Stochastic Distributed Algorithms for Target Surveillance

Luis Caffarelli[1], Valentino Crespi[2], George Cybenko[2], Irene Gamba[1], and Daniela Rus[3]

[1] Department of Mathematics, University of Texas at Austin
{caffarel,gamba}@mail.ma.utexas.edu
[2] Thayer School of Engineering, Dartmouth College
{vcrespi,gvc}@dartmouth.edu
[3] Computer Science Department, Dartmouth College gvc@dartmouth.edu

Summary. In this paper we investigate problems of target surveillance with the aim of building a general framework for the evaluation of the performance of a system of autonomous agents. To this purpose we propose a class of semi-distributed stochastic navigation algorithms, that drive swarms of autonomous scouts to the surveillance of grounded targets, and we provide a novel approach to performance estimation based on analysing sequential observations of the system's state with information theoretical techniques. Our goal is to achieve a deeper understanding of the interrelations between randomness, resource consumption and ergodicity of a decentralized control system in which the decision–making process is stochastic.

Key words: UAV Systems, decentralized control and multi-agent systems.

1 Introduction

Employment of Agent-based technology in the design of distributed control systems is notoriously limited by the lack of a general unified model for performance estimation. The difficulty in this matter concerns both the qualification of candidate solutions and the quantification of their performance.

In this paper we deal with problems related to target surveillance[4]. Informally, we want to deploy a number of flying UAV's (Unmanned Autonomous Vehicles) with the main objective of monitoring a set of targets located on ground at fixed positions. We design a distributed agent-based navigation control system that allows the scouts to provide full coverage of the sites and, at the same time, confound the enemy to prevent its counter measures, with the

[4] Research partially supported by the DARPA TASK program under grant number F30602-00-2-0585.

least possible consumption of fuel per target visit. The importance of those problems relies upon the recent increasing demand for large fault-tolerant autonomous systems composed of fully automated, unmanned, easy-to-deploy and cheap vehicles.

The main purpose of this investigation is to understand nontrivial interrelations between various elements that characterize a distributed multiagent system. In particular we study how randomization affects metrics like energy consumption and the degree of unpredictability of vehicle trajectories. We provide an original application of techniques from Information Theory to the analysis of sequential observations of the system's state in order to quantify the quality of the proposed solutions.

Unlike in classical methods, based on solving a global optimization problem to provide an optimal solution we look for distributed solutions under conditions in which traditional techniques, like dynamic programming, do not apply. Our idea of approaching these problems using agent-based technologies is motivated by

- the need for decentralization in control systems characterized by hundreds of autonomous agents;
- requirements of robustness and fault tolerance with respect to abrupt increase or decrease of the number of UAV's in play;
- the need for a chaotically nondeterministic evolution of the system's state, dictated by the necessity for protection against confidentiality attacks (trajectories or throughput rates prediction).

Methodologically, we follow a three-stage process:

1. Under the assumption that the resources available to the UAV's are unbounded (e.g., they have full knowledge of the global state), we define a family of initial solutions, typically by applying techniques from classical control theory and dynamic systems theory. In this case we define a family of randomized potential functions for the navigation of the UAV's that combines, in an original fashion, a stochastic decision–making procedure with an artificial potential technique (examples of applications of artificial potentials and artificial fields in UAV's navigation and robot path planning are contained in [1, 2] and in [3, 4, 5, 6], respectively).
2. Global resources are replaced with local resources compatibly with the constraints of the given decentralized environment. During this stage we look for phase transitions in order to identify portions of the parameter space in which the Objective is fulfilled.
3. Finally, we tune the parameters of the artificial potential functions for efficient control of the scouts. This allows the system to evolve with enough nondeterminism at a reasonable cost. The resulting algorithms (implemented using a Montecarlo D/E numerical simulation) drive each scout (UAV) along the steepest descending direction of a stochastically determined potential. Instances of our methodology can be found in [7, 2].

