Development of a New Body Motion Parameters Acquisition System

Pāvels Maksimkins
Institute of industrial Electronics and Electrical engineering
Riga Technical University
Riga, Latvia
pavels.maksimkins@rtu.lv

Armands Šenfelds
Institute of industrial Electronics and Electrical engineering
Riga Technical University
Riga, Latvia
armands.senfelds@rtu.lv

Jūlija Maksimkina
Institute of industrial Electronics and Electrical engineering
Riga Technical University
Riga, Latvia
julija.maksimkina@rtu.lv

Leonīds Ribickis
Institute of industrial Electronics and Electrical engineering
Riga Technical University
Riga, Latvia
leonids.ribickis@rtu.lv

***Abstract*—This paper address problem related to data acquisition of real body motion parameters with respect to following data processing and implementation on robotic motion simulator systems. Transition from real world use case to physical motion simulation systems involve processing of acquired motion data by measurements and application of motion cueing algorithms for particular robotic motion simulator. This paper present requirement analysis, prototype development and testing towards user friendly and IoT based mobile object motion parameter extraction system for motion simulator or digital twin applications.**

***Keywords*—manipulators, motion simulator, classic washout filter, Internet of Things, mobile applications, accelerometers, motion measurements, electromechanical sensors, Node-RED, Digital twin**

I. Introduction

As time goes by necessity for robot-based motion simulators increases just as their capabilities expand. That leads to many use-cases that demand easy and user-friendly solutions. Since such robots as drones could be used as human transportation vehicles it is worthwhile to make available motion simulators to incorporate drone simulation applications where human interaction is possible or either static simulation where human only observes.

This paper focuses on developing a system, which senses main rigid body kinematic parameters such as angular velocities, accelerations, position in terms of global coordinates and velocity, displays data in real time and securely saves on remote server.

Nowadays robot-based motion simulators (shown in Fig.1) are being used either for static simulations [1] (i.e. simulation process cannot be affected) or interactive simulations [2] (i.e. computer game-like, human supposed to control simulation process and interact).

The system, which is described in this paper, might be used to conveniently reconstruct environment in Virtual Reality, such as car racing track could be reproduced just by making few laps on it or create static (non-interactive) simulation of for example rollercoaster (combined with data gathered with video camera). Nevertheless, this paper does not describe algorithms of using gathered data for VR replication.

Since this system is aimed at motion simulators, it is not considered a main goal to get only acceleration data, this system development is a step towards high-quality simulation creation, where simulation-sickness effect is minimized.

Fig. 1. Robot-based motion simulator.

Simulation-sickness effect arises because in motion human tends to keep his head parallel to horizon, while gathering acceleration data might be interfered by dynamic object orientation change, which then might change gathered data slightly so vector of gravity interfere measurements [3]. One of the possible use of gathered data is to use as an input for Classical Washout filter (motion cueing algorithm) used successfully in motion simulators [4].

One of the benefits of a system described here is flexibility, it could be adjusted for many already made and implemented solutions. Node-RED software being used here, which has wide community and active forums, good documentation is considered as a benefit as it gives many integration options and support [5]. As a server Raspberry Pi 3 Model B is used, its performance is enough to serve all the needs including handling of graphical interface and communication.

Being in a stage of proof-of-concept main goal of this work is to see how orientation of rigid body is affecting acceleration measurements and experience a worst-case results to see do they really need to be compensated or corrected.

One of the challenges is to use gathered data to get linear accelerations with no gravity vector acting on measurements.

This paper is structured in nine sections where Section II discusses requirements taken into account when developing new system, Section III shows existing solutions available on

978-1-6654-2538-4/21/$31.00 ©2021 IEEE

market and their features, Section IV describes architecture of a whole system. Section V addresses prototype and software development process and results. Section VI discuss data acquiring and interpretation process. Paper concludes with Section VII where results are discussed.

II. System Requirements

There would be no challenge and need to create new system which only senses accelerations and writes them down to a file. The main goal is to create a system, which outputs data that can directly be used with large range motion simulator. Performance is also a concern because large range motion simulator expects to see new data every 12 milliseconds.

Requirements for new system were based on the need for a Plug and Play solution which needs no additional inputs from user.

