Autonomous Robotics manuscript No.
(will be inserted by the editor)

On the Relationship Between Dynamics and Complexity in Multi-agent Collision Avoidance

Jeffrey Kane Johnson

Received: date / Accepted: date

Abstract This work examines how dynamics and complexity are related in multi-agent collision avoidance. Motivated particularly by work in the field of automated driving, this work considers a variant of the reciprocal n-body collision avoidance problem. In this problem, agents must avoid collision while moving according to individual reward functions in a crowded environment. The main contribution of this work is the result that there is a quantifiable relationship between system dynamics and the requirement for agent coordination, and that this requirement can change the complexity class of the problem dramatically: from P to NEXP or even NEXP^{NP}. A constructive proof is provided that demonstrates the relationship, and potential practical applications are discussed.

Keywords complexity · dynamics · collision avoidance

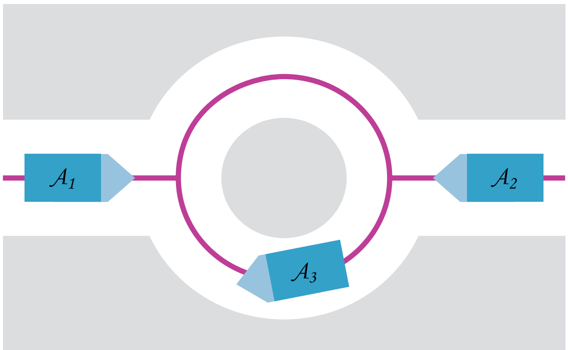


Fig. 1: Agents A_1, A_2, and A_3 attempt to navigate past each other along fixed paths. This work examines how system dynamics affect the need for them to coordinate their actions.

1 Introduction

In industries as varied as mining, agriculture, health care, and automated driving, many practical applications in robotics involve navigating through dynamic environments in the presence of intelligent agents. A large and relatively mature body of literature has been developed that examines various types of these multi-agent systems and the theoretical complexity of planning within them. The focus of this work is specifically how system dynamics interact with problem complexity. For single agent systems, an early result due to Reif and Sharir (1985) showed that adding velocity bounds to one type of motion planning problem can change its complexity from NP-hard to PSPACE-hard. This result indicates that system dynamics can play a role in determining complexity class, however, relatively little attention has been paid to the role that system dynamics play in the complexity of multi-agent problems.

As will be shown, one of the key factors affecting complexity of multi-agent collision avoidance problems is agent coordination. Despite this, handling coordination is often a secondary consideration when formulating solutions to these problems. In real world applications for multi-agent systems, path planning tends to be addressed first with agent coordination being added on. In many cases, this prioritization may cause the problem to be modeled in a way that introduces prohibitive amounts of complexity. Take, for example, the case of an automated vehicle moving along a pre-defined road network. While it is tempting to model the problem primarily as a path planning problem, doing so may put

J. Johnson
School of Informatics, Computing, and Engineering
Indiana University
Bloomington, Indiana 47408

in place requirements that make addressing agent coordination more difficult, specifically, the long planning horizons and precise knowledge of future world states that many path planning techniques require. However, as robotics research has moved further from laboratories and into the real world, these types of coordination problems have become more important. The team behind the planner used in the Bertha Benz drive (Ziegler et al. (2014)), as well as the winning teams of the 2007 DARPA Urban Challenge (Urmson et al. (2007), Montemerlo et al. (2008), and Bacha et al. (2008)) all cite coordination ahead of path planning as an area of future work. The coordination problem also has deep ties to long-standing problems in optimal control theory. Mitter and Sahai (1999) identified the coordination problem as the primary difficulty in designing an optimal controller for Witsenhausen's counterexample (Witsenhausen, 1968). Given the importance of the coordination problem and that practically any real-world system must reason under dynamic constraints, it is important to understand the interplay between coordination, dynamics, and complexity.

One effort to address coordination among road vehicles is explicit, vehicle-to-vehicle (V2V) communication capabilities. V2V could be used for many things, but it is not clear that the availability of the communication channels can be guaranteed to levels needed for safety-critical applications, such as collision avoidance (Harding et al., 2014). But it's also unclear to what degree explicit communication is actually necessary: human drivers navigate successfully with only very limited[1] forms of communication, which implies that the coordination they do is also very limited. This raises the question of whether, and to what extent, coordination is actually required for navigating multi-agent systems, and it implies that a better understanding of that requirement will lead to the development of more practical and robust navigation algorithms.

This work examines agent coordination in a variant of the reciprocal n-body collision avoidance problem described by van den Berg et al. (2009). The key insight is that system dynamics can introduce a requirement for coordination where there otherwise would be none, and a constructive proof is given that allows the existence of this requirement to be determined. The importance of the coordination requirement is that once it exists within a system, the space of appropriate models for the problem changes, which changes the complexity class of any solutions to the problem. This result demonstrates the existence of fundamental ties between system dynamics and problem complexity for multi-agent collision avoidance problems.

This document revises and extends Johnson (2016) by simplifying and unifying definitions, addressing some theoretical weaknesses, providing new extensions and discussion and an appendix with additional results. A selection of relevant background literature is covered and then the main results are derived. Finally, future work and conclusions are discussed.

2 Related Work

This section will detail a selection of the large body of relevant work on collision avoidance, planning, and complexity. Several general theoretical complexity results are described, followed by a survey of notable solution techniques for multi-agent collision avoidance, and, finally, a brief survey of solution techniques for hybrid dynamical systems is given.

In the absence of dynamic constraints and other moving agents, the problem of planning a collision-free path through an environment is typically referred to as the *mover's problem*, which is the problem of moving an articulated polyhedral body through a Euclidean space populated with static polyhedral obstacles. Reif (1979) showed the general problem to be complete for PSPACE and the classical problem, referred to as the *piano mover's problem*, where the moving body is a rigid polyhedron moving in $\mathbb{R}^2$ or $\mathbb{R}^3$, to be in P under the condition that geometric constraints can be expressed algebraically. Work by Halperin and Sharir (1996) further showed near quadratic bounds for the $\mathbb{R}^2$ case. The multi-body variant of the piano mover's problem, known as the *warehouseman's problem*, was shown by Hopcroft et al. (1984) to be PSPACE-hard. Reif and Sharir (1985) additionally showed that introducing agents that follow fixed-trajectories into the piano mover's problem for $\mathbb{R}^3$ changes the complexity class of the problem to NP-hard, and that adding velocity bounds makes the problem PSPACE-hard.

For multiple agents following non-fixed trajectories, the problem is generally formulated in terms of sequential decision making in a discretized space rather than geometric motion in a continuous space. When planning among the agents can be done independently while still achieving a jointly optimal solution, the problem can be formulated as a type of Markov decision process (MDP), which Papadimitriou and Tsitsiklis (1987) showed belongs to complexity class P (and Littman et al. (1995) notes that the MDP complexity results

[1] Indicator lights are a common channel of communication, but they are notoriously unreliable. Horns also provide a form of communication, but are limited by context. Relative positions and speeds can convey intent, but, as channels of communication, these are very low bandwidth.

rely on certain complexity assumptions regarding transition and cost functions). However, as Boutilier (1996) discusses, independent planning cannot in general guarantee a globally optimal plan; only joint planning can guarantee global optimality. While joint planning problems with a central planner that computes motions for all agents can also be formulated as types of MDP's, and therefore belong to P, others, such as the unlabeled variant, where multiple agents must reach multiple goal positions without restrictions for which agent reaches which goal, were shown by Solovey and Halperin (2015) to be PSPACE-hard. For decentralized problems Bernstein et al. (2002) showed that for cooperative agents (i.e., agents that share a reward function) this class of problems is at least complete for NEXP in both the jointly fully observable (DEC-MDP) and jointly partially observable (DEC-POMDP) cases. Goldsmith and Mundhenk (2007) showed that the partially observable stochastic game (POSG), which is the non-cooperative version of this problem (i.e., the problem in which agents do not share a reward function), is complete for $\mathrm{NEXP}^{\mathrm{NP}}$.

In the context of collision avoidance in multi-agent systems, Fiorini and Shiller (1998) introduced the notion of *velocity obstacles* to address the pairwise collision avoidance problem. In this approach the set of velocities resulting in collision between a robot and another moving agent are computed explicitly, and this set is called the velocity obstacle (VO). Collision avoidance is then guaranteed by assigning velocities outside the VO to the agent. Fraichard and Asama (2004) described the more general idea of an *inevitable collision state* (ICS) as a "state for which, no matter what the future trajectory followed by the system is, a collision with an obstacle eventually occurs." Similar state descriptors had been proposed by LaValle and Kuffner (2001). Owing to the inherent computational complexity of the ICS representation, Bekris (2010) examined sampling-based approximation methods. The Optimal Reciprocal Collision Avoidance (ORCA) framework introduced by van den Berg et al. (2009) expanded the ideas of VO and ICS to multi-agent systems without inertial constraints. Later, van den Berg et al. (2011) extended their results to include some consideration for inertially constrained systems. Pairwise collision avoidance for holonomically constrained systems was demonstrated by Wilkie et al. (2009) and extended to general multi-agent systems by Alonso-Mora et al. (2010). When coordination among agents is allowed, Bekris et al. (2012) demonstrated that non-collision can be guaranteed for a broad class of de-centralized motion planning problems. Shoham and Tennenholtz (1995) describe an alternate approach to these types of problems that imposes artificial rules, or *social laws*, on agent coordination in order to remove the need for online coordination altogether. An interesting related theoretical formulation for factored POSGs due to Oliehoek et al. (2012) is the influence-based abstraction (IBA), which allows multi-agent systems with weakly coupled interactions to be decomposed into local models.

