Resilient Picosecond-Range Network Clock Ensemble Running in a White Rabbit Network

M.A. Weiss[1], R. González Escudero[2], S. Kannoth Mavila[2], F.M.J. Cozijn[2], K. Steinebach[2], K.S.E. Eikema[2], E.F. Dierikx[3], Y. Xie[3], S. Klemann[4], P. Klop[4], S. Sarkar[5], N.J. van Druten[5], F. Schreck[5], J.C.J. Koelemeij[2]

1 Marc Weiss Consulting LLC, Aptos, CA, United States of America.
2 Vrije Universiteit Amsterdam, Amsterdam, The Netherlands.
3 VSL, Delft, The Netherlands.
4 SURF, Utrecht, The Netherlands.
5 University of Amsterdam, Amsterdam, The Netherlands.

Research carried out in collaboration with/with support from:

VU VRIJE UNIVERSITEIT AMSTERDAM
VSL
SURF
UNIVERSITY OF AMSTERDAM
THE FRAMEWORK PROGRAMME FOR RESEARCH AND INNOVATION HORIZON 2020
AQuRA
Quantum Delta NL

Motivation for this project

Technologies in need of picosecond (ps)-range time synchronization:

- Terrestrial back-up and augmentation systems for positioning at the decimeter-to-centimeter level in urban environments, e.g. *terrestrial positioning system with 10 cm uncertainty based on 0.1 ns time through networks**
- radio astronomy through aperture synthesis¶
- quantum communication, e.g. future Quantum Networks, require *photon emissions be timed within 0.1 nanosecond*§
- improving the realization and dissemination of UTC itself from the nanosecond range to the picosecond range

Nanosecond-range time errors are inherent to current space-based time transfer technology as a fundamental limitation

White Rabbit (WR) is one possible solution

- WR residual timing errors can be as small as 0.1 ns = 100 ps
- WR timing errors can be even a few tens of ps if asymmetrical delays are adequately characterized

* Koelemeij et al., *Nature* **611,** 473 (2022)

¶ Boven et al. *Exp Astron* **61**, 3 (2026). https://doi.org/10.1007/s10686-025-10038-4

§ Moehring et al., *Nature* **449,** 68 (2007); Stolk et al., *PRX Quantum* **3,** 020359 (2022); *Sci. Adv.* **10,** eadp6442 (2024)

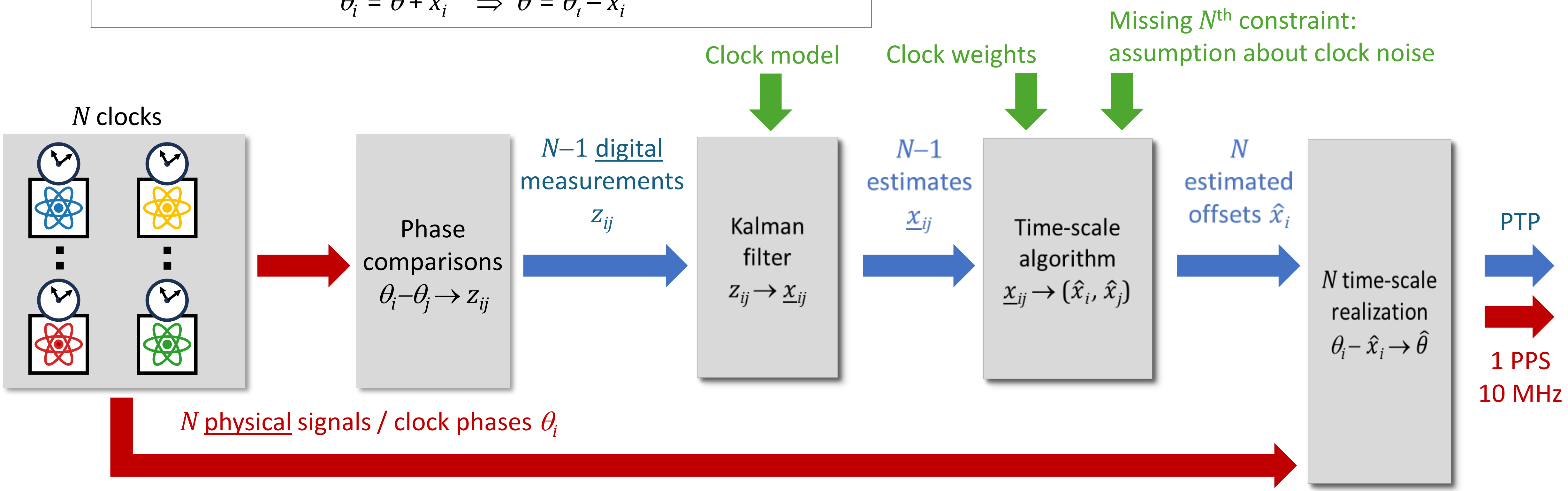
Time-scale algorithm

Stein's time-scale algorithm

S.R. Stein, 'Time Scales Demystified', Proc. 2003 IFCS/17th EFTF, p. 223 (2003).

Clock time (or phase) = time scale + clock-dependent correction

$$\theta_i = \theta + x_i \quad \Rightarrow \quad \theta = \theta_i - x_i$$



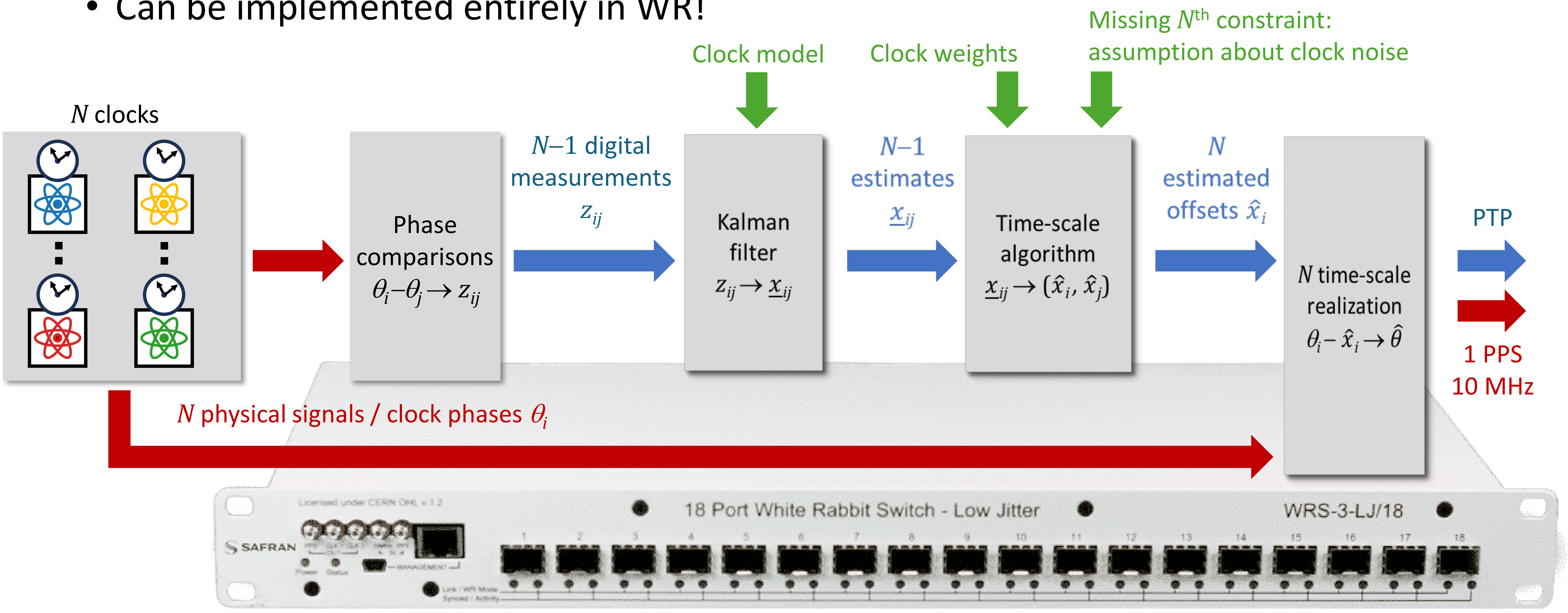
Result: from each clock signal the time scale can be realized using

$$\hat{\theta} = \theta_i - \hat{x}_i$$

Stein's time-scale algorithm

S.R. Stein, 'Time Scales Demystified', Proc. 2003 IFCS/17th EFTF, p. 223 (2003).

- Can be implemented entirely in WR!



Why bother?

