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 filter has been used on a simplified 2-dimensional 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.

![Particle Filter based Data Fusion](image)

This solution is 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 widespread 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 including the set of received measurements.
In principle an optimal estimate of
the estimation problem. In principle an optimal estimate of
information, it may be said to be the complete solution to
this problem when some of the particles have zero weights
approach leads to divergence and suffers from degeneracy
used in [4], this numerical instability
can be removed. Finally the
corresponding matrix. Finally the main idea is to
approximate \( P(x_{k+1} | y_{1:k}) \) with a sum of delta-Dirac
functions located in the \( x_k^{(i)} \) samples. Using the
importance weights the posterior can be written as :

\[
P(x_{0:k} | y_{1:k}) = \sum_{i=1}^{N} \omega_k^{(i)} \delta(x_{0:k} - x_k^{(i)})
\]

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 \( x_0^{(i)} \)\n
from the initial distribution \( P(x_0) \).
2. Update the weights for \( i=1:N \) which is including :
   - Computation : \( \omega_k^{(i)} = P(y_k | x_k^{(i)}) \)
   - Normalization : \( \frac{\omega_k^{(i)}}{\sum_{j=1}^{N} \omega_k^{(j)}} \)
3. Generate a new set \( \{x_k^{(j)}\}_{k=1}^{N} \) by resampling with
   replacement \( N \) times from \( \{x_k^{(i)}\}_{i=1}^{N} \) with the
   following probability :
     \[
     P(x_k^{(j)} | x_k^{(i)}) = \frac{1}{P_k^{(i)} + \delta_{ij}}
     \]
   \( i = 1:N \) \( \delta_{ij} \) is the Kronecker
4. Compute the number of effective weights :
   - \( N_{\text{eff}} = 1/\sum_{i=1}^{N} (\omega_k^{(i)})^2 \)
   - if : \( N_{\text{eff}} \leq N_{\text{th }} \) ( \( N_{\text{th}} = \frac{2N}{3} \) ) reset the
     weights to : \( \omega_k^{(i)} = \frac{1}{N} \)
5. Predict new particles using (1) and different noise
   realizations for the particles :
     - \( x_{k+1}^{(i)} = f_k(x_k, u_k, w_k) \)
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):

$$y_k = y_k - \hat{y}_k$$  \hspace{1cm} (10)

where $y_k$ and $\hat{y}_k$ 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:

$$e_{M,k} = \frac{1}{M} \sum_{i=k-M+1}^{k} y_i^T r_i$$  \hspace{1cm} (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 $\hat{y}_k$ 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 $N_k \leq N_{\text{max}}$ and $e_{M,k} \geq e_{\text{threshold}}$ then increase $N_k$.

II. If $N_k \geq N_{\text{min}}$ and $e_{M,k} \leq e_{\text{threshold}}$ then decrease $N_k$.

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

$N_{\text{max}}$, $N_{\text{min}}$ and $e_{\text{threshold}}$ 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:

$$N_k = N_{k-1} + \Delta N_k$$  \hspace{1cm} (12)

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

$$\text{DoM}_{M,k} = e_{M,k} - e_{\text{threshold}}$$  \hspace{1cm} (13)

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

If $u_1$ is $A_{1,1}$ and $u_2$ is $A_{2,2}$ and so... 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 $\Delta N_k$: $I$ = 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) defuzzification method. The method of implementation is similar to the Kalman filter enhancement using fuzzy logic [11]-[12].

