Rook-drawing for planar graphs

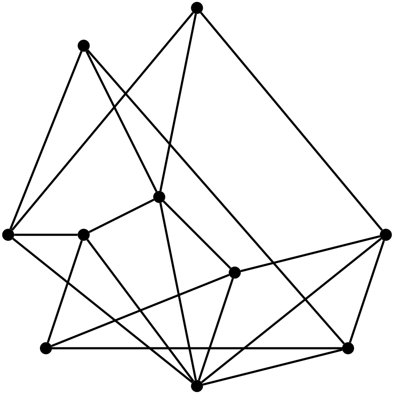
Claire Pennarun
From joint work with David Auber, Nicolas Bonichon and Paul Dorbec

LaBRI, Bordeaux

January 16^{th}, 2015

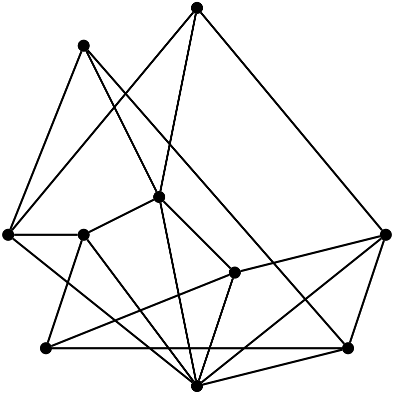
Drawing graphs

How can we draw graphs without too much effort when the graph changes ?



Drawing graphs

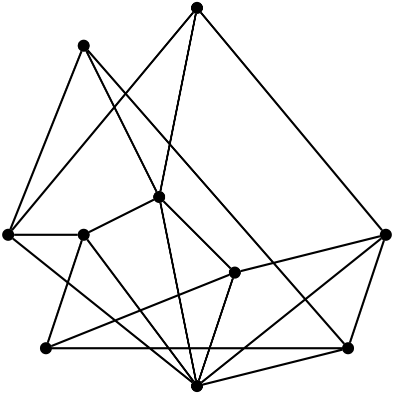
How can we draw graphs without too much effort when the graph **changes** ?



Drawing graphs

How can we draw graphs without too much effort when the graph **changes** ?

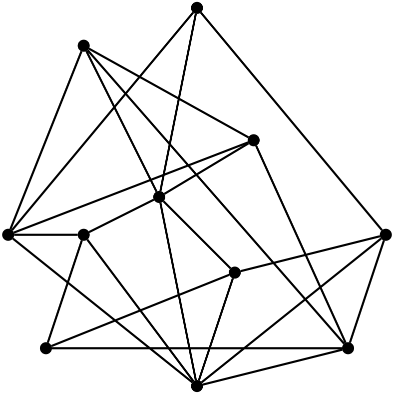
- addition of nodes



Drawing graphs

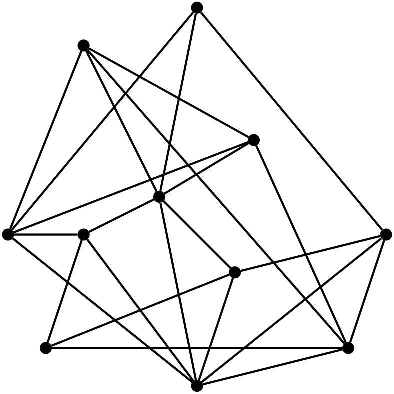
How can we draw graphs without too much effort when the graph **changes** ?

- addition of nodes



How can we draw graphs without too much effort when the graph **changes** ?

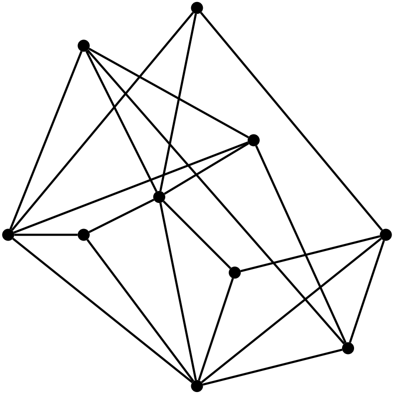
- addition of nodes
- deletion of nodes



Drawing graphs

How can we draw graphs without too much effort when the graph **changes** ?

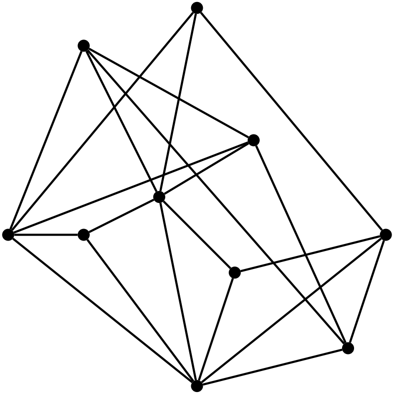
- addition of nodes
- deletion of nodes



Drawing graphs

How can we draw graphs without too much effort when the graph **changes** ?

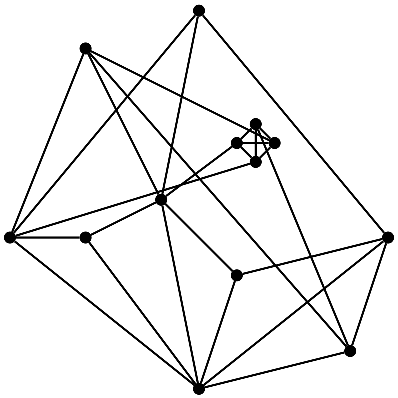
- addition of nodes
- deletion of nodes
- node "expansion"



Drawing graphs

How can we draw graphs without too much effort when the graph **changes** ?

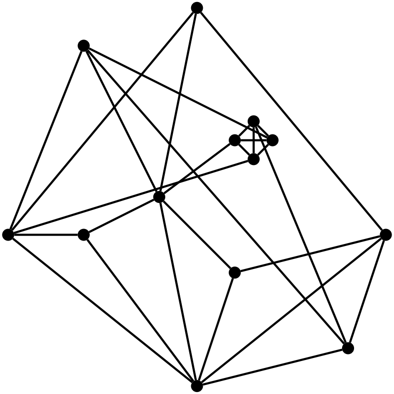
- addition of nodes
- deletion of nodes
- node "expansion"



Drawing graphs

How can we draw graphs **without too much effort** when the graph changes ?

- addition of nodes
- deletion of nodes
- node "expansion"

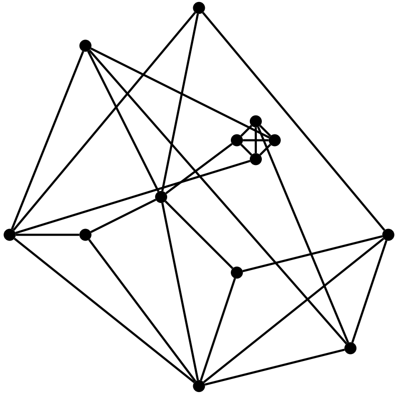


Drawing graphs

How can we draw graphs **without too much effort** when the graph changes ?

