A Novel Data Fusion Approach in an Integrated GPS/INS System Using Adaptive Fuzzy Particle Filter

Ali Asadian and Behzad Moshiri

Control and Intelligent Processing Center of Excellence, Department of Electrical and Computer Engineering, University of Tehran, Tehran, Iran

a.asadian@ece.ut.ac.ir , moshiri@ut.ac.ir

Ali Khaki Sedigh

Control Branch, Department of Electrical Engineering, University of K.N.T,Tehran, Iran

sedigh@eetd.kntu.ac.ir

Abstract* – *In this paper we propose a new data fusion method based on particle filtering and fuzzy logic in order to adaptively integrate global positioning system and strapdown inertial navigation system (GPS/SDINS). This approach will reduce the dependence of the stable solution on stochastic properties of the system which is a function of vehicle dynamics and environmental conditions So the proposed scheme will enhance the estimation performance in comparison with generic particle filter specially in the case of facing modeling uncertainty. It will also give us more reliable solution when encountering satellite signal blockage as a probable problem in land navigation. The results have clearly demonstrated that the hybrid fuzzy particle filter would improve the guidance from the point of accuracy and robustness to the mentioned problems.

Index Terms* – *Adaptive Particle Filter, Data Fusion, Fuzzy Logic, GPS/INS.

I. Introduction

It is well established that global positioning system (GPS) can provide position and velocity information of moving platforms with consistent accuracy throughout the surveying mission. The limitations of GPS are related to the loss of accuracy in the narrow-street environment, intentional disruption of the service, poor geometrical-dilution-of-precision (GDOP) coefficient and the multipath reflections. GPS-based navigation system requires at least four satellites, so a major drawback of GPS dependence navigation systems is that their accuracy degrades significantly with satellites' outages. Signal outage is more significant for land vehicle positioning in urban centers, which takes place when encountering highway overpasses or tunnels due to the obstructed signals. So it is suitable to integrate this type of navigation system with a different type of navigation system in order to reach a greater autonomy. From this point of view, the inertial navigation system (INS) is ideal. Rather than using signals receiving from satellites, in the case of GPS, the INS is based on measurements of linear accelerations and angular velocities. INS measures the linear acceleration and angular rates of moving vehicles through its accelerometers and gyroscopes sensors. The main interest is the position determination, which is possible after a double integration of the accelerations to obtain linear displacements and a single integration of the angular velocities to obtain the angles of rotation. The INS error bound grows with time, due to the unbounded positioning errors caused by the uncompensated gyro and accelerometer errors affecting the INS measurements. INS provides high-accuracy three-dimensional positioning when the GPS positioning is poor or unavailable over short periods of time. In addition, it provides much higher update positioning rates compared with the output rate conventionally available from GPS [1].

Anyway in order to utilize the benefits of these two navigation sensors and gain the advantages of the data fusion, we fuse the data gathered by each and use integrated system. There are several integration schemes using a blending filter such as particle filter to combine the GPS and INS data [2],[8]-[10]. In order to reduce the impact of accuracy decreasing when GPS becomes unavailable and reaching a high resolution in compare with extended Kalman filter as a classic approach, a particle filer has been used on a simplified 2-dimentional navigation error model, built from stand-alone INS on one hand, and from the GPS on the other hand [2]. This fact has been illustrated in Fig. 1. For this purpose, the GPS pseudoranges are excellent external measurements for updating the INS, thus improving its long-term accuracy.

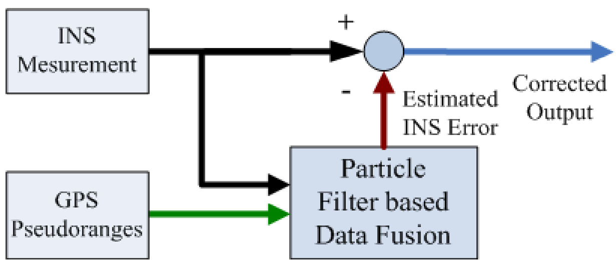

Fig. 1. Tight GPS/INS integration using particle filter

This solution in also applied to a full order navigation model in [8]-[9]. The above network can be used during the availability time of reference system. So the measurements from GPS and INS are used to calculate optimal weights of particles filter consisting of behavior of the INS in some special scenarios of vehicle motion. Particle filter as a widespreading approach provides poor prediction of position errors, when encountering lack of particles. In all of the previous works, it's been assumed that the number of particles is constant so no adaptation has been applied. This assumption could be problematic when facing complicated non-linear dynamics. Estimation accuracy is directly depending on particle numbers. Here in this paper will combine fuzzy adaptation rules with particle filter in order to gain more resolution. So after a brief description of particle filter basics, we will focus on the new contribution.

