Constructing Order Type Graphs Using an Axiomatic Approach

Sergey Bereg(✉) and Mohammadreza Haghpanah

University of Texas at Dallas, Richardson, TX 75080, USA
{besp,Mohammadreza.Haghpanah}@utdallas.edu

Abstract. A given order type in the plane can be represented by a corresponding point set. However, it might be difficult to recognize the orientations of some point triples. Recently, Aichholzer *et al.* [3] introduced exit graphs for visualizing order types in the plane. We present a new class of geometric graphs, called *OT-graphs*, using abstract order types and their axioms described in the well-known book by Knuth [15]. Each OT-graph corresponds to a unique order type. We develop efficient algorithms for recognizing OT-graphs and computing a minimal OT-graph for a given order type in the plane. We provide experimental results on all order types of up to nine points in the plane including a comparative analysis of exit graphs and OT-graphs.

Keywords: Order type · CC systems · Chirotopes · Oriented matroids

1 Introduction

The *orientation* of three noncollinear points in the plane is either clockwise CW or counterclockwise CCW. In this paper we assume that point sets are in general position the plane. Two finite point sets in the plane have the *same order type* if there is a bijection between them preserving orientation of any three distinct points. The equivalence classes defined by this equivalence relation are the *order types* [14].

Recently, Aichholzer *et al.* [3] asked "... suppose we have discovered an interesting order type, and we would like to illustrate it in a publication." This is exactly the problem that we were facing in our recent paper [5] where we found that the order type 1874 for 9 points from the database [2] provides a (tight) lower bound for Tverberg partitions with tolerance 2, see Fig. 2(a). Of course, any order type in the plane can be represented by a corresponding point set (or explicit coordinates of the points). However, it might be difficult to recognize the orientations of some point triples. Aichholzer *et al.* [3] introduced exit graphs for visualizing order types in the plane. Let S be a set n points in the plane and let $a, b, c \in S$. Then (a, b) is an *exit edge* with *witness* c if there is no $p \in S$ such that line ap separates b from c or line bp separates a from c, see Fig. 1(a). Geometrically, it means that an *hourglass* defined by a, b, c is empty. The set of exit edges form the *exit graph* of S. To verify that (a, b) is an exit edge with witness

W. Wu and Z. Zhang (Eds.): COCOA 2020, LNCS 12577, pp. 802–816, 2020.
https://doi.org/10.1007/978-3-030-64843-5_54

c, one can check that every point $p \in S \setminus \{a, b, c\}$ is in $A \cup B$, see Fig. 1(a). Note that $A \cap S$ and $(B \cap S) \cup \{c\}$ is a partition of $S \setminus \{a, b\}$ by line ab. We will define a new class of graphs called *OT-graphs* using such partitions.

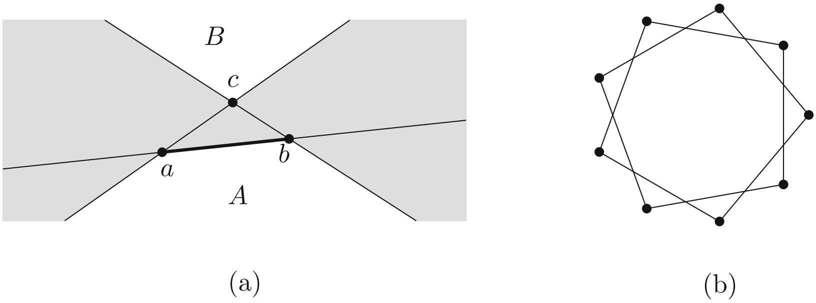


Fig. 1. (a) Exit edge (a, b) with witness c. The hourglass-shaped region (shown in gray) is empty of points. (b) An exit graph for 9 points in convex position.

We define an *OT-graph* on S using two ingredients. First, every edge (a, b) in an OT-graph is equipped with the partition of S by line ab, i.e. $S \setminus \{a, b\} = S^+_{ab} \cup S^-_{ab}$ where S^+_{ab} (S^-_{ab}) contains points $c \in S$ such that a, b, c has counterclockwise (clockwise) orientation. Second, we assume that an OT-graph contains a sufficient number of edges to decide the order type of points using axioms described in the well-known book by Knuth [15]. It is easy to visualize the partitions of S for the edges of an OT-graph by drawing lines through them. This may result in a dense drawing, so we omit lines in the drawing if their partitions can be easily seen. For example, the OT-graph for the order type 1874 for 9 points from the database [2] shown in Fig. 2(b) has ten edges and only two lines are sufficient. The property of this graph (since it is an OT-graph) is that the orientation of any triple abc can be decided either (a) directly from the graph if there is an edge with both endpoints in $\{a, b, c\}$, or (b) algebraically using five axioms [15].

Comparison of OT-Graphs and Exit Graphs. Both exit graphs and OT-graphs can be used for visualizing order types of points. It is not sufficient for verifying an order type to just draw such graphs. For exit graphs, one needs to see the witness and the hourglass for every exit edge. For OT-graphs, one needs to see only the lines extending the edges. The hourglasses for exit graphs and the lines for OT-graphs are needed only when some triples of points are almost collinear.

