Small-Area Orthogonal Drawings of 3-Connected Graphs

Therese Biedl\textsuperscript{1} and Jens M. Schmidt\textsuperscript{2(EG)}

\textsuperscript{1} David R. Cheriton School of Computer Science, University of Waterloo, Waterloo, Canada
\textsuperscript{2} Institute of Mathematics, TU Ilmenau, Ilmenau, Germany
\texttt{jens.schmidt@tu-ilmenau.de}

Abstract. It is well-known that every graph with maximum degree 4 has an orthogonal drawing with area at most \(\frac{49}{64} n^2 + O(n) \approx 0.76n^2\). In this paper, we show that if the graph is 3-connected, then the area can be reduced even further to \(\frac{9}{16} n^2 + O(n) \approx 0.56n^2\). The drawing uses the 3-canonical order for (not necessarily planar) 3-connected graphs, which is a special Mondshein sequence and can hence be computed in linear time. To our knowledge, this is the first application of a Mondshein sequence in graph drawing.

1 Introduction

An orthogonal drawing of a graph \(G = (V, E)\) is an assignment of vertices to points and edges to polygonal lines connecting their endpoints such that all edge-segments are horizontal or vertical. Edges are allowed to intersect, but only in single points that are not bends of the polygonal lines. Such an orthogonal drawing can exist only if every vertex has degree at most 4; we call such a graph a 4-graph. It is easy to see that every 4-graph has an orthogonal drawing with area \(O(n^2)\), and this is asymptotically optimal [17].

For planar 2-connected graphs, several authors showed independently [10,15] how to achieve area \(n \times n\), and this is optimal [16]. We measure the drawing-size as follows. Assume (as we do throughout the paper) that all vertices and bends are at points with integral coordinates. If \(H\) rows and \(W\) columns of the integer grid intersect the drawing, then we say that the drawing occupies a \(W \times H\)-grid with width \(W\), height \(H\), half-perimeter \(H + W\) and area \(H \cdot W\). Some papers count as width/height the width/height of the smallest enclosing axis-aligned box. This is one unit less than with our measure.

For arbitrary graphs (i.e., graphs that are not necessarily planar), improved bounds on the area of orthogonal drawings were developed much later, decreasing from \(4n^2\) [11] to \(n^2\) [1] to 0.76\(n^2\) [9]. (In all these statements, we omit lower-order terms for ease of notation.)
Our Results: In this paper, we decrease the area-bound for orthogonal drawings further to $0.56n^2 + O(n)$ under the assumption that the graph is 3-connected. The approach is similar to the one by Papakostas and Tollis [9]: add vertices to the drawing in a specific order, and pair some of these vertices so that in each pair one vertex re-uses a row or column that was used by the other. The main difference in our paper is that 3-connectivity allows the use of a different, stronger, vertex order.

It has been known for a long time that any planar 3-connected graph has a so-called canonical order [7], which is useful for planar graph drawing algorithms. It was mentioned that such a canonical order also exists in non-planar graphs (e.g. in [4, Remark on p.113]), but it was not clear how to find it efficiently, and it has to our knowledge not been used for graph drawing algorithms. Recently, the second author studied the so-called Mondshein sequence, which is an edge partition of a 3-connected graph with special properties [8], and showed that it can be computed in linear time [13]. A Mondshein sequence is the appropriate generalization of the canonical order to (not necessarily planar) 3-connected graphs [13] and is most naturally defined by ear decompositions. However, in order to highlight its relation to canonical orders, we define a Mondshein sequence here as a special vertex partition and call it a 3-canonical order.

We use this 3-canonical order to add vertices to the orthogonal drawing. This almost immediately lowers the resulting area, because vertices with one incoming edge can only occur in chains. We then mimic the pairing-technique of Papakostas and Tollis, and pair groups of the 3-canonical order in such a way that even more rows and columns can be saved, resulting in a half-perimeter of $3\frac{3}{2}n + O(1)$ and the area-bound follows.

No previous algorithms were known that achieve smaller area for 3-connected 4-graphs than for 2-connected 4-graphs. For planar graphs, the orthogonal drawing algorithm by Kant [7] draws 3-connected planar 4-graphs with area $(\frac{2}{3}n)^2 + O(n)$ [14], while the best-possible area for planar 2-connected graphs is $n^2$ [16].

