Additional Simulation Results for ResE Synchronization Using Filtered Phase and Instantaneous Frequency Adjustments

Geoffrey M. Garner (Consultant)
Gmgarner@comcast.net

SAMSUNG Electronics

IEEE 802.3 RESG 2005 San Jose
2005.11.14

Outline

- Introduction
- Review of end-to-end application jitter/wander requirements
- Review of synchronization approaches and simulation model
- Assumptions common to all simulation cases
- Summary of simulation cases
- Results
- Conclusions
- Future work
- References
- Appendix I – Clock noise model (includes definition of TDEV)
- Appendix II – Definition of MTIE
- Appendix III – TDEV results for simulation cases

Introduction

- In a previous presentation [13], a number of approaches for distributing synchronization over ResE were summarized
  - The approaches are all based on time stamp principles used in IEEE 1588
  - The approaches differ in how the time stamp information is used, as well as in the details of the time stamp exchange process (e.g., use of one-way and two-way versus only two-way exchanges, etc.)
- In previous work [1], [2], a simulation model for several of the approaches was described and initial results were presented
  - Looked at approaches using only phase updates and using both phase and frequency updates, with and without filtering of the phase updates (approaches [2] – [4] on slide 8)
    - Frequency updates, when performed, were instantaneous
  - Neglected time-stamp measurement error
  - Neglected granularity of the frequency adjustment
  - Assumed unrealistically small phase measurement granularity (1 ns)

Introduction

- The present work continues the work of [1] and [2] by looking at
  - More realistic phase measurement granularity
  - wider range of synch intervals and frequency adjustment intervals
  - Frequency adjustment granularity
  - Different filter bandwidth (in one case)
- Error in the time stamp measurement is neglected
  - This will be considered in a future presentation

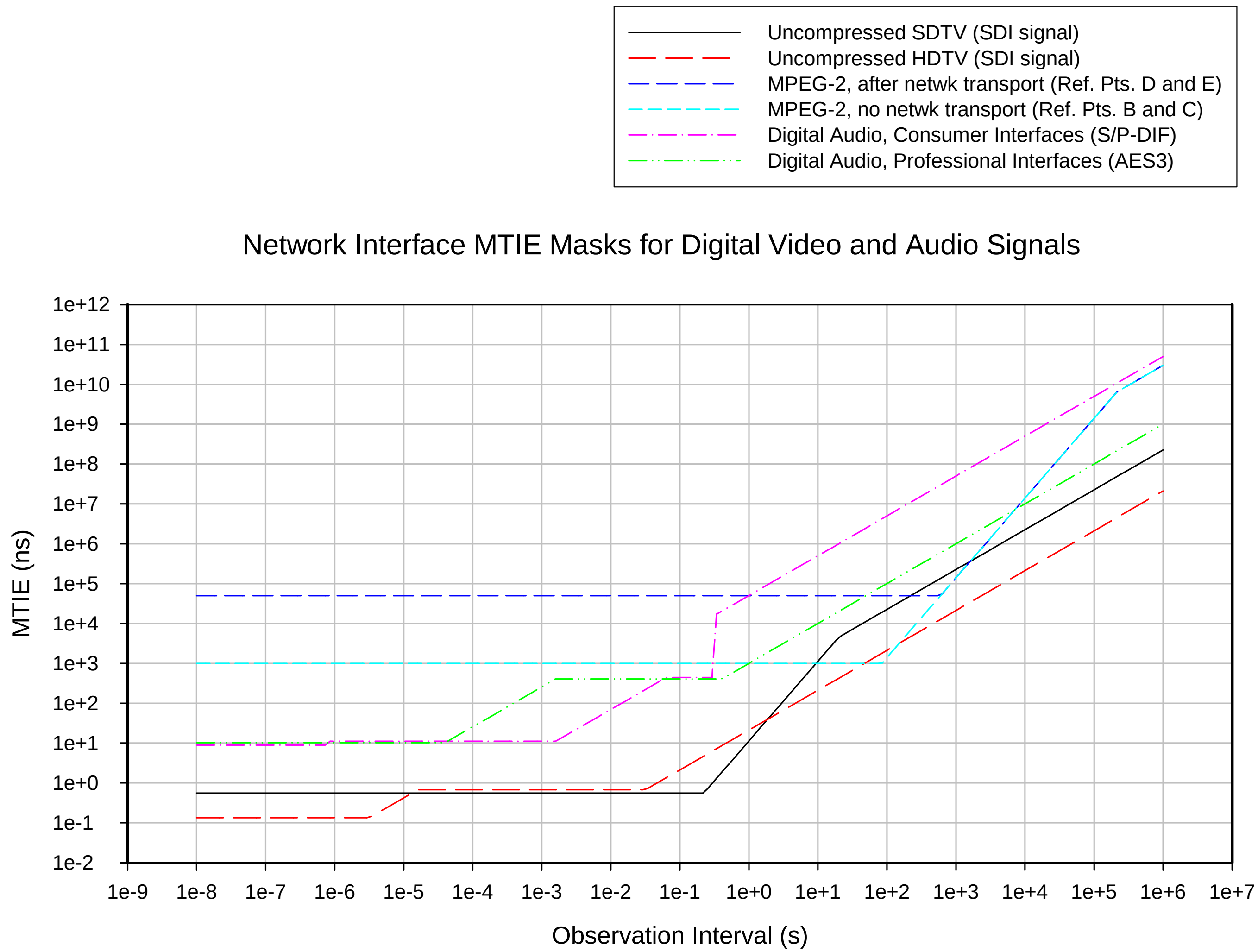
End-to-End Requirements - Review

Summary of End-to-End Application Jitter and Wander Requirements
(see[3] and references given there)

Requirement	Uncompressed SDTV	Uncompressed HDTV	MPEG-2, with network transport	MPEG-2, no network transport	Digital audio, consumer interface	Digital audio, professional interface
Wide-band jitter (UIpp)	0.2	1.0	50 μs peak-to-peak phase variation requirement (no measurement filter specified)	1000 ns peak-to-peak phase variation requirement (no measurement filter specified)	0.25	0.25
Wide-band jitter meas filt (Hz)	10	10			200	8000
High-band jitter (UIpp)	0.2	0.2			0.2	No requirement
High-band jitter meas filt (kHz)	1	100			400 (approx)	No requirement
Frequency offset (ppm)	±2.79365 (NTSC) ±0.225549 (PAL)	±10	±30	±30	±50 (Level 1) ±1000 (Level 2)	±1 (Grade 1) ±10 (Grade 2)
Frequency drift rate (ppm/s)	0.027937 (NTSC) 0.0225549 (PAL)	No requirement	0.000278	0.000278	No requirement	No requirement

End-to-End Requirements - Review

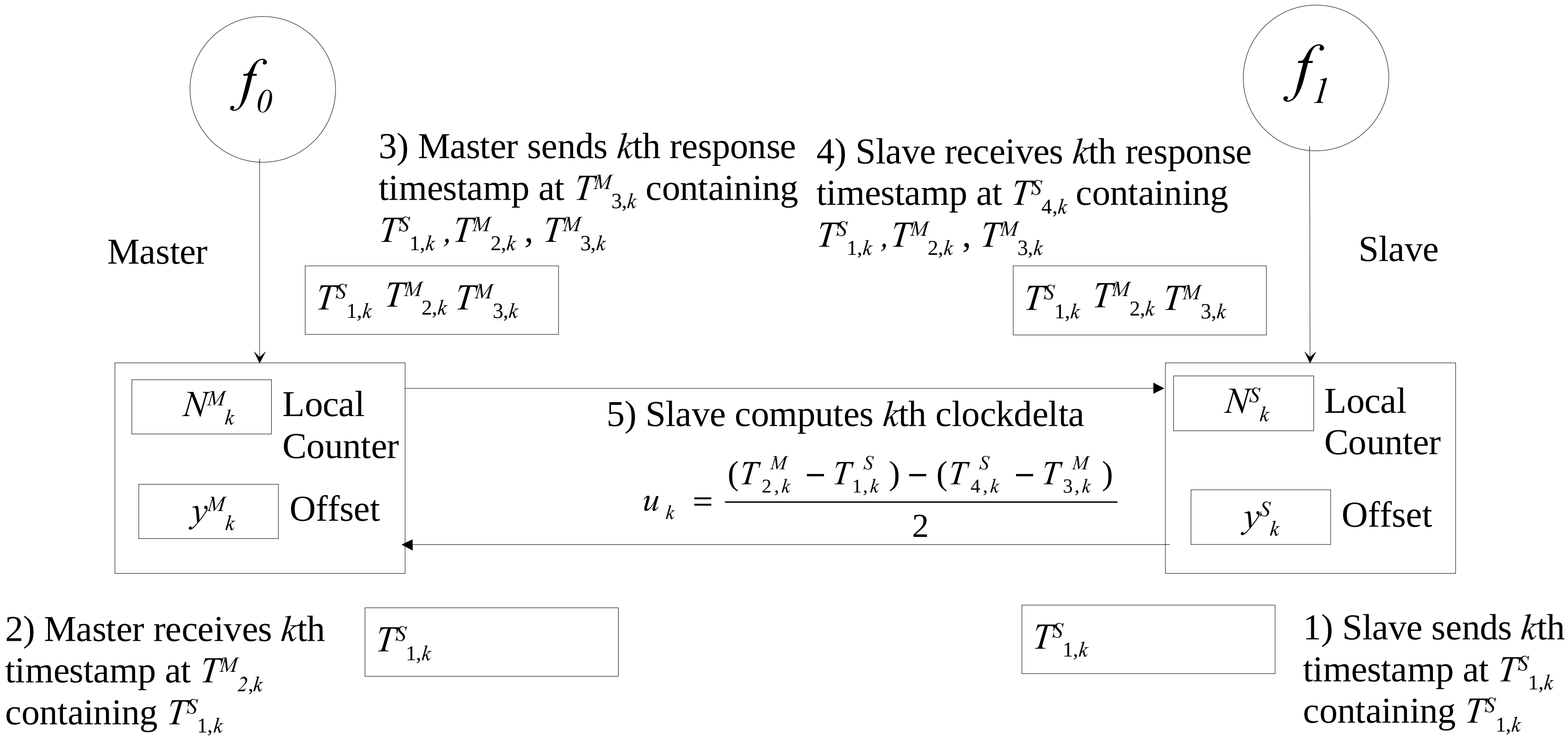
End-to-End Application Jitter and Wander Requirements
Expressed as MTIE Masks [3] (see Appendix II for MTIE definition)



Synchronization Approaches - Review

Basic 2-Way Time Stamp Approach used in IEEE 1588

- ResE will use this basic approach; however, a number of variations are possible
- Generally assumed a filtering function will be present at the endpoint
  - May be present at intermediate nodes (i.e., in some variations)



6) Slave computes current offset y_k in terms of current and possibly past clockdelta's u_k

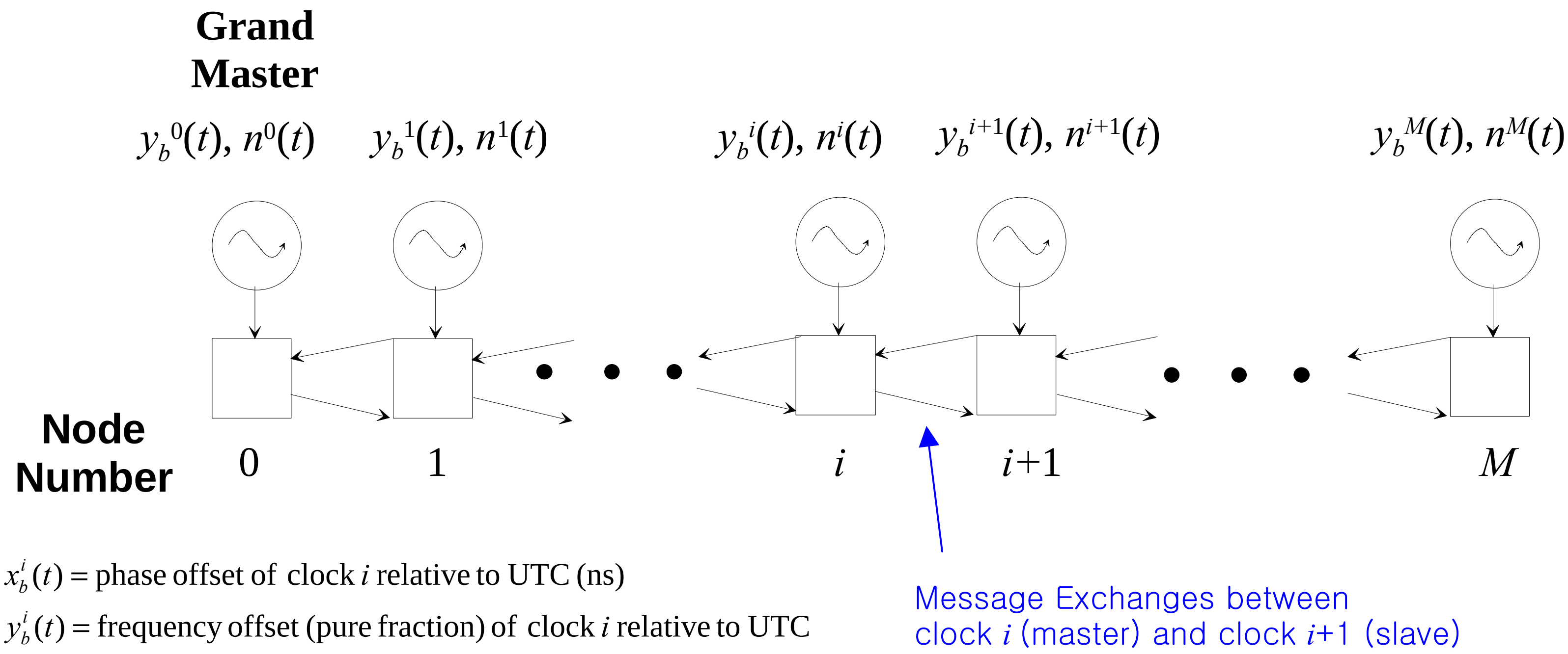
Synchronization Approaches - Review

Variations/Choices

- This is ongoing work; here we give further results for (4) with instantaneous frequency adjustments as in (3)
- plan to look at all the approaches

1) Use one-way time stamp scheme with less frequent two-way exchange; obtain delay from two-way exchange and assume delay is fixed until next two-way exchange
2) Instantaneous phase adjustments at intermediate nodes
3) Instantaneous phase and frequency adjustments at intermediate nodes (with instantaneous frequency adjustments possibly less frequent)
   – Described in [4]
4) **Filtered phase adjustments at intermediate nodes, using digital filter running at local clock rate (with or without instantaneous frequency adjustments)**
5) Full phase-locked loops (PLLs) at intermediate nodes (i.e., filtered phase and frequency adjustments)
6) Use of transparent clock nodes
   a) End-to-end versus peer-to-peer
   b) Whether or not to adjust rate of local oscillator in transparent clock and, if so, whether to do filtering
7) Time stamp reflects current time versus delay by some number of frames
8) Time stamp reflects local free-running clock time versus latest corrected time based on most recent time stamps and possible filtering)

Synchronization Model - Review



$x_b^i(t) = \text{phase offset of clock } i \text{ relative to UTC (ns)}$

$y_b^i(t) = \text{frequency offset (pure fraction) of clock } i \text{ relative to UTC}$

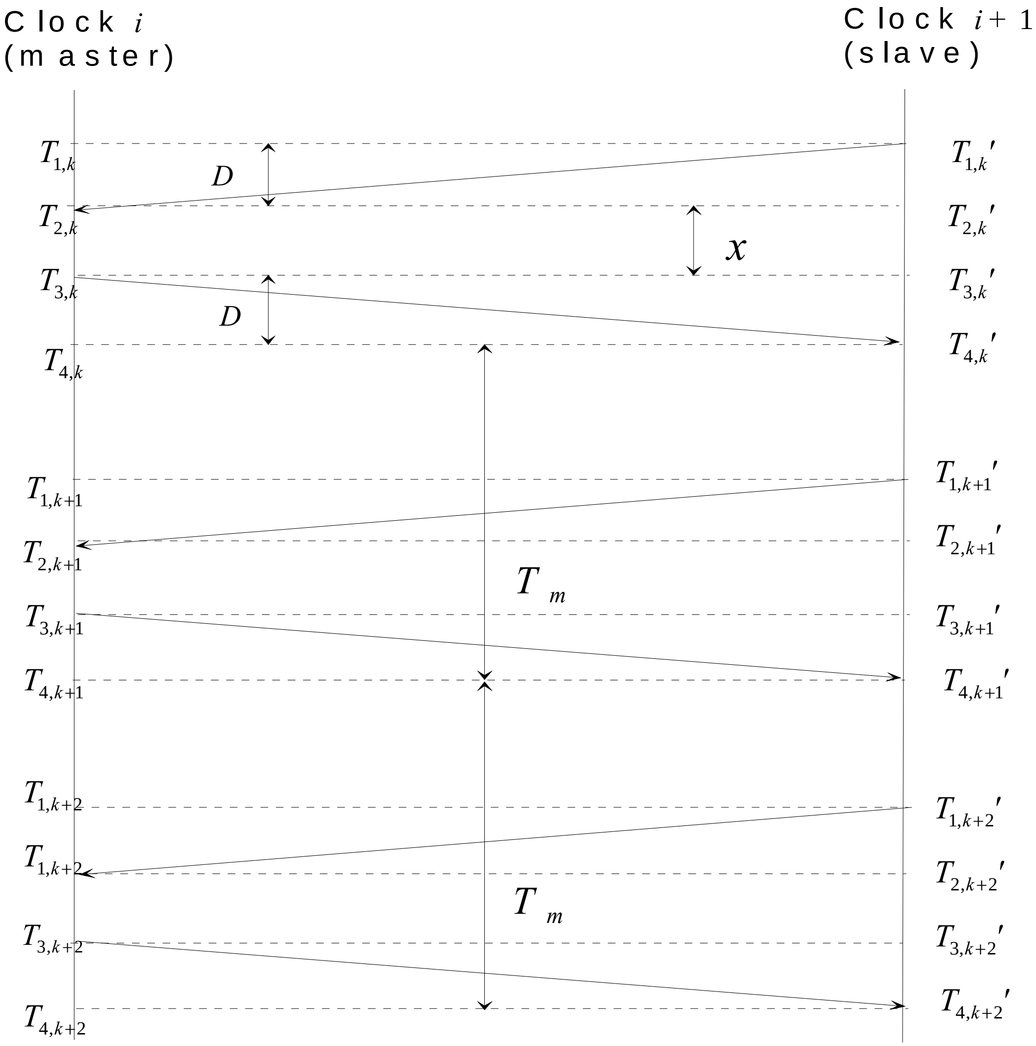
$n^i(t) = \text{phase noise of clock } i \text{ (ns)}$

$x_b^i(i) = \int_0^t y_b^i(t)\,dt + n^i(t) = y_b^i t + n^i(t)$

- Assumes the frequency offset is constant over time
- Assumes the phase offsets are zero at $t = 0$
- We are interested in timing relative to the GM; therefore, can set $y_b^0(t) = n^0(t) = 0$

–Note that messages from master to slave and slave to master do not necessarily occur at the same times
–Note that messages from master to slave and slave to master may not occur at the same rates

Synchronization Model - Review



T_m = time between successive messages from master to slave, measured relative to UTC

D = propation delay between master and slave

x = time offset between master and slave (will be initialized randomly between 0 and T_m and either kept constant or allowed to change by frequency offset between master and slave multiplied by T_m

- Assume $D << T_m$, and therefore probablity that messages from master to slave and slave to master overlap in time is negligible
- Unprimed quantities are relative to master clock
- Primed quantities are relative to slave clock

Then, can express the phase offset in discrete time (k = time index; UTC time at step $k = kT_m$)

$$x_{b,k}^{i} = y_{b}^{i} T_m k + n_k^i$$

Synchronization Model - Review

- Outline of simulation model implementation (see [2] for details)
  1) Compute phase offsets of the free-running node clocks, using frequency offsets (initialized randomly) and phase noise model
  2) Over each frequency adjustment interval, compute estimate of the frequency offset of each master clock relative to its downstream slave clock
  3) Accumulate the frequency offset estimates computed in (2) to obtain the frequency offset of each node relative to the Grand Master
  4) Using the most recently computed frequency offset estimate, compute an improved phase offset estimate for each slave clock
  5) Use the improved phase offset estimates computed in (4) to compute the correction (clockdelta) for each slave clock
  6) Accumulate the corrections computed in (5) to obtain the correction relative to the Grand Master
  7) Filter the sequence of corrections computed in (6) using an appropriate low-pass filter (at present, 2^{nd} order, linear filter with specified equivalent bandwidth and equivalent gain peaking)

Synchronization Model - Review

- Additional aspects of model (see [2] for more detail)
  - Clock noise model is described in Appendix I
  - Simulation time step is a sub-multiple of the inter-message time T_m (cannot exceed T_m)
  - Time between frequency estimate updates is a multiple of T_m
  - Time offset between master$\rightarrow$slave and slave$\rightarrow$master messages may be initialized randomly or initialized with user-specified values
  - Time offset between master$\rightarrow$slave and slave$\rightarrow$master messages may remain constant over the simulation or vary over T_m by the relative frequency offset between master and slave, multiplied by T_m
    - Former requires that the master and slave send messages at the same rate
    - Latter corresponds to messages being sent at the free-running clock rates
  - Finite precision of clock is modeled
    - Granularity, in units of time, is supplied as input parameter
    - Granularity to which frequency can be adjusted is supplied as an input parameter (pure fraction); this effect was not included in [1] and [2]

Parameters Common to All Simulation Cases

- Total of13 simulation cases
- 10 hops
  - GM followed by 10 slave clocks, in chain
- Slave clock frequency tolerance = ± 100 ppm
- Filter at egress modeled as 2nd order, linear filter with 20 dB/decade roll-off
  - Filter BW = 10 Hz in cases 1 – 12, 1 Hz in case 13
  - Filter gain peaking = 0.1 dB
- Clock noise generation model of [1] and [2] used (see Appendix I for details)
- Simulation time step = 0.01 ms
  - Used small time step to ensure phase peaks were captured
- In all cases, frequency updates are performed, though less frequently than phase updates

Parameters Common to All Simulation Cases

- In all cases, filtering of the phase updates is assumed
  - The model development in [2] indicates that filtering the updates at each successive node and adding, versus adding the unfiltered updates and filtering at the endpoint, are equivalent because the operation is linear
    - Which one to do can be left to implementation
- The results of [1] and [2] indicated that both frequency updates and filtering are necessary if all the MTIE requirements are to be met
- If the uncompressed digital video requirements need not be met, it might be possible to forego either frequency updates or filtering (but not both)
  - However, [1] and [2] did not consider the effects of more realistic phase measurement granularity, nor phase measurement error (nor frequency adjustment granularity)
    - These effects will cause MTIE to increase
  - Therefore, for now it is assumed that frequency updates and filtering will most likely be necessary, so we focus on this case

