Space-Aware Reconfiguration*

Dan Halperin[1], Marc van Kreveld[2], Golan Miglioli-Levy[1], and Micha Sharir[1]

[1] The Blavatnik School of Computer Science, Tel Aviv University, Israel
{danha@post.tau.ac.il,golanlevy@mail.tau.ac.il,michas@post.tau.ac.il}
[2] Dept. of Information and Computing Sciences, Utrecht University, the Netherlands
{M.J.vanKreveld@uu.nl}

Abstract. We consider the problem of *reconfiguring* a set of physical objects into a desired target configuration, a typical (sub)task in robotics and automation, arising in product assembly, packaging, stocking store shelves, and more. In this paper we address a variant, which we call *space-aware reconfiguration*, where the goal is to minimize the physical space needed for the reconfiguration, while obeying constraints on the allowable collision-free motions of the objects. Since for given start and target configurations, reconfiguration may be impossible, we translate the entire target configuration rigidly into a location that admits a valid sequence of moves, where each object moves in turn just once, along a straight line, from its starting to its target location, so that the physical space required by the start and the translated target configurations is minimized.

We investigate two variants of space-aware reconfiguration for the often examined setting of n *unit discs* in the plane, depending on whether the discs are distinguishable (labeled) or indistinguishable (unlabeled). For the labeled case, we compute, in $O(n^6)$ time, a shortest valid translation, or one that minimizes the enclosing disc or axis-aligned rectangle of both the start and target configurations. For the significantly harder unlabeled case, we show that for almost every direction, there exists a translation in this direction that makes the problem feasible. We use this to devise heuristic solutions, where we optimize the translation under stricter notions of feasibility. We present an implementation of such a heuristic, which solves unlabeled instances with hundreds of discs in seconds.

Keywords: Computational geometry, Motion planning, Disc reconfiguration, Smallest enclosing disc

* Work by D.H. and G.M.L. has been supported in part by the Israel Science Foundation Grants 825/15, 1736/19, by the Blavatnik Computer Science Research Fund, and by a grant from Yandex. Work by M.v.K. has been supported by the the Netherlands Organisation for Scientific Research under Grant 612.001.651. Work by M.S. has been supported by Grant 260/18 from the Israel Science Foundation, by Grant G-1367-407.6/2016 from the German-Israeli Foundation for Scientific Research and Development, and by the Blavatnik Computer Science Research Fund.

1 Introduction

Consider a set of n objects in the plane or in three-dimensional space and two configurations of these objects, a start configuration S and a target configuration T, where in each configuration the objects are pairwise interior disjoint. A typical *reconfiguration* problem asks to efficiently move the objects from S to T, subject to constraints on the allowable motions, the most notable of which is that all the moves be collision free.

In the specific problem studied in this paper, we are given n unit discs in the plane and we wish to move them from some start configuration to a target configuration. A valid move is a translation of one disc in a fixed direction from one placement to another without colliding with the other discs. The goal in earlier works on this problem was to minimize the number of moves, and the goal in the present study is to find an initial rigid translation of the discs of T, that minimizes the size of the physical space needed for the reconfiguration, under the constraint that each disc moves exactly once. This problem, like most problems in the domain of reconfiguration, comes in (at least) two flavors: *labeled* and *unlabeled*. In the labeled version, each object has a unique label, which marks its start placement and its unique target placement. In the unlabeled version the objects are indistinguishable, and we do not care which object finally gets to any specific target placement, as long as all the target placements are occupied at the end of the process; in particular all the objects are isothetic (as are the unit discs in our study). For the unlabeled case, without an initial shift of the target configuration, Abellanas et al. [1] have shown that $2n - 1$ moves are always sufficient. Dumitrescu and Jiang [7] have shown that $\lfloor 5n/3 \rfloor - 1$ moves are sometimes necessary, and that finding the minimum number of moves is NP-Hard. For the labeled case, Abellanas et al. [1] have shown that $2n$ moves are always sufficient and sometimes necessary. These are several examples of reconfiguration problems that have been studied in discrete and computational geometry; see, e.g., [3,4,5,6]. Varying the type of objects, the ambient space, the constraints on the motion and the optimization criteria, we get a wide range of problems, many of which are hard.

Similar problems arise in robotics. For example, such problems arise when a robot needs to arrange products on a shelf in a store, or when a robot needs to move objects around in order to access a specific product that needs to be picked up; see, e.g., [12,13,14]. In robotics, these problems are often referred to as object *rearrangement* problems. In this paper, though, we will stick to the term reconfiguration, which is also in common use.

Another prominent example from robotics and automation is the *assembly planning* problem (see, e.g., [11]), in which the target configuration of the objects comprises their positions in the desired product. The goal of assembly planning is to design a sequence of motions that will bring the parts together to form the desired product, and we want this sequence to be (collision-free and) optimal according to various criteria [8,9].

We address a certain criterion, which, to the best of our knowledge, has hardly been studied earlier: *minimizing the physical space* needed to carry out the desired assembly or reconfiguration. Abellanas et al. [1] did study a similar set of problems, in which the discs are placed inside different types of confined spaces. Their technique shows how to minimize the number of moves, given a prescribed size for a bounding rectangle of S and T. We adopt a different approach. We consider T a rigid configuration that can be placed anywhere in the workspace, and the goal is to find a placement for T for which there exists a feasible (collision-free) sequence of moves, where each disc moves *exactly once* along the straight segment that connects its start placement and to its target placement. The region occupied by S and by T in its translated location, together with the space required for the reconfiguration motion of all the objects, should be minimal according to various possible criteria. In this paper we consider the setup where we allow T only to be translated. We call this problem *space-aware reconfiguration*; we study it in this paper for the case of unit discs in the plane. Rigidly translating T into a different location in the plane ensures that the target objects maintain the same positional relations between them. This is a desired property in some reconfiguration problems, such as assembly planning. In this approach we do not care where the position of the final product is, as long as the space required for the reconfiguration is minimized. Moreover, as we will see, the variant where only translations of T are allowed is already quite difficult to solve. Tackling the general case, where we allow an arbitrary initial rigid motion of T, is left as a challenging open problem.

We say that a disc is placed at a point p if its center is placed at p. To avoid confusion between placeholder positions for discs (start or target) and the actual discs placed at these positions, we define a *valid configuration* P to be a finite set of points, such that every pair of points in the set lie at distance $\geqslant 2$ from one another, that is, we can place a unit disc at each point of P, so that the discs are pairwise interior disjoint. For any point p, we denote $D_r(p)$ as the disc centered at p with radius r. If r is not specified, then $D(p)$ is a unit disc ($r = 1$). For any valid configuration P, we denote $D(P) = \{D(p) \mid p \in P\}$.

