



# An interval-based target tracking approach for range-only multistatic radar

G.L. Soares<sup>\*†</sup>, A. Arnold-Bos<sup>\*</sup>, L. Jaulin<sup>\*</sup>, J.A. Vasconcelos<sup>†</sup> and C.A. Maia<sup>†</sup>

<sup>†</sup> UFMG - Av. Antônio Carlos 6627, Belo Horizonte-MG, Brazil, CEP 31270-901

<sup>\*</sup> ENSIETA - 2, Rue François Verny, 29806 Brest Cedex 9 - France

{gsoares, joao, maia}@cpdee.ufmg.br<sup>†</sup> and {arnoldan, jaulinlu}@ensieta.fr<sup>\*</sup>

**Abstract**—This paper investigates the use of interval analysis to solve the problem of maneuvering target tracking, using range-only measures collected by a multistatic radar. The problem consists in one transmitter, and some receivers working together as a multistatic radar. The radar process is plagued by several uncertainty sources that affect directly the receivers' measures. As a result, target tracking can be both imprecise and unreliable. This study presents the *Tracking using an Interval-Based Approach (TIBA)* that computes the set of all feasible configurations for the target which are consistent with the measures. The algorithm is compared to a conventional tracking method: particle filtering.

**Index Terms**—target tracking; manoeuvring target; multistatic radar; interval analysis; interval methods.

## I. INTRODUCTION

Radar systems have been used in several fields [1] as in airspace monitoring, marine surveillance, weather prediction, and ground imaging. Most systems today are monostatic, that is, the transmitter and the receiver share the same antenna. There is, however, an increasing interest in bistatic [2] and more generally multistatic systems [3], where one or several receivers are positioned elsewhere than the receiver.

We consider in this paper a simple, theoretical multistatic radar system which returns range-only measures. The true position of the target can be found by the intersection of the Fresnel ellipsoids corresponding to the distances measured by the radars. However, these measures can be corrupted by noise, and the probable position of the target becomes harder to determine: instead of a point, the target could lie in a much larger region. The use of several, consecutive measures, coupled to a tracking algorithm, can reduce this uncertainty on the target's position. However, the mapping from observation (that is, distances) to the Cartesian target's coordinates is non-linear; similarly, the noise has no reason to be Gaussian. Thus methods based on the Kalman filter and its derivatives are non-robust, especially in the presence of outliers.

In this paper, we propose to track targets using an interval analysis approach (TIBA). Interval analysis has begun with Moore [4] for the treatment of the rounding errors in numerical computation and it has been efficiently employed to solve problems with uncertainty parameters [5] and [6]. In interval analysis approaches, sets are used to represent punctual numbers and their uncertainties. Then, a safe manipulation of these sets results in the guaranteed interval where the true punctual

solution is. However, Interval Analysis suffers from some criticisms, as the (relatively) slowness of their implementation, for instance. Hansen and Walster [7] refuted most of these and showed the advantages of the interval approaches as for example: the possibility to obtain the solution of certain problems that can not be solved by non-interval methods, the convergence of their algorithms and the reliability of their results. These facts have motivated the use of the interval methodology to solve the tracking problem.

## II. DESCRIPTION OF THE PROBLEM

Consider a target  $T$  traveling in the 2D space. This target can be represented by the following state vector:

$$\mathbf{X}_n = [x_n, y_n, \dot{x}_n, \dot{y}_n]^t \quad (1)$$

where  $x_n$  and  $y_n$  are the target's Cartesian coordinates, and  $\dot{x}_n, \dot{y}_n$  their derivatives with respect to time. The evolution of the target is described by a state evolution equation:

$$\mathbf{X}_{n+1} = f(\mathbf{X}_n) + \mathbf{V}_n \quad (2)$$

where  $f$  is a deterministic state evolution function, and  $\mathbf{V}_n$  is a random vector representing unpredictable changes in the target's motion. Denoting the time between the states  $n$  and  $n + 1$  by  $\Delta t$ , a simple approximation for  $f$  and  $\mathbf{V}_n$  can be:

$$\mathbf{X}_{n+1} = \underbrace{\mathbf{A}\mathbf{X}_n}_{f(\mathbf{X}_n)} + \underbrace{\mathbf{B}\mathbf{N}_n}_{\mathbf{V}_n} \quad (3)$$

where matrices  $\mathbf{A}$  and  $\mathbf{B}$  are given by:

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} \frac{\Delta t^2}{2} & 0 \\ 0 & \frac{\Delta t^2}{2} \\ \Delta t & 0 \\ 0 & \Delta t \end{bmatrix} \quad (4)$$

respectively, and  $\mathbf{N}_n$  is a vector of dimensions  $2 \times 1$  with entries have mean zero and standard deviation  $\sigma_x$  to horizontal axis, and  $\sigma_y$  to vertical axis.