Summary of Simulation Cases

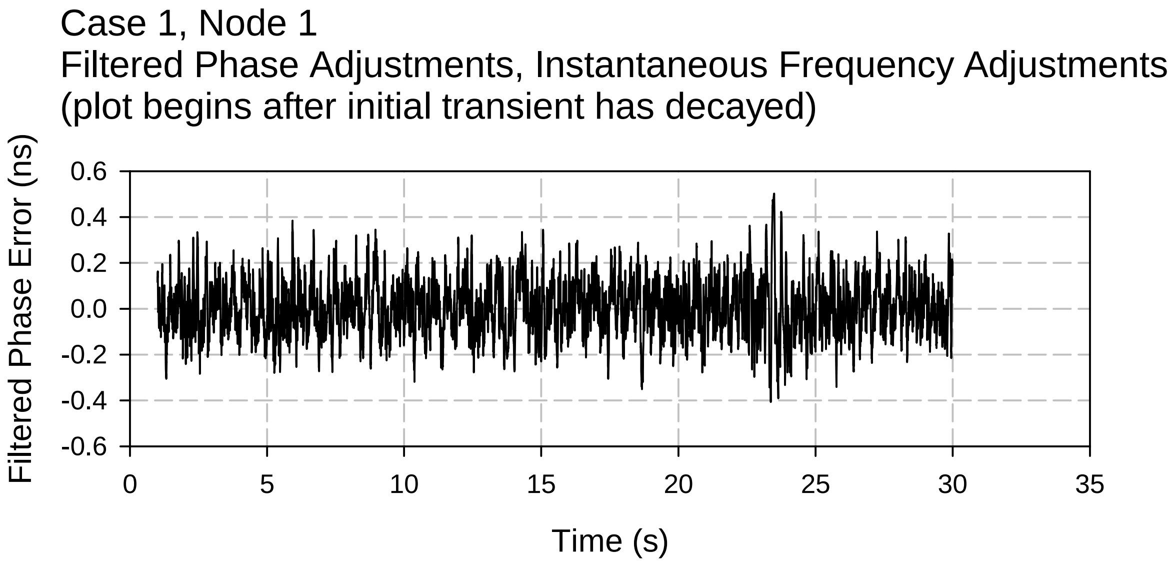
Case	Synch Interval (ms)	Freq adjust interval (ms)	Phase gran (ns)	Freq granul (fraction)	Filt bandwidth (Hz)	Message rates syntonized (y/n)
1	10	100	1	2.3283×10^{-10}	10	n
2	10	100	1	10^{-6}	10	n
3	10	100	1	10^{-6}	10	y
4	10	100	1	10^{-7}	10	y
5	10	100	40	2.3283×10^{-10}	10	n
6	10	100	40	2.3283×10^{-10}	10	y
7	1	10	40	2.3283×10^{-10}	10	y
8	1	100	40	2.3283×10^{-10}	10	y
9	1	1000	40	2.3283×10^{-10}	10	y
10	10	100	40	5.9605×10^{-8}	10	y
11	10	100	40	1.5259×10^{-5}	10	y
12	0.1	1000	40	2.3283×10^{-10}	10	y
13	1	1000	40	2.3283×10^{-10}	1	y

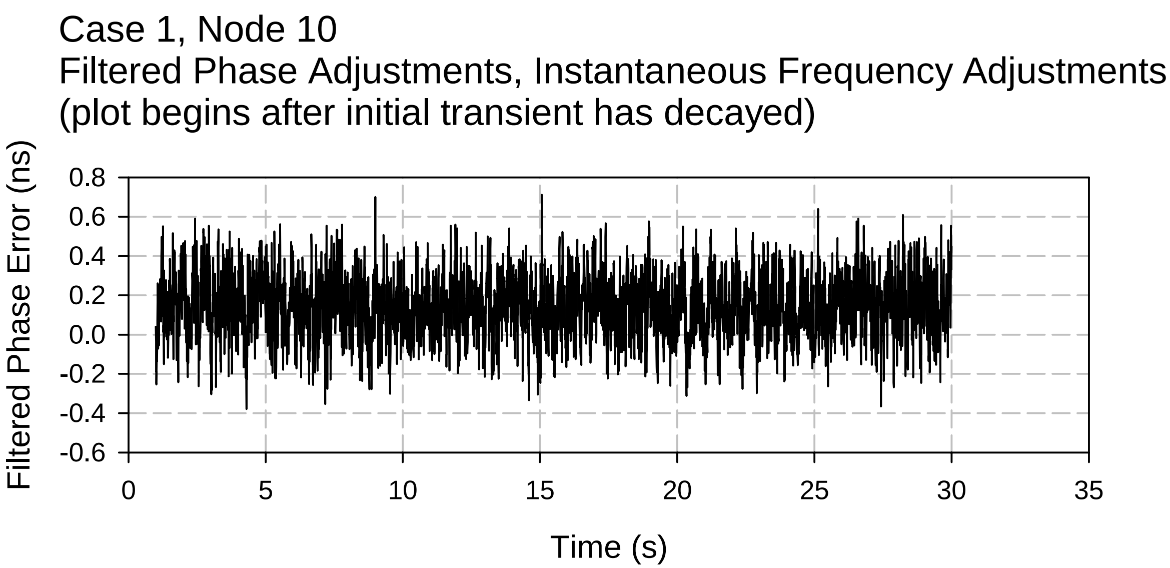
Summary of Simulation Cases (Cont)

❑Note on frequency granularity

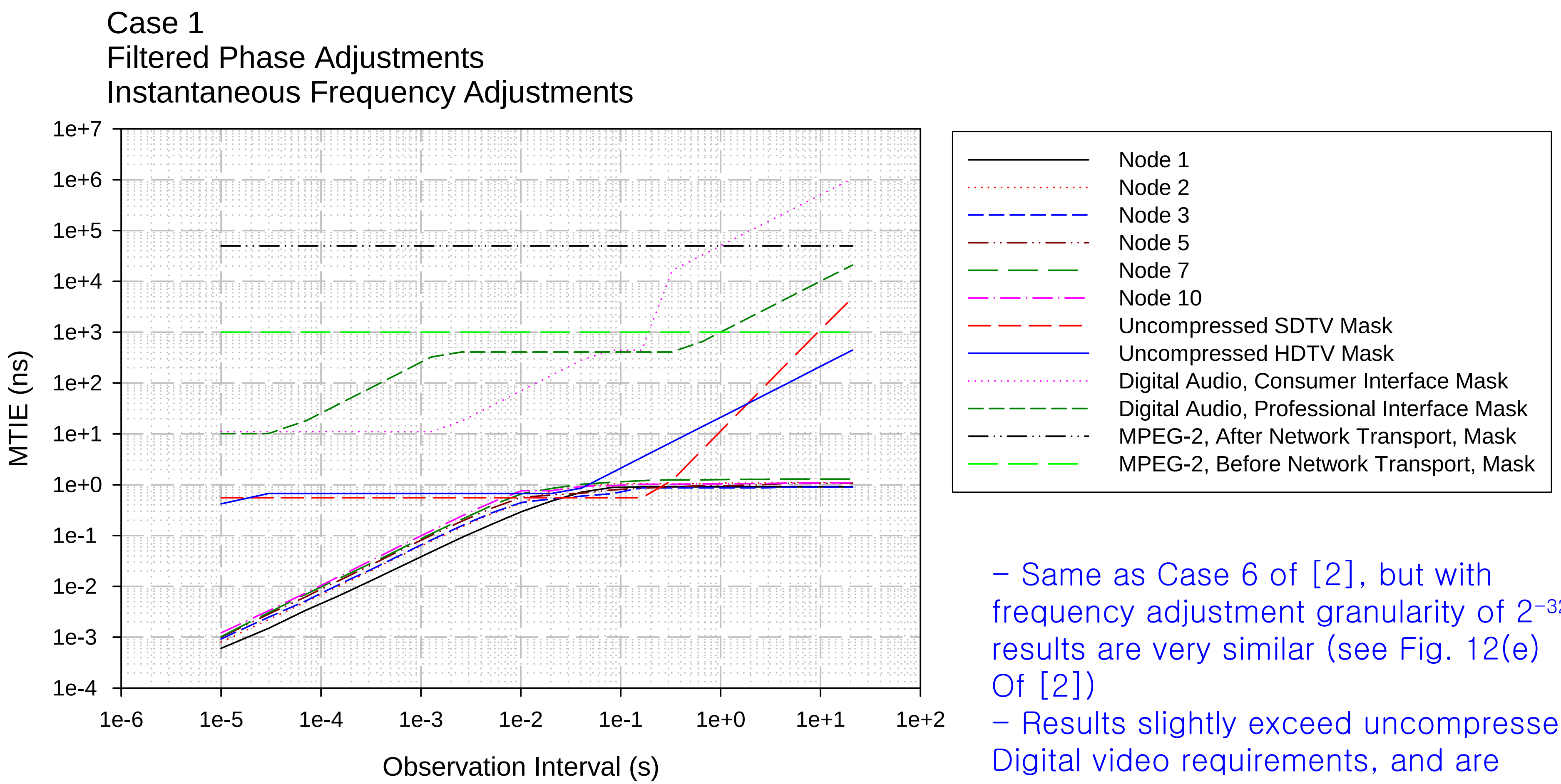
- $2.3283 \times 10^{-10} = 2^{-32}$
- $5.9605 \times 10^{-8} = 2^{-24}$
- $1.5259 \times 10^{-5} = 2^{-16}$

Simulation Case 1 – Phase Error





Simulation Case 1 - MTIE

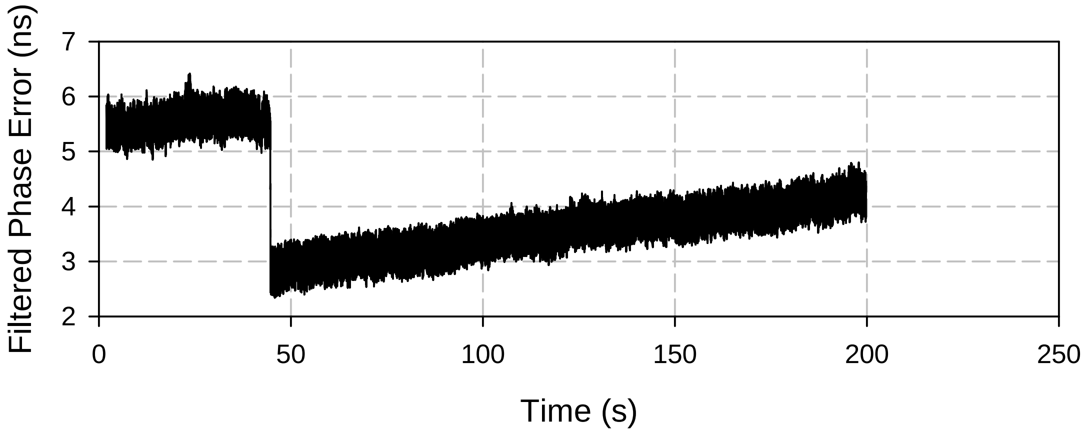


– Same as Case 6 of [2], but with frequency adjustment granularity of 2^{-32}; results are very similar (see Fig. 12(e) Of [2])

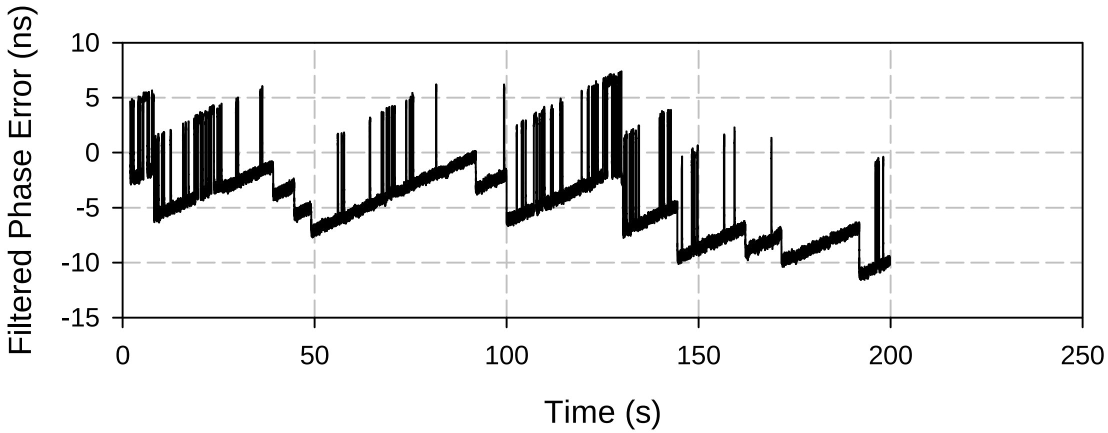
– Results slightly exceed uncompressed Digital video requirements, and are Within audio and compressed video requirements

Simulation Case 2 – Phase

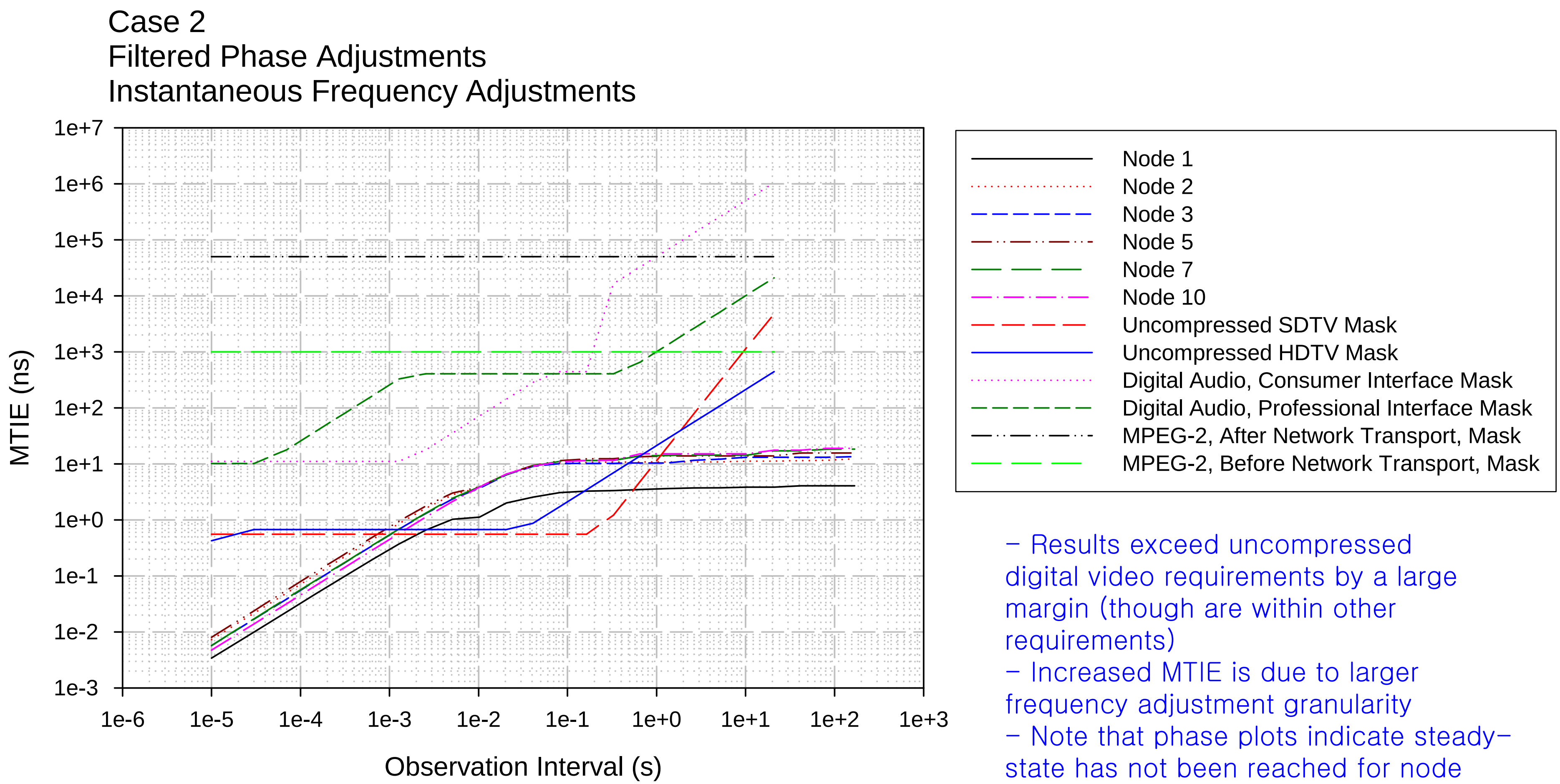
Case 2, Node 1
Filtered Phase Adjustments, Instantaneous Frequency Adjustments
(plot begins after initial transient has decayed)



Case 2, Node 10
Filtered Phase Adjustments, Instantaneous Frequency Adjustments
(plot begins after initial transient has decayed)



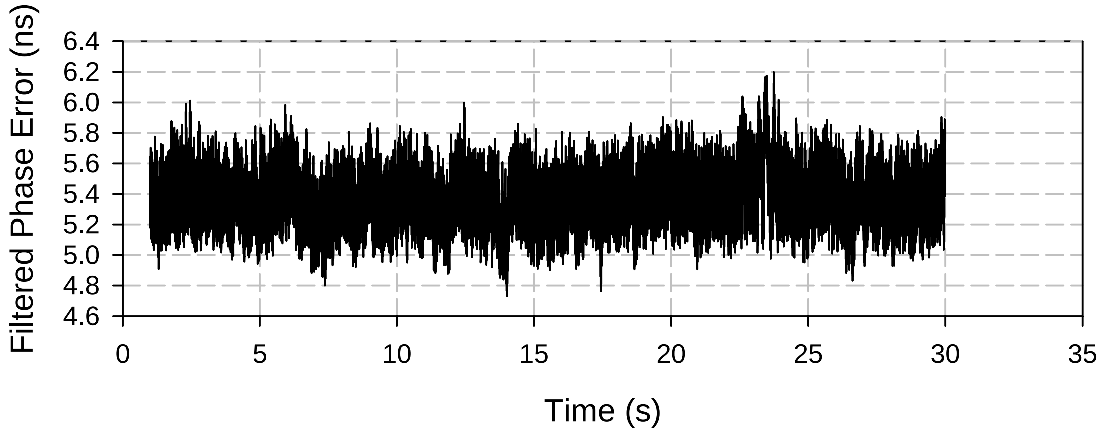
Simulation Case 2 - MTIE



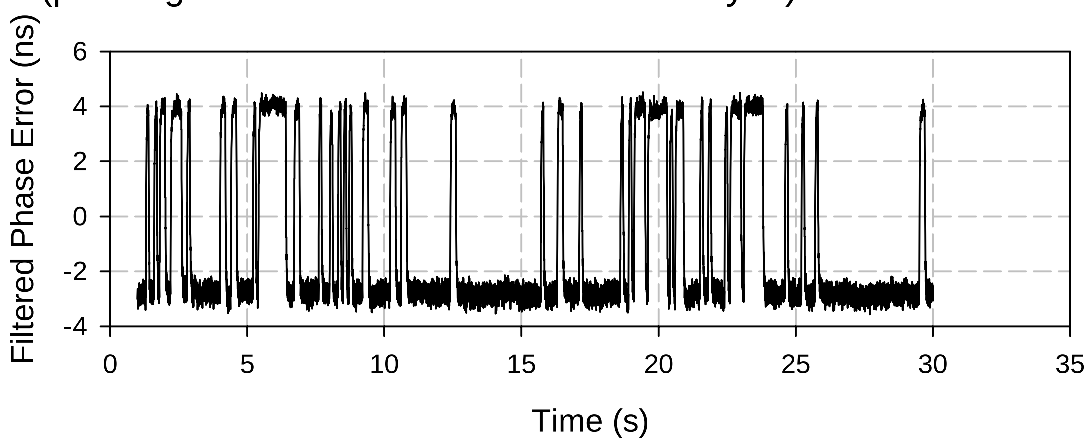
- Results exceed uncompressed digital video requirements by a large margin (though are within other requirements)
- Increased MTIE is due to larger frequency adjustment granularity
- Note that phase plots indicate steady-state has not been reached for node 1 (and may not have been reached for node 10); i.e., true MTIE may be larger

Simulation Case 3 – Phase Error

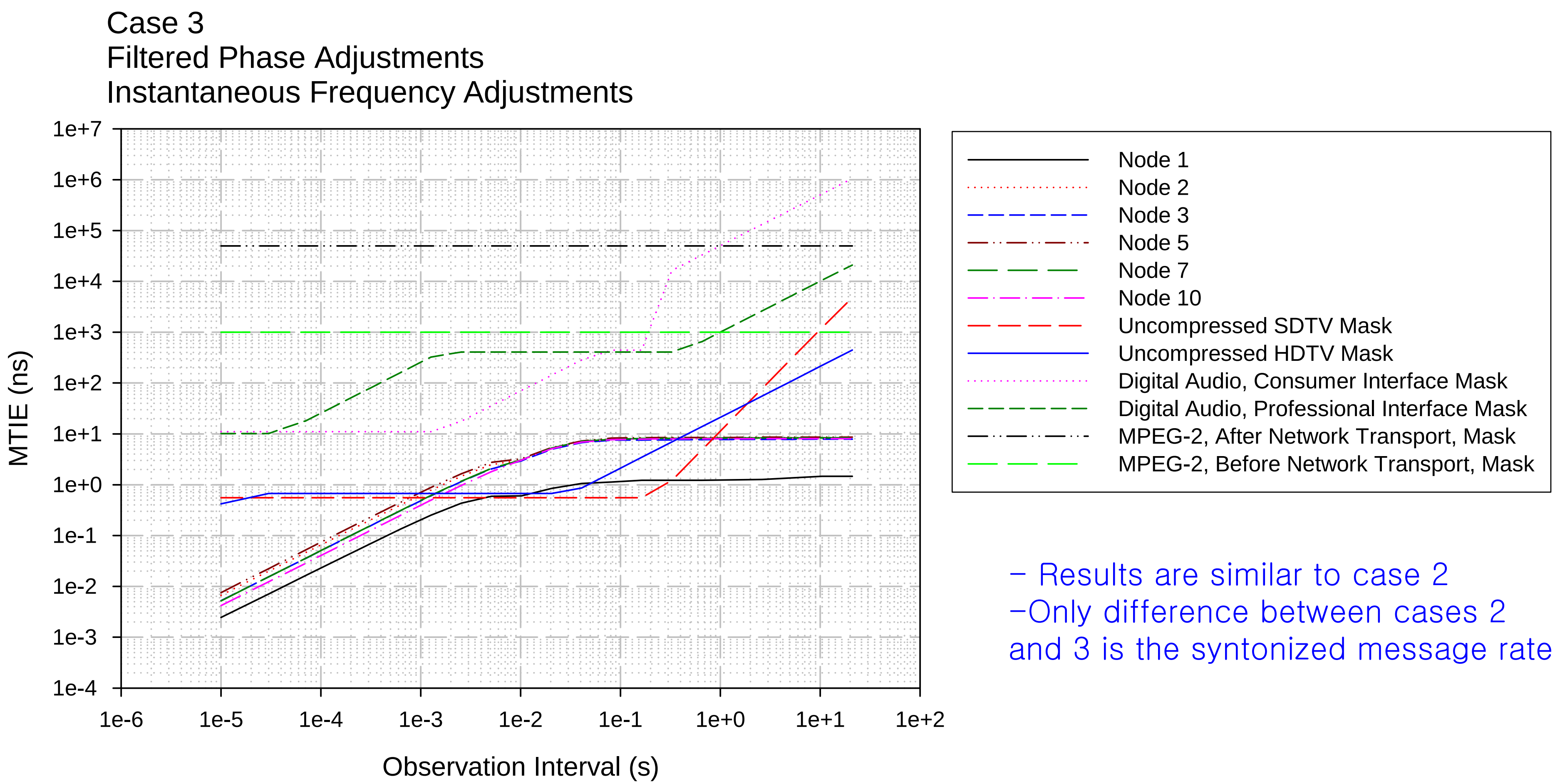
Case 3, Node 1
Filtered Phase Adjustments, Instantaneous Frequency Adjustments
(plot begins after initial transient has decayed)