- addition of nodes
- deletion of nodes
- node "expansion"

- Looking for linear algorithms, easy to implement

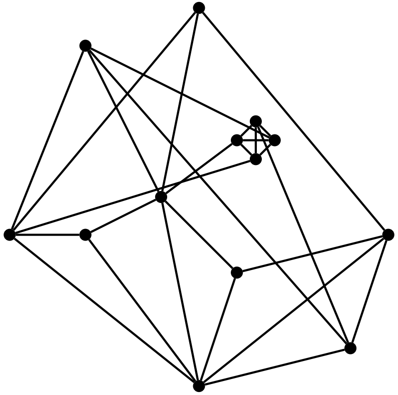


Drawing graphs

How can we draw graphs **without too much effort** when the graph changes ?

- addition of nodes
- deletion of nodes
- node "expansion"

- Looking for linear algorithms, easy to implement
- Preservation of the global "image" of the graph

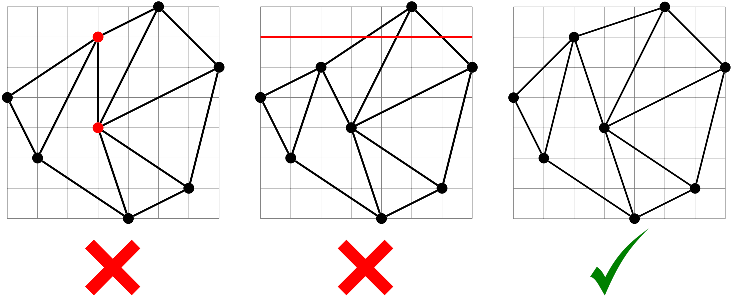


Rook-drawing

Definition

A rook-drawing of a graph of n vertices :

- Straight-lines
- Regular grid $n \times n$
- One vertex per line and column exactly

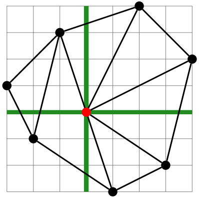


Rook-drawing

Definition

A rook-drawing of a graph of n vertices :

- Straight-lines
- Regular grid $n \times n$
- One vertex per line and column exactly

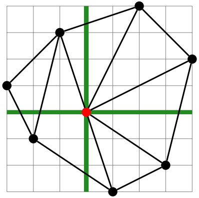


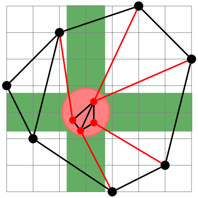
Rook-drawing

Definition

A rook-drawing of a graph of n vertices :

- Straight-lines
- Regular grid $n \times n$
- One vertex per line and column exactly



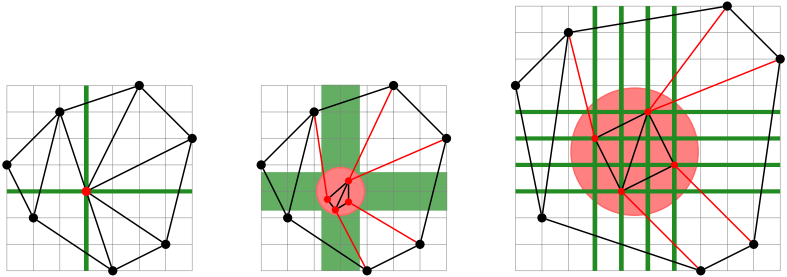


Rook-drawing

Definition

A rook-drawing of a graph of n vertices :

- Straight-lines
- Regular grid $n \times n$
- One vertex per line and column exactly



Planar rook-drawing

Is there a **planar** rook-drawing for every planar graph ?

Planar rook-drawing

Is there a **planar** rook-drawing for every planar graph ?

What we already know :

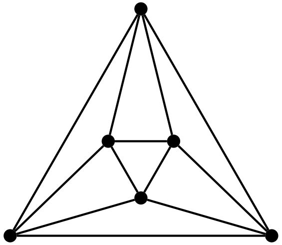
- Straight-lines drawing ([Fáry, 1948] : every planar graph)
- Grid drawing ([de Fraysseix, 1988], [Schnyder, 1990] : every planar graph on an $(n-2) \times (n-2)$ grid)

Planar rook-drawing

Is there a **planar** rook-drawing for every planar graph ?

What we already know :

- Straight-lines drawing ([Fáry, 1948] : every planar graph)
- Grid drawing ([de Fraysseix, 1988], [Schnyder, 1990] : every planar graph on an $(n-2) \times (n-2)$ grid)

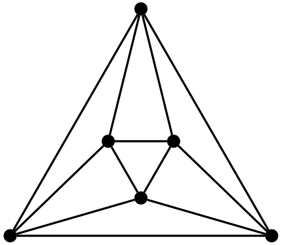


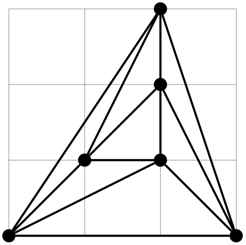
Planar rook-drawing

Is there a **planar** rook-drawing for every planar graph ?

What we already know :

- Straight-lines drawing ([Fáry, 1948] : every planar graph)
- Grid drawing ([de Fraysseix, 1988], [Schnyder, 1990] : every planar graph on an $(n-2) \times (n-2)$ grid)



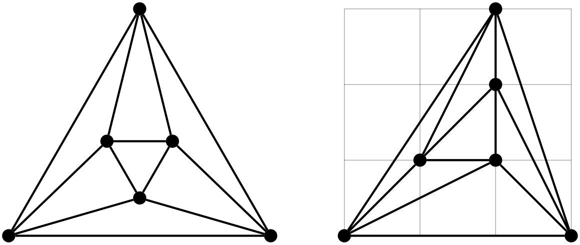


Planar rook-drawing

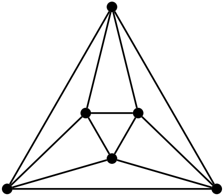
Is there a **planar** rook-drawing for every planar graph ? No !

What we already know :

- Straight-lines drawing ([Fáry, 1948] : every planar graph)
- Grid drawing ([de Fraysseix, 1988], [Schnyder, 1990] : every planar graph on an $(n-2) \times (n-2)$ grid)

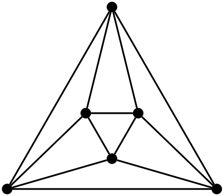


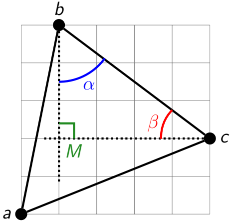
A counter-example



Three exterior nodes a, b and c. Inner nodes : inside the area (ab), (bc) and (ca).