II. A Brief Introduction to the Particle Filter

Navigation problems are often treated as Bayesian interference. Real INS is a non-linear/non-Gaussian dynamical system, therefore the underlying Bayesian equations are non-tractable. To solve in an online application without using linearization or Gaussian assumptions, sequential Monte Carlo method or particle filter could be used. The basics have been discussed in [2]-[8]. In the Bayesian approach to dynamic state estimation, one attempts to construct the posterior probability density function (pdf) of the state based on all available information including the set of received measurements. Since this pdf embodies all available statistical information, it may be said to be the complete solution to the estimation problem. In principle an optimal estimate of the state with respect to any criterion, may be obtained from this pdf. Such a recursive filter consists of essentially two stages: prediction and update.

The prediction stage uses the system model to predict the state pdf forward from one measurement time to the next. Since the state is usually subject to unknown disturbances modeled as random noise, prediction generally translates, deforms, and spreads the state pdf. The update operation uses the latest measurement to modify the prediction pdf. This is achieved using Bayes theorem which is the mechanism for updating knowledge about in the light of extra information from new data.

Consider the following non-linear system :

$$x_{k+1} = f_k(x_k, u_k, w_k) \tag{1}$$

$$y_k = h_k(x_k, v_k) \tag{2}$$

where $x \in R^n$ denotes the state of the system and $y \in R^m$ stands for the observation in step k. The process noise w_k and measurement noise v_k are assumed independent with the densities P_{v_k} and P_{w_k} respectively. Let $Y_k = \{y_1, y_2, ..., y_k\}$ be the set of observations until present step. The Monte Carlo filter approximates the probability density $P(x_k \mid y_{1:k})$ by a large set of N particles $\{x_k^{(i)}\}_{i=1}^N$ where each particle has an assigned relative weight $\{\omega_k^{(i)}\}_{i=1}^N$ such that all weights sum to unity. The location and weight of each particle reflect the value of the density in the region of the state space. The particle filter updates the particle location and the corresponding weights recursively with each new observation. The non-linear prediction density $P(x_{k+1} \mid y_{1:k})$ and filtering density $P(x_{k+1} \mid y_{1:k+1})$ for the Bayesian interference are given below :

$$P(x_{k+1} \mid y_{1:k}) = \int P(x_{k+1} \mid x_k) P(x_k \mid y_{1:k}) dx_k \tag{3}$$

$$P(x_{k+1} \mid y_{1:k+1}) = \frac{P(y_{k+1} \mid x_{k+1}) P(x_{k+1} \mid y_{1:k})}{P(y_{k+1} \mid y_{1:k})} \tag{4}$$

It's often assumed that both the the process noise and measurement noise is additive in the following form :

$$x_{k+1} = \tilde{f}_k(x_k, u_k) + B_{w,k} w_k \tag{5}$$

$$y_k = \tilde{h}_k(x_k) + \tilde{v}_k \tag{6}$$

So $P(y_{k+1} \mid x_{k+1})$ as the likelihood function and $P(x_{k+1} \mid x_k)$ is calculated based on (5) and (6) using the known measurement noise densities P_{w_k} and $P_{\tilde{v}_k}$ as below :

$$P(y_{k+1} \mid x_{k+1}) = P_{\tilde{v}_{k+1},k}(y_{k+1} - \tilde{h}_k(x_{k+1})) \tag{7}$$

$$P(x_{k+1} \mid x_k) = P_{w_k,k}(\hat{B}_{w,k}(x_{k+1} - \tilde{f}_k(x_k, u_k)) \tag{8}$$

where $\hat{B}$ denotes the Moore-Penrose pseudo-inverse for the corresponding matrix. Finally the main idea is to approximate $P(x_k \mid y_{1:k})$ with a sum of delta-Dirac functions located in the $x_k^{(i)}$ samples . Using the importance weights the posterior can be written as :

$$P(x_{0:k} \mid y_{1:k}) \approx \sum_{i=1}^{N} \omega_k^i \delta(x_{0:k} - x_{0:k}^{(i)}) \tag{9}$$

