# Diámetro continuo en grafos

Delia Garijo

Universidad de Sevilla

Sesión Especial de Matemática Discreta y Algorítmica
Congreso Bienal de la RSME 2024

# Diámetro continuo en grafos

Delia Garijo

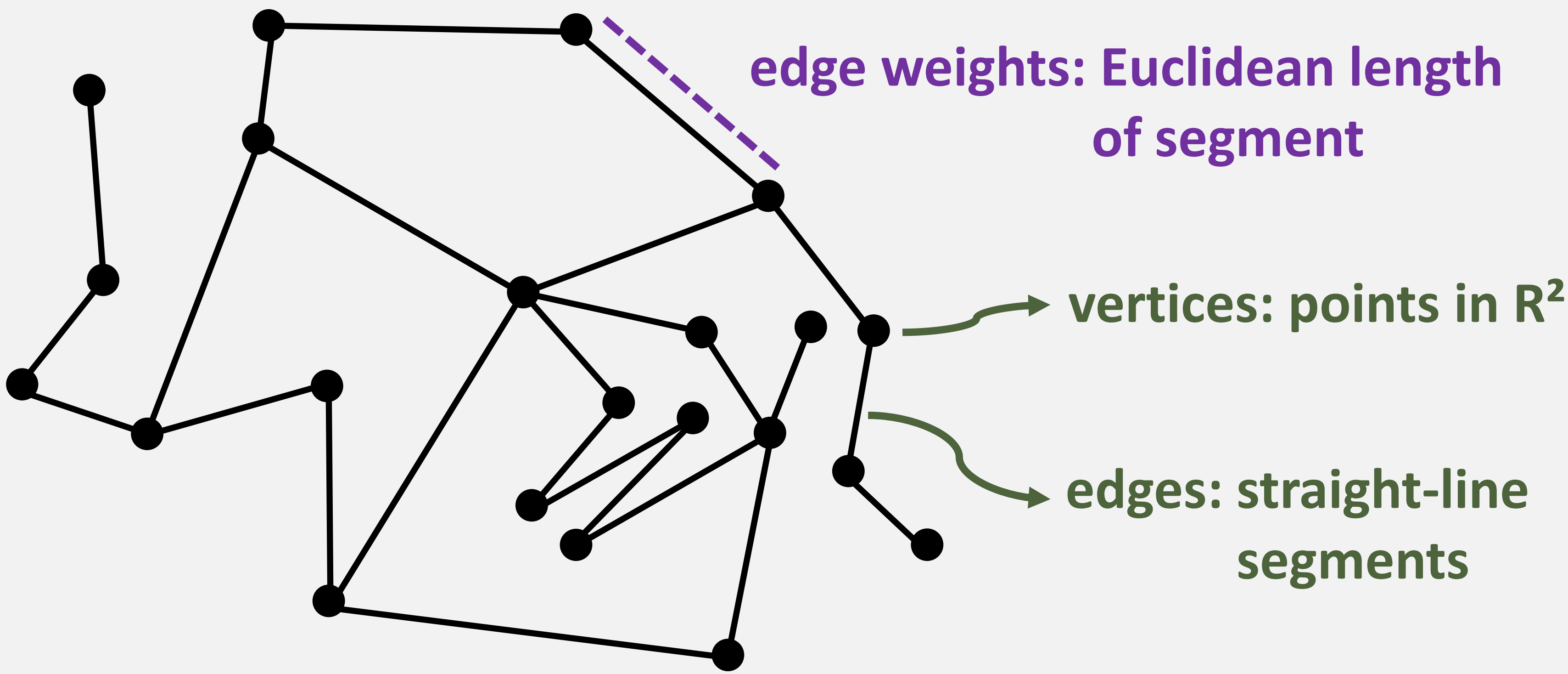
Universidad de Sevilla

**Colaboraciones con:**

- Alberto Márquez | U. Sevilla
- Sergio Cabello| U. Ljubljana
- José Cáceres | U. Almería
- Natalia Rodríguez| U. Buenos Aires
- Fabian Klute | UPC
- Irene Parada| UPC
- Rodrigo I. Silveira | UPC

Sesión Especial de Matemática Discreta y Algorítmica
Congreso Bienal de la RSME 2024

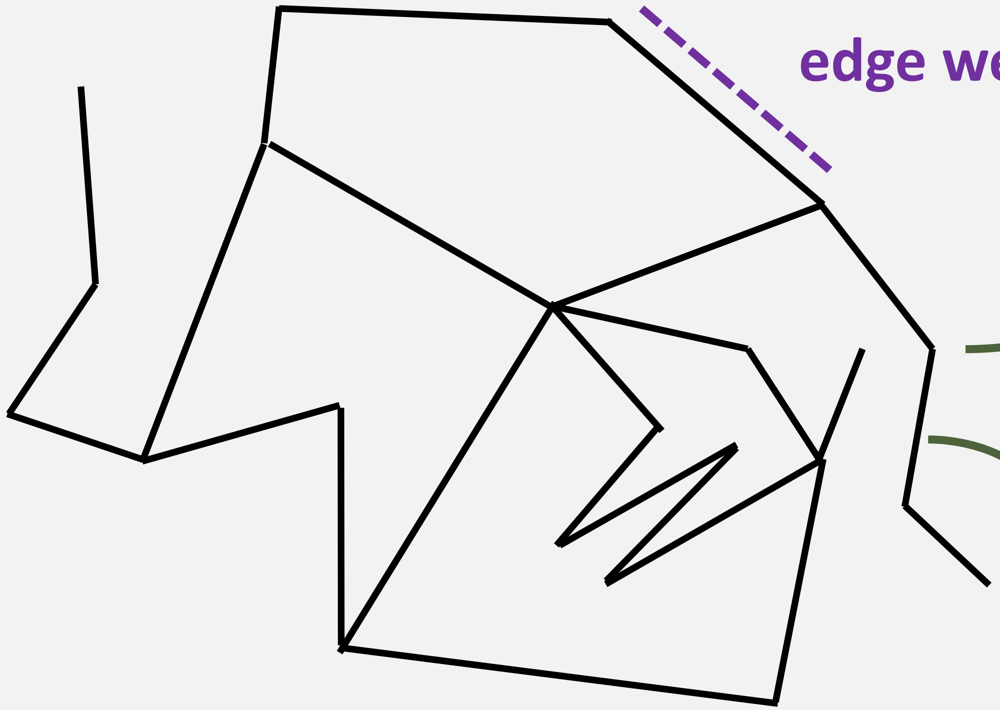
Our object:
A realization of a graph in some Euclidean space




No crossings between edges

Plane geometric graph

# Our object:
# A realization of a graph in some Euclidean space



edge weights: Euclidean length of segment

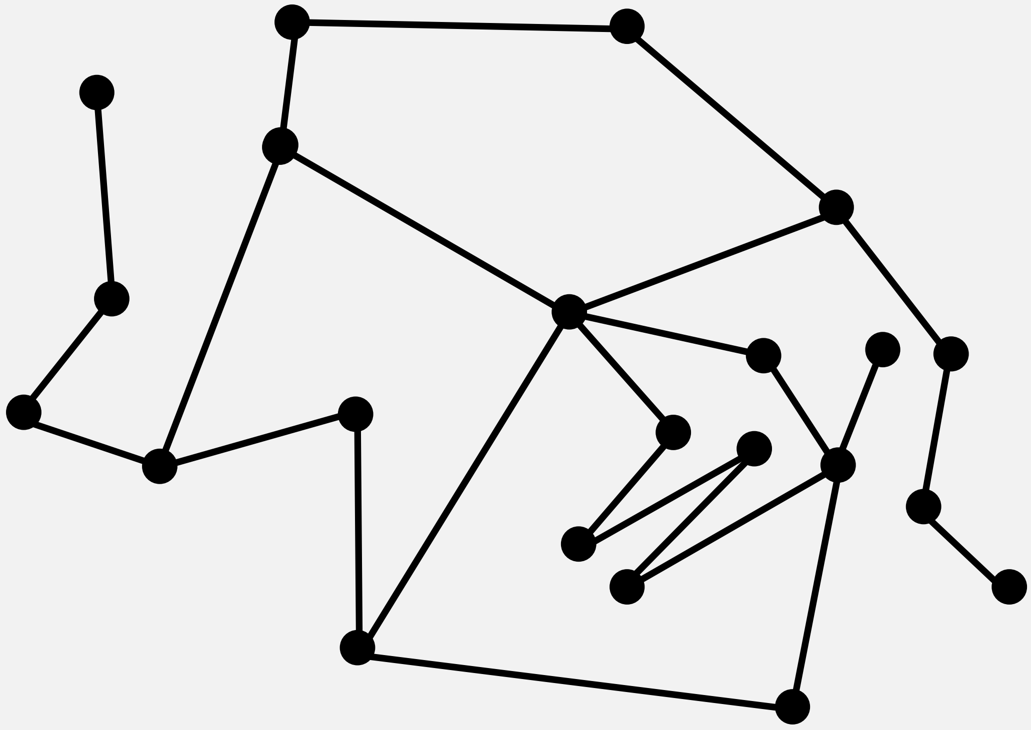
vertices: points in $R^2$

edges: straight-line segments

No crossings between edges

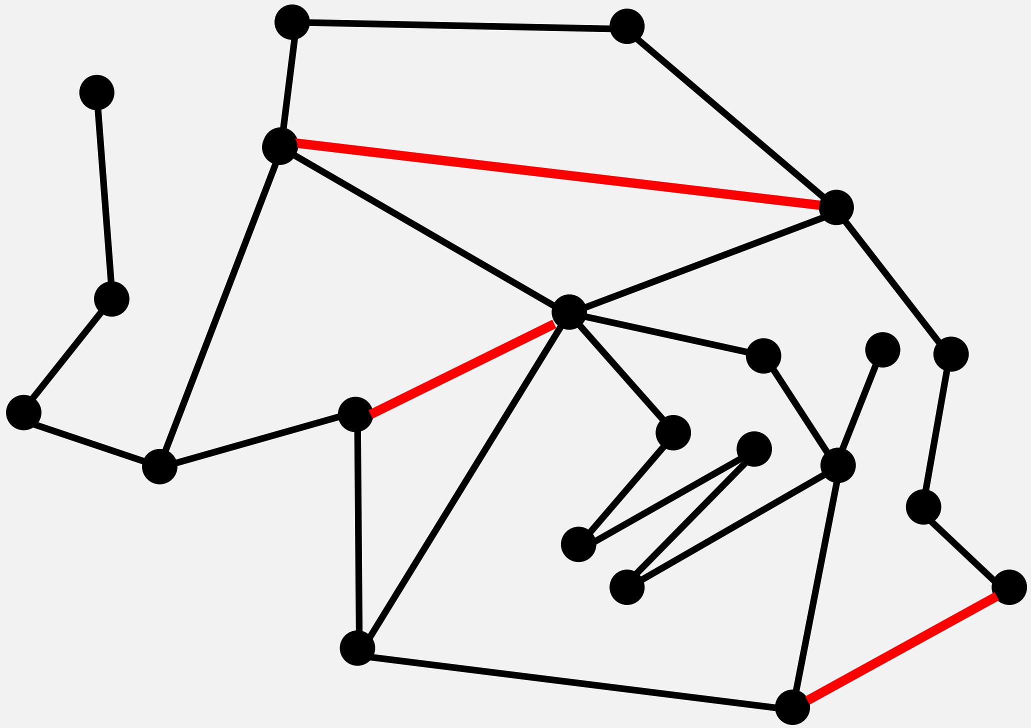
The **LOCUS** of a plane geometric graph

