Stable Kalman Filters for Processing Clock Measurement Data

P. A. Clements
Communications Systems Research Section

B. P. Gibbs and J. S. Vandergraft
Computational Engineering, Inc., Laurel, Maryland

Kalman filters have been used for some time to process clock measurement data. Due to instabilities in the standard Kalman filter algorithms, the results have been unreliable and difficult to obtain. Often, in order to obtain reasonable results, the data has had to be manually edited, the filter fine tuned, or the model adjusted by the analyst. During the past several years, stable forms of the Kalman filter have been developed, implemented, and used in many diverse applications. These algorithms, while algebraically equivalent to the standard Kalman filter, exhibit excellent numerical properties. Two of these stable algorithms, the Upper triangular-Diagonal (UD) filter and the Square Root Information Filter (SRIF), have been implemented to replace the standard Kalman filter used to process data from the DSN hydrogen maser clocks. The data are time offsets between the clocks in the DSN, the timescale at the National Institute of Standards and Technology (NIST), and two geographically intermediate clocks. The measurements are made by using the GPS navigation satellites in mutual view between clocks. The filter programs allow the user to easily modify the clock models, the GPS satellite dependent biases, and the random noise levels in order to compare different modeling assumptions.

The results of this study show the usefulness of such software for processing clock data. The UD filter is indeed a stable, efficient, and flexible method for obtaining optimal estimates of clock offsets, offset rates, and drift rates. A brief overview of the UD filter is also given.

I. Introduction

The requirements of the DSN-complex clocks are that each clock be maintained within 10 microseconds of Coordinated Universal Time (UTC) as realized at the National Institute of Standards and Technology (NIST), with a knowledge of 1 microsecond and that the rate of each clock be kept within $\pm 1E-12df/f$ with a knowledge of

$\pm 3E-13df/f$. The quantity df/f is known as the fractional frequency deviation where

$$df/f = \frac{f_m - f_0}{f_0}$$

and f_m is the measured frequency and f_0 is the reference frequency. Also there is a requirement that a permanent record be kept of the synchronization and syntonization of the DSN-complex clocks. In order to meet these requirements, the time and frequency offset to UTC (NIST) is monitored using the Global Positioning System (GPS) navigation space vehicles. The goal (not yet realized) is to adjust the rates of all the clocks at the same time at approximately 200-day intervals. The clock rates (frequencies) are adjusted when they are above UTC (NBS) by a certain amount to below, and then allowed to drift through zero rate offset from UTC (NIST) until they are again high enough to require adjustment. This technique, done with some care, allows one to keep the clocks within the required time and frequency offset limits and still not have any jumps in the time offset. All of the drift rates of the hydrogen masers are about the same, therefore, the complex clocks can be kept in close synchronization and syntonization.

Monitoring of the DSN-complex clocks is done by the operations analyst at JPL. Adjustments of the clocks is recommended by the operations analyst with the advice of the engineering staff. The adjustment is scheduled after a check of the users of the DSN and is made during a maintenance period at the complex. The actual adjustment is made by the engineering staff at the complex. It is assumed that the drift of the hydrogen maser rate is due to cavity drift, therefore, rate adjustments are made by adjusting the varactor, thereby changing the apparent cavity size of the hydrogen maser.

At critical times during some projects (such as the Voyager Uranus encounter), it is desirable to have the clocks synchronized and syntonized more closely than the above specifications. When this is the situation, the operations analyst will predict the time and rate of the DSN clocks 30 to 90 days prior to the encounter date and set the rate to allow the clocks to drift to the near zero time and rate offset by the critical time. At the time of the Voyager Uranus encounter, JPL engineering staff used a Kalman filter to predict the clock offset 60 days into the future with excellent results [3].

At JPL, a Kalman filter has been used to process clock offset measurements but it has never been used operationally. The present operational environment for clock management in the DSN is a personal computer with a considerable amount of manual intervention. The data is handled in weekly batches. By 1992 the process is to be transferred to the Network Frequency and Timing subsystem (NFT), which is a part of the DSN Frequency and Timing system (DFT). The flow of data and estimate updates will occur more often (perhaps as often as once per hour) so as to allow operations to be able to monitor clock performance in near real time. This will require a fairly robust processing environment with automatic elimination of outliers and other techniques to make the process as autonomous as possible. It is expected that operations-analyst intervention will be relegated to filter adjustments in response to scheduled changes of time or rate in any of the clocks being entered into the filter parameters.

The operations analysts will have the opportunity to do postanalysis of the data. In the near future, it is expected that some data will continue to be available on a weekly basis and that better ephemerides will be available several weeks after the fact. At this time, use of the filter/smoother by the operations analyst can produce better estimates of the clock parameters for archiving. To achieve these goals, a robust filter and smoother program is under development at JPL. The results reported in this article were obtained by running this program with data as described in the next section. No "fine tuning" adjustments or manual data editing were performed. The output, however, was excellent. The filter used in the JPL program is the numerically stable Upper triangular-Diagonal (UD) form of the Kalman filter. The UD filter has been under development for several years [1] and has earned a reputation for being accurate and efficient. The more traditional Kalman formulation has been known for some time to be numerically unstable. That is, due to rounding errors within the computer, the results produced by the filter program are sometimes completely wrong. The smoother used in this code is a recent improvement [2] of the Rauch-Tung-Streibel (RTS) smoother [5]. The revised RTS smoother is also robust and accurate.

