Multi-platform Information-Based Sensor Management

Chris M. Kreucher[a], Keith D. Kastella[a], and Alfred O. Hero III[b]

[a]General Dynamics Michigan Research & Development Facility, 1200 Joe Hall Drive, Ypsilanti MI 48197, USA

[b]The University of Michigan, Department of Electrical Engineering and Computer Science, 1301 Beal Avenue, Ann Arbor MI 48109, USA

ABSTRACT

This paper shows how information-directed diffusion can be used to manage the trajectories of hundreds of smart mobile sensors. This is an artificial physics method in which the sensors move stochastically in response to an information gradient and artificial inter-sensor forces that serve to coordinate their actions.

Measurements received by the sensors are centrally fused using a particle filter to estimate the Joint Multi-target Probability Density (JMPD) for the surveillance volume. The JMPD is used to construct an information surface which gives the expected gain for sensor dwells as a function of position. The updated sensor position is obtained by moving it in response to artificial forces derived from the information surface, which acts as a potential, and inter-sensor forces derived from a Lennard-Jones-like potential. The combination of information gradient and inter-sensor forces work to move the sensors to areas of high information gain while simultaneously ensuring sufficient spacing between the sensors. We evaluate the performance of this approach using a simulation study for an idealized Micro Air Vehicle with a simple EO detector and collected target trajectories. We find that this method provides a factor of 5 to 10 improvement in performance when compared to random uncoordinated search.

Keywords: sensor management, artificial physics, swarms

1. INTRODUCTION

The problem of sensor management is to determine how best to direct a collection of sensors to support some objective. If the sensors are mobile then sensor path planning is also part of the problem. Many types of sensors have multiple modes that must also be optimized. For example, a zoomable imaging system can trade resolution for area coverage. In addition to possessing many degrees of freedom to control, sensor management also has the feature that it is often difficult to determine just what is to be optimized. In a typical intelligence, surveillance, and reconnaissance (ISR) application one simultaneously seeks to optimize detection, tracking and identification performance against all targets within a certain region of interest. It can be difficult to determine in a non-ad hoc way how to properly balance the requirements of different sub-objectives of a mission.

With the continuing advance of processor, memory, communications, power and sensing technology it increasingly becomes feasible to deploy hundreds, thousands or hundreds of thousands of small mobile sensors such as micro-air vehicles (MAVs) or small uninhabited underwater vehicles (UUVs). To be deployed in large numbers these systems must be nearly completely autonomous to avoid overwhelming the system operators. Therefore an automated sensor management is required that can handle situations of this sort.

Recently, significant progress on the scheduling problem has been made using information theoretic methods.[1–4] This approach begins with a probability density describing the degrees of freedom of the problem at

Further author information: Send correspondence to Chris Kreucher, E-mail: Christopher.Kreucher@gd-ais.com, Telephone: 734 480 5203. This work was supported by the USAF Contract No. F33615-02-C-1199, AFRL contract SPO900-96-D-0080, and ARO-DARPA MURI Grant DAAD19-02-1-0262. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the United States Air Force.

Defense Transformation and Network-Centric Systems, edited by Raja Suresh,
Proceedings of SPIE Vol. 5820 (SPIE, Bellingham, WA, 2005)
0277-786X/05/$15 · doi: 10.1117/12.608436

hand. To capture the degrees of freedom in our problem, we use a Joint Multitarget Probability Density (JMPD) for the number of targets, their kinematic state (position and velocity) and their type. The JMPD is efficiently approximated using a novel particle filtering strategy. Given the JMPD, the expected information gain for a set of alternative sensing actions is measured using the Rényi information divergence and computed using conventional Bayesian techniques. The optimum action is then selected and applied, the JMPD is updated with the measurement outcome, and the process repeats itself.

The information based sensor management approach works well for managing a small numbers of sensors but the picture changes when the number of sensors becomes large. While the JMPD computation is unaffected when a large number of sensor inputs are available (some modification is necessary if inter-sensor communication bandwidth is limited), the number of possible distinct sensing actions is exponential in the number of sensors. That is, if each sensor of the N sensors can take one of M actions, then the total number of joint actions is M^N. It is computationally intractable to obtain the globally optimal set of actions except in very special situations where some sort of convexity or factorization condition holds.

A number of methods have been developed within the robotics community to control large numbers of agents such as the hypothetical mobile sensors described above. A widely used and successful approach is based on a set of techniques referred to variously as "artificial physics", "physicomimetics", "virtual force methods", and "potential field methods". Along these lines numerous researchers have proposed and demonstrated the use of artificial physics concepts to guide both single robots and multi-robot groups. For example, Borenstein[5] describes a goal-seeking system in which a virtual force field is used to drive a single robot toward a desired location while mapping and avoiding obstacles detected by an on-board ultrasonic sensor. More recently these ideas have been extended to large numbers of mobile platforms to produce swarming behavior by Spears.[6] In these applications the emphasis is on constructing interactions amongst the robots so that they behave in a coordinated fashion. The underlying theme in these different demonstrations is that the agents have some form of information regarding the distance and direction to goal states as well as distances and directions to other nearby agents and obstructions. Artificial forces on each agent are then computed to drive the system towards a desired overall state. The agents are then commanded to respond to the computed artificial forces.

While the artificial force approach does not guarantee that the resulting policy will satisfy any global optimality criterion, it does have provide a number of useful features. Most importantly, the resulting sensor management strategy can be implemented so that each agent only needs access to local information. As a result, the method scales well with the number of agents (linearly if only local short-range forces are used).

