Distributed Localization and Tracking of Mobile Networks

Presented by
Florian Meyer

University of California, San Diego
La Jolla, CA, USA

joint work with O. Hlinka, H. Wymeersch,
E. Riegler, and F. Hlawatsch

About this Webinar

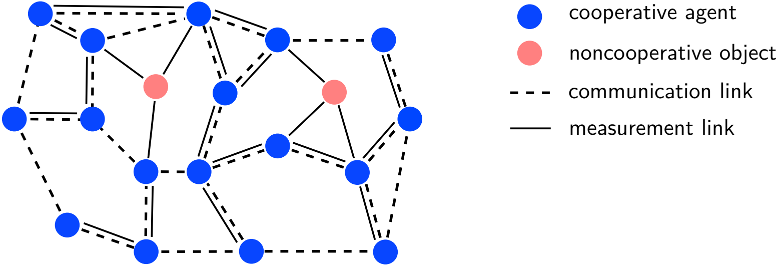
- Location awareness is a cornerstone of future wireless networks and the basis for a wide range of emerging applications
- To enable location awareness in networks, there is a need for distributed, efficient, and scalable estimation algorithms
- This webinar presents a message passing framework for designing distributed Bayesian navigation and tracking algorithms for future wireless networks
- Our focus will be on a set of enabling methodologies including sequential Bayesian estimation, factor graphs and the belief propagation algorithm, particle representations, and consensus

Applications

- Autonomous driving
- Indoor localization
- Maritime situational awareness
- Environmental monitoring
- ...

Simultaneous Object Tracking and Self-Localization

Simultaneous distributed cooperative tracking of moving objects and self-localization of moving agents



General Problem

- We consider distributed (= decentralized), cooperative estimation in a mobile agent network
- Only local computations at the individual agents; no central processing, no fusion center
- Only local communications between neighboring agents; no data flooding, no routing, no long-range communications

Outline

1. Sequential Bayesian Estimation
2. Factor Graphs and the Belief Propagation Algorithm
3. Distributed Sequential Estimation of a Global State
4. Distributed Sequential Estimation of Local States
5. Distributed Sequential Estimation of Local and Global States

Outline

1. Sequential Bayesian Estimation
2. Factor Graphs and the Belief Propagation Algorithm
3. Distributed Sequential Estimation of a Global State
4. Distributed Sequential Estimation of Local States
5. Distributed Sequential Estimation of Local and Global States

The Basis: Bayesian Estimation Framework

- Estimate a parameter/state $\mathbf{x}$ from a measurement/observation $\mathbf{y}$
- Minimum mean-square error (MMSE) estimator:

$$\hat{\mathbf{x}} = \mathrm{E}\{\mathbf{x}|\mathbf{y}\} = \int \mathbf{x}\, f(\mathbf{x}|\mathbf{y})\, \mathrm{d}\mathbf{x}$$

- Posterior probability density function (pdf):

$$f(\mathbf{x}|\mathbf{y}) \propto \underbrace{f(\mathbf{y}|\mathbf{x})}_{\text{Likelihood function}} \overbrace{f(\mathbf{x})}^{\text{Prior pdf}} \qquad \text{(Bayes' theorem)}$$

[**Kay, 93**] S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*, Prentice-Hall, 1993.

Methodologies

State-Space Model

Sequential Bayesian Estimation

Factor Graph

Belief Propagation Algorithm

Message Representation

Consensus

Methodologies

- State-Space Model
- Sequential Bayesian Estimation
- Factor Graph
- Belief Propagation Algorithm
- Message Representation
- Consensus

State-Space Model

- Consider a sequence of states $\mathbf{x}_n$ and a sequence of measurements $\mathbf{y}_n$

State-transition model

State $\mathbf{x}_n$ evolves according to

$$\mathbf{x}_n = \mathbf{g}_n(\mathbf{x}_{n-1}, \underbrace{\mathbf{u}_n}_{\text{Driving noise (white)}}), \quad n = 1, 2, \ldots$$

This determines the state-transition pdf $f(\mathbf{x}_n | \mathbf{x}_{n-1})$

Measurement model

Measurement $\mathbf{y}_n$ depends on state $\mathbf{x}_n$ according to

$$\mathbf{y}_n = \mathbf{h}_n(\mathbf{x}_n, \underbrace{\mathbf{v}_n}_{\text{Measurement noise (white)}}), \quad n = 1, 2, \ldots$$

This determines the likelihood function $f(\mathbf{y}_n | \mathbf{x}_n)$

Markovian Properties

- Noise sequences $\mathbf{u}_n$ and $\mathbf{v}_n$ are assumed mutually independent and independent of $\mathbf{x}_0$.

- Recall:

$$\mathbf{x}_n = \mathbf{g}_n(\mathbf{x}_{n-1}, \mathbf{u}_n)\,, \quad \mathbf{u}_n \text{ is white}$$
$$\mathbf{y}_n = \mathbf{h}_n(\mathbf{x}_n, \mathbf{v}_n)\,, \quad \mathbf{v}_n \text{ is white}$$

- At time n, the state $\mathbf{x}_n$ summarizes all relevant information about the present and past

- Mathematically expressed by "Markovian properties":

$$f(\mathbf{y}_n|\mathbf{x}_n, \mathbf{y}_{1:n-1}) = f(\mathbf{y}_n|\mathbf{x}_n)$$
$$f(\mathbf{x}_{n+1}|\mathbf{x}_n, \mathbf{y}_{1:n}) = f(\mathbf{x}_{n+1}|\mathbf{x}_n)$$

where $\mathbf{y}_{1:n} \triangleq \begin{pmatrix} \mathbf{y}_1 \\ \vdots \\ \mathbf{y}_n \end{pmatrix}$

Methodologies

State-Space Model

Sequential Bayesian Estimation

Factor Graph

Belief Propagation Algorithm

Message Representation

Consensus

Sequential Bayesian Estimation

- We wish to estimate the current state $\mathbf{x}_n$ from the past and current measurements $\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_n$, i.e., from $\mathbf{y}_{1:n}$, for $n = 1, 2, \ldots$
- MMSE estimator:

$$\hat{\mathbf{x}}_n \;=\; \mathrm{E}\{\mathbf{x}_n | \mathbf{y}_{1:n}\} = \int \mathbf{x}_n \, f(\mathbf{x}_n | \mathbf{y}_{1:n}) \, \mathrm{d}\mathbf{x}_n$$

- The posterior pdf $f(\mathbf{x}_n | \mathbf{y}_{1:n})$ can be calculated recursively/sequentially

[**Anderson & Moore, 79**] J. Anderson and B. Moore, *Optimal Filtering*, Prentice-Hall, 1979.

