Localization and Positioning

Veronica Campana

Outline

- **Positioning problem**

  Sensing systems for positioning

  Positioning problem definition

  Techniques for Distance Estimation

  Static Estimator

  Ultra-Wideband (UWB) Technology
- **Localization problem**

  Dynamic Estimator
- **Self Localization approach**

Positioning systems approaches

Three main categories:

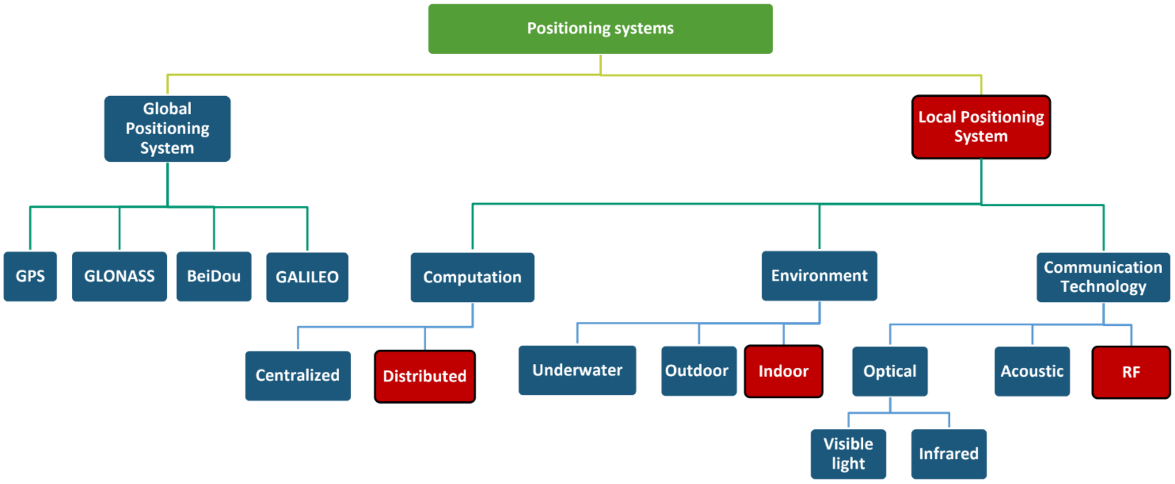
Vision-based → Main **advantages**: - High accuracy and extensive environmental details; → Main **drawbacks**: - It suffers low visibility conditions,presence of dust or fog, change of illumination.

Infrared/Lidar-based → Main **advantages**: - Accurate 3D distance measurements, works in low light; → Main **drawbacks**: - Sensitive to reflective surfaces, fog, or dust scattering.

Radio-based → Main **advantages**: -Robust to lights changes, dust and fog - Low computational cost → Main **drawbacks**: - Lower spatial resolution, can suffer multipath or interference.

Overview of Positioning Systems

Red ones —> the ones we will consider



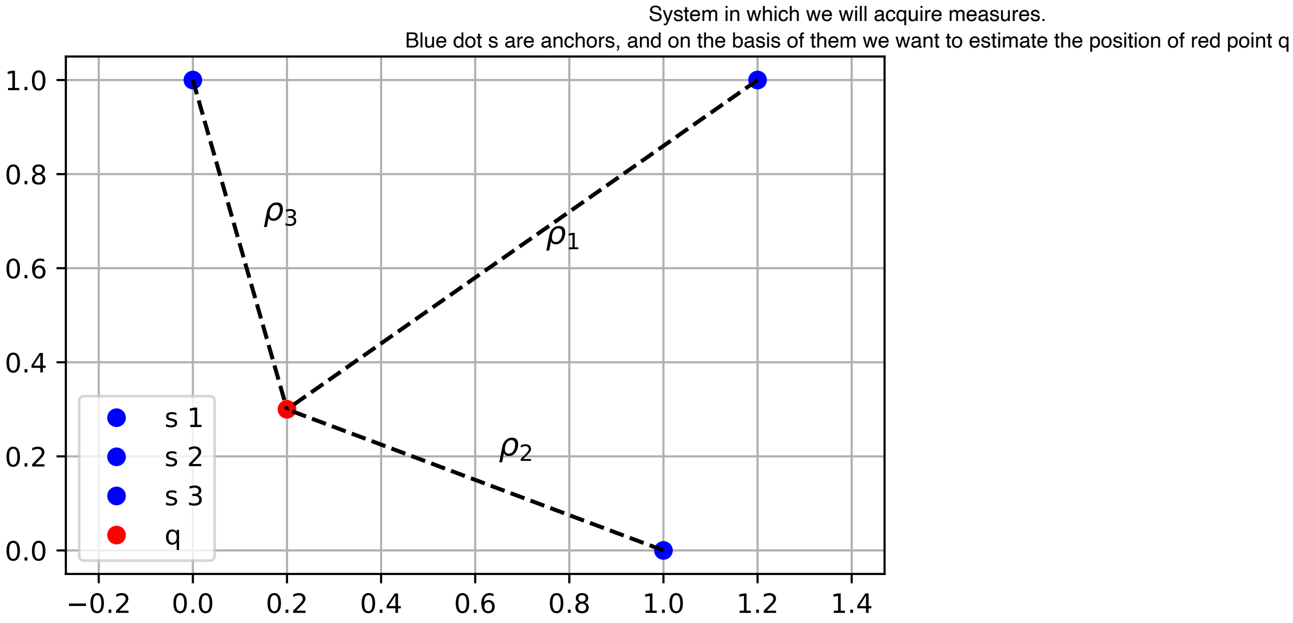
Positioning Problem Definition

In the analysis, we will consider the following model for the measurement:

$$\mathbf{b} = h(\mathbf{q}) + \mathbf{w}$$

where $\mathbf{q} \in \mathbb{R}^n$ is in general the position to be found, with $n = 2$ for 2D problems, $\mathbf{b} \in \mathbb{R}^m$ is the vector of the m measurements, and $\mathbf{w} \in \mathbb{R}^m$ is the vector of the sensor uncertainties.

Display of the positioning problem


System in which we will acquire measures.
Blue dot s are anchors, and on the basis of them we want to estimate the position of red point q

Techniques for Distance Estimation

The distance between the tag and the anchor can be estimated using different techniques. The most common are:

- Time of Arrival (ToA)
- Time Difference of Arrival (TDoA)
- Angle of Arrival (AoA)

LOS vs. NLOS

An important concept for RF positioning systems is the **Line Of Sight (LOS)** configuration, i.e. the existence of an obstacle-free path from a source node to a destination node. All the RF positioning solution performance are affected by LOS, even though with different levels of problems. The effect of a **Non Line Of Sight (NLOS)** condition is the presence of a bias, which poses serious problems in estimation and characterisation of systems. Therefore, in **NLOS** conditions, various mitigation techniques should be considered.

