Master's degree in Computer Engineering for Robotics and Smart Industry

Robotics, vision and control

Robotics homeworks

Enrico Bonoldi - VR502852

July 2024

Contents

1 Homework 1 - Point-to-point trajectories

Polynomial trajectories (defined by polynomial functions) are the simplest form of trajectories ensuring continuity and smoothness.

$$q(t) = a_0 + a_1 t + a_2 t^2 + ... + a_n t^n \quad (1)$$

The degree of the polynomial depends on the number of conditions that we want to control such as initial and final position, velocity and/or acceleration.

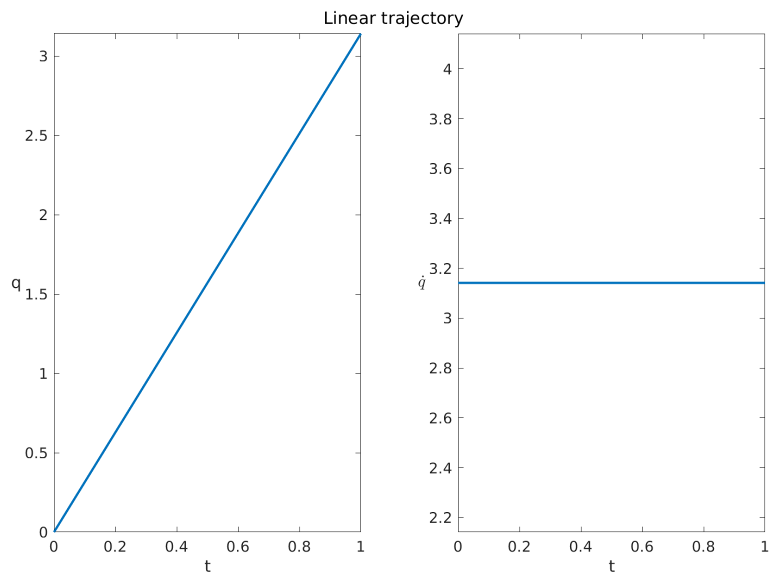

Figure 1: Linear trajectory

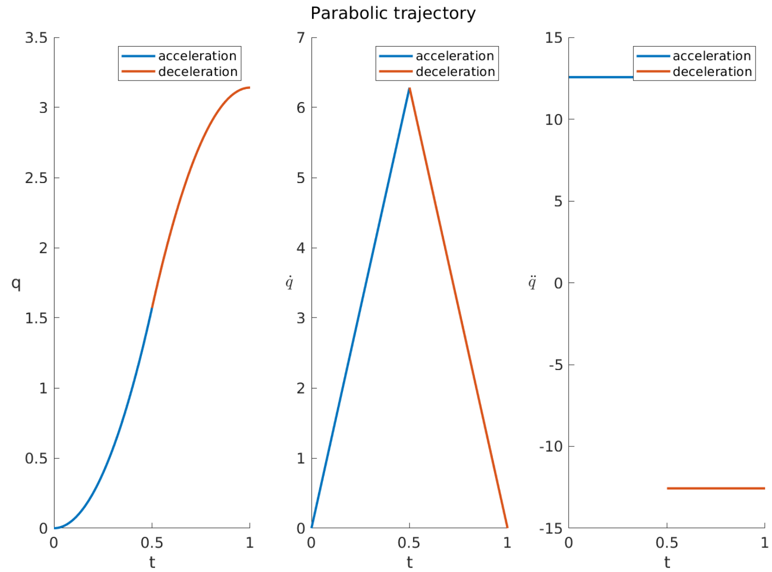

Figure 2: Parabolic trajectory

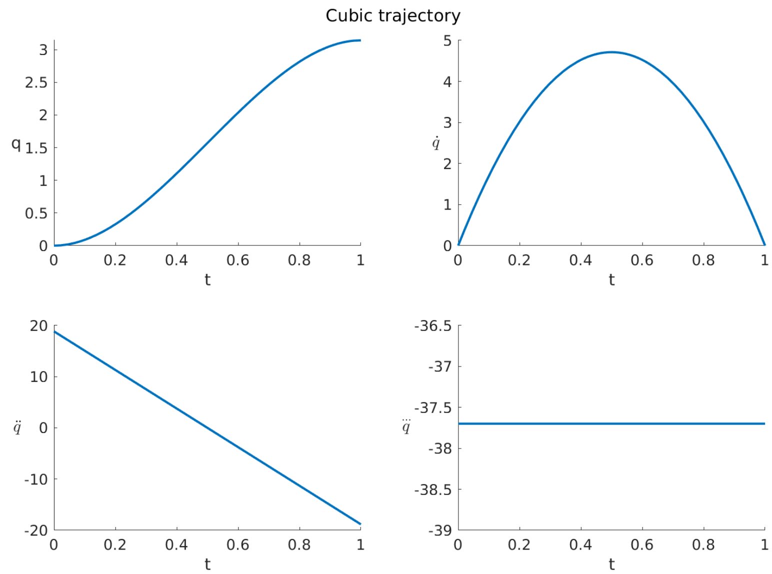

Figure 3: Cubic trajectory

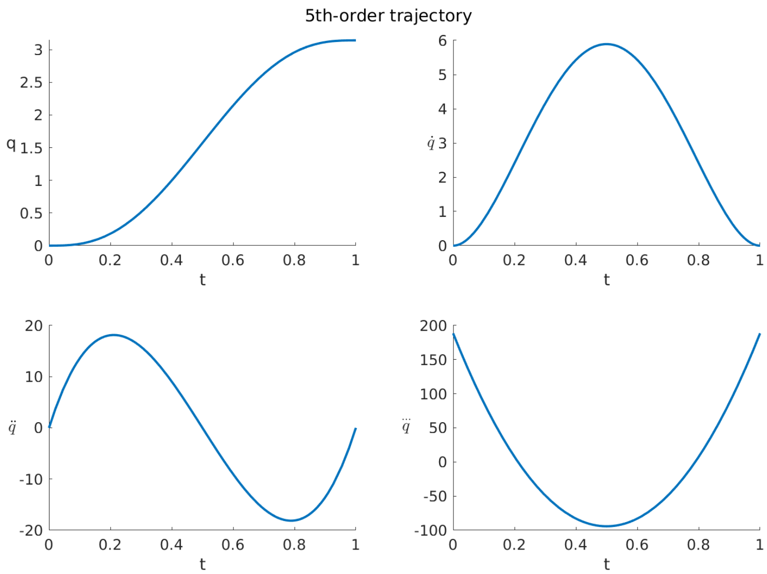

Figure 4: 5^{th}-order trajectory

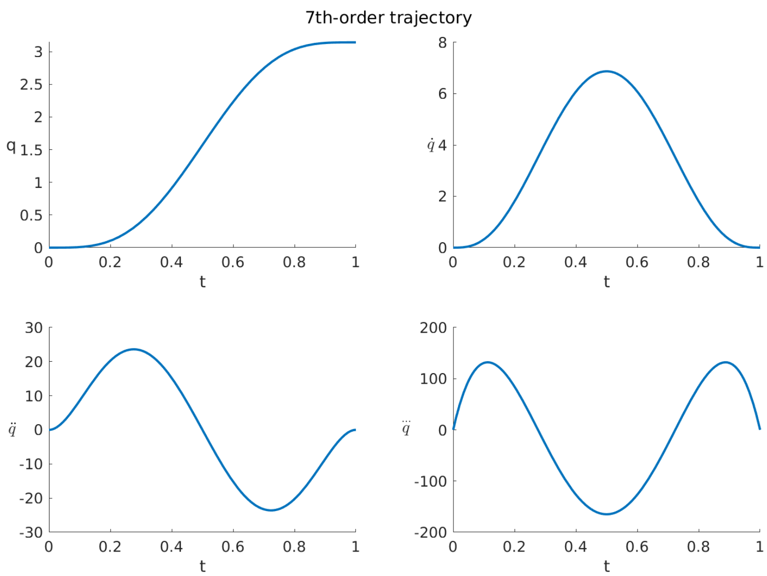

Figure 5: 7^{th}-order trajectory

2 Homework 2 - Trapezoidal velocity trajectories

Unlike polynomial trajectories, which use smooth polynomial functions, trapezoidal velocity profiles are piecewise linear and consist of three segments: acceleration, constant velocity, and deceleration.
The performance changes by taking into account different parameters.
In figure 6 the parameters are the acceleration, deceleration intervals duration (1 second) and total execution time (as a time inverval).
In figure 7 the parameters are initial and final velocity (2 m/s and 0 m/s respectively), maximum acceleration (10 m/s^2) and total execution time (as a time inverval).
In figure 8 the parameters are initial and final velocity, maximum acceleration (1 m/s^2) and maximum velocity (3 m/s). In this case the total execution time is one of the unknowns for which we solve (minimizing it, which is usually the case).

