4. The Scaling Problem

INDEX

ROBOTICS
UNIVERSIDAD DE ZARAGOZA

1. Limitations of the basic EKF SLAM algorithm

The scaling problem

The EKF SLAM algorithm

Algorithm 1 SLAM:

$\mathbf{x}_0^B = \mathbf{0}$; $\mathbf{P}_0^B = \mathbf{0}$ $\{Map\ initialization\}$

$[\mathbf{z}_0, \mathbf{R}_0] = \text{get\_measurements}$

$[\mathbf{x}_0^B, \mathbf{P}_0^B] = \text{add\_new\_features}(\mathbf{x}_0^B, \mathbf{P}_0^B, \mathbf{z}_0, \mathbf{R}_0)$

for $k = 1$ to steps **do**

$[\mathbf{x}_{R_k}^{R_{k-1}}, \mathbf{Q}_k] = \text{get\_odometry}$

$[\mathbf{x}_{k|k-1}^B, \mathbf{P}_{k|k-1}^B] = \text{EKF\_prediction}(\mathbf{x}_{k-1}^B, \mathbf{P}_{k-1}^B, \mathbf{x}_{R_k}^{R_{k-1}}, \mathbf{Q}_k)$

$[\mathbf{z}_k, \mathbf{R}_k] = \text{get\_measurements}$

$\mathcal{H}_k = \text{data\_association}(\mathbf{x}_{k|k-1}^B, \mathbf{P}_{k|k-1}^B, \mathbf{z}_k, \mathbf{R}_k)$

$[\mathbf{x}_k^B, \mathbf{P}_k^B] = \text{EKF\_update}(\mathbf{x}_{k|k-1}^B, \mathbf{P}_{k|k-1}^B, \mathbf{z}_k, \mathbf{R}_k, \mathcal{H}_k)$

$[\mathbf{x}_k^B, \mathbf{P}_k^B] = \text{add\_new\_features}(\mathbf{x}_k^B, \mathbf{P}_k^B, \mathbf{z}_k, \mathbf{R}_k, \mathcal{H}_k)$

end for

The prediction step

EKF SLAM prediction

$$\hat{\mathbf{x}}^B_{k|k-1} = \begin{bmatrix} \hat{\mathbf{x}}^B_{R_{k-1}} \oplus \hat{\mathbf{x}}^{R_{k-1}}_{R_k} \\ \hat{\mathbf{x}}^B_{F_{1,k-1}} \\ \vdots \\ \hat{\mathbf{x}}^B_{F_{m,k-1}} \end{bmatrix}$$

$$\mathbf{P}^B_{k|k-1} \simeq \mathbf{F}_k \mathbf{P}^B_{k-1} \mathbf{F}_k^T + \mathbf{G}_k \mathbf{Q}_k \mathbf{G}_k^T$$

$$\mathbf{F}_k = \begin{bmatrix} \mathbf{J}_{1\oplus}\left\{\hat{\mathbf{x}}^B_{R_{k-1}}, \hat{\mathbf{x}}^{R_{k-1}}_{R_k}\right\} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{I} & & \vdots \\ \vdots & & \ddots & \\ \mathbf{0} & \cdots & & \mathbf{I} \end{bmatrix} \; ; \; \mathbf{G}_k = \begin{bmatrix} \mathbf{J}_{2\oplus}\left\{\hat{\mathbf{x}}^B_{R_{k-1}}, \hat{\mathbf{x}}^{R_{k-1}}_{R_k}\right\} \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \end{bmatrix}$$

EKF SLAM prediction

$$\mathbf{P}^{B}_{k-1|k-1} = \begin{pmatrix} \mathbf{P}_R & \mathbf{P}_{RF_1} & \dots & \mathbf{P}_{RF_n} \\ \mathbf{P}^T_{RF_1} & \mathbf{P}_{F_1} & \dots & \mathbf{P}_{F_1F_n} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{P}^T_{RF_n} & \mathbf{P}^T_{F_1F_n} & \dots & \mathbf{P}_{F_n} \end{pmatrix}$$

$$\mathbf{P}^{B}_{k|k-1} = \begin{pmatrix} \mathbf{J}_{1\oplus}\mathbf{P}_R\mathbf{J}^T_{1\oplus} + \mathbf{J}_{2\oplus}\mathbf{Q}_k\mathbf{J}^T_{2\oplus} & \mathbf{J}_{1\oplus}\mathbf{P}_{RF_1} & \dots & \mathbf{J}_{1\oplus}\mathbf{P}_{RF_n} \\ \mathbf{J}^T_{1\oplus}\mathbf{P}^T_{RF_1} & \mathbf{P}_{F_1} & \dots & \mathbf{P}_{F_1F_n} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{J}^T_{1\oplus}\mathbf{P}^T_{RF_n} & \mathbf{P}^T_{F_1F_n} & \dots & \mathbf{P}_{F_n} \end{pmatrix}$$

EKF prediction is *O(n)*

Adding new features

EKF SLAM: add new features

$$\mathbf{x}_k^B = \begin{pmatrix} \mathbf{x}_{R_k}^B \\ \mathbf{x}_{F_{1,k}}^B \\ \vdots \\ \mathbf{x}_{F_{n,k}}^B \end{pmatrix} \Rightarrow \mathbf{x}_{k+}^B = \begin{pmatrix} \mathbf{x}_{R_k}^B \\ \mathbf{x}_{F_{1,k}}^B \\ \vdots \\ \mathbf{x}_{F_{n,k}}^B \\ \mathbf{x}_{F_{n+1,k}}^B \end{pmatrix} = \begin{pmatrix} \mathbf{x}_{R_k}^B \\ \mathbf{x}_{F_{1,k}}^B \\ \vdots \\ \mathbf{x}_{F_{n,k}}^B \\ \mathbf{x}_{R_k}^B \oplus \mathbf{z}_i \end{pmatrix}$$

Linearization:

$$\mathbf{x}_{k+}^B \simeq \hat{\mathbf{x}}_{k+}^B + \mathbf{F}_k(\mathbf{x}_k^B - \hat{\mathbf{x}}_k^B) + \mathbf{G}_k(\mathbf{z}_i - \hat{\mathbf{z}}_i)$$

$$\mathbf{P}_{k+}^B = \mathbf{F}_k \mathbf{P}_k^B \mathbf{F}_k^T + \mathbf{G}_k \mathbf{R}_k \mathbf{G}_k^T$$

Where:

$$\mathbf{F}_k = \frac{\partial \mathbf{x}_{k+}^B}{\partial \mathbf{x}_k^B} = \begin{pmatrix} \mathbf{I} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \ddots & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{I} \\ \mathbf{J}_{1\oplus}\left\{\hat{\mathbf{x}}_{R_k}^B, \hat{\mathbf{z}}_i\right\} & \mathbf{0} & \cdots & \mathbf{0} \end{pmatrix}; \mathbf{G}_k = \frac{\partial \mathbf{x}_{k+}^B}{\partial \mathbf{z}_i} = \begin{pmatrix} \mathbf{0} \\ \vdots \\ \mathbf{0} \\ \mathbf{J}_{2\oplus}\left\{\hat{\mathbf{x}}_{R_k}^B, \hat{\mathbf{z}}_i\right\} \end{pmatrix}$$

EKF SLAM: add new features

$$\mathbf{P}_k^W = \begin{pmatrix} \mathbf{P}_R & \mathbf{P}_{RF_1} & \cdots & \mathbf{P}_{RF_n} \\ \mathbf{P}_{RF_1}^T & \mathbf{P}_{F_1} & \cdots & \mathbf{P}_{F_1F_n} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{P}_{RF_n}^T & \mathbf{P}_{F_1F_n}^T & \cdots & \mathbf{P}_{F_n} \end{pmatrix}$$

$$\mathbf{P}_{k+}^W = \begin{pmatrix} \mathbf{P}_R & \mathbf{P}_{RF_1} & \cdots & \mathbf{P}_{RF_n} & \mathbf{P}_R\mathbf{J}_{1\oplus}^T \\ \mathbf{P}_{RF_1}^T & \mathbf{P}_{F_1} & \cdots & \mathbf{P}_{F_1F_n} & \mathbf{P}_{RF_1}^T\mathbf{J}_{1\oplus}^T \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \mathbf{P}_{RF_n}^T & \mathbf{P}_{F_1F_n}^T & \cdots & \mathbf{P}_{F_n} & \mathbf{P}_{RF_n}^T\mathbf{J}_{1\oplus}^T \\ \mathbf{J}_{1\oplus}\mathbf{P}_R & \mathbf{J}_{1\oplus}\mathbf{P}_{RF_1} & \cdots & \mathbf{J}_{1\oplus}\mathbf{P}_{RF_n} & \mathbf{J}_{1\oplus}\mathbf{P}_R\mathbf{J}_{1\oplus}^T + \mathbf{J}_{2\oplus}\mathbf{R}_k\mathbf{J}_{2\oplus}^T \end{pmatrix}$$

Adding new features is *O(n)*

The update step

EKF SLAM: map update

m observations:

$$\mathbf{z}_k = \mathbf{h}_k(\mathbf{x}^B_{\mathcal{F}_k}) + \mathbf{w}_k$$

$$\mathbf{z}_k \simeq \mathbf{h}_k(\hat{\mathbf{x}}^B_{\mathcal{F}_{k|k-1}}) + \mathbf{H}_k(\mathbf{x}^B_{\mathcal{F}_k} - \hat{\mathbf{x}}^B_{\mathcal{F}_{k|k-1}})$$

$$\mathbf{H}_k = \left.\frac{\partial \mathbf{h}_k}{\partial \mathbf{x}^B_{\mathcal{F}_k}}\right|_{(\hat{\mathbf{x}}^B_{\mathcal{F}_{k|k-1}})}$$

Filter gain: $$\mathbf{K}_k = \mathbf{P}^B_{k|k-1}\mathbf{H}_k^T(\mathbf{H}_k\mathbf{P}^B_{k|k-1}\mathbf{H}_k^T + \mathbf{R}_k)^{-1}$$

State update: $$\hat{\mathbf{x}}^B_k = \hat{\mathbf{x}}^B_{k|k-1} + \mathbf{K}_k(\mathbf{z}_k - \mathbf{h}_k(\hat{\mathbf{x}}^B_{k|k-1}))$$

Covariance update: $$\mathbf{P}^B_k = (\mathbf{I} - \mathbf{K}_k\mathbf{H}_k)\mathbf{P}^B_{k|k-1}$$

The innovation matrix

$$\mathbf{S}_k = \mathbf{H}_k \mathbf{P}_{k|k-1} \mathbf{H}_k^T + \mathbf{R}_k$$

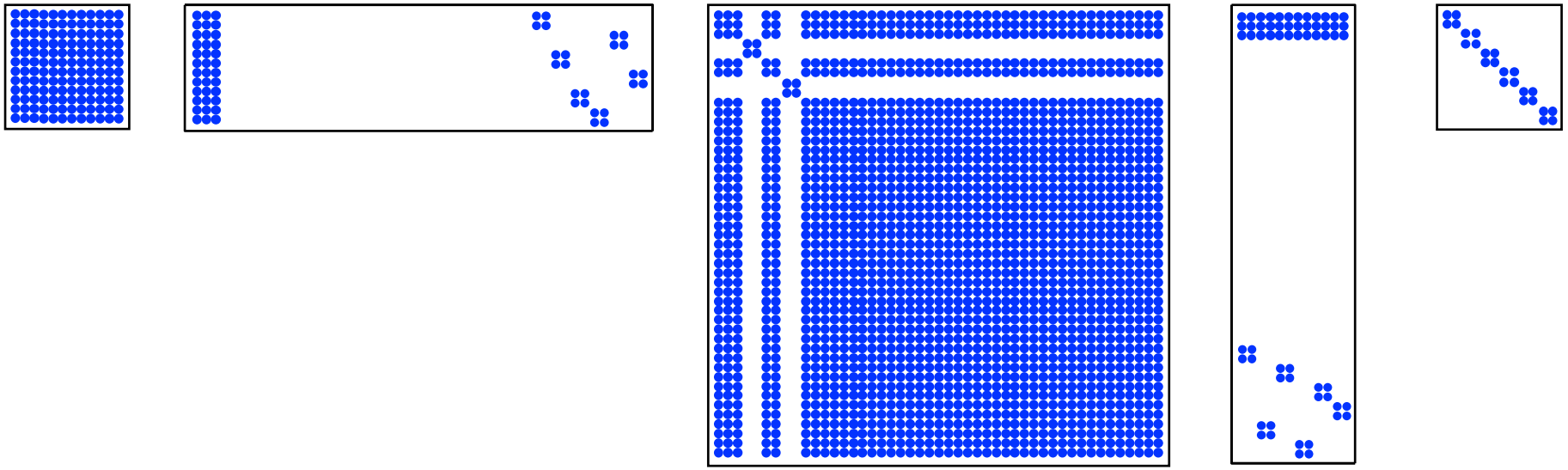
rxr *rxn* *nxn* *nxr* *rxr*

O(rn²) operations?

The innovation matrix

$$\mathbf{S}_k = \mathbf{H}_k \mathbf{P}_{k|k-1} \mathbf{H}_k^T + \mathbf{R}_k$$

rxr *rxt* *nxn* *txr* *rxr*

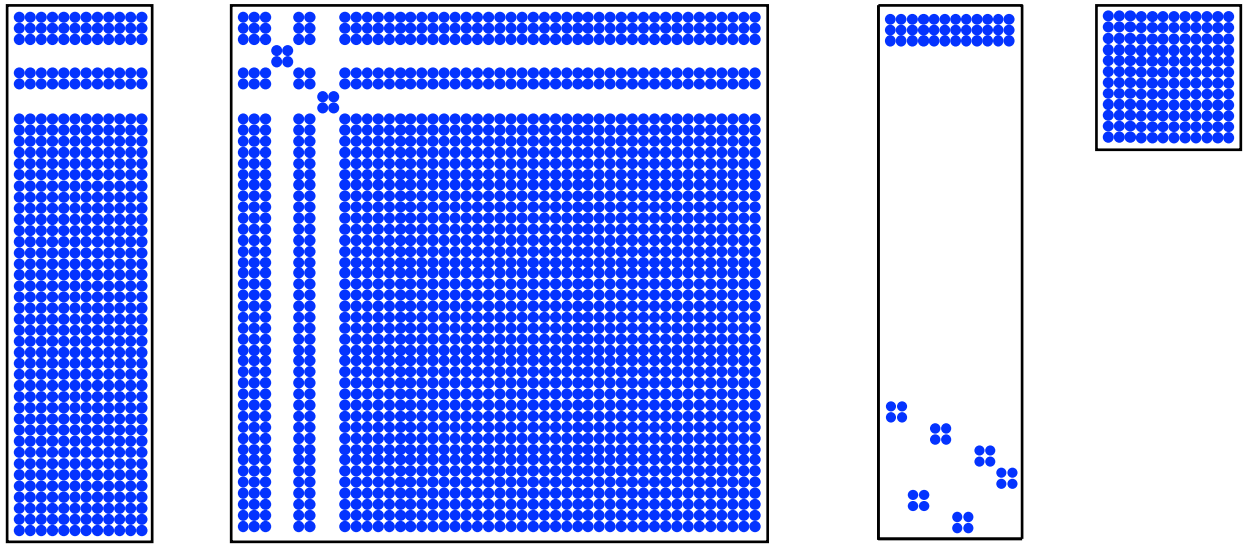


O(rn) = O(n) operations

The Kalman gain matrix

$$\mathbf{K}_k = \mathbf{P}_{k|k-1} \mathbf{H}_k^T (\mathbf{S}_k)^{-1}$$

nxr *nxn* *txr* *rxr*



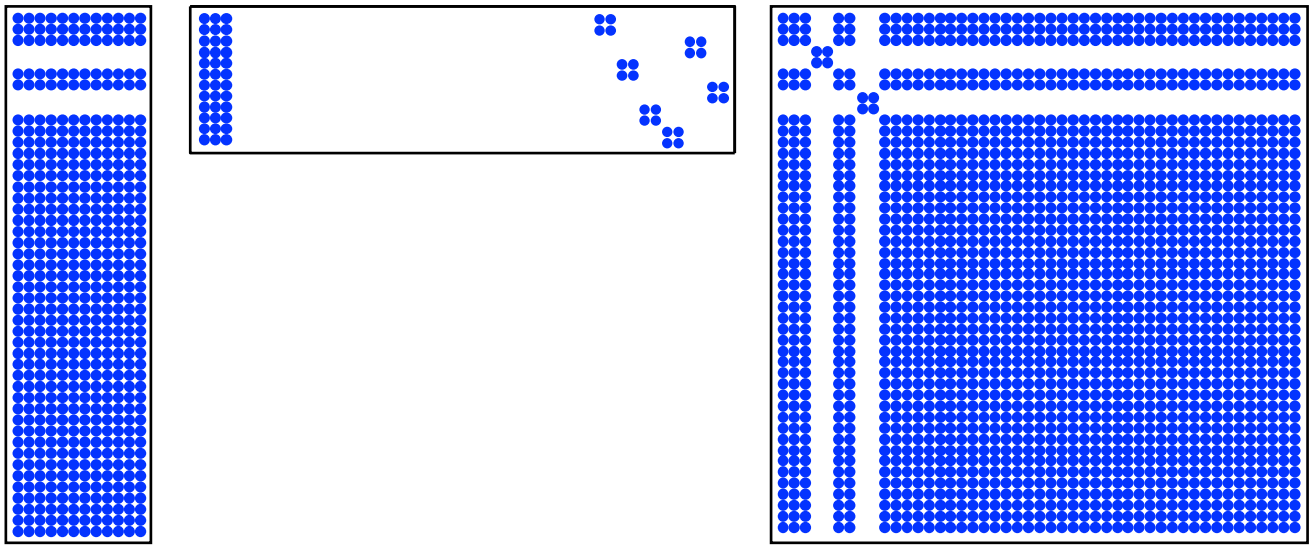
$O(r^2 n) = O(n)$ operations

The covariance matrix

$$\mathbf{P}_k = (\mathbf{I} - \mathbf{K}_k\mathbf{H}_k)\mathbf{P}_{k|k-1}$$

$$\ldots\ldots = \ldots\ldots \mathbf{K}_k \qquad \mathbf{H}_k \qquad \mathbf{P}_{k|k-1}$$

$$\ldots\ldots \qquad \ldots\ldots$$



nxr *rxt* *nxn*

$O(rn^2) = O(n^2)$ operations

Efforts to reduce complexity

- Decoupled Stochastic Mapping (Leonard and Feder, 2000) (Jensfelt 2001) ***O(1)***
- Local Mapping Algorithm (Chong and Kleeman 1999) ***O(1)***
- Suboptimal SLAM (Guivant and Nebot 2001) ***O(n)***
- Sparse Weight Filter (Julier 2001) ***O(n)***
- Sparse Extended Information Filter (Thrun et al 2003) ***O(1)***

Approximate, or pessimistic solutions

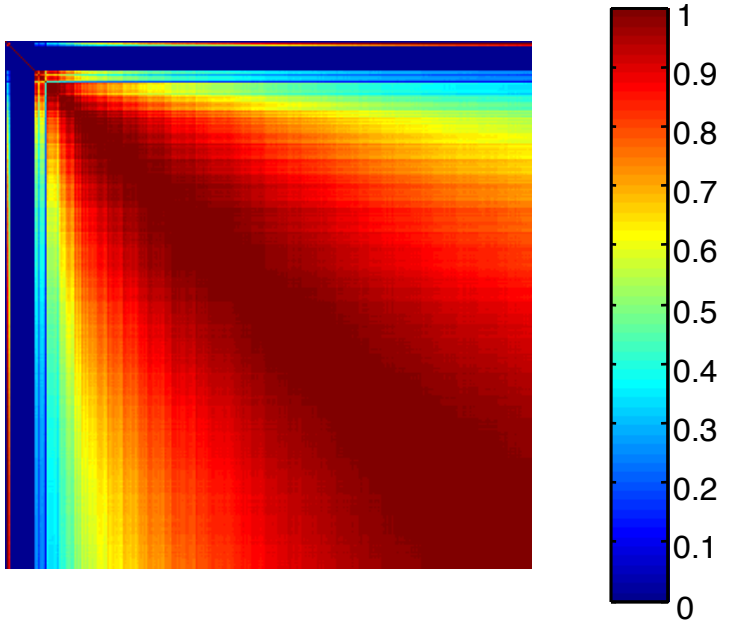
- Postponement (Davidson 1998, Knight, Davidson and Reed 2001)
- Compressed Filter (Guivant and Nebot 2001)
- Constrained Local Submap Filter (Williams 2001)
- Map Joining (Tardós et. al, 2002)

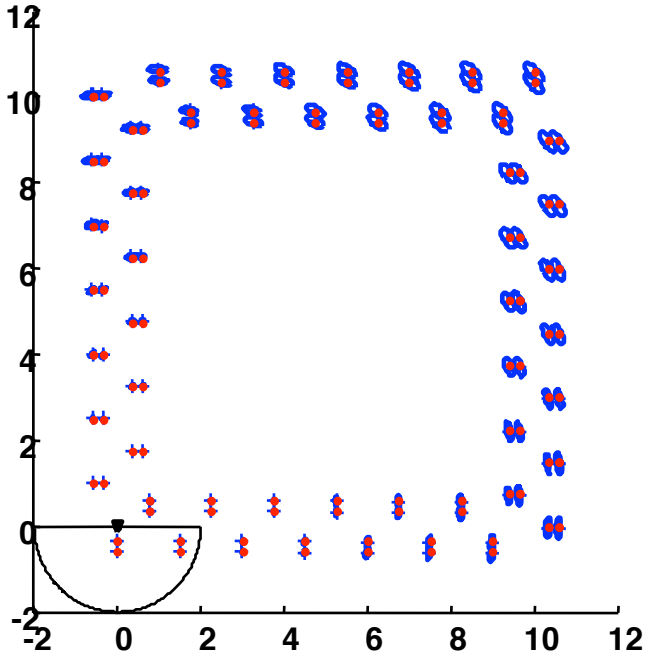
Exact solutions that delay global map updating, and strongly reduce cost.

But still $O(n^2)$

ROBOTICS
UNIVERSIDAD DE ZARAGOZA

Are correlations necessary?



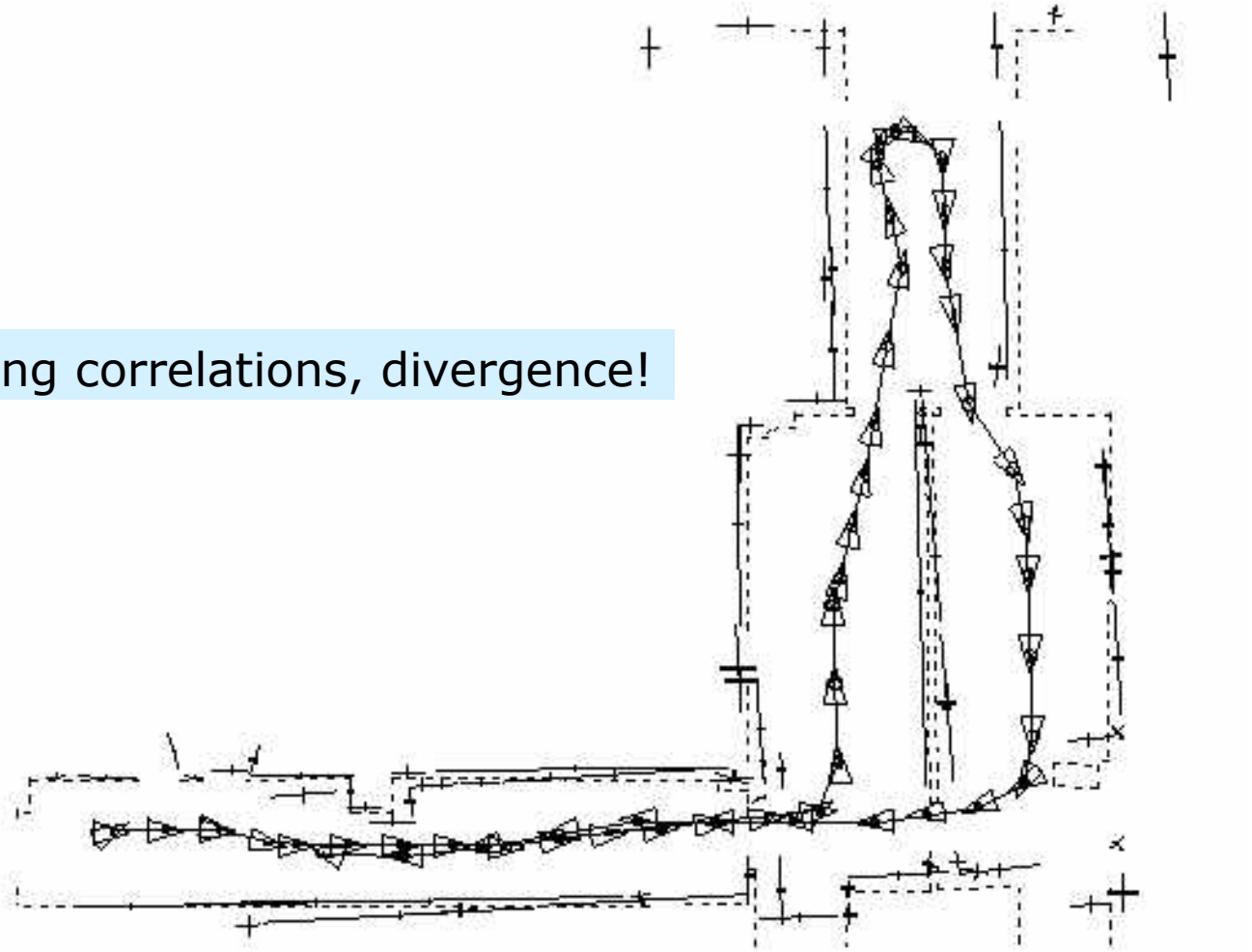



```
sigmas = sqrt(diag(Cov))' ;
Corr=diag(1./sigmas)*Cov*diag(1./sigmas) ;
```

ROBOTICS
UNIVERSIDAD DE ZARAGOZA

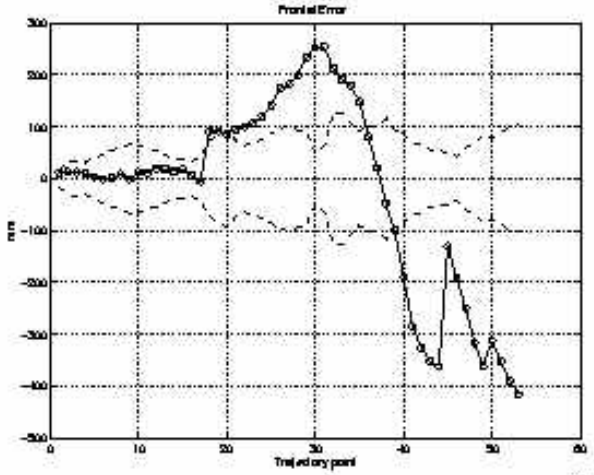
The importance of Correlations

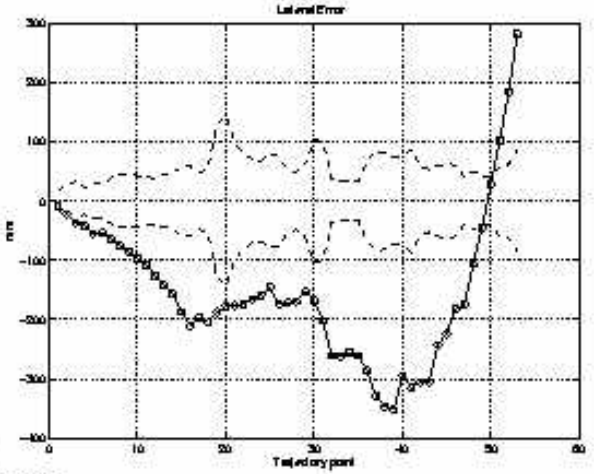
Ignoring correlations, divergence!



The importance of Correlations

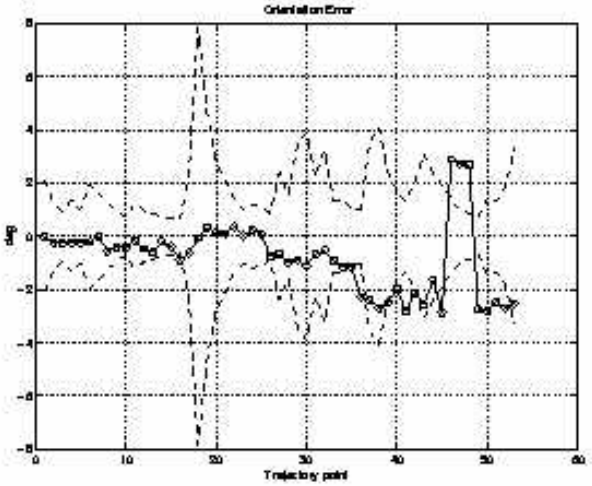
Frontal





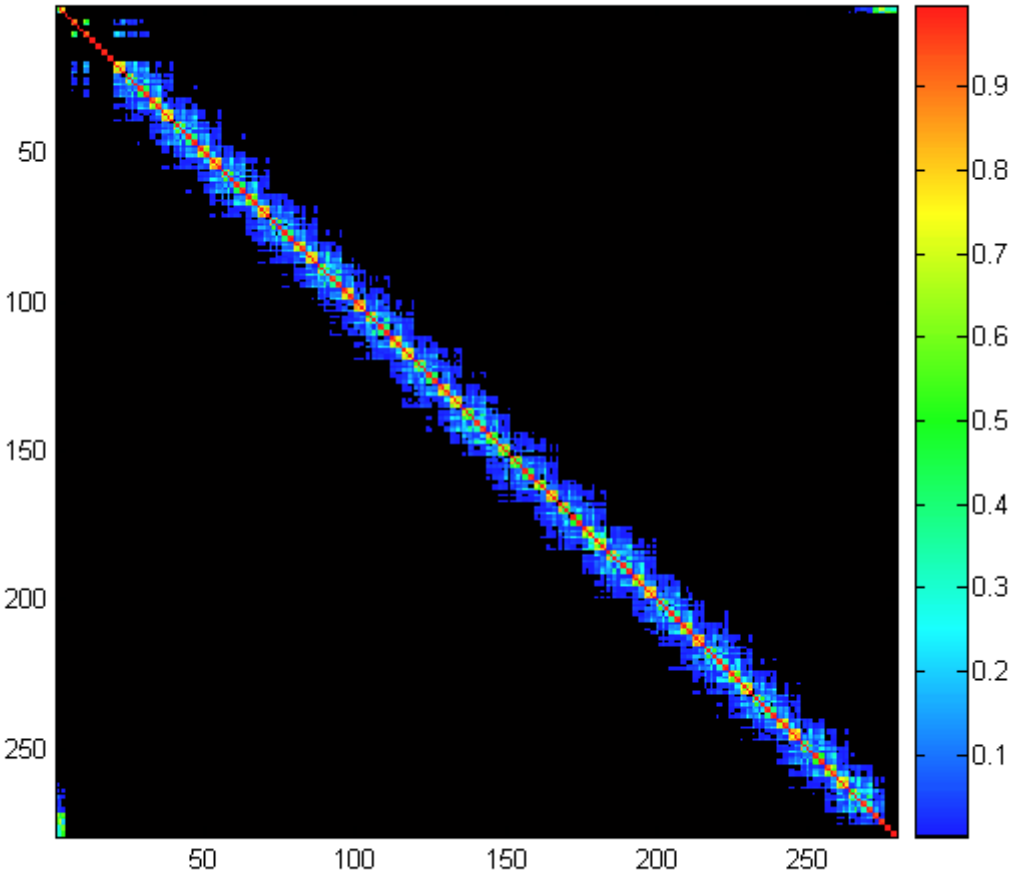
Lateral

Angular



ROBOTICS
UNIVERSIDAD DE ZARAGOZA

Inverse correlations



This observation is the basis of SEIFs

The linearization problem

Consistency of EKF-SLAM

- Nice **"convergence"** properties of $\mathbf{P}_k^W$ (Dissanayake et al. 2001):
  - Landmark covariance decreases monotonically
  - In the limit, landmarks become fully correlated
  - In the limit, landmark covariance reaches a lower bound related to the initial vehicle covariance

- But SLAM is a non-linear problem
  - The inherent approximations due to linearizations can lead to **divergence (inconsistency)** of the EKF
    - » see for example (Jazwinski, 1970)

EKF-SLAM: Robot Motion

$$\mathbf{x}_{R_k}^{B} = \mathbf{x}_{R_{k-1}}^{B} \oplus \mathbf{x}_{R_k}^{R_{k-1}}$$

Odometry model (white noise):

$$\begin{aligned} \mathbf{x}_{R_k}^{R_{k-1}} &= \hat{\mathbf{x}}_{R_k}^{R_{k-1}} + \mathbf{v}_k \\ E[\mathbf{v}_k] &= \mathbf{0} \\ E[\mathbf{v}_k \mathbf{v}_j^T] &= \delta_{kj} \mathbf{Q}_k \end{aligned}$$