This paper addresses the problem of Multi-platform sensor scheduling using an approach that combines physicomimetics and information theory. There are three elements to the information driven artificial physics implementation discussed here. First, there is the coupling to the external field designed to support the overall system goal. Here that role is played by the information gain field. The second element of the design is coupling between agents. This force works to prevent the agents from moving en masse towards the same goal state of the external field. The third element of the artificial physics method adopted here is the inclusion of a random diffusive component to the agent motion. This promotes exploration of the local neighborhood by the agents and appears to be useful for sensing applications.

This paper proceeds as follows. Section 2 describes Bayesian multitarget tracking using recursive estimation of the joint multitarget probability density (JMPD). The JMPD is constructed using kinematic and sensor models along with the collected measurements to capture the uncertainty about the surveillance area (here the uncertainty includes both the number of targets and states of the individual targets). Second, in Section 3, we show how we use the JMPD and associated models to determine which portions of the surveillance region are expected to yield the most gain in information if interrogated at the next time epoch. This method relies on computing expectations of information gained for each possible sensor positioning, where information gain is measured by the Rényi (alpha) Divergence. Readers familiar with information-based sensor management may wish to begin with the Section 4 which describes how we use expected information gain in a physicomimetic approach. The combined approach is used to drive the sensor positions over time, resulting in an easy-to-compute and robust method of sensor state evolution. In Section 5, we give simulation results showing the performance obtained with and without including the information surface in the physicomimetic approach. Finally, we give some remarks about performance and conclusions in Section 6.

2. THE JOINT MULTITARGET PROBABILITY DENSITY

In this section, we give the details of a Bayesian method of multitarget tracking predicated on recursive estimation of a probabilistic entity called the Joint Multitarget Probability Density (JMPD). The JMPD and its particle filter implementation are discussed more thoroughly in other works.[7,8] We review only the necessary details here.

Recursive estimation of the JMPD provides a means for tracking an unknown number of targets in a Bayesian setting. The statistical model employed uses the joint multitarget conditional probability density

$$p(\mathbf{x}_1^k, \mathbf{x}_2^k, ...\mathbf{x}_{T-1}^k, \mathbf{x}_T^k, T^k|\mathbf{Z}^k) = p(\mathbf{x}_1^k, \mathbf{x}_2^k, ...\mathbf{x}_{T-1}^k, \mathbf{x}_T^k|T^k, \mathbf{Z}^k)p(T^k|\mathbf{Z}^k) \tag{1}$$

as the probability density for exactly T targets with the states $\mathbf{x}_1, \mathbf{x}_2, ...\mathbf{x}_{T-1}, \mathbf{x}_T$ at time k based on a set of past observations $\mathbf{Z}^k$. We abuse terminology by calling $p(\mathbf{x}_1^k, \mathbf{x}_2^k, ...\mathbf{x}_{T-1}^k, \mathbf{x}_T^k, T^k|\mathbf{Z}^k)$ a density since T is a discrete valued random variable. In fact, as (1) shows, the JMPD is a continuous discrete hybrid as it is a product of the probability mass function $p(T^k|\mathbf{Z}^k)$ and the probability density function $p(\mathbf{x}_1^k, \mathbf{x}_2^k, ...\mathbf{x}_{T-1}^k, \mathbf{x}_T^k|T^k, \mathbf{Z}^k)$.

The number of targets at time k, T^k, is a variable to be estimated simultaneously with the states of the T^k targets. The JMPD is defined for all T^k, $T^k = 0 \cdots \infty$. The observation set $\mathbf{Z}^k$ refers to the collection of measurements up to and including time k, i.e. $\mathbf{Z}^k = \{\mathbf{z}^1, \mathbf{z}^2, ...\mathbf{z}^k\}$, where each of the $\mathbf{z}^i$ may be a single measurement or a vector of measurements made at time i.

Each of the $\mathbf{x}_t$ in the density $p(\mathbf{x}_1^k, \mathbf{x}_2^k, ...\mathbf{x}_{T-1}^k, \mathbf{x}_T^k|T^k, \mathbf{Z}^k)$ is a vector quantity and may (for example) be of the form $[x, \dot{x}, y, \dot{y}]$. We refer to each of the T target state vectors $\mathbf{x}_1, \mathbf{x}_2, ...\mathbf{x}_{T-1}, \mathbf{x}_T$ as a partition of the multitarget state $\mathbf{X}$. For convenience, the density will be written more compactly in the traditional manner as $p(\mathbf{X}^k|T^k, \mathbf{Z}^k)$, which implies that the state-vector $\mathbf{X}^k$ represents a variable number of targets each possessing their own state vector. We will drop the time superscript k for notational simplicity when no confusion will arise.

The likelihoods $p(\mathbf{z}|\mathbf{X}, T)$ and the joint multitarget probability density $p(\mathbf{X}, T|\mathbf{Z})$ are conventional Bayesian objects manipulated by the usual rules of probability and statistics. Thus, a multitarget system has state $\mathbf{X} = (\mathbf{x}_1, \cdots, \mathbf{x}_T)$ with probability distribution $p(\mathbf{x}_1, \cdots, \mathbf{x}_T, T|\mathbf{Z})$. This can be viewed as a hybrid stochastic system where the discrete random variable T governs the dimensionality of $\mathbf{X}$. Therefore the normalization condition that the JMPD must satisfy is

$$\sum_{T=0}^{\infty} \int d\mathbf{x}_1 \cdots d\mathbf{x}_T p(\mathbf{x}_1, \cdots, \mathbf{x}_T, T|\mathbf{Z}) = 1 \ . \tag{2}$$

where the single integral sign is used to denote the T integrations required.

