WCCI 2012 IEEE World Congress on Computational Intelligence
June, 10-15, 2012 - Brisbane, Australia
FUZZ IEEE

Tracking Algorithm Compensating Acceleration for 3D Maneuvering Target with PSO-FCM

Hyun Seung Son
Dept. of Electrical and
Electronic Engineering,
Yonsei University, Korea
E-mail: ssonhyung@yonsei.ac.kr

Jin Bae Park
Dept. of Electrical and
Electronic Engineering,
Yonsei University, Korea
E-mail: jbpark@yonsei.ac.kr

Young Hoon Joo
Dept. of Control and Robot Engineering,
Kunsan National University, Korea
Telephone: (+82) 63-469-4706
E-mail: yhjoo@kunsan.ac.kr

***Abstract*—This paper presents an intelligent tracking method for nonlinear maneuvering target by compartmentalizing external noises of 3D maneuvering target. Proposed method makes the filter recognize the maneuvering target as linear one by separating acceleration properly from the overhaul noises. For achieving that, we use the particle swam optimization-fuzzy c-means (PSO-FCM) clustering as the criteria of methodology. The positional difference between measured point and predicted one is separated into acceleration and noise. Compartmentalized external noises plays the role of acceleration in accordance with assigned position and its quantity. Proposed algorithm makes approximated acceleration to be compensated and approximated noise is filtered by Kalman filter (KF). To depict the real maneuvering target and track the target with unlimited condition, we handle 3D dynamic model. Finally, some examples are provided to show the effectiveness of the proposed algorithm.**

I. Introduction

The target tracking problem has become not only a military interest but also closely linked applications to our lives. Basically, target tracking is model based and the most general model is state-space models [1], [2]. Even though the target model is well designed, the error always exist. To reduce error, a variety of techniques have been studied and developed in the field of the state estimation over decade [5], [23].

One stem for reducing error and precise tracking is the modelling and the other one is the filtering. With Kalman filter (KF) introduced by Kalman [6], [7], tracking technique was developed diversely in 1960s. Directly improved methodology related to the filter is the first approach. The examples are alpha-beta filter [8], information filter [3], square-root filter [9], smoothing and prediction [4], [10], particle filter (PF) [11], [12], and extended Kalman filter (EKF) [13], [14]. The others are the adaptive methods including input estimation (IE) [15], [16], variable state dimension (VSD) [17], generalized pseudo-Bayesian estimator of first order (GPB-1) [18], [19], generalized pseudo-Bayesian estimator of second order (GPB-2) [20], interacting multiple model (IMM) [21], and adaptive interacting multiple model (AIMM) [22], [23].

Filter related methods are much vulnerable to the nonlinear target [6]-[12]. It is difficult to approximate the tsrget because the overall process noise is time-varying. To alleviate these difficulties, artificial intelligent methods like IMM were proposed [21]. Unfortunately, the multiple structure algorithm requires the predefined sub-models with different conditions, and may not guarantee good performance if one of sub-models does not exactly match to the target motion. In AIMM, the acceleration intervals for the different acceleration levels should be determined in advance and the delay involved in estimating target's acceleration should be also treated appropriately.

Motivated by the above observations, we design the adaptive tracking algorithm. Proposed algorithm extracts acceleration in the overhaul noses term effectively. We use the particle swarm optimization (PSO) [24] based fuzzy c-means (FCM) clustering [25], [26] as the criteria which separate overhaul noises between acceleration and mere noise. PSO has the following characteristics. It has simple concept, easy realization, effective calculation, rapidly development in a short time, and stable convergence. FCM makes the clustering by setting fuzzy membership with the distance in Euclidean space and optimizing this one. FCM has better clustering effect than existing hard methods. Proposed extracting acceleration method based on segmentalized PSO-FCM has the following benefits. First, acceleration can be distinguishable from the noise term adaptively not being influenced by the noises of other axes. Second, extracted acceleration is compensated to dynamic model of maneuvering target. This one prevent from falling down the performance of filter due to external factor. Third, we make a realization for maneuvering target by modelling 3D motion. We construct real time algorithm excluding off-line process.

This paper is organized as follows: Section 2 shows the maneuvering target model and briefly reviews the PSO and FCM as the applicable methods. The details of the proposed algorithm is described in Section 3. In Section 4, the effectiveness of the proposed intelligent tracking method compared with the conventional one is shown. Finally, conclusions are drawn in Section 5.

