Adaptive Sampling strategy for Mobile Robotic Sensor Network using Proximal ADMM

PRESENTER:
Viet-Anh Le

BACKGROUND:

- Adaptive sampling is the fundamental problem of Mobile Sensor Network applications, it is how to optimally drive the mobile sensors on sampling paths so that the information gained by the sensor measurements is maximal
- In applications where a Gaussian Process (GP) is employed to model a spatial field and then to predict the field at unobserved locations, the adaptive sampling problem can be formulated as minimizing the negative log determinant of the GP predicted covariance matrix

$$\begin{aligned} &\underset{\{\mathbf{s}_{t+1},\mathbf{u}_t\}}{\text{minimize}} \; -\log\det\boldsymbol{\Sigma}(\mathbf{s}_{t+1}) \\ &\text{subject to} \\ &\quad \mathbf{s}_{i,t+1} = \mathbf{s}_{i,t} + \mathbf{u}_{i,t}\Delta T, \; \forall i = 1,\dots,M \\ &\quad \mathbf{u}_{i,t} \in \mathcal{U}_i, \; \mathbf{s}_{i,t+1} \in \Omega_{i,t}, \; \forall i = 1,\dots,M. \end{aligned} \tag{5}$$

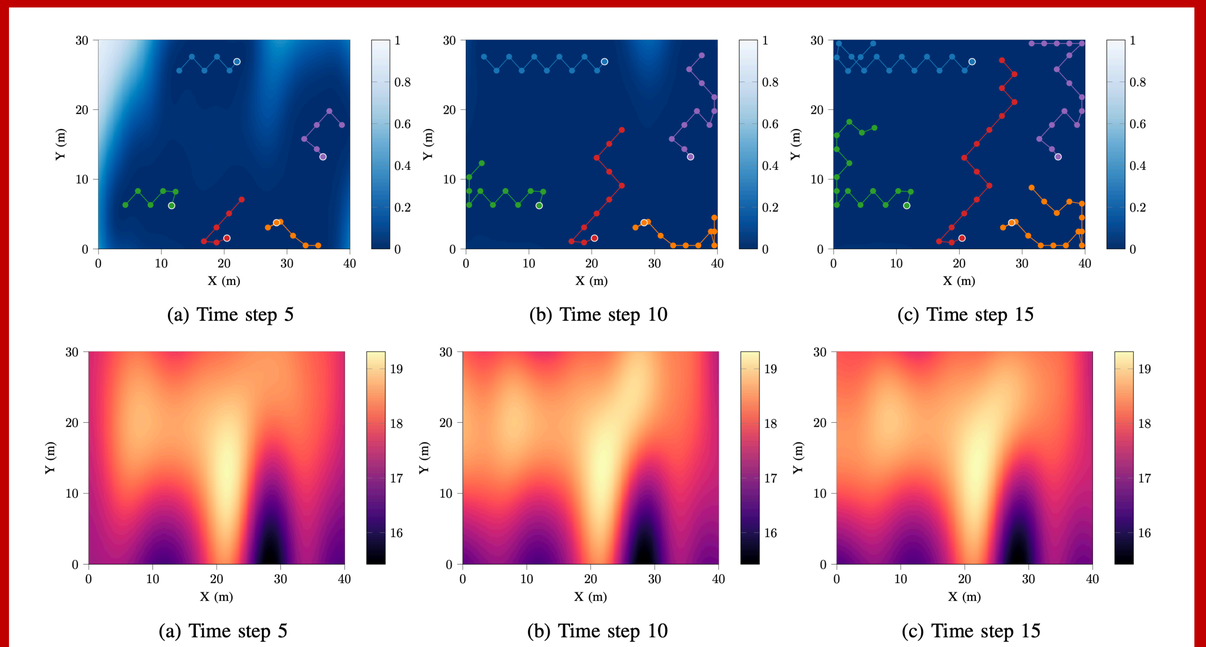
CHALLENGES:

- Solve in continuous domain: the optimization problem is highly complex and nonconvex
- Solve in discrete domain (grid-based methods): combinatorial NP-hard and only a near-optimal solution can be obtained

METHODS

- Split the problem into two parts: one involving the log determinant of the covariance matrix (non-convex), one involving other convex constraints → a consensus optimization problem
- Use the first-order approximation of the log determinant of the predicted covariance matrix

An efficient approach to address **the adaptive sampling optimization problem** in a **Mobile Robotic Sensor Network** for environmental monitoring.

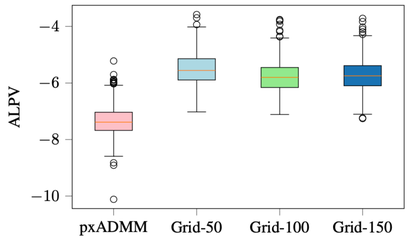


(a) Time step 5 (b) Time step 10 (c) Time step 15

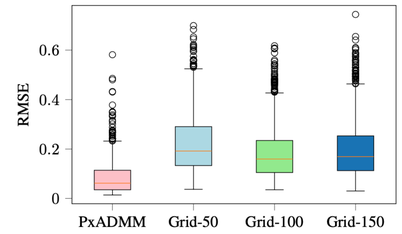
(a) Time step 5 (b) Time step 10 (c) Time step 15

The predicted variances with the sampling trajectories of the mobile sensors (top) and predicted means (bottom) at some specific time instants

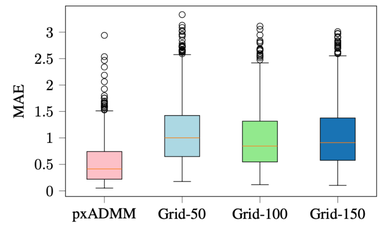
COMPARISONS



(a) The average logarithm of the predicted variances



(b) The root mean squared errors



(c) The maximum absolute errors

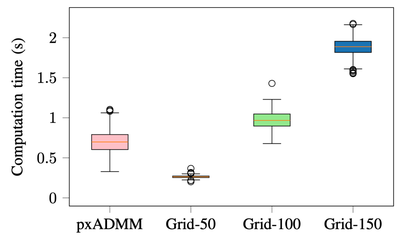


Fig. 8: The boxplot for average computation times in 1000 simulations.

Viet-Anh Le, Truong X. Nghiem