

## An Ongoing Project to Improve the Rectilinear and the Pseudolinear Crossing Constants

Oswin Aichholzer<sup>1</sup> Frank Duque<sup>2</sup> Ruy Fabila-Monroy<sup>3</sup> Oscar E. García-Quintero<sup>3</sup> Carlos Hidalgo-Toscano<sup>4</sup>

<sup>1</sup>Institute for Software Technology, Graz University of Technology, Graz, Austria

<sup>2</sup>Instituto de Matemáticas, Universidad de Antioquia, Colombia

<sup>3</sup>Departamento de Matemáticas, CINVESTAV, Mexico

<sup>4</sup>INFOTEC Centro de Investigación e Innovación en Tecnologías de la Información y Comunicación, Mexico

### Abstract

A drawing of a graph in the plane is *pseudolinear* if the edges of the drawing can be extended to doubly-infinite curves that form an arrangement of pseudolines, that is, any pair of these curves crosses precisely once. A special case is *rectilinear* drawings where the edges of the graph are drawn as straight line segments. The rectilinear (pseudolinear) crossing number of a graph is the minimum number of pairs of edges of the graph that cross in any of its rectilinear (pseudolinear) drawings. In this paper we describe an ongoing project to continuously obtain better asymptotic upper bounds on the rectilinear and pseudolinear crossing number of the complete graph  $K_n$ .

Submitted: July 2019	Reviewed: January 2020	Revised: February 2020	Accepted: July 2020
	Final: July 2020	Published: July 2020	
Article type: Regular paper		Communicated by: William Evans	



This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 734922.

Additionally, Duque, Fabila-Monroy, García-Quintero and Hidalgo-Toscano were partially supported by Conacyt of Mexico grant 253261

*E-mail addresses:* oaich@ist.tugraz.at (Oswin Aichholzer) rodrigo.duque@udea.edu.co (Frank Duque) ruyfabila@math.cinvestav.edu.mx (Ruy Fabila-Monroy) opraupe0017@gmail.com (Oscar E. García-Quintero) carlos.hidalgo@infotec.mx (Carlos Hidalgo-Toscano)

## 1 Introduction

Let  $G$  be a graph. In a drawing  $D$  of  $G$  in the plane the vertices of  $G$  are distinct points in the plane and every edge of  $G$  is represented by a simple closed polygonal chain connecting the two points representing its vertices. We require that in  $D$  the edges do not pass through a point representing a vertex (other than their endpoints). A crossing in  $D$  is a proper intersection of the interior of two line segments in a pair of polygonal chains representing two edges. Obtaining a drawing of a given graph  $G$  that minimizes the number of crossings among all possible drawings of  $G$  is a central topic in many areas like graph drawing and discrete and computational geometry. Depending on how drawings are defined and on the precise way crossings are counted, there exist a huge number of variants of crossing number problems; see [15] for a comprehensive survey. In this work we concentrate on rectilinear and pseudolinear drawings of the complete graph  $K_n$ .

The *crossing number* of  $G$  is the minimum number,  $\text{cr}(G)$ , of pairs of edges of  $G$  that cross in any drawing of  $G$ . A set of points in the plane is in *general position* if no three of its points are collinear. A *rectilinear drawing* of a graph  $G$  is a drawing of  $G$  in the plane where its vertices (points) are in general position and its edges are drawn as straight line segments. The *rectilinear crossing number* of  $G$  is the minimum number,  $\overline{\text{cr}}(G)$ , of pairs of edges of  $G$  that cross in any rectilinear drawing of  $G$ .

A generalization of rectilinear drawings are pseudolinear drawings. A drawing of a graph in the plane is *pseudolinear* if the edges of the drawing can be extended to doubly-infinite curves that form an arrangement of pseudolines, that is, any pair of these curves crosses precisely once. Similar as before, the *pseudolinear crossing number* of a graph is the minimum number,  $\tilde{\text{cr}}(G)$ , of pairs of edges of  $G$  that cross in any pseudolinear drawing of  $G$ . Clearly,  $\text{cr}(G) \leq \tilde{\text{cr}}(G) \leq \overline{\text{cr}}(G)$ .

Finding the crossing number, rectilinear crossing number or pseudolinear crossing number of the complete graph  $K_n$  are important open problems in discrete geometry. For the crossing number of  $K_n$  Harary and Hill conjectured the following.

