# Initial Simulation Results for Time Error Accumulation in an IEC/IEEE 60802 Network

**Geoffrey M. Garner**
**Huawei (Consultant)**

**gmgarner@alum.mit.edu**

*IEEE 802.1 TSN TG*
2020.03.16

# Outline

- Introduction
- Simulation Model
- Assumptions for Simulation Cases
- Results
- Conclusion

# Introduction

- At the November 2019 IEC/IEEE 60802 meeting, the author of this presentation indicated during the comment resolution discussion that it would be possible to perform simulations of end-to-end dynamic time error (dTE) performance for an Industrial (i.e., 60802) network
- The presentation [1] at the January 2020 IEC/IEEE 60802 meeting described the assumptions that decisions would be needed on for these simulations; these include
  - Hypothetical Reference Model (HRM) (e.g., number of PTP Instances in the reference chain)
  - Protocol parameters (e.g., message rates, residence time, etc.)
  - Physical parameters (e.g., timestamp granularity)
  - Local clock model (e.g., maximum frequency offset and drift rate)
  - Endpoint filter parameters (e.g., 3 dB bandwidth, gain peaking)
- As a result of the discussion of [1], initial decisions were made on the assumptions
- The current presentation summarizes the simulation model, and presents the simulation results for dynamic time error (dTE) accumulation in an IEC/IEEE 60802 network

# Simulation Model - 1

- Previous simulations for 802.1AS performance were presented in 802.1 presentations during the 2006 – 2011 period
- The simulation model was summarized in [2] and [3]; since this material was presented many years ago, the relevant slides are repeated here for the benefit of participants who did not attend IEEE 802.1 during that period
  - Some additions/modifications to those slides have been made to reflect the new local clock models needed for IEC/IEEE 60802 networks, i.e., models based on sinusoidal or triangular-wave frequency variation (see [4])
- Model is discrete-event; the events are the sending and receiving of Sync, Pdelay_Req, and Pdelay_Resp messages
  - For simplicity, only one-step behavior is modeled (one-step versus two-step has minimal impact on time error performance)

# Simulation Model (adapted from [2] and [3]) – 1

- Each node contains a free-running clock, for which the following is specified
  - Frequency tolerance $y$
    - At initialization, the actual frequency offset can be chosen randomly from a uniform distribution over $[-y, y]$, or a specific value can be chosen
  - Frequency drift rate $D$
  - Phase measurement granularity
  - Whether a power-law noise model is used and, if so, parameters of power-law noise models (details of these models given in later slides):
    - White Phase Modulation (WPM)
    - Flicker Phase Modulation (FPM)
    - White Frequency Modulation (WFM)
    - Flicker Frequency Modulation (FFM)
    - Random Walk Frequency (RWFM)
  - Whether  a sinusoidally-varying phase/frequency model is used and, if so, amplitude, frequency, and initial phase (which can be random if desired) of the variation
  - Whether a triangular wave frequency variation model is used and, if so, amplitude, frequency, and initial phase (which can be random if desired) of the variation

# Simulation Model (adapted from [2] and [3]) - 2

- Each link is associated with a delay model
  - For now, the link delay is fixed, but can be asymmetric (but it is taken as symmetric in the initial simulation cases presented here because the focus here is on dTE)
- For now, only syncLocked mode is modeled; this means that each transmitting port in the HRM has the same mean sync interval
  - At present, the IEC/IEEE 60802 profile document draft specifies a single mean Sync rate, which means this assumption is satisfied
  - Note that when the simulation model was originally developed (for [2] and [3]), the focus was 802.1AS-2011, for which the normal behavior was what is now called SyncLocked mode

# Simulation Model (adapted from [2] and [3]) - 3

- Times associated with messages; fixed for now and supplied as input
  - Sync interval
  - Pdelay interval
  - Pdelay turnaround time (time between receipt of Pdelay_Req and sending of Pdelay_Resp)
  - Residence time (time between receipt of Sync by a PTP Instance that is not the Grandmaster (GM) and sending of Sync to the next PTP Instance in the HRM)
  - Both IEEE 802.1AS-2020 and IEEE 1588-2019 specify how the Sync and Pdelay intervals are allowed to vary
    - Since the simulator was developed, models for this variation were subsequently developed for use in simulating ITU-T Telecom profile (ITU-T Rec. G.8275.1) performance (in a separate simulator)
    - These models could be added to the current simulator if desired; however, their impact would likely be minimal because of the use of syncLocked mode here (G.8275.1 does not specify syncLocked mode)
      - If syncLocked mode is not used, the main impact is that the time between receipt and sending of Sync could as large as an actual Sync interval

# Simulation Model (adapted from [2] and [3]) - 4

- The basic operation of the simulator is

generateInitialEvents(); /* Sending initial Sync by GM; sending initial Pdelay_Req from each time-aware system to the next upstream time-aware system */

while (timer <= endTime) {

removeNextEvent();

computeFreeRunningClockTimesAtTimeOfNextEvent(); /* local clock wander generation model (slide 5) is invoked here */

computeUnfilteredSynchronizedTimeEstimateAtTimeOfNextEvent();

/* based on current estimate of rateRatio relative to GM and most recent (freeRunningTime, synchronizedTime) association */

computeFilteredSynchronizedTimeEstimateAtTimeOfNextEvent();

eventHandler(); /* neighborRateRatio and cumulativeRateRatio computations are performed in handling relevant events */

}

- The events are maintained in a linked list, in chronological order relative to global timer

# Simulation Model (adapted from [2] and [3]) - 5

- The endpoint filter is modeled as a linear, second-order phase-locked loop with 20 dB/decade roll-off, as described in [1] (see slides 10 and 11 of [1]).
- Such a filter can be modeled and simulated in many different ways; here, the approach of ITU-T Rec. G.8251, Appendix VIII is used. This approach can be summarized as follows:
  - The second order filter equations are written in state variable form (i.e., a system of two first-order linear differential equations
  - The state vector at current time step is written as convolution integral of the input vector and the impulse response matrix
  - The impulse response matrix is calculated exactly (as a matrix exponential), and the integral is evaluated using a trapezoidal approximation for the input
  - The output is written in terms of states
  - Note that this approach is numerically stable for all values of time step because the impulse response matrix is computed exactly; however, aliasing of input frequencies above the Nyquist frequency can still occur

# Simulation Model (adapted from [2] and [3]) - 6

- In setting the integration time step for the filter, the time between the current and next event is divided into the smallest number of time steps such that the size of the time step is not larger than a specified maximum, i.e.,
  - If $T$ = time between events
  - $\Delta t_{max}$ = maximum time step (input parameter)
  - $\Delta t$ = actual time step
  - Then
  - $N_{steps} = \text{ceil}\,(T/\Delta t_{max})$
  - $\Delta t = T/N_{steps}$

# Simulation Model (adapted from [2] and [3]) - 7

- The following is a high-level overview of the processing of each event type
- Sending Pdelay_Req event
  - Generate time stamp relative to free-running clock (compute free-running time corresponding to current value of timer)
  - Schedule next sending of Pdelay_Req event and add to linked
- Receipt of Pdelay_Req event
  - Generate time stamp relative to free-running clock (compute free-running time corresponding to current value of timer)
  - Schedule sending of Pdelay_Resp event
- Sending of Pdelay_Resp event
  - Generate time stamp relative to free-running clock (compute free-running time corresponding to current value of timer)
  - Place Pdelay turnaround time in message structure
  - Schedule receipt of Pdelay_Resp event

# Simulation Model (adapted from [2] and [3]) - 8

- Receipt of Pdelay_Resp event
  - Generate time stamp relative to free-running clock (compute free-running time corresponding to current value of timer)
  - Compute neighborRateRatio
    - A granularity for the neighborRateRatio computation can be specified (e.g., based on a given number of bits of precision for the computation)
  - Compute neighborPropDelay
  - Note that there is no new event to generate in this case
- Sending of Sync event
  - Generate time stamp relative to free-running time corresponding to current value of timer)
  - Compute residence time, corrected for cumulativeRateRatio, based on time stamp and saved time stamp (relative to free-running timer) of most recently received Sync
  - Add residence time and current neighborPropDelay to correctionField
  - Schedule receipt of Sync at downstream node

# Simulation Model (adapted from [2] and [3]) – 9

- Receipt of Sync event
  - Generate time stamp relative to free-running time corresponding to current value of timer)
  - Compute correctedMasterTime (GM time estimate), which is the sum of the preciseOriginTimestamp, correctionField, and neighborPropDelay)
  - Compute cumulativeRateRatio relative to GM using received cumulativeRateRatio and current neighborRateRatio
  - Compute unfiltered phase offset, which is the difference between the correctedMasterTime and current local clock time (the time stamp for receipt of the Sync)
    - Note that the (time stamp, correctedMasterTime) becomes the current association of free-running and GM time

# Local Clock Noise Generation Model

- The sinusoidal and triangular wave noise generation models are described in [4], and will therefore not be repeated here
- The flicker frequency modulation (FFM) noise generation requirement of IEEE Std 802.1AS-2020, Annex B is described in [4]
  - A simulation model for generating FFM noise at the level of the requirement (mask) of 802.1AS, Annex B is not described in [4]; however, such a model is described in [3]
  - In this model, the FFM power spectral density (proportional to $1/f^3$) is approximated by a series of steps, which can be realized by passing white noise through a set of successive lead/lag filters
  - This technique is based on work of Barnes, Jarvis, and Greenhall (see Reference 3. of the Appendix); the details are given in the Appendix (the slides there are adapted from [3])

# Assumptions for HRM - 1

- These assumptions are based on the discussion of [1] at the January 2020 IEC/IEEE 60802 meeting, and the latest IEC/IEEE 60802 draft (D1.1)
- These assumptions on the HRM are common to all simulation cases
- The HRM is a linear chain that consists of 100 PTP Instances, and therefore with 99 PTP links connecting each successive pair of PTP Instance
  - The first PTP Instance in the chain is the Grandmaster PTP Instance
  - The next 98 PTP Instances are PTP Relay Instances
  - The last PTP Instance is a PTP End Instance
  - The PTP End Instance contains an endpoint filter, through which the transported time is computed

# Assumptions for HRM - 2

- The GM and each PTP Relay Instance do not filter the timestamps with an endpoint filter when computing the value of the originTimestamp and correctionField of each transmitted Sync message
  - Rather, these fields are computed using the same fields of the most recently received Sync message, the <syncEventIngressTimestamp> of the most recently received Sync message, the <syncEventEgressTimestamp of the Sync message being transmitted, and the current value of rateRatio (i.e., cumulative rateRatio)
- However, the information at each PTP Relay Instance is used to separately compute a filtered (recovered) time, which could be used, e.g., by a co-located end application
- The GM is assumed to have zero time error
  - This is equivalent to the transported time being computed relative to the GM

# Other Assumptions Common to All Cases - 1

- These assumptions for the simulations are based on the results of the discussion of [1] at the January 2020 IEC/IEEE 60802 meeting
- Timestamp granularity: 8 ns
- Time between successive Sync messages: 0.03125 s
  - Mean Sync message rate = 32 message/s
- Time between successive Pdelay_Req messages: 1 s
  - Mean Pdelay_Req message rate = 1 message/s
- Use syncLocked mode (since all ports have same mean Sync interval)
- Residence time: 10 ms
- Pdelay turnaround time (i.e., time between receipt of Pdelay_Req and sending of Pdelay_Resp): 10 ms
- Endpoint filter 3 dB bandwidth and gain peaking: 3.78 Hz, 1.049 dB
  - Equivalent to proportional gain of 20 and integral gain of 80, both normalized to VCO gain of 1 (see [1])
- neighborRateRatio computation granularity: $2.328 \times 10^{-10}$

# Other Assumptions Common to All Cases - 2

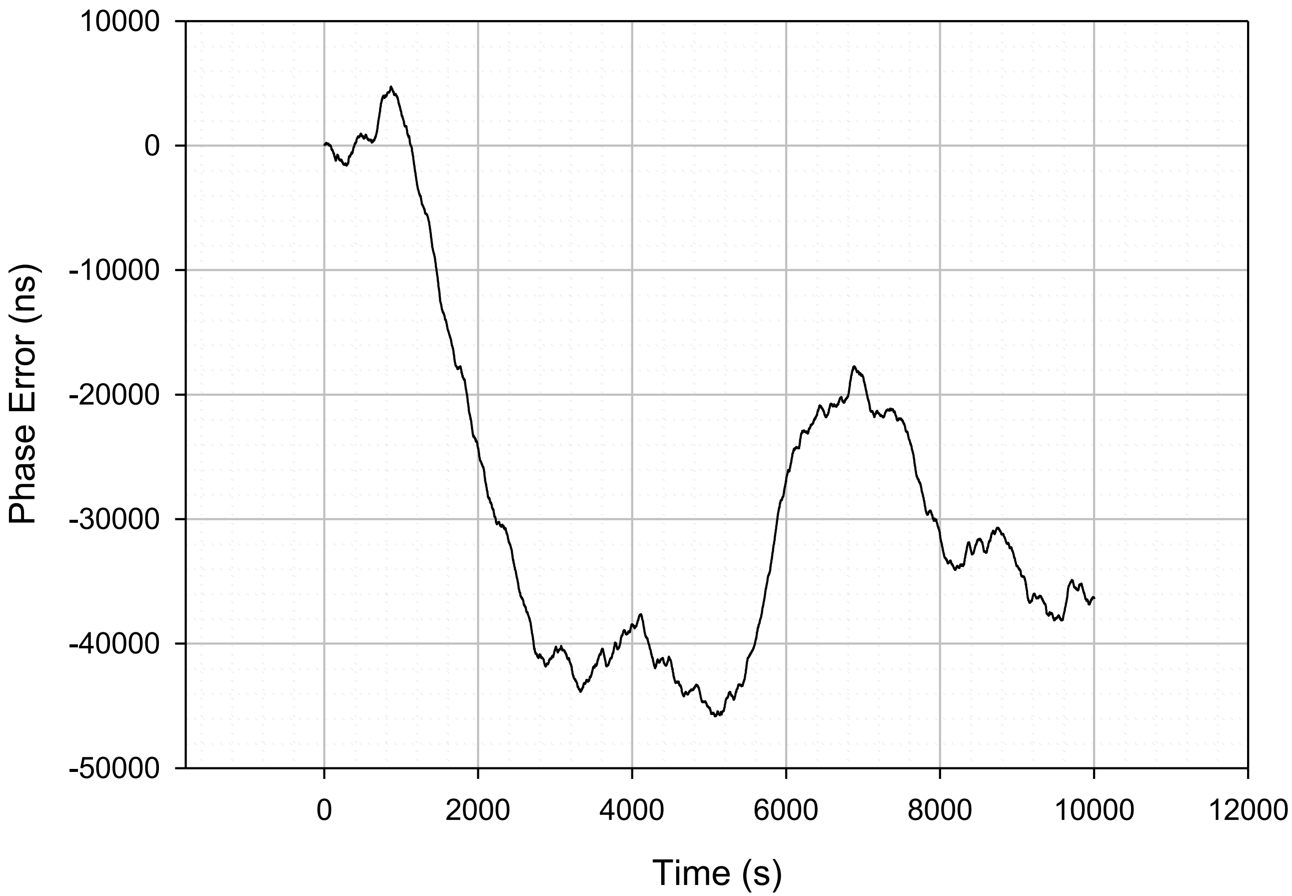
- Simulation time: 3100 s
- Discard the first 100 s when computing statistics (e.g., max|dTE|) to eliminate the effect of any startup transients

# Simulation Cases - 1

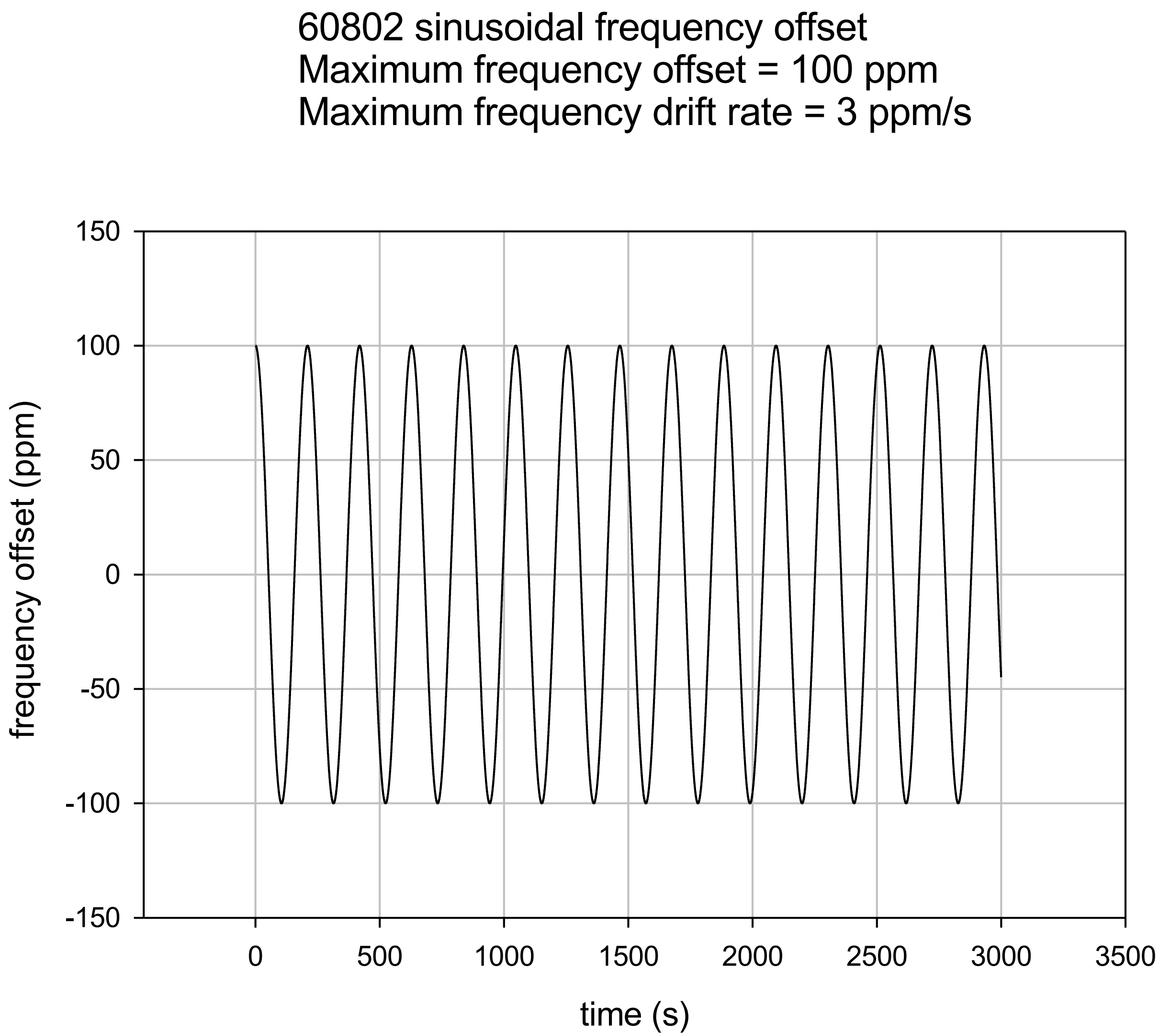
- In the simulation cases, the clock model designation is (models 1, 2, and 3):
  - **Model 1**: flicker frequency modulation (FFM) at level of 802.1AS-2020, Annex B TDEV mask (Figure B-1 of 802.1AS-2020) (see the Appendix for details on how this is simulated, and see [4] for details on the requirement
  - **Model 2**: Sinusoidal phase and frequency variation, with frequency zero-to-peak amplitude of 100 ppm and maximum frequency rate of change of 3 ppm/s
    - Corresponding phase/time offset variation: $x(t) = A \sin(2\pi f t)$, with $A = 3.33\ \mathrm{ms}$ and $f = 4.7746\ \mathrm{mHz}$ (see [4])
  - **Model 3**: Triangular wave frequency variation, with 100 ppm zero-to-peak frequency modulation amplitude and 133.3 s frequency modulation frequency (see [4] for details, and corresponding phase/time offset variation)

# Sample Model 1 Time Error History (from Appendix)

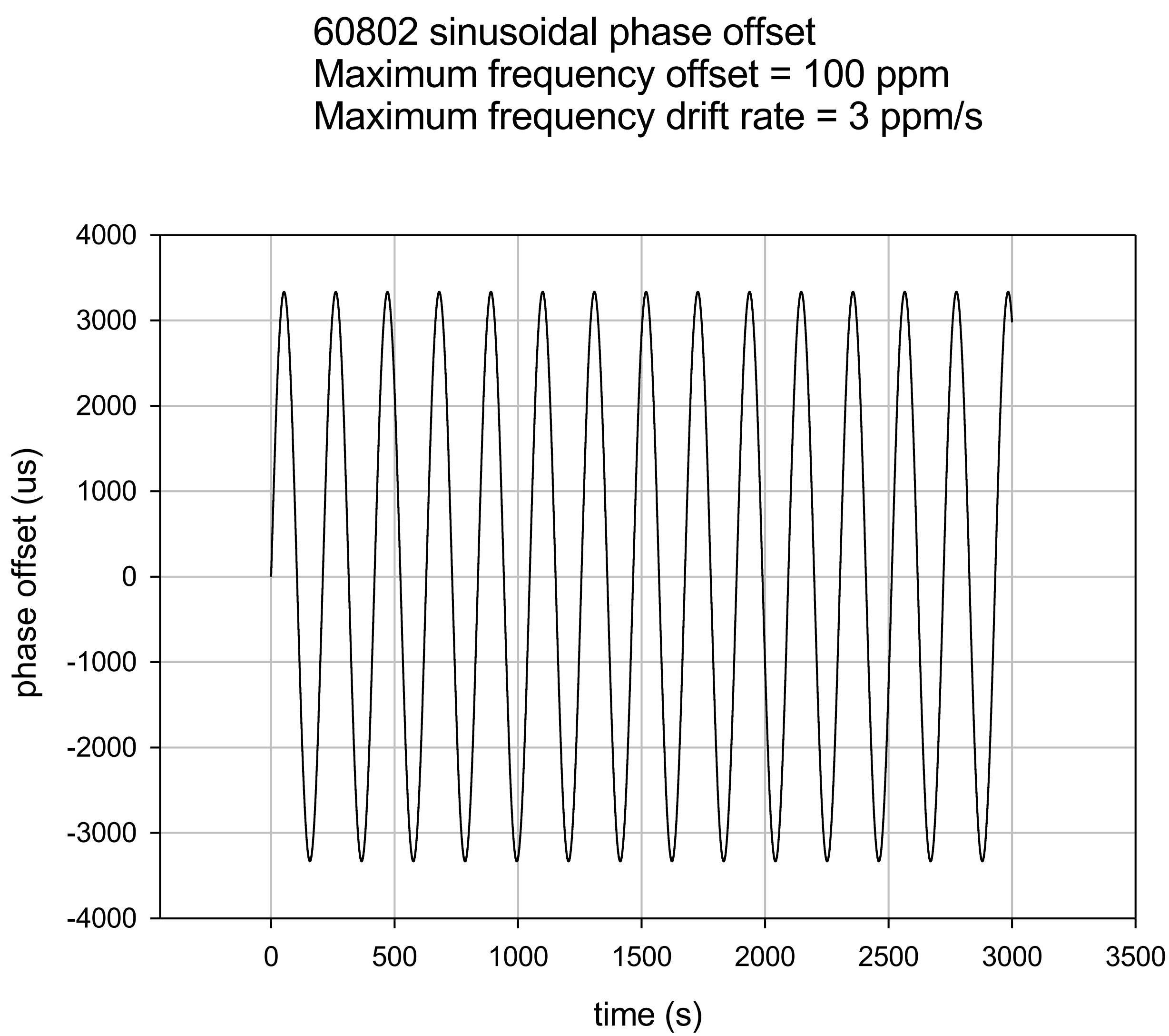
Sample phase error history corresponding to 802.1AS Annex B, Figure B-1 TDEV mask