Summarizing, we formalize our Surveillance Problem (Section 2) along with the metrics we measure (Section 3). Then we introduce a new class of navigation algorithms[5] (Section 4) and conduct an experimental analysis of their performance (Section 5). We execute a phase-transition analysis in order to identify the conditions under which a fair trade of resources can take place and finally we discuss fuel vs "space aperiodicity" trade-offs. Considerations on future extensions of this work conclude the paper (Section 6).

2 Problem Definition

We want to design a distributed algorithm that drives a swarm of UAV's to surveil targets disseminated on a given battlefield under the following assumptions consistent with the available technology:

- The vehicles can access global information by communicating with a satellite, in general, under bandwidth limitations. The sort of information that each vehicle can collect, despite such limitations, includes the coordinates of all the targets on the ground and the statistics of the monitoring level of any target at a specific time.
- The vehicles are able to fly at different altitudes so they never collide.
- The vehicles have unlimited maneuvering capabilities in the ray of curvature and in the acceleration, i.e., they can perform sharp turns.

We formulate our Surveillance problem in 2D as follows:

Instance: A battlefield consisting of a terrain and a set of m targets located at fixed positions.

Problem: Deploy n scouts (UAV's) $\{s_1, s_2, \ldots, s_n\}$ with the following Objective and Optimization requirements:

1. **Objective:** guarantee that each target receives a proper share of visits over time (e.g., equal coverage).
2. **Optimization/Trade-off:** drive scouts along unpredictable trajectories minimizing the consumption of fuel per target visit (see Section 3).

In this paper we will adopt the following terminology. We denote with $\mathbf{x}_i(t) \in \mathbf{R}^2$ the position of UAV s_i at time t and with $\mathbf{y}_j \in \mathbf{R}^2$ the position of target $j \in [m]$.

3 Objectives, Metrics and Measurements

We are going to provide a quantitative analysis of our algorithms, based on the following metrics:

[5] See http://actcomm.thayer.dartmouth.edu/task/demos.html

1. *Energy consumption per target visit of UAV's.*
   In our model we assume that energy and fuel consumption of a single UAV is proportional to the arc length of its trajectory. Thereby, in our analysis we estimate the lengths of those trajectories by adding lengths of segments of polygonal curves that approximate the actual curves.
2. *Predictability and stability of the trajectories.*
   This metric is important as if the potential enemy knows in advance which target is going to be visited in a certain time frame, it will be able to adopt undesirable counter measures. Let us consider the following event generation process: each time a target is visited by any scout we record the corresponding target id. Then as the system evolves we will be able to collect a (potentially infinite) sequence of target id's. Our idea is to study the compressibility of such sequences in order to estimate the entropy of the stochastic source that has generated it. Maximum entropy would then correspond to total unpredictability of the next target to be visited or, equivalently, to space aperiodicity of the trajectories[6]. We measure space aperiodicity by applying information theoretical tools. In particular we run our simulations recording the sequence of target id's as they are visited. Then we use Lempel-Ziv [8]-based Universal coding schemes to compress the string of such recordings and finally we compute the compression ratio. These values provide us with a measure of the entropy of the source.
3. *Application performance with respect to the Objective.*
   We measure of how well the algorithms ensure that targets receive a proper share of attention (in this case uniform) from the scouts. Formally, let $\theta_u = \frac{1}{m}\mathbf{1}_m$ be the uniform distribution over the set of targets, representing equal number of visits (on average) per target in a time unit. Then we compute stable (stationary) statistics of the portions of surveillance per target, say θ, and evaluate its distance from θ_u according to some suitable distance metric. In this paper we consider what in the theory of mixing Markov Chains is called *the variation distance* $d(\theta_u, \theta) = \frac{1}{2}\sum_j |\frac{1}{m} - \theta_j|$. We also compute the information entropy[7] of θ: $H(\theta) = \sum_j \theta_j \log_2 \frac{1}{\theta_j}$ that should provide us with a measure of the imbalance of θ with respect to uniformity.