**Conjecture 1 (Harary-Hill [12])**

$$\text{cr}(K_n) = \frac{1}{4} \left\lfloor \frac{n}{2} \right\rfloor \left\lfloor \frac{n-1}{2} \right\rfloor \left\lfloor \frac{n-2}{2} \right\rfloor \left\lfloor \frac{n-3}{2} \right\rfloor.$$

According to [12] drawings achieving this bound had been found independently and Guy published them for the first time in [11]. The lower bounds on  $\overline{\text{cr}}(K_n)$  and  $\tilde{\text{cr}}(K_n)$ , and the upper bound on  $\text{cr}(K_n)$  together imply that for sufficiently large  $n$  we have that  $\overline{\text{cr}}(K_n) \geq \tilde{\text{cr}}(K_n) > \text{cr}(K_n)$ . However, it is unknown whether  $\overline{\text{cr}}(K_n) > \tilde{\text{cr}}(K_n)$  (for some sufficiently large  $n$ ). See [4] for a nice survey. When bounding  $\overline{\text{cr}}(K_n)$  and  $\tilde{\text{cr}}(K_n)$  these numbers are often considered together, see e.g. [3]; since  $\tilde{\text{cr}}(K_n) \leq \overline{\text{cr}}(K_n)$ , an upper bound on  $\overline{\text{cr}}(K_n)$  is also an upper bound on  $\tilde{\text{cr}}(K_n)$ , and similar for lower bounds.

It is known that

$$\lim_{n \rightarrow \infty} \frac{\overline{\text{cr}}(K_n)}{\binom{n}{4}} = q^*,$$

where  $q^*$  is a positive constant known as the *rectilinear crossing constant*. Similarly, we have

$$\lim_{n \rightarrow \infty} \frac{\widetilde{\text{cr}}(K_n)}{\binom{n}{4}} = \widetilde{q}^*,$$

where  $\widetilde{q}^*$  is a positive constant, called the *pseudolinear crossing constant*. As it is unknown whether  $\overline{\text{cr}}(K_n) > \widetilde{\text{cr}}(K_n)$  for sufficiently large  $n$ , it is a challenging open problem whether  $q^* > \widetilde{q}^*$ .

The currently best lower bounds [3] are

$$0.379972 < \widetilde{q}^* \leq q^*$$

and the previously best upper bounds [6, 9] are

$$\widetilde{q}^* < 0.380448 \quad q^* < 0.380473.$$

In this note we describe an ongoing project to improve these upper bounds on  $q^*$  and  $\widetilde{q}^*$ . The best upper bounds we have obtained so far<sup>1</sup> are

$$q^* \leq \frac{43317371729896}{113858494707069} < 0.3804491869$$

and

$$\widetilde{q}^* \leq \frac{5995534434121}{15759524733750} < 0.3804387846.$$

The arXiv version of this paper [arXiv:1907.07796] will be regularly updated with new upper bounds.

## 1.1 Constructing Good Drawings

To derive upper bounds on  $q^*$  one way is to produce rectilinear drawings of  $K_n$  with few crossings for arbitrarily large values of  $n$ . The first general construction for such sets was given by Jensen [14] who gave explicit coordinates for the points of  $K_n$ . Around the same time Singer [16] proposed another approach. His construction takes a drawing of  $K_n$  and produces a drawing of  $K_{3n}$ . If the former drawing has few crossings, then so does the latter. Using this drawing of  $K_{3n}$  and repeating the process gives a good drawing of  $K_{9n}$ , and so on. This approach of iteratively generating larger sets has been successful in improving the upper bound on  $q^*$  several times, see [7, 5, 2, 1]. The current best iterative construction is that of Ábrego, Cetina Fernández-Merchant, Leaños and Salazar ([2] and [1]).

A feature of these iterative constructions is that to improve the upper bound on  $q^*$  it is sufficient to find for a specific, constant value of  $n$ , a sufficiently good rectilinear drawing of  $K_n$ . Fabila and López [9] used an heuristic to improve the

---

<sup>1</sup>As of 2020-07-15.

best known rectilinear drawing of  $K_{75}$ , obtaining a rectilinear drawing of  $K_{75}$  with 450492 crossings. This rectilinear drawing together with the construction of [2, 1] provides the until now best upper bound of  $q^* < 0.380473$ .

