



HAL
open science

On The Impact Of Temporal Variation On GNSS Position Error Models

Syed Ali Kazim, Nourine Ait Tmazirte, Juliette Marais

► **To cite this version:**

Syed Ali Kazim, Nourine Ait Tmazirte, Juliette Marais. On The Impact Of Temporal Variation On GNSS Position Error Models. ION ITM 2021 - International Technical Meeting of Institute of Navigation, Jan 2021, Virtual, France. 10p. hal-03157781

HAL Id: hal-03157781

<https://hal.science/hal-03157781>

Submitted on 31 May 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

On The Impact Of Temporal Variation On GNSS Position Error Models

Syed Ali Kazim, *COSYS-LEOST, Univ Gustave Eiffel, France*

Nourdine Aït Tmazirte, *Institut de Recherche Technologique Railenium, France*

Juliette Marais, *COSYS-LEOST, Univ Gustave Eiffel, France*

BIOGRAPHY

Syed Ali Kazim has done a master degree in Electronic Engineering. He received a specialization degree in Navigation and Related Applications from Politecnico di Torino (Italy) in 2017. He is currently a Ph.D. student at University Gustave Eiffel. His research activities are mainly devoted to GNSS signal characterization, error modeling and localization systems.

Nourdine Aït Tmazirte works at French Institute of Technology Railenium since 2018. He got his engineering and M.Sc. degree in automation engineering from Ecole Centrale de Lille, France, both in 2010. His research interests include multi-sensor fault-tolerant fusion for localization and integrity assessment.

Dr. Juliette Marais received an engineering degree from Institut Supérieur de l'Electronique et du Numérique. She received a Ph.D. degree in electronics and the "Habilitation à diriger des recherches" from the University of Lille, France, in 2002 and 2017 respectively. Since 2002, she has been a research fellow with Université Gustave Eiffel (former IFSTTAR). She is involved in two main research projects: integrity monitoring for land transport applications and GNSS propagation characterization in railway environments, with some contributions in EU or national projects. Her research interests principally include propagation phenomena, positioning and pseudorange error modeling, filtering technics, and simulation.

ABSTRACT

The accuracy of the satellite-based positioning system is severely affected in the presence of local phenomena like multipath, NLOS and interferences. These local effects are difficult to model due to spatial and (or) temporal dependencies. This study aims to model the position error on the track and later use it for the track classification. The characterization of the track errors can be beneficial as it can help to increase the capacity of the railway track by providing adaptable protection level and also for sending warning alerts to the other trains if any of them is in degraded mode to avoid any sort of accidents. We present a very simple machine learning approach to model the position error. For this purpose, we investigate the Expectation-Maximization (EM) algorithm to estimate the parameters of the Gaussian Mixture Model (GMM). We intuitively selected 3 classes to represent different error shape features to avoid complexity and to keep some diversity. In the learning phase, 5 different runs are used to capture as much as the error variations on the track throughout the day. The models estimated in the learning process is applied to the test samples for the track characterization. The results show that class representation varies due to the error variation within the day. For some particular obstacles, the track is always represented by the class that shows deteriorated conditions. It also shows that the class representation is very similar to the learning samples when the error models are applied to the test sample collected on a different day with a nearly similar satellite configuration.

Introduction

With the growing potential of GNSS-based solutions, the number of safety-relevant applications of GNSS is continuously increasing. These applications require verification and validation of their algorithms that can be complex for example, in the land transportation, the area where the performance is strongly dependent on the reception environment. In order to assess their performance in the various scenarios and types of environments that can be encountered, the development of GNSS simulators is highly investigated. Simulation tools offer several advantages: They make it possible to reduce development costs while considering various scenarios and validation environments. They also facilitate the labeling of faults such as multipath, NLOS or interferences that are often not known in real datasets, as well as the a priori knowledge of their occurrence where the fault injection techniques can play their role. Finally, the ability to generate a large amount of data. The contribution of a simulator in the preliminary phases of the development cycle of these functions is a major asset to complement and limit the need for real data and complex experimentation campaigns.

Railway innovation programs are interested and involved in these developments. The Shift2Rail European rail initiative focused on research and innovation supports the development of tools to push towards zero-on site testing and the development of new

laboratory test frameworks. The Gate4rail project [1] funded in this context develops a geo-distributed simulation and verification infrastructure which can evaluate the performances and properties of some fail-safe train positioning components in nominal and fault conditions.

The difficulty comes with the need to propose realistic error models that reflect the system behavior no matter the time or the place. The closer the simulator will represent real cases, the better they will be accepted for verification and validation purposes in a certification process.

