An Interactive Tool to Explore and Improve the Ply Number of Drawings

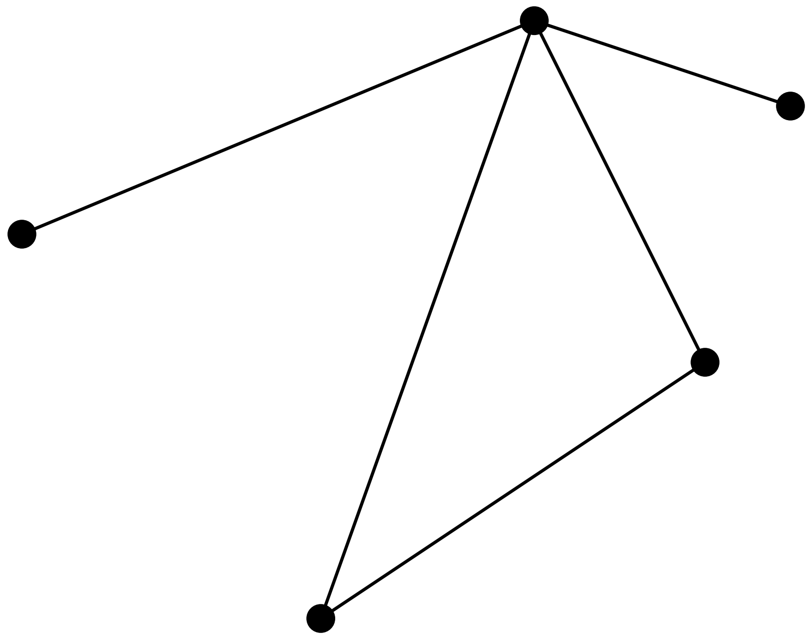
Niklas Heinsohn, Michael Kaufmann

Wilhelm-Schickhard-Institut für Informatik
Universität Tübingen

September 25, 2017

Ply Number of Drawings

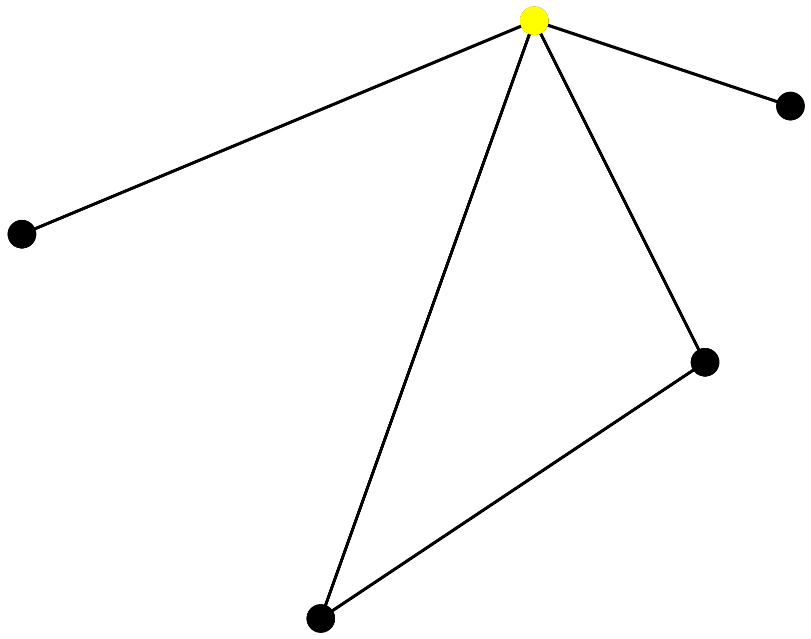
given a straight-line drawing in the plane



Ply Number of Drawings

given a straight-line drawing in the plane

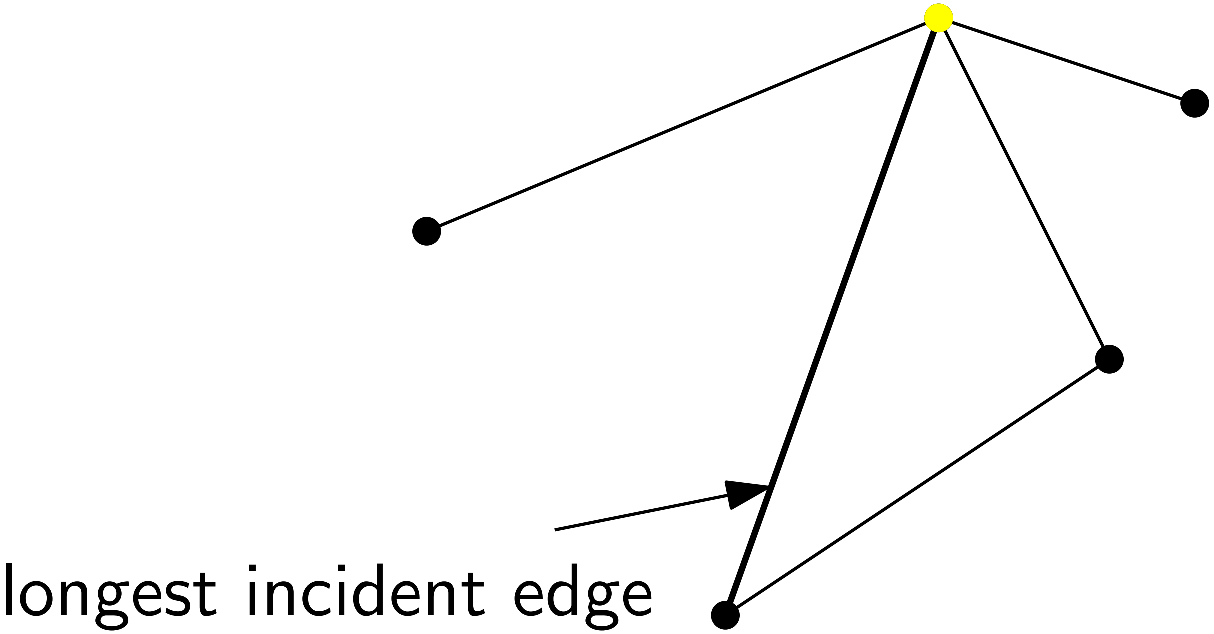
ply-disk $D(v)$ for every v:



Ply Number of Drawings

given a straight-line drawing in the plane

ply-disk $D(v)$ for every v:

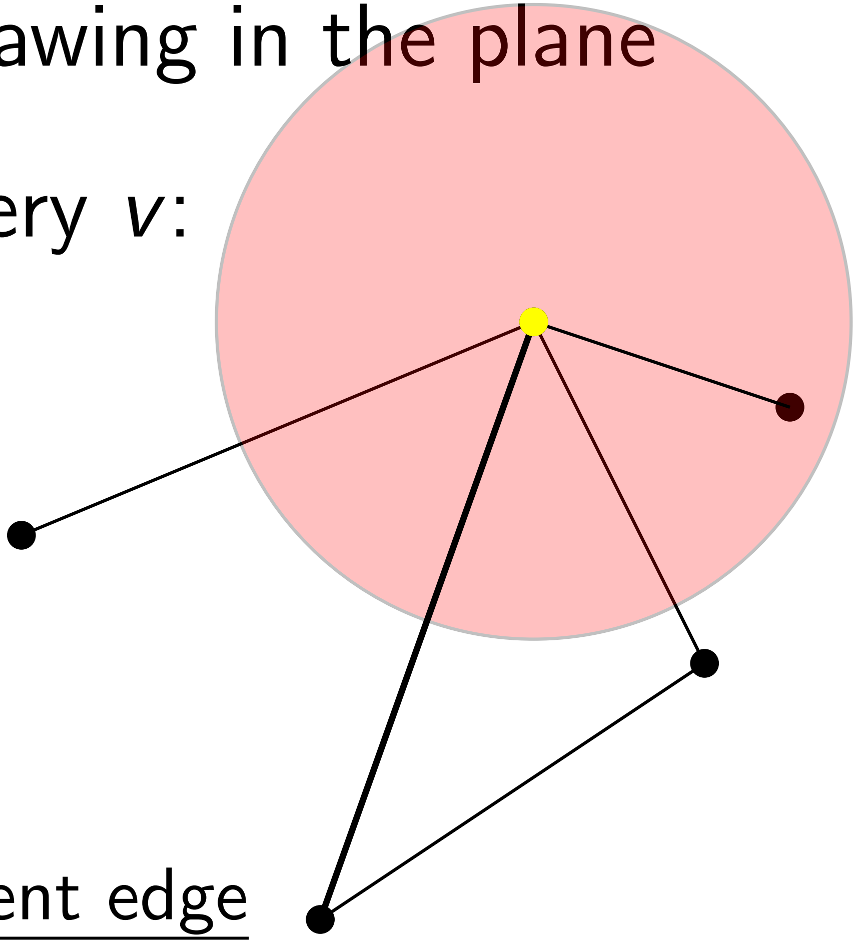


longest incident edge

Ply Number of Drawings

given a straight-line drawing in the plane

ply-disk $D(v)$ for every v:

