



ISKO-Maghreb



جامعة بجاية
Tasdawit n Bgayet
Université de Béjaïa



ISKO MAGHREB 2018

The concept of a Pseudo-polygon and its application to Big Data and IoT

Ahcène Bounceur

November 25th, 2018



ANR Project: PERSEPTEUR



- 3D Smart City and IoT WSN simulator (SCI-WSN)
- 3D Signal propagation and interferences
- Physical and spatial modeling of the radio channel
- Mobility and dynamicity
- Visualization and optimization of the radio frequency space
- Designing reliable networks (QoS, test and Security)



Laurent Clavier (IEMN)



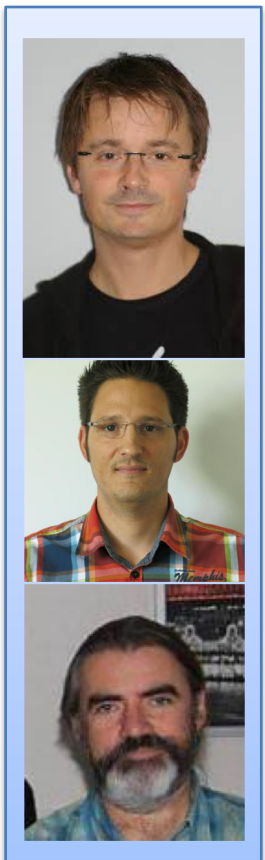
Pierre Combeau (Xlim)



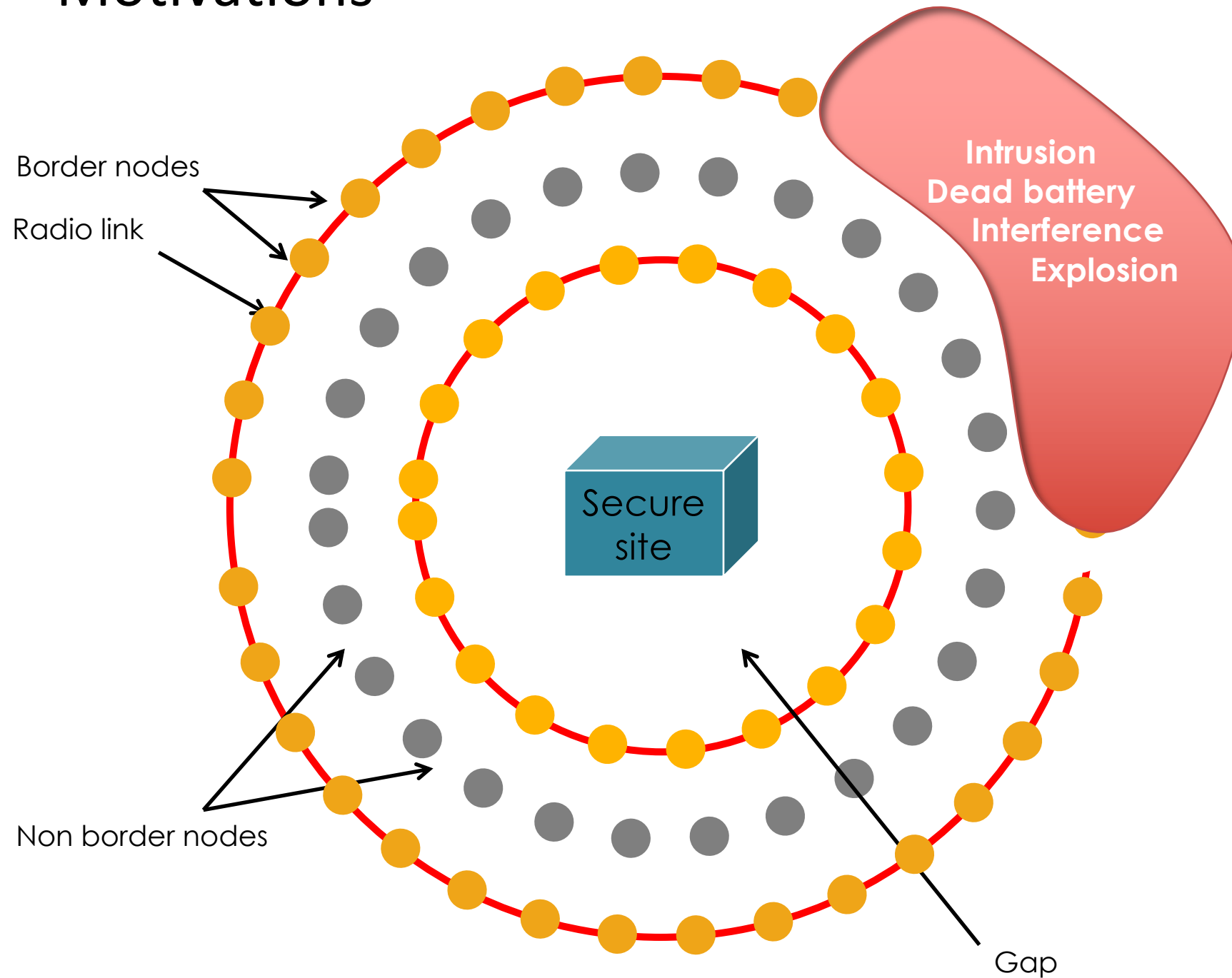
Virtualys
Virtual Vision & Process

Olivier Marc (Virtualys)

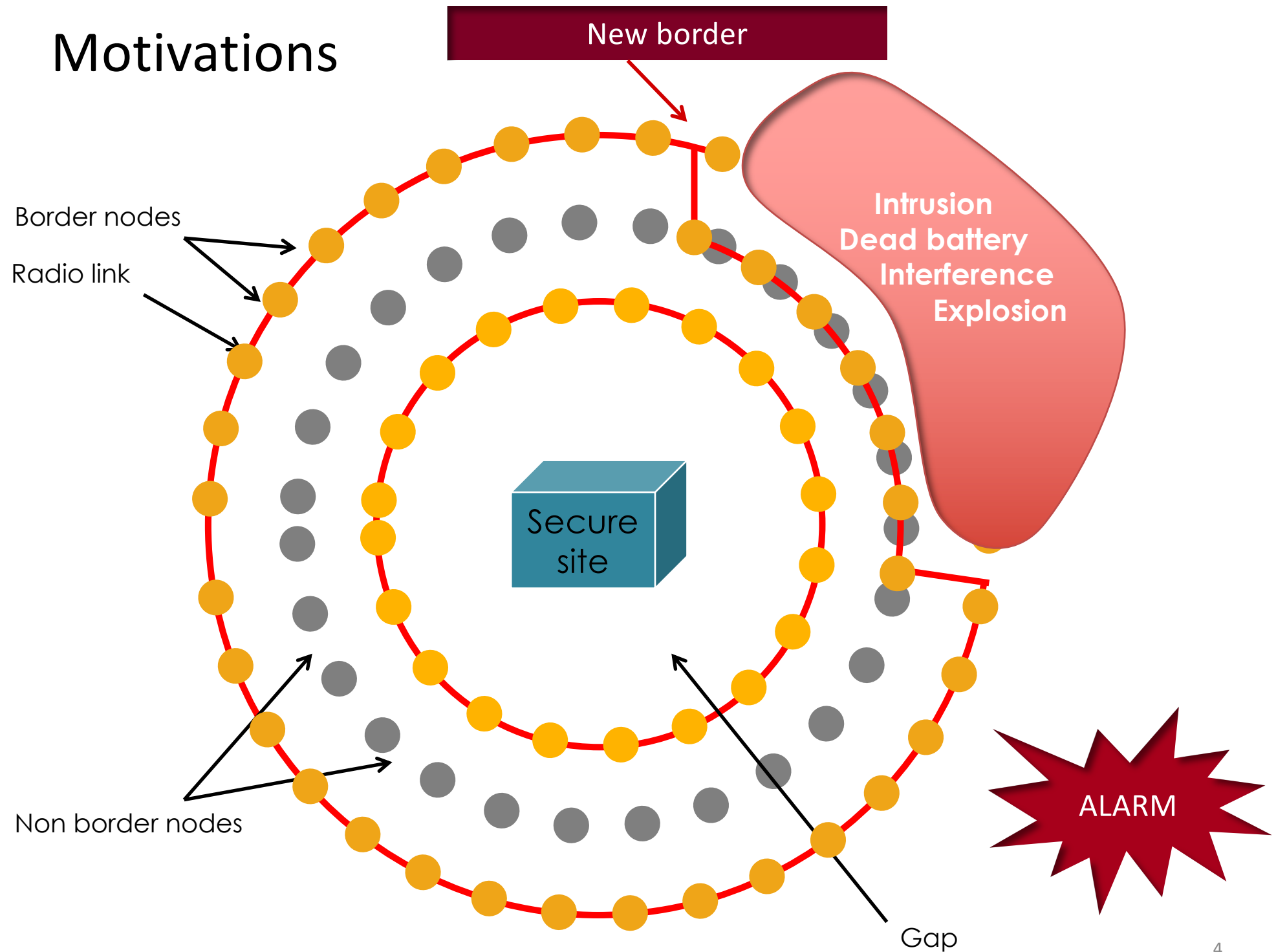
Partners



Motivations



Motivations



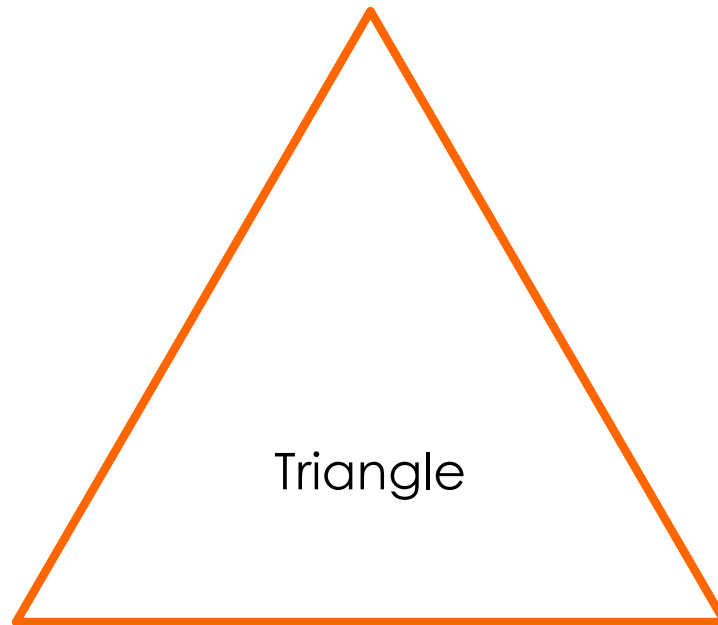
Outline

- Part 1: Polygons and Pseudo-Polygons
- Part 2: CupCarbon for IoT Simulation
- Part 3: Polygon Hulls and Leader Election
- Part 4: Some applications
- Part 5: Conclusion and Future work

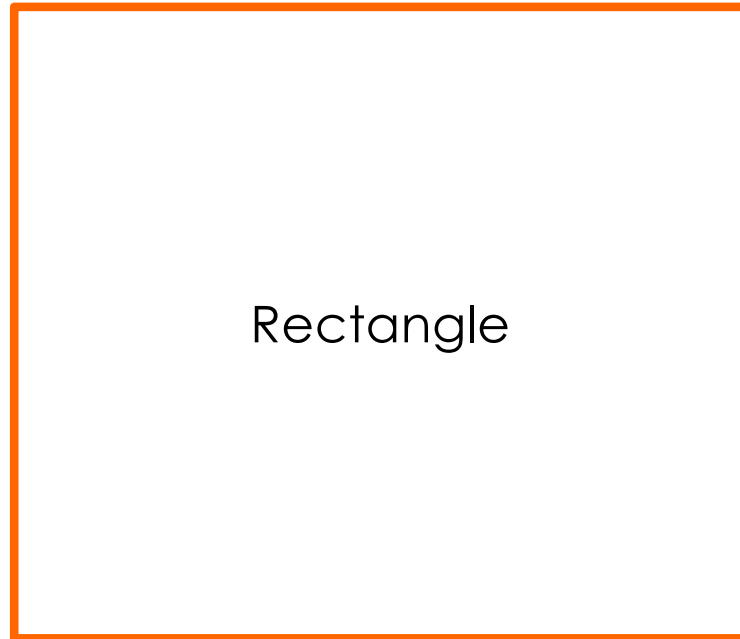
1

POLYGONS AND PSEUDO- POLYGONS

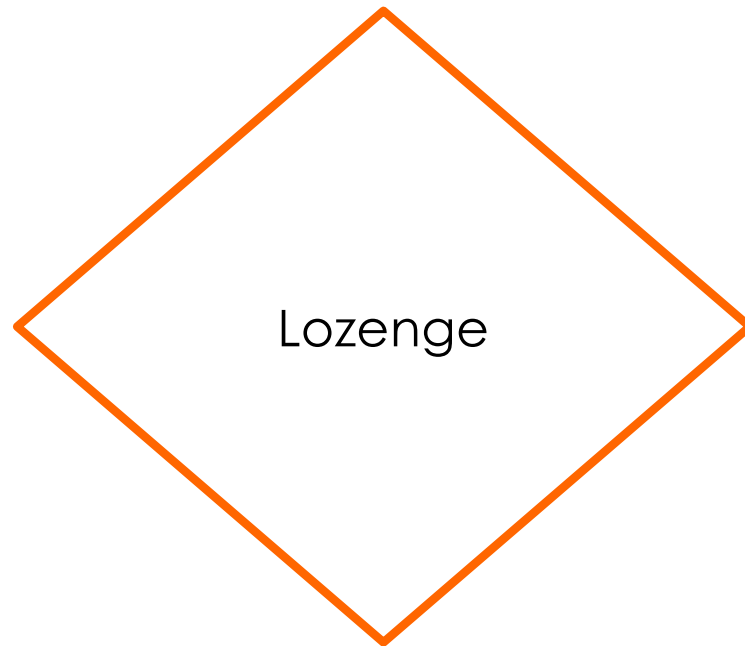
What is a Polygon?



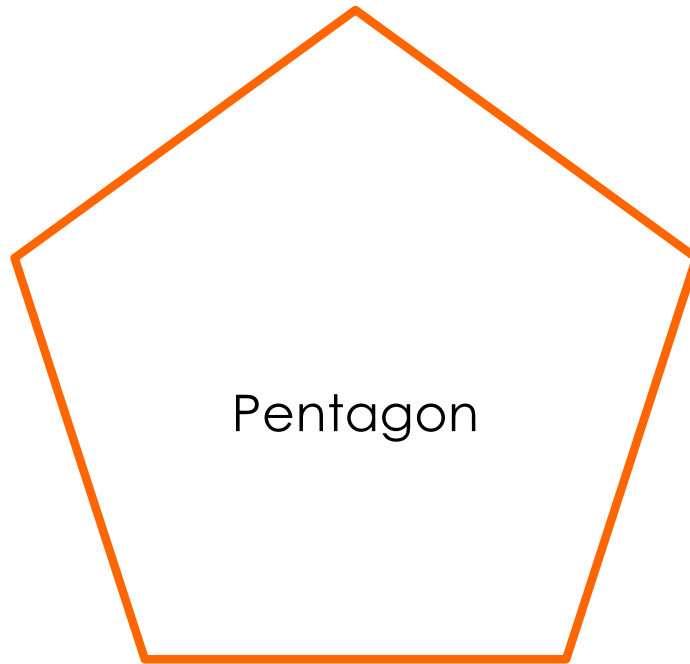
What is a Polygon?



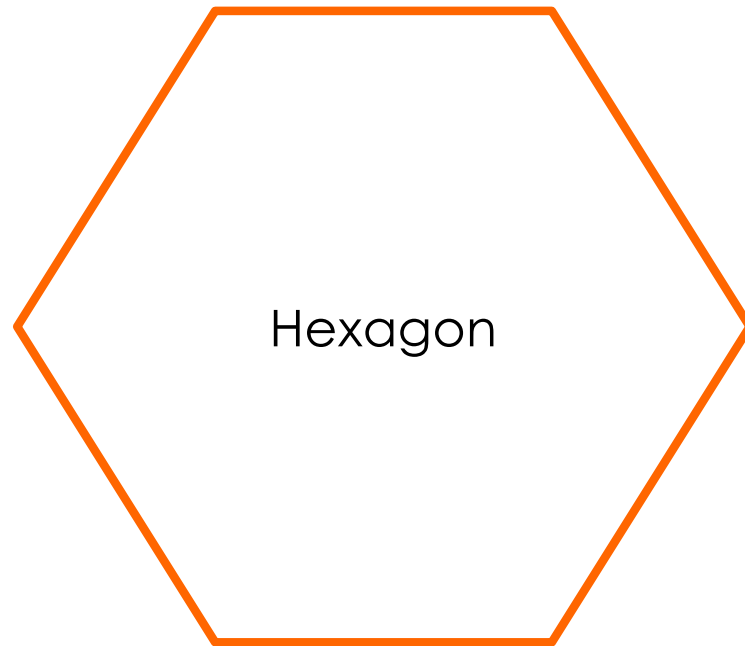
What is a Polygon?



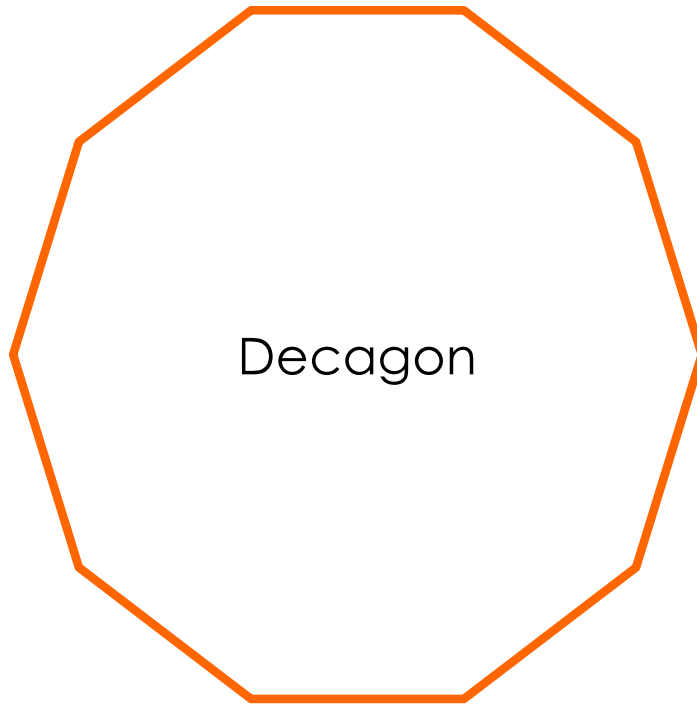
What is a Polygon?



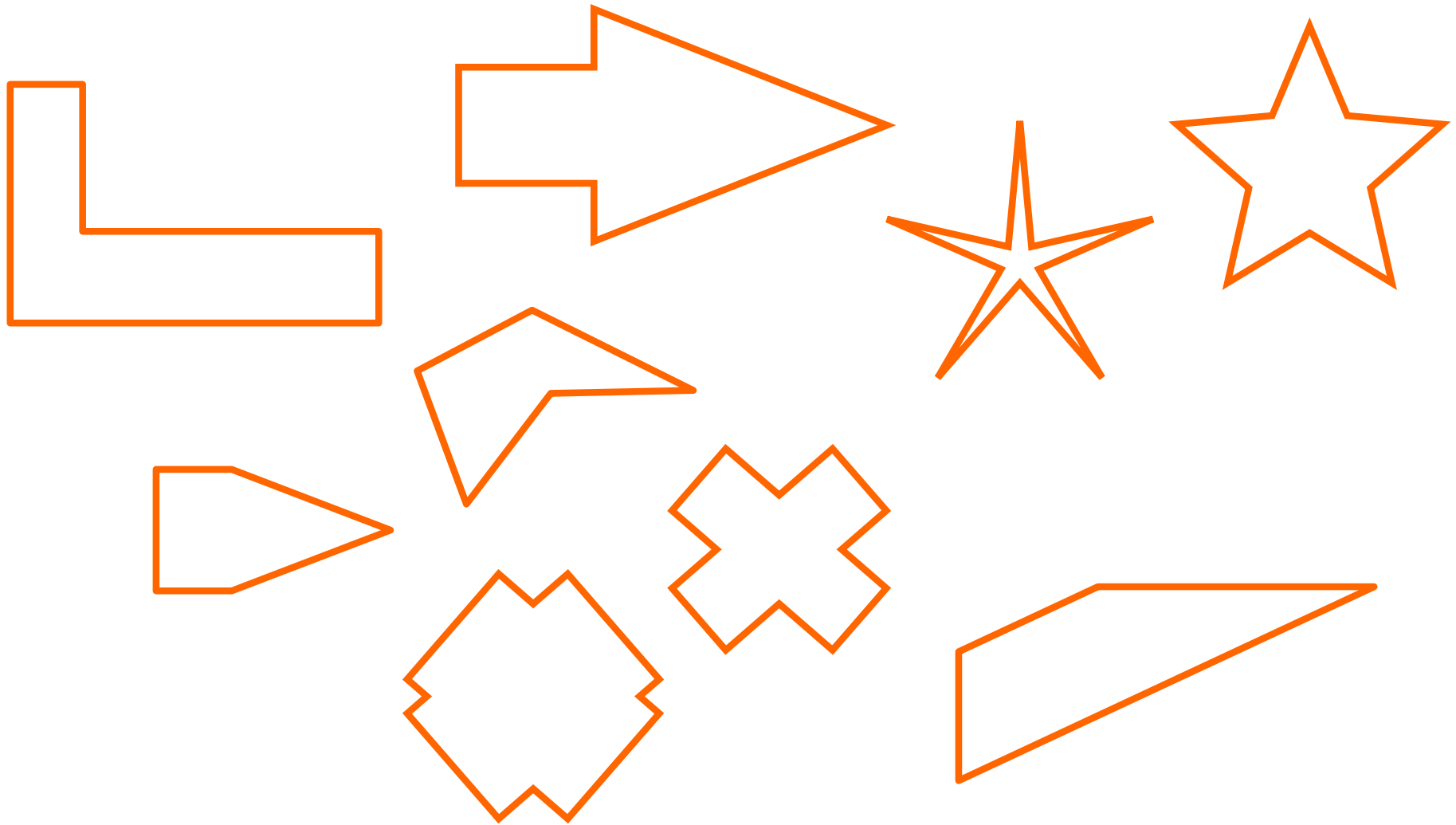
What is a Polygon?



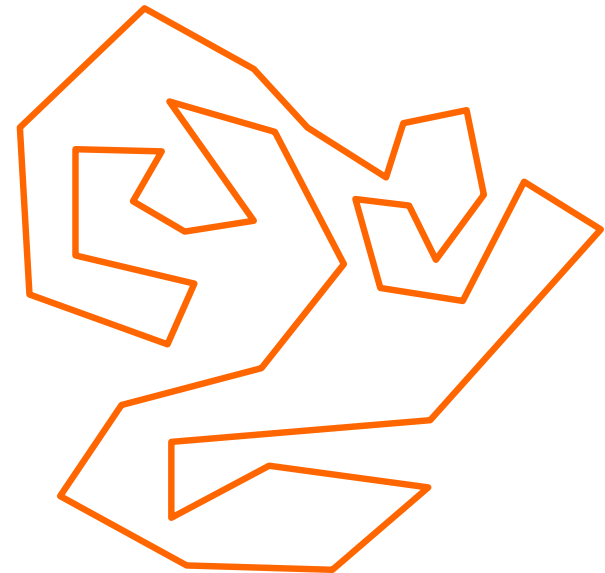
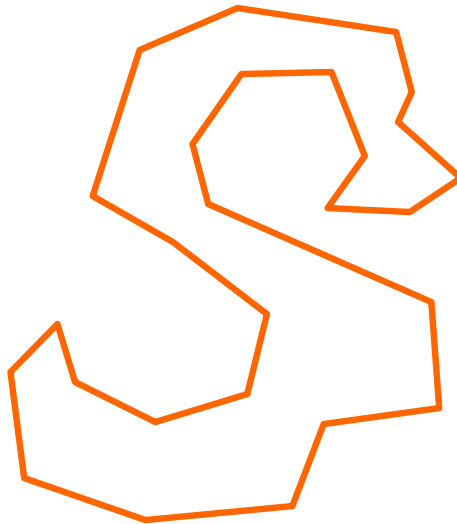
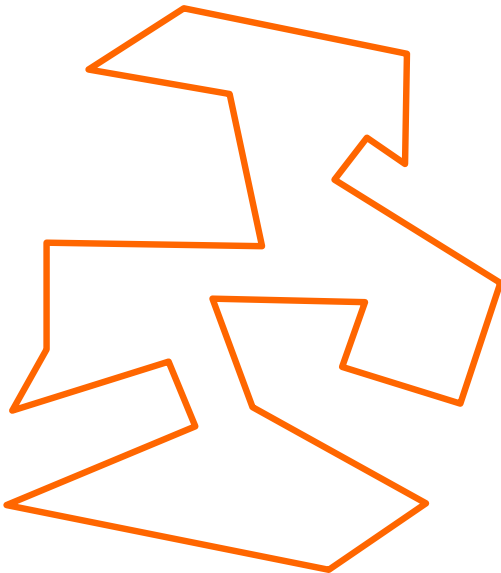
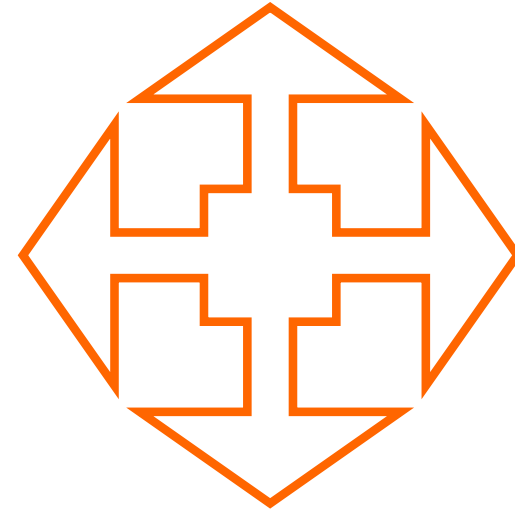
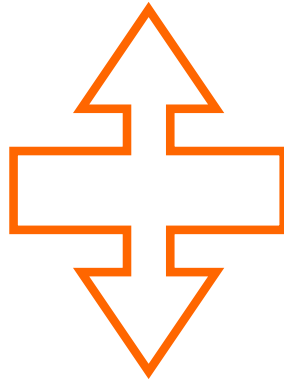
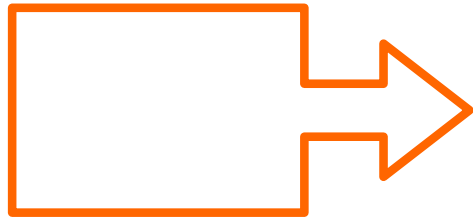
What is a Polygon?



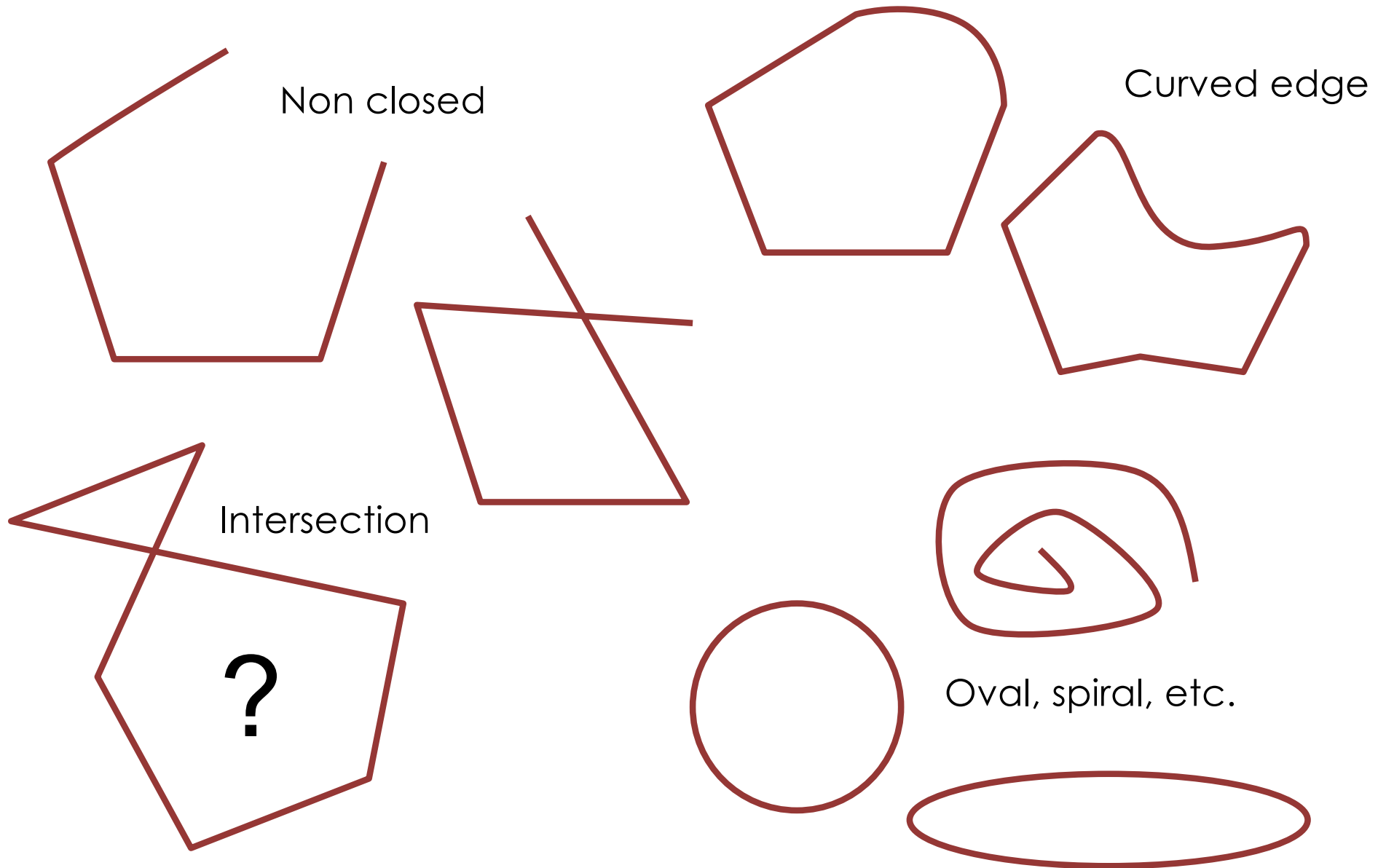
What is a Polygon?



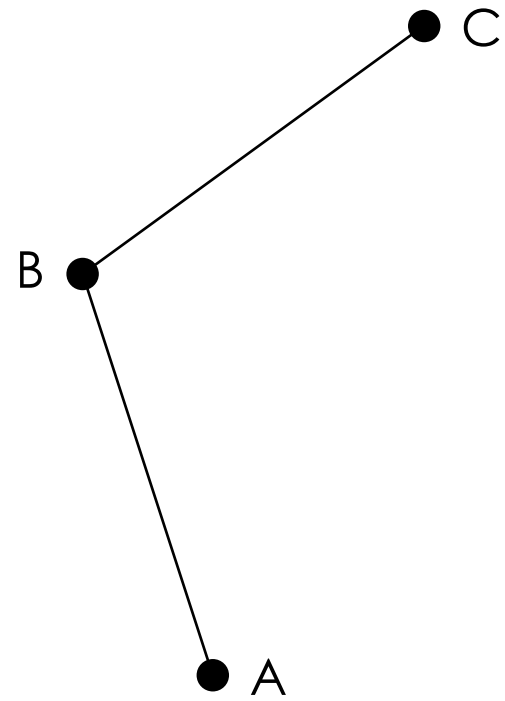
What is a Polygon?



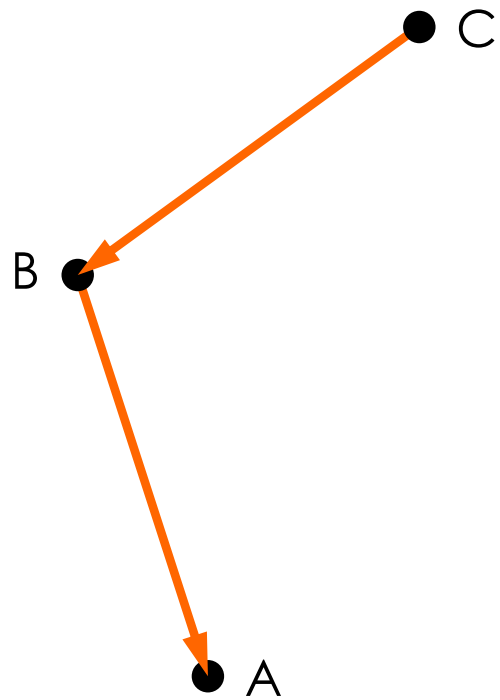
What is a NON-Polygon?



