Distributed Information-Theoretic Target Detection Using Physics-Inspired Motion Coordination

Nitin Sydney
Department of Aerospace Engineering
University of Maryland
College Park, Maryland
Email: nitin.sydney.ctr@nrl.navy.mil

Donald Sofge
Naval Research Laboratory
Washington D.C., USA
Email: donald.sofge@nrl.navy.mil

***Abstract*—In this paper we describe a distributed, information theoretic motion planning strategy for multi-agent target detection. Agents assimilate measurements into a likelihood-ratio tracker, which provides a probability distribution for potential target locations. Information from other agents is fused with the local agents' probability density using an Information Weighted Consensus Filter when in communication range. Each agent uses a physics-inspired motion planning strategy to reactively cover the domain and informatively gather measurements based on the posterior of the likelihood ratio tracker. The proposed strategy produces emergent behavior that optimally collects information about the environment in a reactive and scalable manner that is resilient to communication drop outs. The algorithm is tested in simulation to verify the performance.**

I. Introduction

With the advent of low cost autonomous vehicles, there has been an increase in research devoted to algorithms for multi-agent cooperation to accomplish coverage and mapping tasks. One specific problem in this domain of research is the task of cooperative target detection. In an ideal target detection mission, all agents would communicate with each other to coordinate their motion in an informative manner. However, in practice, agents must deal with unavoidable environmental factors such as communication loss, time delays, and sensor noise. In this paper we design a distributed cooperative target detection algorithm that is resilient to communication loss between agents.

There are many approaches to multiple target tracking in the literature. Probabilistic approaches such as [1]–[4] use Bayesian inference combined with maximum likelihood or multiple hypothesis trackers to track multiple (moving) targets. Although the above approaches address how to combine sensor measurements to track and detect targets, only a few techniques ([5], [6]) effectively move the mobile platforms to find targets and most of these techniques focus on the use of a single sensor platform. Recently, several authors, e.g., [7], [8], have used information-based approaches to design optimal sensor placement/control, but it is difficult to scale these approaches to a large number of vehicles due to the complexity in calculating mutual information. Most similar to the work presented here are physics-inspired swarm controllers ([9]–[11]), where vehicles switch behaviors based on their "temperature" to improve the speed and quality of information gathering. However, one thing that all of these approaches have in common is that they are not distributed.

In this paper, we describe a potential-based motion-planning strategy for multiple mobile sensor platforms to collaboratively search for multiple mobile targets using visual sensors with a finite field of view. The sensors give a binary measurement of target presence, i.e., the measurement is one if a target is within the field of view and zero otherwise. Sensor measurement data from an agent are assimilated by a Bayesian likelihood ratio tracker (LRT) that uses a recursive formulation to produce a probability density function over the set of possible target positions. The probability density functions are distributively combined using an Information Weighted Consensus Filter (ICF) [12] with other agents that are in range. Since each agent has its own estimate of the probability density, they can operate on their own, but can also take advantage of information collected by other agents when they are in communication range. The rest of the paper is organized as follows. Section II reviews the Likelihood Ratio Tracker and our sensor and motion models. Section III discusses how the separate probability density functions are fused using a consensus filter. Section IV describes a physics inspired motion planner for distributed cooperative target detection.

II. Likelihood Ratio Tracker

This section reviews the likelihood ratio tracker [3] and formulates sensor and motion models for the vehicles and targets, respectively. A likelihood ratio tracker is used to estimate the position of possibly multiple targets. This framework is sometimes called "track-before-detect" [3], because it is based on recursive Bayesian estimation. A Bayes filter is a probabilistic approach for assimilating noisy measurements into a probability density function over the target state space, in this case two-dimensional position. The filter implementation comprises the discrete steps of predicting and updating. Let $\theta_k = (x_k, y_k) \in \Omega$ denote the target position (state) at time step k and z_k denote the measurement at time k. The predict step involves computing the conditional probability [3]

$$p(\theta_k|z_{k-1}) = \int_\Omega p(\theta_k|\theta_{k-1})p(\theta_{k-1}|z_{k-1})d\theta_{k-1}. \quad (1)$$

The measurement update is proportional to the product of the measurement likelihood $p(z_k|\theta_k)$ and the predicted state [3]:

$$p(\theta_k|z_k) = \frac{p(z_k|\theta_k)p(\theta_k|z_{k-1})}{p(z_k|z_{k-1})}, \quad (2)$$

