ORIGINAL STUDY

Gauss process regression for real-time ionospheric delay estimation from GNSS observations

Balazs Lupsic[1] 🄳 · **Bence Takacs**[1]

Abstract

The number of devices equipped with global satellite positioning has exceeded seven billion recently. There are a wide variety of receivers regarding their accuracy and reliability. Low cost, multi-frequency units have been released on the market latterly; however, the number of single-frequency receivers is still significant. Since their measurements are influenced by ionospheric delay, accurate ionosphere models are of utmost importance to reduce the effect. This paper summarizes how Gauss process regression (GPR) can be applied to derive near real-time regional ionosphere models using raw Global Navigation Satellite System (GNSS) observations of permanent stations. While Gauss process is widely used in machine learning, GPR is a nonparametric, Bayesian approach to regression. GPR has several benefits for ionosphere monitoring since it is quite robust and efficient to derive a grid model from data available in irregular set of ionospheric pierce points. The corresponding instrumental delays are estimated by a parallel Kalman filter. The presented algorithm can be applied near real-time, however the results are offline calculated and are compared to two high quality TEC map products. Based on the analysis, the accuracy of the GPR modell is in 2 TECu range. The developed methods could be efficiently applied in the field of autonomous vehicle navigation with meeting both accuracy and integrity requirements.

Keywords GNSS · Ionosphere modelling · Regional TEC map · Gauss process regression

1 Introduction

The ionosphere is the upper part of the Earth's atmosphere with an altitude between about 40 and 2000 km. The ionosphere is ionized by solar radiation which means that there are free electrons and positive ions in this region Schaer (1999). In addition to playing and

---

✉ Balazs Lupsic
  lupsic.balazs@emk.bme.hu

  Bence Takacs
  takacs.bence@emk.bme.hu

[1] Department of Geodesy and Surveying, Faculty of Civil Engineering, Budapest University of Technology and Economics, Budapest, Hungary

important role in the Earth's magnetic field, the ionosphere influences radio signal propagation from satellites to the Earth.

One of the major parts of the error budget in the GNSS (Global Navigation Satellite System) measurements is the ionospheric delay. The effect depends on the number of charged particles, the Total Electron Content (TEC) along the path of the signal through the ionosphere. With multi-frequency devices, the delay can be removed by the so-called ionospheric-free linear combination Hofmann-Wellenhof et al. (2008). However, still major part of the devices operates on a single-frequency; thus the ionosphere modelling is crucial to mitigate its effect.

One of the earliest and simplest models is the Klubochar model Klobuchar (1987). The corresponding parameters are broadcast by the GPS satellites among navigational messages. In general, Klobuchar model can describe just a limited part of the whole effect (Feess and Stephens 1987; Filjar et al. 2009), however in extreme conditions, during ionospheric storms for instance, its accuracy can be degraded even more (Bergeot et al. 2010; Gordienko et al. 2005; Hu et al. 2005). The most recent satellite system, the European Galielo, broadcasts its own ionosphere model, the NeQuick one Angrisano et al. (2013). According to a throughout investigation and comparison of Klobuchar and NeQuick model Farah (2008), it can be concluded in general that NeQuick model offers a better behaviour when the ionosphere is stable but slightly poorer behaviour with higher variability of ionosphere or close to maximum TEC values.

High quality TEC maps can be derived from the measurements of permanent station with post-processing Schaer (1999). The spherical harmonics and B-spline based TEC models are adequate for global and regional representation, while the polynomial based models are limited for regional modelling. These models describe the TEC of the ionosphere as a function of time and space Hernández-Pajares et al. (2009). The commonly accepted exchange format is the IONEX Schaer et al. (1998), that represents the 2- and 3-dimensional TEC maps given in a geographic grid. Thanks to the rapid development of computer processing, providing global and regional tec maps can be done in near-real-time (Coster et al. 1992; Bergeot et al. 2014; Erdogan et al. 2017; Chang et al. 2019; Renga et al. 2018).

The Gauss process regression (GPR) method is well-known in the machine learning community and proved its utility in various domain. It is also worth to mention that the mathematical background of the GPR and popular Kriging in geostatistics are the same Stein (1999). Data from a single station in India was used to forecast ionospheric delay using GPR technique with almost 100% accuracy Lakshmi et al. (2020). Another GPR approach was used to predict daily TEC values based upon the TEC values recorded at various permanent GNSS station in Turkey Inyurt et al. (2020). A comparison of Gaussian process (GP) and neural network model was performed by Ackermann et al. (2011) over a test area in South Africa. The GP framework presented many advantages over competing modeling strategies, such as providing powerful and convenient ways of incorporating prior knowledge and requiring less training data than neural networks. Another recent study shows the GP ability to enhance the positioning performance by improved TEC estimation in real-time Lin et al. (2019).

The safety critical applications demand for high reliability besides accuracy. Auxiliary services are preferred to achieve this high standard of requirements. For the European region, the EGNOS (European Geostationary Navigation Overlay Service) grants real-time corrections from the measurements of RIMS (Ranging and Integrity Monitoring Stations) Ventura-Travest and Flament (2007). The stations are located across the European region in a sparse network (Fig. 1). The very first initiative to monitor the achievable accuracy and
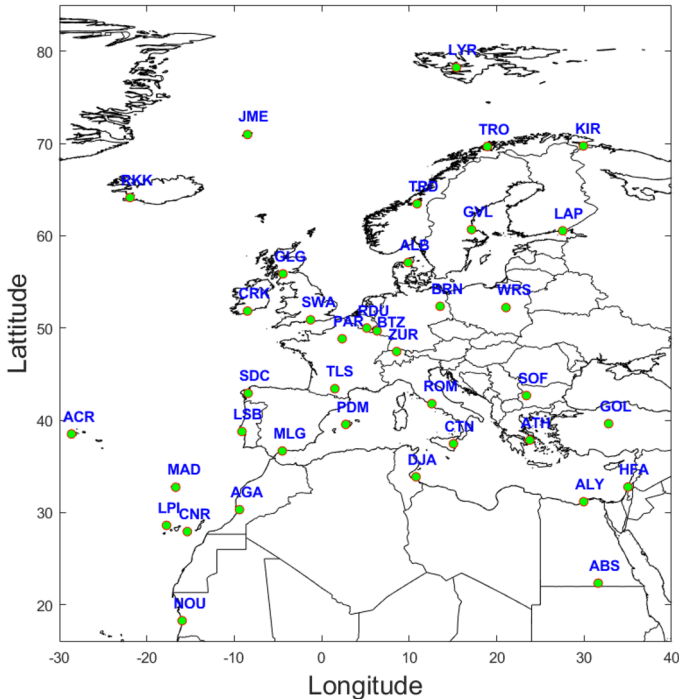
Fig. 1 RIMS positions across the European region

integrity with EGNOS was performed by the EGNOS Data Collection Network (EDCN) project Soley et al. (2004). EGNOS performance can be slightly degraded at the boundary of coverage area due to lack of reliable ionosphere modelling over the periphery Grunwald et al. (2016) or Lupsic and Takács (2019).

This paper presents a GPR based novel ionospheric model for real-time TEC map generation from RIMS data available in the European region. The accuracy of the developed model is investigated by comparing to widely used and referred global and regional models developed using raw GNSS observations.

1.1 Algorithm outline

A brief overview about the developed algorithm is presented in Fig. 2 and the mathematical details of each block will be featured in the following sections. A regional receiver network, in our case the EGNOS system provides code and phase measurements in double frequencies. Combined measurements cancel out the geometry dependency and the result directly reflects the state of the ionosphere and the combined instrumental delays of receiver and the tracked satellite. This is the so called L_4 combination and the rigorous mathematical description of its calculation is presented in Sect. 2.1. In case of known instrumental delays, the slant ionospheric delays from the L_4 observation can be calculated. A single layer model determines the conversion from the slant ionospheric delay to vertical ionospheric delay alongside the location of the ionospheric pierce point (IPP) [Sec. 2.1]. The vertical ionospheric delay with their corresponding locations are the input to the
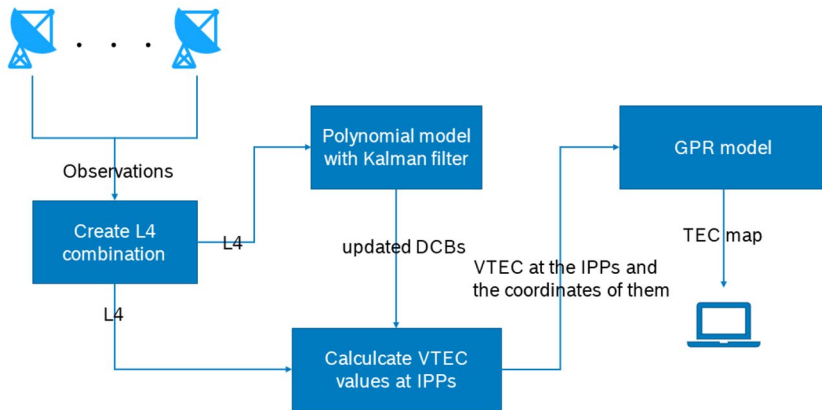
Fig. 2 Algorithm outline

Gauss process regression model which is described in Sect. 2.2. The GPR model creates the Total Electron Content map and updates it in real time. The presented algorithm in the paper is designed to be capable of run in real time, therefore the instrumental delays shall be estimated, and monitored continuously in parallel with the GPR. For that purpose, a polynomial model was used to describe the vertical total electron content (VTEC) and the data was processed iterative with a Kalman filter. The Sect. 2.4 is dedicated to present this part of the algorithm. It is worth to mention any other near-real time estimation could be used which provides an adequate accuracy for instrumental delays.

