

# Finding Optimal Trajectory Points for TDOA/FDOA Geo-Location Sensors

Ran Ren, Mark L. Fowler and N. Eva Wu  
Department of Electrical and Computer Engineering  
State University of New York at Binghamton  
Binghamton, New York 13902  
Email: {rran1, mfowler, evawu}@binghamton.edu

**Abstract**—In emitter geo-location estimation systems, it is well known that the geometry between sensors and the emitter can seriously impact the accuracy of the location estimate. Here we consider a case where a set of sensors is tasked to perform a sequence of location estimates on an emitter as the sensors progress throughout their trajectories. The goal is to select the trajectories so as to optimally improve the location estimate at each step in the sequence. To build the optimal trajectories, the aircraft, at their current locations, need to know their optimal next states at the time of next estimation, under the constraint of a reachable set due to limited reachable velocity or thrust. In this paper, we propose a one-step method to tackle the optimal next state(ONS) problem using the Particle Swarm Optimization(PSO) by solving the optimal amount of applied thrust along the flying trajectories. Simulation results show that the proposed method dramatically improves the estimation accuracy along the flying trajectories, compared to the random walk and constant velocity scheme. We also show that the estimation accuracy performance is also insensitive to the problem dimensionality.

## I. INTRODUCTION

Emitter location estimation has been proved to be a widely applicable technique in many fields. Specifically, we consider the scenario that a group of airborne sensors try to passively estimate the location of a non-cooperative RF emitter on a battlefield. In recent interest, those sensors are placed on small and slow moving *Unmanned Aerial Vehicles* (UAVs). At each time instance, the UAVs fly at certain velocities within the reachable velocity range. We consider that the sensors employ Time Difference of Arrival (TDOA) and Frequency Difference of Arrival (FDOA) to estimate the geo-location of the emitter, in which case, UAVs must be paired to estimate those values. For simplicity, we will not explore the effect of optimal sensor selection and pairing [1], rather we simply consider that the UAVs are paired arbitrarily.

In emitter geo-location estimation systems, it is well known that the geometry between sensors and the emitter can seriously impact the accuracy of the location estimate. Here we consider a case where a set of sensors is tasked to perform a sequence of location estimates on an emitter as the sensors progress throughout their trajectories. The goal is to select the trajectories so as to optimally improve the location estimate at each step in the sequence. The position and velocity of the sensor at the scheduled time of the next estimation task is called the sensor’s “next state”, as defined

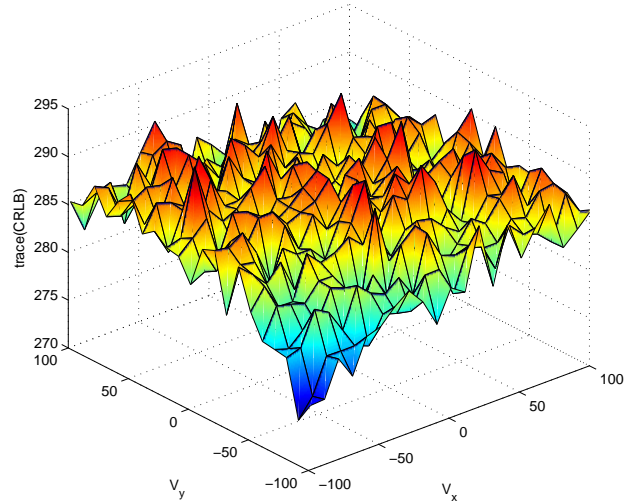


Fig. 1. The Multimodal Characteristics of the Emitter Location Estimation Problem

in [2]. To build the optimal trajectories, each aircraft, at their current states, needs to know its reachable set of next states, computes the optimal next states within the reachable set, and then solves a state/control-constrained problem to steer it to the optimal next state in a time coordinated manner.