Up to and including the work of Fabila and López [9] upper bounds for  $\tilde{q}^*$  were simply derived from the upper bounds of  $q^*$ , that is,  $\tilde{q}^* \leq q^* < 0.380473$ . Only recently Balko and Kynčl [6] obtained the first upper bound on  $\tilde{q}^*$  which is below the upper bound of  $q^*$  by showing  $\tilde{q}^* < 0.380448$ . The method they used is a generalization of [9] to pseudolinear drawings. They give a nice presentation of pseudolinear drawings with  $n$ -signatures<sup>2</sup> and show that the construction of [2, 1] can be adopted to the pseudolinear setting; see below for details.

Our goal is to be able to improve the upper bounds on the crossing constants in a semi-automatic way. We thus implemented the construction of [2, 1] and also its extension to pseudolinear drawings [6], as well as various heuristics to improve a given rectilinear or pseudolinear drawing of  $K_n$ . In this paper we describe our approach in detail. In Section 1.2 we briefly recall the constructions of [2] and [1] and the modification by [6]. In Section 2 we describe several heuristics we used to improve known rectilinear and pseudolinear drawings of  $K_n$ . In Section 3 we describe how these tools play together in an iterative way to regularly obtain better upper bounds on  $q^*$  and  $\tilde{q}^*$ .

## 1.2 The Construction of [2], [1], and [6]

Note that a rectilinear drawing of  $K_n$  is determined by the position of its vertices. Let  $S := \{p_1, \dots, p_n\}$  be a set of  $n$  points in general position in the plane. Let  $\overline{\text{cr}}(S)$  be the number of crossings in the rectilinear drawing of  $K_n$  where the vertices are placed at  $S$ .

A *halving line* of  $S$  is a straight line  $\ell$  passing through at least one point of  $S$  such that in the two open half-planes defined by  $\ell$  there are the same number of points of  $S$ . Note that if  $n$  is odd then  $\ell$  passes only through a single point of  $S$ , and if  $n$  is even then  $\ell$  passes through two points of  $S$ .

Let  $G$  be the bipartite graph with vertex partition  $(A, B)$ , where  $A := S$  and  $B$  is the set of the halving lines of  $S$ . A pair  $(p, \ell)$  in  $(A, B)$  is adjacent in  $G$  if and only if  $\ell$  passes through  $p$ . A *halving matching* of  $S$  is a matching of  $G$ , in which every point in  $A$  is matched to a halving line in  $B$ . If  $n$  is odd then a halving matching of  $S$  always exists. In this case every halving line contains exactly one point of  $S$  and every point of  $S$  is contained in at least one halving line (actually an infinite number of them). So any choice of a halving line for every point of  $A$  is a halving matching. A halving matching may not exist if  $n$  is even. For example, a set of 4 points in which exactly three of them are in the convex hull does not contain a halving matching.

A rough description of the construction of [2] and [1] is as follows. Let  $M = \{(p_1, \ell_1), \dots, (p_n, \ell_n)\}$  be a halving matching for  $S$ . For every  $\ell_i$  assume that  $\ell_i$  is directed, and let  $\vec{v}_i$  be the direction vector of  $\ell_i$ . Let  $S'$  be the point set that results by replacing each  $p_i$  in  $S$  with the pair of points  $p_i + \varepsilon\vec{v}_i$  and

<sup>2</sup>An  $n$ -signature is a function which assigns to every triple of vertices an orientation  $\{+, -\}$ .

$p_i - \varepsilon \vec{v}_i$  for an arbitrarily small (but positive) value of  $\varepsilon$ .

This construction was first described in [2]; furthermore, they showed that if  $S$  has an even number of points then  $S'$  also has a halving matching. This allows for the iterative construction mentioned above. If  $S$  has an odd number of points then it always has a halving matching; however, it is not obvious that  $S'$  should also have a halving matching. In [1] they showed that in this case  $S'$  also has a halving matching. The following theorem gives the upper bound on  $q^*$  that is derived from this construction.

