Surveillance of Unmanned Aerial Vehicles Using Probability Collectives

Přemysl Volf, David Šišlák, Dušan Pavlíček, and Michal Pěchouček

Agent Technology Center, Department of Cybernetics
Faculty of Electrical Engineering, Czech Technical University in Prague
volf@labe.felk.cvut.cz
http://agents.felk.cvut.cz

Abstract. A rising deployment of unmanned aerial vehicles in complex environment operations requires advanced coordination and planning methods. We address the problem of multi-UAV-based area surveillance and collision avoidance. The surveillance problem contains non-linear components and non-linear constraints which makes the optimization problem a hard one. We propose discretization of the problem based on the definition of the points of interest and time steps to reduce its complexity. The objective function integrates both the area surveillance and collision avoidance sub-problems. The optimization task is solved using a probability collection solver that allows to distribute computation of the optimization. We have implemented the probability collective solver as a multi-agent simulation. The results show the approach can be used for this problem.[1]

Keywords: Surveillance, Collision Avoidance, Probability Collectives, Multi-Agent Systems.

1 Introduction

Deployment of unmanned aerial vehicles for performing autonomous tasks is a very important topic these days. The algorithm for persistent cooperative surveillance of a given area is provided in the paper. Operation of several autonomous vehicles in a shared airspace above the area where surveillance is executed also involves a parallel task of maintaining safe flight operation.

A combined task of autonomous coordinated surveillance and collision avoidance of airplanes is formally defined in the form of an objective function. The combined task is formulated as a search for a certain combination of actions of airplanes which minimizes the objective function.

The problem of multi-UAV surveillance has been studied lately and a variety of approaches from reactive policies to deliberative search-based methods have been proposed. However, no approach to UAV surveillance has been found that explicitly deals with non-linear constraints such as occlusion.

[1] This work is supported by the Air Force Office of Scientific Research, Air Force Material Command, USAF, under grant number FA8655-06-1-3073 and by Czech Ministry of Education grant number 6840770038.

V. Mařík, P. Vrba, and P. Leitão (Eds.): HoloMAS 2011, LNAI 6867, pp. 235–245, 2011.

A reactive approach to surveillance with multiple UAVs is described in [11]. The authors start from a simple optimal reactive policy for one UAV and two distinct points in space that have to be continually monitored to derive a structure for the policy. This simple policy is then extended to the case of multiple points and finally to multiple UAVs. In the case of multiple UAVs a space partitioning algorithm is utilized to divide the space among the UAVs and then the UAVs use a single-UAV reactive policy to determine their flight paths in their respective partitions.

In [4] the authors propose a class of semi-distributed stochastic navigation algorithms based on minimization of artificial potentials with two aims: 1. to provide a robust and efficient algorithm for surveillance and 2. to decrease the predictability of trajectories the group of UAVs follows. In comparison to our proposed method, the algorithm supposes all airplanes are flying in different altitudes and thus the safe separation is ensured.

A notable work on routing problems for vehicles with non-holonomic motion constraints is [12]. The work describes single and multi UAV routing problems for aircraft and provides a set of approximation algorithms with stated upper and lower bounds on their performance. The author introduces two novel approximation algorithms for planning the shortest trajectory through a set of points.

The problems addressed in the above mentioned papers are defined as more specific sub-problems of the surveillance problem and they are solved very efficiently. We propose a method that is able to solve a more general definition of the surveillance problem, but still keep the feasibility and efficiency.

The *Probability Collectives* approach [8,15] is used as an optimization solver in the paper. The probability collectives is a stochastic optimizer using probabilistic operators optimizing over a variable space. The probability collectives approach differs from other existing stochastic approaches such as Genetic Algorithms [7] and Particle Swarm Optimization [13] which operate on the design of variables' values (mutation, crossover) rather than on the probability distributions of their values.

The use of the probability collectives algorithm makes the designed algorithm *distributable*, and thus suitable for a multi-agent system.

