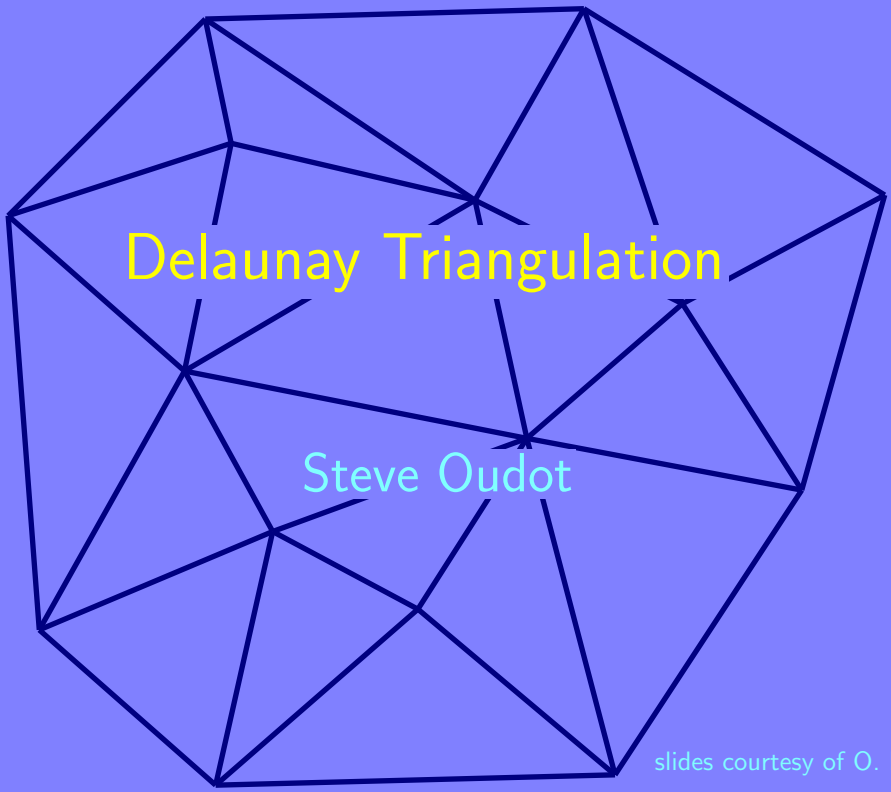
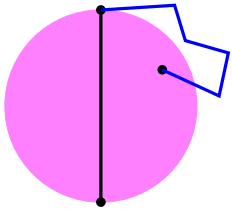
Delaunay Triangulation

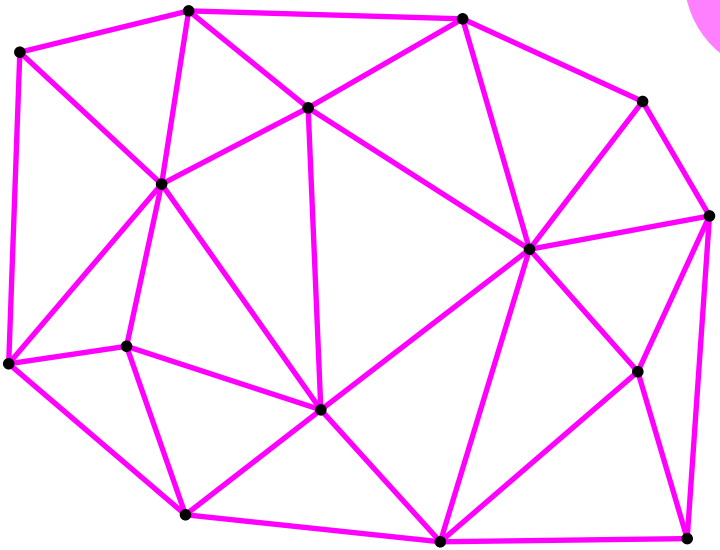
Steve Oudot



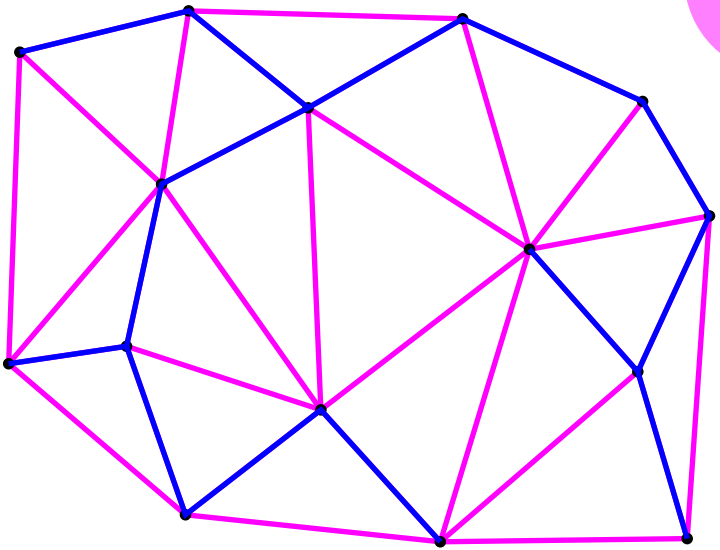
slides courtesy of O. Devillers

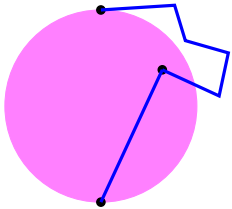
MST



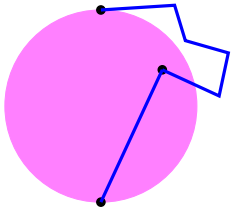


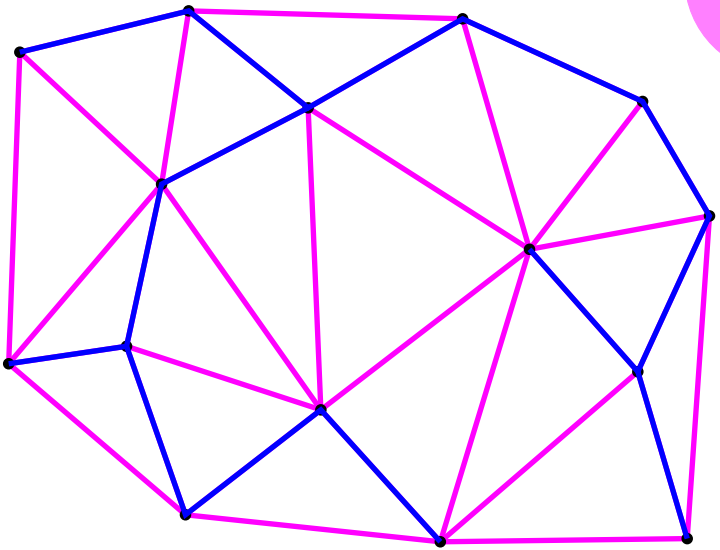
MST





MST

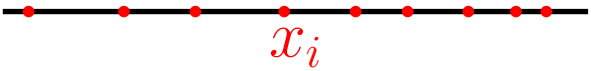




$\rightarrow$ use Kruskal's algorithm with Del as input $\rightarrow O(n \log n)$

Last: lower bound for Delaunay

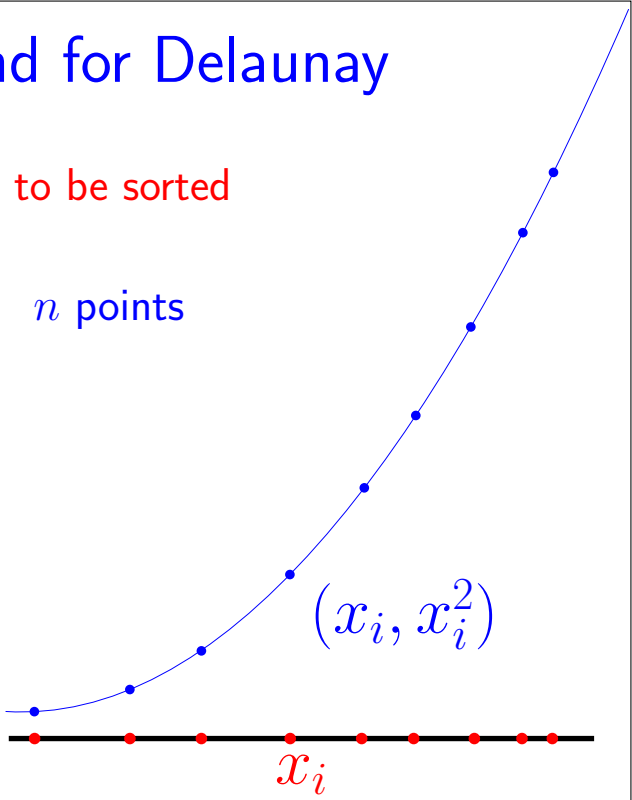
Let $x_1, x_2, \ldots, x_n \in \mathbb{R}$, to be sorted



Last: lower bound for Delaunay

Let $x_1, x_2, \ldots, x_n \in \mathbb{R}$, to be sorted

$(x_1, x_1^2), \ldots, (x_n, x_n^2)$ n points



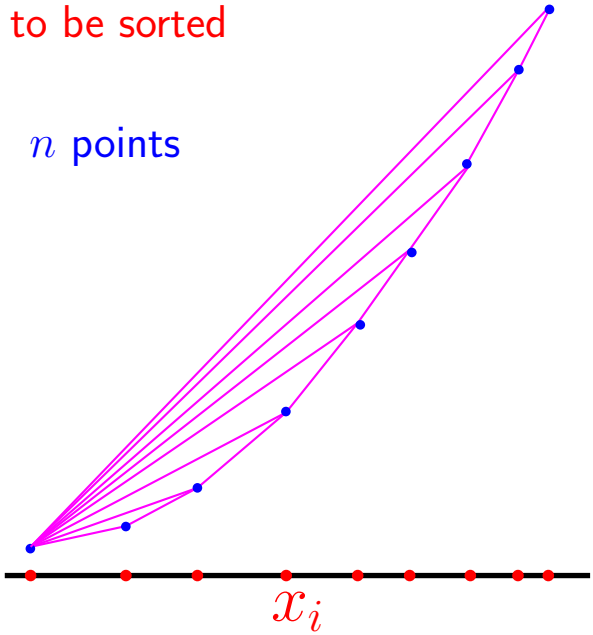
Last: lower bound for Delaunay

Let $x_1, x_2, \ldots, x_n \in \mathbb{R}$, to be sorted

$(x_1, x_1^2), \ldots, (x_n, x_n^2)$ n points

Delaunay

$\longrightarrow$ order in x



Last: lower bound for Delaunay

Let $x_1, x_2, \ldots, x_n \in \mathbb{R}$, to be sorted

$\downarrow O(n)$

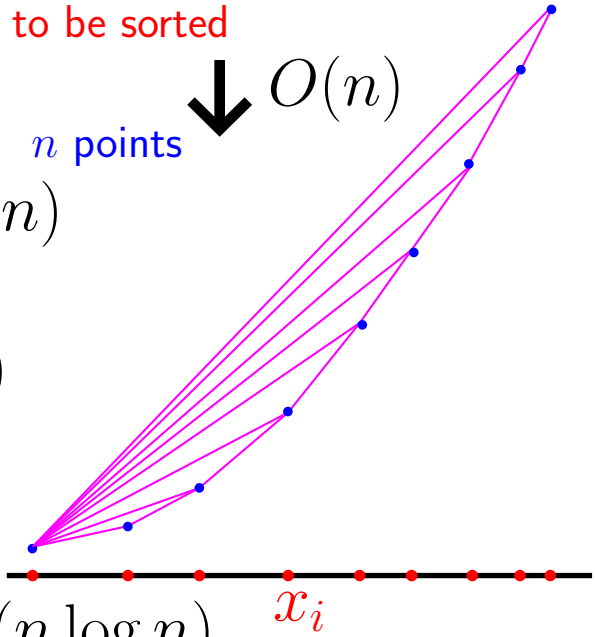
$(x_1, x_1^2), \ldots, (x_n, x_n^2)$ n points

$\downarrow f(n)$

Delaunay

$\downarrow O(n)$

$\longrightarrow$ order in x



$$O(n) + f(n) \in \Omega(n \log n)$$

Last: lower bound for Delaunay

$$\Omega(n \log n)$$

Optimal algorithm for computing Delaunay

Division – Fusion

L. J. Guibas and J. Stolfi. Primitives for the manipulation of general subdivisions and the computation of Voronoi diagrams. *ACM Trans. on Graphics*, 4(2):74–123, April 1985

Division-Fusion

Classical approach example: sort

Problem of size n

$\longrightarrow$ division into 2 pbs of size $O\left(n/2\right)$

$\longrightarrow$ recursive call on sub-problems

$\longrightarrow$ fusion

Division-Fusion

Classical approach

example: sort

Problem of size n

$\longrightarrow$ division into 2 pbs of size $O\left(n/2\right)$

$O(n)$

$\longrightarrow$ recursive call on sub-problems

$2\ f\left(\frac{n}{2}\right)$

$\longrightarrow$ fusion

$O(n)$

Division-Fusion

Classical approach

example: sort

Problem of size n

$$f(n) = O(n) + 2f\left(\frac{n}{2}\right) = O(n \log n)$$

$\longrightarrow$ division into 2 pbs of size $O\left(n/2\right)$

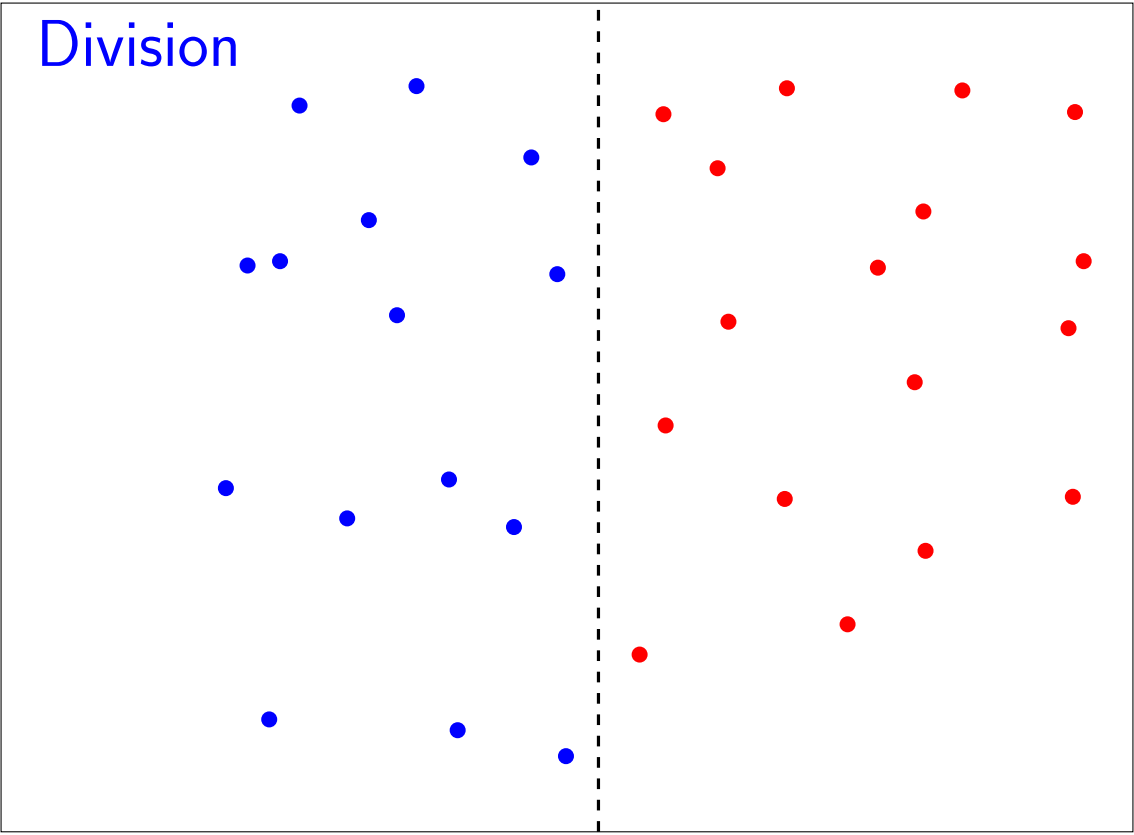
$O(n)$

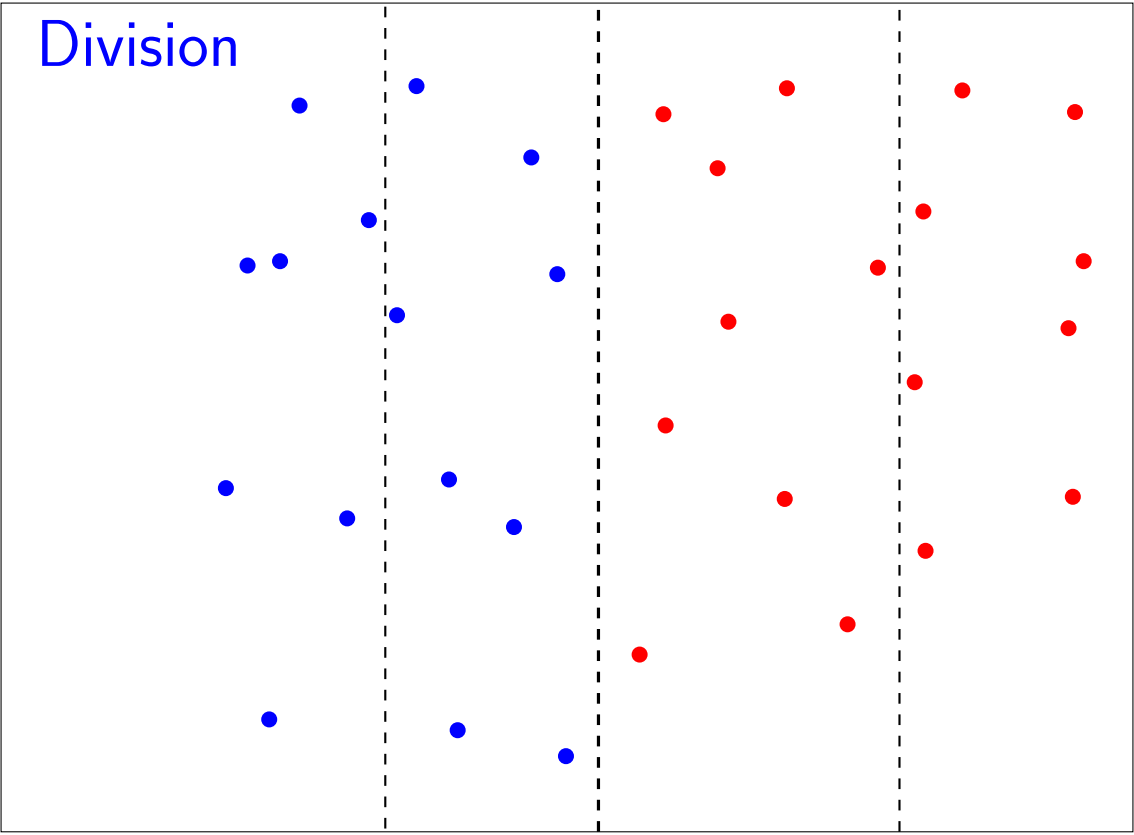
$\longrightarrow$ recursive call on sub-problems

