PRACTICAL PERFORMANCE OF THE UTC(CH.R) REAL TIME REALIZATION OF UTC(CH) AND PROSPECTS FOR IMPROVEMENT

Laurent-Guy Bernier, Gregor Dudle

METAS - Swiss Federal Office of Metrology and Accreditation
Lindenweg 50, CH-3003 Bern-Wabern, Switzerland
e-mail: laurent-guy.bernier@metas.ch fax: +41-31-3233573

Keywords: timescales, steering, prediction methods

Abstract

UTC(CH.R) is a hardware real-time realization of UTC(CH) generated from the reference cesium clock by means of a micro-phase-stepper. The purpose of the steering algorithm is to keep UTC(CH.R) as close as possible to UTC(CH) by means of a time-frequency control loop which must be stable despite the delay of 1 d between the epoch of the last computed state and the epoch of the daily steering. The paper presents the practical performance of the UTC(CH.R) realization based on measurement data recorded from April 2002 to April 2003. The performance of the UTC(CH.R) steering algorithm is compared against simulation results that apply alternative steering algorithms to the same free running clock data. In conclusion plans to use a hydrogen maser instead of a cesium clock for the generation of a future, improved UTC(CH.R) are discussed.

1 Introduction

The METAS time & frequency metrology laboratory is an exception among NMI laboratories contributing to TAI and UTC in the sense that UTC(CH) is not a physical clock but a computed timescale. All the cesium clocks, including the reference clock (REF), are free running. The inputs to the time scale algorithm are the time differences between the reference clock and the other hardware clocks measured by means of a 5 MHz phase comparator. The output of the time scale algorithm are the daily computed states of each clock versus the time scales UTC(CH) and TA(CH). TA(CH) is a free running time scale while UTC(CH) is steered monthly to UTC. Both time scales are generated using the same time scale algorithm and are based on the same clock data. The advantage of a computed UTC(CH) proceeds from the fact that a time scale is more stable and more reliable than a single physical clock. The obvious disadvantage is that UTC(CH) is not readily available in real-time. Although it is always possible to perform both time and frequency calibration measurements against the REF clock and to refer the calibration results *a posteriori* to UTC(CH) by processing the clock data, it is more convenient to use a hardware clock which tracks UTC(CH) with a known uncertainty and to get instant results. Hence the development of UTC(CH.R) in our laboratory. UTC(CH.R) is a real-time realization of UTC(CH) generated from the REF clock by means of a micro-phase-stepper (MPS) and a daily steering algorithm. The difficulty in realizing the closed-loop steering control of UTC(CH.R) arises from the fact that, at the epoch of the steering, the most recent time scale data available is for epoch UTC 00:00 of the day before. This is illustrated on the timing diagram of Figure 1. The epoch variable is noted n in day units and refers to UTC 00:00 of the current MJD. We define the constant $u = 1$ d to simplify the notation. The steering is performed at hh:00 UTC every day. The epoch of the current daily steering is therefore noted $n + h$ with $h =$ hh/24 d. Every night at UTC 00:00 the local clocks data for the previous day becomes available but not the remote clocks data. This is why the *Autotime* system computes the timescale products only up to UTC 00:00 of the previous day. This leaves a whole day to solve the eventual remote file transfer problems and to collect the data from the remote clocks. However, the consequence is that the delay in the measurement arm of the control loop is $u + h$ i.e. more than one day. This configuration is problematic from the point of view of the stability of the control loop. In order to increase the bandwidth of the steering control loop and to decrease the rms deviation without falling into instability problems, we have opted for a predictive control loop, i.e. the steering error signal is based on the prediction of the steered clock state UTC(CH.R)-UTC(CH) from the epoch of last computation up to the epoch of the daily steering, i.e. a $u+h$ prediction interval. The closed loop predictive

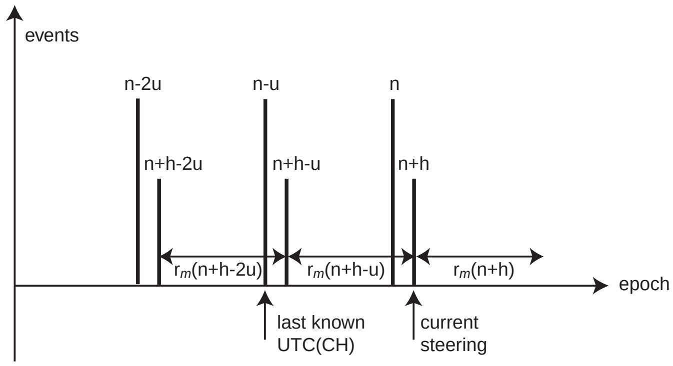

Figure 1: Timing diagram of UTC(CH.R) steering. Constant definitions: $u = 1$ d, $h =$ hh/24 d.

