International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Index Copernicus Value (2013): 6.14 | Impact Factor (2014): 5.611

GPS/INS Integration Using Sigma Point Kalman Filter

Prithviraj Patil[1], Sunita S. Shinde[2]

[1]Department of Electronics and Telecommunication, ADCET, Ashta, MH. India

[2]HOD and Professor, Department of Electronics and Telecommunication, ADCET, Ashta, MH. India

Abstract: ***The accuracy of Global Positioning System (GPS) is increased with the coupling to Inertial Navigation System (INS) to accomplish navigation. This paper proposes an algorithm to filter and fuse the GPS and INS information. Sigma point kalman filter is employed to simulate the information convergence of the dynamic model which maintains better performance in nonlinear system. So we can obtain a better precise filtering result when both are online. At the same time, the INS data is trained with the result as training target when it is the unique input. This paper raises the concept that Support Vector Machine (SVM) is adopted to train the INS data when GPS is offline and the simulated annealing is applied to realize the optimization of the parameters of kernel function and the penalty function in the SVM algorithm. Therefore, the integration navigation could retain almost as precise as the GPS when the GPS is out of coverage***

Keywords: Sigma point Kalman Filter; Localization; GPS; INS; SVM

1. Introduction

Navigation is a field that focuses on the process of monitoring & controlling the movement of craft or vehicle from one place to another. This includes four general categories: Land Navigation, Marine Navigation Aeronautics Navigation Space Navigation. It is also term of art used for the specialized knowledge used by the navigators to perform navigation task. All navigational techniques involve locating the navigator‟s position compared to known locations or patterns. Navigation, in the broad sense, can refer to any skill that involves locating the position & direction. In the sense, navigation includes orienteering & pedestrian navigation.

Global Positioning System (GPS) based navigation systems have been used in Land Vehicle Navigation Systems (LVNS) due to their low price, easy installation, and other beneficial factors. The level of performance required of an LVNS recently increased with the successful implementation of LVNS in unmanned land vehicles, with the development of augmented reality for land vehicles & the availability of high grade LVNSs [1].

From the point of view of different environmental conditions, The Inertial Navigation System (INS) is ideal, rather than using signals from satellites, in case of GPS[2]. 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. 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 in case of 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 [3].

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[4]. 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.Both INS and GPS suffer from various error sources and deficiencies which propel the accompaniment of the two complementary systems. INS exhibit relatively low noisy outputs which tend to drift over time[5]. Contrary to INS, GPS outputs are relatively noisy but do not exhibit long-term drift. Combining both of these systems gives a superior navigation performance than standalone system [6]. So to achieve strong performance, GPS/INS integrated system widely used.But major drawback of GPS/INS integrated system is that some time GPS lost its signals in critical conditions for example in tunnel, hilly areas etc [7]. So at that time it is necessary to some type of mechanism should be adopted to train the INS data during GPS outage. For this purpose Support Vector Machine (SVM) is used to train the INS data during GPS outage & the simulated annealing is applied to realize the optimization of the parameters of the kernel function& the penalty function in the SVM algorithm. Therefore the integrated navigation could retail almost as precise as the GPS when GPS is off-line.

2. Model

Two models are discussed in this section, first one the model of GPS/INS integrated navigation system and the training model of GPS/INS..

A. GPS/INS Integrated Navigation System

The model of the integrated navigation system adopted in this paper is designed by using nonlinear filtering; the errors of the navigation system are selected as system state. They are combined as the state vector that the errors of the attitude, velocity, position, gyroscopes and accelerometers,

X = [ϕE, ϕN, ϕU, δVE, δVN, VU δλ, δL, δh, εbx, εby, εbcεrx, εry, εrz, λx, λy, λz]

where the E, N, U are the describes the east, north and up orientation of the ENU frame respectively. ϕ is the error-angle of the platform, δVrepresents velocity error. And $\delta\lambda$, $\delta L$and $\delta h$are the longitude, latitude and height value respectvely. $\varepsilon$is the gyroscopes drift.