Goal: “**improve**” a graph



Goal: “**improve**” a graph **by adding edges**

**distances**

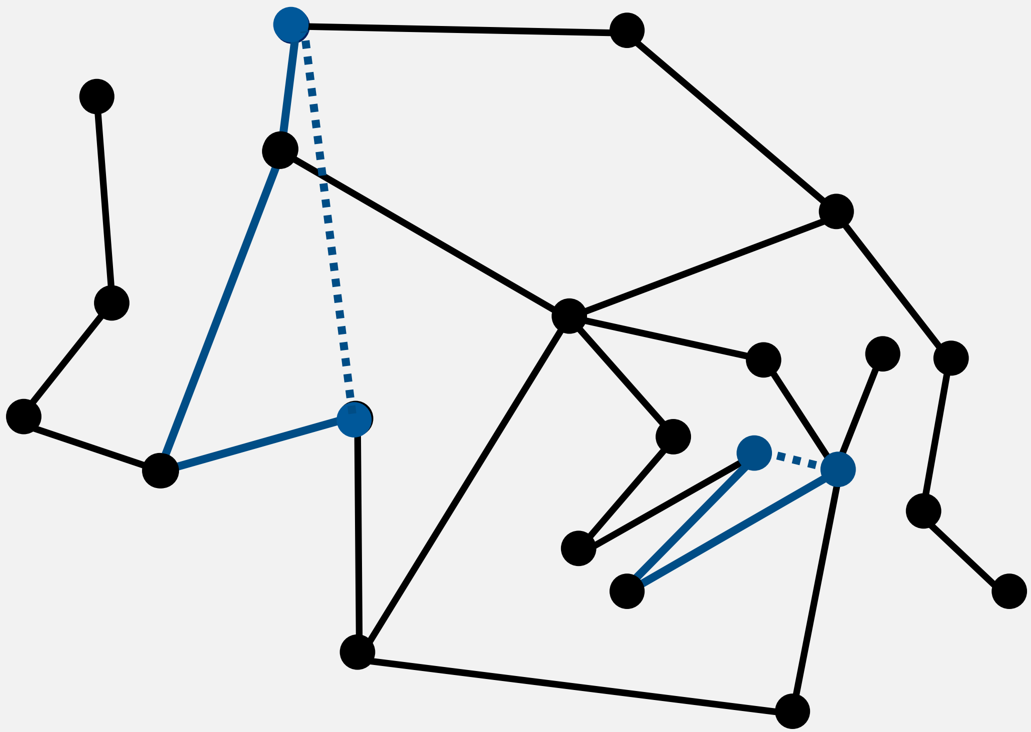


Goal: "**improve**" a graph

**by adding edges**

**distances** → **dilation**

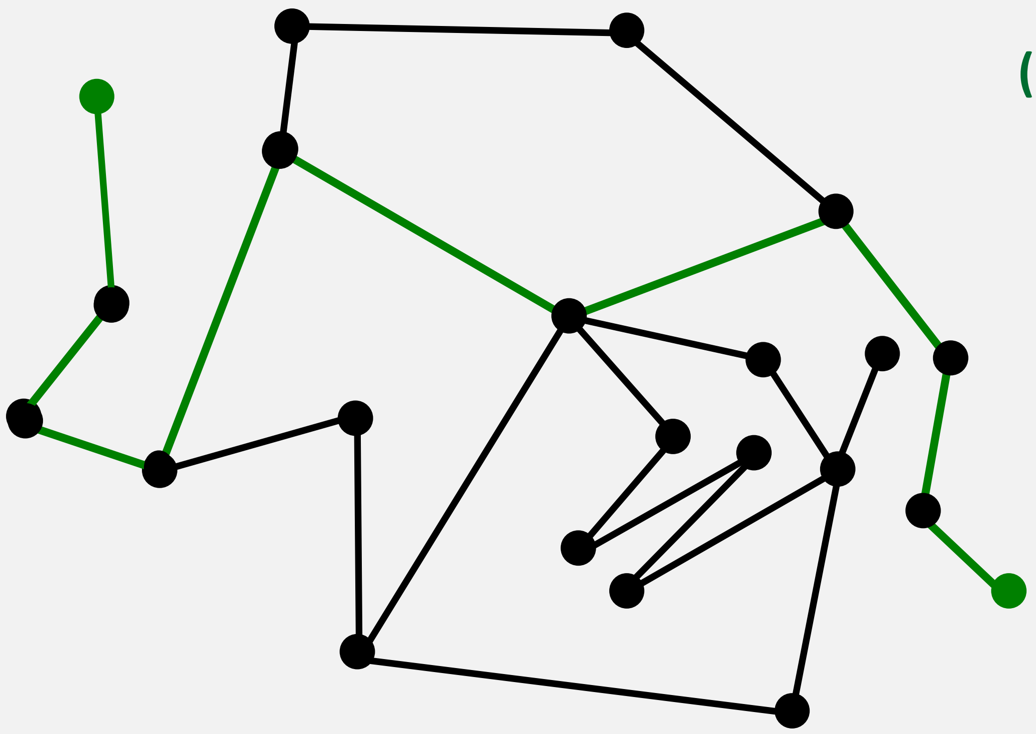
**(max detour between vertices)**



Goal: "**improve**" a graph **by adding edges**

**distances** → **dilation**
**(max detour between vertices)**

**diameter**
**(max distance between vertices)**



Goal: “**improve**” a graph

**by adding edges**

**distances** → **dilation**

**(max detour between vertices)**

**diameter**

**(max distance between vertices)**

**Optimal k-augmentation problem:**

Insert k additional edges to minimize some measure on the resulting graph

Goal: "**improve**" a graph **by adding edges**

→ **distances** → **dilation** **(max detour between vertices)**

**diameter** **(max distance between vertices)**

**Optimal k-augmentation problem:**

Insert k additional edges to minimize some measure on the resulting graph

→ **Adding one edge/diameter:**

**Grobe et al., 2015** (trees embedded in a metric space)
**Wang, 2017** (paths embedded in a metric space)
**Biló, 2018** (trees embedded in a metric space)
**Wang and Zhao, 2021** (unicycle graphs and trees embedded in a metric space)

**Adding k edges/diameter:**

**Biló et al. , 2023** (trees embedded in a metric space)

Goal: "**improve**" a graph **by adding edges**

→ **distances** → **dilation** (max detour between vertices)

**diameter** (max distance between vertices)

## Optimal k-augmentation problem:

Insert k additional edges to minimize some measure on the resulting graph

### Adding one edge/diameter:

**Grobe et al., 2015** (trees embedded in a metric space)
**Wang, 2017** (paths embedded in a metric space)
**Biló, 2018** (trees embedded in a metric space)
**Wang and Zhao, 2021** (unicycle graphs and trees embedded in a metric space)

### Adding k edges/diameter:

**Biló et al. , 2023** (trees embedded in a metric space)

### Adding one edge/dilation:

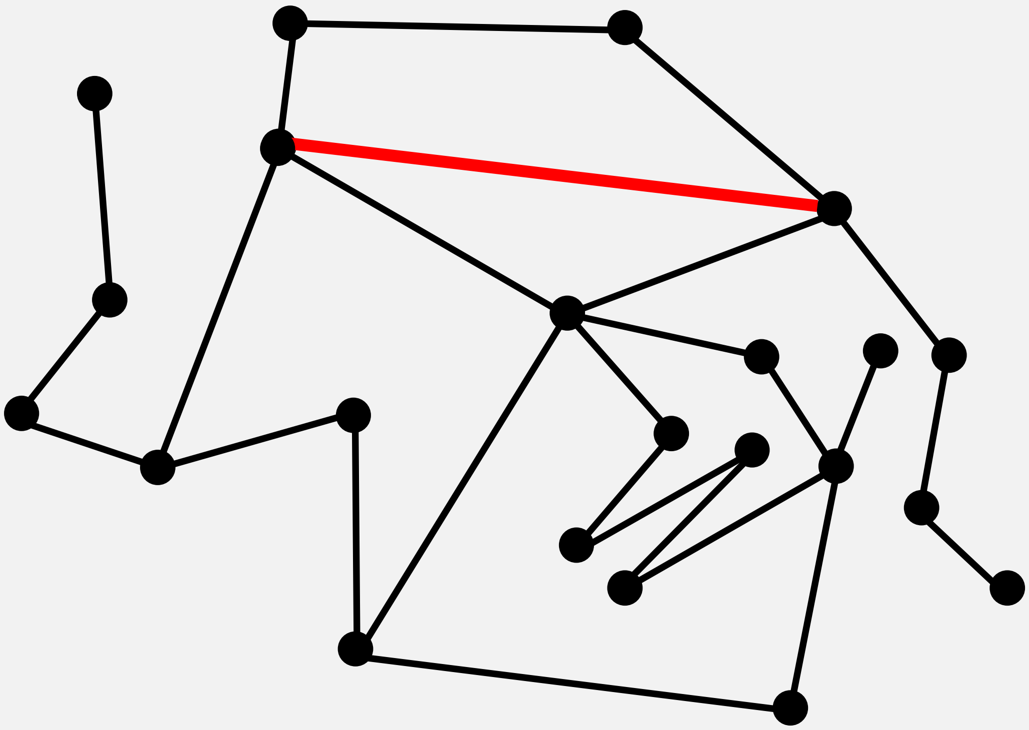
**Farshi et al., 2004**
plane Euclidean graphs in $R^d$
**Wulff-Nilsen, 2010**
graphs embedded in a metric space