2 Preliminaries

Let $G = (V, E)$ be a graph with $n = |V|$ vertices and $m = |E|$ edges. The degree of a vertex $v$ is the number of incident edges. In this paper, all graphs are assumed to be 4-graphs, i.e., all vertex degrees are at most 4. A graph is called 4-regular if every vertex has degree exactly 4; such a graph has $m = 2n$ edges.

A graph $G$ is called connected if, for any two vertices $u, v$, there is a path in $G$ connecting $u$ and $v$. It is called 3-connected if $n > 3$ and, for any two vertices $u, v$, the graph $G - \{u, v\}$ is connected.

A loop is an edge $(v, v)$ that connects an endpoint with itself. A multi-edge is an edge $(u, v)$ for which another copy of edge $(u, v)$ exists. When not otherwise stated, the graph $G$ that we want to draw is simple, i.e., it has neither loops nor multi-edges. While modifying $G$, we will sometimes temporarily add a double edge, i.e., an edge for which exactly one other copy exists (we refer always to the added edge as double edge, the copy is not a double edge).
2.1 The 3-Canonical Order

Definition 1. Let $G$ be a 3-connected graph. A 3-canonical order (or Mondshein sequence) is a partition of $V$ into groups $V = V_1 \cup \cdots \cup V_k$ such that

- $V_1 = \{v_1, v_2\}$, where $(v_1, v_2)$ is an edge.
- $V_k = \{v_n\}$, where $(v_1, v_n)$ is an edge.
- For any $1 < i < k$, one of the following holds:
  • $V_i = \{z\}$, where $z$ has at least two predecessors and at least one successor.
  • $V_i = \{z_1, \ldots, z_\ell\}$ for some $\ell \geq 2$, where
    - $z_1, \ldots, z_\ell$ is an induced path in $G$ (i.e. edges $z_1 - z_2 - \cdots - z_\ell$ exist, and there are no edges $(z_i, z_j)$ with $i < j - 1$),
    - $z_1$ and $z_\ell$ have exactly one predecessor each, and these predecessors are different,
    - $z_j$ for $1 < j < \ell$ has no predecessor,
    - $z_j \in V_i$ for $1 \leq j \leq \ell$ has at least one successor.

Here, a predecessor [successor] of a vertex in $V_i$ is a neighbor that occurs in a group $V_h$ with $h < i$ [$h > i$]. See Fig. 1 for a 3-canonical order.

We call a vertex group $V_i$ a singleton if $|V_i| = 1$, and a chain if $|V_i| \geq 2$ and $i \geq 2$. We distinguish chains further into short chains with $|V_i| = 2$ and long chains with $|V_i| \geq 3$. A 3-canonical order imposes a natural orientation on the edges of the graph from lower-indexed groups to higher-indexed groups and, for edges within a chain, from one (arbitrary) end of the path to the other. This implies in-degree $\text{indeg}(v) \geq 2$ for any singleton, $\text{indeg}(v) = 2$ for exactly one vertex of each chain, and $\text{indeg}(v) = 1$ for all other vertices of a chain.

Fig. 1. A 4-regular 3-connected graph with a 3-canonical order, and the drawing created with our algorithm. For illustrative purposes, we show the drawing exactly as created, even though many more grid lines and bends could be saved with straightforward compaction steps. $V_2$ is a long chain, $V_4$ is a short chain, $V_5$ is a 2-2-singleton, $V_3, V_6, V_7$ and $V_8$ are 3-1-singletons.
Numerous related methods of ordering vertices of 3-connected graphs exist, e.g. $(2,1)$-sequences [8], non-separating ear decompositions [2,13], and, limited to planar graphs, canonical orders for maximal planar graphs [6], canonical orders for 3-connected planar graphs [7] and orderly spanning trees [3]. A Mondshein sequence (i.e. a 3-canonical order) of a 3-connected graph implies all these orders, up to minor subtleties.

The most efficient way known to compute a Mondshein sequence (proving in particular that one exists) uses non-separating ear decompositions [2,13]. This is a partition of the edges into ears $P_1 \cup \cdots \cup P_k = E$ such that $P_1$ is an induced cycle, $P_i$ for $i > 1$ is a non-empty induced path that intersects $P_1 \cup \cdots \cup P_{i-1}$ in exactly its endpoints, and $G - \bigcup_{j=1}^{i} P_j$ is connected for every $i < k$. Such a non-separating ear decomposition exists for any 3-connected graph [2], and we can even fix two edges $v_1v_2$ and $v_2v_n$ and require that $v_1v_2$ is in the cycle $P_1$ and that $v_n$ is the only vertex in $P_k$; hence, $P_k$ will be a singleton.

