Distributed Perception and Estimation in Multi-Robot Systems

Principles of Multi-Robot Systems - Workshop at RSS 2015

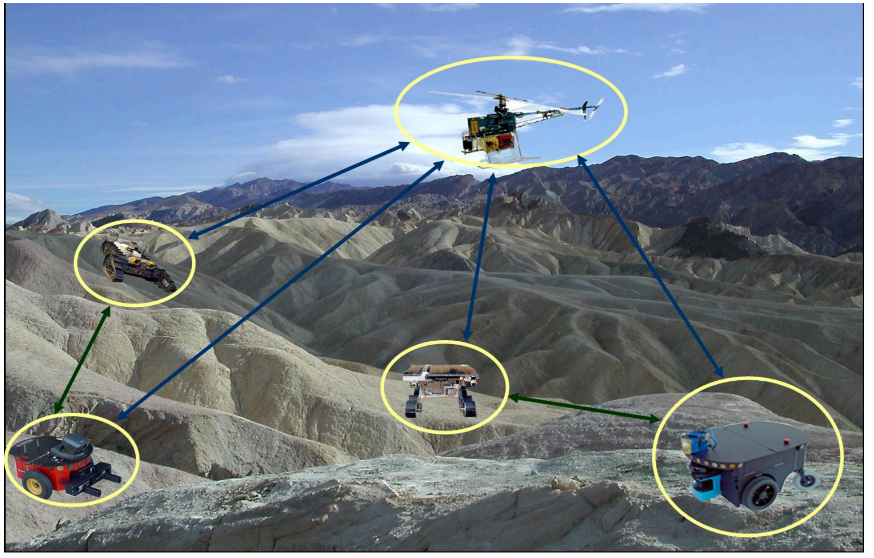
Vadim Indelman

TECHNION
Israel Institute of Technology

TECHNION Israel Institute of Technology

Introduction

- Distributed perception and estimation – central problem in multi-robot systems
- Applications include
  - Localization & navigation
  - Tracking
  - Mapping
  - SLAM
  - ...



- Multi-robot collaboration provides key capabilities but introduces a number of challenges

Image courtesy of Y. Hidaka

This Talk

- Overview of main issues in distributed perception and estimation (**focus** on multi-robot SLAM and cooperative localization, as application)
- Address two key challenges
  - Consistent decentralized estimation
  - Robust decentralized perception
- Naturally, not all aspects are covered
- Only a few approaches/papers are mentioned – apologies!

Outline

- Introduction
- Probabilistic formulation
- Centralized framework
- Distributed framework
- Two particular challenges
  - Consistent distributed estimation
  - Robust distributed perception

Bayesian Inference

- State transition model

$$x_{k+1} = f\left(x_k, u_k\right) + w_k$$

$$p\left(x_{k+1} | x_k, u_k\right)$$

- Observation model

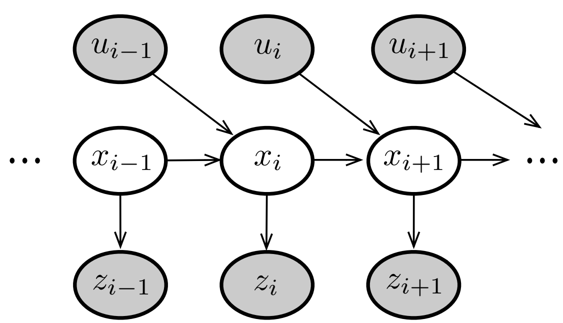
$$z_k = h\left(x_k\right) + v_k$$

$$p\left(z_k | x_k\right)$$

- A posteriori joint pdf:

$$p\left(x_{0:k} | u_{0:k-1}, z_{1:k}\right) = \eta p\left(x_0\right) \prod_{i=1}^{k} p\left(x_i | x_{i-1}, u_{i-1}\right) p\left(z_i | x_i\right)$$

- A posteriori pdf (marginalizing out past states): $p\left(x_k | u_{0:k-1}, z_{1:k}\right)$

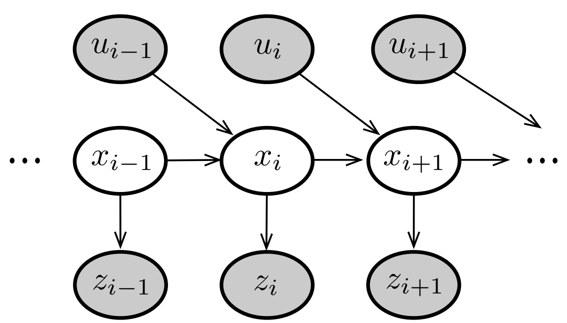


Bayesian Inference

- Objective - Maximum a posteriori (MAP) estimation:

$$x_k^\star = \arg\max_{x_k} p\left(x_k | u_{0:k-1}, z_{1:k}\right)$$

- Common approaches include
  - EKF, EIF
  - UKF
  - Incremental smoothing (iSAM)
  - PF

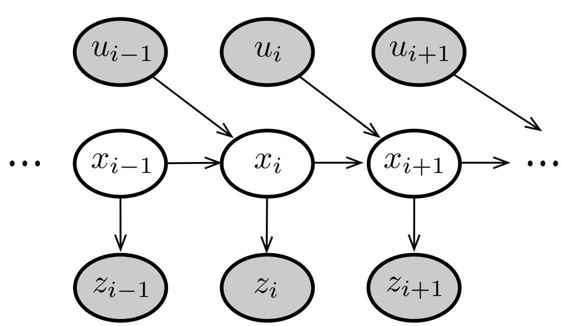


Multi-Robot Perception, Localization and SLAM

Centralized

Inference Over What?