As we will observe experimentally, energy consumption and predictability of the trajectories are metrics that cannot be optimized at the same time. Intuitively, optimization of fuel results into the scouts moving deterministically and this in turn makes their trajectories very predictable. So, in general we look at methods to obtain trade-offs between energy and space aperiodicity, under the condition that targets are visited on equitable basis.

[6] We focus on space aperiodicity since time periodicity can be easily broken by randomizing the speed of the scouts (within their capabilities).

[7] This should not be confused with the entropy of the source of target id's that we mentioned before.

4 Our Algorithms

4.1 Randomized Potentials

As anticipated our algorithms are based on artificial potentials to be minimized through a gradient descent method combined with a stochastic decision–making rule.

Each target j has a dynamic mass $m_j = M(p_j)$ that generates a gravitational potential V_j. This mass depends in general upon a dynamic priority $p_j(t)$. At each time t each drone is subjected to the linear superposition of a stochastically selected subset of all the potentials. Formally, scout i in position $\mathbf{x}$ at time t resents of the influence of potential

$$V^{(i)}(\mathbf{x},t) = \sum_{j=1}^{m} \sigma_{i,j} \cdot V_j(\|\mathbf{x} - \mathbf{y}_j\|, t) = \sum_{j=1}^{m} \sigma_{i,j} \cdot (-1) \frac{M(p_j(t))}{\|\mathbf{x} - \mathbf{y}_j\|^{\alpha}}, \qquad (1)$$

where $V_j(r,t) = -\frac{m_j(t)}{r^{\alpha}}$ and $\sigma_{i,j} \in \{0,1\}$ are Bernoulli random variables.

Targets are given dynamic priorities that increase with time and are reset when they are visited by any scout. Each scout s_i needs to make a selection of one or more targets through a randomized assignment of K ones to $\sigma_{i,j}$, for $j \in [m] = \{1, 2, \ldots, m\}$. The value $K = \sum_j \sigma_{i,j} = \#\{\sigma_{i,j} = 1 \mid j \in [m]\}$ can be fixed a priori or free to range from 1 to m and, as we will see in a moment, it is very important for the control of the level of randomness incorporated in the algorithm.

So, at time t scout i either has been assigned a target (binary) vector $\sigma_i = (\sigma_{i,j})$ defining a subset of targets that exert attraction to it, or has to select a new target vector by sampling from a probability distribution that is defined through the dynamic priorities and the Euclidean distances $\{d(s_i, l_j) = \|\mathbf{x}_i - \mathbf{y}_j\| \mid j \in [m]\}$. The selection of a target vector is irrevocable and the selecting scout has to reach one of the constituent targets before making a new selection. The quantity $p_j(t)$, the dynamic priority of target j at time t can be defined as:

$$\begin{cases} p_j(t+1) := p_j(t) + \delta \cdot w_j & \text{target } j \text{ not visited at time } t \\ p_j(t+1) := 0 & \text{target } j \text{ visited at time } t \end{cases}$$

The vector $\mathbf{w} = \{w_1, w_2, \ldots, w_m\}$ is intended to be a vector of static priorities.

With these ingredients we can define a set of n basic probability distributions (one per scout) over the set of targets:

$$q_{i,j}(t) = \frac{p_j(t) \cdot d(s_i, t_j)^{-\eta}}{Q_i}, \quad Q_i = \sum_j p_j(t) \cdot d(s_i, t_j)^{-\eta}, \qquad (2)$$

where $\eta = \alpha + 1$ and α is the parameter that occurs in the formula of the gravitational potentials generated by the targets. And from those a set of n probability distributions:

$$P_{i,j}(\beta) = \frac{q_{i,j}^{\beta}}{N_i(\beta)}, \quad N_i(\beta) = \sum_j q_{i,j}^{\beta} \ . \tag{3}$$

Observe that, as $\beta \to 0^+$, $P_{i,j}(\beta) \to \frac{1}{m-1}$, whereas as $\beta \to \infty$ we can prove the following lemma.