A counter-example

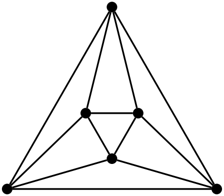




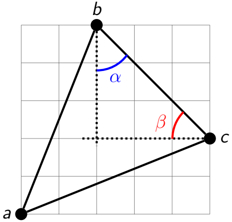
Three exterior nodes *a*, *b* and *c*. Inner nodes : inside the area (*ab*), (*bc*) and (*ca*).

$\alpha \geq 45°$, $\beta \geq 45°$. *Mab* right-angled

A counter-example

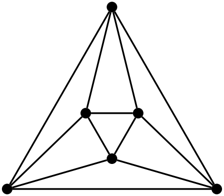


Three exterior nodes a, b and c. Inner nodes : inside the area (ab), (bc) and (ca).

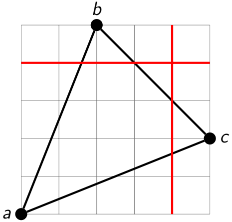


$\alpha \geq 45°$, $\beta \geq 45°$. *Mab* right-angled
$\alpha = \beta = 45° \rightarrow x(b) = y(c)$.

A counter-example

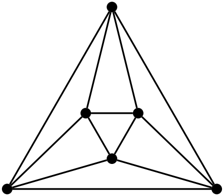


Three exterior nodes *a*, *b* and *c*. Inner nodes : inside the area (*ab*), (*bc*) and (*ca*).

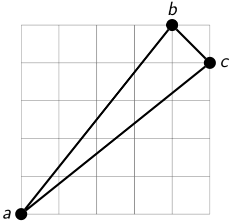


$\alpha \geq 45°$, $\beta \geq 45°$. *Mab* right-angled
$\alpha = \beta = 45° \rightarrow x(b) = y(c)$.
(*bc*) prevents any node to fill the line under *b* or the column at the left of *c*.

A counter-example

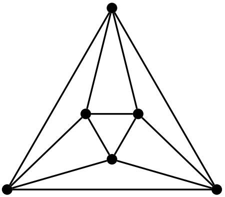


Three exterior nodes a, b and c. Inner nodes : inside the area (ab), (bc) and (ca).

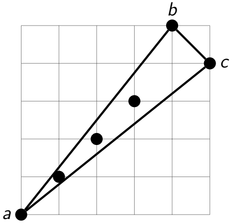


$\alpha \geq 45°$, $\beta \geq 45°$. *Mab* right-angled
$\alpha = \beta = 45° \rightarrow x(b) = y(c)$.
(bc) prevents any node to fill the line under b or the column at the left of c.
Fill these line and column with c and b !
$x(b) = y(c) = n - 1$.

A counter-example



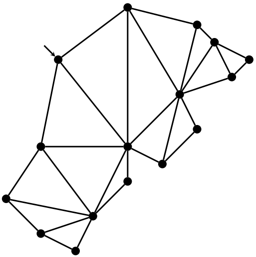
Three exterior nodes a, b and c. Inner nodes : inside the area (ab), (bc) and (ca).



$\alpha \geq 45°$, $\beta \geq 45°$. Mab right-angled
$\alpha = \beta = 45° \rightarrow x(b) = y(c)$.
(bc) prevents any node to fill the line under b or the column at the left of c.
Fill these line and column with c and b!
$x(b) = y(c) = n - 1$.
Inner nodes : along a diagonal.

Rook-drawing for outerplanar graphs

A graph is outerplanar if it has a planar drawing such that all its vertices are on the outer face.



Result

Every outerplanar graph has a rook-drawing which can be computed in linear time.

Rook-drawing for outerplanar graphs

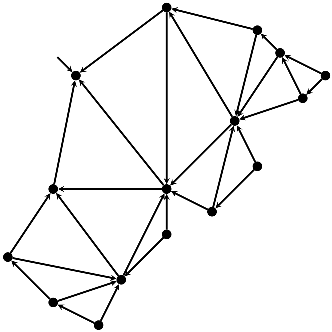
[Bonichon, Gavoille, Hanusse, 2005]

- edges of G outerplanar map $\rightarrow$ T_r, T_b
- edges of T_r : between u and its first non-descendant found after u in a clockwise pre-ordering of T_b.

Rook-drawing for outerplanar graphs

[Bonichon, Gavoille, Hanusse, 2005]

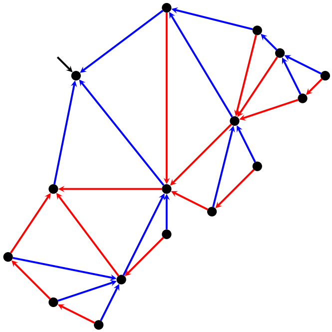
- edges of G outerplanar map $\rightarrow$ T_r, T_b
- edges of T_r : between u and its first non-descendant found after u in a clockwise pre-ordering of T_b.



Rook-drawing for outerplanar graphs

[Bonichon, Gavoille, Hanusse, 2005]

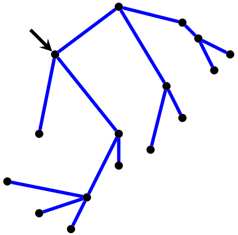
- edges of G outerplanar map $\rightarrow$ T_r, T_b
- edges of T_r : between u and its first non-descendant found after u in a clockwise pre-ordering of T_b.



Rook-drawing for outerplanar graphs

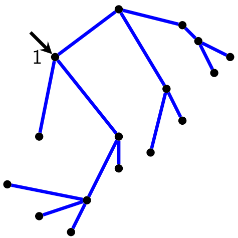
[Bonichon, Gavoille, Hanusse, 2005]

- edges of $G \rightarrow T_r, T_b$
- edges of T_r : between u and its first non-descendant found after u in a clockwise pre-ordering of T_b.



[Bonichon, Gavoille, Hanusse, 2005]

- edges of $G \rightarrow T_r$, T_b
- edges of T_r : between u and its first non-descendant found after u in a clockwise pre-ordering of T_b.

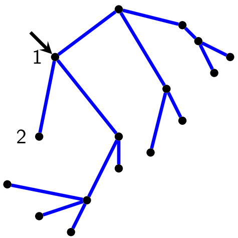


- x : ccw pre-order depth-first
- y : ccw post-order depth-first

Rook-drawing for outerplanar graphs

[Bonichon, Gavoille, Hanusse, 2005]

- edges of $G \rightarrow T_r$, T_b
- edges of T_r : between u and its first non-descendant found after u in a clockwise pre-ordering of T_b.

