Pareto optimal coordination on roadmaps*

Robert Ghrist(1), Jason M. O'Kane(2), and Steven M. LaValle(2)

Departments of Mathematics(1) and Computer Science(2), University of Illinois, Urbana, IL 61801, USA

Abstract. Given a collection of robots sharing a common environment, assume that each possesses an individual roadmap for its C-space and a cost function for attaining a goal. Vector-valued (or PARETO) optima for collision-free coordination are by no means unique: in fact, continua of optimal coordinations are possible. However, for CYLINDRICAL obstacles (those defined by pairwise interactions), we prove a finite bound on the number of optimal coordinations. For such systems, we present an exact subquadratic algorithm for reducing a coordination scheme to its Pareto optimal representative.

1 Introduction

In numerous settings, the coordination of multiple robots remains a basic and challenging research issue. Autonomous guided vehicles (AGVs) are used in a wide variety of industrial settings for problems such as material handling, palletizing, paper roll handling, assembly, conveying, and people moving. Typically, AGVs reliably traverse a fixed roadmap of paths in a complicated factory environment. Although the paths avoid collisions with obstacles in the workspace, the efforts of numerous AGVs may have to be coordinated in a way that avoids collisions between AGVs while at the same time maximizing productivity.

If we wish to coordinate the motions of N robots in a common environment, what is an appropriate notion of optimality? Minimizing the average time robots take to reach their goal? Minimizing the time that the last robot takes? Such approaches are common (e.g., [12,16,23]) and may be appropriate in some cases; however, it is important to recognize that scalarization of a vector of N criteria occurs in this process. Each robot has its own cost function, e.g., elapsed time. These N criteria are then converted — often in an arbitrary manner — into a single criterion to be optimized.

In this paper, we investigate the optimization problem for multiple robot coordination without scalarizing the vector-valued cost function. This centers on the notion of *Pareto optimality* [19,20], a concept which is widely used in mathematical economics to model individual consumers striving to optimize distinct economic goals. This brand of optimization is more "faithful" in the sense that no data is lost by scalarization. In particular:

* Supported by NSF Career DMS-0337713 [RG] and NSF IIS-0296126 [JO,SL].

1. Any optimum of *any* scalarization of the vector-valued cost function (which is monotone increasing in the components) is in fact one of the Pareto optima: see Lemma 1.
2. For settings in which priorities change over time, knowledge of the Pareto optima allow for easy adaptation. This is particularly relevant in automation settings where repetitive motions are subject to priority changes.
3. By filtering out all of the motion plans that are not worth considering and presenting the user with a small set of the best alternatives, additional criteria, such as priority or the amount of sacrifice one robot makes, can be applied to automatically select a particular motion plan.

Given the desire to filter the space of all possible coordination schemes to a small set of best cases independent of biases on the agents, we are certainly most interested in the cases where this collection of optima is *finite*. A finiteness criterion and an exact algorithm for these optima are the principal results of this paper.

1.1 History

The problem of coordinating robots along fixed roadmaps can be considered as a special case of general motion planning for multiple robots. Previous approaches to multiple-robot motion planning are often categorized as *centralized* or *decoupled*. A centralized approach typically constructs a path in a composite configuration space, which is derived from the Cartesian product of the configuration spaces of the individual robots (e.g., [1,2,21]). A decoupled approach typically generates paths for each robot independently, and then considers the interactions between the robots (e.g., [4,10]). In [3,6,18,22] robot paths are independently determined, and a coordination diagram is used to plan a collision-free trajectory along the paths. In [14,24], an independent roadmap is computed for each robot, and coordination occurs on the Cartesian product of the roadmap path domains. The suitability of one approach over the other is usually determined by the trade-off between computational complexity associated with a given problem, and the amount of completeness that is lost. In some applications, such as the coordination of AGVs, the roadmap might represent all allowable mobility for each robot.

In this paper, we focus on the multiplicity of Parteo-optimal path coordinations among cylindrical obstacles — those determined by pairwise collisions (see §1.4 below). In [15], an approximate Dijkstra-like algorithm to find Pareto-optimal solutions in this context was given. To our knowledge, the only previously known exact solution is that of [8], which applies only to the case of two robots that translate on acyclic roadmaps.

1.2 Coordination spaces

This paper concerns the coordination of N robots, each having a roadmap Γ_i (a graph within the C-space of the i^{th} robot) precomputed independent of the other agents.

Definition 1. A ROADMAP COORDINATION SPACE of $\{\Gamma_i\}_1^N$ is any space of the form $\mathcal{X} := (\Gamma_1 \times \cdots \times \Gamma_N) - \mathcal{O}$, where $\mathcal{O}$ denotes an (open) obstacle set.

For the remainder of the paper, all coordination spaces are assumed to be *piecewise linear* (PL) and *regularly closed* (i.e., $\mathcal{X}$ is the closure of its interior). These assumptions rule out obstacle sets $\mathcal{O}$ which are too intricate to have a nice local structure or for which the system "locks up" in a singular configuration. We expect that the results of this paper hold under much weaker assumptions (one should be able to ignore regularly closed and replace PL with piecewise analytic or algebraic).

A special example of a roadmap coordination space arises when all of the roadmaps Γ_i are identical, and the obstacle set is an open neighborhood of $\{x_i = x_j : \text{for some } i \neq j\}$. In this case, one can consider the workspace to be the graph itself, and the roadmap coordination space is precisely the configuration space of N labelled objects on the graph.

