Local Obstacle Avoidance Trajectory Generation using Dijkstra Algorithm

Project for
MEEN 689: Decision Making for Autonomous Vehicles

by Shivam Vashi (334003552)

Problem

How do we generate a trajectory that avoids obstacles along a given global trajectory (Lane Centerline) ?

Literature

1. Optimal Trajectory Generation for Dynamic Street Scenarios in a Frenet Frame [1]

   The paper provides an algorithm to generate **jerk-minimizing** local trajectories that avoid obstacles while obeying lane-width constraints.

   Formulation-

   The requirement is to minimize a cost which is a function of Jerk (J_t), time to reach the global path (T) and the end state (p_1) given by-

   $$C = k_j J_t(p(t)) + k_t g(T) + k_p h(p_1)$$

   Where, Jerk term- $J_t(p(t)) = \int_{t_0}^{t_1} \left(\dddot{p}(t)\right)^2 dt$

   and all the coefficients (k) are weightages for corresponding terms.

How to Minimize the Jerk term (J_t)?

One of the citations [2] prove that quintic (5th degree) polynomials minimize the jerk term (J_t)

Defining the cost function of jerk, the local path is obtained as functions constraining the sudden acceleration change.

The cost function C is defined as the time integral of the square of jerk when the AGV moves between the two positions within the time interval T.

$$C = \frac{1}{2}\int_0^T L\,dt$$

, where the performance index L is

$$L = (\frac{d^3x}{dt^3})^2 + (\frac{d^3y}{dt^3})^2.$$

Here, x and y indicate the AGV's position components on the $x-y$ coordinate system fixed on the ground (Figure 2).

The problem is to deduce the local path $(x(t), y(t))$ minimizing the cost function C.

For any functions x and y which are sufficiently differentiable in the interval $0 \leq t \leq T$, the cost function C is analytically assumed to be an extremum when $x(t)$ and $y(t)$ are the solution of the following Euler equations. (The terms depending on the two position components, x and y can be uncoupled.)

$$\frac{\partial L}{\partial x} - \frac{d}{dt}(\frac{\partial L}{\partial \dot{x}}) \cdots + (-1)^n \frac{d^n}{dt^n}\frac{\partial L}{\partial (x^{(n)})} = 0$$

$$\frac{\partial L}{\partial y} - \frac{d}{dt}(\frac{\partial L}{\partial \dot{y}}) \cdots + (-1)^n \frac{d^n}{dt^n}\frac{\partial L}{\partial (y^{(n)})} = 0$$

From these equations and the performance index L, the following equations are deduced.

$$\frac{d^3}{dt^3}\frac{\partial(\dddot{x}^2)}{\partial(\dddot{x})} = 0$$

$$\frac{d^3}{dt^3}\frac{\partial(\dddot{y}^2)}{\partial(\dddot{y})} = 0$$

Then

$$\frac{d^6x}{dt^6} = 0,$$

$$\frac{d^6y}{dt^6} = 0.$$

From the above differential equations, the following equations are derived.

$$\left.\begin{aligned} x(t) &= a_0 + a_1t + a_2t^2 + a_3t^3 + a_4t^4 + a_5t^5 \\ y(t) &= b_0 + b_1t + b_2t^2 + b_3t^3 + b_4t^4 + b_5t^5 \end{aligned}\right\}$$

Resulting solutions $(x(t), y(t))$ that minimize the time integral of the square jerk are represented by fifth order polynomials

Optimal Solution of the Cost Function

The paper then proves that quintic (5th degree) polynomials also minimize the Cost function defined earlier.