II. Preliminaries

A. Dynamic Model

The dynamic models of the maneuvering target are described for all axes by

$$x(k+1) = Fx(k) + G\big[a(k) + w(k)\big] \tag{1}$$

$$z(k) = Hx(k) + \nu(k) \tag{2}$$

where $x(k) = [x_x(k),\ \dot{x}_x(k),\ x_y(k),\ \dot{x}_y(k),\ x_z(k),\ \dot{x}_z(k)]^T$ is the state vector with position and velocity for a maneuvering target. The term $a(k)$ and $w(k)$ are the unknown deterministic acceleration and the process noise, $z(k)$ and $\nu(k)$ are the measurement vector and the measurement noise, $w(k)$ and $\nu(k)$ are considered as zero-mean white Gaussian noise sequences with variances q and r, respectively. The system matrix F, the gain matrix G, and the measurement matrix H are specified as follows when k is the sampling period:

$$F = \begin{bmatrix} 1 & k & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & k & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & k \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \quad G = \begin{bmatrix} k^2/2 & 0 & 0 \\ k & 0 & 0 \\ 0 & k^2/2 & 0 \\ 0 & k & 0 \\ 0 & 0 & k^2/2 \\ 0 & 0 & k \end{bmatrix},$$

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}.$$

In the case of nonlinear maneuver, the unknown acceleration of the target is regarded as an additive process noise. Hence, (1) can be rewritten as:

$$x(k+1) = Fx(k) + G\bar{\omega}(k) \tag{3}$$

where $\bar{\omega}(k) = a(k)+w(k)$ is assumed to be the overall process noise.

B. Particle Swarm Optimization - Fuzzy C-means Clustering (PSO-FCM)

1) Particle Swarm Optimization (PSO): PSO is a stochastic optimization technique inspired by the social behavior of a bird flocking or fish schooling which was proposed by Kennedy and Eberhart in 1995 [24]. As simple in concept and easy to implement, it develops rapidly in a short time, and has been applied in various fields. In this paper, we use this algorithm combined with FCM clustering.

2) Fuzzy c-means clustering (FCM): FCM proposed by Bezdek is the data classification algorithm categorizing each data to a certain cluster by the strength of membership function [25] and [26]. The membership function takes proper one between 0 and 1 in the FCM. Using this characteristics, FCM will be used to divide acceleration from mere noise in this paper. The process consisted of following 4 steps.

STEP 1: **(Initializing membership function and parameter decision)** The initial membership function μ_{ij} and the parameters are set to meet the following conditions:

$$\sum_{i=1}^{c} \mu_{ij} = 1, \quad j = 1, 2, \ldots, n, \tag{4}$$

$$0 \ \langle \ \sum_{k=1}^{n} \mu_{ik} \ \langle \ n \tag{5}$$

where c is the number of cluster, n is the number of input data.

Cost function J is defined as:

$$J(\mu_{ik}, \gamma_i) = \sum_{i=1}^{c} \sum_{k=1}^{n} \mu_{ik}^{m} (d_{ik})^2 \tag{6}$$

where

$$d_{ik} = d(x_k - \gamma_i) = \left[\sum_{j=1}^{l} (x_{kj} - \gamma_{ij})^2 \right]^{\frac{1}{2}},$$
$$\gamma_i = \left\{ \ \gamma_{i1}, \quad \gamma_{i2}, \quad \ldots, \quad \gamma_{il} \ \right\},$$

m is exponential weight, l is the dimension of data, d_{ik} is the Euclidean distance of x_k and γ_i and γ_i is a center of a i_{th} cluster.

STEP 2: **(Calculation of cluster's center)** To minimize the cost function, the cost function is divided into the centers of the clusters γ_{ij} and membership function μ_{ij} as follows:

$$\gamma_{ij} = \frac{\sum_{k=1}^{n} (\mu_{ik})^m x_{kj}}{\sum_{k=1}^{n} (\mu_{ik})^m}, \tag{7}$$

$$\begin{aligned} \mu_{ik} &= \frac{\left(1/\|x_k - \gamma_i\|^2\right)^{\frac{1}{m-1}}}{\sum\limits_{j=1}^{c} \left(1/\|x_k - \gamma_j\|^2\right)^{\frac{1}{m-1}}} \\ &= \frac{1}{\sum\limits_{j=1}^{c} \left[\frac{\|x_k - \gamma_i\|}{\|x_k - \gamma_j\|}\right]^{\frac{2}{m-1}}} \\ &= \frac{1}{\sum\limits_{j=1}^{c} \left[\frac{d_{ik}}{d_{jk}}\right]^{\frac{2}{m-1}}}. \end{aligned} \tag{8}$$

STEP 3: **(Updating membership function)** The updated membership function μ_{ik}^{r+1} is obtained by the centers of the clusters obtained at STEP 2 as follows:

$$\mu_{ik}^{r+1} = \frac{1}{\sum_{j=1}^{c} \left[\frac{d_{ik}^r}{d_{jk}^r}\right]^{\frac{2}{m-1}}}. \tag{9}$$

STEP 4: **(Comparison with critical value and repetition)** Updating centers and membership function is continued until the gap of membership function (10) is less than the critical value Λ

$$\Lambda = \| \ U^{r+1} - U^r \ \| = max_{ik} \mid \mu_{ik}^{r+1} - \mu_{ik}^{r} \mid \tag{10}$$

where U^r is the set of r repeated membership function μ_{ij}^r. If we change the initial value of membership function μ_{ik}, the final center γ_i will be changed easily. So it is important to get the best initial value.