Sequential Bayesian Estimation

- The Markovian properties enable sequential calculation of $f(\mathbf{x}_n|\mathbf{y}_{1:n})$
- One recursion consists of two steps:

Prediction step

$$\underbrace{f(\mathbf{x}_n|\mathbf{y}_{1:n-1})}_{\text{Predicted posterior pdf}} = \int \underbrace{f(\mathbf{x}_n|\mathbf{x}_{n-1})}_{\text{State-transition pdf}} \underbrace{f(\mathbf{x}_{n-1}|\mathbf{y}_{1:n-1})}_{\text{Previous posterior pdf}} \mathrm{d}\mathbf{x}_{n-1}$$

Measurement update step

$$\underbrace{f(\mathbf{x}_n|\mathbf{y}_{1:n})}_{\text{Posterior pdf}} \propto \underbrace{f(\mathbf{y}_n|\mathbf{x}_n)}_{\text{Likelihood function}} \underbrace{f(\mathbf{x}_n|\mathbf{y}_{1:n-1})}_{\text{Predicted posterior pdf}}$$

- Unfortunately, computationally infeasible in general $\Rightarrow$ feasible approximations required
- We will discuss feasible approximations later (in a generalized setting)

Sequential Bayesian Estimation

- Consider joint posterior pdf $f(\mathbf{x}_{0:n}|\mathbf{y}_{1:n})$
- Sequential calculation of the "marginal" posterior pdf $f(\mathbf{x}_n|\mathbf{y}_{1:n})$ can be interpreted as a **factorization and marginalization** of the joint posterior pdf $f(\mathbf{x}_{0:n}|\mathbf{y}_{1:n})$

Factorization

$$f(\mathbf{x}_{0:n}|\mathbf{y}_{1:n}) \propto f(\mathbf{x}_0) \prod_{n'=1}^{n} f(\mathbf{y}_{n'}|\mathbf{x}_{n'})\, f(\mathbf{x}_{n'}|\mathbf{x}_{n'-1})$$

Marginalization

$$f(\mathbf{x}_n|\mathbf{y}_{1:n}) \propto \int f(\mathbf{x}_0) \left(\prod_{n'=1}^{n} f(\mathbf{y}_{n'}|\mathbf{x}_{n'})\, f(\mathbf{x}_{n'}|\mathbf{x}_{n'-1}) \right) \mathrm{d}\mathbf{x}_{0:n-1}$$

Outline

1. Sequential Bayesian Estimation
2. Factor Graphs and the Belief Propagation Algorithm
3. Distributed Sequential Estimation of a Global State
4. Distributed Sequential Estimation of Local States
5. Distributed Sequential Estimation of Local and Global States

Methodologies

State-Space Model

Sequential Bayesian Estimation

Factor Graph

Belief Propagation Algorithm

Message Representation

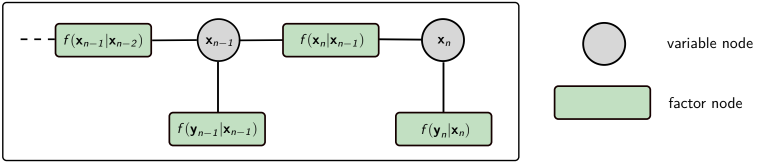
Consensus

Factor Graph

- Recall factorization:

$$f(\mathbf{x}_{0:n}|\mathbf{y}_{1:n}) \propto f(\mathbf{x}_0) \prod_{n'=1}^{n} f(\mathbf{y}_{n'}|\mathbf{x}_{n'})\, f(\mathbf{x}_{n'}|\mathbf{x}_{n'-1})$$

- Representation by **factor graph**:



[**Kschischang et al., 01**] F. R. Kschischang, B. J. Frey, and H.-A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Trans. Inf. Theory*, 2001.

Message Passing

Prediction step → message filtering

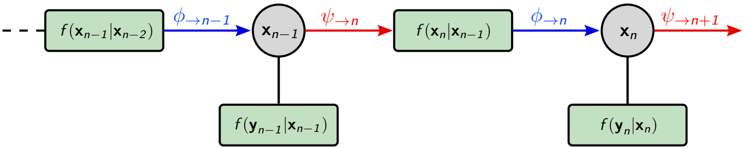
$$f(\mathbf{x}_n|\mathbf{y}_{1:n-1}) = \int f(\mathbf{x}_n|\mathbf{x}_{n-1})\, f(\mathbf{x}_{n-1}|\mathbf{y}_{1:n-1})\, \mathrm{d}\mathbf{x}_{n-1}$$

$$\phi_{\to n}(\mathbf{x}_n) = \int f(\mathbf{x}_n|\mathbf{x}_{n-1})\, \psi_{\to n}(\mathbf{x}_{n-1})\, \mathrm{d}\mathbf{x}_{n-1}$$

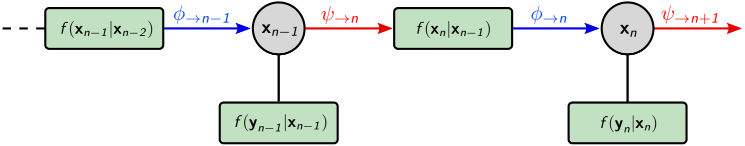
Measurement update step → message multiplication

$$f(\mathbf{x}_n|\mathbf{y}_{1:n}) \propto f(\mathbf{y}_n|\mathbf{x}_n)\, f(\mathbf{x}_n|\mathbf{y}_{1:n-1})$$

$$\psi_{\to n+1}(\mathbf{x}_n) = f(\mathbf{y}_n|\mathbf{x}_n)\, \phi_{\to n}(\mathbf{x}_n)$$