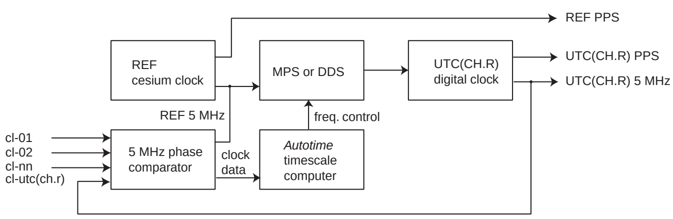

Figure 2: System block diagram of UTC(CH.R) steering.

steering algorithm was reported in [1]. The prediction algorithm itself was reported in [2] . Up to February 2002, UTC(CH.R) was based on a classical, non predictive, control loop reported in [1]. The disadvantage of the non predictive control loop was a very long closed loop time constant, necessary to avoid instability. Sudden changes such as frequency steps of the REF clock or steerings of UTC(CH) used to generate long transients. Our hope was that the new predictive algorithm would yield a stable control loop with a shorter time constant. The new steering algorithm was implemented into our in-house software package *Autotime* which is also used for the timescale generation. The state of the steered UTC(CH.R) clock is measured and recorded in the same way as the free running clocks, i.e. via a 5 MHz phase comparison system as shown of Figure 2. The system has been running continuously during most of 2002 and 2003. We are presently testing a new DDS (Direct Digital Synthesizer) to replace the MPS which is no longer available commercially. In this paper we present the experimental results obtained with UTC(CH.R) as recorded from April 2002 to April 2003. The observed experimental deviation of UTC(CH.R) versus UTC(CH) is discussed and compared with simulation results based on several alternative steering algorithms. Plans to use our new hydrogen maser instead of the REF clock for the generation of UTC(CH.R) are also discussed.

2 UTC(CH.R) Algorithm

We define r_m as the rate correction programmed into the MPS so that the input rate r_i is transformed into the output rate $r_o = r_i + r_m$. The steering algorithm has a feedforward term and a feedback term. The new rate correction $r_m(n+h)$ to be programmed into the MPS starting at the epoch of the current steering is given by

$$r_m(n+h) = -r_f(n-u, N_2) - \Delta r_m(n+h) \tag{1}$$

where the feedforward term $r_f(n-u, N_2)$ is the moving average filtered rate of the free running REF clock versus UTC(CH)

$$r_f(n-u, N_2) = \frac{x_f(n-u) - x_f(n-u-N_2)}{N_2}. \tag{2}$$

where N_2 is the averaging interval over which the frequency offset of the REF clock is estimated. The feedforward term corrects the frequency error of the REF clock versus UTC(CH). The proportional feedback term $\Delta r_m(n+h)$, on the other hand, is a small supplementary rate adjustment of the MPS that corrects the time error of the steered clock UTC(CH.R)

$$\Delta r_m(n+h) = \frac{\hat{x}_s(n-u+N_1)}{N_3}. \tag{3}$$

$\hat{x}_s(n-u+N_1)$ is the predicted time error of UTC(CH.R) projected from the last known epoch $n-u$ up to the epoch $n+h$ of the current steering which implies a prediction interval $N_1 = u+h$. $N_3 = 1$ d is the time constant for the correction of the time error. The prediction is performed using the simple prediction algorithm described in [1],