Then model can be defined as follows:

X = F(t)X(t) + G(t)W(t),

Z(t) = H(t)X(t) + v(t),

where*F*(*t*) is the non-linear transition matrix ofthe error model, and *G*(*t*) is the noise driving matrix. *H*(*t*) is the measurement matrix.

The H(t) and v(t) are defined as the following, where v(t) is the measurement noise matrix.

Hv(t) = [03×6, diag{111}, 03×9],

vv(t) = [δvEG, δvNG, δvUG].

B. The Training Model

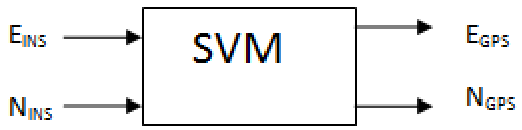

Figure 1: The SVM training model.

At first the progress of the dynamic system is described as the following. The velocity signal of north and east E_{GPS} N_{GPS} from GPS is trusted as real speed when four or more GPS satellites are in the sight. And at the same time, the model is trained with the velocity signal of north and east from the INS E_{INS} N_{INS} and time t as the input, and the VEGPS and VNGPS as the output.

3. Algorithm

This section implies , the algorithms applied, & optimized SVM are detailed specially.

4. Methodology

The INS data are divided into 2 groups, first one of which is used to obtain the online optimum value by the UKF with the corresponding GPS data, and applied as training data along with the optimum value, and the second group is used to testify the regression. The methodology is expressed in the Fig.2. After optimization of each filtering step is deduced by the UKF, the final output is determined by the GPS data and the optimum error data collectively. In case of GPS outage, the regression model of INS and the final output is necessary which is accomplished training by the SVM

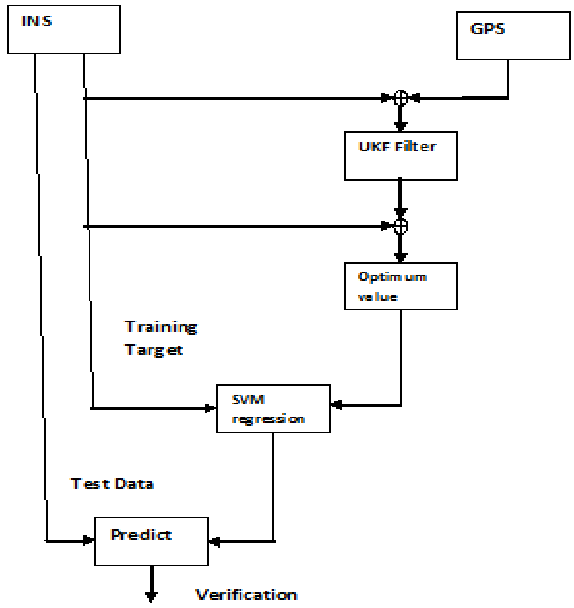

Figure 2: Methodology Process

5. Sigma Point Kalman Filter

Introduction of Sigma Point Kalman Filter (GPS/INS Integration)

Instead of linearization required by EKF, the SPKF does not require to approximate nonlinear system dynamic and measurement models using the Jacobian in order to calculate the covariance of random vector propagated through the nonlinear models. Instead, a set of deterministically selected sigma-points is chosen, which have the same mean and covariance as the original random vector. Then, these sigma points are propagated through the nonlinear models, and the mean and the covariance accurately to the second order for arbitrary nonlinear functions, while the EKF achieves first order accuracy only. The different types of sigma-point filters, such unscented Kalman filter or central difference Kalman filter, are distinguished by the weights and the scaling parameter associated with the sigma-points. Throughout this paper, the unscented Kalman filer is used.

However, UKF has been designed for many practical systems that have non-linear state update and/or measurements equations [20]. In addition, the performance of the EKF heavily depends on how the system dynamics and measurements are modelled [21]. The state transition equation can be expressed as:

$X_K = f(X_{K-1},U_{K-1}) + W_K.$

The non-linear state function f is used to determine the predicted state from the previous state. The observation vector can be expressed as function of the state through the measurement equation:

$Z_K = h(X_K) + V_K.$

The observation function relates the measurements with the true state . As mentioned before the UKF algorithm is composed of two main phases: „predict phase" and „update phase" described as follows.