III. Main Description

A. Compartmentalization by Segmentalized PSO-FCM at Each Axis

The most important concern of the tracking system is to diminish the error between the measurement point $z(k)$ and the predicted one $\hat{u}(k)$. Existing manners yield velocity by differentiating the positional error and obtain the approximated acceleration by differentiating the velocity. In this case, the process of differentiating is executed with the overhaul noises, so the estimated value is always accompanied noise. Many methods treating the overhaul noises could reduce the error to some degree but those could not be a fundamental solution. The reason is the acceleration in the noise term plays a role of the noise with bigger magnitude than real noise.

The basic idea of the proposed method arises from the fact that it is difficult to separate acceleration directly from other noises because it is given with other components. So, we should approximate the acceleration of a maneuvering target and utilize this one to estimate the target. First, we set the positional error between measurement and predicted point to be used as a data to get the approximated acceleration and we need a criteria which split this error into acceleration and noise. We call this criteria maximum noise level and PSO-FCM is used as a dividing technique. Going through with PSO-FCM, the exceedances are regarded as approximated acceleration and the rest is regarded mere noise. Only mere noise is on going filtering process and the filtered output is added corresponding acceleration.

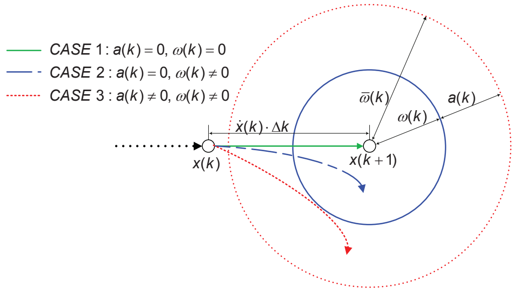
1) Compartmentalization of the positional errors: To approximate acceleration, we make the maximum noise level as the permissive noise value corresponding with the capability of the equipment. This includes only the noisy factor like process noise and measured one excluding acceleration. Noise only makes an error to the dynamic model. Let us suppose a linear moving target. When we take a look at the relation between neighboring two points, $x(k)$ and $x(k+1)$, later one depends on the prior one's velocity and elapsed time $\dot{x}(k) \cdot \Delta k$ and the noise ω is added to this. If we restrict ω to a certain value, we can conjecture about the distance gap between the two points as a certain range. Next, let us think of the nonlinear maneuvering case. In this case, the distance gap will be much variable by the acceleration input incurring the maneuvering. What the movement of the maneuvering target is largely varied means that the distance gap of $x(k)$ and $x(k+1)$ gets out of the range ω. In Fig. 1, we can see the positional relation of linear moving, noise added one, and acceleration added maneuvering.

Considering linear movement and nonlinear maneuvering together, we get the predicted point $\hat{u}(k+1|k)$ as follows:

$$\hat{u}(k+1|k) = \hat{x}(k|k) + \dot{x}(k|k) \cdot \Delta k. \tag{11}$$

Then, we get the positional error $\hat{e}$ between the predicted point $\hat{u}(k+1|k)$ and measurement one $z(k+1|k+1)$ as follows:

$$\hat{e}(k+1|k) = z(k+1|k+1) - \hat{u}(k+1|k). \tag{12}$$



$CASE\ 1 : a(k) = 0,\ \omega(k) = 0$
$CASE\ 2 : a(k) = 0,\ \omega(k) \neq 0$
$CASE\ 3 : a(k) \neq 0,\ \omega(k) \neq 0$
$\overline{\omega}(k)$
$a(k)$
$\omega(k)$
$\dot{x}(k) \cdot \Delta k$
$x(k)$
$x(k+1)$

Fig. 1. Relative positional gap of external noise input

The set of the dispersion for the positional errors $\hat{E}$ from $t=0$ to $t=k$ is formed around zero. It is shown in Fig. 2-(a) and the set $\hat{E}_{lk}$ is represented as follows:

$$\hat{E}_{lk} = \left\{ \begin{array}{c} \hat{e}_1 \\ \vdots \\ \hat{e}_l \end{array} \right\} = \left\{ \begin{array}{ccc} \hat{e}_{11} & \ldots & \hat{e}_{1k} \\ \vdots & \ddots & \vdots \\ \hat{e}_{l1} & \ldots & \hat{e}_{lk} \end{array} \right\} \tag{13}$$

where l means the dimension of the data, $\hat{e}_d$ does the data set from 0 to k at l_{th} axis, and k does the sampling time.

We proceed segmentalized PSO-FCM with the set of the dispersion $\hat{E}$ conjugating the original FCM process expressed in (4)-(10) with PSO algorithm. This procedure is proceeded at the x,y,z-$axis$, respectively. To evaluate each element in the mass, the fitness function is defined as follows:

$$f\left(\hat{e}_i\right) = \frac{K}{J_m\left(\Theta, \Gamma\right)} \tag{14}$$

where K is a constant, $J_m(\Theta, \Gamma)$ is the objective function, Θ is the set of membership function μ_{ij}, and Γ is the set of cluster γ_i. If $J_m(\Theta, \Gamma)$ become smaller, the clustering effect has better one and individual fitness $f(\hat{e}_i)$ also becomes better.

