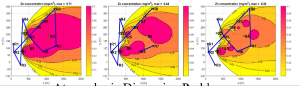
Efficient Distributed State Estimation of Hidden Markov Models over Unreliable Networks

Amirhossein Tamjidi[1], Reza Oftadeh[2],
Suman Chakravorty[1] and, Dylan Shell[2]

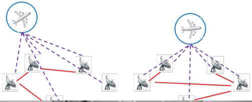
[1]Department of Aerospace Engineering, Texas A&M University
[2]Department of Computer Science & Engineering, Texas A&M University

Overview

Motivation



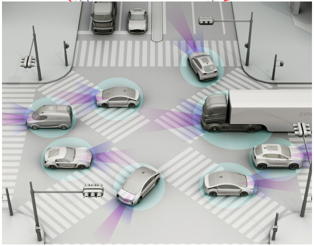
Atmospheric Dispersion Problem



Scalablity

Robustness to network failure

No global knowledge of the communication network



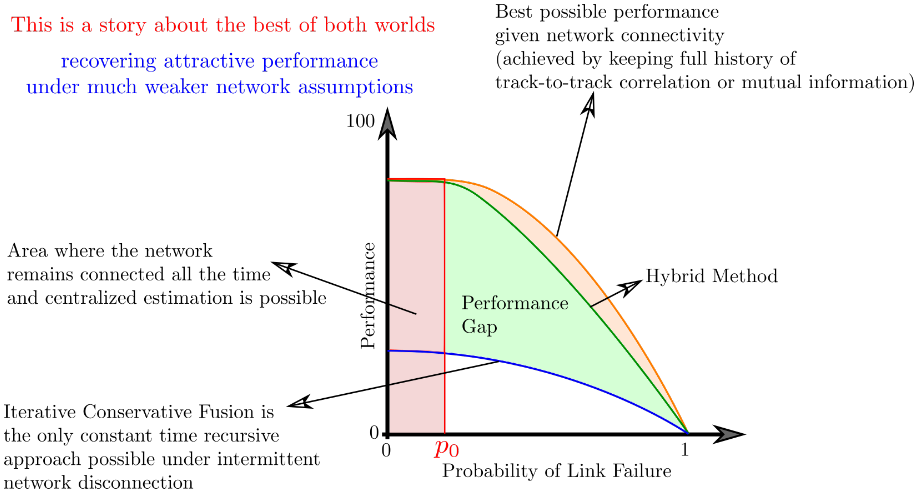
source: https://sites.google.com/a/ncsu.edu/firefighting-drone-challenge/

source: http://statescoop.com/self-driving-in-north-dakota-new-research-to-target-data-privacy-safety

Synopsis

This is a story about the best of both worlds

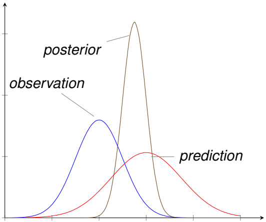
recovering attractive performance
under much weaker network assumptions



Bayesian Estimation

$$\overbrace{p(\mathbf{X}_k|\{\mathbf{z}_k^i\}_{k\in \boldsymbol{I}_k}^{i\in \boldsymbol{I}_n})}^{\textit{posterior}} = \frac{1}{\eta}\overbrace{p(\{\mathbf{z}_k^i\}^{i\in \boldsymbol{I}_n}|\mathbf{X}_k)}^{\textit{observation model}}\overbrace{p(\mathbf{X}_k|\{\mathbf{z}_k^i\}_{k\in \boldsymbol{I}_{k-1}}^{i\in \boldsymbol{I}_n})}^{\textit{prediction}}$$

$$= \frac{1}{\eta}\prod_{i=1}^{n} \overbrace{p(\mathbf{z}_\mathbf{k}^\mathbf{i}|\mathbf{X}_\mathbf{k})}^{\textit{agent obs. mdl.}} \int \overbrace{p(\mathbf{X}_\mathbf{k}|\mathbf{X}_{\mathbf{k-1}})}^{\textit{motion model}}\overbrace{p(\mathbf{X}_{k-1}|\{\mathbf{z}_k^i\}_{k\in \boldsymbol{I}_{k-1}}^{i\in \boldsymbol{I}_n})}^{\textit{prior}} d\mathbf{X}_{k-1}$$