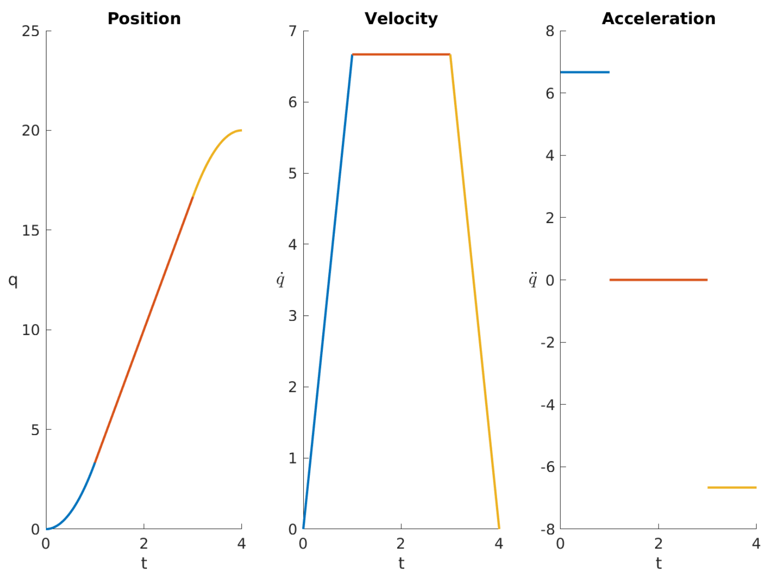

Figure 6: Trapezoidal velocity trajectory (2a). Acceleration and deceleration intervals as parameters

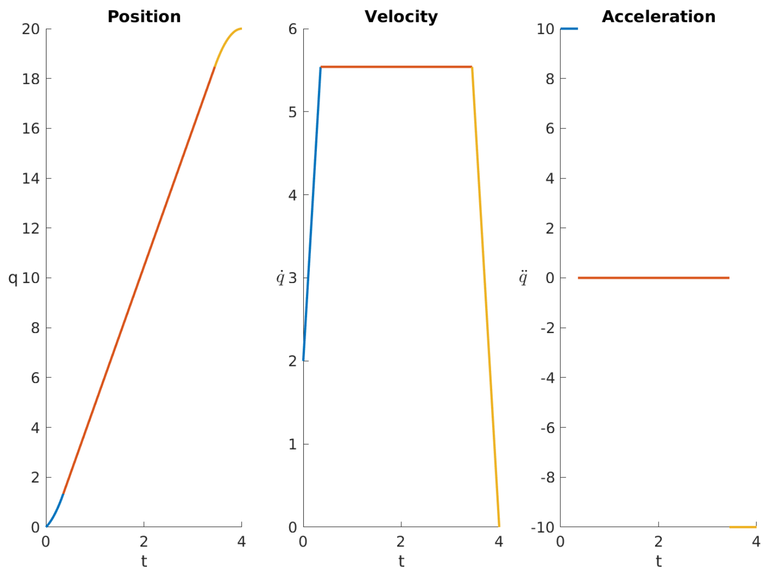

Figure 7: Trapezoidal velocity trajectory (2b). Initial and final velocities as parameters

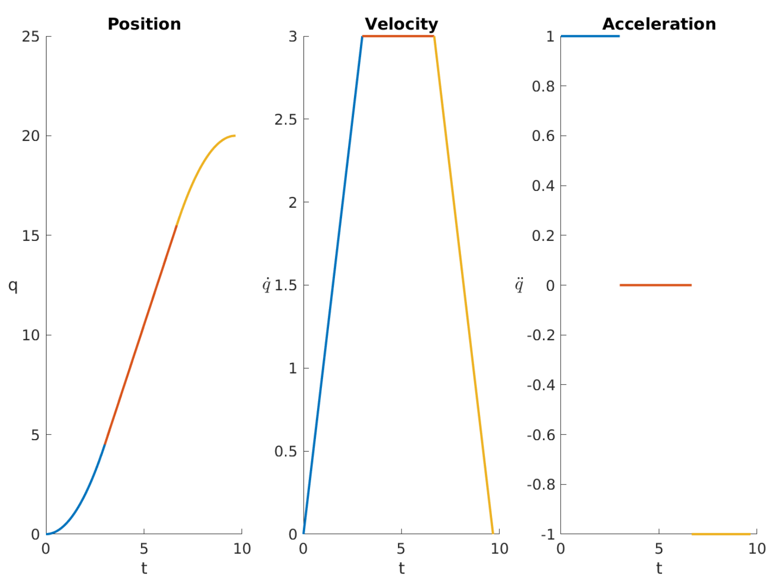

Figure 8: Trapezoidal velocity trajectory (2c). Maximum velocity and acceleration as parameters