Lemma 1. *Let $\alpha = 1$ and $u_{i,j} = p_j/d_{i,j}^2$. Define $I_i = \arg\max\{u_{i,k} \mid 1 \leq k \leq m\}$ as the set of indices of the maximal elements in the sequence $(u_{i,k})_k$. Then as $\beta \to \infty$:*

$$P_{i,j}(\beta) \to P_{i,j} = \begin{cases} \frac{1}{|I_i|} & \text{if } j \in I_i, \\ 0 & \text{otherwise.} \end{cases}$$

Proof. Let $j_0 \in I_i$. Then substituting 2 in 3 and rearranging terms we obtain:

$$P_{i,j}(\beta) = \frac{u_{i,j}^{\beta}}{\sum_k u_{i,k}^{\beta}} = \frac{u_{i,j}^{\beta}}{u_{i,j_0}^{\beta}(|I_i| + \sum_{k \notin I_i} (\frac{u_{i,k}}{u_{i,j_0}})^{\beta})}$$

which clearly tends to $1/|I_i|$ for $j \in I_i$ and to 0 otherwise. We observe in fact that terms $\frac{u_{i,k}}{u_{i,j_0}}$ in the summation are less than 1.

Finally the K ones are sampled from the distribution $P_{i,j}(\beta)$ as follows. Suppose we have an urn containing m numbers $1, 2, \ldots, m$, then we extract the K values $j_1, j_2, \ldots, j_K$ one by one without re-insertion. At stage u the probability of extracting j_u is defined as $P\{j_u\} = P_{i,j_u}(\beta)/\{1 - \sum_{k=1}^{u-1} P_{i,j_k}(\beta)\}$.

4.2 The Simulation

Our algorithms are based on a gradient descent method simulated with a Discrete/Event mechanism[8]. Each agent updates its position following the steepest decreasing direction of its potential [9]. This is precisely the opposite of the gradient vector of the potential function. So, according to this method, each scout i computes the vector $-\nabla V^{(i)} = -\sum_j \sigma_{i,j} \nabla V_j$ and moves along its direction. One way to implement this is through the following rule:

$$\mathbf{x}_i(t+1) = \mathbf{x}_i(t) - \alpha^{\frac{1}{\alpha+1}} \cdot \frac{\nabla V^{(i)}(\mathbf{x}_i(t))}{\|\nabla V^{(i)}(\mathbf{x}_i(t))\|^{\frac{\alpha+2}{\alpha+1}}}. \tag{4}$$

Since

[8] We have produced Matlab, C and Java implementations.

$$\nabla V_j(\|\mathbf{x}-\mathbf{y}\|) = V_j'(\|\mathbf{x}-\mathbf{y}\|)\frac{\mathbf{x}-\mathbf{y}}{\|\mathbf{x}-\mathbf{y}\|} \ ,$$

V_j' has constant sign and $V^{(i)}(\mathbf{x}_i) \approx V_j(\|\mathbf{x}-\mathbf{y}_j\|)$, when scout i is sufficiently close to target j (and $\sigma_{i,j}=1$), it is not hard to see that

$$\alpha^{\frac{1}{\alpha+1}} \cdot \frac{\nabla V^{(i)}(\mathbf{x}_i(t))}{\|\nabla V^{(i)}(\mathbf{x}_i(t))\|^{\frac{\alpha+2}{\alpha+1}}} \approx \mathbf{x}_i - \mathbf{y}_j$$

so when a scout reaches the proximity of one of its targets its new gradient adjusts gracefully avoiding overshooting.