$$\hat{x}_s(n-u+N_1) = x_s(n-u) + \Delta \tag{4}$$

where

$$\Delta = h \times \Delta r_m(n+h-2u) + u \times \Delta r_m(n+h-u). \tag{5}$$

(4) and (5) can be described as follows. The prediction $\hat{x}_s(n-u+N_1)$ depends on the last known state $x_s(n-u)$ of the steered clock UTC(CH.R) extrapolated by the feedback rate correction $\Delta r_m(n+h-2u)$ that was applied at this epoch. After an interval h the steering of yesterday occurred and then the new feedback rate correction $\Delta r_m(n+h-u)$ was applied for a full day, until the current steering. Figure 3 shows the resulting UTC(CH.R)-UTC(CH) recorded daily from MJD 52381 to MJD 52755. The rms deviation from the mean is 10.5 ns and the mean offset is -9 ns. As discussed in the next section, the simulation of this algorithm predicts a smaller rms deviation and a negligible offset. Therefore it seems that there was a problem in the practical implementation of the algorithm. This will be discussed later on. Nevertheless the rms deviation, although not optimal, is quite good. Moreover, Figure 4 representing the Allan deviation of UTC(CH.R)

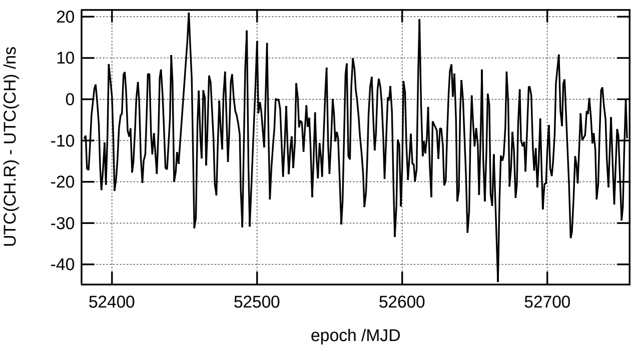

Figure 3: UTC(CH.R)-UTC(CH) from MJD 52381 to MJD 52755. Rms deviation from the mean: 10.5 ns. Offset: -9 ns.

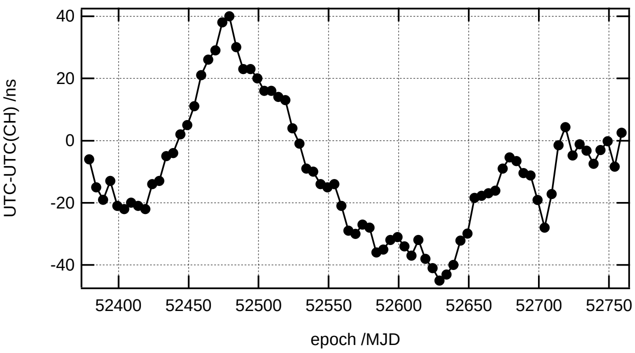

Figure 4: UTC-UTC(CH) from MJD 52379 to MJD 52759 as published by BIPM in Circular T.

shows that the closed-loop time constant is approximately 2 d. This is remarkable for a stable control loop with a delay of more than one day in the measurement branch. Figure 4 shows UTC-UTC(CH), as published by BIPM in Circular T, for the same period. Assuming a 20 ns uncertainty (1 sigma) of UTC(CH.R) vs UTC(CH) which includes both the noise and the bias, and assuming an uncertainty of 20 ns (1 sigma) uncertainty of UTC(CH) vs UTC, i.e. half the peak to peak variation during this period, we can state that the uncertainty of UTC(CH.R) as a real-time realization of UTC via UTC(CH) is approximately 30 ns (1 sigma).

3 UTC(CH.R) Simulations

In order to understand better the behavior of the steering control loop and to optimize the parameters, we have recently performed a simulation of the UTC(CH.R) steering system. The simulation is based on the actual, recorded, experimental free running REF clock data. For the current day, the simulated state $x_s(n)$ of the steered clock UTC(CH.R) is computed as

$$x_s(n) = x_s(n-u) + \Delta_1 + \Delta_2 \tag{6}$$

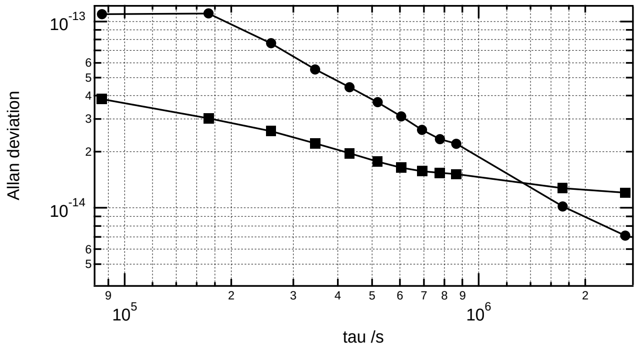

Figure 5: Allan deviation of the steered clock UTC(CH.R) (circles) and of the free running REF clock (squares).