Let S and T be two valid configurations, of n points each. We look for a sequence of n moves that bring the discs from S to T. A move consists of a single translation of one disc from $D(S)$ to a position in T, such that the disc does not collide with any other (stationary) disc on its way—neither with a disc in a start position, which has not been moved yet, nor with a disc that has already been moved to a target position. Each disc has to perform exactly one such move. We call such a sequence of moves an *itinerary.* We say that an itinerary is *valid* if all of its moves are collision-free. We denote such an Unlabeled Single Translation instance of the problem by UST(S,T), and a Labeled Single Translation instance by LST(S,T,M), where M is the matching between S and T induced by the labels; that is, each position in S is matched by M to the position in T with the same label. We call an instance of the problem *feasible* if it has a valid (collision-free) itinerary.

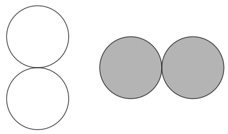


Fig. 1. An infeasible instance for the unlabeled (or labeled) version. The discs of $D(S)$ are drawn empty while the discs of $D(T)$ are drawn shaded.

It is easy to see (consider Figure 1) that even in the unlabeled version, this problem may not have a solution. If the shaded discs in the figure were placed higher (so that their centers were collinear with the center of the top empty disc, say), the problem would have been feasible. We now look for a vector $\vec{v}$ for which a valid itinerary exists from S to $T + \vec{v}$ (i.e., T translated by $\vec{v}$).[3] In the labeled case, the translated targets retain their labels after the translation. That is, if target position i was at the point t_i, the point $t_i + \vec{v}$ is now the ith target position. Observe that the initial location of T (when $\vec{v} = (0, 0)$) is now meaningless. From now on, we assume that the input location of T is placed to overlap with S as much as possible, e.g., S and T share their centers of mass or the centers of their smallest enclosing discs. A placement of this kind is ideal for the space-aware paradigm used in this work (although in practice it may not be valid).

In short, we look for a translation $\vec{v}$ such that (a) S and $T + \vec{v}$ admit a valid itinerary, and (b) some measure of 'nearness' of $T + \vec{v}$ to S is minimized. We denote these space-aware variants as SA-UST(S,T), and SA-LST(S,T,M), for the unlabeled and labeled variants, respectively.

There is a large body of related research on algorithms for multi-robot motion planning and multi-agent path finding; for recent reviews, see, e.g., [10] and [17] respectively. Notice however that in contrast to these more general problems, the focus of our work here (as in [1,3,4,6,7]) is on the special case where every object is transferred by a very small number (typically one or two) of simple atomic moves (e.g., translation along a segment). Therefore the planning techniques are of a rather different nature.

Contribution. We present, in Section 2, an algorithm for the labeled case that runs in $O(n^6)$ time, for constructing the space of all valid translations. We then show, in Section 3, how to find a valid translation that minimizes some measure of space-aware optimality, for example minimizing the area of the smallest enclosing disc of $D(S) \cup D(T + \vec{v})$. All the variants that we study can be solved by algorithms that run in $O(n^6)$ time; they actually run faster when the space of valid translations is already available.

The unlabeled case is much harder (see the full version of the paper for hardness results). We first show, in Section 4, that we can find a valid translation in almost any prescribed direction, if we translate T sufficiently far away. We study in Section 5.1 practical heuristic techniques that aim to find shorter valid translations, at the cost of further restricting the notion of validity, and in Section 5.2

[3] The translations from T to $T + \vec{v}$ do not count as moves. We often refer to it as *the initial translation.*

we present bounds on how well this heuristic approach works. Finally, in Section 5.3, we present experimental results of an implementation of the heuristic algorithm, for the more restrictive notions of validity, and show that it performs well in practice. Various technical details and additional comments are delegated to the full version of the paper (to be uploaded soon to the arXiv).

2 Labeled Version: Analysis of the Translation Plane

In this section we consider the labeled version LST of the problem. We are given two valid configurations S and T of n points each, and a one-to-one matching M between the positions of S and those of T, which is the set of pairs $\{(s, M(s)) \mid s \in S\}$, where s and $M(s)$ share the same label, for each $s \in S$. Our goal is to find a translation $\vec{v} \in \mathbb{R}^2$ such that there is a valid collision-free itinerary of n unit discs from S to $T + \vec{v}$ with respect to the matching M. That is, the goal is to define an ordering on the elements of M, denoted by $(s_1, M(s_1)), (s_2, M(s_2)), \ldots, (s_n, M(s_n))$, so that, for each $i = 1, \ldots, n$ in this order, we can translate the disc placed at s_i to the position $M(s_i) + \vec{v}$, so that it does not collide with any still unmoved discs, placed at $s_{i+1}, \ldots, s_n$, nor with any of the already translated discs, placed at $M(s_1) + \vec{v}, \ldots, M(s_{i-1}) + \vec{v}$.

We call a translation $\vec{v}$ a *valid translation* if it yields at least one valid itinerary. In the labeled version, which is easier to solve, we show how to compute the set of all valid translations in $O(n^6)$ time. We then present, in Section 3, three algorithms, each of which finds a valid translation $\vec{v}$ that minimizes a different measure of proximity between S and $T + \vec{v}$, as reviewed in the introduction.

We first address the subproblem in which $\vec{v}$ is fixed and our goal is to order M so as to obtain a valid itinerary, if at all possible. Let $A = (s, M(s))$ be a pair in the matching. For convenience, we denote s and $M(s)$ by A^S and A^T, respectively. Define the *hippodrome* of two unit discs D, D' to be the convex hull of their union. Observe that the hippodrome is exactly the area that a unit disc will cover while moving from D to D' along a straight trajectory. Define $H_{\vec{v}}(A)$ to be the hippodrome of $D(A^S)$ and $D(A^T + \vec{v})$. Denote by $k_{\vec{v}}$ the overall number of intersecting pairs of hippodromes $\{H_{\vec{v}}(A), H_{\vec{v}}(B)\}$, for all $A \neq B \in M$. See Figure 2 for an illustration.

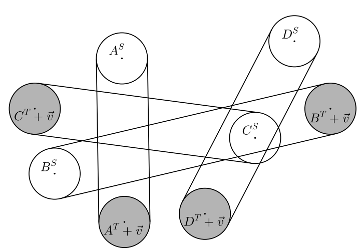


Fig. 2. The hippodromes $H_{\vec{v}}$ for four pairs $A, B, C, D \in M$ and some fixed $\vec{v}$. Notice that even though $H_{\vec{v}}(A) \cap H_{\vec{v}}(B) \neq \emptyset$, there is no restriction that the motion of A must precede or succeed the motion of B. Such restrictions do exist for many other pairs, such as B and C (C has to perform a motion before B).