Case 3, Node 10
Filtered Phase Adjustments, Instantaneous Frequency Adjustments
(plot begins after initial transient has decayed)



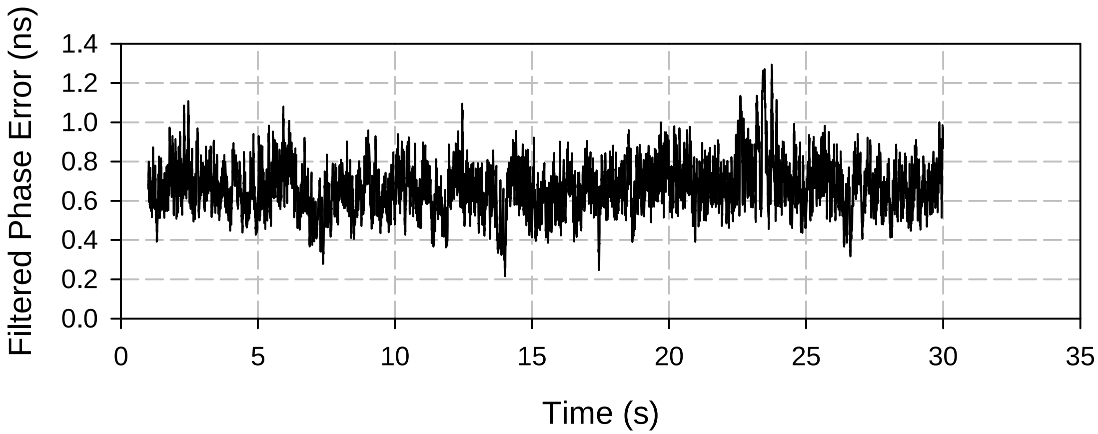
Simulation Case 3 - MTIE



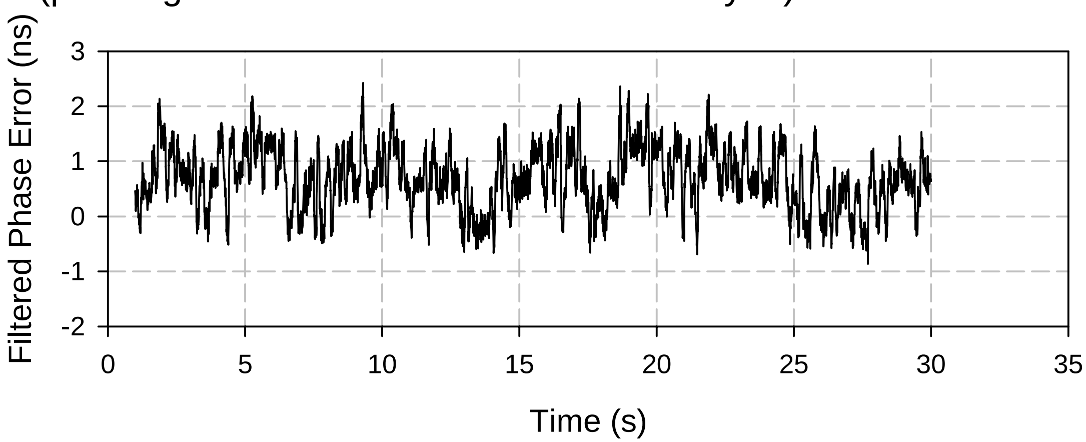
- Results are similar to case 2
- Only difference between cases 2 and 3 is the syntonized message rates

Simulation Case 4 – Phase Error

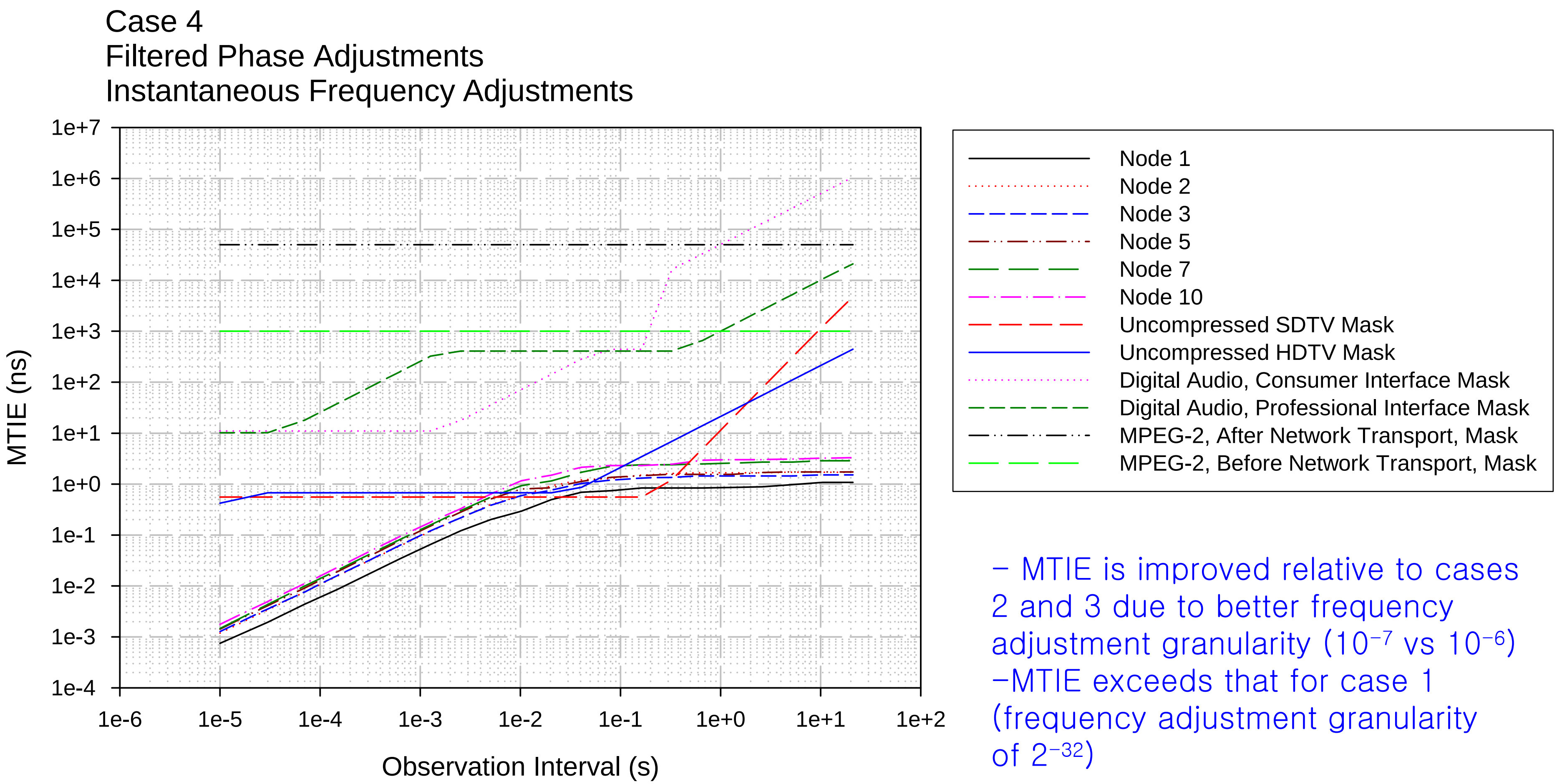
Case 4, Node 1
Filtered Phase Adjustments, Instantaneous Frequency Adjustments
(plot begins after initial transient has decayed)



Case 4, Node 10
Filtered Phase Adjustments, Instantaneous Frequency Adjustments
(plot begins after initial transient has decayed)

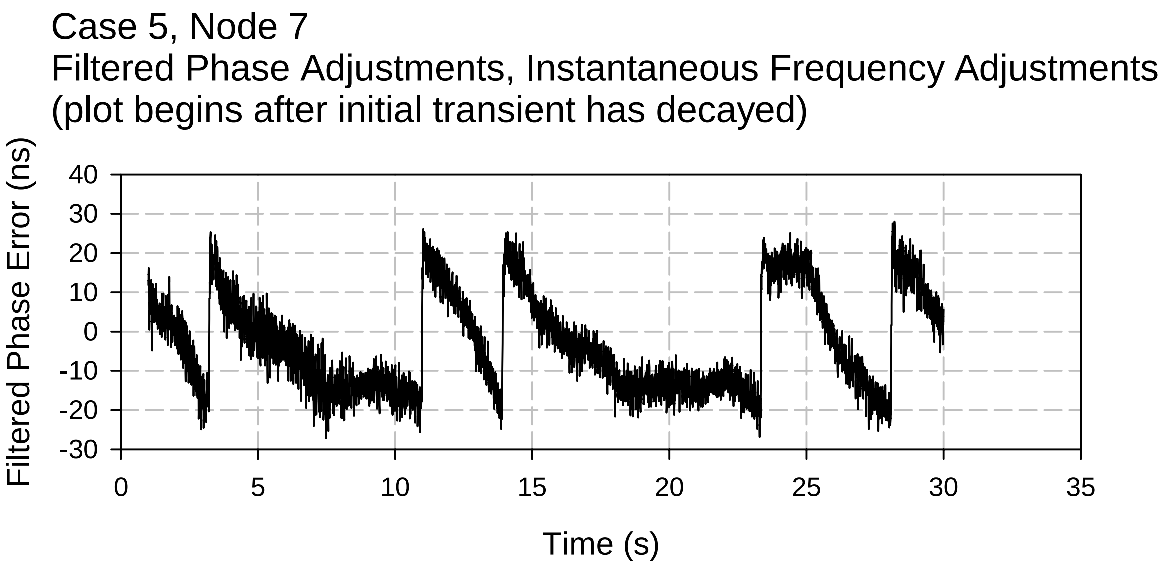


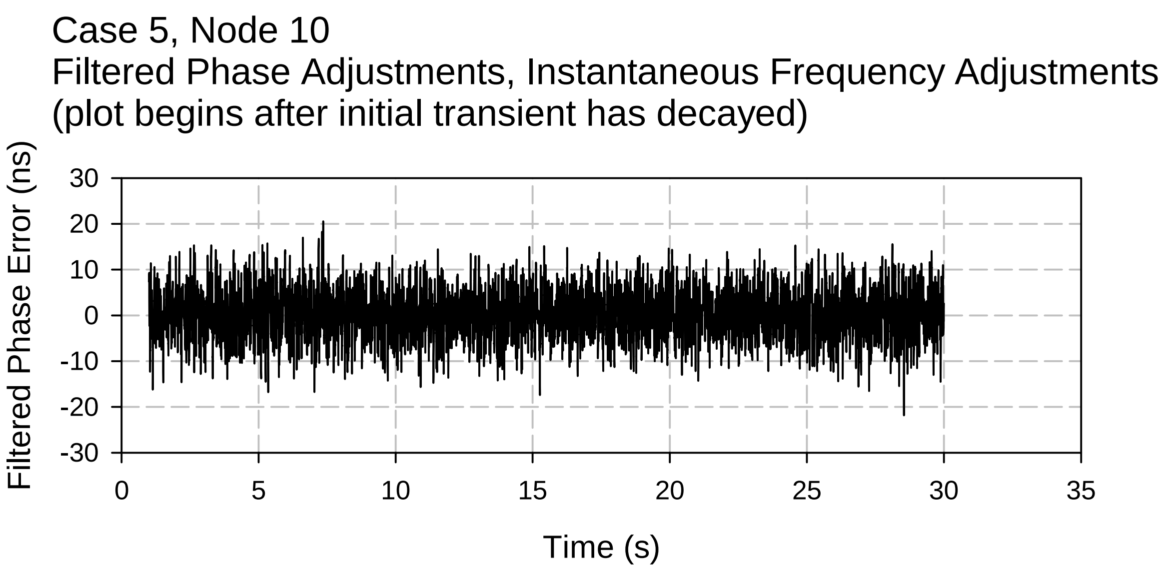
Simulation Case 4 - MTIE



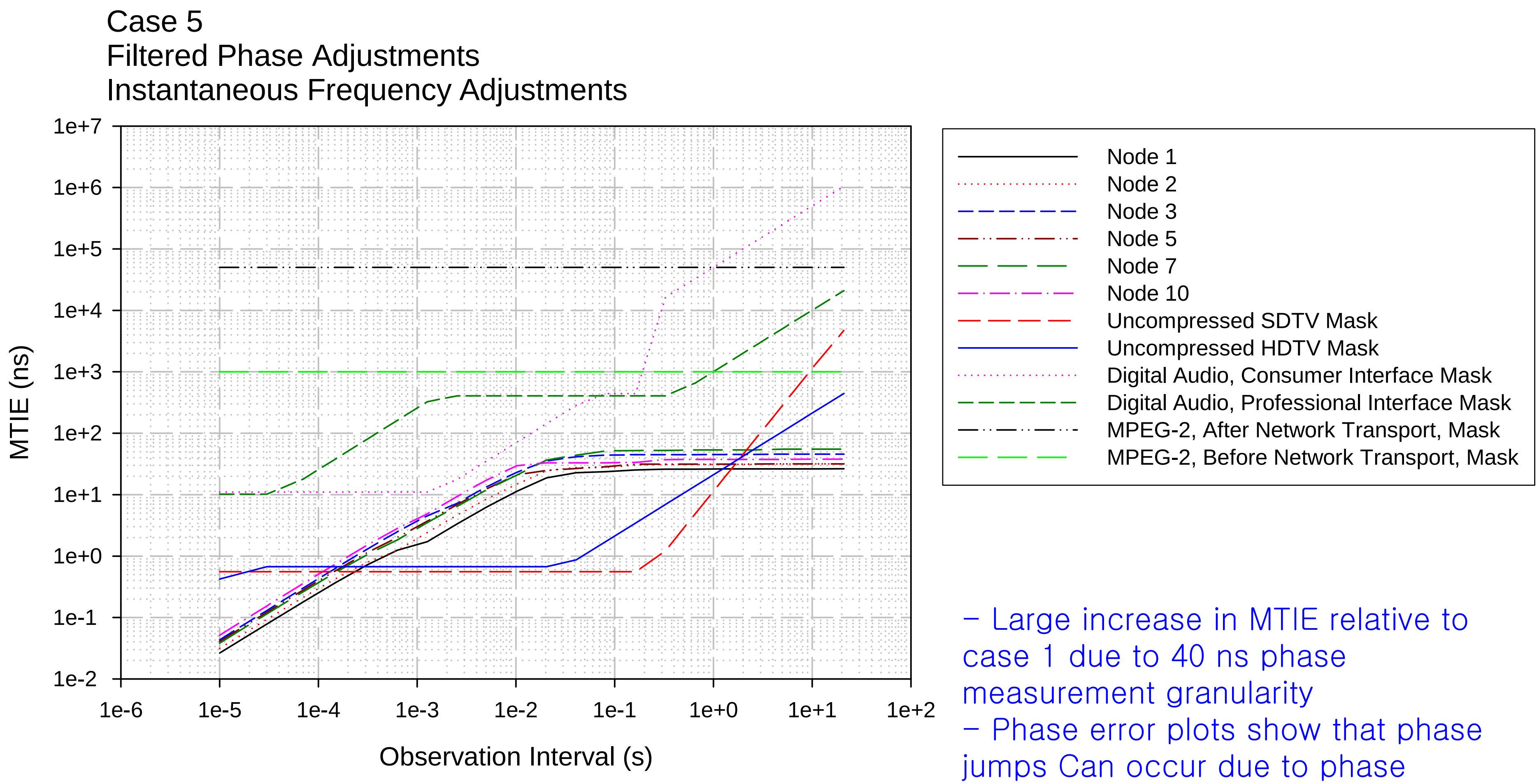
- MTIE is improved relative to cases 2 and 3 due to better frequency adjustment granularity (10^{-7} vs 10^{-6})
- MTIE exceeds that for case 1 (frequency adjustment granularity of 2^{-32})

Simulation Case 5 – Phase Error





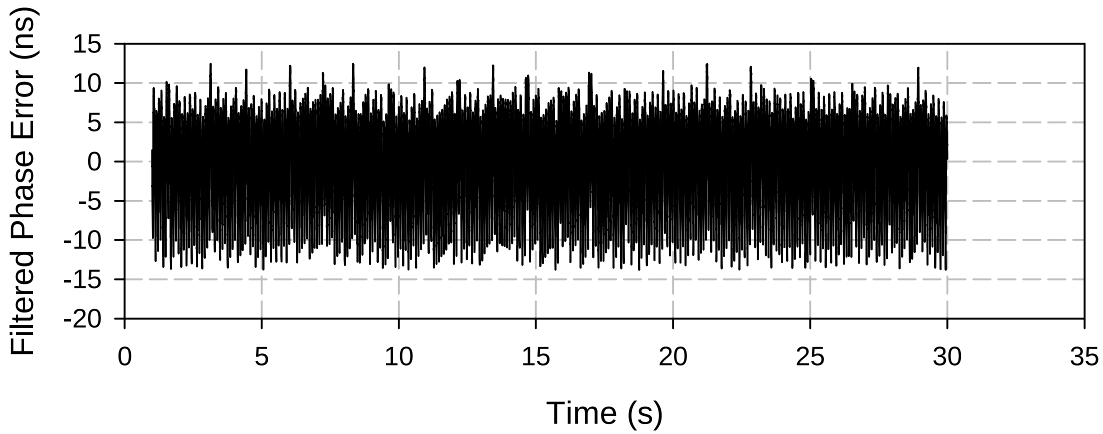
Simulation Case 5 - MTIE



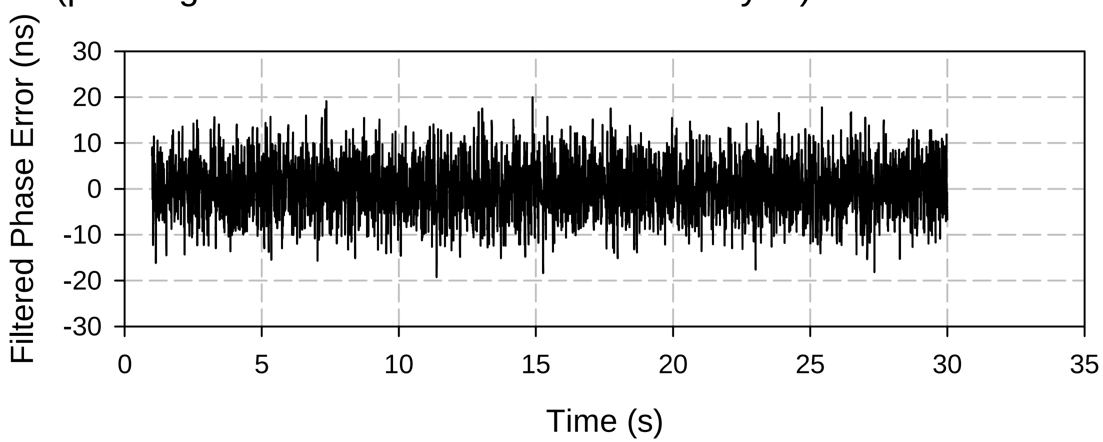
- Large increase in MTIE relative to case 1 due to 40 ns phase measurement granularity
- Phase error plots show that phase jumps Can occur due to phase measurement Granularity

Simulation Case 6 – Phase Error

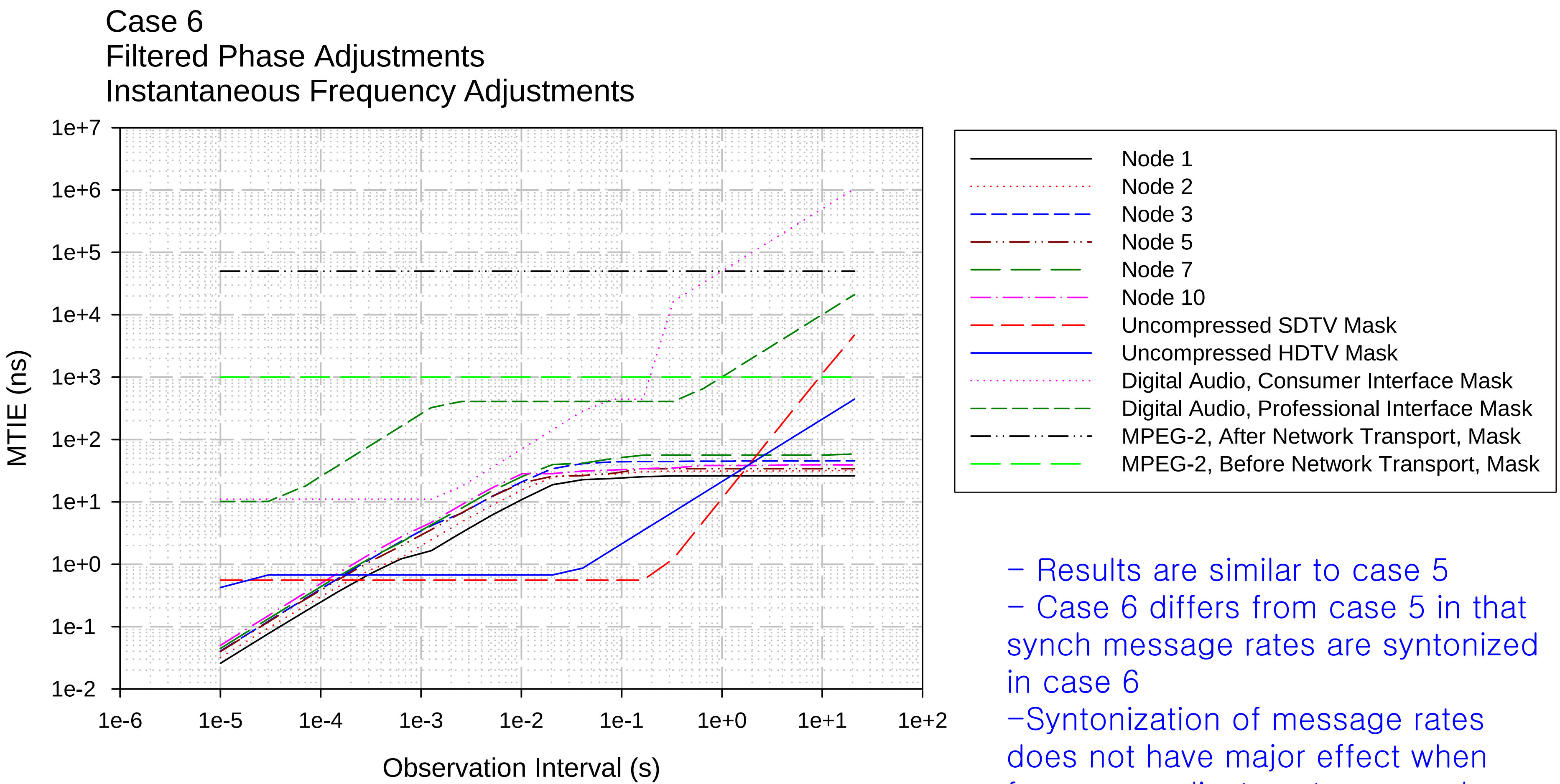
Case 6, Node 1
Filtered Phase Adjustments, Instantaneous Frequency Adjustments
(plot begins after initial transient has decayed)