EKF prediction:

$$\hat{\mathbf{x}}_{k|k-1}^{B} = \begin{bmatrix} \hat{\mathbf{x}}_{R_{k-1}}^{B} \oplus \hat{\mathbf{x}}_{R_k}^{R_{k-1}} \\ \hat{\mathbf{x}}_{F_{1,k-1}}^{B} \\ \vdots \\ \hat{\mathbf{x}}_{F_{m,k-1}}^{B} \end{bmatrix}$$

$$\mathbf{F}_k = \begin{bmatrix} \mathbf{J}_{1\oplus}\left\{\hat{\mathbf{x}}_{R_{k-1}}^{B}, \hat{\mathbf{x}}_{R_k}^{R_{k-1}}\right\} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{I} & & \vdots \\ \vdots & & \ddots & \\ \mathbf{0} & \cdots & & \mathbf{I} \end{bmatrix}$$

$$\mathbf{P}_{\mathcal{F}_{k|k-1}}^{B} = \mathbf{F}_k \mathbf{P}_{k-1}^{B} \mathbf{F}_k^T + \mathbf{G}_k \mathbf{Q}_k \mathbf{G}_k^T$$

$$\mathbf{G}_k = \begin{bmatrix} \mathbf{J}_{2\oplus}\left\{\hat{\mathbf{x}}_{R_{k-1}}^{B}, \hat{\mathbf{x}}_{R_k}^{R_{k-1}}\right\} \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \end{bmatrix}$$

Linearization

EKF-SLAM: Map Update

Feature observations:

$$\mathbf{z}_k = \mathbf{h}_k(\mathbf{x}_k^B) + \mathbf{w}_k$$

$$\mathbf{z}_k \simeq \mathbf{h}_k(\hat{\mathbf{x}}_{k|k-1}^B) + \mathbf{H}_k(\mathbf{x}_k^B - \hat{\mathbf{x}}_{k|k-1}^B)$$

$$\mathbf{H}_k = \left.\frac{\partial \mathbf{h}_k}{\partial \mathbf{x}_k^B}\right|_{(\hat{\mathbf{x}}_{k|k-1}^B)}$$

Linearization

EKF map update:

$$\hat{\mathbf{x}}_k^B = \hat{\mathbf{x}}_{k|k-1}^B + \mathbf{K}_k(\mathbf{z}_k - \mathbf{h}_k(\hat{\mathbf{x}}_{k|k-1}^B))$$

$$\mathbf{P}_k^B = (\mathbf{I} - \mathbf{K}_k\mathbf{H}_k)\mathbf{P}_{k|k-1}^B$$

$$\mathbf{K}_k = \mathbf{P}_{k|k-1}^B\mathbf{H}_k^T(\mathbf{H}_k\mathbf{P}_{k|k-1}^B\mathbf{H}_k^T + \mathbf{R}_k)^{-1}$$

The Consistency Problem

True map value

$$\mathbf{x}_k^W$$

EKF-SLAM estimation

$$\hat{\mathbf{x}}_k^W$$

$$\mathbf{P}_k^W$$

- An estimator is **consistent** if:

$$E\left[\mathbf{x}_k^W - \hat{\mathbf{x}}_k^W\right] = 0$$

$$E\left[\left(\mathbf{x}_k^W - \hat{\mathbf{x}}_k^W\right)\left(\mathbf{x}_k^W - \hat{\mathbf{x}}_k^W\right)^T\right] = \mathbf{P}_k^W$$

Unbiased

The Mean Square Error matches the filter computed Covariance

  - Pessimistic covariance is OK (but not too pessimistic)
  - Optimistic covariance = Inconsistency = Filter divergence

Consistency Testing

1. Normalized Estimation Error Squared NEES

$$D^2 = \left(\mathbf{x}_k^W - \hat{\mathbf{x}}_k^W\right)^T \left(\mathbf{P}_k^W\right)^{-1} \left(\mathbf{x}_k^W - \hat{\mathbf{x}}_k^W\right)$$

$$D^2 \leq \chi^2_{r,1-\alpha}$$

True map required → Simulations

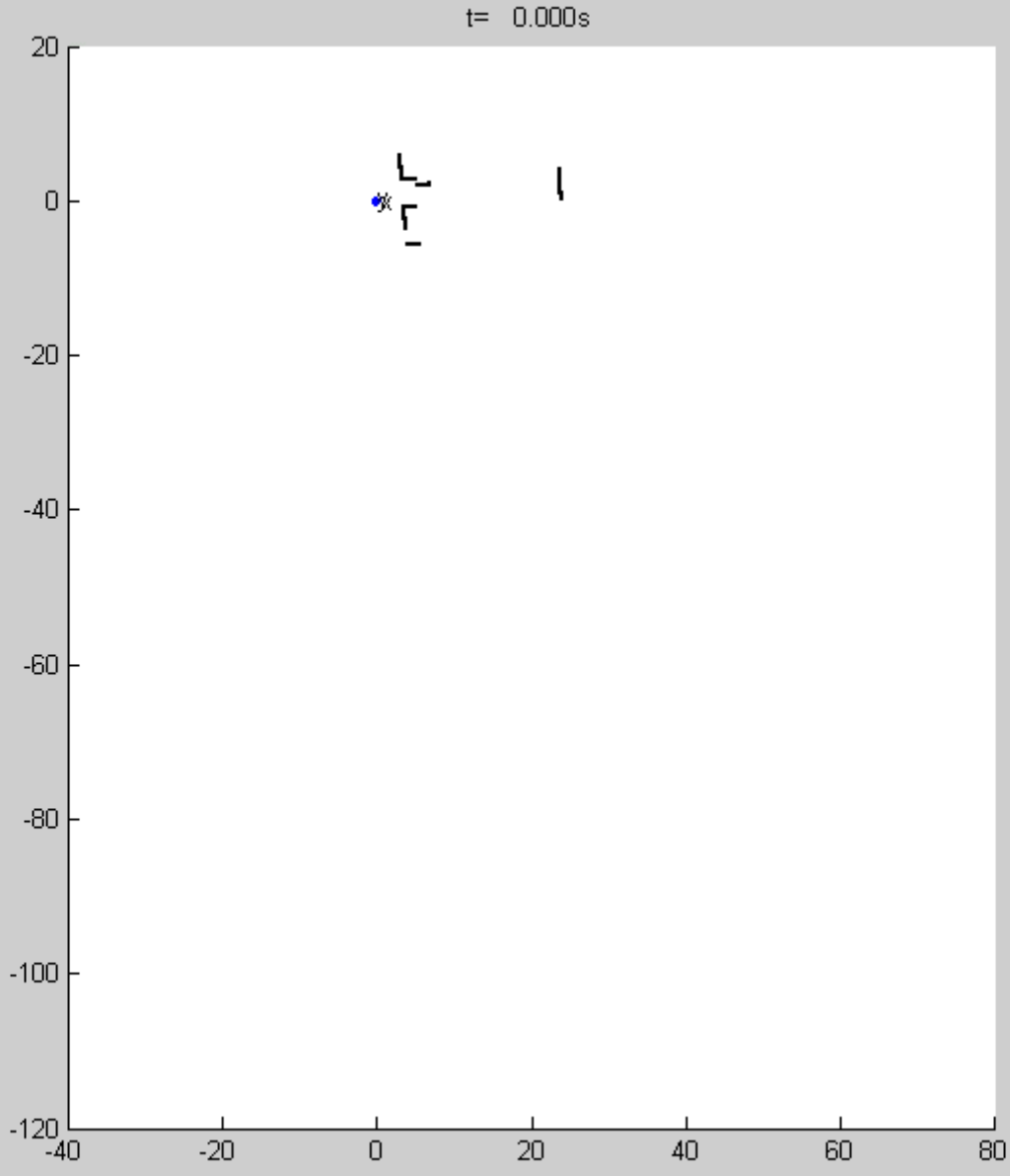
2. Innovation test (observation *i* → map feature *j*)

$$D_{ij}^2 = \left(\mathbf{z}_i - \mathbf{h}_j(\hat{\mathbf{x}}_k^W)\right)^T \left(\mathbf{H}_j \mathbf{P}_k^W \mathbf{H}_j^T + \mathbf{R}_i\right)^{-1} \left(\mathbf{z}_i - \mathbf{h}_j(\hat{\mathbf{x}}_k^W)\right)$$

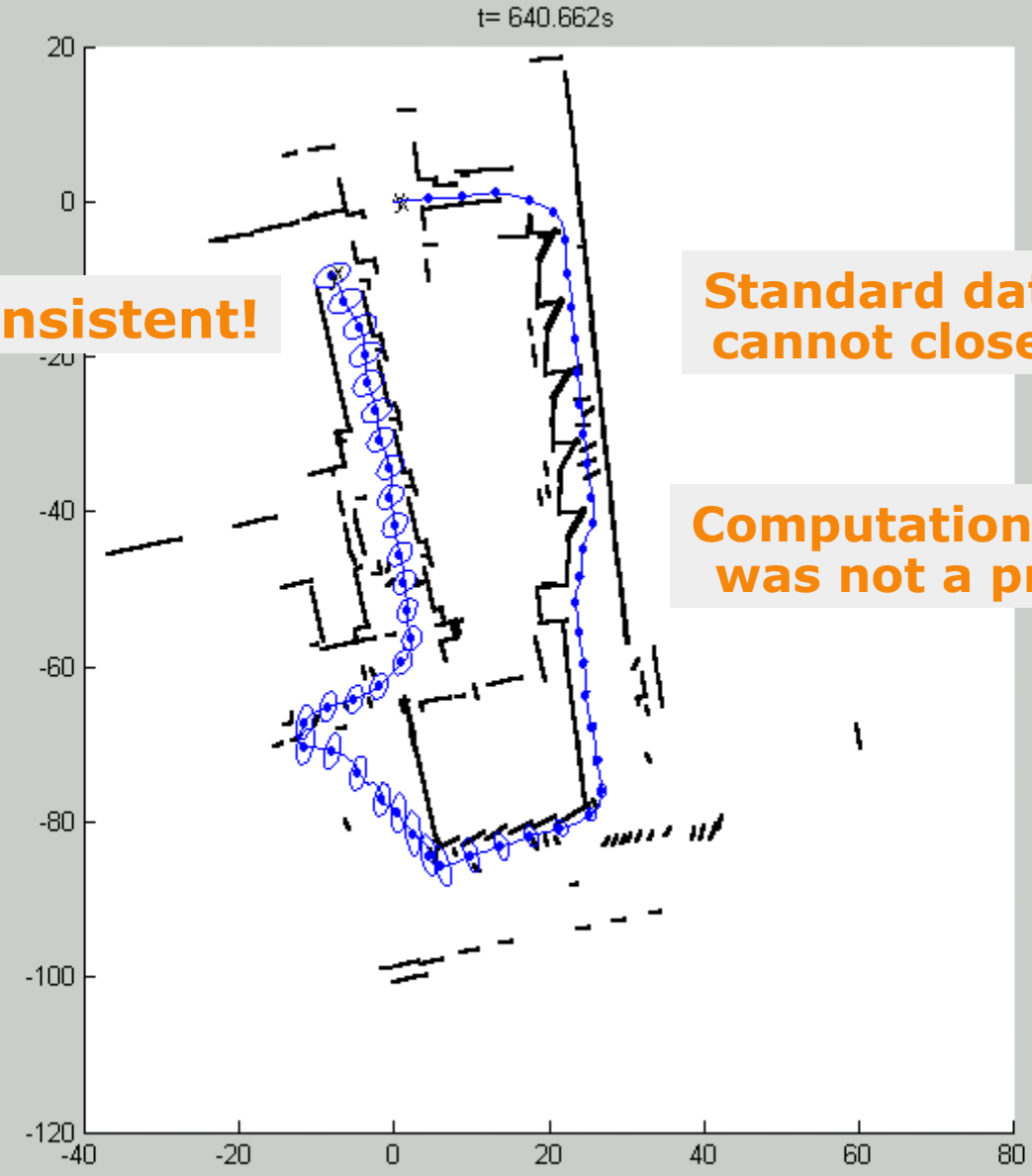
$$D_{ij}^2 \leq \chi^2_{d,1-\alpha}$$

Critical when closing big loops

EKF-SLAM: Real Example



EKF-SLAM: Real Example



Inconsistent!

Standard data association cannot close a 250m loop

Computational complexity was not a problem here!

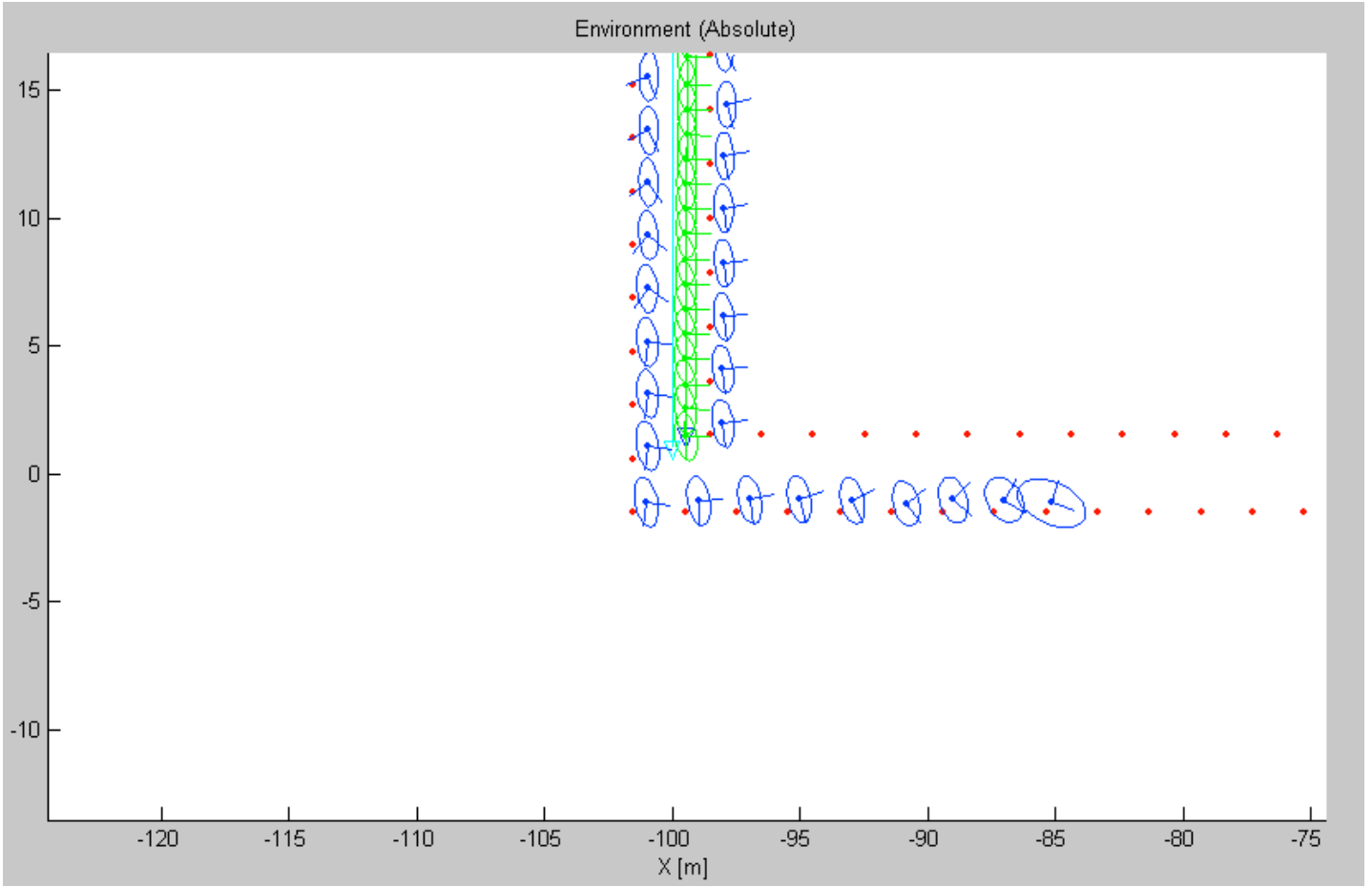
EKF-SLAM: Simulation

- Simulation conditions
  - Perfect data association
  - Ideal odometry and measurement noise
    - » white, Gaussian, known covariance

- Advantages of simulation:
  - Consistency can be tested against the true map
  - A simulation with noise=0 gives the theoretical map covariance (without linearization errors)

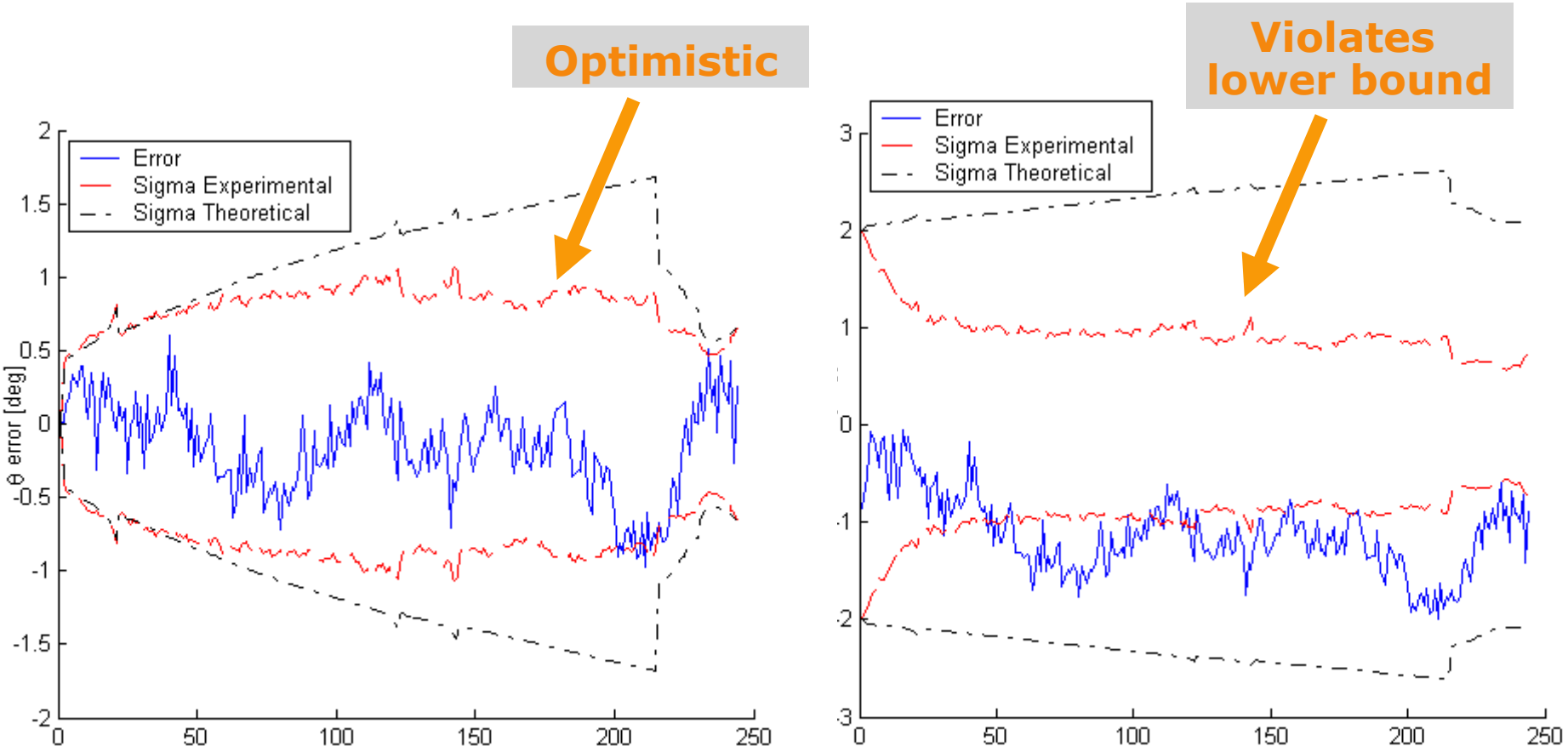
EKF-SLAM: Simulat

Perfect data association and noise model



ROBOTICS
UNIVERSIDAD DE ZARAGOZA

EKF-SLAM: Covariance



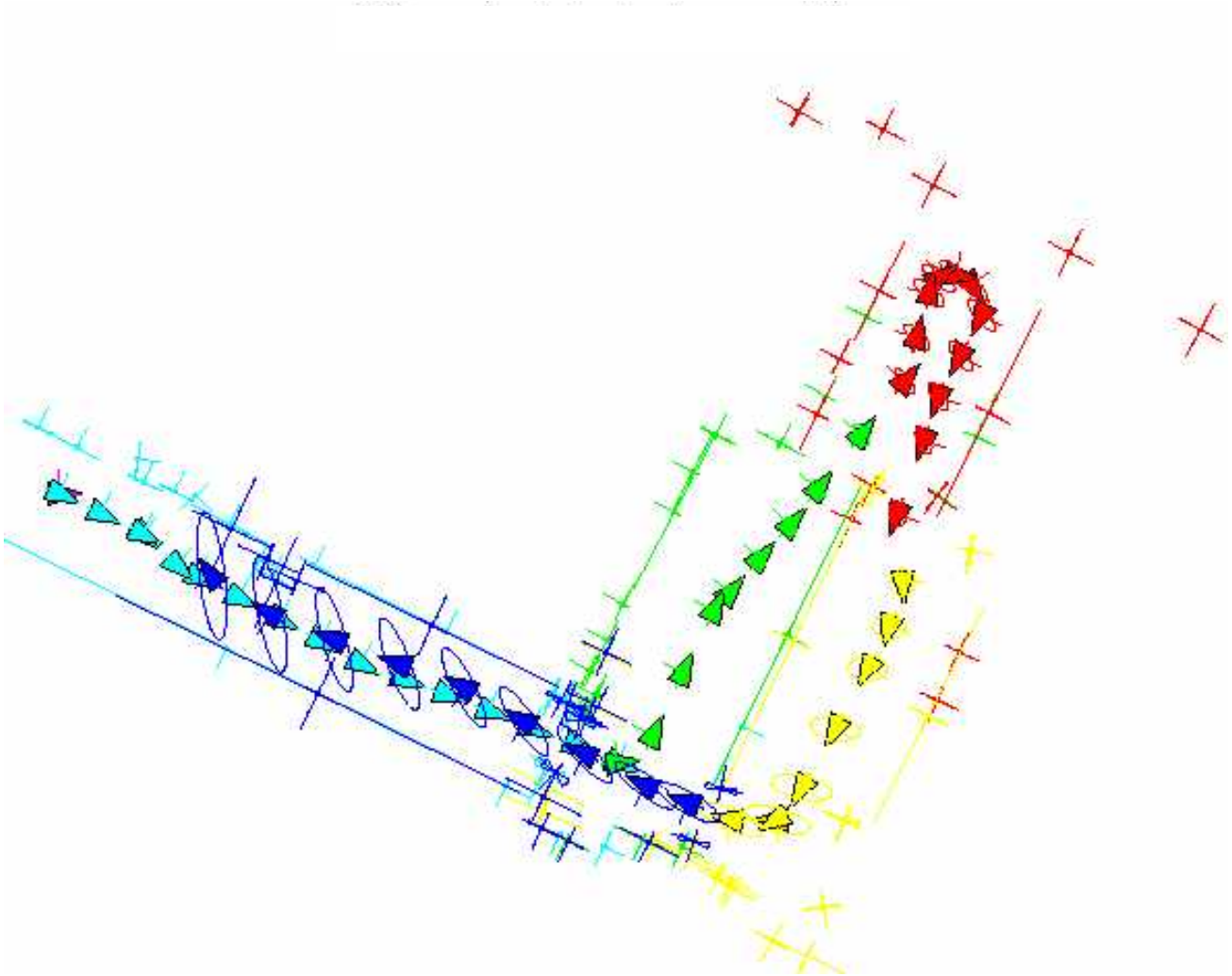
Initial uncertainty = 0

Initial uncertainty > 0

J.A. Castellanos, J. Neira, J.D. Tardós, **Limits to the Consistency of EKF-based SLAM**, 5th IFAC Symposium on Intelligent Autonomous Vehicles, Lisbon, July 2004

Overcoming these problems: Local maps

Local mapping



Local map building

- Periodically, the robot starts a new map, relative to its current location:

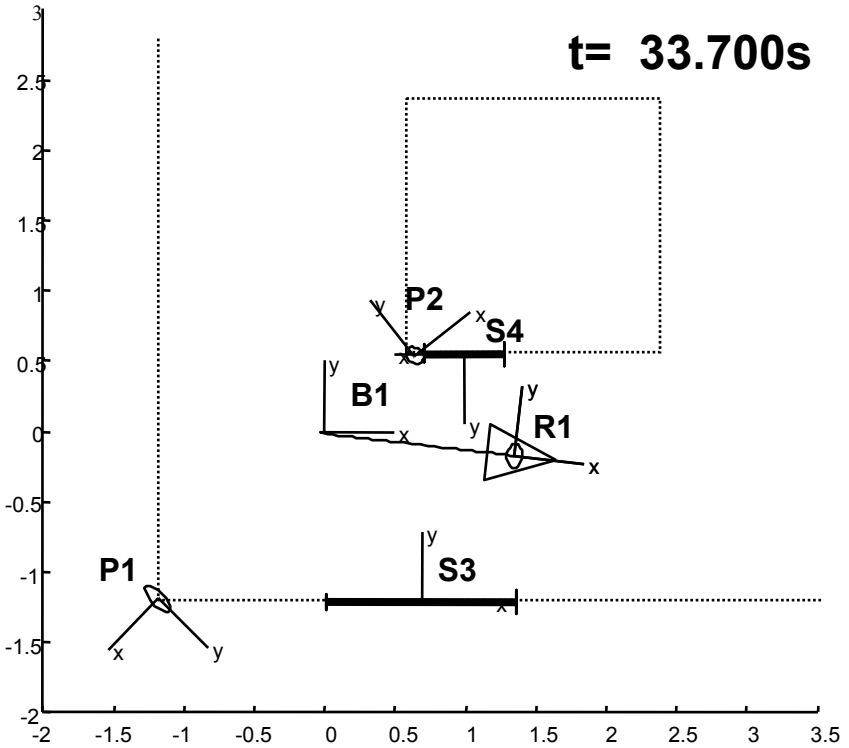
$$\hat{\mathbf{x}}_{R_0}^{B} = 0$$
$$\mathbf{P}_{R_0}^{B} = 0$$

- Given measurements:

$$D^{1\ldots k_1} = \left\{\mathbf{u}_1\, \mathbf{z}_1 \ldots \mathbf{u}_{k_1}\, \mathbf{z}_{k_1}\right\}$$
$$\mathbf{u}_k = \hat{\mathbf{x}}_{R_k}^{R_{k-1}}$$

- EKF approximates the conditional mean:

$$\hat{\mathbf{x}}_{\mathcal{F}_1}^{B_1} \simeq E\left[\mathbf{x}_{\mathcal{F}_1}^{B_1} \mid D^{1\ldots k_1},\ \mathcal{H}^{1\ldots k_1}\right]$$



Local map building

- Second map: $D^{k_1+1\ldots k_2} = \left\{ \mathbf{u}_{k_1+1}\, \mathbf{z}_{k_1+1} \cdots \mathbf{u}_{k_2}\, \mathbf{z}_{k_2} \right\}$

$$\hat{\mathbf{x}}_{\mathcal{F}_2}^{B_2} \simeq E\left[\mathbf{x}_{\mathcal{F}_2}^{B_2} \mid D^{k_1+1\ldots k_2},\ \mathcal{H}^{k_1+1\ldots k_2} \right]$$

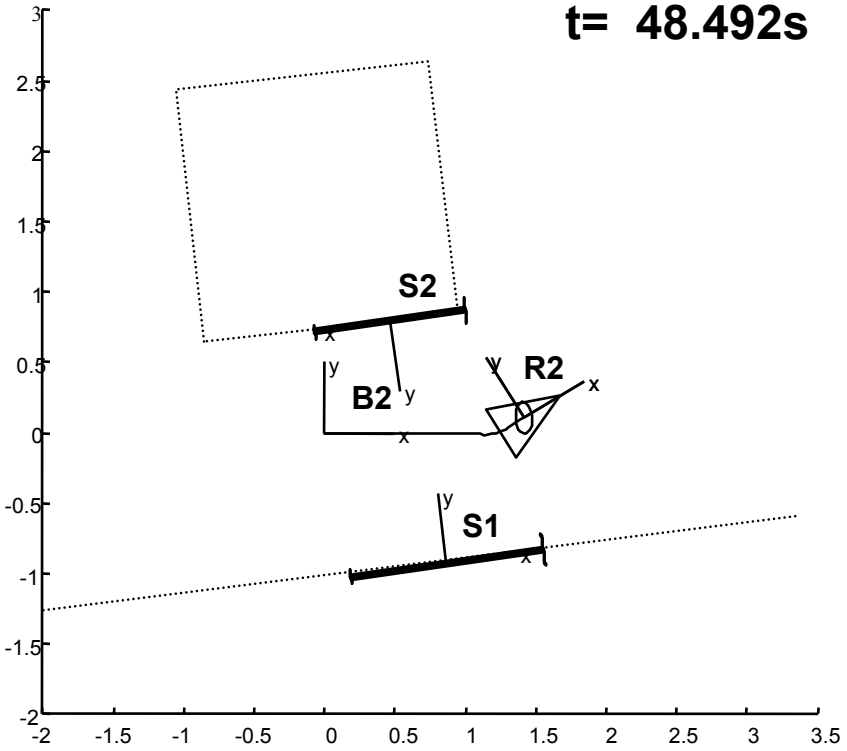
- No information is shared:

$$D^{1\ldots k_1} \cap D^{k_1+1\ldots k_2} = \emptyset$$

Maps are uncorrelated

- Common reference:

$$B_2 = R_1$$

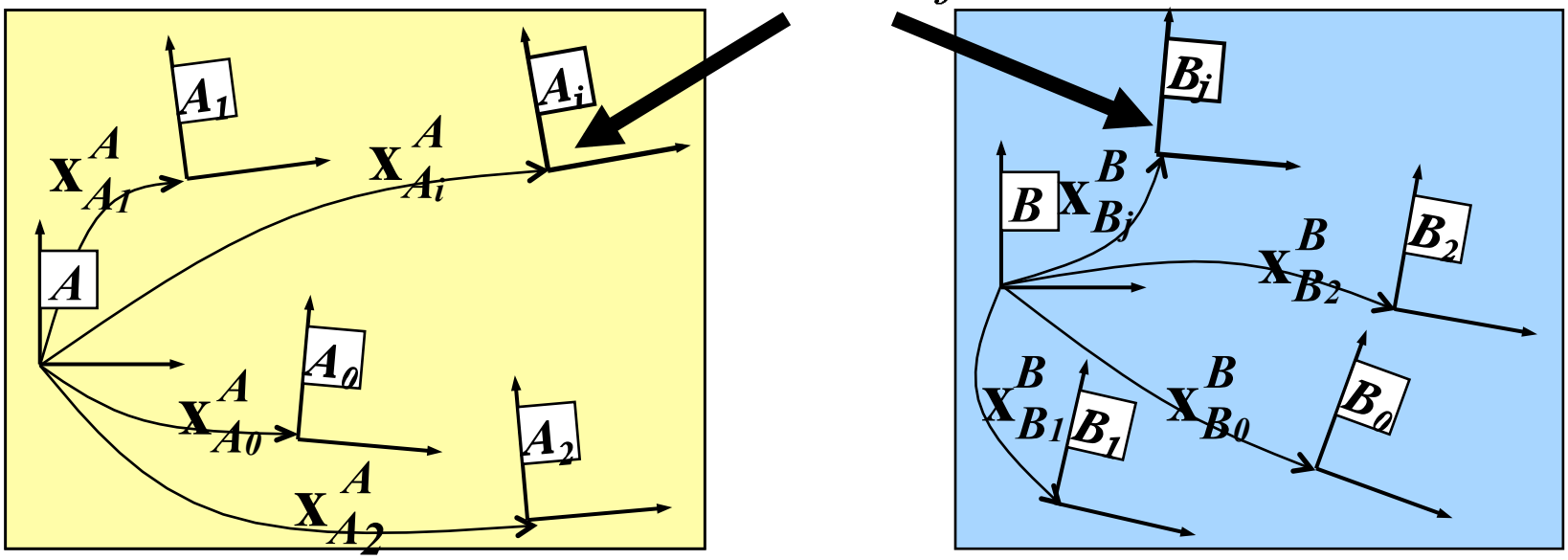


Map Joining

Map Joining

- Given:
  1. Two statistically independent stochastic maps
  2. A common reference