2 The Problem Specification

Airplanes $A_i \in \mathcal{A}$ operate in a shared three dimensional *airspace* $Air \subseteq \mathbb{R}^3$. The airspace is situated above an area $E \subseteq \mathbb{R}^2$. The fourth dimension is continuous time, $Time \subseteq \mathbb{R}^+$. The airspace is naturally limited by the ground surface (terrain obstacles) and airspace boundaries.

Each airplane $A_i \in \mathcal{A}$ is modeled as a three-dimensional non-holonomic system[2] (a three-dimensional version of a non-holonomic Dubins car [5]). The airplane is modeled as a mass point called *pivot* with a given position $\overline{x}$ and direction

[2] A non-holonomic system is a system whose state depends on the path taken to achieve that state [3].

vector $\overline{v}$. The airplane has several restrictions on its performance characteristics included in the model – minimum and maximum velocity, maximum acceleration and deceleration, maximum pitch angle and minimum horizontal and vertical turning radius dependent on the current speed of the airplane.

Each UAV is equipped with the camera sensor C. The sensor is defined as a function $C : Air \rightarrow \mathscr{P}(D)$[3]. In complex urban environments, the field of view of a UAV's on-board sensors can become occluded in the presence of tall buildings and/or narrow streets. This can result in uncovered areas that need to be covered by changing flight trajectories or by introduction of another airplane.

The *flight plan* fp_i of the airplane A_i is defined as a function of time – $fp_i : Time \rightarrow Air_i$, where $fp_i(t)$ represents the position of the airplane A_i at time t. The path given by fp_i is smooth[4] in all coordinates, velocity and time. fp_i respects the airplane's flight constraints described above at any position and thus it is performable by A_i. $\mathcal{FP} = \{fp_i\}_{A_i \in \mathcal{A}}$ is a set of current flight plans of all airplanes.

The *persistent area surveillance* is one of the most common information collection tasks, typically defined as a problem of maintaining an up-to-date picture of the situation in a given area. We define the general *information age* objective function $\mathcal{I}_{age}$, with t_s being the initial time and t_f the final time of the optimization task assuming all airplanes using flight plans:

$$\mathcal{I}_{age}(\mathcal{FP}, t_s, t_f) = \int_{t_s}^{t_f} \left[\iint_E value(x, y, \mathcal{FP}, t) dx dy \right] dt \tag{1}$$

The double integral part expresses the age of information about the area at one particular time moment. The function $value(x, y, \mathcal{FP}, t)$ defines the age of each point in the area. The function has a general form, it can be non-linear and it does not need to be continuous. The function can compute its value based on the last time the point was seen by any airplane, the priority of the point, aging function etg. The $\mathcal{I}_{age}(\mathcal{FP}, t_s, t_f)$ can be used for definition of any information age function supposing independent evaluation of the area points.

We define the multi-UAV common operational picture maintenance problem to be an optimization problem: i.e., find a set of trajectories

$$\mathcal{FP}^* = \arg\min_{\mathcal{FP}} \{\mathcal{I}_{age}(\mathcal{FP}, t_s, t_f)\} \tag{2}$$

All airplanes executing their surveillance task operate asynchronously implementing the *free flight concept* – flying along the best flight trajectory and implementing the See & Avoid capability for collisions. The collision avoidance mechanism is crucial to allow safe and efficient operation of multiple UAVs within a shared airspace and has to be included in the surveillance task.

The collision detection algorithm uses a *safety zone* to ensure the minimum separation requirements. The safety zone $S(A_i, \overline{x})$ considers an airplane $A_i \in \mathcal{A}$ and defines a cylindrical space with a given center $\overline{x} \in \mathbb{R}^3$ of the cylinder. The

[3] $\mathscr{P}(D)$ stands for the power set of D.

