Land & Localize: An Infrastructure-free and Scalable Nano-Drones Swarm with UWB-based Localization

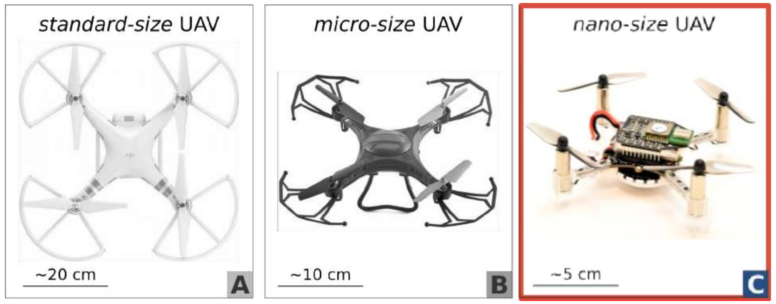
Mahyar Pourjabar[1], Ahmed AlKatheeri[2], Manuele Rusci[3], Agata Barcis[2]
Vlad Niculescu[4], Eliseo Ferrante[2], Daniele Palossi[5,4], Luca Benini[1,4]

[1] DEI - University of Bologna, Italy
[2] Technology Innovation Institute, UAE
[3] KU Leuven, Belgium
[4] IIS - ETH Zurich, Switzerland
[5] IDSIA, USI and SUPSI, Switzerland

Why Nano-Drone Swarm?

What are Nano-drones?

- Small form factor (~10cm)
- Limited payload (~15g)
- Limited computing power budget (<100mW)

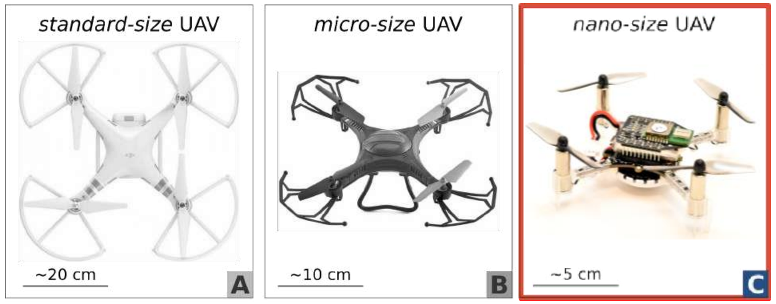


UAV	Standard-sized	Nano-sized
Size [⌀, weight]	~50cm / ~ few Kg	~10cm / ~50g
Tot. Power	~ 100 W	~ 5W
Processing device	High-end CPU	Low-power MCU

Why Nano-Drone Swarm?

What are Nano-drones?

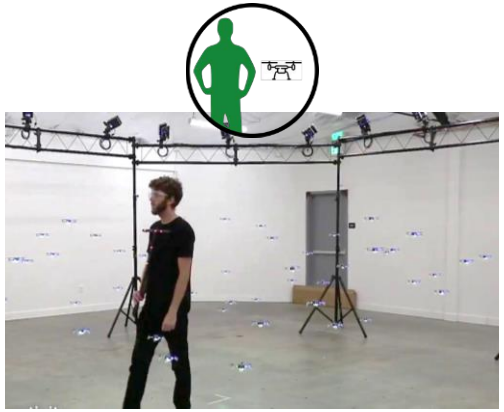
- Small form factor (~10cm)
- Limited payload (~15g)
- Limited computing power budget (<100mW)



UAV	Standard-sized	Nano-sized
Size [⌀, weight]	~50cm / ~ few Kg	~10cm / ~50g
Tot. Power	~ 100 W	~ 5W
Processing device	High-end CPU	Low-power MCU

Why Nano-Drone Swarm?

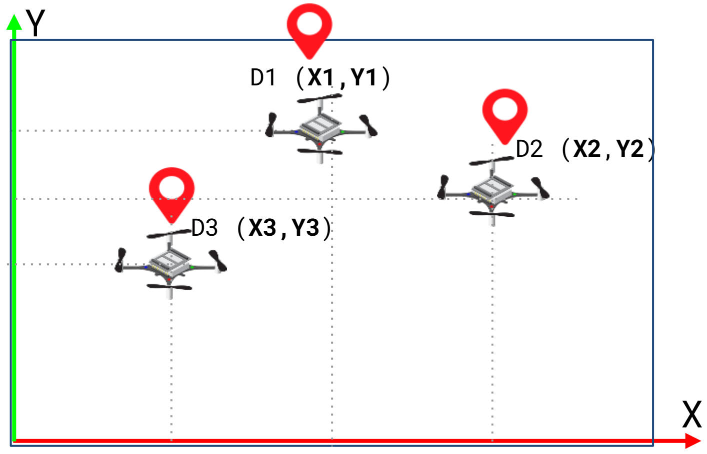
- Small form factor enabling indoor applications
- Ensure safe Human-Robot Interaction (HRI)
- Reduced costs



Localizing Nano-Drone Swarm

What is Localization?

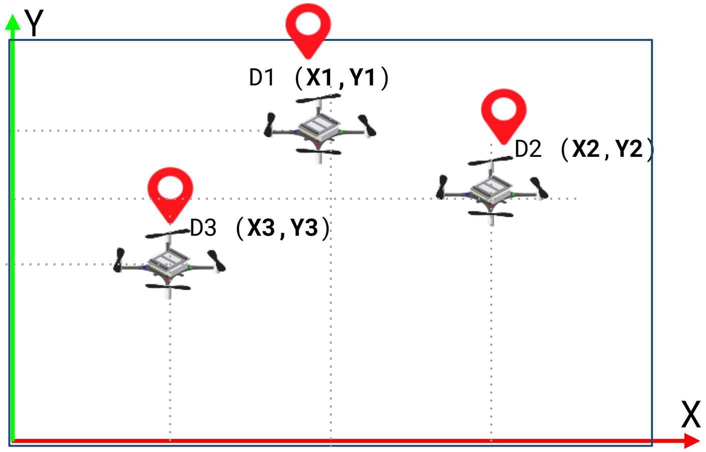
- Localization is the ability of the Swarm agents to identify **their positions**



Localizing Nano-Drone Swarm

What is Localization?

- Localization is the ability of the Swarm agents to identify **their positions**



Why Localizing a Nano-Drone Swarm?

- A Wide range of **indoor applications** can benefit from the Nano-Drone swarm
- **Localization is a key** for any Nano-drone Swarm applications

Indoor Inspection*

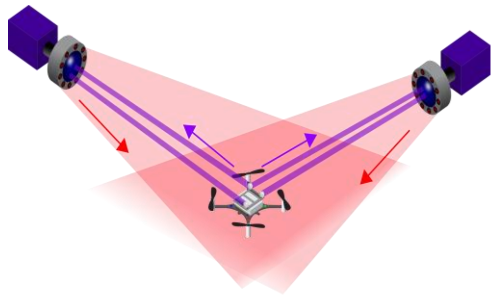
Flight Formation

Research

[*] Sniffy Bug: A Fully Autonomous Swarm of Gas-Seeking Nano Quadcopters in Cluttered Environments.” IROS 2021

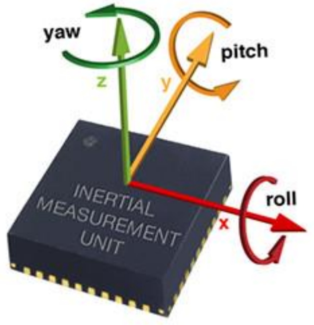
Common indoor positioning approaches

Motion Tracking Systems



- ✔ *High resolution*
- ✔ *Light computation*
- ✔ *Low power*
- ✘ *Costly infrastructure*

IMUs

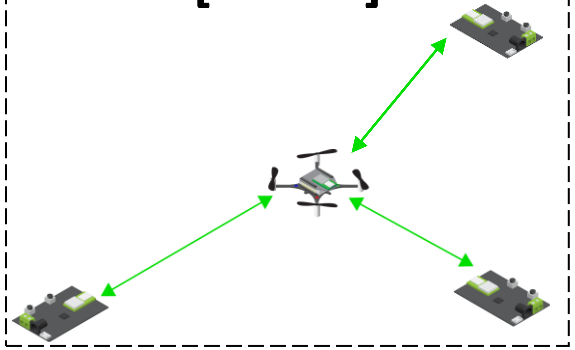


- ✘ Low resolution
- ✔ Light computation
- ✔ Low power
- ✔ Low cost

Visual Odometry

- ✔ High resolution
- ✘ *Heavy computation*
- ✘ *High power*
- ✘ *medium cost*

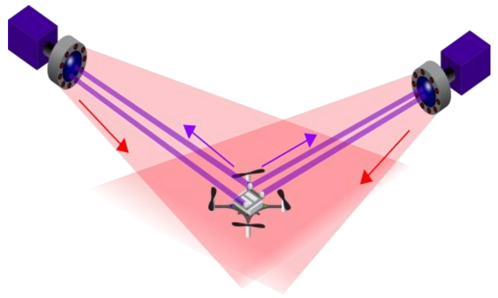
Ultra Wideband [UWB]



- ✔ *cm-level resolution*
- ✔ *Light computation*
- ✔ Low power
- ✔ Low cost

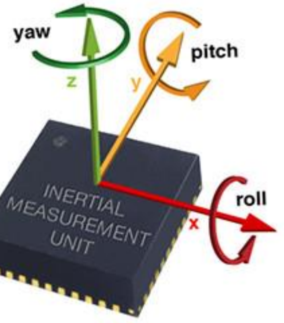
Common indoor positioning approaches

Motion Tracking Systems



- ☑ *High resolution*
- ☑ *Light computation*
- ☑ *Low power*
- ☒ *Costly infrastructure*

IMUs

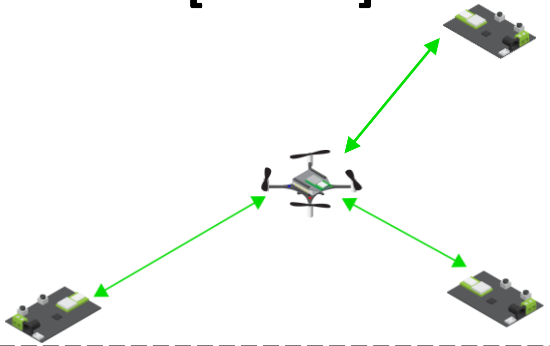


- ☒ Low resolution
- ☑ Light computation
- ☑ Low power
- ☑ Low cost

Visual Odometry

- ☑ High resolution
- ☒ *Heavy computation*
- ☒ *High power*
- ☒ *medium cost*

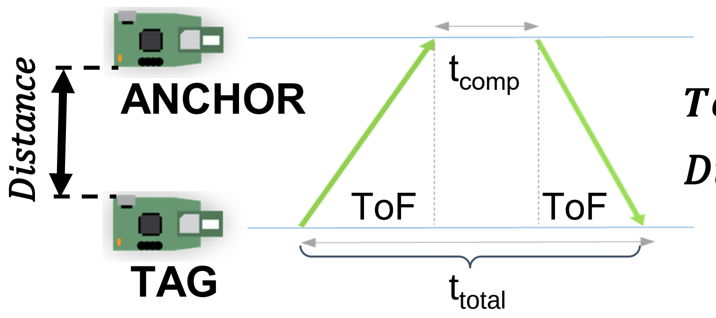
Ultra Wideband [UWB]



- ☑ *cm-level resolution*
- ☑ *Light computation*
- ☑ Low power
- ☑ Low cost

UWB-Localization & challenges

UWB Ranging

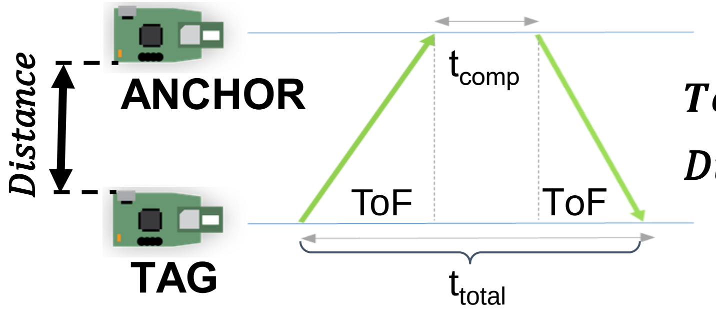


$$\boldsymbol{ToF} = (\boldsymbol{t}\_total - \boldsymbol{t}\_comp)/2$$

$$\boldsymbol{Distance} = ToF \times speed\_of\_light$$

UWB-Localization & challenges

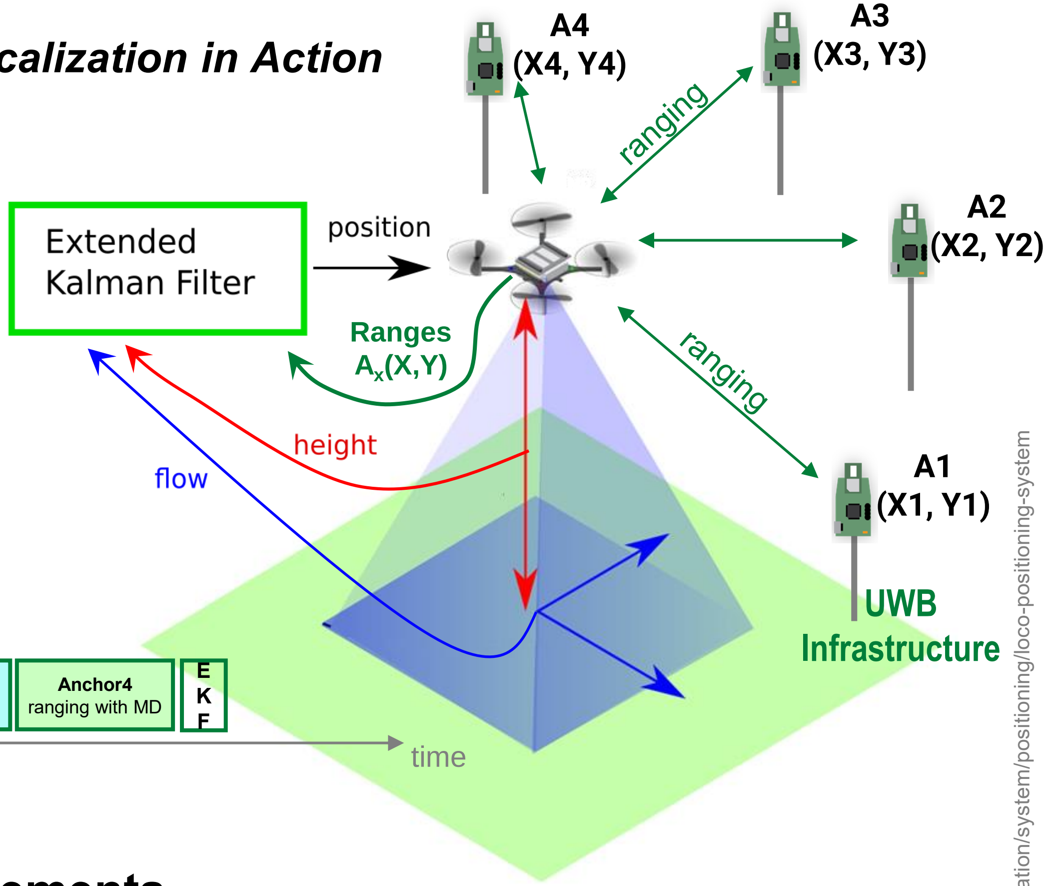
UWB Ranging

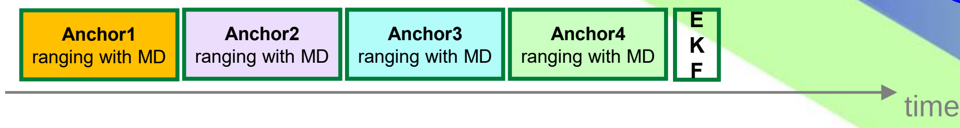


$$\boldsymbol{ToF} = (\boldsymbol{t\_total} - \boldsymbol{t\_comp})\ /2$$

$$\boldsymbol{Distance} = ToF \times speed\_of\_light$$

UWB Localization in Action





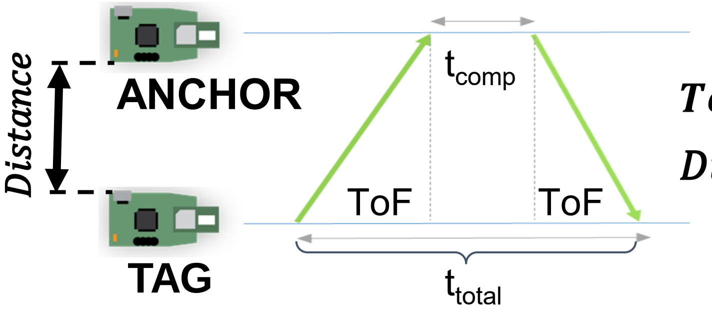
Requirements

1. **Installation** of Anchor modules infrastructure
2. Anchors' positions must be ***known***