One could hope to achieve:

- A time scale that is more stable than the individual clocks
- More clocks $\Rightarrow$ better time-scale stability $\Rightarrow$ more faithful representation of 'true time'
- WR: clocks in remote locations realizing same time scale to within << 1 ns ?
  - **BUT this requires a modification of Stein's algorithm to avoid nanosecond errors between time scale**
    [R. González Escudero et al., submitted]

Proceedings of the 2003 IEEE International Frequency Control Symposium and PDA Exhibition Jointly with the 17th European Frequency and Time Forum

Time Scales Demystified

S.R. Stein
Timing Solutions Corporation
Boulder, CO

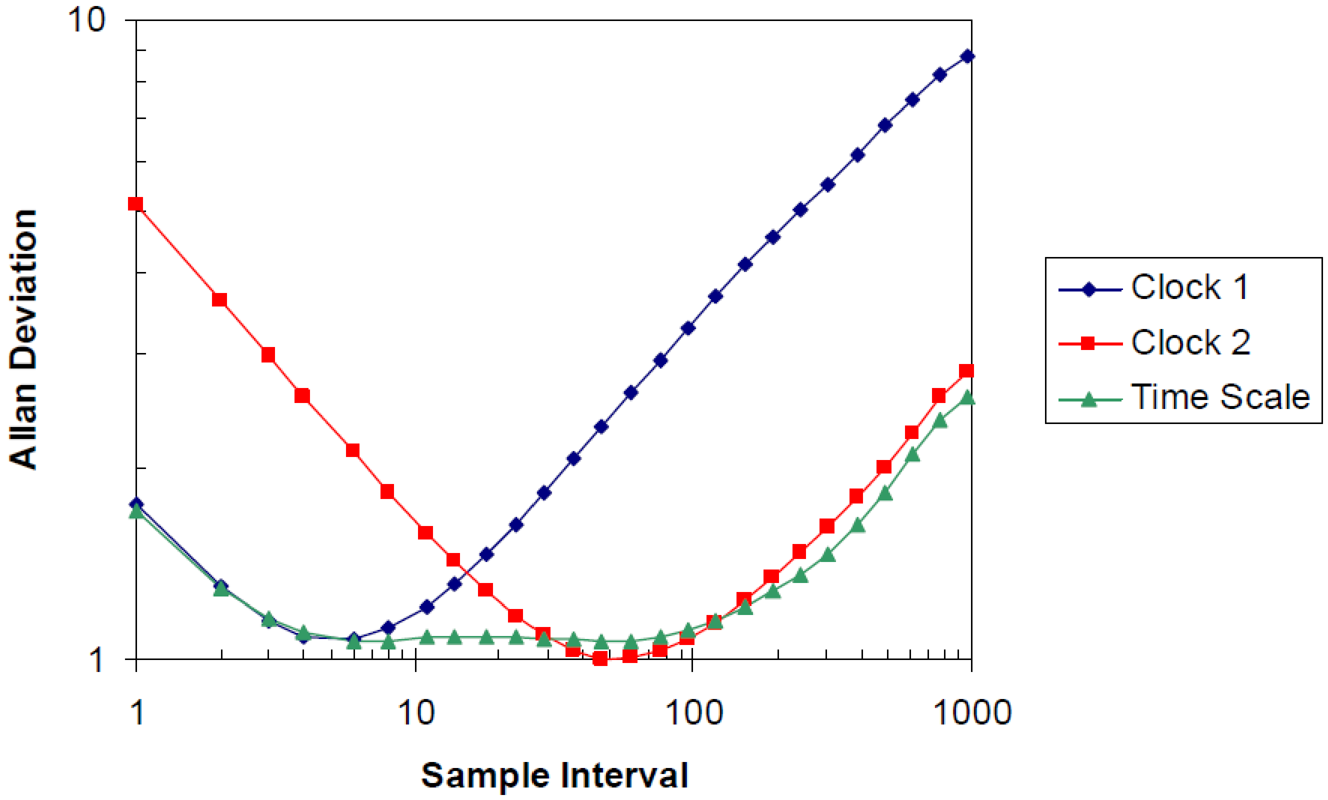
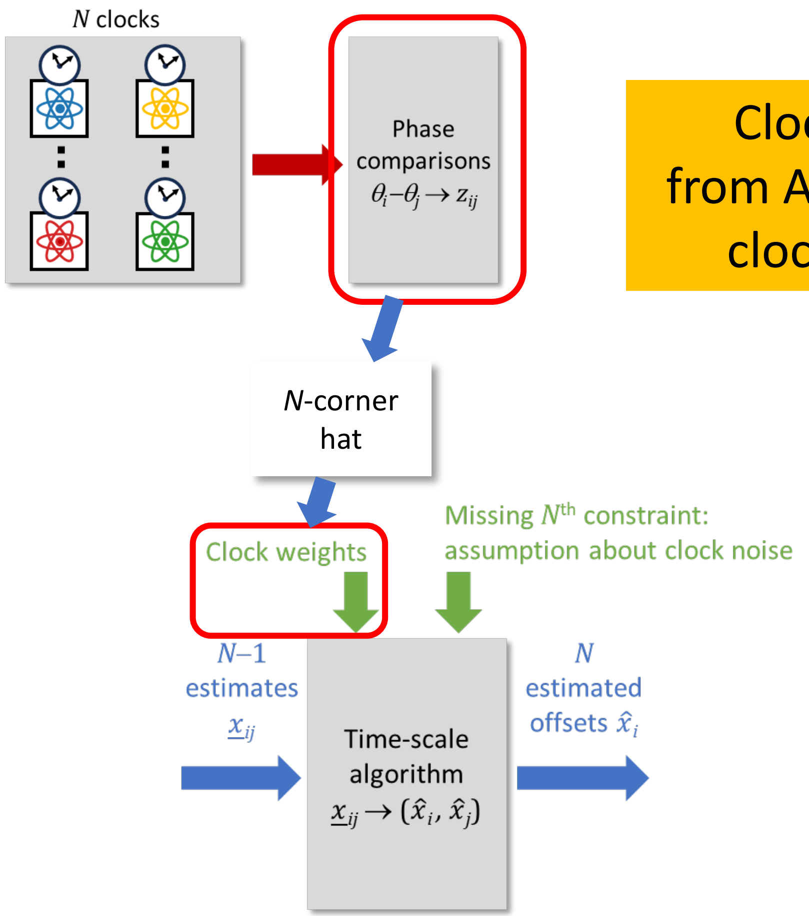


Figure 4: Two-clock time scale with "optimum" weights

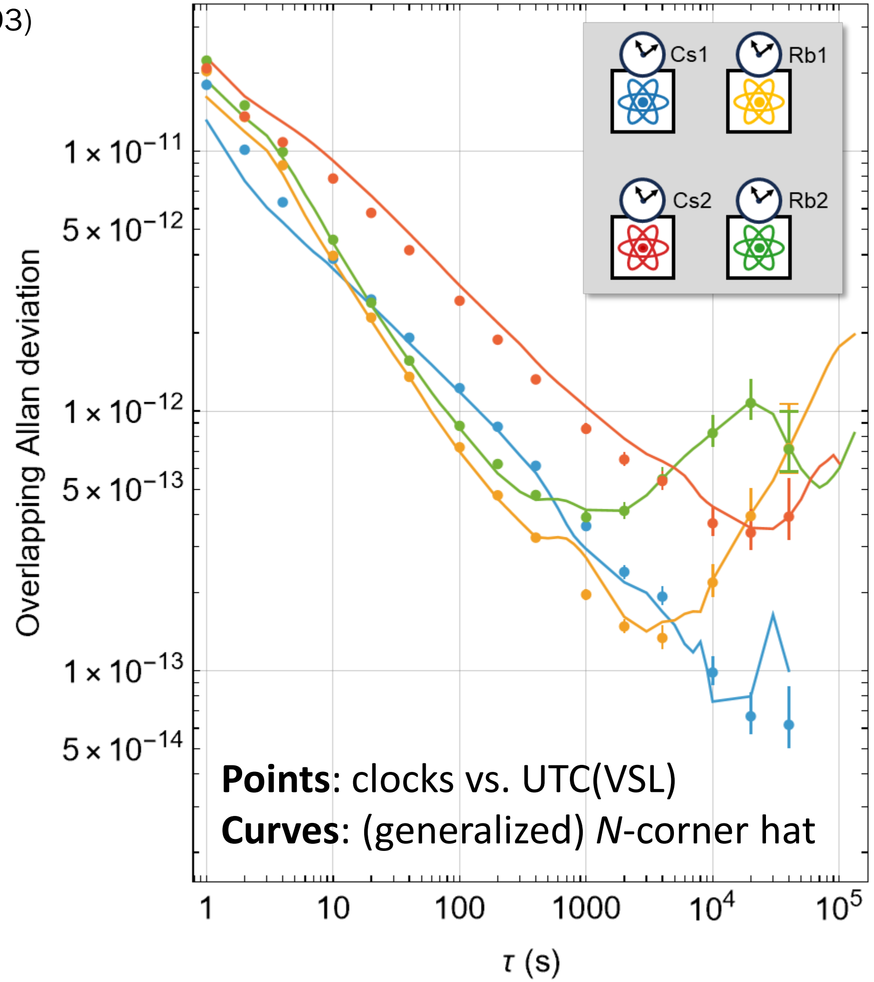
Clock weights

- Clock weights determined from Allan deviation slopes (of individual clocks)
- Adevs obtained via generalized N-corner hat (see e.g. work of Premoli & Tavella et al., BIPM)