[4] By smooth we mean that its first derivation is continuous.

radius and height of the safety zone is given by the parameters of the airplane A_i and include both the required separation for the airplane A_i and the allowed deviation level between the true and the assumed position of the airplane itself. Two flight plans fp_i and fp_j have a *collision* at time t :

$$\mathbf{col}(fp_i, fp_j, t) = \begin{cases} 1 & \text{if } i \neq j,\ fp_i(t) \in S(A_j, fp_j(t)) \\ 1 & \text{if } i \neq j,\ fp_j(t) \in S(A_i, fp_i(t)) \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

Two different UAVs A_i and A_j (with their flight plans fp_i and fp_j) have a *collision* if and only if $\exists t : \mathbf{col}(A_i, A_j, t) = 1$. The collision is denoted as $A_i \otimes A_j$ or $A_j \otimes A_i$. The *collision avoidance* problem (CA) for a given $\mathcal{FP}$ is defined as a process of finding such $\{fp'_i\}_{A_i \in \mathcal{A}}$ for which no collision exists at any time.

3 Discretized Approach

The general surveillance problem in its continuous form is an *optimal control* constraint optimization problem with non-linear constraints (UAV dynamics) and a non-linear objective function (due to sensor model). These problems are known to be intractable in a general case [14]. It is also known that the travelling repairman problem for a Dubins vehicle, a problem closely related to the optimal area information age minimization, is NP-hard [12]. We propose an approximation of the problem, as it is necessary to achieve feasible computation performance.

The approximation method is based on the discretization of the problem. We have performed this discretization in two ways. The first step is to create a suitable set of *points of interest* within the surveillance area covering it with a certain density and respecting its surface and structure. The optimization algorithm then minimizes the information age of these points. The second step is to discretize the time into time steps allowing computation of the information age function as a combination of *area situation pictures*. The area situation picture captures the information age of the points of interest at a single moment. The altitude and the velocity of the airplanes is constrained to a constant value to simplify the definition of optimization variables[5]. To reduce the complexity of the optimization, we define an optimization task for a single moment in time and prediction of the future movement of the airplanes. To achieve a desired optimization approximation during the whole surveillance problem, the optimization process is regularly re-invoked with updated predictions of future trajectories of airplanes according to the actual situation. We define the optimization task (corresponding to equation 2) at time t_0 and look-ahead *control interval* ΔT with discrete variables $\overline{a}$ specified as

$$\arg\min_{\overline{a} \in \mathscr{A}} \mathrm{G}(\overline{a}, t_0) \ , \tag{4}$$

[5] Different altitudes and velocities can be reached by enlarging the set of possible values defined for the optimization variables $\overline{a}$.

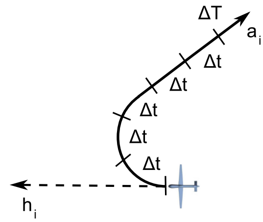

Fig. 1. Application of the new heading

where G is the *objective function* and $\overline{a} = \langle a_1, a_2, \ldots a_n \rangle$ is the *joint move*. The joint move is a variable vector, where each variable a_i is assigned to the airplane A_i. a_i represents a control action of the airplane. The a_i defines the desired heading of the airplane, while the actual heading is denoted as h_i. If $a_i = h_i$, the airplane applies a straight flight for the control interval ΔT. If $a_i \neq h_i$, the airplane applies an appropriate turn to reach the heading a_i using the minimum turn radius, see Figure 1. The value of a_i is within the range of $\langle 0, 360)$ discretized by the predefined *heading step* h_{step}. We use fp_{a_i} to denote a flight plan based on the selected heading a_i. Flight plan fp_{a_i} is valid for a single selection of a_i from time t_0 onwards. Let us denote $\mathcal{FP}_{\overline{a}} = \{fp_{a_1}, fp_{a_2}, \ldots fp_{a_n}\}$.

The objective function is defined as

$$\mathrm{G}(\overline{a}, t_0) = \mathrm{G}^{surv}(\overline{a}, t_0) + \alpha \mathrm{G}^{col}(\overline{a}, t_0) \; . \tag{5}$$

G^{surv} corresponds to the definition of the information age (equation 1), but it is defined at time t_0 and for look-ahead ΔT. It computes the *discretized information age* of the area E for the control interval ΔT assuming the application of flight plans fp_{a_i} computed using a_i. The function G^{surv} is computed using a finite set of points of interests $P = \{p_1, p_2, \ldots, p_k\}$ defined for the area E.

$$\mathrm{G}^{surv}(\overline{a}, t_0) = \sum_{t=t_0}^{t_0+\Delta T} w^{surv}(t) \sum_{i=1}^{k} value(p_i, \mathcal{FP}_{\overline{a}}, t) \tag{6}$$

The inner sum represents the situation picture of the area at the respective time and the $w^{surv}(t)$ function represents the weight of the area situation picture depending on its distance in the future. The further away in the future the area situation picture is, the less weight is assigned to it, because there exists a higher probability that flight plans of the airplanes will be changed during some of the future optimization rounds.

