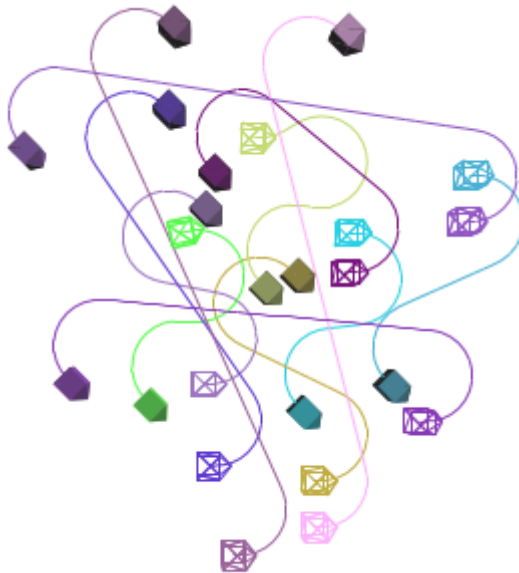


Multi-Robot Path Planning on Predefined Routes

Problem Definition

A set of robots that are instructed to move along specified paths. The paths of the robots are subject to change in the event of an interruption during their movement. A robot N is defined in the 2-D configuration space C such that:

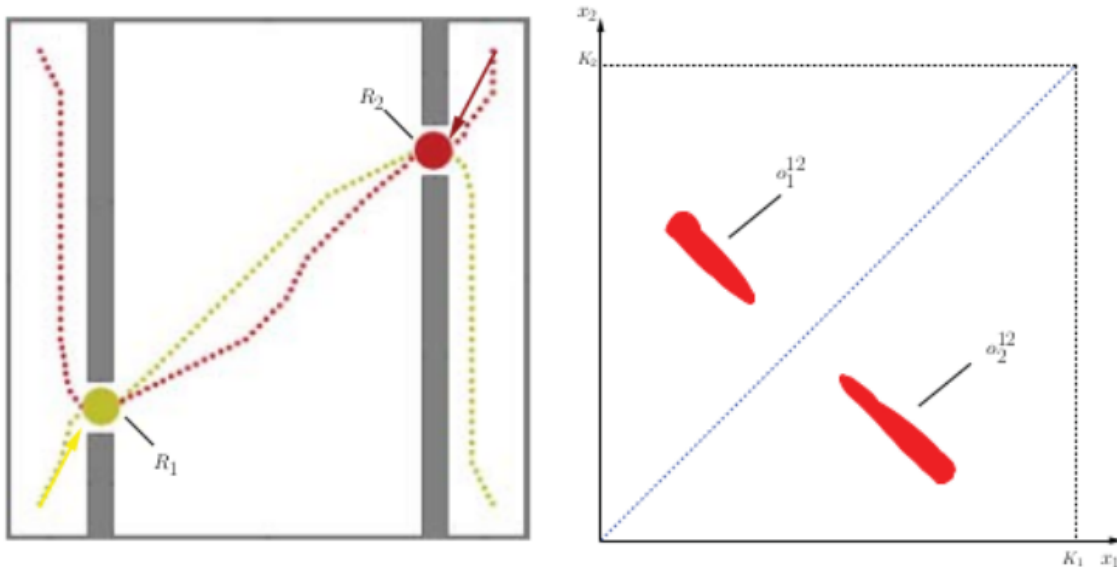
- Every robot follows a path π_n .
- A path is a sequence of adjacent segments and each segment $s_i \in [0, l(\pi_i)]$, where $l(\pi_i)$ is the total length of the path.
- Finally, the space occupied by a robot in configuration x_i is represented as $A(x_i)$.



Methodology

Coordination Diagram

Coordination diagrams represent all possible configurations and collision regions along a robots path. Finding an obstacle free map from $(0,0)$ to $(1,1)$ in a coordination diagram provides a solution to the coordination problem, therefore they are commonly used in path coordination tasks. They were first introduced in [1].