www.bitcraze.io/documentation/system/positioning/loco-positioning-system

UWB-Localization & challenges

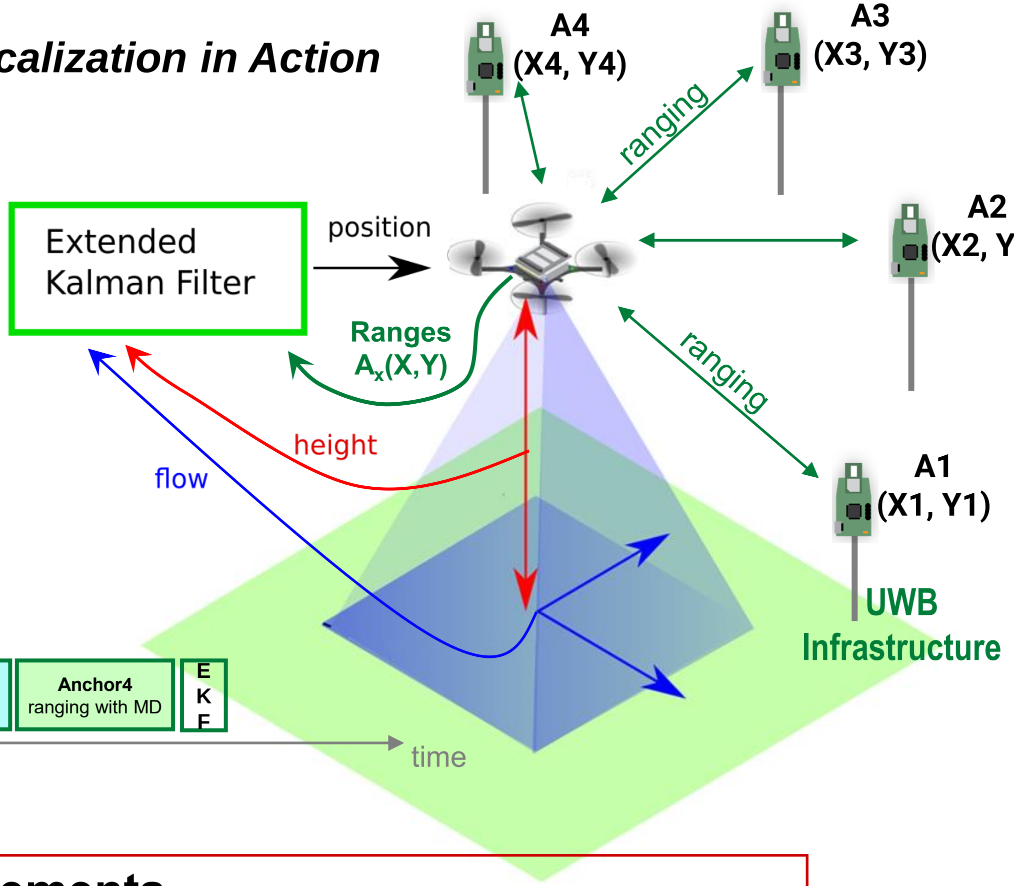
UWB Ranging

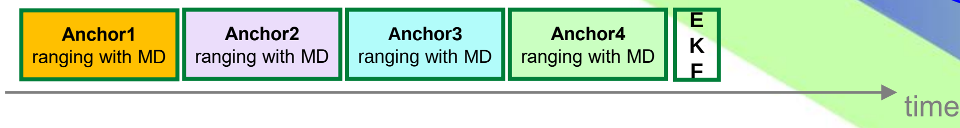


$$\boldsymbol{ToF} = (\boldsymbol{t}\_total - \boldsymbol{t}\_comp)\ /2$$

$$\boldsymbol{Distance} = ToF \times speed\_of\_light$$

UWB Localization in Action





Limit usefulness due to need for in-advanced installation of infrastructure

Static UWB infrastructure imposes a major rigidity!

Requirements

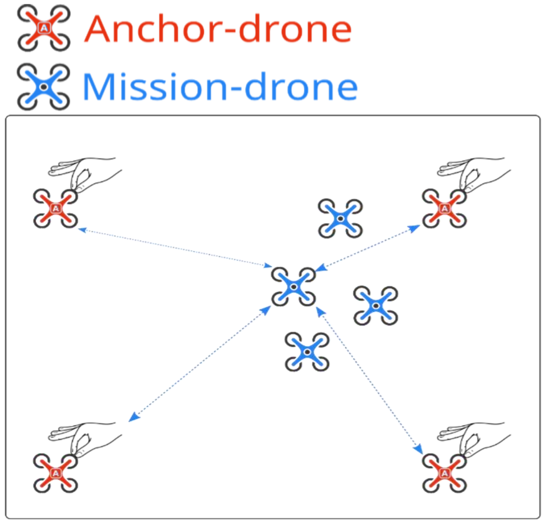
1. **Installation** of Anchor modules infrastructure
2. Anchors' positions must be ***known***

www.bitcraze.io/documentation/system/positioning/loco-positioning-system

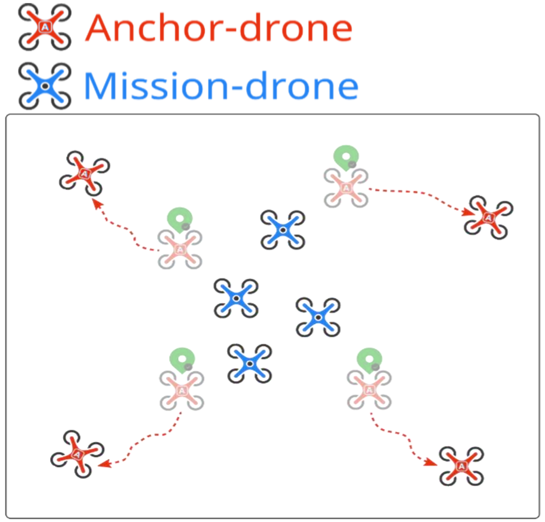
Contribution

Infrastructure-less UWB localization systems

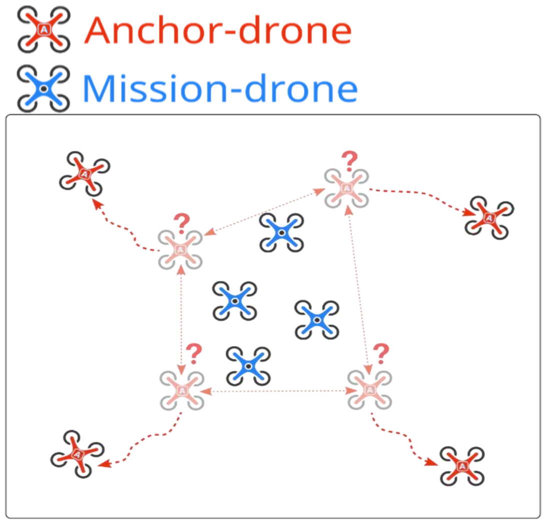
1. Exploiting **Drones as Dynamic Anchors** to provide anchor deployment at run-time
2. A **Self-Localization system** to compute anchor drones' initial position
3. An open-source **UWB Software Library (USL)** enabling fast prototyping of UWB localization



Ideal Landing



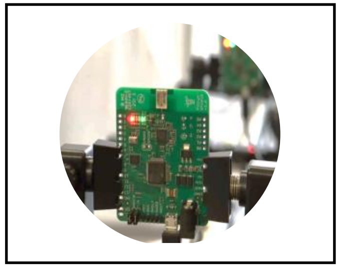
Dynamic Anchors



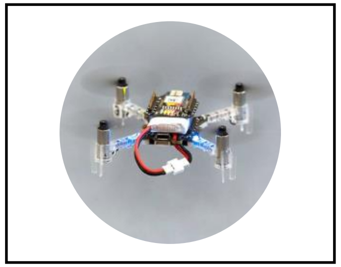
Self-localize

More Flexible Systems

Anchor Infrastructure

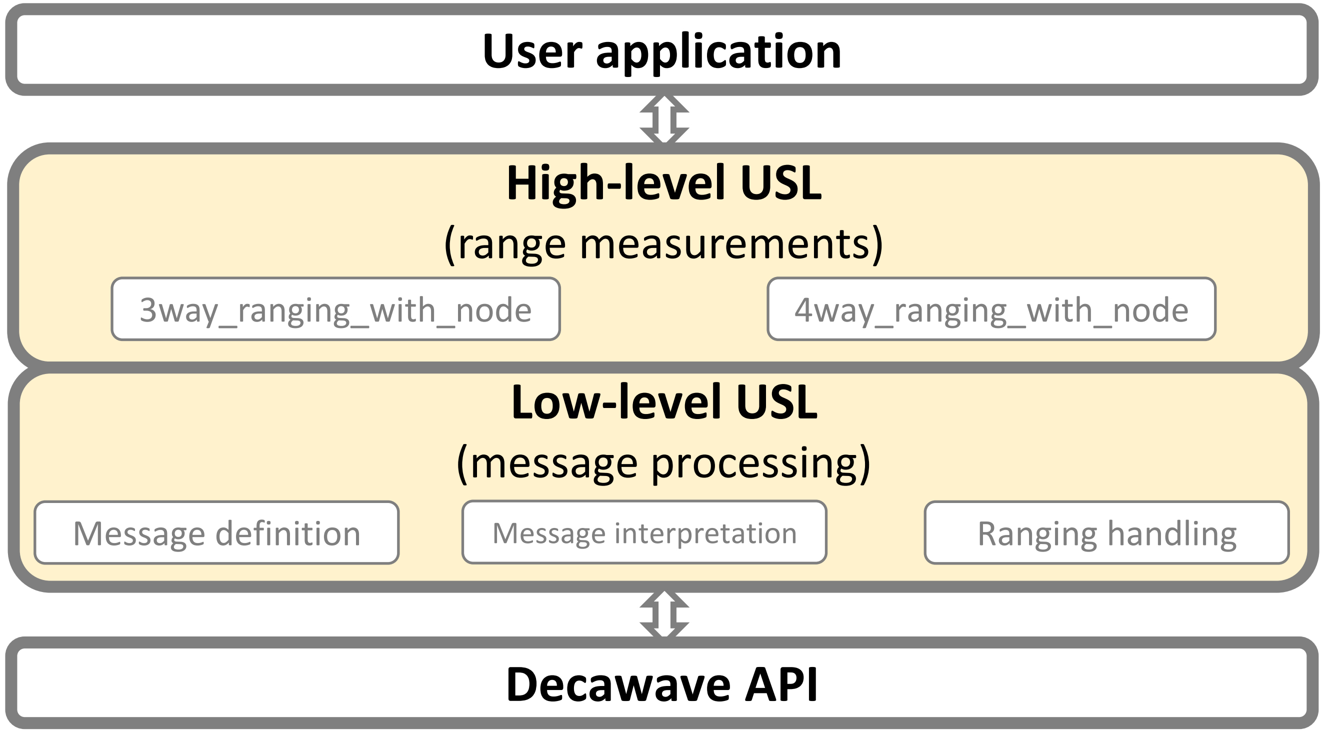


Dynamic Anchor



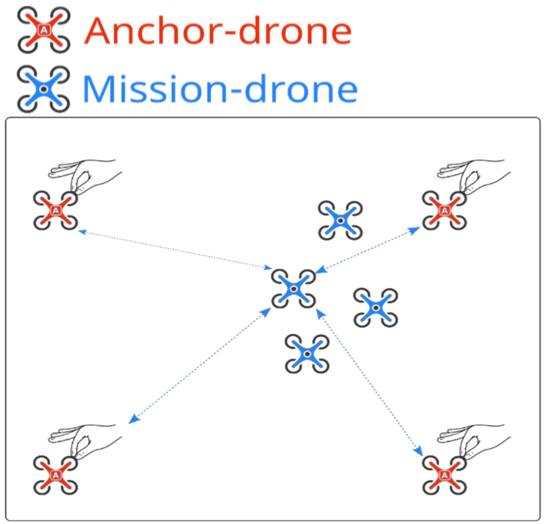
UWB Software library (USL)

- *An open-source Library*
- *High-level API*
- *Enable Fast prototyping*
- *Minimal interaction with HW-level complexities*

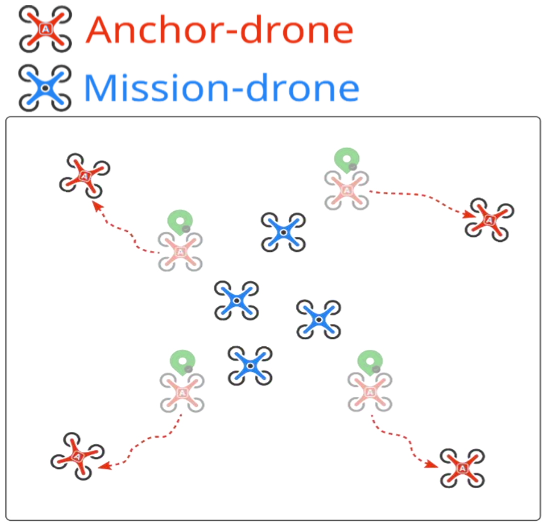


GitHub: https://github.com/vladniculescu/uwb-software-library

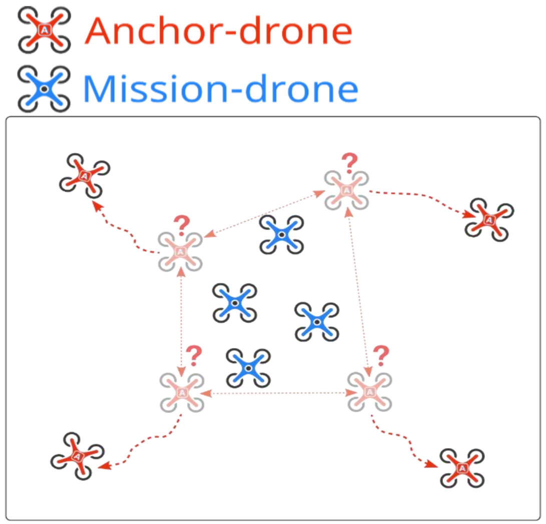
Ideal Landing (Baseline)



Ideal Landing



Dynamic Anchors

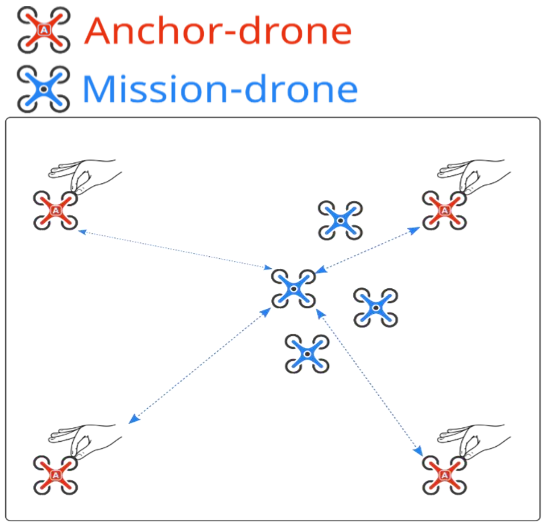


Self-localize

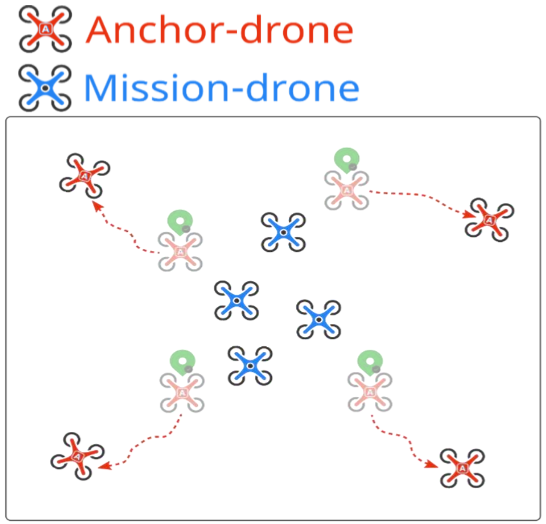
Assumptions:

- Anchor installation
- Anchors' positions must be ***known***

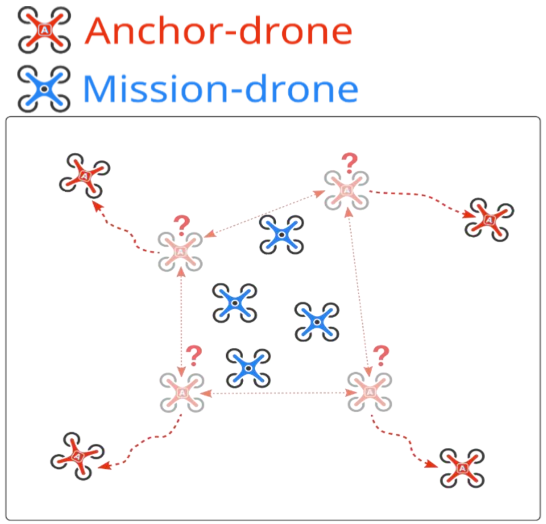
Dynamic Anchor System



Ideal Landing



Dynamic Anchors



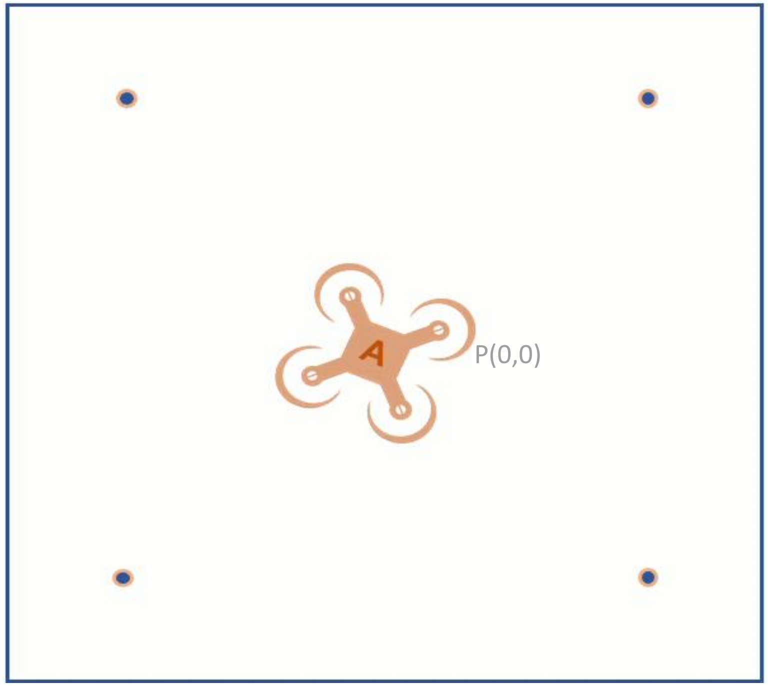
Self-localize

Assumptions:

- ~~Anchor installation~~
- Anchors' positions must be ***known***

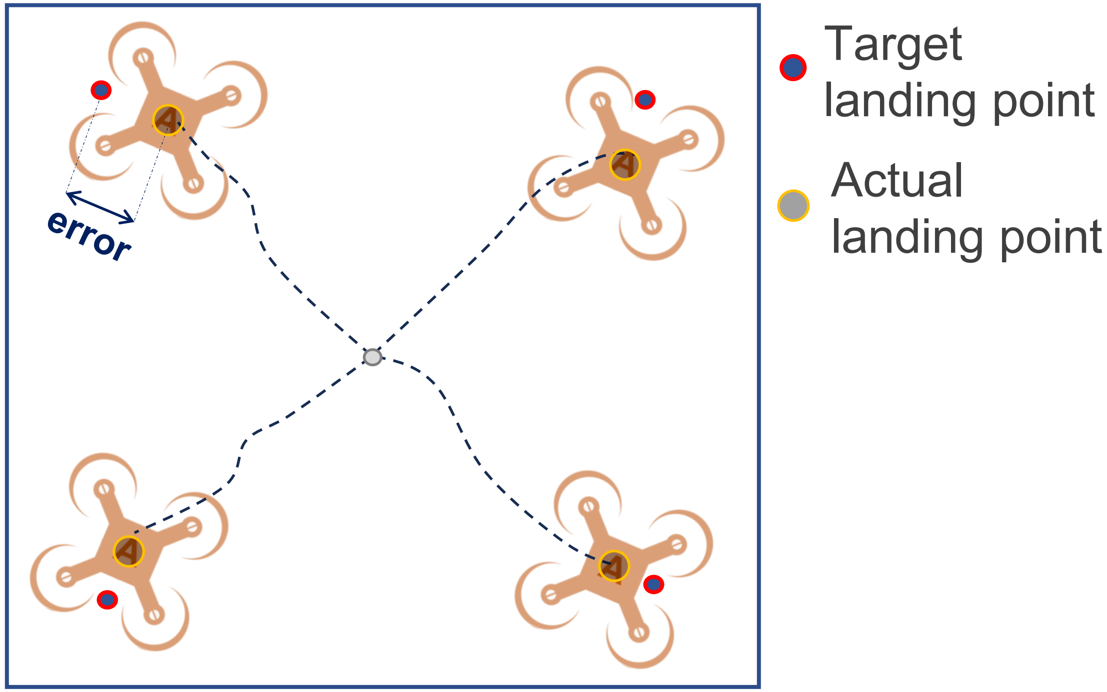
Dynamic Anchor System

1 Anchor-Drones **fly** & **land** to a target location



Navigation based only on the onboard state estimation (Optical-Flow, IMU)

2 After landing Anchor-Drones start **ranging** with MDs

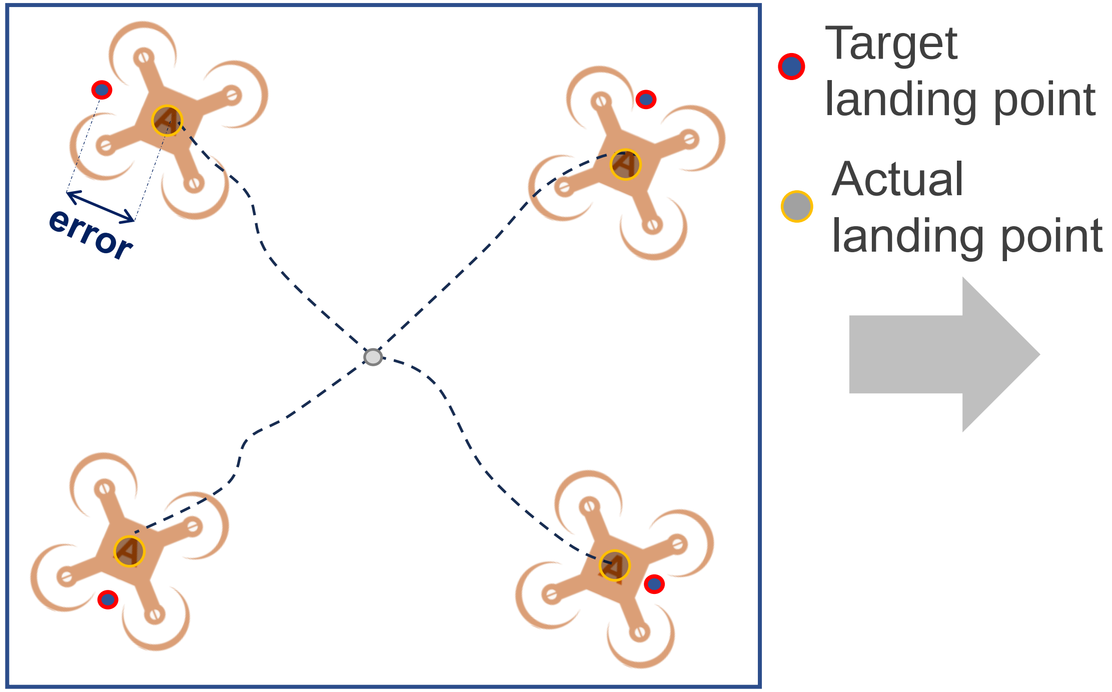


There is an **error** between target landing point and the actual one

Dynamic Anchor System

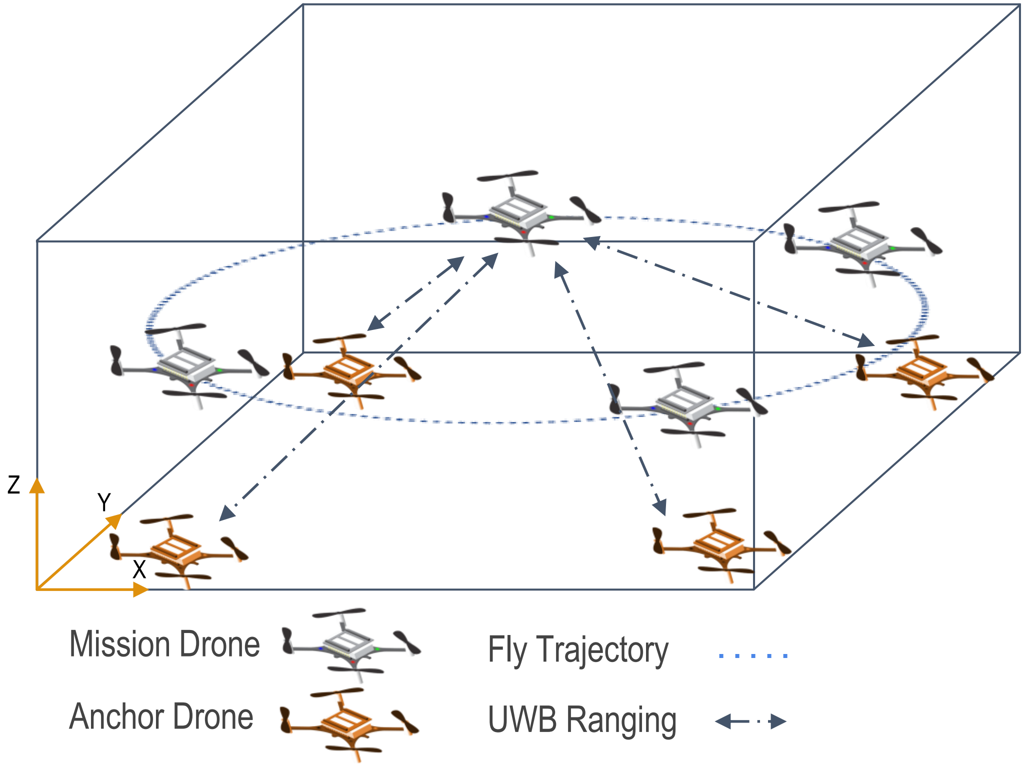
2

After landing Anchor-Drones start **ranging** with MDs

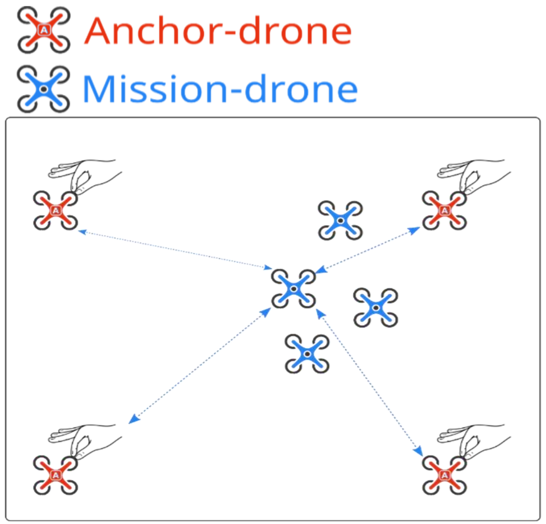


There is an **error** between target landing point and the actual one

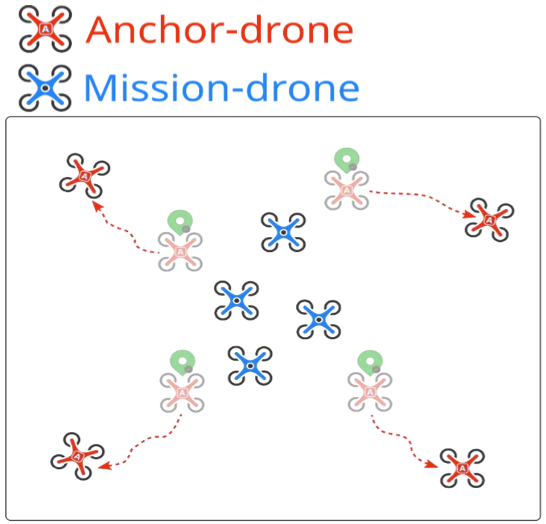
Mission-Drones are receiving **ranging information** containing Estimated Anchor Drones' positions



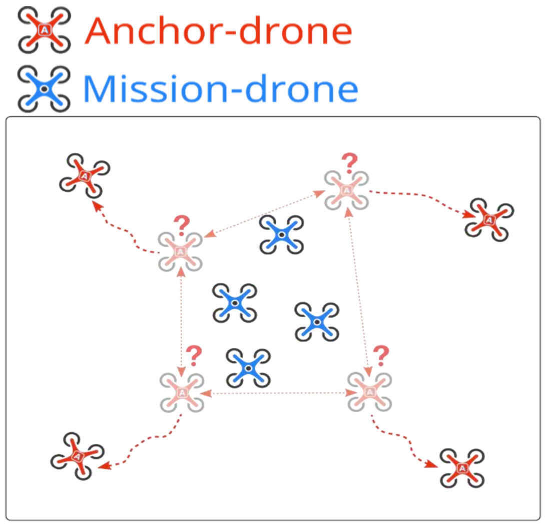
Dynamic Anchor System



Ideal Landing



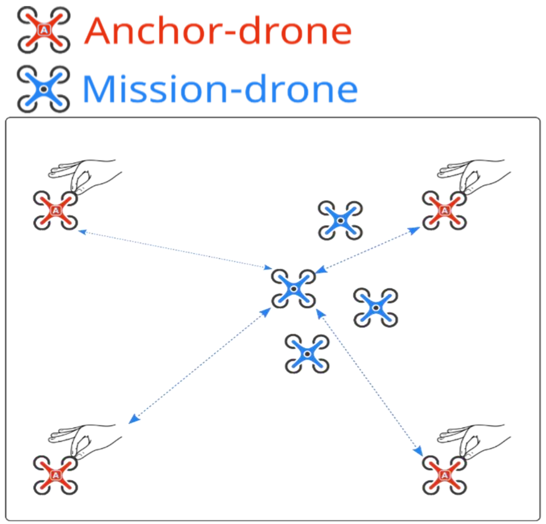
Dynamic Anchors



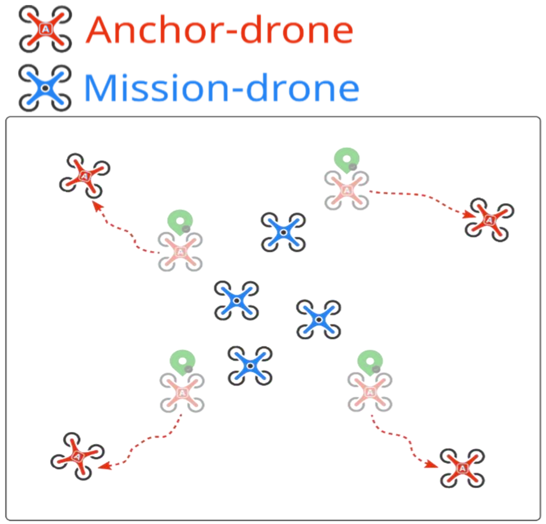
Self-localize

- ☑ ~~installation of anchor infrastructure~~
- ✗ Anchors' positions must be ***known***

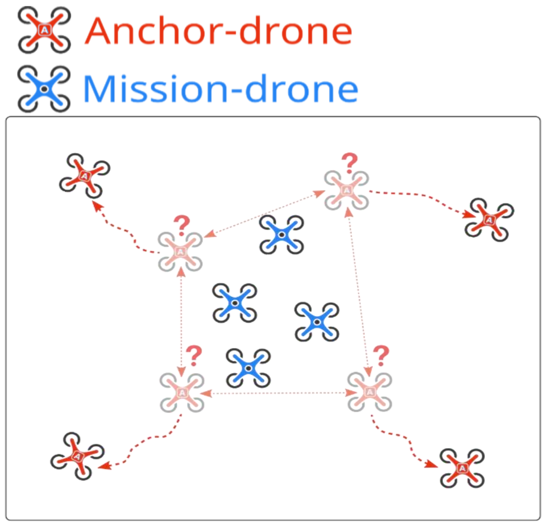
Dynamic Anchor System



Ideal Landing



Dynamic Anchors



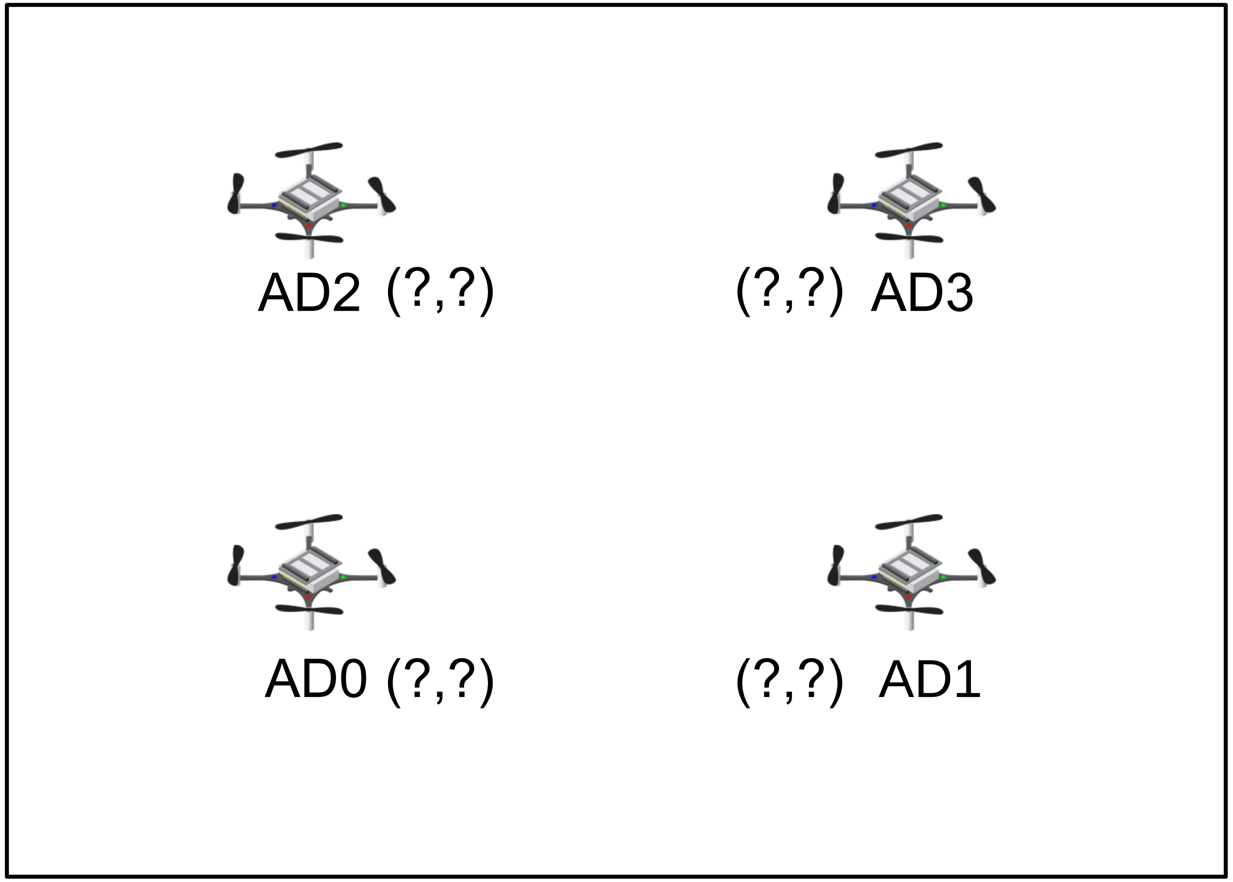
Self-localize

- ☑ ~~installation of anchor infrastructure~~
- ☑ ~~Anchors' positions must be ***known***~~

How does Self-localization System work?

