Orthogonal Drawings of Series-Parallel Graphs

by

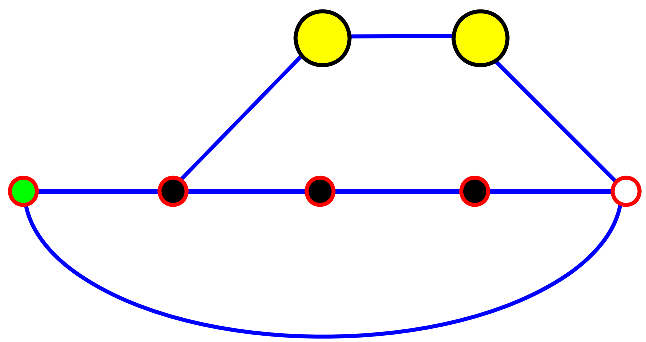
Takao Nishizeki

Joint work with Xiao Zhou

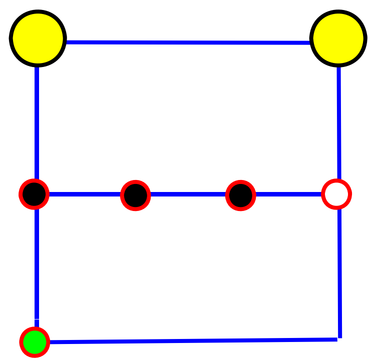
Tohoku University

Orthogonal Drawings of Series-Parallel Graphs with Minimum Bends

1. each vertex is mapped to a point
2. each edge is drawn as a sequence of alternate horizontal and vertical line segments
3. any two edges don't cross except at their common ends



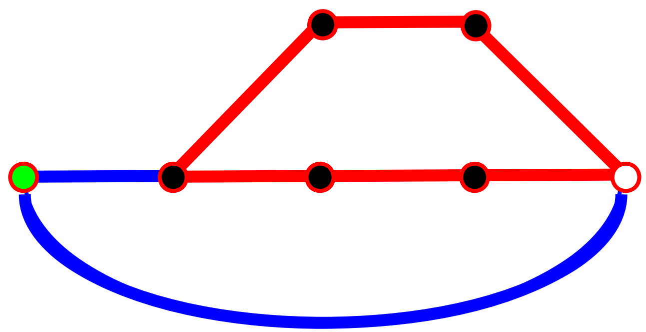
planar graph



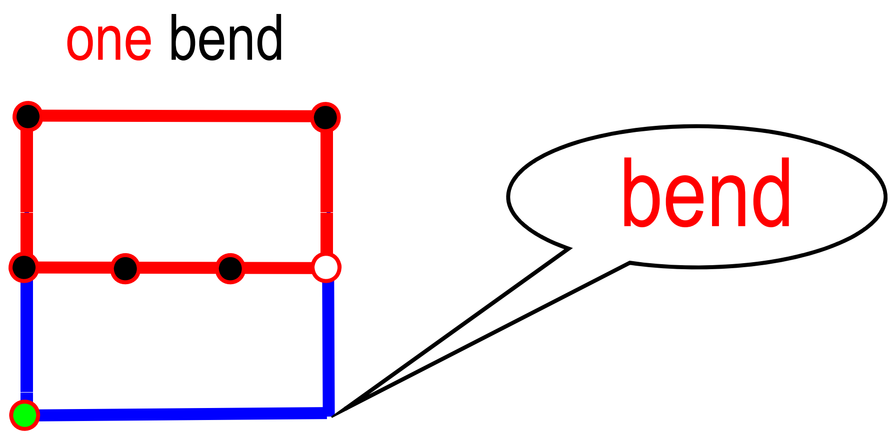
orthogonal drawings

Orthogonal Drawings of Series-Parallel Graphs with Minimum Bends

1. each vertex is mapped to a point
2. each edge is drawn as a sequence of alternate horizontal and vertical line segments
3. any two edges don't cross except at their common ends



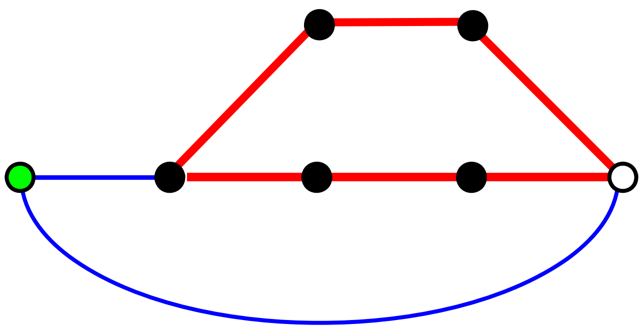
planar graph

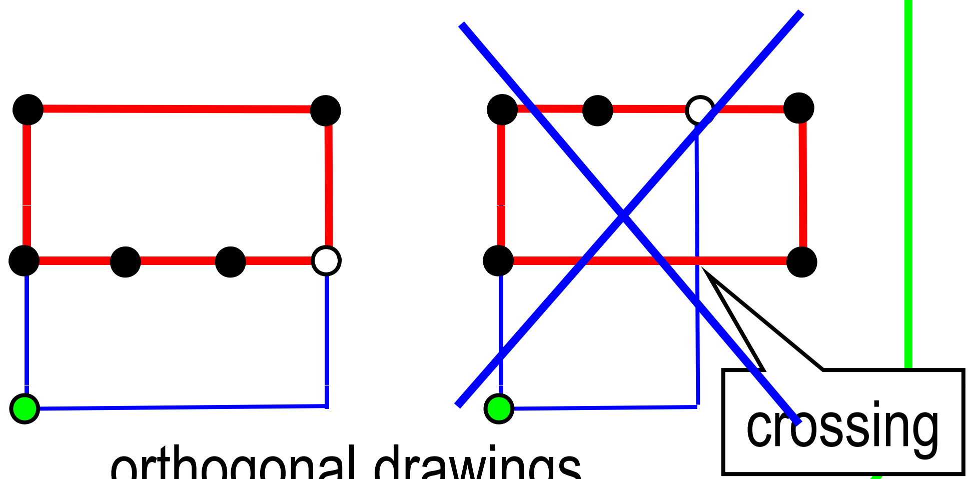


orthogonal drawings

Orthogonal Drawings of Series-Parallel Graphs with Minimum Bends

1. each vertex is mapped to a point
2. each edge is drawn as a sequence of alternate horizontal and vertical line segments
3. any two edges don't cross except at their common ends

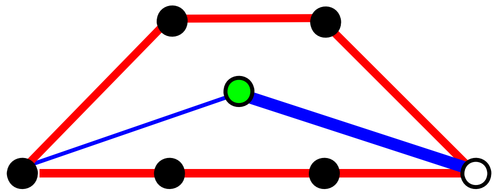

planar graph



orthogonal drawings

Orthogonal D… …arallel Graphs

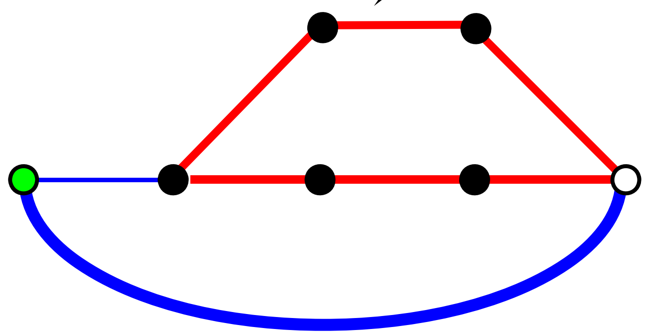
another embedding



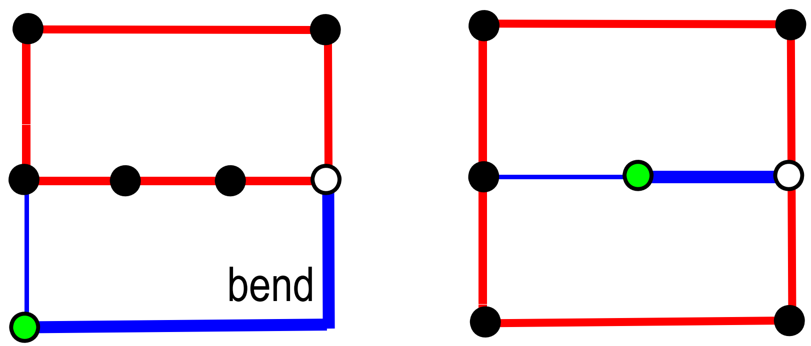
1. each vertex is …
2. each edge is dra… …quen… …nate horizontal and vertical line s… …nts
3. any two edges … …t cross except at their comm… …n ends

one bend

no bend



planar graph



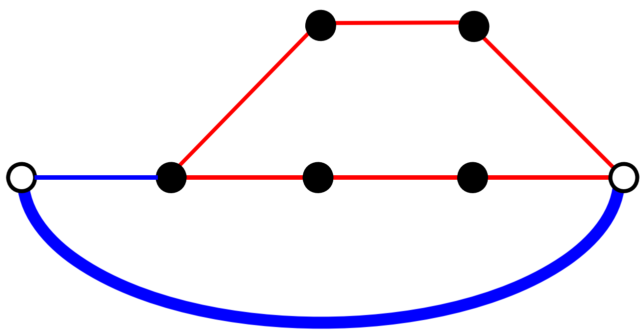
orthogonal drawings

Optimal orthogonal drawing

An orthogonal drawing of a planar graph *G* is optimal if it has the minimum # of bends among all possible orthogonal drawings of *G*.

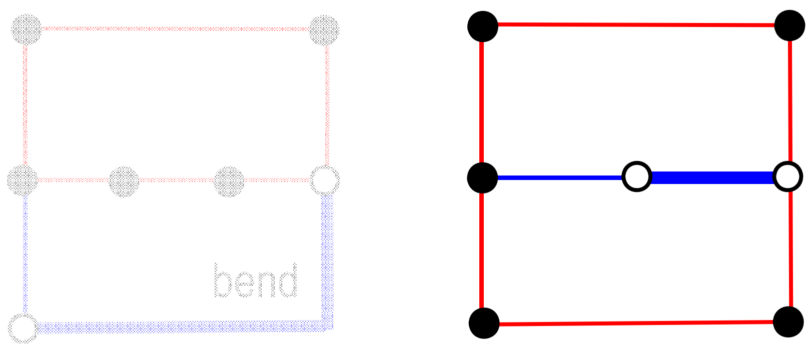
VLSI design

bend ↔ via-hole, through-hole



planar graph

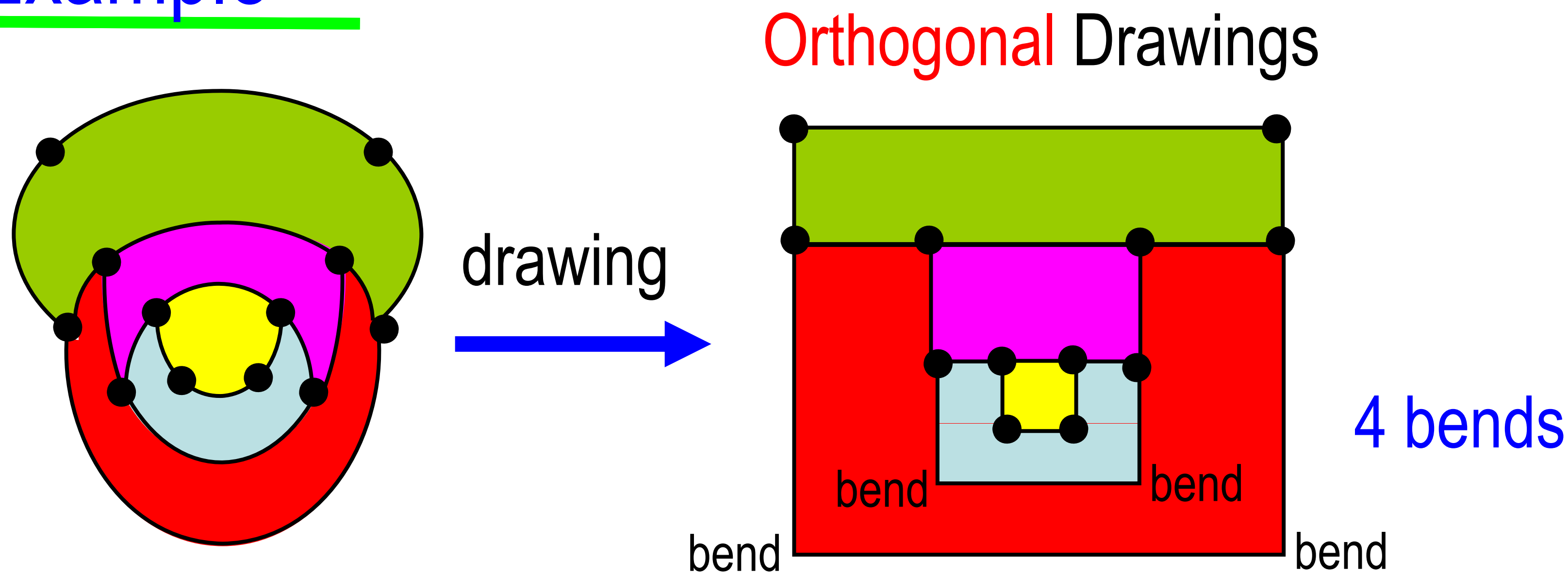
one bend no bend



orthogonal drawings

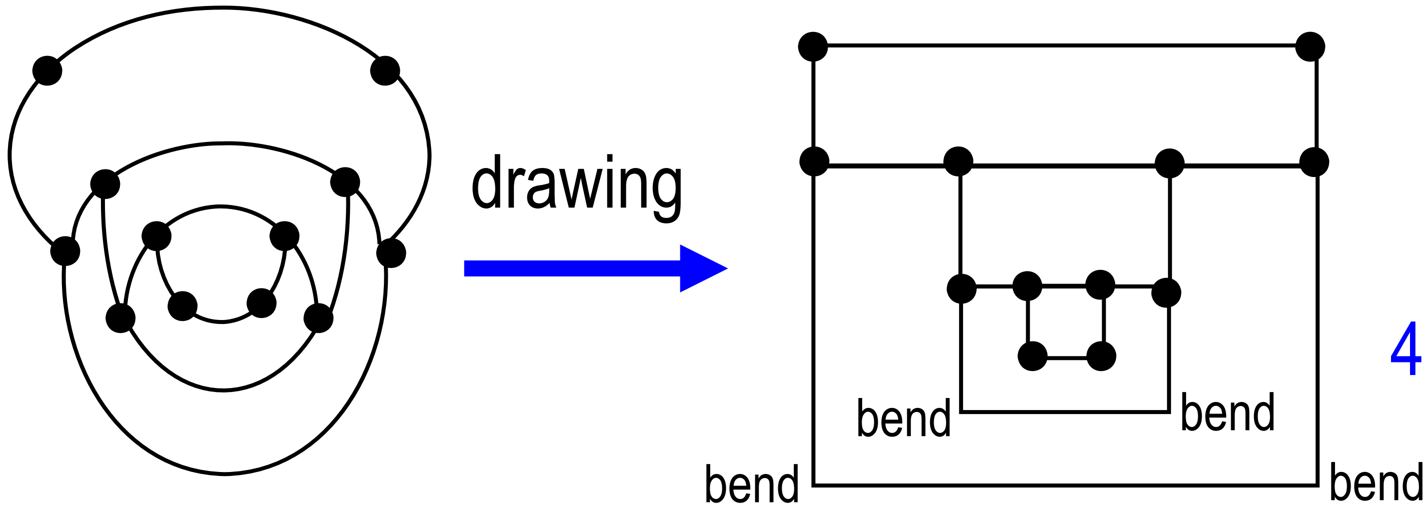
optimal

Example



Example

Orthogonal Drawings

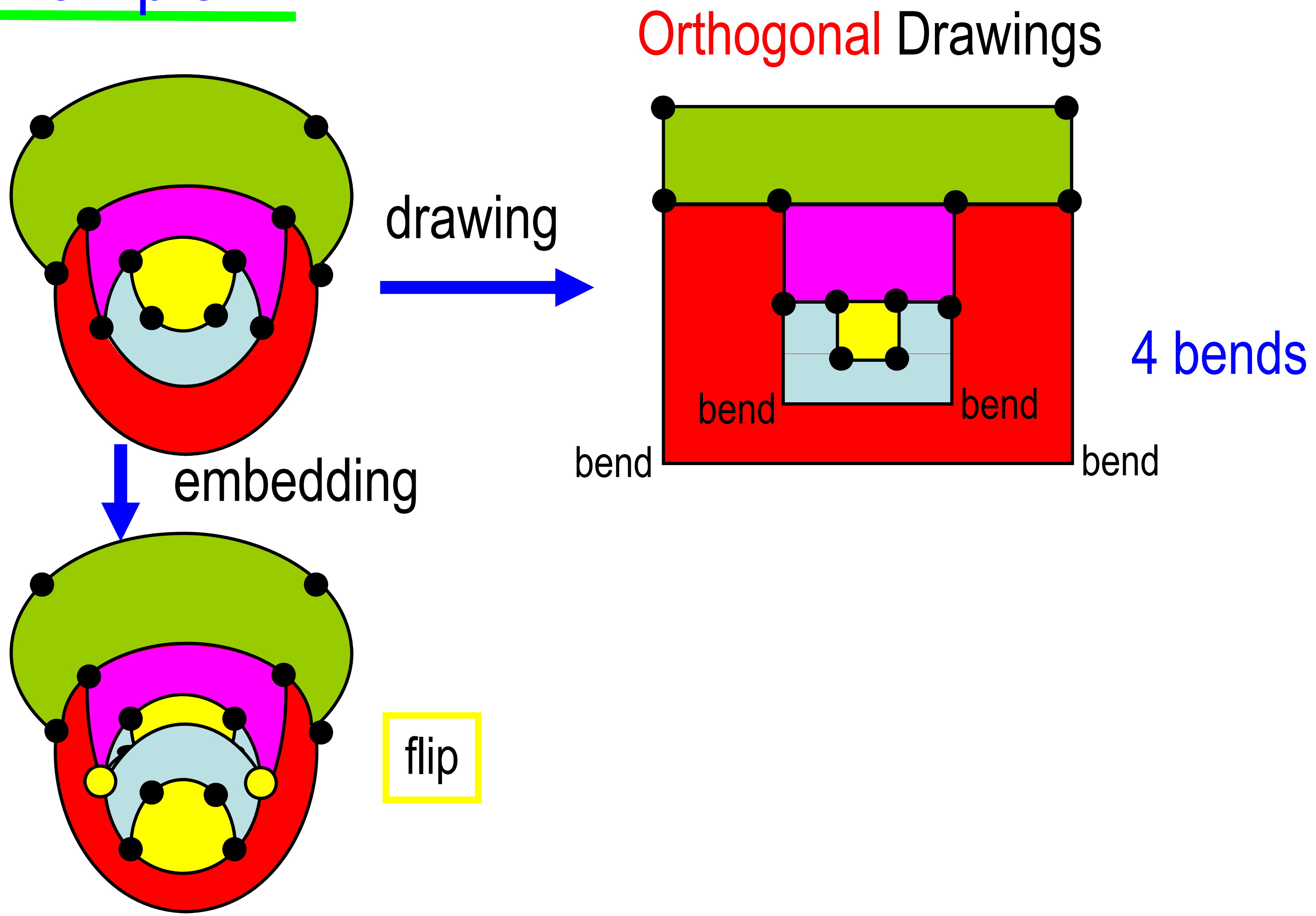


4 bends

Optimal ?

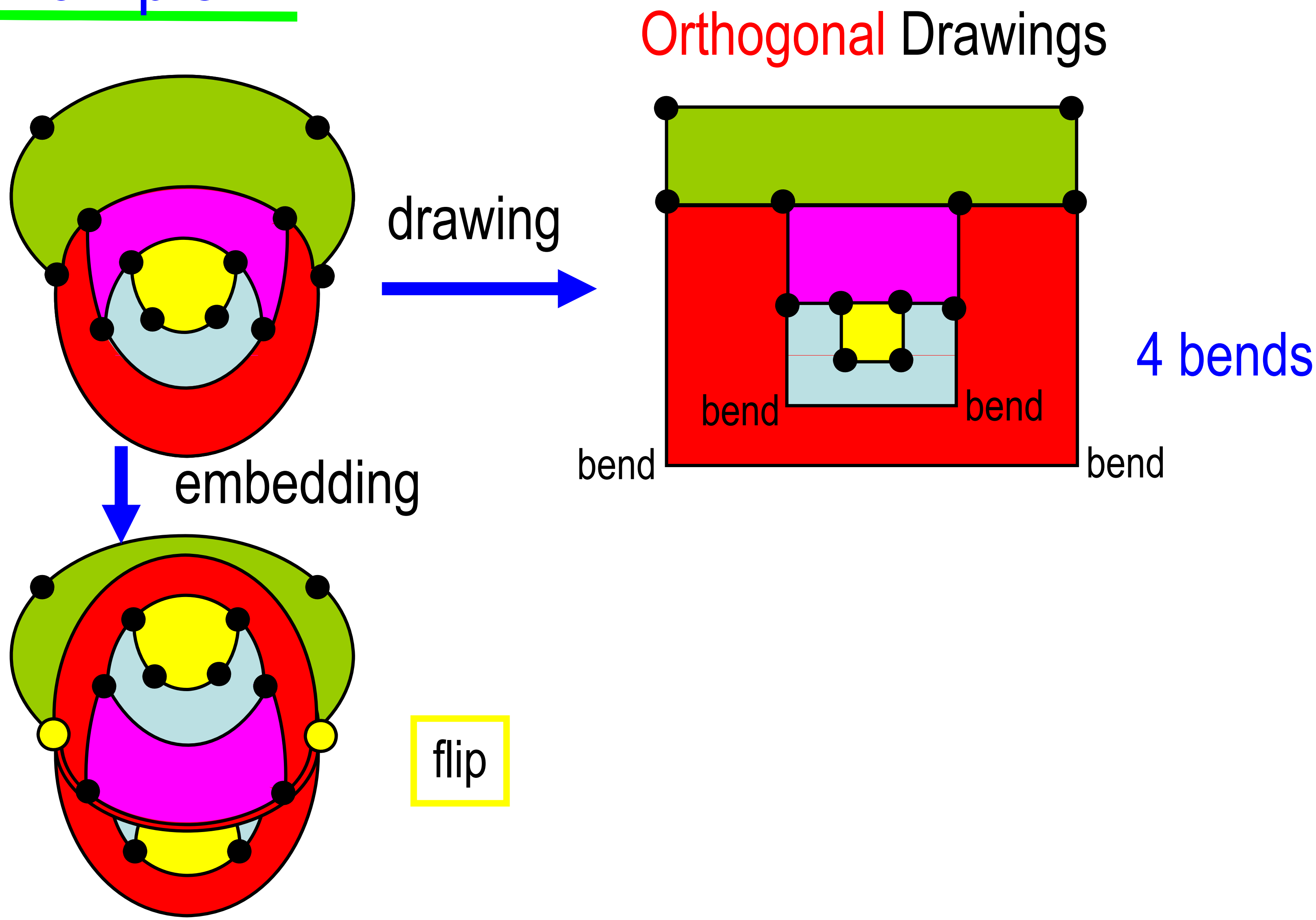
No

Example

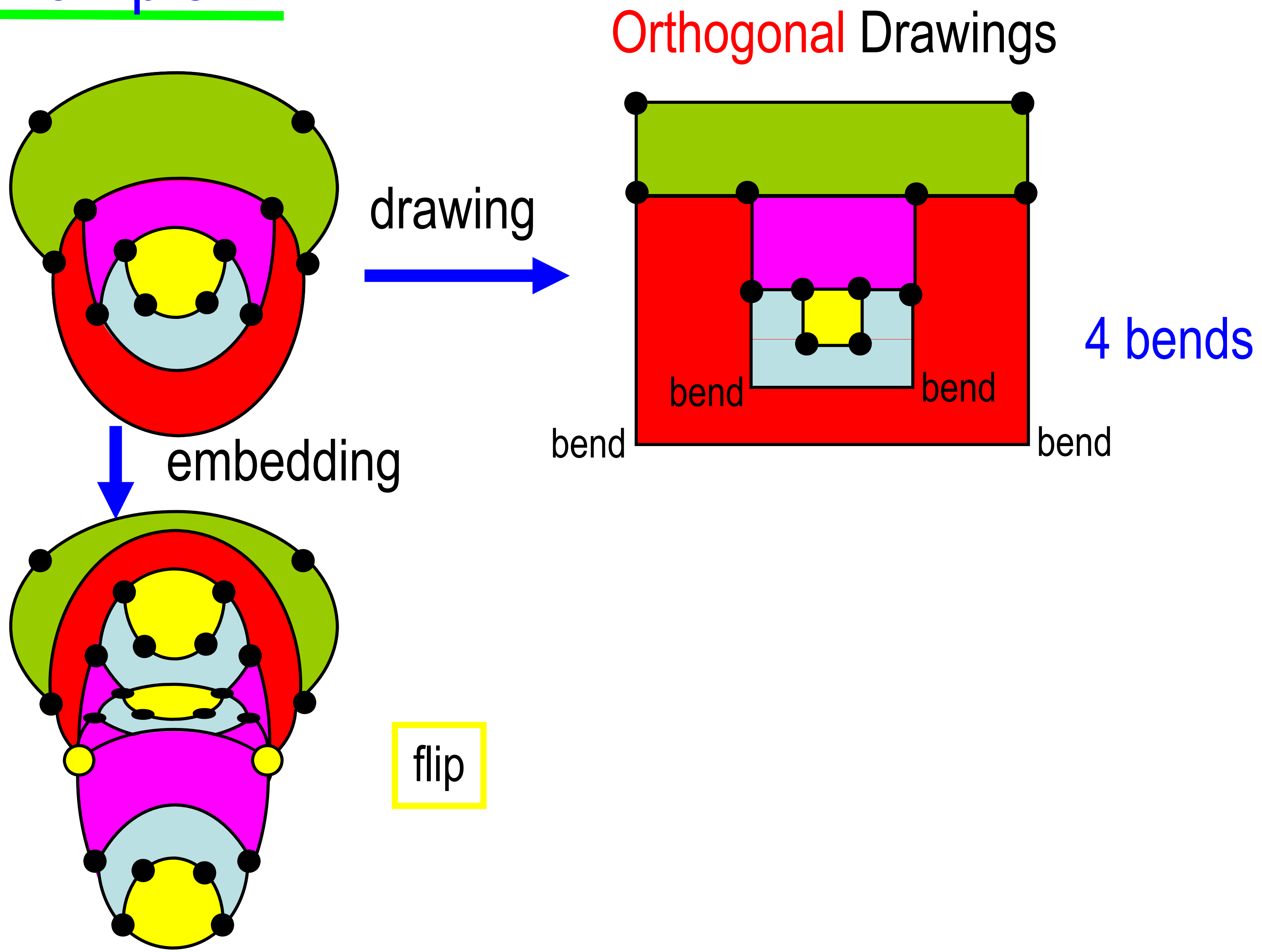


Orthogonal Drawings
drawing
embedding
flip
bend
bend
bend
bend
4 bends

Example


Orthogonal Drawings
drawing
embedding
flip
bend
bend
bend
bend
4 bends

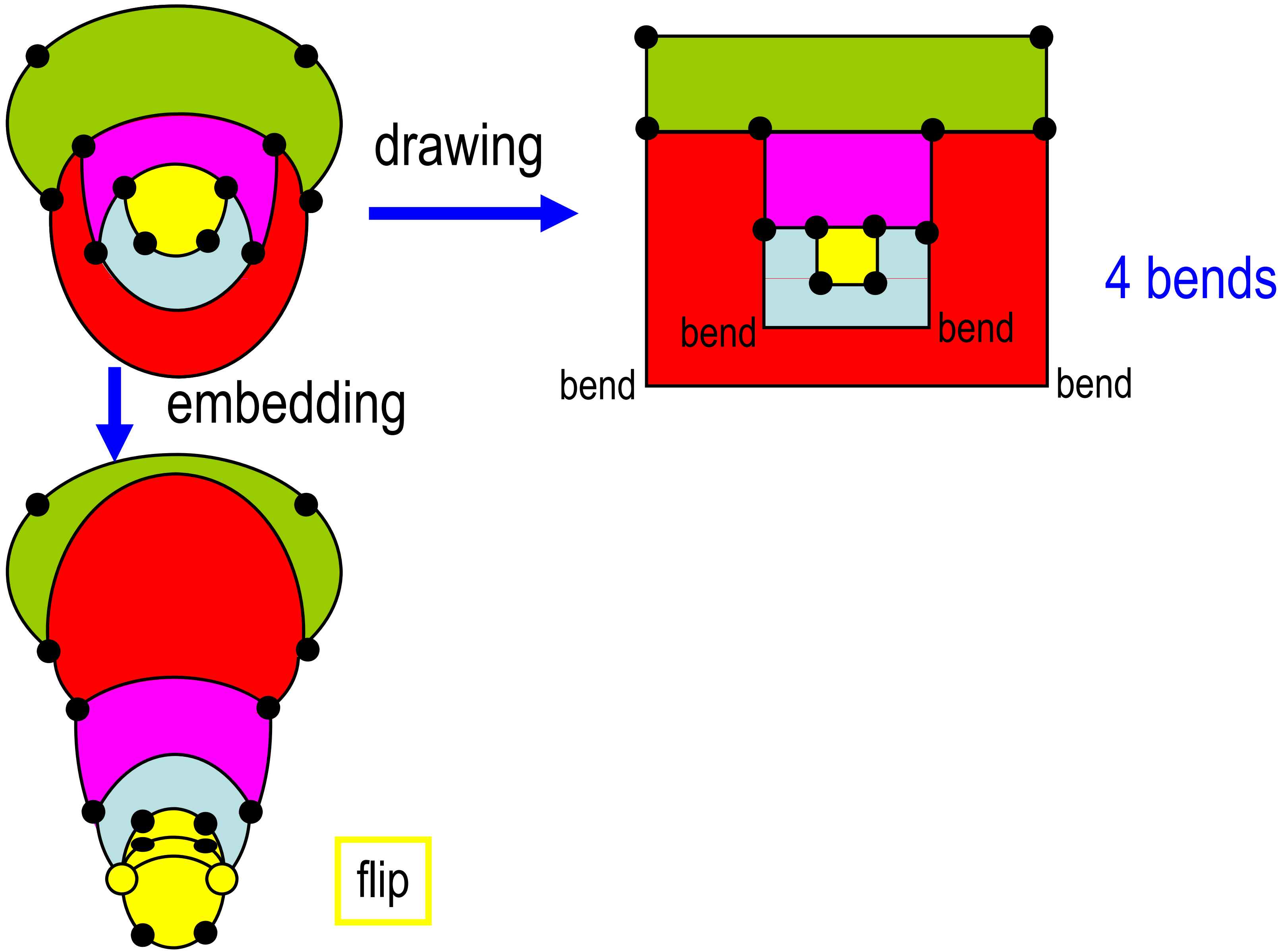
Example



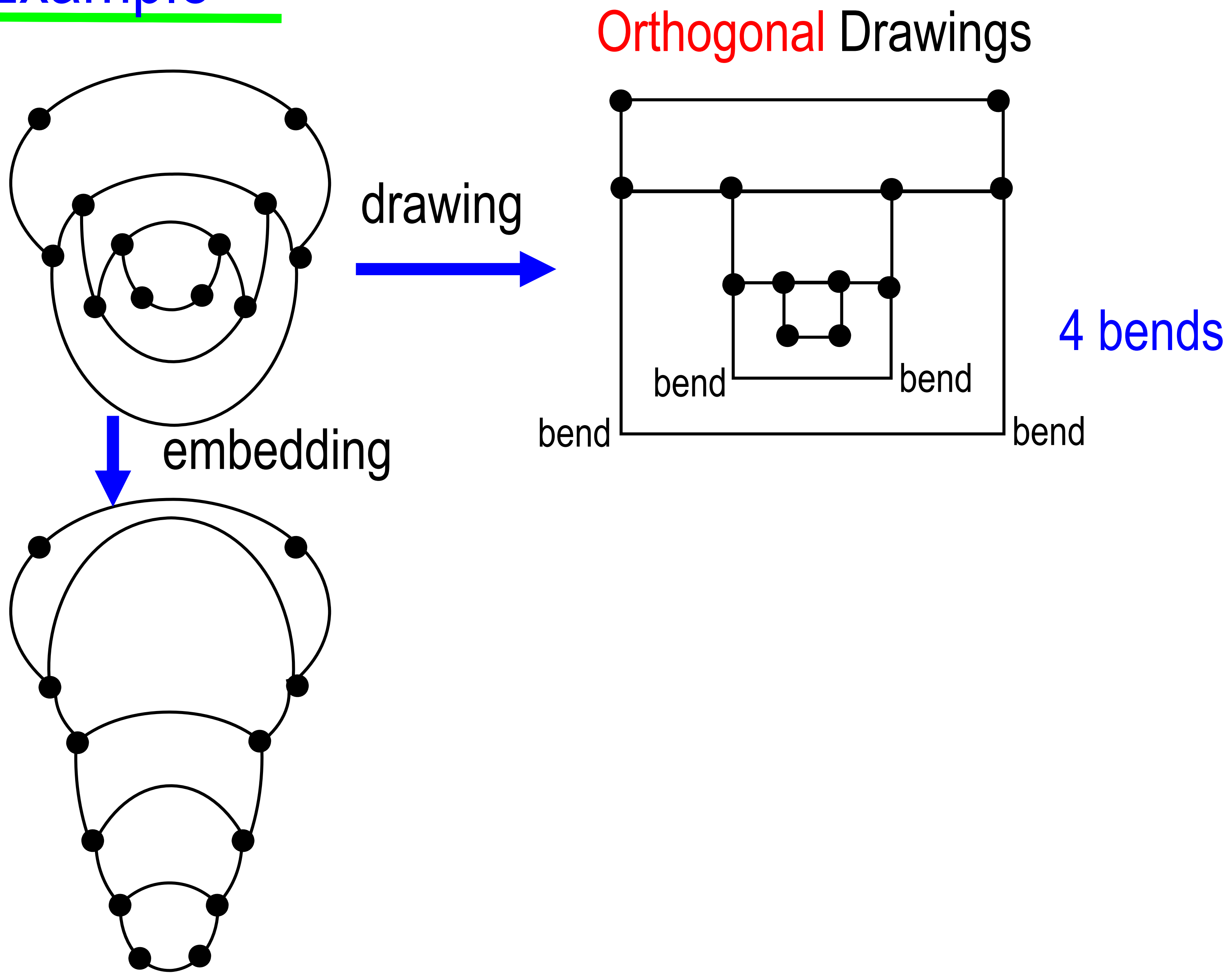
Orthogonal Drawings
drawing
4 bends
bend
bend
bend
bend
embedding
flip

