categorized as optimal, sub-optimal, or failures. The results are also graphed on contour plots to provide intuition on the relationship between constraints and results.

In each experiment a single intermediate waypoint is optimized between fixed start and end waypoints. The intermediat waypoint was initialized with random values for each optimization. There are situations where it may not be possible to move the desired start and end times (e.g. when an aircraft needs to land within a specific window of time). To show this functionality, all our experiments keep the starting and ending waypoint's times and states (position, velocity, and acceleration) fixed.

All code was developed and run using Python on an Intel i7-12th generation processor with 16 gigabytes of RAM. The optimization problem was solved using the constrained SQP algorithm found in the SciPy Python package. Due to the tolerance parameters and variable scaling of the sqp algorithm, a result is considered optimal if it is within 0.01% of the true geometric minimum solution. Each optimization took on average 2.5 seconds to run.

Experiment results are color-coded for easy identification. Red indicates that the optimization failed and the solution is invalid. Blue indicates that a sub-optimal solution was found. Finally, green indicates that an optimal solution was found between points A and B. Contour plots follow the same color scheme, where colors are interpolated at the appropriate scale (i.e. levels that are within the sqp algorithm'stolerance of the true global optimum will be colored green, levels that are suboptimal are colored blue, and failed optimizations are colored red). Some contour plots appear to have black regions, but these are simply an artifact of having tightly packed contour levels.

A. Experiment 1: Straight Line, No Obstacle

The first experiment illustrates the relationship between the velocity and acceleration constraints and the cost function when no obstacles are present. A trajectory will be classified as optimal if it forms a straight line from point A to point B without violating velocity or acceleration constraints. This experiment verifies that 4D 5^{th}-order polynomial splines can be optimized with constraints on velocity and acceleration and fixed starting and ending waypoints.

Here, we outline the parameters of this experiment. Point A was a fixed waypoint at the origin with the prescribed velocity and acceleration being zero (the UAS starts at rest). Point B was a fixed waypoint at t = 60s, moved 300

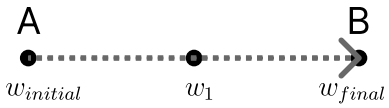
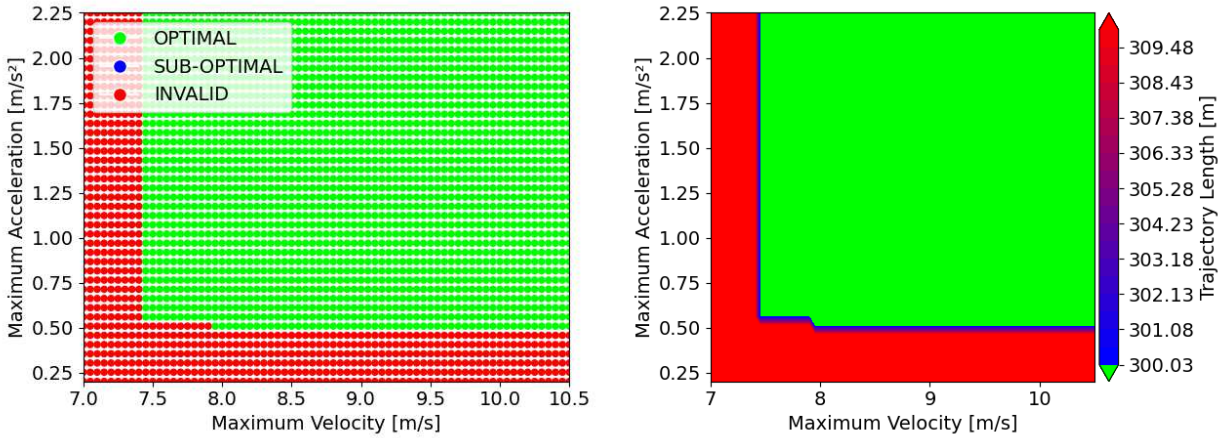


Fig. 1 Straight line from A to B has no obstacles in the way

meters away from point A along the x-axis, with the prescribed final velocity and acceleration being at rest. The fixed initial waypoint can be written as $w_i = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0]^T$ and the fixed final waypoint can be written as $w_f = [300m, 0, 0, 0, 0, 0, 0, 0, 0, 60s]^T$. An intermediate waypoint at $t = 30$s is optimized with a variety of combinations of velocity and acceleration constraints. The v_{max} constraint was chosen to be between 7m/s and 10.5m/s at 0.05m/s intervals. The a_{max} constraint was chosen to be between 0.2m/s^2 and 2.25m/s^2 at 0.05m/s^2 intervals. There are no obstacles between points A and B (see fig. 1).