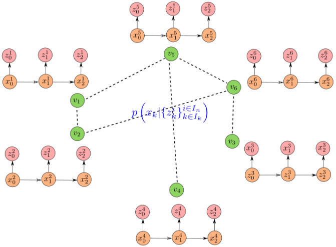
k : time step
$\mathbf{X}_k$: state (ex. position of a robot)
$\mathbf{z}_k^i$: observation by agent i
(ex. distance to wall)
$\boldsymbol{I}_k = \{1, 2, \cdots, k\}$ index set

Estimation on Sensor Networks

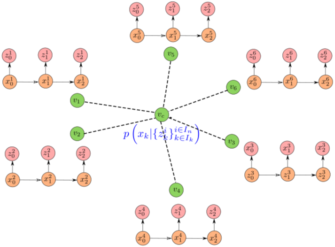
Recursive State Estimation is the process of recursively computing the posterior probability of a random dynamic process $\mathbf{X}_k$ conditioned on a sequence of measurements $\{\mathbf{z}_k^i\}_{k\in \boldsymbol{I_k}}^{i\in \boldsymbol{I_n}}$.

$$\overbrace{\mathrm{p}(\mathbf{X}_k|\{\mathbf{z}_k^i\}_{k\in \boldsymbol{I_k}}^{i\in \boldsymbol{I_n}})}^{\text{posterior}} = \frac{1}{\eta}\overbrace{\mathrm{p}(\{\mathbf{z}_k^i\}^{i\in \boldsymbol{I_n}}|\mathbf{X}_k)}^{\text{observation model}}\overbrace{\mathrm{p}(\mathbf{X}_k|\{\mathbf{z}_k^i\}_{k\in \boldsymbol{I_{k-1}}}^{i\in \boldsymbol{I_n}})}^{\text{prediction}}$$

$$= \frac{1}{\eta}\prod_{i=1}^{n}\overbrace{\mathrm{p}(\mathbf{z_k^i}|\mathbf{X_k})}^{\text{agent obs. mdl.}} \int \overbrace{\mathrm{p}(\mathbf{X_k}|\mathbf{X_{k-1}})}^{\text{motion model}}\overbrace{\mathrm{p}(\mathbf{X}_{k-1}|\{\mathbf{z}_k^i\}_{k\in \boldsymbol{I_{k-1}}}^{i\in \boldsymbol{I_n}})}^{\text{prior}}d\mathbf{X}_{k-1}$$



Distributed



Centralized

Distributed State Estimation (DSE)

Why Distributed State Estimation?

- Difficulty of centralized estimator implementation due to bandwidth and energy constraints
- scalability
- modularity
- robustness to network failure

state	static (Xiao et al., 2005)	dynamic (Simonetto et al., 2010)
transition model	linear (Olfati-Saber, 2005)	non-linear (Battistelli et al., 2014)
topology of the network	always connected (Battistelli et al., 2014)	intermittent disconnection (Tamjidi et al., 2016; Xiao et al., 2005)
knowledge of the network topology	global(Xiao et al., 2005)	local (Tamjidi et al., 2016; Xiao et al., 2005; Battistelli et al., 2014)
treatment of mutual information	channel filter (Durrant-Whyte et al., 2001)	avoid double counting (Wang and Li, 2012; Hu et al., 2012)
noise	gaussian (Olfati-Saber, 2005; Cattivelli and Sayed, 2010)	non-gaussian (Mao and Yang, 2014)

Table: Categories of Methods

The DSE algorithms are not guaranteed to match the performance of the centralized estimator all the time

Hidden Markov Model

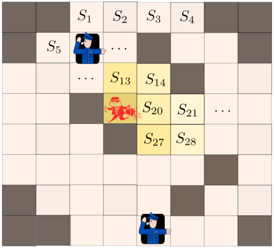
Consider a finite state hidden markov model (HMM) with the following specification: n_s possible states $\mathcal{X} = \{S_1, \cdots, S_{n_s}\}$ and n_z possible observation symbols listed as $\mathcal{Z} = \{O_1, \cdots, O_{n_z}\}$. The random variables $\mathbf{X_k} \in \mathcal{X}$ and $\mathbf{Z^i_k} \in \mathcal{Z}$ represent the state and observation made by agent i at step k, respectively. The realizations of those random variables at step k is denoted as $\mathbf{x_k}$ and $\mathbf{z^i_k}$.

