A Novel Short-Medium Term Satellite Clock Error Prediction Algorithm Based on Modified Exponential Smoothing Method

Qiang Liu,1,2 Xihong Chen,1 Yongshun Zhang,1 Zan Liu,1 Chenlong Li,1 and Denghua Hu1

1 Air and Missile Defense College, Air Force Engineering University, Xi’an 710051, China
2 Unit 94259 of the PLA, Penglai, Shandong 265660, China

Correspondence should be addressed to Qiang Liu; dreamlq@163.com

Received 28 March 2018; Revised 7 August 2018; Accepted 12 August 2018; Published 26 August 2018

1. Introduction

High precision time synchronization is not only of vital importance for the operation of satellites, which will influence the precision of satellites’ navigation, locating, and timing, but also for distributed weapon systems, like the distributed netted radar system or the multistation radar system, etc., which will determine the precision of tracking, guiding, and locating directly. In order to keep high precision time synchronization, satellites usually take two way time transfer (TWTT) with the stations in the earth during the period of time when satellites are in the visual angle of earth stations. While satellites are out of the visual angle, the clocks in satellites have to operate by themselves, which have to predict the clock errors between satellites and earth stations. In clock error prediction, the predicted time length is always not long. When satellites fly into the visual angle of earth stations, the transfer could be reestablished. The break of transfer resulted from comparative position is usually in several hours. Also there are other factors resulting in transfer break, like interference, clock trouble, satellite fault, etc., which have to take medium term clock error prediction. The long term clock error prediction is few [1].

Aiming at the short-medium term clock error prediction, there are many studies which had been done, like grey model, quadratic polynomial model, LS-SVM algorithm, ARMA algorithm, functional network, etc. [2–5]. As the break of clock messages will influence real-time precise point positioning (PPP), some methods, like optimal arc length identifying, different polynomial order predicting and empirical model composed of a sixth-order harmonic function, etc., were brought forth, which improved the performances of PPP [6–8]. Besides, Wang et al. put forth a predicted method by a wavelet neural network model, which got the conclusion that the method could improve the predicted precision compared with the IGU-P clock products from examples [9]. Strandjord et al. took advantage of the repeatability of the clock variations and the potential of observed variations...
to predict the clock errors and the results implied that the methods could improve the predicted precision, which is of vital importance for the real-time GPS PNT [10]. Lu et al. put forth a fusion predicted method based on 4 typical prediction models and proved that the method improved accuracy and stability [11]. Further, exponential smoothing (ES) method has been widely used in time series prediction, like business, transportation, industry, meteorology, etc. [12–17]. In 2017 Wang et al. brought ES to clock error prediction, which had studied the performances of ES in short-medium term clock error prediction and made paralleled comparisons with other methods [18]. As for ES, Kolassa combined exponential smoothing forecasts and interval forecasts using Akaike weights and found a longer history tendency for better prediction interval coverage [19]. In order to forecast time series jointly in correlated random disturbances, Ana et al. studied initial conditions’ importance in ES models while considering forecast errors and prediction intervals, which were proved by examples [20]. Vercher et al. presented the Bayesian analysis of a general multivariate ES series jointly in correlated random disturbances. Akaikeweightsandfoundalongerhistorytendencyforbetter
term clock error prediction and made paralleled comparisons which had studied the performances of ES in short-medium
to make some paralleled analysis by examples. Finally con-

dividendstepsfortheprediction[24].

In single ES (SES), the weighted parameter (WP) \( \alpha \) is fixed, leading to growing predicted errors. Aiming at the application of ES in short-medium term satellite clock error prediction, we modify ES algorithm by a dynamic \( \alpha \) based on a sliding window (SW). Besides, in order to improve the predicted performances, we introduce grey model (GM) to learn the ES predicted errors, which also are combined with SW. All algorithms are proved by examples.

The rest of the paper is organized as follows. In Section 2, we present ES and modified ES (MES) models. The MES algorithms’ applications in short-medium term clock error prediction are presented in this Section. Then in Section 3, we make some paralleled analysis by examples. Finally conclusions are drawn in Section 4.

