Probabilistic Robotics Course

EKF SLAM without data association

D. Schlegel
schlegel@diag.uniroma1.it

Department of Computer, Control and Management Engineering
Sapienza University of Rome

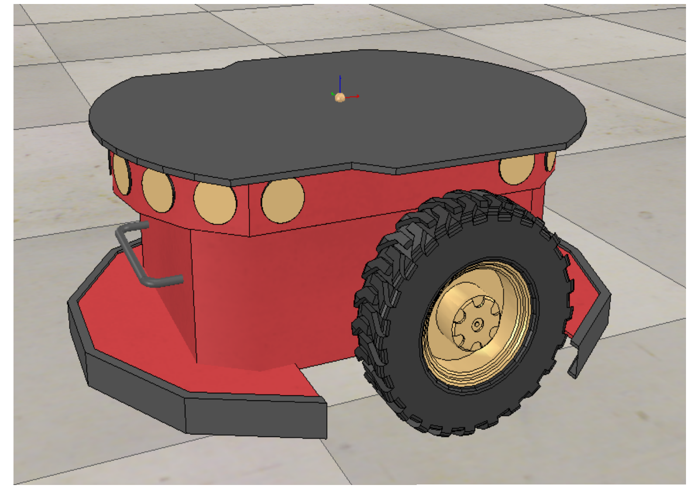
Outline

- Scenario
- Controls
- Observations
- Non-Linear Systems and Gaussian Noise
- Data Association

Scenario

Orazio moves on a 2D plane

- Is controlled by translational and rotational velocities
- Senses a set of **not distinguishable** landmarks through a “2D landmark sensors”
- The location of the landmarks in the world is **not known**



Approaching the problem

We want to develop a KF based algorithm to track the position of Orazio as it moves (localization) and, at the same time, the position of the observed plant-landmarks (mapping) while performing data association.

The inputs of our algorithms will be

- velocity measurements
- landmark measurements

We have no prior knowledge of the map.

Domains

Define

- state space

$$\mathbf{x}_t^r = \begin{pmatrix} x_t \\ y_t \\ \theta_t \end{pmatrix} \in \Re^3$$

poses mapped to 3D vectors

landmarks in the state

$$\mathbf{x}_t^{[n]} = \begin{pmatrix} x_t^{[n]} \\ y_t^{[n]} \end{pmatrix} \in \Re^2 \quad {}_{n=1..N}$$

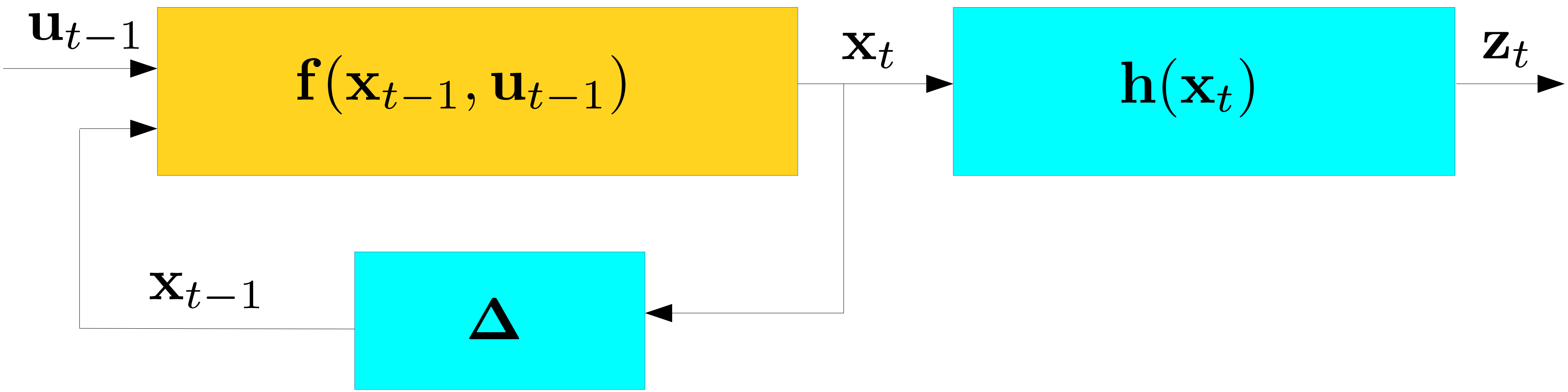
- space of controls (inputs)

$$\mathbf{u}_t = \begin{pmatrix} u_t^1 \\ u_t^2 \end{pmatrix} \in \Re^2$$

- space of observations (measurements)

$$\mathbf{z}_t^{[m]} = \begin{pmatrix} x_t^{[m]} \\ y_t^{[m]} \end{pmatrix} \in \Re^2 \quad {}_{m=1..M}$$