The process of PSO-FCM can be described as follows:

(1) Give the number of clusters c, fuzzy index m, population size n, learning factors c_1 and c_1, inertia weight w.

(2) Initialize cluster centers and its position, create the first generation of particles.

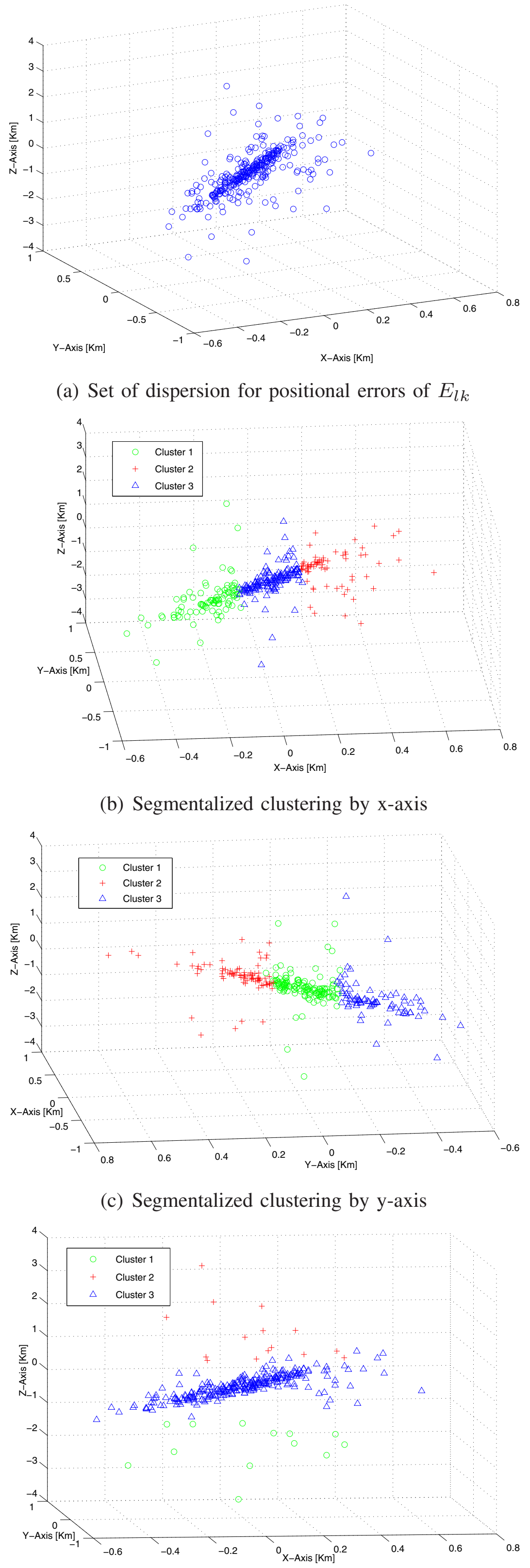
(3) Calculate the cluster center γ_j as follows:

$$\forall j, \gamma_j = \frac{\sum\limits_{i=1}^{n} \mu_{ij}^m \hat{e}_i}{\sum\limits_{i=1}^{n} \mu_{ij}^m}. \tag{15}$$

(4) Calculate μ_{ij} of each cluster center as follows:

$$\mu_{ij} = \frac{1}{\sum\limits_{r=1}^{c} \left[\frac{d_{ij}}{d_{ir}}\right]^{\frac{2}{m-1}}}. \tag{16}$$

(5) Calculate the fitness value of each particle according to (14). If the fitness of arbitrary particle is better than that of the current particle's best location, then update the best location of the individual particles. If the fitness of all the particles' best location is better than that of the global best current location, then update the global location.



(a) Set of dispersion for positional errors of E_{lk}

(b) Segmentalized clustering by x-axis

(c) Segmentalized clustering by y-axis

(d) Segmentalized clustering by z-axis

Fig. 2. Set of dispersion for 3D data and segmentalized clustering by each axis

(6) Update the velocity and location of each particle and produce the next generation of particle swarm.

(7) If the current iteration number reaches the pre-set maximum, stop iteration and find the best solution in the last generation. Otherwise, return step (3).

By undergoing through a series of process (11)-(16), each input data has following 3 degrees of membership function for the clusters classified by each axis.

$$\begin{aligned} \mu_{x|ik} &= \{\mu_{x1},\ \mu_{x2},\ \mu_{x3}\}, \\ \mu_{y|ik} &= \{\mu_{y1},\ \mu_{y2},\ \mu_{y3}\}, \\ \mu_{z|ik} &= \{\mu_{z1},\ \mu_{z2},\ \mu_{z3}\} \quad for\ \ e(\hat{k}) \end{aligned} \tag{17}$$

where $\mu_{x1} + \mu_{x2} + \mu_{x3} = 1$, $\mu_{y1} + \mu_{y2} + \mu_{y3} = 1$, and $\mu_{z1} + \mu_{z2} + \mu_{z3} = 1$.