G^{col} penalizes separation violation among all airplanes $\mathcal{A}$ using their predicted flight plans fp_{a_i} and required separation distances. It is computed as

$$\begin{gathered}\mathrm{G}^{col}(\overline{a}, t_0) = \sum_{a_i, a_j \in \overline{a}, a_i \neq a_j} \mathrm{G}^{col}_{pair}(a_i, a_j, t_0) \; , \\ \mathrm{G}^{col}_{pair}(a_i, a_j, t_0) = \sum_{t=t_0}^{t_0+\Delta T} w^{col}(t) \mathbf{col}(fp_{a_i}, fp_{a_j}, t)\end{gathered} \tag{7}$$

The $w^{col}(t)$ function is analogous to $w^{surv}(t)$ and represents the weight of the collisions depending on their distance in the future.

4 Probability Collectives Optimizer

In this section, we describe the details of the Probability Collectives (PC) theory applicable to the optimization problem with discrete variables. The PC theory can be viewed as an extension to the conventional game theory. Let us have a game with N players $i \in \mathcal{I}$. A mixed strategy of the player i is a probability distribution $q_i(x_i)$ over the player's possible pure strategies (a definition set of x_i). Each player i chooses its strategy (a value of the variable x_i) independently by sampling $q_i(x_i)$. There is no direct communication between players in the game. Players learn to co-operate through repeated plays, their probability distributions are updated using the received reward based on the objective function $G(\overline{x})$ combining all variables. The probability distribution of the *joint-move* $q(\overline{x})$ is

$$q(\overline{x}) = \prod_{i \in \mathcal{I}} q_i(x_i) \ . \tag{8}$$

Bounded rational players [6] balance their choice of the best move with the need to explore other possible moves. The information theory shows that the equilibrium of a game played by bounded rational players is the optimum of a Lagrangian of the probability distribution of the agents' joint-moves [15]. This equilibrium corresponds to at least a local minimum of the original objective function G. The expected world utility of all the players with a common world utility G under given players' distributions $q_i(x_i)$ is

$$\mathrm{E}_q(\mathrm{G}(\overline{x})) = \sum_{\overline{x} \in \mathcal{X}} \mathrm{G}(\overline{x}) q(\overline{x}) = \sum_{\overline{x} \in \mathcal{X}} [\mathrm{G}(\overline{x}) \prod_{i \in \mathcal{I}} q_i(x_i)] \ . \tag{9}$$

In the Nash equilibrium, every player adopts a mixed strategy that maximizes its expected utility with respect to the mixed strategies of others[6]. The Nash equilibrium assumption requiring full rationality is replaced by the information available to the players. This amount of information is the negated *Shannon entropy* of the distribution $q(\overline{x})$,

$$S(q) = -\sum_{\overline{x} \in \mathcal{X}} [q(\overline{x}) \ln[q(\overline{x})]] \ . \tag{10}$$

Using the *maximum entropy principle* (Maxent) [9], each player searches for the probability distribution q that minimizes the expected utility

$$\arg\min_q \ \mathrm{E}_q(\mathrm{G}(\overline{x})) \ , \tag{11}$$

[6] In this paper, maximization of utility is replaced with minimization of information age to be consistent with the defined objective function.

subject to $\sum_{x_i \in \mathcal{X}_i} q_i(x_i) = 1$ and $q_i(x_i) \geq 0$ for each $i \in \mathcal{I}$. From the *gradient-based optimization* we have to find the critical point of the *Maxent Lagrangian*

$$\mathfrak{L}(q, T) \equiv \mathrm{E}_q(\mathrm{G}(\overline{x})) - TS(q) \ , \tag{12}$$

where T is the Lagrange parameter (also referred to as the *temperature*). We need to find q and T such that $\frac{\partial \mathfrak{L}}{\partial q} = \frac{\partial \mathfrak{L}}{\partial T} = 0$.