Quantities of interest can be deduced from the JMPD. For example, the probability that there are exactly T targets present in the surveillance area is given by the marginal distribution

$$p(T|\mathbf{Z}) = \int d\mathbf{x}_1 \cdots d\mathbf{x}_T p(\mathbf{x}_1, \cdots, \mathbf{x}_T, T|\mathbf{Z}) \ . \tag{3}$$

The temporal update of the posterior likelihood proceeds according to the usual rules of Bayesian filtering. The model of how the JMPD evolves over time is given by $p(\mathbf{X}^k, T^k|\mathbf{X}^{k-1}, T^{k-1})$ and will be referred to as the kinematic prior (KP). The kinematic prior includes models of target motion, target birth and death, and any additional prior information that may exist such as terrain and roadway maps. In the case where target identification is part of the state being estimated, different kinematic models may be used for different target types. The time-updated (prediction) density is computed via the *model update* equation as

$$p(\mathbf{X}^k, T^k|\mathbf{Z}^{k-1}) = \sum_{T^{k-1}=0}^{\infty} \int_{\mathbf{X}^{k-1}} d\mathbf{X}^{k-1} p(\mathbf{X}^k, T^k|\mathbf{X}^{k-1}, T^{k-1}) p(\mathbf{X}^{k-1}, T^{k-1}|\mathbf{Z}^{k-1}) \ . \tag{4}$$

The *measurement update* equation uses Bayes' rule to update the posterior density with a new measurement $\mathbf{z}^k$ as

$$p(\mathbf{X}^k, T^k|\mathbf{Z}^k) = \frac{p(\mathbf{z}^k|\mathbf{X}^k, T^k) p(\mathbf{X}^k, T^k|\mathbf{Z}^{k-1})}{p(\mathbf{z}^k|\mathbf{Z}^{k-1})} \ . \tag{5}$$

2.1. Kinematic Modeling : The Model $p(\mathbf{X}^k, T^k|\mathbf{X}^{k-1}, T^{k-1})$

The Bayesian framework outlined above requires a model of how the system state evolves, $p(\mathbf{X}^k, T^k|\mathbf{X}^{k-1}, T^{k-1})$. This includes both how the number of targets changes with time (i.e. T^k versus T^{k-1}), what states targets tend to arrive in and depart from, and how individual targets that persist over time evolve (i.e. $\mathbf{x}^k$ versus $\mathbf{x}^{k-1}$). In general, this model is chosen using the physics of the particular system under consideration. The target motions in the simulation studies presented in this paper come from a set of real ground targets recorded during a military battle exercise. Therefore, here we specialize the state models to this application. More general models (or simply different models) are possible and can be implemented similarly if warranted by the physics of the situation.

To specify the state model, we need to generate an expression for how the state of the system evolves, $p(\mathbf{X}^k, T^k|\mathbf{X}^{k-1}, T^{k-1})$, which can be evaluated for any set of multitarget states and target counts.

We first define a set of spatially varying priors on target arrival and departure from the surveillance region. We refer to these two events as target birth and death. Let $\alpha^k(\mathbf{x})$ denote the *a priori* probability that a target will arrive (birth) at location $\mathbf{x}$ at time k. Similarly, let the *a priori* probability that a target in location $\mathbf{x}$ will leave the surveillance region (death) be denoted by $\beta^k(\mathbf{x})$. Target arrival may indicate a target actually passing into the surveillance region (e.g., along the border of the region) or may indicate the target has taken an action that first makes it detectible to the sensor (e.g., motion by a target when the only sensor is a moving target indicator). Target death may indicate a target actually leaving the surveillance region or becoming permanently extinguished. These model parameters specify how target number changes with time.

For targets that persist over a time step (those targets that are present at both time step $k-1$ and k), we model the target motion as linear and independent for each target. Using $\mathbf{x} = (x, \dot{x}, y, \dot{y})$ to denote the state vector of an individual target, the model is

$$\mathbf{x}_i^k = \mathbf{F}\mathbf{x}_i^{k-1} + \mathbf{w}_i^k \ , \tag{6}$$

where

$$\mathbf{F} = \begin{pmatrix} 1 & \tau & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & \tau \\ 0 & 0 & 0 & 1 \end{pmatrix} \ . \tag{7}$$

Here $\mathbf{w}_i^k$ is 0-mean Gaussian noise with covariance $\mathbf{Q} = diag(20, .2, 20, .2)$, which was selected based on an empirical fit to the data. We emphasize here that Linear/Gaussian models are not a requirement of the formulation, but are used as they have been found to perform well in simulation studies with the real data. More complicated models of target motion can be inserted where appropriate without directly effecting computations in the algorithm.

2.2. Sensor Modeling : The Model $p(\mathbf{z}^k|\mathbf{X}^k, T^k)$

The model problem considered in Section 5 consists of M sensors that make detections on a pixelated grid for the purposes of detecting and tracking a group of moving targets.

The measurement process is idealized as follows. The surveillance region is broken into $N_x \times N_y$ contiguous pixels. The x- and y- ground-plane projection of each pixel is Δ_x and Δ_y. The sensor response within pixel i is uniform for targets in pixel i and vanishes for targets outside pixel i. It is convenient to define the occupation number $n_i(\mathbf{X})$ for pixel i as the number of targets in $\mathbf{X}$ that lie in i. The single target signal-noise-ratio (SNR), assumed constant across all targets, is denoted λ. We assume that when multiple targets lie within the same pixel their amplitudes add non-coherently (this will be an accurate model for unresolved optical targets and radar targets not moving as a rigid body). Then the effective SNR when there are n targets in a pixel is $\lambda_n = n\lambda$ and we use $p_n(z_i)$ to denote the pixel measurement distribution (note that the background distribution is obtained by setting $n = 0$).

