Milos SOTAK

Armed Forces Academy

Sequential Monte Carlo methods for navigation systems

***Abstract**. The paper deals with new approach to navigation information processing using Sequential Monte Carlo Methods known as particle filtering. Although, the Sequential Monte Carlo Methods require huge amount of computing, these methods are more efficient than Kalman filters especially when the system is nonlinear or if probability density function of the errors is non-Gaussian. The paper presents integration of Inertial Navigation System (INS) and Global Positioning System (GPS) using Sequential Monte Carlo Methods for navigation information processing. Navigation systems were created in simulation environment. An original asset of the work consists in creation of models in the simulation environment to confirm the algorithms.*

***Streszczenie.** W artykule przedstawiono nowy system przetwarzania danych w nawigacji wykorzystujący metodę sekwencyjną Monte Carlo znaną jako częściowe filtrowanie. Chociaż ta metoda wymaga dużych zasobów komputerowych jest jednak bardziej skuteczna niż filtry Kalmana, szczególnie jeśli system jest nieliniowy lub gęstość prawdopodobieństwa nie jest gaussowska. W metodzie wykorzystano też systemy GPS oraz INS. (**Sekwencyjna metoda Monte Carlo w systemach nawigacji**)*

Keywords: INS, GPS, navigation systems, particle filter
Słowa kluczowe: GPS, nawigacja, metoad Monte Carlo.

Introduction

The INS and GPS are widely used navigation systems in several applications. The main reason for their using is their dimensions and weight, and their relatively simple implementation in navigation system. The main objective of the INS/GPS integration is to merge information from INS and GPS sensors and provide estimates of the states of the vehicle with greater accuracy than relying on the information from the individual sensors [7, 11].

The inertial navigation is based on measurements of vehicle specific forces and rotation rates obtained from on-board instrumentations. These measurements are used for determination of a vehicle position, velocity and course using Newton's equations of motion. The on-board sensors consist of accelerometers and gyros, which comprise Inertial Measurement Units. Along with computer hardware and software IMU make up INS.

The INS may be mechanized in either gimbaled or strapdown configuration. Gimbaled systems are usually heavier and more expensive than strapdown systems. This is reason why a strapdown INS are used on UAVs or in other systems where the weight and size plays significant role. The main advantage [4] of using an INS is that all parameters measured by system (i.e. acceleration, angular rotation) are provided at high update rates. However, there are also disadvantages like errors caused by bias in the sensor readings and errors caused by the misalignment of the unit's axes [9]. The accuracy of an INS is therefore highly dependent on the sensor quality, navigation system mechanization and dynamics of the flight vehicle. For proper function the INS has to be initialized when turned on by initialization procedure containing alignment procedure whose is in charge to determine initial attitude [8, 10, 12].

The GPS is a space based radio navigation system. This system can provide high accurate positioning. The main characteristic of the GPS is its bounded errors. Accuracy of the GPS system depends on many factors, for instance receiver clock bias, bias due to receiver clock drift, bias due to system clock error, ionospheric delay, tropospheric delay, random noise, etc. However, compared to INS system, the GPS receiver is low frequency sensor, thus providing the state information at low update rates.

Sequential Monte Carlo Methods

Numerical methods known as Monte Carlo methods can be described as statistical simulation methods, where statistical simulation is defined as a method that utilizes sequences of random numbers to perform the simulation. Despite the fact that Monte Carlo methods are known for such a long time only the nowadays progress in technique allows to apply those methods on complex applications. Monte Carlo methods are now used routinely in many diverse fields from the simulation of complex physical phenomena.