where

$$\Delta_1 = x_f(n) - x_f(n-u) \tag{7}$$

is the recorded daily increment of the free running REF clock and where

$$\Delta_2 = h \times r_m(n+h-2u) + u \times r_m(n+h-u) \tag{8}$$

represents the simulated action of the MPS which is split into two periods: before and after the steering that occurred at epoch $n+h-u$. The current rate correction $r_m(n+h)$ is computed using the algorithm variant selected for a given simulation run. Variant A is a feedforward feedback predictive algorithm, in principle identical to the one implemented in *Autotime*. In the simulation (1), (2) and (3) apply. The prediction is given by

$$\hat{x}_s(n-u+N_1) = x_s(n-u) + r_s \times N_1 + c \tag{9}$$

instead of (4), where

$$r_s = r_f(n-u, N_2) + r_m(n+h-2u). \tag{10}$$

(9) and (10) express the fact that the prediction starts from epoch $n-u$ which implies that the MPS rate correction applied was $r_m(n+h-2u)$. However, the steering of epoch $n+h-u$ occurred during the prediction interval. Therefore the prediction must be corrected by a term

$$c = u \times [r_m(n+h-u) - r_m(n+h-2u)]. \tag{11}$$

which represents the effect of the correction rate change over the last 24 h up to the epoch of the current steering.

On the other hand, variant B is a feedback only predictive algorithm with integrator where $r_m(n)$ is given by

$$r_m(n+h) = r_m(n+h-u) - \Delta r_m(n+h) \tag{12}$$

instead of (1). (12) is the recursive definition of an integrator since $-\Delta r_m(n+u)$ is the very last correction term while $r_m(n+h-u)$ is the sum of all the previous correction terms. Table 1 summarizes the results of the steering control algorithm simulations. In all simulations the steering

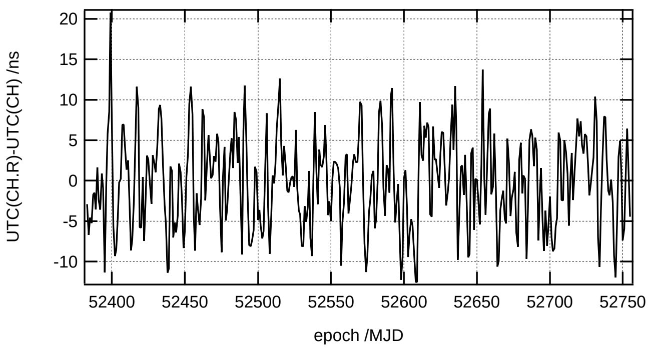

Figure 6: Simulation of the steered clock UTC(CH.R) using the steering control algorithm, variant A, $N_1 = 1.16$ d, $N_2 = 15$ d, $N_3 = 0.8$ d, rms deviation from UTC(CH) = 5.7 ns.

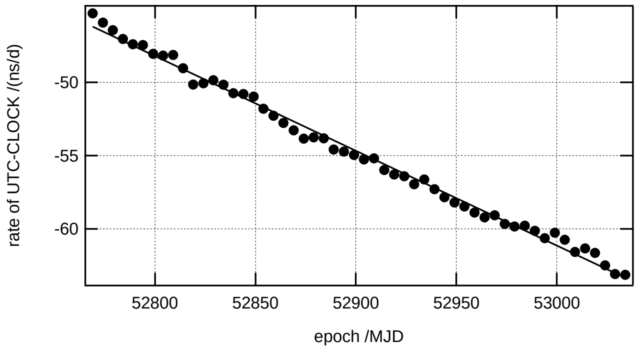

Figure 7: Daily rate of METAS hydrogen maser (BIPM ID 1405701) recorded from MJD 52769 to MJD 53034. Drift coefficient is -0.0647 ns/d^2 i.e. -7.5×10^{-16} /d.

