

Optimization of a trajectory AB inside a 10x10 room by avoiding collisions against obstacles using MATLAB CORO & EMARO (Master I)

Description

Consider a room of size 10x10. The room has obstacles in the form of four circles of definite radii and centers. Two rectangular walls, each of size 4x2 with their origins at (4, 0) and (4, 6) separates the test region into two rooms. A trajectory AB has to travel from the left side of the room (preferably from the top) to the right side of the room (preferably to the bottom). The trajectory has to pass through 4 to 7 intermediate points before it has to reach its destination B. The objective of the project is to optimize this trajectory containing the intermediate points (guess points) in such a way that they pass either tangentially to the circles or away from the circles as well as pass in between the walls as it tries to go from the left to the right side of the room. The algorithm and report for the project must satisfy the below requirements:

- The discretization has to be done for each individual segments of the trajectory with n number of points. For example, if there is one guess point, the discretization has to be done between A and the guess point followed by discretization between the guess point and final point B. If multiple guess points are initialized, successful discretization on the segments between the guess points has to be done.
- The objective function should be written which will aim to minimize the sum of the norms of the individual segments.
- Use *fmincon* that calls the objective function and the constraint function (explained in next step) in order to find the optimal trajectory that meets the project requirement.
- The constraint function sequence should be done in the following ways:
 - Implement a constraint function inside the room having just one circle and one guess point.
 - Introduce 2 circles with 5 intermediate points and test the results.
 - Introduce 4 circles with 7 intermediate points and test the results.
 - The rectangular walls can now be introduced with two set of circles on the left side of the room and two others on the right side. Additional constraints must be defined in order to make the trajectory pass perfectly in the area between the walls.
- Plot all the results with the room and the circle, representing the optimal as well as the non-optimal trajectory for all the constraint function sequences defined above. The labels *A* and *B* can be represented in the plot as well a number can be assigned to the earlier and the optimized positions of the guess points. An example of the final result is shown below in Figure 1.

Comment about the optimization method individually for all sequences. Briefly explain at the end the methodology of *fmincon* and how the solution obtained can be improved (Alternate approaches).

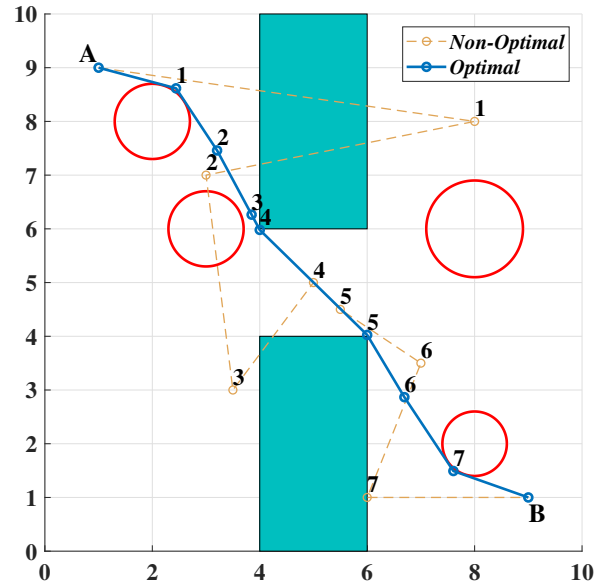


Figure 1: Example of the final solution representing the initial guess points and the optimal points

Some useful information

- The radii and the centers for the circles have to be chosen by the user. They have to fit inside the test space and must have a meaningful representation.
- The intermediate guess points has to be decided by the user. The initial and final points of the trajectory A and B can preferably be $(1,9)$ and $(9,1)$ for the final constraint definition.
- The iteration definition in `optimset` has to be carefully defined especially the tolerances in order to have a better final result.
- The following MATLAB functions can be useful for the project: `norm`, `viscircles`, `rectangle`, `num2str`, `grid on`, `text`, `legend`. Structures can also be used to reduce the lines of codes.

Additional questions

- Employ random intermediate points reducing the work from user for fixing the guess points.
- The discretization technique can no longer be used. With 5 guess points and the trajectory A-B, implement the final constraint function and try to obtain the optimal solution.