Intersected all clusters with other clusters formed by other axes, we can get the stereoscopic structured degree of membership function composed with 27 clusters.

2) Making Maximum Noise Level: In degree of membership function μ_{xyz}, 1 means the cluster which has deceleration input to relevant axis, 2 does mere noise to that, and 3 does acceleration input to that, respectively. 27 degree of membership functions μ_{xyz} has the following characteristics. Each element of the 1^{st} row has the maximum weight of deceleration input for x-$axis$, the 2^{nd} one does mere noise, and the 3^{rd} one does acceleration input, respectively. Each element of the 1^{st} column has the maximum weight of deceleration input for y-$axis$, the 2^{nd} one does mere noise, and the 3^{rd} one does acceleration input. All the elements of the 1^{st} layer have the maximum weight of deceleration input for z-$axis$, the 2^{nd} one do mere noise, and the 3^{rd} one do acceleration input. Adding more data as time passes, this value is getting accurate about maneuvering pattern.

B. Adaptive Tracking Model

With all the foregoing methods, we set up the one cycle of the PSO-FCM based adaptive tracking (PFAT) model following five steps.

1. Initialization: Substituting the predicted point $\hat{u}_{k+1|k}$ into the measurement z_{k+1} yields estimated positional error $\hat{e}_{k+1|k}$ from the foregoing (1),(2),(3),(11),(12). The set of dispersion of estimated positional error from 0 to k $\hat{E}$ is used as input data for compartmentalization.

2. Compartmentalization of external noise: Going through the segmentalized PSO-FCM with the result of STEP 1 $\hat{E}$, the algorithm yields the 27 clusters representing acceleration, deceleration, and mere noise as shown in Fig. 2-(b),(c),(d). The degree of membership function $\mu_{xyz}(k+1|k)$ means the weight of the estimated positional error $\hat{e}_{k+1|k}$ to each cluster.

3. Set up the acceleration and noise: With the result of STEP 2, the acceleration and noise are obtained by the multiplication $\hat{e}(k)$ and μ_{xyz}. Row and column of a maximum degree of membership function mean the axis and characteristics of that. Compartmentalizing external noise to yield expected acceleration and noise is divided into following 3 cases:

CASE 1 ($Maximum\ weight\ for\ mere\ noise$) : With no acceleration input and no deceleration one, the expected noise is made by the multiplication the positional error data $\hat{e}_x(k+1|k)$ with degree of membership function $\mu_{x2}(k+1|k)$ at time k. In this procedure, the approximated noise $\hat{\eta}_x(k+1|k)$ is set as

$$\hat{\eta}_x(k+1|k) = \hat{e}_x(k+1|k) \cdot \mu_{x2}(k+1|k). \tag{18}$$

To reflect no external noise, the weight for the acceleration and deceleration would be shown as follows:

$$\hat{a}_x(k+1|k) = 0, \tag{19}$$

$$\hat{d}_x(k+1|k) = 0. \tag{20}$$

CASE 2 ($Maximum\ weight\ for\ acceleration\ input$) : When acceleration coexists with noise, estimated acceleration input $\hat{a}_x(k+1|k)$ on relevant axis is made by the multiplication positional error $\hat{e}_x(k+1|k)$ with degree of membership function $\mu_{x3}(k+1|k)$ at time k. In this case, $\hat{a}_x(k+1|k)$ is calculated as follows:

$$\hat{a}_x(k+1|k) = \hat{e}_x(k+1|k) \cdot \mu_{x3}(k+1|k). \tag{21}$$

Expected noise is obtained by the multiplication the positional error $\hat{e}_x(k+1|k)$ with degree of membership function $\mu_{x2}(k+1|k)$ at time k as follows:

$$\hat{\eta}_x(k+1|k) = \hat{e}_x(k+1|k) \cdot \mu_{x2}(k+1|k). \tag{22}$$

To reflect acceleration input, the weight for the deceleration would be zero as follows:

$$\hat{d}_x(k+1|k) = 0. \tag{23}$$

CASE 3 ($Maximum\ weight\ for\ deceleration\ input$) : This is the case when deceleration coexists with noise contrary to CASE 2. Estimated deceleration input on relevant axis is made by the multiplication positional error $\hat{e}_x(k+1|k)$ with degree of membership function $\mu_{x1}(k+1|k)$ at time k as follows:

$$\hat{d}_x(k+1|k) = \hat{e}_x(k+1|k) \cdot \mu_{x1}(k+1|k). \tag{24}$$

Expected noise is obtained by the multiplication the positional error $\hat{e}_x(k+1|k)$ with degree of membership function $\mu_{x2}(k+1|k)$ at time k as follows:

$$\hat{\eta}_x(k+1|k) = \hat{e}_x(k+1|k) \cdot \mu_{x2}(k+1|k). \tag{25}$$

To reflect acceleration input, the weight for the acceleration would be zero as follows:

$$\hat{a}_x(k+1|k) = 0. \tag{26}$$

The process for y-$axis$ and z-$axis$ is equal to the above one.

