IEEE P802.1DP / SAE AS6675
IEEE 802.1 Sept 2023 Interim Session

Default Algorithm(s) for the Fault Tolerant Module

Andrew Loveless [NASA-JSC], Brendan Hall [NASA-JSC (Jacobs)], Richard Tse [Microchip], Rob Donnelly [Jet Propulsion Laboratory, California Institute of Technology], Paul Miner [NASA-LaRC]

Fault Tolerant Module (FTM) for P802.1DP

The goal of this contribution is to fulfill this note.

7.1.4 Fault Tolerance Module

A fault tolerance module (FTM) operating at the application layer according to IEEE Std. 802.1AS-2020, Clause 9, is specified for aerospace applications to be implemented in all time-aware Bridges or end stations that support multiple time domains. The FTM manages the selection of a clock source from amongst two or more PTP domains (and PTP instances) to support increased availability and integrity as compared to single domain solutions. Figure 7-1 illustrates the fault-tolerant model operating with three PTP instances. The FTM module could use the local clock as an input in the selection algorithm. A default selection algorithm is defined in this standard.

<< Editor's Note: The default selection algorithm and further details of the fault-tolerant module will be added here as the committee develops it further>>

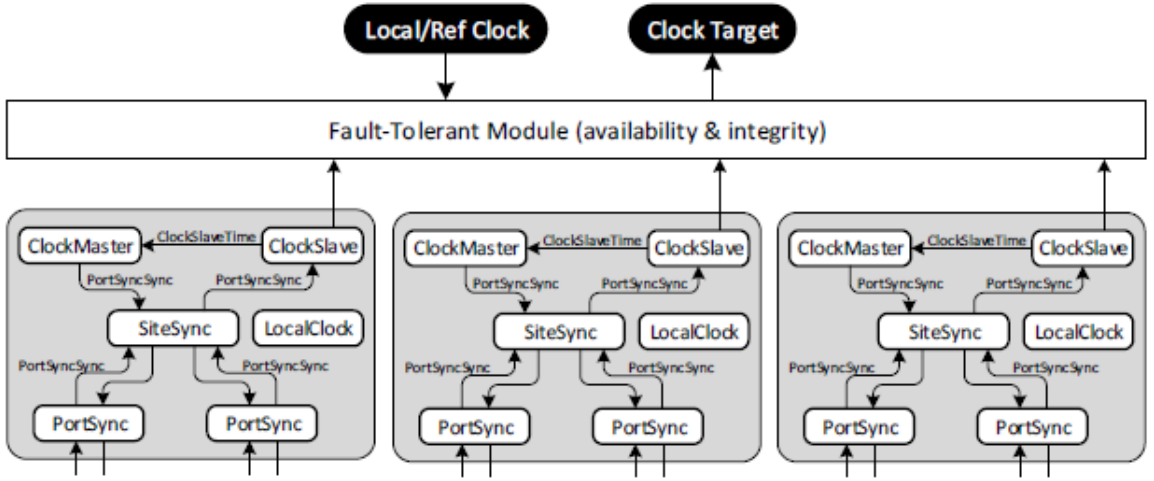


Figure 7-1—Aerospace Fault-Tolerant Module

Desirements, Properties & Basic Assumptions

- Desirements
  - Open and works with current COTS hardware and software stacks
  - Scalable and supports different degrees of fault tolerance and graceful degradation
  - Network topology agnostic
  - Conceptually simple with
    - Minimal impact on existing COTS software and hardware
    - Minimal additional hardware overhead
      - Not require self-checking inter-stages**
    - Minimal software support for PTP Relay Instances
- Basic Assumptions
  - All non-faulty Grandmaster PTP Instance (GM) clocks are agreed within a bounded accuracy and precision
  - End-station and bridge hardware may fail-arbitrarily
    - May manifest consistent & inconsistent value and temporal signal corruption
- Properties
  - Time at all end-stations and bridges will be within a bounded offset of a non-faulty GM's time