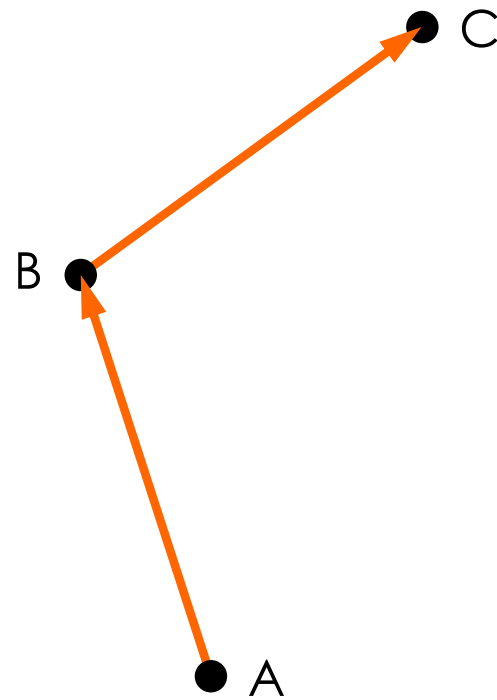
A 2 HOP VISIT



A 2 HOP VISIT

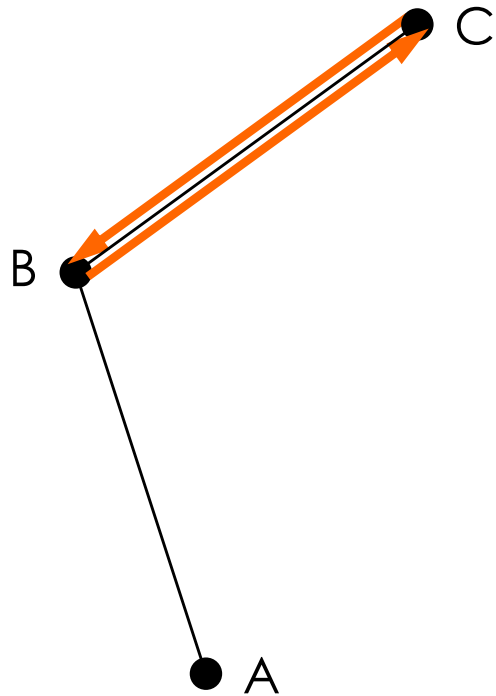


$C \rightarrow B \rightarrow A$

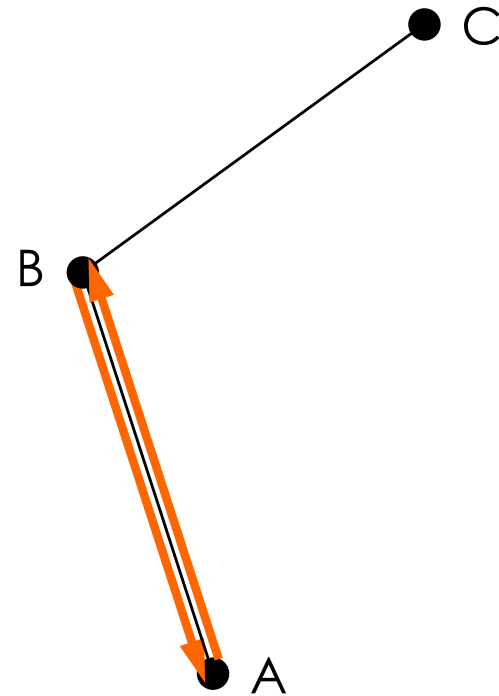


$A \rightarrow B \rightarrow C$

A 2 HOP VISIT

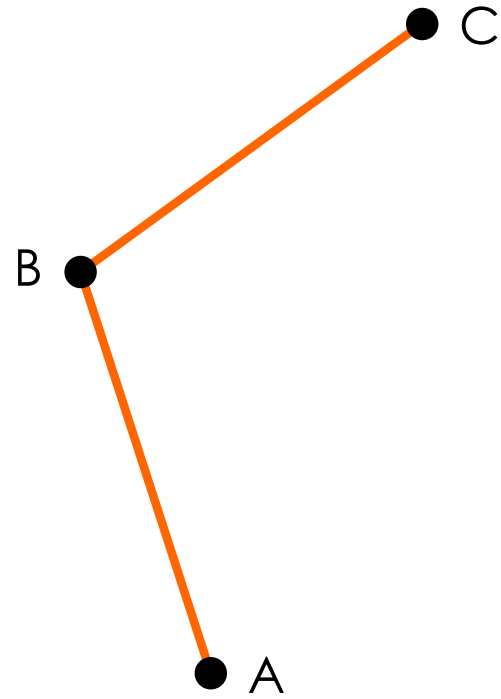


B → C → B
OU
C → B → C

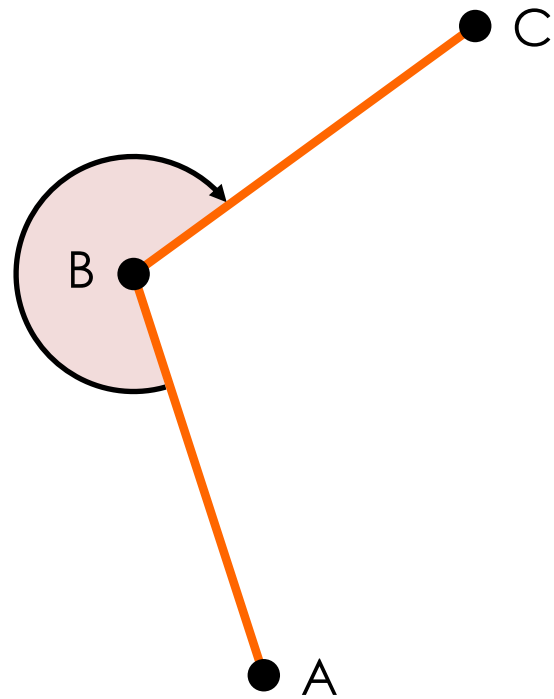


B → A → B
OU
A → B → A

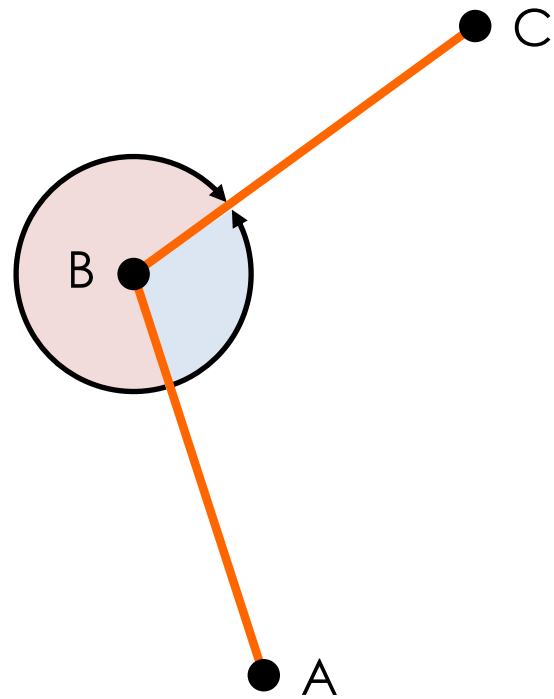
An Angle



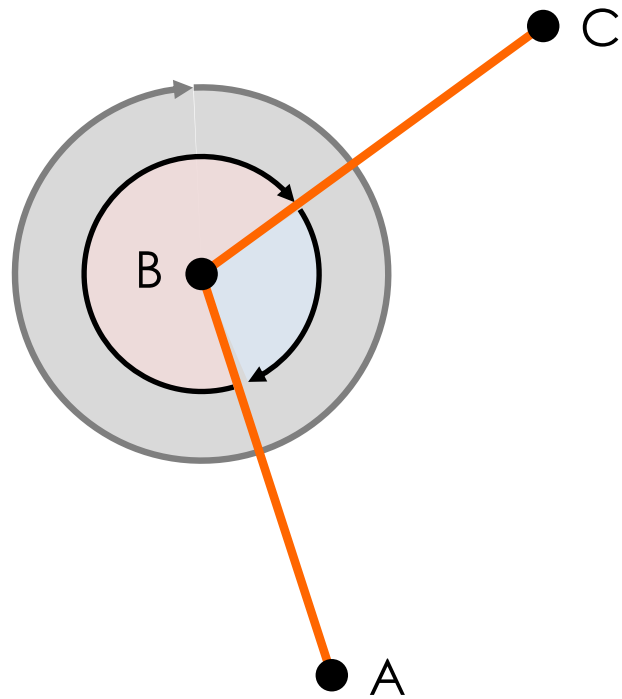
An Angle



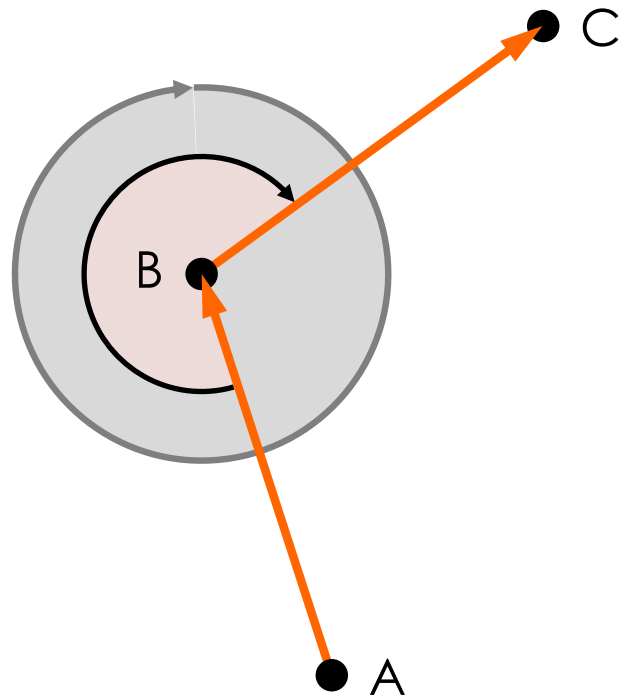
An Angle



A POLAR Angle

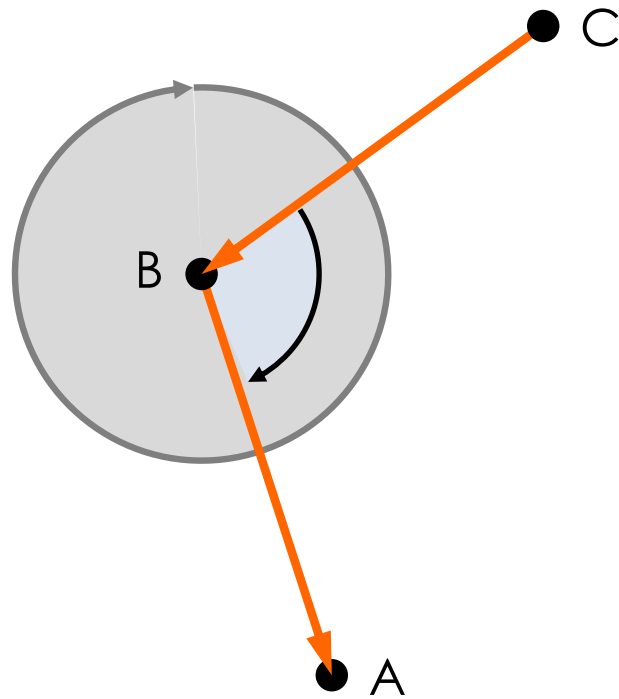


A 2 HOP VISIT and a POLAR Angle



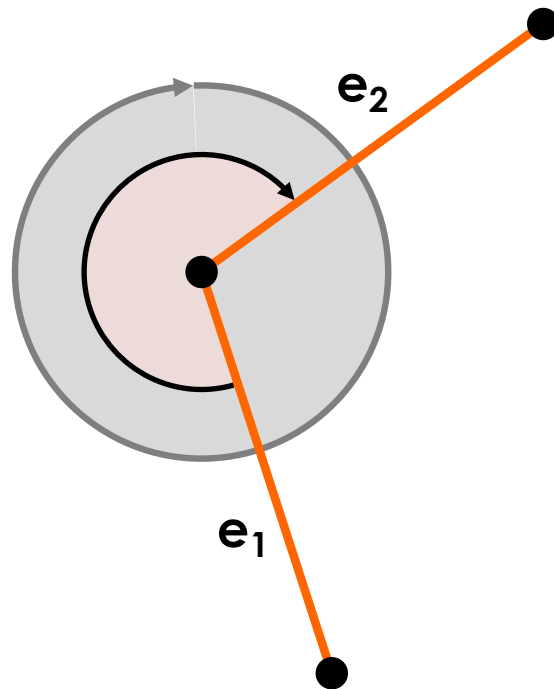
$A \rightarrow B \rightarrow C$

A 2 HOP VISIT and a POLAR Angle



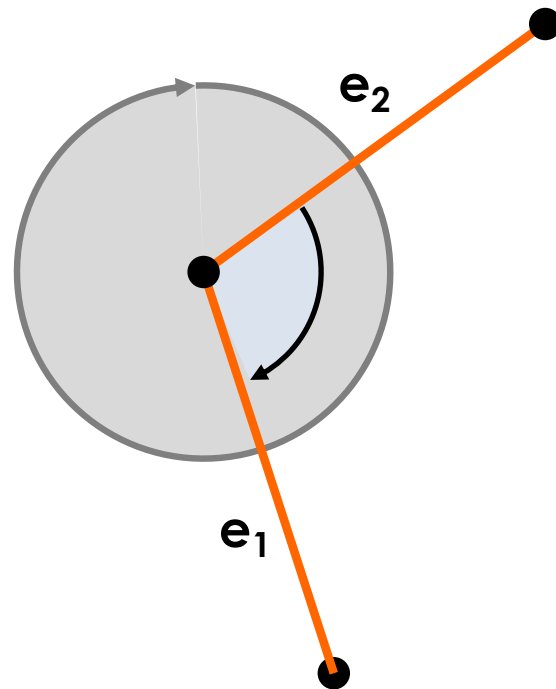
$C \rightarrow B \rightarrow A$

EDGES and a POLAR Angle



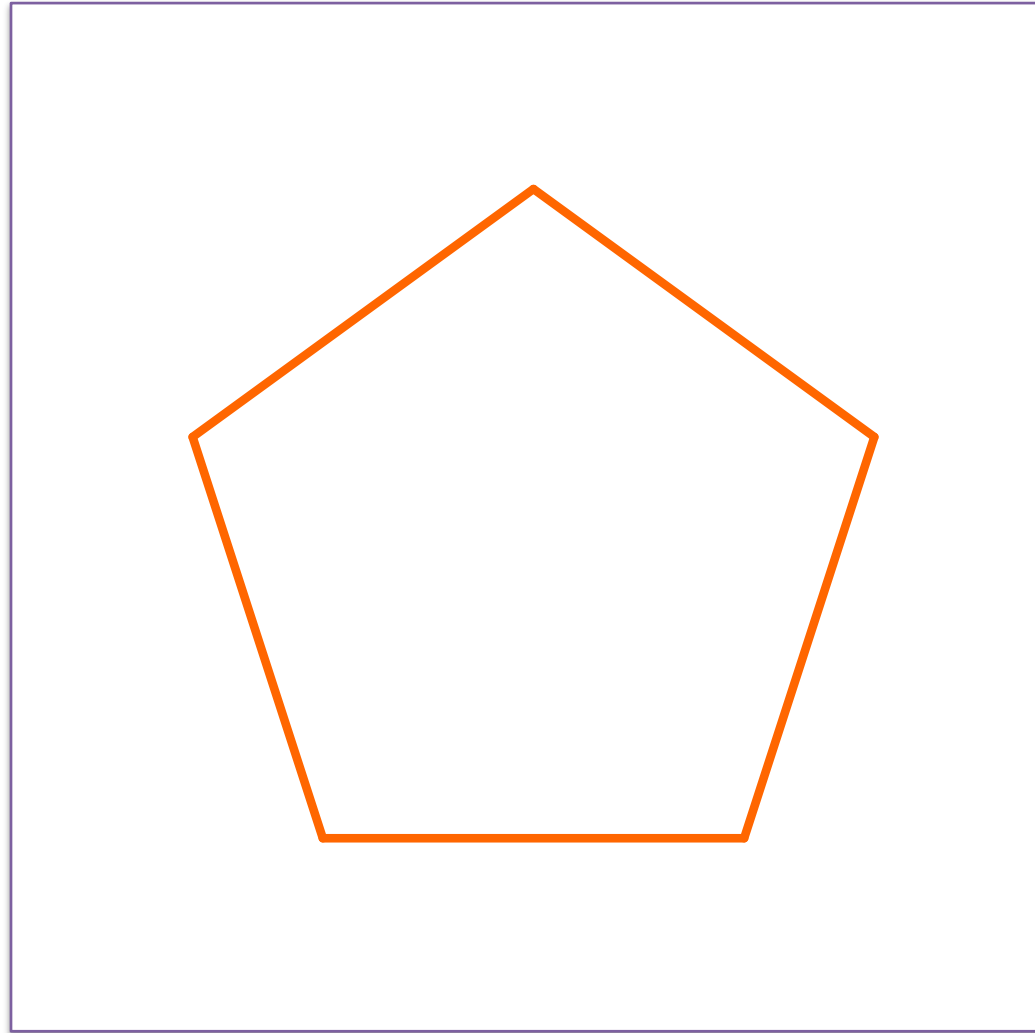
$e_1 \rightarrow e_2$

EDGES and a POLAR Angle

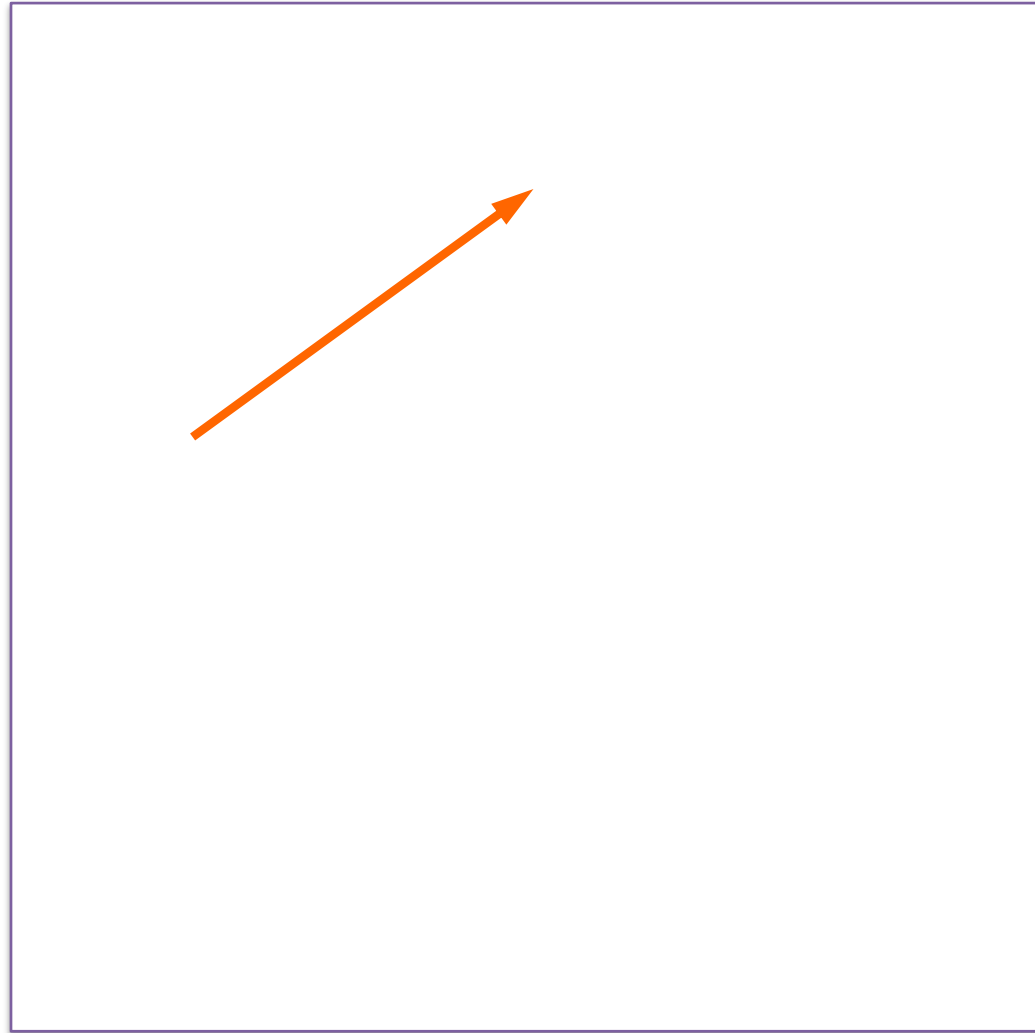


$e_2 \rightarrow e_1$

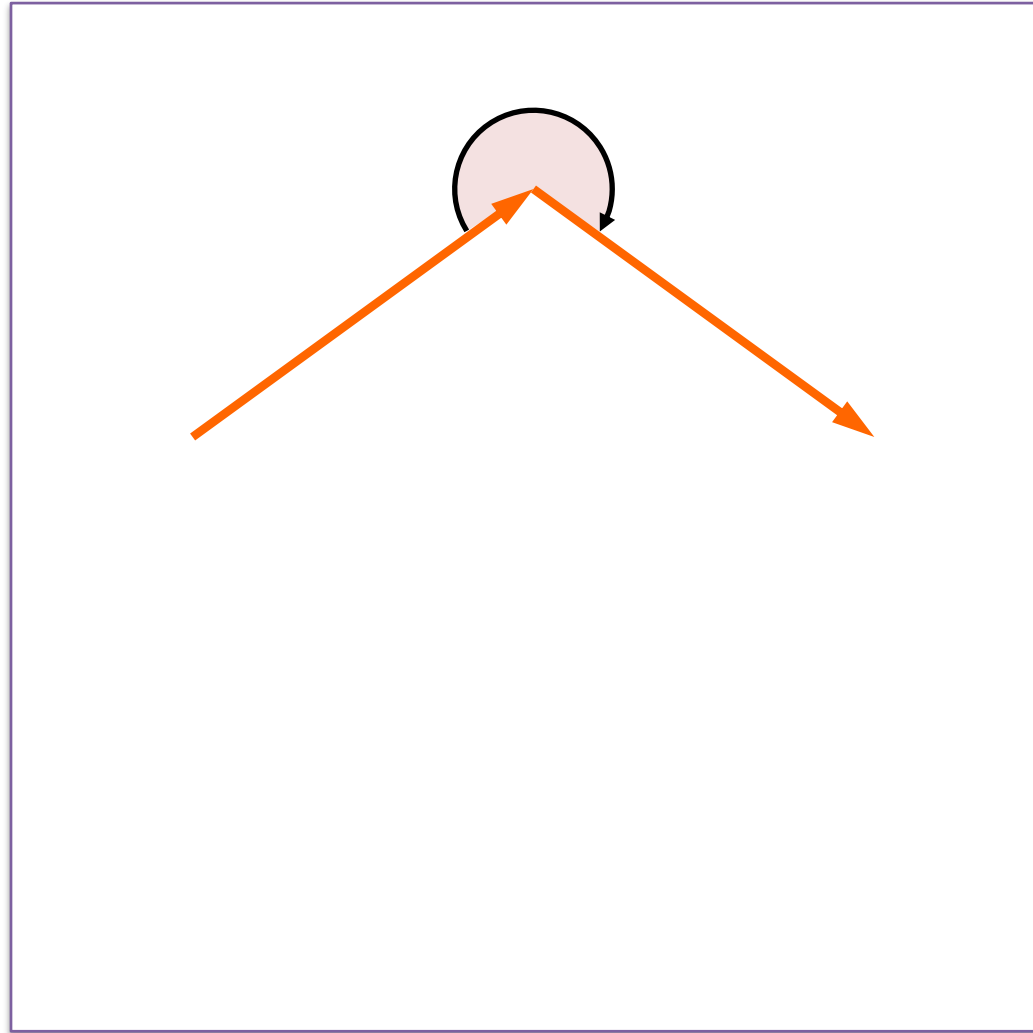
A Polygon



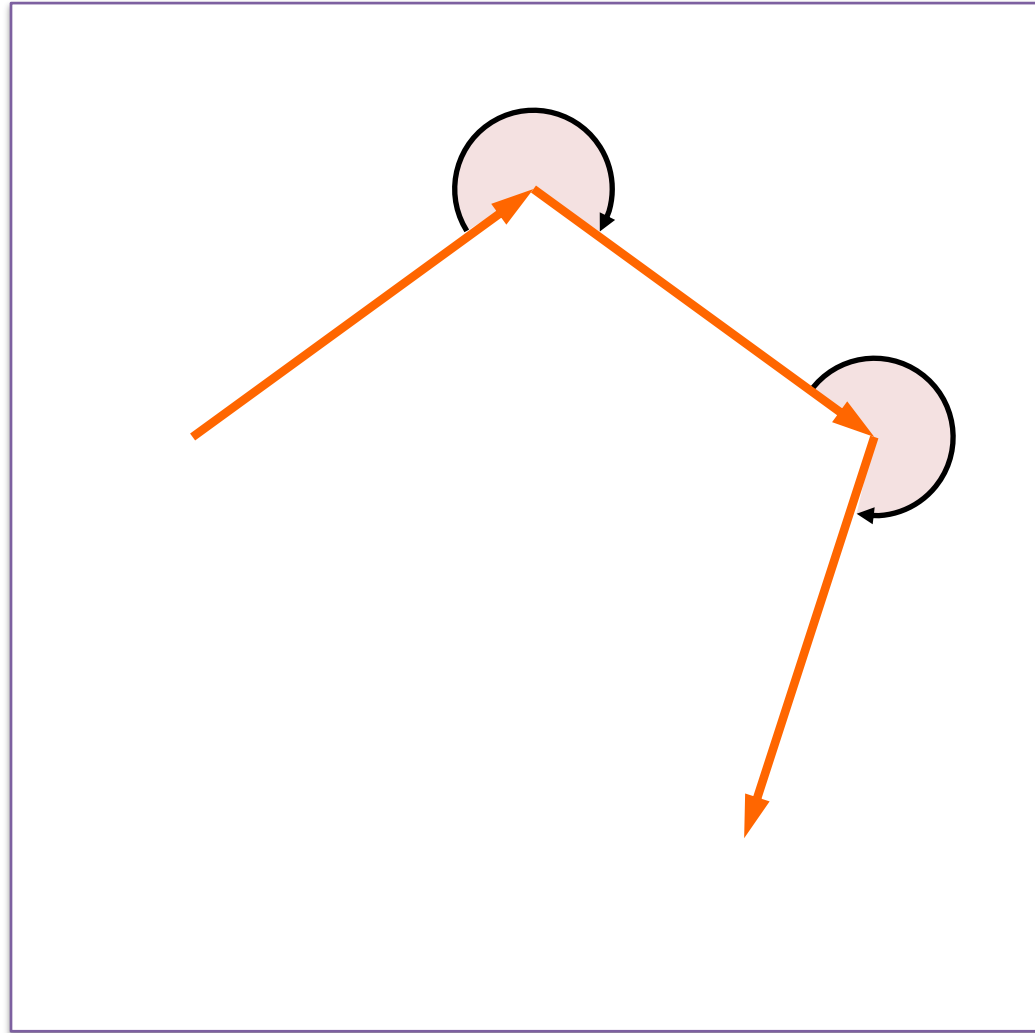
A Polygon



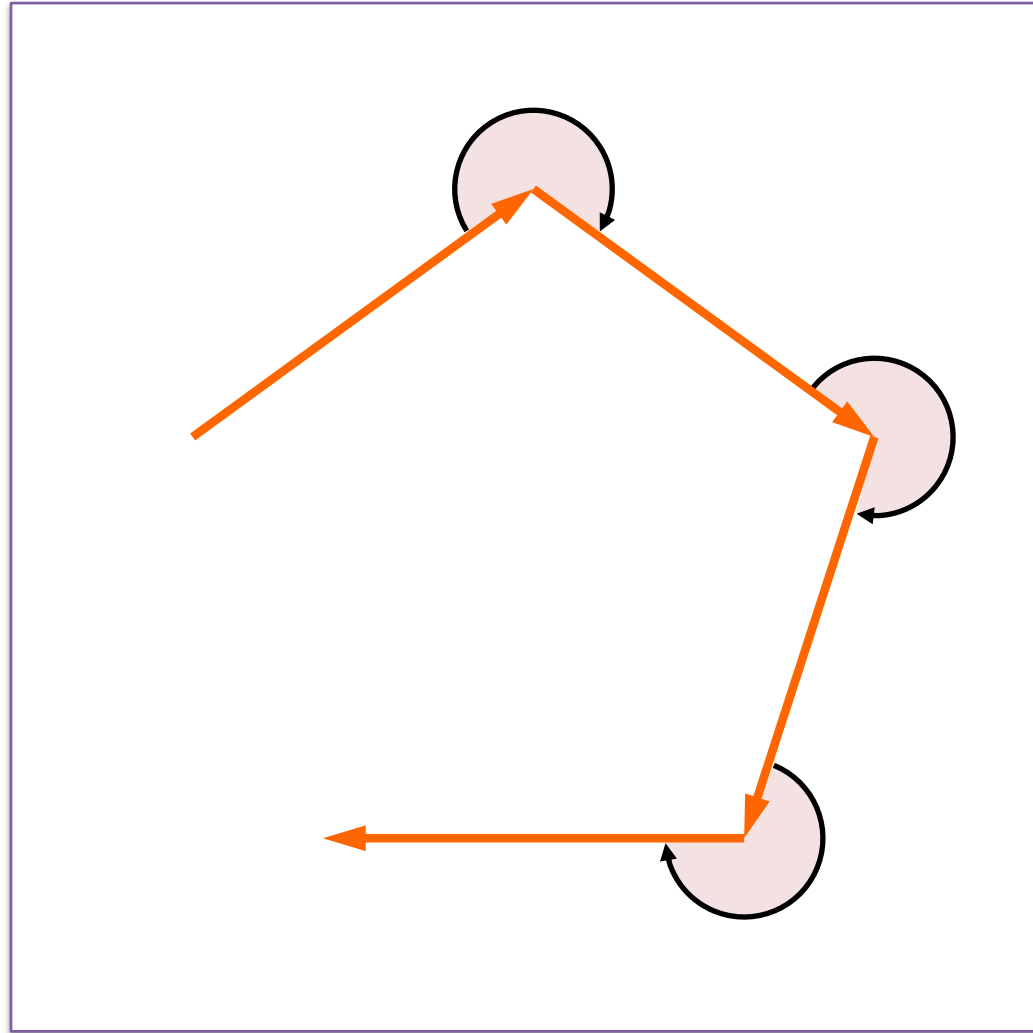
A Polygon



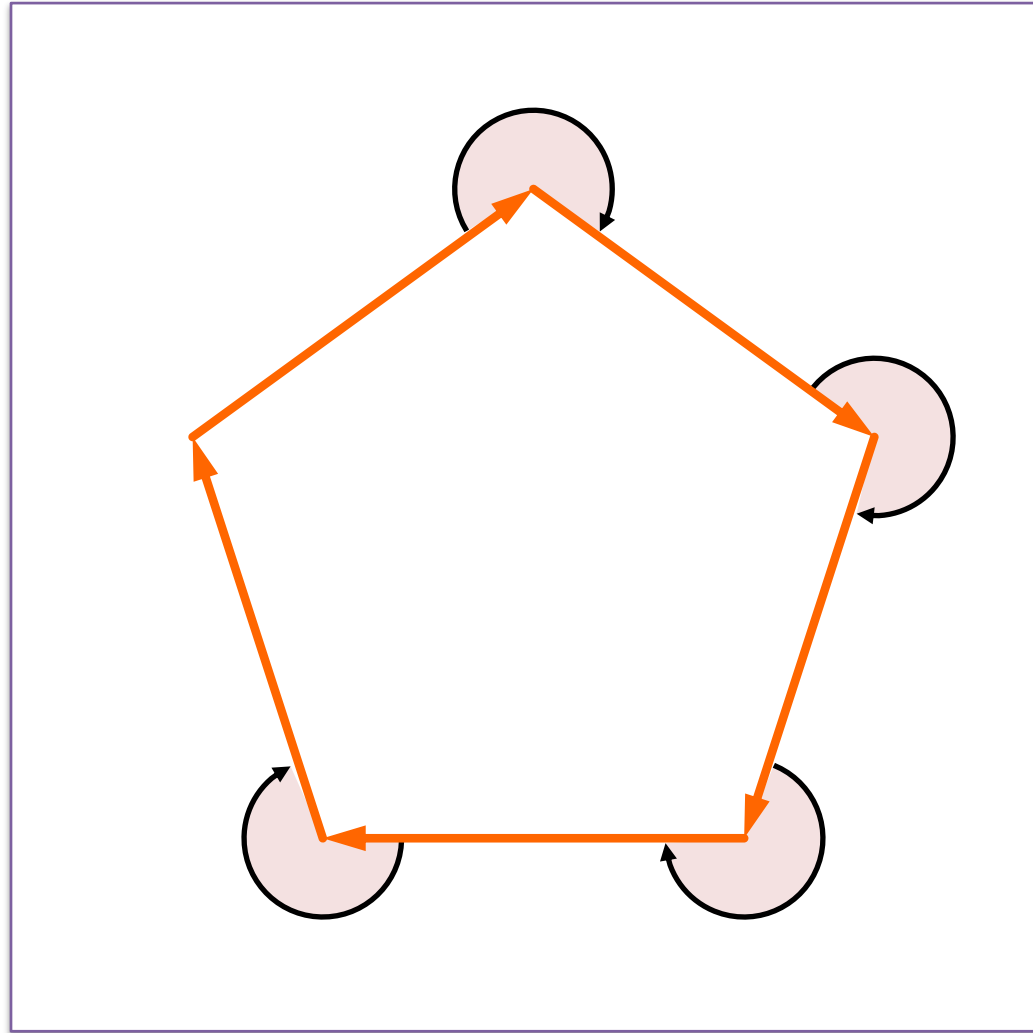
A Polygon



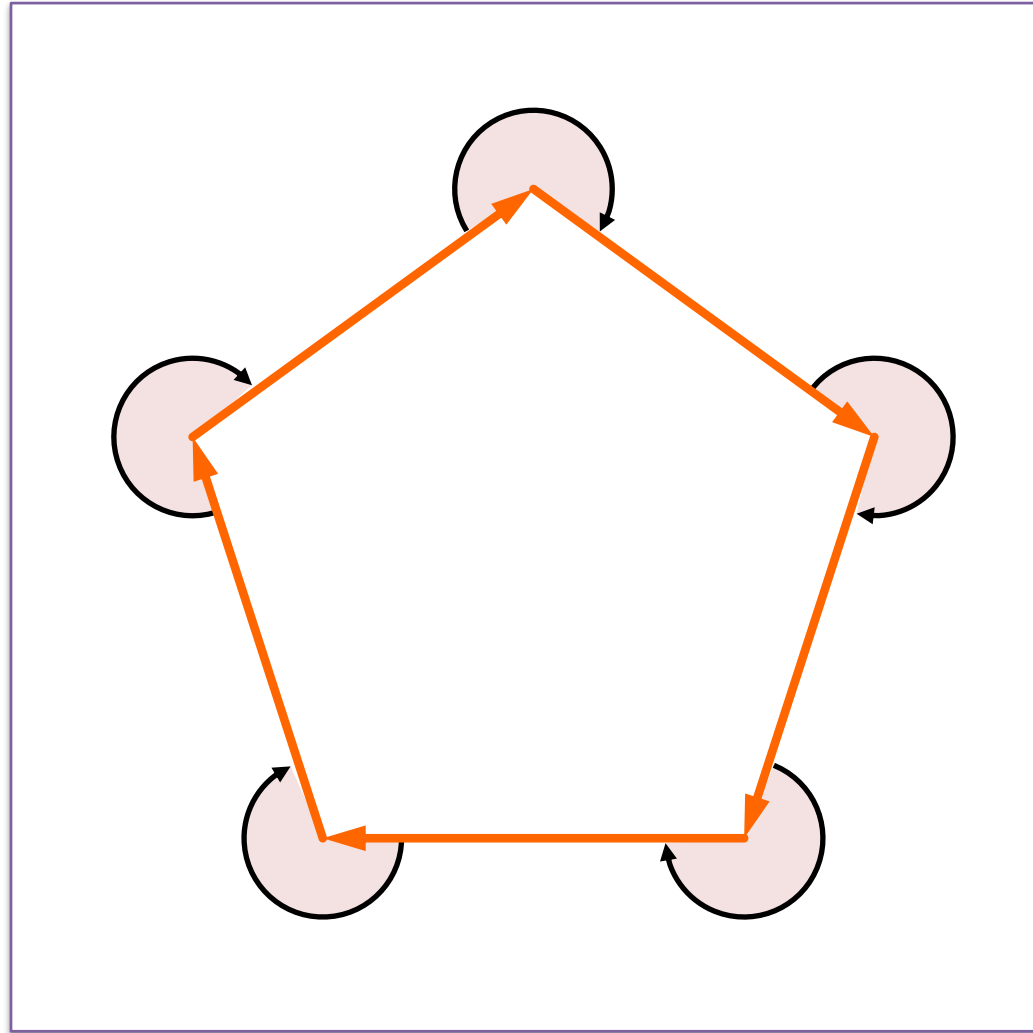
A Polygon