Further, such a non-separating ear decomposition can be computed in linear time [12,13]. The sets of newly added vertices for each $P_i$ will be the vertex groups of a 3-canonical order (additionally, $P_1$ is split into the groups $V_1 := \{v_1,v_2\}$ and $V_2 := V(P_1) - \{v_1,v_2\}$). Although vertices in $V_i$ may have arbitrarily many incident edges in a non-separating ear decomposition, we can easily get rid of these extra edges by a simple short-cutting routine in linear time (see Lemmas 8 and 12 in [12]). This gives a 3-canonical order. Thus, a linear-time algorithm for computing a 3-canonical order follows immediately from the one for non-separating ear decompositions.

2.2 Making 3-Connected 4-Graphs 4-Regular

It will greatly simplify the description of the algorithm if we only give it for 4-regular graphs. Thus, we want to modify a 3-connected 4-graph $G$ such that the resulting graph $G'$ is 4-regular, draw $G'$, and then delete added edges to obtain a drawing of $G$. However, we must maintain a simple graph since the existence of 3-canonical orders depends on simplicity. This turns out to be impossible (e.g. for the graph obtained from the octahedron by subdividing two distinct edges with a new vertex and joining the new vertices by an edge), but allowing one double edge is sufficient.

Lemma 2. Let $G$ be a simple 3-connected 4-graph with $n \geq 5$. Then we can add edges to $G'$ such that the resulting graph $G'$ is 3-connected, 4-regular, and has at most one double edge.

Proof. Since $G$ is 3-connected, any vertex has degree 3 or 4. If there are four or more vertices of degree 3, then they cannot be mutually adjacent (otherwise $G = K_4$, which contradicts $n \geq 5$). Hence, we can add an edge between two non-adjacent vertices of degree 3; this maintains simplicity and 3-connectivity.

We repeat until only two vertices of degree 3 are left (recall that the number of vertices of odd degree is even). Now an edge between these two vertices is added, even if one existed already; this edge is the only one that may become a double edge. The resulting graph is 4-regular and satisfies all conditions. □
3 Creating Orthogonal Drawings

From now on, let $G$ be a 3-connected 4-regular graph that has no loops and at most one double edge. Compute a 3-canonical order $V = V_1 \cup \cdots \cup V_k$ of $G$ with $V_k = \{v_n\}$, choosing $v_1v_n$ to be the double edge if there is one. Let $x^\text{short}$ and $x^\text{long}$ be the number of short and long chains. Let $x^{j-\ell}$ be the number of vertices with in-degree $j$ and out-degree $\ell$. Since $G$ is 4-regular, we must have $j + \ell = 4$.

A $j,\ell$-singleton is a vertex $z$ that constitutes a singleton group $V_i$ for $1 < i \leq k$ and that has in-degree $j$ and out-degree $\ell$.

**Observation 3.** Let $G$ be a 4-regular graph with a 3-canonical order. Then

1. $x^{0-4} = x^{4-0} = 1$
2. $x^{1-3} = x^{3-1}$
3. Every chain $V_i$ contributes one to $x^{2-2}$ and $|V_i| - 1$ to $x^{1-3}$.

**Proof.** (1) holds, since every vertex that is different from $v_n$ has a successor, and every vertex that is different from $v_1$ has an incoming edge from either a predecessor or within its chain. For (2), observe that $2n = m = \sum_v \text{indeg}(v) = x^{1-3} + 2x^{2-2} + 3x^{3-1} + 4x^{4-0}$ and $n = x^{0-4} + x^{1-3} + x^{2-2} + x^{3-1} + x^{4-0}$, and rearrange. For (3), say $V_i = \{z_1, \ldots, z_\ell\}$ is directed from $z_1$ to $z_\ell$. Then $\text{indeg}(z_\ell) = 2$ and $\text{indeg}(z_j) = 1$ for all $j < \ell$. \qed

3.1 A Simple Algorithm

As in many previous orthogonal drawing papers [1,7,9], the idea is to draw the graph $G_i$ induced by $V_1 \cup \cdots \cup V_i$ in such a way that all unfinished edges (edges with one end in $G_i$ and the other in $G - G_i$) end in a column that is unused above the point where the drawing ends.