4. Filtering and compensating acceleration: Substituting (21)-(26) into (1) yields expected target $\hat{x}(k+1)$ based on expected acceleration $\hat{a}_x$,$\hat{a}_y$,$\hat{a}_z$, expected deceleration $\hat{d}_x$,$\hat{d}_y$,$\hat{d}_z$, and expected mere noise $\hat{\eta}_x$,$\hat{\eta}_y$,$\hat{\eta}_z$ as follows:

$$\begin{aligned}\hat{x}(k+1|k) \approx & F\hat{x}(k) + G\big(\hat{a}(k+1|k) + \hat{d}(k+1|k) \\ & + \hat{\eta}(k+1|k)\big)\end{aligned} \tag{27}$$

According to the 3 CASEs, term $\hat{a}(k+1|k) + \hat{d}(k+1|k) + \hat{\eta}(k+1|k)$ can be $\hat{a}(k+1|k)+\hat{\eta}(k+1|k)+0$, $\hat{d}(k+1|k)+\hat{\eta}(k+1|k)+0$, or $\hat{\eta}(k+1|k)+0$. Approximated mere noise $\hat{\eta}(k+1|k)$ will be strained by KF. The input for filtering process $i_f(k+1|k)$ would be defined as follows:

$$i_f(k+1|k) = z(k+1|k+1) - \hat{A}(k+1|k) \tag{28}$$

where

$$\hat{A}(k+1|k) = \begin{bmatrix} \hat{a}_x(k+1|k) + \hat{d}_x(k+1|k) \\ \hat{a}_y(k+1|k) + \hat{d}_y(k+1|k) \\ \hat{a}_z(k+1|k) + \hat{d}_z(k+1|k) \end{bmatrix}.$$

If $\hat{o}_f(k+1|k)$ is defined as output of the filter, approximated acceleration $\hat{a}(k+1|k)$ and approximated deceleration $\hat{d}(k+1|k)$ yield final expected value $\hat{x}(k+1|k+1)$ by compensating the position and velocity of $\hat{o}_f(k+1|k)$ as follows:

$$\hat{x}(k+1|k+1) = \hat{o}_f(k+1 \mid k) + G \cdot \hat{A}(k+1|k) \tag{29}$$

where

$$\hat{o}_f(k+1|k) = \begin{bmatrix} \hat{x}_x & \dot{\hat{x}}_x & \hat{x}_y & \dot{\hat{x}}_y & \hat{x}_z & \dot{\hat{x}}_z \end{bmatrix}^T.$$

5. Yielding estimates and repetition: Calculated $\hat{x}(k+1|k+1)$ is used as the initial input $\hat{x}(k+2|k+1)$ at the next sampling time $k+2$ and repeat the all steps.

Remark 1: The proposed algorithm has the following advantages. Learning process is proceeded with noise analysis at every sampling time. So this algorithm can be applied to the on-line system.

IV. Simulations

To show the effectiveness of the proposed intelligent tracking method, we introduce a simulated tracking scenario for an incoming cruise missile. The simulation results of the proposed method are compared with those of the KF. The target is set as an incoming cruise missile on the 3D plane. The initial position of the target is at (150, 100)km away on the distant horizon from the observer, and it moves with a constant velocity of $0.8856km/sec$ along a -225 degree line to x-$axis$. For the each axis, the standard deviation of the zero mean white Gaussian measurement noise is 0.001km and that of a process noise is 0.01. The sampling time k is $1sec$ and the number of iteration is 300.

The target has the lateral acceleration as shown in Fig. 3 and the corresponding target motion is determined from (1)-(2) and illustrated in Figs. 4,5. The simulation results for the proposed algorithm of which performances are compared with those of the KF is shown in Figs. 6, 7, and Table I, respectively. The Figures and the Table show that the proposed algorithm produces smaller errors than those of comparison one. Table I shows that proposed method has reduced by 83.78% compared with comparison one.