To illustrate the challenging nature in determining the optimal next state, among other steps, consider 4 sensors paired into two distinct pairs and consider the optimization of the X and Y velocities of just one sensor of these 4 sensors. As discussed later we use the trace of the CRLB matrix of the geo-location estimate as our objective function, which is shown in Fig. 1 for this case. A similar lack of convexity calculated entropy as an objective function forced [2] to resort to exhaustive search for the best next velocity only on a predefined grid to make the problem manageable in real time.

It has been demonstrated in many cases, those kind of optimization problems are hard to solve using classical derivative-based optimization approaches. Moreover, derivative-based approaches in multimodal problems are

very likely to converge to local minima in which case, the problem would end up with suboptimal solutions. Some of those problems are even not differentiable at certain points, which greatly restricts the usefulness of derivative-based approaches in multi-dimensional and multimodal problems. An obvious optimal scheme is brute force, also known as the grid search approach, but it requires a prohibitive amount of computations, especially in multi-dimensional problems, which is generally infeasible to apply in real time and resource constrained applications.

A common resort for multi-dimensional and multimodal optimization problems is the metaheuristic approaches, e.g. *Genetic Algorithm* (GA), *Ant Colony Optimization* (ACO), and *Particle Swarm Optimization* (PSO) etc. Particle swarm optimization (PSO) was originally proposed by Kennedy and Eberhart in 1995 [3]. It is widely adopted and focused by researchers due to its simple implementation and fast convergence. The original intent was to graphically simulate the graceful but unpredictable choreography of a bird flock. Previous research [4] indicates that PSO is relatively more capable for global exploration and converges faster than most major metaheuristic algorithms.

We test the algorithm to demonstrate the accuracy improvement of the algorithm. Results show that the proposed PSO-ONS (Optimal Next State) algorithm improves the accuracy of the estimation along the platform flying trajectories dramatically. We also show the proposed method is relatively insensitive to the problem dimensionality which indicates the robustness of the algorithm.

The rest of the paper is structured as follows. The location estimation problem is described in Section 2, and the PSO algorithm is briefly introduced in Section 3. Section 4 presents the simulation scenarios and shows the experimental results. Computational complexity issue is discussed in Section 5. Section 6 concludes the paper with comments on future work.

## II. PROBLEM SETUP

Without loss of generality, we assume there are totally  $N_s$  sensors (and assume each UAV is equipped with a single sensor, so we can use the term sensor and aircraft interchangeably), and consider a 2-D Cartesian plane where the true emitter is located at the position  $\{x_e, y_e\}$ . The  $\mathbf{p} = \{\mathbf{p}_1, \mathbf{p}_2, \dots, \mathbf{p}_{N_s}\}$  and  $\mathbf{v} = \{\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_{N_s}\}$  represent the sensor positions and velocities, where  $\mathbf{p}_t = \{x_t, y_t\}$  and  $\mathbf{v}_t = \{v_{xt}, v_{yt}\}$ . Sensors intercept the signal and make TDOA and FDOA measurements at certain instants. At each time interval, a certain amount of thrust is applied on each sensor which is denoted as  $\mathbf{u}_t = \{u_{xt}, u_{yt}\}$ .

### A. Dynamic Model

Here we propose a dynamic model in which the sensor position and velocity update rule is defined. At state  $t + 1$ , the position vector  $\mathbf{p}_{t+1} = \mathbf{p}_t + \mathbf{v}_{t+1} \cdot \Delta t$ , where  $\Delta t$  is the time interval between two adjacent states. From the Newton's Second and Third Law, thrust applied on an aircraft at each state changes the velocity of the aircraft, and the dynamics could be described as  $\mathbf{v}_{t+1} = \mathbf{v}_t + \frac{1}{m} \cdot \mathbf{u}_{t+1} \cdot \Delta t$ , where  $m$  is the mass of the aircraft. Written in the matrix form, the state dynamic model is,

$$\begin{pmatrix} \mathbf{p}_{t+1} \\ \mathbf{v}_{t+1} \end{pmatrix} = \mathbf{A} \begin{pmatrix} \mathbf{p}_t \\ \mathbf{v}_t \end{pmatrix} + \mathbf{B} \mathbf{u}_{t+1} \quad (1)$$

where  $\mathbf{A}$  and  $\mathbf{B}$  are

$$\mathbf{A} = \begin{pmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad \mathbf{B} = \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ \frac{1}{m} & 0 \\ 0 & \frac{1}{m} \end{pmatrix} \quad (2)$$

and  $\|u_t\| \leq u_{max}, \|v\| \leq v_{max}$ .

