

Promise into Practice: Implementing TA2 on Real Clocks at NIST

M. A. Weiss and T. Weissert

Time and Frequency Division, 847.5
National Institute of Standards and Technology
325 Broadway
Boulder, CO 80303, U.S.A.

Abstract

TA2 is a time scale algorithm designed to run as a post-processor by smoothing, i.e. combining estimates from runs in both forward and backward temporal directions. It has been tested using the time-scale data at NIST from 1989 through the first half of 1990. We review the characterization parameters for clocks with white FM, σ_w , and random-walk FM, σ_r , and the dependence of these estimates on integration time, τ . We also discuss the transformation of these parameters from their use as time deviations to frequency variances. We compare the scale generated by TA2 over 541 d starting at the beginning of 1989, with the two other scales at NIST, AT1 and TA. We find that TA2 and AT1 agree well in frequency, though TA, the pure Kalman-filter algorithm, walks off more than 2 parts in 10^{15} over the interval. Using GPS data to link with other time scales around the world, we compute the stabilities of the three NIST scales, AT1, TA, and TA2 using N-cornered hat estimates. We find that they have similar short-term stabilities for integration times out to 16 d, within the confidence of our estimates. TA appears to be worse in stability than the other two scales by a factor of about 2. There is some indication that TA2 improves slightly over AT1 in the long term, matching previous results from simulation, though the confidence on these estimates is poor.

Introduction

As we reported previously [1], TA2 is a smoother, an algorithm that runs both forward and backward in time. First we estimate frequency using the AT2 algorithm [2] in both directions. Then we combine these frequency estimates in a third (forward) pass to estimate time, using only the time estimate portions of AT2. We are now ready to run TA2 at NIST in parallel with the other scales, AT1 [2] and TA [3]. One of our goals with this algorithm is to improve long term frequency stability [4], using the advantage of post-processing with a smoother.

Please note that in this paper we will use the same name for a time scale algorithm and the time scale it produces. Thus we refer to TA2, TA, and AT1 both as algorithms and as the time scale output from a run on a particular ensemble of clocks over a particular interval. We trust that the context will keep these things clear.

AT2, the single-pass algorithm underlying TA2, is designed to generate a time scale in real time that is more stable than the best clock in the ensemble, both in long and short term. For each clock in the ensemble,

it adaptively estimates the clock's weight and a confidence on the frequency estimate. It detects frequency steps using the frequency estimate confidence [2].

We run TA2 over a period of 1 month plus 10 d and use only the data from the month as definitive results. Thus, all definitive results have the benefit of information from both the past and the future. The clock characterization parameters at the end of the month from the third pass are saved for initializing the forward pass for the next month. We run month-by-month over past data for two reasons. First, we plan to run TA2 operationally after the tenth day of each month for the previous month. Also, as was discussed previously [1], the random walk of the scale itself creates problems for combining the forward pass estimate of frequency with that of the backward pass, and these problems become more serious as the length of the pass grows. For the NIST clocks, we find that the amount the frequency of the scale walks off from the beginning of the month through the end and back again does not cause significant problems.

While installing TA2 in the NIST time-scale system, we encountered several problems concerning the parameters we use to characterize the clocks. The time interval between measurements is 2 h, as compared to the 1 d interval we had used in developing the algorithms for GPS time-transfer data. Also, in a real ensemble, clocks must enter and leave the scale at arbitrary times. Both of these factors made us aware of subtle considerations with our new algorithm, whose resolution we report in this paper.

We also present the results of running TA2 on the clocks at NIST over the period from January 1989 through June 1990. 1989 was chosen to provide a rigorous test of the scale because of a failure of the measurement system which occurred in July of that year. Both the time and frequency offsets of each clock in the ensemble had to be estimated to restart the time scales after this failure. Apparently we succeed in this test because there seems to be no evidence of the discontinuity in the measurements in the resultant scale.