### Adding k edges/dilation:

**Gudmundsson and Wong, 2022**
graphs embedded in a metric space

Goal: “**improve**” a graph

**by adding edges**

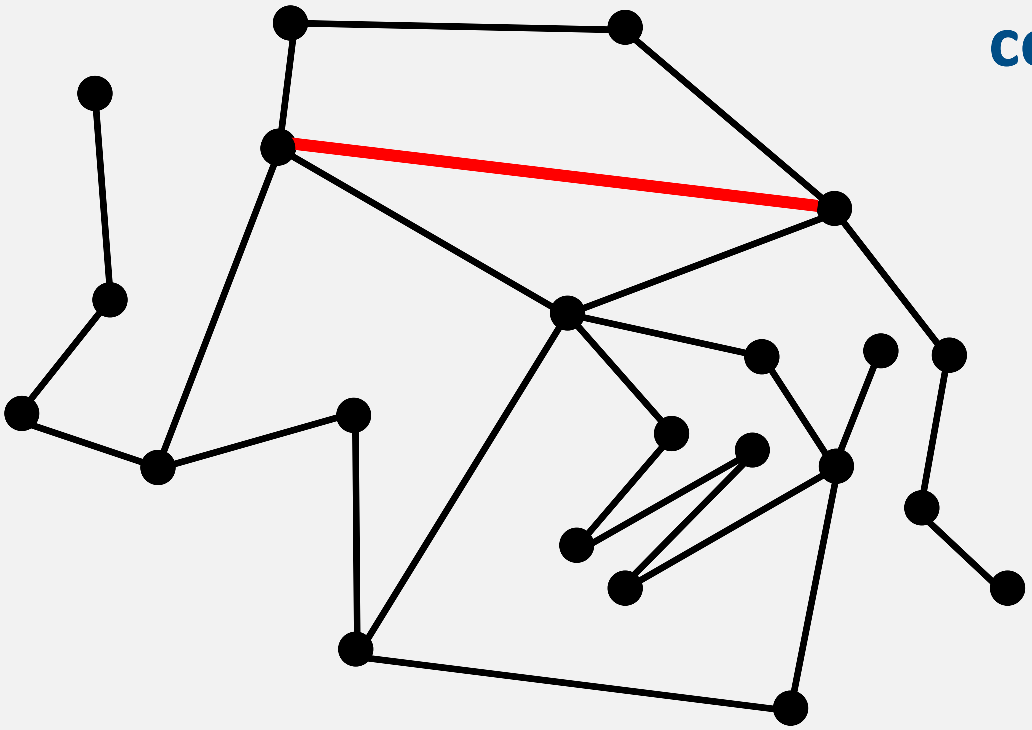


the locus of a graph

Goal: "**improve**" a ~~graph~~

by adding ~~edges~~

by adding segments
connecting any two points

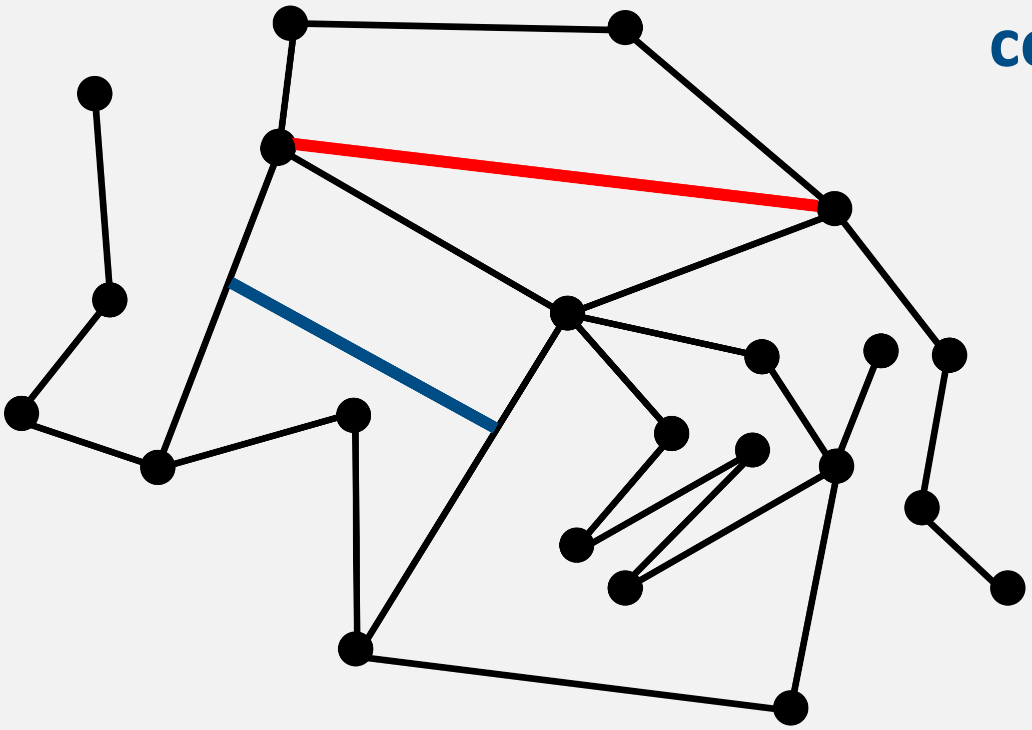


the locus of a graph

Goal: "**improve**" a ~~graph~~

by adding ~~edges~~

by adding segments connecting any two points

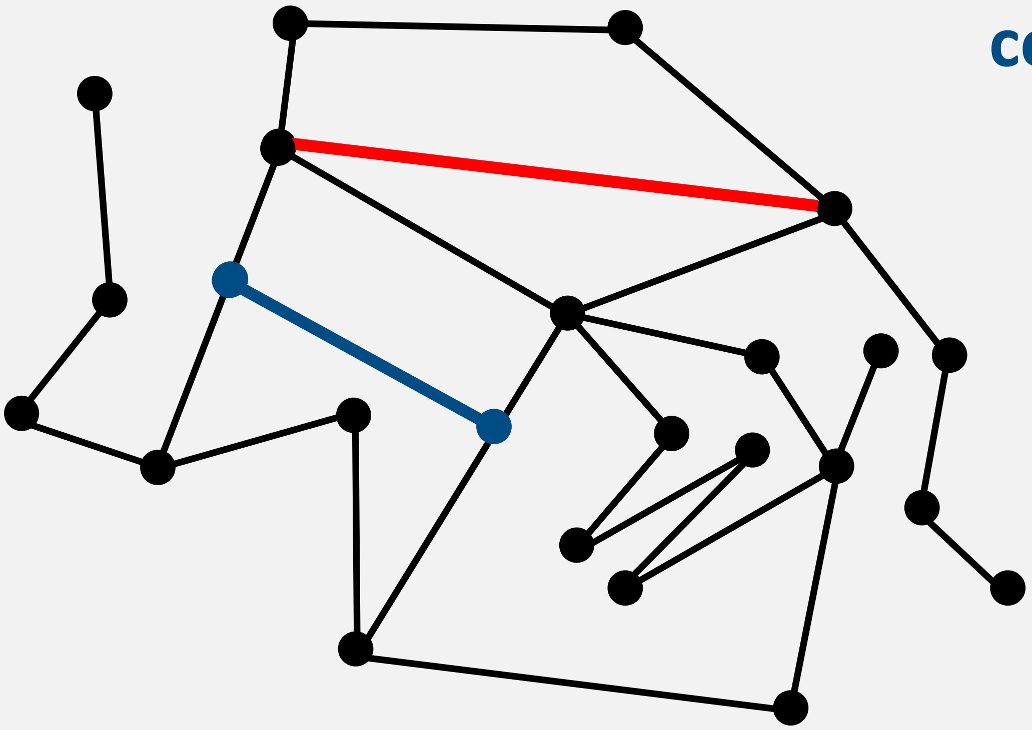


the locus of a graph

Goal: “**improve**” a ~~graph~~

by adding ~~edges~~

by adding segments connecting any two points

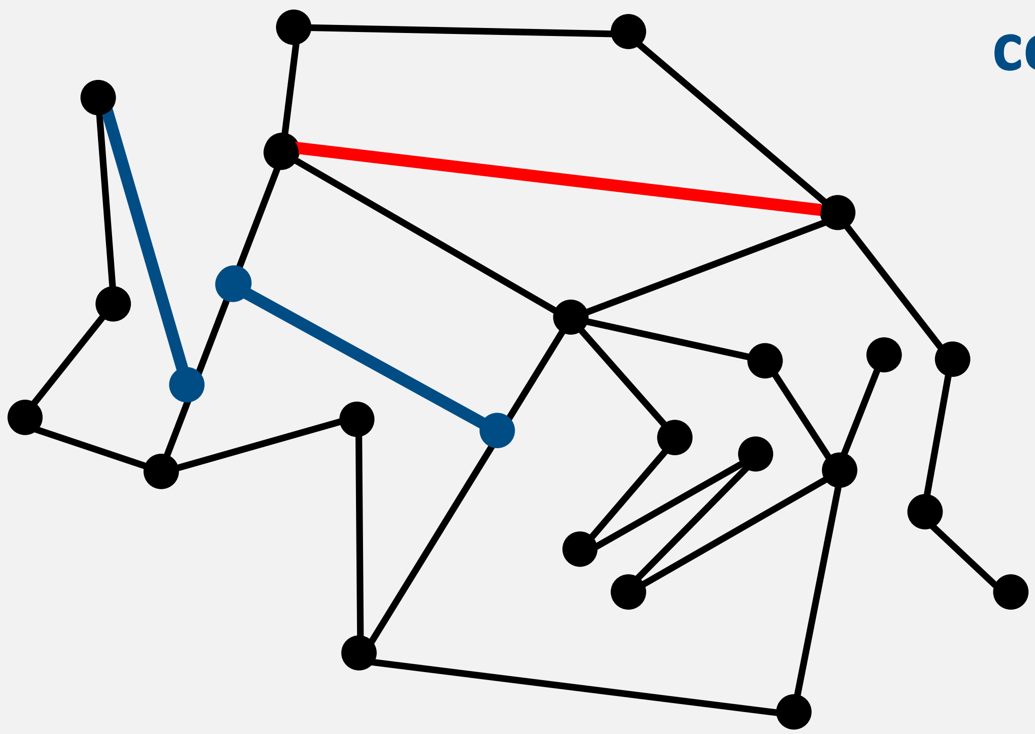


the locus of a graph

Goal: "**improve**" a graph

by adding edges