Constructing the Diagram: Collision Zones

In order to construct the diagram, collision zones can be defined as in [2]:

$$X_{ij}^{coll} = \{ (s_1, \dots, s_N) \in S \mid A(\pi_i(s_i)) \cap A(\pi_j(s_j)) \neq \emptyset \}$$

A Simpler Approach

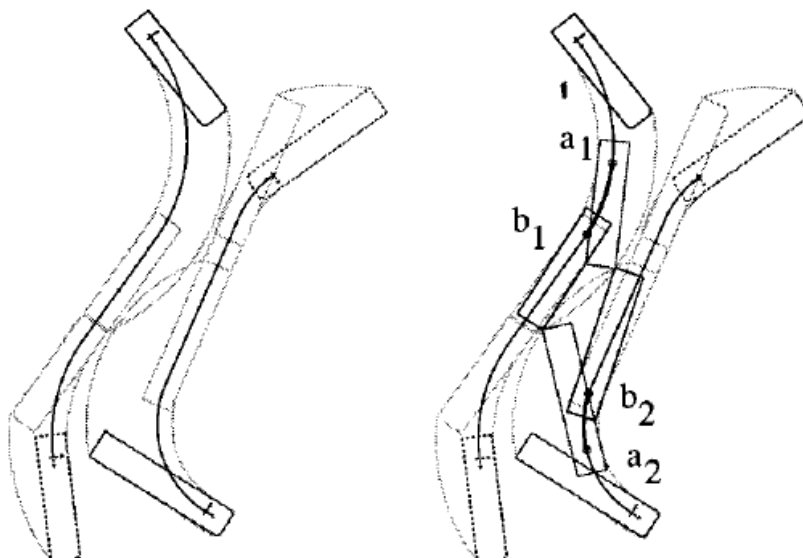
If the robots assumed to be circular in shape and equal in size, the formulation above can be simplified to as in [3]:

$$X_{ij}^{coll} = \{ (s_1, \dots, s_N) \in S \mid \|\pi_i(s_i) - \pi_j(s_j)\| \geq 2r \}$$

Where r is the radius of a robot.

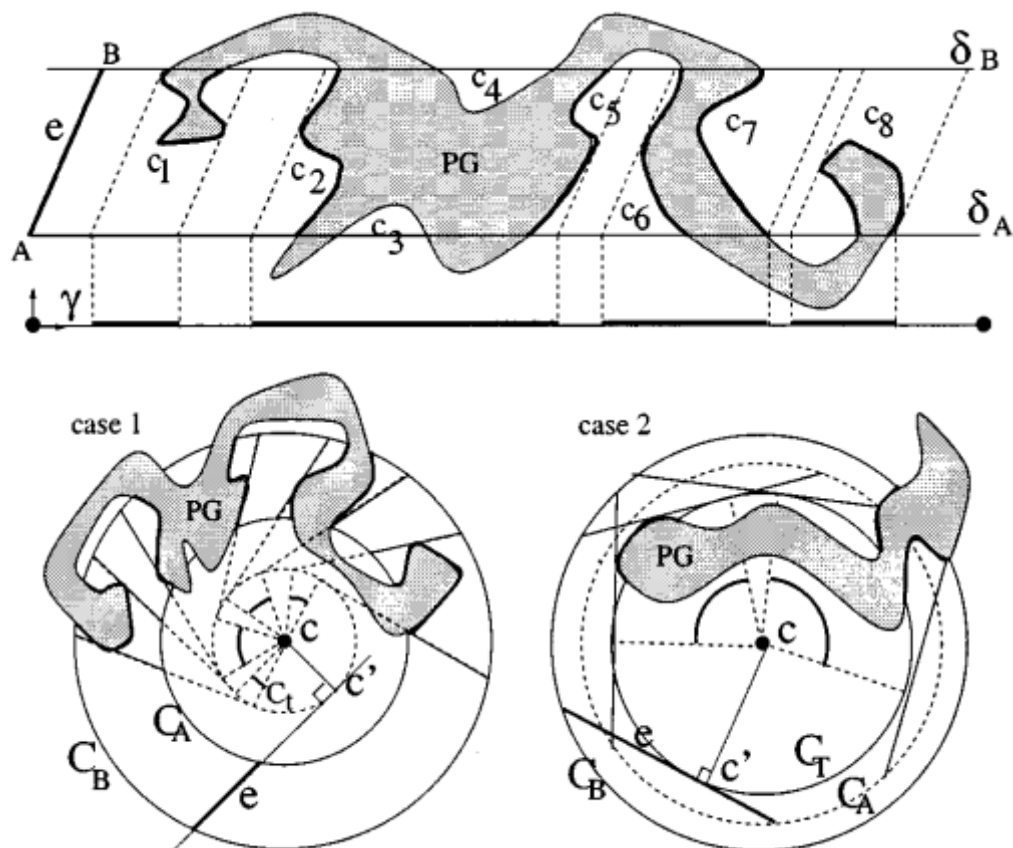
Traces

In order to get a more accurate representation of the collision space, the traces that robots create during their path can be examined as was done in [4].