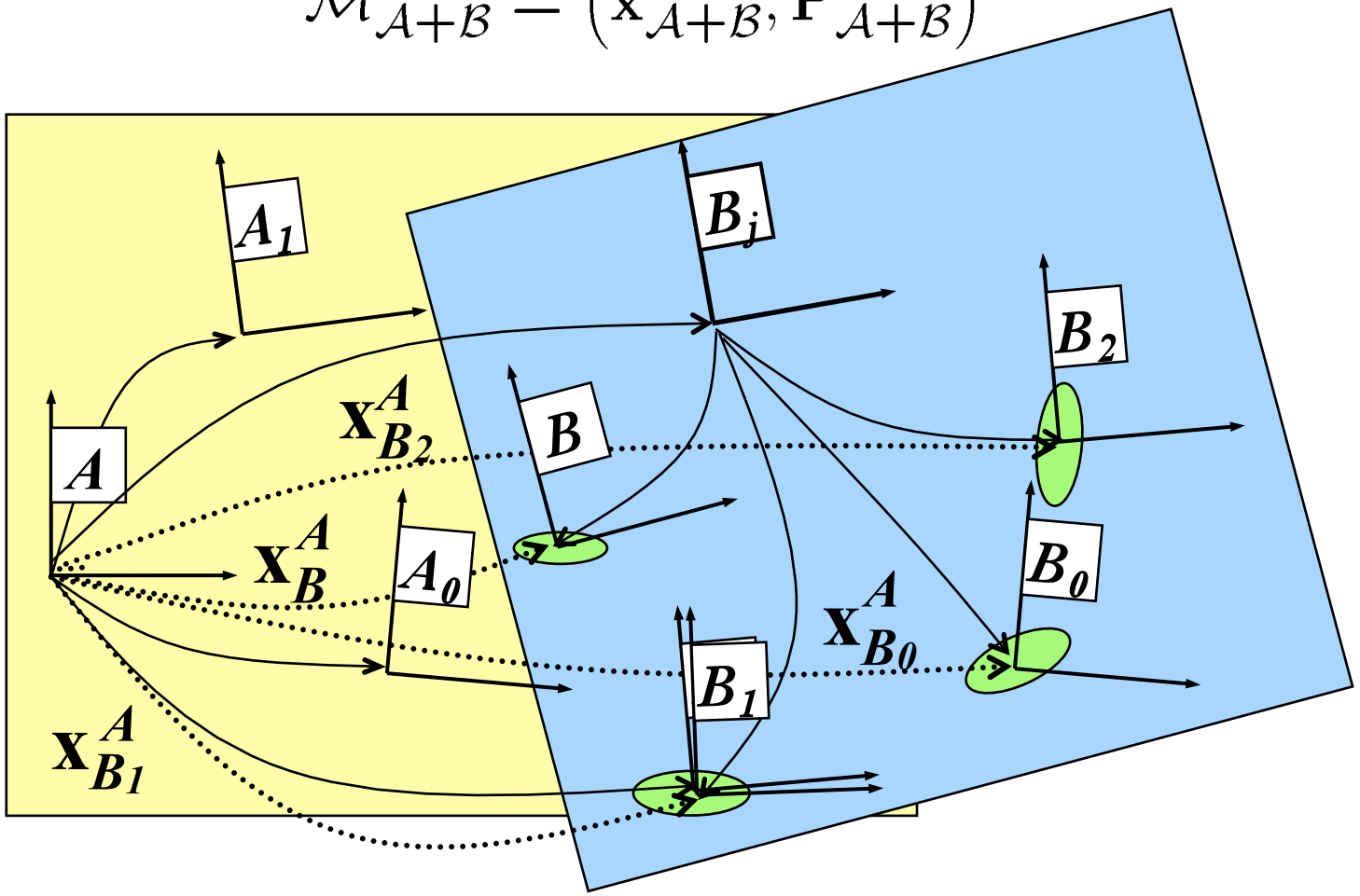
$\mathcal{M}_{\mathcal{A}}^{A} = \left(\hat{\mathbf{x}}_{\mathcal{A}}^{A}, \mathbf{P}_{\mathcal{A}}^{A}\right)$ $A_i = B_j$ $\mathcal{M}_{\mathcal{B}}^{B} = \left(\hat{\mathbf{x}}_{\mathcal{B}}^{B}, \mathbf{P}_{\mathcal{B}}^{B}\right)$



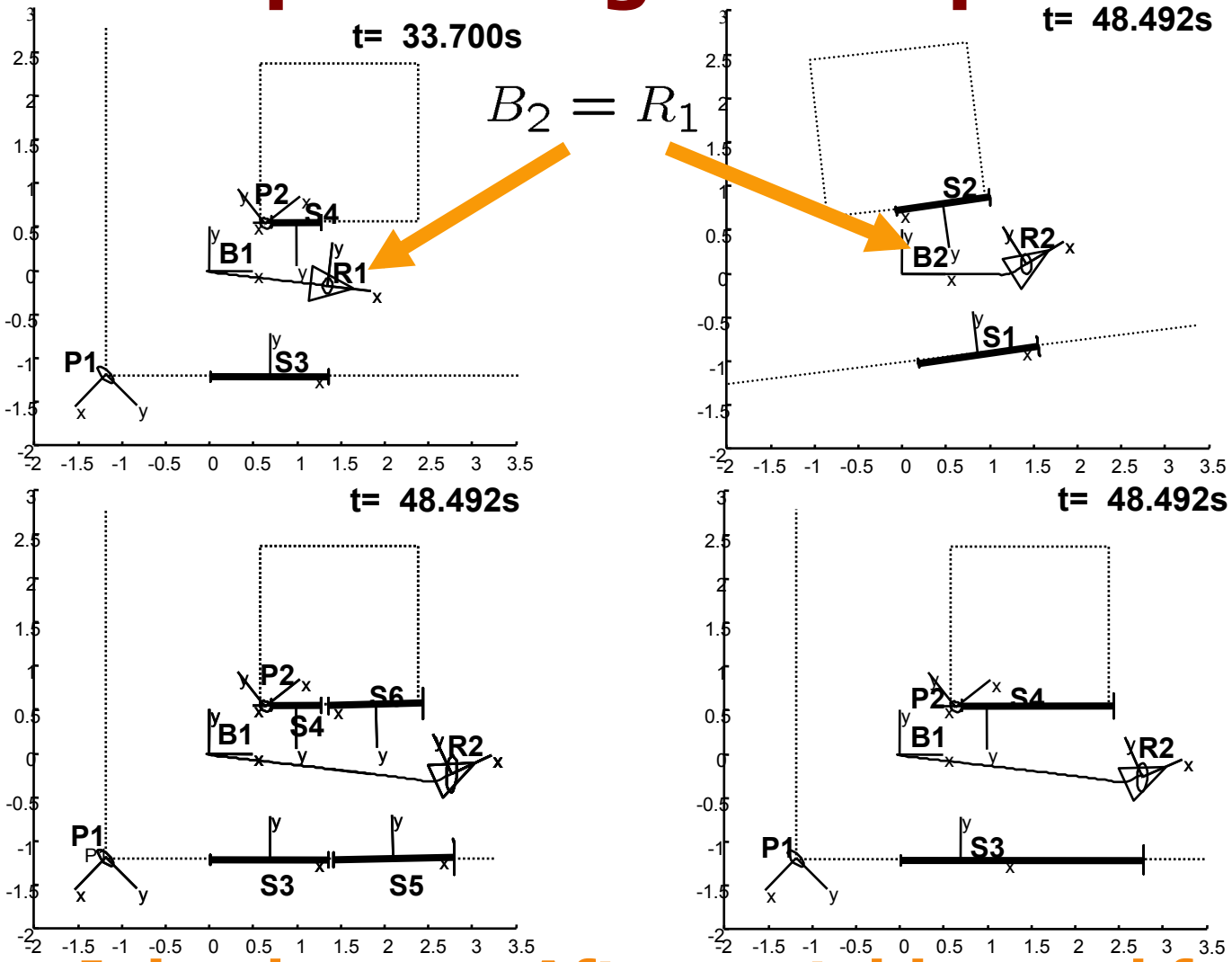
Map Joining

- Conveys the information of the two maps into a single **fully consistent** stochastic map:

$$\mathcal{M}^{A}_{\mathcal{A}+\mathcal{B}} = \left(\hat{\mathbf{x}}^{A}_{\mathcal{A}+\mathcal{B}}, \mathbf{P}^{A}_{\mathcal{A}+\mathcal{B}}\right)$$



Map Joining: Example



Joined map After matching and fusion

J.D. Tardós, J. Neira, P.M. Newman and J.J. Leonard, Robust Mapping and Localization in Indoor Environments using Sonar Data, The Int. Journal of Robotics Research, Vol. 21, No. 4, April, 2002, pp 311 –330

ROBOTIC
UNIVERSIDAD DE ZARA

Map Joining

- New state vector: $\hat{\mathbf{x}}^A_{\mathcal{A}+\mathcal{B}} = \begin{bmatrix} \hat{\mathbf{x}}^A_{\mathcal{A}} \\ \hat{\mathbf{x}}^A_{\mathcal{B}} \end{bmatrix} = \begin{bmatrix} \hat{\mathbf{x}}^A_{\mathcal{A}} \\ \hat{\mathbf{x}}^A_{A_i} \oplus \hat{\mathbf{x}}^{B_j}_{B_0} \\ \vdots \\ \hat{\mathbf{x}}^A_{A_i} \oplus \hat{\mathbf{x}}^{B_j}_{B_m} \end{bmatrix}$

- New covariance matrix:

$$\begin{aligned} \mathbf{P}^A_{\mathcal{A}+\mathcal{B}} &= \mathbf{J}^{\mathcal{A}+\mathcal{B}}_{\mathcal{A}} \mathbf{P}^A_{\mathcal{A}} \mathbf{J}^{\mathcal{A}+\mathcal{B}^T}_{\mathcal{A}} + \mathbf{J}^{\mathcal{A}+\mathcal{B}}_{\mathcal{B}} \mathbf{P}^{B_j}_{\mathcal{B}} \mathbf{J}^{\mathcal{A}+\mathcal{B}^T}_{\mathcal{B}} \\ &= \begin{bmatrix} \mathbf{P}^A_{\mathcal{A}} & \mathbf{P}^A_{\mathcal{A}} \mathbf{J}_1^T \\ \mathbf{J}_1 \mathbf{P}^A_{\mathcal{A}} & \mathbf{J}_1 \mathbf{P}^A_{\mathcal{A}} \mathbf{J}_1^T \end{bmatrix} + \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{J}_2 \mathbf{P}^{B_j}_{\mathcal{B}} \mathbf{J}_2^T \end{bmatrix} \end{aligned}$$

$$\mathbf{J}^{\mathcal{A}+\mathcal{B}}_{\mathcal{A}} = \frac{\partial \hat{\mathbf{x}}^A_{\mathcal{A}+\mathcal{B}}}{\partial \hat{\mathbf{x}}^A_{\mathcal{A}}} = \begin{bmatrix} \mathbf{I} \\ \mathbf{J}_1 \end{bmatrix} \qquad \mathbf{J}_1 = \begin{bmatrix} \mathbf{0} & \cdots & \mathbf{J}_{1\oplus}\left\{\hat{\mathbf{x}}^A_{A_i}, \hat{\mathbf{x}}^{B_j}_{B_0}\right\} & \cdots & \mathbf{0} \\ \vdots & & \vdots & & \vdots \\ \mathbf{0} & \cdots & \mathbf{J}_{1\oplus}\left\{\hat{\mathbf{x}}^A_{A_i}, \hat{\mathbf{x}}^{B_j}_{B_m}\right\} & \cdots & \mathbf{0} \end{bmatrix}$$

$$\mathbf{J}^{\mathcal{A}+\mathcal{B}}_{\mathcal{B}} = \frac{\partial \hat{\mathbf{x}}^A_{\mathcal{A}+\mathcal{B}}}{\partial \hat{\mathbf{x}}^{B_j}_{\mathcal{B}}} = \begin{bmatrix} \mathbf{0} \\ \mathbf{J}_2 \end{bmatrix} \qquad \mathbf{J}_2 = \begin{bmatrix} \mathbf{J}_{2\oplus}\left\{\hat{\mathbf{x}}^A_{A_i}, \hat{\mathbf{x}}^{B_j}_{B_0}\right\} & \cdots & \mathbf{0} \\ \vdots & \ddots & \vdots \\ \mathbf{0} & \cdots & \mathbf{J}_{2\oplus}\left\{\hat{\mathbf{x}}^A_{A_i}, \hat{\mathbf{x}}^{B_j}_{B_m}\right\} \end{bmatrix}$$

Map joining is $O(n_1 n_2)$ on the maps

Matching and Fusion

- Matching function:

$$\mathbf{f}_{ij_i}(\mathbf{x}) = 0$$

- Joint matching function for the hypothesis:

$$\mathbf{f}_{\mathcal{H}}(\mathbf{x}) = \begin{bmatrix} \mathbf{f}_{1j_1}(\mathbf{x}) \\ \vdots \\ \mathbf{f}_{mj_m}(\mathbf{x}) \end{bmatrix} \simeq \mathbf{h}_{\mathcal{H}} + \mathbf{H}_{\mathcal{H}}(\mathbf{x} - \hat{\mathbf{x}}) = 0$$

- Joint innovation test:

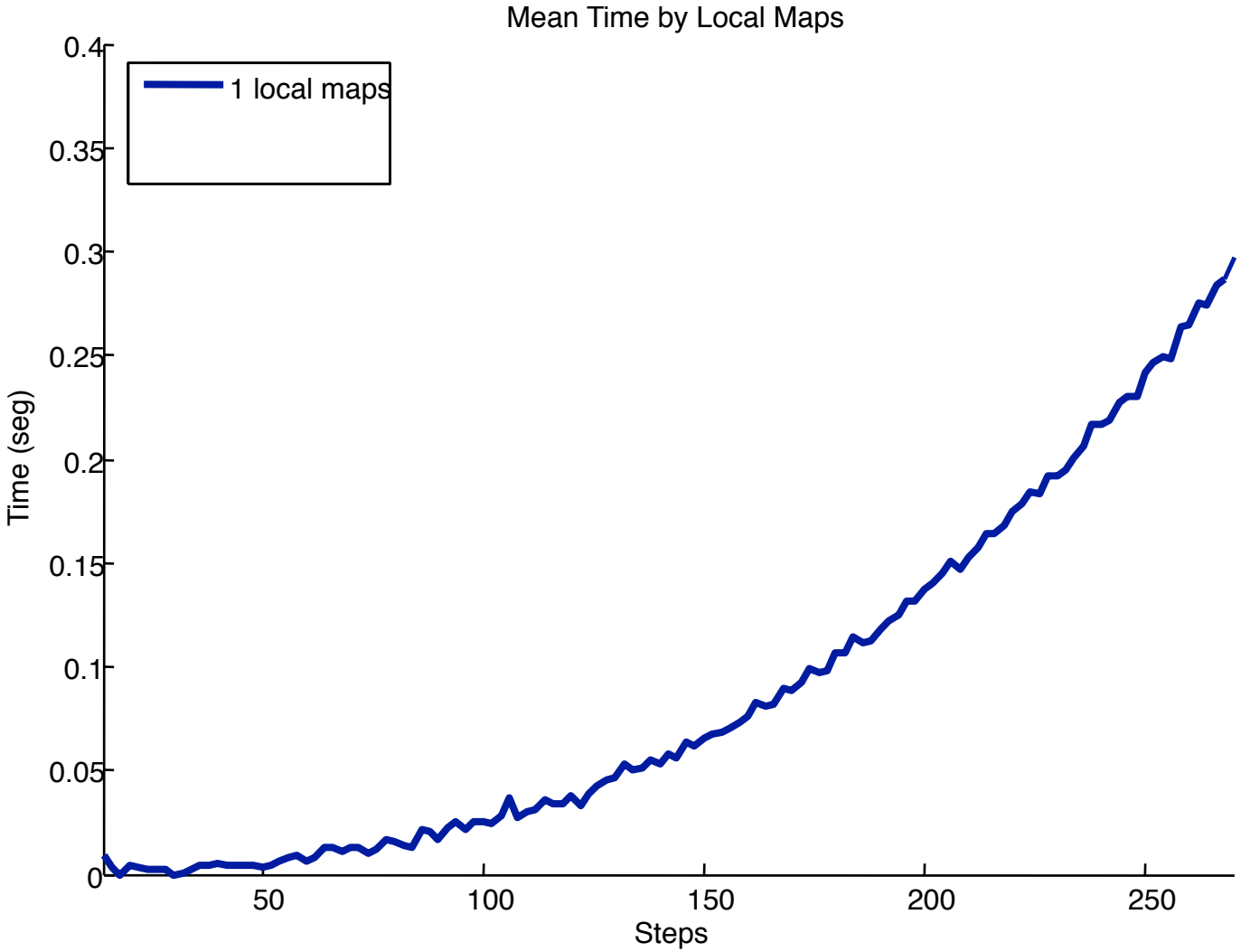
$$D^2_{\mathcal{H}} = \mathbf{h}^T_{\mathcal{H}} \left(\mathbf{H}_{\mathcal{H}} \mathbf{P} \mathbf{H}^T_{\mathcal{H}} \right)^{-1} \mathbf{h}_{\mathcal{H}} < \chi^2_{d,\alpha}$$

- Map update using EKF:

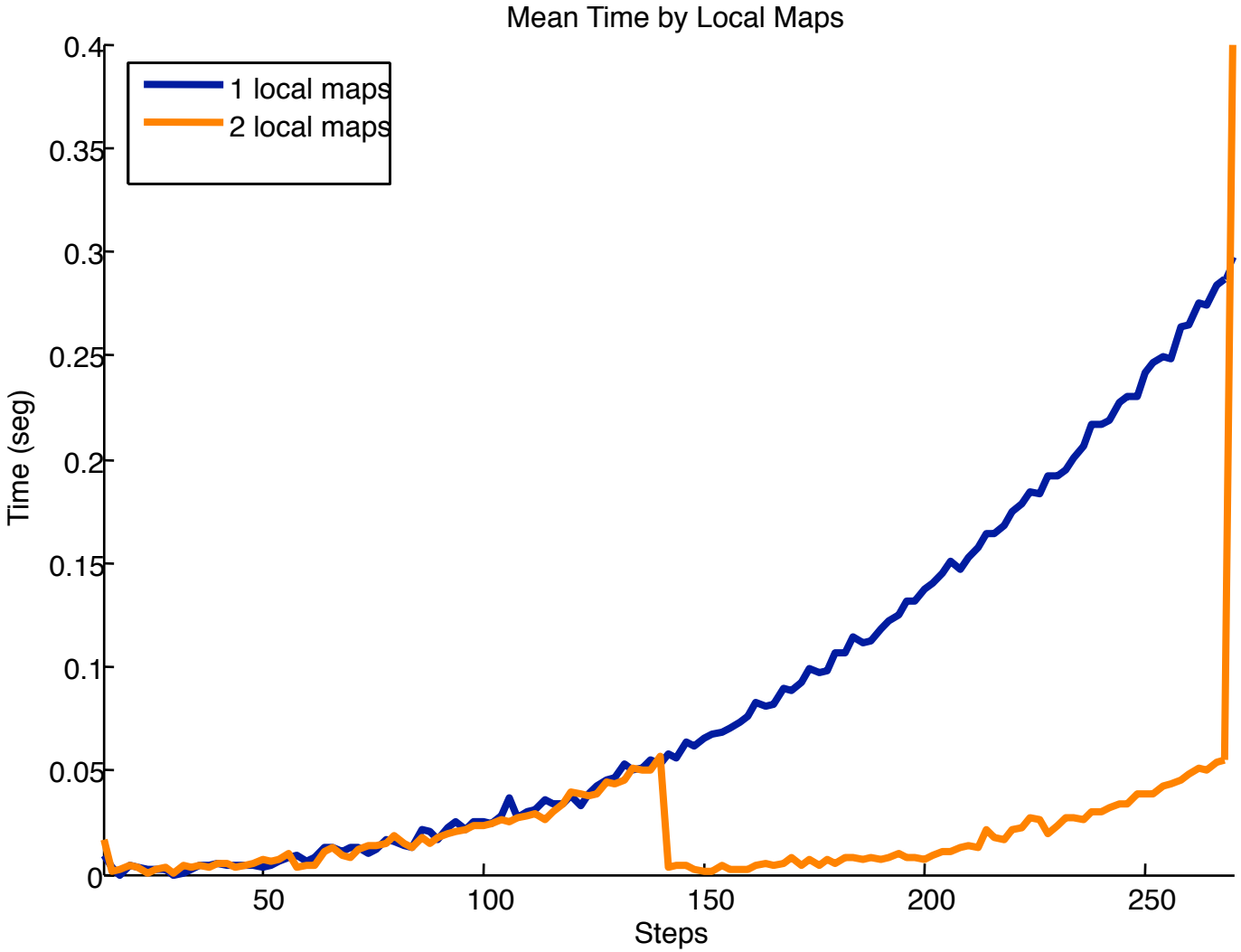
$$\begin{aligned} \hat{\mathbf{x}}_k &= \hat{\mathbf{x}}_{k-1} - \mathbf{K}_k \mathbf{h}_{\mathcal{H}} \\ \mathbf{P}_k &= (\mathbf{I} - \mathbf{K}_k \mathbf{H}_{\mathcal{H}}) \, \mathbf{P}_{k-1} \\ \mathbf{K}_k &= \mathbf{P}_{k-1} \mathbf{H}^T_{\mathcal{H}} \left(\mathbf{H}_{\mathcal{H}} \mathbf{P}_{k-1} \mathbf{H}^T_{\mathcal{H}} \right)^{-1} \end{aligned}$$

Matching and fusion is $O((n_1+n_2)^2)$

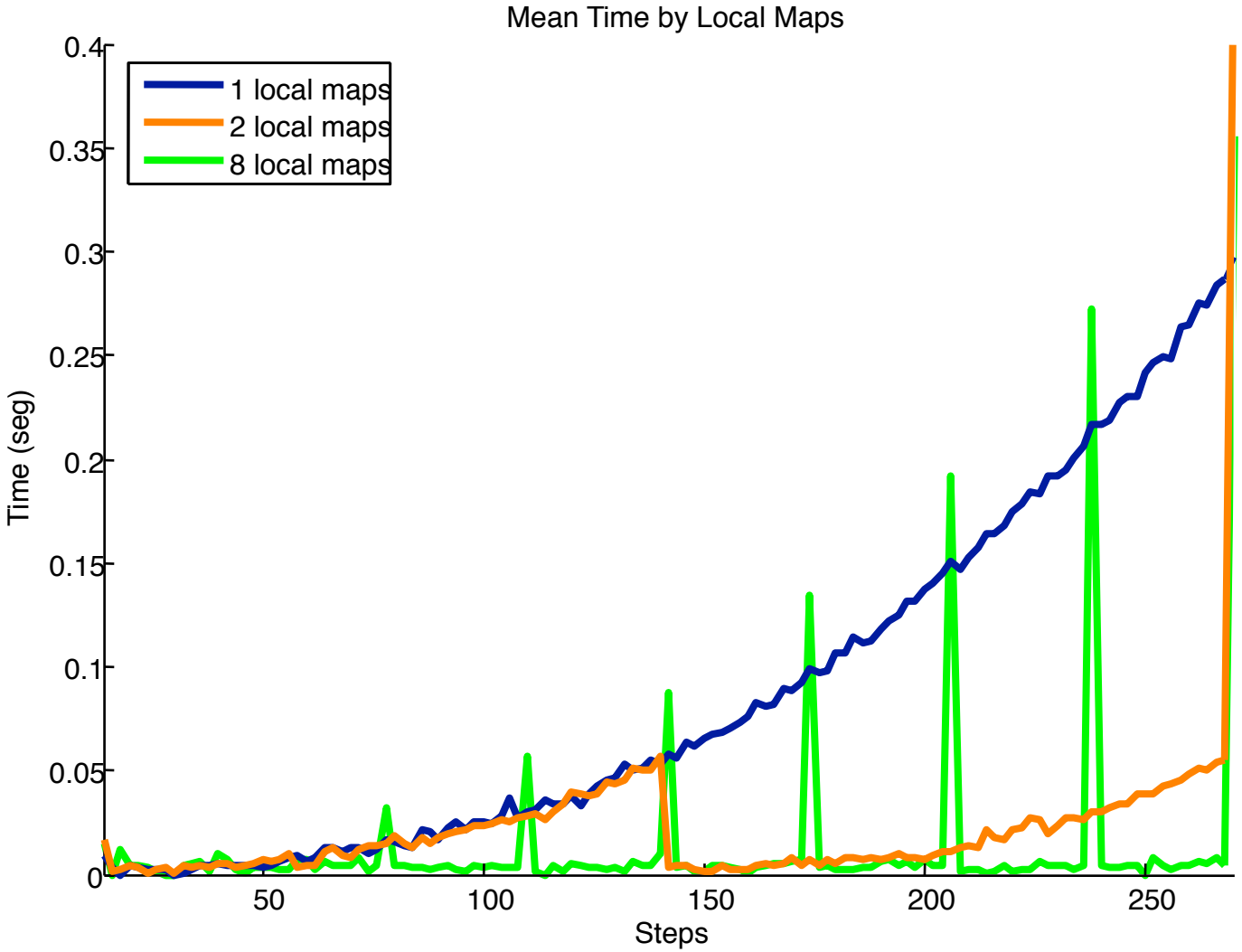
EKF updates


Mean Time by Local Maps
1 local maps
Time (seg)
Steps
0
0.05
0.1
0.15
0.2
0.25
0.3
0.35
0.4
50
100
150
200
250

Map Joining


Mean Time by Local Maps
1 local maps
2 local maps
Time (seg)
Steps
0
0.05
0.1
0.15
0.2
0.25
0.3
0.35
0.4
50
100
150
200
250

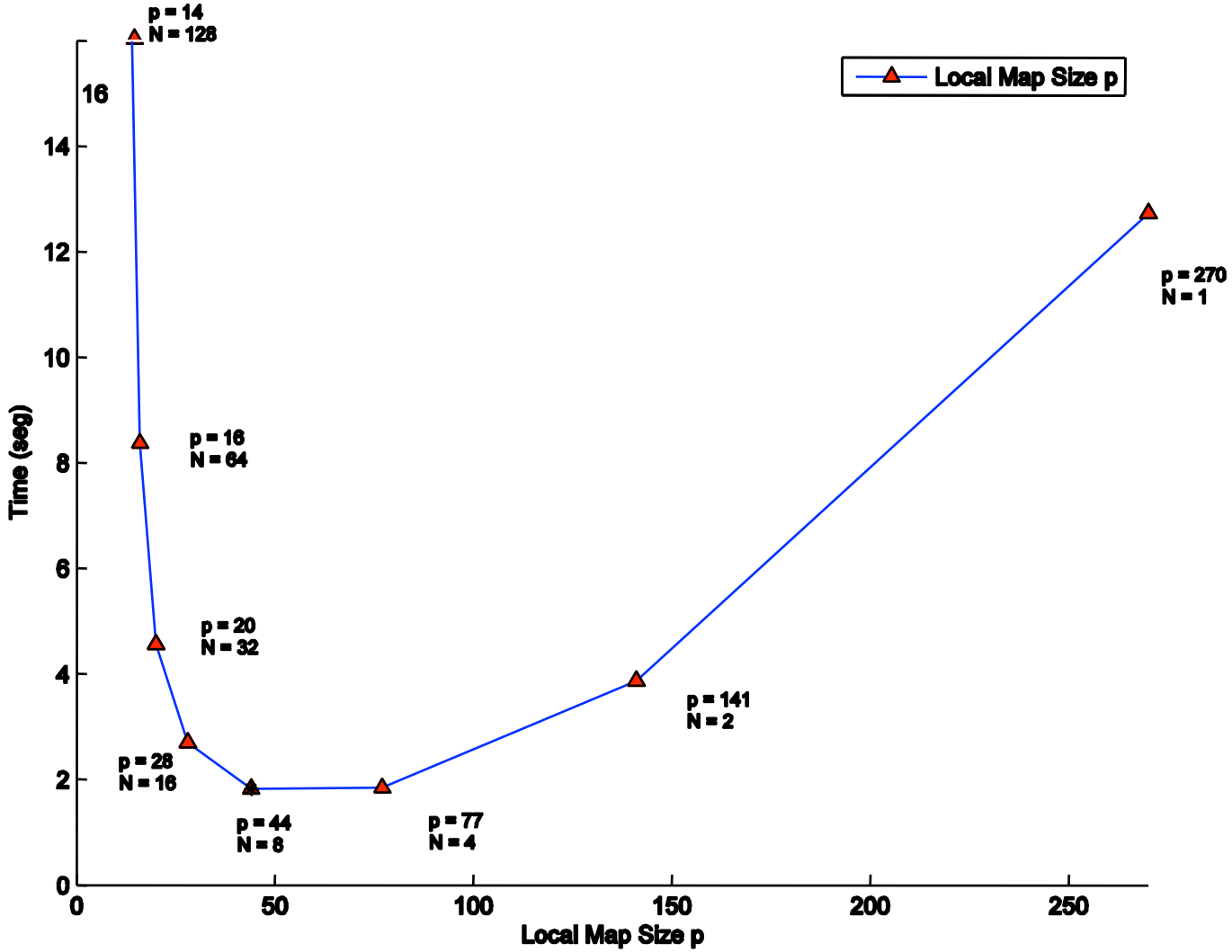
Map Joining



Map joining is $O(n^2)$

Local map size

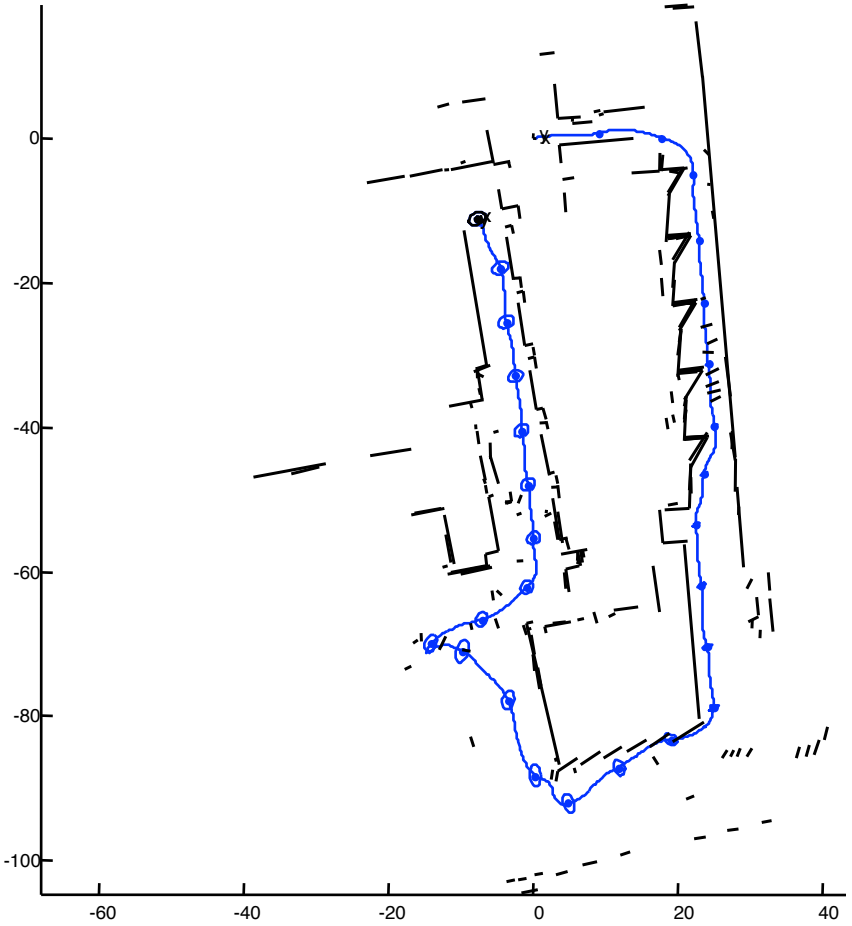
- Size matters!