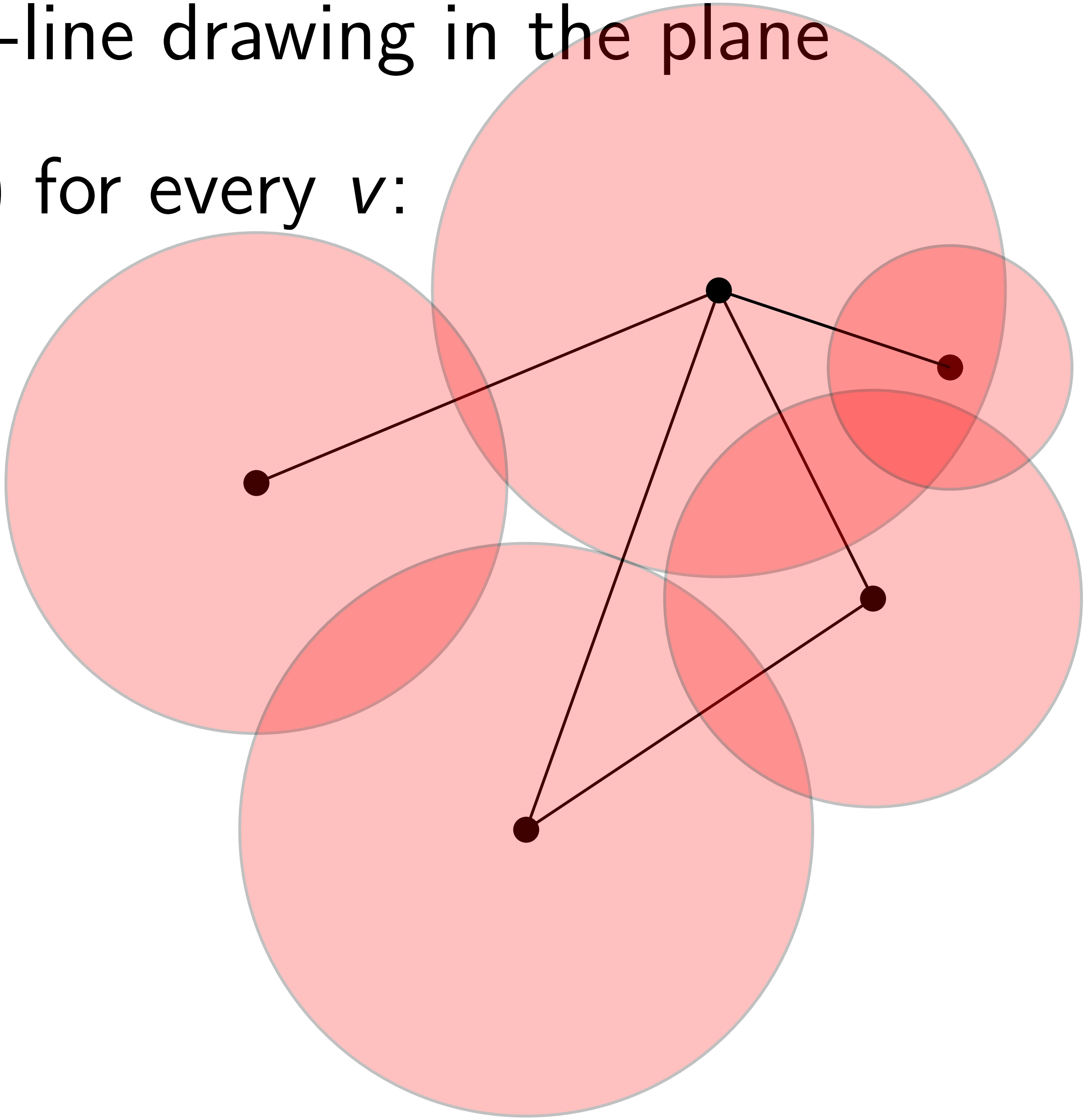


radius : $\dfrac{\text{longest incident edge}}{2}$

Ply Number of Drawings

given a straight-line drawing in the plane

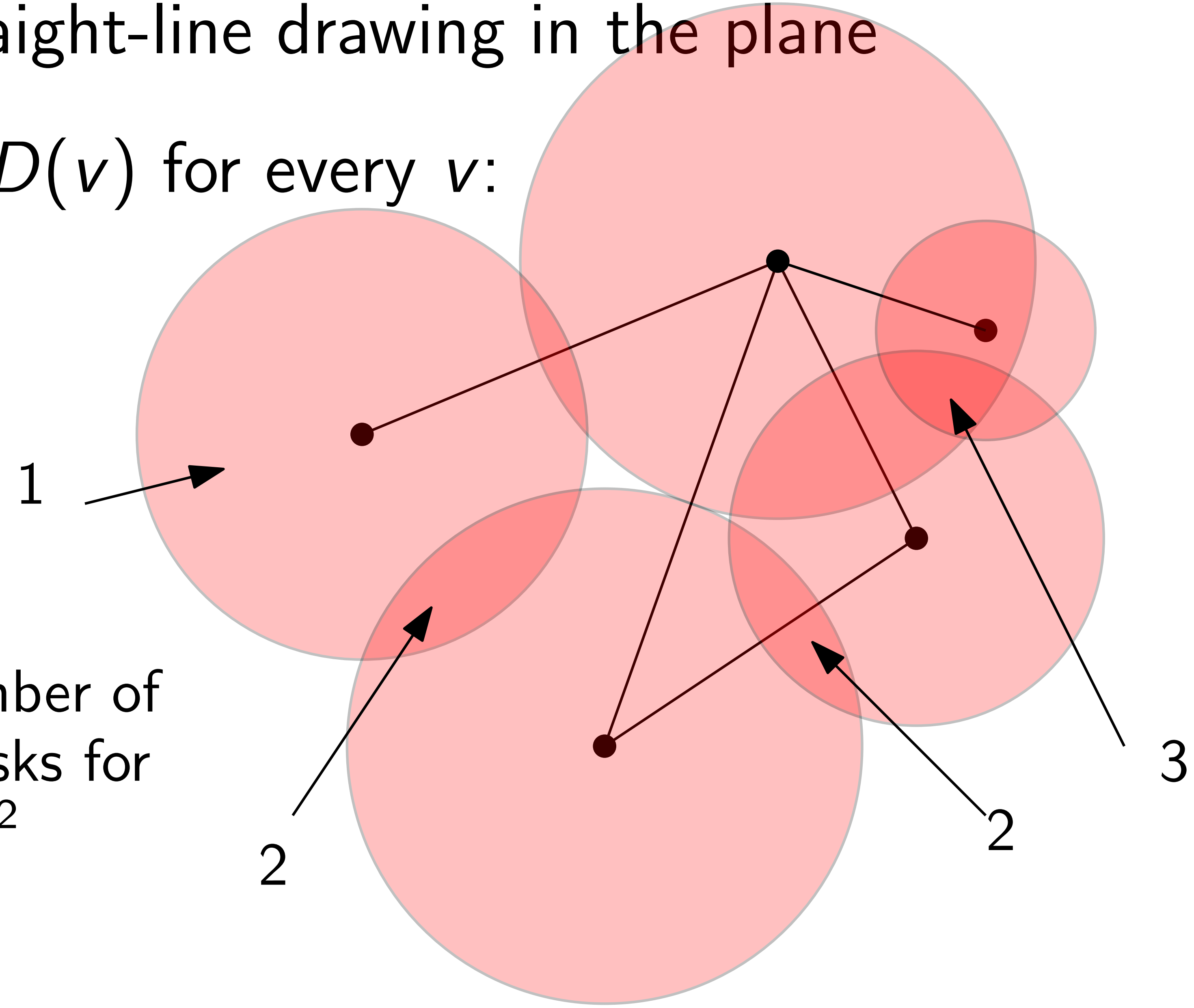
ply-disk $D(v)$ for every v:



Ply Number of Drawings

given a straight-line drawing in the plane

ply-disk $D(v)$ for every v:



maximum number of overlapping disks for all points in $\mathbb{R}^2$

Motivation

mixture of egde-length distribution
and distribution of the vertices

Motivation

mixture of egde-length distribution
and distribution of the vertices

indicator for the quality of a drawing

Motivation

mixture of egde-length distribution
and distribution of the vertices

indicator for the quality of a drawing

An Experimental Study on the Ply Number of Straight-line Drawings
[Felice De Luca et al - WALCOM 2017]

ply as parameter

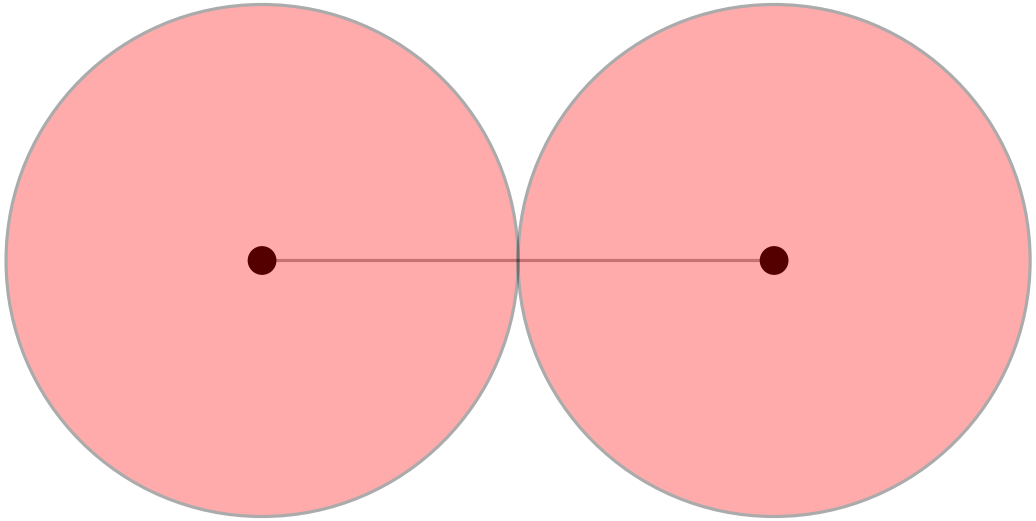
close relation to stress
force-directed algorithms on sparse graphs

Explore the Ply Number

for any drawing the ply number is at most $n - 1$

Explore the Ply Number

for any drawing the ply number is at most $n - 1$



ply-disks incident to the longest edge do not overlap

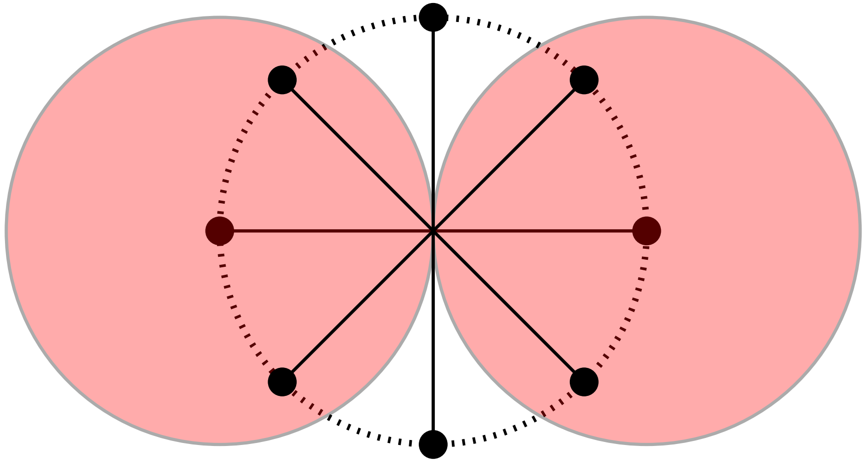
Explore the Ply Number

any graph can be drawn with $\leq \left\lfloor \frac{n}{2} \right\rfloor$ ply

Explore the Ply Number

any graph can be drawn with $\leq \lfloor \frac{n}{2} \rfloor$ ply

place vertices of K_n regularly on a circle

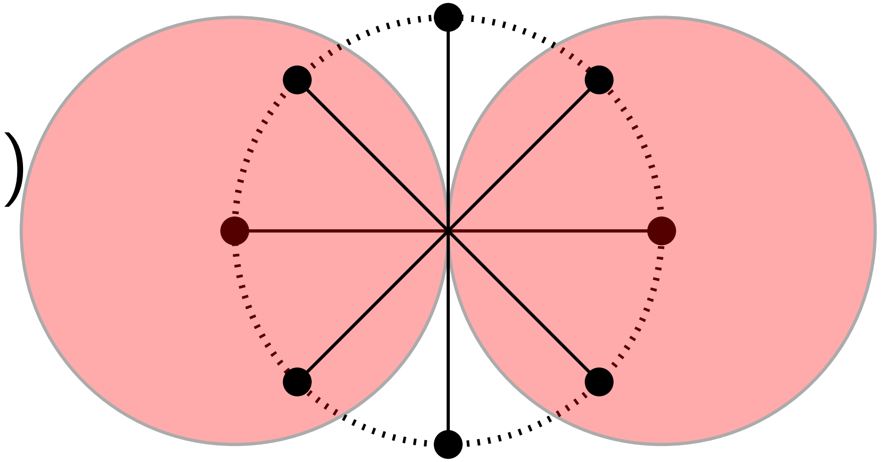


Explore the Ply Number

any graph can be drawn with $\leq \lfloor \frac{n}{2} \rfloor$ ply

place vertices of K_n regularly on a circle

for every ply-disk contributing to the ply number there exists (at least) one ply-disk, not contributing.

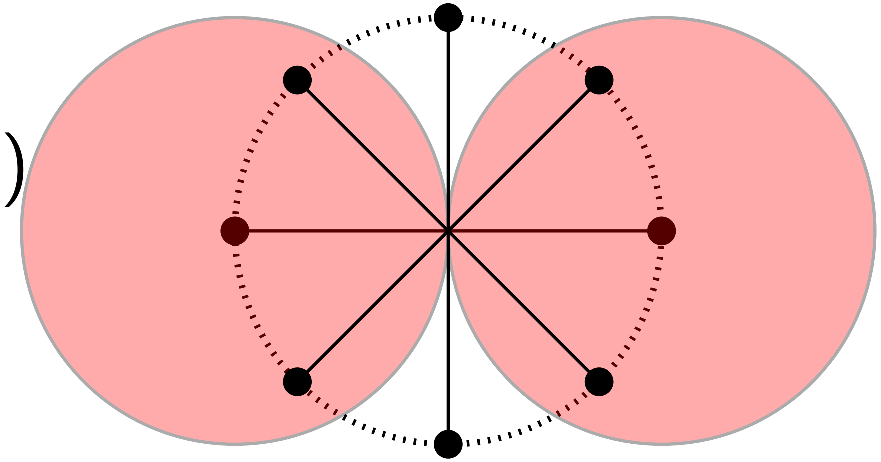


Explore the Ply Number

any graph can be drawn with $\leq \lfloor \frac{n}{2} \rfloor$ ply

place vertices of K_n regularly on a circle

for every ply-disk contributing to the ply number there exists (at least) one ply-disk, not contributing.



removing edges cannot increase the ply number

Explore the Ply Number

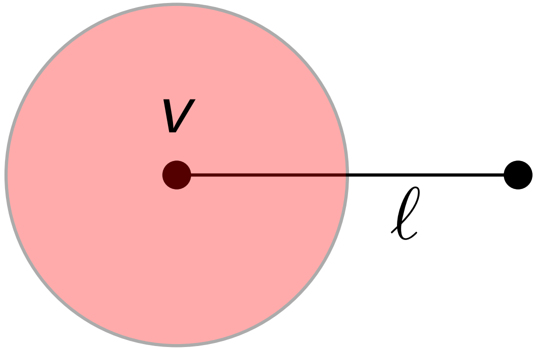
a drawing of a graph has ply 1 if

a) for any vertex v all edges have uniform length ℓ

Explore the Ply Number

a drawing of a graph has ply 1 if

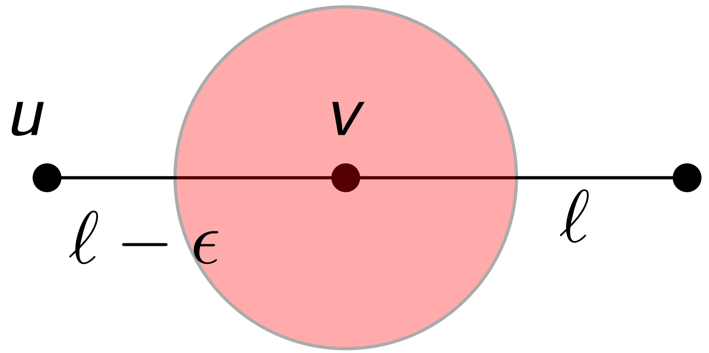
a) for any vertex v all edges have uniform length ℓ



Explore the Ply Number

a drawing of a graph has ply 1 if

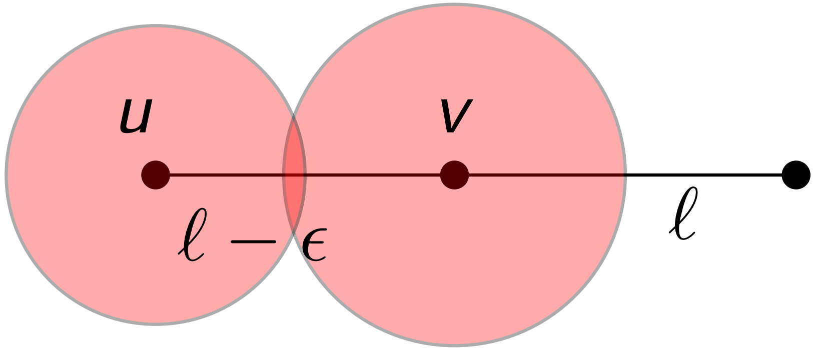
a) for any vertex v all edges have uniform length ℓ



Explore the Ply Number

a drawing of a graph has ply 1 if

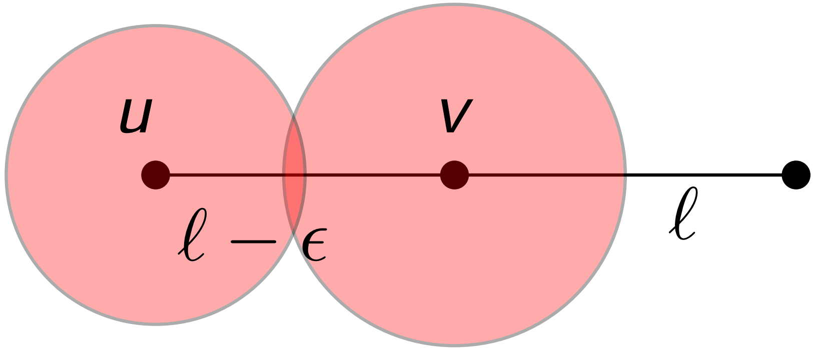
a) for any vertex v all edges have uniform length ℓ