Exit graphs and OT-graphs are also different in the following sense. For a given order type (as a point set), the exit graph is unique but OT-graphs are not since OT-graphs are defined using combinatorial axioms of Knuth [15]. Therefore we have an optimization problem of computing a minimum-size OT-graph for a given order type. We believe that this optimization problem is NP-hard. For example, we believe that the OT-graph shown in Fig. 2(b) has the least number

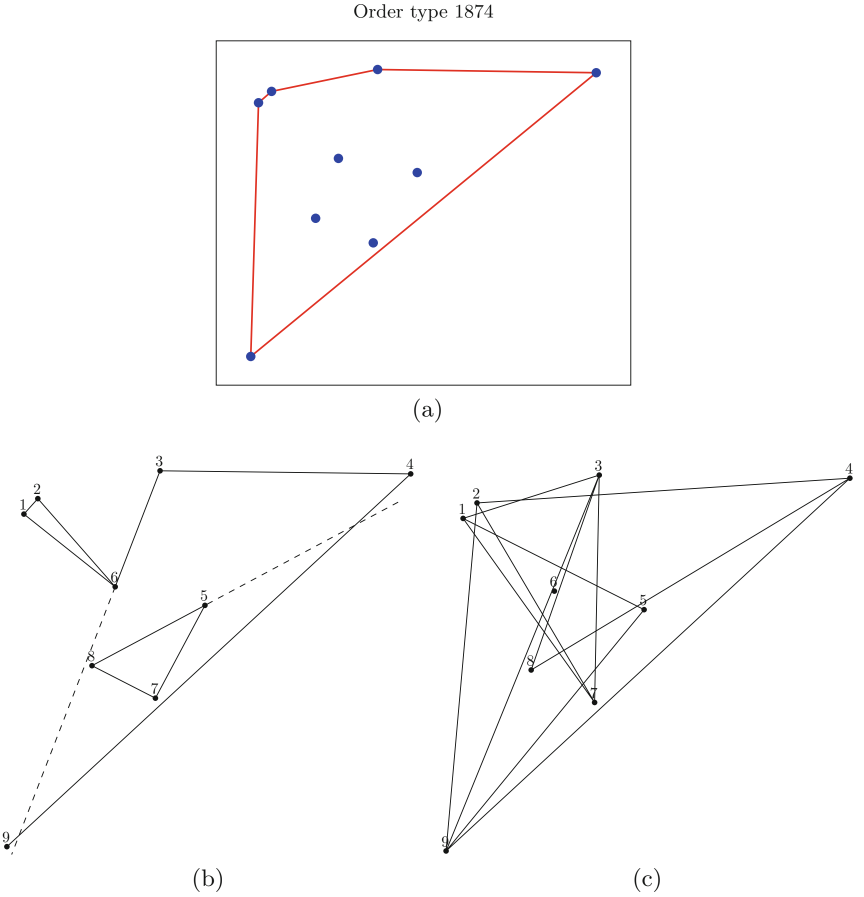


Fig. 2. (a) The order type 1874 for 9 points from the database [2]. (b) An OT-graph with 9 edges for the order type 1874 (several OT-graphs with 9 edges were computed by an extensive search). (c) The exit graph for the order type 1874.

of edges (9) for order type 1874 but we do not have a proof for it. Note that the OT-graph has 9 edges but the edge graph has 12, see Fig. 2(c) .

Identification of Order Types. Aichholzer *et al.* [3] suggested requirements for a graph representing an order type: "... we want to reduce the number of edges in the drawing as much as possible, but so that the order type remains uniquely identifiable." OT-graphs (including the set of edges and the corresponding partitions) characterize order types, i.e. each OT-graph corresponds to only one order type. Unfortunately, it does not hold for the exit graphs. As an example, Aichholzer *et al.* [3] constructed two sets each of 14 points[1] such that the exit edges are the same but the order types are different. With respect to minimiz-

[1] Using a pseudoline arrangement from [11].

ing the number of edges, we provide a comparative analysis of exit graphs and OT-graphs of all order types of up to 9 points in Sect. 6. Except few cases, OT-graphs have smaller number of edges. For example, Fig. 3 shows order type 1268 of 9 points where the exit graph has 15 edges but the OT-graph has only 8 edges. Furthermore, the OT-graph shown in Fig. 3(b) has non-crossing edges.

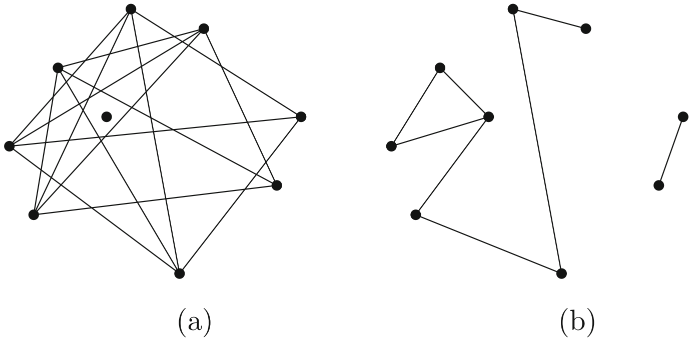


Fig. 3. The order type 1268 of 9 points represented as (a) the exit graph and (b) the OT-graph.

and we provide some algorithms for computing OT-graphs using combinatorial proofs and axioms in Sect. 5.

An interesting question is to find the smallest OT-graphs for points in convex position in the plane. Let c_n be the minimum number of edges in an OT-graph for n points in convex position.

Theorem 1. *For any $n \geq 4$, $c_n \leq \lfloor 2n/3 \rfloor$.*

It is interesting to find exact values of sequence c_n. We experimented with our randomized algorithm from Sect. 5 and conjecture that the bound in Theorem 1 is tight for all n up to 20. It is also interesting that the exit graph for n points in convex position has n edges, see Fig. 1(b) for an example.