Theorem 1 (Abellanas et al. [1]). *Let S and T be two valid configurations of n points each, and let $\vec{v}$ be a fixed translation. Let $M : S \to T$ be a bijection between the two configurations. Then one can compute, in $O(n \log n + k_{\vec{v}})$ time, a valid itinerary for S and $T + \vec{v}$ with respect to M, if one exists.*

We review the proof of the theorem, adapting it to our notations, and exploit later the ingredients of the analysis for the general problem (where we allow T to be translated). The constraints that the positions of the discs impose on the problem are as follows. We say that a pair $A = (A^S, A^T)$ (in M) has to perform a motion (of $D(A^S)$ to $D(A^T + \vec{v})$) before another pair B, for a given translation $\vec{v}$, if in any ordering Π of M that yields a valid itinerary, the index of A in Π is smaller than the index of B. In other words, for any two pairs $A, B \in M$, A has to perform a motion before B if either the disc $D(A^S)$ blocks the movement of $D(B^S)$ to the position $B^T + \vec{v}$, or the disc $D(B^T + \vec{v})$ blocks the movement of $D(A^S)$ to the position $A^T + \vec{v}$. Formally, we have:

Lemma 1. *Given pairs $A, B \in M$ and a fixed translation $\vec{v}$, A has to perform a motion before B (with respect to $\vec{v}$) if and only if at least one of the following conditions holds: (1) $D(A^S) \cap H_{\vec{v}}(B) \neq \emptyset$; (2) $D(B^T + \vec{v}) \cap H_{\vec{v}}(A) \neq \emptyset$.*

We next create a digraph whose vertices are the pairs of M, and whose edges are all the ordered pairs $(A, B) \in M^2$, for $A \neq B$, that satisfy (1) or (2). Borrowing a similar notion from assembly planning [11], we call the graph, for a fixed translation $\vec{v}$, the *translation blocking graph* (TBG), and denote it as $G_{\vec{v}}$. Denote the number of edges in $G_{\vec{v}}$ as $m_{\vec{v}}$, and observe that $m_{\vec{v}} \leqslant k_{\vec{v}}$. Indeed, for every edge $(A, B) \in G_{\vec{v}}$ the hippodromes $H_{\vec{v}}(A)$ and $H_{\vec{v}}(B)$ intersect, as is easily verified, but not every pair of intersecting hippodromes necessarily induce an edge; see the pairs A, B in Figure 2. As proved in [1], and as is easy to verify, the subproblem for a fixed $\vec{v}$ is feasible if and only if $G_{\vec{v}}$ is acyclic.

The circular arcs of a hippodrome can be split into two arcs, each of which is x-monotone. This allows us to construct $G_{\vec{v}}$ in $O(n \log n + k_{\vec{v}})$ time, by the sophisticated sweep-line algorithm of Balaban [2], which applies to any collection of well-behaved x-monotone arcs in the plane. (A standard sweeping algorithm would take $O(n \log n + k_{\vec{v}} \log n)$ time.) Checking whether $G_{\vec{v}}$ is acyclic, and, if so, performing topological sorting on $G_{\vec{v}}$, takes $O(n + m_{\vec{v}})$ time. By definition, any topological order of the vertices of $G_{\vec{v}}$, that is of M, defines a valid itinerary. If $G_{\vec{v}}$ has cycles, no valid itinerary exists for $\vec{v}$.

We now consider the translation plane $\mathbb{R}^2$, each of whose points corresponds to a translation vector $\vec{v}$. We say that a point $\vec{v}$ in the translation plane is valid, if the corresponding translation vector is valid, i.e., admits a valid itinerary from S to $T+\vec{v}$. Our goal is to construct the region Q of all the valid points (translations), and to partition Q into maximal connected cells, so that all translations in the same cell have the same TBG. Thus, for each cell, either all its points are valid (with the same set of common valid itineraries) or all its points are invalid.

Remark. For some instances, Q is empty, as in the scenario depicted in Figure 3. In that case, our algorithms will report that no valid translation exists. We also remark that tangency is not a necessary characteristic of infeasible instances, as we will shortly show.

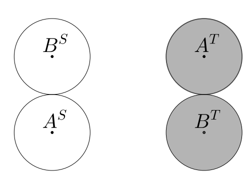


Fig. 3. No valid itinerary exists between S and $T+\vec{v}$, for any translation $\vec{v}$.

We first fix two pairs $A, B \in M$, and consider the region $\mathcal{V}_{AB}$, which is the locus of those $\vec{v}$ for which the (directed) edge AB is present in $G_{\vec{v}}$. We can write $\mathcal{V}_{AB} = \mathcal{V}_{AB}^{(1)} \cup \mathcal{V}_{AB}^{(2)}$, where $\mathcal{V}_{AB}^{(1)}$ (resp., $\mathcal{V}_{AB}^{(2)}$) is the locus of all $\vec{v}$ for which condition (1) (resp., (2)) in Lemma 1 holds. We thus have

$$\mathcal{V}_{AB}^{(1)} = \{\vec{v} \in \mathbb{R}^2 \mid D(A^S) \cap H_{\vec{v}}(B) \neq \emptyset\},$$
$$\mathcal{V}_{AB}^{(2)} = \{\vec{v} \in \mathbb{R}^2 \mid D(B^T + \vec{v}) \cap H_{\vec{v}}(A) \neq \emptyset\}.$$

We call $\mathcal{V}_{AB}^{(1)}$ (resp., $\mathcal{V}_{AB}^{(2)}$) the *vippodrome* of (A, B) of the first (resp., second) type.

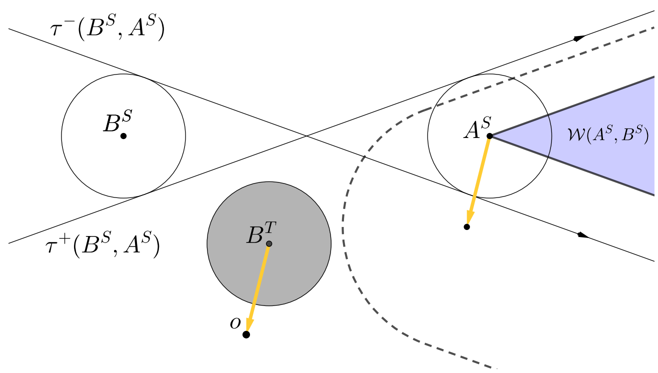


Fig. 4. The vippodrome $\mathcal{V}_{AB}^{(1)}$, which is the region to the right of the dashed curve in the translation plane. The wedge $\mathcal{W}(A^S, B^S)$ is colored in blue. The vippodrome is obtained by first expanding $\mathcal{W}(A^S, B^S)$ by $D_2(o)$, and then by shifting by the vector $-B^T$ (in orange).