3 Homework 3 & 4 - Multipoint trajectories

Multipoint trajectories involve planning a path that passes through multiple specified points(via points), ensuring smooth and continuous movement between these points.
This can be handled using various techniques like polynomial functions and piecewise polynomial functions (splines with or without smoothing).
As shown in figure 9, there are N via points (6 in this case) which implies the degree needed for the polynomial function is N - 1. Figures 10 and 11 show how splines can be interpolated to make a trajectory that passes through the via points, the first one uses the Euler's method to approximate the velocity at the via points (which does not guarantee a continuous acceleration), the second one ensures continuous acceleration imposing initial and final condition on each segment velocity (2 of these are the initial and final velocities of the trajectory).
Smoothing splines (figure 12) approximate a trajectory given a set of via points. There is a trade-off between accuracy and smoothness of the trajectory which is weighted by $\mu \in [0, 1]$

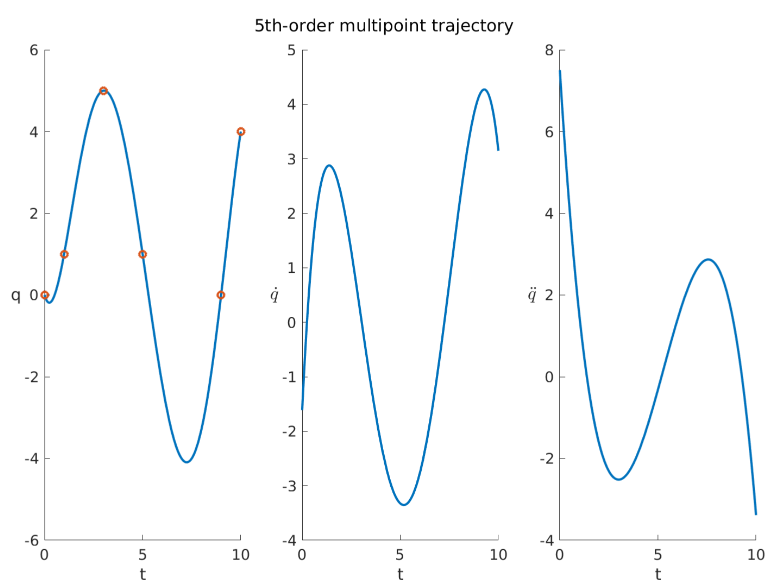

Figure 9: Multipoint trajectory. n-th order polynomial.

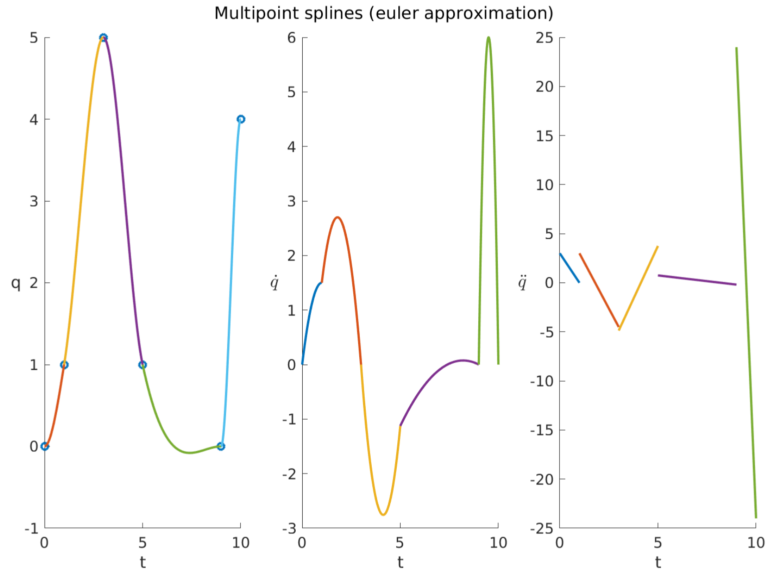

Figure 10: Multipoint trajectory. Splines, Euler's method for velocity approximation.