Proposition 1: Given the start state $P_0 = [p_0, \dot{p}_0, \ddot{p}_0]$ at t_0 and $[\dot{p}_1, \ddot{p}_1]$ of the end state P_1 at some $t_1 = t_0 + T$, the solution to the minimization problem of the cost functional

$$C = k_j J_t + k_t g(T) + k_p h(p_1)$$

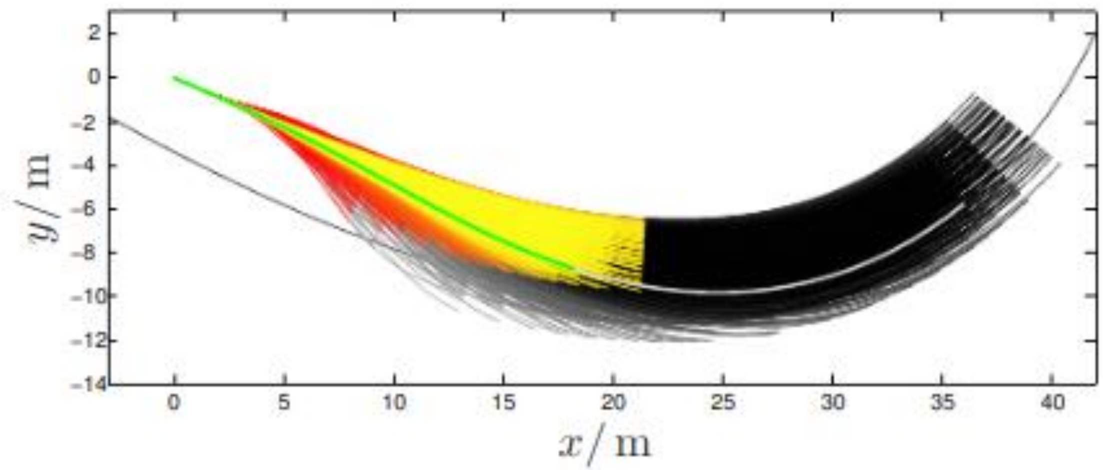
with arbitrary functions g and h and $k_j, k_t, k_p > 0$ is also a quintic polynomial.

Proof:[4] Assume the optimal solution to the proposed problem was not a quintic polynomial. It would connect the the two points P_0 and $P_1(p_{1,\text{opt}})$ within the time interval T_{opt}. Then a quintic polynomial through the same points and the same time interval will always lead to a smaller cost term $\int_{t_0}^{t_1} \dddot{p}^2(\tau)$ in addition to the same two other cost terms. This is in contradiction to the assumption so that the optimal solution has to be a quintic polynomial. ∎

[4]From an optimal control's perspective this is directly clear, as the end point costs $g(T)$ and $h(p_1)$ do not change the Euler-Lagrange equation.

Literature (Contd.)

To find the optimal solution, a set of quintic polynomials with varying values of T and p_1 is generated and then the trajectories that violate vehicle constraints (maximum velocity, acceleration, and curvature) and obstacle collisions are filtered out. Out of the remaining trajectories, the one that minimizes the cost (C) is chosen (shown in green)

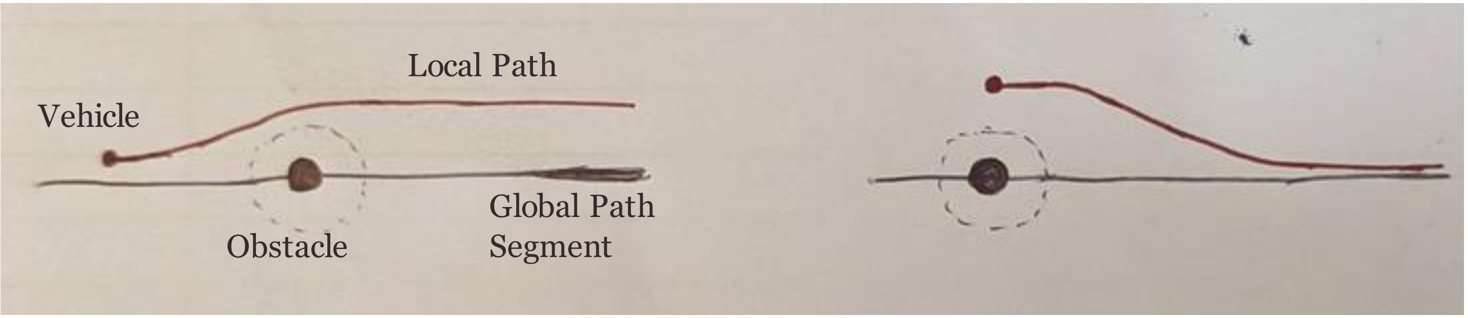


Colormap of the cost C for the set of local trajectories. Green is lowest and Red is highest

Note that the end of each local trajectory has a lateral velocity of zero. This can be changed but the computation time would drastically increase because of an additional variable.