### B. TDOA/FDOA Estimation

We employ the idea of using TDOA and FDOA to estimate the emitter location [5], in which case sensors are being paired to get TDOA and FDOA measurements. How sensors are paired in order to maximize the accuracy of the estimation is beyond the scope of this paper. Interested readers could refer to [1]. We assume that sensors are paired arbitrarily. Before using TDOA and FDOA measurements to estimate the emitter location, we must first estimate TDOA and FDOA themselves from the noisy signal data. Assume the  $i$ th and the  $j$ th sensor are paired, then the noise free TDOA( $\tau_{ij}$ ) and FDOA( $\omega_{ij}$ ) measurements can be deterministically computed as follows,

$$\tau_{ij} = \frac{1}{c} \left[ \sqrt{(x_i - x_e)^2 + (y_i - y_e)^2} - \sqrt{(x_j - x_e)^2 + (y_j - y_e)^2} \right] \quad (3)$$

$$\omega_{ij} = \frac{f_e}{c} \left[ \frac{(x_i - x_e)v_{xi} + (y_i - y_e)v_{yi}}{\sqrt{(x_i - x_e)^2 + (y_i - y_e)^2}} - \frac{(x_j - x_e)v_{xj} + (y_j - y_e)v_{yj}}{\sqrt{(x_j - x_e)^2 + (y_j - y_e)^2}} \right] \quad (4)$$

where  $c$  is the speed of light.

In the presence of noise, the TDOA and FDOA measurements will be perturbed by error, i.e., for the  $p$ th pair,  $p \in \{1, 2, \dots, N_s/2\}$ ,

$$\hat{\tau}_p = \tau_p + \Delta\tau_p \quad (5)$$

$$\hat{\omega}_p = \omega_p + \Delta\omega_p \quad (6)$$

where  $n$  is the iteration time,  $\mathbf{H} = [\mathbf{H}_1^T, \dots, \mathbf{H}_{N_s/2}^T]$ , and

$$\mathbf{H}_m = \begin{pmatrix} \frac{\partial \hat{\theta}_1}{\partial x_e} & \frac{\partial \hat{\theta}_1}{\partial y_e} \\ \vdots & \vdots \\ \frac{\partial \hat{\theta}_{N_s/2}}{\partial x_e} & \frac{\partial \hat{\theta}_{N_s/2}}{\partial y_e} \end{pmatrix}$$

Define the estimate  $\hat{\theta} = \{\hat{\tau}, \hat{\omega}\}$  which can be obtained using the Maximum Likelihood (ML) estimator [6]. Due to the *Asymptotic property* of the Maximum Likelihood estimator [7], as more data observations are collected, the distribution of the estimate becomes Gaussian, i.e.,  $\hat{\theta}_m = \{\Delta \hat{\tau}_m, \Delta \hat{\omega}_m\}^T \sim \mathcal{N}(\mathbf{0}, \mathbf{F}\mathbf{I}_m^{-1})$ .  $\mathbf{F}\mathbf{I}_m$  is the *Fisher Information Matrix* (FIM) of the estimate  $\hat{\theta}_m = \{\Delta \hat{\tau}_m, \Delta \hat{\omega}_m\}^T$  of the  $m$ th pair, defined as,

$$\mathbf{F}\mathbf{I}_m = 2Re \left[ \frac{\partial \vec{s}_m^H(\vec{\theta}_m)}{\partial \vec{\theta}_m} \mathbf{C}_m^{-1} \frac{\partial \vec{s}_m(\vec{\theta}_m)}{\partial \vec{\theta}_m} \right] \quad (7)$$

where  $\mathbf{C}_m$  is the covariance matrix of the AWGN among two sensors in the  $m$ th pair,  $\mathbf{s}_m$  denotes the received signal, and  $\vec{\theta}_m$  is the actual TDOA and FDOA measurements. However, the result given in (8) requires knowledge of the signal and analytical results for the derivatives, [8] has shown that (8) can be computed numerically using the data measurements.