The filter and smoother programs make use of subroutines from the Estimation Subroutine Library (ESL). This is a collection of FORTRAN77 subroutines for constructing filters and smoothers. The routines in the ESL have been carefully coded and tested to ensure their accuracy and reliability. Overall, the goal is to produce optimal estimates of clock parameters without manual intervention. The results of this first attempt are extremely encouraging.

II. Clock Measurements

There is an NIST-designed GPS timing receiver located at each complex. A timing pulse from the complex master clock is fed to each receiver so that the data output from the receiver is the offset of the complex master clock to GPS timescale. The receivers are maintained on the mutual view schedule, which is generated by the Bureau International des Poids et Measures (BIPM) in Paris, France. These data are kept in a database along with data from UTC (RRL) and UTC (TAO) in Japan, UTC (NIST), and the NIST cesium clock at WWVH Hawaii, U. S. The data from Japan are obtained from the GE MKIII catalogue, which is administered worldwide by the BIPM and inside the U. S. by the United States Naval Observatory (USNO). Each of the data lines is tagged to indicate the receiver from which it came and then placed in a file and stored on a disk in a personal computer. There is one file for each day, and it is named by Modified Julian Day.[1]

The first process is to take the first difference of the data to obtain the differences between the clocks on the ground, thus eliminating the space vehicle clocks. There are six clocks used; because of the geometry of the clock locations there are 11 possible mutual views available (Fig. 1). A program is used to produce a file which contains as entries the station pair, the satellite number, the MJD, the time (second) the data was collected, and the difference in time in tenths of nanoseconds between the first and second clock. (The above format could easily be adapted to include clock difference data from other sources such as two-way time coordination.) Data was generated by this program from MJD 47000 to 47132. There were approximately 4000 measurements in the data set.

III. Mathematical Model

In order to describe the mathematical model used for this analysis, the following notation is introduced:

$s_i(t)$: the clock offset at station i. That is, $s_i = c_i - T$, where c_i is the clock reading at station i and T is the "true" time. For this analysis, "true" time is taken to be the time given by UTC (NIST).

$r_i(t)$: the clock offset rate, also called the frequency error. This is just the time derivative of the offset, i.e., $r_i = ds_i/dt$.

$d_i(t)$: the drift rate, that is, the second derivative of the offset, $d_i = d^2 s_i/dt^2$.

$b_{ikl}(t)$: the measurement biases. These depend on the two stations, i and k, and on the satellite, l, being observed. These biases will be explained in more detail later.

The dynamics model describes how the above quantities change over time. This is expressed as follows:

$$\dot{s}_i(t) = r_i(t) \tag{1}$$

$$\dot{r}_i(t) = d_i(t) \tag{2}$$

$$\dot{d}_i(t) = w_i(t) \tag{3}$$

where $\cdot$ denotes time differentiation, and $w_i(t)$ is white noise, with zero mean and known power spectral density (psd) W. That is,

$$E\Big[w_i(t)w_i(\tau)\Big] = W\delta(t-\tau) \tag{4}$$

where $E[\cdot]$ is the *expected value* operation, and where δ is the Dirac delta function. Note that Eqs. (1) and (2) are just restatements of the definitions of r_i and d_i. Equation (3) says that the rates are "approximately" constant.

In order to apply a discrete filter to this problem, Eqs. (1–3) must be put into the form of a discrete dynamics equation

$$\underline{x}(t+\Delta t) = \Phi(t,\Delta t)\underline{x}(t) + w(t) \tag{5}$$

where $\underline{x}$ denotes a vector with components that include s_i, r_i, d_i for each station. Note that all of these quantities are time-dependent; however, to simplify the notation, this dependence will often be suppressed. The matrix Φ, called the transition matrix, is determined by writing out a discrete form of Eqs. (1–3). A simple way to do this, for example, at least for the offsets and rates, is to use a discrete form of Taylor's Theorem:

$$s_i(t+\Delta t) = s_i(t) + r_i(t)\Delta t + 0.5 d_i(t)\Delta t^2 \tag{6}$$

$$r_i(t+\Delta t) = r_i(t) + d_i(t)\Delta t \tag{7}$$

[1] The Modified Julian Day (MJD) is a continuous day count with an initial epoch of 0000 hours UT on November 17, 1858.