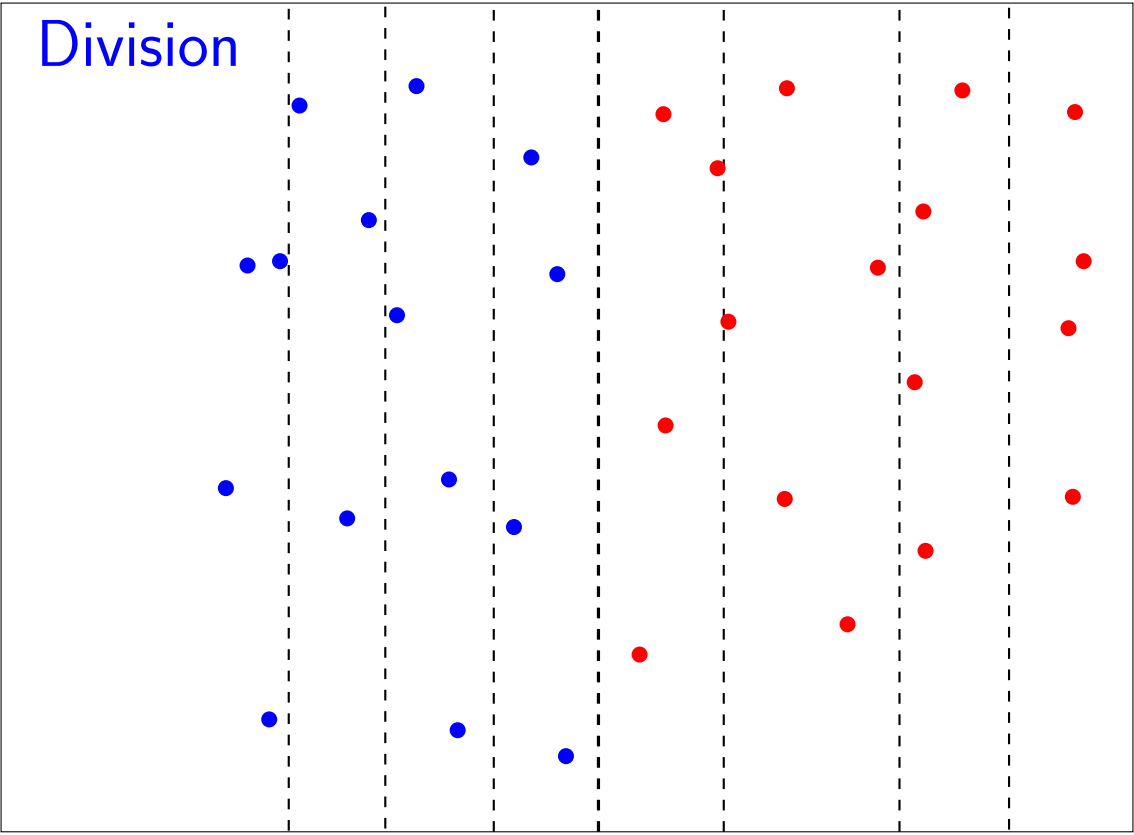
$2\ f\left(\frac{n}{2}\right)$

$\longrightarrow$ fusion

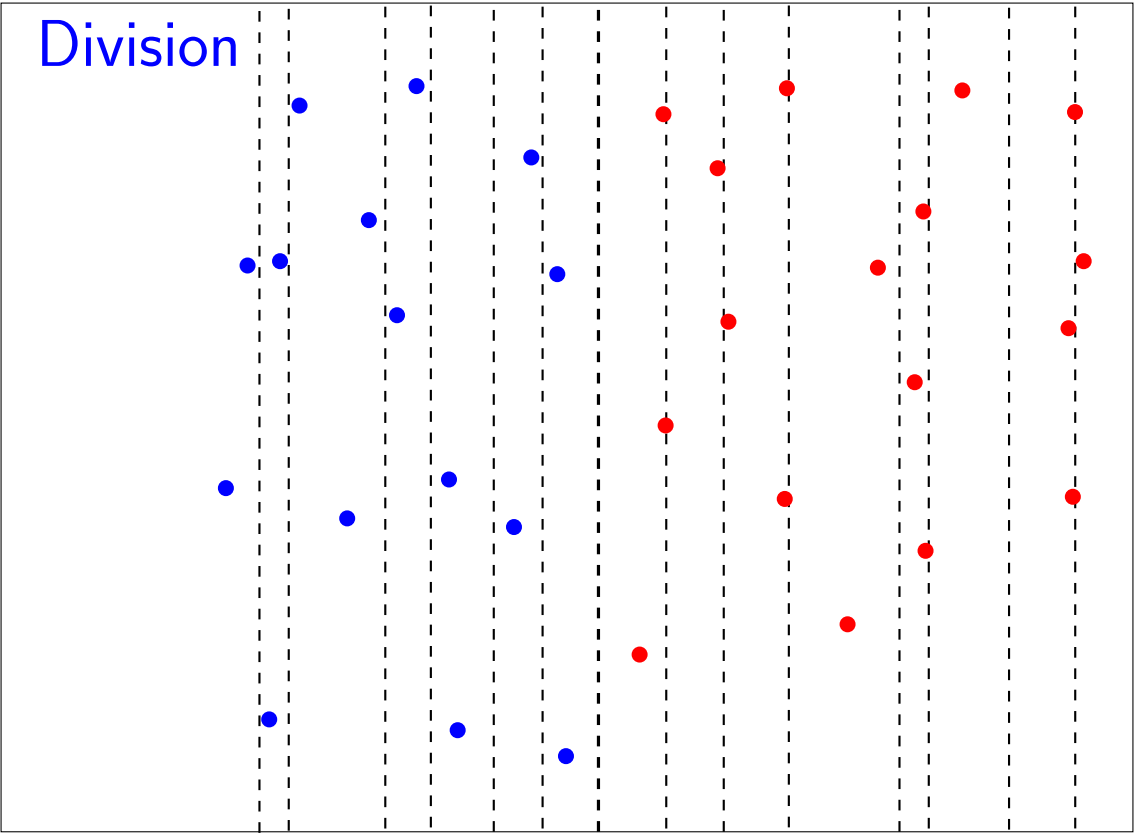
$O(n)$

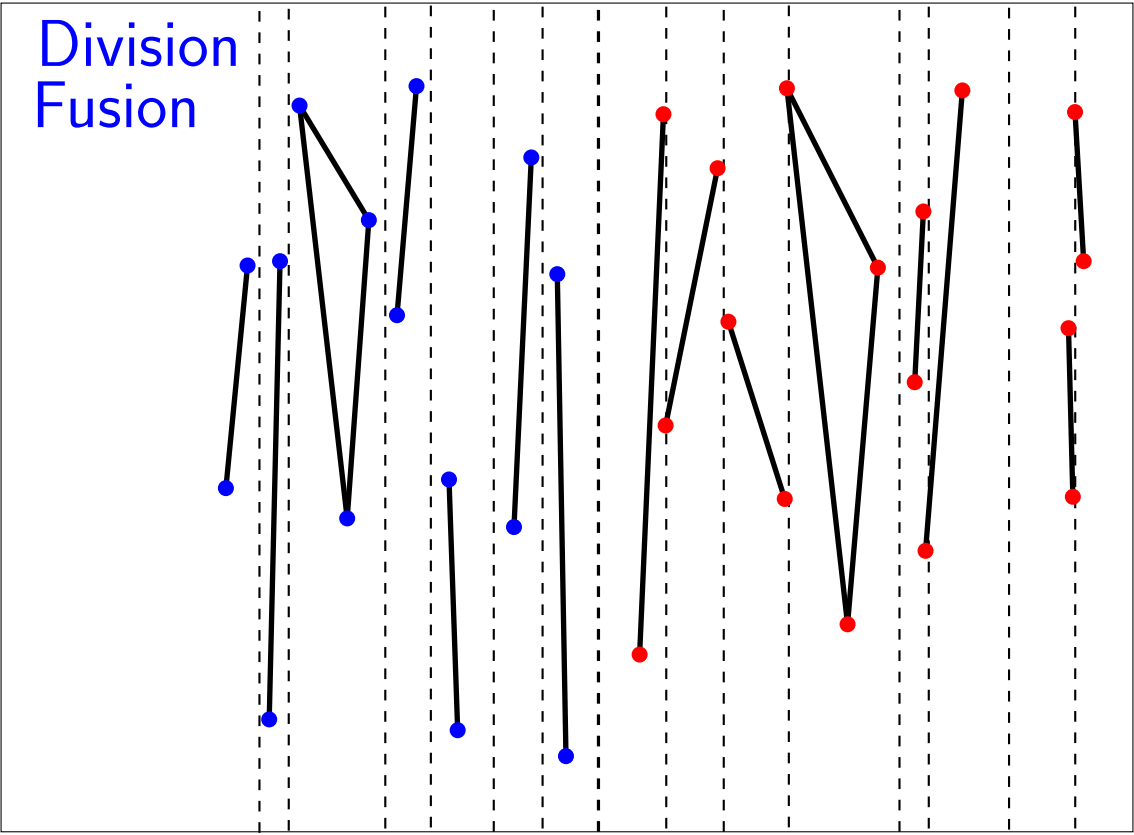

Division

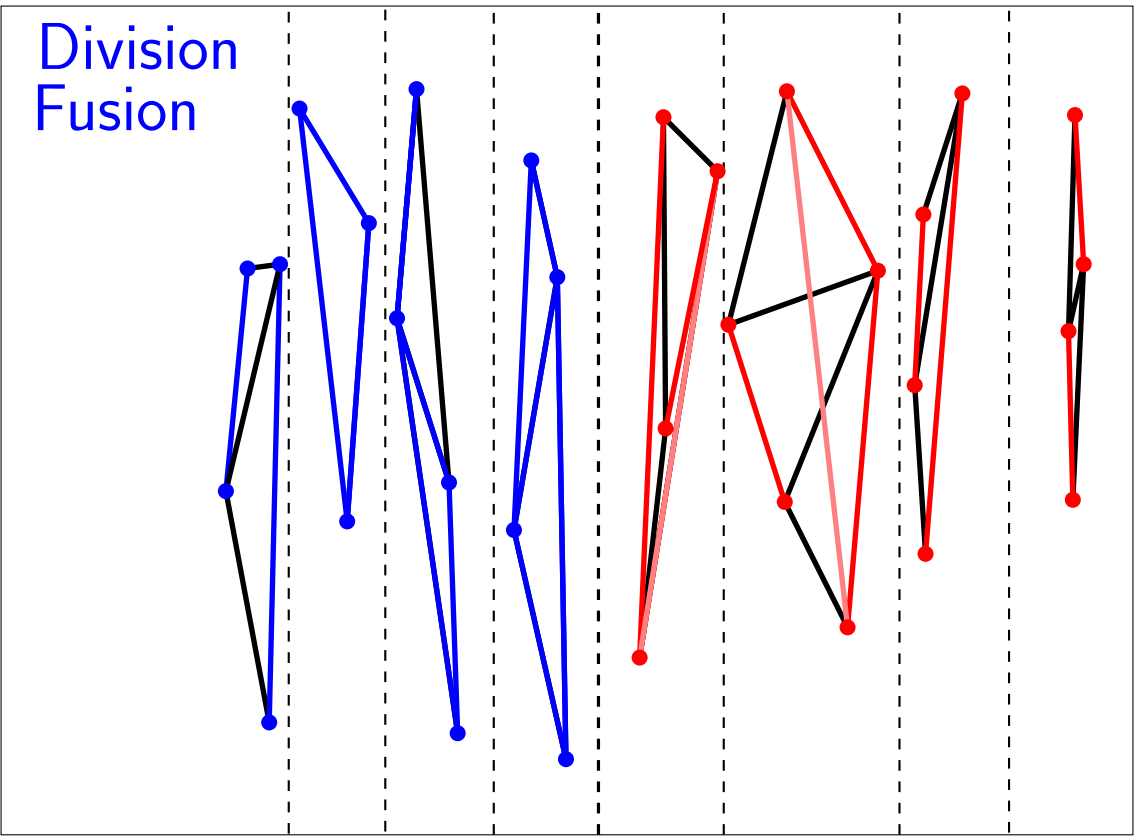

Division

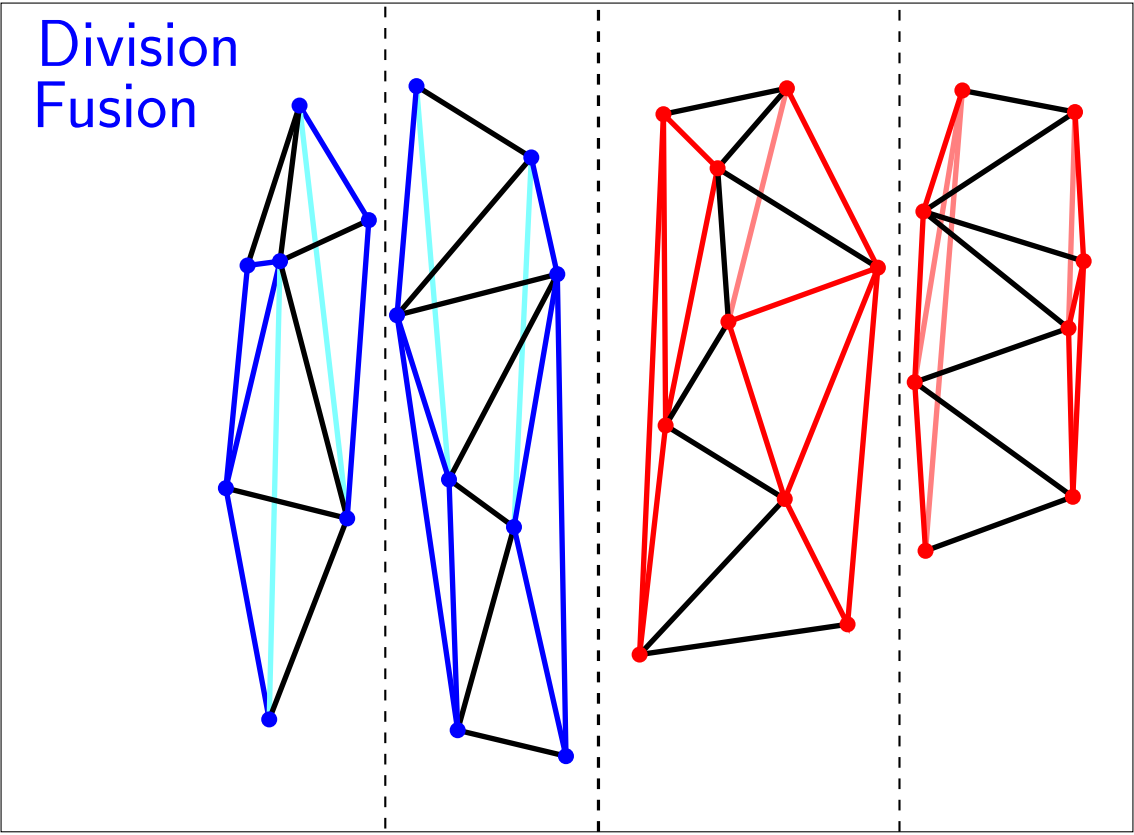

Division

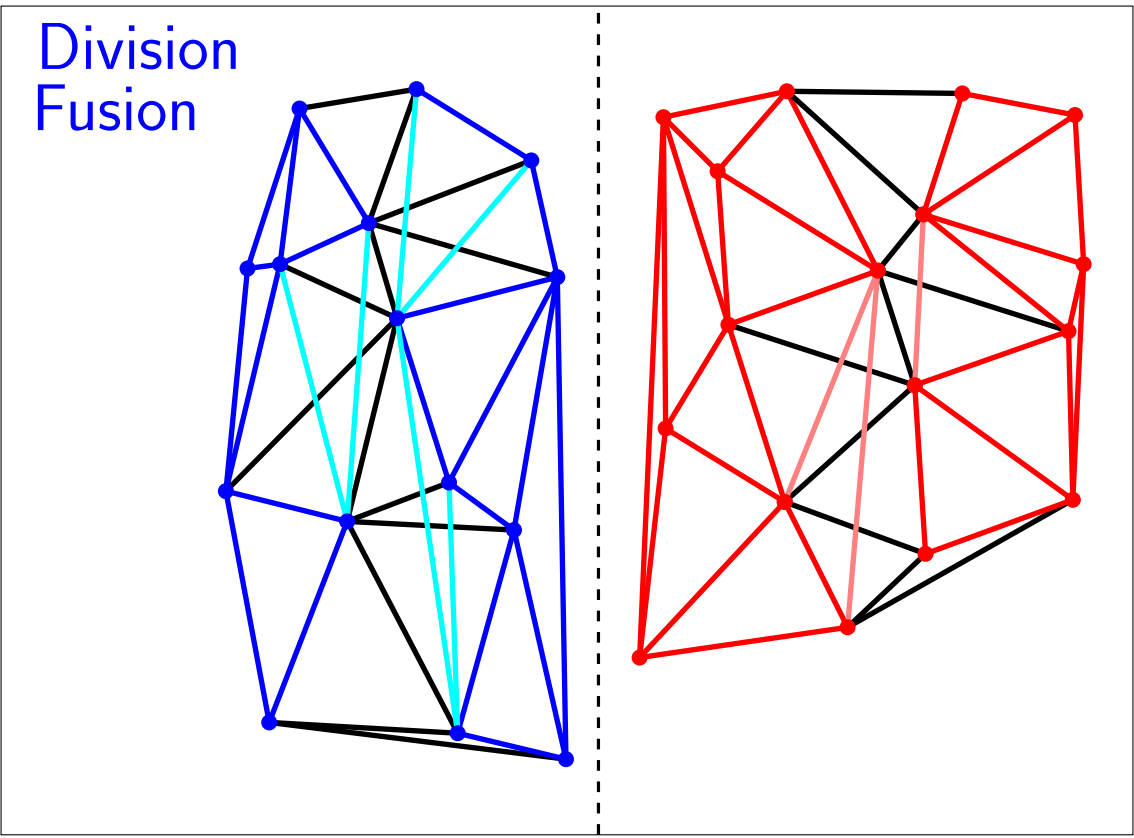
Division

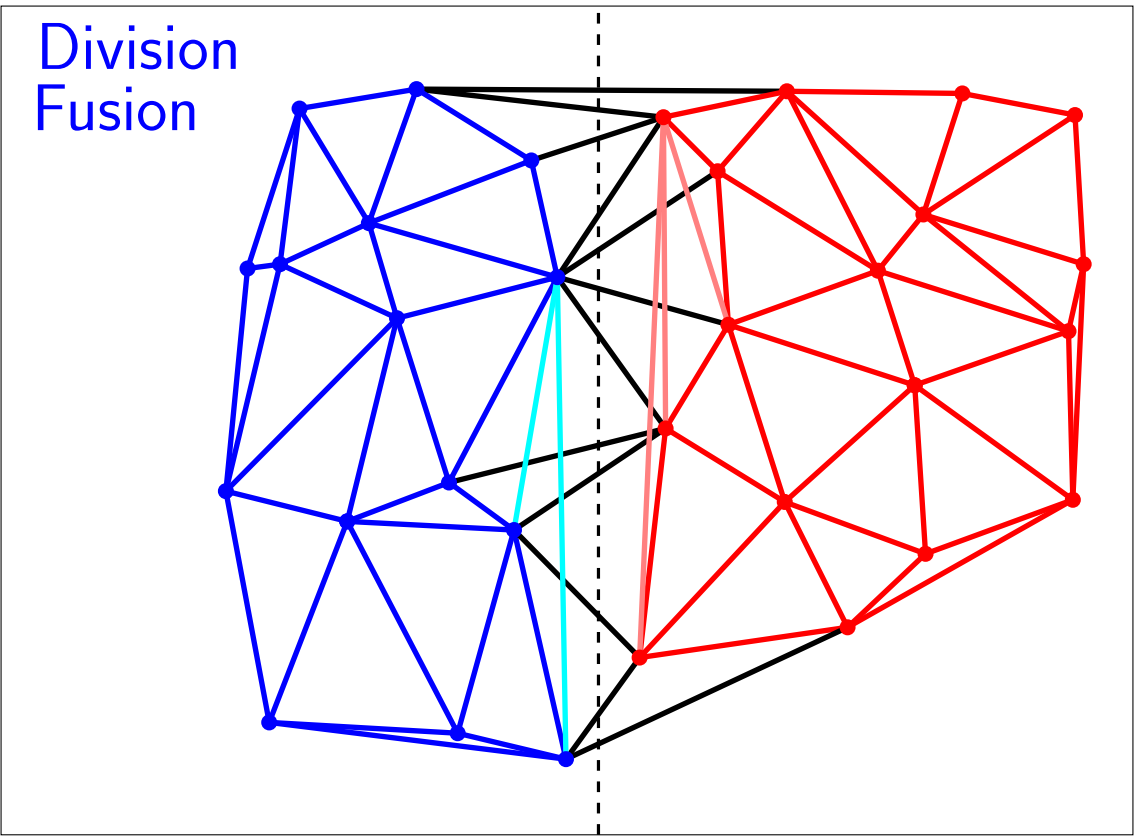


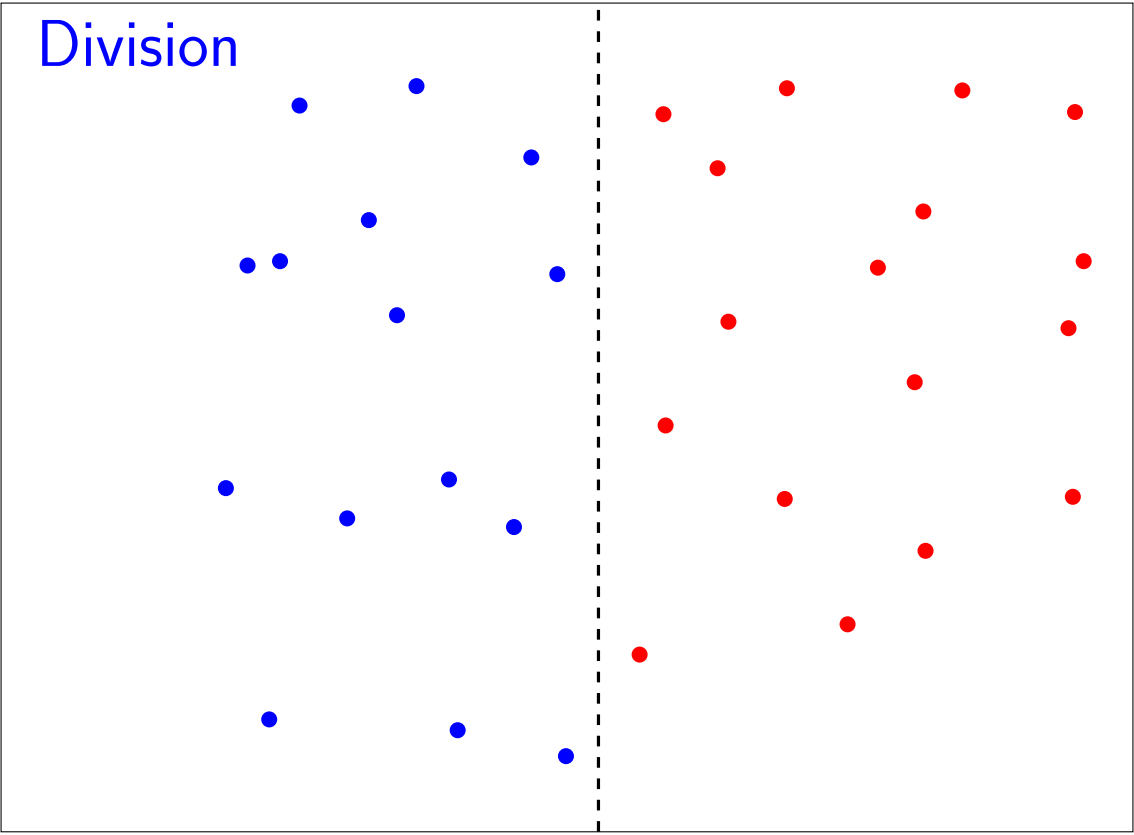

Division
Fusion

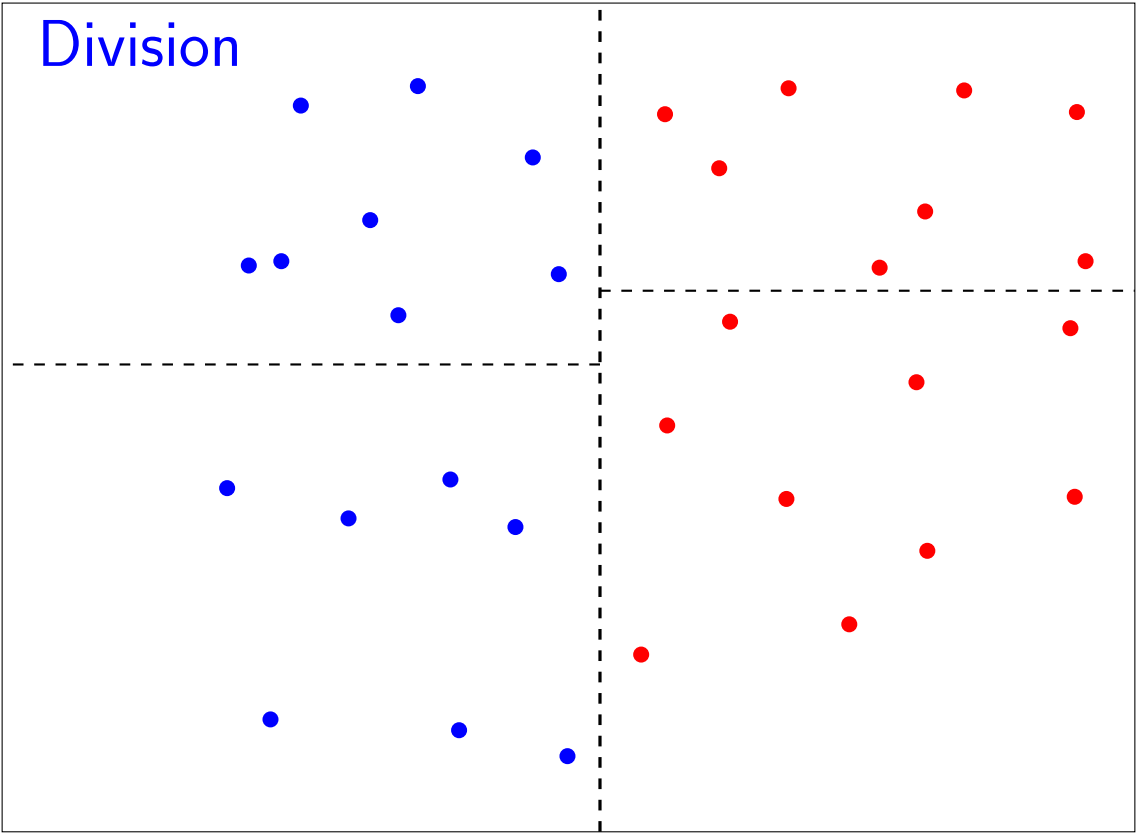

Division
Fusion

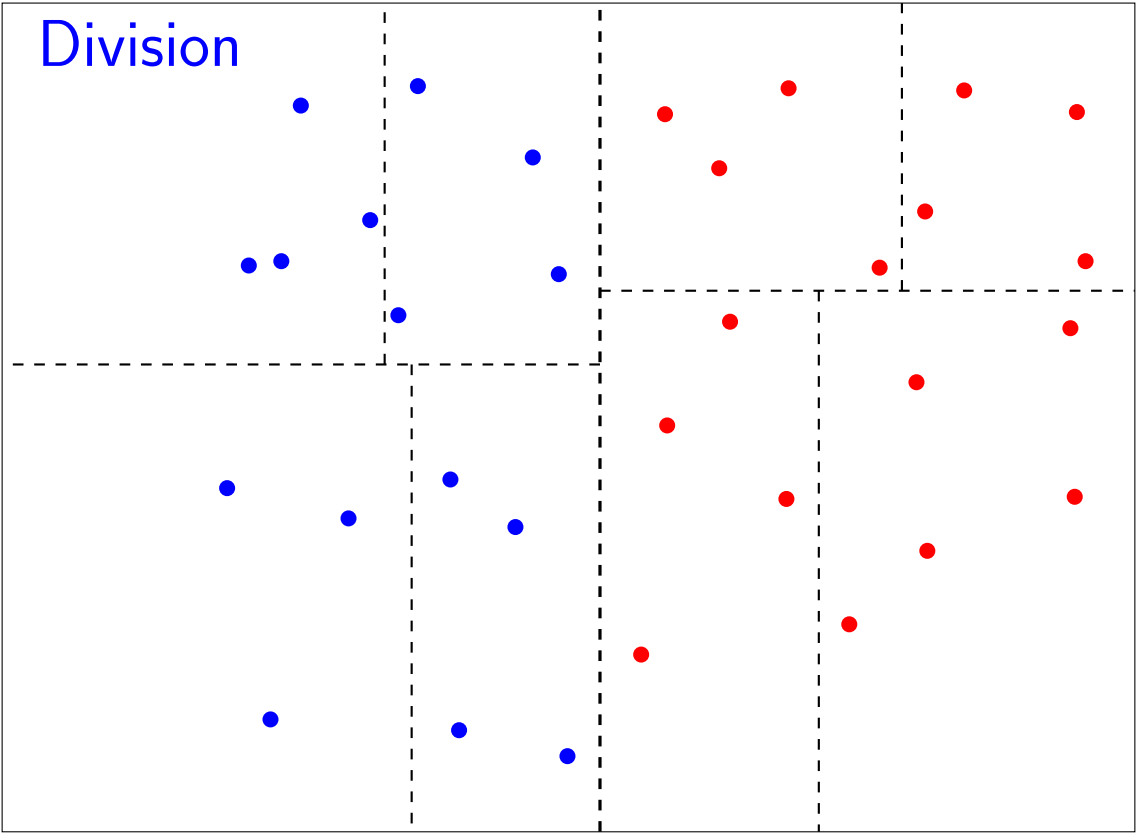

Division
Fusion

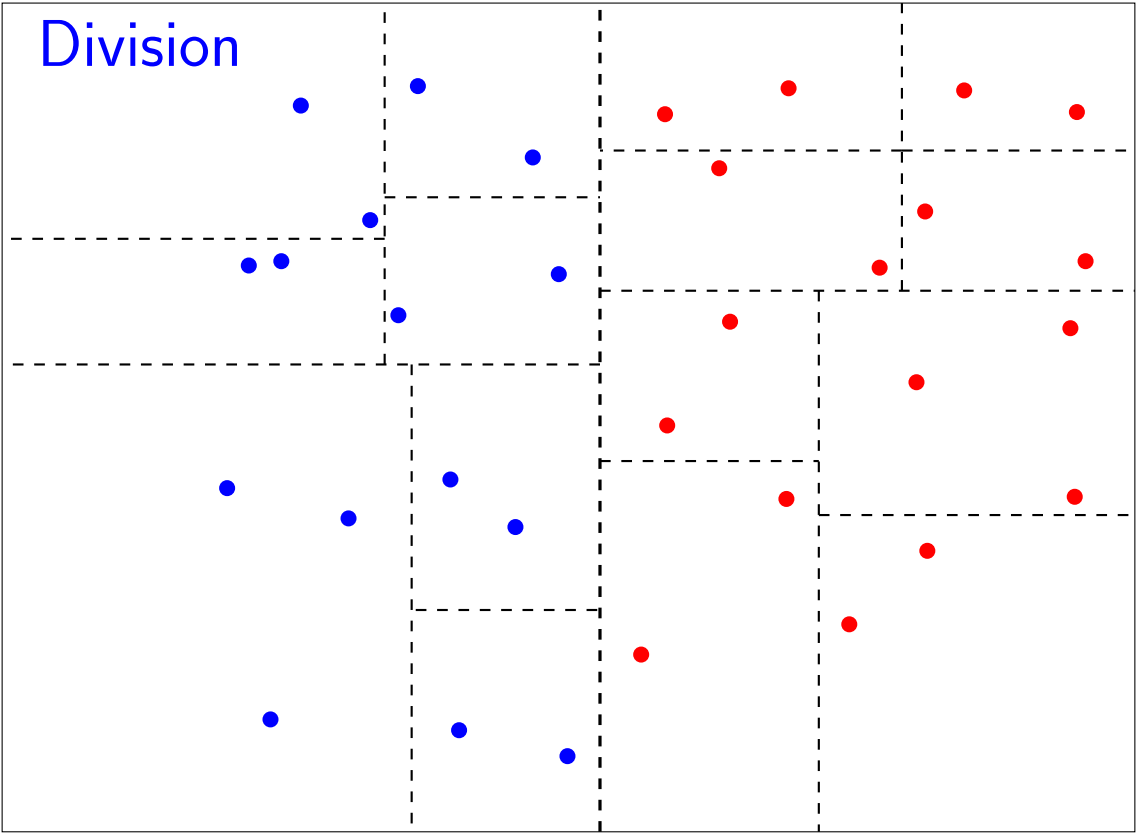

Division
Fusion

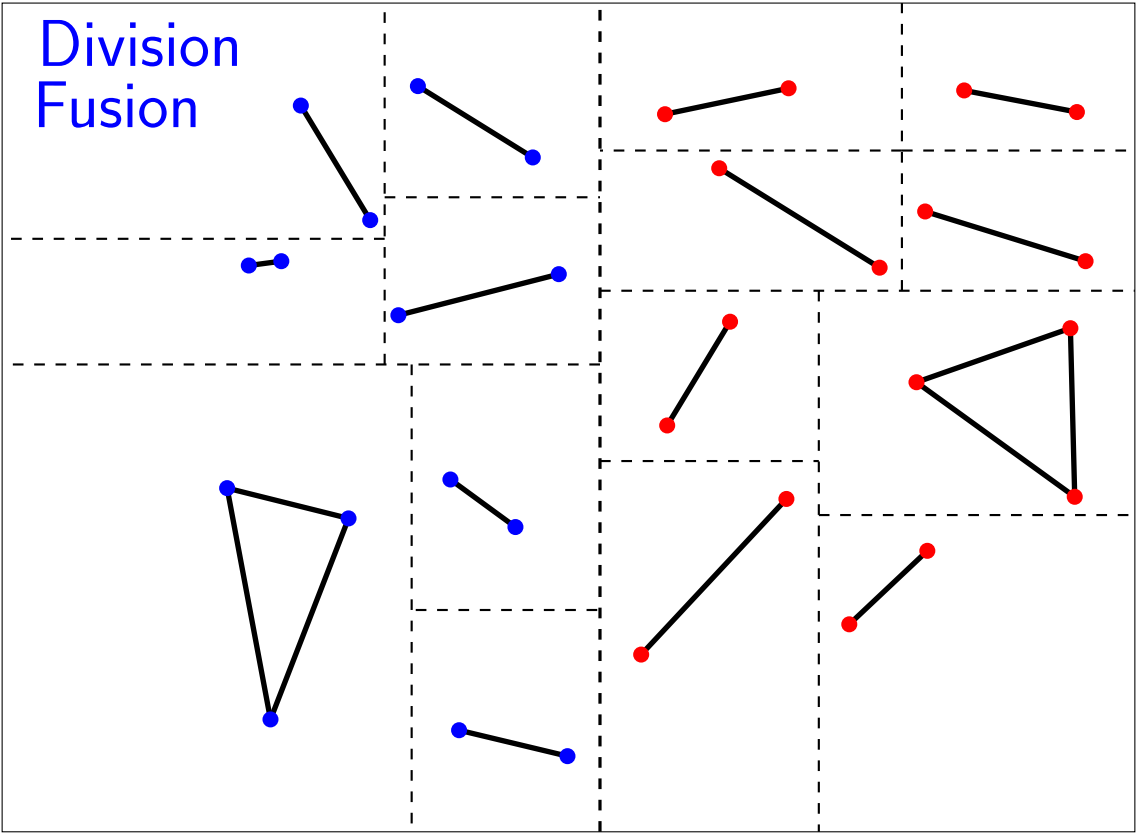

Division
Fusion

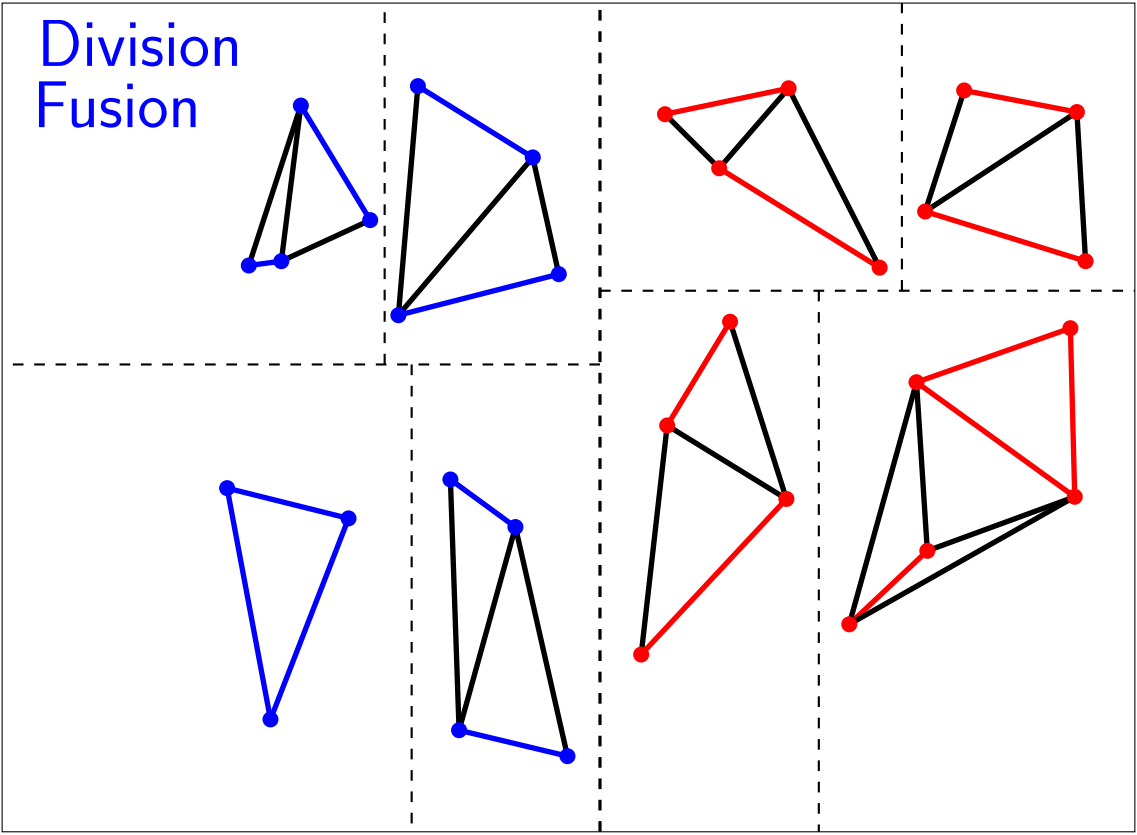

Division

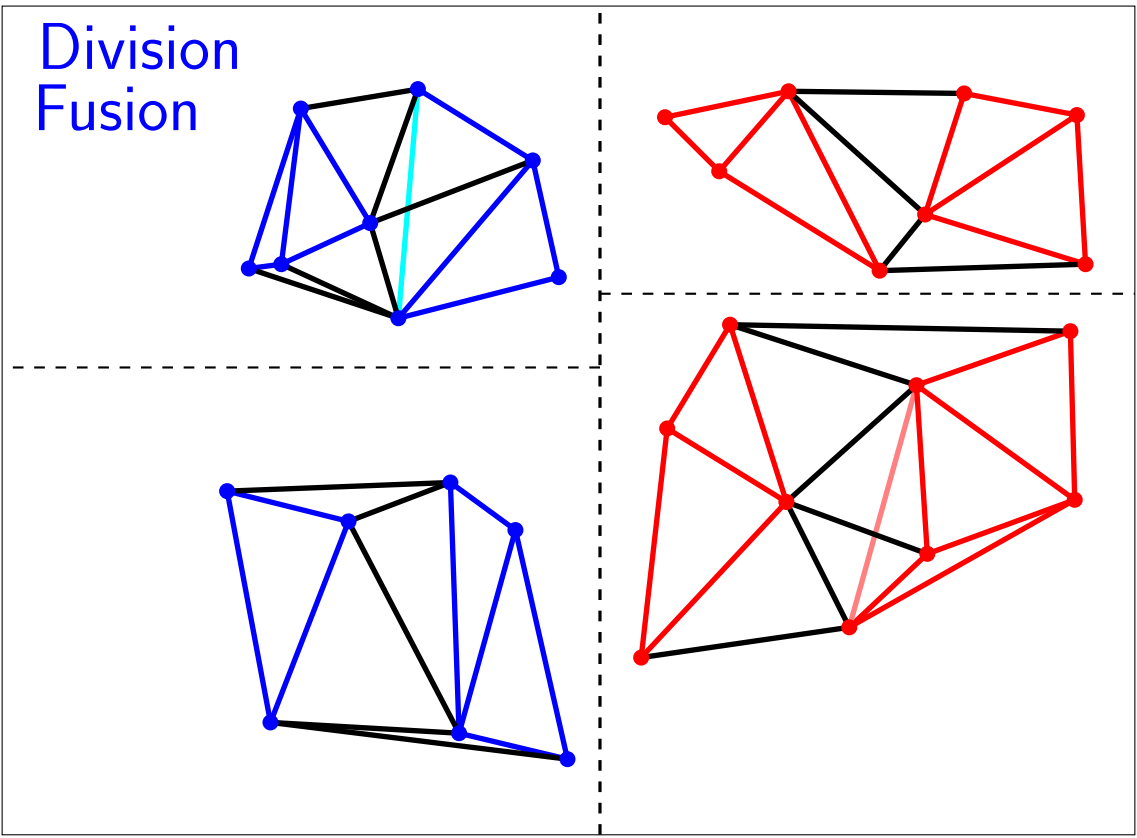

Division

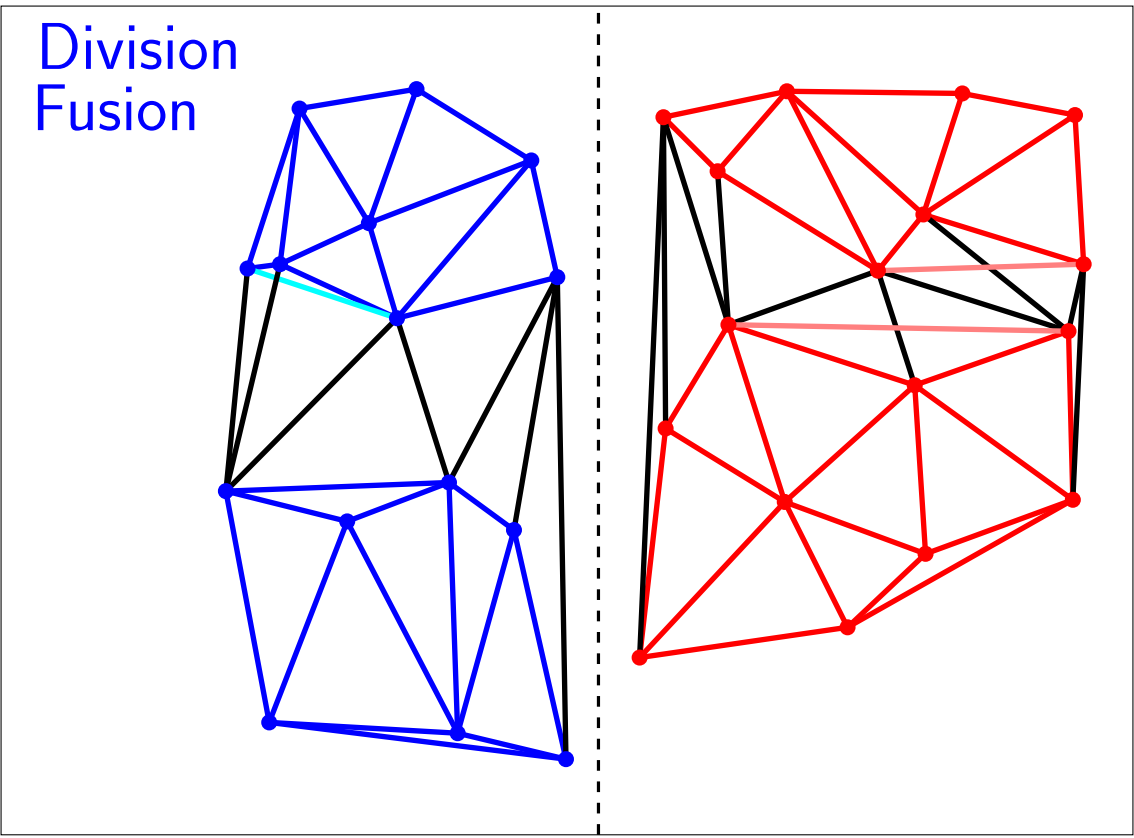

Division

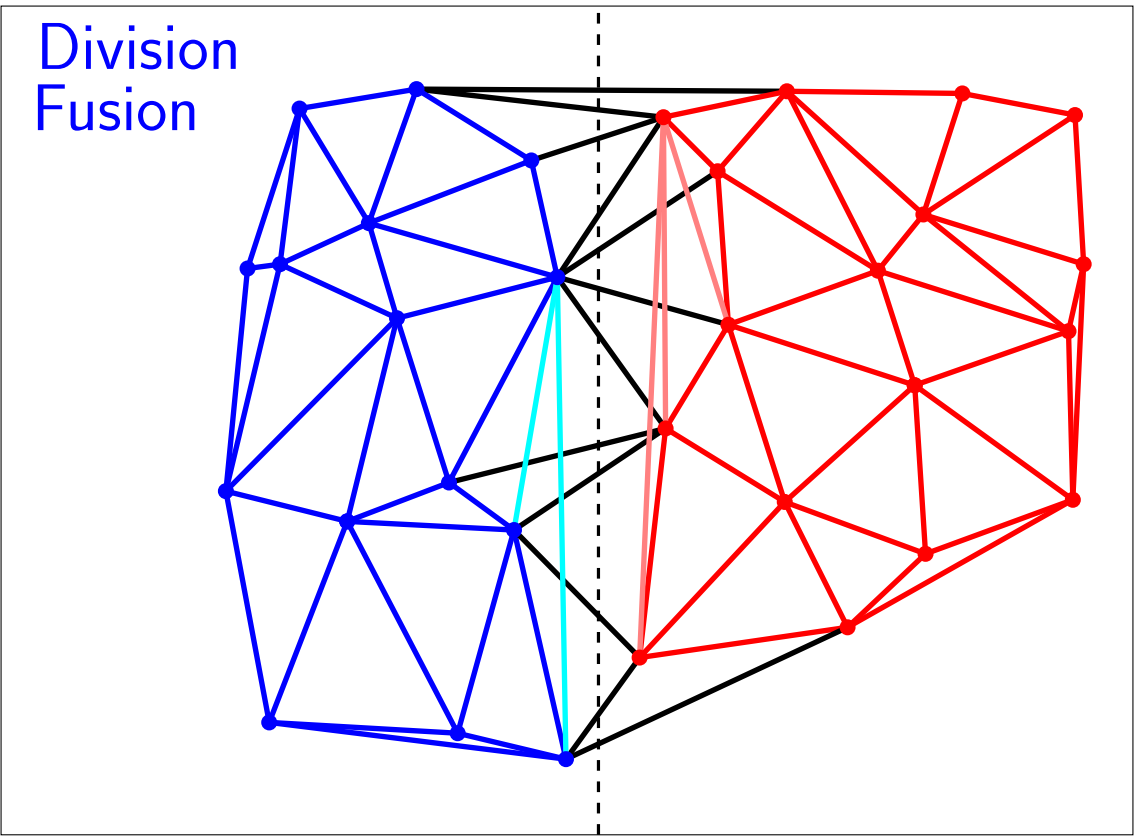

Division

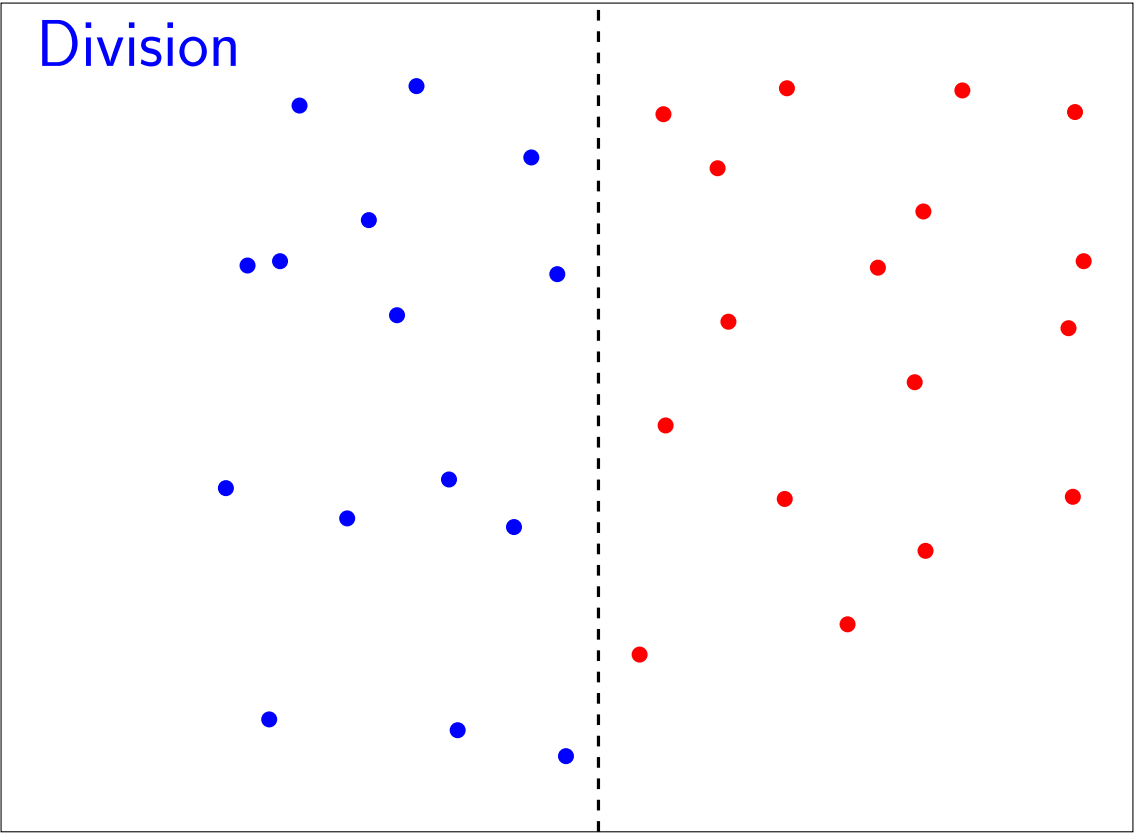

Division
Fusion

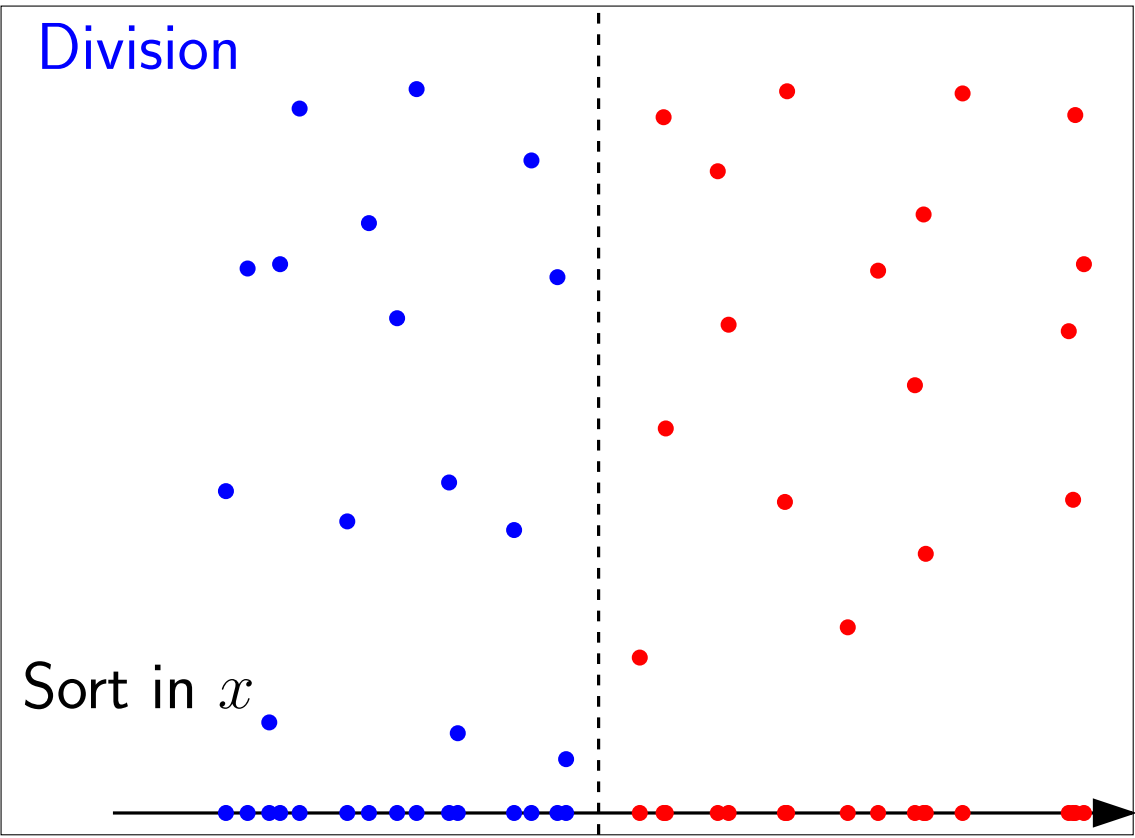

Division
Fusion

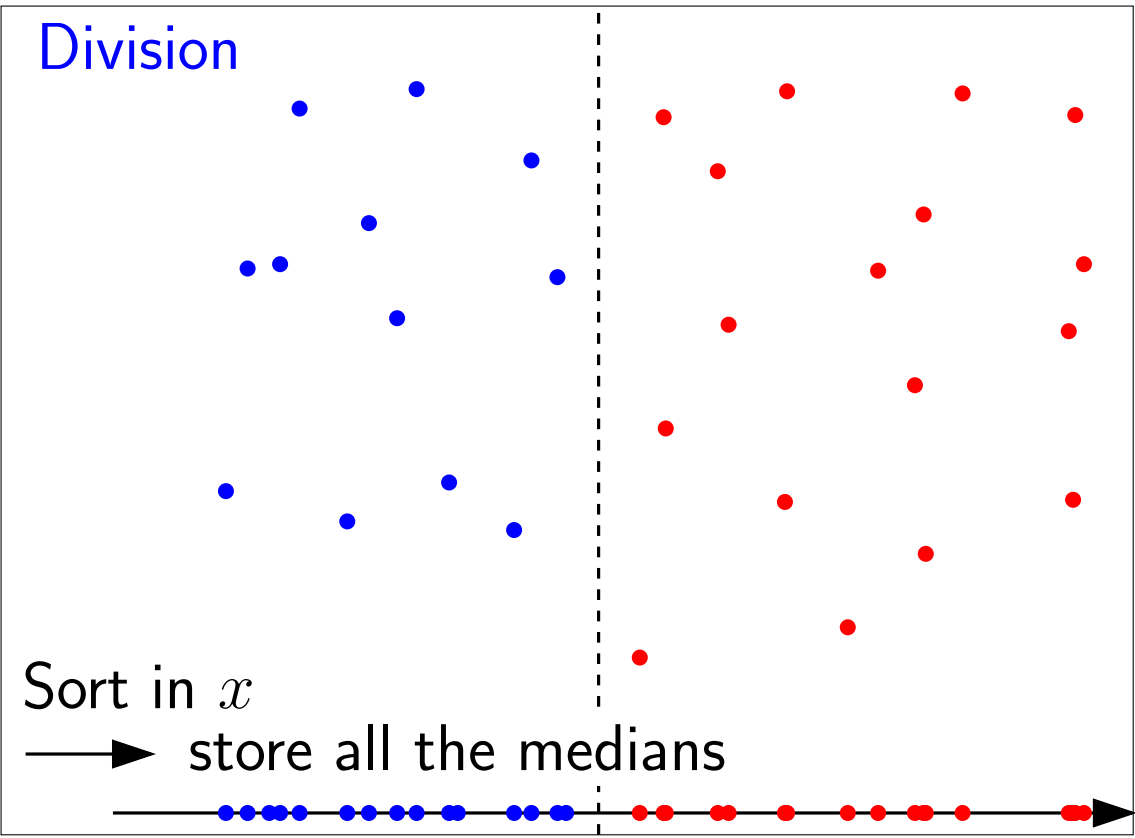

Division
Fusion

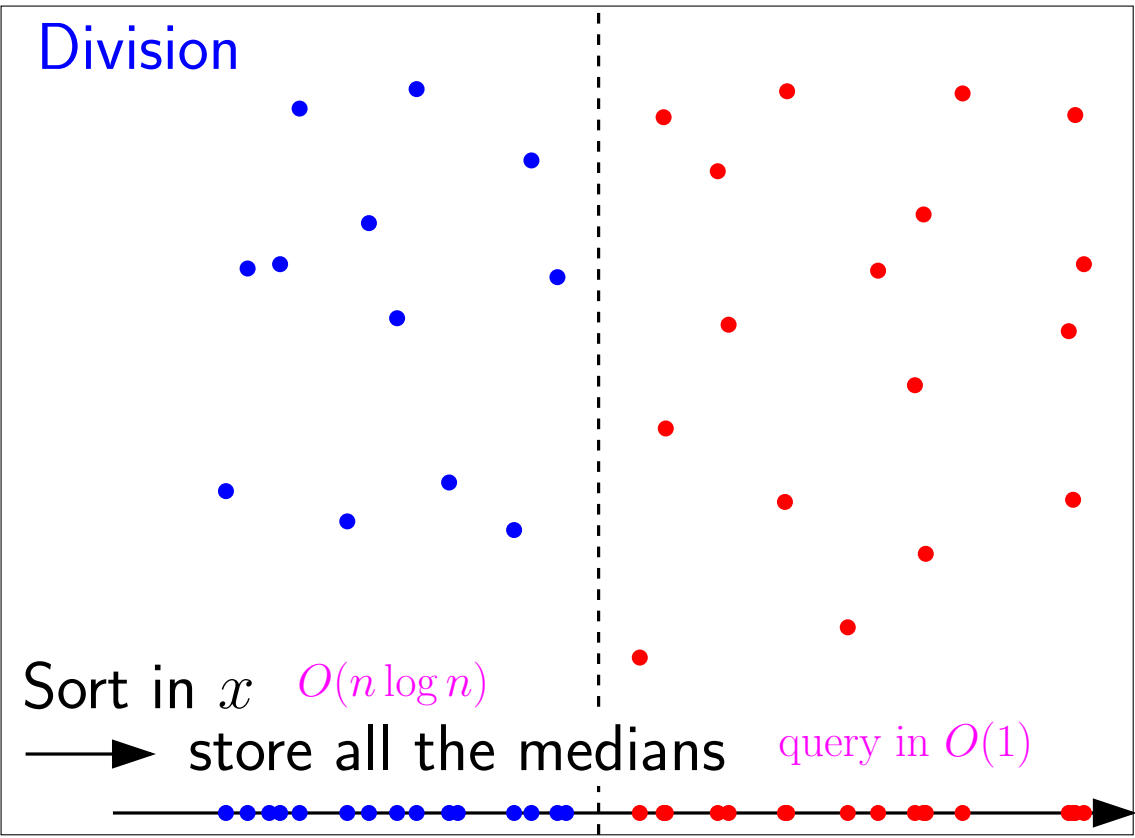

Division
Fusion


Division
Fusion

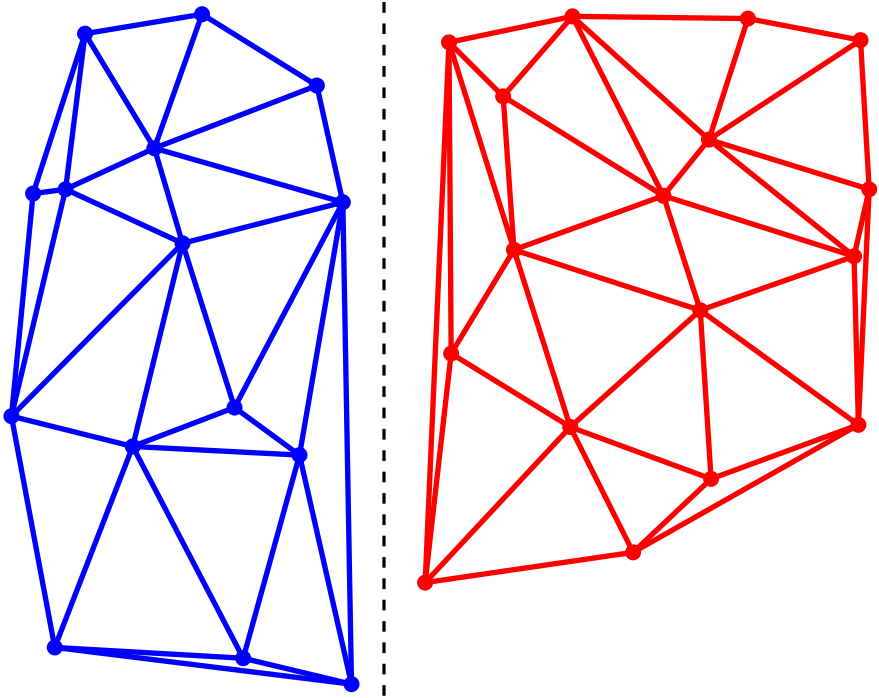

Division


Division
Sort in x

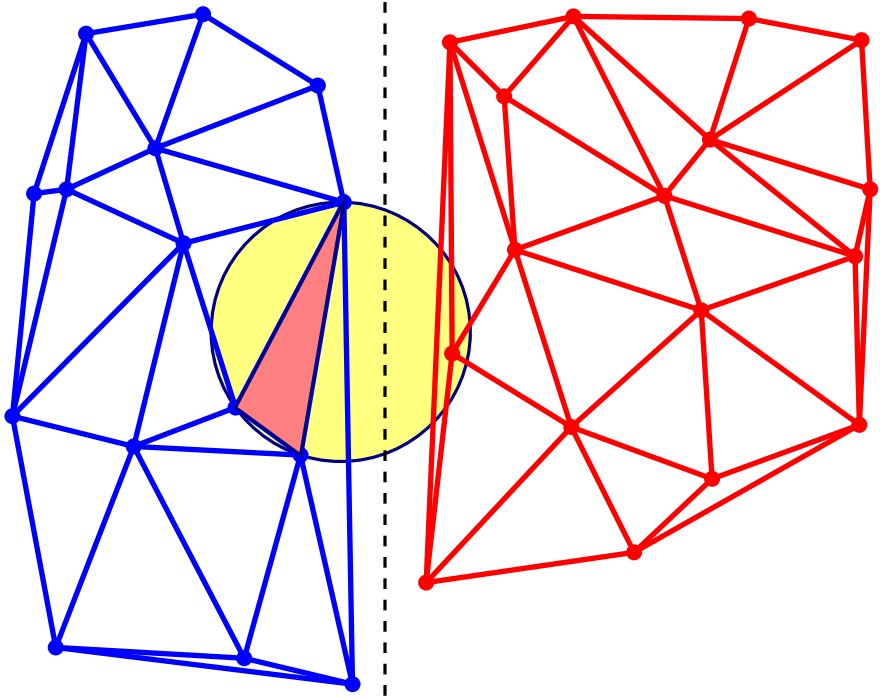

Division
Sort in x
store all the medians