P. Tavella & A. Premoli, *IEEE Trans. Instr. Meas.*, **IM-42**, pp. 7-13 (1993)

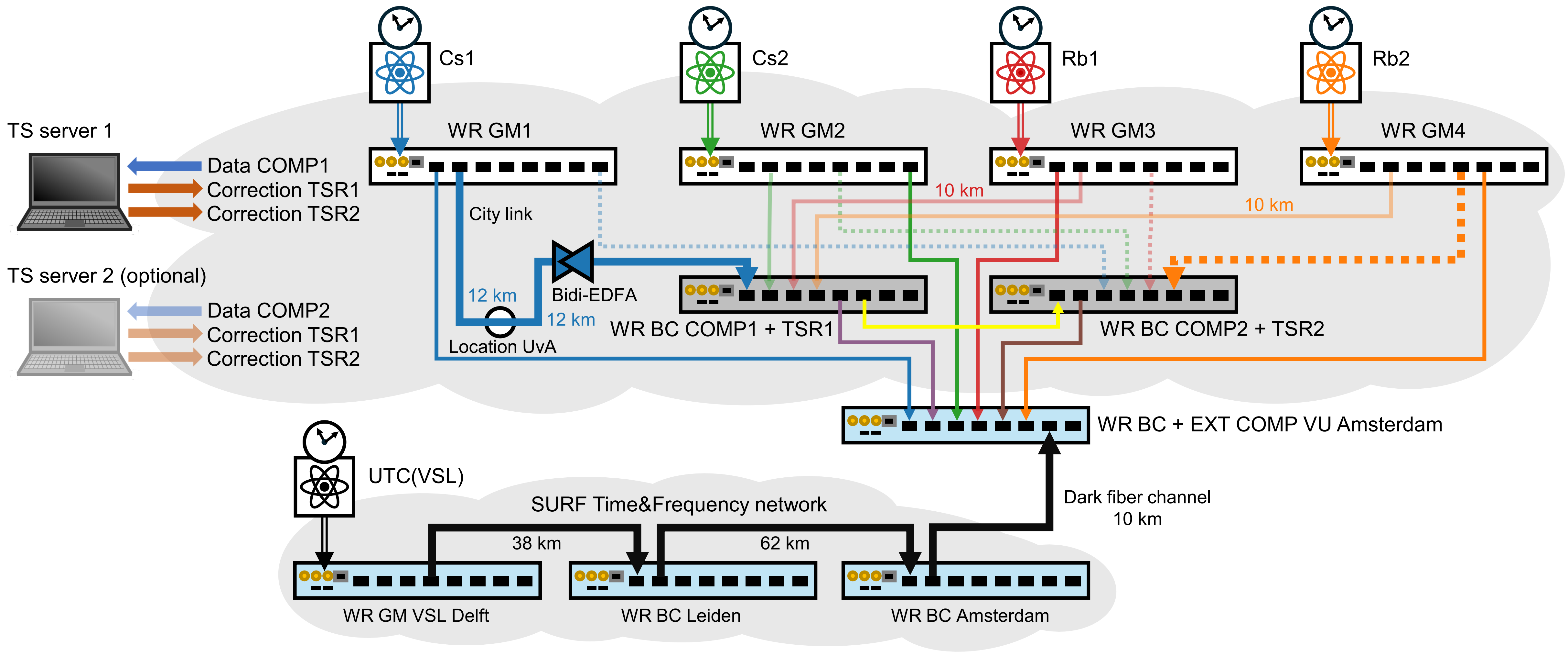


Clock weights derived from Adev slopes using only clock comparison data

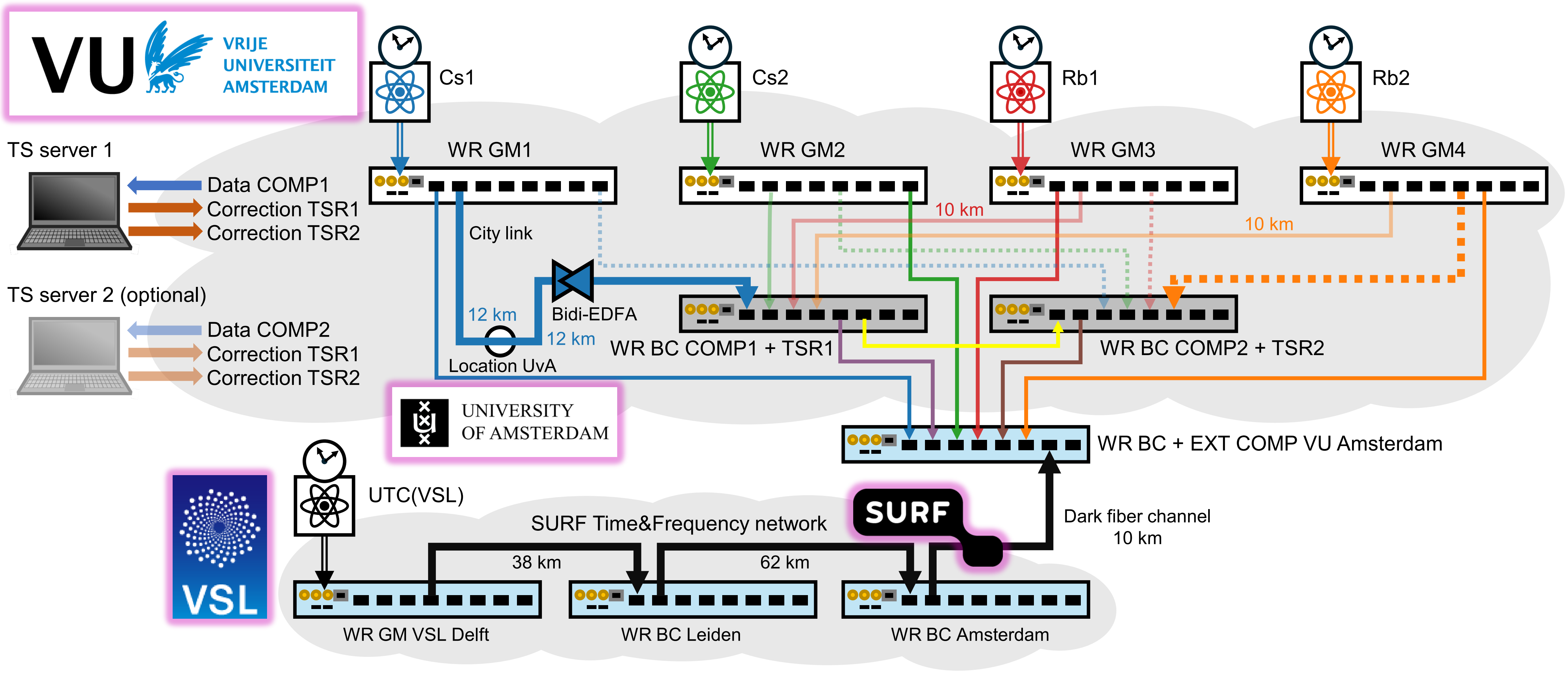


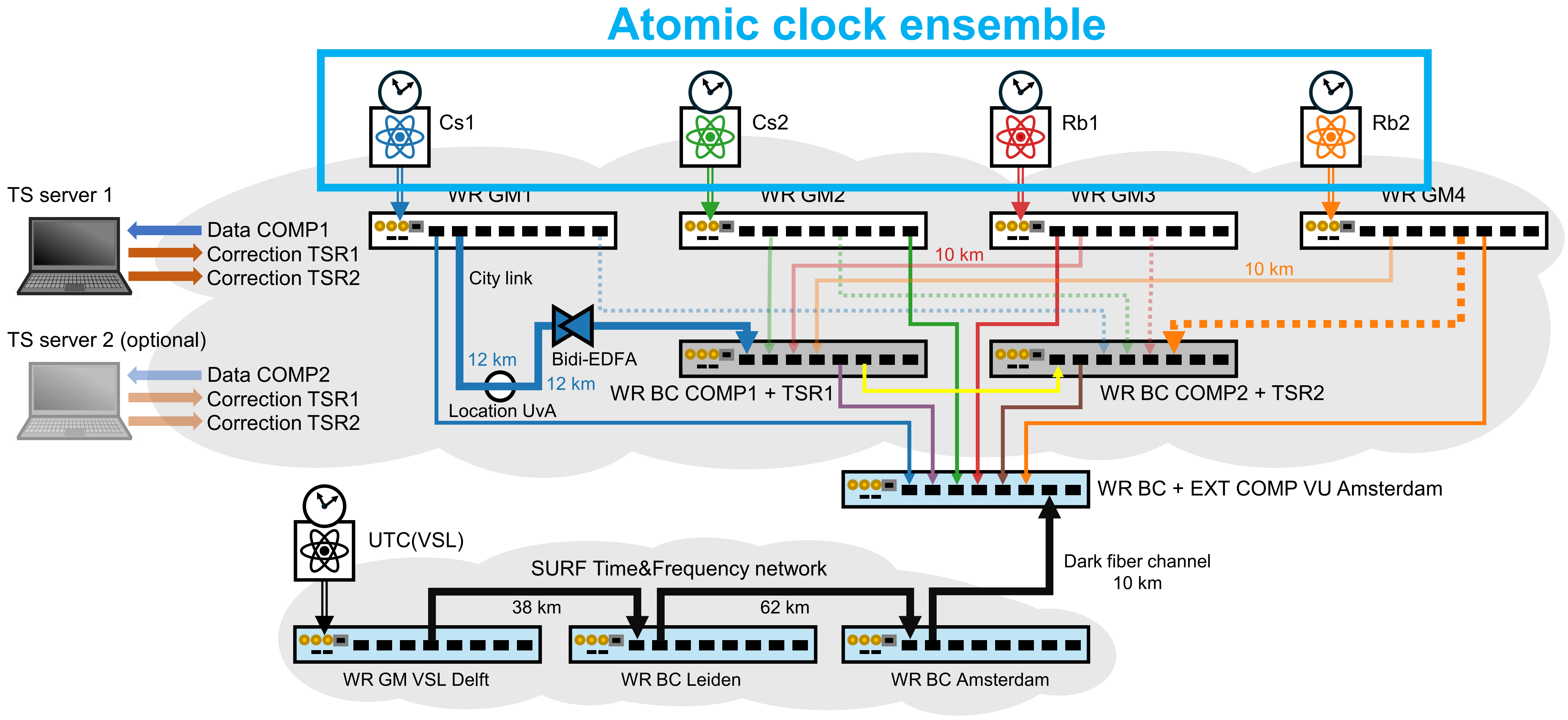
Time scale implementation in WR

Amsterdam Metropolitan testbed and link to VSL

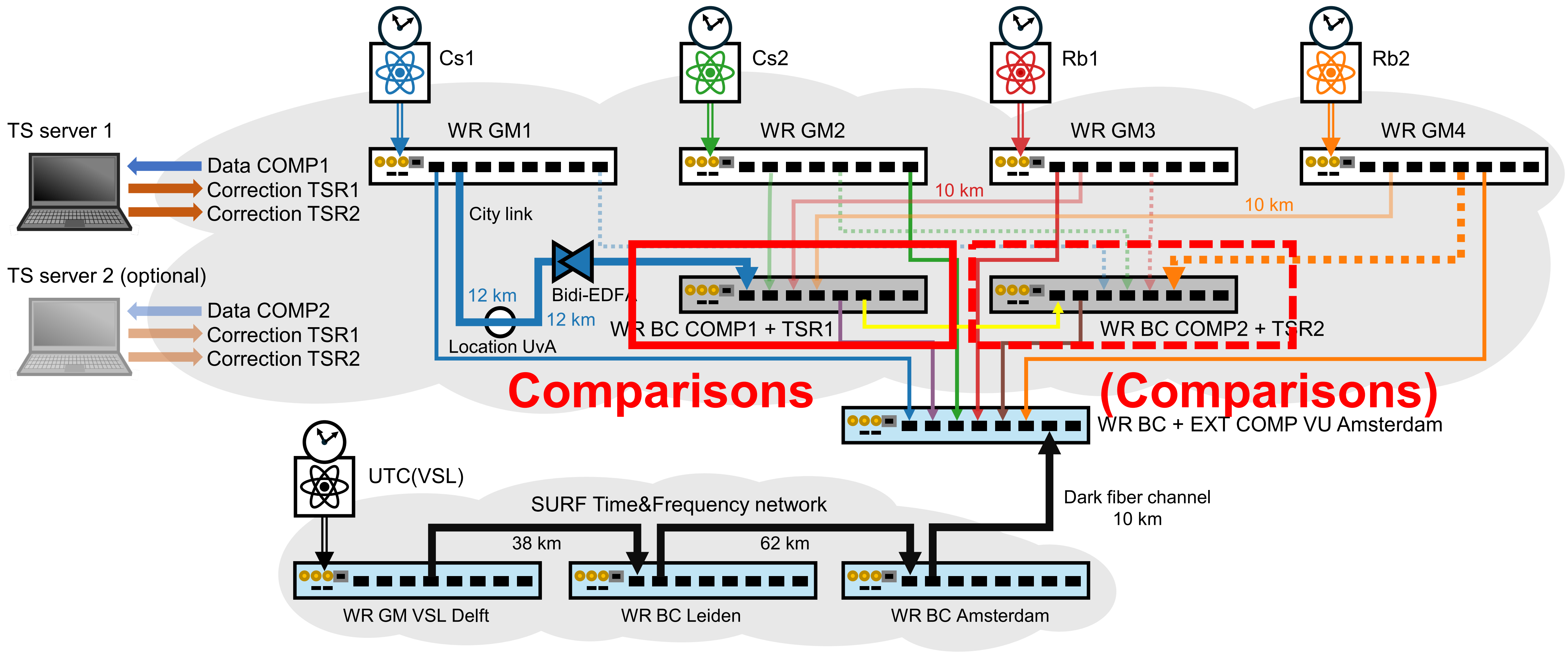


Amsterdam Metropolitan testbed and link to VSL

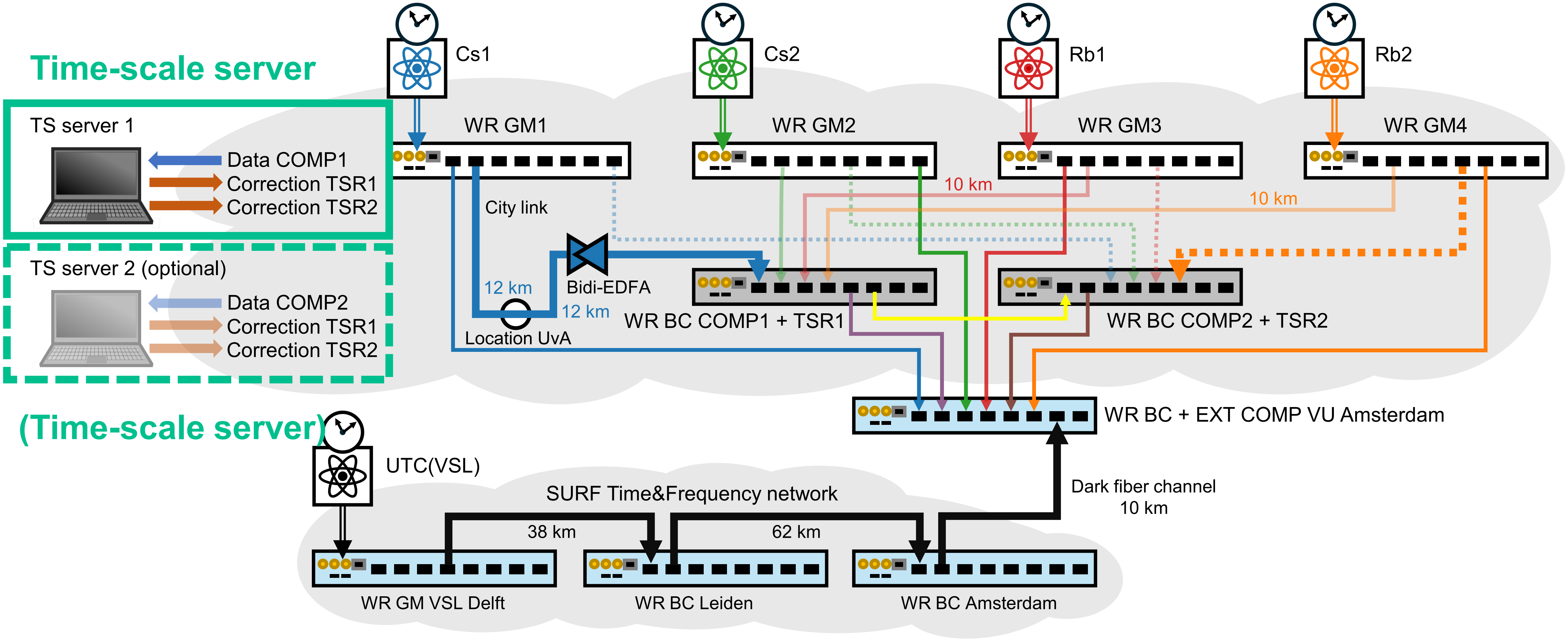