The algorithm may be described as follows. At the beginning, say at time $t=0$, all the agents, deployed uniformly at random over the region of interest, compute their current destinations, together with the needed travel times using random dynamic priorities and so purely random target vectors. This is accomplished by applying the position update rule (4) and assuming that the velocities are constant in norm (and here for simplicity equal to v for all). This first step produces:

- a description of the system state that consists of the current agent positions and velocities $X=\{\mathbf{x}_i\}$, $V=\{\mathbf{v}_i\}$ and
- a discrete set E of events (t,i) signifying that agent i will reach its current designated destination at time t.

Then the simulation proceeds endlessly as follows:

1. Extraction of the event (t,j) that is meant to occur first, i.e., the one that minimizes the element t. Let t_0 be such time;
2. The positions $X(t)$ of all the drones are let evolve by computing iteratively $X(t+\Delta t)$; $t \leftarrow t+\Delta t$; in accordance to the assumed kinematics (rectilinear uniform motion along the directions $\mathbf{v}_i$), until either $t+h\cdot\Delta t > t_0$ or one or more targets are visited by any scouts;
3. In the first case, the event fires and the position update rule is applied to the agent j associated with the extracted event (t,j) to compute its new designated destination and, as a consequence, its new velocity $\mathbf{v}_j'$ and travel time t_j':

$$\mathbf{x}_j' = \mathbf{x}_j - \alpha^{\frac{1}{\alpha+1}} \cdot \frac{\nabla V^{(j)}(\mathbf{x}_j)}{\|\nabla V^{(j)}(\mathbf{x}_j)\|^{\frac{\alpha+2}{\alpha+1}}}, \mathbf{v}_j' = v\frac{\mathbf{x}_j'-\mathbf{x}_j}{\|\mathbf{x}_j'-\mathbf{x}_j\|}, t_j' = \frac{\|\mathbf{x}_j'-\mathbf{x}_j\|}{v} \quad (5)$$

   This in turn causes the generation of a new event (t_0+t_j', j) to be included in the set E.
4. In the second case, all the scouts that are crossing over targets will have to sample a new target vector, apply the position update rule, compute speeds and travel times and cast the appropriate events in E.

A pseudo-code description of our Discrete Event Simulation algorithm would be the following[9]:

```
MainProgram
Input: α, β, δ, w
begin
  E ← ∅; t ← 0
  for j ∈ Targets AND i ∈ Scouts do
    (x_i, y_j, p_j, σ_{i,j}) ← RandomValues
    m_j ← 1
    UpdateRule(i, t)
  endo
  while E ≠ ∅ do
    (t_0, i_0) ← min(E)
    E ← E \ {(t_0, i_0)}
    while t + Δt < t_0 do
      X(t) ← X(t + Δt); t ← t + Δt
      [VisitedTrgts, VisitingScts] ← CheckVisits(X(t), Y)
      GenerateEvent(VisitingScts,VisitedTrgts )
      p ← p + δ · w
    endo
    X(t) ← X(t_0); t ← t_0
    [VisitedTrgts, VisitingScts] ← CheckVisits(X(t), Y)
    GenerateEvent(VisitingScts,VisitedTrgts)
    UpdateRule( i_0, t )
    p ← p + δ · w
  endo
end
```

```
GenerateEvent(A, B)
begin
  for i ∈ A AND j ∈ B do
    p_j ← 0
    σ_i ← ExtractTgtVector(K, β, p)
    UpdateRule(i, t)
  endo
end
```

UpdateRule(i,t)
begin

$$\mathbf{x}_i' = \mathbf{x}_i - \alpha^{\frac{1}{\alpha+1}} \cdot \frac{\nabla V^{(i)}(\mathbf{x}_i)}{\|\nabla V^{(i)}(\mathbf{x}_i)\|^{\frac{\alpha+2}{\alpha+1}}};$$

$$\mathbf{v}_i' = v\frac{\mathbf{x}_i'-\mathbf{x}_i}{\|\mathbf{x}_i'-\mathbf{x}_i\|}; \quad t_i' = \frac{\|\mathbf{x}_i'-\mathbf{x}_i\|}{v}$$