Case 6, Node 10
Filtered Phase Adjustments, Instantaneous Frequency Adjustments
(plot begins after initial transient has decayed)



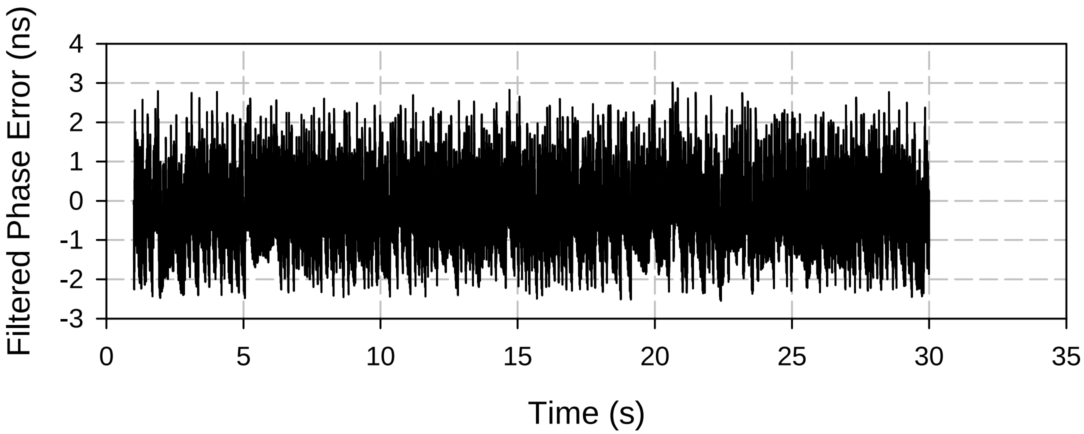
Simulation Case 6 – MTIE



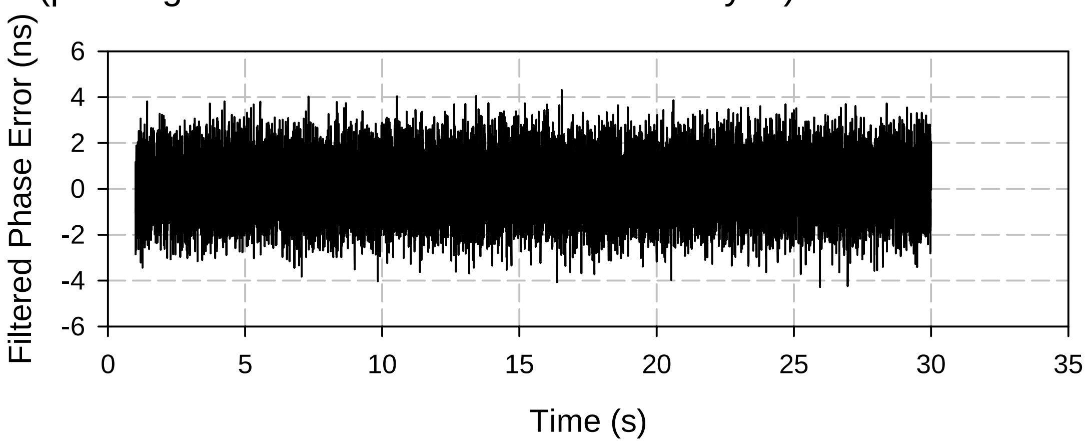
- Results are similar to case 5
- Case 6 differs from case 5 in that synch message rates are syntonized in case 6
- Syntonization of message rates does not have major effect when frequency adjustments are made (see [1] and [2])

Simulation Case 7 – Phase Error

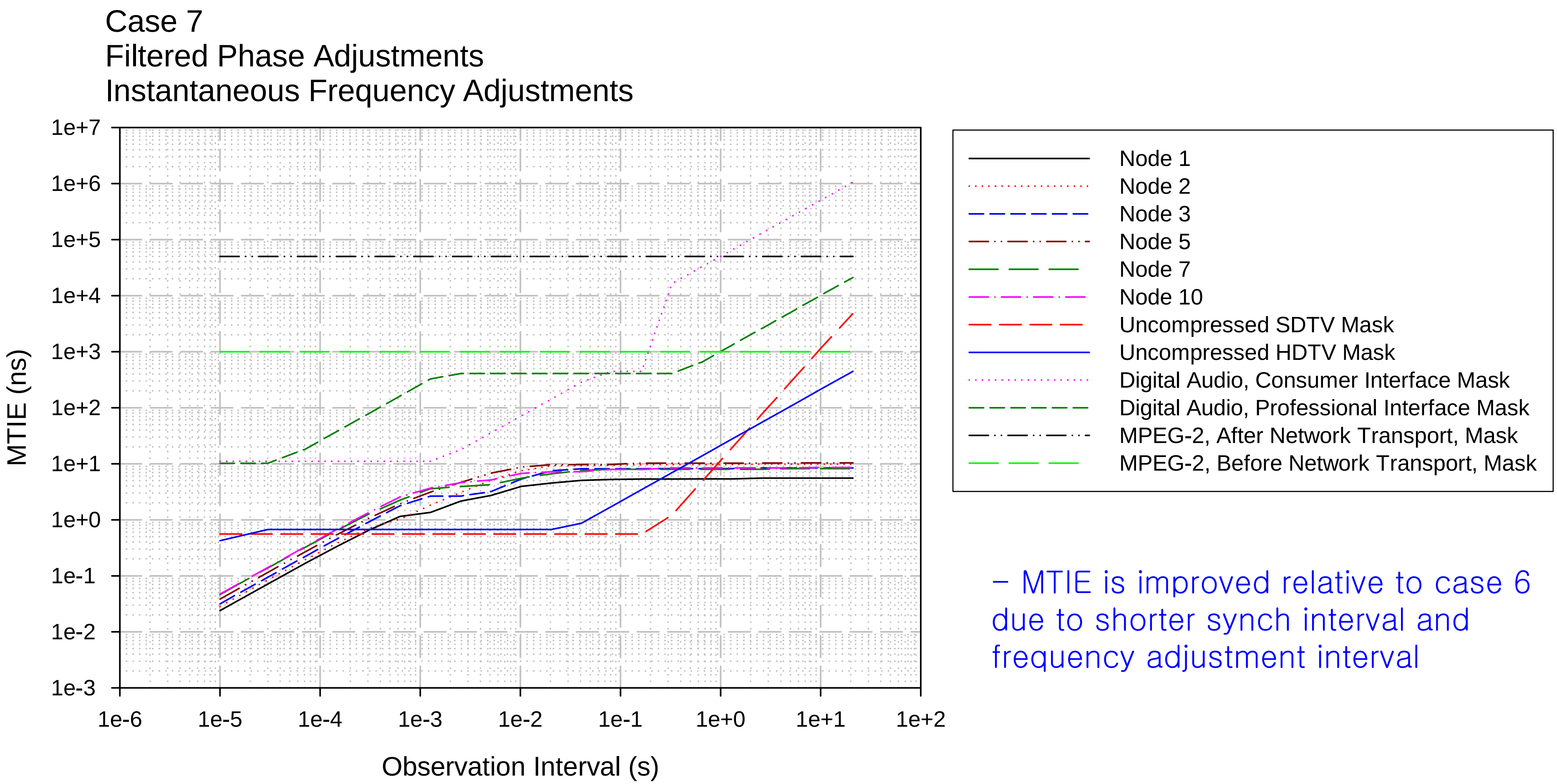
Case 7, Node 1
Filtered Phase Adjustments, Instantaneous Frequency Adjustments
(plot begins after initial transient has decayed)



Case 7, Node 10
Filtered Phase Adjustments, Instantaneous Frequency Adjustments
(plot begins after initial transient has decayed)



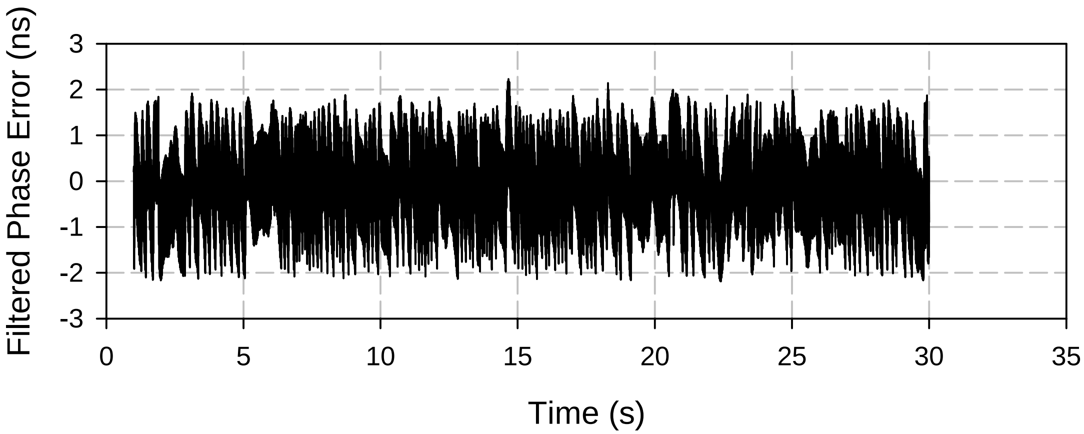
Simulation Case 7 - MTIE



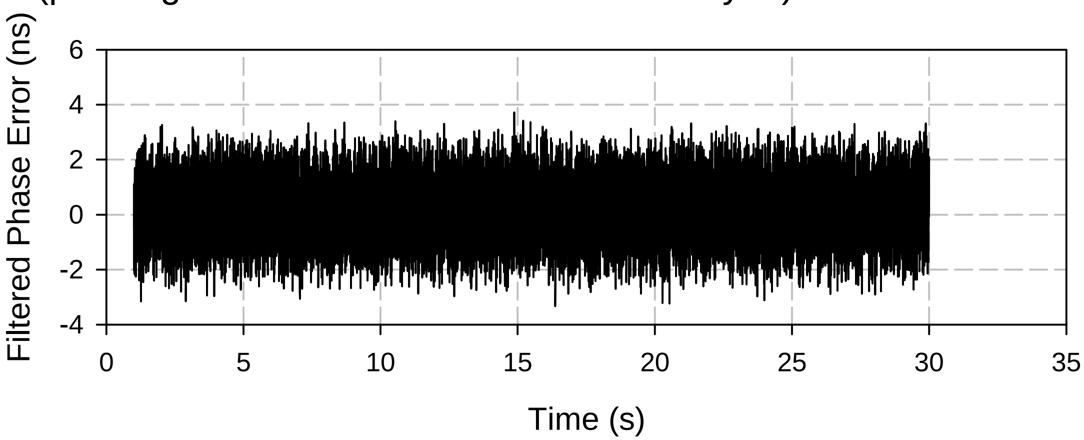
- MTIE is improved relative to case 6 due to shorter synch interval and frequency adjustment interval

Simulation Case 8 – Phase Error

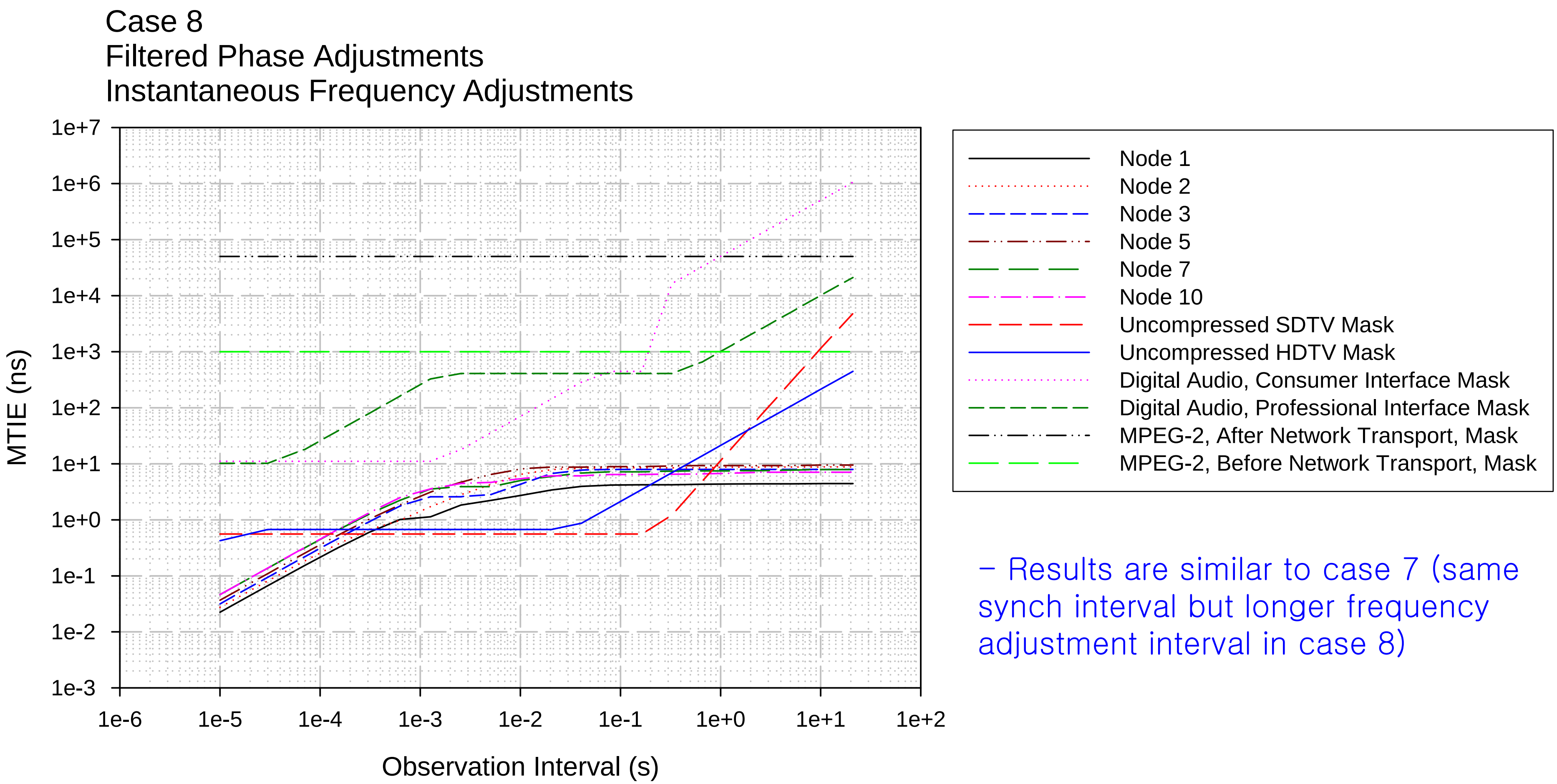
Case 8, Node 1
Filtered Phase Adjustments, Instantaneous Frequency Adjustments
(plot begins after initial transient has decayed)



Case8, Node 10
Filtered Phase Adjustments, Instantaneous Frequency Adjustments
(plot begins after initial transient has decayed)



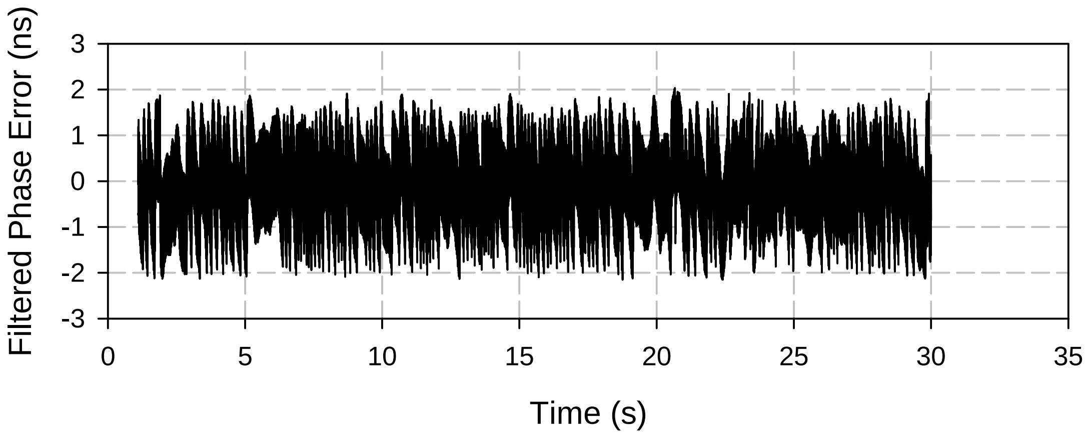
Simulation Case 8 - MTIE



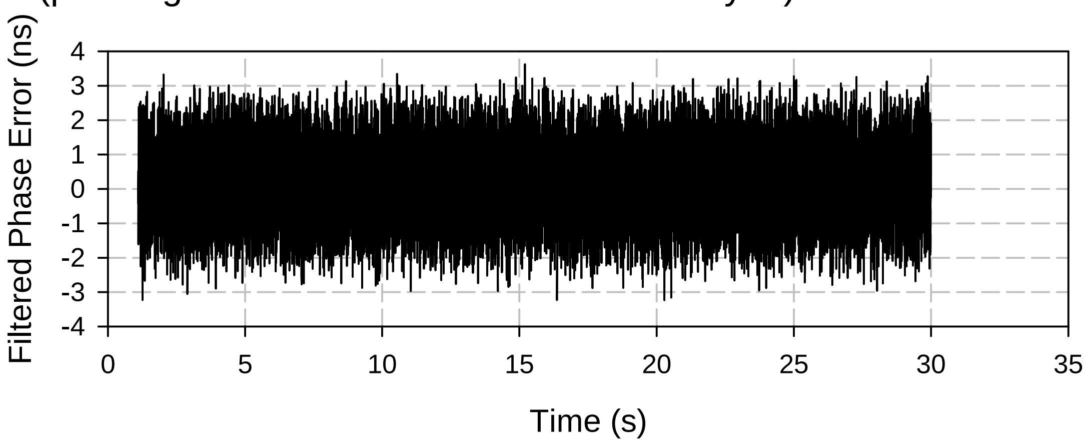
- Results are similar to case 7 (same synch interval but longer frequency adjustment interval in case 8)

Simulation Case 9 – Phase Error

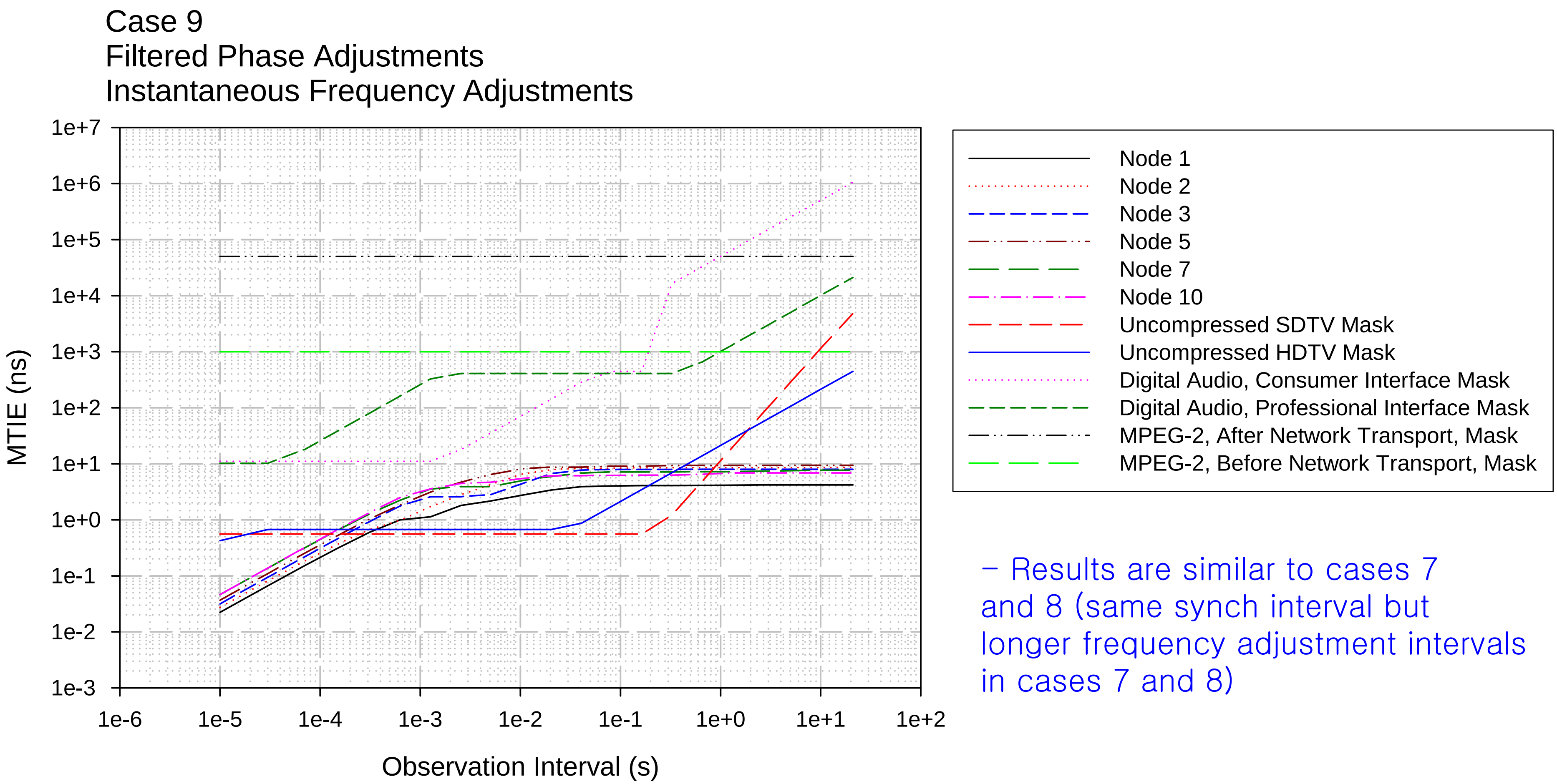
Case 9, Node 1
Filtered Phase Adjustments, Instantaneous Frequency Adjustments
(plot begins after initial transient has decayed)



Case9, Node 10
Filtered Phase Adjustments, Instantaneous Frequency Adjustments
(plot begins after initial transient has decayed)