**Theorem 1 ([1, 2])** *Let  $S$  be a set of  $n$  points in general position in the plane, such that  $S$  has a halving matching. Then*

$$q^* \leq \frac{24\overline{\text{cr}}(S) + 3n^3 - 7n^2 + (30/7)n}{n^4}.$$

The above theorem is derived by showing that in every doubling step from  $S$  to  $S'$  it holds that  $\overline{\text{cr}}(S') = 16\overline{\text{cr}}(S) + (n/2)(2n^2 - 7n + 5)$ . It is not hard to see that the concept of halving matching also translates to pseudolinear drawings. Consequently it is shown in [6] that the construction of doubling can be applied to pseudolinear drawings and that the halving matching property is also preserved. Thus, a similar relation holds when doubling a pseudolinear drawing  $D$  to  $D'$ , namely  $\tilde{\text{cr}}(D') = 16\tilde{\text{cr}}(D) + 2n(\lceil n/2 \rceil^2 + \lfloor n/2 \rfloor^2) - 7n^2/2 + 5n/2$  [6]. For even  $n$  this is the same bound as for the rectilinear case. Although not explicitly stated in [6] this therefore leads to the following bound for  $\tilde{q}^*$ .

**Corollary 2** *Let  $D$  be a pseudolinear drawing of  $K_n$ , such that  $D$  has a halving matching. Then*

$$\tilde{q}^* \leq \frac{24\tilde{\text{cr}}(D) + 3n^3 - 7n^2 + (30/7)n}{n^4} \text{ for } n \text{ even; and}$$

$$\tilde{q}^* \leq \frac{24\tilde{\text{cr}}(D) + 3n^3 - 7n^2 + (81/14)n}{n^4} \text{ for } n \text{ odd.}$$

While the above two results will be directly used to derive our improved upper bounds, we are also interested in actually obtaining the good sets of doubled cardinality; see also Section 3.

For rectilinear drawings the above construction has been implemented as a Python program in [10] and produces point sets with integer coordinates. The choice of using integer coordinates helps to avoid numerical issues, since as long as we use a library (or programming language) that handles arbitrarily large integers, arithmetic precision is not an issue. We therefore used the implementation of [10] also for our computations.

For pseudolinear drawings no numerical issues arise, as no realization with points is needed and thus no coordinates are computed. We follow the lines of [6] and use a representation of a pseudolinear drawing by  $n$ -signatures. This is a function which assigns to every triple of vertices an orientation  $\{+, -\}$ . The advantage of this representation is that it is easy to check if a given  $n$ -signature

can be realized as a pseudolinear drawing and that the number of crossings can be computed directly from the representation. Moreover, the above mentioned results for the doubling construction have been derived; see again [6] for more details. We implemented this approach also as a Python program to generate good pseudolinear drawings of twice the size.

## 2 Heuristics

We now describe various heuristics that attempt to improve a given drawing of  $K_n$ . For a rectilinear drawing the vertices are the points of the point set  $S$  and  $p$  is a point of  $S$ . For a pseudolinear drawing  $D$  all triple orientations  $v_i v_j v_k$ ,  $1 \leq i < j < k \leq n$  are given.

### 2.1 Moving a Point to a New Random Location

In [9] the following simple heuristic is used for rectilinear drawings. Choose a random point  $q \notin S$  in a certain neighborhood of  $p$ . If  $\overline{\text{cr}}(S \setminus \{p\} \cup \{q\}) \leq \overline{\text{cr}}(S)$  then replace  $p$  with  $q$  in  $S$ . In [9],  $\overline{\text{cr}}(S \setminus \{p\} \cup \{q\})$  is computed in  $O(n^2)$  time. If after some time no improvement is found then the size of the neighborhood of  $p$  from which  $q$  is chosen, is made smaller. We use the amortized faster algorithm of [8]. In that algorithm a set  $Q$  of  $\Theta(n)$  candidate points for the new position of  $p$  is chosen. The set of values

$$\{\overline{\text{cr}}(S \setminus \{p\} \cup \{q\}) : q \in Q\}$$

is computed in  $O(n^2)$  time. Note that this is linear per point in  $Q$ . We then consider the best point in  $Q$  as a possible replacement for  $p$ .

For pseudolinear drawings a very similar idea is used. Obviously there are no points to be moved. But observe that moving a point continuously in the rectilinear case corresponds to changing the orientation of point triples one after another (since there are no collinear points), that is, changing the order type of the set step by step. The difficulty in the rectilinear setting is that this is a geometric process, and even if all points have integer coordinates, moving the point on an integer grid might not be sufficient to get all triple changes separately.