To study the short- and long-term stability of TA2 from this run, we compare it with both the other local scales at NIST, AT1 and TA, and with scales at other international laboratories using GPS time transfer. AT1 is the real-time time scale at NIST used to calibrate various oscillators through the NIST two-frequency measurement system [6], and to generate frequency and time signals

electrically through a microphase stepper. Like all the scales at NIST, and other scales such as ALGOS [5], AT1 estimates time as a weighted average of the measured clock differences [2]. In AT1, the weights are adaptively estimated in an attempt to optimize short term stability. Frequency is estimated using exponential filters with time constants chosen differently for each clock to optimize long term stability. TA is a Kalman filter, run once a month as a post-processor to generate a scale over the previous month [6]. Recently, design problems have been discussed concerning the use of a pure Kalman filter as a time-scale algorithm [2]. We show results here comparing TA to AT1 and TA2 which suggest that TA has been walking off in frequency faster than AT1 and TA2. By comparing the scales with other international scales using an N-cornered hat analysis, we find that the stability of TA becomes significantly worse than AT1 and TA2 by a factor of 2 at an integration time of about 128 d. TA2 is slightly better than AT1 at 128 d and appears to improve even more at 256 d. The confidence on both of these points is poor enough that we perhaps should wait for more data to draw hard conclusions, but the improvement of TA2 over AT1 at 128 d matches closely results predicted from simulation [1], hence giving us more confidence in our results. We also note no degradation in short term stability even with good long term performance.

Anomalous Behavior Detection

The TA2 algorithm could be applied to generate an ensemble from any group of standards, such as voltage, time, or length, but the way it detects and responds to anomalous behavior is what makes it particularly adapted to clocks, and hence to generating a time scale. By characterizing anomalous behavior in another type of standard, and then designing and implementing an algorithm for detecting and responding to such behavior, we could customize this algorithm for application to a set of that type of standard.

The anomalous clock behavior with which we are concerned consists of time steps and frequency steps, which of course must be detected in the presence of the usual stochastic processes of the clocks. In addition, these stochastic processes can change. Defining these in a language independent of clocks, we would say that a time step is a change in the estimated value of the standard's offset from the ensemble, while a frequency step is a change in the estimated slope. In either case, the step is defined only when the magnitude significantly exceeds the changes due to the stochastic processes. This becomes complicated since we assume both white noise and random-walk deviations of the slope. Hence we must estimate the normal size of these stochastic deviations to detect and respond to steps. Finally, for clock steps, we assume the change in value is sudden and final. Though some mechanisms have been postulated for clocks for the gradual decay of a sudden change in slope (a frequency step) we assume no decay mechanism when presented with a particular frequency step since we use no information to correlate that step with a causing mechanism. It is very rare for a time step to decay. In other standards, such as standard chemical

voltage cells, one might want to include decay mechanisms in the anomalous behavior algorithm.

Clock Parameters in TA2

Clocks in TA2 are characterized with the same deterministic and stochastic parameters as TA [7]. The deterministic parameters are time, X , frequency, Y , and frequency drift offsets from the scale. The stochastic parameters are white and random-walk frequency modulation (FM), denoted σ_t and σ_f , respectively. These parameters represent the expected size of the noise impulses driving their respective noise process. Thus σ_t represents the expected size of the white noise impulses which are integrated to form the random walk process in Y . We estimate the values of these parameters as the expected amount of time deviation due to the particular noise process after a set time interval. In TA2, time and frequency offsets and white FM are estimated adaptively, whereas frequency drift and random-walk FM, both of which deal with long term phenomena, are set by the operator. In addition, the algorithm estimates a confidence, P , on Y , based on σ_t and σ_f , which responds to the adaptability of σ_t .

While there is provision in TA2 for using frequency drift parameters, in practice we rarely use them. The benefit of using frequency drift parameters should be to improve the long-term stability of the scale, since it should allow us to see the smaller random walk of clocks underneath the drift. The problem is that the drift of very few systems remains constant long enough to be useful. Also, drift in a cesium clock is a sign of pathology, in which case the clock probably will not last to contribute long term stability.

Let us discuss aspects of the stochastic parameters. Giving σ_t the correct units requires some care. We enter σ_t initially as the expected time deviation due to the white FM process after a time interval of $\tau_0 = 1$ d. We then use σ_t in the algorithm in different ways. σ_t is estimated adaptively in the algorithm from the predictability of the time offset, just as in AT1, assuming the measurement interval, τ , is short enough that the white FM process of the clock is dominant. The interval $\tau = 1/12$ d is different than τ_0 , the interval used in estimating σ_t for entry into the algorithm. Thus σ_t must be converted from the value as entered to initialize the adaptive, exponentially filtered estimate of σ_t .