The multistatic radar system consists in one transmitter  $E$  and a series of receivers  $R_i$ ,  $i = 1, \dots, l$  which are not positioned at the same place. Each couple  $B_i = (E, R_i)$  forms a bistatic radar and the whole set  $(B_1, \dots, B_n)$  is the multistatic radar mentioned earlier. We consider in our simulations that  $R_1$  shares the same antenna as  $E$ , thus  $B_1$  is really only a monostatic radar. An example of the multistatic radar is presented in Fig. 1. Each bistatic radar provides outputs  $r_n^i$  corresponding to the total distance traveled by the

This work was in part sponsored by CAPES, CNPq, FAPEMIG and PUC Minas from Brazil, ENSIETA and UBO from France and the regional council of Brittany.

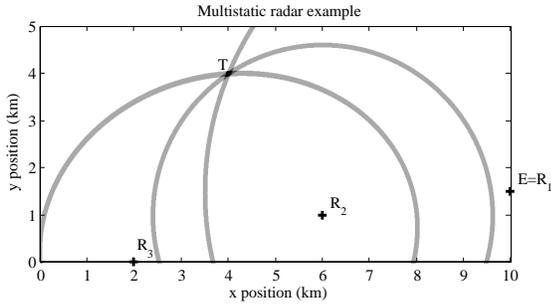


Fig. 1. Multistatic problem with three receivers as used in this paper. The isorange ellipses (in gray) represent the target location by each receiver, and their intersection (in black), the true target location (4,4). Note that the range resolution is 100 m in this figure.

radio wave from  $E$ , to  $T$ , then to  $R_i$ . Hence, the target position is the solution of the following system of equations:

$$r_n^i = d_{ET} + d_{TR_i}, \quad i = 1, \dots, l \quad (5)$$

where  $d_{ET}$  and  $d_{TR_i}$  denote the distance from the transmitter to the target, and the target to the  $i$ -th receiver. The output of the multistatic radar can be stored in an observation vector:

$$\mathbf{Y}_n = [r_n^1, \dots, r_n^l]^t \quad (6)$$

Beyond the uncertainty sources listed before, the radar system can face other hard problems. For instance, noise might be mistaken for the target; this could be a strong spike coming from the scene clutter, or a strong return in the secondary lobe of the antenna. In these cases, the measurement process yields an outlier. There could also be missing measures, if no echo significantly arises above the noise level. Thus, the measurement system is also unreliable, and the tracking system must be able to cope with that.

### III. TRACKING USING INTERVALS

Usually, the traditional filtering algorithms work within the probabilistic framework. These algorithms assume that the noise (or other uncertainties) follow a probability distribution. Without being exhaustive, the best known algorithms are the Kalman filter and its numerous variants (among which the most celebrated one today is probably the Unscented Kalman filter [8]); or the particle filter [9]. Their goal is to estimate the probability density function (pdf) of  $\mathbf{X}_n$  given the observations  $\mathbf{Y}_1, \dots, \mathbf{Y}_n$ . A review of the applicability of these methods for range-only, multistatic radar tracking may be found in [3].

In the interval approach presented in this section, the uncertainty sources are not modeled by pdf's but strictly bounded by intervals. Intervals have a specific notation which are described below. Then TIBA itself is presented.

#### A. Interval Analysis

**Definition 1** A real, closed and connected interval  $[z]$  is defined by:

$$[\underline{z}, \bar{z}] = \{z \in \mathbb{R} \mid \underline{z} \leq z \leq \bar{z}\}, \quad (7)$$

where  $\underline{z}$  and  $\bar{z}$  represent, respectively, the lower and the upper bounds of  $[z]$ . The domain of all real numbers intervals is  $\mathbb{IR}$ .

**Definition 2** Denote  $\diamond$  as binary operator to represent the four classical operations of real arithmetic  $+$ ,  $-$ ,  $*$  and  $/$ . Then:

$$[z] \diamond [w] = \{z \diamond w \mid z \in [z], w \in [w]\} \quad (8)$$

**Definition 3** A box  $[\mathbf{z}]$  is an interval vector in  $\mathbb{IR}^n$  defined as the Cartesian product of  $n$  closed intervals:

$$[\underline{\mathbf{z}}, \bar{\mathbf{z}}] = [z_1] \times [z_2] \times \dots \times [z_n], \quad (9)$$

where  $\underline{\mathbf{z}}$  and  $\bar{\mathbf{z}}$  are the lower and the upper bounds of  $[\mathbf{z}]$ .