- What are the variables of interest each robot aims to estimate?
- Depends on the problem at hand!
  - May be the same variables for all robots (e.g. tracking)
  - Different variables (e.g. localization)
  - Combination of both (e.g. SLAM)



Collaborative Estimation

- Key capability:
  - By sharing information between robots and formulating multi-robot constraints, performance of individuals in the group can be greatly improved
  - Additional advantages, according to application (e.g. mapping - extend sensing horizon)

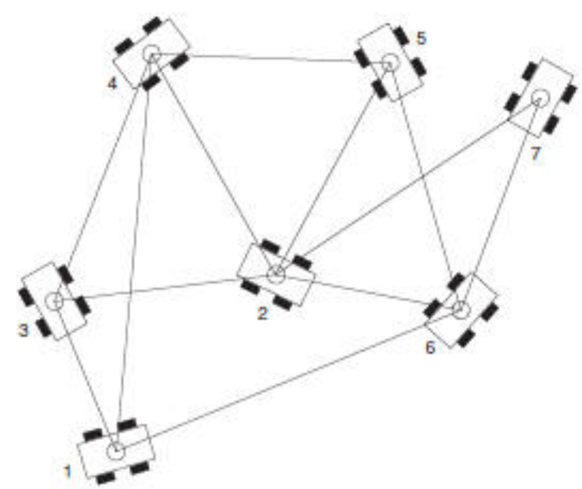
TECHNION Israel Institute of Technology

(Direct) Multi-Robot Observations

- Multi-robot measurement equation (between robots r and r')

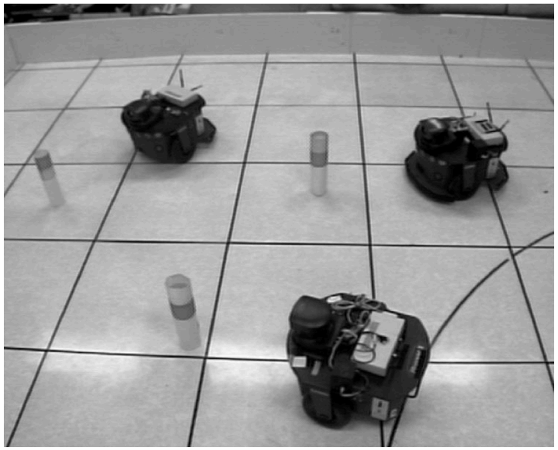
$$z = h\left(x_k^r, x_k^{r'}\right) + v$$

- Common observation types (depends on available sensors)
  - Range
  - Bearing
  - Bearing + range
  - Relative pose
    (relative position, relative orientation)