This was the original estimation idea. However, this approach leads to divergence and suffers from degeneracy problem when some of the particles have zero weights relative to the others. By introducing a sub algorithm called resampling as proposed in [4], this numerical instability could be removed. Finally the *sampling importance resampling (SIR)* pseudo-code is summarized as below :

1. Set *k=0* and generate N samples $\{x_0^{(i)}\}_{i=1}^N$ from the initial distribution $P(x_0)$.
2. Update the weights for *i=1:N* which is including :
   - Computation : $\omega_k^{(i)} = P(y_k \mid x_k^{(i)})$
   - Normalization : $\tilde{\omega}_k^i = \omega_k^{(i)} / \sum_{j=1}^{N} \omega_k^{(j)}$
3. Generate a new set $\{x_0^{(i*)}\}_{i=1}^N$ by resampling with replacement N times from $\{x_0^{(i)}\}_{i=1}^N$ with the following probability :
   - $\{\tilde{\omega}_k^{(j)}\}_{i=1}^N = \Pr(\{x_k^{(i*)}\}_{i=1}^N = \{x_k^{(j)}\}_{i=1}^N)$
4. Compute the number of effective weights :
   - $N_{eff} = 1 / \sum_{i=1}^{N} (\omega_k^{(i)})^2$
   - if : $N_{eff} \le N_{th}$ ($N_{th} = \frac{2N}{3}$) reset the weights to : $\{\omega_k^{(i)}\}_{i=1}^N = 1/N$.
5. Predict new particles using (1) and different noise realizations for the particles :
   - $x_{k+1}^i = f_k(x_k^{i*}, u_k, w_k)$
6. Increase k and iterate to step 2.

For more details, one can refer to the mentioned references. It has been shown that under some conditions, the estimation error is bounded by g_t / N . The function g_t grows with the time but doesn't depend on the dimension of the state vector. On the other hand, one problem in using this method is the computational cost. For a high dimensional system, getting reasonable accuracy means using a large N, which results in a heavy computational cost. Most of the times we choose this value by trial and error based on our priori knowledge about dynamical system. So it's mandatory to minimize the number of particles with respect to our desired resolution. No optimal criterion has been found in the literature, in order to determine the best N [2]-[8]. Here in this paper we present an adaptive approach to change the number of particles based on fuzzy logic by monitoring the error of estimation problem.

III. THE FUZZY LOGIC BASED ADAPTIVE PARTICLE FILTER

Particle filtering is a form of optimal estimation characterized by recursive evaluation which requires that all the plant dynamics and noise processes are exactly known as a priori else divergence problems may occur so the optimality of the filter is closely connected to some conditions including the particle numbers. We can readjust the particle numbers based on the information obtained in real time from the measurements as they become available.

One good way to verify whether the filter is performing well enough, is to monitor the residuals. The residuals are the differences between actual measurements and measurement predictions based on the filter's internal model. They reflect the degree of fit between the model and the data and could be used to adapt the filter. Residuals could be defined as in (10) :

$$r_k = y_k - \tilde{y}_k \tag{10}$$

where y_k and $\tilde{y}_k$ denote the actual and estimated outputs of the filter at step number k. Here in this application each of them, refers to the states of navigation system. Now we define average error of estimation through averaging inside a moving estimation window of size M as below :

$$e_{M,k} = \frac{1}{M} \sum_{i=k-M+1}^{k} r_i^T r_i \tag{11}$$

The window size is chosen empirically to give some statistical smoothing. The objective of the adjustments is to change the particle numbers in order to minimize the error signal defined by (11). This adjustment mechanism lends itself very well to be dealt with using a fuzzy-logic based approach as a knowledge-based system, operating on linguistic variables and it's main advantage with respect to more traditional adaptation schemes, are the simplicity of the approach and the application of knowledge about the controlled system.

