Abstract Order Type Extension and New Results on the Rectilinear Crossing Number - Extended Abstract

Oswin Aichholzer* Hannes Krasser†

Abstract

We provide a complete data base of all realizable order types of 11 points in general position in the plane. Moreover, we develop a novel and efficient method for complete extension to (abstract) order types of size 12 and more. With our approach we have been able to determine the exact rectilinear crossing number for up to $n = 17$, and slightly improved the asymptotic upper bound. We briefly discuss further applications of this approach.

1 Introduction

A finite point set in the plane belongs to the most common ingredients for computational and combinatorial geometry problems. For quite many, especially combinatorial problems, the exact metric properties are not relevant, but the combinatorial properties of the underlying point set play the main role. More precisely, the crossing properties of the line segments spanned by the point set already determine the problem. Triangulations, crossing numbers, convexity problems are among other famous examples. Order types provide a means to encode the combinatorial properties of finite point sets. The order type of a point set $S = \{p_1, .., p_n\}$ is a mapping that assigns to each ordered triple (p_i, p_j, p_k) an orientation. Throughout this work we assume that S is in general position, that is, the orientation of each point triple is either clockwise or counter-clockwise. Two point sets S_1, S_2 are of the same order type if and only if there is a bijection between S_1 and S_2 such that either all (or none) corresponding triples are of equal orientation.

To achieve results for point sets of fixed size for the problems mentioned above, it is sufficient to check one instance of each order type instead of looking at all (infinitely many) point sets. A data base containing all order types of size up to 10 already exists [2] and has been applied fruitfully to many problems in computational and combinatorial geometry [6].

Order types have played a crucial role in gathering knowledge about crossing numbers. The crossing number of a graph G is the least number of edge crossings attained by a drawing of G in the plane. We consider the problem of finding the rectilinear (edges are required to be straight line segments) crossing number $\overline{cr}(K_n)$ of the complete graph K_n on n vertices [12]. Determining $\overline{cr}(K_n)$ is commonly agreed to be a difficult task, see [3] for references and details. So far the exact values of $\overline{cr}(K_n)$ have been known for $n \leq 12$ [2, 3]. In Section 4 we extend this range to $n \leq 17$. Moreover, we also present an improvement on the asymptotic upper bound of $\overline{cr}(K_n)$. Our results are available on-line [1]. We close with a brief discussion of further applications of our approach.

*Institute for Softwaretechnology, Graz University of Technology, Graz, Austria, oaich@ist.tugraz.at

†Institute for Theoretical Computer Science, Graz University of Technology, Graz, Austria, hkrasser@igi.tugraz.at

2 Order type data base for n=11

A complete data base of order types for sets with up to 10 points has already been established [2]. We present an extension to this data base for point sets of size 11. Our approach is strongly related to [2] and uses improved techniques to cover the following three steps, cf. [2] for the necessary concepts and definitions.

1. Generating a complete candidate list of abstract order types
2. Grouping abstract order types into projective classes and deciding realizability
3. Realizing all realizable order types by point sets with "nice" coordinate representation

For the first step, we acquired 2 343 203 071 inequivalent abstract order types. We only stored one representative of each projective class explicitly at this time. This evaluates to 41 848 591 abstract projective order types of size $n = 11$, see Table 1.

The second step - deciding realizability - is the hardest part of the construction. The trouble is, that this decision problem is known to be NP-hard [13] and no practical algorithms are known, not even for small sets, say of size 10 or 11. We tried to find realizations and started by applying refined versions of the heuristic methods from [2] for each projective order type class. These worked for most of the abstract order types in question. For classifying non-realizable order types, we used a well-known practical algorithm for a non-realizability proof developed by Bokowski and Richter [9]. To our benefit, the heuristics for finding realizations and proving non-realizability were sufficient to completely settle the case for $n = 11$, see Table 1.

The main goal of the third step is to store the data base in an application friendly way. To this end, we provide two representations of the data base. An explicit version of the data base contains one point set for each planar order type, all in 16-bit integer representation.

projective abstract o.t.	41 848 591
− thereof non-realizable	155 214
= projective order types	41 693 377
abstract order types	2 343 203 071
− thereof non-realizable	8 690 164
= order types	2 334 512 907

Table 1: Number of order types of cardinality $n = 11$.