To construct $\mathcal{V}_{AB}^{(1)}$, we proceed as follows; see Figure 4. For given pairs $A, B \in M$, consider the two inner tangent lines, $\tau^-(B^S, A^S)$ and $\tau^+(B^S, A^S)$, to $D(B^S)$ and $D(A^S)$, and assume that they are both directed from B^S to A^S, so that B^S lies to the right of $\tau^-(B^S, A^S)$ and to the left of $\tau^+(B^S, A^S)$, and A^S lies to the left of $\tau^-(B^S, A^S)$ and to the right of $\tau^+(B^S, A^S)$. Let $\mathcal{W}(A^S, B^S)$ denote the wedge whose apex is at A^S and whose rays are parallel to (and directed in the same direction as) $\tau^-(B^S, A^S)$ and $\tau^+(B^S, A^S)$. Denote the origin as o. We then have the following representation.

Lemma 2.

$$\mathcal{V}_{AB}^{(1)} = \mathcal{W}(A^S, B^S) \oplus D_2(o) - B^T = (\mathcal{W}(A^S, B^S) - B^T) \oplus D_2(o) \quad (1)$$
$$\mathcal{V}_{AB}^{(2)} = -\mathcal{W}(B^T, A^T) \oplus D_2(o) + A^S = -(\mathcal{W}(B^T, A^T) - A^S) \oplus D_2(o).$$

Proof. $\mathcal{V}_{AB}^{(1)}$ is the locus of all translations $\vec{v}$ at which $D(A^S)$ intersects $H_{\vec{v}}(B)$. Equivalently, $\mathcal{V}_{AB}^{(1)}$ is the locus of all translations $\vec{v}$ at which $D_2(A^S)$ intersects the segment $e_{\vec{v}} = (B^S,\ B^T + \vec{v})$. The boundary of $\mathcal{V}_{AB}^{(1)}$ thus consists of all translations $\vec{v}$ for which either $e_{\vec{v}}$ is tangent to $D_2(A^S)$ or $B^T + \vec{v}$ touches $D_2(A^S)$. That is, $\partial\mathcal{V}_{AB}^{(1)}$ consists of all translations $\vec{v}$ for which $B^T + \vec{v}$ lies on the boundary of $\mathcal{W}(A^S, B^S) \oplus D_2(o)$, from which the claim easily follows. The claim for $\mathcal{V}_{AB}^{(2)}$ follows by a symmetric argument, switching between S and T and reversing the direction of the translation. □

Note that the boundary of a vippodrome $\mathcal{V}_{AB}^{(1)}$ is the smooth concatenation of two rays and a circular arc, where the rays are parallel to the rays of $\mathcal{W}(A^S, B^S)$, and where the arc is an arc of the disc $D_2(A^S - B^T)$, of central angle $\pi - \theta$, where θ is the angle of $\mathcal{W}(A^S, B^S)$. The same holds for $\mathcal{V}_{AB}^{(2)}$, with the same disc $D_2(A^S - B^T)$. Hence the boundaries of $\mathcal{V}_{AB}^{(1)}$ and of $\mathcal{V}_{AB}^{(2)}$ (more precisely, the circular portions of these boundaries) might overlap. See Figure 5 for an illustration.

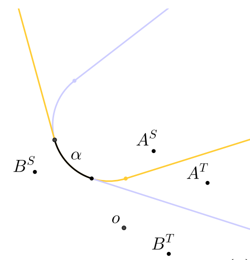


Fig. 5. The vippodromes $\mathcal{V}_{AB}^{(1)}$ and $\mathcal{V}_{AB}^{(2)}$, colored in blue and orange, respectively. The arc α is the overlap portion of the two vippodrome boundaries.

Using vippodromes, we can give, as promised, a scenario of a start and target configurations that admit no valid translation even though the discs of each configuration do not touch each other. See Figure 6 for an illustration.

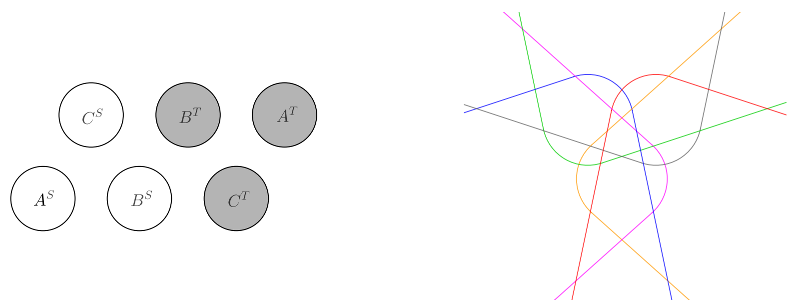


Fig. 6. A scenario where no valid translation exists. Left: The start and target configurations. Right: All the 12 vippodromes are drawn (in the translation plane). Observe that the vippodromes $\mathcal{V}_{AB}^{(2)}$ and $\mathcal{V}_{BA}^{(1)}$ coincide (their boundaries are drawn in orange), and so does every other similar pair of vippodromes for the other pairs of positions. Each point in the translation plane is covered by at least two vippodromes of contradicting constraints (e.g., for points inside the orange vippodrome, A has to perform a motion before B and vice versa), and so every translation is invalid (its TBG contains a 2-cycle).

Let $\mathcal{V}^\partial = \{\partial\mathcal{V}_{AB}^{(i)} \mid i \in \{1,2\}, A \neq B \in M\}$ and observe that $|\mathcal{V}^\partial| = 2n(n-1)$. Define the vippodrome arrangement $\mathcal{A}(\mathcal{V}^\partial)$, induced by M, to be the arrangement formed by the curves of $\mathcal{V}^\partial$; it is an arrangement of $O(n^2)$ rays and circular arcs. Assuming general position, the only overlaps between features of the arrangement are between circular arcs of the two vippodromes of the same ordered pair A, B (see Figure 5 again). To avoid these overlaps, we partition each of these arcs into two subarcs at the point where the overlap begins or ends (so each circle $\partial D_2(A^S - B^T)$ contributes at most three arcs to the arrangement). In the rest of the paper, we assume general position of pairs of unrelated vippodromes, to simplify the presentation. We do, however, allow tangency between discs, which will require some special extra consideration.