For the drift rates, this same approach produces

$$d_i(t+\Delta t) = d_i(t) + \int_o^{\Delta t} w_i(\tau)d\tau \qquad (8)$$

where $w_i(t)$ represents the random noise. However, the effects of the noise on the drift rates are shown directly in Eq. (8), but this same noise also affects the offsets and frequency errors because of the presence of $d_i(t)$ in Eqs. (6) and (7). As written above, however, there will be a delay, caused by the time discretization, before the noise at time t propagates into the offset and rate. To avoid this delay, noise terms are included directly in the equations for s_i and r_i. This is done by using the continuous form of the transition matrix to propagate the noise over the interval t to $t+\Delta t$. That is, if the noise that enters the continuous system at time t is given by the vector

$$w(t) = \left(0, 0, w_r(t)\right)^T$$

where $w_r(t)$ is white noise with zero mean and psd W_r, then the integrated effect of the noise at time $t+\Delta t$ will be the vector

$$q(t+\Delta t) = (q_s, q_r, q_d)^T$$

where $(q_s, q_r, q_s)^T \sim N(0, Q)$ and

$$Q = E\left[\int_0^{\Delta t} \Phi(\Delta t - \tau)\, w(\tau)d\tau \int_0^{\Delta t} w^T(\sigma)\Phi^T(\Delta t - \sigma)d\sigma\right]$$

$$= E\left[\int_0^{\Delta t}\int_0^{\Delta t} \Phi(\tau)w(\Delta t - \tau)w^T(\Delta t - \sigma)\Phi^T(\sigma)d\sigma d\tau\right]$$

$$= \left(0.5\tau^2, \tau, 1\right)^T W_r(0.5\tau^2, \tau, 1)d\tau \qquad (9)$$

since

$$\Phi(t) = \begin{bmatrix} 1 & t & 0.5t^2 \\ 0 & 1 & t \\ 0 & 0 & 1 \end{bmatrix}$$

and $E[w_r^T w_r] = W_r$. The integration in Eq. (9) gives

$$Q = W_r \begin{bmatrix} \frac{1}{20}\Delta t^5 & \frac{1}{8}\Delta t^4 & \frac{1}{6}\Delta t^3 \\ \frac{1}{8}\Delta t^4 & \frac{1}{3}\Delta t^3 & \frac{1}{2}\Delta t^2 \\ \frac{1}{6}\Delta t^3 & \frac{1}{2}\Delta t^2 & \Delta t \end{bmatrix} \qquad (10)$$

Thus the discrete dynamics model for these filter states is

$$\begin{pmatrix} s_i(t+\Delta t) \\ r_i(t+\Delta t) \\ d_i(t+\Delta t) \end{pmatrix} = \Phi_i \begin{pmatrix} s_i(t) \\ r_i(t) \\ d_i(t) \end{pmatrix} + w_i \qquad (11)$$

where

$$\Phi_i = \begin{bmatrix} 1 & \Delta t & \frac{1}{2}\Delta t^2 \\ 0 & 1 & \Delta t \\ 0 & 0 & 1 \end{bmatrix} \qquad (12)$$

$w_i \sim N(0, Q_i)$, and Q_i is given by Eq. (10). There will be a transition, of the form of Eq. (11), for each station. Thus, the full transition matrix Φ in Eq. (5) will have blocks of the form of Eq. (12) on its diagonal. The process noise term, $w(t)$, in Eq. (5) will have as its covariance matrix a block diagonal matrix, with 3×3 matrices of the form of Eq. (10) on the diagonal.

The measurements to be used for estimating the offsets, drifts, and drift rates are the differences in clock values from the GPS times, as measured at two different stations. That is, the measurements are

$$z_{ikl}(t_j)$$

which are assumed to be of the form

$$z_{ikl}(t_j) = \left(c_i(t_j) - GPS_l\right) - \left(c_k(t_j) - GPS_l\right) + b_{ikl}(t_j) + v_{ik}(t_j)$$

$$= c_i(t_j) - c_k(t_j) + b_{ikl}(t_j) + v_{ik}(t_j) \qquad (13)$$

where, as defined earlier, $c_i(t_j)$ is the clock reading at station i at time t_j. The fact that $s_i(t_j) = c_i(t_j) - NIST(t_j)$ allows one also to write this as

$$z_{ikl}(t_j) = \Big(c_i(t_j) - NIST(t_j)\Big) - \Big(c_k(t_j) - NIST(t_j)\Big)$$
$$+ b_{ikl}(t_j) + v_{ik}(t_j)$$
$$= s_i(t_j) - s_k(t_j) + b_{ikl}(t_j) + v_{ik}(t_j)$$

The first expression is used to obtain the measurement data as described in Section II. That is, the data actually used for the analysis are values for $c_i(t_j) - GPS_l(t_j)$. However, for the filter/smoother, the measurements are assumed to be of the second form, where the biases, $b_{ikl}(t_j)$, satisfy the trivial dynamics equation

$$b_{ikl}(t + \Delta t) = b_{ikl}(t) \tag{14}$$

and the *measurement noise*, $v_{ik}(t_j)$, is, as usual, a normally distributed zero mean random variable with constant variance r. The biases are appended to the "core states" to obtain the actual state vector used in the filter:

$$x = (s_1, r_1, d_1, s_2, r_2, d_2, \ldots, s_m, r_m, d_m, b_{111}, b_{112}, b_{113}, \ldots)$$

The transition matrix will have an identity matrix attached to it to reflect Eq. (14). Then the measurements can be written in standard form as

$$z_j = H_j \underline{x}_j + v_j \tag{15}$$

where H_j is a $1 \times n$ measurement matrix, consisting, in this case, of two positive ones and a negative one, in appropriate positions. For example, a measurement based on observations from stations 1 and 3, and satellite 6, will have a measurement matrix

$$H = (1, 0, 0, 0, 0, 0, -1, 0, 0, \ldots, 0, 1, 0, \ldots)$$

where the final one appears in the position corresponding to bias b_{136}.