The sequential Monte Carlo approach is known as the bootstrap filtering, the condensation algorithm, and the particle filtering [5]. Particle filters are simulation-based filtering methods where realizations (samples) of the state vector are produced to obtain an empirical approximation of the joint posterior distribution. In fact, particle filters are "tracking" a variable of interest as it evolves over time, typically with a non-Gaussian probability density function. In particle filters the probability density function is calculated using likelihood function. For this reason multiple copies (particles) of the variable of interest are used, each with a specific weight and the variable of interest is then obtained by the weight sum of all the particles, in other word, the normalized importance weight and corresponding particles constitute an approximation of the filtering density [3]. The particle filter is recursive (similarly to LKF and EKF) and operates in two phases: prediction and update. That means that after each operation, each particle is modified according to the variable of interest then its weight is recalculated and particles with small weights are rejected (this process is called resampling).

The best solution, how to utilize advantages and eliminate errors is an integration of INS and GPS sensors. In this system the accuracy of the INS computations is improved by updating those using GPS measurements that are processed by the navigation particle filter.

To achieve correct function of this integrated system it is necessary to focus the estimation of the INS errors. Navigation particle filter computes the position, velocity and attitude errors of the INS. Dynamic of these errors is described by nonstationary error model (equation) (3) and error state vector is defined as

$$\delta \boldsymbol{x}_t^{15x1} = \begin{bmatrix} \delta \boldsymbol{r}^n & \delta \boldsymbol{v}^n & \delta \boldsymbol{\psi}^n & \boldsymbol{b}_{acc} & \boldsymbol{d}_{\omega} \end{bmatrix}^T \quad (1)$$

The model under consideration here is

$$\begin{aligned} \delta \boldsymbol{x}_{t+1} &= \boldsymbol{F}_t \cdot \delta \boldsymbol{x}_t + \boldsymbol{G}_t \cdot \boldsymbol{u}_t \\ \delta \boldsymbol{y}_t &= \boldsymbol{H} \cdot \delta \boldsymbol{x}_t + \boldsymbol{v}_t \end{aligned} \quad (2)$$

and error model [6] is

$$\text{(3)}\quad \boldsymbol{F}_t^{9x9} = \begin{bmatrix} -\boldsymbol{\Omega}_{en}^n & \boldsymbol{I}^{3x3} & \boldsymbol{0}^{3x3} & \boldsymbol{0}^{3x3} & \boldsymbol{0}^{3x3} \\ \Upsilon^e & -\left(2\boldsymbol{\Omega}_{ie}^n + \boldsymbol{\Omega}_{en}^n\right) & \left[\boldsymbol{f}^n \times\right] & \boldsymbol{C}_b^n & \boldsymbol{0}^{3x3} \\ \boldsymbol{0}^{3x3} & \boldsymbol{0}^{3x3} & -\left(\boldsymbol{\Omega}_{ie}^n + \boldsymbol{\Omega}_{en}^n\right) & \boldsymbol{0}^{3x3} & \boldsymbol{C}_b^n \\ \boldsymbol{0}^{3x3} & \boldsymbol{0}^{3x3} & \boldsymbol{0}^{3x3} & \boldsymbol{I}^{3x3} & \boldsymbol{0}^{3x3} \\ \boldsymbol{0}^{3x3} & \boldsymbol{0}^{3x3} & \boldsymbol{0}^{3x3} & \boldsymbol{0}^{3x3} & \boldsymbol{I}^{3x3} \end{bmatrix}$$

where $\delta \boldsymbol{r}^e, \delta \boldsymbol{v}^e, \delta \boldsymbol{\psi}^e$ are errors in position, velocity and angle misalignments respectively, $\boldsymbol{b}_{acc}$, $\boldsymbol{d}_{\omega}$ are a bias of accelerometers and gyroscopes. $\left[\boldsymbol{f}^n \times\right]$ is skew symmetric matrix of specific force in the navigation frame, $\boldsymbol{\Omega}_{ie}^n$ is skew symmetric matrix of earth rotation in navigation frame $\boldsymbol{\Omega}_{en}^n$ is skew symmetric matrix of craft rate, and Υ^e is gravity gradient, $\boldsymbol{C}_b^n$ is transformation matrix from body frame to the navigation frame. Observation matrix is represented by identity matrix $\boldsymbol{H}_t = \boldsymbol{I}^{6x9}$ because states $\delta \boldsymbol{x}_t$ and measurements $\delta \boldsymbol{y}_t$ are expressed in navigation frame. Moreover, $\boldsymbol{w}_t$ and $\boldsymbol{v}_t$ are process and measurement noises respectively with known probability densities $p(\boldsymbol{u}_t)$ and $p(\boldsymbol{v}_t)$.