Division
Sort in x
$O(n \log n)$
store all the medians
query in $O(1)$

Fusion

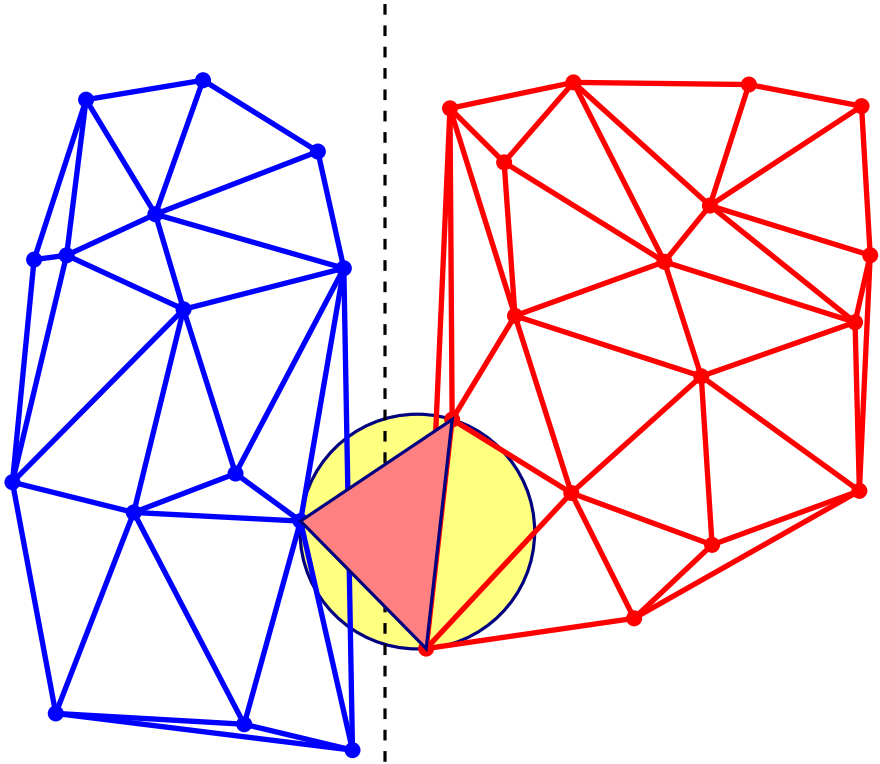


Fusion

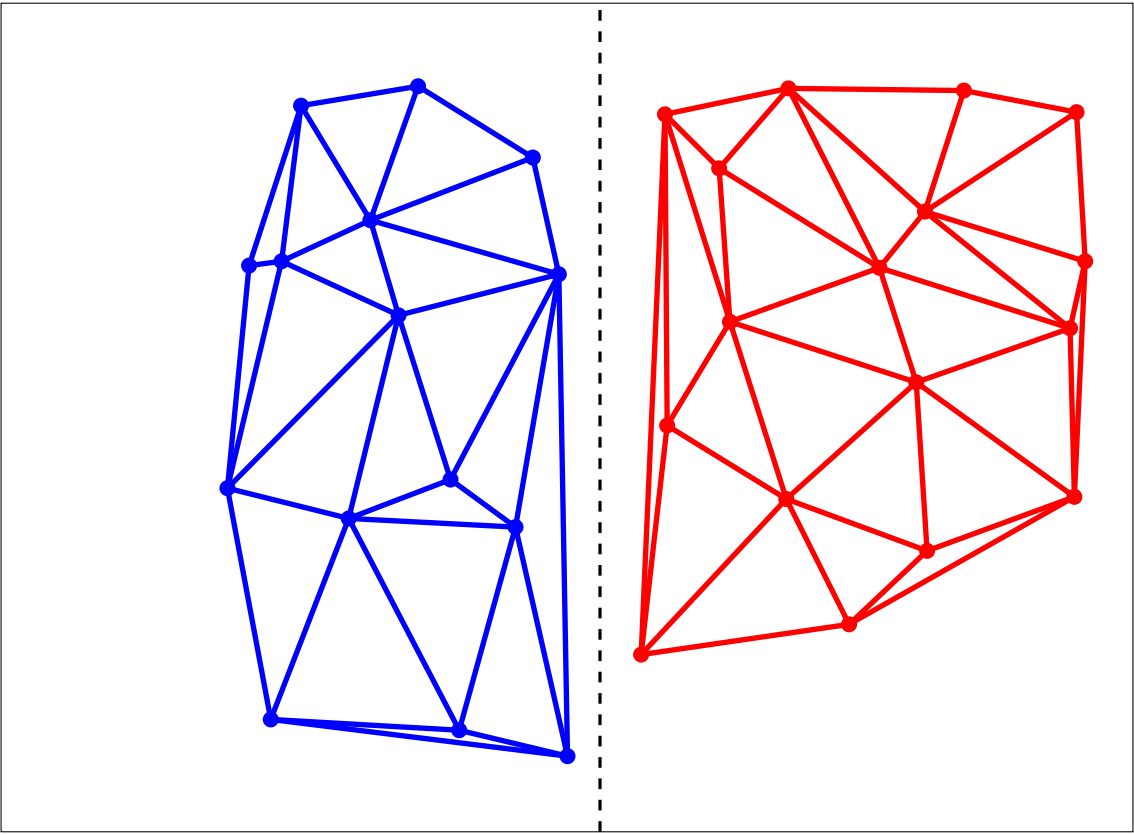


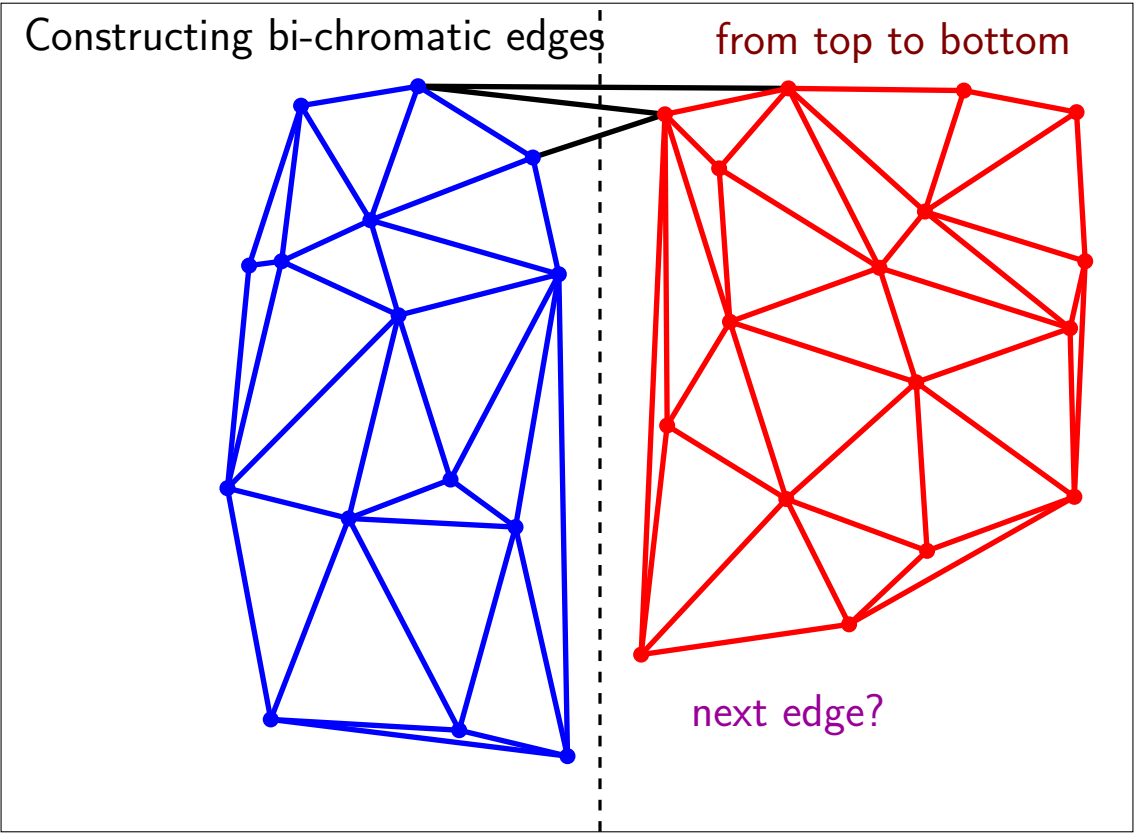
Monochromatic triangles to be deleted

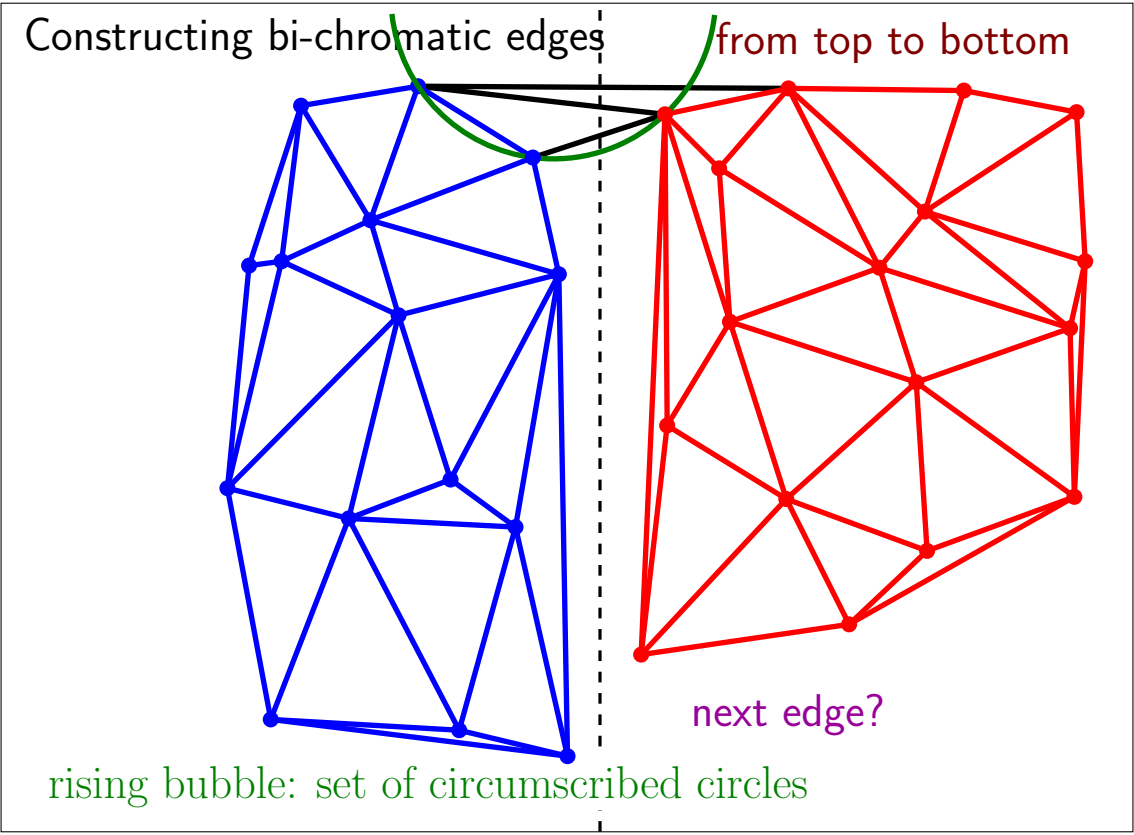
Fusion

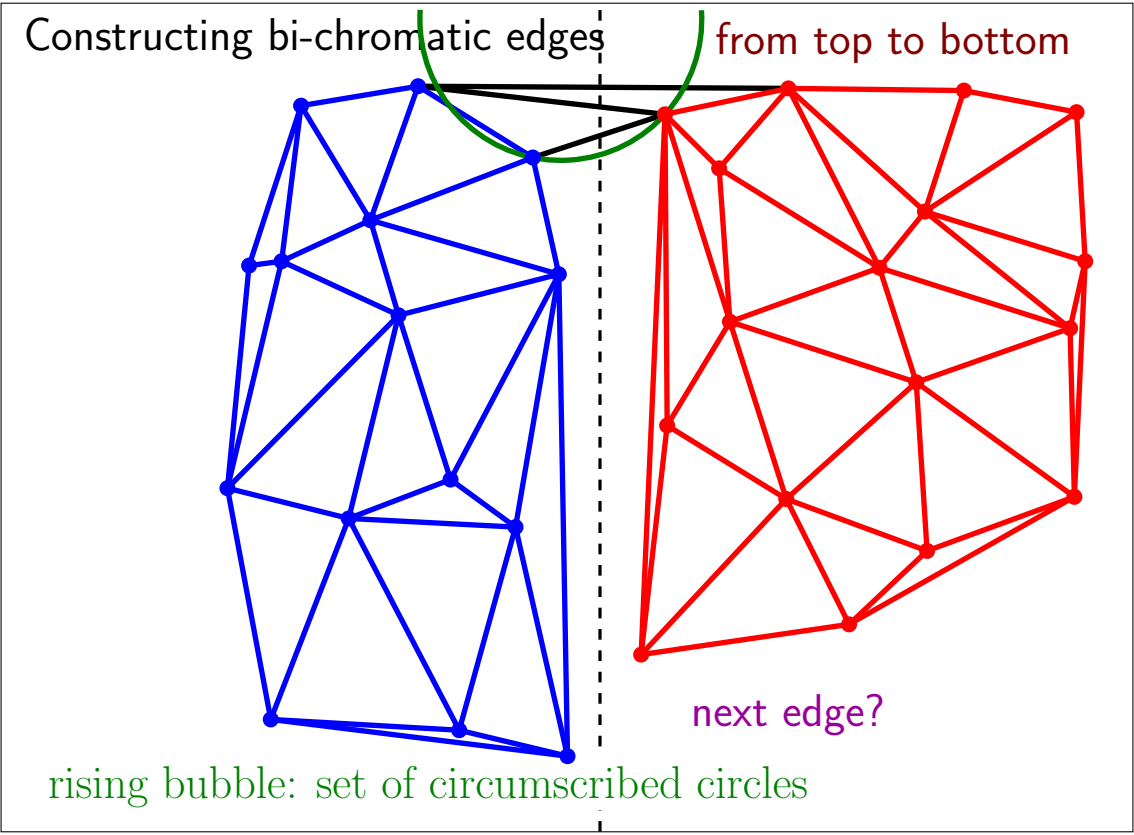


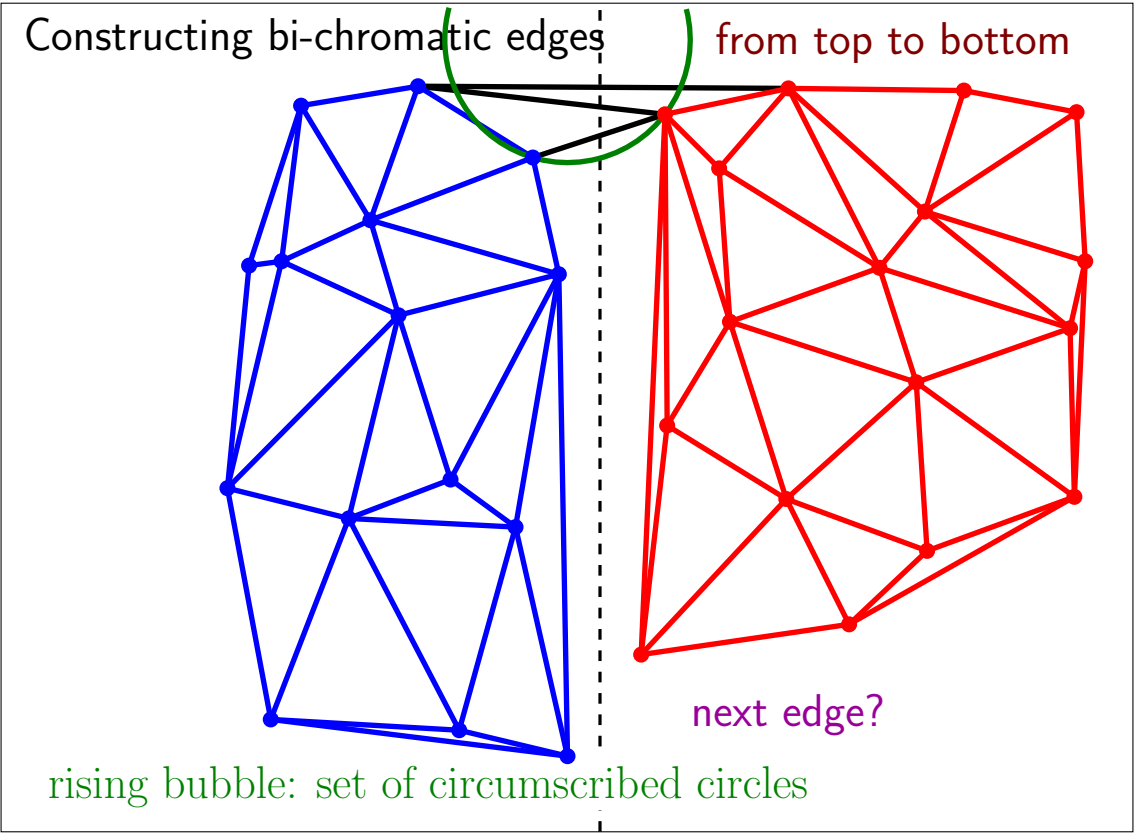
Bi-chromatic triangles to be construced

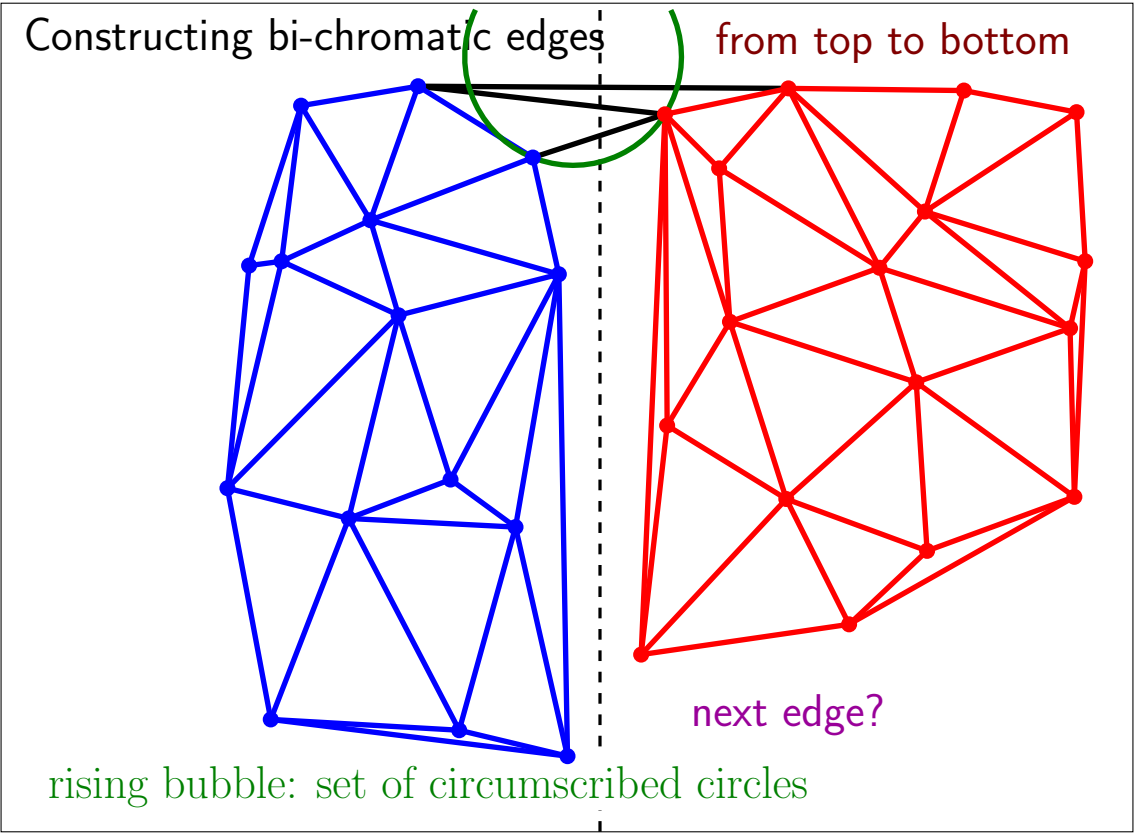


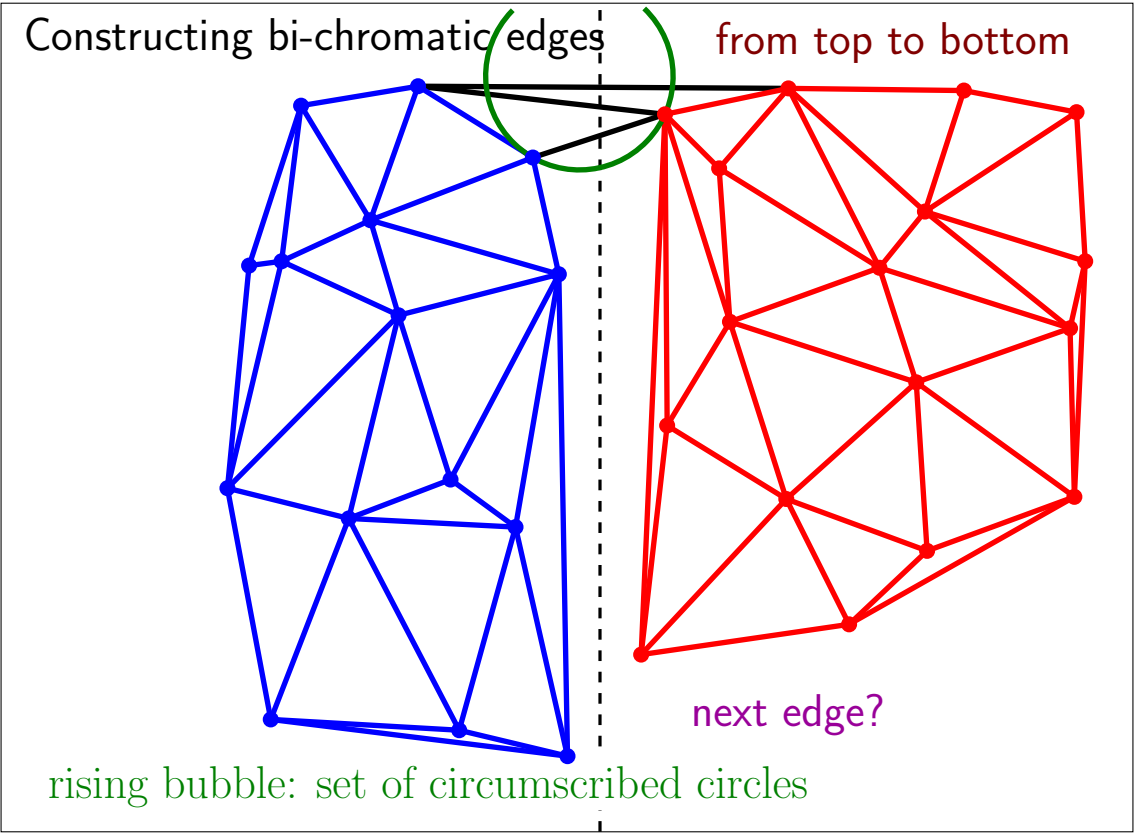

Constructing bi-chromatic edges
from top to bottom
next edge?

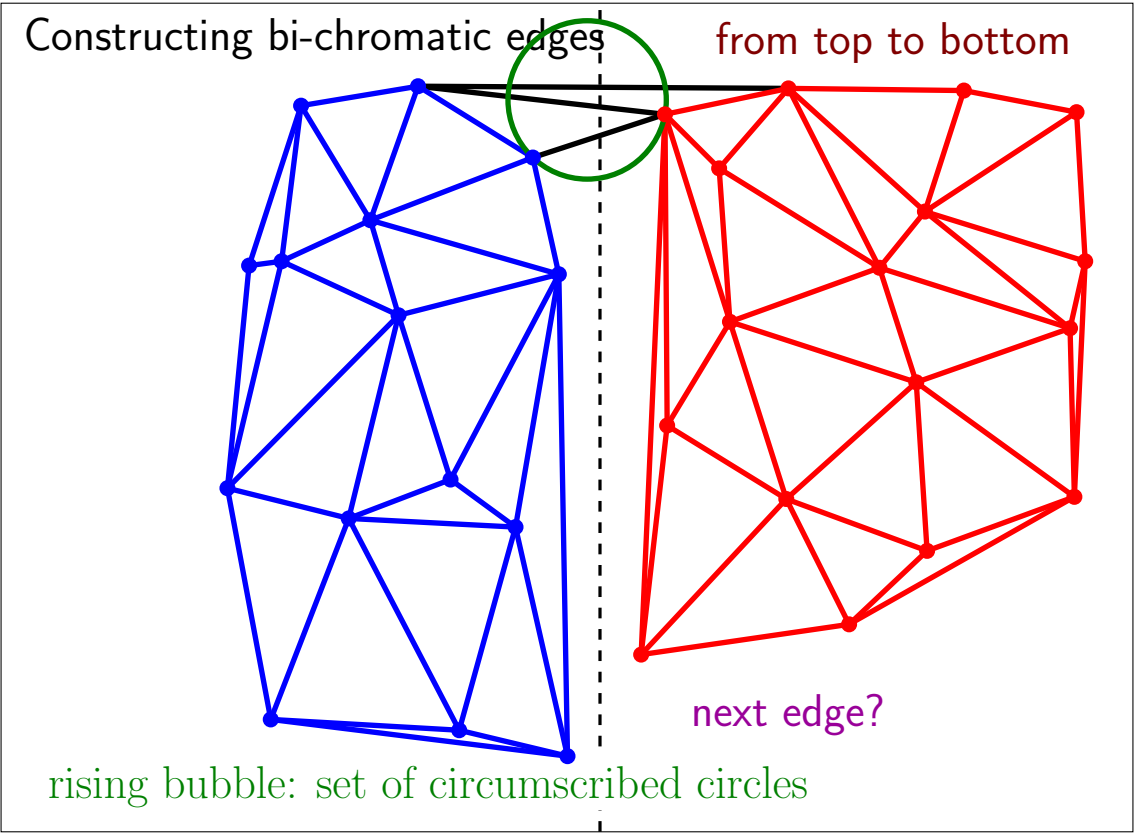

Constructing bi-chromatic edges
from top to bottom
next edge?
rising bubble: set of circumscribed circles

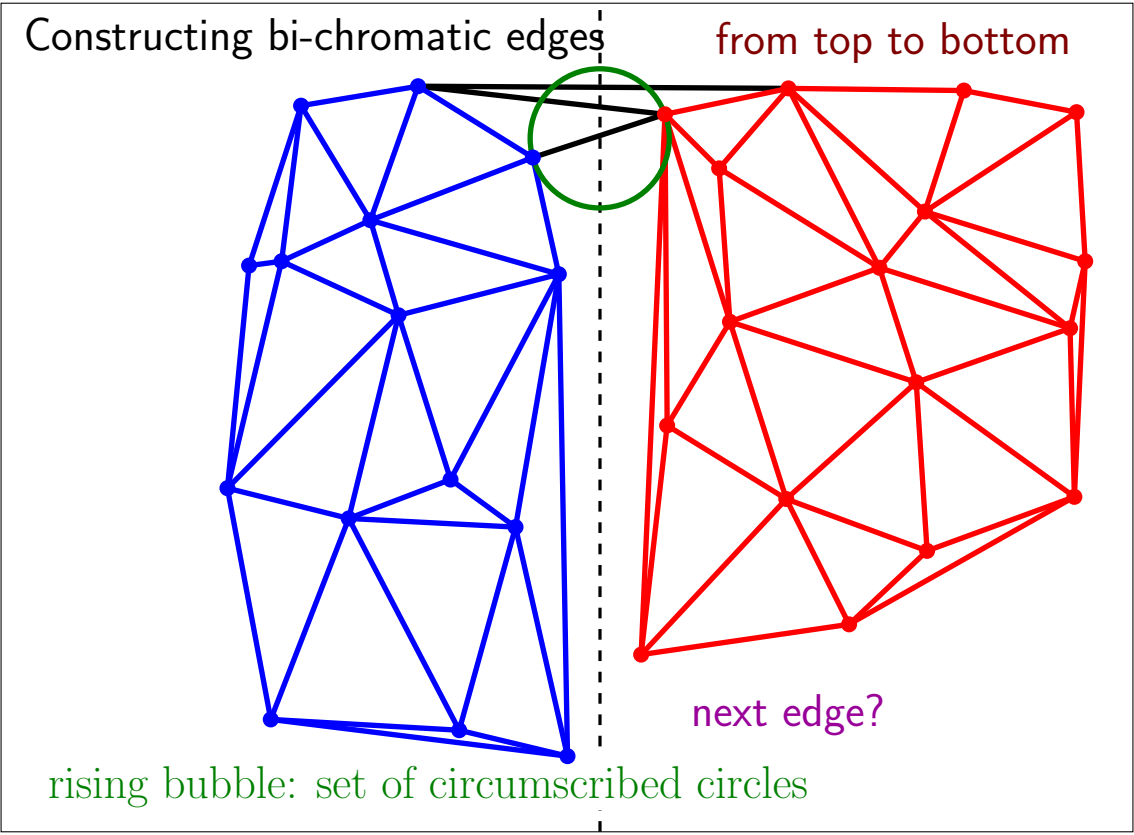

Constructing bi-chromatic edges
from top to bottom
next edge?
rising bubble: set of circumscribed circles

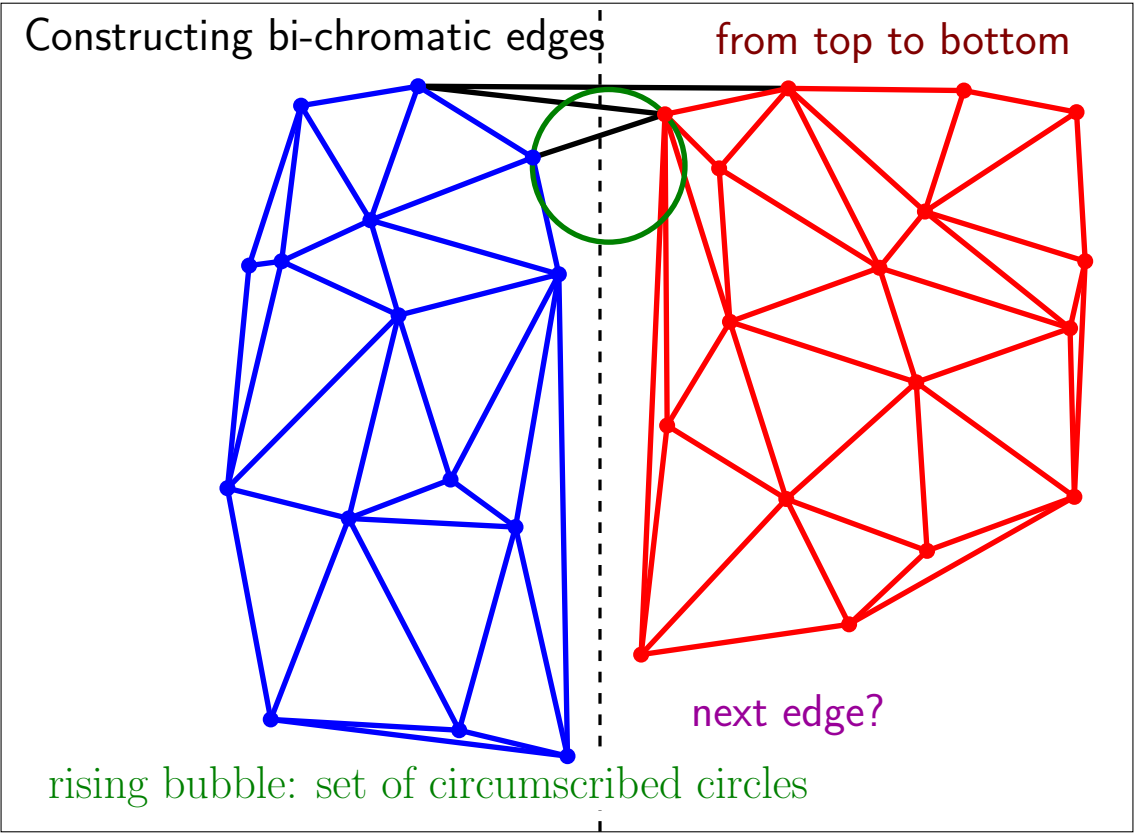

Constructing bi-chromatic edges
from top to bottom
next edge?
rising bubble: set of circumscribed circles

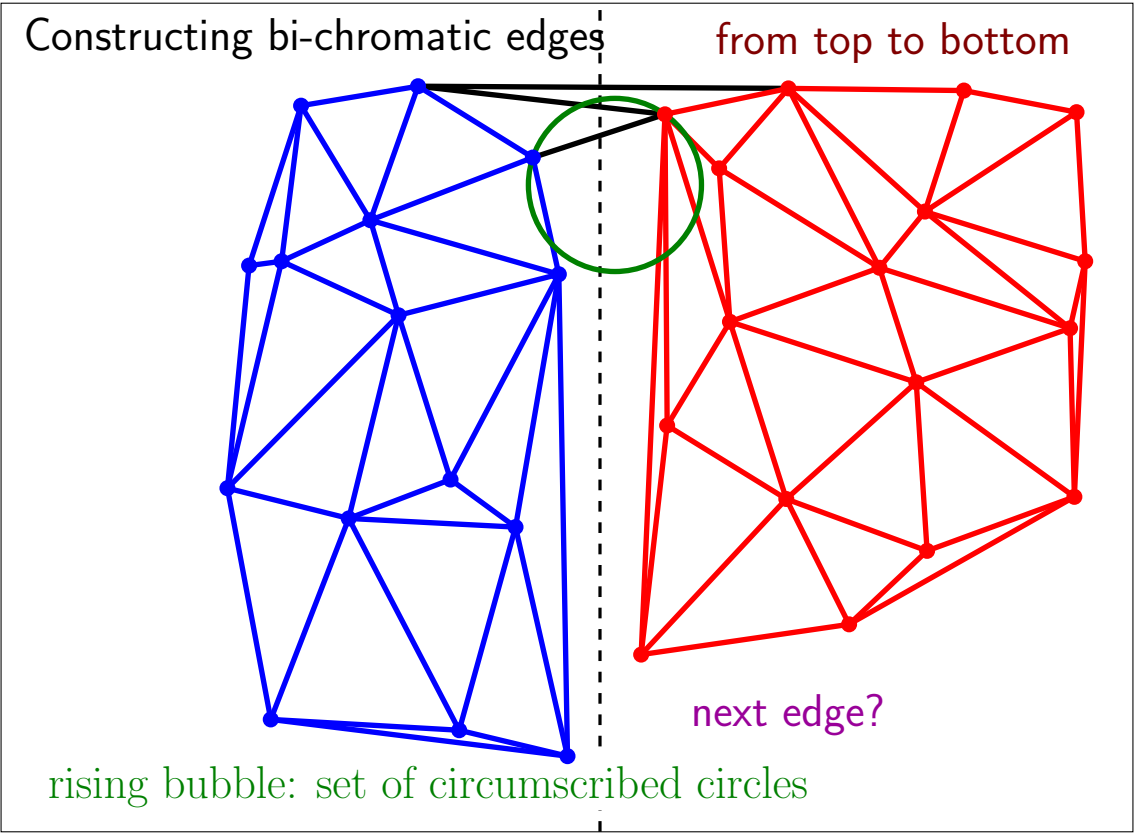

Constructing bi-chromatic edges
from top to bottom
next edge?
rising bubble: set of circumscribed circles

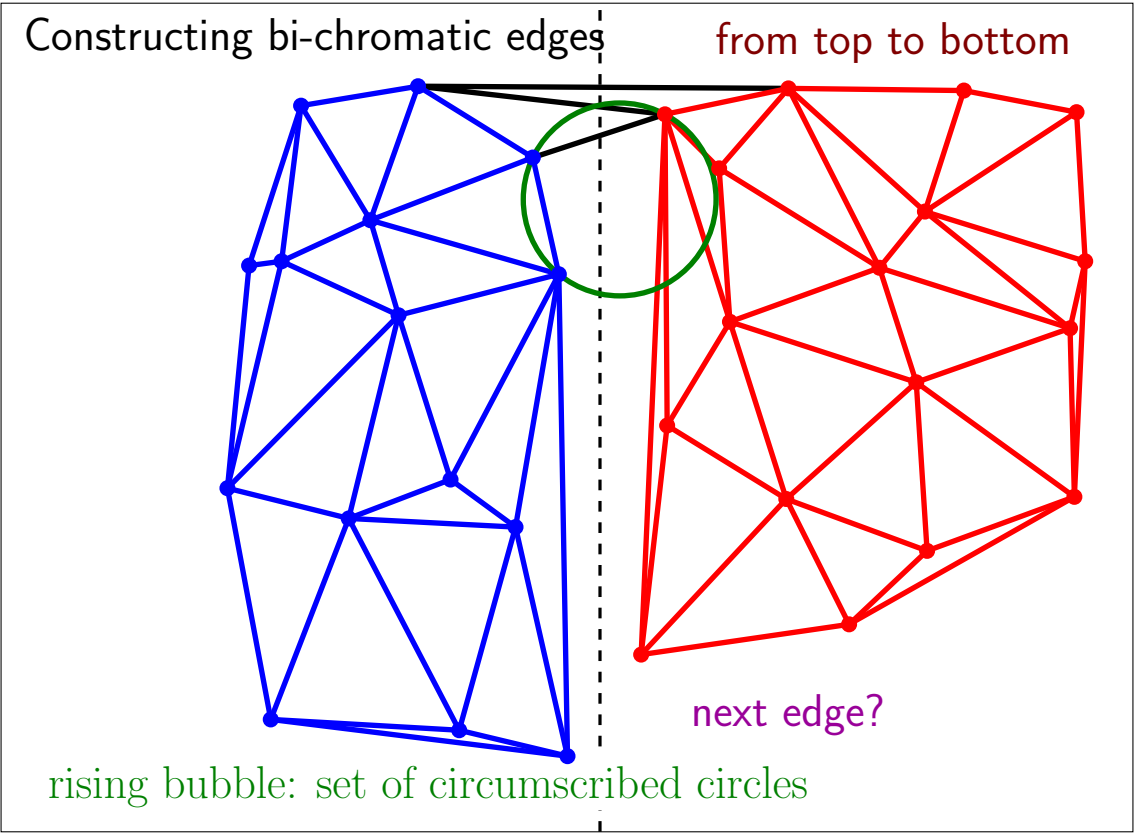

Constructing bi-chromatic edges
from top to bottom
next edge?
rising bubble: set of circumscribed circles

