Stien Coverage
A Distribution-Matching Multi-agent Deployment for Coverage

Solmaz S. Kia
Graduate student contributors: Yi-Fan Chung and Donipolo Ghimire

Mechanical and Aerospace Engineering Dept.
University of California Irvine
Dec. 2023

30th IEEE International Conference on High Performance Computing, Data, and Analytics
Workshop on Indo-US Initiatives
Multi-agent coverage problem

Credit: https://www.mdpi.com/2673-6489/2/2/19

Credit: https://openai.discovery/id-100769566430091344

Credit: https://www.nsenergybusiness.com/features/worlds-biggest-offshore-oil-spills/

Credit: https://x.com/PrinSciAdvOff/status/1430874334038532096?s=20
Multi-agent coverage problem: the challenge

Vast area
Limited resources
A ticking clock

Objective: the best use of resources

The coverage problem involves strategically placing a limited number of agents across the area of interest in such a way that a certain coverage measure is maximized.
How to approach the problem

The information at hand:

\[ p(x) \]: distribution of the coverage event
In coverage problems with a limited number of agents, the essence of the proposed solutions is

- divide the area into subregions and
- assign an agent to each one,

with the goal of maximizing a certain coverage measure.

\[ \{x_i\}_{i=1}^n \leftarrow \minimize \sum_{i=1}^n \int (||x - x_i||^2 - r_i^2)p(x)dx \]
Why alternatives to Voronoi partitioning?

\[
\{ x_i \}_{i=1}^n \leftarrow \text{minimize} \sum_{i=1}^n \left( \| x - x_i \|^2 - r_i^2 \right)p(x)dx
\]
Every agent has

- a quality-of-service distribution $s_i(x|x_i, \theta_i)$, and
- An effective coverage footprint $C_i(x|x_i, \theta_i)$

Collective quality of service of the agents
\[
q(x) = \frac{1}{N} \sum_{i=1}^{N} s_i(x|x_i, \theta_i)
\]

Distribution-Matching Deployment
\[
(x_i^*, \theta_i^*)_{i=1}^{N} = \text{argmin } KL[q(x)||p(x)]
\]

$p(x)$: coverage event distribution
**Distribution-matching deployment: proposed solution**

\[ \mathcal{A}: \text{Agent set} \]

\[ \mathcal{S}: \text{set of points of interest (PoIs)} \]

Optimal Agent Assignment

\[ \mathbf{Z}^* = \arg \min \sum_{j \in \mathcal{S}} \sum_{i \in \mathcal{A}} Z_{i,j} C_{i,j}^*, \]

\[ Z_{i,j} \in \{0, 1\}, \quad i \in \mathcal{A}, \quad j \in \mathcal{S}, \]

\[ \sum_{i \in \mathcal{A}} Z_{i,j} = 1, \quad \forall i \in \mathcal{A}, \]

\[ \sum_{j \in \mathcal{S}} Z_{i,j} \leq 1, \quad \forall j \in \mathcal{S}. \]

\[ C_{i,j}^* = \min_{\theta_i \in \Theta} \{ \mathcal{KL} \left( s(x|x_j, \theta_i) \| p(x) \right) \} \text{ for } x \in \mathcal{C}_i(x|x_j, \theta_i) \]

**Research question:** how to determine the PoIs?
Distribution-matching deployment: a GMM clustering approach

\[
p^i(x) = \sum_{k=1}^{N_s} \pi_k^i \mathcal{N}(x | \mu_k^i, \Sigma_k^i)
\]

\[
s_i(x_i, \theta_i) = \mathcal{N} \left( x_i, \begin{bmatrix} (\sigma_x^i)^2 & 0 \\ 0 & (\sigma_y^i)^2 \end{bmatrix}, \theta_i \right)
\]

Clustering is sensitive to initialization
- Does not necessarily do a good job in clustering for lower number of robots than real number of clusters
- Does not handle the overlapping area well
Use statistical methods to find samples from the distribution $p(x)$:

- Markov Chain Monte Carlo (MCMC)
- **Stein variational Gradient Descent (SVGD)**
  
  [Q. Liu, J. Lee, and M. Jordan, ICML, 2016]

Extracting points of interests using Stein variational gradient descent

**Algorithm 1**: Stein Variational Gradient Descent

**Input**: The score function $\nabla_x \log p(x)$.

**Goal**: A set of particles $\{x_i\}_{i=1}^n$ that approximates $p(x)$.

**Initialize** a set of particles $\{x_i^{(0)}\}_{i=1}^n$; choose a positive definite kernel $k(x, x')$ and step-size.

**For** iteration $l$ do

$$x_i^{(l+1)} \leftarrow x_i^{(l)} + \phi^*(x_i^{(l)}) \quad \forall i \in \{1, \ldots, n\}$$

where,

$$\phi^*(x) = \frac{1}{n} \sum_{j=1}^n \left[ \nabla \log p(x_j) k(x_j, x) + \nabla x_j k(x_j, x) \right].$$

---

**SVGD**: Inference through deterministic map

A weighted gradient of the log $p(x)$, known as the repulsive force, intuitively pushes particles apart when they approach each other closely, preventing them from collapsing into a single mode.

$$x \leftarrow T(x) = x + \epsilon \phi(x)$$

**Perturbation direction**

**Step size**
Extracting points of interest using Stein variational gradient descent: how to avoid overlaps

Trivial implementation of the SVGD

Through proper design of the repulsive force of SVGD
(a) Sensor deployment for heterogeneous sensors: (left) Stein Coverage, (right) Voronoi partitioning using power diagrams.

(b) Comparison of sensor deployment: (left) proposed Stein Coverage, (right) Voronoi partitioning using power diagrams, when the number of sensors is less than the number of clusters.
(a) Sensor deployment for anisotropic heterogeneous sensors. The level sets represent the multimodal distribution of the target points (blue dots) and the ellipsoid encircling the level sets represents the configuration of anisotropic footprint calculated using the proposed algorithm.

(b) Sensor deployment for heterogeneous anisotropic sensors when the number of sensors is less than the number of clusters.
Monitoring problems can be cast as deployment problem:

- Using predefined sweep patterns
- Footprint is defined by battery life

- Stien Coverage
  - Identifies important points to start the sweep from
  - The points can be given to Voronoi partitioning to divide the area.
Summary

\( \mathcal{A}: \) Agent set

\( \mathcal{S}: \) set of points of interest (Pols)

Optimal Agent Assignment

\[
\mathbf{Z}^* = \arg \min \sum_{j \in \mathcal{S}} \sum_{i \in \mathcal{A}} Z_{i,j} C_{i,j}^*,
\]

\( Z_{i,j} \in \{0, 1\}, \quad i \in \mathcal{A}, \quad j \in \mathcal{S}, \)

\[
\sum_{i \in \mathcal{A}} Z_{i,j} = 1, \quad \forall i \in \mathcal{A},
\]

\[
\sum_{j \in \mathcal{S}} Z_{i,j} \leq 1, \quad \forall j \in \mathcal{S}.
\]

Use SVGD to find Pols

\[
C_{i,j}^* = \min_{\theta_i \in \Theta} \{ KL \left( s(x_j | x_i, \theta_i) \| p(x_i) \right) \} \quad \text{for} \ x \in \mathcal{C}_i(x_j | x_i, \theta_i)
\]


Student Contributors

Yi-fan Chung

Doni Ghimire

Contact information: solmaz@uci.edu