Coordinating Multiple Robots with Kinodynamic Constraints along Specified Paths

Jufeng Peng

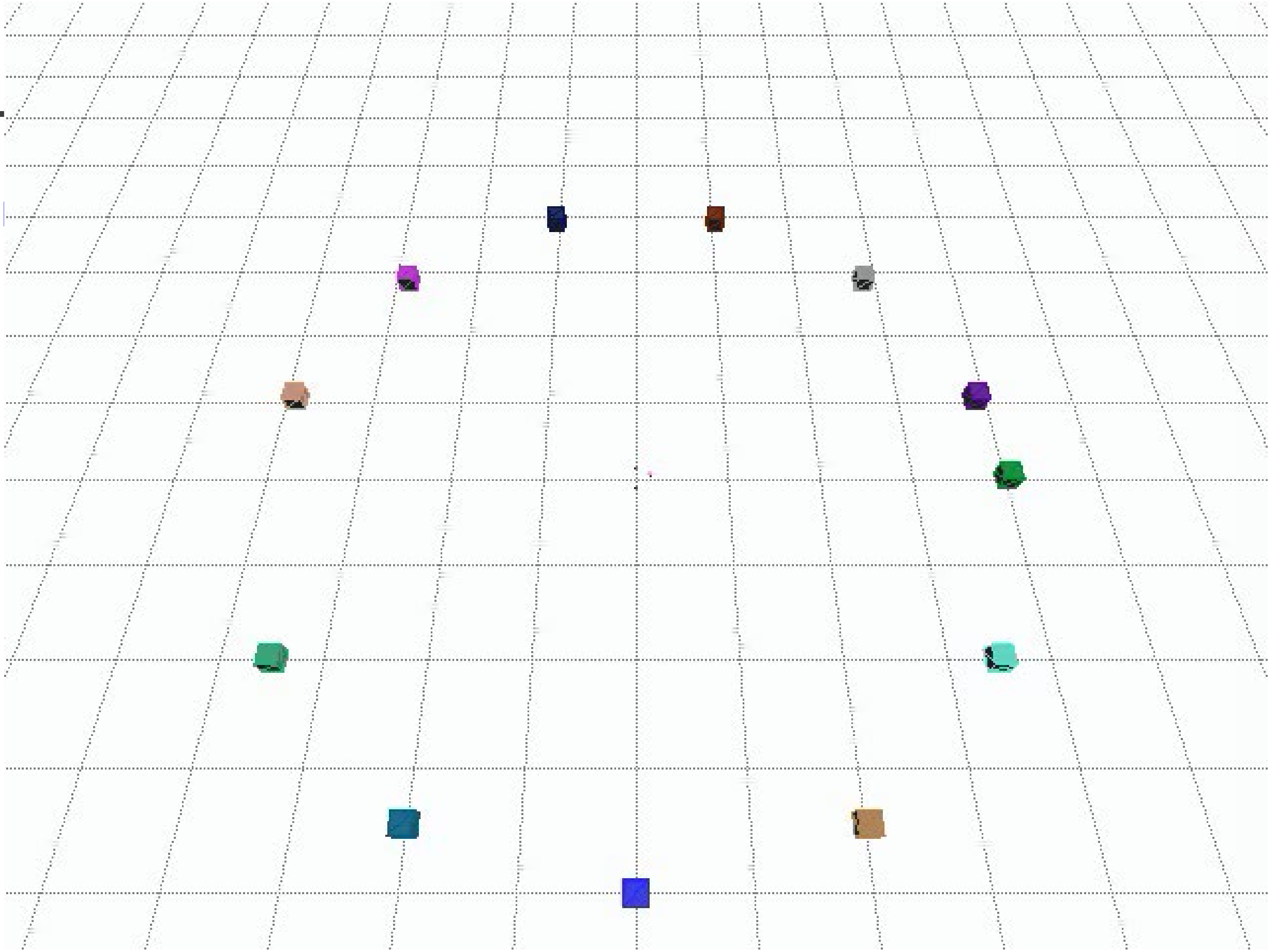
Srinivas Akella

Rensselaer Polytechnic Institute

Coordination Problem

- **Given:** Multiple robots with specified paths
- **Find:** Continuous velocity coordination schedule that is minimum time and collision-free

12 Robot Example



Motivation

- AGVs in factories, harbors, and airports

- Manufacturing cells (RobotWorld, Minifactory, welding and painting robots)
- Air traffic control

Overview

- Robots are modeled as double integrators; paths divided into collision and collision-free segments
- Optimal continuous velocity schedule formulated as mixed integer nonlinear program (MINLP)

  Difficult to solve!
- Upper and lower bounds found by solving two mixed integer linear programming (MILP) formulations
- Upper bound formulation gives a continuous velocity schedule

Related Work

- **Motion planning for multiple robots:** Hopcroft, Schwartz, Sharir (1984); Erdmann and Lozano-Perez (1987); Barraquand, Langlois, and Latombe (1992); Svestka and Overmars (1996); Aronov et al. (1999); Sanchez and Latombe (2002)
- **Single robot among moving obstacles:** Reif and Sharir (1985); Kant and Zucker (1986)
- **Path coordination:** O'Donnell and Lozano-Perez (1989); LaValle and Hutchinson (1998); Simeon, Leroy, and Laumond (2002)

  Trajectory coordination: Akella and Hutchinson (2002)

Related Work (cont.)