The main idea of adaptation used by a FIS to dynamically tuning N, is as follows. It can be noted that N could be used in order to reduce the discrepancies between y_k and $\tilde{y}_k$ so if the actual observed error signal value, lies within a predefined range, the filter acts almost perfectly and no change is needed to be made on the value of N. If the actual error is greater than our desired value, the value of N should be increased. On the contrary, when the estimation error is less than its threshold, the value of N could be decreased to save computational effort. The general scheme has been represented in Fig. 2.

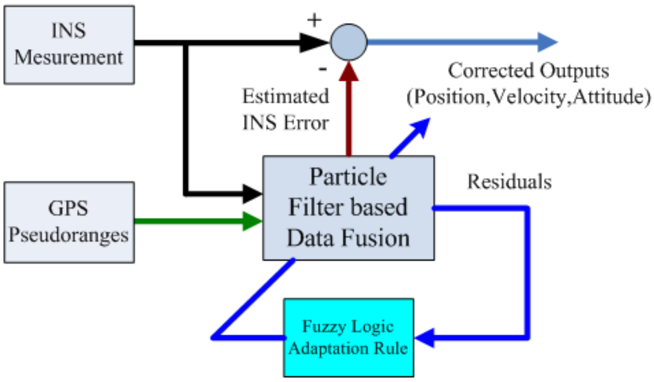

Fig. 2. Tight GPS/INS integration using adaptive fuzzy logic based particle filter

From here, general rules of adaptation are defined as :

I. If $N_k \le N_{\max}$ and $e_{M,k} \ge e_{threshold}$ then increase N_k.

II. If $N_k \ge N_{\min}$ and $e_{M,k} \le e_{threshold}$ then decrease N_k.

III. If $e_{M,k} \cong e_{threshold}$ then keep N_k unchanged.

$N_{\max}$, $N_{\min}$ and $e_{threshold}$ are three predefined parameters of the FIS which the first one relates to the maximum allowable computational effort while the two other supply our desired estimation accuracy. Thus N is adjusted according to :

$$N_k = N_{k-1} + \Delta N_k \tag{12}$$

Additionally a new variable called the Degree of Matching *(DoM)* is defined as (13) :

$$DoM_{M,k} = e_{M,k} - e_{threshold} \tag{13}$$

A Takagi - Sugeno fuzzy system is used to adapt the filter with the general rule as below :

If u_1 *is* $A_{1,i}$ and u_2 *is* $A_{2,i}$ and … then $b_i = g(.)$

So the FIS could be implemented considering fuzzy sets for example three fuzzy sets for *DoM* : *N= Negative*, *Z = Zero*, and *P = Positive* and three fuzzy sets for ΔN_k : *I = Increase*, *M = Maintain*, and *D =Decrease*. These two membership functions are shown in Fig. 3 and Fig. 4. For the functional fuzzy system, we use singleton fuzzification and the center of area *(COA)* defuzzificatiou method. The method of implementation is similar to the Kalman filter enhancement using fuzzy logic [11]-[12].

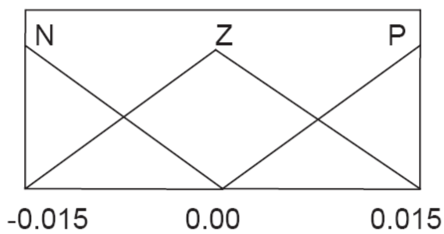

Fig. 3. Membership function for *DoM*

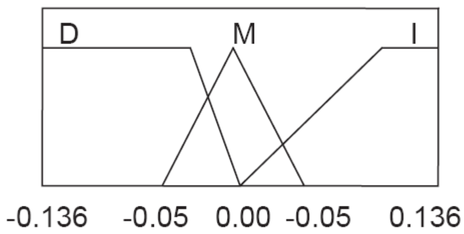

Fig. 4. Membership function for normalized ΔN_k

IV. Mathematical Model of an Integrated GPS/SDINS System

Several mathematical models of different orders have been proposed in order to integrate INS and GPS sensors [1]-[2], [8]-[10]. Here we use the model proposed in [8] and [9]. Note that measurements by accelerometers and gyros are expressed in the platform frame while the GPS measurements are given in an rectangular Earth Centered Earth Fixed (ECEF) frame. The geodetic coordinate system is defined according to the familiar longitude (λ), latitude (ϕ), and height (h) coordinate system so the earth's geoid is approximated by an ellipsoid based on the parameters given in Table I [8]. The relation between these two coordinate system is also given by (14-16).