Land transport environments are multiple and complex. GNSS effects as well as related positioning performance are temporal and spatial dependent. As illustrated in [2], several runs are required to reliably characterize GNSS effects along a railway line. Indeed, satellites are moving with time, and the environment around the antenna is changing along the run. Moreover, if train runs can show a certain level of repeatability (same departure time and arrival every day for example), as long as the train is driven by a human driver, the dynamic of the train cannot be perfectly identical between two different runs. In the ERSAT GGC project, the tracks have been classified considering GNSS local events (multipath, NLOS, interference...). In [4] the GNSS data have been used to classify the environment type along the track. In this paper, we intend to characterize the track based on position errors.

In a previous study [3] we proposed an environment classification of a railway track and we have divided, as an example and without being exhaustive, the track into three classes, namely Open sky, Urban/Suburban areas and Forests. For each of the chosen classes, we have presented a dedicated error model. The results showed the inability to provide Gaussian-type models, insofar as different behaviors were likely to appear in the same class (different types of urban environment, building heights, etc. the same class considered as urban).

The objective is to provide a model representing the positioning error behaviour along a track that will remain representative enough of real errors and that will cover the different ranges of the errors encountered, in order to be used by the simulator for intensive tests and validation of performance. In the present study, we propose to investigate some machine learning techniques for error classification, without the need for any a priori information on the environment.

The paper is divided as follows: In the first section, we will provide background on error representation, some brief view on the chosen machine learning methodology and the classification creation. The second section will be on the results and analysis where we will present the GMM curves representing the different error shape features. We will also present the track characterization for different runs in different periods of the day to show variations in the classification as a result of error variations. We will also apply the models to test samples (different from the learning samples collected on a different day but in the same period) to show the consistency in the classification.

POSITION ERROR MODELING AND TRACK CHARACTERIZATION

We presume here the errors in the GNSS to be Gaussian distributed. We expect that the typical environments around the track characterized in [3] will have show particular effects from the surroundings that can give rise to multi Gaussian error distribution. In this study, we aim to find multi Gaussian distribution that is usually engulfed under one large Gaussian distribution curve and use it for the track characterization. For this purpose, we used Expectation-Maximization (EM) for Gaussian Mixture Model (GMM).

Position error computation and representation into track frame

This study is related to the position error modeling of a satellite-based positioning system. It requires the estimation of the receiver position. A post-processing tool (RTK-post) is used to obtain the position from RINEX experimental records. The position is estimated using a classical Least Square (LS) algorithm from the GPS and Galileo observations. The position error is computed from the difference between the reference and the estimated positions. The position error is given as:

$$E_{\text{pos}} = X_{\text{measured}} - X_{\text{ref}} = \begin{bmatrix} E_{\text{East}} \\ E_{\text{North}} \end{bmatrix}. \quad (1)$$

where X_{measured} and X_{ref} represent the estimated and the reference positions; E_{East} and E_{North} represent the position error components in the East and North directions.

In the railway's operational point of view, the more suitable alternative to the Horizontal Positioning Error (HPE) is the along-track error (ATE) and cross-track error (XTE) components. Indeed both error components have different impacts: the undesirable event with XTE is a wrong track discrimination and with ATE, the risk is collision with another train. This is why, in the following, the position errors are projected to the track frame.

It is given as:

$$E_{fwd_lat} = \begin{bmatrix} E_{forward} \\ E_{lateral} \end{bmatrix} = R \cdot E_{pos} \quad (2)$$

$$R = \begin{bmatrix} \cos \vartheta & \sin \vartheta \\ -\sin \vartheta & \cos \vartheta \end{bmatrix} \quad \text{and } \vartheta = \tan^{-1}(ref_{\Delta N}/ref_{\Delta E}) \quad (3)$$

where R represents the rotation matrix, $ref_{\Delta E}$ and $ref_{\Delta N}$ are the distances of two consecutive reference points in the East and the North direction respectively. It is expected that two error components will be affected differently by surrounding environmental conditions. It will vary depending on the orientation of the railway line and the placement of the nearby obstacles.

EM-GMM learning approach and track characterization

The term GMM indicates the mixture of several Gaussian distributions that represent the normally distributed subpopulations within the complete data samples. It is the form of unsupervised learning, as it does not require a priori knowledge about the membership of the data samples. It automatically performs the soft assignment with the help of the model.

A Gaussian mixture model is parameterized by the component weights, means and variances/covariances. A Gaussian mixture with K components, the k^{th} component in the multivariate case is described by the mean μ_k , covariance matrix Σ_k and component weights w_k for the component C_k . The Gaussian density function is given by [8].

$$\mathcal{N}(X | \mu_k, \Sigma_k) = \frac{1}{\sqrt{(2\pi)^{D/2} |\Sigma_k|^{1/2}}} \exp(-0.5(x_n - \mu_k)^T \Sigma_k^{-1} (x_n - \mu_k)) \quad (4)$$

where D is the number of variables.