Lower Bound. Another interesting question is to find the smallest OT-graph for an order type of n points in the plane. Based on our experiments, it is achieved for points in convex position if n is up to 9. Is it true for any n? One can argue that $\lceil n/4 \rceil$ is a lower bound for the number of edges in any OT-graph for n points in convex position. It is based on the fact that two consecutive points in the clockwise order along the boundary cannot be both isolated in an OT-graph.

Upper Bound. An obvious upper bound the smallest OT-graph for an order type of n points is $\binom{n}{2}$. We prove an upper bound in Sect. 4 which is smaller than $n^2/4$. The proof uses the idea of restricting the axioms in OT-graphs. Specifically, we prove the bound by using only Axioms 1, 2, and 3. Surprisingly, in this case, the smallest OT-graphs for any order type of n points have the same number of edges depending on n only.

Algorithms. For any set T triples with orientations, one can define its *CC-closure* $Cl(T)$ as the set of all triples that can be derived using Axioms 1–5. It is straightforward to make an algorithm for testing in $O(n^5)$ time whether a set of triples T is the closure of itself, i.e. $Cl(T) = T$. This can be modified to an algorithm for computing the CC-closure for an OT-graph (i.e. the set of triples defined by G). The algorithm repeats the following step. If new triples are found in the testing algorithm, they are added to the set of triples. This algorithm has $O(n^8)$ running time. We show that it can be improved to $O(n^5)$ time. We also develop a randomized algorithm for computing an OT-graph for a given order type in $O(n^5)$ time. We implemented it and run on many order types. For example, the smallest OT-graphs for all order types for $n = 4$ and $n = 5$ are shown in Fig. 4.

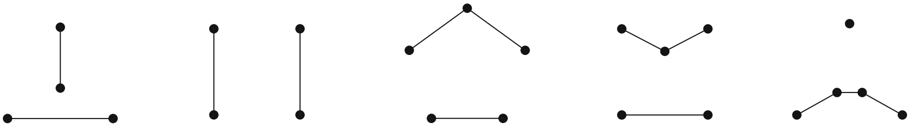


Fig. 4. Order types for $n = 4$ and $n = 5$.

Experiments. In Sect. 6 we provide experimental results using our algorithms on all order types of up to nine points in the plane. We also discuss a comparative analysis of exit graphs and OT-graphs using the size of the graphs.

Related Work. Order types are studied extensively, see for example the surveys [10,16]. Aichholzer *et al.* [4] studied representation of order types using radial orderings. Cabello [7] proved that the problem of deciding whether there is a planar straight-line embedding of a graph on a given set of points is NP-complete. Goaoc *et al.* [12] explored the application of the theory of limits of dense graphs to order types. The order types of random point sets were studied in [8,9] Goaoc and Welzl [13] studied convex hulls of random order types.

2 Preliminaries

Knuth [15] introduced and studied *CC-systems* (short for "counterclockwise systems") using properties of order types for up to five points. A *CC-system* for n points assigns true/false value for every ordered triple of points such that they satisfy the following axioms.

Axiom 1 (cyclic symmetry). $pqr \implies qrp$.
Axiom 2 (antisymmetry). $pqr \implies \neg prq$.
Axiom 3 (nondegeneracy). Either pqr or prq.
Axiom 4 (interiority). $tqr \wedge ptr \wedge pqt \implies pqr$.
Axiom 5 (transitivity). $tsp \wedge tsq \wedge tsr \wedge tpq \wedge tqr \implies tpr$.

Any set of n points in general position in the plane induces a CC-system if we use the "counterclockwise" relation on the points. The converse is not true due to the 9-point theorem of Pappus [6,15]. When defining a graph for order types using partitions (by the lines extending the edges) one should be careful. For example, we can ask whether a given set of orientations of some triples can be extended somehow to a CC-system. If by "extended" we mean finding a CC-system such that the given orientations are preserved in the CC-system, then this problem is NP-complete. Knuth [15] proved that it is NP-complete to decide whether specified values of fewer than $\binom{n}{3}$ triples can be completed to a CC-system. We define OT-graphs using the extension of the given orientations by simply applying 5 axioms. Note that Axioms 1, 2, 4, and 5 imply some orientations. Axiom 3 also can be formulated as an implication:

Axiom 3' (nondegeneracy). $\neg pqr \implies prq$.

Definition 2. *Let G be a graph for a point set S in the plane and let T be the set of triples abc such that ab, ac, or bc is an edge of G. Then G is the* OT-graph *if the orientation of every triple on S can be derived from T usings Axioms, 1,2,3',4, and 5.*

3 Convex Position

In this section, we explore OT-graphs for point sets in convex position and prove Theorem 1. Recall that c_n is the minimum number of edges in an OT-graph for n points in convex position. First, we prove that $c_n \leq n$ (Fig. 5).

Lemma 1. *Let S be a set of n points in convex position and let G be the graph (S, E) where E contains the edges of the convex hull of S. Then G is an OT-graph for S.*

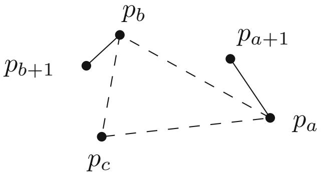


Fig. 5. Proof of Lemma 1.