Shortcomings

1. Since the end of each local trajectory in the set has a lateral velocity of zero, the local trajectories generated before avoiding an obstacle will never converge to the global trajectory until the replanning occurs after passing the obstacle.



This implies that we must replan as frequently as possible to ensure optimality.

2. Since the algorithm generates all possible local trajectories from the current position, the computation time would explode if a local trajectory until the end of the global trajectory is to be designed.

Alternative

2. [Local Path Planning Algorithm for Autonomous Vehicle Based on Multi-objective Trajectory Optimization in State Lattice][3]

   The paper provides a Graph Based Method to creating a State Lattice where every edge is a Cubic Polynomial. It suggests using Dijkstra Shortest Path Algorithm on the Graph to find the optimal local path for obstacle avoidance.

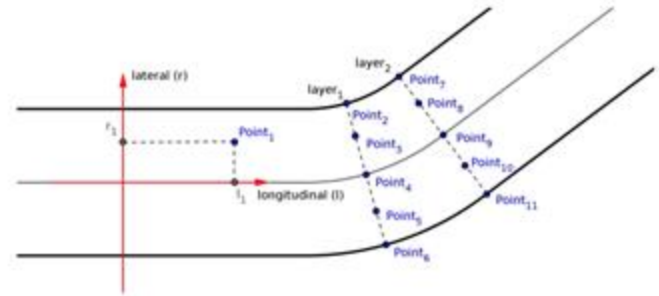
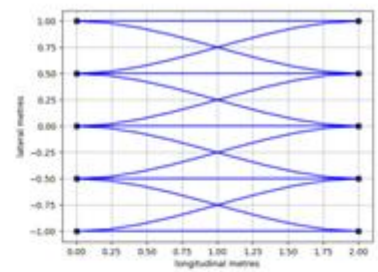
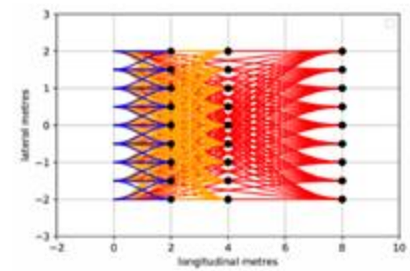


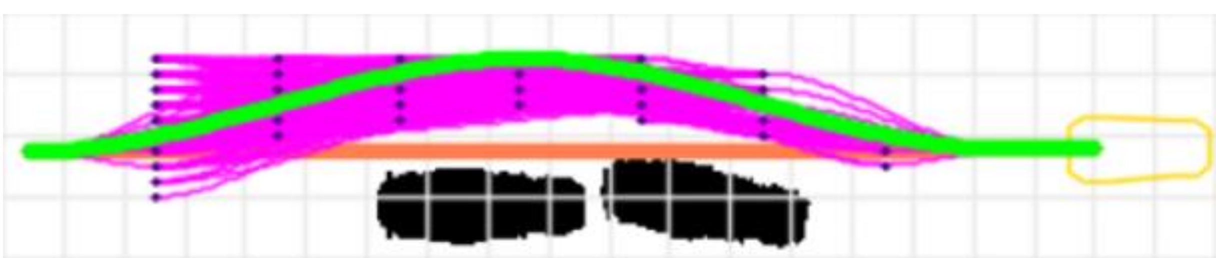
Figure 1: Curvilinear coordinate system. The points $Point_2$ - $Point_{11}$ form single graph layer.