Linear combination of Gaussian distributions is given as:

$$\mathcal{M}(\vec{x} | \Theta) = \sum_{k=1}^K w_k \mathcal{N}(x_n | \mu_k, \Sigma_k) \quad (5)$$

Where Θ is the set of parameters given as, $\Theta = \{\{w_1, w_2, \dots, w_K\}, \{\mu_1, \mu_2, \dots, \mu_K\}, \{\Sigma_1, \Sigma_2, \dots, \Sigma_K\}\}$, K is the number of Gaussian distributions and w_k is the mixture ratio of the k^{th} distribution.

The GMM parameters for the track errors are estimated by the EM algorithm. It is a 2-step iterative process that at first performs the Expectation (E) step to evaluate the expected likelihood from the currently estimated parameters. Then it performs the Maximization (M) step, which finds the model parameter that maximizes the expected value of the log-likelihood function. This process repeats until convergence.

These models are applied to the test samples (track errors) that are different from the samples used in the learning phase. The track errors from the test sample are classified based on the criterion to maximum expectation value. Finally, the track characterization is represented in a kml format where the color points represent the classification. Figure 1 represents the complete methodology used for track error modeling and classification.

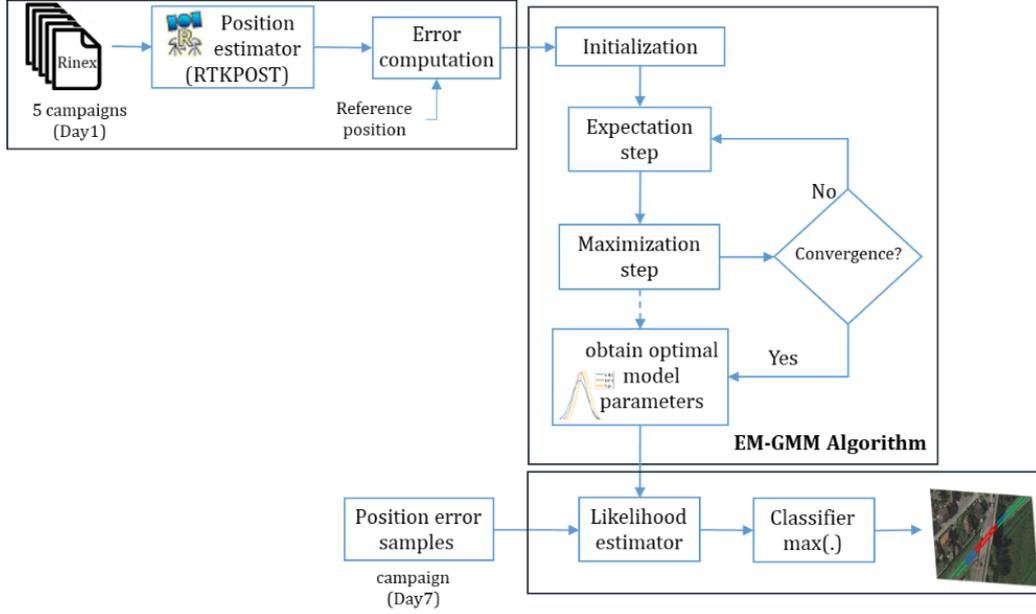


Figure 1. A complete methodological flow describing the learning method (EM-GMM algorithm) and the classification criterion for the track characterization.

The corresponding algorithms are also given (algorithms 1 and 2) before application on real data.

```

Algorithm 1: EM-GMM algorithm for track error modeling.

Input :    $X = \{x_1, x_2, \dots, x_N\}$  // training data
Output:    $w = \{w_1, w_2, \dots, w_K\}$ 
           $\mu = \{\mu_1, \mu_2, \dots, \mu_K\}$ 
           $\Sigma = \{\Sigma_1, \Sigma_2, \dots, \Sigma_K\}$ 

Random initialization  $w, \mu, \Sigma$ 
Convergence = False;
while (Convergence == False) or (t<=maxIter) do
    // E-step
    for  $n = 1 : N$  do
        for  $k = 1 : K$  do
            
$$\gamma_n(z_k) = \frac{w_k N(x_n | \mu_k, \Sigma_k)}{\sum_{i=1}^K w_i N(x_n | \mu_i, \Sigma_i)}$$

        end
    end
    // M-step
    for  $k = 1 : K$  do
        
$$\mu_k = \frac{\sum_{n=1}^N \gamma_n(z_k) x_n}{\sum_{n=1}^N \gamma_n(z_k)}$$

        
$$\Sigma_k = \frac{\sum_{n=1}^N \gamma_n(z_k) (x_n - \mu_k)(x_n - \mu_k)^T}{\sum_{n=1}^N \gamma_n(z_k)}$$

        
$$w_k = \frac{1}{N} \sum_{n=1}^N \gamma_n(z_k)$$

    end
if  $L_t - L_{t-1} < \epsilon$  then // log likelihood convergence check
    Convergence = True
end(if)
end(while)