In a likelihood ratio tracker we replace the measurement likelihood with the measurement likelihood ratio, i.e., the ratio of two likelihood functions. The numerator of the likelihood ratio represents the conditional probability of the measurement given that the target is in state θ_k^+, whereas the denominator represents the conditional probability of the measurement given that the target is not in state θ_k^-. Thus, the log likelihood ratio is

$$\log \mathcal{L}(z_k|\theta_k) = \log \frac{p(z_k|\theta_k^+)}{p(z_k|\theta_k^-)}. \quad (3)$$

Let $P = \log(p)$. The update step in the log likelihood ratio tracker becomes

$$P(\theta_k|z_k) = \log \frac{\mathcal{L}(z_k|\theta_k)p(\theta_k|z_{k-1})}{p(z_k|z_{k-1})}.$$

When the target probability reaches a critical threshold, the target is declared detected; otherwise, the sub-threshold target probabilities are maintained as hypotheses. Note, the inverse log likelihood ratio posterior represented by temperature is $-P(\theta_k|z_k)$.

A. Predict step: Integrating the diffusion equation

The predict step (2) involves updating the target probability density function in the absence of measurement information. We allow targets to move randomly, though we do not estimate their velocities. The target motion model is a random walk which can be described mathematically by the diffusion equation with constant diffusivity, also called the heat equation.

Consider the two-dimensional heat equation with diffusivity α,

$$\frac{\partial P}{\partial t} = \alpha \left(\frac{\partial^2 P}{\partial x^2} + \frac{\partial^2 P}{\partial y^2} \right).$$

The alternating direction implicit (ADI) method [13] is a fractional-step method for integrating the heat equation. ADI employs the difference operators A_x and A_y representing the spatial derivatives in x and y directions, respectively. To ensure that no information crosses the boundary of the search domain, we enforce Neumann boundary conditions around the perimeter of Ω by specifying that the normal gradient is zero.

B. Update step: The sensor measurement model

Consider a measurement data model based on an imperfect binary sensor. Suppose the criterion for a positive response is chosen such that targets within range ρ_{target} are detected with probability P_d and false alarms occur with probability P_f per time step. The sensitivity m of each sensor is $m = z(P_d) - z(P_f)$ [14], where $z(\cdot)$ represents the z-transformation into standard deviation units given by the quantile function $z(p) = \sqrt{2}\text{erf}^{-1}(2p-1)$. For example, $P_d = 0.95$ and $P_f = 0.1$ yields $m = 2.92$. Let w_k represent unit-normal measurement noise in standard deviation units at time step k. When the target is absent, the measurement data is $z_k = w_k$, whereas when the target is present the measurement data is $z_k = m + w_k$. Assuming a zero-mean Gaussian sensor model yields

$$p(z_k|\theta_k^-) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{z_k^2}{2}\right) \quad (4)$$

$$p(z_k|\theta_k^+) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{(z_k - m)^2}{2}\right). \quad (5)$$

III. Distributed Information Gathering

In this section we use an Information-Weighted Consensus Filter (ICF), developed in [12], to combine the likelihood-ratio surface produced by each agent in a distributed manner. At its core the ICF is a Kalman-like filter with an embedded consensus filter. In its implementation the ICF discretizes the field over which consensus is being performed, which in this case is the LR surface.

Each agent fuses its own measurements into an LRT to create $P^j(\theta_k, z_k)$, i.e., the likelihood ratio surface for agent j, where $j \in \{1, \ldots, N\}$. When in communication range with another agent, the agent transmits both its likelihood ratio surface as well as a confidence surface $C^j(\theta_k)$. In the IFC $C^j(\theta_k)$ is the inverse of the covariance matrix produced by a Kalman filter that each agent runs to "estimate" the LR surface.

The ICF has three inputs. The first is $C^j(\theta_k)$, the second is $P^j(\theta_k, z_k)$, and the third is the observation matrix $H^j(t_k)$ defined as follows. Let $H^j(t_k)$ be a diagonal, $M \times M$ matrix with M being the number of discretized grid points at which the LR surface is evaluated. The diagonal elements are defined as

$$H^j(t_k)_{ii} = \begin{cases} 1, & \text{if } \theta_k \in \rho_{target} \\ 0, & \text{otherwise.} \end{cases} \quad (6)$$

Furthermore, let $y^j(t_k) = H^j(t_k)$ be a measurement of the LR surface. Equation (6) simply picks out the values of the LR surface that are within radius ρ_{target} of agent j.

