An Algorithm for Error Reducing in IMU

***Abstract*—During the last few years microminiaturized inertial sensors were introduced in many applications. Their small size, low power consumption, rugged construction open doors to many areas of implementation. The main drawback of these sensors is gyro drift, leading to an unavoidable accumulation of errors. In the paper an approach is proposed to diminish error accumulation.**

Keywords - Inertial sensors, change detection

I. INTRODUCTION

Inertial Measurement Unit (IMU) consists from one or more sensors, measuring the change of kinematic energy of a moving body. The sensors are divided in two groups: gyro sensors and accelerometers. Gyro sensor gives rotation rate of the body. Accelerometer provides information about linear acceleration of the body. Usually description of 3D motion of a body is given by 3 orthogonally placed accelerometers giving transition dynamic of the body and 3 orthogonally placed gyro sensors determining the orientation of the body. The axes of the both types of sensors normally coincide – e.g. in a 3D orthogonal coordinate system there are sensors to measure linear accelerations on each of the axes and rotation rate of the same axes. Thus the calculation process is also simplified. Two type of IMU were realized in the years. The first one is a classical gyroscope, which preserve one and the same (initial) position, remaining independent of body movement. The real orientation of the body is measured as a difference between gyroscopes axes orientation and present orientation of the body - its roll, pitch and yaw. The second one, called also strap-down gyro sensor, is fixed tightly on the body and provides measurement of rate of rotation of the body. Usually the strap-down sensors are produced as a MEM device with extremely high robustness and low power consumption. In this work such a type of devices will be considered. The body attitude is calculated using simultaneously the measurements of 6 sensors. Body orientation is given by integration of gyro sensors measurements. Transition of the body is calculated by double integration of accelerometers readings, according current body orientation. The integration process quickly accumulates errors. Due to existence of almost constant gravitational acceleration even small errors in the estimates of orientation of the body cause big deviation in the decomposition of gravitational acceleration on the axes, leading to large scale of attitude errors. Due to the quality of sensors IMU are divided in four groups of class of accuracy [1]:

TABLE I. ACCUMULATED ERROR DUE TO ACCELEROMETER BIAS ERROR

Grade	Accel. Bias Error [mg]	Horizontal Position Error [m]			
		1 s	**10 s**	**60 s**	**1 hr**
Navigation	0.025	0.00013	0.012	0.44	1600
Tactical	0.3	0.0015	0.15	5.3	19000
Industrial	3	0.015	1.5	53	190000
Automotive	125	0.62	60	2200	7900000

TABLE II. ACCUMULATED ERROR DUE TO ACCELEROMETER MISALIGNMENT

Accelerometer Misalignment [deg]	Horizontal Position Error [m]			
	1 s	**10 s**	**60 s**	**1 hr**
0.050	0.0043	0.43	15	57000
0.10	0.0086	0.86	31	110000
0.50	0.043	4.3	150	570000
10	0.086	8.6	310	1100000

TABLE III. ACCUMULATED ERROR DUE TO GYRO ANGLE RANDOM WALK

Grade	Gyro Angle Random Walk [deg/√hr]	Horizontal Position Error [m]			
		1 s	**10 s**	**60 s**	**1 hr**
Navigation	0.002	0.00001	0.0001	0.0013	620
Tactical	0.07	0.0001	0.0032	0.046	22000
Industrial	3	0.01	0.23	3.3	1500000
Automotive	5	0.02	0.45	6.6	3100000

As it can be seen from Table I, Table II and Table III, even small errors in gyro angle estimation may discredit navigation.

II. PROBLEM DESCRIPTION

The body motion in an inertial coordinate system can be described by following equation:

$$\vec{r}_i = \vec{r}_0 + \int_0^t \int_0^t \left(\vec{g} + \vec{a}_i(t)\right) dt dt \qquad (1)$$

where $\vec{r}_i$ denotes the vector of attitude coordinates in inertial coordinate system; $\vec{g}$ is gravitational acceleration, regarded constant for the time and space of the body movement; $\vec{a}_i$ is acceleration vector of forces, influencing the body, adjusted to inertial coordinate system; $\vec{r}_0$ is the initial body attitude in inertial coordinate system at time *t*=0.

The body space orientation can be described accordingly:

$$\vec{v} = \vec{v}_0 + \int_0^t \vec{\omega}_i dt \tag{2}$$

where $\vec{v}_0$ denotes the initial body orientation in inertial coordinate system at time *t*, $\vec{\omega}_i$ is the vector 3D rate turn, adjusted to inertial coordinate system.

