Sensor Network Topology Inference

Dimitri Marinakis, Gregory Dudek *Centre for Intelligent Machines, McGill University*
3480 University St, Montreal, Quebec, Canada H3A 2A7
{dmarinak,dudek}@cim.mcgill.ca

***Abstract*— We present a method for inferring the topology of a sensor network given non-discriminating observations of activity in the monitored region. This is accomplished based on no prior knowledge of the relative locations of the sensors and weak assumptions regarding environmental conditions. Our approach employs a two-level reasoning system made up of a stochastic Expectation Maximization algorithm and a higher level search strategy employing the principle of Occam's Razor to look for the simplest solution explaining the data. The result of the algorithm is a Markov model describing the behaviour of agents in the system and the underlying traffic patterns. Numerical simulations and experimental assessment conducted on a real sensor network suggest that the technique could have promising real world applications in the area of sensor network self-configuration.**

I. Introduction

In this paper we address the self-calibration problem of inferring the *topology*, or inter-node connectivity, of a sensor network given non-discriminating observations of activity in the environment. We are interested in recovering a representation of the network that identifies physical inter-sensor connectivity from the point of view of an agent navigating the environment. This topological information differs from a metric representation which identifies the relative locations of the sensors but does not provide information about the layout of the region, or obstructing objects within it. We assume that we have no prior knowledge of the relative locations of the sensors and that we have only a limited knowledge of the type of activity present in the environment. We must use observational data returned from our sensors to understand the motion of agents present in the environment. By infering underlying patterns in their motions we can then recover the relationships between the sensors of our network.

Our approach employs a two-level reasoning system. The first level is made up of our fundamental topology inference algorithm that takes the sensor observations and environmental assumptions as inputs and returns the network parameters. This algorithm is formulated using Monte Carlo Expectation Maximization (MCEM), but it depends on fixed values for certain numerical parameters that represent *a priori* knowledge regarding traffic patterns in the environment. The second level searches over the input parameter space of the first level algorithm to find a global solution that optimizes a more abstract objective function based on the principle of Occam's Razor. [1] The final output of the two-level approach is a probabilistic model of the sensor network connectivity graph and the underlying traffic trends.

[1]Occam's Razor is the principle enunciated by William of Occam that the simplest explanation is the best.

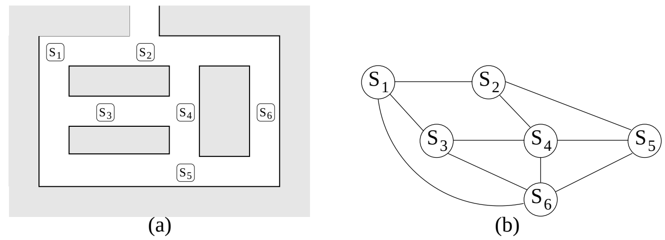

Fig. 1. An example of a sensor network which we wish to calibrate. a) The original ad-hoc deployment. b) The desired topological connectivity map of the network.

It is recognized that self-calibration and other more general self-configuration algorithms are important issues for both multi-robot systems and for sensor networks [1], [2]. The main point is that a network must operate autonomously in an dynamic environment. It should be capable of re-organizing itself to handle network changes such as individual node failures or changes in communication range.

A lot of work has focused on the issue of sensor network self-calibration, especially in regards to metric self-localization [3], [4], although there have also been past efforts to recover topological inter-sensor relationships [5], [6], [7]. The work conducted by Ellis, Makris, and Black [7] [6] on topology inference for camera-networks attempts to solve a similar problem to the one we consider in this work. Their technique relies on exploiting temporal correlation in observations of agent movements throughout the environment by employing a threshold technique that looks for peaks in the temporal distribution of travel times between entrance-exit pairs; a clear peak suggesting that a correspondence exists. The presented results from a six camera network that suggest their approach could be used to efficiently produce an approximate network connectivity graph but when the network dynamics are complex or the traffic distribution exhibits substantial variation, it would appear the technique will have difficulty.

II. Problem Description

In this work, our goal is to exploit the motion of agents in the environment in order to recover the connectivity parameters of a sensor network. Specifically, we want to learn the inter-node transition likelihoods A and the inter-node delay time distributions D. This information will be a product of the environment, the placement of the individual network nodes, and the behaviour of the agents within the system.

For example, let us assume that the network show in Figure 1(a) has been deployed for some purpose such as surveillance and requires knowledge of the inter-node connectivity in order to fulfill its function. During some initial calibration period the network collects observations of agents passing by each sensor. The problem we are trying to solve is how to use these collected observations to construct the topological description of the network shown in Figure 1(b).

We further restrict the problem by allowing observations collected at each of the nodes to indicate only the present or absence of motion. In other words, we assume that the sensors are non-discriminating and are capable of reporting only that they have detected something, but are not capable of providing a description or signature of what they have detected.