** Although additional robustness may be gained with more relaxed fault models justified by such hardware

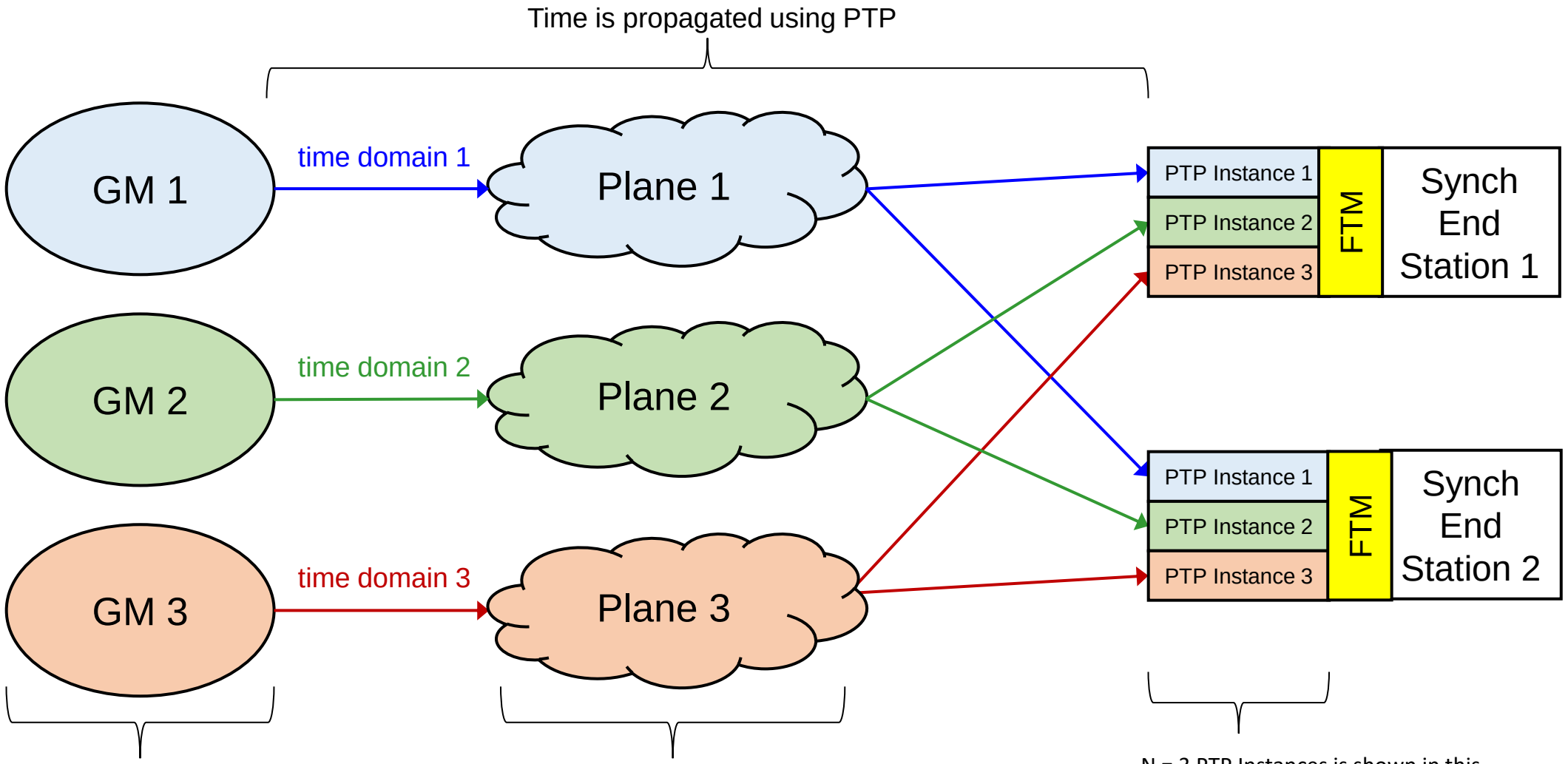
Focus and Scope

- Time Synchronization for P802.1DP requires:
  - Agreement on time generation among all GMs
    - **Out of scope** for this contribution and, likely, of P802.1DP

      Note: The eventual solution will need to address system start up, and potential GM-clique resolution. This is considered to be future work, perhaps for an informative annex of P802.1DP.
  - Propagation of agreed time to all synchronous end stations and bridges
    - **In scope** of this contribution
    - Under justifiable failure assumptions local masking and selection is sufficient to support bounded accuracy of sync-clients and time-aware shaping bridges

- Clock diagnosis is out of scope of this contribution
  - There is no expectation or need for synchronous end stations and bridges to agree on the *best clock, nor isolate bad clocks*
    - *This would introduce a requirement for global consensus and increase the system vulnerability to Byzantine failure [1][2]*