The goal of navigation is to find coordinates of a body and its orientation. In the case of IMU sensor the task is solved based on IMU measurements and integral equation (1) and (2). A simple algorithm for coordinate determination is presented below. The calculation scheme is based on Euler angles. Let us denote the rotation matrix, transforming a vector from the moving body to inertial coordinate system by $C_b^i(t)$. Then an acceleration vector $a_b(t)$ in the body coordinate system will be transformed to inertial coordinate system by (3):

$$a_i(t) = C_b^i(t) a_b(t) \tag{3}$$

Now the rotation matrix $C_b^i(t)$ will be represented by Euler angles [2]:

$$C_b^i(t) = C_z^{'}(t) C_y^{'}(t) C_x^{'}(t), \tag{4}$$

where

$$C_x(t) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos(\varphi(t)) & \sin(\varphi(t)) \\ 0 & -\sin(\varphi(t)) & \cos(\varphi(t)) \end{pmatrix},$$
$$C_y(t) = \begin{pmatrix} \cos(\theta(t)) & 0 & -\sin(\theta(t)) \\ 0 & 1 & 0 \\ \sin(\theta(t)) & 0 & \cos(\theta(t)) \end{pmatrix},$$
$$C_z(t) = \begin{pmatrix} \cos(\psi(t)) & \sin(\psi(t)) & 0 \\ -\sin(\psi(t)) & \cos(\psi(t)) & 0 \\ 0 & 0 & 1 \end{pmatrix},$$

are the rotation matrixes that rotate vectors on angles $\varphi(t)$, $\theta(t)$, $\psi(t)$ on axes *x*, *y* and *z*. It is important to mention that the order of rotation is important. If the angles of rotation are sufficiently small:

$$\delta t \to \begin{cases} \delta\varphi \to 0 \\ \delta\theta \to 0 \\ \delta\psi \to 0 \end{cases}$$

or in other words the measurement sampling rate is sufficiently high – satisfies Nyquist sampling rate, which guarantees that you capture a signal properly because you sample it at least twice per cycle of the highest frequency component it contains, the following substitutions for an angle α may be applied: $\cos\delta\alpha \approx 1$ and $\sin\delta\alpha \approx \delta\alpha$. The product of small angles can be also approximated by zero: $\delta\alpha * \delta\alpha \approx 0$. The final expression for the change in rotation matrix will be:

$$C_b^i(\delta t)\Big|_{\delta t \to 0} = \begin{pmatrix} 1 & -\delta\psi & \delta\theta \\ \delta\psi & 1 & -\delta\varphi \\ -\delta\theta & \delta\varphi & 1 \end{pmatrix} =$$
$$= \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} + \begin{pmatrix} 0 & -\delta\psi & \delta\theta \\ \delta\psi & 0 & -\delta\varphi \\ -\delta\theta & \delta\varphi & 0 \end{pmatrix} = I + \Delta_\upsilon \tag{5}$$

The final rotation matrix can be presented as a product of the rotation matrix at t and calculated above rotation matrix $C_b^i(\delta t)$, corresponding to small additional rotations, committed in time interval δt :

$$C_b^i(t + \delta t) = C_b^i(t) C_b^i(\delta t) = C_b^i(t)(I + \Delta_\upsilon) \tag{6}$$

Let now express the derivative of rotation matrix:

$$\dot{C}_b^i(t) = \lim_{\delta t \to 0} \frac{C_b^i(t+\delta t) - C_b^i(t)}{\delta t} = \lim_{\delta t \to 0} \frac{C_b^i(t)(I + \Delta_\upsilon) - C_b^i(t)}{\delta t} =$$
$$= C_b^i(t) \lim_{\delta t \to 0} \frac{\Delta_\upsilon}{\delta t} = C_b^i(t) \dot{\Delta}_\upsilon , \tag{7}$$

where $\dot{\Delta}_\upsilon = \begin{pmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{pmatrix}$ and $\omega_x, \omega_y, \omega_z$ are the lastly received measurements of rotation rates from gyro sensors on corresponding axis.

The solution of (7) is $C_b^i(t) = C_b^i(0) \ell^{\dot{\Delta}_\upsilon}$.