**Definition 4** The center  $c([\mathbf{z}]) \in \mathbb{R}^n$  and the width  $w([\mathbf{z}]) \in \mathbb{R}$  are defined by:

$$c_i([\mathbf{z}]) = (\bar{z}_i + z_i)/2, \quad i = 1, \dots, n \quad (10)$$

$$w([\mathbf{z}]) = \max_i (\bar{z}_i - z_i), \quad i \in \{1, \dots, n\} \quad (11)$$

**Definition 5** Denote the Euclidean distance from  $\mathbf{p} \in \mathbb{R}^n$  to  $\mathbf{z} \in \mathbb{R}^n$  by  $d_{\mathbf{p}\mathbf{z}}$ . Then, the interval distance  $[d]_{\mathbf{p}[\mathbf{z}]} \in \mathbb{IR}^n$  between a point  $\mathbf{p}$  and an interval  $[\mathbf{z}] \in \mathbb{IR}^n$  is:

$$[d]_{\mathbf{p}[\mathbf{z}]} = [\min(d_{\mathbf{p}\mathbf{z}}, \max(d_{\mathbf{p}\mathbf{z}})], \quad \forall \mathbf{z} \in [\mathbf{z}] \quad (12)$$

**Definition 6** Consider the intervals  $[z] \in \mathbb{IR}$  and  $[w] \in \mathbb{IR}$ . The interval displacement  $[\Delta d]_{[z][w]} \in \mathbb{IR}$  is given by:

$$[\Delta d]_{[z][w]} = [w] - [z] \quad (13)$$

**Definition 7** Consider the set  $[\mathbf{Z}] = \{[z]_1, \dots, [z]_m\} \subseteq \mathbb{IR}^n$ . The box of narrowest width containing all elements of  $[\mathbf{Z}]$  is:

$$\begin{aligned} \Phi([\mathbf{Z}]) &= [Z_1] \times [Z_2] \times \dots \times [Z_n], \quad (14) \\ [Z_i] &= [\min_{j=1, \dots, m} (z_{ij}), \max_{j=1, \dots, m} (\bar{z}_{ij})], \quad i = 1, \dots, n \end{aligned}$$

**Definition 8** An operation that divides a box  $[z]$  in two off-spring boxes  $[z]_1$  and  $[z]_2$  of the same size is called bisection.

**Definition 9** Consider the functions  $\mathbf{f}$  from  $\mathbb{R}^n$  to  $\mathbb{R}^m$  and  $[\mathbf{f}]$  from  $\mathbb{IR}^n$  to  $\mathbb{IR}^m$ . The  $[\mathbf{f}]$  is an inclusion function for  $\mathbf{f}$  if:

$$\forall [\mathbf{z}] \in \mathbb{IR}^n, \mathbf{f}([\mathbf{z}]) \subset [\mathbf{f}]([\mathbf{z}]) \quad (15)$$

#### B. SIVIA

The Set Inverter Via Interval Analysis (SIVIA) method [5] searches the preimage of a given set under a function. Its inputs are a search interval  $[z]$ , an image box  $[\mathbf{r}]$  and the accuracy parameter  $\varepsilon$ . SIVIA bisects  $[z]$  until it classifies its offsprings. The typical outputs are a set composed by *solution boxes*  $[\mathbf{Z}]_s$  that belong to the preimage of  $[\mathbf{r}]$ , a set with *non-solution boxes*  $[\mathbf{Z}]_n$ , and a set that contains *boundary boxes*  $[\mathbf{Z}]_b$ . In our case, consider what the radar system provides to SIVIA: a) the range values  $r_n^i$ ; b) the transmitter and receivers position,  $E$  and  $R_i$ ; c) the maximal uncertainty on the measurement  $\varepsilon$ ; and d) the search box  $[z]$ . Then, SIVIA is used to solve the following equation:

$$[r_n^i - \varepsilon, r_n^i + \varepsilon] = [d]_{E[\mathbf{z}]} + [d]_{[\mathbf{z}]R_i}, \quad i = 1, \dots, l \quad (16)$$

For instance, Fig. 2 presents SIVIA's output for the radar configuration in Fig. 1. We implemented SIVIA to return the

