

UDC 004.031.2

B.I. Tymchenko

Odessa National Polytechnic University, 1 Shevchenko Ave., Odessa, Ukraine 65044; e-mail: tim4bor@gmail.com

## GLOBAL POSITION SYSTEM SENSOR MODEL FOR ROBOTICS SIMULATOR

*Б.І. Тимченко. Модель системи глобального позиціонування сенсора для робототехнічного симулятора.* Сьогодні гостро стоїть проблема автоматичної навігації для різного типу роботів, безпілотних автомобілів та людей. Збільшення кількості робототехнічних засобів вимагає від них більш точної навігації в просторі. Розробка алгоритмів високоточної навігації потребує велику кількість вихідних даних з сенсорів, а тестування деяких ситуацій є нездійсненним в реальних умовах. Імітаційне моделювання як метод дослідження подібних об'єктів є перспективним у розв'язанні цієї проблеми. Метою роботи є розробка імітаційної моделі для універсального датчика системи глобального позиціонування (GPS) і конфігурованої моделі атмосферних ефектів для імітації вимірювань реального приймача GPS в звичайному середовищі. Для досягнення поставленої мети проведено дослідження роботи GPS приймачів та виконано моделювання. Проблему моделювання сенсору розглянуто для системи геопозиціонування GPS, проведено моделювання земної атмосфери, проте отримані результати можуть бути легко адаптовані для інших сенсорів, наприклад GLONASS та GALILEO. Середовищем моделювання обрано пакет Unreal Engine 4 через його точне фізичне моделювання, що дозволило інтегрувати моделі роботів безпосередньо в модель середовища. За допомогою пакету Unreal Engine розроблено та протестовано модель атмосфери та приймача GPS. Можливість конфігурування моделі дозволила протестувати відповідність моделі до реальних умов навколишнього середовища. Отримана точність відповідно до реального GPS приймача становила більше 95%.

*Ключові слова:* роботизовані транспортні засоби, GPS, атмосфера, обробка сигналів, навігація, симулятор, Unreal Engine

*B.I. Tymchenko. Global position system sensor model for robotics simulator.* Today there is acute problem of automatic navigation for different types of robots, unmanned vehicles and people. Increasing number of robotic vehicles requires them to have more accurate navigation in the environment. Development of algorithms for precise navigation requires a large amount of original data from sensors and it is impossible to test some situations the real world. Simulation as a method to study such objects is promising in solving this problem. The aim of this work is to develop a simulation model for universal global position system (GPS) sensor and configurable models of atmospheric effects to simulate real GPS receiver measurements in the normal environment. To achieve this goal a study of GPS receivers and modeling was done. The problem of modeling of the sensor system is considered for GPS system and Earth's atmosphere, but the results can be easily adapted to other sensors, such as GLONASS and GALILEO. As a simulation package for environment, we chose Unreal Engine 4 because of its precise physical simulation, allowing integration model of the model directly into the environment. Using Unreal Engine package we developed and tested model of the atmosphere and GPS receiver. Possibility of models' configuration allowed us to test compliance of our model to the real environment. The resulting accuracy in accordance with real GPS receiver is over 95 %.

*Keywords:* robotic vehicles, GPS, atmosphere, signal processing, navigation, simulation, Unreal Engine

**Introduction.** Modern computer systems make it possible to solve numerous problems. Especially among all areas interest of modern science stands to the control algorithms and robotics. At present moment, begins the era of robotic vehicles, which would be able to follow the designated route autonomously. Precise autonomous vehicle self-positioning in the outer world is a critical factor that affects traffic safety in general. Many new vehicle features (lane keeping, autonomous routing, and advanced driver assistance) use GPS sensor data fused with other sensors to provide stable positioning on the road and related to map data. This article presents the theoretical development and practical implementation of the GPS sensor and atmosphere signal attenuation model for the simulator environment. This model with the simulator then used to test control algorithms for cars software and car-like robots in a simulated environment. In addition, provided model is suitable for simulating untrusted GPS data for failsafe algorithms design.