Supporting the reliability in the construction of our data base, all algorithms to generate the complete data base of abstract order types are of purely combinatorial nature. The applied methods for deciding realizability are heuristics, but the acquired results can be checked in a deterministic way.

The vast storage and the lack of applicability are the two main reasons - apart from calculation time - that we do not have a complete data base of order types with 12 or more points.

3 Complete abstract point extension

For several problems and conjectures the complete order type data base of sets of up to 11 points has been sufficient to give a final answer, cf. [3]. However, many problems tend to be harder and cannot be settled just by checking all cases for size up to 11. Still it looks highly plausible to gain significantly more insight with a few additional points, say 12 or 13 points. To evade these obstacles we make use of well-known theoretical results. For many problems on point sets there exist inductive restrictions, so-called subset properties.

Definition 1 *(***Subset property***) Let S_n be an order type consisting of n elements, $n \geq 4$, and consider some property that is valid for S_n. Then this property is called a subset property if and only if there exists some $S_{n-1} \subset S_n$ of $n-1$ elements such that a similar property holds for S_{n-1}.*

Our general idea is to exploit subset properties for order type based problems to obtain results beyond point sets of size 11. First, we are applying the order type data base to completely determine the problem for point sets of small size, that is, up to $n = 11$. This gives a set of result order types of cardinality 11, all realized by point sets. Next, we enumerate all order types of size 12 that contain one of the 11-point result order types as a subset. Applying the subset property, we are able to filter these 12-point order types. Only order types that fulfill the subset property are kept. Then we repeat this procedure, theoretically extending the set of result order types to arbitrary n.

For this technique, we require an algorithm that calculates for a given order type of cardinality n all $(n+1)$-point order types that contain the input order type as a sub-order type. We call this step complete point extension. It is well known that an extension technique relying only on the geometric realizations of the data base cannot guarantee completeness of the extension, see Figure 1. For a specific n-point realization of an order type we cannot derive all required $n+1$ order types just by adding a new point to this realization. To achieve completeness of the extension, we use an abstract extension method, that is, applying a combinatorial extension technique. We provide a one-element extension to an abstract order type by adding a pseudoline to the dual pseudoline arrangement in all combinatorially possible ways.

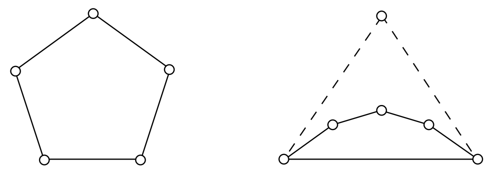

Figure 1: Two realizations of the order type of five points in convex position. Only the right point set can be extended in a way such that the resulting point set has three points on its convex hull.

For specific applications with a subset property, we define an order type extension graph. In this graph each order type is represented by a node. For each order type of size $n + 1$ (son), there is exactly one connection by an edge to a predecessor sub-order type of size n (father). By this definition we have that each order type corresponds to a unique predecessor order type by removal of a single point. On the other hand, an extension process that only extends corresponding to the edges of an order type extension graph (from father to son) enumerates each extended order type exactly once.

In general, the algorithm of complete abstract point extension extends one input order type point by point, then continuing on the remaining set of order types. After extension with one abstract point, we check if the created order type of size $n + 1$ (son) has the initial order type of size n as its predecessor order type (father) in the order type extension graph. Only if this is the case we keep it as a candidate for the output. The very general approach with the order type extension graph guarantees to avoid duplicates in the construction process, thus it can be used for recursive enumeration techniques, known as reverse search, cf. Avis and Fukuda [7]. An additional benefit of this technique is that it can be applied iteratively, i.e., extending from n points to $n+1$, $n+2$, and so on, without storing intermediate results. In fact, only the order types corresponding to a single path of the order type extension graph have to be kept in memory, that is, the edges describing the father-son relationship between order types of size n, $n+1$, $n+2$, and so on. This allows calculations which otherwise would not be possible because of enormous storage requirements for intermediate steps. In addition, applications based on the order type extension graph are easily executed in parallel. Thus, highly time intensive problems may be settled through distributed computing approaches.