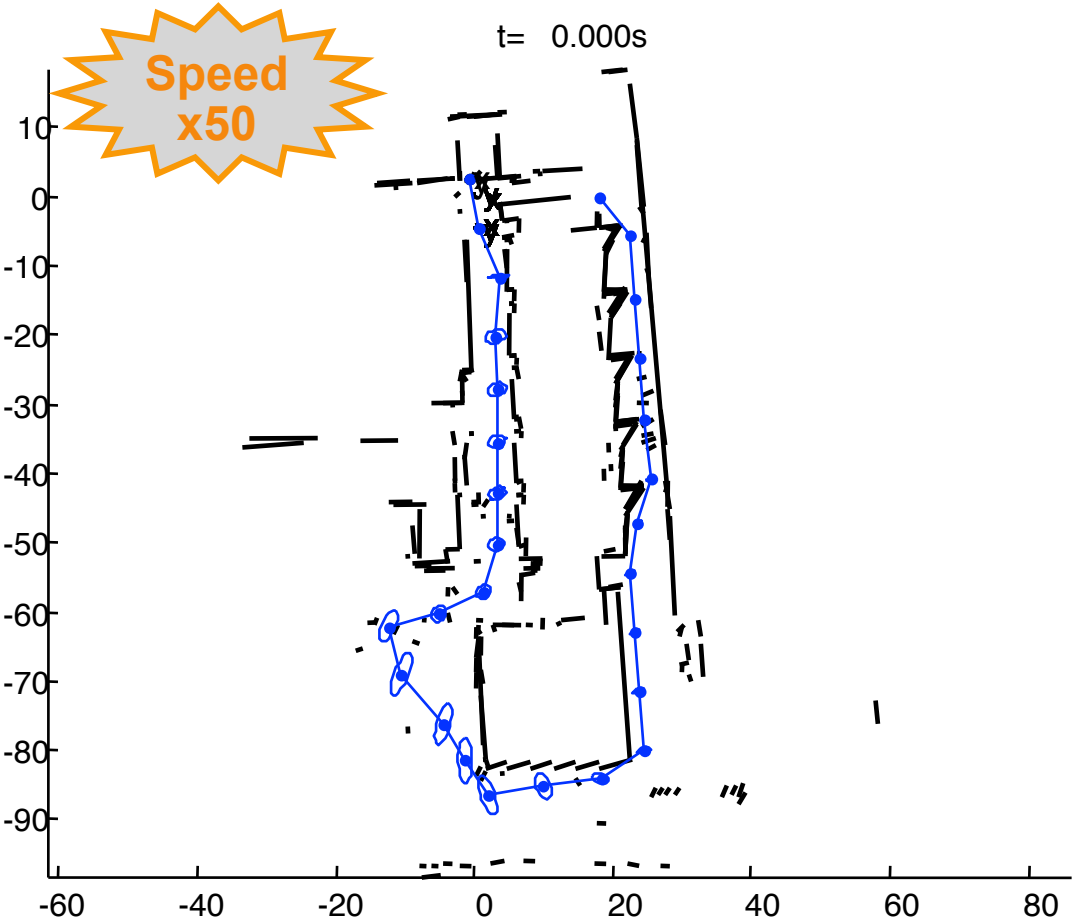
Computational cost for different local map sizes

Map Joining closes the loop!

- One full SLAM run



- Map joining of 28 local maps

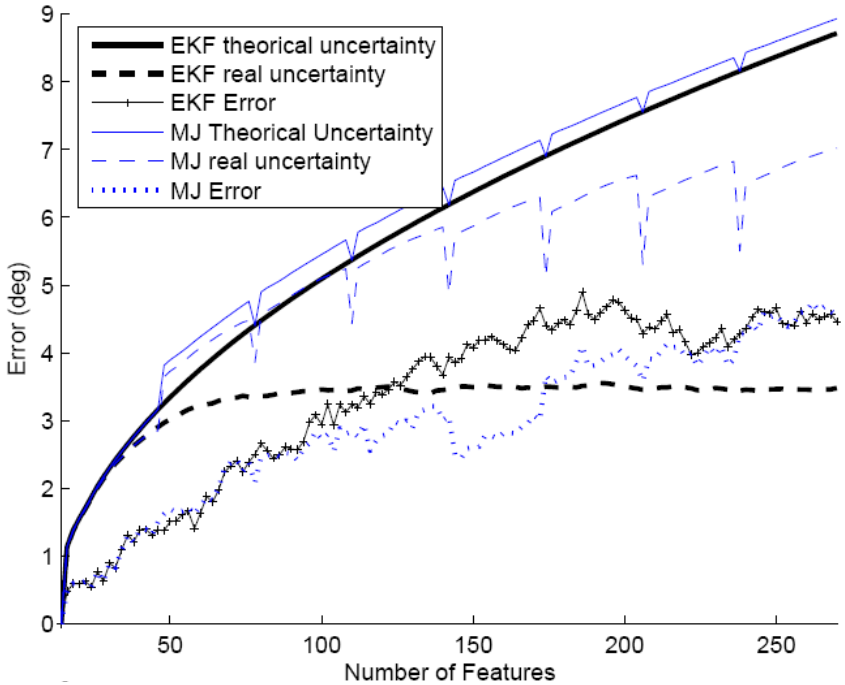


Local maps bound linearization error effects

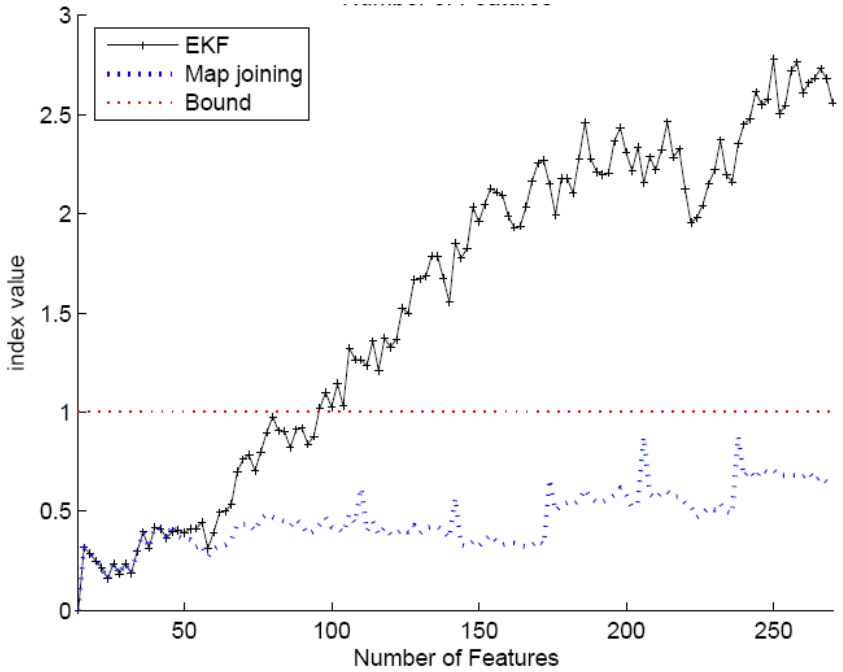
ROBOTICS
UNIVERSIDAD DE ZARAGOZA

Map Joining

- Mean vehicle orientation error for full EKF and Map Joining 8 local maps

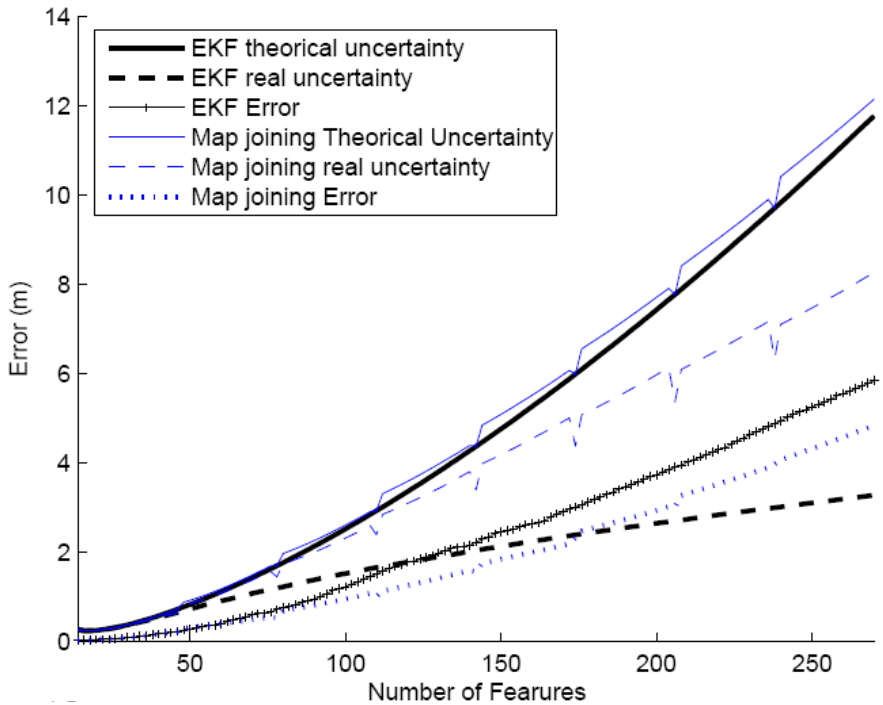


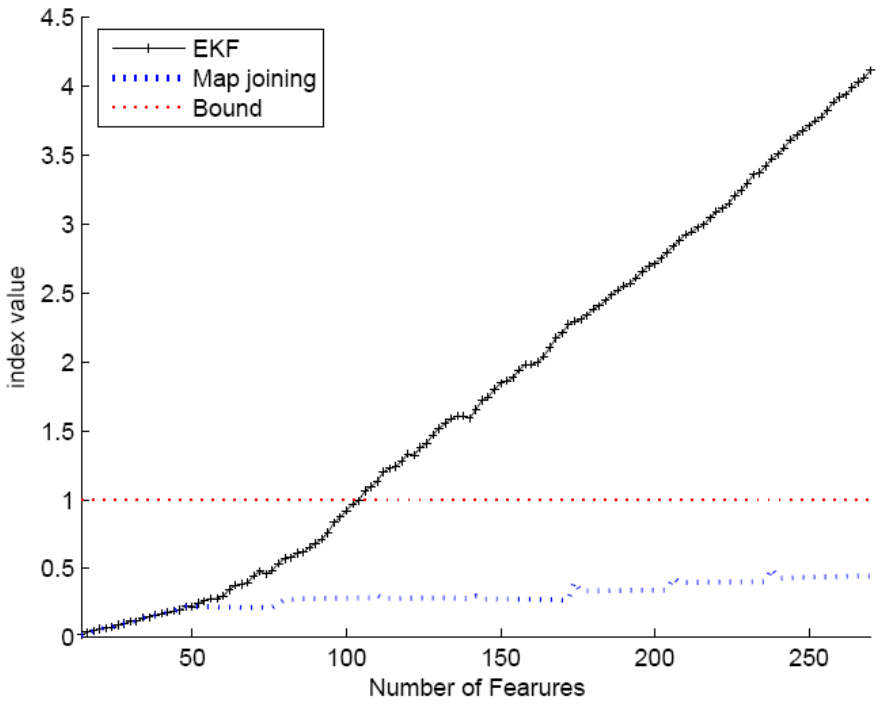
Vehicle orientation error



Vehicle orientation consistency index

Map Joining





Mean feature position error

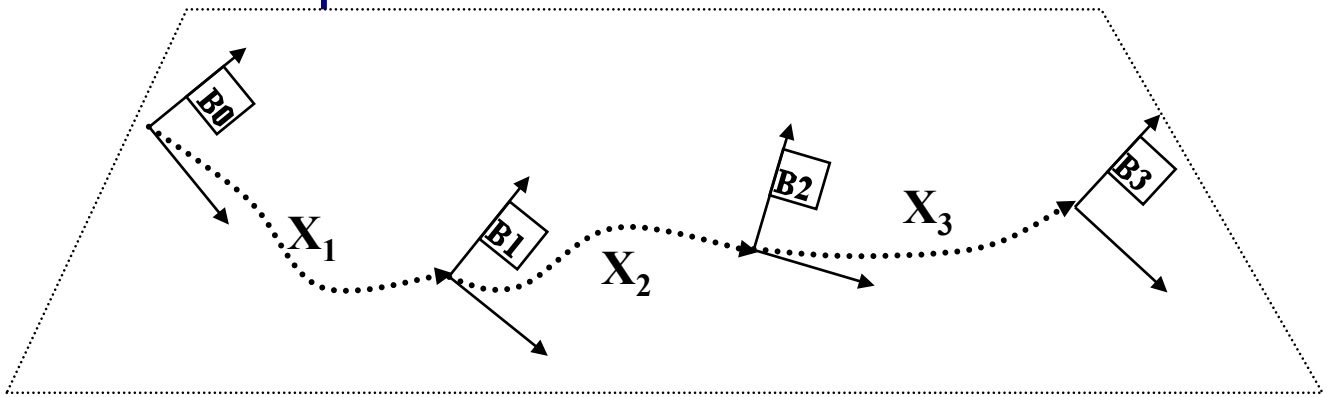
Mean feature consistency index

ROBOTICS
UNIVERSIDAD DE ZARAGOZA

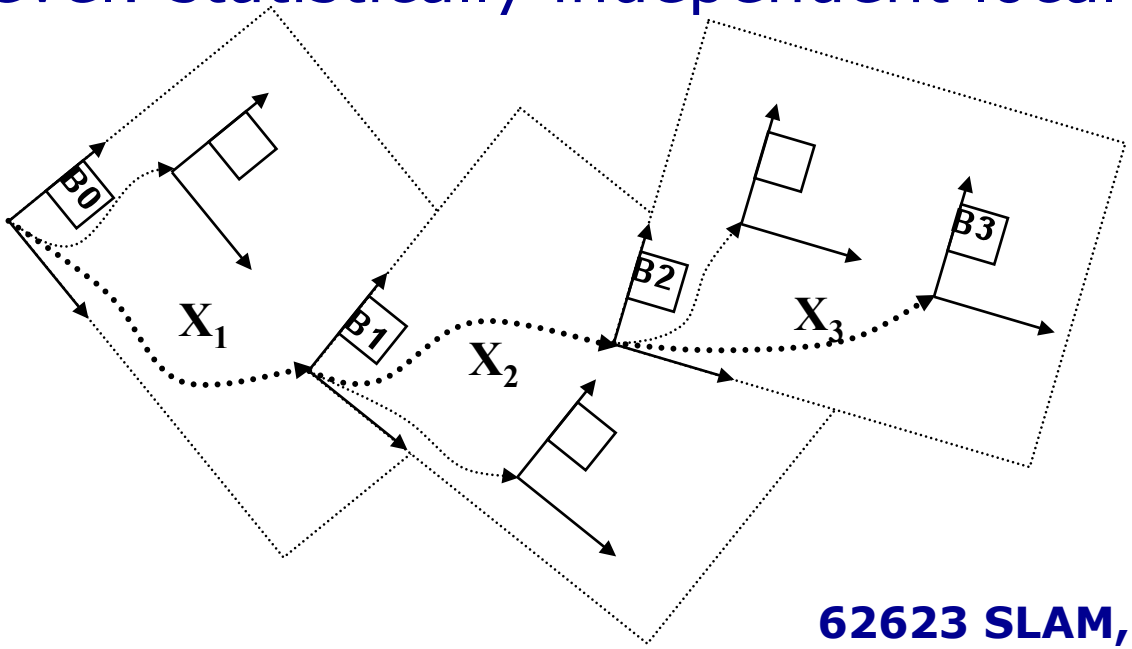
Hierarchical SLAM

Hierarchical SLAM

- Global level: adjacency graph and relative stochastic map



- Local level: statistically independent local maps



Hierarchical SLAM

- Local maps:

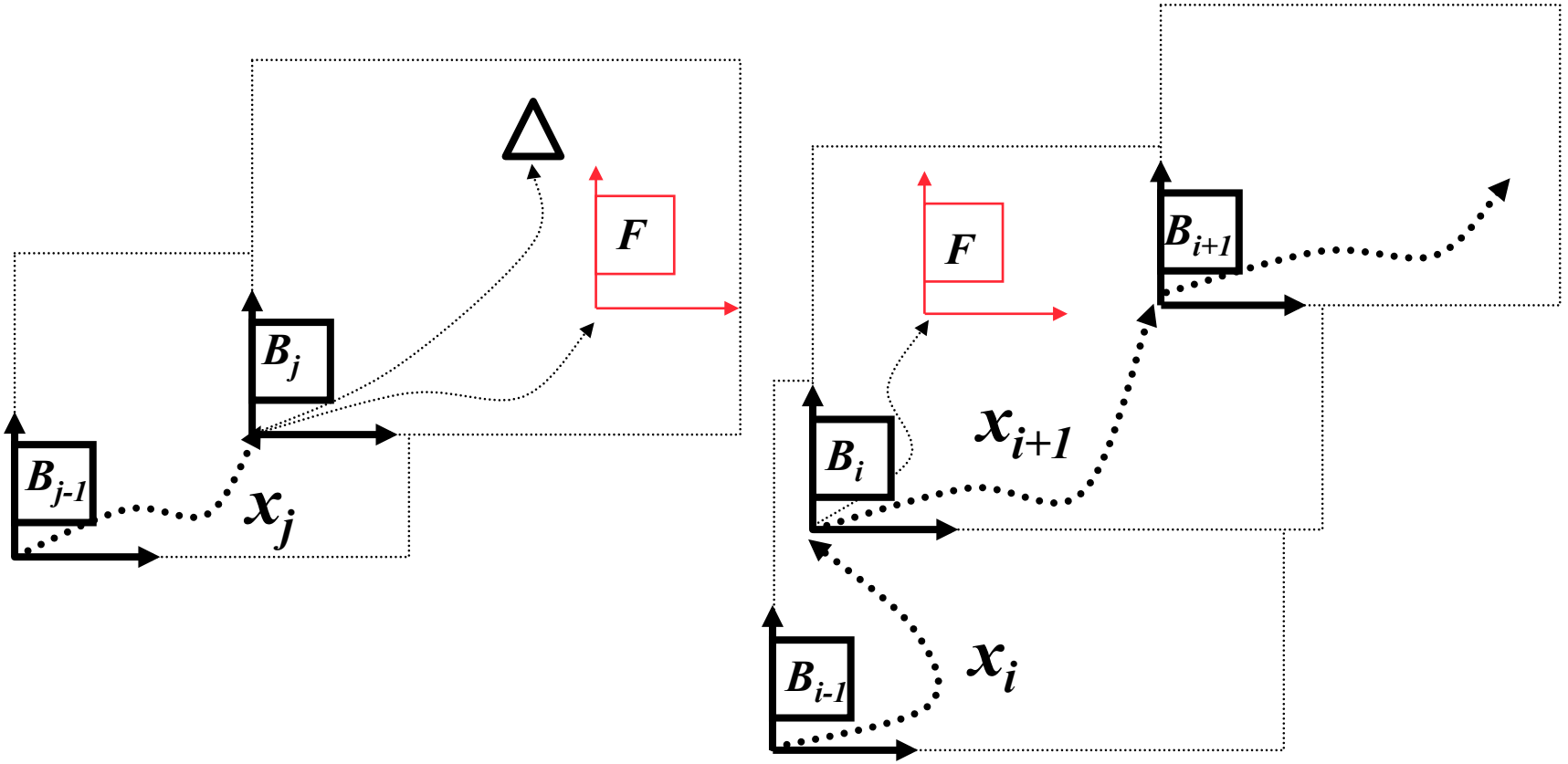
$$\hat{\mathbf{x}}^B_{\mathcal{F}} = \begin{bmatrix} \hat{\mathbf{x}}^B_R \\ \vdots \\ \hat{\mathbf{x}}^B_{F_n} \end{bmatrix}; \; \mathbf{P}_{\mathcal{F}} = \begin{bmatrix} \mathbf{P}^B_R & \cdots & \mathbf{P}^B_{RF_n} \\ \vdots & \ddots & \vdots \\ \mathbf{P}^B_{F_nR} & \cdots & \mathbf{P}^B_{F_nF_n} \end{bmatrix}$$

- Global relative map:

$$\hat{\mathbf{x}} = \begin{bmatrix} \vdots \\ \hat{\mathbf{x}}_i \\ \vdots \end{bmatrix}; \; \mathbf{P} = \begin{bmatrix} . & 0 & 0 \\ 0 & \mathbf{P}_i & 0 \\ 0 & 0 & . \end{bmatrix}$$

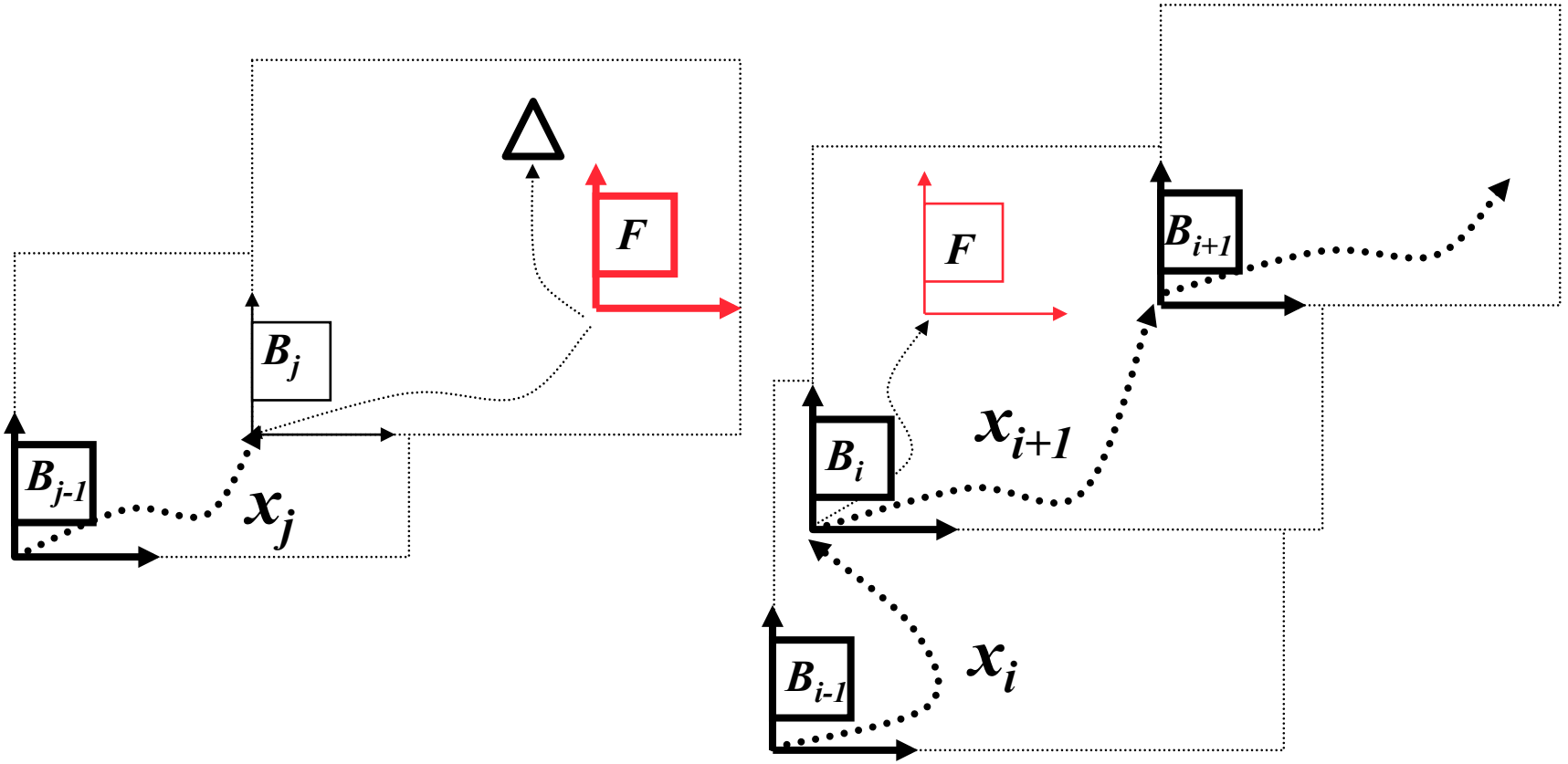
Block diagonal

Loop closing



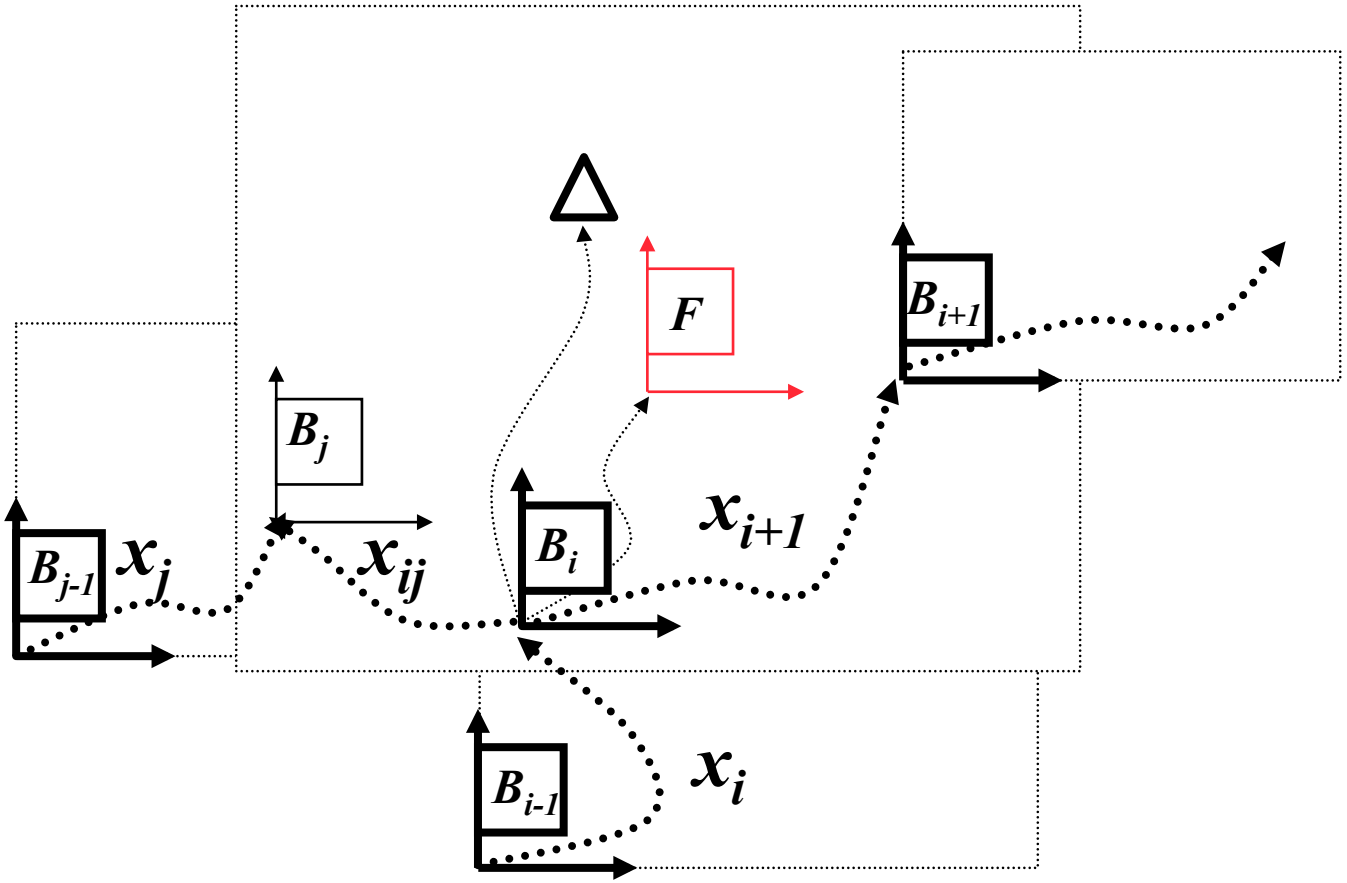
Find a common reference between maps *i* and *j*

Loop closing



Change base reference of map *i* to *F*.

Loop closing



Join maps *i* and *j*.

Hierarchical SLAM

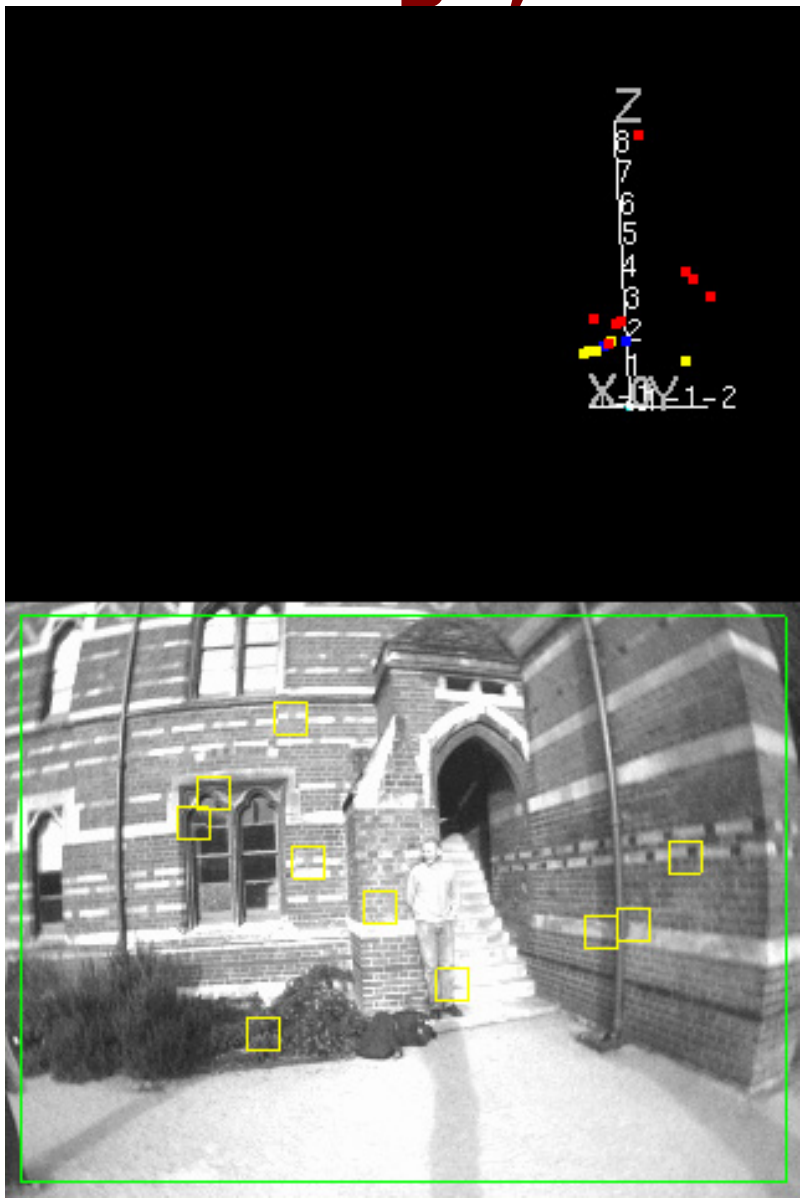
- Global relative map before loop closing:

$$\hat{\mathbf{x}} = \begin{bmatrix} \vdots \\ \hat{\mathbf{x}}_{i+1} \\ \vdots \\ \hat{\mathbf{x}}_j \end{bmatrix}; \; \mathbf{P} = \begin{bmatrix} . & 0 & 0 & 0 \\ 0 & \mathbf{P}_{i+1} & 0 & 0 \\ 0 & 0 & . & 0 \\ 0 & 0 & 0 & \mathbf{P}_j \end{bmatrix}$$

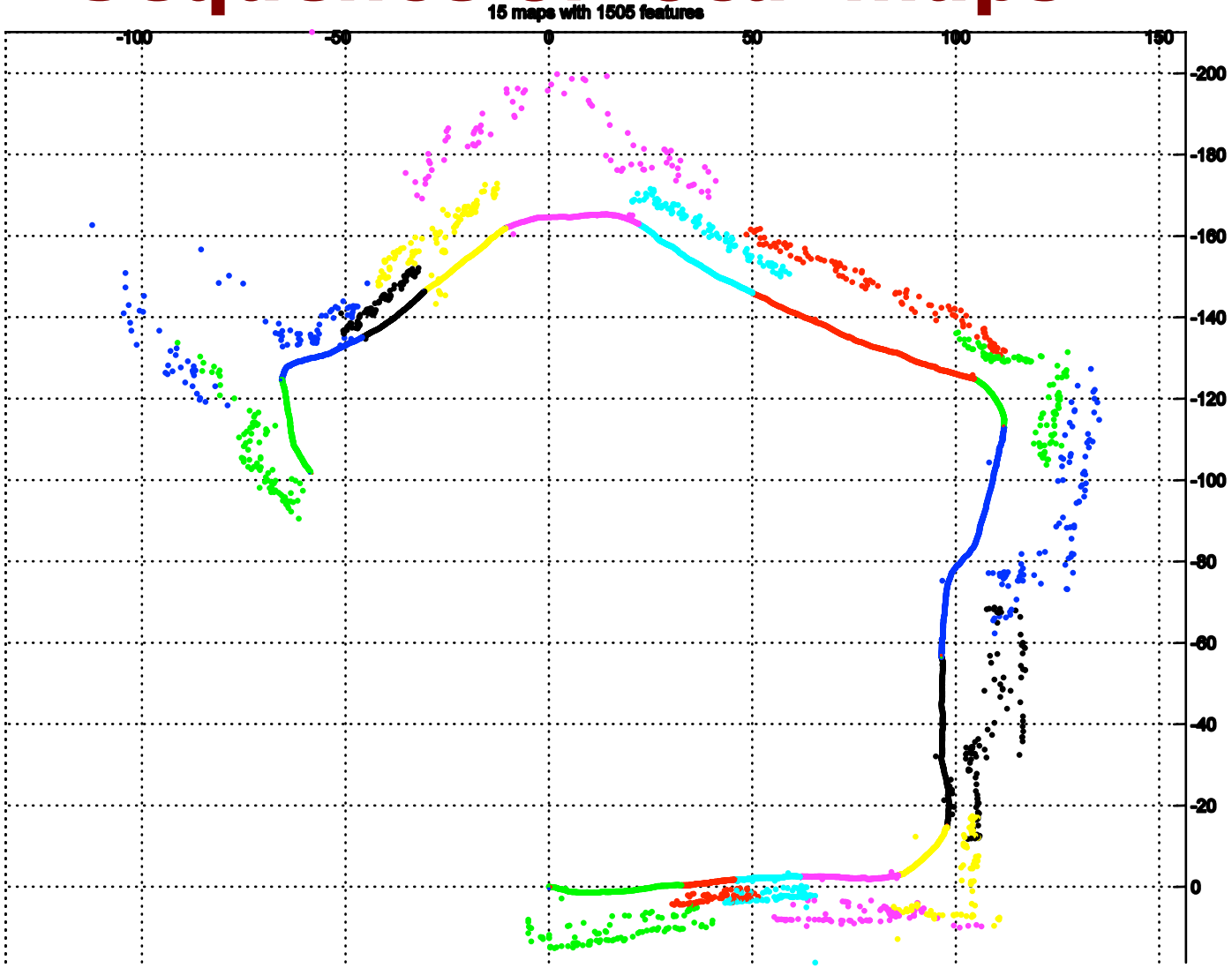
- After loop closing:

$$\hat{\mathbf{x}} = \begin{bmatrix} \vdots \\ \hat{\mathbf{x}}_{i+1} \\ \hat{\mathbf{x}}_{ij} \\ \vdots \\ \hat{\mathbf{x}}_j \end{bmatrix}; \; \mathbf{P} = \begin{bmatrix} . & 0 & 0 & 0 & 0 \\ 0 & \mathbf{P}_{i+1} & \mathbf{P}_{i+1,ij} & 0 & 0 \\ 0 & \mathbf{P}^T_{i+1,ij} & \mathbf{P}_{ij} & 0 & 0 \\ 0 & 0 & 0 & . & 0 \\ 0 & 0 & 0 & 0 & \mathbf{P}_j \end{bmatrix}$$