# Sample Model 3 Frequency Error History (from [4])

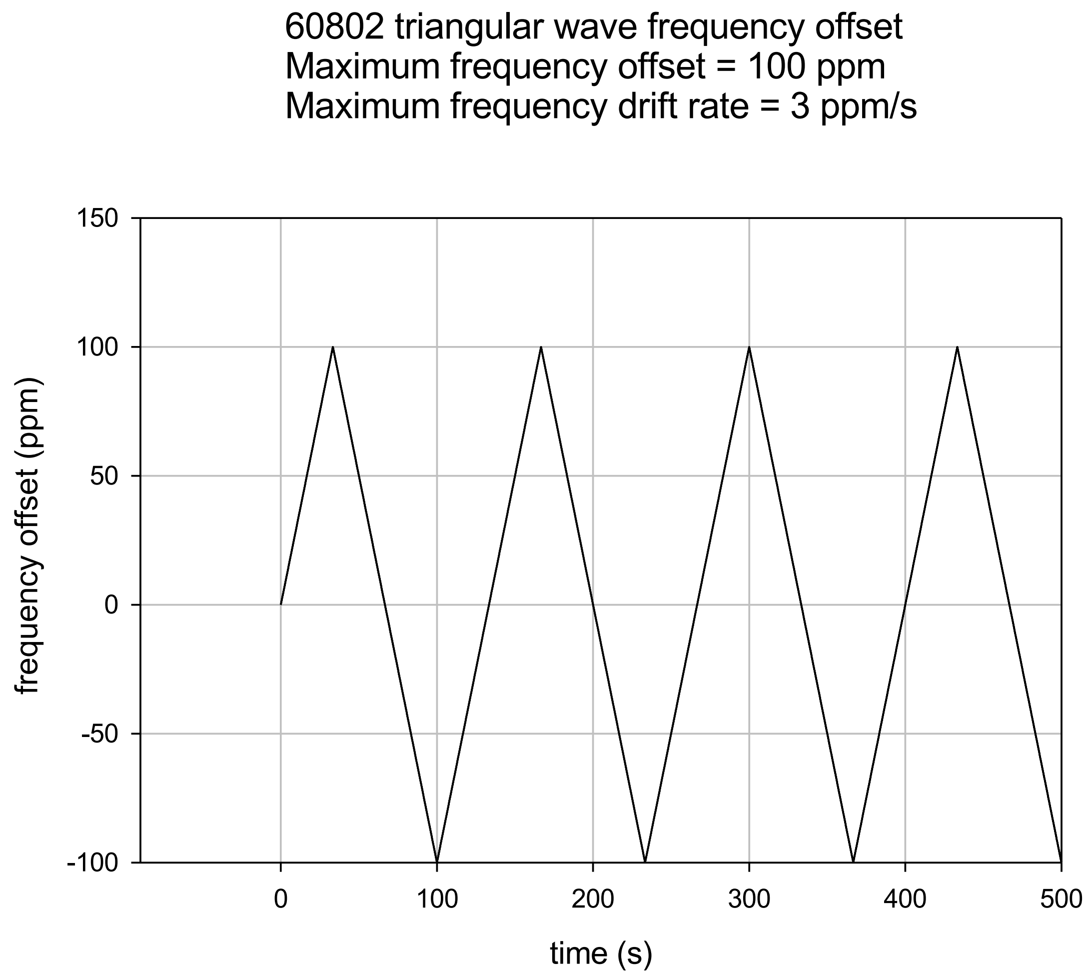


# Sample Model 2 Time Error History (from [4])

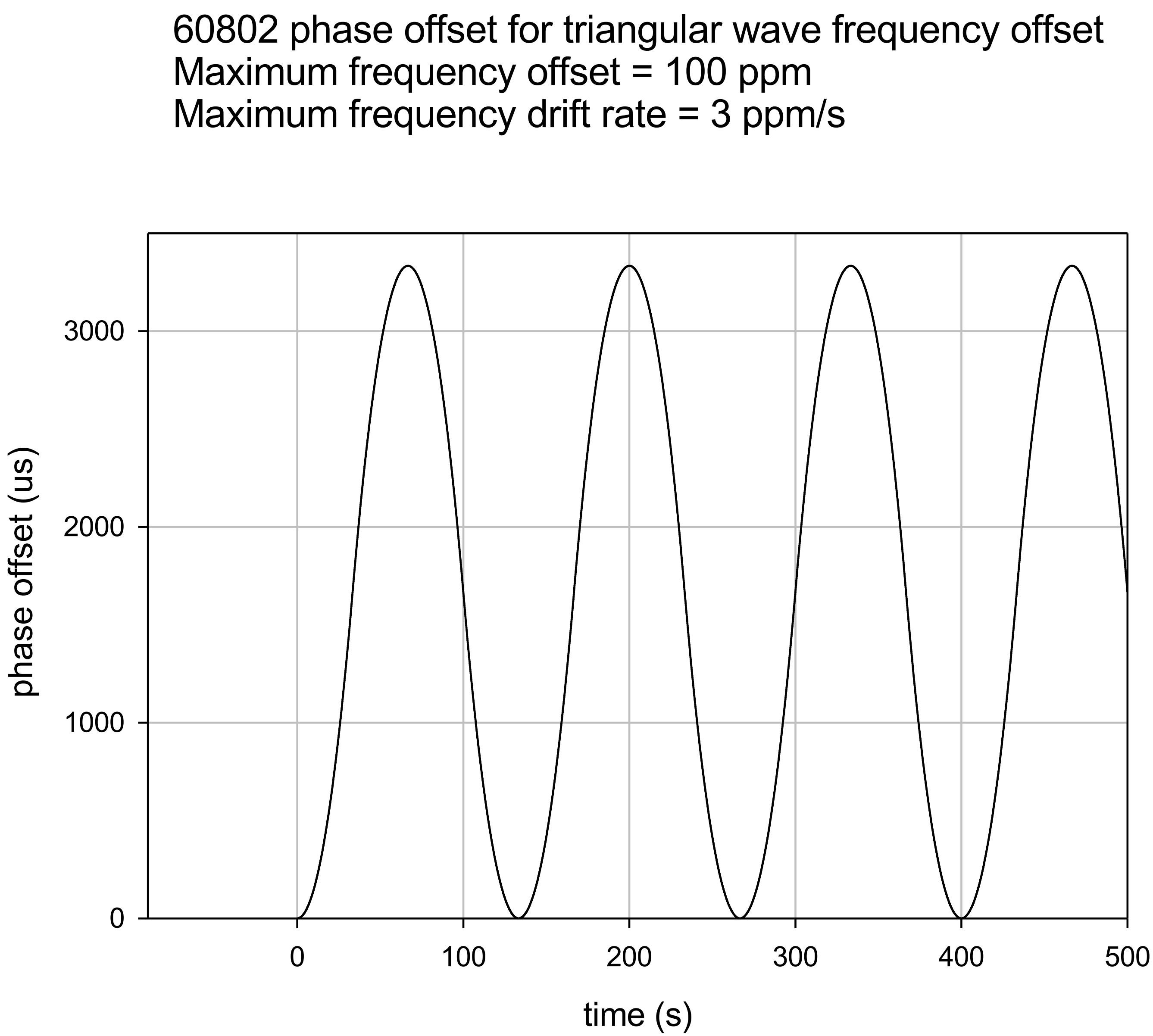



Note that phase and frequency error are 90° ($\pi/2$ rad) out of phase, due to Differentiation (compare with previous slide)

# Sample Model 3 Frequency Error History (from [4])

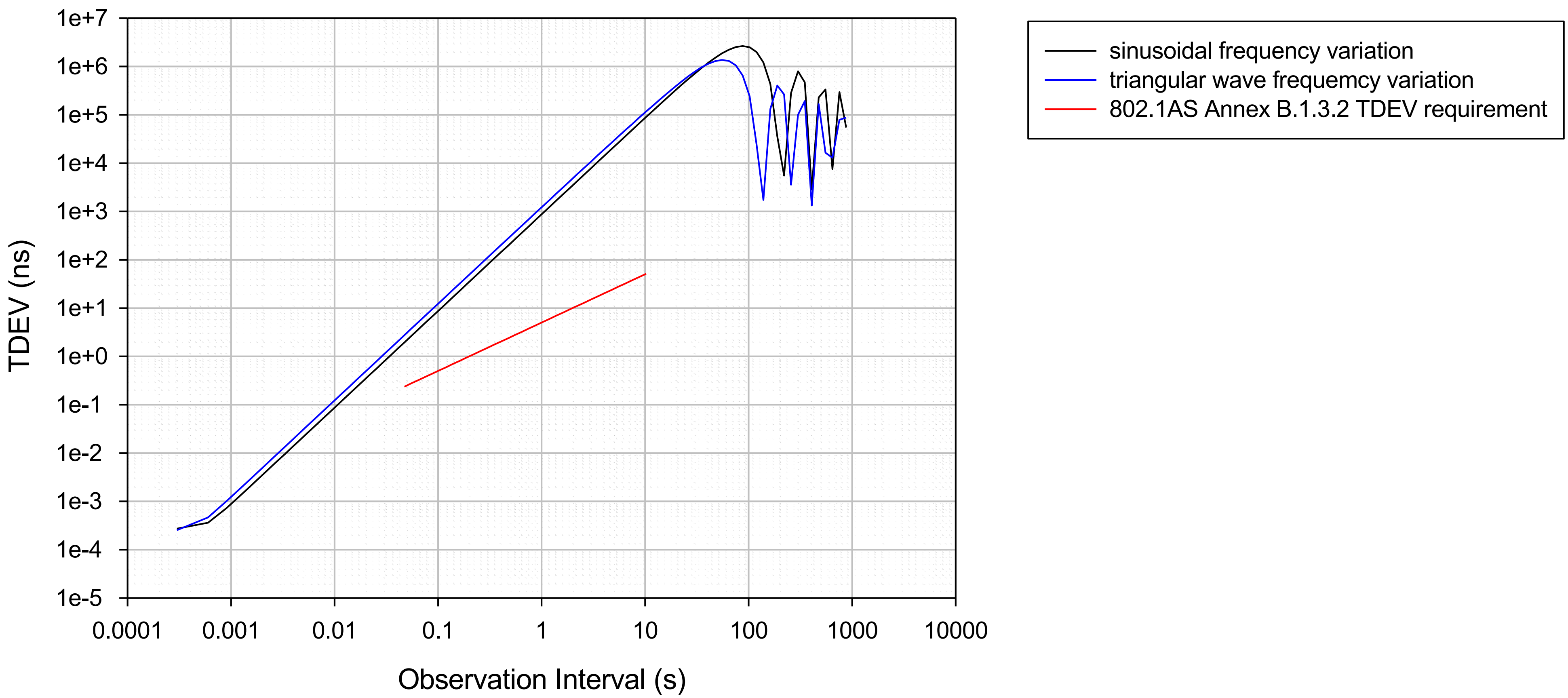


# Sample Model 3 Time Error History (from [4])



# Comparison of 60802 and 802.1AS clock stability (from [4])

Comparison of TDEV for 60802 frequency drift rate (3 ppm/s)
and 802.1AS-2020 TDEV requirement of Annex B.1.3.2
Assumes sinusoidal and triangular wave frequency variation,
with maximum frequency offset of 100 ppm and maximum
frequency drift rate of 3 ppm/s



# Simulation Cases - 2

| Case | Clock Model | Relative phase of clock frequency modulation (models 2 and 3) |
|---|---|---|
| 1 | 1 | Not applicable |
| 2 | 2 | 0 |
| 3 | 2 | random |
| 4 | 3 | 0 |
| 5 | 3 | random |

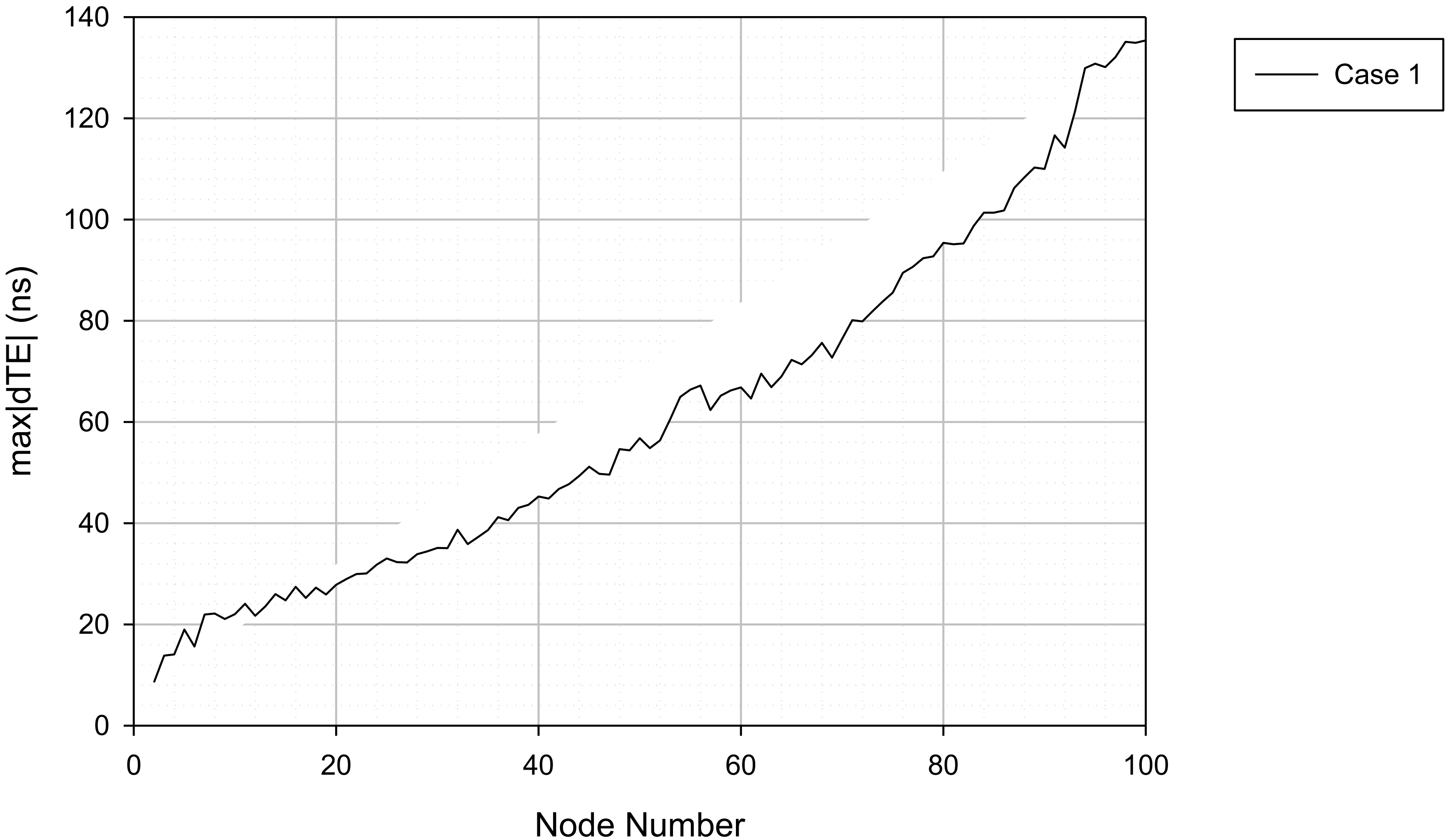
- Note: Relative phase offset of zero corresponds to frequency modulation waveforms at all the PTP Instances being in phase. Relative phase offset random corresponds the relative phases of the sinusoidal or triangular frequency waveform chosen randomly at initialization.

# Simulation Cases - 3

- For each simulation case, a single replication of the simulation case is run
- For each simulation case, we compute:
  - Time history of dynamic time error (dTE) at each PTP instance
  - max|dTE| at each PTP instance
- In future work, multiple replications will be run, in order to obtain statistical confidence intervals for the results
- In future work, MTIE and TDEV will also be computed, in addition to max|dTE|

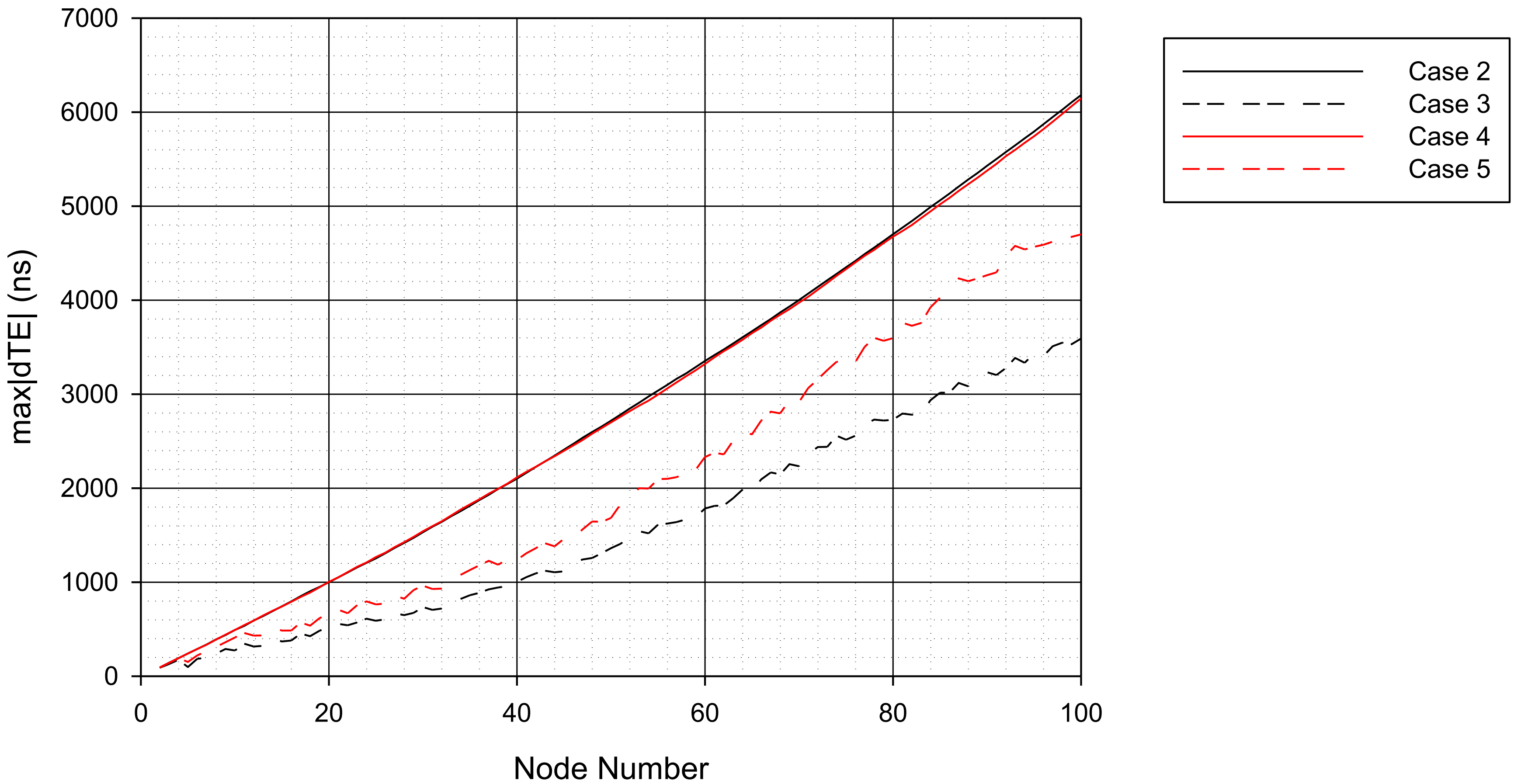
# Max|dTE| results - 1

Simulation Case 1
Single replication of simulation
Clock Model 1 (FFM)



# Max|dTE| results - 2

Simulation Cases 2, 3, 4, and 5
Single replication of simulation
Clock Model 2 (sinusoidal frequency variation, cases 2 and3)
Clock Model 3 (triangular wave freqeuncy variation, cases 4 and 5)

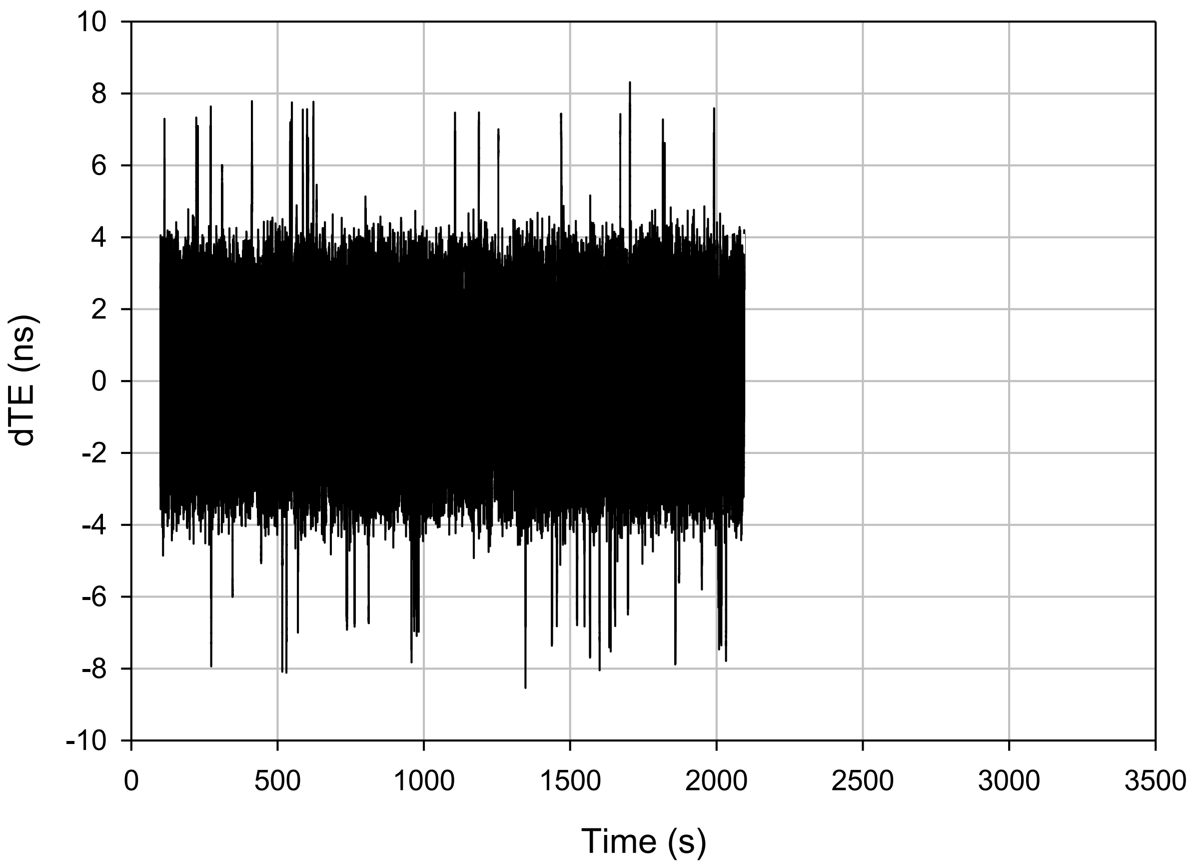


# Max|dTE| results - 3

- Accumulated max|dTE|, after 100 hop, is
  - 136 ns for clock model 1
  - 6400 ns (6.4 μs) for clock models 2 and 3, if the phase variations of the local clocks are all in phase
  - 3600 ns (3.6 μs) and 4700 ns (4.7 μs)) for clock models 2 and 3, if the phase variations of the local clocks are random
  - The fact that the increase in max|dTE| with node number is not strictly monotonic is due to statistical variation
- While the desired1 μs max|dTE| can be met for clock model 1, it is exceeded by a large margin for clock models 2 and 3
- In the following slides, the nature of the accumulated time/phase variation is examined more closely