4 New Rectilinear Crossing Numbers

4.1 Subset Property for $\overline{cr}(K_n)$

The next two well-known lemmas (see e.g. Guy [11] for references) provide the necessary relations to obtain a subset property for $\overline{cr}(K_n)$.

Lemma 1 $\overline{cr}(K_n) \geq \lceil \frac{n}{n-4}\, \overline{cr}(K_{n-1}) \rceil$

Corollary 2 (**Crossing number subset property**) *For any drawing of K_n with c crossings there exists at least one sub-drawing K_{n-1} with at most $\lfloor \frac{n-4}{n}\, c \rfloor$ crossings.*

Lemma 3 *Let $n \in \mathbb{N}$ be odd. Consider a straight-line drawing of K_n with c crossings. Then: $c \equiv \binom{n}{4} (mod\, 2)$.*

A drawing of K_{13} with 229 (or fewer) crossings contains at least one sub-drawing K_{12} with $\lfloor \frac{9}{13}\ 229 \rfloor = 158$ (or fewer) crossings. Recursive application shows that there exists a sub-drawing of size 11 with $\lfloor \frac{8}{12}\ 158 \rfloor = 105$ crossings. By the parity property we can further reduce the number of crossings for the 11-point subset to at most 104. Thus to achieve a data base of all order types of size 13 with 229 (or fewer) crossings, one can start with a complete data base of order types of size 11 defining drawings of K_n with at most 104 crossings, i.e., either 102 or 104 crossings.

4.2 Results on $\overline{cr}(K_n)$ for $n \geq 12$

Using the crossing number subset property, we were able to calculate the rectilinear crossing numbers for $n = 12, ..., 17$, see Table 2.

n	12	13	14	15	16	17
$\overline{cr}(K_n)$	153	229	324	447	603	798
d_n	1	4 534	20	16 001	36	$\geq$ 37269

Table 2: $\overline{cr}(K_n)$ for $n = 12, ..., 17$.

The numbers d_n of inequivalent drawings of K_n minimizing the number of crossings are given in the last row of Table 2. To obtain these numbers we had to perform the more challenging task of deciding the realizability of the calculated abstract order types. Our heuristics - see Section 2 - found realizing point coordinates for all optimal abstract drawings for $n \leq 16$. Thus, the calculated values are exact. Note that the numbers of inequivalent optimal drawings of K_n follow a parity pattern. There are relatively few drawings of K_n with $\overline{cr}(K_n)$ crossings for even n compared to the case of odd n. This property is the main reason that allows complete abstract extension to work so well, as the problem itself cuts down on the number of interesting sets periodically.

In addition to new results on $\overline{cr}(K_n)$ for constant n, we also achieved an improvement on the asymptotic upper bound. We constructed a set of 54 points with 115999 crossings such that with the strategy of lens replacement [3] we were able to prove the next theorem. The previously best known bound was $\overline{\nu}^* < 0.38074$, whereas $\overline{\nu}^* > 0.37533$ still holds as a lower bound [8].

Theorem 4 $\overline{\nu}^* = \lim_{n\to\infty} \overline{cr}(K_n)/\binom{n}{4} < 0.38058$

5 Further Applications

5.1 Happy End Problem

Erdös and Szekeres asked in 1935 for the smallest number $g(k)$, such that each point set in the plane with at least $g(k)$ points contains a convex k-gon [10]. For $k > 5$ this problem is still unsolved, where it is known that $g(6) \leq 37$. The conjecture is that the true value for $g(6)$ is 17. To answer this conjecture our plan is to apply our abstract extension technique in order to obtain all sets without empty convex hexagons for $n \leq 17$. If we cannot find a set for $n = 17$ this will prove the conjecture to be true. The subset property for this problem is obvious: any $n-1$ point subset of a set of n points to be considered must not contain a convex hexagon.