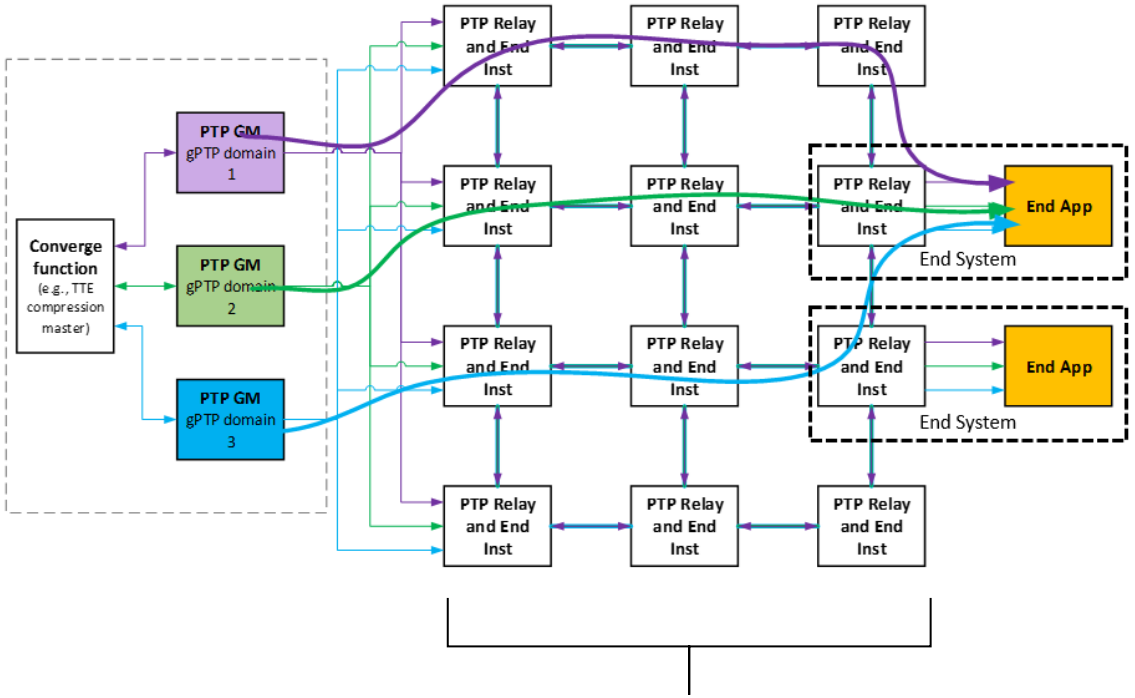
Time Propagation Model



“Planes” from Previous Contributions

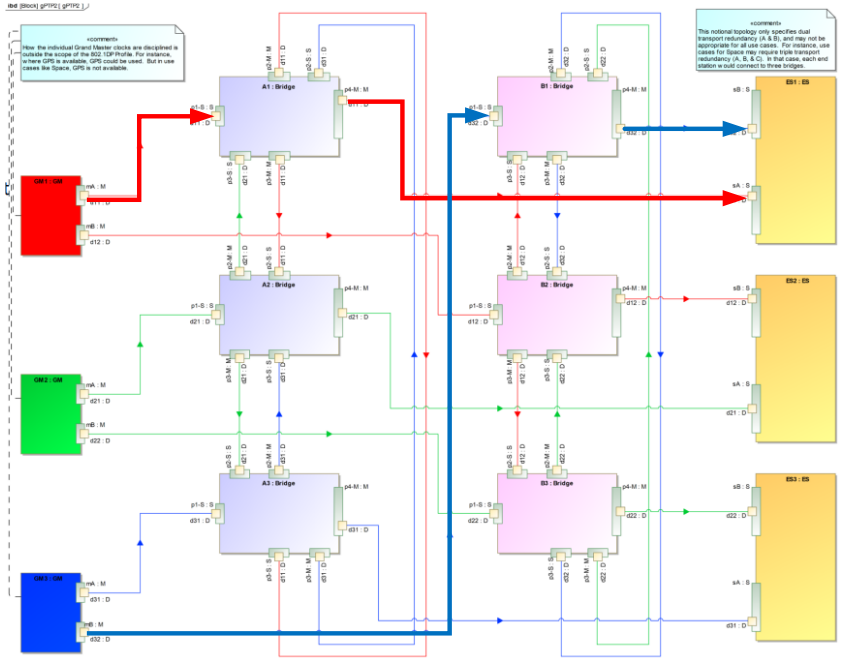
- The term “planes” is new, but the concept is not, as shown below.