- **Trajectory planning:** Bobrow, Dubowsky, Gibson (1985); Shin and McKay (1985); Sahar and Hollerbach (1986); Shiller and Dubowsky (1989); Canny, Rege, Reif (1991); Donald et al. (1993); Reif and Wang (1997); Fraichard (1999); LaValle and Kuffner (2001); Hsu et al. (2001)
- **Trajectory coordination of two robots:** Lee and Lee (1987); Bien and Lee (1992); Chang, Chung, and Lee (1994); Shin and Zheng (1992);
- **Air Traffic Control:** Tomlin, Pappas, Sastry (1998); Bicchi and Pallottino (2000); Schouwenaars et al. (2001); Pallottino, Feron, and Bicchi (2002);

Multiple Robot Coordination Problem

- **Given:** A set of robots $\{A_1, \ldots, A_n\}$ with specified paths
- **Find:** Continuous velocity profiles that minimize completion time and avoid collisions

Path γ_i for robot A_i is a curve in configuration space, parameterized by s_i

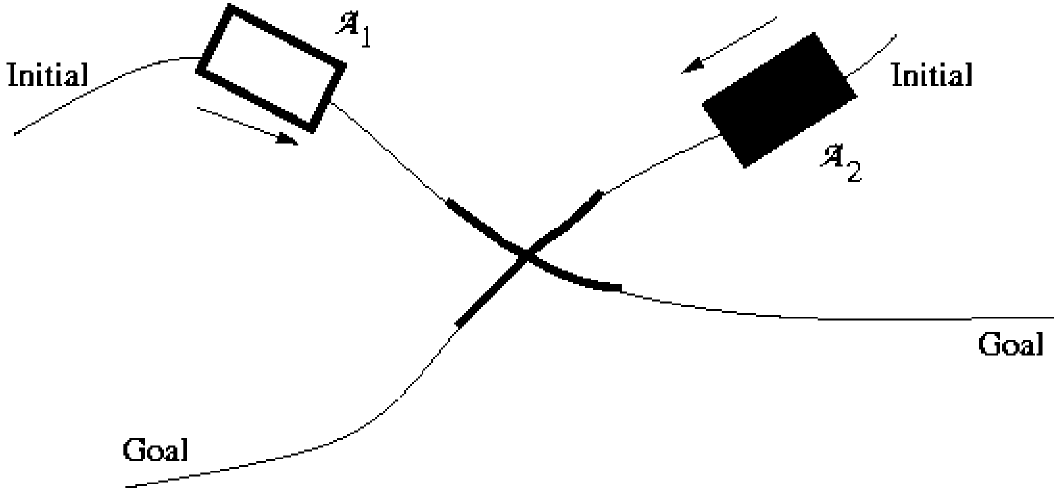
Assumptions

- Robot paths are specified, and are free of static obstacles
- Initial and goal configurations of robots are collision-free
- Each robot moves monotonically along its path
- Each path is sufficiently long for robot to attain maximum velocity v_{max}

Collision Segments

- A collision segment for robot A_i with robot A_j is a contiguous interval of path positions s_i such that

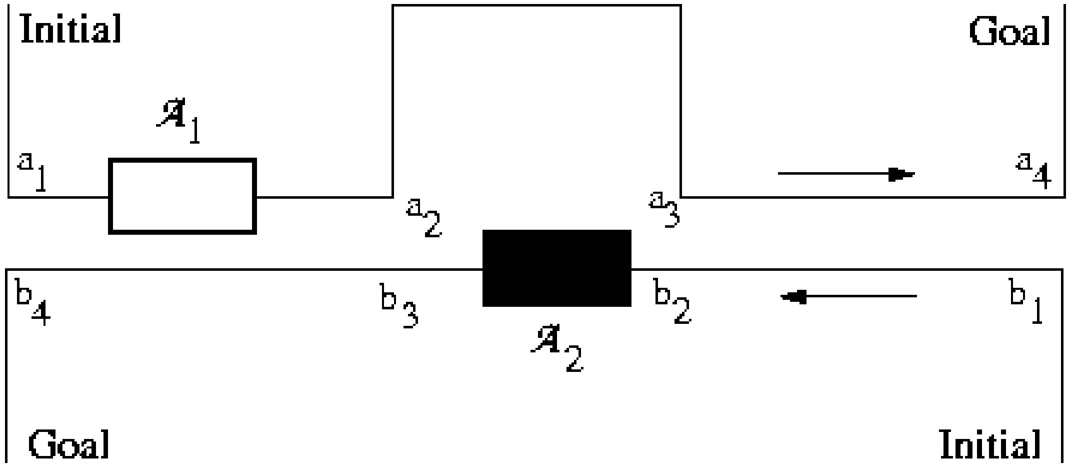
$$\mathcal{A}_i(\gamma_i(s_i)) \cap \mathcal{A}_j(\gamma_j(s_j)) \neq \emptyset$$