2 Mathematical background

2.1 Ionospheric TEC measurements from GNSS

The ionospheric refraction is frequency-dependent, therefore with a dual-frequency receiver this delay can be eliminated in first order by the ionospheric-free combination. On the other hand, with the geometry-free combination, one could extract information about the state of the ionosphere Ciraolo et al. (2007). The combination can be created from the pseudorange measurements and the phase measurements. The noise of phase observation is typically two or three orders smaller than the code but suffers from the initial ambiguity. This ambiguity can be resolved by quasi ionosphere-free (QIF) (Teunissen 1995; Zhang et al. 2016).

The ionospheric delay depends on the frequency and the total electron content along the signal path between satellite and receiver.

$$STEC = \int_r^s N_e(l)dl, \tag{1}$$

where N_e is the electron density in el/m^2 dimension, which varies along the ray path. The vertical total electron content (VTEC) definition follows the form of Eq. (1) but the integration path is vertical.

$$I_{r,j}^s = \frac{40.3 \cdot STEC}{f_j^2} = \frac{f_i^2}{f_j^2} I_{r,i}^s \tag{2}$$

Converting the ionospheric delay from one frequency to another can be easily done with the Eq. (2), where j is the target and i is the reference frequency. $I_{r,i}^s$ is the delay on the reference signal. The code and phase observation can be divided to frequency dependent and independent parts.

$$R_{r,j}^s = \rho_r^s + c(dt_r - dt^s) + T_r^s + I_{r,j}^s + \xi_{r,j}^s + d_{r,j} + d_j^s + \varepsilon_{r,j}^s \tag{3}$$

$$\Phi_{r,j}^s = \rho_r^s + c(dt_r - dt^s) + T_r^s - I_{r,j}^s + \zeta_{r,j}^s + \delta_{r,j} + \delta_j^s + \lambda_j N_{r,j}^s + \epsilon_{r,j}^s, \tag{4}$$

where j indicates the carrier frequency, $R_{r,j}^s, \Phi_{r,j}^s$ code and phase measurement respectively, ρ is the real geometry distance between the r receiver and s satellite, dt_r is the receiver clock bias, dt^s stands for the satellite clock bias, T_r^s is the tropospheric delay, $I_{r,j}^s$ the ionospheric delay in j frequency, $d_{r,j}, \delta_{r,j}$ are the hardware bias of the receiver in j frequency of the code and phase observation respectively, d_j^s, δ_j^s are for the hardware bias of the satellite in j frequency of the code and phase measurement respectively, $\varepsilon_{r,j}^s, \epsilon_{r,j}^s$ denote stochastic, Gaussian type noises, $N_{r,j}$ is the integer phase ambiguity, λ_j the wavelength of j frequency, $\xi_{r,j}^s, \zeta_{r,j}^s$ denote the multipath. Subtracting the code and phase measurement in frequnecy 1, 2 yields a geometry-free combination.

$$R_{r,I}^s = R_{r,1}^s - R_{r,2}^s = \alpha I_{r,1}^s + b_r + b^s + M_{r,I}^s + \varepsilon_{r,P}^s \tag{5}$$

$$\alpha = \left(1 - \frac{f_1^2}{f_1^2}\right),$$
$$b_r = d_{r,1} - d_{r,2},$$
$$b^s = d_1^s - d_2^s,$$
$$\varepsilon_{r,P}^s = \varepsilon_{r,1}^s - \varepsilon_{r,2}^s,$$
$$M_{r,P}^s = \xi_{r,j}^s - xi_{r,j}^s,$$

where α is a frequency dependent constant, b_r, b^s are the differential code biases, $\varepsilon_{r,P}^s, \xi_{r,P}^s$ are the combined noise, and multipath delay respectively. The same combination can be created from the phase measurement as well,

$$\Phi_{r,I}^s = \Phi_{r,2}^s - \Phi_{r,1}^s = \alpha I_{r,1}^s + B_r + B^s + M_{r,\Phi}^s + C_{arc,r}^s + \varepsilon_{r,\Phi}^s, \tag{6}$$

$$C_{arc,r}^s = \lambda_2 N_{r,2}^s - \lambda_1 N_{r,1}^s$$
$$B_r = \delta_{r,2} - \delta_{r,1},$$
$$B^s = \delta_2^s - \delta_1^s,$$
$$\epsilon_{r,\Phi}^s = \epsilon_{r,2}^s - \epsilon_{r,1}^s,$$
$$M_{r,\Phi}^s = \zeta_{r,2}^s - \zeta_{r,1}^s,$$

where $C_{arc,r}^s$ is the ambiguity bias, B_r, B^s are the receiver and satellite interfrequency bias (ISB), $M_{r,\Phi}^s, \epsilon_{r,\Phi}^s$ are the combined multipath delay and noise. The significantly lower noise

level of the phase observation can be exploited in the L_4 combination. The CPB_r^s (Carrier Phase Bias) is the offset between the $\Phi_{r,I}^s$ and $R_{r,I}^s$ combination and it can be estimated averaging out the geomatry-free code and phase differences in a cycle slip free continuous arc.

$$CPB_r^s \approx \frac{1}{N} \sum_{n=1}^{N} (\Phi_{r,I}^s - R_{r,I}^s)_n, \tag{7}$$

where N is the number of observation in the continuously observed arc. The leveled geometry-free combination, $L_{r,4}^s$ for a given arc is

$$L_{r,4}^s = \Phi_{r,I}^s - CPB_r^s = \alpha STEC + b_r + b^s + \epsilon_{L_4} \tag{8}$$

To achieve a near real-time ionosphere estimation the levelling method is not quite suitable Xiang et al. (2017), instead of that a Hatch-filter smoothing was applied. The smoothed geometry-free code $\hat{R}_{r,I}^s(n)$ can be calculated as:

$$\hat{R}_{r,I}^s(k) = \frac{1}{n} R_{r,I}^s(k) + \frac{n-1}{n} \left[\hat{R}_{r,I}^s(k-1) + \Phi_{r,I}^s(k) - \Phi_{r,I}^s(k-1) \right], \tag{9}$$

where, $n = k$ when $k < N$ and $n = N$ when $k \geq N$. The N value was chosen to 50 in this paper, but to avoid the nonconvergent smoothed code $\hat{R}_{r,I}^s(n)$ create a significant bias to the estimation, the first 10 samples in each new arc were not used for TEC modelling. A cycle slip detection is substantial to avoid abrupt biases in L_4 smoothing and in case of this event the Hatch-filter algorithm resets.

Once the L4 is calculated and smoothed, one can create a direct link with corresponding STEC value and the satellite and receiver biases.

$$L_{r,4}^s(t) = \alpha STEC(t) + b_r(t) + b^s(t) + \epsilon_{L_4}, \tag{10}$$

where the α, b_r, b^s represents the same terms from Eq. (5) and ϵ_{L_4} is the noise of the smoothed $L_{r,4}^s$ measurement. The vertical total electron content is one of the main properties of the ionosphere. One of the most used, and simplest method to establish connection between the slant and vertical TEC is a single layer model (SLM).

$$STEC(t) = m(z)VTEC(\lambda, \phi, t), \tag{11}$$

where $STEC$ is the total electron content between a receiver and a satellite, $m(z)$ is the elevation dependent mapping function. Fig. 3. depicts the concept of SLM, where the the vertical dimension of ionosphere is reduced to a single layer. The slant total electron content is considered at the ionospheric point, where the line of sight crosses the single layer. The $m(z)$ mapping function converts the STEC to VTEC value therefore the different line of sight measurements are comparable. The latitude and longitude of the sub-ionospheric point are the characteristic properties of the IPPs alongside with its VTEC value. Substituting Eq. (11) to Eq. (10) yields the following:

$$L_{r,4}^s(t) = \alpha m(z)VTEC(\lambda, \phi, t) + b_r(t) + b^s(t) + \epsilon_{L_4}, \tag{12}$$

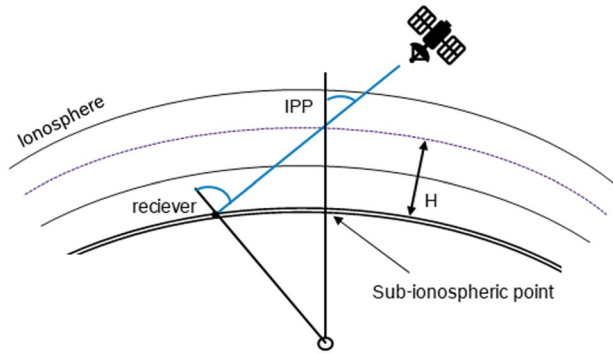In case of known b_r and b^s the VTEC value can be calculated by rearranging the Eq. (12).