TABLE I
Parameters of WGS84 Reference Frame

Symbol	Quantity	Value
a	semi major axis	6378.137 km
b	semi minor axis	6345.752 km
ω_{ie}	earth's angular velocity	7.292115×10^{-5}
f	ellipsoid's flatness	3.352511×10^{-3}
e	ellipsoid's eccentricity	0.05781

$$R_\lambda = a(1-e^2)/\sqrt{\{1-e^2\sin^2(\phi)\}^3} \quad (14)$$

$$R_\phi = a/\sqrt{1-e^2\sin^2(\phi)} \quad (15)$$

$$\begin{pmatrix} X_m \\ Y_m \\ Z_m \end{pmatrix} = \begin{pmatrix} (R_\phi+h)\cos(\lambda)\cos(\phi) \\ (R_\phi+h)\cos(\lambda)\sin(\phi) \\ (R_\phi(1-e^2)+h)\sin(\lambda) \end{pmatrix} \quad (16)$$

For simplicity we assume that the gyros and the accelerometers are aligned with the axis in the platform frame. Also we assume that the body frame and the platform frame are aligned, and the center of the coordinate system is the same for both frames. The transformation from body frame to local geographical frame is calculated at every moment and expressed as below :

$$dR_{b2g} = R_{b2g}\Omega_{gb}^b dt \quad (17)$$

where :

$$\Omega_{gb}^b = \begin{pmatrix} 0 & -r & q \\ r & 0 & -p \\ -q & p & 0 \end{pmatrix} \quad (18)$$

$$\omega_{gb}^b = \begin{pmatrix} p \\ q \\ r \end{pmatrix} = \begin{pmatrix} \tilde{p} \\ \tilde{q} \\ \tilde{r} \end{pmatrix} + \begin{pmatrix} b_p \\ b_q \\ b_r \end{pmatrix} - R_{g2b}\begin{pmatrix} \omega_{ie}\cos(\phi) + V_E/(R_\lambda + h) \\ -V_N/(R_\phi + h) \\ -\omega_{ie}\sin(\phi) + V_E\tan(\phi)/(R_\lambda + h) \end{pmatrix} \quad (19)$$

ω_{gb}^b is the inertial angular rate expressed in the body frame where $(\tilde{p}\ \tilde{q}\ \tilde{r})^T$ is the measured angular rate and $(b_q\ b_p\ b_r)^T$ is the bias in the angular rate measurement.

The GPS receiver receives the signal corrupted by noise and other sources of error. If we could neglect the ionospheric and tropospheric errors, the observation equations or pseudoranges provided by i^{th} GPS satellite, have the following form :

$$\rho_i = \sqrt{(X_{si} - X_m)^2 + (Y_{si} - Y_m)^2 + (Z_{si} - Z_m)^2} + c\delta \quad (20)$$

where (X_m, Y_m, Z_m) and (X_{si}, Y_{si}, Z_{si}) are the coordinates of the receiver and the i^{th} satellite respectively. c is speed of the light and δ equals clock drift. Finally the GPS clock drift and the SDINS equations constitute key dynamics in an integrated SDINS/GPS system as in (21-26) :

$$d\begin{pmatrix} V_N \\ V_E \\ V_D \end{pmatrix} =$$

$$\begin{pmatrix} -V_E^2\tan(\phi)/(R_\lambda+h) - 2\omega_{ie}\sin(\phi)V_E + V_N V_D/(R_\phi+h) \\ V_E V_N\tan(\phi)/(R_\phi+h) + \omega_{ie}\{\sin(\phi)V_N + 2\cos(\phi)V_D\} + V_E V_D/(R_\lambda+h) \\ -V_N^2/(R_\phi+h) - 2\omega_{ie}\cos(\phi)V_E - V_E^2/(R_\lambda+h) \end{pmatrix} dt$$