- Paths are divided into collision segments and collision-free segments

Collision Zones

- A collision zone is an ordered pair of maximal collision segments $([s_i^{start}, s_i^{end}], [s_j^{start}, s_j^{end}])$ s.t. any point in one interval results in a collision with at least one point in the other interval



- Collision zones are $([a_1,a_2],[b_3,b_4])$, $([a_3,a_4],[b_1,b_2])$

Sufficient Conditions for Collision-free Scheduling

- To avoid collisions between A_i and A_j, sufficient to ensure A_i and A_j are not simultaneously in a collision zone

- **Collision avoidance constraints** are:

  $t_{jh} \geq t_{i(k+1)}$ ($\mathcal{A}_j$ enters h after $\mathcal{A}_i$ exits k)

  or

  $t_{ik} \geq t_{j(h+1)}$ ($\mathcal{A}_i$ enters k after $\mathcal{A}_j$ exits h)

  t_{ik} : time when A_i begins moving along its kth segment

Simplified Coordination Problem

Assume robots can start and stop instantaneously

- **Given:** A set of robots with specified paths
- **Find:** Velocity profiles to minimize completion time so there are no collisions

Instantaneous Model

- A robot in motion always moves at its maximum velocity v_{max}
- Robots have infinite acceleration and deceleration so they can start and stop instantaneously

- Model yields discontinuous velocity profiles
- Provides a lower bound on the optimal schedule

Segment Traversal Times

- Let τ_{ik} be traversal time for robot A_i to pass through segment k

$$t_{i(k+1)} = t_{ik} + \tau_{ik}$$

- Minimum and maximum traversal times for A_i to traverse segment of length S_{ik}

$$\Delta T_{ik}^{min} = S_{ik}/v_{i,max}$$

$$\Delta T_{ik}^{max} = \infty$$

- **Traversal time constraints** are:

$$\Delta T_{ik}^{max} \geq \tau_{ik} \geq \Delta T_{ik}^{min}$$

Instantaneous Model: MILP Formulation

Minimize C_{max}
subject to:

$C_{max} \geq t_{i,last} + \tau_{i,last}$ for $i = 1, \ldots, n$

$t_{ik} \geq 0$

$t_{i(k+1)} = t_{ik} + \tau_{ik}$

$\Delta T_{ik}^{max} \geq \tau_{ik} \geq \Delta T_{ik}^{min}$

$t_{jh} - t_{i(k+1)} + M(1 - \delta_{ijkh}) \geq 0$

$t_{ik} - t_{j(h+1)} + M\delta_{ijkh} \geq 0$

$\delta_{ijkh} \in \{0, 1\}$

δ_{ijkh} is 1 if $\mathcal{A}_i$ goes first along its kth segment and 0 if $\mathcal{A}_j$ goes first along its hth segment

Back to Original Coordination Problem

- Robot velocity profiles must be continuous, and satisfy velocity and acceleration bounds

- Model robot as a double integrator (Bryson and Ho, 1975)

Single Robot on a Segment

- Find min and max traversal times for double integrator:

Minimize or Maximize $\Delta T = \int_0^{\Delta T} 1 dt$

subject to:

$$\begin{pmatrix} \dot{x} \\ \dot{v} \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} x \\ v \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \end{pmatrix} a(t)$$

$$x(0) = 0 \qquad x(\Delta T) = S$$

$$v(0) = v_{start} \qquad v(\Delta T) = v_{end}$$

$$0 \leq v \leq v_{max}$$

$$-a_{max} \leq a \leq a_{max}$$

Minimum Time Cases

- **Case 1:** Robot can reach max velocity

If $S \geq \frac{1}{2}\left(\frac{(v_{max}^2 - v_{start}^2)}{a_{max}} + \frac{(v_{max}^2 - v_{end}^2)}{a_{max}}\right)$,

$$\Delta T^{min} = \frac{S}{v_{max}} - \frac{((v_{max}^2 - v_{start}^2) + (v_{max}^2 - v_{end}^2))}{2a_{max} \cdot v_{max}} + \frac{v_{max} - v_{start}}{a_{max}} + \frac{v_{max} - v_{end}}{a_{max}}$$

