

First project: Finding Dory

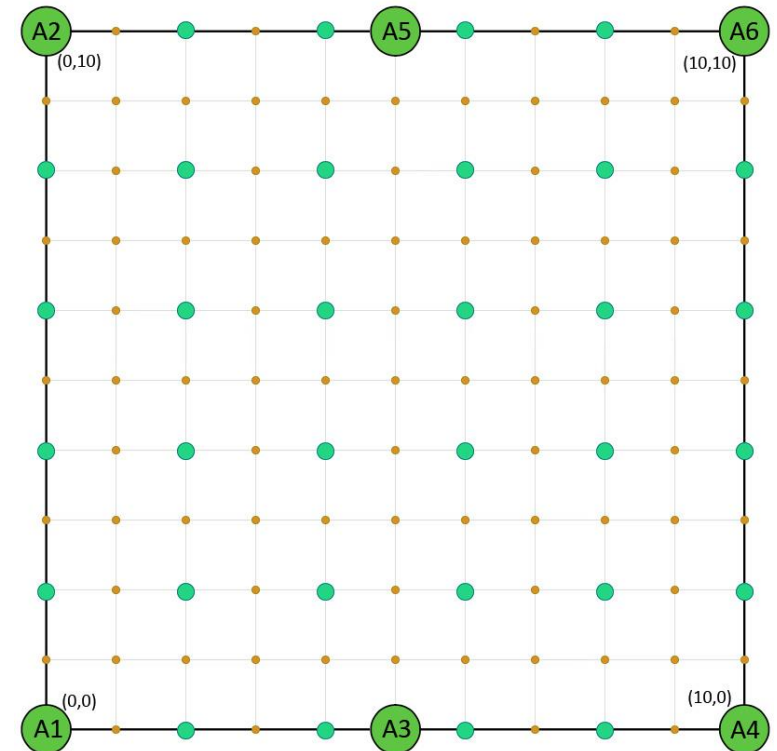
Key of the project: Fingerprint-based localization

Fingerprint localization is based on:

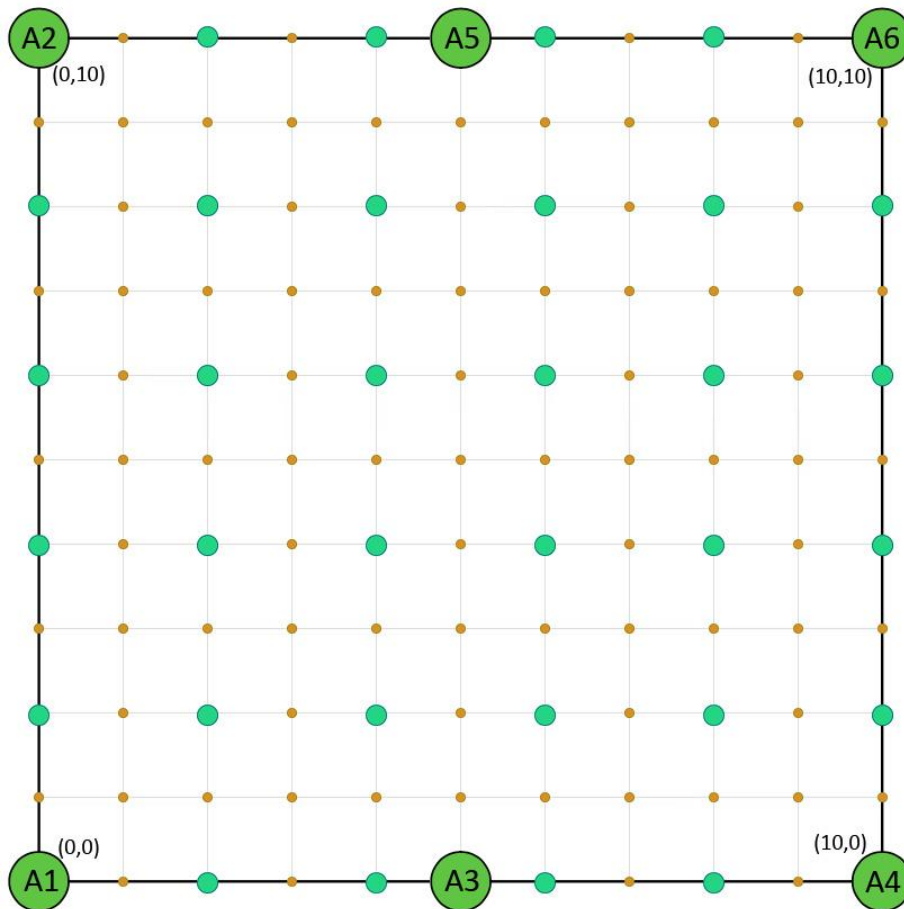
- A **set of anchors** with known positions
- A **dataset** containing the RSSI (with respect to the anchors) for each point in the grid