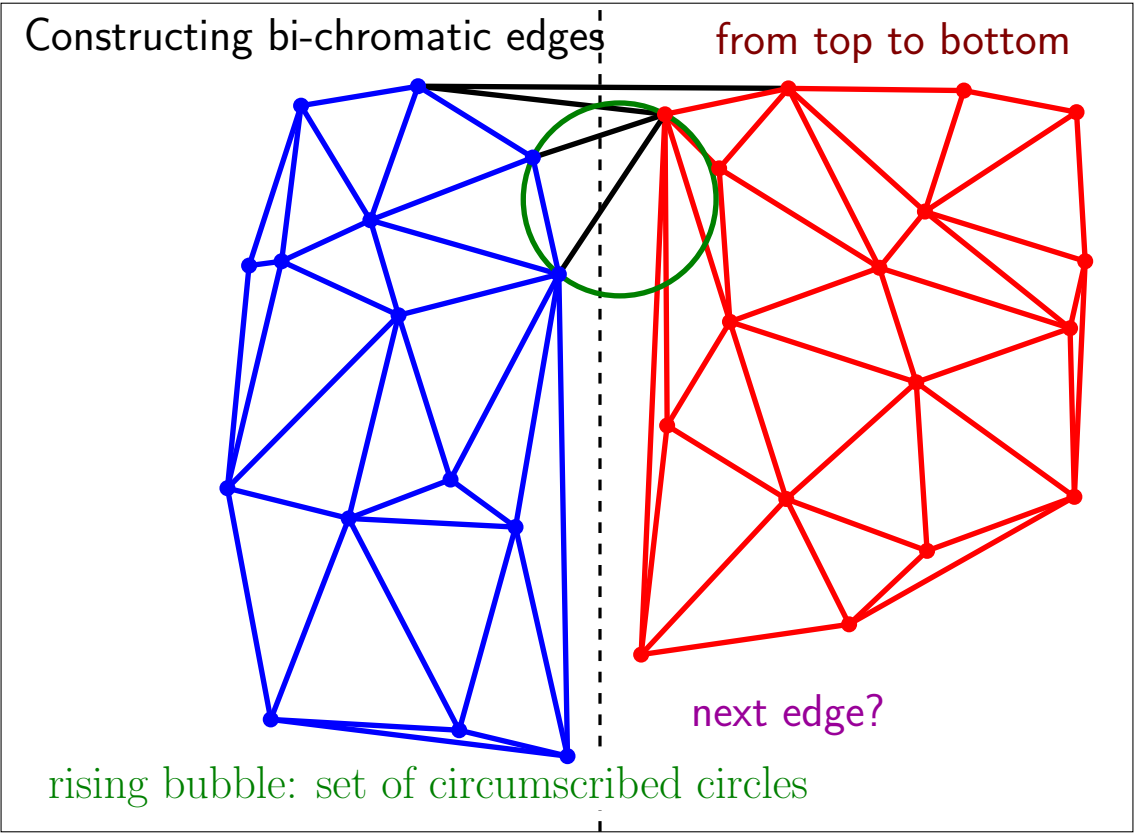

Constructing bi-chromatic edges
from top to bottom
next edge?
rising bubble: set of circumscribed circles

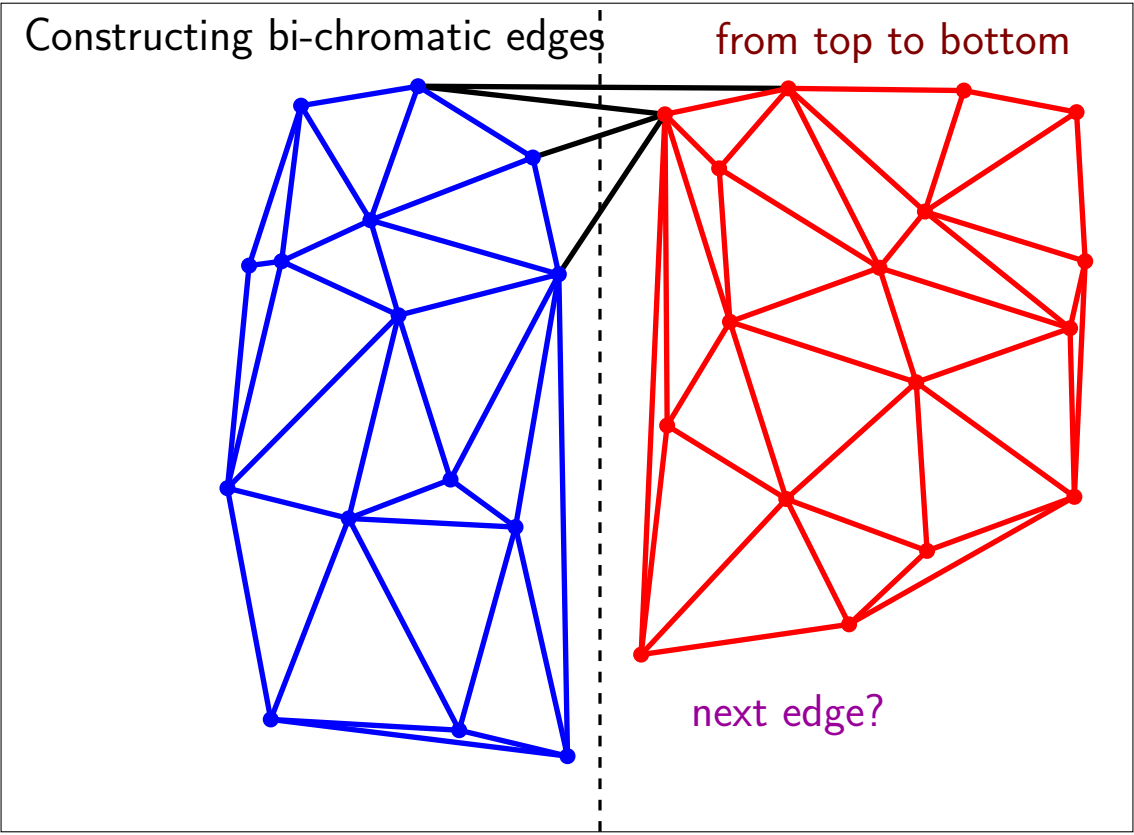

Constructing bi-chromatic edges
from top to bottom
next edge?
rising bubble: set of circumscribed circles

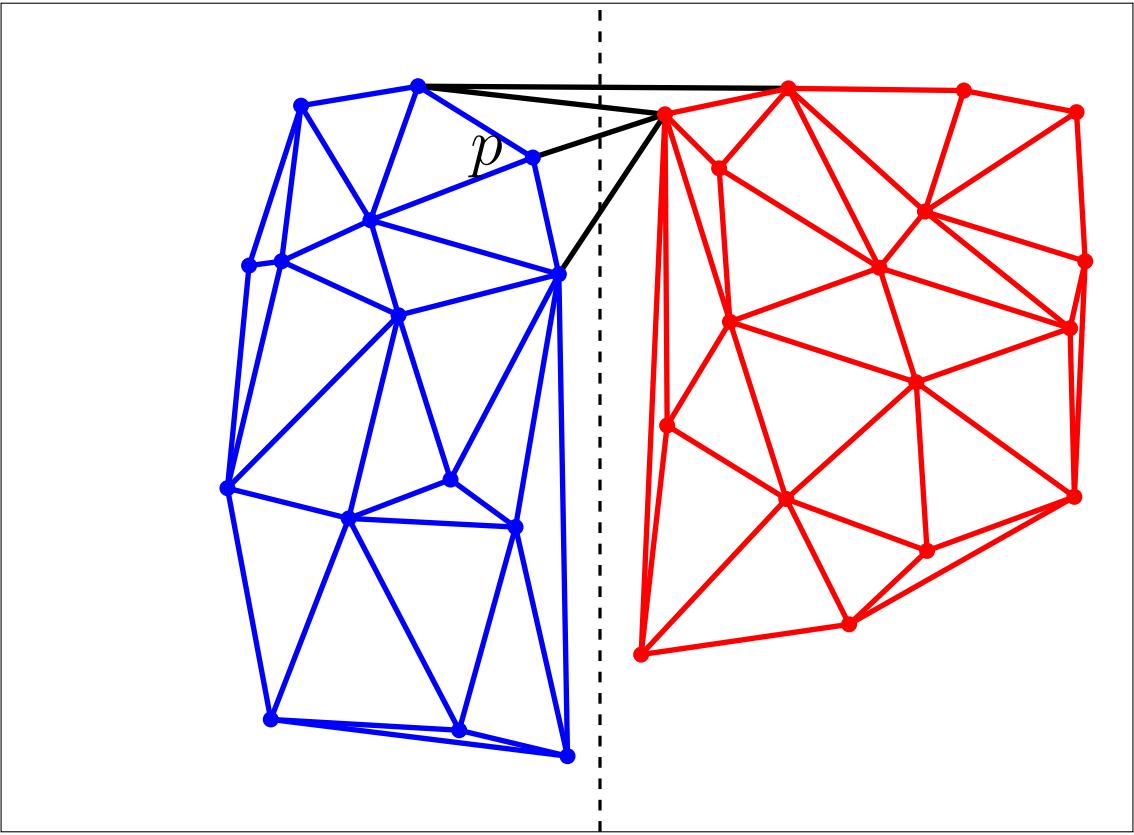

Constructing bi-chromatic edges
from top to bottom
next edge?
rising bubble: set of circumscribed circles

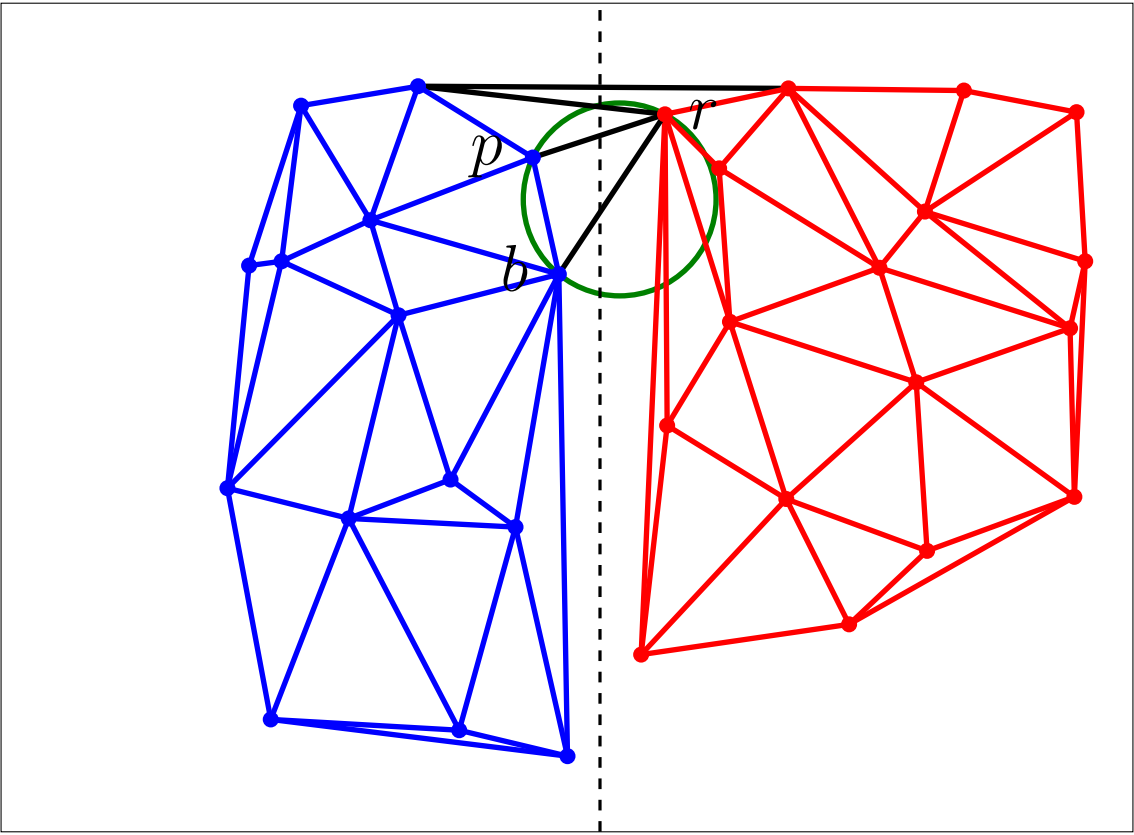

Constructing bi-chromatic edges
from top to bottom
next edge?
rising bubble: set of circumscribed circles

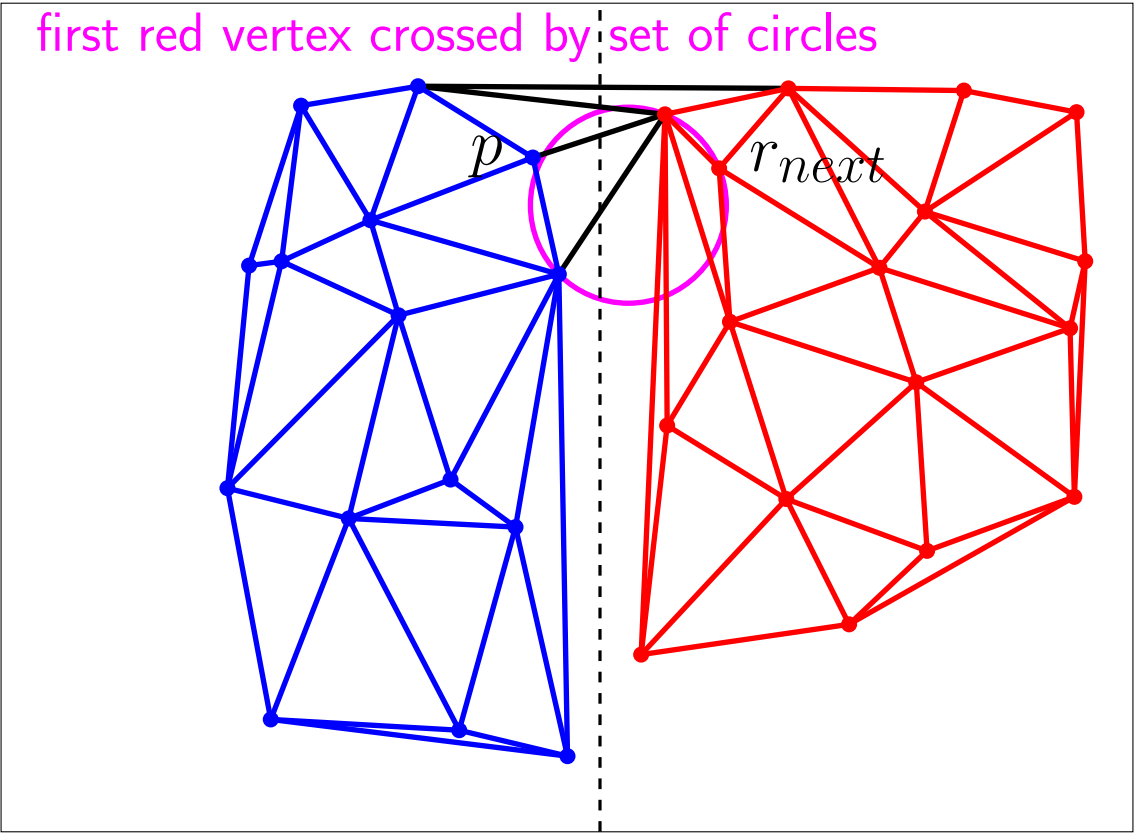

Constructing bi-chromatic edges
from top to bottom
next edge?
rising bubble: set of circumscribed circles

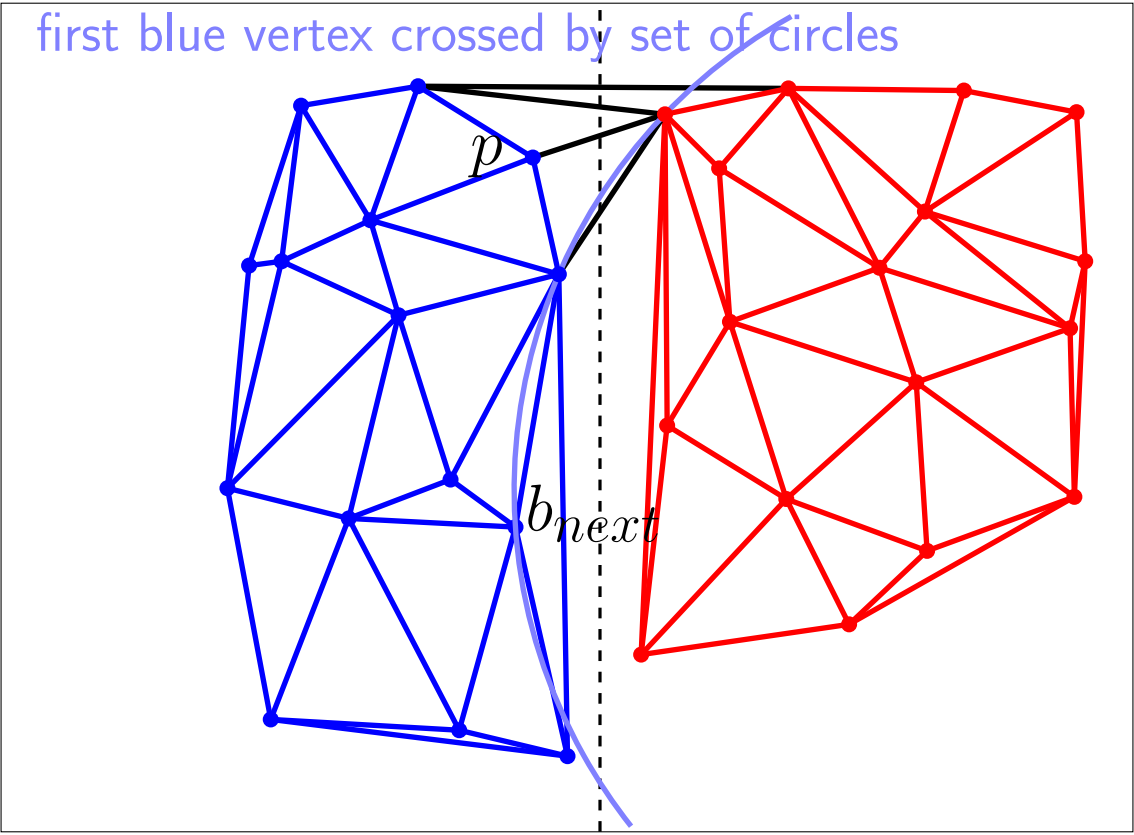

Constructing bi-chromatic edges
from top to bottom
next edge?
rising bubble: set of circumscribed circles

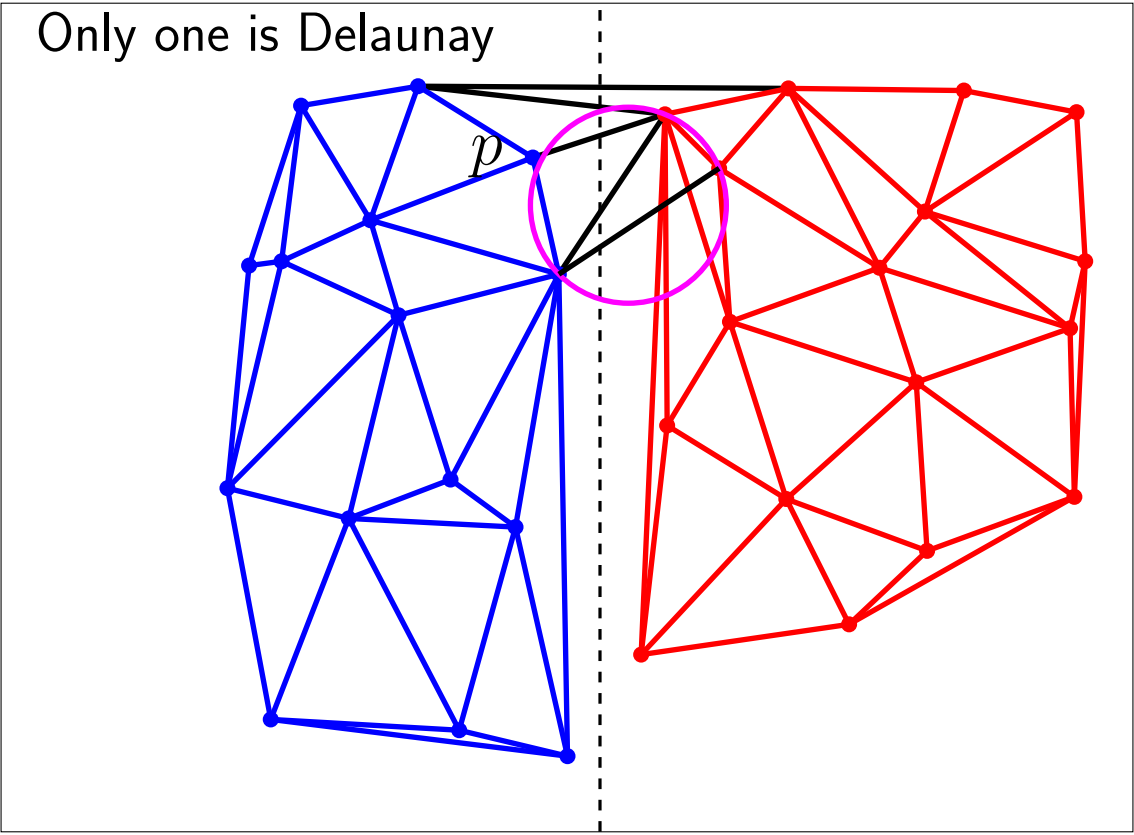

Constructing bi-chromatic edges
from top to bottom
next edge?


p

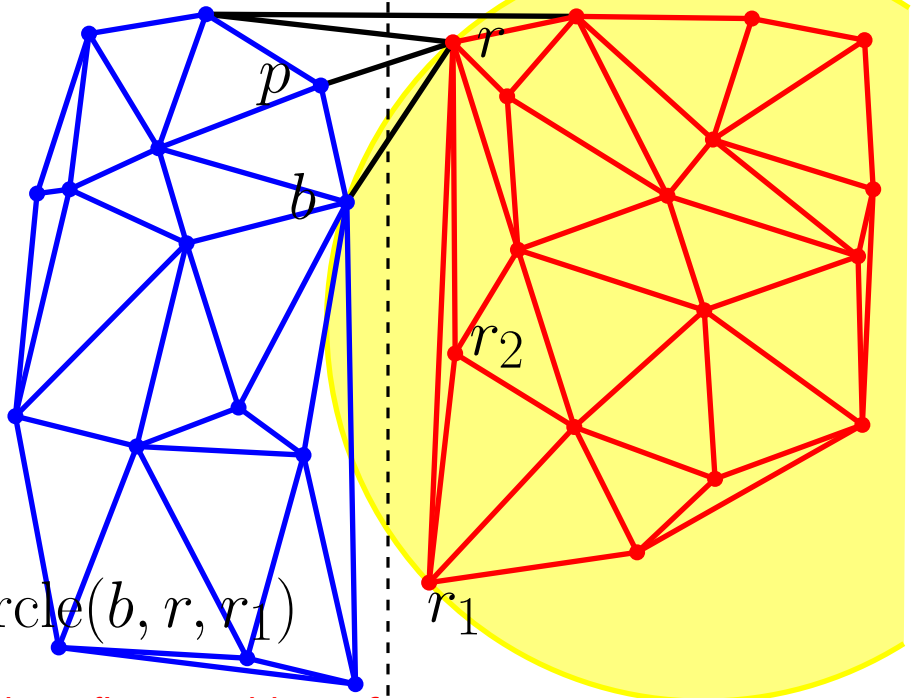

p
r
b


first red vertex crossed by set of circles
p
r_{next}


first blue vertex crossed by set of circles
p
b_{next}


Only one is Delaunay
p

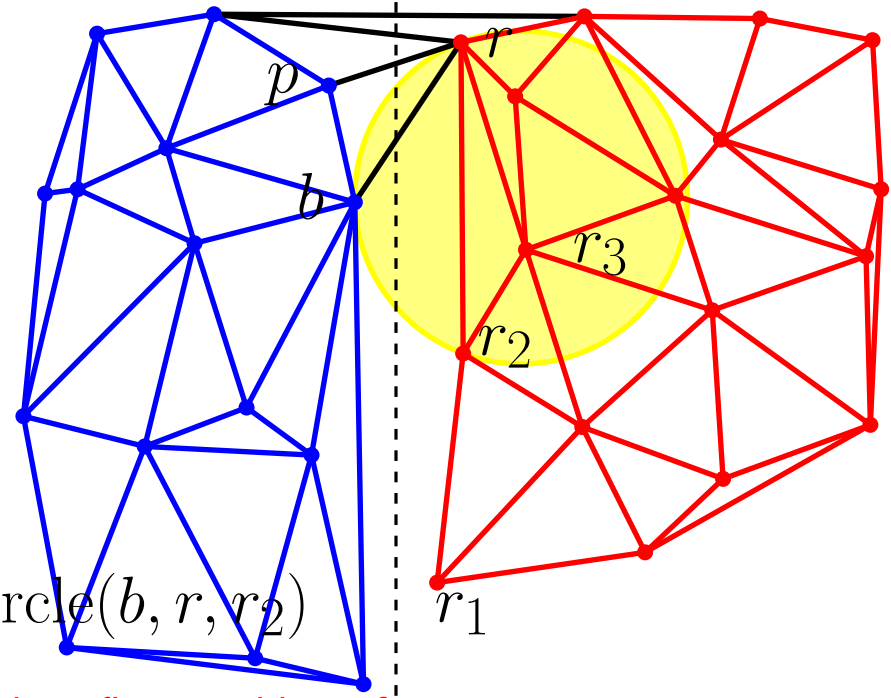
first red vertex crossed by set of circles



$r_2 \in \text{circle}(b, r, r_1)$

Always look at first neighbor of r ccw

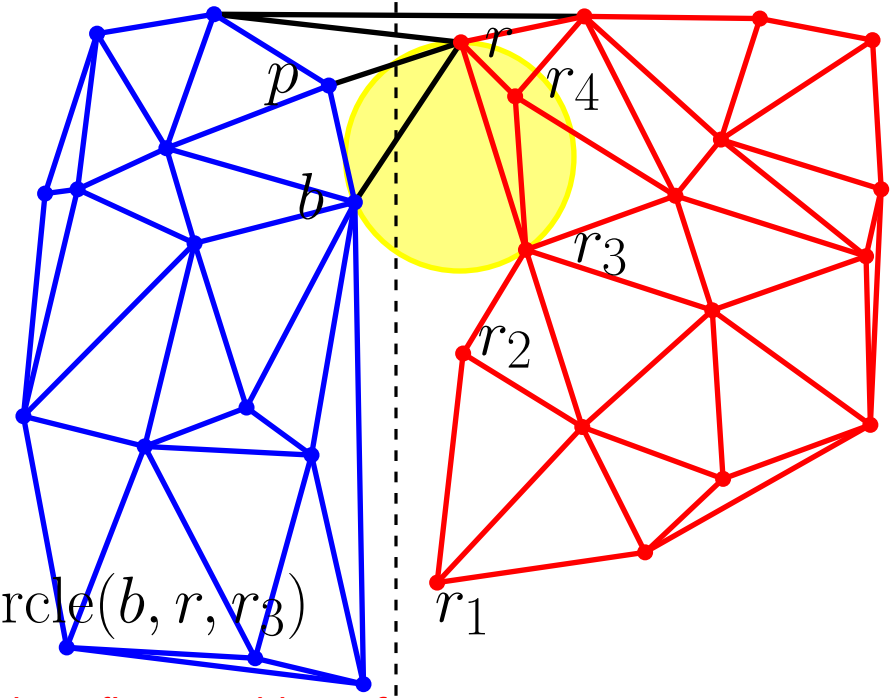
first red vertex crossed by set of circles



$r_3 \in \text{circle}(b, r, r_2)$

Always look at first neighbor of r ccw

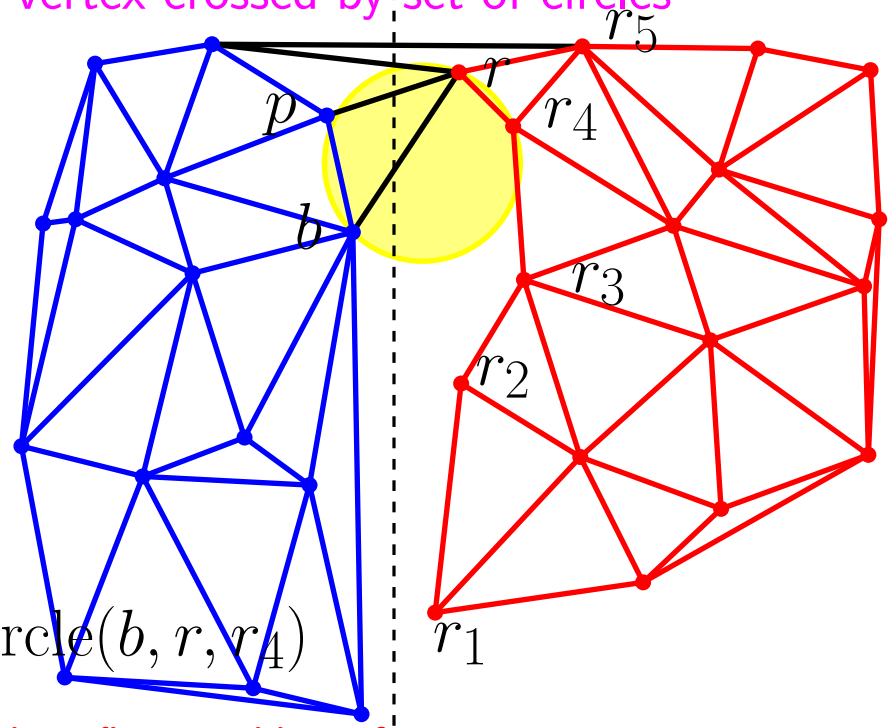
first red vertex crossed by set of circles



$r_4 \in \text{circle}(b, r, r_3)$

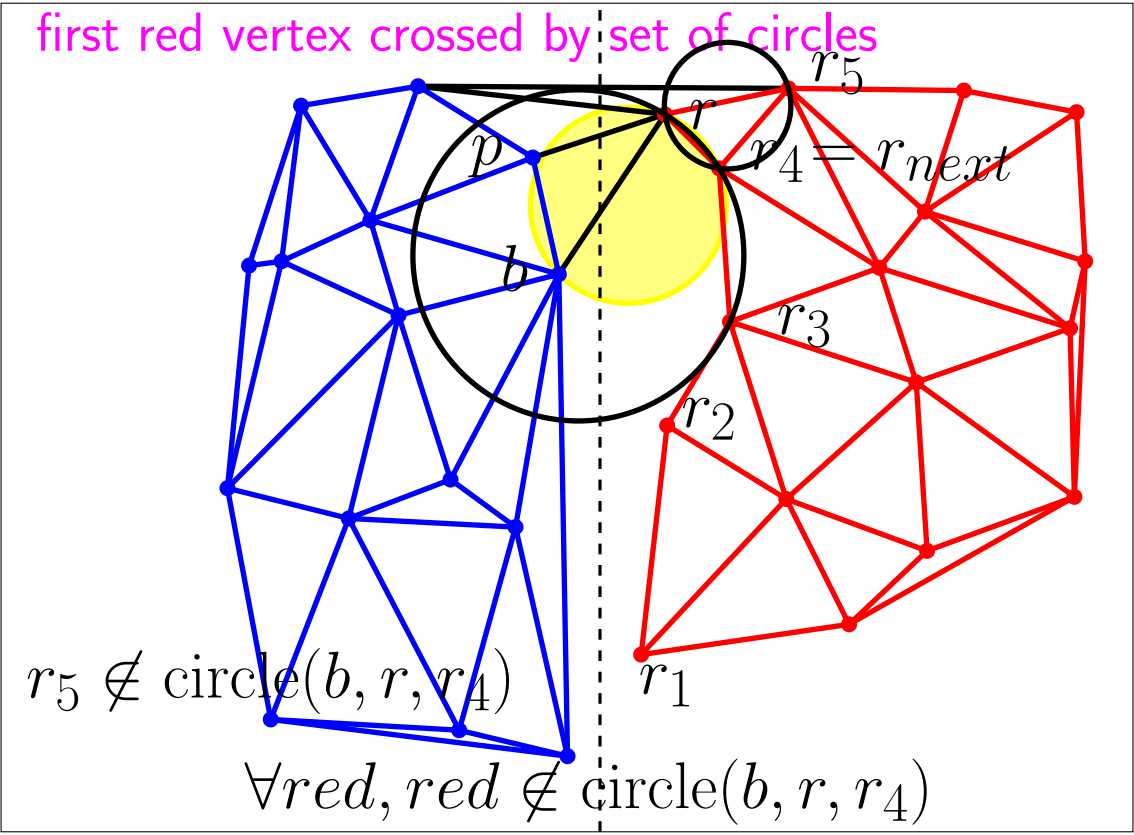
Always look at first neighbor of r ccw

first red vertex crossed by set of circles

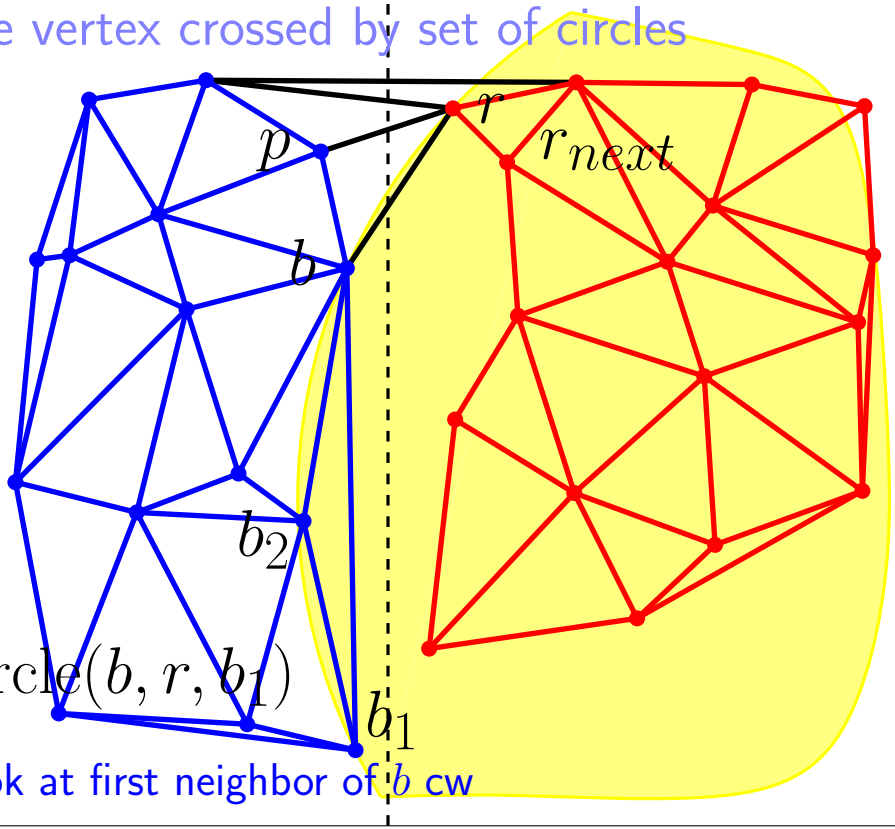


$r_5 \notin \text{circle}(b, r, r_4)$

Always look at first neighbor of r ccw

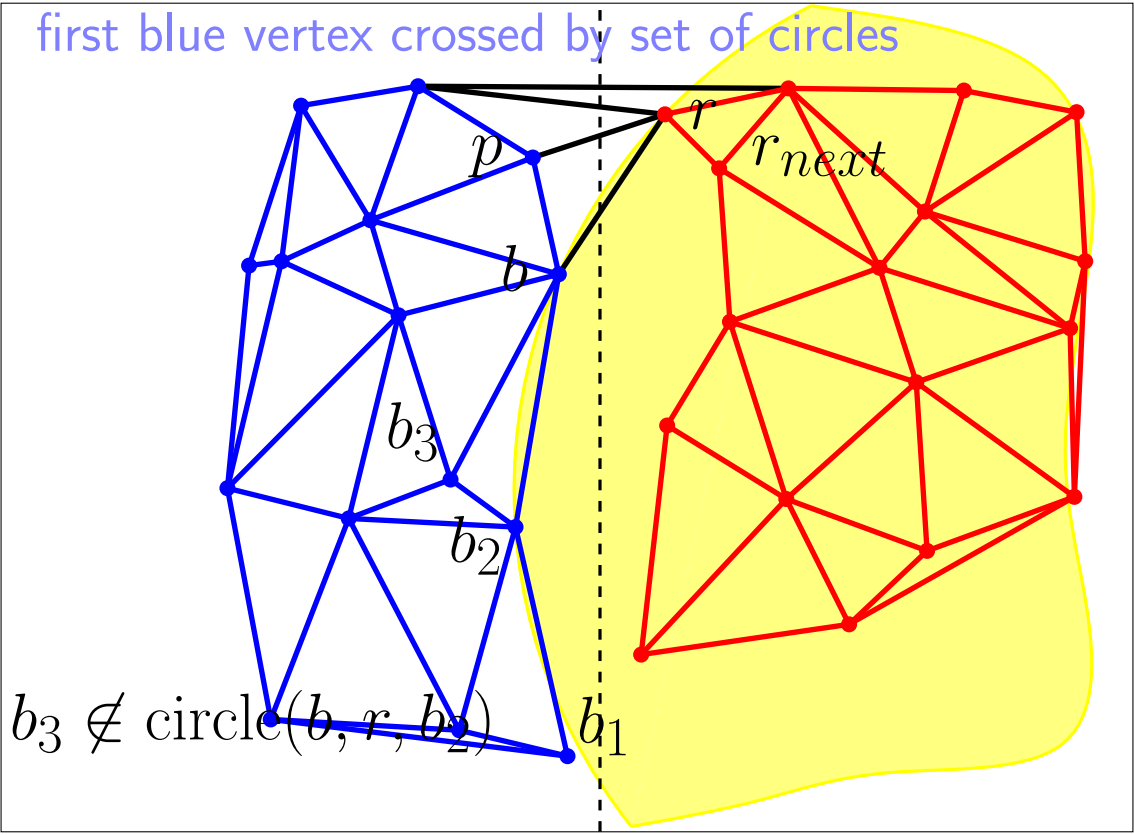

first red vertex crossed by set of circles
r_5
r
p
$r_4 = r_{next}$
b
r_3
r_2
r_1
$r_5 \notin \text{circle}(b, r, r_4)$
$\forall red, red \notin \text{circle}(b, r, r_4)$