Keble College, Oxford

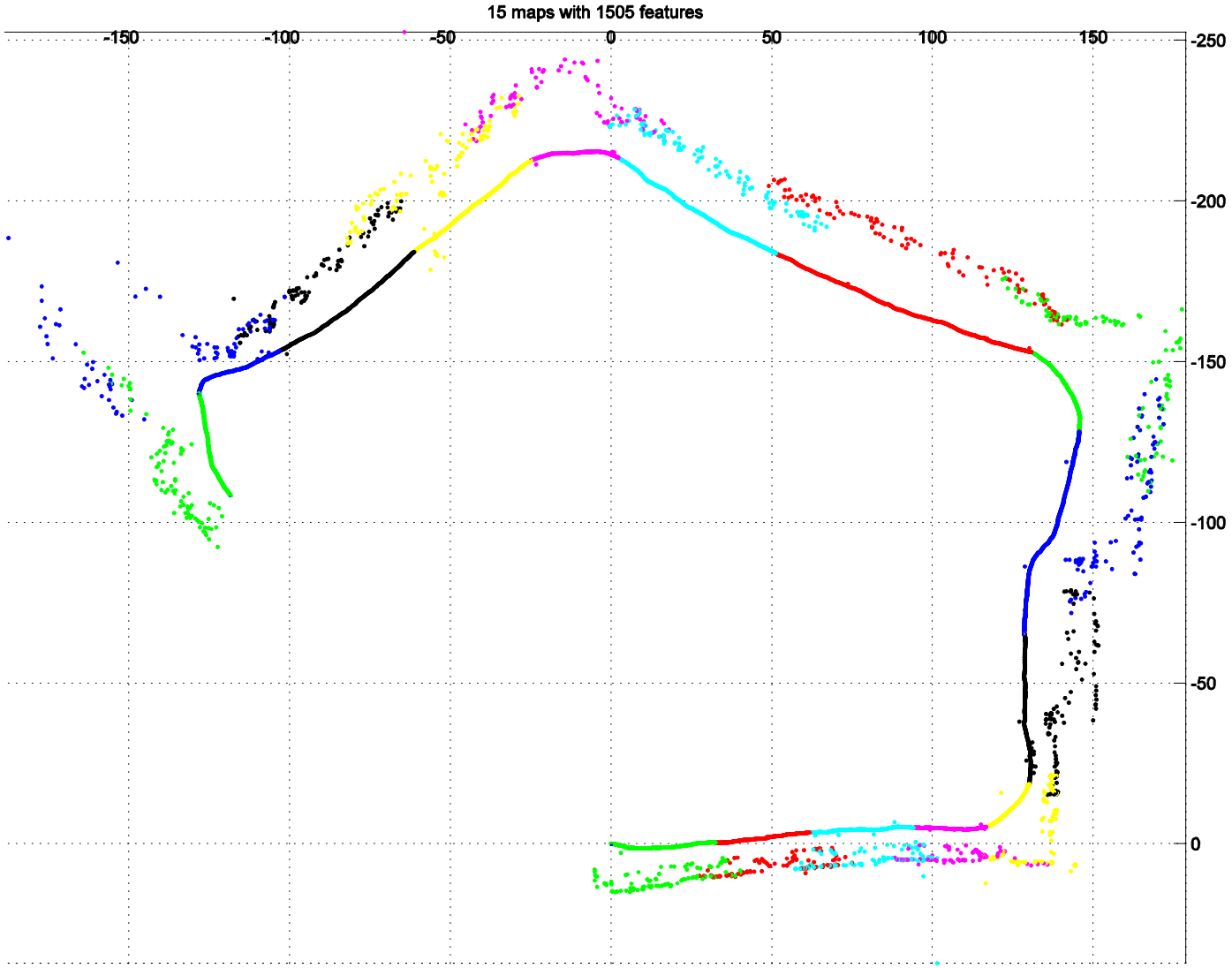


Sequence of local maps



The scale is arbitrary (not observable)

With scale compensation



Map Matching

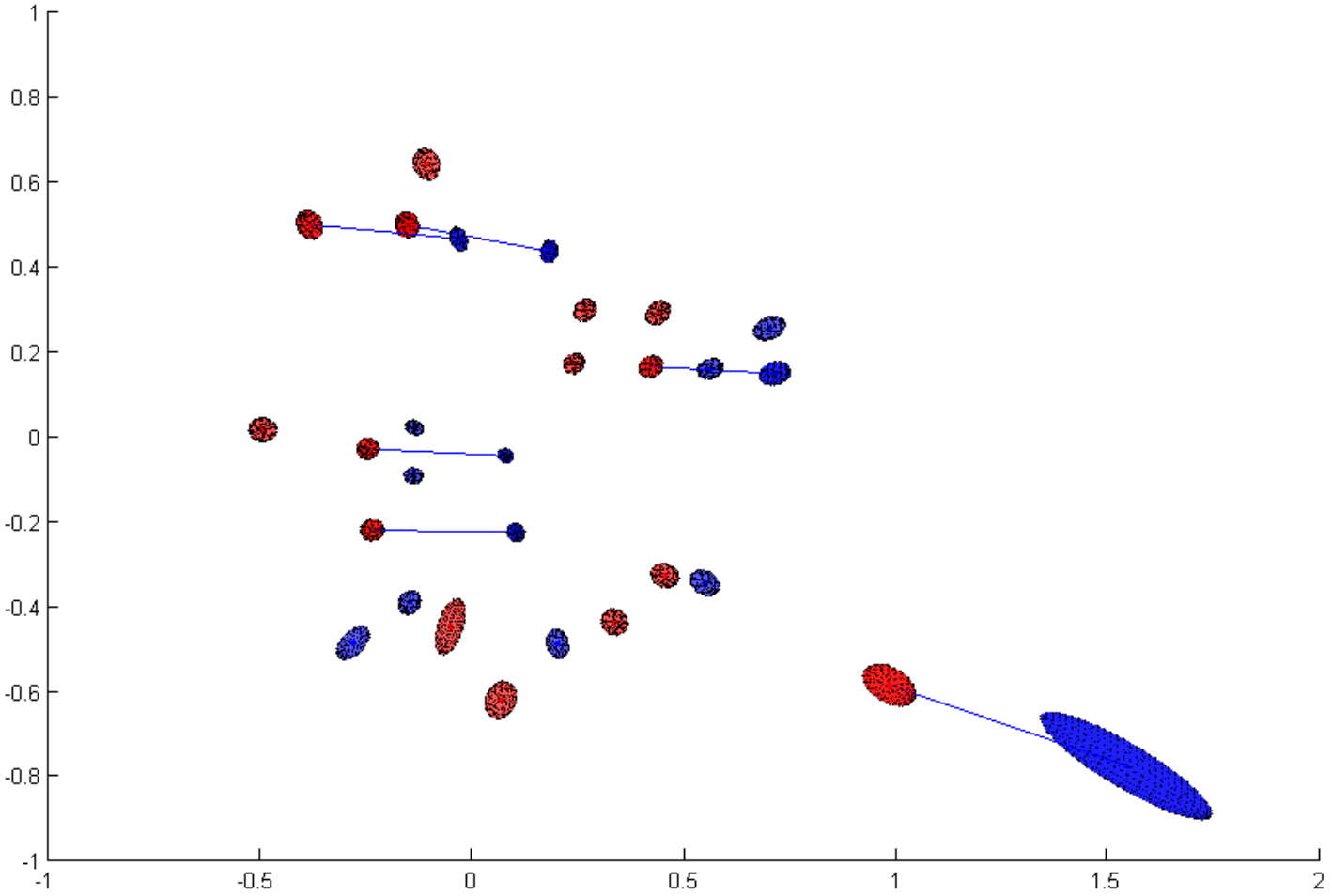
Unary Constraints

Cloud 1

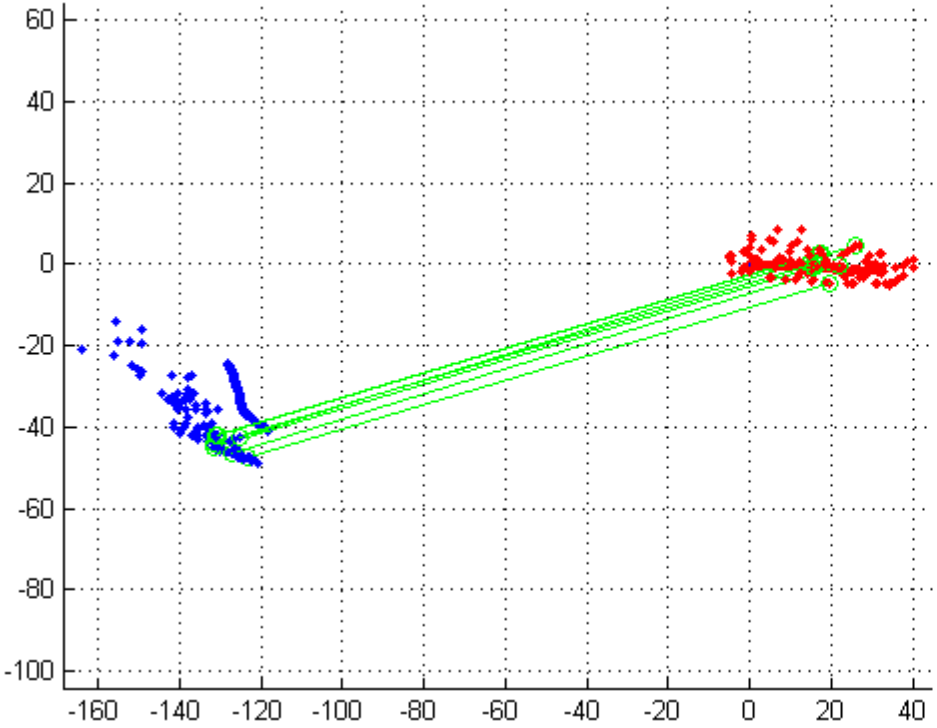
Cloud 2

Map Matching

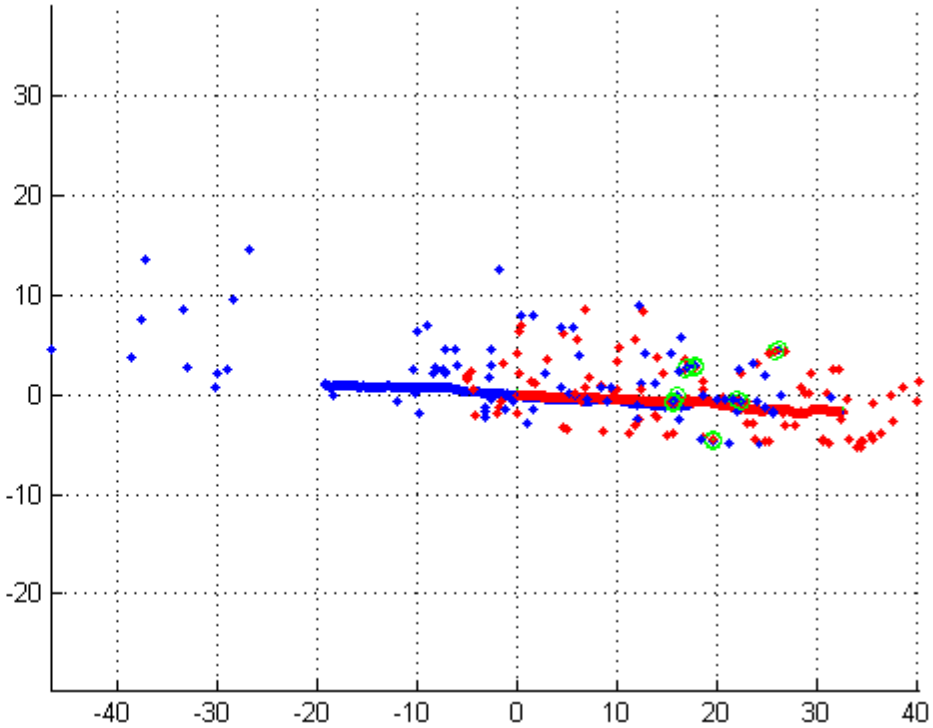
Binary Constraints



Map-to-Map Matching



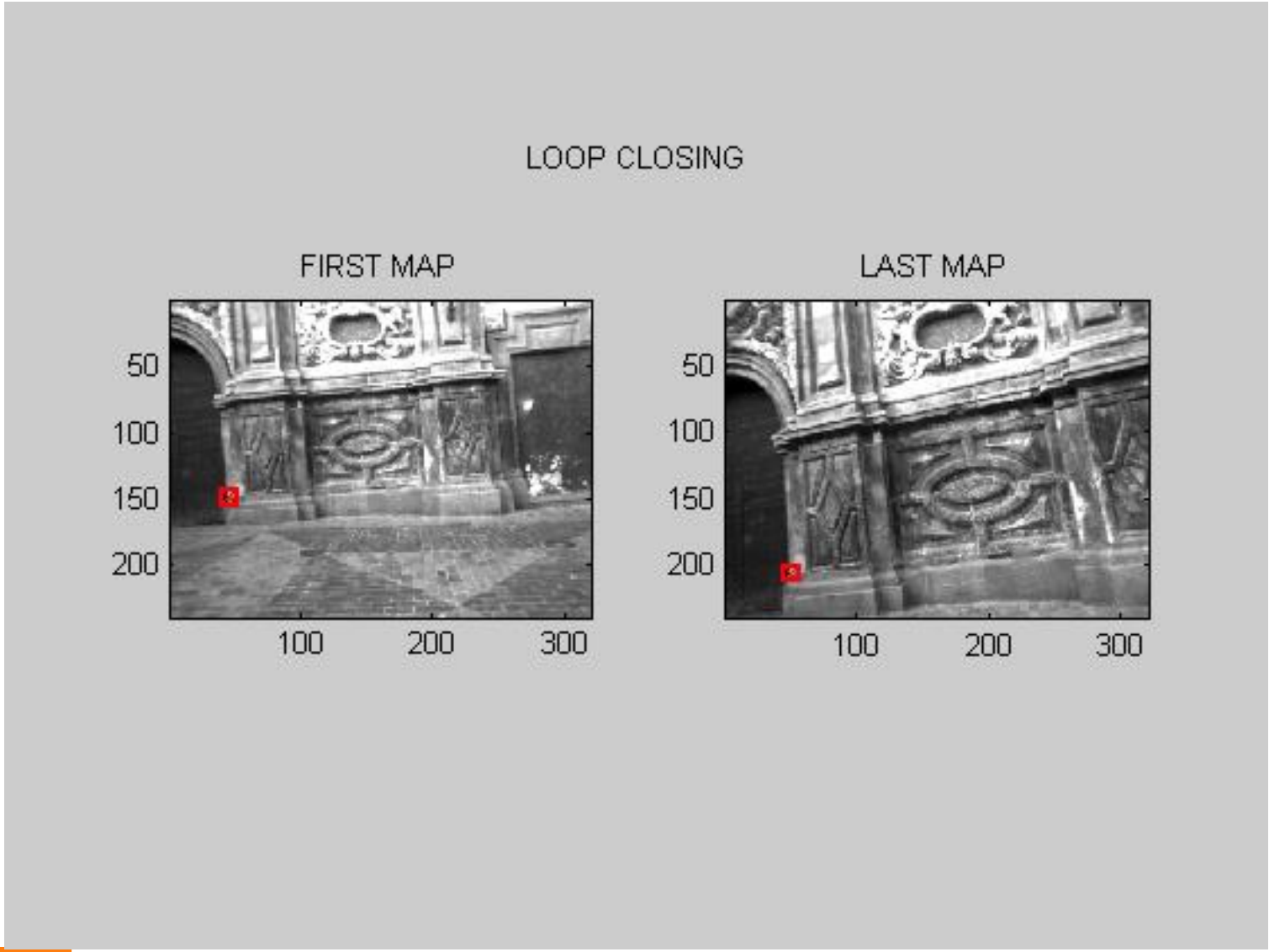
Matchings found



Aligned Submaps

ROBOTICS
UNIVERSIDAD DE ZARAGOZA

Map Matching



Nonlinear constrained optimization

- Minimize corrections to the global map, subject to the loop constraint:

$$\min_{\mathbf{x}} \frac{1}{2}(\mathbf{x}-\hat{\mathbf{x}})^T \mathbf{P}^{-1}(\mathbf{x}-\hat{\mathbf{x}})$$
$$\mathbf{h}(\mathbf{x}) = \mathbf{0}$$

- Sequential Quadratic Programming (SQP) :

$$\mathbf{H}_i = \left[\left.\frac{\partial \mathbf{h}}{\partial \mathbf{x}_1}\right|_{\hat{\mathbf{x}}_i} \left.\frac{\partial \mathbf{h}}{\partial \mathbf{x}_2}\right|_{\hat{\mathbf{x}}_i} \cdots \left.\frac{\partial \mathbf{h}}{\partial \mathbf{x}_{n-1}}\right|_{\hat{\mathbf{x}}_i} \left.\frac{\partial \mathbf{h}}{\partial \mathbf{x}_n}\right|_{\hat{\mathbf{x}}_i} \right]$$

$$\mathbf{P}_i = \mathbf{P}_0 - \mathbf{P}_0\mathbf{H}_i^T \left(\mathbf{H}_i\mathbf{P}_0\mathbf{H}_i^T\right)^{-1} \mathbf{H}_i\mathbf{P}_0$$

$$\hat{\mathbf{x}}_{i+1} = \hat{\mathbf{x}}_i - \mathbf{P}_i\mathbf{P}_0^{-1}(\hat{\mathbf{x}}_i - \hat{\mathbf{x}}_0) - \mathbf{P}_0\mathbf{H}_i^T \left(\mathbf{H}_i\mathbf{P}_0\mathbf{H}_i^T\right)^{-1} \hat{\mathbf{h}}_i$$

» Iterate until convergence

ROBOTICS
UNIVERSIDAD DE ZARAGOZA

Nonlinear constrained optimization

- A more efficient version:

$$\hat{\mathbf{x}}_{i+1} = \hat{\mathbf{x}}_0 + \mathbf{P}_0 \mathbf{H}_i^T \left(\mathbf{H}_i \mathbf{P}_0 \mathbf{H}_i^T \right)^{-1} \left(\mathbf{H}_i \left(\hat{\mathbf{x}}_i - \hat{\mathbf{x}}_0 \right) - \hat{\mathbf{h}}_i \right)$$

  » Iterate until convergence

- Complexity:
  - P_0 is block diagonal
  - H_i is sparse with nonzeros only for the maps in the loop
  - The iteration is linear with the number of maps in the loop
- Convergence:
  - Converges in 2 or 3 iterations (for loops around 300m)
  - For bigger errors, may it converge to a local minimum ??

Solution 2 (for EKF fans)

- We could impose the loop constraints using an imprecise measurement function:

$$\mathbf{z} = \mathbf{h}(\mathbf{x}) + \mathbf{w} = \mathbf{0}$$

- With Covariance:

$$\mathbf{P}_z = Cov(\mathbf{w}) = \begin{bmatrix} \mathbf{P}_{z_1} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{P}_{z_2} & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{P}_{z_l} \end{bmatrix}$$

- Estimated errors in closing the loops:

$$\mathbf{h}(\hat{\mathbf{x}}) = \begin{bmatrix} \mathbf{h}_1(\hat{\mathbf{x}}) \\ \mathbf{h}_2(\hat{\mathbf{x}}) \\ \vdots \\ \mathbf{h}_l(\hat{\mathbf{x}}) \end{bmatrix}$$

Iterated Extended Kalman Filter

- Jacobian of the measurement function:

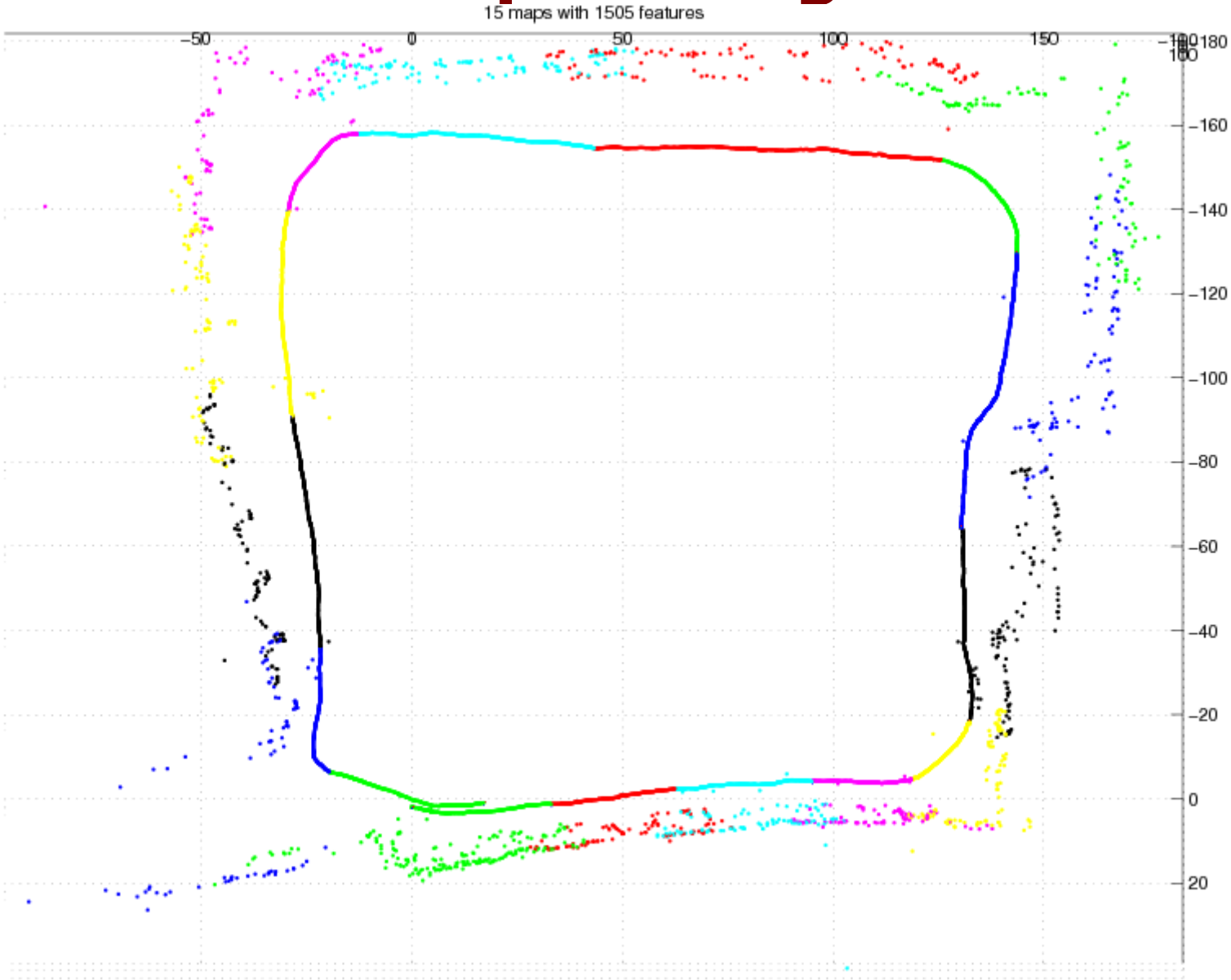
$$\mathbf{H}_i = \left[\left.\frac{\partial \mathbf{h}}{\partial \mathbf{x}_1}\right|_{\hat{\mathbf{x}}_i} \left.\frac{\partial \mathbf{h}}{\partial \mathbf{x}_2}\right|_{\hat{\mathbf{x}}_i} \cdots \left.\frac{\partial \mathbf{h}}{\partial \mathbf{x}_{n-1}}\right|_{\hat{\mathbf{x}}_i} \left.\frac{\partial \mathbf{h}}{\partial \mathbf{x}_n}\right|_{\hat{\mathbf{x}}_i} \right]$$

- Iterated EKF equations:

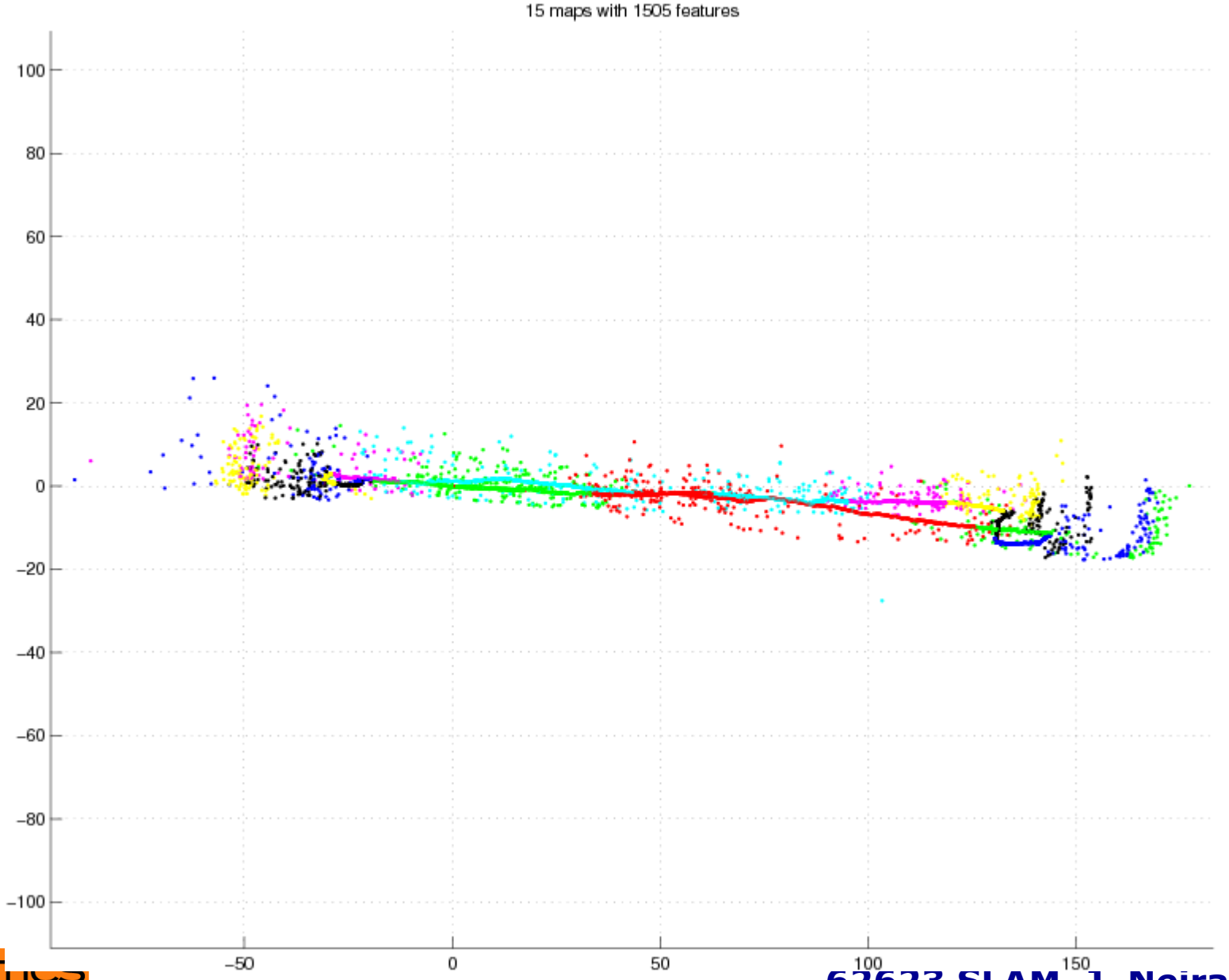
$$\begin{aligned} \mathbf{P}_i &= \mathbf{P}_0 - \mathbf{P}_0\mathbf{H}_i^T \left[\mathbf{H}_i\mathbf{P}_0\mathbf{H}_i^T + \mathbf{P}_z\right]^{-1} \mathbf{H}_i\mathbf{P}_0 \\ \hat{\mathbf{x}}_{i+1} &= \hat{\mathbf{x}}_i - \mathbf{P}_i\mathbf{P}_0^{-1}(\hat{\mathbf{x}}_i - \hat{\mathbf{x}}_0) + \mathbf{P}_0\mathbf{H}_i^T \left(\mathbf{H}_i\mathbf{P}_0\mathbf{H}_i^T + \mathbf{P}_z\right)^{-1} \left(\mathbf{z} - \hat{\mathbf{h}}_i\right) \end{aligned}$$

- With exact loop constraint, z = 0 and P_z = 0, IEKF is equivalent to nonlinear optimization with SQP

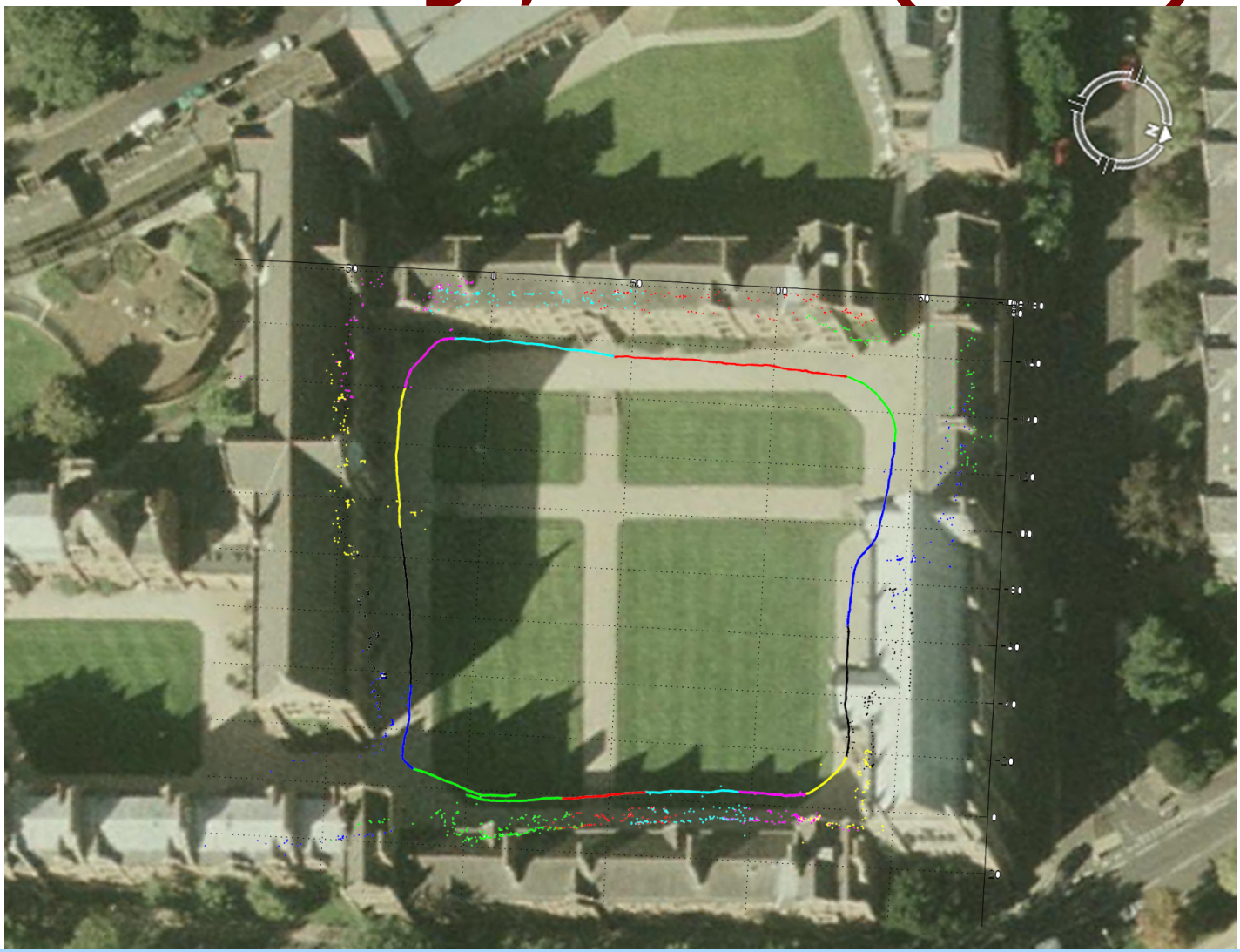
Loop closing



Loop closing (lateral view)