Transition Function

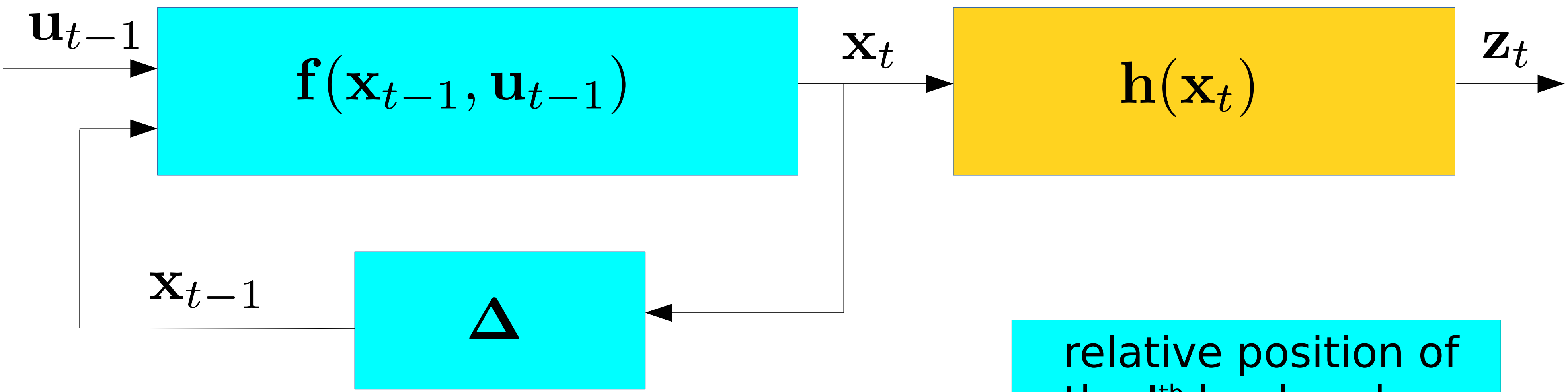


pose update like localization

the landmarks don't move

$$\mathbf{x}_t = \mathbf{f}(\mathbf{x}_{t-1}, \mathbf{u}_{t-1}) = \begin{pmatrix} x_{t-1} + u^1_{t-1} \cdot \cos(\theta_{t-1}) \\ y_{t-1} + u^1_{t-1} \cdot \sin(\theta_{t-1}) \\ \theta_{t-1} + u^2_{t-1} \\ \mathbf{x}^{[1]}_t \\ \mathbf{x}^{[2]}_t \\ \dots \\ \mathbf{x}^{[N]}_t \end{pmatrix}$$

Measurement Function



relative position of the l[th] landmark w.r.t the robot at time t

$$
\begin{aligned}
\mathbf{z}_t^{[n]} &= \mathbf{h}^{[n]}(\mathbf{x}_t) \\
&= \mathbf{R}_t^T(\mathbf{x}_t^{[n]} - \mathbf{t}_t) \\
&= \begin{pmatrix} \cos\theta_t(x_t^{[n]} - x_t) + \sin\theta_t(y_t^{[n]} - y_t) \\ -\sin\theta_t(x_t^{[n]} - x_t) + \cos\theta_t(y_t^{[n]} - y_t) \end{pmatrix}
\end{aligned}
$$

We have [i] measurement functions, one per landmark

Control Noise

We assume the velocity measurements are effected by a Gaussian noise resulting from the sum of two aspects

- a constant noise
- a velocity dependent term whose standard deviation grows with the speed
- translational and rotational noise are assumed independent

$$\mathbf{n}_{u,t} \sim \mathcal{N}\left(\mathbf{n}_{u,t}; \mathbf{0}, \begin{pmatrix} (u_t^1)^2 + \sigma_v^2 & 0 \\ 0 & (u_t^2)^2 + \sigma_\omega^2 \end{pmatrix}\right)$$

Measurement Noise

We assume it is zero mean, constant

$$\mathbf{n}_z \sim \mathcal{N}\left(\mathbf{n}_z; \mathbf{0}, \left(\begin{array}{cc} \sigma_z^2 & 0 \\ 0 & \sigma_z^2 \end{array} \right)\right)$$