A device is localized comparing its RSSIs with the dataset, to estimate its position

The dataset is hidden in MQTT, CoAP and HTTP resources in a remote server



First project: Finding Dory



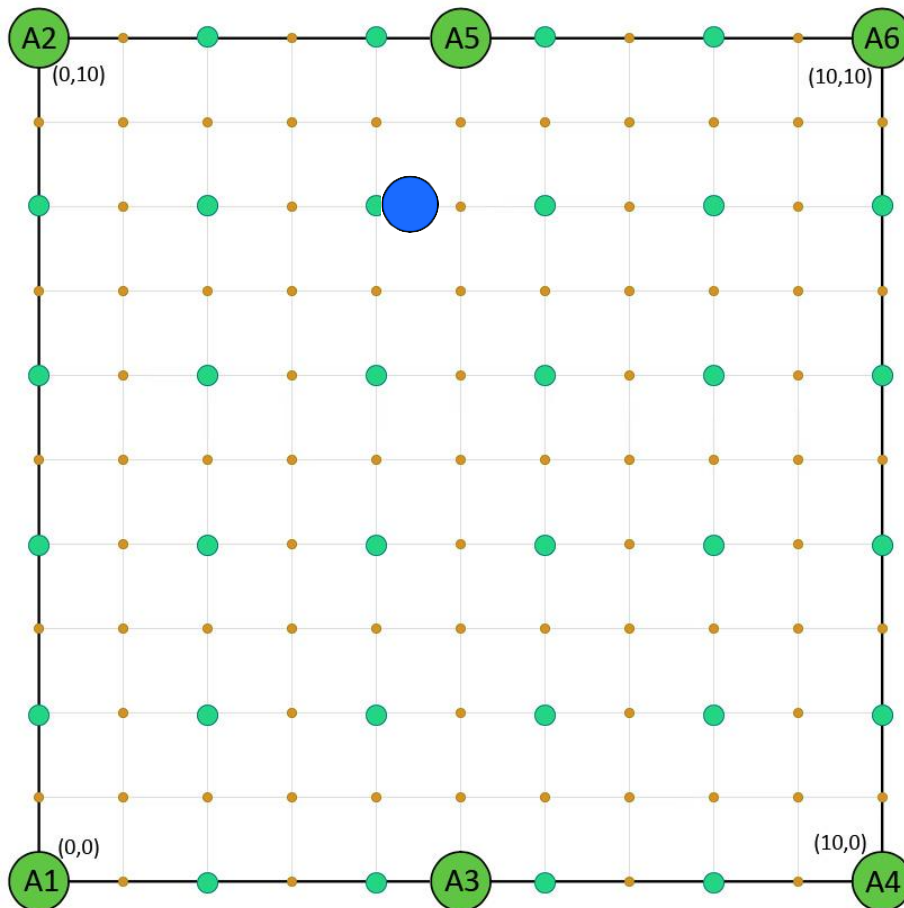
6

Fingerprint example:

(X,Y)	[A1]	[A2]	[A3]	[A4]	[A5]	[A6]
(0,0)	[-22]	[-60]	[-42]	[-60]	[-65]	[-80]
(4,8)	[-63]	[-57]	[-62]	[-64]	[-52]	[-60]

.....