Similarly, σ_f is entered as a time deviation due to random-walk FM at τ_0 . It is used internally as the expected frequency variation due to random-walk FM over the measurement interval. Thus it too must be converted. These conversions are shown in Table I, where, for the frequency variance, we have simply solved for $\sigma_f^2(\tau)$ with the proper power law dependence on τ .

For initial entry, σ_t and σ_f can be estimated [3] from the Allan variance $\sigma_y(\tau)$ as follows [8]:

$$\sigma_y^2(N\tau_0) = \frac{\sigma_a^2}{N\tau_0^2} + \frac{\sigma_b^2(2N^2+1)}{6N\tau_0^2} \quad (1)$$

This equation gives σ_a and σ_b as the time dispersion due to the specific noise types after the interval τ_0 . Please note that, in this paper, τ_0 is the time interval which applies to σ_a and σ_b when entered externally, while τ is the time interval between measurements. For TA2, σ_a and σ_b are entered estimated at $\tau_0 = 1$ d, while the time between measurements at NIST is $\tau = 1/12$ d.

We give an example of the use of equation 1 for estimating σ_a and σ_b from the Allan variance $\sigma_y^2(\tau)$ for a particular clock, the HP601 at NIST. Figure 1 gives the Allan deviation of the HP601 over the 541 d starting January 1989. Two lines are drawn on this plot, one for the white FM component of the variance with a slope of $-1/2$ and one for the random-walk FM component with a slope of $+1/2$. We have then marked that the intercepts of these lines at the one day point are $2.7 \cdot 10^{-14}$ for the white FM and $0.61 \cdot 10^{-14}$ for the random-walk FM. Applying equation (1) for each of σ_a with $\tau_0 = 0.864 \cdot 10^{14}$ ns we find $\sigma_a = 1.1$ ns. For σ_b we must realize that the line we have drawn is the asymptotic form of equation 1 for large N. In this case we have

$$\sigma_b^2|_{\tau} = 3 \cdot \tau \cdot \sigma_y^2(\tau) \quad (2)$$

Substituting the above values in (2) we have $\sigma_b = 1.6$ ns.

In steady-state TA2, we use our estimate σ_a in two different ways: for determining the relative weights of the clocks for the time update, and for filtering the white FM in our estimate of frequency. The value of σ_a is also involved in the anomalous behavior routines. Since the measurement interval is fixed for TA2 at NIST, the value of τ is not important in determining the relative weights of the clocks. However, in order to use σ_a as the estimate of the white FM level in filtering the frequency, we must transform it from being a time deviation to a frequency variance. σ_b must also be adjusted from its being time deviation at τ_0 as entered to its use as a frequency variance at τ . The relations we use for these are shown in Table 1.

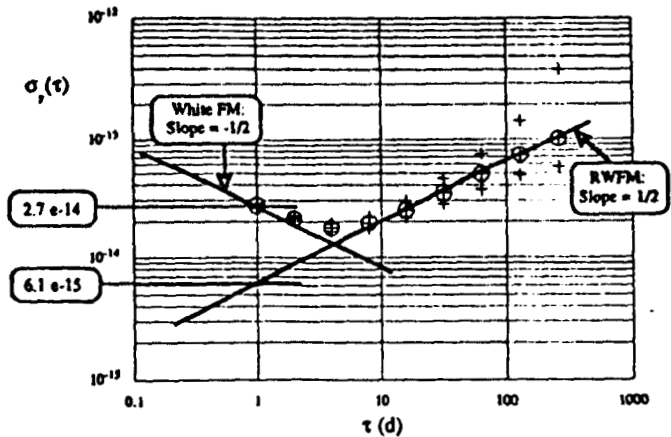


Figure 1. Allan deviation plot of the NIST clock HP601 over 541 d starting January 1989 showing white FM and random-walk FM components and levels.