Time of Arrival (ToA) Approach

We assume that the target node emits a signal at time t that is received by the i-th base station at time t_i.

The distance ρ_i is then given by:

$$\rho_i = (t_i - t)c,$$

where c is typically the velocity of light $3 \cdot 10^8$ m/s (e.g., for electromagnetic signals) or the velocity of sound 340 m/s (e.g., for ultrasound systems). From the previous expression, it turns out the necessity of **node synchronization**, while the quantity $(t_i - t)$ is the Time of Flight (ToF).

Time of Arrival (ToA) Drawbacks

ToA has some drawbacks:

- Both the target node and the base nodes need to be synchronized.
- The messages sent should be timestamped.
- The positions of the base nodes should be known.
- Suffers from Non-Line-of-Sight (NLOS) conditions.

Positioning

Recalling the model for the measurement:

$$b_i = h_i(\mathbf{q}) + w_i.$$

Saying w to be negligible, the model can be given by:

$$h_i(\mathbf{q}) \overset{\text{Distance}}{=} \rho_i = (t_i - t)c = \sqrt{(\mathbf{q} - \mathbf{s}_i)^T(\mathbf{q} - \mathbf{s}_i)},$$

where q is the position to be found, s_i is the position of the i-th anchor, and ρ_i is the distance between the i-th anchor and the tag.

$$\mathbf{q} = \begin{bmatrix} x \\ y \end{bmatrix}, \quad \mathbf{s}_i = \begin{bmatrix} x_i \\ y_i \end{bmatrix}.$$

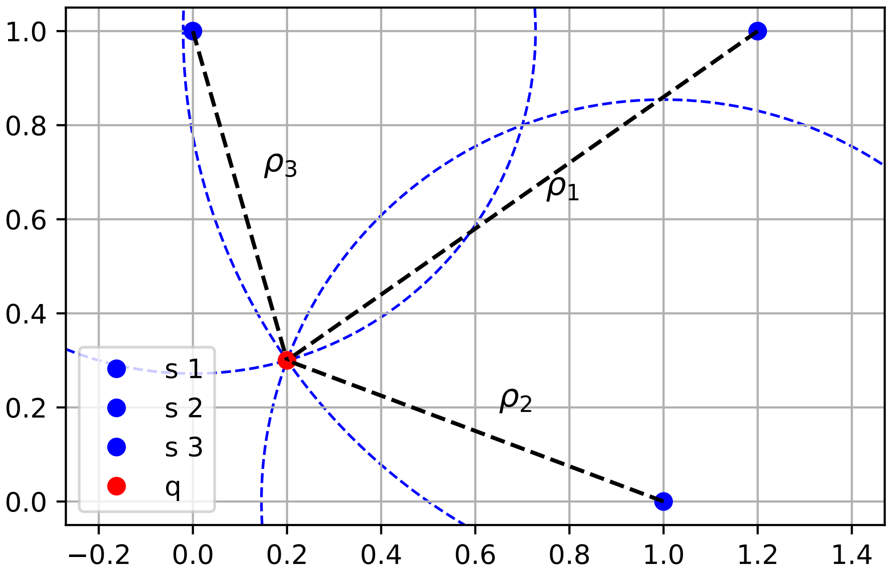
Positioning

It is now evident that the possible positions of the tag can be geometrically interpreted as a **circle**. Indeed, we immediately have:

$$\rho_i^2 = (x - x_i)^2 + (y - y_i)^2. \quad \text{Formula cerchio}$$

Moreover, with simple algebraic observations, it turns out that **at least three** base stations are needed.

Display of the positioning problem



Positioning

Since we have m base stations, we have the vectorial representation:

$$\mathbf{b}_m = h(q) + \mathbf{w}_m,$$

where,

$$\mathbf{b}_m = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{bmatrix}, \quad h(\mathbf{q}) = \rho = \begin{bmatrix} h_1(\mathbf{q}) \\ h_2(\mathbf{q}) \\ \vdots \\ h_m(\mathbf{q}) \end{bmatrix}, \quad \mathbf{w}_m = \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_m \end{bmatrix}.$$

Static Estimator

The objective of an estimator is to retrieve the estimate $\hat{q}$ and possibly a measure of its uncertainty. For the ToA approach, static estimators can be:

- **Nonlinear Weighted Least Squares** minimizing:

We don't make any kind of assumption on noise/unceartanty

$$J(\hat{q}) = \sum_{i=1}^{m} \frac{(\overset{\text{Actual measure}}{b_i} - \overset{\text{Residual part}}{\sqrt{(\hat{x} - x_i)^2 + (\hat{y} - y_i)^2}})^2}{\sigma_i^2 \text{Variance of each measure}};$$

- **Maximum Likelihood** , which turns out to be a Nonlinear Weighted Least Squares in the case of Gaussian noises;
- **Least Squares** on a linearized version of the measurements.

Static Estimator

For the latter case, let us consider the squares of the measurements:

$$b_i^2 = (\hat{x} - x_i)^2 + (\hat{y} - y_i)^2.$$

expanding the squares, we have:

$$b_i^2 = \hat{x}^2 - 2x_i\hat{x} + x_i^2 + \hat{y}^2 - 2y_i\hat{y} + y_i^2.$$

Static Estimator

By defining $r = x^2 + y^2$, we can rewrite the previous expression as:

$$b_i^2 - x_i^2 - y_i^2 = r - 2xx_i - 2yy_i + \eta_i. \quad \text{Linearized version of the model}$$

By defining:

Rewriting in vectorial formula

$$\mathbf{b}^*_m = \begin{bmatrix} b_1^2 - x_1^2 - y_1^2 \\ b_2^2 - x_2^2 - y_2^2 \\ \vdots \\ b_m^2 - x_m^2 - y_m^2 \end{bmatrix}, \quad \theta = \begin{bmatrix} x \\ y \\ r \end{bmatrix}, \quad \mathbf{H} = \begin{bmatrix} -2x_1 & -2y_1 & 1 \\ -2x_2 & -2y_2 & 1 \\ \vdots & \vdots & \vdots \\ -2x_m & -2y_m & 1 \end{bmatrix},$$

we can rewrite the previous expression as:

$$\mathbf{b}^*_m = \mathbf{H}\theta.$$

Static Estimator

Notice that in the previous example both the estimates $\hat{\mathbf{q}}$ and $\hat{r}$ are derived. Since $\hat{r} = \hat{\mathbf{q}}^T\hat{\mathbf{q}}$, this constraint should be enforced and hence **constrained Least Squares** solutions should be applied.