Convention:

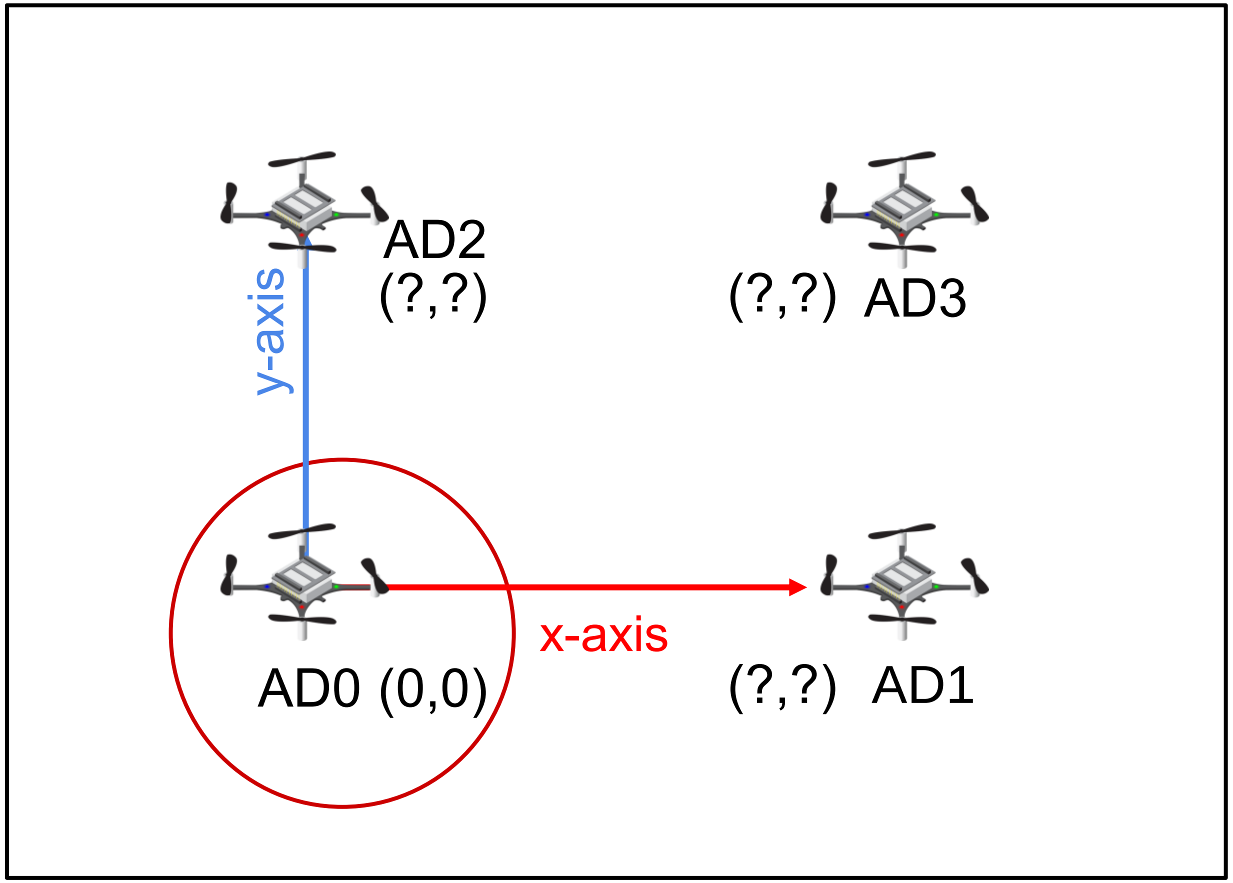
- ADs placed in unknown positions



How does Self-localization System work?

Convention:

- ADs placed in unknown positions
- AD0 act as the origin of the coordinate system
- AD0 - AD1 form x-coordinate
- AD2 considered the positive y-coordinate



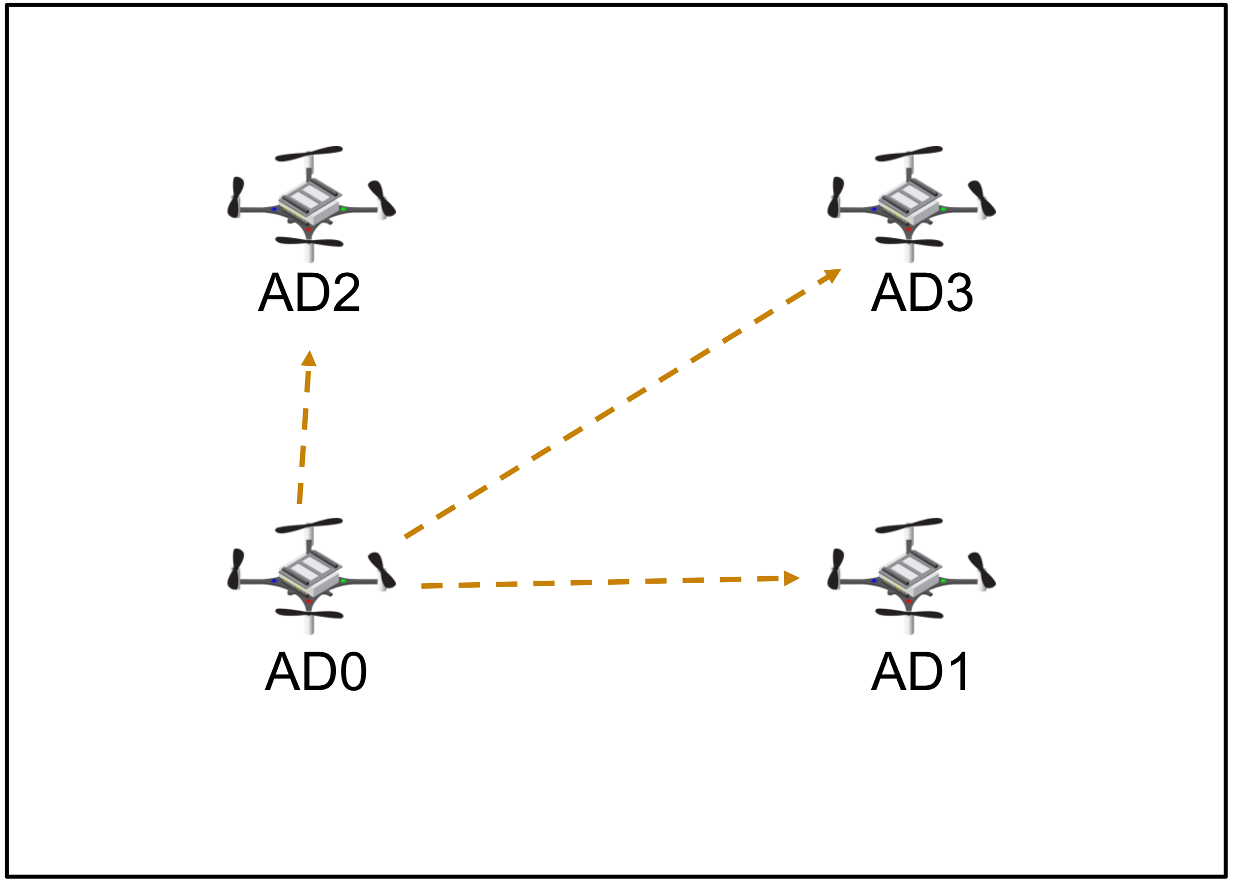
How does Self-localization System work?

Convention:

- ADs placed in unknown positions
- AD0 act as the origin of the coordinate system
- AD0 - AD1 form x-coordinate
- AD2 considered the positive y-coordinate

Communication flow:

1. AD0 broadcasts a UWB message to all Ads



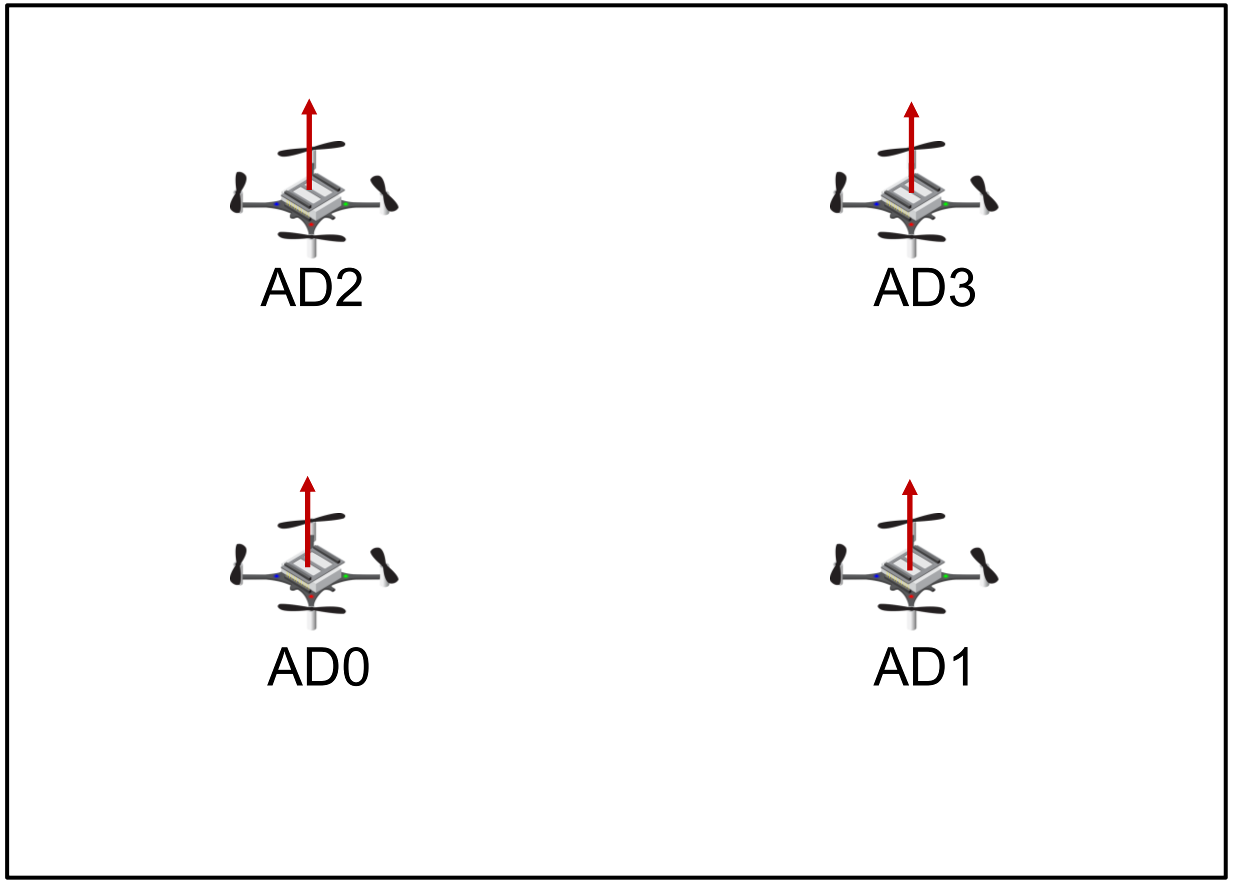
How does Self-localization System work?

Convention:

- ADs placed in unknown positions
- AD0 act as the origin of the coordinate system
- AD0 - AD1 form x-coordinate
- AD2 considered the positive y-coordinate

Communication flow:

1. AD0 broadcasts a UWB message to all Ads
2. The ADs take off and hover at 0.5m



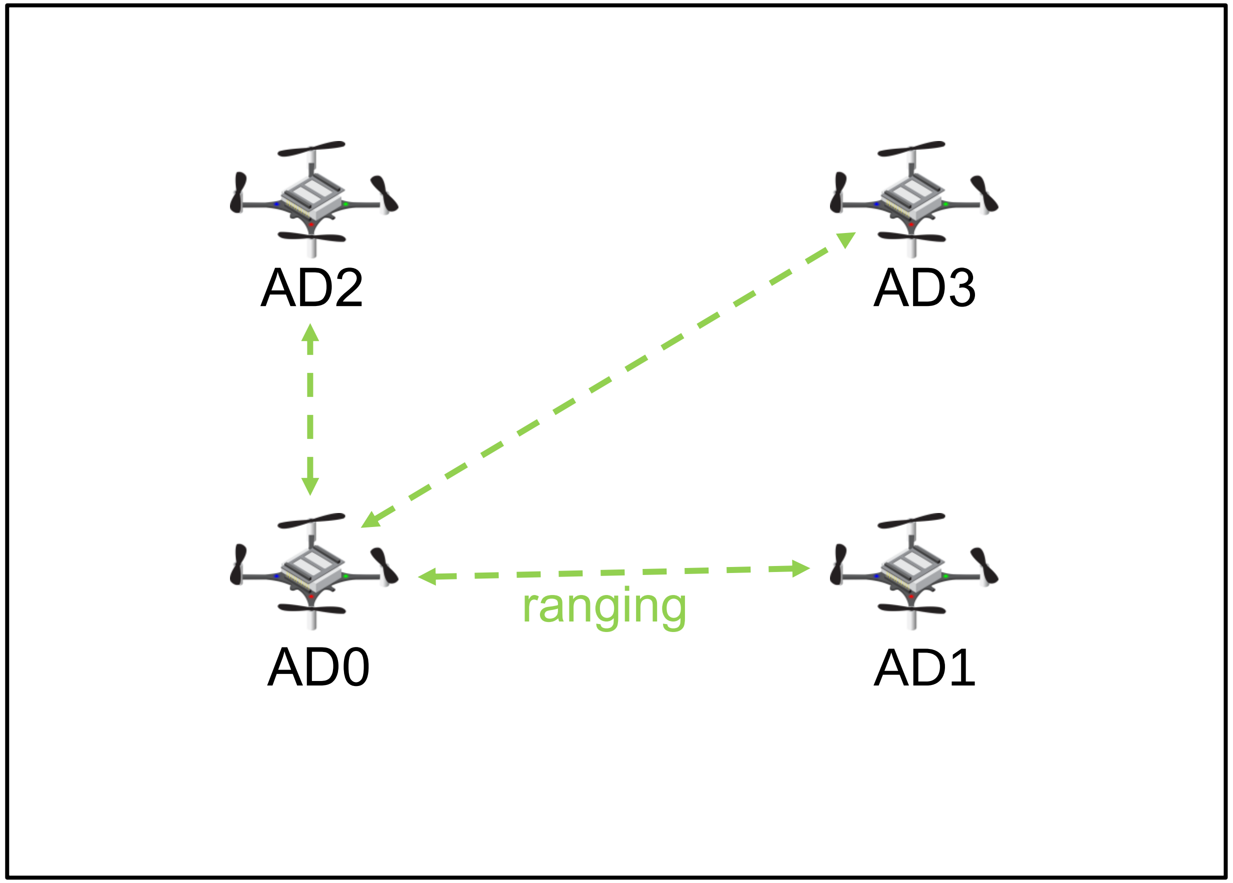
How does Self-localization System work?

Convention:

- ADs placed in unknown positions
- AD0 act as the origin of the coordinate system
- AD0 - AD1 form x-coordinate
- AD2 considered the positive y-coordinate

Communication flow:

1. AD0 broadcasts a UWB message to all Ads
2. The ADs take off and hover at 0.5m
3. ADi does ranging with AD(i+1)..N



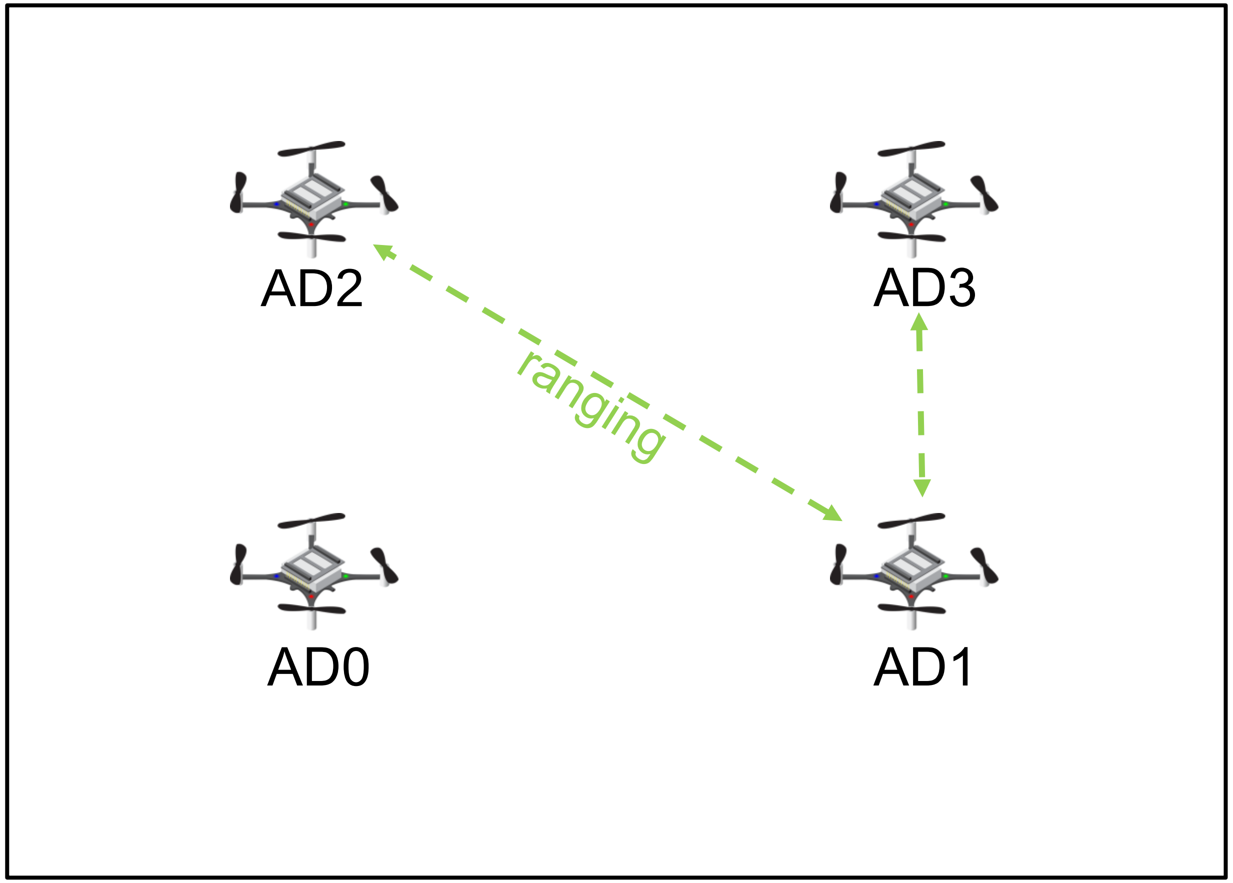
How does Self-localization System work?