A final note on the application of the stochastic clock parameters concerns sudden increases in σ_a initiated at certain times to avoid transients. There are four such times: 1) when starting a run for a new month based on estimates of parameters from the previous month, 2) when a clock is first entered as a new clock, 3) when a frequency step is found, and 4) when a clock goes off line and comes back on in one month. These increases in σ_a are necessary because the weights of the clocks in the time update are proportional to $1/\sigma_a^2$, thus it is necessary to increase σ_a to de-weight a clock. The factors we use to increase σ_a in these cases are purely phenomenological; we basically picked values that worked.

When starting a run for a new month, the estimates of σ_a^2 for clocks from the previous month are multiplied by a factor of 1.5, because the frequency estimates from the end of the last month are smoothed estimates based on data including 10 days into the current month. Hence because there is more

Table 1

	σ_a	σ_b
Entered as:	$\sigma_a _{\tau_0}$	$\sigma_b _{\tau_0}$
Used as: time variance	$\sigma_a^2 _{\tau} = \frac{\tau}{\tau_0} \cdot \sigma_a^2 _{\tau_0}$	not used
Used as: frequency variance	$\left(\frac{K}{\tau}\right)^2 \cdot \sigma_a^2 _{\tau}$	$\frac{K^2}{3\tau_0^2} \cdot \left(\frac{\tau}{\tau_0}\right) \cdot \sigma_b^2 _{\tau_0}$

K in this table determines the units for frequency. For TA2 we use ns/d internally, hence $K = 8.64 \cdot 10^{13}$ ns/d.

information implicitly contained in them than is available for the new first pass, these initial frequencies may not match the raw data well for the forward pass of the current month. The 1.5 factor increase of σ^2 compensates for this mismatch by allowing the algorithm a little extra time to learn the forward frequency, yet still starting each clock with the smoothed frequency from the previous month. The algorithm remembers the old and learns the new frequency of a clock using an exponential filter. Thus the old frequency is gradually forgotten with a "half-life" equal to the exponential filter time constant. This time constant adjusts adaptively with σ_n and should equal approximately the minimum of the Allan variance curve, the integration time when the dominant noise form changes from white FM to random-walk FM. The factor 1.5 adjustment at the beginning of a month additionally makes the definitive (third pass) estimates for the beginning of the month more influenced by the backward run with its longer foresight of information than the forward run.

When a clock is first entered as a new clock, even though one may have characterized its parameters well, we find that the scale still needs some time to fine tune its parameters. Toward this end, the initial value of σ^2 is multiplied by a factor of 3. Similarly, when a frequency step is detected, the scale is re-run with the stepped clock's weight set to 0 for a period long enough for the algorithm to learn its new frequency; we then re-admit the clock with non-zero weight and multiply the estimated σ_n^2 by a factor of 2.

When a clock goes off line and does not return before the end of the month, it then comes back on line in the reverse direction as a new clock. Alternatively, the clock can come back on line again in the forward direction before the end of the month. In either case, when a clock goes off line and comes back on two corrections are applied to σ_n^2 . We multiply by a factor of 2 when it returns because, even though we assume the white FM level of the clock to be nominally the same, we don't want to weight the clock in the same way for the time update since the frequency may have changed in unpredictable ways. We also add to σ_n^2 the amount of dispersion of the scale frequency during the off-line interval. If we denote σ_{rn} as the random-walk FM deviation of the scale frequency in ns/d at one day, and δ as the time interval for which the clock has been off line, then we add to σ_n^2 the value $\delta^3 \cdot \sigma_{rn}^2$, the time domain dispersion of the scale itself over the interval δ due to random-walk FM.

TA2 from January 1989 through June 1990

One of the features of the AT2 algorithm, inherited by TA2, is automatic frequency-step detection and response [1]. When the AT2 algorithm detects a frequency step, it removes that clock from the scale until the scale can learn its new frequency. Although this works automatically in TA2, we can also manually tell the scale when a clock has a step and what its magnitude is. This latter action, though more labor intensive, allows TA2 to keep the clock on line, thus providing a small improvement to the scale's stability. In this action we are using our

knowledge of clock behavior after the fact beyond what TA2 will do automatically. In the January 1989 - June 1990 run we estimated and inserted frequency steps by hand whenever the algorithm detected a step in a clock with a large amount of weight. Thus the results we obtained were through a lot of careful grooming of the ensemble.