**Embedding the First Two Vertices:** If $(v_1, v_n)$ is a single edge, then $v_1$ and $v_2$ are embedded exactly as in [1]; refer to Fig. 2. If $(v_1, v_n)$ is a double edge, then it was added only for the purpose of making the graph 4-regular and need not be drawn. In that case, we omit one of the outgoing edges of $v_1$ that has a bend.

**Embedding a Singleton:** If $V_i$ is a singleton $\{z\}$, we embed $z$ exactly as in [1]; refer to Fig. 2. For $\text{indeg}(z) \in \{2, 3\}$, this adds one new row and $\text{outdeg}(z) - 1 = 3 - \text{indeg}(z)$ many new columns. For $\text{indeg}(z) = 4$, $z = v_n$; if $(v_1, v_n)$ is a double edge, we omit the edge having two bends.

**Embedding Chains:** Let $V_i$ be a chain, say $V_i = \{z_1, \ldots, z_\ell\}$ with $\ell \geq 2$. For chains, our algorithm is substantially different from [1]. Only $z_1$ and $z_\ell$ have predecessors. We place the chain-vertices on a new horizontal row above the previous drawing, between the edges from the predecessors; see Fig. 3. We add new columns as needed to have space for new vertices and outgoing edges without using columns that are in use for other unfinished edges. We also use a second new row if the chain is a long chain.
Fig. 2. Embedding the first two vertices, and a singleton with in-degree 2, 3, 4. Newly added grid-lines are dotted.

Fig. 3. Embedding short and long chains.

**Observation 4.** The increase in the half-perimeter is as follows:
- For the first and last vertex-group: $O(1)$
- For a 3-1-singleton: $+1$ (we add one row)
- For a 2-2-singleton: $+2$ (we add one row and one column)
- For a short chain: $+3$ (we add one row and two columns)
- For a long chain $V_i$: $+2|V_i|$ (we add two rows and $2|V_i| - 2$ columns)

**Corollary 5.** The half-perimeter is at most $\frac{3}{2}n + \frac{1}{2}x^{2-2} - x^{short} + O(1)$.

**Proof.** From Observation 4 and using Observation 3.3 the half-perimeter is at most $x^{3-1} + 2x^{2-2} + 2x^{1-3} - x^{short} + O(1)$. By Observation 3.2 this is at most $\frac{3}{2}x^{3-1} + 2x^{2-2} + \frac{3}{2}x^{1-3} - x^{short} + O(1)$, which gives the result. \(\square\)

**Theorem 6.** Every simple 3-connected 4-graph has an orthogonal drawing of area at most $\frac{25}{36}n^2 + O(n) \approx 0.69n^2$.

**Proof.** First, make the graph 4-regular, compute the 3-canonical order, and consider the number $x^{2-2}$ of 2-2-vertices.

1. If $x^{2-2} \leq n/3$, apply the above algorithm. By Corollary 5, the half-perimeter is at most $\frac{3}{2}n + \frac{1}{6}n + O(1) \leq \frac{5}{6}n + O(1)$.
2. If $x^{2-2} \geq n/3$, apply the algorithm from [9]. They state their area bound as $0.76n^2 + O(1)$, but one can observe (see [9, Theorem 3.1, ll.2–5]) that their half-perimeter is at most $2n - \frac{1}{2}(x^{1-3} + x^{2-2}) + O(1)$, since they pair at least $x^{1-3} + x^{2-2}$ vertices. Using Observation 3.2 and ignoring $O(1)$ terms, we have $x^{1-3} + x^{2-2} = \frac{1}{2}x^{1-3} + x^{2-2} + \frac{1}{2}x^{3-1} = \frac{1}{2}n + \frac{1}{2}x^{2-2}$. Hence, the half-perimeter of their algorithm is at most $\frac{7}{4}n - \frac{1}{4}x^{2-2} + O(1) \leq (\frac{7}{4} - \frac{1}{12})n + O(1) = \frac{5}{3}n + O(1)$.