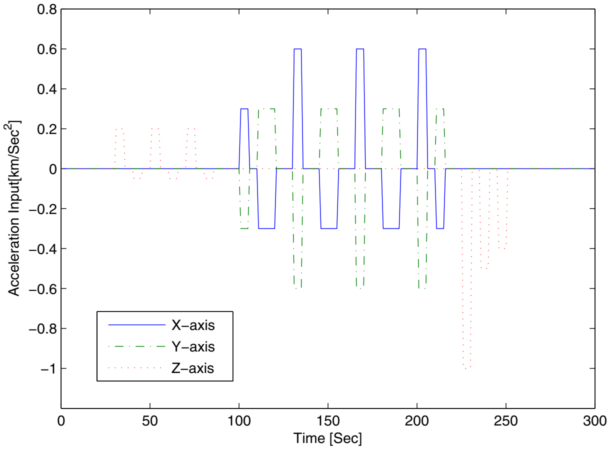


Fig. 3. Acceleration input

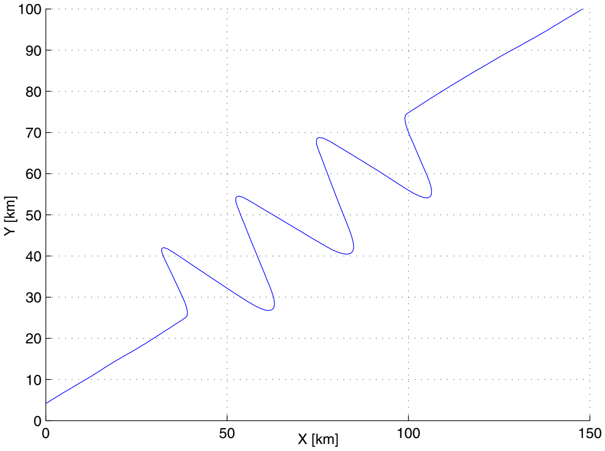


Fig. 4. Ideal target motion on x-y plane for incoming cruise missile

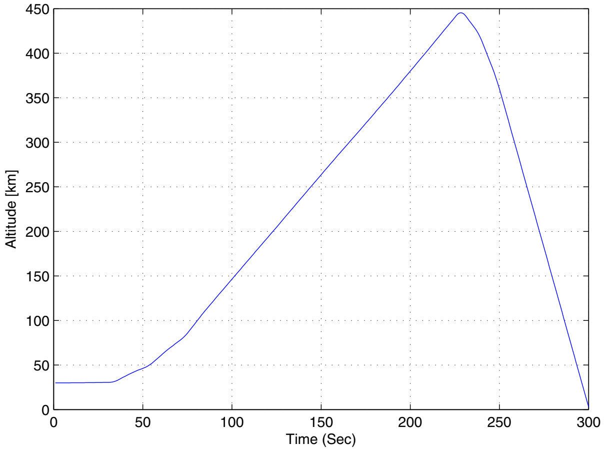


Fig. 5. Ideal target motion on z plane for incoming cruise missile

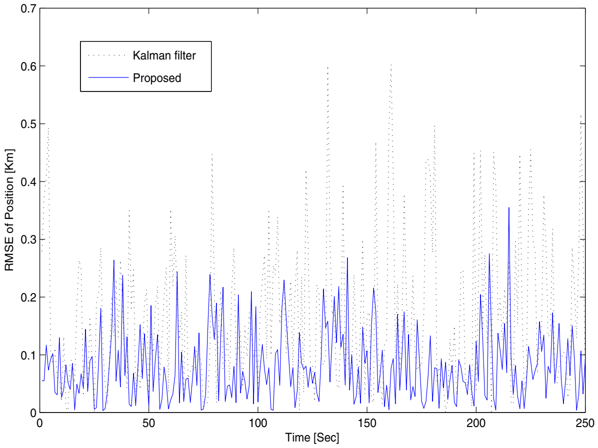


Fig. 6. RMSE comparison of Position with KF

V. Conclusions

In this paper, the intelligent model for racking a maneuvering target has been proposed. In the proposed model, PSO fused FCM has adaptively compartmentalized the external noises. With that, the proposed algorithm have extracted acceleration and compensated tracking errors. The proposed algorithm shows the better performances compared with the comparisons.

Acknowledgment

This work has been supported by Yonsei University Institute of TMS Information Technology, a Brain 21 program, Korea and the Human Resources Development of the Korea Institute of Energy Technology Evaluation and Planning (KETEP) grant funded by the Korea government Ministry of Knowledge Economy. (No. 20104010100590)

References

[1] Y. Bar-Shalom, X. R.Li, and T. Kirubarajan, Estimation with Applications to Tracking and Navigation: Theory, Algorithms, and Software, Wiley, New York, 2001.

[2] Hinrichsen and Diederich, Dynamical systems theory modelling: State space analysis, stability and robustness, Springer, New York, 2011.

[3] P. S. Maybeck, Stochastic models, estimation and control, Vol. I, Academic Press, New York, 1979.

[4] Maybeck, Peter S., Stochastic models, estimation, and control, Vols. II and III, Academic Press, New York, 1982.

[5] X. R. Li and V. P. Jilkov, “Survey of maneuvering target tracking. Part I: Dynamic models,” *IEEE Trans. Aerospace and Electronic Syst.*, vol. 39, no. 4, pp. 1333-1364, 2003.

[6] R. E. Kalman, “A new approach to linear filtering and prediction problems,” *Journal of Basic Engineering*, vol. 82, no. 1, pp. 35-46, 1960.

[7] R. A. Singer, “Estimating optimal tracking filter performance for manned manoeuvering targets,” *IEEE Trans. Aerospace and Electronic Syst.*, vol. 6, no. 4, pp. 473-483, 1970.

[8] J. E. Gray and W. Murray, “A derivation of an analytic expression for the tracking index for the alpha-beta-gamma filter,” *IEEE Trans. on Aerospace and Electronic Syst.*, vol. 29, no. 3, pp. 1064-1065, 1993.