Comma-Separated values structure was chosen as it is one of the most convenient structure to reformat or use in additional user-preferred software [6].

System should have option to send data to server in real time and be able to do this securely. It should also have option to display data in real time to be able to decide if it is being collected correctly.

Data export must be easy and system must have options to conveniently export data to user-preferred software (i.e. Matlab etc. [7]). Industrial communication protocols are considered as a benefit.

Measurement unit must be in convenient form factor and should have possibility to be powered from easily accessible power source.

System should have option to use many measurement units and use them together for synchronous data acquiring.

Measurement unit must incorporate acceleration sensor, gyroscope and GPS module.

III. Existing Solutions

Speaking about existing solutions there are not many. As real-time data visualization is cruel it already wipes out almost every accessible solution. Even though Table I shows existing solutions, where:

I – system being described in this paper.
II – Gdata X16-4.
III – AccelFox.
IV – Aaronia GPS logger.

TABLE I. Existing solutions comparison

	I	II	III	IV
PlugNPlay	+	+	+	+
.csv structure	+	+	+	-
Network capabilities	+	-	-	-
Secure networking	+	-	-	-
UDP support	+	-	-	-
Real-Time data visualization	+	-	-	-
WiFi/4G connection	+	-	-	-
Accelerometer	+	+	+	+
Gyroscope	+	-	+	+
GPS	+	-	+	+
Customizable	+	-	-	-
Device synchronization	+	-	-	-

If only there were no requirement for real-time capabilities "AccelFox" solution would be close enough to say that it could be used for certain applications. But since there are no candidate that fulfills requirements it was decided to create new system. "AccelFox" system is powered AAA batteries while system being described in this paper used MiniUSB with DC 3.3V-6V allowed power source, which allows power source to be replaced on the go.

IV. System Architecture

As far as hardware goes Raspberry Pi 3 Model B was chosen to be a local server for data storage, visualization and for local communication purposes. It is relatively cheap computer with all needed interfaces and enough performance to handle all the tasks.

Measurement unit is based on ElectricIMP imp003 microcontroller. LSM9DS0 was chosen as inertia measurement unit and uBlox NEO-6M as a GPS module. Whole system architecture is shown in Fig. 2.

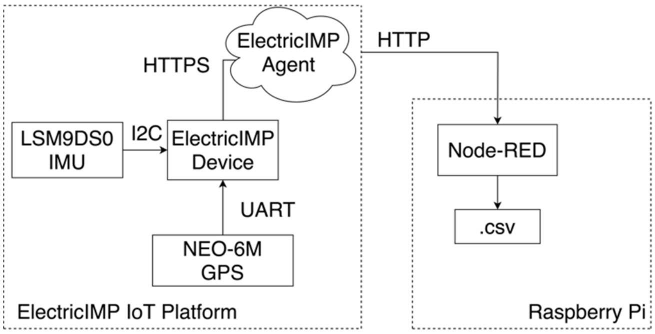

Fig. 2. System architecture.

It is worth to mention few things about ElectricIMP IoT platform.

Platform consists of device (microcontroller with impOS) and agent (cloud companion of device). ElectricIMP platform is the only IoT platform which is certified for cybersecurity (UL 2900-2-2) [8]. That goes from fact that communication between device (microcontroller) and agent is maintained through secure managed tunnel and cloud agent being protected by ElectricIMP (either it can be hosted on ElectricIMP inc. servers or on private servers). Microcontroller is being programmed through web interface and it needs internet connection to update code.

As a software Node-RED is used to manage connection, data formatting, data visualization and interaction with measurement unit.

V. Device Prototype And Software Development

Key features while developing device prototype to consider was that ElectricIMP platform (both device and agent) use Squirrel as a programming language which is not widely used nowadays for a microcontroller programming purposes.

LSM9DS0 uses i2c protocol to communicate with device and NEO-6M uses UART at speed 9600 baud which can be configured for higher speeds.

GPS module is set to read GPS data at 1Hz rate. However accelerometer and gyro reading are going as fast as possible. Since gyro reads degrees per second it is important to consider time difference between readings.

Fig. 3. represent assembled device prototype ready to take measurements.