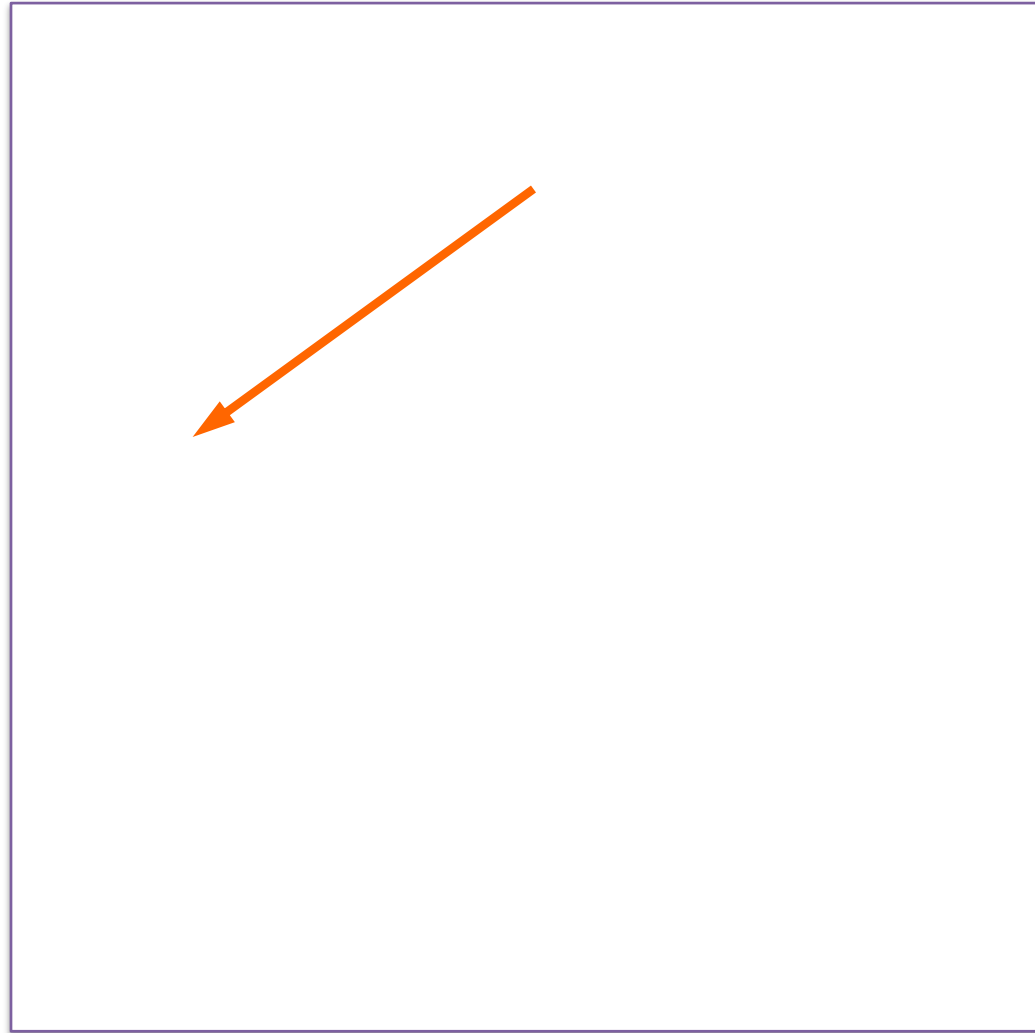
A Polygon



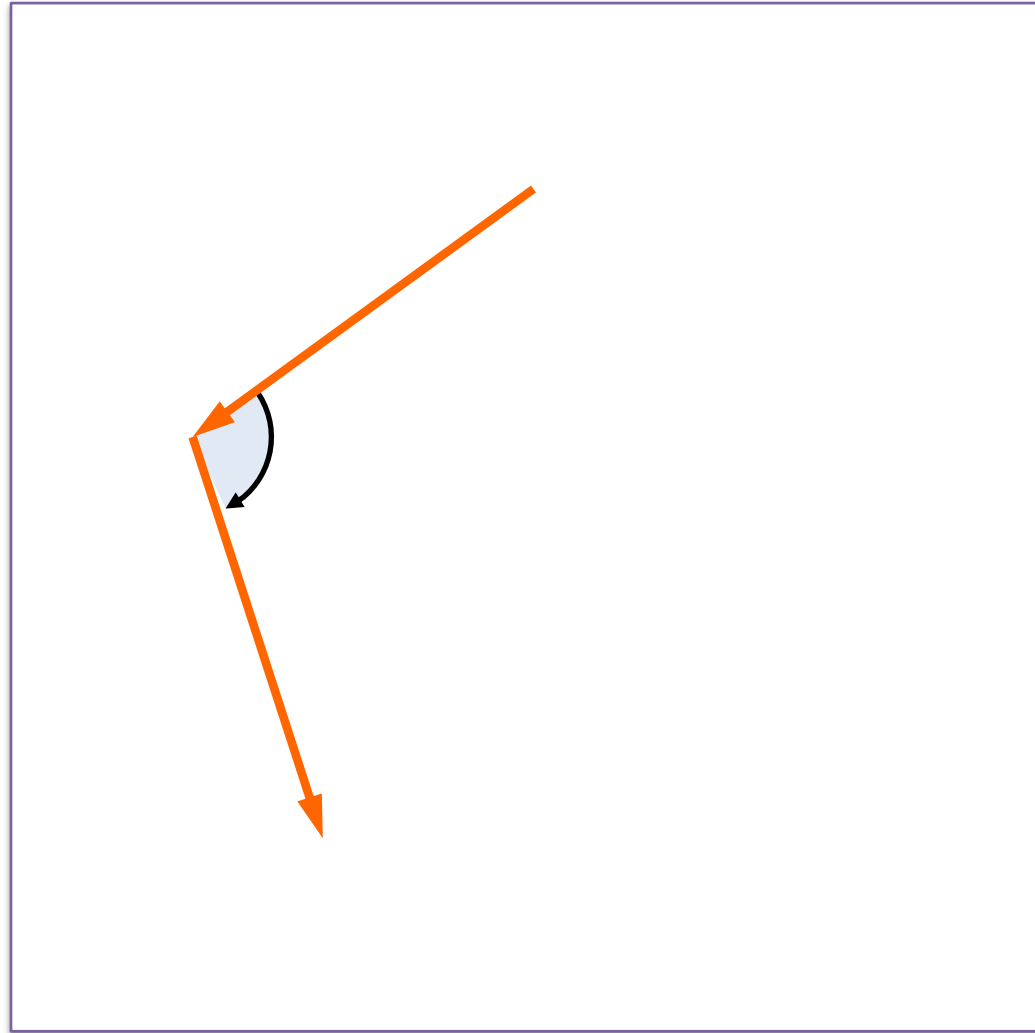
A Polygon



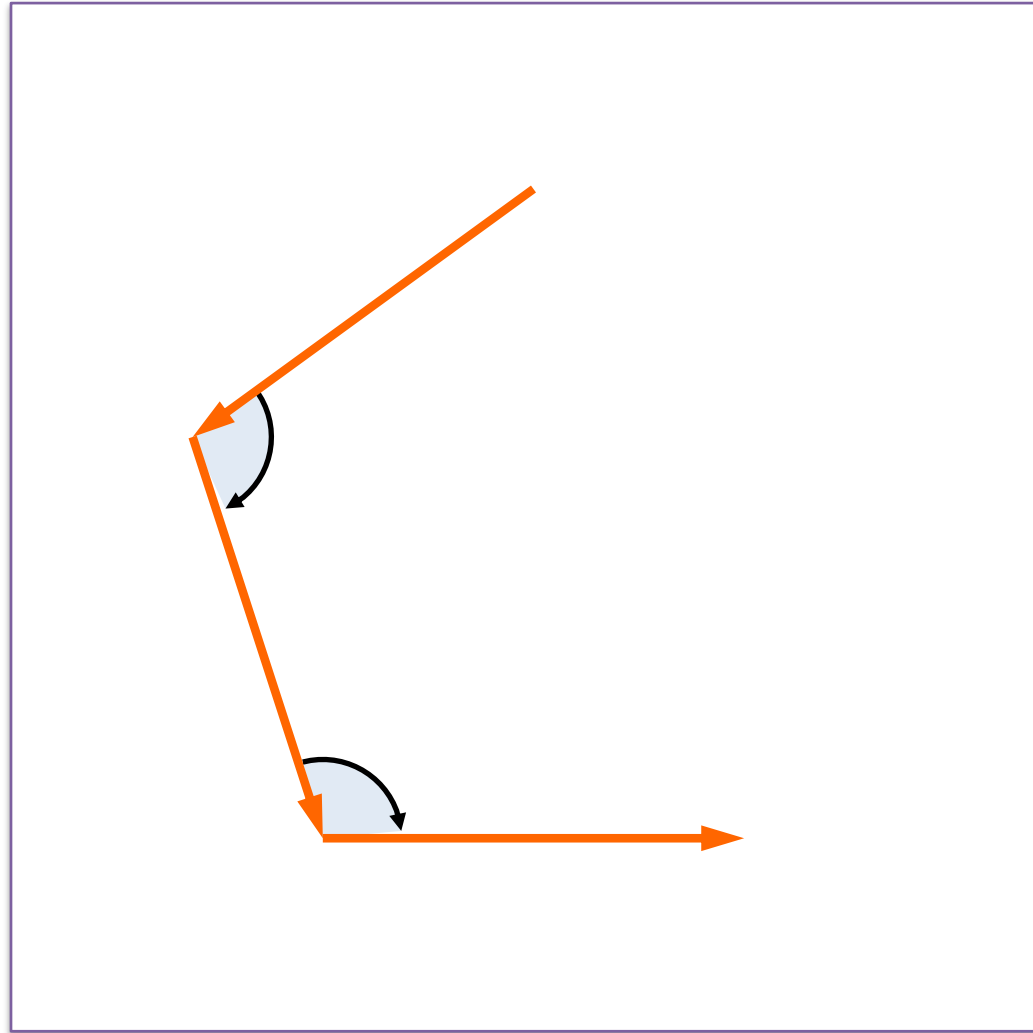
A Polygon



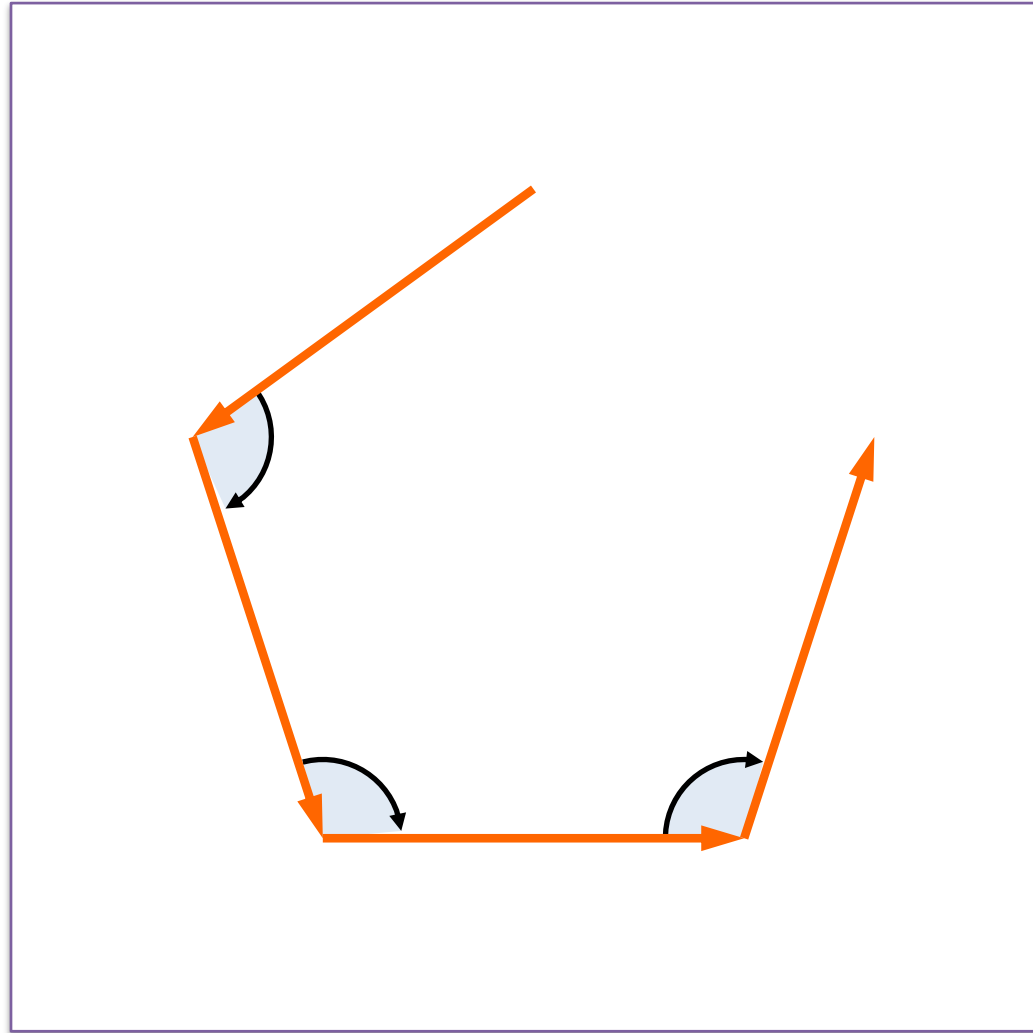
A Polygon



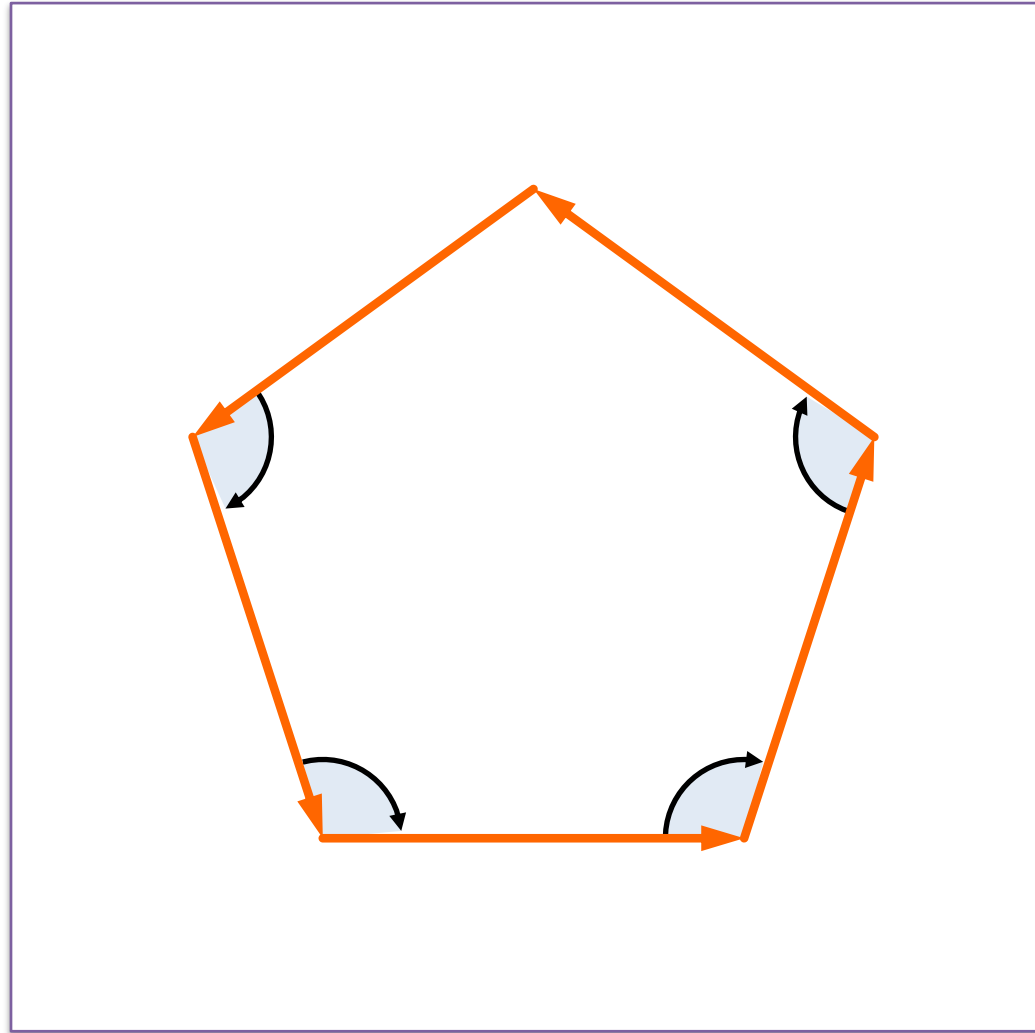
A Polygon



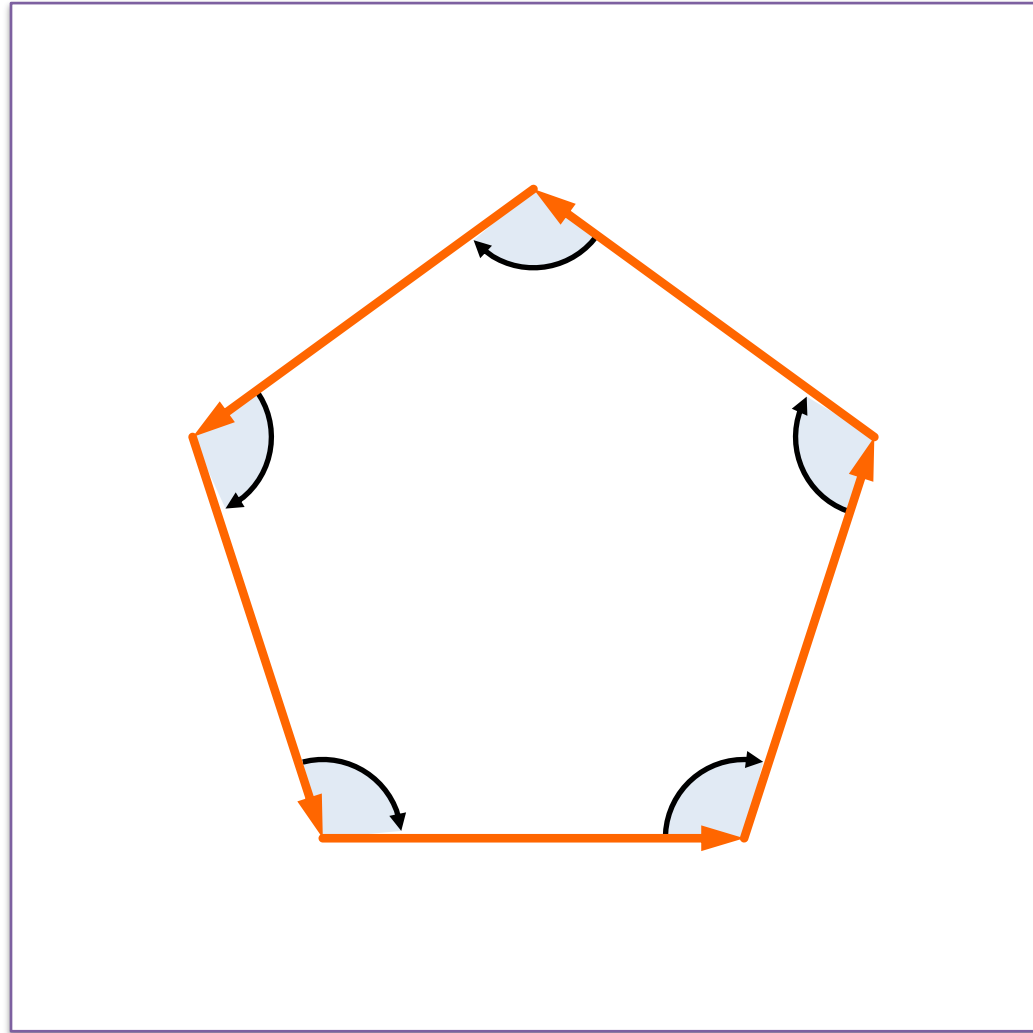
A Polygon



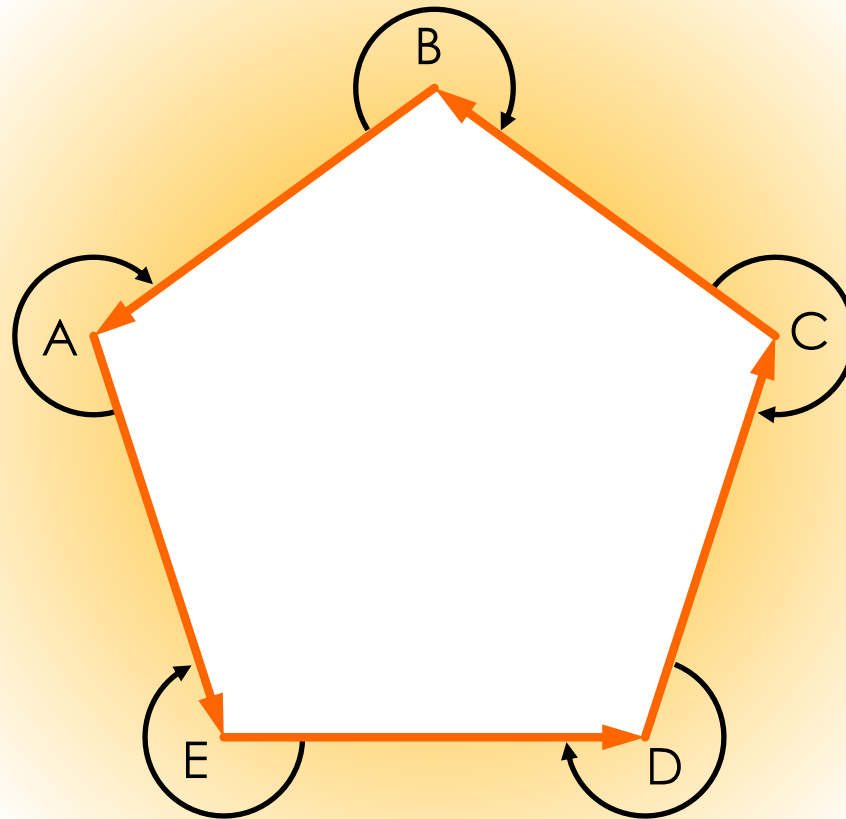
A Polygon



A Polygon

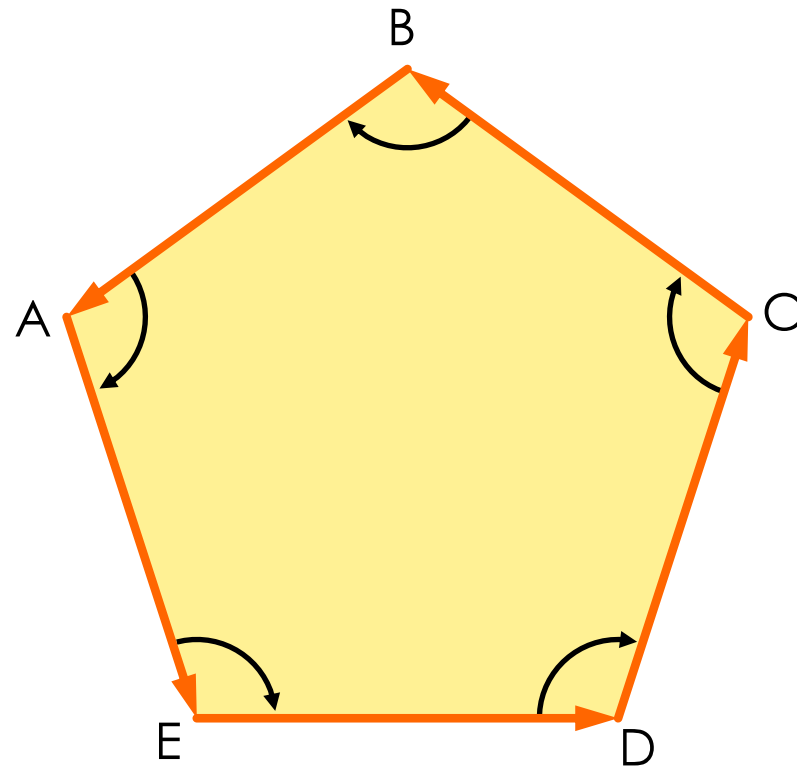


EXTERIOR Polygon



$A \rightarrow B \rightarrow C \rightarrow D \rightarrow E \rightarrow A$

INTERIOR Polygon



$A \rightarrow E \rightarrow D \rightarrow C \rightarrow B \rightarrow A$

Conclusion

A POLYGON is defined by its
VISITED ANGLES

Definition

A **POLY****GON**

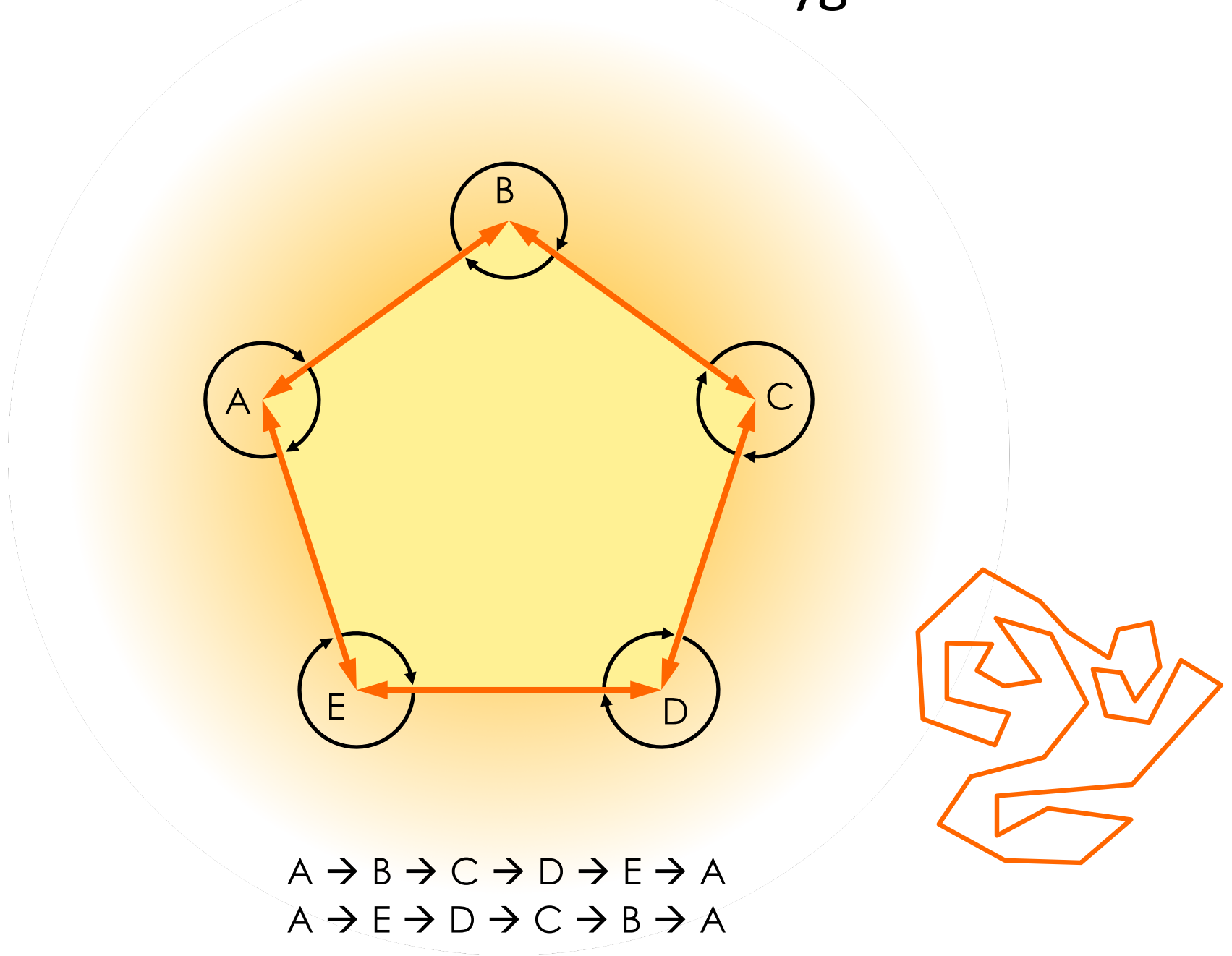
POLus = many

GONia = angles

Question

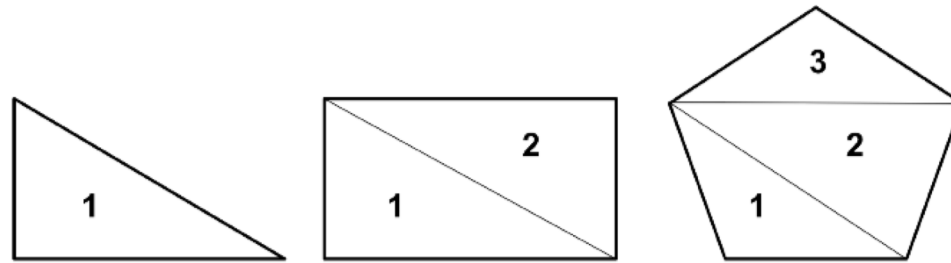
**How to determine that a
polygon is **INTERIOR** or
EXTERIOR?**

INTERIOR or EXTERIOR Polygon?



INTERIOR ANGLES

There are $(n-2)$ triangles in a polygon with n vertices



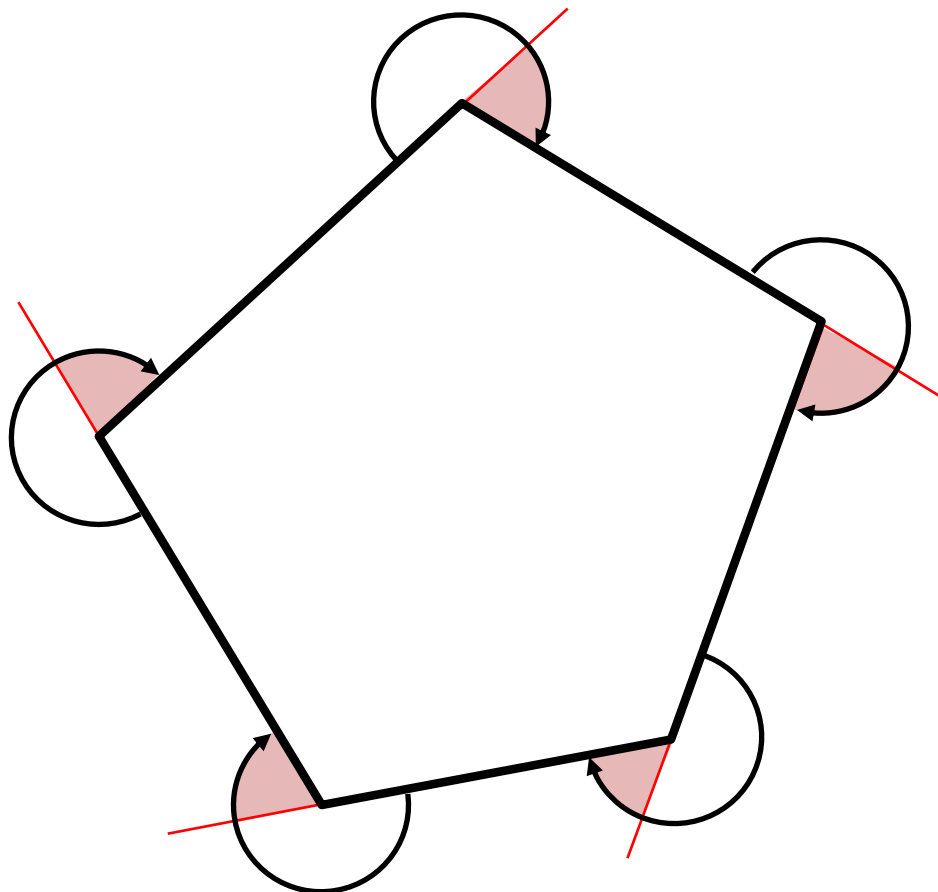
The sum of the interior angles of a triangle is equal to 180°

**The sum of the interior angles of a polygon is
equal to**

$$(n - 2) \times 180^\circ$$

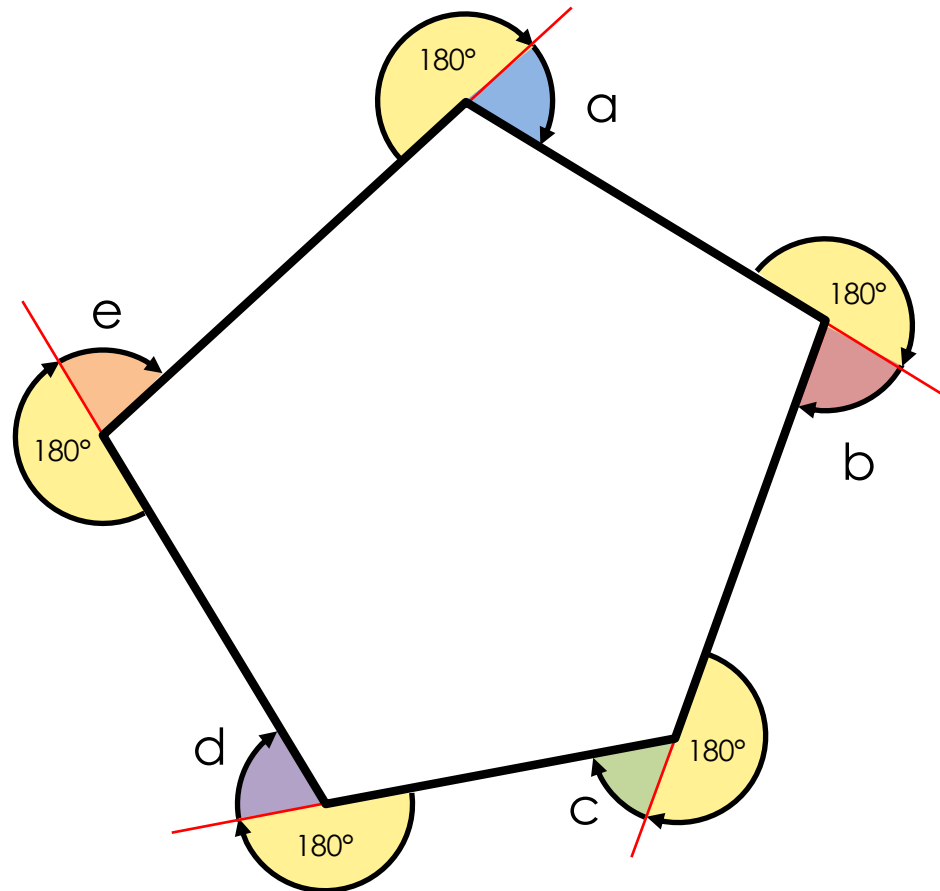
EXTERIOR ANGLES

The sum of the exterior (angles)



