RATIONAL SWARMS FOR DISTRIBUTED ON-LINE BAYESIAN SEARCH

Alfredo Garcia, Chenyang Li and Fernan Pedraza

University of Virginia

In this paper, we are interested in peer-to-peer network applications involving the search, identification, and location of targets across a search area that is often vast and possibly dangerous to navigate. Mobile robots and/or sensors employed for such applications provide promise to more rapidly and safely locate targets in different military, homeland security and/or disaster recovery scenarios. Advantages of these type of applications include reduced risk for human search and/or rescue teams and significantly enhanced search capabilities.

Specifically, we propose a novel scheme for distributed search in mobile sensors networks that is inspired by collective forms of intelligence present in many biological systems typically referred to as "swarm intelligence". *Unlike* the established paradigms of swarm intelligence, we posit a form of individual rationality governing each agent's decision. Hence the term "rational" swarm. Under the proposed scheme a network of N mobile sensors is tasked to find N targets. The sensing technology is imperfect so there are non-negligible probabilities for false positives and false negatives. Mobile sensors leave two 'trails' across potential target locations that have been explored. One trail is associated with the frequency with which a given location x (in a grid X) has been probed, say $\lambda^t(x)$, while the other relates to the Bayes updated likelihood that a target is present, say $\mu^t(x)$. These trails are reminiscent of the pheromone trail used by ant colonies to find the shortest path between their nest and a food source. *Unlike* the established paradigms of swarm intelligence, agents process the implicit information encapsulated in the two trails and choose a decision that is aimed at maximizing the chance of detecting a target without unnecessary duplication in probing, i.e. if s_i^t is agent i's current location, the next location s_i^{t+1} is defined as follows

$$s_i^{t+1} \in \arg \max_{x \in N(s_i^t)} [\mu^t(x)(\lambda_i^t(x) - \lambda_{-i}^t(x))]$$

where $N(s_i^t) \subset X$ is the set of reachable locations from s_i^t in one time period and $\lambda_i^t(x)$ and $\lambda_{-i}^t(x)$ are measures of probing frequency in location x by agent i and agents other than i, respectively. The distributed feedback loop is illustrated in Figure 1. By endowing mobile sensors with this simple optimization rule, we show that a form of 'rational swarm' intelligence emerges as sensors successfully coordinate indirectly (i.e. they locate all targets) through active manipulation of the trails. Specifically, we show that for every $x \in X^*$ (where X^* is the set of target locations), there exists an agent $i \in \{1, 2, ..., N\}$ such that

$$\lim_{t \to \infty} [\mu^t(x)(\lambda_i^t(x) - \lambda_{-i}^t(x))] = 1 \quad \text{w.p. } 1$$

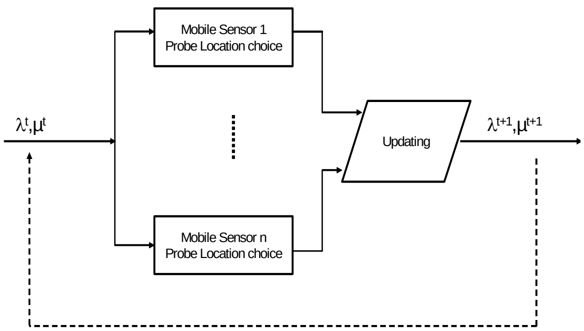

Fig. 1. Schematic for Distributed Search

This feature guarantees the proposed scheme is both *reconfigurable* and *scalable*. Reconfigurability follows from the fact that agents only need to know how often a given location has been probed in the past (regardless of the identity of the sensor(s) that executed the probes) and the updated Bayesian probabilities. Thus, sensors do not need to know the makeup of the group so new sensors can enter the network and others can exit. Scalability follows from the fact that bilateral communication amongst sensors is not required. Instead, agents must be able to access the values of the two trails. This can be achieved by having a geographically distributed array of stationary motes in charge of keeping track of $\lambda^t(x)$ and $\mu^t(x)$, for locations $x \in X$.

In a simulation testbed, we compare the performance of our distributed search algorithm with a *centralized* search scheme, where for each iteration t, there is a "virtual" base station that instructs each agent on the best next location to probe. Note that such in such a centralized scheme, unnecessary duplication in search efforts is avoided. We tested the two algorithms with a total of $5, 10, 20$ targets, $6, 12, 24$ mobile agents, and area size 5^2, 10^2, 20^2 respectively. For each scenario, we repeated the experiment 200 times with randomized targets locations each time. The results are shown in Table 1.

Mean (Std Dev)	# Iterations to Full Detection	
# Targets	Distributed	Centralized
5	19.33 (8.87)	12.95 (6.61)
10	48.71 (22.60)	45.56 (18.82)
20	138.20 (47.59)	132.15 (49.13)

Table 1: Comparison

This evidence suggests the performance of our distributed search scheme is very close to the centralized scheme and it improves when the scale is up.

Finally, we developed a simple physical testbed consisting of four agents, and four targets distributed in an 8 by 10 feet square searching field. A virtual grid was generated to divide the field into 1 foot square cells, where the individual sensing took place. The testbed can be described as the integration of the following main components:

Agents A set of four Lego Mindstorm NXT robots using a three-wheel configuration were used as mobile platforms.

Positioning System An elevated webcam together with onboard LED's were used to determine agent positions and facing direction within the searching field.

Target Detection A light sensor pointing downwards was placed in each agent to detect darker sectors (targets) that contrasted with the white field.

Measurement Error False positive and false negative outputs were introduced within each agent posterior target sensing.

Data Handling Bluetooth links were established between a data repository (Laptop) and each agent independently to share the desired information.

Restricted Movement Agents are only allow to move to non-diagonal adjacent cells.

A movie describing a sample path with this physical testbed can be downloaded at

people.virginia.edu/~ag7s/papers/Lego_video.avi