In this scenario, the distance flown between A and B will be perfectly minimized or it will not be possible without breaking the constraints. If the maximum velocity is too small there isn't physically enough time to fly the distance between A and B. If the acceleration constraint is not sufficiently large then the velocity needed to fly the distance will not be achieved within the time frame of 60 seconds.

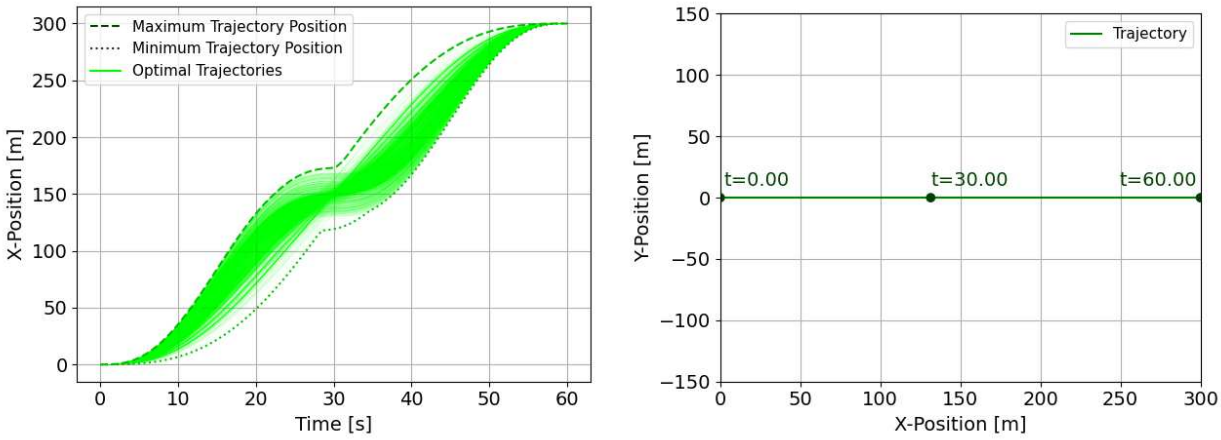


(a) ***Trajectory Cost Classification*** **(b)** ***Trajectory Cost Contours***

Fig. 2 ***Trajectory Classification***

The results of this experiment can be seen in fig. 2. The classification of the combinations of velocity and acceleration constraints is shown in fig. 2a and the contour plot of the optimal region can be seen in fig. 2b. A sharp cutoff between invalid and optimal solutions can be seen at $v_{max} \leq 7.35$m/s. As the maximum velocity increases, the maximum acceleration needed to fly an optimal straight path decreases.

Figure 3a shows the trajectories that are classified as optimal as a function of position versus time. The lines are colored green to emphasize that these are only the trajectories that were within 0.01% the length of the geometric shortest path. The dashed lines represent the maximum and minimum envelope of all optimal trajectories in our dataset.



(a) ***Optimal Trajectory Timelines*** **(b)** ***Example Trajectory***

Fig. 3 ***Optimal Trajectories***

This figure shows that trajectories can vary widely in their timing and still result in optimal paths.

An example optimal trajectory is shown in fig. 3b. This trajectory was chosen randomly from the set of optimal trajectories. The dots indicate the location of a waypoint making up the 4D spline. The waypoint's timestamp is displayed next to it to give an idea of how long it would take a UAS to travel between the waypoints.

This experiment successfully minimizes the length of the trajectory between point A and point B, given that the maximum velocity and acceleration are sufficiently large. The contour plot is flat with a sharp cutoff (fig. 2b), indicating that either an optimal trajectory can be found or no feasible path exists that does not violate the velocity and acceleration constraints. This intuitively makes sense, because if the UAS cannot average 5m/s over 60 seconds it will not be able to fly the distance between A and B. Because the UAS is starting from rest it would need to reach a max speed higher than 5m/s to average 5m/s along the trajectory, hence why the cutoff between invalid and optimal solutions is at $v_{max} = 7.35$m/s.