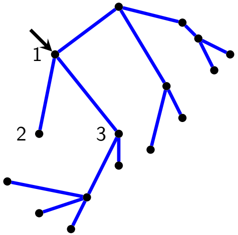


- x : ccw pre-order depth-first
- y : ccw post-order depth-first

Rook-drawing for outerplanar graphs

[Bonichon, Gavoille, Hanusse, 2005]

- edges of $G \rightarrow T_r, T_b$
- edges of T_r : between u and its first non-descendant found after u in a clockwise pre-ordering of T_b.

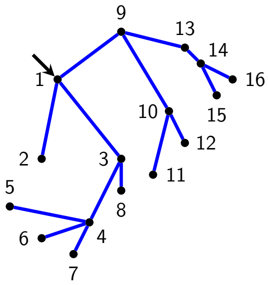


- x : ccw pre-order depth-first
- y : ccw post-order depth-first

Rook-drawing for outerplanar graphs

[Bonichon, Gavoille, Hanusse, 2005]

- edges of $G \rightarrow T_r$, T_b
- edges of T_r : between u and its first non-descendant found after u in a clockwise pre-ordering of T_b.

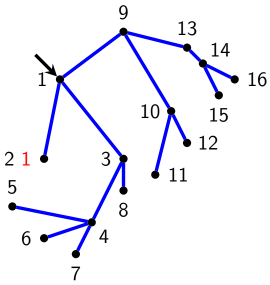


- x : ccw pre-order depth-first
- y : ccw post-order depth-first

Rook-drawing for outerplanar graphs

[Bonichon, Gavoille, Hanusse, 2005]

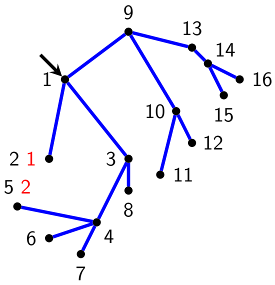
- edges of $G \to T_r$, T_b
- edges of T_r : between u and its first non-descendant found after u in a clockwise pre-ordering of T_b.



- x : ccw pre-order depth-first
- y : ccw post-order depth-first

[Bonichon, Gavoille, Hanusse, 2005]

- edges of $G \rightarrow T_r$, T_b
- edges of T_r : between u and its first non-descendant found after u in a clockwise pre-ordering of T_b.

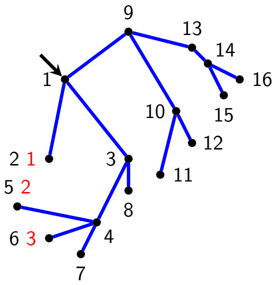


- x : ccw pre-order depth-first
- y : ccw post-order depth-first

Rook-drawing for outerplanar graphs

[Bonichon, Gavoille, Hanusse, 2005]

- edges of $G \rightarrow T_r$, T_b
- edges of T_r : between u and its first non-descendant found after u in a clockwise pre-ordering of T_b.

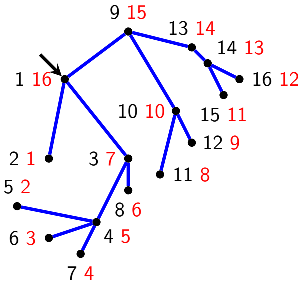


- x : ccw pre-order depth-first
- y : ccw post-order depth-first

Rook-drawing for outerplanar graphs

[Bonichon, Gavoille, Hanusse, 2005]

- edges of $G \to T_r$, T_b
- edges of T_r : between u and its first non-descendant found after u in a clockwise pre-ordering of T_b.

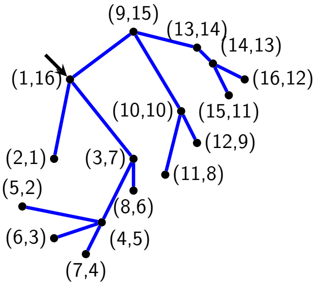


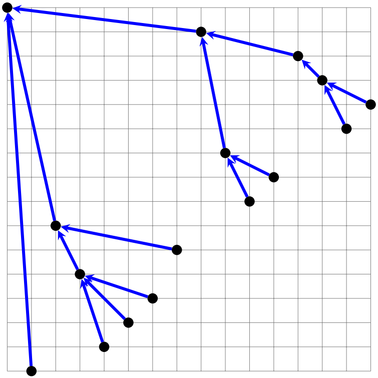
- x : ccw pre-order depth-first
- y : ccw post-order depth-first

Rook-drawing for outerplanar graphs

[Bonichon, Gavoille, Hanusse, 2005]

- edges of $G \to T_r$, T_b
- edges of T_r : between u and its first non-descendant found after u in a clockwise pre-ordering of T_b.



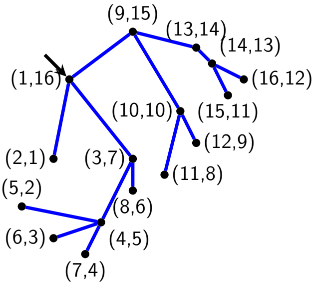


- x : ccw pre-order depth-first
- y : ccw post-order depth-first

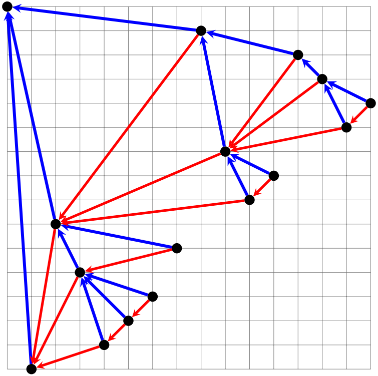
Rook-drawing for outerplanar graphs

[Bonichon, Gavoille, Hanusse, 2005]

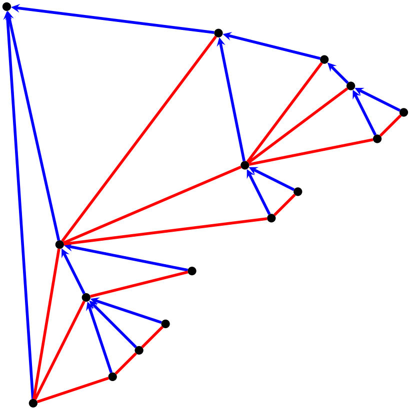
- edges of $G \rightarrow T_r$, T_b
- edges of T_r : between u and its first non-descendant found after u in a clockwise pre-ordering of T_b.