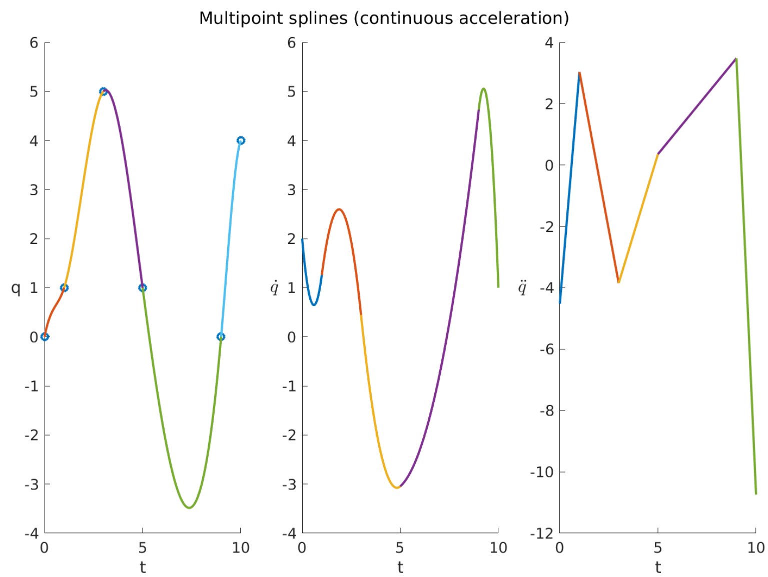

Figure 11: Multipoint trajectory. Continuous aceleration splines.

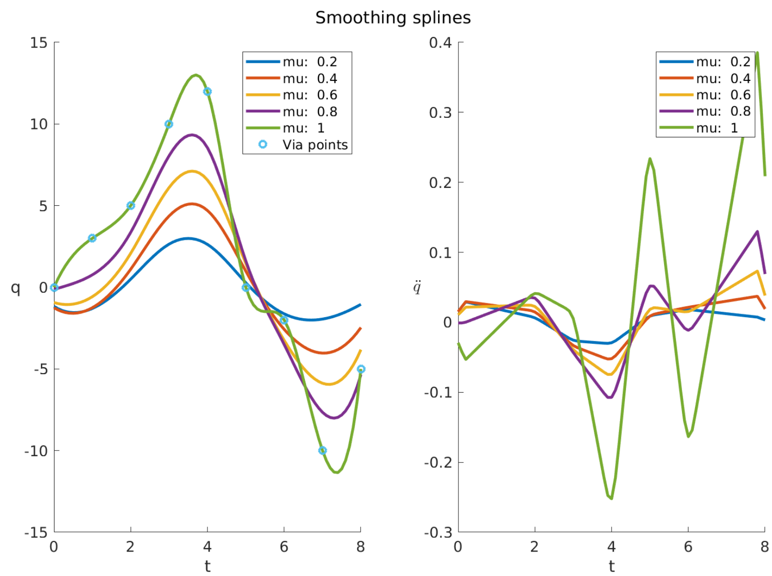

Figure 12: Multipoint trajectory. Smoothing splines with different smoothing factors.

4 Homework 5 - 3D trajectories

Building 3D trajectories with motion primitives involves generating a smooth and continuous path through multiple via points in three-dimensional space. Motion primitives are basic building blocks of motion planning that can be combined to form complex trajectories.
In figure 13 the trajectory is determined by interpolating the via points using the linear motion primitive (segments 1-2 and 4-5) and circular motion primitive (segments 2-3 and 3-4).
Figure 14 shows how, using the same technique as the one used in **Homework 1**, is possible to fit via points in 3 dimensions into a continuous trajectory. This is done by designing three different trajectories (one for each principal axis) and combining them all together in the final motion.