A wide range of trajectories can be considered optimal. Because the trajectory length is not related to time there is no single optimal solution. Some trajectories traverse a small distance in the first 30 seconds and then speed up, some move at a consistent rate, while others move quickly for the first 30 seconds and then slow down (see fig. 3a).

B. Static Obstacle

The second experiment (depicted in fig. 4) illustrates how well our optimization approach works when needing to re-route around a large static obstacle. The resulting trajectories successfully avoid the obstacle while maintaining the fixed starting and ending waypoint state and time values. However, this experiment also highlights a weakness in using splines as the trajectory formulation. The splines may not be able to match the shape and curvature of the obstacle making it impossible to minimize the distance of the trajectory to be the same as the analytical solution. The

obstacle used in this experiment is a sphere placed exactly halfway between the first and last waypoints. While additional waypoints could be added to better match the sphere's curvature, this would further increase the optimization complexity and it is unclear how many waypoints are needed in every situation to ensure an optimal solution. Delving into this question was left as an item of future work.

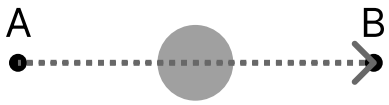


Fig. 4 Static Obstacle

In this experiment, point A was a fixed waypoint at the origin with the prescribed velocity and acceleration being at rest. Point B was a fixed waypoint at t = 60s, moved 300 meters away from point A along the x-axis, with the prescribed final velocity and acceleration being at rest. The fixed initial waypoint can be written as $w_i = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0]^T$ and the fixed final waypoint can be written as $w_f = [300m, 0, 0, 0, 0, 0, 0, 0, 0, 60s]^T$. An intermediate waypoint at t = 30s is optimized with various velocity and acceleration constraint combinations. The v_{max} constraint was chosen to be between 7m/s and 10.5m/s at 0.05m/s intervals and a_{max} was chosen to be between 0.25m/s^2 and 2.25m/s^2 at 0.05m/s^2 intervals. The spherical obstacle has a radius of 106.6m. This was chosen so that the radius would be sufficiently large compared to the resolution of the spline to highlight the spline's shortcomings. The shortest path between A and B requires that an optimal trajectory follow the curve of the sphere at exactly one-fourth the way around the circumference (see fig. 5), something that a 5$^{\text{th}}$-order spline is mathematically incapable of doing.

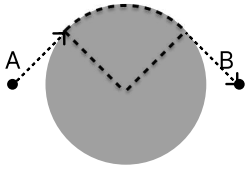
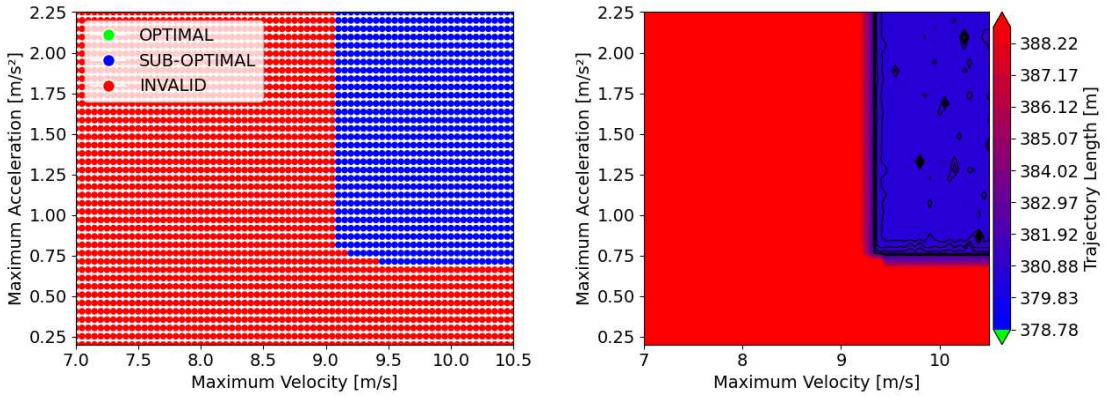


Fig. 5 Sphere radius requires one-fourth the circumference of the sphere to be followed to minimize distance.

Again, we define optimality as being within 0.01% of the geometric minimum distance. In this case, the minimum geometric distance between A and B is 378.74m. If a trajectory's length is within 3.7cm of 378.74m it is considered optimal.

