Convex Grid Drawings of Plane Graphs with Rectangular Contours

Akira Kamada (Tohoku University)
Kazuyuki Miura (Fukushima University)
Takao Nishizeki (Tohoku University)
Drawings

vertex

edge

plane graph

no edge-intersection

convex grid drawing
Convex grid drawing

vertex

edge

plane graph

vertex: grid point

dge: straight line segment without edge-intersection

face: convex polygon

outer face
Known results (1)

- Plane graph
- 2-connected
- Internally 3-connected

Known results (1)
Convex grid drawing
Known results (1)

- Plane graph
- No cut vertex
- 2-connected

Cut vertex
Known results (1)

plane graph

2-connected

inner vertex

cannot be simultaneously drawn as convex polygons
Known results (1)

plane graph

2-connected

inner vertex

outer vertex

cannot be simultaneously drawn as convex polygons
Known results (1)

for any separation pair \{ u, v \} of \( G \),
1) both \( u \) and \( v \) are outer vertices, and
2) each component of \( G - \{ u, v \} \) contains an outer vertex
Known results (1)

plane graph

2-connected

internally 3-connected

known results (1)

convex grid drawing
Decomposition tree
Decomposition tree

separation pair
Decomposition tree
Decomposition tree

repeat this operation
until no more splits are possible
Decomposition tree

no more splits are possible
Decomposition tree

no more splits are possible
Decomposition tree

ring (cycle)

merge

no more splits are possible
Decomposition tree

3-connected component decomposition tree  [HT73]

○ : leaf
Known results (2)

leaves $\leq 3$

convex grid drawing
[CK97], [MAN05]

triangular contour

$n$ vertices
Known results (2)

- Internally 3-connected
- Leaves ≤ 3
- Leaves = 4: open problem
- Leaves ≥ 4

Convex grid drawing: polynomial size

Our result?
Our result

leaves = 4

rectangular contour

convex grid drawing

n vertices

n^2

2n
Our result

leaves = 4

convex grid drawing

rectangular contour

corner vertex
Our result

leaves = 4

rectangular contour

convex grid drawing

corner vertex

triangular contour
Our result

leaves = 4

rectangular contour

convex grid drawing

corner vertex
Our result

leaves $= 4$

convex grid drawing
Main idea

leaves = 4

divide

internally 3-connected graphs

leaves = 2
Main idea

inner convex grid drawings

internally 3-connected graphs

leaves = 2

new “shift method”
Main idea

inner convex grid drawings $\rightarrow$ convex grid drawing
Main idea

Input graph

Convex grid drawing

$n$ vertices

$n^2$ vertices
Algorithm

leaves = 4

divide

internally 3-connected graphs

leaves = 2
Algorithm

leaves = 4

divide

internally 3-connected graphs

leaves = 2
Algorithm

inner convex grid drawings

internally 3-connected graphs

leaves = 2

new "shift method"
Algorithm

inner convex grid drawing

internally 3-connected graph

leaves $= 2$

canonical decomposition

$\Pi = (U_1, U_2, \ldots, U_{m-1}, U_m)$

new “shift method”
Canonical decomposition

\[ \Pi = (U_1, U_2, \ldots, U_{m-1}, U_m) \]
Canonical decomposition

\[ \Pi = (U_1, U_2, \ldots, U_{m-1}, U_m) \]
Canonical decomposition

\[ \Pi = (U_1, U_2, \ldots, U_{m-1}, U_m) \]

- One or more neighbors
- Two or more

\( G_{k-1} \)

- One or more (each vertex)
- Exactly one (leftmost and rightmost vertices)

\( G_k \) is induced by \( U_1 \cup U_2 \cup \cdots \cup U_k \)
Canonical decomposition

\[ \Pi = ( U_1, U_2, \ldots, U_{m-1}, U_m ) \]

- One or more neighbors
- Two or more
- One or more (each vertex)
- Exactly one (leftmost and rightmost vertices)

\[ G_k \text{ is induced by } U_1 \cup U_2 \cup \ldots \cup U_k \]
Algorithm

① convex polygon

② slope = 0 or ±1

③ convex apex has upper neighbors

new “shift method”
Algorithm

new "shift method"

inner face

convex polygon
Algorithm

\[ \text{slope} = 0 \text{ or } \pm 1 \]