Particle filter implementation can be described by following algorithm:

1. *Initialization:* Generate $\delta \boldsymbol{x}_0^{(i)} \sim p(\delta \boldsymbol{x}_0)$, $i=1,....,N$ sample of the state vector is referred to as a particle

2. *Measurement update:* Update the weights by the likelihood

$$\boldsymbol{w}_t^{*(i)} = \boldsymbol{w}_{t-1}^{(i)} \cdot p\left(\delta \boldsymbol{y}_t \mid \delta \boldsymbol{x}_t^{(i)}\right) = \boldsymbol{w}_{t-1}^{(i)} \cdot p_{\boldsymbol{v}_t}\left(\delta \boldsymbol{y}_t - h\left(\delta \boldsymbol{x}_t^{(i)}\right)\right) \qquad i=1,....,N$$

Calculate likelihood by

$$p_{\boldsymbol{v}_t}\left(\delta \boldsymbol{y}_t - h\left(\delta \boldsymbol{x}_t^{(i)}\right)\right) = \frac{1}{(2\pi)^{\dim(\delta \boldsymbol{y}_t)}\sqrt{|\boldsymbol{R}|}} \cdot \exp\left[-\frac{1}{2}\left(\delta \boldsymbol{y}_t - h\left(\delta \boldsymbol{x}_t^{(i)}\right)\right)^T \cdot \boldsymbol{R}^{-1} \cdot \left(\delta \boldsymbol{y}_t - h\left(\delta \boldsymbol{x}_t^{(i)}\right)\right)\right]$$

and normalize to

$$\boldsymbol{w}_t^{(i)} = \frac{\boldsymbol{w}_t^{*(i)}}{\sum_{i=1}^{N} \boldsymbol{w}_t^{*(i)}}$$

3. *Resampling:* Replicate particles in proportion to their weights. Only resample as above when the effective number of samples is less than a threshold $N_{threshold}$.

$$N_{eff} = \frac{1}{\sum_{i=1}^{N}\left(\boldsymbol{w}_t^{(i)}\right)^2} < N_{threshold}, \qquad 1 \le N_{eff} \le N$$

where the upper bound is attained when all particles have the same weight, and the lower bound when all probability mass is at one particle. The threshold can be chosen as $N_{threshold} = 2N/3$ [2].

4. *Estimation of states & Prediction of particles:*

For estimation (approximation) of states can be use MMSE (Minimum Mean Square Error Estimate) or MAP (Maximum A Posteriori Estimate) estimators. MMSE estimator was used for its smoothness.

For prediction of particles: Take a $\boldsymbol{u}_t^{(i)} \sim p_{\boldsymbol{u}_t}$ and simulate

$$\delta \boldsymbol{x}_{t+1}^{(i)} = \boldsymbol{F}_t \cdot \delta \boldsymbol{x}_t^{(i)} + \boldsymbol{G}_t \cdot \boldsymbol{u}_t^{(i)} \qquad i=1,....,N$$

5. *Let* $t = t+1$ and iterate to item 2).