Predict Phase

The predict phase is used to predict the a priori state vector on the basis of the previous a posteriori estimated state vector. Considering a PV model for the state equation in 2-dimensional case, the state vector can be defined as X_k= [$X_k Y_k$. X_k . Y_k] where .$X_K$and .$Y_K$denote the speed of the mobile target node along the X &Y axis, respectively. Accordingly, the matrix can be defined as follows:

$$F_K = \begin{bmatrix} 1 & 0 & \Delta tk & 0 \\ 0 & 1 & 0 & \Delta tk \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Where Δt_k represents the time elapsed between the previous estimation time and the current one. The estimated covariance matrix related to the current a priori state vector is evaluated from the previous a posteriori covariance matrix and the process noise covariance matrix.

The Q matrix takes into account un-modeled factors of the system. For , it can be defined as follows:

$$Q = A.\begin{bmatrix} 6x^2 & 0 \\ 0 & 6y^2 \end{bmatrix}.A^T$$

Where $б^2_x$ and $б^2_y$ denote the variances of the acceleration noise along the and axis, respectively. The matrix A can be defined as:

$$A = \begin{bmatrix} \frac{\Delta Tk}{2} & I2 \\ \Delta Tk & I2 \end{bmatrix}$$

Where I_2 represents the identity matrix of dimensions *2*. The covariance matrix P_0 related to the initial state vector X_0 , for the PV model in 2-dimensional case can be defined as:

$$P_{0=} \begin{bmatrix} 6x^2 & 0 & 0 & 0 \\ 0 & 6y^2 & 0 & 0 \\ 0 & 0 & 6x0^2 & 0 \\ 0 & 0 & 0 & 6y0^2 \end{bmatrix}.$$

Where $б^2_x$ and $б^2_y$ represent the initial variances of the state vector components.

Update Phase

The update phase, also called correction phase, further refines the a priori position estimate by using the observation vector Z_k. First of all, the innovation vector Y_k is calculated as the residual between the observed measurement Z_k and the expected measurement $h(X_k)$:

$Y_k = Z_k - h(X_k)$

The covariance matrix S_k of the innovation vector can be computed as follows:

$S_K = H_K.P_{K|K-1} + R_K.$

Where R_k represents the covariance matrix related to the observation vector and H_k represents Jacobian matrix related to expected measurements.

Finally, the a posteriori state covariance matrix P_k computed by correcting the a**Support Vector Machine (SVM):**

Introduction

SVMs have been developed in the reverse order to the development of neural networks (NNs). SVMs evolved from the sound theory to the implementation and experiments, while the NNs followed more heuristic path, from applications and extensive experimentation to the theory. It is interesting to note that the very strong theoretical background of SVMs did not make them widely appreciated at the beginning. The publication of the first papers by Vapnik, Chervonenkis (Vapnik and Chervonenkis, 1965) and co-workers went largely unnoticed till 1992. This was due to a widespread belief in the statistical and/or machine learning community that, despite being theoretically appealing, SVMs are neither suitable nor relevant for practical applications. They were taken seriously only when excellent results on practical learning benchmarks were achieved (in numeral recognition, computer vision and text categorization). Today, SVMs show better results than (or comparable outcomes to) NNs and other statistical models, on the most popular benchmark problems.

The learning problem setting for SVMs is as follows: there is some unknown and nonlinear dependency (mapping, function) $y = f(\mathbf{x})$ between some high-dimensional input vector **x** and scalar output y (or the vector output **y** as in the case of multiclass SVMs). There is no information about the underlying joint probability functions here. Thus, one must perform a *distribution-free learning*. The only information available is a training data set

Note that this problem is similar to the classic statistical inference. However, there are several very important differences between the approaches and assumptions in training SVMs and the ones in classic statistics and/or NNs modeling. Classic statistical inference is based on the following three fundamental assumptions

Data can be modeled by a set of linear in parameter functions; this is a foundation of a parametric paradigm in learning from experimental data

In the most of real-life problems, a stochastic component of data is the normal probability distribution law, that is, the underlying joint probability distribution is a Gaussian distribution.