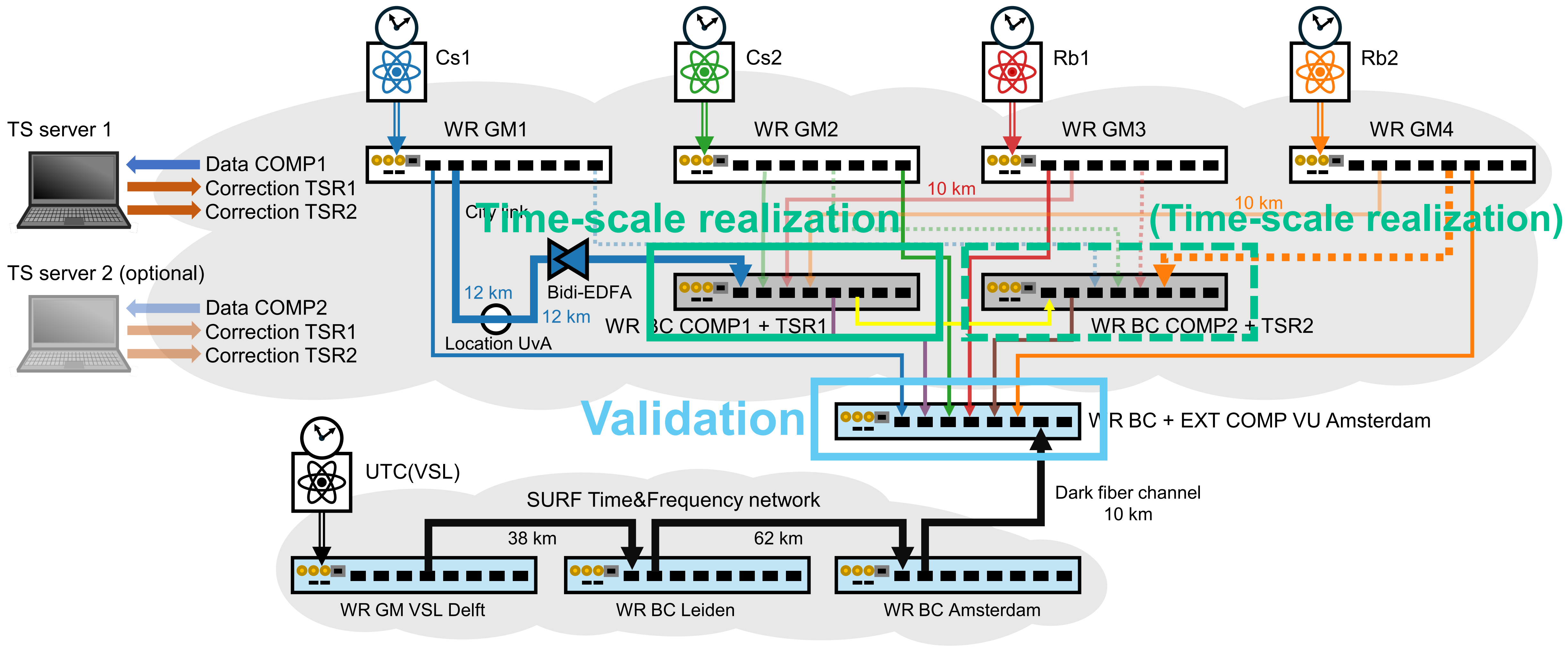
Atomic clock ensemble
Cs1
Cs2
Rb1
Rb2
WR GM1
WR GM2
WR GM3
WR GM4
TS server 1
Data COMP1
Correction TSR1
Correction TSR2
TS server 2 (optional)
Data COMP2
Correction TSR1
Correction TSR2
City link
10 km
10 km
12 km
Bidi-EDFA
12 km
Location UvA
WR BC COMP1 + TSR1
WR BC COMP2 + TSR2
WR BC + EXT COMP VU Amsterdam
UTC(VSL)
SURF Time&Frequency network
Dark fiber channel
10 km
38 km
62 km
WR GM VSL Delft
WR BC Leiden
WR BC Amsterdam