We now observe that any pair of features (rays and circular arcs) of the (modified) arrangement intersect at most twice. Hence the number of vertices in $\mathcal{A}(\mathcal{V}^\partial)$ is at most $O(n^4)$, and so the overall complexity of the arrangement is also $O(n^4)$. Consider a face f of $\mathcal{A}(\mathcal{V}^\partial)$. We showed that for every ordered pair of pairs $(A, B) \in M^2$, $A \neq B$, the edge AB is either in every graph $G_{\vec{v}}$, for $\vec{v} \in f$, or in none of these graphs. Hence all the graphs $G_{\vec{v}}$, for $\vec{v} \in f$, are identical, and we denote this common graph as G_f.

We can construct $\mathcal{A}(\mathcal{V}^\partial)$ either in $O(n^4 \log n)$ time using a plane-sweep procedure, or in $O(n^2\lambda_4(n^2))$ time [4], using the incremental procedure described in [16, Theorem 6.21, p. 172].

After $\mathcal{A}(\mathcal{V}^\partial)$ has been constructed, we traverse its faces and construct the graphs G_f, over all faces f, noting that when we cross from a face f to an adjacent face f', the graph changes by the insertion or deletion of a single edge.[5] We then test each graph G_f for acyclicity. The union of (the closure of) all the faces f for which G_f is acyclic is the desired region Q of valid translations. For each face f that participates in Q, we run a linear-time procedure for topological sorting of G_f, and the order that we obtain[6] defines a valid itinerary for all translations $\vec{v} \in f$. The running time, for a fixed face f, is $O(n + m_f)$, where m_f is the number of edges of G_f. In the worst case we have $m_f = O(n^2)$, so the overall cost of the algorithm is $O(n^6)$. We have thus obtained the following main result of this section.

Theorem 2. *Given a labeled instance* LST(S, T, M) *of the reconfiguration problem, with a valid start configuration S, a valid target configuration T, of n points each, and a matching M between S and T, we can compute the region of all valid translations for T in $O(n^6)$ time.*

Remark. It would be interesting to see whether dynamic algorithms for maintaining acyclicity in a directed graph, under insertions and deletions of edges (namely, fully dynamic cycle detection algorithms), could be applicable when we traverse the faces of $\mathcal{A}(\mathcal{V}^\partial)$. Such algorithms can be found in [15], but they do not seem to improve the asymptotic running time of our algorithm.

[4] Here $\lambda_s(m)$ denotes the maximum length of a Davenport-Schinzel sequence of order s on m symbols; see [16] for details.

[5] As already mentioned, although we assume general position, circular arcs of vippodromes may still overlap (see Figure 5). Notice, however, that the overlapping arcs bound vippodromes that induce the same constraint on the itinerary and hence crossing the overlapping arcs still incurs insertion or deletion of a single edge.

[6] In general, G_f can have exponentially many topological orders, each of which yields a valid itinerary.

3 Labeled Version: Space-Aware Optimization

We examine three different variants of the optimization criterion for the SA-LST problem: SA-LST$_{|\vec{v}|}(S,T,M)$, for minimizing the length of the translation vector $\vec{v}$; SA-LST$_{\mathsf{AABR}}(S,T,M)$, for minimizing the area of the axis-aligned bounding rectangle of (the union of the discs of) $D(S) \cup D(T + \vec{v})$, denoted as $\mathsf{AABR}(D(S) \cup D(T + \vec{v}))$; and SA-LST$_{\mathsf{SED}}(S,T,M)$, for minimizing the area of the smallest enclosing disc of (the union of the discs of) $D(S) \cup D(T+\vec{v})$, denoted as $\mathsf{SED}(D(S) \cup D(T + \vec{v}))$. In all three variants, we assume that the space Q of valid translations has already been computed, by the algorithm of Theorem 2, in $O(n^6)$ time. As we show, once Q is available, the cost of the optimization procedures is smaller.

We review here some key ideas in our solution of the SA-LST$_{\mathsf{SED}}(S,T,M)$ problem, which is the most involved instance among our space-optimization problems. (The solution of SA-LST$_{|\vec{v}|}(S,T,M)$ and of SA-LST$_{\mathsf{AABR}}(S,T,M)$, as well as the full details of the solution of SA-LST$_{\mathsf{SED}}(S,T,M)$, appear in the full version of the paper.) We denote the smallest enclosing disc of a set P as $\mathsf{SED}(P)$, and its radius as $r(P)$. Our goal is to minimize $r(S \cup (T + \vec{v}))$,[7] over all valid translations $\vec{v} \in Q$. Consider the farthest-neighbor Voronoi diagrams $\mathsf{FVD}(S)$ of S and $\mathsf{FVD}(T)$ of T. Note that $\mathsf{FVD}(T + \vec{v}) = \mathsf{FVD}(T) + \vec{v}$. If we fix $\vec{v}$, the smallest enclosing disc $D = \mathsf{SED}(S \cup (T + \vec{v}))$ is centered either at a Voronoi vertex ξ of $\mathsf{FVD}(S \cup (T + \vec{v}))$, in which case ∂D passes through the three points that lie farthest from (i.e., define) ξ, or at a Voronoi edge e of this diagram, in which case D is the diametral disc formed by the two points that lie farthest from (i.e., define) e. In the former case either (i) the three farthest points belong to the same set (S or $T + \vec{v}$), or (ii) two points belong to one set and the third belongs to the other set. In the latter case, either (iii) the two farthest points belong to the same set, or (iv) they belong to different sets. For each case, we collect every disc D that fits our requirements ($\vec{v}$ is a valid translation and D is centered at a vertex or an edge of $\mathsf{FVD}(S \cup (T + \vec{v}))$), as described above. We then output $\mathsf{SED}(S \cup (T + \vec{v}))$ as the disc with the smallest radius among the candidate discs.

In Case (i) we have $O(n)$ candidates for the center of $\mathsf{SED}(S \cup (T + \vec{v}))$, each of which is either a stationary vertex of $\mathsf{FVD}(S)$ or a vertex of $\mathsf{FVD}(T)$ shifted by $\vec{v}$. By the symmetry of the setup, it suffices to focus, without loss of generality, on vertices of $\mathsf{FVD}(S)$. Let ξ be such a vertex. Let $D(\xi)$ be the smallest disc such that $S \subset D(\xi)$ (its radius is determined by the distance to the farthest neighbor(s) of ξ in S). In order for $D(\xi)$ to contain $T + \vec{v}$ for some translation $\vec{v}$, ξ has to lie in the intersection of all the discs of radius $r(D(\xi))$ centered at the points of $T + \vec{v}$. Observe that this region of translations, denoted as $V(\xi)$, is the intersection $\bigcap_{t \in T}(D(\xi) - t)$. We thus construct $V(\xi)$ and overlay it with

[7] This is indeed an equivalent formulation to the one given earlier: The smallest enclosing disc of $D(S) \cup D(T + \vec{v})$ has the same center as the disc that we find, and its radius is larger by 1.

the valid portion Q of the arrangement $\mathcal{A}(\mathcal{V}^{\partial})$ of the vippodrome boundaries. It is then easy to find, in time proportional to the complexity of the overlay, a valid translation $\vec{v}$ such that $T + \vec{v} \subset D(\xi)$, namely a translation in $Q \cap V(\xi)$, if one exists. Since the complexity of the overlay is still $O(n^4)$ (its new vertices are intersections of $O(n)$ edges of $V(\xi)$ with $O(n^2)$ vippodrome boundaries, and there are only $O(n^3)$ such intersections), this takes $O(n^4)$ time. Multiplying by the number of vertices ξ, we get a total of $O(n^5)$ time in this case.