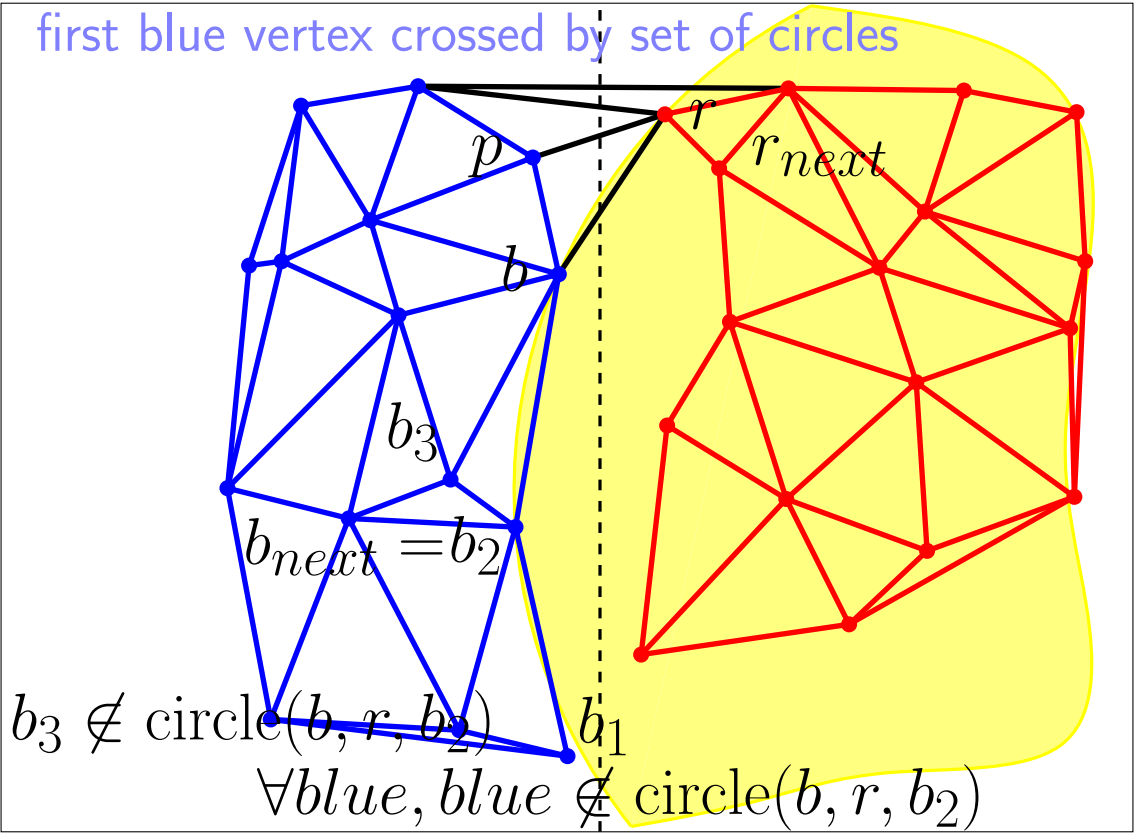
first blue vertex crossed by set of circles

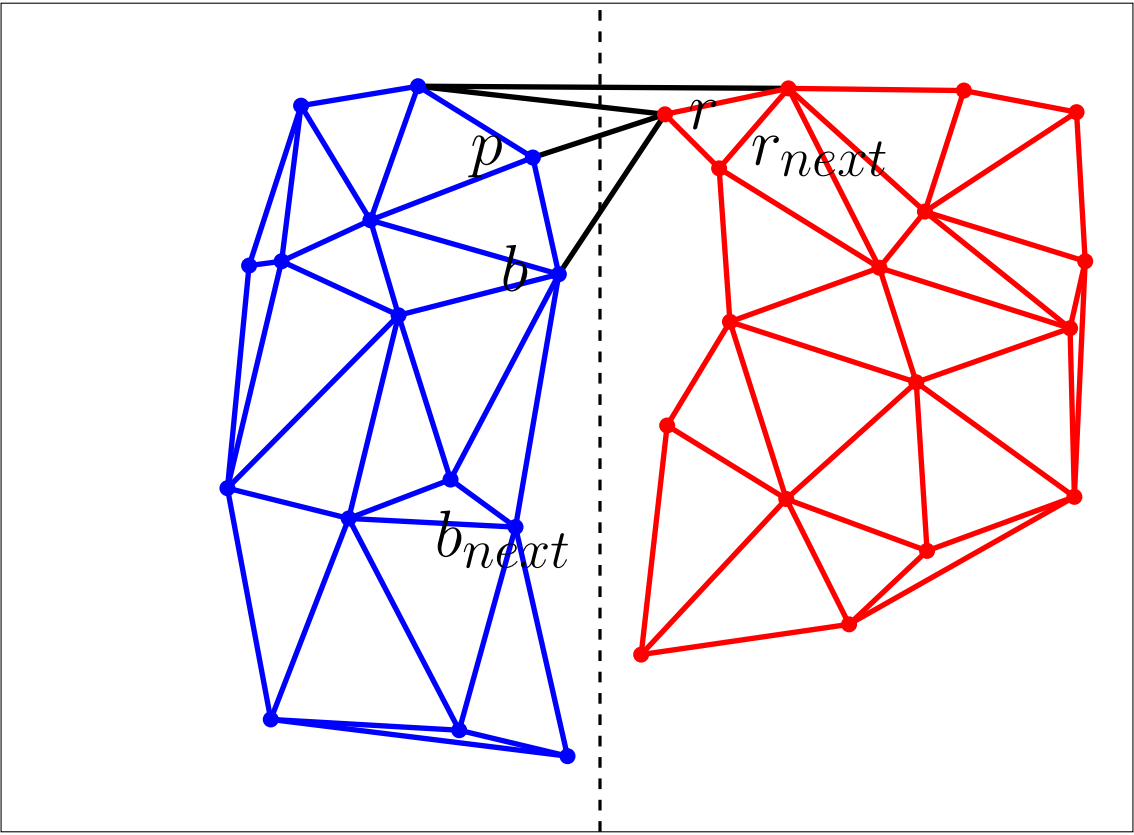


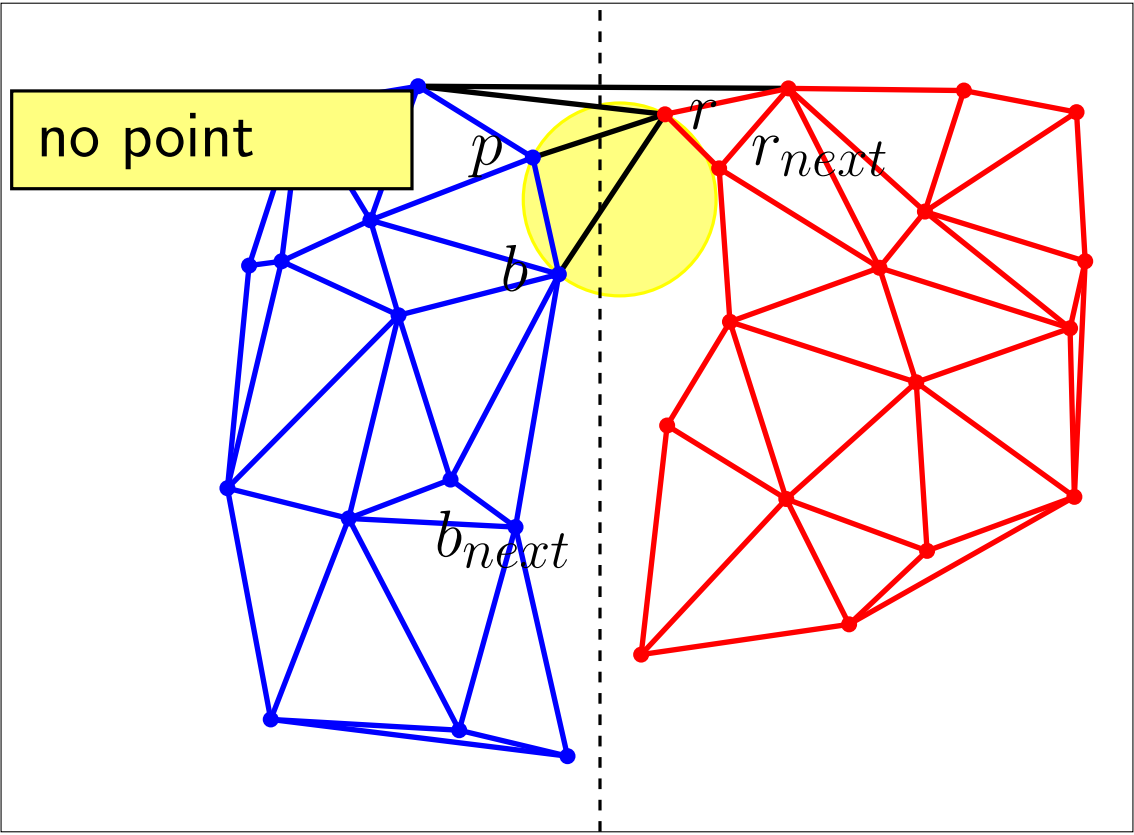
$b_2 \in \text{circle}(b, r, b_1)$

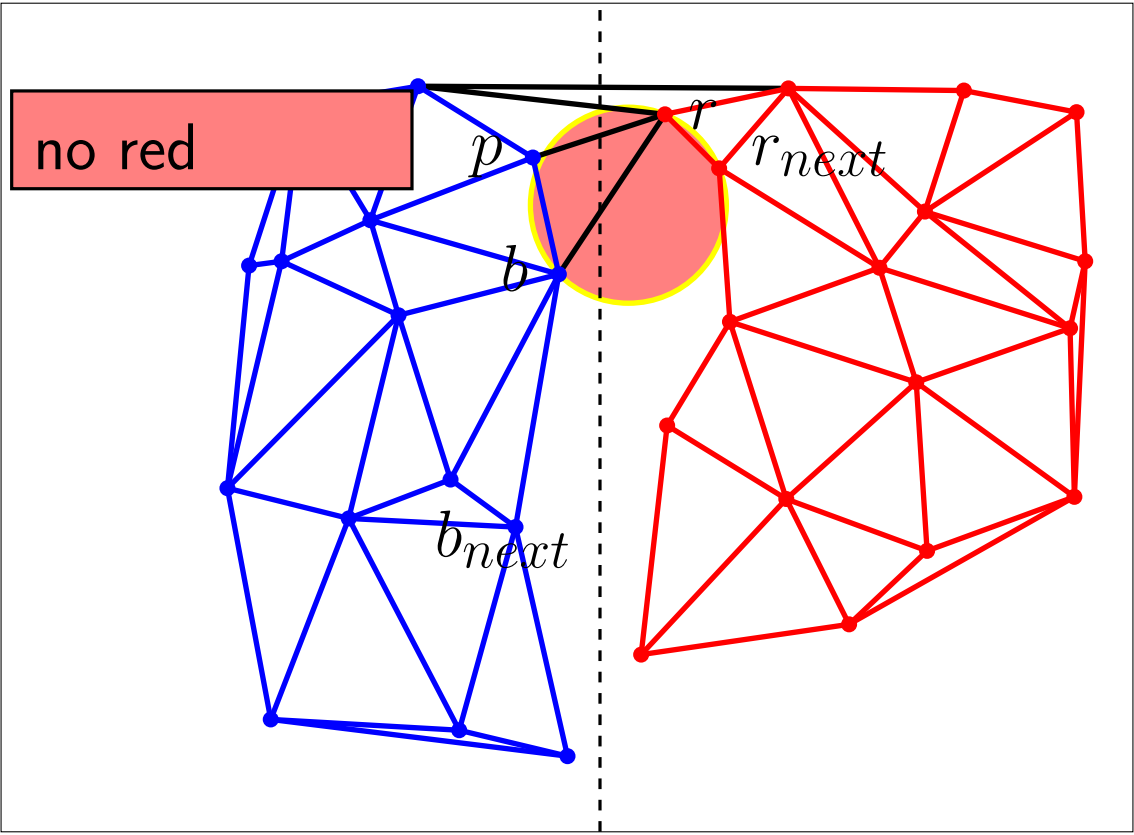
Always look at first neighbor of b cw

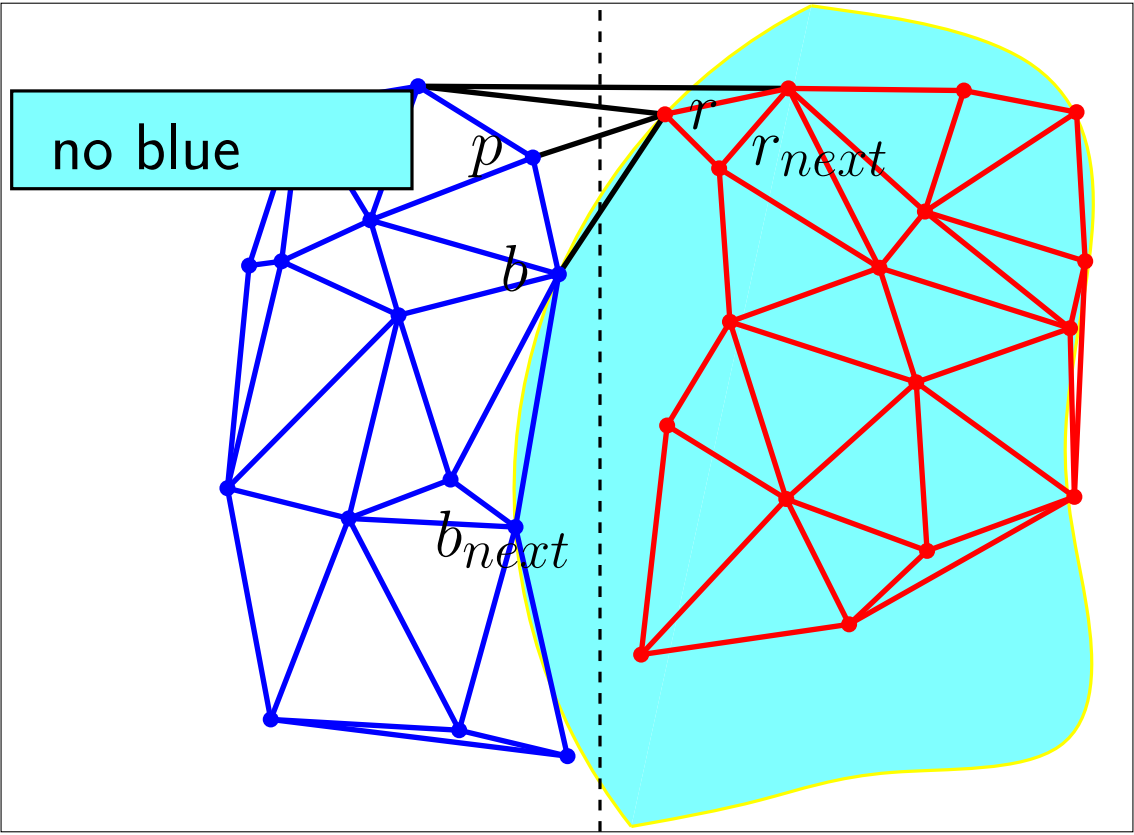

first blue vertex crossed by set of circles
r
r_{next}
p
b
b_3
b_2
b_1
$b_3 \notin \text{circle}(b, r, b_2)$

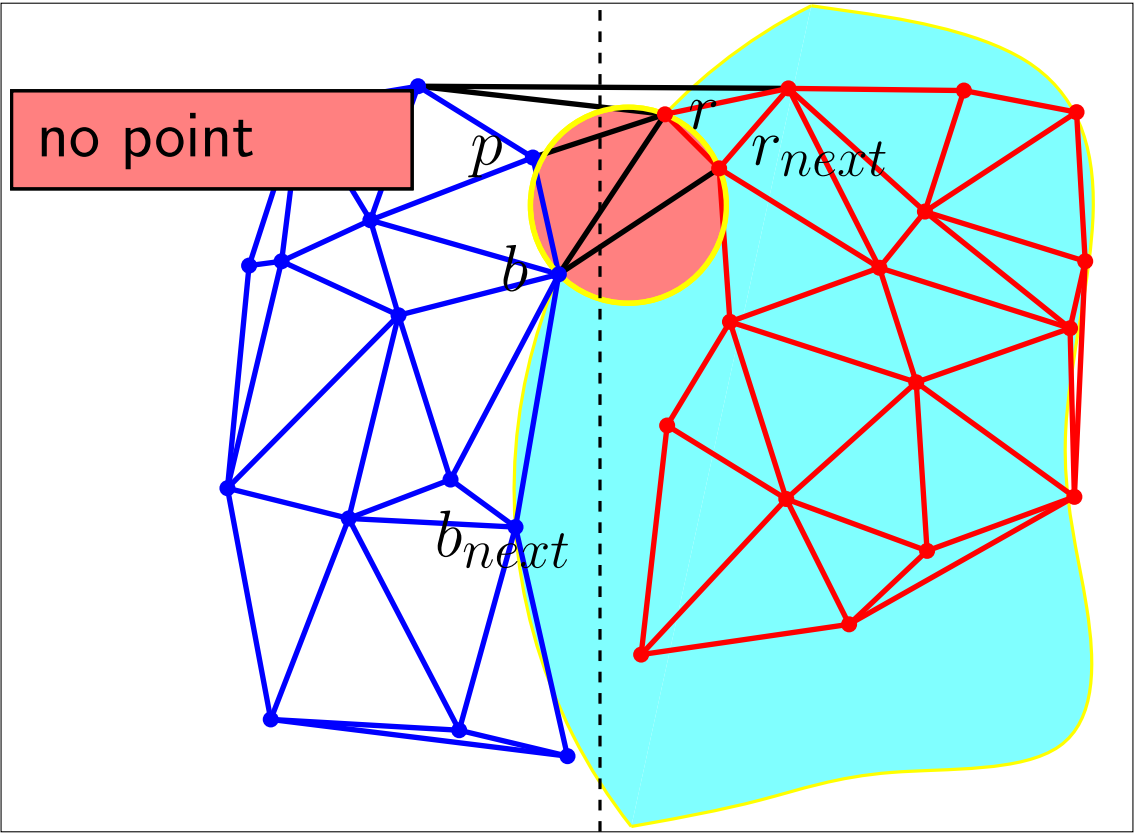

first blue vertex crossed by set of circles
r
r_{next}
p
b
b_3
$b_{next} = b_2$
$b_3 \notin \text{circle}(b, r, b_2)$
b_1
$\forall blue, blue \notin \text{circle}(b, r, b_2)$