Formally, we describe the problem of topology inference in terms of the inference of a weighted directed graph which captures the spatial relationships between the positions of the sensors' nodes. The motion of multiple agents moving asynchronously through a sensor network embedded region can be modeled as a semi-Markov process. The network of sensors is described as a directed graph $G = (V, E)$, where the vertices $V = v_i$ represent the locations where sensors are deployed, and the edges $E = e_{i,j}$ represent the connectivity between them; an edge $e_{i,j}$ denotes a path from the position of sensor v_i to the position of sensor v_j. The motion of some number N agents in this graph can be described in terms of their transition probability across each of the edges $A_n = \{a_{ij}\}$. The goal of our work is to estimate the parameters describing this semi-Markov process given the observations O and the vertices V.

III. Approach

Our approach employs a two-level reasoning system. The first level is made up of a topology inference algorithm based on random sampling. The algorithm takes the sensor observations and some assumptions regarding the environment as inputs and returns the inferred network parameters. The second level searches over the input parameter space of the first level algorithm to find a global solution.

The first level algorithm attempts to learn the network topology by dividing the problem into two inter-dependent sub-problems: first, inferring the association between sensor observations and motion sources (agents) moving though the environment, and second, inferring the network connectivity parameters that best describe these inter-node transitions.

In the sampling portion of the problem, we select *data associations* that match up each detection event observed at a particular sensor with one of the agents assumed to be moving through the environment. We refer to an individual sample of this data association as an *ownership vector*. By assigning each observation to a specific agent, an ownership vector essentially constructs a trajectory through the environment for each agent believed to be present. We then use these samples of the ownership vector to re-estimate our connectivity parameters.

We formulate the problem as a stochastic version of the Expectation Maximization algorithm and simultaneously solve both the correct observation data correspondences and the correct network parameters. We iterate over the following two steps:

1) *The E-Step:* which calculates the expected log likelihood of the complete data given the current parameter guess: $Q\big(\theta, \theta^{(i-1)}\big) = E\Big[\log p(O, Z|\theta)|O, \theta^{(i-1)}\Big]$ where O is the vector of binary observations collected by each sensor, and Z represents the hidden variable that determines the data correspondence between the observations and agents moving throughout the system.
2) *The M-Step:* which then updates our current parameter guess with a value that maximizes the expected log likelihood: $\theta^{(i)} = \mathrm{argmax}_\theta\, Q\big(\theta, \theta^{(i-1)}\big)$

We employ Monte Carlo Expectation Maximization [8] to calculate the E-Step because of the intractability of summing over the high dimensional data correspondences.

At every iteration we obtain M samples of the ownership vector L, which are then used to re-estimate the connectivity parameter θ (the M-Step). At every iteration of the algorithm the likelihood of the ownership vector increases, and the process is terminated when subsequent iterations result in very small changes to our current belief of the network parameters.

The second level of our approach treats the topology inference algorithm as a 'black box' and attempts to search over its input parameter space to find reasonable solutions. We construct a heuristic evaluation function that quantitatively assesses a potential solution based on the principle of Occam's Razor.

The first level topology inference algorithm takes the following inputs: the observations O; the assumed number of agents in the environment N; and a second parameter that determines the probability (in the framework of our algorithm) at which a particular observational data point is considered an outlier and is discarded. The outputs of the algorithm are the network parameters θ and the *ratio* R_{data} of data incorporated into the parameter updates. Different input values result in different environmental assumptions and, hence, produce different outputs.

We create a metric that attempts to assess the validity of a solution by making the assumption that a good solution both explains the majority of the data and is as *simple* as possible. This principle, known as Occam's razor, states, "if presented with a choice between indifferent alternatives, then one ought to select the simplest one." The concept is a common theme in computer science and underlies a number of approaches in AI; *e.g.* hypothesis selection in decision trees and Bayesian classifiers [9].

Our simplicity metric incorporates a measure of the simplicity of the transition matrix and the amount of data explained by the solution:

$$Q_{simp} = \Bigg(\sum_{a_i \in A} (a_i)^2\Bigg)^{\kappa} \Big(R_{adj}\Big)^{\lambda}$$

where κ and λ reflect the relative weights assigned to the two portions.

IV. Results

Fig. 2. a) Complete setup and, b) close up of a deployed photocell-based sensor constructed out of a flashlight and a Crossbow wireless sensor. (Plastic containers were used as protective covering during experiments.)

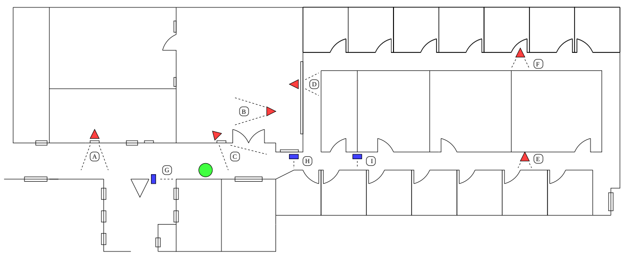

Fig. 3. The layout of the nine senor (heterogeneous) network used for the experiment. Labeled triangles represent vision-based sensor positions (A-F) and labeled rectangles represent low-powered photo-based sensors (G-I). The circle represents the location of the central server.