First project: Finding Dory



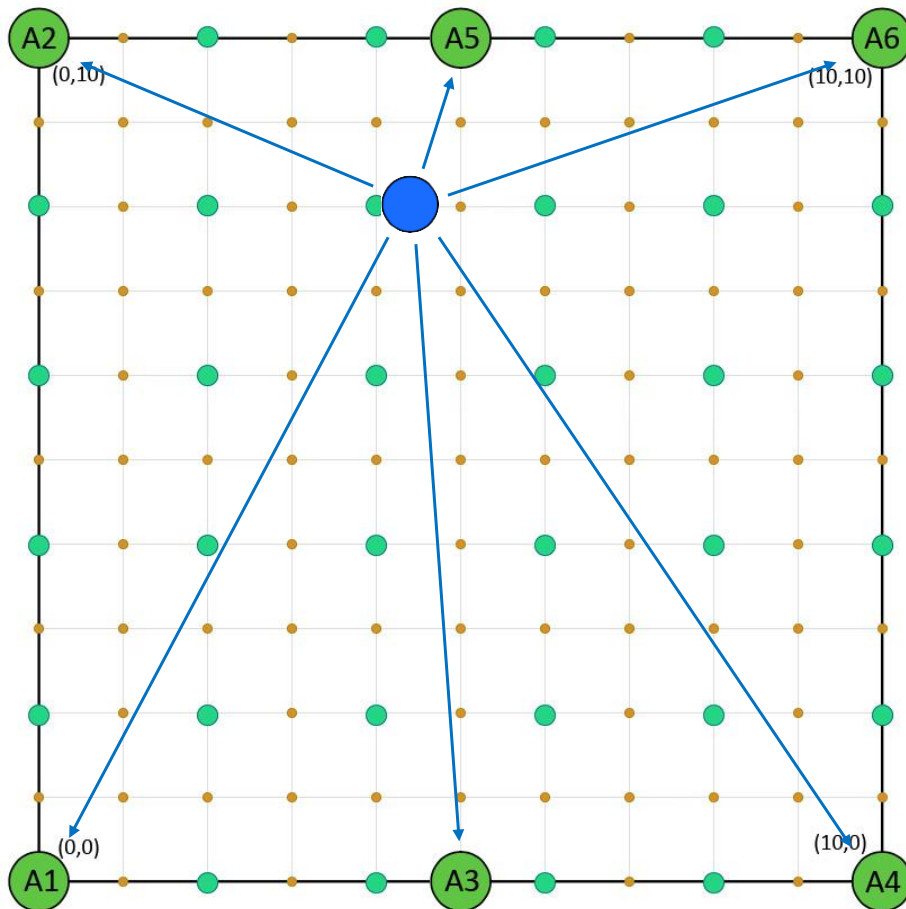
6

Fingerprint example:

(X,Y)	[A1]	[A2]	[A3]	[A4]	[A5]	[A6]
(0,0)	[-22]	[-60]	[-42]	[-60]	[-65]	[-80]
(4,8)	[-63]	[-57]	[-62]	[-64]	[-52]	[-60]

.....