**Related works.** The problem of sensor modelling has discussed in many ways in the literature. Meyer et al. [1] references Hector GPS model what does not use atmosphere model at all, providing ideal data to the simulated robot. Several other methods have proposed to estimate the uncertainty of GPS measurements. One of the simplest ways to model GPS uncertainty is to assume a constant value, for instance used by Goel et al. [2]. They claim that it is not desirable to fuse GPS signals with gyroscope and odometry signals using an EKF as low GPS accuracy deteriorates the position estimate on

DOI: 10.15276/opus.3.53.2017.12

©2017 The Authors. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

small movements too much. Another method of modeling the GPS uncertainty is dynamically adjust the covariance according to a parameter. For example, Thrapp et al. [3] used the number of available satellites.

To overcome this, some models use atmospheric effects for GPS propagation calculation, however, they are not configurable. For example, ITU-R [4] have recommendations to use common approximation over all the radio beam from receiver to satellite. Hum et al. [5] use atmospheric refraction to model uncertainty in GPS readings. All these models are good for engineering purposes or for estimate of the GPS error in robotics or navigation applications. They are not suitable for robotics simulation due to their lack of configurability.

**The aim.** Nowadays there are a many concepts of automatically driven vehicles, such as Google cars, Mercedes approaches, Tesla Autopilot, etc. The first step to design such complex systems is the simulated model or prototype. To design the simulated model of vehicle, we need robust and mainly precise simulation of the environment where the model will executed. Contemporary automotive algorithms use machine learning and predictive control. For stable positioning sophisticated algorithms are used. Sensor fusion is a common approach what involves such sensors as GPS, odometry, cameras, LIDARs and range sensors. Every kind of these sensors has its own noise characteristics. Sensor fusion techniques are sensitive to noise characteristics of sensors used in equipment [6].

For GPS sensor, surrounding environment (buildings, trees) and atmospheric effects high influence on its precision and error distribution. For example, buildings mostly affect accuracy of the GPS sensor and atmospheric effects affect its precision.

The aim of this work is to develop simulation model for generic GPS sensor and configurable model for atmospheric effects in order to simulate measurements of real GPS receiver in realistic environment.

**GPS system description.** The GPS system concept is based on time and known positions of satellites at this time. GPS constellation consists of 31 satellites with orbit height of 20180 km. Nowadays constellation consists of 6 orbital planes with approximately 55° inclination. Anywhere in the world, there are at least 6 satellites visible at the same time. At least 4 satellites needed for reliable position determination.

Each GPS satellite continuously broadcasts its position and time. A GPS receiver tracks multiple satellites and solves equation for its position and time deviation from real GPS time. These signals are transmitted on two separate carrier frequencies that are common to all satellites in the network. Two different encodings used: a public encoding that enables lower resolution navigation, and an encrypted encoding used by the U.S. military. Precise sensors also use phase difference between carriers to eliminate some of the ionospheric effects. We will not cover this case in the article.

To model GPS orbit parameters, we use almanac file. The almanac consists of coarse orbit and status information for all satellites period. In addition, it contains ionospheric correction model that will be described further. Almanac files in RINEX (Receiver Independent EX change) format is common tool for GPS supplementing and they are provided by many observatories all over the world. We use RINEX files provided by NASA observatory. In our model, the satellite positions are predicted only based on almanac record for certain hour of the day and are valid for two hours from this time. GPS ephemerides are accurate up to 10 m.

**Coordinate frames in model.** We are working in Unreal Engine 4 environment, so our sensor is just a regular object in scene. To simplify development we use a priori known position in local coordinate frame (GLF, game level frame) and then add errors according to modelled situation. We use approximation of local tangent plane for small pieces (up to 4 km). This was considered acceptable, as mean distance from this plane to the satellites is sufficiently larger. Using these considerations, we can use ECEF (earth-centered, earth-fixed) coordinate frame for both sensor and satellites (Fig. 1), which is suitable in error model calculation. To get user output, noised ECEF coordinates are then converted to generic LLA coordinates (latitude-longitude-altitude). Velocity and heading are computed on the base of these LLA coordinates.