by adding segments connecting any two points

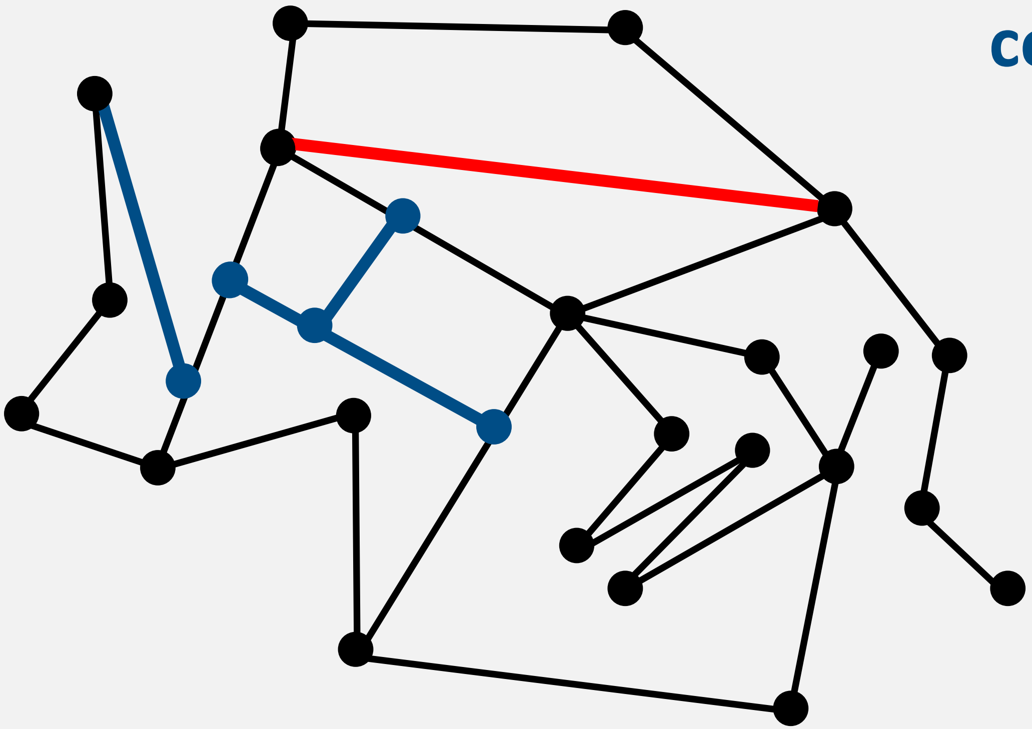


the locus of a graph

Goal: "**improve**" a graph

by adding edges

by adding segments connecting any two points

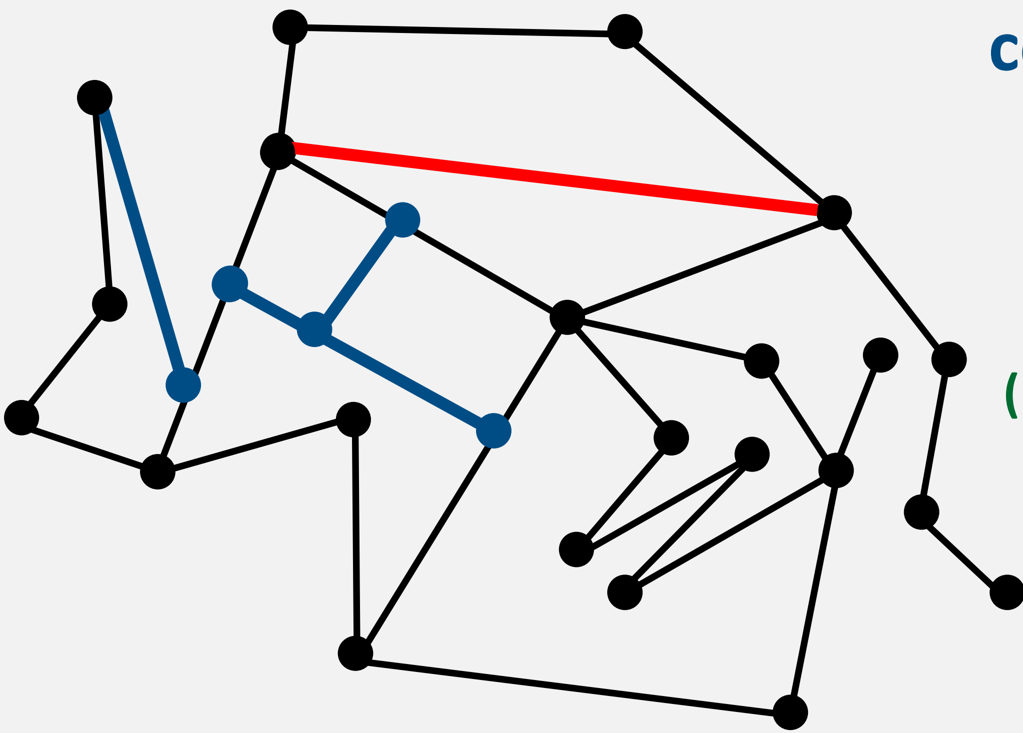


the locus of a graph

Goal: "**improve**" a ~~graph~~

by adding ~~edges~~

by adding segments
connecting any two points



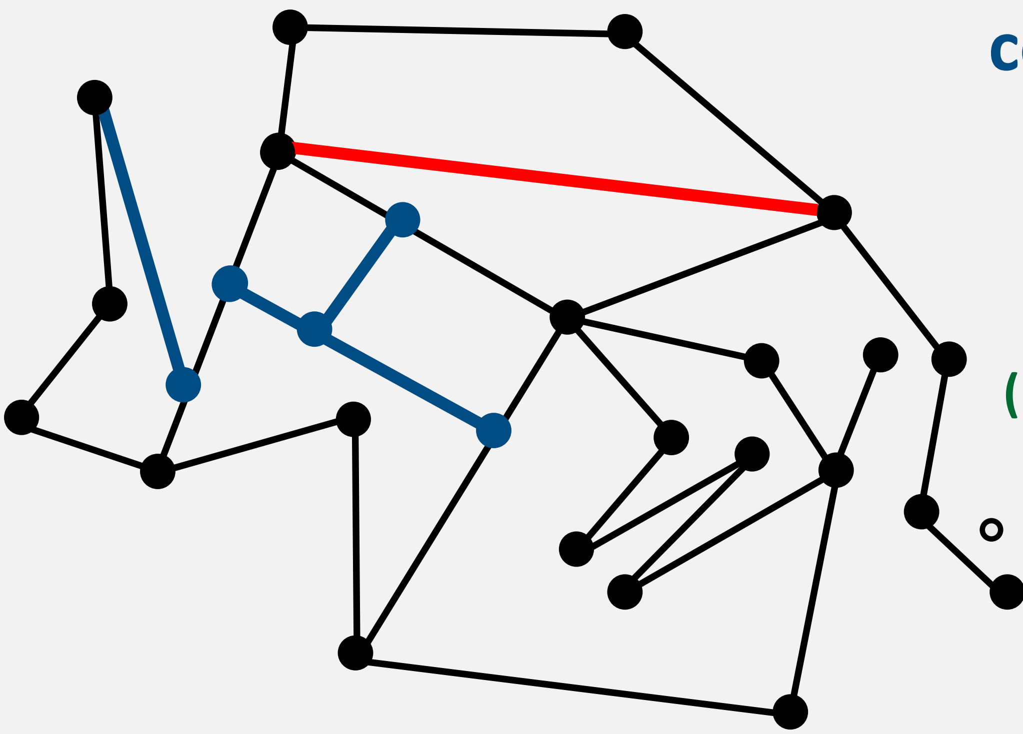
Reduce/minimize:

continuous diameter
(max distance between any two points)

Goal: “**improve**” a ~~graph~~ the locus of a graph

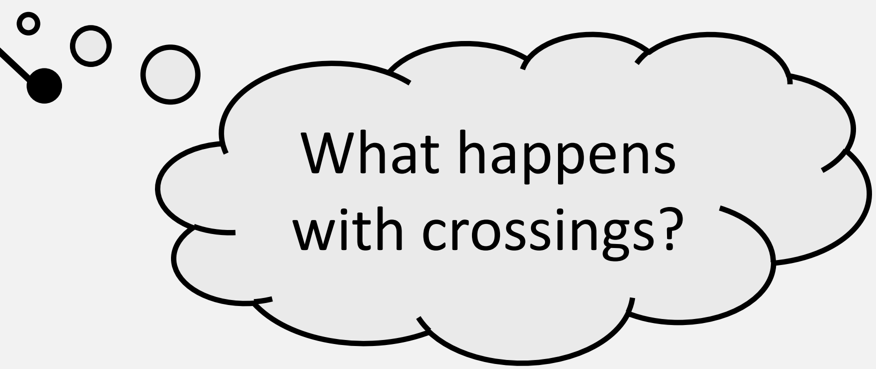
~~by adding edges~~

by adding segments connecting any two points



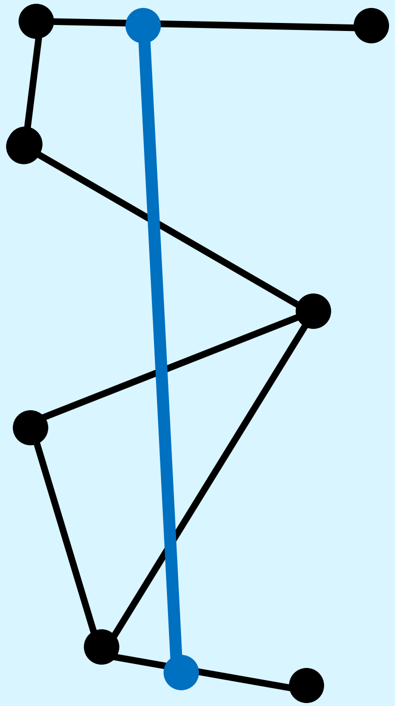
**Reduce/minimize:**

continuous diameter
(max distance between any two points)