2. ES and Modified ES

As some satellite clock error series have big absolute value, which will bring extra computational complexity, especially in exponential calculation, so we make the original clock

\[
S_i' = S_{i-1}' + \alpha (x_i - S_{i-1}')
\]

(1)

From (1), we could get

\[
S_i' = (1 - \alpha)S_{i-1}' + \alpha x_i
\]

\[
= (1 - \alpha)[(1 - \alpha)S_{i-2}' + \alpha x_{i-1}] + \alpha x_i
\]

\[
= (1 - \alpha)^2 S_{i-2}' + (1 - \alpha)\alpha x_{i-1} + \alpha x_i
\]

(2)

The predicted value \( F_{t+1} \) of SES in time \( t + 1 \) is

\[
F_{t+1} = S_i' = \alpha x_i + (1 - \alpha) F_t
\]

(3)

Then we could get the double ES (DES) predicted value \( F_{t+m} \) in time \( t + m \)

\[
F_{t+m} = A_t + B_t m
\]

(4)

where

\[
A_t = 2S_i' - S_i''
\]

\[
B_t = \frac{\alpha}{1 - \alpha} (S_i' - S_i''
\]

(5)

\[
S_i'' = \alpha S_i' + (1 - \alpha)S_{i-1}''
\]

And in the same way, we will get the triple ES (TES) predicted value \( F_{t+m} \) in time \( t + m \)

\[
F_{t+m} = L_t + M_t m + \frac{1}{2} P_t m^2
\]

(6)

where

\[
L_t = 3S_i' - 3S_i'' + S_i'''
\]

\[
M_t = \frac{\alpha}{2(1 - \alpha)^2} \left[ (6 - 5\alpha)S_i' - 2(5 - 4\alpha)S_i''
\]

\[
+ (4 - 3\alpha)S_i'''
\]

(7)

\[
P_t = \frac{\alpha^2}{(1 - \alpha)^2} \left( S_i' - 2S_i'' + S_i'''
\]

\[
S_i''' = \alpha S_i'' + (1 - \alpha)S_{i-1}'''
\]

Usually we define \( S_i''' = S_i'' = S_i' = x_i \). The SES is suitable for the time series which are not change obvious and double for linear condition and triple for nonlinear [25]. Considering satellite clock data series’ characteristics, we analyze the clock predicted performances of DES and TES.

2.2. ES with Sliding Window (ESSW). As ES has the obvious defect of fixed weighted parameter \( \alpha \), we make some modification based on ES, which is showed in Figure 1. We divide the clock predicted series into \( k \) parts. In each part, we choose the best WP \( \alpha \) to minimize the predicted errors, which is interpreted by

\[
RMSE = \sqrt{\frac{\sum_{i=1}^{n} (x(i) - \bar{x}(i))^2}{n}}
\]

(8)
We choose root mean square error (RMSE) to evaluate the prediction performance. As $0 < \alpha < 1$, in order to get the best $\alpha_i$ for each part series, we divide the range into 99 parts averagely, where the initial value is 0.01 and the final value is 0.99. Then for each $\alpha_i$, we will get the corresponding RMSE. For comparative analysis, we could get the minimum RMSE and the best $\alpha_i$ for the part series. Then we will get the predicted series based on the ES prediction.

As the ES algorithm has clear defect of error accumulation, we bring in SW to update the learning swatches to reduce the effect, which refreshes the learning series by new predicted series. In part 2, we choose the predicted series as the learning series. Then we repeat the steps in part 1 to search the best WP and get the predicted series in part 2. Finally, we will get the complete series. As a result, we summarize steps of the ES algorithm with SW to predict clock error as follows.

**Step 1.** Make difference of the original clock series.

**Step 2 (parameters initialization).** The parameters related to the algorithm should be initialized firstly, including those in ES and SW.

**Step 3 (searching best $\alpha_i$).** In the learning window, we search the minimal RMSE corresponding to $\alpha_i$.

**Step 4 (prediction).** Based on the best WP, we use ES to predict the following series.

**Step 5 (sliding window).** We take part 1 for example. While the predicted length reaches $l_1$, we stop predicting. We update the learning swatches by the predicted series from part 1. Then we repeat the steps in part 1 to search the best WP $\alpha_2$, and get the predicted series in part 2. And then we will get the complete series.

### 2.3. ES+GM with Sliding Window.

In order to improve the precision of the prediction, we bring in the GM to predict the ES predicted errors based on foregoing algorithm, which is showed in Figure 2. The algorithm could be interpreted as follows. Taking part 1, for example.

**Step 1 to Step 3** are the same as the steps in ESSW

**Step 4 (ES prediction (ESP)).** Based on the best WP, we take ES to predict the following series. We call the step as ESP.

**Step 5 (GM learning).** In the ES learning window, based on the best $\alpha_i$, we will get the error series after ES learning. Then we take deep learning by GM. In the GM learning step, we will get the fitting parameters for GM, i.e., $\hat{a}$, $\hat{b}$.