Explore the Ply Number

a drawing of a graph has ply 1 if

a) for any vertex v all edges have uniform length ℓ



b) the distance between any two vertices is at least ℓ

Compute the Ply Number

implementation presented in [De Luca et al- WALCOM'17]

plane-sweep-algorithm

Compute the Ply Number

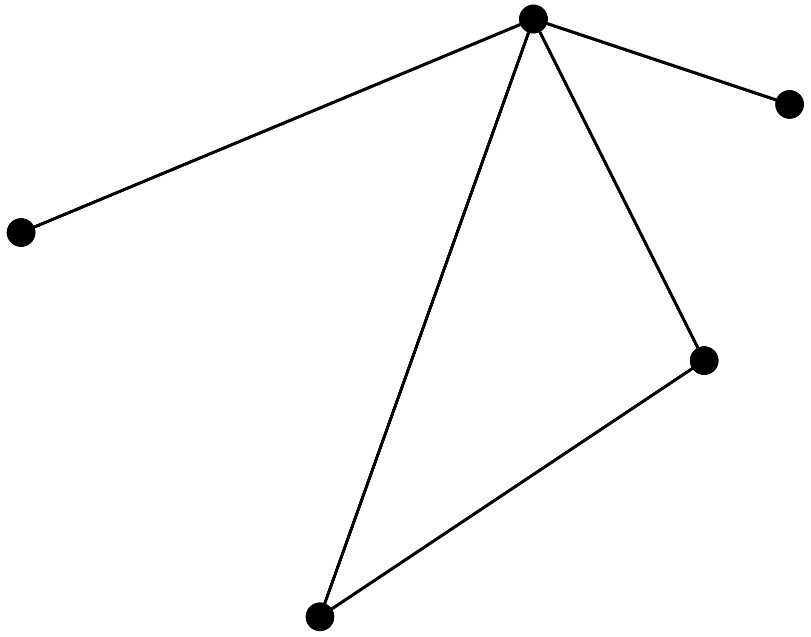
implementation presented in [De Luca et al- WALCOM'17]

plane-sweep-algorithm

- similar datastructure
- significantly faster

Plane-Sweep-Algorithm

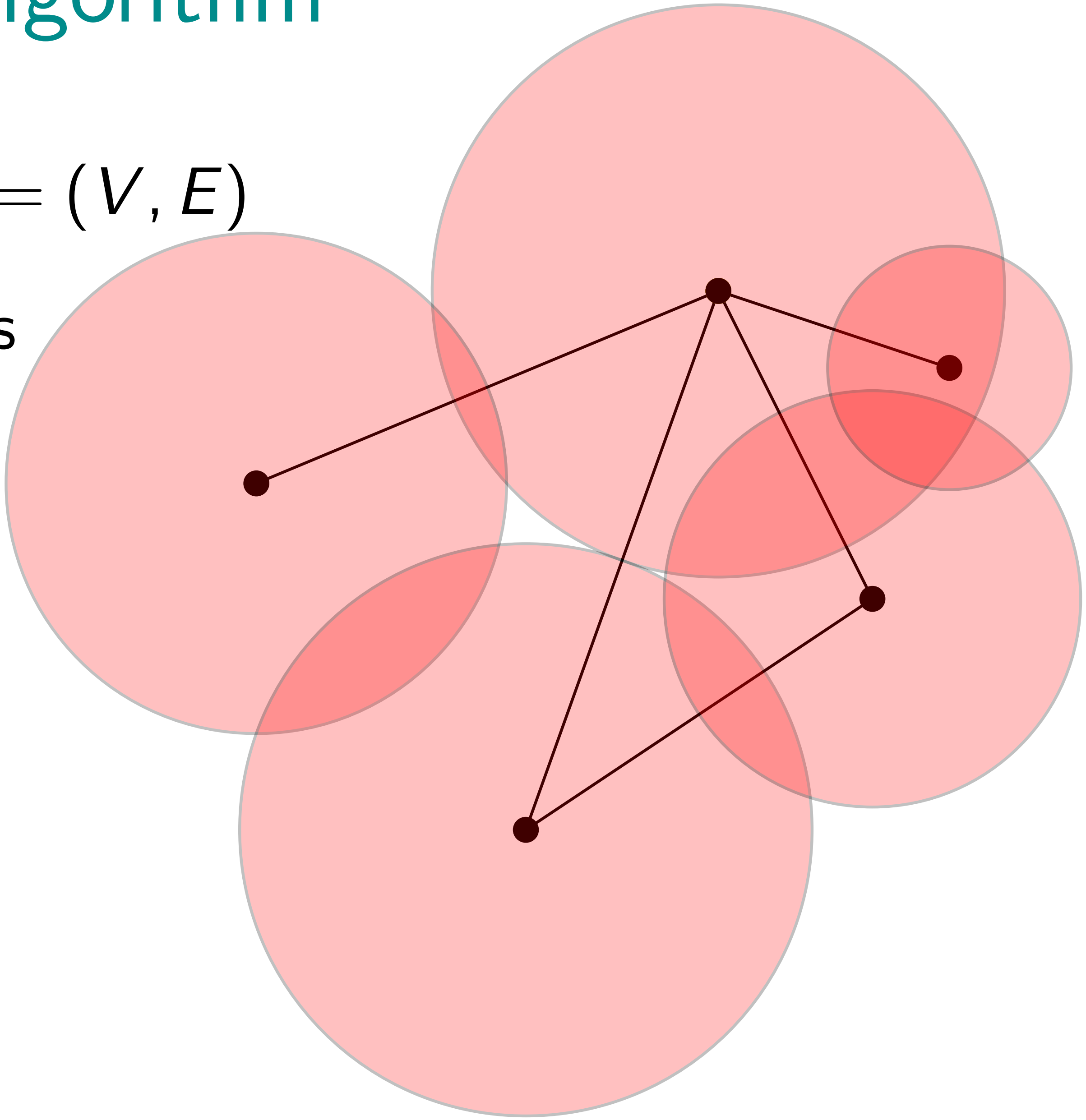
Input: drawing of $G = (V, E)$



Plane-Sweep-Algorithm

Input: drawing of $G = (V, E)$

compute all ply-disks

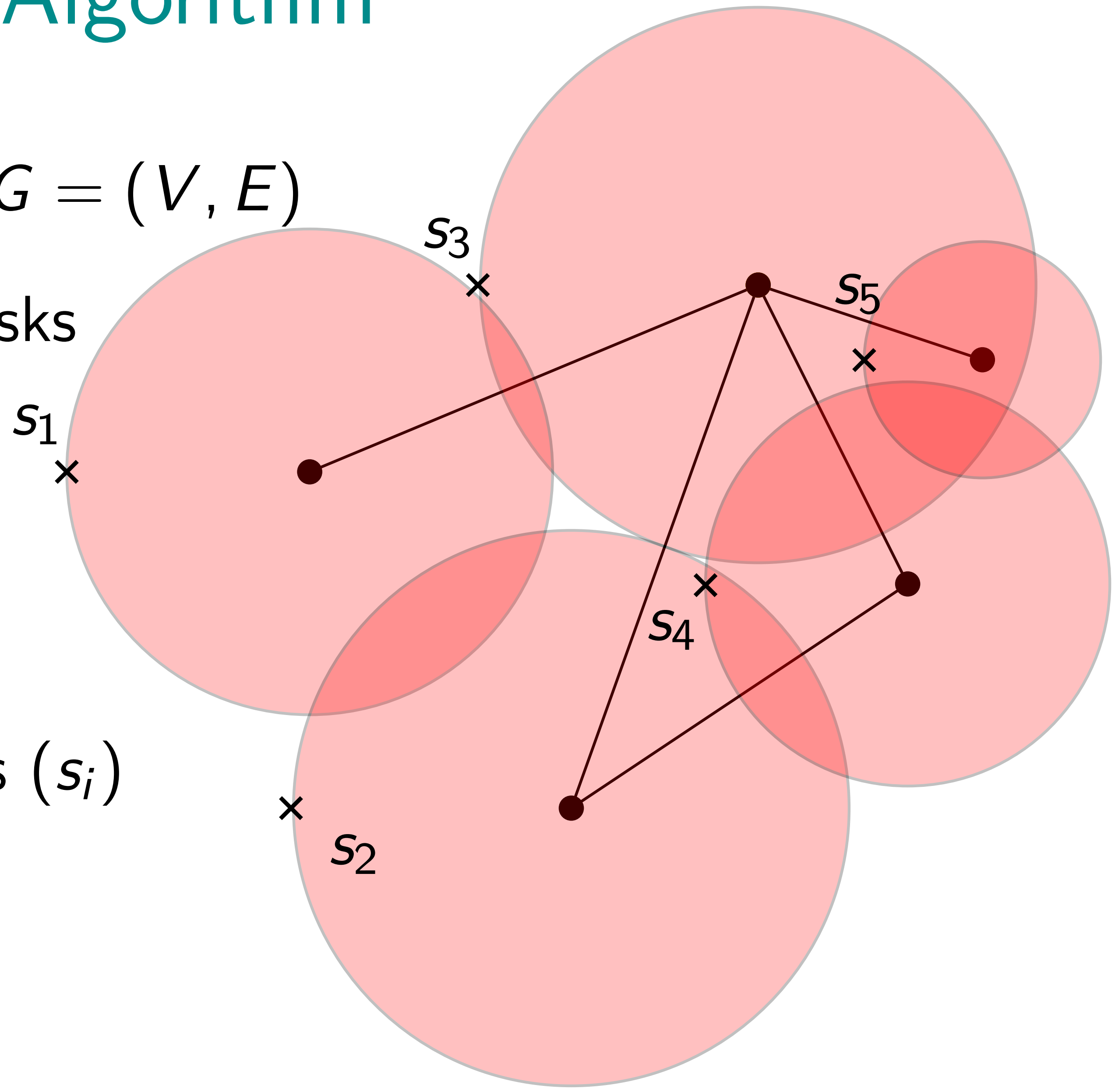


Plane-Sweep-Algorithm

Input: drawing of $G = (V, E)$

compute all ply-disks

create start-events (s_i)

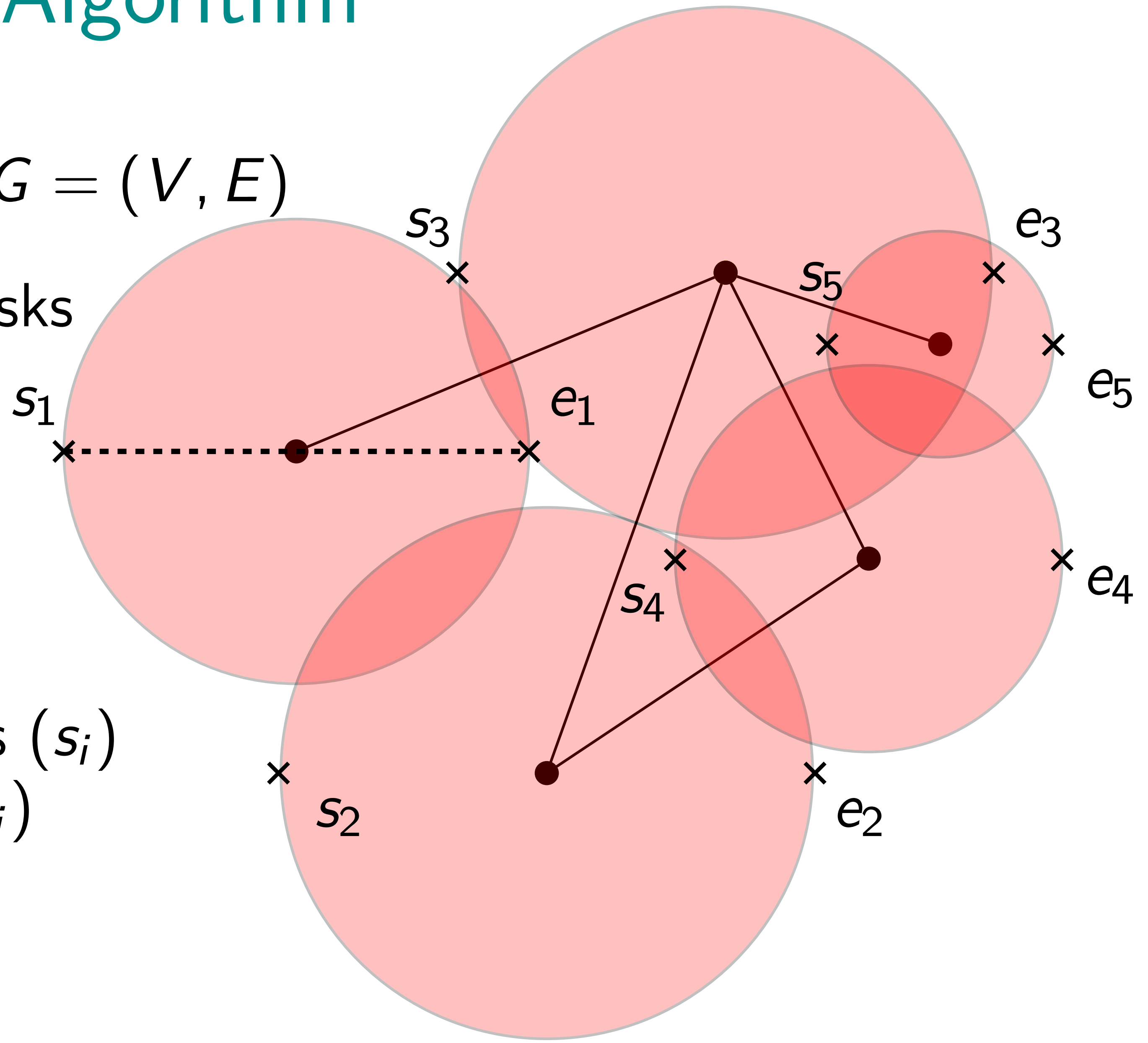


Plane-Sweep-Algorithm

Input: drawing of $G = (V, E)$

compute all ply-disks

create start-events (s_i) and end-events (e_i)