A sensor hovers above the surveillance region and measures the pixel that is directly below it. Therefore, a scan consists of M returns, and a measurement $\mathbf{z}$ consists of the pixel output vector $\mathbf{z} = [z_1, \ldots, z_M]$, where

z_i is the output of pixel i. In general, z_i can be an integer, real, or complex valued scalar, a vector or even a matrix, depending on the sensor. If the data are thresholded, then each z_i will be either a 0 or 1. Note that for thresholded data, $\mathbf{z}$ consists of both threshold exceedances and non-exceedances. The failure to detect a target at a given location can have as great an impact on the posterior distribution as a detection.

We model both measurements of spatially separated pixels and multiple measurements of the same pixel at as conditionally independent given the state, i.e.,

$$p(\mathbf{z}|\mathbf{X},T) = \prod_i p(z_i|\mathbf{X},T) \ . \tag{8}$$

Let $\chi_i(\mathbf{x}_t)$ denote the indicator function for pixel i, defined as $\chi_i(\mathbf{x}_t) = 1$ when a (single) target in state $\mathbf{x}_t$ projects into sensor pixel i (i.e., couples to pixel i) and $\chi_i(\mathbf{x}_t) = 0$ when the target does not project into sensor pixel i. Observe a pixel can couple to multiple targets and single target can contribute to the output of multiple pixels, say, by coupling through side-lobe responses. The indicator function for the joint multitarget state is constructed as the logical disjunction

$$\chi_i(\mathbf{X},T) = \bigvee_{t=1}^{T} \chi_i(\mathbf{x}_t) \ . \tag{9}$$

The set of pixels that couple to $\mathbf{X}$ is $i_{\mathbf{X}} = \{i|\chi_i(\mathbf{X},T) = 1\}$. For the pixels that do not couple to $\mathbf{X}$, the measurements are characterized by the background distribution, denoted $p_0(z_i)$. With this, (8) can be written as

$$p(\mathbf{z}|\mathbf{X},T) = \prod_{i\in i_{\mathbf{X}}} p(z_i|\mathbf{X},T) \prod_{i\notin i_{\mathbf{X}}} p_0(z_i) \propto \prod_{i\in i_{\mathbf{X}}} \frac{p(z_i|\mathbf{X},T)}{p_0(z_i)} \ . \tag{10}$$

With these modeling assumptions, the measurement distribution in pixel i depends only on its occupation number and (10) becomes

$$p(\mathbf{z}|\mathbf{X},T) \propto \prod_{i\in i_{\mathbf{X}}} \frac{p_{n_i(\mathbf{X}),T}(z_i)}{p_0(z_i)} \ . \tag{11}$$

To complete the specification of the sensor model, we must give its dependence on SNR. Many models are plausible, depending on the detailed nature of the sensor physics. In this work, we have elected to use Rayleigh-distributed measurements. This distribution corresponds to envelope detected signals under a complex Gaussian radar return model, and has been used, for example, to model interfering targets in a monopulse radar system[9, 10] and to model clutter and target returns in turbulent environments.[11] Rayleigh models are also often used for diffuse fading channels. In the pre-thresholded case, this implies

$$p_n(z) = \frac{z}{1+n\lambda} \exp\left(-\frac{z^2}{2(1+n\lambda)}\right) \ . \tag{12}$$

When the tracker only has access only to thresholded measurements, we use a constant false-alarm rate (CFAR) model for the sensor. If the background false alarm rate is set at P_f, then the detection probability when there are n targets in a pixel is

$$P_{d,n} = P_f^{\frac{1}{1+n\lambda}} \ . \tag{13}$$

This extends the usual relation $P_d = P_f^{\frac{1}{1+\lambda}}$ for thresholded Rayleigh random variables at SNR λ.[12]

Even for modest problems, the sample space of the JMPD is enormous as it contains all possible configurations of state vectors $\mathbf{x}_t$ for all possible values of T. Specifically, if the state of an individual target is given by the 4-tuple $\left(x, \dot{x}, y, \dot{y}\right)$, the sample space of the JMPD then contains vectors of length $4N$ for all positive finite N. Thus, to estimate the JMPD in a computationally tractable manner, a more sophisticated approximation method is required.

We use a particle filter to represent the JMPD and approximately implement the time and measurement update equations (4) and (5),

$$p(\mathbf{X}, T|\mathbf{Z}) \approx \sum_{p=1}^{N_{part}} w_p \delta(\mathbf{X} - \mathbf{X}_p) \tag{14}$$

where

$$\delta(\mathbf{X} - \mathbf{X}_p) = \begin{cases} 0 & T \neq T_p \\ \delta_D(\mathbf{X} - \mathbf{X}_p) & \text{otherwise} \end{cases} . \tag{15}$$

In earlier works,[7,8] we describe the novel importance density design that makes this approach tractable. The importance density is designed to recognize when it is permissible to factor the JMPD into a product of smaller densities and also biases particle proposals toward areas of high measurement likelihood.

3. INFORMATION BASED SENSOR MANAGEMENT

In this section, we show how we use information theory to compute the locations in the surveillance region that the sensors should prefer to measure. This is done by computing the expected information gain as a function of sensor location $\mathbf{r}$. This map of important locations will then be used to bias the physicomimetic method as described in Section 4.

We measure the value of a sensing action by the information that is gained by its execution. Since we wish to determine the best sensing action to take before actually executing it, we use the information gain that a sensing action is *expected* to produce.