$$+ R_{b2g}\left\{\begin{pmatrix} \tilde{a}_u \\ \tilde{a}_v \\ \tilde{a}_w \end{pmatrix} + \begin{pmatrix} b_u \\ b_v \\ b_w \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ g \end{pmatrix}\right\} dt + dw_t^v$$

$$d\begin{pmatrix} \phi \\ \lambda \\ h \end{pmatrix} = \begin{pmatrix} 1/(R_\phi+h) & 0 & 0 \\ 0 & 1/[(R_\lambda+h)\cos(\phi)] & 0 \\ 0 & 0 & -1 \end{pmatrix}\begin{pmatrix} V_N \\ V_E \\ V_D \end{pmatrix} dt$$

$$d\begin{pmatrix} b_u \\ b_v \\ b_w \end{pmatrix} = -a_{acc}\begin{pmatrix} b_u \\ b_v \\ b_w \end{pmatrix} dt + dw_t^{b_{acc}}$$

$$d\begin{pmatrix} b_p \\ b_q \\ b_r \end{pmatrix} = -a_{gyro}\begin{pmatrix} b_p \\ b_q \\ b_r \end{pmatrix} dt + dw_t^{b_{gyro}}$$

$$dx_t = -a_{clock}x_t dt + dw_t^x$$

$$d\delta = x_t dt \quad (21\text{-}26)$$

where g is the gravitational acceleration, $(\tilde{a}_u\ \tilde{a}_v\ \tilde{a}_w)^T$ is the accelerometer measurement expressed in the body frame, $(b_u\ b_v\ b_w)^T$ is the accelerometer measurement bias again expressed in the body frame. w_t^v , w_t^x , $w_t^{b_{acc}}$ and $w_t^{b_{gyro}}$ are vectors of brownian motion process with zero means and known covariance matrixes. The clock drift δ is represented by the integration of an exponentially correlated random process x_t .

V. Simulation Results

The simulation was made with a relatively low IMU sample rate, 10 Hz in order to speed up the simulation. Pseudoranges are available with the rate of 1 Hz. The kinematic data used were generated by Satnav toolbox created by GPSoft. In our test, the following profile as Fig. 5, containing one pitch manoeuvre in the beginning and one 90 degree turn in the middle of scenario, was used. The numerical parameters for GPS/SDINS mathematical modelling has been represented in appendix. Based on experimentation, it was found that a good size for moving windows was $M=25$.

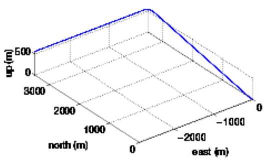

Fig. 5. The flight scenario

Two experiments has been carried out to compare generic particle filter with its adaptive version. The results clearly demonstrated that adaptive changing of particle numbers, would cause to achieve more resolution. The following figures represent the condition of horizontal position and velocity estimation, using the two approaches. Numerical results are also given in Table II and III. It was assumed for the generic particle filter that $N = 250$ while for the fuzzy adaptation : $N_{\max} = 800$ and $N_{\min} = 50$.

Note that by increasing the value of *N*, simple particle filter could produce much more better results itself, but here in this article our focus is not on absolute increase of the particle numbers.