1.3 Pareto optimality

Pareto optimization refers to optimization of vector-valued functions. In the context of robotics applications, Pareto optimization arises when distinct agents possess distinct goals and cost functions for evaluating performance. Each agent wishes to optimize its cost function independently of the others.

Mathematically, this is characterized as follows. Given a parameterized path $\gamma : [0, T] \to \mathcal{X}$ in a coordination space, each robot executes the projected path $\gamma_i := \text{PROJ}_i \circ \gamma$, where PROJ_i denotes projection onto the i^{th} factor. Given cost functions $\{\tau_i\}$, $i = 1 ... N$, the cost vector for γ is the vector $\tau(\gamma) := (\tau_i(\gamma_i))_i^N$. The case in which τ_i measures *elapsed time* from start to goal is an important and characteristic example, though more general cost functions are allowed.

We assume that each roadmap Γ_i is outfitted with a metric such that the speed of the i^{th} robot travelling along this graph at maximal speed is exactly one; hence all admissible paths have speeds whose components are bounded above by one. It will be assumed for simplicity that the cost functions τ_i agree with elapsed time (a suitable class of more general cost functions can be used by deforming the geometry of the coordination space).

A path $\gamma : [0, T] \to \mathcal{X}$ is PARETO-OPTIMAL iff $\tau(\gamma)$ is minimal with respect to the partial order on vectors:

$$\tau(\gamma) \leq \tau(\gamma') \Leftrightarrow \tau_i(\gamma_i) \leq \tau_i(\gamma_i') \,\forall\, i. \tag{1}$$

The Pareto optima comprise the set of all optima for all monotone scalarizations (such as, *e.g.*, average time-to-goal and all non-linear generalizations thereof):

Lemma 1. *For any scalarization $f : \mathbb{R}^n \to \mathbb{R}$ with $\partial f / \partial x_i > 0$, all minima of $f \circ \tau$ are Pareto optima.*

Proof: Given any minimal path for $f \circ \tau$ which is not Pareto-optimal, deforming it to a Pareto-optimal path decreases some τ_i without increasing any of the others; hence it decreases the f-value: contradiction. □

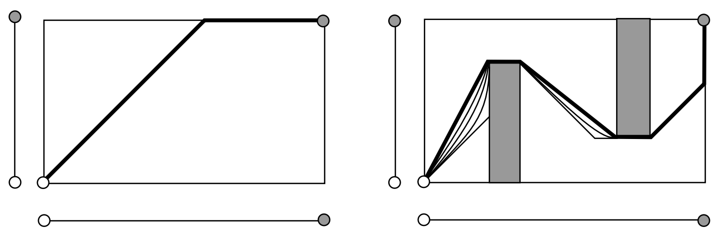

Fig. 1. [left] The unique Pareto optimal path on a rectangle with regards to elapsed time; [right] an envelope of Pareto optimal paths weaving through obstacles forms a single equivalence class.

Pareto optimal paths are rarely unique. Two paths γ and γ' are PARETO EQUIVALENT iff they are homotopic through Pareto-optimal paths which are equal in the partial order; *i.e.*, $\tau(\gamma) = \tau(\gamma')$. Fig. 1[right] illustrates a single Pareto-optimal class with many representatives. We show in §2 that certain roadmap coordination spaces admit a continuum of Pareto-optimal *classes*.

1.4 Contributions

For a special class of coordination spaces, we give a **finiteness bound** for the number of Pareto optima and an **exact algorithm** for computing Pareto-optimal representatives of paths.

We say that the obstacle set $\mathcal{O}$ for $\mathcal{X} = \prod \Gamma_i - \mathcal{O}$ is CYLINDRICAL if $\mathcal{O}$ is of the form

$$\mathcal{O} = \bigcup_{i<j} \left(\Delta_{i,j} \times \prod_{k \neq i,j} \Gamma_k \right), \tag{2}$$

with $\Delta_{i,j} \subset \Gamma_i \times \Gamma_k$. Physically, this means that whenever two agents experience an obstacle (*e.g.*, collision), the states of all other agents are irrelevant:

collisions are pairwise-determined. We require each $\Delta_{i,j}$ to be the interiors of disjoint PL polygonal subsets of $\Gamma_i \times \Gamma_j$.

Main Theorem: Let $\mathcal{X}$ be a cylindrical coordination space:

1. Locally Pareto-optimal path classes are in bijective correspondence with homotopy classes of paths (fixing the endpoints).
2. There is a finite bound on the number of globally Pareto-optimal path classes.

The first statement is that every path in $\mathcal{X}$ is homotopic to a unique class which is Pareto-optimal among paths in its homotopy class; the second statement is that only a finite number of these local optima are in fact global. Proofs appear in §4.

In §5, we present an exact algorithm for determining the local Pareto-optimal representative of a path in $\mathcal{X}$ (under the reasonable assumption that robots do not backtrack). The algorithm is output sensitive with respect to the complexity of the path it computes. Fix the number of robots; let m denote the complexity of the obstacle set $\mathcal{O}$; and let p and p' denote the complexities of the input and output paths respectively. Algorithm 2 executes in $O(p + m \log mp + p'\sqrt{m} \log m)$ time. An extension to arbitrary paths involving backtracking is outlined.

2 Examples

We illustrate a few examples of simple coordination spaces and Pareto optimal path classes.