The distinction between sequential decision making problems and continuous geometric motion planning problems is typically formulated mathematically as the problem of choosing among a finite number of homotopy channels in some state space (decision making), and generating actuation commands to navigate those channels (motion planning). In practice, most interesting problems have characteristics of both problem types, and therefore are hybrid problems with hybrid solutions. An early approach from Kambhampati et al. (1991) solved these hybrid problems by interleaving graph planning with motion planning in predefined discrete spaces. Later, Kaelbling and Lozano-Pérez (2013) dealt with uncertainty and introduced sophisticated task description languages. When the topography of the state space is not known beforehand, Alterovitz et al. (2007) introduced a sampling-based approach that can be used to construct roadmaps in the state space.

3 Problem Description

As stated in §1, this work examines agent coordination under a variant of the reciprocal n-body collision avoidance problem described by van den Berg et al. (2009). In the formulation used here, the problem is extended slightly to allow general system dynamics, and to make each agent's task to choose an appropriate control command rather than velocity:

Problem 1 Let $\mathbb{A}$ be a set of agents navigating a shared space with a shared inertial reference frame and assume that collision is always avoidable in the initial system state. Assume each agent can fully observe the instantaneous dynamic state of the environment. Assume global constraints on dynamics, efficient methods for dynamics computations,[2] and that agents are under decentralized control. Each agent may assume with certainty that other agents will prefer both to avoid collision and to avoid causing collision, but that otherwise the future actions of other agents are not generally observable. Agents may coordinate, or negotiate, their future actions via communication under the following restrictions:

[2] §5 gives an overview of efficient methods for the various types of dynamics computations Problem 1 entails.

1. Communications are strictly pairwise
2. Agents may only communicate with regard to their own actions (i.e., they may not relay information)
3. There is always some non-zero cost associated with communication

When $|\mathbb{A}| > 2$, how can a given agent choose a control with the guarantee that it will be possible for *all* agents to remain collision free for some time horizon?

One of the difficulties in addressing Problem 1 is that it becomes a partially observable multi-agent system when more than two agents need to communicate in order to coordinate actions. The focus of this work is on how system dynamics affect the model space and complexity of Problem 1 by affecting the need for communication and coordination.[3]

3.1 Notations and Definitions

Assume all agents operate in a shared workspace $\mathcal{W}$, and let $\mathcal{S}$ denote the time-indexed configuration space for all agents. Let Φ denote the set of control trajectories available to an agent A, where for each $\phi \in \Phi$ the state of A at time t from state $\mathbf{x} \in \mathcal{S}$ under control trajectory ϕ is $\mathbf{x}_t = \phi(\mathbf{x}, t)$. Let $A(\mathbf{x})$ denote the region of state space occupied by A at state $\mathbf{x}$. Assume agents may be *interacting* or *non-interacting* as defined below.

Definition 1. An *interacting agent* is one whose dynamic state is a function of both the state of the external environment and an internal policy (for example, pedestrians or animals could be interacting agents).

Definition 2. A *non-interacting agent* is one whose dynamic state is a function of only the state of the external environment (for example, trees or rolling rocks could be non-interacting agents).

Definition 3. The actions of two agents are said to require *coordination* when the feasibility the control sequence that either agent uses is not independent of the other's.

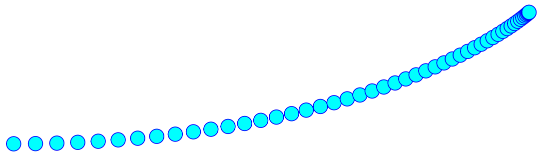
Definition 4. For an agent A navigating an environment that has a set of interacting and non-interacting agents $\mathbb{A}$, an *obstacle* O is a member of the set of obstacles $\mathbb{O}$, which is defined as:

$$\mathbb{O} = \mathbb{A} \setminus A$$

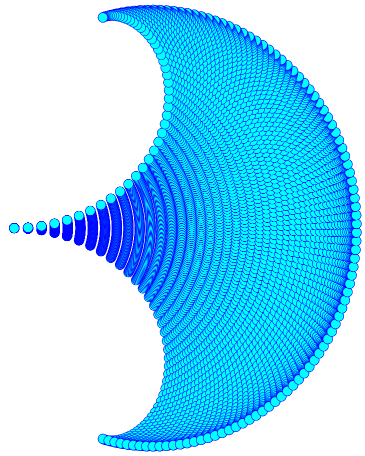
Definition 5. A *state space obstacle* $\mathcal{B}$ is the set of states swept out by an obstacle O as it evolves from an initial state $\mathbf{x}_i$ to an infinite time horizon T under some control trajectory ϕ_i:

$$\mathcal{B} = \bigcup_{t \in T} O(\phi(\mathbf{x}_i, t))$$

[3] The problem of choosing when and with whom to communicate, while also difficult, is not a focus of this paper.



(a) For a given agent state $A(\mathbf{x})$ and path P, the *stopping path* $\mathrm{SP}(A(\mathbf{x}), P)$ is the minimal set of states A must occupy while coming to zero velocity from $\mathbf{x}$ along P. Here disc agent A starts on the left and comes to a stop in the upper right. In the illustration the motion is discretized at fixed time intervals, so the spacing between steps indicates relative speed.



(b) For a given agent state $A(\mathbf{x})$ and complete set of followable paths $\mathbb{P}$, the *stopping region* $\mathrm{SR}(A(\mathbf{x}), \mathbb{P})$ is the union of all SPs over $\mathbb{P}$. This illustration shows the SR for disc agent A from (a). The region is plotted by sampling agent trajectories generated by sweeping steering from hard right to hard left.

Fig. 2: Illustrations of the shapes of an SP (Definition 8) and an SR (Definition 9) for a hypothetical disc agent following constant control trajectories with unicycle dynamics traveling along a 2D plane. The system was initialized with non-zero velocity, bounded deceleration, and bounded yaw rate.

Definition 6. An *inevitable collision state* (ICS) for an agent A is a state from which all feasible future trajectories of A result in collision:

$$\mathbf{x} \text{ is ICS} \leftrightarrow \forall \phi, \exists \mathcal{B}_i, \exists t :: A(\phi(\mathbf{x}, t)) \cap \mathcal{B}_i \neq \emptyset$$

Notations and definitions relating to ICSs are adapted from Fraichard and Asama (2004).

It is important to note that, due to Definition 5, the computation of ICS space requires knowledge of future control trajectories of all obstacles.

Definition 7. A *contingency plan* is a control sequence that an agent can execute that is guaranteed to avoid ICS space.

Definitions 8 & 9 below introduce concepts that will aid in the analysis of Problem 1.

Definition 8. For a state $\mathbf{x}$ and path P, the *stopping path* $SP(A(\mathbf{x}), P)$ is the minimal set of agent states A must occupy while coming to zero velocity from $\mathbf{x}$ along P (Figure 2a).

Definition 9. For a given agent state $A(\mathbf{x})$ let $\mathbb{P}$ be the set of all followable paths and let I be its index set. Define the *stopping region* $SR(A(\mathbf{x}), \mathbb{P})$ as the disjoint union of all SPs over $\mathbb{P}$ (Figure 2b):

$$SR(A(\mathbf{x}), \mathbb{P}) = \bigsqcup_{i \in I} SP(A(\mathbf{x}), P)$$

4 Theory

This section will derive the main results of the work. First the specific conditions are derived for which solutions to Problem 1 can make non-collision guarantees with and without agent coordination. Then an explicit problem formulation for Problem 1 is given and it is shown that the complexity of the problem is directly influenced by system dynamics via the coordination requirement. The section is closed with a discussion of the results.

To aid with the derivation, Assumption 1 makes explicit the assumption in Problem 1 that agents will maintain and invoke contingency plans during navigation:

Assumption 1. *Agents in a multi-agent system will compute and maintain motion plans independently of interaction effects with other agents to use as collision avoidance maneuvers.*

The following conjecture posits that the stopping path is the unique type of contingency plan satisfying Assumption 1 under Problem 1.

Conjecture 1 Stopping paths are the unique category of motion plan that can enable coordination-free contingency plans under Problem 1.

Conjecture 1 is taken as a reasonable assumption because stopping paths are an obvious choice for contingency plans: they do not rely on coordination, and their execution by a set of agents leads quickly to a stasis in which collision avoidance can guaranteed indefinitely. It is an interesting, and open, question whether and how Conjecture 1 could be proven for a multi-agent system, and it is a point of future work.

4.1 Relating Dynamics to Coordination

This section derives the relationship between system dynamics and the requirement for agent coordination under Problem 1.

Lemma 1 *A necessary condition to guarantee that a system be able to remain collision free is that a zero false negative membership test for ICS space must be computable from any state.*

Proof. This follows from Definition 6. In order to guarantee that a system can remain collision free, it must only move into states that are *not* in ICS space. In order to identify such states, a membership test for ICS space must be computable, and a negative result from the test must imply non-membership. A positive test, however, need not imply membership; such false positives result in smaller regions of space identified as non-ICS space, but do not necessarily preclude collision avoidance. □

Lemma 2 *Under Problem 1, zero false negative ICS space membership is not computable without coordination among agents.*

Proof. This follows from Definitions 5 & 6: in order to compute non-membership ICS space the future control trajectories of *all* agents must be fully observable. However, under Problem 1, the future control trajectories are not fully observable without coordination among the agents. □

Theorem 1 *Under Problem 1, coordination is required in general for each agent to maintain the ability to remain collision free.*

Proof. It follows directly from Definition 7 that all agents must have a contingency plan in order to guarantee the system can remain collision free. By Lemma 1 and Definition 7, computing a contingency plan requires computing membership in ICS space. By Lemma 2, computing the required ICS space membership requires coordination among agents. □

The SR concept from Definition 9 will now be used to frame the coordination requirement in terms of system dynamics.