- **Case 2:** Robot cannot reach max velocity

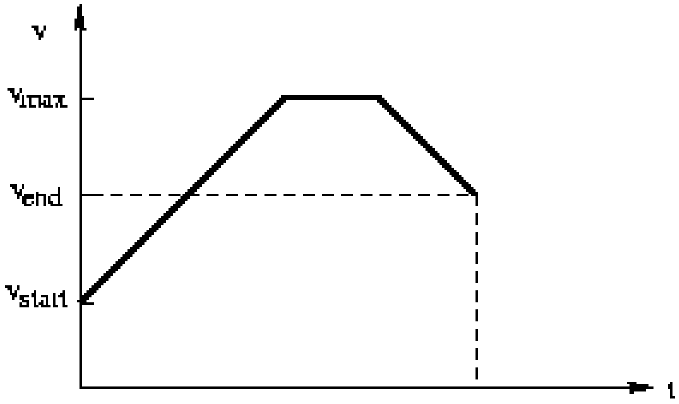
If $\frac{1}{2}\left(\frac{(v_{max}^2 - v_{start}^2)}{a_{max}} + \frac{(v_{max}^2 - v_{end}^2)}{a_{max}}\right) > S \geq \frac{1}{2}\frac{|v_{end}^2 - v_{start}^2|}{a_{max}}$,

$$\Delta T^{min} = \frac{(v_{middle} - v_{start})}{a_{max}} + \frac{(v_{middle} - v_{end})}{a_{max}}$$

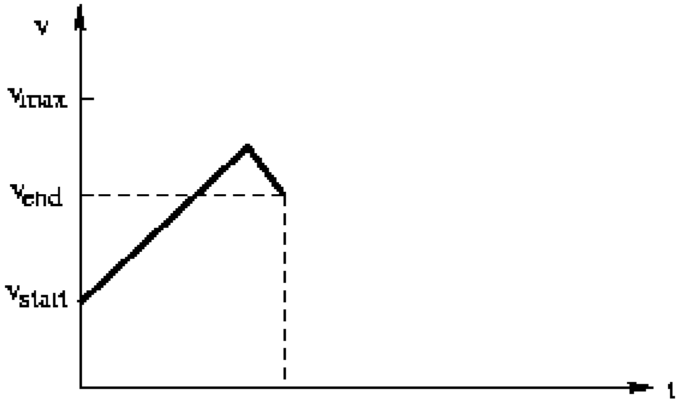
where $v_{middle} = \frac{1}{2}(2v_{start}^2 + 2v_{end}^2 + 4Sa_{max})^{\frac{1}{2}}$

Minimum Time Cases

- Case 1



Case 2



Maximum Time Cases

- **Case 1:** Robot can reach zero velocity:

  If $S \geq \frac{1}{2}\frac{(v_{start}^2+v_{end}^2)}{a_{max}}, \quad \Delta T^{max} = \infty.$

- **Case 2:** Robot cannot reach zero velocity

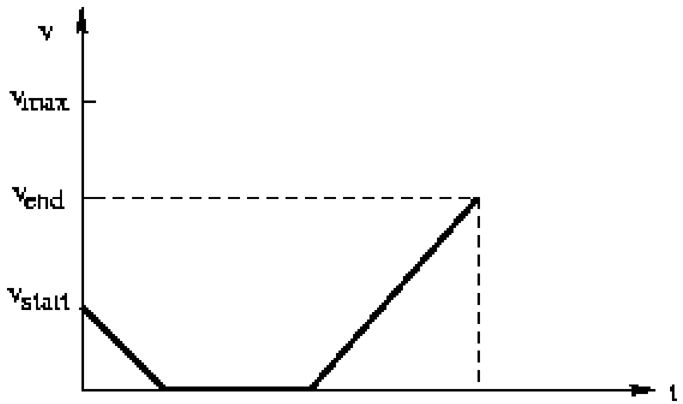
  If $\frac{1}{2}\frac{(v_{start}^2+v_{end}^2)}{a_{max}} > S \geq \frac{1}{2}\frac{|(v_{end}^2-v_{start}^2)|}{a_{max}},$

  $$\Delta T^{max} = \frac{(v_{start}-v_{middle})}{a_{max}} + \frac{(v_{end}-v_{middle})}{a_{max}}$$

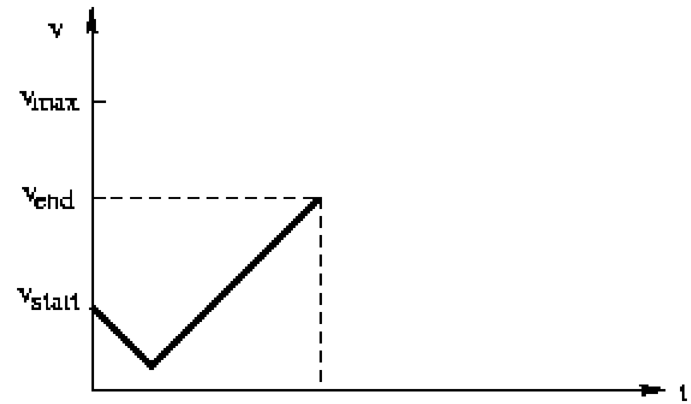
  where $v_{middle} = \frac{1}{2}(2v_{start}^2+2v_{end}^2-4Sa_{max})^{\frac{1}{2}}$