Proof. Let $p_0, p_1, \ldots, p_{n-1}$ be the points of S in counterclockwise order. It suffices to prove that any triple $p_a p_b p_c$ with $0 \leq a < b < c \leq n-1$ has a CCW orientation. We prove it by induction on $m = \min\{b-a, c-b, a-c+n\}$. In the base case, $m = 1$. Then $(p_a, p_b), (p_b, p_c)$, or (p_c, p_a) is in E. Thus, p_a, p_b, p_c has a CCW orientation.

Suppose that $m > 1$ and $m = c - b$. Then $a + 1 < b$ and $b + 1 < c$. Edges (p_a, p_{a+1}) and (p_b, p_{b+1}) imply that triples $p_a p_{a+1} p_b, p_a p_{a+1} p_{b+1}, p_a p_{a+1} p_c$, and $p_a p_b p_{b+1}$ have a CCW orientation. By induction hypothesis, triple $p_a p_{b+1} p_c$ has a CCW orientation. By Axiom 5, $p_a p_b p_c$ has a CCW orientation.

Proof of Theorem 1. Let $p_0, p_1, \ldots, p_{n-1}$ be the points of S in counterclockwise order. We denote set $\{0, 1, \ldots, n-1\}$ by $[n]$.

First, suppose that $n = 3k$ for some $k \geq 2$. Consider a graph G with $2k$ edges as shown in Fig. 6. We prove that it is an OT-graph. By Lemma 1, it suffices to show that for any $i, j \in [n]$ with $j \neq i, i+1$ (modulo n), triple $p_i p_{i+1} p_j$ has a CCW orientation[2].

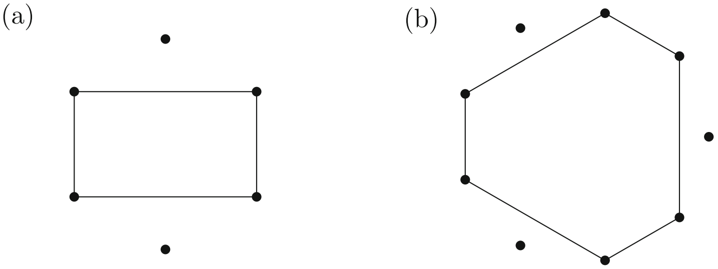


Fig. 6. OT-graphs for n points in convex position. (a) $n = 6$, (b) $n = 9$.

There are 3 cases to consider, see Fig. 7. Case (a) is clear since (p_i, p_{i+1}) is an edge of G. In Case (b), we can assume that $j \neq i+2, i-1$. Then it follows by Axiom 5 if we choose $t = p_{i+1}, s = p_{i+2}, p = p_j, q = p_{i-1}$, and $r = p_i$. In Case (c), we can assume that $j \neq i+2, i-1$. Knuth [15] proved that Axioms $1, 2, 3$, and 5 imply an axiom dual to Axiom 5.

Axiom 5' (dual transitivity). $stp \wedge stq \wedge str \wedge tpq \wedge tqr \implies tpr$.

Then Case (c) follows by Axiom 5' if we choose $t = p_i, s = p_{i-1}, p = p_{i+1}, q = p_{i+2}$, and $r = p_j$.

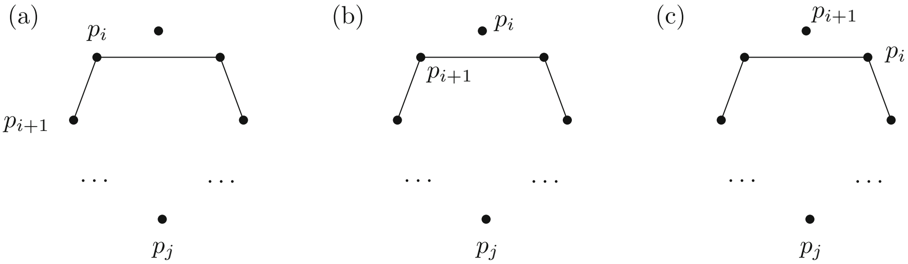


Fig. 7. Proof of Theorem 1 for $n = 3k$.

Now, suppose that $n = 3k+1$ for some $k \geq 2$. Consider a graph G with $2k$ edges as shown in Fig. 8(a). We prove that it is an OT-graph. By Lemma 1,

[2] This condition for a fixed i implies that (p_i, p_{i+1}) could be an edge in an OT-graph.

it suffices to show that for any $i, j \in [n]$ with $j \neq i, i+1$ (modulo n), tripe $p_i p_{i+1} p_j$ has a CCW orientation. If p_i or p_{i+1} is an isolated vertex in G then the argument is the same as in Case (b) and (c) for $n = 3k$, see Fig. 7(b) and (c). If (p_i, p_{i+1}) is an edge of G then $p_i p_{i+1} p_j$ has a CCW orientation. It remains to consider the case where (p_i, p_{i+1}) is one of two missing edges in the convex hull at the top, see Fig. 8(a). By symmetry, we assume that (p_i, p_{i+1}) is as shown in Fig. 8(b).

Suppose that vertex p_j has degree 2 in G. Let l be the length of path $p_j p_i$ in G. We show a CCW orientation of $p_i p_{i+1} p_j$ by induction on l. If $l = 1$ the orientation follows from edge $p_{i-1} p_i$ of G. If $l > 1$ then it follows by Axiom 5 if we choose $t = p_{i+1}, s = p_{i+2}, p = p_j, q = p_{j+1}$, and $r = p_i$. Note that $p_{i+1} p_j q$ has a CCW orientation since (p_j, q) is an edge of G. Also, $p_{i+1} q p_i$ has a CCW orientation by the induction hypothesis.