Because of the second assumption, the induction paradigm for parameter estimation is the maximum likelihood method, which is reduced to the minimization of the sum-of-errors-squares cost function in most engineering applications.

All three assumptions on which the classic statistical paradigm relied turned out to be inappropriate for many contemporary real-life problems (Vapnik, 1998) because of the following facts:

1) Modern problems are high-dimensional, and if the underlying mapping is not very smooth the linear paradigm needs an exponentially increasing number of terms with an increasing dimensionality of the input space *X* (an increasing number of independent variables). This is known as „the curse of dimensionality".
2) The underlying real-life data generation laws may typically be very far from the normal distribution and a model-builder must consider this difference in order to construct an effective learning algorithm.
3) From the first two points it follows that the maximum likelihood estimator (and consequently the sum-of-error-

squares cost function) should be replaced by a new induction paradigm that is uniformly better, in order to model non-Gaussian distributions.

In addition to the three basic objectives above, the novel SVMs‟ problem setting and inductive principle have been developed for standard contemporary data sets which are typically high-dimensional and sparse (meaning, the data sets contain small number of thetraining data pairs). SVMs are the so-called „nonparametric‟ models. „Nonparametric‟ does not mean that the SVMs‟ models do not have parameters at all. On the contrary, their „learning‟ (selection, identification, estimation, training or tuning) is the crucial issue here.

6. Simulation

This chapter describes the implementation of algorithms and theoretical work which has been explained in previous chapter. First implementation of sigma point Kalman Filter is explained then effect of multipath environment on sigma point Kalman Filter is explained. After this the SVM explained which has been used to predict the multipath environment.

Simulation Scenario in MATLAB:

All work is done in MATLAB 7.9. The simulation scenario consists of a straight 5 km road on which vehicles are moving.Vehicle travelling along a straight road consist different regions (environments) such as an open environments and multipath environments

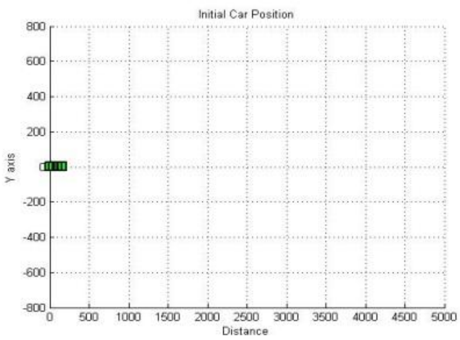

Figure 3: Initial Position of the Vehicles

There are total five regions are shown which consists of open as well as multipath environment. Particularly, an open environment does not consist of multipath effect while multipath environment consist of multipath effect because it consisting inner city area and where high buildings are located. The vehicles are moving in straight road with constant velocity. The simulation period is equal to total time required for a vehicle to cover a distance of 5 km i.e. 360 seconds (vehicles velocity is 50 km/hr). In simulation five regions are considered with an open as well as multipath environment. The first region consists of open environment which is of 2000 meter. Second region consists of multipath environment which is of 300 meter. Third region consists of open environment of 500 meter, fourth region consists of multipath environment which is of 400 meter and last fifth region consists of open environment which is of 1900 meter.