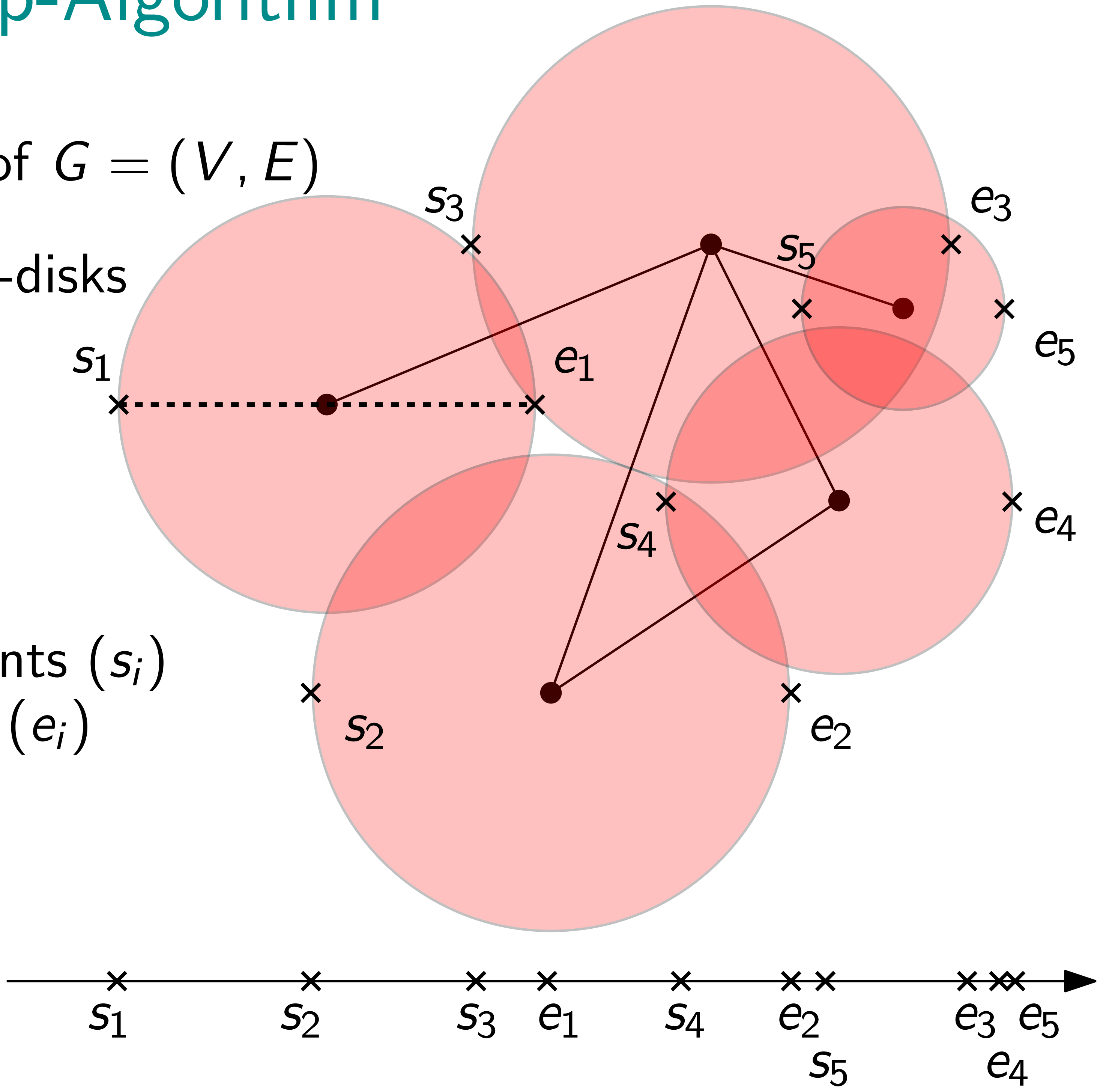
Plane-Sweep-Algorithm

Input: drawing of $G = (V, E)$

compute all ply-disks

create start-events (s_i)
and end-events (e_i)