The matrix exponent in solution can be presented as an infinite sum:

$$\ell^{\dot{\Delta}_\upsilon} = \sum_{k=0} \frac{\dot{\Delta}_\upsilon}{k!} = I + \frac{\dot{\Delta}_\upsilon}{1!} + \frac{\dot{\Delta}_\upsilon^2}{2!} + \cdots + \frac{\dot{\Delta}_\upsilon^k}{k!} + \cdots$$

Taking into account only the first two terms (linear approximation) we receive an approximate formula for recurrent calculation of rotation matrix:

$$C_b^i(t + \delta t) = C_b^i(t)(I + \Delta_\upsilon) \tag{8}$$

Let now calculate the exact expressions for angle derivatives. The differential equation (7) will be used, where the rotation matrix from (4) in explicit form will be substituted:

$$C_b^i =$$

$$= \begin{pmatrix} \cos\theta\cos\psi & -\cos\varphi\sin\psi + \sin\varphi\sin\theta\cos\psi & \sin\varphi\sin\psi + \cos\varphi\sin\theta\cos\psi \\ \cos\theta\sin\psi & \cos\varphi\cos\psi + \sin\varphi\sin\theta\sin\psi & -\sin\varphi\cos\psi + \cos\varphi\sin\theta\sin\psi \\ -\sin\theta & \sin\varphi\cos\theta & \cos\varphi\cos\theta \end{pmatrix} \tag{9}$$

The matrix equation will be resolved for matrix element (3,1) (3-rd row, 1-st column). The corresponding equation looks like:

$$\frac{d(-\sin\theta)}{dt} = \begin{pmatrix} -\sin\theta & \sin\varphi\cos\theta & \cos\varphi\cos\theta \end{pmatrix} \begin{pmatrix} 0 \\ \omega_z \\ -\omega_y \end{pmatrix} \tag{10}$$

$$-\cos\theta * \dot{\theta} = \sin\varphi\cos\theta\ \omega_z - \cos\varphi\cos\theta\ \omega_y$$

Therefore:

$$\dot{\theta} = \cos\varphi\ \omega_y - \sin\varphi\ \omega_z \tag{11}$$

For matrix element (3,2) we receive:

$$\cos\varphi\cos\theta\ \dot{\varphi} - \sin\varphi\sin\theta\ \dot{\theta} = \sin\theta\ \omega_z + \cos\varphi\cos\theta\ \omega_x \tag{12}$$

Substituting $\dot{\theta}$ from (11) into (12) and expressing $\dot{\varphi}$ we receive:

$$\dot{\varphi} = \omega_x + tg\theta\left(\sin\varphi\ \omega_y + \cos\varphi\ \omega_z\right) \tag{13}$$

To find the expression for $\dot{\psi}$ the equations have to be used for matrix elements that contain ψ. For example, if the element (1,1) is used:

$$-\sin\theta\cos\psi\ \dot{\theta} - \cos\theta\sin\psi\ \dot{\psi} =$$

$$= \left(\cos\varphi\sin\psi + \sin\varphi\sin\theta\cos\psi\right)\omega_z +$$

$$+ \left(\sin\varphi\sin\psi + \cos\varphi\sin\theta\cos\psi\right)\left(-\omega_y\right) \tag{14}$$

Using (11) to substitute $\dot{\theta}$ and simplifying we receive:

$$\dot{\psi} = \frac{1}{\cos\theta}\left(\sin\varphi\ \omega_y + \cos\varphi\ \omega_z\right) \tag{15}$$

The equation (11), (13) and (15) are most often used for calculation of rotation angles between two successive gyro measurements with a linear approximation only.

Let now consider errors in sensor measurements.

The error propagation for acceleration sensors only looks like:

$$\vec{r}_i = \vec{r}_0 + \frac{\vec{g}t^2}{2} + \int_0^t\int_0^t \left(\vec{a}_i(t) + \vec{\varepsilon}_a\right)dtdt =$$

$$= \vec{r}_0 + \frac{\vec{g}t^2}{2} + \frac{\vec{\varepsilon}_a t^2}{2} + \int_0^t\int_0^t \vec{a}_i(t)dtdt \tag{16}$$

Here $\vec{\varepsilon}_a$ denotes the error vector of acceleration sensors.

The error propagation for gyro sensors only looks like:

$$\vec{v} = \vec{v}_0 + \int_0^t \left(\vec{\omega}_i(t) + \vec{\varepsilon}_\omega\right)dt = \vec{v}_0 + \vec{\varepsilon}_\omega t + \int_0^t \vec{\omega}_i(t)dt \tag{17}$$

Here $\vec{\varepsilon}_\omega$ denotes the error vector of gyro sensors.