```

```

Algorithm 2: GMM classification algorithm for track
characterization.

Input :    $X = \{x_1, x_2, \dots, x_N\}$  // test data
           $w = \{w_1, w_2, \dots, w_K\}$ 
           $\mu = \{\mu_1, \mu_2, \dots, \mu_K\}$ 
           $\Sigma = \{\Sigma_1, \Sigma_2, \dots, \Sigma_K\}$ 
Output :   $y = \{y_1, y_2, \dots, y_K\}$  // class labels
for  $n = 1 : N$  do
    for  $k = 1 : K$  do
        
$$\gamma_n(z_k) = \frac{w_k N(x_n | \mu_k, \Sigma_k)}{\sum_{i=1}^K w_i N(x_n | \mu_i, \Sigma_i)}$$

    end
    
$$y_n = \arg_k \max \gamma_n(z_k)$$

end

```

RESULTS AND ANALYSIS

Data acquisition and experimental hardware

The experimental data were acquired along a railway operational line in Switzerland in 2018 under the framework of the EU STARS project [9]. For this study, we have processed 5 different campaign-runs collected at different times of a same day for the same train track and another run at the same time but on a different day.

The train track is nearly 25 km long that starts from Locarno station and ends at Castione-Arbedo station. In each run, the acquisition is done for nearly 1 hour and data is collected at the rate of 1 Hz using Septentrio (ASTERx4) receiver. Table 1 contains some other information about each run: acquisition date, starting time of data collection and the number of samples. The ground truth positions are provided with the data and are used as a reference in the error computation. The track shown in Figure 2 incorporates typical obstacles like spare trees, forests, tunnels, mountains, metallic and overhead bridges, buildings and stations in urban/suburban areas.

Table 1. Summary of the campaigns used in the paper

Acquisition day	Start time (24hr format)	Number of samples
09 Jan 2018 (learning data)	09:59	1595
	13:59	1640
	15:59	1504
	18:03	1608
	19:59	1754
16 Jan 2018 (test data)	10:00	1556

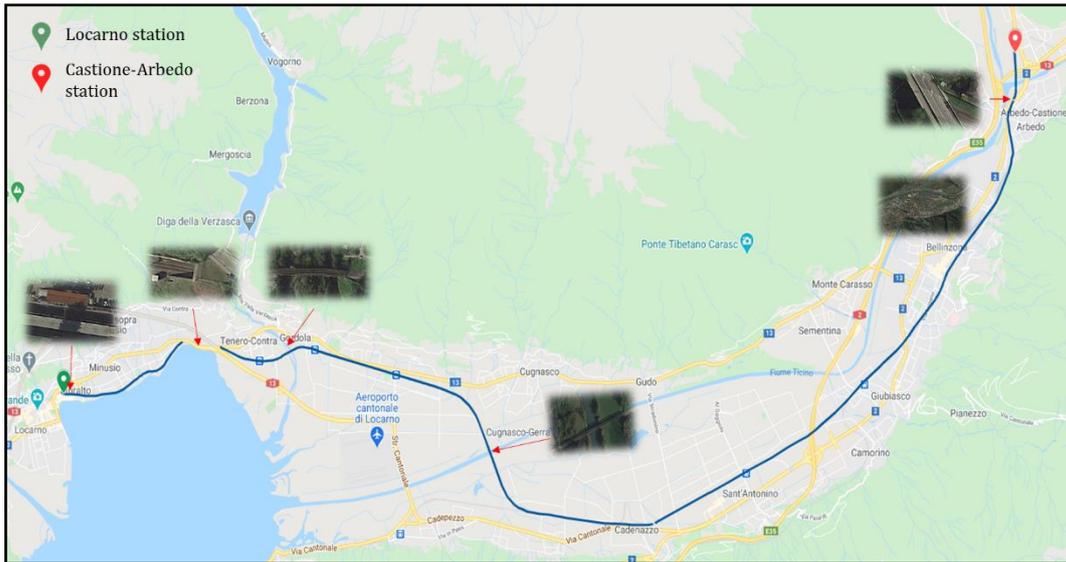


Figure 2. Railway line in Switzerland where the acquisition campaigns have been performed with some sample obstacles (small window) present around the track.

Learning/clustering results

As justified before, we have chosen to present positioning errors as Along-track error (ATE) and Cross-track error (XTE). Figure 3 shows the histogram of the along-track and cross-track position error components. Therefore, the two error components are used separately in the learning process to have a higher degree of freedom. The algorithm extracts different shape features of the error present in the samples.