In both cases, we get a drawing with half-perimeter $\frac{5}{3}n + O(1)$. The area of it is maximal if the two sides are equally large and thus at most $(\frac{5}{6}n + O(1))^2$. \(\square\)
3.2 Improvement via Pairing

We already know a bound of $\frac{3}{2}n + \frac{1}{2}x^{2-2} - \frac{1}{2}x^{\text{short}} + O(1)$ on the half-perimeter. This section improves this further to half-perimeter $\frac{3}{2}n + O(1)$. The idea is strongly inspired by the pairing technique of Papakostas and Tollis [9]. They created pairs of vertices with special properties such that at least $\frac{1}{2}(x^{2-2} + x^{1-3})$ such pairs must exist. For each pair, they can save at least one grid-line, compared to the $2n + O(1)$ grid-lines created with [1].

Our approach is similar, but instead of pairing vertices, we pair groups of the canonical order by scanning them in backward order as follows:

1. Initialize $i := k - 1$. (We ignore the last group, which is a 4-0-singleton.)
2. While $V_i$ is a 3-1-singleton and $i > 2$, set $i := i - 1$.
3. If $i = 2$, break. Else, $V_i$ is a chain or a 2-2-singleton and we choose the partner of $V_i$ as follows: Initialize $j := i - 1$. While $V_j$ is a 3-1-singleton whose successor is not in $V_i$, set $j := j - 1$. Now, pair $V_i$ with $V_j$. Observe that such a $V_j$ with $j \geq 2$ always exists, since $i > 2$ and $V_2$ is not a 3-1-singleton.
4. Update $i := j - 1$ and repeat from Step (2) onwards.

In the small example in Fig.1, the 2-2-singleton $V_5$ gets paired with the short chain $V_4$, and all other groups are not paired.

Observe that, with the possible exception of $V_2$, every chain is paired and every 2-2-vertex is in a paired group (either as 2-2-singleton or as part of a chain). Hence there are at least $\frac{1}{2}(x^{2-2} - 1)$ pairs. The key observation is the following:

Lemma 7. Let $V_i, V_j$ be two vertex groups that are paired. Then there exists a method of drawing $V_i$ and $V_j$ (without affecting the layout of any other vertices) such that the increase to rows and columns is at most $2|V_i \cup V_j| - 1$.

We defer the (lengthy) proof of Lemma 7 to the next section, and study here first its consequences. We can draw $V_1$ and $V_k$ using $O(1)$ grid-lines. We can draw $V_2$ using $2|V_2| = 2x^{2-2}_V + 2x^{1-3}_V$ new grid-lines, where $x^{\ell-k}_W$ denotes the number of vertices of in-degree $\ell$ and out-degree $k$ in vertex set $W$. We can draw any unpaired 3-1-singleton using one new grid-line. Finally, we can draw each pair using $2|V_i \cup V_j| - 1 = 2x^{2-2}_{V_i \cup V_j} + 2x^{1-3}_{V_i \cup V_j} - 1$ new grid-lines. This covers all vertices, since all 2-2-singletons and all chains belong to pairs or are $V_2$, and since there are no 1-3-singletons.

Putting it all together and using Observation 3.2, the number of grid-lines hence is $2x^{1-3} + 2x^{2-2} + x^{3-1} - \#\text{pairs} + O(1) \leq 2x^{1-3} + \frac{3}{2}x^{2-2} + x^{3-1} + O(1) = \frac{3}{4}n + O(1)$ as desired. Since a drawing with half-perimeter $\frac{3}{2}n$ has area at most $(\frac{3}{4}n)^2 = \frac{9}{16}n^2$, we can conclude:

Theorem 8. Every simple 3-connected 4-graph has an orthogonal drawing of area at most $\frac{9}{16}n^2 + O(n) \approx 0.56n^2$.

We briefly discuss the run-time. The 3-canonical order can be found in linear time. Most steps of the drawing algorithm work in constant time per vertex,
hence $O(n)$ time total. One difficulty is that to place a group we must know the relative order of the columns of the edges from the predecessors. As discussed extensively in [1], we can do this either by storing columns as a balanced binary search tree (which uses $O(\log n)$ time per vertex-addition), or using the data structure by Dietz and Sleator [5] which allows to find the order in $O(1)$ time per vertex-addition. Thus, the worst-case run-time to find the drawing is $O(n)$.

4 Proof of Lemma 7

Recall that we must show that two paired vertex groups $V_i$ and $V_j$, with $j < i$, can be embedded such that we use at most $2|V_i| + 2|V_j| - 1$ new grid-lines. The proof of this is a massive case analysis, depending on which type of group $V_i$ and $V_j$ are, and whether there are edges between them or not.\footnote{The constructions we give have been designed as to keep the description simple; often even more grid-lines could be saved by doing more complicated constructions.} We first observe some properties of pairs.