Message Passing



- Sequential calculation of the marginal posterior pdf $f(\mathbf{x}_n|\mathbf{y}_{1:n})$ can be formulated as **message passing on a factor graph**

Generalized Factorization

- Consider state vectors $\mathbf{x}_k$, $k = 1, \ldots, K$, total state vector $\mathbf{x} = (\mathbf{x}_1^{\mathsf{T}} \cdots \mathbf{x}_K^{\mathsf{T}})^{\mathsf{T}}$, and measurement vector $\mathbf{y}$
- General "pairwise" factorization of joint posterior pdf:

$$f(\mathbf{x}|\mathbf{y}) \propto \left(\prod_{l=1}^{K} r(\mathbf{x}_l) \right) \prod_{(k',l') \in \mathcal{E}} r(\mathbf{x}_{k'}, \mathbf{x}_{l'}; \mathbf{y}_{k'l'})$$

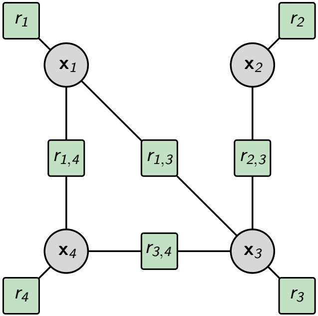
where the $\mathbf{y}_{kl}$ are certain subvectors of $\mathbf{y}$ and $\mathcal{E} \subseteq \{1, \ldots, K\}^2$ ("edge set")

Factor Graph

- Recall factorization:

$$f(\mathbf{x}|\mathbf{y}) \propto \left(\prod_{l=1}^{K} r(\mathbf{x}_l)\right) \prod_{(k',l')\in\mathcal{E}} r(\mathbf{x}_{k'}, \mathbf{x}_{l'}; \mathbf{y}_{k'l'})$$

- Representation by factor graph (example):



- $K = 4$ state vectors $\mathbf{x}_k$
- $\mathcal{E} = \{(1,3),(1,4),(2,3),(3,4)\}$
- $r_l \triangleq r(\mathbf{x}_l)$
- $r_{k,l} \triangleq r(\mathbf{x}_k, \mathbf{x}_l; \mathbf{y}_{kl})$

Marginalization

- MMSE estimator of state $\mathbf{x}_k$: $\hat{\mathbf{x}}_k = \int \mathbf{x}_k f(\mathbf{x}_k|\mathbf{y})\mathrm{d}\mathbf{x}_k$
- The posterior pdf $f(\mathbf{x}_k|\mathbf{y})$ is obtained by marginalization of the joint posterior pdf $f(\mathbf{x}|\mathbf{y})$:

$$f(\mathbf{x}_k|\mathbf{y}) \propto \int f(\mathbf{x}|\mathbf{y})\mathrm{d}\mathbf{x}_{\sim k}$$
$$\propto \int \left(\prod_{l=1}^{K} r(\mathbf{x}_l)\right) \prod_{(k',l')\in\mathcal{E}} r(\mathbf{x}_{k'}, \mathbf{x}_{l'}; \mathbf{y}_{k',l'})\mathrm{d}\mathbf{x}_{\sim k}$$

- “Marginalize a product of functions” (MPF) problem
- The complexity of MPF computations can be reduced dramatically by an appropriate hierarchical organization of the products and integrals $\rightarrow$ **belief propagation algorithm** aka **sum-product algorithm**
- Systematic, “automated” exploitation of statistical independence structure

Methodologies

State-Space Model

Sequential Bayesian Estimation

Factor Graph

Belief Propagation Algorithm

Message Representation

Consensus

Belief Propagation (BP) Algorithm

Message and belief calculation rules

- Message ("extrinsic information") from variable node $\mathbf{x}_l$ to function node $r(\mathbf{x}_k, \mathbf{x}_l; \mathbf{y}_{kl})$:

$$\psi_{l\to k}(\mathbf{x}_l) = r(\mathbf{x}_l) \prod_{k' \in \mathcal{N}_l \setminus \{k\}} \phi_{k'\to l}(\mathbf{x}_l)$$

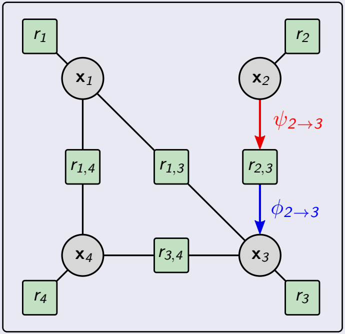
where $\mathcal{N}_l \triangleq \{k \in 1, \ldots, K \,:\, (l,k) \in \mathcal{E}\}$

- Message from function node $r(\mathbf{x}_k, \mathbf{x}_l; \mathbf{y}_{kl})$ to variable node $\mathbf{x}_k$:

$$\phi_{l\to k}(\mathbf{x}_k) = \int r(\mathbf{x}_k, \mathbf{x}_l; \mathbf{y}_{kl})\, \psi_{l\to k}(\mathbf{x}_l)\, \mathrm{d}\mathbf{x}_l$$

- Belief of variable $\mathbf{x}_k$:

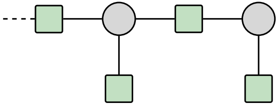
$$b(\mathbf{x}_k) \propto r(\mathbf{x}_k) \prod_{l \in \mathcal{N}_k} \phi_{l\to k}(\mathbf{x}_k)$$



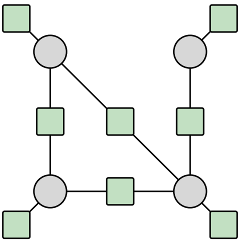
[**Kschischang et al., 01**] F. R. Kschischang, B. J. Frey, and H.-A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Trans. Inf. Theory*, 2001.