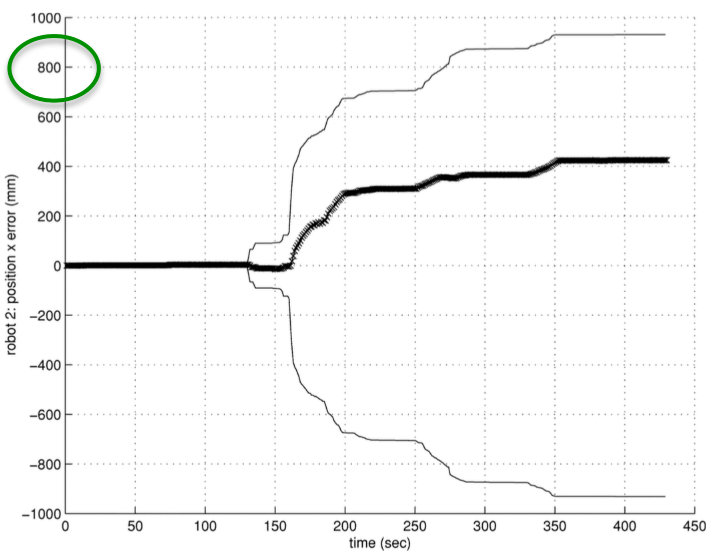
TECHNION
Israel Institute
of Technology

Example

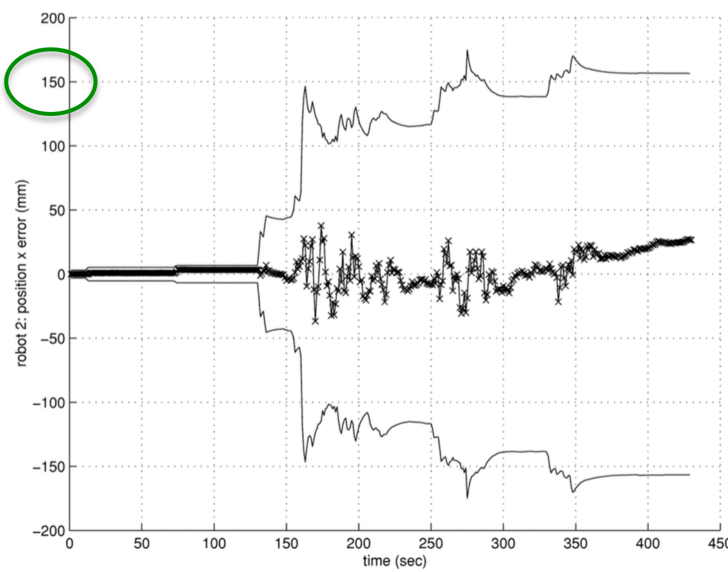


- Experiment setup
  - 3 Pioneer robots
  - Wheel-odometry based dead reckoning
  - Relative pose measurements of each other

Position estimation error
without multi-robot collaboration



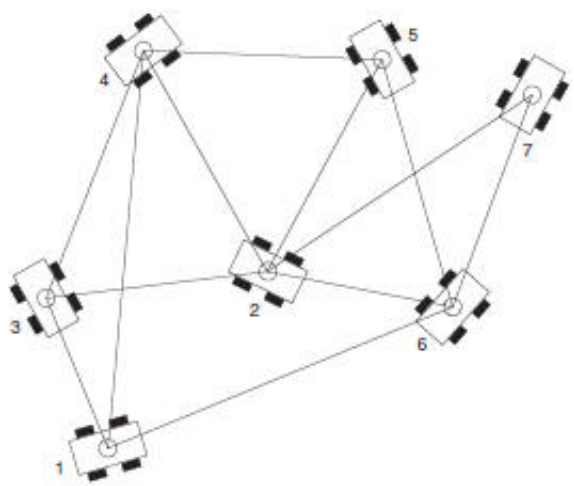
Position estimation error
with multi-robot collaboration

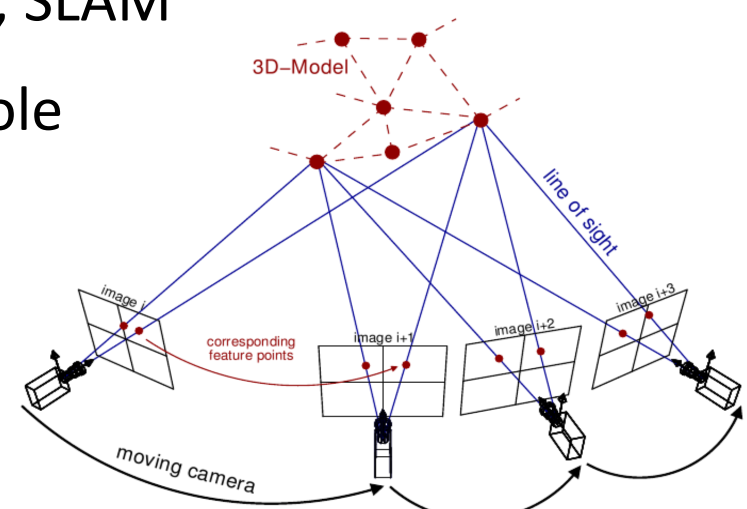


Images from "Distributed multi-robot localization". *IEEE Trans. Robot. Automat.*, 2002.

Multi-Robot Perception and SLAM

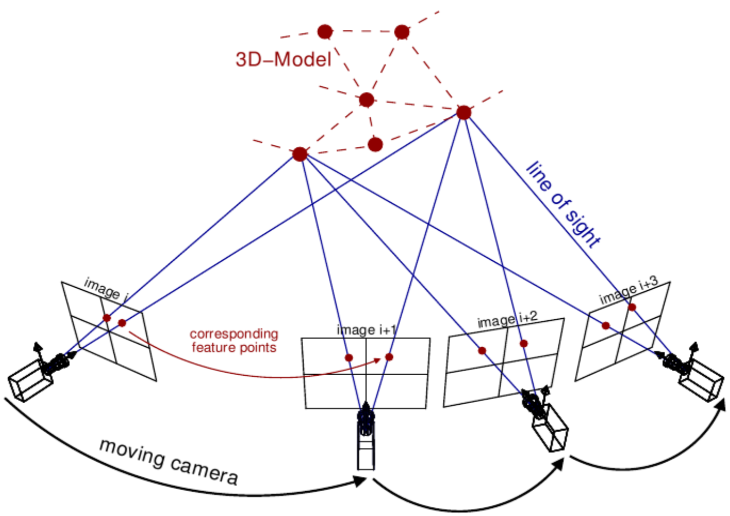
- So far – direct multi-robot observations: robots observe and make measurements **wrt each other**
- Instead, how about **mutually observing the environment**?
  - Environment is known (map is given) – localization problem
  - Environment is unknown – mapping, SLAM
  - In a a key role