The ICF is summarized in Algorithm 1 (from [12]). In the algorithms, the parameters r and q are tuning parameters that are used to adjust how fast the measurement (step 2) and prediction (step 5) updates converge (respectively). The main consensus step is step 3. In this step the agent communicates and receives the information matrix and field estimate of all agents within communication range $\mathcal{N}_j$. The difference between this particular consensus filter and other similar filters is that it weights the information content from each agent in the estimation step (step 5).

Note that using (6) as the observation matrix is crucial. This can be seen in step 2 of Algorithm 1; the observation matrix will add to the information only in the area surrounding the agent, where the parameter r is used to adjust the amount of added information. In other areas the information decreases through the prediction step (step 5). The parameter q in step 5 dictates how fast the information decreases.

IV. Physics-Inspired Motion Planning

The agent motion is driven by a combination of two artificial forces. The first force guides the vehicle down the gradient of the inverse log likelihood surface, i.e., the temperature surface, in a manner that resembles a flowing liquid. The speed of movement down the gradient is determined by the temperature at the vehicle location, where colder temperatures correspond to slower speeds. The second force is the gradient of a Lennard-Jones potential [15] between the agents. The Lennard-Jones potential is a common approximation used in physical chemistry to model molecular gas dynamics. Figure 1 shows the vehicles moving like gas molecules in areas where there are no targets and gathering like solid molecules with other agents where there are likely to be targets. In

Algorithm 1 ICF at grid point a at time step k

Require: prior LR surface estimate $P^j(\theta_k, z_k)$, prior information matrix $C^j(\theta_k)$, and observation matrix $H^j(t_k)$

1) Get the measurement $y^j(t_k)$

2) Compute the consensus proposals

$W_0 = \frac{1}{M} C^j(\theta_k) + \frac{1}{r} H^j(t_k)^T H^j(t_k)$

$w_0 = \frac{1}{M} P_i^j + \frac{1}{r} H^j(t_k) y^j(t_k)$

3) Perform average consensus on W_0 and w_0 independently

for l = 1 to L **do**

a) Send W_{l-1}^j and w_{l-1}^j to all neighbors $i \in \mathcal{N}_j$

b) Receive W_{l-1}^i and w_{l-1}^i to all neighbors $i \in \mathcal{N}_j$

c) Update:

$W_l^j = W_{l-1}^j + \epsilon \sum_{i \in \mathcal{N}_j} (W_{l-1}^i - W_{l-1}^j)$

$w_l^j = w_{l-1}^j + \epsilon \sum_{i \in \mathcal{N}_j} (w_{l-1}^i - w_{l-1}^j)$

end for

4) Compute *a poteriori* state estimate and information matrix for time k

$P^j(\theta_k, z_k) = (W_L^j)^{-1} w_L^j$

$C^j(\theta_k) = N W_L$

5) Predict for the next time step

$C^j(\theta_{k+1}) = ((C^j(\theta_{k+1})^{-1} + q)^{-1}$

this algorithm, as opposed to other physics-inspired swarm controllers, the states-of-matter behavior is emergent.

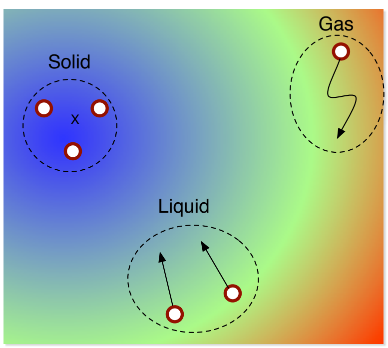


Fig. 1. Diagram depicting the three emergent behaviors of the proposed algorithm. Vehicles in cold areas that are near a target will form a crystalline formation with nearby agents. Vehicles that are in warm areas will flow on the temperature surface like a liquid. Agents in hot areas will travel quickly in a random, gas-like fashion.

Let r_k^j denote the position of agent $j \in \{1, \ldots, N\}$ at time step k. The Lennard-Jones potential is [15]

$$V_k^j = \sum_{i \neq j}^{N} 4\epsilon \left(\sigma^{12} ||r_k^j - r_k^i||^{-12} - \sigma^6 ||r_k^j - r_k^i||^{-6} \right), \quad (7)$$

where ϵ is the depth of the well and σ is the distance at which the potential between two agents is zero. Note that the strength of the potential drops quickly to zero for large distances. Thus, the interaction between two agents only happens when they are close to one another.

