Multi-Agent Path Optimization using Calculus of Variations

Course Project
MATH 146 — Methods of Applied Mathematics

Author: Robert Lee and Vedant Yogishwar
Instructor: Dr. Shiba Biswal

March 20, 2024

Contents

1 List of Symbols **3**

2 Introduction **3**

3 Theory **3**
- 3.1 Problem Definition 3
- 3.2 Objective Function 3
- 3.3 First-Order Necessary Conditions 4
- 3.4 Differential Equation 4

4 Cases **4**

5 Results **4**

6 Discussion **4**

7 Conclusion **4**

8 Acknowledgements **4**

Appendices **4**

A MATLAB Code **4**

Abstract

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

1 List of Symbols

N	number of robots
M	number of circular obstacles
$\mathbf{x}_i$	initial position of the i-th robot
$\mathbf{y}_i$	target position of the i-th robot
$\mathbf{p}_i$	path of the i-th robot
r	radius or repulsive distance of robots
$\mathbf{c}_j$	center of the j-th circular obstacle
R	radius of circular obstacles
F	objective function
L	Lagrangian function
g	barrier function

2 Introduction

3 Theory

3.1 Problem Definition

Add description of problem here.

Let $\mathbf{p}_i \in C^1([0,1])^2$ be the vector valued function corresponding to the path of the i-th robot. Then we have that $\mathbf{p}_i(0) = \mathbf{x}_i$ and $\mathbf{p}_i(1) = \mathbf{y}_i$. The length of the path is defined as

$$l_i = \int_0^1 \left\|\mathbf{p}_i'(t)\right\|^2 dt. \tag{1}$$

3.2 Objective Function

We wish to minimize the total length of all robot paths. Define F as the total path length

$$F(P) = \int_0^1 \sum_{i=1}^{N} \left\|\mathbf{p}_i'(t)\right\|^2 dt, \tag{2}$$

where $P = [\mathbf{p}_1, \ldots, \mathbf{p}_N]^\intercal$ is the path ensemble.

We will modify the objective function in order to represent the no-collision and obstacle constraints. As an example, consider the obstacle constraint. In order to check whether or not a robot collides with an obstacle, a function g can be defined such that it evaluates to infinity if the distance between the robot and the center of the obstacle is less than R and evaluates to 0 when not. However, g would be discontinuous and forbid the use of calculus of variations techniques.

A barrier function $g_{d,\mu}$ is a continuous function that goes to infinity as it approaches a "barrier" value d. A parameter μ can be varied in order for g to approach the discontinuous form. In this case, we wish to devise a function that goes to infinity when approached from above to act as a repulsor. In this project, we will consider two types of barrier functions

1. Log function:

$$g_{d,\mu}(x) = \frac{-1}{\mu}\log(x-d), \quad g_{d,\mu}'(x) = \frac{-1}{\mu(x-d)}. \tag{3}$$

2. Inverse function:

$$g_{d,\mu}(x) = \frac{1}{\mu(x-d)}, \quad g'_{d,\mu}(x) = \frac{-1}{\mu(x-d)^2}. \tag{4}$$

Using this barrier function we can rewrite F as

$$F(P) = \int_0^1 \left[\sum_{i=1}^{N} \left\| \mathbf{p}'_i(t) \right\|^2 + \sum_{i,j=1;i\neq j}^{N} g_{r,\mu_1} \left(\|\mathbf{p}_i(t) - \mathbf{p}_j(t)\|^2 \right) + \sum_{i=1}^{N} \sum_{j=1}^{M} g_{R,\mu_2} \left(\|\mathbf{p}_i(t) - \mathbf{c}_j\|^2 \right) \right] dt, \tag{5}$$

and the optimization problem becomes an unconstrained optimization problem.

3.3 First-Order Necessary Conditions

To derive first-order necessary condition, we first identify the Lagrangian L,

$$L(Y,Z) = \sum_{i=1}^{N} \|\mathbf{z}_i\|^2 + \sum_{i,j=1;i\neq j}^{N} g_{r,\mu_1} \left(\|\mathbf{y}_i - \mathbf{y}_j\|^2 \right) + \sum_{i=1}^{N} \sum_{j=1}^{M} g_{R,\mu_2} \left(\|\mathbf{y}_i - \mathbf{c}_j\|^2 \right). \tag{6}$$

Applying the Euler–Lagrange equation w.r.t. the x-component of the i-th robot,

$$\frac{d}{dt} \left(\mathbf{p}'_{i,x} \right) = \sum_{j\neq i} g'_{r,\mu_1} \left(\|\mathbf{p}_i - \mathbf{p}_j\|^2 \right) (\mathbf{p}_{i,x} - \mathbf{p}_{j,x}) + \sum_{j=1}^{M} g'_{R,\mu_2} \left(\|\mathbf{p}_i - \mathbf{c}_j\|^2 \right) (\mathbf{p}_{i,x} - \mathbf{c}_{j,x}). \tag{7}$$

An analogous equation can be derived w.r.t. its y-component by replacing x with y.

3.4 Differential Equation

The Euler-Lagrange equation derived in Equation (7) provides a system of differential equations that govern the motion of each robot's path.

4 Cases

5 Results

6 Discussion

7 Conclusion

8 Acknowledgements

Appendices

A MATLAB Code