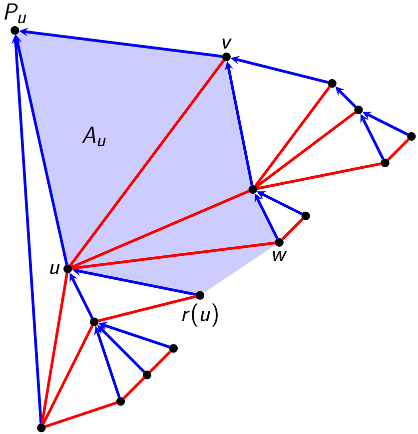


- x : ccw pre-order depth-first
- y : ccw post-order depth-first



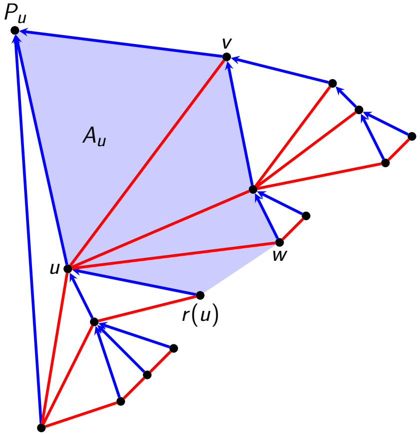
Rook-drawing for outerplanar graphs





For each vertex u not a leaf of T_r : define an area A_u with only red edges leading to u (the areas A_i are disjoint).

Rook-drawing for outerplanar graphs

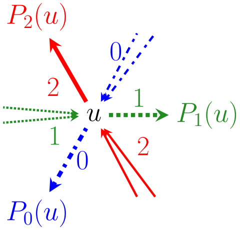


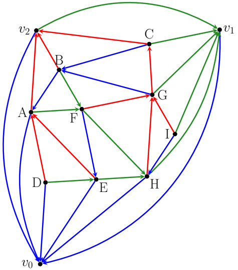
For each vertex u not a leaf of T_r : define an area A_u with only red edges leading to u (the areas A_i are disjoint).

The drawing is planar within A_u and blue and red edges can not cross.

Schnyder woods

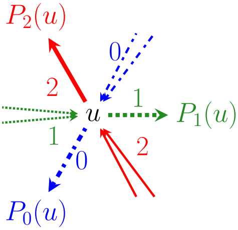
A Schnyder wood is a partition of the edges of a triangulation in three trees T_0, T_1 and T_2 (directed toward the root) and with a particular configuration around each node :

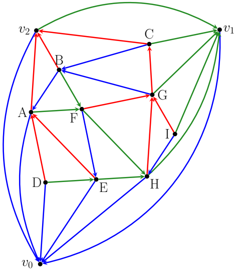




Schnyder woods

A Schnyder wood is a partition of the edges of a triangulation in three trees T_0, T_1 and T_2 (directed toward the root) and with a particular configuration around each node :





[Schnyder 1989]

Every plane triangulation admits at least one Schnyder wood, and it can be computed in linear time.

Polyline rook-drawing for planar graphs

We consider G a maximal plane graph (with exterior nodes v_0, v_1 and v_2).

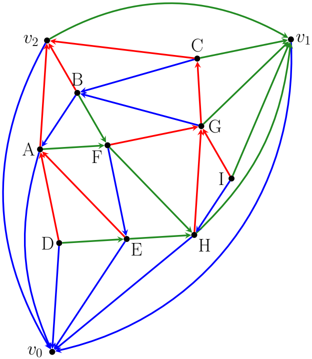
Main result

Every planar graph with n nodes admits a planar polyline rook-drawing, with at most $n-2$ bends (at most one per edge). Such a drawing can be computed in linear time.

Proof : based on an algorithm of [Bonichon, Mosbah, Le Saëc, 2002] optimizing the area of a polyline drawing.

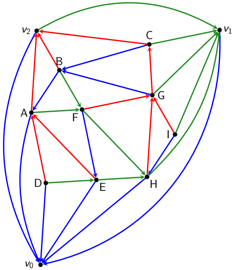
Planar polyline rook-drawing - Nodes

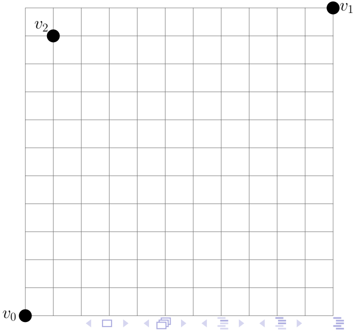
- (T_0, T_1, T_2) : Schnyder wood of G.



Planar polyline rook-drawing - Nodes

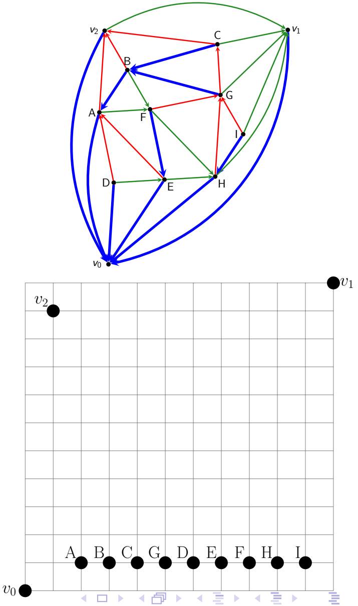
- (T_0, T_1, T_2) : Schnyder wood of G.
- $v_0 = (0, 0)$, $v_1 = (n-1, n-1)$, $v_2 = (1, n-2)$.





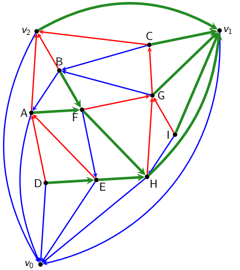
Planar polyline rook-drawing - Nodes

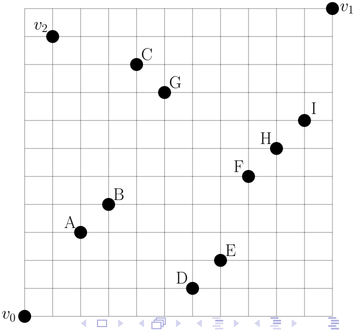
- (T_0, T_1, T_2) : Schnyder wood of G.
- $v_0 = (0,0)$, $v_1 = (n-1, n-1)$, $v_2 = (1, n-2)$.
- x : clockwise preordering of T_0 $= \{ABCGDEFHI\}$.



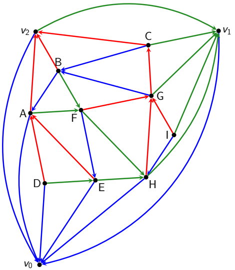
Planar polyline rook-drawing - Nodes

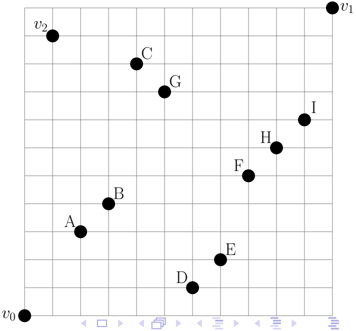
- (T_0, T_1, T_2) : Schnyder wood of G.
- $v_0 = (0,0)$, $v_1 = (n-1, n-1)$, $v_2 = (1, n-2)$.
- x : clockwise preordering of T_0 $= \{ABCGDEFHI\}$.
- y : clockwise postordering of $T_1 = \{DEABFHIGC\}$.