**Fig. 3. Membership function for DoM**
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 gyroscopes 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 ($\phi$), latitude ($\lambda$), 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).

$$R_k = a(1-e^2)^3 / [1 - e^2 \sin^2(\phi)]^{3/2}$$  \hspace{1cm} (14)

$$R_k = a / \sqrt{1 - e^2 \sin^2(\phi)}$$  \hspace{1cm} (15)

$$\begin{bmatrix} X_w \\ Y_w \\ Z_w \end{bmatrix} = \begin{bmatrix} (R_k + h) \cos(\lambda) \sin(\phi) \\ (R_k + h) \sin(\lambda) \sin(\phi) \\ (R_k(1-e^2) + h) \sin(\lambda) \end{bmatrix}$$  \hspace{1cm} (16)

For simplicity we assume that the gyro 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:

$$\begin{pmatrix} p \\ q \\ r \end{pmatrix} = R_{b2g} \begin{pmatrix} \tilde{p} \\ \tilde{q} \\ \tilde{r} \end{pmatrix} + R_{b2g} \begin{pmatrix} b_p \\ b_q \\ b_r \end{pmatrix} \begin{pmatrix} 0 \\ q \\ p \end{pmatrix}$$  \hspace{1cm} (17)

$$\begin{pmatrix} b_p \\ b_q \\ b_r \end{pmatrix} = \begin{pmatrix} a_w \\ a_v \\ a_u \end{pmatrix}$$  \hspace{1cm} (18)

$$\begin{pmatrix} \phi \\ h \end{pmatrix} = \begin{pmatrix} 1/(R_k + h) \\ 0 \\ 0 \end{pmatrix} V_N dt$$  \hspace{1cm} (19)

$$\begin{pmatrix} \phi \\ h \end{pmatrix} = \begin{pmatrix} 0 \\ 1/[(R_k + h) \cos(\phi)] \\ 0 \end{pmatrix} V_E dt$$  \hspace{1cm} (20)

$$dX = -a_{acc} \omega_{acc} \omega_{acc} dt + d\omega_{acc}^c$$  \hspace{1cm} (21)

$$d\delta = g \omega \omega \omega + d\omega \omega \omega$$  \hspace{1cm} (22)

$$dt = x_t dt$$  \hspace{1cm} (23)

where $g$ is the gravitational acceleration, $(\tilde{u}, \tilde{v}, \tilde{w})$ is the accelerometer measurement expressed in the body frame, $(\tilde{u}, \tilde{v}, \tilde{w})$ is the accelerometer measurement bias again expressed in the body frame, $w_t^c$, $w_t^s$, $w_t^{bias}$ are vectors of brownian motion process with zero means and known covariance matrices. The clock drift $\delta$ 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](image)

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 : $N_{\text{max}} = 800$ and $N_{\text{min}} = 50$.

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](image)

![Fig. 7. Velocity estimation using generic particle filter](image)

![Fig. 8. Horizontal position estimation using adaptive fuzzy particle filter](image)

![Fig. 9. Velocity estimation using adaptive fuzzy particle filter](image)

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.

<table>
<thead>
<tr>
<th></th>
<th>$V_N$</th>
<th>$V_E$</th>
<th>$V_D$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Generic Particle Filter</td>
<td>19.94</td>
<td>27.91</td>
<td>1.65</td>
</tr>
<tr>
<td>Proposed Particle Filter</td>
<td>14.46</td>
<td>18.05</td>
<td>0.86</td>
</tr>
</tbody>
</table>

It is obvious that the second solution has a better performance than the generic particle filter as a result of fuzzy adaptation with $N_{\text{max}} = 10000$ and $N_{\text{min}} = 500$. 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 $N_{\text{average}} = 6814$, 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)](image1)

![Fig. 11. Position estimation error using adaptive fuzzy particle filter](image2)

**Table IV**

<table>
<thead>
<tr>
<th>Position RMSE of the Networks During Satellites Outage</th>
</tr>
</thead>
<tbody>
<tr>
<td>RMSE(m)</td>
</tr>
<tr>
<td>Generic Particle Filter 2.684</td>
</tr>
<tr>
<td>Proposed Particle Filter 1.123</td>
</tr>
</tbody>
</table>

**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:

\[
\begin{align*}
& a_{clock} = 0.002, \quad \sigma_{w} = 10^{-12}, \quad a_{gyro} = a_{acc} = 0.0015, \\
& \sum_{w} = 10^{-5}I, \quad \sum_{w} = 0.09I, \quad \sum_{w} = 4.905 \times 10^{-4}I.
\end{align*}
\]

**REFERENCES**