Multi-Robot Perception and SLAM

- Robots operate in and make observations of unknown environments
- The corresponding multi-robot constraints describe different robots observing a mutual scene, ***not necessarily*** at the same time
- Measurement equations either involve additional random variables (e.g. landmarks) or robot states from different time instances
- Two common formulations
  - **Pose-SLAM, Collaborative localization**
  - **Full-SLAM, Structure from Motion (SfM)**



Multi-Robot Perception and SLAM

- Multi-robot Pose-SLAM
  - Estimate relative motion from raw observations (match images)
  - Formulate multi-robot constraints, e.g.: $z = h\left(x_k^r, x_j^{r'}\right) + v$
  - Joint pdf: $p\left(X|Z\right) \propto \prod_r \left[p\left(x_0^r\right) \prod_i p\left(x_i^r | x_{i-1}^r, u_{i-1}^r\right)\right] \prod_{(r,r',i,j)} p\left(z_{i,j}^{r,r'} | x_i^r, x_j^{r'}\right)$

Multi-robot constraints

  - Efficient MAP inference (sparsity, re-use calculations)

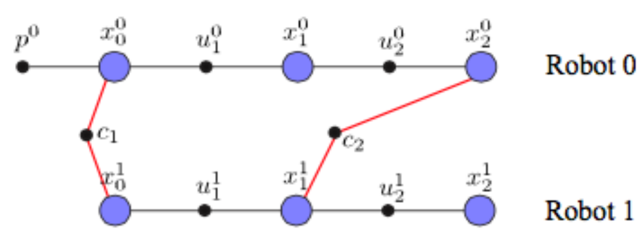


Image from "Multiple Relative Pose Graphs for Robust Cooperative Mapping", *ICRA 2010*

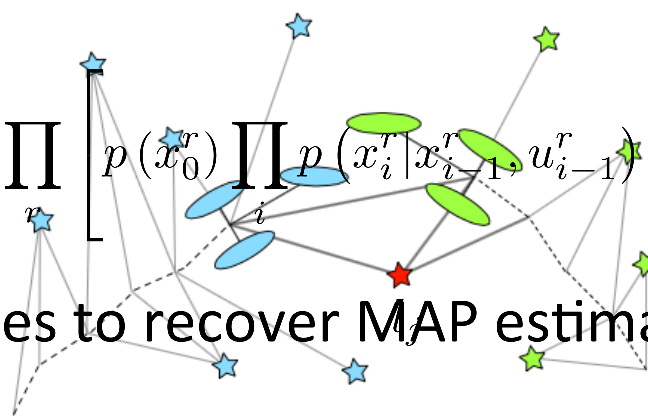
Multi-Robot Perception and SLAM

- Multi-robot Full-SLAM
  - Both robot states and the map are inferred
  - e.g.: robots r and r' observe the same landmark l_j:

$$z^r_{k,j} = h\left(x^r_k, l_j\right) + v \qquad\qquad p\left(z^r_{k,j} | x^r_k, l_j\right) p\left(z^{r'}_{i,j} | x^{r'}_i, l_j\right)$$
$$z^{r'}_{i,j} = h\left(x^{r'}_i, l_j\right) + v$$

  - Joint pdf:

$$p\left(X, L | Z\right) \propto \prod_r \left[p\left(x^r_0\right) \prod_i p\left(x^r_i | x^r_{i-1}, u^r_{i-1}\right) \prod_{j \in \mathcal{M}_i} p\left(z^r_{i,j} | x^r_i, l_j\right) \right]$$



  - Similar approaches to recover MAP estimate

Image from "C-SAM: Multi-Robot SLAM using Square Root Information Smoothing", *ICRA 2008*

Multi-Robot Perception and SLAM

- Notes:
  - All methods require multi-robot data association
  - Common reference frame
  - Thus far – centralized framework

Multi-Robot Perception, Localization and SLAM

Distributed

- Decentralized EKF, Decentralized EIF
- DDF
- Consensus

Cooperative Localization - Decentralized EKF

- Simultaneous localization of robots capable of sensing each other [Roumeliotis and Bekey '02]
- A single EKF estimator for the entire group
- Equations can be written in a decentralized form
  - Each robot maintains an augmented covariance matrix
  - Each robot calculates its **own** update

$$P(t_k) = \begin{bmatrix} P_{11} & P_{12} & P_{13} \\ P_{12}^T & P_{22} & P_{23} \\ P_{13}^T & P_{23}^T & P_{33} \end{bmatrix}$$