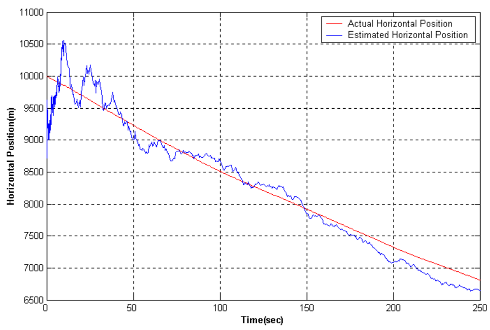

Fig. 6. Horizontal position estimation using generic particle filter

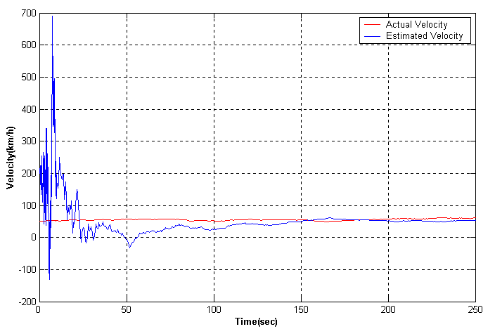

Fig. 7. Velocity estimation using generic particle filter

TABLE II
COMPARISON OF POSITION RMSE OF THE TWO APPROACHES

	Horizontal Position	Vertical Position
Generic Particle Filter	22.6	14.12
Proposed Particle Filter	7.41	3.27

TABLE III
COMPARISON OF VELOCITY RMSE OF THE TWO APPROACHES

	V_N	V_E	V_D
Generic Particle Filter	19.94	27.91	1.65
Proposed Particle Filter	14.46	18.05	0.86

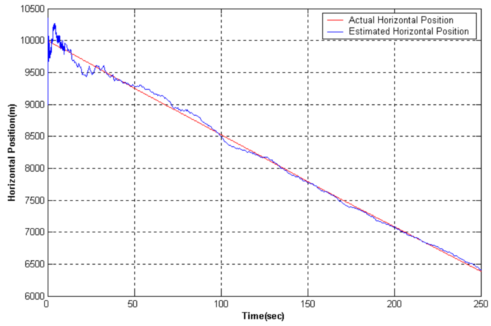

Fig. 8. Horizontal position estimation using adaptive fuzzy particle filter

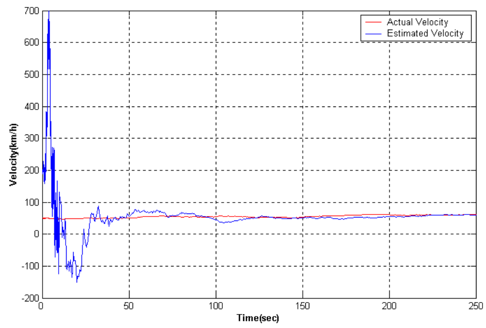

Fig. 9. Velocity estimation using adaptive fuzzy particle filter

Finally a complete GPS signal outage of 120 seconds starting at time 450 was intentionally introduced within the GPS data and both algorithms were used to predict the INS dynamic, during this period. In order to reduce the computational effort, the estimation was performed with constant particle number of *N*=2000 till the time 450. The RMSE of the two networks during this period are compared in Table IV.

It is obvious that the second solution has a better performance than the generic particle filter as a result of fuzzy adaptation with $N_{\max} = 10000$ and $N_{\min} = 500$. In this case during the signal blockage, adaptive particle filter needs a computational effort, nearly 7.36 times larger than the simple particle filter with $N_{average} = 6814$, but based on the results, an improvement index of 58.2% in position estimation could be achieved.

This fact could also be seen in Fig. 10. and Fig. 11. The first graph relates to the generic particle filter while the second one refers to the adaptive algorithm : their position errors and 3σ limits.

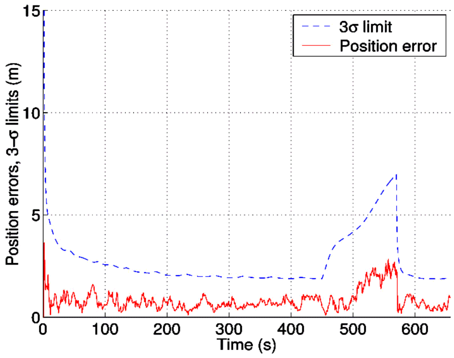

Fig. 10. Position estimation error using generic particle filter (N=2000)