Belief Propagation (BP) Algorithm

- The BP algorithm performs an exact marginalization, i.e., $b(\mathbf{x}_k) = f(\mathbf{x}_k|\mathbf{y})$, if the factor graph is a tree



- The BP algorithm performs an approximate marginalization, i.e., $b(\mathbf{x}_k) \approx f(\mathbf{x}_k|\mathbf{y})$, if the factor graph has loops (cycles)
- In the loopy case, the BP algorithm is executed iteratively



Methodologies

State-Space Model

Sequential Bayesian Estimation

Factor Graph

Belief Propagation Algorithm

Message Representation

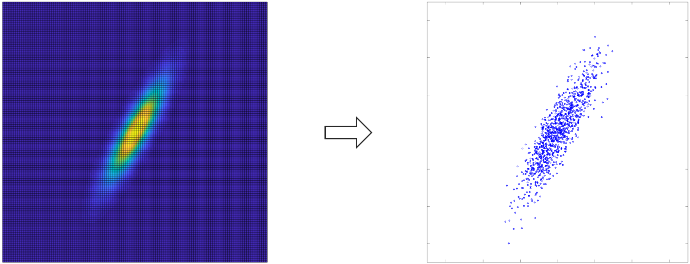
Consensus

Message Representation

- Direct implementation of the BP algorithm (message and belief calculation rules) is still computationally infeasible
- Two alternative feasible approximations:
  - using a parametric representation for the messages and beliefs
    $\Rightarrow$ Gaussian BP, Kalman filtering, ...
  - using a particle representation for the messages and beliefs
    $\Rightarrow$ nonparametric BP, particle filtering, ...
- Here, we use a particle representation

Particle Representation / Nonparametric BP

- Each message or belief is represented by a large number of particles and weights: $f(\mathbf{x}) \sim \left\{ \left(\mathbf{x}^{(j)}, w^{(j)} \right) \right\}_{j=1}^{J}$

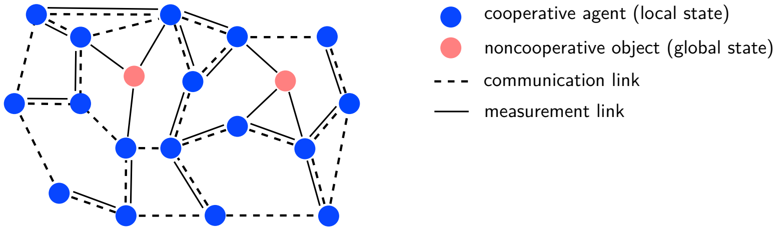


- Nonparametric BP uses a particle representation and is suited to arbitrary nonlinear, non-Gaussian systems

[**Ihler et al., 05**] A. T. Ihler, J. W. Fisher, R. L. Moses, and A. S. Willsky, "Nonparametric belief propagation for self-localization of sensor networks," *IEEE J. Sel. Areas Commun.*, 2005.

Going Distributed

- Let us next consider a distributed implementation
- Recall that our problem is the simultaneous distributed, cooperative tracking of moving objects and self-localization of moving agents



- More generally, we will consider the distributed sequential estimation of both global states and local states

Outline

1. Sequential Bayesian Estimation
2. Factor Graphs and the Belief Propagation Algorithm
3. Distributed Sequential Estimation of a Global State
4. Distributed Sequential Estimation of Local States
5. Distributed Sequential Estimation of Local and Global States

Distributed Sequential Estimation of a Global State

- Time-dependent state vector $\mathbf{x}_n$, $n = 1, 2, \ldots$ (e.g., location of a noncooperative object)
- Agent network consisting of K agents $k = 1, \ldots, K$
- Each agent k acquires a time-dependent measurement vector $\mathbf{y}_{n,k}$
- Each agent k aims to estimate $\mathbf{x}_n$ from $\mathbf{y}_{1:n}$ (i.e., from all the measurements $\mathbf{y}_{n',k'}$ for $k' = 1, \ldots, K$ and $n' = 1, \ldots, n$)
- Fully distributed: no fusion center, only local communications

Distributed Sequential Estimation of a Global State

State-transition model

$$\mathbf{x}_n = \mathbf{g}_n(\mathbf{x}_{n-1}, \underbrace{\mathbf{u}_n}_{\text{Driving noise (white)}}), \quad n = 1, 2, \ldots$$

This determines the state-transition pdf $f(\mathbf{x}_n|\mathbf{x}_{n-1})$

Measurement model at agent k

$$\mathbf{y}_{n,k} = \mathbf{h}_{n,k}(\mathbf{x}_n, \underbrace{\mathbf{v}_{n,k}}_{\text{Measurement noise (white, independent across } k)}), \quad n = 1, 2, \ldots, \quad k = 1, \ldots, K$$

This determines the local likelihood function $f(\mathbf{y}_{n,k}|\mathbf{x}_n)$

- MMSE estimator: $\hat{\mathbf{x}}_n = \mathrm{E}\{\mathbf{x}_n|\mathbf{y}_{1:n}\} = \int \mathbf{x}_n \, f(\mathbf{x}_n|\mathbf{y}_{1:n}) \, \mathrm{d}\mathbf{x}_n$
- We need a distributed algorithm for recursive/sequential calculation of the posterior pdf $f(\mathbf{x}_n|\mathbf{y}_{1:n})$

Distributed Sequential Estimation of a Global State

- In a distributed setting, the measurements $\mathbf{y}_{n,k}$ are dispersed among the agents k
- Disseminating the locally available information through the network is an essential part of distributed estimation algorithms

- Design issues:
  - What kind of local processing is performed?
  - What quantities are communicated?
  - How is the communication organized?
- Computationally feasible estimation algorithms can be based on parametric (e.g., Gaussian) or particle representations
- Here, we use a particle representation, which leads to a distributed particle filter

Distributed Particle Filter

- Nonparametric (particle-based) implementation of distributed sequential Bayesian estimation