We define the Fisher information matrix for the estimator  $[\hat{\theta}_1, \hat{\theta}_2, \dots, \hat{\theta}_{N_s/2}]$  as follows,

$$\mathbf{J}_\theta = \begin{pmatrix} \mathbf{F}\mathbf{I}_1 & \mathbf{I}_{12} & \cdots & \mathbf{I}_{1N_s/2} \\ \mathbf{I}_{21} & \mathbf{F}\mathbf{I}_2 & \cdots & \mathbf{I}_{2N_s/2} \\ \vdots & \ddots & \ddots & \vdots \\ \mathbf{I}_{N_s/21} & \mathbf{I}_{N_s/22} & \cdots & \mathbf{F}\mathbf{I}_{N_s/2} \end{pmatrix} \quad (8)$$

where  $\mathbf{I}_{ij}$ ,  $i \neq j \in \{1, 2, \dots, N_s/2\}$  is the cross term FIM between  $i$ th and  $j$ th pair. Note that we assume the sensors are paired arbitrarily and no sharing among pairs, hence the  $\mathbf{I}_{ij}$ ,  $i \neq j \in \{1, 2, \dots, N_s/2\}$  equal zero.

### C. Emitter Geo-Location Estimation

After TDOA and FDOA are estimated by the Maximum Likelihood estimator, those TDOA and FDOA estimates are further used to estimate the emitter geo-location. From the asymptotic assumption of the TDOA/FDOA MLE, we can take the TDOA/FDOA estimates as Gaussian and derive the MLE for the emitter geo-location, which is a Weighted Least Squares estimator. It's clear that in (3) and (4), TDOA and FDOA measurements are nonlinear in the emitter location parameters  $([x_e, y_e])$  we wish to estimate, therefore there is no closed form solution [7]. The Nonlinear Least-Squares with Gauss-Newton method is applied to estimate the emitter location.

Using the iterative Nonlinear Least-Squares algorithm, the estimate in each iteration is represented as below [9] [7].

$$\hat{\mathbf{x}}_{n+1} = \hat{\mathbf{x}}_n + (\mathbf{H}^T \mathbf{J}_\theta \mathbf{H})^{-1} \mathbf{H}^T \mathbf{J}_\theta \Delta \mathbf{f}(\hat{\mathbf{x}}_n) \quad (9)$$

is the Jacobian matrix of the  $m$ th pair of sensors, and  $T$  denotes the transpose.  $\Delta \mathbf{f}(\hat{\mathbf{x}}_n) = \mathbf{H}[\mathbf{x} - \hat{\mathbf{x}}_n] + \mathbf{v}$  is the prediction error due to the Taylor series approximation when applying Gauss-Newton method. And the *Crámer-Rao Lower Bound* of this problem can be represented as,

$$\text{var}\{\mathbf{x}\} \geq (\mathbf{H}^T \mathbf{J}_\theta \mathbf{H})^{-1} \quad (10)$$

We could draw an ellipsoid of the CRLB matrix, the area of which corresponds to the covariance of the estimation. Our goal is to find the optimal next state for each sensor based on the current received signal, the current geo-location estimate and sensor nav data, i.e. the estimation accuracy at those optimal next states is maximized, but subject to the vehicle velocity and thrust constraint. To achieve this goal, technically the CRLB of the emitter location estimator should be minimized. Therefore, the optimization problem could be stated as,

$$\arg \min_{\mathbf{u}} \det \left[ \left( \sum_{i=1}^k \mathbf{H}_i^T \mathbf{J}_\theta \mathbf{H}_i \right)^{-1} \right] \quad (11)$$