Fig. 3 Single layer ionospheric model, SLM

$$VTEC(\lambda, \phi, t) = \frac{L_{r,4}^s(t) - b^s(t) - b_r(t) - \epsilon_{L_4}}{\alpha m(z)} \qquad (13)$$

The latitude and longitude of the IPP can be calculated from the single layer model and from the known position of the receiver and satellite. These are the input set to the GPR modell, and in case of RIMS stations 200–400 separate IPPs with their VTEC values can be calculated in each epoch (Fig. 4). The next section will present the details how one can create a consistent TEC map from the fore-mentioned snapshot data with Gauss-process regression.
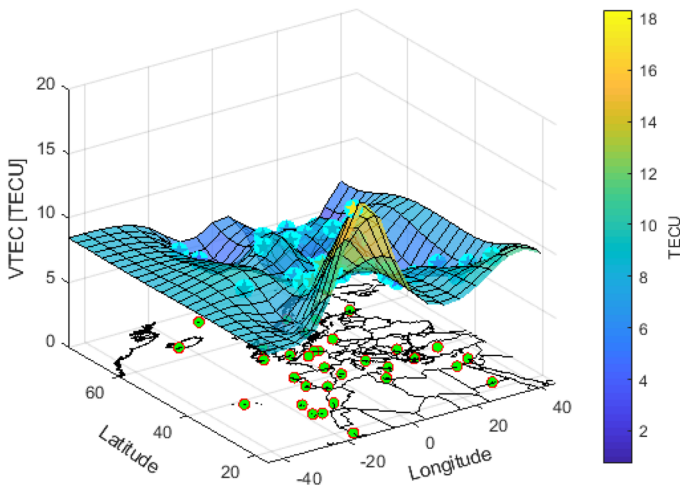
Fig. 4 VTEC map created by GPR model. Green circles represents the positions of the RIMS stations, and the spherical stars are the positions of the ionospheric pierce points. Their colors and heights depict the corresponding VTEC values derived from the $L4$ observations and from the known DCBs values according to Eq. (13). The stars represent a snapshot of input value set to the GPR model, and the grid surface is the VTEC output of GPR modell

2.2 Gauss process regression

The central understanding of Gaussian process lays in the multivariate normal distribution. A random vector $\boldsymbol{y} = (y_1, \ldots, y_l)$ has the multivariate normal distirbution if it is derived from the following transformation of $\boldsymbol{z} = (z_1, \ldots, z_M)$, where each $z_m \sim \mathcal{N}(0, 1)$ Blitzstein and Hwang (2019):

$$\boldsymbol{y} = \boldsymbol{A}\boldsymbol{z} + \boldsymbol{\mu}, \tag{14}$$

where $\boldsymbol{A}$ is a $K \times M$ matrix and $\boldsymbol{\mu} \in \mathbb{R}^K$. This represented as $\boldsymbol{y} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, where $\boldsymbol{\Sigma} = Cov(\boldsymbol{y}, \boldsymbol{y}) = \boldsymbol{A}\boldsymbol{A}^T$. The mean vector represented by $\boldsymbol{\mu}$, and the covariance matrix is the $\boldsymbol{\Sigma}$. Therefore the probabilistic functions of $\boldsymbol{y}$ is

$$f(\boldsymbol{y}) = \frac{1}{(2\pi)^K |\boldsymbol{\Sigma}|^{\frac{1}{2}}} exp\left(-\frac{1}{2}(\boldsymbol{y} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1}(\boldsymbol{y} - \boldsymbol{\mu})\right) \tag{15}$$

Consider splitting a multivariate normal variable $\boldsymbol{y}$ into $(\boldsymbol{y}_1, \boldsymbol{y}_2)$ two sub-vectors. Each of it has multivariate normal distribution hence we can split the $\boldsymbol{A}$ matrix above accordingly. Rewriting the $\boldsymbol{y}$ vector in terms of its sub-vectors gives

$$\boldsymbol{y} = \begin{pmatrix} \boldsymbol{y}_1 \\ \boldsymbol{y}_2 \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} \boldsymbol{\mu}_1 \\ \boldsymbol{\mu}_2 \end{pmatrix}, \begin{pmatrix} \boldsymbol{\Sigma}_{11} & \boldsymbol{\Sigma}_{12} \\ \boldsymbol{\Sigma}_{21} & \boldsymbol{\Sigma}_{22} \end{pmatrix}\right). \tag{16}$$

The conditional distribution of $\boldsymbol{y}_2$ given $\boldsymbol{y}_1$ is $P(\boldsymbol{y}_1|\boldsymbol{y}_2) = \mathcal{N}(\boldsymbol{\mu}_{1|2}, \boldsymbol{\mu}_{1|2})$, where

$$\boldsymbol{\mu}_{1|2} = \boldsymbol{\mu}_2 + \boldsymbol{\Sigma}_{21} \boldsymbol{\Sigma}_{11}^{-1}(\boldsymbol{y}_1 - \boldsymbol{\mu}_1), \tag{17}$$

$$\boldsymbol{\Sigma}_{1|2} = \boldsymbol{\Sigma}_{22} - \boldsymbol{\Sigma}_{21} \boldsymbol{\Sigma}_{11}^{-1} \boldsymbol{\Sigma}_{12}. \tag{18}$$

Gaussian process is a collection of random variables, any finite number of which have a joint Gaussian distribution. Formally, an $\boldsymbol{f}$ function is a Gaussian process if any finite set of values $f(\boldsymbol{x}_1), \ldots f(\boldsymbol{x}_N)$ has a multivariate normal distribution on any S domain, $\boldsymbol{x}_1, \ldots \boldsymbol{x}_N \in S$. GP is specifed by a mean function $m(\boldsymbol{x})$ and a covariance function $k(\boldsymbol{x}, \boldsymbol{x}')$, which is denoted as $f \sim \mathcal{GP}(m(\boldsymbol{x}), k(\boldsymbol{x}, \boldsymbol{x}'))$. The mean and covariance of f GP for any $\boldsymbol{x}, \boldsymbol{x}'$ are $m(\boldsymbol{x}) = E[f(\boldsymbol{x})]$ and $k(\boldsymbol{x}, \boldsymbol{x}') = Cov(f(\boldsymbol{x}), f(\boldsymbol{x}'))$. The goal is to estimate function values of GP conditioned on some training data. Denote the set of inputs as $X \in \mathbb{R}^{N \times D}$, and the corresponding function values as $\boldsymbol{f} \in \mathbb{R}^N$, where D is the domain dimension. In the simplest case, the mean function is assumed as constant zero, the $\boldsymbol{f}$ values are noise-free. The $k(\boldsymbol{x}, \boldsymbol{x}')$ covariance function is driven by its $\boldsymbol{\theta}$ hyperparameters, or kernel parameters. One of the most popular kernel function is the *squared exponential* given by the following formula:

$$k(\boldsymbol{x}, \boldsymbol{x}') = \sigma^2 \exp\left(-\frac{(\boldsymbol{x} - \boldsymbol{x}')}{2l^2}\right), \tag{19}$$

where σ_f^2 and l are the hyperparameters, denoted by $\boldsymbol{\theta}$ above. To estimate the function values $\boldsymbol{f}_*$ for a new set of inputs X_* similarly to Eq. (16) assume

$$\begin{pmatrix} \boldsymbol{f} \\ \boldsymbol{f}_* \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} \boldsymbol{0} \\ \boldsymbol{0} \end{pmatrix}, \begin{pmatrix} K_{XX} & K_{XX_*} \\ K_{X_*X} & K_{X_*X_*} \end{pmatrix}\right). \tag{20}$$

The $K_{XX}, K_{XX_*}, K_{X_*X}, K_{X_*X_*}$ covariance matrices are constructed from the kernel function. The conditional distribution of f_* can be calculated from the conditional properties of multivariate normal.

$$P(f_*|X_*, X, f, \theta) = \mathcal{N}(K_{X_*X}K_{XX}^{-1}f, K_{X_*X_*}K_{XX}^{-1}K_{XX_*}) \tag{21}$$

This derived formula is called the *predictive distribution* and it models the distribution of an unobserved set of inputs. The function values can be estimated by taking the mean, and furthermore the covariance structure of the estimated values are generally known. In practice the function values are not accessed directly but can be observed with some noise.