First project: Finding Dory



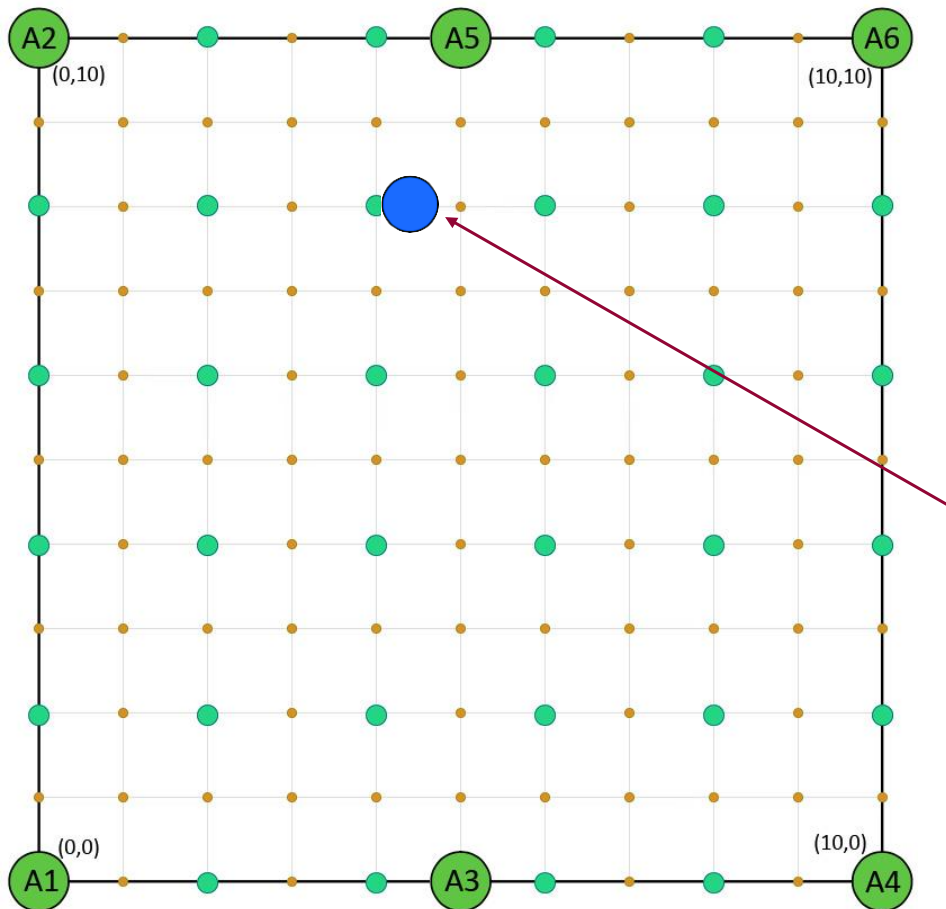
6

Fingerprint example:

(X,Y)	[A1]	[A2]	[A3]	[A4]	[A5]	[A6]
(0,0)	[-22]	[-60]	[-42]	[-60]	[-65]	[-80]
(4,8)	[-63]	[-57]	[-62]	[-64]	[-52]	[-60]

.....