$$W(E_i) = k_{safe} * w_{safe}(E_i) + k_{dist} * w_{dist}(E_i) + k_{man} * w_{man}(E_i)$$

Cost Function for the edges



Shortcomings

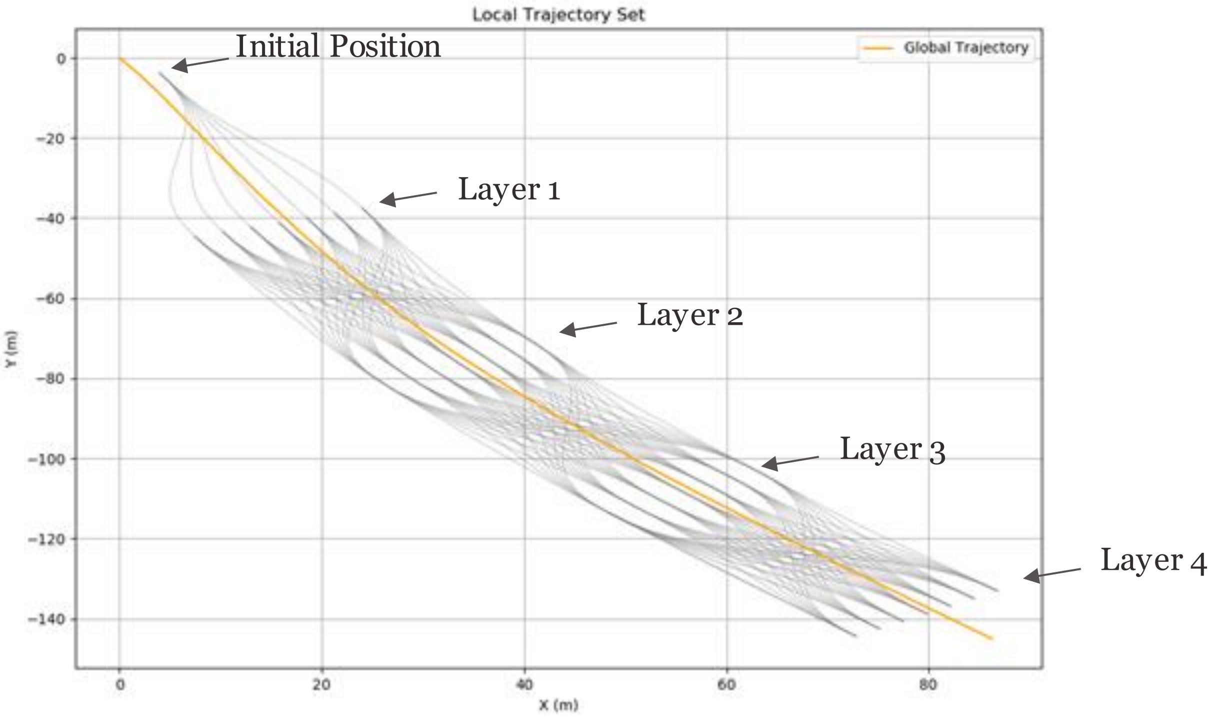
No optimality between the node layers (edges) as there is no particular reason for the use of cubic polynomials.

Viable Solution

Use the Quintic Polynomial Local Path Generation scheme from [1] to create the edges of the graph and obtain the optimal path using the Dijkstra's method as described in [3].