Example 1. Consider the case where $N = 2$, $\Gamma_1 = \Gamma_2 = [-2, 2]$, and $\mathcal{O} = \{(x, y) : x^2 + y^2 < 1\}$, which corresponds to a pair of identical disc-shaped AGV's sliding along interval roadmaps which intersect in the workspace at right angles. There are exactly two Pareto-optimal classes of paths from $(-2, -2)$ to $(2, 2)$, as illustrated in Fig. 2[left]. The difference between these two paths lies in which robot decides to pause in order to allow the other to pass through the intersection. Note also that this is a cylindrical coordination space and hence satisfies the Main Theorem above.

Example 2. We modify the previous example by letting $N = 3$ and choosing $\mathcal{O}$ to be a round ball of radius 1 at the origin. By the symmetry of $\mathcal{X}$ about the diagonal of the cube, it is clear that there is a circle's worth of paths which begin at $(-2, -2)$, trace a straight line which is tangent to $\mathcal{O}$, and then exit this sphere tangentially with slope one. The projection of this family of paths to the first two coordinates includes as special cases the distinct Pareto optima of Example 1, as well as a continuum of paths whose goal times interpolate between these two. Hence, there is a continuum of Pareto optimal classes.

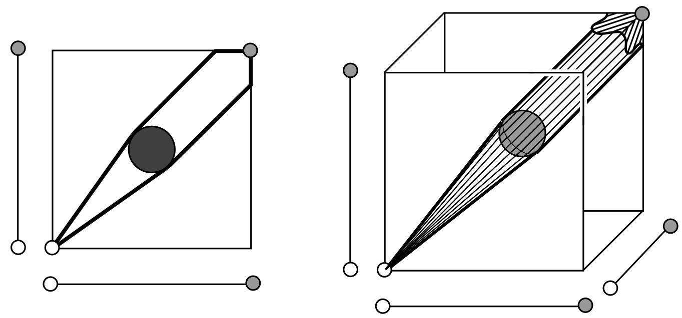

Fig. 2. [left] The case of two agents with a disc obstacle possesses two Pareto-optimal classes; [right] Three agents with a spherical obstacle possesses a continuum of Pareto-optimal classes.

Though neither of these examples is PL, the arguments for the number of Pareto-optimal classes are robust and valid for PL approximations to $\mathcal{O}$.

Example 3. Coordination problems without the fixed path constraint can have continua of optima much more easily than in the fixed path case. Consider the problem of translating two unit squares in the plane in such a way as to exchange their positions. Assume that the squares are centered at $(-1,0)$ and $(1,0)$. It is straightforward to show a continuous family of Pareto-optimal classes. Extrema of this family consist of those coordinations for which one square translates exclusively in the horizontal direction, while the other square moves vertically in order to get out of the way. The intermediate coordinations involve one square translating up by an amount $0 < h < 1$ and the other translating down by a total amount $1 - h$. This example works with arbitrary translations, or with translations restricted to coordinate axes directions.

3 Discretized coordination spaces and cube paths

The key step in our analysis of cylindrical coordination spaces is a spatial discretization of them into cubical complexes. This step is inspired by techniques in geometric group theory (*e.g.*, [17]) which have found other applications in robotics contexts [11].

Consider a coordination space $\mathcal{X}$ for roadmaps $\{\Gamma_i\}$. For simplicity, assume that all the edge lengths of the graphs Γ_i are rationally related. Because of this, one may choose a length $\delta > 0$ so that all edge lengths are integer multiples of δ. Denote by $\Gamma_i^{(n)}$ the graph obtained from Γ_i by subdividing it into edges of length $2^{-n}\delta$.

Definition 2. For each $n \geq 0$, denote by $\mathcal{X}^{(n)} \subset \Gamma_1^{(n)} \times \cdots \times \Gamma_N^{(n)}$ the maximal closed subcomplex of the product cubical complex which does not intersect the obstacle set $\mathcal{O}$. This cubical complex is called the STAGE-n DISCRETIZED COORDINATION SPACE.

These discretized coordination spaces "converge" (in homotopy type) to the full coordination space $\mathcal{X}$ as $n \to \infty$. The cubical nature of these spaces forms a convenient structure with which to manipulate paths. Our strategy is to approximate Pareto-optimal paths with paths which follow along the diagonals of cubes in $\mathcal{X}^{(n)}$ for n sufficiently large.

Definition 3. Let X denote a cubical complex. A CUBE PATH from vertices v_0 to v_N in X is an ordered sequence of closed cubes $\mathcal{C} = \{C_i\}_1^N$ of X which satisfy (1) $C_i \cap C_{i+1} = v_i$, a vertex; and (2) C_i is the smallest cube of X containing v_{i-1} and v_i. A cube path is said to be LEFT-GREEDY if in addition $\forall i$ (3) $C_{i+1} \cap \mathsf{St}(C_i) = v_i$, where $\mathsf{St}(C_i)$ is the STAR of C_i (all cells, including C_i, which have C_i as a face).

To each cube path is associated a PL path in X given by the chain of straight line segments from v_i to v_{i+1} for $i = 0...N-1$. Roughly speaking, a left-greedy cube path is one which uses the highest dimensional cubes as early as possible in the path. This hints at the intuition of left-greedy paths as Pareto-optimal, an intuition which is entirely justified.