Over the run, frequency steps were found in seven clocks: one step in two clocks, two steps in one clock, three steps in three clocks and six steps in one clock. Thus a total of nineteen frequency steps were found. There were four other clocks that had no steps, making a total of 11 clocks. All of the frequency steps were estimated and inserted manually except for two in one of the three step clocks and four in the six step clock.

In figure 2a we see the comparison TA2-TA run on the same clocks at NIST over the period MJD 47527-48068, all of 1989 and the first half of 1990. Figure 2b is the frequency plot of this same data. Clearly, there are some frequency excursions between the two scales. Comparing both of these

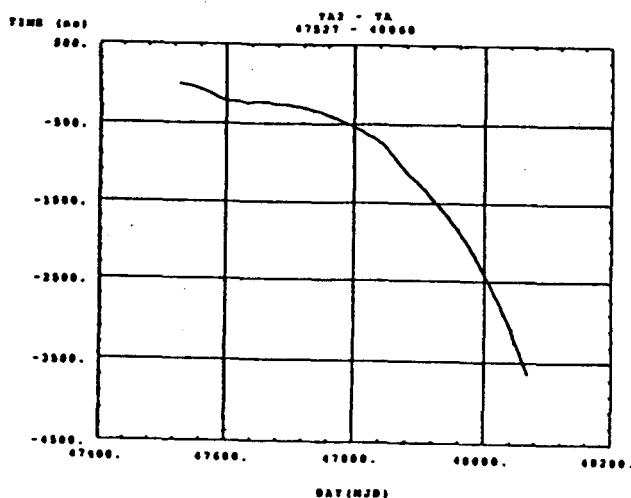


Figure 2a. The time deviation TA2 - TA from January 1989 through June 1990.

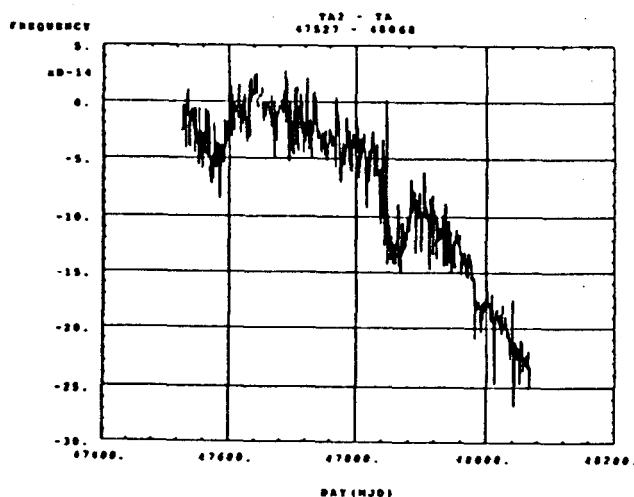


Figure 2b. The frequency deviation TA2 - TA from January 1989 through June 1990.

scales with AT1, TA-AT1 in figures 3a and 3b, and TA2-AT1 in figures 4a and 4b, we see that AT1 and TA2 agree quite well in frequency for the period. Thus we conclude with reasonable probability that the pure Kalman filter algorithm, TA, has walked off more than 2 parts in 10^{13} over this time period.

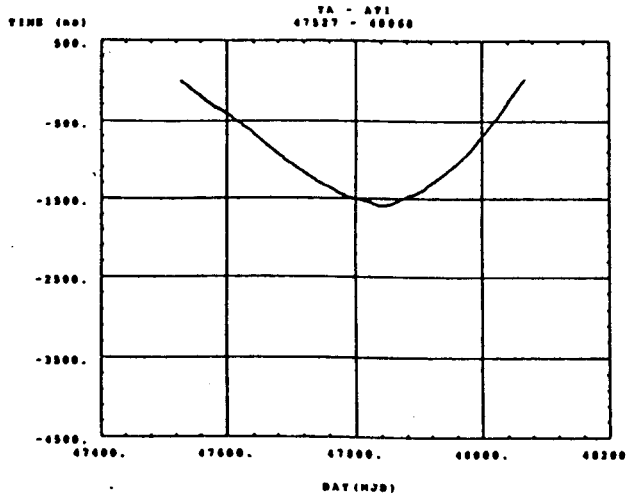


Figure 3a. The time deviation TA - AT1 from January 1989 through June 1990.