EXTERIOR ANGLES

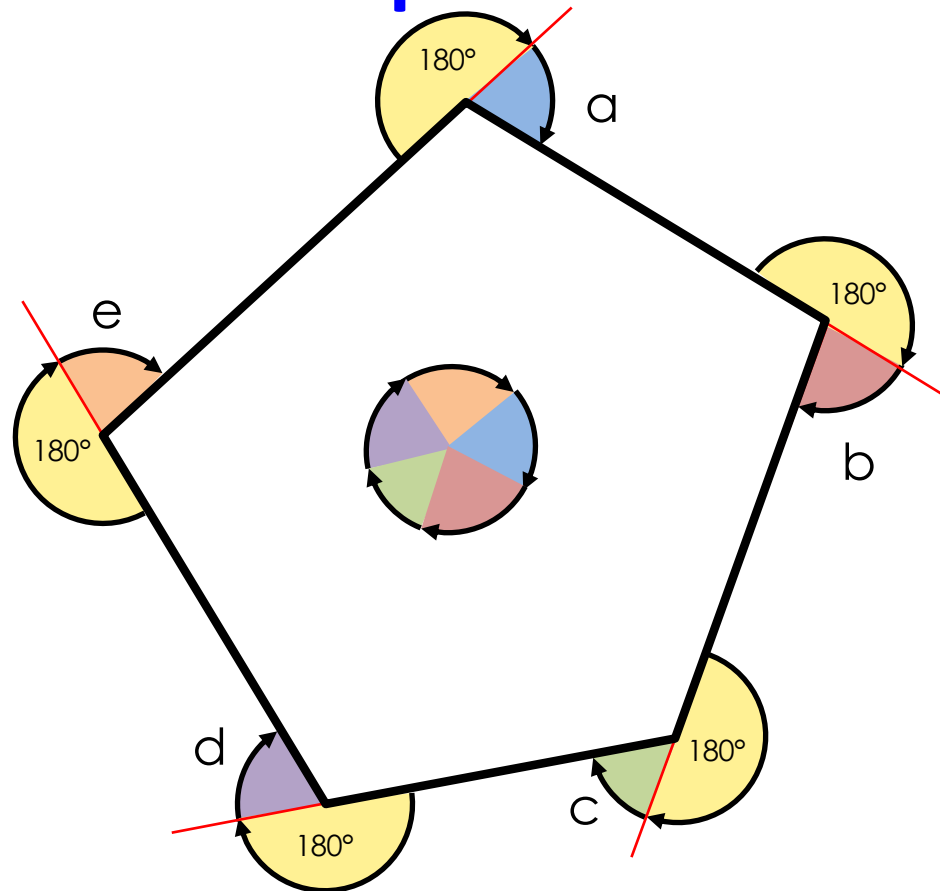
The sum of each (exterior angle – 180) of a polygon is equal to



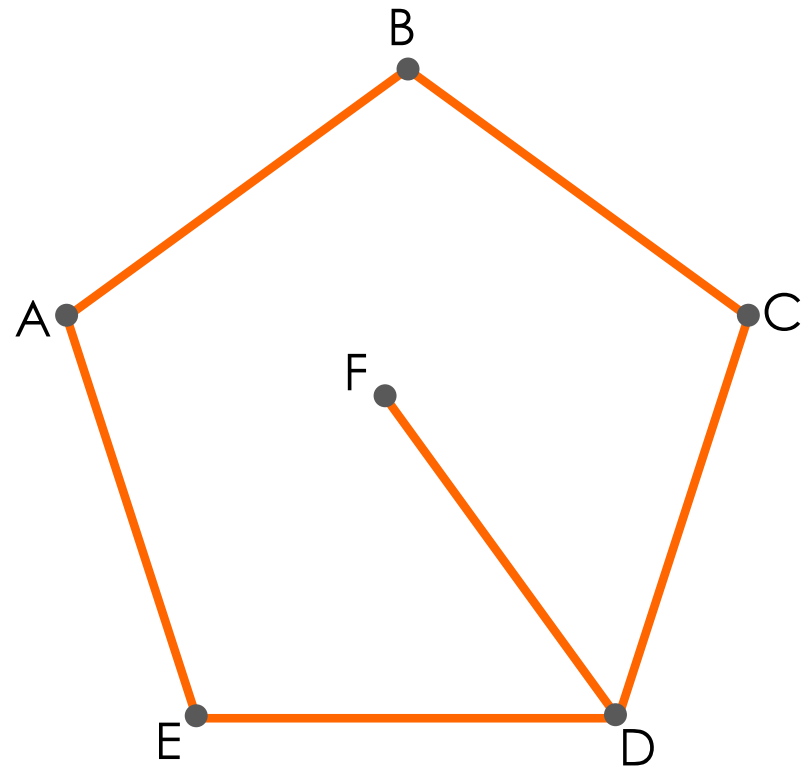
EXTERIOR ANGLES

The sum of each (exterior angle – 180) of a polygon

is equal to 360°

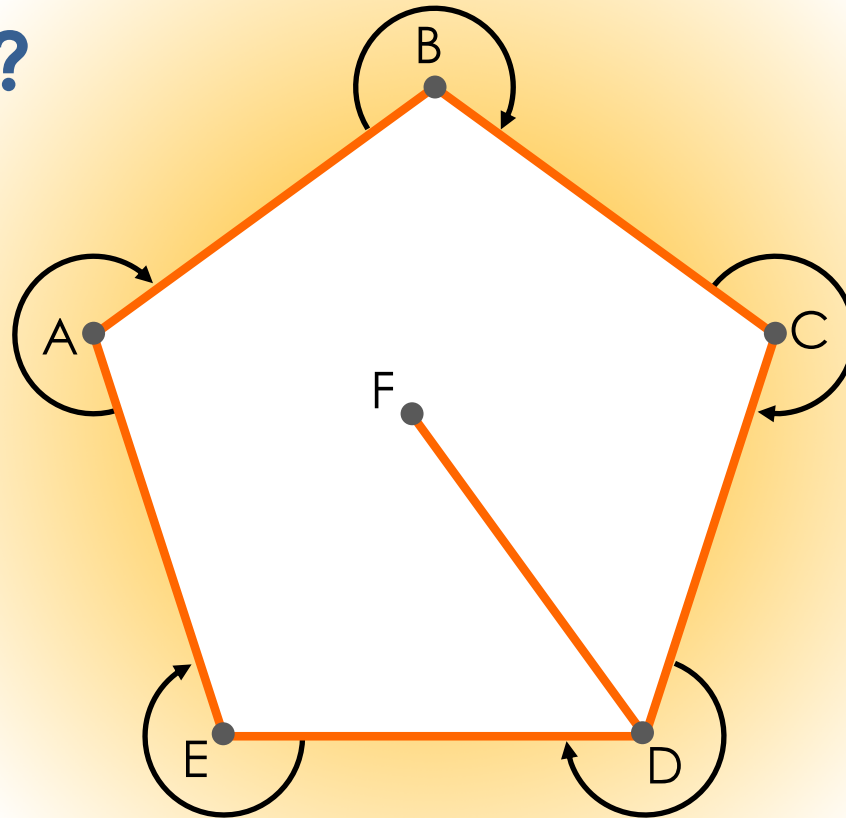


And what about this shape?



What about this shape?

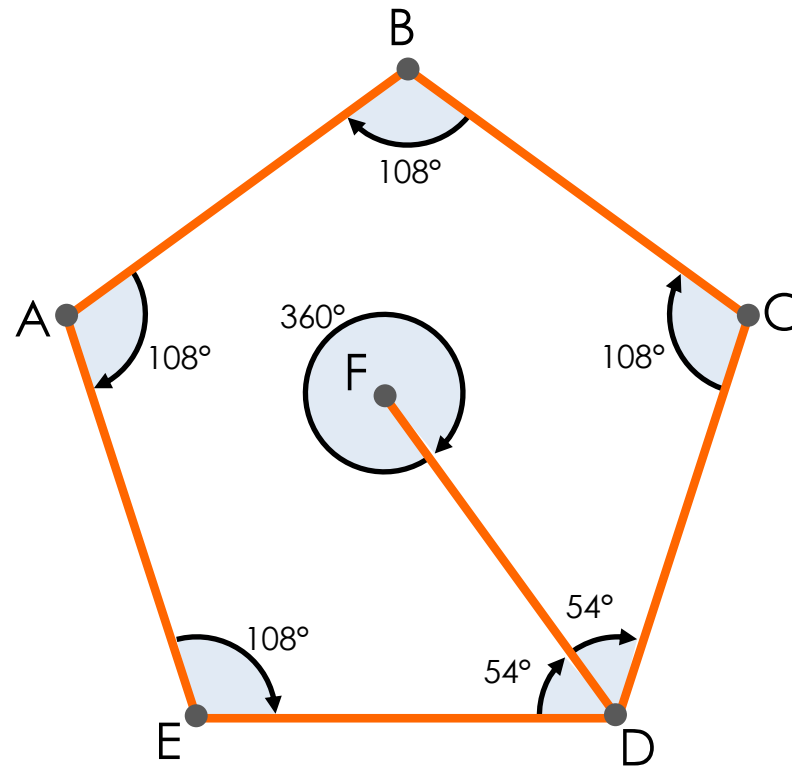
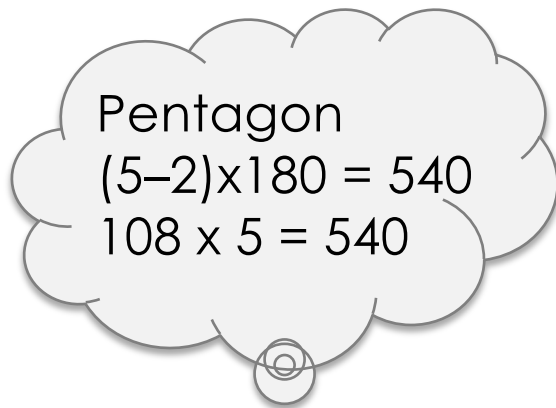
Is it EXTERIOR ?



EXTERIOR ✓

What about this shape?

Is it INTERIOR ?



$$(6-2) \times 180 = 720$$
$$108 \times 5 + 360 = 900$$

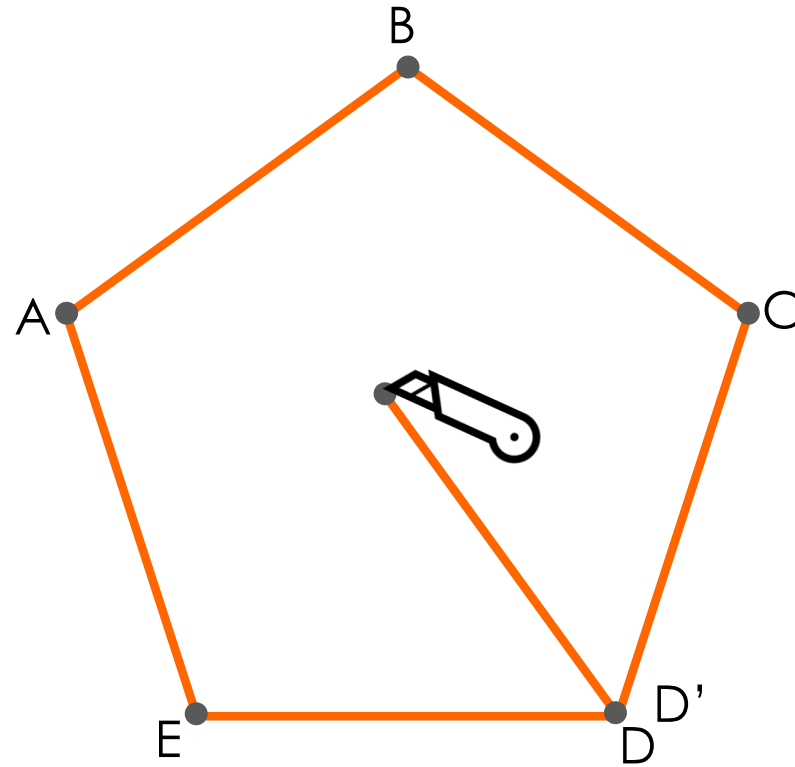
NON-INTERIOR X

BUT ...



What is a **TORN** Polygon

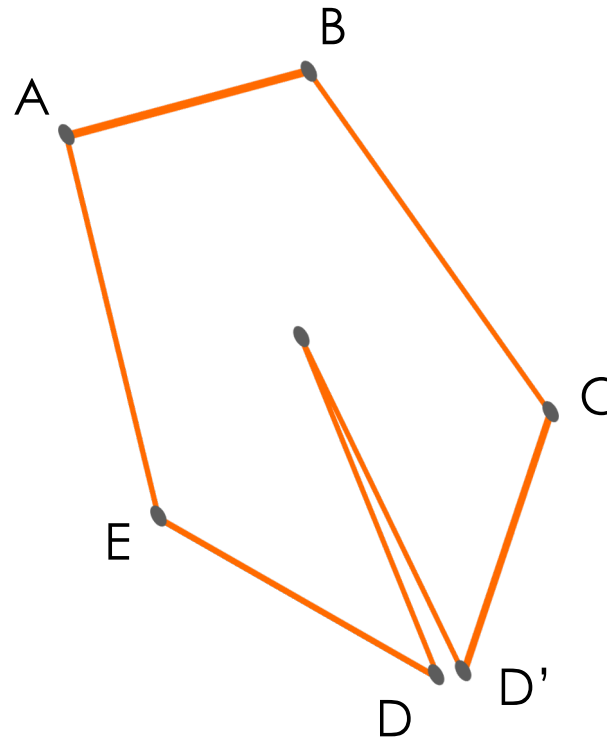
Is it **INTERIOR** ?



2 overlapped vertices

What is a **TORN** Polygon

Is it **INTERIOR** ?



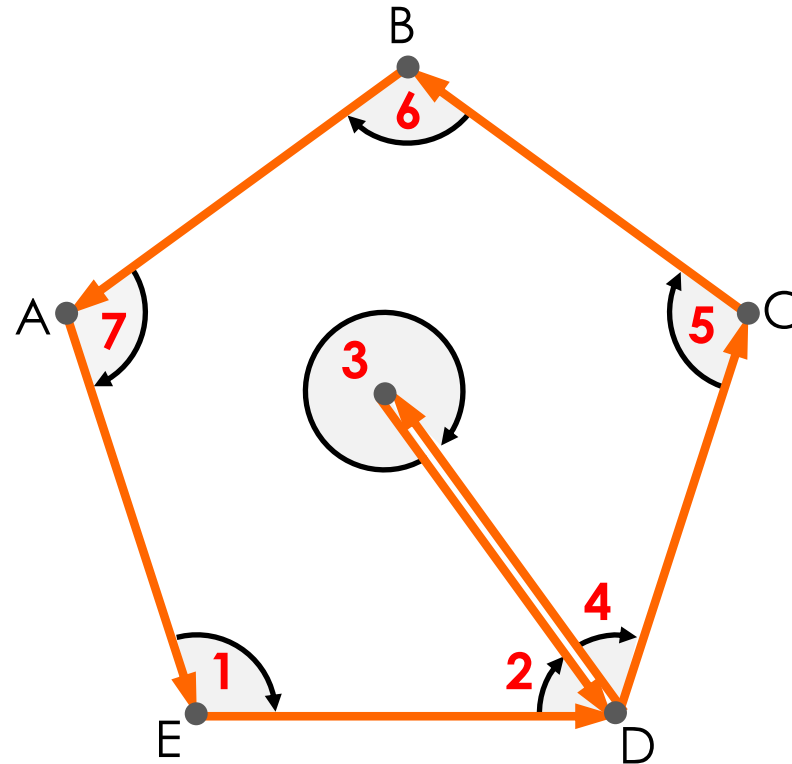
In this cas, this shape
looks like a polygon
with **7** vertices

$$(7-2) \times 180 = 900$$
$$108 \times 5 + 360 = 900$$

INTERIOR ✓

What about **visits** & **angles**?

Is it **INTERIOR** ?



$$n=7$$

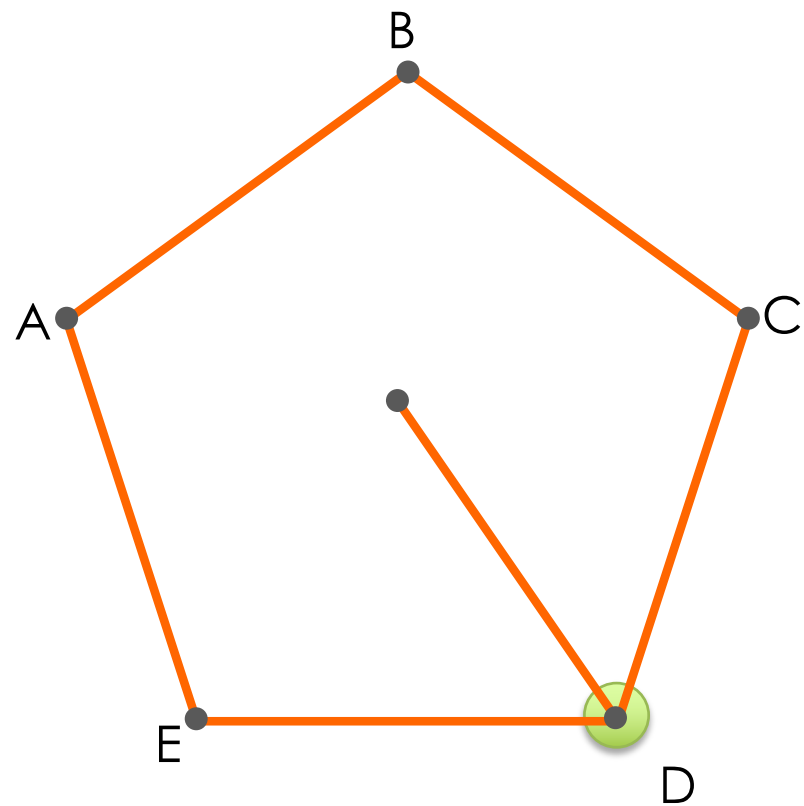
$$(7-2) \times 180 = 900$$

$$108 \times 5 + 360 = 900$$

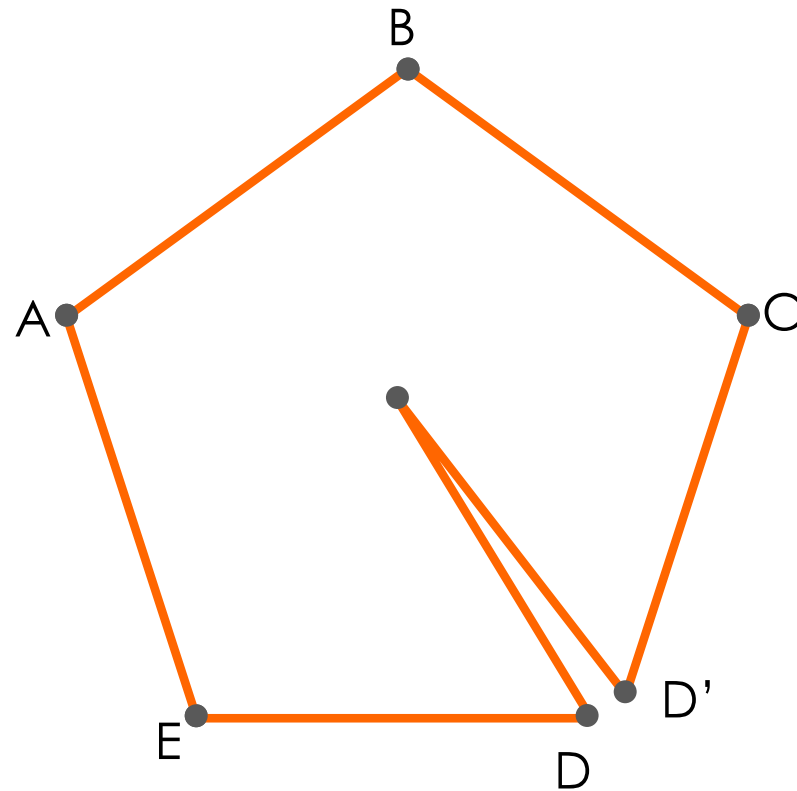
Let us consider **n** as the **number of formed angles** instead just the number of vertices

INTERIOR ✓

A PSEUDO-POLYGON



A PSEUDO-POLYGON

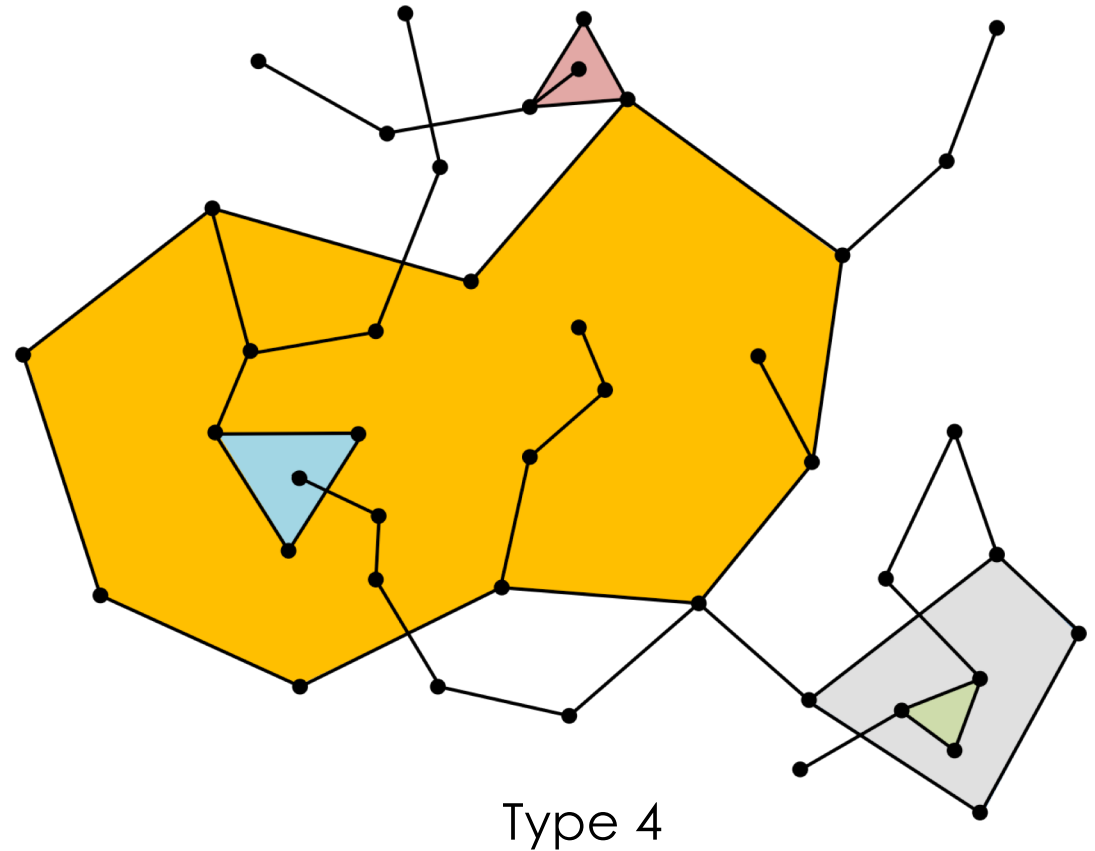
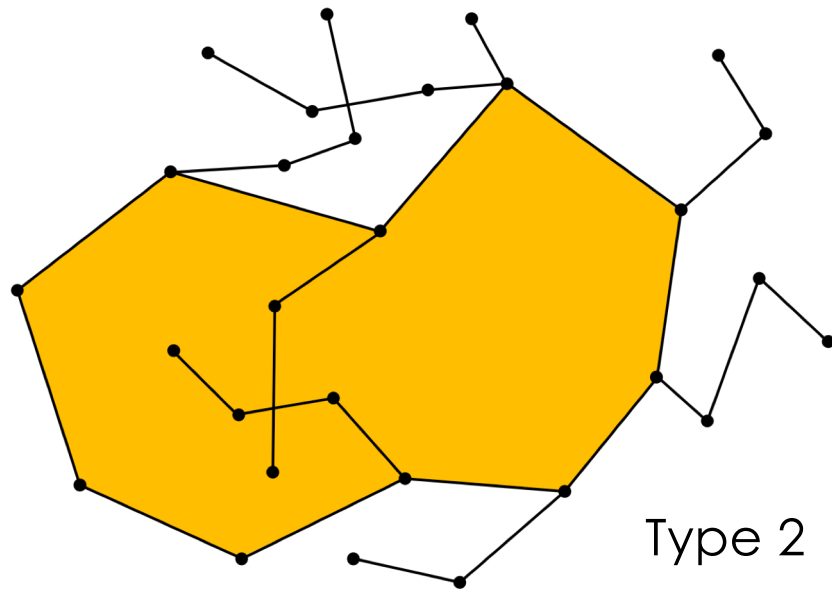
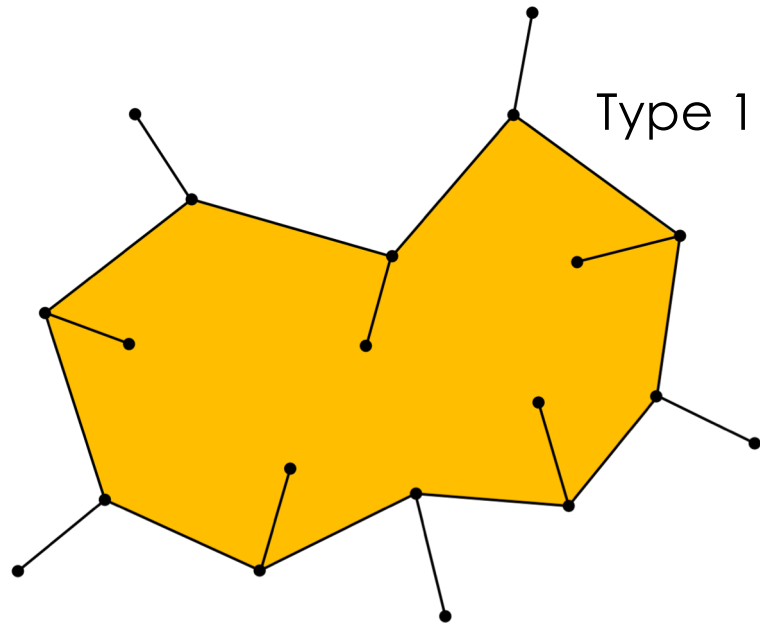


The equivalent Torn Polygon

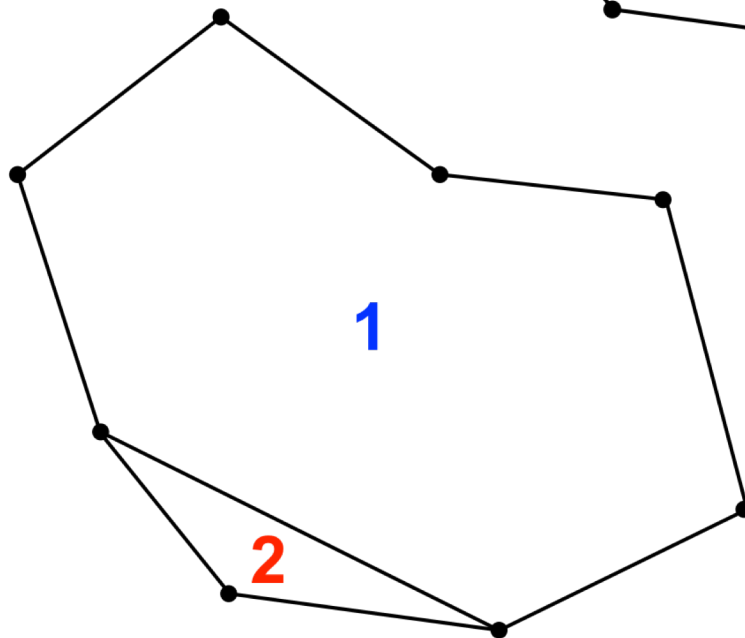
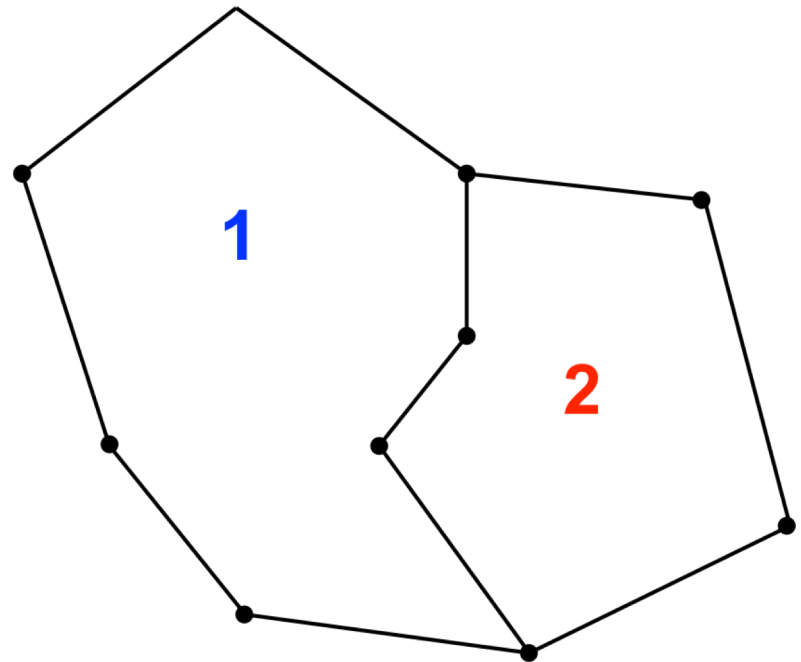
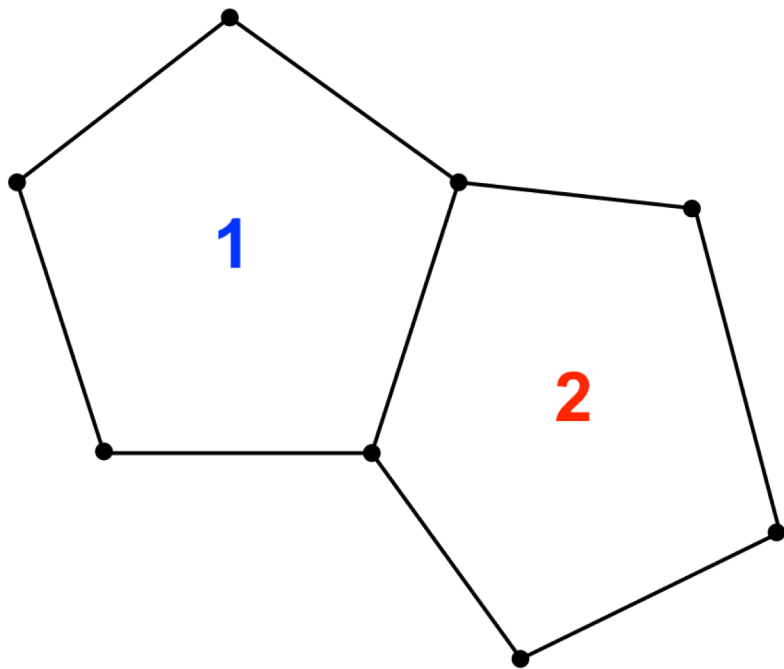
A **PSEUDO-POLYGON**

A pseudo-polygon P is a set of simple polygons in which each couple of polygons is connected by at most one chain in such a way that none of the polygons is subdivided into two adjacent polygons.

Some Pseudo Polygons



A NON-PSEUDO-POLYGON



2



CUPCARBON FOR IOT SIMULATION

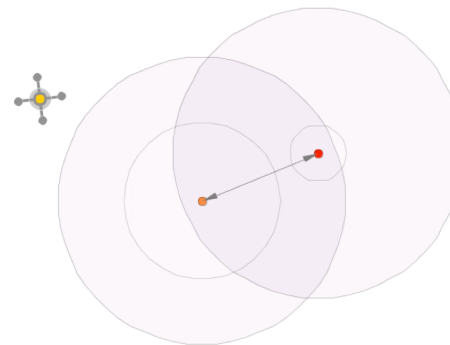
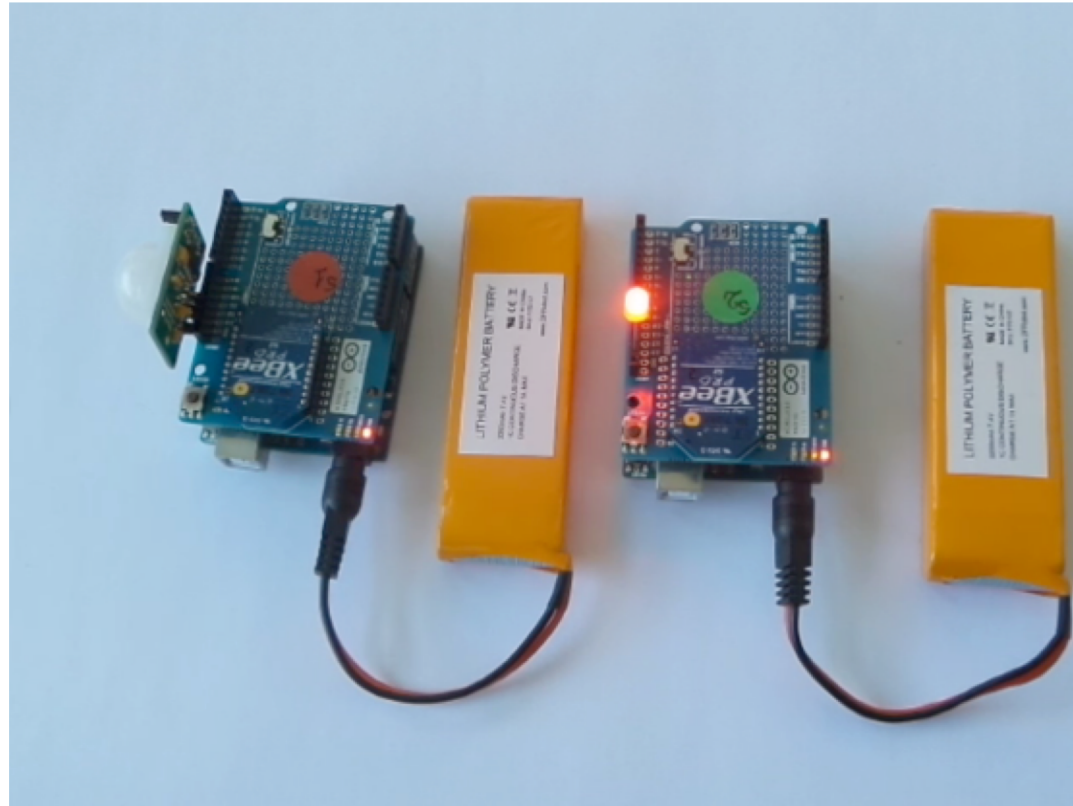
SenScript: Example

S1

```
loop
dreadsensor x
if ($x==1)
    send A
else
    send B
endif
stop
```

S2

```
loop
wait
read x
if ($x==A)
    led 13 1
else
    led 13 0
endif
```



NEW: CupCarbon/SenScript

CupCarbon 3.0.0 (U-One) [dlpcn2.cup]

Simulation Time: 51,7251 s

Left Sidebar:

- Devices & Objects
- Device Parameters
- Radio Parameters
- Marker Parameters
- Simulation Parameters
 - SenScript
 - Simulation time: 100.0 s
 - Speed: 10 ms
 - Arrow speed: 50 ms
 - Run Simulation
 - Stop Simulation
 - Apply
 - ☒ Symmetrical links ☐ Visibility
 - ☒ Results ☐ Log
 - ☐ Mobility / Events ☐ Clock Drift
 - ☐ ACK ☒ Show
 - Probability: 1.0

Map View: A street map showing a network of nodes (yellow circles) and links (red lines). Nodes are labeled with IDs like dlpcn2_S64 [0], dlpcn2_S65 [0], etc. The map includes street names like RUE SAINT-MARC, RUE KERVIN, RUE LOUIS PIDOUX, RUE LAENNEC, RUE LEVOT, RUE FRANÇOIS RIVIÈRE, RUE PIERRE SEMAR, RUE JEANNE MAISTRE, RUE JOSE MARIA DE HEREDIA, RUE DU FORESTOU HUELLA, RUE DU CAPITAINE GUYNEMER, RUE TOUROT, RUE LAENNEC, RUE LEVOT, RUE FRANÇOIS RIVIÈRE, RUE PIERRE SEMAR, RUE JEANNE MAISTRE, RUE JOSE MARIA DE HEREDIA, RUE DU FORESTOU HUELLA, RUE DU CAPITAINE GUYNEMER, RUE TOUROT, RUE LAENNEC, RUE LEVOT, RUE FRANÇOIS RIVIÈRE, RUE PIERRE SEMAR, RUE JEANNE MAISTRE, RUE JOSE MARIA DE HEREDIA, RUE DU FORESTOU HUELLA, RUE DU CAPITAINE GUYNEMER.