Maximum Time Cases

- Case 1



Case 2



Continuous Velocity Schedule: MINLP Formulation

Minimize C_{max}
subject to:

$C_{max} \geq t_{i,last} + \tau_{i,last}$ for $i = 1, \ldots, n$

$t_{ik} \geq 0$

$t_{i(k+1)} = t_{ik} + \tau_{ik}$

$\Delta T_{ik}^{max} \geq \tau_{ik} \geq \Delta T_{ik}^{min}$

$t_{jh} - t_{i(k+1)} + M(1 - \delta_{ijkh}) \geq 0$

$t_{ik} - t_{j(h+1)} + M\delta_{ijkh} \geq 0$

$\delta_{ijkh} \in \{0, 1\}$

$v_{i,max} \geq v_{ik} \geq 0$

$v_{i,initial} = v_{i,goal} = 0$

$$S_{ik} \geq \frac{(v_{i(k+1)}^2 - v_{ik}^2)}{2a_{i,max}} \geq -S_{ik}$$

$$(S_{ik} - \frac{(v_{i,max}^2 - v_{ik}^2) + (v_{i,max}^2 - v_{i(k+1)}^2)}{2a_{i,max}}) - My_{ik,1} \leq 0$$

$$(S_{ik} - \frac{(v_{i,max}^2 - v_{ik}^2) + (v_{i,max}^2 - v_{i(k+1)}^2)}{2a_{i,max}}) + My_{ik,2} \geq 0$$

$$\Delta T_{ik,1}^{min} = \frac{S_{ik}}{v_{i,max}} - \frac{(v_{i,max}^2 - v_{ik}^2 + v_{i,max}^2 - v_{i(k+1)}^2)}{2a_{i,max}v_{i,max}} + \frac{v_{i,max} - v_{ik}}{a_{i,max}} + \frac{v_{i,max} - v_{i(k+1)}}{a_{i,max}}$$

$$\Delta T_{ik,2}^{min} = \frac{(v_{middle,ik}^{min} - v_{ik})}{a_{i,max}} + \frac{(v_{middle,ik}^{min} - v_{i(k+1)})}{a_{i,max}}$$

$$(v_{middle,ik}^{min})^2 = \frac{1}{4}(2v_{ik}^2 + 2v_{i(k+1)}^2 + 4S_{ik}a_{i,max})$$

$$\Delta T_{ik}^{min} = y_{ik,1} \cdot \Delta T_{ik,1}^{min} + y_{ik,2} \cdot \Delta T_{ik,2}^{min}$$

$$y_{ik,1} + y_{ik,2} = 1 \qquad y_{ik,1}, y_{ik,2} \in \{0, 1\}$$

$$(S_{ik} - \frac{v_{ik}^2 + v_{i(k+1)}^2}{2a_{i,max}}) - Mz_{ik,1} \leq 0$$

$$(S_{ik} - \frac{v_{ik}^2 + v_{i(k+1)}^2}{2a_{i,max}}) + Mz_{ik,2} \geq 0$$

$$\Delta T_{ik,1}^{max} = \infty$$

$$\Delta T_{ik,2}^{max} = \frac{(v_{ik} - v_{middle,ik}^{max})}{a_{i,max}} + \frac{(v_{i(k+1)} - v_{middle,ik}^{max})}{a_{i,max}}$$

$$(v_{middle,ik}^{max})^2 = \frac{1}{4}(2v_{ik}^2 + 2v_{i(k+1)}^2 - 4S_{ik}a_{i,max})$$

$$\Delta T_{ik}^{max} = z_{ik,1} \cdot \Delta T_{ik,1}^{max} + z_{ik,2} \cdot \Delta T_{ik,2}^{max}$$

$$z_{ik,1} + z_{ik,2} = 1 \qquad z_{ik,1}, z_{ik,2} \in \{0,1\}$$

- Gives optimal continuous velocity schedule
- Difficult to solve this MINLP!

Bounding Optimum Schedule

Idea: Approach optimum schedule by bounding it from above and below

- **Upper bound**: Set velocity at segment endpoints to be max possible velocity

  Velocity profile is continuous
- **Lower bound**: Use improved instantaneous model

Setpoint Model (Upper Bound)

- Velocity v_{ik} at segment endpoints is set to max possible velocity that satisfies velocity and acceleration constraints

  Gives a continuous velocity profile
- Any continuous velocity schedule is guaranteed to be an upper bound on optimal continuous velocity schedule

Setpoint Model

For clarity, assume first and last segments are sufficiently long for robot to accelerate to v_{max} and to decelerate to zero resp.

$$\Delta T_{ik}^{min} = \begin{cases} S_{ik}/v_{i,max} & \text{if interior segment} \\ S_{ik}/v_{i,max} + v_{i,max}/2a_{i,max} & \text{if first or last segment} \end{cases}$$

$$\Delta T_{ik}^{max} = \begin{cases} \infty & \text{if } S_{ik} \geq v_{i,max}^2/a_{i,max} \\ \frac{2v_{i,max} - 2(v_{i,max}^2 - a_{i,max}S_{ik})^{\frac{1}{2}}}{a_{i,max}} & \text{if } S_{ik} < v_{i,max}^2/a_{i,max} \end{cases}$$