IV. The UD Form of the Kalman Filter

The Kalman Filter has become the standard tool for computing optimal estimates of a state vector, $\underline{x}(t)$, governed by a mathematical model of the form of Eqs. (5) and (15). To describe the form of the filter that is of interest here, the following notation will be used:

$\underline{x}_{k|j}$: estimate of $\underline{x}(t_k)$ using data up to time t_j

$P_{k|j}$: covariance matrix for the error in $\underline{x}_{k|j}$ as an approximation to $\underline{x}(t_k)$

The Kalman Filter time update, also called the prediction step is

$$\underline{x}_{k|k-1} = \Phi_k \underline{x}_{k-1|k-1} \tag{16}$$

$$P_{k|k-1} = \Phi_k P_{k-1|k-1} \Phi_k^T + Q_k \tag{17}$$

The measurement update is accomplished via the following set of equations:

$$B_k = H_k P_{k|k-1} H_k^T + R_k \tag{18}$$

$$K_k = P_{k|k-1} H_k^T B_k^{-1} \qquad \text{(Kalman gain)} \tag{19}$$

$$P_{k|k} = (I - K_k H_k) P_{k|k-1} \tag{20}$$

$$\nu_k = z_k - H_k x_{k|k-1} \qquad \text{(innovations)} \tag{21}$$

$$\underline{x}_{k|k} = \underline{x}_{k|k-1} + K_k \nu_k \tag{22}$$

The numerical instability of the Kalman equations, as noted in Section I, is due to the structure of these equations. The most common indication that something has gone wrong is when rounding error causes $P_{k|k}$ to be not positive definite. Note that it is not clear from these equations that $P_{k|k}$ is even symmetric.

The UD form of the Kalman Filter is algebraically equivalent to the above equations, but has much better numerical properties. It is based on the fact that any positive definite matrix P can be written as a product of matrices

$$P = UDU^T \tag{23}$$

where U is an upper triangular matrix with ones on the diagonal, and D is a diagonal matrix, with positive diagonal elements. The idea behind the UD formulation is to start with the UD factors of $P_{0|0}$, then compute the factors of all the remaining covariances, without ever computing the covariances themselves. It can be shown that by avoiding the covariances, the accuracy of the results is greatly improved. On the other hand, if the covariances are needed for output they can be easily obtained by simply multiplying the factors together.

To describe the time update in UD form, assume that the factors of $P_{k|k}$ are known from the previous step. That is, $U_{k|k}$ and $D_{k|k}$ have already been computed, where

$$P_{k|k} = U_{k|k} D_{k|k} U_{k|k}^T \tag{24}$$

Then $\underline{x}_{k+1|k}$ is computed just as in the Kalman formulation; but the equation for $P_{k+1|k}$ becomes

$$\begin{aligned} P_{k+1|k} &= \Phi_k P_{k|k} \Phi_k^T + Q_k \\ &= \Phi_k U_{k|k} D_{k|k} U_{k|k}^T \Phi_k^T + Q_k \end{aligned} \tag{25}$$

Since Φ_k is upper triangular, so is the product $\Phi_k U_{k|k}$. Hence, if Q_k were zero, then the factors of $P_{k+1|k}$ would be just

$$R_k D_k R_k^T$$

where

$$R_k = \Phi_k U_{k|k} \tag{26}$$

In order to account for the process noise, Q_k is factored into its UD factors

$$Q_k = U_q D_q U_q^T$$

which can also be written as

$$Q_k = \sum_{i=1}^{n} d_i q_i q_i^T \tag{27}$$

where q_i is the ith column of U_q. Then Eq. (25), using Eqs. (26) and (27) becomes

$$P_{k+1|k} = R_k D_k R_k + \sum_{i=1}^{n} d_i q_i q_i^T \tag{28}$$

Now the time update step is completed by finding the UD factors of $P_{k+1|k}$. This is done by a series of "rank one" adjustments to previous factors. That is, consider the general problem: given factors R, D, compute the factors of the matrix

$$RDR^T + dqq^T$$

where d is a positive scalar, and q is a vector.

The matrix dqq^T is a matrix of rank one. Hence dqq^T is called a "rank one" adjustment to RDR^T. There is a very accurate and efficient algorithm for computing the factors of this new matrix, given the factors R, D, and the quantities d and q. It is important here that the factors are not multiplied together, as this would destroy the numerical accuracy. This rank-one update algorithm, called the Turner-Agee Algorithm [1], is applied repeatedly to Eq. (28) to complete the time update step of the filter.