SenScript Editor (dlpcn2_sp.csc):

```
1 tab t 10 1
2 set idx -1
3 atget id cid
4 getpos2 ex cy
5 set once 1
6 atnd n
7 set i 0
8 set first 39
9 set mid_ref 0
10 loop
11   if (($cid==$first) && ($once==1))
12     set color 2
13     set xmin $cx
14     set ymin $cy
15     minus px $cx 1
16     set py $cy
17     set once 0
18     set type SN
19     set id $first
20     set idx -1
21     data p 0 SN $px $py $xmin
22   else
23     wait
24     read p
25     rdata $p id type
26   end
27
28   if ($type==AC)
29     data p $cid CS $cx $cy
```

Bottom Status Bar:

Simulation time: 64 N = 34 RT = 13,6140 s

Presentation of CupCarbon

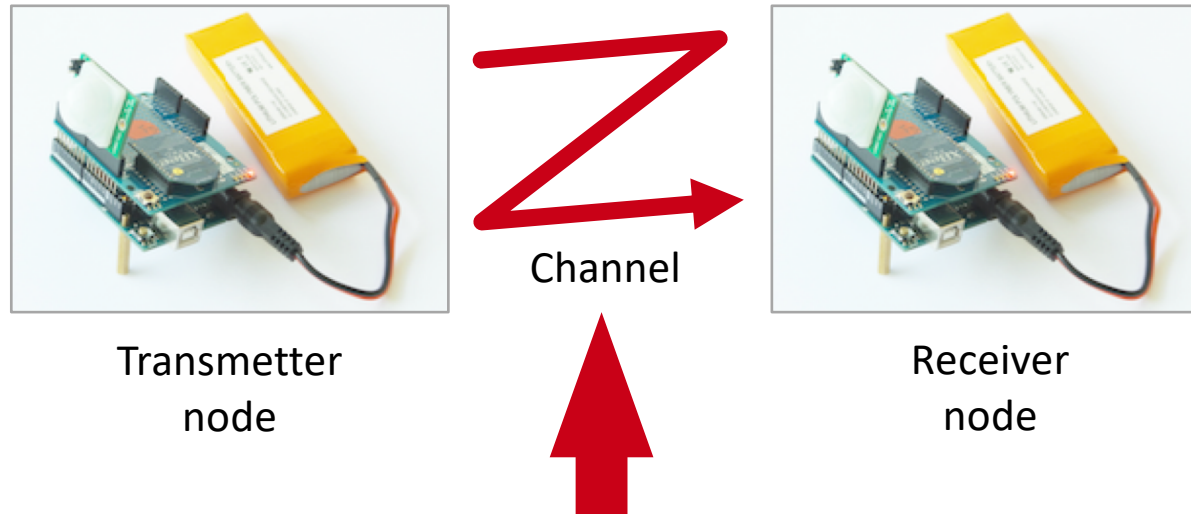
CupCarbon is a WSN simulator dedicated to Smart Cities and Internet of Things

- Its objective is to **design, visualize, debug, simulate and validate distributed algorithms,**
- It can be used for **educational** and **scientific** projects,
- It offers an **ergonomic** and **easy to use interface** using the **OpenStreetMap** (OSM) framework to deploy sensors directly on the map,

Presentation of CupCarbon

- It includes a script called **SenScript** which allows to **program** and to **configure** each sensor node individually,
- It is possible to **generate codes** for **hardware platforms** such as Arduino/XBee,
- CupCarbon offers the possibility to **simulate many algorithms** and scenarios in **several steps**,
- The **energy consumption** can be **modelled**, calculated and displayed as a function of the simulated time,

Modelling the radio propagation channel

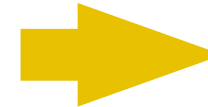


Existing WSN simulators:

- NS-2/3
- OMNET
- TOSSIM
- OPNET
- ...

➡
Classical
models used in
WSN

- Free space model
- Two rays ground model
- Log normal model
- ...

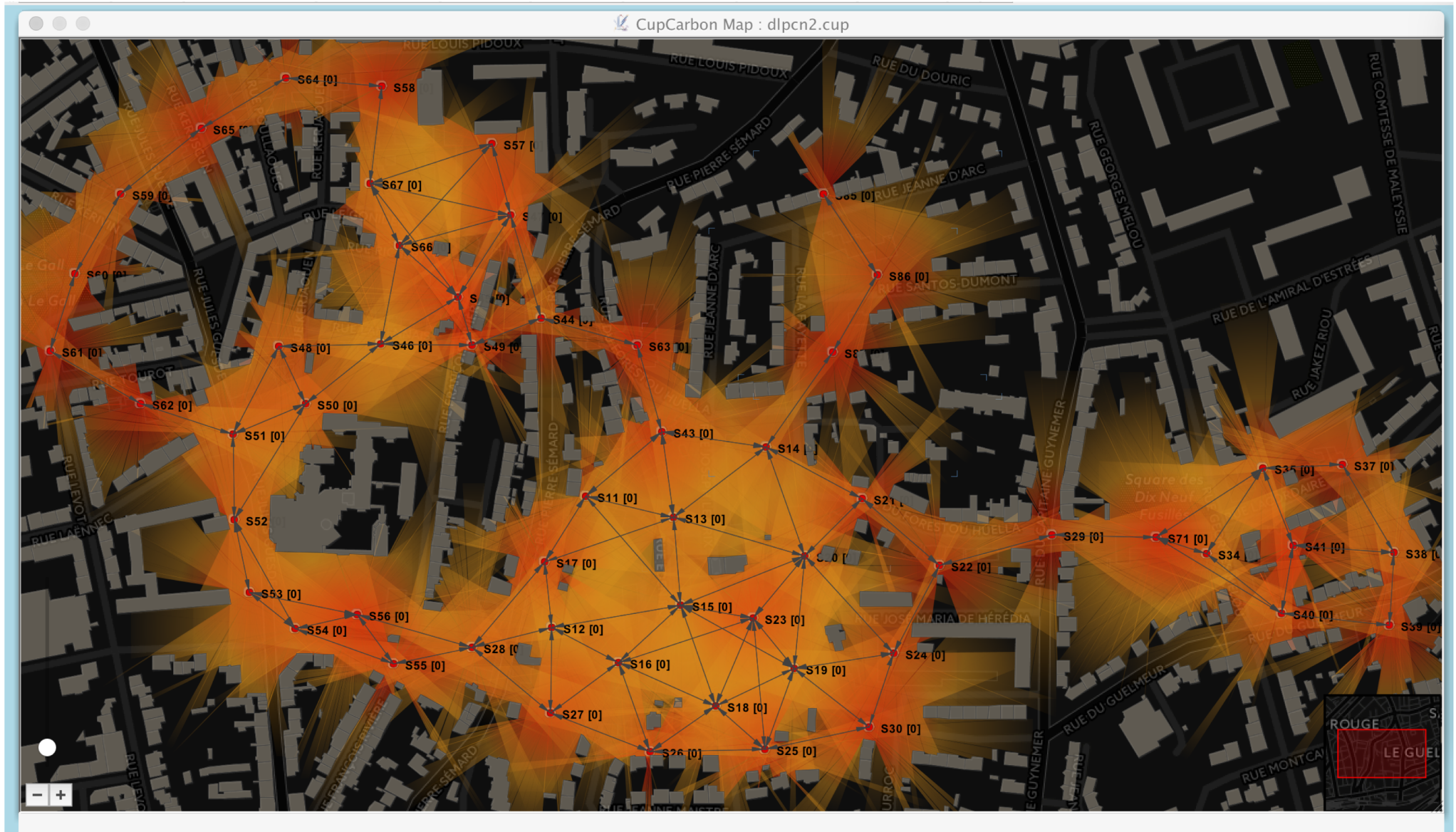


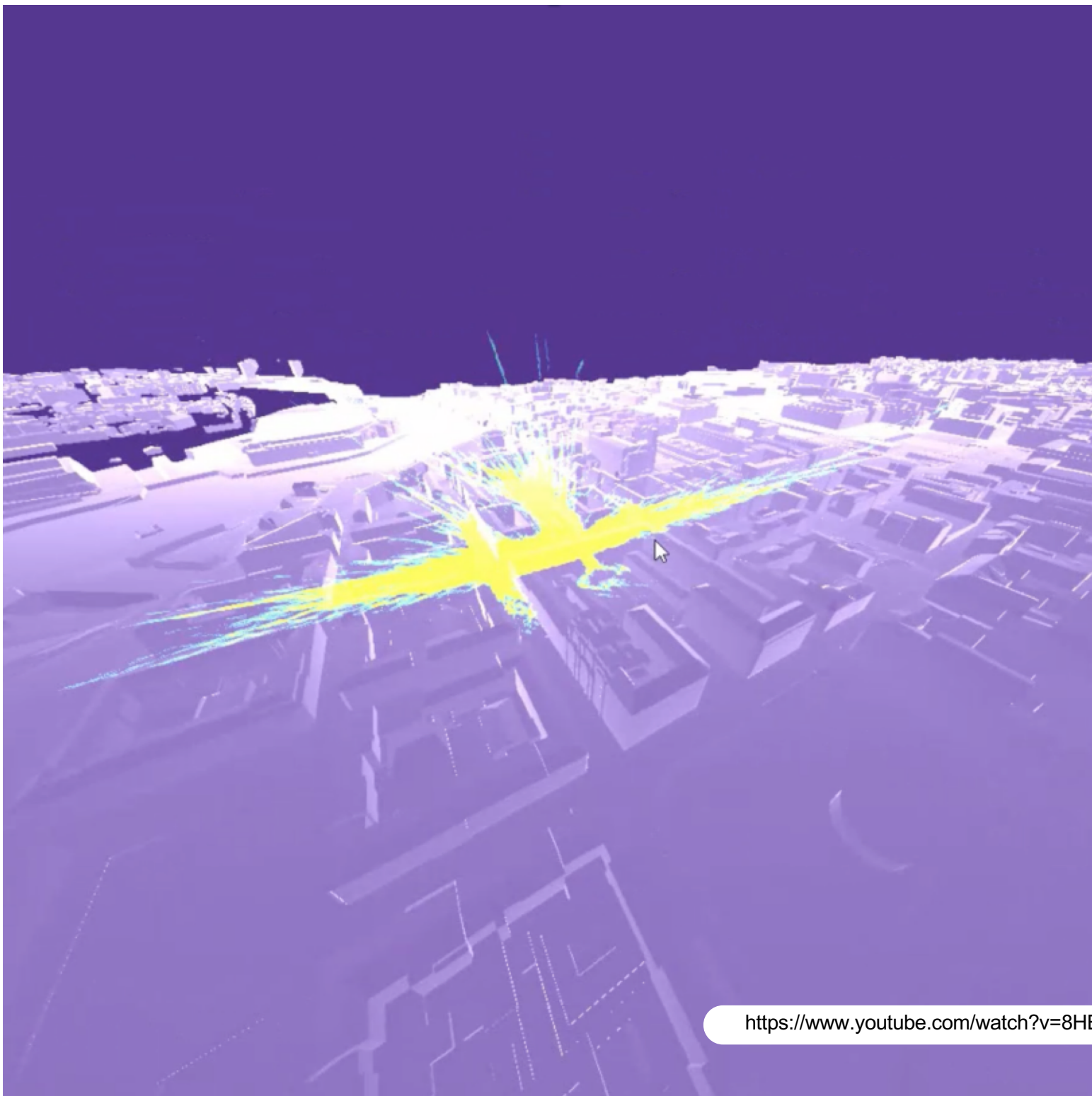
- + → Very simple and fast models ;
- → Not Site specific ;
- → Does not take into account multipath propagation phenomenon.

Modelling the radio propagation channel : Visibility tree



Modelling the radio propagation channel : Visibility tree



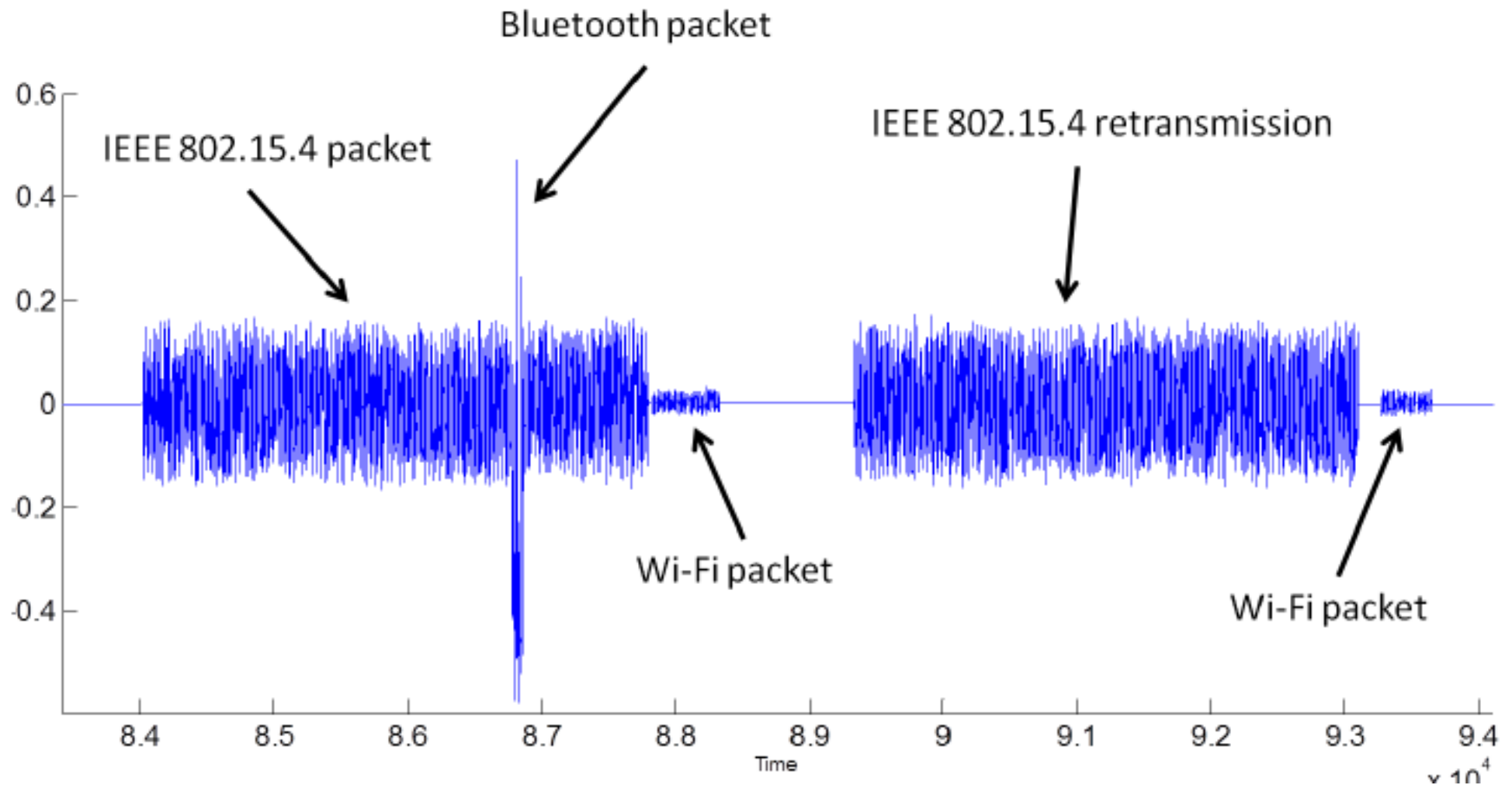


<https://www.youtube.com/watch?v=8HBhjVNIZgl>

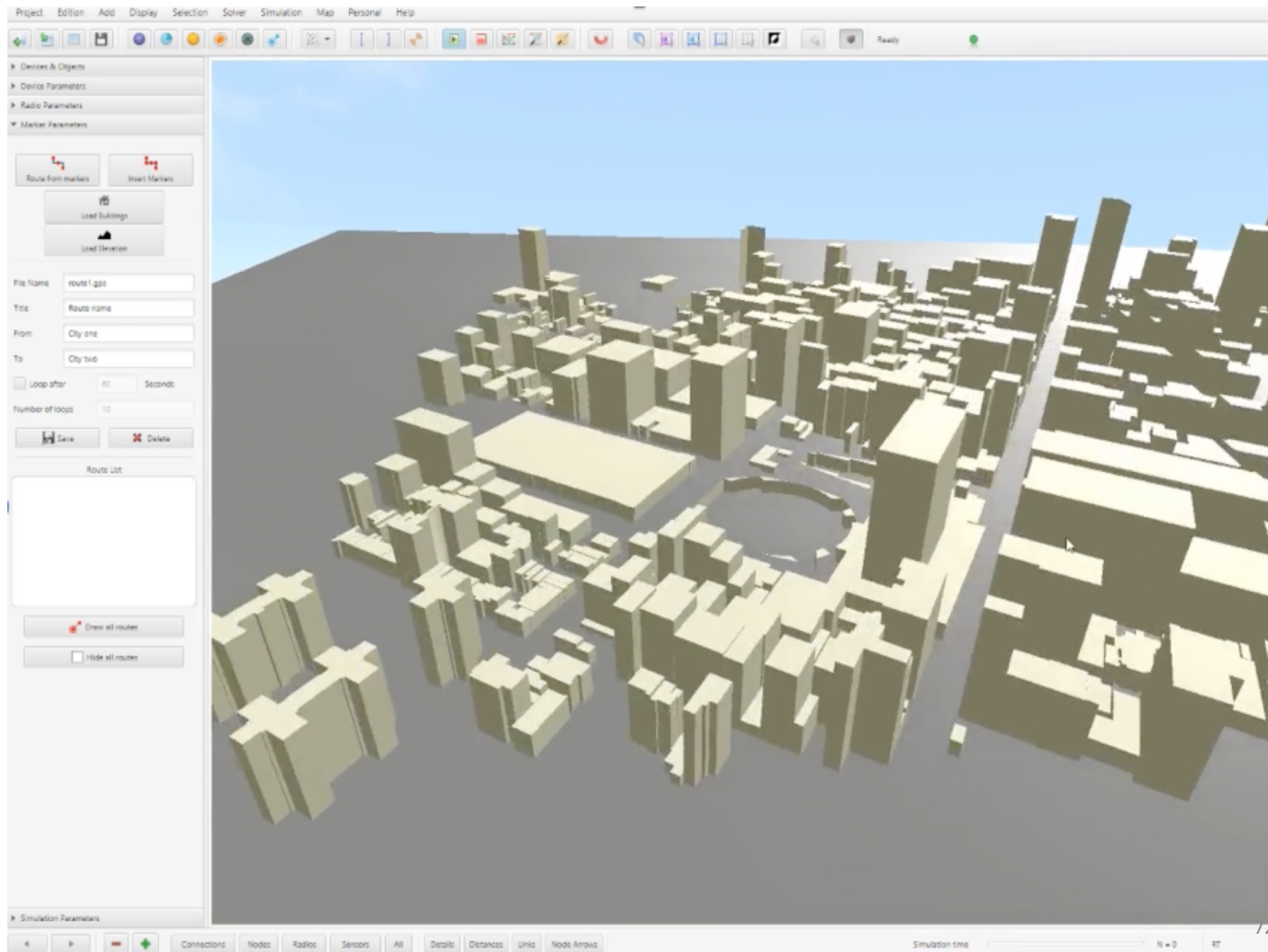


Interference

- Interference: noise characterization



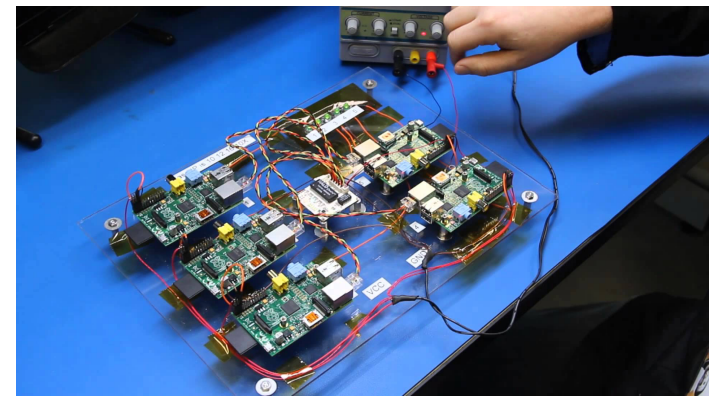
CupCarbon 3D (Demo)



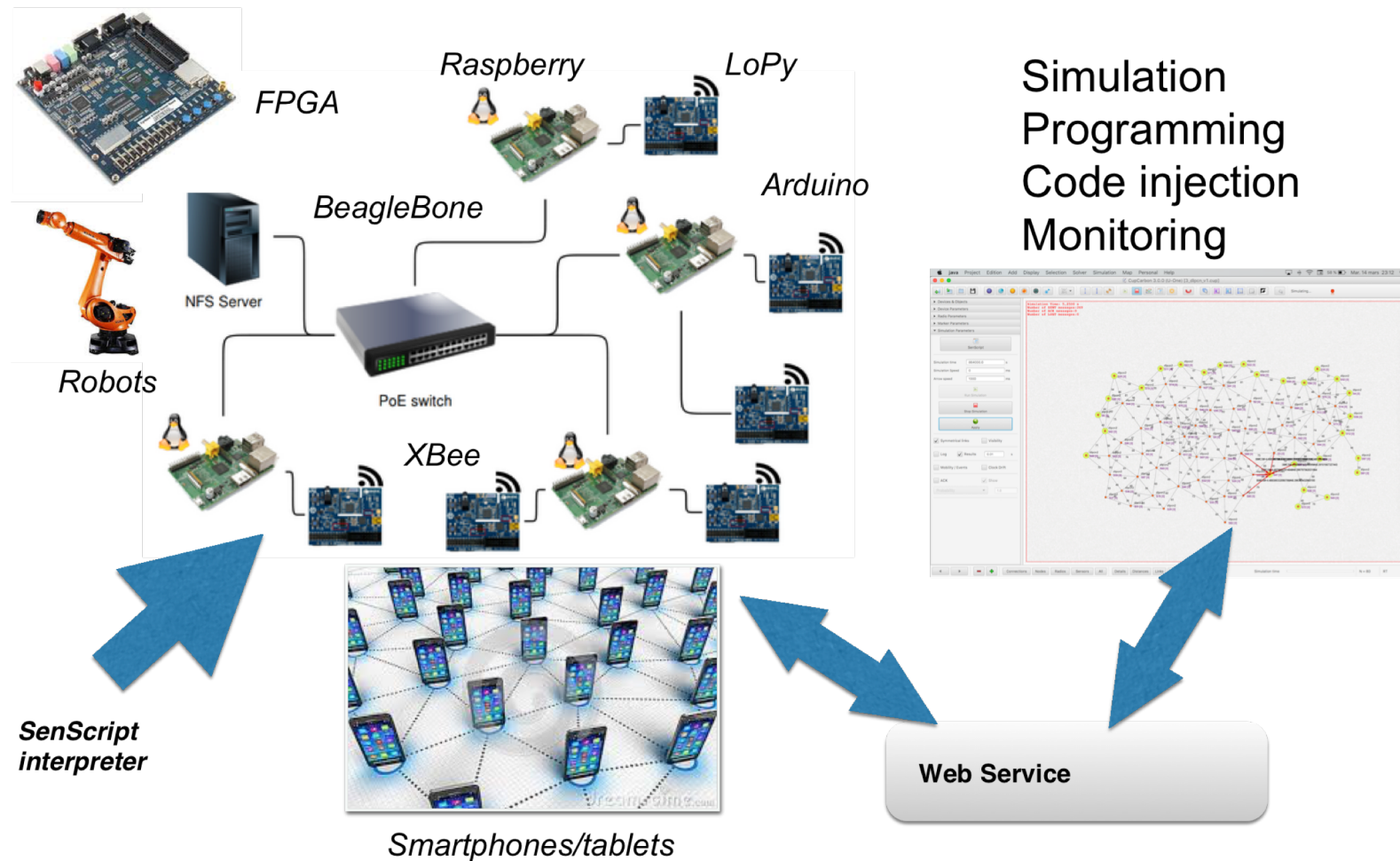
From simulation to real network



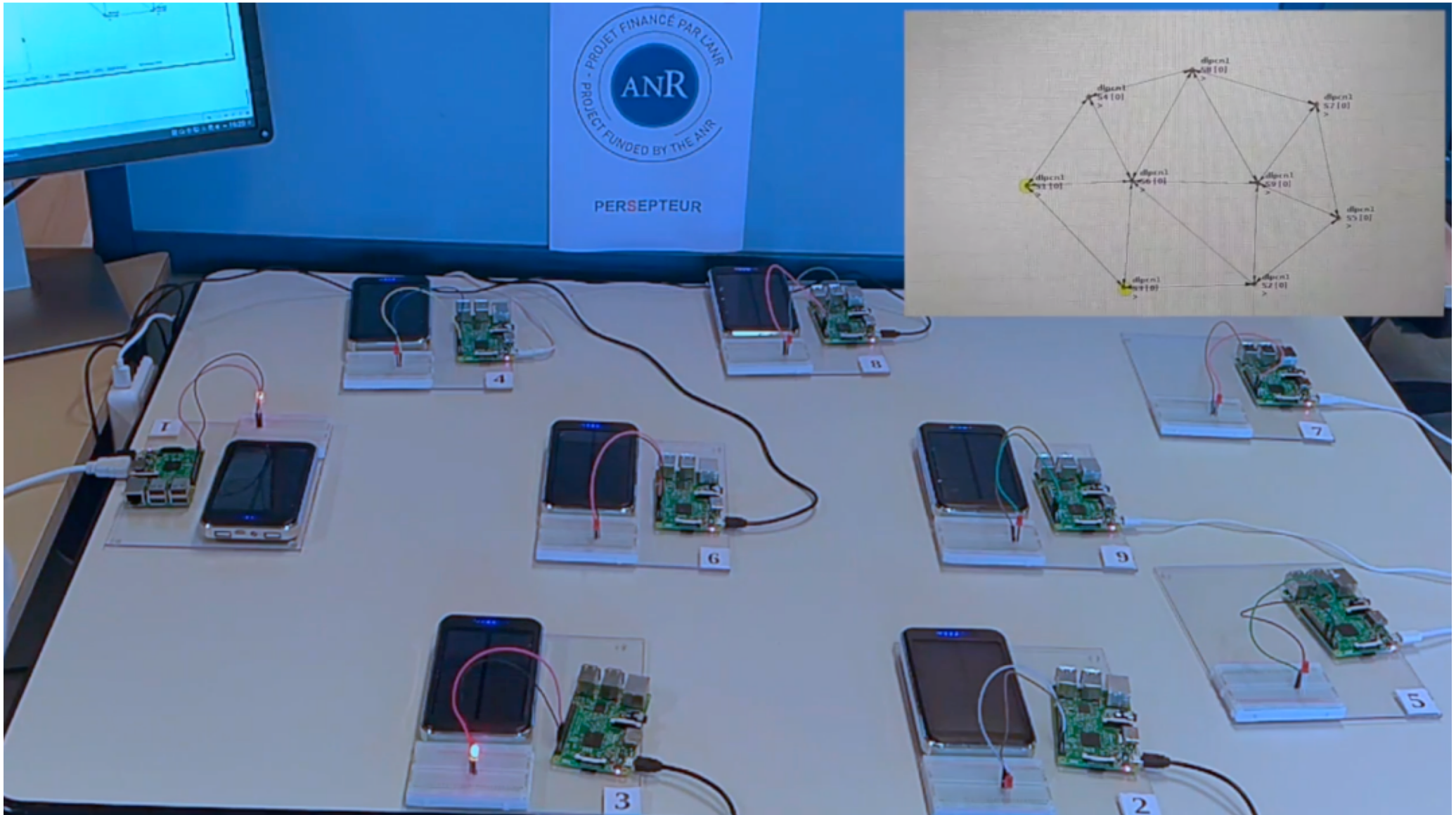
CupCarbonLab



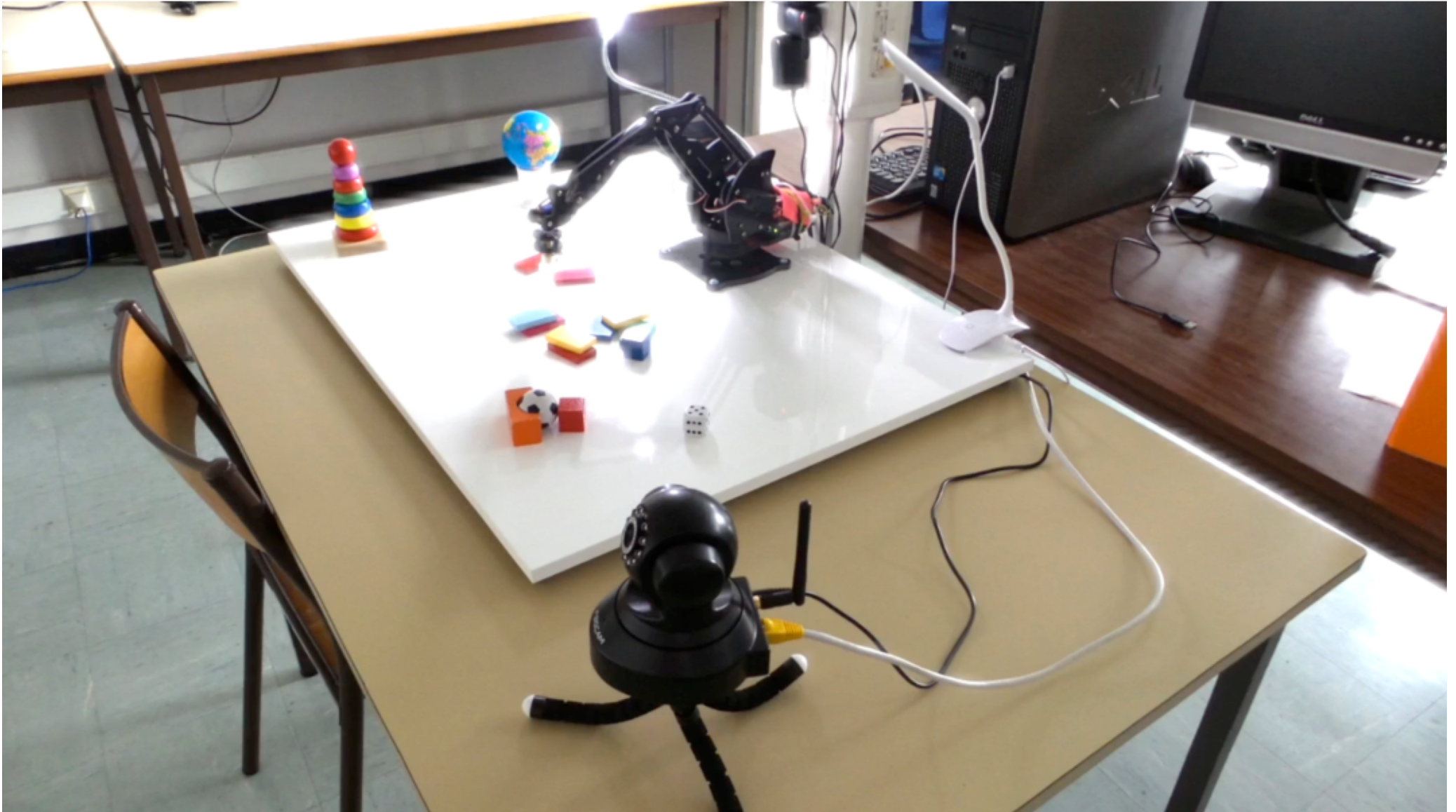
- CupCarbon-Lab: IoT Emulator
- Cybersecurity



BCS (By Card) Simulation – Direct Network Reconfiguration



BCS (By Card) Simulation – Remote Monitoring IoRGame Platform



CupCarbon : Quelques stats

450 visites / mois



Pays	Utilisateurs	% Utilisateurs
1. India	2 550	24,21 %
2. France	1 054	10,01 %
3. United States	969	9,20 %
4. United Kingdom	402	3,82 %
5. Brazil	379	3,60 %
6. Pakistan	289	2,74 %
7. Iran	279	2,65 %
8. Italy	251	2,38 %
9. Algeria	229	2,17 %
10. South Korea	220	2,09 %

CupCarbon : Quelques stats

150 Téléchargements / mois



Pays	Utilisateurs	% Utilisateurs
1. India	923	27,60 %
2. United States	233	6,97 %
3. United Kingdom	176	5,26 %
4. France	155	4,64 %
5. Pakistan	116	3,47 %
6. Ecuador	102	3,05 %
7. Brazil	101	3,02 %
8. Italy	101	3,02 %
9. Iran	94	2,81 %
10. Algeria	85	2,54 %

CupCarbon : Quelques stats

100 Téléchargements du guide utilisateurs / mois



Pays	Utilisateurs	% Utilisateurs
1. India	570	24,52 %
2. United States	221	9,51 %
3. France	174	7,48 %
4. Brazil	90	3,87 %
5. United Kingdom	90	3,87 %
6. Pakistan	71	3,05 %
7. Spain	64	2,75 %
8. Algeria	58	2,49 %
9. Ecuador	56	2,41 %
10. Iran	53	2,28 %

CupCarbon : Quelques stats

Evaluation of CupCarbon Network Simulator for Wireless Sensor Networks

Cristina López-Pavón¹, Sandra Sendra^{1,2}, Juan F. Valenzuela-Valdés¹

¹Dept. of Signal Theory, Telematics and Communications Department (TSTC),
Universidad de Granada, Granada, Spain

C/ Periodista Daniel Saucedo Aranda, s/n. 18071 Granada (Spain)

²Instituto de Investigación para la Gestión Integrada de zonas Costeras, Universitat
Politécnica de València.