*s.t.*  $\|\mathbf{u}_i\| \leq u_{max}$

In (11), we use the determinant of the CRLB matrix as the evaluation criterion for estimation accuracy. However, [10] has shown that using the determinant as a single-value criterion has several drawbacks which may limit it to be a suboptimal solution in some case. The authors have also shown that the trace of the CRLB matrix is a good alternative to the determinant, and at the same time, mitigates major drawbacks of the determinant operator. Here, we accept the claim and use the trace instead of the determinant as the single-value evaluation of the estimation accuracy.

According to the dynamic model defined earlier, sensor positions can be calculated using  $\mathbf{p}_{t+1} = \mathbf{p}_t + \mathbf{v}_{t+1} \cdot \Delta t$ , so actually we only need to optimize over sensor velocities which only depend on the initial velocities and thrusts applied in each state. In the 2-D case for example, if we have  $N_s = 10$  sensors, the number of variables to be optimized over is  $N_s \times 2 = 20$ , which is fairly large. Moreover, it's also clear that (11) is not a convex optimization problem. Performing classical optimization approaches even on resources-limited UAVs is computationally expensive. In the next section, we briefly discuss a metaheuristic method called the *Particle*

*Swarm Optimization* to cope with this multivariate multimodal optimization problem.

### III. PARTICLE SWARM OPTIMIZATION

Particle Swarm Optimization (PSO) is a population-based evolutionary algorithm which is generally inspired by biological evolution behavior. Similar to GA, PSO is also an evolutionary algorithm based on swarm intelligence. But, unlike GA, PSO has no explicit evolution operator such as crossover and mutation, instead of simple operators including additions and multiplications in its iteration equations, which simplifies the algorithm and makes the algorithm less computational intensive.

In PSO, the potential solutions, called particles, fly through the solution space by following the current optimal particles. Particles in the entire swarm are initialized randomly in the entire  $D$ -dimensional solution space. During a run, particles update their velocities and positions according to the best solutions found so far by themselves and the entire group jointly. Since the algorithm was first reported in 1995 [3], hundreds of PSO variants have been proposed in the literature, and have been successfully examined in real world applications. Comparison results have shown that most PSO variants present similar results in the early part of their convergences. In real time applications like the optimal next state, reaching a reasonable solution while occupying limited amount of time and computational power should be given the highest priority. Neither large number of fitness function evaluations nor additional complex operation is feasible. Therefore, the PSO algorithm with inertia weight [11] is employed in the present paper, which can be represented mathematically as follows,

$$\begin{aligned} v_{id}(t+1) &= w(t) \times v_{id}(t) + c_1 \times rand_1(\cdot) \\ &\quad \times (p_{id} - x_{id}) + c_2 \times rand_2(\cdot) \\ &\quad \times (p_{gd} - x_{id}) \end{aligned} \quad (12)$$

$$x_{id}(t+1) = x_{id}(t) + v_{id}(t+1) \quad (13)$$

$$1 \leq i \leq N, 1 \leq d \leq D$$

where  $N$  is the number of particles,  $D$  denotes the dimensionality;  $\mathbf{V}_i = (v_{i1}, v_{i2}, \dots, v_{iD})$ ,  $v_{id} \in [-v_{max}, v_{max}]$  is the velocity vector of the particle  $i$ . Similarly,  $\mathbf{X}_i = (x_{i1}, x_{i2}, \dots, x_{iD})$ ,  $x_{id} \in [-x_{max}, x_{max}]$  is the position vector of the particle  $i$  which is a potential solution in the solution space. The quality of the solution is measured by the value of a fitness function (commonly the objective function);  $w(t)$  is the *inertia weight* decreasing linearly from 0.9 to 0.2 during a run;  $c_1, c_2$  are positive

constants, called the *acceleration factors* generally set to 2.0 [11];  $rand_1(\cdot)$  and  $rand_2(\cdot)$  are random numbers distributed uniformly over the range  $[0, 1]$ ; and  $\mathbf{p}_g$ ,  $\mathbf{p}_i$  are the best solutions discovered so far by the group and the  $i$ th particle, respectively.

