B-Splines for Robotic Applications

Randal W. Beard
Brigham Young University

Updated: February 11, 2022

Abstract

These notes provide an overview of b-splines and their applications to robotics. In particular, we have in mind applications to path planning for aerial vehicles, model predictive control, and fixed-lag state estimation.

1 Motivation

The objective of these notes is to explore spline methods for robotic applications. In general, the position of a robot in Euclidian space can be described by a time parametrized trajectory $\mathbf{p}(t) \in \mathbb{R}^3$, $t \in [a, b]$. The time parametrized trajectory can be parameterized using a weighted sum of basic function as

$$\mathbf{p}(t) = \sum_{j=0}^{n-1} \mathbf{c}_j \phi_j(t),$$

where $\mathbf{c}_j \in \mathbb{R}^3$, and $\phi_j(t)$ are a set of basis functions. For example, the basis functions could be the set of polynomial power function $\phi_j(t) = t^j/j!$, or the set of sinusoidal function $\phi_j(t) = \sin(\frac{2\pi j}{n}t)$. The disadvantage of both the polynomial power functions and sinusoidal functions is that the basis functions are defined for all $t \in [a, b]$ and so each control points $_j$ influences the entire trajectory. Another disadvantage is that a large number of basis functions may be required to represent complicated trajectories.

In these notes, we will use b-spline basis functions which have a number of very nice properties that we will explore. In particular, a *b-spline* trajectory has the following form

$$\mathbf{p}(t) = \sum_{j=0}^{n-1} \mathbf{c}_j B_j^m(t; \mathbf{k}),$$

where $\mathbf{c}_j \in \mathbb{R}^4$ are the control points, $\mathbf{k} = (k_1, k_2, \ldots, k_K)$ are called the knot points where $i < j \implies k_i \leq k_j$, and $B_j^m(t; \mathbf{k})$ are the b-spline basis functions. The spline trajectories will be defined for t in the span of the knot points, i.e., $t \in [k_1, k_K]$.

Section 2 defines the b-spline basis function $B_j^m(t; \mathbf{k})$ and describes some of their properties that will be useful for path planning.

2 B-Spline Basis Functions

3 B-Spline Trajectories

4 B-Spline Planning for Chains of Integrators

5 B-Splines on Lie Groups

This section gives an overview of b-splines and their properties. In the next section, we will discuss path planning for quadrotors using b-splines.

In general, the b-spline basis functions are defined as

$$
\begin{aligned}
B_j^0(t;\mathbf{k}) &= \begin{cases} 1, & \text{if } k_j \leq t \leq k_{j+1} \\ 0, & \text{otherwise} \end{cases} \\
B_j^m(t;\mathbf{k}) &= \left(\frac{t-k_j}{k_{j+m}-k_j}\right) B_j^{m-1}(t;\mathbf{k}) + \left(\frac{k_{j+m+1}-t}{k_{j+m+1}-k_{j+1}}\right) B_{j+1}^{m-1}(t;\mathbf{k})
\end{aligned}
$$

where terms with a divide by zero caused by $k_{j+m} = k_j$ and $k_{j+m+1} = k_{j+1}$ are evaluated as zero. Note that the m^{th}-order basis function $B_j^m(t;\mathbf{k})$ is non-zero on the interval $t \in [k_j, k_{j+m}]$ and zero otherwise.

In this book we will only consider *uniform b-splines* where the non-identical knot points are uniformly spaced. In particular, we will pick the knot vector as a scaled and shifted version of

$$
\mathbf{k}_N^M = \alpha[\ \underbrace{0, 0, \dots, 0,}_{\text{repeat } M+1\text{-times}} \quad 1, 2, \dots N-1, \quad \underbrace{N, N, \dots, N}_{\text{repeat } M+1\text{-times}}\ ],
$$

where we will require that $N \geq M+1$. For example, the following are valid uniform knot vectors:

$$
\begin{aligned}
\mathbf{k}_1^0 &= [0, 1] \\
\mathbf{k}_2^0 &= [0, 1, 2] \\
\mathbf{k}_2^1 &= [0, 0, 1, 2, 2] \\
\mathbf{k}_4^1 &= [0, 0, 1, 2, 3, 4, 4] \\
\mathbf{k}_6^2 &= [0, 0, 0, 1, 2, 3, 4, 5, 6, 6, 6] \\
\mathbf{k}_6^3 &= [0, 0, 0, 0, 1, 2, 3, 4, 5, 6, 6, 6, 6],
\end{aligned}
$$

where we note that $\mathbf{k}_N^M \in \mathbb{R}^{N+2M+1}$. Uniform knot vectors will ensure that the spline trajectory is begins at the first control point and ends at the last control point, and that the trajectory is defined on the interval $t \in [0, N]$, and that the trajectory is M-times differentiable.

6 Conclusions

References