$$\pi_{k-1} \triangleq \mathrm{p}\left(\mathbf{X}_{k-1} | \{\mathbf{z}^i_k\}^{i \in \boldsymbol{I_n}}_{k \in \boldsymbol{I_{k-1}}}\right),$$

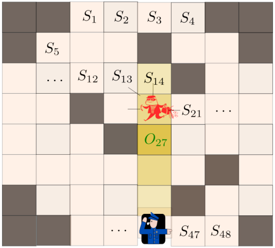
$$\tilde{\pi}_k \triangleq \mathrm{p}\left(\mathbf{X}_{k} | \{\mathbf{z}^i_k\}^{i \in \boldsymbol{I_n}}_{k \in \boldsymbol{I_{k-1}}}\right),$$

$$\pi_k \triangleq \mathrm{p}\left(\mathbf{X}_{k} | \{\mathbf{z}^i_k\}^{i \in \boldsymbol{I_n}}_{k \in \boldsymbol{I_{k}}}\right),$$

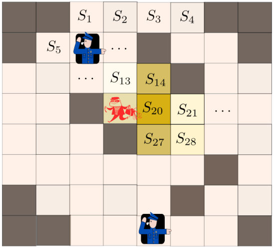
Hidden Markov Model



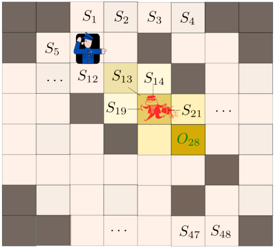
prior



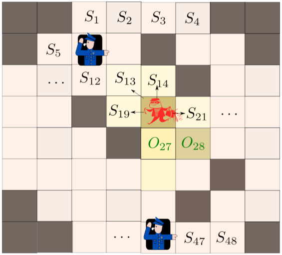
observation 1



motion model



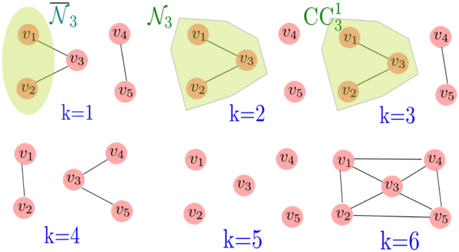
observation 2



posterior

Network Topology

Assume that we have n homogeneous agents associated with the nodes of a graph. These agents can communicate with each other under a time-varying undirected network topology $G_k = \langle \mathcal{V}, \mathcal{E}_k \rangle$ where $\mathcal{V}$ and $\mathcal{E}_k$ are the set of graph nodes and edges at step k respectively. The node corresponding to the i^{th} agent is denoted by v_i. If $(v_i, v_j) \in \mathcal{E}_k$, it means that agents i and j can communicate directly at step k. The set $\overline{\mathcal{N}}_i$ represent neighbors of node v_i that are connected by an edge to v_i. The set $\mathcal{N}_i = \overline{\mathcal{N}}_i \cup \{v_i\}$ will also be used in some of the equations. The set CC_k^i represents the set of agents that are *path-connected* to agent i at step k.



Centralized Estimation on Sensor Networks

In this approach there is a central node in the network that receives observations $\mathbf{z}_k^{\boldsymbol{I_n}} \triangleq \{\mathbf{z}_k^i\}^{i \in \boldsymbol{I_n}}$ from the rest of the nodes.

$$\text{prediction: } \tilde{\pi}_k = \pi_{k-1}\mathcal{P}_{k|k-1}.$$

$$\text{update: } \pi_k = \frac{1}{\eta}\tilde{\pi}_k\mathcal{O}_k.$$

where, $\mathcal{O}_k$ is a $n_s \times n_s$ diagonal matrix of likelihoods $\mathrm{p}(\mathbf{z}_k^{\boldsymbol{I_n}}|\mathbf{X}_k)$.

