Report of Course Numerical Optimization
Duboids: Extended Dubins path with Clothoids Junctions

Mattia Piazza   University of Trento

August 10, 2023

Introduction

The Dubins path is a well known problem in robotics and trajectory planning. The problem consists in finding the shortest path to connect two points in a plane with a prescribed heading (initial and final) with a maxim curvature constraint. This problem is suitable for mobile robots that are not omnidirectional. The solution can be derived analytically and is composed of a sequence of circular arcs and straight lines.[1, 2, 3] The Dubins path is a special case of the Reeds-Shepp path[4] where also backward motion is allowed.

However, the Dubins path is not suitable for vehicles that have a limited steering rate. In this case, the curvature of the path cannot change instantaneously.

Duboids is the proposed suboptimal solution to account for limits in curvature rate and therefore in steering rate of a vehicle. Duboids is a combination of Dubins path and clothoids.

Dubins

The problem can be stated as follows:

$$
\begin{aligned}
\min_{\kappa} \quad & \int_0^T v\,\mathrm{d}t = \min_{\kappa} \quad vT \\
\text{s.t.} \quad & \dot{x}(t) = v\cos(\theta(t)) \\
& \dot{y}(t) = v\sin(\theta(t)) \\
& \dot{\theta}(t) = v\kappa(t) \\
& x(0) = x_0, \ x(T) = x_T \\
& y(0) = y_0, \ y(T) = y_T \\
& \theta(0) = \theta_0, \ \theta(T) = \theta_T \\
& -\kappa_{max} \leq \kappa(t) \leq \kappa_{max}
\end{aligned}
\tag{1}
$$

Where v is a fixed velocity, x, y and θ are the position and heading of the vehicle, κ is the curvature of the path and κ_{max} is the maximum curvature allowed. The analytic solution to this problem is at most a sequence of 3 arcs, either left of right circular arcs at maximum curvature or straight lines.[1]

This problem and its solution can be applied in vehicle moving slowly and where the curvature change is almost instantaneous. However, real vehicles have a physical limit in the maximum curvature and maximum curvature rate. For this reason, we are looking to an extension of the Dubins path to incorporate this limitation as we will see in the next section exploiting clothoids.[5, 6]

Problem

The problem we want to solve is to minimize the time while connecting two points in the Cartesian space with a prescribed heading and curvature both at initial and final time. This problem can be formulated in the following way:

$$
\begin{aligned}
\min_{J} \quad & vT \\
\text{s.t.} \quad & \dot{x}(t) = v\cos(\theta(t)) \\
& \dot{y}(t) = v\sin(\theta(t)) \\
& \dot{\theta}(t) = v\kappa(t) \\
& \dot{\kappa}(t) = J(t) \\
& x(0) = x_0, \ x(T) = x_T \\
& y(0) = y_0, \ y(T) = y_T \\
& \theta(0) = \theta_0, \ \theta(T) = \theta_T \\
& \kappa(0) = \kappa_0, \ \kappa(T) = \kappa_T \\
& -\kappa_{max} \leq \kappa(t) \leq \kappa_{max} \\
& -J_{max} \leq J(t) \leq J_{max}
\end{aligned}
\tag{2}
$$

Where v is a fixed velocity, x, y and θ are the position and heading of the vehicle, κ is the curvature of the path, κ_{max} is the maximum curvature allowed, J is the controlled curvature rate (Jerk) and J_{max} is the maximum curvature rate allowed.

This problem can be translated into a BVP (Boundary Value Problem) and solved in a semi-analitycal fashion. In fact, the solution is composed of several arcs either at maximum rate (positive or negative or at zero rate).