The other cases (ii)–(iv) are more involved. Again, for lack of space, further details appear in the full version of the paper. Overall, we obtain the following theorem (split in the full version into three separate theorems, one theorem for each problem).

Theorem 3. *Once Q has been computed, the space-aware labeled problems* SA-LST$_{|\vec{v}|}(S,T,M)$*,* SA-LST$_{\mathsf{AABR}}(S,T,M)$*, and* SA-LST$_{\mathsf{SED}}(S,T,M)$*, can be solved in $O(n^2 \log n)$, $O(n^2 \log n)$, and $O(n^5 \log n)$ time, respectively.*

4 Unlabeled Version: Preliminary Analysis

In this section we study the reconfiguration problem for unlabeled discs. The main result of the section is summarized the following theorem.

Theorem 4. *Let S and T be two valid configurations, of n points each. For every direction δ, except possibly for finitely many directions, there exists a translation $\vec{v} \in \mathbb{R}^2$ in direction δ such that the unlabeled problem* UST*$(S,T + \vec{v})$ is feasible.*

Proof. Let C be a valid configuration of n points in the plane. Let c, c' be two points in C and let $b(c, c')$ denote their perpendicular bisector. Put $\mathcal{B}(C) = \{b(c, c') \mid c, c' \in C, dist(c, c') = 2\}$, which is the set of all perpendicular bisectors (common inner tangents) of any pair of touching discs of $D(C)$. We say that a direction is *generic* for C if it is not parallel to any line in $\mathcal{B}(C)$. (Note that, by Euler's formula for planar maps, there are only $O(n)$ non-generic directions.) We fix a generic direction δ for both S and T. Observe that δ is also generic for $T+\vec{v}$, for any vector $\vec{v}$. Without loss of generality, assume that δ is horizontal and points to the right. We define $\Pi_\delta(C)$ to be the reverse lexicographical order of the points in C, that is, $\Pi_\delta(C) = (c_1, c_2, \ldots, c_n)$, so that, for any $1 \leqslant i < j \leqslant n$, c_i is to the right of (or at the same x-coordinate but above) c_j. We now fix a matching M_δ according to the orders $\Pi_\delta(S)$ and $\Pi_\delta(T)$, by aligning both orders, i.e., $M_\delta(s_i) = t_i$, for $i = 1, \ldots, n$. The matching M_δ transforms the problem to the labeled version LST(S,T,M_δ). We claim that this specific instance is always feasible, and, moreover, admits valid translations in direction δ. Order M_δ in the same order of $\Pi_\delta(S)$ and $\Pi_\delta(T)$, i.e., $(s_1, t_1), \ldots, (s_n, t_n)$, and denote this order as $\Pi(M_\delta)$. We claim that one can always choose $\vec{v}$, in direction δ, such that $(s_1, t_1 + \vec{v}), \ldots, (s_n, t_n + \vec{v})$ is a valid itinerary.

We apply a simpler variant of the techniques developed in Section 2. Since we have already assigned the fixed order $\Pi(M_\delta)$ to M_δ, we do not need to take into consideration all the vippodromes, but only the ones that impose constraints that violate $\Pi(M_\delta)$. Let A_i be the pair $(s_i, t_i) \in M_\delta$ (so $D(s_i)$ is the disc that moves at step i). Let

$$\mathcal{V}_{\mathrm{bad}}(\delta) = \{\mathcal{V}^{(i)}_{A_k A_l} \mid i \in \{1,2\}, A_k, A_l \in M_\delta, k > l\},$$

and observe that $|\mathcal{V}_{\mathrm{bad}}(\delta)| = n(n-1)$. In other words, $\mathcal{V}_{\mathrm{bad}}(\delta)$ is the subset of all the vippodromes V, such that, for each $\vec{v} \in V$, the constraint that V represents violates the itinerary according to $\Pi(M_\delta)$ between S and $T + \vec{v}$. Thus, in order to find a valid translation $\vec{v}$ in direction δ, it suffices to show that the ray ρ from the origin in direction δ (the positive x-axis by assumption) is not fully contained in the union of the vippodromes in $\mathcal{V}_{\mathrm{bad}}(\delta)$.

We claim that there exists a ray $\rho' \subseteq \rho$ such that $\rho' \cap V = \varnothing$ for every $V \in \mathcal{V}_{\mathrm{bad}}(\delta)$. Indeed, let $V = \mathcal{V}^{(1)}_{BA}$, such that $A, B \in M_\delta$ and A performs a motion before B according to $\Pi(M_\delta)$; that is, $V \in \mathcal{V}_{\mathrm{bad}}(\delta)$. Let $\tau^-(A^S, B^S), \tau^+(A^S, B^S)$ be the rays of $\mathcal{W}(B^S, A^S)$ (recall the setup discussed in Section 2, depicted in Figure 4, and note that here A and B change roles). By construction, if A performs a motion before B according to the itinerary, then A^S is lexicographically larger than B^S, and so B^S is to the left of (or has the same x-coordinate and is below) A^S. Since the positive x-direction δ is generic, $D(A^S)$ and $D(B^S)$ cannot lie vertically above one another and have a common inner tangent. Let σ be the ray that emanates from B^S in the direction from A^S to B^S. By our assumption, σ points either directly downwards, or else to the left (contained in the open left vertical halfplane that contains B^S on its right boundary). Note that σ is the mid-ray of the wedge $\mathcal{W}(B^S, A^S)$ (see Figure 4). In the former case, the opening angle of $\mathcal{W}(B^S, A^S)$ is strictly smaller than π, and in the latter case, it is at most π. In either case, $\mathcal{W}(B^S, A^S)$ is disjoint from the rightward-directed horizontal ray from B^S (the ray in direction δ). This implies that $\mathcal{W}(B^S, A^S)$ cannot fully contain any rightward-directed ray. Since $V = \mathcal{W}(B^S, A^S) \oplus D_2(o) - A^T$, the same claims hold for V as well. The argument for $\mathcal{V}^{(2)}_{BA}$ is similar. In conclusion, ρ must exit from every vippodrome of $\mathcal{V}_{\mathrm{bad}}(\delta)$, which establishes the claim.