**Step 6 (GM error predicting (GEP)).** Based on Step 5, we take GM to predict the ES prediction errors.

**Step 7 (fusion error predicting (FEP)).** We combine the ESP and GEP as FEP.

Step 8 is the same as Step 5 in ESSW, while the difference is that the updating swathes are the FEP series. The GM for ES predicted errors could be interpreted as

$$\frac{dy^{(1)}}{dt} + ay^{(1)} = b$$

Then we will get

$$\hat{y}^{(0)}(k+p) = \left[ y^{(0)}(1) - \frac{\hat{b}}{\hat{a}} \right] e^{-\hat{a}(k+p-1)} \cdot (1 - e^{\hat{a}})$$
where $p \geq 1$ is the predicted length and we will get the fitting parameters by least square estimate as

$$\begin{bmatrix} \hat{a} \\ \hat{b} \end{bmatrix} = (A^T A)^{-1} (A^T B)$$

(11)

where

$$A = \begin{bmatrix} -\frac{1}{2} y^{(1)}(1) + y^{(1)}(2) \\ -\frac{1}{2} y^{(1)}(2) + y^{(1)}(3) \\ \vdots \\ -\frac{1}{2} y^{(1)}(N-1) + y^{(1)}(N) \end{bmatrix}$$

and

$$B = \begin{bmatrix} y^{(0)}(2) \\ y^{(0)}(3) \\ \vdots \\ y^{(0)}(N) \end{bmatrix}$$

(12)

After we get the complete predicted series, we take inverse differential operation for the predicted clock error series. We will get the predicted errors compared with the original clock series.

3. Examples and Analysis

In order to study the performance of the algorithm above, we take the 1980th and 1981th GPS week clock error data (2017.12.17-2017.12.30), for example. We choose clocks from Rb clock and Cs clock in different types stochastically, which is showed in Table 1. And we choose IGS final precise ephemeris file for analysis.

We analyze the algorithm in short and medium term. We take the schemes as follows.

3.1. Short Term Predicted Performances. We take one day long prediction, for example, to analyze the short term predicted performances. As the predicted length is not long, we do not bring in sliding window in the algorithm. Besides, we choose GM as a contradictistinctive algorithm to analyze the algorithm the paper put forward.

**Scheme 1 (DES versus TES versus GM).** We choose one day clock error series for leaning, then we get the statistical predicted results of the chosen clocks, which are showed in Figure 3 and Table 2. We choose RMSE as the evaluated criterion and list the maximum and minimum of the predicted errors. Also we calculate the average (Avg) and standard (Std) of the schemes.

From Figure 3 and Table 2, we could get that the short term predicted performances of DES and TES are roughly in the same precision, while that of the DES is better than TES appreciably. But the precision of GM in short term prediction is worse than DES and TES. Apart from algorithms, we could find that the precision of Rb clocks is better than Cs clocks in short term prediction.
Table 2: The statistical chart of above schemes (unit in ns.).