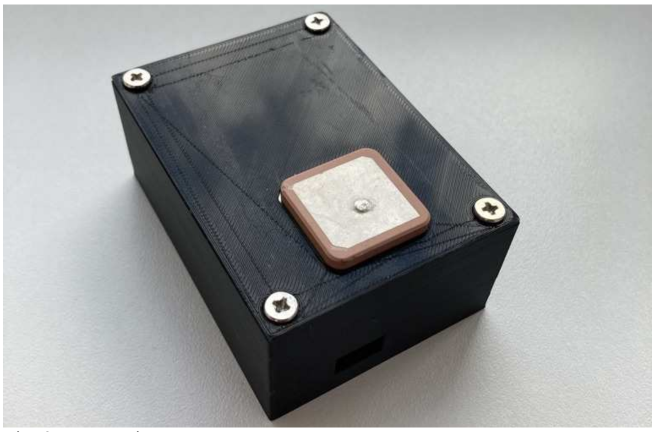

Fig. 3. Device prototype.

Since ElectricIMP platform incorporates both device and agent and both codes run asynchronously, their algorithms is shown together on Fig. 4. It was needed to create additional buffer on Agent side because quantity of HTTPS requests per second is limited by network capabilities (HTTP error code 429 “Too Many Requests” [9]).

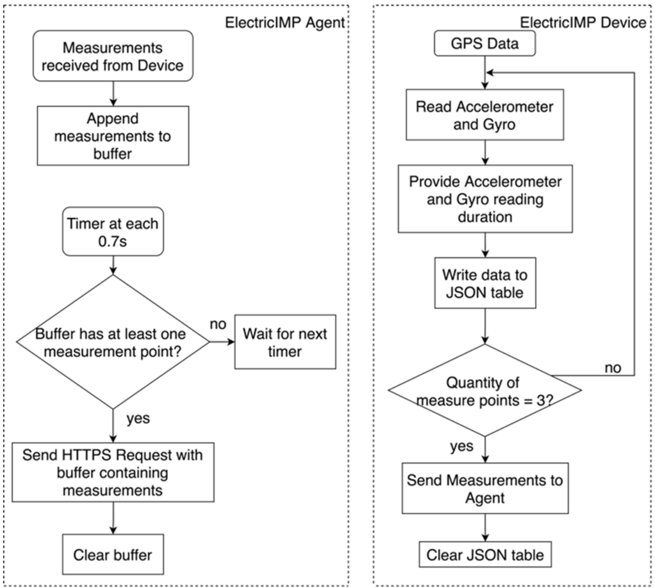

Fig. 4. Agent and Device algorithms.

Node-RED has been chosen to be a server for the system. It is low-code programming tool used for event-driven applications [10]. In this system it is used mainly to process HTTPS request, format data, save to Comma-separated values file and visualize data in real time. Fig. 4. demonstrate graphical user interface. As seen from ElectricIMP Agent code (Fig.4.) visualization updates each 0.7 seconds which is quite enough if real time interpretation is needed.

Fig. 6. illustrate simplified algorithm of Node-RED flow. It is also possible to send only accelerometer and gyro data with no GPS (if GPS satellites are not accessible).

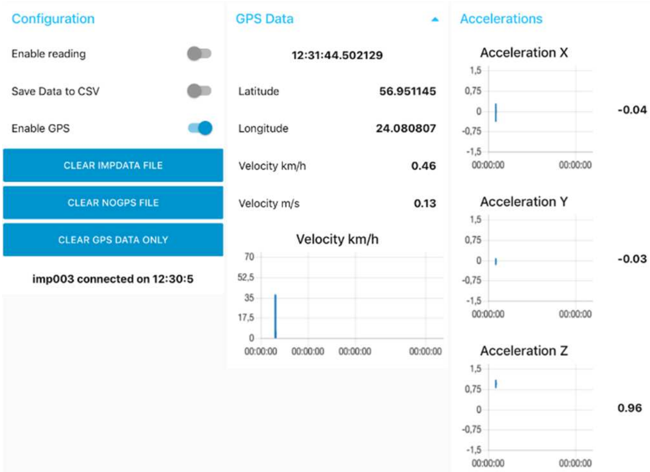

Fig. 5. Graphical user interface.