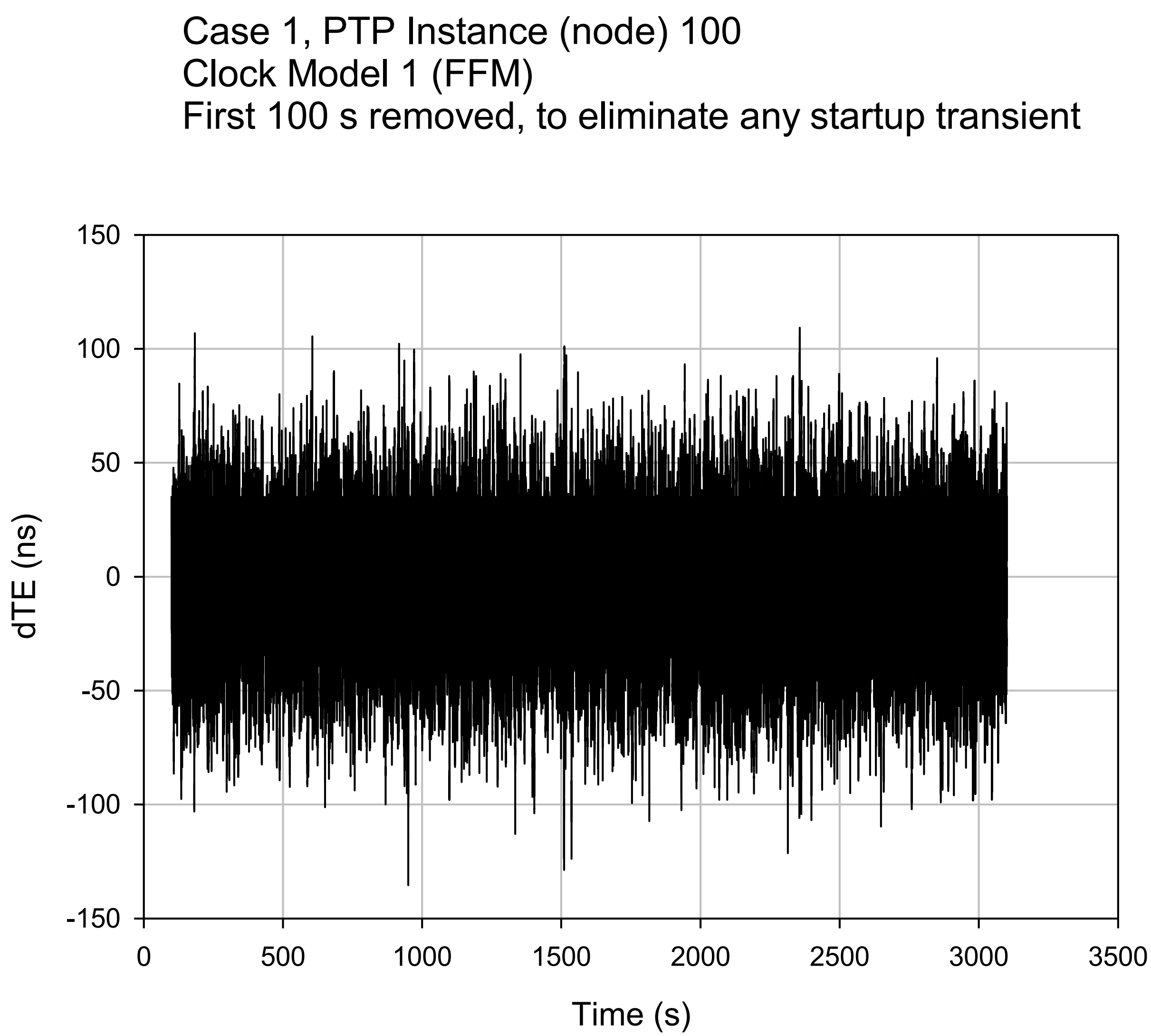
# Case 1 dTE, node 2 (first PTP Instance after GM)

Case 1, PTP Instance (node) 2
Clock Model 1 (FFM)
First 100 s removed, to eliminate any startup transient




Max|dTE| of 8 ns is consistent with plot on slide 27

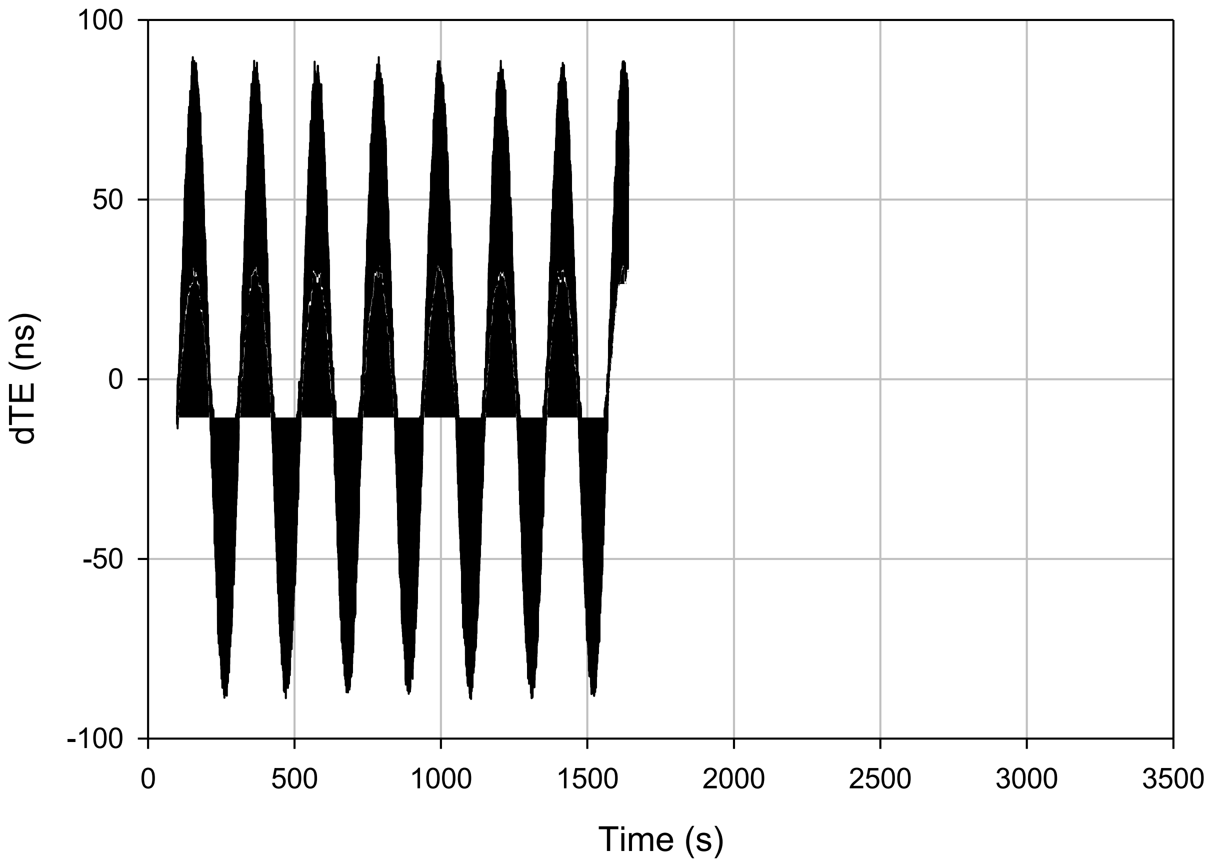
# Case 1 dTE, node 100 (last PTP Instance in HRM)




Max|dTE| of approximately 135 ns is consistent with plot on slide 27

# Case 2 dTE, node 2 (first PTP Instance after GM)

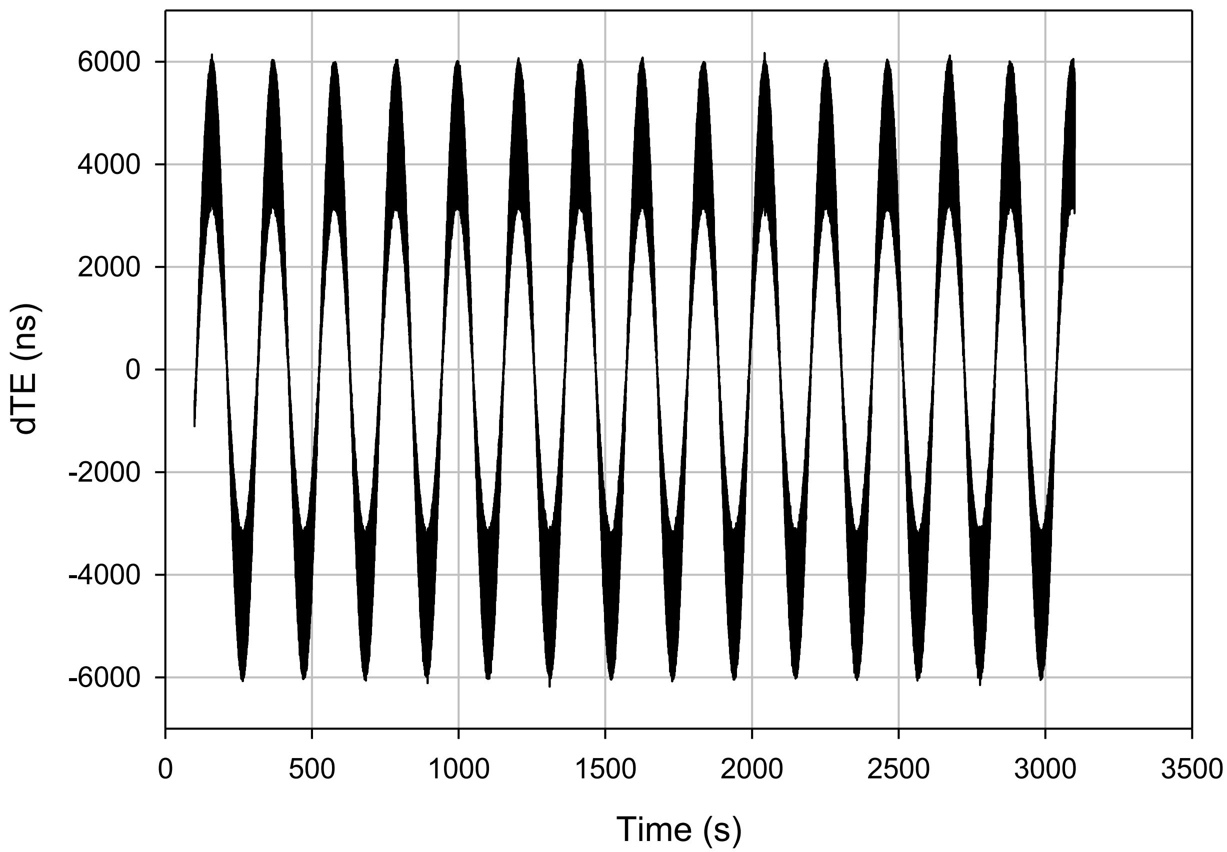
Case 2, PTP Instance (node) 2
Clock Model 2 (Sinusoidal phase and freqeuncy error variation,
with zero phase offset of this variation at each node)
First 100 s removed, to eliminate any startup transient




Max|dTE| of approximately 90 ns is consistent with plot on slide 28

# Case 2 dTE, node 100 (last PTP Instance in HRM)

Case 2, PTP Instance (node) 100
Clock Model 2 (Sinusoidal phase and freqeuncy error variation,
with zero phase offset of this variation at each node)
First 100 s removed, to eliminate any startup transient

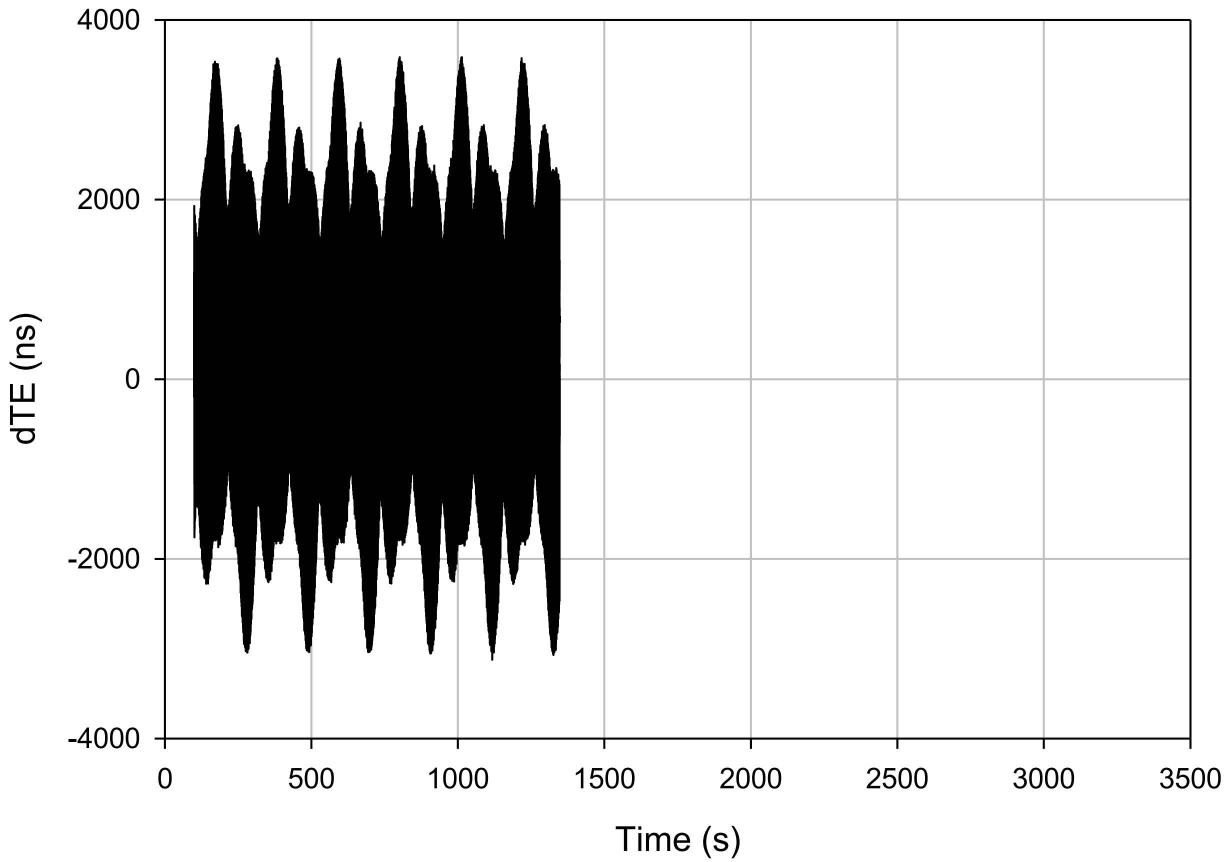



Max|dTE| of approximately 6100 ns is consistent with plot on slide 2 8

Note that qualitative shape of dTE versus time is the same as for node 2. This is because the relative phase of the local clock phase and frequency variation at each node is zero, i.e., the phase variations of each local clock are in phase.

# Case 3 dTE, node 100 (last PTP Instance in HRM)

Case 3, PTP Instance (node) 100
Clock Model 2 (Sinusoidal phase and freqeuncy error variation,
with random phase offset of this variation at each node)
First 100 s removed, to eliminate any startup transient

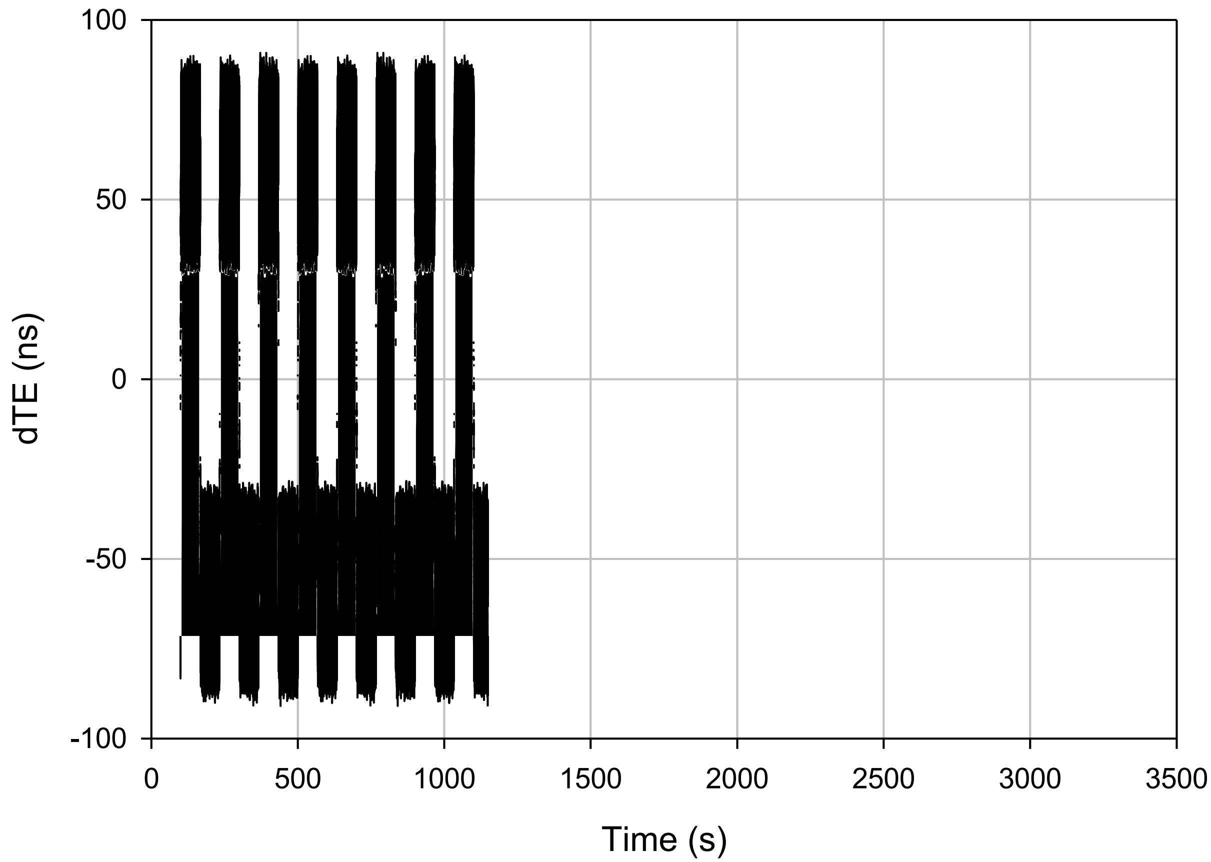



- Max|dTE| of approximately 3600 ns is consistent with plot on slide 28

Note that qualitative shape of dTE versus time is different from node 2. This is because the relative phase of the local clock phase and frequency variation at each node is nonzero, i.e., the phase variations of each local clock are out of phase.

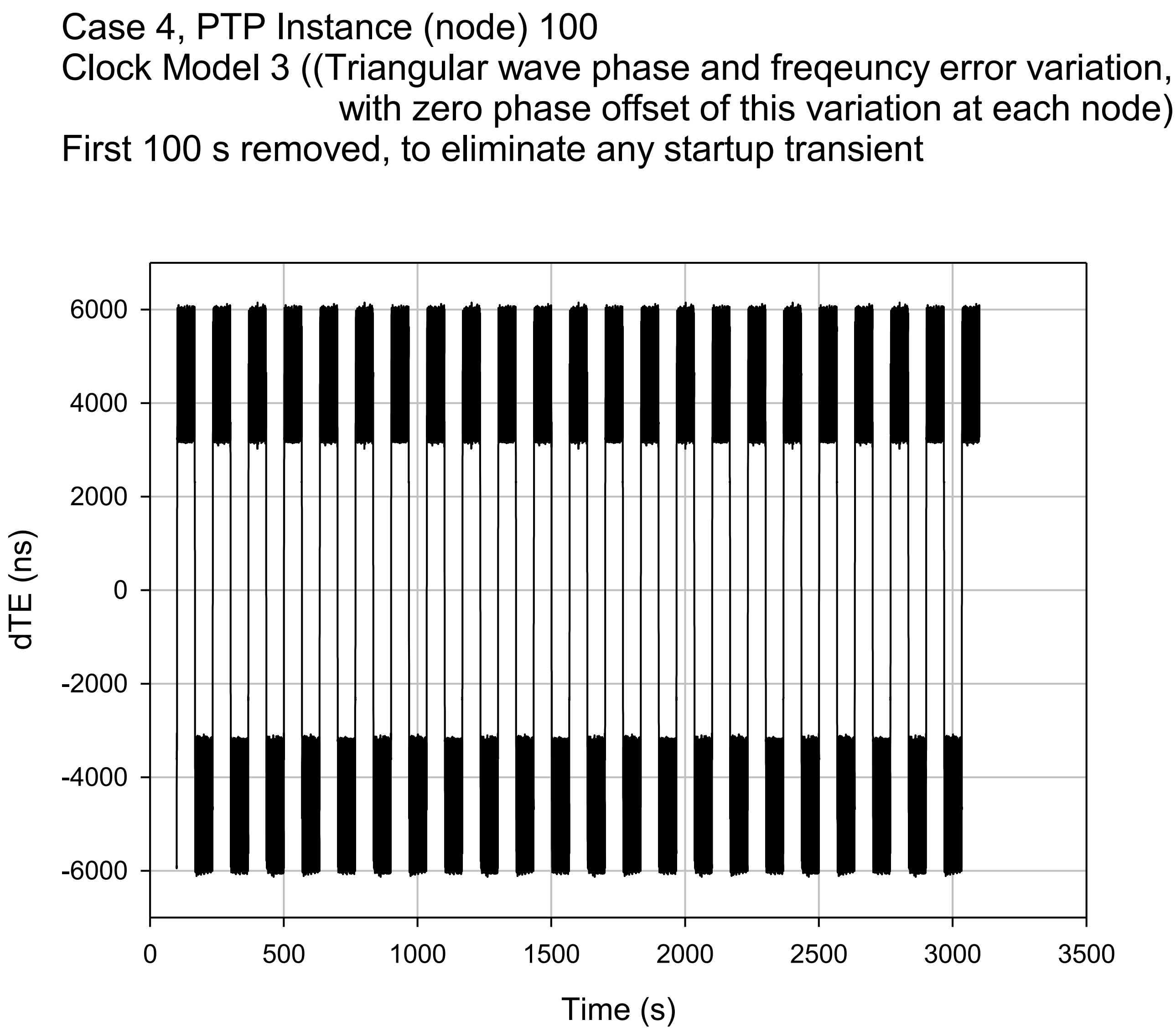
# Case 3 dTE, node 2 (first PTP Instance after GM)

Case 4, PTP Instance (node) 2
Clock Model 3 (Triangular wave phase and freqeuncy error variation,
with zero phase offset of this variation at each node)
First 100 s removed, to eliminate any startup transient




Max|dTE| of approximately 90 ns is consistent with plot on slide 28

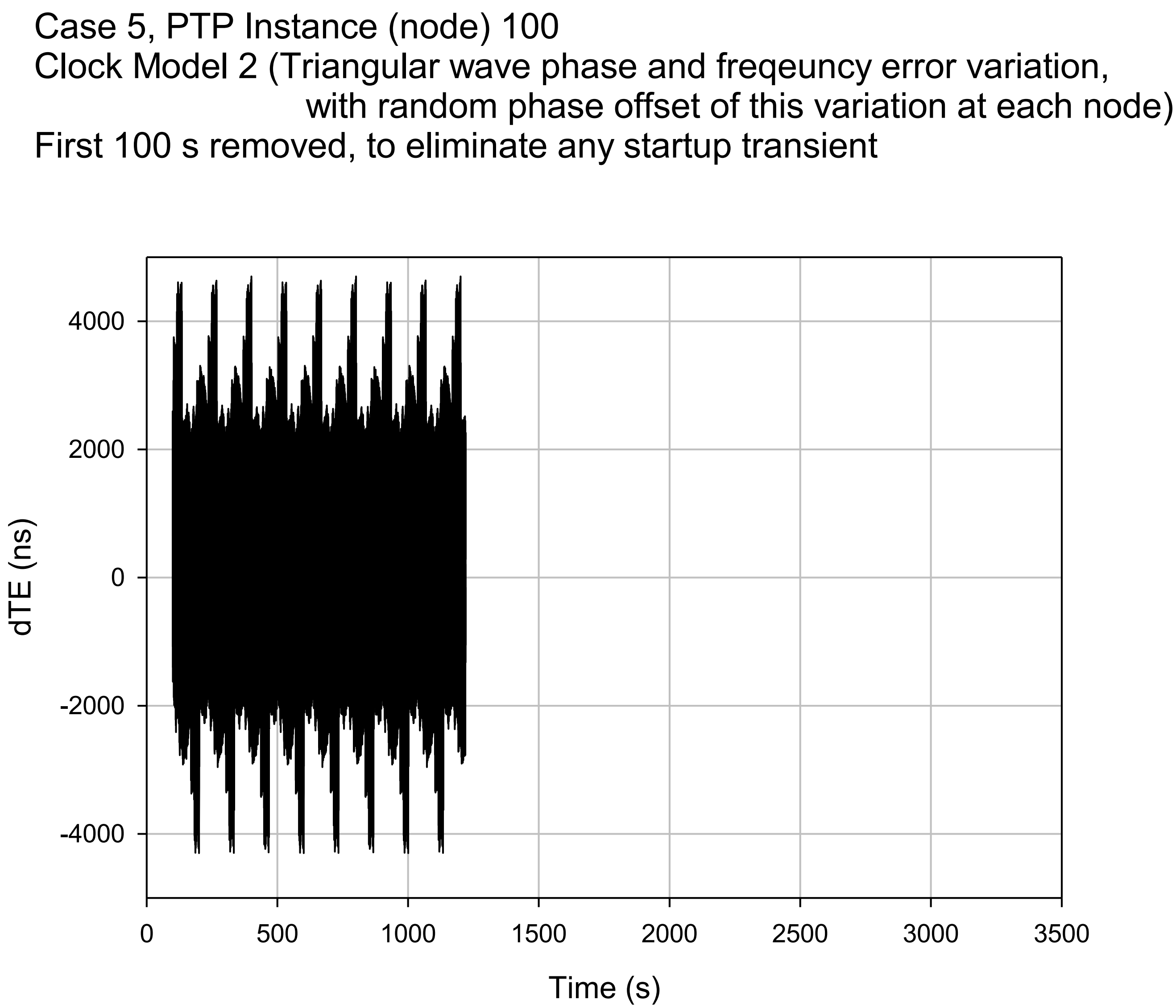
# Case 3 dTE, node 100 (last PTP Instance in HRM)




- Max|dTE| of approximately 6100 ns is consistent with plot on slide 2 8

Note that qualitative shape of dTE versus time is the same as for node 2. This is because the relative phase of the local clock phase and frequency variation at each node is zero, i.e., the phase variations of each local clock are in phase.