**GPS error sources.** There are two factors affect overall accuracy of the GPS system. The quality of visible constellation geometry represented by multiplicative factor denoted as dilution of precision

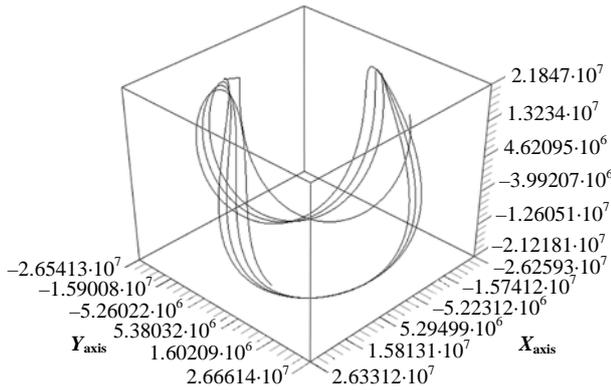


Fig. 1. Trajectory of the satellite in ECEF frame for 46 hours (values in meters)

(DOP) and errors in pseudo range estimation represented by user equivalent range error (UERE). Overall positioning error is expressed as following:

$$E = \text{DOP} \cdot \sigma_{\text{uere}},$$

where  $\sigma_{\text{uere}}$  – standard deviation for UERE.

Standard deviation  $\sigma_{\text{uere}}$  is affected by surrounding buildings (multipath effects), atmospheric effects and clock imprecision. In our model, we assume multipath effects only if the satellite is not directly visible, as conventional GPS receivers can successfully filter out this case of multipath error.

When the satellite is not directly visible, we consider each reflection of signal from building

wall consumes half of the beam power regardless to material of the wall. If there is no viable path with up to 16 reflections to satellite. This satellite is considered as invisible at the current point at all. We do not consider the clock imprecision.

**Atmospheric model.** We implemented fully configurable model of the Earth atmosphere with following objects: troposphere, stratosphere, mesosphere, thermosphere, ionosphere, clouds of different types and rain. Atmosphere is the main factor of UERE, as it both signal-to-noise ratio (SNR) and propagation delay (PD) of the satellites. To deal with SNR calculation, we simulate propagation of the beam with known start energy through attenuating atmospheric layers. We consider layers as uniform for simplicity. Gas molecules interact with the radio wave electromagnetic field. This may cause energy loss, e.g.,  $\text{H}_2\text{O}$  molecules are asymmetric and align to the electric field.

In Earth atmosphere, main attenuation factors at GPS frequencies (1575.42 MHz) are oxygen and water vapor. We calculate attenuation of each layer separately for convenience of model configuration. In this calculation, we use spherical approximation of earth instead of the geoid, as we already reject satellites that have elevation lower than 5 degrees. For higher elevation angles, difference between geoid and spherical approximation is negligible. When the path length through each layer is found, we calculate vapor and oxygen attenuation. We use nonlinear curve fit for equivalent concentration relative to height ( $h$ ):

$$k = e^{-\frac{gmh}{RT}},$$

where  $g$  – gravitational constant,

$m$  – molar mass of dry air,

$R$  – universal gas constant,

$T$  – temperature at sea level in the given point.

Equivalent concentration then is used in signal attenuation calculation.

We use Mason models [7] for attenuation with equivalent gas concentration correction. For water vapor, the model is following:

$$A_{\text{vape}} = \left\{ \begin{aligned} &0.05 + 0.0021 \cdot k\rho + \frac{3.6}{(f - 22.2)^2 + 8.5} + \\ &+ \frac{10.6}{(f - 183.3)^2 + 9.0} + \frac{8.9}{(f - 325.4)^2 + 26.3} \end{aligned} \right\} \cdot f^2 \cdot k\rho \cdot 10^{-4},$$

for oxygen attenuation, we use:

$$A_{02} = \left( 7.19 \cdot 10^{-3} + \frac{6.09}{f^2 + 0.277} + \frac{4.81}{(f - 57)^2 + 1.5} \right) \cdot k \cdot f^2 \cdot 10^{-3},$$

where  $k$  – correction term,

$\rho$  – water vapor concentration in  $\text{g/m}^3$ ,

$f$  – frequency in GHz.  $A$  is expressed in dB/km.

Integrating attenuation over beam path we achieve total atmospheric losses in signal power.

Clouds also have sufficient impact on the signal strength. We divide several types of clouds by their density from fog to thunderstorm. They are considered as water vapor of predefined density [1], we compute clouds attenuation in the same way as for water vapor in the atmosphere. Now we consider cloud as a thick layer at the given height, not considering its complex shape and borderline refraction. At the receiver, signal strength and SNR of satellite is computed as:

$$P_{rec} = P_{sat} - L_s - L_{atm} - L_{iono} - L_{clouds} \quad (dB),$$

$$SNR = P_{rec} - P_{ambient}.$$

$P_{ambient}$  is a normal ambient noise, which is a thermal noise and has power of  $-204$  dBW. In addition to thermal noise, we add surrounding electromagnetic noise caused by environment. Additional noise power varies from 1 dBW in countryside to 40 dBW in industrial area. Cutoff SNR for satellite is 5 dBW, lower than this margin, satellite is considered as unreliable and not taken into position fix.