Hence, there are infinitely many translations $\vec{v}$ in ρ that do not belong to any vippodrome $V \in \mathcal{V}_{\mathrm{bad}}(\delta)$. By construction, this implies that UST(S, $T + \vec{v}$) is feasible for every such $\vec{v}$ (with the valid itinerary induced by $\Pi(M_\delta)$). Furthermore, the above holds for every generic direction δ. This completes the proof of the theorem. □

It is now fairly simple to devise an algorithm for finding a valid translation $\vec{v}$ and for constructing a valid itinerary from S to $T + \vec{v}$. First, choose a generic direction δ, in $O(n \log n)$ time, and assume it to point in the positive x-direction. Calculate $\Pi_\delta(S), \Pi_\delta(T)$ and M_δ in $O(n \log n)$ time. Construct $\mathcal{V}_{\mathrm{bad}}(\delta)$ according to M_δ, in $O(n^2)$ time. Intersect all the vippodromes of $\mathcal{V}_{\mathrm{bad}}(\delta)$ with the positive x-axis, and find the rightmost intersection point $\vec{v}_{\max}$ with these vippodromes, which can be done in $O(n^2)$ time. Any translation $\vec{v}$ to the right of $\vec{v}_{\max}$ has a valid itinerary from S to $T + \vec{v}$, given by the order $\Pi(M_\delta)$. The overall running time of this algorithm is therefore $O(n^2)$.

5 Unlabeled Version: Space-Aware Practical Solutions

5.1 Heuristics for Short Valid Translations

The analysis in Section 4, while providing an abundance of valid translations, has the disadvantage that the valid translations that it yields are potentially too long (one needs to go sufficiently far away in the δ-direction to get out of all the 'bad' vippodromes). This is undesirable with our space-aware objective in mind, where we seek short valid translations. In this subsection, we provide a heuristic for finding shorter valid translations for the unlabeled variant, thereby obtaining shorter heuristic solutions to SA-UST$_{|\vec{v}|}(S,T)$. For more heuristics, also for SA-UST$_{\mathsf{AABR}}(S,T)$ and SA-UST$_{\mathsf{SED}}(S,T)$, see the full version of the paper. The resulting algorithms are faster than those obtained for the labeled case (at the cost of not guaranteeing optimality). As the unlabeled problem is much harder than the labeled case (and we believe it to be NP-hard), we make no attempt at solving it exactly.

In the next subsection we will show how to choose a good direction δ for which we can give reasonable upper bounds on the length of the shortest valid translation, or of the valid translation $\vec{v}$ that minimizes the smallest enclosing disc of $S \cup (T + \vec{v})$. For now, fix a generic direction δ for S and T, and assume, for simplicity and with no loss of generality, that it is the positive x-direction. In practice, one might want to choose a sufficiently dense set of generic directions, in the hope of improving the quality of the following solutions.

We can modify the algorithm of Section 4 so that it returns the valid translation in direction δ that is closest to the origin. To do so, we construct all the bad vippodromes, intersect their boundaries with the δ-directed ray, and sort them along the ray, from which the shortest valid translation along the ray is easily obtained. The cost of this improvement is $O(n^2 \log n)$ time.

5.2 Bounding the Heuristic Solutions

The analysis in the preceding subsection provides heuristics for obtaining short valid translations, but gives no guarantees on how well we approximate the optimal valid translation. In this subsection we show how to choose a good direction δ for which we can give a reasonable (and sometimes asymptotically optimal) upper bound on the length of the shortest valid translation in direction δ, or of a valid translation $\vec{v}$ in direction δ that minimizes $\mathsf{SED}(S \cup (T + \vec{v}))$. [8] Recall that we denote by $r(P)$ the radius of the smallest enclosing disc of a set of points P. Our main result (see the full version of the paper for its proof) is:

Theorem 5. *Let S and T be two valid configurations, of n points each, such that S and T share the centers of their smallest enclosing discs. There exists a translation $\vec{v}$ such that* $\mathrm{UST}(S, T + \vec{v})$ *is feasible and* $|\vec{v}| = O((r(S) + r(T))n)$. *The same asymptotic bound also applies to* $r(S \cup (T + \vec{v}))$.

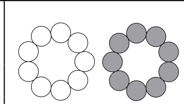
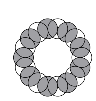
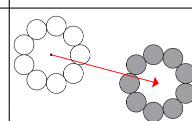
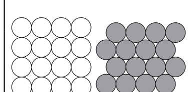
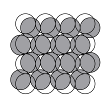
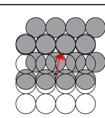
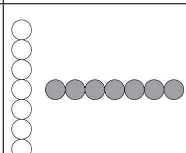
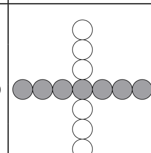
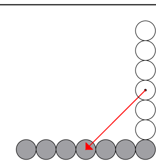
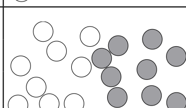
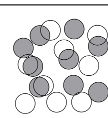
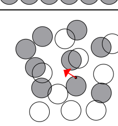
[8] Similar techniques can be applied to the variant of optimizing the area of the axis-aligned bounding rectangle.

The physical space needed for the reconfiguration is worse by the factor $O(n)$, compared to the ideal bound $O(r(S) + r(T))$, which is (asymptotically) the minimum value of $r(S \cup (T + \vec{v}))$, over all translations $\vec{v}$. Interestingly, we can attain this bound if the discs of $D(S)$, as well as the discs of $D(T)$, are sufficiently separated. That is, we have (see the full version of the paper for the proof):

Theorem 6. *Let S and T be two valid configurations, of n points each, such that S and T share the centers of their smallest enclosing discs. Assume that there exists a fixed constant $\varepsilon > 0$, so that the distance between any pair of points in S, or any pair of points in T, is at least $2+\varepsilon$. Then, for* any *direction δ, there exists a translation $\vec{v}$ in direction δ, such that* $\mathrm{UST}(S, T+\vec{v})$ *is feasible and $|\vec{v}| = O((r(S)+r(T))/\sqrt{\varepsilon})$. The same asymptotic bound also holds for $r(S \cup (T+\vec{v}))$.*

Note that Theorem 6 is stronger than Theorem 5 also in that it holds for *every* direction δ, whereas Theorem 5 only holds for restricted values of δ.

Table 1. Different input types to UST. For each input type, the configurations are first presented separated, for better visualization, then in their initial positions (sharing the centers of their smallest enclosing discs) and with T translated according to an approximate shortest valid translation (in red), produced by our heuristic algorithm.