Alternatively, we can get rid of r by computing $b_i^2 - b_j^2$, i.e.,

$$b_i^2 - b_j^2 - x_i^2 + x_j^2 - y_i^2 + y_j^2 = -2(x_i - x_j)x - 2(y_i - y_j)y + \eta_i - \eta_j.$$

As in the previous case, we can derive the matrix formulation for this problem as well, which, again, requires just the **Least Squares** to be computed.

Solution using Linear Algebra

Using linear algebra is possible to find the solution of the **Least Square** inverting the matrix H

$$\mathbf{b} = \mathbf{H}\mathbf{p} \quad \rightarrow \quad \mathbf{p} = \mathbf{H}^{-1}\mathbf{b}.$$

This is true if the matrix **H** is square and full rank. If the matrix is rectangular, the pseudo-inverse should be computed instead of the inverse with the Moore-Penrose formula $\mathbf{H}^+ = (\mathbf{H}^T\mathbf{H})^{-1}\mathbf{H}^T$.

Using the trick of the previous slide we can rewrite the matrix b and H for the case with 3 anchors as

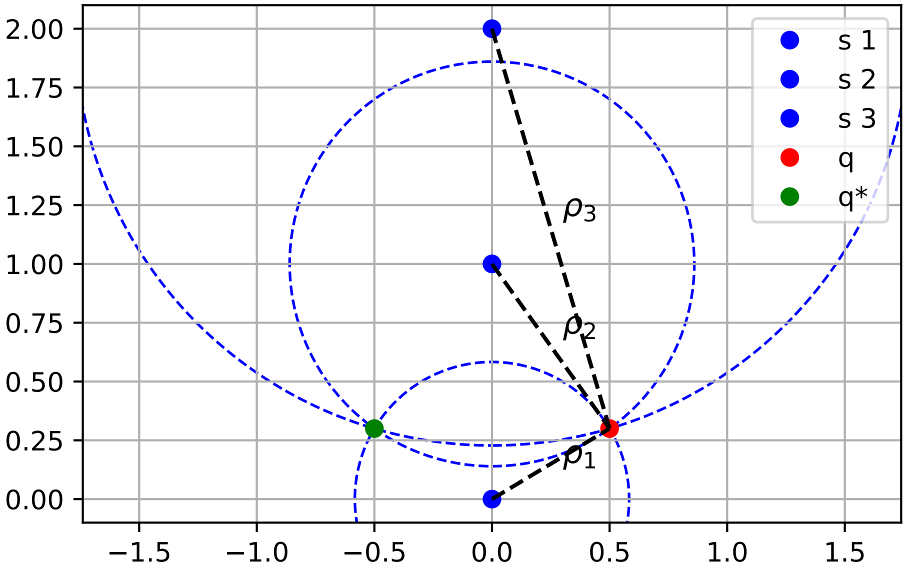
$$\bar{\mathbf{b}} = \begin{bmatrix} b_1^2 - b_2^2 - x_1^2 + x_2^2 - y_1^2 + y_2^2 \\ b_2^2 - b_3^2 - x_2^2 + x_3^2 - y_2^2 + y_3^2 \end{bmatrix}, \quad \bar{\mathbf{H}} = \begin{bmatrix} -2x_1 + 2x_2 & -2y_1 + 2y_2 \\ -2x_2 + 2x_3 & -2y_2 + 2y_3 \end{bmatrix}.$$

So the solution can be computed as:

$$\bar{\mathbf{b}} = \bar{\mathbf{H}}\mathbf{p} \quad \rightarrow \quad \mathbf{p} = \bar{\mathbf{H}}^{-1}\bar{\mathbf{b}}.$$

What if the rank of the matrix is not full?

If the matrix is not full rank, the solution is not unique. It can see geometrically that when the anchors are aligned, the matrix is not full rank. In this case, the solution is not unique and the problem is ill-posed.



Time Difference of Arrival

We introduce now the Time Difference of Arrival (TDoA). As for the ToA, the TDoA comprises the target node and a set of base stations, whose positions are assumed to be known.

TDoA, as ToA, is an active technique and the position is either computed by the element being localised or by the base stations. Similarly, in coplanar cases, just three base stations are needed, otherwise four such nodes are needed.

Basically ToA and TDoA share the same configuration. So, what are the differences?

Time Difference of Arrival (TDoA) Model

Let us again model the TDoA for the planar case. We consider the position to be determined $\mathbf{q} = [x, y]^T$ and $\mathbf{s}_i = [x_i, y_i]^T$ the positions of $i = 1, \dots, m$ base stations. The target node emits a signal at time t that is received by the i-th base station at time t_i.

The distance ρ_i is again given by:

$$\rho_i = (t_i - t)c,$$

Difference with ToA

but now we assume that t is unknown to the base stations. However, considering the difference of times when two base stations receive the message, then:

$$t_i - t_j = (t_i - t) - (t_j - t) = \frac{\rho_i}{c} - \frac{\rho_j}{c},$$

where $t_i - t_j = \delta_{ji}$ is the measured TDoA.

Geometric Interpretation of TDoA

It is now evident why the TDoA can be **geometrically interpreted** as the locus of all points (x, y) such that the difference of the distances from $\mathbf{s}_i$ and $\mathbf{s}_j$ is constant and equal to ρ_{ji}. Hence, it turns out that at least three base stations are needed for the planar case.

Measurement Model

Recalling the measurement model:

Bij is the difference between the two measurements

$$b_{ji} = h_{ji}(q) + w_{ji},$$

it is evident that $b_{ji} = \rho_{ji}$ (distance between the base station i and the base station j), $w_{ji} \sim \mathcal{N}(0, \sigma_{ji}^2)$, and the measurement model can be given by:

$$h_{ji}(q) = \rho_{ji} = \sqrt{(q - s_i)^T(q - s_i)} - \sqrt{(q - s_j)^T(q - s_j)},$$

since $\mathbf{q} = [x, y]^T$ and $\mathbf{s}_i = [x_i, y_i]^T$ and $\mathbf{s}_j = [x_j, y_j]^T$ are the positions of the i-th and j-th base stations.