priority queue



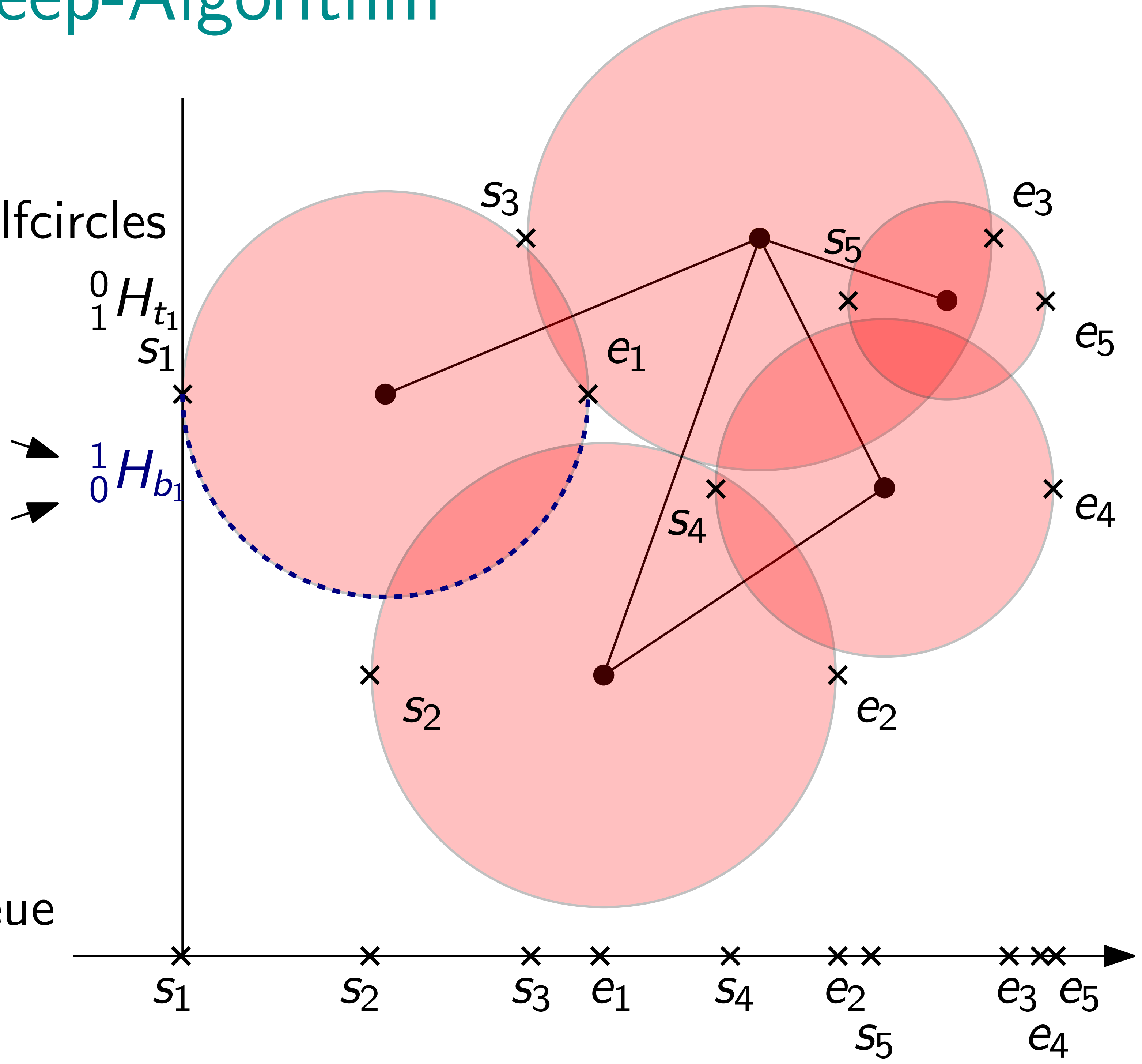
Plane-Sweep-Algorithm

start-event:
introduce halfcircles

ply above

ply below

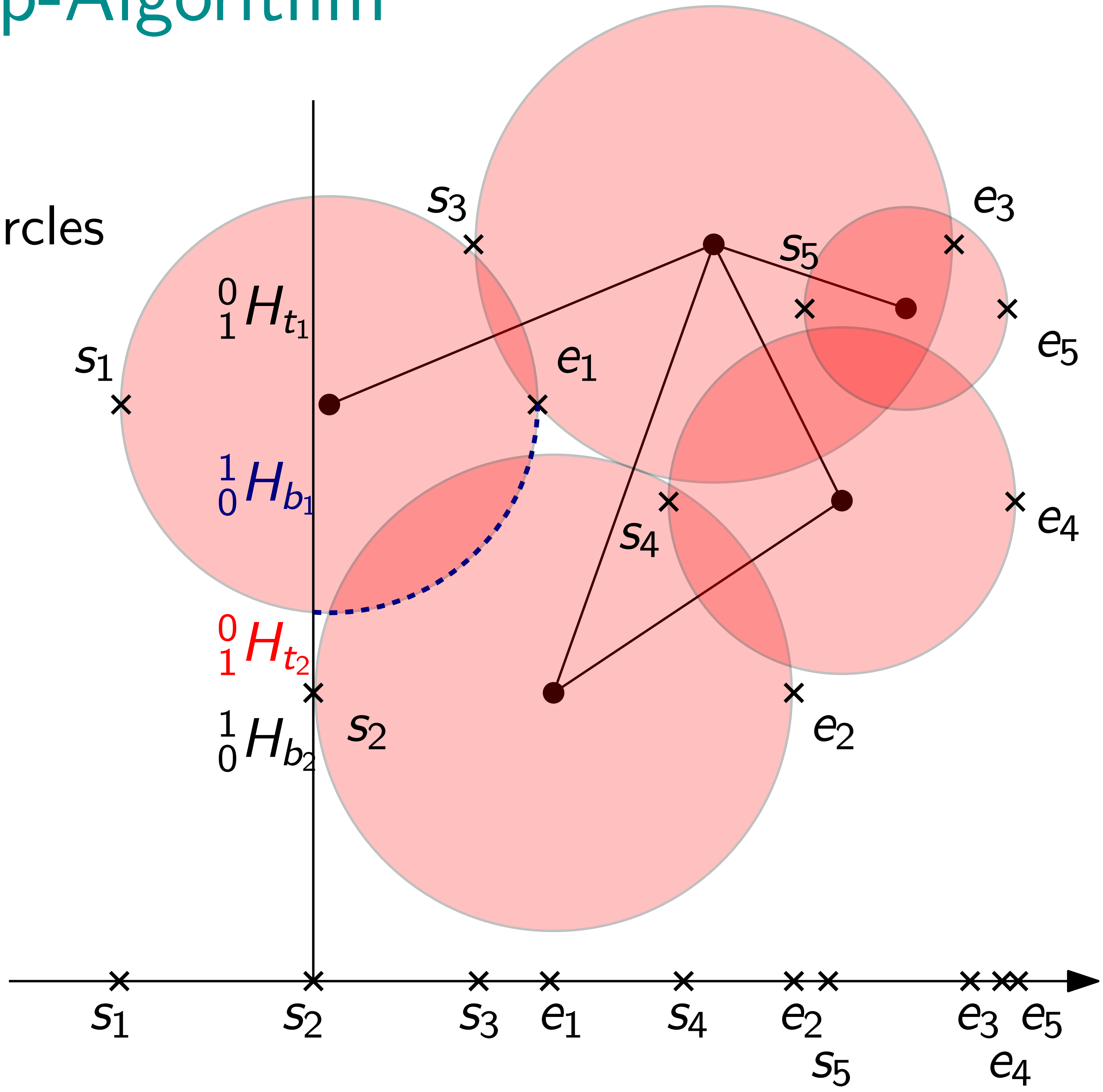
priority queue



Plane-Sweep-Algorithm

start-event:
introduce halfcircles

priority queue

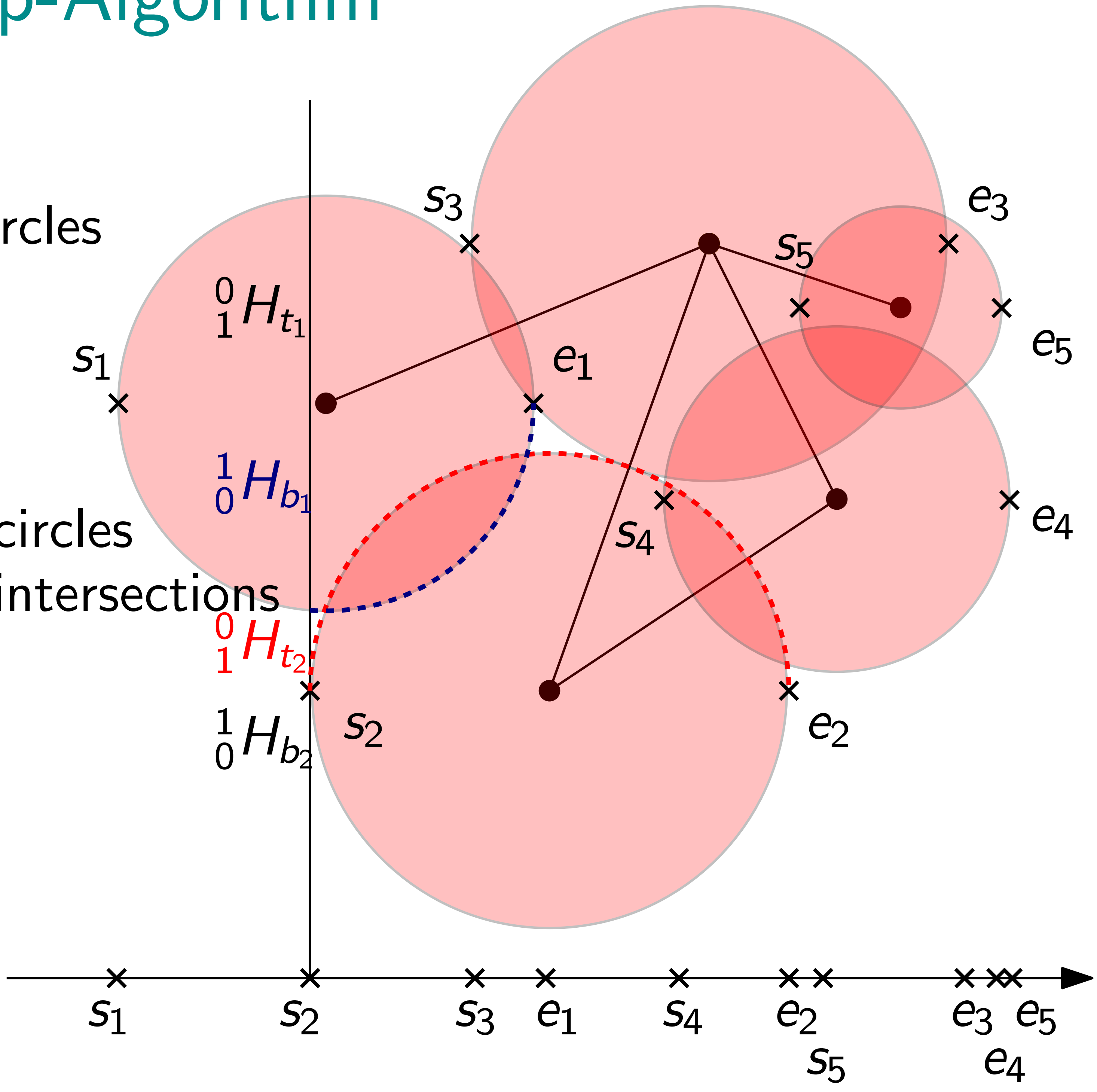


Plane-Sweep-Algorithm

start-event:
introduce halfcircles

consecutive halfcircles
are checked for intersections

priority queue

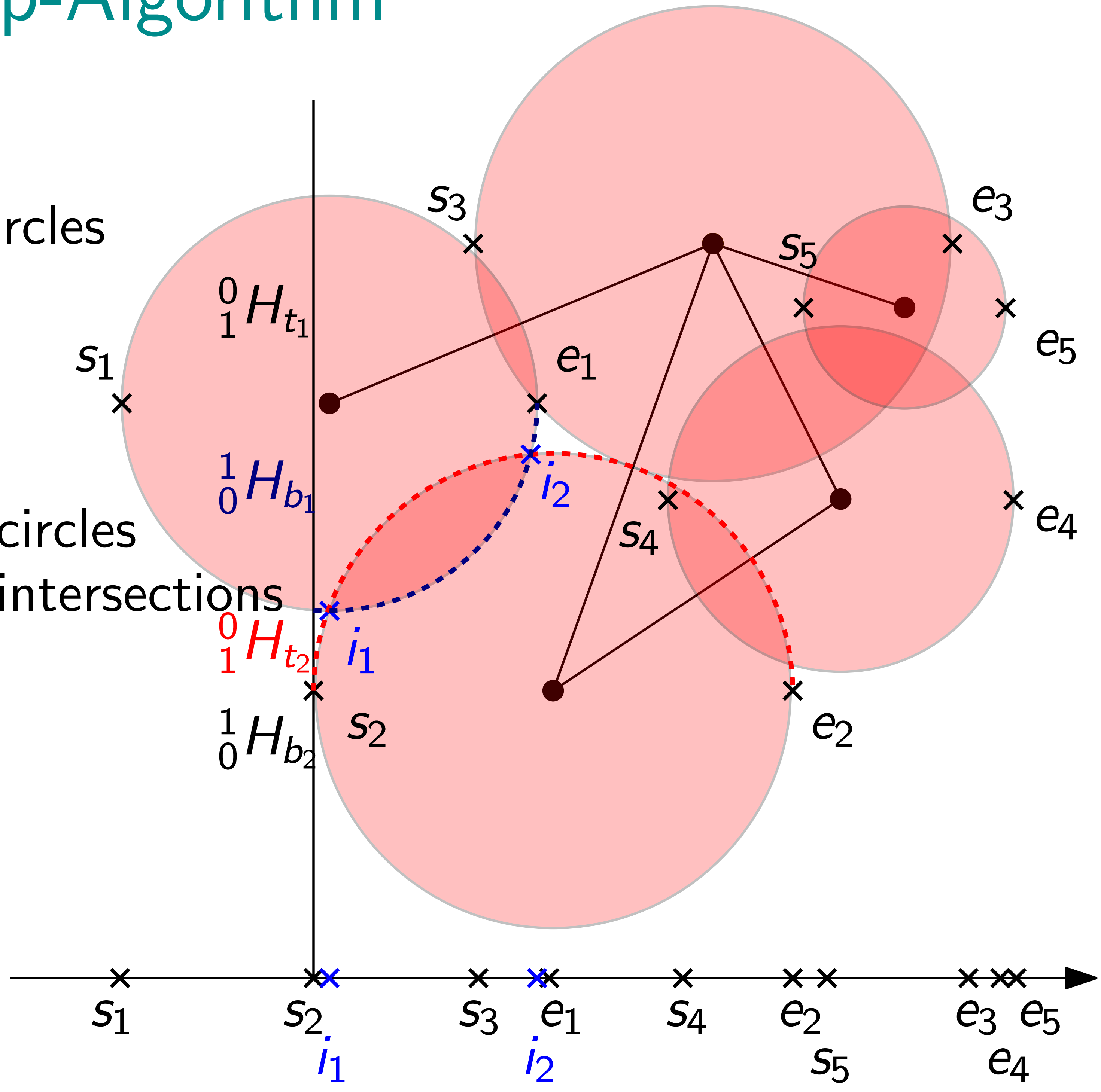


Plane-Sweep-Algorithm

start-event:
introduce halfcircles

consecutive halfcircles
are checked for intersections

priority queue

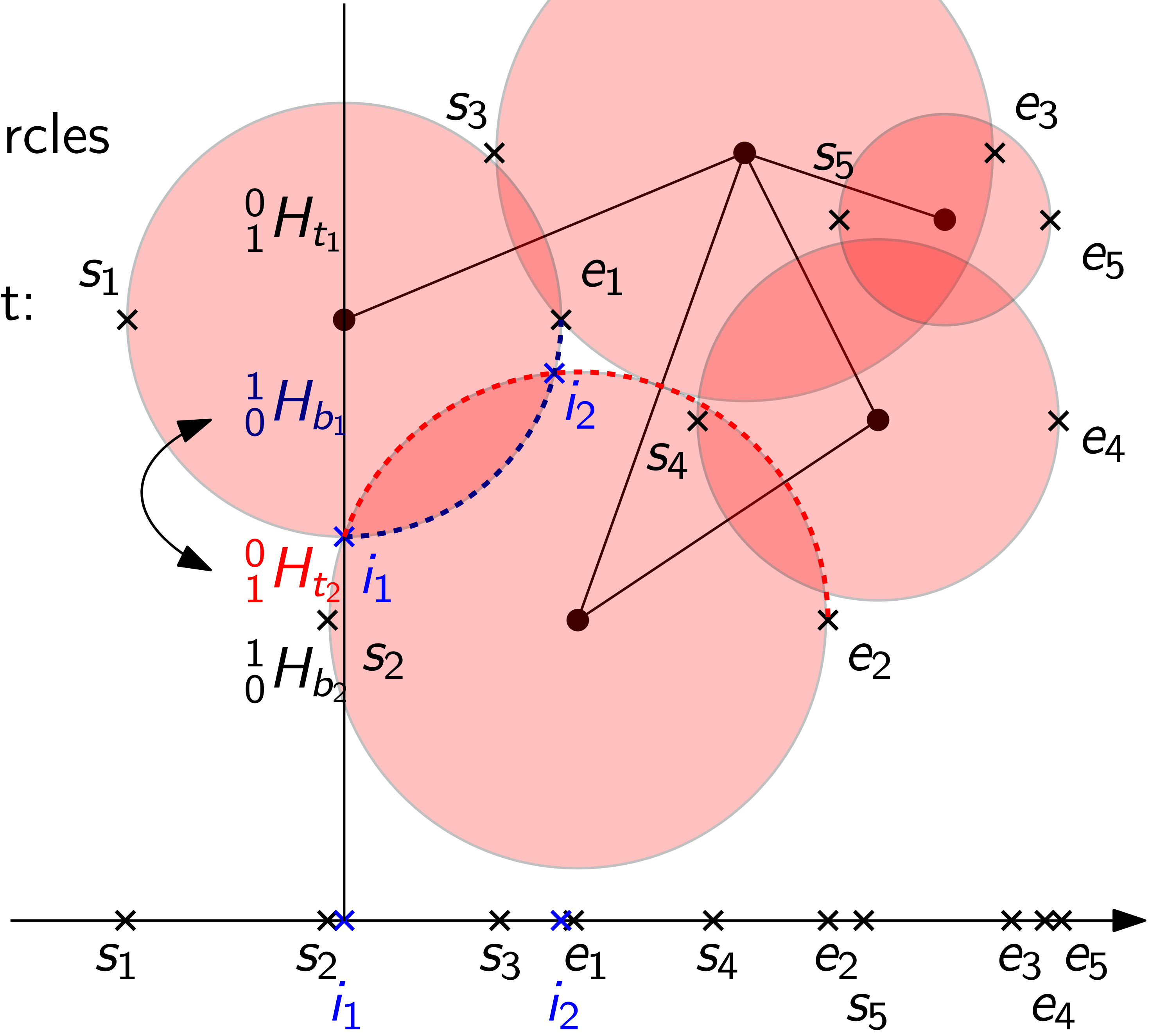


Plane-Sweep-Algorithm

start-event:
introduce halfcircles

intersection-event:
swap halfcircles
update ply

priority queue

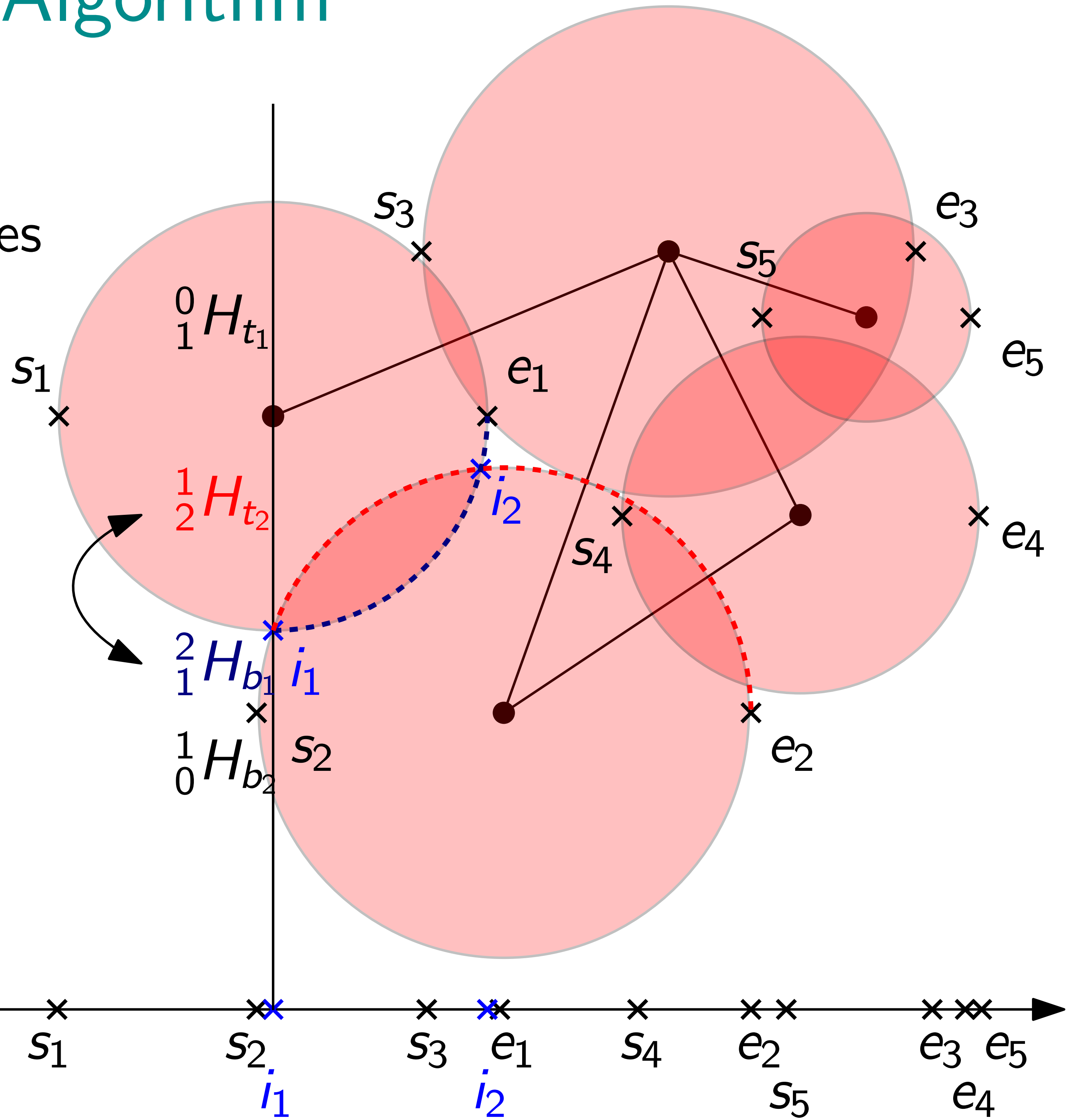


Plane-Sweep-Algorithm

start-event:
introduce halfcircles

intersection-event:
swap halfcircles
update ply

priority queue



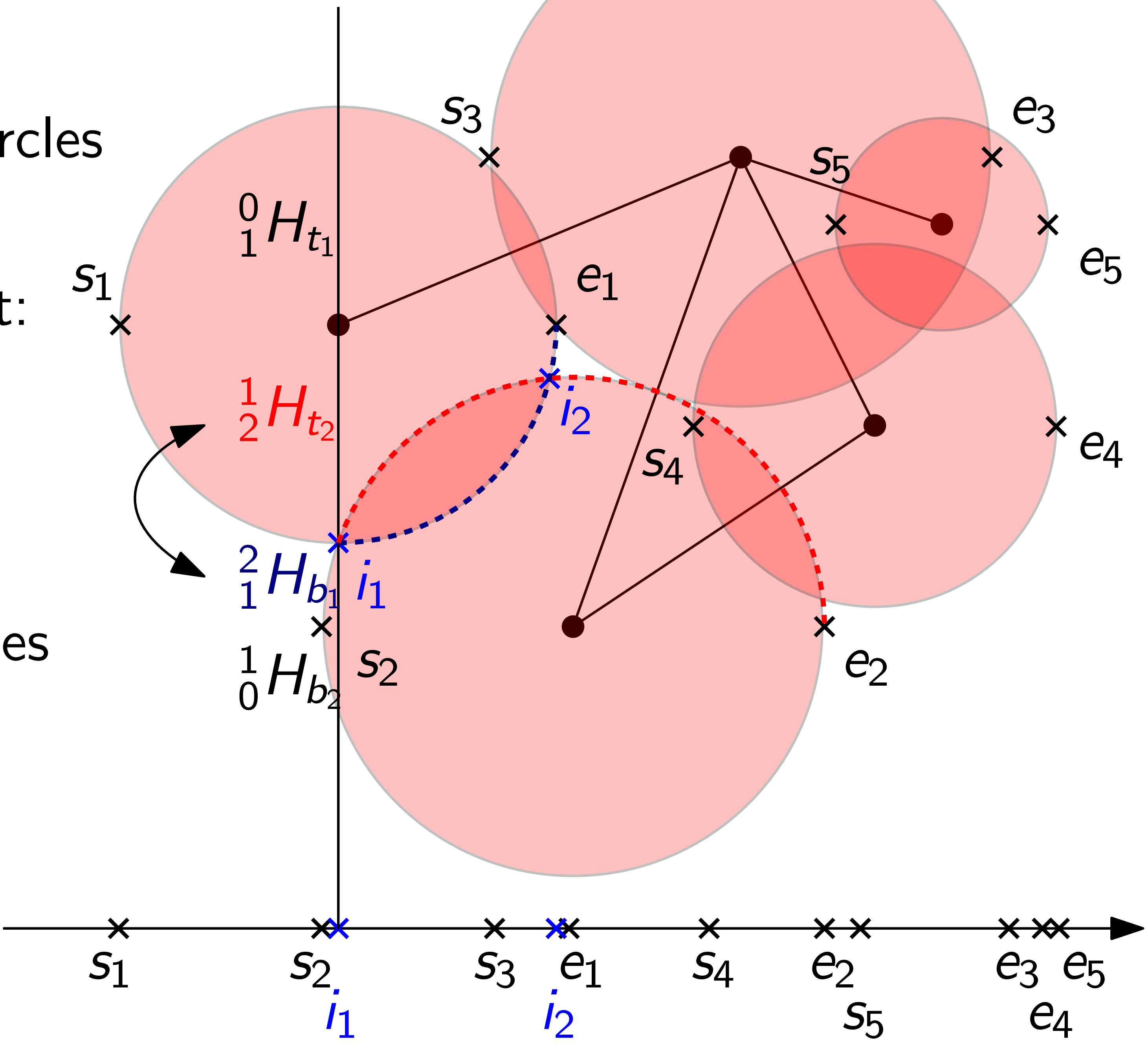
Plane-Sweep-Algorithm

start-event:
  introduce halfcircles

intersection-event:
  swap halfcircles