C/ Paranimf nº 1. 46730 Grao de Gandía. Valencia (Spain)

E-mail: cristinalopezpavon@correo.ugr.es, ssendra@ugr.es, juanvalenzuela@ugr.es

Received: March 26, 2018

Accepted: June 21, 2018

Published: June 29, 2018

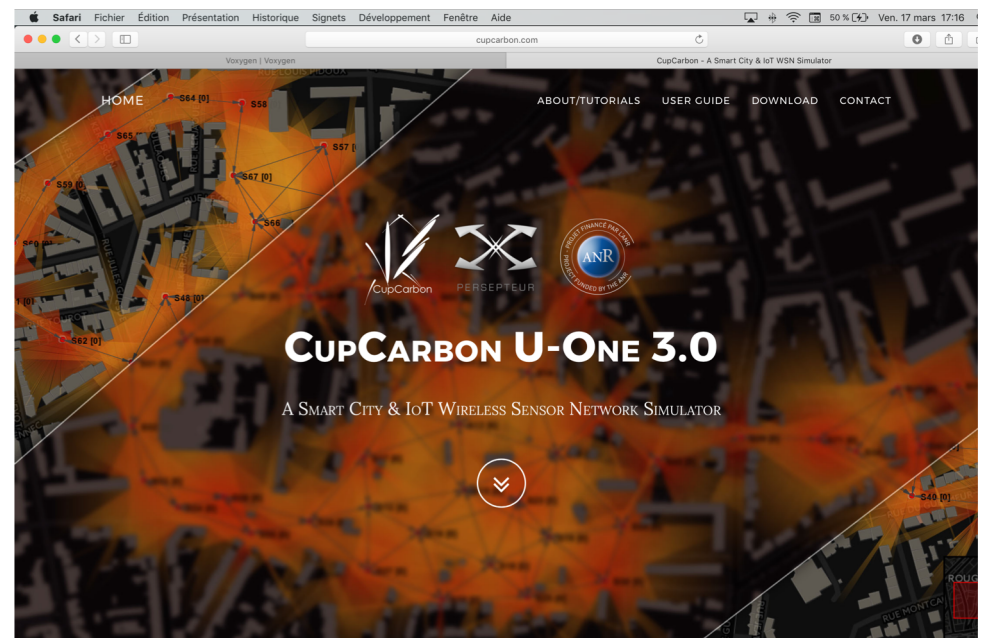
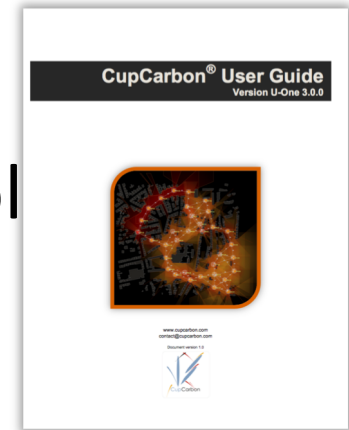
DOI: 10.5296/npa.v10i2.13201

URL: <https://doi.org/10.5296/npa.v10i2.13201>

CupCarbon web site

www.cupcarbon.com

- Presentation of the simulator (video)
- Source code (Git Hub) and Jar file (executable)
- User Guide
- 17 (Basic) + 5 (Advanced) tutorials
- 2 tutorials (video)
- Newsletter
- Forum
- Facebook page



3

POLYGON HULLS & LEADER ELECTION

- 3.1. LPCN Algorithm*
- 3.2. D-LPCN Algorithm*
- 3.3. Leader Election*

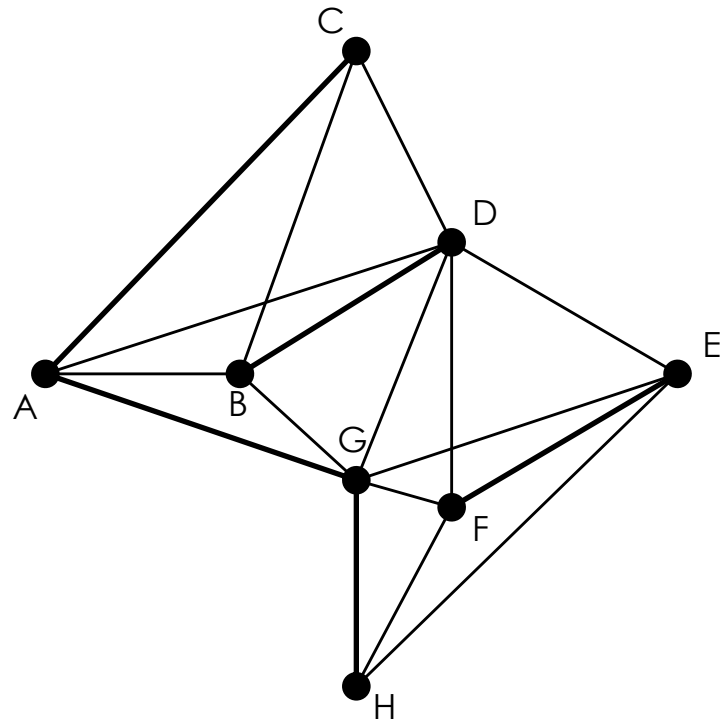
Ahcène Bounceur, Trouver l'Enveloppe Polygonale d'un Réseau de Capteurs Sans Fil : Théorie et Pratique, séminaire, GDR-RO, CEA, Paris, May 2015.

Ahcène Bounceur, Madani Bezoui, Reinhardt Euler and Marc Sevaux, Finding the Boundary Nodes of a Wireless Sensor Network Without Conditions on the Starting Node, Invited talk, EURO 2016, Poznan.

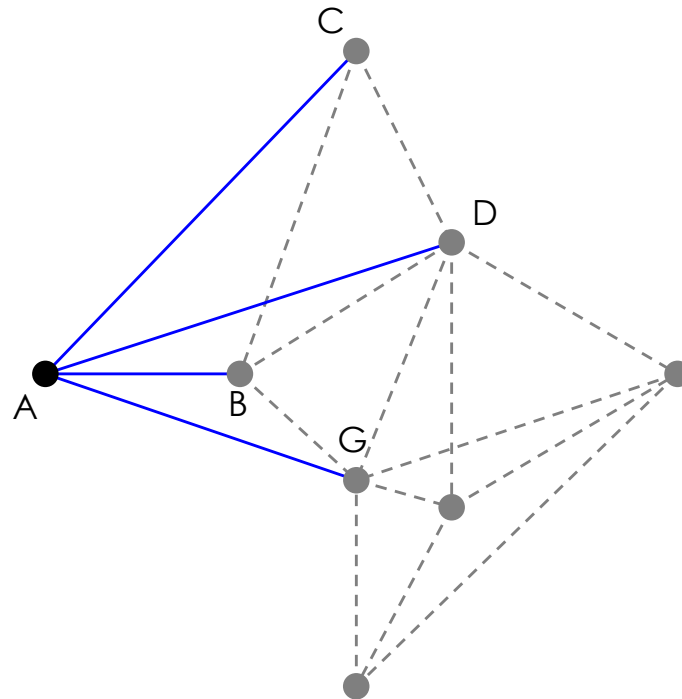
3.1

LPCN ALGORITHM

LPCN Algorithm

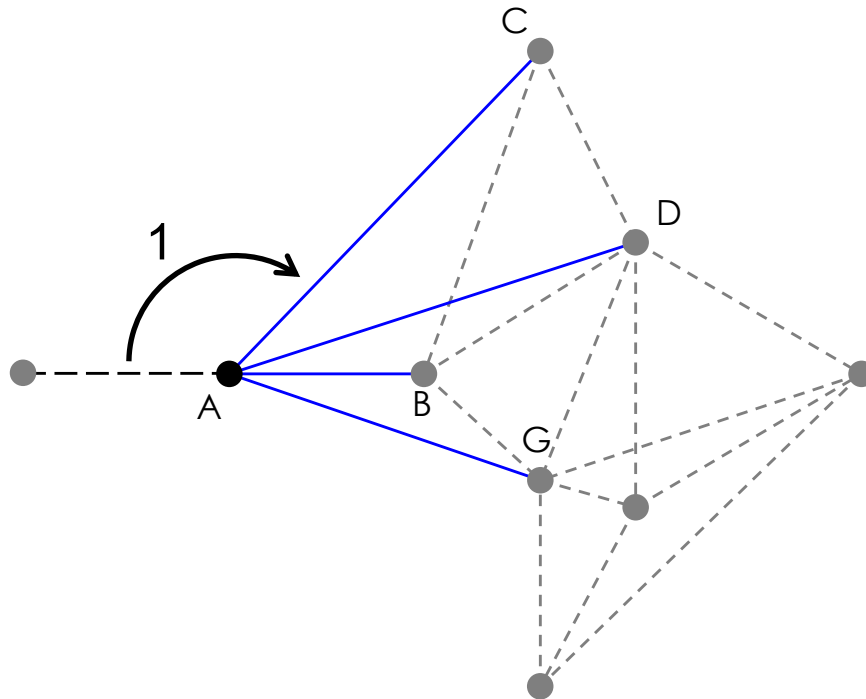


LPCN Algorithm



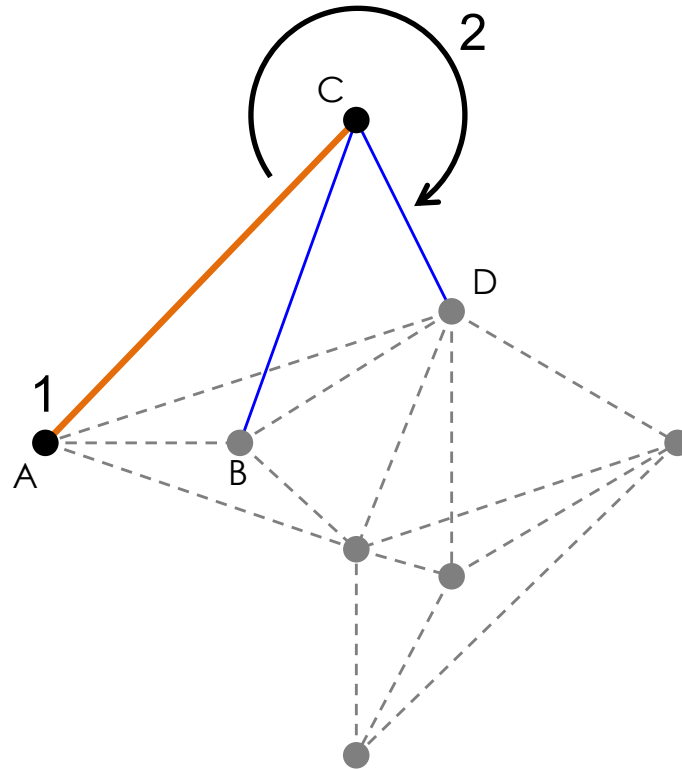
[A]

LPCN Algorithm



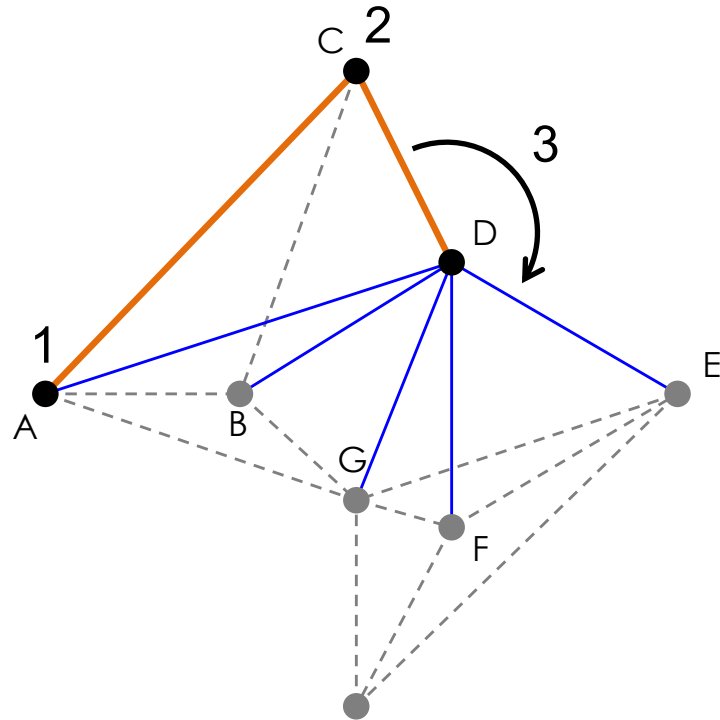
[A]

LPCN Algorithm



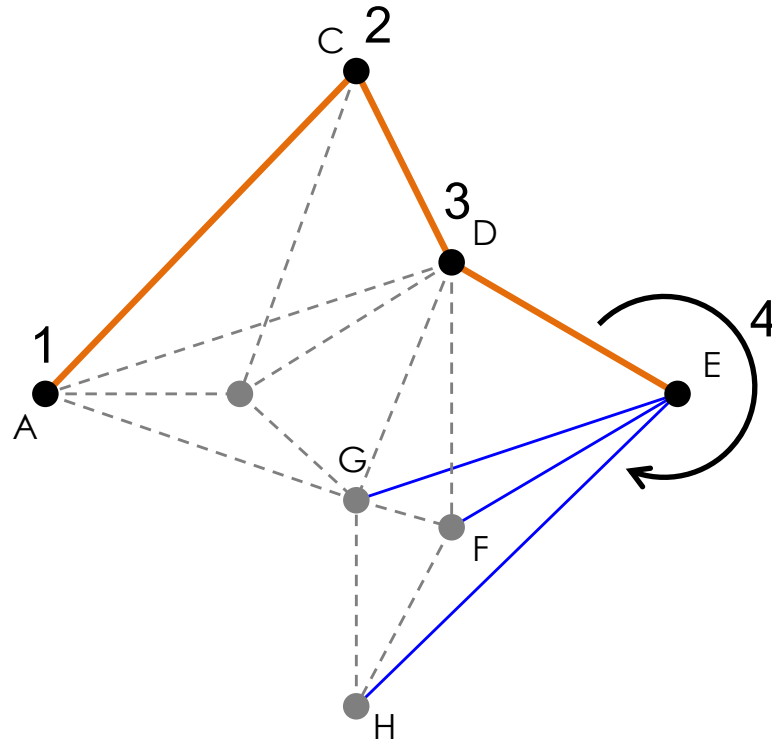
[A, C]

LPCN Algorithm



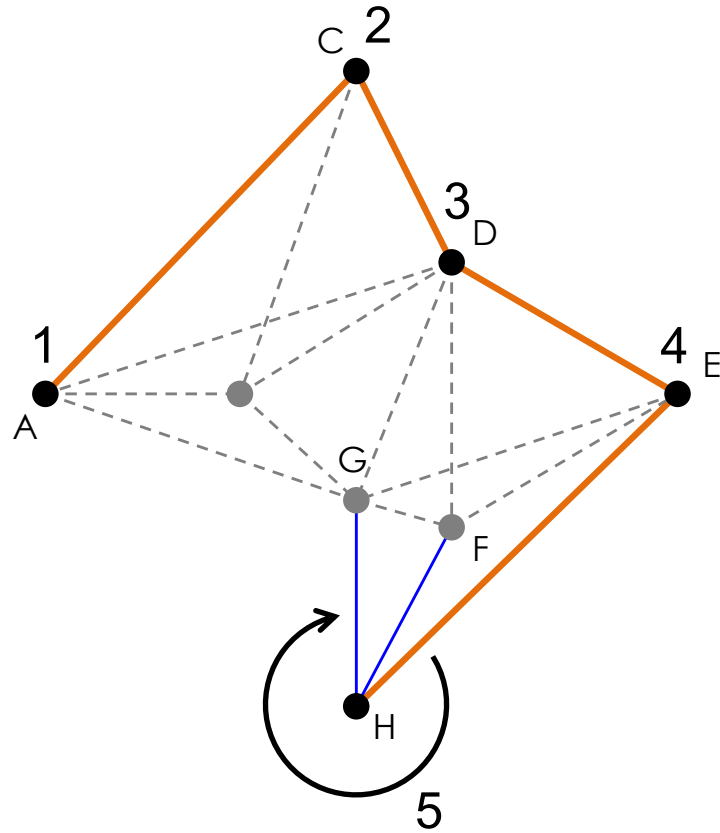
[A, C, D]

LPCN Algorithm



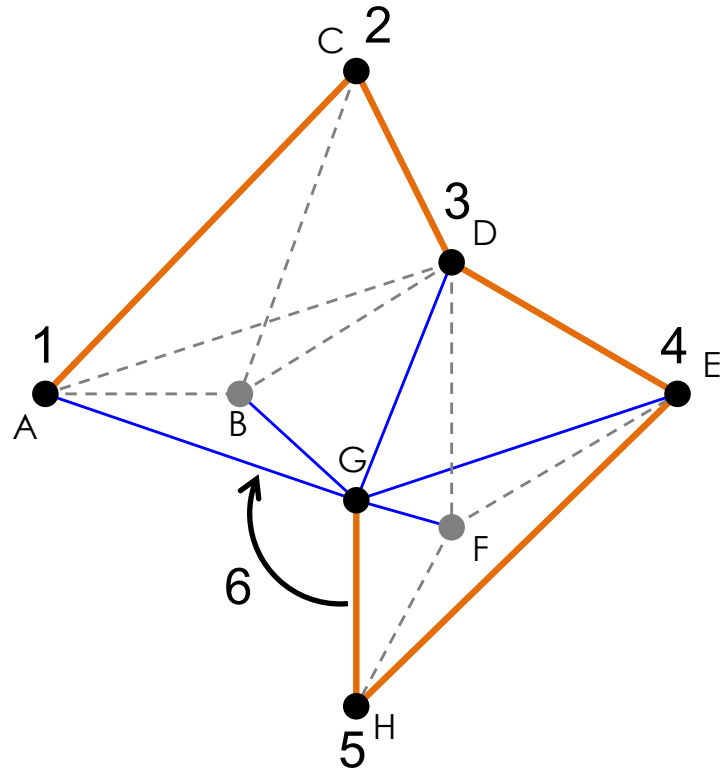
[A, C, D, E]

LPCN Algorithm



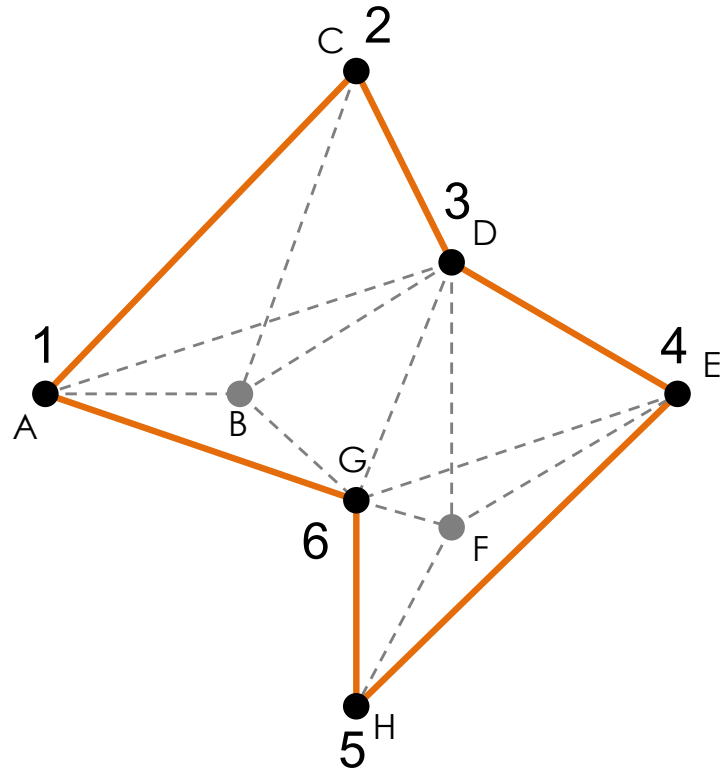
[A, C, D, E, H]

LPCN Algorithm



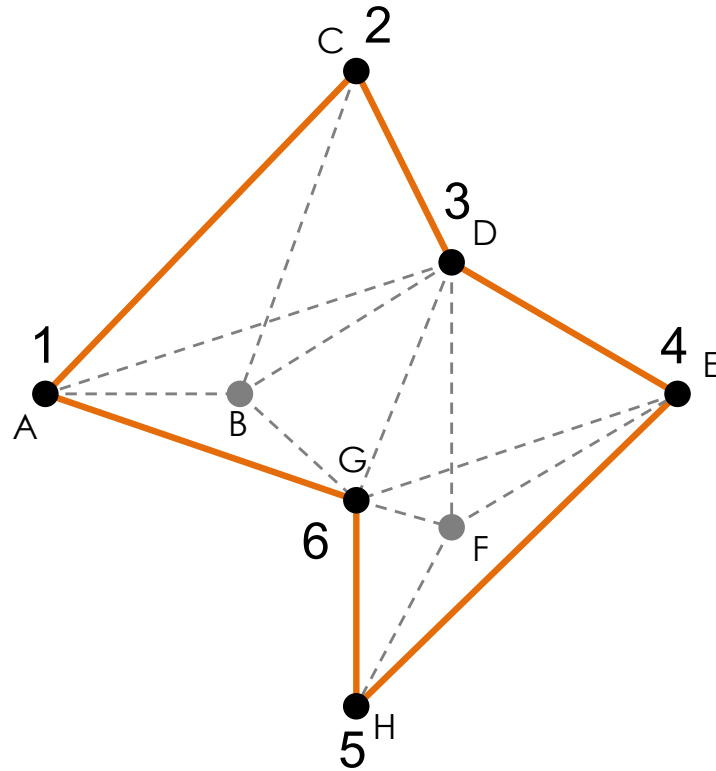
[A, C, D, E, H, G]

LPCN Algorithm



[A, C, D, E, H, G, A]

LPCN Algorithm



[A, C, D, E, H, G, A]

Version 1

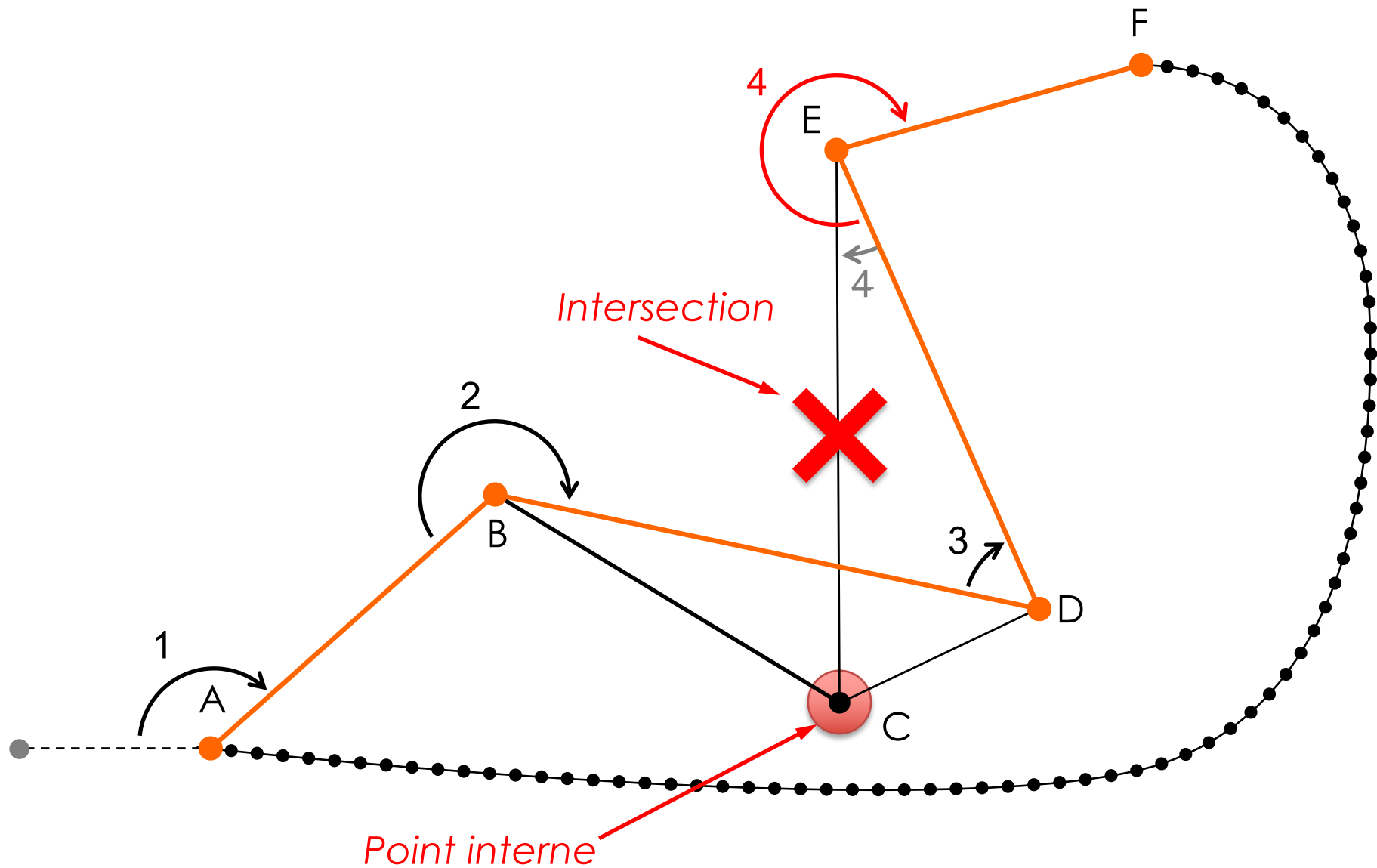
```

2:  $P_c \leftarrow$  point having the minimum x-coordinate
3:  $\mathbb{B}_V \leftarrow [P_c]$ 
4:  $P_f \leftarrow P_c$ 
5:  $P_p \leftarrow$  fictive point situated on the left of  $P_f$ 
6: repeat
7:    $P_v = \underset{P_j \in N(P_c)}{\operatorname{argmin}} \{pa(P_p, P_c, P_j)\}$ 
8:    $\mathbb{B}_V \leftarrow [\mathbb{B}_V, P_v]$ 
9:    $P_p \leftarrow P_c$ 
10:   $P_c \leftarrow P_v$ 
11: until  $P_v = P_f$ 
12: return  $\mathbb{B}_V$ 
13: end procedure
  
```

LPCN Algorithm

- Version 1
 - Complexity: $O(kh)$
 - k : Maximum degree of the graph
 - h : Number of the border nodes

LPCN Algorithm: with a Boat graph



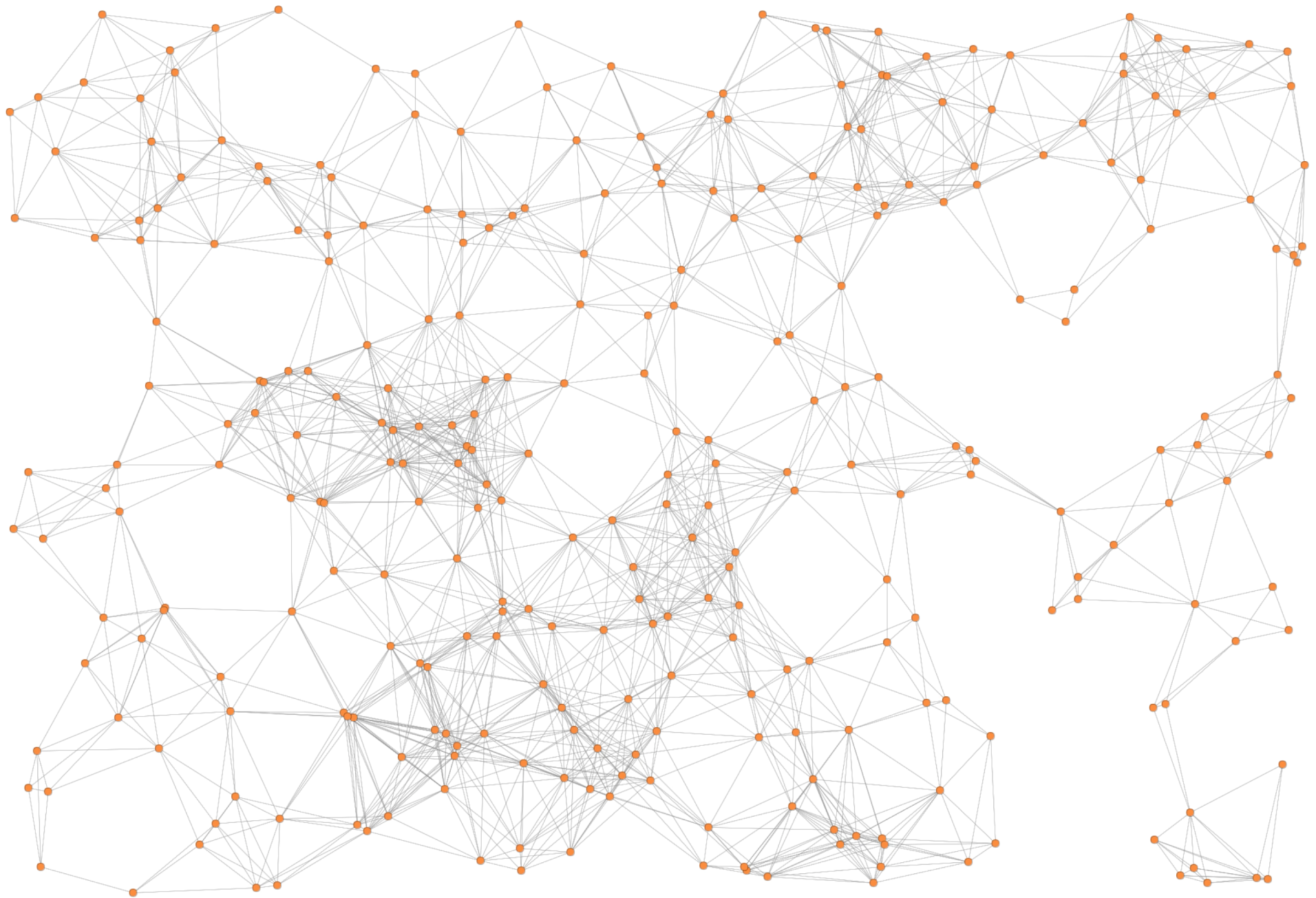
LPCN Algorithm: Version 2

```
1: procedure LPCN( $V, E$ )
2:    $P_0 \leftarrow P_c \leftarrow$  point having the minimum  $x$ -coordinate
3:    $P_p \leftarrow$  fictive point situated in the left of  $P_c$ 
4:    $\mathbb{B}_V \leftarrow \{P_c\}$ 
5:    $\mathbb{B}_E \leftarrow \emptyset$ 
6:    $once \leftarrow true$ 
7:   repeat
8:      $\mathbb{A} \leftarrow \{P \in N(P_c) / \mathbb{B}_E \cap \{\{P_c, P\}\} = \emptyset\}$ 
9:      $P_{min} \leftarrow \underset{P \in \mathbb{A}}{\operatorname{argmin}} \{\varphi(P_p, P_c, P)\}$ 
10:     $\mathbb{B}_V \leftarrow \mathbb{B}_V \cup \{P_{min}\}$ 
11:     $\mathbb{B}_E \leftarrow \mathbb{B}_E \cup \{\{P_c, P_{min}\}\}$ 
12:     $P_p \leftarrow P_c$ 
13:     $P_c \leftarrow P_{min}$ 
14:    if ( $once = true$ ) then
15:       $once \leftarrow false$ 
16:       $P_{first} \leftarrow P_{min}$ 
17:    end if
18:  until (( $P_c = P_0$ ) and ( $P_{min} = P_{first}$ ))
19:  return  $\mathbb{B}_V, \mathbb{B}_E$ 
20: end procedure
```

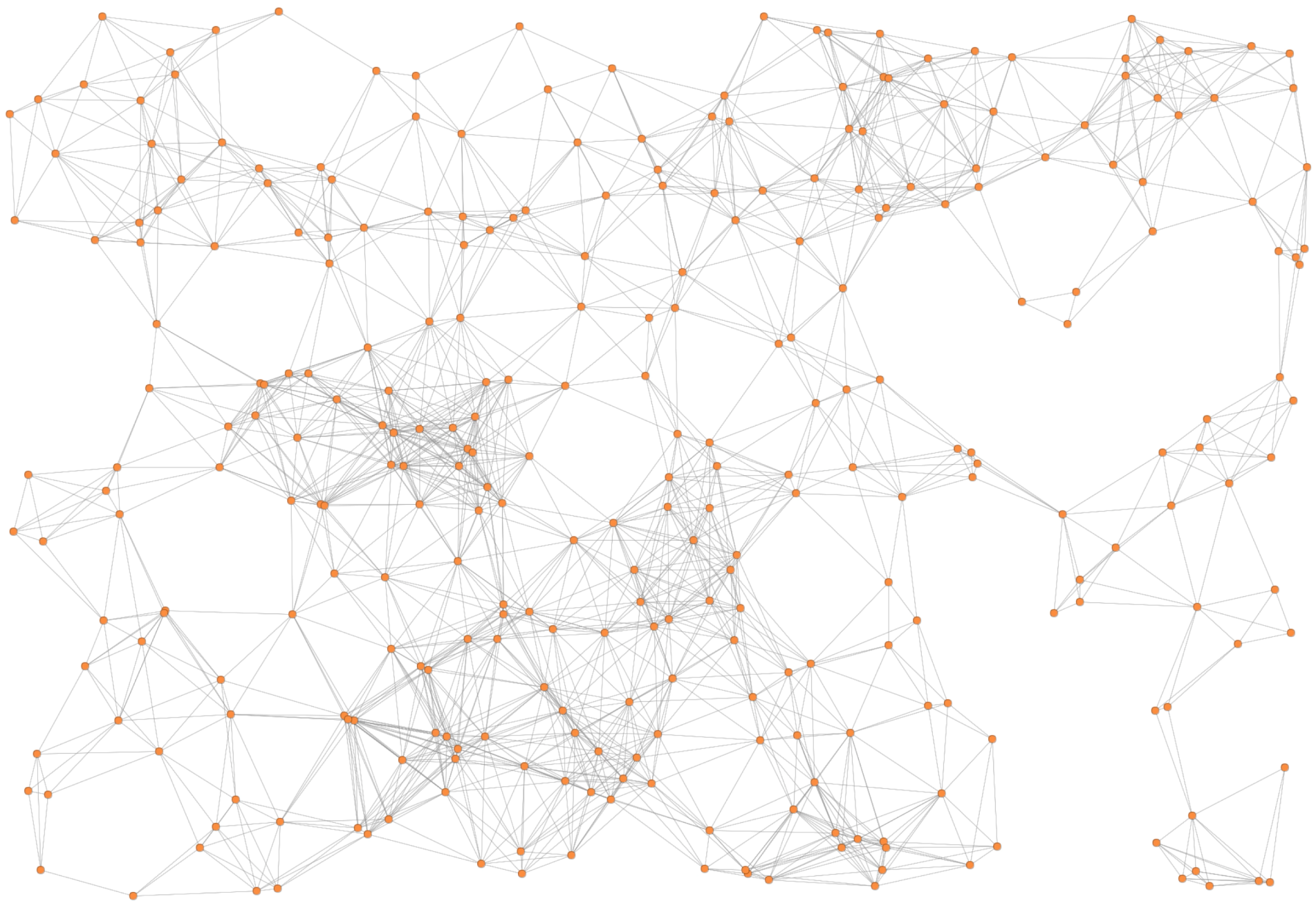
LPCN Algorithm

- Version 1
 - Complexity: $O(kh)$
 - k : Maximum degree of the graph
 - h : Number of the border nodes
- Version 2
 - Complexity: $O(kh^2)$
 - k : Maximum degree of the graph
 - $h(h+1)/2$: Number of the border nodes by considering the intersection detection