In the  $t+1$  step of iteration, particle  $i$  uses  $p_g$  and  $p_i$  as the heuristic information to update its own velocity and position. The first term in (12) could be viewed as the diversification term, while the second and third are the intensifications, which could also be seen as the trustworthiness towards itself and the entire social system, respectively. Therefore, group diversification and individual intensification are carefully balanced.

### IV. SIMULATION RESULTS

To show how PSO-ONS algorithm improves the system's estimation accuracy along the optimal trajectories, we apply the algorithm in each intermediate state to find the optimal next state. In order to give quantitative result of estimation accuracy improvement, emitter location estimation procedure is performed in each of those optimal intermediate states. For comparison purposes we also show results for two other schemes: random walk and constant velocity. In the random walk scheme the next optimal state is chosen randomly within the reachable set and in the constant velocity scheme the sensors continue to each next state along a constant velocity path. The random walk attempts to achieve geometric diversity through random selections; the constant velocity approach mimics the naive idea of simply letting the sensors continue to fly as they were originally.

In our experiments, we consider 6 sensors distributed randomly over the space and are paired arbitrarily into three disjoint pairs. Trajectories of which are optimized over all 6 sensors in the PSO-ONS scenario. We assume the received signal-to-noise ratio varies as in free-space propagation scenario [12]; the emitter signal frequency is 10GHz; the total number of intermediate states for each aircraft is 20; time between each intermediate state is  $\Delta t = 5s$ ; and platform velocity is bounded  $\|v\| \leq 100m/s$  and the applied thrust  $\|u\| \leq 1000$  Newton, which is a reasonable assumption for UAVs [13]. The size of the particle swarm is 5, and we do 20 iterations in each PSO procedure. Emitter true location is  $[40000/\sqrt{2}, 40000/\sqrt{2}]$ . Experimental result is shown in Fig.2.

Fig.2 demonstrates the comparison of estimation error between the PSO-ONS, the random walk and the constant velocity schemes. Same set of sensor states are initialized for all three cases, and therefore they have the same initial accuracy. We can see that the estimation accuracy is decreasing steadily along the flying trajectory while the other two suffer performance fluctuation due to the multimodal

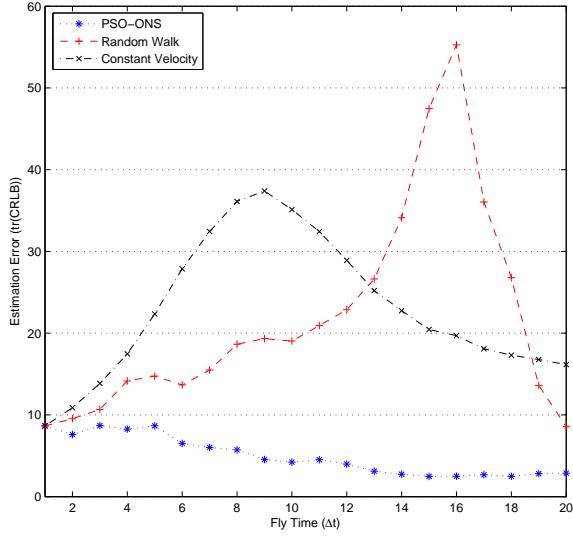


Fig. 2. Estimation Accuracy Improvements

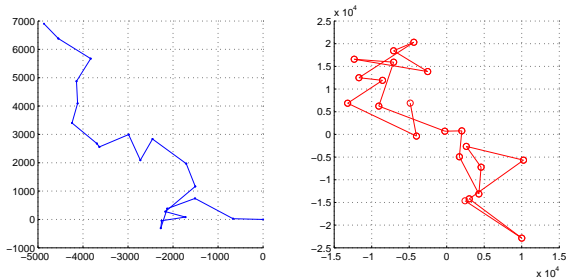


Fig. 3. Optimal and Random Thrust on Vector Field

characteristics of the problem.