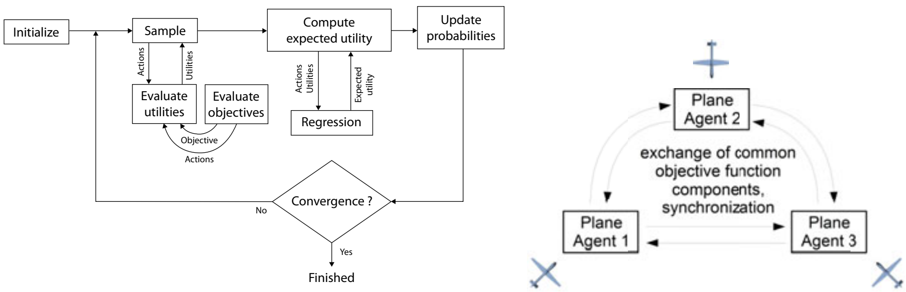

Fig. 2. Iterative procedure lowering $\mathrm{E}_q(\mathrm{G}(\overline{x}))$ (left). The multi-agent implementation of Probability Collectives optimization (right) [2].

The algorithm lowering $\mathrm{E}_q(\mathrm{G}(\overline{x}))$ is an iterative procedure of the following steps (see Figure 2, [2]):

1. Initialize the value of the Lagrange parameter T.
2. Minimize the Lagrangian $\mathfrak{L}(q, T)$ with respect to q at specified T (sample the system, update the probabilities).
3. Reduce the value of T and repeat from Step 2 until q converges (q does not vary more than the specified threshold for a couple of iterations).
4. The $\overline{x}$ selected according to the final q is the solution of Equation 4.

The sequence lowering T is the *annealing schedule*, in reference to the Simulated Annealing [1]. For a given T, each player i optimizes

$$\begin{aligned} \mathfrak{L}_i(q_i, T) = & \sum_{x_i \in \mathcal{X}_i} [q_i(x_i) \mathrm{E}[\mathrm{G}|x_i]] \\ & - T \sum_{x_i \in \mathcal{X}_i} [q_i(x_i) \ln[q_i(x_i)]] \ . \end{aligned} \tag{13}$$

The function $\mathfrak{L}_i$ is convex, has a single minimum in the interior, the temperature T controls the trade-off between exploration and exploitation [16]. The first term $\sum_{x_i \in \mathcal{X}_i} [q_i(x_i) \ \mathrm{E}[\mathrm{G}|x_i]]$ in Equation 13 is minimized by a perfectly *rational* player while the second term $-T \sum_{x_i \in \mathcal{X}_i} [q_i(x_i) \ln[q_i(x_i)]]$ is minimized by a perfectly *irrational* player (by a perfectly uniform mixed strategy q_i). In the limit, $T \to 0$, the set of q that simultaneously minimizes the Lagrangian is the same as the set of q minimizing the objective function G.

Input: $\mathcal{A}$
Output: a_i

1 $\mathscr{A}_i \leftarrow$ `Get` a_i `definition set(`h_{step}`)`;
2 $q_i \leftarrow$ `Get initial distribution(`$\mathscr{A}_i$`)`;
3 $T \leftarrow$ `Initialize temperature`;
4 **while** *true* **do**
5     $s_i[N_{SB}] \leftarrow$ `Sampling(`$\mathscr{A}_i$, q_i, N_{SB}`)`;
6     $pred_i[N_{SB} \times \Delta T] \leftarrow$ `Predictions(`s_i, ΔT`)`;
7     $wait\_set \leftarrow \mathcal{A} \setminus A_i$;
8   `Send(`$pred_i$, $wait\_set$`)`;
9   **while** $wait\_set \neq \emptyset$ **do**
10         $A_j, pred_j \leftarrow$ `Fetch other predictions`;
11         $wait\_set \leftarrow wait\_set \setminus A_j$;
12   **end**
13     $\mathrm{G} \leftarrow$ `Compute(`$pred_i$, $pred_j$`)`;
14     $q_i \leftarrow$ `Update distribution(`$\mathscr{A}_i$, q_i, s_i, G, T`)`;
15     $T \leftarrow$ `Update temperature(`T`)`;
16   **if** `Converged(G)` **then break**;
17 **end**
18 **return** `Sample final control(`$\mathscr{A}_i$, q_i`)`;