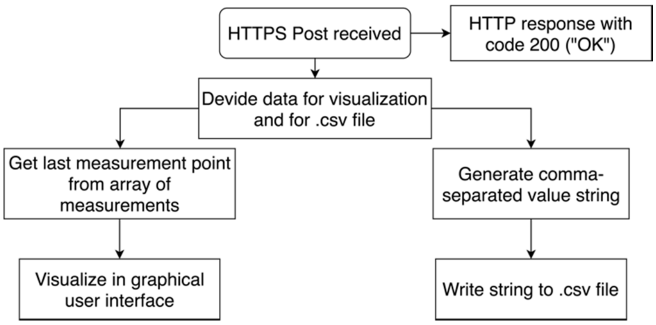

Fig. 6. Simplified Node-RED flow algorithm.

VI. Data Acquiring and Interpretation

It was chosen to make an experimental test using regular passenger vehicle Renault Clio II. Device was fixed close to the steering axis precisely centered. Fig. 7. Shows prototype being fixed inside the car. This position allows for good GPS signal reception and easy access to device if position adjustment is needed.

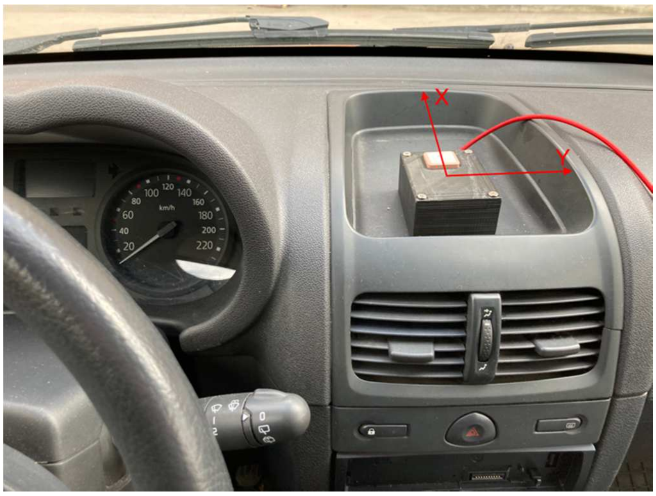

Fig. 7. Device prepared for experiment.

It was crucial to get car to flat surface so that inertia measurement sensor could be put parallel to the ground. That procedure were carried out carefully so accelerations in X and Y axis were zeroes and Z-axis acceleration were close to 1. It is enough to use only accelerometer reading to get the orientation right, because the data then will be used for motion simulator, and human vestibular system will not be able sense if there will be any imprecise measurements due to minor imperfections in device positioning. Because system has real-time data visualization it was easy to find right orientation of inertia measurement sensor.

To make data interpretation process easier and more precise such methodology was chosen – make one turn at high speed so accelerations and angle rates would be more prominent. Fig. 8. Shows chosen curve and GPS trajectory of chosen curve recorded by device.

Fig. 8. Chosen curve with GPS trajectory.

It is worth to mention that this experiment is based not only on the fact that car suspension consists of springs that compresses while cornering (mass shift), but also many elastic elements such as tires or rubber bushings, which leads to a car being rolled [11]. Experiment gives a opportunity to test if Y-axis acceleration (perpendicular to car driving direction) will reach theoretical values that will be calculated later, and define how car roll affects acceleration measurements. It is very important because, as said in introduction, human tends to keep his head parallel to a horizon (parallel to a ground) and accelerations acting on human's vestibular system are required, not the accelerations acting on car itself.

Before starting interpretation of gathered data it is needed to theoretically calculate values of Y-axis acceleration for chosen curve.

For demonstration purposes it has been chosen to review data for an entrance state of the curve (where acceleration in Y-axis reaches its maximal value), data is shown in Fig. 10. where Y-acceleration is shown as a blue line, angular velocity around X-axis is shown in orange, theoretical Y-axis acceleration is shown as dotted line and Z-axis acceleration is shown as a black line. Note that angular velocity is shown as degree per program tick, not degree per second.