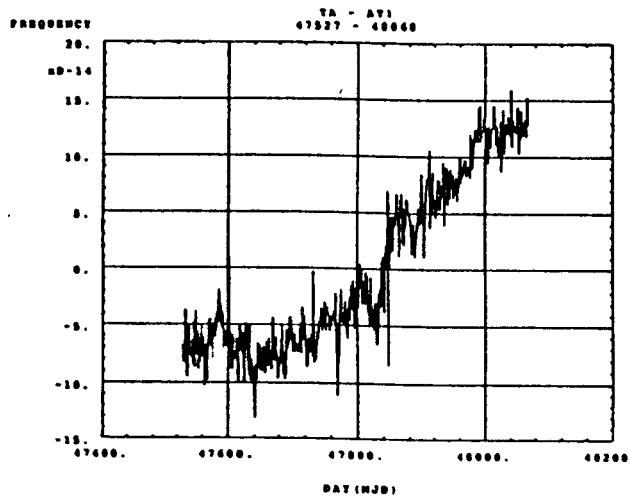


Figure 3b. The frequency deviation TA - AT1 from January 1989 through June 1990.

A second analysis of the results over the 541 d run was performed comparing the NIST time scales with UTC(lab) for several time standards laboratories around the world using GPS data. The laboratories used were the National Research Council (NRC), Ottawa, Canada, l'Observatoire de Paris (OP), Paris, France, the Tokyo Astronomical Observatory (TAO), Tokyo, Japan, the Technical University of Graz (TUG), Graz, Austria, and the United States Naval Observatory (USNO), Washington D.C., U.S.A. We did a type of separation of variance analysis called "N-cornered hat," [9] on all four labs plus each (independently) of AT1, TA and TA2.

The N-cornered hat technique requires that the input time series be independent, which is not true in our case for two reasons. First, since the labs were compared using

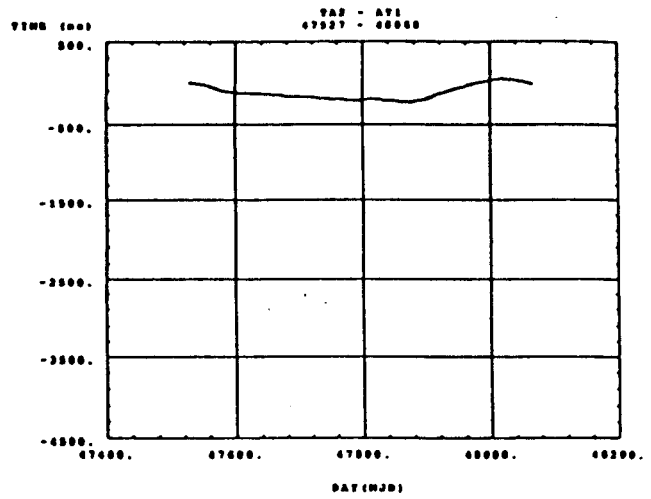


Figure 4a. The time deviation TA2 - AT1 from January 1989 through June 1990.

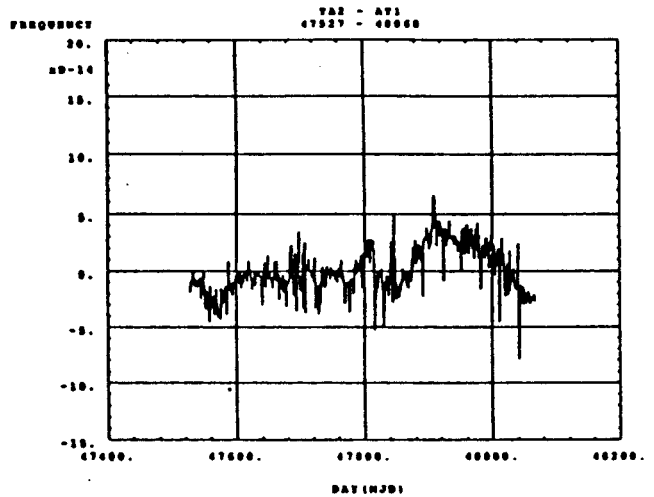


Figure 4b. The frequency deviation TA2 - AT1 from January 1989 through June 1990.