$\mathbf{x}_i \leftarrow \mathbf{x}_i$; $\mathbf{v}_i \leftarrow \mathbf{v}_i'$

$E \leftarrow E \cup \{(t+t_i', i)\}$

end

[9] Here we have adopted the following terminology: $K = \sum_j \sigma_{i,j}$, $\mathbf{m} = (m_i)$, $M(\mathbf{m}) = (M(m_i))$, $\mathbf{p} = (p_i)$, $\mathbf{w} = (w_i)$, $\sigma_i = (\sigma_{i,j})$, $X(t) = (\mathbf{x}_i(t))$, $Y = (\mathbf{y}_i)$, $V(t) = (\mathbf{v}_i(t))$ and $X(t+\Delta t) = X(t) + V(t)\cdot \Delta t$.

5 Experiments

Let us first discuss the role and importance of parameters K, the number of ones that occur in the target vectors σ_i and β, the exponent that regulates the relation between the sample probabilities $P_{i,j}$ and the dynamic priorities p_j (the feedback control). We observe the following facts.

- The values of both K and β control in some sense the amount of randomness that is incorporated in the target vector selection.
- For $K = m - 1$ (all ones except for the target last visited that is zero), we obtain a totally deterministic algorithm where no random choice is made at all. Moreover, for α large enough, the gradient direction will tend to align along $\nabla \sup_j V_j$. This implies that scouts would always be headed towards their closest targets.
- At the other extreme is the case $K = 1$, where scout i selects only one target j with probability $P_{i,j}(\beta)$. This means that any target is selectable and reachable in principle as long as the corresponding probability is nonzero. So, arbitrary target vectors imply the capability of scouts to penetrate through any "cloud" of targets and reach the farthest ones.

We have verified experimentally that the distribution of the proportions of surveillance realized by the algorithm stabilizes relatively quickly (after 6000 steps). Besides all our experiments have been conducted in stationary regime.

5.1 Space aperiodicity.

To measure this metric we exploited the incompressibility of random sequences of numbers. We have considered many choices of n scouts and m targets and run hundreds of simulations for thousands of steps. For each run we have recorded the sequence of targets as they were visited and measured the compressibility of those sequences using the Lempel-Ziv coding scheme. Given a sequence of target recordings S and its compressed representation $c(S)$, we define $r = 100 \cdot (1 - l(c(S))/l(S))$ ($l(x)$ is the length of string x). So, $r = 0$ means no compression at all (no regularities) whereas $r \approx 100$ means high compression (high regularity).

5.2 *K* varying: phase-transition analysis of the reachability phenomenon.

In this experiment we set $\beta = 1$ and let K vary from 1 to $m - 1$. We observed a breakdown of $H(\theta)$ as K grows above 2 (see Fig. 2). This confirms our theoretical intuition that "fat" target vectors (numerous multiple destinations) drastically limit the reachability of the peripheral targets as scouts restrict their patrols to targets located nearby the center of mass of the system.

5.3 $K = 1$: space aperiodicity vs fuel consumption trade-off.

In order to maintain a "fair" distribution of surveillance: $\theta \approx \frac{1}{m}\mathbf{1}$ (fulfillment of the Objective), we set $K = 1$ and study the energy consumption and space aperiodicity as β ranges from 0 to ∞. Our idea here is to trade energy with space aperiodicity under the invariant condition that all the targets are to be visited on equitable basis. We observed that as β tends to infinity the probability distribution $P_{i,j}(\beta)$ tends very often to a $(0,1)$-vector $P_{i,j}$ (see Lemma 1, in case $|I_i| = 1$) with only one component set to 1. This means that as β grows the choice of the next target to visit becomes more and more of a deterministic function of the inter-target Euclidean distances and their associated dynamic priorities. And this results into a lower fuel consumption per target visit (see Fig 1). The intuition behind this is that, in presence of total determinism, scouts would tend to travel to the closest targets with consequent saving of fuel. This saving would be attained at the expenses of a more chaotic behavior desired to confound the enemy. Figure 1 displays values of fuel consumption and space aperiodicity, $n = 5$ scouts, $m = 8$ targets arranged on a fixed deployment, $\alpha = 1$ and constant masses. The compression rates have been computed on 10000 target recordings and the results averaged over 50 trials.