Lemma 3 *Consider a system with a set of agents* $\mathbb{A} = \{A_1(\mathbf{x}_1), \ldots, A_n(\mathbf{x}_n)\}$*, a set of followable path sets* $\mathcal{P} = \{\mathbb{P}_1, \ldots, \mathbb{P}_n\}$*, and an index set* $I = \{1, \ldots, n\}$*. For each* $j \in I$*, the set* $\mathbb{P}_j$ *contains the followable paths* $\{P_1, \ldots, P_m\}$ *for agent* A_j*. For* $i, j \in I$ *define the union of obstacle SRs as:*

$$\mathcal{SR}_j(A(\mathbf{x})_j, \mathbb{P}_j) = \bigcup_{i \neq j} SR(A_i(\mathbf{x}_i), \mathbb{P}_i)$$

For each j*, also define an intersection set* $\mathcal{Q}_j$*:*

$$SR(A_j(\mathbf{x}_j), \mathbb{P}_j) \cap \mathcal{SR}_j(A_j(\mathbf{x}_j), \mathbb{P}_j) = \mathcal{Q}_j$$

If and only if for all j *there exists an SP such that* $SP(A_j(\mathbf{x}_j), P_j) \cap \mathcal{Q}_j = \emptyset$*, then it can be guaranteed without coordination that no* $\mathbf{x} \in \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ *is an ICS.*

Proof. First, testing SP disjointness with respect to the *union* of obstacle SRs is valid by Definitions 8 & 9 because SPs and SRs are independent of the states of other agents. Second, the statement that for all agent indices j a stopping path exists that is disjoint from $\mathcal{Q}_j$ implies that all agents have disjoint stopping paths available. By Definition 8 this means that all n agents can come to a stop without intersecting. Therefore, *if* the condition in the theorem holds, collision is not inevitable in any state $\mathbf{x}_1, \ldots, \mathbf{x}_n$. This is also true *only if* the condition holds because if $SP(A_j(\mathbf{x}_j), P_j) \cap \mathcal{Q}_j \neq \emptyset$, then under assumption[4] of Conjecture 1, agents would otherwise all need to find a *unique* set of SPs such that collision is avoided if all agents execute exactly that set of contingency plans. But because remaining collision free would now rely on each agent executing exactly one control sequence whose feasibility directly depends on every other agent executing exactly one control sequence, the system now requires coordination by Definition 3, and so is no longer coordination free. □

Theorem 2 establishes a condition under which any general dynamic system is guaranteed to be able to remain collision free without coordination:

Theorem 2 *A multi-agent system is guaranteed to be able to remain collision free without coordination if and only if for all* $O \in \mathbb{O}$ *relative to each* A *there exists an* $SP(A(\mathbf{x}), P)$ *such that* $SP(A(\mathbf{x}), P) \cap SR(O, \mathbb{P}) = \emptyset$*.*

Proof. This follows directly from Lemma 3. By Definition 6 a system is capable of remaining collision free if and only if it is not in an ICS. By Lemma 3, this is true if and only if all agents have stopping paths disjoint from the stopping regions of all other agents. □

[4] Breaking this assumption weakens the logical connection in Lemma 3 from a biconditional (*if and only if*) to a material condition (*if*).

The intuition behind Theorem 2 is closely tied to the notion of ICS space, and it would be equivalent to state that under certain conditions, it is impossible to test for membership in *any portion* of ICS space without knowing the future actions of other agents. This idea is related to the sufficient safety condition for partial motion plans derived by Petti and Fraichard (2005), which states that if the final state of a collision-free trajectory is not an ICS, then no state along the trajectory is an ICS state.

The following definition is made for convenience:

Definition 10. The condition that satisfies Theorem 2, that all agents have at least one disjoint stopping path, is called *SP disjointness*.

Theorem 2 states that coordination is unnecessary under SP disjointness. But how, in practice, could agents maintain that property without coordination? Trivially, if an agent modulates its dynamics such that it can always come to a stop without *possibly* intersecting the path of any other agent, the property is satisfied. Agents need no knowledge of the plans of other agents for this; they simply need knowledge of the dynamics of the system. This is the approach taken, for example, by Mazer et al. (1998) in the Ariadne's Clew algorithm. For most inertially constrained systems, however, this behavior would be too conservative to be useful. Worse, it's possible to specify initial conditions in an inertially constrained system such that it is not possible to satisfy the property required by Theorem 2 (see §4.5). This is why, for example, the algorithm for multi-agent collision avoidance for inertially constrained systems given by Bekris et al. (2012) requires coordination in order to maintain its guarantees.

But if Bekris et al. (2012) requires coordination, why is it that the ORCA framework of van den Berg et al. (2009) does not? ORCA is an efficient collision avoidance algorithm based on the VO representation that guarantees non-collision for very complex scenes without the need for explicit agent coordination. As it turns out, Theorem 2 allows the requirement for agent coordination for certain systems to always be dropped. Theorem 3 will establish this possibility:

Lemma 4 *In a system without inertial constraints, it holds that* $SR(A(\mathbf{x}), P) = A(\mathbf{x})$*.*

Proof. That A can instantaneously stop means that the minimal set of states A must occupy while coming to a stop along *any* path P is exactly $A(\mathbf{x})$. This implies further that $A(\mathbf{x}) = \bigsqcup_{P \in \mathbb{P}} SP(A(\mathbf{x}), P)$ which is equal to $\mathrm{SR}(A(\mathbf{x}), \mathbb{P})$ by Definition 9. □

Theorem 3 *A multi-agent system without inertial constraints that is not currently in collision is guaranteed to be able to remain collision free without coordination.*

Proof. Lemma 4 implies that for such systems, SP disjointness holds for all non-collision states. By Theorem 2, such a system is guaranteed to be able to remain collision free without coordination. □

Note that any system described by the VO formulation necessarily lacks inertial constraints and therefore Theorem 3 applies to it. This means the fact that Bekris et al. (2012) required coordination for their solution and van den Berg et al. (2009) did not is directly a result of the system dynamics they employed: the former described systems with inertial constraints, and the latter systems without.

At a deeper level, these results speak to the fundamental problems encountered when ignoring inertial constraints in dynamical systems. The VO representation, for instance, is an inertially unconstrained approximation often used for mutual collision avoidance in multi-agent systems because of its simplicity and elegance: it only needs information about the instantaneous dynamic state of a system to maintain its guarantees. However Theorem 3 suggests, and Lemma A.3 shows, that the VO approach alone is not always sufficient to guarantee non-collision. This is demonstrated empirically by Wilkerson et al. (2014) who showed that using the VO representation in a system with inertial constraints can result in collisions, even though the algorithms guarantee non-collision. The Appendix provides a simple proof of thisf.

4.2 Collision Avoidance as a Decision Problem

This section defines an instance of Problem 1. The problem formulation is constructed such that it might reasonably map to real-world problems but it is only intended in this work to aid in the derivation of the main result.

In any practical instance, Problem 1 would be a hybrid decision making/motion planning problem, so its model would also take a form similar to the hybrid models mentioned in §2. Let $\mathcal{R}$ be some sufficiently dense roadmap approximation to $\mathcal{S}$ (for instance, a Stochastic Motion Roadmap Alterovitz et al. (2007)), where "sufficiently dense" means dense enough to allow solutions to be found. Assume each agent is initialized at some vertex of $\mathcal{R}$, and assume all agents plan at a uniform and aligned frequency. Assume all agents have full knowledge of system dynamics and of $\mathcal{R}$, and that some efficient method for computing SRs exists (see §5.1). Assume that agents are capable of coordinating their actions if they so choose (subject to the restrictions outlined in Problem 1) in a way similar to that presented in Bekris et al. (2012), in which agents negotiate joint-contingency plans.

Define G as an instance of Problem 1 in its most general as a POSG:

Problem 2 Let $G = (\mathbb{A}, O, \mathcal{C}, c_0, A, T, \Omega, R)$, where:

- $\mathbb{A}$ is a set of agents whose states include intent, which defines how the agent's internal policy affects its actuation
- O is a set of observations (mapping of observable agent states to vertices of $\mathcal{R}$)
- $\mathcal{C}$ is a set of configurations of the system (mapping of full agent states to vertices of $\mathcal{R}$)
- c_0 is a designated initial configuration
- A is a set of actions that enable transition between any two vertices on $\mathcal{R}$
- $T : \mathcal{C} \times A^k \times \mathcal{C} \to [0, 1]$ is the transition probability function, where $T(c, a_1, \ldots, a_k, c')$ is the probability that configuration c' is reached from configuration c when each agent i chooses an action a_i
- $\Omega : \mathcal{C} \times I \to O$ is the observation function, where $\Omega(c, i)$ is the observation made in configuration c by agent i. The observation of one other agent may include the result of a negotiation (a joint-contingency plan); for all others the observation includes a distribution over contingency plans
- $R : \mathcal{C} \times A^k \times I \to \mathfrak{R}$ is a reward function, where $R(c, a_1, \ldots, a_k, i)$ is the reward gained by agent i in configuration c when the agents take actions $a_1, \ldots, a_k$

4.3 Main Result

This section derives main result of the work: that the complexity of solving an instance of Problem 1 can be modulated directly via the coordination requirement, which can exist solely dependent on the dynamics of the system. The intuition is that the complexity of the problem lies in the partial observability of agent intents, but if agents are capable of remaining collision free regardless of the intents of others, then the partially observed agent intents can be ignored, making the problem deterministic. The ability to toggle the observability of the problem in this way is due to the agents being independently capable of examining system dynamics to maintain SP disjointness, which removes the coordination requirement from the system.

The following lemma will aid the argument:

Lemma 5 *For a time horizon T, an agent A can assume arbitrary policies for all $O \in \mathbb{O}$ and maintain the non-collision guarantee provided the assumed policies are guaranteed to maintain SP disjointess through T.*

Proof. This follows from Theorem 2. To guarantee the ability to remain collision free, the action an agent takes is irrelevant so long as SP disjointness is maintained. Therefore, if, for every reachable agent state $A(\mathbf{x})$ for $t \in T$, at least one disjoint SP is guaranteed to exist, then the non-collision guarantee holds. □

To further aid in the results of this section, a policy template will be defined that encodes Assumption 1 and Conjecture 1. To take advantage of Lemma 5, the policy template will define behaviors over a time horizon T.