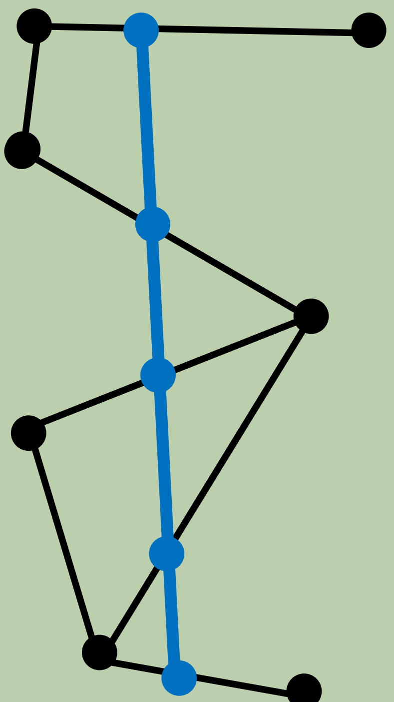
Goal: "**improve**" a graph

**by adding segments connecting any two points**



ignore crossings

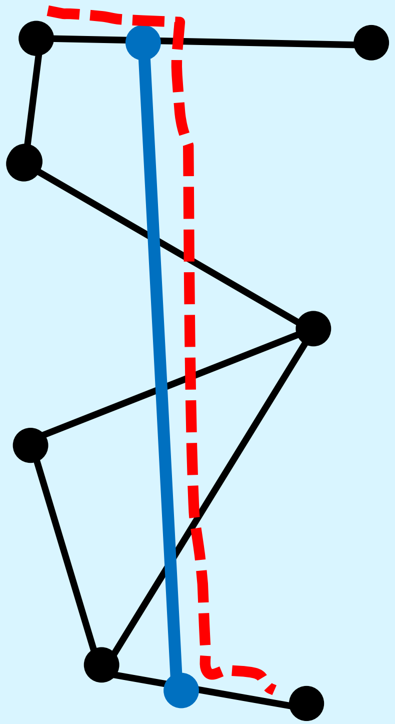
**Highway model**



Each crossing creates a new vertex

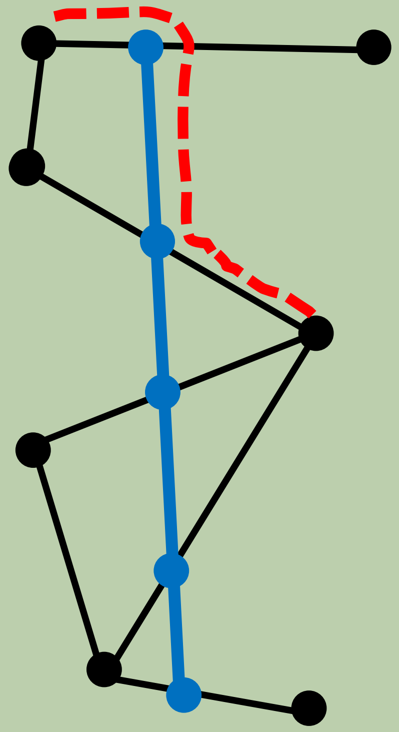
**Planar model**

# Highway model vs. Planar model



ignore crossings

**Highway model**

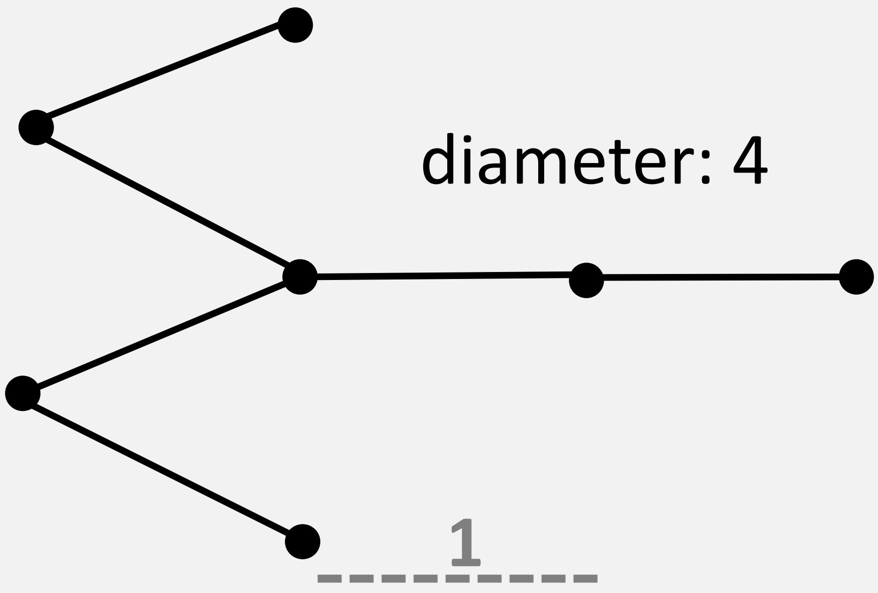


Each crossing creates a new vertex

**Planar model**

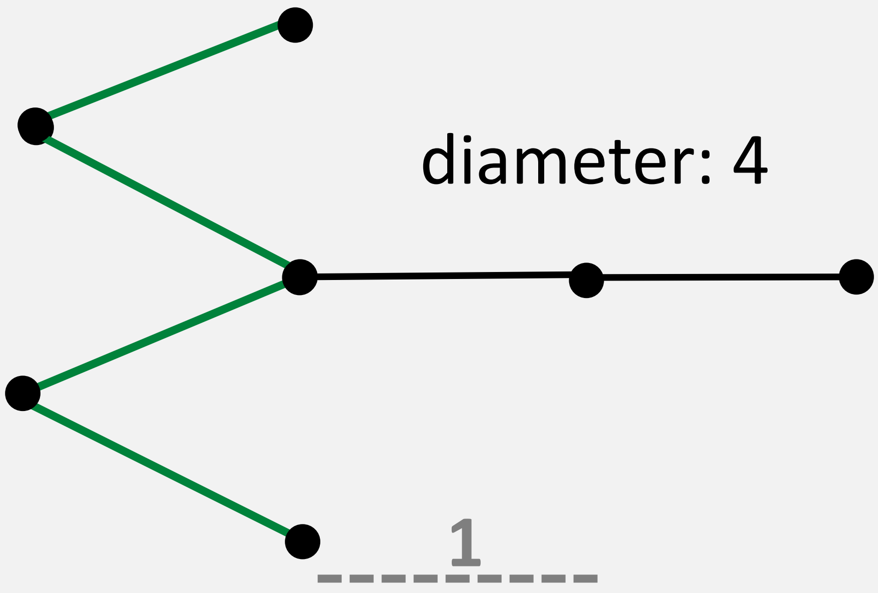
# Highway model vs. Planar model

**Reduce:** **continuous diameter (max distance between any two points)**



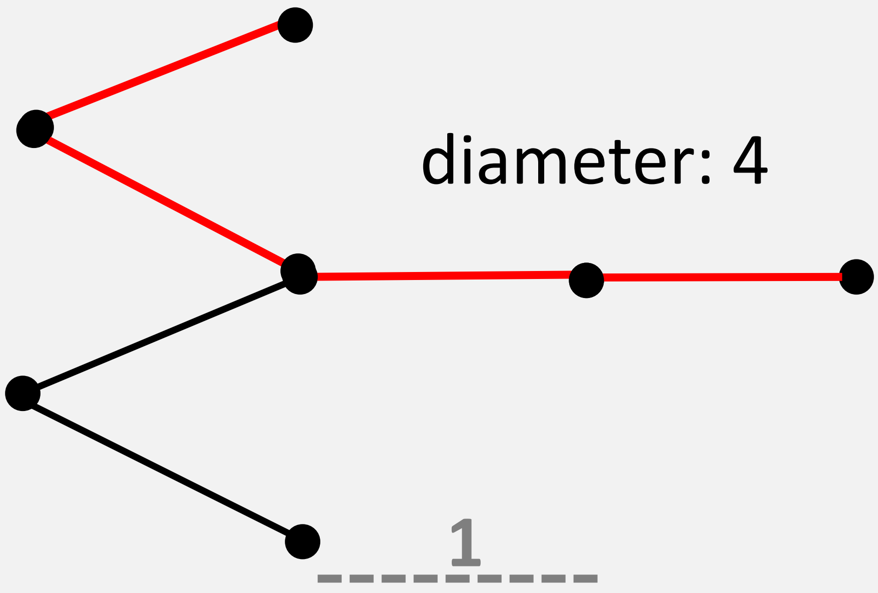
# Highway model vs. Planar model

**Reduce: continuous diameter (max distance between any two points)**



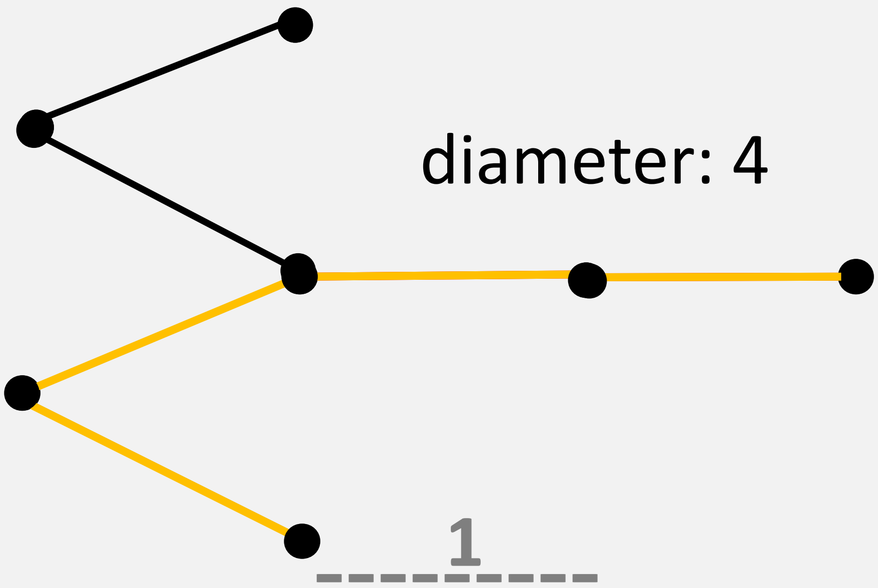
# Highway model vs. Planar model

**Reduce:** **continuous diameter (max distance between any two points)**



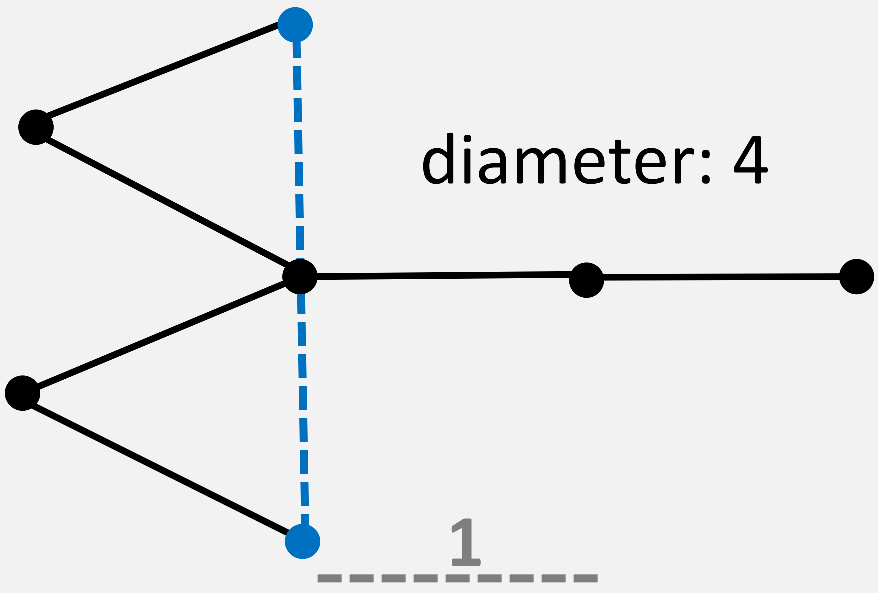
# Highway model vs. Planar model

**Reduce:** **continuous diameter  (max distance between any two points)**



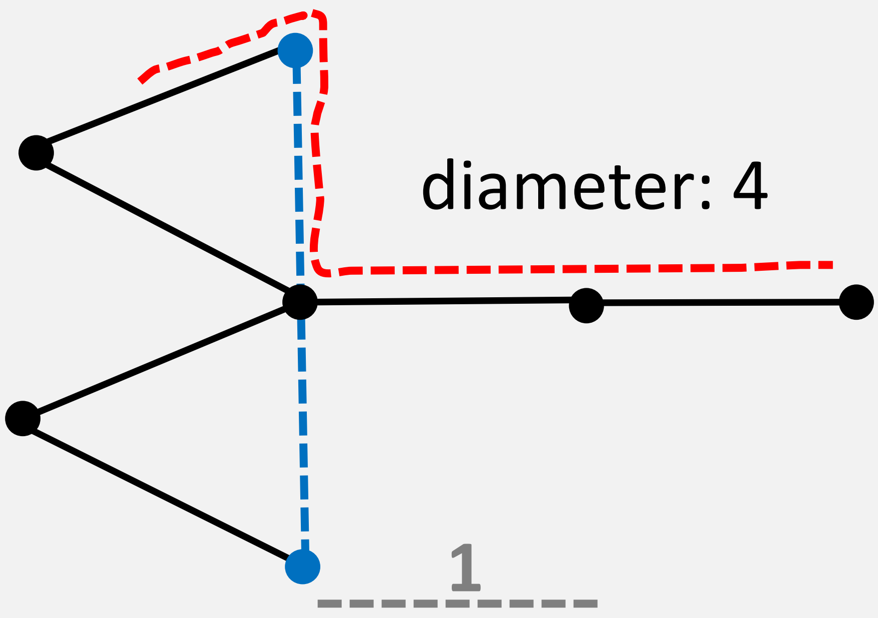
# Highway model vs. Planar model

**Reduce:** **continuous diameter (max distance between any two points)**



# Highway model vs. Planar model

**Reduce:** **continuous diameter (max distance between any two points)**

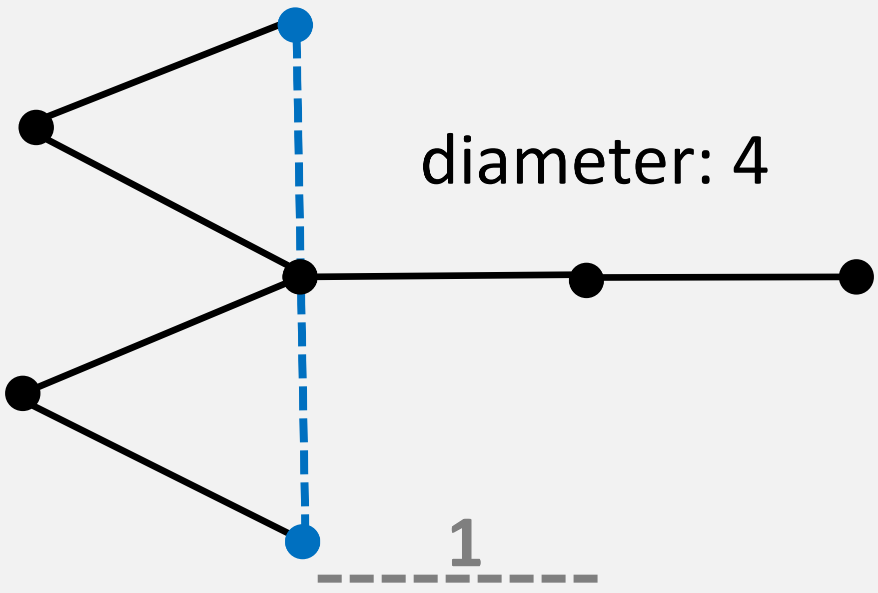


# Highway model vs. Planar model

**Reduce:** continuous diameter (max distance between any two points)

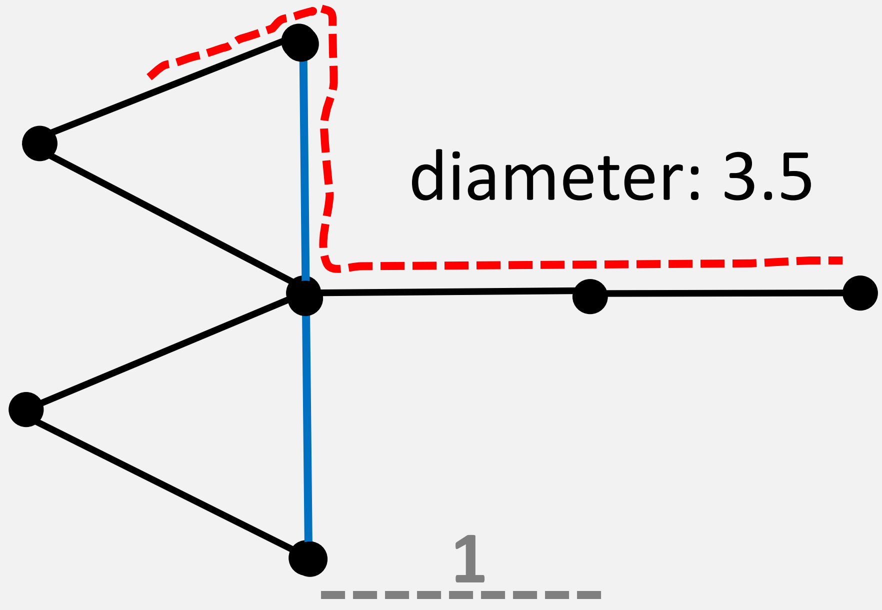
## Highway model

No single segment improves diameter




## Planar model

diameter improved



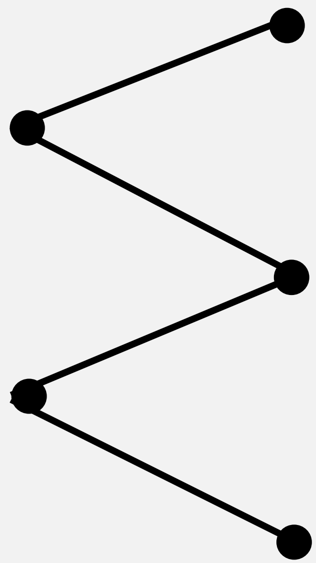
# Our problem:<br>find optimal shortcuts in the **planar model**

Given a graph, find k segments such that in the resulting graph the **continuous diameter** is minimum (over all posible segments)

# Our problem:
# find optimal shortcuts in the **planar model**

Given a graph, find k segments such that in the resulting graph the **continuous diameter** is minimum (over all posible segments)

k=1

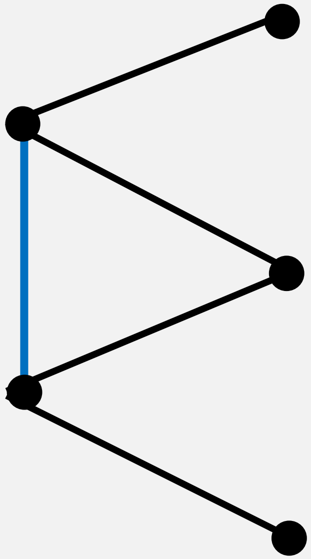


initial diameter: 4

# Our problem:
# find optimal shortcuts in the **planar model**

Given a graph, find k segments such that in the resulting graph the **continuous diameter** is minimum (over all posible segments)

k=1



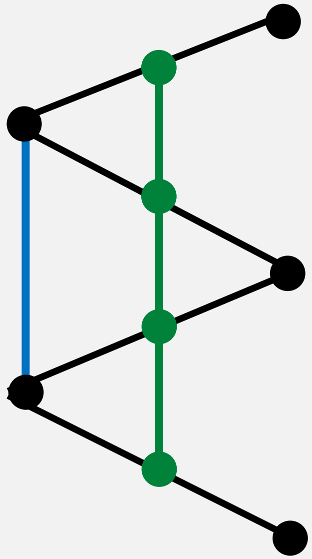
initial diameter: 4

**new diameter: 3** → **shortcut: improves**

# Our problem:
# find optimal shortcuts in the **planar model**

Given a graph, find k segments such that in the resulting graph the **continuous diameter** is minimum (over all posible segments)

k=1



initial diameter: 4

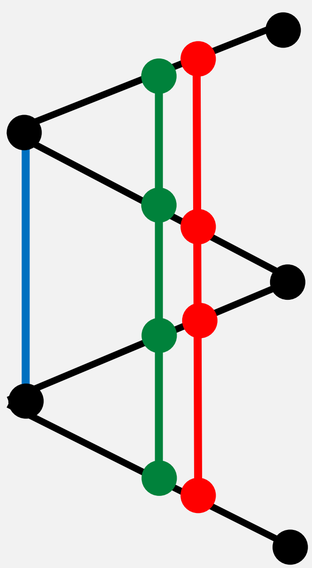
**new diameter: 3** → **shortcut: improves**

**new diameter: 2.5**

## Our problem:
## find optimal shortcuts in the **planar model**

Given a graph, find k segments such that in the resulting graph the **continuous diameter** is minimum (over all posible segments)