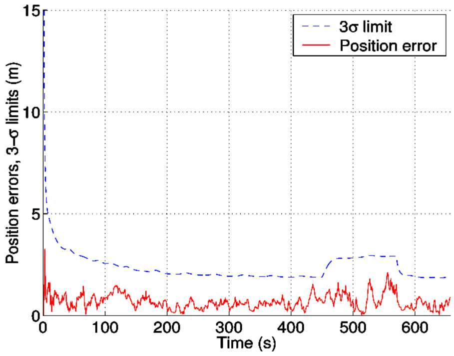

Fig. 11. Position estimation error using adaptive fuzzy particle filter

TABLE IV

POSITION RMSE OF THE NETWORKS DURING SATELLITES OUTAGE

	RMSE(m)
Generic Particle Filter	2.684
Proposed Particle Filter	1.123

VI. CONCLUSION

In this paper, fuzzy logic applied to the particle filter based navigation system as a method to improve the estimation problem. Obtained results demonstrated the improved performance of this method over conventional particle filter as a direct benefit of variable-size particle filter. Although the proposed solution needs more computation effort, but it assures reaching the desired error threshold with the least computational burden and shows outstanding performance in critical situations such as satellites outage, which is much likely in land navigation. Other adaptation rules will be considered in the future researchs.

APPENDIX

The following parameters were used for GPS/SDINS mathematical modelling :

$a_{clock} = 0.002$, $\sigma_{w^x} = 10^{-12}$, $a_{gyro} = a_{acc} = 0.0015$, $\sum_{w^v} = 10^{-5} I$, $\sum_{w^{b_{gyro}}} = 0.09 I$, $\sum_{w^{b_{acc}}} = 4.905 \times 10^{-4} I$.

REFERENCES

[1] J. A. Farrel and M. Barth, The global positioning system and inertial navigation, New York: McGraw-Hill, 1999.

[2] H. Carvalho, P. Del Moral, A. Monin and G. Salut, “Optimal nonlinear filtering in GPS/INS integration,” *IEEE Trans. Aerosp. Electron. Syst.*, vol 33, no. 3, July 1997, pp. 835-850.

[3] J. H. Kotecha and P. M. Djuric, “Gaussian particle filtering,” *IEEE Trans. Sig. Proc.*, vol 51, no 10, October 2003, pp. 2592- 2601.

[4] S. Arulampalam ,S. Maskell ,N. Gordon and T. Clapp, “A tutorial on particle filters for on-line non-linear/non-gaussian bayesian tracking”. *IEEE Trans. Sig. Proc.*, vol 50, no 2, February 2002, pp. 174-188.

[5] C. Hue, J. P. Le Cadre and P. Pérez, “Sequential monte carlo methods for multiple target tracking and data fusion,” *IEEE Trans. Sig. Proc.*, vol 50, no 2, February 2002, pp. 309-325.

[6] D. Crisan and A. Doucet, “A survey of convergence results on particle filtering methods for practitioners,” *IEEE Trans. Sig. Proc.*, vol 50, no 3, March 2002, pp. 736-746.

[7] P. Fearnhead and M. College, “Sequential monte carlo methods in filter theory,” Ph.D. Dissertation, Oxford Univ., 1998.

[8] B. Azimi Sadjadi and P. S . Krishmaprasad, “Approximate Nonlinear Filtering and its Application in Navigation,” Ph.D. Dissertation, Dept. Elec. Eng., Maryland Univ., College Park, 2001.

[9] A.Asadian, B. Moshiri and A. Khaki Sedigh, “Nonlinear optimization in an integrated GPS/INS system in critical situations using particle filters”, in *Proc ICEE2005 Conf.*, Zanjan, Iran, May 2005, pp. 93-99.

[10] B. Boberg and S. L. Wirkander, “Integrating GPS and INS: comparing the Kalman estimator and particle estimator,” in *Proc 7th IEEE Int. Conf. on Control, Automation, Robotics and Vision (ICARCV 2002)*, Singapore, December 2002, pp. 484-490.

[11] W. Leach, J. Dillon, Clarence W. De Silva and R. Rahbari, “Adaptive tuning of a Kalman filter using the fuzzy integral for an intelligent navigation system,” in *Proc IEEE Int. Symp. on intelligent Control (ISIC 2004)*, Vancouver, Canada, October 2002, pp. 252-257.

[12] J. Z. Sasiadek, Q. Wang and M. B. Zeremba, “Fuzzy adaptive Kalman filtering for INS/GPS data fusion,” in *Proc IEEE Int. Symp. on intelligent Control (ISIC 2000)*, Rio, Patras, Greece, July 2000, pp. 181-186.