Definition 11. An *SP Disjoint Policy* is a policy that only maps states to actions that maintain SP disjointness for some time horizon T.

Note that the notion of an SP disjoint policy is closely related to the ideas exploited by the ORCA framework.

The following derivations provide the main result.

Lemma 6 *Under SP disjointness and shared time horizon T, G can be modeled as an MDP.*

Proof. Lemma 5 states that the ability to remain collision free over T can be assured whenever an agent A assigns arbitrary polices to other agents so long as SP disjointness is maintained. This is key, because it means A does not need to *observe* anything about the other agents beyond what is already fully observable in order to plan and execute motions in the presence of those other agents. A is free to assume an SP disjoint policy for all other agents, which makes the problem fully observable. This allows Ω to be redefined as a deterministic, one-to-one mapping $C \times I \mapsto C$. Another consequence of assuming an SP disjoint policy is that R can be replaced with a globally shared reward function $C \times A^k \mapsto \mathfrak{R}$. These function redefinitions encode that it is only important that in the current state the agents can rely on each other not to violate SP disjointness, and, as discussed previously, this can be done strictly with knowledge of system dynamics. Under the assumption of full observability, A can then incorporate the state of other agents into its own transition function, as in Boutilier (1996), which effectively centralizes the decision process. Thus, G is now equivalent to a fully observable, centralized, single-agent system. In other words, G is an MDP. □

Lemma 7 *With shared time horizon T, but without SP disjointness, G remains a POSG.*

Proof. Since SP disjointness does not exist, the assurance provided by Lemma 5 that the system can remain collision free over T does not exist. Therefore, it must be secured another way. In order to compute a solution satisfying Problem 1, agents must reason in some way about the future actions of other agents, but Theorem 2 says that in the absence of SP disjointness, coordination among agents is required to do so, which means centralized control cannot be assumed. In the worst case, the agent SRs may intersect in a way that requires more than two agents to coordinate. Communication limitations, however, necessarily induce partial observability of the intent of at least one of the agents. Thus, Ω remains a probability distribution. In addition to a partially observable world, reasoning about future actions involving other agents also requires consideration of non-shared reward functions because they are what determine the distribution over future actions. Thus, R remains a non-shared reward function. Under these conditions, the decision process is decentralized, multi-agent, and partially observable with non-shared reward functions. By definition this is a POSG. □

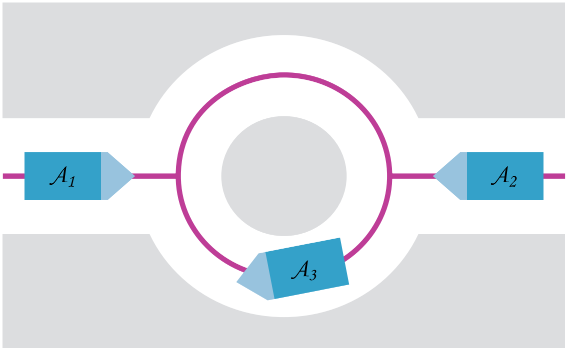
Theorem 4 *There exist instances of Problem 1 in which the existence, or lack, of the coordination requirement alone changes the complexity class of the solution.*

Proof. Consider that Problem 2 is an instance of Problem 1. Lemmas 6 & 7 demostrate that the problem model can be changed depending on the existence of SP disjointness. By Lemma 3 existence, or lack, of SP disjointness is equivalent to the existence, or lack, of the coordination requirement. And, as discussed in §2, the different problem models used by Lemmas 6 & 7 belong to quantifiably different complexity classes. □

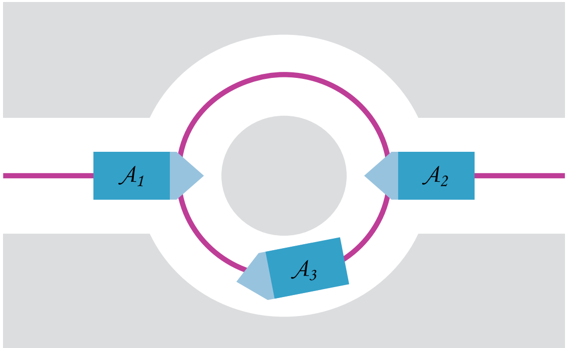
4.4 Discussion

It must first be emphasized that the assumption that agents follow an SP disjoint policy template means that Theorem 4 applies only to systems that can be modeled as guided collision avoidance problems. It does not generally apply, for instance, to systems where agents *must* find plans that are optimal with respect to non-collision criteria. But for applicable systems, the assumption in Problem 1 that efficient methods for dynamics computations exist allows these systems to move between complexity classes without compromising their solutions. As discussed, the differences in complexity can be staggering, with MDPs belonging to complexity class P, and POSGs to NEXP in the cooperative case or NEXP^{NP} in the non-cooperative case.

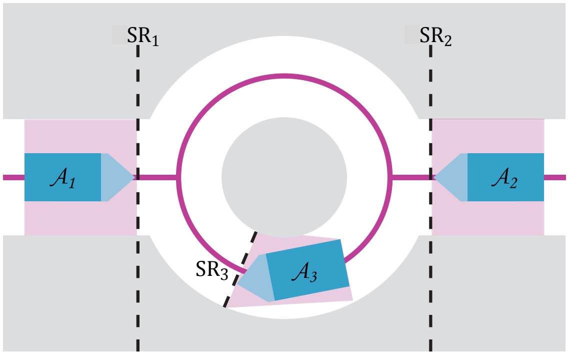
The results above demonstrate that the complexity of the system can be manipulated to keep it within a tractable realm simply by controlling the dynamics. This is both surprising and powerful, and may provide insight into how humans are capable of efficiently



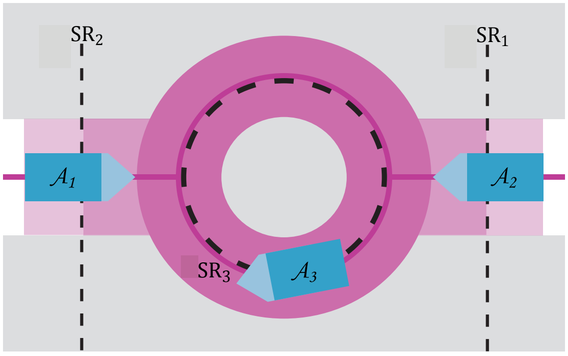
(a) A simple multi-agent system: agents A_1, A_2, and A_3 are moving along shared paths and must navigate around each other.



(b) At the branching point in the path, A_1 and A_2 must make a decision about which branch to follow.



(c) In a system without inertial constraints, the SR's are invariant to the state of the system, and only extend longitudinally to the extents of the agents. The pink regions illustrate the extent of the SR along the path being followed. In such a scenario, the SP disjointness property always holds for any initial state.



(d) An inertially constrained system with a high initial velocity results in an initial system state without a guarantee of SP disjointness. SR_1 and SR_2 extend beyond the median, and SR_3 circles the median. In such a scenario, there is significant overlap in the SR's for each agent, indicated by the darker shaded regions.

Fig. 3: Exemplar problem. SRs visualized as exaggerated pink regions with dashed lines indicating extents.

and successfully navigating complex, multi-agent systems. In the case of roadways, for instance, the environment constrains the set of motions to such an extent that virtually any forward motion ensures progress toward the goal, so geometric optimality of the plan is of limited value. Instead, if agents prioritize maintaining SP disjointness in the system, then safe, goal-directed navigation can be achieved by choosing controls from within the set of controls that maintain SP disjointness. Thus, a planning problem that is, in principle, wildly intractable becomes comfortably tractable.

To summarize, it has been shown that system dynamics alone can be responsible for moving a problem between two types of problem models. This result demonstrates that the dynamics of a system can fundamentally change both the complexity class and model space of the problem.

4.5 Exemplar Problem

This section presents a simple multi-agent system that can be manipulated in certain ways to demonstrate the ideas of this work. The problem is as follows: three agents A_1, A_2, and A_3 are traversing a fixed path that splits around a single median (Figure 3a). All agents can only move along the path or one of its branches. A_1 and A_2 desire to make it past the median, and A_3 desires to stay around the median. As A_1 and A_2 traverse the path, they reach a point where they must make a decision about how to proceed (Figure 3b). Note that because the problem is collision avoidance, it is not a criterion for success that the agents can make it past each other successfully. Success only requires that they remain collision free, so, for instance, a deadlock situation satisfies the requirements, even if it is not the most desirable outcome.

Suppose the agents occupy a system without inertial constraints. In this case, their SRs are disjoint unless or until they actually collide. Theorem 3 guarantees that, for any initial velocity, they can all proceed without coordination while maintaining the guarantee that collision is not inevitable (Figure 3c). On the other hand, assume the agents occupy an inertially constrained system. For a sufficiently high initial velocity, none of the three agents have an SP that can be guaranteed to be disjoint of all other SRs (Figure 3d). It should be clear both by inspection and by Theorem 2 that maintaining any non-collision guarantee is *only* possible in this case if they somehow coordinate.

4.6 Incorporating Environmental Information into Dynamics Computation

So far, agents have been assumed to have no knowledge of how others move aside from their dynamic capabilities, but this could be generalized to give all agents access to a set of pre-defined rules or *social laws* as described by Shoham and Tennenholtz (1995). For instance, right-of-way rules could be defined that allow the need for coordination to be removed from more complex interactions because the rules guarantee the existence of contingency plans, which enable SP disjointness to hold. In fact, the SP disjointness condition itself is essentially a pre-defined rule, but one that is derivable strictly from the dynamics of the system. But if all agents have access to a shared set of rules and the environment is marked in a way that unambiguously indicates which rules to follow at any given time, then it is straightforward to incorporate this information into the computation of the stopping regions.