$$y_n = f(x_n) + \epsilon_n, \tag{22}$$

where $\epsilon_n \sim \mathcal{N}(0, \sigma^2)$. The observation noise can be incorporated into the kernel function by adding σ^2 to every diagonal term in each covariance matrix. The new kernel takes the form of

$$k'(x, x') = k(x, x') + \sigma_n^2 I(x = x'), \tag{23}$$

where $I(\cdot)$ represents the indicator function. The predicted distribution of the unobserved y_* values of the X_* set of input can be derived similarly as Eq. (21).

$$P(y_*|X_*, X, y, \theta) = \mathcal{N}(K_{X_*X}K_{XX}^{-1}y, K_{X_*X_*}K_{XX}^{-1}K_{XX_*}) \tag{24}$$

2.3 TEC map estimation with Gauss process regression

In the previous section the model was considered with a zero mean function, however, in many cases a trend function defines the relation between the input and output data. Consider a training sata set $\{(x_i, y_i; i = 1, 2, \ldots, n)\}$, with a following linear connection:

$$y = h(x)^T \beta + \varepsilon, \tag{25}$$

where $x_i \in \mathbb{R}^d$ and $y_i \in \mathbb{R}$, $\varepsilon \sim \mathcal{N}(0, \sigma^2)$. The $h(x)$ is a chosen base function and the β coefficients and the σ standard deviation are estimated from the training data set. In case of ionosphere modelling, the $x_i \equiv \{\phi_i, \lambda_i\}, y_i \equiv VTEC_i; i = 1, 2, \ldots, n$, where n is the number of observaion used for the TEC map construction, ϕ_i, λ_i are the latitude and longitude of the IPP, and the $VTEC_i$ is the corresponding vertical electron content derived from the observations and the single-layer mapping model Dach et al. (2007). The $h(x)$ function can be polynomial or spherical harmonics which established their viability in TEC modelling (Komjathy et al. 2005; Sun et al. 2020), and one of its variant is used for DCBs estimation in Eq. (44). Instead of solving Eq. (25) for β, a location dependent GP will be introduced. The ε noise parameter is implicitly in the kernel like in Eq. (23). Consider the following model, where the definition of x, y are the same as in Eq. (25).

$$y = h(x)^T \beta + f(x), \tag{26}$$

where $f(x)$ is a zero mean GP with covariance function $k(x, x')$, $f(x) \sim GP(0, k(x, x'))$. The basis functions $h(x)$ transforms the original x vector to a $\mathbb{R}^p$ feature space. The β represents the coefficients of the basis vector with its p-by-1 dimension. The GPR model for an instance of response y is the following:

$$P(y_i|f(\boldsymbol{x}_i), \boldsymbol{x}_i) \sim \mathcal{N}(y_i|h(\boldsymbol{x}_i)^T \boldsymbol{\beta} + f(\boldsymbol{x}_i), \sigma^2) \tag{27}$$

The GPR model is nonparametric, hence it introduces an $f(\boldsymbol{x}_i)$ latent variable for each $\boldsymbol{x}_i$ observation. The vector form of the model is

$$P(\boldsymbol{y}|\boldsymbol{f}, \boldsymbol{X}) \sim \mathcal{N}(\boldsymbol{y}|\boldsymbol{H}\boldsymbol{\beta} + \boldsymbol{f}, \sigma^2 \boldsymbol{I}), \tag{28}$$

where

$$\boldsymbol{X} = \begin{pmatrix} \boldsymbol{x}_1^T \\ \boldsymbol{x}_2^T \\ \vdots \\ \boldsymbol{x}_n^T \end{pmatrix}, \boldsymbol{y} = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix}, \boldsymbol{H} = \begin{pmatrix} h(\boldsymbol{x}_1^T) \\ h(\boldsymbol{x}_2^T) \\ \vdots \\ h(\boldsymbol{x}_n^T) \end{pmatrix}, \boldsymbol{f} = \begin{pmatrix} f(\boldsymbol{x}_1) \\ f(\boldsymbol{x}_2) \\ \vdots \\ f(\boldsymbol{x}_n) \end{pmatrix}. \tag{29}$$

The basis matrix $\boldsymbol{H}$ has a dimension of $n \times p$, where n is the number of observations, or training data, and p is the dimension of feature space. In case of the missing $\boldsymbol{H}$ matrix, the form converts back to the Eq. (22). The simplest choice for basis function is constant, so the feature space p would be one dimension, so the $\boldsymbol{H} = \boldsymbol{1}$ basis matrix is an $n \times 1$ vector of ones. In this paper the constant base matrix was used. Worth to mention the linear and quadratic form, where $\boldsymbol{H}_{lin} = [\boldsymbol{1}, \boldsymbol{X}]$ and $\boldsymbol{H}_{quad} = [\boldsymbol{1}, \boldsymbol{X}, \boldsymbol{X}_2]$ respectively. Extracted form of the matrices in details in case of ionosphere modelling are:

$$\boldsymbol{H}_{lin} = [\boldsymbol{1}, \boldsymbol{X}] = \begin{pmatrix} 1 & \phi_1 & \lambda_1 \\ 1 & \phi_2 & \lambda_2 \\ \vdots & \vdots & \vdots \\ 1 & \phi_n & \lambda_n \end{pmatrix},$$

$$H_{quad} = [\boldsymbol{1}, \boldsymbol{X}, \boldsymbol{X}_2] = \begin{pmatrix} 1 & \phi_1 & \lambda_1 & \phi_1^2 & \lambda_1^2 \\ 1 & \phi_2 & \lambda_2 & \phi_2^2 & \lambda_2^2 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & \phi_n & \lambda_n & \phi_n^2 & \lambda_n^2 \end{pmatrix}.$$

The $\boldsymbol{f}$ latent variable in the Eqs. (29) and (28) has a joint distribution by definition, and it follows the form of

$$P(\boldsymbol{f}|\boldsymbol{X}) \sim \mathcal{N}(\boldsymbol{f}|\boldsymbol{0}, K(\boldsymbol{X}, \boldsymbol{X})), \tag{30}$$

where the $K(\boldsymbol{X}, \boldsymbol{X})$ has the form of

$$K(\boldsymbol{X}, \boldsymbol{X}) = \begin{pmatrix} k(\boldsymbol{x}_1, \boldsymbol{x}_1) & k(\boldsymbol{x}_1, \boldsymbol{x}_2) & \dots & k(\boldsymbol{x}_1, \boldsymbol{x}_n) \\ k(\boldsymbol{x}_2, \boldsymbol{x}_1) & k(\boldsymbol{x}_2, \boldsymbol{x}_2) & \dots & k(\boldsymbol{x}_2, \boldsymbol{x}_n) \\ \vdots & \vdots & \vdots & \vdots \\ k(\boldsymbol{x}_n, \boldsymbol{x}_1) & k(\boldsymbol{x}_n, \boldsymbol{x}_2) & \dots & k(\boldsymbol{x}_n, \boldsymbol{x}_n) \end{pmatrix} \tag{31}$$

The used $k(\boldsymbol{x}, \boldsymbol{x}')$ covariance function in this paper is the Matern52 kernel defined as

$$k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \sigma_f^2 \left(1 + \frac{\sqrt{5}r}{\sigma_l} + \frac{\sqrt{5}r^2}{3\sigma_l^2} \right) exp \left(-\frac{\sqrt{5}r}{\sigma_l} \right), \tag{32}$$

where r is the Euclidean distance between $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$. The $\boldsymbol{\theta}$ hyperparameters in the above mentioned kernel are σ_f and σ_l. The $k(\boldsymbol{x}_i, \boldsymbol{x}_j|\boldsymbol{\theta})$ form highlights the dependency of the $k(\boldsymbol{x}_i, \boldsymbol{x}_j)$ kernel from the $\boldsymbol{\theta}$ hyperparameters. To create a TEC map, first the $\boldsymbol{\beta}$ basis function

coefficients, the σ^2 noise variance and the kernel's θ hyperparamaters shall be estimated from the given (X, y) training data set. The used approach is the $P(y|X)$ likelihood maximization as a function of β, θ and σ^2. The estimation is conducted by maximization of the marginal log likelihood function:

$$\hat{\beta}, \hat{\theta}, \hat{\sigma}^2 = \arg\max_{\beta, \theta, \sigma^2} \log P(y|X, \beta, \theta, \sigma^2). \tag{33}$$

The marginal log likelihood function has a form of:

$$\log P(y|X, \beta, \theta, \sigma^2) = -\frac{1}{2}(y - H\beta)^T M^{-1}(y - H\beta) - \frac{n}{2} \log 2\pi - \frac{1}{2} log|M|, \tag{34}$$

where

$$M = K(X, X|\theta) + \sigma^2 I. \tag{35}$$

First maximize the log likelihood function with respect to β for given θ and σ^2. The $\hat{\beta}(\theta, \sigma^2)$ estimation is

$$\hat{\beta}(\theta, \sigma^2) = [H^T M^{-1} H]^{-1} H^T M^{-1} y. \tag{36}$$

Substituting Eq. (36) to Eq. (34) yields a β-profiled log likelihood.