Jacobian 1

$$\mathbf{f}(\mathbf{x}_{t-1}, \mathbf{u}_{t-1}) = \begin{pmatrix} x_{t-1} + u_{t-1}^1 \cdot \cos(\theta_{t-1}) \\ y_{t-1} + u_{t-1}^1 \cdot \sin(\theta_{t-1}) \\ \theta_{t-1} + u_{t-1}^2 \\ \mathbf{x}_t^{[1]} \\ \mathbf{x}_t^{[2]} \\ \dots \\ \mathbf{x}_t^{[N]} \end{pmatrix}$$

Our usual Jacobians:

$$\mathbf{A}_t = \frac{\partial \mathbf{f}(\cdot)}{\partial \mathbf{x}} = \begin{pmatrix} \frac{\partial \mathbf{f}(\cdot)}{\partial \mathbf{x}^r} & \frac{\partial \mathbf{f}(\cdot)}{\partial \mathbf{x}^{[1]}} & \frac{\partial \mathbf{f}(\cdot)}{\partial \mathbf{x}^{[2]}} & \dots & \frac{\partial \mathbf{f}(\cdot)}{\partial \mathbf{x}^N} \end{pmatrix}$$

$$\mathbf{B}_t = \frac{\partial \mathbf{f}(\cdot)}{\partial \mathbf{u}}$$

Jacobian 2

Our landmark sensor perceives points, thus our measurement function will be:

$$\mathbf{h}^{[n]}(\mathbf{x}_t) \quad = \quad \mathbf{R}_t^T(\mathbf{x}_t^{[n]} - \mathbf{t}_t)$$

Consequently, the Jacobian can be computed as:

$$\boxed{\frac{\partial \mathbf{h}^{[n]}(\cdot)}{\partial \mathbf{x}_t^r} = \begin{pmatrix} -\mathbf{R}_t^T & \frac{\partial \mathbf{R}_t^T}{\partial \theta_t}\left(\mathbf{x}_t^{[n]} - \mathbf{t}_t\right) \end{pmatrix}} \qquad \boxed{\frac{\partial \mathbf{h}^{[n]}(\cdot)}{\partial \mathbf{x}_t^{[n]}} = \mathbf{R}_t^T}$$

$$\mathbf{C}_t^{[n]} = \frac{\partial \mathbf{h}^{[n]}(\cdot)}{\partial \mathbf{x}_t} = \begin{pmatrix} \underbrace{\boxed{\frac{\partial \mathbf{h}^{[n]}}{\partial \mathbf{x}_t^r}}}_{\text{pose block}} & \mathbf{0} & \cdots & \underbrace{\boxed{\frac{\partial \mathbf{h}^{[n]}}{\partial \mathbf{x}_t^{[n]}}}}_{\text{landmark block}} & \cdots & \mathbf{0} \end{pmatrix}$$

Data Association

We do **not** observe the landmark ids.

When a new landmark appears, it's our duty to assign a unique id.

For convenience, we can keep unchanged the state-id mapping structure seen in the previous lesson

$$\mathbf{id\_to\_state\_map} = \begin{pmatrix} -1 & -1 & \dots & \dots & -1 & \end{pmatrix}$$

$$\mathbf{state\_to\_id\_map} = \begin{pmatrix} -1 & -1 & \dots & \dots & -1 & \end{pmatrix}$$