If vertex p_j is isolated in G then we choose p, q, r, s, t in the same way, see Fig. 8(c). Then triple tpq has a CCW orientation from the previous case (pq is an edge of convex hull). And triple tqr has a CCW orientation from the previous case (q has degree 2). By Axiom 5 tripe $p_i p_{i+1} p_j$ has a CCW orientation.

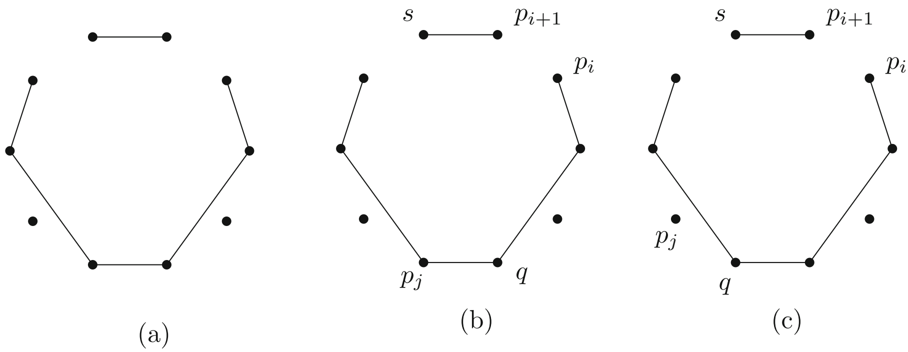


Fig. 8. Proof of Theorem 1 for $n = 3k + 1$.

Finally, suppose that $n = 3k + 2$ for some $k \geq 2$. Consider the graph shown in Fig. 9. It is an OT-graph by the same argument as for $n = 3k + 1$. □

Remark. The OT-graphs presented in the proof of Theorem 1 are not unique. Our program finds also other graphs of the same size, see Fig. 10.

4 Axioms 1, 2, and 3 only

Let $e_{123}(S)$ be the minimum number of edges in an OT-graph for a set S of points in general position in the plane if only Axioms 1,2, and 3 are used. Surprisingly, for any set S of n points (i.e. for any order type), the smallest OT-graph *always* contains the same number of edges depending on n only.

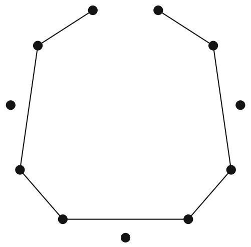


Fig. 9. OT-graph for $n = 3k + 2$.

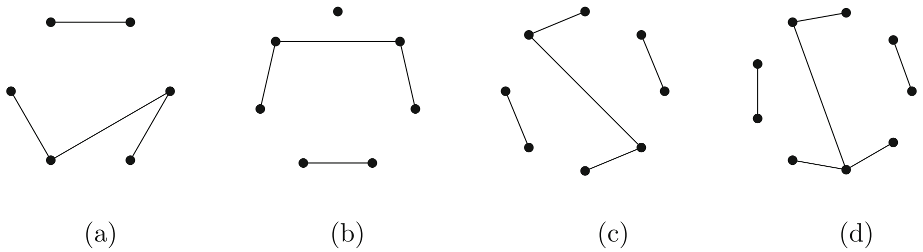


Fig. 10. OT-graphs for $n = 6, 7, 8, 9$ points in convex position computed by a program.

Theorem 3. *For any set S of $n \geq 2$ points in general position in the plane, $e_{123}(S) = \lfloor \frac{n}{2} \rfloor \lfloor \frac{n-1}{2} \rfloor$.*

We omit the proof due to space constraint.

5 Algorithms

Let $G = (S, E)$ be an OT-graph for a set S of n points in the plane. Let $T(G)$ be the set of triples abc such that (a, b), (a, c), or (b, c) is an edge of G. Note that the orientation of abc is given by the partition of the corresponding edge. We define the *CC-closure* of G as the set all triples that can be proven by applying Axioms 1–5 from $T(G)$. Note that the *CC-closure* can be defined for any subset of triples of points with orientations.

Problem 4 (ComputingCC-Closure)

Given *an OT-graph G.*
Compute *the CC-closure of G.*

A naive approach to solve ComputingCC-Closure is to use an algorithm for testing CC-closure.

Problem 5 (TestingCC-Closure)

Given *a set of triples with orientations for n points.*
Decide *whether a new triple can be derived using Axioms 1–5. If so, find a new triple using Axioms 1–5.*

By applying an algorithm for TestingCC-Closure to $T(G)$ we can extend $T(G)$ (if possible) and solve ComputingCC-Closure. TestingCC-Closure can be done in $O(n^5)$ time (since Axiom 5 requires 5 points). There are $\binom{n}{3}$ triples and, thus, the naive approach takes $O(n^8)$ time. We show that it can be done much faster.

Theorem 6. ComputingCC-Closure *can be solved in $O(n^5)$ time.*

Algorithm 1.

1. Make a list L_1 of all input triples with orientations (list L_1 stores all triples with known orientations). Copy $L_2 = L_1$.
2. While list L_2 is not empty, remove any triple abc from list L_2. Apply Axioms as follows. Find new triples using Axioms 1,2,3',4, and 5 such that triple abc is used in the condition of the axiom with the same orientation. If a new triple (i.e. not in L_1) is found, say pqr, then add it to L_1 and L_2.
3. Return list L_1.