4 Topological bounds on Pareto optima

In this section we demonstrate a uniqueness result for Pareto optimal classes. The strategy of our proof is to show that any Pareto-optimal path class contains a representative which is Pareto-equivalent to a unique left-greedy cube path. For the remainder of the paper, $\mathcal{X}$ will denote a cylindrical roadmap coordination space. The cylindrical structure yields uniqueness results for left-greedy cube paths.

Proposition 1. *Any path between vertices of $\mathcal{X}^{(n)}$ is homotopic to a unique left-greedy cube path.*

Proof: Existence of a left-greedy path follows from Algorithm 1 and Lemma 4 following. The techniques of the proof for uniqueness are simple, but lie outside the scope of this paper. Briefly, one uses the cylindrical structure of $\mathcal{X}$ to prove that $\mathcal{X}^{(n)}$ is NONPOSITIVELY CURVED (see, *e.g.*, [11] for a definition). Then, a standard result from geometric group theory (Prop. 3.3 of [17]) implies that left-greedy paths are unique up to homotopy. □

Lemma 2. *Any L^∞ geodesic path on $\mathcal{X}$ can be continuously perturbed to an L^∞-close cube path in $\mathcal{X}^{(n)}$ for n sufficiently large.*

Proof: Given any L^∞ geodesic in $\mathcal{X}$, we begin by "pushing" it to an L^∞-approximate path α which lies completely in the interior of $\mathcal{X}$. This step relies crucially on the assumption that $\mathcal{X}$ is regularly closed, so that the interior is always accessible from the path.

We next approximate α by a cube path. Since the image of α is compact, there exists an $\epsilon > 0$ so that a tube of radius ϵ about the image of α does not intersect $\partial\mathcal{O}$. Choose n so that the grid size δ is less than ϵ. Modify α so that its endpoints are at vertices of $\mathcal{X}^{(n)}$. Consider the vertex $v^0 := \alpha(0)$ and the L^∞ balls $B_r := B_r^\infty(v^0)$ of radius r about v^0 in $\mathcal{X}^{(n)}$. Let $w^1 := \alpha \cap \partial B_\delta$ and $w^2 := \alpha \cap \partial B_{2\delta}$.

Let $F := B_\delta(v^0) \cap B_\delta(w^2)$. Clearly, $w^1 \in F$ and for any $x \in F$ it holds that the PL path $v^0 \to x \to w^2$ is of the same L^∞ length as the restriction of α to this neighborhood. We claim that F contains a vertex of the grid structure on $\mathcal{X}^{(n)}$. To see this, note that F is a subset of some face of B_δ defined by the set of all points $x = (x_i)$ such that

$$x_i \in [v_i^0 - \delta, v_i^0 + \delta] \cap [w_i^2 - \delta, w_i^2 + \delta] \quad ; \quad i = 1 \ldots N \tag{3}$$

Since these intervals have the same length, it follows that either $v_i^0 - \delta$ or $v_i^0 + \delta$ lies within the intersection for each i. Therefore, F contains a gridpoint v^1 of $\mathcal{X}^{(n)}$, and we may replace the segment of the path α with a PL path from v^0 to v^1 to w^2. Repeat the argument inductively, beginning at the vertex v^1 and considering the L^∞ balls about v^1 of radius δ and 2δ.

Marching along yields a PL path passing through vertices of $\mathcal{X}^{(n)}$ which are sequentially of L^∞-distance δ. This is therefore a cube path in $\mathcal{X}^{(n)}$. $\square$

Lemma 3. *Let α be a Pareto-optimal path on $\mathcal{X}$. One can find homotopic cube paths with endpoints close to those of α such that all the goal times approximate those of α.*

Proof: Apply Lemma 2 inductively and use the fact that the restriction of a Pareto-optimal path to any subpath is an L^∞ geodesic. $\square$

A simple algorithm, `LeftGreedy`, converts a cube path $\mathcal{C}$ into a left-greedy cube path by sweeping along $\mathcal{C}$ from left to right, comparing the star of each cube with its neighbor to the right (line 3), and shifting the common directions to the left cube (lines 4-5).

Lemma 4. *Given a Pareto-optimal cube path $\mathcal{C}$,* `LeftGreedy`$(\mathcal{C})$ *is a left-greedy cube path which is Pareto equivalent to $\mathcal{C}$.*

Proof: Since common directions are shifted to the left, each sweep through $\mathcal{C}$ for which $C_{i+1} \cap \mathsf{St}(C_i) \neq \{v_i\}$ for some i strictly decreases the positive integer-valued function $\sum_i i \cdot \dim(C_i)$; hence, convergence.

To argue that `LeftGreedy` can be executed through Pareto-equivalence, note first that the shifting process of lines 4-5 results in a homotopic path,

Algorithm 1 `LeftGreedy`($\mathcal{C}$)

Require: $\mathcal{C} = \{C_i\}_1^m$ is an L^∞-geodesic cube path
1: **while** $\mathcal{C}$ is not left-greedy **do**
2:     **for** $i = 1...m$ **do**
3:         $Z \leftarrow C_{i+1} \cap \mathsf{St}(C_i)$
4:         $C_{i+1} \leftarrow C_{i+1}/Z$
5:         $C_i \leftarrow C_i \times Z$
6:     **end for**
7: **end while**

since all modifications take place within $\mathsf{St}(C_i) \cup C_{i+1}$. If the j^{th} coordinate x_j reaches its goal at vertex v_i of $\mathcal{C}$, then all subsequent vertices v_k $(k \geq i)$ lie within the hyperplane of $\mathcal{X}$ having the same j^{th} coordinate. Hence, in line 3, the common factor Z is trivial in the j^{th} coordinate for v_i and beyond. Thus, the time-to-goal for x_j is unchanged by steps 4-5. □