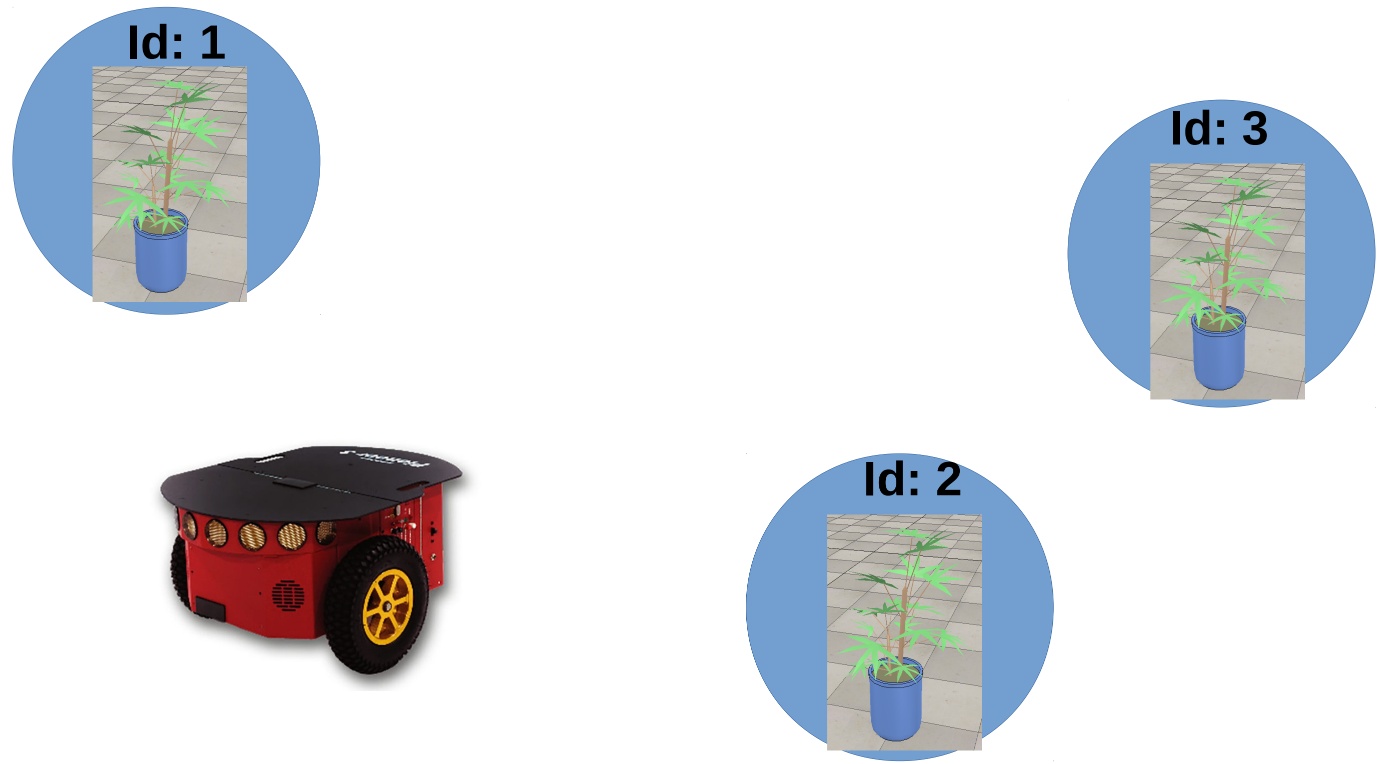
Data Association

Time: t

current mean

$$\mu_t = \begin{pmatrix} x_t^r \\ y_t^r \\ \theta_t^r \\ l_t^{[1]} \\ l_t^{[2]} \\ l_t^{[3]} \end{pmatrix}$$

The first time an unmatched landmark is seen, results in the creation and assignment of a new id