Steps performed at time n by agent k

1. Input: $\mathbf{y}_{n,k}$ and $\left\{\left(\mathbf{x}_{n-1,k}^{(j)}, w_{n-1,k}^{(j)}\right)\right\}_{j=1}^{J} \sim f(\mathbf{x}_{n-1}|\mathbf{y}_{1:n-1})$
2. Prediction: "Predicted" particles $\mathbf{x}_{n,k}^{(j)}$ are drawn from $f\left(\mathbf{x}_n \middle| \mathbf{x}_{n-1,k}^{(j)}\right)$
3. Update: Weights are calculated as $w_{n,k}^{(j)} \propto w_{n-1,k}^{(j)} \, f\left(\mathbf{y}_n \middle| \mathbf{x}_{n,k}^{(j)}\right)$
4. Estimation: $\hat{\mathbf{x}}_n = \sum_{j=1}^{J} w_{n,k}^{(j)} \mathbf{x}_{n,k}^{(j)}$ (approximates the MMSE estimator)
5. Resampling (optional)

- Problem: The global likelihood function $f(\mathbf{y}_n|\mathbf{x}_n)$ is not available at the agents

Methodologies

State-Space Model

Sequential Bayesian Estimation

Factor Graph

Belief Propagation Algorithm

Message Representation

Consensus

Consensus

- The global likelihood function factorizes (because the measurement noises $\mathbf{v}_{n,k}$ are independent across k):

$$f(\mathbf{y}_n|\mathbf{x}_n) = \prod_{k=1}^{K} f(\mathbf{y}_{n,k}|\mathbf{x}_n)$$

- Equivalently,

$$f(\mathbf{y}_n|\mathbf{x}_n) = \exp\left(\sum_{k=1}^{K} \log f(\mathbf{y}_{n,k}|\mathbf{x}_n)\right)$$

- In the update step of the particle filter with J particles, the global likelihood function has to be evaluated J times, i.e.,

$$w_{n,k}^{(j)} \propto w_{n-1,k}^{(j)} \exp\left(\sum_{k=1}^{K} \log f\left(\mathbf{y}_{n,k}|\mathbf{x}_n^{(j)}\right)\right), \quad j = 1, \ldots, J$$

Consensus

- Recall update step of the particle filter:

$$w_{n,k}^{(j)} \propto w_{n-1,k}^{(j)} \exp\left(\sum_{k=1}^{K} \log f\left(\mathbf{y}_{n,k} \middle| \mathbf{x}_n^{(j)}\right) \right) = w_{n-1,k}^{(j)} \exp\left(a_{n,j}(\mathbf{y}_n)\right)$$

  with

$$a_{n,j}(\mathbf{y}_n) = \sum_{k=1}^{K} \log f\left(\mathbf{y}_{n,k} \middle| \mathbf{x}_n^{(j)}\right)$$

- The local contributions $\log f\left(\mathbf{y}_{n,k} \middle| \mathbf{x}_n^{(j)}\right)$, $j = 1, \ldots, J$ are calculated locally at each agent k
- The sum of local contributions, $a_{n,j}(\mathbf{y}_n)$, can be calculated in a distributed manner using a consensus algorithm

Consensus

Consensus-based calculation of $a_{n,j}(\mathbf{y}_n)$

1. Initialize the local iterate as $\zeta_k^{(0)} = \log f\left(\mathbf{y}_{n,k} \middle| \mathbf{x}_n^{(j)}\right)$
2. For $i = 1, 2, \ldots, i_{\max}$:
   - Update the local iterate according to
     $$\zeta_k^{(i)} = \omega_{k,k}\,\zeta_k^{(i-1)} + \sum_{k' \in \mathcal{N}_k} \omega_{k,k'}\,\zeta_{k'}^{(i-1)}$$
   - Broadcast $\zeta_k^{(i)}$ to all neighbors $k' \in \mathcal{N}_k$
3. Calculate $\tilde{a}_{n,j}(\mathbf{y}_n) \triangleq K\zeta_k^{(i_{\max})}$

- For $i_{\max} \to \infty$, $\tilde{a}_{n,j}(\mathbf{y}_n)$ is guaranteed to converge to $a_{n,j}(\mathbf{y}_n)$ if the network is connected and the weights $\omega_{k,k'}$ are chosen appropriately
- The number of agents K needs to be known at each agent, and the random number generators used at the agents to draw the predicted particles need to be synchronized

Simulation of Distributed Particle Filter

We consider distributed object tracking:

- One mobile noncooperative object, five static cooperative agents at known locations
- The state of the mobile object consists of location and velocity, i.e., $\mathbf{x}_n = (x_{1,n}\ x_{2,n}\ \dot{x}_{1,n}\ \dot{x}_{2,n})^\mathsf{T}$
- State-transition model for the mobile object:

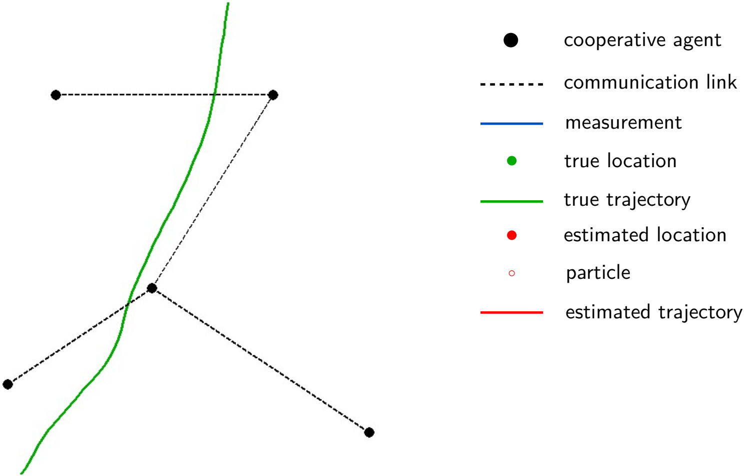
$$\mathbf{x}_n = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \mathbf{x}_{n-1} + \begin{pmatrix} 0.5 & 0 \\ 0 & 0.5 \\ 1 & 0 \\ 0 & 1 \end{pmatrix} \mathbf{u}_n \,, \quad \text{with } \mathbf{u}_n \sim \mathcal{N}(\mathbf{0}, \sigma_u^2 \mathbf{I})$$