**Observation 9.** By choice of the pairing, the following holds:

1. For any pair $(V_i, V_j)$ such that $j < i$, $V_i$ is either a 2-2-singleton or a chain.
2. If $V_i$ is paired with $V_j$ such that $j < i$, then all predecessors of $V_i$ are in $V_j$ or occurred in a group before $V_j$.

The following notation will cut down the number of cases a bit. We say that groups $V_i$ and $V_j$ are *adjacent* if there is an edge from a vertex in one to a vertex in the other group. If two paired groups $V_i, V_j$ are not adjacent, then by Observation 9.2 all predecessors of $V_i$ occur before group $V_j$. We hence can safely draw $V_i$ first, and then draw $V_j$, thereby effectively exchanging the roles of $V_i$ and $V_j$ in the pair. Now, we distinguish five cases:

1. At least one of $V_i$ and $V_j$ is a short chain. Say $V_i$ is the short chain, the other case is similar. Recall that the standard layout for a short chain uses 3 new grid-lines, but $x_{2V_i}^{2-2} + x_{V_i}^{1-3} = 2$. So the layout of a short chain automatically saves one grid-line. We do not change the algorithm at all in this case; laying out $V_i$ and $V_j$ exactly as before results in at most $2x_{V_i\cup V_j}^{2-2} + 2x_{V_i\cup V_j}^{1-3} - 1$ new grid-lines. (This is what happens in the example of Fig. 1.)

![Fig. 4. Reusing the column freed by a 3-1-singleton with a later chain or singleton. In this and the following figures, the re-used grid-line is dotted and red.](image)

\footnote{The constructions we give have been designed as to keep the description simple; often even more grid-lines could be saved by doing more complicated constructions.}
V
\_i
Fig. 5. Sharing the extra row between two long chains when there are 0, 1 or 2 predecessors of \( V_i \) in \( V_j \).

2. One of \( V_i \) and \( V_j \) is a 3-1-singleton. By Observation 9, the 3-1-singleton must be \( V_j \). By the pairing algorithm, the unique outgoing edge of the 3-1-singleton must lead to \( V_i \). Draw \( V_j \) as before. We can then draw \( V_i \) such that it re-uses one of the columns that were freed by \( V_j \); see Fig. 4.

3. \( V_i \) and \( V_j \) are both long chains. In this case, both \( V_i \) and \( V_j \) can use the same extra row for the “detours” that their middle vertices (by which we mean vertices that are neither the first nor the last vertex of the chain) use. Since we can freely choose into which columns these middle vertices are placed, we can ensure that none of these “detours” overlap and, hence, one row suffices for both chains. This holds even if one or both of the predecessors of \( V_i \) are in \( V_j \), as these are distinct and the two corresponding incoming edges of \( V_i \) extend the edges that were already drawn for \( V_j \); see Fig. 5.

4. None of the previous cases applies and \( V_j \) is a 2-2-singleton. By Observation 9.1 and since Case (1) does not apply, \( V_i \) is either a 2-2-singleton or a long chain. There are two columns reserved for edges from predecessors of \( V_j \). Since predecessors of \( V_i \) are distinct, at most one of them can be the 2-2-singleton in \( V_j \). Thus, there also is at least one column reserved for an edge from a predecessor of \( V_i \) not in \( V_j \). We call these three or four columns the predecessor-columns. We have three sub-cases depending on the relative location of these columns:

(a) The leftmost predecessor-column leads to \( V_j \). In this case, we save a column almost exactly as in [9]. Place \( V_j \) as before, in the right one of its predecessor-columns. This leaves the leftmost predecessor-column free to be reused. Now no matter whether \( V_i \) is a 2-2-singleton or a long chain, or whether \( V_i \) is adjacent to \( V_j \) or not, we can re-use this leftmost column for one outgoing edge of \( V_i \) with a suitable placement; see Fig. 6.

(b) The rightmost predecessor-column leads to \( V_j \). This case is symmetric to the previous one.