Distributed Consensus Based Estimation

If all agents share the same prior information, they can recover the centralized estimator's performance if they can reach a consensus over the product of measurement probabilities.

$$\tilde{l}_k = \frac{1}{n} \log \prod_{i=1}^{n} \mathcal{O}_k^i = \frac{1}{n} \sum_{i=1}^{n} \log \mathcal{O}_k^i = \frac{1}{n} \sum_{i=1}^{n} \tilde{l}_k^i$$

Distributed averaging methods can be applied here. All the nodes need to reach a consensus over the log of the joint measurement probabilities (log-of-likelihood). Once the consensus is reached, the updated estimate is

$$\pi_k = \frac{1}{\eta} \overbrace{\underbrace{\pi_{k-1}}_{\text{prior}} \mathcal{P}_{k|k-1}}^{\text{prediction}} \underbrace{e^{n\tilde{l}_k}}_{\text{likelihood}} .$$

Distributed Averaging by Metropolis-Hastings-Markov-Chains

Consensus over likelihoods using a distributed averaging method based on Metropolis-Hastings Markov Chains (MHMC), Xiao and Boyd (2003).

$$\psi^{\mathbf{i}}(\mathbf{m}+\mathbf{1}) = \sum_{\mathbf{j}=\mathbf{1}}^{|\mathcal{N}_{\mathbf{i}}|} \gamma_{\mathbf{ij}}(\mathbf{m})\psi^{\mathbf{j}}(\mathbf{m}),$$

$$\text{s.t. } \sum_{m} \gamma_{ij}(m) = 1, \ \psi^{\mathbf{i}}(\mathbf{0}) = \tilde{\mathbf{l}}^{\mathbf{i}}_{\mathbf{k}},$$

to calculate the average of the values on the graph nodes in which $d_i(m) = |\overline{\mathcal{N}}^i|$ is the degree of the node v_i, and

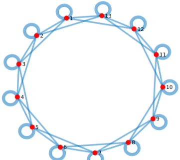
$$\gamma_{ij}(m) = \begin{cases} \frac{1}{1+\max\{d_i(m), d_j(m)\}} & \text{if } (i,j) \in \mathcal{E}_m, \\ 1 - \sum\limits_{(i,n)\in\mathcal{E}} \gamma_{in} & \text{if } i = j, \\ 0 & \text{otherwise.} \end{cases}$$

With this messaging passing protocol,

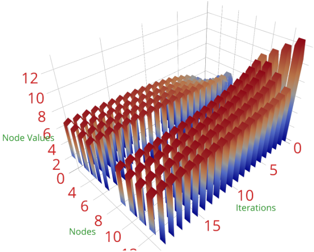
$$\lim_{m\to\infty} \psi^{\mathbf{i}}(\mathbf{m}) = \tilde{\mathbf{l}}_{\mathbf{k}}.$$

Distributed Averaging by Metropolis-Hastings-Markov-Chains

How MHMC distributed averaging works in practice



Network Topology



Evolution of node values

Why can't we use distributed averaging when priors are not the same?

We have to calculate the following

$$\pi_k = \frac{1}{\eta}\left[\cup_{i=1}^{n} \pi_{k-1}^{i}\right]\mathcal{P}_{k|k-1} e^{n\tilde{l}_k}.$$

We need to avoid double counting of mutual information

$$\pi^i \cup \pi^j = \pi^i + \pi^j - \pi^i \cap \pi^j$$

Conservative Fusion

If priors have shared information

- track the mutual information (Channel Filter)
- calculate a conservative approximation of fused priors

Conservative Approximation of a PMF

A set of sufficient conditions that a PMF $\tilde{p}(\mathbf{X})$ has to satisfy in order to be a conservative approximation of another PMF $p(\mathbf{X})$, which are (Bailey et al. (2012))

- The non-decreasing entropy property,

$$H(p(\mathbf{X})) \leq H(\tilde{p}(\mathbf{X}))$$

- The order preservation property that, $\forall \mathbf{x_i}, \mathbf{x_j} \in \mathbf{X}$,

$$p(\mathbf{x_i}) \leq p(\mathbf{x_j}) \text{ iff } \tilde{p}(\mathbf{x_i}) \leq \tilde{p}(\mathbf{x_j})$$