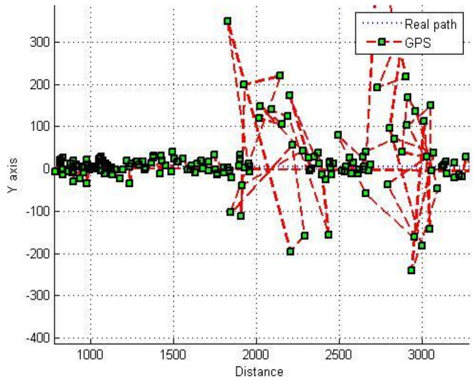

Figure 4: Multipath reflection of GPS data

Sigma Point Kalman Filter Implementation and Results:

The GPS and INS are two important localization techniques. But some errors are presents due to these two techniques when we are going to find localization of a particular node (in this work node is considered as a vehicle). Basically these errors are due to local and global errors of GPS and INS measurement errors. If there is any error due the INS measurement i.e. due to the vehicles velocity or direction the system will be affected. Thus this type of errors can be represented UKF. In a UKF the error in the state is represented by process noise covariance matrix,

$$Q = 6Q^2 I.$$

The value of ζQ is taken as 0.5 m. similarly the local and global errors in the GPS receiver are represented by measurement noise covariance matrix of the Kalman Filter, $R = 6R^2 I$.
Where ζR is set to 10 m to 15 m. It shows the variation of GPS in open environment but the same value in multipath environment is 100 m to 150 m because of loss of GPS signal in multipath environment due to high buildings, number of trees near to the road etc.

The simulation is done by using MATLAB in which we can consider it as the vehicle is moving in same direction with constant velocity. For every second Kalman Filter‚s time update state will estimate location of a vehicle

Figure 6 shows the unscented Kalman Filter estimate localization using loosely coupled approach travelling over 5 km with constant velocity of 50 km/hr. It shows the effect of multipath environment is minimized in the UKF output.

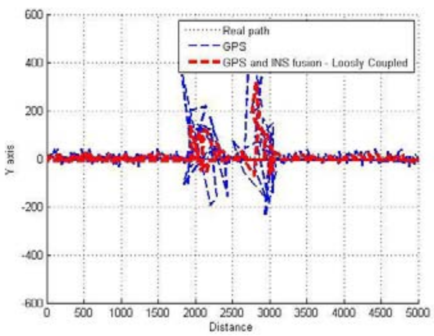

Figure 5: GPS/INS fusion (loosely coupled).

The INS displacement error with respect to real path is shown in below:

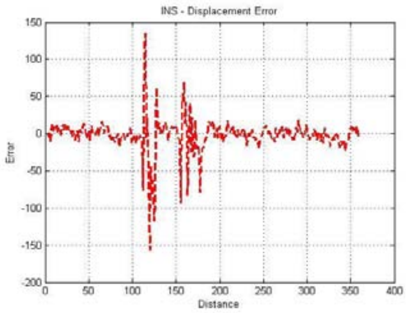

Figure 6: INS displacement error (loosely coupled).

Tightly-Coupled integration eliminates the usage of cascaded filters unlike loosely coupled integration. Therefore, correlation of the measurements in GPS navigation filter is prevented. Instead of navigation solution of GPS, pseudo ranges and pseudo range rates obtained from Doppler data are used as the measurements of the INS/GPS integrated Kalman filter. The difference between tightly coupled & loosely coupled integration is shown in below:

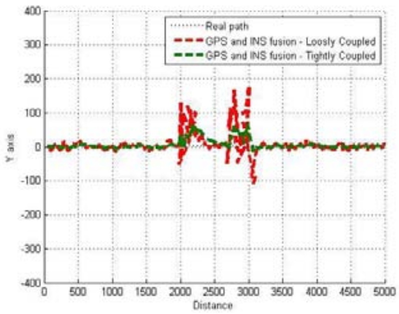

Figure 7: Error diff. between tightly & loosely coupled approach

So we can easily minimized INS displacement error with the help of tightly coupled approach.

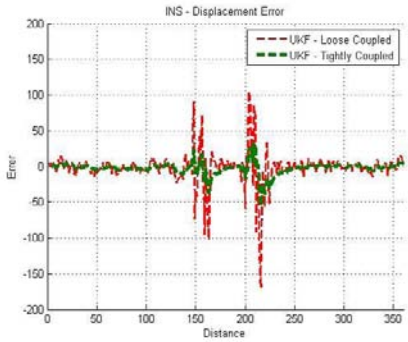

Figure 8: INS displacement error

Choice of these measurements brings nonlinearity to the measurement model of the integrated filter. Even when less than four satellites are available, integration filter will keep on operating since this system does not require a full GPS solution to aid the INS.