Improved Instantaneous Model (Lower Bound)

- Tighten lower bound by considering time to accelerate to max velocity $v_{\max}$, and to decelerate to zero

  So minimum traversal times are now identical to those of setpoint model

- MILP formulation identical to setpoint model, except for ΔT^{max} values

MILP Formulations

- Setpoint and improved instantaneous formulations differ only in ΔT^{max} values

Minimize C_{max}

subject to:

$C_{max} \geq t_{i,last} + \tau_{i,last}$ for $i = 1, \ldots, n$

$t_{ik} \geq 0$

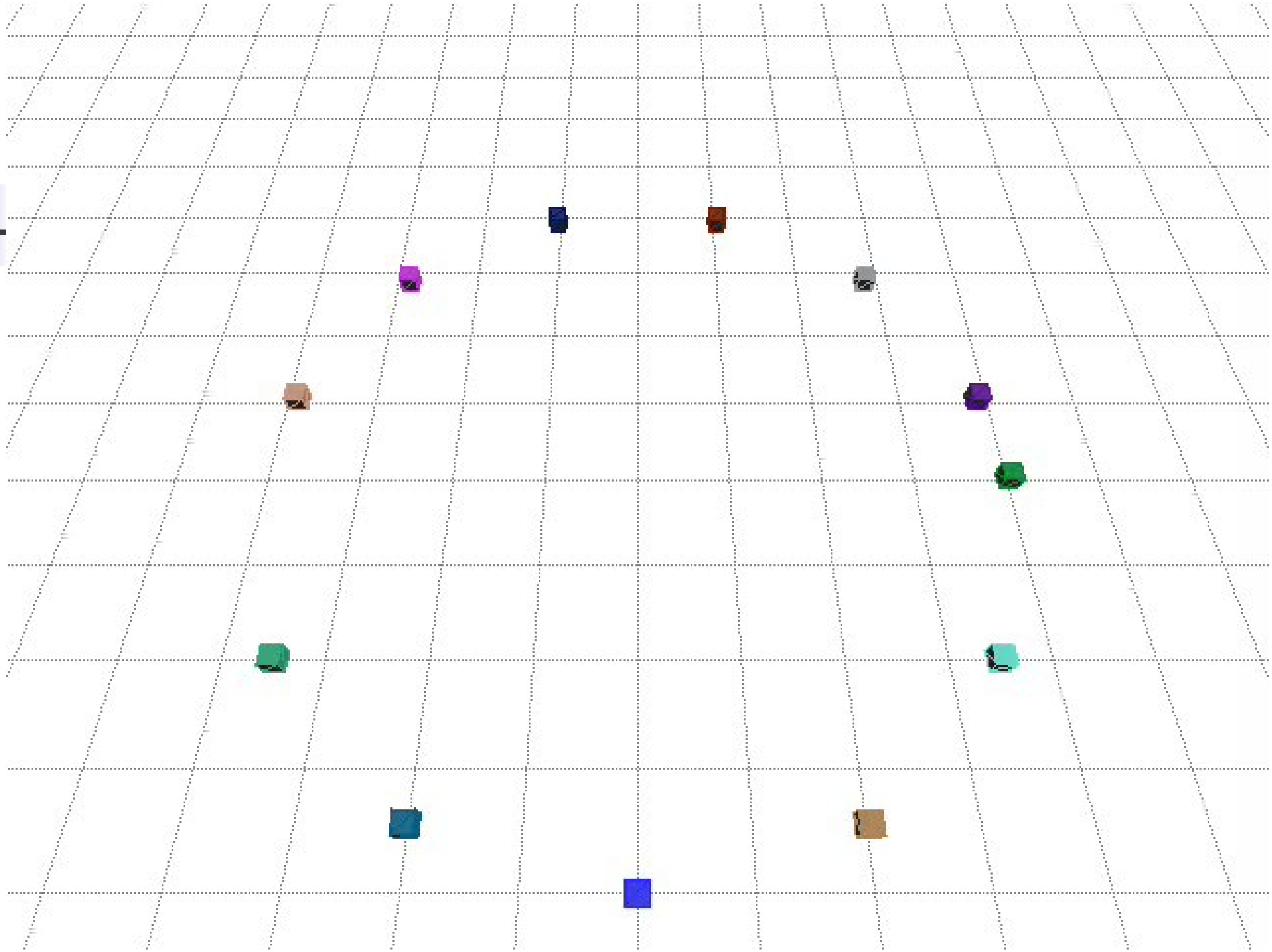
$t_{i(k+1)} = t_{ik} + \tau_{ik}$

$\Delta T_{ik}^{max} \geq \tau_{ik} \geq \Delta T_{ik}^{min}$

$t_{jh} - t_{i(k+1)} + M(1 - \delta_{ijkh}) \geq 0$

$t_{ik} - t_{j(h+1)} + M\delta_{ijkh} \geq 0$

$\delta_{ijkh} \in \{0, 1\}$



Implementation

- C++, PQP (Larsen et al. 2000), CPLEX