Proof. To implement Algorithm 1 efficiently, we store triples of L_1 in a 3-dimensional array A_1. The value of $A_1[a, b, c]$ is *true*/*false* if abc has a CCW/CW orientation; otherwise $A_1[a, b, c]$ = *null*. Using array A_1, we can decide in $O(1)$ time whether a triple is in list L_1 or not. Each triple abc is processed in Step 2 in $O(n^2)$ time since
(i) Axioms 1,2, and 3' can be applied at most one time,
(ii) Axiom 4 can be applied at most $n - 3$ times and
(iii) Axiom 5 can be applied at most $(n - 3)(n - 4)$ times.
Each triple is added to (and removed from) list L_2 at most one time. The number of triples removed from L_2 in Step 2 is $O(n^3)$. Therefore, the total time complexity of the algorithm is $O(n^5)$.

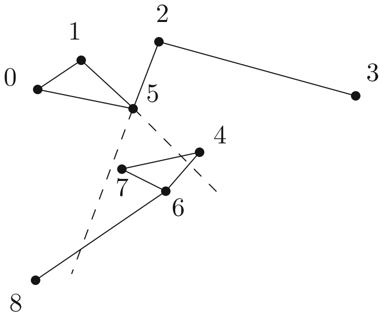
In our implementation of Algorithm 1, we do not maintain list L_1. Instead, we compute it in the end using array A_1.

The problem of computing the smallest OT-graph for a given order type seems complicated. Note that, if we restrict the axioms to Axioms 1,2, and 3 then a simple polynomial-time algorithm for computing the smallest OT-graph exists by Theorem 3 (by constructing two cliques). Next, we extend Algorithm 1 to a randomized algorithm for computing an OT-graph without increasing the running time. We incrementally add edges to a graph $G = (S, E)$ until G is an OT-graph for S. We store L_1, a list of triples abc such that $(a, b), (a, c)$, or (b, c) is an edge of G. Note that the orientation of abc can be computed using the coordinates of a, b, and c in $O(1)$ time. As in Algorithm 1, we have list L_2 which is useful for computing the CC-closure of G.

```
Algorithm 2.
Input: an order type given by a set of points $S$.
Output: an OT-graph $G$ for $S$

1. Set $E = \emptyset$. Set $countCC = 0$, the number of triples in the CC-closure of
   $G = (S, E)$.
2. Compute list $R$ of $\binom{n}{2}$ edges in the complete graph for $S$.
3. Initialize array $A_1[n, n, n]$ with entry values $null$ and empty list $L_2$.
4. While $countCC < n(n-1)(n-2)$
   (a) Remove a random edge $(a, b)$ from $R$.
   (b) If $A_1[a, b, c] \neq null$ for all $c \in S \setminus \{a, b\}$ then continue the "while" loop
       otherwise do the following steps (c) and (d).
   (c) Add $(a, b)$ to list $E$. For each $c \in S \setminus \{a, b\}$ such that $A_a[a, b, c] \neq null$,
       add one of the triples $(a, b, c)$ or $(b, a, c)$ to list $L_2$ which has a CCW
       orientation.
   (d) Process list $L_2$ as in Algorithm 1.
```

Algorithm 2 (if repeated several times) can find the smallest OT-graph for a given order type, see for example Fig. 10. We also make a program that helps to verify the proof of an OT-graph. Note that a triple can be proven differently using Axioms 1–5. We develop a program for finding a human-readable proof. Once the best OT-graph for a given order type is found, the program computes a proof only for triples that require Axioms 4 and 5 (Axioms 1–3 are obvious). For example, Fig. 11 illustrates an OT-graph among all order type of 9 points and the format of the proof.



Proof:
Triple (7, 2, 0), Axiom 5 [2, 5, 0, 4, 7]
Triple (7, 3, 0), Axiom 5 [3, 2, 0, 4, 7]
Triple (7, 2, 1), Axiom 5 [2, 5, 1, 4, 7]
. . .

Fig. 11. An OT-graph for order type of 9 points and a part of the proof of it. The format for Axiom 4 is $[p, q, r, t]$ and the format for Axiom 5 is $[p, q, r, s, t]$.

Greedy Algorithm. Each iteration Algorithm 2 is quite fast (for relatively small n). However, it may require many runs to find small OT-graphs. Another possibility is a greedy algorithm where all possible edges for adding to the current graph are tested and the edge maximizing the size of the CC-closure is selected. Since the computation of the CC-closure takes $O(n^5)$ time, this approach is computationally expensive (it takes $O(n^7)$ time for selecting one edge and $O(n^9)$ for constructing the OT-graph). We developed a different greedy algorithm where the edge maximizing the size of the CC-closure using only Axioms 1,2,3 is selected. We found an implementation of this algorithm without increasing the running time, i.e. with running time $O(n^5)$. We add a new 2-dimensional array $C[..]$ for counting triples corresponding to the edges. Initially, $C[a, b] = n - 2$ for all pairs (a, b) of points $a \neq b$. Every time a new triple, say abc, is proven using Axioms we subtract one from $C[x, y]$ for all possible $x \neq y \in \{a, b, c\}$. Then, the greedy selection can be done by finding an edge (a, b) maximizing $C[a, b]$.

The total running time of this algorithm has two components. It is $O(n^5)$ time as in the randomized algorithm plus the total time for processing new array $C[..]$. There are $O(n^3)$ new triples and each triple requires $O(1)$ to update $C[..]$. This step takes $O(n^3)$ time in total. The computation of a new edge for G takes $O(n^2)$ time. Thus, the total time for computing the edges of $G = (S, E)$ is $O(mn^2)$ where $m = |E|$. Therefore, the total time for processing array $C[..]$ is $O(n^4)$. *Minimal OT-graphs.* When an OT-graph with m edges is computed, it can be checked for minimality. An OT-graph for some order type is *minimal* if removal of any edge results in a graph which not an OT-graph, i.e. its CC-closure does not contain all possible triples. This can be decided by applying the algorithm for ComputingCC-Closure m times.