[9] H. R. Itzkowitz, and R. S. Baheti, “Demonstration of square root Kalman filter on warp parallel computer,” American Cont. Conf. 1989, pp. 1754-1762, 2010.

[10] A. Sage and J. Melsa, Estimation theory with applications to communications and control, McGraw-hill, New York, 1971.

[11] Y. Liu and R. Chen, “Sequential Monte Carlo methods for dynamic systems,” *Journal of the American Statistical Association*, vol. 93, pp. 1032-1044, 1998.

[12] M. Pitt and N. Shephard, “Filtering via simulation: auxiliary particle

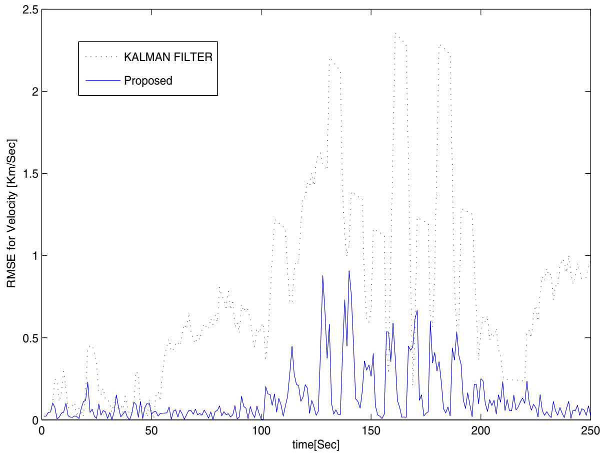


Fig. 7. RMSE comparison of velocity with KF

TABLE I
COMPARISONS OF RMSE FOR POSITION AND VELOCITY

Configuration	Position	Velocity	Average
KF	0.1650	1.2215	0.6932
Proposed	0.0807	0.1443	0.1125

filter," *Journal of the American Statistical Association*, vol. 94, pp. 590-599, 1999.

[13] H. Weiss and J.B. Moore, "Improved extended Kalman filter design for passive tracking," *IEEE Trans. Auto. Cont.*, vol. 9, no. 6, pp. 795-809, 2001.

[14] Taek L. Song and J. L. Speyer, "A stochastic analysis of a modified gain extended Kalman filter with applications to estimation with bearings only measurements," *IEEE Trans. Auto. Cont.*, vol. 30, no. 10, pp. 940-949, 1985.

[15] Y. T. Chan, A. G. C. Hu, and J. B. Plant, "A Kalman filter based tracking scheme with input estimation," *IEEE Trans. Aerospace and Elect. Syst.*, vol. 15, no. 2, pp. 237-244, 1979.

[16] P. L. Bogler, "Tracking a maneuvering target, using input estimation," *IEEE Trans. Aerospace and Elect. Syst.*, vol. 23, no. 3, pp. 298-310, 1987.

[17] Y. Bar-Shalom and K. Birmiwal, "Variable dimension filter for maneuvering target tracking," *IEEE Trans. Aerospace and Elect. Syst.*, vol. 18, no. 5 pp. 621-629, 1982.

[18] A. G. Jaffer and S. C. Gupta, "Optimal sequence estimation of discrete processes with Markov interrupted observations," *IEEE Trans. Auto. Cont.*, vol. 18, no. 5 pp. 621-629, 1982.

[19] A. G. Jaffer and S. C. Gupta, "Recursive bayesian estimation with uncertain observation," *IEEE Trans. Information theory*, vol. 17, no. 5, pp. 614-616, 1971.

[20] C. B. Chang, and M. Athans, "State estimation for discrete sytems with switching parameters," *IEEE Trans. Aerospace and Elect. Syst.*, vol. 14, no. 3, pp. 418-425, 1978.

[21] H. A. P. Blom, "The interacting multiple model algorithm for systems with markovian switching coefficients," *IEEE Trans. Auto. Cont.*, vol. 33, no. 8, pp. 780-783, 1988.

[22] A. Munir and D. P. Atherton, "Adaptive interacting multiple model algorithm for tracking a maneuvering target," *IEE Proc. Radar, Sonar and Navigation*, vol. 142, pp. 11-17, 1995.

[23] S. McGinnity and G. W. Irwin, "Fuzzy logic approach to maneuvering target tracking," *IEE Proc. Radar, Sonar and Navigation*, vol. 145, no. 6, pp. 337-341, 1998.

[24] J. Kennedy and R. Eberhart, "Particle swarm optimization," IEEE International Conf. Neural Networks, vol. 4, pp. 1942-1948, 1995.

[25] J. C. Bezdek, R. Ehrlich, and W. Full, "FCM: The fuzzy c-means clustering algorithm," *Computers and Geosciences*, vol. 10, no. 2-3, pp. 191-203, 1984.

[26] R. L. Cannon, J. V. Dave, and J. C. Bezdek, "Efficient implementation of the fuzzy c-means clustering algorithms," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 8, no. 2, pp. 248-255, 1986.