The results of this experiment can be seen in fig. 6. The classification of the combinations of velocity and acceleration constraints is shown in fig. 6a and the contour plot of the sub-optimal region can be seen in fig. 6b. There are no optimal solutions in this experiment because the spline cannot perfectly match the curvature of the sphere. Each sub-optimal trajectory is guaranteed to be safe. An example trajectory is shown in fig. 7. This trajectory does not approach the sphere at the optimal 45° angle required to minimize path length.

This second experiment illustrates the shortcomings of a polynomial spline in minimizing length. Polynomials cannot perfectly match the curvature of the sphere. Increasing the order of the polynomial, or increasing the number of



(a) ***Result Classification*** **(b)** ***Result Contours***

Fig. 6 ***Trajectory Classification***

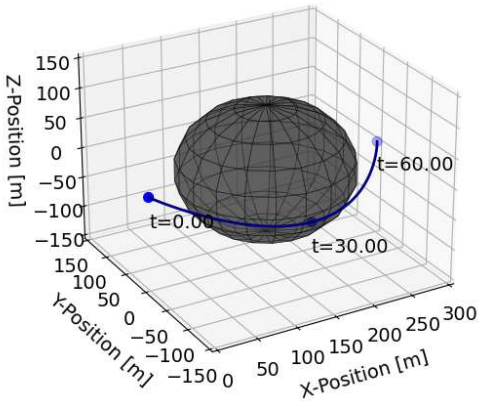


Fig. 7 ***Example Trajectory***

waypoints would help minimize the length of the path but increase computation time. A Dubins path or some other circular trajectory formulation would be able to exactly minimize the length of the trajectory around this obstacle while remaining computationally efficient.

C. Dynamic Obstacle

In this final experiment, we optimize a 4D trajectory in an environment with a moving obstacle. The results for this experiment once again show the utility of using 4D waypoints to guarantee successful achievement of the end waypoint (both in time and state) while maintaining a desired distance from obstacles. The obstacle is a sphere that moves perpendicular to the straight-line path between A and B (see fig. 8). To successfully minimize the distance traveled, the trajectory must articulate its velocity and acceleration to miss the obstacle moving across the straight line between point A and point B.