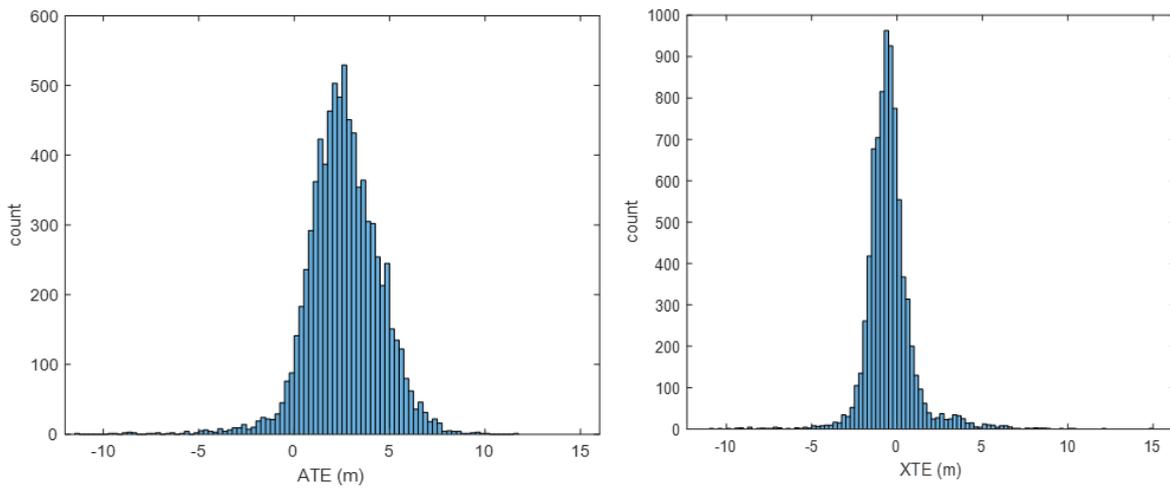


Figure 3. Histogram of the data samples used in the learning process representing track error components Along-track error (ATE) (left panel) and Cross-track error (XTE) (right panel).

Figure 4 shows the scatter plot of the track errors, where the x-axis and the y-axis represent the along-track and the cross-track error component respectively. The EM-GMM algorithm as discussed in the previous section is applied to cluster the samples in 3 different groups. The choice of the number of classes is out of scope for this study. Here, as a first investigation, we arbitrarily selected 3 classes to avoid complexity and to keep some diversity. Each sample in the scatter plot is represented by the RGB color shades. The color proportions are set from the expectation value of the sample belonging to each of the classes. For example, a given sample with a likelihood $[0.1, 0.2, 0.7]$ is 70% probable that it belongs to the blue color class. For this reason, the EM-GMM algorithm is known for a soft assignment.

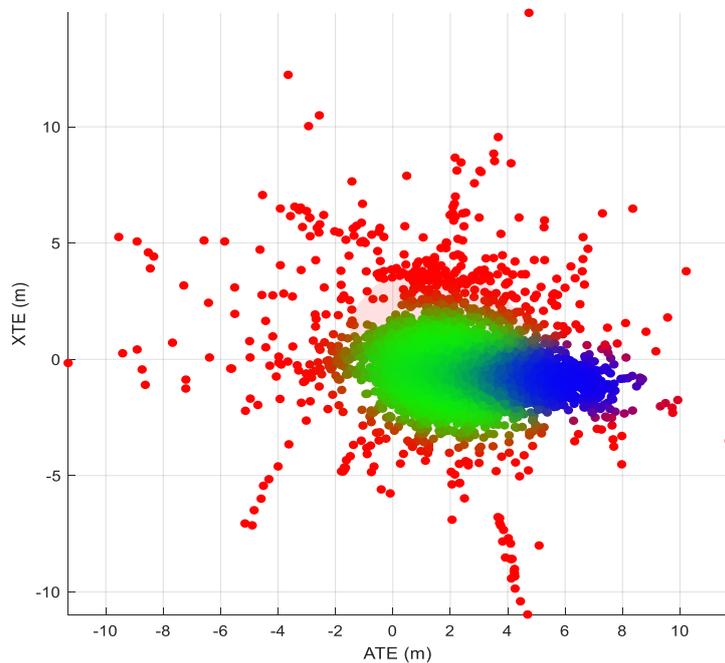


Figure 4. Scatter plot of the samples used in the learning process with Along-track error (ATE) and Cross-track error (XTE) component. The samples belonging to the Run (1-5) are combined to get 3 (user-defined classes) error models of different shape features.

Figure 5 shows the track characterization of the samples that were used in the learning process. Here, each reference position is assigned a class group (green, blue and red color) represented by the same color convention as used in the scatter plot. In this step, it is a hard class assignment which is done by considering the probability values i.e. a sample is assigned a class group

where it has the highest probability. It is to represent the variation of the error class features throughout the day for the same track. This behavior is very particular to satellite-based positioning systems, as the satellite's visibility varies continuously so as the signal interaction with the permanent obstacle present along the trackside. It is interesting to see that some track sections still show very similar effects at different times of the day. As a sample case, an example of a bridge with a metallic structure is illustrated wherein each run, this section is majorly characterized by red color. This shows that the GNSS based position in most of the day is severely degraded when the train passing over the bridge.