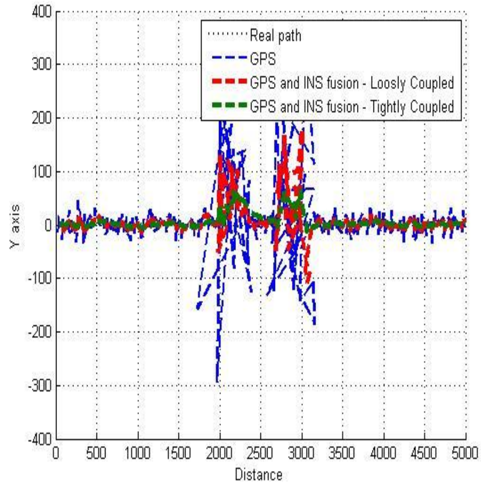

Figure 9: Tightly coupled GPS/INS Integration

There is only INS available during GPS outage; therefore it is necessary to establish a well-trained model to predict the positioning information. Artificial neural network (ANN) can be used for this purpose. In comparison with ANN, Support Vector Machine (SVM) can provide better genetic ability, thus it takes shorter time for training to obtain better training performance. SVM predicts the pattern of GPS data with the help of past samples & INS data.

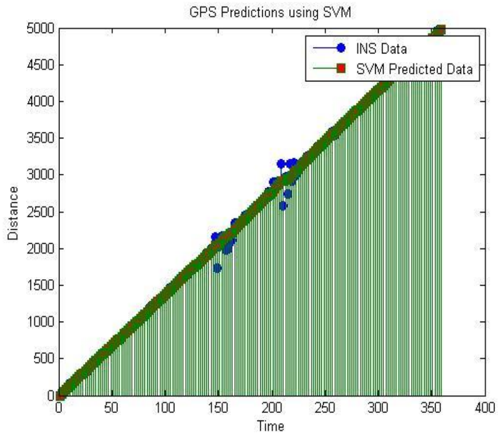

Figure 10: SVM prediction

7. Conclusion

This paper proposes that algorithm of sigma point kalman filter can be implemented in GPS/INS integrated system to perform the better navigation jointly when GPS is online. Then it presents a off line GPS/INS integration model in the case of GPS outage. For that purpose, support vector machine (SVM) is used to train the regression data. From the simulated results we can say that tightly coupled model of GPS/INS navigation system maintains better stability than that of loosely coupled model of GPS/INS navigation system.

References

[1] P. Misra, P. Enge, Global Positioning System: Signals, measurements and performance, 2nd ed., Ganga-Jamuna Press, 2006, pp. 22-25.
[2] M.S.Grewal, L.R.Weill, and A.P.Andrews, Global positioning system and inertial navigation, New York: Wiley, 2001.
[3] S.Y. Cho and H.K. Lee, "Modified RHKF Filter for Improved DR/GPS Navigation against Uncertain Model Dynamics," ETRI J., vol. 34, no. 3, June 2012, pp. 379-387
[4] A. Kleusberg, and R.B. Langley, The limitations of GPS, in GPS World, vol 1(2), pp.50-52, March/April 1990.
[5] Dongliang Huang, Leung, H., and El-Sheimy, N., Expectation maximization based GPS/INS integration for land-vehicle navigation, IEEE Transactions on Aerospace and Electronic Systems.vol. 43(3), pp.1168-1177, July 2007.
[6] J. A. Farrel and M. Barth, The global positioning system and inertial navigaton, New York: McGraw-Hill, 1999.
[7] Sunita S. Shinde, Dr. Ravi M. Yadahalli, Arifa. S. Tamboli ―Vehicular Ad-HocNetwork Localization Techniques: A Review‖ IJECT Vol. 3, Issue 2, April - June 2012.