A sequence of the PSO-ONS optimized thrust applied on a single platform is plotted in 2-D vector field, while a sequence of random thrust is also shown in Fig.3. As random thrusts drive the platform to move unintentionally over the whole space, the platform moves towards a clear direction along which more accurate estimation can be achieved. Moreover, a wiggling behavior of the platform is clearly seen along the optimal trajectory which allows the platform to collect data which contains more differences in time-of-arrival and frequency-of-arrival. This kind of movement is desirable and beneficial to the location estimation in general.

Another advantage of the proposed PSO-ONS algorithm is that the performance of PSO-ONS is insensitive to the increasing problem dimensionality which shows the algorithm is stable and expandable. We define the success rate as the number of times PSO-ONS gets better estimation accuracy than the random walk over the total number of simulation runs. As we can see from Fig.4, as the number of sensors increases, the dimension of the optimization problem increases

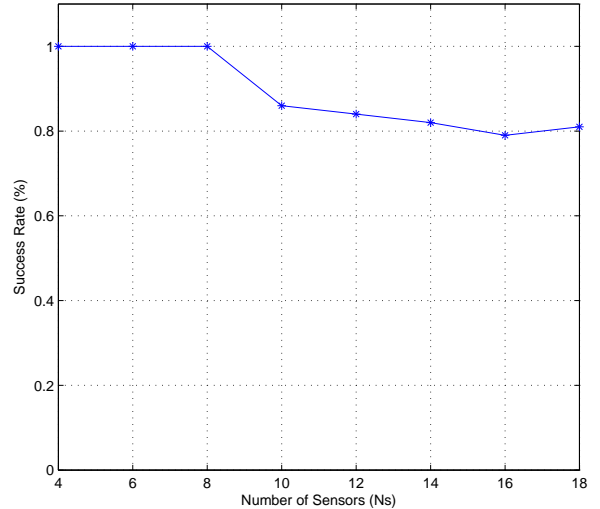


Fig. 4. Sensitivity to the Problem Dimensionality

correspondingly. However, the success rate doesn't change drastically. When  $N_s = 18$ , the dimensions of the problem is  $18 \times 2 = 36$  which is prohibitive for the grid search scheme or even classical optimization methods.

#### V. COMPUTATIONAL COMPLEXITY ANALYSIS

Simulation results above have provided strong evidence that the proposed PSO-ONS algorithm achieves accurate estimation. It could be also demonstrated that the proposed method is computationally efficient, compared to some straightforward brute force approaches, such as the grid search scheme. Consider a scenario where 4 aircrafts are deployed to perform the emitter location estimation. We have already shown this optimization problem is 8 dimensional, and further consider the vehicle thrust constraint is  $-100 \sim 100$  Newton, and the step length in the grid search is 5m/s, the number of grid point of a single next optimal next state computation would be  $40^8 = 6.5 \times 10^{12}$  which is prohibitively huge and not feasible to compute on a resource-constrained real-time system. Even with a relatively coarse grid with step length 20m/s, the number of grid point would still be  $10^8$ .

On the contrary, the random walk and the constant velocity schemes are the least computational expensive approaches. Moreover, for the constant velocity case, no thrust needs to be applied therefore it is the most energy efficient. If the measurement of computational cost is the number of evaluations of the trace of CRLB matrix, it only requires a single function evaluation in each next state. However, in terms of estimation accuracy, the random walk and constant velocity schemes, as shown in Fig.2 suffer great fluctuations and in general are less accurate than the proposed method.

The newly proposed PSO-ONS algorithm, while achieving improved estimation accuracy, is also efficient in computations

and simple in structure, which make the proposed algorithm applicable for real-world emitter location estimation problems. In this paper, where we have 5 particles, each of which iterates 20 times, the total number of computations within each optimal next state calculation is 100, which is drastically smaller than the grid search.