sim	algo	N_1/d	N_2/d	N_3/d	rms/ns
1	A	1.16	5	1	6.3
2	A	1.16	15	1	5.8
3	A	1.16	15	0.8	5.7
4	A	2.16	15	2	6.6
5	B	2.16	15	5	8.2

Table 1: Simulations of UTC(CH.R) with REF clock.

is scheduled for 04:00 UTC therefore $h = 0.16$ d. Simulation 1 corresponds to variant A with the parameters of the *Autotime* algorithm. The simulation predicts a rms deviation of 6.3 ns and no bias. This is to be compared with the experimental result of Figure 3. Simulation 2 shows that by taking the optimal averaging interval $N_2 = 15$ d, i.e. minimizing the prediction error according to the criteria of the GSF-1 prediction [1], instead of 5 d, the rms deviation can be improved. Simulation 3 shows that reducing further the correction time constant N_3 improves a little more the rms deviation. If N_3 is further reduced the loop becomes unstable. The simulation with variant A and the optimal set of parameters is illustrated on Figure 6. Simulation 4 shows that if the prediction interval is increased so that the control loop tries to cancel the error predicted for the epoch of tomorrow's steering, $N_1 = 2u + h$, instead of today's steering, $N_1 = u + h$, and if the correction time constant is optimized for that condition, the rms deviation degrades. Simulation 5, finally, shows that the pure feedback variant B of the algorithm is not as good as the combined feedforward feedback variant A. Note again that both variants are predictive. A classical control loop without prediction would be unstable with comparable values of the correction time constant. In conclusion we see that the original algorithm, i.e. variant A, is the best and we see that the original parameters are quite close to the optimum so that only a small improvement can be obtained with optimization. The question that remains open is why the *Autotime* algorithm yields a rms deviation twice as large as the value predicted by the simulation with the algorithm variant A and why there is a bias. The explanation is as follows. Equations (4) and (5) which describes the prediction algorithm implemented into the *Autotime* code is an over-simplification of the steering algorithm variant A. It assumes implicitly that the moving average frequency of the REF clock does not change from day to day. More precisely, equations (4) and (5) are equivalent to (9), (10) and (11) only if

$$r_f(n-u, N_2) = r_f(n-2u, N_2) = r_f(n-3u, N_2) \quad (13)$$

which is true only in first approximation. The result was an *Autotime* steering algorithm that was functional but not optimal.

4 UTC(CH.R) from H-maser

We are looking forward to implementing the steering algorithm variant A in practice, instead of the original *Autotime* algorithm, and to verifying experimentally the improvement in performance predicted by the simulations reported above. Moreover, we now have a hydrogen maser that can be used as the free running REF clock and a DDS that can replace the late MPS for the generation of UTC(CH.R). Figure 7 shows the daily rate of our model EFOS-C hydrogen maser manufactured by Neuchâtel Observatory. This maser does not have an Automatic Cavity Tuning (ACT) system. However, the drift which is at the level of $+7.5 \times 10^{-16}$/d, as shown on Figure 7, is extremely stable. This makes the clock state highly predictable. Figure 8 represents the drift removed residual (the value of UTC-CLOCK in ns as published by BIPM, minus the best fit parabola) which varies in a range of only [-15 ns, + 15 ns] for a recording interval of 265 d. A simulation of the UTC(CH.R) generation was performed with the steering algorithm variant C and the free running maser data CLOCK-UTC(CH) used as the REF clock. Variant C of the algorithm is the same as variant A except for the addition of a drift parameter in the clock state prediction. Because of the drift the GSF-1 prediction algorithm of [1]

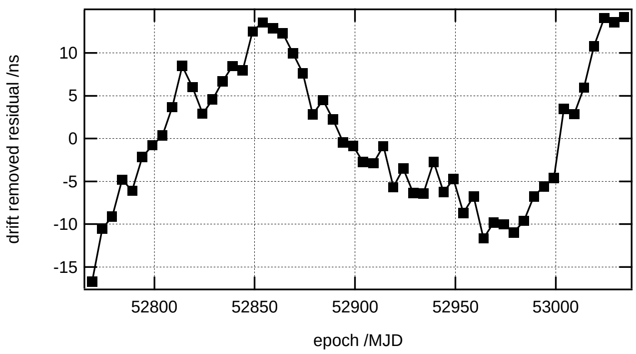

Figure 8: Drift removed residual of METAS hydrogen maser (BIPM ID 1405701) computed from MJD 52769 to MJD 53034. Residual in ns is given by (UTC-CLOCK) minus best fit parabola computed over recorded interval of 265 d.