TS server 1
Data COMP1
Correction TSR1
Correction TSR2
TS server 2 (optional)
Data COMP2
Correction TSR1
Correction TSR2
Cs1
Cs2
Rb1
Rb2
WR GM1
WR GM2
WR GM3
WR GM4
City link
10 km
10 km
12 km
Bidi-EDFA
12 km
Location UvA
WR BC COMP1 + TSR1
WR BC COMP2 + TSR2
Comparisons
(Comparisons)
WR BC + EXT COMP VU Amsterdam
UTC(VSL)
SURF Time&Frequency network
Dark fiber channel
10 km
38 km
62 km
WR GM VSL Delft
WR BC Leiden
WR BC Amsterdam



Time-scale server
TS server 1
Data COMP1
Correction TSR1
Correction TSR2
TS server 2 (optional)
Data COMP2
Correction TSR1
Correction TSR2
(Time-scale server)
Cs1
Cs2
Rb1
Rb2
WR GM1
WR GM2
WR GM3
WR GM4
10 km
10 km
City link
12 km
Bidi-EDFA
12 km
Location UvA
WR BC COMP1 + TSR1
WR BC COMP2 + TSR2
WR BC + EXT COMP VU Amsterdam
UTC(VSL)
SURF Time&Frequency network
Dark fiber channel
10 km
38 km
62 km
WR GM VSL Delft
WR BC Leiden
WR BC Amsterdam



Cs1
Cs2
Rb1
Rb2
TS server 1
Data COMP1
Correction TSR1
Correction TSR2
WR GM1
WR GM2
WR GM3
WR GM4
10 km
10 km
Time-scale realization
(Time-scale realization)
TS server 2 (optional)
Data COMP2
Correction TSR1
Correction TSR2
12 km
Bidi-EDFA
12 km
Location UvA
WR BC COMP1 + TSR1
WR BC COMP2 + TSR2
Validation
WR BC + EXT COMP VU Amsterdam
UTC(VSL)
SURF Time&Frequency network
Dark fiber channel
10 km
38 km
62 km
WR GM VSL Delft
WR BC Leiden
WR BC Amsterdam

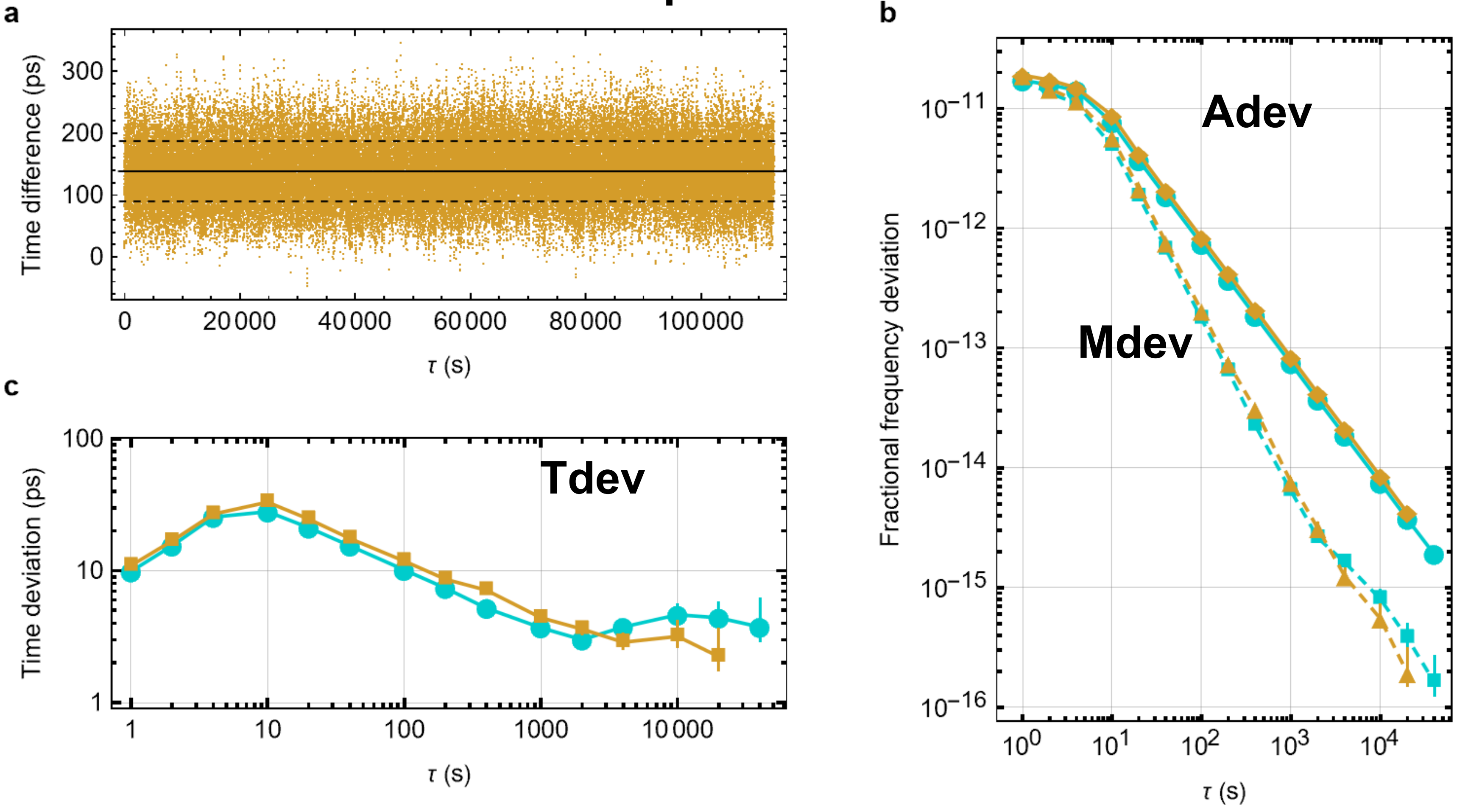
Do the time scales produce the same time?

Time errors need be suppressed to picosecond-range for seamless fail-over

Offset between two independent time-scale realizations

R. González Escudero et al., submitted

Residual time errors at 100 ps level

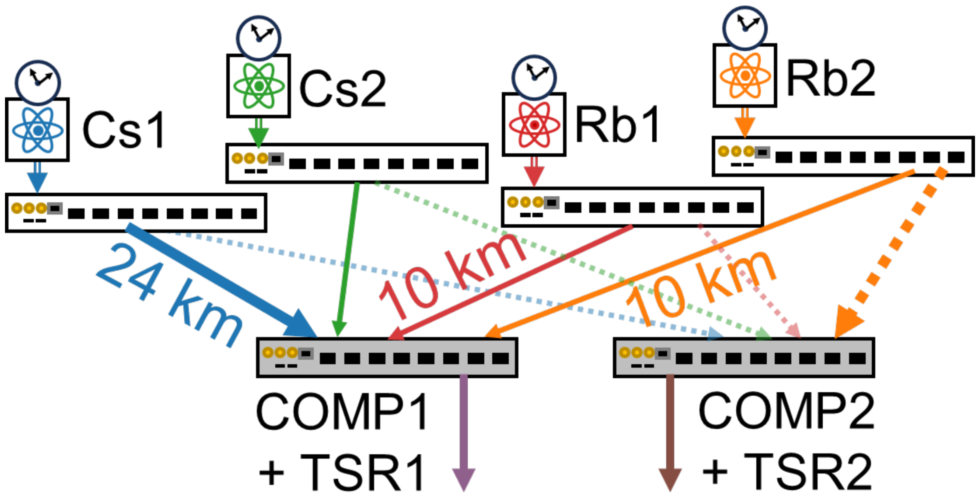


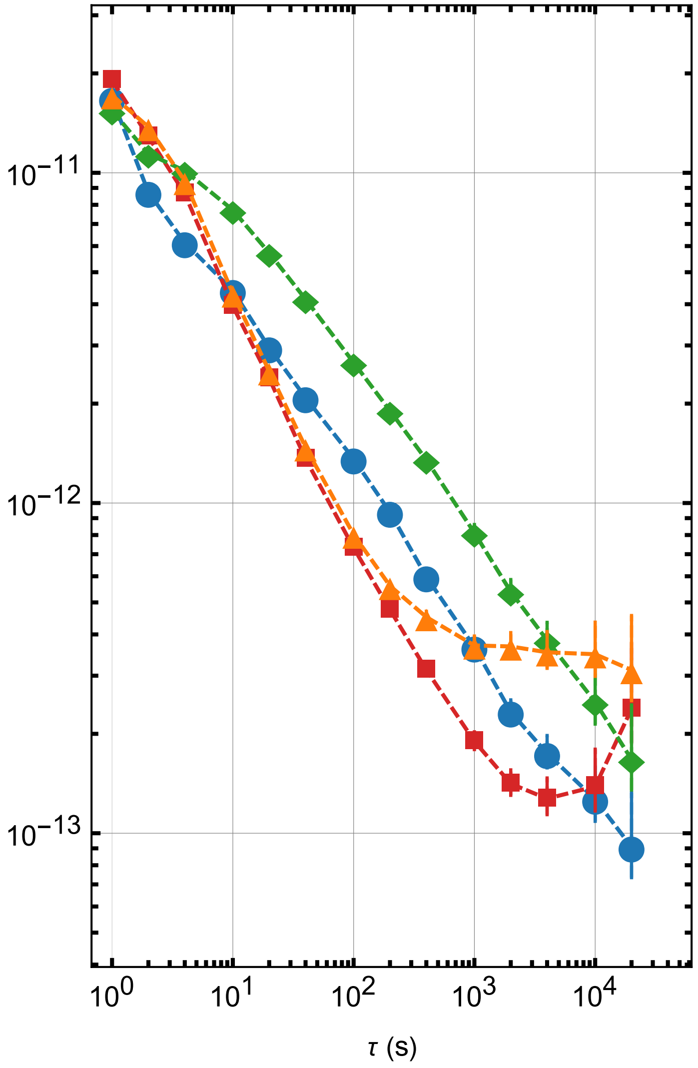
Does the time scale offer improved stability?