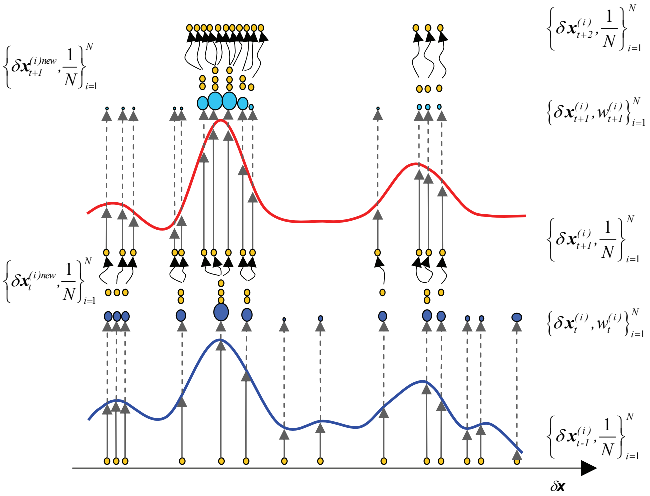

Fig. 1 Resampling

This algorithm was used for navigation system integration. Fig. 2 shows uncoupled integrated navigation system which was implemented in simulation environment.

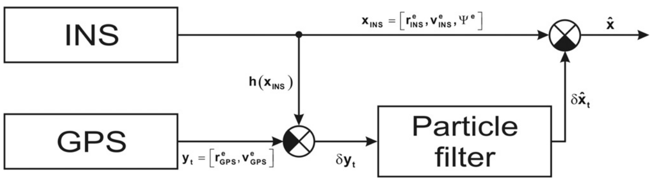

Fig. 2 Uncoupled integrated navigation system

Simulations & Results

Based on the previous chapters the error model (eq. 1-3) of the INS and GPS was created in the simulation environment. As the input in particle filter (Fig.2) was used measurement of the errors in position and velocity. These measurement were generated using models of INS and GPS, where error of the INS consists of accelerometers errors with bias *0,1 mm.s*$^{-2}$ and noise *0,1 mm.s*$^{-2}$*.Hz*$^{-1/2}$, and errors of gyros with bias *0,01 deg.h*$^{-1}$ and noise *0,005 deg.h*$^{-1}$*.Hz*$^{-1/2}$ that are typical values for navigation-grade INS. For GPS model was considered receiver working on C/A code with position error $\sigma_{GPSpos} = 7\ meters$ in all axes and velocity error $\sigma_{GPSvel} = 0.1\ m/s$. Output data of these sensors were expressed in earth frame and were processed by particle filter as was mentioned earlier.

To compare characteristics of the particle filter implementation in INS and GPS integrated system were used two etalon models of the movement:

- Model with low dynamics
- Model with high dynamics

These trajectories are depicted in Fig.4. and Fig.5 respectively. Trajectory with low dynamics is characterized by minimum changes of movement parameters, for example: maximum speed is only *15 m.s*$^{-1}$ also accelerations in all directions are small. For movement with high dynamics was considered object with higher speed and with dynamics changes of accelerations in north and east directions.

The trajectory of this object was evaluated by model of the INS and GPS sensors, respectively. To compare results of the INS/GPS integration using particle filter the LKF based INS and GPS integrated navigation system was used. Using of the LKF was based on fact that similar error models could be used and outputs of the estimators are similar as well.

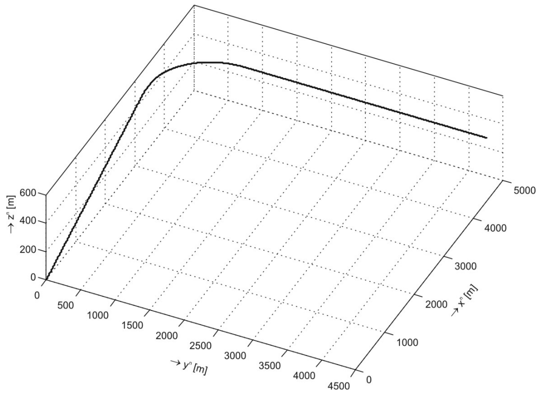

Fig. 3. Etalon trajectory (Low dynamics)