# Case 4 dTE, node 100 (last PTP Instance in HRM)




- Max|dTE| of approximately 4700 ns is consistent with plot on slide 28
- Note that qualitative shape of dTE versus time is different from node 2. This is because the relative phase of the local clock phase and frequency variation at each node is nonzero, i.e., the phase variations of each local clock are out of phase.

# Detailed Consideration of Case 2, Node 2

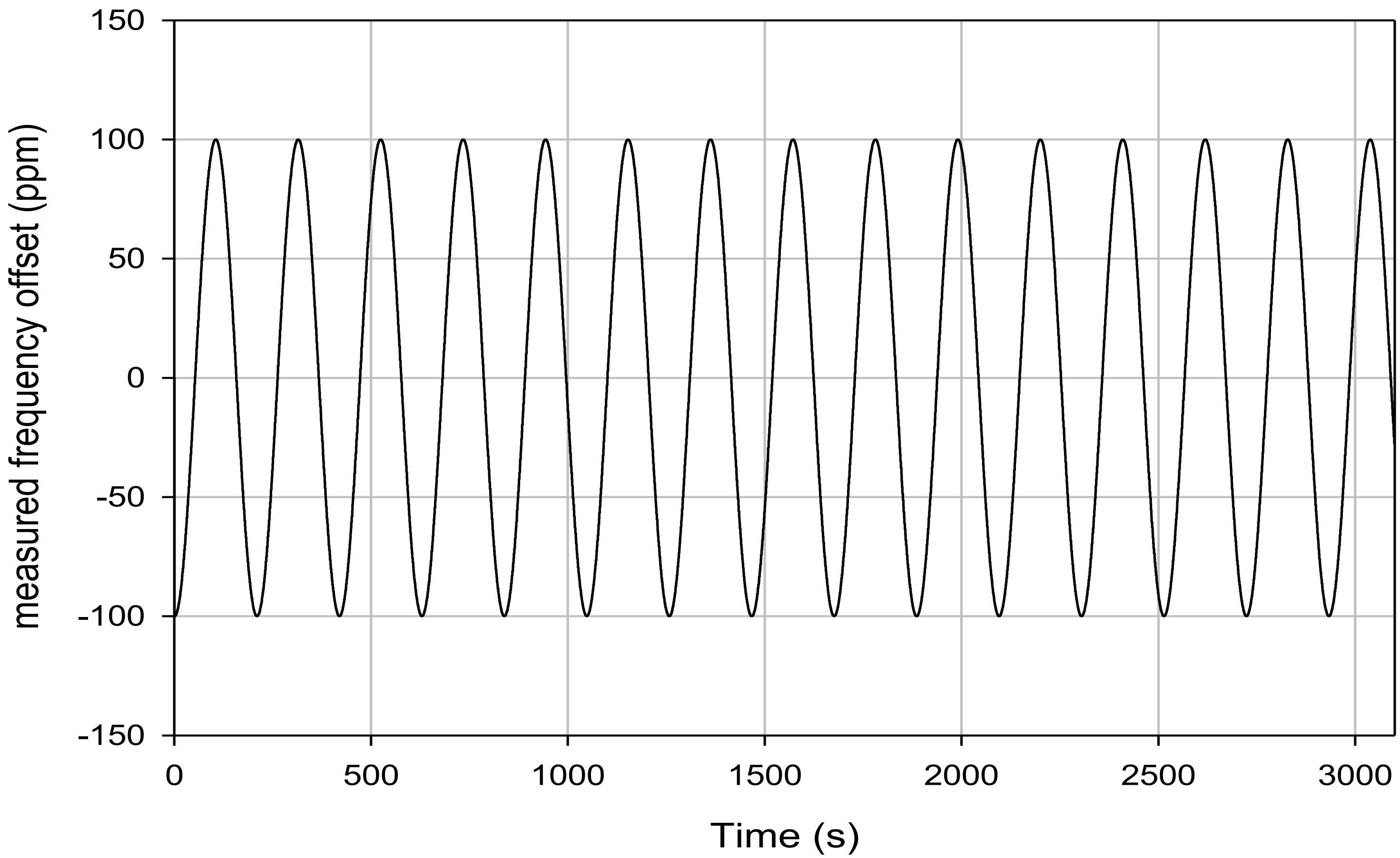
- We now consider Case 2, node 2 in detail
  - Node 2 is considered because here we can consider the effect of local clock frequency offset, phase adjustment on receipt of a Sync message, and filtering, without regard to accumulation of time error over multiple nodes
- The full time history of filtered phase/time error at node 2 is shown on slide 33
- The following slides show the time history of measured frequency offset at node 2 (based on the neighborRateRatio measurement using Pdelay messages), followed by the time history phase/time error at node 2, for successively smaller intervals (i.e., zoomed in):
  - 0 – 3100 s (measured frequency offset only, since phase/time error is already shown on slide 33)
  - 450 s – 550 s
  - 500 s – 520 s
  - 510 s – 515 s
  - 511 s – 512 s (phase/time error only)

# Detailed Consideration of Case 2, Node 2

- For the time error history results, the filtered time history of phase/time error is shown for each case, followed by the unfiltered phase/time error (i.e., before the 3.78 Hz endpoint filter)

# Case 2, node 2 measured frequency offset

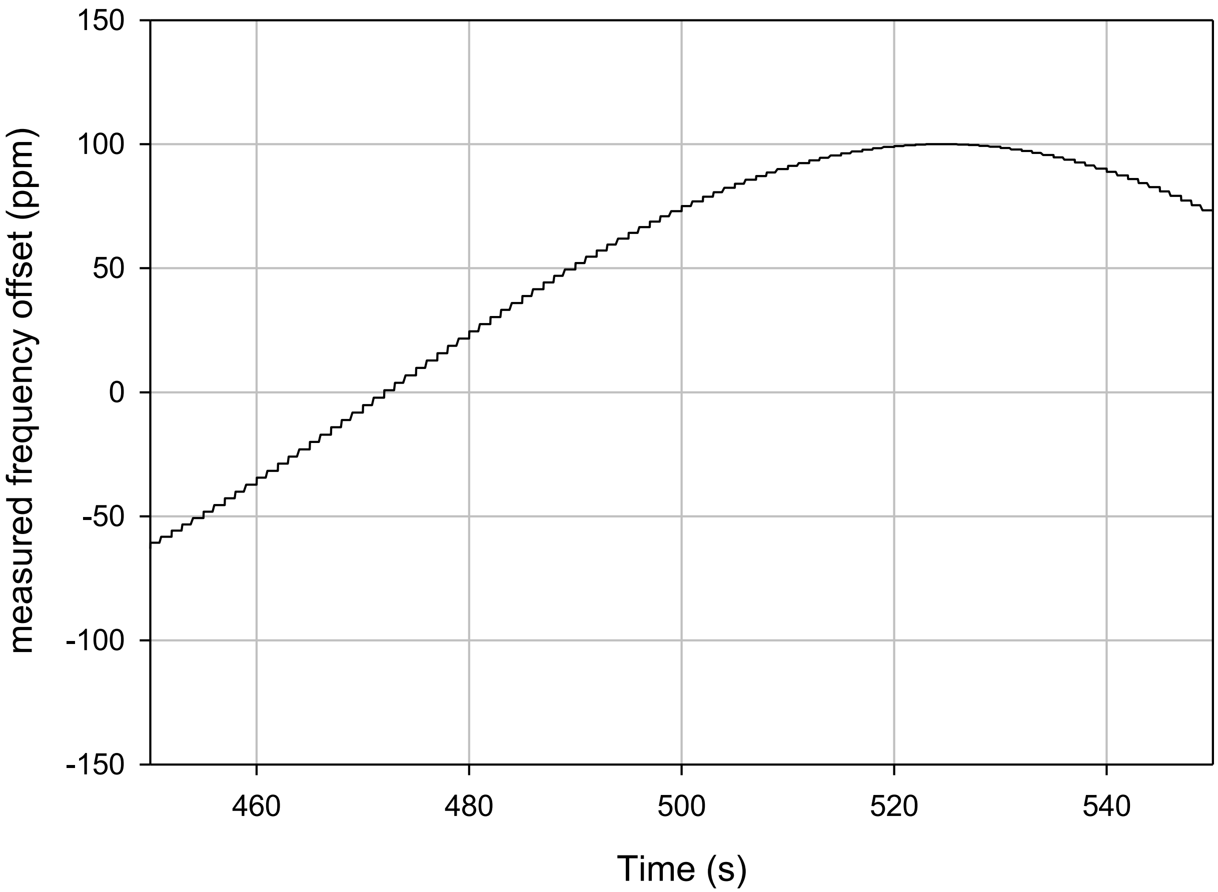
Case 2, PTP Instance (node) 2
Measured frequency offset of node 2 relative to node 1
Clock Model 2 (Sinusoidal phase and freqeuncy error variation,
with zero phase offset of this variation at each node)




Note that apparent "vertical jumps" are "Pixelation" effects

# Case 2, node 2 measured frequency offset, detail of 450 s - 550 s

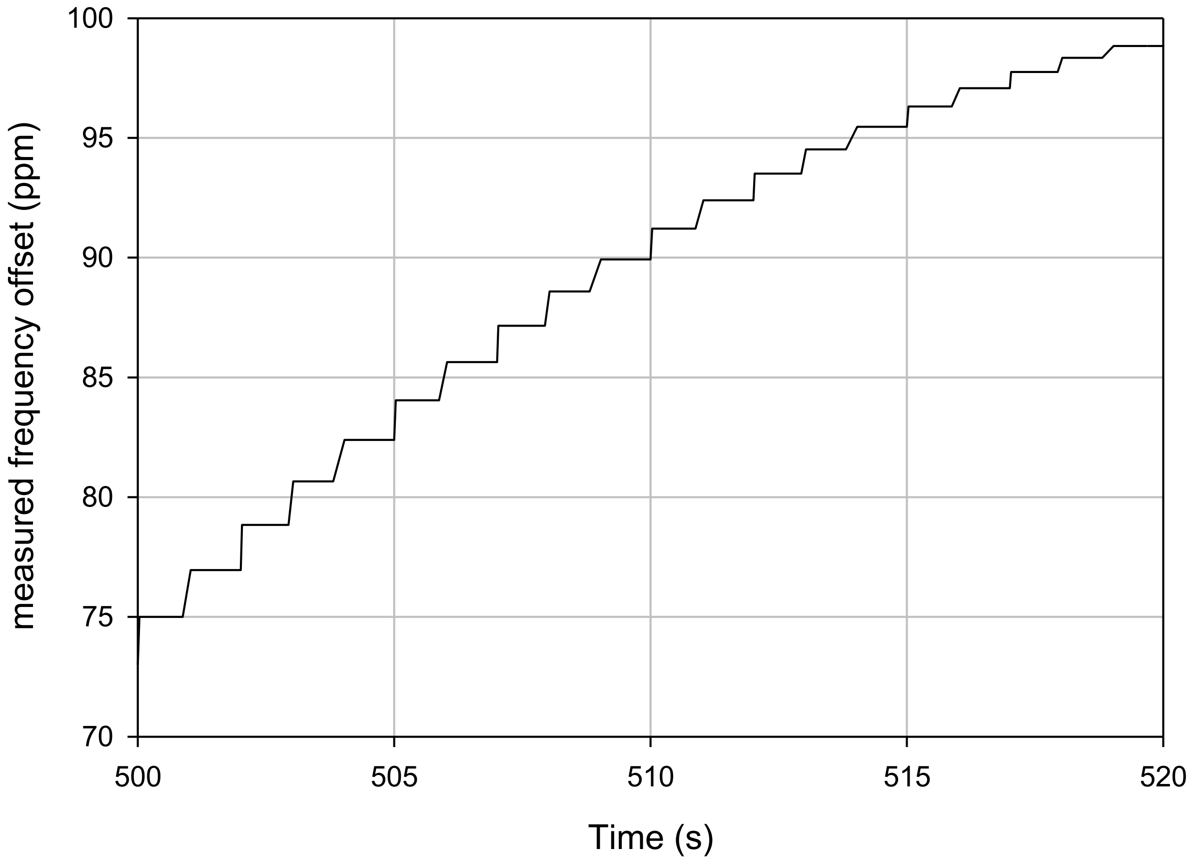
Case 2, PTP Instance (node) 2
Measured frequency offset of node 2 relative to node 1
Clock Model 2 (Sinusoidal phase and freqeuncy error variation,
with zero phase offset of this variation at each node)
Detail of 450 s - 550 s




Vertical jumps are
Due to frequency offset
being measured only
once per second (will
be apparent in
following slides)

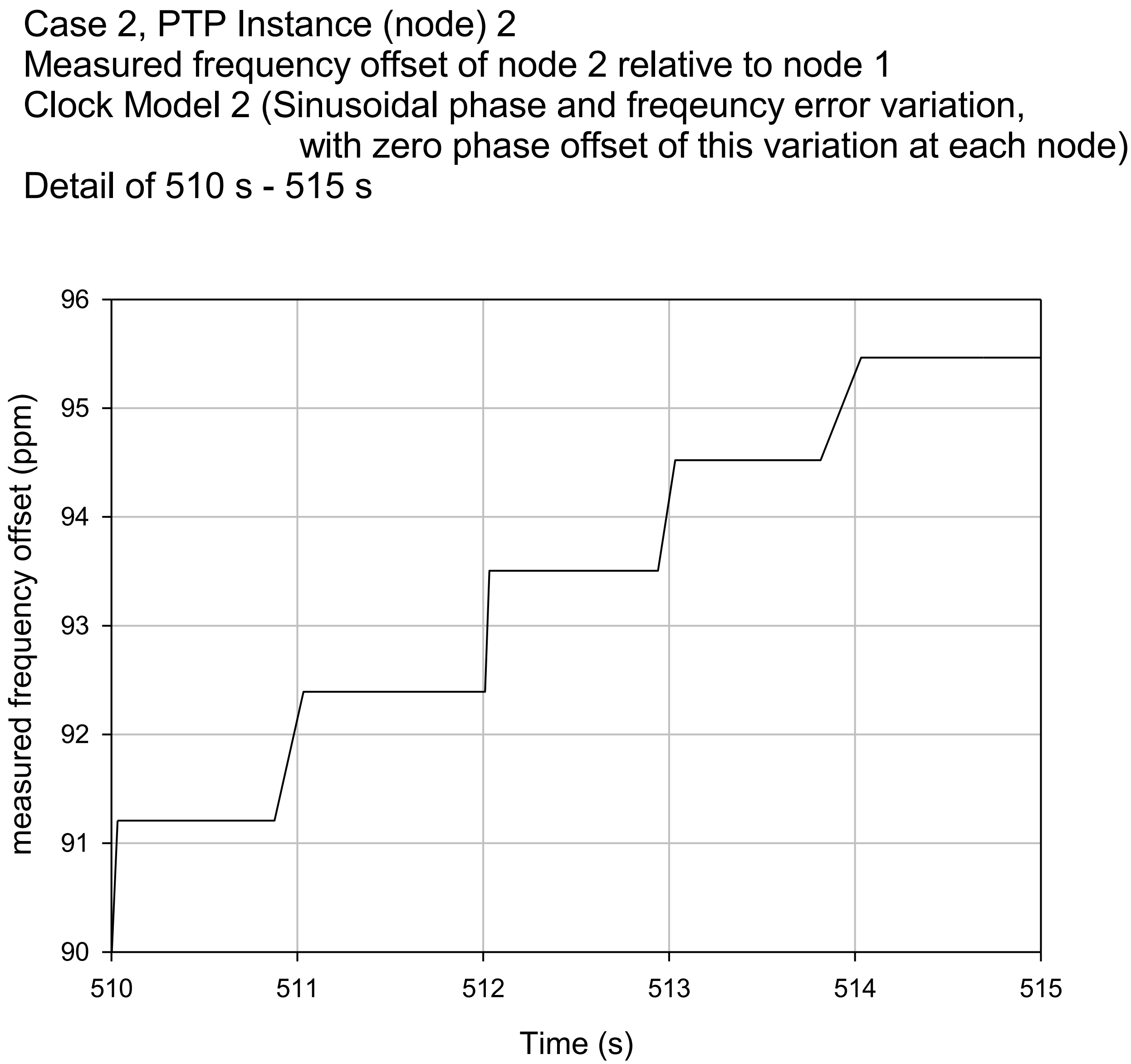
# Case 2, node 2 measured frequency offset, detail of 500 s - 520 s

Case 2, PTP Instance (node) 2
Measured frequency offset of node 2 relative to node 1
Clock Model 2 (Sinusoidal phase and freqeuncy error variation,
with zero phase offset of this variation at each node)
Detail of 500 s - 520 s




Vertical jumps are due to frequency offset being measured only once per second

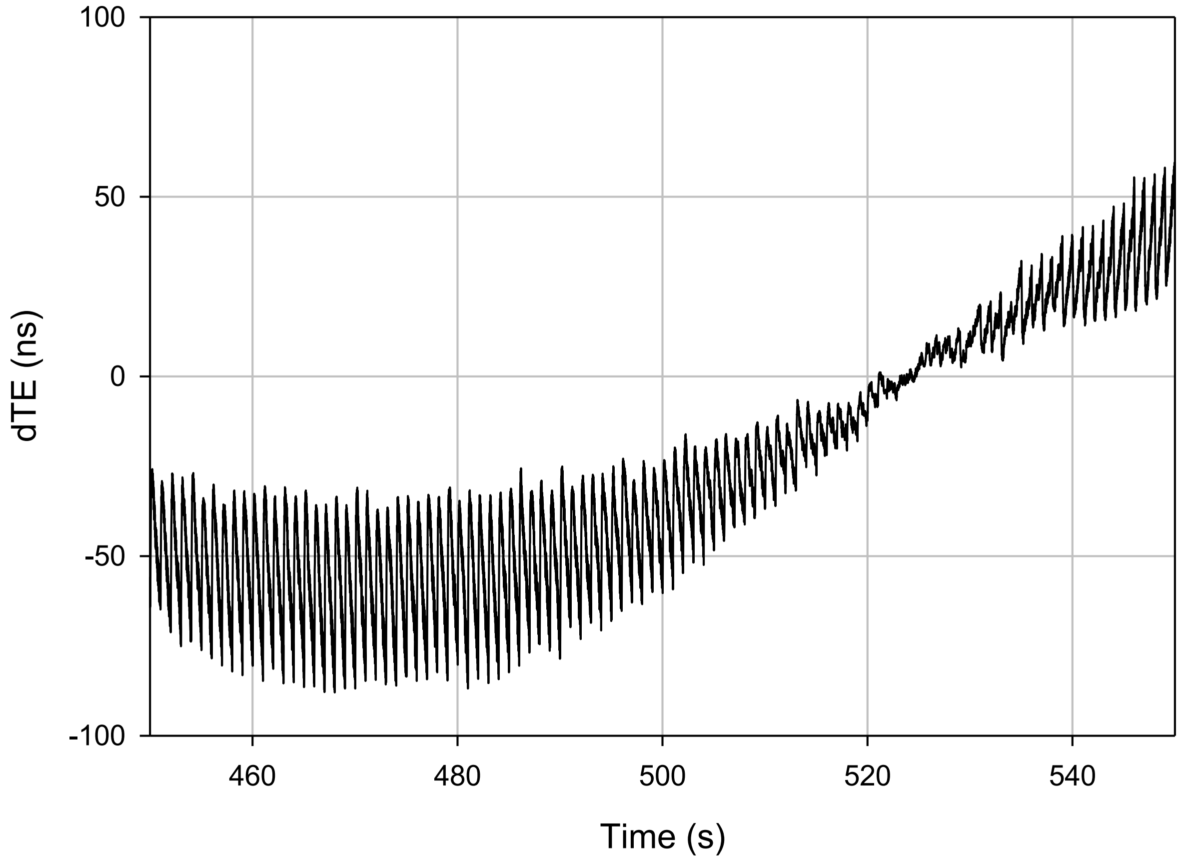
# Case 2, node 2 measured frequency offset, detail of 510 s - 515 s




Vertical jumps are due to frequency offset being measured only once per second

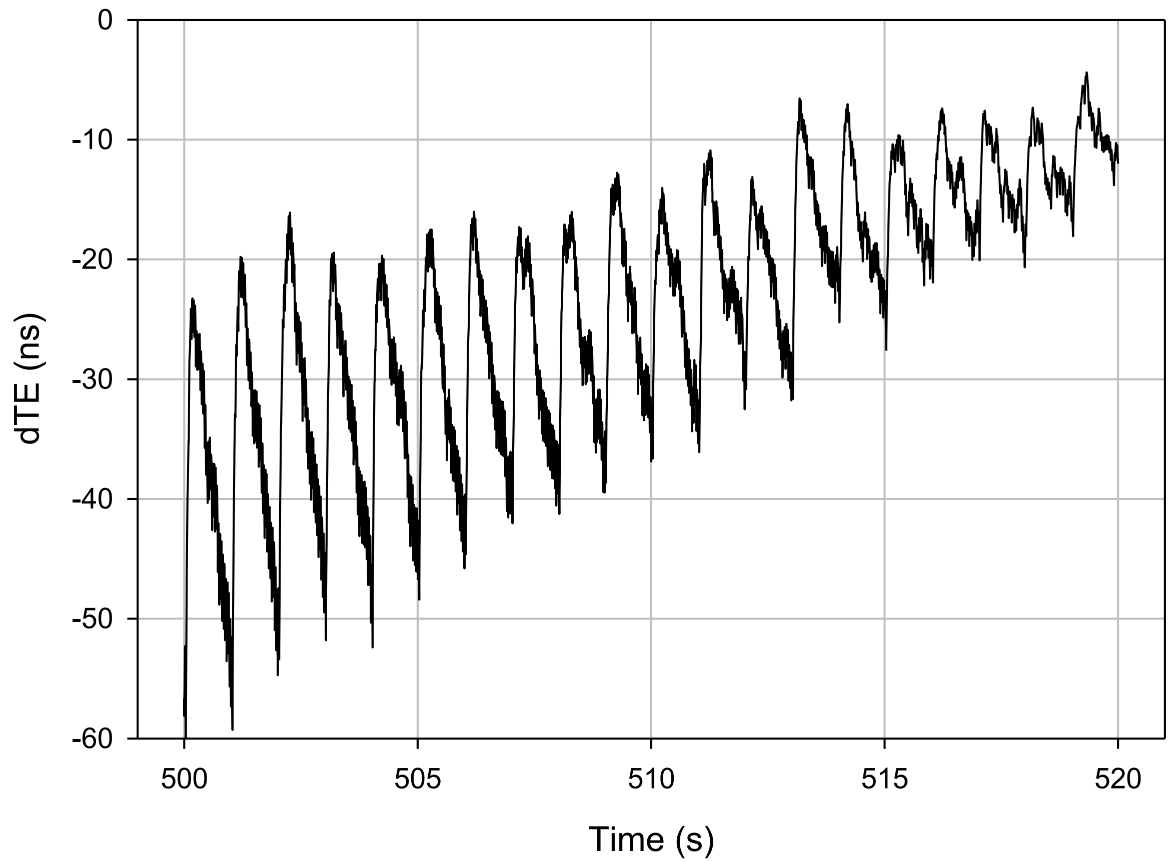
# Case 2, node 2 filtered time error - detail of 450 - 550 s

Case 2, PTP Instance (node) 2
Clock Model 2 (Sinusoidal phase and freqeuncy error variation,
with zero phase offset of this variation at each node)
Detail of 450 s - 550 s