$$\log P(y|X, \theta, \sigma^2) = -\frac{1}{2}(y - H\hat{\beta})^T M^{-1}(y - H\hat{\beta}) - \frac{n}{2} \log 2\pi - \frac{1}{2} log|M| \tag{37}$$

Equation (37) is maximized over θ and σ^2 to find their estimates, then back substitute the optimized θ and σ^2 to the Eq. (36) to fix the β coefficients. The last step is to make prediction to the new y_g TEC values in the given X_g data set, to create a TEC map. The observation noise free prediction follows as

$$P(y_g|y, X, X_g) = \mathcal{N}(y_g|H_{X_g}\beta + \mu, \Sigma), \tag{38}$$

where

$$\Sigma = K(X_g, X_g) - K(X_g, X)(K(X, X) + \sigma^2 I)^{-1} K(X, X_g), \tag{39}$$

and y_g, X_g, H_g have similar from as in Eq. (29) but with the new $X_g = (x_1^g, x_2^g, \dots, x_n^g)$ consists the grid coordinates and y_g vector contains the corresponding VTEC values. The expected TEC values of the created TEC map can be derived from Eq. (38) and found as

$$E(y_g|y, X, X_g), \beta, \theta, \sigma^2) = (H_{X_g}\beta + K(X, X_g|\theta)\alpha, \tag{40}$$

where

$$\alpha = (K(X, X|\theta) + \sigma^2 I)^{-1}(y - H\beta). \tag{41}$$

2.4 Differential code bias estimation with Kalman filter

Kalman filter is a common choice for real-time estimation. Consider extending the Eq. (12) where the following $m(z)$ mapping function of the used single layer model Dach et al. (2007):

$$m(z) = \left(1 - \left(\frac{R_e}{R_e + H}sin(\alpha_m z)\right)^2\right)^{-\frac{1}{2}}, \tag{42}$$

with single-layer height $H = 506.7$ km, Earth radius $R_e = 6371$ km, and $\alpha_m = 0.9782$. The *VTEC* is estimated with second degree polynomials in λ latitude and ϕ longitude. The following equation can be derived from the general form of Eq. (25), with polynomials as $h(x)$ base function and β as the adjustable polynomial coefficients.

$$VTEC(\lambda, \phi) = \sum_{j=0}^{5} \omega^j(\lambda, \phi) VTEC^j(\lambda, \phi | \lambda_0^j, \phi_0^j), \tag{43}$$

where

$$
\begin{aligned}
VTEC^j(\lambda, \phi | \lambda_0^j, \phi_0^j) &= a_0^j + a_1^j(\lambda_0^j - \lambda) + a_2^j(\phi_0^j - \phi) \\
&+ a_3^j(\lambda_0^j - \lambda)^2 + a_4^j(\phi_0^j - \phi)^2 \\
&+ a_5^j(\lambda_0^j - \lambda)(\phi_0^j - \phi)
\end{aligned} \tag{44}
$$

The a_n^j, $n, j \in 0, \dots 5$ are the coefficients estimated by the Kalman filter are the same declared as β in Eq. (25). The λ_0^j, ϕ_0^j, $j \in 0, \dots 5$ are the center of the polynomials. The $\omega(\lambda, \phi)$ is a weight factor. The inverse distance of the IPP and the center of polynomial is a common choice for the weight factor but in this paper the closest polynomial is the only contributor.

$$\omega^j(\lambda, \phi) = \begin{cases} 1, & \text{if } d^j \equiv \min_{n \in 0 \dots 5} d^n \\ 0, & \text{otherwise}, \end{cases} \tag{45}$$

where $d^j = \left\| (\lambda, \phi) - (\lambda_0^j, \phi_0^j) \right\|$ is the Euclidean distance of the (λ, ϕ) IPP coordinates and the origo of the *j*-th polynomial.

The first four polynomials cover the European region. The number and position of them are arbitrary and summarized in Table 1. During the model alignment, it was found that even one polynomial is able to follow the main trend of the ionosphere which is suffice for DCB estimation, however usage of four polynomials has gained better and more consistent results. The number of six, and nine were also tested, but due to the non-uniform spatial distribution of IPPs, some polynomials suffered the lack of nearby measurement points and struggled to converge, causing biases in DCBs. The

Table 1 Center coordinate of polynomials

Polynomials	Coordinates of centers	
	Latitude [°]	Longitude [°]
0.	60	0
1.	60	30
2.	30	30
3.	30	0
4.	− 25.88	27.7
5.	46.07	− 64.47

4th and 5th polynomial are dedicated to HBK and MON station respectively. They are located in North-America and South-Africa, therefore their observations cannot contribute to the TEC calculation of the European region but served well for DCB estimation of the satellites.

The Kalman filter has a prediction step and update step. The estimated x state vector contains the a_n^i polynomial coefficients and the DCBs of satellites and permanent stations. The equations follow as

$$\begin{aligned} x_k &= F_k x_{k-1} + \omega_{k-1} \\ y_k &= D_k x_k + e_k, \end{aligned}$$

(46)

where F_k is the transition matrix, which in this model corresponds to a simple identity matrix. The y_k is the observation vector, which contains the L_4 combinations. The D_k design matrix transforms the x_k unknown parameters from the state space to the observation space according to Eqs. (12) and (43).

The ω_k is the process noise vector and e_k is the measurement noise vector. Both of them are assumed as a white noise with zero expected values and the two vector mutually independent of each other.

$$\begin{aligned} E[\omega_k] &= 0; E[\omega_k, \omega_l^T] = \delta_{k,l} \Sigma_\omega \\ E[e_k] &= 0; E[e_k, e_l^T] = \delta_{k,l} \Sigma_y \\ E[\omega_k, e_l^T] &= 0, \end{aligned}$$

(47)

where $\delta_{k,l}$ is the Kronecker delta. The covariance matrix of the process noise Σ_ω consist three different sub-blocks, Σ_ω^{poly} for the polynomial coefficients, Σ_ω^{sv} for satllites' DCBs and Σ_ω^{stat} for stations' DCBs.

$$\Sigma_\omega = \begin{pmatrix} \Sigma_\omega^{poly} & 0 & 0 \\ 0 & \Sigma_\omega^{sv} & 0 \\ 0 & 0 & \Sigma_\omega^{stat} \end{pmatrix},$$

(48)

where Σ_ω^{sv} and Σ_ω^{stat} have σ_{sv}^2 and σ_{stat}^2 in diagonal and zeros in off-diagonal. The Σ_ω^{poly} has a form of:

$$\Sigma_\omega^{poly} = \begin{pmatrix} \Sigma_\omega^{poly_0} & & 0 \\ & \ddots & \\ 0 & & \Sigma_\omega^{poly_5} \end{pmatrix},$$

(49)

where $\Sigma_\omega^{poly_i} \in i = [0, \dots, 5]$ have only diagonal elements but differ in respect of coefficients. The GNSS observation was considered An elevation dependent wighting scheme was applied on the Σ_y observation covariance matrix. The non-diagonal elements were set to zero, and the σ_y^i diagonal variances were calculated by the following formula (adapted from Wang et al. 2015):

$$\sigma_y^i = \sigma^2 (1 + sin^2(z_i)),$$

(50)

where z_i is the zenith angle corresponding to the ith IPP, and the σ^2 comes from the likelihood maximization presented in Eq. (33).

3 Results

3.1 Concept validation with simulation

This section presents the GPR's capabilities for TEC map generation from the available VTEC values at the corresponding coordinates of IPPs. The assessment consists a simulation of the VTEC values in the IPPs for a given epoch from a known reference map. The IPPs position have been calculated from the postion of RIMS and GPS satellites with an ionospheric layer height 350 km. The used reference map is the product of Royal Observatory of Belgium (ROB) Bergeot et al. ([2014](#)) at date of 2020-01-23. The published ROB map has a 15 minutes time update and $0.5° \times 0.5°$ spatial resolution in lattidue and longitude. The VTEC values come from the ROB product, then a zero mean Gaussion noise scaled by the obliquity factor is added to the simulated data. As first step, the L_4^s combination was calculated from the reference ROB map.

$$L_4^s = \alpha_f m(z_{ipp}) VTEC(\lambda_{ipp}, \phi_{ipp})^R + b_r + b^s + \varepsilon_{L_4}^s, \tag{51}$$

where L_4^s is the simulated measurement, $\varepsilon_{L_4}^s \sim \mathcal{N}(0, \sigma_s^2)$ is a zero mean Gaussian noise. In the second step, the *VTEC* values are calculated from the L_4^s for a given epoch. The $VTEC(\lambda_{ipp}, \phi_{ipp})^s$ comes from the Eq. ([51](#)) as

$$VTEC(\lambda_{ipp}, \phi_{ipp})^s = \frac{L_4^s - b_r - b^s}{\alpha_f m(z_{ipp})} \tag{52}$$

The DCBs are considered as known values therefore the $VTEC^s$ distribution comes only from L_4^s, scaled by the invers of the obliquity factor.

$$VTEC(\lambda_{ipp}, \phi_{ipp})^s \sim \mathcal{N}\left(VTEC(\lambda_{ipp}, \phi_{ipp})^R, \frac{\sigma_s^2}{m(z_{ipp})^2} \right) \tag{53}$$

The $VTEC^s$ values with the corresponding IPP coordinates are collected for a given epoch and feeded to the GPR model to estimate a TEC map at matching grid points of the ROB map. A day long data set was used, therefore 24×4 TEC map were estimated with $2.5° \times 2.5°$ spatial resolution, altogether more then 20000 points.