3.1. The Rényi Divergence

The calculation of information gain between two densities p_1 and p_0 is done using the Rényi information divergence,[13,14] also known as the α-divergence:

$$D_\alpha(p_1||p_0) = \frac{1}{\alpha - 1} ln \int p_1^\alpha(x) p_0^{1-\alpha}(x) dx \tag{16}$$

The α parameter may be used to adjust how heavily one emphasizes the tails of the two distributions p_1 and p_0. In the limiting case of $\alpha \to 1$ the Rényi divergence becomes the commonly utilized Kullback-Leibler (KL) discrimination

$$\lim_{\alpha \to 1} D_\alpha(p_1||p_0) = \int p_0(x) ln \frac{p_0(x)}{p_1(x)} dx \ . \tag{17}$$

If $\alpha = 0.5$, the Rényi information divergence becomes the Hellinger affinity $2ln \int \sqrt{p_1(x)p_0(x)}dx$, which is related to the Hellinger-Battacharya distance squared[15] via

$$D_{Hellinger}(p_1||p_0) = 2\Big(1 - exp\Big(.5 D_{\frac{1}{2}}(p_1||p_0)\Big)\Big) \ . \tag{18}$$

3.2. Rényi Divergence Between the Prior and Posterior JMPD

The function D_α in (16) is a measure of the divergence between the densities p_0 and p_1. In our application, we are interested in computing the divergence between the predicted density $p(\mathbf{X}^k, T^k|\mathbf{Z}^{k-1})$ and the updated density after a measurement is made by a single sensor at location $\mathbf{r}$, denoted $p(\mathbf{X}^k, T^k|\mathbf{Z}^k, \mathbf{r})$. Therefore, the relevant divergence is

$$D_\alpha\left(p(\cdot|\mathbf{Z}^k, \mathbf{r})||p(\cdot|\mathbf{Z}^{k-1})\right) = \frac{1}{\alpha - 1} ln \int_{\mathbf{X}} p(\mathbf{X}^k, T^k|\mathbf{Z}^k, \mathbf{r})^\alpha p(\mathbf{X}^k, T^k|\mathbf{Z}^{k-1})^{1-\alpha} d\mathbf{X}^k \ , \tag{19}$$

where the multitarget state $\mathbf{X}$ is a vector of variable dimension. The symbol $\int_{\mathbf{X}} f(\mathbf{X})d\mathbf{X}$ is used as short hand notation to denote the integral over the domain. This can be precisely written as

$$\int_{\mathbf{X}} d\mathbf{X} f(\mathbf{X}, T) \doteq \sum_{T=0}^{\infty} \int d\mathbf{x}_1 ... \mathbf{x}_T f(\mathbf{x}_1 \cdots \mathbf{x}_T, T) \ . \tag{20}$$

Using Bayes' formula applied to the JMPD (5) we obtain

$$D_\alpha \left(p(\cdot|\mathbf{Z}^k, \mathbf{r}) || p(\cdot|\mathbf{Z}^{k-1}) \right) = \frac{1}{\alpha - 1} ln \frac{1}{p(\mathbf{z}^k|\mathbf{Z}^{k-1}, \mathbf{r})^\alpha} \int_{\mathbf{X}} p(\mathbf{z}^k|\mathbf{X}^k, T^k, \mathbf{r})^\alpha p(\mathbf{X}^k, T^k|\mathbf{Z}^{k-1}) d\mathbf{X}^k \ . \tag{21}$$

To make the formulation explicitly clear, we use $p(\mathbf{z}^k|\mathbf{Z}^{k-1}, \mathbf{r})$ to denote the distribution on sensor outputs at time k by a sensor at $\mathbf{r}$ given the set of previous measurements $\mathbf{Z}^{k-1}$.

3.3. The Expected Rényi Divergence for a Sensing Action

Our real aim is to predict the value of different sensing actions *before actually receiving* the measurement $\mathbf{z}$. To this end, we calculate the expected value of the divergence for each possible action. This expectation predicts the amount of information gain that would be received if a sensor were positioned above a particular cell for all possible positionings.

The expected value of the divergence may be written as an integral over all possible outcomes $\mathbf{z}$ when performing a measurement at $\mathbf{r}$ as

$$\phi(\mathbf{r}) \equiv \langle D_\alpha \rangle_{\mathbf{r}} = \int d\mathbf{z}^k p(\mathbf{z}|\mathbf{Z}^{k-1}, \mathbf{r}) D_\alpha \left(p(\cdot|\mathbf{Z}^k, \mathbf{r}) || p(\cdot|\mathbf{Z}^{k-1}) \right) \ . \tag{22}$$

As is discussed in,[1] this integral can be calculated in $O(C * N_{parts})$ where C is the number of sensor cells and N_{parts} is the fidelity of the particle filter approximation to the JMPD.

4. INFORMATION BIASED PHYSICOMIMETICS FOR MULTIPLE PLATFORM SENSOR MANAGEMENT

In this section we present the details of the physicomimetic model used to generate sensor platform motion. The underlying model is taken from molecular dynamics as discussed by Allen and Tildesley.[16] The sensors obey a Brownian dynamics model similar to the colored noise model sometimes used to model maneuvers in target tracking applications. The new feature is that artificial forces from the other sensors and the information field are also coupled to the Brownian dynamics. The forces used here are constructed so that they depend only on the sensor positions and are independent of their velocities. Let $\mathbf{f}_i$ be the force on sensor i due to the information field gradient and all of the other sensors (to be defined below). Then the acceleration of a unit mass object obeys the Langevin equation

$$\ddot{\mathbf{r}}_i(t) = -\frac{1}{\tau} \dot{\mathbf{r}}_i(t) + \mathbf{f}_i(t) + d\beta_i(t) \ , \tag{23}$$

where $\beta_i(t)$ is an 2-vector white noise process with $E[d\beta(t) d\beta(t')^\top] = 2\frac{\sigma_m^2}{\tau} \delta(t - t') \mathbf{1}_{2\times 2}$, $\mathbf{1}_{2\times 2}$ is the 2×2 identity matrix, and $\mathbf{r}_i$ is a real 2-vector giving the position of sensor i. In this model the noise processes driving each sensor are independent, in analogy to the usual assumption of molecular Brownian dynamics. In the absence of external forces, this leads to a zero-mean random noise process with exponentially auto-correlated velocity, $E[\dot{\mathbf{r}}(t) \dot{\mathbf{r}}(t + t')] = \sigma_m^2 \exp(-|t'|/\tau)$.