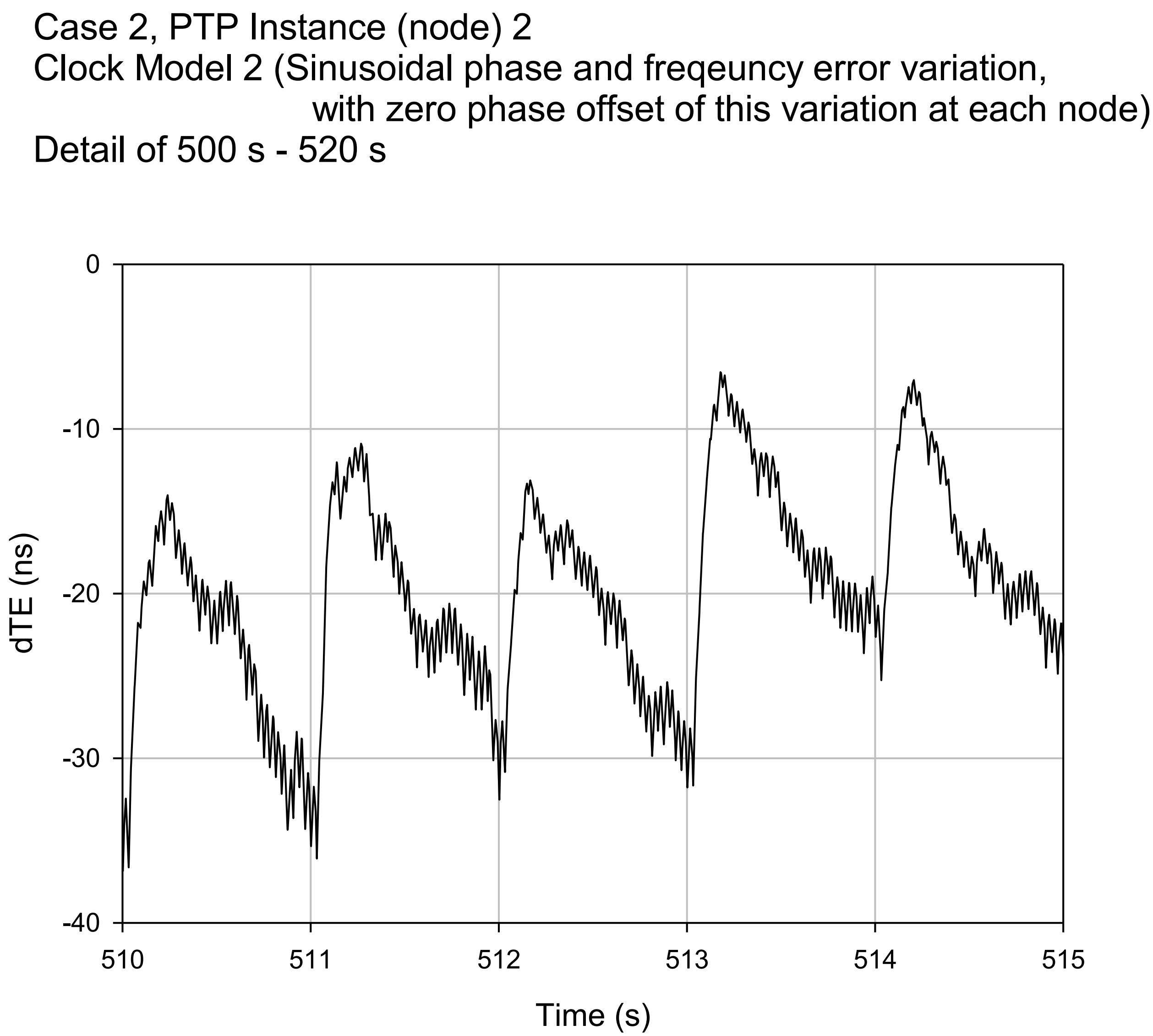


# Case 2, node 2 filtered time error - detail of 500 - 520 s

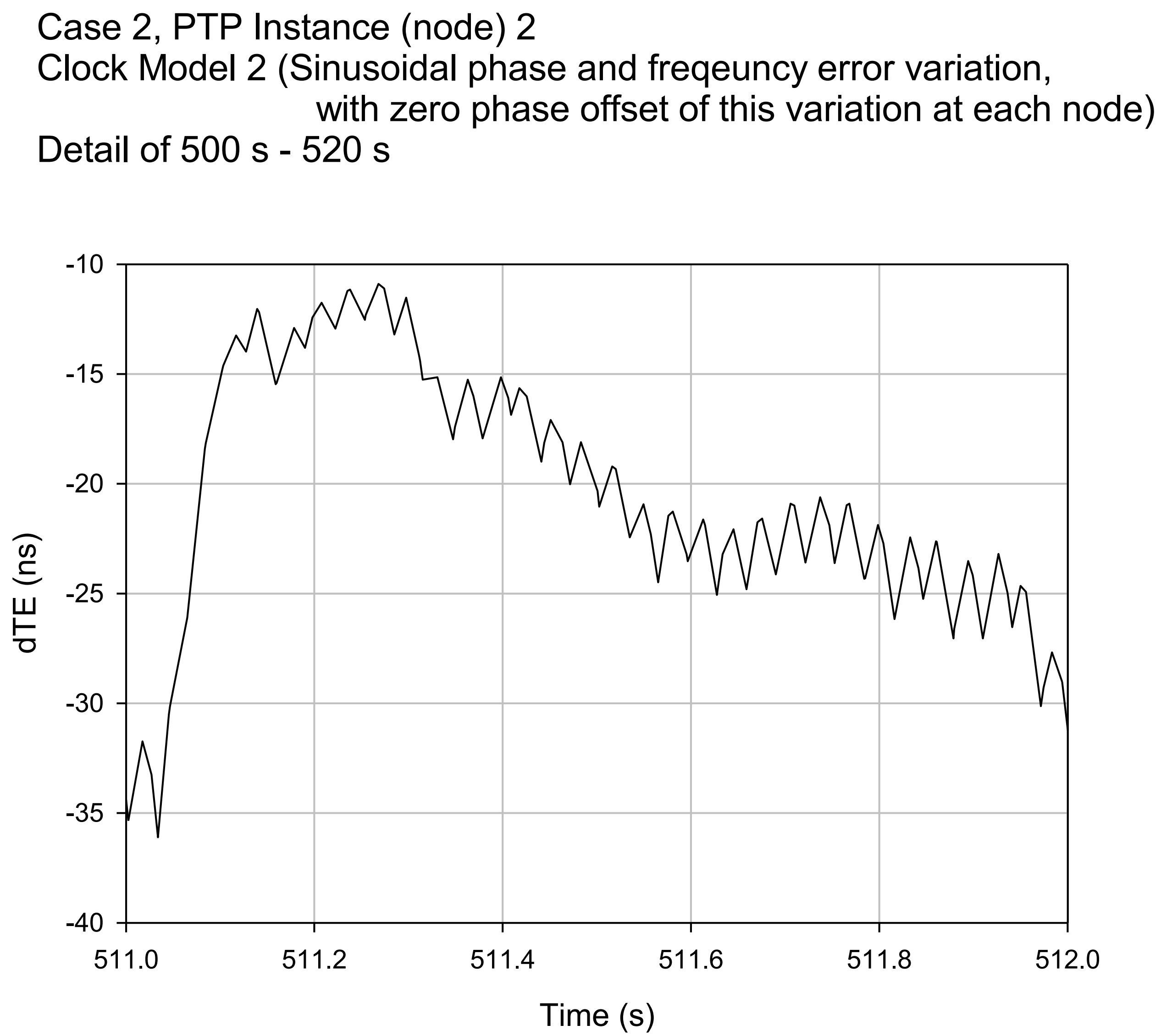
Case 2, PTP Instance (node) 2
Clock Model 2 (Sinusoidal phase and freqeuncy error variation,
with zero phase offset of this variation at each node)
Detail of 500 s - 520 s



# Case 2, node 2 filtered time error - detail of 510 - 515 s

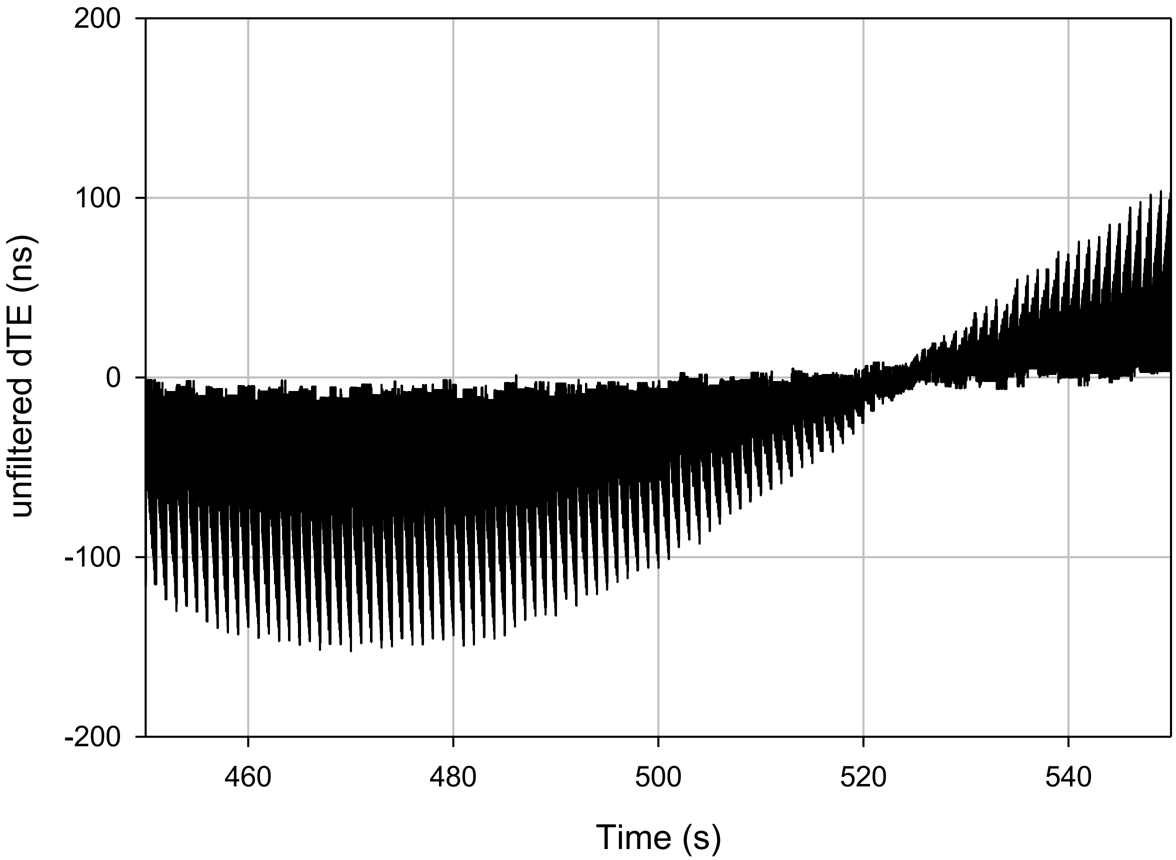


# Case 2, node 2 filtered time error - detail of 511 - 512 s



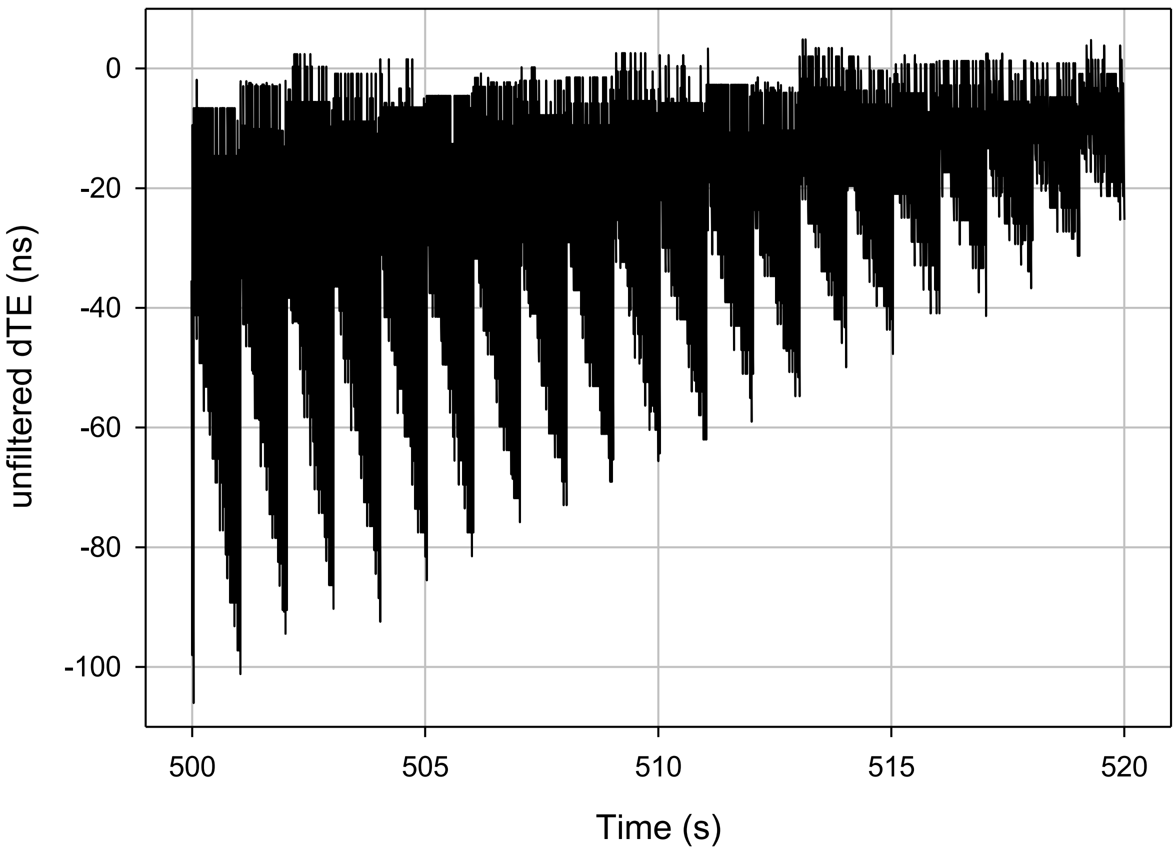
# Case 2, node 2 unfiltered time error - detail of 450 - 550 s

Case 2, PTP Instance (node) 2
Unfiltered time error
Clock Model 2 (Sinusoidal phase and freqeuncy error variation,
with zero phase offset of this variation at each node)
Detail of 450 s - 550 s



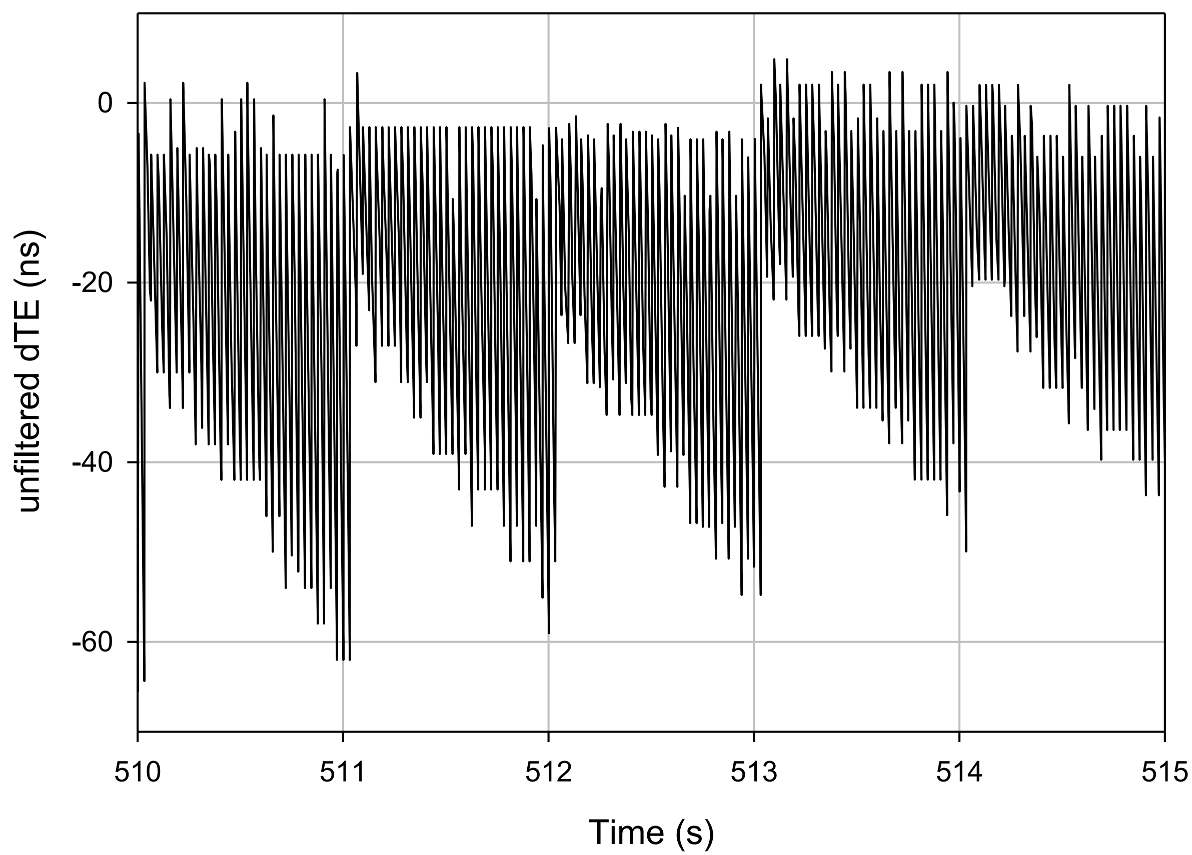
# Case 2, node 2 unfiltered time error - detail of 500 - 520 s

Case 2, PTP Instance (node) 2
Unfiltered time error
Clock Model 2 (Sinusoidal phase and freqeuncy error variation,
with zero phase offset of this variation at each node)
Detail of 500 s - 520 s

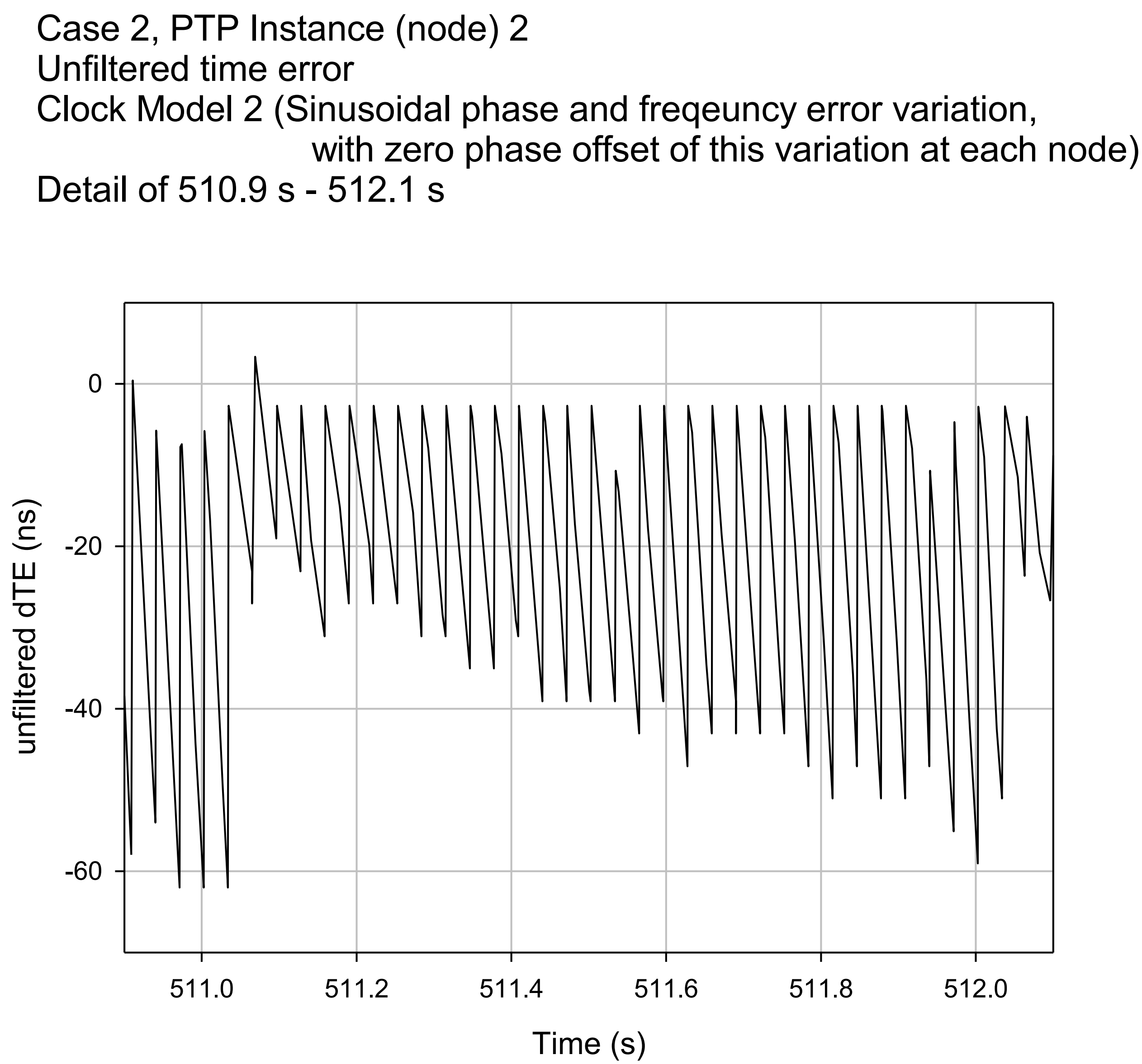


# Case 2, node 2 unfiltered time error - detail of 510 - 515 s

Case 2, PTP Instance (node) 2
Unfiltered time error
Clock Model 2 (Sinusoidal phase and freqeuncy error variation,
with zero phase offset of this variation at each node)
Detail of 510 s - 515 s



# Case 2, node 2 unfiltered time error - detail of 510.9 - 512.1 s



# Case 2, Node 2 – Discussion - 1

- On slides 46, 47, 50, and 51, the jumps every 1 s correspond to updates of cumulative rateRatio
  - Actually, it is neighborRateRatio that is measured every 1 s, but this is then used to compute rateRatio for the next Sync message
    - In any case, rateRatio and neighborRateRatio are equal at the input to node 2 because the GM is the immediate upstream node (node 1)
- On slides 51 and 52, the smaller jumps in between the large jumps every 1 s correspond to receipt of a Sync message
  - The time between these smaller jumps is 0.03125 s (i.e., the sync interval)
  - This corresponds to a Sync rate of 32 messages/s (and there are approximately 0.2 s/ 0.03125 s = 6.4 smaller jumps between the 0.2 s gradations in slide 52
- In slides 45-48, the jumps of slides 49-52, respectively are filtered by the 3.78 Hz (with 1.049 dB gain peaking) endpoint filter (note that this only filters the time recovered at node 2; it does not filter the computations of originTimestamp for Sync messages sent by node 2

# Case 2, Node 2 – Discussion - 2

- The longer time constant (the filter has 2 time constants because it is 2nd order) of this filter is approximately $1/[(2\pi)(3.78\text{ Hz}) = 0.042$ s
  - 3 time constants is 0.126 s
  - Since the sync interval is 0.03125 s, this means that when a Sync message arrives, the filter has not had time for the response to the previous Sync message to complete
  - This results, in the filtered response, in reduction of the small peaks of the unfiltered response

# Case 2, Node 2 – Discussion - 3

- In slide 46, the magnitude of the time error slope between successive Sync messages increases in going from one rateRatio update to the next
  - This is due to the rate of increase in the frequency offset
  - For example, on slide 44 the measured frequency offset between 511 s and 512 s is approximately 92.4 ppm
    - But, during this period, the actual frequency offset is increasing at a rate of approximately 1.2 ppm/s (by considering the approximately slope of the “staircase” curve in slide 44
    - Over 1 s (between 511 s and 512 s), the frequency offset increases by 1.2 ppm = 1200 ns/s
      - This means that the change in phase over a 0.03125 s Sync interval in the vicinity of 511 s – 512 s, increases by (1200 ns/s)(0.03125 s) = 37.5 ns
      - In considering the increase in he size of the jumps on arrival of a Sync message in going from 511 s to 512 s
        - » The small jump after the frequency adjustment at 511 s is approximately -25 ns
        - » The small jump just before the frequency adjustment at 512 s is approximately -60 ns
        - » The increase in the magnitude of the jump is approximately 35 ns