Conservative Fusion of two PMFs

Two probability distribution functions $\mathrm{p}_a(\mathbf{X}|\mathbf{I_a})$ and $\mathrm{p}_b(\mathbf{X}|\mathbf{I_b})$ can be fused together following the *Geometric Mean Density Rule (GMD)*

$$\begin{aligned}\mathrm{p}_c(\mathbf{X}) &= \frac{1}{\eta_c}\mathrm{p}_a(\mathbf{X}|\mathbf{I_a})^{\omega}\mathrm{p}_{\mathbf{b}}(\mathbf{X}|\mathbf{I_b})^{\mathbf{1}-\omega} \\ &= \frac{1}{\eta_c}\mathrm{p}_a(\mathbf{X}|\mathbf{I_a}\setminus\mathbf{I_b})^{\omega}\mathrm{p}_{\mathbf{b}}(\mathbf{X}|\mathbf{I_b}\setminus\mathbf{I_b})^{\mathbf{1}-\omega}\mathrm{p}_{\mathbf{a}}(\mathbf{X}|\mathbf{I_a}\cap\mathbf{I_b}),\end{aligned}$$

Several criteria have been proposed to choose the ω_i. One such criteria is (Ajgl and Šimandl, 2015)

$$\tilde{\pi} = \arg\min_{\pi}\max_{i}\{\mathcal{D}(\pi\|\tilde{\pi}^i)\},$$

in which the $\mathcal{D}(\pi\|\tilde{\pi}^i)$ is the Kullback-Leibler divergence between densities π and $\tilde{\pi}^i$.

Iterative Conservative Fusion

At first iteration of consensus, $m = 0$, for each agent j, take the most recent local estimate π^j_{k-1} and calculate the prediction $\tilde{\pi}^j_k$. Initialize the local consensus variable using local prediction and local observation $\mathcal{O}^j_k$

$$\phi^j(0) = \frac{1}{\eta_i}\tilde{\pi}^j_k \mathcal{O}^j_k.$$

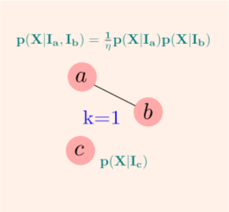
Iteratively update

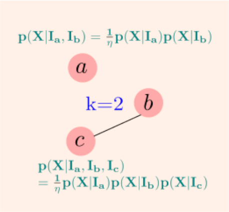
$$\phi^i(m+1) = \frac{1}{\eta^*}\prod_{j\in\mathcal{N}^i(m)} \left[\phi^j(m)\right]^{\omega^*_j}.$$

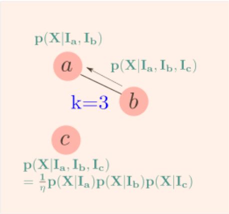
After convergence (CC^i_k is the set of agents that form a connected group with agent i at step k)

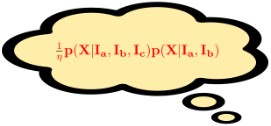
$$\lim_{m\to\infty} \phi^{\mathbf{i}}(\mathbf{m}) = \phi^* = \frac{\mathbf{1}}{\eta}\prod_{\mathbf{j}\in \boldsymbol{I}_{\mathbf{CC^i_k}}} \left[\phi^{\mathbf{j}}(\mathbf{0})\right]^{\omega^*_{\mathbf{j}}}, \forall \mathbf{i} \in \mathbf{CC^i_k}.$$

Why Hybrid Algorithm?





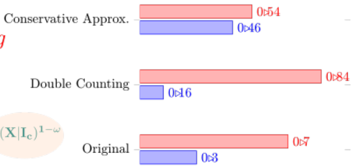


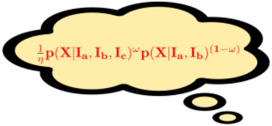


Distributed Consensus

$$\frac{1}{\eta}p(X|I_a)p(X|I_b)p(X|I_c) \quad p(X|I_a)p(X|I_b)$$

Double Coounting





Conservative Fusion

$$\frac{1}{\eta}p(X|I_a)p(X|I_b)p(X|I_c)p(X|I_c)^{1-\omega}$$

Information Dilution

Hybrid Algorithm

Recap) We wanted to get as close as possible to centralized estimator

$$\pi_k = \frac{1}{\eta}\pi_{k-1}\mathcal{P}_{k|k-1}\prod_{i=1}^{n}\mathcal{O}_k^i = \frac{1}{\eta}\overbrace{\underbrace{\pi_{k-1}}_{\text{prior}}\mathcal{P}_{k|k-1}}^{\text{prediction}}\underbrace{e^{n\tilde{l}_k}}_{\text{likelihood}}.$$