Radius of curve was measured as 50.45m and using basic equation Y-axis acceleration was calculated as:

$$\vec{a}_{theor.} = \frac{v^2}{R} = \frac{15.22^2}{50.45} = 4.59\frac{m}{s^2}. \quad (1)$$

As seen from graph, acceleration in Y-axis (blue line) gets as high as 0.67g. Converting this value gives:

$$\vec{a}_{exp.} = 9.81 * 0.67 = 6.57\frac{m}{s^2}. \quad (2)$$

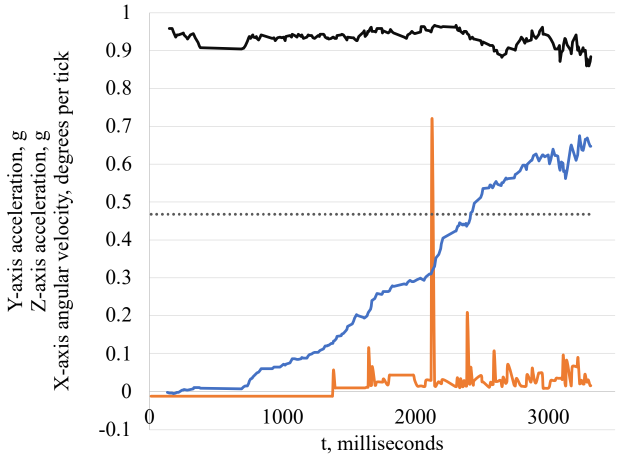

Fig. 10. Experimental data.

Difference between theoretical value seems evident at:

$$\vec{a}_{diff.} = \vec{a}_{exp.} - \vec{a}_{theor.} = 6.57 - 4.59 = 1.98\frac{m}{s^2}. \quad (3)$$

The key thing here is that orange line shows angular rate about X-axis. As it seems while cornering car rotates around X-axis (rolls), which affects Y-axis acceleration measurements.

Being at 0.96*g* (9.42 m/s^2) at steady state Z-axis acceleration drops to 0.86*g* (8.43 m/s^2) when Y-axis acceleration rises (while cornering). Which generates difference of:

$$\vec{a}_{g.diff.} = 9.42 - 8.43 = 0.99\frac{m}{s^2}. \quad (4)$$

Integrating angle velocity for corner entrance gives a graph displayed in Fig. 11. Which stands for roll angle (angle around X-axis).

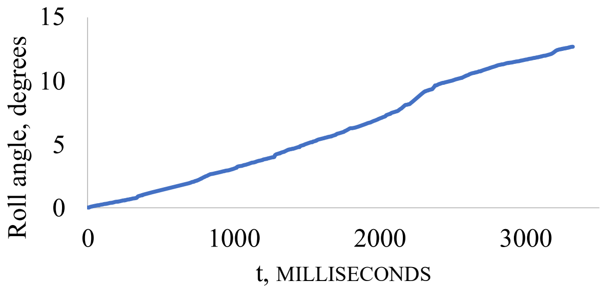

Fig. 11. Roll angle for corner entrance.

At the top of the corner car is rolled at angle of 12.66 degrees which means that gravity vector is directly acting on Y-axis acceleration generating additional acceleration of:

$$\vec{a}_{roll\ interference} = 9.81 * \sin(12.66°) = 2.15\frac{m}{s^2}. \quad (5)$$

Fig. 12. show car being in steady state (on the left) and car being rolled in curve (on the right), "g" represents vector of

gravity, φ represents roll angle and Y_g is the gravity interference.

To verify results Y-axis acceleration is calculated with a roll angle taken into account (and no gravity vector interference):

$$\vec{a}_{\text{theor. with roll}} = \frac{\vec{a}_{\text{exp.}} * 9.81 * \sin(\varphi)}{\cos(\varphi)} = \frac{6.57 - 9.81 * \sin(12.66°)}{\cos(12.66°)} = 4.53\frac{\text{m}}{\text{s}^2}. \quad (6)$$

Which shows that with roll angle taken into account system shows nearly the same acceleration on Y-axis as it was calculated in equation (1).

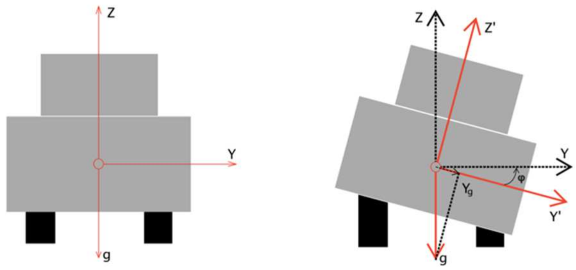

Fig. 12. Car representation.