(c) The leftmost and rightmost predecessor-columns lead to \( V_i \). This implies that \( V_i \) has two predecessors not in \( V_j \). Hence, \( V_i \) cannot be adjacent to \( V_j \). If \( V_i \) is a 2-2-singleton, then (as discussed earlier) we can exchange
the roles of $V_i$ and $V_j$, which brings us to Case 4(a). If $V_i$ is a long chain, then place $V_j$ in the standard fashion. We then place the long chain $V_i$ such that the “detours” of its middle vertices re-use the row of $V_j$. See Fig. 7.

5. None of the previous cases applies. Then $V_j$ is a chain, say $V_j = \{z_1, \ldots, z_\ell\}$, and $\ell \geq 3$ since Case (1) does not apply. We assume the naming is such that the predecessor column of $z_1$ is left of the predecessor column of $z_\ell$.

Since we are not in a previous case, $V_i$ must be a 2-2-singleton, say $z$. If $V_i$ is not adjacent to $V_j$, then we can again exchange the roles of $V_i$ and $V_j$, which brings us to Case (4). Hence, we may assume that there are edges between $V_j$ and $V_i$. We distinguish the following sub-cases depending on how many such edges there are and whether their ends are middle vertices.

(a) $z$ has exactly one neighbor in $V_j$, and it is either $z_1$ or $z_\ell$. We rearrange $V_i \cup V_j$ into two different chains. Let $z$ be adjacent to $z_1$ (the other case is symmetric). Then $\{z, z_1\}$ forms one chain and $\{z_2, \ldots, z_\ell\}$ forms another. Embed these two chains as usual. Since $\{z, z_1\}$ forms a short chain, this saves one grid-line; see Fig. 8(left).

(b) $z$ has exactly one neighbor in $V_j$, and it is $z_h$ for some $1 < h < \ell$. Embed the chain $V_j$ as usual, but omit the new column next to $z_h$. For embedding $z$, we place a new row below the rows for the chain. Using this new row, we can connect the bottom outgoing edge of $z_h$ to the horizontal incoming edge of $z$; see Fig. 8(right).

(c) $z$ has two neighbors in $V_j$, and both of them are middle vertices $z_g, z_h$ for $1 < g < h < \ell$. Embed the chain $V_j$ as usual, but omit the new columns next to $z_g$ and $z_h$. Place a new row between the two rows for the chain and use it to connect the two bottom outgoing edges of $z_g$ and $z_h$ to place $z$, re-using the row for the detours to place the bottom outgoing edge of $z$. This uses an extra column for $z$, but saved two columns at $z_g$ and $z_h$, so overall one grid-line has been saved; see Fig. 9(top left).

(d) $z$ is adjacent to $z_1$ and $z_2$ (the case of adjacency to $z_{\ell-1}$ and $z_\ell$ is symmetric). Embed $z_2, \ldots, z_\ell$ as usual for a chain, then place $z_1$ below $z_2$. The horizontally outgoing edge of $z_2$ intersects one outgoing edge of $z_1$. Put $z$ at this place to save a row and a column; see Fig. 9(top right).

(e) $z$ is adjacent to $z_1$ and $z_h$ with $h > 2$ (the case of adjacency to $z_\ell$ and $z_h$ with $h < \ell - 1$ is symmetric). Draw the chain $V_j$ with the modification that $z_h$ is below $z_{h-1}$, but still all middle vertices use the same extra row for their downward outgoing edges. This uses 3 rows, but now $z$ can be placed using the two left outgoing edges of $z_1$ and $z_h$, saving a row for $z$ and a column for the left outgoing edge of $z_h$; see Fig. 9(bottom), both for $h < \ell$ and $h = \ell$.

This ends the proof of Lemma 7 and hence shows Theorem 8.
Fig. 6. Reusing the predecessor-column freed by a 2-2-singleton $V_j$ in Case 4(a). Left two pictures: $V_i$ is not adjacent to $V_j$. Right two: $V_i$ is adjacent to $V_j$.

Fig. 7. If the predecessor-columns of $V_j$ are between the ones of $V_i$, then we can either revert to Case 4(a) or the long chain $V_i$ can re-use the row of $V_j$.

Fig. 8. $V_j$ is a long chain, $V_i$ is a 2-2-singleton with one predecessor in $V_j$.