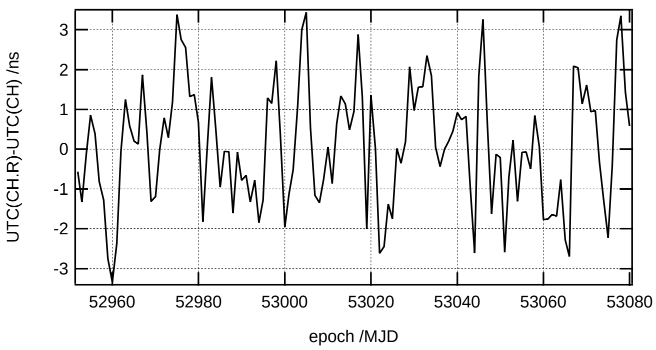

Figure 9: Simulation of steered clock UTC(CH.R) using algorithm variant C and the METAS hydrogen maser (BIPM ID 1405701) as the reference. Parameters are $N_1 = 1.16$ d, $d = 0.026$ ns/d^2, $N_2 = 20$ d, $N_3 = 0.8$ d, rms deviation from UTC(CH) = 1.9 ns.

has to be replaced by the DGSF-1 prediction algorithm of [3] which optimizes not only the frequency averaging interval N_2 but also a drift parameter d, which is half the drift coefficient. In variant C, (9) becomes

$$\hat{x}_s(n-u+N_1) = x_s(n-u)+r_sN_1+\frac{1}{2}dN_1^2(1+\frac{N_2}{N_1})+c \quad (14)$$

with r_s and c defined by (10) and (11) as before. The best results are obtained with the optimum clock prediction parameters $N_2 = 20$ d and $d = 0.026$ ns/d^2 and the same optimal control loop parameters as before, i.e. $N_1 = 1.16$ d and $N_3 = 0.8$ d. The result is an rms deviation from UTC(CH) of only 1.9 ns as shown on Figure 9.

5 Conclusion

In this paper we have reported that the predictive steering control loop implemented in the *Autotime* software in April 2002 and operated for more than a year has yielded a UTC(CH.R)-UTC(CH) rms deviation from the mean of 10.5 ns, with a bias of -9 ns. This version of the steering algorithm has demonstrated that it is possible to implement a fast control loop, with a time constant of 2 d, despite the delay of more than 1 d in the measurement arm of the control loop. The recent acquisition of a hydrogen maser and of a DDS to replace the late MPS opens new possibilities to improve the generation of UTC(CH.R). Moreover, simulations show that the *Autotime* steering control algorithm was too simple to reach optimum results. Simulations with variant A of the steering control algorithm (both feedforward and feedback terms, optimized prediction based on the moving average of the rate of the REF clock) show that an rms deviation of 6 ns with no bias could be reached using the same cesium clock as the reference. This is an expected improvement by a factor of two with respect to the *Autotime* present implementation. Simulations with variant C of the steering control algorithm (both feedforward and feedback terms, optimized prediction based on the moving average of the rate of the REF clock plus a drift parameter) show that an rms deviation of 2 ns with no bias could be reached using an hydrogen maser as the reference. This is an expected improvement by a factor of five with respect to the *Autotime* present implementation. We plan to test experimentally the combination hydrogen maser, DDS and variant C of the algorithm in the near future.

References

[1] L. G. Bernier, "Experimentation at METAS with a Simple Steering Algorithm based on Linear Prediction," in *Proc. IVth International Time Scale Algorithms Symposium*, BIPM, Sèvres, March 18-29, 2002.

[2] L. G. Bernier, "Use of the Allan Deviation and Linear Prediction for the Determination of the Uncertainty on Time Calibrations Against Predicted Timescales," in *IEEE Trans. on Instrum. and Meas.*, vol. 52, no. 2, pp. 483–486, April 2003.

[3] L. G. Bernier, "Application of the GSF-1 Algorithm to the Near-Optimal Timescale Prediction of the Hydrogen Maser," in *Proc. PTTI 2003*, San-Diego, December 2003.