An Optimal Modification of a Kalman Filter for Time Scales

Charles A. Greenhall
Jet Propulsion Laboratory
California Institute of Technology

This work was carried out by the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration.

Outline

Motivation

Time scale generalities

Empirical stability results

Stage 1: Raw Kalman scale

Stage 2: Kalman plus weights

Covariance x-reduction

Stage 3: Reduced Kalman scale

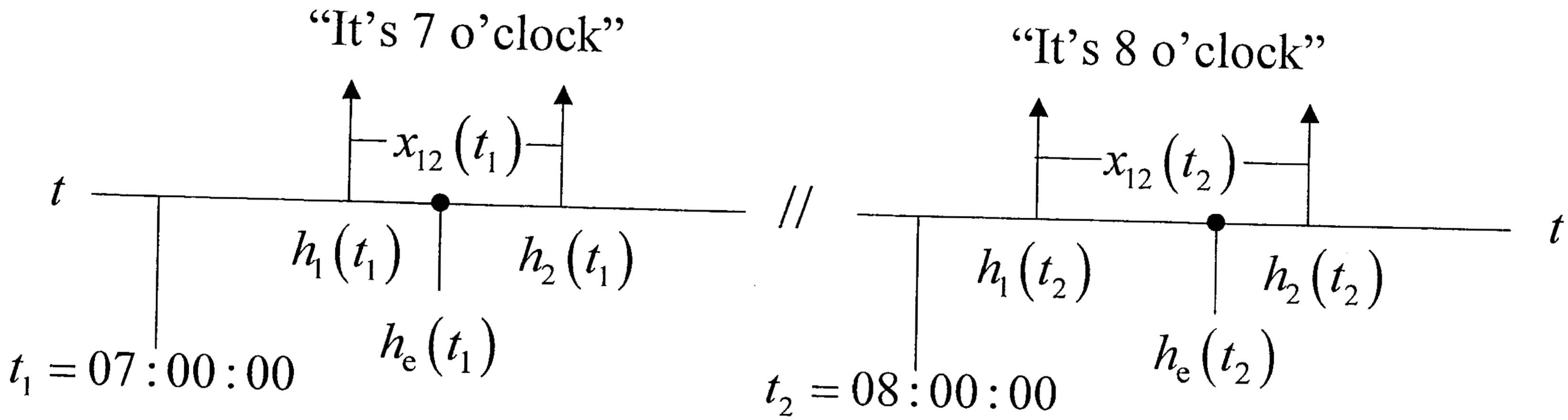
Motivation

TA(NIST) time scale based on Kalman filter (Jones & Tryon, 1983-).
Follows best long-term clock, regardless of short-term noise (Weiss & Weissert, 1989, 1991).

Goals of simulation study

- Reproduce behavior of this time scale.
- Understand it.
- Improve it.

What is a Time Scale?