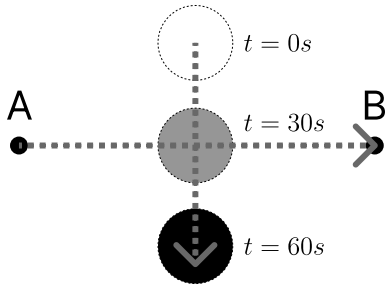


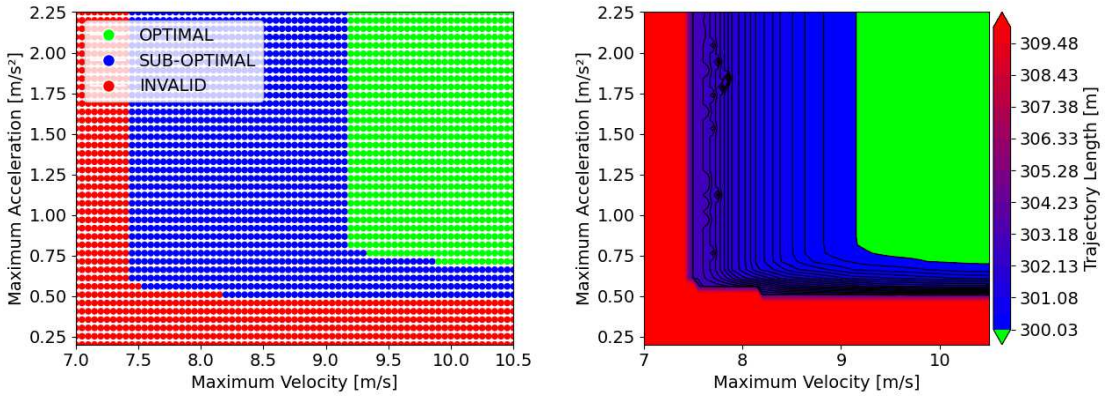
Fig. 8 Dynamic Obstacle

In this experiment, we once again set point A at the origin and point B 300 meters away from point A along the x-axis. Point B is given a waypoint time of 60s and both points require the UAS to have no acceleration or velocity at those locations and times. The fixed initial waypoint can be written as $w_i = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0]^T$ and the fixed final waypoint can be written as $w_f = [300m, 0, 0, 0, 0, 0, 0, 0, 0, 60s]^T$. An intermediate waypoint at t = 30s is optimized under a variety of combinations of velocity and acceleration constraints. The v_{max} constraint was chosen to be between 7m/s and 10.5m/s at 0.05m/s intervals and a_{max} was chosen to be between 0.25m/s^2 and 2.25m/s^2 at 0.05m/s^2 intervals. The spherical obstacle has a radius of 20m. The object is placed 120m on the y axis and 150m on the x axis. It moves in the negative y direction at 4m/s. Between t = 25s and t = 35s, part of the sphere will lie in the way of the optimal path.

The results of this experiment are shown in fig. 9. The classification of the combinations of velocity and acceleration constraints is shown in fig. 9a and the contour plot of the optimal region can be seen in fig. 9b. The classification graph shows that the minimum velocity that every optimal trajectory must achieve is $9.25m/s$. The boundary between invalid and sub-optimal solutions is the same as the boundary between invalid and optimal solutions from Experiment 1. The contour plot shows a gradually decreasing trajectory length (shown in blue) until the optimal solution can be found (shown in green). Black regions of the contour graph are regions where the contours are so tightly packed they appear black.

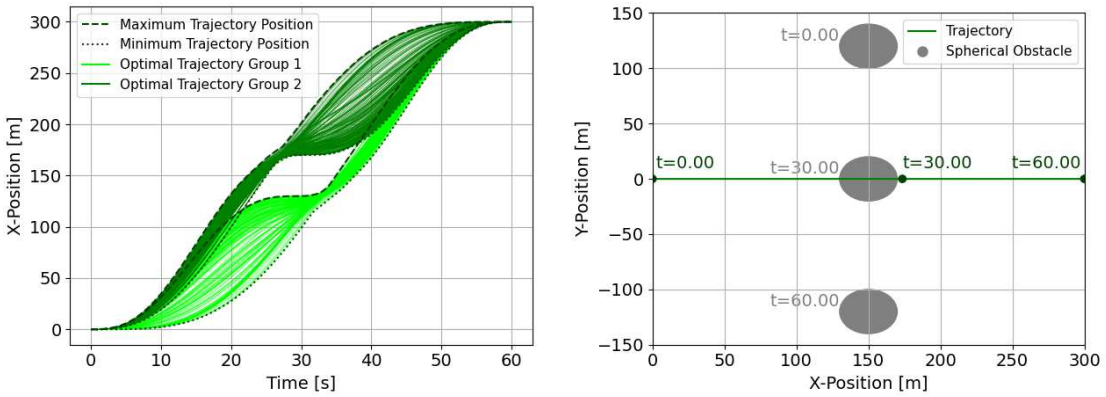
The trajectories that are classified as optimal are shown in fig. 10a. The lines are colored green to emphasize that these are only the trajectories that were within 0.01% the length of the geometric shortest path. The two different shades of green indicate that optimal solutions fall into two separate distributions. The dark green trajectories are trajectories that moved passed the middle of the straight line before the obstacle crossed (see fig. 10b for an example), while the light green trajectories waited for the obstacle to pass before continuing.

It is interesting to note that in fig. 10a at $t = 30s$ the gap between the minimum envelope of the dark green distribution and the maximum light green distribution is 40 meters which is the diameter of the sphere. Additionally, the light green distribution does not cross $x = 150m$ until $t = 35s$, where the sphere has completely passed the straight line between A and B. The dark green distribution passes $x = 150m$ before $t = 25s$, indicating that this distribution outruns the sphere.



(a) ***Result Classification*** **(b)** ***Result Contours***

Fig. 9 ***Trajectory Classification***



(a) ***Optimal Trajectory Timelines*** **(b)** ***Example Trajectory***

Fig. 10 ***Optimal Trajectories***

An example of a safe trajectory that outruns the sphere is shown in fig. 10b. Sub-optimal trajectories deviate from the straight line between point A and point B but do not violate vehicle, safety, and time constraints.

IV. Conclusion

We have shown that 4D 5$^{\text{th}}$-order splines can be optimized using gradient descent-based optimization to create safe trajectories. The splines are constrained to match the specified states and times of the endpoints while operating in environments with static and moving obstacles. Furthermore, we ensure the full trajectory conforms to constraints on the vehicle's velocity and acceleration.