Algorithm

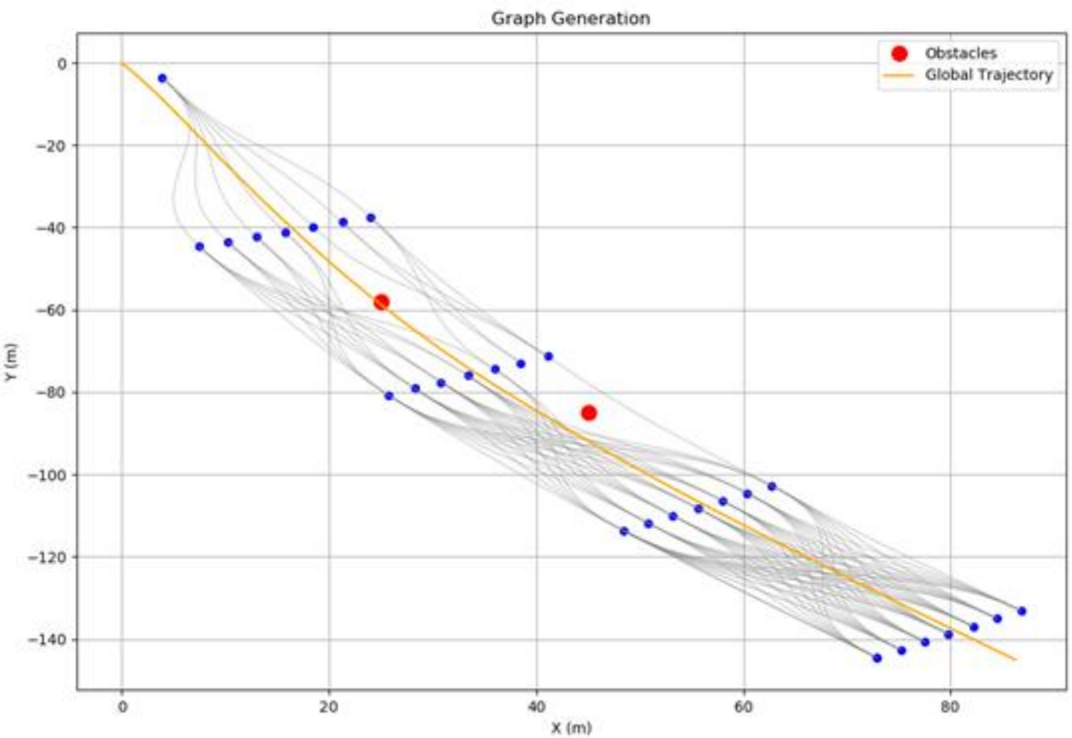
1. From the initial vehicle state, for a given planning time (T), maximum road width, and increment of the road width, a set of quintic polynomial local trajectories are generated as described in [1]. The end points of each local trajectory become the start state for the next iteration. We can thus generate a desired number of 'node layers' as shown-



Algorithm (Contd.)