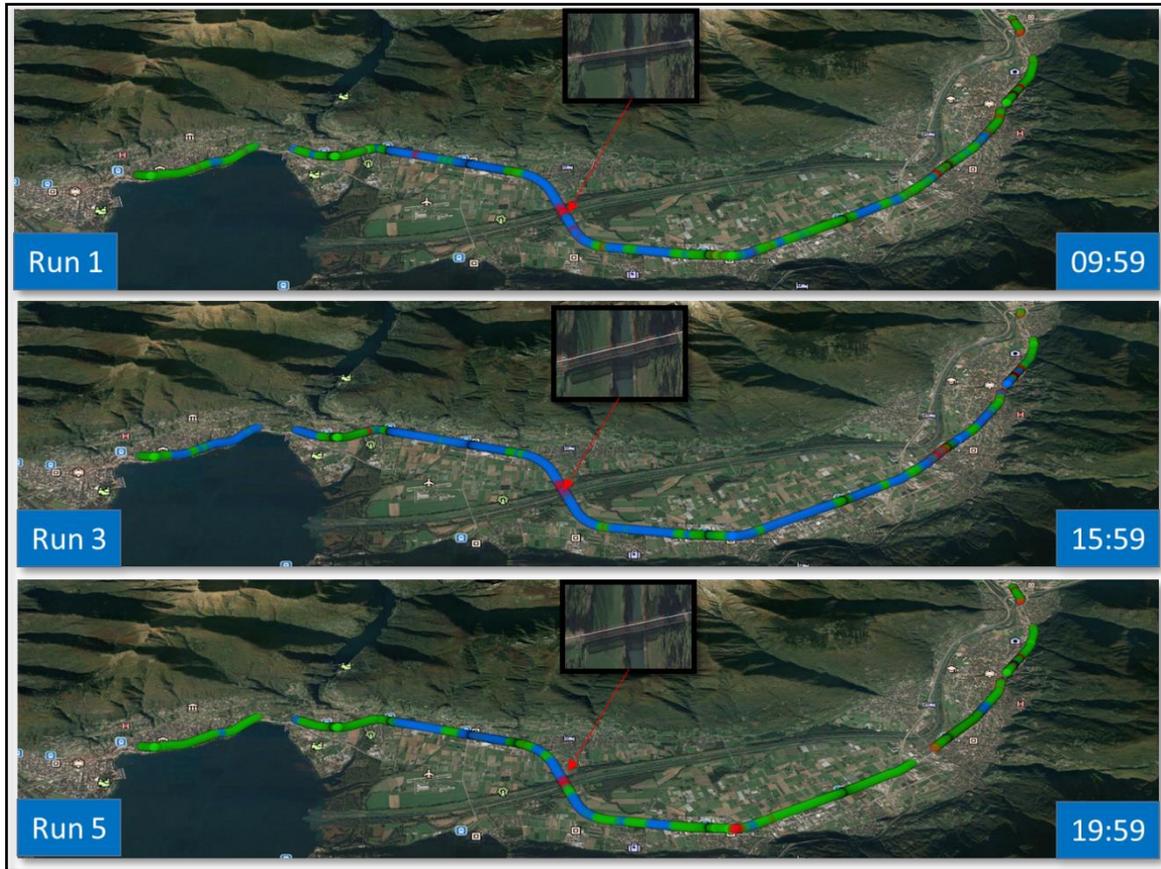


Figure 5. The Google map representing the track characterization for Run 1, 3 and 5 with the clustering outcome provided by the learning algorithm. The small frame points at the region where the bridge is located which is classified as a red color class.

Figure 6 shows the 3D Gaussian curves to represent the error model for each of the 3 clusters. Let's notice that these curves are only representative of the track error samples used in the learning process. The models are labeled with respect to the positioning error modes 1, 2 and 3. Mode 1 appears to be representative of relatively small positioning errors, Mode 2 moderate errors with much more impact on the ATE component and Mode 3 constituted by large errors that are present in the queues. It can be seen that the error models represented by green and blue ellipses share a common area in which the samples have nearly the same or high enough probability not to be neglected from any of the 2 classes. The overlapping area will certainly affect the classification outcome. This effect is also present in Figure 5 where some points have nearly the same probability to be in the blue or the green class. In [5][6][7] the problem of overlapping clustering is discussed and strategies are presented to increase the effectiveness of the clustering that will be investigated in further researches. To some extent, it could also be achieved by finding the optimal number of the cluster for the provided samples. As mentioned before, the aspect of the number of clusters suitable for the given samples is out of the scope of this study and is not discussed here.