Next, consider the measurement update, as given in the Kalman form by Eqs. (18–22). The Kalman equation can be used to process vector measurements. However, the UD form is more efficient if the measurements are scalars. In this case, in the measurement update equations, B_k is a scalar, as is v_k, and K is a vector. The measurement update is done several times, once for each scalar measurement for each time update. After the previous time update, the factors $U_{k|k-1}$ and $D_{k|k-1}$ of $P_{k|k-1}$ are available. The (scalar) B_k as defined by Eq. (18) is computed as follows:

$$\begin{aligned} B_k &= H_k U_{k|k-1} D_{k|k-1} U_{k|k-1}^T H_k^T + R_k \\ &= y_k^T D_{k|k-1} y_k + R_k \end{aligned}$$

where

$$y_k = U_{k|k-1}^T H_k^T$$

is a vector. Next, Eqs. (19) and (20) are written as

$$\begin{aligned} P_{k|k} &= (I - K_k H_k) P_{k|k-1} \\ &= P_{k|k-1} - \frac{1}{B_k} P_{k|k-1} H_k^T H_k P_{k|k-1} \\ &= U_{k|k-1} D_{k|k-1} U_{k|k-1}^T \\ &\quad - \frac{1}{B_k} U_{k|k-1} D_{k|k-1} U_{k|k-1}^T H_k^T H_k U_{k|k-1} D_{k|k-1} U_{k|k-1}^T \\ &= U_{k|k-1} \left[D_{k|k-1} - \frac{1}{B_k} V V^T \right] U_{k|k-1}^T \end{aligned} \tag{29}$$

where V is the vector defined by

$$V = D_{k|k-1} U_{k|k-1}^T H_k^T$$

All that remains is to compute the factors U_v, D_v of

$$D_{k|k-1} - \frac{1}{B_k} V V^T \tag{30}$$

Then it follows that

$$P_{k|k} = \underbrace{U_{k|k-1} U_v}_{U_{k|k}} \; \underbrace{D_v}_{D_{k|k}} \; \underbrace{U_v^T U_{k|k-1}^T}_{U_{k|k}^T} \tag{31}$$

Notice that Eq. 30 is a rank-one modification of a diagonal matrix. Hence its factors can be computed accurately and efficiently, just as was done in the time update.

While this UD formulation may seem more complicated than the relatively simple matrix equations of the Kalman form (Eqs. 16–22), a computer program to implement the UD filter is very straightforward. Two special subroutines are needed:

(1) A subroutine to efficiently compute the product of two upper triangular matrices, such as $\Phi_k U_k$

(2) A subroutine to adjust the factors of a matrix by a rank-one addition: given factors U, D, a positive scalar c, and a vector y, compute the factors of $U D U^T + c y y^T$

Then the time update is done by the following steps:

(1) Compute $R = \Phi_k U$ (sometimes called the deterministic update)

(2) For each process noise term $d_i q_i q_i^T$, update the factors using the rank-one adjustment routine

The measurement update is also broken down into a series of simple steps, one for each scalar measurement:

(1) Compute the scalar $B = y^T D y + r$ where $y = U^T H^T$

(2) Use the rank-one adjustment routine to compute the factors of $D - \frac{1}{B} V V^T$ where $V = D U^T H^T$

(3) Combine these factors with the previous factors, according to Eq. 31

V. Filter Implementation

In addition to the clock measurement data, as described in Section II, the filter routine must be given the following information:

(1) Prior estimates of the state vector

(2) Uncertainties associated with the state estimates

(3) Process-noise standard deviations

(4) Measurement-noise standard deviations

Generally, the filter is somewhat insensitive to prior estimates of the state; hence, they are usually initialized at zero. The uncertainties must be large enough so that the filter will accept the measurement data and use it to adjust the state estimates, except that the offsets for the NIST station should remain close to zero. The process noise can depend on the station; however, for the results of Section VI, all stations were given the same amount of noise. Similarly, the measurement noise can be made a function of the stations and the space vehicle. For simplicity during this first analysis, however, the measurement noise uncertainty was assumed to be the same for all measurements.

In order to run the smoother, certain information must be saved at each measurement update time. Thus, to limit the amount of storage required to run the filter and smoother, the program was set up to process measurements in so-called "mini-batches." Each mini-batch consists of all measurements obtained during a fixed timespan, say one day, or a half a day. The midpoint of this timespan is chosen as the time of the update (also called the "epoch time"). The measurements within the mini-batch are translated to the epoch time by using the same transition matrix, $\Phi(t_1, t_2)$, as is used in the dynamics equation. More precisely, suppose that the actual measurement is

$$z(t_1) = H x(t_1) + w \tag{32}$$

where t_1 denotes the time at which the measurement was taken. Let t_2 be the epoch time for this mini-batch. Then, if it is assumed that the timespan for this mini-batch is sufficiently short, so that the integrated effect of the noise is negligible, one can write

$$x(t_1) = \Phi(t_2, t_1) x(t_2) \tag{33}$$

so that Eq. (32) becomes

$$z(t_1) = H \Phi(t_2, t_1) x(t_2) + w \tag{34}$$

That is, the measurement $z(t_1)$ becomes a measurement of the state at the epoch time, but with the measurement matrix given by $H\Phi$ instead of just H. It is important to realize that this use of mini-batches can change the filter results because it assumes that the effect of the process noise over the timespan of the mini-batch is negligible. The code that was developed allows the user to select one or more mini-batches per day, or to process each measurement as it is received.