Example

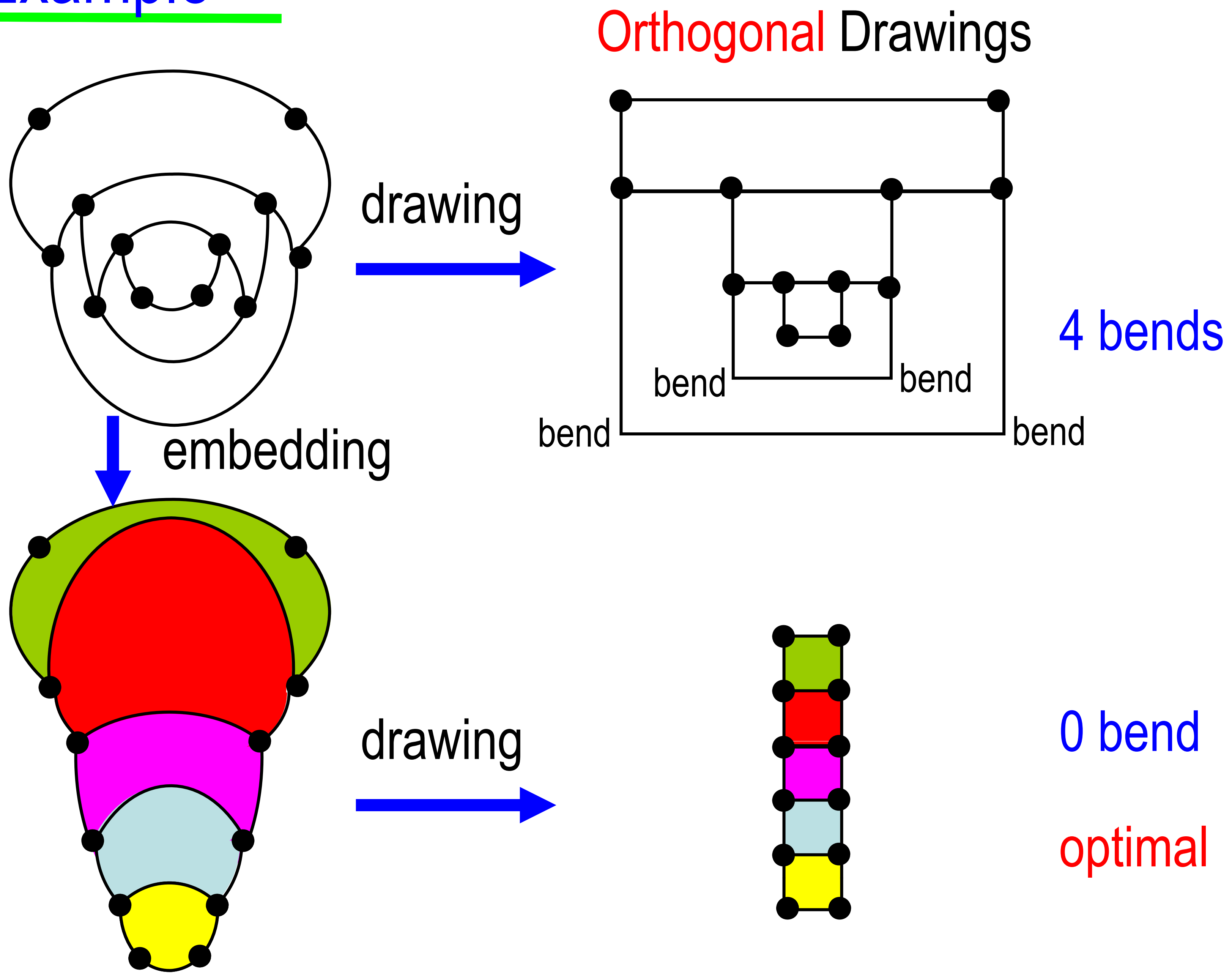
Orthogonal Drawings



Example

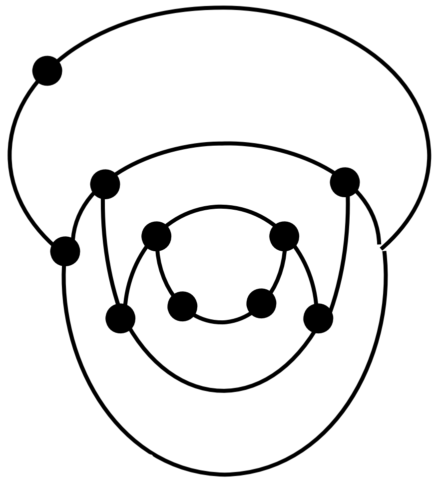

Orthogonal Drawings
drawing
embedding
bend
bend
bend
bend
4 bends

Example



Orthogonal Drawings
drawing
embedding
bend
bend
bend
bend
4 bends
drawing
0 bend
optimal

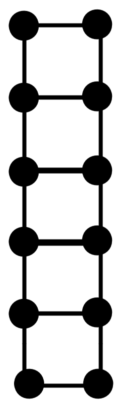
Given a planar graph



Problem

Find an optimal orthogonal drawing of a given planar graph.

Wish to find
an optimal orthogonal drawing



optimal

Optimal orthogonal drawing

An orthogonal drawing of a planar graph *G* is optimal if it has the minimum # of bends among all possible orthogonal drawings of *G*.

Problem

Find an optimal orthogonal drawing of a given planar graph.

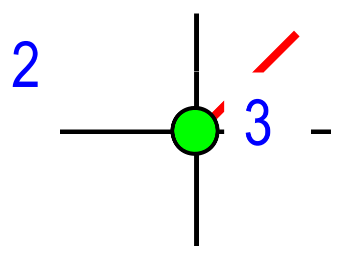
Known results

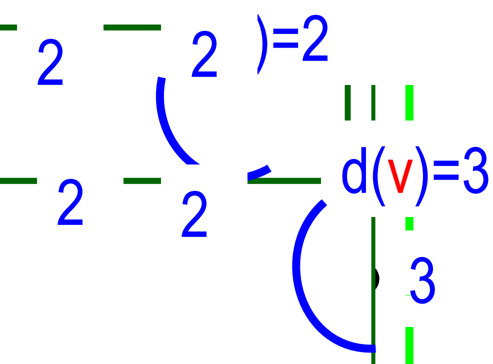
The problem:

NP-complete for planar graphs of $\Delta \leqq 4$

A. G R. Tamassia, 2001

If $\Delta \geqq 5$, then no orthogonal drawing.





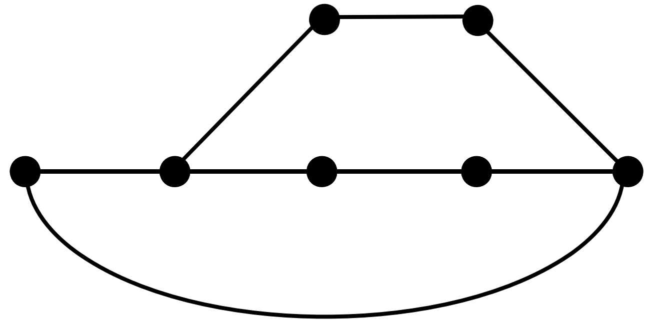
Known results

The problem:

NP-complete for planar graphs of $\Delta \leqq 4$

A. Garg, R. Tamassia, 2001

If $\Delta \leqq 3$, $O(n^5 \log n)$ time D. Battista, *et al*. 1998



Δ=3, n=7

where
n: # of vertices

Known results

The problem:

NP-complete for planar graphs of $\Delta \leqq 4$

A. Garg, R. Tamassia, 2001

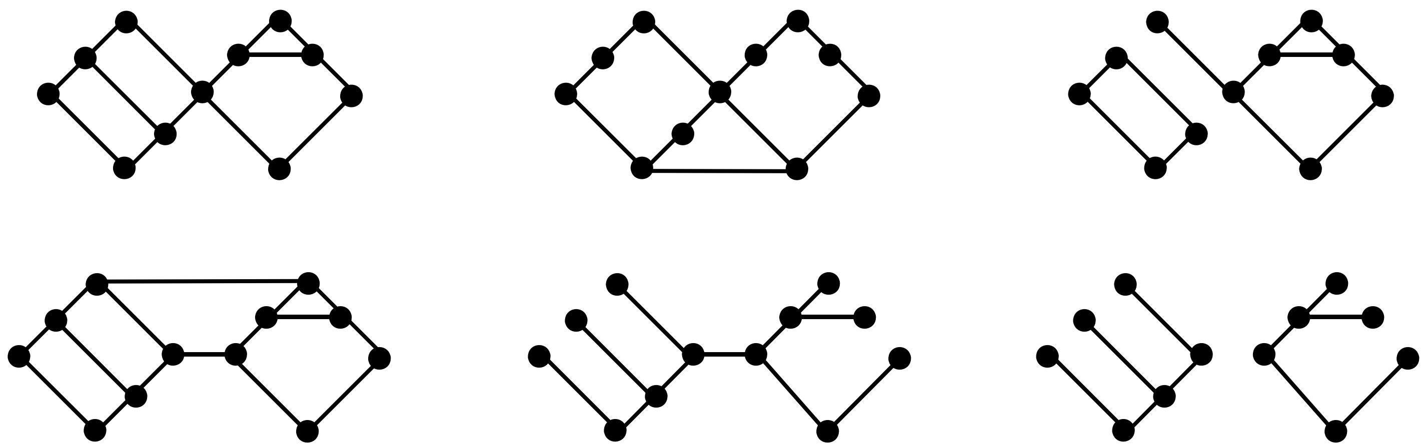
If $\Delta \leqq 3$, $O(n^5 \log n)$ time D. Battista, *et al.* 1998

For biconnected series-parallel graphs

If $\Delta \leqq 4$, $O(n^4)$ time D. Battista, *et al.* 1998

If $\Delta \leqq 3$, $O(n^3)$ time D. Battista, *et al.* 1998

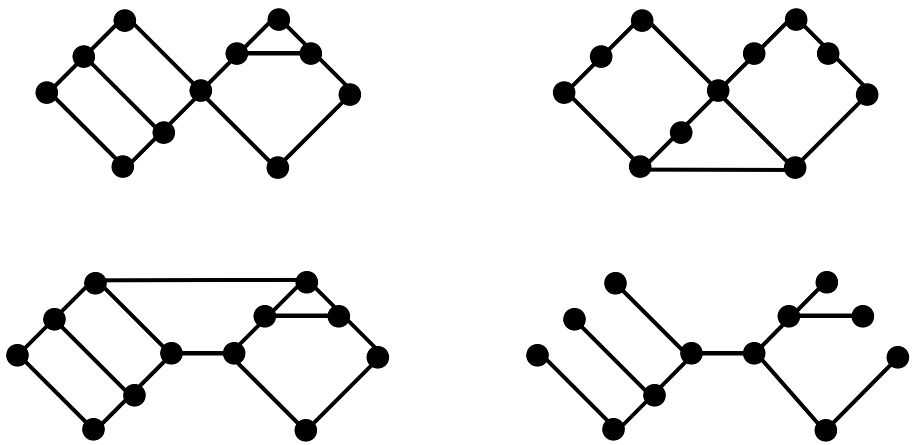
Connected graphs ?



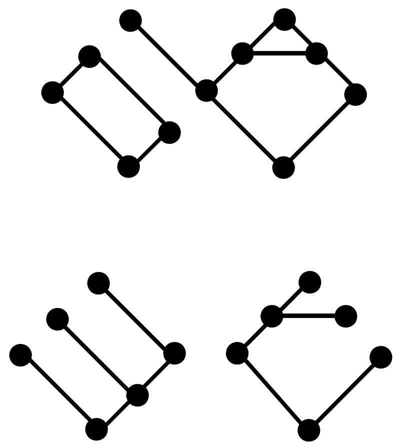
For biconnected series-parallel graphs

If $\Delta \leqq 4$,	$O(n^4)$ time	D. Battista, *et al.* 1998
If $\Delta \leqq 3$,	$O(n^3)$ time	D. Battista, *et al.* 1998

Connected graphs



Non-connected graphs

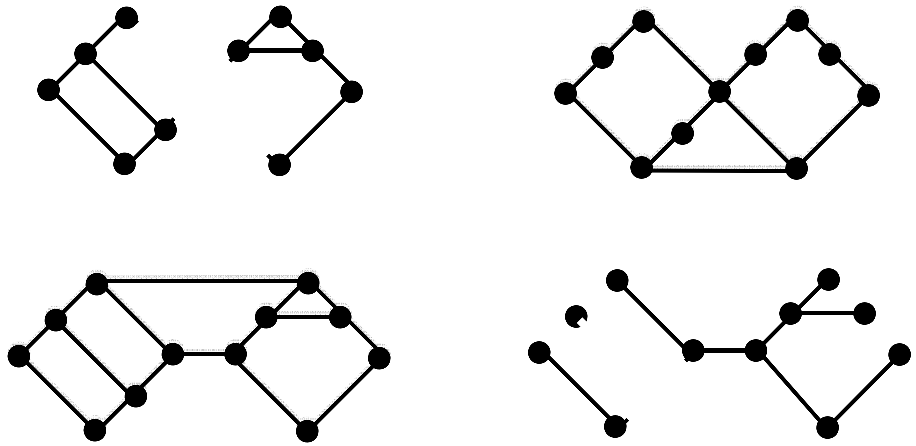


For biconnected series-parallel graphs

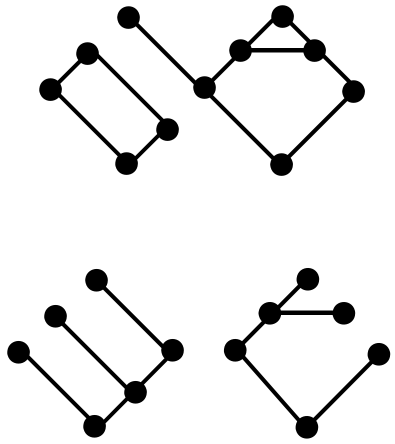
If $\Delta \leqq 4$, $O(n^4)$ time D. Battista, *et al.* 1998

If $\Delta \leqq 3$, $O(n^3)$ time D. Battista, *et al.* 1998

Connected graphs



Non-connected graphs



Biconnected graphs

Known results

The problem:

NP-complete for planar graphs of $\Delta \leqq 4$

A. Garg, R. Tamassia, 2001

If $\Delta \leqq 3$, $O(n^5 \log n)$ time D. Battista, *et al*. 1998

For biconnected series-parallel graphs

If $\Delta \leqq 4$, $O(n^4)$ time D. Battista, *et al*. 1998

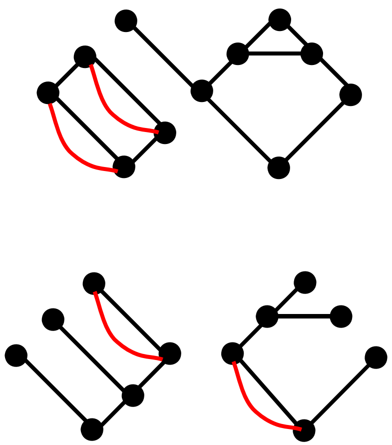
If $\Delta \leqq 3$, $O(n^3)$ time D. Battista, *et al*. 1998

Our results

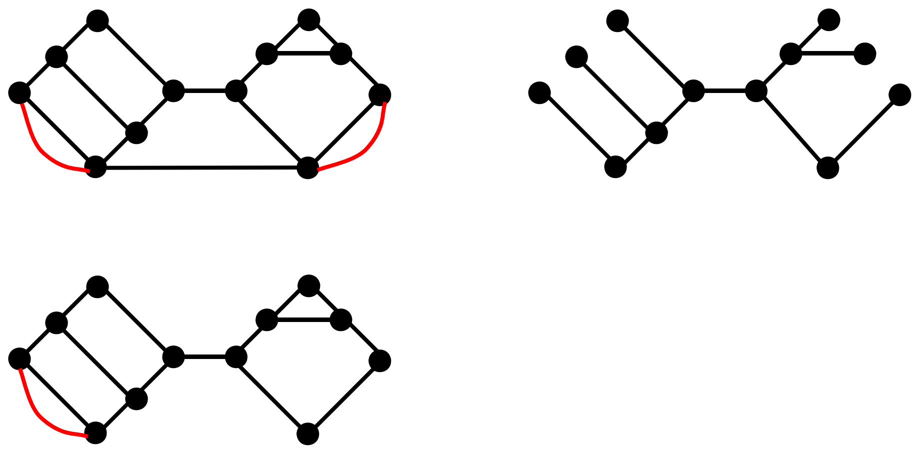
For series-parallel (multiple) graphs

If $\Delta \leqq 3$, $O(n)$ time **our result**

Non-connected



Connected



Our results

For series-parallel graphs

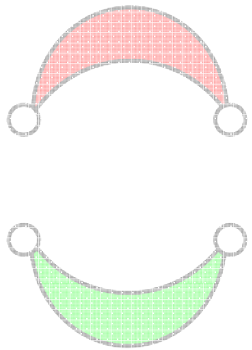
If $\Delta \leqq 3$, $O(n)$ time $\boxed{\text{our result}}$

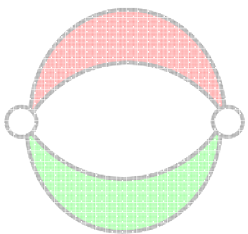
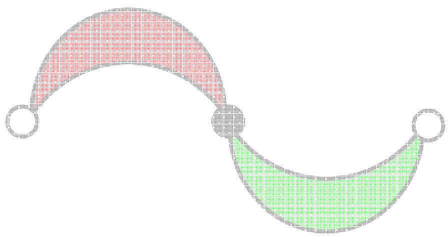
much simpler and faster than the known algorithms

Series-Parallel Graphs

A SP graph is recursively defined as follows:

(a)  a single edge is a SP graph.

(b) if  are SP graphs,

then   are SP graphs

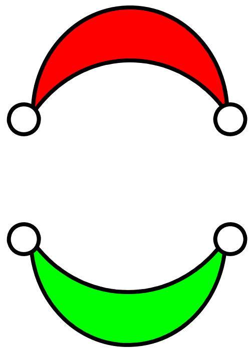
Series-Parallel Graphs

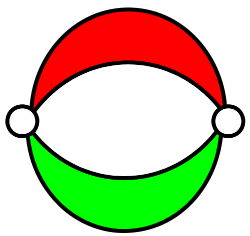
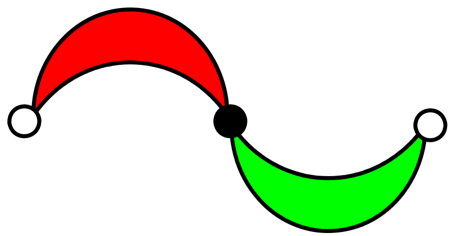
A SP graph is recursively defined as follows:

(a) 

a single edge

is a SP graph.

(b) if  are SP graphs,

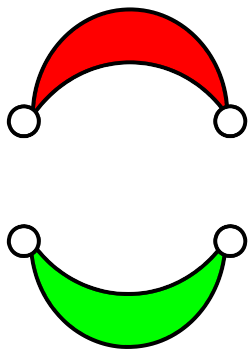
then   are SP graphs

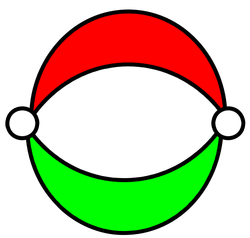
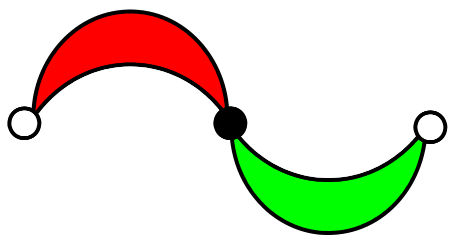
Series-Parallel Graphs

A SP graph is recursively defined as follows:

(a)  is a SP graph.

a single edge

(b) if  are SP graphs,

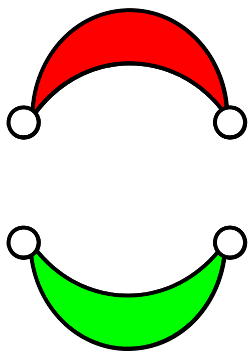
then   are SP graphs

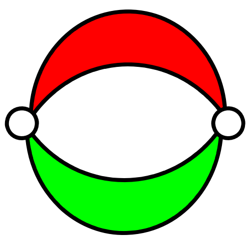
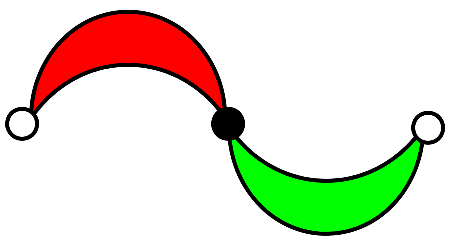
Series-Parallel Graphs

A SP graph is recursively defined as follows:

(a)  is a SP graph.

a single edge

(b) if  are SP graphs,

then   are SP graphs

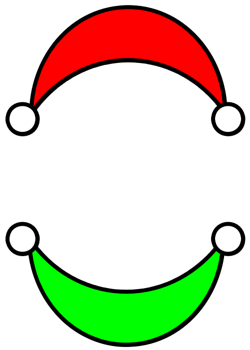
parallel-connection

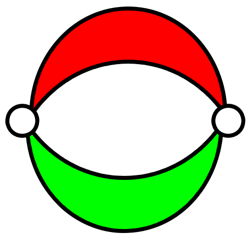
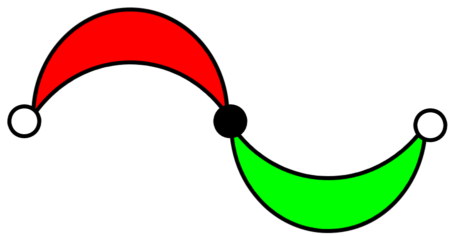
Series-Parallel Graphs

A SP graph is recursively defined as follows:

(a)  is a SP graph.

a single edge

(b) if  are SP graphs,

then   are SP graphs

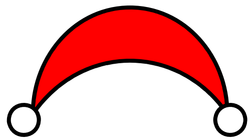
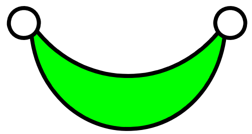
parallel-connection series-connection

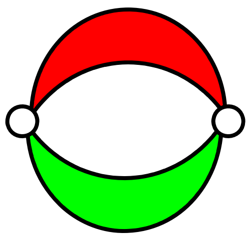
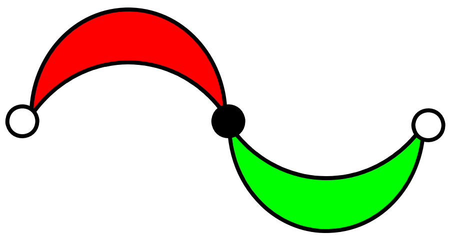
Series-Parallel Graphs

A SP graph is recursively defined as follows:

(a) 

a single edge is a SP graph.

(b) if   are SP graphs,

then   are SP graphs

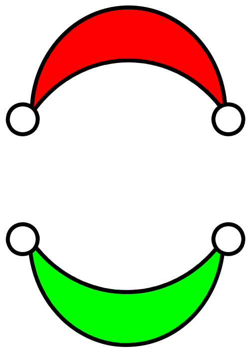
parallel-connection series-connection

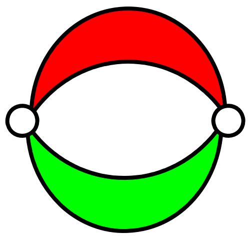
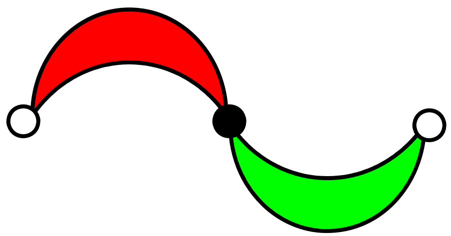
Series-Parallel Graphs

A SP graph is recursively defined as follows:

(a) 

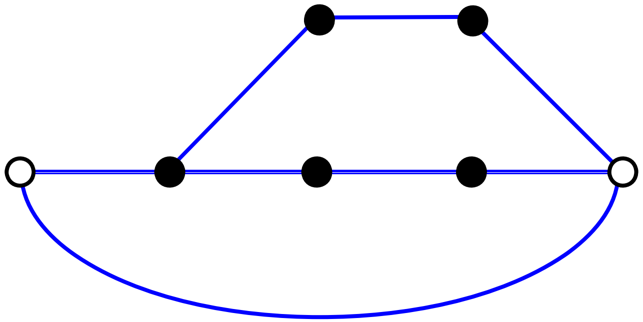
a single edge is a SP graph.