The resulting data reveal that we can reasonably trade energy for space aperiodicity by tuning parameter β. The two curves $E(\beta)$ and $r(\beta)$ are indeed monotonic (decreasing the first and increasing the second) as expected. In particular we can see that $\beta = 2$ allows the lowest consumption of energy while keeping the compression ratio optimal ($r = 0$).

We also would like to observe that our trade is most meaningful for $\beta < 6$ as for higher values no appreciable saving of energy can be attained. This was expected, as our algorithms were not designed to compete with the best global deterministic solutions. In fact, the special case $n = 1$ (one scout) would be roughly equivalent to the Euclidean TSP (Traveling Salesman Problem) that is notoriously NP-hard [10]. Moreover the validity of the triangular inequality guarantees approximability with performance (error) ratio ≤ 2, whereas our algorithms in pure deterministic regime would not be guaranteed to attain a bounded performance (error) ratio.

6 Conclusions and Future Work

We have introduced a new class of semi-distributed algorithms for a multi-agent surveillance system and conducted an experimental analysis to synthesize an optimized solution. Now, we would like to investigate the mathematical relation between the quantities E, r and H. Besides, our algorithms still require knowledge of the dynamic priorities of all the targets at any time and so the next step will be to investigate systems in which scouts have only limited access to that information and need to exchange their "experiences" when

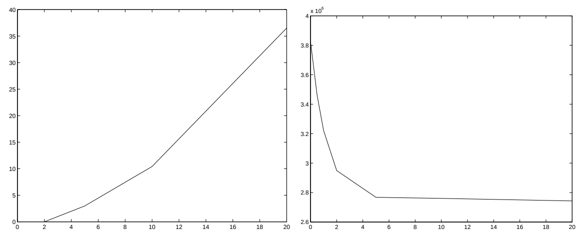

Fig. 1. Trade-off Analysis: $D = 5$: $r(\beta)$ (left) and $E(\beta)$ (right).

they come at close range. Finally, our phase-transition analysis leads to possible formalizations of the system as a percolation process since we may see agents as particles that float around the targets. Then one problem is to study the circulation of those particles (e.g., whether or not they remain imprisoned inside potential wells). This direction is interesting but must be pursued with care as it may lead to intractable problems. Statistical mechanics provides examples of multi-agent systems whose performance analysis of their emergent behaviors requires the computation of NP-hard quantities (e.g. partition functions in Ising models for ferromagnets) [11].

References

1. Valentino Crespi and George Cybenko. Agent-based Systems Engineering and Intelligent Vehicles and Road Systems. *Darpa Task Program white paper. http://actcomm.thayer.dartmouth.edu/task/*, April 2001.
2. Valentino Crespi, George Cybenko, Daniela Rus, and Massimo Santini. Decentralized Control for Coordinated flow of Multiagent Systems. In *Proceedings of the 2002 World Congress on Computational Intelligence. Honolulu, Hawaii*, May 2002.
3. J.C. Latombe. *Robot Motion Planning.* Kluver Academic Publishers, 1991.
4. Elon Rimon and Daniel E. Koditschek. Exact Robot Navigation using Artificial Potential Functions. *IEEE Transactions on Robotics and Automation, October 1992*, 8(5):501–518, 1992.
5. Leena Singh, Herry Stephanou, and John Wen. Real-time Robot Motion Control with Circulatory Fields. In *Proceedings of the International Conference on Robotics and Automation*, pages 2737–2742, 1996.
6. D. Keymeulen and J. Decuyper. The Fluid Dynamics Applied to Mobile Robot Motion: The Stream Field Method. In *Proceedings of the International Conference on Robotics and Automation*, pages 378–386, 1994.
7. Valentino Crespi, George Cybenko, and Daniela Rus. Decentralized Control and Agent–Based Systems in the Framework of the IRVS. *Darpa Task Program paper. http://actcomm.thayer.dartmouth.edu/task/*, Apr 2001.
8. Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory.* Wiley Interscience, 1991.
9. D.P. Bertsekas and J.N. Tsitsiklis. Gradient Convergence in Gradient Methods. *SIAM J. on Optimization*, 10:627–642, 2000.