k=1



initial diameter: 4

**new diameter: 3** → **shortcut: improves**

**new diameter: 2.5**

**new diameter: 2.3** → **optimal shortcut**

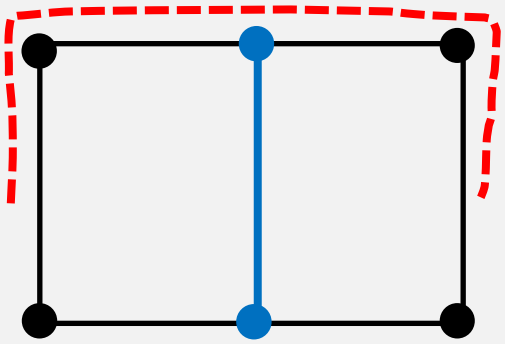
# Our problem:
# find optimal shortcuts in the **planar model**

Given a graph, find k segments such that in the resulting graph the **continuous diameter** is minimum (over all posible segments)

**no single shortcut exists**

# Our problem:
# find optimal shortcuts in the **planar model**

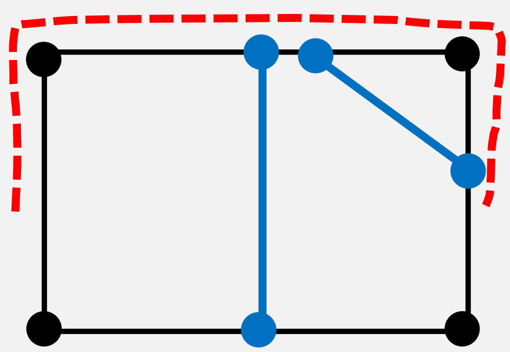
Given a graph, find k segments such that in the resulting graph the **continuous diameter** is minimum (over all posible segments)



**no single shortcut exists**

# Our problem:
# find optimal shortcuts in the **planar model**

Given a graph, find k segments such that in the resulting graph the **continuous diameter** is minimum (over all posible segments)



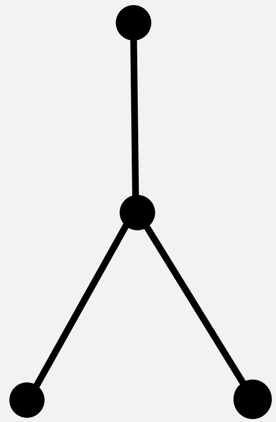
**no single shortcut exists**

**pair of shortcuts (k=2)**

# Our problem:
# find optimal shortcuts in the **planar model**

Given a graph, find k segments such that in the resulting graph the **continuous diameter** is minimum (over all posible segments)

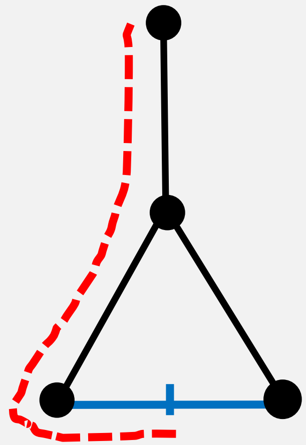
**Adding a segment may worsen the diameter!**



# Our problem:
# find optimal shortcuts in the **planar model**

Given a graph, find k segments such that in the resulting graph the **continuous diameter** is minimum (over all posible segments)

**Adding a segment may worsen the diameter!**



## Some results

**Minimize: continuous diameter**

### Highway model

**Optimal** set of k **shortcuts:**

**De Carufel et al. (2016):**
geometric paths (k=1),
geometric convex cycles (k=2)

**Oh and Ahn (2016):**
weighted trees (k=1)

**De Carufel et al. (2017):**
geometric trees (k=1)

**Bae et al. (2017):**
circles (k≤7)

### Planar model

**Approximation algorithms-optimal shortcut (k=1)**

Yang (2013): certain types of paths

**General geometric graphs-optimal shortcut (k=1)**

G., Márquez, Rodríguez, Silveira (2019)

## Some results

**Minimize: continuous diameter**

### Planar model

**General geometric graphs-optimal shortcut (k=1)**

G., Márquez, Rodríguez, Silveira (2019)

**General graphs**

- Computation of **optimal shortcut** (k=1) is polynomial
- Approximation possible via discretization

**Paths**

- Compute diameter after inserting segment **$\Theta(n)$**
- Compute optimal horizontal shortcut **$O(n^2 \log n)$**
- Compute optimal simple shortcut **$O(n^2)$**

Some results

**Minimize: continuous diameter**

## Planar model

**General geometric graphs-optimal shortcut (k=1)**

G., Márquez, Rodríguez, Silveira (2019)

How fast can an optimal shortcut in general graphs be computed?

## Some results

**Minimize: continuous diameter**

### Planar model

**General geometric graphs-optimal shortcut (k=1)**

G., Márquez, Rodríguez, Silveira (2019)

How fast can an optimal shortcut in ~~general graphs~~ be computed?