We now may prove the Main Theorems:

Theorem 1. *Locally Pareto-optimal path classes on $\mathcal{X}$ are in bijective correspondence with homotopy classes of paths (fixing the endpoints).*

Proof: Assume there are two locally Pareto optimal paths, α and α', which are homotopic. Approximate these paths by homotopic cube paths $\mathcal{C}$ and $\mathcal{C}'$ on $\mathcal{X}^{(n)}$ for some large n via Lemma 3. Feed these cube paths to `LeftGreedy`. Via Lemma 4, the outputs are left-greedy cube paths which are Pareto equivalent to $\mathcal{C}$ and $\mathcal{C}'$ respectively. By Proposition 1, $\mathcal{C}$ and $\mathcal{C}'$ are the same path and hence have the same output times. Therefore, α and α' have output times which are arbitrarily close. □

It does not automatically follow that there is a bound on the number of Pareto optimal paths, since any cylindrical coordination space $\mathcal{X}$ which is not simply connected must have a countably infinite number of homotopy classes of paths. However, all but a finite set of these can be eliminated as global optima:

Theorem 2. *The number of globally Pareto optimal paths between fixed endpoints on $\mathcal{X}$ is finite.*

Proof: Fix endpoints and consider an infinite collection of Pareto optimal paths, all in different homotopy classes. We induct on K, the dimension of the smallest hyperplane (subset obtained by fixing certain coordinates) of $\mathcal{X}$ containing all the paths. The induction hypothesis is that for any such infinite collection of paths, there is an infinite subsequence of paths whose goal times $\{T_i^j\}$ satisfy $T_{i+1}^j \geq T_i^j + 1$ for $i = 1...\infty$ and $j = 1...N$. Assume that $K = 1$. Then, the subspace in which the paths live is a graph, and any infinite collection of homotopy classes must contain representatives which go around a cycle an arbitrarily high number of times, leading to a set of uniformly increasing goal times.

Assuming the hypothesis for systems in a hyperplane of dimension K, consider without loss of generality a system with $K+1$ degrees of freedom. Given an infinite number of Pareto optimal classes of paths $\{\gamma_i\}$, we may, by choosing a subsequence and reindexing, assume that T_i^1 converges to a constant T^1 as a non-increasing sequence and that $T_i^{K+1} \to \infty$. By compactness of the L^∞-ball of radius T^1, we have that the points $\gamma_i(T^1)$ converge to some point $x \in \mathcal{X}$. After reaching the first goal, the remainders of all the paths lie in a hyperplane of dimension K containing the point x.

We know by induction that, after restricting to a subsequence, the goal times T_i^j $(j = 2...K+1)$ must march to infinity at a rate bounded from below, since they all have starting point approximately at x. The only way for these paths to be Pareto optimal is to have the times T_i^1 be a *strictly* decreasing sequence. However, one can perform an arbitrarily small perturbation of the paths near x to switch to a low-i path after arriving at x. This makes all the other goal times shorter, contradicting the Pareto assumption. □

It follows easily from the compactness of $\mathcal{X}$ that there is a finite bound on the number of optima which is independent of the endpoints. This completes the proof of the Main Theorem.

5 Algorithms

We now consider the algorithmic problem of computing exact Pareto optimal paths. For concreteness, we restrict attention to coordination problems for N robots $\mathcal{R}_1, \ldots, \mathcal{R}_N$ in which each robot is modelled by a convex polygon translating along a PL path in $\mathbb{R}^2$. As a first step, we develop an algorithm for deforming a path to its Pareto-optimal representative. This is analogous to Algorithm 1 but operates directly on $\mathcal{X}$, rather than using the cube paths in some $\mathcal{X}^{(n)}$. This algorithm is presented in §5.1, followed by an implementation in §5.2 and a discussion in §5.3 of the complexity of computing the set of all Pareto optimal solutions.

5.1 Computing Pareto optimal paths

A path is MONOTONE if it is nondecreasing in each dimension. This is equivalent to the requirement that robots may not "back up" along their paths. For simplicity, we present our algorithm for the case of monotone paths and then outline the changes required to deal with non-monotone paths.

We know from §4 that it suffices to compute γ' to be the unique left greedy path homotopic to γ. Informally, left greedy implies that each robot should move at the fastest possible speed. Each speed change can be characterized in one of four ways: (1) stopping to avoid an immediate collision (when the

Algorithm 2 ExactLeftGreedy($\mathcal{X}, \gamma$)

Require: γ is a monotone collision-free coordination in $\mathcal{X}$.
$x_0 \leftarrow (0, \ldots, 0)$
$w \leftarrow \emptyset$
$\mathcal{O} \leftarrow$ ObstacleRegions($\mathcal{X}$)
$E \leftarrow$ CriticalEvents($\mathcal{O}, \gamma$)
$t \leftarrow 0$
while $x_t \neq x_{goal}$ **do**
  $t \leftarrow t + 1$
  $v \leftarrow$ MaximalLocallyCollisionFreeVelocity($x, w, \mathcal{O}$)
  $e \leftarrow$ NextEvent($x, v, E, \mathcal{O}$)
  $w \leftarrow$ UpdateWaitForSet(p, e, w)
  $x_t \leftarrow x_{t-1} + v \cdot e.\mathrm{time}$
end while
return $(x_0, \ldots, x_t)$

path reaches the boundary of a coordination space obstacle); (2) stopping to avoid a future collision (when a projection of the path reaches a horizontal or vertical extremum of some obstacle about which it may not go around); (3) restarting from case 2 when one of the obstacles has been successfully traversed; and (4) stopping when any robot reaches its goal.