# Case 2, Node 2 – Discussion - 4

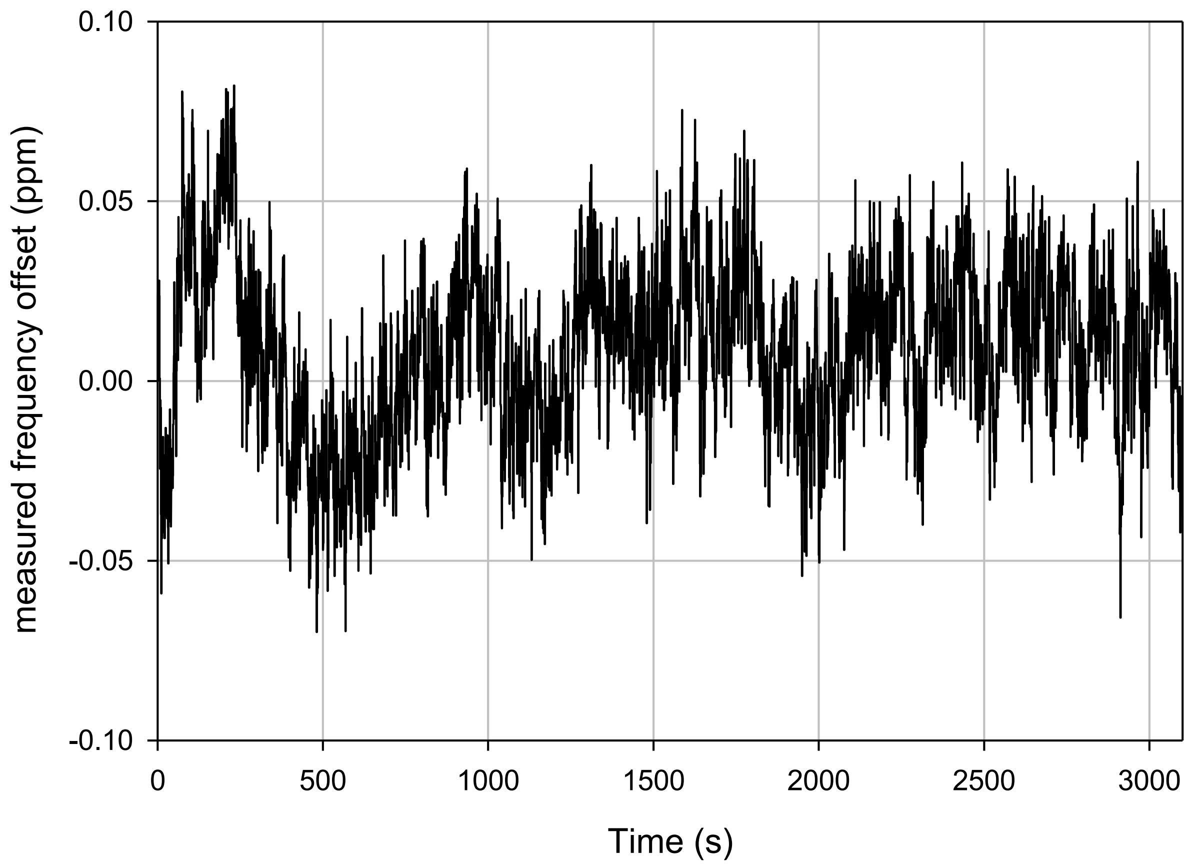
- Additional changes in magnitude of fast jumps (i.e., every 0.03125 s), in between successive frequency changes (i.e., every 1 s) are due to effect of 8 ns timestamp granularity

# Detailed Consideration of Case 1, Node 2

- This case uses clock model 1 (802.1AS-2020, Annex B requirement; FFM model)
- The following slides show measured frequency offset (using Pdelay message exchange), filtered time error history, and unfiltered time error history, for selected time intervals of the time history
- Note that while the following results are statistically the same as the previous case 1 (FFM) results, they are not identical (i.e., there is statistical variation) because the following results were produced using a network consisting of 2 nodes (since we are only interested in node 2 for this case)
    - This resulted in a different stream of random numbers used at node 2 to generate the FFM phase noise, which resulted in different FFM and time error samples
    - This was done to save both run time and disk storage (for the following cases, additional outputs were generated, e.g., measured frequency offset)

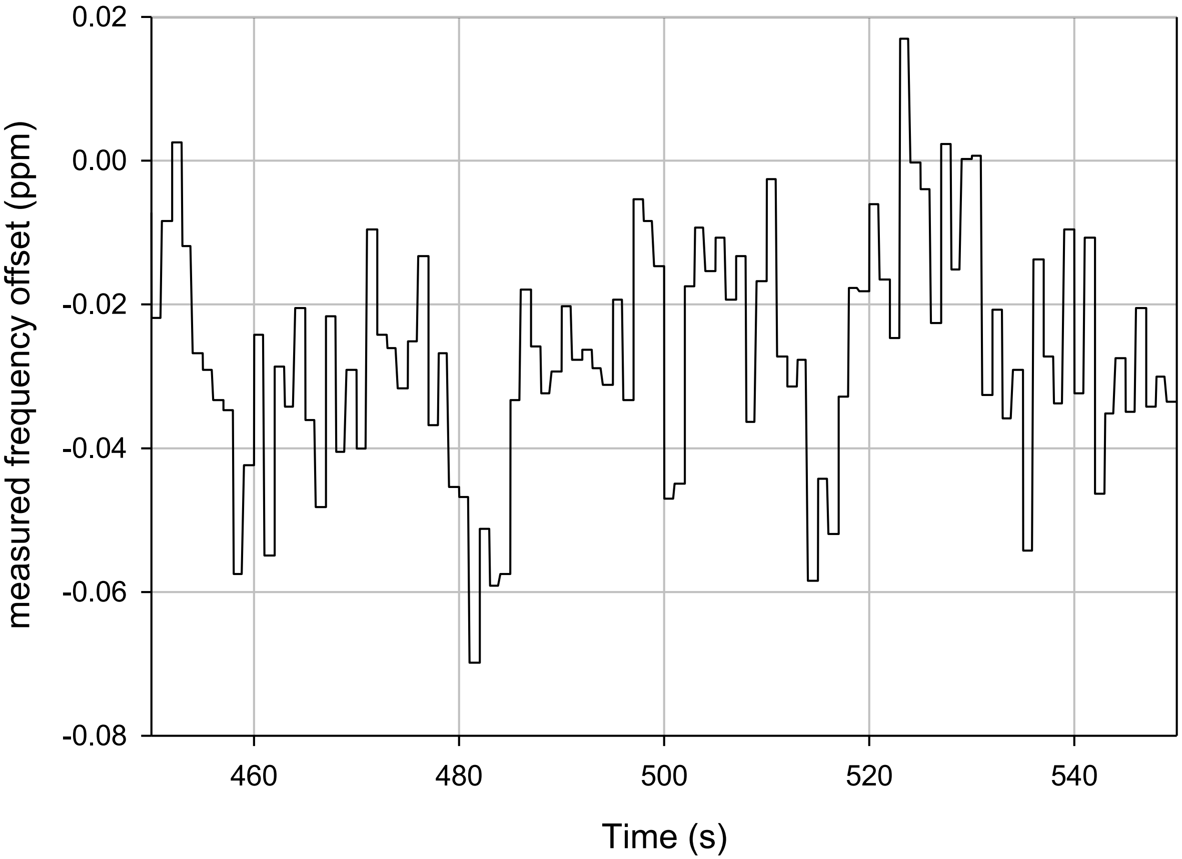
# Case 1, node 2 measured frequency offset

Case 1, PTP Instance (node) 2
Measured frequency offset of node 2 relative to node 1
Clock Model 1 (Flicker Frequency Modulation (802.1AS-2020 Annex B)



# Case 1, node 2 measured frequency offset, detail of 450 s - 550 s

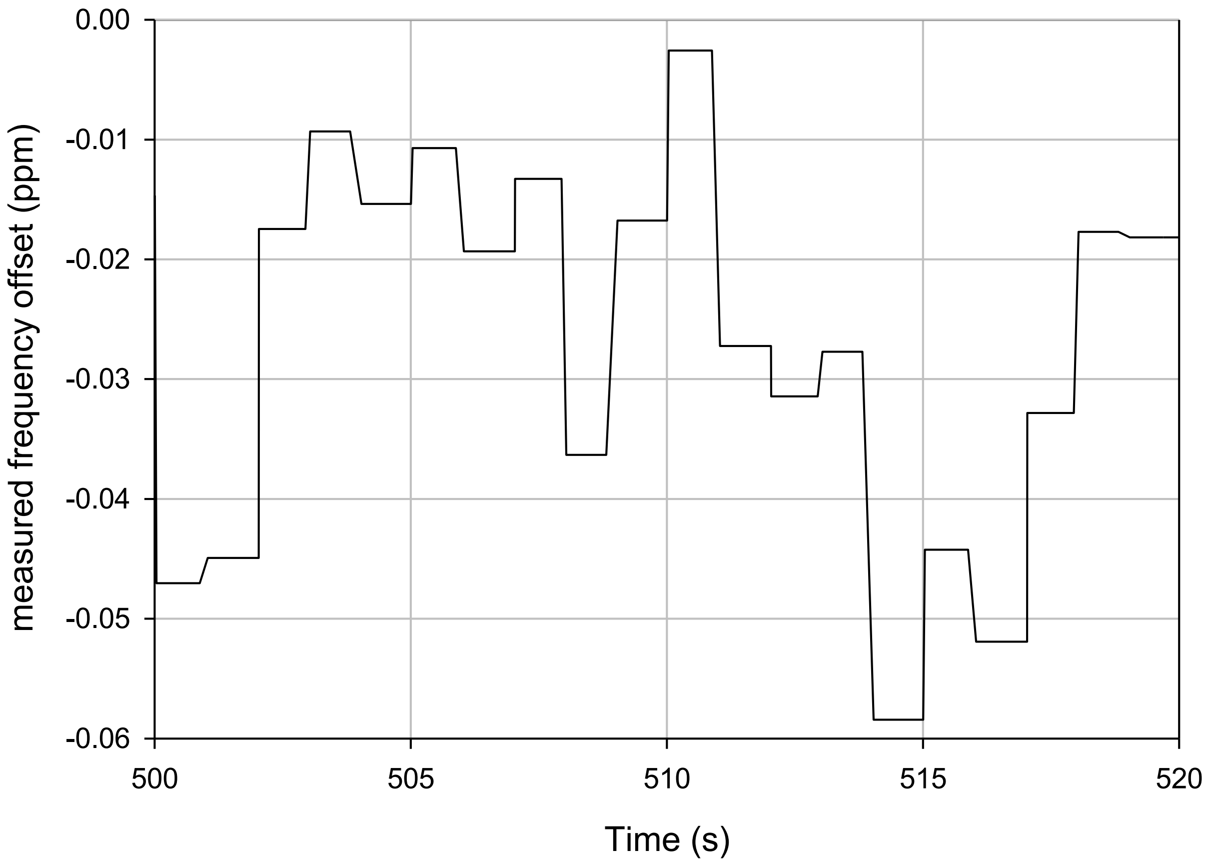
Case 1, PTP Instance (node) 2
Measured frequency offset of node 2 relative to node 1
Clock Model 1 (Flicker Frequency Modulation (802.1AS-2020 Annex B)
Detail of 450 s - 550 s




Vertical jumps are due to frequency offset being measured only once per second

# Case 1, node 2 measured frequency offset, detail of 500 s - 520 s

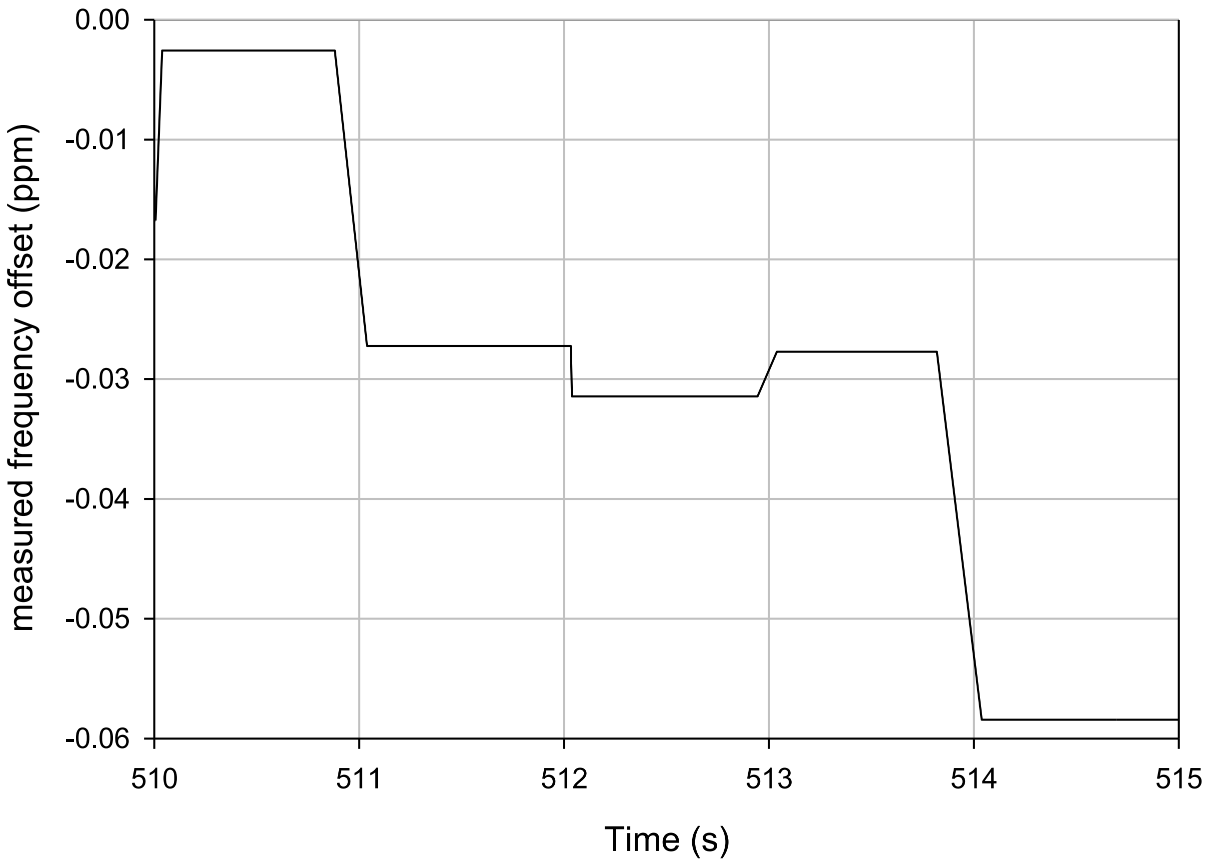
Case 1, PTP Instance (node) 2
Measured frequency offset of node 2 relative to node 1
Clock Model 1 (Flicker Frequency Modulation (802.1AS-2020 Annex B)
Detail of 500 s - 520 s




Vertical jumps are due to frequency offset being measured only once per second

# Case 1, node 2 measured frequency offset, detail of 510 s - 515 s

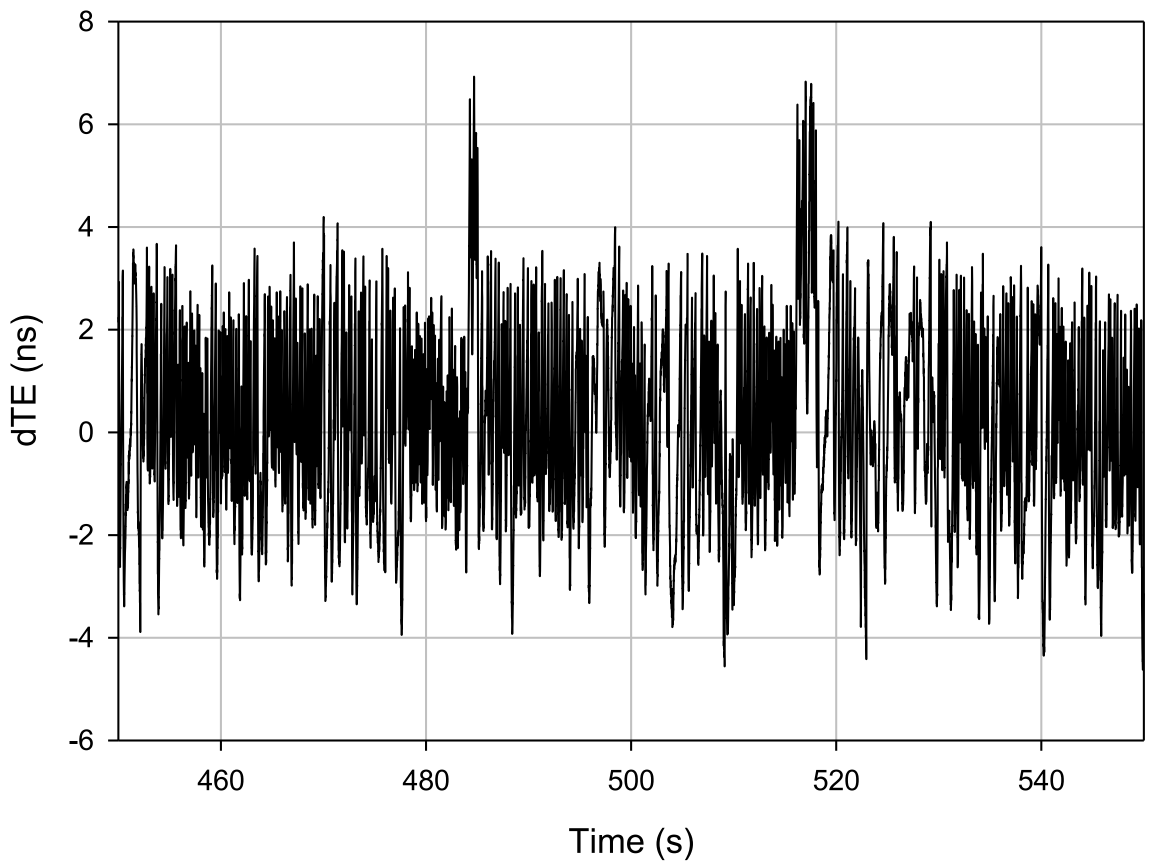
Case 1, PTP Instance (node) 2
Measured frequency offset of node 2 relative to node 1
Clock Model 1 (Flicker Frequency Modulation (802.1AS-2020 Annex B)
Detail of 510 s - 515 s




Vertical jumps are due to frequency offset being measured only once per second

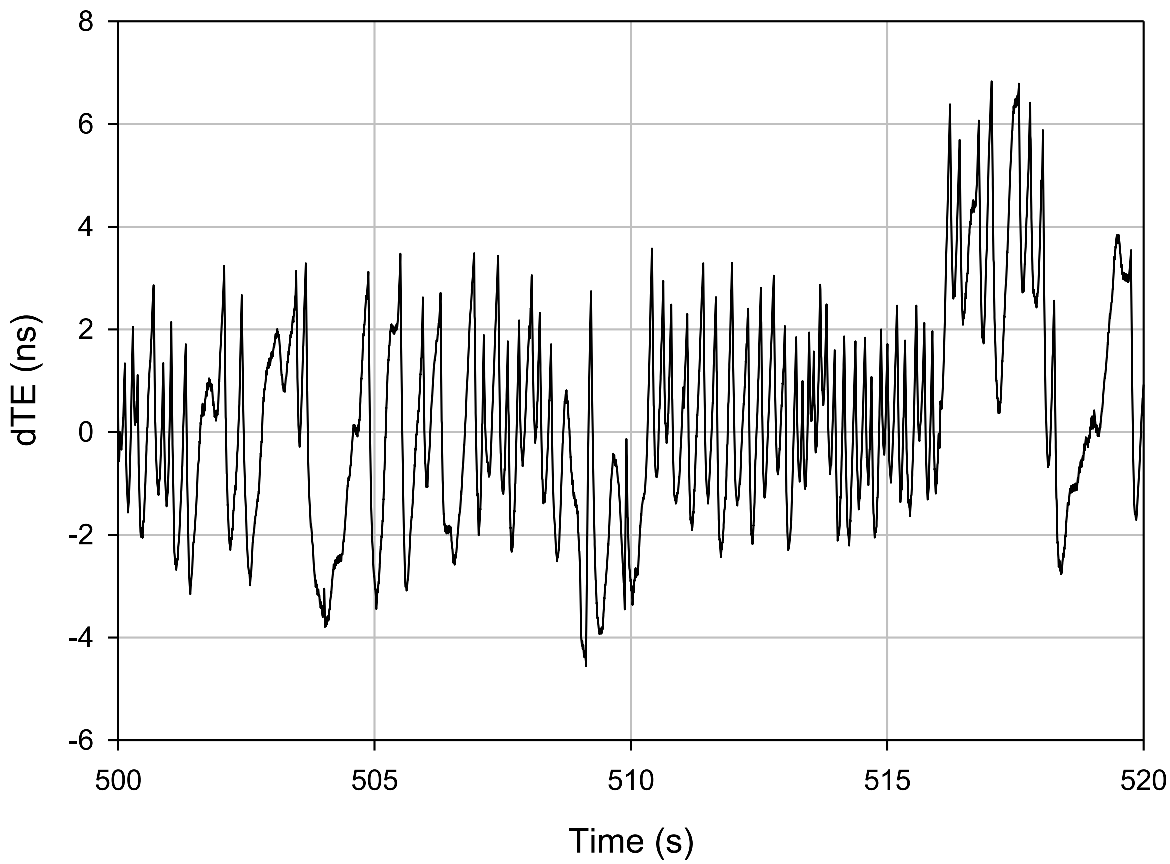
# Case 1, node 2 filtered time error - detail of 450 - 550 s

Case 1, PTP Instance (node) 2
Clock Model 1 (Flicker Frequency Modulation (802.1AS-2020 Annex B)
Detail of 450 s - 550 s