If the network remained connected all the time and If enough time was given for consensus algorithm to converge, agent priors matched the prior of the centralized estimator and distributed averaging alone could solve the DSE problem. Iterative Conservative Fusion would help with unequal prior, network disconnection and, avoiding double counting.

$$\pi_k^* = \frac{1}{\eta}\prod_{j\in \boldsymbol{I}_{\mathrm{cc}_k^i}}\left[\pi_{k-1}^i\right]^{\omega_j^*}\mathcal{P}_{k|k-1}\prod_{j\in \boldsymbol{I}_{\mathrm{cc}_k^i}}\left[\mathcal{O}_k^i\right]^{\omega_j^*}.$$

Hybrid of ICF on priors and Distributed Averaging on Likelihoods will give us

$$\pi_k^* = \frac{1}{\eta}\prod_{j\in \boldsymbol{I}_{\mathrm{cc}_k^i}}\left[\pi_{k-1}^i\right]^{\omega_j^*}\mathcal{P}_{k|k-1}\prod_{j\in \boldsymbol{I}_{\mathrm{cc}_k^i}}\left[\mathcal{O}_k^i\right].$$

Input : π^i_{k-1}

Use (1) to calculate $\tilde{\pi}^i_k$

Collect local observation z^i_k and calculate $\mathcal{O}^i_k$ and $\tilde{I}^i_k$

Initialize consensus variables

$$\phi^i(0) = \tilde{\pi}^i_k, \quad \psi^i(0) = \tilde{I}^i_k$$

$m = 0$

while *NOT CONVERGED* **do**

  BROADCAST$[\psi^i(m), \phi^i(m)]$

  RECEIVE$[\psi^j(m), \phi^j(m)] \quad \forall j \in \mathcal{N}^i$

  Collect received data

$$\mathcal{C}^i(m) = \{\phi^{j \in \mathcal{N}^i}(m)\}, \quad \mathcal{M}^i(m) = \{\psi^{j \in \mathcal{N}^i}(m)\}$$

  Do one iteration of ICF on consensus variables for local prior information $\mathcal{C}^i_m$

$$\phi^i(m+1) = \mathrm{ICF}(\mathcal{C}^i(m))$$

  Do one iteration of MHMC on consensus variables for new information

$$\psi^i(m+1) = \mathrm{MHMC}(\mathcal{M}^i(m))$$

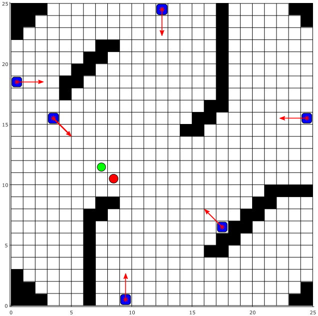
  $m = m + 1$

end

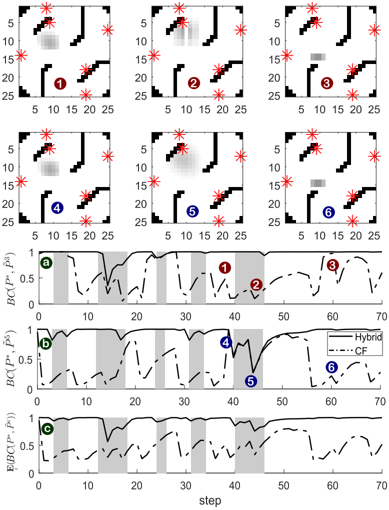
Calculate the posteriors according to:

$$\pi^i_k = e^{|\mathrm{CC}^j_k|\psi^i(m)}\phi^i(m)$$

Algorithm 1: Hybrid Method



Multi Agent Tracking Scenario



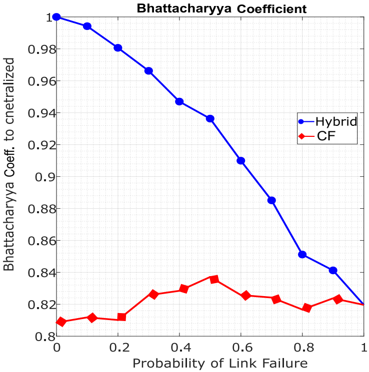
Estimation performance in the tracking example