The response to reaching a critical event depends on γ. The homotopy class of a monotone path is uniquely determined by assigning to each obstacle in each projection an orientation CW or CCW indicating the direction around which the path passes. As a result, if we build γ' in such a way that it induces the same orientation as γ on each obstacle, we can be certain that γ and γ' are homotopic. This is accomplished by stopping only at vertical critical events for obstacles with CW orientation and horizontal critical events for obstacles with CCW orientation.

Our algorithm builds a path by starting with a single point at $(0, \ldots, 0)$ and adding linear path segments sequentially until the goal is reached. We have already classified the critical events at which a robot may need to change speed in a left-greedy coordination. Thus, at every step, we need to compute the appropriate left-greedy velocity v, use v to determine which of these events will occur next, and add a segment to our coordination advancing to this point. We maintain as an invariant that w records the set of robots halted because of a critical event. This method is summarized in Algorithm 2; details follow. In our analysis, N denotes the number of robots, m the total complexity of the pairwise obstacle regions, and p the number of linear segments in γ. We consider N fixed.

The function ObstacleRegions computes the m obstacle regions using a simple collision detection method. In CriticalEvents, we can generate the critical events using a straightforward generalization of the standard vertical decomposition algorithm in [9] in time $O(m \log m)$. To assign an orientation to an obstacle $o \subseteq \mathcal{O}$, we need to determine whether o is above or below γ. Since γ divides each projection of $\mathcal{X}$ into two regions with PL boundaries, computing the orientation for each obstacle is a planar point location problem easily solved for all obstacles $O(p+m\log p)$ time using recursive triangulation methods [13]. Computing NEXTEVENT involves a ray-shooting query in each projection requiring total time $O(\sqrt{m}\log m)$ to execute the algorithm of [7].

Computing local left greedy gradients Given a point x in coordination space, we want to compute the maximal velocity $v = (v_1, \ldots, v_N)$ that does not result in a local collision. It is straightforward to show that v is the solution to a linear program in $O(N^2)$ constraints with maximization objective $\sum_i v_i$. For fixed N, such a program can be solved in constant time. In practice, the number of constraints is a function of the complexity of obstacle intersections and should be rather small.

Analysis Our algorithm is output sensitive in the sense that the run time is a function of the complexity p' of the locally Pareto optimal path generated. Each iteration of loop on lines 6-12 adds an additional segment to γ'. Therefore, Algorithm 2 executes in $O(p + m\log mp + p'\sqrt{m}\log m)$ time.

Dealing with nonmonotonicity We now relax the monotonicity requirement. The primary difficulty to be overcome is that obstacle orientations are no longer sufficient to determine a path's homotopy class. Indeed, one may revisit obstacles numerous times in opposite orientations.

To collate this data, extend rays upward and downward from each obstacle vertex that is a horizontal local extremum and do the same *mutatis mutandis* for vertical extrema. For each ray, assign a symbol a_i to represent the action of crossing this ray in the forward direction and let a_i^{-1} denote crossing the ray backward. The homotopy class of a path is completely determined by the sequence in which it crosses these critical events (extending our earlier usage of the term) and the direction of each crossing. That is, a path's homotopy class is uniquely determined by its representation in the language $L = \{a_i, a_i^{-1}\}^*$.

Conversely, if two paths are homotopic, then by continuously deforming one into the other, the crossing sequence will change by a simple transposition when the deformation passes an intersection point between a pair of critical events or when the order of successive events in different projections is reversed. Thus, we admit commutativity between symbols corresponding to events in different projections and to intersecting events in the same projection. We consider L with these relations as a group under word concatenation.

One can show that the left greedy path in each homotopy class will have a minimal-length crossing sequence in the group L. This dictates, for each direction in each projection, the order in which critical events should be crossed.

The straightforward method for doing word reduction in L is quadratic in the length of the word, which itself is $\Theta(pm)$.

In light of this analysis, it is a simple matter to extend Algorithm 2 to optimize non-monotone paths. Having computed the minimal crossing sequence from γ, we can build γ' segment-by-segment, enforcing the sequence of critical event crossings prescribed for each robot. Each robot should stop rather than cross any event other than the next prescribed in the sequence. At each step, the correct direction for each robot (*i.e.*, forward or backward) is determined by the direction required to reach the next event in the sequence for that robot. Aside from the additional preprocessing time required to do word-reduction in L, the complexity remains unchanged.

5.2 Simulations

We have implemented a simple version of the algorithm for monotone paths. In particular, we perform both the point-location step and the next event selection using the obvious quadratic time algorithms. Our implementation is in C++ on Linux and experiment times shown are for a 2.55GHz processor. Figure 3 illustrates a simple coordination problem with $n = 3$ for which we found the set of three Pareto optima by exhaustively enumerating monotone homotopy classes. These coordinations took approximately 0.3 seconds to compute. Figure 4 shows two more complex coordination simulations — one in which 8 robots translate left-to-right, and a 'swap' problem in which 20 agents switch sides.