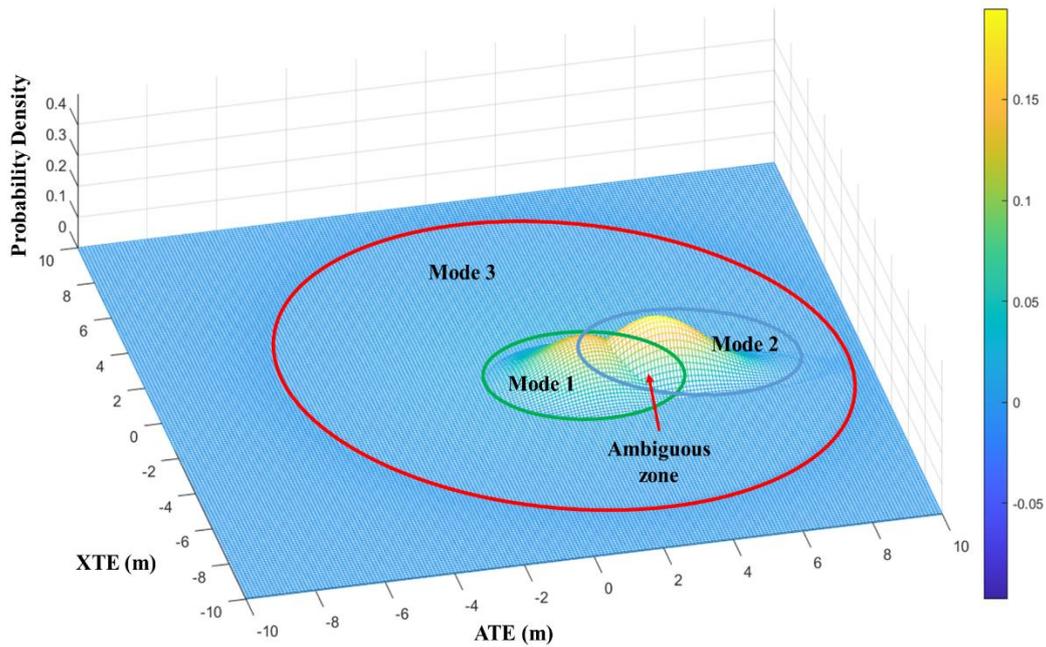


Figure 6. The curves representing the Gaussian mixtures present in the learning dataset (Run 1-5). Three modes are defined: mode 1 (small errors), mode 2 (moderate error with much more impact on the ATE component) and mode 2 (large error the fall on the queues with very low occurrence).

Classification with the test data

Figure 7 shows the track characterization from the clustering outcome of Run 1 and when the model is applied to the test dataset (Run 6). The data collected in Run 1 and Run 6 are at the same time but 7 days later to Run1. The model parameters estimated in the learning phase from Run 1-5 were applied to find the likelihood of the error samples of Run 6. This is to show the variation of error classification on a different day. The overall classification outcome of the learning samples and the test samples seems to be identical with the changing day. In particular, the largest errors are identified in the same areas as colours are globally comparable. They are certainly not point-to-point identical as the comparison is limited by the fact that we did not take into consideration the speed of the train and also the different repeat times of the used satellite constellation. As for the GPS, nearly the same configuration of satellites is visible with approximately 4 mins shift between the 2 consecutive days.