new "shift method"
Algorithm

3. Convex apex has upper neighbors

Convex apex

Concave apex

Less than 180°

New “shift method”
Algorithm

1. convex polygon
2. slope = 0 or ±1
3. convex apex has upper neighbors

new “shift method”
Algorithm

1. convex polygon
2. slope = 0 or ±1
3. convex apex has upper neighbors

new “shift method”
Algorithm

3. **convex apex** has upper neighbors

2. slope = 0 or ±1

1. convex polygon

**new “shift method”**

shift $\leftarrow$ shift

$G_1$ $\rightarrow$ $U_2$
Algorithm

1. **Convex polygon**
2. **Slope** = 0 or ±1
3. Convex apex has upper neighbors

New "shift method"
Algorithm

1. convex polygon

2. slope $= 0$ or $\pm 1$

3. convex apex has upper neighbors

new “shift method”
Algorithm

① convex polygon

② slope = 0 or ±1

③ convex apex has upper neighbors

new “shift method”
Algorithm

1. Convex polygon
2. Slope $= 0$ or $\pm 1$
3. Convex apex has upper neighbors

New “shift method”
Algorithm

① convex polygon

② slope = 0 or ±1

③ convex apex has upper neighbors

new “shift method”
Algorithm

① **convex polygon**

② slope = 0 or ±1

③ convex apex • has upper neighbors

new “shift method”
Algorithm

① convex polygon

② slope = 0 or ±1

③ convex apex has upper neighbors

new "shift method"
Algorithm

① convex polygon

② slope = 0 or ±1

③ convex apex has upper neighbors

new “shift method”
Algorithm

① convex polygon

② slope = 0 or ±1

③ convex apex has upper neighbors

new “shift method”
Algorithm

① convex polygon

② slope = 0 or ±1

③ convex apex has upper neighbors

new “shift method”
Algorithm

① convex polygon

② slope = 0 or ±1

③ convex apex has upper neighbors

new “shift method”
Algorithm

1. convex polygon
2. \( \text{slope} = 0 \) or \( \pm 1 \)
3. convex apex has upper neighbors

new “shift method”

\[ G_7 \rightarrow U_8 \]
Algorithm

1. Convex polygon

2. Slope = 0 or ±1

3. Convex apex has upper neighbors

new “shift method”
Algorithm

① convex polygon

② slope = 0 or ±1

③ convex apex has upper neighbors

new “shift method”
Algorithm

1. **convex polygon**
   - new "shift method"

2. slope = 0 or ±1

3. convex apex has upper neighbors
   - $G_9$
Grid size
Algorithm

inner convex grid drawing

internally 3-connected graph

leaves = 2

new “shift method”
Algorithm

inner convex grid drawings

internally 3-connected graphs

leaves = 2

new “shift method”
Algorithm

inner convex grid drawings

internally 3-connected graphs

leaves = 2

\[ n = n_1 + n_2 \]

\[ \max\{2n_1, 2n_2\} < 2n \]
Algorithm

inner convex grid drawings

internally 3-connected graphs

leaves = 2

every inner face is a convex polygon
Algorithm

inner convex grid drawings

internally 3-connected graphs

leaves = 2

\[ |\text{slope}| \leq 1 \]
Algorithm

**inner** convex grid drawings

internally 3-connected graphs

leaves $= 2$

$| \text{slope} | \leq 1$

$2n \quad n_1^2 \quad n_2$

$\begin{align*}
| \text{slope} | & \leq 1 \\
n_1^2 & \to n_1 \\
n_2 & \to n_2 \\
2n & \to \text{inner convex grid drawings}
\end{align*}$
Algorithm

inner convex grid drawings \rightarrow convex grid drawing

\[
|\text{slope}| \leq 1
\]

non-convex polygon

equal-intersection

2n
Algorithm

inner convex grid drawings $\rightarrow$ convex grid drawing

$|\text{slope}| \leq 1$

$n_2^2$ $n_1^2$

$2n$ $2n + 1$
Algorithm

**inner** convex grid drawings \( \xrightarrow{\quad} \) convex grid drawing

| slope | \( \leq 1 \) | 1 < | slope |

\( n_1^2 \) \( \quad \) \( 2n+1 \)

\( n_2^2 \) \( \quad \) 2n
**Algorithm**

*inner* convex grid drawings $\rightarrow$ convex grid drawing

- $|\text{slope}| \leq 1$
- $1 < |\text{slope}|$
- no edge-intersection, and every face newly created is a convex polygon

$n^2_2$

$2n + 1$

$2n$
Algorithm

inner convex grid drawings \rightarrow convex grid drawing

\[ H = n_1^2 + 2n + 1 + n_2^2 < n^2 \]
Conclusions

leaves = 4

convex grid drawing

linear time

$n$ vertices

$n^2$

$2n$
Conclusions

3-connected graph

degree = 4

convex grid drawing

3-connected graph

degree = 4

ring

degree = 4

Conclusions

3-connected graph

degree = 4

convex grid drawing

ring

degree = 4
Conclusions

- \( n \) vertices
- Ring
- Degree = 4

\[ 2n + 1 \]

\[ 2n_1 \]

\[ 2n \]

\[ 2n_2 \]

\[ 4n \]
Conclusions

Degree = 3

Degree = 3

Degree = 4

n vertices

2n + 1

2n_1

2n_2

4n

2n

2n_1

2n_2
Conclusions

degree = 3

degree = 3

\[ 2n^2 \]

\[ 4n \]

\[ 2n_2 \]

\[ 2n+1 \]

\[ 2n_1 \]

\[ 2n \]

\[ 4n \]

\[ n \] vertices
Open problem

Internally 3-connected

Leaves $\geq 5$: open problem

Leaves $= 4$

Known results (2)