(b) if  are SP graphs,

then   are SP graphs

Series-Parallel Graphs

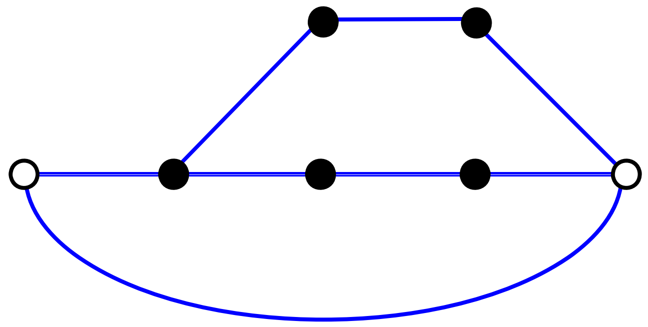
Example



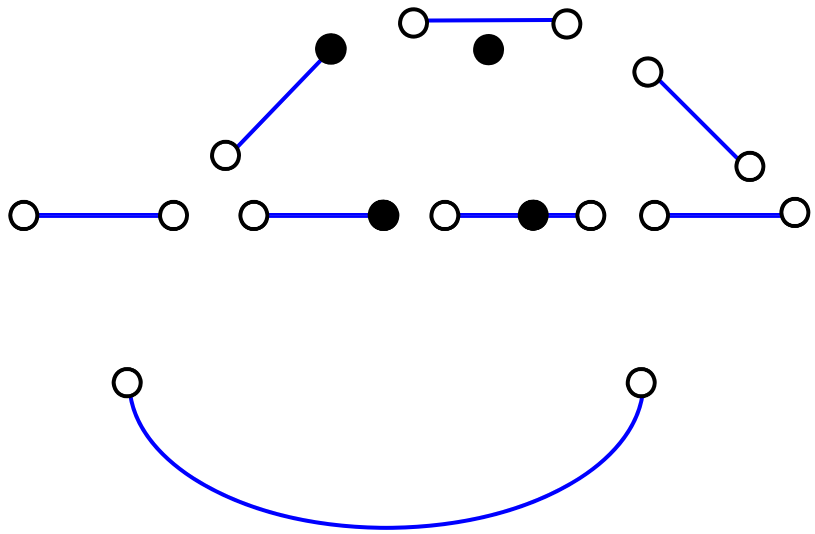
SP graph

Series-Parallel Graphs

Example



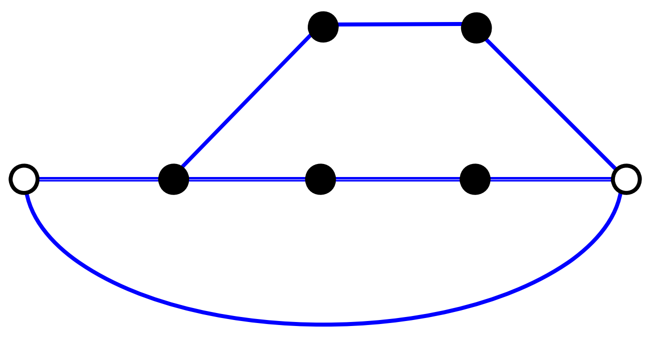
SP graph



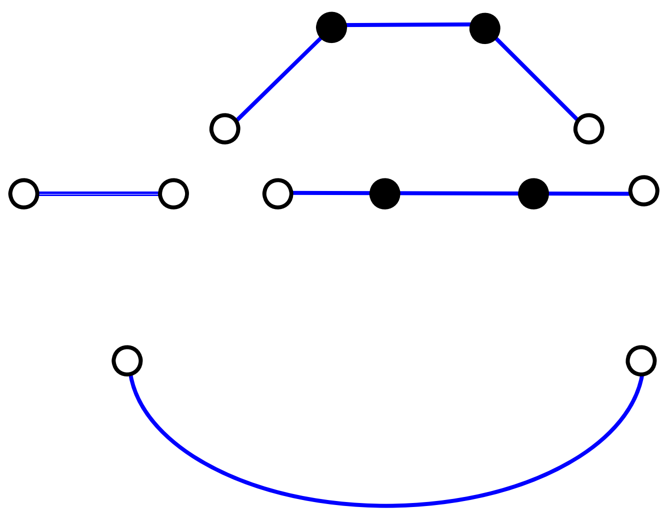
series-connection

Series-Parallel Graphs

Example



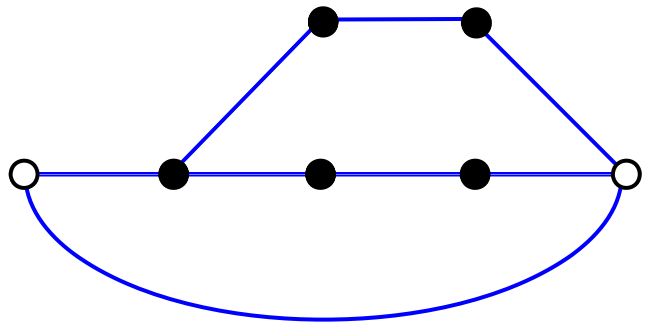
SP graph



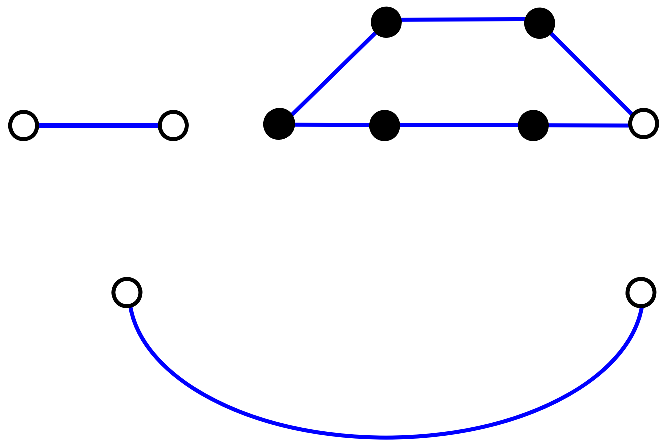
parallel-connection

Series-Parallel Graphs

Example



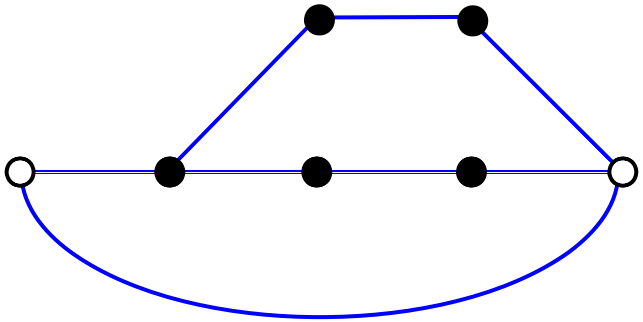
SP graph



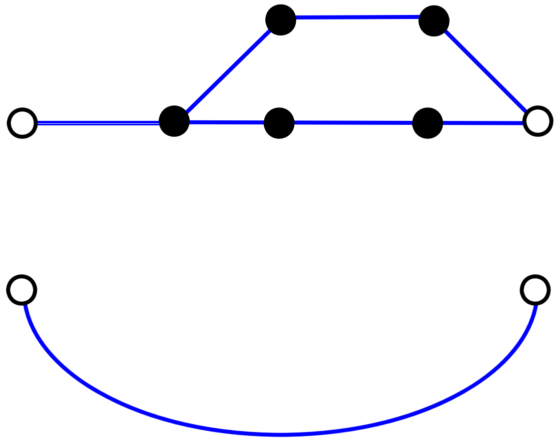
Series-connection

Series-Parallel Graphs

Example



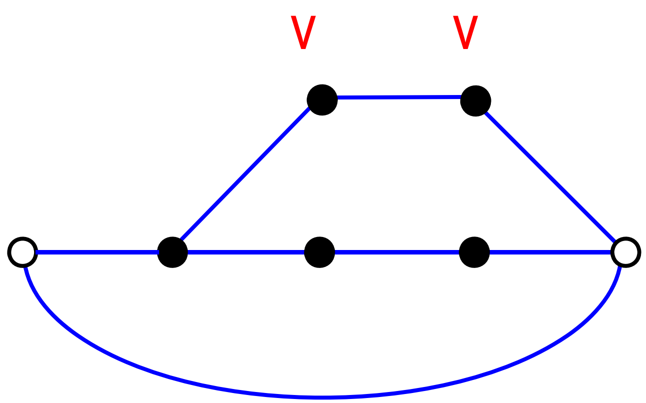
SP graph



Parallel-connection

Series-Parallel Graphs

Example



connected SP graph

Biconnected graphs G:

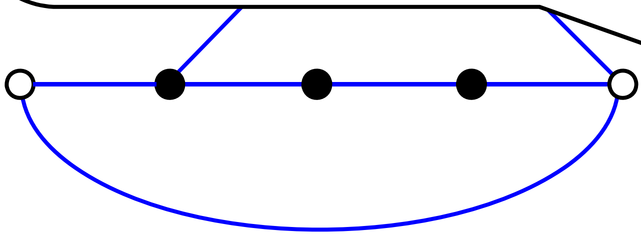
G – v is connected for each vertex v.

Biconnected SP graphs

S

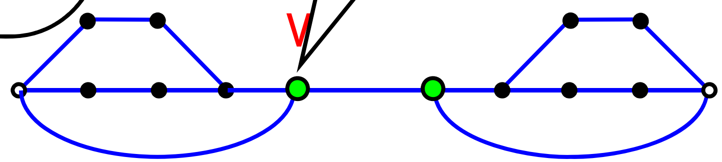
Biconnected graphs G:

$G - v$ is connected for each vertex v.



biconnected SP graph

cut vertex

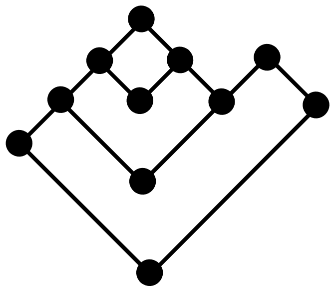


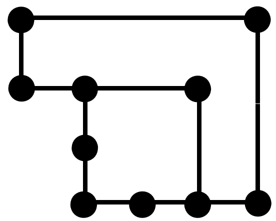
not connected

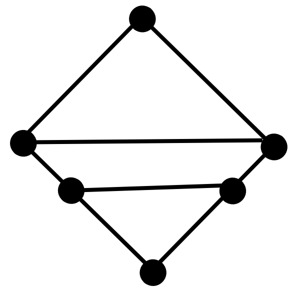
Biconnected SP graphs

Series-Parallel Graphs

Example







biconnected SP graphs

Series-Parallel Graphs

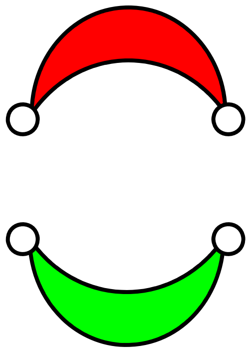
A SP graph is recursively defined as follows:

(a)

a single edge is a SP graph.

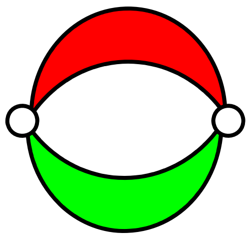
(b)

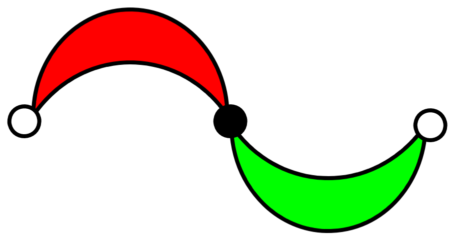
if



are SP graphs,

then





are SP graphs

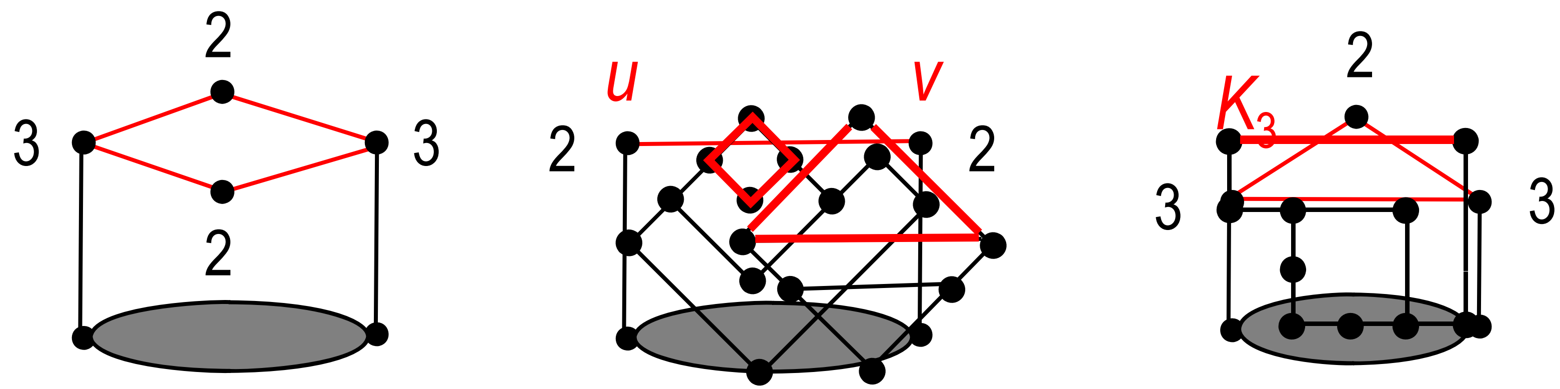
Lemma 1 (Our Main Idea)

Every biconnected SP graph G of $\Delta \leqq 3$ has one of the following three substructures:

(a) a diamond C

(b) two adjacent vertices u and v s.t. $d(u)=d(v)=2$

(c) a triangle K_3.



(a) (b) (c)

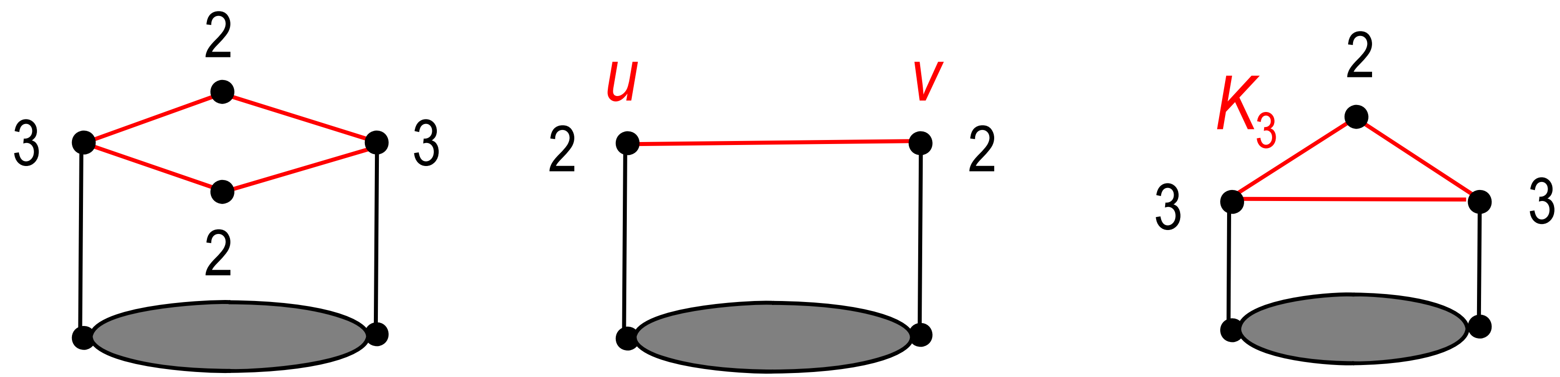
Lemma 1 (Our Main Idea)

Every biconnected SP graph G of $\Delta \leqq 3$ has one of the following three substructures:

(a) a diamond C

(b) two adjacent vertices u and v s.t. $d(u)=d(v)=2$

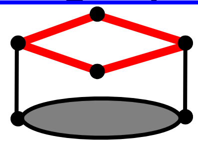
(c) a triangle K_3.



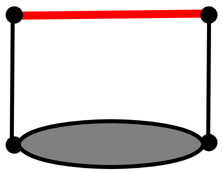
(a) (b) (c)

Algorithm(G)

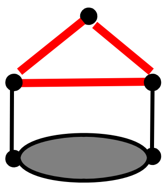
Let G be a biconnected SP graph of $\Delta \leqq 3$.

Case (a): ∃ a diamond  ,

Recursively find an optimal drawing.

Case (b): ∃ 

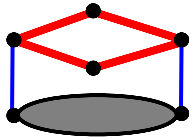
Decompose to smaller subgraphs in series or parallel, and iteratively find an optimal drawing

Case (c): ∃ 

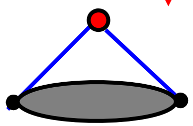
Similar as Case (b).

Algorithm(G)

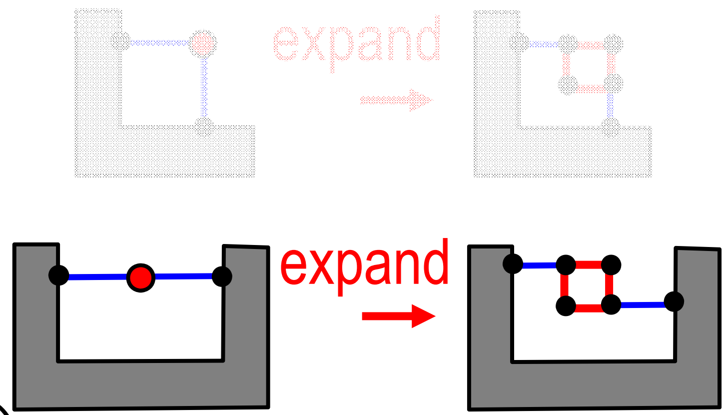
Let G be a biconnected SP graph of $\Delta \leqq 3$.

Case (a): $\exists$ a diamond  ,

contract

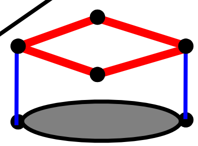
Algorithm(), find

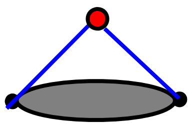
Return



optimal

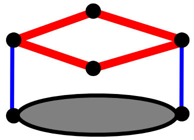
optimal

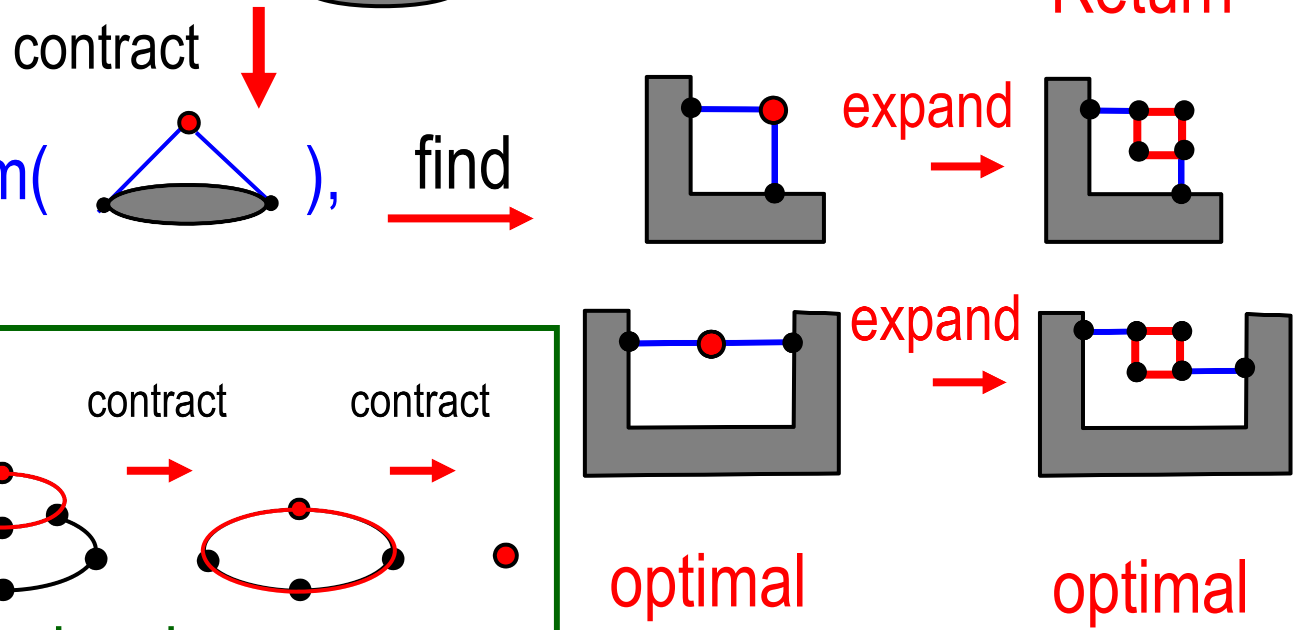


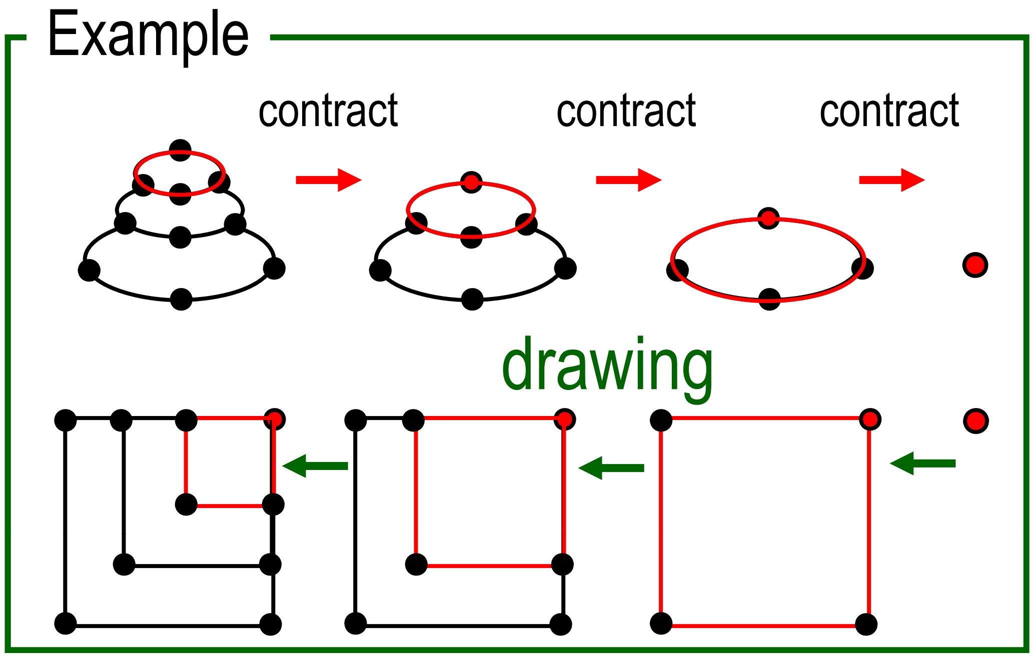
Recur G to a smaller one 

Algorithm(G)

Let G be a biconnected SP graph of $\Delta \leqq 3$.

Case (a): $\exists$ a diamond  ,

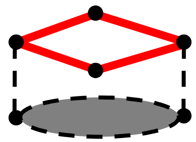




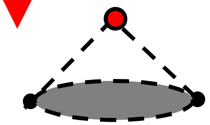
Algorithm(G)

Let G be a biconnected SP graph of $\Delta \leqq 3$.

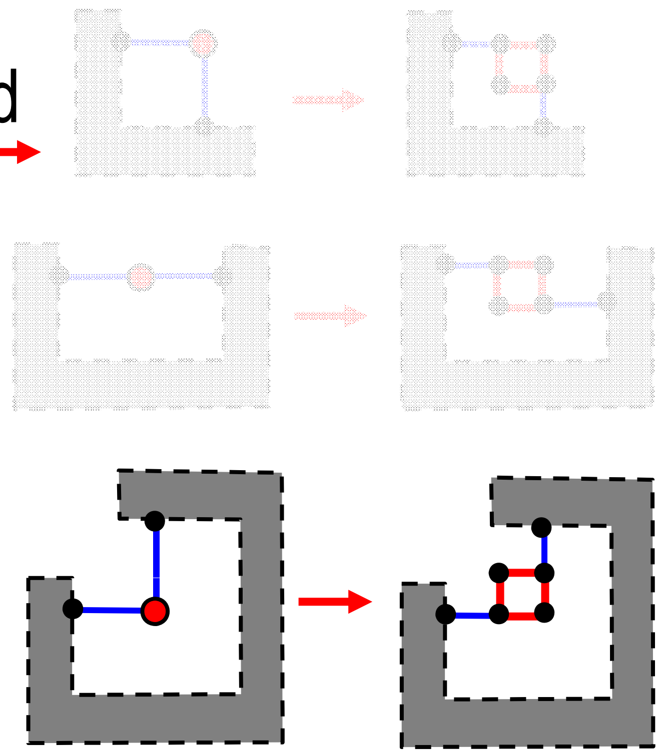
If $n(G) < 6$, Then find an optimal drawing of G

Else If $\exists$ a diamond  ,

contract

Then Algorithm(), find

Return



Recur G to a smaller one

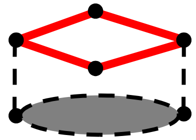
optimal

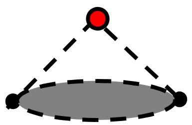
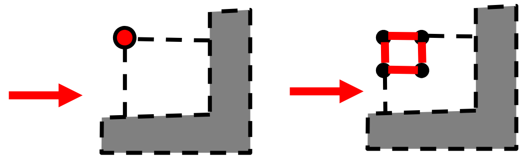
optimal

Algorithm(G)

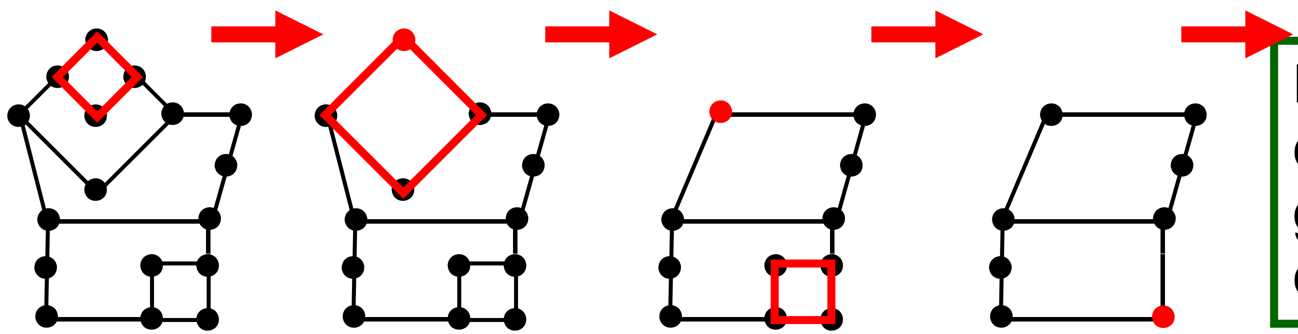
Let G be a biconnected SP graph of $\Delta \leqq 3$.

If $n(G) < 6$, Then find an optimal drawing of G

Else If $\exists$ a diamond  ,

Then Algorithm(), 

Example

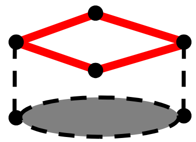


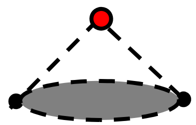
Find an optimal drawing of SP graphs without diamonds

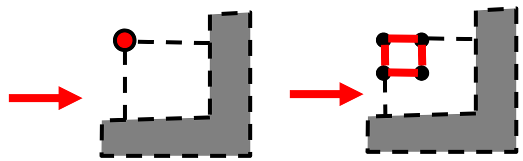
Algorithm(G)

Let G be a biconnected SP graph of $\Delta \leqq 3$.

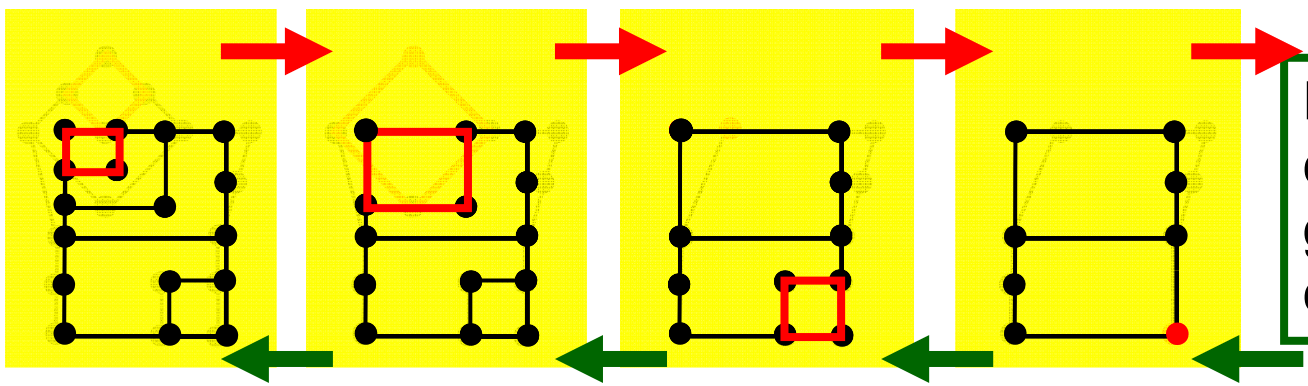
If $n(G) < 6$, **Then** find an optimal drawing of G

Else If $\exists$ a diamond  ,

Then Algorithm(),



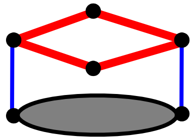
Example



Find an optimal drawing of SP graphs without diamonds

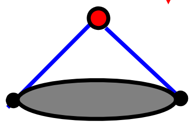
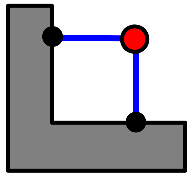
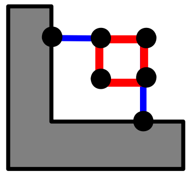
Algorithm(G)

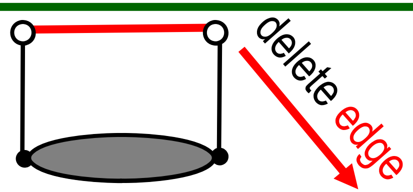
Let G be a biconnected SP graph of $\Delta \leqq 3$.