Never let the good stability of a clock ensemble go to waste

Time scale frequency stability

Clock stabilities (adev)

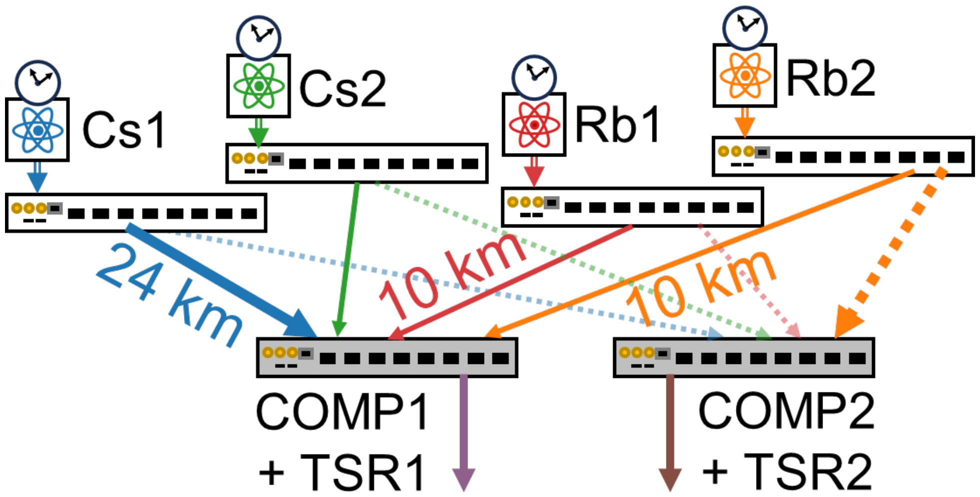


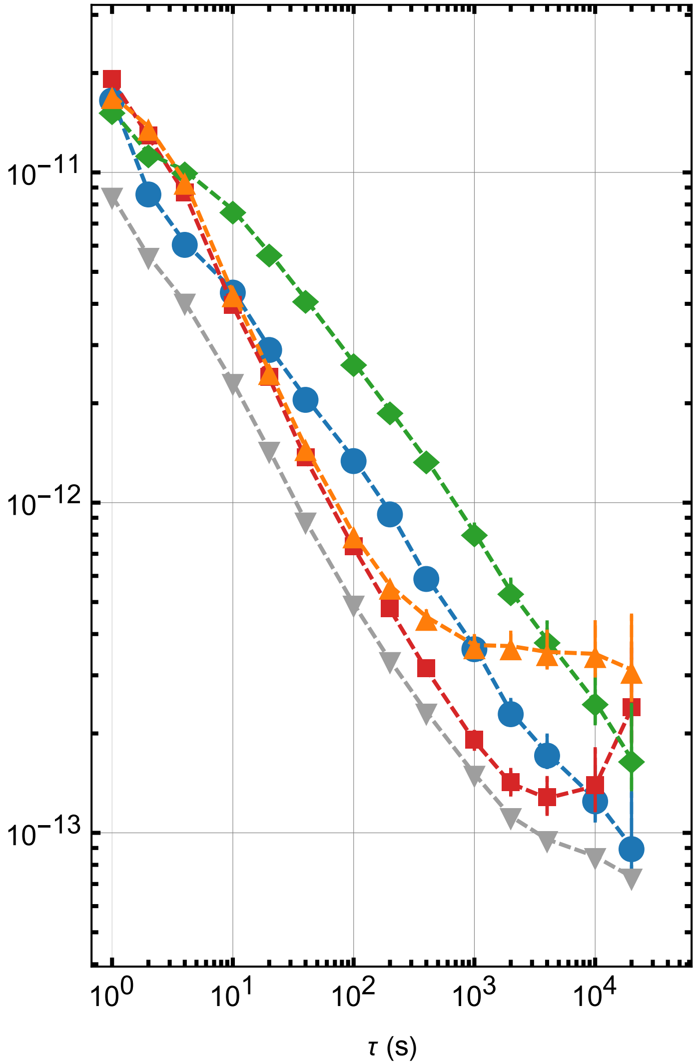


Time scale frequency stability

Clock stabilities (adev)

Theoretical optimum stability



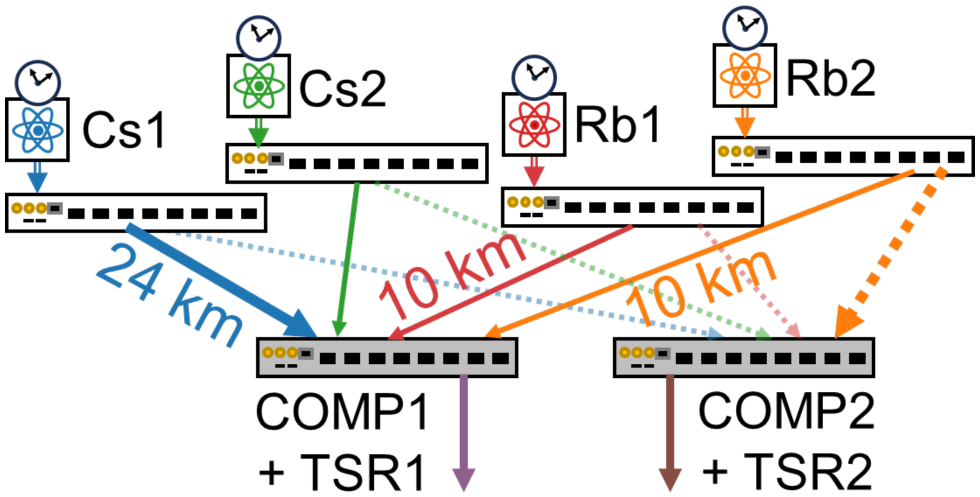


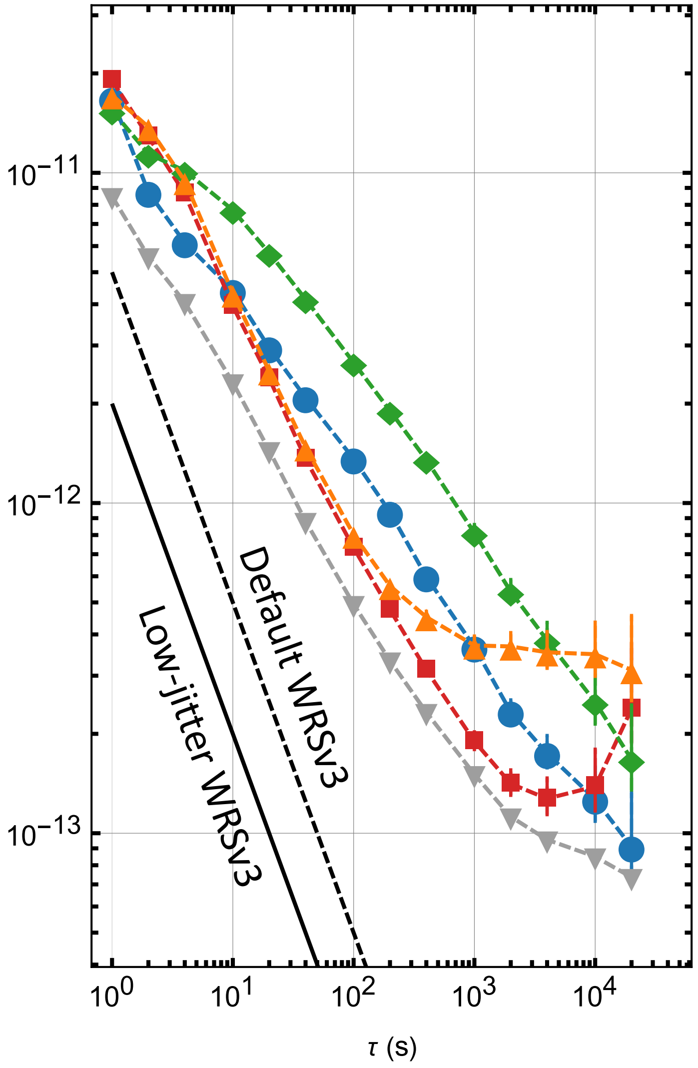
Time scale frequency stability

Clock stabilities (adev)

Theoretical optimum stability

WR noise floor (regular & low jitter)