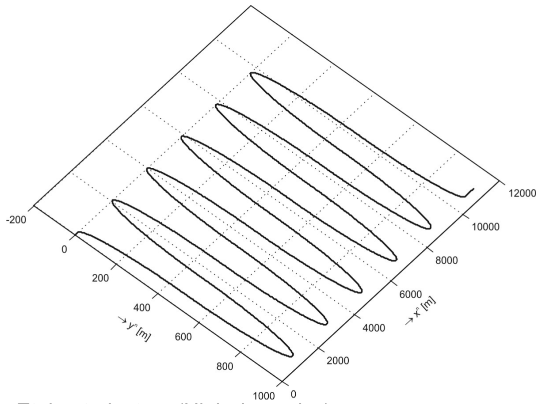

Fig. 4. Etalon trajectory (High dynamics)

As results, the statistic data RMS of error (difference etalon trajectory and trajectory indicated by systems respectively), and maximum error during tests were monitored. According to the table 1, after simulation, that took *600 seconds*, the INS maximum error in was *-16 meters*, which growing in the time. The GPS maximum error was approximately *24 meters*. The maximum error for integrated system based on the linearized Kalman filter was approximately *9 meters* and RMS of error was around *3 meters*. These results show that accuracy of the integrated INS and GPS system using LKF increased approximately 2 times with respect to GPS.

These results were used to compare with integrated INS and GPS navigation system using particle filter. The maximum error of this system (for 50000 particles) was approximately *3.9 meters* and RMS of error was *1.3 meters*. It can be said that accuracy increased approximately 5 times comparing these results to results of the standalone GPS system. All results are depicted in table 1, where is also shown that accuracy of particle filter depend on the number of using particles. However, different between particle filter using 50000 particles and for example 20000 particles are not so noticeable.

The Fig. 5 shows different between position error in system using LKF (red solid line), system using PF (blue solid line) and INS (black solid line) respectively. The error of the system using LKF grew in time, when dynamics of the object were changed e.g. in time *250 seconds* from start of simulation. This error growth was operated using linearized model in LKF.

The estimations of the state vector $\delta\hat{\boldsymbol{x}}_{MMSE}$ (in axis it is $\delta\hat{y}^e_{MMSE}$) are computed from particles that represents calculated errors of the state vector $\delta \boldsymbol{x}_t^{(i)}$ (in axis y it is δy^e) using expression $\Delta y^e = \left[y^e_{etalon} \text{-} (y^e_{INS} \text{-} \delta\hat{y}^{e\,PF}_{MMSE}) \right]$. In Fig. 6 is shown evaluation of 50000 particles every *15 seconds* of total time of simulation.

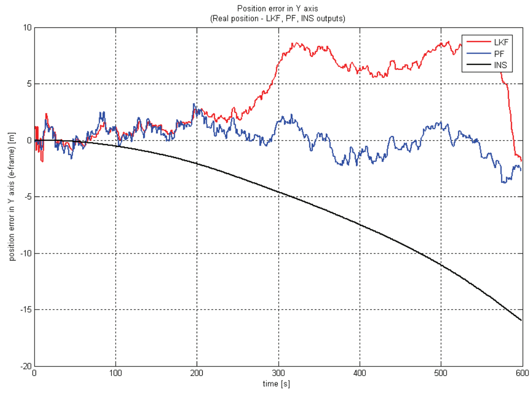

Fig. 5. Position error in Δy^e axis

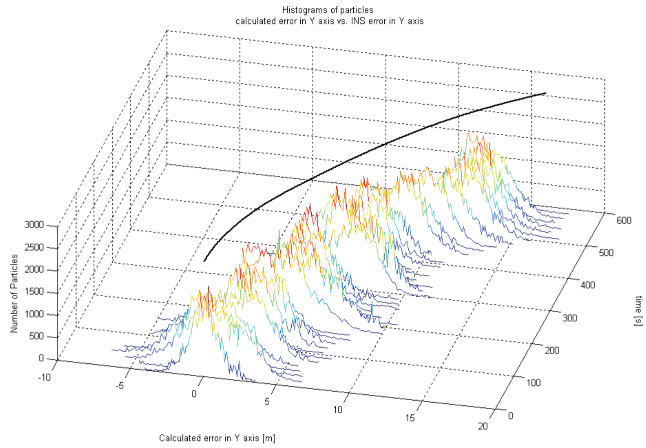

Fig. 6. Histograms of particles – calculated errors in δy^e, solid line is INS error in y^e