Let Γ be a set of rules where each rule $r \in \Gamma$ is a set of parameterized dynamic constraints. A rule r is "applied" to an agent A if the constraints of r are applied to the motion model of A. Assume all agents have full knowledge of Γ. Let $M_\Gamma : (A_i, O) \mapsto \gamma_i$ be a mapping of an agent/observation pair to a set of active rules $\gamma \subseteq \Gamma$. In other words, M_Γ is a function that indicates for a given observation which rules are active for a given agent.

It is now straightforward to incorporate environmental rules into SR computation: there is simply an additional step during control computations that modifies the dynamic models of observed agents according to the active rules given by M_Γ. These modified models now affect how SRs are computed, which provides an encoding for the effects of the rules into the problem.

5 Complexity of Dynamics Computations

The results of this work require the existence of efficient methods for the dynamics computations used to compute SRs, test for collision, and produce motion controls. For the generation of motion controls §2 gave results showing P-time complexity for the $\mathbb{R}^2$ or $\mathbb{R}^3$ piano mover's problem. In cases where online feedback control suffices, rather than full motion planning, there exist many linear- or constant-time control laws that can be used (Paden et al., 2016).

The remainder of this section provides a brief survey of techniques that could be used during SR computation and collision detection.

5.1 SR Computation

The problem of SR computation is a specific instance of the more general problem of computing reachable regions of state space. Ó'Dúnlaing (1987) gave a P-time algorithm for point agents and obstacles moving in one dimension under inertial constraints. Johnson and Hauser (2012) generalized the result to convex agents and obstacles moving along fixed paths in $\mathbb{R}^2$ while maintaining P-time complexity.

In higher-dimensional applications or with articulated agents, the problem of computing reachable regions becomes intractable, so approximation techniques can be employed. Valtazanos and Ramamoorthy (2011) employed pre-computed reachable region templates that are composed online in order to plan efficiently. Allen et al. (2014) similarly employed machine learning to efficiently approximate reachable regions online.

Geometrically, computing SRs is the problem of computing swept volumes (Baek et al., 1999; Abrams and Allen, 2000). In general, computation of swept volumes to arbitrary precision cannot be done efficiently, but many techniques have been developed to allow efficient and practical approximations (Kim et al., 2004; Täubig et al., 2011; von Dziegielewski et al., 2010).

In any practical application the method chosen for computing SRs will necessarily be instance-specific, but a variety of tools exists for performing these types of computations efficiently.

5.2 The Collision Detection Problem

Many planning and motion control techniques require explicit collision detection. Sampling-based techniques in particular, which are commonly used in practice, tend to spend a majority of their computational budget on collision detection (LaValle, 2006).

Because of its natural application in video gaming, graphics, and simulation, there is a significant literature on collision detection techniques and theory (Weller, 2013; Jiménez et al., 2001; Kamat, 1993). In the context of agent navigation, collisions are usually tested between 2D or 3D convex polygons using tools from computational geometry, many of which stem from work by Shamos (1978).

For the 2D case, linear- and logarithmic-time intersection test algorithms have been known for some time (Chazelle and Dobkin, 1980). In order to achieve those low theoretical complexities, however, a significant amount of bookkeeping is necessary, often to the extent that the bookkeeping dominates the running time. Simpler algorithms based on the hyperplane separation theorem (Eberly, 2008) are typically much faster in practice despite P-time complexity.

Often it is also desirable to have a measure of minimum separating distance in addition to just an intersection test. In the 2D case, again, this can be accomplished with linear time complexity (Gilbert et al., 1988). But these algorithms are necessarily more complex than just intersection tests, so they will often be much slower, which is especially a burden when dealing with 3D polygons. In many situations, collisions are tested *over time* which means information computed during one time step may be used to inform computations during the next. Closest feature tracking techniques, pioneered by Lin and Canny (1991), exploit exactly this temporal coherence in order to achieve expected constant time performance.

When the geometric models are static or pre-defined, bounding-volume hierarchies (Ericson, 2005) and spatial decompositions (Hornung et al., 2013) can enable extremely fast collision detection and distance approximations between objects of high geometric complexity.

Depending on the motion control method, the choice of collision detection method may play an important part in maintaining real-time capability for a mobile agent. Thankfully, the field of collision detection is well studied and understood, and there exist many established and efficient techniques.

The next section will discuss future work and possible applications for these results.

6 Future Work & Applications

One of the motivating problems for this work is that of vehicles navigating roadways. Current solutions to this problem often suffer from tractability problems while trying to consider inter-agent interactions. To alleviate this, the problem could be reformulated to minimize SR intersections by treating the minimization as a minimum constraint removal problem (Hauser (2012), Erickson and LaValle (2013)). Such a formulation would allow fine-grained control over the degree of coordination necessary and with whom it must be done.

Another very practical avenue of future work is loosening the assumption that all agents always behave in a self-preserving way. The results of this paper may be useful in helping to define categorization routines that identify the likelihood that agents are obeying system assumptions and to alter system behavior to account for it. This type of categorization and reaction happens frequently in real systems, particularly among human drivers.[5] Results from Schoettle and Sivak (2015) indicate that the inability to identify and adjust for aberrant behavior has contributed to accidents involving autonomous vehicles.

Related to the above would be loosening the assumption that the dynamic state of the system is fully observable. In practice, one of the big causes of partial dynamic observability is occlusion; however, assuming an agent can detect it, an interesting approach to dealing with this would be to treat the occlusion itself as an agent and assign to it SRs that, by some measure, reasonably represent the SR that any actual agent or agents emerging from that occlusion might have. This obviates the need to maintain any hypothesis about what the state of the occluded portion of the world might be while providing sufficient information to maintain SP disjointness in the system, and thereby ensure collision avoidance.

The applicability of the results could be expanded further by extending the stopping region concept to include *holding patterns*, which are control trajectories of agents that follow some fixed pattern. Coordination requirements could then potentially be derived for systems consisting of airplane-like agents.

Problem 1 also does not explicitly specify a discrete or continuous time system. In practice, most systems are discrete time. It should be straightforward to amend these results to deal with them explicitly, and the incorporation of system timing will be vital for any practical application. Bekris et al. (2012) dealt extensively with the general problem of discrete time steps in multi-agent systems and would serve as an excellent resource.

As noted in §4, the results of this paper assume that Conjecture 1 holds for a given system. Future work will examine both whether and to what extent the conjecture can be demonstrated mathematically, and the generalization of these results for system in which the conjecture cannot reasonably be assumed.

[5] There is even a special tool, the horn, for alerting those around us that someone's behavior is aberrant.

6.1 Stopping Regions with Reference Velocities

A potentially useful extension to the SR model is to incorporate an explicit notion of reference velocity, which is related to the notion of an optimization velocity from van den Berg et al. (2009). This can be used to incorporate environmental influence into SR computation, or, by introducing an artificial reference velocity, it can simplify SR computation in systems with high absolute and low relative velocities due to the fact that stopping regions defined in terms of relative velocities can be much smaller than their absolute counterparts. These extended stopping regions require only minor refinement in the existing definitions.

To introduce the reference velocity term, the reference frame used in Problem 1 is extended to include a velocity:

Definition 12. A *reference velocity* is a velocity component of the reference frame with respect to which agent velocities are measured for SP and SR computation.

Rather than computing a stopping velocity with respect to zero velocity in the reference frame, agents are allowed to choose any reference velocity. Naturally, this induces a dependence of the coordination requirement on the reference velocity. Lemma 8 extends Theorem 2 to account for this:

Lemma 8 *A multi-agent system is guaranteed to be able to remain collision free without coordination if there exists a reference velocity such that Theorem 2 holds.*

Proof. Suppose there exists a reference velocity $\mathcal{V}$ such that Theorem 2 holds. Let $\mathcal{V}$ be the reference velocity component of reference frame. Theorem 2 can now be applied. □

Stopping regions with a reference velocity can now be defined:

Definition 13. For a given agent state $A(\mathbf{x})$, reference velocity v_r, and set of followable paths P, the *stopping region*, denoted $\mathrm{SR}(A(\mathbf{x}), v_r, P)$, is the SR computed for the reference velocity v_r (Figure 4).

The stopping path can defined similarly. Note that there's no fundamental incompatibility between the results derived in this work and this extended notion of stopping regions; the reference velocity chosen for any problem is arbitrary to begin with. Formally including it in the definition simply acknowledges that fact.

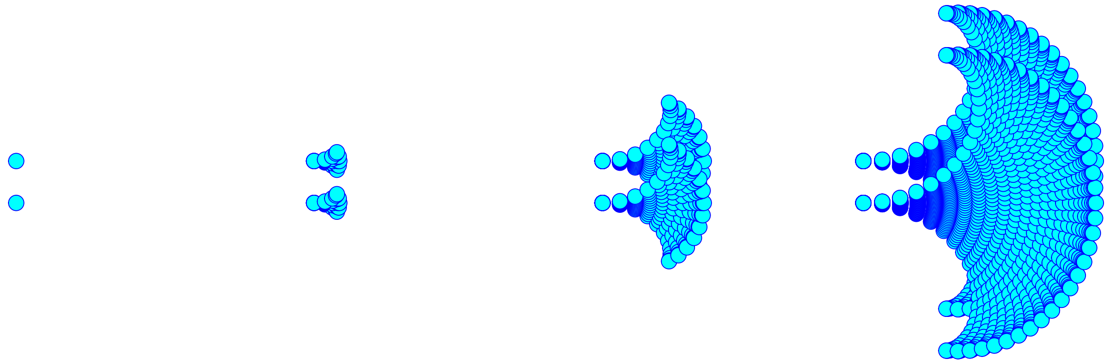
The immediately obvious application of a reference velocity is to incorporate it as some sort of global parameter. This, however, would require some form of coordination between agents. A more interesting method for using it would be to assume agents will search independently for reference velocities that satisfy SP disjointness. The existence of such a reference velocity can be computed independently of agent intents, so the assumption would not immediately induce coordination, but future work needs to explore whether, or under what conditions, different agents choosing different reference velocities may or may not violate any required properties.