Our result

Convex grid drawing $2n \times n^2$ size
Open problem

leaves = 5

convex grid drawing

$n$ vertices
Open problem

leaves $= 5$

$n$ vertices

convex grid drawing
Decomposition tree

separation pair
Decomposition tree

virtual edge
Decomposition tree
Decomposition tree

repeat this operation
until no more splits are possible
Decomposition tree

repeat this operation until no more splits are possible
Decomposition tree

repeat this operation until no more splits are possible
Decomposition tree

repeat this operation
until no more splits are possible
Decomposition tree

repeat this operation until no more splits are possible
Decomposition tree

repeat this operation
until no more splits are possible
Decomposition tree

triangle

merge

no more splits are possible
Decomposition tree

ring(cycle)

merge
Decomposition tree

3-connected component decomposition tree  [HT73]
Canonical decomposition

\[ \Pi = (U_1, U_2, \ldots, U_{m-1}, U_m) \]
Canonical decomposition

\[ \Pi = ( U_1, U_2, \ldots, U_{m-1}, U_m ) \]

- One or more neighbors
- Two or more
- One or more (each vertex)
- Exactly one (leftmost and rightmost vertices)

\( G_k \) is induced by \( U_1 \cup U_2 \cup \cdots \cup U_k \)

\( G_{k-1} \) is induced by \( U_k \)
Canonical decomposition

\[ \Pi = (U_1, \underbrace{U_2, \ldots, U_{m-1}}_{\text{one or more}}, U_m) \]

- One or more neighbors
- Two or more

- One or more (each vertex)
- Exactly one (leftmost and rightmost vertices)

\( G_k \) is induced by
\[ U_1 \cup U_2 \cup \ldots \cup U_k \]

\( G_{k-1} \) is induced by \( U_1 \cup U_2 \cup \ldots \cup U_k \)
**Canonical decomposition**

- One or more neighbors
- Two or more
- One or more (each vertex)
- Exactly one (leftmost and rightmost vertices)

$G_{k-1}$ is induced by $U_1 \cup U_2 \cup \cdots \cup U_k$

**canonical decomposition**

$\Pi = (U_1, [U_2, \ldots, U_{m-1}], U_m)$
Canonical decomposition

One or more neighbors

Two or more

One or more (each vertex)

Exactly one (leftmost and rightmost vertices)

\[ G_{k-1} \]

Canonical decomposition

\[ \Pi = (U_1, U_2, \ldots, U_{m-1}, U_m) \]

\[ G_k \text{ is induced by } U_1 \cup U_2 \cup \cdots \cup U_k \]
Canonical decomposition

\( \Pi = (U_1, \overline{U_2}, \cdots, U_{m-1}, U_m) \)

- One or more neighbors
- Two or more

- One or more (each vertex)
- Exactly one (leftmost and rightmost vertices)

\( G_k \) is induced by
\( U_1 \cup U_2 \cup \cdots \cup U_k \)
Canonical decomposition

\[ \Pi = (U_1, [U_2, \ldots, U_{m-1}], U_m) \]

- One or more neighbors
- Two or more
- One or more (each vertex)
- Exactly one (leftmost and rightmost vertices)

\( G_k \) is induced by
\( U_1 \cup U_2 \cup \ldots \cup U_k \)

\( G_{k-1} \) is induced by
\( U_k \)
Canonical decomposition

\[
\Pi = (U_1, \overline{U_2}, \ldots, \overline{U_{m-1}}, U_m)
\]

- One or more neighbors
- Two or more
- One or more (each vertex)
- Exactly one (leftmost and rightmost vertices)

\(G_k\) is induced by \(U_1 \cup U_2 \cup \cdots \cup U_k\)

\(G_k\) is induced by \(U_1 \cup U_2 \cup \cdots \cup U_{k-1} \cup U_k\)

\(G_7\) is induced by \(U_1 \cup U_2 \cup \cdots \cup U_8\)
Canonical decomposition

\[ \Pi = (U_1, U_2, \ldots, U_{m-1}, U_m) \]

- one or more neighbors
- two or more

- one or more (each vertex)
- exactly one (leftmost and rightmost vertices)

\( G_k \) is induced by \( U_1 \cup U_2 \cup \cdots \cup U_k \)
Canonical decomposition

one or more neighbors

\[ U_k \]

two or more

\[ G_{k-1} \]

one or more (each vertex)

\[ U_k \]

exactly one (leftmost and rightmost vertices)

\[ G_{k-1} \]

\[ \Pi = (U_1, U_2, \ldots, U_{m-1}, U_m) \]

canonical decomposition

\[ U_{10} = U_m \]

\[ G_k \] is induced by \[ U_1 \cup U_2 \cup \cdots \cup U_k \]

\[ G_9 \]

\[ G_k \] is induced by

\[ U_1 \cup U_2 \cup \cdots \cup U_k \]
Canonical decomposition

\[ \Pi = (U_1, U_2, \cdots, U_{m-1}, U_m) \]
$$2n \times n^2 \text{ size}$$

$$n_1^2 + n_2^2 + 2n+1$$

$$= (n_1 + n_2)^2 - 2n_1n_2 + 2n+1$$

$$\leq n^2$$

$$n = n_1 + n_2$$

$$n_1, n_2 \geq 5$$

convex grid drawing