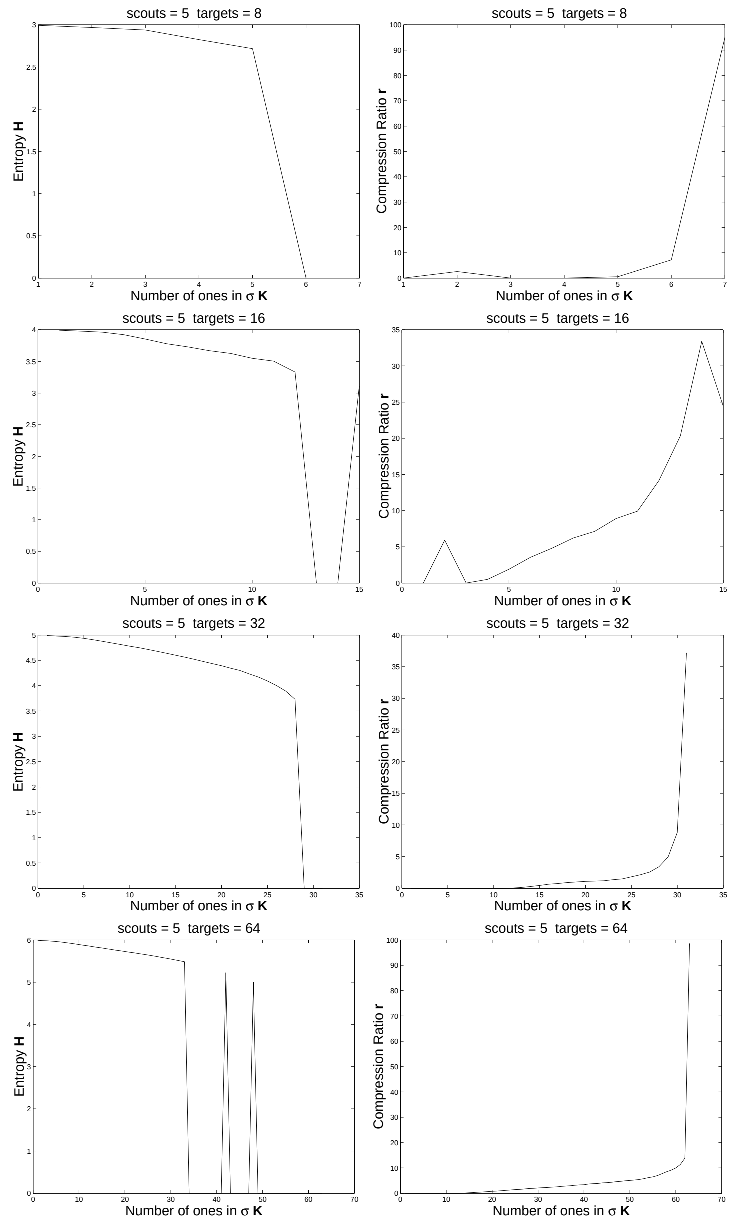

Fig. 2. Phase transition analysis: $D = 5$, $T \in \{8, 16, 32, 64\}$.

10. Micheal R. Garey and David S. Johnson. *Computers and Intractability.* W.H. Freeman and Company, 1979.
11. Mark Jerrum and Alistair Sinclair. *Polynomial Time Approximation Algorithms for the Ising Model*, volume 443. Springer Verlag, 1990.