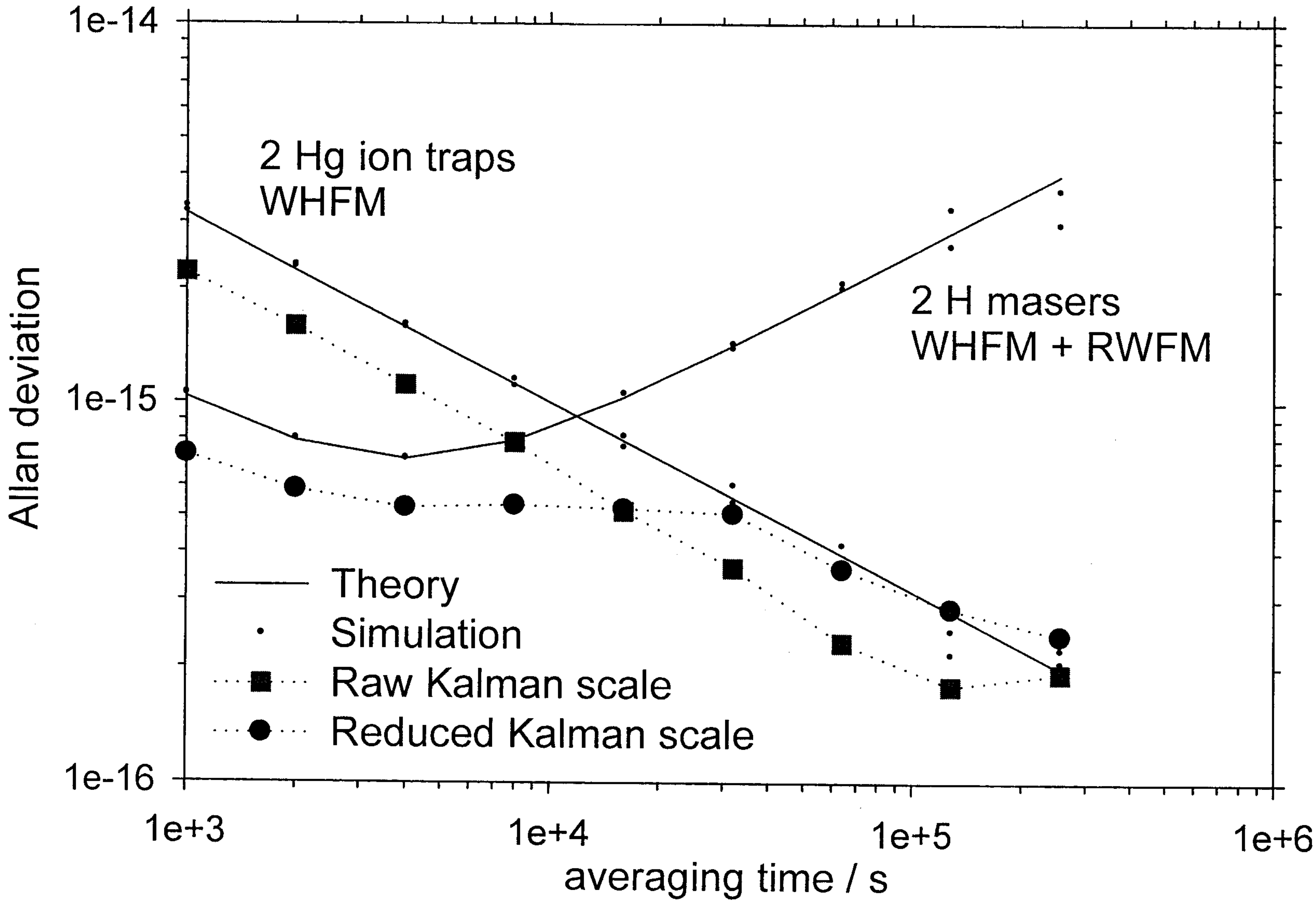


Time scale algorithm: Sequentially determines ensemble time $h_e(t)$ relative to $\{h_i(t)\}$, given the measurements $x_{ij}(t) = h_j(t) - h_i(t)$ at a sequence of dates t.

Phase (time) residual: $x_i(t) = t - h_i(t)$ (to be modeled).

Measurements: $x_{ij}(t) = x_i(t) - x_j(t)$.

Results for simulated 4-clock ensemble



Stage 1: Raw Kalman Scale (TA(NIST))

System state for n clocks: $X = [x_1, y_1, \ldots, x_n, y_n]$

$x_i =$ phase of clock i: WHFM + RWFM

$y_i =$ frequency **state** of clock i: RWFM only

Noiseless measurements at a sequence of dates t:

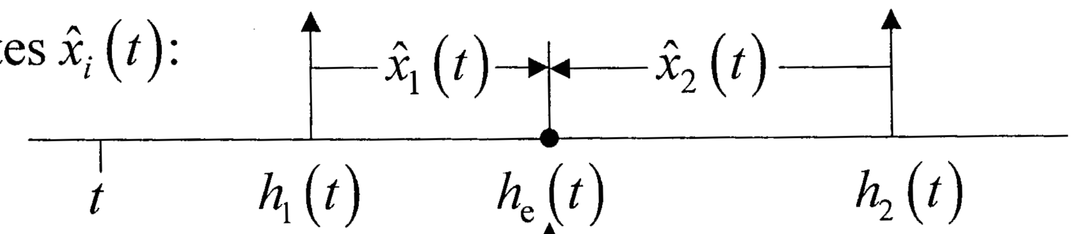
$$x_{ij}(t) = x_i(t) - x_j(t)$$

At measurement date t, Kalman filter produces:

Estimated state $\hat{X}(t) = [\hat{x}_1(t), \hat{y}_1(t), \ldots, \hat{x}_n(t), \hat{y}_n(t)]$;

Error covariance matrix $P(t) = \mathrm{cov}\left(X(t) - \hat{X}(t)\right)$.