To the contrary, changing triple orientations in a pseudolinear drawing given as  $n$ -signature is trivial. We choose a random vertex triple  $v_i v_j v_k$ ,  $1 \leq i < j < k \leq n$ , of the drawing  $D$  and invert its orientation. Then we check if this new  $n$ -signature is still realizable as a pseudolinear drawing  $D'$ . In case  $D'$  exists, we keep the new orientation if  $\tilde{\text{cr}}(D') \leq \tilde{\text{cr}}(D)$  and set  $D = D'$ . Otherwise we revert the change. Iterating this process eventually gives pseudolinear drawings with less crossings.

It is interesting to observe that our experiments show that this local optimization heuristic works in average significantly better for pseudolinear drawings than for rectilinear drawings. The main reason might be that for pseudolinear drawings the algorithm has a combinatorial flavor, while for rectilinear drawings

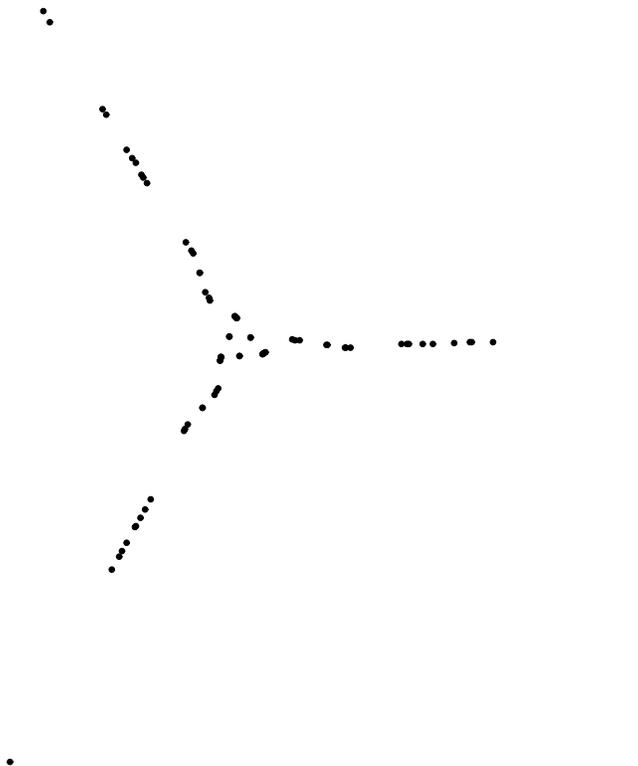


Figure 1: A drawing of  $K_{75}$  with 450492 crossings

the precise geometry of the arrangement of the point set (e.g. the size of cells) plays a role. We therefore present in the next section an approach for rectilinear drawings which avoids this bottleneck.

## 2.2 The Point Set Explorer

Consider the line arrangement,  $\mathcal{A}$ , spanned by the set of lines passing through every pair of points of  $S \setminus \{p\}$ . Let  $C$  be a cell of  $\mathcal{A}$  and let  $q$  be a point in  $C$ . Note that  $\overline{\text{co}}(S \setminus \{p\} \cup \{q\})$  has the same value regardless of the choice of the precise location of  $q$  within  $C$ .

All the best known rectilinear drawings of  $K_n$  consist of three “arms” with close to  $n/3$  points each; the points in each arm are close to being collinear. See Figure 1 for an example. This implies that many of the cells in the line arrangements of such sets have area close to zero. Therefore, in the heuristic described in 2.1 many of the candidate points fall in the same cells and many cells are never visited.

In [13] the following algorithm is implemented. The algorithm computes

the cell of  $\mathcal{A}$  that contains  $p$  in  $O(n^2 \log n)$  time. Afterwards, moving between adjacent cells, and finding a point in the interior of these cells, takes  $O(\log^2 n)$  time.

Using this implementation we produce a sequence  $C = C_1, \dots, C_m$  of consecutive adjacent cells. At each  $C_i$  we choose a point  $q_i$  as a candidate to replace  $p$  in  $S$ . Since we are moving between adjacent cells,  $\overline{\text{cr}}(S \setminus \{p\} \cup \{q_i\})$  can actually be computed in  $O(1)$  time.

In this way the size of the cells in the arrangement and thus the precise geometry of our point set does not play a role anymore. We obtain an algorithm with a more combinatorial behavior than with the previous heuristic.