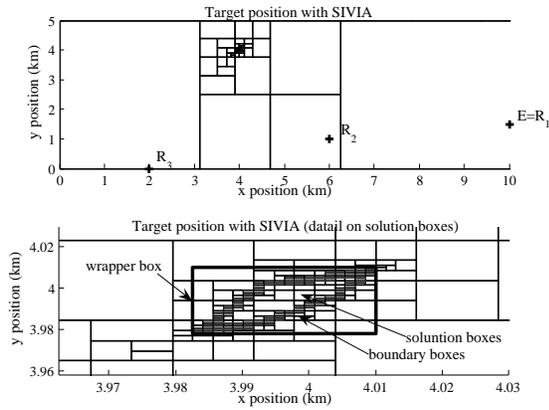


Fig. 2. Typical SIVIA graphics. Above, the results are plotted at the same coordinates that in Fig. 1. Below, a zoom on the region where the solution lies. (CPU time= 0.7s;  $[\mathbf{z}] = [0, 25]km \times [0, 10]km$ ;  $\varepsilon = 2m$ ,  $\epsilon = 12m$ )

wrapping box (or target box) given by  $\Phi([\mathbf{Z}]_s \cup [\mathbf{Z}]_b)$ . Computing the wrapping box is faster than computing separately each small box, however it is less precise.

In the presence of outliers, (16) can yield no solution. Thus, SIVIA has been adapted to use the outlier minimal number estimator (OMNE) [10], to find the solutions consistent with at least two measurements.

### C. TIBA

TIBA is a heuristic technique addressing target tracking problems. It is an interval-based method, hence the variables and the equations have to be adapted to intervals. For instance, the state vector (1) lies in the following box:

$$[\mathbf{X}]_n = [[x]_n, [y]_n, [\dot{x}]_n, [\dot{y}]_n]^t \quad (17)$$

Now, to present how TIBA works, consider a radar system composed by a transmitter and three receivers as in the configuration in Fig. 1. TIBA assumes that the following information is available: a) the location of the radar equipments  $E$  and  $R_i$ ; b) the range of the radar system  $[\mathbf{X}]_0$ ; c) the maximum deviation from receiver's measure  $\epsilon$ ; d) the interval noise  $[\mathbf{V}]_n$ ; and e) the delay between observations  $\Delta t$ . Then, when an object is detected, an iteration of TIBA is done. A block diagram of TIBA is presented in Fig. 3. In the *initialization* step, two consecutive target positions  $[\mathbf{X}]_1$  and  $[\mathbf{X}]_2$  are computed by SIVIA. The displacement  $[\Delta d]_{[\mathbf{X}]_1[\mathbf{X}]_2}$  (only cartesian components) by  $\Delta t$  gives the speed components to  $[\mathbf{X}]_2$ , that is sent to next step. Henceforth  $[\mathbf{X}]_2$  will be known as  $[\mathbf{X}]_n$ . In the *prediction* stage, the system tries to guess the next target's position by using (3). The result is denoted  $[\mathbf{X}]_{n+1|n}$ . After that,  $[\mathbf{X}]_{n+1|n}$  is evaluated by the expression in (5), represented in the block diagram by function  $[\mathbf{h}]$ , resulting in a *hypothetical interval observation vector*  $[\tilde{\mathbf{Y}}]_{n+1}$ , that is input to *correction* step. The correction consist in the intersection between  $[\tilde{\mathbf{Y}}]_{n+1}$  with the observation  $[\mathbf{Y}]_{n+1} = [\mathbf{Y}_n - \epsilon, \mathbf{Y}_n + \epsilon]$ . When there is no intersection, TIBA discards the observation. The *updated interval observation vector*  $[\hat{\mathbf{Y}}]_{n+1}$  is then fed to SIVIA to find the target box to (16). After that, the target box  $[\mathbf{X}]_{n+1}$  is

sent to the prediction step. The algorithms loops until receivers stop their observations.

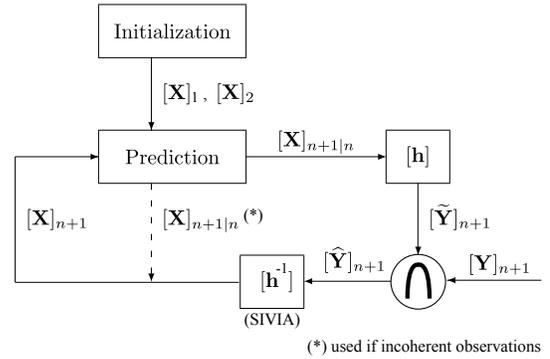


Fig. 3. TIBA's block diagram. Note that the correction step is implemented in the observation space.