6.2 Generalizing to Soft Stopping Regions

For systems with low absolute and low relative velocities (such as pedestrian navigation), low-energy collisions may be permissible, or even unavoidable, and the stopping' region concept could be extended to *soft stopping* regions. A soft stopping region is a region in which an agent will either come to a stop or enter into a low-severity collision, where severity is measured by *collision-induced velocity change* ΔV (Jansson, 2005) and compared to some threshold. The definitions and lemmas below establish this concept.

Definition 14. For a given agent state $A(\mathbf{x})$, collision velocity v_c, and set of followable paths $\mathbb{P}$, the *soft stopping region* $\mathrm{SR}(A(\mathbf{x}), v_c, \mathbb{P})$, is the SR computed for the target velocity of the stopping paths set to v_c.

Clearly Definition 14 is only a minor change from Definitions 9 & 13: the target velocity of the stopping path is now just a parameter. While this may seem a trivial change, the soft stopping region is actually specifying something fundamentally different than a stopping region because v_c depends on the *policy* of the other agent(s) involved in potential collision. Thus, the results of this work do not straightforwardly extend to soft stopping regions because of the effect v_c has on determining SP disjointness: if it is impossible to estimate v_c for other agents, then it is also impossible to estimate SP disjointness, and the absence of the coordination requirement can no longer be guaranteed. In order to use soft stopping regions effectively, then, it is necessary that agents have some mechanism to measure, or estimate from observation (such as described in §4.6), acceptable values for v_c. If such information is available, one simple approach to dealing with multiple agents is to compute v_c such that it satisfies the ΔV threshold for any potential collision. Future work could explore possibilities for defining and implementing these soft stopping regions.



(a) $v_r = v$ (b) $v_r = 0.66v$ (c) $v_r = 0.33v$ (d) $v_r = 0$

Fig. 4: For a given agent state $A(\mathbf{x})$, reference velocity v_r, and set of followable paths P, the stopping region $\mathrm{SR}(A(\mathbf{x}), v_r, \mathbb{P})$ is the SR computed with respect to the reference velocity v_r. This figure illustrates SRs for various reference velocities in a two agent system. The two disc agents are traveling on a 2D plane with the same velocity v. In (a) the reference velocity is taken as v, so the stopping regions are the agents themselves. In (b)-(d) the reference velocity is taken as progressively smaller fractions of v. Here agents obey the same dynamics as in in Figure 2. It is important to note that these illustrations are 2D projections of 3D swept volumes, so the overlap in SRs is not quite as severe as it appears.

7 Conclusions

This work presented results showing that system dynamics can have a direct impact on both the theoretical complexity and solution space of multi-agent collision avoidance problems. The result is based on the fact that a requirement for agent coordination in a multi-agent system can fundamentally alter the problem model, and it was shown that system dynamics alone can add or remove this requirement. The proof of this assertion is constructive in nature, which allows the coordination requirement to be quantified. An exemplar problem was given to demonstrate the results and then future work and applications were discussed. In addition, the Appendix provides a proof that inertially unconstrained models cannot conservatively approximate constrained systems, and it provides a re-formulation of the velocity obstacle concept within the ICS family of representations.

Conceptually, this work deals with the fundamental question of how to appropriately model certain problems involving real-world interacting agents. Modeling as optimal decision making processes enables elegant formulations but requires the use of problem models that can be intractable to actually solve. This has serious practical implications that tend to be overlooked in academic literature. Daskalakis and Papadimitriou (2005) raised this issue in a study of complexity in large, multi-agent systems by posing the question: "How can one have faith in a model predicting that a group of agents will solve an intractable problem?" In the realm of multi-agent systems, this work suggests that such questions may be avoided by employing models that allow agents to independently modulate the problem complexity.

Acknowledgements The author would like to acknowledge the reviewers for providing many insightful comments and corrections, as well as Philipp Robbel and Elmar Mair for comments and feedback, and Valerie Aquila for assistance with figures and editing.

A Appendix

The first portion of the appendix will prove that the VO representation cannot conservatively approximate inertially constrained systems.It will also show that the VO representation belongs to the family of ICS representations using the *inevitable collision obstacle* (ICO) concept.

The last portion of the appendix will describe a conjecture about the problem complexity of finding a unique set of, collision-free, non-disjoint SPs.

A.1 The inevitable collision obstacle

Definition 15. An *inevitable collision obstacle* (ICO) is the set of states of an agent A that result in collision with $\mathcal{B}_i$ for any control sequence ϕ is applied to A:

$$\mathrm{ICO}(\mathcal{B}_i) = \{\mathbf{x} \mid \forall\phi, \exists t :: A(\phi(\mathbf{x}, t)) \cap \mathcal{B}_i \neq \emptyset\}$$

The ICO is closely related to the ICS concept, both of which were introduced by Fraichard and Asama (2004).

A.2 The velocity obstacle

This section recalls the velocity obstacle and relevant properties. We use the definitions from Fiorini and Shiller (1998), and the reader is referred to that work for more detail[6].

In this section, assume $t \in T$, where $T = [0, \infty)$ is a finite time horizon. Let Φ_v be the set of feasible velocity commands for A, and let $\phi_v(\mathbf{x}, t)$ denote the state of A after constant velocity v is applied to initial state $\mathbf{x}$ for a time t.

Definition 16. The *velocity obstacle* for A due to O_i, written $\text{VO}_{A|O_i}$, is the set of velocities such that A at some point enters into a collision state with O. In other words, given initial state $\mathbf{x}$, and for all feasible velocity commands $v \in \Phi_v$ there is a collision at some time $t \in T$ between $A(\mathbf{x})$ and the state space obstacle $\mathcal{B}_i$ due to O_i:

$$\text{VO}_{A|O_i} = \{v \mid \exists t :: A(\phi_v(\mathbf{x}, t)) \cap \mathcal{B}_i \neq \emptyset\}$$

A.3 Velocity obstacles and inertially constrained systems

Lemma 9 *The VO representation cannot guarantee collision avoidance in inertially constrained systems.*

Proof. By Definition 16, the complement of the velocity obstacle is exactly the set of all velocities that, when instantaneously applied, would avoid collision. However, controlling to a velocity instantaneously is impossible in an inertially constrained system. Therefore, the complement of the velocity obstacle is unreachable, and by Lemma 1, it cannot be used to guarantee non-collision. □

Velocity obstacle and inevitable collision obstacle equivalence

The reader will note the similarity between Definition 15 & 16, and work by Shiller et al. (2010) suggests that a deeper relationship exists. The proof proceeds by exploiting the similarity and showing that ICO computations are both necessary and sufficient in order to compute the VO.

Definition 17. A *velocity* ICO (ICO_v) for a given state space obstacle $\mathcal{B}_i$ is an ICO computed over the velocity control trajectory set Φ_v:

$$\text{ICO}(\mathcal{B}_i)_v = \{\mathbf{x} \mid \forall \phi_v, \exists t :: A(\phi_v(\mathbf{x}, t)) \cap \mathcal{B}_i \neq \emptyset\}$$

Lemma 10 *Computing a velocity obstacle is exactly equivalent to computing an inevitable collision obstacle over a restricted control space.*

Proof. For a given obstacle O_i and corresponding state space obstacle $\mathcal{B}_i$, use Definition 16 to perform a variable rewrite on the definition of a velocity ICO (Definition 17):

$$\begin{aligned}\text{ICO}(\mathcal{B}_i)_v &= \{\mathbf{x} \mid \forall \phi_v, \exists t :: A(\phi_v(\mathbf{x}, t)) \cap \mathcal{B}_i \neq \emptyset\} \\ &= \{\mathbf{x} \mid \forall \phi_v, v \in \text{VO}_{A|O_i}\}\end{aligned}$$

Thus, the $\text{ICO}(\mathcal{B})_v$ and $\text{VO}_{A|O}$ are equivalent, which means that the velocity obstacle representation is equivalent to the ICO representation over a restricted control space. □

[6] To avoid problems in dealing with dynamic constraints Wilkie et al. (2009) defined *generalized velocity obstacles* that are derived in control space rather than velocity space.

The result of Lemma 10 provides a simple but formal unification of two common techniques for collision avoidance under the same theoretical framework: that velocity obstacles are exactly inevitable collision obstacles over a restricted set of the inputs.

A.4 A Special Case of Coordination

The proof of Lemma 3 asserts through Definition 3 that finding a unique set of collision-free, non-disjoint SPs induces a coordination requirement. Invoking this type of coordination has interesting complexity implications because the general problem of identifying a unique assignment of such SPs is likely reducible to a Unique-SAT problem, which is coNP-Hard (Blass and Gurevich, 1982). The following conjecture captures this:

Conjecture 2 There is no efficiently computable (i.e. P-time) solution to identifying a unique set of collision-free stopping paths in a system that does not exhibit SP disjointness.

Investigation of Conjecture 2 would be an interesting point for future work.