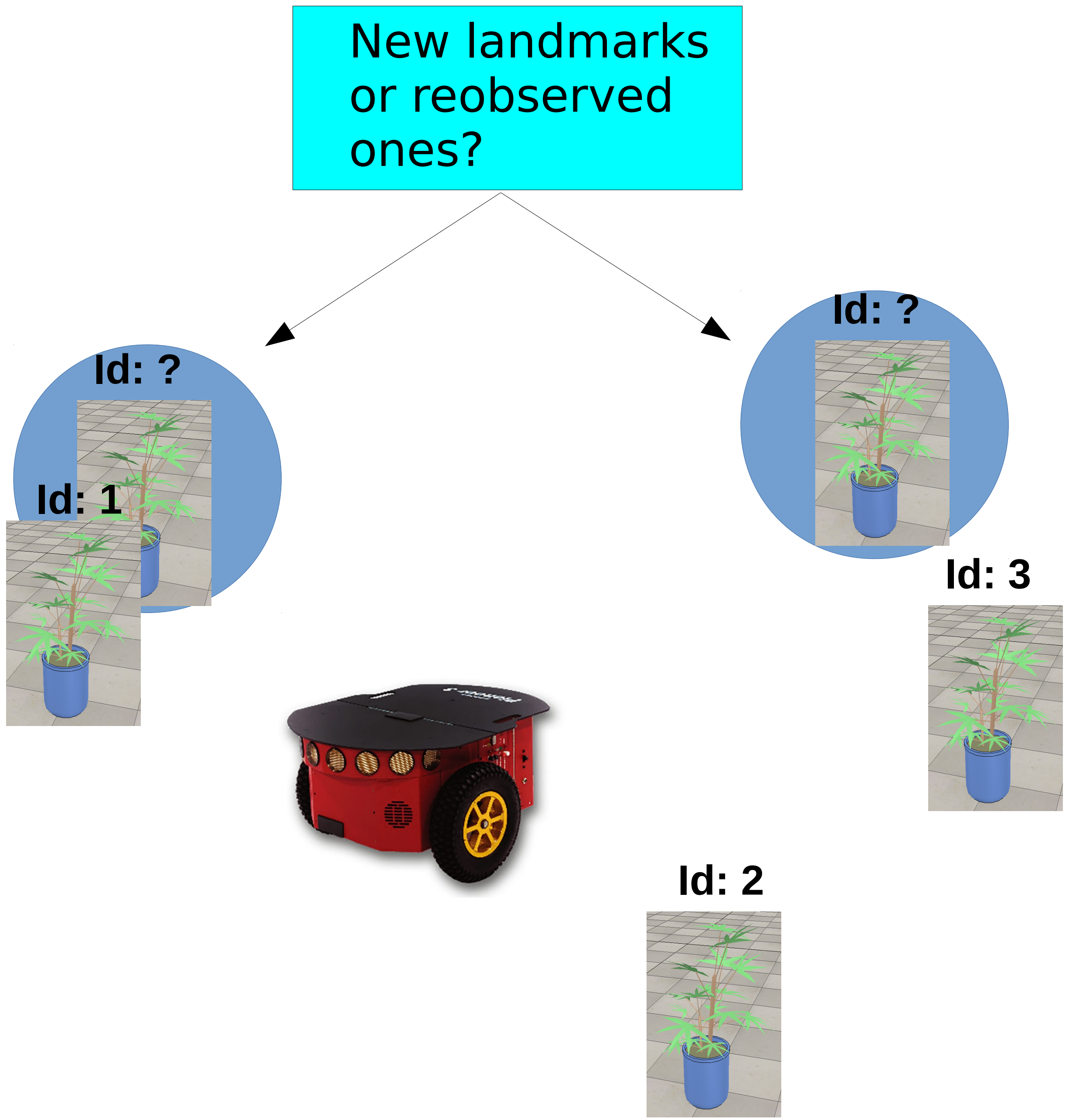


Data Association

Time: t+1

current mean

$$\mu_{t+1} = \begin{pmatrix} x^r_{t+1} \\ y^r_{t+1} \\ \theta^r_{t+1} \\ l^{[1]}_{t+1} \\ l^{[2]}_{t+1} \\ l^{[3]}_{t+1} \\ ?? \end{pmatrix}$$



Data Association

At each time step, precompute the likelihood for each landmark/measurement pair:

$$a_{mn} = (\mathbf{z}^{[m]} - \mu_z^n)^T \Sigma_{m,m}^{-1} (\mathbf{z}^{[m]} - \mu_z^n)$$

and assemble them in a cost matrix

$$\mathbf{A} = \begin{pmatrix} a_{11} & a_{12} & a_{13} & \cdots a_{1N} \\ a_{21} & a_{22} & a_{23} & \cdots a_{2N} \\ \vdots & & & \\ a_{M1} & a_{M2} & a_{M3} & \cdots a_{MN} \end{pmatrix}$$

Gating

$$\mathbf{A} = \begin{pmatrix} a_{11} & a_{12} & a_{13} & \cdots a_{1N} \\ a_{21} & a_{22} & a_{23} & \cdots a_{2N} \\ \vdots & & & \\ a_{M1} & a_{M2} & a_{M3} & \cdots a_{MN} \end{pmatrix}$$

- Choose a threshold τ_{accept}
- Extract the minimum for each row a_{mn}
- *If* $a_{mn} < \tau_{accept}$
  - *then* observation m is associated with landmark n
  - *otherwise*, m is a new landmark.

Multiple measurements can be assigned to the same landmark

Best Friends

$$\mathbf{A} = \begin{pmatrix} a_{11} & a_{12} & a_{13} & \cdots a_{1N} \\ a_{21} & a_{22} & a_{23} & \cdots a_{2N} \\ \vdots & & & \\ a_{M1} & a_{M2} & a_{M3} & \cdots a_{MN} \end{pmatrix}$$

- Take all the accepted associations from gating and check
- *If* $a_{mn} = \min_m a_{mn}$ AND $a_{mn} = \min_n a_{mn}$
  - *then* keep the association
  - o*therwise* discard it

Lonely Best Friends

$$\mathbf{A} = \begin{pmatrix} a_{11} & a_{12} & a_{13} & \cdots a_{1N} \\ a_{21} & a_{22} & a_{23} & \cdots a_{2N} \\ \vdots & & & \\ a_{M1} & a_{M2} & a_{M3} & \cdots a_{MN} \end{pmatrix}$$

- Define a smaller threshold γ and, for all surviving associations, extract the second best association for measurements $\hat{a}_n$ and landmarks $\hat{a}_m$ and check
- *If* $a_{mn} - \hat{a}_m > \gamma \text{ AND } a_{mn} - \hat{a}_n > \gamma$
  - *then* keep the association
  - *otherwise* discard it

Hands On!