## IV. COMPARISON AND EXPERIMENTS

The experiment consists in the comparison of Particle Filtering (PF) [9] and the TIBA when they track a hypothetical target moving in the 2D space. In this example, we consider the receiver's position and initial location of the target are the same, as showed in Fig. 1. The trajectory was created by using (3) with  $\Delta t = 0.1$  s, and  $\mathbf{W}_n$  has zero mean and standard deviation  $100 \text{ m}\cdot\text{s}^{-2}$ . The observations were obtained by the following equation:

$$\mathbf{Y}_n = \mathbf{A}\mathbf{X}_n + \mathbf{W}_n, \quad (18)$$

where the random vector  $\mathbf{W}_n$  is a mix of Gaussian-distributed observations with zero mean and standard deviation  $\sigma_r = 4$  m, intertwined with outliers and missing measures happening with a 10% probability. In the end, 1000 observations were processed. The graphical results present the guessed positions and the error with the real target positions. The methods were implemented in MATLAB and they were executed on a Pentium-4 CPU, 3 GHz, 1024 MB RAM, using Windows XP SP2. The total tracking time for PF was about 1 s per iteration for 1000 particles against 0.1 – 35 s for TIBA. The re-sampling of the particles was done only based on the range observations: for each particle, the likelihood of each observation is computed (for each radar); then all observations are combined by using the max operator. Some regularization is needed, especially on the particle speeds; otherwise, the PF fails to converge. Particle filtering has others probabilistic parameters that requires a lot of tuning, which is not needed by TIBA. Finally, particle filtering is a Monte Carlo method; whereas TIBA is deterministic.

Fig. 4 shows the output of TIBA. It is clear that some target boxes are far wider than many others in their neighborhood. In these cases, multiple outliers and/or measure missing are present simultaneously and there is no consistent solution with the receiver's observations. Thus, TIBA returns the prediction box. The prediction boxes can become wider when outliers and/or measure missing happen in consecutive measurements. In all observations, the prediction boxes have enveloped the next target position safely, even in the presence of the events

above. However, sometimes the outliers perturbation yielded measurements inside the box prediction, and TIBA encloses a false solution, as showed in the detail of Fig. 4. This problem could probably be solved by using more receivers.

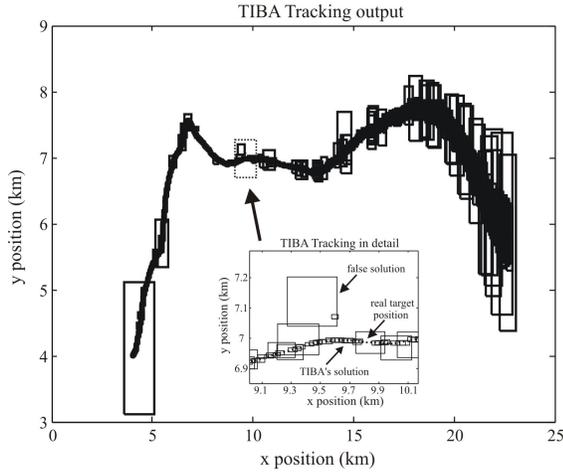


Fig. 4. Full TIBA tracking output. The target boxes, false solution boxes and the real target location are in the detail.

In Fig. 5, the centers of the target boxes are presented with the true target trajectory. Although these centers are considered as a punctual estimation of the target's position, the true TIBA solution is still the target box. The details on Fig. 5 present the quality of the estimation in two parts of the trajectory with the same altitude. The relative position between the receivers and the target causes the ellipsoid's intersections to be smaller in region (a) than region (b). The boxes are wider for the worse ellipsoids' intersections.

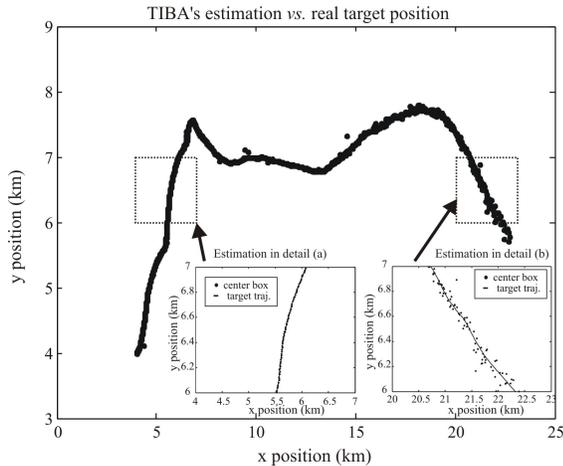


Fig. 5. TIBA's estimation on the target location. The quality of the TIBA's estimation depends on the quality of the isorange ellipses intersections.