Our studies of the relationships between vehicle kinematic constraints, time constraints, obstacles, and the minimization of the path distance show that optimal and safe trajectories can be found using 4D 5$^{\text{th}}$-order polynomial splines. The polynomial splines were also shown to be a specific instance of Bézier splines, so they retain the advantages

inherent in that wider category of splines.

In our experiments, we required that all waypoints remain at fixed times. While this does not affect the trajectory length, it does affect the velocity and acceleration profiles. In future work, we will add the ability for the optimizer to shift the time on waypoints that are being optimized and add in new intermediary waypoints. Future work will also investigate how timing constraints are affected by wind and disturbances, potentially inhibiting a vehicle's ability to follow the trajectory, as well as an investigation into the optimal number of waypoints needed to successfully avoid a collision.

Acknowledgements

This project was supported by NSF SBIR Phase 1 project 2111827 and 4D Avionic Systems, LLC. We thank Dr. Garth Thompson for his feedback and insightful suggestions.

References

[1] Sigala, A., and Langhals, B., "Applications of Unmanned Aerial Systems (UAS): A Delphi Study Projecting Future UAS Missions and Relevant Challenges," *Drones*, Vol. 4, No. 1, 2020, p. 8. https://doi.org/10.3390/drones4010008, URL https://www.mdpi.com/2504-446X/4/1/8.

[2] Lattimer, B. Y., Huang, X., Delichatsios, M. A., Levendis, Y. A., Kochersberger, K., Manzello, S., Frank, P., Jones, T., Salvador, J., Delgado, C., Angelats, E., Parés, M. E., Martín, D., McAllister, S., and Suzuki, S., "Use of Unmanned Aerial Systems in Outdoor Firefighting," *Fire Technology*, Vol. 59, No. 6, 2023, pp. 2961–2988. https://doi.org/10.1007/s10694-023-01437-0, URL https://link.springer.com/10.1007/s10694-023-01437-0.

[3] Merz, M., Pedro, D., Skliros, V., Bergenhem, C., Himanka, M., Houge, T., Matos-Carvalho, J. P., Lundkvist, H., Cürüklü, B., Hamrén, R., Ameri, A. E., Ahlberg, C., and Johansen, G., "Autonomous UAS-Based Agriculture Applications: General Overview and Relevant European Case Studies," *Drones*, Vol. 6, No. 5, 2022, p. 128. https://doi.org/10.3390/drones6050128, URL https://www.mdpi.com/2504-446X/6/5/128.

[4] Weibel, R. E., and Hansman, R. J., "Safety considerations for operation of unmanned aerial vehicles in the national airspace system," Tech. rep., 2006.

[5] Chen, Y., Singletary, A., and Ames, A. D., "Guaranteed Obstacle Avoidance for Multi-Robot Operations With Limited Actuation: A Control Barrier Function Approach," *IEEE Control Systems Letters*, Vol. 5, No. 1, 2021, pp. 127–132. https://doi.org/10.1109/LCSYS.2020.3000748, URL https://ieeexplore.ieee.org/document/9112342/.

[6] Lindqvist, B., Mansouri, S. S., Agha-mohammadi, A.-a., and Nikolakopoulos, G., "Nonlinear MPC for Collision Avoidance and Control of UAVs With Dynamic Obstacles," *IEEE Robotics and Automation Letters*, Vol. 5, No. 4, 2020, pp. 6001–6008. https://doi.org/10.1109/LRA.2020.3010730, URL https://ieeexplore.ieee.org/document/9145644/.

[7] Patrikar, J., Dantas, J., Ghosh, S., Kapoor, P., Higgins, I., Aloor, J. J., Navarro, I., Sun, J., Stoler, B., Hamidi, M., et al., "Challenges in close-proximity safe and seamless operation of manned and unmanned aircraft in shared airspace," *arXiv preprint arXiv:2211.06932*, 2022.

[8] Tian, Y., He, X., Xu, Y., Wan, L., and Ye, B., "4D Trajectory Optimization of Commercial Flight for Green Civil Aviation," *IEEE Access*, Vol. 8, 2020, pp. 62815–62829. https://doi.org/10.1109/ACCESS.2020.2984488, publisher: Institute of Electrical and Electronics Engineers Inc.