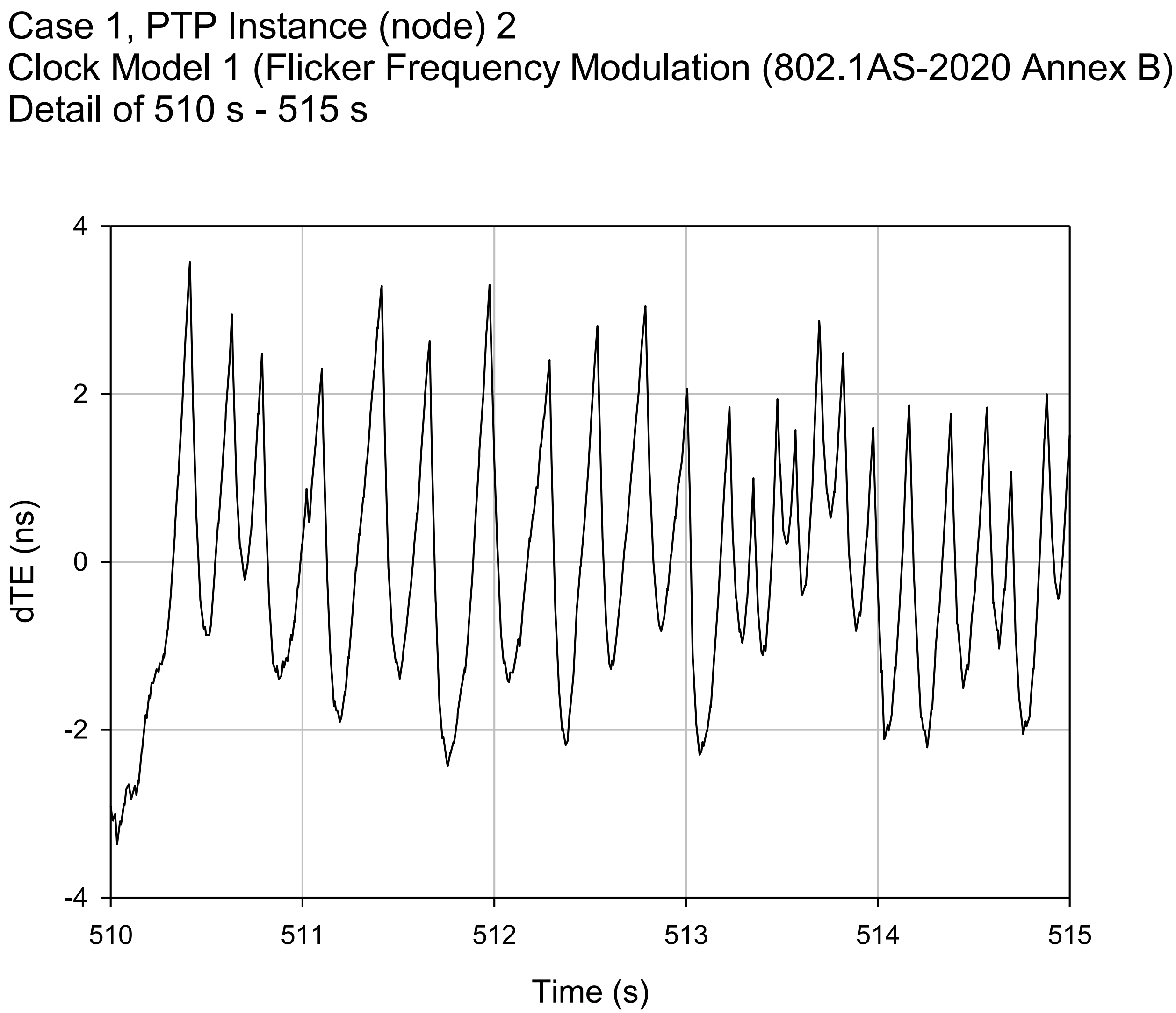


# Case 1, node 2 filtered time error - detail of 500 - 520 s

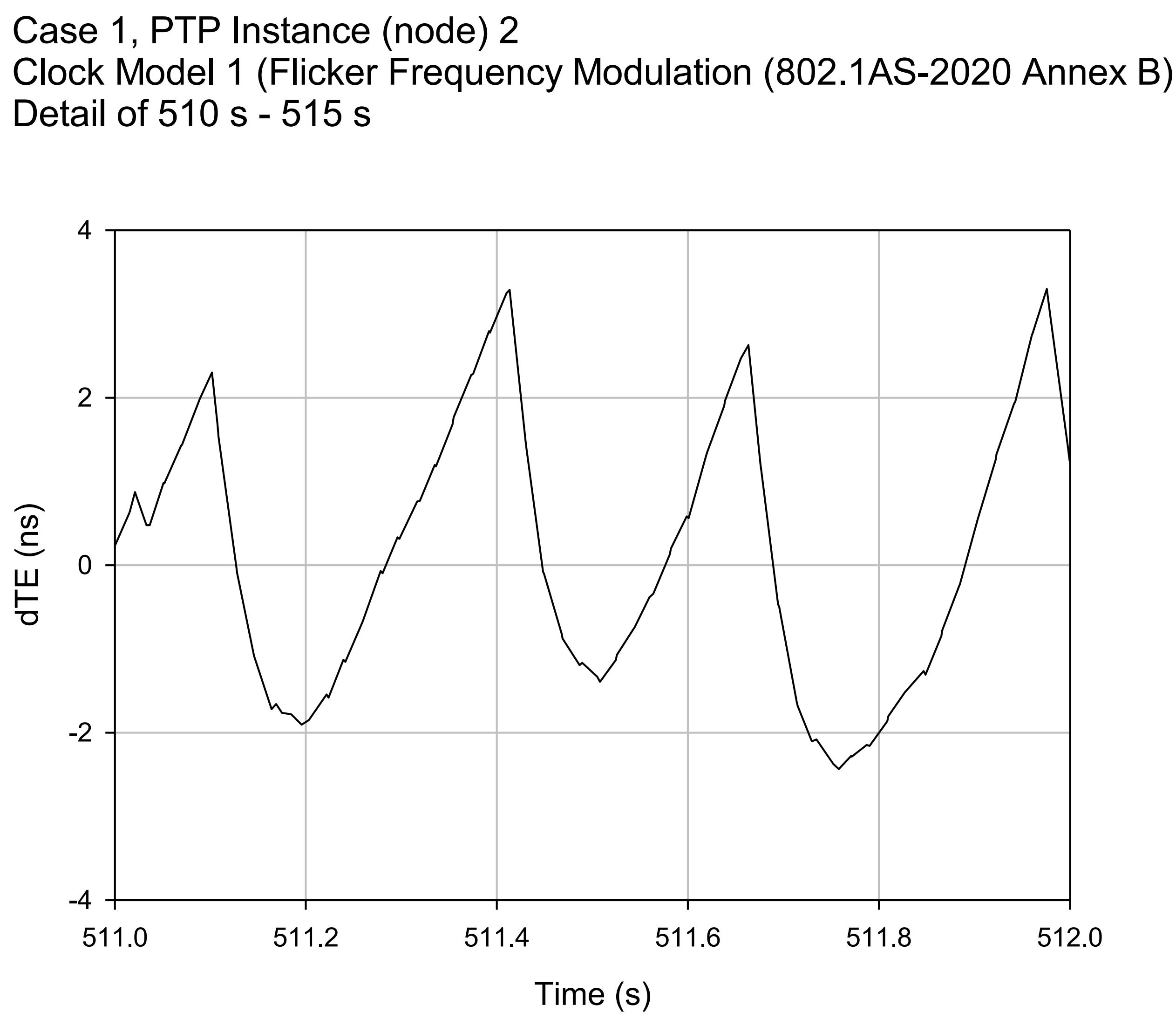
Case 1, PTP Instance (node) 2
Clock Model 1 (Flicker Frequency Modulation (802.1AS-2020 Annex B)
Detail of 500 s - 520 s



# Case 1, node 2 filtered time error - detail of 510 - 515 s

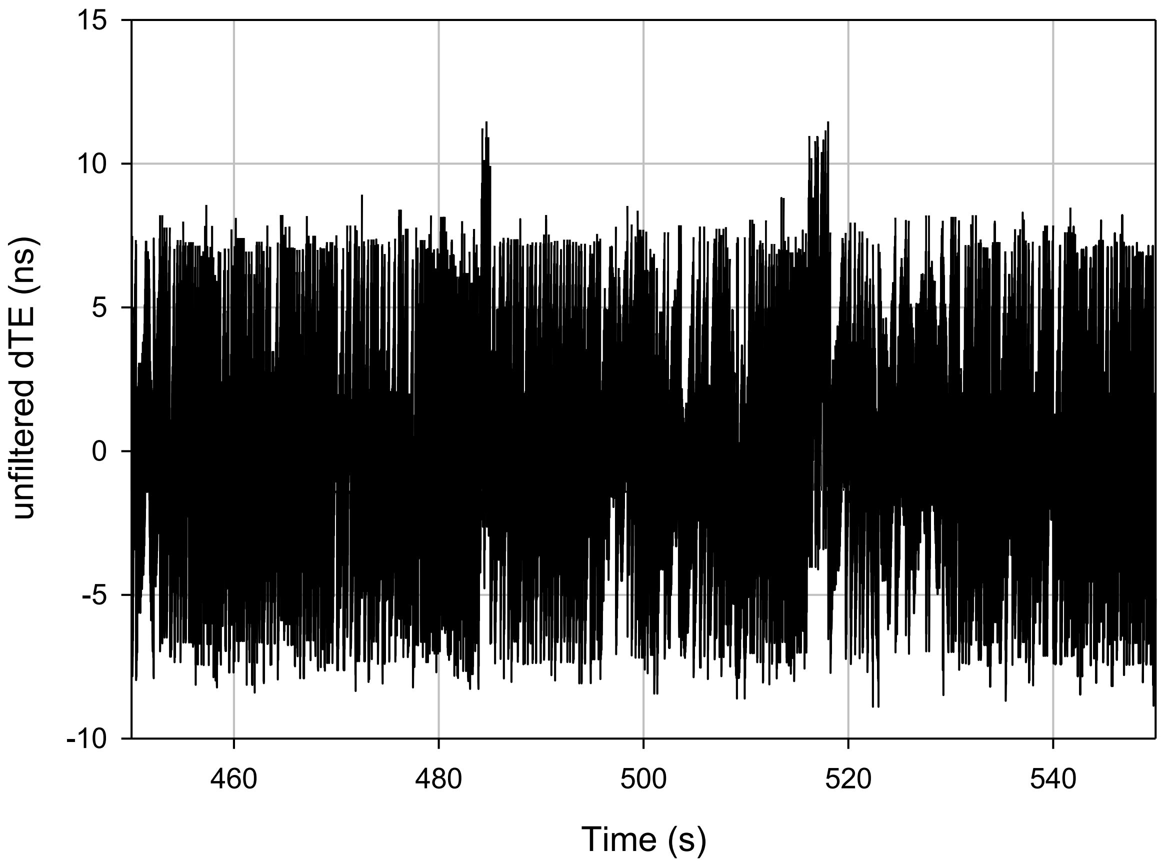


# Case 1, node 2 filtered time error - detail of 511 - 512 s



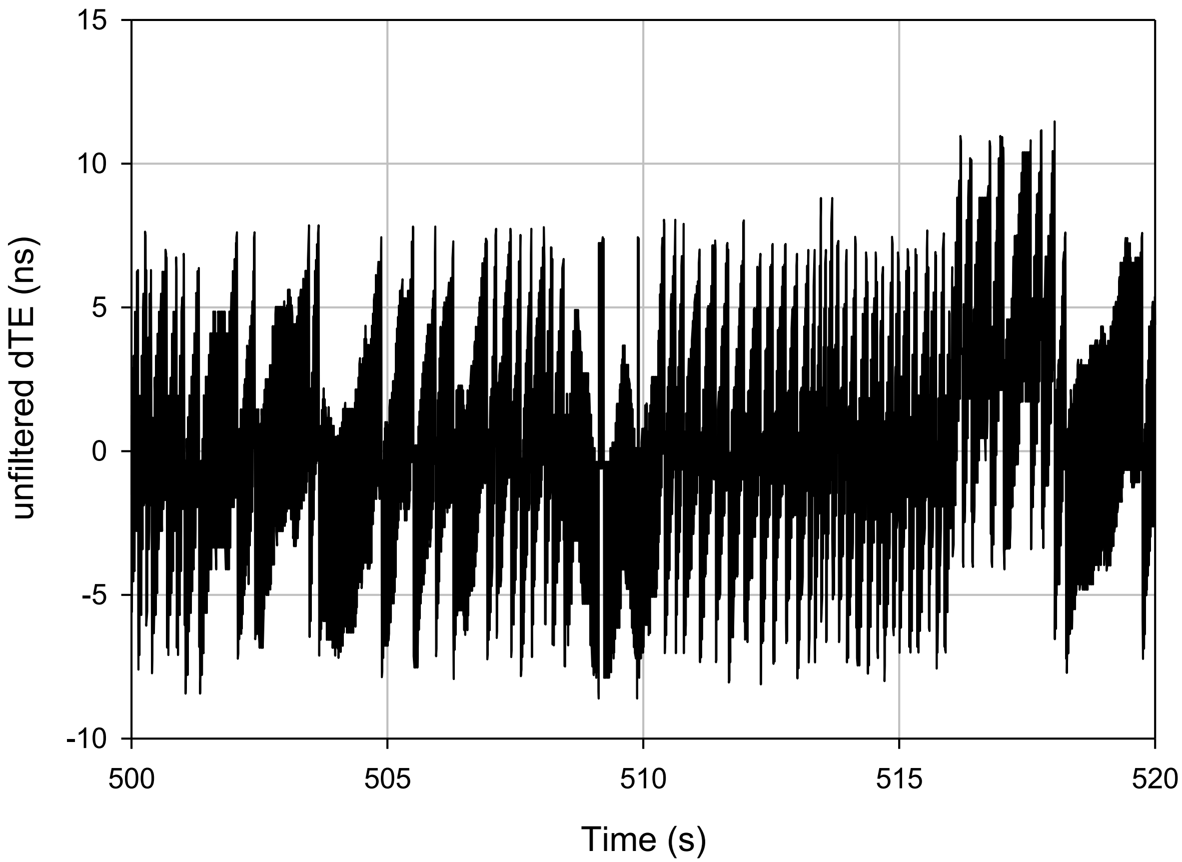
# Case 1, node 2 unfiltered time error - detail of 450 - 550 s

Case 1, PTP Instance (node) 2
Unfiltered time error
Clock Model 1 (Flicker Frequency Modulation (802.1AS-2020 Annex B)
Detail of 450 s - 550 s

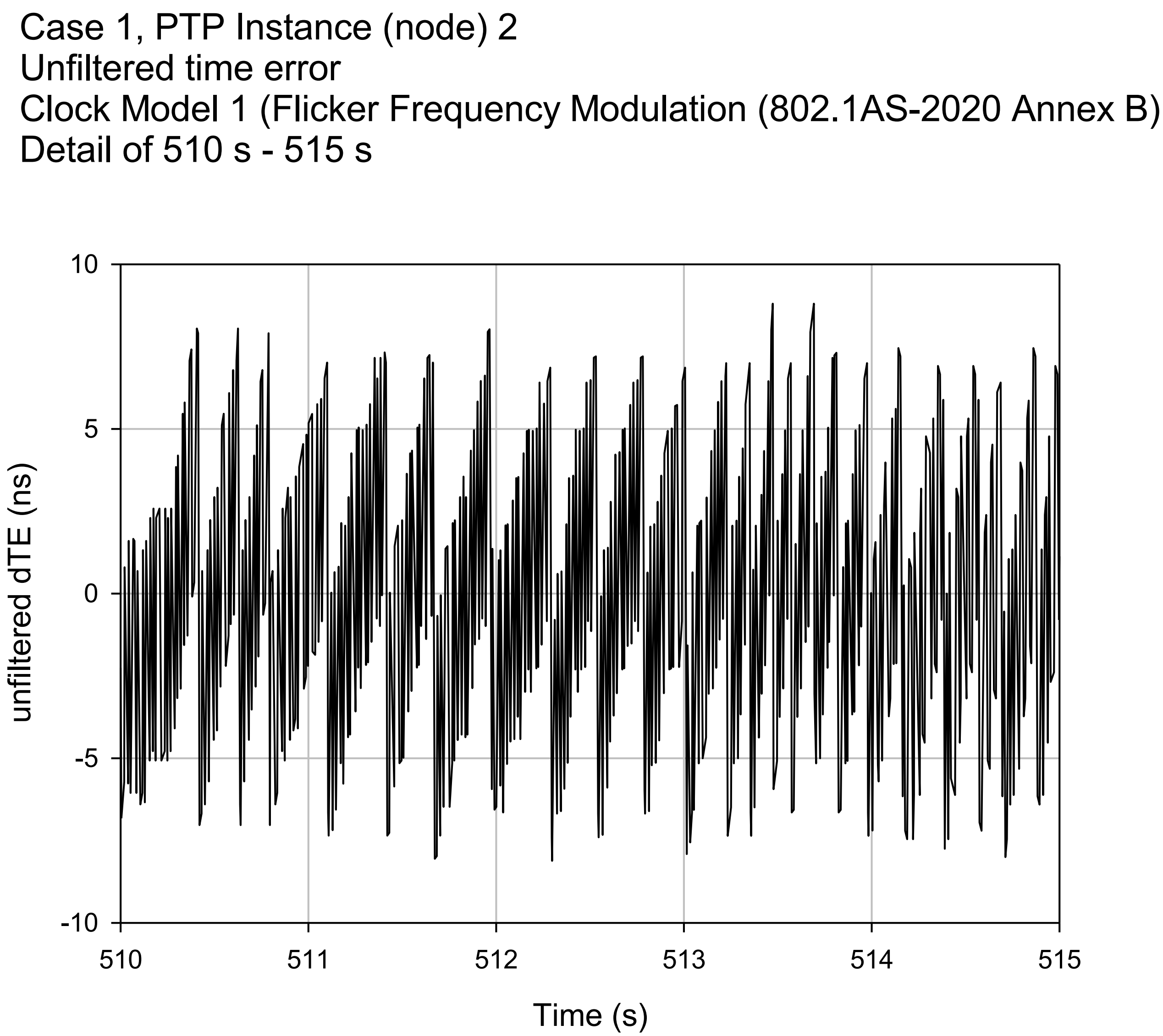


# Case 1, node 2 unfiltered time error - detail of 500 - 520 s

Case 1, PTP Instance (node) 2
Unfiltered time error
Clock Model 1 (Flicker Frequency Modulation (802.1AS-2020 Annex B)
Detail of 500 s - 520 s

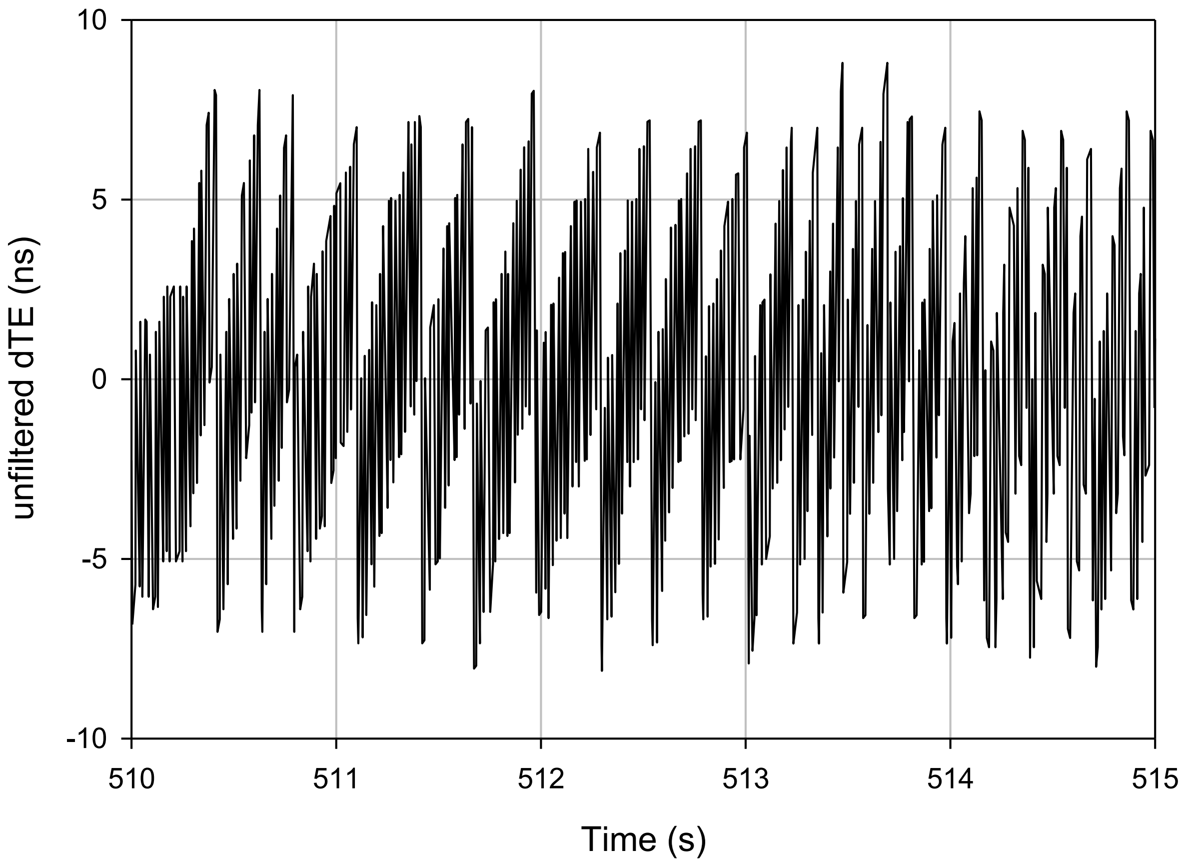


# Case 1, node 2 unfiltered time error - detail of 510 - 515 s



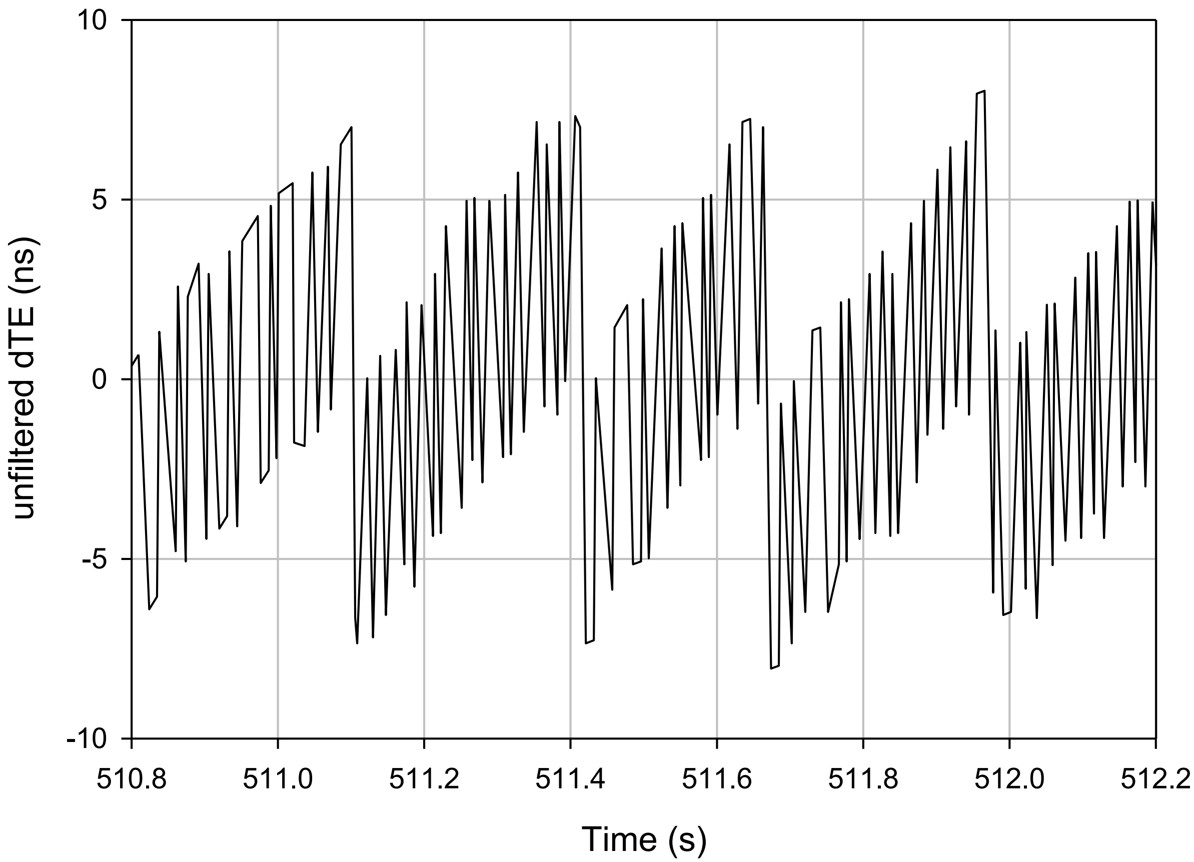
# Case 1, node 2 unfiltered time error - detail of 510 - 515 s

Case 1, PTP Instance (node) 2
Unfiltered time error
Clock Model 1 (Flicker Frequency Modulation (802.1AS-2020 Annex B)
Detail of 510 s - 515 s



# Case 1, node 2 unfiltered time error - detail of 510.8 - 511.2 s

Case 1, PTP Instance (node) 2
Unfiltered time error
Clock Model 1 (Flicker Frequency Modulation (802.1AS-2020 Annex B)
Detail of 510.8 s - 512.2 s



# Case 1 Discussion

- Fastest occurring jumps, most evident in slides 69 and 70, occur when successive Sync messages are received
- Less frequency jumps, occurring approximately every 0.3 s in slide 70, are due to 8 ns timestamp granularity
- Effect of frequency measurements every 1 s is less evident because frequency adjustments are small (see slides 60 and 61)
  - Frequency adjustments are on the order of several hundredths of a ppm

# Discussion of Results - 1

- The results for models 2 and 3 (slide 29) show an accumulated max|dTE| over 100 PTP Instances (nodes) that ranges from approximately 2.5 μs to 6.2 μs, depending on the exact assumptions
  - The best case (2.5 μs) is for model 2 (sinusoidal local clock frequency variation) and relative phases of the local clock frequency variation random
  - The worst case (6.2 μs) is the same for both sinusoidal and triangular wave frequency variation, under the assumption that the phases of the local clock frequency variation are the same (i.e., relative phases of zero)
- **This means that, with the current assumptions (slides 15-17), the desired 1 μs for total TE (i.e., max|TE|) cannot be met, it exceeded by a factor of 2.5 – 6.2 just for the dTE component**

# Discussion of Results - 2

- In addition, while it is unlikely that all 100 local clocks would have frequency variation with exactly the same phase, note that in reality the frequencies of the frequency variation (i.e., the modulation frequencies) also would not be identical; in reality, there would be a "beating" effect and over time there would be periods when the phase/time error peaks at the different nodes would line up
  - In any case, even with the assumption of random phases, the desired 1 μs is exceeded by at least a factor of 2.5
- The results for model 1 (slide 28) show an accumulated max|dTE| over 100 PTP Instances (nodes) of approximately 135 ns
  - This is well within the desired max|TE| of 1 μs
  - **Therefore, it appears it is possible to meet the desired 1 μs max|TE| if the local clock stability meets the requirement of 802.1AS-2020, Annex B, though this depends on other budget components of max|TE| (e.g., cTE)**

# Discussion of Results - 3

- The accumulated max|dTE| performance is better for local clock model 1 compared to models 2 and 3 by more than a factor of 10
  - The reason the performance is so much better for model 1 compared to models 2 and 3 is that the stability of clock model 1 is much better than for models 2 and 3
  - In slide 25, TDEV for model 1 is more than a factor of 10 below TDEV for models 2 and 3
    - Related to this, the actual frequency adjustments are on the order of a few hundredths of a ppm for model 1, versus 1 ppm for models 2 and 3

# Conclusion

- The current IEC/IEEE 60802 draft specifies maximum frequency offset and frequency drift rate of 100 ppm and 3 ppm/s, corresponding to models 2 and 3
  - However, the draft also describes an HRM of 100 PTP Instances (nodes) and max|TE| of 1 $\mu$s
  - The results show that these objectives, along with the other assumptions given here, are not compatible
- On the following slides, possible approaches towards meeting the above objectives are discussed

# Possible Approaches Toward Meeting Objectives - 1

- Note: some or all of the approaches listed below might be used
- Use a more stable oscillator for the local clock, e.g., with TDEV on the order of the 802.1AS Annex B requirement (red curve on slide 25)
  - Would need to consider relative cost (both relative to model 2 and 3 oscillator, but also relative to the entire network element)
- Increase the Pdelay rate, so that the neighborRateRatio updates are more frequent
  - This would result in smaller frequency error and smaller frequency adjustments
  - But note that if we do this, the error due to timestamp granularity and taking a backward difference over the Pdelay interval would increase; at some point, this would exceed the effect of making the frequency measurements more frequently
    - Initial suggestion: increase Pdelay rate to 32 messages/s

# Possible Approaches Toward Meeting Objectives - 2

- Use an endpoint filter with narrower bandwidth an possibly smaller gain peaking
  - This would more effectively filter the dTE peaks (see slides 45-52)
  - However, need to consider relative cost of this, as narrower bandwidth endpoint filter will require a more stable oscillator (VCO or DCO) in order to not have larger noise generation
    - As for local clock oscillator, would need to consider relative cost both relative to model 2 and 3 oscillator, but also relative to the entire network element
  - Initial suggestion: Use 0.1 Hz for 3 dB bandwidth and 0.1 dB gain peaking

Thank you

# References

[1] Geoffrey M. Garner, *Discussion of Assumptions Needed for 60802 Network Simulations*, IEEE 802.1 presentation, January 2020.