Because velocity of car is chosen to be constant, X-axis acceleration is not prominent and it is logically to think that the acceleration value generated by car rolling on one side is precisely the interference of gravity vector acting on measurements, not the accelerometer fault.

To conclude interpretation it is considered successful result that gravity interference acceleration (2.15 m/s^2) is comparable to a difference between theoretical Y-axis acceleration and Y-axis acceleration measured by experiment (1.98 m/s^2).

VII. Conclusions

New data acquisition system has been suggested. Incorporating many new features it is seen as a worthwhile to continue developing given system as gathered experimental data seem to be trustworthy and gravity vector interference has been successfully compensated as seen in Section VIII.

As seen from Section VIII system has successfully gathered data that could be used as an input data for, for example, Classical Washout Filter. Being in active development stage many new features might be added if needed although on this stage it is considered to be enough of them already.

Given the simple inertia measurement unit is being used in Measurement device, system could benefit if more advanced sensors will be utilized such as BOSCH "BNO055" where sensor fusion algorithms are used to get more precise measurements of tilt, yaw and pitch angles.

Moreover, it is considered a step forward to achieve HTTPS certificate for Node-RED so that system will become completely secured.

References

[1] N. Pham, J.-H. Kim and H.-S. Kim, «Development of a new 6-DOF parallel-kinematic motion simulator (ICCAS 2008),» *2008 International Conference on Control, Automation and Systems,* pp. 2370-2373, 2008.

[2] Y. S. Chiew, M. K. A. Jalil and M. Hussein, «Motion cues visualisation of a motion base for driving simulator,» *2008 IEEE International Conference on Robotics and Biomimetics,* pp. 1497-1502, 2009.

[3] J.-S. Hu and K.-C. Sun, «A Robust Orientation Estimation Algorithm Using MARG Sensors,» *IEEE Transactions on Instrumentation and Measurement,* т. 64, pp. 815-822, 2015.

[4] A. P. Duc, *A Study on State-of-the-Art Motion Cueing Algorithms applied to Planar Motion with Pure Lateral Acceleration — Comparison, Auto-Tuning and Subjective Evaluation on a KUKA Robocoaster Serial Ride Simulator,* 2017.

[5] Z. Chaczko and R. Braun, «Learning Data Engineering: Creating IoT Apps using the Node-RED and the RPI Technologies,» *2017 16th International Conference on Information Technology Based Higher Education and Training (ITHET),* pp. 1-8, 2017.

[6] Toshiyuki Shimono; Uhuru Corporation, «A Hacking Toolset for Big Tabular Files (codenames: bin4tsv, Kabutomushi),» *2016 IEEE International Conference on Big Data (Big Data),* pp. 2902-2910, 2016.

[7] A. Senfelds, «Analysis of motion modelling approaches for industrial robot applications,» *2019 IEEE 7th IEEE Workshop on Advances in Information, Electronic and Electrical Engineering (AIEEE),* pp. 1-4, 2019.

[8] Electric Imp inc., «Electric Imp Platform UL 2900-2-2 Cybersecurity Certification,» 2020. [Online]. Available: https://developer.electricimp.com/hardware/ul.

[9] Mozilla , «429 Too Many Requests,» 2020. [Online]. Available: https://developer.mozilla.org/en-US/docs/Web/HTTP/Status/429.

[10] Node-RED organization, «Node-RED,» 2020. [Online]. Available: https://nodered.org.

[11] J. Sun and Q. Yang, «Comfort and Safety Analysis of Automotive Active Suspension Based on Adaptive Filter Theory,» 2009 4th IEEE Conference on Industrial Electronics and Applications, pp. 1730-1734, 2009.

This is a post-print of a paper published in Proceedings of the 2020 IEEE 8th Workshop on Advances in Information, Electronic and Electrical Engineering (AIEEE) and is subject to IEEE copyright.
https://doi.org/10.1109/AIEEE51419.2021.9435779
Electronic ISBN: 978-1-6654-2538-4.
Print on Demand (PoD) ISBN: 978-1-6654-3087-6.