[9] Deori, L., Garatti, S., and Prandini, M., "4-D Flight Trajectory Tracking: A Receding Horizon Approach Integrating Feedback Linearization and Scenario Optimization," *IEEE Transactions on Control Systems Technology*, Vol. 27, No. 3, 2019, pp. 981–996. https://doi.org/10.1109/TCST.2018.2810201, publisher: Institute of Electrical and Electronics Engineers Inc.

[10] Alejo, D., Cobano, J. A., Heredia, G., and Ollero, A., "Collision-free 4D trajectory planning in Unmanned Aerial Vehicles for assembly and structure construction," *Journal of Intelligent and Robotic Systems: Theory and Applications*, Vol. 73, No. 1-4, 2014, pp. 783–795. https://doi.org/10.1007/s10846-013-9948-x, publisher: Kluwer Academic Publishers.

[11] Malyuta, D., Reynolds, T. P., Szmuk, M., Lew, T., Bonalli, R., Pavone, M., and Açıkmeşe, B., "Convex Optimization for Trajectory Generation: A Tutorial on Generating Dynamically Feasible Trajectories Reliably and Efficiently," *IEEE Control Systems*, Vol. 42, No. 5, 2022, pp. 40–113. https://doi.org/10.1109/MCS.2022.3187542, URL https://ieeexplore.ieee.org/document/9905530/.

[12] Mercy, T., Van Parys, R., and Pipeleers, G., "Spline-Based Motion Planning for Autonomous Guided Vehicles in a Dynamic Environment," *IEEE Transactions on Control Systems Technology*, Vol. 26, No. 6, 2018, pp. 2182–2189. https://doi.org/10.1109/TCST.2017.2739706, URL https://ieeexplore.ieee.org/document/8022960/.

[13] Behroo, M., and Banazadeh, A., "Near-optimal trajectory generation, using a compound B-spline interpolation and minimum distance criterion with dynamical feasibility correction," *Robotics and Autonomous Systems*, Vol. 74, 2015, pp. 79–87. https://doi.org/10.1016/j.robot.2015.07.003, URL https://linkinghub.elsevier.com/retrieve/pii/S0921889015001414.

[14] Ushani, A. K., Carlevaris-Bianco, N., Cunningham, A. G., Galceran, E., and Eustice, R. M., "Continuous-time estimation for dynamic obstacle tracking," *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, Hamburg, Germany, 2015, pp. 1137–1143. https://doi.org/10.1109/IROS.2015.7353513, URL http://ieeexplore.ieee.org/document/7353513/.

[15] Chu, W., and Zhang, P., "Dynamic obstacle avoidance of vision sensor mobile robot based on prior knowledge," *2020 2nd International Conference on Information Technology and Computer Application (ITCA)*, IEEE, Guangzhou, China, 2020, pp. 729–735. https://doi.org/10.1109/ITCA52113.2020.00157, URL https://ieeexplore.ieee.org/document/9421990/.

[16] Lee, D., Song, H., and Shim, D. H., "Optimal path planning based on spline-RRT* for fixed-wing UAVs operating in three-dimensional environments," *2014 14th International Conference on Control, Automation and Systems (ICCAS 2014)*, IEEE, Gyeonggi-do, Korea (South), 2014, pp. 835–839. https://doi.org/10.1109/ICCAS.2014.6987895, URL https://ieeexplore.ieee.org/document/6987895/.

[17] Klinefelter, M., Stone, A., Miller, J., Peterson, C. K., and Salmon, J. L., "Conflict Detection and Optimal 4D Trajectory Resolution Leveraging Polynomial Based Methods," *BYU Scholars Archive*, 2024.

[18] Choi, J.-w., Curry, R., and Elkaim, G., "Path Planning Based on Bezier Curve for Autonomous Ground Vehicles," *Advances in Electrical and Electronics Engineering - IAENG Special Edition of the World Congress on Engineering and Computer Science 2008*, IEEE, San Francisco, California, USA, 2008, pp. 158–166. https://doi.org/10.1109/WCECS.2008.27, URL http://ieeexplore.ieee.org/document/5233184/.

[19] Pastva, T. A., "Bezier curve fitting," *Calhoun Insitutional Archive of the Naval postgraduate School*, 1998.