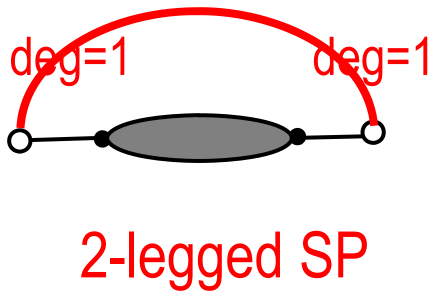
Case (a): $\exists$ a diamond  ,

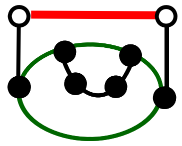
contract

Return

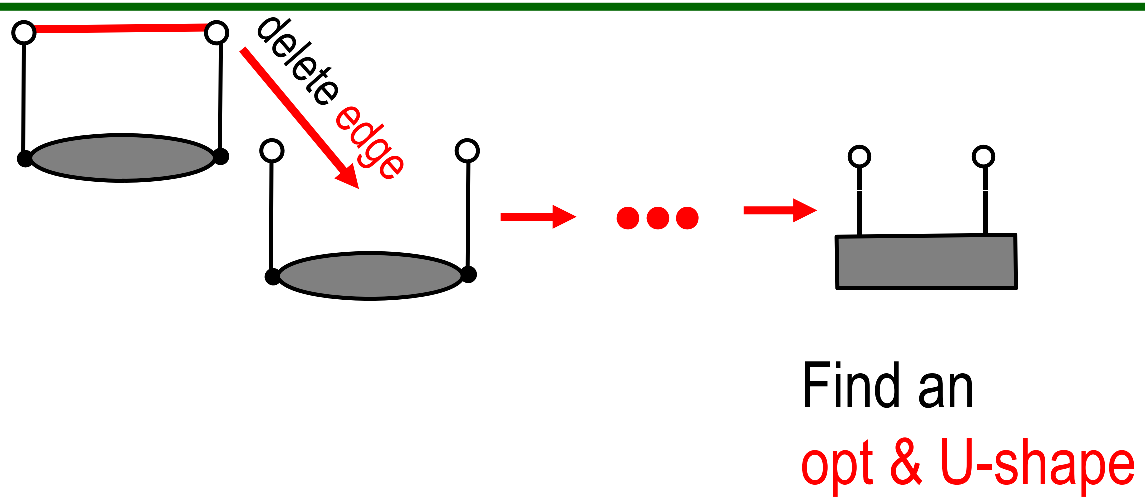
Algorithm(), find  expand 

Case (b): $\exists$ 

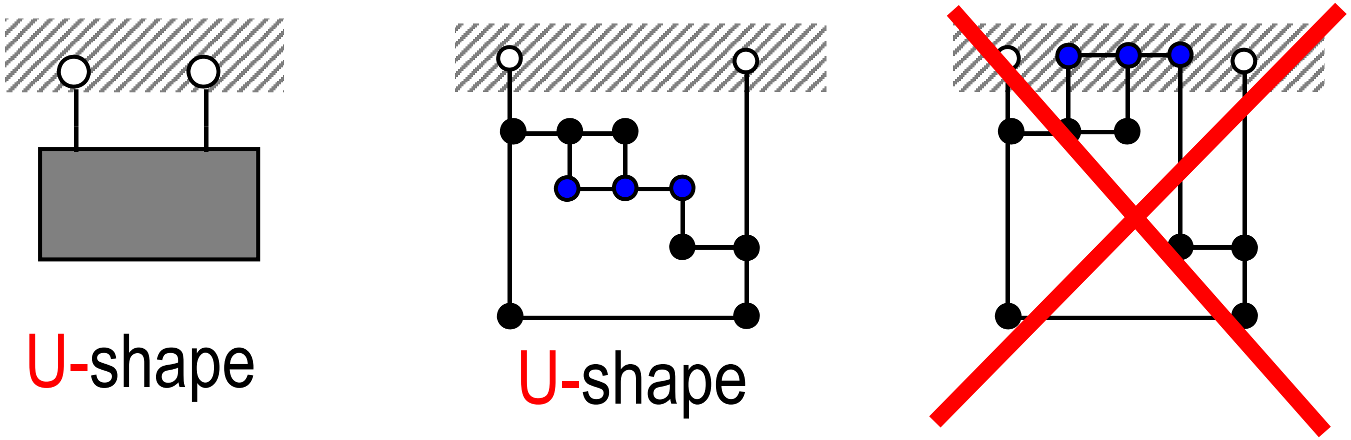




Case (b): ∃

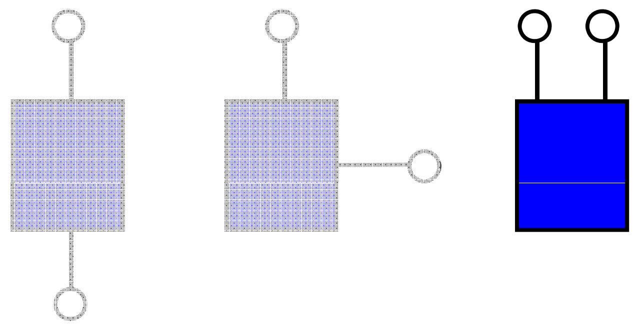


Our Main Idea



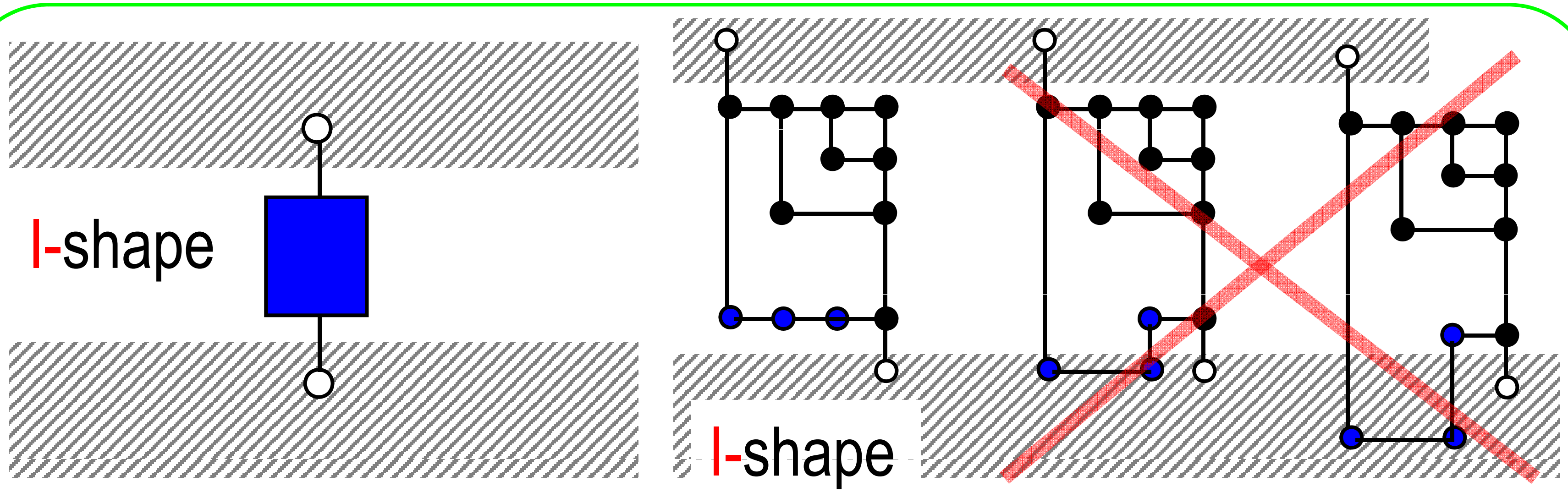
Defin , L- and U-shaped drawings

- terminals are drawn on the outer face;
- the drawing except terminals doesn't intersect the north side



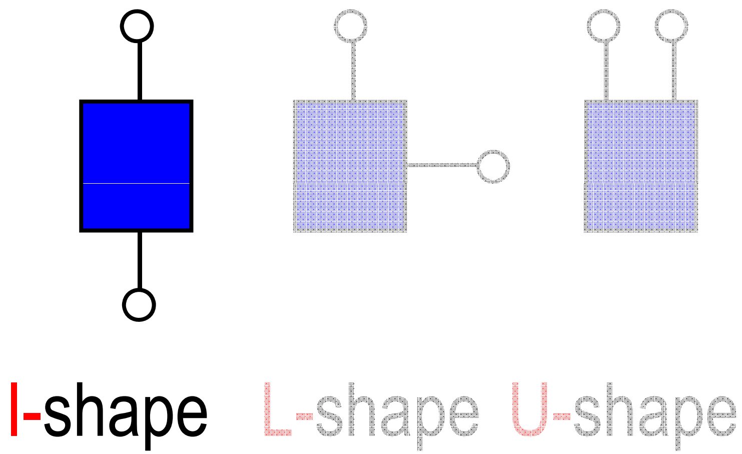
I-shape L-shape U-shape

Our Main Idea

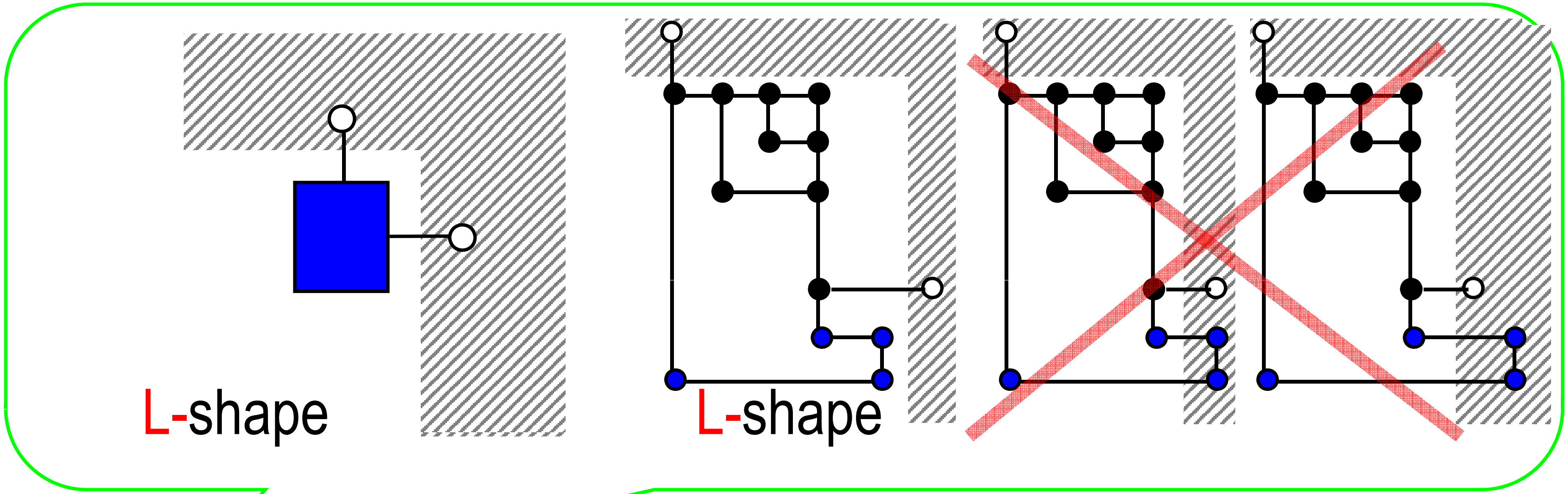


Def… , L- and U-shaped drawings

- terminals are drawn on the outer face;
- the drawing except terminals intersects neither the north side nor the south side

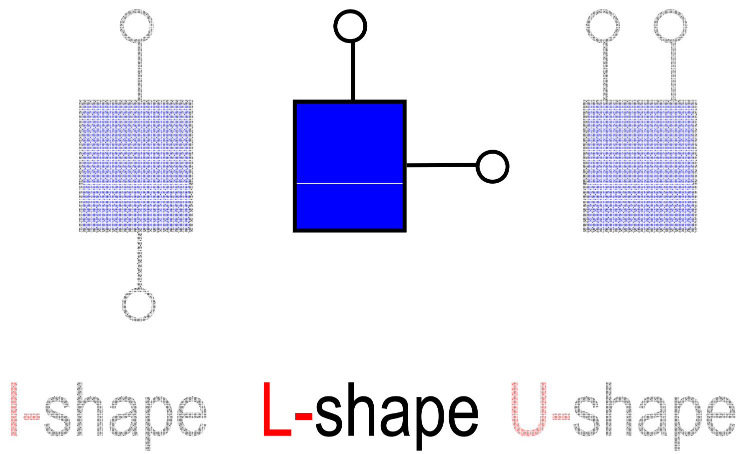


Our Main Idea



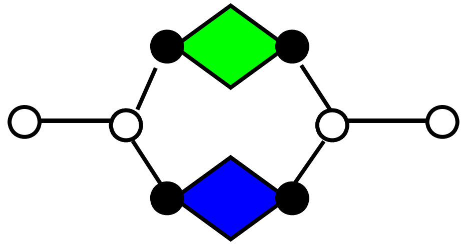
Defi , L- and U-shaped drawings

- terminals are drawn on the outer face;
- the drawing except terminals intersects neither the north side nor the east side

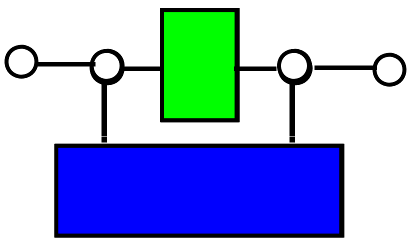


Lemma 2

Every 2-legged SP graph without diamond has optimal I-, L- and U-shaped drawings

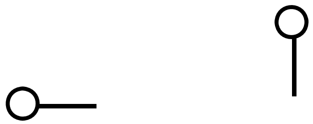


optimal



I-shape

optimal



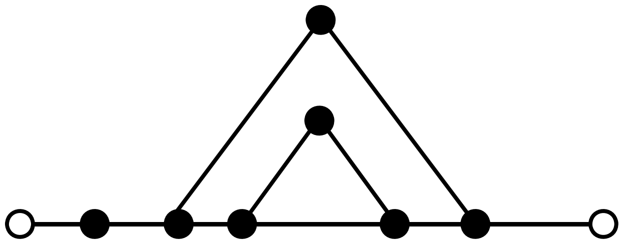
L-shape

optimal

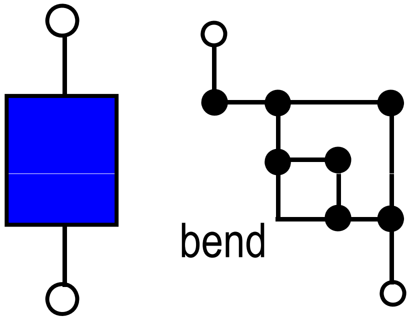
U-shape

Lemma 2

Every 2-legged SP graph without diamond has optimal I-, L- and U-shaped drawings

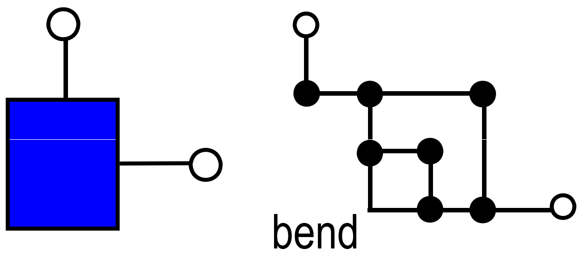


optimal



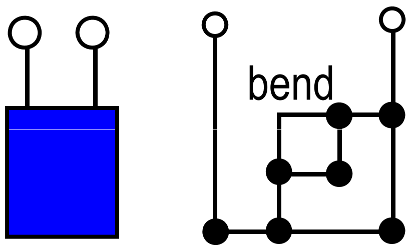
I-shape

optimal



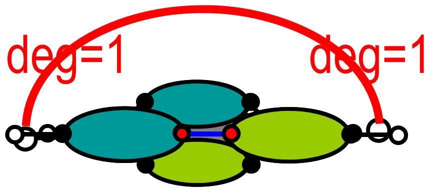
L-shape

optimal



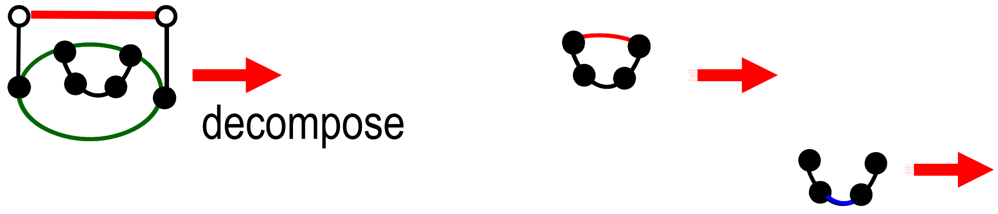
U-shape

series connection

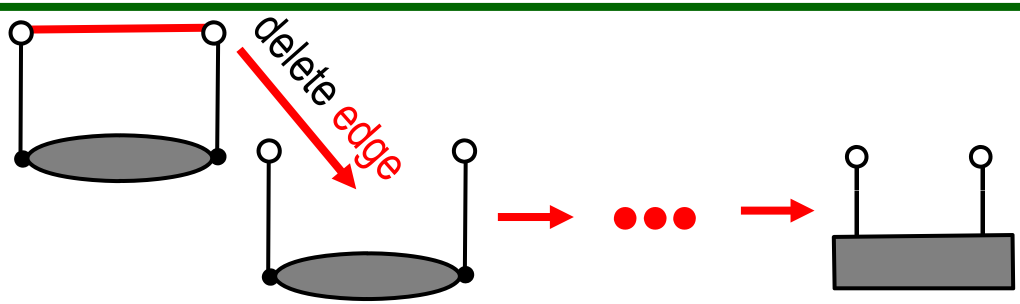


2-legged SP

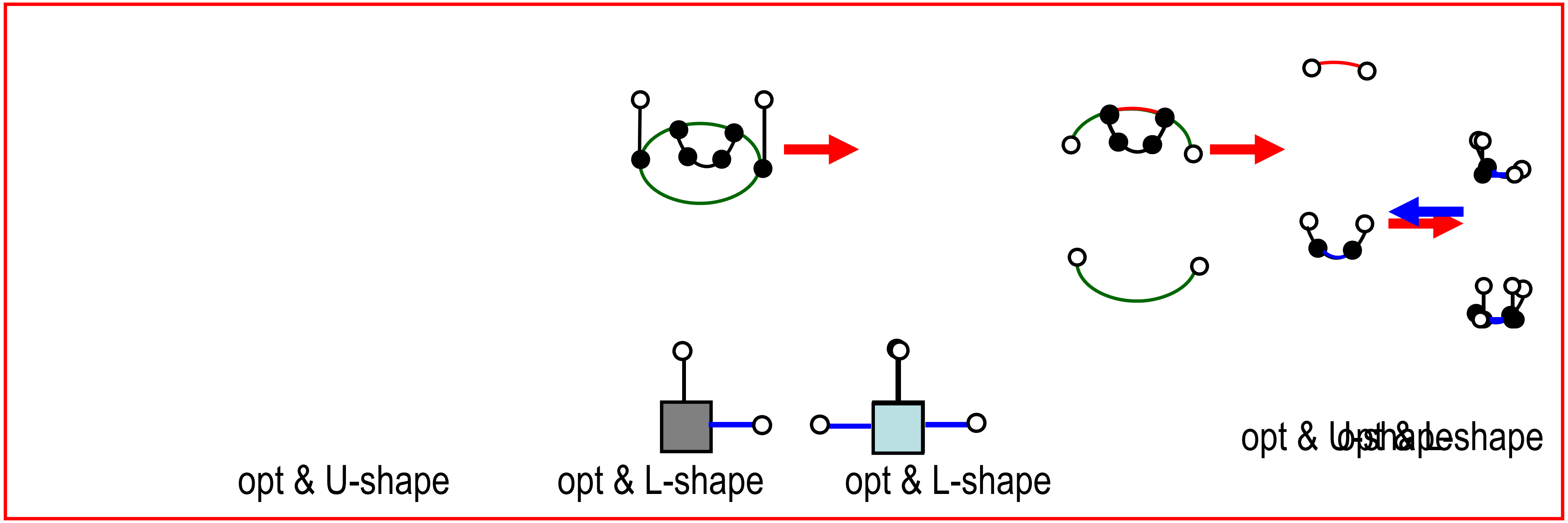
$\Delta \leqq 3$



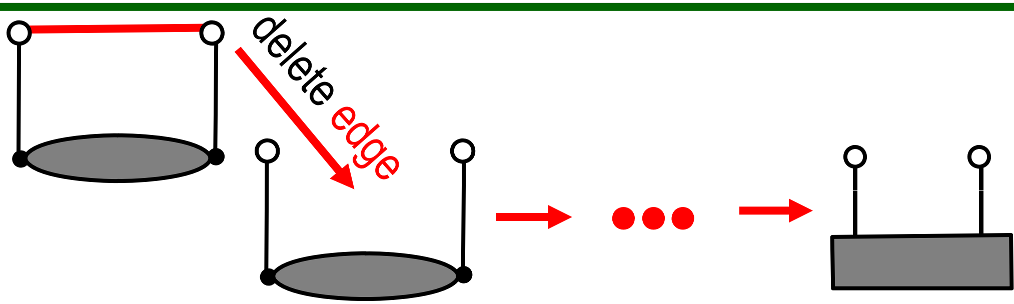
Case (b): ∃



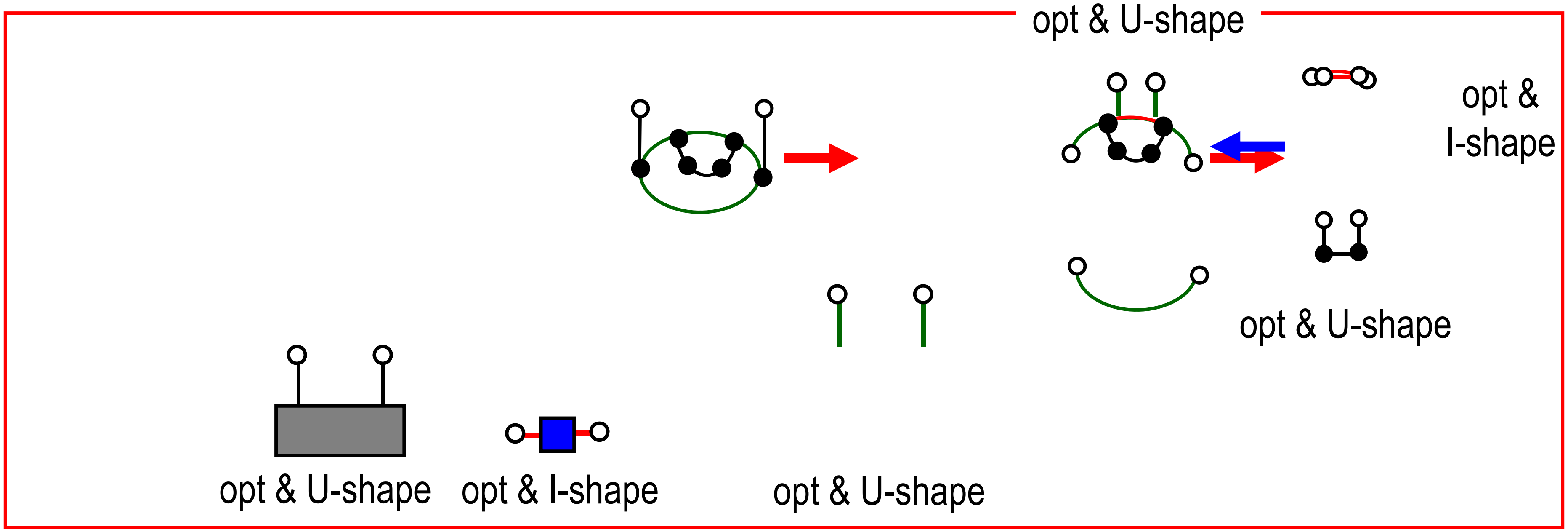
Find an
opt & U-shape

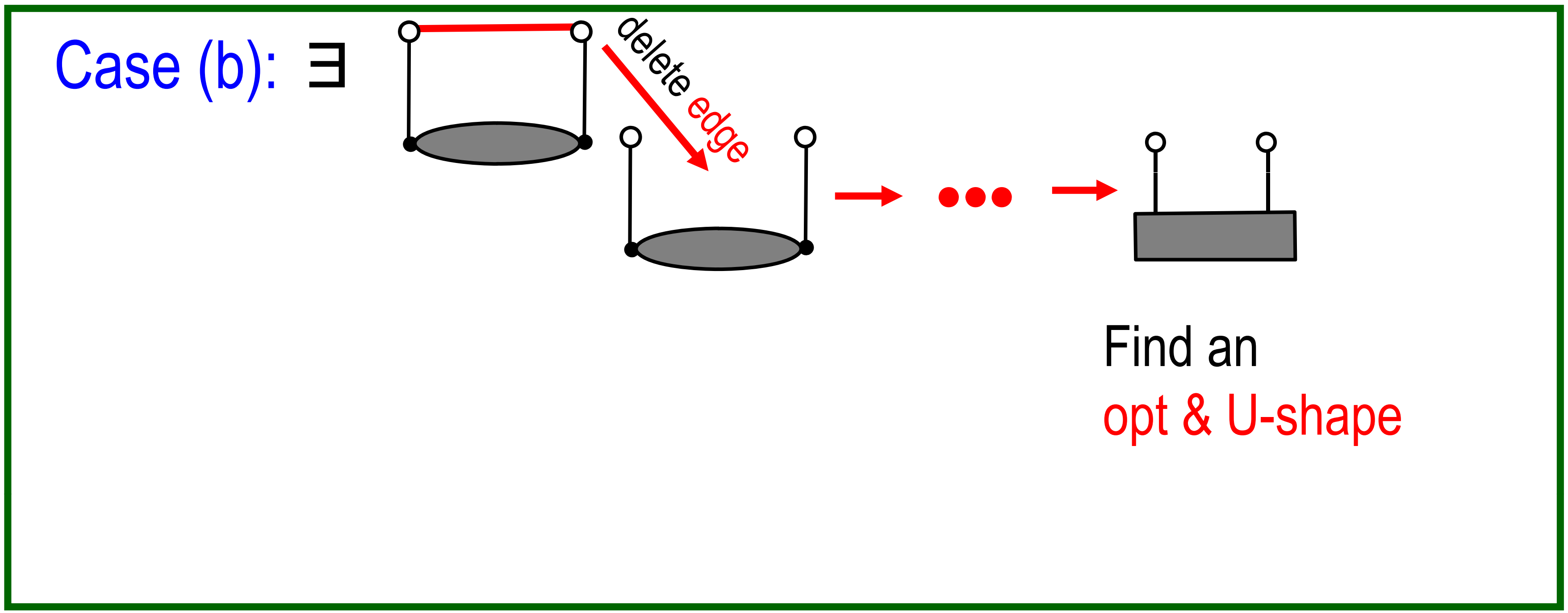


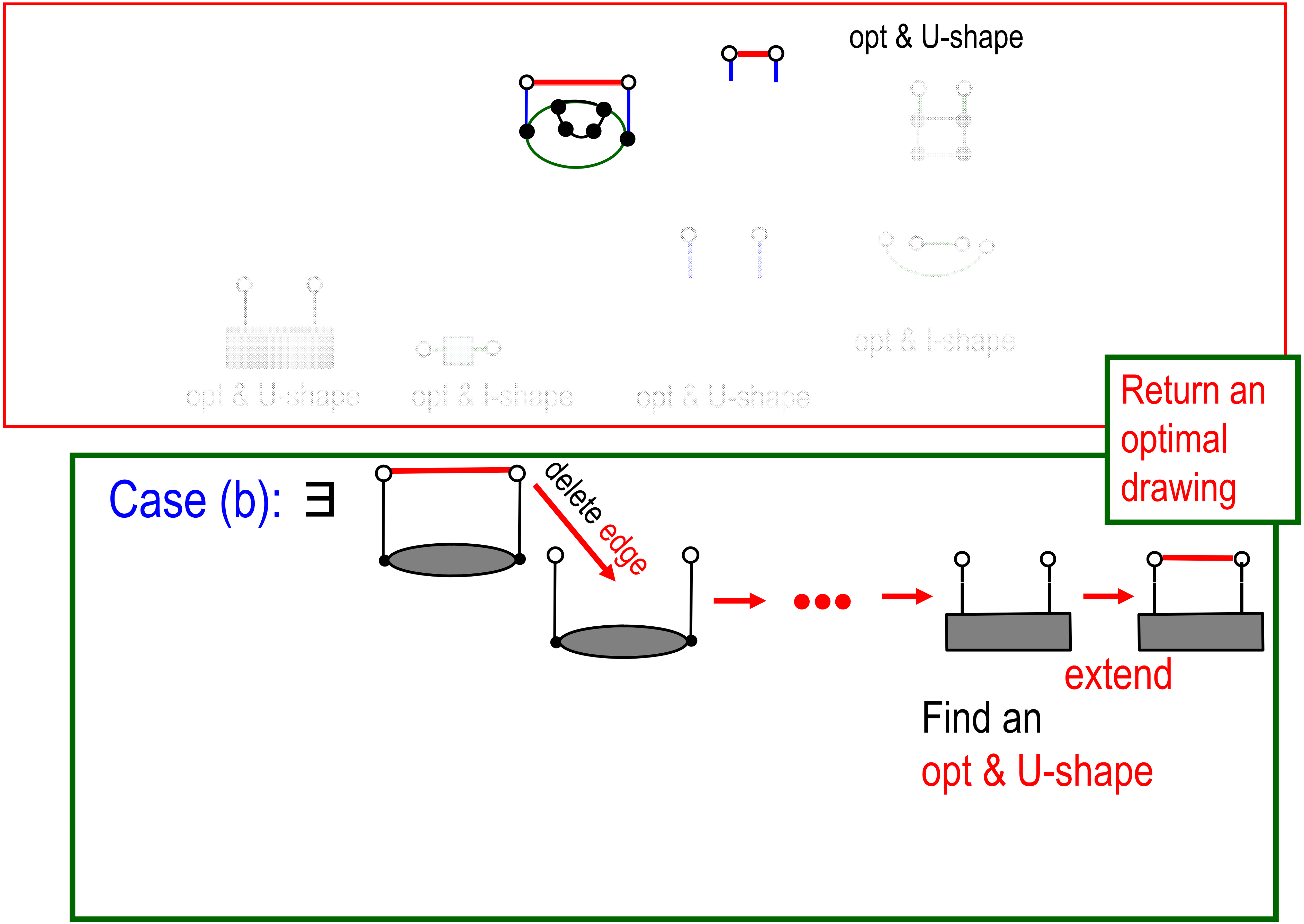
Case (b): ∃



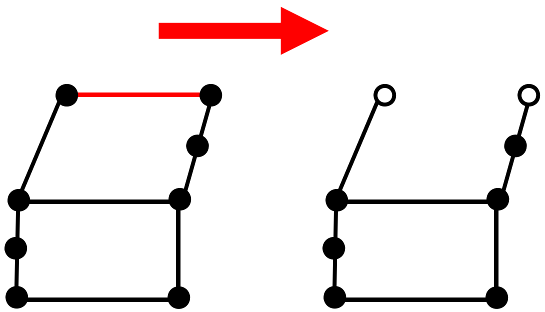
Find an
opt & U-shape


opt & U-shape
opt &
I-shape
opt & U-shape
opt & U-shape
opt & I-shape
opt & U-shape

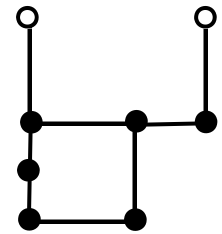

Case (b): ∃
delete edge
Find an
opt & U-shape


opt & U-shape
opt & I-shape
opt & U-shape
opt & I-shape
opt & U-shape
Return an optimal drawing
Case (b): ∃
delete edge
extend
Find an
opt & U-shape

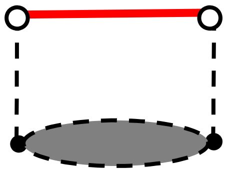
Example



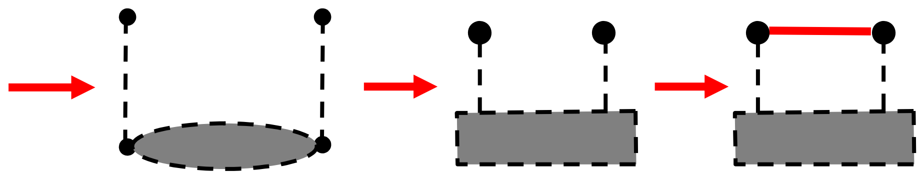
How to find an optimal U-shaped drawing ?