Positioning

Since we have m base stations, we have the vectorial representation:

$$\mathbf{b}_{m-1} = h(q) + \mathbf{w}_{m-1},$$

where, as in the previous cases,

$$\mathbf{b}_{m-1} = \begin{bmatrix} b_{12} \\ b_{13} \\ \vdots \\ b_{1m} \end{bmatrix}, \quad h(\mathbf{q}) = \rho = \begin{bmatrix} h_{12}(\mathbf{q}) \\ h_{13}(\mathbf{q}) \\ \vdots \\ h_{1m}(\mathbf{q}) \end{bmatrix}, \quad \mathbf{w}_{m-1} = \begin{bmatrix} w_{12} \\ w_{13} \\ \vdots \\ w_{1m} \end{bmatrix}.$$

Static Estimator

The objective of an estimator is to retrieve the estimate $\hat{q}$ and possibly a measure of its uncertainty. For the TDoA approach, static estimators can be:

- **Nonlinear Weighted Least Squares**;
- **Maximum Likelihood** , which turns out to be a Nonlinear Weighted Least Squares in the case of Gaussian noises;
- **Least Squares** on a linearized version of the measurements.

Static Estimator

For the latter case:

Distance between base-station 1 and base-station i

$$b_{1i} = \sqrt{(q - si)^T(q - s_i)} - \sqrt{(q - s_1)^T(q - s_1)} + w_{1i} \Rightarrow$$

$$b_{1i} + \sqrt{(q - s_1)^T(q - s_1)} = \sqrt{(q - si)^T(q - s_i)} + w_{1i} \Rightarrow$$

$$b_{1i} + \rho_1 = \rho_i + w_{1i}$$

computing the squares of the measurements, we have:

$$b_{1i}^2 + \rho_1^2 + 2b_1\rho_1 = \rho_i^2 + \eta_i$$

Static Estimator

We can then rewrite:

$$b_{1i}^2 + x_1^2 - x_i^2 + y_1^2 - y_i^2 = 2(x_1 - x_i)x + 2(y_1 - y_i)y - 2b_{1i}\rho_1 + \eta_i$$

Considering m base stations, we can define

$$\mathbf{b}_m^* = \begin{bmatrix} b_{12}^2 + x_1^2 - x_2^2 + y_1^2 - y_2^2 \\ b_{13}^2 + x_1^2 - x_3^2 + y_1^2 - y_3^2 \\ \vdots \\ b_{1m}^2 + x_1^2 - x_m^2 + y_1^2 - y_m^2 \end{bmatrix}, \quad \theta = \begin{bmatrix} x \\ y \\ \rho_1 \end{bmatrix}, \quad \eta = \begin{bmatrix} \eta_2 \\ \vdots \\ \eta_m \end{bmatrix} \quad \mathbf{H} = 2 \begin{bmatrix} x_1 - x_2 & y_1 - y_2 & -b12 \\ x_1 - x_3 & y_1 - y_3 & -b13 \\ \vdots & \vdots & \vdots \\ x_1 - x_m & y_1 - y_m & -b1m \end{bmatrix}.$$

Static Estimator

We can rewrite the previous set of equations as

$$\mathbf{b}_{m*} = \mathbf{H}\theta + \eta$$

hence the Least Squares solution can be applied. Notiche that in the previous example both the estimate $\hat{q}$ and ρ_1 are derived. Since $\rho_1 = \sqrt{(\hat{q} - s_1)^T(\hat{q} - s_1)}$, this constraint should be enforced and hence **constrained LS** solutions should be applied.

Advantages and Drawbacks of TDoA

TDoA has the following advantages with respect to ToA:

- Only the base nodes need to be synchronized, i.e., the target node can be unsynchronized.
- The messages sent are not timestamped.

TDoA still has the following drawbacks:

- The positions of the base nodes should be known.
- Suffers from Non-Line-of-Sight (NLOS) conditions.

Angle of Arrival (AoA)

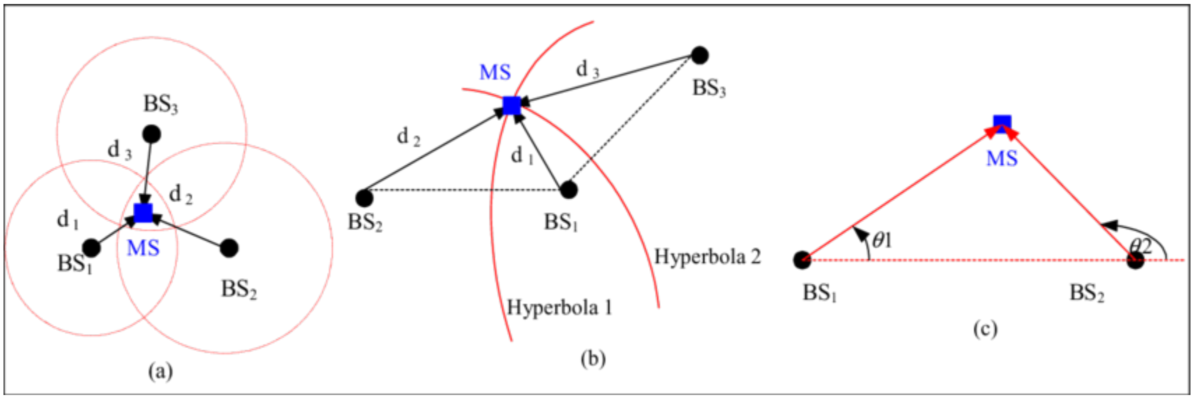
It is used to estimate the position of the target node mobile station by measuring the angle at which the signal arrives at multiple receiving base stations. The i-th base station measures the angle from which the signal emitted by the target node is received. If we know the locations of the base stations then the target node's position can be determined by finding the intersection of the angle lines from each base station. Measuring the angle of arrival of the signal, each base station can determine the line of bearing that extends in the direction in which the signal was received. The intersection of two directional lines of bearing with angles θ_i and θ_j defines a unique position, each formed by a radial from a base station to the target node. In the planar case, at least two base stations are needed.

Drawbacks of AoA

AoA has some drawbacks:

- The positions of the base nodes should be known;
- Suffers from Non-Line-of-Sight (NLOS) conditions;
- It requires antenna arrays for each base station.

Geometric Interpretation Comparison



(a) Time of Flight (ToF),
(b) Time Difference of Arrival (TDoA),
(c) Angle of Arrival (AoA).