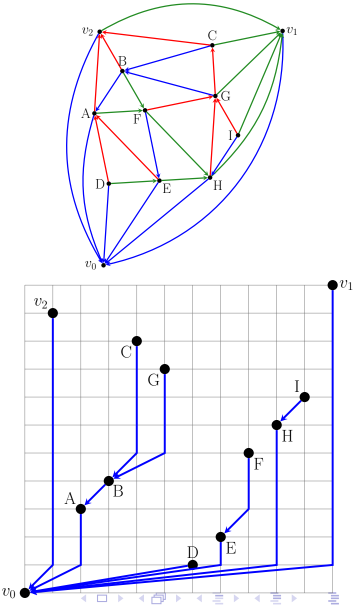
- (T_0, T_1, T_2) : Schnyder wood of G.
- $v_0 = (0,0)$, $v_1 = (n-1, n-1)$, $v_2 = (1, n-2)$.
- x : clockwise preordering of T_0 $= \{ABCGDEFHI\}$.
- y : clockwise postordering of $T_1 = \{DEABFHIGC\}$.





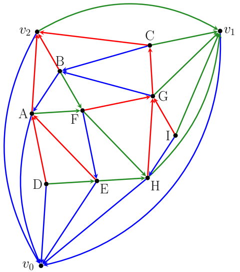
Planar polyline rook-drawing - Edges

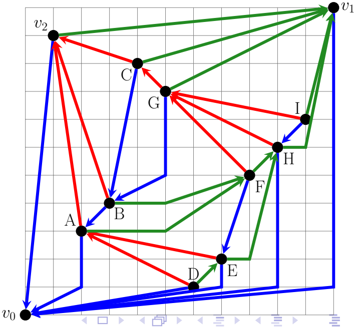
- The edges $(u, P_0(u))$ are bent at $(x(u), y(P_0(u)) + 1)$



Planar polyline rook-drawing - Edges

- The edges $(u, P_0(u))$ are bent at $(x(u), y(P_0(u)) + 1)$
- The edges $(u, P_1(u))$ are bent at $(x(\text{last descendant}(u)), y(u))$
- The edges $(u, P_2(u))$ are not bent.

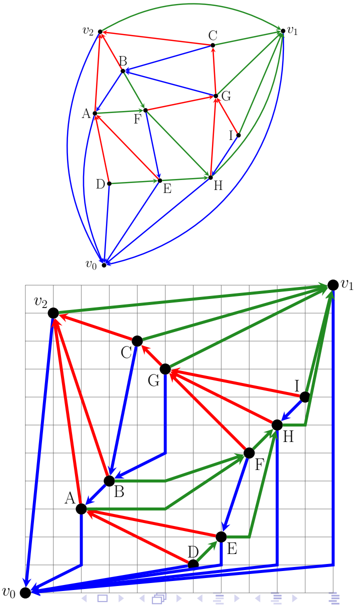




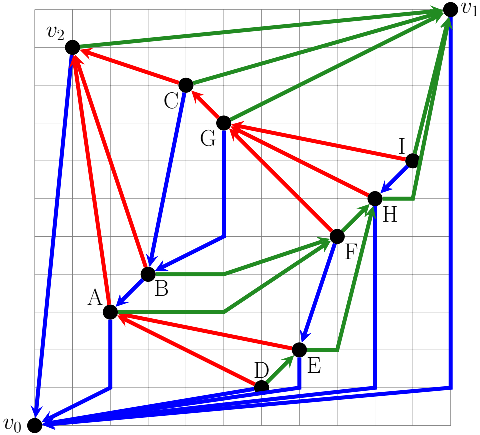
Planar polyline rook-drawing - Edges

- The edges $(u, P_0(u))$ are bent at $(x(u), y(P_0(u)) + 1)$
- The edges $(u, P_1(u))$ are bent at $(x(\text{last descendant}(u)), y(u))$
- The edges $(u, P_2(u))$ are not bent.

Around a node :
$\#bends = \#\{\text{blue children}\}$
$\rightarrow n - 2$ bends in the drawing

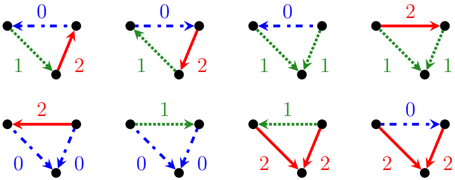


Proof of planarity



Schnyder wood properties

If u is a descendant of v in T_i, then u is not a parent of v in T_j, $j \neq i$.



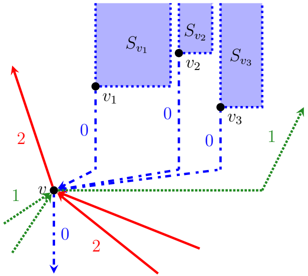
If u is parent of v in T_i, then u is before v in counterclockwise preordering of T_{i-1} and after v in counterclockwise preordering of T_{i+1}.

Edges direction

Edges direction

For each inner node u :

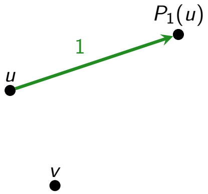
- $P_0(u)$ is left and below u.
- $P_1(u)$ is right and above u.
- $P_2(u)$ is left and above u.



- South-East : edges from children in T_2
- South : edge toward $P_0(v)$
- South-West : edges from children in T_1
- North-West : edge toward $P_2(v)$
- North-East : edges from children in T_0
- East : edge toward $P_1(v)$

Properties of green/red edges

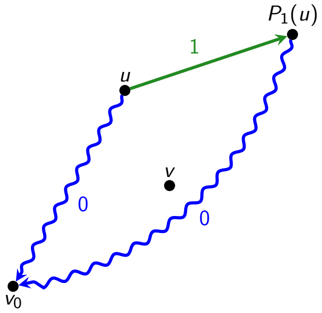
Every node v with $x(u) < x(v) < x(P_1(u))$ has $y(v) < y(u)$ if v is not a descendant of u in T_0.



v : between u and $P_1(u)$.

Properties of green/red edges

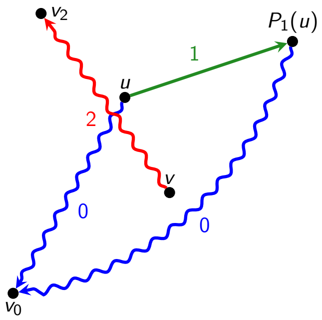
Every node v with $x(u) < x(v) < x(P_1(u))$ has $y(v) < y(u)$ if v is not a descendant of u in T_0.



v : between u and $P_1(u)$. $\rightarrow$ v in area (v_0, u), $(u, P_1(u))$, $(P_1(u), v_0)$.

Properties of green/red edges

Every node v with $x(u) < x(v) < x(P_1(u))$ has $y(v) < y(u)$ if v is not a descendant of u in T_0.

