Distributed deployment with location dependent Sensing Model

Notes from summer work at DRL

09/01/06

ABSTRACT

We focus on the problem of distributed deployment a sensor network in 1D and 2D domains where the sensor's perception model is a function of its location.

1. MOTIVATION

Coverage literature has focussed on scenarios where the sensor's performance does not depend on its location. In a lot of applications, the sensor's performance is influenced by environmental factors which vary within the domain. For example, in a network of cameras deployed to detect an intruder, each camera's field of view depends on lighting, contrast, etc, which will be a function of its location. The communication range of radios is a complex function of environmental factors and will vary from location to location within a network.

2. FORMULATION

Q is a bounded convex region, $\mathbf{p} = \{\mathbf{p_1}, \mathbf{p_2}, ..., \mathbf{p_n}\}$ are locations of n sensors. The sensing performance of each sensor i at point $\mathbf{q}$ is given by $f(\mathbf{p_i}, \mathbf{q})$. Let $\Phi(\mathbf{q})$ be the density function at $\mathbf{q}$. Depending on the application, we might be interested in

1. **Average Coverage:** Minimize the expected degradation given by

$$H = \int_{\mathbf{q} \in Q} \min_i \{f(\mathbf{p_i}, \mathbf{q})\} \Phi(\mathbf{q}) \tag{1}$$

2. **Worst case Coverage:** Minimize the worst case degradation given by

$$W = \max_{\mathbf{q} \in Q} \{\min_i \{f(\mathbf{p_i}, \mathbf{q})\} \Phi(\mathbf{q})\} \tag{2}$$

The above formulation results in a generalized voronoi partition of the domain Q. The term$\{\min_i\{f(\mathbf{p_i}, \mathbf{q})\}\Phi(\mathbf{q})\}$ breaks down into a summation over the partition. Assume f has a form that is separable that can be expressed as $f(\mathbf{p_i}, \mathbf{q}) = A(\mathbf{p_i}).g(||\mathbf{p_i} - \mathbf{q}||)$ Assume that f has a form that is separable that can be expressed as $f(\mathbf{p_i}, \mathbf{q}) = A(\mathbf{p_i}).g(||\mathbf{p_i} - \mathbf{q}||)$ where g is the same for all sensors and $A(\mathbf{p_i})$ is a scaling factor that is location dependent. (Note that g can be made g_i so that its different for each sensor without depending on their location).

In general, depending on the nature of f the voronoi partition can have holes. The condition for no-holes can be expressed as `[Ajay please add]`

3. PATH COVERAGE

Given an arbitrarily complex path and a sensing model which is a function of location. The sensors do not have access to the global sensing model but can estimate it over their neighborhood. Find a distributed control law for the motion of sensors such that path is covered.

3.1. Approach

- **sequential:** If the global sensing model is know a priori, we can use a sequential greedy approach. If the range function is not known, we can start with a uniform distribution of nodes, estimate the range function and follow the sequential placement strategy. This method will give optimal solution if the path does not have loops. `[Sameera add a proof]`. Since the global sensing model is known, we can also control the direction of cameras.

- **distributed:** Each node can estimate the local behavior of the range function using the readings from its neighbors. It computes the points at which its performance is equal to that of its neighbors is equal. It chooses a new location such that its performance at these "intersection" points is equal. `[add figure]`. We can prove convergence `[Ajay please add]`. This approach gives good results in simulation. The nodes converge even when the path is a loop and the range function is discontinuous (we used step functions) or varying rapidly. Currently, we do not control the direction of cameras along the path.

3.2. Extension to 2D

In the absence of obstacles, if the perception function is uniform, this is exactly Bullo's framework. Using the perception function in place of the "density function" in his framework does not solve the problem.

Consider the case when the perception model is a function of the sensors and not location i.e., each sensor has a different range but the range does not change with location. An example is mobile server nodes in a sensor network that have different capacities. The capacities do not change with location. We can use Lyod's descent here with additively weighted voronoi diagrams and prove convergence.

For the case when the sensor's range also depends on location, we can prove convergence for the linear case. `[Ajay please add]`.

3.3. Camera Coverage

Location optimization of camera networks with infinite range has been extensively studied in art gallery literature. The same will apply for cameras with limited field of view placed in environments with obstacles if the dimensions of the open area are small compared to the camera's range. Consider the other extreme where the dimensions of the open areas are very large compared to the camera's range. This problem boils down to packing that is addressed for other sensors.

One approach is to solve the coverage problem along the boundaries of obstacles. This will reduce to a 1D path type problem. The open space can be covered using packing.