Analytic solution

The Hemiltonian function of the problem is:

$$
\begin{aligned}
H =& \lambda_1(t)v\cos(\theta(t)) + \lambda_2(t)v\sin(\theta(t)) \\
& + \lambda_3(t)v\kappa(t) + \lambda_4(t)J(t) \\
& + \mu_1(t)(\kappa(t) - \kappa_{max}) \\
& + \mu_2(t)(-\kappa(t) + \kappa_{max})
\end{aligned}
\tag{3}
$$

The costate equations are:

$$
\begin{aligned}
\dot{\lambda}_1(t) &= 0 \\
\dot{\lambda}_2(t) &= 0 \\
\dot{\lambda}_3(t) &= \lambda_1(t)v\sin(\theta(t)) - \lambda_2(t)v\cos(\theta(t)) \\
\dot{\lambda}_4(t) &= -\lambda_3(t)v - \mu_1(t) + \mu_2(t)
\end{aligned}
\tag{4}
$$

Which yield that λ_1 and λ_2 are constant. Moreover, the control is

$$
J(t) = \underset{J \in [-J_{max}, J_{max}]}{\operatorname{argmin}} H \tag{5}
$$

However, H is linear in J, thus the second derivative with respect to the control is null, and the problem became singular. In the case of a singular arc, the control is either at the maximum or minimum of the control set or at zero.

$$
J(t) = \begin{cases} +J_{max} & \text{if } \lambda_4(t) > 0 \\ -J_{max} & \text{if } \lambda_4(t) < 0 \\ 0 & \text{if } \lambda_4(t) = 0 \end{cases} \tag{6}
$$

Physical interpretation

The physical interpretation is that the vehicle is either changing the curvature at maximum rate, or keeping the curvature constant. The only case when the curvature is kept constant is when the vehicle is travelling straight or when it is travelling at maximum curvature. Thus, the problem can be treated as a mixed integer optimization that in general is NP-hard.

Figure 1 illustrates all the possible combination of maneuvers connecting point P_0 and P_T. All intermediate points (at most 7) are switching point between clothoids and circular arcs or straight lines. From the starting point (and configuration) the vehicle can either steer toward maximum, minimum or zero curvature. If point P_0 already satisfy a bound the arc of clothoid L_1 is not necessary and will have zero length. There are some useless connection such as the repetition of two straight line or two arcs with same curvature which are accounted as special case connecting directly the point to the

final configuration.

Figure 1 represents in fact a graph connecting the initial and final configuration with all the possible combination of maneuvers. The problem could be solved with a graph search algorithm. However, the number of possible path inside the graph is not high and know at priory. Thus, a naive exploration of all the possible combination is feasible.

From figure 1, we can count 12 possible 7-arcs connections with a strange familiar resemblance to the Dubins path. In addition, 3 are the 3-arcs connections and 6 are the 5-arcs connections. The total number of possible combination is 21 as shown in table 1.

However, we are neglecting the possibility that, for some configuration of initial and final point the vehicle could perform maneuvers not reaching the maximum curvature values. This, will need a separate analysis when it comes to the naive exploration.
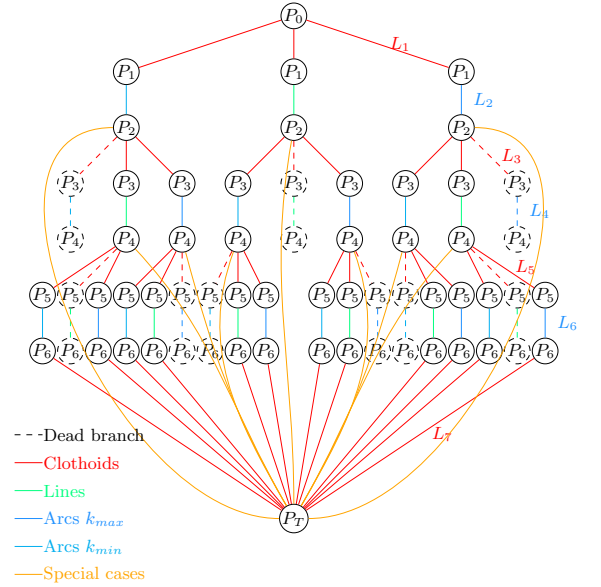

Figure 1: Possible Combination

Numeric solution

The problem was tackled with two numerical approaches. The first writing the problem as an Optimal Control Problem (OCP) and solving it with the indirect direct method and constraints formulated as penalization with PINS (PINS Is Not a Solver)[7]. The second approach was a naive exploration of all the possible maneuver combination to solve the problem.