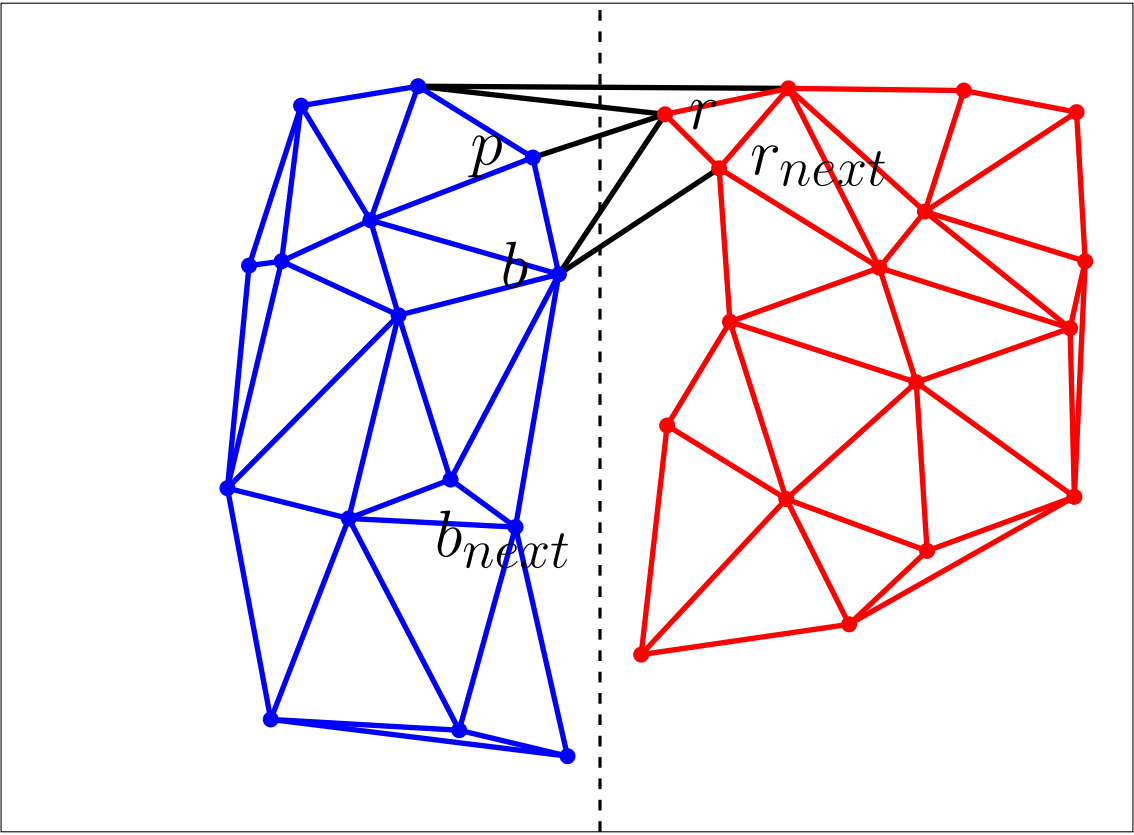

r
r_{next}
p
b
b_{next}

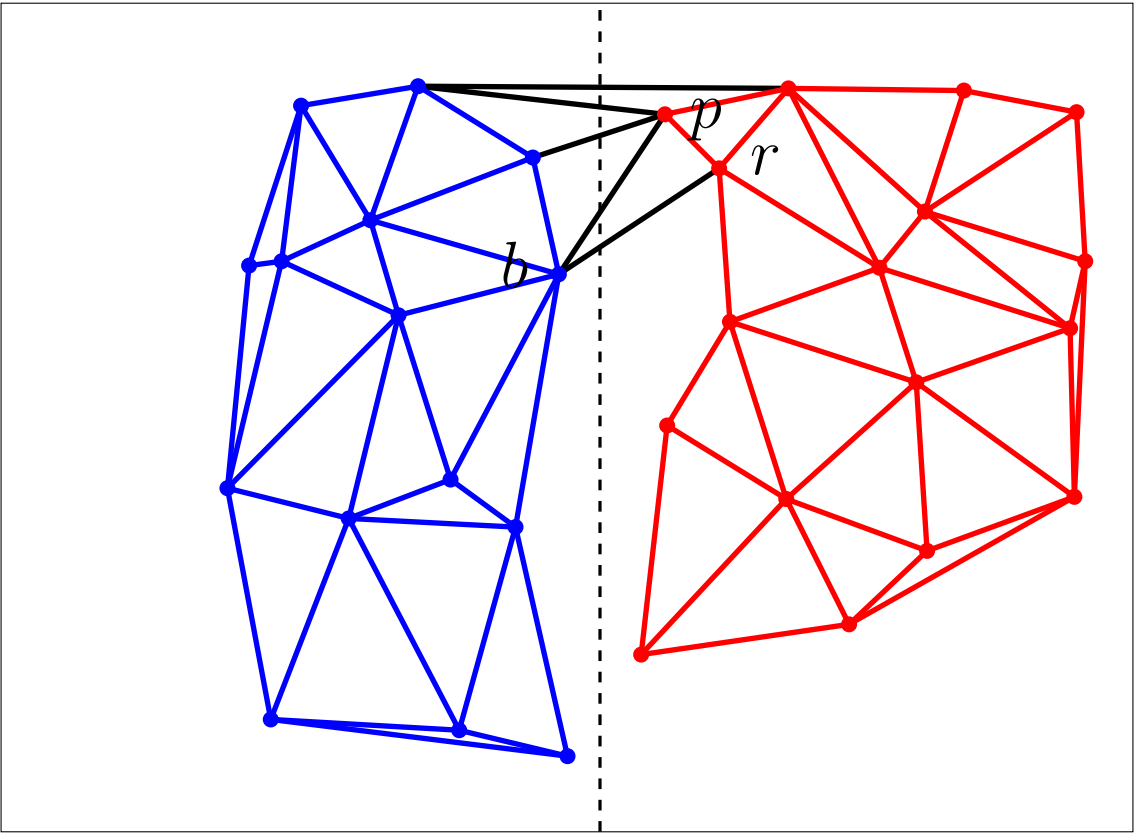

no point
p
r
r_{next}
b
b_{next}

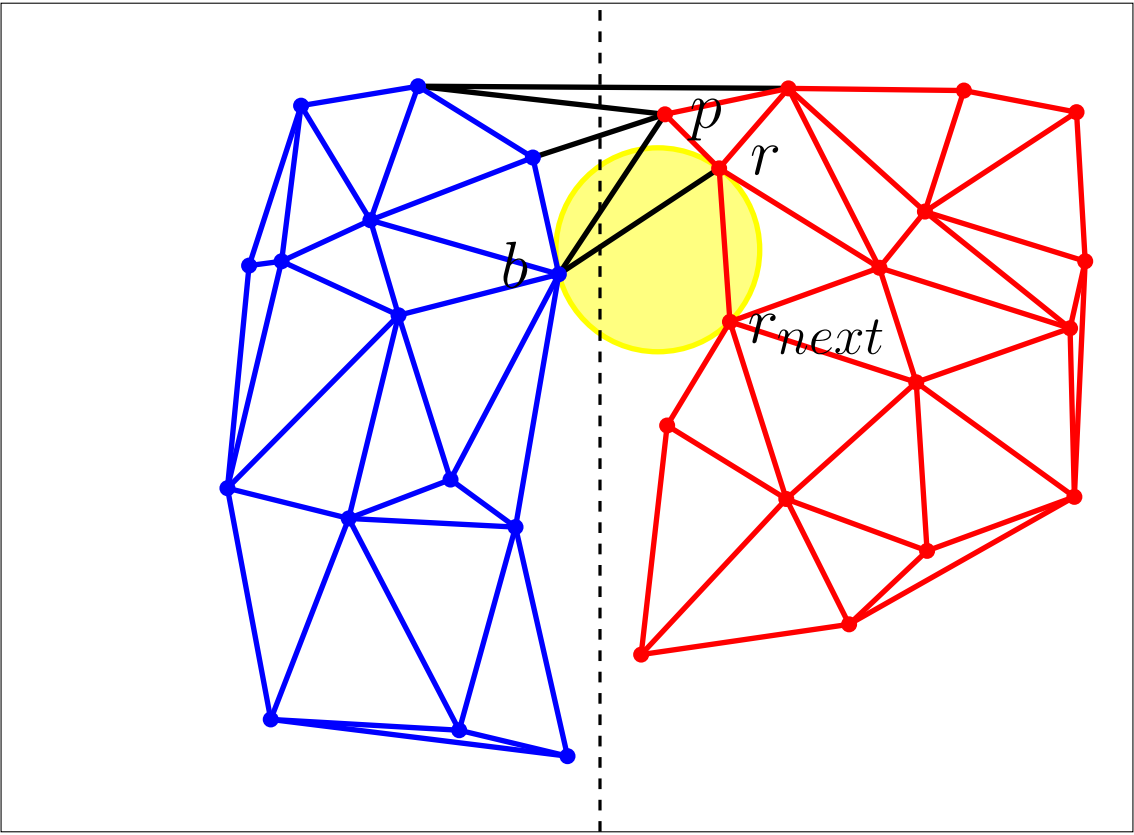

no red
p
r
r_{next}
b
b_{next}

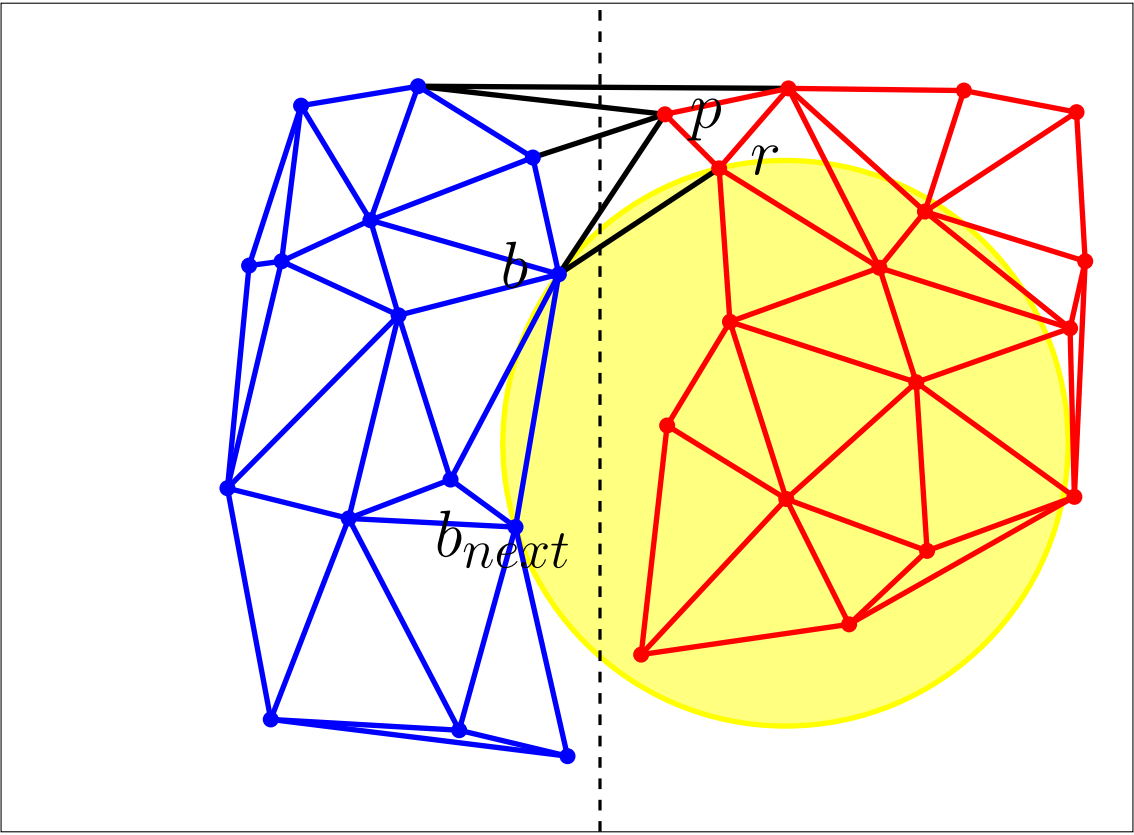

no blue
p
r
r_{next}
b
b_{next}

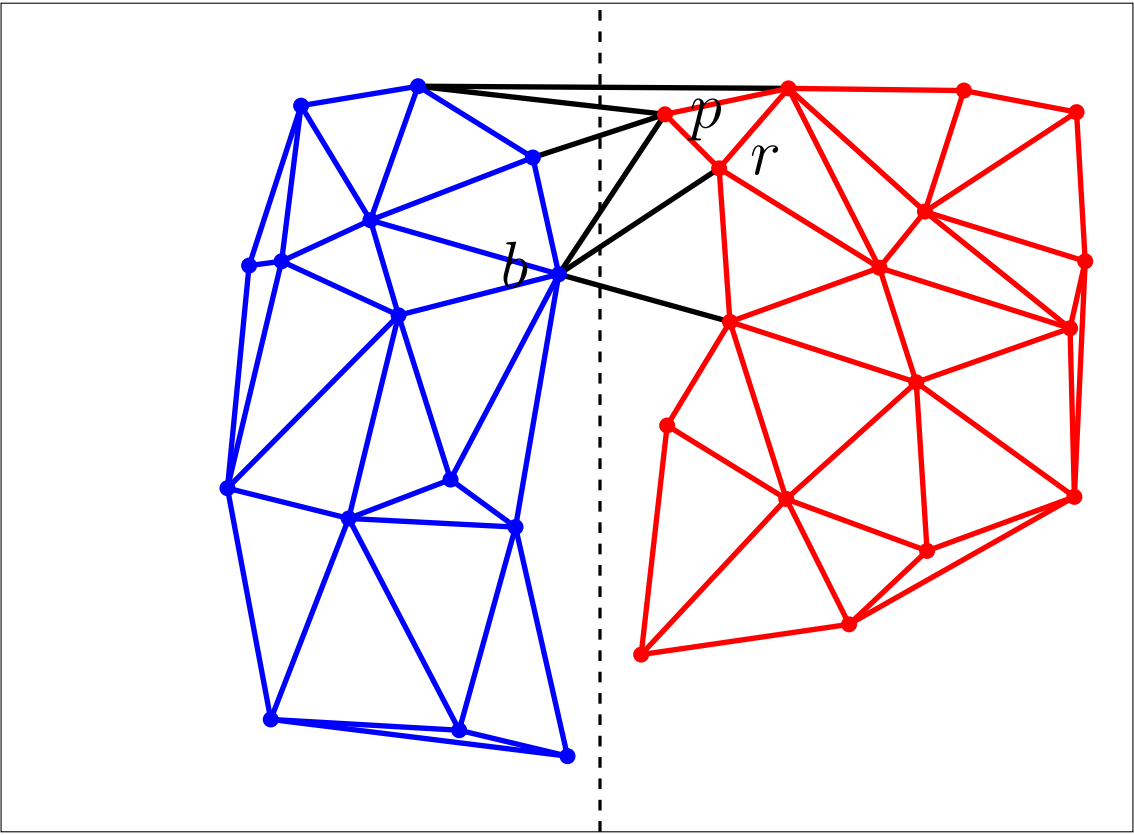

no point
p
r
r_{next}
b
b_{next}

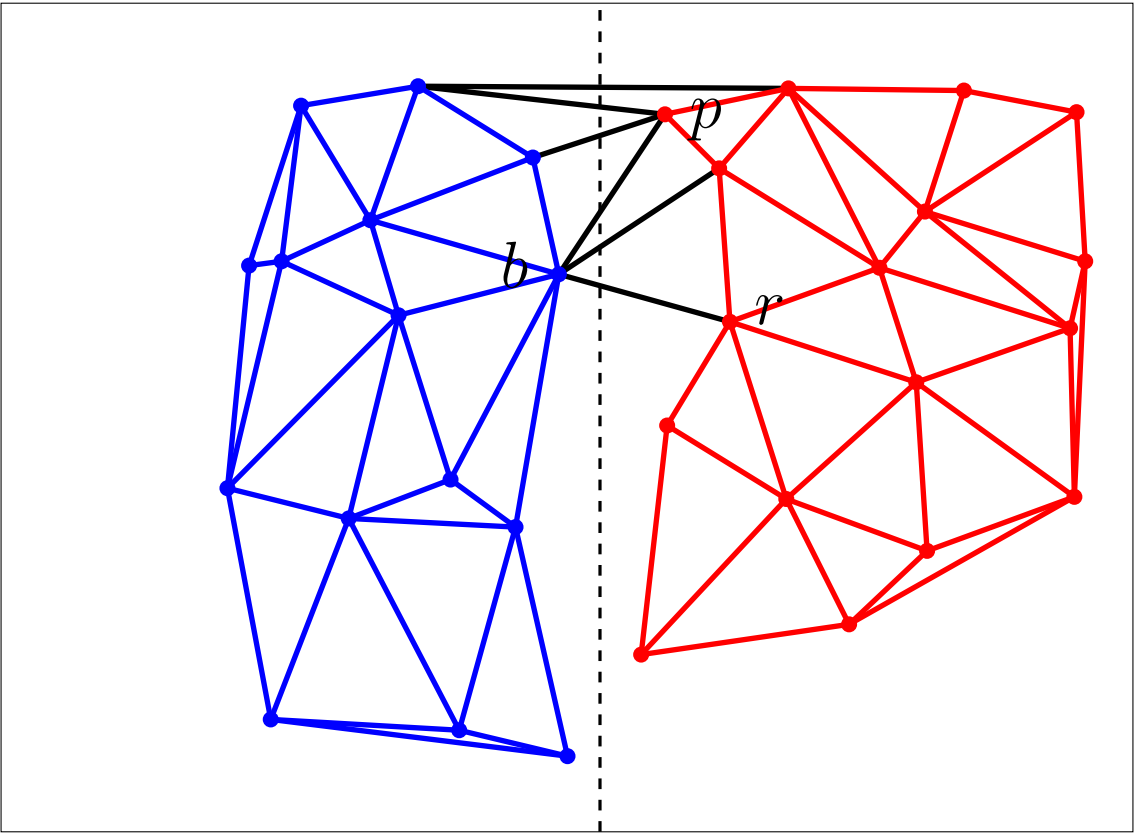

r
p
r_{next}
b
b_{next}

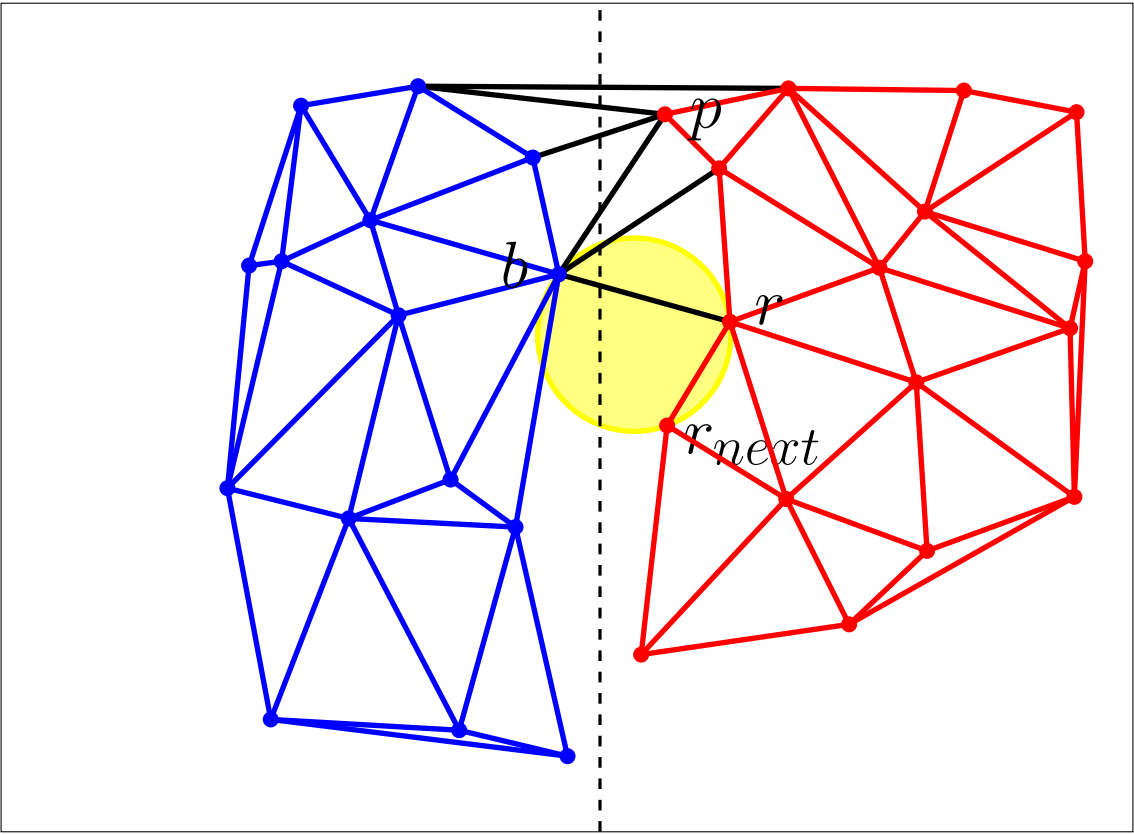

p
r
b

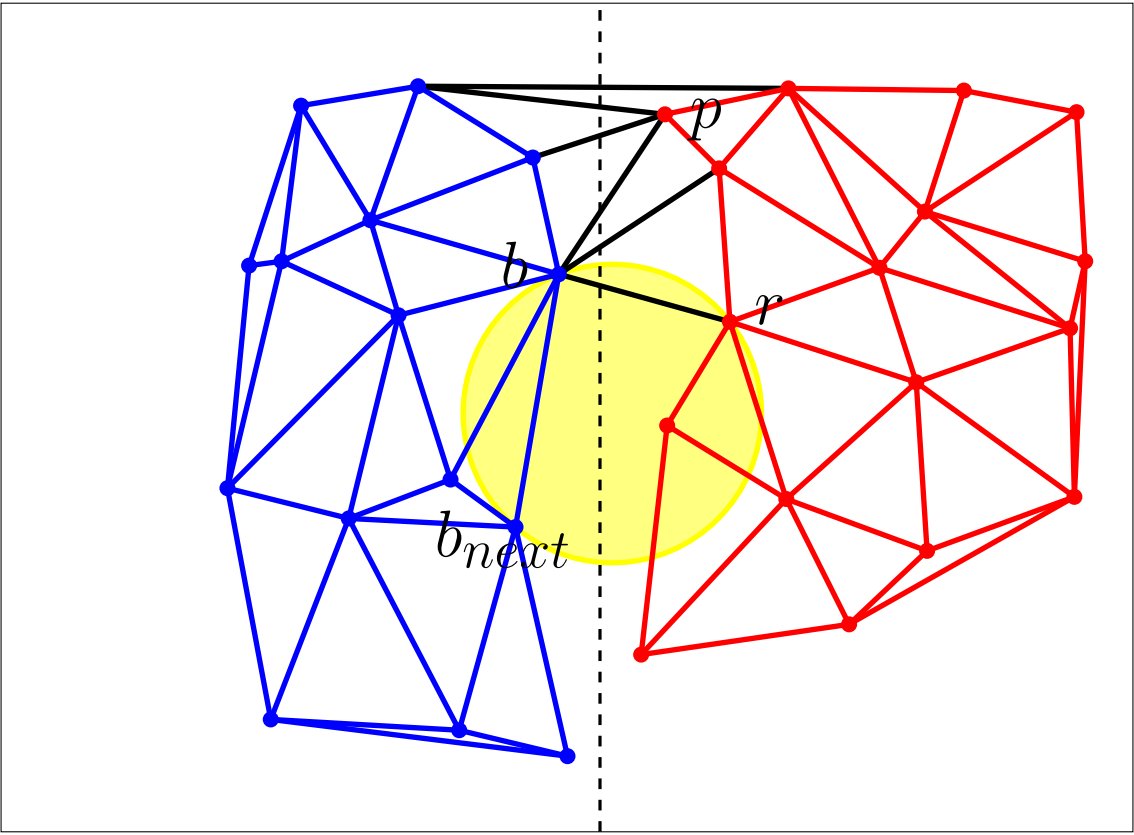

p
r
b
r_{next}

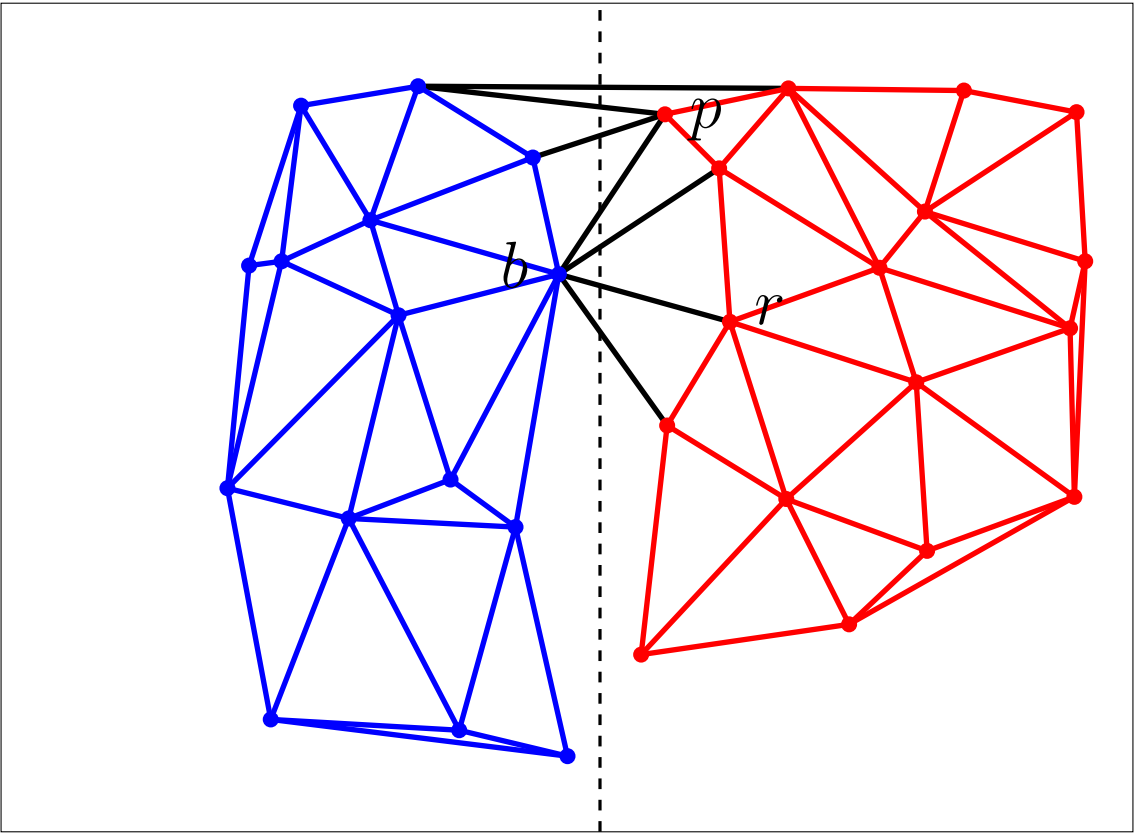

p
r
b
b_{next}

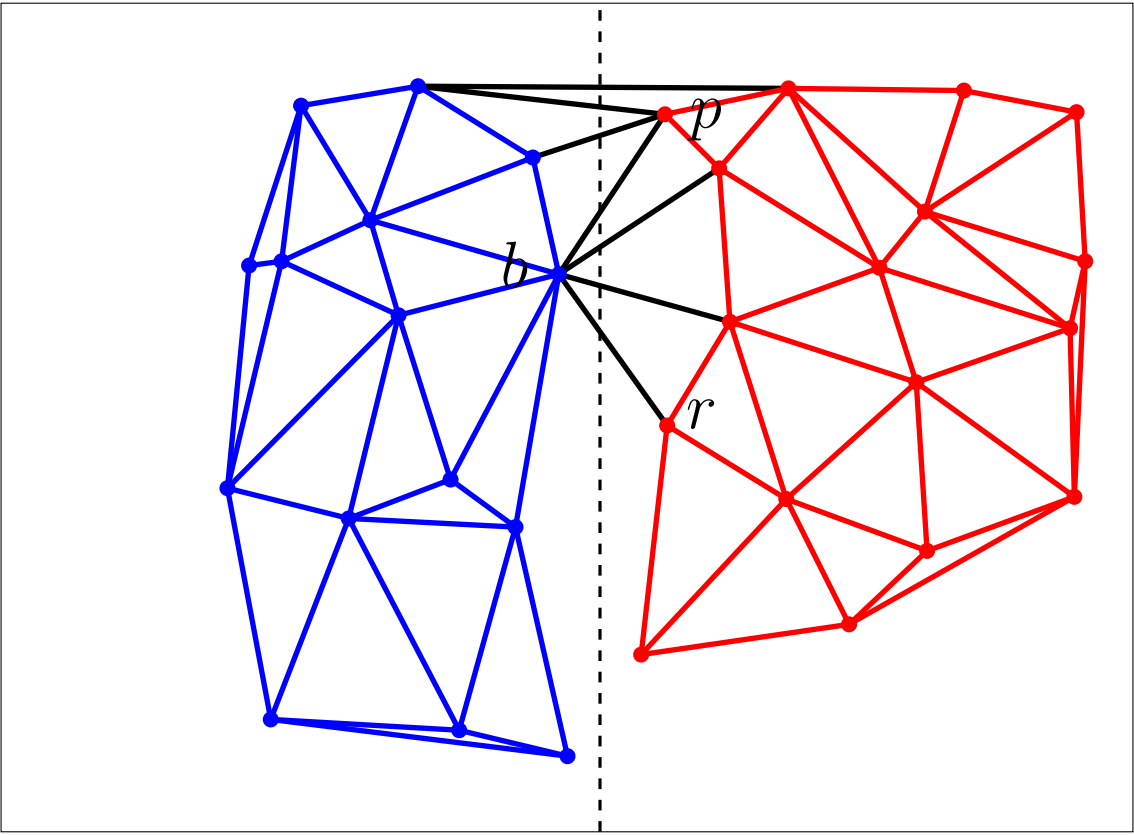

p
r
b

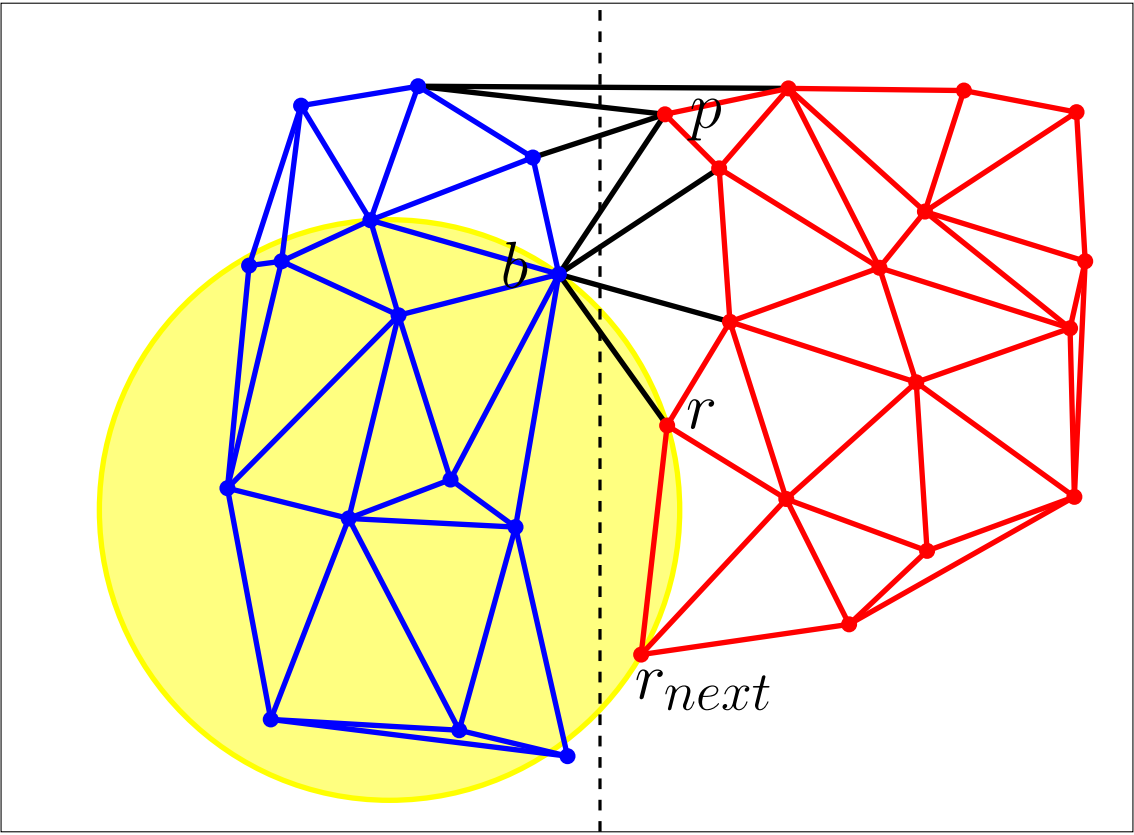

p
b
r

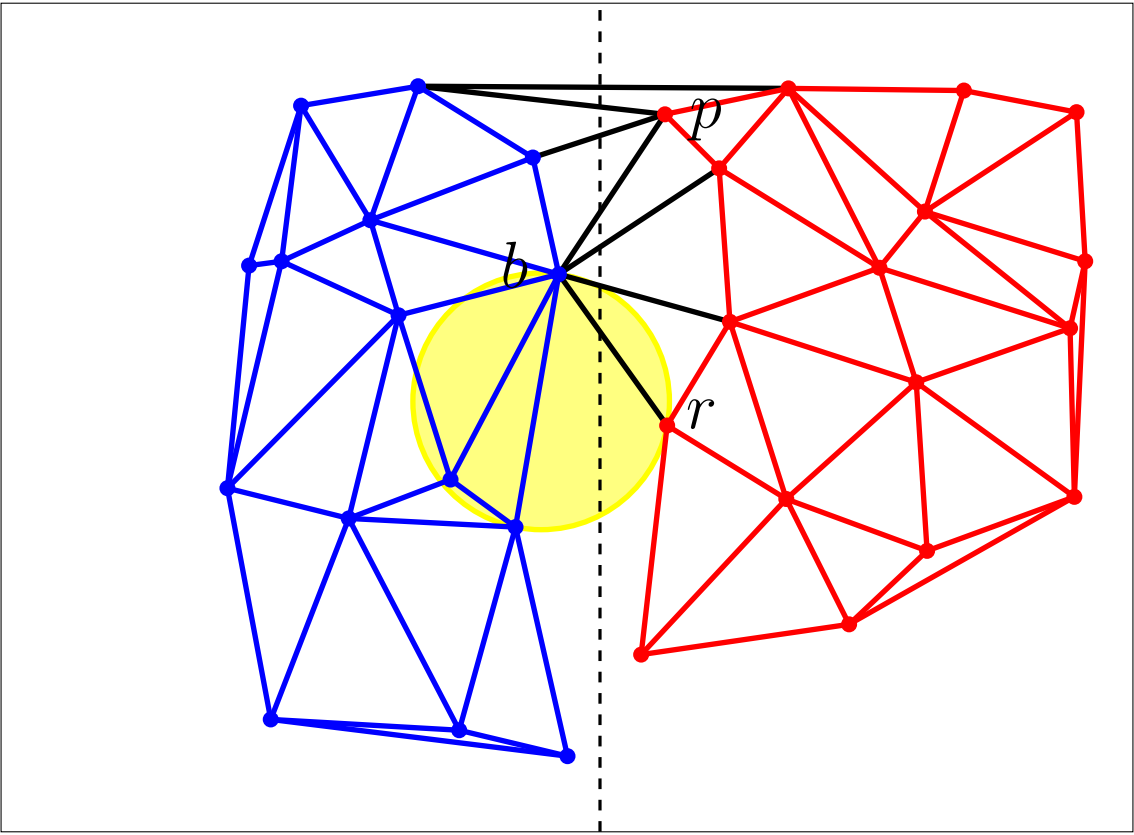

p
b
r
r_{next}

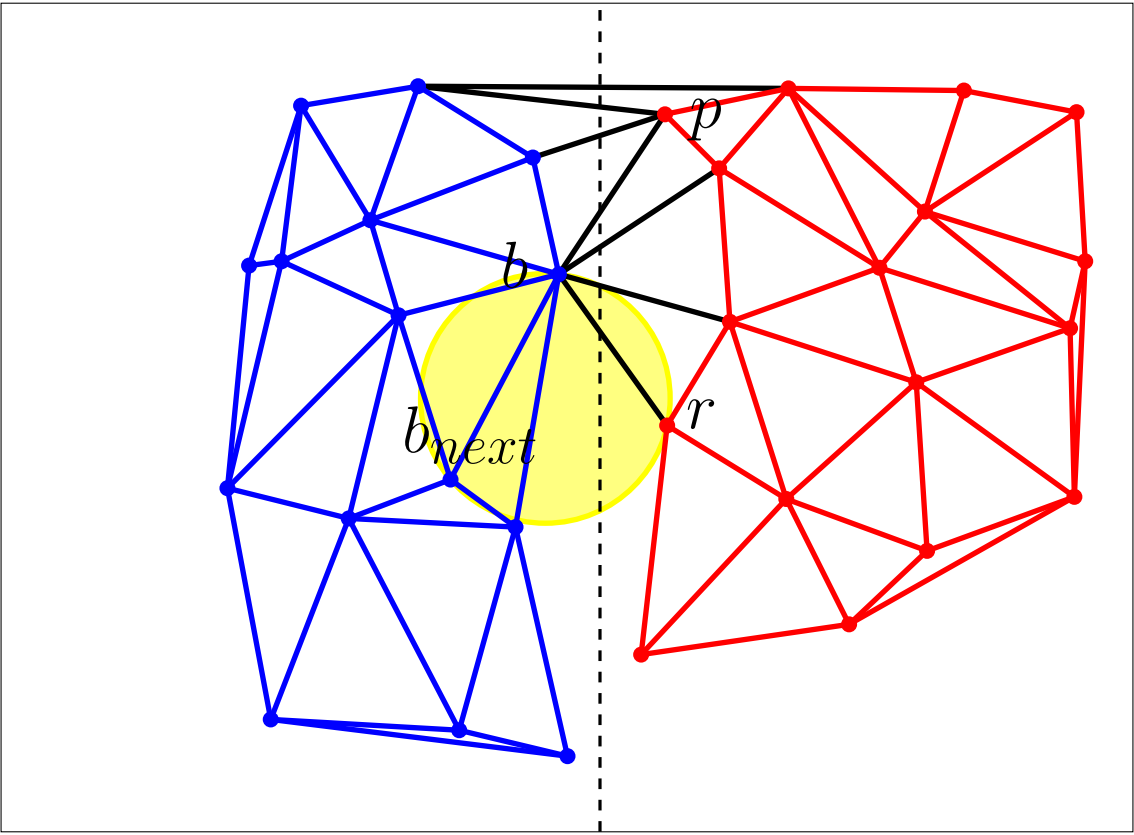

p
b
r
b_{next}

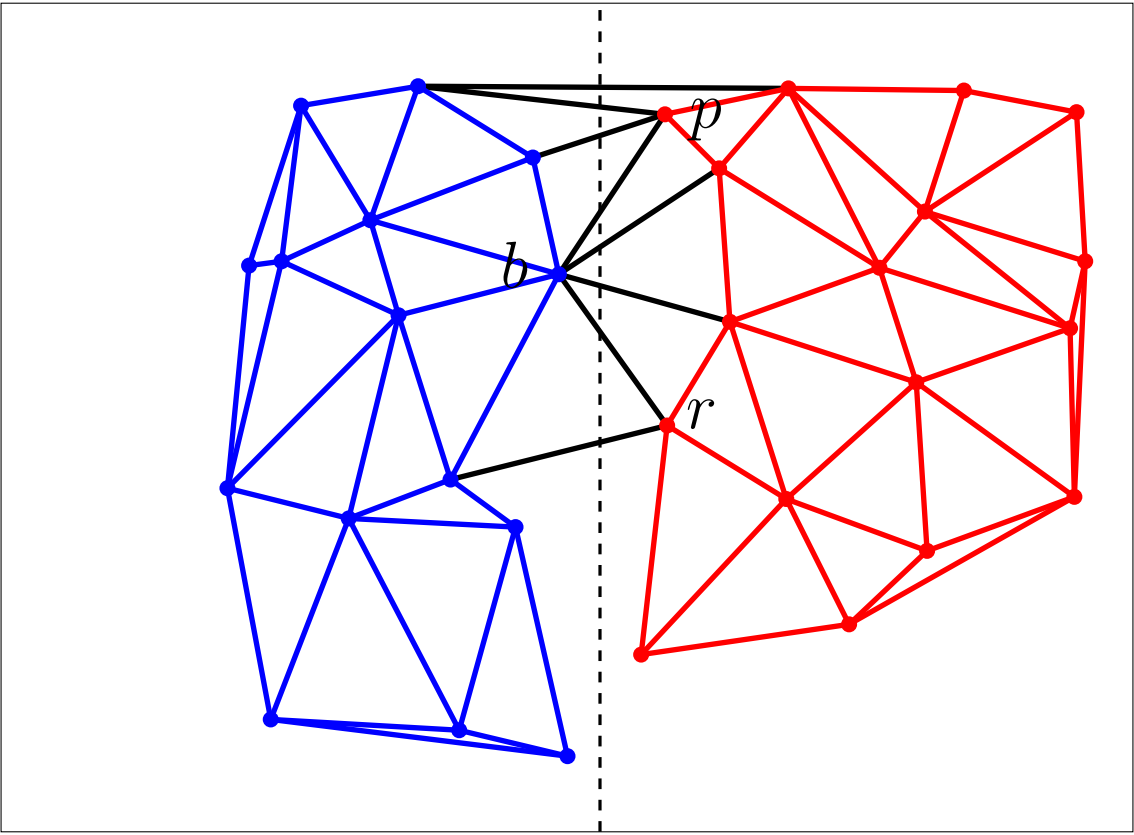

p
b
r

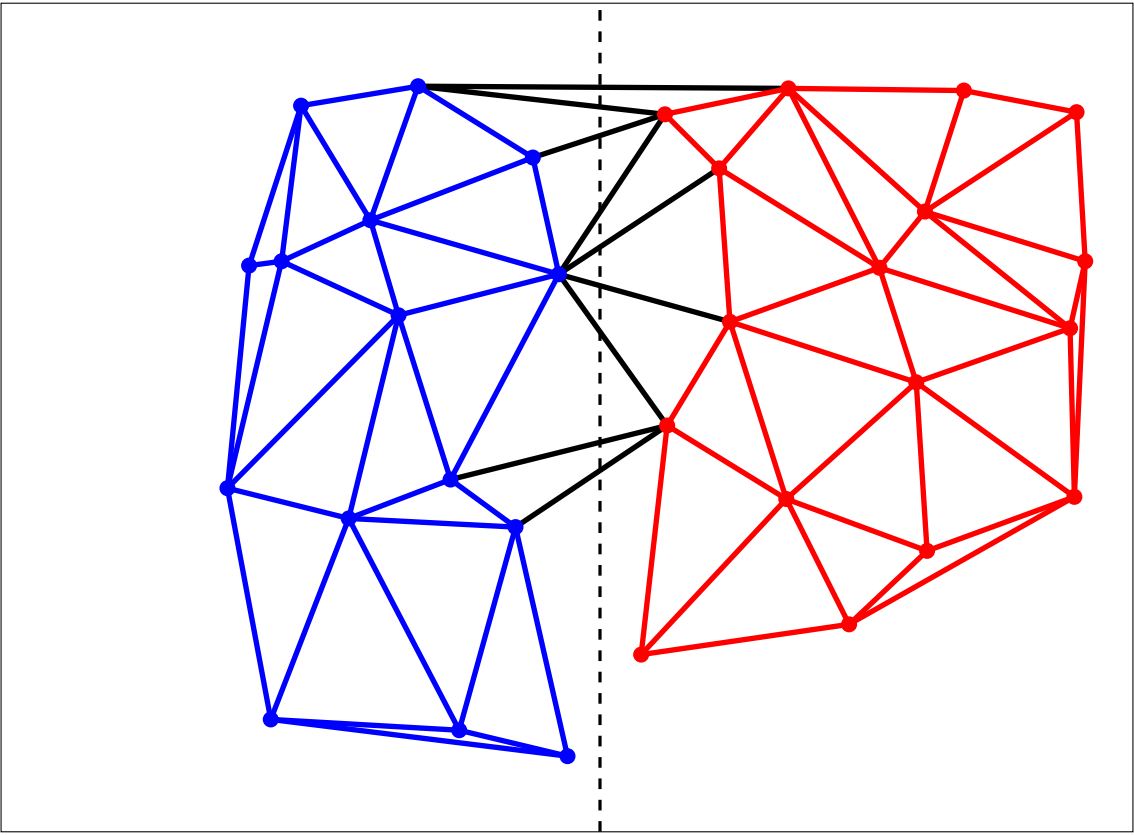

p
b
r

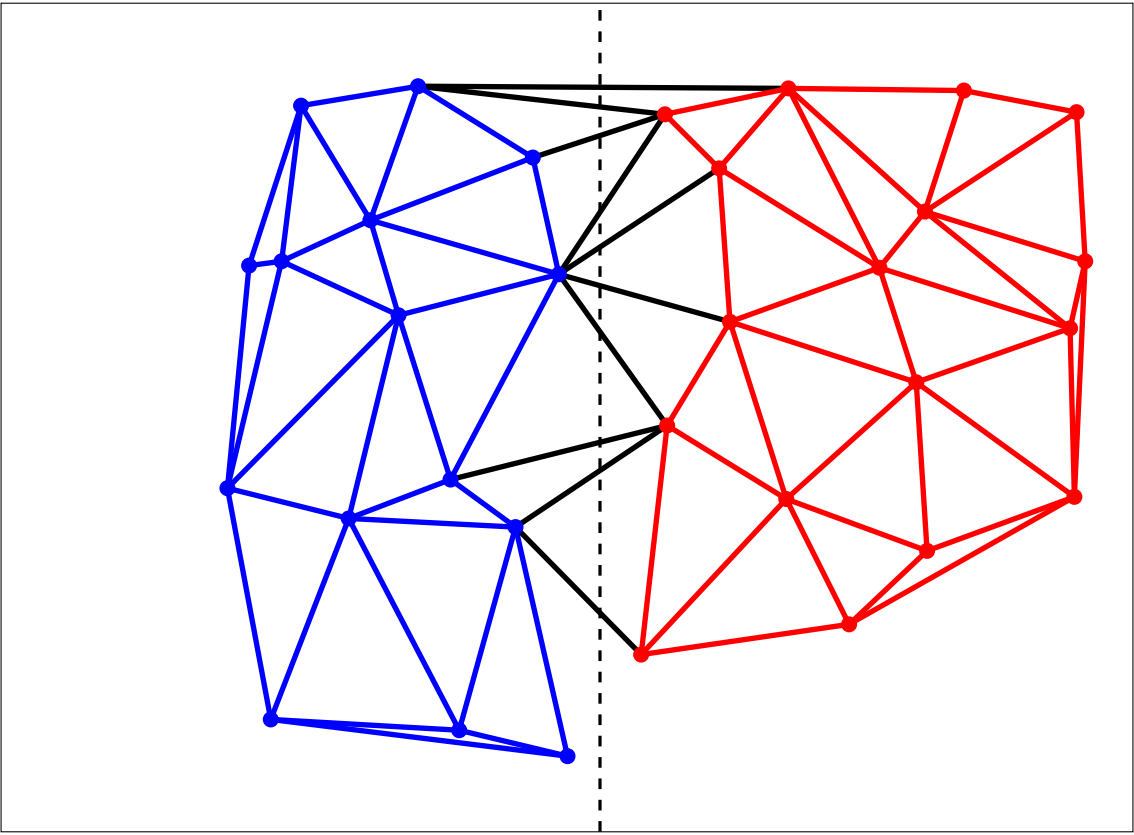

p
b
r
r_{next}

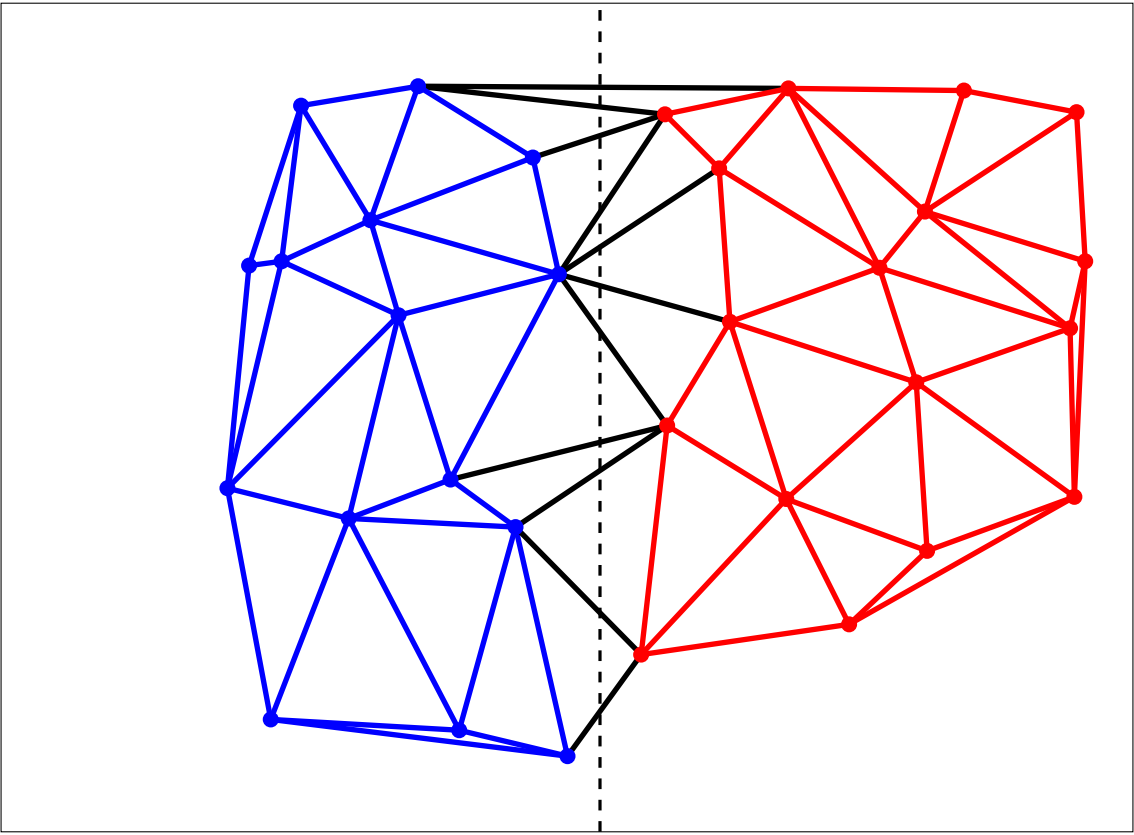

p
b
r

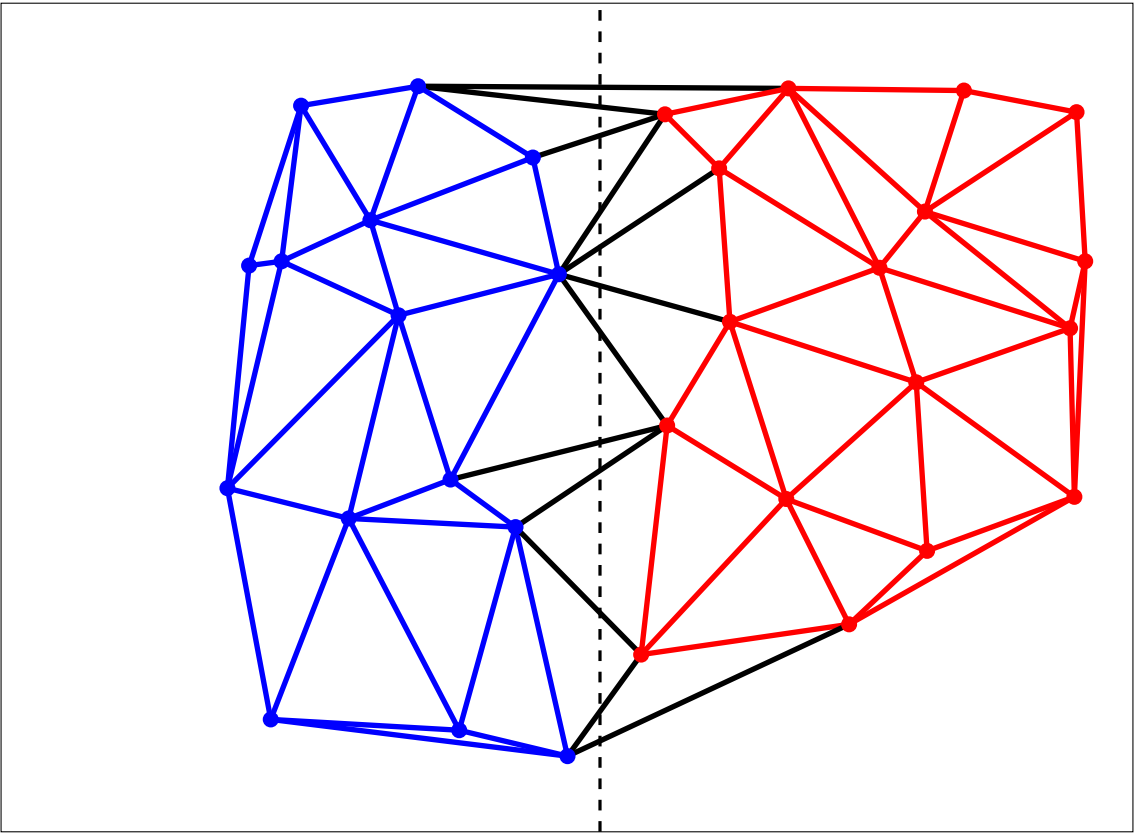

p
b
r
b_{next}


p
b
r









Complexity of Fusion

Complexity of Fusion

At each step of the search for r_{next}

Complexity of Fusion

At each step of the search for r_{next}

A red edge is deleted

Complexity of Fusion

At each step of the search for r_{next}

A red edge is deleted

At each step of the search for b_{next}

Complexity of Fusion

At each step of the search for r_{next}

A red edge is deleted

At each step of the search for b_{next}

A blue edge is deleted

Complexity of Fusion

At each step of the search for r_{next}

A red edge is deleted

At each step of the search for b_{next}

A blue edge is deleted

After the choice between r_{next} and b_{next}

Complexity of Fusion

At each step of the search for r_{next}

A red edge is deleted

At each step of the search for b_{next}

A blue edge is deleted

After the choice between r_{next} and b_{next}

A black edge is created

Complexity of Fusion

Complexity $\leq$ $\sharp$ red edges
$+\sharp$ blue edges
$+\sharp$ black edges

Complexity of Fusion

$$\text{Complexity} \leq \begin{array}{l} \sharp \text{ red edges} \\ +\sharp \text{ blue edges} \\ +\sharp \text{ black edges} \end{array}$$

$$\leq 3\frac{n}{2} + 3\frac{n}{2} + 3n = O(n)$$

Overall Complexity

Division $= O(n)$ for any sub-problem

$+ O(n \log n)$ (preprocessing)

Fusion $= O(n)$ on sub-pb of size n

Division-Fusion $\Longrightarrow O(n \log n)$

Generalizations

Voronoi diagram

Voronoi diagram

Q Nearest neighbor of q among $\mathcal{S}$

Voronoi diagram

Q Nearest neighbor of q among $\mathcal{S}$

Change

Voronoi diagram

Q Nearest neighbor of q among $\mathcal{S}$

Change

ambient space (for q)