Num. of robots	Num. of collision zones	Collision time (secs)	Num. of binary variables	MILP-S time (secs)	Num. of binary variables	MILP-I time (secs)
5	13	18.67	20	0.04	14	0
8	42	55.67	64	0.13	62	0.08
10	71	88.26	102	0.53	100	0.17
12	82	115.81	124	0.61	123	0.25
8 radial, unsymm.	32	54.76	54	0.167	54	0.095
12 radial, unsymm.	94	170.15	128	2.2	128	0.49
8 radial, symm.	29	30.53	54	3.87	54	0.095
12 radial, symm.	86	70.53	128	160	128	60.67

- Running time depends primarily on number of collision zones

Can We Guarantee Optimality?

Gap is guaranteed to be zero in (at least) these cases:

- Each robot can collide with at most one other robot, and both share a single collision zone
- Each path segment is sufficiently long (so robot velocity can go to zero)

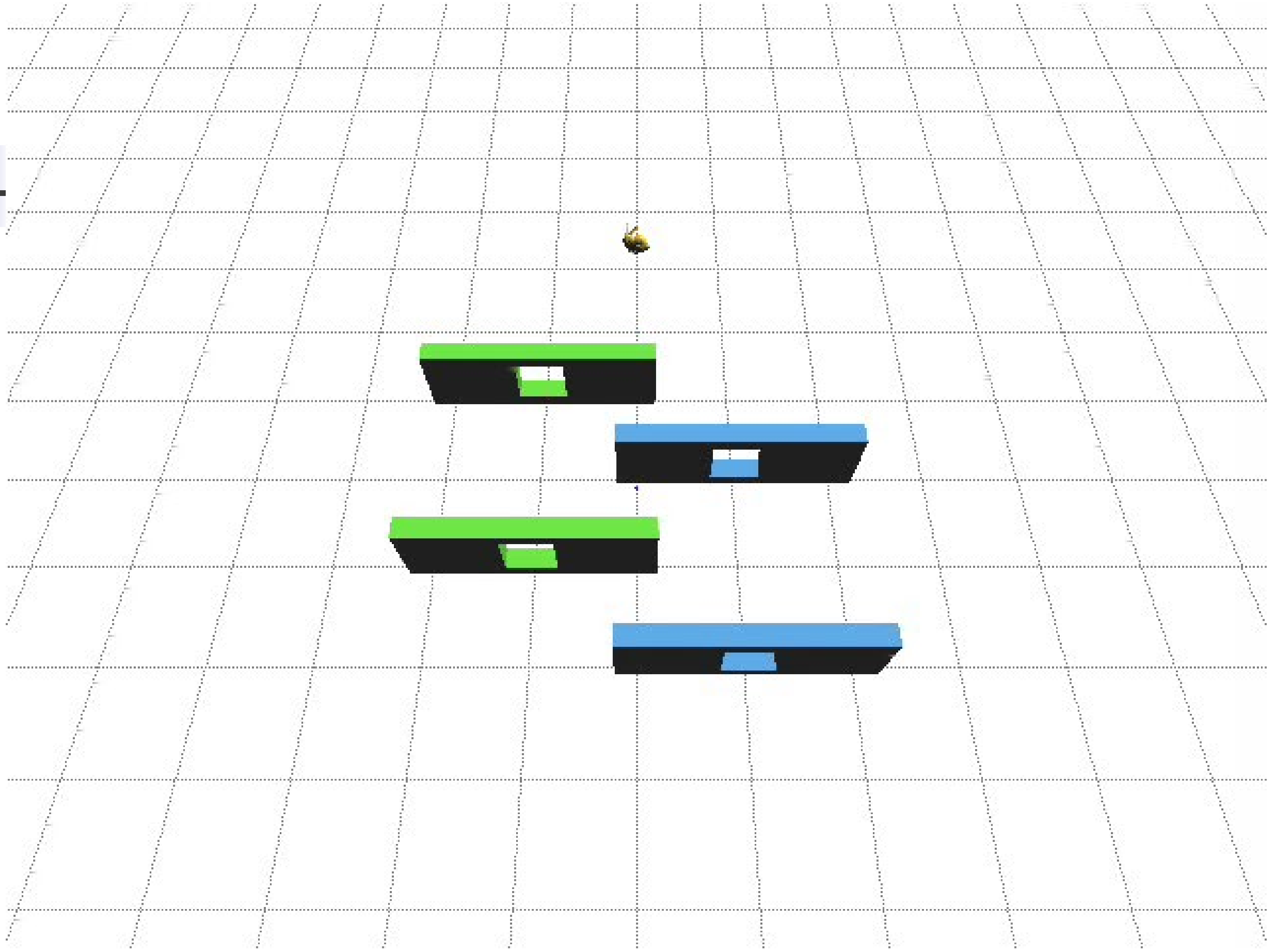
Complexity of Upper and Lower Bound Coordination