5.3 Complexity of enumeration

Coordination of large numbers of agents under tight constraints is computationally challenging.

Example 4. Consider the generalization of Example 1 with $\Gamma_i = [-2, 2]$ for $i = 1...N$ and having N obstacles, each of the form $x_i^2 + x_j^2 < 1$ for $i < j$. The two Pareto optimal classes from Example 1 generalize (by the cubical symmetry of the space) to yield 2^{N-1} optimal classes between opposite corners in this coordination space.

This type of obstacle set — N intersecting cylinders — can be realized physically by an AGV system with N disc-like robots sliding along intervals, the collection of which have an N-fold intersection at their midpoints. The exponential complexity of this coordination problem is intuitively clear: since everyone cannot pass through the shared center at the same time, they must proceed through one at a time. There are a factorial number of ways so to do. On the other hand, for systems in which the cylindrical obstacles have bounded degrees of intersection in N, we expect the computational complexity to be polynomial in N.

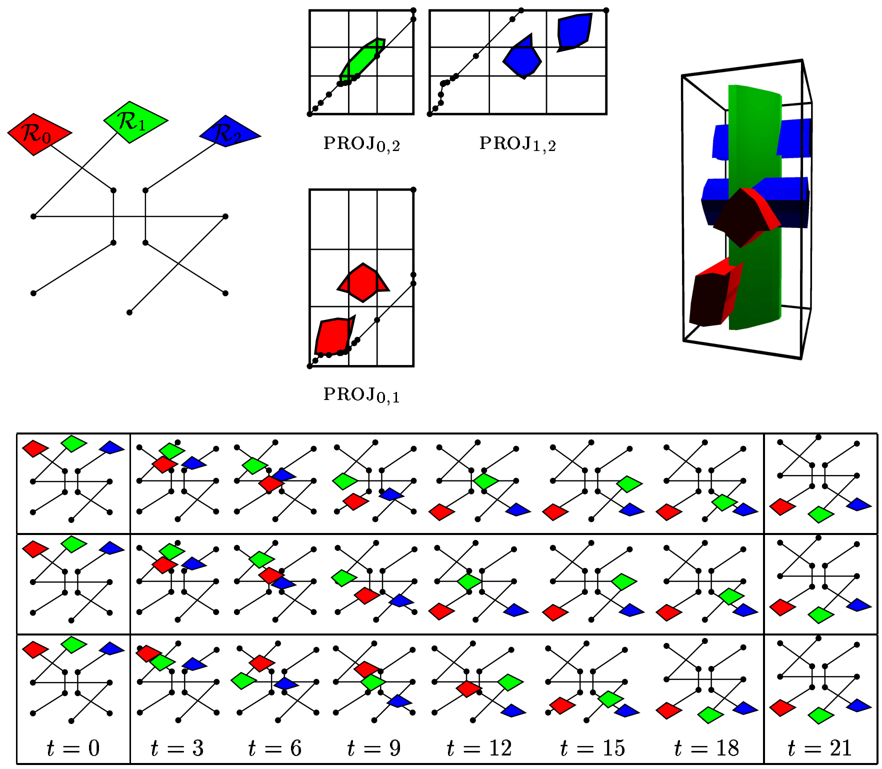

Fig. 3. [top] A coordination problem with 3 robots, its coordination space, and the projections thereof. [bottom] Snapshots of the three Pareto optima.

6 Discussion

It comes as no surprise that the enumeration of all Pareto optima is of high complexity: coordination is inherently difficult, and Lemma 1 implies that such an enumeration is akin to determining all minima for all monotone scalarizations of the cost functions. The algorithms we present are effective enough to be useful in factory AGV systems, where in practice the tracks have relatively low multiplicity of intersection.

What is a surprise is that the set of Pareto optima is finite, the crucial reason being that the cylindrical coordination spaces can be approximated by cubical complexes without any "concave" corners (more specifically, vertices of positive curvature). For such spaces, Pareto-optimal cube path classes are unique up to homotopy. This is not the case for general cube complexes. For example, if one approximates the spaces of Examples 2 and 3 by discretizations $\mathcal{X}^{(n)}$, then, as before, there are a finite number of optimal left-greedy cube paths. However, they are not unique up to homotopy, and the number

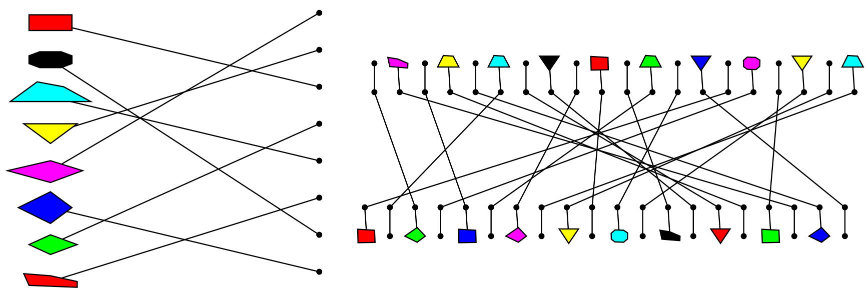

Fig. 4. Two complex coordination problems; [left] for 8 robots, the 37 optimal coordinations were computed in 171.6 seconds by an exhaustive search of monotone classes; [right] for 20 robots, a single coordination was computed in 1.53 seconds.

of such paths grows exponentially in the discretization step n, giving rise to the continuum of optima in the limit.