### 2.3 Finding Good Subdrawings

It is often the case that for  $m > n$ , drawings of  $K_m$  with few crossings contain drawings of  $K_n$  with few crossings. We observed that when optimizing drawings of  $K_m$  with the heuristics presented in the previous section, they often contain better subdrawings of  $K_n$  than before the optimization. In other words, it is possible to improve drawings of  $K_n$  by optimizing drawings of  $K_m$ . It might sound counter intuitive to optimize the larger set. But observe that there are many subsets of size  $n$  in a set of size  $m$ . So if just one of these subdrawings is improved, this approach is successful.

Given a drawing of  $K_m$ , we use the following heuristics to find drawings of  $K_n$  with few crossings.

#### Removing one point at a time

For rectilinear drawings we remove one point of  $S$  at a time, to obtain smaller drawings. We use an implementation of an algorithm described in [8] that does the following. In  $O(n^2)$  time, it computes the set of values

$$\{\overline{\text{cr}}(S \setminus \{p\}) : p \in S\}.$$

We remove the point  $p$  that minimizes  $\overline{\text{cr}}(S \setminus \{p\})$ , thus finding a rectilinear drawing of  $K_{m-1}$ . We proceed iteratively in this way to find rectilinear drawings of  $K_n$  with  $n = m - 1, m - 2, \dots, 27$  points.

For pseudolinear drawings this is done in a similar way.

#### Removing two or more points at a time

It is known that point sets minimizing the rectilinear crossing number have a triangular convex hull, and most of them have several layers consisting of three vertices each. Moreover, during the process of generating good examples to get an improved crossing constant we could observe that the cardinality of the best sets seems to be a multiple of 3. So actually it turned out that sometimes it is better to remove more than one point at each step. That means we look at a tuple or triple of points which, when being removed at once, reduces the crossing number the most. Occasionally, this provides rectilinear drawings that

cannot be obtained by removing one point at a time as in the previous heuristic. However, the amortized speed up described above is lost, and thus this method needs more computational resources.

To our own surprise good pseudolinear drawings show precisely the same behavior. There also the best drawings (w.r.t. the obtainable pseudolinear crossing constant) we found so far have a cardinality which is a multiple of 3. We therefore use the same approach of removing a tuple or triple of vertices at the same time also for pseudolinear drawings.

### 3 Joining Everything Together

Several computers run all the heuristics described in Section 2 permanently in the background. Processes are started for the best drawings of  $K_n$  for some specific values of  $n$  - typically the currently best three or four cardinalities for both types of drawings, rectilinear and pseudolinear. The newly-found drawings are sent daily to a central node. If after for some time, no new improvements are made on the upper bound of either  $q^*$  or  $\tilde{q}^*$ , we take the corresponding drawing of  $K_n$  providing the best upper bound and apply the doubling construction described in Section 1.2 to obtain a drawing of  $K_{2n}$  providing the same upper bound on  $q^*$  or  $\tilde{q}^*$ , respectively. From this set we then compute good subsets by removing points/vertices as described in Section 2.3. In parallel we locally optimize the new drawing  $K_{2n}$  and all the obtained subdrawings by the methods described in Sections 2.1 and 2.2. The steps of local optimization and computing subdrawings are interleaved and iterated. Note, however, that the local optimization is applied only for a limited time. The best sets obtained in these iterations serve as new starting sets of our optimization, and the whole process is restarted.

#### 3.1 Results for Rectilinear Drawings

With the just described approach we were able to obtain good rectilinear drawings of  $K_n$  for up to  $n = 3240$  points. The best crossing constant is obtained by using a rectilinear drawing of  $K_{2643}$  which has 771218714414 crossings. The interested reader can download the file with the coordinates of the points from <http://www.crossingnumbers.org/projects/crossingconstants/rectilinear.php>. The reason why larger sets do not necessarily provide a better constant lies in the before mentioned search for good subsets, which is an essential ingredient of our mixed heuristic. Plugging the values of this drawing into Theorem 1 we obtain the following result.

**Theorem 3**

$$q^* \leq \frac{43317371729896}{113858494707069} < 0.3804491869$$

### 3.2 Results for Pseudolinear Drawings

For pseudolinear drawings we have been able to get good results for  $K_n$  for up to  $n = 2502$ . The reason why the maximal cardinality is smaller than the one for rectilinear drawings is that here we have to store every triple orientation explicitly, which needs several hundreds MB for each set. The best crossing constant is obtained by using a pseudolinear drawing of  $K_{2205}$  which has 373382224051 crossings. The interested reader can download the file with the information of the triple orientation from <http://www.crossingnumbers.org/projects/crossingconstants/pseudolinear.php>. Plugging the values of this drawing into Corollary 2 we obtain the following result.