To quantify differences, we use the Bhattacharyya coefficient (BC) Bhattacharyya (1946) between the estimation results and the centralized estimator. BC can be used to evaluate the similarity of two probability mass functions, $\pi_1(\mathbf{X})$, $\pi_2(\mathbf{X})$ as:

$$BC(\pi_1(\mathbf{X}), \pi_2(\mathbf{X})) = \sum_{\mathbf{x} \in \mathbf{X}} \sqrt{\pi_1(\mathbf{x})\pi_2(\mathbf{x})}.$$

In the case of complete similarity, $p_1 = p_2$, we have $BC(p_1, p_2) = 1$. Moreover, $BC(p_1, p_2) = 0$ describes maximal dissimilarity.

Experiments: Random High Dimensional HMM



In a second experiment we have evaluated the robustness of the proposed method for networks with different likelihoods of link failure. We report the Bhattacharyya coefficient vs. link failure probability for a general decentralized HMM with a network of size 20 and state size 30 with each node roughly connected to 10% of the other nodes. We simulate the system multiple times, each time for 150 time steps but with different probability of link failure. At each step, given a probability of failure for each link, some links in the graph will randomly be disconnected.

Questions ?

References I

Ajgl, J. and Šimandl, M. (2015). Design of a robust fusion of probability densities. In *American Control Conference (ACC), 2015*, pages 4204–4209. IEEE.

Bailey, T., Julier, S., and Agamennoni, G. (2012). On conservative fusion of information with unknown non-gaussian dependence. In *Information Fusion (FUSION), 2012 15th International Conference on*, pages 1876–1883. IEEE.

Battistelli, G., Chisci, L., and Fantacci, C. (2014). Parallel consensus on likelihoods and priors for networked nonlinear filtering. *IEEE Signal Processing Letters*, 21(7):787–791.

Bhattacharyya, A. (1946). On a measure of divergence between two multinomial populations. *Sankhyā: The Indian Journal of Statistics*, pages 401–406.

References II

Cattivelli, F. S. and Sayed, A. H. (2010). Diffusion strategies for distributed kalman filtering and smoothing. *IEEE Transactions on automatic control*, 55(9):2069–2084.

Durrant-Whyte, H., Stevens, M., and Nettleton, E. (2001). Data fusion in decentralised sensing networks. In *Proceedings of the 4th International Conference on Information Fusion*, pages 302–307.

Hu, J., Xie, L., and Zhang, C. (2012). Diffusion kalman filtering based on covariance intersection. *Signal Processing, IEEE Transactions on*, 60(2):891–902.

Mao, L. and Yang, D. W. (2014). Distributed information fusion particle filter. In *Intelligent Human-Machine Systems and Cybernetics (IHMSC), 2014 Sixth International Conference on*, volume 1, pages 194–197.

References III

Olfati-Saber, R. (2005). Distributed kalman filter with embedded consensus filters. In *Decision and Control, 2005 and 2005 European Control Conference. CDC-ECC'05. 44th IEEE Conference on*, pages 8179–8184. IEEE.

Simonetto, A., Keviczky, T., and Babuka, R. (2010). Distributed nonlinear estimation for robot localization using weighted consensus. In *2010 IEEE International Conference on Robotics and Automation*, pages 3026–3031.

Tamjidi, A., Chakravorty, S., and Shell, D. (2016). Unifying consensus and covariance intersection for decentralized state estimation. In *Intelligent Robots and Systems (IROS), 2016 IEEE/RSJ International Conference on*, pages 125–130. IEEE.

Wang, Y. and Li, X. (2012). Distributed estimation fusion with unavailable cross-correlation. *Aerospace and Electronic Systems, IEEE Transactions on*, 48(1):259–278.

References IV

Xiao, L. and Boyd, S. (2003). Fast linear iterations for distributed averaging. In *Decision and Control, 2003. Proceedings. 42nd IEEE Conference on*, volume 5, pages 4997–5002 Vol.5.

Xiao, L., Boyd, S., and Lall, S. (2005). A scheme for robust distributed sensor fusion based on average consensus. In *Proceedings of the 4th international symposium on Information processing in sensor networks*, page 9. IEEE Press.