Ultra-Wideband (UWB) Technology

Ultra-wide band (UWB) refers to the wireless technology that can access the frequency spectrum larger than 500 MHz, usually in the range 3 to 10 GHz.

UWB has unique advantages, including wall penetration capability, low transmission power, simple transceiver structure, and high temporal as well as spatial resolutions. These features potentially allow centimeter-level positioning accuracy with high precision.

We are not going into the details of the solution, what is of interest to us are the positioning techniques that can be adopted.

Ultra-Wideband (UWB) Technology

Usually, UWB is adopted for ToA or TDoA approaches. UWB uses two different communication techniques:

- Impulse Radio: transmission of pulses that occupy the entire bandwidth.

- Multi-band Orthogonal Frequency Division Multiplexing: the spectrum is used to transmit several symbols on several sub-bands.

Given the different technique, the algorithm to extract the timing information changes, but for our purposes nothing changes.

Localisation

The problem of **positioning** thus analysed refers to state estimation with a **static estimator**, i.e., when the state is estimated using a single shot of the available measurements. When this condition does not hold but the problem is observable, a **dynamic estimator** can be still designed using explicitly the system dynamic. An example is the Kalman Filter (i.e., Bayesian approach): using the prior knowledge on the system dynamics, it is possible to determine the state of the system, i.e., to locate the object of interest with respect to a known reference frame. This problem takes the name of **localisation** problem.

Dynamic Estimator

Unicycle Model

- The unicycle model represents a mobile robot on a plane.
- Typical example: differential robot with two lateral wheels.
- Nonholonomic constraint: movement is only allowed in the direction of the orientation.

State Coordinates and Control Variables

When trying to estimate the position of a moving target, the problems become more complex. An additional term should be considered in the model, i.e., the orientation of the target. The new coordinates are $\mathbf{q} = [x, y, \theta]^T$, where:

- x, y are the position of the target.
- θ is the orientation of the target.

The control variable of the system are:

- v the velocity of the target.
- ω the angular velocity of the target.

The dynamics of the system can be given by:

$$\begin{aligned} \dot{x} &= v \cos(\theta), \\ \dot{y} &= v \sin(\theta), \\ \dot{\theta} &= \omega. \end{aligned}$$

Discretizing the Model

To simulate numerically, we discretize with time step Δt:

$$\begin{cases} x_{k+1} = x_k + v_k \cos\theta_k \, \Delta t \\ y_{k+1} = y_k + v_k \sin\theta_k \, \Delta t \\ \theta_{k+1} = \theta_k + \omega_k \, \Delta t \end{cases}$$

Relative Localization Example:

"WhereAreYou: an UWB relative tracking system for pedestrian using only ranging information" Luca Santoro, Matteo Nardello, Davide Brunelli, Daniele Fontanelli

Relative-localization problem

Localization of the nodes of a network with respect to each other in a local reference frame, without a fixed external and global reference frame.

What are the information available?

- Distance between the master and the anchors
- Distance between the anchors

Given the points

$$\mathbf{N} = \begin{bmatrix} \mathbf{N}_0 & \cdots & \mathbf{N}_n \end{bmatrix} = \begin{bmatrix} x_0 & \cdots & x_n \\ y_0 & \cdots & y_n \end{bmatrix},$$

where $\mathbf{N}_i$ is the position of the i-th anchor. Let us assume that the i-th node has access to the distances

$$\rho_{i,j} = \|\mathbf{N}_i - \mathbf{N}_j\| = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2},$$

so that the symmetric squared Euclidean distance matrix

$$\mathbf{D} = \begin{bmatrix} 0 & \rho_{0,1}^2 & \cdots & \rho_{0,n}^2 \\ \rho_{1,0}^2 & 0 & \cdots & \rho_{1,n}^2 \\ \vdots & \vdots & \ddots & \vdots \\ \rho_{n,0}^2 & \rho_{n,1}^2 & \cdots & 0 \end{bmatrix}$$

Encodes all the informations that we have
,

can be built. **This is the matrix that we have access to.**

How do we extract the points form the distances?

Using the double centering matrix

$$\mathbf{H} = \mathbf{I}_{n+1} - \frac{1}{n+1}\mathbf{e}\mathbf{e}^T,$$

where $\mathbf{e}\mathbf{e}^T = \mathbf{1}_{n+1}\mathbf{1}_{n+1}^T$, $\mathbf{1}_{n+1}$ is a column vector filled with $n+1$ ones and $\mathbf{I}_{n+1}$ is the identity matrix of dimension $n+1 \times n+1$, to transform $\mathbf{D}$, we obtain the Gram matrix

$$\mathbf{G} = -\frac{1}{2}\mathbf{HDH}.$$

This transformation turns pairwise Euclidean distances into pairwise inner products of vectors.

Let us define $\mathbf{P} = \begin{bmatrix} \mathbf{p}_0 & \mathbf{p}_1 & \cdots & \mathbf{p}_n \end{bmatrix}^T$ as the matrix of node coordinates that generates the symmetric Euclidean matrix $\mathbf{D}$, and that is a replica of $\mathbf{N}$ but affected by geometric ambiguities.

Gram Matrix and Its Properties

The Gram matrix $\mathbf{G}$ has a special structure: if $\mathbf{G}$ is positive semi-definite, it can be decomposed into the product of coordinates (or vectors) that represent the points. Specifically, if $\mathbf{G} = \mathbf{P}\mathbf{P}^T$, where $\mathbf{P}$ contains the coordinates of the points we're trying to find, then $\mathbf{G}$ preserves the pairwise distances in the following sense:

$$\mathbf{G}_{ij} = \langle \mathbf{p}_i, \mathbf{p}_j \rangle = \mathbf{p}_i^T \mathbf{p}_j,$$

This means that each entry $\mathbf{G}_{ij}$ in the Gram matrix corresponds to the inner product of the coordinates of points $\mathbf{p}_i$ and $\mathbf{p}_j$. Since inner products determine the relative geometry of points, reconstructing $\mathbf{P}$ will preserve these distances up to an affine transformation.

Eigen-Decomposition for Coordinate Recovery

To derive $\mathbf{P}$, the following optimization problem needs to be solved:

$$\arg\min_{\mathbf{P}} \|\mathbf{G} - \mathbf{P}\mathbf{P}^T\|^2.$$

The solution to this optimization problem is given by the eigen-decomposition, i.e.

$$\mathbf{P} = \begin{bmatrix} \mathbf{p}_0 & \cdots & \mathbf{p}_n \end{bmatrix} = \begin{bmatrix} \tilde{x}_0 & \cdots & \tilde{x}_n \\ \tilde{y}_0 & \cdots & \tilde{y}_n \end{bmatrix} = \mathbf{U}\sqrt{\mathbf{V}},$$

where $\mathbf{V}$ is the diagonal matrix of the eigenvalues, and $\mathbf{U}$ is the eigenvector matrix of $\mathbf{G}$.