In principal, the sensor platform motion is obtained by integrating (23) in time. Since closed form solutions are not generally available, numerical methods must be used. There are a host of solution approaches within the molecular dynamics literature, most of which are based on finite differencing schemes. It is important to understand qualitatively the trade between speed, accuracy and stability. If insufficient care is exercised in selecting the differencing scheme then very small time integration steps are required to obtain an accurate, stable approximation to the underlying continuum equations. In artificial physics one is not attempting to

model a particular physical system, so accuracy is lower priority than in molecular dynamics applications, but stability is still a requirement. For this reason we have elected to use the Brownian dynamics Verlet algorithm (a predictor-corrector algorithm detailed in[16]). The algorithm proceeds in time steps of size κ. This requires storage of both position $\mathbf{r}_i^k$ and velocity $\dot{\mathbf{r}}_i^k$ for each sensor i at time step k. The force on sensor i at time k due to the information field and the locations of all other sensors is denoted $\mathbf{f}_i^k$. The first step in Brownian Verlet is to update the sensor positions according to

$$\mathbf{r}_i^{k+1} = \mathbf{r}_i^k + c_1\kappa\dot{\mathbf{r}}_i^k + c_2\kappa^2\mathbf{f}_i^k + \delta\mathbf{r}_i^k \quad , \tag{24}$$

where $\delta\mathbf{r}_i^k$ is a random noise sequence specified below. The updated positions are used to compute $\mathbf{f}_i^{k+1}$ and the velocities are updated as

$$\dot{\mathbf{r}}_i^{k+1} = c_0\dot{\mathbf{r}}_i^k + (c_1 - c_2)\kappa\mathbf{f}_i^k + c_2\kappa\mathbf{f}_i^{k+1} + \delta\dot{\mathbf{r}}_i^{k+1} \quad , \tag{25}$$

where

$$c_0 \equiv \exp(-\kappa/\tau) \tag{26}$$

$$c_1 \equiv \frac{\tau}{\kappa}(1 - c_0) \tag{27}$$

$$c_2 \equiv \frac{\tau}{\kappa}(1 - c_1) \tag{28}$$

and $(\mathbf{r}_i^{k+1\top}, \dot{\mathbf{r}}_i^{k+1\top})^\top$ is a zero-mean discrete time Gaussian noise process. It is fully specified by requiring that the x- and y-components are un-correlated and that $(r_{ix}^{k+1}, \dot{r}_{ix}^{k+1})^\top$ has covariance $\mathbf{Q_s}$ with elements are

$$Q_{srr} = \sigma_m^2\tau^2(4\exp(-\kappa/\tau) - 3 - \exp(-2\kappa/\tau) + 2\kappa/\tau) \tag{29}$$

$$Q_{s\dot{r}\dot{r}} = \sigma_m^2(1 - \exp(-2\kappa/\tau)) \tag{30}$$

$$Q_{sr\dot{r}} = \sigma_m^2\tau(1 - 2\exp(-\kappa/\tau) + \exp(-2\kappa/\tau)) \tag{31}$$

The information gain surface is a scalar field. To derive a force from it, we simply take its gradient,

$$F_I(\mathbf{r}) = -\beta\nabla_{\mathbf{r}}\phi(\mathbf{r}) \tag{32}$$

where β is a coupling constant that must be designed to obtain good performance. This component specifies the relative strength of the information gradient force. The minus sign arises from the fact that sensors are to be attracted to regions of high information gain.

To provide a coordinating interaction amongst the sensors we use a generalization of the so-called Lennard-Jones potential that serves as a zeroeth order model for the intermolecular forces of liquids (it approximates the dipole-dipole caused by charge fluctuations in polarizable neutral molecules). The Lennard-Jones potential for a pair of objects separated by a distance r is

$$V_{LJ}(r) = \epsilon\left[\left(\frac{\gamma}{r}\right)^m - \left(\frac{\gamma}{r}\right)^n\right] \tag{33}$$

where $m > n > 0$. For the standard Lennard-Jones potential $m = 12$ and $n = 6$ so it is sometimes referred to as the 6-12 potential. The shape of the potential for several values of m and n is shown in Figure (4).

Observe that V_{LJ} is repulsive at short range and attractive at long range. For the standard 6-12 case, the attractive region has quite a short range. Also, as m is reduced, the attractive range increases. The Lennard-Jones force is radial and has signed magnitude

$$F_{LJ}(r) = -\epsilon\left[m\frac{\gamma^m}{r^{m+1}} - n\frac{\gamma^n}{r^{n+1}}\right] \tag{34}$$

The total external force on sensor i is then the sum of the Lennard-Jones forces felt from all of the other sensors (34) and the information gradient force (32).