Fig. 9. $V_j$ is a long chain, $V_i$ is a 2-2-singleton, and there are exactly two edges between them. (Top) Cases 5(c) and (d). (Bottom) Case 5(e).
5 Conclusion

In this paper, we gave an algorithm to create an orthogonal drawing of a 3-connected 4-graph that has area at most \( \frac{9}{16}n^2 + O(n) \approx 0.56n^2 \). As a main tool, we used the 3-canonical order / Mondshein sequence for non-planar 3-connected graphs, whose existence was long known but only recently efficient algorithms for it were found. To our knowledge, this is the first application of the 3-canonical order on non-planar graphs in graph-drawing. Among the many remaining open problems are the following:

- Can we draw 2-connected 4-graphs with area less than \( 0.76n^2 \)? A natural approach would be to draw each 3-connected component with area \( 0.56n^2 \) and to merge them suitably, but there are many cases depending on how the cut-vertices and virtual edges are drawn, and so this is far from trivial.
- Can we draw 3-connected 4-graphs with \( (2 - \varepsilon)n \) bends, for some \( \varepsilon > 0 \)? With an entirely different algorithm (not given here), we have been able to prove a bound of \( 2n - x^2 - 2 + O(1) \) bends, so an improved bound seems likely.
- Our algorithm was strongly inspired by the one of Kant [7] for 3-connected planar graphs. Are there other graph drawing algorithms for planar 3-connected graphs that can be transferred to non-planar 3-connected graphs by using the 3-canonical order?

Acknowledgments. We wish to thank the anonymous reviewers for their constructive comments.

References

1. Biedl, T.C., Kant, G.: A better heuristic for orthogonal graph drawings. Comput. Geom. 9(3), 159–180 (1998)
2. Cheriyan, J., Maheshwari, S.N.: Finding nonseparating induced cycles and independent spanning trees in 3-connected graphs. J. Algorithms 9(4), 507–537 (1988)
3. Chiang, Y.-T., Lin, C.-C., Lu, H.-I.: Orderly spanning trees with applications. SIAM J. Comput. 34(4), 924–945 (2005)
4. de Fraysseix, H., de Mendez, P.O.: Regular orientations, arboricity and augmentation. In: Tamassia, R., Tollis, I.G. (eds.) GD 1994. LNCS, vol. 894, pp. 111–118. Springer, Heidelberg (1995)
5. Dietz, P., Sleator, D.: Two algorithms for maintaining order in a list. In: 19th Annual ACM Symposium on Theory of Computing, pp. 365–372 (1987)
6. de Fraysseix, H., Pollack, R., Pach, J.: How to draw a planar graph on a grid. Combinatorica 10, 41–51 (1990)
7. Kant, G.: Drawing planar graphs using the canonical ordering. Algorithmica 16(1), 4–32 (1996)
8. Mondshein, L.F.: Combinatorial ordering and the geometric embedding of graphs. Ph.D. thesis, M.I.T. Lincoln Laboratory / Harvard University (1971)
9. Papakostas, A., Tollis, I.G.: Algorithms for area-efficient orthogonal drawings. Comput. Geom. 9(1–2), 83–110 (1998)
10. Rosenstiehl, P., Tarjan, R.E.: Rectilinear planar layouts and bipolar orientation of planar graphs. Discrete Comput. Geom. 1, 343–353 (1986)
11. Schäffer, M.: Drawing graphs on rectangular grids. Discrete Appl. Math. 63, 75–89 (1995)
12. Schmidt, J.M.: The Mondshein sequence (2013). http://arxiv.org/pdf/1311.0750.pdf
13. Schmidt, J.M.: The Mondshein sequence. In: Esparza, J., Fraigniaud, P., Husfeldt, T., Koutsoupias, E. (eds.) ICALP 2014. LNCS, vol. 8572, pp. 967–978. Springer, Heidelberg (2014)
14. Biedl, T.: Optimal orthogonal drawings of triconnected plane graphs. In: McCune, W., Padmanabhan, R. (eds.) Automated Deduction in Equational Logic and Cubic Curves. LNCS, vol. 1095, pp. 333–344. Springer, Heidelberg (1996)
15. Tamassia, R., Tollis, I.: A unified approach to visibility representations of planar graphs. Discrete Comput. Geom. 1, 321–341 (1986)
16. Tamassia, R., Tollis, I.G., Vitter, J.S.: Lower bounds for planar orthogonal drawings of graphs. Inf. Process. Lett. 39, 35–40 (1991)
17. Valiant, L.G.: Universality considerations in VLSI circuits. IEEE Trans. Comput. C–30(2), 135–140 (1981)