Let $T_k^j = -P(\theta_k | z_k, j)$ in (4) be the temperature at vehicle j at time k and V_k^j be the Lennard-Jones potential (7) for the jth vehicle. We prescribe the desired velocity for each vehicle by prescribing a feedback controller. The desired velocity for the jth vehicle is given by

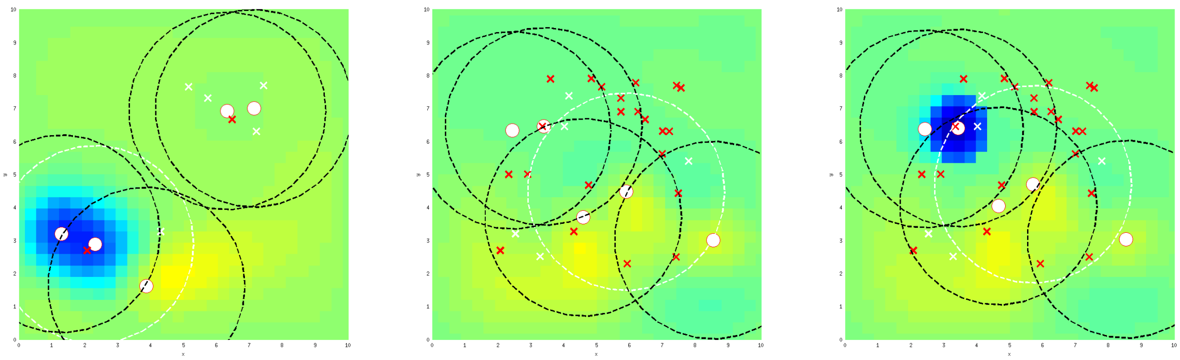
$$\dot{r}_k^j = -\min(1 + e^{T_k^j}, v_{max}) k_P \nabla T_k^j - k_V \nabla V_k^j, \quad (8)$$

where k_P and k_V are control gains and v_{max} is the maximum speed of the agent (assumed to be the same for all agents). The dependence on temperature causes the vehicles to slow down when they are near a possible target in order to collect more measurements in that area. This behavior, along with the sensor aggregation caused by the Lennard-Jones potential, seeks to increase the chances that a target will be detected. The algorithm works well in simulations when the gain on the Lennard-Jones potential is one order of magnitude smaller than the gradient-following gain. Additionally, σ is set to ensure that the vehicles do not collide with each other, which depends on the scale of the vehicle. Consequently, there are only three parameters to tune: either k_P or k_V, ϵ (the well depth), and T_0 (the detection threshold).

As seen in equation (8), we multiply the gradient of the temperature potential by a heuristic scaling factor. This scaling factor, in combination with the Lennard-Jones potential, is what causes the states-of-matter behavior to be emergent. In the scaling factor, the speed of the vehicle is scaled by the temperature such that the speed increases when the temperature is high, and decreases when the temperature is slow. This causes vehicles to “freeze” in place when they are near an area of high probability and, conversely, to speed up when they are in areas of low probability. The full architecture (in pseudocode form) is shown in Algorithm 2. The architecture shown in Algorithm 2 avoids local minima in an informative, non-heuristic fashion. A vehicle can fall into a local minima if it is surrounded by hills of low target probability. However, if there is no target present underneath the vehicle, the temperature will increase (from negative information) to the point where the vehicle is pushed away from its location. Note, this behavior is completely emergent and is provided inherently by connecting the detection scheme to the vehicle control. In addition to informatively handling local minima, we have the following theorem.

Proposition 1: The mutual information between sensor measurement z_k and target location θ_k using the sensor model given by (4)–(5) is locally maximized when the vehicle moves along the gradient of the prior target distribution $p(\theta_k)$.

Figure 2 shows three snapshots of a simulation of the full system. The small white circles represent the agents, the white X’s are undetected targets, and the red Xs are detected targets. The large dashed circles indicate the communication range of each agent. The white dashed circle is the agent whose temperature surface is visible as the colormap (who we will call agent 1). In Figure 2(a) we see that the three agents in the lower left are all communicating with each other and therefore the temperature surface for agent 1 has fused information from both of those sources. However, it is not in range of the two top right agents, and hence has no knowledge of their actions. Figures 2(b) and 2(c) show how quickly information can be propagated through the system. Between time steps 290 and 291, agent 1 comes into contact with an agent that is near a target. In the course of one time step, agent 1 runs the ICF to