5.2 Decomposition

Similar to the convex-decomposition problem of decomposing a point set into convex polygons one might allow the resulting faces to be either convex polygons or pseudo-triangles [4]. When investigating this problem it turned out to be important to know optimal decompositions of small sets. In this context we asked for independent (disjoint) empty convex polygons spanned by the set. Let us briefly mention two results we got from the data base, see [4] for details. First: Any set of 8 points contains either an empty convex pentagon or two independent empty convex quadrilaterals. And with a similar flavor: Any set of 11 points contains either an empty convex hexagon or an independent empty convex pentagon and an empty convex quadrilateral. The mentioned results directly lead to an upper bound of $7n/10$ for the number of convex or pseudotriangular faces used to decompose a set of n points.

5.3 Counting Triangulations

Counting the number of triangulations of a set of points in the plane is another interesting geometric problem. Exact numbers, using our data base, are known for all sets with $n \leq 11$ points. The best general asymptotic lower bound for this problem is based on these results for small sets [5]. To improve the bound it will be useful to obtain a tight lower bound for $n = 12, 13, ...$ As a subset property for this task we can use the fact that adding an interior point to a given set increases the number of triangulations by some constant factor.

6 Open Problems

The next steps of our investigation will be to compute $\overline{cr}(K_{18})$. The possible range for $\overline{cr}(K_{18})$ is $\{1026, 1027, 1028, 1029\}$, where our conjecture is $\overline{cr}(K_{18}) = 1029$. Using heavy distributed computing we consider this task to be realistic in the near future.

An interesting open problem is whether there always exists at least one optimal drawing of K_n which contains an optimal sub-drawing of K_{n-1}. A potential counter-example is $n = 18$, as all 17-point subsets of the only known drawing of K_{18} with 1029 crossings determine more than $\overline{cr}(K_{17}) = 798$ crossings.

7 Acknowledgments

We would like to thank Birgit Breitenlechner, Clemens Huemer and Bernhard Kornberger for implementing many algorithms used for our investigations. We are grateful to H.F. Jensen for generously sharing his data and ideas about crossing numbers with us.

References

[1] O.Aichholzer, *Rectilinear Crossing Number Page.* `http://www.ist.tugraz.at/staff/aichholzer/crossings.html`

[2] O.Aichholzer, F.Aurenhammer, H.Krasser, *Enumerating order types for small point sets with applications.* Order, 19:265-281, 2002.

[3] O.Aichholzer, F.Aurenhammer, H.Krasser, *On the crossing number of complete graphs.* 18^{th} ACM Symposium on Computational Geometry (SoCG), Barcelona, Spain, pages 19-24, 2002.

[4] O.Aichholzer, C.Huemer, S.Renkl, B.Speckmann, and C.Toth, *Partitioning Point sets into Empty Convex Polygons and Pseudo triangles.* manuscript, 2004.

[5] O.Aichholzer, F.Hurtado, and M.Noy, *A lower bound on the number of triangulations of planar point sets.* Computational Geometry: Theory and Applications, 29(2):135-145, 2004.

[6] O.Aichholzer, H.Krasser, *The point set order type data base: A collection of applications and results.* 13^{th} Canadian Conference on Computational Geometry (CCCG), Waterloo, Ontario, Canada, pages 17-21, 2001.

[7] D.Avis, K.Fukuda, *Reverse search for enumeration.* Discrete Applied Mathematics 65, pages 618-632, 1996.

[8] J.Balogh, G. Salazar, *On k-sets, convex quadrilaterals, and the rectilinear crossing number of K_n.* Manuscript, submitted.

[9] J.Bokowski, J.Richter, *On the finding of final polynomials.* European J. Combinatorics 11, pages 21-34, 1990.

[10] P.Erdös, G.Szekeres, *A combinatorial problem in geometry.* Compositio Mathematica, (2):463-470, 1935.

[11] R.K.Guy, *A combinatorial problem.* Nabla (Bulletin of the Malayan Mathematical Society), 7, 68-72, 1960.

[12] F.Harary, A.Hill, *On the number of crossings in a complete graph.* Proc. Edinburgh Math. Society (2) 13 (1962), 333-338.

[13] N.E.Mnëv, *On manifolds of combinatorial types of projective configurations and convex polyhedra.* Soviet Math. Dokl., 32, pages 335-337, 1985.