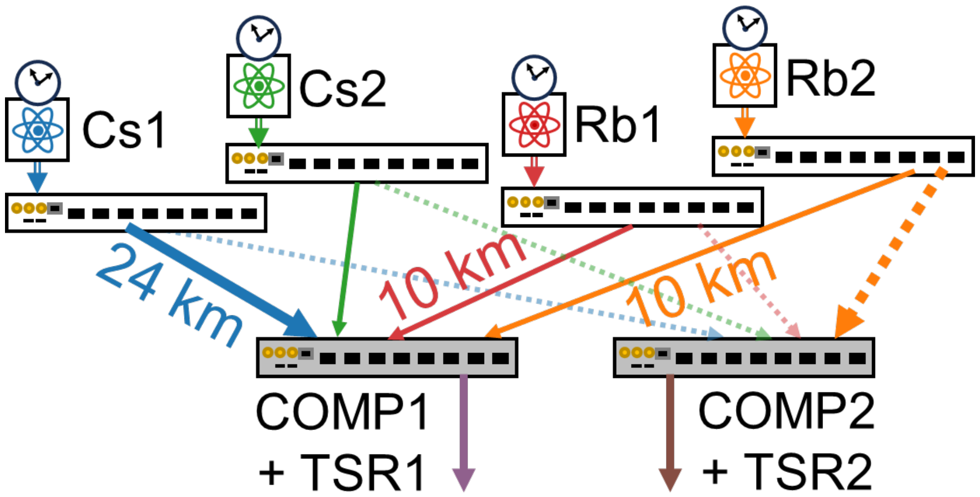
Time scale frequency stability

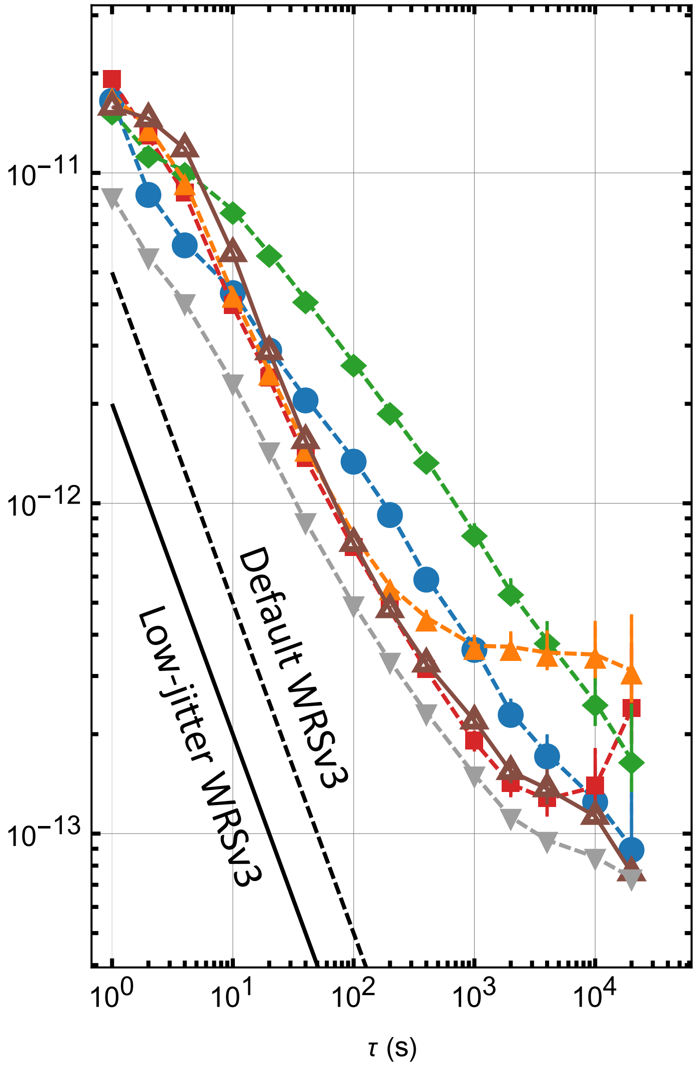
Clock stabilities (adev)

Theoretical optimum stability

WR noise floor (regular & low jitter)

Time scale 2 (derived from Rb2)





Time scale frequency stability

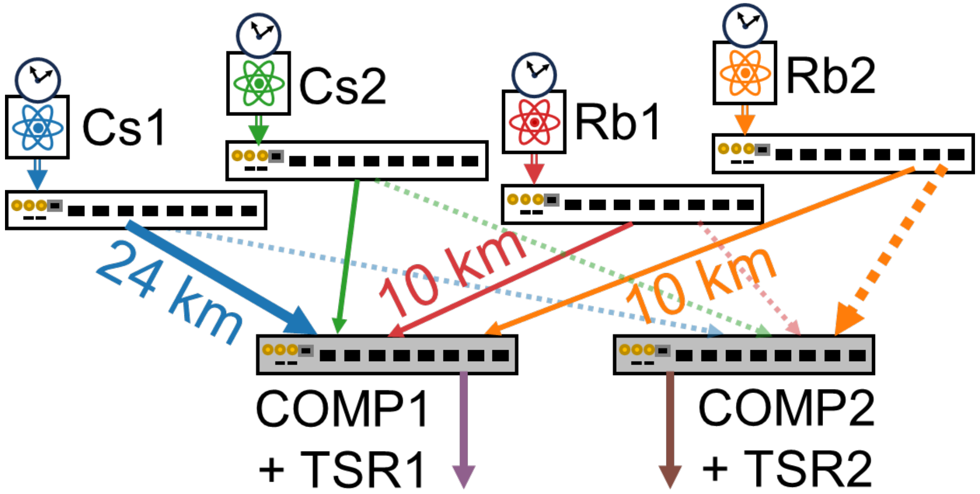
Clock stabilities (adev)

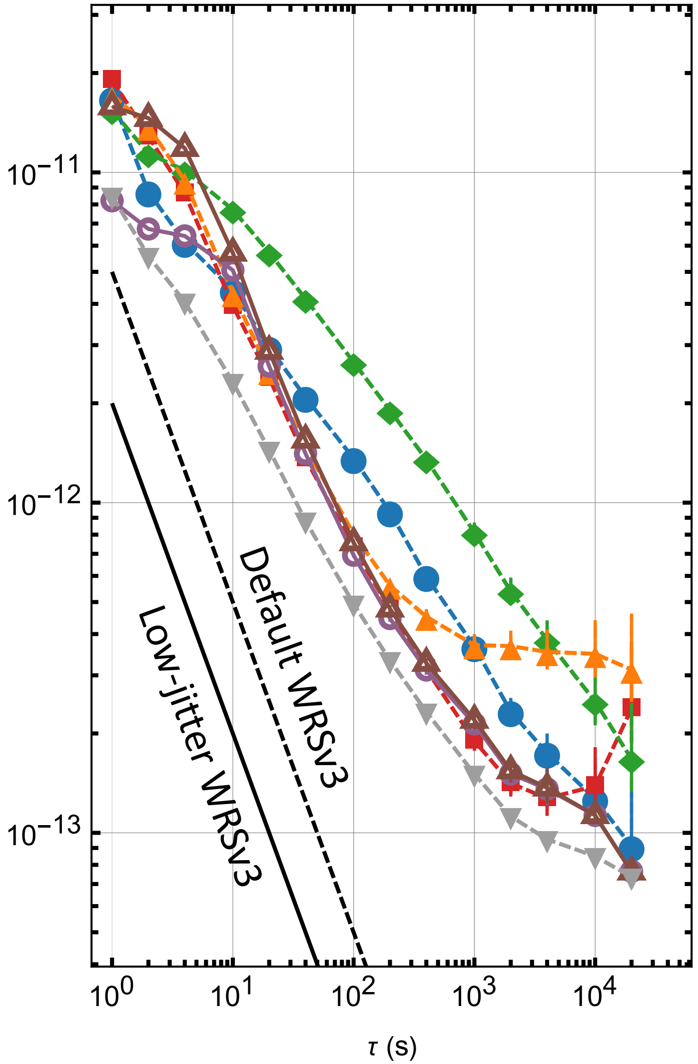
Theoretical optimum stability

WR noise floor (regular & low jitter)

Time scale 2 (derived from Rb2)

Time scale 1 (derived from Cs1)





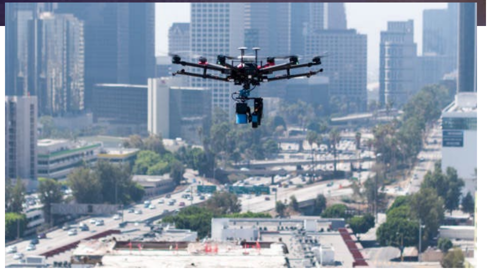
Next steps

- Truly distributed clock ensemble
- Longer measurement runs
- Demonstrate fail-over
- Global steering of the time scale

Outlook

- Towards scalable, robust WR networks in the form of a continuous, picosecond-range distributed time scale
- Distributed and fully redundant (no single point of failure)
- GNSS backup, cnPRTC, terrestrial PNT, quantum communication, radio astronomy, …

Terrestrial PNT
Koelemeij et al., *Nature* **611,** 473–478 (2022)



Radio astronomy
Boven et al., Exp. Astronomy 61,3 (2026)

cnPRTC
Cosart et al., *IEEE Comms. Mag.* April 2023, 0163-6804

Quantum Internet
Stolk et al., *Sci. Adv.* **10,** eadp6442 (2024)

Thank you!