**Theorem 4**

$$\tilde{q}^* \leq \frac{5995534434121}{15759524733750} < 0.3804387846.$$

*The rectilinear drawing of  $K_{2643}$  with 771218714414 crossings can be found in the arXiv version of this paper.*

## References

- [1] B. M. Ábrego, M. Cetina, S. Fernández-Merchant, J. Leños, and G. Salazar. 3-symmetric and 3-decomposable geometric drawings of  $K_n$ . *Discrete Applied Mathematics*, 158(12):1240–1258, 2010. Traces from LA-GOS’07 IV Latin American Algorithms, Graphs, and Optimization Symposium Puerto Varas - 2007. doi:10.1016/j.dam.2009.09.020.
- [2] B. M. Ábrego and S. Fernández-Merchant. Geometric drawings of  $K_n$  with few crossings. *J. Combin. Theory Ser. A*, 114(2):373–379, 2007. doi:10.1016/j.jcta.2006.05.003.
- [3] B. M. Ábrego, S. Fernández-Merchant, J. Leños, and G. Salazar. A central approach to bound the number of crossings in a generalized configuration. *Electronic Notes in Discrete Mathematics*, 30(0):273–278, 2008. The IV Latin-American Algorithms, Graphs, and Optimization Symposium. doi:10.1016/j.endm.2008.01.047.
- [4] B. M. Ábrego, S. Fernández-Merchant, and G. Salazar. The rectilinear crossing number of  $K_n$ : Closing in (or are we?). In J. Pach, editor, *Thirty Essays on Geometric Graph Theory*, pages 5–18. Springer New York, 2013. doi:10.1007/978-1-4614-0110-0\_2.
- [5] O. Aichholzer, F. Aurenhammer, and H. Krasser. On the crossing number of complete graphs. *Computing*, 76(1-2):165–176, 2006. doi:10.1007/s00607-005-0133-3.
- [6] M. Balko and J. Kynčl. Bounding the pseudolinear crossing number of  $K_n$  via simulated annealing. In *Abstracts XVI Spanish Meeting on Computational Geometry*, pages 37–40, 2015.
- [7] A. Brodsky, S. Durocher, and E. Gethner. Toward the rectilinear crossing number of  $K_n$ : new drawings, upper bounds, and asymptotics. *Discrete Math.*, 262(1-3):59–77, 2003. doi:10.1016/S0012-365X(02)00491-0.
- [8] F. Duque and R. Fabila-Monroy. Updating the Number of Crossings in Rectilinear Drawings of the Complete Graph. *ArXiv e-prints*, Sept. 2016. arXiv:1609.00867.
- [9] R. Fabila-Monroy and J. López. Computational search of small point sets with small rectilinear crossing number. *Journal of Graph Algorithms and Applications*, 18(3):393–399, 2014. doi:10.7155/jgaa.00328.
- [10] O. E. García-Quintero. Duplicando conjuntos en posición general con bajo número de cruce. Master’s thesis, Departamento de Matemáticas, Cinvestav, 2016. In Spanish.
- [11] R. Guy. A combinatorial problem. *Bull. Malayan Math. Soc.*, 7:68–72, 1960.

- [12] F. Harary and A. Hill. On the number of crossings in a complete graph. *Proc. Edinburgh Math. Soc. (2)*, 13:333–338, 1962/63. doi:10.1017/S0013091500025645.
- [13] C. Hidalgo-Toscano. Un algoritmo para recorrer las celdas de un arreglo de rectas. Master’s thesis, Departamento de Matemáticas, Cinvestav, 2015. In Spanish.
- [14] H. F. Jensen. An upper bound for the rectilinear crossing number of the complete graph. *J. Combinatorial Theory Ser. B*, 10:212–216, 1971.
- [15] M. Schaefer. The graph crossing number and its variants: a survey. *The Electronic Journal of Combinatorics*, Dynamic Survey 21, 2013/2017.
- [16] D. Singer. Rectilinear crossing numbers. Manuscript, 1971.