The equations (16) and (17) give error propagation in the simplest case of independent errors. In practice there are many types of errors, influencing one to others. The influence of rotation rate error measurements on angle determination is obvious from (11), (13) and (15). As a consequence the error propagation in (17), for example, generates/induces nonlinear errors in estimation of accelerations, leading to quickly growing errors in estimated system position. That is why (16) and (17) are used only to approximate the order of generated errors and are not of practical use.

The sensors are subject to different types of errors due to sensor imperfectness, model inaccuracy or computational errors.

The main errors influencing on the attitude estimation accuracy may be grouped into three categories [3, 4]:

A. *Sensors do not provide perfect and complete data.*

- Bias errors produce constant or almost constant shift of sensor values from the true ones.
- The scale factor errors cause lack of correspondence between real turn velocities and real straight linear accelerations and output sensors readings (gyro and accelerometer correspondingly).
- Errors due to manufacturing imperfections in IMU. Usually they are caused by non-orthogonally placed accelerometer or gyro sensors on the chip or by lack of coincidence between axes of corresponding accelerometer and gyro sensors. The last error more often is initiated by the first one, but sometimes can exist alone.

- The sensors readings are also contaminated by additive Gaussian noise.
- Temperature dependent errors. Temperature deviation affects output readings.
- There is time synchronization problem. Sensors readings do not belong to one and the same moment of time.
- Dynamic error (lag of sensor reaction/response to force implementation).

B. Imperfectness of the used models and computational arithmetic

- The model inaccuracy usually is caused by inexact sensor approximation, incorrect gravitational acceleration estimate.
- The computational errors are caused by limitations of computer arithmetic, iterative procedures for optimization, calculations of trigonometric functions, loss of orthonormality of matrices, etc.

C. External sources of disturbances (uncontrolled, unpredictable even unknown sources of different type disturbances)

- Platform vibration. The vibration counteracts to sensor accuracy. It depends of different random factors, platform dynamics, mass distribution, switching on/off of different devices, and etc.
- Others

The Fig. 1 below displays the influence of different types of errors on quality of attitude estimation.