Table 1. Test results for low dynamics movement

Results for Low dynamics movement	INS			GPS			LKF		
	x^e	y^e	z^e	x^e	y^e	z^e	x^e	y^e	z^e
RMS [m]	-	-	-	6.91	7.03	7.32	1.00	3.16	3.73
maximum error [m]	17.88	-16.00	-18.01	22.84	24.03	24.31	2.65	8.75	9.63
Filters	PF 1000			PF 20000			PF 50000		
	x^e	y^e	z^e	x^e	y^e	z^e	x^e	y^e	z^e
RMS [m]	1.34	3.86	1.44	1.22	1.36	1.26	1.07	1.36	1.22
maximum error [m]	6.50	6.14	4.26	4.07	4.51	4.52	3.56	3.39	3.98

In Fig. 7 and Fig. 8 are shown the results of the error estimation using particle filter with 50000 particles for the high dynamics movement. It is clear that linearized Kalman filter has greater problems with this kind of movement. These problems come out from linearization of error model. Comparing these results with system using the particle filter it is clear, the particle filter still yields very accurate range estimates.

The summary of high dynamics movement results for integrated INS and GPS navigation system using linearized Kalman filter and particle filter respectively is in table2. From results is clear that integrated systems using particle filter has advance against nowadays used systems based on linearized Kalman filter or extended Kalman filter.

System integration using particle filter is suitable for all applications, but it is necessary use hardware with high performance because procedures works with huge matrixes and huge amount of computation.

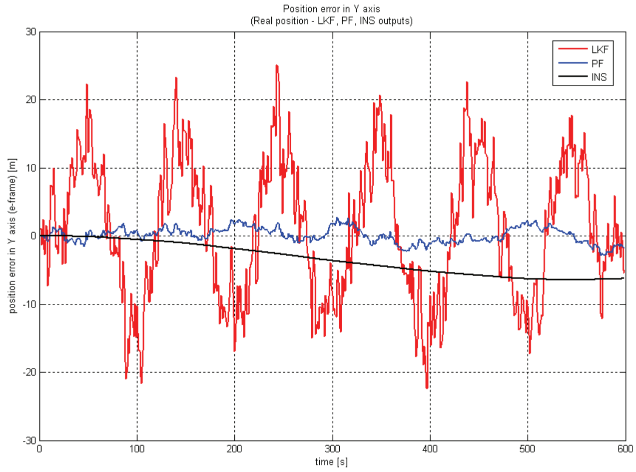

Fig. 7. Position error in Δy^e axis

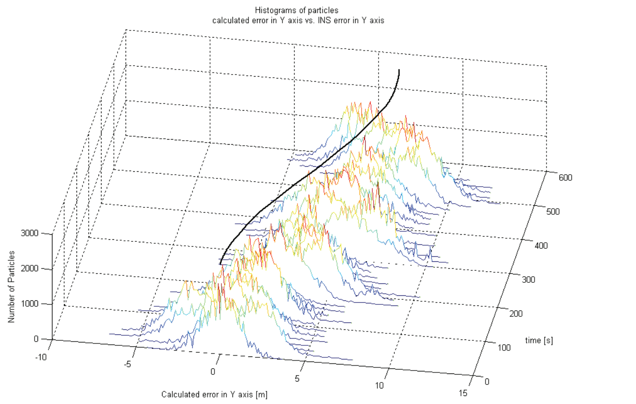

Fig. 8. Histograms of particles – calculated errors in δy^e , solid line is INS error in y^e