Algorithm 1. Agent PC optimization pseudocode

4.1 Multi-Agent Implementation

The Probability Collectives optimization can be implemented in a parallel way as a multi-agent system. Collectives can be viewed as a group of self-interested, learning agents that act together to minimize the objective function (Equation 4), see Figure 2. Each agent searches for an optimal action for a single airplane that corresponds to one variable in the optimization. In this section, $A_i \in \mathcal{A}$ denotes the agent providing control to the airplane A_i. Each A_i keeps the current probability distribution q_i for its action variable a_i. Computation of the expected utility value (value of the common objective function $\mathrm{G}(\overline{a}, t_0)$ in some time moment t_0, Equation 5) and the convergence test requires cooperation of all agents. Sampling and updating of all variables in the iterative procedure of the PC algorithm can be performed independently in a parallel way.

Each A_i is configured by parameters: the size of the heading step h_{step} , the size of the sample block in each iteration N_{SB}, the look-ahead control interval ΔT, the balancing factor α, the parameters defining the weight of future area situation pictures and collisions $w^{surv}(t)$ and $w^{col}(t)$, and annealing schedule parameters.

Algorithm 1 presents a distributed implementation of the PC optimization procedure executed by each agent. First, each agent performs an initial setup of the definition set $\mathscr{A}_i$, the probability distribution q_i as an uniform discrete distribution over $\mathscr{A}_i$, and the temperature T according to the selected annealing schedule initial value (lines 1–3).

The iterative optimization loop lowering $\mathrm{E}_q(\mathrm{G}(\overline{a}, t_0))$ from Figure 2 is implemented at lines 4–17. Agents prepare sample blocks s_i (N_{SB} actions selected

Fig. 3. Screenshots of the system with six airplanes. Lighter color means lower age of the data

from $\mathscr{A}_i$ using Monte-Carlo sampling [10]) and prediction points $pred_i$ (for each action in s_i agents compute positions in the predicted flight plan fp_{a_i} from t_0 to $t_0 + \Delta T$ using step Δt), lines 5 and 6. Then agents exchange their $pred_i$, lines 7 and 8. Each agent waits for other sample block predictions from all agents from the set $\mathcal{A} \backslash A_i$. When the agent has all predictions, it computes the objective function (equation 5), line 13.

Update of the agent's probability distribution minimizing Lagrangian $\mathfrak{L}_i(q_i, T)$ with the current temperature T is done at line 14. Then the temperature T is decreased according to the common annealing schedule, line 15. The convergence test of the iterative optimization procedure is done simultaneously by all agents, line 16. It is not necessary to communicate during this phase as all agents have the same $\mathrm{G}(\overline{a}, t_0)$ value. Eventually, the agent selects the final control according to its stabilized probability distribution q_i, line 18.

5 Evaluation

Experiments, see Figure 3, were performed using a fixed flat square area 500 x 500 m covered by 900 points of interest placed into a regular grid (matrix 30 x 30). All airplanes were flying with the velocity 25 m/s, turning radius 20 m and 15 m safety zone. The heading step h_{step} was set to 11.25 degrees, the computation step Δt was set to 0.5 sec and the optimization process was repeated every 3.5 seconds. A number of parameters (e.g. airplanes and their initial positions) were varied throughout the experiments.

We discuss the influence of parameters on the performance of the simulation. The initial positions of the airplanes (fixed positions, semi-randomized, randomized) have minimal influence on the performance in the very beginning of the simulation but not later.

Setting a lower initial age of points of interest leads to an increased emphasis on exploration rather than surveillance in the initial phase. The exploration task forces airplanes to cover unattended points first.