$$K(t_k) = \begin{pmatrix} K_1 \\ K_2 \\ K_3 \end{pmatrix}$$

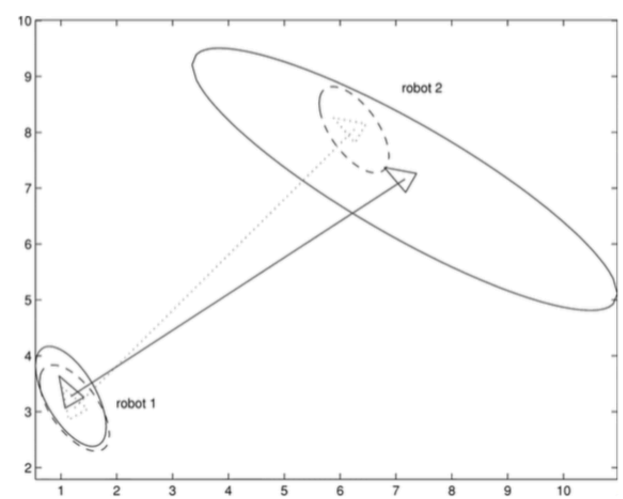


Image from "Distributed multi-robot localization". *IEEE Trans. Robot. Automat.*, 2002.

Decentralized EIF

- Designed for single-beacon cooperative (acoustic) navigation of multiple client underwater vehicles [Webster et al. '13]
- Ranges and state information from a single vehicle (server) are used to improve estimation of other vehicles (clients)
- Calculations in **information form**
- Algorithm yields **identical** results compared to a centralized version

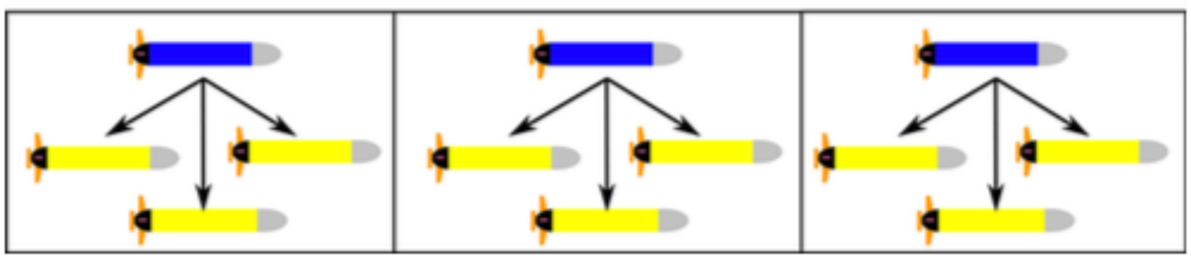


Image from "Decentralized Extended Information Filter for Single-Beacon Cooperative Acoustic Navigation: Theory and Experiments", TRO, 2013

Decentralized Data Fusion (DDF)

- DDF framework [Durrant-Whyte and Stevens '01]
  - Robots infer variables of interest based on local measurements and information communicated by nearby robots
  - No central computational unit
  - Numerous advantages over a centralized framework (scalability, robustness to failure)

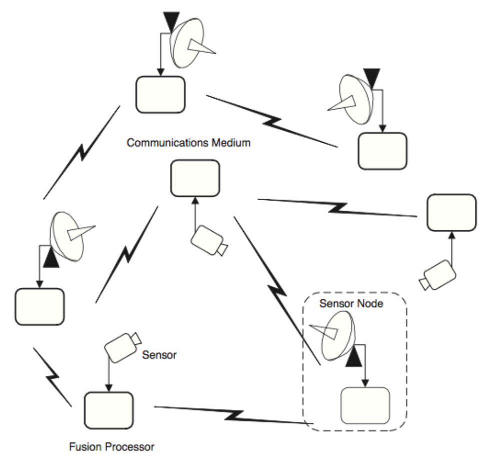


Image from "Data Fusion in Decentralised Sensing Networks ", *Fusion*, 2001

DDF – Calculations in Information Form

- Information vector and matrix $\eta \doteq \Sigma^{-1} x \qquad \Lambda \doteq \Sigma^{-1}$

- For simplicity, consider linear observation model: $z_i = H_i x + v_i \quad , \quad v_i \sim N\left(0, \Sigma_{vi}\right)$

- Prior information $p\left(x\right) = N\left(\hat{x}_0, \Sigma_0\right)$

- Posterior given observations **from other sensors/robots**:

$$p\left(x|Z\right) \propto p\left(x\right) \prod_i p\left(z_i|x\right) \ = N\left(\hat{x}, \Sigma\right)$$

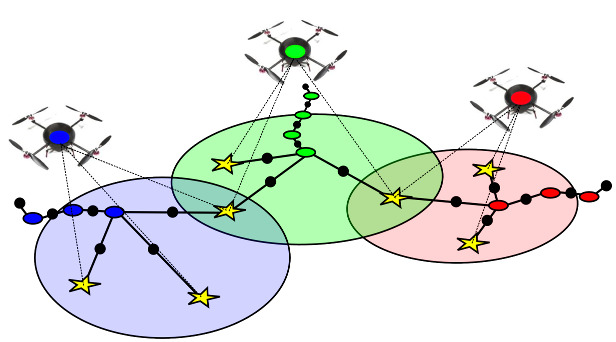
- Inference can be efficiently performed in information form:

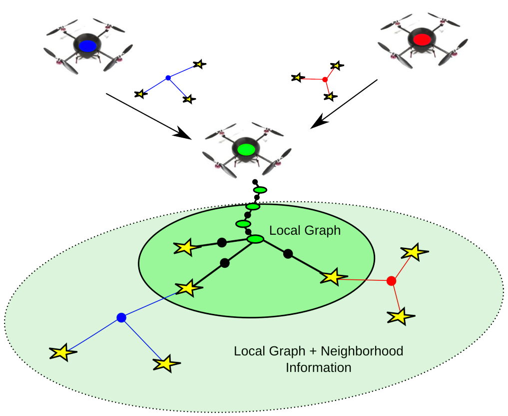
$$\Lambda = \Lambda_0 + \sum_i H_i^T \Sigma_{vi}^{-1} H_i \qquad \eta = \eta_0 + \sum_i H_i^T \Sigma_{vi}^{-1} z_i \qquad \rightarrow \hat{x}, \Sigma$$

- **Avoid double counting information** via information down-dating (more soon)

DDF-SAM

- Extension of DDF to multi-robot smoothing and mapping (SAM) [Cunningham et al. '12, '13]
- Each robot
  - Communicates only with its neighbors
  - Calculates and sends marginal distributions over variable of interest (e.g. landmarks)
  - Consistent estimation by explicitly avoiding double-counting information (discussed next)

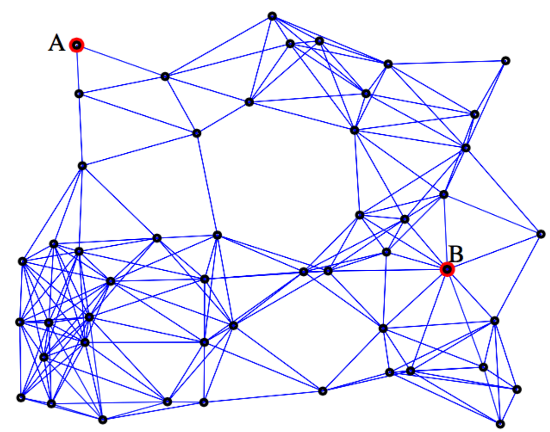




Images from "DDF-SAM 2.0: Consistent Distributed Smoothing and Mapping ", *ICRA*, 2013

Average Consensus Algorithms

- Distributed algorithms to integrate information across network [Olfati-Saber and Murray, IEEE TAC ’04]
- Have been applied to **distributed estimation** [Xiao et al. ’05]
  - Centralized: $\boldsymbol{\theta}_{\mathrm{ML}} = (\Sigma_{i=1}^{N} \boldsymbol{\Lambda}_i^{-1})^{-1} \Sigma_{i=1}^{N} \boldsymbol{\Lambda}_i^{-1} \mathbf{x}_i$
  - Distributed (information form):



Initialization: $\mathbf{P}_i(0) = \boldsymbol{\Lambda}_i^{-1}, \quad \mathbf{q}_i(0) = \boldsymbol{\Lambda}_i^{-1} \mathbf{x}_i$

Each iteration:

$$\mathbf{P}_i(t+1) = \mathbf{P}_i(t) + \sum_{j \in \mathcal{N}_i(t)} a_{ij}(t)(\mathbf{P}_j(t) - \mathbf{P}_i(t)),$$

$$\mathbf{q}_i(t+1) = \mathbf{q}_i(t) + \sum_{j \in \mathcal{N}_i(t)} a_{ij}(t)(\mathbf{q}_j(t) - \mathbf{q}_i(t)),$$

L. Xiao, S. Boyd, and S. Lall. A scheme for robust distributed sensor fusion based on average consensus. In Proceedins of the International Conference on Information Processing in Sensor Networks, 2005.

Average Consensus Algorithms

- Have been recently extended for **distributed map merging** [Aragues et al. '12]
  - Exploit additive operations in information form
  - Robots execute in parallel consensus algorithm on each entry of the information matrix and information vector