Approach 1: Spatial Discretization

[4] mentions that the robots in question are considered to be convex polygonal robots that move along paths that are either straight line segments or arcs of a circle. Utilizing this property, it is possible to decompose the path into elementary collusion subpaths and compute separate coordination diagrams for decomposed path segments and combine them later.



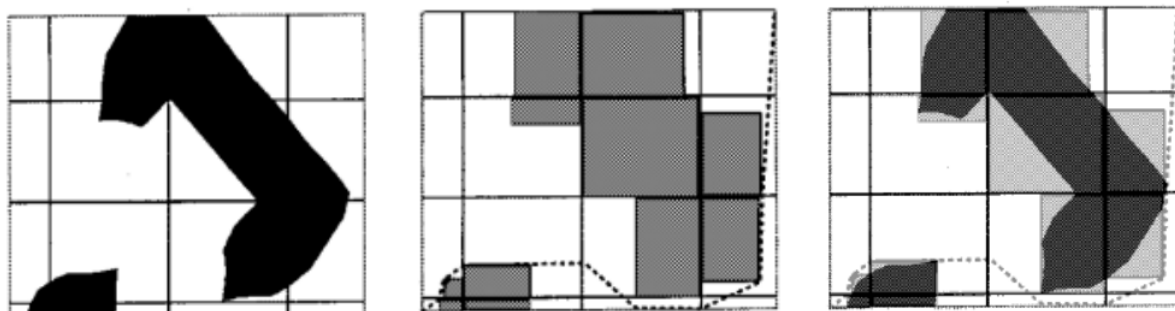
Complexity of the collusion zone finding algorithm in [4]: $O(n + k \log(k))$ where n represents amount of curves and k represents amount of parts which is half of the intersection amount $2k$.

Approach 2: Temporal Discretization

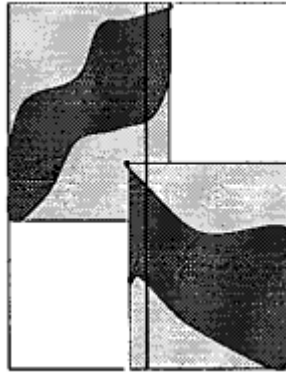
[9] suggests a method that is more precise than the method mentioned above.

Constructing the Diagram: Bounded Box Representation

With the bounding box approach an exact computation of the obstacles is not needed.



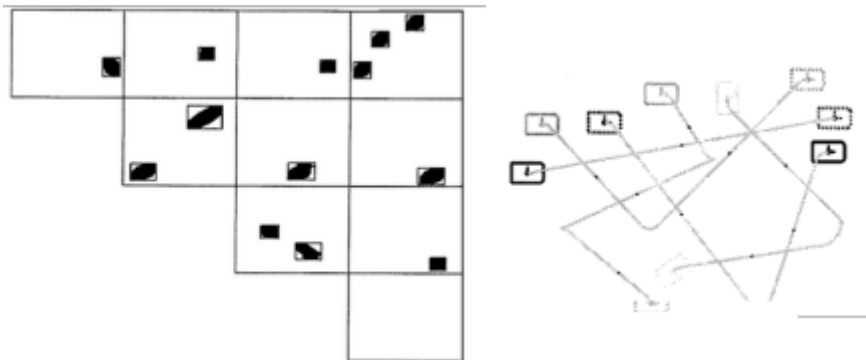
However, this method has its setbacks as an existing solution can be lost if there exist one vertical and one horizontal line intersecting two obstacles as seen below.