Convention:

- ADs placed in unknown positions
- AD0 act as the origin of the coordinate system
- AD0 - AD1 form x-coordinate
- AD2 considered the positive y-coordinate

Communication flow:

1. AD0 broadcasts a UWB message to all Ads
2. The ADs take off and hover at 0.5m
3. ADi does ranging with AD(i+1)..N



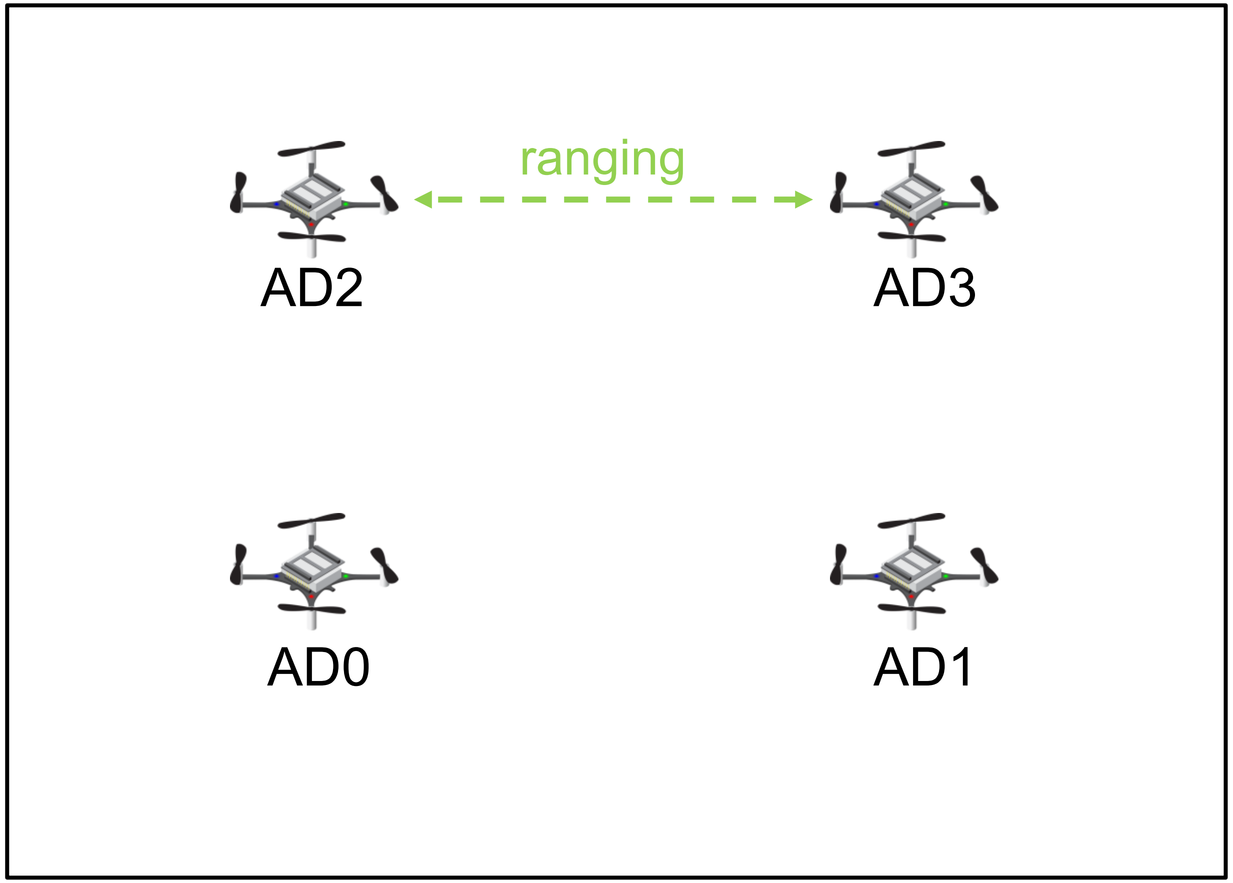
How does Self-localization System work?

Convention:

- ADs placed in unknown positions
- AD0 act as the origin of the coordinate system
- AD0 - AD1 form x-coordinate
- AD2 considered the positive y-coordinate

Communication flow:

1. AD0 broadcasts a UWB message to all Ads
2. The ADs take off and hover at 0.5m
3. ADi does ranging with AD(i+1)..N



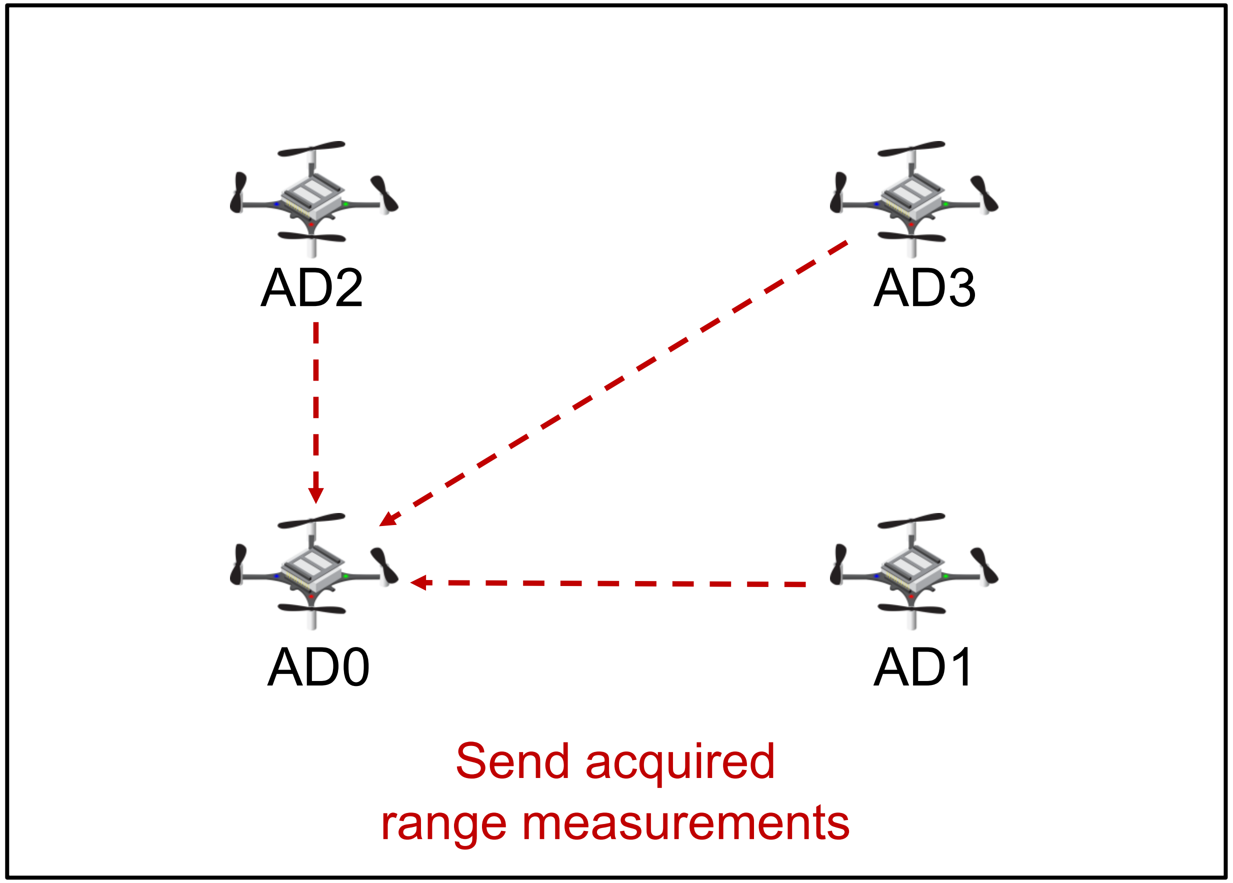
How does Self-localization System work?

Convention:

- ADs placed in unknown positions
- AD0 act as the origin of the coordinate system
- AD0 - AD1 form x-coordinate
- AD2 considered the positive y-coordinate

Communication flow:

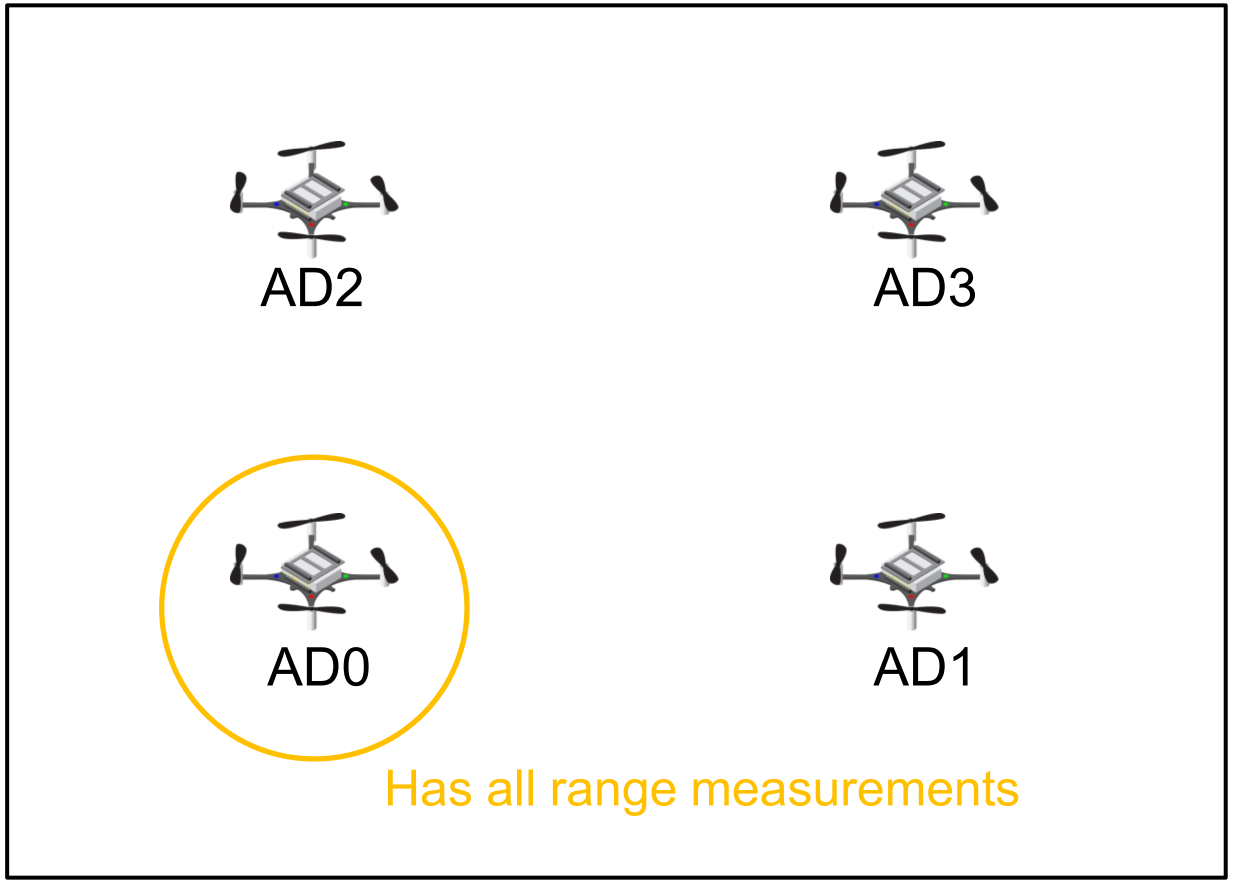
1. AD0 broadcasts a UWB message to all Ads
2. The ADs take off and hover at 0.5m
3. ADi does ranging with AD(i+1)..N
4. ADs send all measurements to AD0



How does Self-localization System work?

Convention:
- ADs placed in unknown positions
- AD0 act as the origin of the coordinate system
- AD0 - AD1 form x-coordinate
- AD2 considered the positive y-coordinate

Communication flow:
1. AD0 broadcasts a UWB message to all Ads
2. The ADs take off and hover at 0.5m
3. ADi does ranging with AD(i+1)..N
4. ADs send all measurements to AD0
5. AD0 builds the distance matrix D and runs **Multi-Dimensional Scaling** (MDS)



$$D := \begin{pmatrix} d_{1,1} & d_{1,2} & \cdots & d_{1,M} \\ d_{2,1} & d_{2,2} & \cdots & d_{2,M} \\ \vdots & \vdots & & \vdots \\ d_{M,1} & d_{M,2} & \cdots & d_{M,M} \end{pmatrix} \rightarrow \boxed{\text{MDS}} \rightarrow$$

Runs on AD0

Anchor Drone Positions AD0, AD1, … ADM

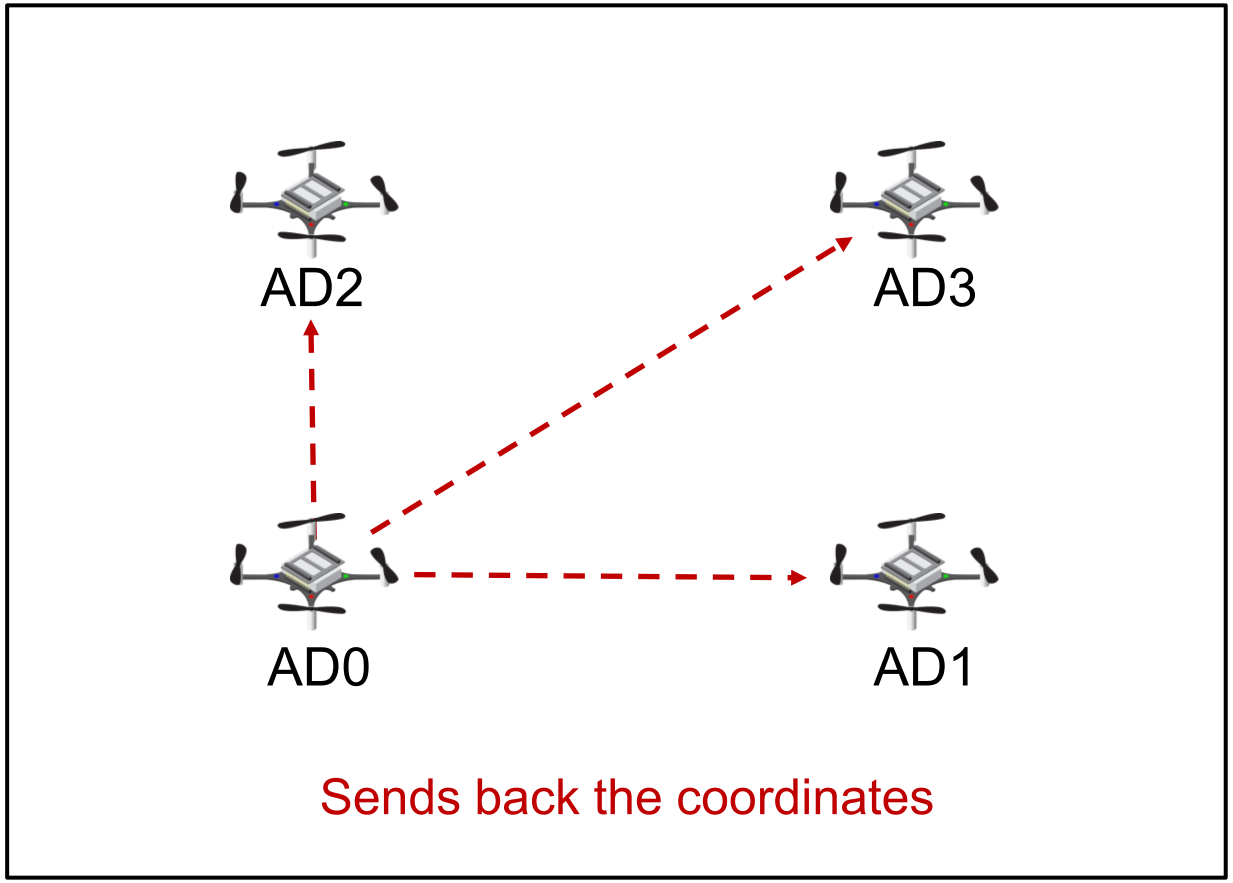
How does Self-localization System work?