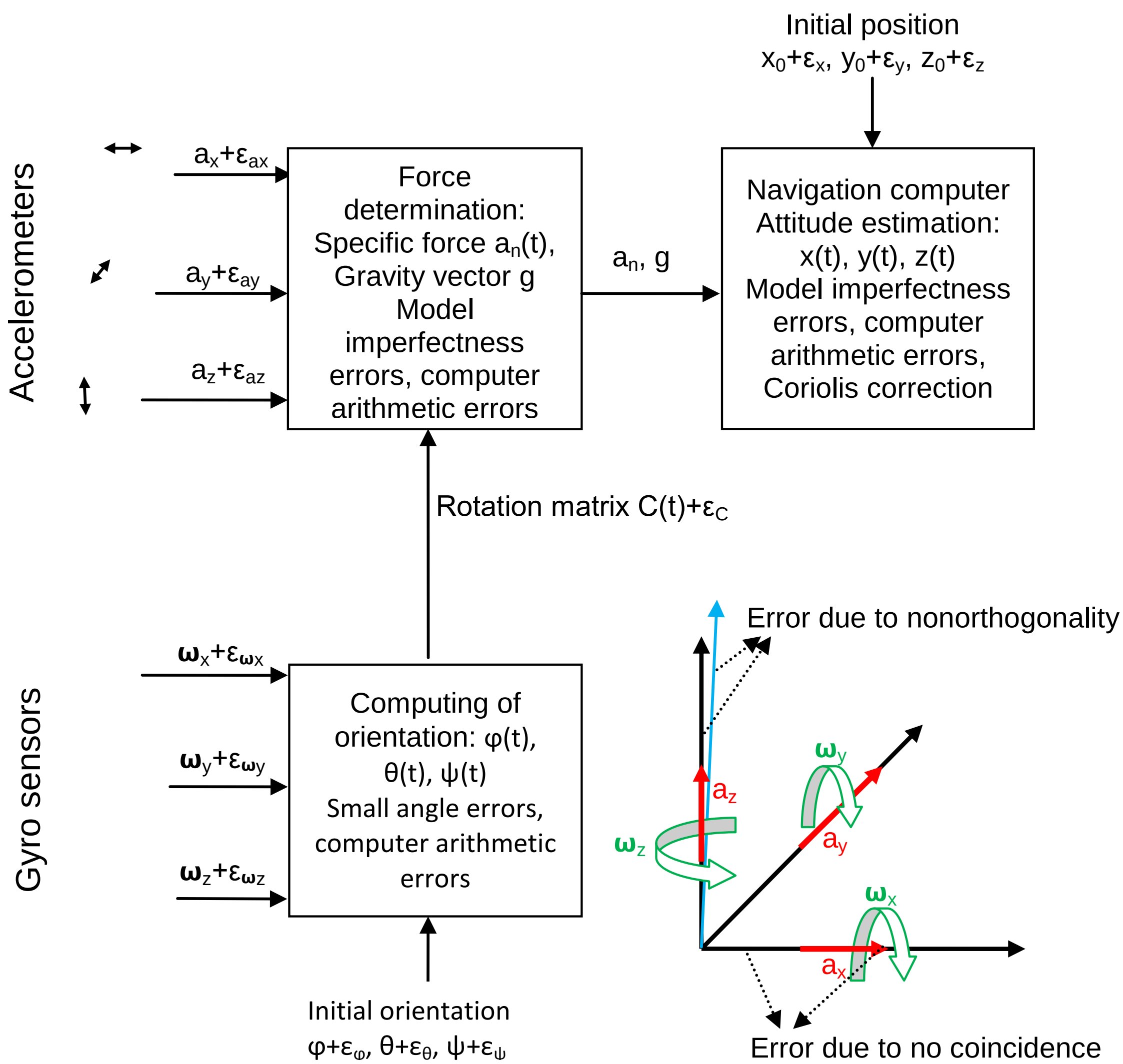

Figure 1. Errors in an IMU

In order to minimize different type of errors we have to estimate their influence on the position estimate.

There are many well established methods for self-consistency check and normalization. One of them concerns the rows/columns of the rotation matrix. The rotation matrix is direction cosine matrix, which row/columns are projections of unity vector onto orthogonal axes. That means, that the sum of squares of values in each row/column have to be equal to 1 and due to their orthogonality, their scalar products have to be zero. In the cases of using quaternions the normalization means that the sum of squares of quarternion elements has to be equal to 1. This normalization usually doesn't correct errors. Even if optimization procedure is started, the best received result does not guarantee the error compensation. Moreover, it usually propagates the error over correct terms. That is why the precise error expression is not of practical use.

III. THE PROPOSED ALGORITHM FOR ERROR PROPAGATION MINIMIZATION

The IMU sensor systems have unavoidable sources of errors. Through normalization and orthogonality checks the error propagation can be only slightly enhanced, if ever. The effect is most often dilution of the errors on all variables. To stop the process of error accumulation we have to stop the process of integration/double integration and reinitialize calculations. We cannot do permanently that because of need to estimate platform position. To minimize integration time we will discover motion of the platform and only when the motion is detected the integration process will be switched on. When there is no motion detected, the integration process will be stopped and the platform orientation and position will remain the same. This idea is not new one. For example, in the embedded software on newest MEMS an activity threshold is inserted for acceleration sensors. Only accelerations, exceeding threshold, switch on the flag "Activity". In spite of its simplicity the realization of this functionality gives the system engineers knowledge when to initialize the sensor or when to start recalibration procedure. In this work we develop this idea further. We realize more precise algorithm for activity detection, which is less susceptible from the sensor signal bias.

Two different types of change can be distinguished. "Abrupt" change is an instantaneous change in the parameters of the system. Here instantaneous change means that the transition from one to other state is committed faster with respect to the sampling period of the measurements. The second type of change is "slow" change. We are interested in detection of both types of change, independent on magnitude value of the change. Moreover, IMU measurements characterize usually with small and not necessarily fast – changes.

Change detection may be statistically formulated as a random process, which statistical parameters change significantly at a point, called change point. To estimate statistical parameters, an interval with sufficient length has to be considered. The change point divides the time-series data of two parts with different distribution characteristics. The change detection problem may be resolved by model based change detection algorithms or by model-free algorithms. Typically model-based approaches decompose time-series data into trend, periodical data and residual components. For navigation purposes the model-based approaches are not suitable due to their complexity and lack of general models. The model-free algorithms deal with data values directly. One of the most popular representatives of this type of algorithms is CUSUM [5]. Page, the author of CUSUM, examined a "quality number", by which he denotes a parameter of the explored probability distribution; for example, the mean. The devised recursive procedure calculates a cumulative sum and compares it with a threshold. To detect the exact point of change, Page defined also the average run length as the expected number of processed measurements before action is taken. Usually, model-free change detection algorithms are computationally simple, more robust in comparison with model-based ones.