<table>
<thead>
<tr>
<th>Scheme</th>
<th>No.</th>
<th>RMSE</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>Avg</th>
<th>Std</th>
</tr>
</thead>
<tbody>
<tr>
<td>DES</td>
<td></td>
<td></td>
<td>0.3363</td>
<td>0.2405</td>
<td>0.3225</td>
<td>0.2570</td>
<td>0.5088</td>
<td>0.8182</td>
<td>0.4139</td>
<td>0.2198</td>
</tr>
<tr>
<td></td>
<td>Max</td>
<td>1.2321</td>
<td>0.1874</td>
<td>0.1672</td>
<td>0.5926</td>
<td>1.2149</td>
<td>2.1844</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Min</td>
<td>-1.4544</td>
<td>-0.5476</td>
<td>-0.7088</td>
<td>-0.5061</td>
<td>-0.7692</td>
<td>-1.5594</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>RMSE</td>
<td>0.4691</td>
<td>0.5258</td>
<td>0.8922</td>
<td>0.2881</td>
<td>0.5435</td>
<td>0.8214</td>
<td></td>
<td>0.5900</td>
<td>0.2267</td>
</tr>
<tr>
<td>TES</td>
<td></td>
<td></td>
<td>1.1801</td>
<td>0.9491</td>
<td>1.7628</td>
<td>0.5479</td>
<td>1.0825</td>
<td>1.9872</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Max</td>
<td>-1.5104</td>
<td>-0.0705</td>
<td>0.0125</td>
<td>-0.5682</td>
<td>-0.9944</td>
<td>-1.7465</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>RMSE</td>
<td>0.3298</td>
<td>0.1253</td>
<td>0.2004</td>
<td>0.2509</td>
<td>0.4298</td>
<td>0.7968</td>
<td></td>
<td>0.3555</td>
<td>0.2404</td>
</tr>
<tr>
<td>GM</td>
<td></td>
<td></td>
<td>1.5075</td>
<td>0.2875</td>
<td>0.4857</td>
<td>0.5229</td>
<td>1.0631</td>
<td>1.6884</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Max</td>
<td>-1.1804</td>
<td>-0.3155</td>
<td>-0.4108</td>
<td>-0.5805</td>
<td>-1.1460</td>
<td>-2.0537</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>RMSE</td>
<td>0.3321</td>
<td>0.2401</td>
<td>0.3214</td>
<td>0.2562</td>
<td>0.5037</td>
<td>0.7996</td>
<td></td>
<td>0.4089</td>
<td>0.2131</td>
</tr>
<tr>
<td>DES+GM</td>
<td></td>
<td></td>
<td>0.4367</td>
<td>0.3113</td>
<td>0.7305</td>
<td>0.2843</td>
<td>0.5900</td>
<td>0.2267</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Max</td>
<td>1.0742</td>
<td>0.6572</td>
<td>1.5336</td>
<td>0.5605</td>
<td>0.7266</td>
<td>1.9872</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Min</td>
<td>-1.6213</td>
<td>-0.2455</td>
<td>-0.1339</td>
<td>-0.5561</td>
<td>-1.4685</td>
<td>-1.7465</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TES+GM</td>
<td></td>
<td></td>
<td>0.4001</td>
<td>1.0364</td>
<td>1.2312</td>
<td>0.3363</td>
<td>1.5164</td>
<td>0.7894</td>
<td>0.8772</td>
<td>0.4685</td>
</tr>
<tr>
<td></td>
<td>Max</td>
<td>-1.8322</td>
<td>-0.4108</td>
<td>-0.3155</td>
<td>-0.5805</td>
<td>-1.1460</td>
<td>-2.0537</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>RMSE</td>
<td>0.3321</td>
<td>0.2401</td>
<td>0.3214</td>
<td>0.2562</td>
<td>0.5037</td>
<td>0.7996</td>
<td></td>
<td>0.4089</td>
<td>0.2131</td>
</tr>
<tr>
<td>S2</td>
<td></td>
<td></td>
<td>1.2165</td>
<td>0.1878</td>
<td>0.1696</td>
<td>0.5897</td>
<td>0.7448</td>
<td>2.0719</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Max</td>
<td>-1.4703</td>
<td>-0.5475</td>
<td>-0.7061</td>
<td>-0.5134</td>
<td>-1.3902</td>
<td>-1.6737</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>RMSE</td>
<td>0.3171</td>
<td>0.5586</td>
<td>1.2093</td>
<td>0.2891</td>
<td>1.1880</td>
<td>0.7354</td>
<td>0.7163</td>
<td>0.4995</td>
<td>0.2164</td>
</tr>
<tr>
<td>TES+SW</td>
<td></td>
<td></td>
<td>1.4326</td>
<td>4.5575</td>
<td>6.4957</td>
<td>2.6798</td>
<td>1.2568</td>
<td>0.7520</td>
<td>2.8624</td>
<td>2.2443</td>
</tr>
<tr>
<td></td>
<td>Max</td>
<td>4.2711</td>
<td>4.8356</td>
<td>10.1357</td>
<td>5.8127</td>
<td>1.4003</td>
<td>2.2027</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>S3</td>
<td></td>
<td></td>
<td>3.7280</td>
<td>7.2823</td>
<td>15.7942</td>
<td>3.8862</td>
<td>1.3800</td>
<td>0.7894</td>
<td>5.2158</td>
<td>5.6630</td>
</tr>
<tr>
<td></td>
<td>Max</td>
<td>1.1801</td>
<td>15.6289</td>
<td>34.2391</td>
<td>0.0548</td>
<td>1.1644</td>
<td>2.2027</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Min</td>
<td>-6.0656</td>
<td>-0.0705</td>
<td>0.0125</td>
<td>-7.5281</td>
<td>-3.5282</td>
<td>-2.4249</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TES+GM</td>
<td></td>
<td></td>
<td>1.4326</td>
<td>4.5575</td>
<td>6.4957</td>
<td>2.6798</td>
<td>1.2568</td>
<td>0.7520</td>
<td>2.8624</td>
<td>2.2443</td>
</tr>
<tr>
<td></td>
<td>Max</td>
<td>4.2711</td>
<td>4.8356</td>
<td>10.1357</td>
<td>5.8127</td>
<td>1.4003</td>
<td>2.2027</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TES+GM</td>
<td></td>
<td></td>
<td>0.7098</td>
<td>3.0143</td>
<td>3.2632</td>
<td>1.7600</td>
<td>1.0822</td>
<td>0.6773</td>
<td>1.7555</td>
<td>1.4300</td>
</tr>
<tr>
<td></td>
<td>Max</td>
<td>1.8674</td>
<td>4.3660</td>
<td>9.8627</td>
<td>3.3411</td>
<td>0.5669</td>
<td>1.5817</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Min</td>
<td>-1.8219</td>
<td>-7.9424</td>
<td>-0.9719</td>
<td>-0.8493</td>
<td>-1.5469</td>
<td>-2.8538</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GM</td>
<td></td>
<td></td>
<td>0.2429</td>
<td>0.1440</td>
<td>0.2624</td>
<td>0.2692</td>
<td>0.7284</td>
<td>0.7310</td>
<td>0.3963</td>
<td>0.2621</td>
</tr>
<tr>
<td></td>
<td>Max</td>
<td>1.5075</td>
<td>0.4878</td>
<td>0.4857</td>
<td>0.8203</td>
<td>2.2155</td>
<td>2.5908</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Min</td>
<td>-1.1804</td>
<td>-0.4490</td>
<td>-0.9562</td>
<td>-0.8304</td>
<td>-1.1460</td>
<td>-2.0537</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