First project: Finding Dory



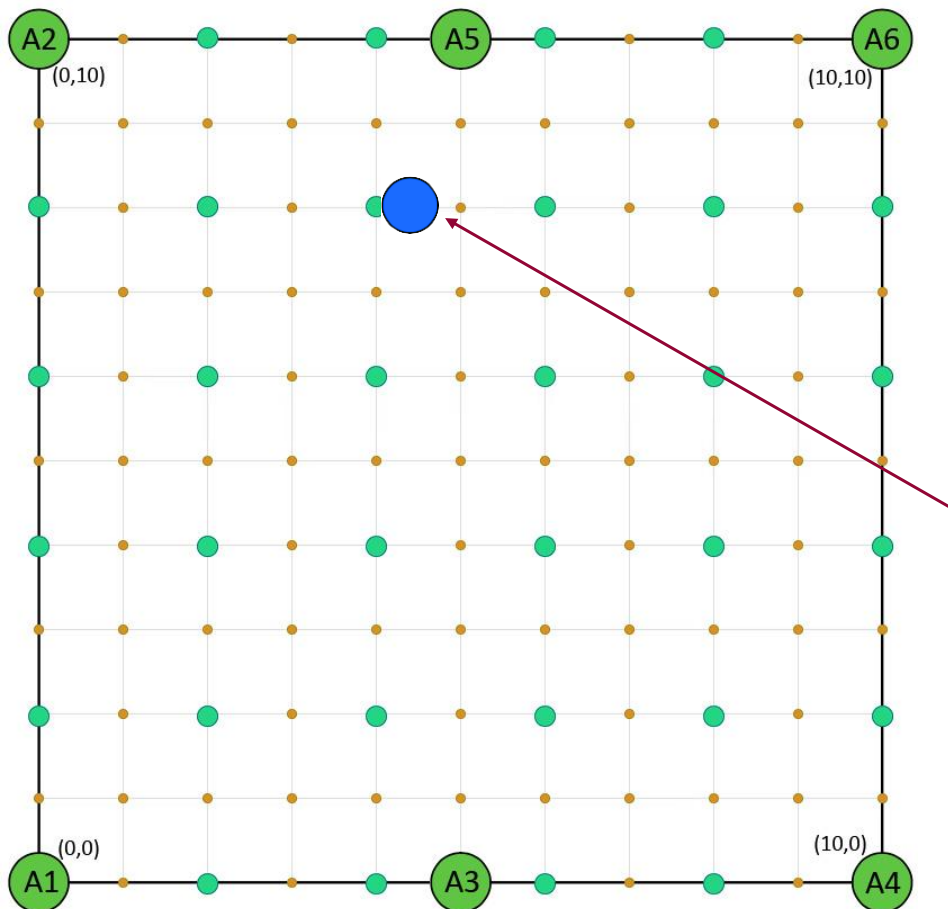
6

Fingerprint example:

(X,Y)	[A1]	[A2]	[A3]	[A4]	[A5]	[A6]
(0,0)	[-22]	[-60]	[-42]	[-60]	[-65]	[-80]
(4,8)	[-63]	[-57]	[-62]	[-64]	[-52]	[-60]
.....						

RSSI vector: [-63] [-57] [-62] [-64] [-51] [-59]

First project: Finding Dory



6

Fingerprint example:

(X,Y)	[A1]	[A2]	[A3]	[A4]	[A5]	[A6]
(0,0)	[-22]	[-60]	[-42]	[-60]	[-65]	[-80]
(4,8)	[-63]	[-57]	[-62]	[-64]	[-52]	[-60]

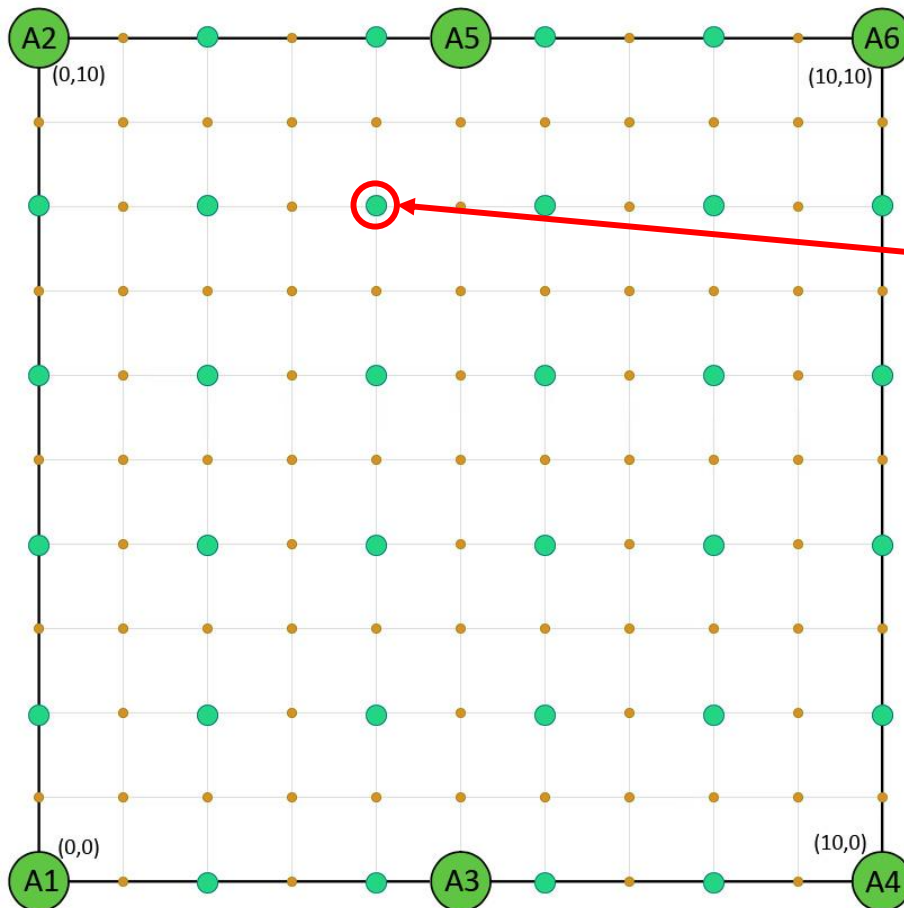
.....