The filter/smoother was implemented in FORTRAN-77, making use of routines from the Estimation Subroutine Library (ESL) to do most of the numerical computations. The UD form of the Kalman filter, as described in Section IV was used for the filter; the smoother made use of Bierman's modification of the RTS smoother [2]. The general structure of the program is as follows:

(1) Read a namelist file to initialize the filter:

Start and stop times

Maximum number of measurements to process

Initial error covariances and state estimate

(2) For each measurement:

If new minibatch,

(a) Do time update

(b) Print estimates and estimate error standard deviations

(c) Translate measurements to epoch time and do the measurement update; write the smoother gains to a file for later use

(3) After all measurements have been processed, print out the correlation coefficients

(4) If the smoother has been requested, run the smoother back to the start time

The results of Section VI were obtained by running the program with 6 stations, 11 station pairs, and 6 satellites. Hence the filter state vector has 84 components. An important feature of the filter implementation is the data editing capability. Before each measurement is actually included into the estimate, a test of the measurement's consistency with past measurements is performed. This is done as follows. First the residual (or innovations) as defined by Eq. (21) is computed, as is the variance of this residual which is just the quantity B_k defined by Eq. (18). If the square of the residual divided by its variance is less than a user-specified tolerance, then the measurement is accepted and used to update the current estimates. For the results described in Section VI, a tolerance of 400 was used. This means that a measurement was rejected only if its residual was more than 20 standard deviations away from its mean value of zero.

VI. Numerical Results

The numerical results described in this section were obtained by running the filter and smoother on the data described in Section II. Approximately 10 data points were rejected by the data editor that is built into the filter. Figure 2 gives an overview of the results of the smoother. This figure shows the offsets from UTC (NIST) of the three clocks in the DSN. This is a useful display for the operations analyst. It can be used to tell when adjustments to the rates are in order. Notice that a rate adjustment would have been appropriate around MJD 47080; however an adjustment was not made until MJD 47130. The Allan variances of the smoother results, shown in Fig. 3, are consistent with those reported in [4], but are somewhat better. It is felt that these variances can be further improved with better modeling.

In order to contrast the output from the smoother with the input to the filter, see Fig. 4(a). This shows the raw data for the California clock, with the smoother results superimposed. A rate of $1.2E-13$ was removed from both plots. The smoother nicely bridges a gap in the data at about MJD 47050. During this gap, the standard deviations computed by the smoother increased from about 4 nanoseconds, on MJD 47046, to 40 nanoseconds on MJD 47052, then dropped back to 4.4 nanoseconds on MJD 47053. Other data gaps were also handled well. This ability of the filter/smoother to bridge data gaps will be important for operational uses. Figure 4(b) shows a superposition of the results produced by the JPL smoother and the NIST filtered estimates. Again, the $1.2E-13$ rate has been removed from both results. The agreement is quite good, with less than 8 nanoseconds difference. The consistent bias, with the JPL results slightly lower, may be a result of the way in which the space vehicle biases are handled.

The offset rate of the complex clocks is the most important parameter from the operational point of view. Presently, the rate is determined by taking the mean of all the GPS measurements for ten days, then doing a least-squares linear fit and using the slope of that fit as an estimate of the rate.

The results of these "hand calculations" are shown as dots in Fig. 5. Generally, there is good agreement between these results and those produced by the filter/smoother. A careful examination of the times at which the hand calculations differ noticeably from the smoother results revealed two reasons for these discrepancies. For example, in the vicinity of MJD 47060–47070, there was very little data for the clock in Spain. This caused the smoother to report a large uncertainty in its rates over this interval, and explains the difference between the smoother output and the hand calculation shown in Fig 5(c). The point obtained by the hand calculations for the clock in Australia on MJD 47070 was found, on closer examination, to have been calculated incorrectly. The correctly calculated value is very close to that produced by the smoother.

VII. Conclusions and Future Work

The UD form of the Kalman Filter appears to solve many of the problems encountered in the past with processing clock data on a routine basis. The numerical stability of the filter/smoother provides an accurate and reliable method for computing optimal estimates of the offsets, offset rates, and drift rates. The automatic data editing feature removes bad data points appropriately.

The first several days worth of the data that were used for this study seemed to be very poor. For this reason, data corresponding to the first four days were skipped. As a comparison, this same data was also processed by using a Square Root Information Filter (SRIF). The SRIF is an even more numerically reliable form of the Kalman Filter. This increased reliability, however, is attained at the cost of an increase in computational time. The SRIF was able to handle the bad data at the beginning of the data set with no problems. Since computer run-time is of little concern for this application, it may be advisable to replace the UD filter with a SRIF.

Possible improvements to the mathematical model used in the filter would be of interest. For example, it is clear that the assumption about the biases being constant is incorrect. An improved model would allow the biases to be time varying. The rate at which they should vary, however, would have to be determined by statistical parameter estimation techniques, such as maximum likelihood estimation. Also, in the current model, all clocks are treated as if they were identical. In fact there are three different types of clocks in the DSN. Special characteristics of the clocks could be taken into account to improve the accuracy of the model.

References

[1] G. J. Bierman, *Factorization Methods for Discrete Sequential Estimation*, New York: Academic Press, 1977.

[2] G. J. Bierman, "A Reformulation of the Rauch-Tung-Streibel Discrete Time Fixed Interval Smoother," *IEEE Conference on Decision and Control*, Austin, Texas, pp. 840–844, Dececember 1988.