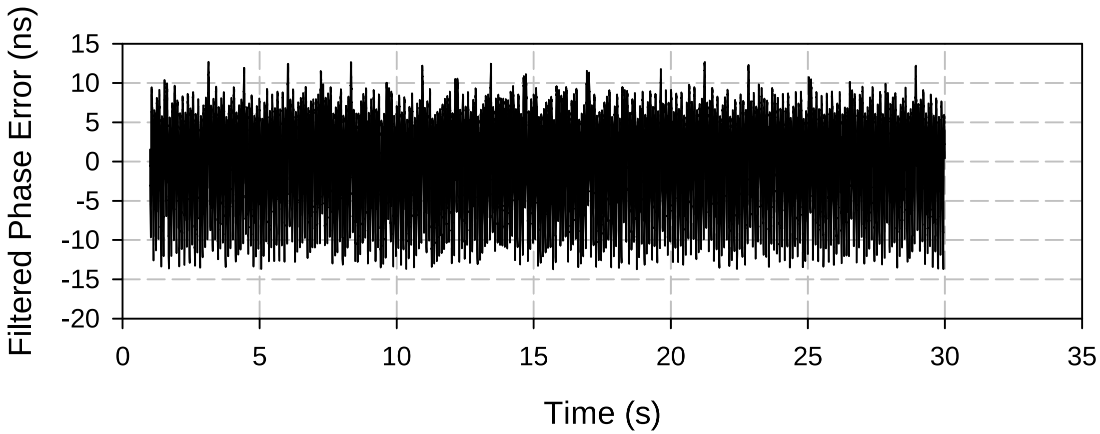
Simulation Case 9 - MTIE



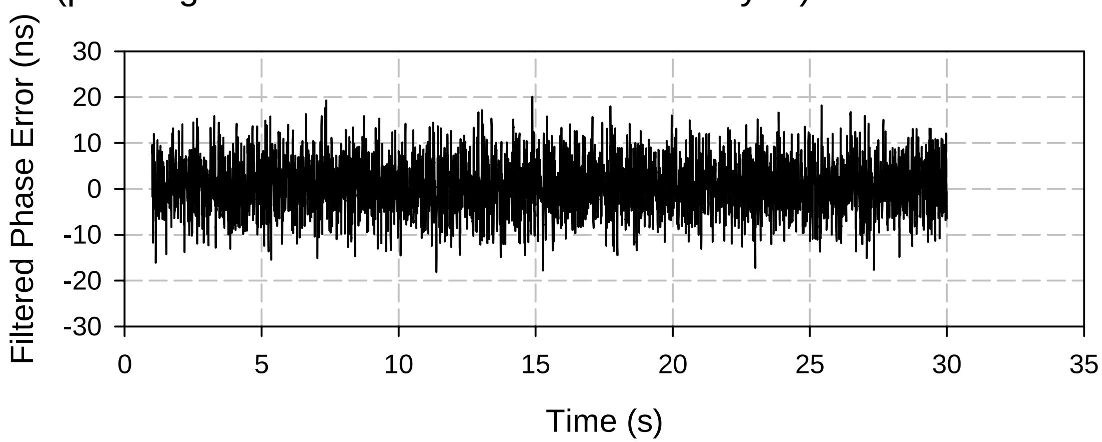
– Results are similar to cases 7 and 8 (same synch interval but longer frequency adjustment intervals in cases 7 and 8)

Simulation Case 10 – Phase Error

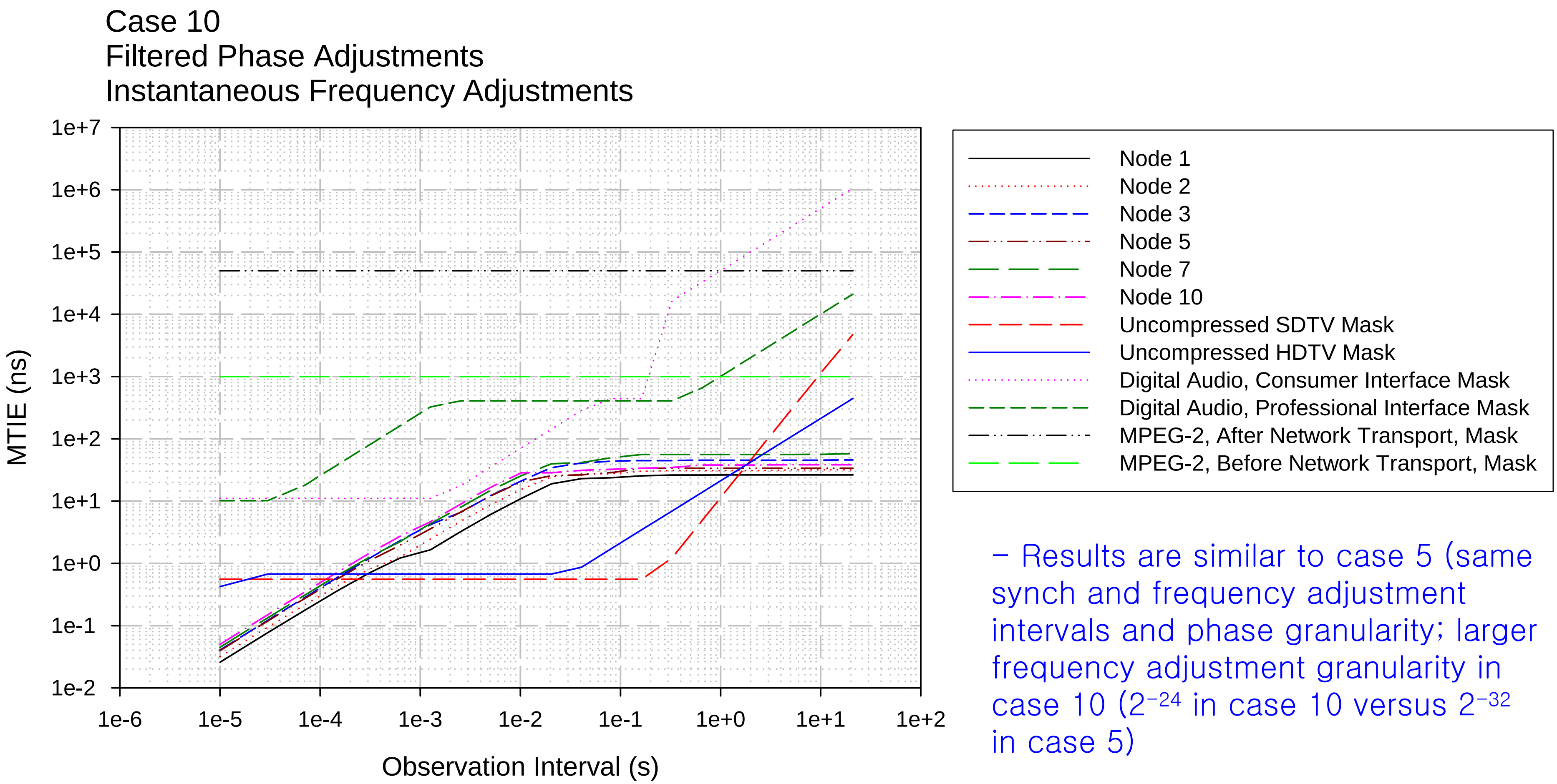
Case 10, Node 1
Filtered Phase Adjustments, Instantaneous Frequency Adjustments
(plot begins after initial transient has decayed)



Case10, Node 10
Filtered Phase Adjustments, Instantaneous Frequency Adjustments
(plot begins after initial transient has decayed)

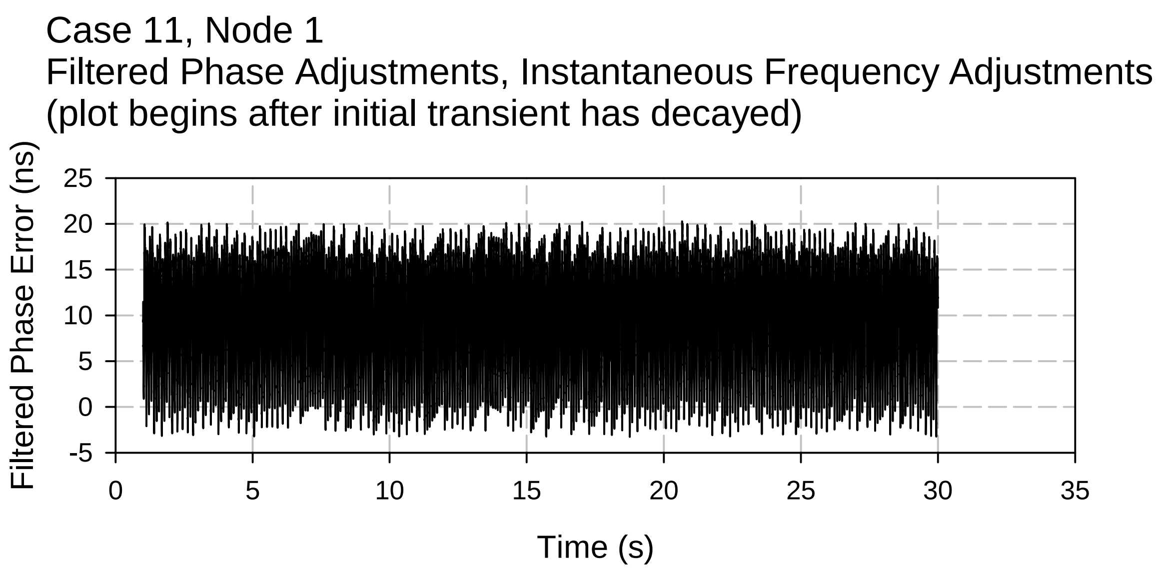


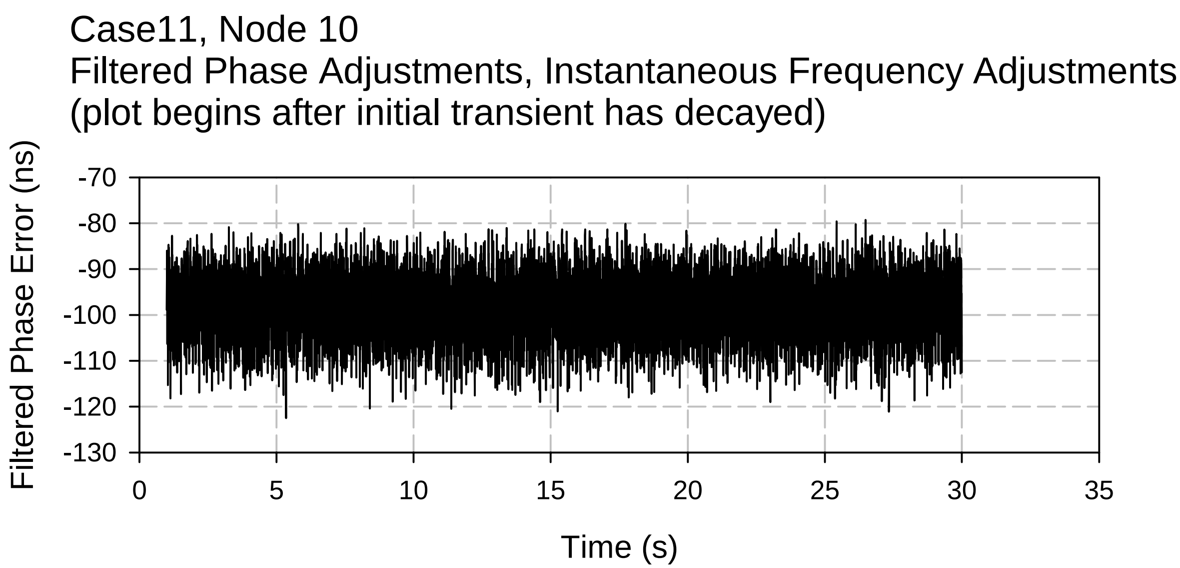
Simulation Case 10 - MTIE



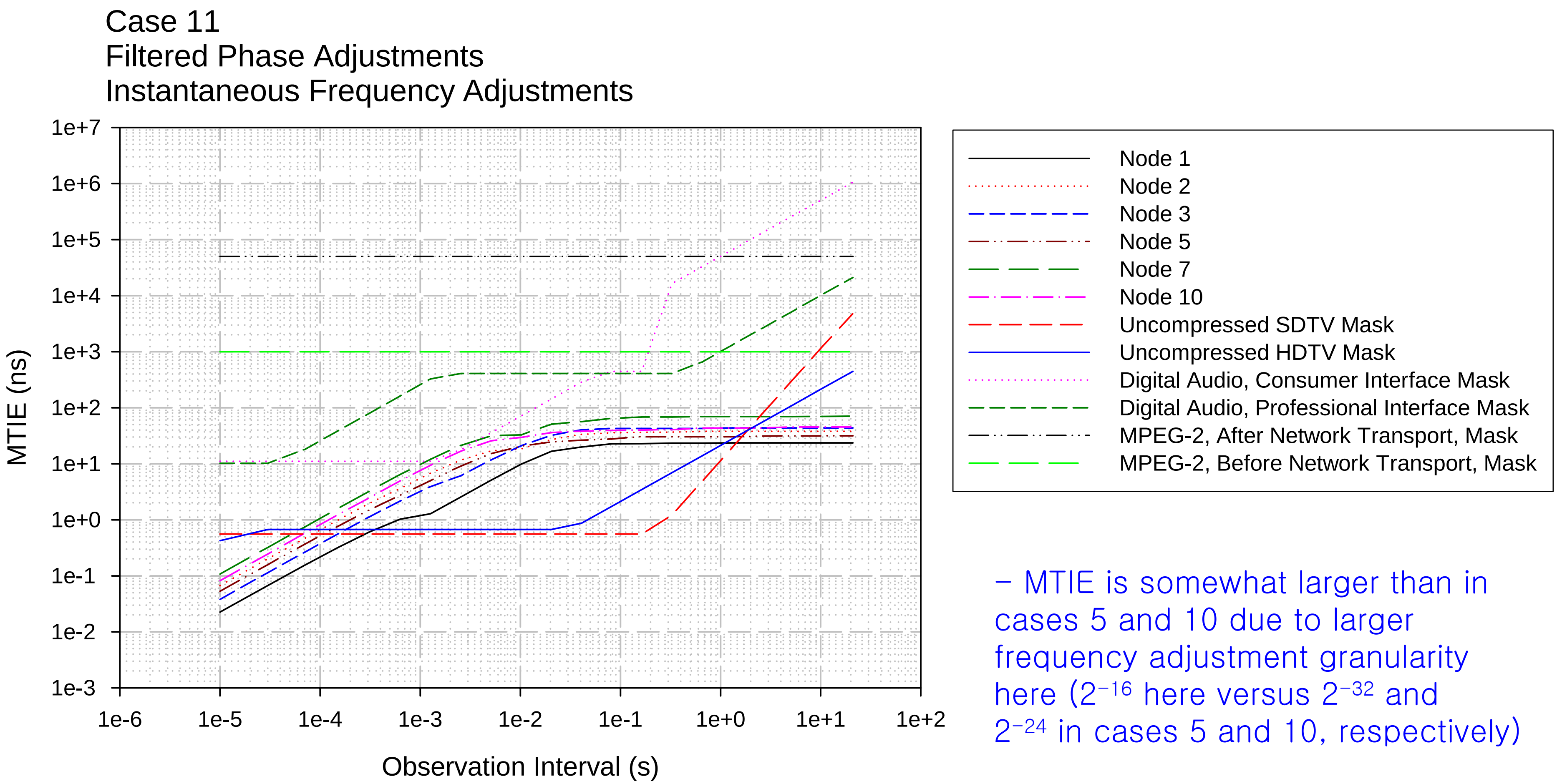
– Results are similar to case 5 (same synch and frequency adjustment intervals and phase granularity; larger frequency adjustment granularity in case 10 (2^{-24} in case 10 versus 2^{-32} in case 5)

Simulation Case 11 – Phase Error





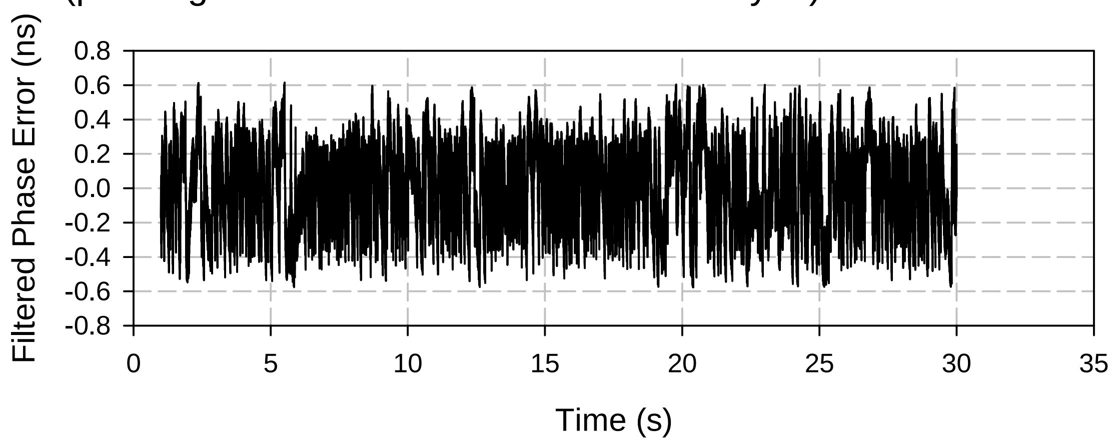
Simulation Case 11 - MTIE



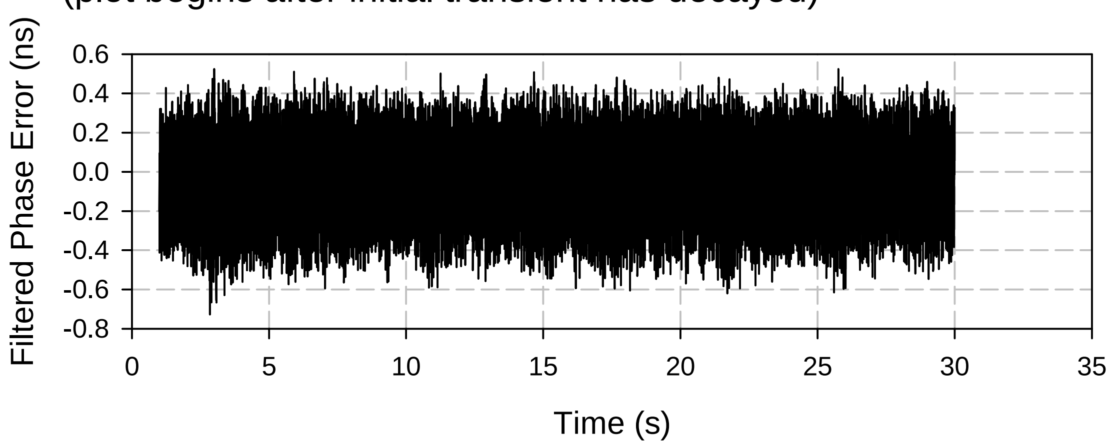
– MTIE is somewhat larger than in cases 5 and 10 due to larger frequency adjustment granularity here (2^{-16} here versus 2^{-32} and 2^{-24} in cases 5 and 10, respectively)

Simulation Case 12 – Phase Error

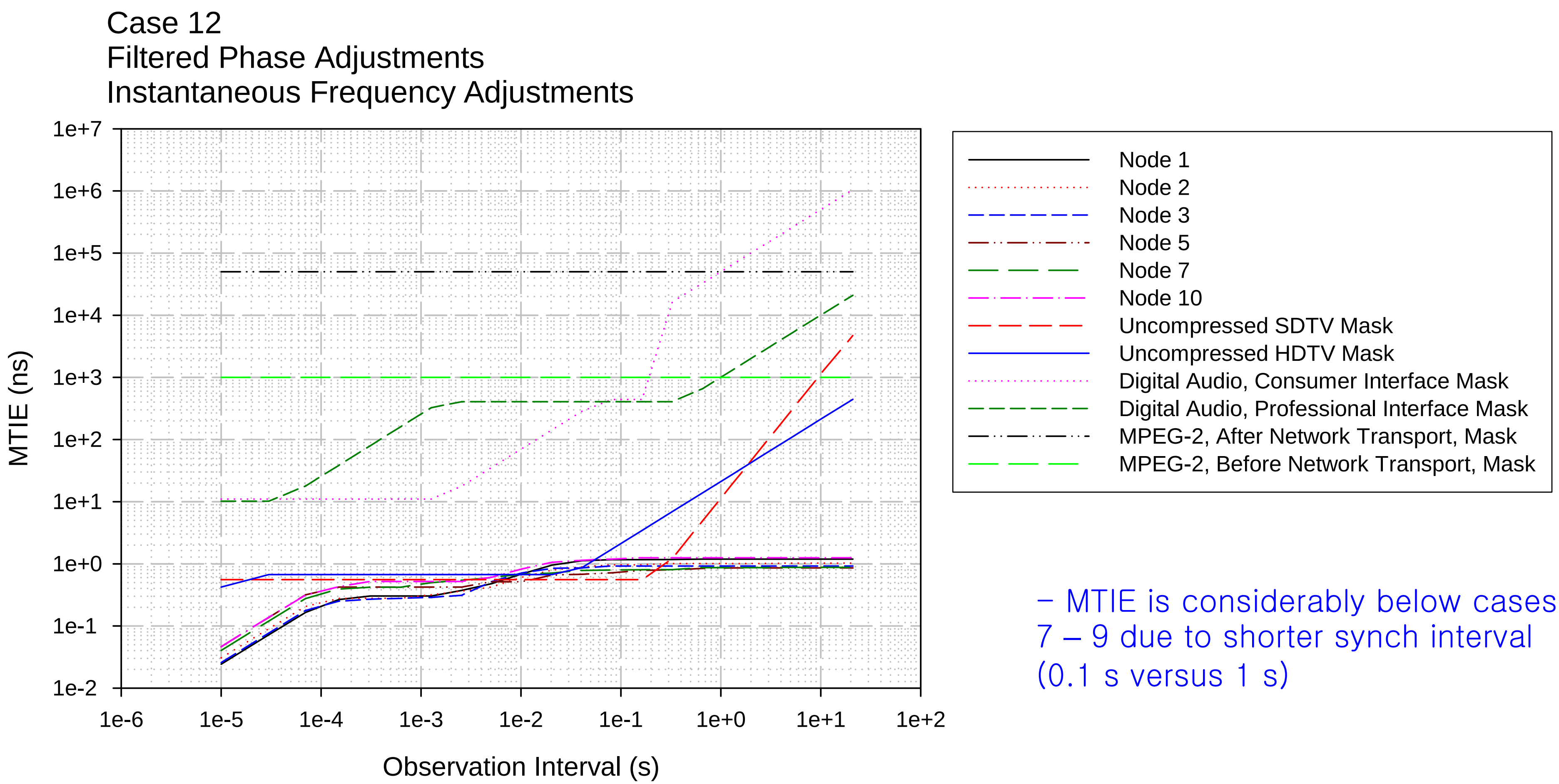
Case 12, Node 1
Filtered Phase Adjustments, Instantaneous Frequency Adjustments
(plot begins after initial transient has decayed)



Case12, Node 10
Filtered Phase Adjustments, Instantaneous Frequency Adjustments
(plot begins after initial transient has decayed)