(a) Time step $k = 20$ (b) Time step $k = 290$ (c) Time step $k = 291$

Fig. 2. Snapshots of the simulation at time steps (a) 20, (b) 290, and (c) 291

Algorithm 2 Physics-Inspired Target Search and Detection Algorithm

Require: $\mathrm{N}, \epsilon, \sigma,\ k_P,\ k_V$
Initialize log likelihood ratio $P_0(\theta|z)$
for $k = 1$ till end of task **do**
    for $j \ldots N$ **do**
        Get measurement z_k^j
        Calculate likelihood ratio $\log \mathcal{L}(z_k^j|\theta_k)$ using (3)
        Update: $P_k(\theta|z) = P_{k-1} + \log \mathcal{L}(z_k^j|\theta_k)$
    end for
    Normalize likelihood ratio: $P_k(\theta|z) = P_k(\theta|z) - \mathrm{sum}((P_k(\theta|z)))$
    Get r_k for every vehicle
    for $j = 1 \ldots N$ **do**
        Compute $\nabla V_k^j(r_k)$ and $\nabla P(\theta_k|z_k)$
        Calculate desired velocity $\dot{r}_k^j$
        Apply control input to platform to achieve $\dot{r}_k^j$
    end for
end for

update its temperature surface based on the agent that comes within communication range.

V. Conclusion

In this paper, we discuss a physics-inspired, distributed motion planning strategy for target detection using agents with limited communication range. The proposed strategy collects information in an optimal manner using a reactive planner for which all behavior is emergent. We use a likelihood ratio tracker to fuse binary measurements into an estimate of target locations. Agents run an information consensus filter to fuse their probability distributions together using and information weighted consensus filter. Incorporating a range-based communication architecture ensures that the proposed strategy is resilient to communication drop outs. In ongoing work, we are working towards experimentally testing the algorithm on an aerial testbed.

Acknowledgments

This work was performed at the Naval Research Laboratory and was funded by the Office of Naval Research under grant number N0001413WX21045, "Mobile Autonomous Navy Teams for Information Surveillance and Search (MANTISS)"

References

[1] W. Blanding, P. Willett, and Y. Bar-Shalom, "Multiple target tracking using maximum likelihood probabilistic data association," in *2007 IEEE Aerospace Conference*, March 2007, pp. 1–12.

[2] P. Willett, Y. Ruan, and R. Streit, "Pmht: problems and some solutions," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 38, no. 3, pp. 738–754, Jul 2002.

[3] L. D. Stone, C. A. Barlow, and T. L. Corwin, *Bayesian Multiple Target Tracking*. Artech House, 1999.

[4] D. Clark and J. Bell, "Bayesian multiple target tracking in forward scan sonar images using the phd filter," *IEEE Proceedings of Radar, Sonar and Navigation*, vol. 152, no. 5, pp. 327–334, October 2005.

[5] F. Rafi, S. Khan, K. Shafiq, and M. Shah, "Autonomous target following by unmanned aerial vehicles," vol. 6230, jun 2006.

[6] J. Kim and Y. Kim, "Moving ground target tracking in dense obstacle areas using uavs," in *Proceedings of the 17th IFAC World Congress*, 2008.

[7] G. M. Hoffmann and C. Tomlin, "Mobile sensor network control using mutual information and particle filters," *Automatic Control, IEEE Transactions on*, vol. 55, no. 1, pp. 32–47, Jan 2010.

[8] C. M. Kreucher, K. D. Kastella, and A. O. Hero III, "Information-based sensor management for multitarget tracking," in *Optical Science and Technology, SPIE's 48th Annual Meeting*, 2003, pp. 480–489.

[9] E. Cuevas, A. Echavarria, and M. A. Ramirez-Ortegon, "An optimization algorithm inspired by the states of matter that improves the balance between exploration and exploitation," *Applied Intelligence*, vol. 40, no. 2, pp. 256–272, 2014.

[10] T. B. Apker and M. A. Potter, "Robotic swarms as solids, liquids and gasses." in *AAAI Fall Symposium: Human Control of Bioinspired Swarms*, 2012.

[11] W. M. Spears, D. F. Spears, J. C. Hamann, and R. Heil, "Distributed, physics-based control of swarms of vehicles," *Autonomous Robots*, vol. 17, no. 2-3, pp. 137–162, 2004.

[12] A. T. Kamal, J. A. Farrell, and A. K. Roy-Chowdhury, "Information weighted consensus filters and their application in distributed camera networks," *Automatic Control, IEEE Transactions on*, vol. 58, no. 12, pp. 3112–3125, Dec 2013.

[13] N. Moin, *Fundamentals of Engineering Numerical Analysis*, 2nd ed. Cambridge, 2010.

[14] N. A. Macmillan and C. D. Creelman, *Detection Theory: A User's Guide*, 2nd ed. Psychology Press, 2008.

[15] J. Lennard-Jones, "On the determination of molecular fields," *Proc. R. Soc. Lond.*, vol. 106, no. 738, pp. 463–477, 1924.