[3] P. A. Clements, A. Kirk, and R. Unglaub, "Results of Using the Global Positioning System to Maintain the Time and Frequency Synchronization in the Deep Space Network," *Proceedings of the Eighteenth Annual Precise Time and Time Interval Applications and Planning Meeting*, pp. 395–405, December 1986, (also published in *TDA Progress Report 42-89*, Jet Propulsion Laboratory, Pasadena, California, pp. 67–72, May 15, 1987).

[4] D. A. Howe, "Progress Toward One-Nanosecond Two-way Time Transfer Accuracy Using Ku-Band Geostationary Satellites," *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, vol. UFFC-34, no. 6, pp. 639–646, November 1987.

[5] H. E. Rauch, F. Tung, and C. T. Striebel, "Maximum Likelihood Estimates of Linear Dynamic Systems," *AIAA Journal*, vol. 3, no. 8, pp. 1445–1450, August 1965.

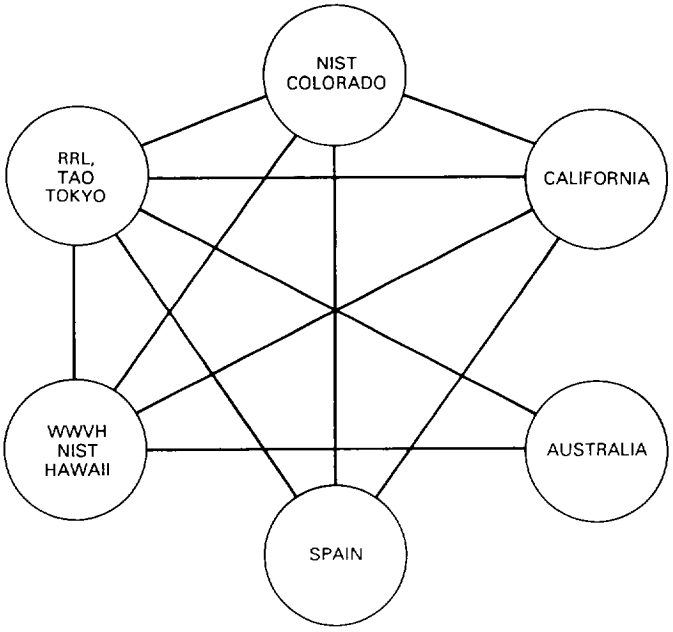

Fig. 1. Eleven possible mutual views used in the DSN time and frequency coordinates.

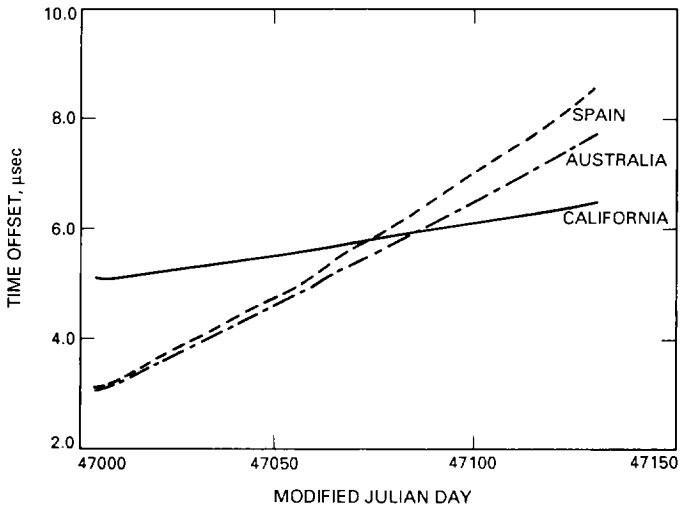

Fig. 2. Time-offset outputs of the UD smoother: DSN clock time minus VTL (NIST) clock time.

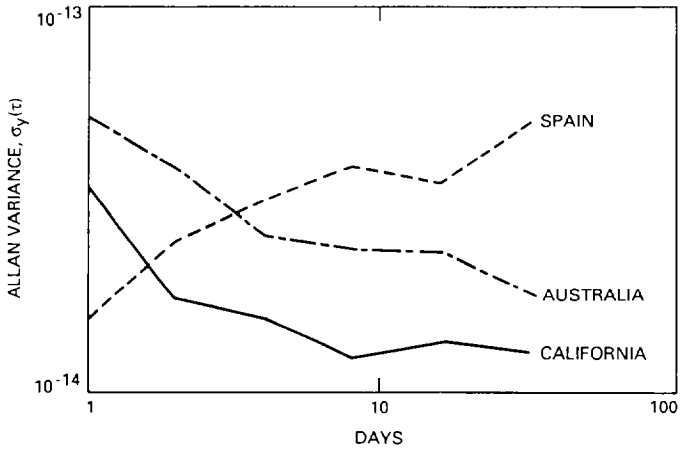

Fig. 3. The Allan variance of the time offsets from the UD smoother.