Else If ∃

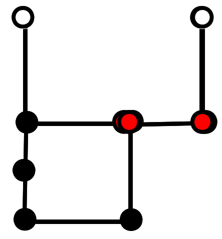


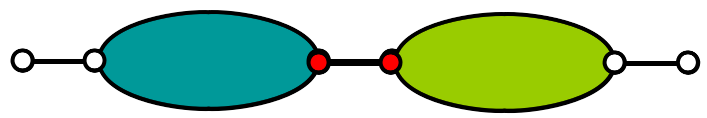
Then

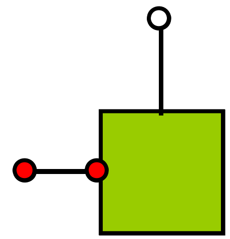


series connection

Example



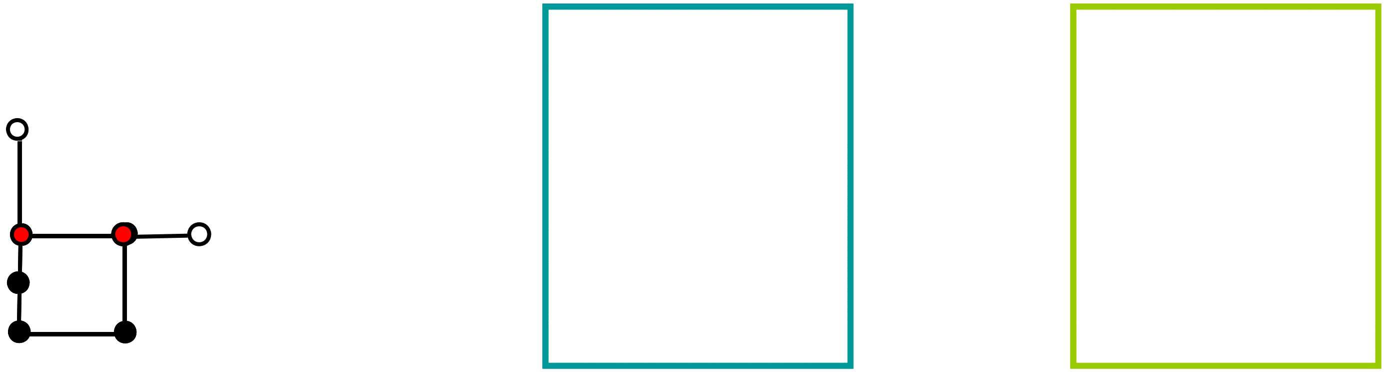


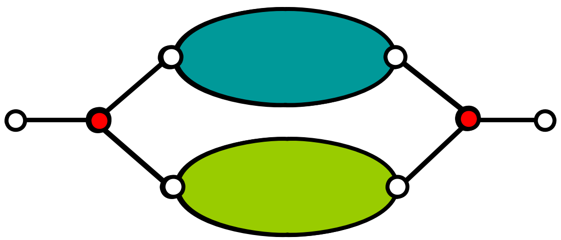


L-shaped draw

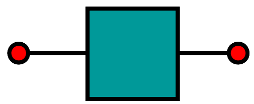
parallel connection

Example

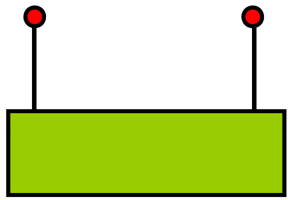




I-shape



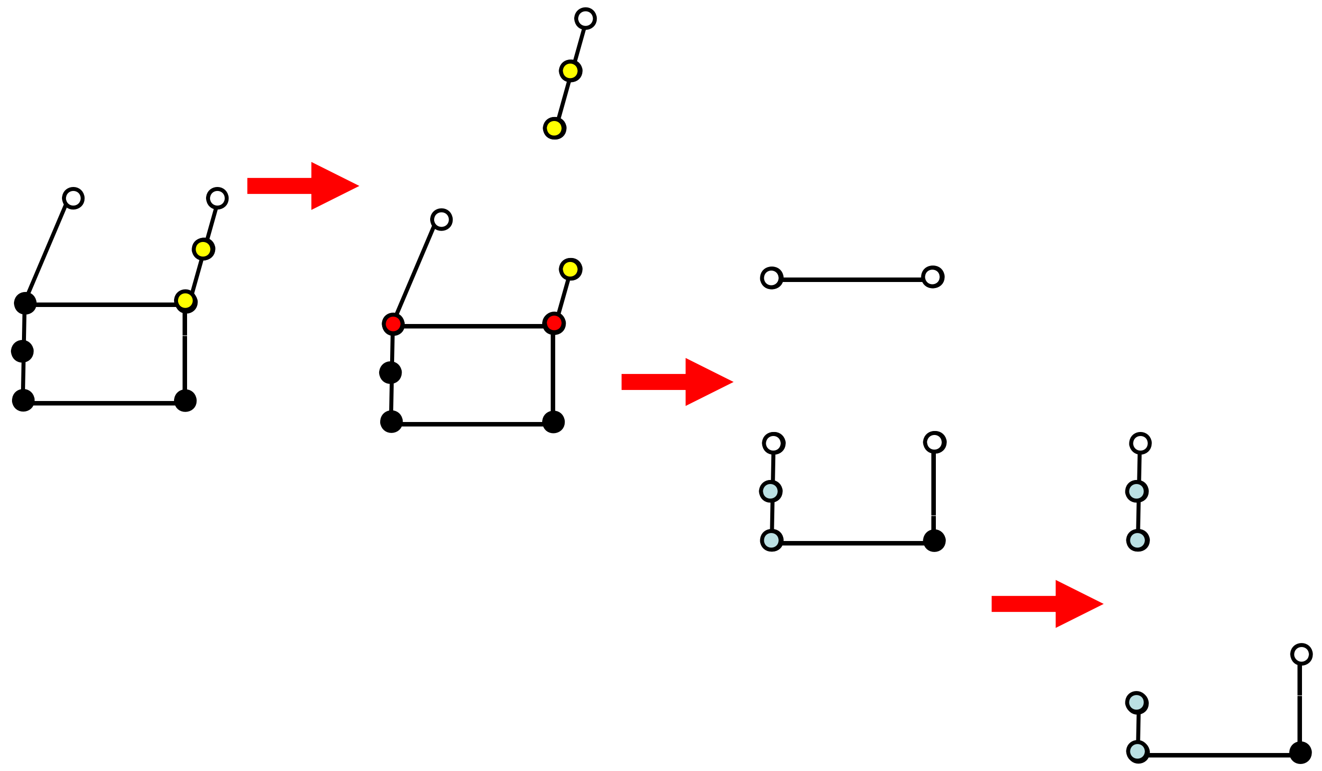
U-shape

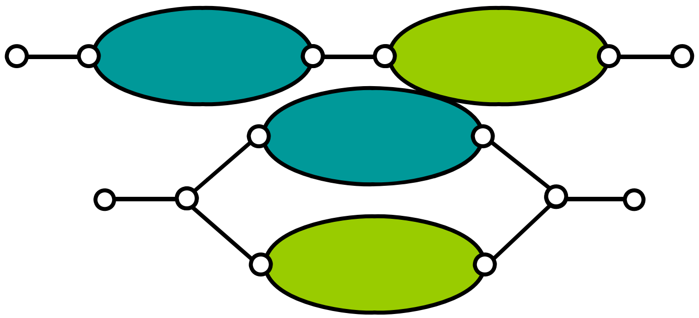


recursively

I-shape
U-shape

Example

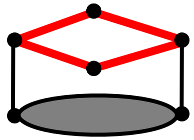




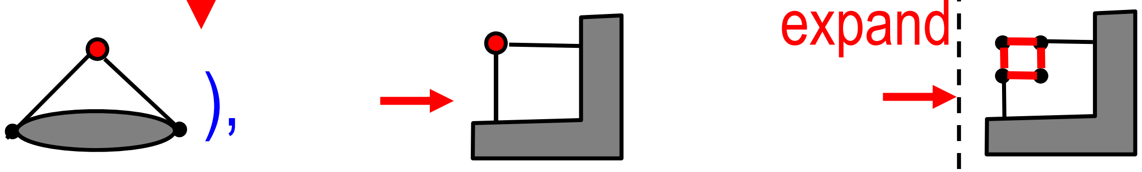
Algorithm(G)

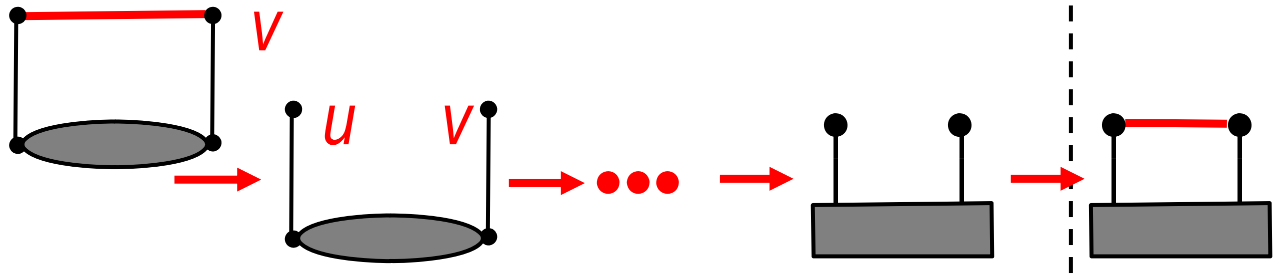
Let G be a biconnected SP graph of $\Delta \leqq 3$.

Return optimal drawing

Case (a): ∃ a diamond ,

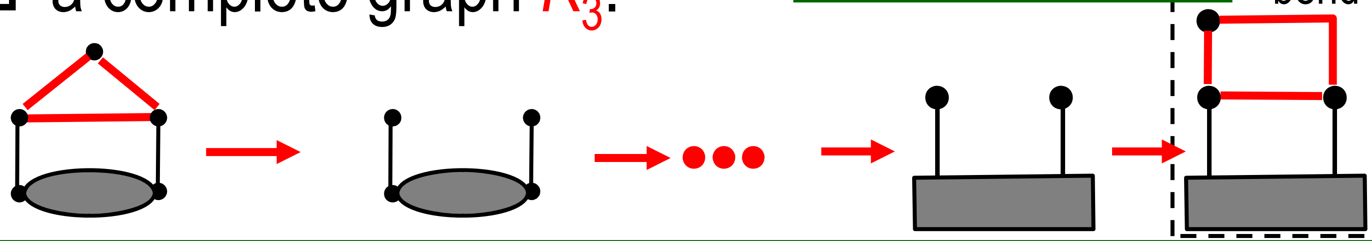
contraction

Algorithm(

Case (b): ∃ 

opt & U-shape

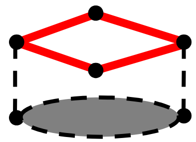
Case (c): ∃ a complete graph K_3.

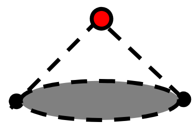
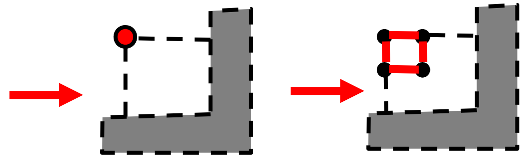


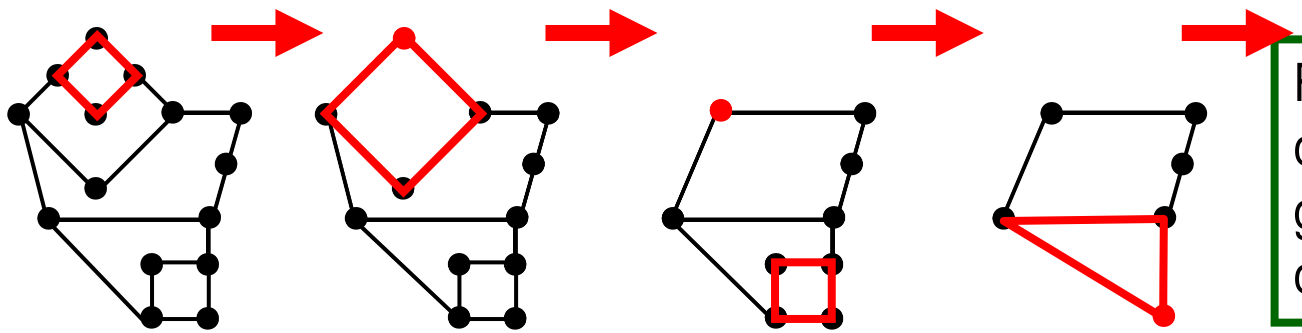
Algorithm(G)

Let G be a biconnected SP graph of $\Delta \leqq 3$.

If $n(G) < 6$, Then find an optimal drawing of G

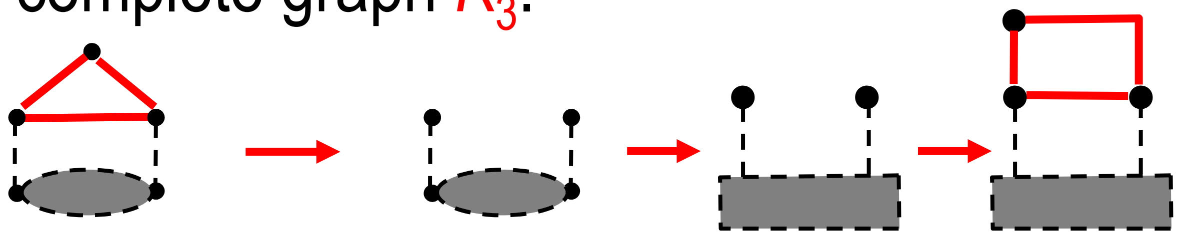
Else If $\exists$ a diamond  ,

Then Algorithm(), 



Find an optimal drawing of SP graphs without diamonds

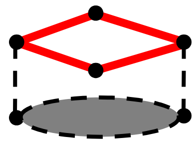
Else $\exists$ a complete graph K_3.

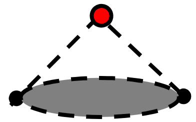
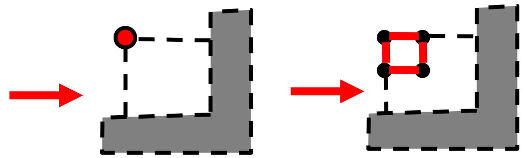


Algorithm(G)

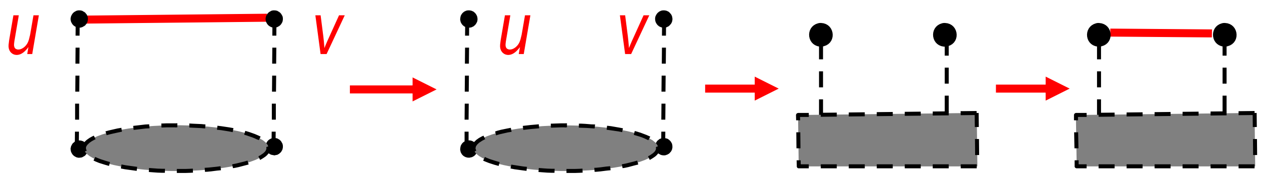
Let G be a biconnected SP graph of $\Delta \leqq 3$.

If $n(G) < 6$, Then find an optimal drawing of G

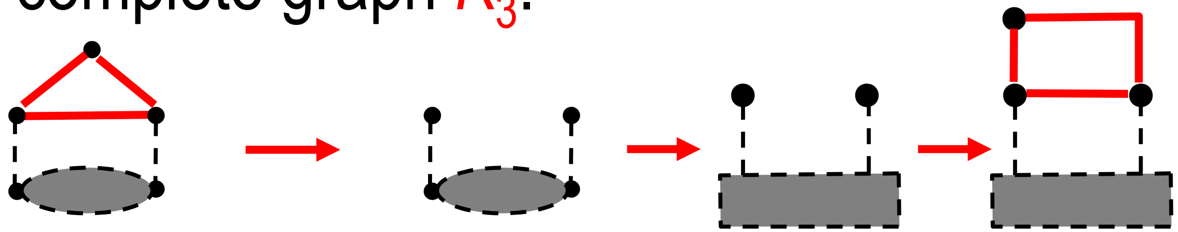
Else If $\exists$ a diamond  ,

Then Algorithm(), 

Else If $\exists$ two adjacent vertices u,v s.t. $d(u)=d(v)=2$

Then 

Else $\exists$ a complete graph K_3.

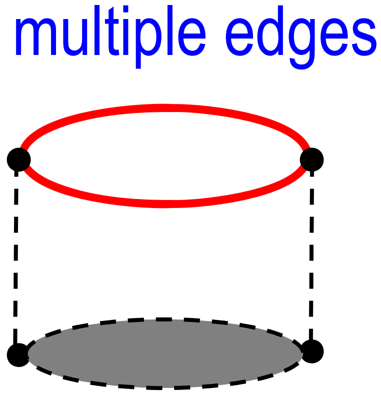


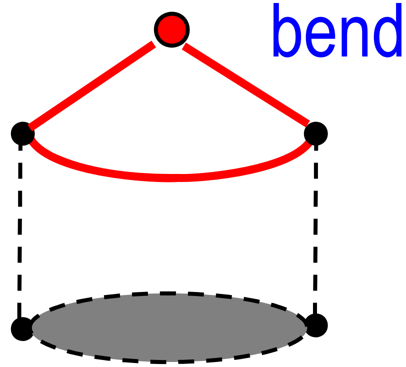
Conclusions

Theorem 1

An optimal orthogonal drawing of a biconnected SP graph G of $\Delta \leqq 3$ can be found in linear time.

Our algorithm works well even if G has multiple edges





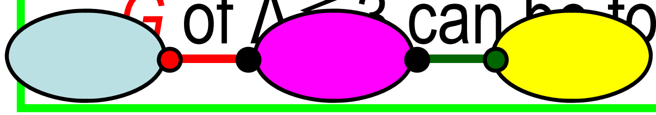
Theorem 1

An optimal orthogonal drawing of a biconnected SP graph G of $\Delta \leqq 3$ can be found in linear time.

Theorem 1

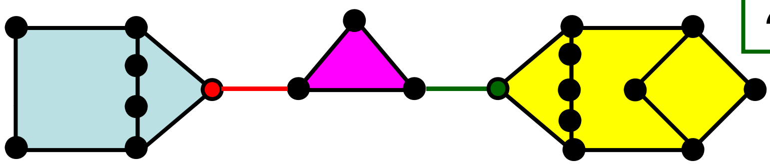
Our algorithm works well even if G is not biconnected.

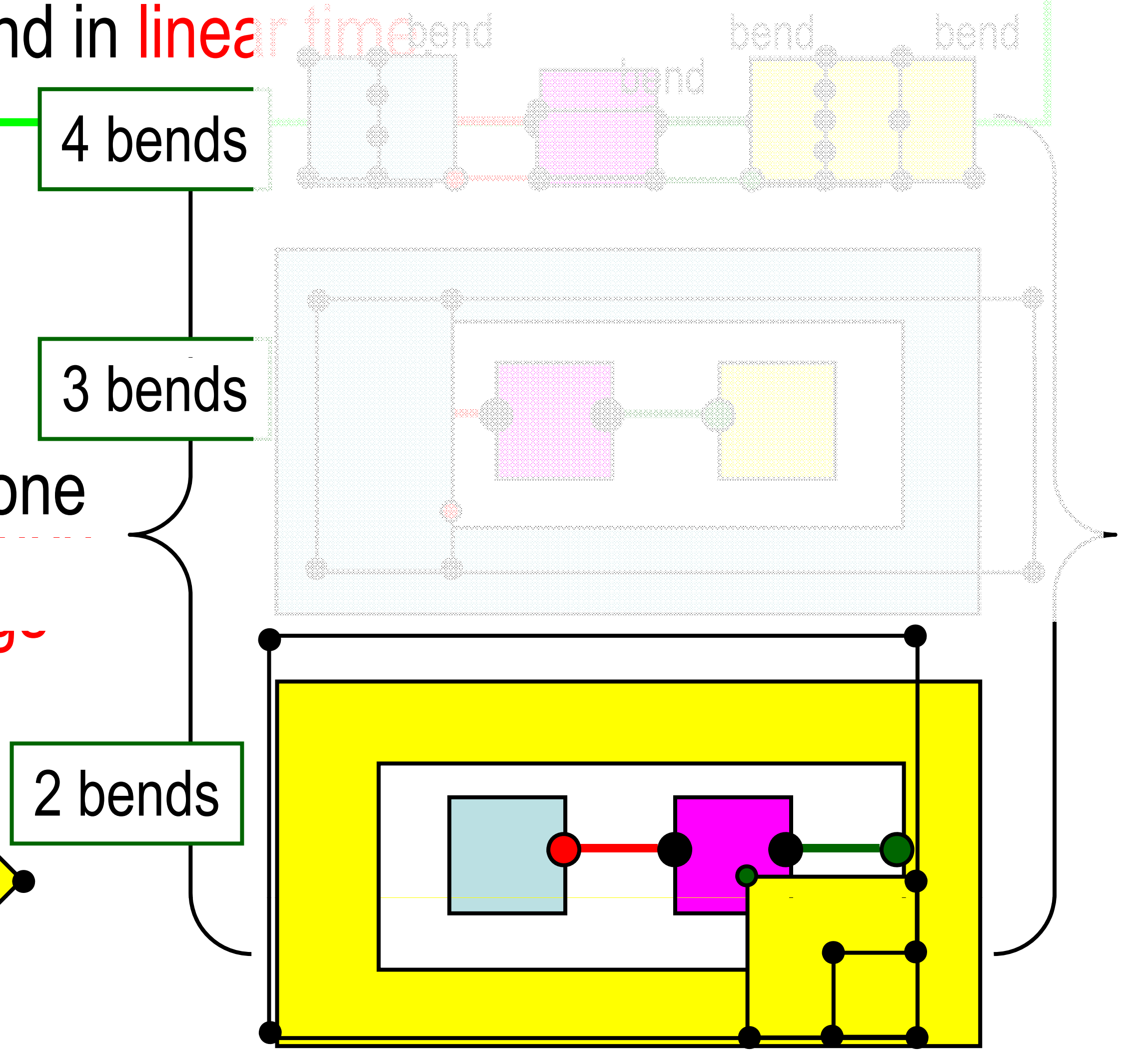
An optimal orthogonal drawing of a biconnected SP graph G of $\Delta \leq 3$ can be found in linear time.



Optimal drawing

find the best one of drawings





Conclusions

Theorem 1

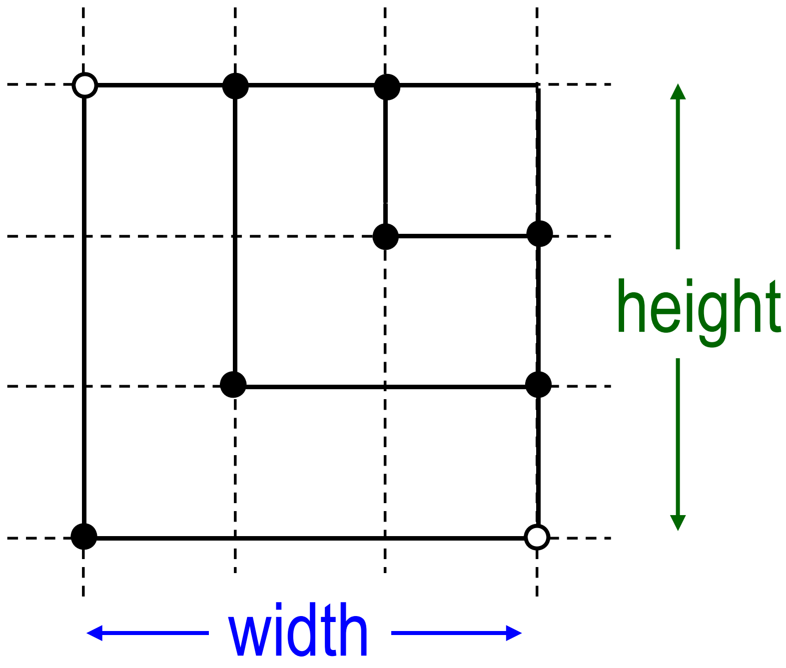
An optimal orthogonal drawing of a SP graph G of $\Delta \leqq 3$ can be found in linear time.

Conclusions

bend(G) $\leqq \lceil n/3 \rceil$ for biconnected SP graphs G of $\Delta \leqq 3$

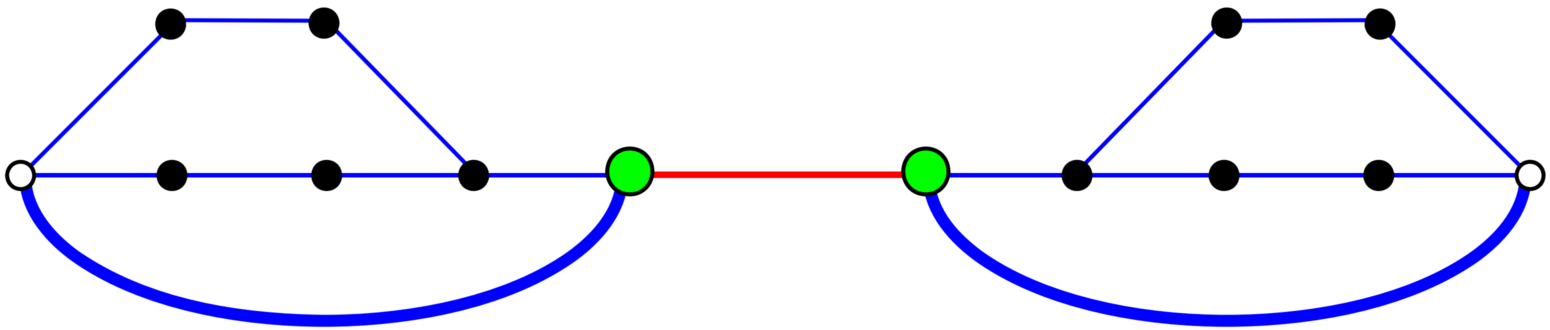
Grid size $\leqq 8n/9$

=width + height



Thank You

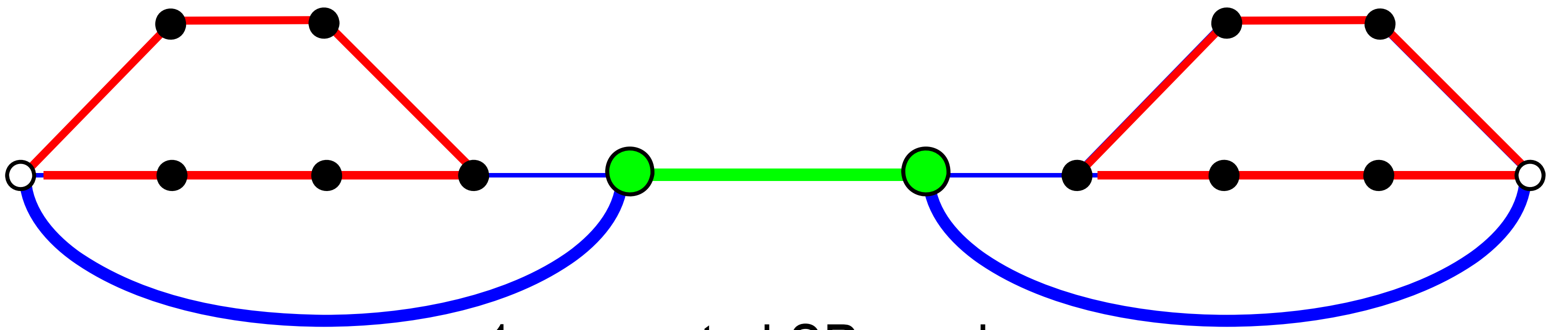
Optimal orthogonal drawing



planar graph

Optimal orthogonal drawings ?

Optimal orthogonal drawing



1-connected SP graph

∃ a one-bend orthogonal drawing ?

a 2-bend orthogonal drawing

one bend

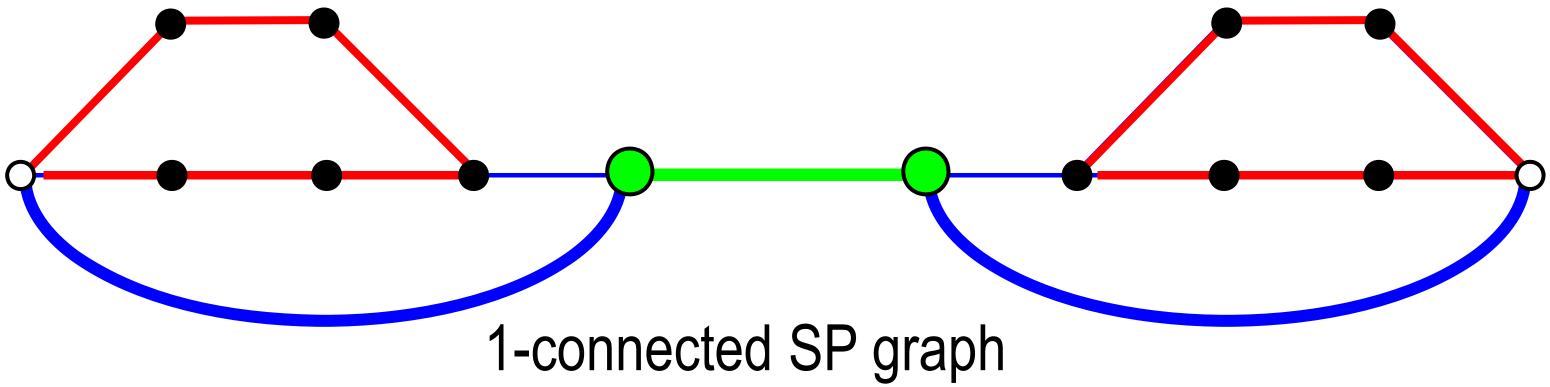
not optimal

Is this optimal ?