$\mathbb{R}^2$ $\mathbb{R}^3$ $\mathbb{R}^d$

Voronoi diagram

Q Nearest neighbor of q among $\mathcal{S}$

Change

metrics

Euclidean L_2

L_1, L_∞, L_p

hyperbolic

additive weights

multiplicative weights

Voronoi diagram

Q Nearest neighbor of q among $\mathcal{S}$

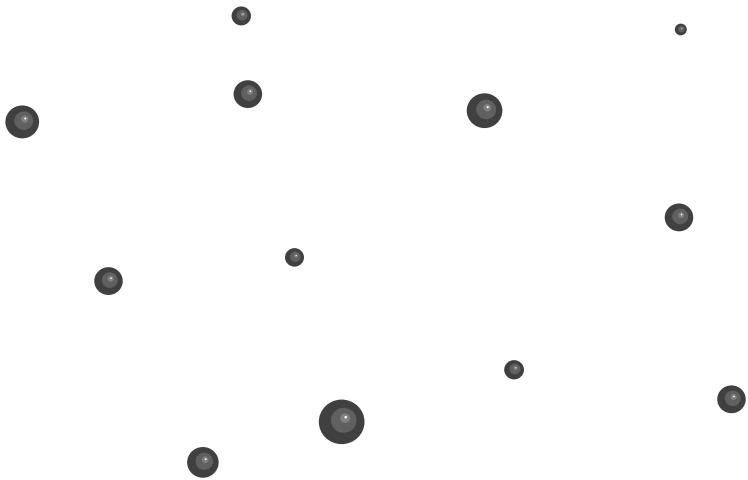
Change

universal set $\supset \mathcal{S}$

points of $\mathbb{R}^d$ segments of $\mathbb{R}^d$

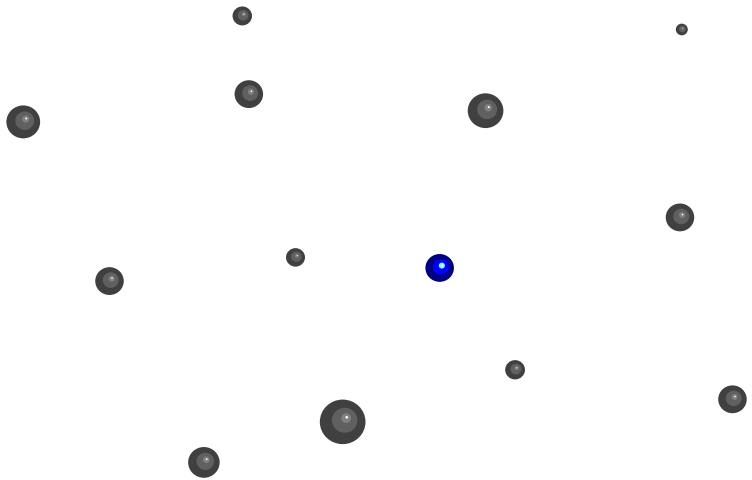
spheres of $\mathbb{R}^d$

Points in $\mathbb{R}^3$



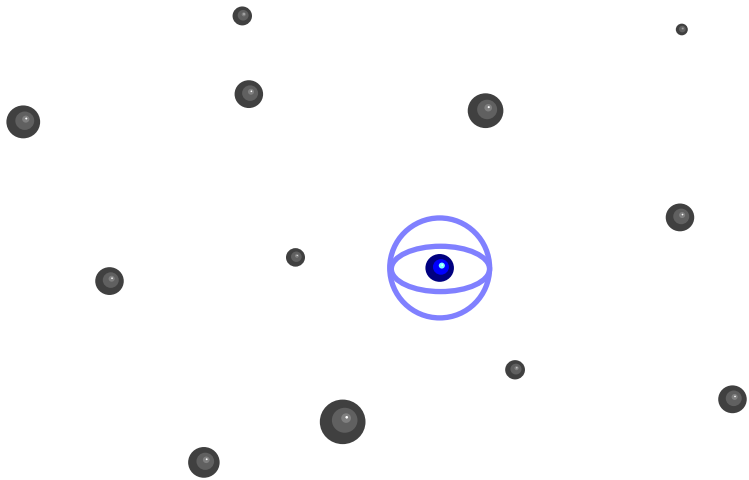
Points in $\mathbb{R}^3$

query



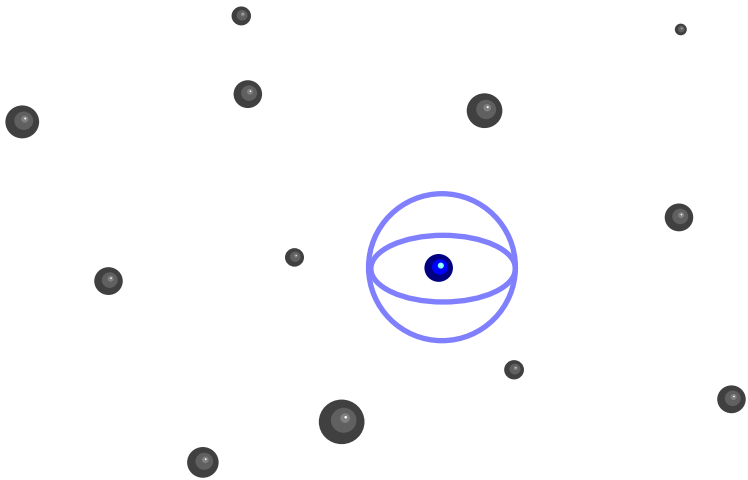
Points in $\mathbb{R}^3$

query



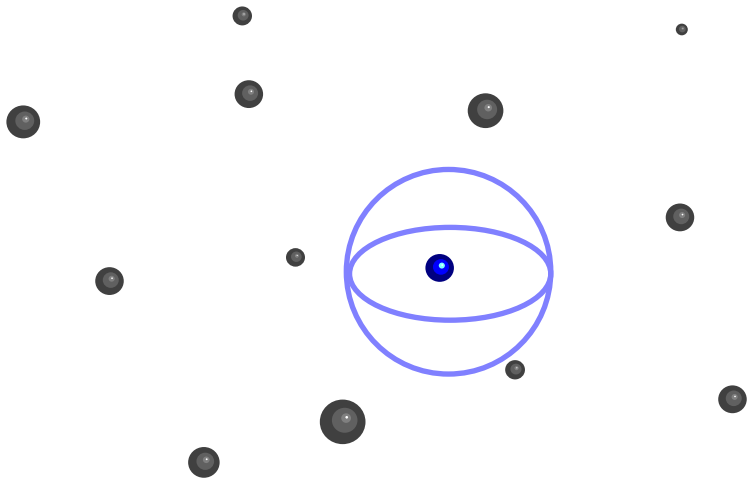
Points in $\mathbb{R}^3$

query



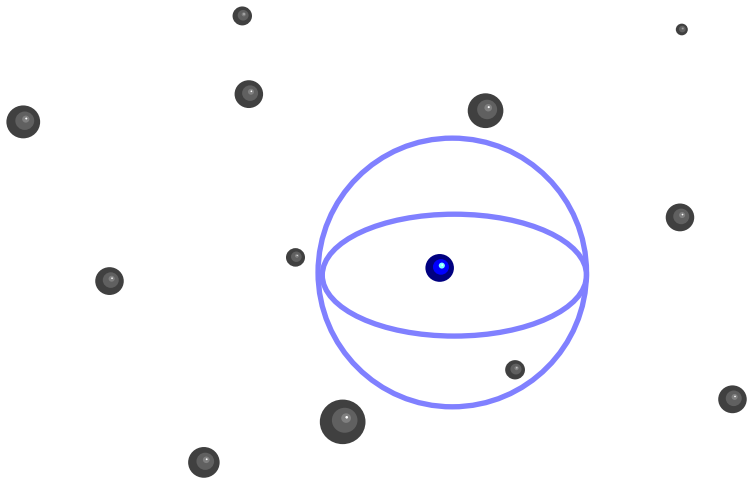
Points in $\mathbb{R}^3$

query

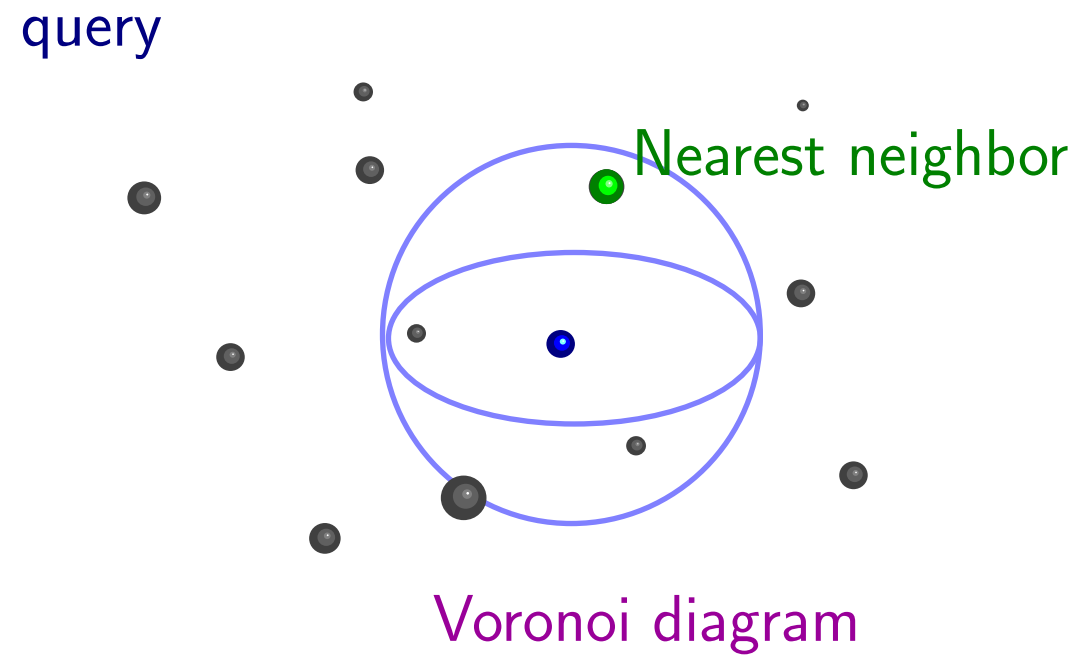


Points in $\mathbb{R}^3$

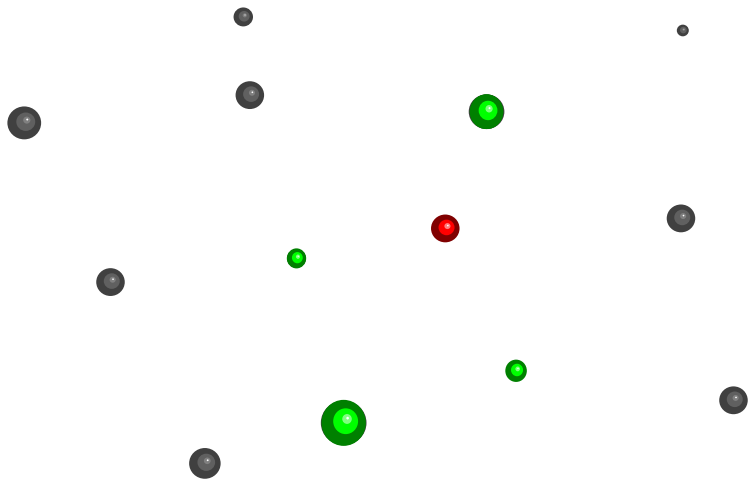
query



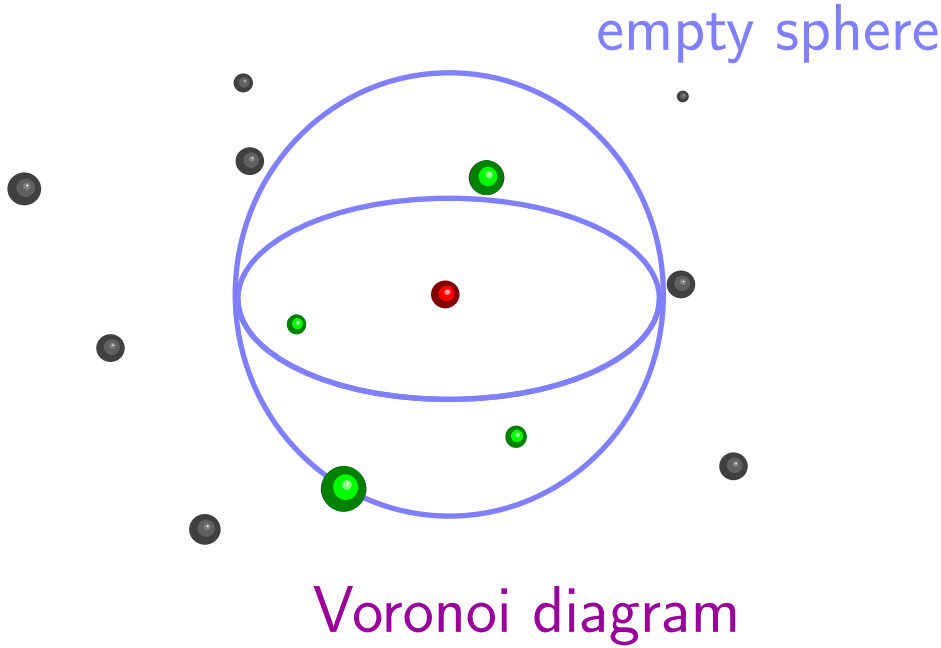
Points in $\mathbb{R}^3$



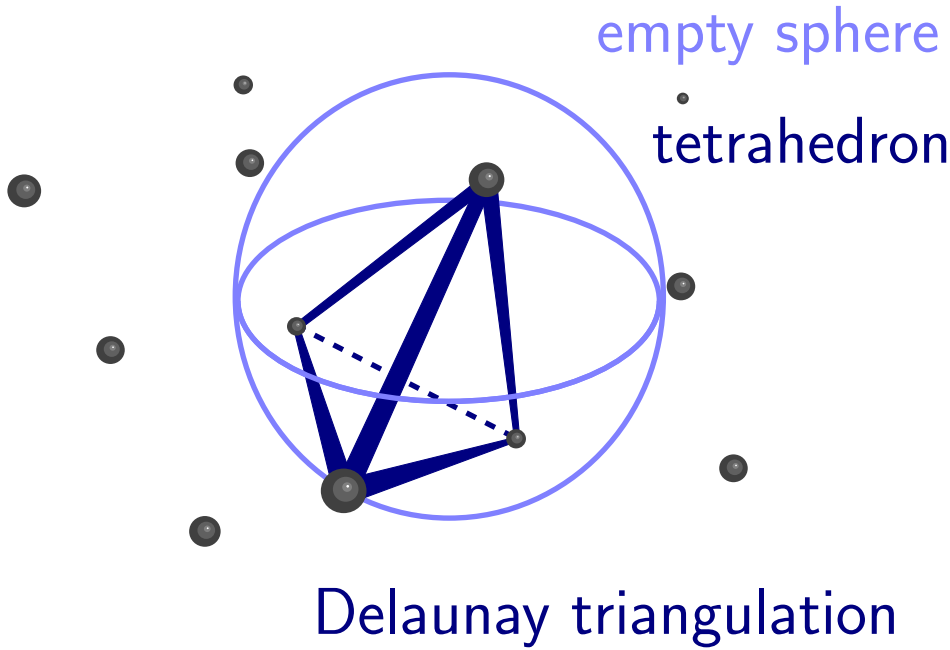
Points in $\mathbb{R}^3$



Points in $\mathbb{R}^3$



Points in $\mathbb{R}^3$



Delaunay in 3D

Similar to 2D:

- Delaunay test (empty sphere)
- incremental algorithm
- randomized algorithm
- duality with convex hull in 4D

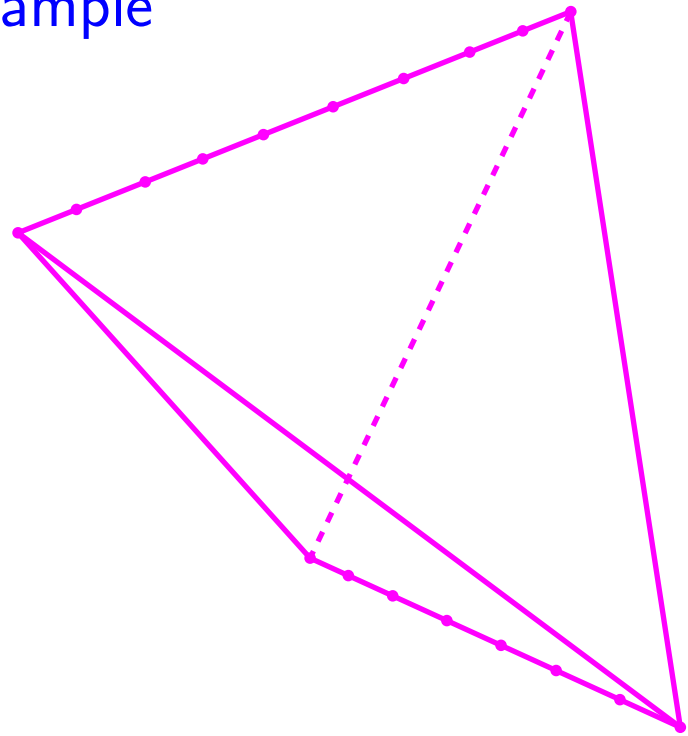
Delaunay in 3D

Similar to 2D:

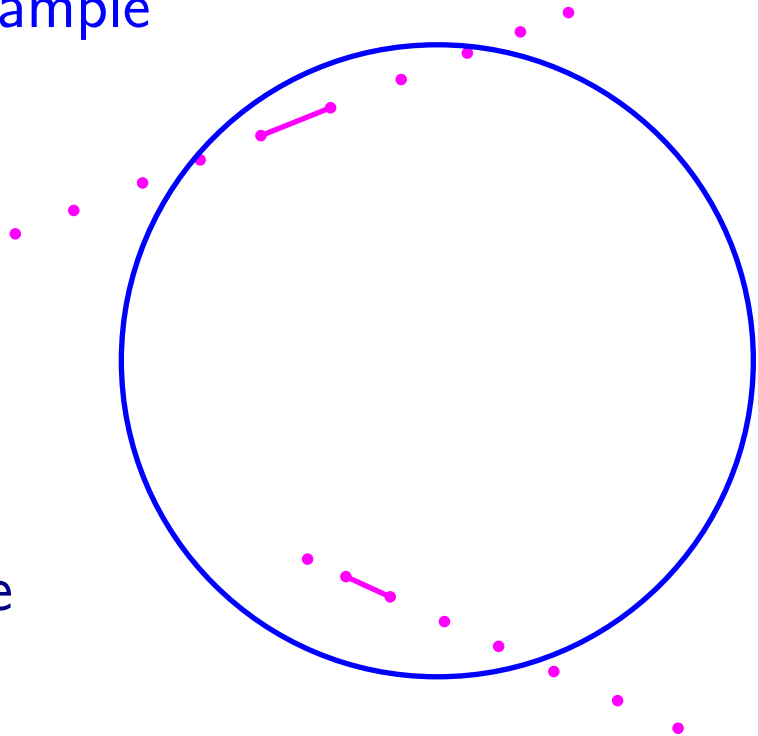
different from 2D

- variable size (linear — quadratic)