The applied in this paper change detection algorithm is based on Shewhart control chart [6]. Due to many types of error sources, influencing on sensor data, it is assumed that the time-sequences from inertial sensors can be represented as a signal disturbed by additive Gaussian distributed noise. It can be considered that any change in dynamic of the examined platform leads to a change in the output data of one or more strapdown inertial sensors. We are looking for a change of the mean of a sample with length equal to N by the following sufficient statistic [6]:

$$S_{j+1}^{j+N} = \frac{\mu_{k+1} - \mu_k}{\sigma^2} \sum_{i=j+1}^{j+N} \left(y_i - \mu_k - \frac{\mu_{k+1} - \mu_k}{2} \right), \text{ where}$$

$$\mu_k = \frac{1}{N} \sum_{i=NK+1}^{N(K+1)} y_i$$

The change is detected when the inequality is fulfilled:

$$\left| \mu_{k+1} - \mu_k \right| \geq m \frac{\sigma}{\sqrt{N}}$$

The tuning parameters of the procedure are m and N. The procedure is applied on the time-series from the all six inertial sensors – three gyros and three accelerometers. The output results are fused in a final time-series with two states only – "0" and "1". "Zero" means to stop integration of sensor output data and remains in the same state (the same position, velocity and acceleration) and preserve the same orientation of the platform in the space. When "One" appears for the first time the integration process is restarted and a new state vector and a new orientation of the platform are calculated.

IV. EMPIRICAL RESULTS

The algorithm was tested on a platform with MPU-6050 strapdown inertial sensors. The platform commits a simple move following contours of a quadrate with a side, equal to 10 cm. The data flow from 3 gyros and 3 accelerometers were saved and two types of algorithm were applied. The calculated platform trajectory received by standard navigation algorithm (without activity detection) is shown on Fig. 2.

Figure 2. The reconstracted platform trajectory without change detection

The raw gyro and accelerometer signals are presented on Fig. 3 and Fig.4.

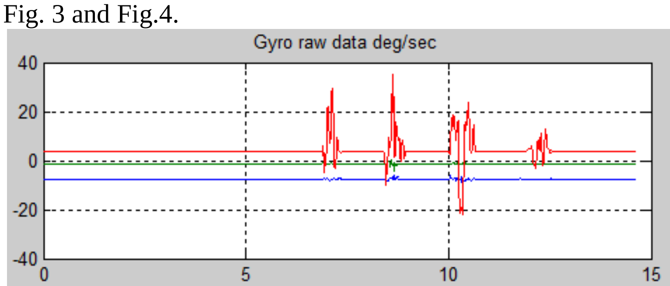

Figure 3. Gyro raw signals

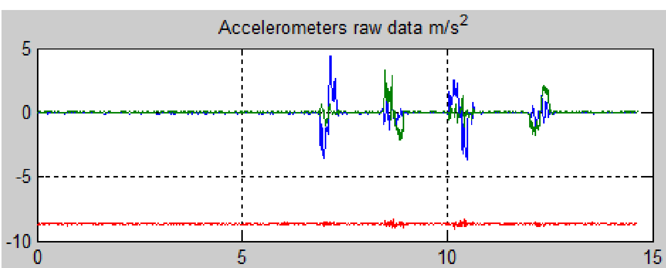

Figure 4. Accelerometer raw signals

On the Fig. 5 the change detection algorithm is illustrated. The first graphic depicts one of accelerometer signals, where change detection algorithm is applied on (second graphic). The Fig. 6 shows the fused (from all six sensors) result from change detection algorithms with reconstructed trajectory (Fig.7).

V. Conclusion

The contemporary strapdown inertial MEMs are far behind in accuracy from the precise, very heavy and costly navigation platforms. In spite of this a lot of applications are waiting for more precise inertial sensors. The implementation of change detection algorithms enhances the accuracy of inertial MEMs and open door for realization of some of the ideas. The main drawback of the proposed change detection algorithm, which has to be considered, is the very small changes. They are neglected in the process. Or, nothing better than one accurate sensor!