#	L_2	L_4	L_6
1	L	S	L
2	L	S	R
3	L	R	L
4	L	R	S
5	S	L	S
6	S	L	R
7	S	R	L
8	S	R	S
9	R	L	S
10	R	L	R
11	R	S	L
12	R	S	R
13	L	–	–
14	S	–	–
15	R	–	–
16	L	S	–
17	L	R	–
18	S	L	–
19	S	R	–
20	R	L	–
21	R	S	–

Table 1: Possible combination.
L = Left arc, R = Right arc,
S = Straight line, $-$ = none

PINS

The problem was developed and solved using PINS to analyze the numerical solution of the optimal control problem and to have a reference solution to compare the naive exploration of all the possible combination of maneuvers.

Numerical Results

Duboids MATLAB implementation and exploration

The problem can be solved with a naive exploration of all the possible combination of maneuvers. In fact, all possible connection are a combination of the simple elementary Dubins connected by segments with linearly varying curvature also known as clothoids.[5]

An analysis done by hand and the of the numerical solution using PINS suggest, that there are at most 7 connection between two points. With 3 arcs at constant curvature and 4 arcs that connect all the arcs. Moreover, from figure 1 and table 1 we can see that there are 21 possible combination of maneuvers.

As stated before, this approach do not account for combination of maneuvers not reaching the maximum curvature values. This is a limitation of the approach that can be overcome with a more complex exploration of the possible combination. However, the achieved algorithm yields a valid suboptimal solution.

Algorithm structure

The algorithm was implemented in MATLAB. The implementation is divided in two main parts. The first part is a class that act as a collector for all the possible combination of maneuvers generated. This class will determine if a connection of a certain type is suitable/feasible for the specific initial and final point. The second part is a class for the single Duboid maneuver. Given the topology (shape) of the maneuver as in table 1 the class will generate the maneuver matching if possible the boundary conditions and compute the length of the maneuver.

The first class will store a list of all feasible candidate-to-be-best maneuver and select the best one according to the minimum-time/minimum length criteria.

Arch length computation

The Duboids lacks of a closed form solution at this stage of the development therefore the length of the maneuver is computed with a numerical optimization using *fmincon* function of MATLAB.

The constraint on the final curvature is already satisfied by the definition of the problem. However, coordinate and orientation of the final point depends on the choice of the maneuver and length of segment 2, 4 and 6. In general, P_7 the "final" point do not coincide with P_T. Thus, we should compute a suitable triplet of length for the segment L_2, L_4 and L_6. To satisfy this constraint we can minimize a function cost such as:

$$\min_{L_2,L_4,L_6} \quad (x_7 - x_T)^2 + (y_7 - y_T)^2 + (\theta_7 - \theta_T)^2$$
$$\text{s.t.} \quad [x_7, y_7, \theta_7] = \text{Duboid}(L_1, L_2, L_3)$$
$$0 \le L_2 \le L_{2,max}$$
$$0 \le L_4 \le L_{4,max}$$
$$0 \le L_6 \le L_{6,max}$$

(7)

where $L_{2,max}$, $L_{4,max}$ and $L_{6,max}$ are the maximum length of the segments L_2, L_4 and L_6 respectively. For circular arcs the maximum length is a full circle

with radius $R_{max} = 1/\kappa_{max}$. For the straight line the maximum length is not trivial, however we can safely find a reasonable upper bound for our application (i.e. 10 times the distance between initial and final point).

Duboids results

In this section we can see some of the results obtained with the Duboids algorithm. The results are shown in figures 2, 3, 4, 5 and 6.

The first figure (Figure 2) shows results obtained with initial condition set to zero for position, heading and curvature. The final condition is set to $x = 40$, $y = 20$, $\theta = 0$ and $\kappa = 0$.