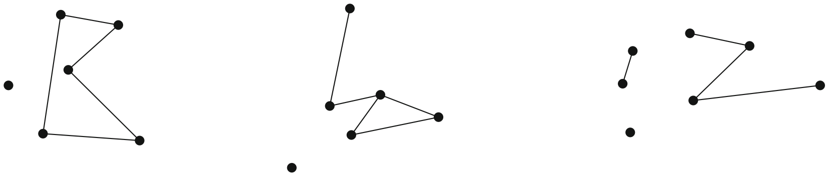
6 Experiments

We implemented the randomized algorithm (Algorithm 2) and the greedy algorithm for computing OT-graphs. The programs are written in Java 8 using multi-threading and thread synchronization. We used a Linux server with 32 CPUs and 62 GB RAM to execute our program. We have computed the exit graphs and the OT-graphs on the database of order types [2] for $n = 3, 4, \ldots, 9$. To achieve current database and ensure the minimality of edges of OT-graphs, we run it around more than 3 days on the dataset. The results are shown in Table 1. Our experiments show that in many cases the greedy algorithm outperforms Algorithm 2 by the size of an OT-graph. Therefore, we iterate the greedy algorithm (with random tie-breaking) first and then iterate Algorithm 2 searching for a possible improvement. The number of iterations used for the greedy algorithm was 300,1200,10000 for $n = 7, 8, 9$ respectively. The number of iterations used for Algorithm 2 significantly larger (100000 for $n = 9$). About 70% of OT-graphs in Table 1 were computed using the greedy algorithm. The improvement achieved by Algorithm 2 was rather small: typically one edge reduction for an order type. The program implementing Algorithm 2 is still running and hopefully, new OT-graphs will be computed in a few months.

It is interesting that the smallest OT-graphs are achieved for 14 order types with $n = 6$, for 2 order types with $n = 7$, for 26 order types with $n = 8$, and for 124 order types with $n = 9$. There are only 2 order types for $n = 6$ whose

Table 1. OT-graphs for n up to 9. Column $i, i = 1, 2, \ldots, 11$ contains the number of OT-graphs with i edges.

n	1	2	3	4	5	6	7	8	9	10	11	Total
3	1											1
4		2										2
5			3									3
6				14	2							16
7				2	79	54						135
8					26	696	1,802	791				3,315
9					1	234	9,379	49,331	73,906	25,671	295	158,817



(a) order type 7

(b) order type 15

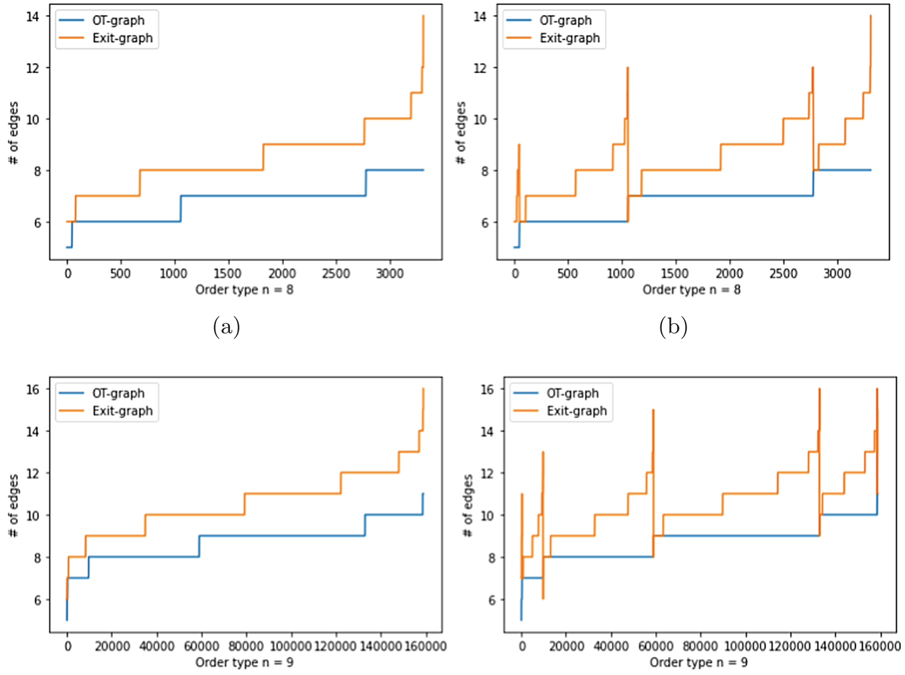
(c) order type 17

Fig. 12. Extreme OT-graphs for $n = 6$ and $n = 7$. (a), (b) Two order types for $n = 6$ maximizing the number of edges. (c) An order type for $n = 7$ (different from the convex case) minimizing the number of edges.

OT-graphs require 5 edges. They are shown in Fig. 12(a) and (b). The two order types for $n = 7$ that admit OT-graph with 4 edges are shown in Fig. 10(b) (the convex position) and in Fig. 12(c).