References

Abrams S, Allen PK (2000) Computing swept volumes. Journal of Visualization and Computer Animation 11(2):69–82, DOI 10.1002/1099-1778(200005)11:2⟨69::AID-VIS219⟩3.0.CO;2-7, URL `http://dx.doi.org/10.1002/1099-1778(200005)11:2<69::AID-VIS219>3.0.CO;2-7`

Allen RE, Clark AA, Starek JA, Pavone M (2014) A machine learning approach for real-time reachability analysis. In: 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems, Chicago, IL, USA, September 14-18, 2014, IEEE, pp 2202–2208, DOI 10.1109/IROS.2014.6942859, URL `http://dx.doi.org/10.1109/IROS.2014.6942859`

Alonso-Mora J, Breitenmoser A, Rufli M, Beardsley PA, Siegwart R (2010) Optimal reciprocal collision avoidance for multiple non-holonomic robots. In: Martinoli A, Mondada F, Correll N, Mermoud G, Egerstedt M, Hsieh MA, Parker LE, Støy K (eds) Distributed Autonomous Robotic Systems - The 10th International Symposium, DARS 2010, Lausanne, Switzerland, November 1-3, 2010, Springer, Springer Tracts in Advanced Robotics, vol 83, pp 203–216, DOI 10.1007/978-3-642-32723-0_15, URL `http://dx.doi.org/10.1007/978-3-642-32723-0_15`

Alterovitz R, Siméon T, Goldberg KY (2007) The stochastic motion roadmap: A sampling framework for planning with markov motion uncertainty. In: Burgard W, Brock O, Stachniss C (eds) Robotics: Science and Systems III, June 27-30, 2007, Georgia Institute of Technology, Atlanta, Georgia, USA, The MIT Press, URL `http://www.roboticsproceedings.org/rss03/p30.html`

Bacha A, Bauman C, Faruque R, Fleming M, Terwelp C, Reinholtz CF, Hong D, Wicks A, Alberi T, Anderson D, Cacciola S, Currier P, Dalton A, Farmer J, Hurdus J, Kimmel S, King P, Taylor A, Covern DV, Webster M (2008) Odin: Team victortango's entry in the DARPA urban challenge. J Field Robotics 25(8):467–492, DOI 10.1002/rob.20248, URL `http://dx.doi.org/10.1002/rob.20248`

Baek N, Shin SY, Chwa K (1999) On computing translational swept volumes. Int J Comput Geometry Appl 9(3):293–317, DOI 10.1142/S0218195999000200, URL http://dx.doi.org/10.1142/S0218195999000200

Bekris KE (2010) Avoiding inevitable collision states: Safety and computational efficiency in replanning with sampling-based algorithms. International Conference on Robotics and Automation (ICRA-10) URL http://www.cs.rutgers.edu/~kb572/pubs/ics_tradeoffs.pdf

Bekris KE, Grady DK, Moll M, Kavraki LE (2012) Safe distributed motion coordination for second-order systems with different planning cycles. I J Robotic Res 31(2):129–150, DOI 10.1177/0278364911430420, URL http://dx.doi.org/10.1177/0278364911430420

van den Berg J, Guy SJ, Lin MC, Manocha D (2009) Reciprocal n-body collision avoidance. In: Pradalier C, Siegwart R, Hirzinger G (eds) Robotics Research - The 14th International Symposium, ISRR 2009, August 31 - September 3, 2009, Lucerne, Switzerland, Springer, Springer Tracts in Advanced Robotics, vol 70, pp 3–19, DOI 10.1007/978-3-642-19457-3_1, URL http://dx.doi.org/10.1007/978-3-642-19457-3_1

van den Berg JP, Snape J, Guy SJ, Manocha D (2011) Reciprocal collision avoidance with acceleration-velocity obstacles. In: IEEE International Conference on Robotics and Automation, ICRA 2011, Shanghai, China, 9-13 May 2011, IEEE, pp 3475–3482, DOI 10.1109/ICRA.2011.5980408, URL http://dx.doi.org/10.1109/ICRA.2011.5980408

Bernstein DS, Givan R, Immerman N, Zilberstein S (2002) The complexity of decentralized control of markov decision processes. Math Oper Res 27(4):819–840, DOI 10.1287/moor.27.4.819.297, URL http://dx.doi.org/10.1287/moor.27.4.819.297

Blass A, Gurevich Y (1982) On the unique satisfiability problem. Information and Control 55(1-3):80–88, DOI 10.1016/S0019-9958(82)90439-9, URL http://dx.doi.org/10.1016/S0019-9958(82)90439-9

Boutilier C (1996) Planning, learning and coordination in multiagent decision processes. In: Shoham Y (ed) Proceedings of the Sixth Conference on Theoretical Aspects of Rationality and Knowledge, De Zeeuwse Stromen, The Netherlands, March 17-20 1996, Morgan Kaufmann, pp 195–210

Chazelle B, Dobkin DP (1980) Detection is easier than computation (extended abstract). In: Miller RE, Ginsburg S, Burkhard WA, Lipton RJ (eds) Proceedings of the 12th Annual ACM Symposium on Theory of Computing, April 28-30, 1980, Los Angeles, California, USA, ACM, pp 146–153, DOI 10.1145/800141.804662, URL http://doi.acm.org/10.1145/800141.804662

Daskalakis K, Papadimitriou CH (2005) The complexity of games on highly regular graphs. In: Brodal GS, Leonardi S (eds) Algorithms - ESA 2005, 13th Annual European Symposium, Palma de Mallorca, Spain, October 3-6, 2005, Proceedings, Springer, Lecture Notes in Computer Science, vol 3669, pp 71–82, DOI 10.1007/11561071_9, URL http://dx.doi.org/10.1007/11561071_9

von Dziegielewski A, Erbes R, Schömer E (2010) Conservative swept volume boundary approximation. In: Elber G, Fischer A, Keyser J, Kim M (eds) ACM Symposium on Solid and Physical Modeling, Proceedings of the 14th ACM Symposium on Solid and Physical Modeling, SPM 2010, Haifa, Israel, September 1-3, 2010, ACM, pp 171–176, DOI 10.1145/1839778.1839804, URL http://doi.acm.org/10.1145/1839778.1839804

Eberly D (2008) Intersection of Convex Objects: The Method of Separating Axes. https://www.geometrictools.com/Documentation/MethodOfSeparatingAxes.pdf, [Online; accessed 02-January-2017]

Erickson LH, LaValle SM (2013) A simple, but NP-hard, motion planning problem. In: desJardins M, Littman ML (eds) Proceedings of the Twenty-Seventh AAAI Conference on Artificial Intelligence, July 14-18, 2013, Bellevue, Washington, USA., AAAI Press, URL http://www.aaai.org/ocs/index.php/AAAI/AAAI13/paper/view/6280

Ericson C (2005) Real-time collision detection. Morgan Kaufmann series in interactive 3D technology, Elsevier, Amsterdam, Boston, Paris, URL http://opac.inria.fr/record=b1121294

Fiorini P, Shiller Z (1998) Motion planning in dynamic environments using velocity obstacles. I J Robotic Res 17(7):760–772, DOI 10.1177/027836499801700706, URL http://dx.doi.org/10.1177/027836499801700706

Fraichard T, Asama H (2004) Inevitable collision states - a step towards safer robots? Advanced Robotics 18(10):1001–1024, DOI 10.1163/1568553042674662, URL http://dx.doi.org/10.1163/1568553042674662

Gilbert EG, Johnson DW, Keerthi SS (1988) A fast procedure for computing the distance between complex objects in three-dimensional space. IEEE J Rob Autom 4(2):193–203, DOI 10.1109/56.2083, URL http://dx.doi.org/10.1109/56.2083

Goldsmith J, Mundhenk M (2007) Competition adds complexity. In: Platt JC, Koller D, Singer Y, Roweis ST (eds) Advances in Neural Information Processing Systems 20, Proceedings of the Twenty-First Annual Conference on Neural Information Processing Systems, Vancouver, British Columbia, Canada, December 3-6, 2007, Curran Associates, Inc., pp 561–568, URL http://papers.nips.cc/paper/3163-competition-adds-complexity

Halperin D, Sharir M (1996) A near-quadratic algorithm for planning the motion of a polygon in a polygonal environment. Discrete & Computational Geometry 16(2):121–134, DOI 10.1007/BF02716803, URL http://dx.doi.org/10.1007/BF02716803

Harding J, Powell G, Yoon R, Fikentscher J, Doyle C, Sade D, Lukuc M, Simons J, Wang J (2014) Vehicle-to-vehicle communications: Readiness of v2v technology for application. Tech. Rep. DOT HS 812 014, U.S. Department of Transportation, National Highway Transportation Safety Administration, National Highway Traffic Safety Administration, 1200 New Jersey Avenue SE. Washington, DC 20590, URL http://www.nhtsa.gov/staticfiles/rulemaking/pdf/V2V/Readiness-of-V2V-Technology-for-Application-812014.pdf

Hauser KK (2012) The minimum constraint removal problem with three robotics applications. In: Frazzoli E, Lozano-Pérez T, Roy N, Rus D (eds) Algorithmic Foundations of Robotics X - Proceedings of the Tenth Workshop on the Algorithmic Foundations of Robotics, WAFR 2012, MIT, Cambridge, Massachusetts, USA, June 13-15 2012, Springer, Springer Tracts in Advanced Robotics, vol 86, pp 1–17, DOI 10.1007/978-3-642-36279-8_1, URL http://dx.doi.org/10.1007/978-3-642-36279-8_1

Hopcroft J, Schwartz J, Sharir M (1984) On the complexity of motion planning for multiple independent objects; pspace-hardness of the “warehouseman’s problem”. The International Journal of Robotics Research 3(4):76?88, URL http://ijr.sagepub.com/content/3/4/76.short

Hornung A, Wurm KM, Bennewitz M, Stachniss C, Burgard W (2013) OctoMap: An efficient probabilistic 3D mapping framework based on octrees. Autonomous Robots DOI 10.1007/s10514-012-9321-0, URL http://octomap.github.com, software available at http://octomap.github.com

Jansson J (2005) Collision Avoidance Theory: With Application to Automotive Collision Mitigation. Linköping studies in science and technology: Dissertations, Department of Electrical Enginering, Univ., URL https://books.google.com/books?id=ik8wNQAACAAJ

Jiménez P, Thomas F, Torras C (2001) 3d collision detection: a survey. Computers & Graphics 25(2):269–285, DOI 10.1016/S0097-8493(00)00130-8, URL http://dx.doi.org/10.1016/S0097-8493(00)00130-8