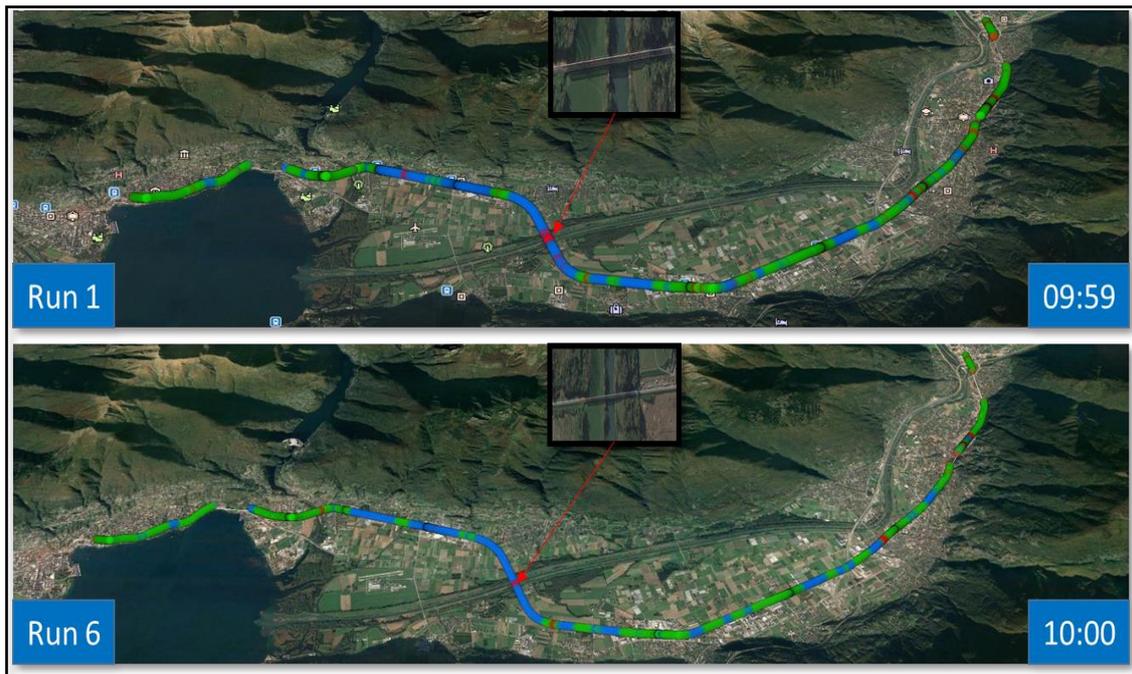


Figure 7. The Google map representing the track characterization for Run 1 (upper panel) from the clustering outcome and when the model is applied to the test samples (Run 6) (lower panel).

CONCLUSION AND PERSPECTIVES

We have used the Expectation-Maximization for Gaussian Mixture Model algorithm to extract hidden multi Gaussian distribution within the data samples as they are often represented by a single overestimated Gaussian curve. As an input, the algorithm requires the number of clusters expected in the data samples. To avoid complexity and to have some diversity, we have intuitively selected 3 clusters to represent the position error samples. The learning data has been formed from the combination of 5 runs collected on the same track in different periods of the day in order to capture several possible events and take into account the variations of the constellation within time. An EM algorithm is used to estimate the GMM parameters by maximizing the log-likelihood function. The models are then applied to the test samples (Run 6) collected 7 days later to characterize the track.

As expected due to the temporal variation of the satellite visibility, the clustering outcome of several runs shows that the class representation on the track varies throughout the day due to variation in the position error. This behavior is very typical for the satellite-based positioning system as the positioning errors have temporal and spatial dependencies. It is also promising to see that for some very particular obstacles like metallic bridge, the classification results captures consistent degradation in the positioning accuracy throughout the day. When the error models (extracted from Run 1-5) are applied to the test samples of Run 6, the overall class representation of the track appears to be identical to the clustering outcome associated with Run 1. It shows that the track characterization does not change much for the 2 runs performed in the same period but on different days considering a shift in the reappearance of the same satellite configuration (for GPS constellation the shift between the 2 consecutive days is nearly 4 mins/day).

This first investigation is promising for track characterization as well as to build error models that will be representative of a railway line. Further investigation will complete this study. First, the effective modeling of the track errors would require an extensive amount of data to achieve ergodicity and stationarity in the models capturing as much as possible the error variations. We combined 5 runs collected on the same day but in a different period which are certainly not enough for the learning model and robust track characterization even though the overall outcome is very promising. Then, some other clustering criteria need to be investigated to reduce the effects of the overlapping clustering problem [5][6][7]. The models need to be applied to more test campaign for the reproducibility of the results.

ACKNOWLEDGMENT

The work presented in this paper benefits from funds of the European JU Shift2Rail in the framework of the Gate4Rail project, and for databases access thanks to the past EU H2020 project STARS.

REFERENCES

1. Cosimo Stallo, Alessandro Neri et al. , Geo-Distributed Simulation and Verification Infrastructure for safe train Galileo-based positioning, Proceedings of the European Navigation Conference 2020.
2. Daniel Gerbeth, Omar García Crespillo, Fabio Pognante, Alessia Vennarini and Andrea Coluccia, Framework to Classify Railway Track Areas According to Local GNSS Threats, Proceedings of the European Navigation Conference 2020.
3. Syed Ali Kazim, Nourdine Aït Tmazirte and Juliette Marais, Realistic position error models for GNSS simulation in railway environments, Proceedings of the European Navigation Conference 2020.
4. Tan, J., Wang, J., & Lu, D. (2019, June). GNSS data driven clustering method for railway environment scenarios classification. In 2019 14th IEEE Conference on Industrial Electronics and Applications (ICIEA) (pp. 2026-2031).
5. Yu, H., & Wang, Y. (2012, August). Three-way decisions method for overlapping clustering. In International Conference on Rough Sets and Current Trends in Computing (pp. 277-286). Springer, Berlin, Heidelberg.
6. Banerjee, A., Krumpelman, C., Ghosh, J., Basu, S., & Mooney, R. J. (2005, August). Model-based overlapping clustering. In Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining (pp. 532-537).
7. Fu, Q., & Banerjee, A. (2008, December). Multiplicative mixture models for overlapping clustering. In 2008 Eighth IEEE International Conference on Data Mining (pp. 791-796).
8. Murphy, K. P. (2012). Machine learning: a probabilistic perspective. MIT press
9. Stamm, B. and Peter Gurník. "STARS project-Satellite technology for advanced railway signalling." (2017).