20 October, 1988

NOTES ON VARIANCE TESTING OF TIME SCALE ALGORITHMS AND CLOCK SETS

by

David W. Allan

The page at the end of this text (dated 15 June 1982) is taken from the Second International Atomic Time Scale Algorithm Symposium. This material is included in the proceedings of this third symposium because it seems to be more relevant now than it was then.

The idea of being able to test the quality of algorithms on real data is very appealing. Since the output of a time scale algorithm is typically designed to be better than the best clock in the system, with what do you compare? The enclosed note shows how we can estimate the variances of clock sets and compare the variances of ensembles generated by different algorithms, assuming independence of clock sets.

A few additional comments (cautions, guidelines and suggestions) are given here regarding the use of the enclosed technique:

1) The technique may be extended to more than three algorithms or more than three clock sets as in the N-cornered hat approach; hence, the final output would be an N by M matrix for each τ value. That is to say, one would obtain a $\sigma_y(\tau)$ plot for algorithm "A" operating on clock set "a" (Aa), then for Ab, etc. In the 4X4 case one would have a $\sigma_y(\tau)$ for each element of the following matrix, where A, B, C, & D are the M = 4 algorithms operating on N = 4 clock sets a, b, c, & d.

Aa	Ab	Ac	Ad
Ba	Bb	Bc	Bd
Ca	Cb	Cc	Cd
Da	Db	Dc	Dd.

By way of reminder the N-cornered hat equations are:

$$\sigma_i^2 \doteq \frac{1}{N-2} \left(\sum_{j=1}^{N} s_{ij}^2 - S \right) ; \; S = \frac{1}{N-1} \sum_{i<j}^{N} s_{ij}^2 ,$$

where s_{ij}^2 could be $\sigma_{y_{ij}}^2(\tau)$. The constraints on N and M are: $N \geq 3$ and obviously $M \geq 2$ for comparison. M can be greater than, equal to, or less than N within the above constraints. The ij elements can be taken from any row in the matrix. A different algorithm doesn't need to be operating on each clock set as in the enclosed note. That is to say, the same algorithm can operate on the N independent clock sets, and then be sorted in their stabilities with the N-cornered hat equations before going to another algorithm.

2) To the authors knowledge, the confidence of the estimate has not been worked out for the N-cornered hat approach. Some approximations have been developed some years ago at NBS, which we will be happy to share with those interested. One of the values of algorithms is to improve long-term stability. Unfortunately, for a finite data set, the confidence of the estimate of stability degrades with increasing τ. So exactly where we would like to look with the most interest, we have the poorest confidence. This problem can be solved with increased data length.

3) Given finite data lengths, it is not uncommon to have apparent correlations. These can lead to negative variances; this is a natural consequence of this kind of analysis. From the confidence of the estimate work previously done at NBS, there were some useful conclusions: if a negative variance occurs, its value can be assumed to be better than that of the minimum variance from any other member for that particular clock set and τ value; and the largest variance for a particular set and τ value has the best associated confidence.

4) The other main approach for evaluating algorithms is by using simulated data. This approach is only as good as the models. Kalman algorithms have not been made to model 1/f FM in a tractable marner yet. The white FM and random walk FM models, though very useful, do not cover the turf -- they leave a modeling hole. Primary frequency standards often exhibit 1/f FM in the long-term, for example, and little or no random walk FM. The variance approach outlined here-in is independent of the models -- dealing only with the real data.

5) The annual variations seen between and among time scales causes one to worry about the assumption of no correlations between clock sets -- at least in the long term. If this is suspected between some of the clock sets involved, a cross-correlation analysis might be in order before proceeding with this variance analysis approach.

CONCLUSION:

Given the increased importance of long-term stability performance of time scales, the variance analysis approach outlined here has more relevance now than it probably ever has. Some laboratories are now in a position with enough clock data, and with different kinds of algorithms to perform such tests. The above cautions should be exercised along with good common-sense judgement, if one proceeds with this approach. This approach may bear some significant fruit and useful insights.

15 June 1982

Note on Testing Time Scale Algorithms and Clock Sets

by

David W. Allan

The virtue of the method outlined below is that it is a test with real time data using the best clocks available; how-be-it, a perfect reference is not needed.

I will use as an example three algorithms under test at NBS: 1) "T" is a Kalman filter algorithm minimizing the time deviations; 2) "F" is a Kalman filter algorithm minimizing the frequency deviations; and 3) "G" is the official UTC(NBS) algorithm. Divide the clocks in the ensemble into three nominally equal groups: A, B, and C. Pick a concurrent time series of adequate length and let each algorithm operate on each clock set as configured below:

$$
\begin{tikzcd} & TA \arrow[dl, dash] \arrow[dr, dash] & \\ FB \arrow[rr, dash] & & GC \end{tikzcd}
\qquad
\begin{tikzcd} & TB \arrow[dl, dash] \arrow[dr, dash] & \\ FC \arrow[rr, dash] & & GA \end{tikzcd}
\qquad
\begin{tikzcd} & TC \arrow[dl, dash] \arrow[dr, dash] & \\ FA \arrow[rr, dash] & & GB \end{tikzcd}
$$

where, for example, "TA" denotes the "T" algorithm operating on the "A" clock set. We measure, of course, the time difference between algorithm-sets: e.g. "TA - FB".

On the assumption of independence one can write:

$$\sigma^2_{TA} = \frac{1}{2}\left[\sigma^2_{TA-FB} + \sigma^2_{TA-GC} - \sigma^2_{FB-GC}\right]$$

Hence, by permuting the algorithms and clock sets through this equation we can determine an estimate of the stability of each algorithm and its clock set. Suppose τ_o is the minimum sample interval, then one can calculate the stability of each of the algorithm clock sets at the corner of each triangle for any $\tau = n\tau_o$ up to that allowed by the data length, where n is an integer. Hence, one can determine for the nine algorithm-clock sets as follows:

$$
\begin{array}{ccc}
\sigma_{y\,TA}(\tau) & \sigma_{y\,TB}(\tau) & \sigma_{y\,TC}(\tau) \\
\sigma_{y\,FA}(\tau) & \sigma_{y\,FB}(\tau) & \sigma_{y\,FC}(\tau) \\
\sigma_{y\,GA}(\tau) & \sigma_{y\,GB}(\tau) & \sigma_{y\,GC}(\tau)
\end{array}
$$

From each of the three series taken by rows we can determine the stability ordering for all appropriate values of τ for each of the three clock sets via each algorithm.

From each of the three series taken by columns we can determine the stability ordering for all appropriate values of τ for each algorithm and each clock set.