  update ply

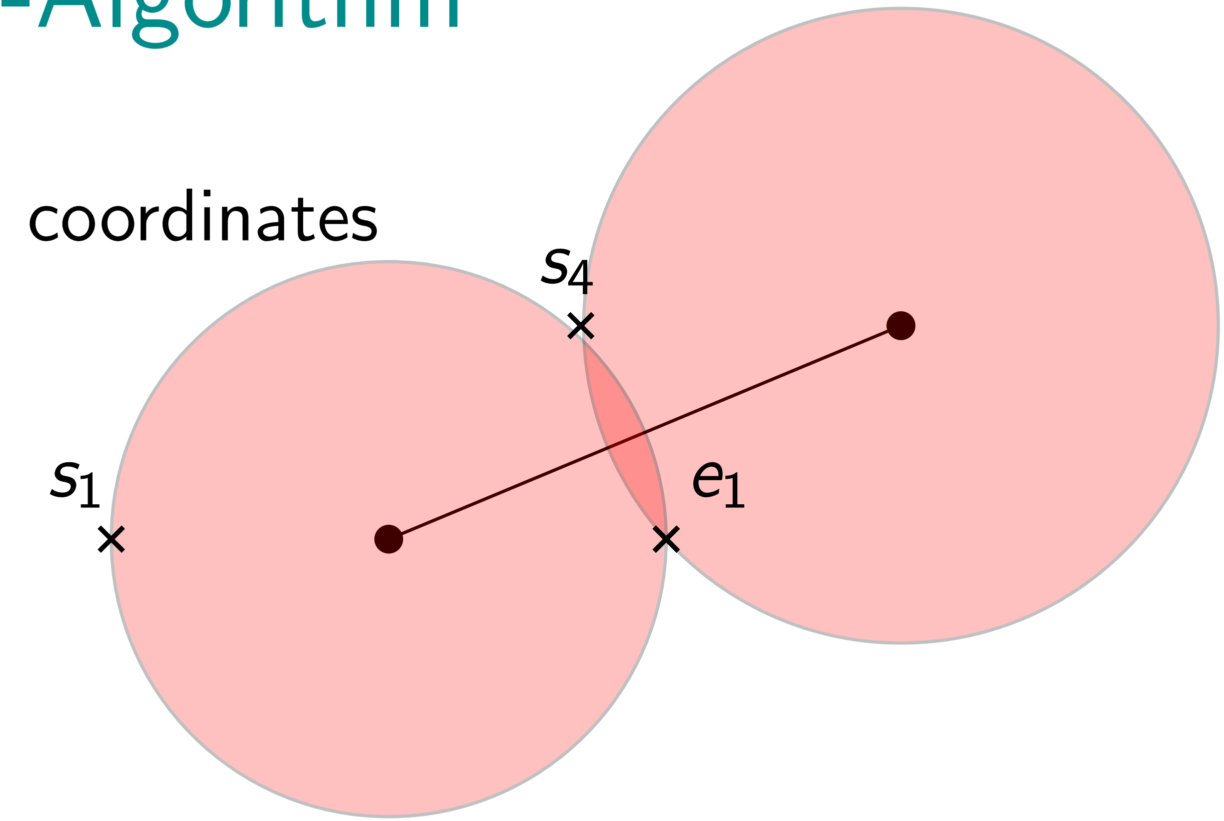
end-event:
  remove halfcircles

priority queue



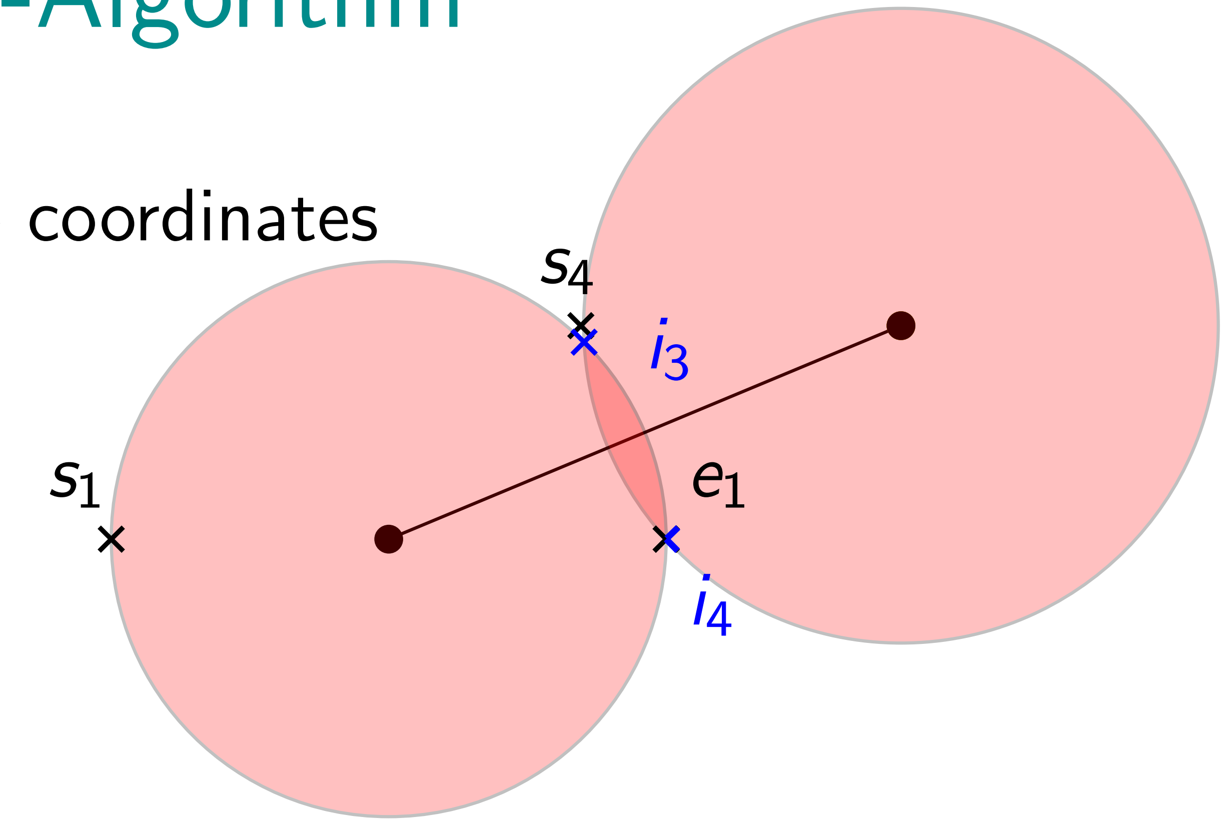
Plane-Sweep-Algorithm

events can share coordinates



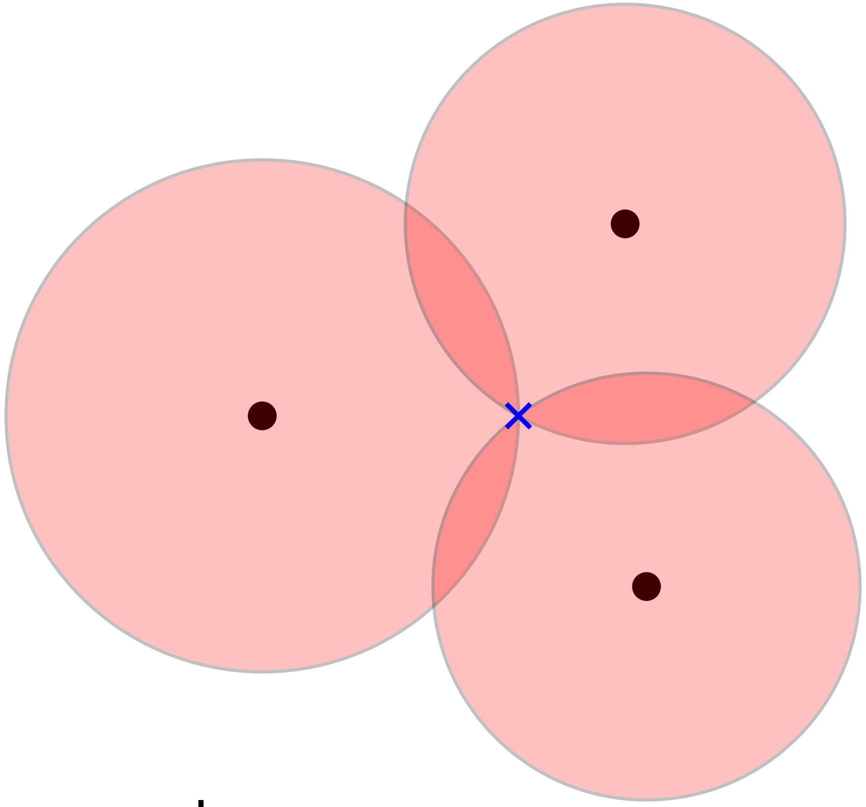
Plane-Sweep-Algorithm

events can share coordinates



Plane-Sweep-Algorithm

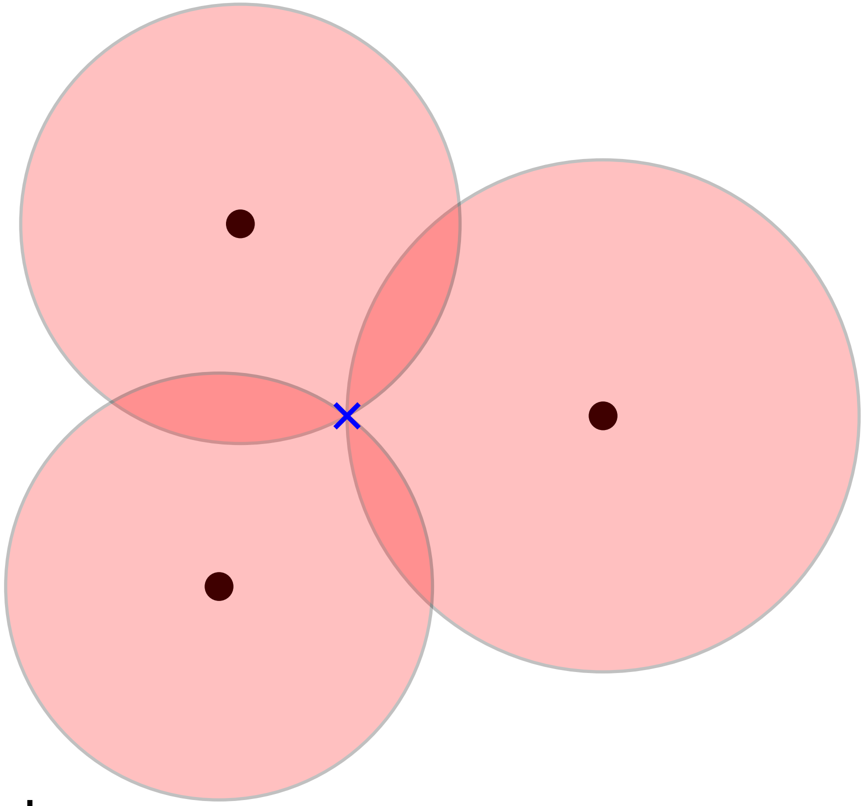
events can share coordinates



end-event before intersection-event

Plane-Sweep-Algorithm

events can share coordinates

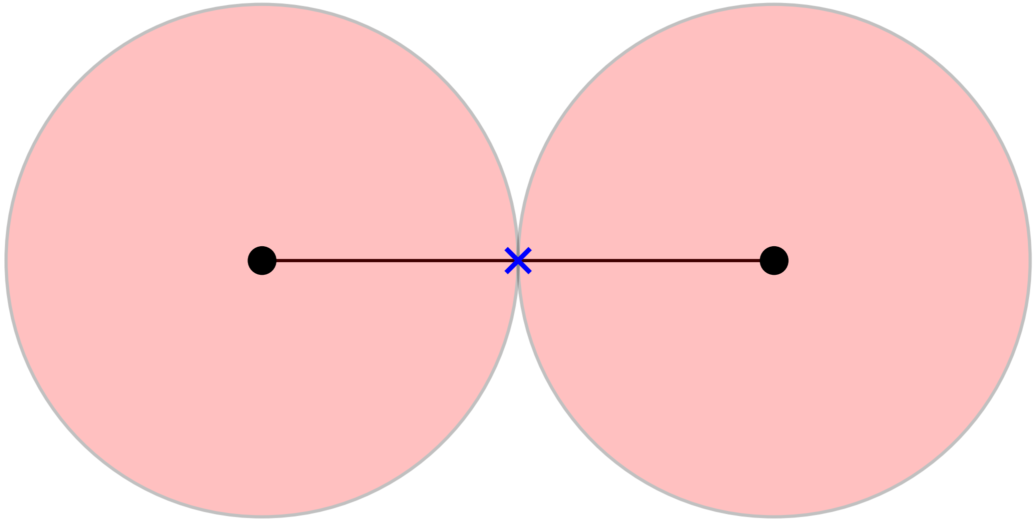


end-event before intersection-event

intersection-event before start-event

Plane-Sweep-Algorithm

events can share coordinates



end-event before intersection-event

intersection-event before start-event

end-event before start-event

Difficulties

previous definition ply-1 drawing:

a) for any vertex v all edges have uniform length ℓ

b) the distance between any two vertices is at least ℓ

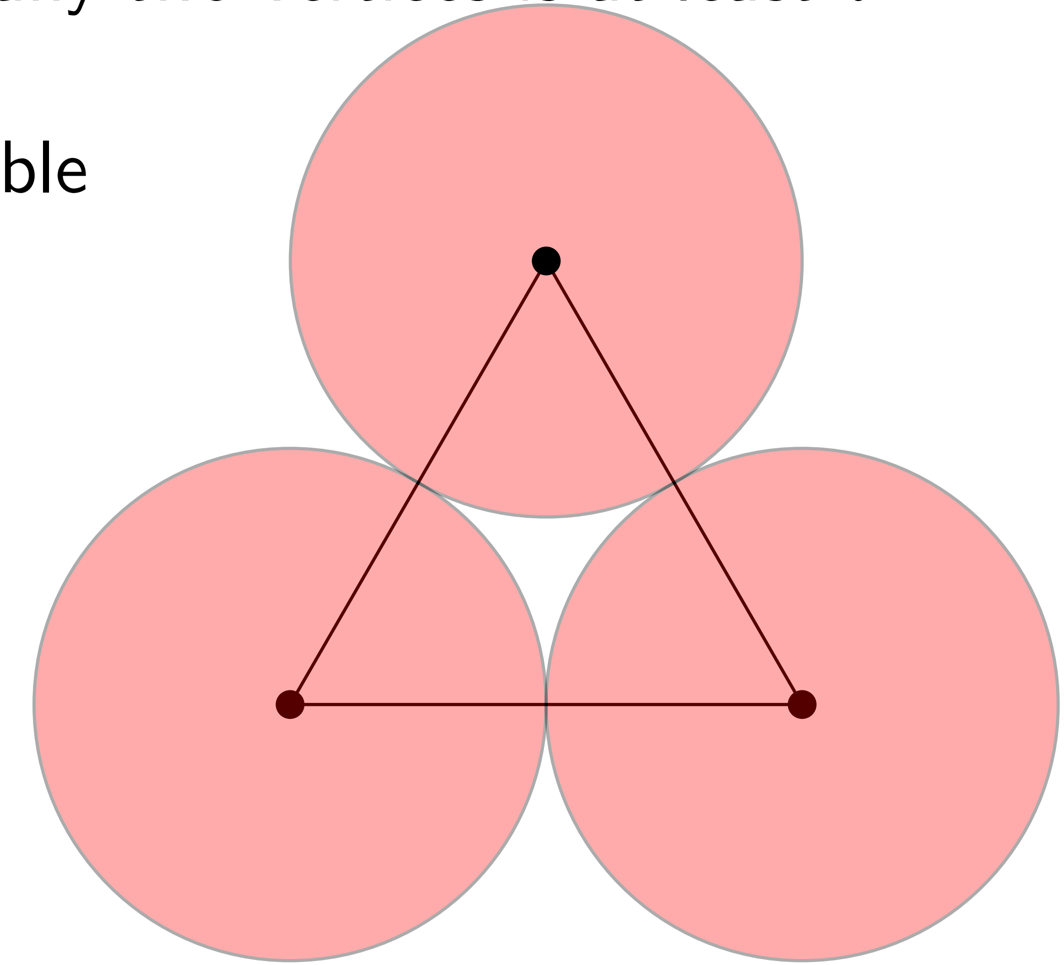
Difficulties

previous definition ply-1 drawing:

a) for any vertex v all edges have uniform length ℓ