- Measurement model:

$$y_{n,k} = \|(x_{1,n}\ x_{2,n})^\mathsf{T} - \mathbf{p}_k\| + v_{n,k}, \quad k = 1,2,3,4$$

  with $v_{n,k} \sim \mathcal{N}(0, \sigma_v^2)$ and known agent location $\mathbf{p}_k$

Simulation of Distributed Particle Filter



Outline

1. Sequential Bayesian Estimation
2. Factor Graphs and the Belief Propagation Algorithm
3. Distributed Sequential Estimation of a Global State
4. Distributed Sequential Estimation of Local States
5. Distributed Sequential Estimation of Local and Global States

Distributed Sequential Estimation of Local States

- With each agent k, there is associated a time-dependent "agent state" vector $\mathbf{x}_{n,k}$, $n = 1, 2, \ldots$ (e.g., time-dependent location of agent)
- Each agent k acquires time-dependent pairwise measurements $\mathbf{y}_{n,kl}$ involving other agents $l \in \mathcal{N}_{n,k}$
- Each agent k aims to estimate its state $\mathbf{x}_{n,k}$ from $\mathbf{y}_{1:n}$ (i.e., from all the measurements $\mathbf{y}_{n',k'l}$ for $k' = 1, \ldots, K$, $l \in \mathcal{N}_{n',k'}$, $n' = 1, \ldots, n$)
- Fully distributed: no fusion center, only local communications

[**Wymeersch et al., 09**] H. Wymeersch, J. Lien, and M. Z. Win, "Cooperative localization in wireless networks," *Proc. IEEE*, 2009.

[**Win et al., 18**] M. Z. Win, F. Meyer, Z. Liu, W. Dai, S. Bartoletti, and A. Conti, "Efficient multi-sensor localization for the Internet-of-Things," *IEEE Signal Process. Mag.*, 2018.

Distributed Sequential Estimation of Local States

State-transition model for agent k

$$\mathbf{x}_{n,k} = \mathbf{g}_{n,k}(\mathbf{x}_{n-1,k}, \underbrace{\mathbf{u}_{n,k}}_{\text{Driving noise}}), \quad n = 1, 2, \ldots$$

Measurement model at agent k

$$\mathbf{y}_{n,kl} = \mathbf{h}_{n,kl}(\mathbf{x}_{n,k}, \mathbf{x}_{n,l}, \underbrace{\mathbf{v}_{n,kl}}_{\text{Measurement noise}}), \quad l \in \mathcal{N}_{n,k}, \quad n = 1, 2, \ldots$$

- MMSE estimator:

$$\hat{\mathbf{x}}_{n,k} = \mathrm{E}\{\mathbf{x}_{n,k} | \mathbf{y}_{1:n}\} = \int \mathbf{x}_{n,k} \, f(\mathbf{x}_{n,k} | \mathbf{y}_{1:n}) \, \mathrm{d}\mathbf{x}_{n,k}$$

- The posterior pdf $f(\mathbf{x}_{n,k} | \mathbf{y}_{1:n})$ can be calculated at agent k by a distributed implementation of a "dynamic" BP algorithm

Distributed Sequential Estimation of Local States

- Consider the joint state $\mathbf{x}_n \triangleq (\mathbf{x}_{n,1}^{\mathsf{T}} \cdots \mathbf{x}_{n,K}^{\mathsf{T}})^{\mathsf{T}}$
- The posterior pdf $f(\mathbf{x}_{n,k}|\mathbf{y}_{1:n})$ is obtained by marginalizing the joint posterior pdf $f(\mathbf{x}_{0:n}|\mathbf{y}_{1:n})$:

$$f(\mathbf{x}_{n,k}|\mathbf{y}_{1:n}) = \int f(\mathbf{x}_{0:n}|\mathbf{y}_{1:n}) \mathrm{d}\mathbf{x}_{\sim n,k}$$

- Factorization of the joint posterior pdf:

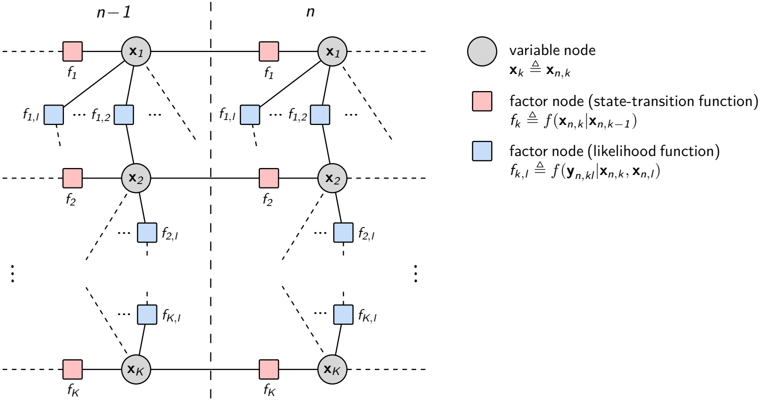
$$f(\mathbf{x}_{0:n}|\mathbf{y}_{1:n}) \propto \left(\prod_{l=1}^{K} f(\mathbf{x}_{0,l}) \right) \prod_{n'=1}^{n} \left(\prod_{k=1}^{K} f(\mathbf{x}_{n',k}|\mathbf{x}_{n'-1,k}) \right. \\ \left. \times \prod_{(k',l') \in \mathcal{E}_{n'}} f(\mathbf{y}_{n',k'l'}|\mathbf{x}_{n',k'}, \mathbf{x}_{n',l'}) \right)$$

Distributed Sequential Estimation of Local States

- Recall factorization:

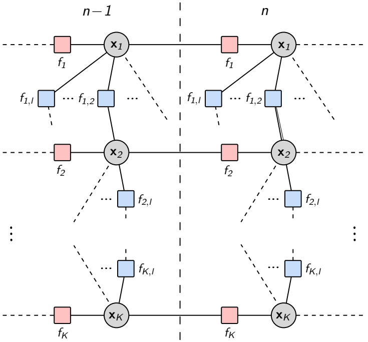
$$f(\mathbf{x}_{0:n}|\mathbf{y}_{1:n}) \propto \left(\prod_{l=1}^{K} f(\mathbf{x}_{0,l})\right) \prod_{n'=1}^{n} \left(\prod_{k=1}^{K} f(\mathbf{x}_{n',k}|\mathbf{x}_{n'-1,k})\right) \prod_{(k',l')\in\mathcal{E}_{n'}} f(\mathbf{y}_{n',k'l'}|\mathbf{x}_{n',k'},\mathbf{x}_{n',l'})$$

- Representation by factor graph:



Distributed Sequential Estimation of Local States

- Factor graph:



- Problem: Factor graph grows with time ⇒ computation and communication requirements per time step increase linearly with time

- Solution: Messages are sent only forward in time and iterative message passing is performed at each time step individually

Distributed Sequential Estimation of Local States

Dynamic BP algorithm:

- "Prediction" message:

$$\phi_{\rightarrow n}(\mathbf{x}_{n,k}) = \int f(\mathbf{x}_{n,k}|\mathbf{x}_{n-1,k})\, b(\mathbf{x}_{n-1,k})\, \mathrm{d}\mathbf{x}_{n-1,k}$$

Since we send messages only forward in time, we directly use the belief $b(\mathbf{x}_{n-1,k})$ instead of some extrinsic information

- "Measurement" message:

$$\phi_{l\rightarrow k}(\mathbf{x}_{n,k}) = \int f(\mathbf{y}_{n,kl}|\mathbf{x}_{n,l}, \mathbf{x}_{n,k})\, \psi_{l\rightarrow k}(\mathbf{x}_{n,l})\, \mathrm{d}\mathbf{x}_{n,l}$$

- Extrinsic information:

$$\psi_{l\rightarrow k}(\mathbf{x}_{n,l}) = \phi_{\rightarrow n}(\mathbf{x}_{n,l}) \prod_{k'\in\mathcal{N}_{n,l}\setminus\{k\}} \phi_{k'\rightarrow l}(\mathbf{x}_{n,l})$$

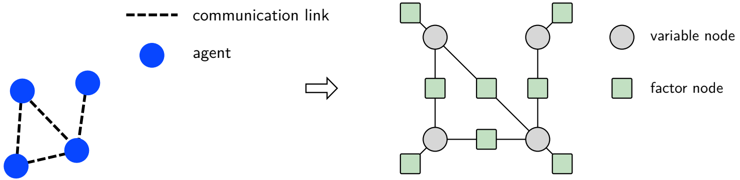
- Belief:

$$b(\mathbf{x}_{n,k}) \propto \phi_{\rightarrow n}(\mathbf{x}_{n,k}) \prod_{l\in\mathcal{N}_{n,k}} \phi_{l\rightarrow k}(\mathbf{x}_{n,k})$$

Distributed Sequential Estimation of Local States

- A **distributed** implementation of the dynamic BP algorithm presupposes that the communication graph of the agent network coincides with the factor graph

  - Agent k in the communication graph corresponds to variable node $\mathbf{x}_{n,k}$ in the factor graph
  - Agent k is able to communicate with all neighboring agents $l \in \mathcal{N}_{n,k}$

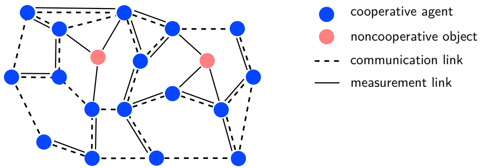


- This correspondence guarantees that all the messages required for calculating the belief $b(\mathbf{x}_{n,k})$ at agent k are within the "communication neighborhood" of agent k

Outline

1. Sequential Bayesian Estimation
2. Factor Graphs and the Belief Propagation Algorithm
3. Distributed Sequential Estimation of a Global State
4. Distributed Sequential Estimation of Local States
5. Distributed Sequential Estimation of Local and Global States

Distributed Sequential Estimation of Local & Global States



- Local states: $\mathbf{x}_{n,k}$, $k = 1, \ldots, K$ (correspond to cooperative agents)
- Global states: $\mathbf{x}_{n,m}$, $m = K+1, \ldots, M$ (correspond to noncooperative objects)
- Measurements: $\mathbf{y}_{n,kl}$, $k = 1, \ldots, K$, $l \in \mathcal{N}_{n,k} \subseteq \{1, \ldots, M\} \backslash \{k\}$ (acquired by cooperative agents)
- Each agent k aims to estimate its own local state $\mathbf{x}_{n,k}$ **and** the global states $\mathbf{x}_{n,m}$, $m = K+1, \ldots, M$ from $\mathbf{y}_{1:n}$ (i.e., from all the measurements $\mathbf{y}_{n',k'l}$ for $k' = 1, \ldots, K$, $l \in \mathcal{N}_{n',k'}$, $n' = 1, \ldots, n$)

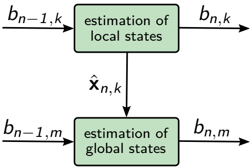
Distributed Sequential Estimation of Local & Global States

- Use a distributed BP algorithm (particle-based implementation)
- Problem: Global states $\mathbf{x}_{n,m}$, $m = K+1, \ldots, M$ correspond to noncooperative objects $\Rightarrow$ some vital information is not communicated to the cooperative agents
- More specifically, for calculating the belief $b(\mathbf{x}_{n,m})$, the product of messages $\prod_{l \in \mathcal{N}_{n,m}} \phi_{l \to m}(\mathbf{x}_{n,m})$ is required — unfortunately, this message product is not available at the cooperative agents
- We solve this problem by calculating the particle weights using consensus, as explained earlier in Section 3

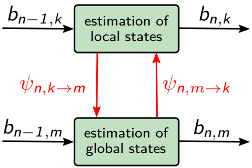
Distributed Sequential Estimation of Local & Global States

Transfer of probabilistic information:

- In separate estimation of local and global states, typically the final local state estimates $\hat{\mathbf{x}}_{n,k}$, $k = 1, \ldots, K$ are used for estimation of the global states