**Ionospheric and tropospheric delays.** Free electrons in the ionosphere and troposphere cause GPS signal to travel slower. This phenomenon affects time calculation and thus directly affects position error. We compute slant path through ionosphere and delay on this path based on total electron content with fitted curve [8]:

$$E_{iono} = 0.162 \cdot (1 + 2.74 \cdot 10^{-6} \cdot (96 - e_{sat})^3) \cdot TEC,$$

where  $TEC$  – total electron content,

$e_{sat}$  – elevation angle in degrees.

Average error for ionosphere is 5 m.

In addition, we use mapping of Niell tropospheric delay model [9]. Tropospheric delay consists of two factors. Hydrostatic component delay caused by the dry gases present at the troposphere its effect varies with local temperature and atmospheric pressure in quite a predictable manner. The error caused by this component is about 2.3 meters in the zenith direction and 10 meters for lower elevations.

Wet component delay caused by the water vapor and condensed water in form of tropospheric clouds and, thence, it depends on weather conditions. The excess delay is small in this case, up to 0.5 m, but this component varies faster than the hydrostatic component and a quite randomly way, being very difficult to model. We include only hydrostatic delay into our model.

Hydrostatic delay is estimated using no meteorological surface data. It considers different obliquity for wet and dry components:

$$E_{tropo} = E_{z,dry} \cdot M_{dry}(e_{sat}, h) + E_{z,wet} \cdot M_{wet}(e_{sat})$$

We consider  $E_{z,wet}$  as null for simplicity.  $e_{sat}$  is elevation angle in degrees,  $E_{z,dry}$  and  $M_{dry}$  are components of the mapping of Niel and denoted as following:

$$E_{z,dry} = ae^{-bh},$$

$$M_{dry} = m(e, a_d, b_d, c_d) + \Delta m(e, h),$$

$$\Delta m(e, h) = \left[ \frac{1}{\sin(e)} - m(e, a_{ht}, b_{ht}, c_{ht}) \right] \cdot h,$$

where  $h$  – height over sea level,  $m(e, a, b, c)$  is the Marini mapping [10] normalized to unity at zenith:

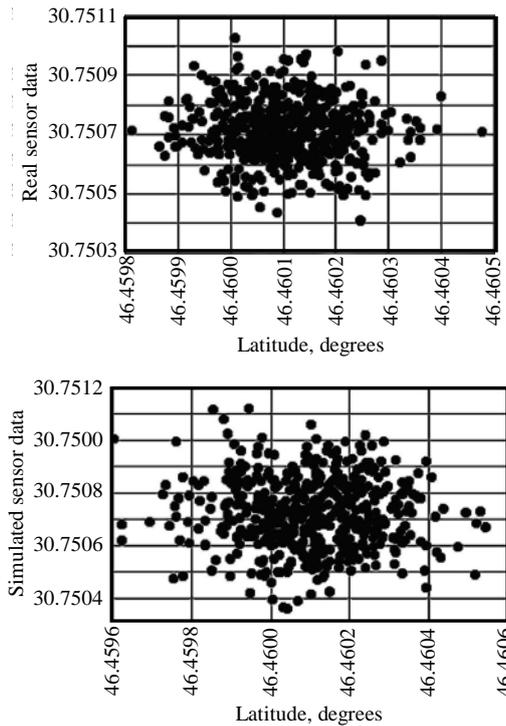


Fig. 2. Examples of data from real and simulated sensors set at the same place

3600 measurements and checked errors distribution (mean and standard deviation). Difference between real and simulated sensors is 3.256 %. Fig. 2 shows the comparison of the part of data for the same spot.

**Conclusions.** In a result of this work, the model of GPS sensor in simulated environment was presented, its accuracy reached 96.744 % compared to live sensor. This model takes into account geometric, atmospheric and ionospheric effects. In addition, test environment and live setup was built for testing and parameters estimation. Achieved results can be used for various types of applications, where precise GPS errors modelling is needed.