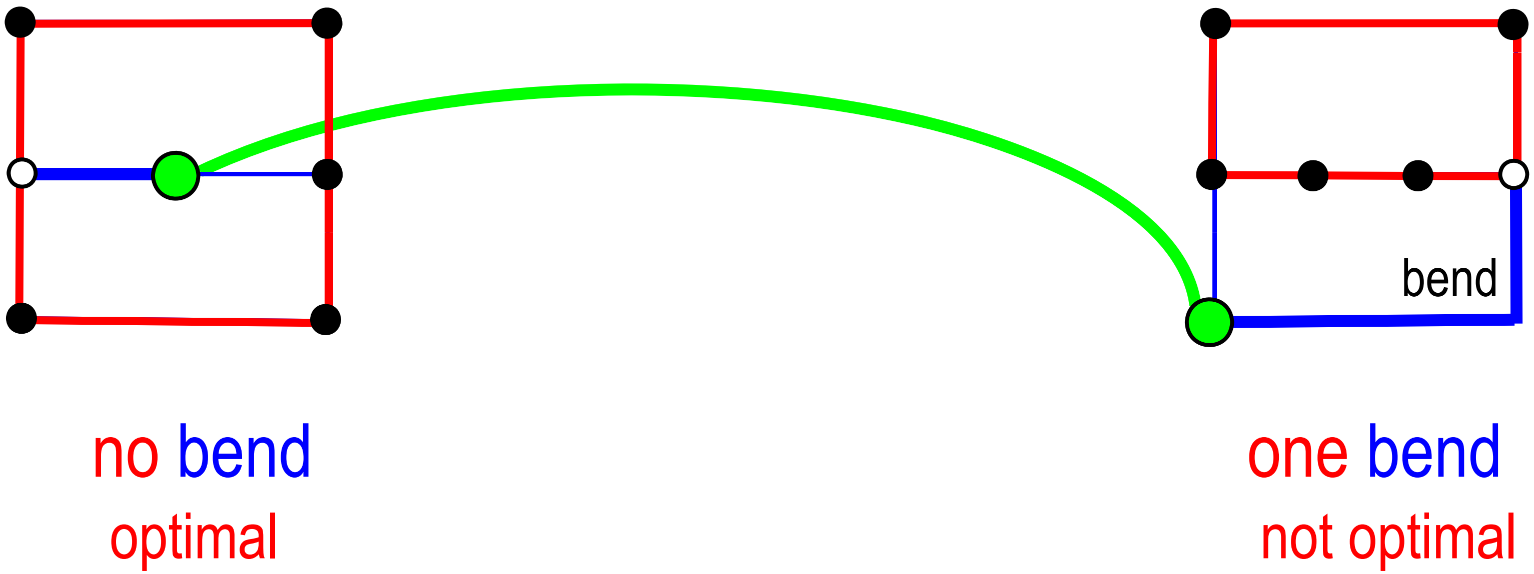
one bend

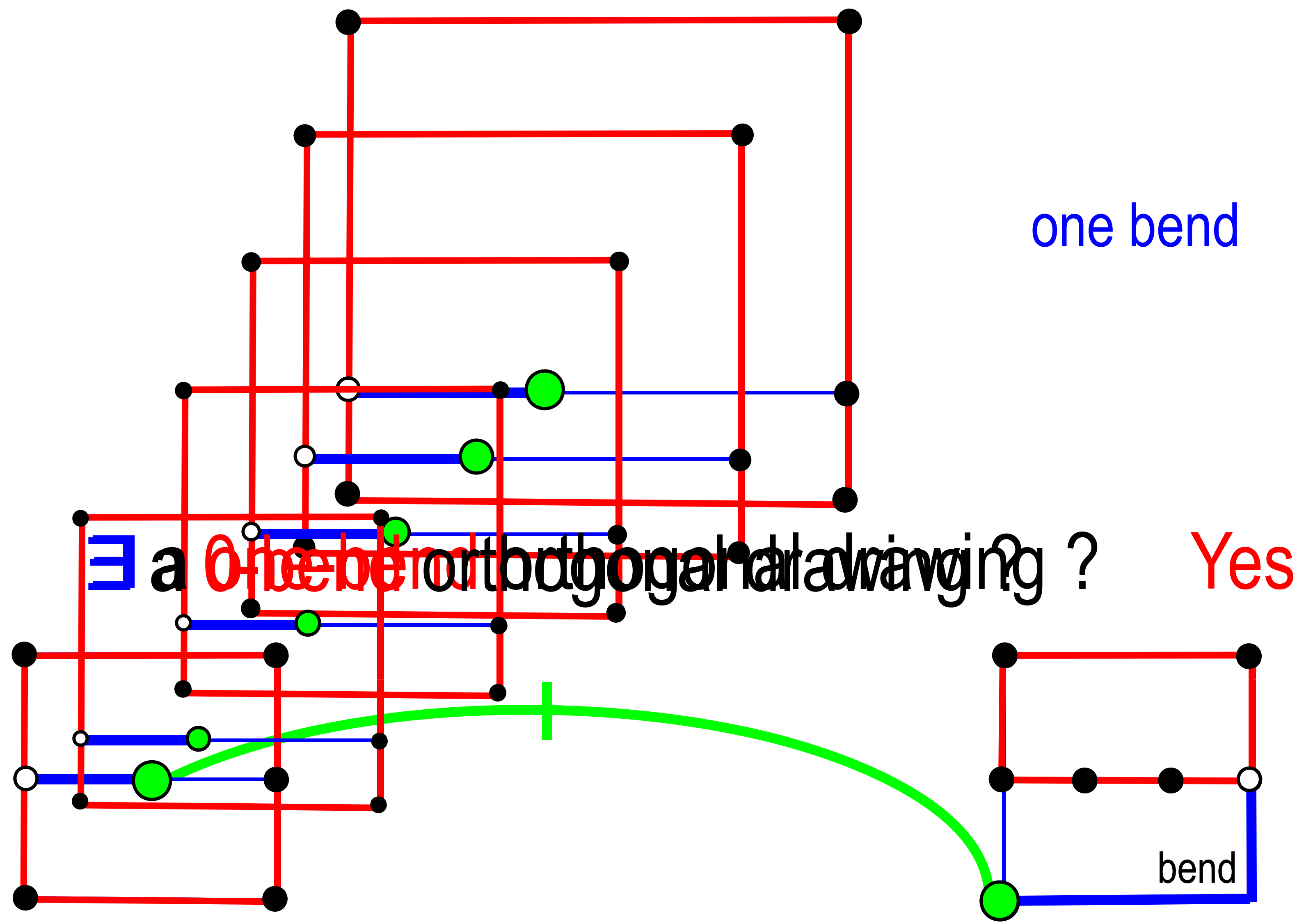
not optimal

Optimal orthogonal drawing



∃ a one-bend orthogonal drawing ?



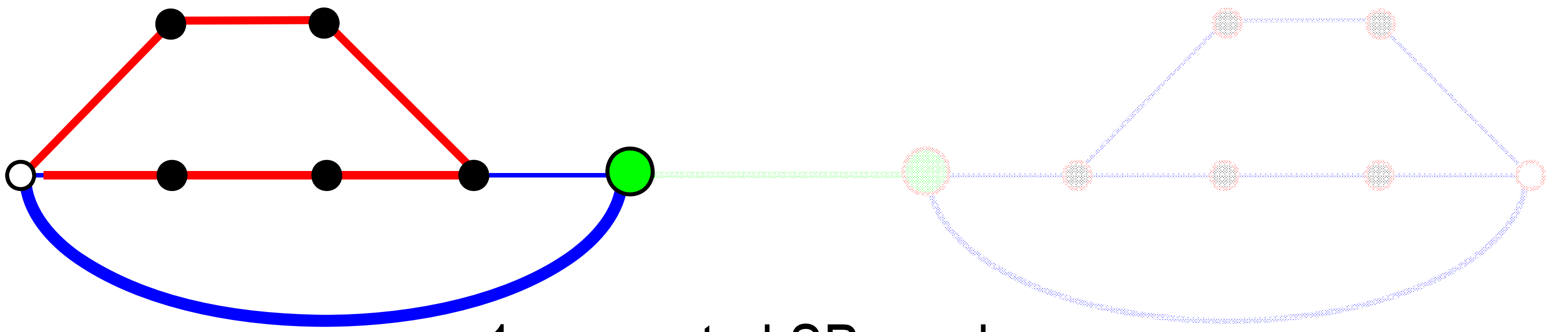


no bend
optimal

Is this optimal ?

one bend
not optimal

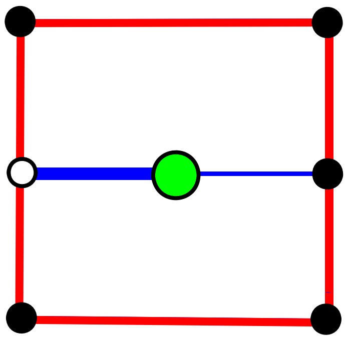
Optimal orthogonal drawing



1-connected SP graph

∃ a 0-bend orthogonal drawing ?

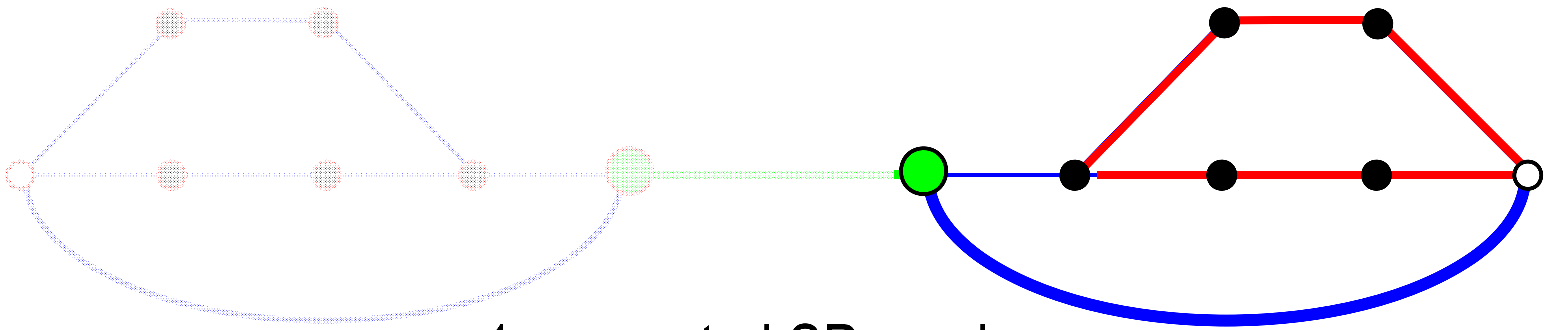
0 bend



optimal

orthogonal drawings

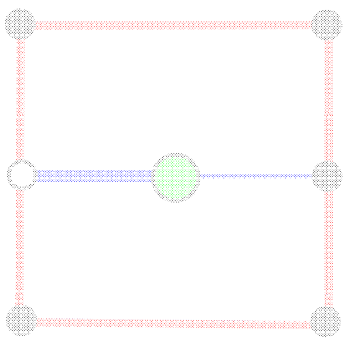
Optimal orthogonal drawing



1-connected SP graph

∃ a 0-bend orthogonal drawing ?

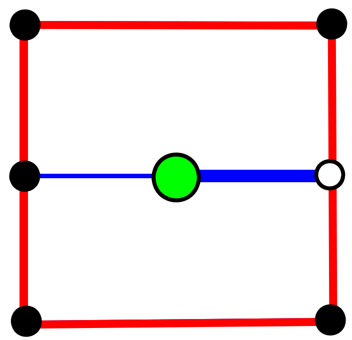
0 bend



optimal

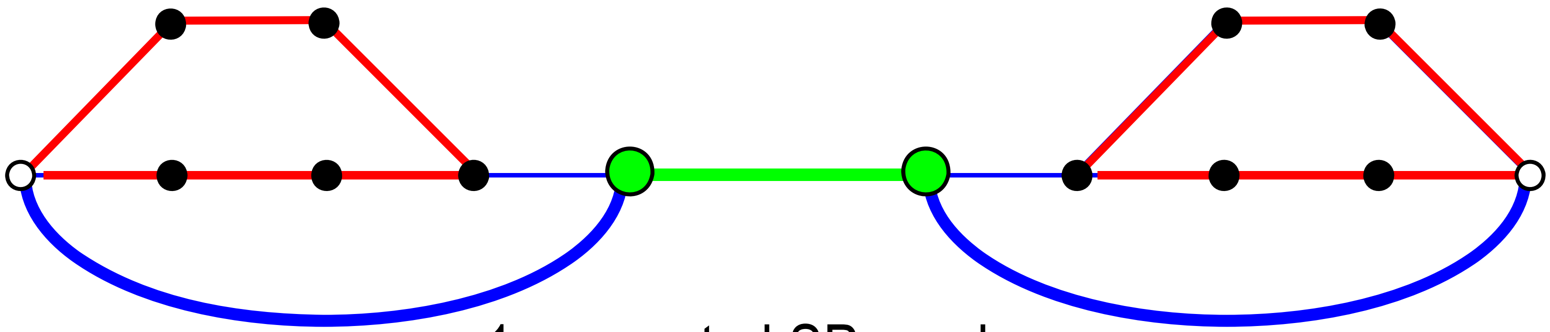
orthogonal drawings

0 bend



optimal

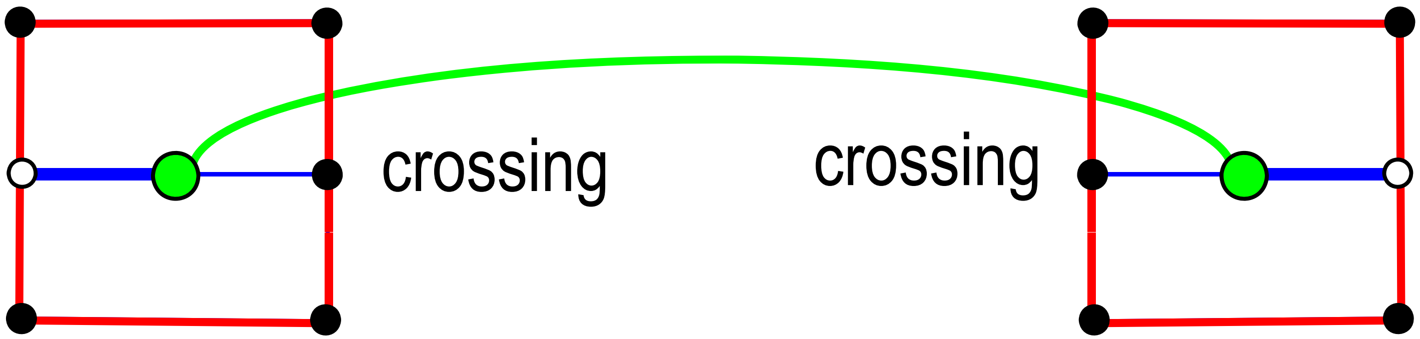
Optimal orthogonal drawing



1-connected SP graph

∃ a 0-bend orthogonal drawing ?

No

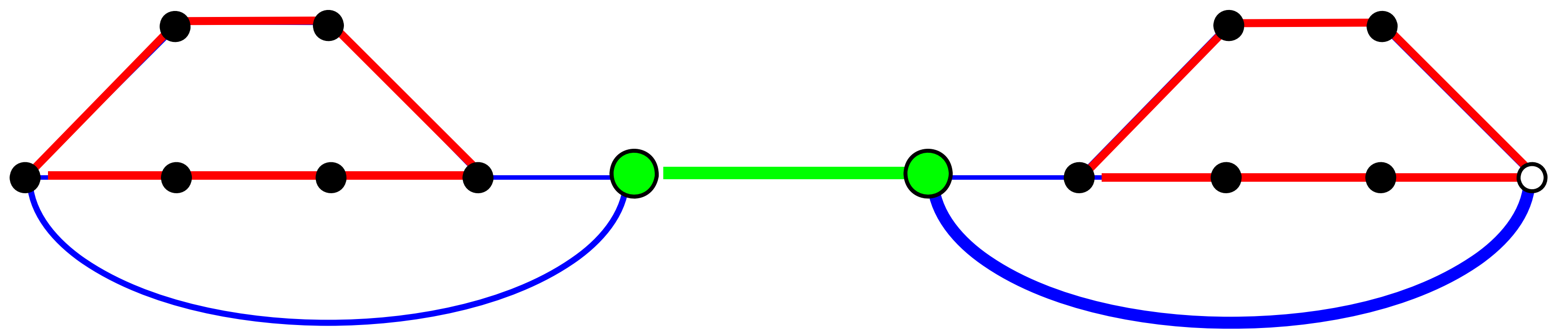


no bend

optimal

no bend

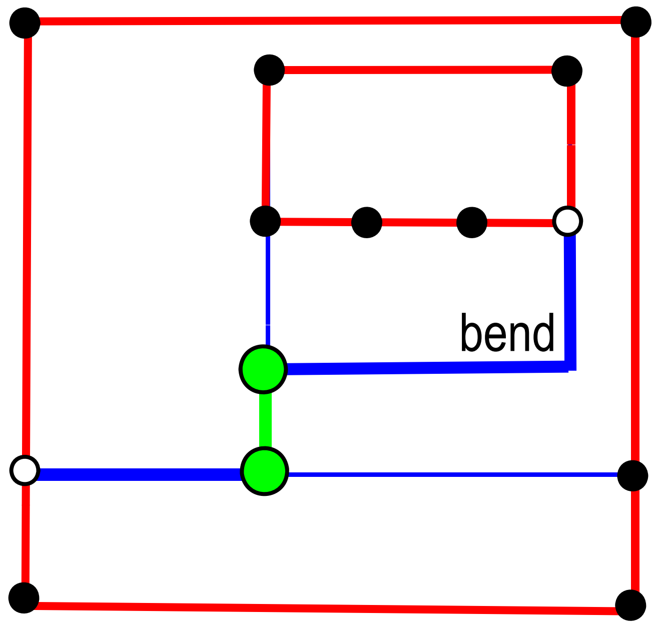
optimal

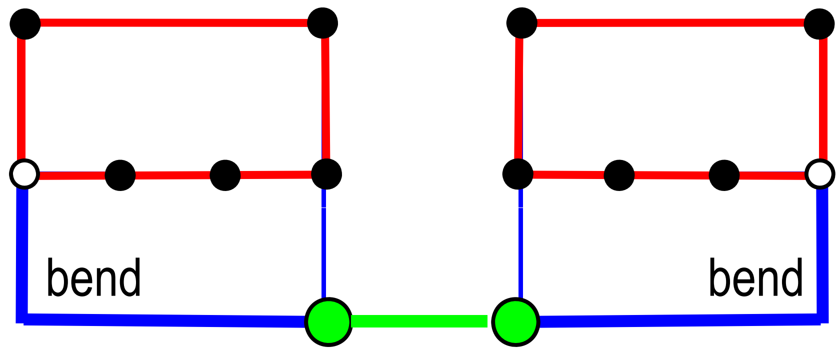


one bend

optimal

1-connected SP graph





two bends

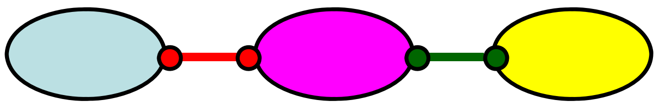
Conclusions

Theorem 1

An optimal orthogonal drawing of a biconnected SP graph G of $\Delta \leqq 3$ can be found in linear time.

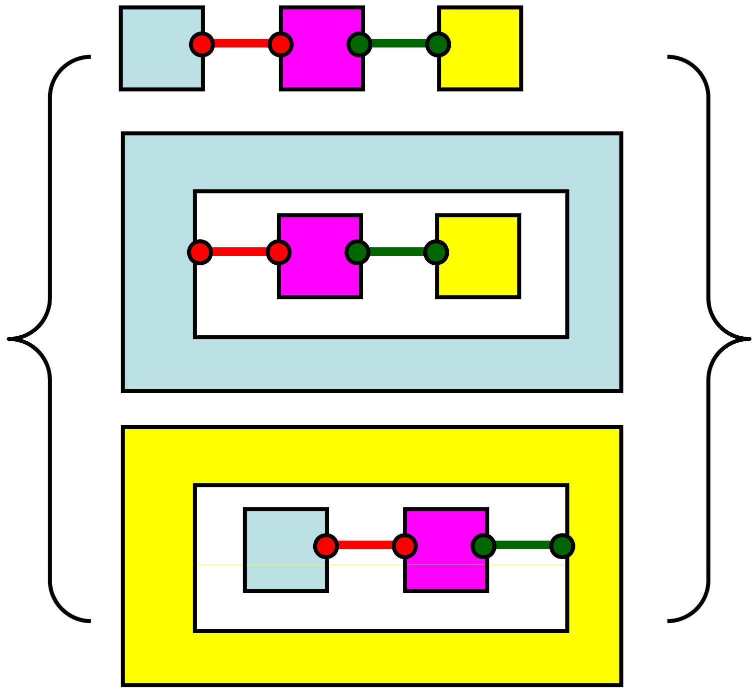
Our algorithm works well even if G is not biconnected.

min # of bends



1-connected SP graph

= min # of bends

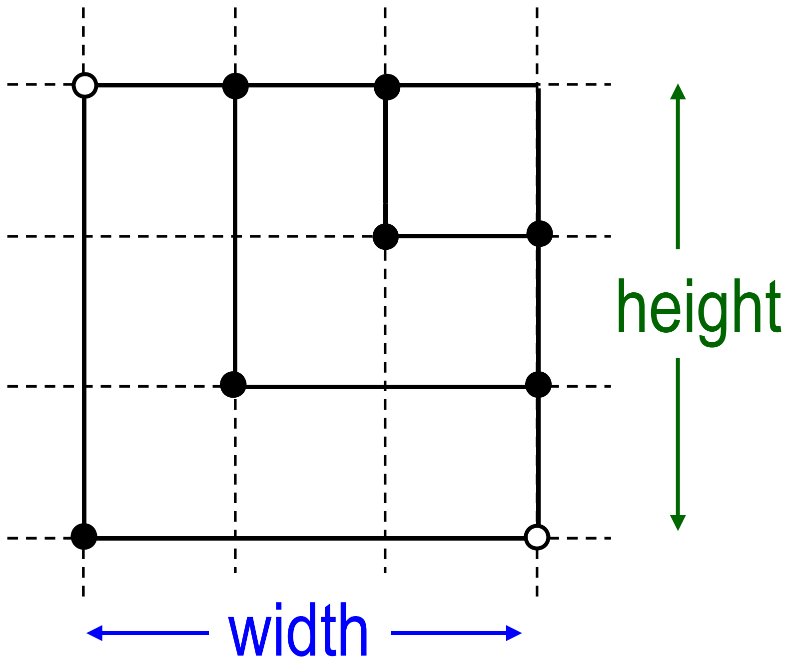


Conclusions

bend(G) $\leqq \lceil n/3 \rceil$ for biconnected SP graphs G of $\Delta \leqq 3$

Grid size $\leqq$ $8n/9$

=width + height

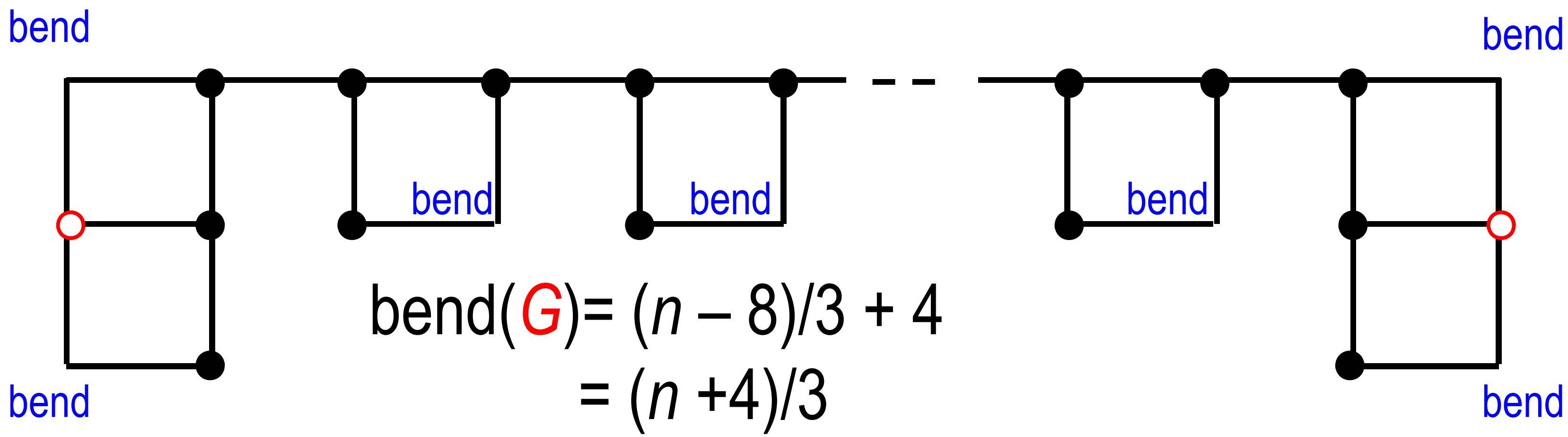


Conclusions

Our algorithm works well even if G is not biconnected.

$\mathrm{bend}(G) \leqq (n+4)/3$ for SP graphs G of $\Delta \leqq 3$

Best possible

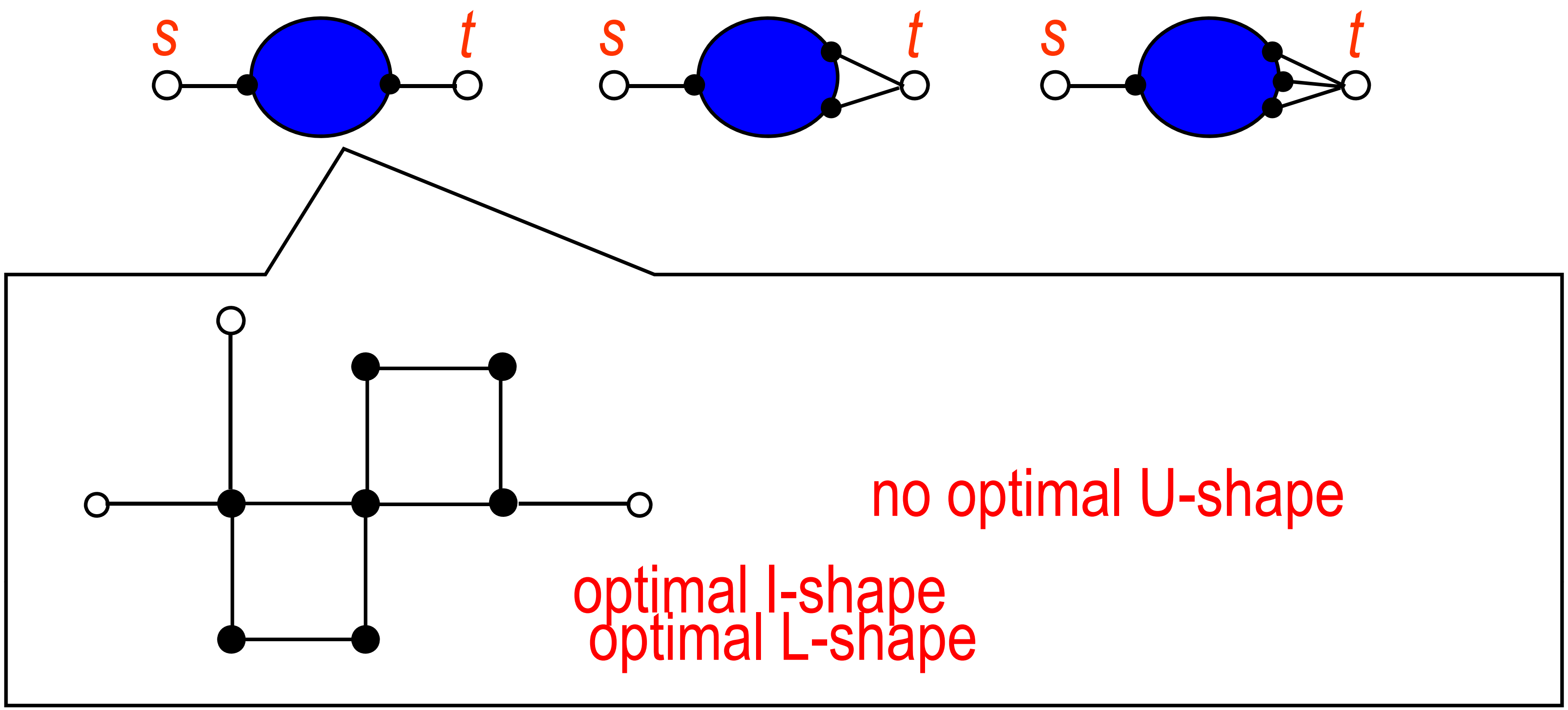


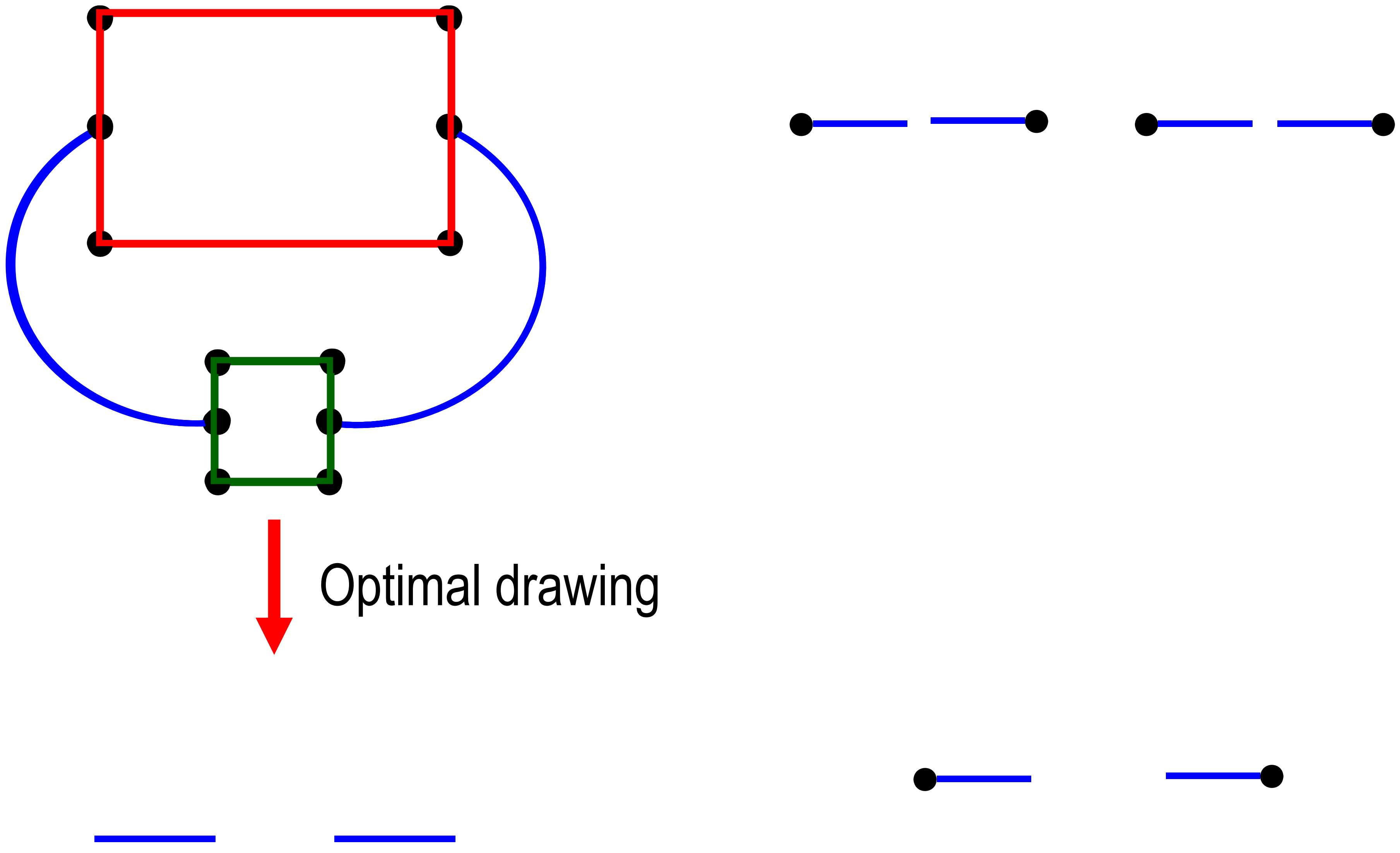
$$\mathrm{bend}(G) = (n-8)/3 + 4$$
$$= (n+4)/3$$

For series-parallel graphs *G* with *Δ*=4

Is there an O(*n*)-time algorithm to find an optimal orthogonal drawing of *G* ?

open

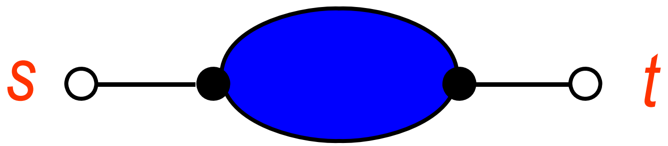


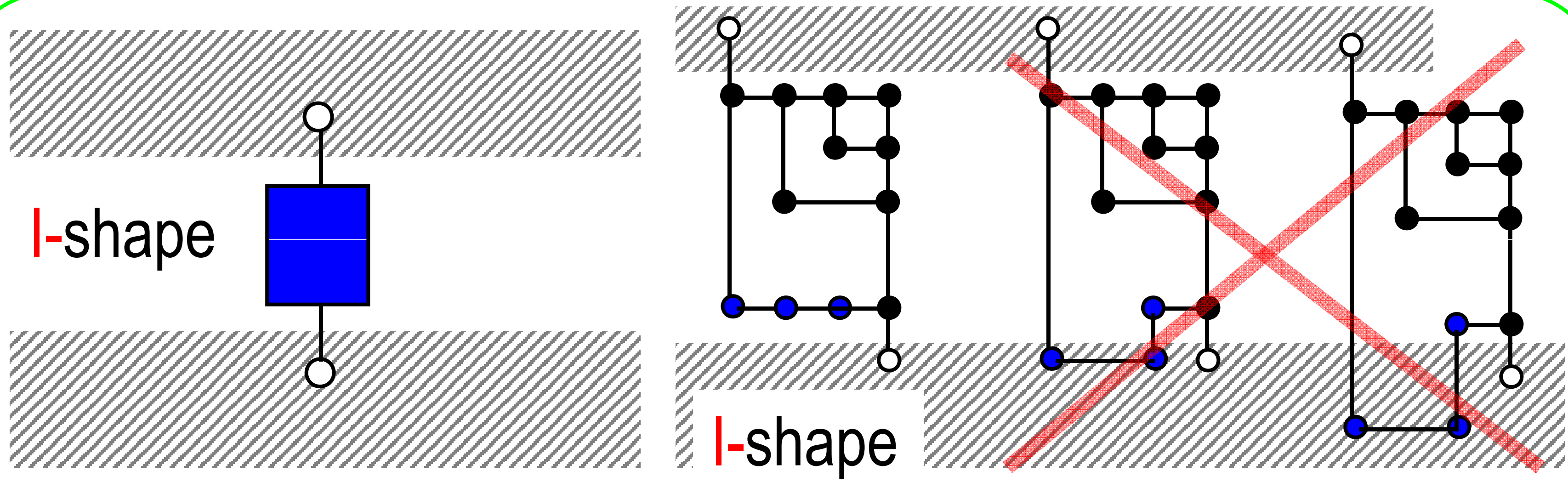

Optimal drawing

Our Main Idea

2-legged SP graph

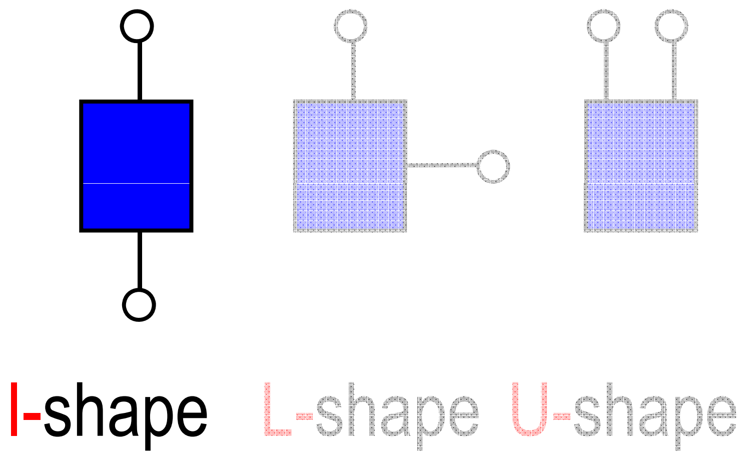
A SP graph G is 2-legged
if $n(G) \geqq 3$ and $d(s)=d(t)=1$ for the terminals s and t.