The length of the control interval ΔT determines linearly the time needed for each optimization computation. Linear increase of solution quality was measured for lower values, slower increase for higher ones.

The balancing factor α is crucial to maintain safe separation between airplanes. It is necessary to set α such that the outcome of the objective function in case of a single predicted collision is worse than any value of the objective function in case of no collisions at all.

The size of the sampling block influences the chance that the best combination of all control actions a_i will be included in the solution. Decreasing the size of the sampling block can lead to a situation where there are so few combinations available that the majority of airplanes with reasonable control actions can outweigh the minority of airplanes with completely unsuitable control actions and thus force the minority to perform useless actions. The selected solution is still the best one from the generated (small) set, but very far from the optimal one. Increasing the size of the sampling block leads to a substantial increase in computation time.

6 Conclusion

The paper addresses a problem of general area surveillance using multiple autonomous UAVs. Simplification is needed as the general problem is intractable. The discretization of time and space (using points of interest) is used as a natural simplification. The discretized problem is formulated as an optimization problem of minimizing information age of points of interest using discrete variables. This transformation of the problem allows to formulate it as a probability collective optimization problem. The probability collectives is a stochastic optimizer using probabilistic operators optimizing over a a set of variables. The variables can be computed in a distributed and parallel way and thus the solver can be implemented as a multi-agent system, where each agent represents one variable. We show the feasibility of this approach to solve a general surveillance problem and effective utilization of multiple UAVs.

References

1. Bar-Yam, Y.: Dynamics of Complex Systems. Perseus Books, Cambridge (1997)
2. Bieniawski, S.R.: Distributed optimization and flight control using collectives. Dissertation Stanford University (2005)
3. Bloch, A.M.: Nonholonomic Mechanics and Control. Springer, NY (2003)
4. Caffarelli, L., Crespi, V., Cybenko, G., Gamba, I., Rus, D.: Stochastic Distributed Algorithms for Target Surveillance. Intelligent Systems Design and Applications, 137 (2003)
5. Dubins, L.E.: On curves of minimal length with a constraint on average curvature, and with prescribed initial and terminal positions and tangents. American Journal of Mathematics (79), 497–516 (1957)
6. Fudenberg, D., Levine, D.K.: The Theory of Learning in Games. MIT Press, Cambridge (1998)
7. Goldberg, D.E.: Genetic Algorithms in Search, Optimization and Machine Learning. Addison-Wesley Longman Publishing Co., Inc., Boston (1989)

8. Lee, C.F., Wolpert, D.H.: Product distribution theory for control of multi-agent systems. In: AAMAS 2004: Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems, pp. 522–529. IEEE Computer Society, Washington, DC, USA (2004)
9. MacKay, D.: Information theory, inference, and learning algorithms. Cambridge University Press, Cambridge (2003)
10. Metropolis, N., Ulam, S.: The monte carlo method. Journal of the American Statistical Association 44(247), 335–341 (1949)
11. Nigam, N., Kroo, I.: Persistent Surveillance Using Multiple Unmanned Air Vehicles. In: 2008 IEEE Aerospace Conference, pp. 1–14 (2008)
12. Savla, K.: Multi UAV Systems with Motion and Communication Constraints. PhD thesis, University of California (2007)
13. Schutte, J., Greonwold, A.: Optimal sizing design of truss structures using the particle swarm optimization algorithm. In: 9th AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, pp. 2002–5639. AIAA (September 2002)
14. Vollmer, H.: Computational complexity of constraint satisfaction. In: Cooper, S.B., Löwe, B., Sorbi, A. (eds.) CiE 2007. LNCS, vol. 4497, pp. 748–757. Springer, Heidelberg (2007)
15. Wolpert, D.H.: Information theory – the bridge connecting bounded rational game theory and statistical physics. In: Braha, D., Minai, A.A., Bar-Yam, Y. (eds.) Complex Engineered Systems, pp. 262–290. Springer, Berlin (2006)
16. Wolpert, D.H., Bieniawski, S.: Distributed control by lagrangian steepest descent. In: 43th IEEE Conference on Decision and Control (December 2004)