Let $\mu(n)$ be the minimum number of edges in an OT-graph for n points. Based on our experiments, we conjecture that $\mu(4) = 2$, $\mu(5) = 3, \mu(6) = \mu(7) = 4, \mu(8) = \mu(9) = 5$. This can be compared with exit graphs where the minimum number of edges is the same for $n = 5, 6, 7, 8$ but is larger for $n = 9$, see Fig. 13.

Figure 13(a), (c) shows the distribution of the graph sizes (OT-graph vs exit graph). Figure 13(b), (d) shows comparison of the graph sizes for each order type (the order types are sorted by the size of OT-graph and exit graph). Except one order type for $n = 8$ and 17 order types for $n = 9$, the OT-graphs are smaller that the exit graphs. For $n = 9$, the maximum size of OT-graph/exit graph is 11/16, respectively. The corresponding total number of edges is 1386819 for OT graphs and 1673757 for exit graphs which is 82.85%.



(a)

(b)

(c)

(d)

Fig. 13. The sizes of OT-graphs and exit graphs of order types for (a,b) $n = 8$ and (c,d) $n = 9$. The order types in (a) and (c) are sorted independently for OT-graphs and exit graphs. The functions in (b) and (d) use the same order type on the x-axis (the order types are sorted lexicographically).

7 Concluding Remarks

In this paper, we introduced new geometric graphs, OT-graphs, for visualizing order types in the plane. This new concept gives rise to many interesting questions. Is it true that the smallest size OT-graphs for all order types of n points are achieved for points in convex position? Is the bound in Theorem 1 tight?

In many cases there are different OT-graphs of minimum size for the same order type. One can use other criteria to optimize OT-graphs, for example, crossings. Figure 4 shows that there exist OT-graphs without crossings for all order types of 4 and 5 points. Theorem 1 shows that there are OT-graphs without crossings for points in convex position. Can it be generalized in this sense?

Finally, we plan to run our program on order types for larger values of n using the database of order types developed by Aichholzer, Aurenhammer, and Krasser [2]. It is a challenging problem since the number of order types grows as $2^{\Theta(n \log n)}$ [1,10] (there are 14,309,547 order types for $n = 10$.)

References

1. A063666: Euclidean order types: number of realizable order types of n points in the plane (2001). https://oeis.org/A063666
2. Aichholzer, O., Aurenhammer, F., Krasser, H.: Enumerating order types for small point sets with applications. Order **19**(3), 265–281 (2002)
3. Aichholzer, O., et al.: Minimal representations of order types by geometric graphs. In: Graph Drawing (Proc. GD 2019), pp. 101–113 (2019)
4. Aichholzer, O., Cardinal, J., Kusters, V., Langerman, S., Valtr, P.: Reconstructing point set order types from radial orderings. Int. J. Comput. Geom. Appl. **26**(03–04), 167–184 (2016)
5. Bereg, S., Haghpanah, M.: New lower bounds for Tverberg partitions with tolerance in the plane. Disc. Appl. Math. **283**, 596–603 (2020)
6. Bokowski, J., Richter, J., Sturmfels, B.: Nonrealizability proofs in computational geometry. Disc. Comput. Geom. **5**(4), 333–350 (1990). https://doi.org/10.1007/BF02187794
7. Cabello, S.: Planar embeddability of the vertices of a graph using a fixed point set is NP-hard. J. Graph Algorithms Appl. **10**(2), 353–363 (2006)
8. Cardinal, J., Monroy, R.F., Hidalgo-Toscano, C.: Chirotopes of random points in space are realizable on a small integer grid. In: Friggstad, Z., Carufel, J.D. (eds.) Proceedings of the 31st Canadian Conference on Computational Geometry, CCCG 2019, University of Alberta, Edmonton, Alberta, Canada, 8–10 August 2019, pp. 44–48 (2019)
9. Devillers, O., Duchon, P., Glisse, M., Goaoc, X.: On order types of random point sets. CoRR abs/1812.08525 (2018). http://arxiv.org/abs/1812.08525
10. Felsner, S., Goodman, J.E.: Pseudoline arrangements. In: Handbook of Discrete and Computational Geometry, pp. 125–157. Chapman and Hall/CRC (2017)
11. Felsner, S., Weil, H.: A theorem on higher Bruhat orders. Disc. Comput. Geom. **23**(1), 121–127 (2000)
12. Goaoc, X., Hubard, A., de Joannis de Verclos, R., Sereni, J., Volec, J.: Limits of order types. In: Arge, L., Pach, J. (eds.) 31st International Symposium on Computational Geometry, SoCG 2015, 22–25 June 2015, Eindhoven, The Netherlands. LIPIcs, vol. 34, pp. 300–314. Schloss Dagstuhl - Leibniz-Zentrum für Informatik (2015)
13. Goaoc, X., Welzl, E.: Convex hulls of random order types. In: Cabello, S., Chen, D.Z. (eds.) 36th International Symposium on Computational Geometry, SoCG 2020, LIPIcs, Zürich, Switzerland, 23–26 June 2020, vol. 164, pp. 49:1–49:15. Schloss Dagstuhl - Leibniz-Zentrum für Informatik (2020)
14. Goodman, J.E., Pollack, R.: Multidimensional sorting. SIAM J. Comput. **12**(3), 484–507 (1983)
15. Knuth, D.E. (ed.): Axioms and Hulls. LNCS, vol. 606. Springer, Heidelberg (1992). https://doi.org/10.1007/3-540-55611-7
16. Richter-Gebert, J., Ziegler, G.M.: Oriented matroids. In: Handbook of discrete and computational geometry, pp. 159–184. Chapman and Hall/CRC (2017)