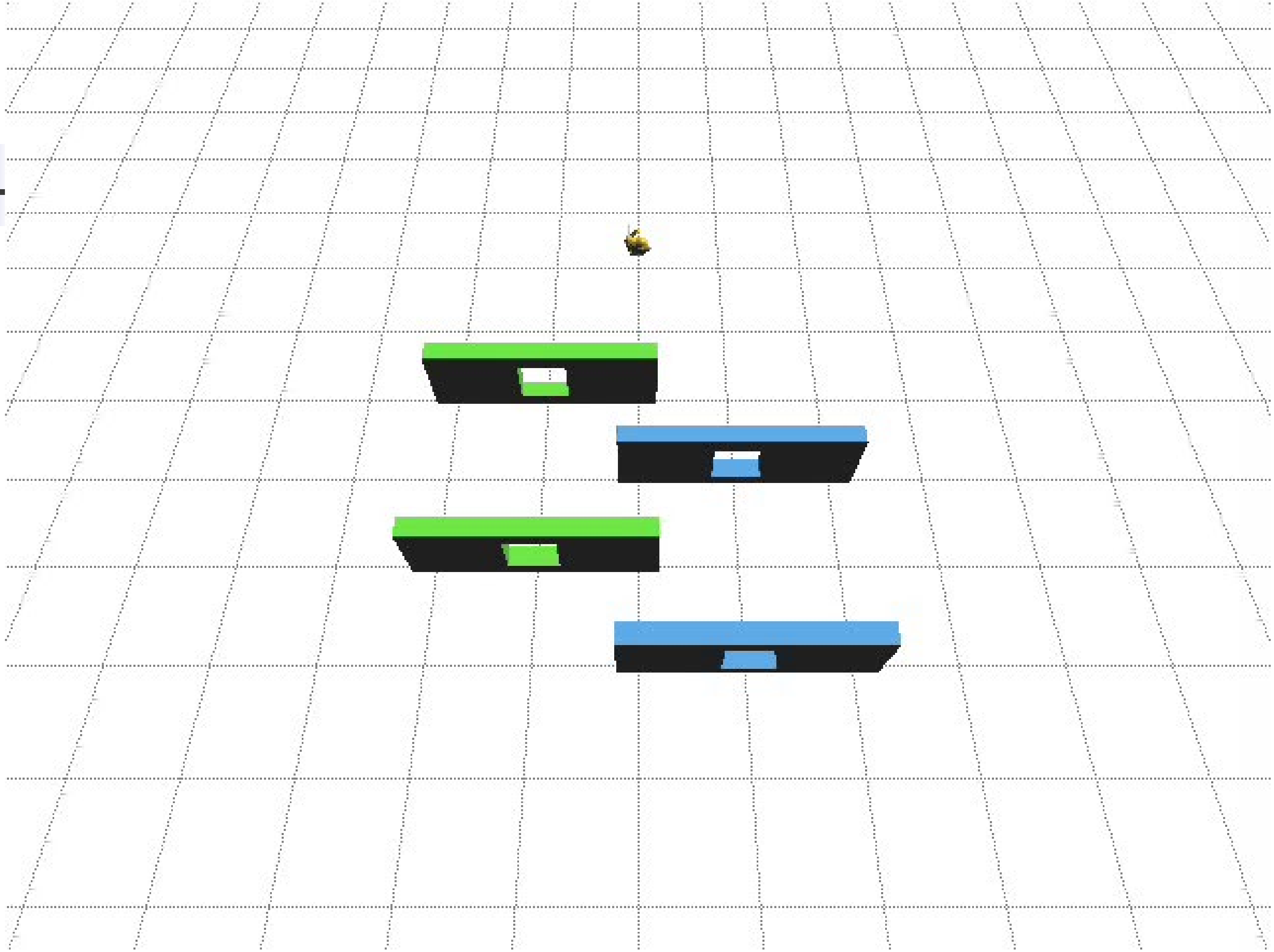
- Upper bound and lower bound coordination problems for multiple robots are NP-hard:

  Reduction from Job Shop Scheduling problem

General Robot Systems

- Moving obstacles
- Car-like robots on continuous curvature paths (Scheuer and Fraichard 1997, 1999; Lamiraux and Laumond 2001)
- Air traffic control
- Manipulators (Bobrow, Dubowsky, and Gibson 1985; Shin and McKay 1985)





Conclusion

- Continuous velocity coordination of double integrator robots formulated as MINLP

  MINLP is difficult to solve
- Approach obtains (near) optimal continuous velocity schedules using bounding MILP formulations
- Complexity depends primarily on number of collision zones (and number of robots)

Acknowledgments

- Animations by Andrew Andkjar

- Support provided in part by:
  Rensselaer Polytechnic Institute
  National Science Foundation