Johnson J, Hauser KK (2012) Optimal acceleration-bounded trajectory planning in dynamic environments along a specified path. In: IEEE International Conference on Robotics and Automation, ICRA 2012, 14-18 May, 2012, St. Paul, Minnesota, USA, IEEE, pp 2035–2041, DOI 10.1109/ICRA.2012.6225233, URL http://dx.doi.org/10.1109/ICRA.2012.6225233

Johnson JK (2016) A novel relationship between dynamics and complexity in multi-agent collision avoidance. In: Proceedings of Robotics: Science and Systems, Ann Arbor, Michigan, DOI 10.15607/RSS.2016.XII.030

Kaelbling LP, Lozano-Pérez T (2013) Integrated task and motion planning in belief space. I J Robotic Res 32(9-10):1194–1227, DOI 10.1177/0278364913484072, URL http://dx.doi.org/10.1177/0278364913484072

Kamat VV (1993) A survey of techniques for simulation of dynamic collision detection and response. Computers & Graphics 17(4):379–385, DOI 10.1016/0097-8493(93)90024-4, URL http://dx.doi.org/10.1016/0097-8493(93)90024-4

Kambhampati S, Cutkosky MR, Tenenbaum M, Lee SH (1991) Combining specialized reasoners and general purpose planners: A case study. In: Dean TL, McKeown K (eds) Proceedings of the 9th National Conference on Artificial Intelligence, Anaheim, CA, USA, July 14-19, 1991, Volume 1., AAAI Press / The MIT Press, pp 199–205, URL http://www.aaai.org/Library/AAAI/1991/aaai91-032.php

Kim YJ, Varadhan G, Lin MC, Manocha D (2004) Fast swept volume approximation of complex polyhedral models. Computer-Aided Design 36(11):1013–1027, DOI 10.1016/j.cad.2004.01.004, URL http://dx.doi.org/10.1016/j.cad.2004.01.004

LaValle SM (2006) Planning algorithms. Cambridge University Press

LaValle SM, Kuffner JJ (2001) Randomized kinodynamic planning. I J Robotic Res 20(5):378–400, DOI 10.1177/02783640122067453, URL http://dx.doi.org/10.1177/02783640122067453

Lin MC, Canny JF (1991) A Fast Algorithm for Incremental Distance Calculation. In: IEEE International Conference on Robotics and Automation, pp 1008–1014, URL http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.30.2174

Littman ML, Dean TL, Kaelbling LP (1995) On the complexity of solving markov decision problems. In: UAI '95: Proceedings of the Eleventh Annual Conference on Uncertainty in Artificial Intelligence, Montreal, Quebec, Canada, August 18-20, 1995, pp 394–402, URL https://dslpitt.org/uai/displayArticleDetails.jsp?mmnu=1&smnu=2&article_id=457&proceeding_id=11

Mazer E, Ahuactzin JM, Bessière P (1998) The ariadne's clew algorithm. J Artif Intell Res (JAIR) 9:295–316, DOI 10.1613/jair.468, URL http://dx.doi.org/10.1613/jair.468

Mitter S, Sahai A (1999) Information and control: Witsenhausen revisited, Springer London, London, pp 281–293. DOI 10.1007/BFb0109735, URL http://dx.doi.org/10.1007/BFb0109735

Montemerlo M, Becker J, Bhat S, Dahlkamp H, Dolgov D, Ettinger S, Hähnel D, Hilden T, Hoffmann G, Huhnke B, Johnston D, Klumpp S, Langer D, Levandowski A, Levinson J, Marcil J, Orenstein D, Paefgen J, Penny I, Petrovskaya A, Pflueger M, Stanek G, Stavens D, Vogt A, Thrun S (2008) Junior: The stanford entry in the urban challenge. J Field Robotics 25(9):569–597, DOI 10.1002/rob.20258, URL http://dx.doi.org/10.1002/rob.20258

Ó'Dúnlaing C (1987) Motion planning with inertial constraints. Algorithmica 2:431–475, DOI 10.1007/BF01840370, URL http://dx.doi.org/10.1007/BF01840370

Oliehoek FA, Witwicki SJ, Kaelbling LP (2012) Influence-based abstraction for multiagent systems. In: Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence, July 22-26, 2012, Toronto, Ontario, Canada., URL http://www.aaai.org/ocs/index.php/AAAI/AAAI12/paper/view/5047

Paden B, Cáp M, Yong SZ, Yershov DS, Frazzoli E (2016) A survey of motion planning and control techniques for self-driving urban vehicles. CoRR abs/1604.07446, URL http://arxiv.org/abs/1604.07446

Papadimitriou CH, Tsitsiklis JN (1987) The complexity of markov decision processes. Mathematics of Operations Research 12(3):441–450, URL http://www.jstor.org/stable/3689975

Petti S, Fraichard T (2005) Safe motion planning in dynamic environments. In: 2005 IEEE/RSJ International Conference on Intelligent Robots and Systems, Edmonton, Alberta, Canada, August 2-6, 2005, IEEE, pp 2210–2215, DOI 10.1109/IROS.2005.1545549, URL http://dx.doi.org/10.1109/IROS.2005.1545549

Reif JH (1979) Complexity of the mover's problem and generalizations (extended abstract). In: 20th Annual Symposium on Foundations of Computer Science, San Juan, Puerto Rico, 29-31 October 1979, IEEE Computer Society, pp 421–427, DOI 10.1109/SFCS.1979.10, URL http://dx.doi.org/10.1109/SFCS.1979.10

Reif JH, Sharir M (1985) Motion planning in the presence of moving obstacles. In: 26th Annual Symposium on Foundations of Computer Science, Portland, Oregon, USA, 21-23 October 1985, IEEE Computer Society, pp 144–154, DOI 10.1109/SFCS.1985.36, URL http://dx.doi.org/10.1109/SFCS.1985.36

Schoettle B, Sivak M (2015) A Preliminary Analysis of Real-World Crashes Involving Self-Driving Vehicles. Tech. rep., The University of Michigan Transportation Research Institute

Shamos M (1978) Computational geometry. PhD thesis, Yale University

Shiller Z, Gal O, Fraichard T (2010) The Nonlinear Velocity Obstacle Revisited: the Optimal Time Horizon. In: Guaranteeing Safe Navigation in Dynamic Environments Workshop, Anchorage, United States, URL https://hal.inria.fr/inria-00562249

Shoham Y, Tennenholtz M (1995) On social laws for artificial agent societies: Off-line design. Artif Intell 73(1-2):231–252, DOI 10.1016/0004-3702(94)00007-N, URL http://

dx.doi.org/10.1016/0004-3702(94)00007-N

Solovey K, Halperin D (2015) On the hardness of unlabeled multi-robot motion planning. In: Robotics: Science and Systems XI, Sapienza University of Rome, Rome, Italy, July 13-17, 2015, URL http://www.roboticsproceedings.org/rss11/p46.html

Täubig H, Bäuml B, Frese U (2011) Real-time swept volume and distance computation for self collision detection. In: 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2011, San Francisco, CA, USA, September 25-30, 2011, IEEE, pp 1585–1592, DOI 10.1109/IROS.2011.6094611, URL http://dx.doi.org/10.1109/IROS.2011.6094611

Urmson C, Anhalt J, Bagnell JAD, Baker CR, Bittner RE, Dolan JM, Duggins D, Ferguson D, Galatali T, Geyer H, Gittleman M, Harbaugh S, Hebert M, Howard T, Kelly A, Kohanbash D, Likhachev M, Miller N, Peterson K, Rajkumar R, Rybski P, Salesky B, Scherer S, Seo YW, Simmons R, Singh S, Snider JM, Stentz AT, Whittaker WRL, Ziglar J (2007) Tartan racing: A multi-modal approach to the darpa urban challenge. Tech. Rep. CMU-RI-TR-, Robotics Institute, Pittsburgh, PA, URL http://www.ri.cmu.edu/publication_view.html?pub_id=6906

Valtazanos A, Ramamoorthy S (2011) Online motion planning for multi-robot interaction using composable reachable sets. In: Röfer T, Mayer NM, Savage J, Saranli U (eds) RoboCup 2011: Robot Soccer World Cup XV [papers from the 15th Annual RoboCup International Symposium, Istanbul, Turkey, July 2011], Springer, Lecture Notes in Computer Science, vol 7416, pp 186–197, DOI 10.1007/978-3-642-32060-6_16, URL http://dx.doi.org/10.1007/978-3-642-32060-6_16

Weller R (2013) New Geometric Data Structures for Collision Detection and Haptics. Springer Series on Touch and Haptic Systems, Springer International Publishing, URL http://www.springer.com/computer/theoretical+computer+science/book/978-3-319-01019-9

Wilkerson JL, Bobinchak J, Culp M, Clark J, Halpin-Chan T, Estabridis K, Hewer G (2014) Two-dimensional distributed velocity collision avoidance. Tech. Rep. NAWCWD TP 8786, Physics Division, Research and Intelligence Department, Naval Air Warfare Center Weapons Division, China Lake, CA 93555-6100, URL http://www.dtic.mil/dtic/tr/fulltext/u2/a598520.pdf

Wilkie D, van den Berg JP, Manocha D (2009) Generalized velocity obstacles. In: 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems, October 11-15, 2009, St. Louis, MO, USA, IEEE, pp 5573–5578, DOI 10.1109/IROS.2009.5354175, URL http://dx.doi.org/10.1109/IROS.2009.5354175

Witsenhausen HS (1968) A counterexample in stochastic optimum control. SIAM Journal on Control 6(1):131–147, DOI 10.1137/0306011, URL http://dx.doi.org/10.1137/0306011, http://dx.doi.org/10.1137/0306011

Ziegler J, Bender P, Dang T, Stiller C (2014) Trajectory planning for bertha - A local, continuous method. In: 2014 IEEE Intelligent Vehicles Symposium Proceedings, Dearborn, MI, USA, June 8-11, 2014, IEEE, pp 450–457, DOI 10.1109/IVS.2014.6856581, URL http://dx.doi.org/10.1109/IVS.2014.6856581