RSSI vector: [-63] [-57] [-62] [-64] [-51] [-59]

By comparing the vector with the dataset entries, we get an approximated position

First project: Finding Dory

6



Fingerprint example:

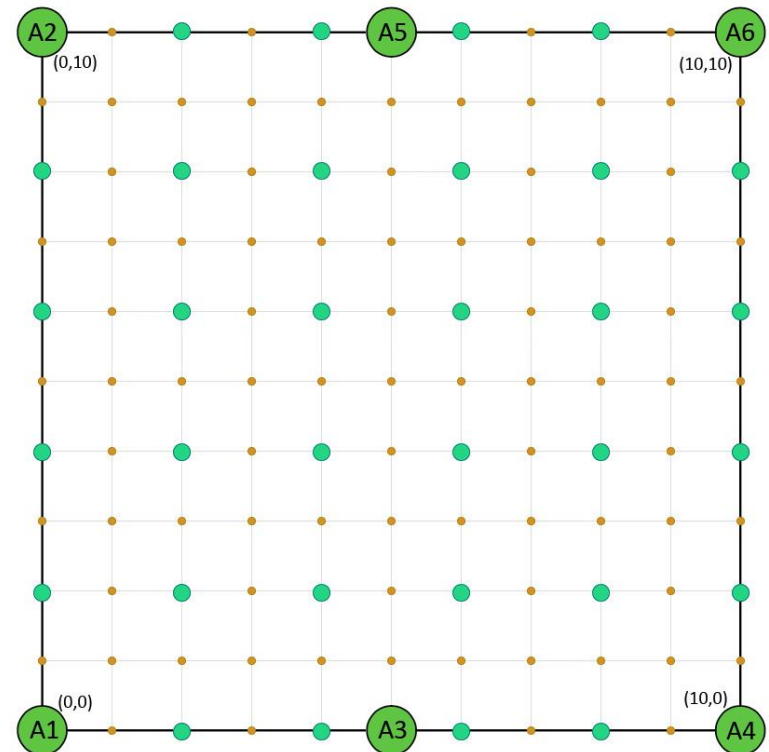
(X,Y)	[A1]	[A2]	[A3]	[A4]	[A5]	[A6]
(0,0)	[-22]	[-60]	[-42]	[-60]	[-65]	[-80]
(4,8)	[-63]	[-57]	[-62]	[-64]	[-52]	[-60]

.....

First project: Finding Dory

7

- In the project, you can find all the entries for the **even positions** (●)
- **Dory** can be situated in either an **even or odd position** in the grid (● or ●)



First project: Finding Dory

8

Project Goals:

1. Find the entries, hidden in **HTTP** and **CoAP** Resources or **MQTT** topics
2. Reconstruct the dataset by cleaning/filtering the found entries
3. Find Dory's RSSI vector, hidden with the dataset entries
4. Estimate the position of Dory from its RSSI

Server address: TO BE DEFINED (will be given on 19th of May)

Ports: HTTP **80**
MQTT **1883**
CoAP **5683**

Hints:

- Use CoAP **discovery** request to «discover» existing resources paths
Get request to «server-address»/.well-known/core
Warning: «hidden» resources do not appear in the discovery response!
- Subscribe in MQTT using **wildcards** to include more topics at once
- Remember the CoAP Observe feature for observable resources:
(GET + Observe=0 acts as an MQTT subscribe request)



First project: Finding Dory

9

What to deliver?

- The obtained fingerprint dataset
- Dory's RSSI
- Dory's estimated position
- A report containing comments of what has been done, decisions taken, considerations etc...

To deliver the project send an email to fabio.palmese@polimi.it

Questions ?