Specifically:

- $\mathbf{G} = \mathbf{U}\mathbf{V}\mathbf{U}^T$, where $\mathbf{U}$ is the matrix of eigenvectors and $\mathbf{V}$ is the diagonal matrix of eigenvalues.
- By taking $\mathbf{P} = \mathbf{U}\sqrt{\mathbf{V}}$, we construct coordinates that satisfy $\mathbf{G} \approx \mathbf{P}\mathbf{P}^T$, meaning that $\mathbf{P}$ will reproduce the distances in $\mathbf{D}$ up to rotations, reflections, or translations.

Geometric Ambiguity

The resulting points $\mathbf{P}$ are not exactly the same as the original points $\mathbf{N}$; they are affine transformations of $\mathbf{N}$, which means that they could be rotated, reflected, or translated versions of $\mathbf{N}$. This is because the Gram matrix $\mathbf{G}$ preserves distances, but it does not preserve absolute positioning or orientation in space. However, any two point sets $\mathbf{N}$ and $\mathbf{P}$ that produce the same $\mathbf{G}$ will have identical pairwise distances, which is the key requirement for applications relying on $\mathbf{D}$.

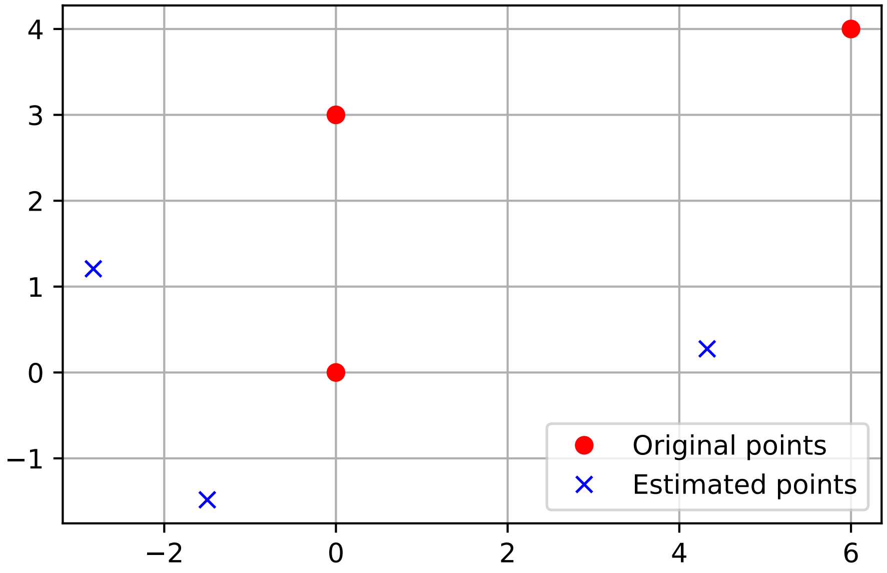
More precisely, if there exists an angle $\theta \neq 2k\pi$ with $k \in \mathbb{N}$ such that

$$\mathbf{N} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \mathbf{P} = \mathbf{R}(\theta)\mathbf{P},$$

then a rotation ambiguity occurs. The flipping problem takes place if

$$\mathbf{N} = \pm \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix} \mathbf{P} = \pm\mathbf{SP}.$$

Example



Return to the relative-localization problem

Until now we understood that given the distances is possible to reconstruct the position of the anchors in the relative frame up to a rotation, reflection, or translation. So we want to know if taking more measurements will help to solve the problem of localisation of the master in the relative frame centered on its estimated initial position.

Illustration of the problem

Position of the master and two anchors at time t_1.

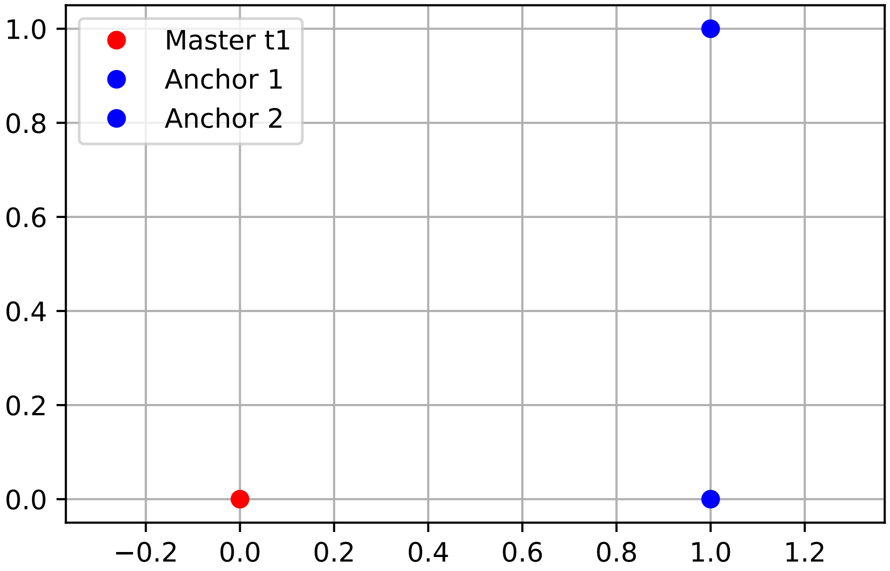


Illustration of the problem

Position of the master and two anchors at time t_2.