data from GPS satellites in common-view [8], there is the same satellite noise on several of the time transfer links. This would affect short term stability estimates out to perhaps a 4 d integration time. Further, each UTC(lab) is steered to match UTC as generated by the International Bureau of Weights and Measures. The steering, however, is usually slow and of small effect within a year. We know, however, that the USNO applied a lot of steering in 1989 and 1990; eight frequency steps were applied of magnitude 1 ns/d, the first six of which were applied in a series 20 d apart at the beginning of the year. We also note that there can be apparent correlations in the data due to the finite data length.

The results of our 5-cornered hat analysis on all four labs plus each of AT1, TA and TA2 are shown in figures 5, 6, and 7, respectively. We see little difference among the short term stabilities of the three different scales at NIST. We see a significant degradation in the long-term stability of TA over both AT1 and TA2, agreeing with our analysis above concerning TA. There is also a small improvement in TA2 over AT1 at 128 d, and an indication of

5 - Corner Hat
via GPS common view
MJD'S 47527 - 48068

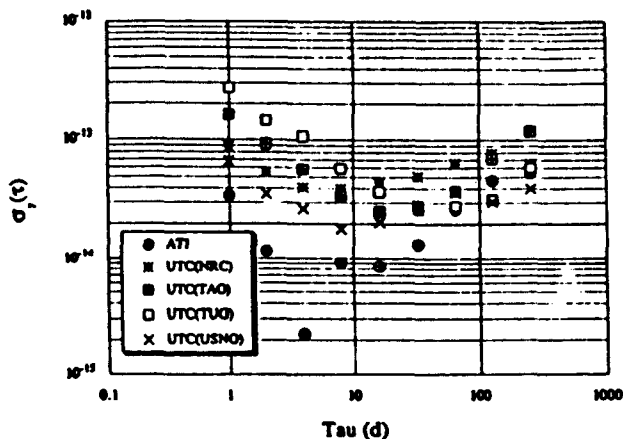


Figure 5. 5-cornered hat Allan deviation estimates for AT1 against UTC(NRC), UTC(TAO), UTC(TUG) and UTC(USNO) from January 1989 through June 1990.

5 - Corner Hat
via GPS common view
MJD'S 47527 - 48068

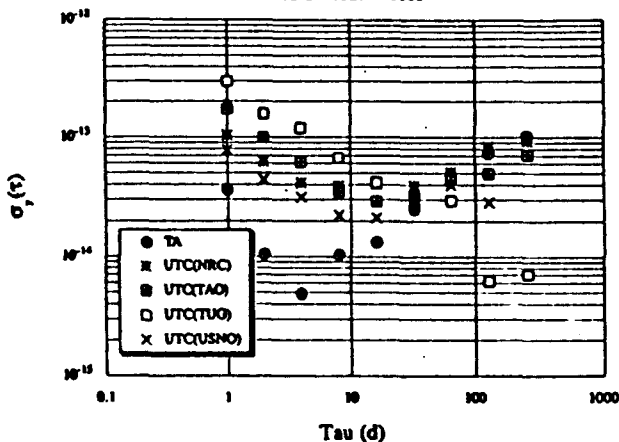


Figure 6. 5-cornered hat Allan deviation estimates for TA against UTC(NRC), UTC(TAO), UTC(TUG) and UTC(USNO) from January 1989 through June 1990.