LPCN Algorithm: Demo 1



LPCN Algorithm: Demo 2



3.2

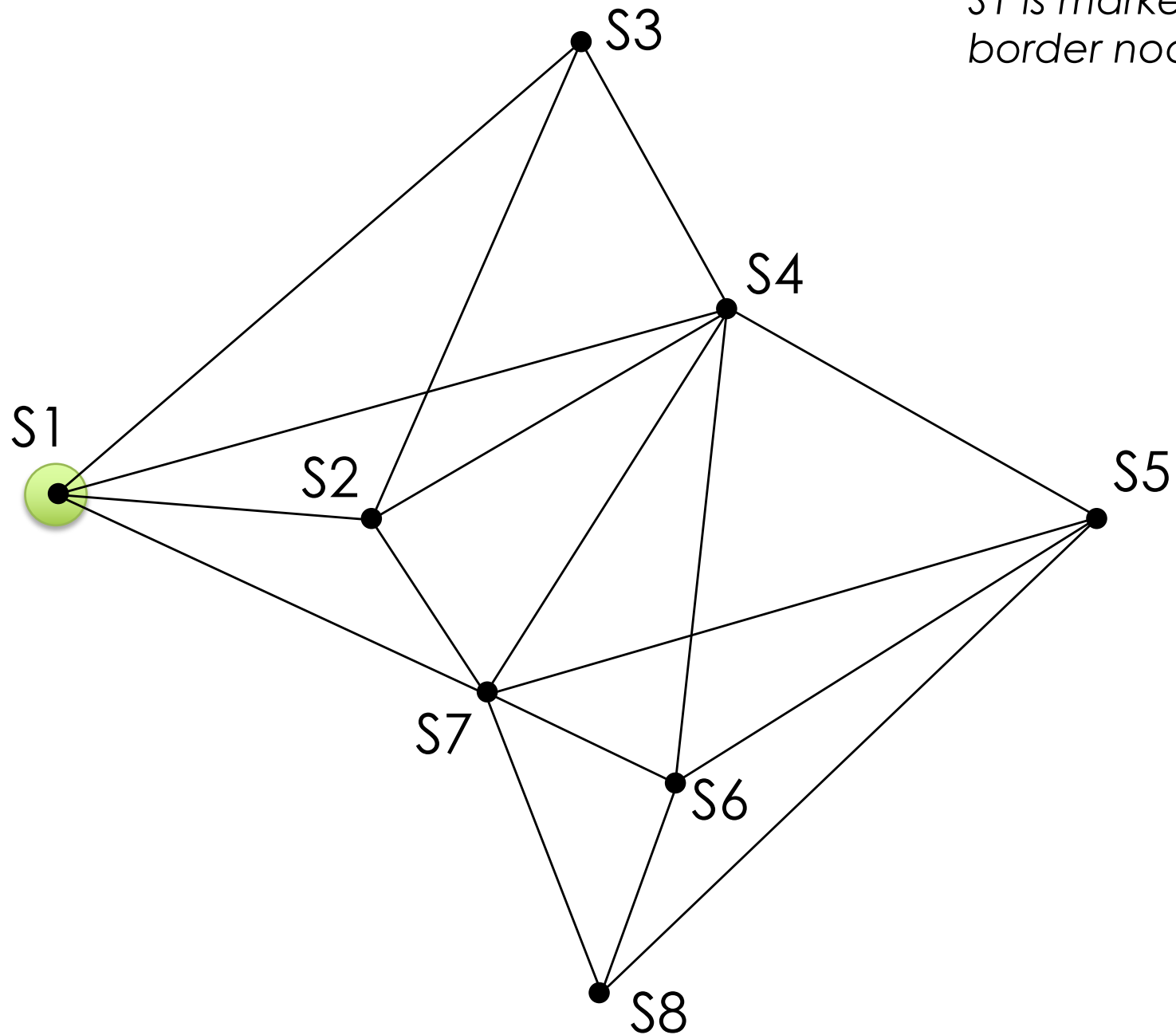
D-LPCN ALGORITHM

D-LPCN Algorithm: Messages

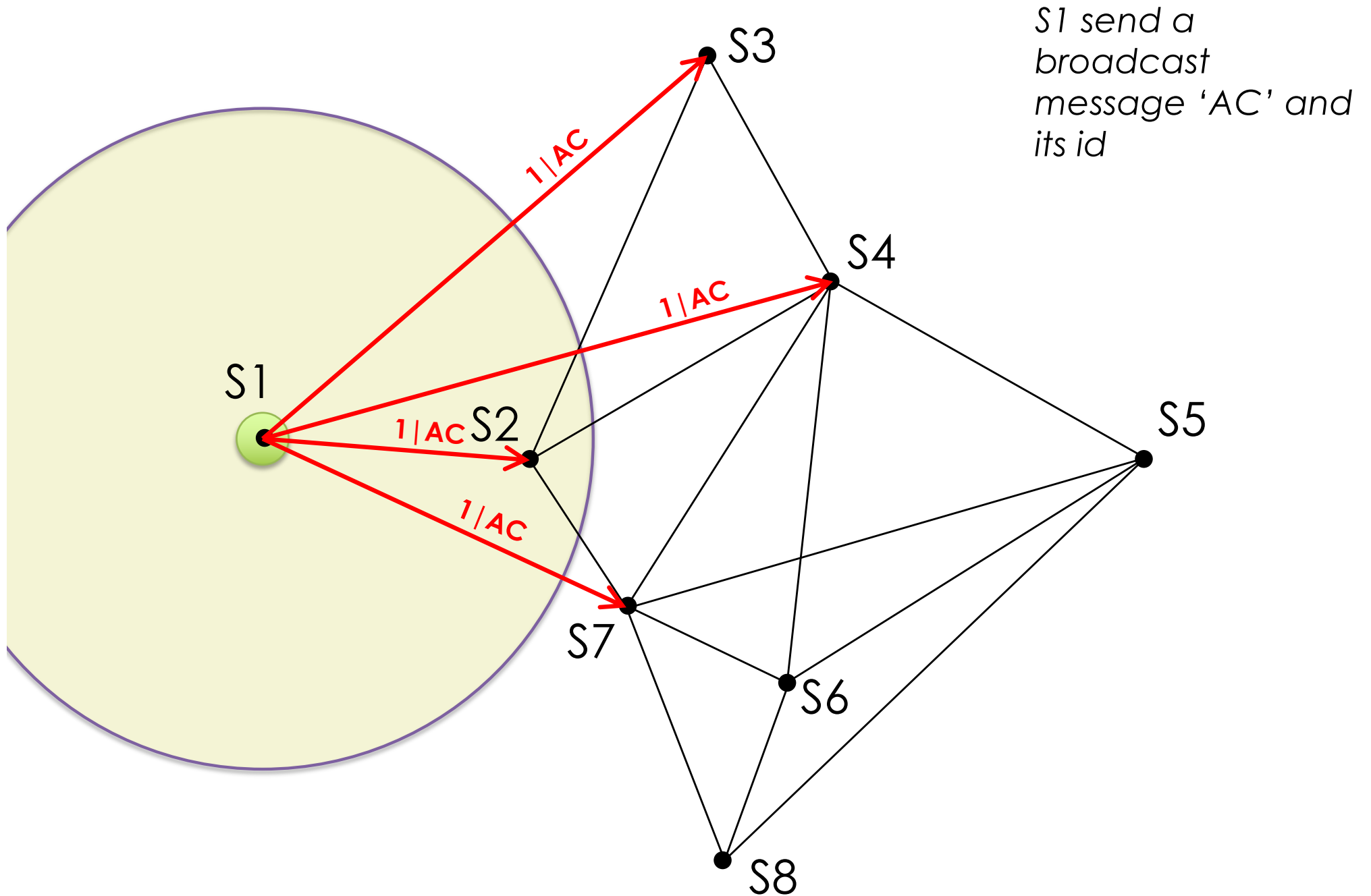
- 3 types of messages :
 - AC : Ask for Coordinates
 - CS : Coordinate Sending
 - SN : Select Node
- Messages format:
 - Message AC : **id | AC**
 - Message CS : **id | CS | x,y**
 - Message SN : **id | SN | x,y**

D-LPCN Algorithm

*S1 is marked as a
border node*



D-LPCN Algorithm



D-LPCN Algorithm

The diagram illustrates a network of 8 nodes (S1-S8) and their communication paths. Node S1 is highlighted in green, while nodes S2-S8 are red. The nodes are distributed across four overlapping yellow regions. Blue arrows indicate the flow of information from S2, S3, and S4 to S1, with labels indicating the source and the set of nodes involved in the communication:

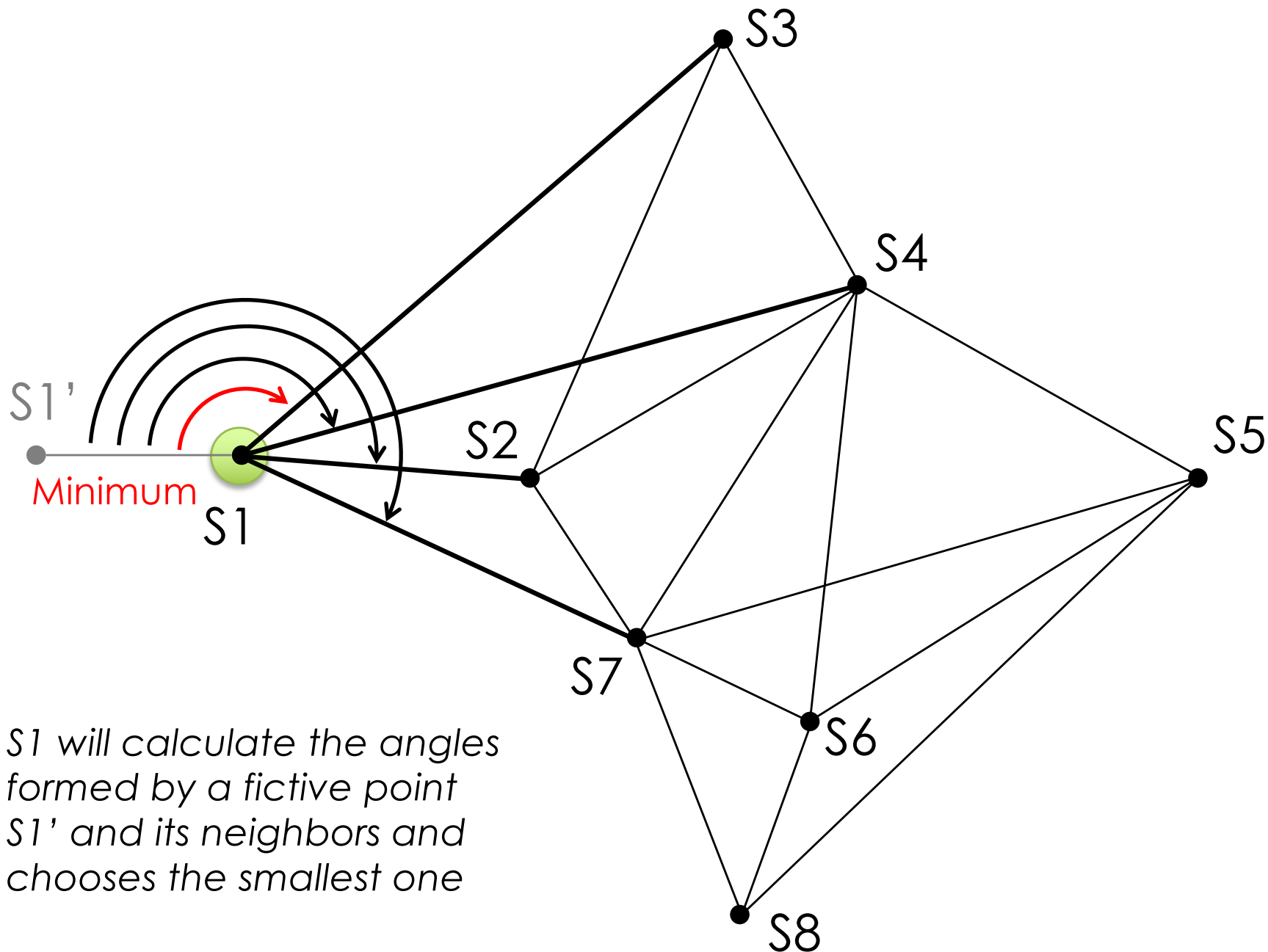
- From S2 to S1: $2|CS|2,4$
- From S3 to S1: $3|CS|4,8$
- From S4 to S1: $4|CS|7,7$
- From S7 to S1: $7|CS|3,3$

Black lines represent the network topology, showing connections between S1 and S2, S3, S4, S7; S2 and S3, S4, S7; S3 and S4, S7; S4 and S5, S6, S7, S8; S5 and S6, S7, S8; S6 and S7, S8; and S7 and S8.

S2, S3, S4 can receive the message 'id' and send to S1 the coordinate id and the message 'id'.

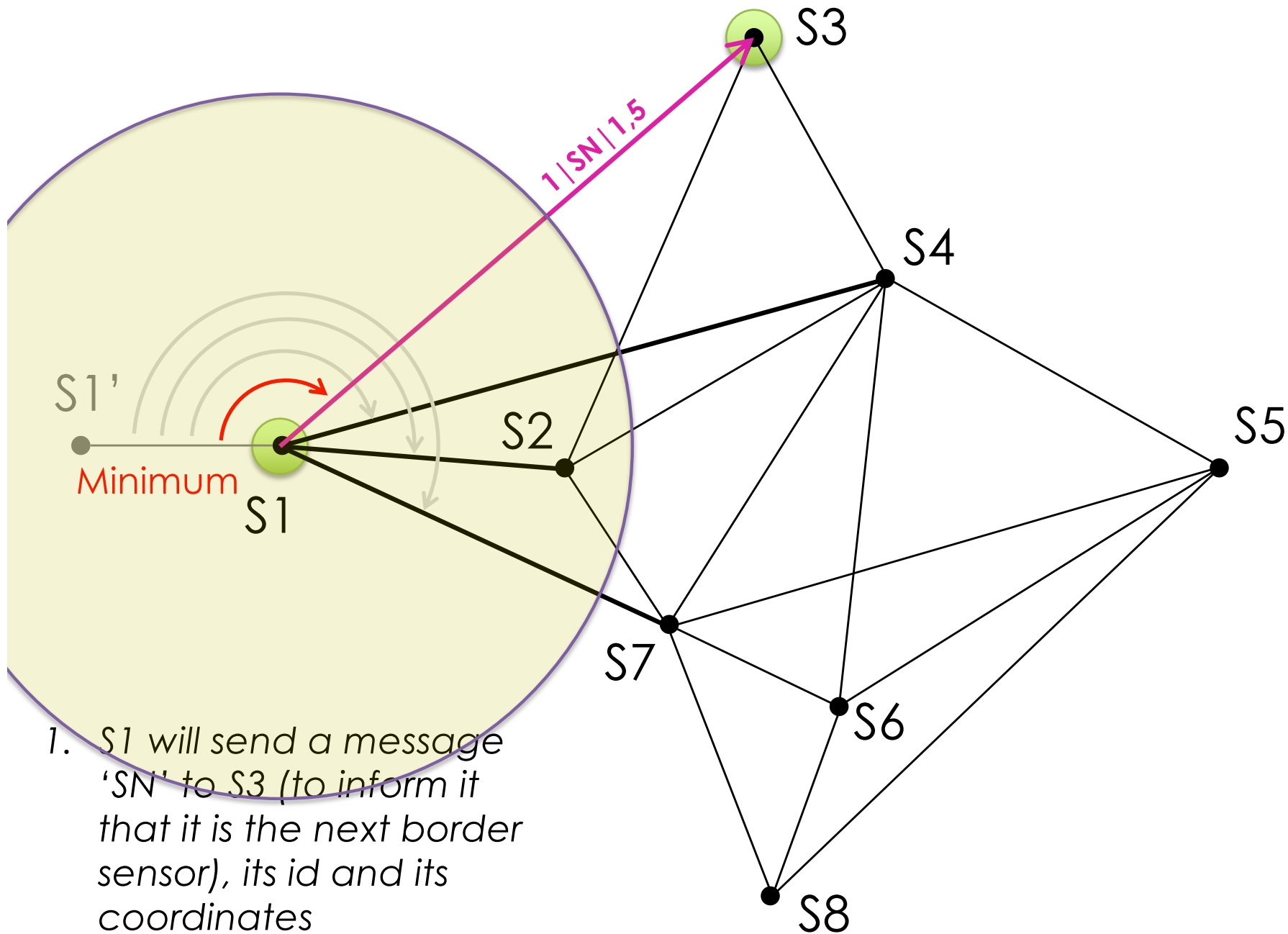
*S2, S3, S4 and S7
receive the
message 'AC' and
send to S1 their
coordinates, their
id and the
message 'CS'*

D-LPCN Algorithm

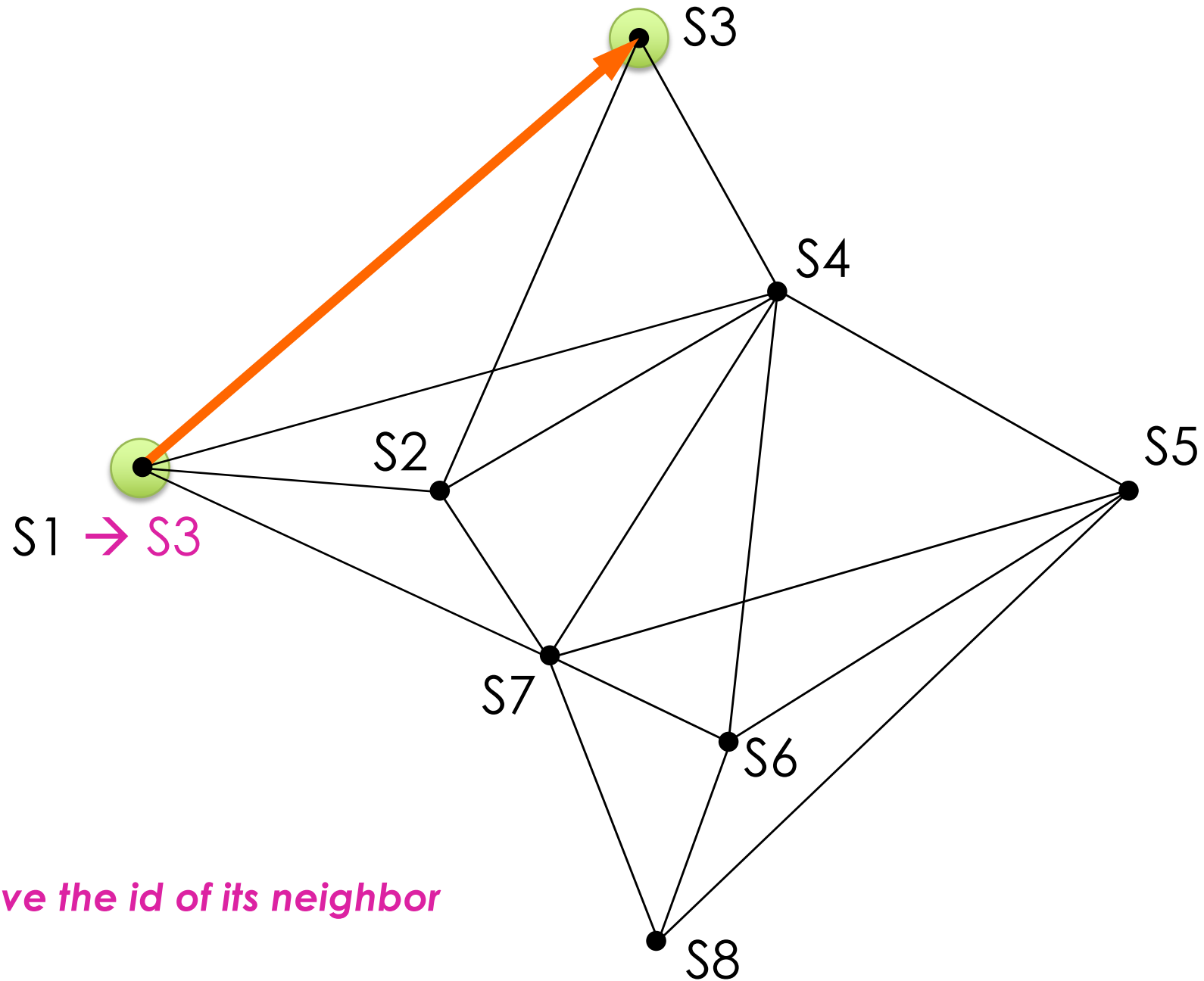


$S1$ will calculate the angles formed by a fictive point $S1'$ and its neighbors and chooses the smallest one

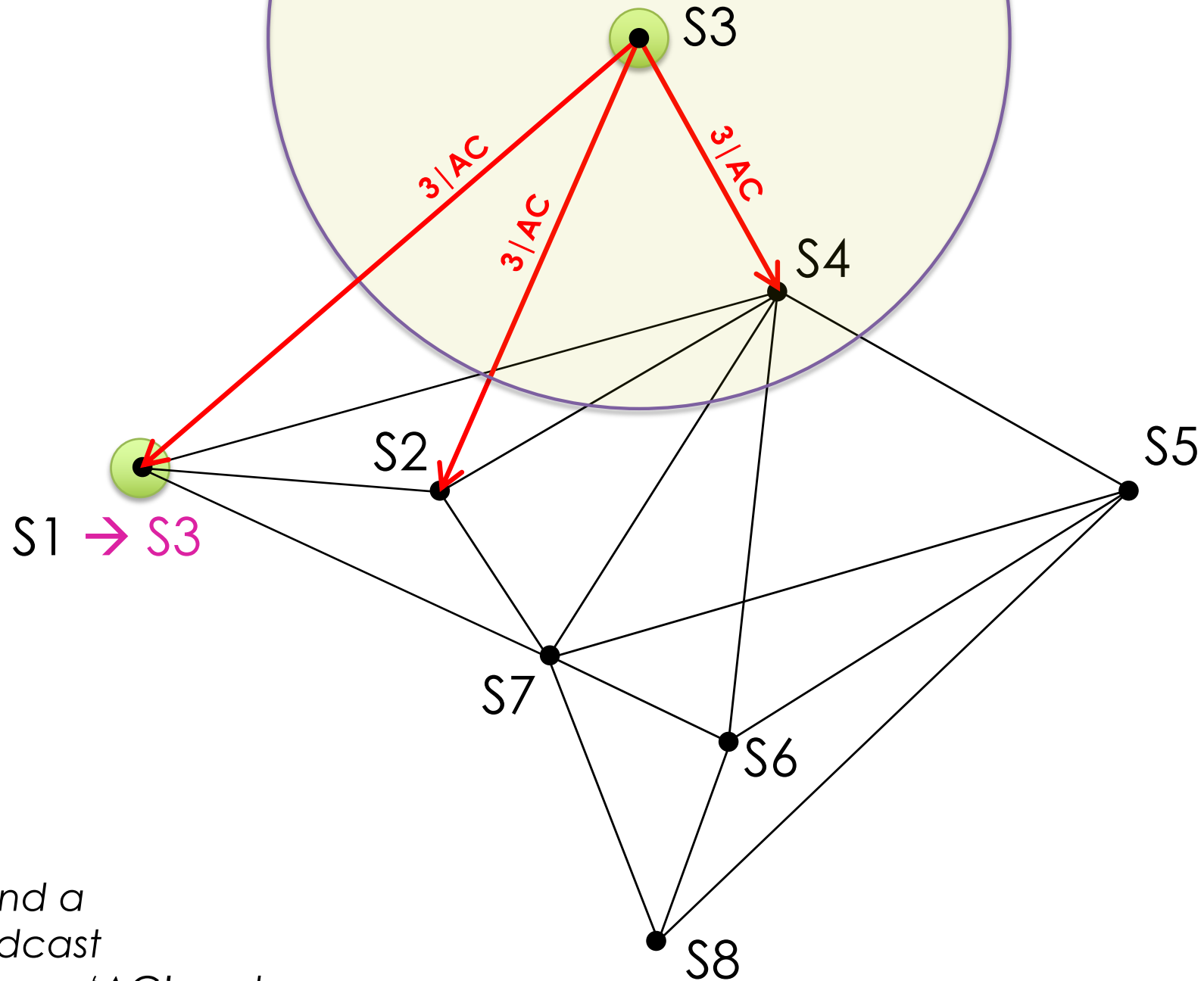
D-LPCN Algorithm



D-LPCN Algorithm

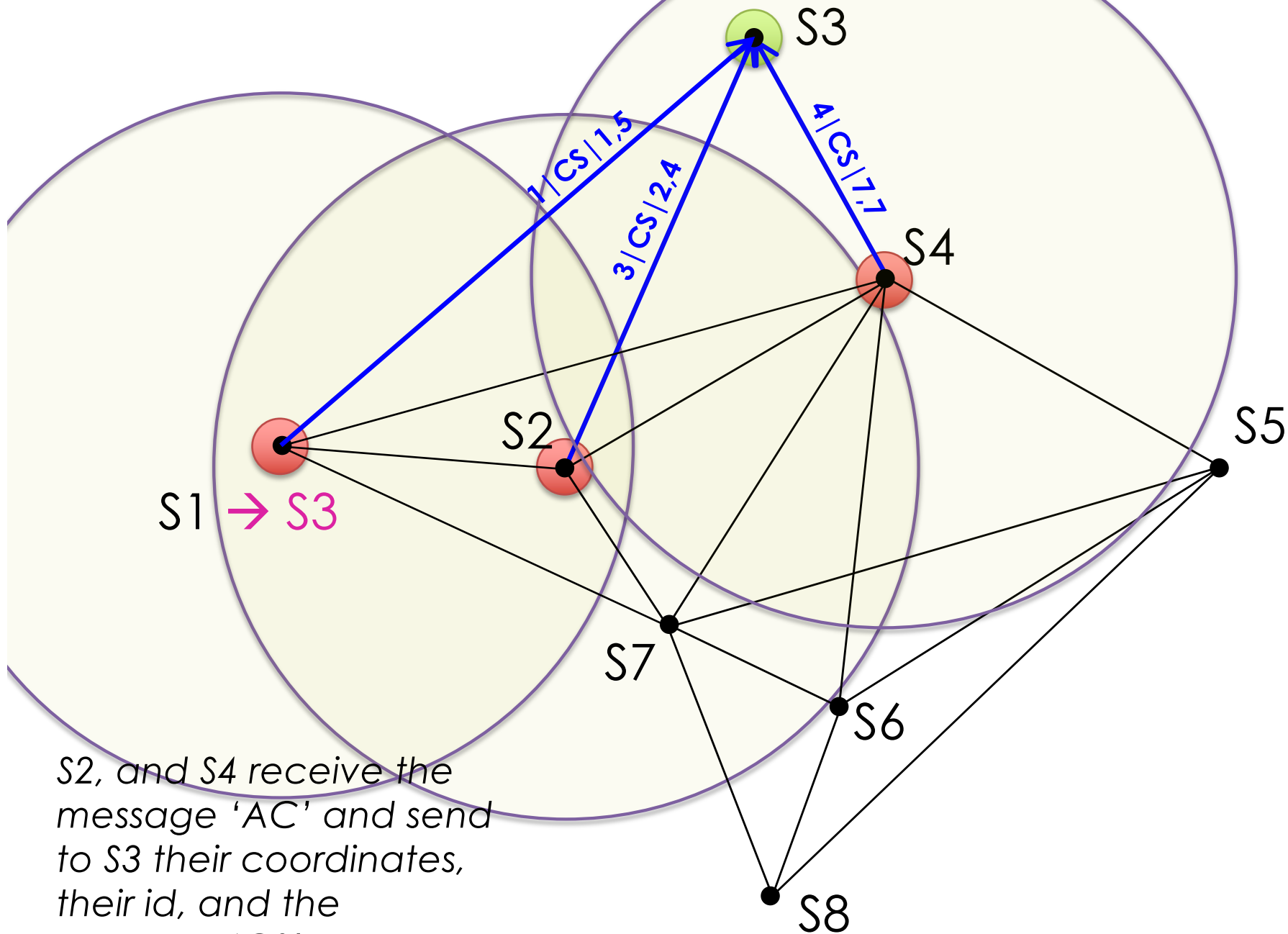


D-LPCN Algorithm



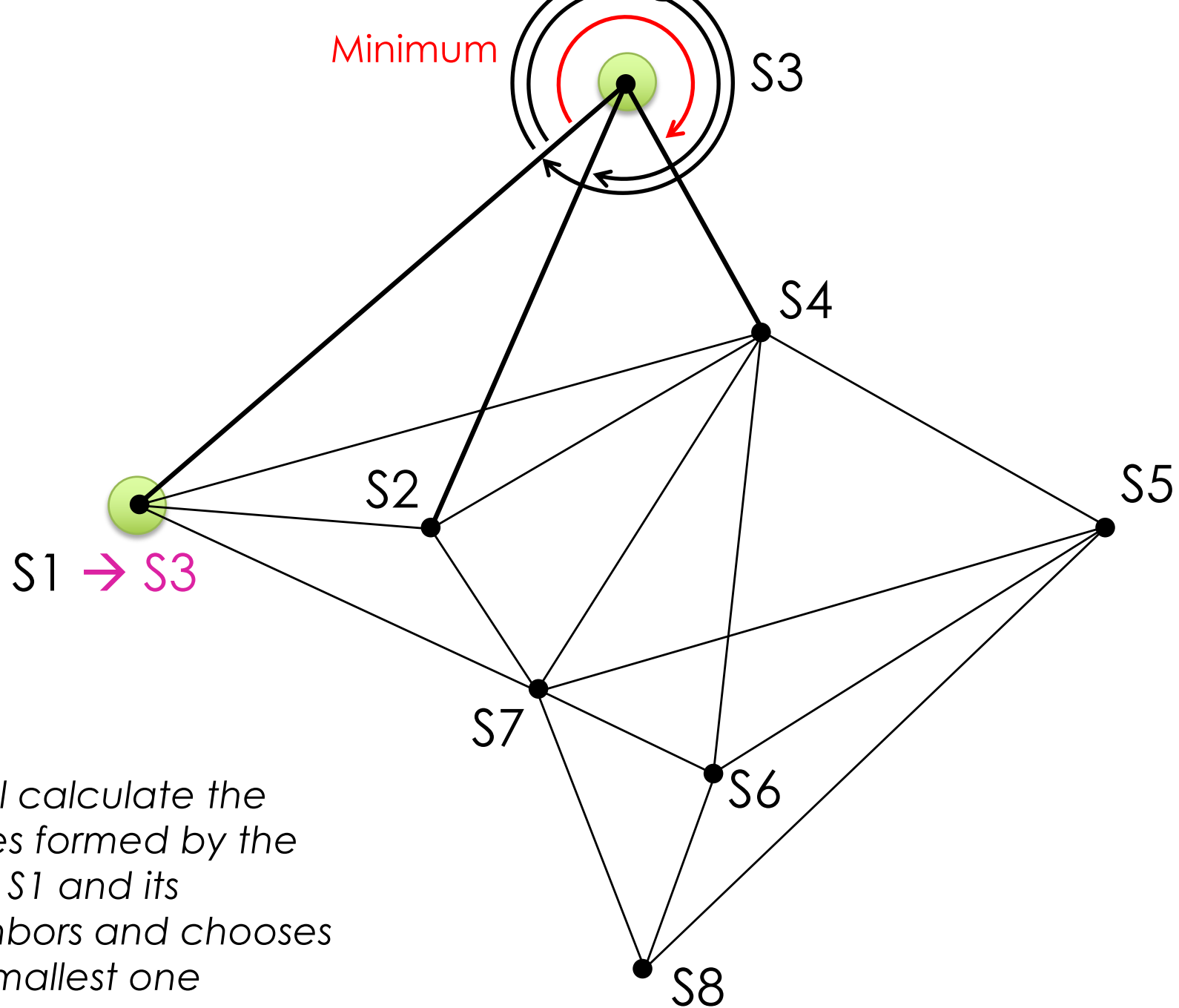
S3 send a
broadcast
message 'AC' and

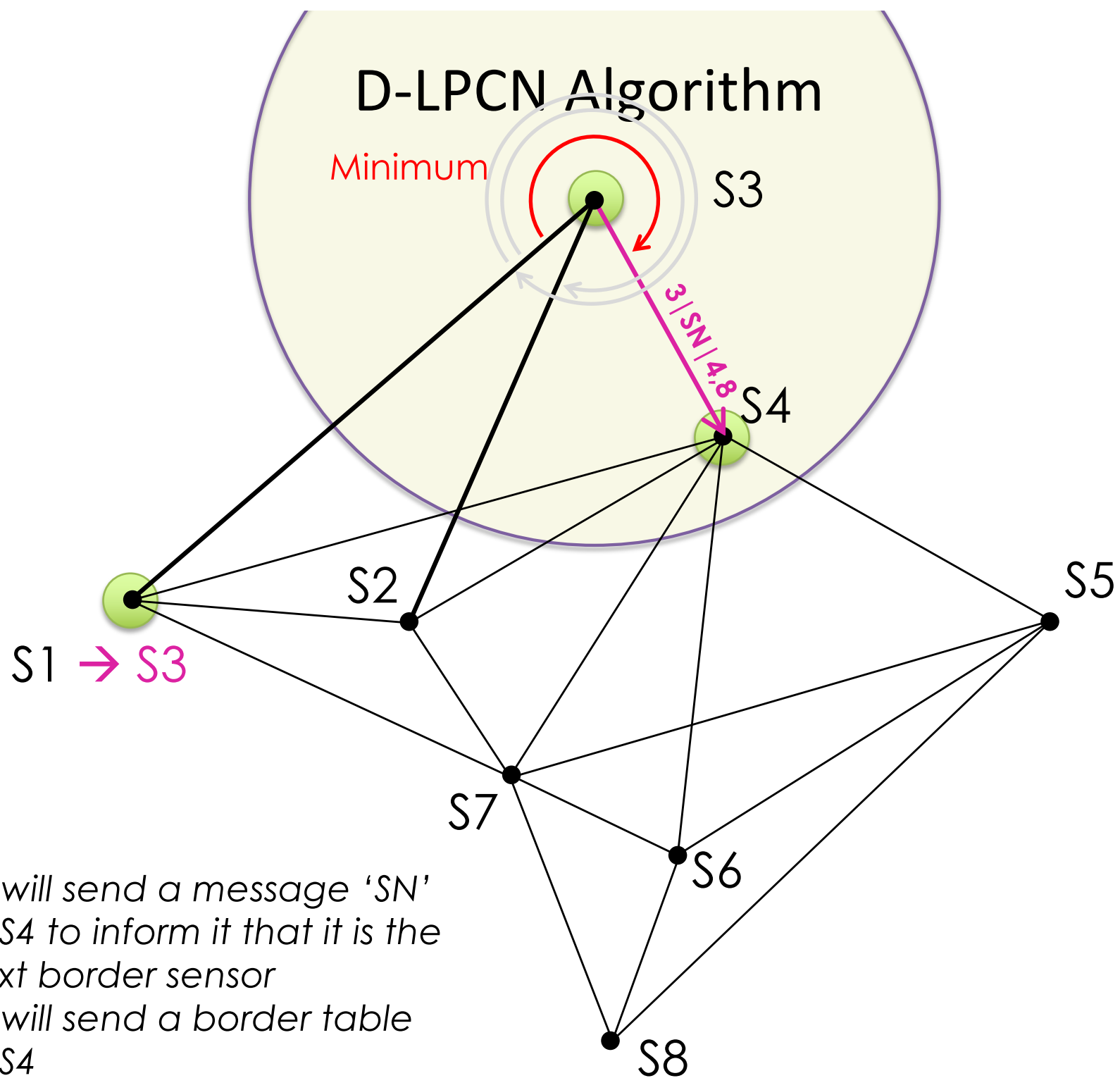
D-LPCN Algorithm



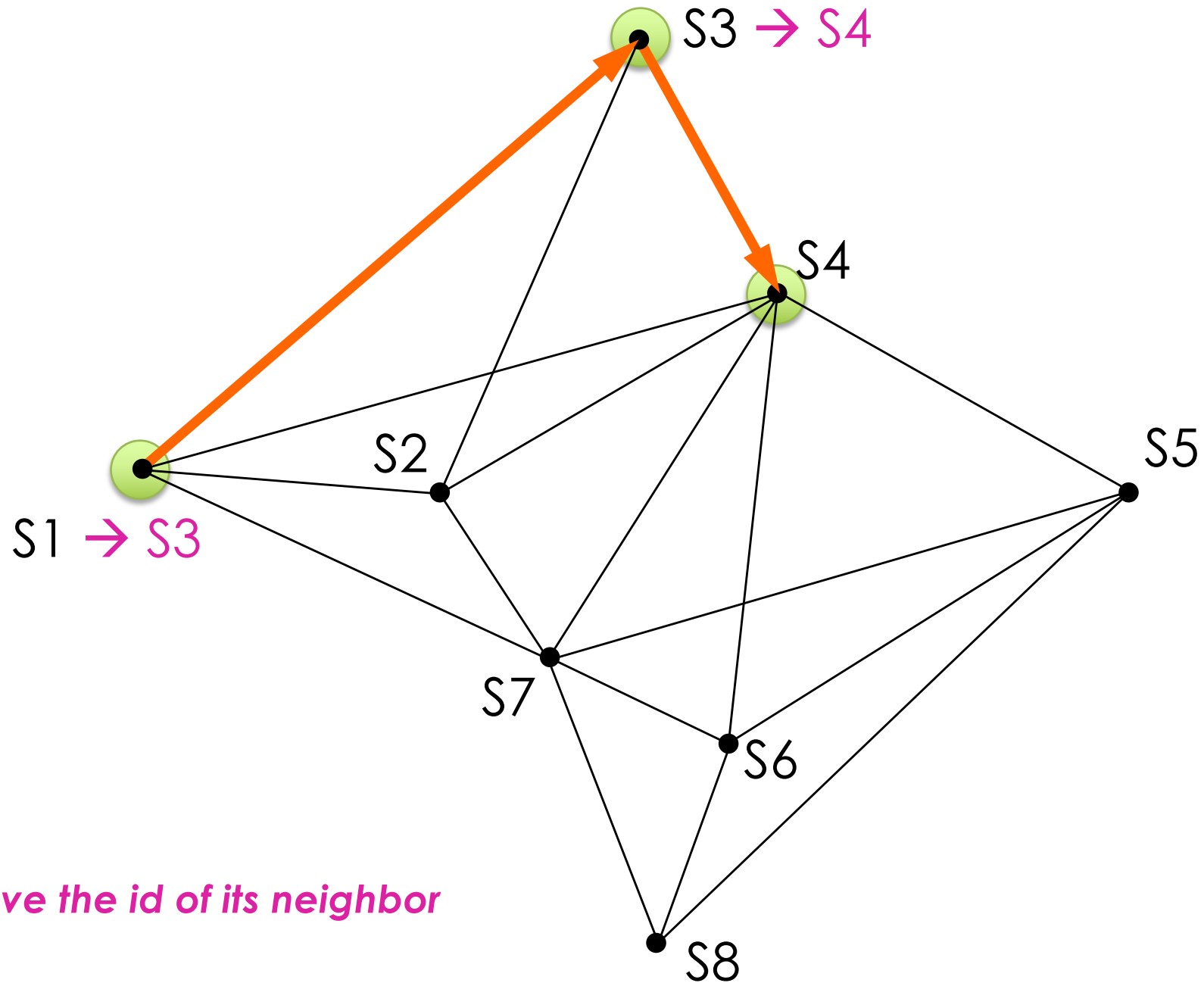
S2, and S4 receive the message 'AC' and send to S3 their coordinates, their id, and the message 'CS'