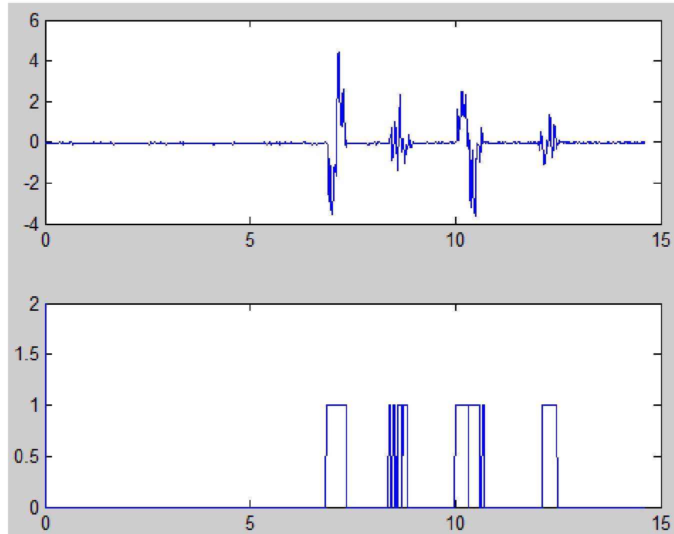

Figure 5. Application of change detection algorithm on a raw signal

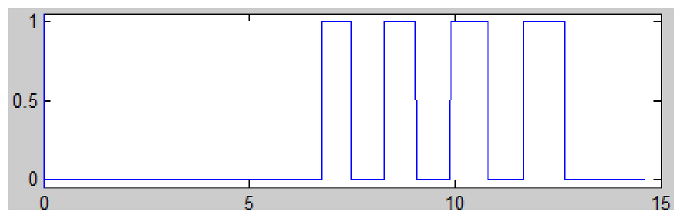

Figure 6. The fused result from change detection algorithm

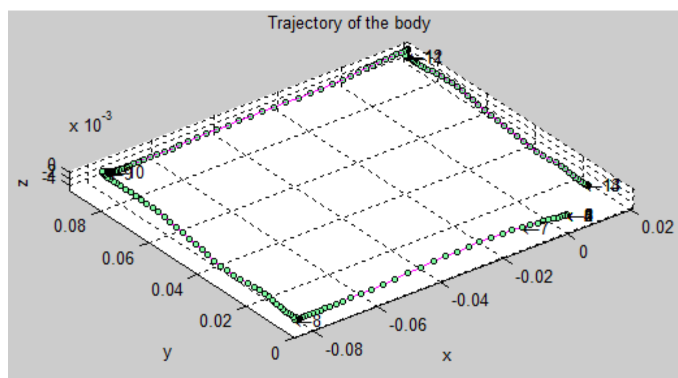

Figure 7. The reconstracted platform trajectory with change detection

Acknoledgement

The research work reported in the paper is partly supported by the project AComIn "Advanced Computing for Innovation", grant 316087, funded by the FP7 Capacity Programme (Research Potential of Convergence Regions) and by the project No DFNI – I01/8 funded by the Bulgarian Science Fund. All data, laboratory equipment were supplied by “MM Solutions” in the framework of the project “Industrial research for development of technology for image enhancement and video stabilization using inertial sensors”, Contract BG161PO003-1.1.06-0037-C0001, Operational

Program "Development of the Competitiveness of the Bulgarian Economy".

REFERENCES

[1] http://www.vectornav.com/index.php?&id=76

[2] David H. Titterton, John L. Weston Navigation Technology - 2nd Edition, The Institution of Electrical Engineers, 2004, ISBN 0 86341 358 7

[3] Grewal, M.S., Weill L.R., Andrews A.P., Global Positioning Systems, Inertial Navigation, and Integration, John Wiley & Sons, 2001, ISBN 0-471-20071-9.

[4] Oliver J. Woodman, An introduction to inertial navigation, Technical Report UCAM-CL-TR-696, ISSN 1476-2986, 2007.

[5] Page, E. S., Continuous Inspection Scheme, Biometrika 41 (1/2), pp. 100–115, http://www.jstor.org/discover/10.2307/2333009?uid=3737608&uid=2&uid=4&sid=21101894206801 JSTOR 233300

[6] Michele Basseville, Igor V. Nikiforov, Detection of Abrupt Changes: Theory and Application, Prentice-Hall, Inc, ISBN 0-13-126780-9, 1993.