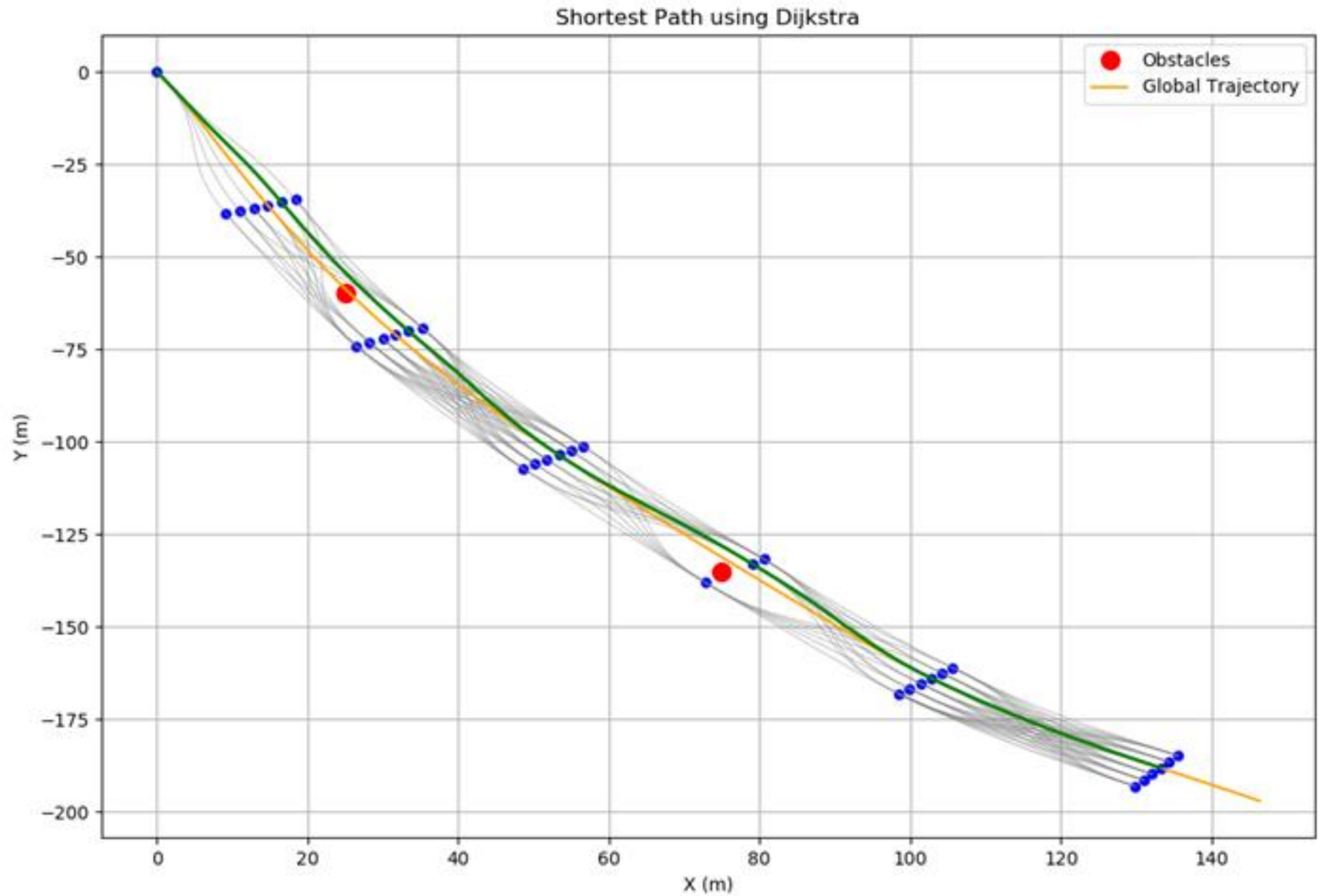
2. The paths that violate the vehicle constraints and obstacle collisions are removed from the set. From the remaining trajectories, each local trajectory is added as an edge with its end points as nodes in a Graph (G). Recall that the weight/cost of the edge is defined as the cost (C) given in [1]-