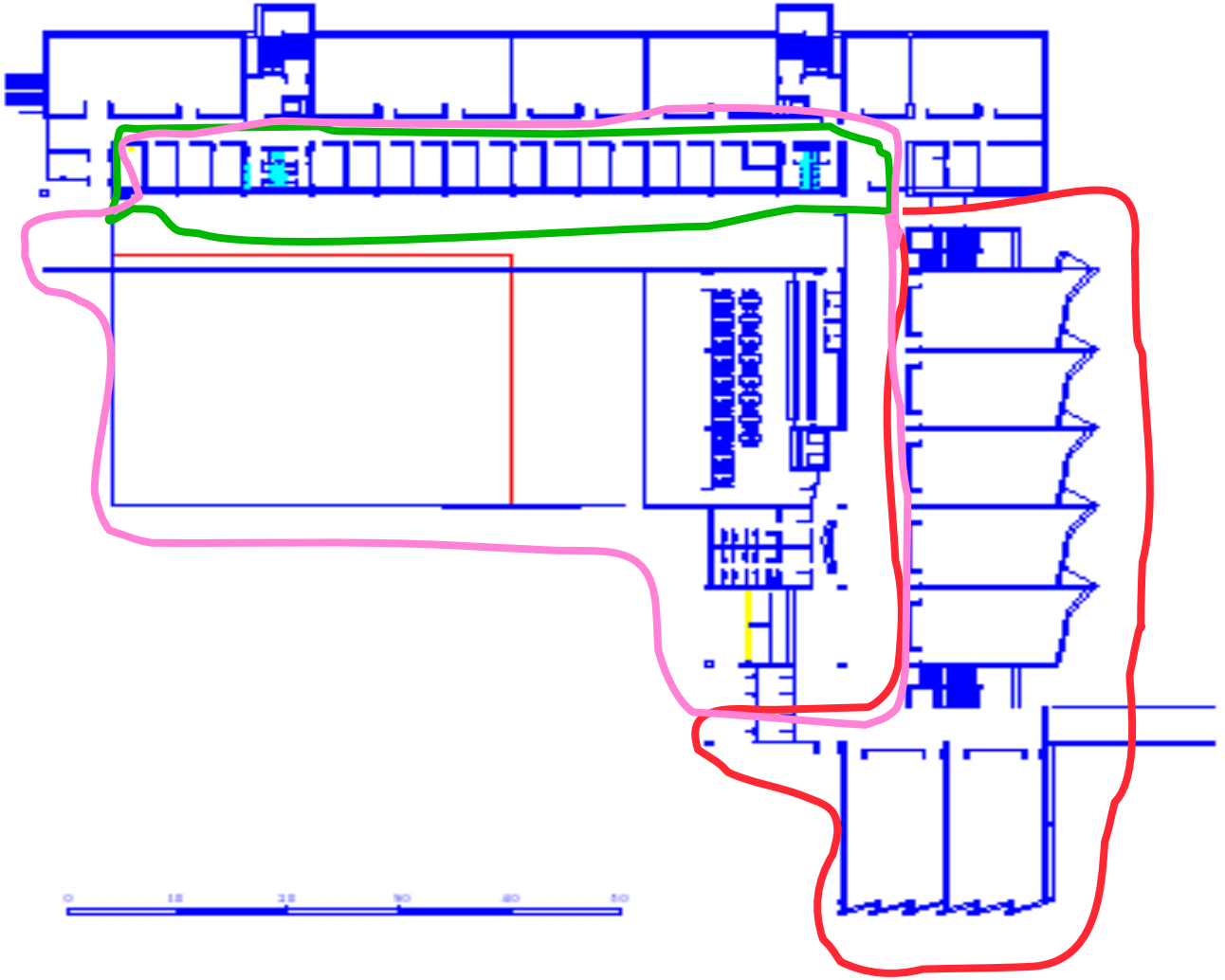
Keble College, Oxford (290m)



L. Clemente, A. Davison, I. Reid, J. Neira and J.D. Tardós **Mapping Large Loops with a Single Hand-Held Camera**. Robotics: Science and Systems, 2007.

Multivehicle SLAM

- Experiment



Multivehicle SLAM

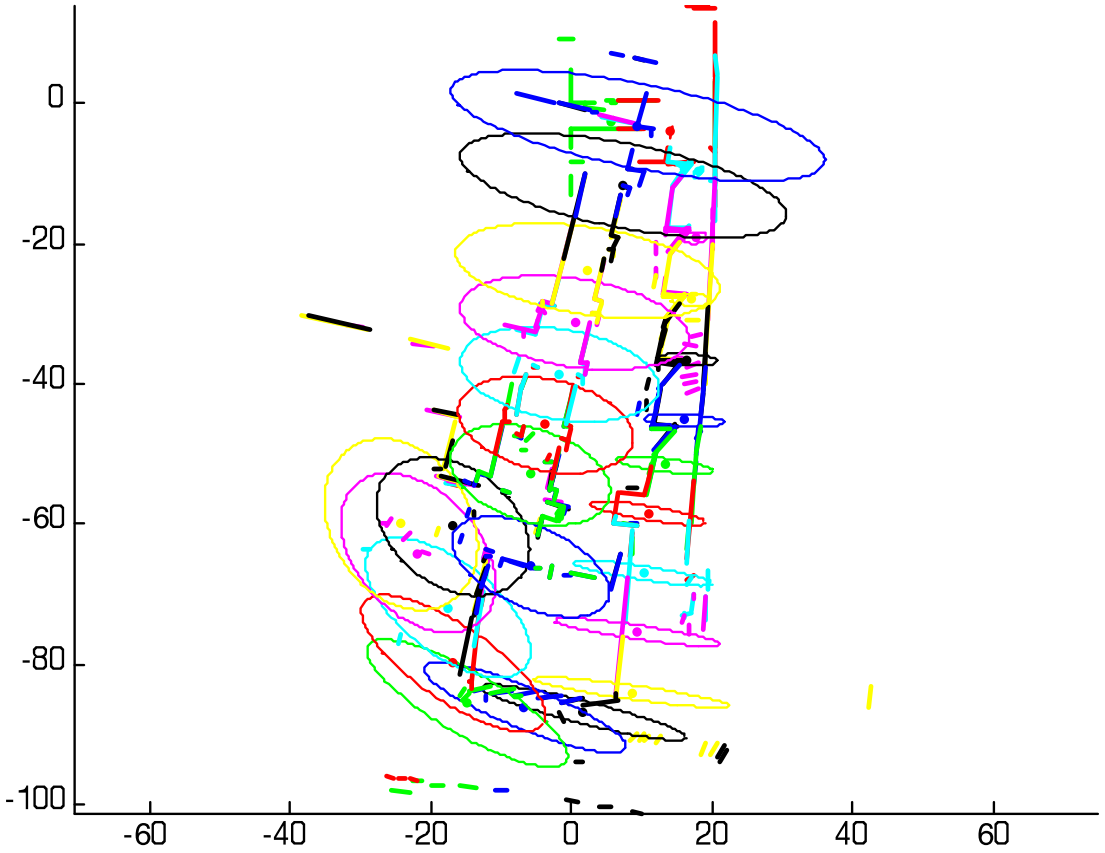
- Generalization to closing several loops simultaneously:

$$\mathbf{h}_j(\mathbf{x}) = \mathbf{x}_{j_1} \oplus \mathbf{x}_{j_2} \oplus \cdots \oplus \mathbf{x}_{j_{n_j-1}} \oplus \mathbf{x}_{j_{n_j}} = 0$$

$$\mathbf{h} = \begin{bmatrix} \mathbf{h}_1 \\ \mathbf{h}_2 \\ \vdots \\ \mathbf{h}_l \end{bmatrix}$$

Experiments

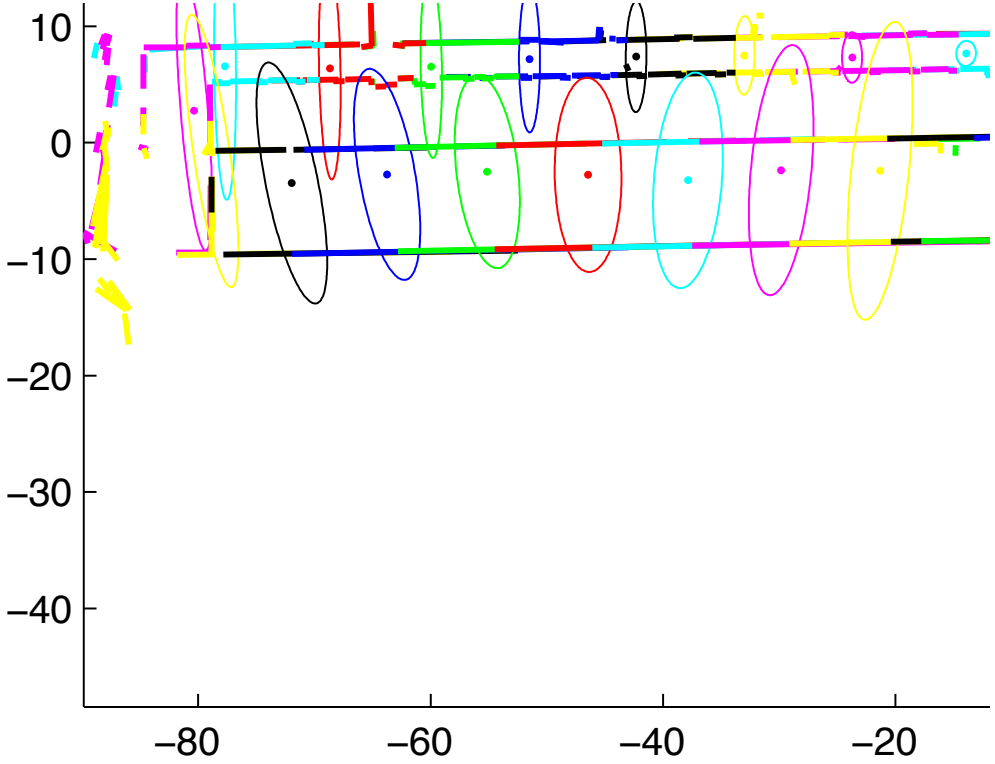
- Local maps, first robot



ROBOTICS
UNIVERSIDAD DE ZARAGOZA

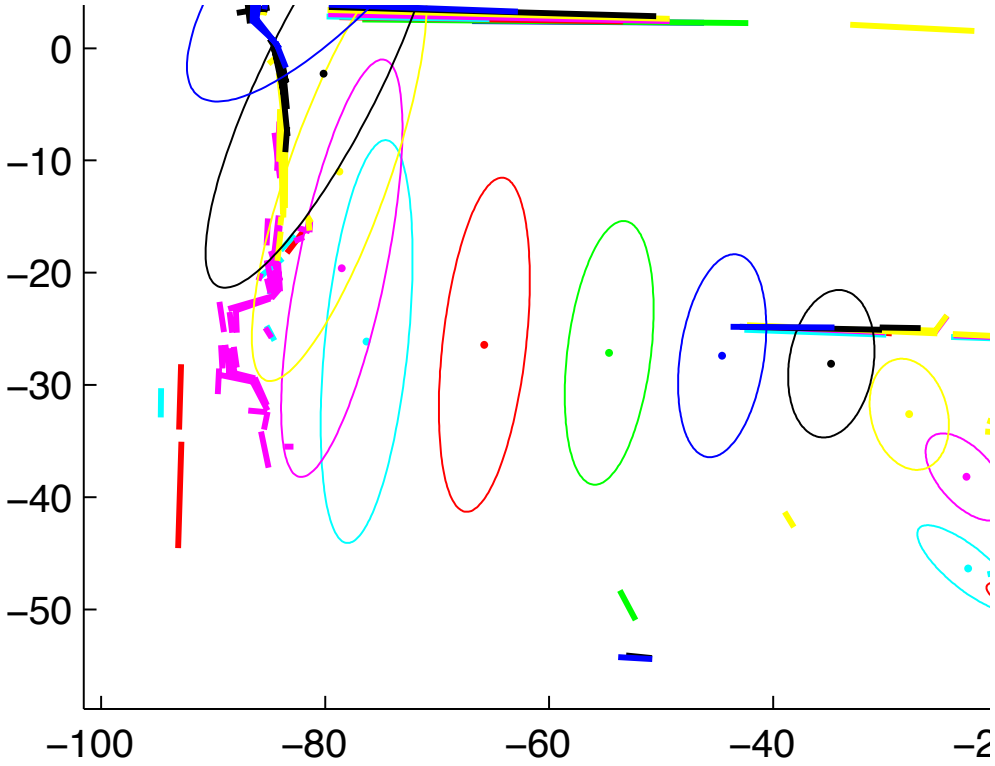
Experiments

- Local maps, second robot



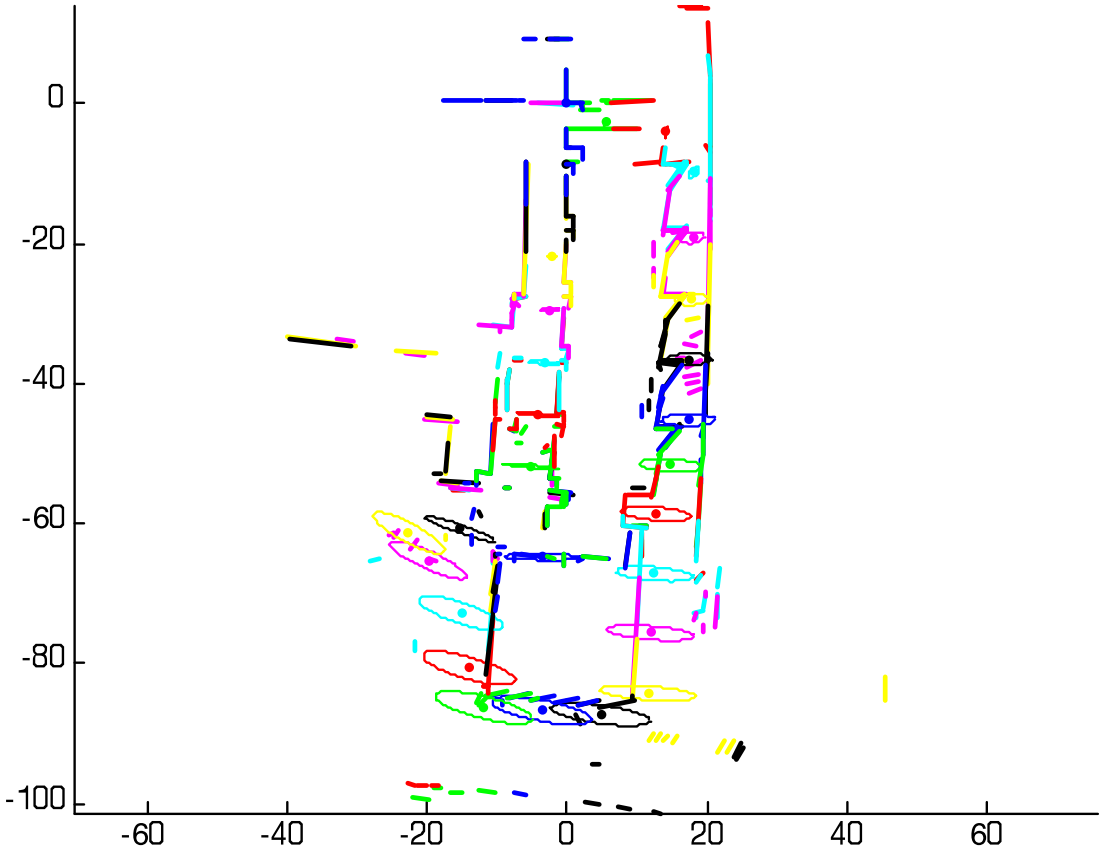
Experiments

- Local maps, third robot



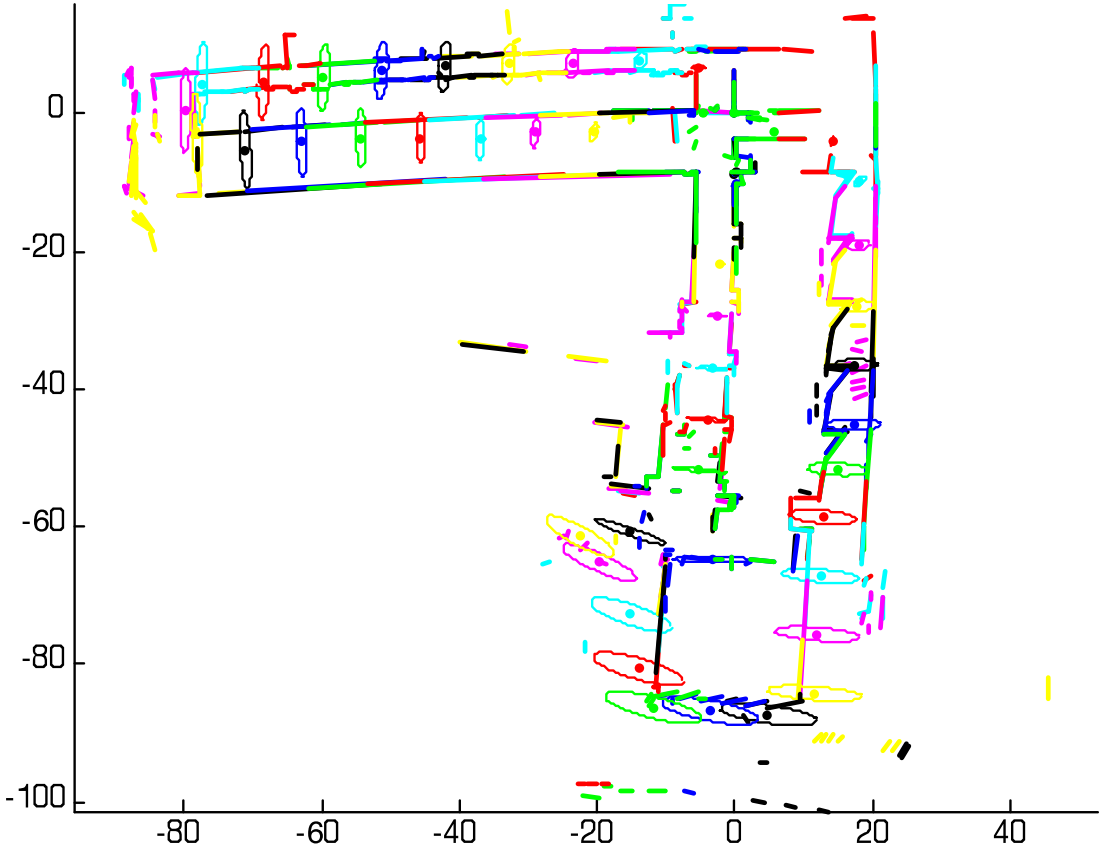
Hierarchical SLAM

- Imposing loop constraint:



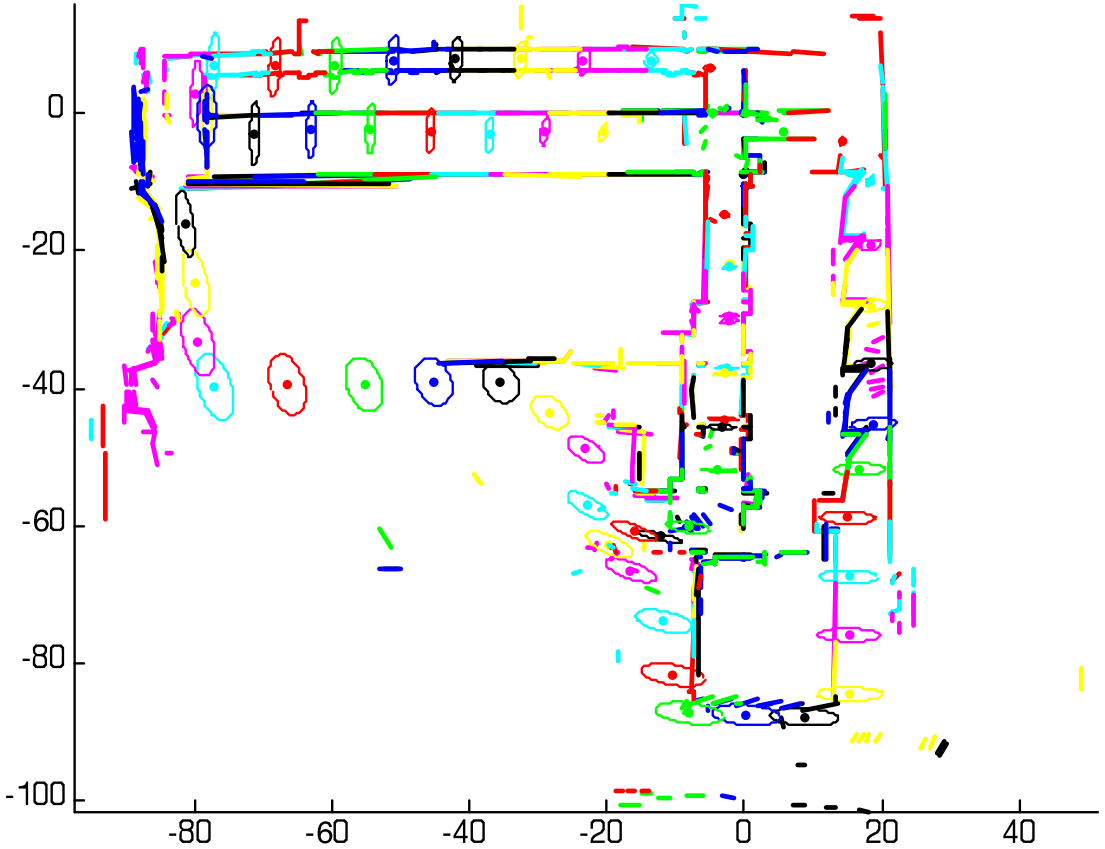
Hierarchical SLAM

- Second robot



Hierarchical SLAM

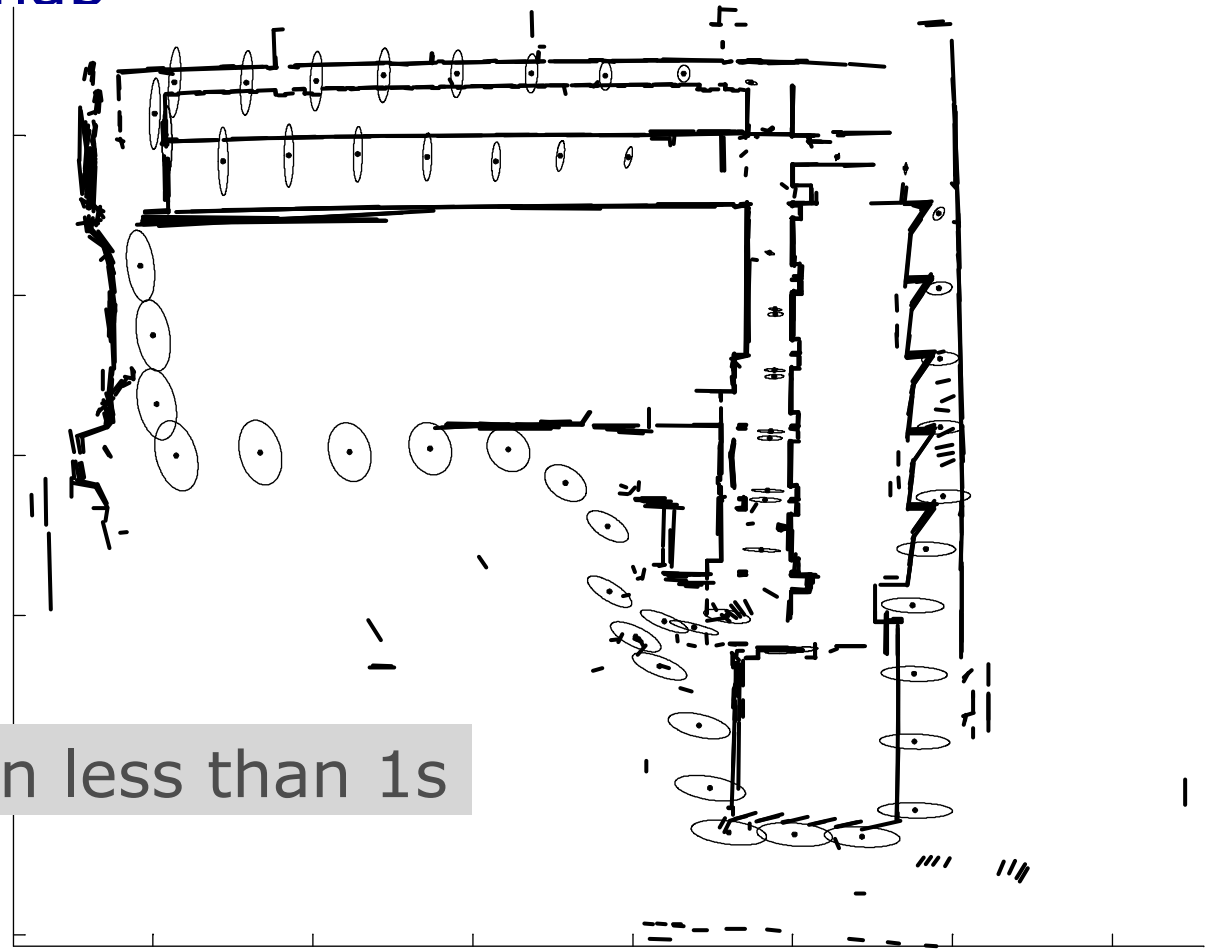
- Third robot



ROBOTICS
UNIVERSIDAD DE ZARAGOZA

Hierarchical SLAM

- Final map



Results in less than 1s

C. Estrada, J. Neira, J.D. Tardós, Hierarchical SLAM: real-time accurate map-ping of large environments. To appear in the IEEE Transactions on Robotics

ROBOTICS
UNIVERSIDAD DE ZARAGOZA

Divide and Conquer SLAM

- Experimental setup

A bumblebee, a laptop and a firewire cable

Pure Stereo SLAM

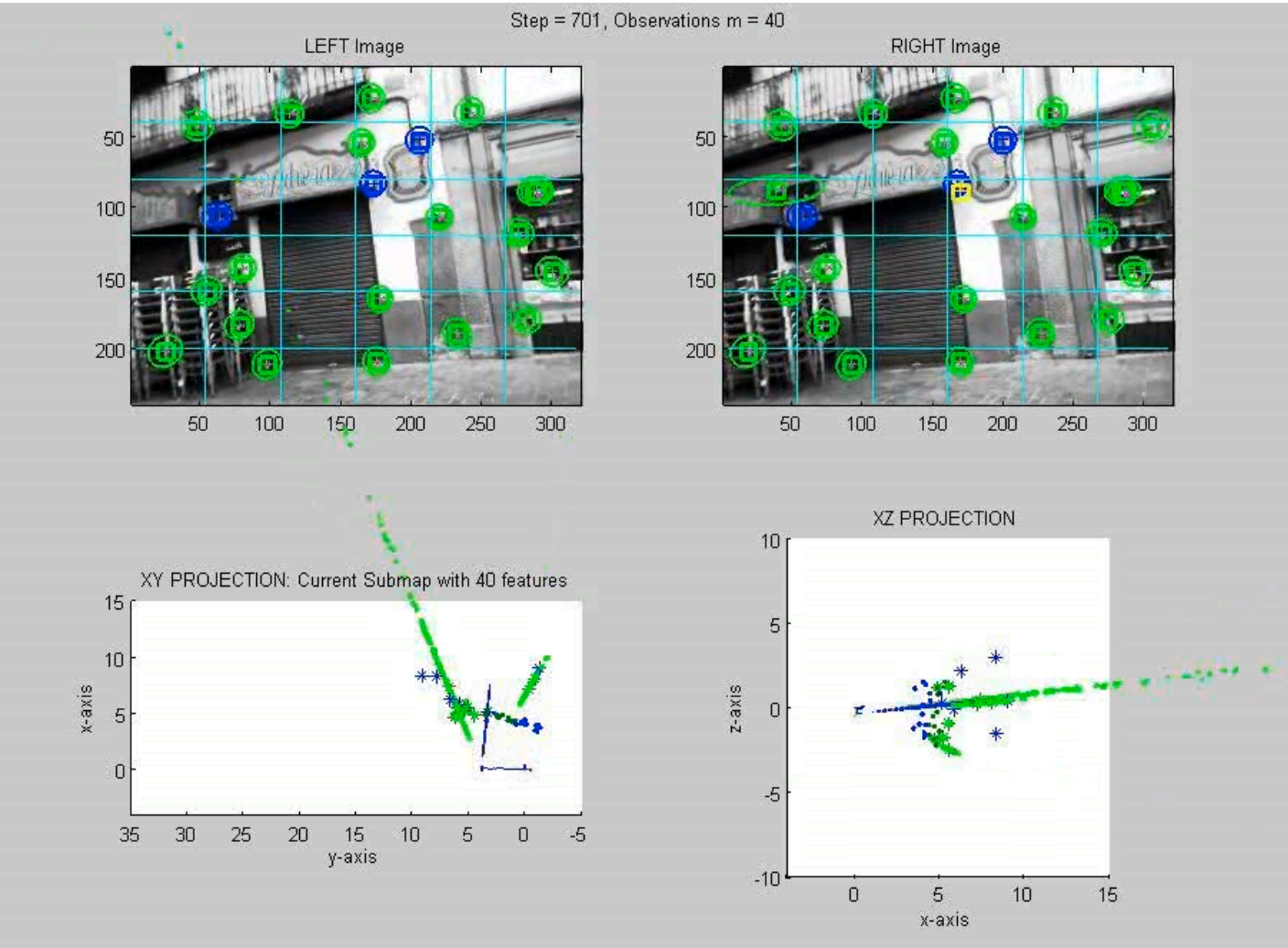
180m indoor loop, CPS, Zaragoza

ROBOTICS
UNIVERSIDAD DE ZARAGOZA

Pure Stereo SLAM

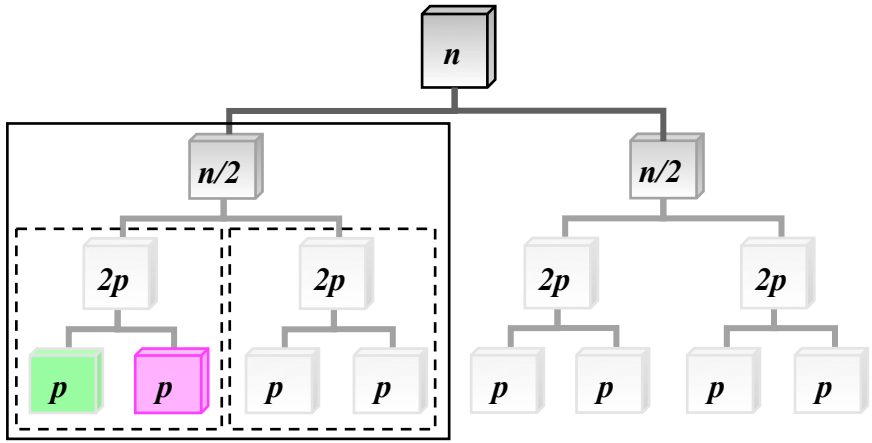
150m outdoor loop, public square, Zaragoza