Convention:

- ADs placed in unknown positions
- AD0 act as the origin of the coordinate system
- AD0 - AD1 form x-coordinate
- AD2 considered the positive y-coordinate

Communication flow:

1. AD0 broadcasts a UWB message to all Ads
2. The ADs take off and hover at 0.5m
3. ADi does ranging with AD(i+1)..N
4. ADs send all measurements to AD0
5. AD0 builds the distance matrix D and runs
6. AD0 sends the coordinates to AD1..N



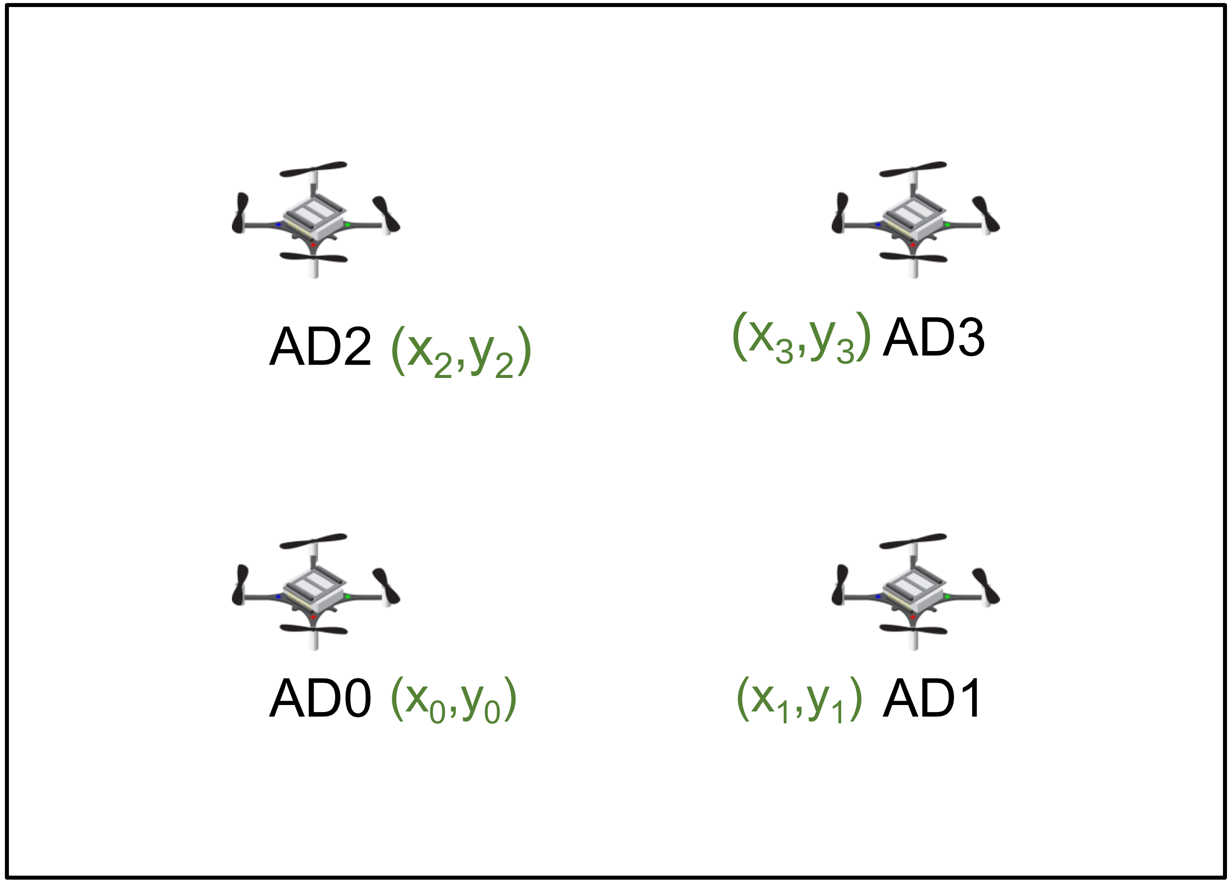
How does Self-localization work?

Convention:

- ADs placed in unknown positions
- AD0 act as the origin of the coordinate system
- AD0 - AD1 form x-coordinate
- AD2 considered the positive y-coordinate

Communication flow:

1. AD0 broadcasts a UWB message to all Ads
2. The ADs take off and hover at 0.5m
3. ADi does ranging with AD(i+1)..N
4. ADs send all measurements to AD0
5. AD0 builds the distance matrix D and runs
6. AD0 sends the coordinates to AD1..N
7. ADs know their relative position



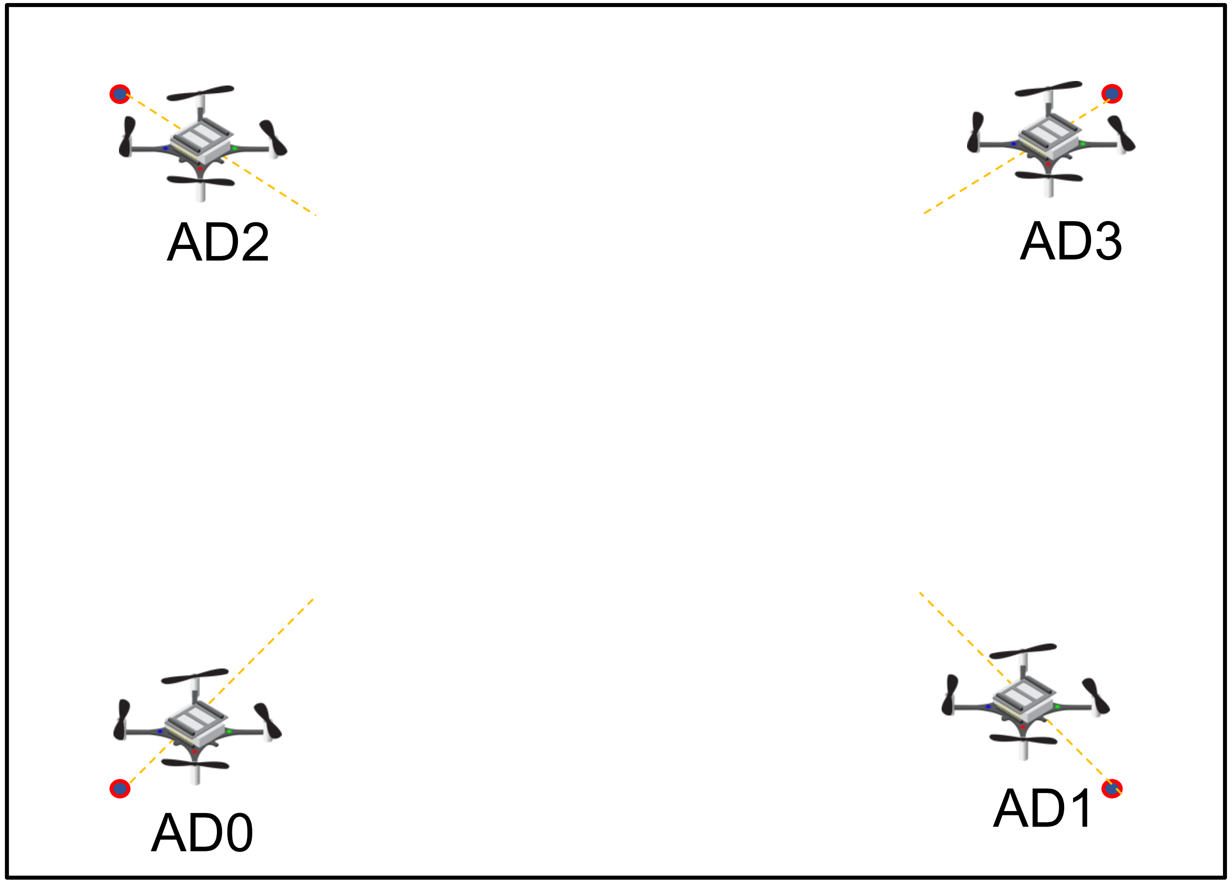
How does Self-localization System work?

Convention:

- ADs placed in unknown positions
- AD0 act as the origin of the coordinate system
- AD0 - AD1 form x-coordinate
- AD2 considered the positive y-coordinate

Communication flow:

1. AD0 broadcasts a UWB message to all Ads
2. The ADs take off and hover at 0.5m
3. ADi does ranging with AD(i+1)..N
4. ADs send all measurements to AD0
5. AD0 builds the distance matrix *D* and runs
6. AD0 sends the coordinates to AD1..N
7. ADs know their relative position
8. ADs fly toward the target landing point



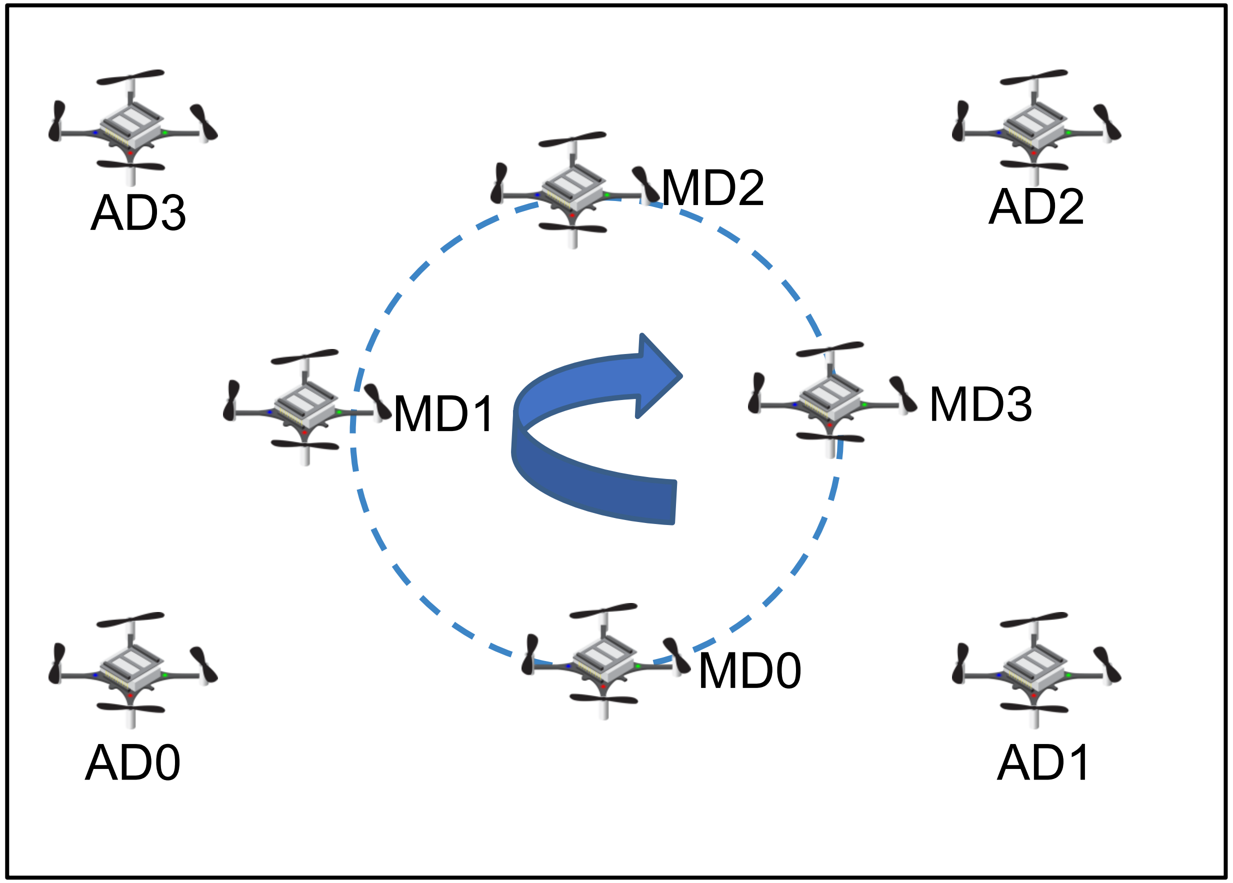
How does Self-localization System work?

Convention:

- ADs placed in unknown positions
- AD0 act as the origin of the coordinate system
- AD0 - AD1 form x-coordinate
- AD2 considered the positive y-coordinate

Communication flow:

1. AD0 broadcasts a UWB message to all Ads
2. The ADs take off and hover at 0.5m
3. ADi does ranging with AD(i+1)..N
4. ADs send all measurements to AD0
5. AD0 builds the distance matrix *D* and runs
6. AD0 sends the coordinates to AD1..N
7. ADs know their relative position
8. ADs fly toward the target landing point
9. Ranging starts and the MDs start flying



In-field Results

Experiment Setup

- 4 Anchor Drones
- 4-8 Mission Drones
- 3 times flying around a 1.5[m] radius circle
- ~1 m/s velocity

Evaluation Metrics

- Localization RMSE
- Control RMSE

Nano-Drone Platform

Crazyflie 2.1

UWB deck

Flow deck v2

In-field Results

Experiment Setup

- 4 Anchor Drones
- 4-8 Mission Drones
- 3 times flying around a 1.5[m] radius circle
- ~1 m/s velocity

Evaluation Metrics

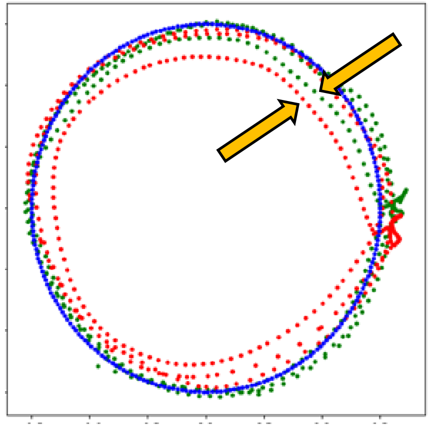
- Localization RMSE → Error: **Estimate** – **GT**
- Control RMSE

Nano-Drone Platform


Crazyflie 2.1


UWB deck


Flow deck v2



In-field Results

Experiment Setup

- 4 Anchor Drones
- 4-8 Mission Drones
- 3 times flying around a 1.5[m] radius circle
- ~1 m/s velocity

Nano-Drone Platform

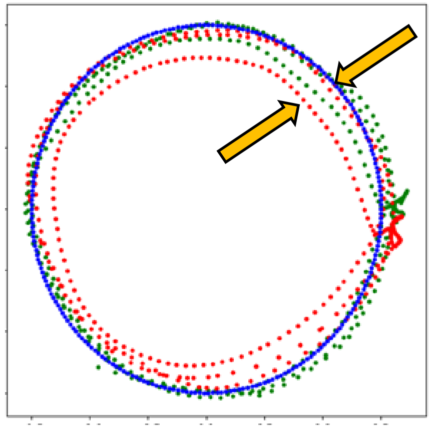

Crazyflie 2.1


UWB deck


Flow deck v2

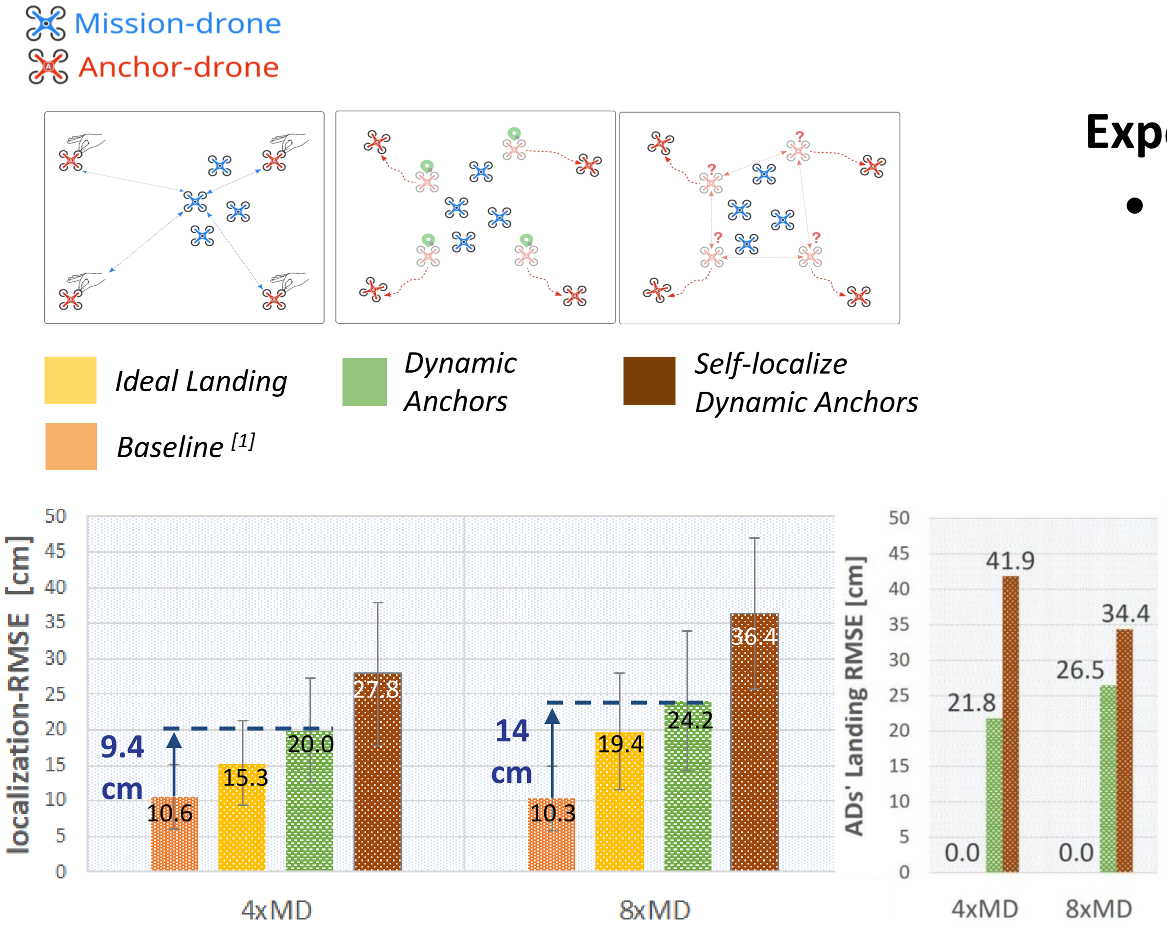
Evaluation Metrics

- Localization RMSE
- Control RMSE → $\underbrace{\textbf{Ideal} \quad \textbf{GT}}_{\text{Error}}$