Results from numerical simulations verified the feasibility our approach. A simulator modeling the problem was constructed, and the technique was tested on hundreds of random networks of different sizes and under a number of different conditions. The technique demonstrated a high degree of accuracy and was both robust to noise and to complex traffic patterns. It appeared that the results obtained by our method compared favorably to related work by Ellis *et al.* [6], [7], although their approach was much less computationally intensive.

Our approach was then further examined with experiments carried out using a heterogeneous sensor network. The network was constructed using two types of sensors: vision-based sensors using PC hardware and webcams, and photocell-based sensors using low-powered MICA2 devices (Figure 2).

Data collected under these real world conditions varied considerably from data generated by the simulator. The imperfect, hardware implemented, sensors were occasionally subject to both missing and spurious observations. These errors often occurred in an unpredictable manner. Additionally, the patterns of motion through the environment were complex and did not consist of only ‘through traffic’. However, the performance of our technique on the experimental data was satisfying. The inferred results closely matched analytically determined ‘ground truth’ values and were consistent with empirical assessments (Figure 3, Figure 4).

V. Conclusion

We have presented a method for inferring the topology of a sensor network given non-discriminating observations of activity in the monitored region. Our technique recovers the network connectivity information opportunistically through the exploitation of existing motion. This task is accomplished based on no prior knowledge of the relative locations of the sensors and only a limited knowledge of the type of activity present in the environment.

Results from both simulations and experiments have shown the ability of our algorithm to generate accurate results under conditions of sensor noise and complex traffic patterns. The technique compares favorably to related approaches and could have promising real world applications in the area of sensor network calibration and self-configuration.

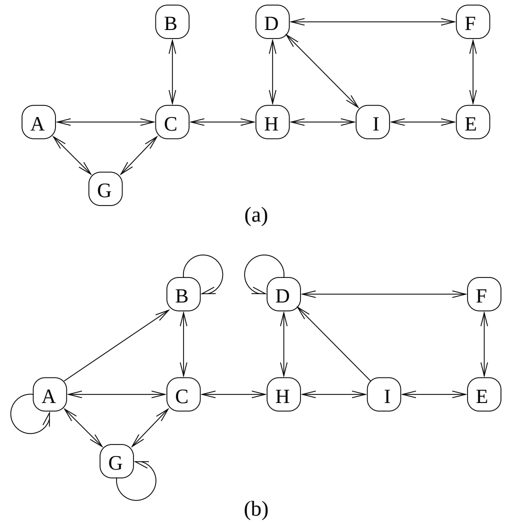

Fig. 4. Topological maps of the environment that were: a) analytically determined based on the layout; b) inferred by the algorithm;

References

[1] N. Bulusu, D. Estrin, L. Girod, and J. Heidemann, “Scalable coordination for wireless sensor networks: self-configuring localization systems,” in *Sixth International Symposium on Communication Theory and Applications (ISCTA-01)*, Ambleside, Lake District, UK, July 2001.

[2] N. Correal and N. Patwari, “Wireless sensor networks: Challenges and opportunities,” in *MPRG/Virgina Tech Wireless Symposium*, 2001.

[3] D. Moore, J. Leonard, D. Rus, and S. Teller, “Robust distributed network localization with noisy range measurements,” in *Proc. of the Second ACM Conference on Embedded Networked Sensor Systems (SenSys ’04)*, Baltimore, November 2004.

[4] D. Niculescu and B. Nath, “Ad hoc positioning system (APS) using AoA,” in *Proc. of INFOCOM*, San Francisco, CA., 2003.

[5] O. Javed, Z. Rasheed, K. Shafique, and M. Shan, “Tracking across multiple cameras with disjoint views,” in *The Ninth IEEE International Conference on Computer Vision*, Nice, France, 2003.

[6] D. Makris, T. Ellis, and J. Black, “Bridging the gaps between cameras,” in *IEEE Conference on Computer Vision and Pattern Recognition CVPR 2004*, Washington DC, June 2004.

[7] T. Ellis, D. Makris, and J. Black, “Learning a multicamera topology,” in *Joint IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance*, Nice, France, October 2003, pp. 165–171.

[8] G. Wei and M. Tanner, “A monte-carlo implementation of the EM algorithm and the poor man’s data augmentation algorithms,” *Journal of the American Statistical Association*, vol. 85(411), pp. 699–704, 1990.

[9] T. M. Michell, *Machine Learning*. Boston: McGraw-Hill, 1997.