The second example is a trivial connection with a

$L_{tot} = \mathbf{45.1441}$, $k_{max} = \mathbf{0.15}$, $J_{max} = \mathbf{0.1}$, **Type = [LSR]**
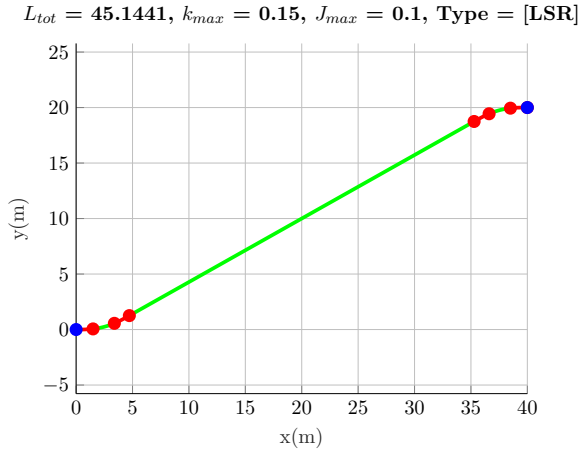
Figure 2: Duboids solution example 1

straight line (Figure 3).

The third example (Figure 4) shows a connection from $x = 0$, $y = 0$, $\theta = 0$ and $\kappa = 0$ to $x = -1$, $y = 0$, $\theta = 0$ and $\kappa = 0$ obtaining a fancy shape given the constraints on curvature and jerk.

The fourth example (Figure 5) shows a connection from $x = 0$, $y = 0$, $\theta = 0$ and $\kappa = 0$ to $x = 0$, $y = 0$, $\theta = \pi$ and $\kappa = 0$ obtaining a water drop like shape.

The fifth example (Figure 6) shows a connection from $x = 0$, $y = 0$, $\theta = 0$ and $\kappa = 0$ to $x = 0$, $y = 0$, $\theta = \pi/2$ and $\kappa = 0$ obtaining a convoluted form.
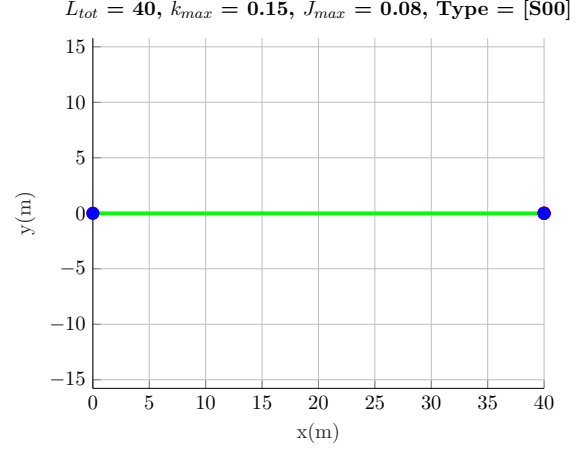
$L_{tot} = \mathbf{40}$, $k_{max} = \mathbf{0.15}$, $J_{max} = \mathbf{0.08}$, **Type = [S00]**

Figure 3: Duboids solution example 2

$L_{tot} = \mathbf{61.891}$, $k_{max} = \mathbf{0.2}$, $J_{max} = \mathbf{0.2}$, **Type = [RSL]**
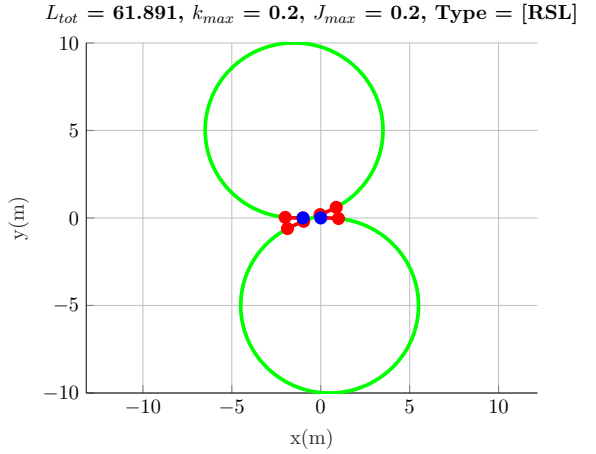
Figure 4: Duboids solution example 3