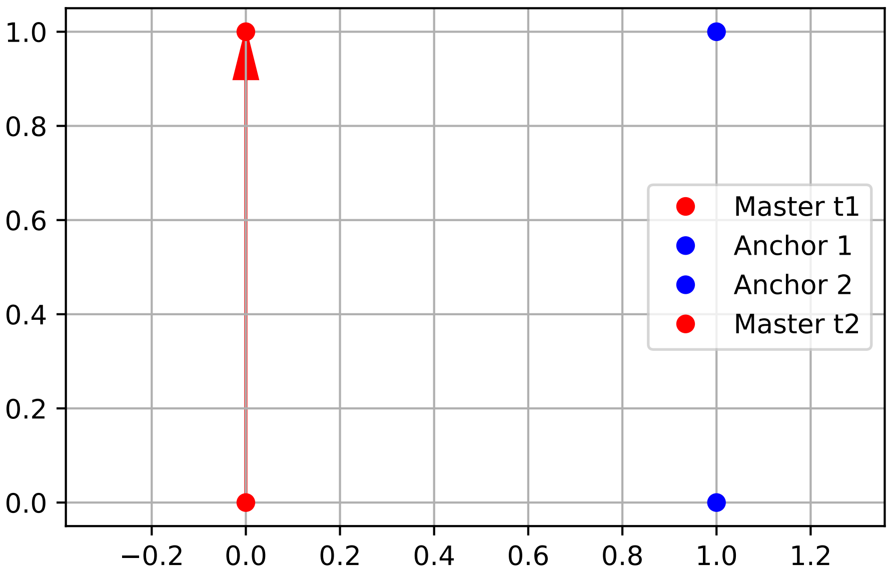
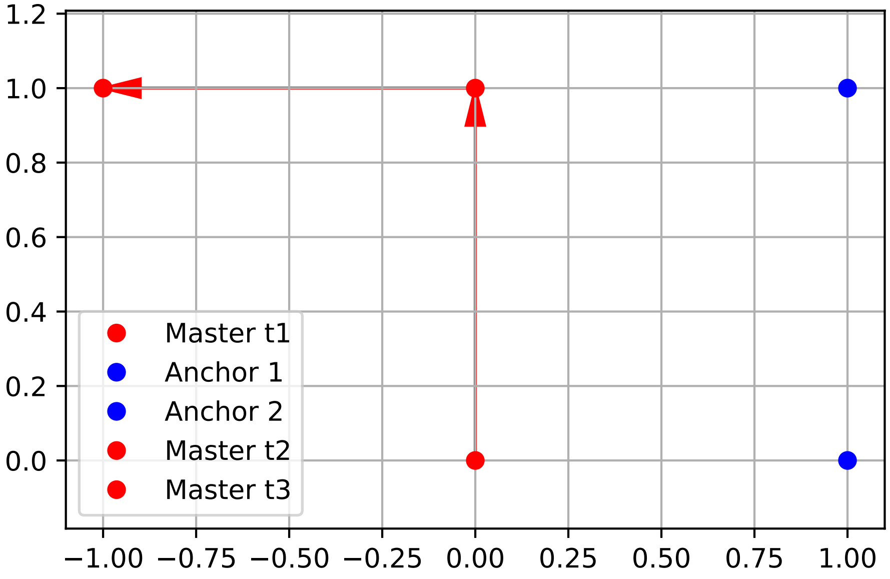


Illustration of the problem

Position of the master and two anchors at time t_3.



Moving Node Setup

Consider three time instants k, $k+1$, and $k+2$ where the moving node $\mathbf{N}_0$:

- Starts at position $\mathbf{N}_0^k$,
- Moves to $\mathbf{N}_0^{k+1} = \mathbf{N}_0^k + \mathbf{t}_k$, and
- Moves to $\mathbf{N}_0^{k+2} = \mathbf{N}_0^{k+1} + \mathbf{t}_{k+1}$.

Here, $\mathbf{t}_k = \begin{bmatrix} \Delta x_k \\ \Delta y_k \end{bmatrix}$ and $\mathbf{t}_{k+1} = \begin{bmatrix} \Delta x_{k+1} \\ \Delta y_{k+1} \end{bmatrix}$ are translation vectors.

Distance Matrix Construction

With measurements $\rho_{i,j} + \eta_{i,j}$ for three consecutive time instants:

- We build distance matrices $\mathbf{D}^k$, $\mathbf{D}^{k+1}$, and $\mathbf{D}^{k+2}$.
- Use these distance matrices to compute estimated positions $\hat{\mathbf{P}}^k$, $\hat{\mathbf{P}}^{k+1}$, and $\hat{\mathbf{P}}^{k+2}$ by solving the optimization problem:

$$\arg\min_{\mathbf{P}} \|\mathbf{G} - \mathbf{P}\mathbf{P}^T\|^2$$

Alignment with Roto-Translation

1. **Centering**:
   - First, center $\hat{\mathbf{P}}^k$ on the moving node 0:

$$\hat{\mathbf{P}}^k = \hat{\mathbf{P}}^k - \hat{\mathbf{p}}_{0,k}$$

2. **Roto-Translation Alignment**: Align $\hat{\mathbf{P}}^{k+1}$ with $\hat{\mathbf{P}}^k$ by solving:

$$\arg\min_{\theta,\mathbf{T}} \|\hat{\mathbf{P}}^k - \left(\mathbf{R}(\theta)\alpha\mathbf{S}\hat{\mathbf{P}}^{k+1} + \mathbf{T}\right)\|$$

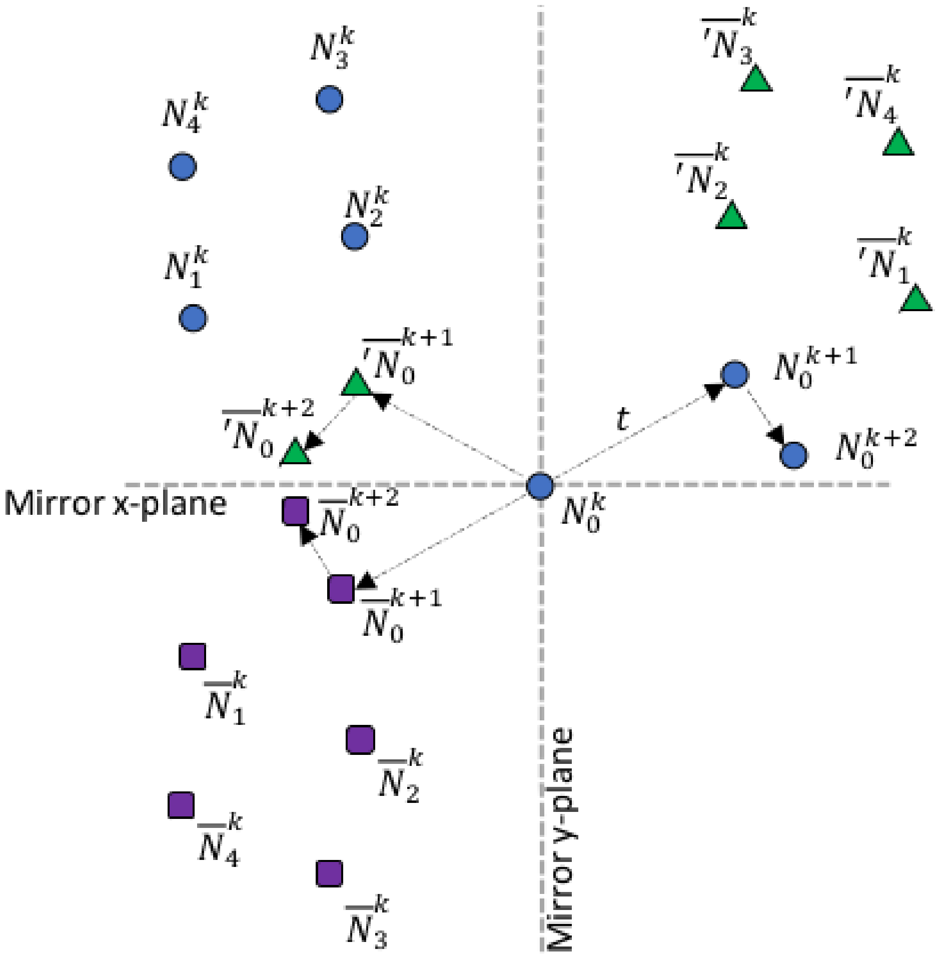
   Here, $\mathbf{R}(\theta)$ represents rotation by θ, and $\mathbf{T}$ is the translation vector.