## Література

1. Meyer J., Sendobry A. Comprehensive Simulation of Quadrotor UAVs using ROS and Gazebo. Sendobry. Heidelberg: Springer, 2012. 401 p.
2. Goel P., Stergios I. Roumeliotis, Gaurav S. Robust localization using relative and absolute position estimates. *Intelligent Robots and Systems, IROS '99*. Proceedings, 1999. vol. 2. P. 1134–1140.
3. Robust localization algorithms for an autonomous campus tour guide/ Thrapp R. et al. ICRA. IEEE, 2001. P. 2065–2071.
4. International Telecommunication Union, ITU-R Recommendation “Attenuation by atmospheric gases”, Geneva, 2016.
5. Atmospheric effects./ Hum V. et al. ECE422: Radio and Microwave Wireless Systems: Toronto, 2012.
6. A Simulation Environment for Benchmarking Sensor Fusion-Based Pose Estimators/ Ligorio G. et al. MDPI, 2015.
7. Mason S. Atmospheric Effects on Radio Frequency (RF) Wave Propagation in a Humid, Near-Surface Environment. California: Naval Postgraduate school, 2010.
8. Giffard R. Estimation of GPS Ionospheric Delay Using L1 Code and Carrier Phase Observables. *31st Annual Precise Time and Time*. Interval Meeting. USA, Palo Alto, 1999. 407–418.

$$m(e, a, b, c) = \frac{1 + \frac{a}{1 + \frac{b}{1 + c}}}{\sin(e) + \frac{a}{\sin(e) + \frac{b}{\sin(e) + c}}}$$

Parameters  $a_d$ ,  $b_d$ ,  $c_d$  are time and latitude depend with following formula:

$$\gamma(\varphi, t) = \gamma_{avg}(\varphi) - \gamma_{amp}(\varphi) \cdot \cos\left(2\pi \frac{t - T_0}{365.25}\right),$$

where  $t$  – time from January 1<sup>st</sup>, in days,

$T$  – taken as day of year.

The parameters  $\gamma_{avg}$  and  $\gamma_{amp}$  are taken from original work [10]. Using tropospheric model that does not depend on direct meteorological measurements allowed us testing our model without need of costly measurements.

**Experimental results.** To test the model, we built virtual map of Odessa region and placed sensor on the terrain. Real sensor was installed on the same spot in Odessa. We took 3600 measurements from real sensor (1 hour, 1 measurement per second). Then we configured model for the same environment conditions and loaded RINEX file for this time. We simulated another

9. Niell, A. Global mapping functions for the atmosphere delay at radio wavelengths. *Journal of Geophysical Research*, 1996. P. 3227–3246.
10. Marini J. Correction of Satellite tracking data for an arbitrary Tropospheric Profile. *Radio Science*, 1972. P. 223–231

## References

1. Meyer, J. (2012). *Comprehensive Simulation of Quadrotor UAVs using ROS and Gazebo*. Heidelberg: Springer.
2. Goel, P., Stergios, I. Roumeliotis, & Gaurav, S. (1999). Robust localization using relative and absolute position estimates. *Intelligent Robots and Systems, IROS '99*. Proceedings. 2, 1134–1140.
3. Thrapp, R. et al. (2001). *Robust localization algorithms for an autonomous campus tour guide*. ICRA. IEEE. 2065–2071.
4. International Telecommunication Union (2016). *ITU-R Recommendation "Attenuation by atmospheric gases"*, Geneva.
5. Hum, V. (2012) Atmospheric effects. ECE422: Radio and Microwave Wireless Systems: Toronto.
6. Ligorio, G. (2015). *A Simulation Environment for Benchmarking Sensor Fusion-Based Pose Estimators*. MDPI.
7. Mason, S. (2010). *Atmospheric Effects on Radio Frequency (RF) Wave Propagation in a Humid, Near-Surface Environment*. California: Naval Postgraduate school.
8. Giffard, R. (1999). Estimation of GPS Ionospheric Delay Using L1 Code and Carrier Phase Observables. *31st Annual Precise Time and Time Interval Meeting*. 407–418.
9. Niell, A. (1996). *Global mapping functions for the atmosphere delay at radio wavelengths*. Journal of Geophysical Research. 3227–3246.
10. Marini, J. (1972) Correction of Satellite tracking data for an arbitrary Tropospheric Profile. *Radio Science*. 223–231.

Received May 10, 2017

Accepted September 13, 2017