After differentiation with the reference VTEC values, the discrepancies have been evauled. The standard deviation of the added noise varied from 0 to 10 TECu. In noise free case, an almost perfect match is expected with the reference map. Besides the GPR, two additional methods were used for interpolation to give more perspective to the results. The assisting methods are the following: radial base function (RBF) with multiquadratic function, and third degree polynomial fit (POLY), adapted from (Yilmaz et al. [2009](#); Yu et al. [2015](#)). The d_n^k denotes the differenced VTEC value, where k is the time index and n is the grid point index, computed as:

$$d_n^k(m) = VTEC_n^{ref}(t_k) - VTEC_n^m(t_k) \tag{54}$$

where $m \in [GPR, RBF, POLY]$ indicates the type of method.

The $d_n^k(GPR)$, $d_n^k(RBF)$ and the $d_n^k(POLY)$ values were calculated with $\sigma_s = 0, 1, 2, 3, 4, 6, 8, 10$ [*TECu*] added noise. In Fig. [5](#), the two-dimensional histograms show the d_n^k values in respect of the reference values. Result of the three interpolation
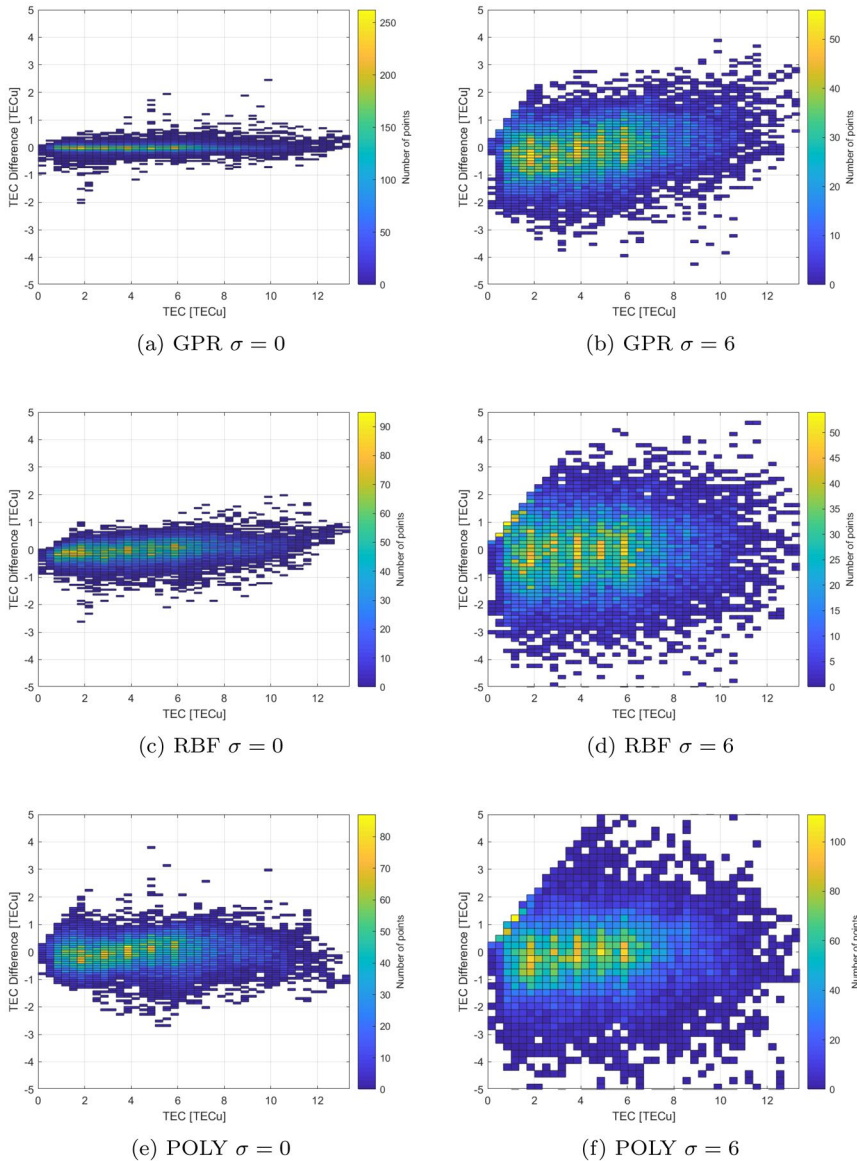
Fig. 5 Histogram of differencies derived from GPR, RBF, POLY methods with added gaussian noise, $\sigma = [0, 6]$ TECu

method with zero and $\sigma_s = 6[TECu]$ added noise. The Fig. 5 contains altogether six sub-plots. The Fig. 5a, b for the GPR, Fig. 5c, d for the RBF and Fig. 5e, f for the POLY method in case of zero and 6 TECu simulation noise. The Gaussian process regression model demonstrates its capabilities to effectively estimates TEC map from a snapshot measurement dataset. In error free case scenario, the GPR has better characteristics than

the RBF or polynomial fit method (Fig. 5.a). In case of large noise the polynomial fit shows inferior quality than the other two examined methods (Fig. 5f).

In Fig. 6, the global mean of $|d_n^k|$ showed in respect of the added noise. The error bars stand for the global standard deviation of d_n^k. In both quality indicators, the GPR shows better performance in overall domain of the increasing measurement noise Hhowever However the performance of the polynomial and the radial base function could be definitely improved with better parameter adjustments strategies. This simulation has no goal to state a comprehensive qaulity assessment of the three chosen estimation method, only to show GPR's ability to create regional TEC maps from epoch-wise observation data of a sparse monitoring network.

3.2 Real-time ionosphere map creation from RIMS observations

The performance of the GPR model was investigated in nine days from 2020, grouped by three consecutive days. The first three days are Jan 23-25, when the sunspot numbers reached the lowest in the current 11 year long cycle McIntosh et al. (2020). The next three days are April 19-21. They were chosen because of the observed high ionospheric activity on April 20. And the last three days are July 19-21. The EGNOS RIMS observation data were used to run the Kalman filter aided GPR model. The created TEC maps are compared to the ROB and CODE products, and they were regarded as reference. To give perspective of the derived quality indicators, the same comparison was also performed to the two reference ionospheric map. Three type of VTEC differences are derived based Eq. (55).

$$d_n^k(m_i, m_j) = VTEC_n^{m_i}(t_k) - VTEC_n^{m_j}(t_k) \tag{55}$$

where $m_i, m_j \in [ROB, CODE, GPR]$ indicates the type of TEC map.

The global mean, the absolute mean and the standard deviation of d_n^k values are derived from Eq. (55) and showed in Fig. 7a, b, c respectively for each day and combination. The
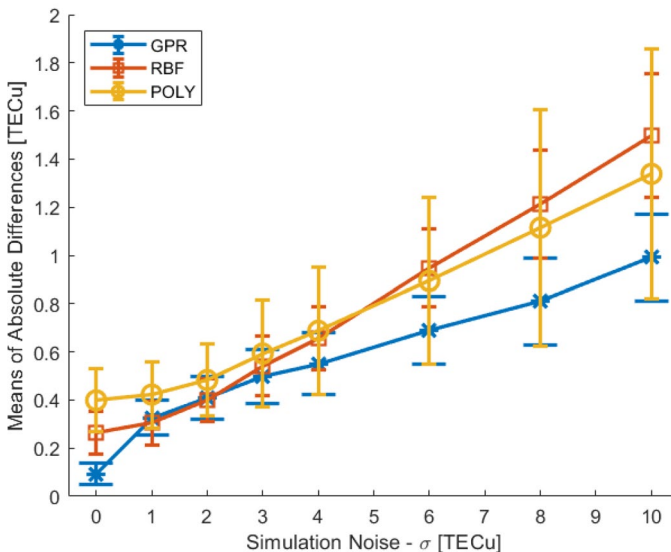

Fig. 6 Global mean of absolute differences in respect of simulation noise

(a) Mean of differences between TEC maps.