[2] Geoffrey M. Garner, *Initial Simulation Results for 802.1AS Synchronization Transport with Longer Sync and Pdelay Intervals*, IEEE 802.1 presentation, May 18, 2009.

[3] Geoffrey M. Garner, *Simulation Results for 802.1AS Synchronization Transport with Clock Wander Generation and Updated Residence and Pdelay Turnaround Times*, IEEE 802.1 presentation, July 12, 2010.

[4] Geoffrey M. Garner, *Comparison of 802.1AS Annex B and 60802 Clock Stability*, IEEE 802.1 presentation, January 2020.

[5] ITU-T Rec. G.8251, The Control of Jitter and Wander within the Optical Transport Network (OTN), ITU-T, Geneva, November, 2018.

# Appendix – Flicker Noise Generation Model

For IEEE Std. 802.1AS-2020, Appendix B Clock No
ise Generation Requirement

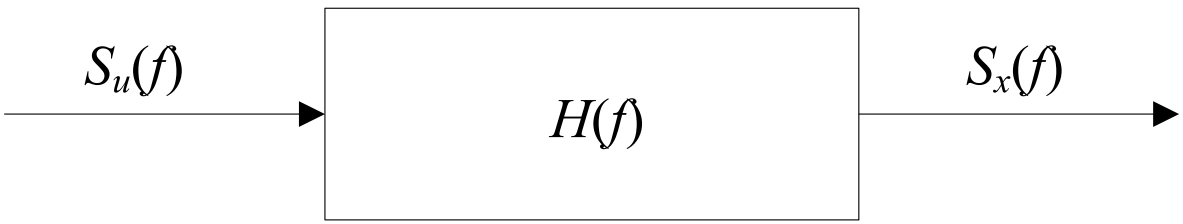
Adapted from Reference [3] of the main references

# Local Clock Wander Generation Model - 8

## ❑Simulation of FPM

- FPM is simulated by passing a sequence of independent, identically distributed random samples through a Barnes/Jarvis filter [1] – [3]
  - If white noise is input to a filter with frequency response $H(f) = f^{-1/2}$, the output is a random process with PSD proportional to $1/f$
  - The Barnes/Jarvis filter approximates an $f^{-1/2}$ frequency response using a bank of lead/lag filters
    - The actual frequency response of this filter is a “staircase”
    - The spacings of the poles and zeros are chosen such that the average slope is –10 dB/decade
- Noise distribution is taken as Gaussian with zero mean
- Variance determines TDEV level
  - Choose variance such that the computed TDEV from a sample history is close to value obtained from above relation between TDEV and PSD

# Local Clock Wander Generation Model - 9




$$S_x(f) = \left|H(f)\right|^2 S_u(f)$$

If $S_u(f) = K = \text{constant}$ (WPM) and $H(f) = 1/\sqrt{f}$ , then

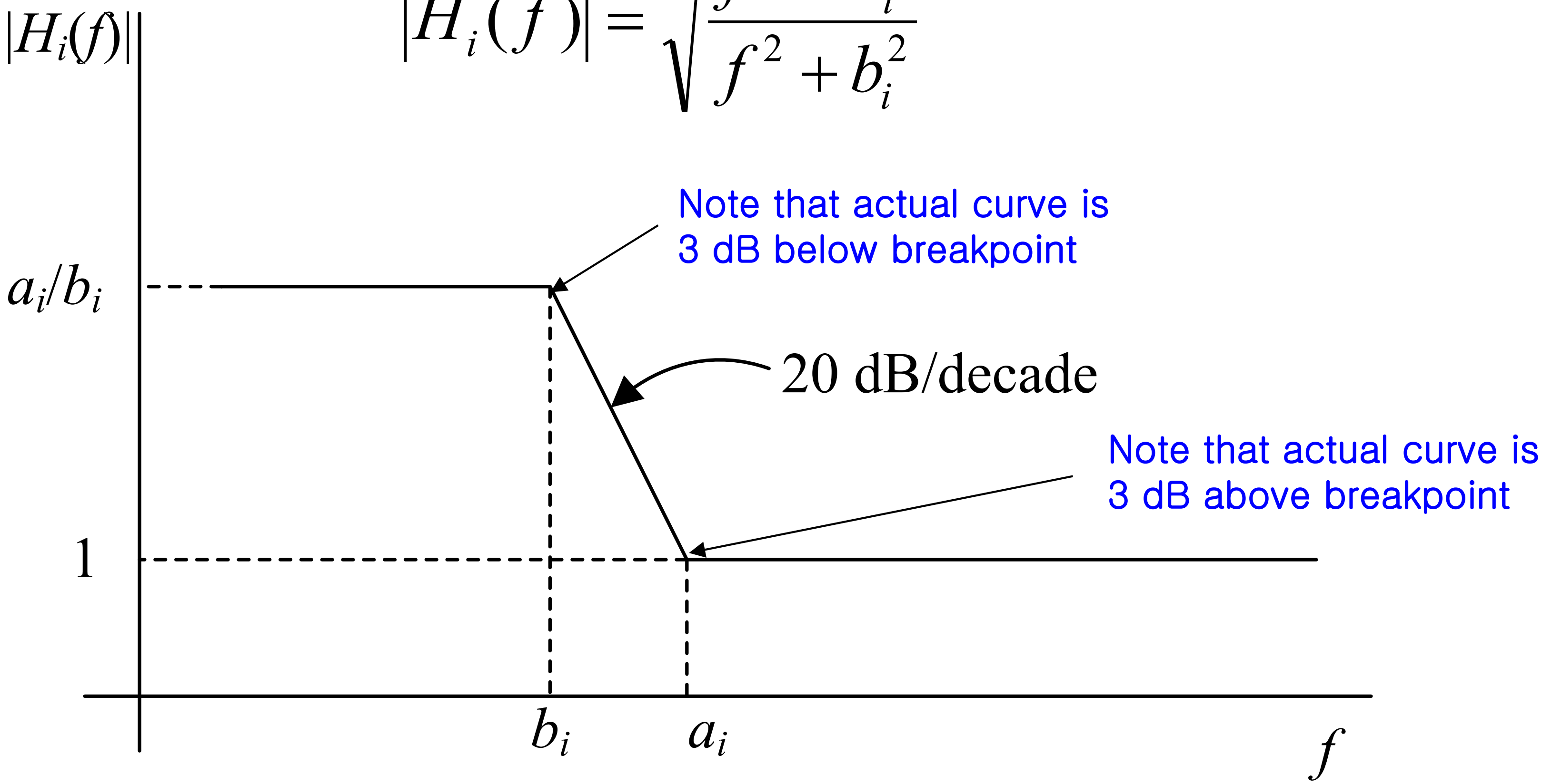
$$S_x(f) = K / f \text{ (FPM)}$$

# Local Clock Wander Generation Model - 10

- Consider

$$H_i(f) = \frac{jf + a_i}{jf + b_i}, \qquad a_i > b_i$$

$$|H_i(f)| = \sqrt{\frac{f^2 + a_i^2}{f^2 + b_i^2}}$$

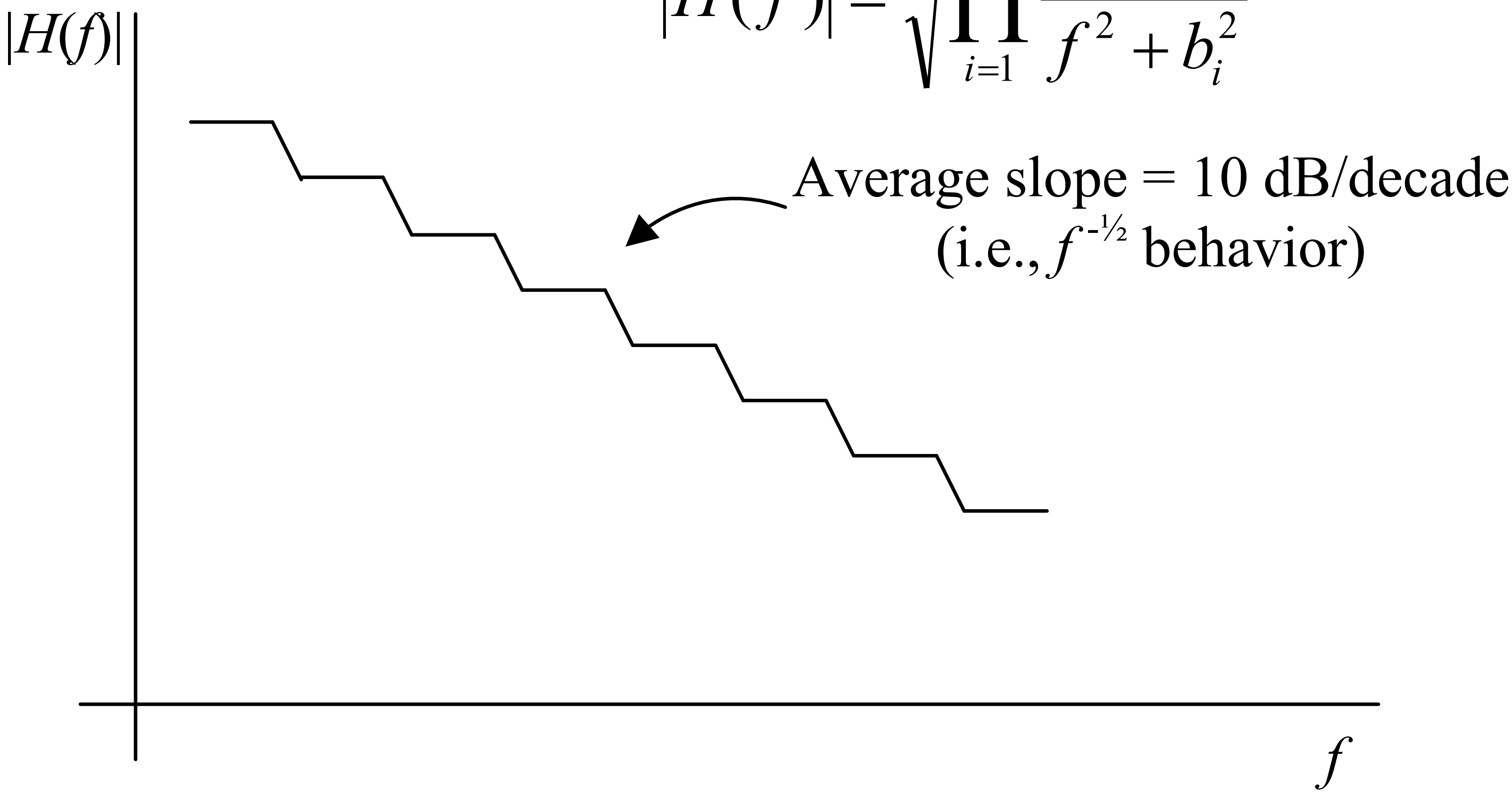


# Local Clock Wander Generation Model - 11

- Next, consider

$$H(f) = \prod_{i=1}^{N} \frac{jf + a_i}{jf + b_i}, \qquad a_i > b_i$$

$$|H(f)| = \sqrt{\prod_{i=1}^{N} \frac{f^2 + a_i^2}{f^2 + b_i^2}}$$

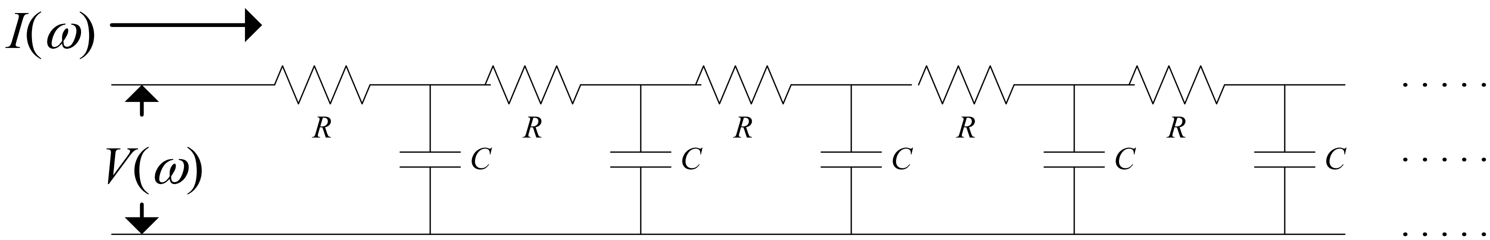


# Local Clock Wander Generation Model - 12

- A discrete-time implementation of the filter bank is given by Barnes and Greenhall in [2]
  - In the implementation here, 8 stages are used, to simulate FPM (and integrate to obtain FFM, see below) over approximately 7 decades
- Simulation of FFM
  - Input a sequence of independent, identically distributed random samples through a Barnes/Jarvis filter followed by an integrator (accumulator)
  - Noise distribution is taken as Gaussian with zero mean
  - Variance determines TDEV level
    - Choose variance such that the computed TDEV from a sample history is close to value obtained from above relation between TDEV and PSD

# Local Clock Wander Generation Model - 13

- Note: It can be shown (see [4]) that the impedance of an $RC$ network approaches a $1/\omega^{1/2}$ dependence in the limit as the extent of the network (in one direction) becomes infinite, $R\to 0$, $C\to 0$, $R/C \to K$ ($K$ is a constant)




$$Z(\omega) = \frac{V(\omega)}{I(\omega)} = \frac{R}{2} + \left(\frac{R^2}{4} + \frac{R}{j\omega C}\right)^{1/2} \to \sqrt{\frac{R}{C}}\frac{1}{(j\omega)^{1/2}}$$

as $R \to 0, C \to 0, R/C \to K$

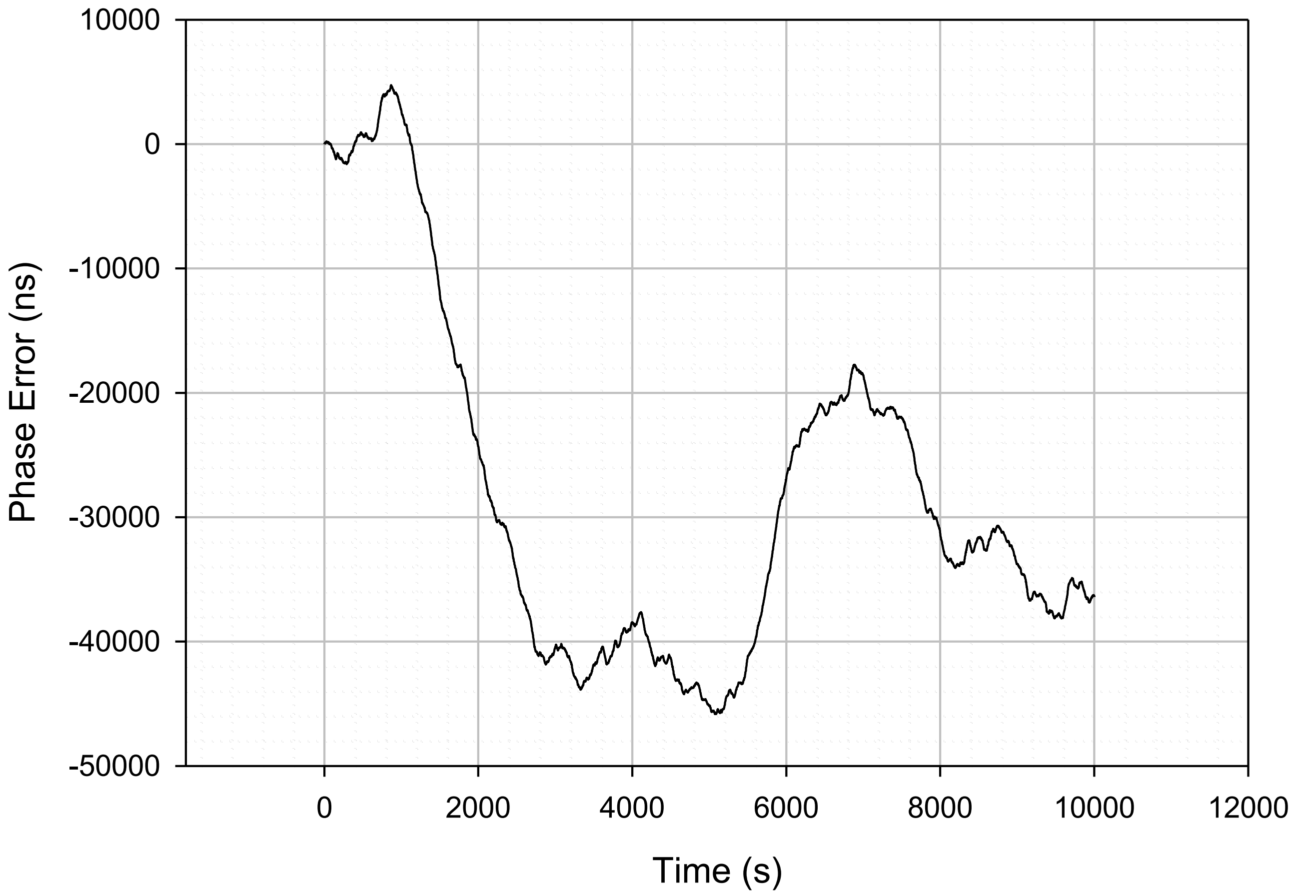
$$|Z(\omega)| \to \sqrt{\frac{R}{C}} \cdot \frac{1}{\omega^{1/2}}$$

$$|Z(\omega)|^2 \to \frac{R}{C} \cdot \frac{1}{\omega} \quad \text{(i.e., has a } 1/\omega \text{ dependence)}$$

# Local Clock Wander Generation Model - 14

## ❑Sample local clock phase noise history

Sample phase error history corresponding to 802.1AS Annex B, Figure B-1 TDEV mask

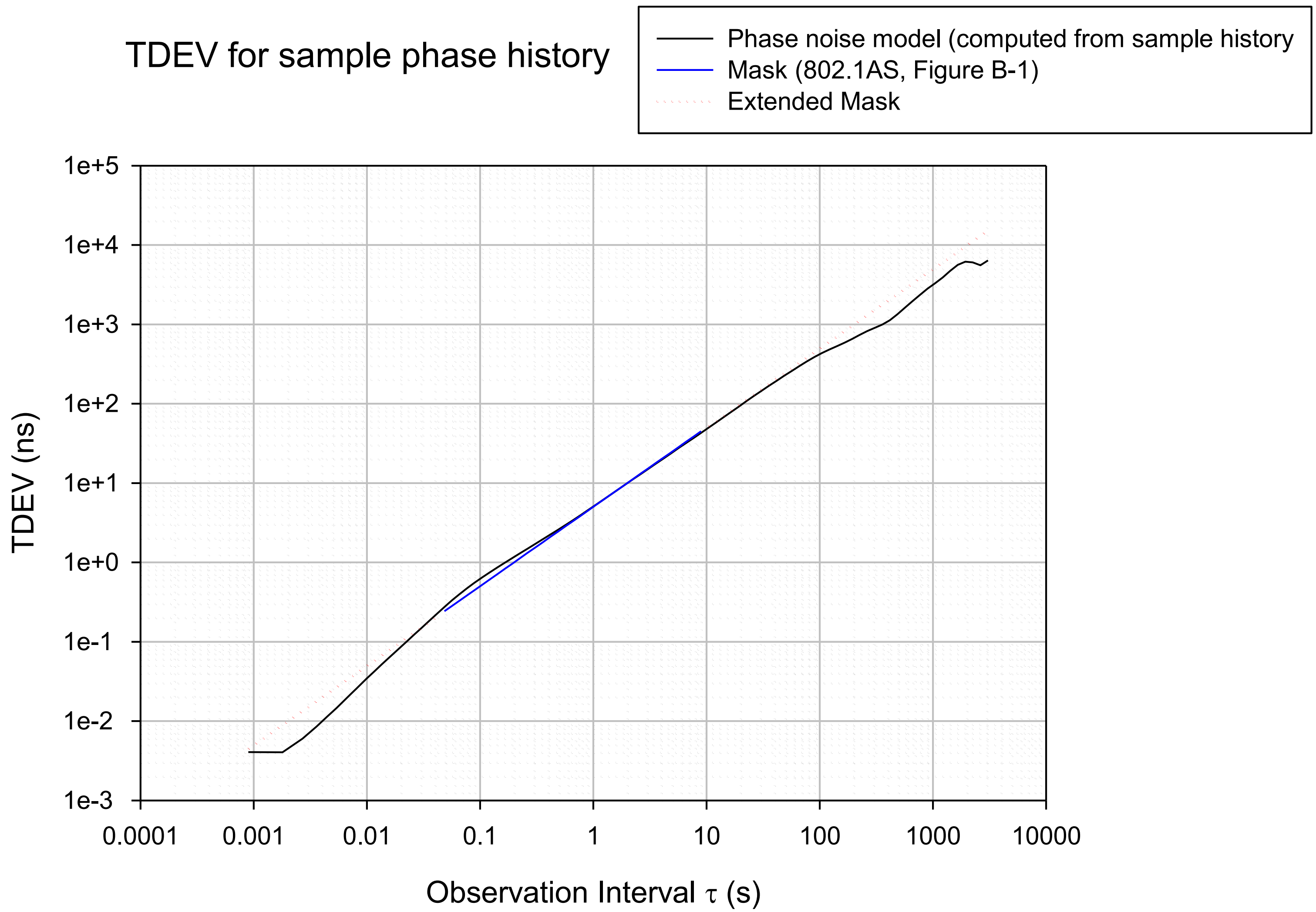


# Local Clock Wander Generation Model - 15

## ❑Sample local clock phase noise history (detail of 1199 – 1209 s)

# Local Clock Wander Generation Model - 16

- TDEV for phase noise sample history, and comparison with 802.1AS Figure B-1 mask



# References for Appendix


1. J.A. Barnes and Stephen Jarvis, Jr., *Efficient Numerical and Analog Modeling of Flicker Noise Processes*, National Bureau of Standards, NBS Technical Note 604, June, 1971.
2. James A. Barnes and Charles A. Greenhall, *Large Sample Simulation of Flicker Noise*, 19th Annual Precise Time and Time Interval (PTTI) Applications Planning Meeting, December, 1987.
3. Giovanni Corsini and Roberto Saletti, *A $1/f^{\gamma}$ Power Spectrum Noise Sequence Generator*, IEEE Transactions on Instrumentation and Measurement, Vol. 37, No. 4, December, 1988, pp. 615 – 619.
4. Igor M. Sokolov, Joseph Klafter, and Alexander Blumen, *Fractional Kinetics*, Physics Today, Vol. 55, No. 11, November, 2002, pp. 48 – 54.