$$C = k_j J_t(p(t)) + k_t g(T) + k_p h(p_1)$$

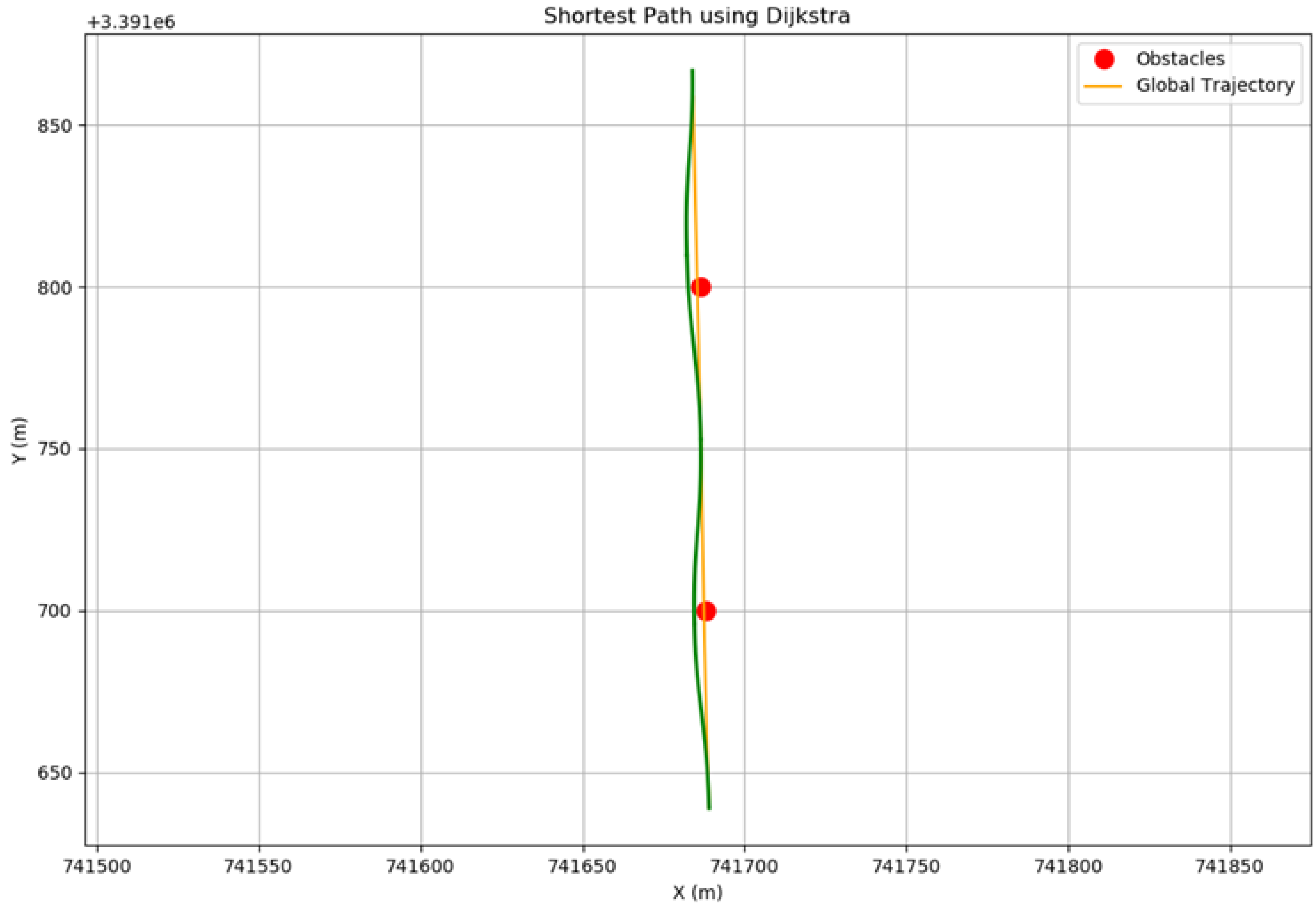


Algorithm (Contd.)

3. Finally, the Shortest Path is obtained using Dijkstra's method.



Results (Deployment on a Self-Driving SUV)



Results (Deployment on a Self-Driving SUV)

Conclusion

The graph search method from [3] is upgraded by incorporating the path generation technique outlined in [1]. This enhancement introduces jerk-minimizing edges in the graph, contrasting with the previous method's reliance on cubic polynomials without considering any optimality features.

Questions?

References

[1] M. Werling, J. Ziegler, S. Kammel and S. Thrun, "Optimal trajectory generation for dynamic street scenarios in a Frenét Frame," 2010 IEEE International Conference on Robotics and Automation, Anchorage, AK, USA, 2010, pp. 987-993, doi: 10.1109/ROBOT.2010.5509799.

[2] A.Takahashi, T.Hongo, Y.Ninomiya, and G.Sugimoto. Local path planning and motion control for AGV in positioning. In IEEE/RSJ International Workshop on Intelligent Robots and Systems '89. The Autonomous Mobile Robots and Its Applications. IROS'89.
Proceedings., pages 392–397,1989.

[3] Kornev, Ivan & Kibalov, Vladislav & Shipitko, Oleg. (2020). Local Path Planning Algorithm for Autonomous Vehicle Based on Multi-objective Trajectory Optimization in State Lattice.

Thank You!