D-LPCN Algorithm

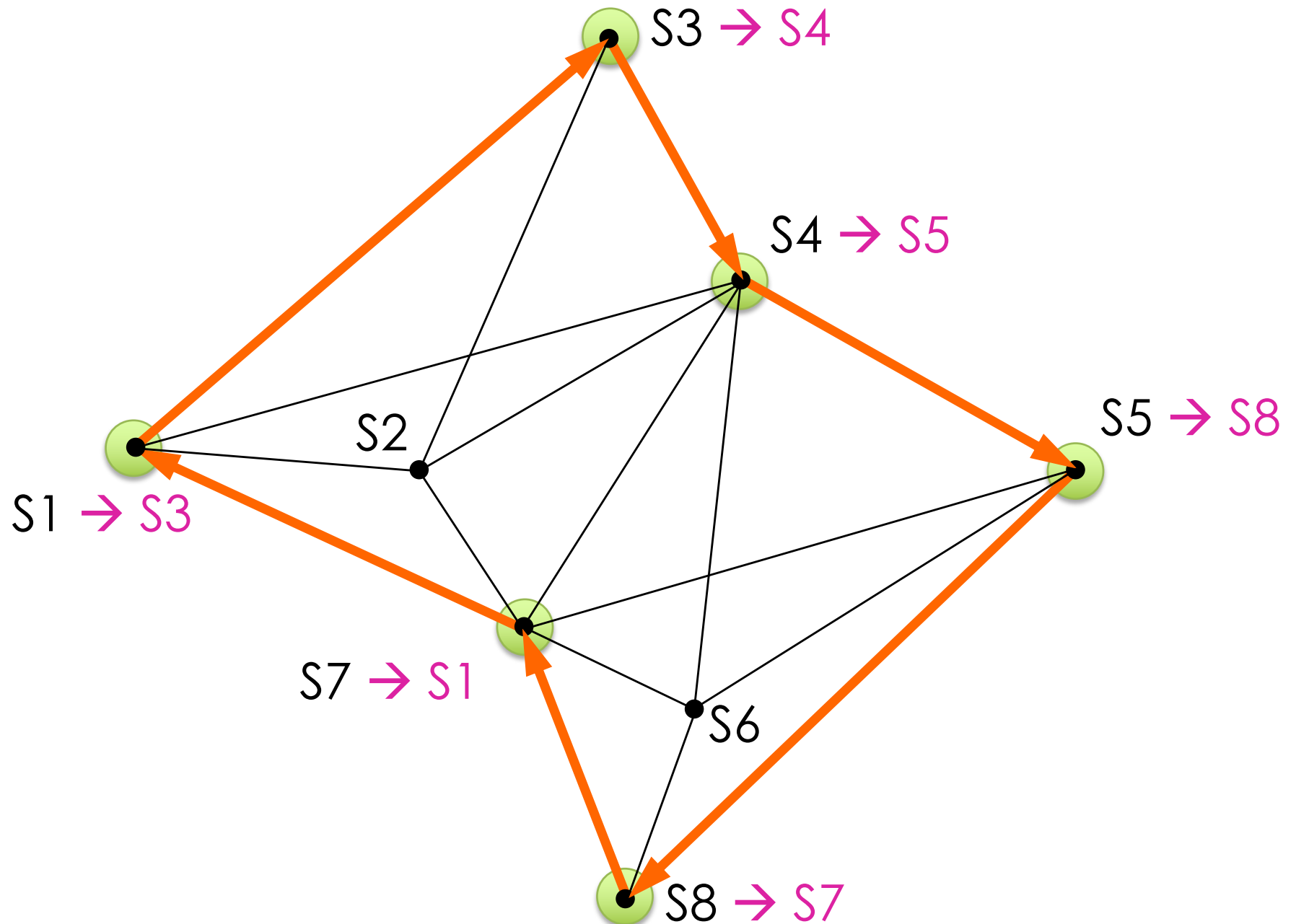




D-LPCN Algorithm



D-LPCN Algorithm



D-LPCN Algorithm: Demo (TelosB/Arduino-XBee)



3.3

How to determine the starting node?

LEADER ELECTION

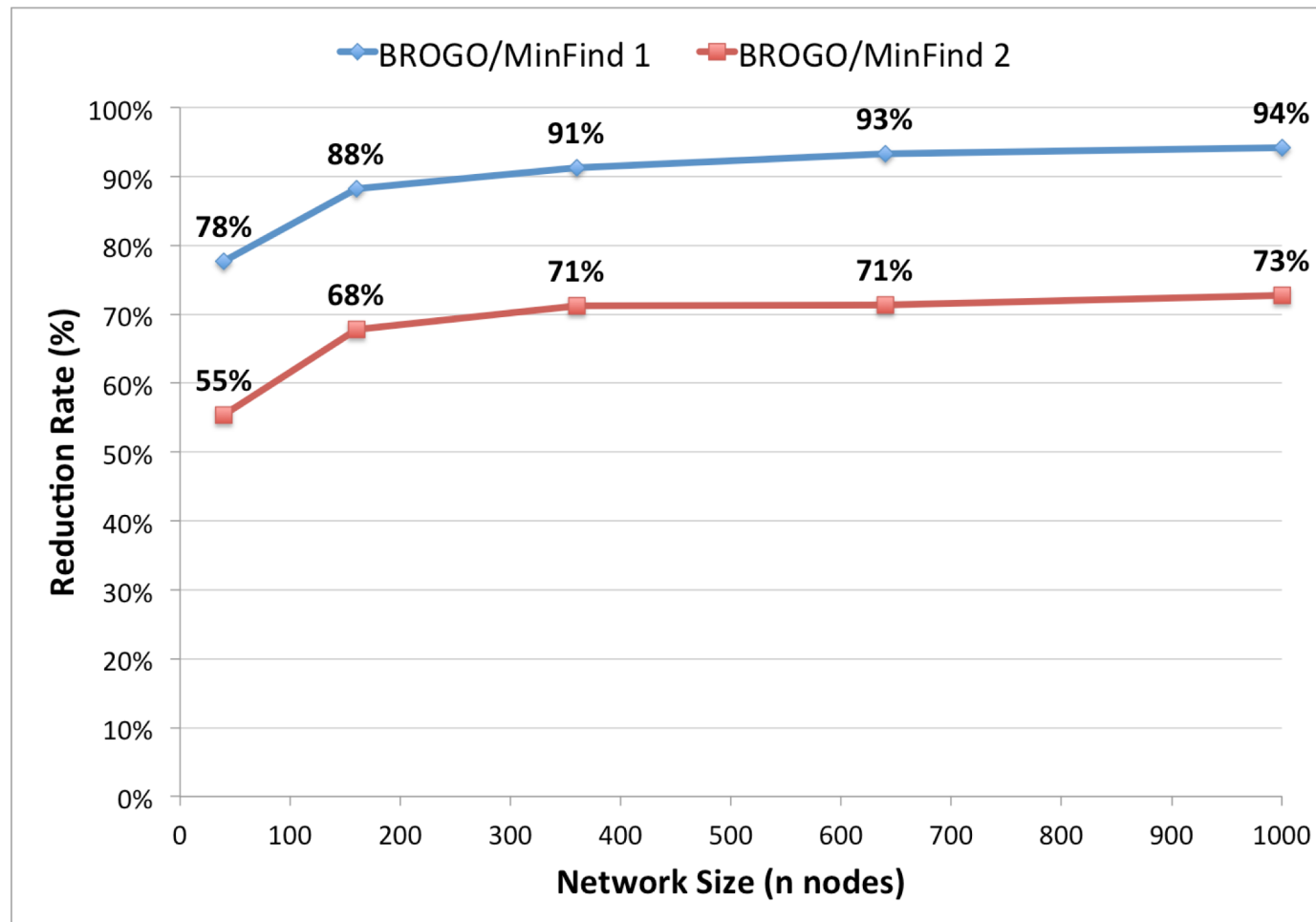
DoTRO Algorithm



Comparison with the MinFind algorithm

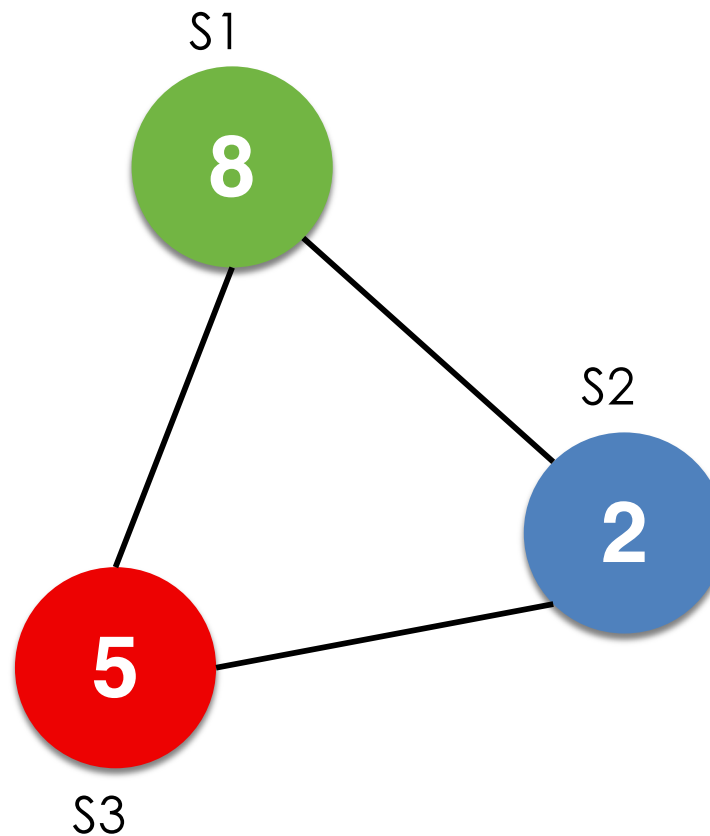
same density of 10 nodes/hm
n varies from 0 to 1000 sensor nodes

MinFind 1: xmin
MinFind 2: random



WBS: Wait-Before-Starting

Values



```
atget id id
mult v $id 1000
set once 0
loop
wait $v
read x
if(($x==\) && ($once==0))
    set once 1
    mark 1
    send A
    stop
end
if(($x==A) && ($once==0))
    set once 1
    send A
    stop
end
```

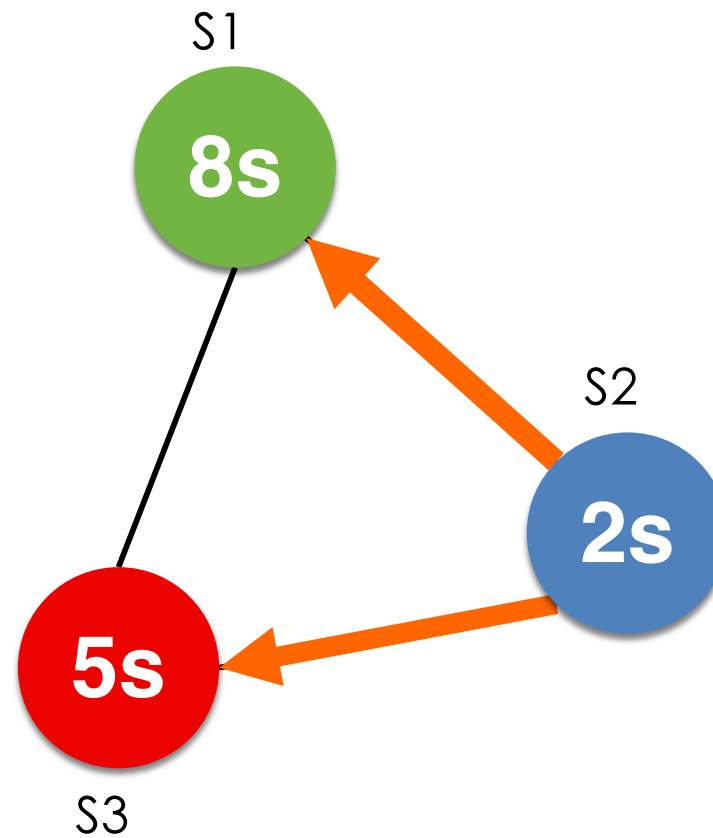
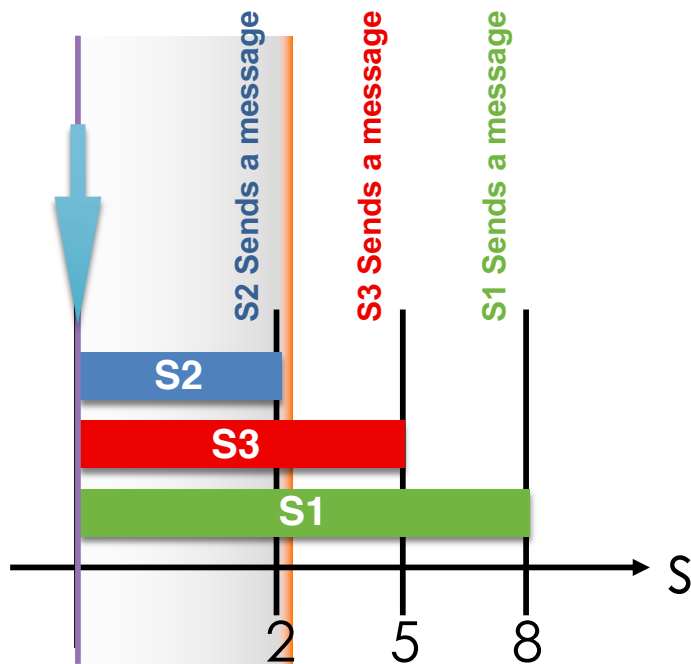

WBS: Wait-Before-Starting

Values

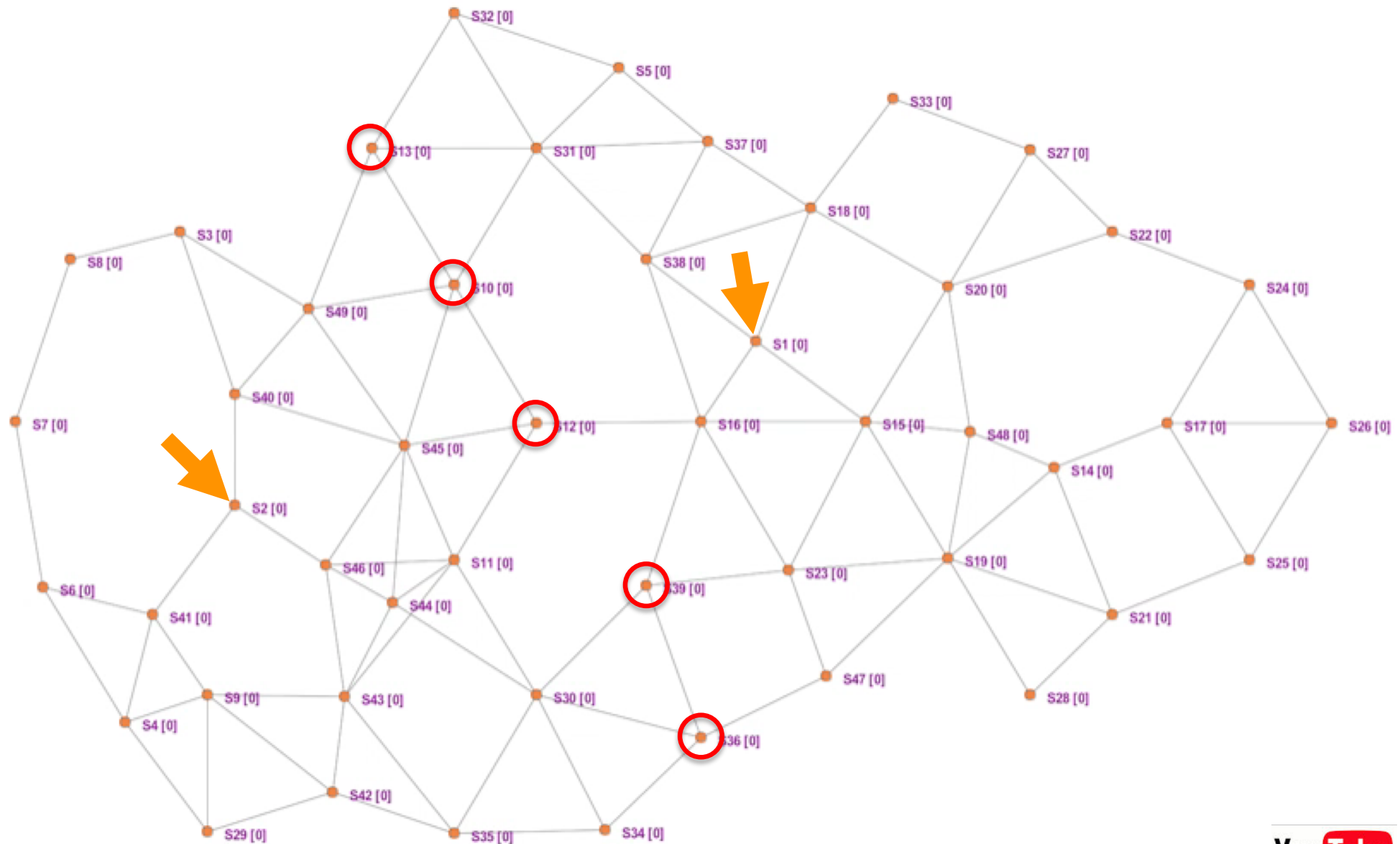
Waiting time unit = 1s

Waiting times

After 2s?



RBROGO & D-LPCN: disconnected network



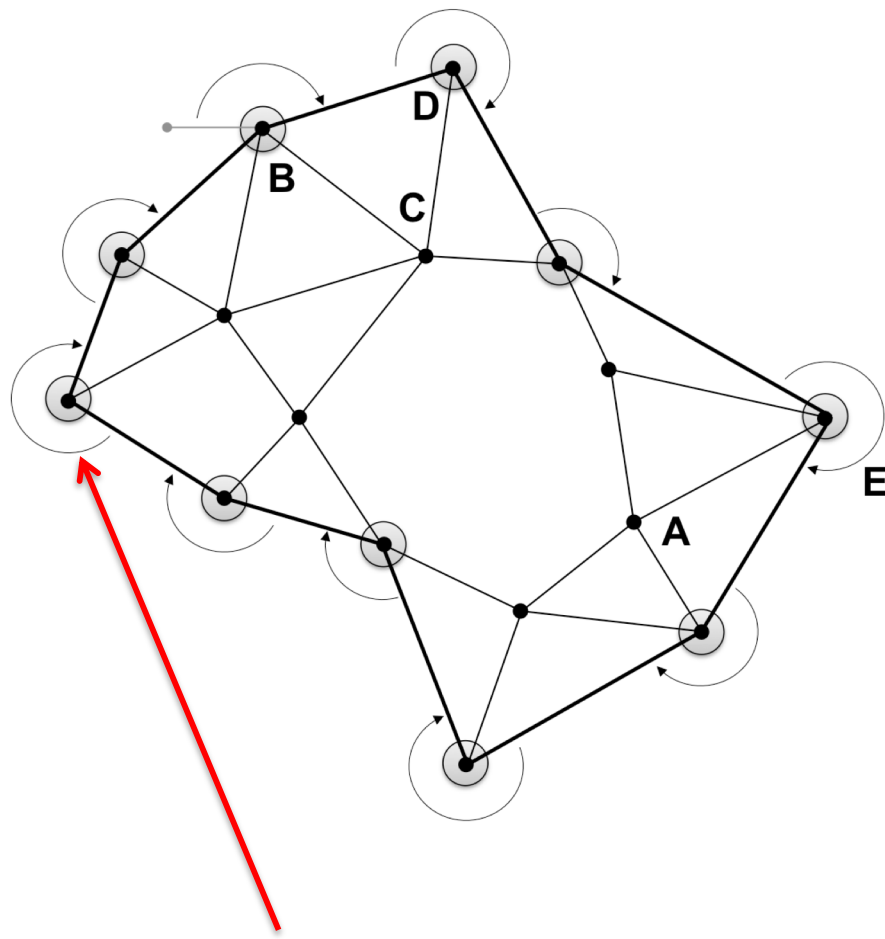
○ Faulty node

➡ Root

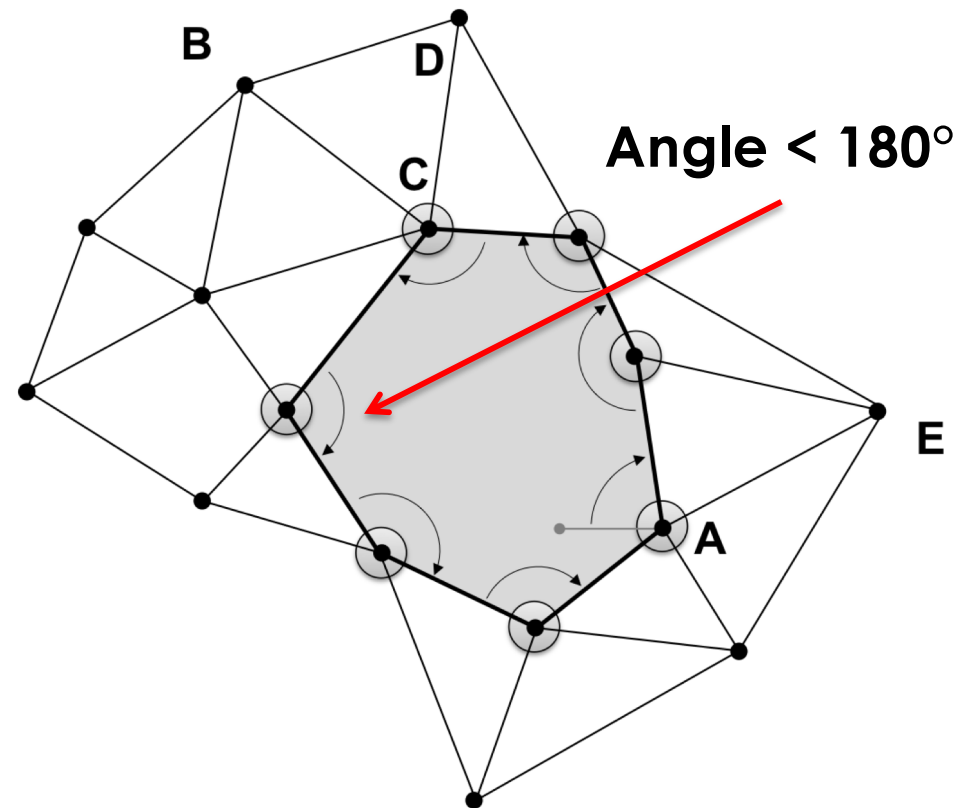
YouTube

<https://www.youtube.com/watch?v=iGStMtjeSHg>

D-LPCN Algorithm: Gap detection



Angle $> 180^\circ$

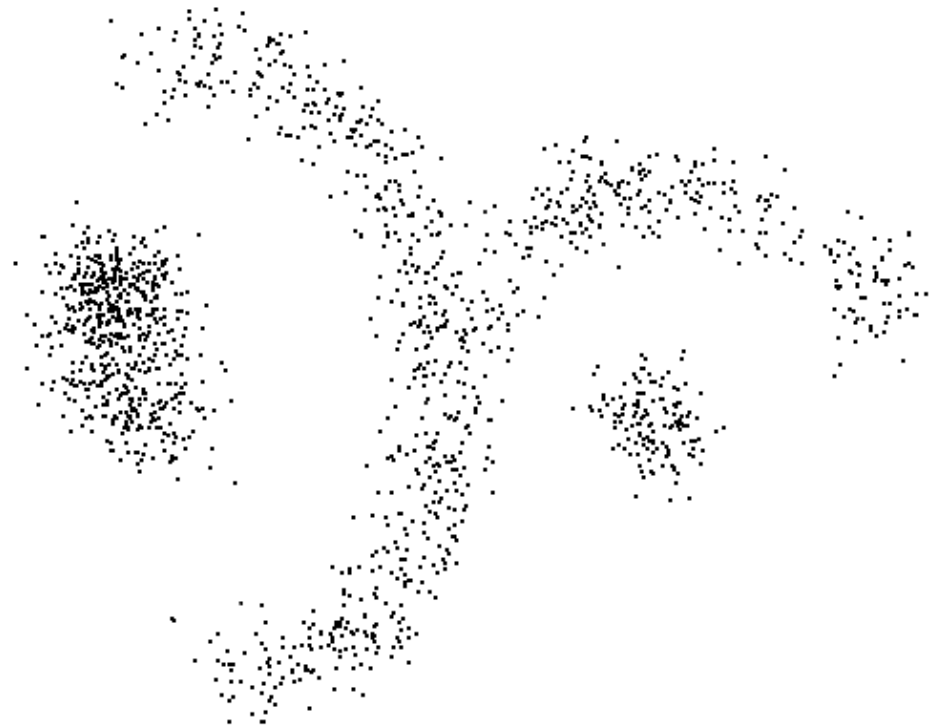


Angle $< 180^\circ$

4

QUELQUES APPLICATIONS

Big Data: Finding Clusters



Big Data: Finding Clusters

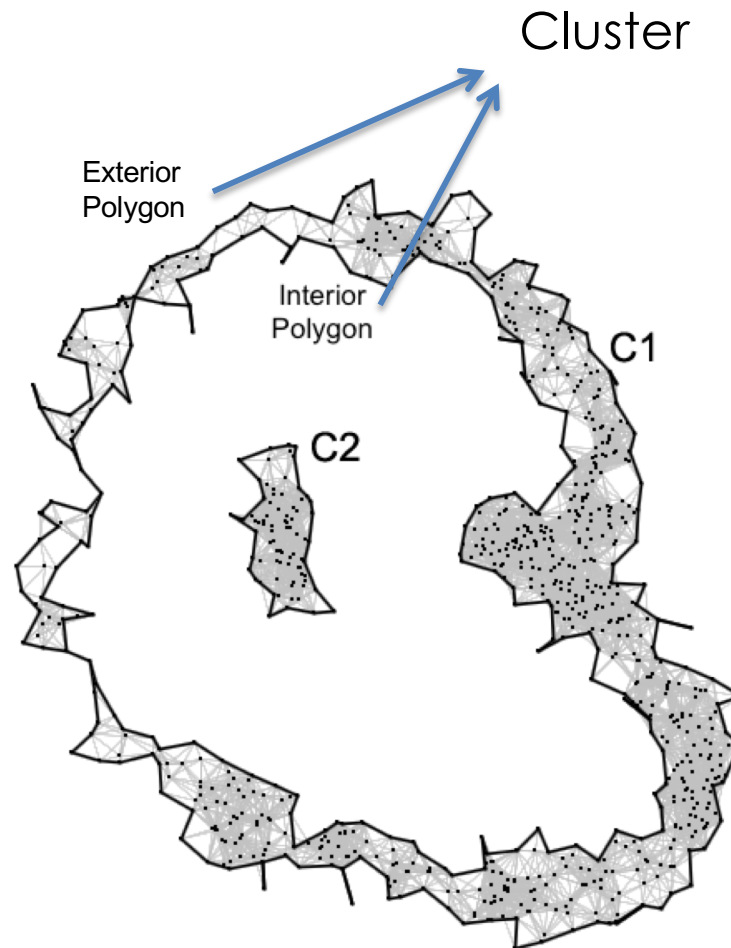


Image Processing: Countour drawing

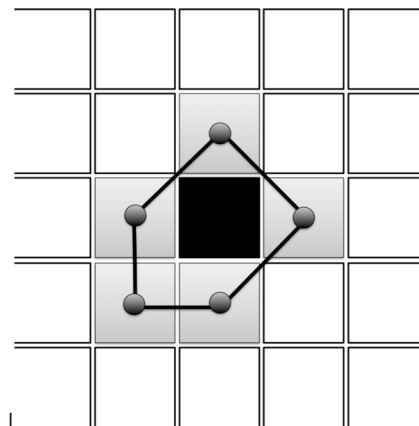
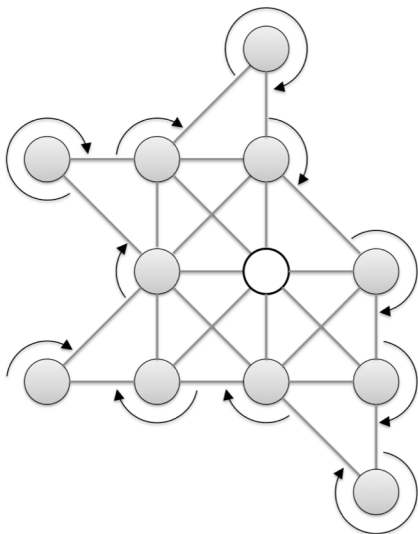
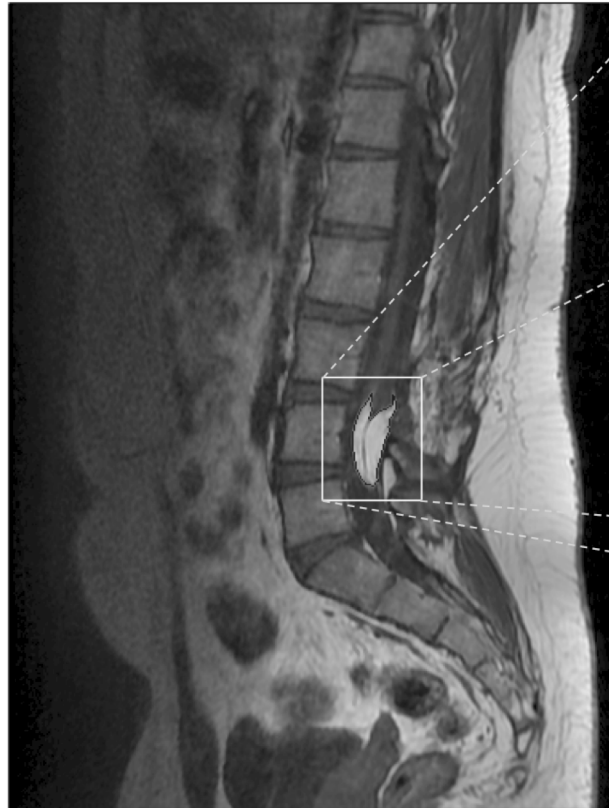
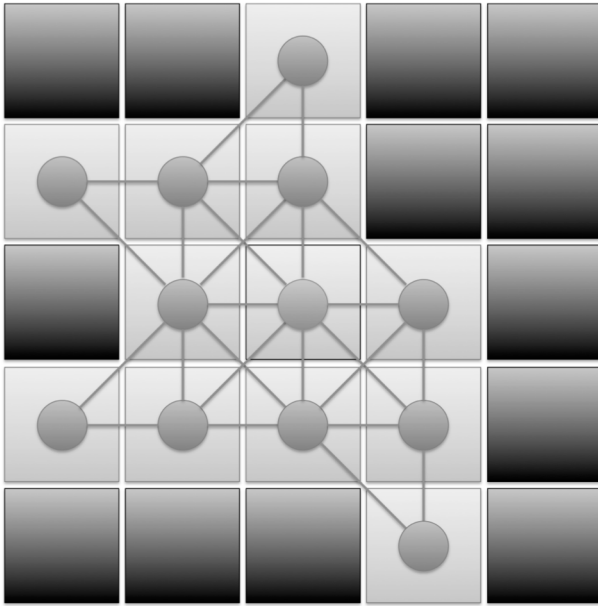


Image Processing: Countour drawing



5

CONCLUSION & FUTURE WORK

Conclusion

- We have presented:
 - A new concept of a Pseudo polygon
 - A new platform for the simulation of WSNs dedicated to Smart-cities and IoT
 - A new distributed algorithms allowing to determine the polygonal hull and the Gaps of Euclidean graphs
- We are developing other new algorithms:
 - What about you?
 - Just use CupCarbon ...

Future Work

- Questions:
 1. Can we consider that a graph is composed of a set of pseudo-polygons?
 2. Can we use pseudo-polygons to secure Data?
 3. Can we use pseudo-polygons to extract knowledge and in digital humanities?
 4. Can we ...

Boundaries and Hulls of Euclidean Graphs: From Theory to Practice



Madani Bezoui



Reinhardt Euler



CRC Press
Taylor & Francis Group





Thank You
for your attention



Ahcène Bounceur
Ahcene.Bounceur@univ-brest.fr

www.cupcarbon.com