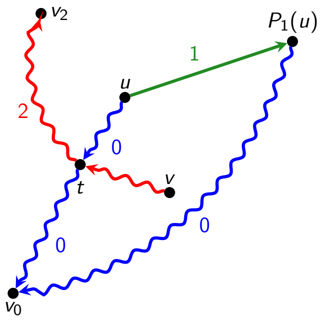


v : between u and $P_1(u)$. $\rightarrow$ v in area (v_0, u), $(u, P_1(u))$, $(P_1(u), v_0)$.

Path $v \rightarrow v_2$ has to "leave" the area !

Properties of green/red edges

Every node v with $x(u) < x(v) < x(P_1(u))$ has $y(v) < y(u)$ if v is not a descendant of u in T_0.



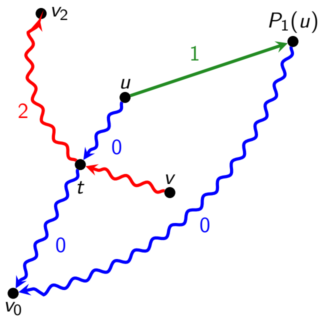
v : between u and $P_1(u)$. $\rightarrow$ v in area (v_0, u), $(u, P_1(u))$, $(P_1(u), v_0)$.

Path $v \rightarrow v_2$ has to “leave” the area !
Intersection t on path (v_0, u).

Paths $v \rightarrow t$ and $u \rightarrow t$.

Properties of green/red edges

Every node v with $x(u) < x(v) < x(P_1(u))$ has $y(v) < y(u)$ if v is not a descendant of u in T_0.



v : between u and $P_1(u)$. $\rightarrow$ v in area (v_0, u), $(u, P_1(u))$, $(P_1(u), v_0)$.

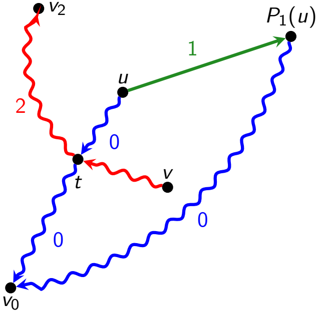
Path $v \rightarrow v_2$ has to "leave" the area ! Intersection t on path (v_0, u).

Paths $v \rightarrow t$ and $u \rightarrow t$.

$\rightarrow$ path from v to u is going upwards $= y(v) < y(u)$.

Properties of green/red edges

Every node v with $x(u) < x(v) < x(P_1(u))$ has $y(v) < y(u)$ if v is not a descendant of u in T_0.



v : between u and $P_1(u)$. $\rightarrow$ v in area (v_0, u), $(u, P_1(u))$, $(P_1(u), v_0)$.

Path $v \rightarrow v_2$ has to "leave" the area ! Intersection t on path (v_0, u).

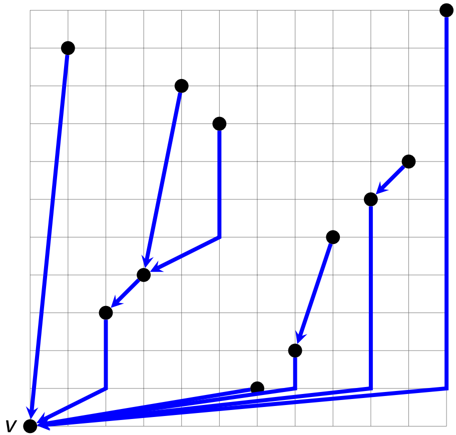
Paths $v \rightarrow t$ and $u \rightarrow t$.

$\rightarrow$ path from v to u is going upwards $= y(v) < y(u)$.

Every node v with $x(P_2(u)) < x(v) < x(u))$ has $y(v) < y(u)$ if v is not a descendant of $P_2(u)$ in T_0.

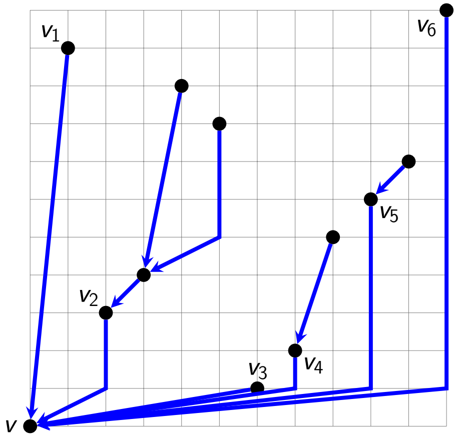
Non-crossing - blue

The edges of T_0 do not cross each other.



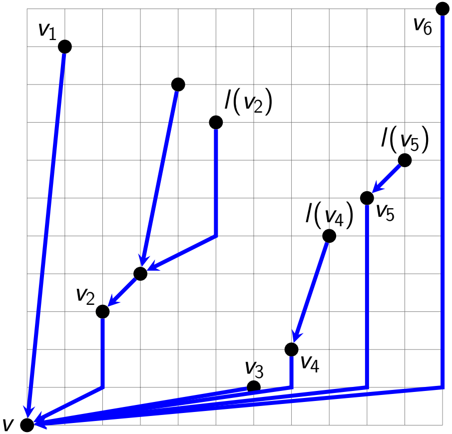
Non-crossing - blue

The edges of T_0 do not cross each other.



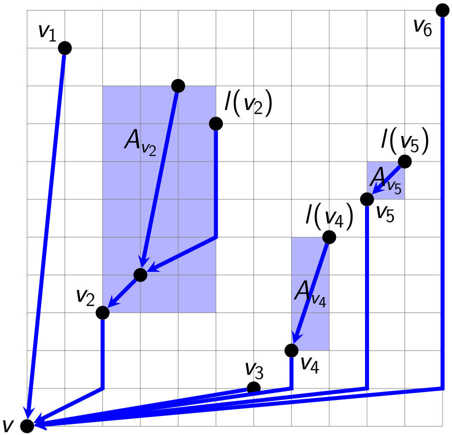
Non-crossing - blue

The edges of T_0 do not cross each other.



Non-crossing - blue

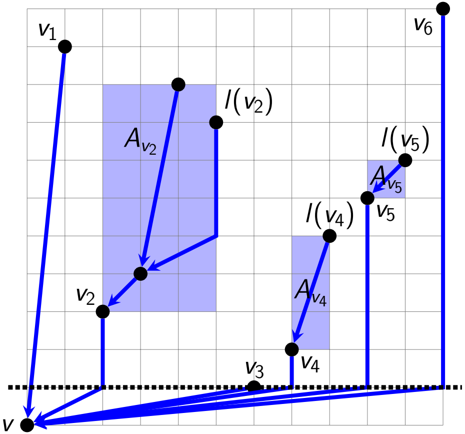
The edges of T_0 do not cross each other.