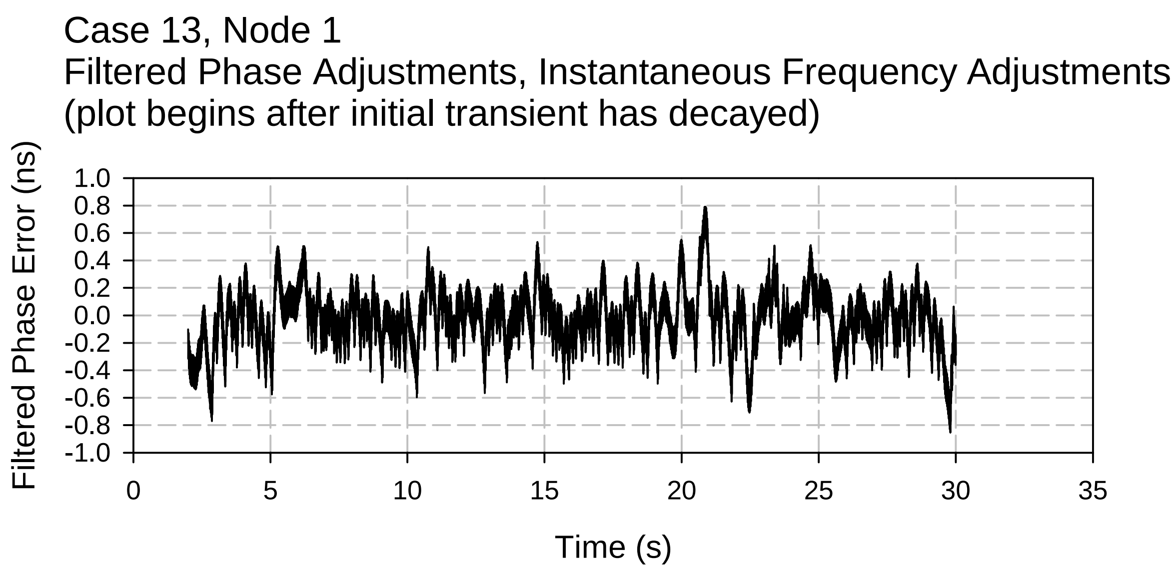


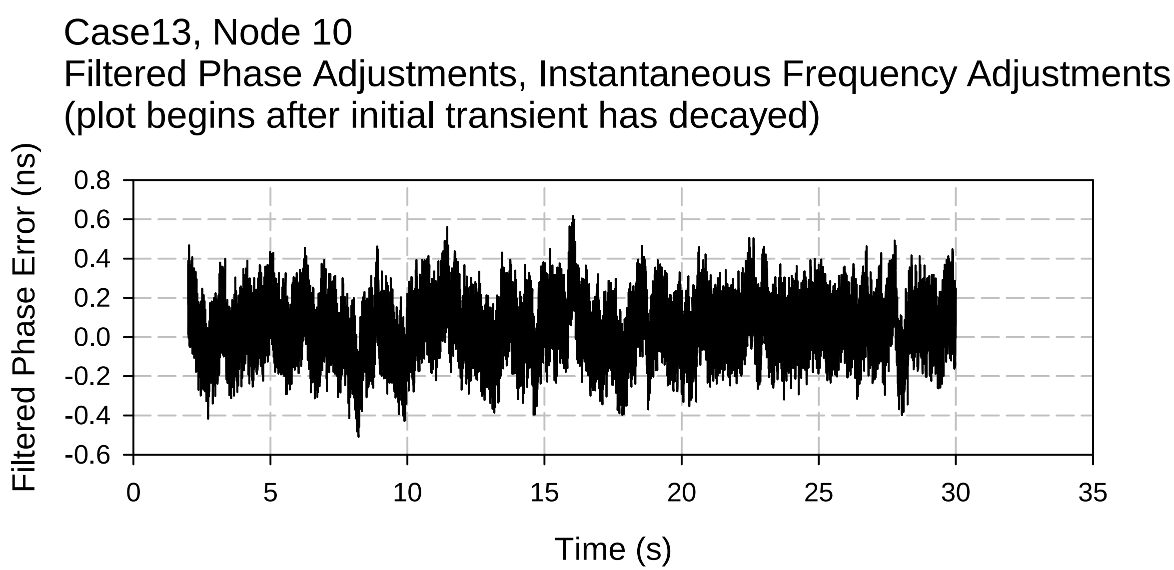
Simulation Case 12 - MTIE



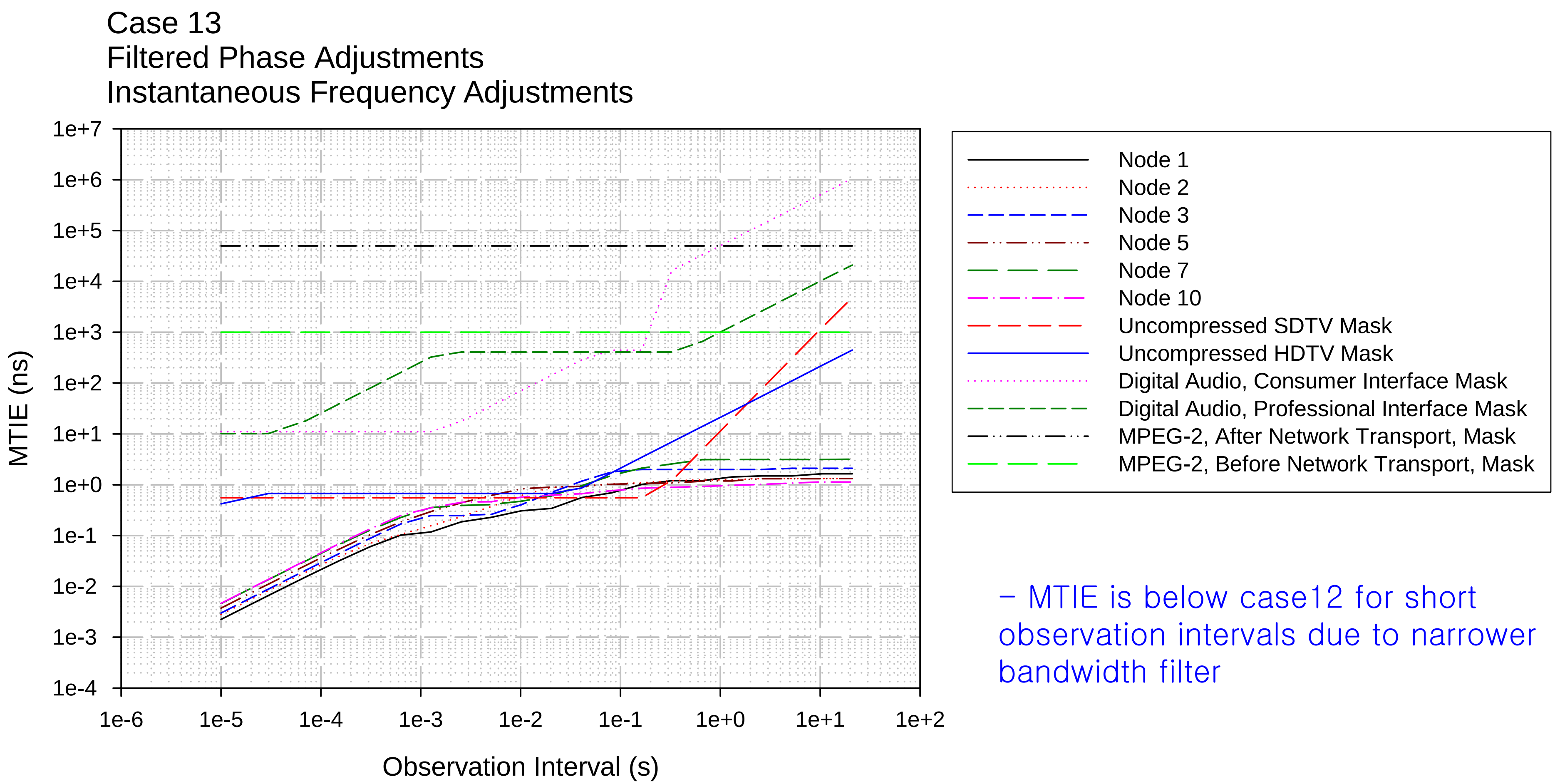
– MTIE is considerably below cases 7 – 9 due to shorter synch interval (0.1 s versus 1 s)

Simulation Case 13 – Phase Error





Simulation Case 13 - MTIE



- MTIE is below case12 for short observation intervals due to narrower bandwidth filter

Conclusions

- All simulation cases except case 11 meet the MTIE requirements for digital audio and compressed digital video
  - Case 11 exceeds the digital audio consumer interface MTIE mask
  - Cases 5 and 6 (10 ms synch interval, 100 ms frequency adjustment interval, 40 ns phase measurement granularity) come somewhat close to the digital audio consumer interface MTIE mask
- None of the simulation cases meet the uncompressed digital video MTIE masks
  - Cases 1, 12, and 13 come close

Conclusions (Cont.)

- To meet the uncompressed digital video MTIE masks with 40 ns phase measurement granularity, will likely need at least
  - 0.1 ms synch interval with 10 Hz filter, or
  - 1 ms synch interval with 1 Hz filter, or
  - Some combination in between
  - Alternatively, a different approach/method might be more effective in meeting the MTIE masks
- The above results do not include the effect of time stamp measurement errors
  - When these are included, the MTIE results will increase
    - Controlling MTIE will require shorter synch interval, narrower bandwidth filter, better phase measurement granularity, or some combination of these items
    - Alternatively, a different approach/method might be more effective in meeting the MTIE masks

Future Work

- Modeling of time stamp measurement error for various measurement points (e.g., at the MAC, between the PHY and the MAC, etc.)
- Analysis of other variations/choices (slide 8)
  - Variations/choices on slide 8 [13]
  - Method described in [14]
- Determination of statistical confidence intervals for MTIE (and possibly TDEV) by running multiple, independent replications of a simulation case)
- If warranted (i.e., if the effect is significant compared to other effects), more accurate modeling of clock noise generation

References

1. Geoffrey M. Garner and Kees den Hollander, *Analysis of Clock Synchronization Approaches for Residential Ethernet*, Samsung presentation at September, 2005 Joint IEEE 802.1/802.3 ResE SG meeting, San Jose, CA, September 29, 2005. Available via http://www.ieee802.org/3/re_study/public/index.html.
2. Geoffrey M. Garner and Kees den Hollander, *Analysis of Clock Synchronization Approaches for Residential Ethernet*, 2005 IEEE 1588 Conference, Winterthur, Switzerland, October 10 – 12, 2005. Paper and VGs available via http://www.ieee802.org/3/re_study/public/index.html.
3. Geoffrey M. Garner, *End-to-End Jitter and Wander Requirements for ResE Applications*, Samsung presentation at May, 2005 IEEE 802.3 ResE SG meeting, Austin, TX, May 16, 2005. Available via http://www.ieee802.org/3/re_study/public/index.html.
4. *Residential Ethernet (RE) (a working paper)*, Draft 0.136, maintained by David V. James and based on work by him and other contributors, August 10, 2005. Available via http://www.ieee802.org/3/re_study/public/index.html
5. ITU-T Recommendation G.810, *Definitions and Terminology for Synchronization Networks*, ITU-T, Geneva, August, 1996, Corrigendum 1, November, 2001.
6. *Phase Noise*, Vectron International, Application Note, available at http://www.vectron.com.
7. *Jitter and Signal Noise in Frequency Sources*, Raltron, Application Note, available at http://www.raltron.com/

References

8. David W. Allan, Marc A. Weiss, and James L. Jespersen, *A Frequency Domain View of Time Domain Characterization of Clocks and Time and Frequency Distribution Systems*, Forty-Fifth Annual Symposium on Frequency Control, Los Angeles, CA, May 29 – 31, 1991, pp. 667 – 678.
9. Stefano Bregni, *Synchronization of Digital Telecommunications Networks*, Wiley, 2002.
10. 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.
11. 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.
12. 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.
13. Geoffrey M. Garner, *Summary of Potential Choices for Synchronization Transport in Residential Ethernet*, Samsung presentation at July, 2005 IEEE 802.3 ResE SG meeting, San Francisco, CA, July 18, 2005. Available via http://www.ieee802.org/3/re_study/public/index.html.
14. Sihai Wang, *Improvements to Boundary Clock Based Time Synchronization Through Cascaded Switches*, Samsung presentation at November, 2005 IEEE 802.3 ResE SG meeting, Vancouver, BC, November 14, 2005. Available via http://www.ieee802.org/3/re_study/public/index.html.

Appendix I – Clock Noise Model

- Clock phase noise may be modeled as a sum of random processes with power spectral density (PSD) of the form $Af^{-\alpha}$
  - In practice, the PSD has 3 terms (see [6] and [7])
    - α = 0, White Phase Modulation (WPM)
    - α = 1, Flicker Phase Modulation (FPM)
    - α = 3, Flicker Frequency Modulation (FFM)
  - Can write the PSD, $S_x(f)$ as

$$S_x(f) = \frac{A}{f^3} + \frac{B}{f} + C, \text{ where } S_x(f) \text{ has units of } \mathrm{ns}^2/\mathrm{Hz}$$

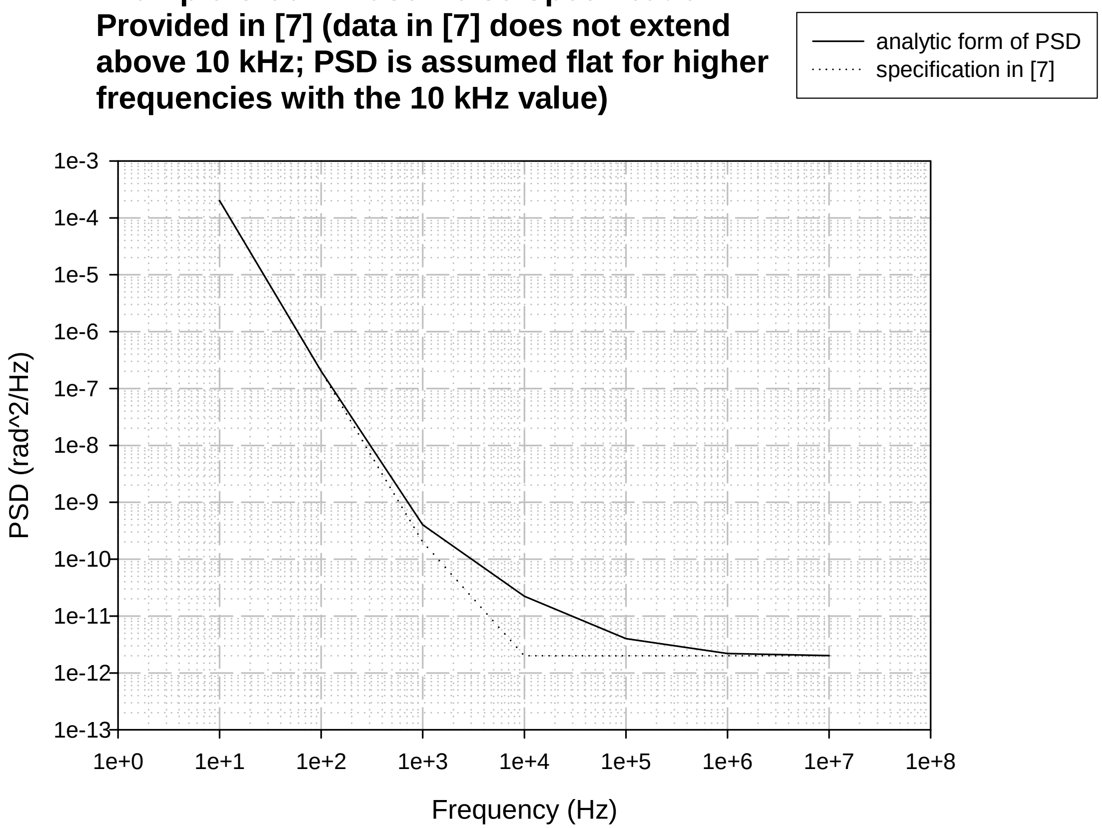
  - Often express as

$$S_\phi(f) = \left(2\pi\nu_0\right)^2 S_x(f), \text{ where units of } S_\phi(f) \text{ are } \mathrm{rad}^2/\mathrm{Hz}$$

  - An example PSD specification is given in Figure 12 of [7], and reproduced on the next slide
    - Data in [7] is given in dBc/Hz; data has been converted to rad^2/Hz
    - Data in [7] is given only for frequencies below 10 kHz; here, we assume the PSD is flat above 10 kHz
    - Dotted curve on the next slide is the converted data of [7]; solid line is a conservative fit of the above power law sum
  - The specifications for the individual products of [6] and [7] are below this example, at least for those products where phase noise specifications are provided

Appendix I – Clock Noise Model

Example Clock Phase Noise Specification Provided in [7] (data in [7] does not extend above 10 kHz; PSD is assumed flat for higher frequencies with the 10 kHz value)



Note: Data in [7] is given in dBc/Hz; data has been converted to rad^2/Hz

Appendix I – Clock Noise Model

- Another measure for clock noise, which is more convenient because it is a time domain parameter, is Time Variance (TVAR)
  - Time Deviation (TDEV) is the square root of TVAR
- TVAR is 1/6 times the expectation of the square of the second difference of the phase error averaged over an interval

$$\mathrm{TVAR}(\tau) = \frac{1}{6} E\left[\left(\Delta^2 \bar{x}\right)^2\right]$$

where $E[\cdot]$ denotes expectation,

$\bar{x}$ denotes average over the integration time τ,

and Δ^2 denotes second difference

- TVAR may be estimated from measured or simulated data using [5]

$$\mathrm{TVAR}(n\tau_0) = \frac{1}{6n^2(N-3n+1)} \sum_{j=1}^{N-3n+1} \left[\sum_{i=j}^{n+j-1} \left(x_{i+2n} - 2x_{i+n} + x_i \right) \right]^2, \quad \mathrm{n} = 1, 2, \ldots, \text{integer part}(N/3)$$

where τ_0 is the sampling interval and $\tau = N\tau_0$

Appendix I – Clock Noise Model

- TVAR is equal to $\tau^2/3$ multiplied by the Modified Allan Variance
- For power-law noises with PSD proportional to $f^{-\alpha}$, TVAR is proportional to τ^β, where $\beta = \alpha - 1$
- The magnitude of TVAR may be related to the magnitude of PSD for power-law noises; see [8] and [9] for details
  - FFM

$$S_x(f) = \frac{A}{f^3} \qquad \mathrm{TVAR}(\tau) = \frac{(2\pi)^2 9 \ln 2}{20} A\tau^2$$

  - FPM (result is from [8]; a more exact expression is given in [9])

$$S_x(f) = \frac{B}{f} \qquad \mathrm{TVAR}(\tau) = \frac{3.37}{3} B$$

  - WPM

$$S_x(f) = C \qquad \mathrm{TVAR}(\tau) = \frac{\tau_0 f_h}{\tau} C$$

f_h = noise bandwidth

Appendix I – Clock Noise Model

- TVAR and TDEV (or Allan Variance or Modified Allan Variance) are used to characterized phase noise in oscillators rather than classical variance
  - The time-domain estimator for classical variance diverges for some power-law noise processes
  - The time-domain estimators for TVAR, Allan Variance, and Modified Allan Variance converge for all power-law noise processes

Appendix I – Clock Noise Model

❑Simulation of WPM

- WPM is simulated as a sequence of independent, identically distributed random samples
- Noise distribution is taken as Gaussian with zero mean
- Variance and sampling time determine TDEV level
  - Choose variance such that, with given sampling time, the computed TDEV from a sample history is close to value obtained from above relation between TDEV and PSD
    - Assume noise bandwidth is equal to line rate (100 MHz)

❑Simulation of FPM

- FPM is simulated by passing a sequence of independent, identically distributed random samples through a Barnes/Jarvis filter [10] – [12]
  - 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

Appendix I – Clock Noise Model

Simulation of FPM (Cont.)

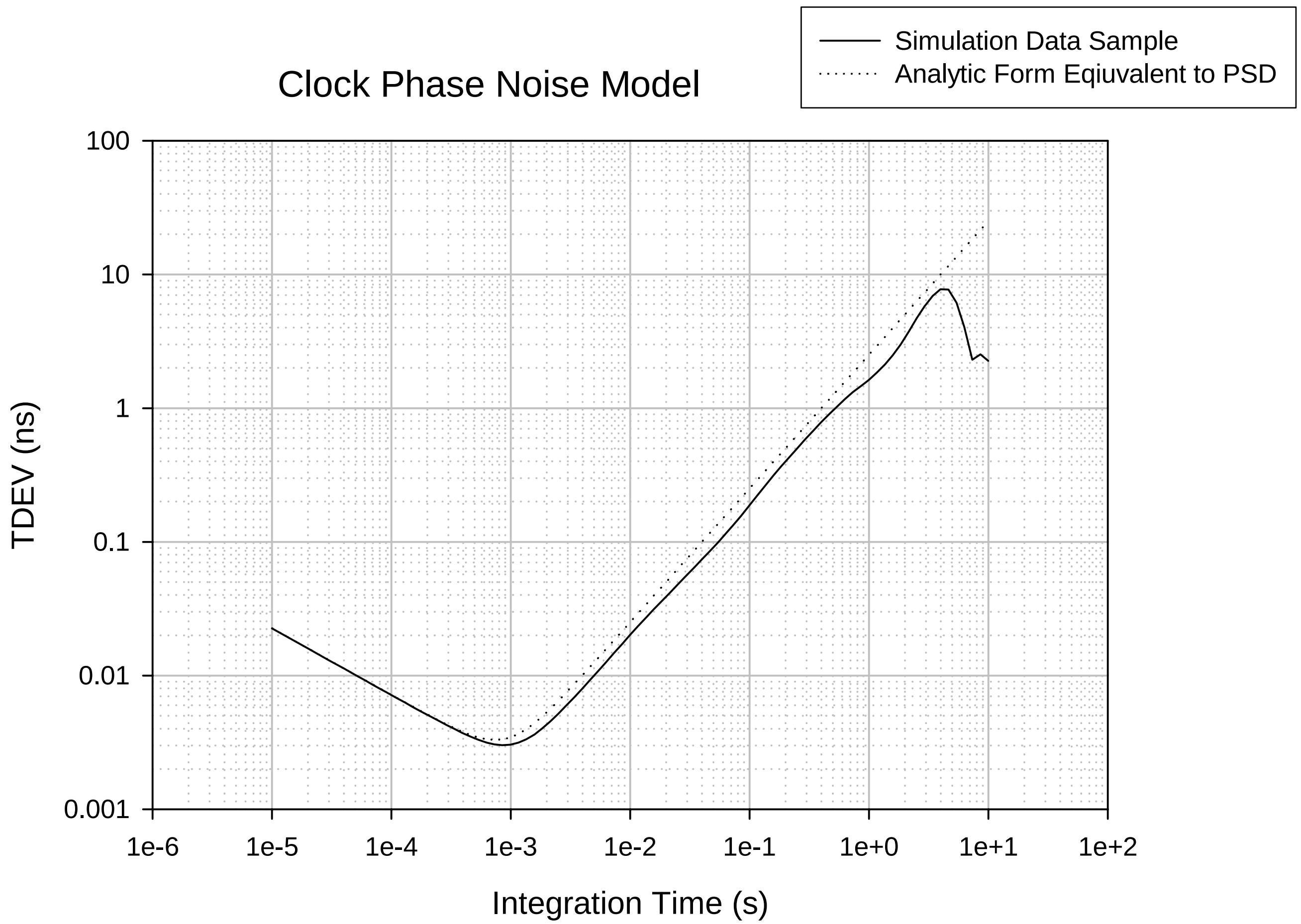
- Noise distribution is taken as Gaussian with zero mean
- Variance determines TDEV level
  - Choose variance such that the computed TDEV from a sample history is close to value obtained from above relation between TDEV and PSD

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

Next slide shows TDEV for simulated data sample (10^{-5} s time step) and analytic form equivalent to PSD (solid curve on slide 48)