(b) Mean of absolute differences between TEC maps

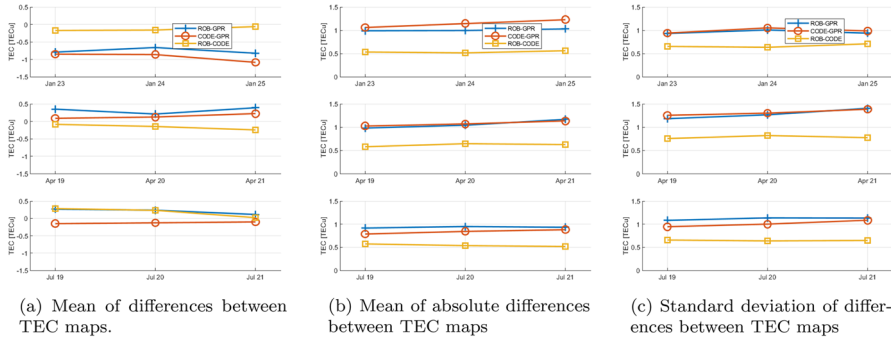(c) Standard deviation of differences between TEC maps

Fig. 7 Daily statistics between the ROB-GPR, CODE-GPR and ROB-CODE products depicted by blue, red and yellow lines respectively. The upper subfigures refer to data collected from January, the middle subfigures' data are from April, and the lower subfigures' data are from July
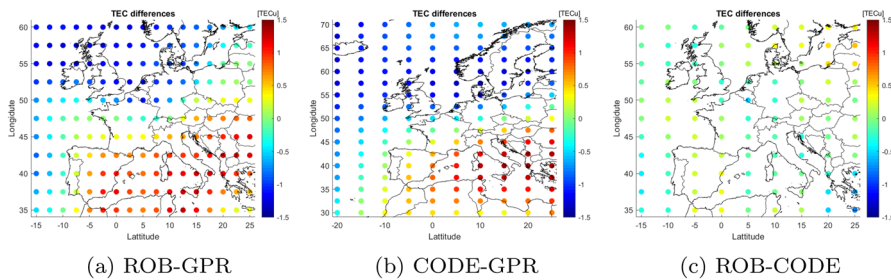

(a) ROB-GPR   (b) CODE-GPR   (c) ROB-CODE

Fig. 8 Mean of TEC value differences between ROB-GPR, CODE-GPR and ROB-CODE products in the assessed grid points of the nine examined day

mean absolute values of ROB-CODE combination are close to 0.5 TECu for each day, meanwhile the GPR combinations with the references have 1 TECu absolute mean. These discrepancies are consistent in the analyzed time intervals. The standard deviation of ROB-CODE are varies around 0.6-0.7 TECu. The standard deviations of GPR-ROB and GPR-CODE are less than twice the ROB-CODE and approximately 0.9-1.2 TECu. The global means of ROB-CODE are also closer to the expected zero than the GPR counterparts.

Spatial characteristics of mean and standard deviations are depicted in Figs. 8, 9 respectively. There is a visible trend in the mean values. The GPR method tends to overestimate the ionospheric delay in the northern region, hence the values are negative. On the other hand, the GPR underestimates the TEC values in the Mediterranean region. This skew does not appear when the two reference maps are analyzed in Fig. 8c. The mean absolute discrepancies of GPR are less than 2 TECu in all grid points. The gridwise standard deviations are less in the terrestrial regions, and shows correlation with the means both in case of ROB-GPR and CODE-GPR.

The two-dimensional histograms help to visualize the distribution of TEC differences in respect of the estimated values (Fig. 10). The plotted data are collected from all 9 investigated day. One can observe again a skewness in the GPR data in the lower TEC region (Fig. 10a, b). The outliers are contained in range of ±5 TECu. This distortion is not visible when the differencies of reference maps are depicted (Fig. 10c).
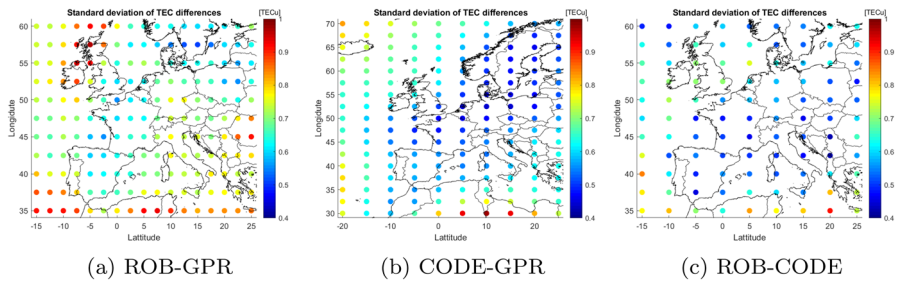
(a) ROB-GPR   (b) CODE-GPR   (c) ROB-CODE

Fig. 9 Standard deviation of TEC value differences between ROB-GPR, CODE-GPR and ROB-CODE products in the assessed grid points of the nine examined day

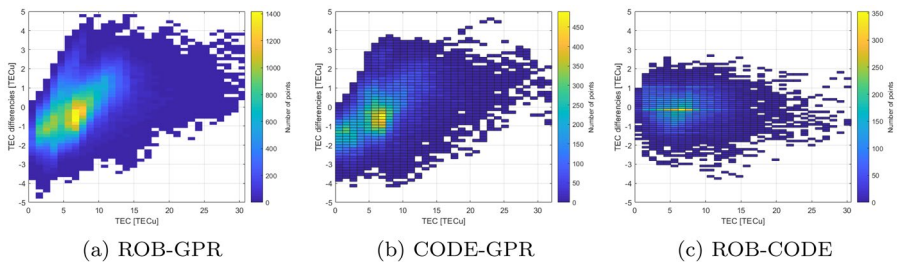(a) ROB-GPR   (b) CODE-GPR   (c) ROB-CODE

Fig. 10 Two-dimensional histograms of the differences between ROB-GPR, CODE-GPR and ROB-CODE TEC maps in respect of the TEC values of ROB product

To gain better insight, six grid points were chosen to show the daily variation of the VTEC values (Fig. 11). The shaded area represents the 2σ standard deviation. There is no shading in case of ROB product because of the lack of available variance information. It is noticeable that GPR estimates correctly the TEC variation in each case. Mark the bias at lower and higher latitude (Fig. 11a, b and e, f). This flaw in the estimation was already visible in Fig. 8, and it is present during the whole day, and not the result of a temporally localized disturbance. The authors assume that the main source of this systematic error is the insufficient DCB estimation of the RIMS receivers.

4 Conclusion

The presented Gaussian process regression approach is a novel and promising method for ionospheric model derivation from multi-frequency GNSS measurements. It is capable to accurately estimate regional Total Electron Content (TEC) maps from snapshot measurements of a relatively sparse monitoring station. Accuracy of the GPR-based models in this paper is about ± 2 TECu, which is comparable to the ± 1 TECu accuracy of the corresponding models in the literature. In addition to the TEC modelling, the hardware delays were also estimated by a Kalman filter continuously and independently of the GPR. In further research, this loosely coupled setup could be tightened with direct DCBs estimation by the GPR. The hyperparameters of the GPR are calculated
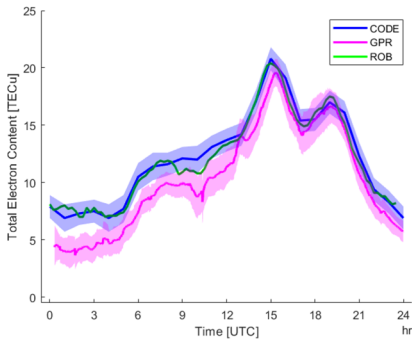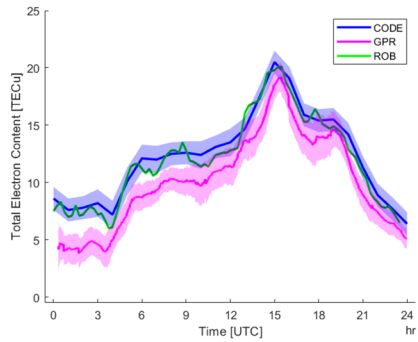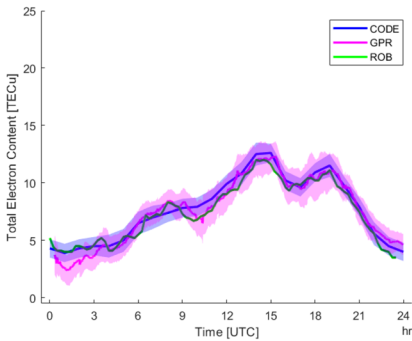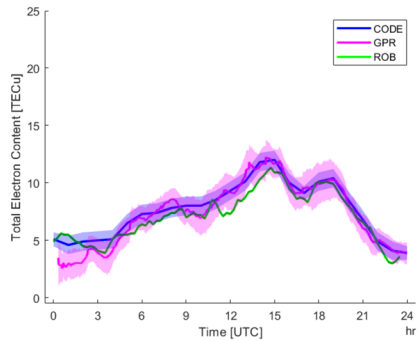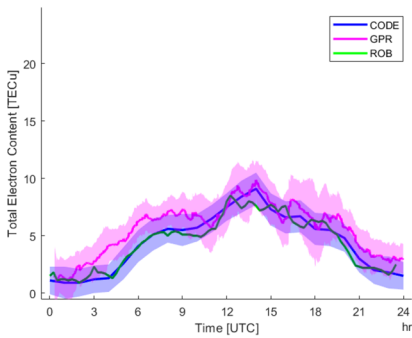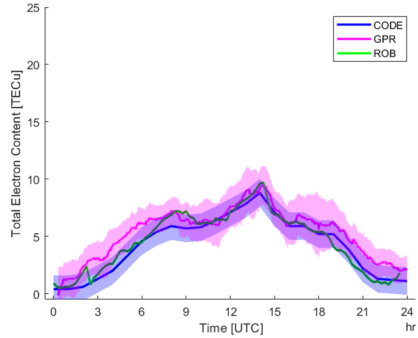
Fig. 11 Daily TEC variations at 6 selected location assessed from GPR model and CODE, ROB products on 2020 April 20