Use phase estimates $\hat{x}_i(t)$:



Corrected clocks (Brown, 1991) coincide.

Raw Kalman scale = corrected clocks from unmodified Kalman filter.

Stage 2: Kalman Plus Weights

The Kalman filter gives good frequency state estimates $\hat{y}_i(t-\tau)$.

Neglect Kalman phase estimates, use frequency state estimates in a conventional Basic Time Scale Equation:

$$\Delta_\tau x_{\mathrm{e}}(t) = \sum_{i=1}^{n} \lambda_i(t)\left[\Delta_\tau x_i(t) - \tau\hat{y}_i(t-\tau)\right]$$

$x_{\mathrm{e}}(t)$: Weighted-average time scale based on Kalman frequency estimates.

Determine the weights $\lambda_i(t)$ ($\Sigma\lambda_i = 1$).

$\hat{y}_i(t-\tau)$ is a good lowpass-filtered estimate of $\Delta_\tau x_i(t)/\tau$.
Thus, each term in the BTSE has good long-term stability.
Choose weights to optimize short-term stability:

$$\lambda_i(t) \propto \frac{1}{\text{WHFM variance}_i} \quad \text{(approximately).}$$

The resulting time scale is called the **KPW scale**.

Covariance X-Reduction

P	x_1	y_1	x_2	y_2	x_3	y_3
x_1	*	*	*	*	*	*
y_1	*	*	*	*	*	*
x_2	*	*	*	*	*	*
y_2	*	*	*	*	*	*
x_3	*	*	*	*	*	*
y_3	*	*	*	*	*	*

$\longrightarrow$

P'	x_1	y_1	x_2	y_2	x_3	y_3
x_1	0	0	0	0	0	0
y_1	0	*	0	*	0	*
x_2	0	0	0	0	0	0
y_2	0	*	0	*	0	*
x_3	0	0	0	0	0	0
y_3	0	*	0	*	0	*

Kalman frequency state error variances empirically well-behaved; phase error variances diverge fast.

Theorem 1. Covariance x-reduction leaves future frequency estimates unchanged.

When forming KPW scale, we may use x-reduction to keep $P(t)$ from running away.

Stage 3: Reduced Kalman Scale

X-reduction does change future phase estimates. Try it: Surprise!
The corrected clocks become at least as stable as the KPW scale.

Reduced Kalman Scale

Run the Kalman filter on the ensemble model and measurements.
X-reduce the covariance matrix after each measurement.
Use the phase estimates to produce the corrected clocks.

Why does this work well?

Theorem 2 (Weiss, Allan & Peppler, 1989). The corrected clocks constitute a weighted-average scale based on Kalman frequency estimates, with implicit weights that depend only on the Kalman gain matrix.

Weights at end of simulation:

	λ_H	λ_{Hg}
Raw	0	0.50
KPW	0.45	0.05
Reduced	0.40	0.10

Optimality of the Reduced Kalman Scale

Theorem 3. Of all weighted-average scales based on Kalman frequency estimates, the reduced Kalman scale has the implicit weights that minimize the variance of the time scale increment

$$\Delta_\tau x_{\mathrm{e}}(t) = \sum_{i=1}^{n} \lambda_i(t)\left[\Delta_\tau x_i(t) - \tau \hat{y}_i(t-\tau)\right].$$

You don't have to solve for the weights; they are automatically implied by the x-reduced Kalman algorithm.

Final Remarks

The unmodified Kalman filter is an *accuracy* algorithm (Weiss & Weissert); it tends to use the clocks that provide the smallest long-term time deviations.

X-reduction turns the Kalman filter into a *stability* algorithm.

White PM and measurement noise are not included.

This study was in a simulation playpen; a practical time scale has to handle outliers, jumps, changes in the ensemble, etc., and provide for estimation of the noise variance coefficients.