We note that the problem of computing Pareto optima is not dissimilar to that of computing Euclidean geodesics. It is well-known that computing shortest paths in 3-d is NP-hard in general [5]; we believe, based on our algorithm for Pareto optimization, that the geodesic problem is likewise simpler in the cylindrical case.

Among the several avenues for extending this work, we find the problem of characterizing the set of Pareto optima for general (non-cylindrical) coordination spaces to be the most fascinating. Working in the "cost space" (the image of τ), one must deal with non-discrete sets in the most general case. Do the set of τ-images have any nice structure in general?

References

1. M. D. Ardema and J. M. Skowronski. Dynamic game applied to coordination control of two arm robotic system. In R. P. Hämäläinen and H. K. Ehtamo, editors, *Differential Games - Developments in Modelling and Computation*, pages 118–130. Springer-Verlag, Berlin, 1991.
2. J. Barraquand and J.-C. Latombe. Robot motion planning: A distributed representation approach. *Int. J. Robot. Res.*, 10(6):628–649, December 1991.
3. Z. Bien and J. Lee. A minimum-time trajectory planning method for two robots. *IEEE Trans. Robot. & Autom.*, 8(3):414–418, June 1992.
4. S. J. Buckley. Fast motion planning for multiple moving robots. In *IEEE Int. Conf. Robot. & Autom.*, pages 322–326, 1989.
5. J. Canny and J. Reif. Lower bounds for shortest path and related problems. In *Proc. 28th Ann. IEEE Symp. Found. Comp. Sci.*, 49–60, 1987.
6. C. Chang, M. J. Chung, and B. H. Lee. Collision avoidance of two robot manipulators by minimum delay time. *IEEE Trans. Syst., Man, Cybern.*, 24(3):517–522, 1994.

7. B. Chazelle, H. Edelsbrunner, M. Grigni, L. J. Guibas, J. Hershberger, M. Sharir, and J. Snoeyink. Ray shooting in polygons using geodesic triangulations. *Algorithmica*, 12:54–68, 1994.
8. H. Chitsaz, J. M. O'Kane, and S. M. LaValle. Pareto-optimal coordination of two translating polygonal robots on an acyclic roadmap. In *Proc. IEEE International Conference on Robotics and Automation*, 2004. To appear.
9. M. de Berg, M. van Kreveld, M. Overmars, and O. Schwarzkopf. *Computational Geometry: Algorithms and Applications.* Springer, Berlin, 1997.
10. M. Erdmann and T. Lozano-Perez. On multiple moving objects. In *IEEE Int. Conf. Robot. & Autom.*, pages 1419–1424, 1986.
11. R. Ghrist. Shape complexes for metamorphic robot systems. In *Algorithmic Foundations of Robotics V, STAR 7*, pages 185–201, 2004.
12. H. Hu, M. Brady, and P. Probert. Coping with uncertainty in control and planning for a mobile robot. In *IEEE/RSJ Int. Workshop on Intelligent Robots and Systems*, pages 1025–1030, Osaka, Japan, November 1991.
13. D.G. Kirkpatrick. Optimal search in planar subdivisions. *SIAM J. Comput.*, 12(1):28–35, 1983.
14. S. M. LaValle and S. A. Hutchinson. Path selection and coordination of multiple robots via Nash equilibria. In *Proc. 1994 IEEE Int'l Conf. Robot. & and Autom.*, pages 1847–1852, May 1994.
15. S. M. LaValle and S. A. Hutchinson. Optimal motion planning for multiple robots having independent goals. *IEEE Trans. on Robotics and Automation*, 14(6):912–925, December 1998.
16. J. Miura and Y. Shirai. Planning of vision and motion for a mobile robot using a probabilistic model of uncertainty. In *IEEE/RSJ Int. Workshop on Intelligent Robots and Systems*, pages 403–408, Osaka, Japan, May 1991.
17. G. A. Niblo and L. D. Reeves. The geometry of cube complexes and the complexity of their fundamental groups. *Topology*, 37(3):621–633, 1998.
18. P. A. O'Donnell and T. Lozano-Pérez. Deadlock-free and collision-free coordination of two robot manipulators. In *IEEE Int. Conf. Robot. & Autom.*, pages 484–489, 1989.
19. L. E. Parker. Cooperative motion control for multi-target observation. In *IEEE/RSJ Int. Conf. on Intelligent Robots & Systems*, pages 1591–1598, 1998.
20. Y. Sawaragi, H. Nakayama, and T. Tanino. *Theory of Multiobjective Optimization.* Academic Press, New York, NY, 1985.
21. J. T. Schwartz and M. Sharir. On the piano movers' problem: III. Coordinating the motion of several independent bodies. *Int. J. Robot. Res.*, 2(3):97–140, 1983.
22. T. Simeon, S. Leroy, and J.-P. Laumond. Path coordination for multiple mobile robots: a resolution complete algorithm. *IEEE Trans. Robot. & Autom.*, 18(1), February 2002.
23. S.-H. Suh and K. G. Shin. A variational dynamic programming approach to robot-path planning with a distance-safety criterion. *IEEE Trans. Robot. & Autom.*, 4(3):334–349, June 1988.
24. P. Svestka and M. H. Overmars. Coordinated motion planning for multiple car-like robots using probabilistic roadmaps. In *IEEE Int. Conf. Robot. & Autom.*, pages 1631–1636, 1995.