● Ideal ● Estimation ● Ground-truth

In-field Results

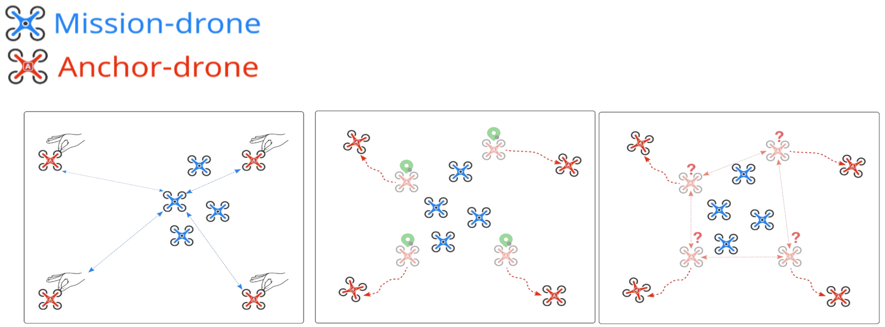


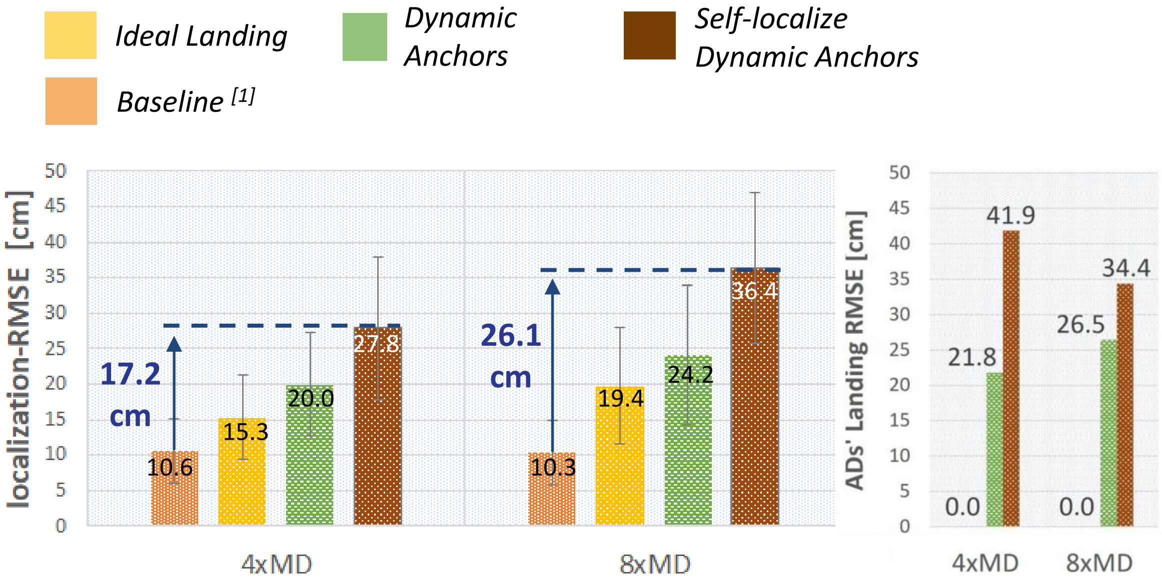
Experiments with 4/8 MDs and 4 ADs:

- Flexibility at the cost of a higher localization error

[1] Fixed 6x Anchor module setup using Bitcraze's UWB localization firmware as the baseline

In-field Results - self-localization





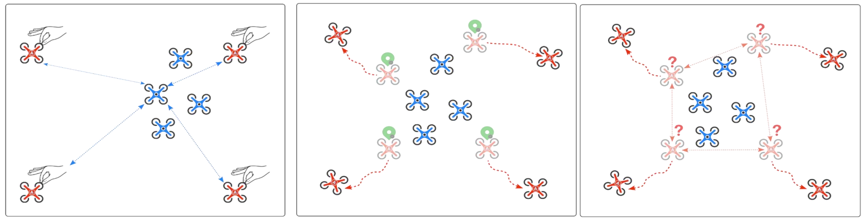
Experiments with 4/8 MDs and 4 ADs:

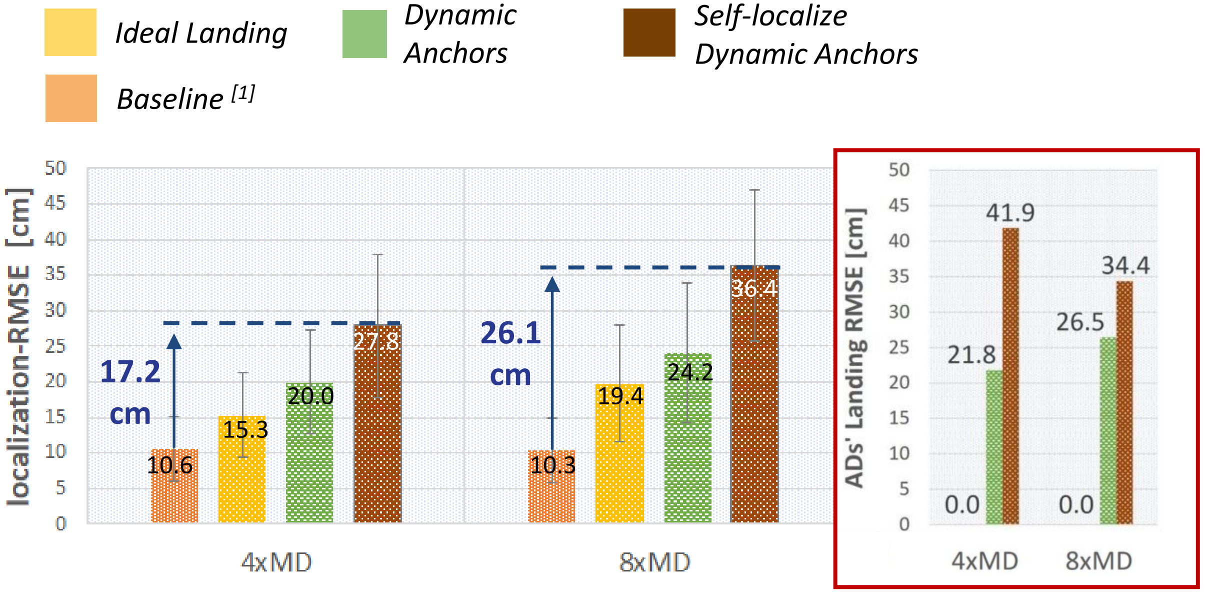
- Flexibility at the cost of a higher localization error
- Self-Localization runs on-board in 5 sec

[1] Fixed 6x Anchor module setup using Bitcraze's UWB localization firmware as the baseline

Results - Impact of AD self-localization

Mission-drone
Anchor-drone



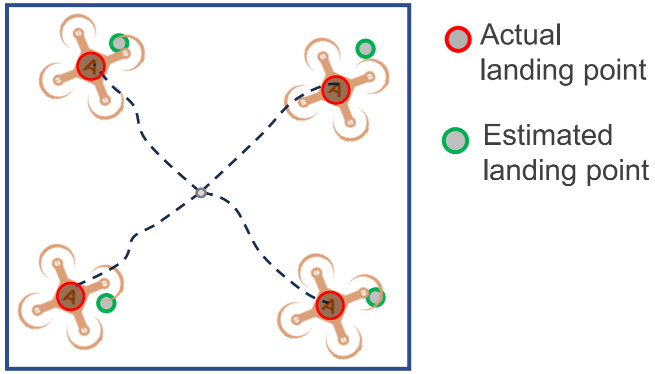


Experiments with 4/8 MDs and 4 ADs:

- Flexibility at the cost of a higher localization error
- Self-Localization runs on-board in 5 sec

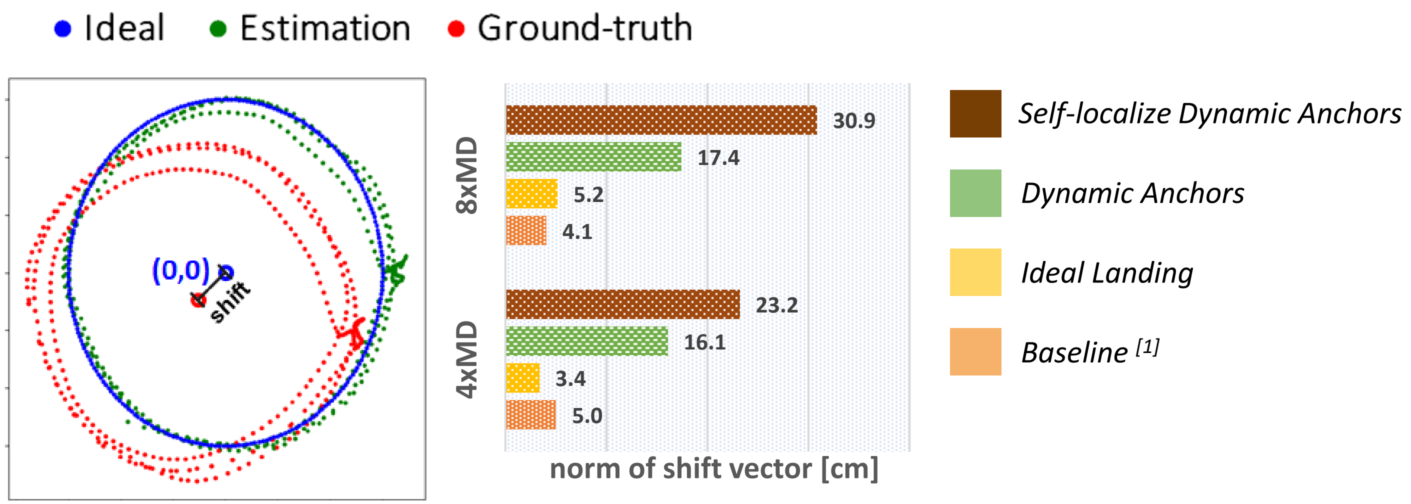
Localization Error is contributed by

- UWB Error (also on static anchors)
- **Anchors' Landing Error**



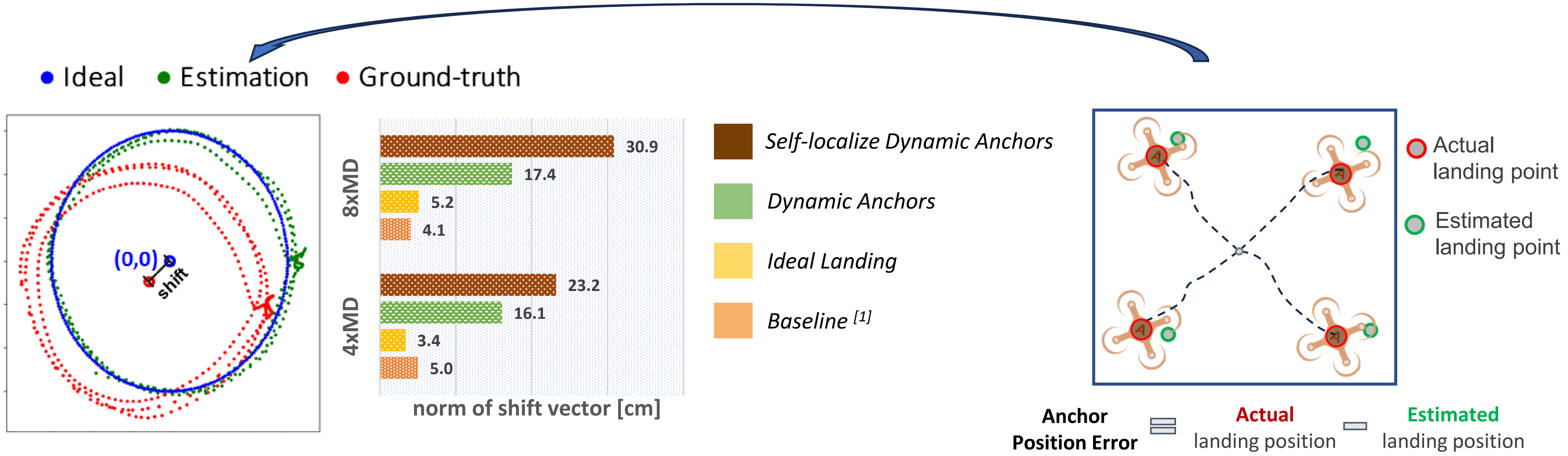
[1] Fixed 6x Anchor module setup using Bitcraze's UWB localization firmware as the baseline

Results



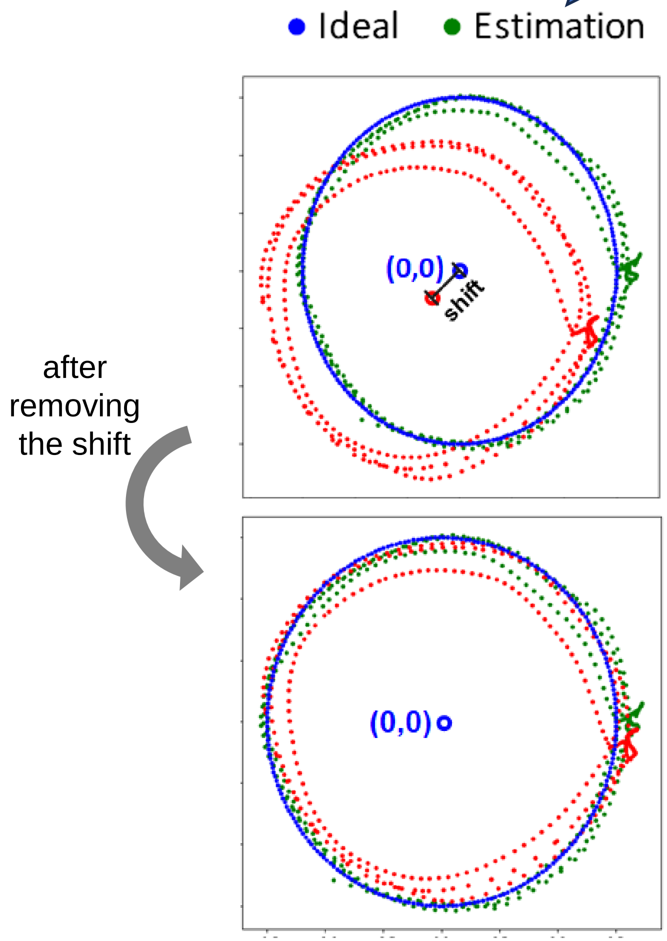
[1] Fixed 6x Anchor module setup using Bitcraze's UWB localization firmware as the baseline

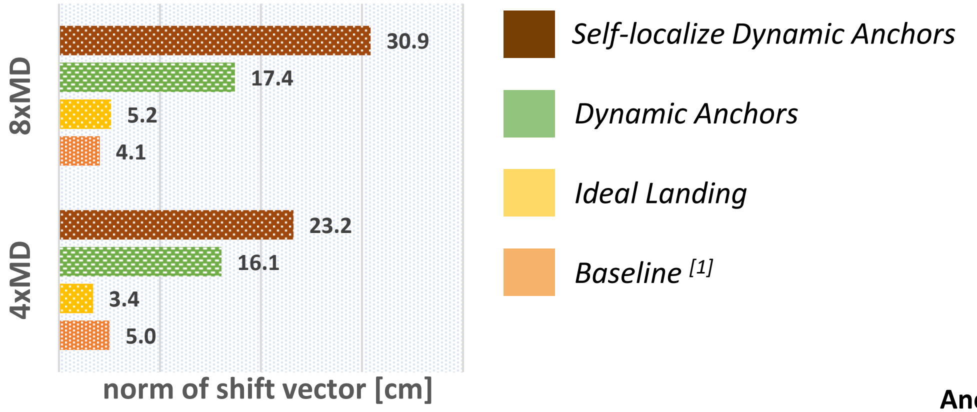
Results

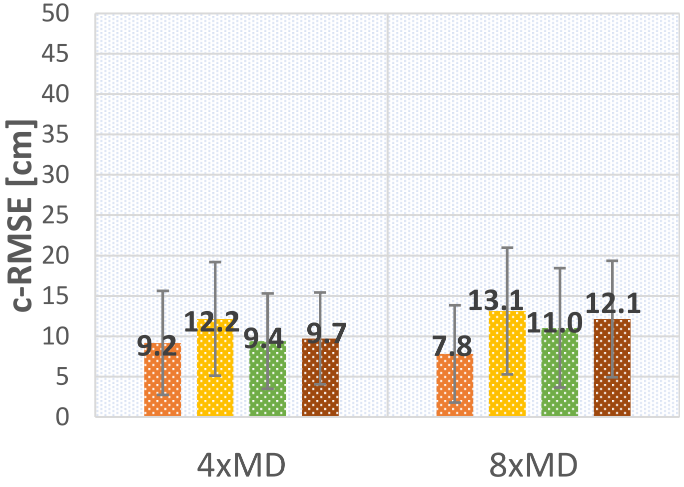


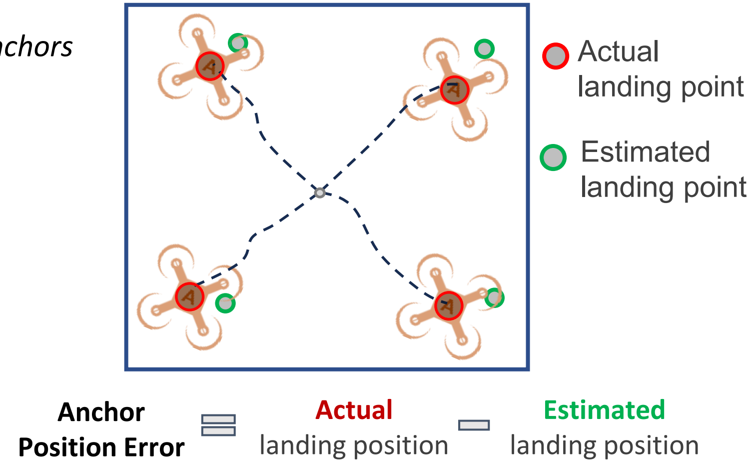
[1] Fixed 6x Anchor module setup using Bitcraze's UWB localization firmware as the baseline