Constructing the Diagram: The n-Dimension Problem

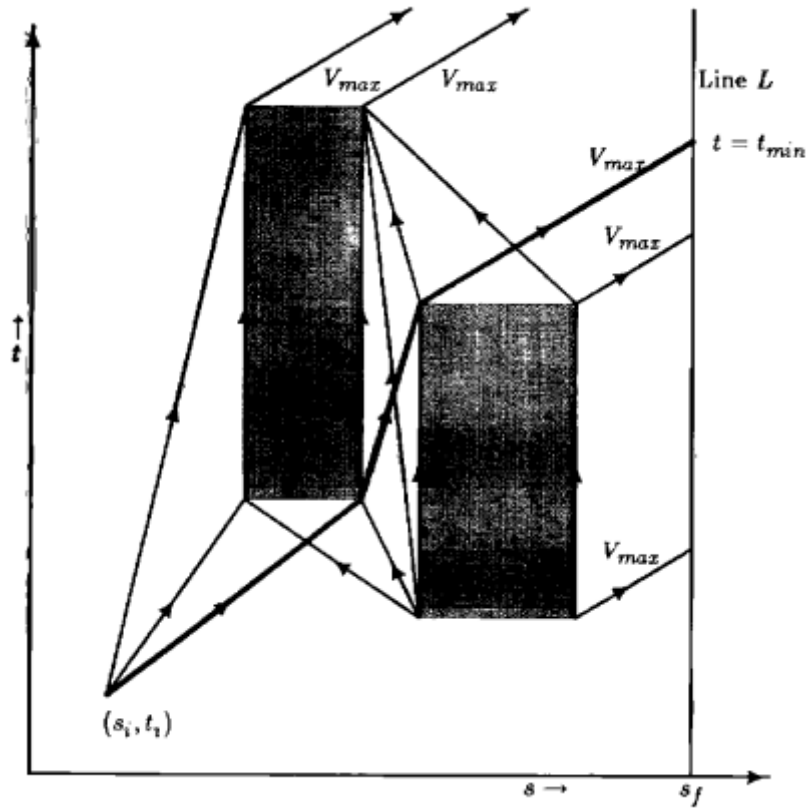
When n robot paths are considered, the coordination diagram that would represent the collusion set becomes an n -dimensional cube. Therefore, an obstacle would be represented as a collection of points (s_1, s_2, \dots, s_n) where at least two robots collide. Computation of multi-dimensional obstacles is costly and hard to implement.

The n -dimensional space can instead be represented as a 2-D space by computing 2-D diagrams of each pair of robots. For n robots $n(n-1)/2$ diagrams will be produced as shown in [4]. One such representation can be seen below where 5 robots occupy a coordination space.



Coordination Planner

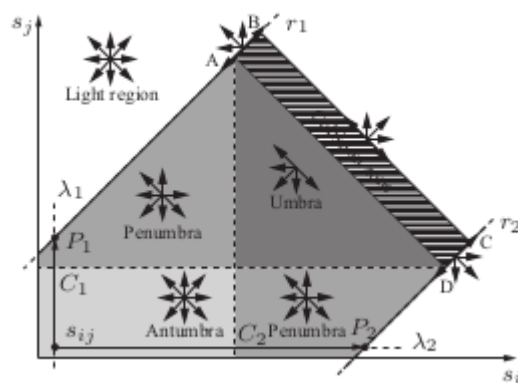
After the coordination diagram is drawn, the optimal coordination path needs to be determined. The problem becomes that of finding a path in a directed graph. Algorithms can be selected to minimize, or fix the arrival time as in [8].



[5] separates the regions on a coordination diagram to put a constraint on the possible paths so that inefficient and blocking actions are filtered out.

- Light region: All actions are possible.
- Antumbra region: Close to collision, all actions are permitted as long as there is no change in region.
- Penumbra region: One robot has entered the path of another robot.
- Umbra region: Robots have reached a common portion of the map.
- Obstacle border: Forbidden.

Note that in a bounding box representation Umbra region would be discarded.



Supplementary Findings

Unexpected Events

In the context of a warehouse, calculating the coordination path again at every change in the overall navigation would be inefficient as there can be constant changes in path-finding due to interruptions. In such a case, an incremental algorithm that makes changes "on the fly" can be more desirable. The algorithm in [5] aims to create such a system that can predict and avoid such interruptions.

Addition of Roadmaps

Introduction of predetermined roadmaps to the system can increase the computation efficiency. This is done by computing sub-coordination diagrams offline and creating the coordination diagram online by combining the necessary sub-coordination diagrams. Depending on the given configuration of robots and paths a new coordination diagram can be produced much faster this way.

Undirected Graph Approach

Many methods such as [6] utilize an undirected graph structure that symbolizes the coordination field. While this approach is not applicable to the problem statement, it can be utilized in certain scenarios where robots would be constrained to use paths that are suitable only for a single robot to pass at a time such as tight corridors.

Kinematics

To be considered later. [7] could provide insight.

References

- [1] Deadlock Free and Collision-free Coordination of Two Robot Manipulators
- [2] Coordination of Multiple AGVs in an Industrial Application
- [3] Online Plan Repair in Multi-robot Coordination with Disturbances
- [4] Path Coordination for Multiple Mobile Robots: A Resolution-Complete Algorithm
- [5] Coordination of Multiple AGVs in an Industrial Application
- [6] Coordination of Multiple Robots along Given Paths with Bounded Junction Complexity
- [7] Coordinating Multiple Robots with Kinodynamic Constraints Along Specified Paths
- [8] Toward Efficient Trajectory Planning: The Path-Velocity Decomposition
- [9] Time-optimal Coordination of Mobile Robots along Specified Paths