b) the distance between any two vertices is at least ℓ

K_3 is empty ply drawable
as equilateral triangle



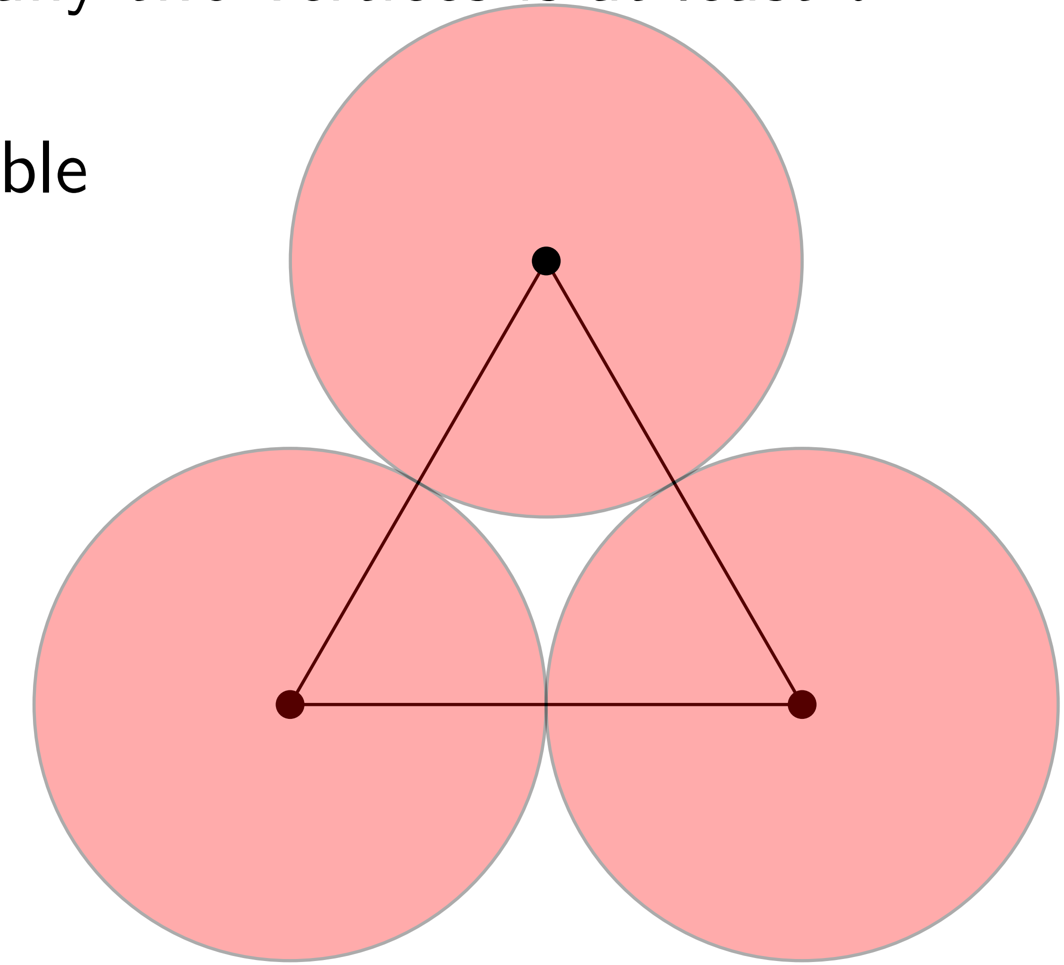
Difficulties

previous definition ply-1 drawing:

a) for any vertex v all edges have uniform length ℓ

b) the distance between any two vertices is at least ℓ

K_3 is empty ply drawable as equilateral triangle



but: need of irrational coordinates (lengths)

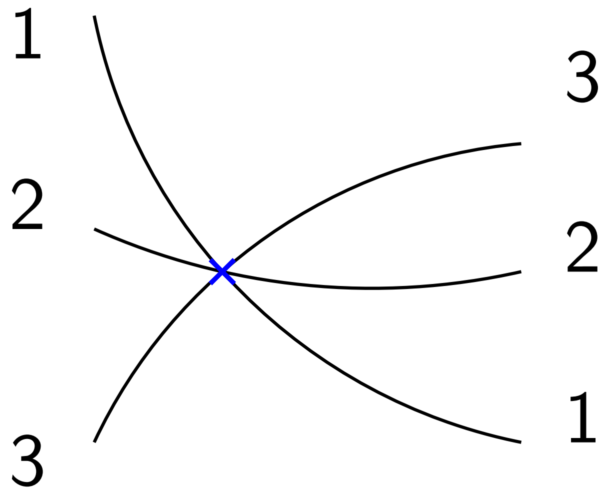
in fact we compute ply 2

Inconsistencies

induce a strange order of events

order of intersection-events

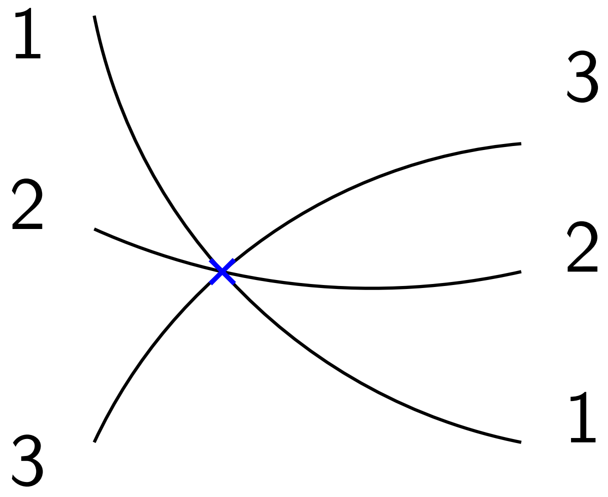
rounding coordinates



Inconsistencies

induce a strange order of events

- order of intersection-events
- rounding coordinates



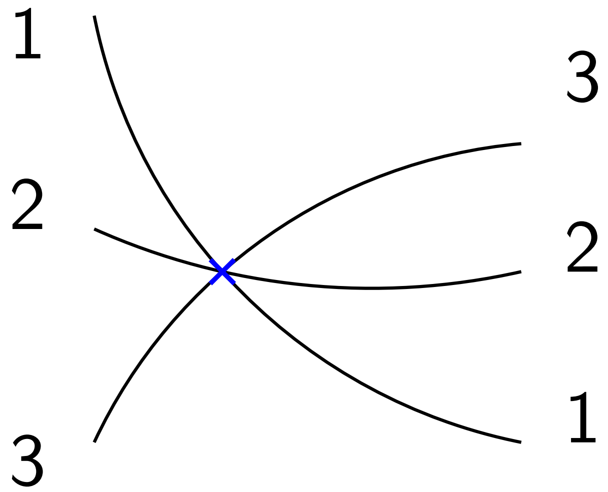
if we encounter `intersect (`H_1`, `H_3`)`

but there is at least one other halfcircle H_2 in between

Inconsistencies

induce a strange order of events

- order of intersection-events
- rounding coordinates



if we encounter `intersect` (H_1, H_3)

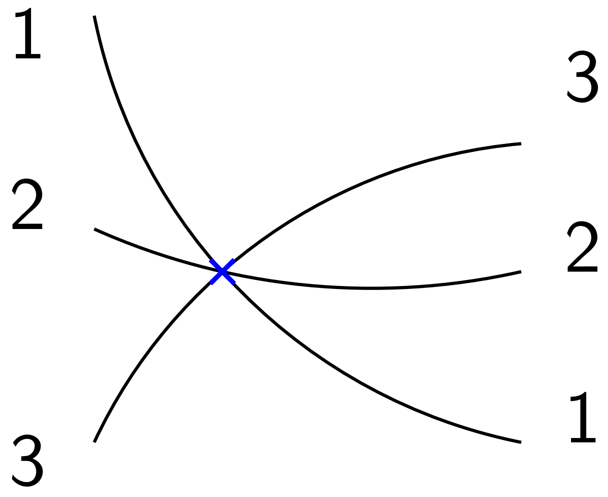
but there is at least one other halfcircle H_2 in between