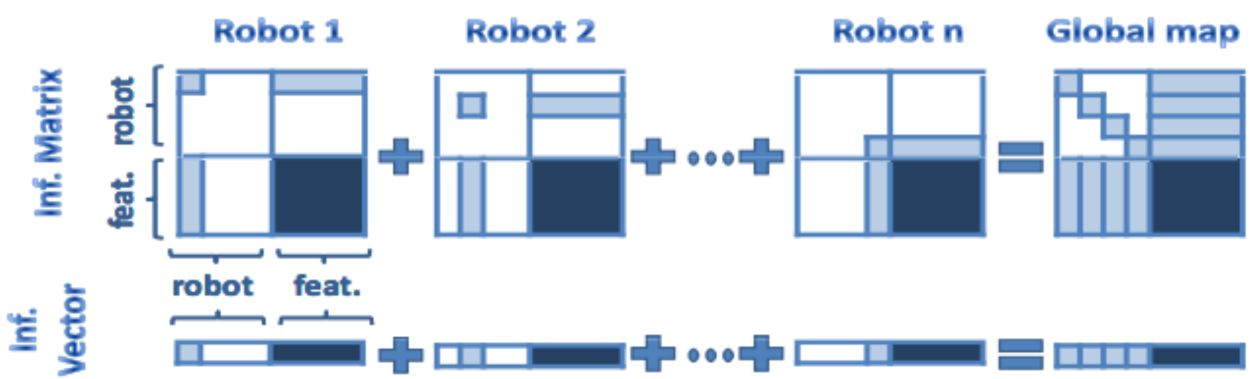


Image from "Distributed Consensus on Robot Networks for Dynamically Merging Feature-Based Maps", TRO 2012

Outline

- Introduction
- Probabilistic formulation
- Centralized
- Decentralized/Distributed

Next

- Two particular challenges
  - Consistent distributed estimation
  - Robust distributed perception (data association)

Consistent Decentralized Estimation

Consistent Decentralized Estimation

- Intuitive example:
  - Consider 3 robots: A,B and C, and a cyclic communication
  - Each robot estimates the variable x based on available data
  - Assume A transmits to B message $p(x|Z_A)$
  - B then passes to C the message $p\left(x|Z_B, Z_A\right)$
  - C sends to A the message $p\left(x|Z_C, Z_B, Z_A\right)$
- **If A treats $p\left(x|Z_C, Z_B, Z_A\right)$ as independent wrt its local belief - it will double count information**

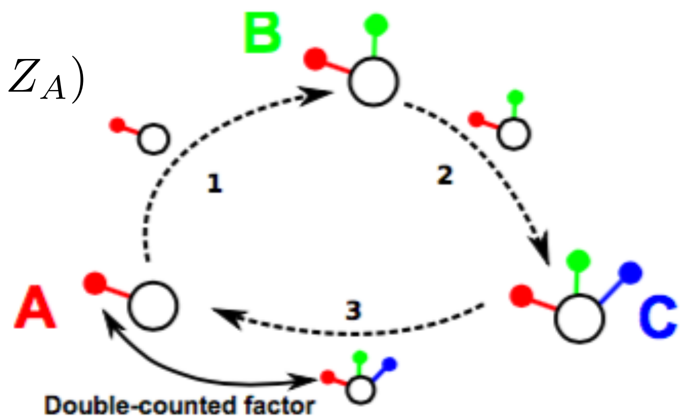


Image courtesy of Alexander Cunningham

Consistent Decentralized Estimation

- Problem becomes more complicated if additional variables are involved, as common in multi-robot perception & SLAM

- Key difficulty:
  - Robots share with each other distributions over landmarks or past poses
  - Need to track common information
  - Typically, the identity of involved variables is **unknown** ahead of time

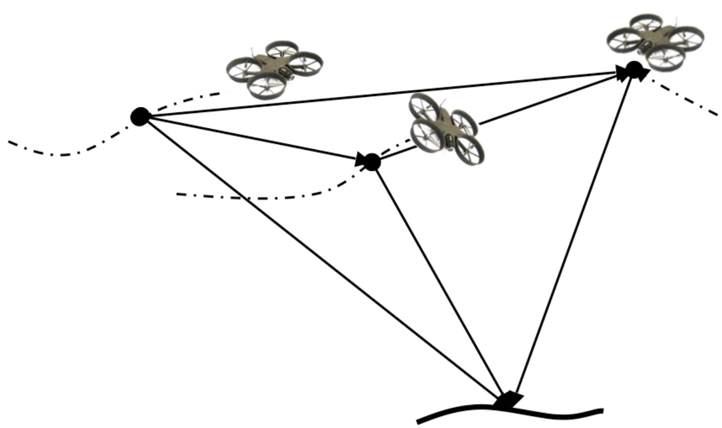


Image from "Distributed vision-aided cooperative localization and navigation based on three-view geometry", *RAS 2012*

TECHNION
Israel Institute
of Technology

Consistent Decentralized Estimation

- Main approaches include:
  - Maintain a bank of filters [Bahr et al. ‘09]
  - Conservative info fusion via covariance intersection [Julier et al. ‘97, Carrillo-Arce et al. ’13]
  - Calculate required correlation terms on demand [Indelman et al. ‘12]
  - Use information down-dating to prevent double counting [Durrant-Whyte et al. ‘01, Cunningham et al. ‘13]

Robust Distributed Perception

Robust Distributed Perception

- Data association problems in **distributed** robot systems
  - **Objective**: determine association between local measurements of the world (e.g. images) and measurements communicated by other robots
  - Extensively investigated by the computer vision community (e.g. RANSAC), typically assuming a centralized framework
  - In the distributed case, **each robot has access to only partial information**