Compare PINS and Duboids

Conclusions

References

[1] A. M. Shkel and V. Lumelsky, "Classification of the dubins set," *Robotics and Autonomous Systems*, vol. 34, no. 4, pp. 179–202, 2001.

[2] Z. Chen and T. Shima, "Shortest dubins paths through three points," *Automatica*, vol. 105, pp. 368–375, 2019.

[3] B. Jha, Z. Chen, and T. Shima, "On shortest dubins path via a circular boundary," *Automatica*, vol. 121, p. 109192, 2020.
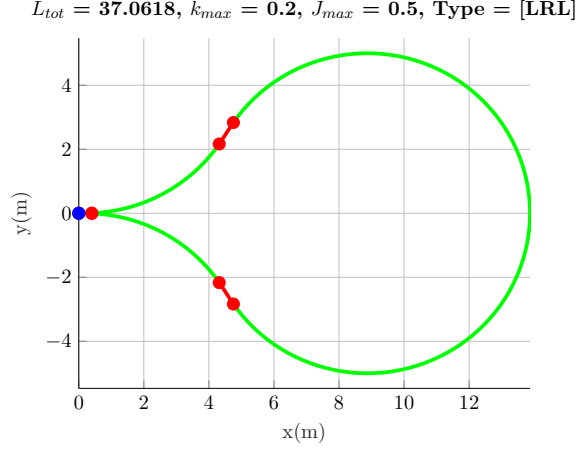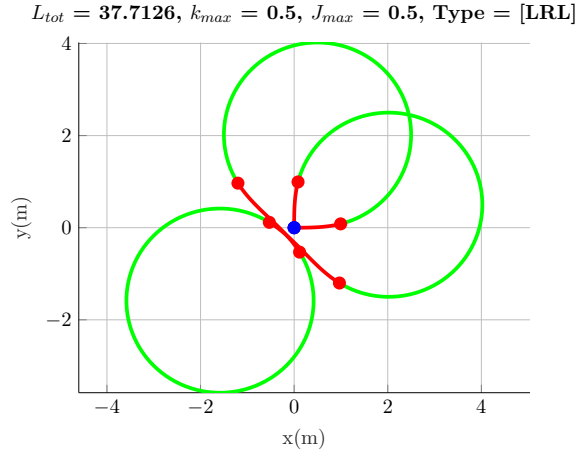
$L_{tot} = 37.0618, k_{max} = 0.2, J_{max} = 0.5, \text{Type} = [\textbf{LRL}]$

Figure 5: Duboids solution example 4

$L_{tot} = 37.7126, k_{max} = 0.5, J_{max} = 0.5, \text{Type} = [\textbf{LRL}]$

Figure 6: Duboids solution example 5

[4] R. Duits, S. P. Meesters, J.-M. Mirebeau, and J. M. Portegies, "Optimal paths for variants of the 2d and 3d reeds–shepp car with applications in image analysis," *Journal of Mathematical Imaging and Vision*, vol. 60, pp. 816–848, 2018.

[5] E. Bertolazzi and M. Frego, "G1 fitting with clothoids," *Mathematical Methods in the Applied Sciences*, vol. 38, no. 5, pp. 881–897, 2015.

[6] E. Bertolazzi, P. Bevilacqua, M. Frego, *et al.*, "Clothoids: a c++ library with matlab interface for the handling of clothoid curves," *Rend. Sem. Mat. Univ. Pol. Torino*, vol. 76, no. 2, pp. 47–56, 2018.

[7] F. Biral, E. Bertolazzi, and P. Bosetti, "Notes on numerical methods for solving optimal con-