Table 2. Test results for high dynamics movement

Results for High dynamics movement	INS			GPS			LKF		
	x^e	y^e	z^e	x^e	y^e	z^e	x^e	y^e	z^e
RMS [m]	-	-	-	6.91	7.03	7.32	4.16	10.03	5.96
maximum error [m]	14.08	-6.43	-20.60	22.84	24.03	24.31	12.68	25.08	-24.93
Filters	PF 1000			PF 20000			PF 50000		
	x^e	y^e	z^e	x^e	y^e	z^e	x^e	y^e	z^e
RMS [m]	1.54	1.01	1.87	1.06	1.09	1.69	1.09	1.12	1.60
maximum error [m]	4.33	2.80	9.11	3.16	2.93	4.19	3.76	2.88	4.75

Conclusions

The paper describes new approach to navigation information processing using efficient particle filtering. The values of the variances when determining object trajectory by described integrated system using Monte Carlo method were approximately five times smaller than values of navigation information from individual sensors. The results of the simulation underline the correctness of the initial hypothesis and the choice of the particle filter for integrated navigation systems.

When the particle filter is used in practice, we often wish to minimize the number of particles to reduce the computational burden. Despite the theoretical independence of accuracy on the particle dimension, it is well-known that the number of particles needs to be quite high-dimensional systems [*1*]. To be able to use a small number of particles and to reduce risk of divergence, a procedure known as Rao-Blackwellization can be applied. The idea is to use the Kalman filter for the part of the state space model that is linear and the particle filter for the other part. These reductions of computation will be main focus of our future work.

There is also need to say that all tests were under ideal condition. It means, that ideal placement of the sensors (no lever-arm) have been used, the other errors such scale factor, non-orthogonality etc. in INS were neglected. Also the model of GPS receiver was simplified.

REFERENCES

[1] Gustafsson F. et al.: Particle Filters for Positioning, Navigation, and Tracking, IEEE Transactions on Signal Processing, 2002
[2] Bergman, N.: Recursive Bayesian Estimation: Navigation and Tracking Applications, Linköping, 1999
[3] Rekleitis I. M.: A Particle Filter Tutorial for Mobile Robot Localization, Montreal, 2002
[4] Grewal M. S. et al.: Global Positioning systems, Inertial Navigation and Integration, 2001
[5] Doucet, A.; de Freitas, N.; Gordon, N. J.: Sequential Monte Carlo Methods in Practice, Springer, New York, 2001
[6] Sotak, M.; Sopata, M.; Breda, R.; Rohac, J.; Vaci, L.: Navigation System Integration. Monograph: 1. ed., printed by Robert Breda, Kosice, the Slovak Republic, 2006, ISBN 80-969619-9-3
[7] Paces, P. - Reinstein, M. - Draxler, K.: Fusion of Smart Sensor Standards and Sensors with Self-Validating Abilities In: Journal of Aircraft. 2010, vol. 47, no. 3, p. 1041-1046. ISSN 0021-8669.
[8] Farrell, J. A.: Aided Navigation: GPS with High Rate Sensors. McGraw-Hill 2008, ISBN: 978-0-07-149329-8
[9] Breda. R.: Computation od navigational equation in the inertial coordinate system. In: Acta Avionica. - ISSN 1335-9479. - Vol. 11, No. 17 2009, p. 71-73
[10] Groves, Paul D.: Principles of GNSS, Inertial, and Multi-Sensor Integrated Navigation Systems. Artech House Publishers, 2007, ISBN: 978-1580532556
[11] Kuffova, M., Kralik, V.: Indoor navigation and mapping robot system. In: ICMT ´09: International conference on Military Technologies 2009, 5 to 6 May 2009, Brno: University of Defence, 2009. ISBN 978-80-7231-648-9. - S. 499-502
[12] Reinstein, M. - Rohac, J. - Sipos, M.: Algorithms for Heading Determination using Inertial Sensors In: Przeglad Elektrotechniczny. 2010, vol. 86, no. 9, p. 243-246. ISSN 0033-2097.

***Author**: Ing. Milos SOTAK PhD., Armed Forces Academy, Department of Electronics, Demanova 393, 031 06 Liptovsky Mikulas 6, Slovakia, E-mail: milos.sotak@gmail.com*