**in paths? (any orientation)**

**in trees?**

## Some results

**Minimize: continuous diameter**

**Planar model**

**General geometric graphs-optimal shortcut (k=1)**

G., Márquez, Rodríguez, Silveira (2019)

How fast can an optimal shortcut in ~~general graphs~~ be computed?

**in paths? (any orientation)**

**in trees?**

**Existence of shortcuts in general geometric graphs**

G., Klute, Márquez, Parada, Silveira (2023)

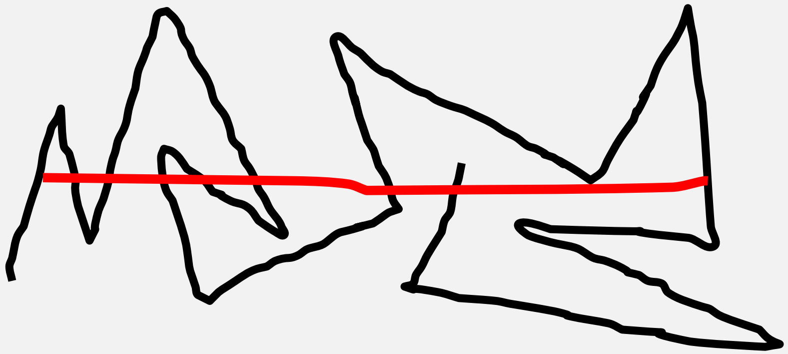
- It is 3SUM-hard to decide if a graph admits a shortcut (k=1)
- It is NP-complete and APX-hard to decide if a graph admits a set of k shortcuts

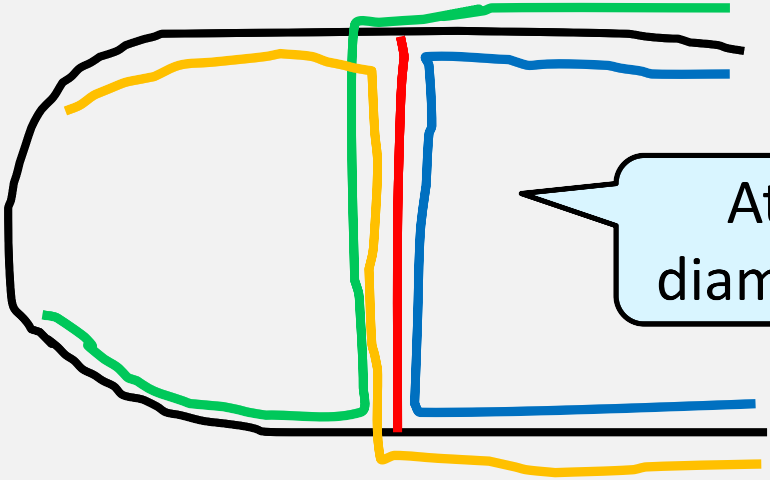
# Paths: not so simple

Highway model: no crossings

# Paths: not so simple

Highway model: no crossings

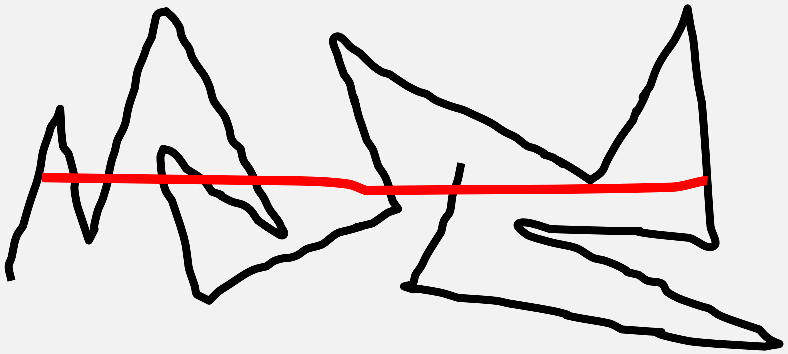


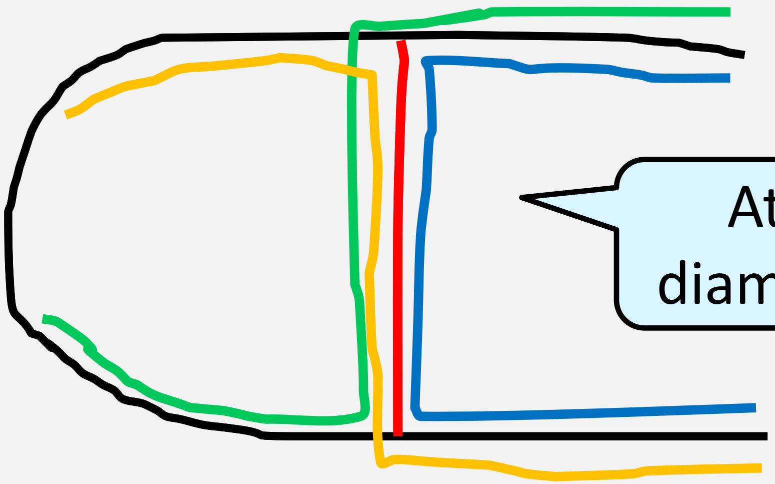


At most 3 diametral pairs

# Paths: not so simple

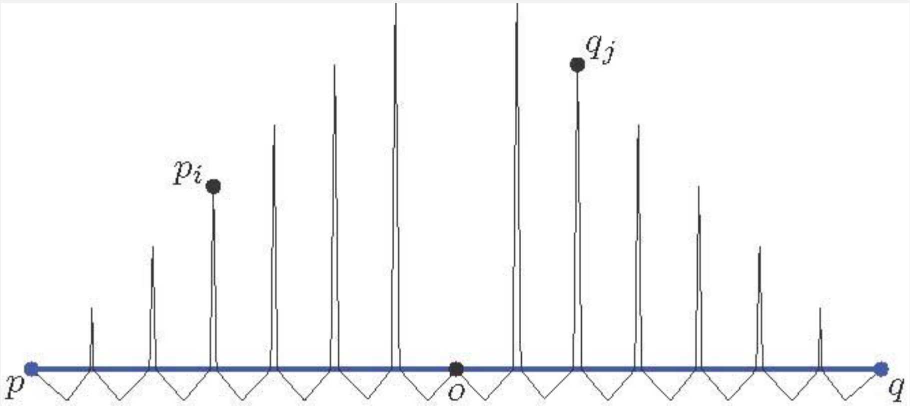
Highway model: no crossings





At most 3 diametral pairs

Planar model




Quadratic number of diametral pairs

# Computation of the continuous diameter

(G., Márquez, Silveira, 2018 and 2023): the **continuous diameter** of a plane geometric graph and the **continuous mean distance** of a plane weighted graph can be computed in quadratic time.

# Computation of the continuous diameter

(G., Márquez, Silveira, 2018 and 2023): the **continuous diameter** of a plane geometric graph and the **continuous mean distance** of a plane weighted graph can be computed in quadratic time.

Cabello (2017): Subquadratic algorithms for the **diameter** and the **sum of pairwise distances** in planar graphs

# Computation of the continuous diameter

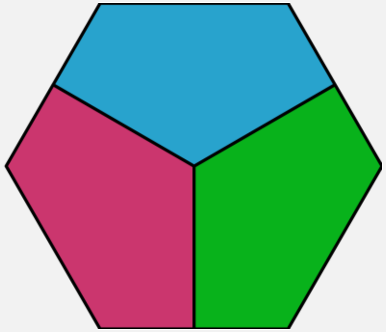
(G., Márquez, Silveira, 2018 and 2023): the **continuous diameter** of a plane geometric graph and the **continuous mean distance** of a plane weighted graph can be computed in quadratic time.

**SUBQUADRATIC???**

**graphs with bounded treewidth**

Cabello (2017): Subquadratic algorithms for the **diameter** and the **sum of pairwise distances** in planar graphs

# Other type of problem: Borsuk number

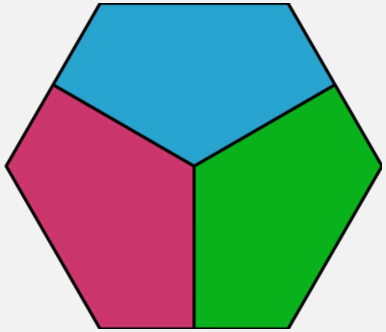


Is it true that every set in $R^n$ can be partitioned into n + 1 closed (sub)sets of smaller diameter?

Answered in the positive for:
n = 2, Borsuk (1932)
n = 3, Perkal (1947)
All n for smooth convex bodies, Hadwiger (1946)
All n for centrally-symmetric bodies, Riesling (1971)
All n for bodies of revolution, Dekster (1995)

The general answer is NO, Kahn and Kalai (1993)
Their construction shows that n + 1 pieces do not suffice for n > 2014

# Other type of problem: Borsuk number

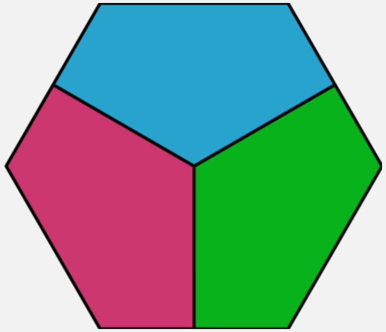


Is it true that every set in $R^n$ can be partitioned into n + 1 closed (sub)sets of smaller diameter?

Answered in the positive for:
n = 2, Borsuk (1932)
n = 3, Perkal (1947)
All n for smooth convex bodies, Hadwiger (1946)
All n for centrally-symmetric bodies, Riesling (1971)
All n for bodies of revolution, Dekster (1995)

The general answer is NO, Kahn and Kalai (1993)
Their construction shows that n + 1 pieces do not suffice for n > 2014

# Other type of problem: Borsuk number

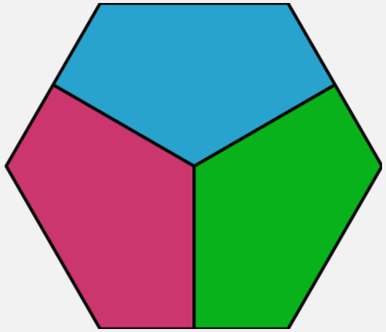


Is it true that every set in $R^n$ can be partitioned into n + 1 closed (sub)sets of smaller diameter?

Answered in the positive for:
n = 2, Borsuk (1932)
n = 3, Perkal (1947)
All n for smooth convex bodies, Hadwiger (1946)
All n for centrally-symmetric bodies, Riesling (1971)
All n for bodies of revolution, Dekster (1995)

The general answer is NO, Kahn and Kalai (1993)
Their construction shows that n + 1 pieces do not suffice for n > 2014

# Other type of problem: Borsuk number



Is it true that every set in $R^n$ can be partitioned into n + 1 closed (sub)sets of smaller diameter?