- In joint estimation of local and global states, as considered here, probabilistic information is transferred between local and global state estimation — this improves the performance of both stages



Simulation Setting

We consider joint distributed object tracking and cooperative self-localization:

- Two mobile noncooperative objects, eight mobile cooperative agents, four anchors (static cooperative agents with perfect prior information)
- The states of the mobile objects and agents consist of location and velocity, i.e., $\mathbf{x}_{n,k} = (x_{1,n,k} \; x_{2,n,k} \; \dot{x}_{1,n,k} \; \dot{x}_{2,n,k})^{\mathrm{T}}$
- State-transition model for mobile objects and agents:

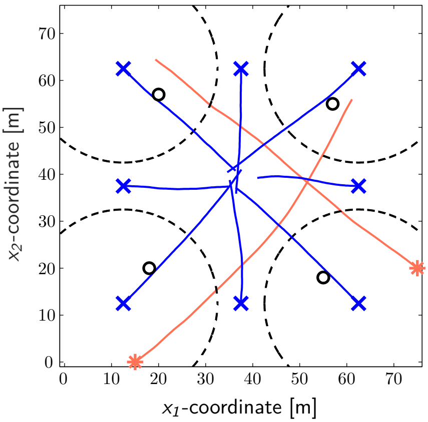
$$\mathbf{x}_{n,k} = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \mathbf{x}_{n-1,k} + \begin{pmatrix} 0.5 & 0 \\ 0 & 0.5 \\ 1 & 0 \\ 0 & 1 \end{pmatrix} \mathbf{u}_{n,k}\,, \;\; \text{with} \;\; \mathbf{u}_{n,k} \sim \mathcal{N}(\mathbf{0}, \sigma_u^2 \mathbf{I})$$

- Measurement model for mobile agents and anchors ($k = 1, \ldots, K$):

$$y_{n,kl} = \|(x_{1,n,k} \; x_{2,n,k}) - (x_{1,n,l} \; x_{2,n,l})\| + v_{n,kl}, \quad l \in \mathcal{N}_{n,k}$$

  with $v_{n,kl} \sim \mathcal{N}(0, \sigma_v^2)$

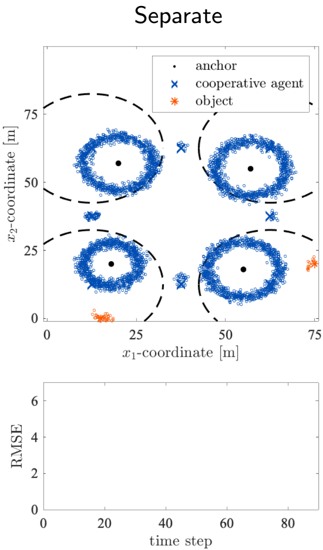
Simulation Scenario



- trajectory of mobile agent (initial location is indicated by ×)
- trajectory of object (initial location is indicated by ∗)
- ∘ location of anchor

Simulation Results

Separate vs. joint cooperative object tracking and self-localization



Conclusion

- General problem: Distributed, cooperative, sequential estimation in a decentralized network
- Both cooperative network nodes ("agents") and noncooperative network nodes ("objects")
- The belief propagation algorithm systematically exploits conditional independencies for reduced complexity and improved scalability
- Only local computations at the individual agents and local communications between neighboring agents are performed
- Considered scenario: joint distributed cooperative object tracking and self-localization
- Improved performance because of bidirectional probabilistic information transfer between object tracking and self-localization stages

Recent Results

- The proposed framework and methodology has been adapted to accommodate additional tasks such as
  - distributed synchronization [**Etzlinger et al., 17**]
  - information-seeking agent control [**Meyer et al., 15**]
- It has also been extended to scenarios involving an unknown and time-varying number of objects as well as object-measurement association uncertainty [**Meyer et al., 17**], [**Meyer & Win, 18**], [**Sharma et al., 19**]

[**Etzlinger et al., 17**] B. Etzlinger, F. Meyer, F. Hlawatsch, A. Springer, and H. Wymeersch, "Cooperative simultaneous localization and synchronization in agent networks," *IEEE Trans. Signal Process.*, Jul. 2017.

[**Meyer et al., 15**] F. Meyer, H. Wymeersch, M. Fröhle, and F. Hlawatsch, "Distributed estimation with information-seeking control in agent networks," *IEEE J. Sel. Areas Commun.*, Nov. 2015.

[**Meyer et al., 17**] F. Meyer, P. Braca, P. Willett, and F. Hlawatsch, "A scalable algorithm for tracking an unknown number of targets using multiple sensors," *IEEE Trans. Signal Process.*, Jul. 2017.

[**Meyer & Win, 18**] F. Meyer and M. Z. Win, "Joint navigation and multitarget tracking in networks," in *Proc. IEEE ICC-18*, May 2018.

[**Sharma et al., 19**] P. Sharma, A.-A. Saucan, D. J. Bucci, and P. K. Varshney, "Decentralized Gaussian filters for cooperative self-localization and multi-target tracking," *IEEE Trans. Signal Process.*, Nov. 2019.

References

[**Anderson & Moore, 79**] J. Anderson and B. Moore, *Optimal Filtering*, Prentice-Hall, 1979.

[**Kay, 93**] S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*, Prentice-Hall, 1993.

[**Kschischang et al., 01**] F. R. Kschischang, B. J. Frey, and H.-A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Trans. Inf. Theory*, 2001.

[**Ihler et al., 05**] A. T. Ihler, J. W. Fisher, R. L. Moses, and A. S. Willsky, "Nonparametric belief propagation for self-localization of sensor networks," *IEEE J. Sel. Areas Commun.*, 2005.

[**Wymeersch et al., 09**] H. Wymeersch, J. Lien, and M. Z. Win, "Cooperative localization in wireless networks," *Proc. IEEE*, 2009.

[**Meyer et al., 16**] F. Meyer, O. Hlinka, H. Wymeersch, E. Riegler, and F. Hlawatsch, "Distributed localization and tracking of mobile networks including noncooperative objects," *IEEE Trans. Signal Inf. Process. Netw.*, 2016.