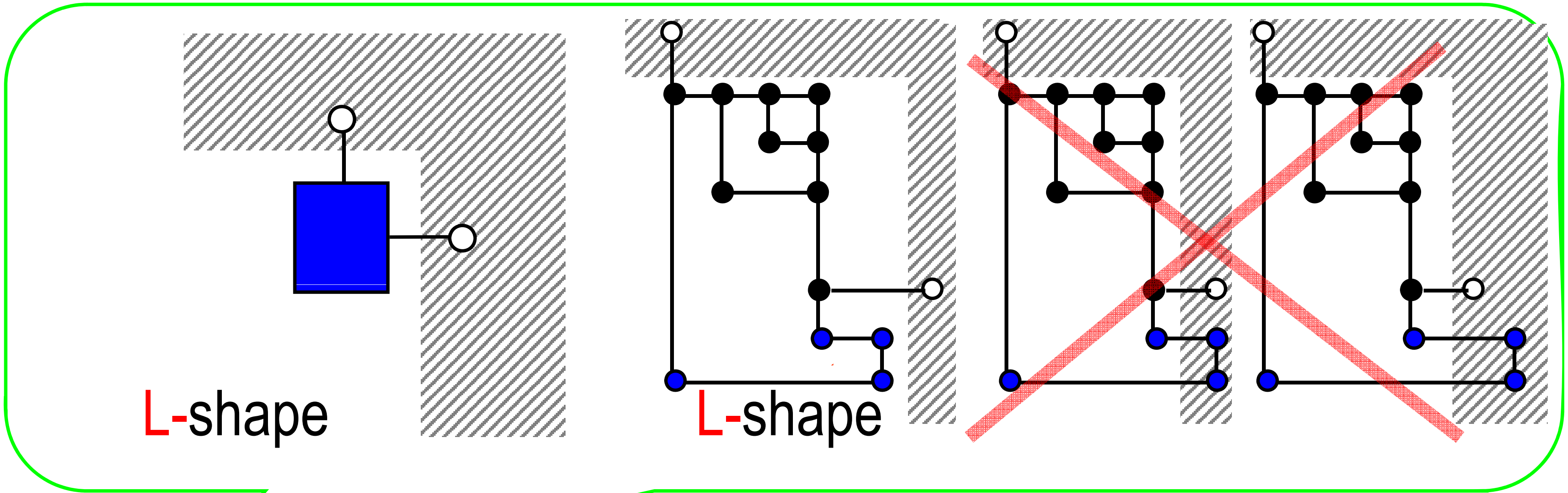


Definition of I- , L- and U-shaped drawings

- terminals are drawn on the outer face;
- the drawing except terminals intersects neither the north side nor the south side

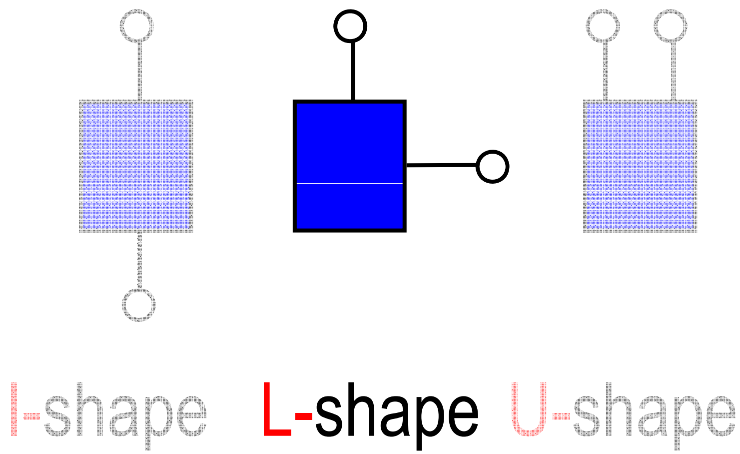


Our Main Idea

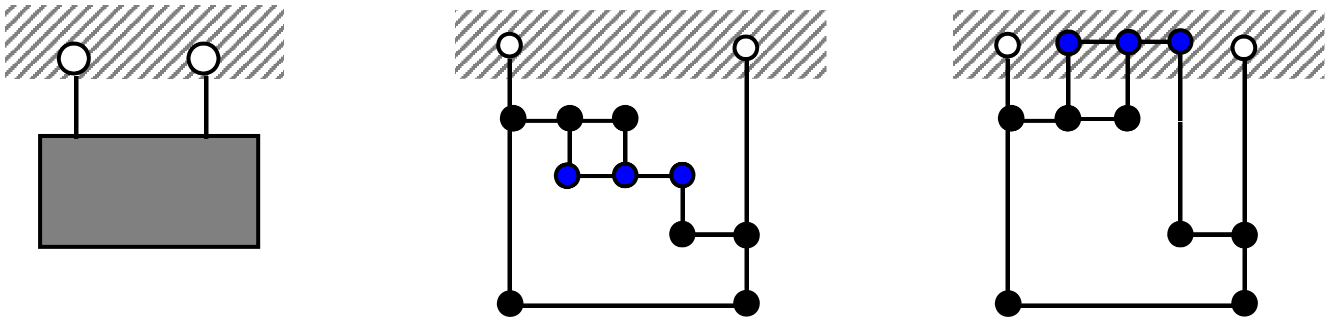


Defin… , L- and U-shaped drawings

- terminals are drawn on the outer face;
- the drawing except terminals intersects neither the north side nor the east side



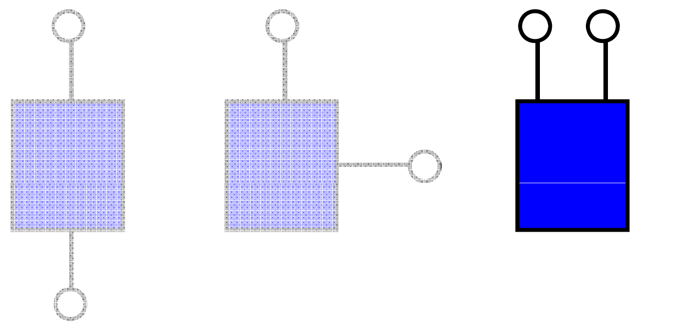
Our Main Idea



U-shape   U-shape   not U-shape

Defi[illegible] I-, L- and U-shaped drawings

- terminals are drawn on the outer face;
- the drawing except terminals doesn't intersect the north side



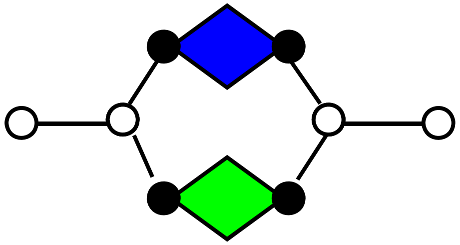
I-shape   L-shape   U-shape

Lemma 2

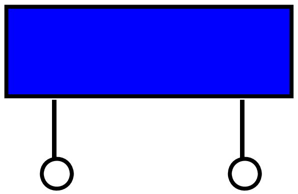
The following (a) and (b) hold for a 2-legged SP graph *G* of $\Delta \leqq 3$ unless *G* has a diamond:

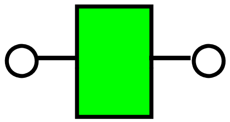
(a) *G* has three optimal I-, L- and U-shaped drawings

(b) such drawings can be found in linear time.



optimal U-shaped drawings

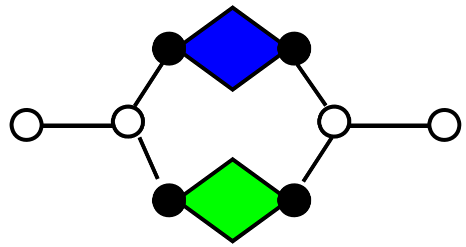




optimal I-shaped drawings

Lemma 2

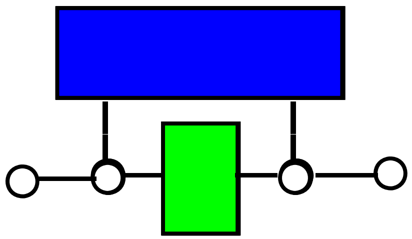
The following (a) and (b) hold for a 2-legged SP graph *G* of $\Delta \leqq 3$ unless *G* has a diamond:



(a) *G* has three optimal I-, L- and U-shaped drawings

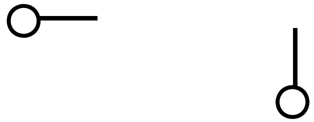
(b) such drawings can be found in linear time.

optimal



I-shape

optimal



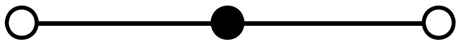
L-shape

optimal

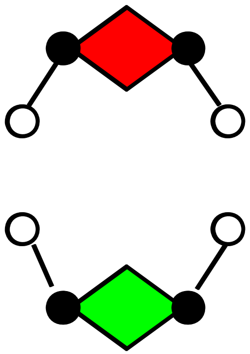
U-shape

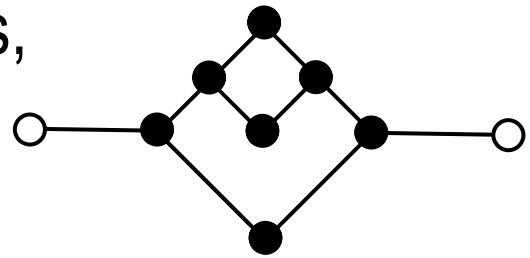
Definition of Diamond Graph

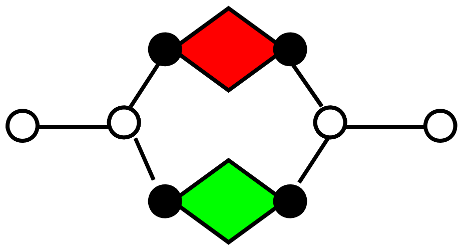
A Diamond graph is recursively defined as follows:

(a)  is a diamond graph.

a path with three vertices

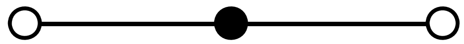
(b) if  are diamond graphs,



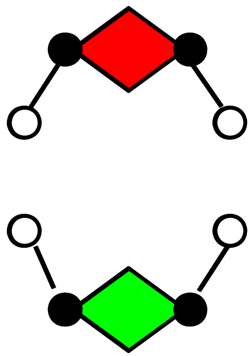
then  is a diamond graph

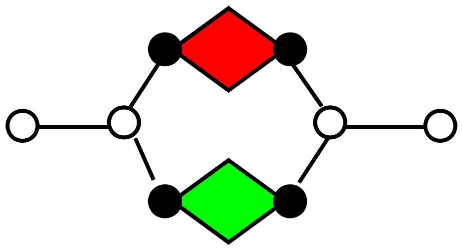
Definition of Diamond Graph

A Diamond graph is recursively defined as follows:

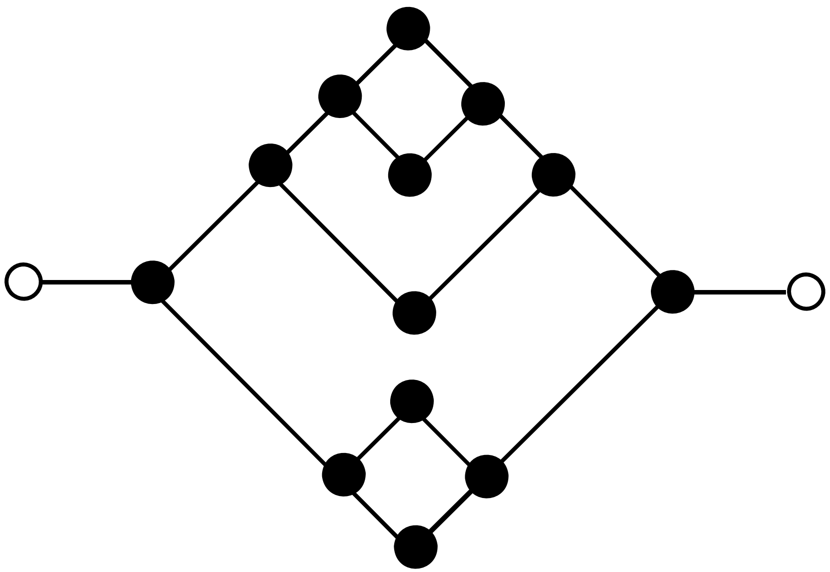
(a) 

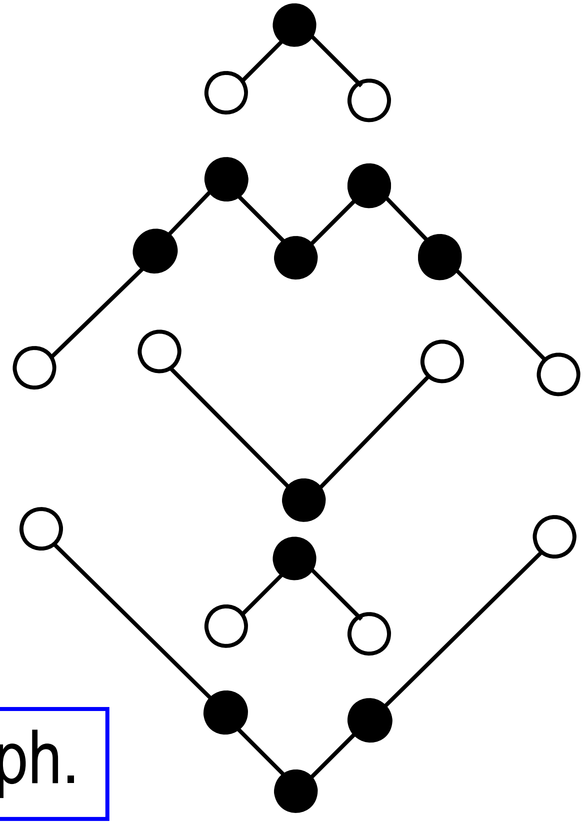
a path with three edges is a diamond graph.

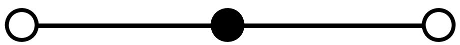
(b) if  are diamond graphs,

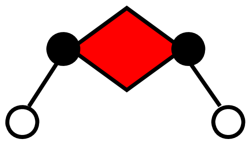
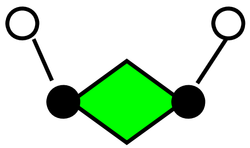
then  is a diamond graph

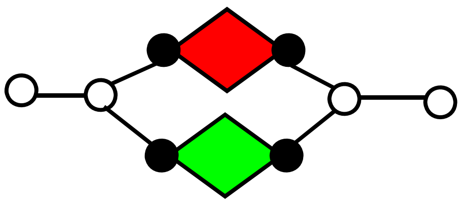
Diamond Graph



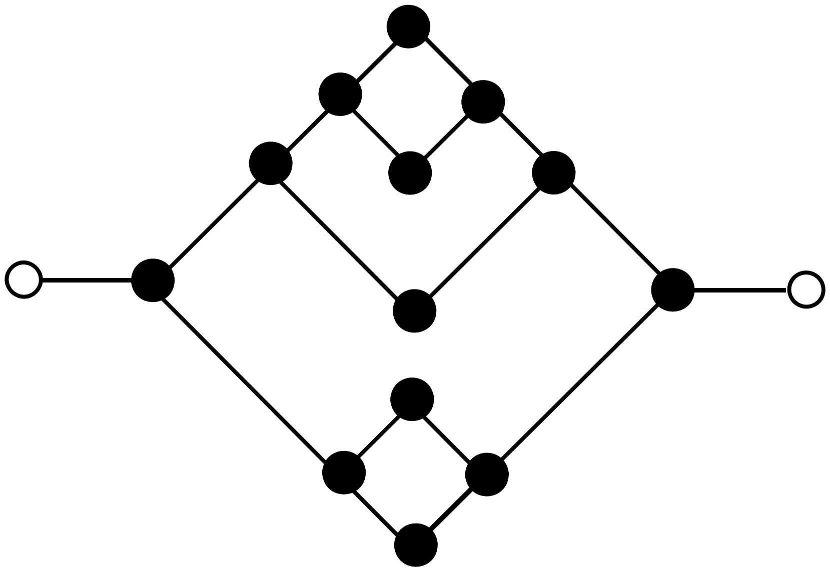


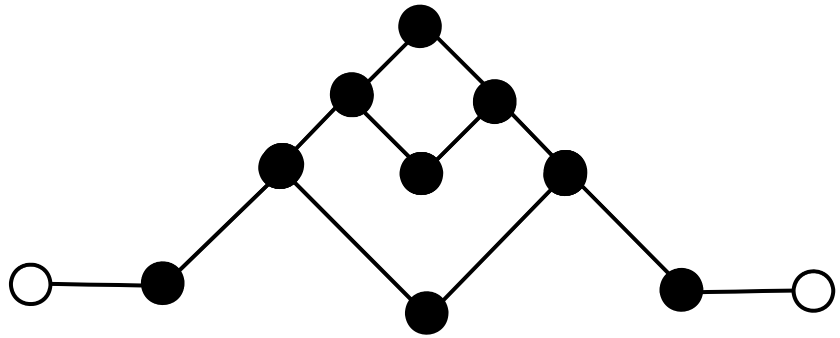
(a)  is a diamond graph.

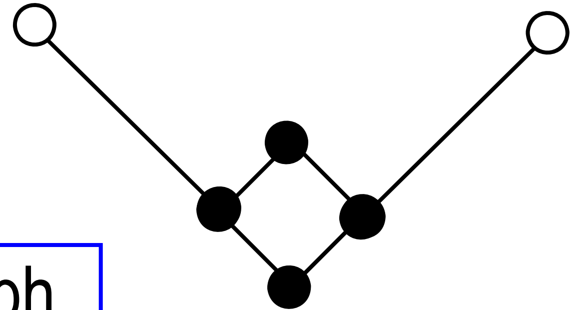
(b) If  and  are diamond graphs,

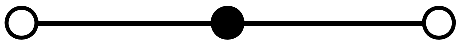
then  is a diamond graph

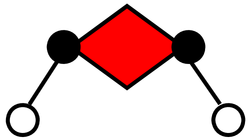
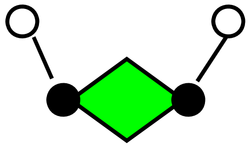
Diamond Graph

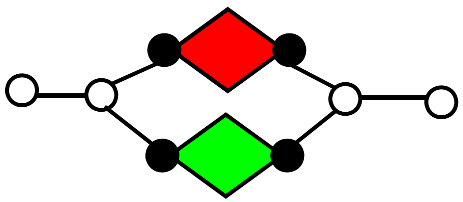






(a)  is a diamond graph.

(b) If  and  are diamond graphs,

then  is a diamond graph

Lemma 1

If G is a diamond graph, then

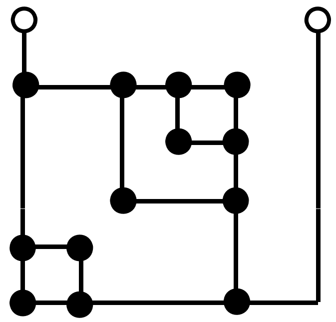
(a) G has both

a no-bend I-shaped dra

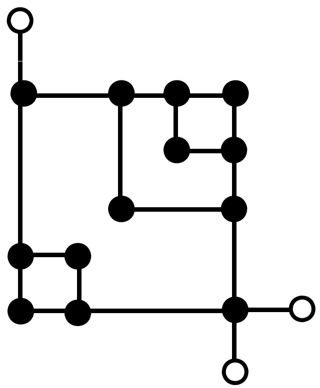
and

a no-bend L-shaped dra

(b) every no-bend drawing is eithe aped or L-shaped.



1-bend
U-shaped
drawing



∃ no-bend U-shaped drawing ?

NO

Lemma 1

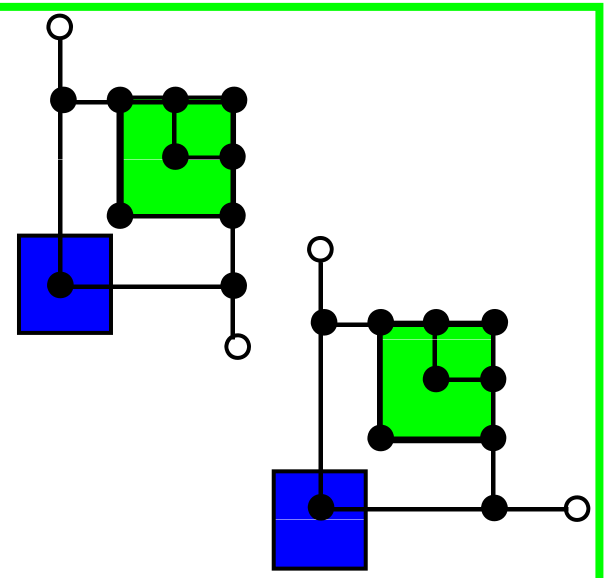
If *G* is a diamond graph, then

(a) *G* has both

a no-bend I-shaped drawing

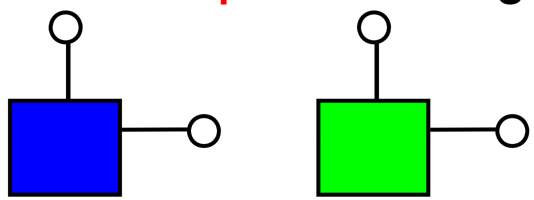
and

a no-bend L-shaped drawing



(b) every no-bend drawing is either I-shaped or L-shaped.

no-bend I-shaped drawings



Lemma 1

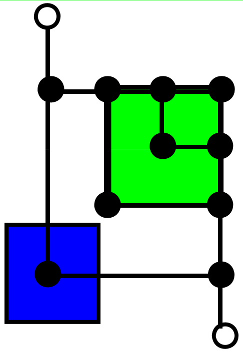
If *G* is a diamond graph, then

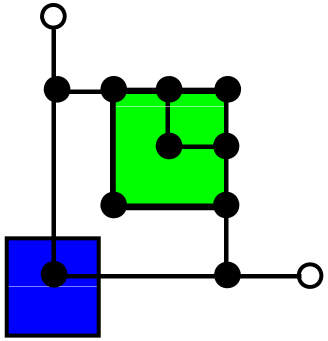
(a) *G* has both

a no-bend I-shaped drawing

and

a no-bend L-shaped drawing

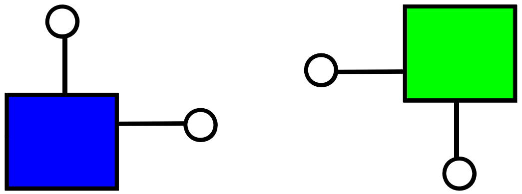




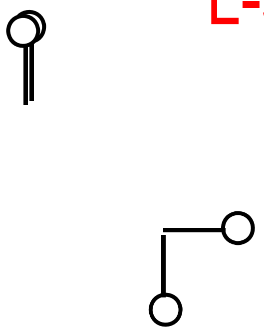
(b) every no-bend drawing is either I-shaped or L-shaped.

Such drawings can be found in linear time.

no-bend I-shaped drawings



L-shaped



Lemma 2

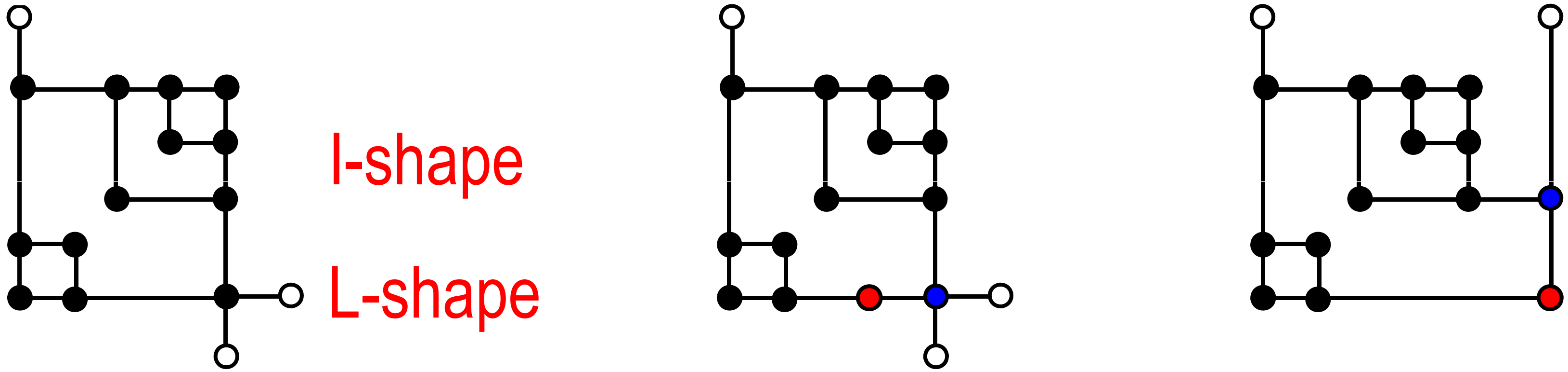
The following (a) and (b) hold for a 2-legged SP graph G of $\Delta \leqq 3$ unless G is a diamond graph

(a) G has three optimal I-, L- and U-shaped drawings

(b) such drawings can be found in linear time.

diamond graph

not a diamond graph



no-bend drawing

L-shape

U-shape

Lemma 2

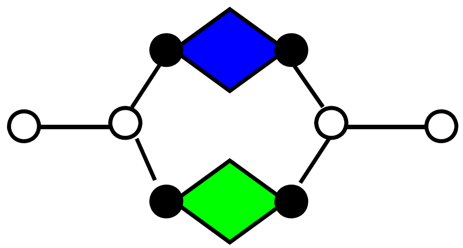
The following (a) and (b) hold for a 2-legged SP graph G of $\Delta \leqq 3$ unless G is a diamond graph

(a) G has three optimal

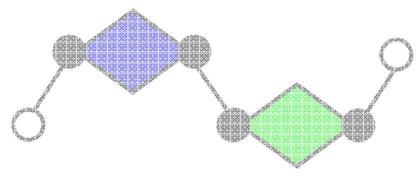
I-, L- and U-shaped drawings

(b) such drawings can be found in linear time.

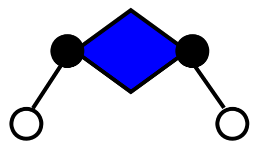
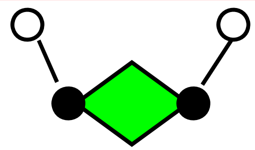
Proof: by an induction on # of vertices.