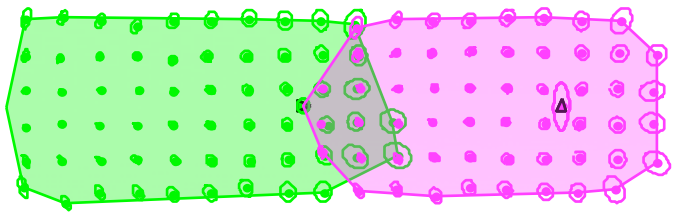
Basic EKF SLAM



Divide & Conquer SLAM

Number of Maps : 2

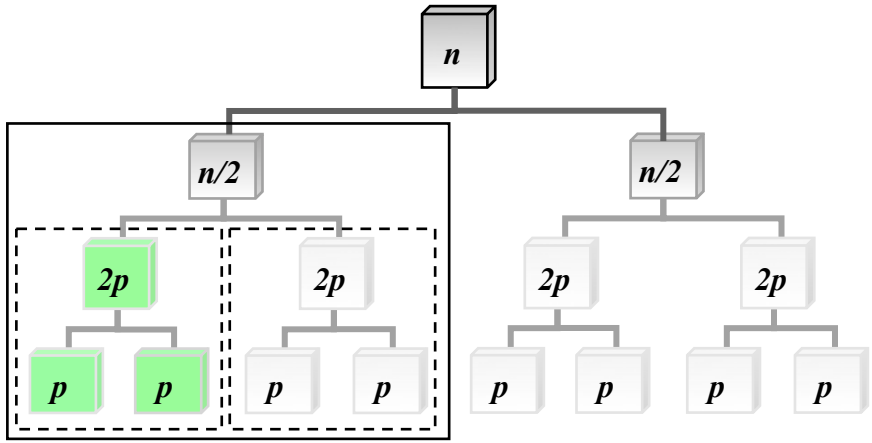




y position(m)

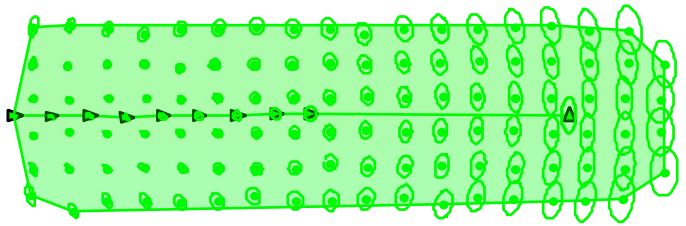
x position(m)

Divide & Conquer SLAM



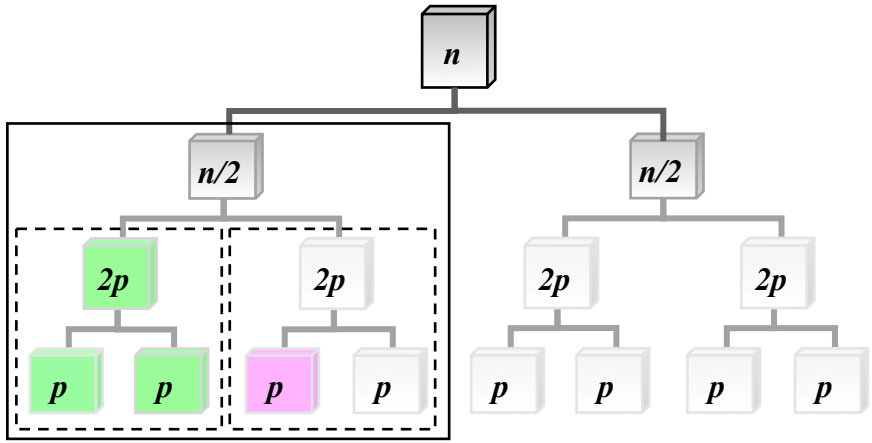
Number of Maps : 1

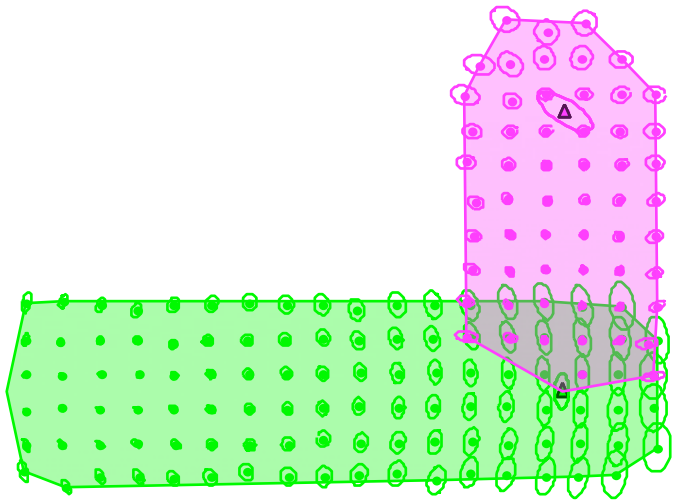
y position(m)



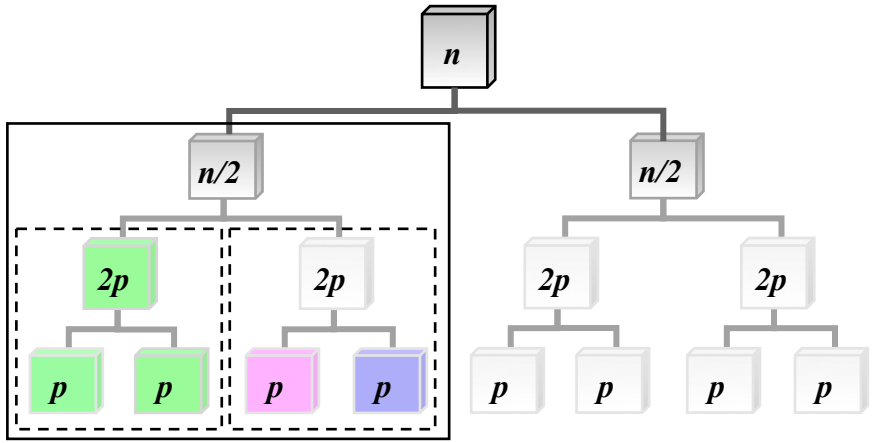
x position(m)