## VI. DISCUSSION AND FUTURE WORK

This paper proposed a computationally efficient way to determine optimal next states for UAVs to perform accurate emitter geo-location estimation. Due to the multi-dimensional and multimodal nature of the optimal next state problem, we incorporated a widely-researched metaheuristic method called the Particle Swarm Optimization (PSO) to attack the problem. Simulation results demonstrate PSO is a promising approach, in terms of high estimation accuracy and low computational cost.

An issue worthwhile discussing is the communication cost. The PSO optimization procedure doesn't really introduce much extra communication cost. The only information needed to be shared among sensors is their position and velocities, none of the actual data they collected needed to be transmitted over the wireless channel. However the sensor network may perform other estimation or tracking applications at the same time, which may also need that information as well. An example is to perform the optimal sensor selection and pairing [1]. In such cases, only the optimized sensor positions and velocities need to be sent back to sensors, which means even less communication overhead. Note that the PSO optimization we employ is a centralized algorithm which needs a "central node" to perform the optimization. This central node might be a powered node far away from the sensors. A more realistic situation is that the central node is selected among the sensors themselves. For robustness considerations, a dynamic selection scheme, based on the remaining energy of the sensors [14], might be applicable.

## REFERENCES

- [1] X. Hu and M. L. Fowler, "Sensor selection for multiple sensor emitter location systems," in *Proc. IEEE Aerospace Conference'08*, Big Sky, Nevada, Mar. 2008.
- [2] N. E. Wu, Y. Guo, K. Huang, M. C. Ruschmann, and M. L. Fowler, "Fault-tolerant guidance and control of an airborne location sensor network," *Special Section on Fault Diagnosis and Fault-Tolerant Systems, International Journal of Control, Automation, and Systems*, vol. 6, pp. 351–363, 2008.
- [3] J. Kennedy and R. C. Eberhart, "Particle swarm optimization," in *Proc. IEEE International Conference on Neural Networks(ICNN'95)*, Perth, Australia.
- [4] R. C. Eberhart and Y. H. Shi, "Particle swarm optimization: developments, applications and resources," in *Proc. IEEE Congress on Evolutionary Computation(CEC'01)*, Piscataway, NJ.
- [5] P. C. Chestnut, "Emitter location accuracy using tdoa and differential doppler," *IEEE Trans. Aerosp. Electron. Syst.*, vol. AES-18, pp. 214–218, Mar. 1982.
- [6] S. Stein, "Differential delay/doppler ml estimation with unknown signals," *IEEE Trans. Signal Process.*, vol. 41, pp. 2717–2719, Aug. 1993.

- [7] S. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*, Prentice Hall, 1993.
- [8] M. L. Fowler and M. Chen, "Fisher-information-based data compression for estimation using two sensors," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 41, pp. 1131–1137, July 2005.
- [9] M. L. Fowler, "Radar location via frequency measurements: An application of nonlinear least squares," 2002.
- [10] M. Chen and M. L. Fowler, "Data compression for multiple parameter estimation with application to emitter location systems," *IEEE Trans. Aerosp. Electron. Syst.*, submitted for publication.
- [11] Y. H. Shi and R. Eberhart, "Empirical study of particle swarm optimization," in *Proc. IEEE Congr. on Evolutionary Computation(CEC'99)*.
- [12] A. Goldsmith, *Wireless Communications*, Cambridge University Press, Cambridge, U.K, 2005.
- [13] T. Samad, J. S. Bay, and D. Godbole, "Network-centric systems for military operations in urban terrain: The role of uavs," *Proceedings of the IEEE*, vol. 95, pp. 92–107, Jan. 2007.
- [14] W. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "An application-specific protocol architecture for wireless microsensor networks," *IEEE Trans. Wireless Commun.*, vol. 1, pp. 660–670, Oct. 2002.