Conf.	Sources/Targets	Initial	Translated	n	$r(S) + r(T)$	$\lvert\vec{v}\rvert$
Circle				100	65.67	190.19
				200	129.32	376.24
				500	320.31	913.79
				1,000	638.60	1,757.26
Packing[9]				100	27.01	5.53
				210	38.88	2.18
				506	60.76	3.25
				1,024	87.22	19.75
Cross				100	200	140.07
				200	400	281.43
				500	1,000	706.15
				1,000	2,000	1,413.47
Random[10]				100	36.78	16.96
				200	51.70	34.01
				500	82.26	78.49
				1,000	116.06	147.61

[9] The numbers n are chosen so that the size of the source configuration will be a square, or close to a square.

[10] The results are averaged over 10 different instances, one of which is depicted.

5.3 Implementation of the Heuristic Algorithm

We implemented the heuristic algorithm for finding an approximate shortest valid translation for UST instances, as outlined in Section 5.1. Our program is written in Python 3.7, and the experiments that we report below were carried out on an Intel Core i7-7500U CPU clocked at 2.9 GHz with 24 GB of RAM.

Table 1 shows the results obtained with our implementation for four different types of input, with the number of discs per type ranging between 100 and 1,024. For each input, we tried 1,000 different directions δ, and in the table we compare the shortest valid translation that the algorithm produced (over all different directions) with the asymptotically optimal value $r(S) + r(T)$. See the full version of the paper for a detailed description of each of the input types. As expected, the running time of the implementation is slightly super-quadratic. Our program runs in about 25 seconds on inputs with 1,000 discs; notice that the number of bad vippodromes in such instances is 999,000.

6 Further Research

Our research can be extended in multiple ways within the space-awareness framework. We could allow two translations per disc while aiming for minimal physical space (that also includes all the intermediate positions), in terms of the size of the bounding rectangle or disc. These problems can be studied with and without a global rigid translation of the target configuration. Alternatively, we could have considered variants where we allow an arbitrary initial rigid motion of the target configuration, or allow other motion paths than straight line paths. Since these problems are more general, they seem harder to solve with optimal space usage. One can also study space-aware reconfiguration for discs of varying sizes (the labeled version only), or for other, more complex shapes.

Viewing assembly planning from the space-aware perspective raises many challenging problems. We aim to find the smallest space (e.g., a round tabletop of minimum radius) where we can put the separate parts that need to be assembled into the final product, and such that the entire assembly process can take place within this space. The problem is more involved since we may need to store intermediate subassemblies, such that we can bring together some subassemblies into their relative placement in the final product, while avoiding other subassemblies, all within the same space.

Acknowledgement. The authors thank Mikkel Abrahamsen, Michael Hoffmann, Bettina Speckmann, and Frank van der Stappen for helpful discussions. They also thank the Lorentz Center in Leiden, where part of the research took place.

References

1. M. Abellanas, S. Bereg, F. Hurtado, A. G. Olaverri, D. Rappaport, and J. Tejel. Moving coins. *Comput. Geom.*, 34(1):35–48, 2006.
2. I. J. Balaban. An optimal algorithm for finding segments intersections. In *Proc. 11th Annu. Symposi. Comput. Geom. (SoCG)*, pages 211–219, 1995.
3. S. Bereg and A. Dumitrescu. The lifting model for reconfiguration. *Discrete Comput. Geom.*, 35(4):653–669, 2006.
4. S. Bereg, A. Dumitrescu, and J. Pach. Sliding disks in the plane. *Int. J. Comput. Geom. Appl.*, 18(5):373–387, 2008.
5. G. Călinescu, A. Dumitrescu, and J. Pach. Reconfigurations in graphs and grids. *SIAM J. Discrete Math.*, 22(1):124–138, 2008.
6. A. Dumitrescu. Mover problems. In J. Pach, editor, *Thirty Essays in Geometric Graph Theory*, pages 185–211. Springer, Berlin-Heidelberg, 2013.
7. A. Dumitrescu and M. Jiang. On reconfiguration of disks in the plane and related problems. *Comput. Geom.*, 46(3):191–202, 2013.
8. T. Geft, A. Tamar, K. Goldberg, and D. Halperin. Robust 2D assembly sequencing via geometric planning with learned scores. In *15th Proc. IEEE Internat. Conf. on Automation Science and Engineering (CASE)*, pages 1603–1610, 2019.
9. M. Goldwasser, J.-C. Latombe, and R. Motwani. Complexity measures for assembly sequences. In *Proc. IEEE Internat. Conf. on Robotics and Automation (ICRA)*, volume 2, pages 1851–1857, 1996.
10. D. Halperin, L. Kavraki, and K. Solovey. Robotics. In J. E. Goodman, J. O'Rourke, and C. Tóth, editors, *Handbook of Discrete and Computational Geometry*, chapter 51, pages 1343–1376. Chapman & Hall/CRC, 3rd edition, 2018.
11. D. Halperin, J. Latombe, and R. H. Wilson. A general framework for assembly planning: The motion space approach. *Algorithmica*, 26(3-4):577–601, 2000.
12. S. D. Han, N. M. Stiffler, A. Krontiris, K. E. Bekris, and J. Yu. High-quality tabletop rearrangement with overhand grasps: Hardness results and fast methods. In *Proc. 13th Conf. Robotics: Science and Systems*, 2017.
13. G. Havur, G. Ozbilgin, E. Erdem, and V. Patoglu. Geometric rearrangement of multiple movable objects on cluttered surfaces: A hybrid reasoning approach. In *Proc. IEEE Internat. Conf. on Robotics and Automation (ICRA)*, pages 445–452, 2014.
14. M. Levihn, T. Igarashi, and M. Stilman. Multi-robot multi-object rearrangement in assignment space. In *Proc. IEEE/RSJ Internat. Conf. on Intelligent Robots and Systems (IROS)*, pages 5255–5261, 2012.
15. D. J. Pearce and P. H. J. Kelly. A dynamic topological sort algorithm for directed acyclic graphs. *ACM Journal of Experimental Algorithmics*, 11:1.7, 2006.
16. M. Sharir and P. K. Agarwal. *Davenport-Schinzel Sequences and Their Geometric Applications.* Cambridge University Press, New York, 1995.
17. R. Stern, N. R. Sturtevant, A. Felner, S. Koenig, H. Ma, T. T. Walker, J. Li, D. Atzmon, L. Cohen, T. K. S. Kumar, R. Barták, and E. Boyarski. Multi-agent pathfinding: Definitions, variants, and benchmarks. In *Proc. 12th Internat. Symposi. on Combinatorial Search (SOCS)*, pages 151–159, 2019.