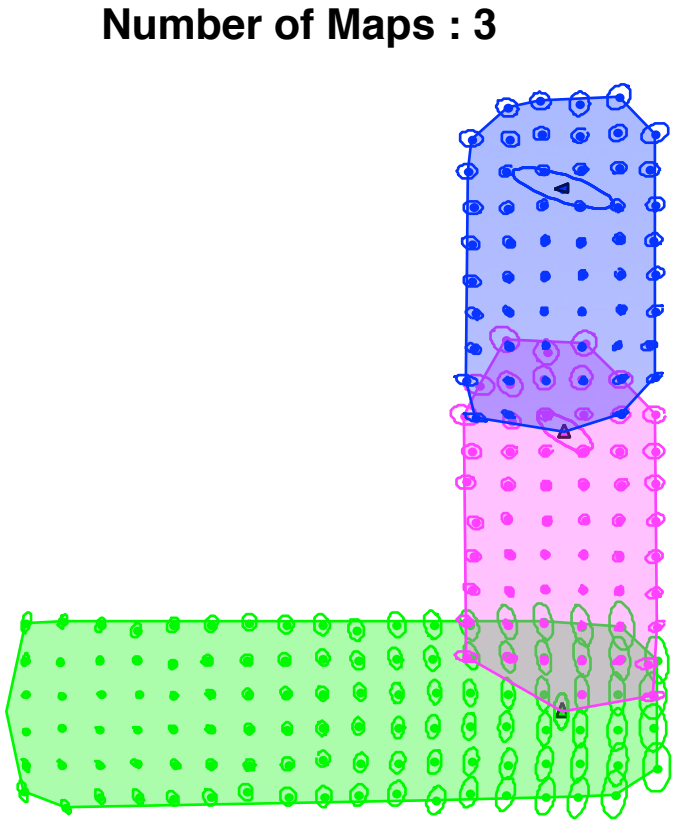
Divide & Conquer SLAM



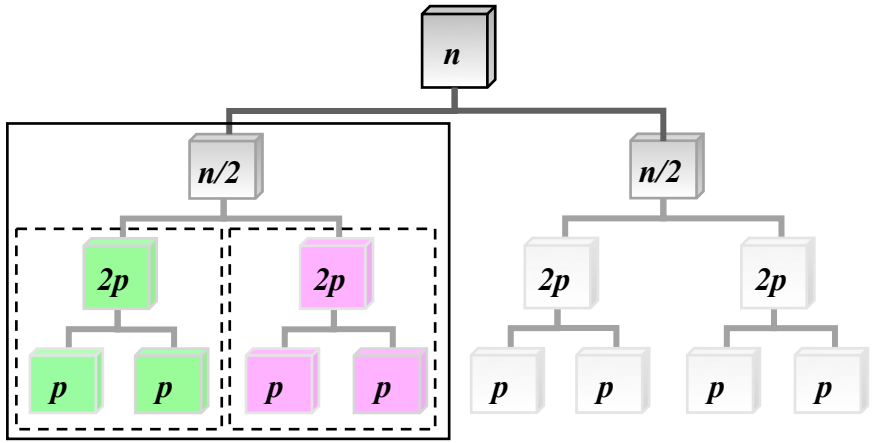


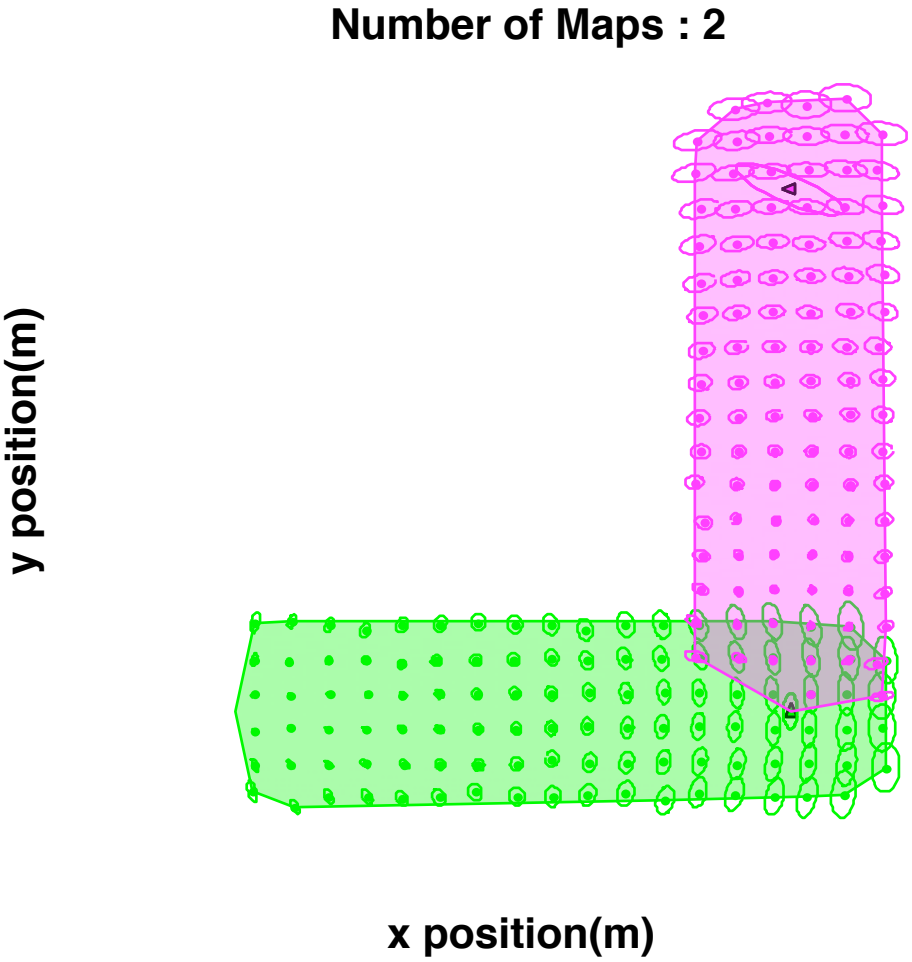
Divide & Conquer SLAM



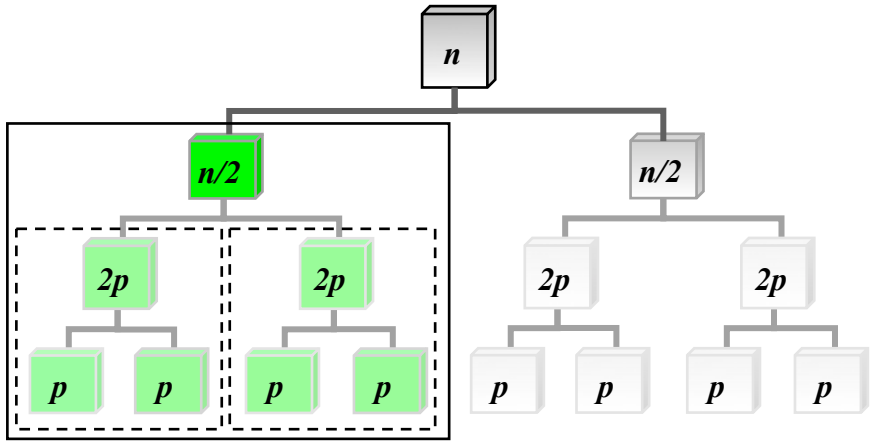


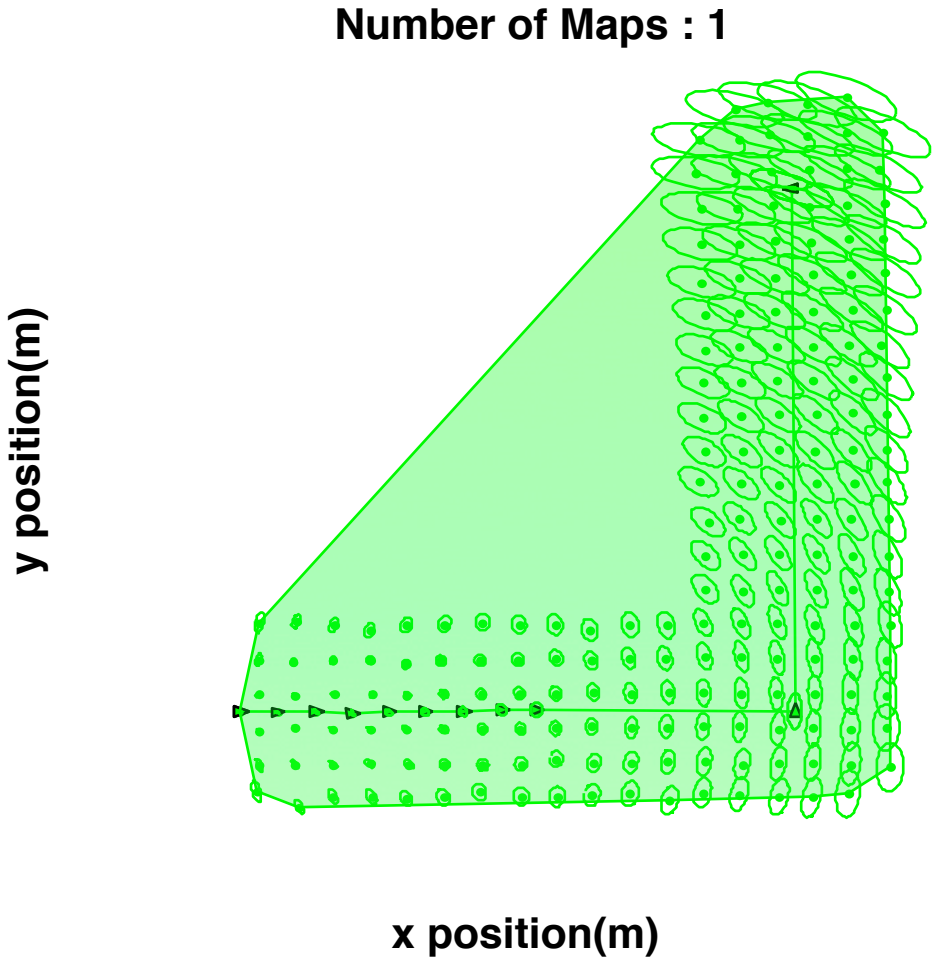
Divide & Conquer SLAM



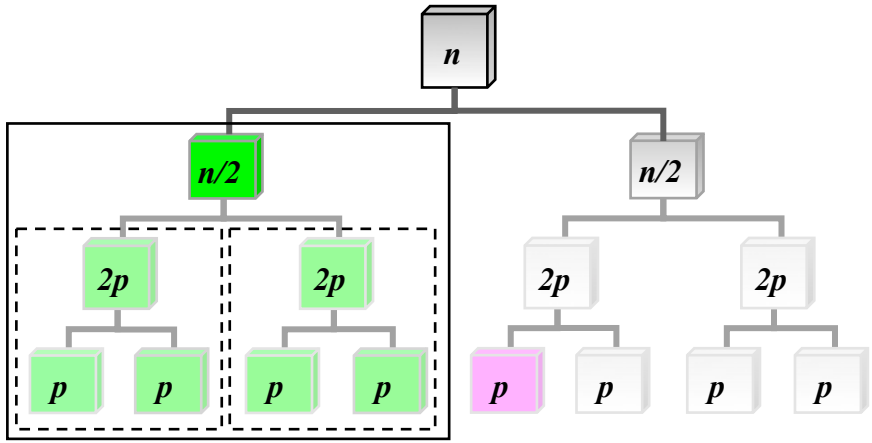


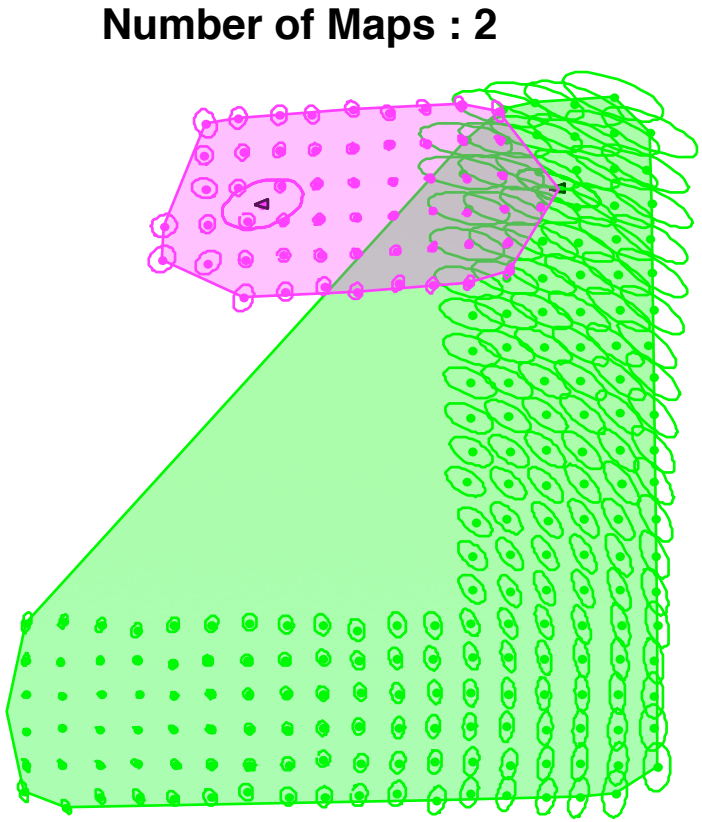
Divide & Conquer SLAM



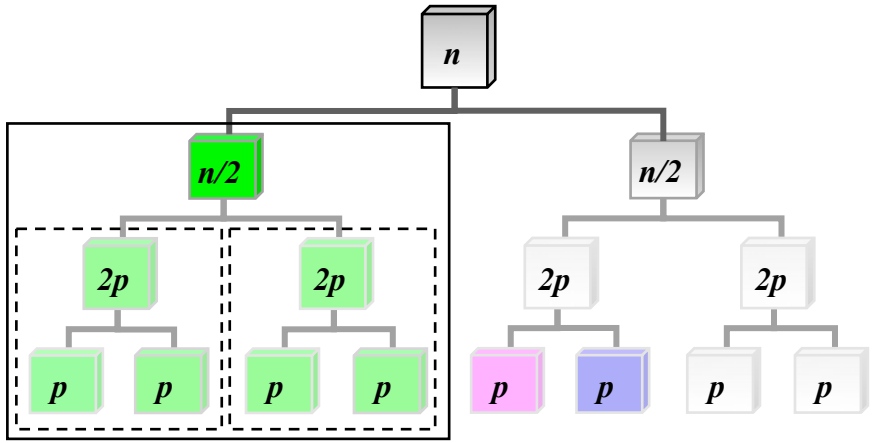


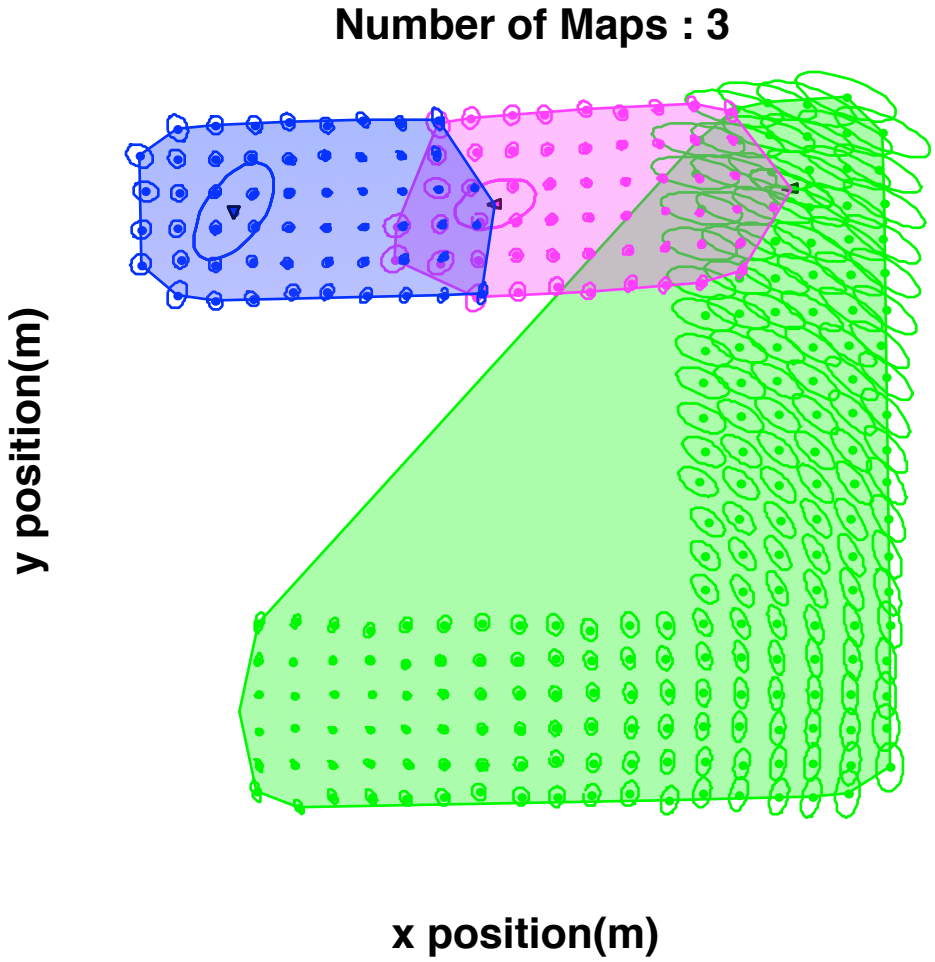
Divide & Conquer SLAM



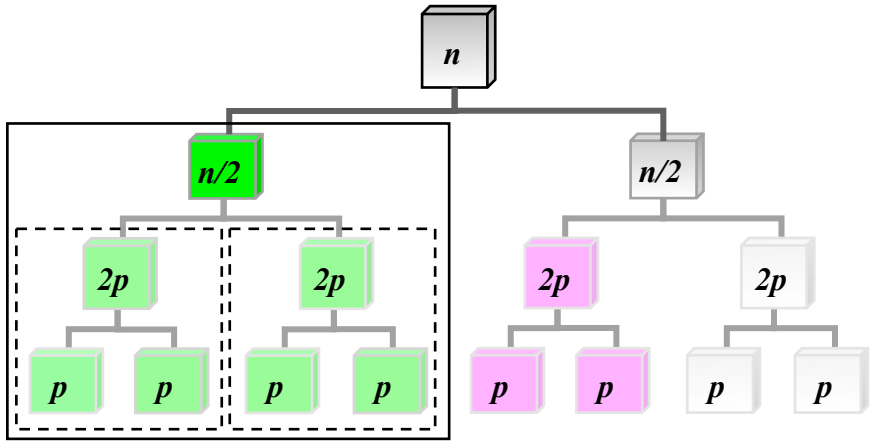


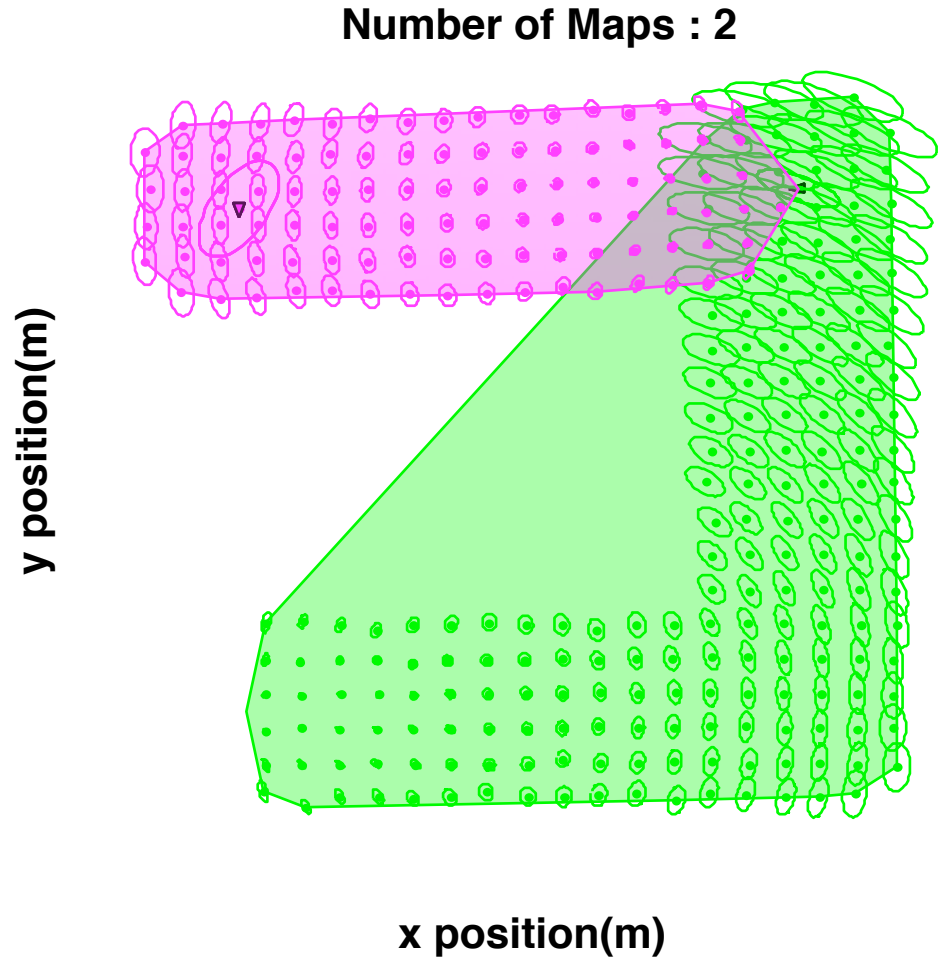
Divide & Conquer SLAM



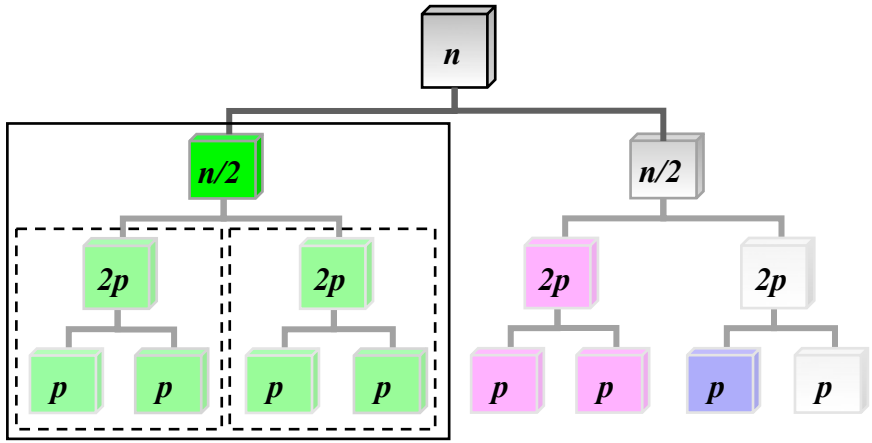


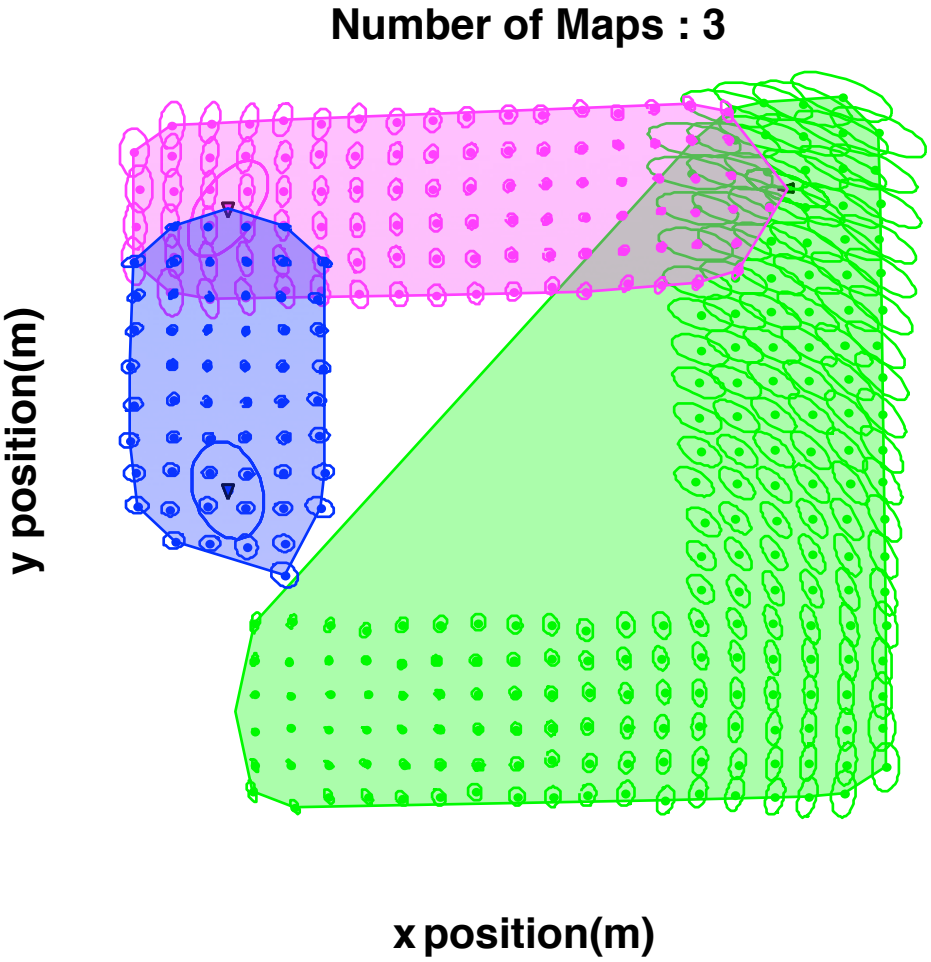
Divide & Conquer SLAM



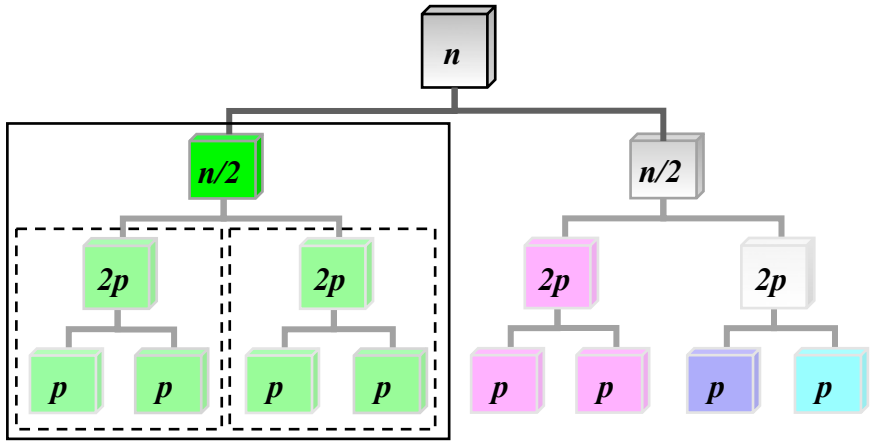


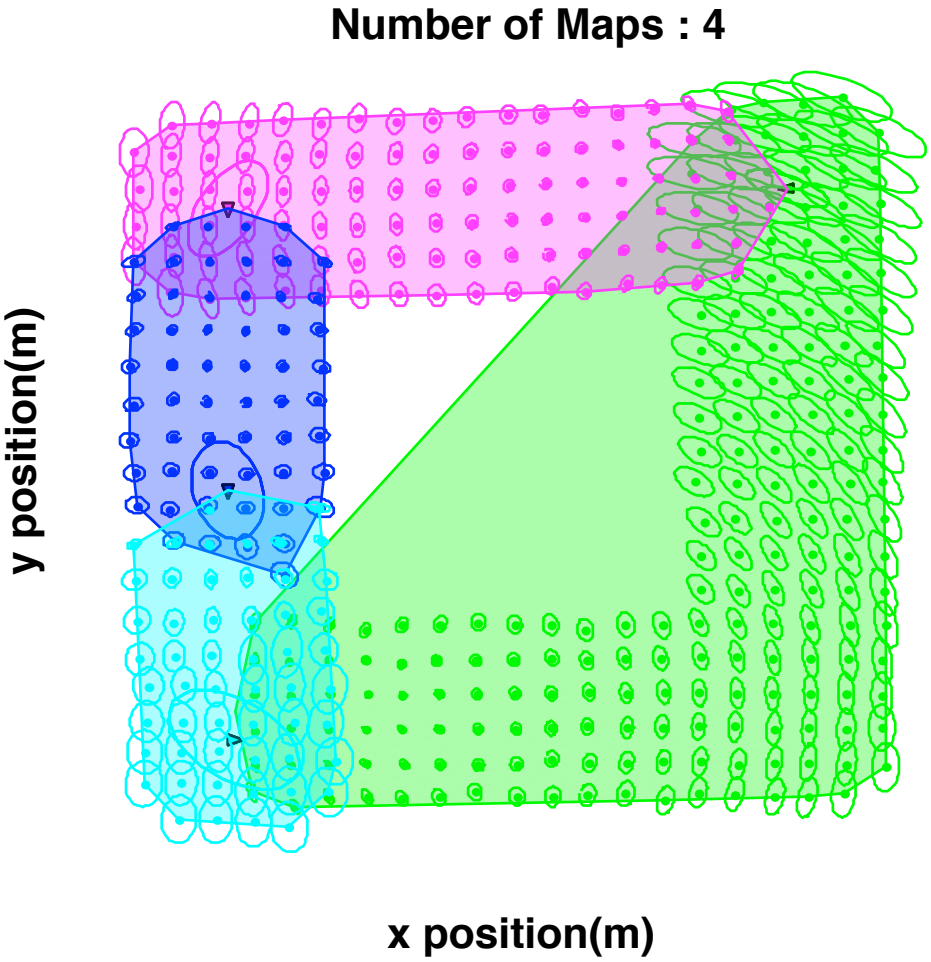
Divide & Conquer SLAM



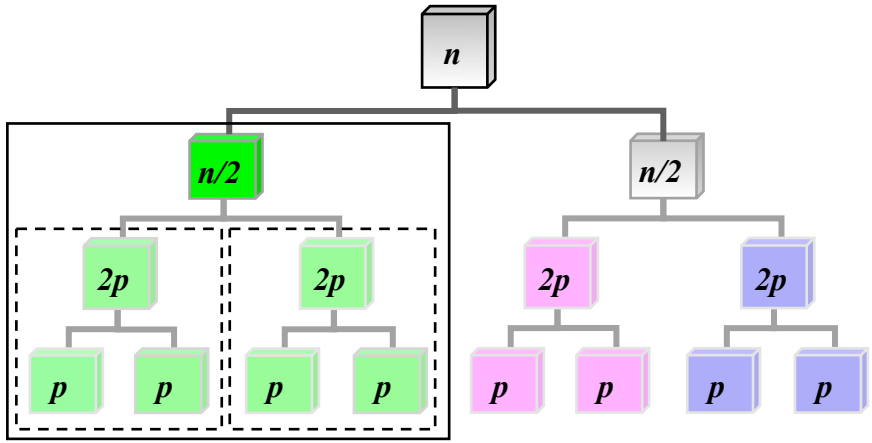


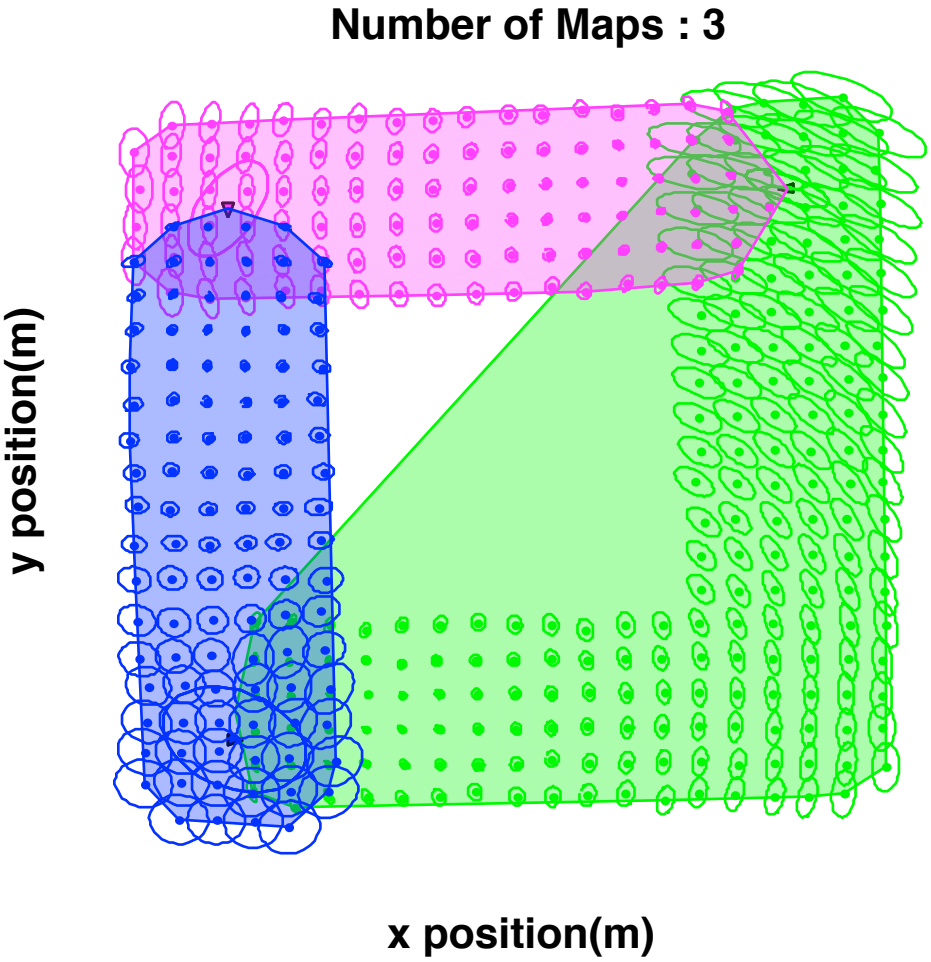
Divide & Conquer SLAM



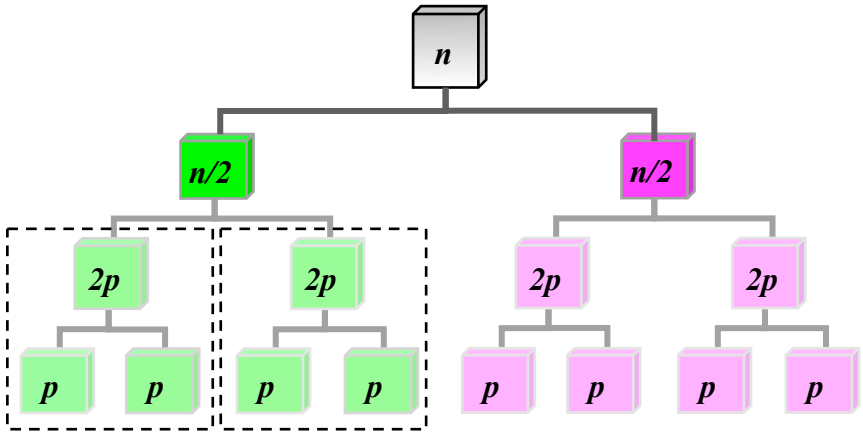


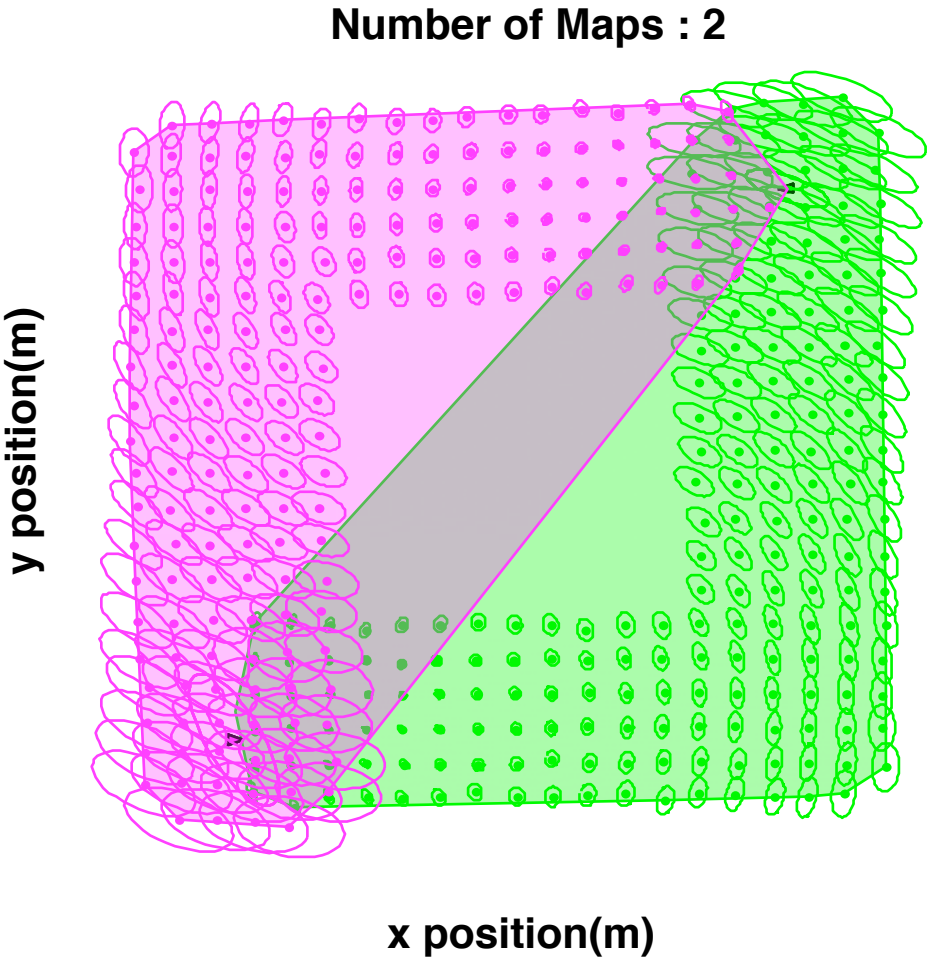
Divide & Conquer SLAM



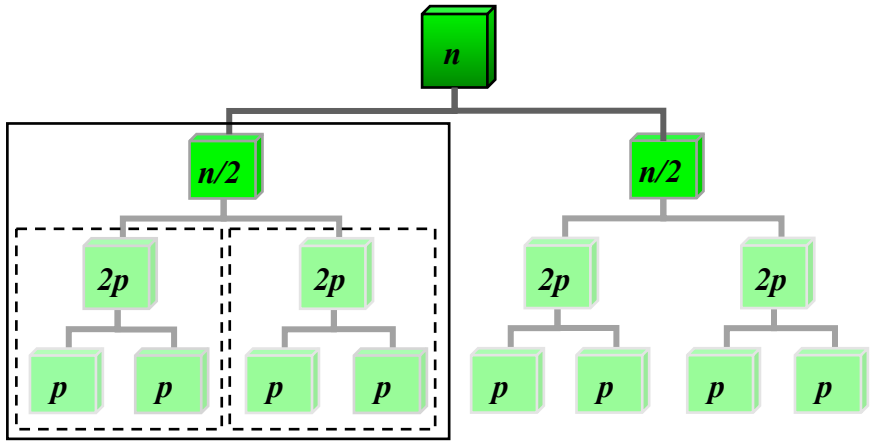


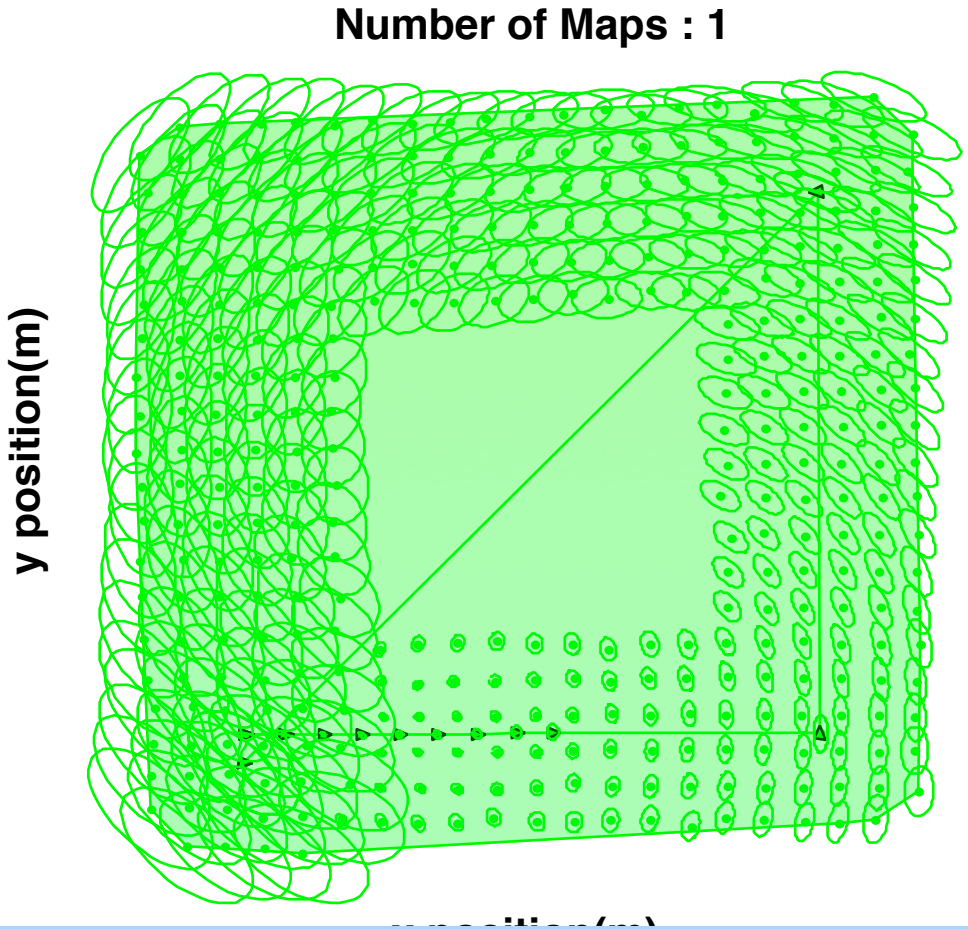
Divide & Conquer SLAM





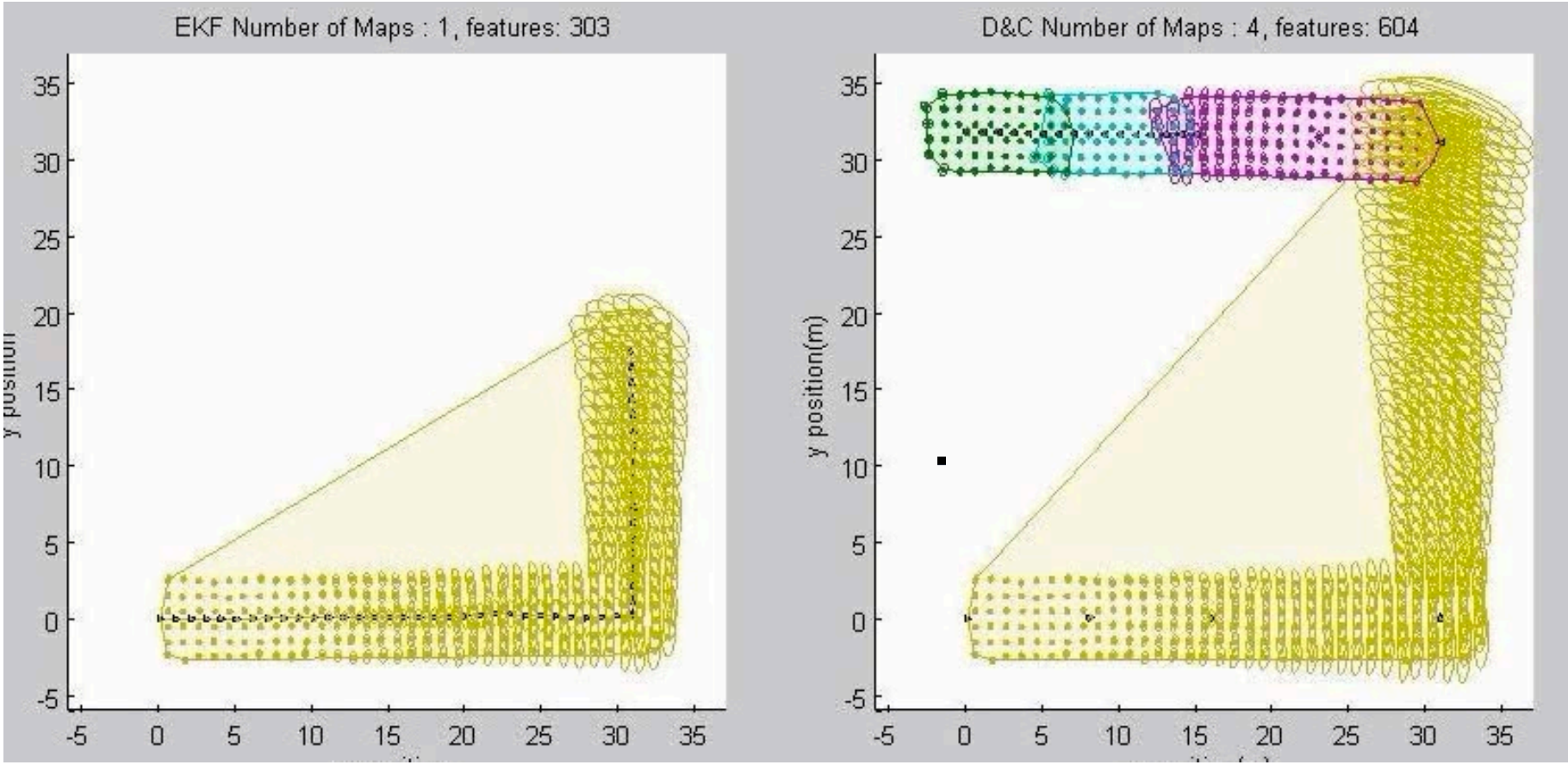
Divide & Conquer SLAM





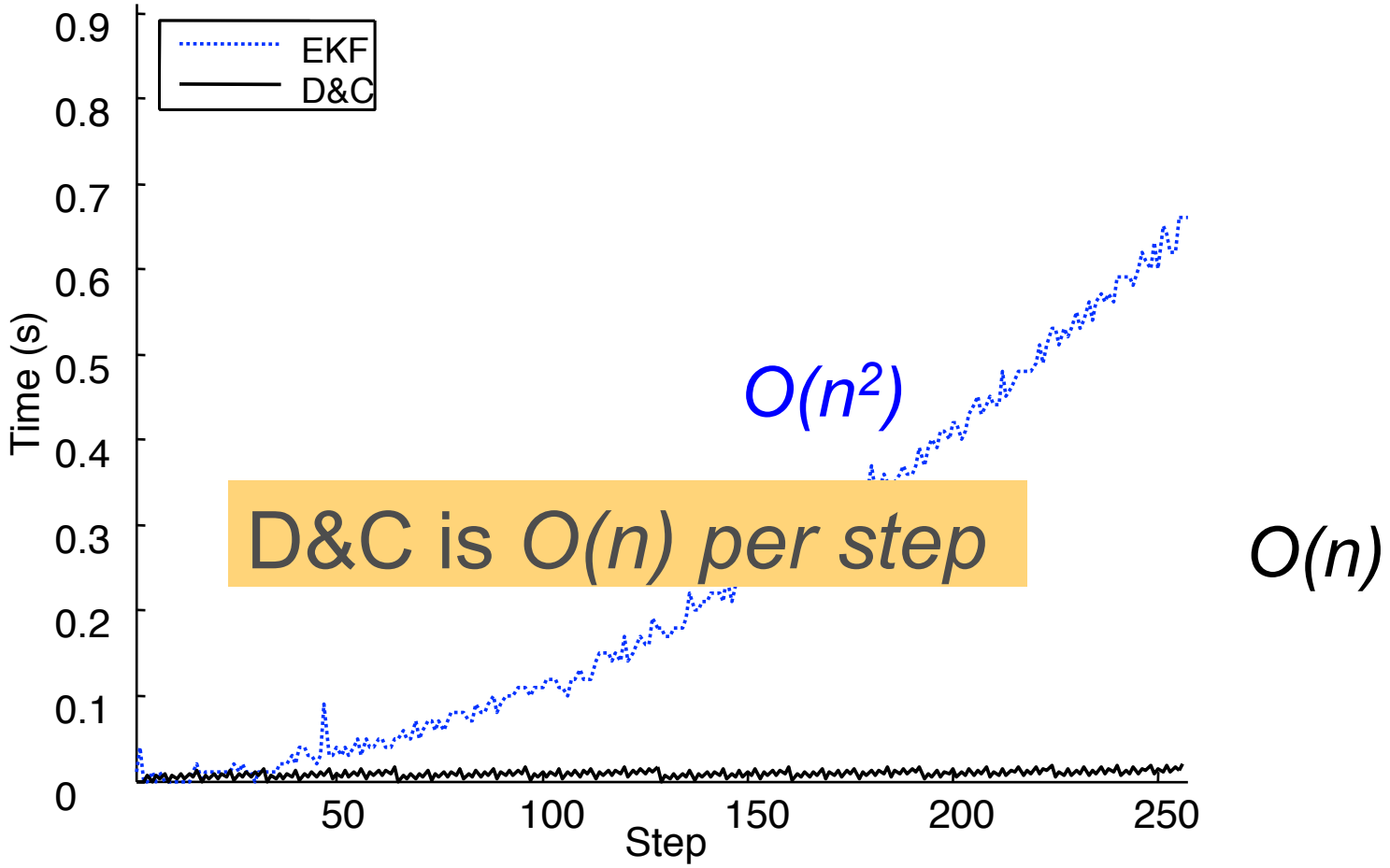
L.M. Paz, P. Jensfelt, J.D. Tardós and J. Neira. **EKF SLAM updates in O(n) with Divide and Conquer SLAM** 2007 IEEE Int. Conf. Robotics and Automation, April 10-14, Rome, Italy

Loop Trajectory

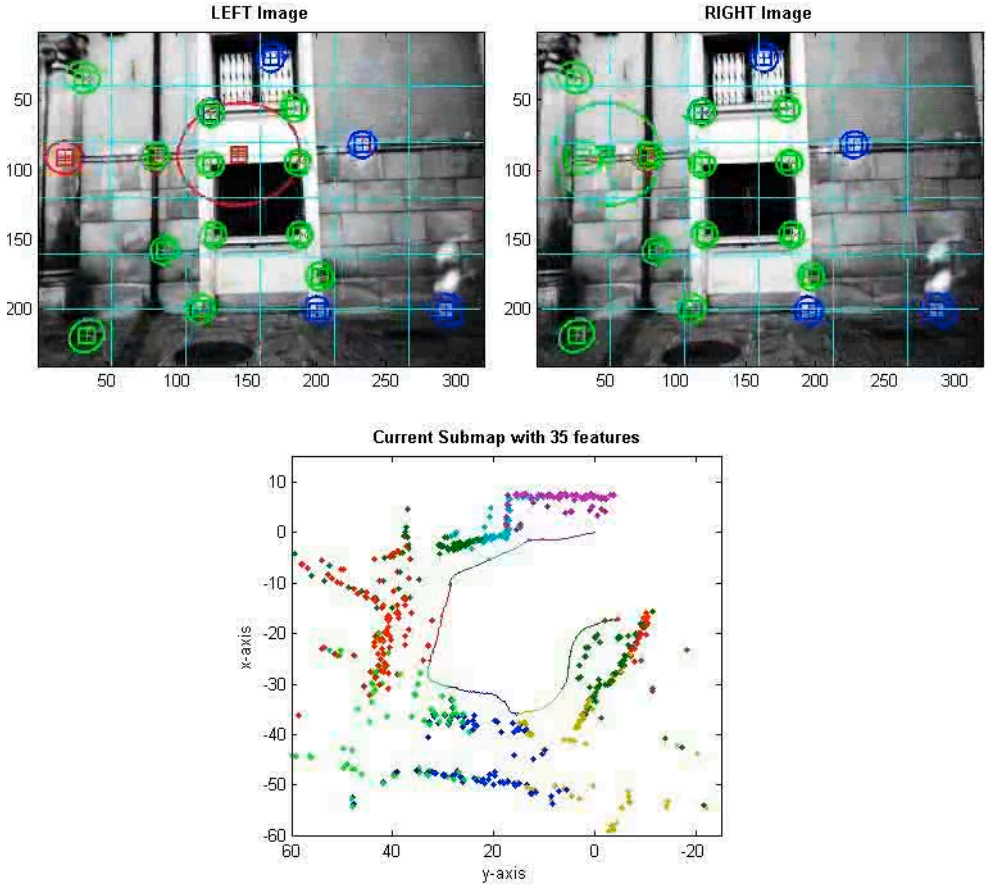


L. Paz, J. Neira and J.D. Tardós **Divide and Conquer: EKF SLAM in O(n)**. IEEE Transactions on Robotics, October 2008.

Amortized cost per step



6DOF SLAM with stereo



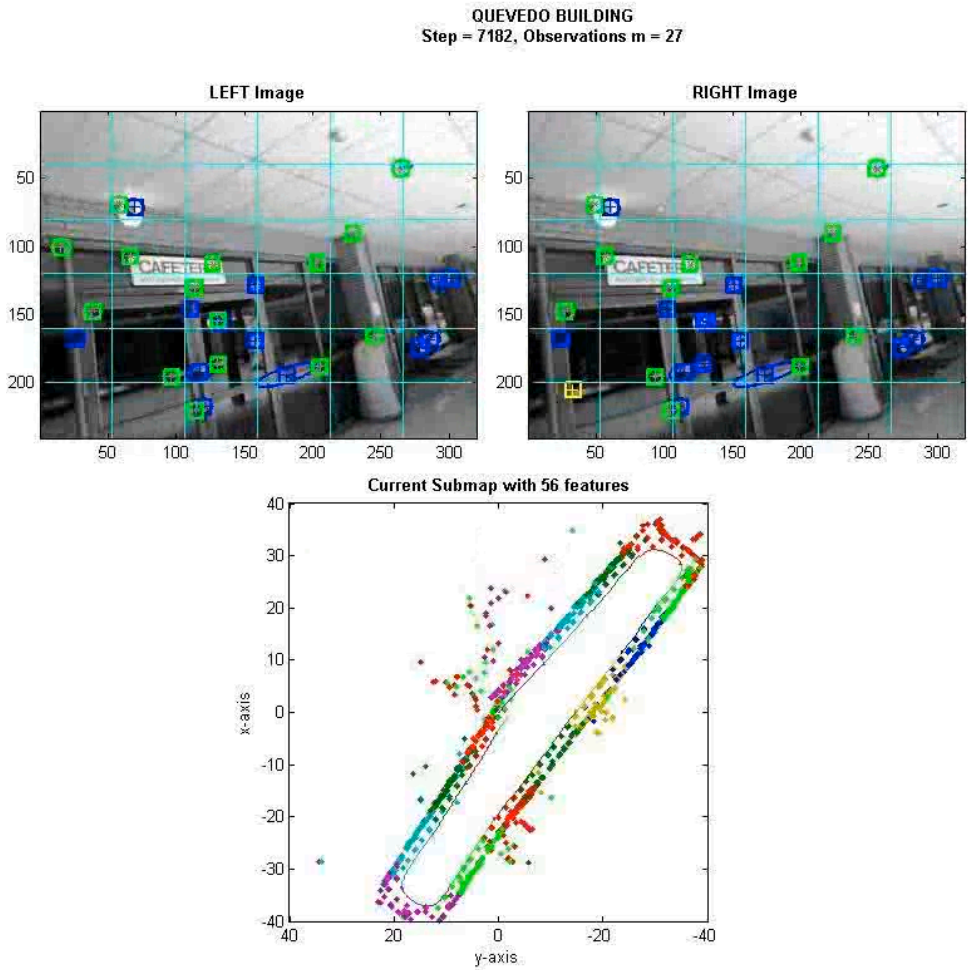
L. Paz, P. Piníes, J. Neira and J.D. Tardós **Large Scale 6DOF SLAM with Stereo-in-Hand.** IEEE Transactions on Robotics, 2008.

6Dof Stereo SLAM, outdoors

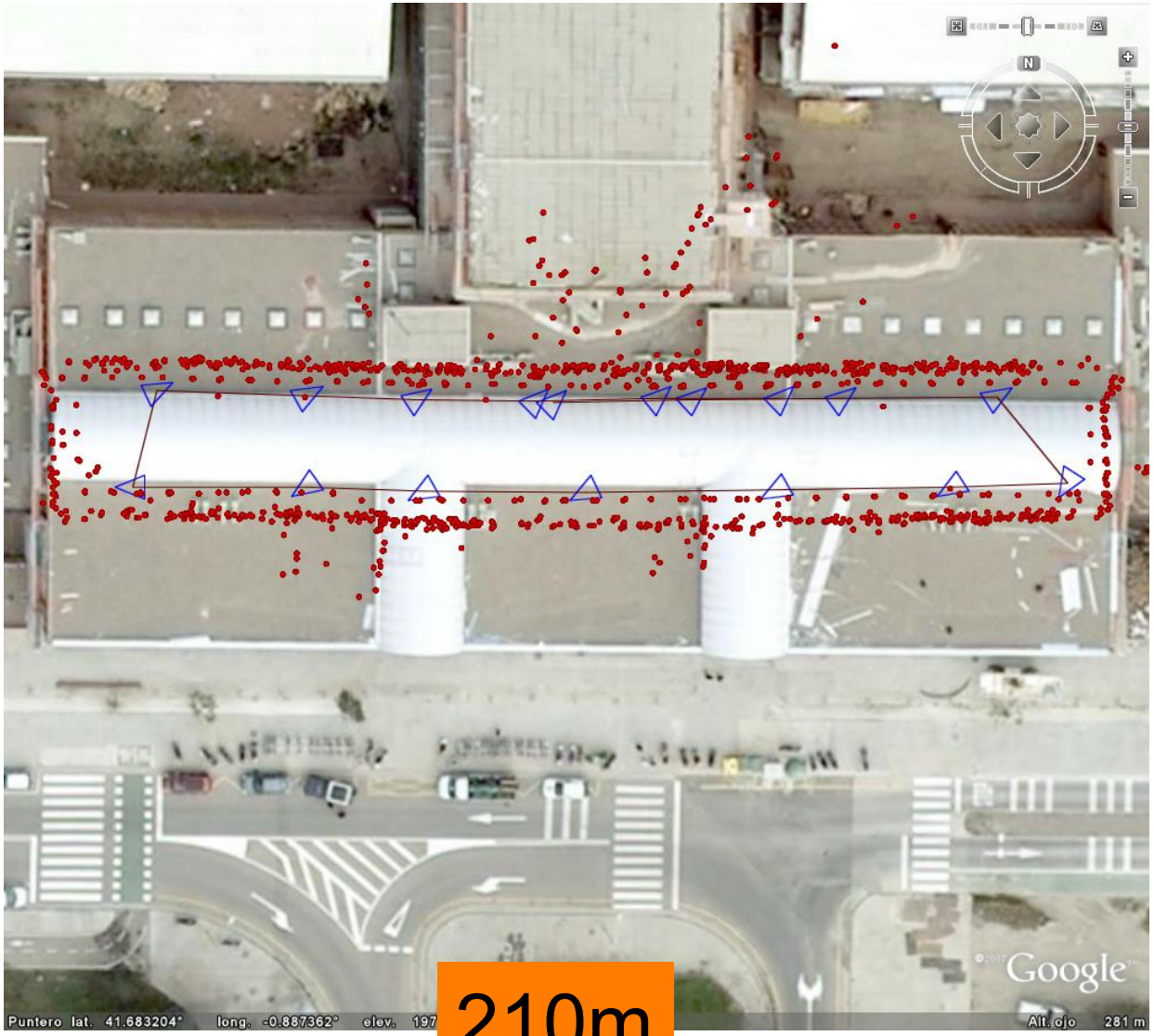
150m

ROBOTICS
UNIVERSIDAD DE ZARAGOZA

6Dof Stereo SLAM, indoors



6Dof Stereo SLAM, indoors



210m

Conclusions

- EKF-SLAM is only consistent for:
  - The linear case (1D robot)
  - Small scale maps (< 100m)

- Inconsistency only becomes evident if:
  - Ground truth is available
  - Trying to close a big loop (> 100m)

- The consistency problem appears before the computational complexity problem !