Robust Distributed Perception

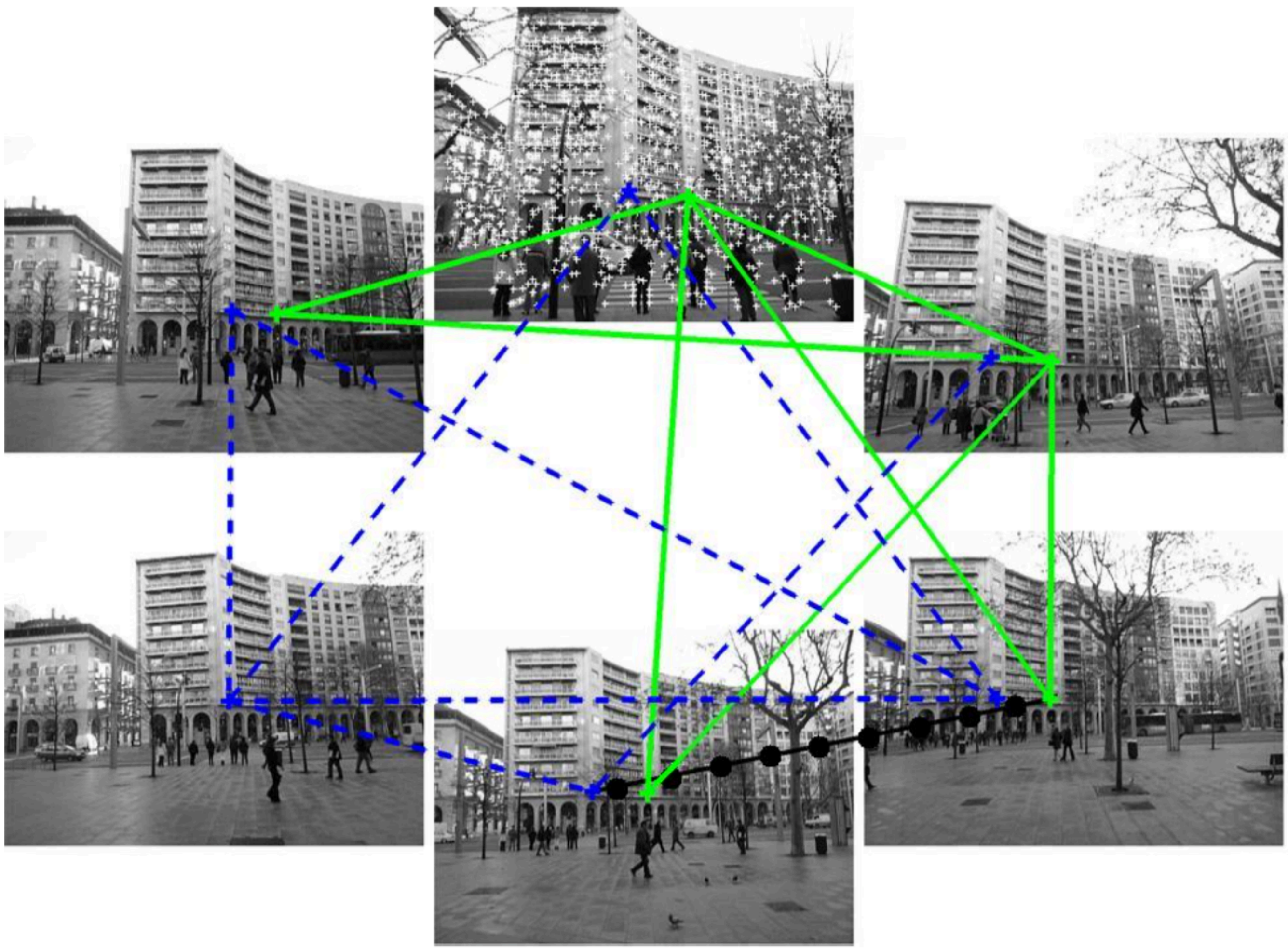


Image from “Consistent data association in multi-robot systems with limited communications“, *RSS 2010*

Robust Distributed Perception

- Main approaches include:
  - Distributed RANSAC with distributed averaging via consensus [Montijano et al. '11, '15]
  - Multi-robot data association within DDF-SAM framework [Cunningham et al. '12]
  - Robust inference - introduce latent variables modeling outlier/inlier correspondences [Latif et al. '12, Sunderhauf and Protzel '12, Lee et al. '13, Indelman et al. '14]

Robust Distributed Perception

- In particular challenging:
  - When information is obtained **incrementally** (as the robots move and explore the environment)
  - In presence of **perceptual aliasing** (e.g. two buildings/ corridors that look alike)
  - Need to decide **when** sufficient information has been accumulated for reliable data association

Summary

- High-level overview of distributed perception and estimation
  - **Centralized** framework
  - **Distributed** framework
  - **Consistent** distributed inference
  - **Robust** distributed perception & inference