Results



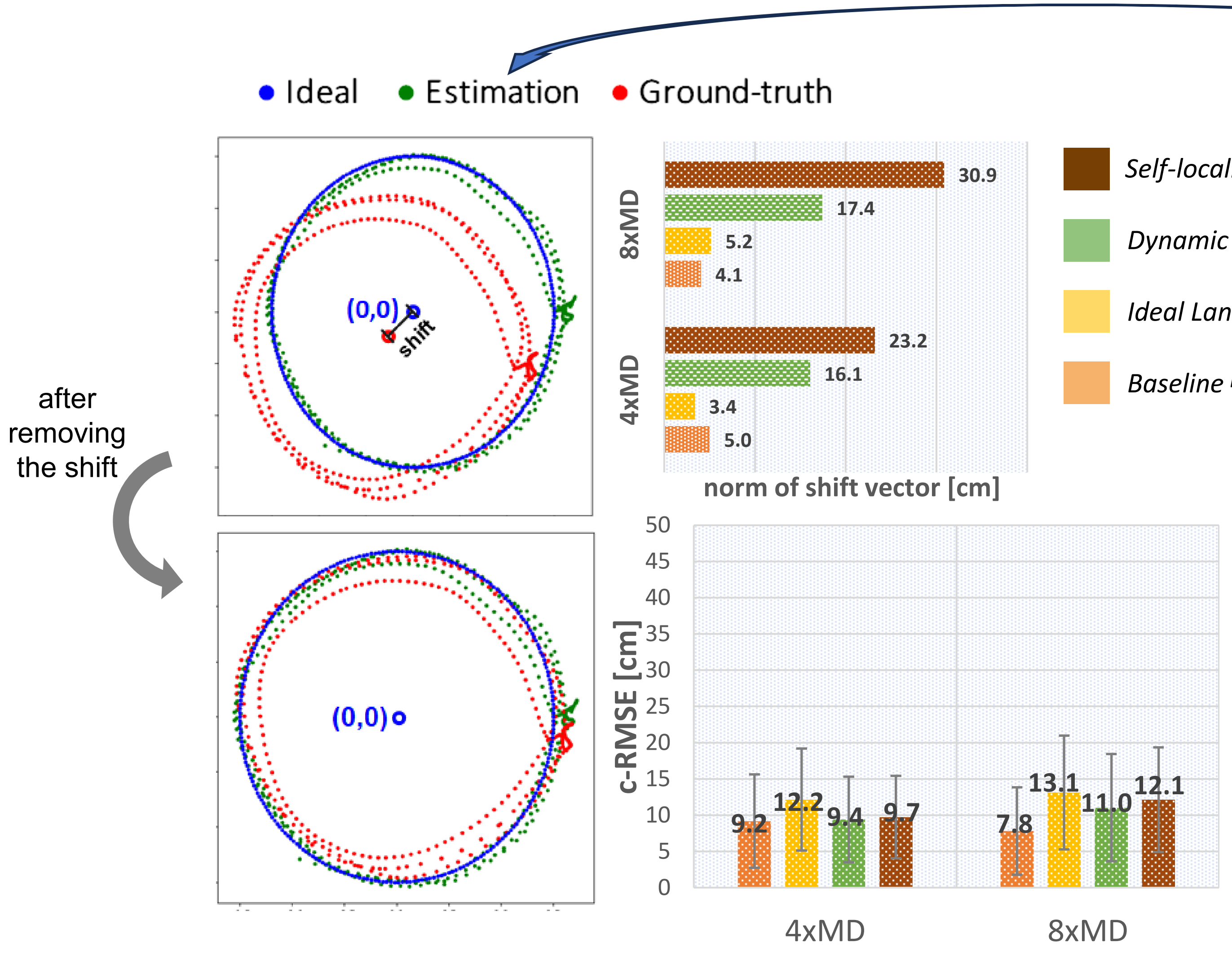


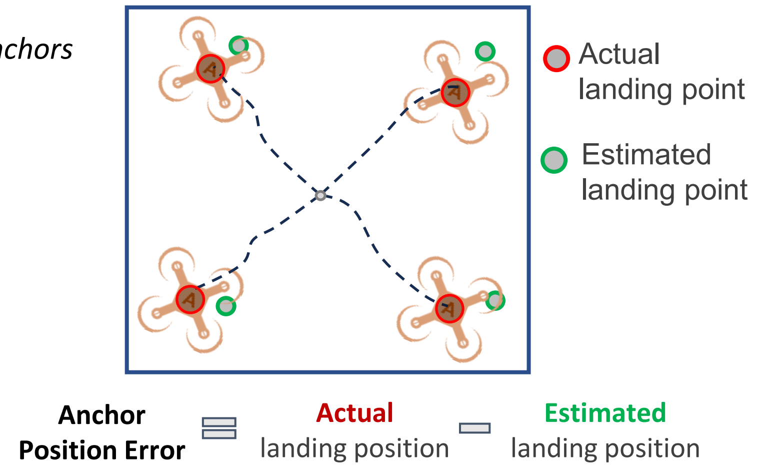




[1] Fixed 6x Anchor module setup using Bitcraze's UWB localization firmware as the baseline

Results





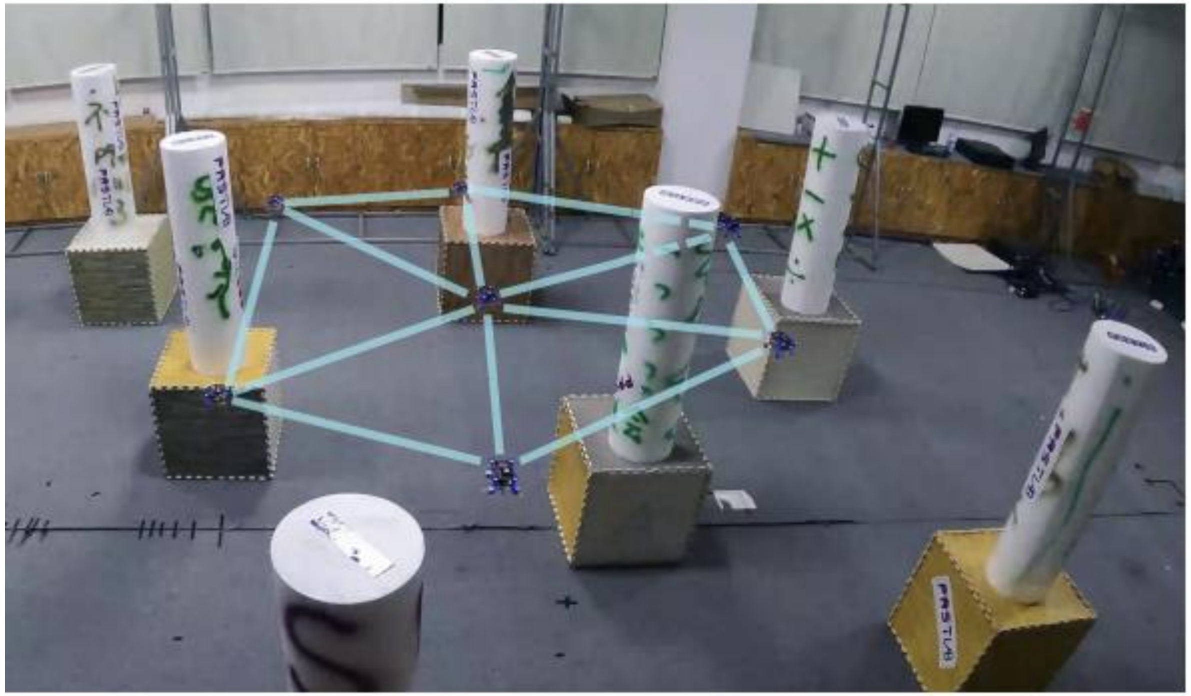
Take away message

Higher localization error in dynamic anchor systems is mainly impacted by a shift vector

However, the control error always remains low to enable drones to precisely follow the commanded trajectory

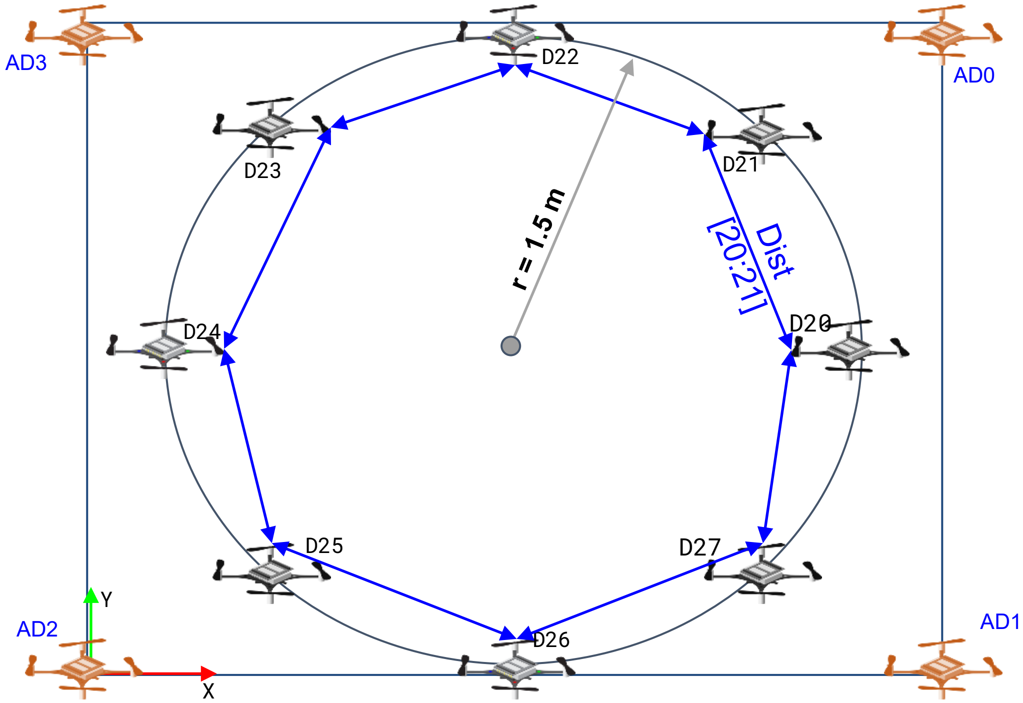
[1] Fixed 6x Anchor module setup using Bitcraze's UWB localization firmware as the baseline

Results



Flight Formation*

Longitude inter-drone distance



Longitude inter-drone distance always bounded below 13.7% i.e., ±15.6 cm

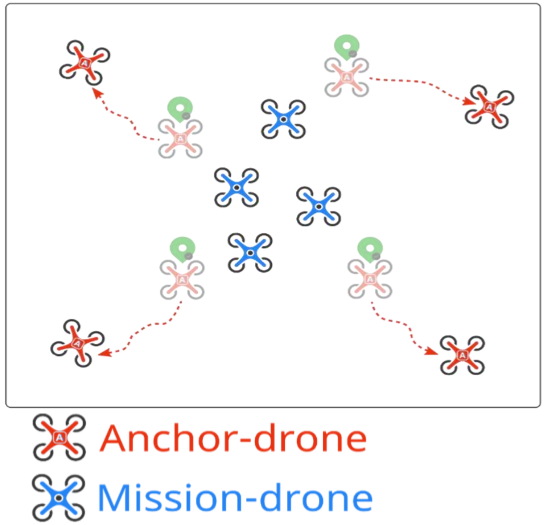
[*] https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9812050

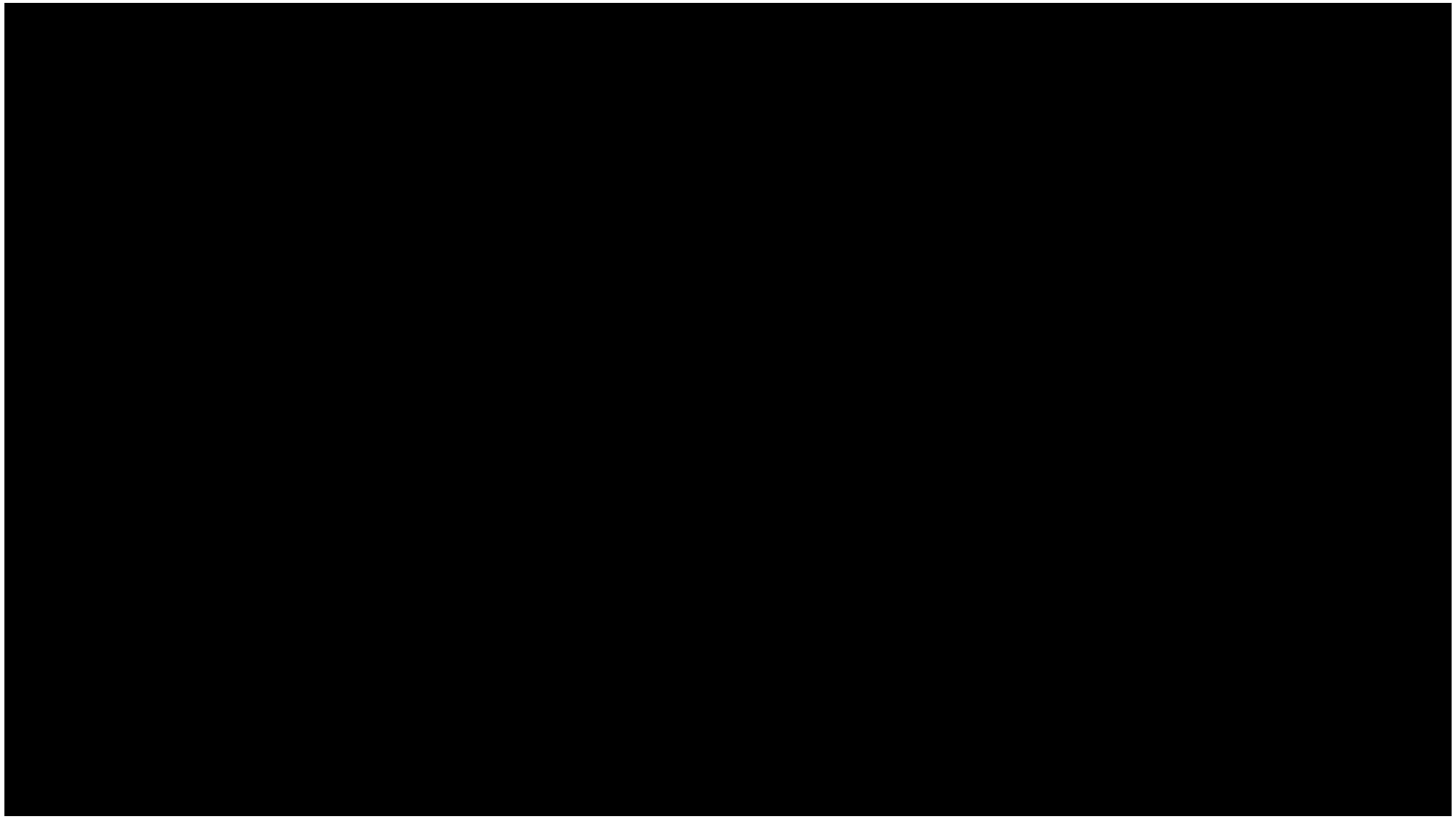
Demo

Setup

- 4 ADs and 8 MDs
- Dynamic Anchor Deployment
- MDs fly in a 1.5[m] circular trajectory

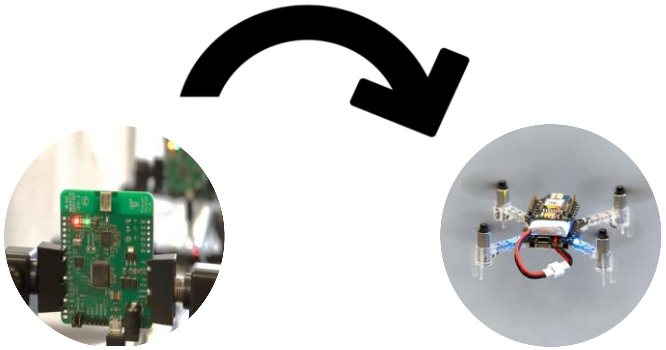
Dynamic Anchors





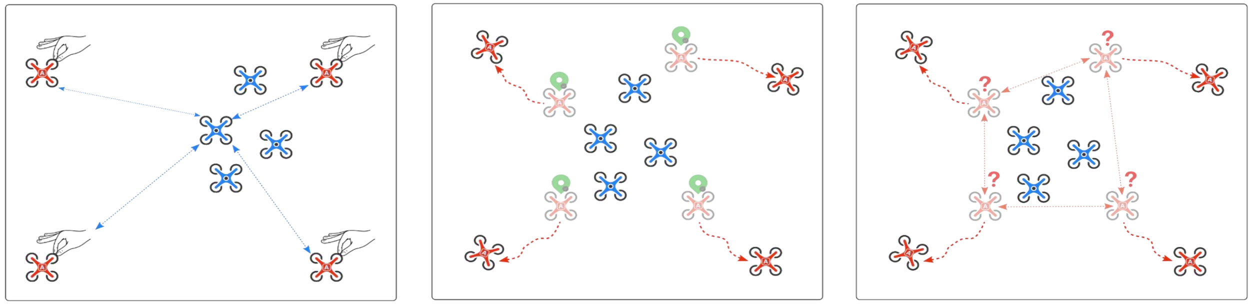
Conclusion

Leveraging Nano-drone as Dynamic Anchor to eliminate the need for a static UWB infrastructure



Anchor Infrastructure

Dynamic Anchor



	Most challenging configuration 8x MD		
	IDEAL LAND	DYNAMIC ANCHOR	SELF LOCALIZE
l-RMSE	19.4	24.2	36.4
c-RMSE	13.1	11.0	12.1

Longitude inter-drone distance always **<15.6 [cm]**

Thank you for your attention!

Mahyar Pourjabar

Ph.D. student at University of Bologna, Italy
mahyar.pourjabar2@unibo.it