Questions:
marcweissconsulting@gmail.com

VU Amsterdam

J.C.J. Koelemeij

Rodrigo Gonzalez Escudero

Sougandh Kannoth Mavila

Marc A. Weiss (ext. consultant)

Frank Cozijn

Kees Steinebach

Rob Kortekaas

Joanne Slijper

Kjeld Eikema

VSL

Erik Dierikx

Yan Xie

University of Amsterdam

Klaasjan van Druten

Sumit Sarkar

Florian Schreck

SURF

Sander Klemann

Paul Klop

CERN

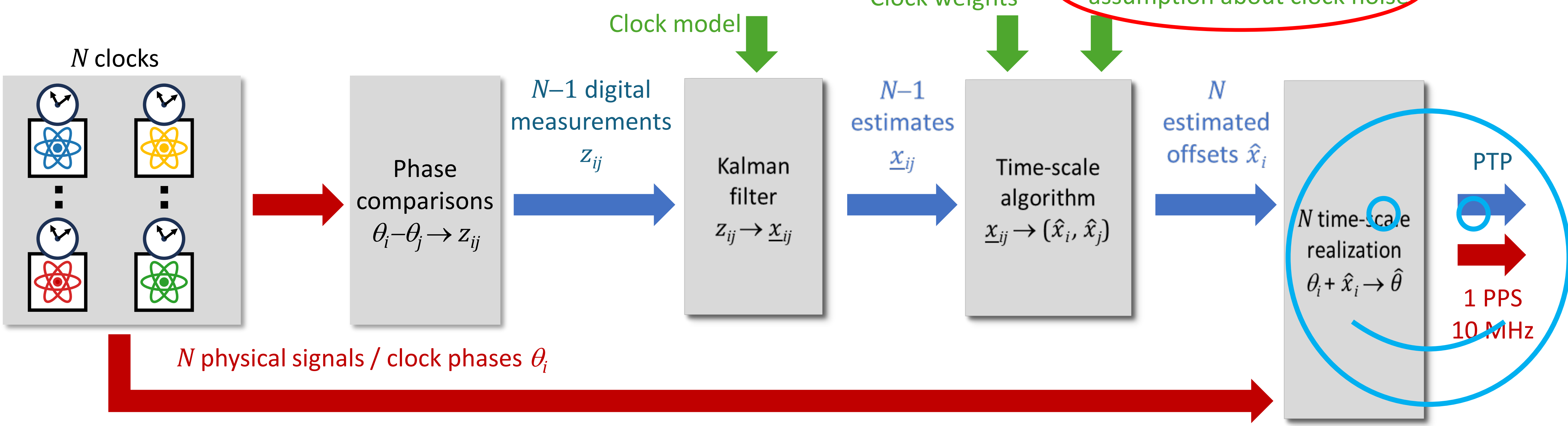
White Rabbit Collaboration

Extra slides

Modified Stein's algorithm

S.R. Stein, 'Time Scales Demystified', Proc. 2003 IFCS/17th EFTF, p. 223 (2003).

- Problem: time scales $\hat{\theta}$ from different clocks i drift apart by > 1 ns/hr due to assumption ($\Rightarrow \hat{x}_i$ contain syst. errors)
- Make error common-mode using KF estimates $\hat{\theta} = \theta_i + \underline{x}_{ji} + \hat{x}_j \approx \theta_j + \hat{x}_j$ for all i
- KF statistical error with mean 0 $\Rightarrow$ very good approximation

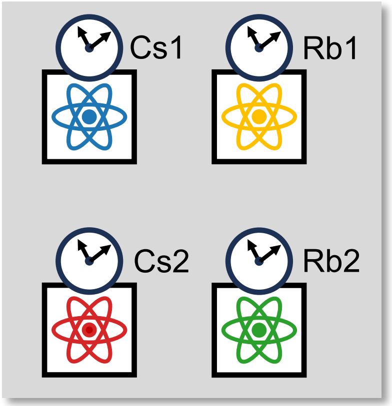


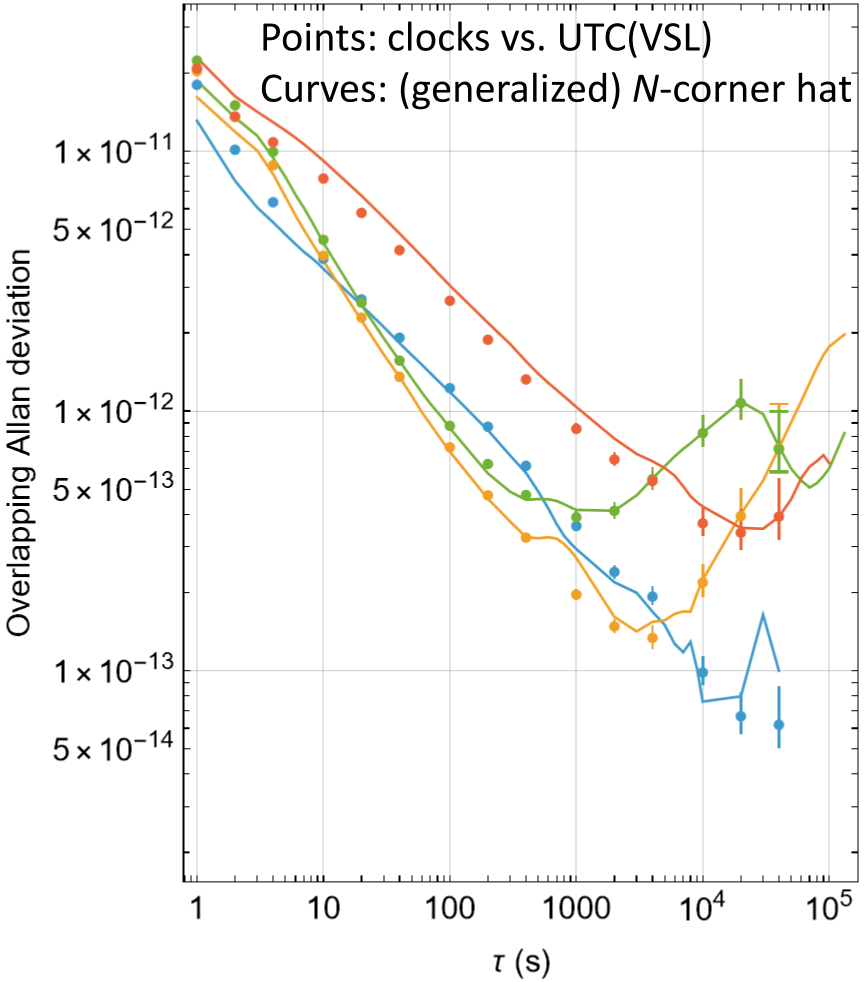
Estimating Clock Allan covariances

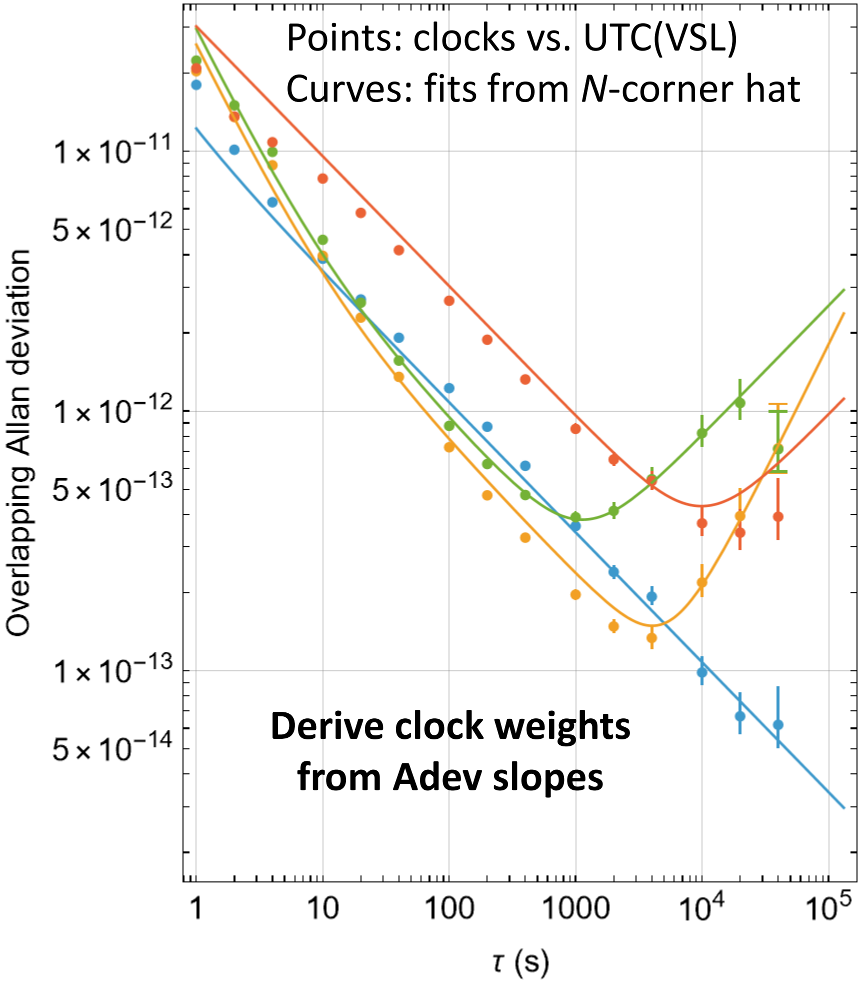
- Essentially generalized *N*-corner hat
- Example: four clocks *N*=1..4
- Measurements: one pivot clock, three series of phase differences (1-2,1-3,1-4)
- Compute Allan covariances ‘Delta’ of clock differences (6 known variables)
  - formula
- What we need to know: Allan covariances for individual clocks (10 unknowns)
  - Matrix relation connecting Sigma with Delta covariances
- For each tau, minimize trace of Sigma matrix subject to matrix relation and requiring Sigma matrix is positive semidefinite (eigenvalues>0, symmetric)
  - Convex optimization (Mathematica or similar)

Results Allan covariance estimation

- Essentially generalized *N*-corner hat







Work in progress…