Appendix I – Clock Noise Model



Appendix II – Definition of MTIE

- Jitter and wander requirements can be expressed in terms of Maximum Time Interval Error (MTIE) masks
- MTIE is peak-to-peak phase variation for a specified observation interval, expressed as a function of the observation interval
  - An estimate of MTIE may be computed by (see [5])

$$\mathrm{MTIE}(n\tau_0) \cong \max_{1\le k\le N-n}\left(\max_{k\le i\le k+n} x(i) - \min_{k\le i\le k+n} x(i)\right), \quad n = 1, 2, \ldots, N-1$$

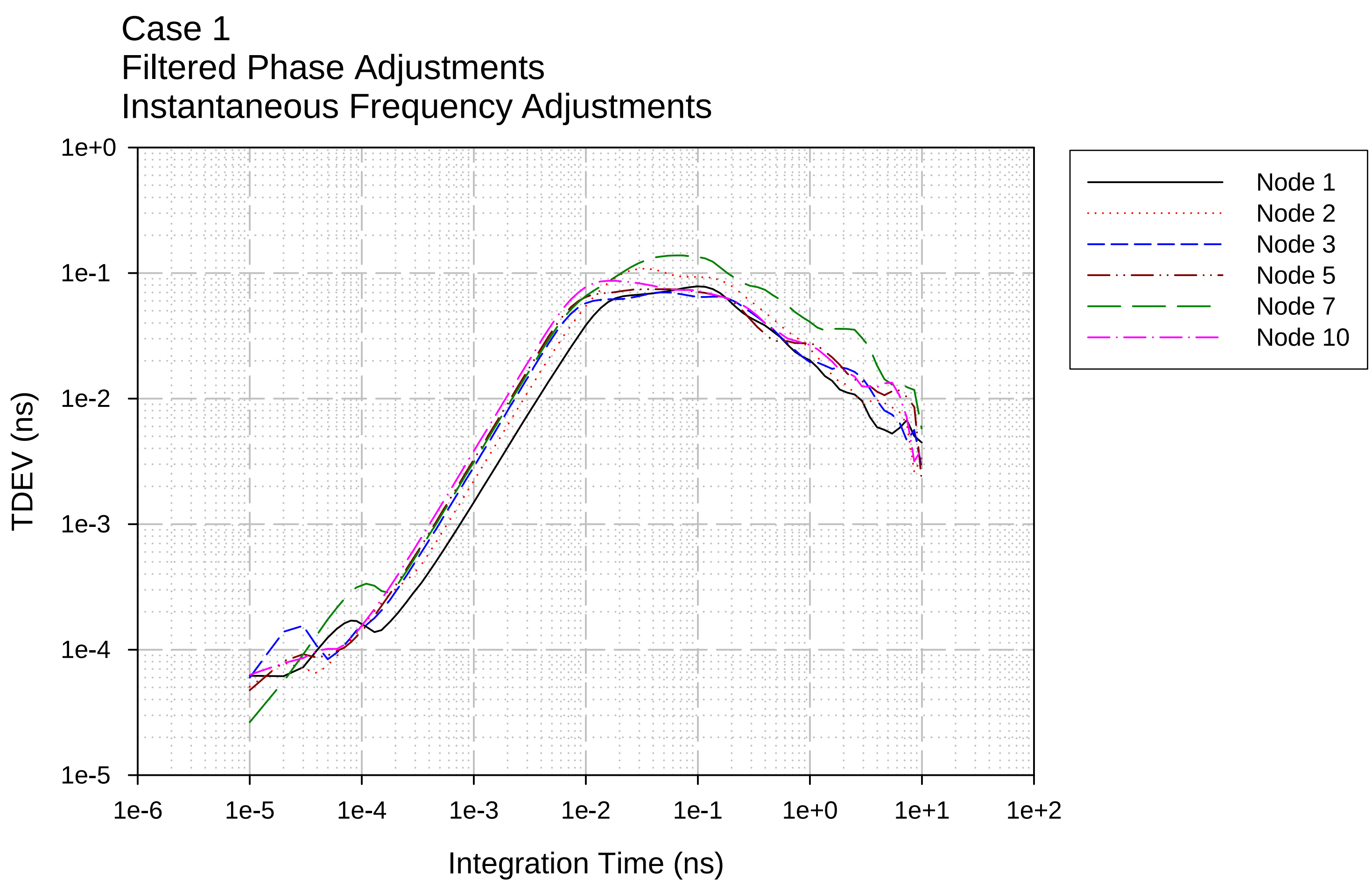
where τ_0 is the sampling interval, $n\tau_0$ is the observation interval, $x(i)$ is the i^{th} phase sample, and N is the number of phase samples ($N\tau_0$ is the measurement interval)

- The derivation of the MTIE masks on slide 6 from the jitter and wander requirements is given in [3]

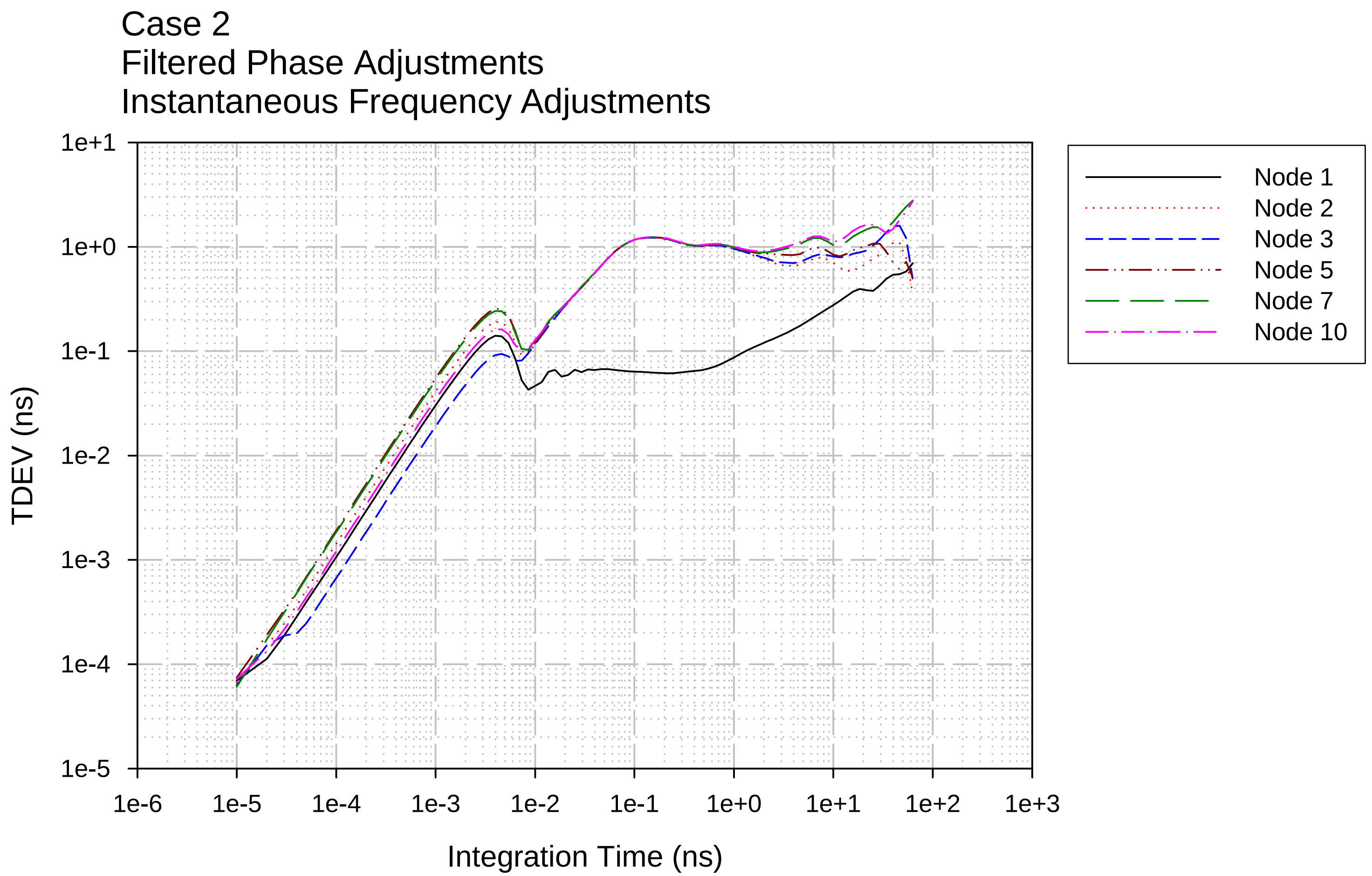
Appendix III – TDEV Results for Simulation Cases

- The following slides give results for TDEV for Simulation Cases 1 – 13
- Results for TDEV rather than classical standard deviation are presented because the classical standard deviation estimator diverges for certain phase noise types, while the TDEV estimator converges