epoch-wise manner without taking into account the previous values, so it opens a window to tighten the hyperparameter searching space based on the previous values.

References

Ackermann ER, De Villiers JP, Cilliers PJ (2011) Nonlinear dynamic systems modeling using Gaussian processes: predicting ionospheric total electron content over South Africa. J Geophys Res Space Phys 116(A10)

Angrisano A, Gaglione S, Gioia C, Massaro M, Troisi S (2013) Benefit of the NeQuick Galileo version in GNSS single-point positioning. Int J Navig Obs

Bergeot et al (2010) Impact of the Halloween 2003 ionospheric storm on kinematic GPS positioning in Europe. GPS Solut 15(2):171–180

Bergeot N, Chevalier J-M, Bruyninx C, Pottiaux E, Aerts W, Baire Q, Legrand J, Defraigne P, Huang W (2014) Near real-time ionospheric monitoring over Europe at the Royal Observatory of Belgium using GNSS data. J Space Weather Space Clim

Blitzstein JK, Hwang J (2019) Introduction to probability. Chapman and Hall/CRC

Chang G, Xu T, Yao Y, Wang H, Zeng H (2019) Ionospheric delay prediction based on online polynomial modeling for real-time cycle slip repair of undifferenced triple-frequency GNSS signals. Measurement 146:289–297 (**ISSN 0263-2241**)

Ciraolo L, Azpilicueta F, Brunini C et al (2007) Calibration errors on experimental slant total electron content (TEC) determined with GPS. J Geod 81:111–120

Coster AJ, Gaposchkin EM, Thornton LE (1992) Real-time ionospheric monitoring system using GPS. Navigation 1992(39):191–204

Dach R, Hugentobler U, Fridez P, Meindl M (2007) Bernese GPS Software Version 5.0. Astronomical Institute, University of Bern, Staempfli Publications AG

Erdogan E, Schmidt M, Seitz F, Durmaz M (2017) Near real-time estimation of ionosphere vertical total electroncontent from GNSS satellites using B-splines in a Kalman filter. Ann Geophys 35(2):263–277

Farah A (2008) Comparison of GPS/Galileo single frequency iono-spheric models with vertical tec maps. Artif Satell 43:75–90

Feess WA, Stephens SG (1987) Evaluation of GPS ionospheric time-delay model. IEEE Trans Aerosp Electron Syst 23(3):332–338

Filjar R, Kos T, Kos S (2009) Klobuchar-like local model of quietspace weather GPS ionospheric delay for northern Adriatic. J Navig 62(3):543–554

Gordienko GI et al (2005) Ionospheric disturbances over Alma-Ata during the October–November 2003 magnetic storms. J Geophys Res 110:A09S35

Grunwald G, Bakuła M, Ciećko A, Kaźmierczak R (2016) Examination of GPS/EGNOS integrity in northeastern Poland. IET Radar Sonar Navig 2016(10):114–121

Hernández-Pajares M, Juan JM, Sanz J, Orus R, Garcia-Rigo A, Feltens J, Komjathy A, Schaer SC, Krankowski A (2009) The IGS VTEC maps: a reliable source of ionospheric information since 1998. J Geod 83:263–275

Hofmann-Wellenhof B, Lichtenegger H, Wasle E (2008) GNSS—global navigation satellite systems: GPS, GLONASS. Springer-Verlag Wien, Galileo & more

Hu Q et al (2005) On the magnetic topology of October/November 2003 events. J Geophys Res 110:A09S03

Inyurt S, Hasanpour Kashani M, Sekertekin A (2020) Ionospheric TECforecasting using Gaussian process regression (GPR) and multiplelinear regression (MLR) in Turkey. Astrophys Space Sci 365:99

Klobuchar JA (1987) Ionospheric time-delay algorithm for single-frequency GPS users. IEEE Trans Aerosp Electron Syst AES-23(3):325-331

Komjathy A, Sparks L, Wilson B, Mannucci AJ (2005) Automated daily processing of more than 1000 ground-based GPS receivers to study intense ionospheric storms. Radio Sci 40:RS6006

Lakshmi Mallika I, Venkata Ratnam D, Saravana Raman, Sivavaraprasad G (2020) Machine learning algorithm to forecast ionospheric time delays using Global Navigation satellite system observations. Acta Astronaut 173:221–231

Lin M, Song X, Qian Q, Li H, Sun L, Zhu S, Jin R (2019) Robust Gaussian process regression for real-time high precision GPS signal enhancement. In: Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery and data mining; Association for omputing Machinery: New York, NY, USA, pp 2838–2847

Llewellyn SK, Bent RB (1973) Documentation and description of the bent ionospheric model. IAFCRL-TR-73-0657, July 1973, AD772733

Lupsic B, Takács B (2019) Analysis of the EGNOS ionospheric model and its impact on the integrity level in the central eastern region. Int Arch Photogramm Remote Sens Spatial Inf Sci XLII-4/W14:159–165

McIntosh SW, Chapman S, Leamon RJ, Egeland R, Watkins NW (2020) Overlapping magnetic activity cycles and the sunspot number: forecasting sunspot cycle 25 amplitude. Solar Phys 295(12):1–4

Renga A, Causa F, Tancredi U, Grassi M (2018) Accurate ionospheric delay model for real-time GPS-based positioning of LEO satellites using horizontal VTEC gradient estimation. GPS Solut 22:1–4

Schaer SC (1999) Mapping and predicting the earth's ionosphere using the global positioning system. Ph.D. Thesis, Astronomical Institute, University of Berne, Bern, Switzerland

Schaer SC, Gurtner W, Feltens J (1998) IONEX: the IONosphere map EXchange format Version 1. IGS AC Workshop, Darmstadt, 1998 (IGS, Pasdena 1998)

Soley S, Farnworth R, A. van den Berg, Kremers R, Sanz J, Macabiau C, Fonseca A (2004) The data collection network: EGNOS revealed. In: Proc. ENC GNSS 2004, The European Navigation Conference, Rotterdam, The Netherlands. European Group of Institutes of Navigation, May 16–19 2004

Stein Michael L (1999) Interpolation of spatial data: some theory for Krigin. Springer

Sun X, You M, Ji Y, Yan S, Fu W, Liang W (2020) Modeling of total electron content in regional ionosphere based on GPS. In: 2020 7th international forum on electrical engineering and automation (IFEEA), pp 103–107

Teunissen PJG (1995) The least-square ambiguity decorrelation adjustment: a method for fast GPS ambiguity estimation. J Geod 1995(70):65–82

Ventura-Travest J, Flament D et al (2007) EGNOS—The European geostationary navigation overlay system—A cornerstone of Galileo. (ESA Publications, 2007), Reference ESA SP-1303

Wang N, Yuan Y, Li Z, Montenbruck O, Tan B (2015) Determination of differential code biases with multi-GNSS observations. J Geod 90:209–228

Xiang Y, Gao Y, Shi J, Xu C (2017) Carrier phase-based ionospheric observables using PPP models. Geod Geodyn 8(1):17–23

Yilmaz A, Akdogan KE, Gurun M (2009) Regional TEC mapping using neural networks. Radio Sci 44(3):1–16

Yu V, Yasyukevich A, Mylnikova A, Polyakova AS (2015) Estimating the total electron content absolute value from the GPS/GLONASS data. Results Phys 5:32–33

Zhang B, Yuan Y, Chai Y (2016) QIF-based GPS long-baseline ambiguity resolution with the aid of atmospheric delays determined by PPP. J Navig 69(6):1278–1292