Parallel-connection



Series-connection

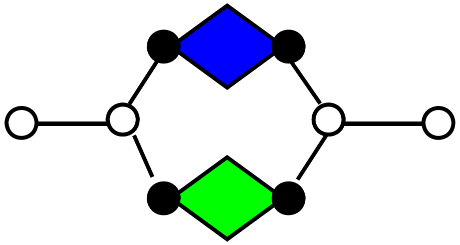
Suppose that the lemma holds true for  and 

Lemma 2

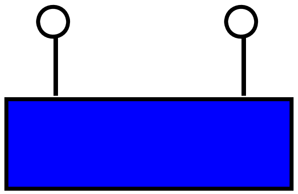
The following (a) and (b) hold for a 2-legged SP graph G of $\Delta \leqq 3$ unless G is a diamond graph

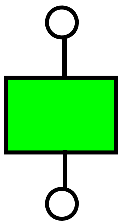
(a) G has three optimal I-, L- and U-shaped drawings

(b) such drawings can be found in linear time.



optimal U-shaped drawings





optimal I-shaped drawings

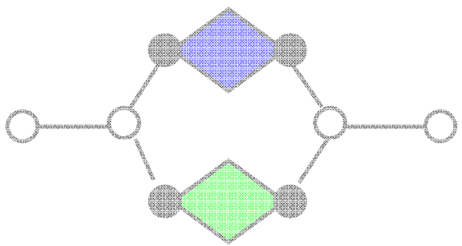
Lemma 2

The following (a) and (b) hold for a 2-legged SP graph G of $\Delta \leqq 3$ unless G is a diamond graph

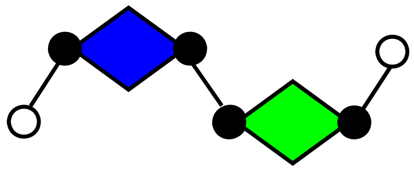
(a) G has three optimal
I-, L- and U-shaped drawings

(b) such drawings can be found in linear time.

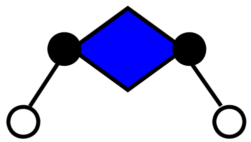
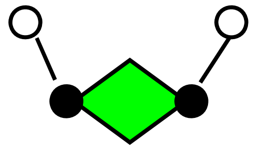
Proof: by an induction on # of vertices.



Parallel-connection



Series-connection

Suppose that the lemma holds true for  and 

Known results

In the fixed embedding setting:

n : # of vertices

For plane graph:

$O(n^2 \log n)$ time — R. Tamassia, 1987

Min-cost flow problem

Known results

In the fixed embedding setting:

n : # of vertices

For plane graph:

$O(n^2 \log n)$ time — R. Tamassia, 1987

$O(n^{7/4}\sqrt{\log n})$ time — A. Garg, R. Tamassia, 1997

improved

Known results

In the fixed embedding setting:

For plane graph:

$O(n^2 \log n)$ time — R. Tamassia, 1987

$O(n^{7/4}\sqrt{\log n})$ time — A. Garg, R. Tamassia, 1997

In the variable embedding setting:

NP-complete for planar graphs of $\Delta \leqq 4$

A. Garg, R. Tamassia, 2001

Known results

In the fixed embedding setting:

For plane graph:

$O(n^2 \log n)$ time — R. Tamassia, 1987

$O(n^{7/4}\sqrt{\log n})$ time — A. Garg, R. Tamassia, 1997

In the variable embedding setting:

NP-complete for planar graphs of $\Delta \leqq 4$

A. Garg, R. Tamassia, 2001

If $\Delta \leqq 3$, $O(n^5 \log n)$ time — D. Battista, *et al*. 1998

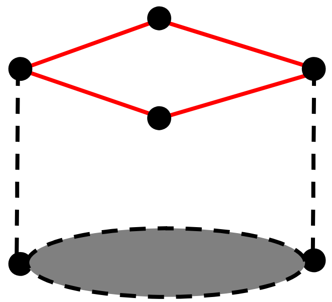
Lemma 3

Every biconnected SP graph G of $\Delta \leqq 3$ has one of the following three substructures:

(a) a diamond C

(b) two adjacent vertices u and v s.t. $d(u)=d(v)=2$

(c) a complete graph K_3.



(a) G

Lemma 1 (Our Main Idea)

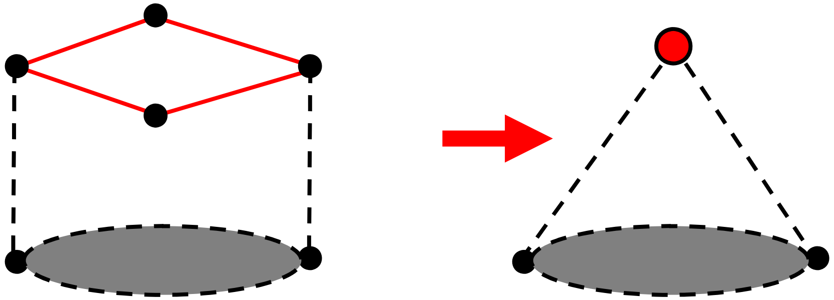
Every biconnected SP graph G of $\Delta \leqq 3$ has one of the following three substructures:

(a) a diamond

(b) two adjacent vertices u and v s.t. $d(u)=d(v)=2$

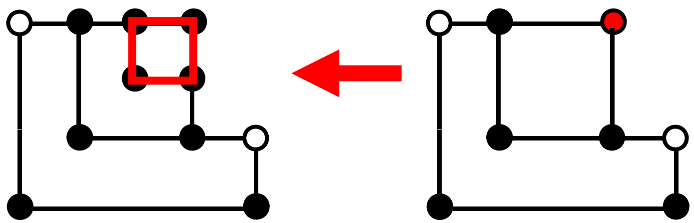
(c) a complete graph K_3.

bend(G)=bend(G')



G $\quad$ G'

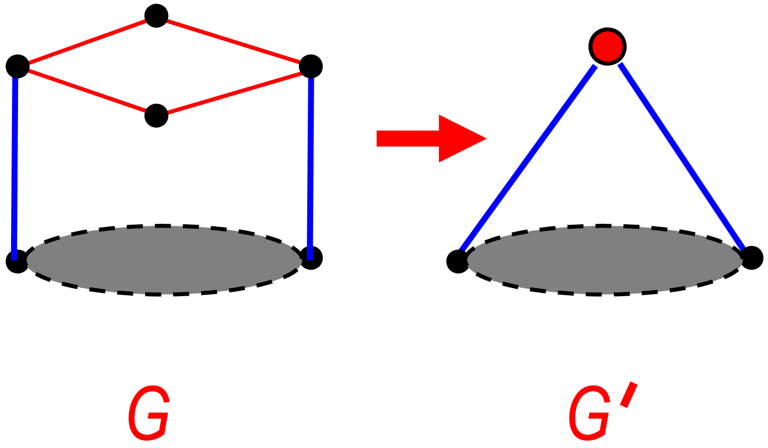
bend(G)=bend(G')=0



G $\quad$ G'

Proof

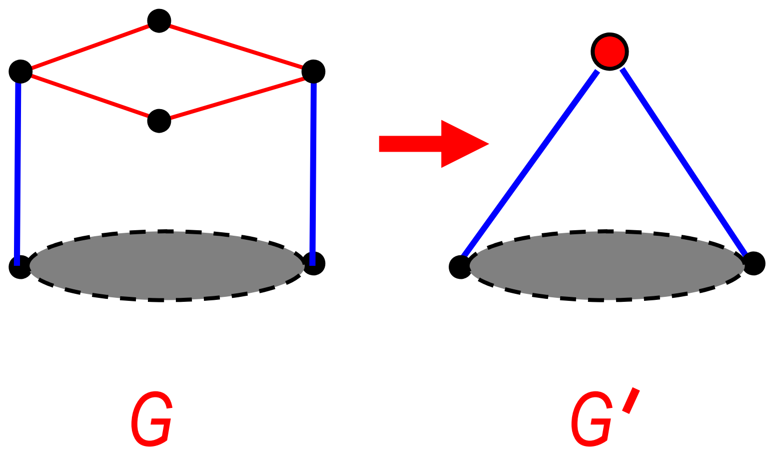
bend(G)=bend(G')

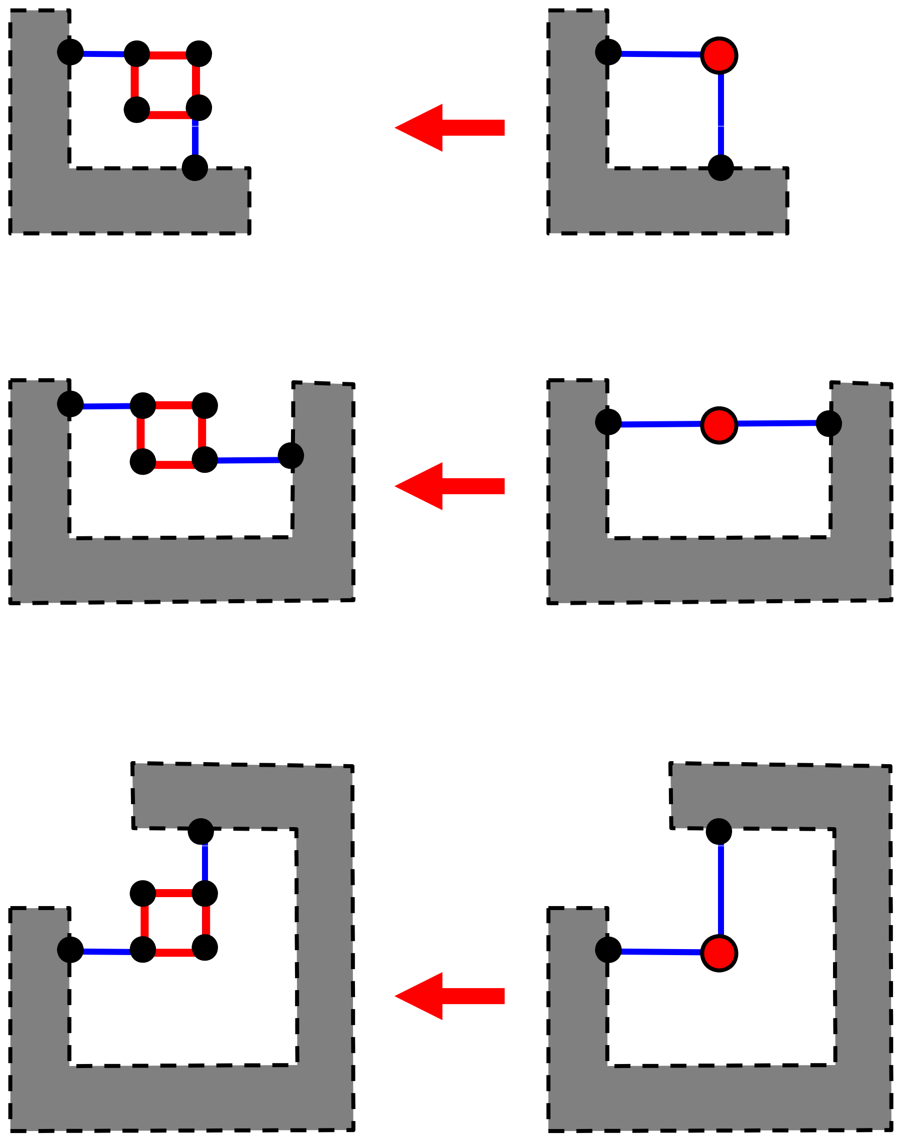


Proof

Given an optimal drawing of G'

bend(G) ≦ bend(G')

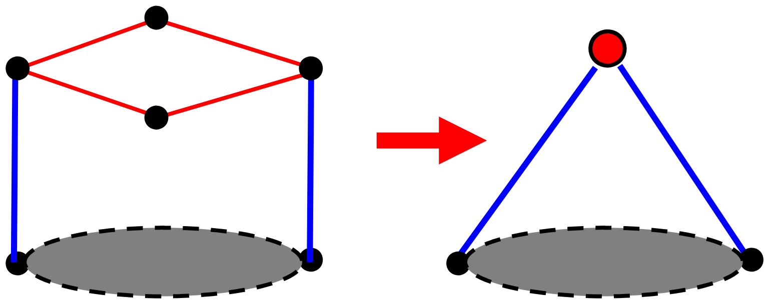




Proof

Given an optimal drawing of G

$\mathrm{bend}(G) \geqq \mathrm{bend}(G')$



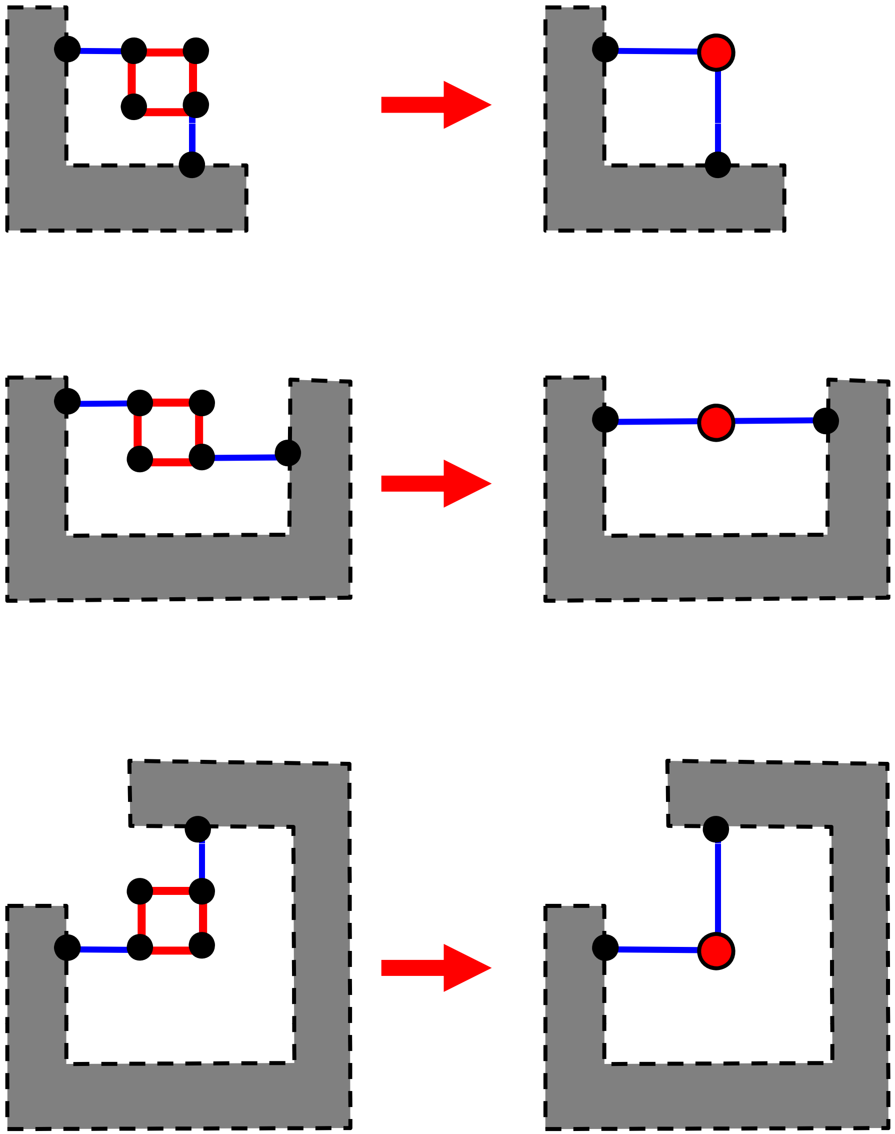
G G'

Case 1: bend( ) = 0

Case 2: bend( ) = 1

omitted

Case 3: bend( ) $\geq$ 2



Lemma 1 (Our Main Idea)

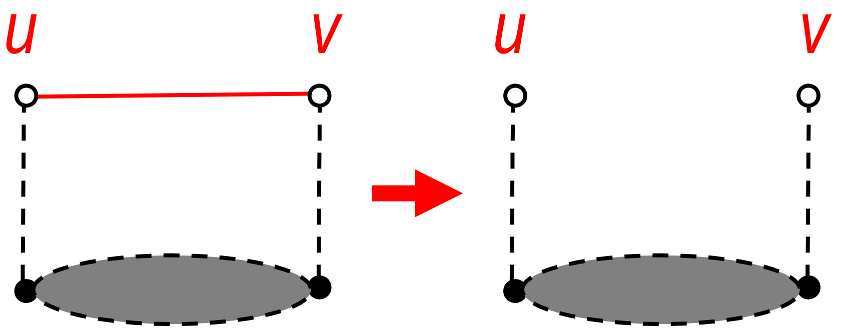
Every biconnected SP graph G of $\Delta \leqq 3$ has one of the following three substructures:

(a) a diamond

(b) two adjacent vertices u and v s.t. $d(u)=d(v)=2$

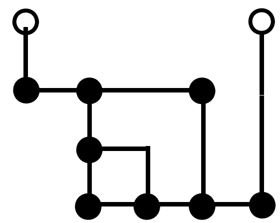
(c) a complete graph K_3.

bend(G)=bend(G')



G No diamonds G'

∃ an optimal U-shape drawing



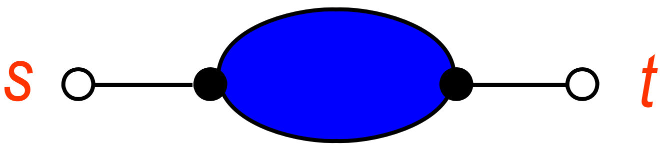
G'

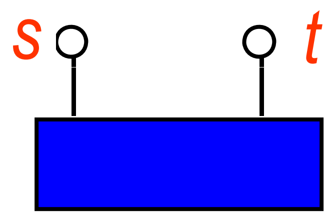
Lemma 2

?

For each SP graph *G′*
If n ≧ 4, Δ≦3 and d(s)=d(t)=1,
then ∃ an optimal U-shape drawing

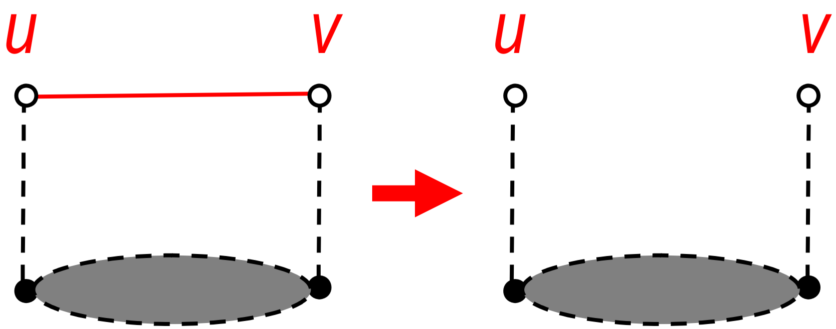
optimal U-shaped drawing



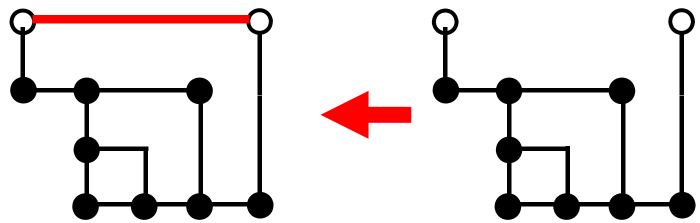


bend(*G*)=bend(*G′*)

∃ an optimal U-shape drawing

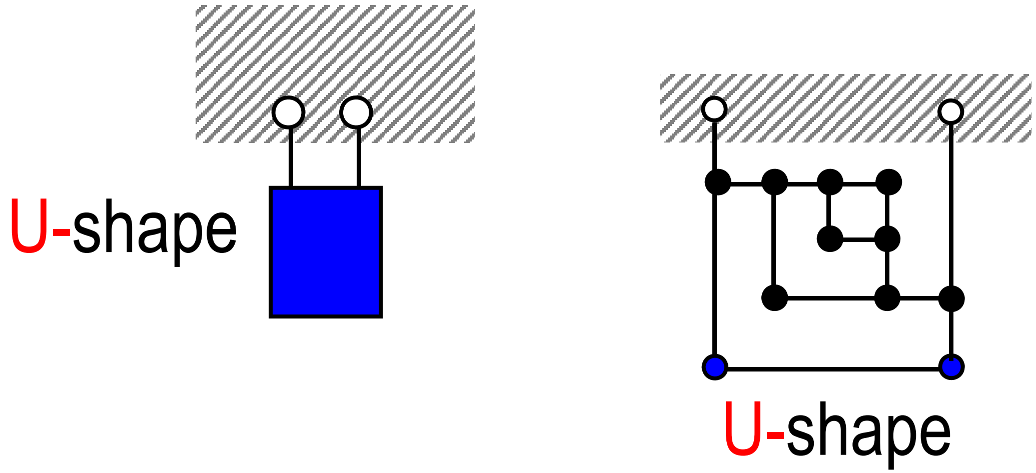


G No diamonds *G′*



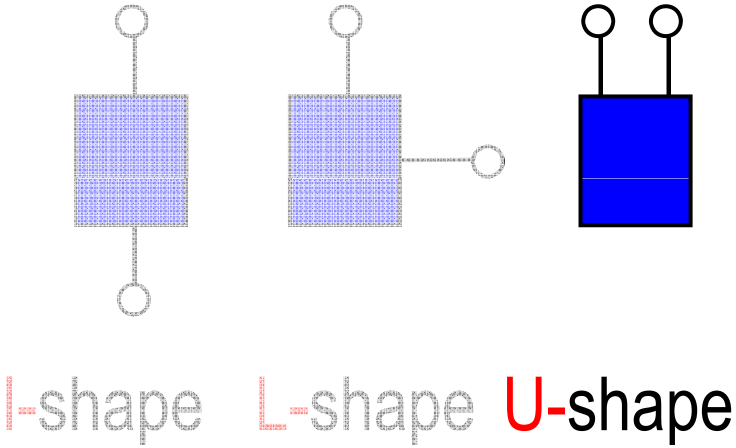
G *G′*

Our Main Idea



Definition of I-, L- and U-shaped drawings

- terminals are drawn on the outer face;
- the drawing except terminals doesn't intersect the north side



Lemma 1 (Our Main Idea)

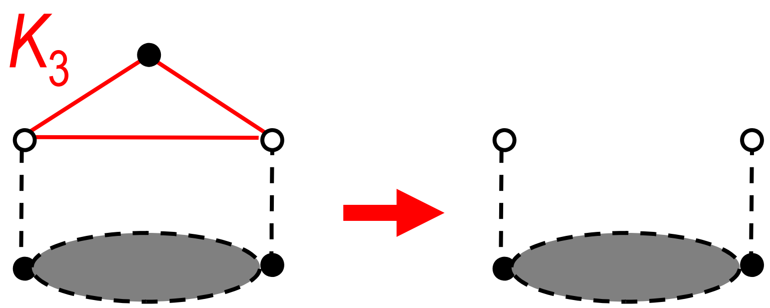
Every biconnected SP graph G of $\Delta \leqq 3$ has one of the following three substructures:

(a) a diamond

(b) two adjacent vertices u and v s.t. $d(u)=d(v)=2$

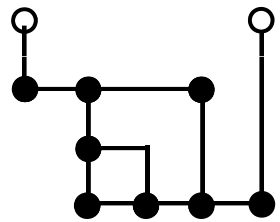
$\boxed{\text{(c) a complete graph } K_3.}$

bend(G)=bend(G')+1



G $\boxed{\text{No diamonds}}$ G'

$\boxed{\exists \text{ an optimal U-shape drawing}}$



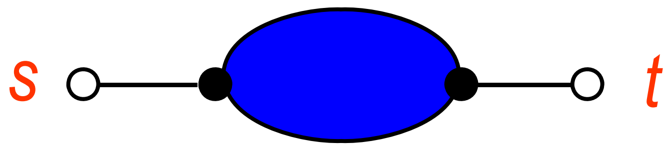
G'

Lemma 2

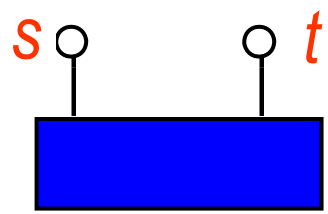
For each SP graph G'

If n ≧ 4, Δ≦3 and d(s)=d(t)=1,

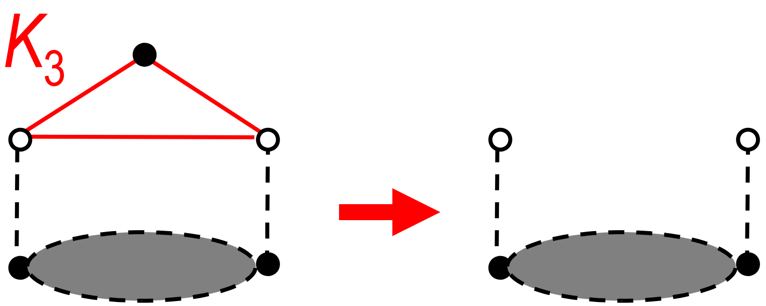
then ∃ an optimal U-shape drawing



optimal U-shaped drawing

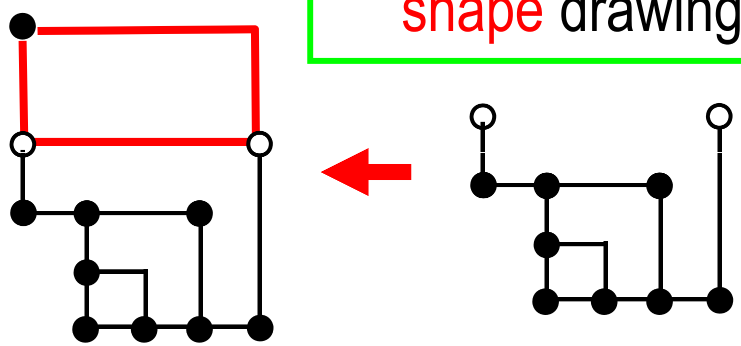


bend(G)=bend(G')+1



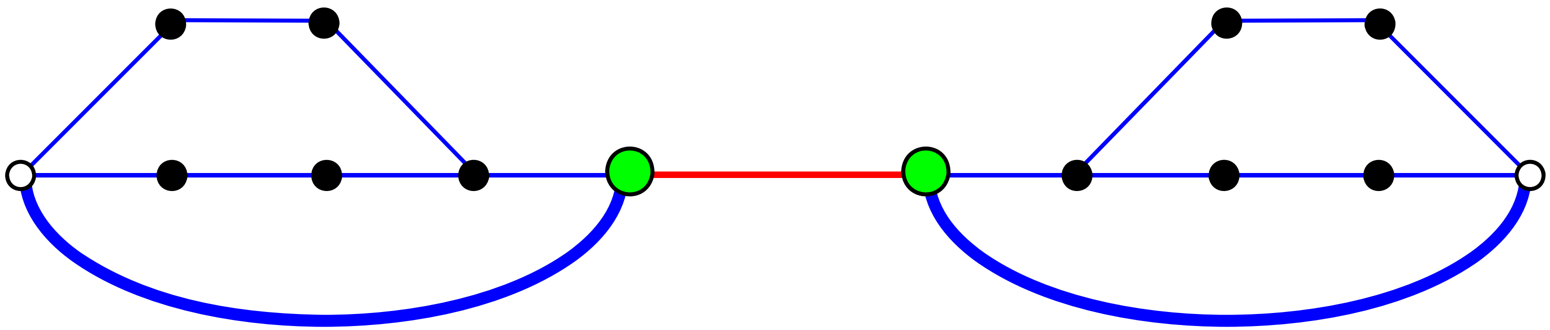
G No diamonds G'

∃ an optimal U-shape drawing



G G'

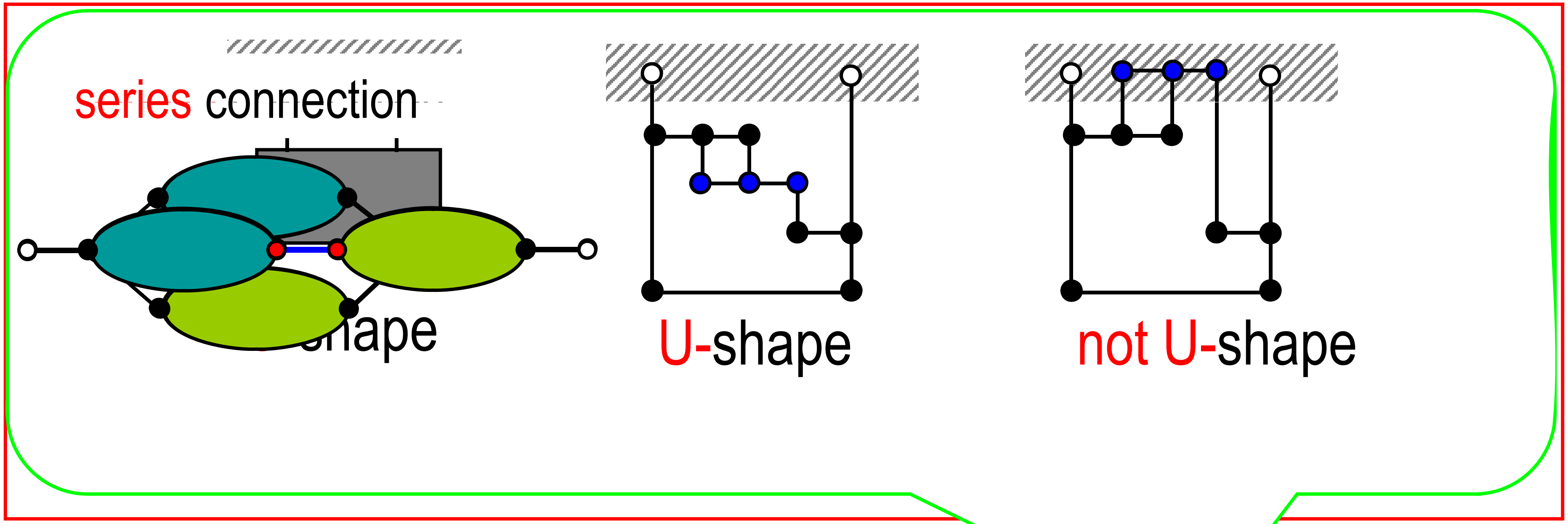
Optimal orthogonal drawing

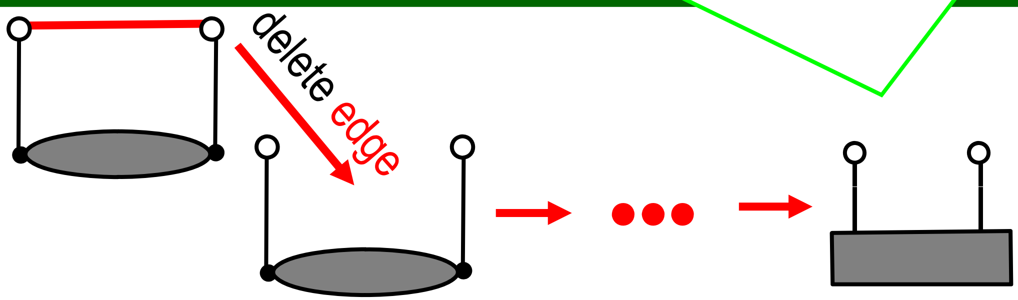


non -connected planar graph

Optimal orthogonal drawings ?

Case (b): ∃





Find an
opt & U-shape