3.2. Medium Term Predicted Performances. We take one week long prediction for example to analyze the medium term predicted performances.

Scheme 3 (DES (TES) versus DES (TES)+SW versus DES (TES)+GM+SW versus GM). In this part, we take DES (TES), DES (TES)+SW, DES (TES)+GM+SW, and GM algorithms to predict the medium term clock errors. We choose clock errors in the 1st day of the chosen days as the learning watches for DES (TES) and GM. And the predicted length is 7 days. As for the SW, we divide the predicted length into two parts averagely; for example, the 1st part predicted length is 3.5 days. We choose the last 1 day predicted clock errors in the 1st part as the learning watches for the 2nd part. Then we operate the 2nd prediction as the 1st part in DES (TES) and...
DES (TES) + GM algorithms. The predicted performances are shown in Figure 5 and Table 2.

From Figure 5 and Table 2, we could get that DES’s the medium term clock error prediction performances are better than TES’s roughly. The SW could reduce the predicted errors at a certain. And error learning in GM could better the performances also. Comparing with GM, we could find that the DES’s predicted errors are roughly in the same precision level, while the TES’s are worse than GM. As for the satellite clocks, Rb clocks in DES’s performances are better than Cs clocks, while in TES they are opposite. From the statistical results in Table 2, we could conclude that the fusion algorithm of DES (TES) with sliding window based on error learning in GM could reduce the predicted errors in 35.37% (66.34%) compared with DES (TES) alone.

4. Conclusion

In this paper, we have mainly presented a novel fusion algorithm of DES (TES) in sliding window based on predicted error learning in GM for short-medium term clock error prediction. We firstly analyze the basic DES and TES algorithm. Then we present ES with sliding window and ES+GM with sliding window algorithms in detailed steps. Furthermore, we take two-GPS week clock error data to analysis the algorithms we put forward. From the calculation results, for short term clock error prediction, we could get that the predicted precisions of algorithms based ES are less than 0.4ns and GM error learning could better the ES predicted performances slightly. As for the medium term clock error prediction, we could conclude that error learning in GM could better the performances and fusion algorithm of DES (TES) with sliding window based on error learning in GM could reduce the predicted errors in 35.37% (66.34%) compared with DES (TES) alone. And the precision of TES is worse than DES in medium term clock error prediction, which is roughly in the same level of GM. The fusion algorithm could not only be applied in clock error prediction in satellites, but also in distributed netted systems, like netted radars systems, which is of vital importance for the performances of the systems' efficiency.
Data Availability

The clock data used in the paper are from the web-page of ftp://igs.ensg.ign.fr/pub/igs/products.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

The paper was supported by National Natural Science Foundation of China (NSFC) under Grant no. 61701525 and China Postdoctoral Science Foundation under Grant no. 2017M623351.

References