Path Estimation

- After alignment, the estimated translation $\mathbf{T}$ provides an approximation of the actual displacement $\mathbf{t}_k$ in the opposite direction.
- This alignment can be extended from $\hat{\mathbf{P}}^k$ and $\hat{\mathbf{P}}^{k+1}$ to the next time step $\hat{\mathbf{P}}^{k+2}$, yielding the path of the moving node $\mathbf{N}_0$ in its relative frame centered on $\hat{\mathbf{p}}_{0,k}$.

Ambiguity in Flipping

- Although alignment provides the positions relative to $\mathbf{N}_0$, it does not fully solve the **flipping problem** modeled by $\alpha\mathbf{S}$.
- Due to the ambiguity, the points may appear as reflections across either the x-axis or y-axis.



Theorem: Flipping Ambiguity

Theorem: Given a set of $m > 0$ node 0 motions,

$$\mathbf{N}_0^{k+q} = \mathbf{N}_0^{k+q-1} + \mathbf{t}_{k+q-1}, \quad q = 1, \dots, m$$

it is impossible to determine $\alpha\mathbf{S}$ (flipping operation) if we have no additional knowledge about $\mathbf{t}_{k+q-1}$. —

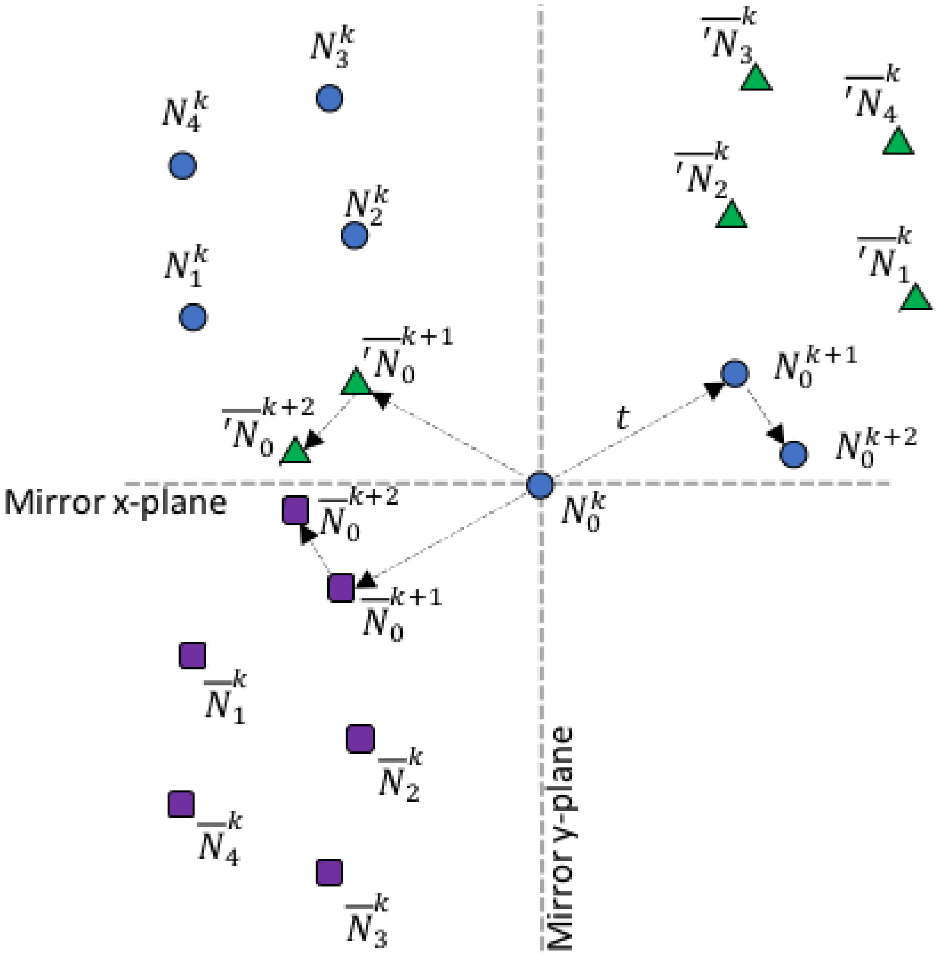
Resolving Flipping with Rotation Direction

Corollary: - By knowing the **sign of angle** $\beta = \arctan \frac{\Delta y_{k+1} - \Delta y_k}{\Delta x_{k+1} - \Delta x_k}$, we can resolve flipping. - The angle β gives the relative rotation direction of t_k to t_{k+1}, indicating whether the movement is clockwise or counterclockwise.

Flipping Resolution: Intuition

- If the moving node 0 has rotated in a specific direction (e.g., counterclockwise), a flipped solution would imply a contradictory movement (e.g., clockwise).
- By verifying the sign of β, we can remove incorrect solutions that do not match the actual direction of movement.

Alignment provides the positions to N0, it does not fully solve the flipping problme modeled by alphaS



Conclusion

- This method allows localization of the moving node N_0 within a relative frame centered on its initial estimated position.
- By incorporating direction of movement, we can resolve the remaining flipping ambiguity.