Answered in the positive for:
n = 2, Borsuk (1932)
n = 3, Perkal (1947)
All n for smooth convex bodies, Hadwiger (1946)
All n for centrally-symmetric bodies, Riesling (1971)
All n for bodies of revolution, Dekster (1995)

The general answer is NO, Kahn and Kalai (1993)
Their construction shows that n + 1 pieces do not suffice for n > 2014

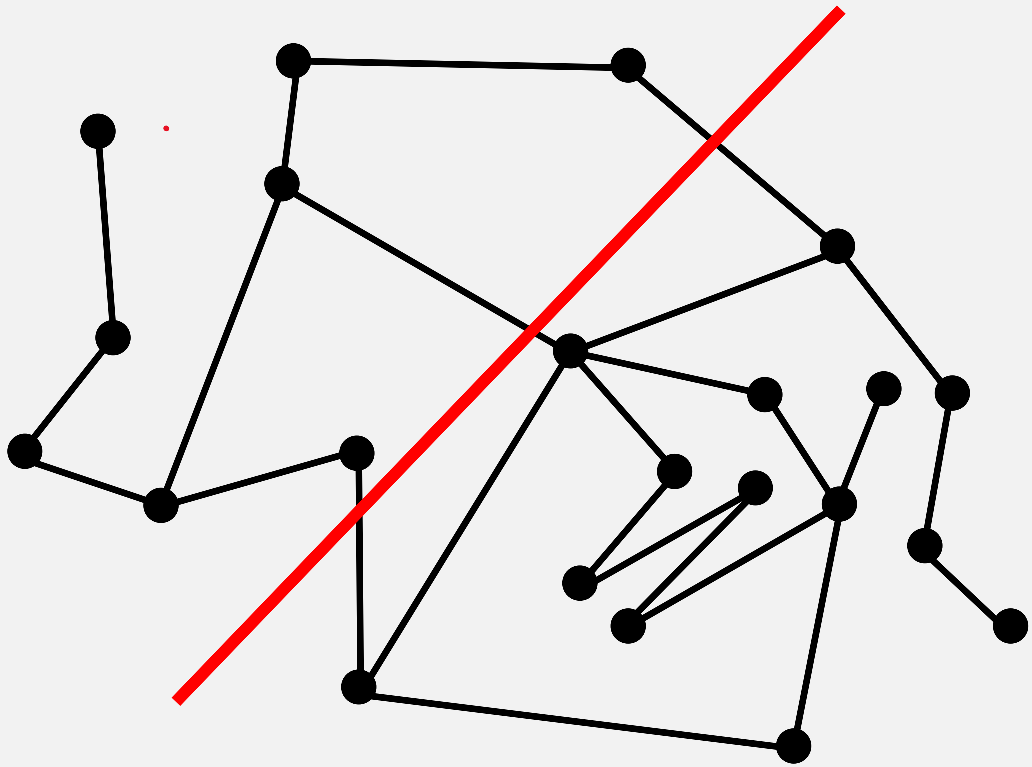
Borsuk number: #pieces needed to obtain max{diameters} < original diameter

# Other type of problem: Borsuk number

**Borsuk number**

minimum number of connected components to obtain (after cutting by lines) :

max{continuous diameters} < original continuous diameter

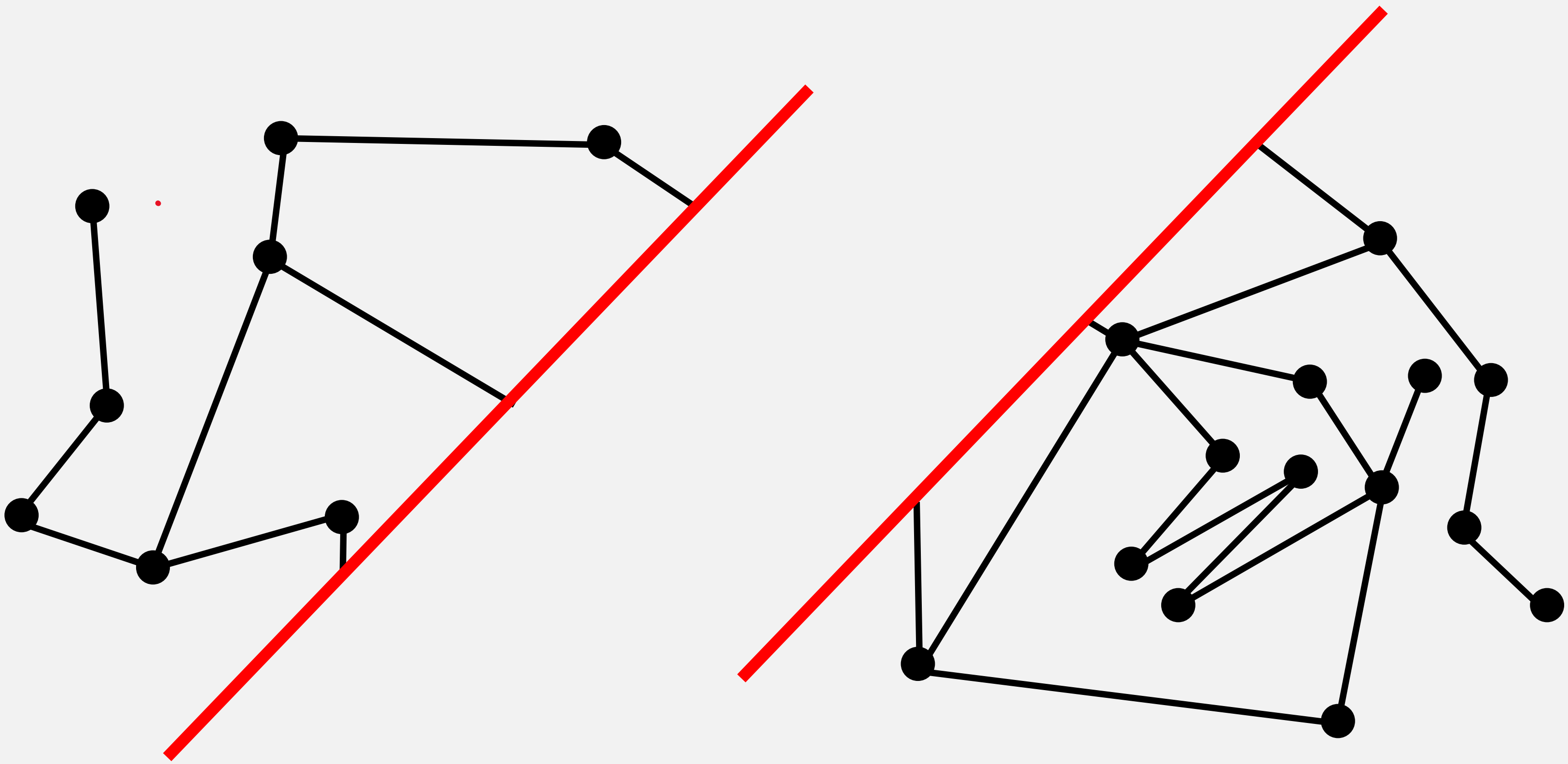


# Other type of problem: Borsuk number

**Borsuk number**

minimum number of connected components to obtain (after cutting by lines) :

max{continuous diameters} < original continuous diameter

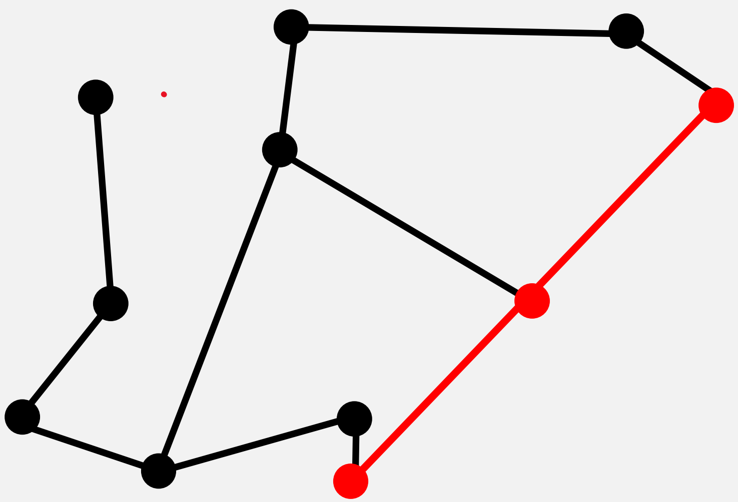


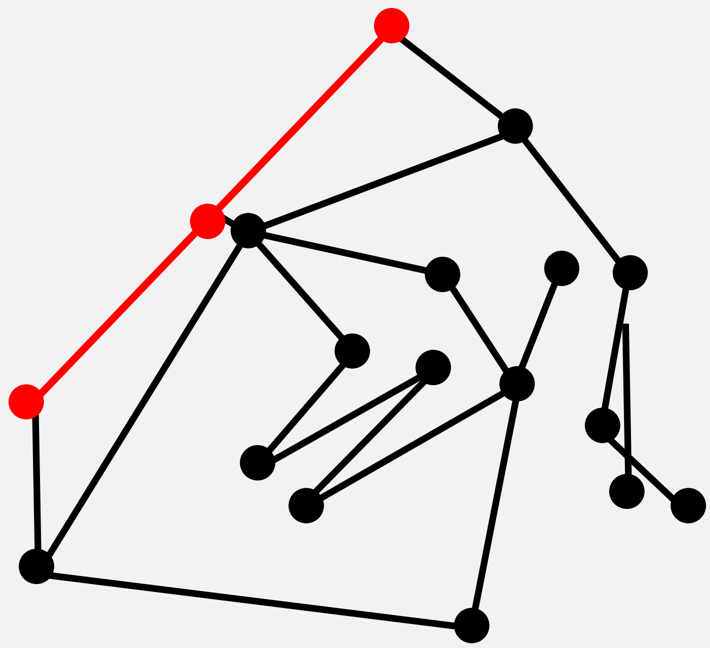
# Other type of problem: Borsuk number

**Borsuk number**

minimum number of connected components to obtain (after cutting by lines) :

max{continuous diameters} < original continuous diameter



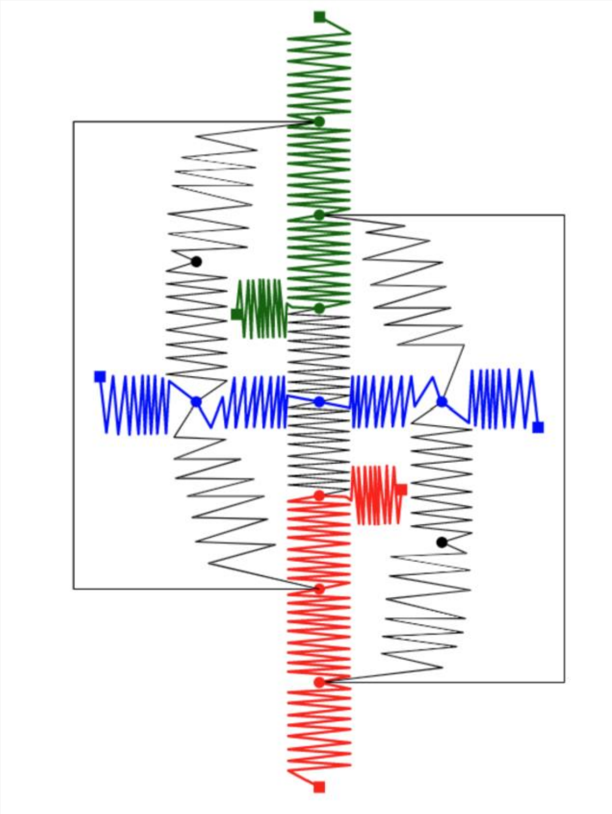


# Other type of problem: Borsuk number

**Borsuk number**

minimum number of connected components to obtain (after cutting by lines) :

max{continuous diameters} < original continuous diameter



# Gracias