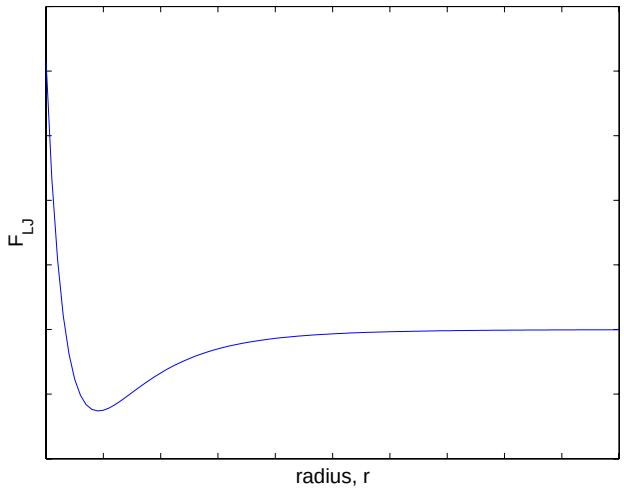

Figure 1. The Lennard-Jones force between two sensors as a function of radial distance, r

5. SIMULATION RESULT

In this section, we present results of the proposed multi-platform sensor management algorithm. We use ten targets moving in a $5000m$ x $5000m$ surveillance area. Each target is modeled using the four-dimensional state vector $[x, \dot{x}, y, \dot{y}]'$. Target trajectories for the simulation come directly from a set of recorded data based on GPS measurements of vehicle positions over time collected as part of a battle training exercise at NTC. Targets routinely come with in sensor cell resolution (i.e. cross). Target positions are recorded at 1 second intervals, and the simulation duration is 500 time steps.

At initialization both the number of targets and the states of the individual targets is unknown to the filter. The goal of the algorithm is to position the sensors over time so as to learn how many targets are in the surveillance region and the states (position and velocity) of each.

The filter assumes constant velocity motion with a large diffusive component as the model of target kinematics. As the simulations use real target motion, this model is severely at odds with the actual target behavior. The approximation to the JMPD uses 250 particles, each of which has an estimate of both target number and the states of the individual targets.[1]

At each time step, the M sensors are updated based on the algorithm described in Section 4 and a measurement is made from each. Measurements are made with a 100m x 100m detection cell resolution. For this simulation, we assume thresholded measurements as described in Section 2. Therefore, when making a measurement the imager returns either a 0 (no detection) or a 1 (detection) governed by P_d, P_f, and SNR. This model is known by the filter and used to evaluate the update equation. In this illustration, we take $P_d = 0.5$, $SNR = 10dB$, and $P_f = P_d^{(1+SNR)}$, which is a standard model for thresholded detection of Rayleigh returns.

The performance of the algorithm is measured in two ways. First, we compare the estimated number of targets to the true number of targets, where the estimated number of targets at time k is defined as

$$\hat{T}^k = \sum_{T=0}^{\infty} T \int_{\mathbf{X}} d\mathbf{X} p(\mathbf{X}, T|\mathbf{Z}) \tag{35}$$

Second, we use the ground truth to calculate the number of actual targets that are successfully tracked by the filter. For each of the hypothesized target t, we have an estimate of the target state as

$$\hat{x}_t^k = \int \mathbf{x}_t d\mathbf{x}_1 \cdots \mathbf{x}_t p(\mathbf{x}_1 \cdots \mathbf{x}_T|\mathbf{Z}) \ . \tag{36}$$

There are issues of permutation symmetry, partition sorting, and the particle filter implementation that are not covered here that allow this estimation to be made. They are discussed at length in[7,8] .

The target estimates are then matched up with the ground truth to give a measure of how many true targets are being successfully tracked, which we denote $r_{tracked}$. Note this measure captures both targets that should have been detected but weren't as well as targets that were successfully detected but then poorly tracked. These two measures taken in combination allow for determination of the number of false targets initiated as well as the number of true targets not under track.

Figure 2 presents the results of a Monte Carlo simulation where the number of sensors in the surveillance region were varied. We contrast the performance of the plain physicomimetic model with the information-biased physicomimetic model.

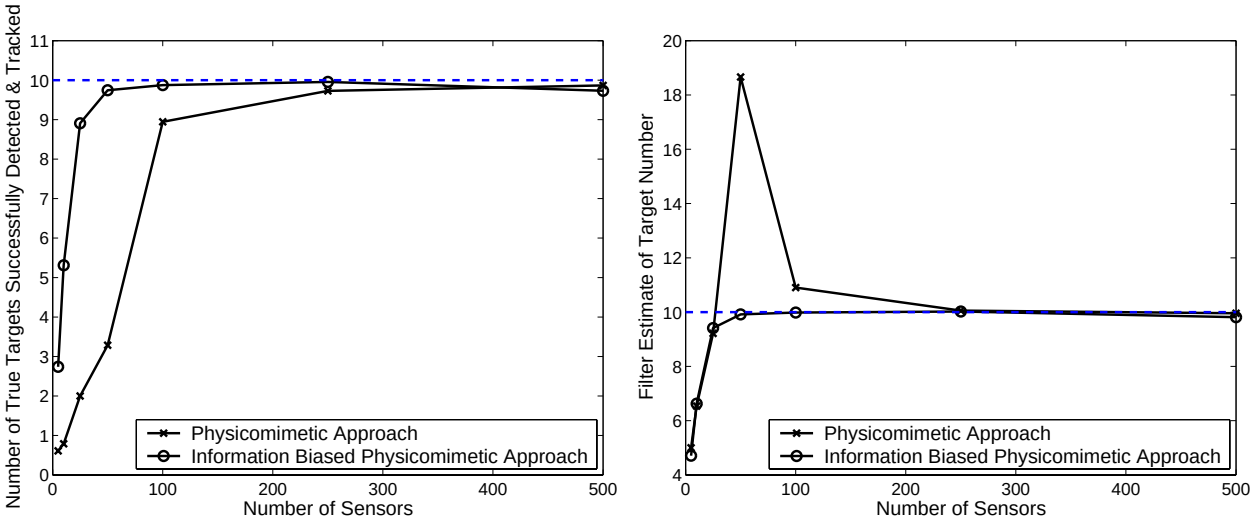

Figure 2. The performance of the information based method of multi-platform tasking described in this paper. The algorithm performance is shown as a function of number of sensors available to the resource manager. The performance is measured in two ways. First (left), performance is measured by the number of targets that are actually correctly detected and tracked. Second (right), performance is measured by the filter estimate of target number. The true number of targets is 10 so in both curves, ideal performance is 10 (shown by the dashed lines). The figures illustrate that the approach that includes coupling to the information gradient with 50 sensors has similar performance to the plain physicomimetic approach with 250 sensors.