- event is not applicable

Inconsistencies

induce a strange order of events

- order of intersection-events
- rounding coordinates



if we encounter `intersect` (H_1, H_3)

but there is at least one other halfcircle H_2 in between

- event is not applicable

due to consistency there must exist

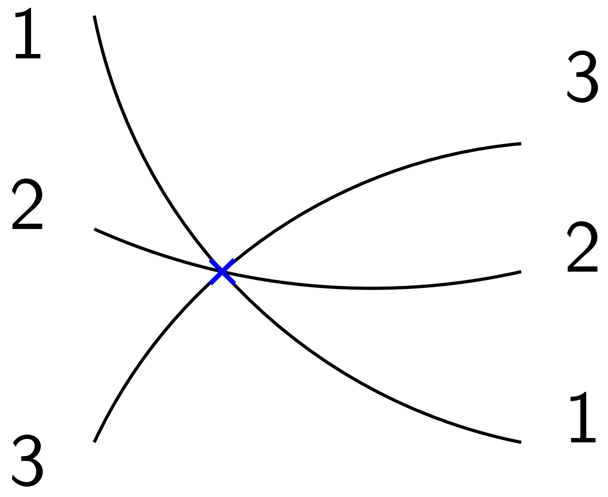
- `intersect`(H_1, H_2) or `intersect`(H_3, H_2)

Inconsistencies

postpone event

until there is a applicable event

try event again

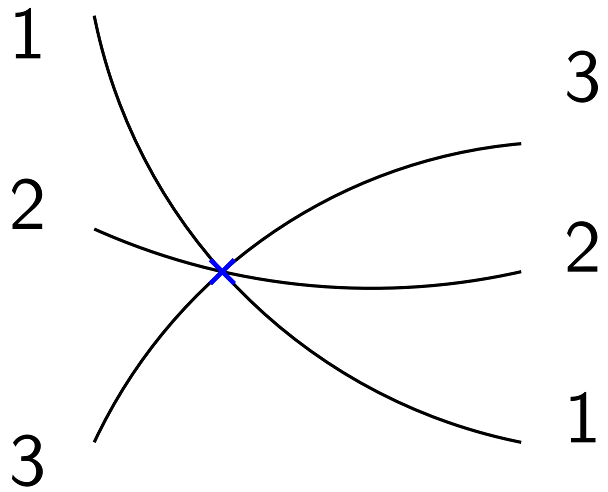


Inconsistencies

postpone event

until there is a applicable event

try event again



[De Luca et al - WALCOM'17]

introduce "fake" events to check for consistency

increase the precision of the computation using `ApFloat`

Speedup

comparison between the two implementations:

			[De Luca et al]			Our Tool	
Density	Vertices	Ply	Events	Time(ms)	Events	Time(ms)	postponed
1.5	250						
	300						
	350						
	400						
	450						
2.5	250						
	300						
	350						
	400						
	450						

100 graphs in total, 10 in each set, by [De Luca - WALCOM'17]

average values

Speedup

comparison between the two implementations:

			[De Luca et al]		Our Tool		
Density	Vertices	Ply	Events	Time(ms)	Events	Time(ms)	postponed
1.5	250	18					
	300	19.8					
	350	23.7					
	400	25.4					
	450	28 (27.9)					
2.5	250	38.1					
	300	45.4					
	350	51.4					
	400	59.3 (58.7)					
	450	64.3 (64.2)					

100 graphs in total, 10 in each set, by [De Luca - WALCOM'17]

average values

Speedup

comparison between the two implementations:

			[De Luca et al]		Our Tool		
Density	Vertices	Ply	Events	Time(ms)	Events	Time(ms)	postponed
1.5	250	18	18334		6430		
	300	19.8	25688.3		8950.8		
	350	23.7	34140.5		11949.7		
	400	25.4	43543.4		15227.5		
	450	28 (27.9)	55643.2		19395.6		
2.5	250	38.1	47248		16539.1		
	300	45.4	68070.5		23892.3		
	350	51.4	90943.4		31850.1		
	400	59.3 (58.7)	118188.7		40606.9		
	450	64.3 (64.2)	148973.3		51640.4		

100 graphs in total, 10 in each set, by [De Luca - WALCOM'17]

average values

Speedup

comparison between the two implementations:

			[De Luca et al]		**Our Tool**		
Density	Vertices	Ply	Events	Time(ms)	Events	Time(ms)	postponed
1.5	250	18	18334	23955	6430	10.2	–
	300	19.8	25688.3	38454.8	8950.8	9.7	
	350	23.7	34140.5	52829.1	11949.7	23.7	
	400	25.4	43543.4	72928.5	15227.5	16.4	
	450	28 (27.9)	55643.2	100653.2	19395.6	23.1	
2.5	250	38.1	47248	92192.7	16539.1	19.5	
	300	45.4	68070.5	147113.6	23892.3	36.7	
	350	51.4	90943.4	217999.9	31850.1	43.5	
	400	59.3 (58.7)	118188.7	309601.8	40606.9	83.8	
	450	64.3 (64.2)	148973.3	426993.5	51640.4	112.9	

100 graphs in total, 10 in each set, by [De Luca - WALCOM'17]

average values

Speedup

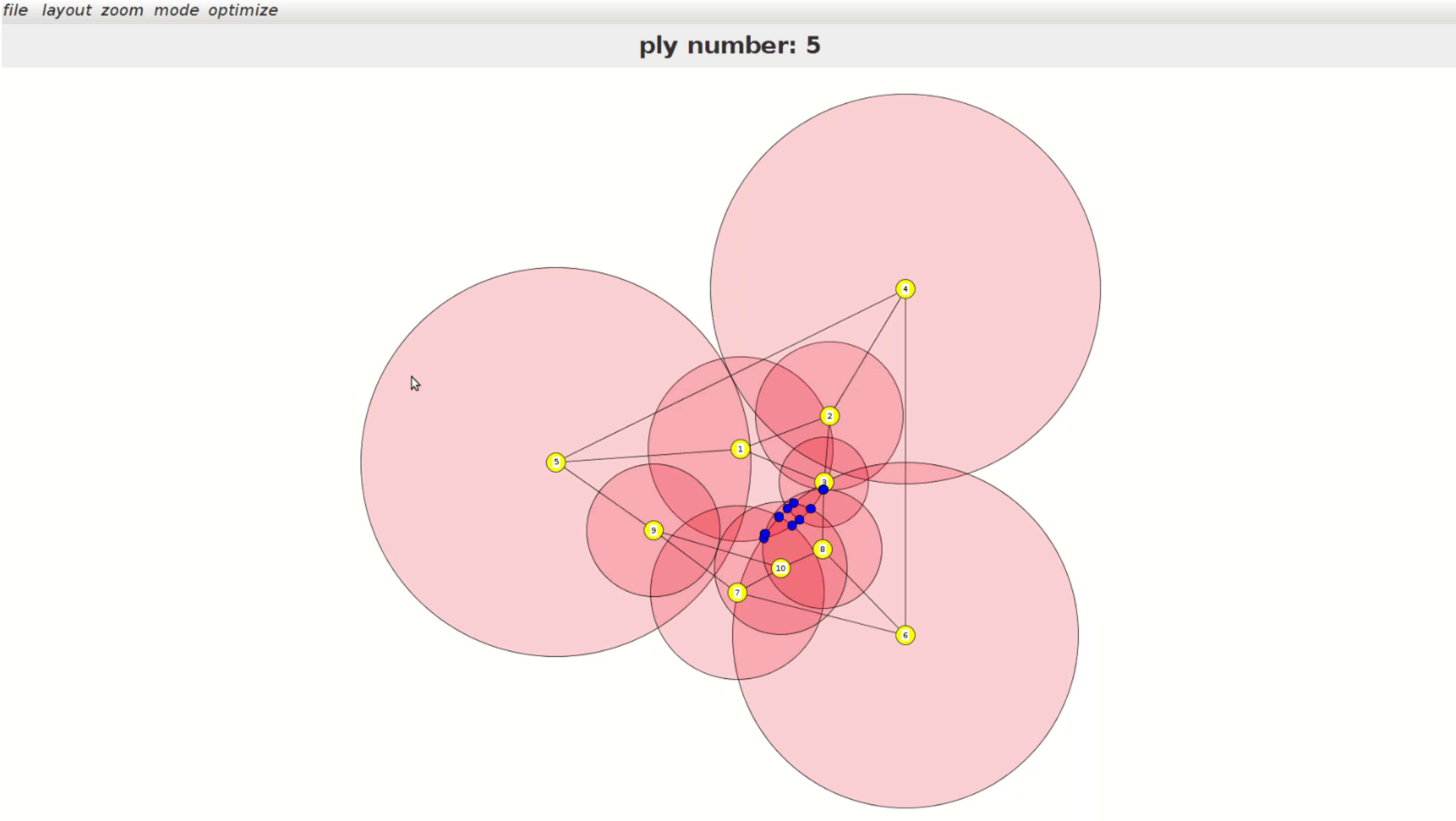
comparison between the two implementations:

			[De Luca et al]		**Our Tool**		
Density	Vertices	Ply	Events	Time(ms)	Events	Time(ms)	postponed
1.5	250	18	18334	23955	6430	10.2	0.4
	300	19.8	25688.3	38454.8	8950.8	9.7	159
	350	23.7	34140.5	52829.1	11949.7	23.7	0.6
	400	25.4	43543.4	72928.5	15227.5	16.4	0.3
	450	28 (27.9)	55643.2	100653.2	19395.6	23.1	0.7
2.5	250	38.1	47248	92192.7	16539.1	19.5	0.5
	300	45.4	68070.5	147113.6	23892.3	36.7	2.2
	350	51.4	90943.4	217999.9	31850.1	43.5	2.5
	400	59.3 (58.7)	118188.7	309601.8	40606.9	83.8	40048.3
	450	64.3 (64.2)	148973.3	426993.5	51640.4	112.9	62776.7

100 graphs in total, 10 in each set, by [De Luca - WALCOM'17]

average values

Demonstration



ply number: 5

Improvement

we can automatically improve the ply number of drawings

best strategy in [De Luca et al.- WALCOM'17]

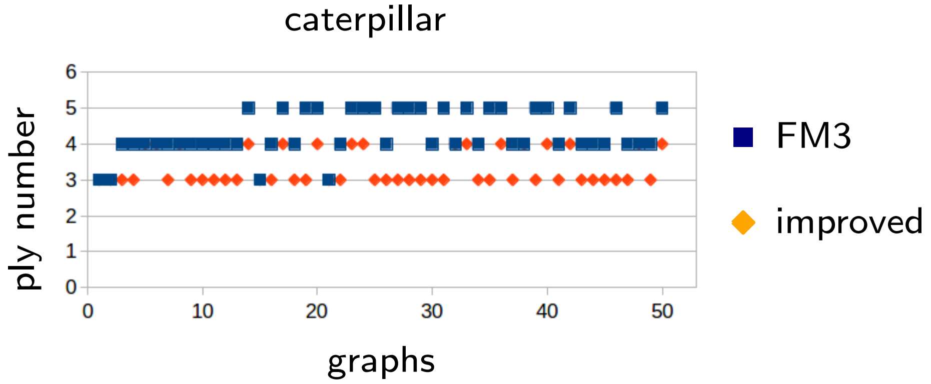
fast multipole multilevel layout (**FM3**)

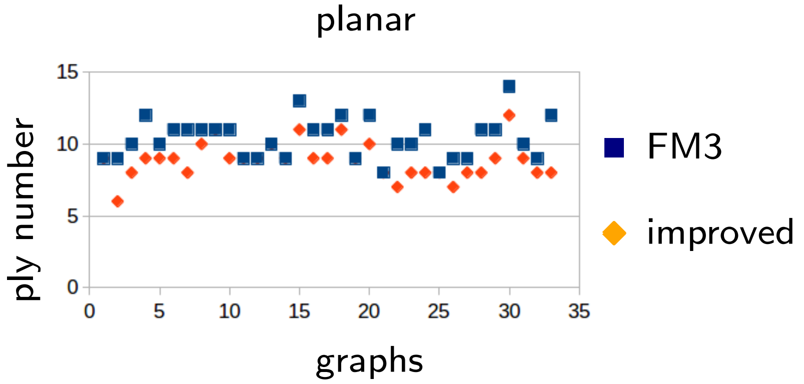
Improvement

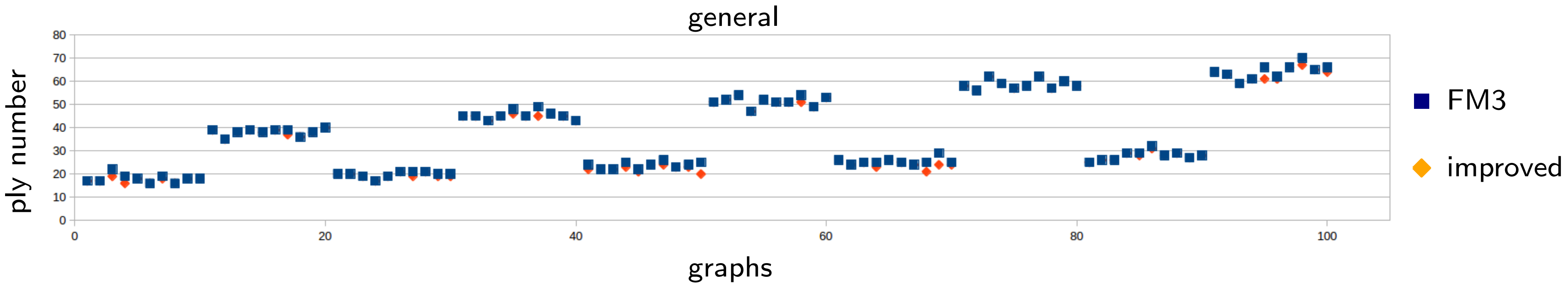
we can automatically improve the ply number of drawings

best strategy in [De Luca et al.- WALCOM'17]

fast multipole multilevel layout (**FM3**)







Future Directions

- area ply
    - small areas with high ply number
    - only consider large areas

Future Directions

area ply

small areas with high ply number

only consider large areas

3-regular planar graphs

admit a drawing where no vetex is contained in any other vertices ply-disk

Future Directions

area ply

small areas with high ply number

only consider large areas

3-regular planar graphs

admit a drawing where no vetex is contained
in any other vertices ply-disk
(definition of empty-ply)

Future Directions

area ply

small areas with high ply number

only consider large areas

3-regular planar graphs

admit a drawing where no vetex is contained in any other vertices ply-disk (definition of empty-ply)

Thank you!