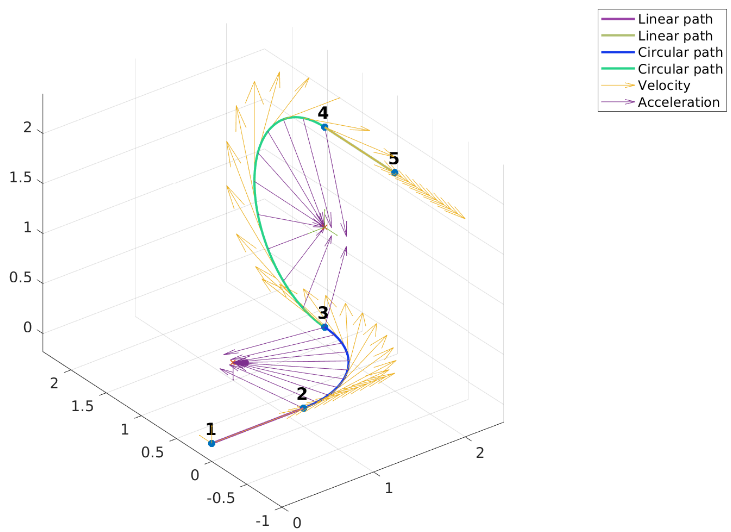

Figure 13: 3D trajectory using circular and linear motion primitives.

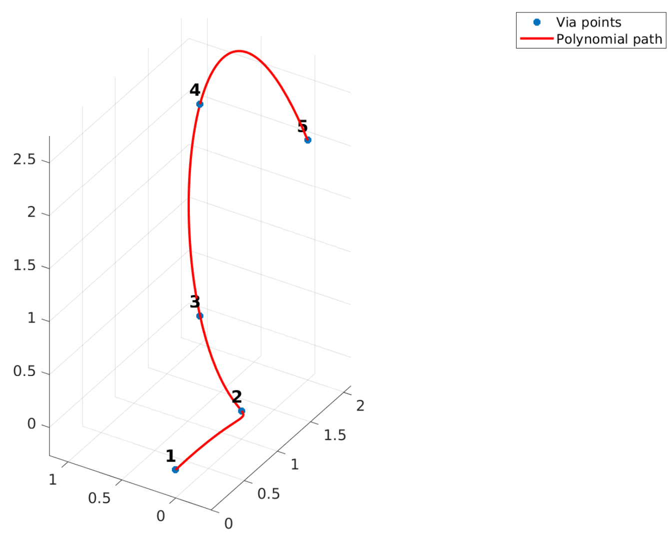

Figure 14: 3D trajectory using a polynomial function.

5 Homework 6 - 3D trajectories on a spherical surface

The fastest way to move from a point to another on a spherical surface is a segment of the great circle (i.e. a geodesic).
To move from point 1 (q_1) to point 2 (q_2) we must follow the great circle, to achieve that we have to rotate the vector q_1 around the geodesic circle pole ($q_1 \times q_2$) by an angle $\theta_{12} = \arccos(q_1 \cdot q_2)$.

Here the **animation** of the final motion (i.e. a constant velocity navigation on the trajectory).

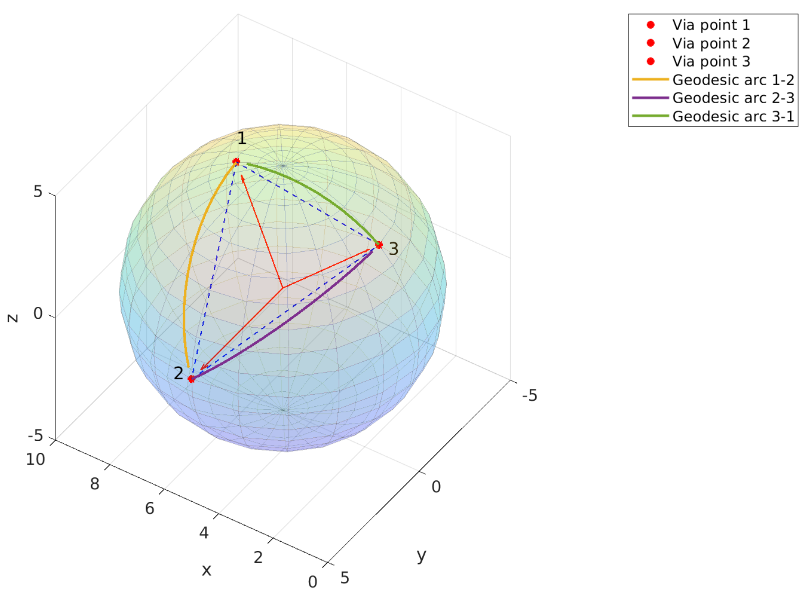

Figure 15: 3D trajectory using geodesics navigation.