6. CONCLUSIONS

The principal objective of this work has been to demonstrate the utility of combining artificial physics and information-based sensor management to direct the motion of large numbers of mobile sensors. This preliminary work clearly demonstrates that coupling the sensors to the information field provides improved detection and tracking performance relative to the case without using the information field. The relative importance of the information field, the Lennard-Jones coordinating field and the Brownian dynamics is less clear and requires further study.

Another interesting question is how to extend this to more complex action spaces where, say, the sensor must maneuver as it does here and in addition, optimize the sensor mode and other degrees of freedom. In this case we have multiple information gain surfaces corresponding to the different degrees of freedom for the sensor.

Perhaps the most challenging questions relate to the stability and optimality of the system. Both the data fusion system and the artificial physics systems in this approach have a large number of parameters. On the data fusion side, we have detection and false alarm probabilities, sensor resolution and target motion parameters. These parameters are directly related to the physics of the system and therefore can be modeled exactly if such

information is available. In the artificial physics implementation we have the parameters characterizing the coupling to the information surface, the parameters characterizing the Lennard-Jones interaction amongst the sensors and the Brownian dynamics parameters. These parameters are not directly coupled to the physics and so in that sense they are more difficult to set optimally. Furthermore, there is no theory indicating how these parameters interact. Ideally, given the sensor and target parameters, we would like to be able to determine all of the AP parameters or at least have some idea how they scale.

For example, if the coupling to the information field is too large, then the sensors will tend to all be attracted to a single region. Are there criteria for adjusting the couplings to guarantee that this cannot occur? Suppose the AP parameters have been selected to provide good performance for targets of a particular SNR. Can we at least derive scaling rules for how a good set of AP parameters should be modified as the SNR, target plant noise and sensor resolution are varied? Currently the answer is no.

REFERENCES

1. C. M. Kreucher, K. Kastella, and A. O. Hero III, "Information based sensor management for multitarget tracking," *Proceedings of SPIE Conference on Signal and Data Processing of Small Targets* , 2003.
2. K. J. Hintz, "A measure of the information gain attributable to cueing," *IEEE Transactions on Systems, Man and Cybernetics* **21**(2), pp. 237–244, 1991.
3. K. Kastella, "Discrimination gain for sensor management in multitarget detection and tracking," *IEEE-SMC and IMACS Multiconference CESA* **1**, pp. 167–172, 1996.
4. R. Mahler, "Global optimal sensor allocation," *Proceedings of the Ninth National Symposium on Sensor Fusion* **1**, pp. 167–172, 1996.
5. J. Borenstein, "Real-time obstacle avoidance for fast mobile robots," *IEEE Transactions on Systems, Man, and Cybernetics* (5), pp. 1179–1187, 1989.
6. W. Spears, R. Heil, D. Spears, and D. Zarzhitsky, "Physicomimetics for mobile robot formations," *Proceedings of the Third International Joint Conference on Autonomous Agents and Multi Agent Systems (AAMAS-04)* **3**, pp. 1528–1529, 2004.
7. C. M. Kreucher, K. Kastella, and A. O. Hero III, "Tracking multiple targets using a particle filter representation of the joint mulitarget proabbility density," *Proceedings of SPIE Conference on Signal and Data Processing of Small Targets* , 2003.
8. C. M. Kreucher, K. Kastella, and A. O. Hero III, "Multitarget tracking using a particle filter representation of the joint multitarget probability density," *to appear in IEEE Transactions on Aerospace and Electronic Systems* , 2005.
9. Y. Bar-Shalom and W. D. Blair, *Multitarget-Multisensor Tracking: Applications and Advances, Volume III*, Artech House, 2000.
10. B. E. Tullsson, "Monopulse tracking of rayleigh targets: A simple approach," *IEEE Transactions on Aerospace and Electronic Systems* **27**(3), pp. 520–531, 1991.
11. C. H. Gowda and R. Viswanatha, "Performance of distributed CFAR test under various clutter amplitudes," *IEEE Transactions on Aerospace and Electronic Systems* **35**(4), pp. 1410–1419, 1999.
12. Y. Bar-Shalom, *Multitarget Multisensor Tracking: Advanced Applications*, Artech House, 1990.
13. A. Rényi, "On measures of entropy and information," *Proceedings of the 4th Berkeley Symposium on Mathematics, Statistics, and Probility* **1**, pp. 547–561, 1961.
14. A. O. Hero III, B. Ma, O. Michel, and J. Gorman, "Applications of entropic spanning graphs," *IEEE Signal Processing Magazine (Special Issue on Mathematics in Imaging)* **19**(5), pp. 85–95, 2002.
15. A. O. Hero III, B. Ma, O. Michel, and J. D. Gorman, "Alpha divergence for classification, indexing and retrieval," *Technical Report 328, Comm. and Sig. Proc. Lab. (CSPL), Dept. EECS, The University of Michigan* , 2001.
16. M. P. Allen and D. J. Tildesley, *Computer Simulation of Liquids*, Oxford University Press, 1989.