The subtrees of children "live" in different areas of width $(x(l(v_i)) - x(v_i))$.

Non-crossing - blue

The edges of T_0 do not cross each other.



The subtrees of children "live" in different areas of width $(x(l(v_i)) - x(v_i))$.

The edges to the children can not cross each other.

Non-crossing - green

The edges of T_1 do not cross each other.

Subtrees live in different areas (by construction). The bends : y-decreasing (by construction) ; x-increasing :

Non-crossing - green

The edges of T_1 do not cross each other.

Subtrees live in different areas (by construction). The bends : y-decreasing (by construction) ; x-increasing :

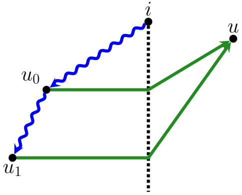
u_i descendant of u_{i+1} in T_0

Non-crossing - green

The edges of T_1 do not cross each other.

Subtrees live in different areas (by construction). The bends : y-decreasing (by construction) ; x-increasing :

u_i descendant of u_{i+1} in T_0

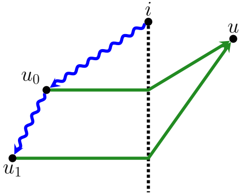


Non-crossing - green

The edges of T_1 do not cross each other.

Subtrees live in different areas (by construction). The bends : y-decreasing (by construction) ; x-increasing :

u_i descendant of u_{i+1} in T_0



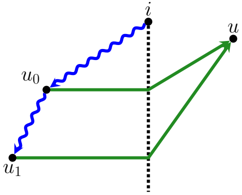
The last descendant of u_{i+1} is on the right to the one of u_i in T_0.

Non-crossing - green

The edges of T_1 do not cross each other.

Subtrees live in different areas (by construction). The bends : y-decreasing (by construction) ; x-increasing :

u_i descendant of u_{i+1} in T_0



The last descendant of u_{i+1} is on the right to the one of u_i in T_0.

u_i not descendant of u_{i+1} in T_0

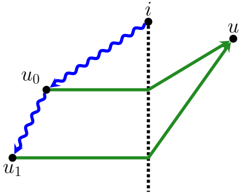
$x(u_i) > x(u_{i+1})$

Non-crossing - green

The edges of T_1 do not cross each other.

Subtrees live in different areas (by construction). The bends : y-decreasing (by construction) ; x-increasing :

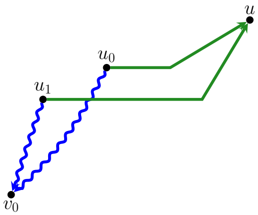
u_i descendant of u_{i+1} in T_0



The last descendant of u_{i+1} is on the right to the one of u_i in T_0.

u_i not descendant of u_{i+1} in T_0

$x(u_i) > x(u_{i+1})$

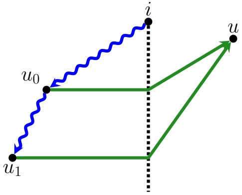


Non-crossing - green

The edges of T_1 do not cross each other.

Subtrees live in different areas (by construction). The bends : y-decreasing (by construction) ; x-increasing :

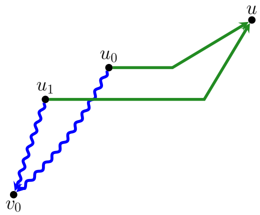
u_i descendant of u_{i+1} in T_0



The last descendant of u_{i+1} is on the right to the one of u_i in T_0.

u_i not descendant of u_{i+1} in T_0

$x(u_i) > x(u_{i+1})$



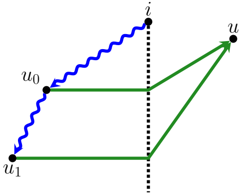
u_i should be below the edge (u_{i+1}, u)

Non-crossing - green

The edges of T_1 do not cross each other.

Subtrees live in different areas (by construction). The bends : y-decreasing (by construction) ; x-increasing :

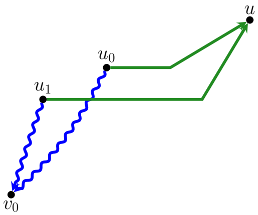
u_i descendant of u_{i+1} in T_0



The last descendant of u_{i+1} is on the right to the one of u_i in T_0.

u_i not descendant of u_{i+1} in T_0

$x(u_i) > x(u_{i+1})$ $x(u_i) < x(u_{i+1})$



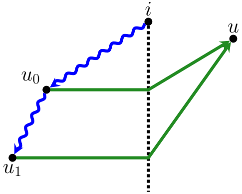
u_i should be below the edge (u_{i+1}, u)

Non-crossing - green

The edges of T_1 do not cross each other.

Subtrees live in different areas (by construction). The bends : y-decreasing (by construction) ; x-increasing :

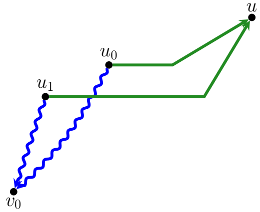
u_i descendant of u_{i+1} in T_0



The last descendant of u_{i+1} is on the right to the one of u_i in T_0.

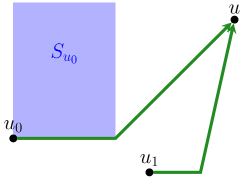
u_i not descendant of u_{i+1} in T_0

$x(u_i) > x(u_{i+1})$



u_i should be below the edge (u_{i+1}, u)

$x(u_i) < x(u_{i+1})$

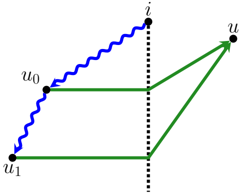


Non-crossing - green

The edges of T_1 do not cross each other.

Subtrees live in different areas (by construction). The bends : y-decreasing (by construction) ; x-increasing :

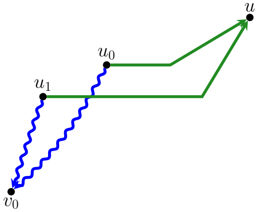
u_i descendant of u_{i+1} in T_0



The last descendant of u_{i+1} is on the right to the one of u_i in T_0.

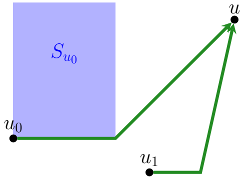
u_i not descendant of u_{i+1} in T_0

$x(u_i) > x(u_{i+1})$



u_i should be below the edge (u_{i+1}, u)

$x(u_i) < x(u_{i+1})$



Descendants of u_i are between $x(u_i)$ and $x(u_{i+1})$.

Conclusion

Open questions :

- Reduce the number of bends necessary to draw a given planar graph ?
- Caracterization of planar graphs for which a straight-lines rook-drawing is not possible / is possible
- What is the minimum grid size requested to draw a straight-lines rook-drawing ?

Thank you for your attention !