Fig. 6 presents the results of the PF tracking for the same trajectory with 1000 particles (which gave better results than 2000 particles or 500 particles). The error plotted here is at  $3\sigma_e = 3\sqrt{\sigma_1^2 + \sigma_2^2}$ , where  $\sigma_1$  and  $\sigma_2$  are the diagonal components of the covariance matrix of the particles cloud. It appears clearly that, globally, PF is outperformed by TIBA.

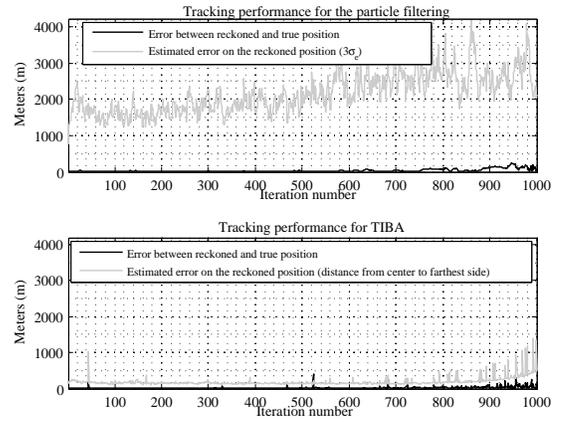


Fig. 6. Comparison between particle filtering (1000 particles) and TIBA in terms of precision and robustness.

## V. CONCLUSIONS AND FUTURE OUTLOOKS

Real time applications like radar tracking need methods giving reliable results using the allotted CPU time. After several tests, we found PF faster than TIBA. For PF, the tracking process is independent of the transmitter-target-receivers positions, otherwise TIBA is dependent. Contrarily to PF, TIBA has fewer tuning parameters. Also, TIBA always converges, even when there are only few consistent measures. TIBA was still reasonably fast (0.4 – 3.2 s by iteration) to envelope the target trajectory if the outliers and the missing measures were not present. Nevertheless, when these events were present simultaneously and consecutively, TIBA spent much more time than usual to converge. Inserting an effort parameter in the SIVIA and executing a correction also in the search space could increase TIBA's performance and let it more competitive against PF or any other radar tracking algorithm.

## REFERENCES

- [1] M. I. Skolnik, *Radar Handbook, Second Edition*. New York: McGraw-Hill, 1990.
- [2] A. Arnold-Bos, A. Khenchaf, and A. Martin, "Bistatic radar imaging of the marine environment. part 1: theoretical background and part 2: simulation and results analysis," *To appear in IEEE Transactions on Geoscience and Remote Sensing, Special Issue on Synthetic Aperture Radar*, 2007.
- [3] M. N. Petsios, E. G. Alivizatos, and N. K. Uzunoglu, "Manoeuvring target tracking using multiple bistatic range and range-rate measurements," *Signal Process.*, vol. 87, no. 4, pp. 665–686, 2007.
- [4] R. Moore, *Dimensional Analysis*, E. Cliffs, Ed. New Jersey: Prentice Hall, 1966.
- [5] L. Jaulin, M. Kieffer, O. Didrit, and E. Walter, *Applied Interval Analysis*. Springer, 2001.
- [6] R. B. Kearfott and V. Kreinovich, Eds., *Applications of Interval Computations*. Kluwer Academic Publishers, 1996.
- [7] E. Hansen and G. Walster, *Global Optimization Using Interval Analysis*, 2nd ed. Marcel Dekker, Inc, 2004.
- [8] S. Julier and J. Uhlmann, "A new extension of the Kalman filter to nonlinear systems," in *Int. Symp. Aerospace/Defense Sensing, Simul. and Controls, Orlando, FL*, 1997.
- [9] N. J. Gordon, D. J. Salmond, and A. F. M. Smith, "Novel approach to nonlinear/non-gaussian bayesian state estimation," *IEE Proceedings*, vol. Part F 140, no. 2, pp. 107–113, 1993.
- [10] L. Jaulin, M. Kieffer, E. Walter, and D. Meizel, "Guaranteed robust nonlinear estimation with application to robot localization," *Systems, Man and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, vol. 32, pp. 374–381, 2002.