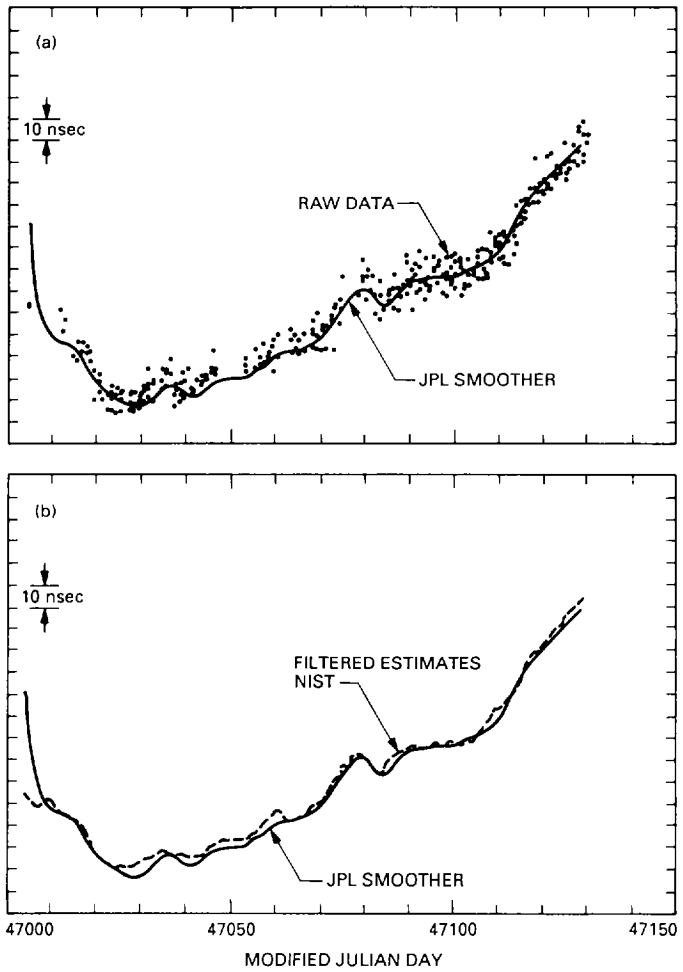

Fig. 4. Smoother output with a rate of 1.2×10^{-13} removed: (a) the smoothed estimate and the raw data; (b) the smoothed estimates compared to NIST estimates.

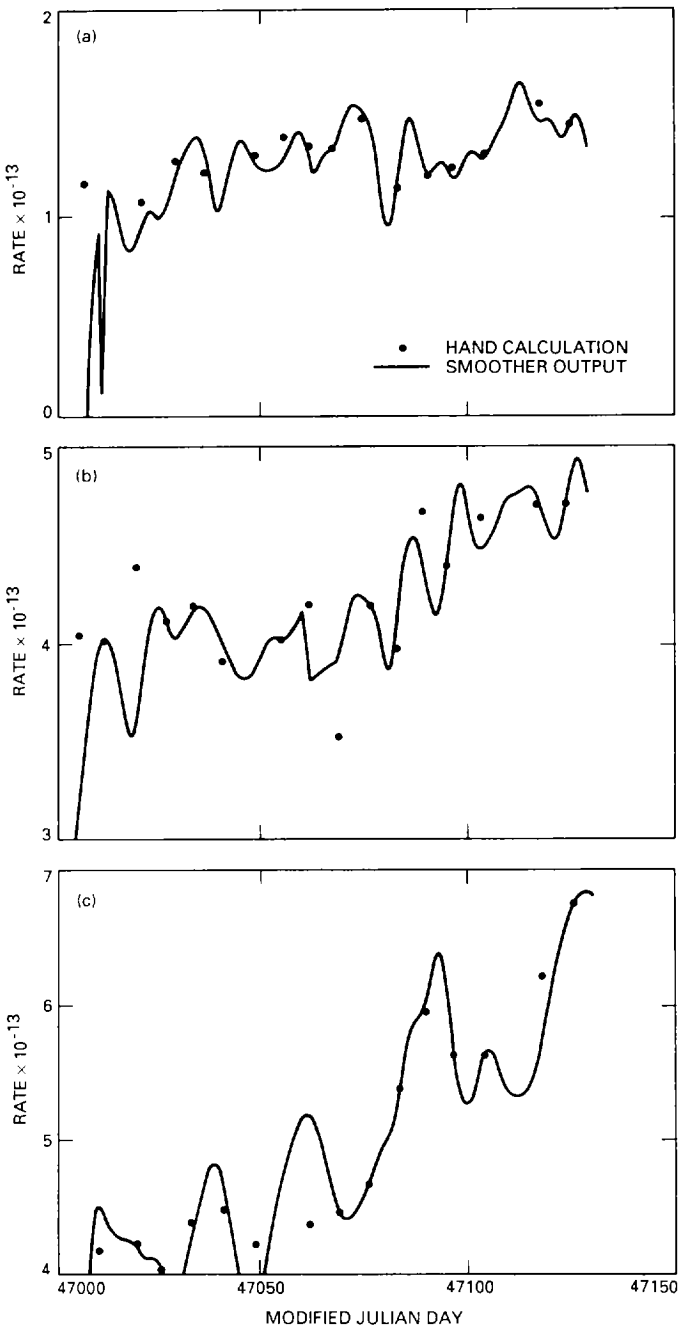

Fig. 5. Kalman smoother output and hand calculation of rate using a linear least-squares fit over 10 days of the time offset: (a) California, (b) Australia, and (c) Spain.