5 - Corner Hat
via GPS common view
MJD'S 47527 - 48068

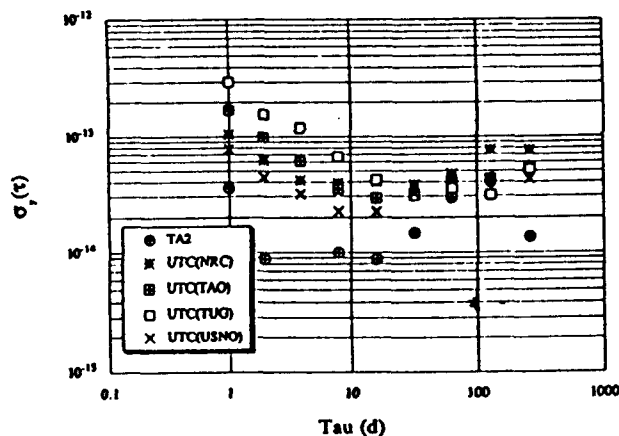


Figure 7. 5-cornered hat Allan deviation estimates for TA2 against UTC(NRC), UTC(TAO), UTC(TUG) and UTC(USNO) from January 1989 through June 1990.

improvement at 256 d, though we have little confidence in this last point. The improvement at 128 d is also probably not statistically significant by itself, but it agrees well with the improvement we expect to find from simulation [1]. It is not known how to estimate confidence bars on N-cornered hat estimates, but the confidence cannot be better than that of a direct Allan deviation computation. These error bars can be seen in figure 1. We conclude that we have some indication that we have realized here in practice the results we predicted in simulation.

Conclusions

We have presented the implementation of TA2 in the NIST time scale system. We discussed how to characterize clocks with white FM and random-walk FM levels, and various relations for these numbers as a function of integration time between treatment as a time deviation or frequency variance. We found little over-all frequency change between AT1 and TA2 over 1989 and the first half of 1990, but a walk in frequency in TA accumulating to more than 2 parts in 10^{13} . Finally, we studied the fractional frequency stability of the three scales in relation to UTC of various labs around the world using an N-cornered hat approach. We note that the stability of TA is worse than AT1 and TA2 by a factor of almost 2 at 128 d. We also conclude that there is indication of an improvement in the stability of TA2 over AT1 matching that predicted by former simulation, though the confidence is poorer than the apparent improvement.

Bibliography

[1] M. A. Weiss and T. Weissert, "TA2, A time scale algorithm for Post-Processing: AT1 plus frequency variance," IEEE Trans. Instrum. Meas., vol. 40, no. 2, April '91, to be published.

[2] M. A. Weiss and T. Weissert, "AT2, A new time scale algorithm: AT1 plus frequency variance," Metrologia, to be published.

[3] R. H. Jones and P. V. Tryon, "Estimating Time From Atomic Clocks," Journal of Research of the National Bureau of Standards, vol. 88, no. 1, pp. 17-24, Jan - Feb 1983.

[4] D. W. Allan, M. A. Weiss, T. K. Pepler, "In Search of the Best Clock," IEEE Trans. on Instrum. and Meas., vol. 38, no. 2, pp. 624-630.

[5] B. Guinot and C. Thomas, "The Establishment of International Atomic Time," Annual Report of the BIPM Time Section, Part D, pp. D1-D22, 1988.

[6] D. W. Allan, D. J. Glaze, J. E. Gray, R. H. Jones, J. Levine, and S. R. Stein, "Recent Improvements in the Atomic Time Scales of the National Bureau of Standards," Proc. of the 15th Annual Precise Time and Time Interval (PTTI) Applications and Planning Meeting, pp. 29-40, Naval Research Laboratory, Washington D. C., Dec. 6-8, 1983.

[7] P. V. Tryon and R. H. Jones, "Estimation of Parameters in Models for Cesium Beam Atomic Clocks," Journal of Research of the National Bureau of Standards, vol. 88, no. 1, pp. 3-16, Jan.-Feb. 1983.

[8] J. A. Barnes, R. H. Jones, P. V. Tryon, and D. W. Allan, "Stochastic Models for Atomic Clocks," NASA Conference Publication 2265-- Proc. of the 14th Annual Precise Time and Time Interval (PTI) Applications and Planning Meeting, pp. 295-306, 1982.

[9] J. Barnes, "Time Scale Algorithms Using Kalman Filters -- Insights from Simulation," Proc. 2nd Symposium on Atomic Time Scale Algorithms, NBS, Boulder, CO, June 23-25, 1982.