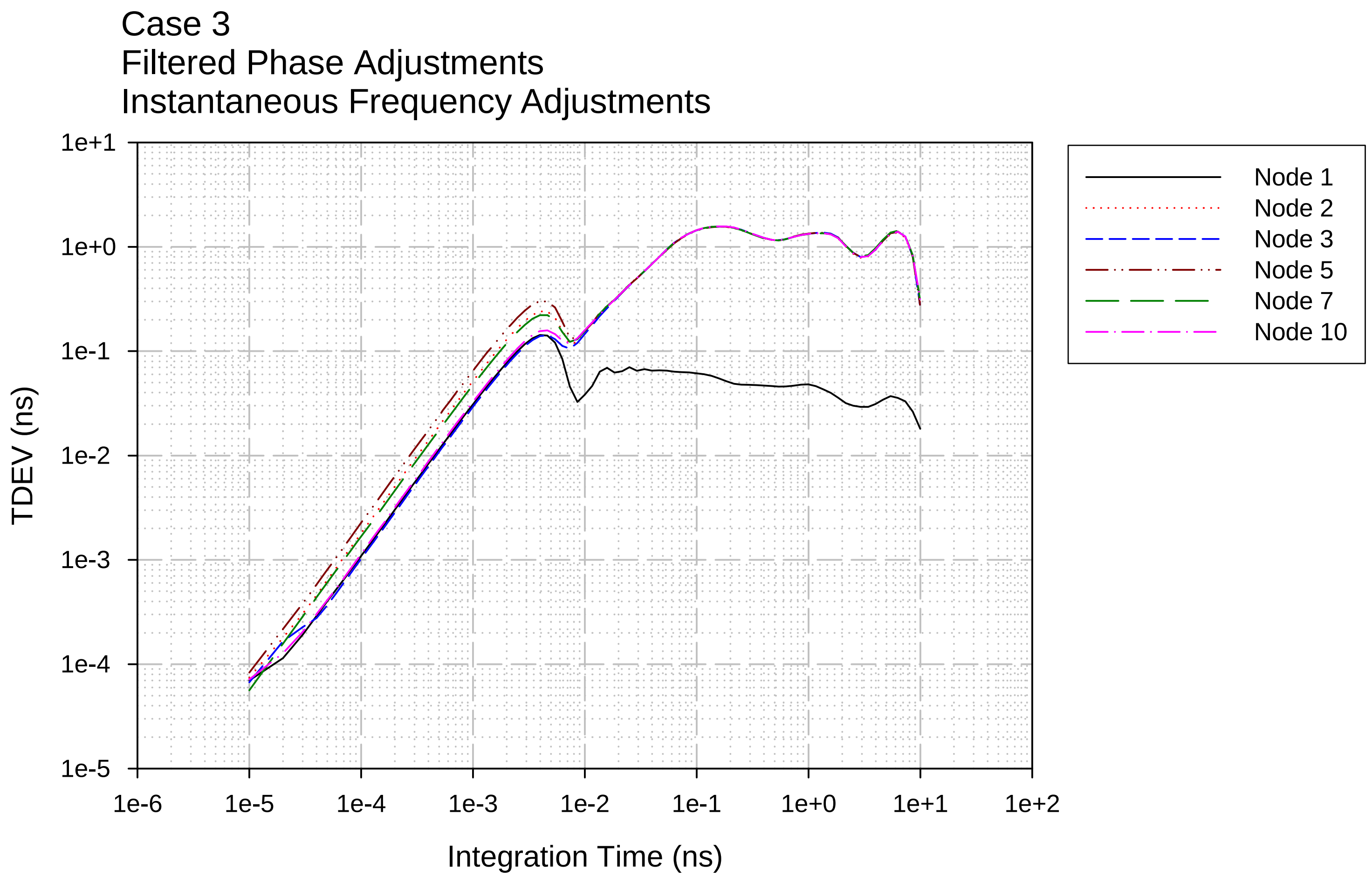
Simulation Case 1 - TDEV



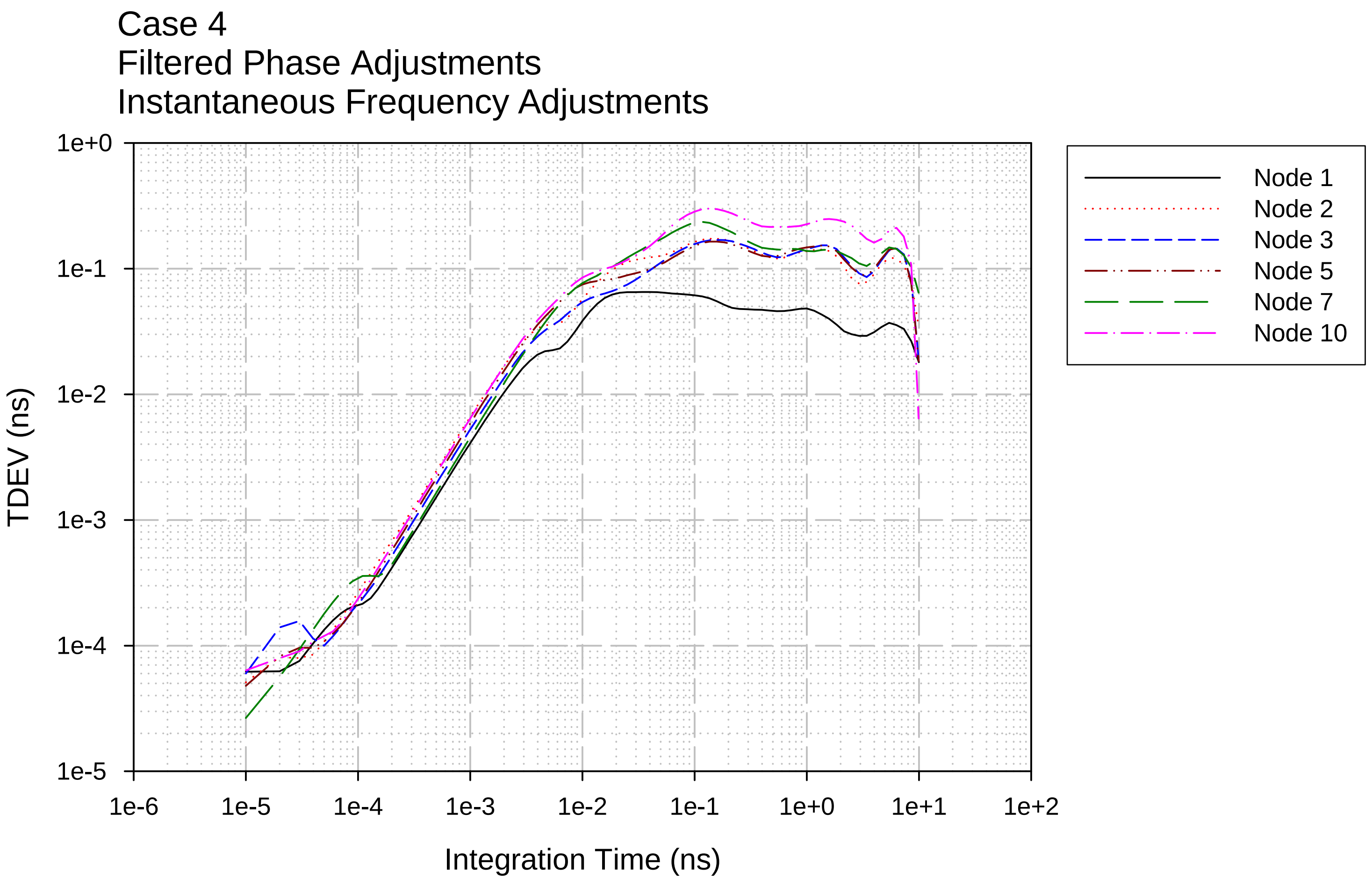
Simulation Case 2 - TDEV



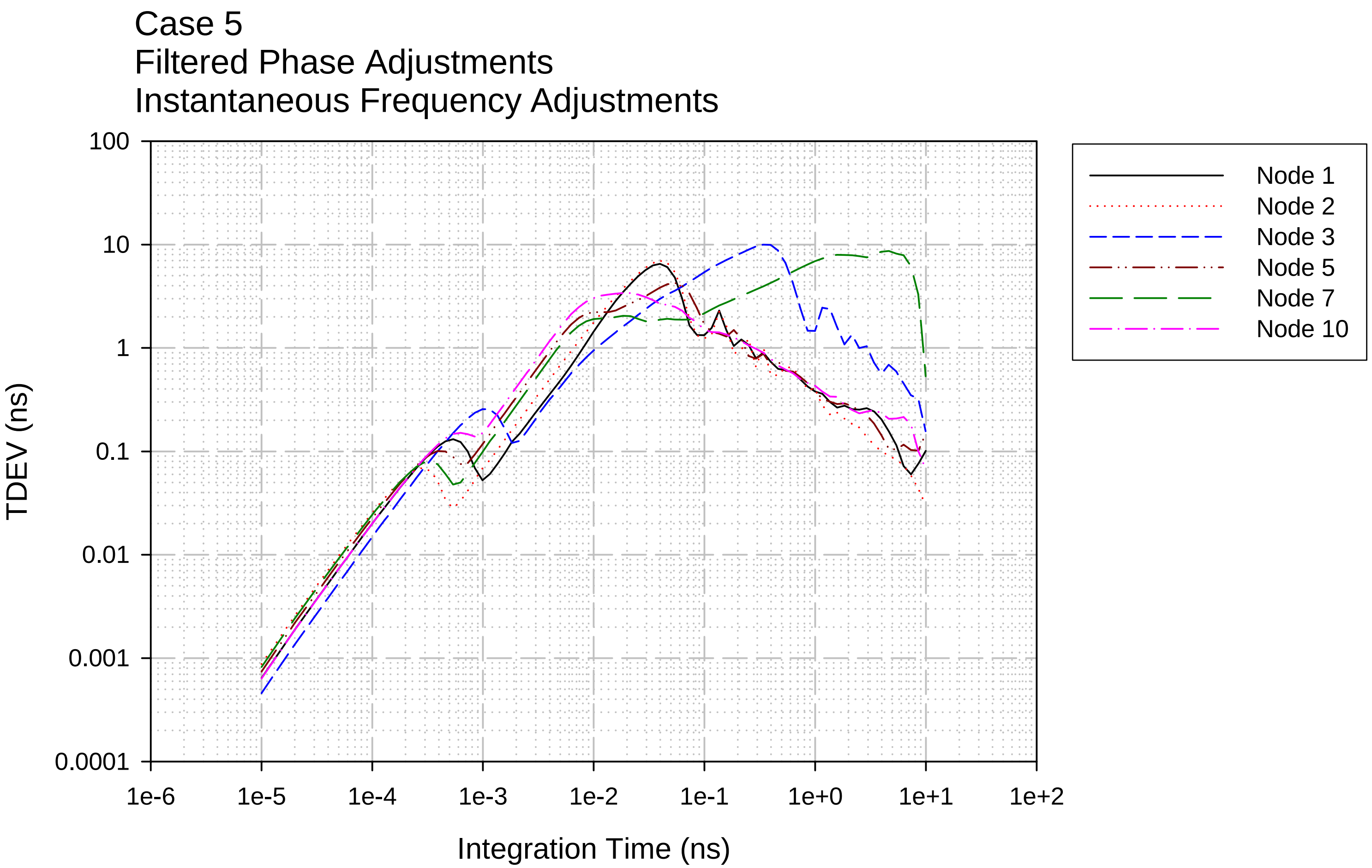
Simulation Case 3 - TDEV



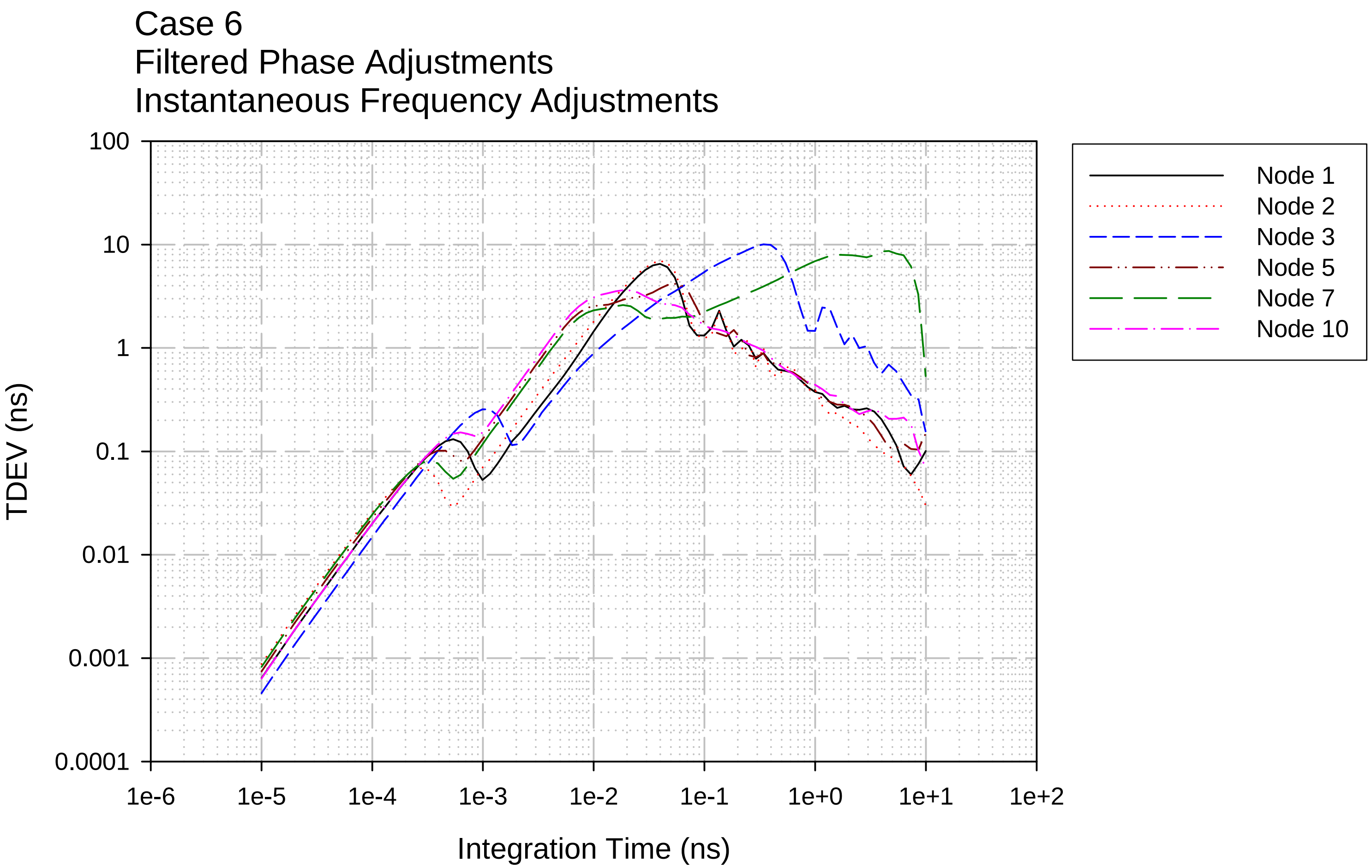
Simulation Case 4 – TDEV



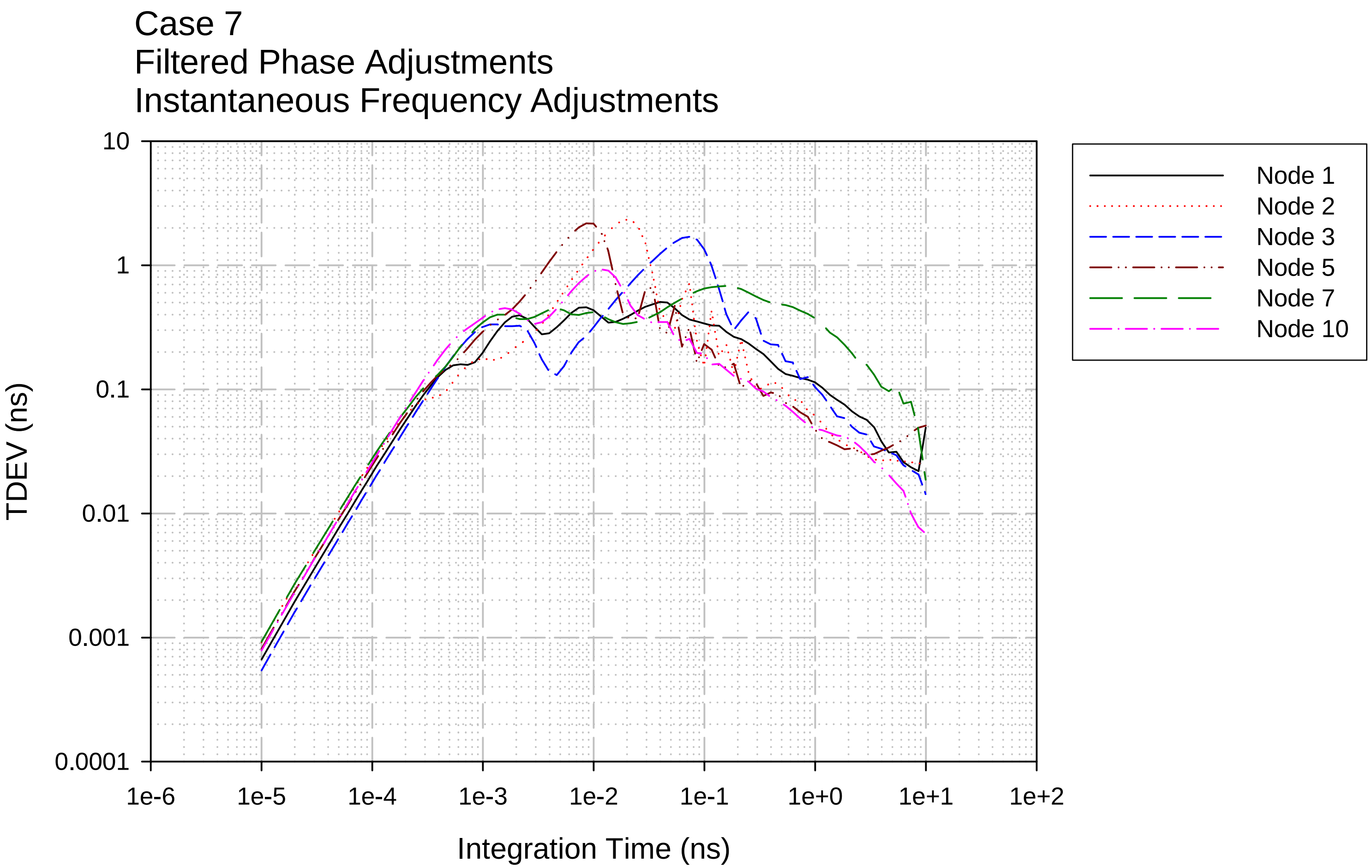
Simulation Case 5 – TDEV



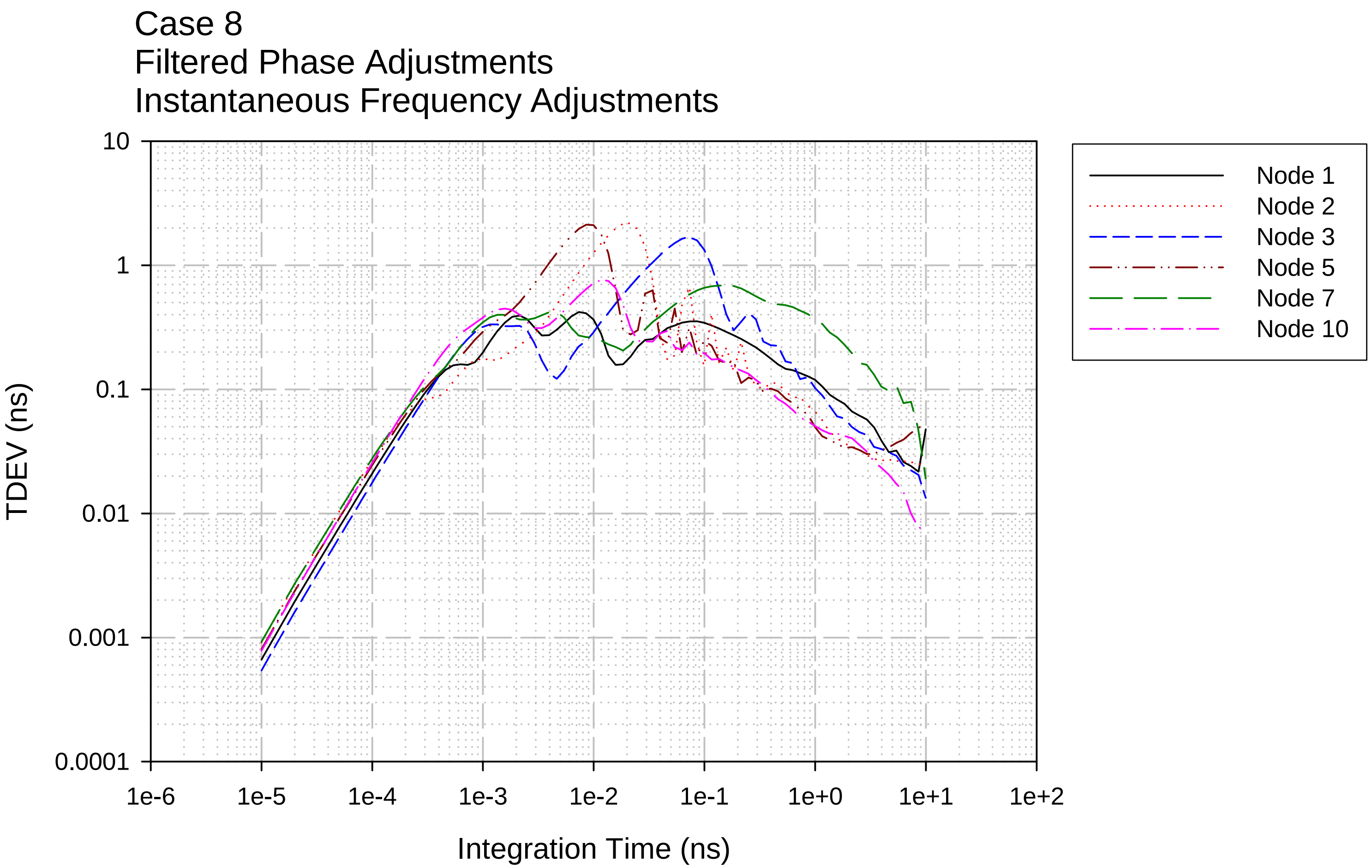
Simulation Case 6 – TDEV



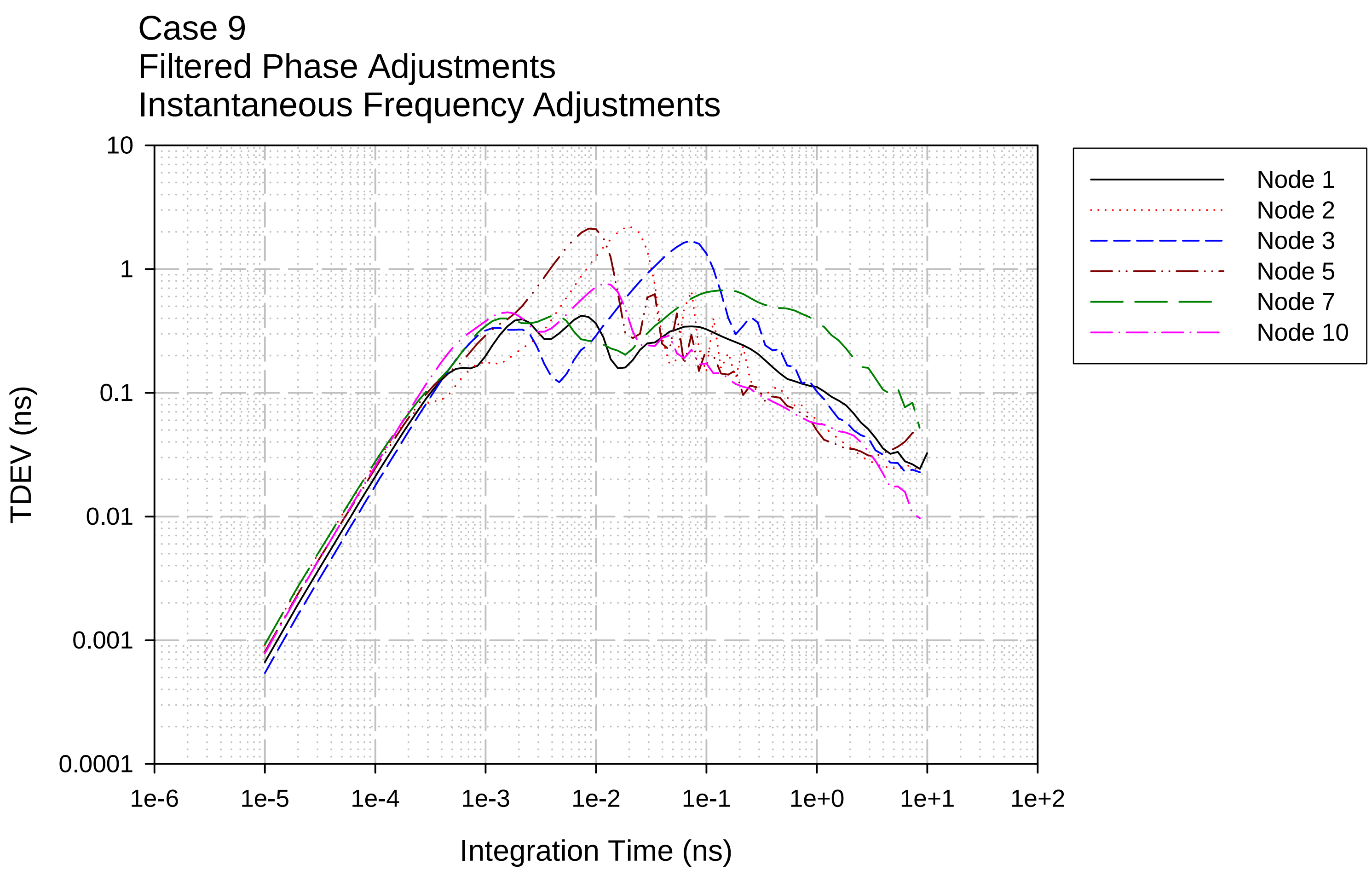
Simulation Case 7 – TDEV



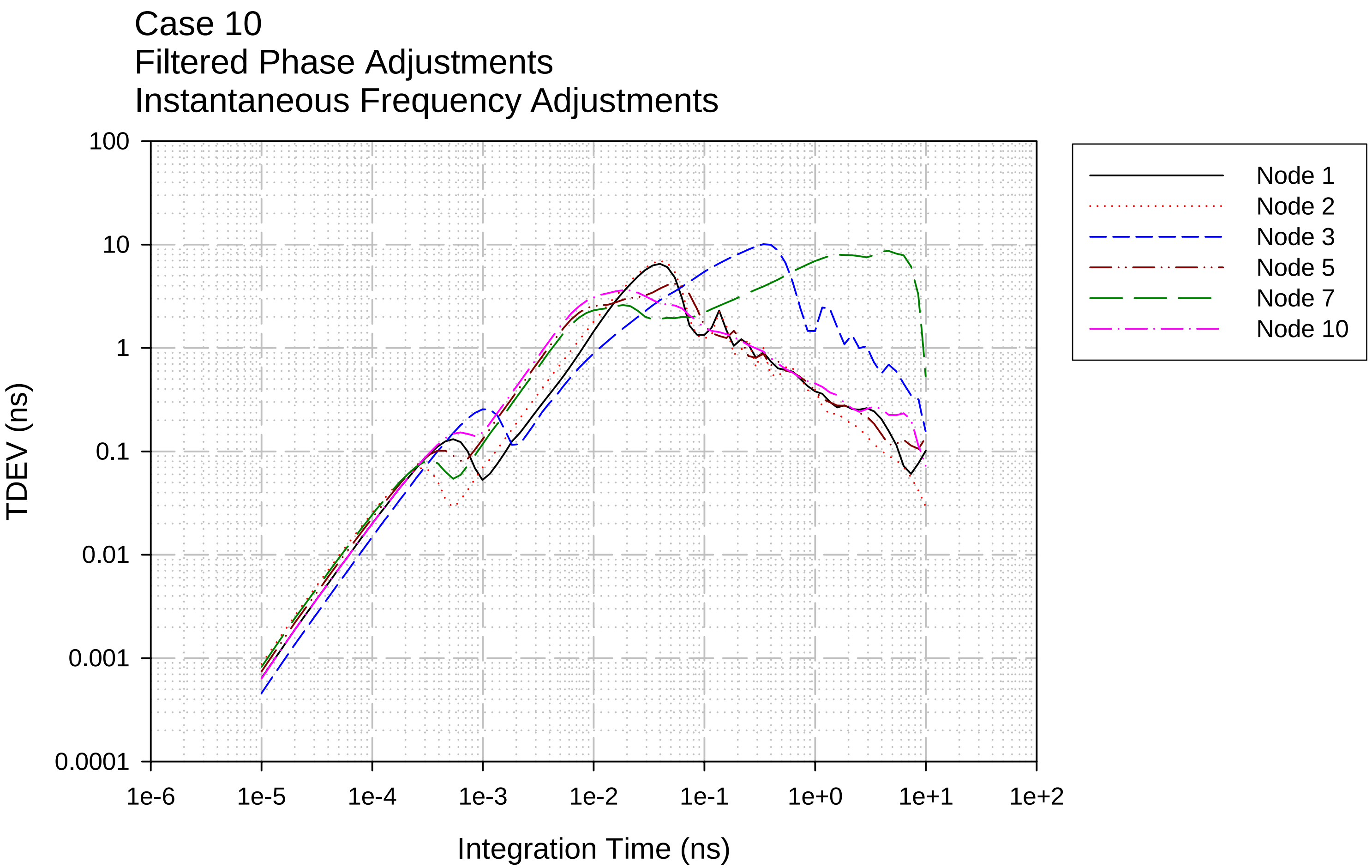
Simulation Case 8 – TDEV



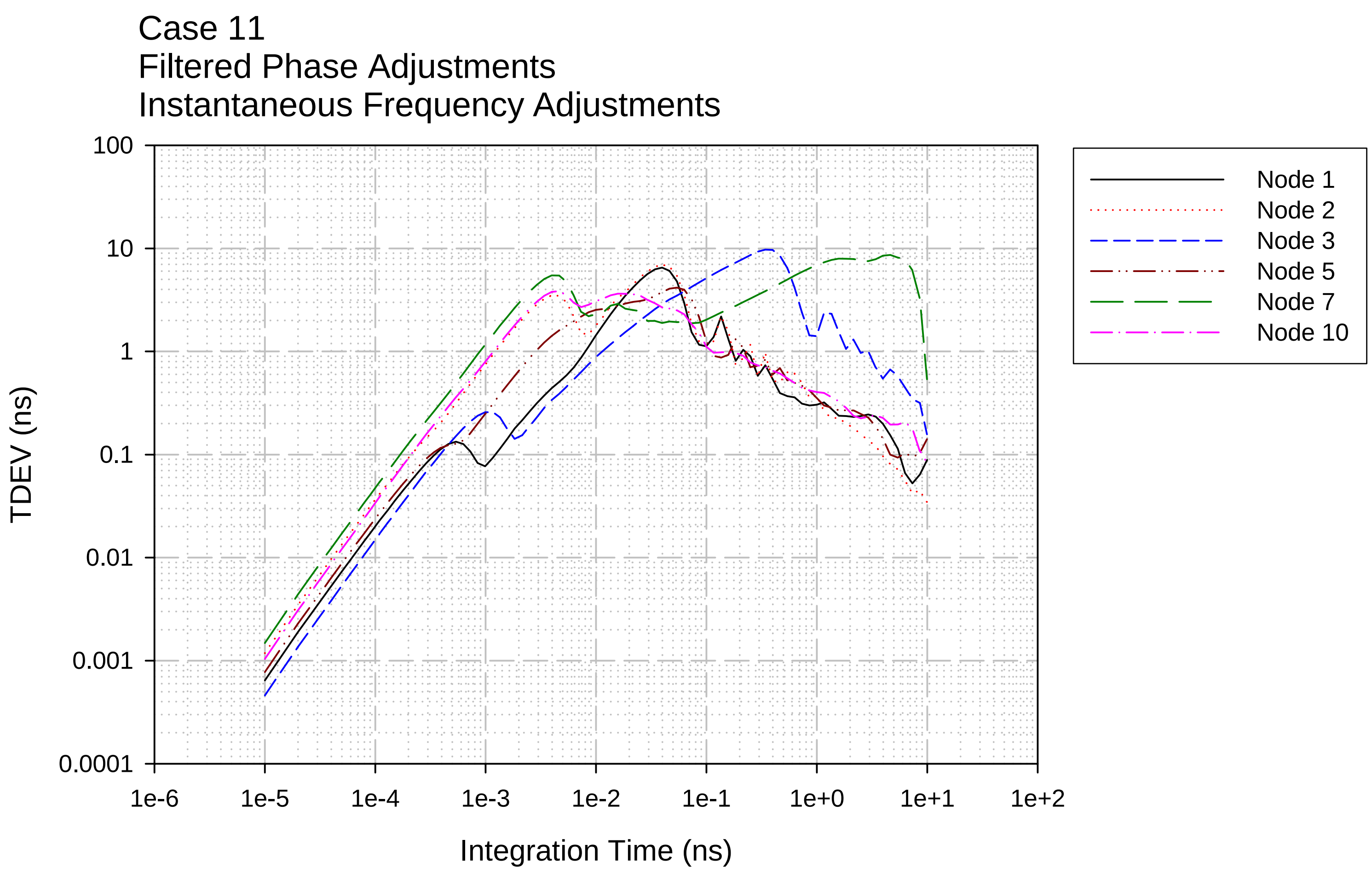
Simulation Case 9 - TDEV



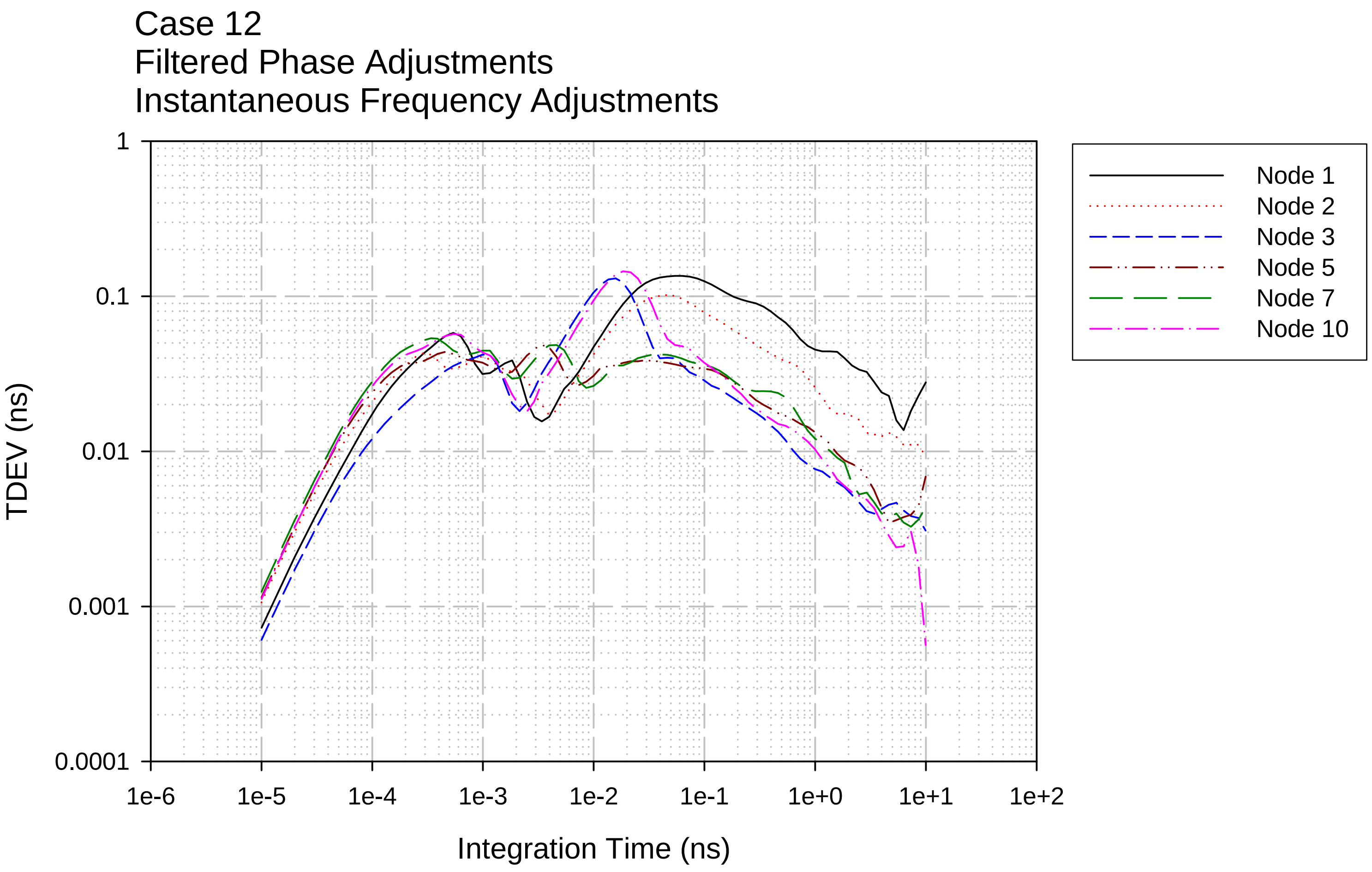
Simulation Case 10 - TDEV



Simulation Case 11 - TDEV



Simulation Case 12 - TDEV



Simulation Case 13 - TDEV