Quadratic example



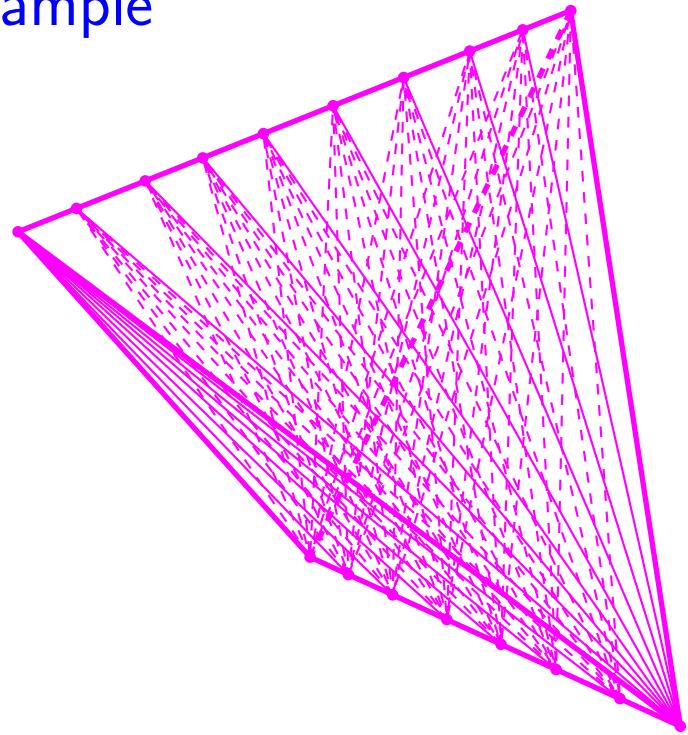
Quadratic example



empty sphere

Quadratic example

$\Omega(n^2)$



Size of Delaunay

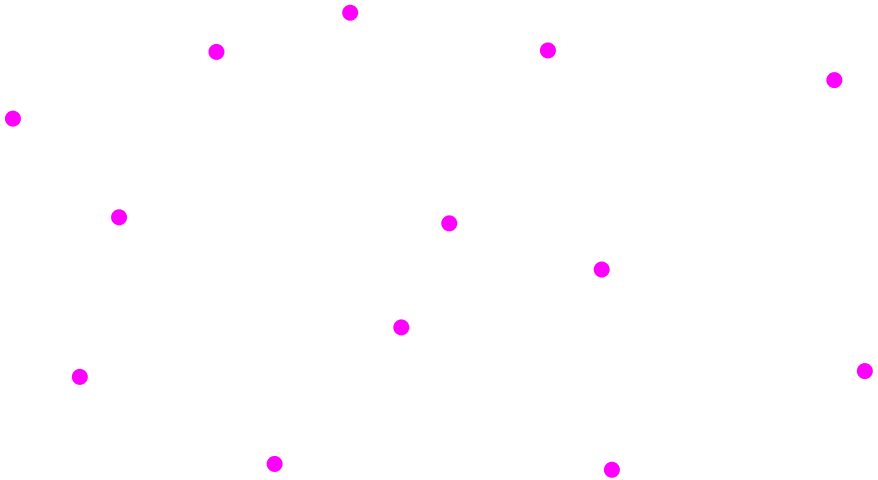
$\Theta(n^2)$ worst case

$\Theta(n)$ uniformly-sampled points in unit cube

$O(n \log n)$ uniformly-sampled points on a surface

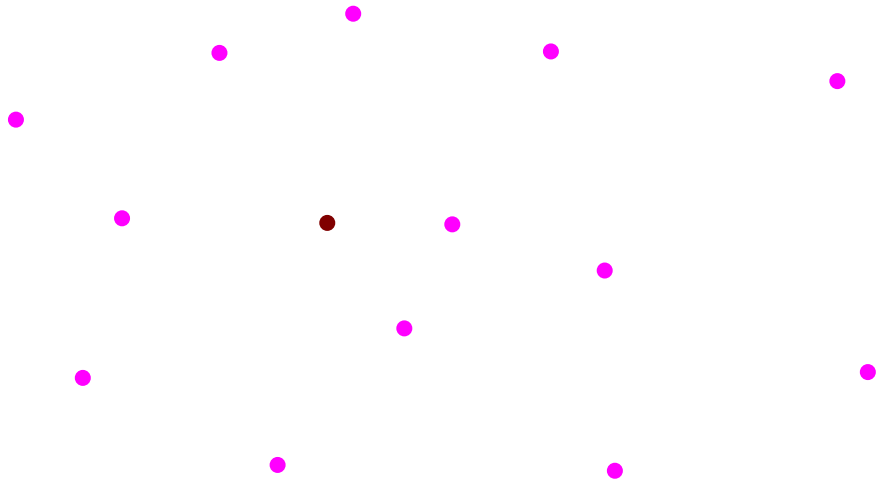
Exotic metrics

Norm L_∞: $\max(|x|, |y|)$



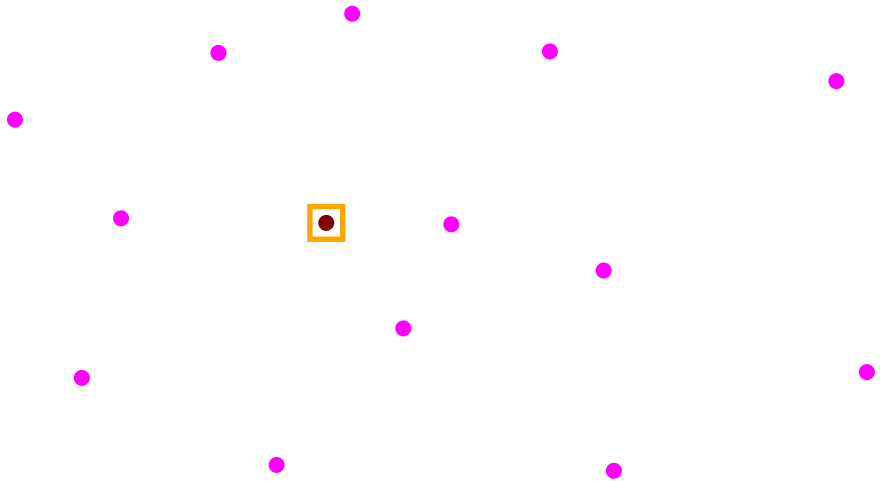
Norm L_∞: $\max(|x|, |y|)$

query



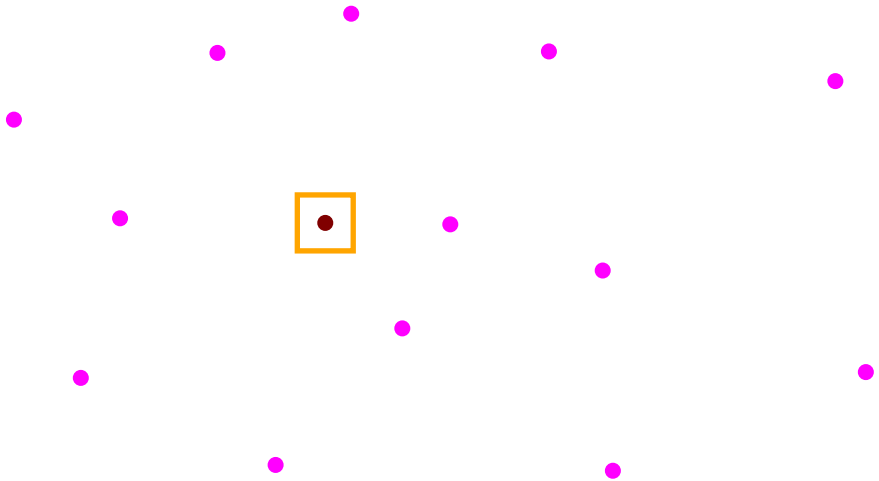
Norm L_∞: $\max(|x|, |y|)$

query



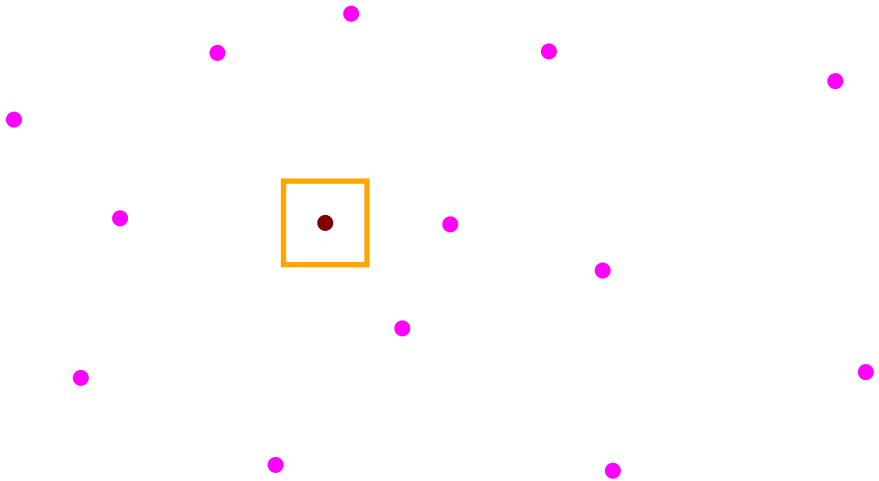
Norm L_∞: $\max(|x|, |y|)$

query



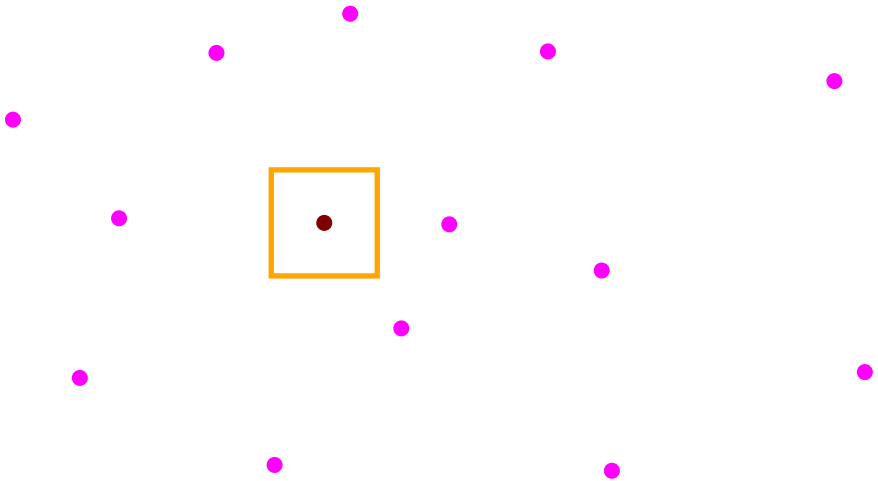
Norm L_∞: $\max(|x|, |y|)$

query



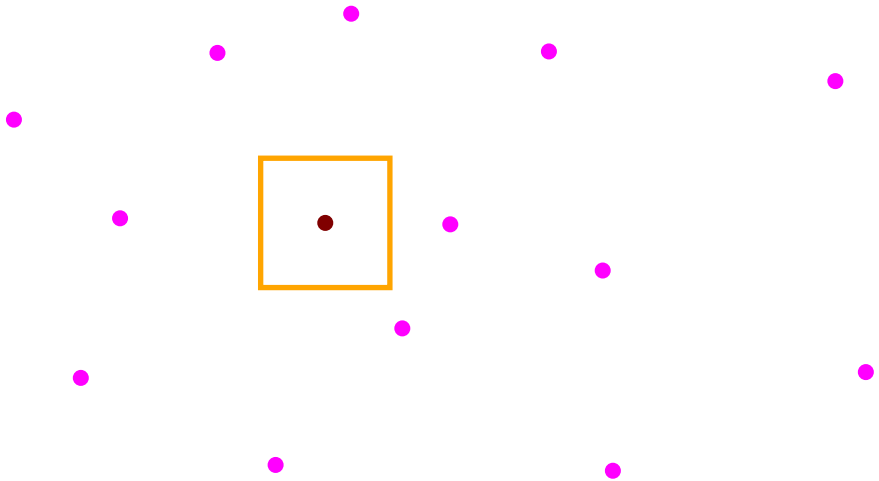
Norm L_∞: $\max(|x|, |y|)$

query



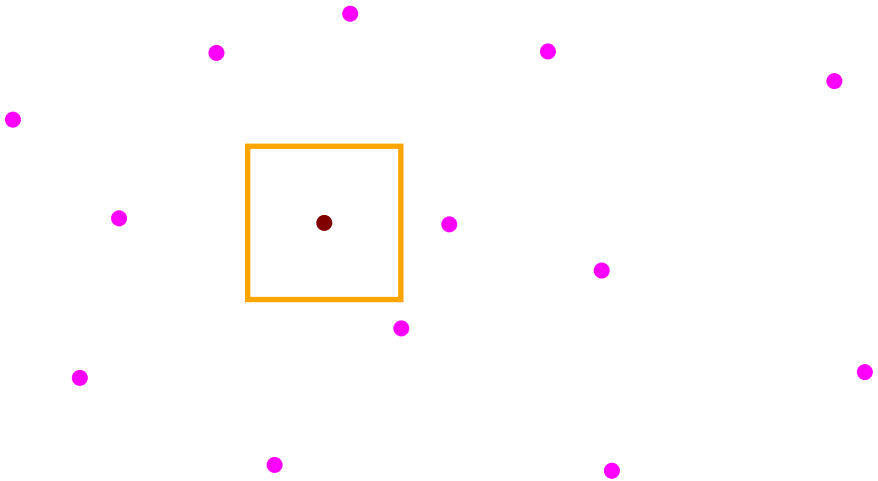
Norm L_∞: $\max(|x|, |y|)$

query



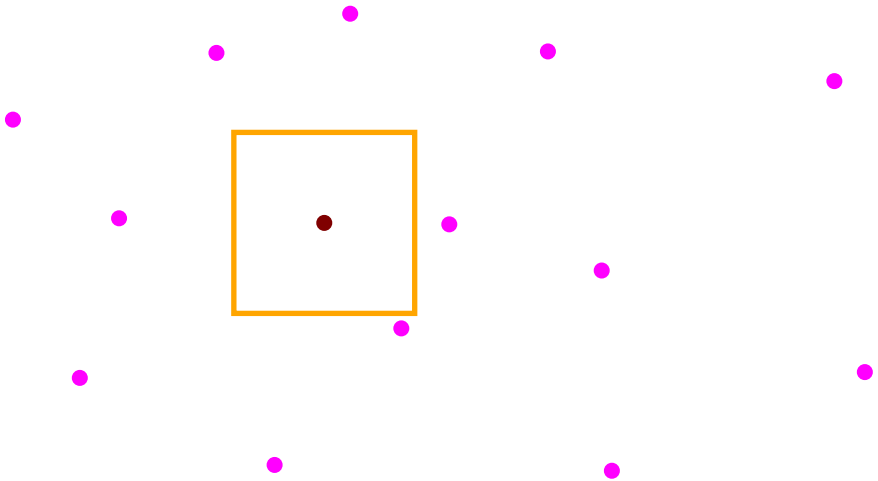
Norm L_∞: $\max(|x|, |y|)$

query



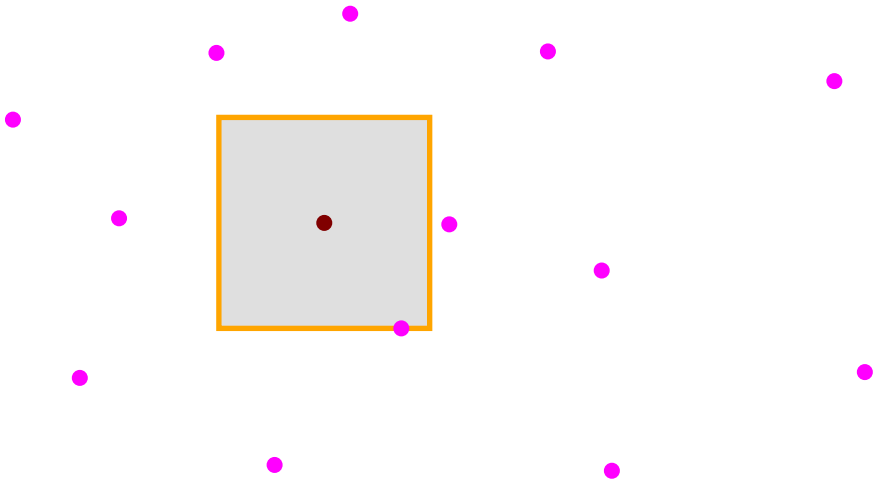
Norm L_∞: $\max(|x|, |y|)$

query

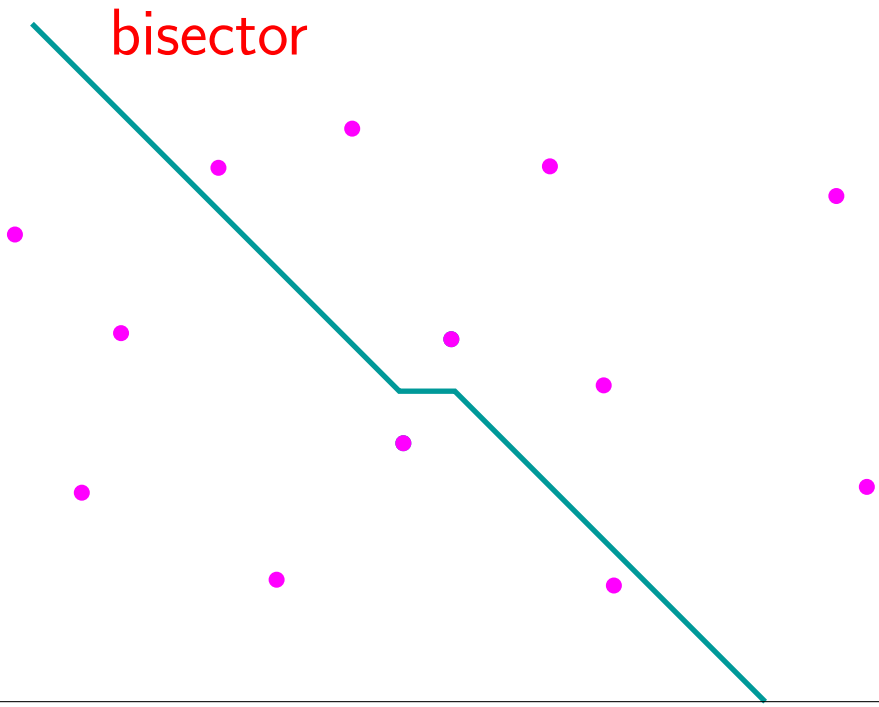


Norm L_∞: $\max(|x|, |y|)$

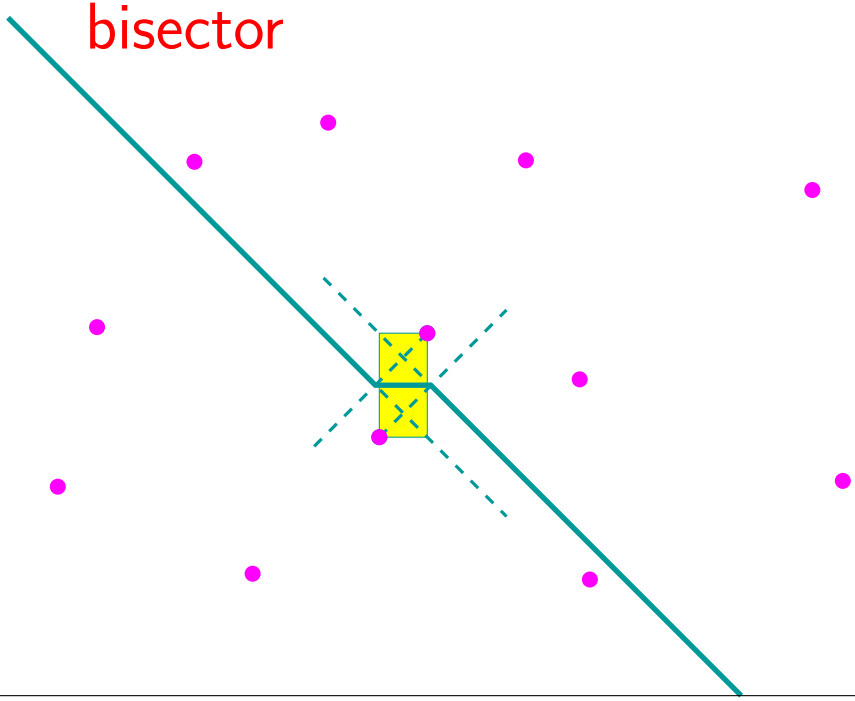
query



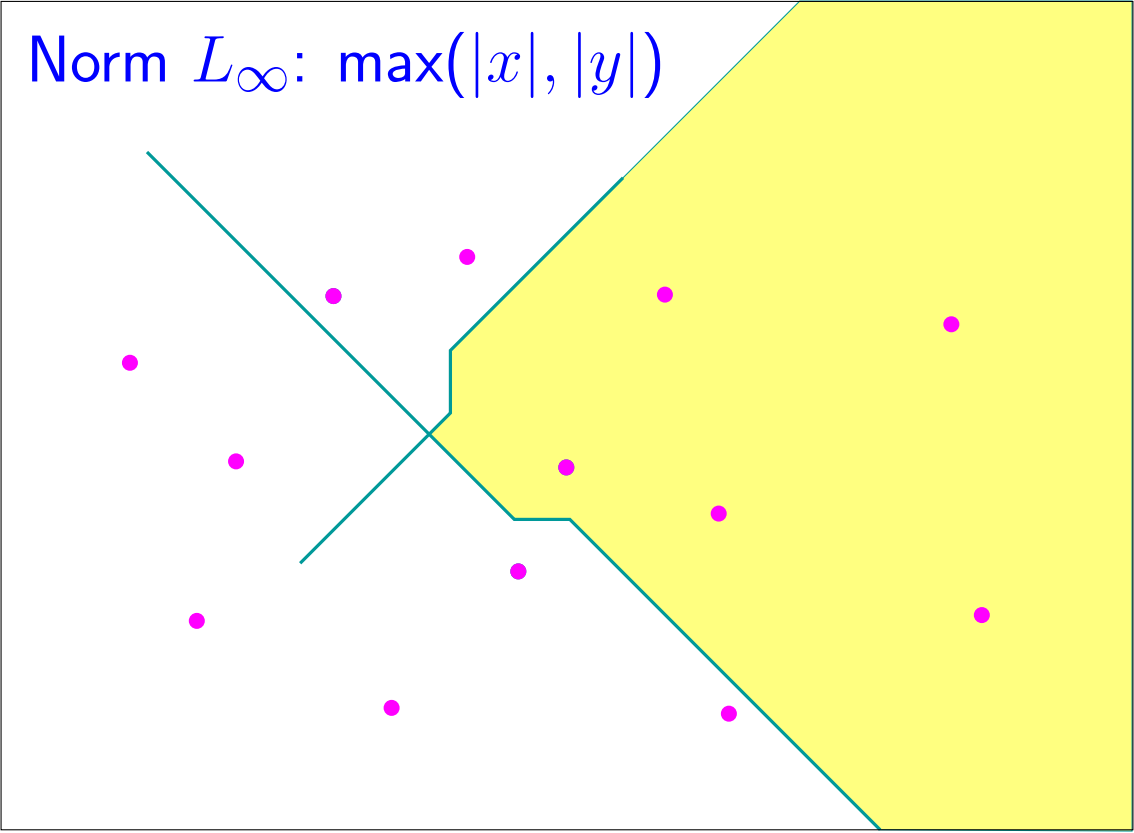
Norm L_∞: $\max(|x|, |y|)$



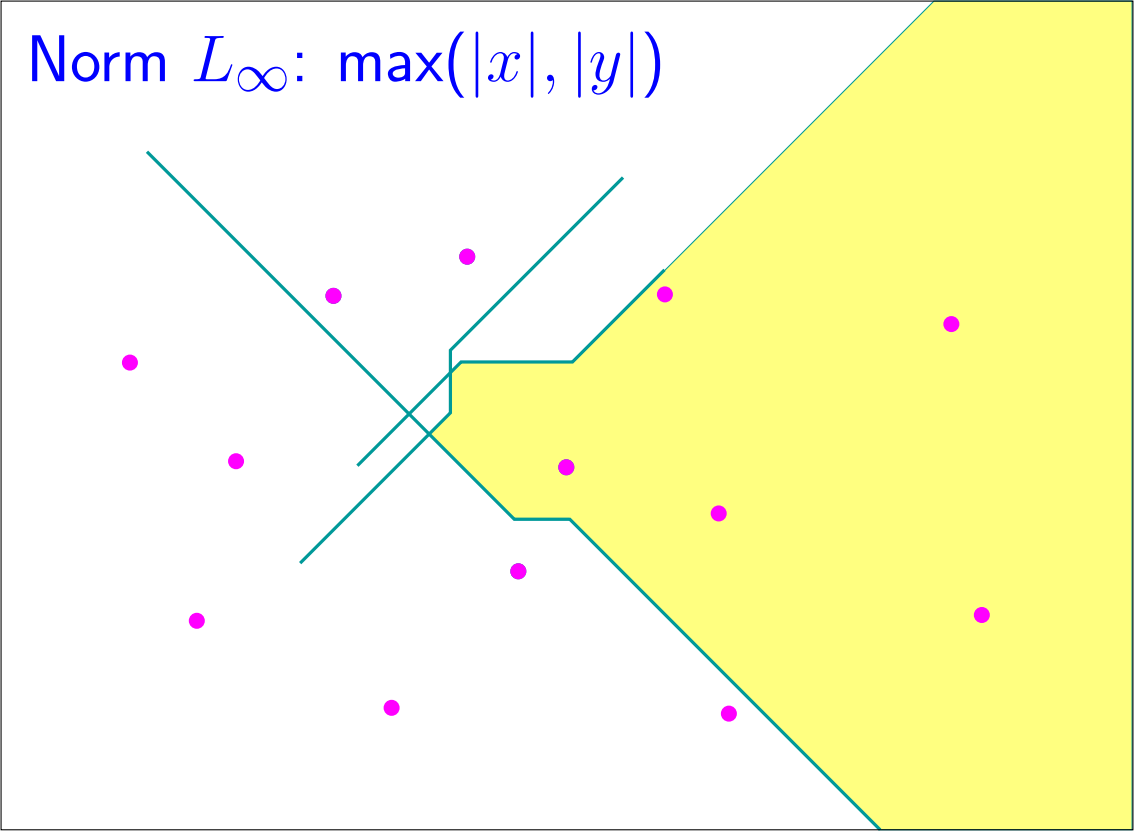
Norm L_∞: $\max(|x|, |y|)$



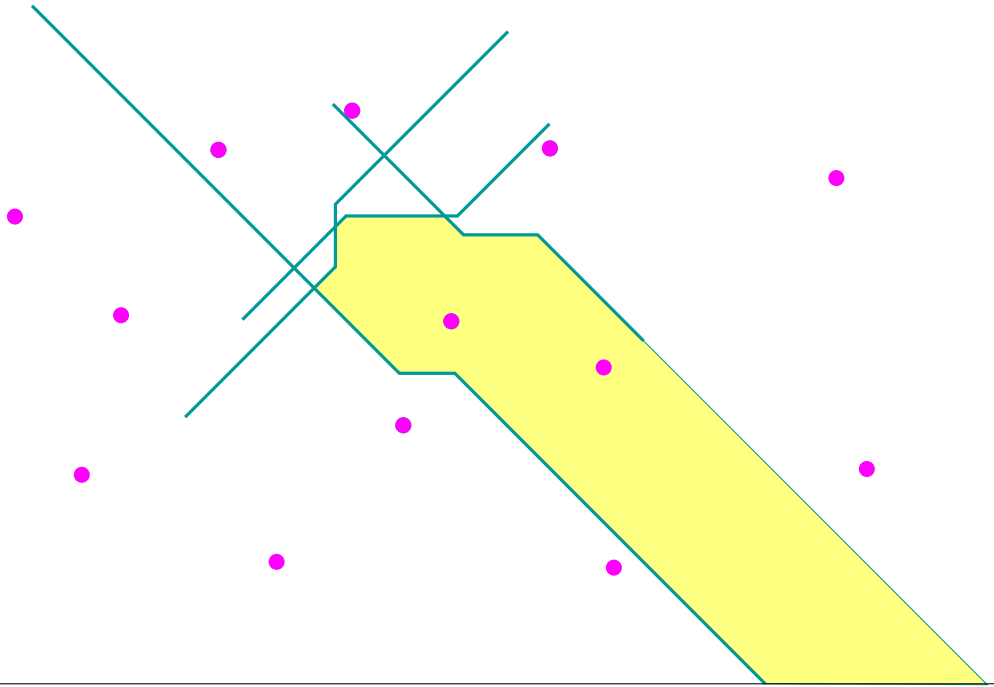
Norm L_∞: $\max(|x|, |y|)$



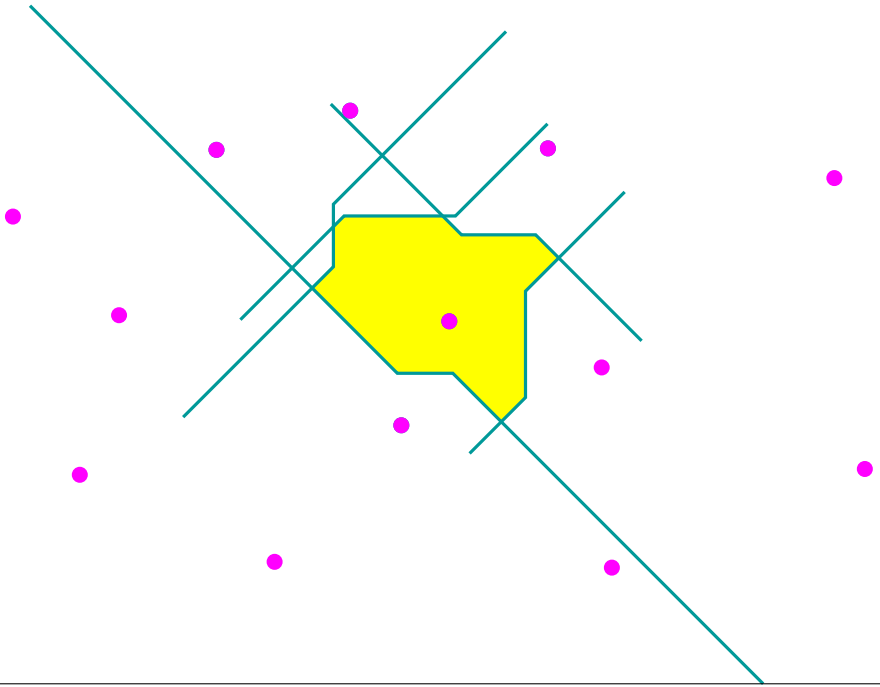
Norm L_∞: $\max(|x|, |y|)$

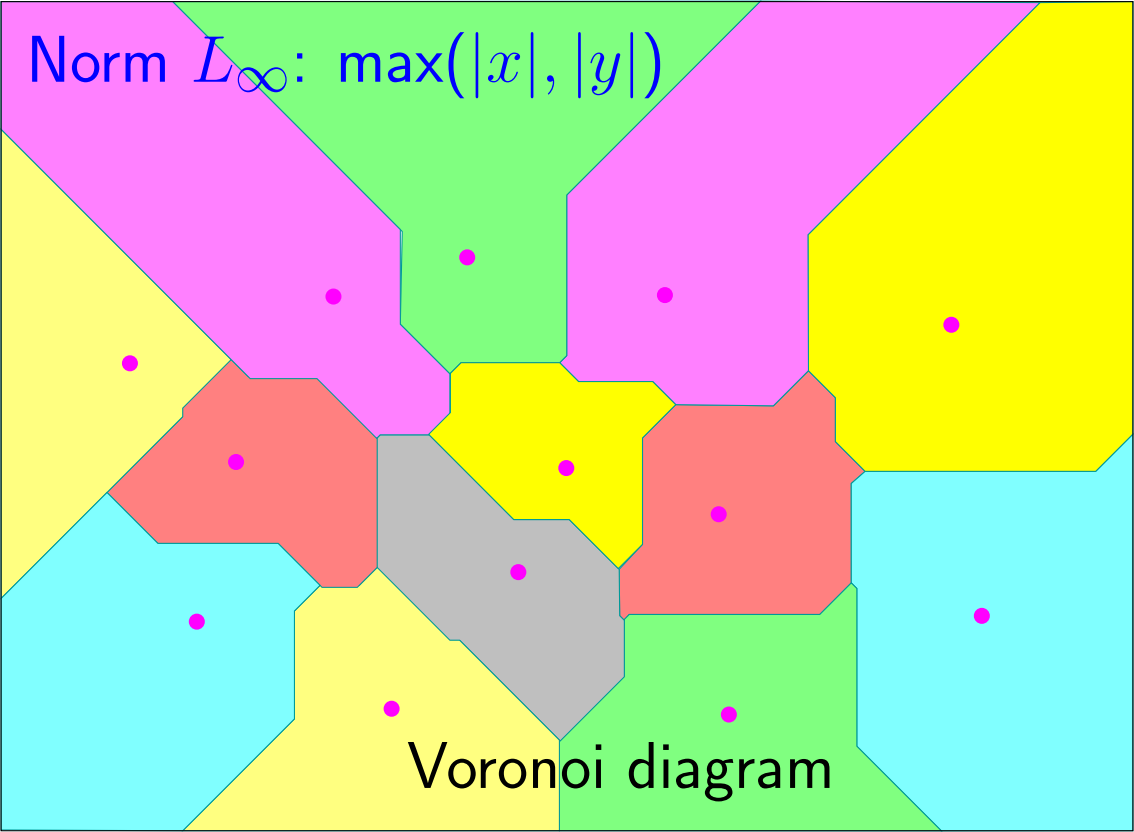


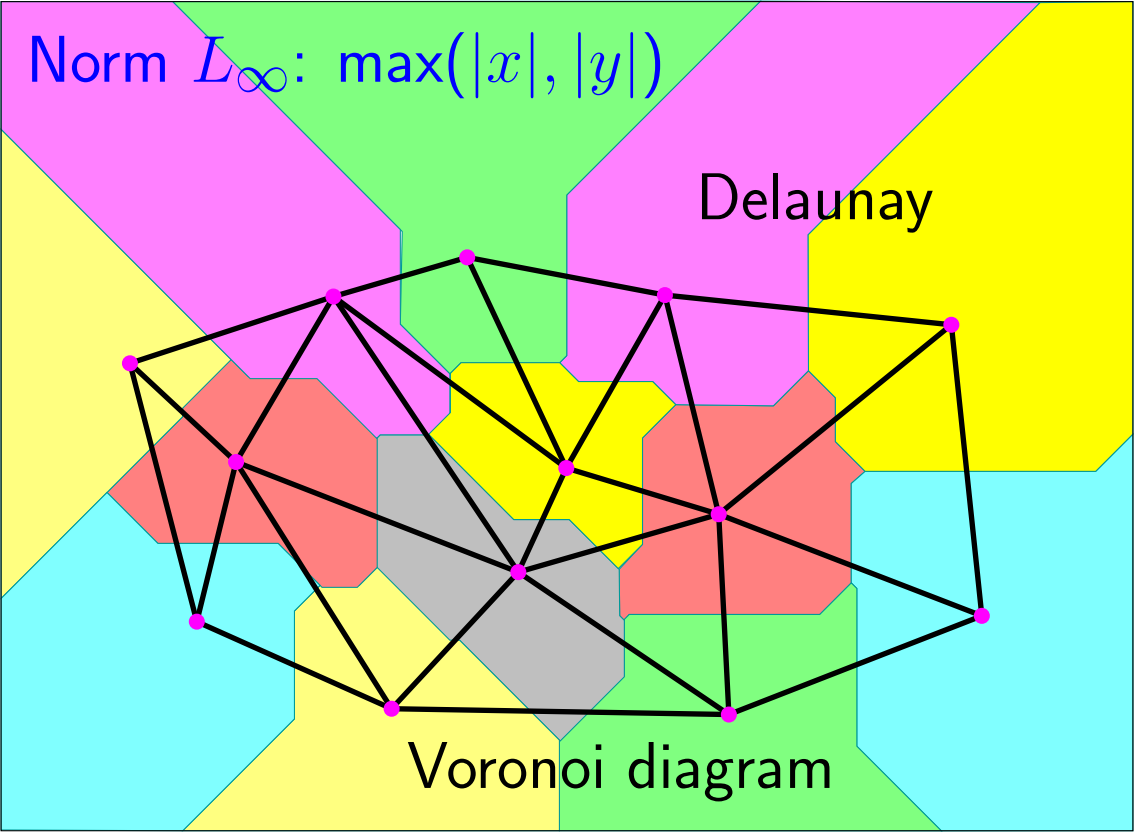
Norm L_∞: $\max(|x|, |y|)$



Norm L_∞: $\max(|x|, |y|)$




Norm L_∞: $\max(|x|, |y|)$
Voronoi diagram

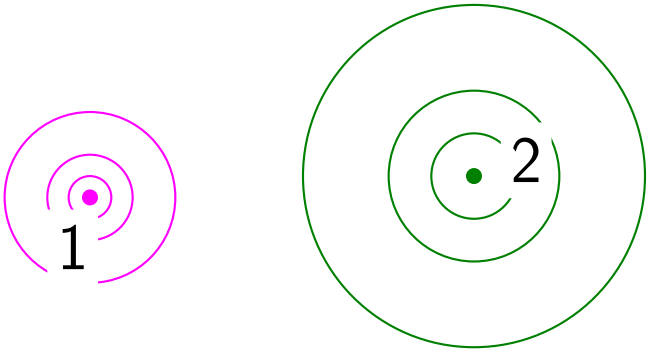

Norm L_∞: $\max(|x|, |y|)$
Delaunay
Voronoi diagram

multiplicatively-weighted points

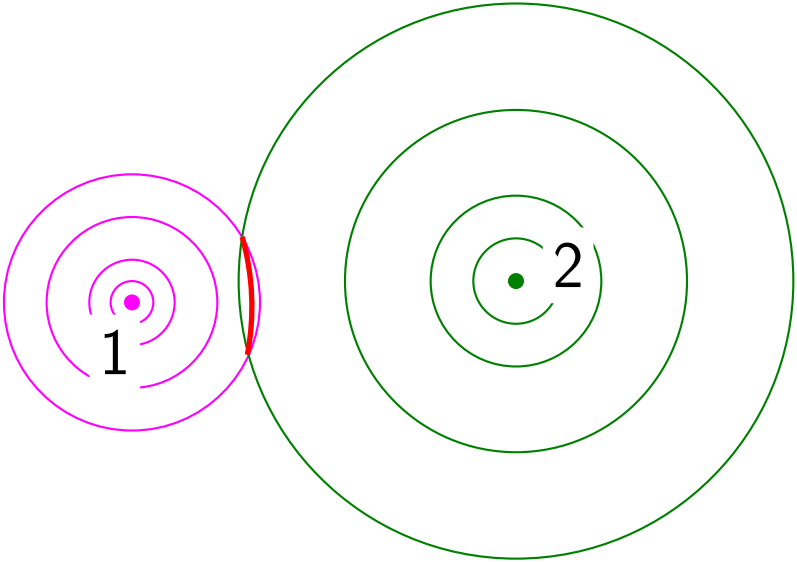
multiplicatively-weighted points

multiplicatively-weighted points

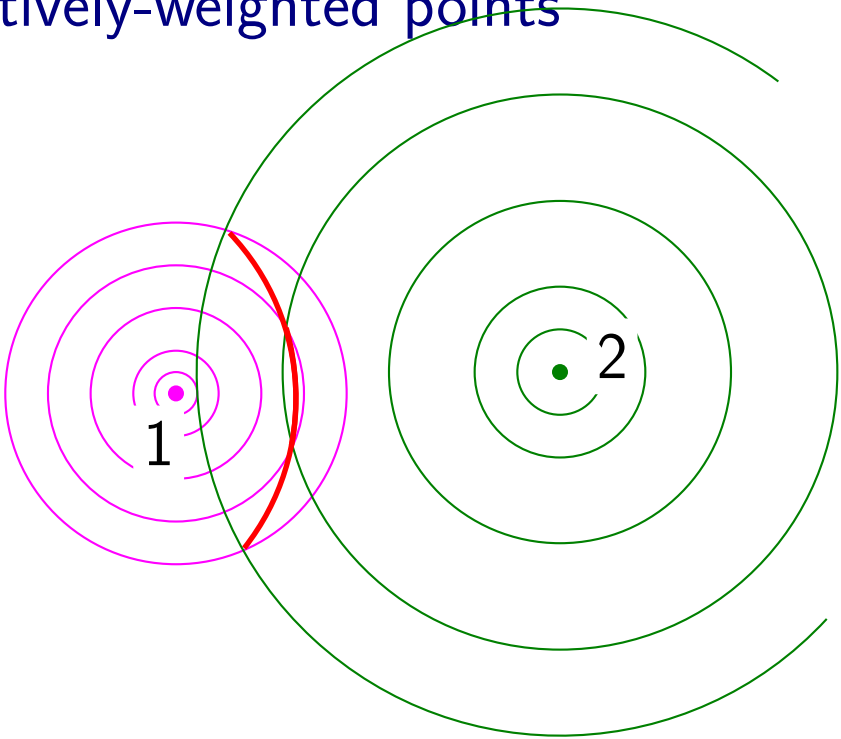
multiplicatively-weighted points



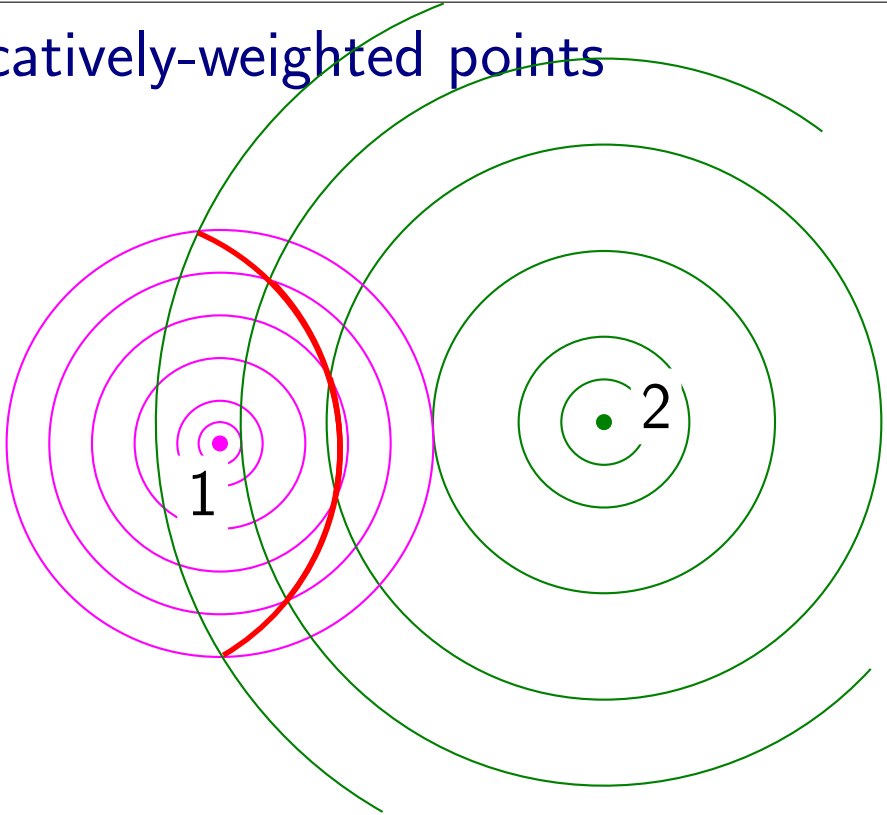
multiplicatively-weighted points

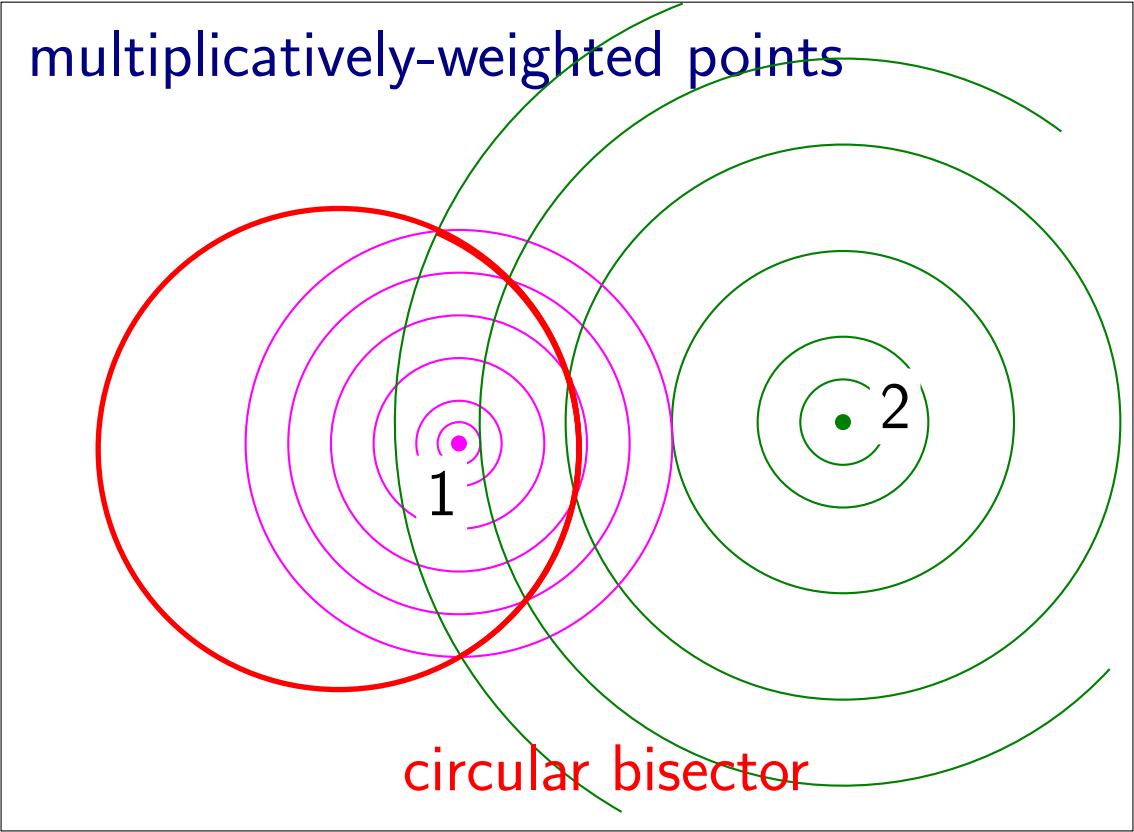


multiplicatively-weighted points

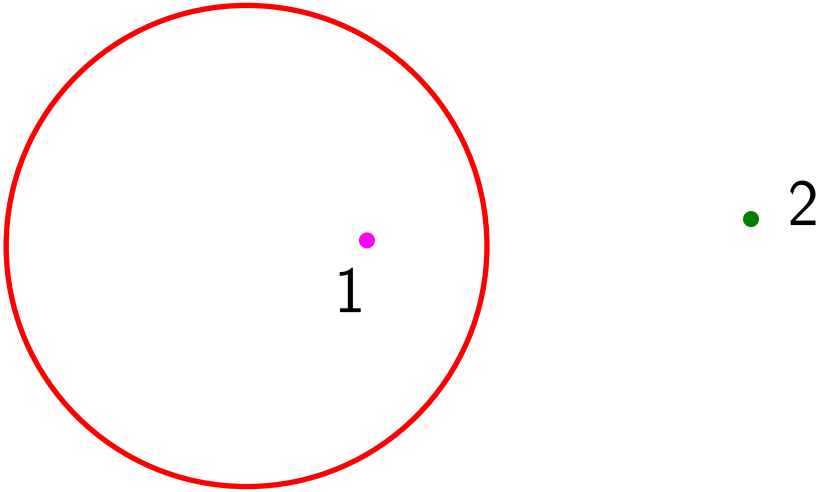


multiplicatively-weighted points



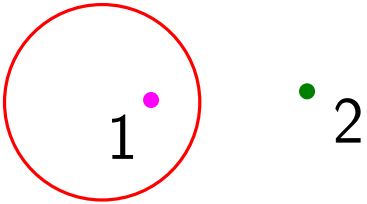

multiplicatively-weighted points
1
2
circular bisector

multiplicatively-weighted points

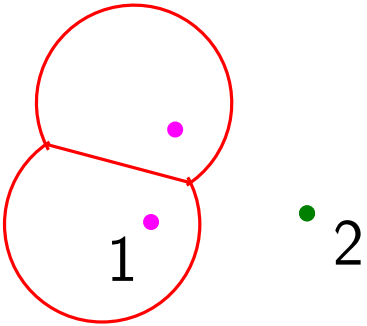


circular bisector

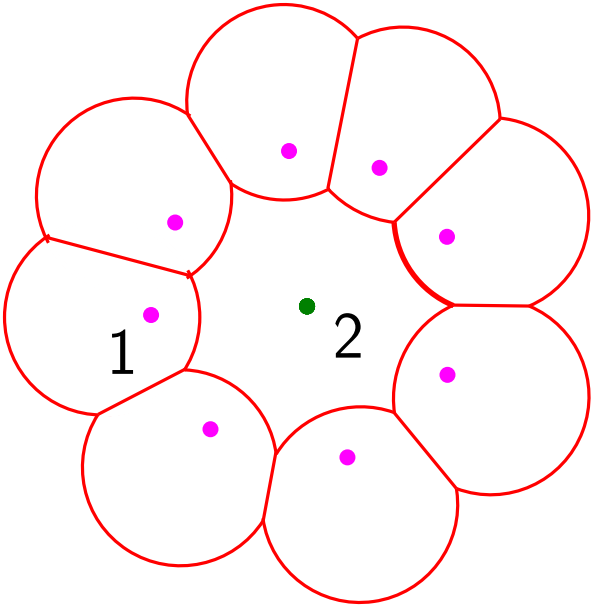
multiplicatively-weighted points



multiplicatively-weighted points



multiplicatively-weighted points

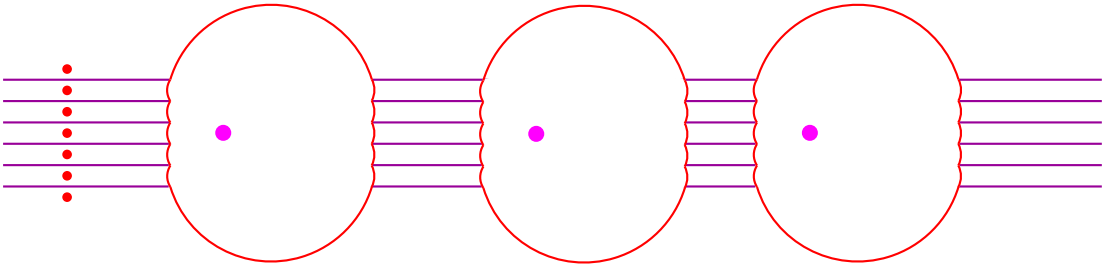


disconnected cell

multiplicatively-weighted points

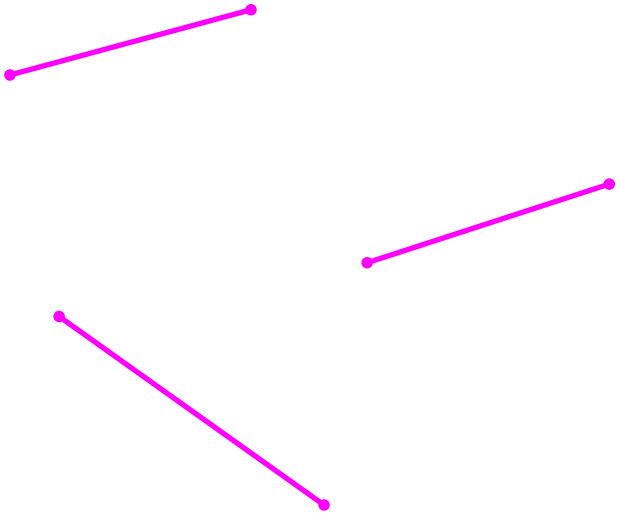
multiplicatively-weighted points

multiplicatively-weighted points

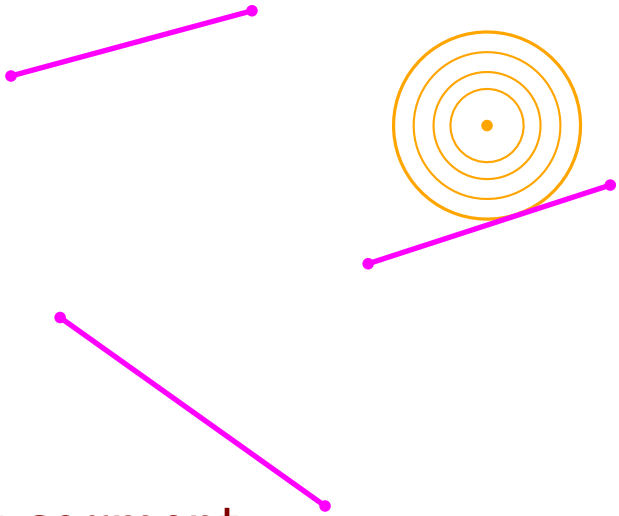


quadratic size

Voronoi diagram of segments

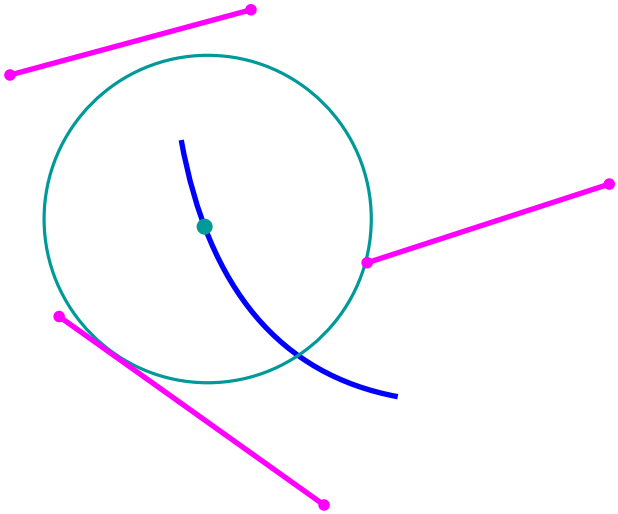


Voronoi diagram of segments



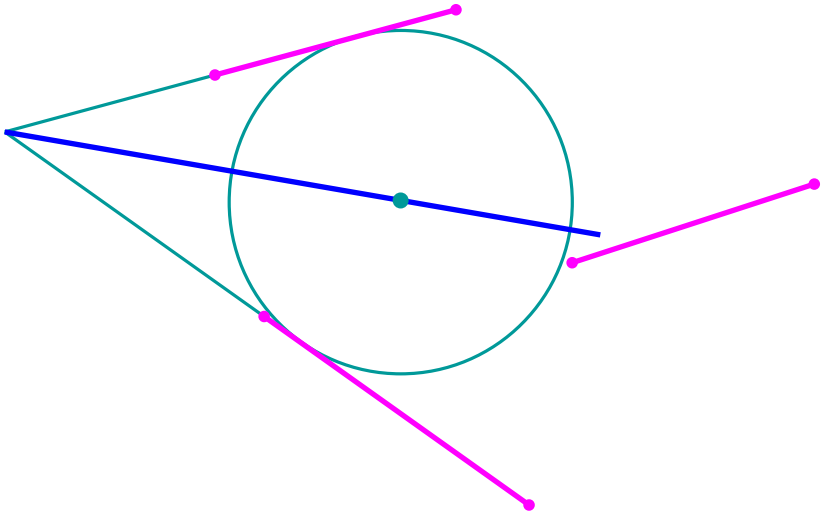
Q nearest segment

Voronoi diagram of segments



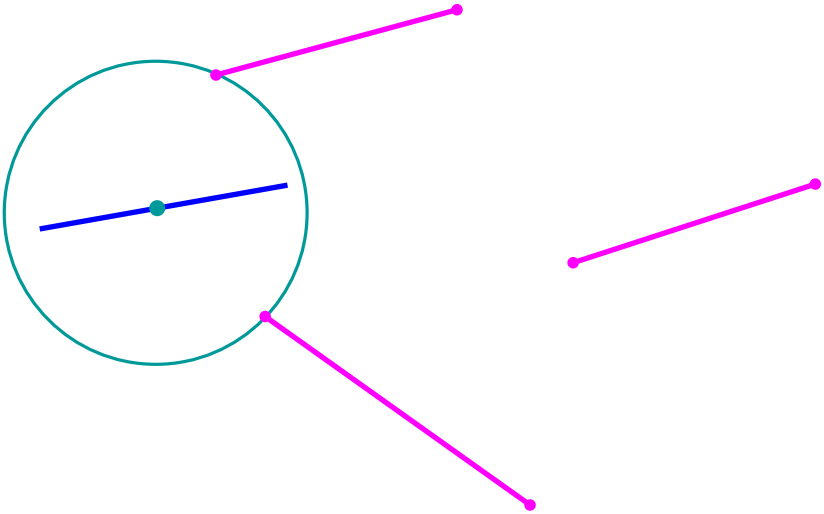
parabolic bisector

Voronoi diagram of segments

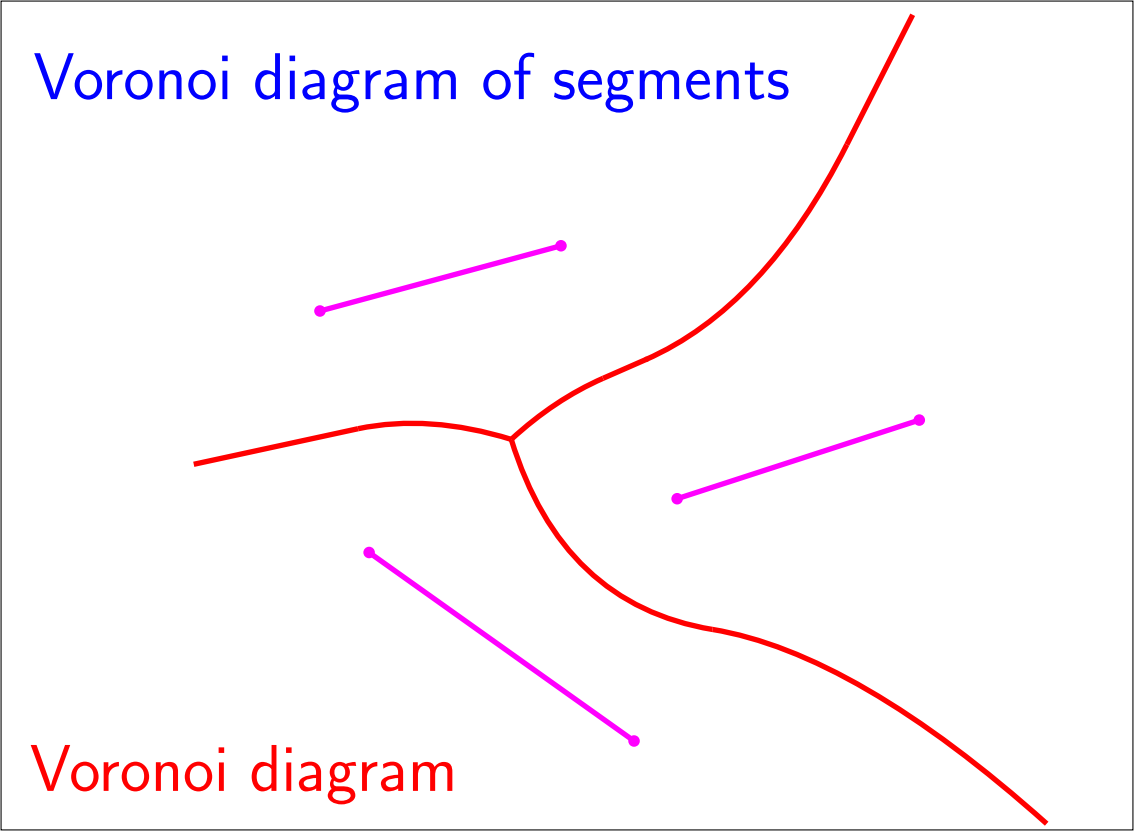


angle bisector

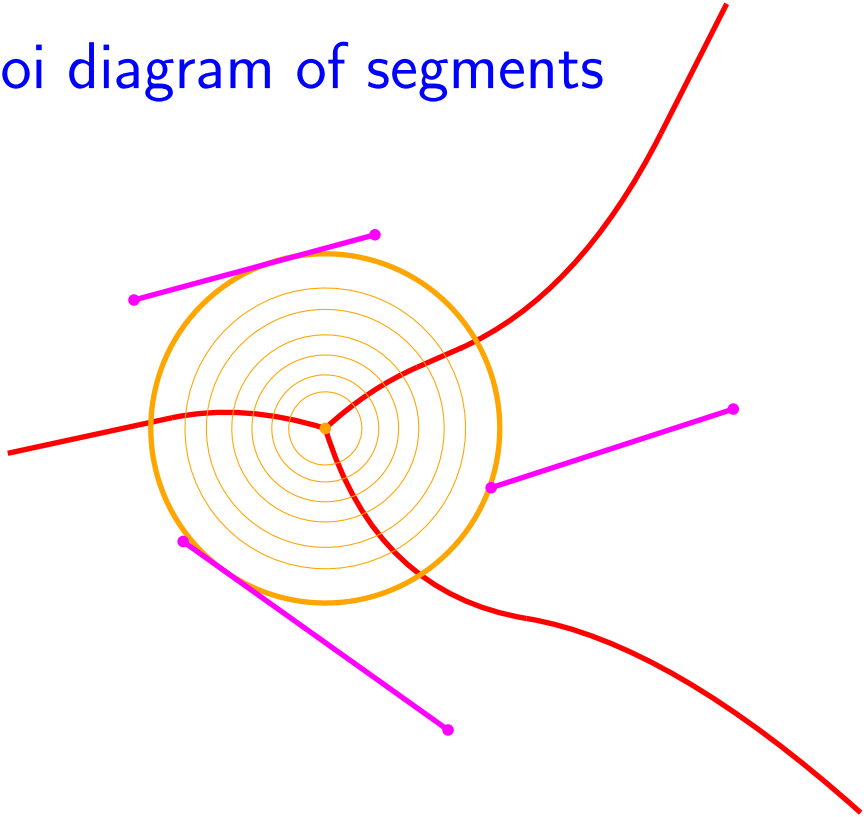
Voronoi diagram of segments



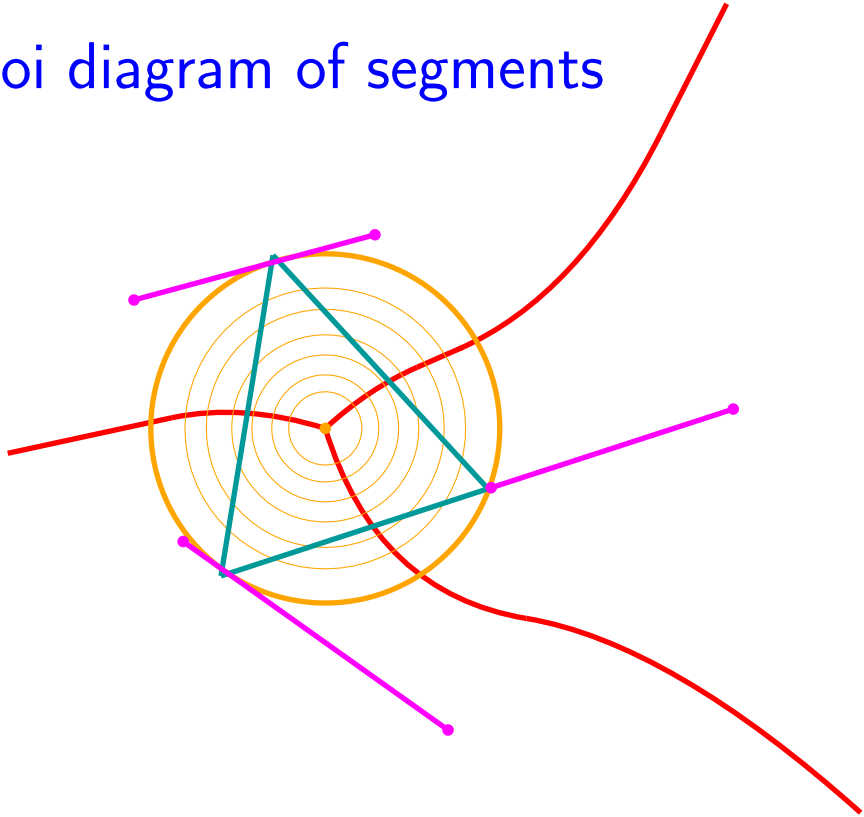
points bisector

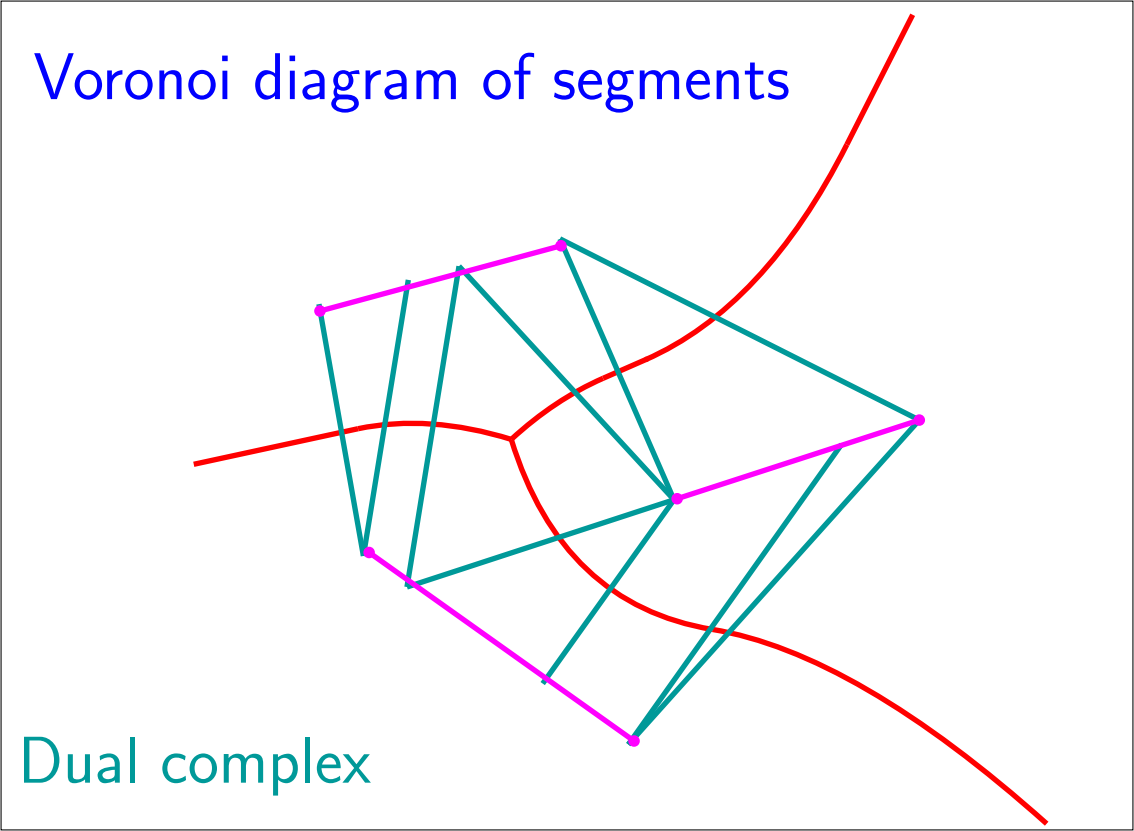

Voronoi diagram of segments
Voronoi diagram

Voronoi diagram of segments



Voronoi diagram of segments




Voronoi diagram of segments
Dual complex

That's all for today