- From https://www.ieee802.org/1/files/public/docs2022/dp-tse-donnelly-fault-tolerant-time-prop-1122-v06.pdf



3 planes highlighted

- From https://www.ieee802.org/1/files/public/docs2023/dp-Lie-fault-tolerant-time-sync-dual-transport-0423-v02.pdf



2 planes highlighted

Fault Hypothesis – Recap

- Plane-based concept enables treatment of each GM and its forwarding network as an independent fault containment region (FCR) and simplifies the analysis
  - The GM and the bridges within a plane could still fail independently
- Considering two failure modes:
  - **Omission** – The GM(s)/plane(s) fail to propagate time to the PTP End Instances of one or more end stations
  - **Arbitrary Temporal and/or Value Corruption** - The GM(s)/plane(s) propagate a time to one or more end stations that deviates from a non-faulty GM/plane by more than (Δ + max_accepted_propagation_skew)
    - Why? There is no distinction between non-faulty behavior and faulty behavior resulting in time propagation within (Δ + max_accepted_propagation_skew)
    - Tolerating such faults is necessary per crewed space flight standards

Note: In both cases, faults:

- May manifest symmetrically or asymmetrically (no difference for propagation)
- May be permanent, transient, or intermittent

Scope for FTM

- No attempt by FTM to globally diagnose or ignore faulty lanes
  - Why? FTM functions may not be high-integrity and may lie; would require Byzantine exchange
  - Don’t want fault FTM from affecting FTMs of other end stations
- No attempt by FTM to locally ignore previously faulty lanes in future synchronization intervals
  - Why? Allows transparent recovery if faulty lane is repaired/replaced or faults are transient
    - Downside: Comes at expense of some robustness under permanent faults

Detailed Assumptions

- N time domains (N=3 for this contribution)
  - Non-faulty GMs generate time with a max skew of Δ
  - Max non-faulty time skew between PTP End Instances = (Δ + max_accepted_propagation_skew)
- N independent planes (1 time domain per plane)
  - Each GM propagates its time to each end station/bridge via one plane
- Each synchronous end station/bridge contains:
  - N PTP End Instances to recover time for N time domains
    - The ToD for each time domain is available to the FTM (see interfaces defined in 9.1 of IEEE 802.1AS)
    - A synchronous bridge requires PTP End Instances to recover ToD for its time aware shapers
  - A local oscillator
  - A FTM
    - Produces FTM_clk, based on the ToD from its PTP End Instances and, in some scenarios, the local oscillator
      - FTM_clk = f(ToDs from PTP End Instances, local oscillator)
      - At startup, FTM_clk is undefined (e.g., arbitrarily free-running)

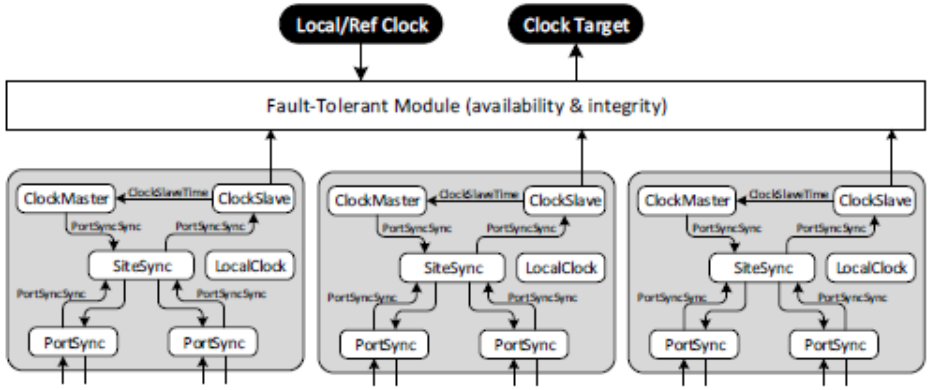


Figure 7-1—Aerospace Fault-Tolerant Module

FTM Algorithmic Choices

- The simplification of the system requirements to limit concerns to propagation of time without global best clock/worst clock consensus enables simpler algorithmic choices
- Two default algorithms are proposed:
  - Hybrid Mid-Value selection
    - The fault tolerant FTM clock is selected from the mid-value of incoming non-faulty time domain clocks
  - Closest-pair selection
    - The fault tolerant FTM clock is selected from one of the two clocks in a clock pair that is deemed to be non-faulty

FTM States

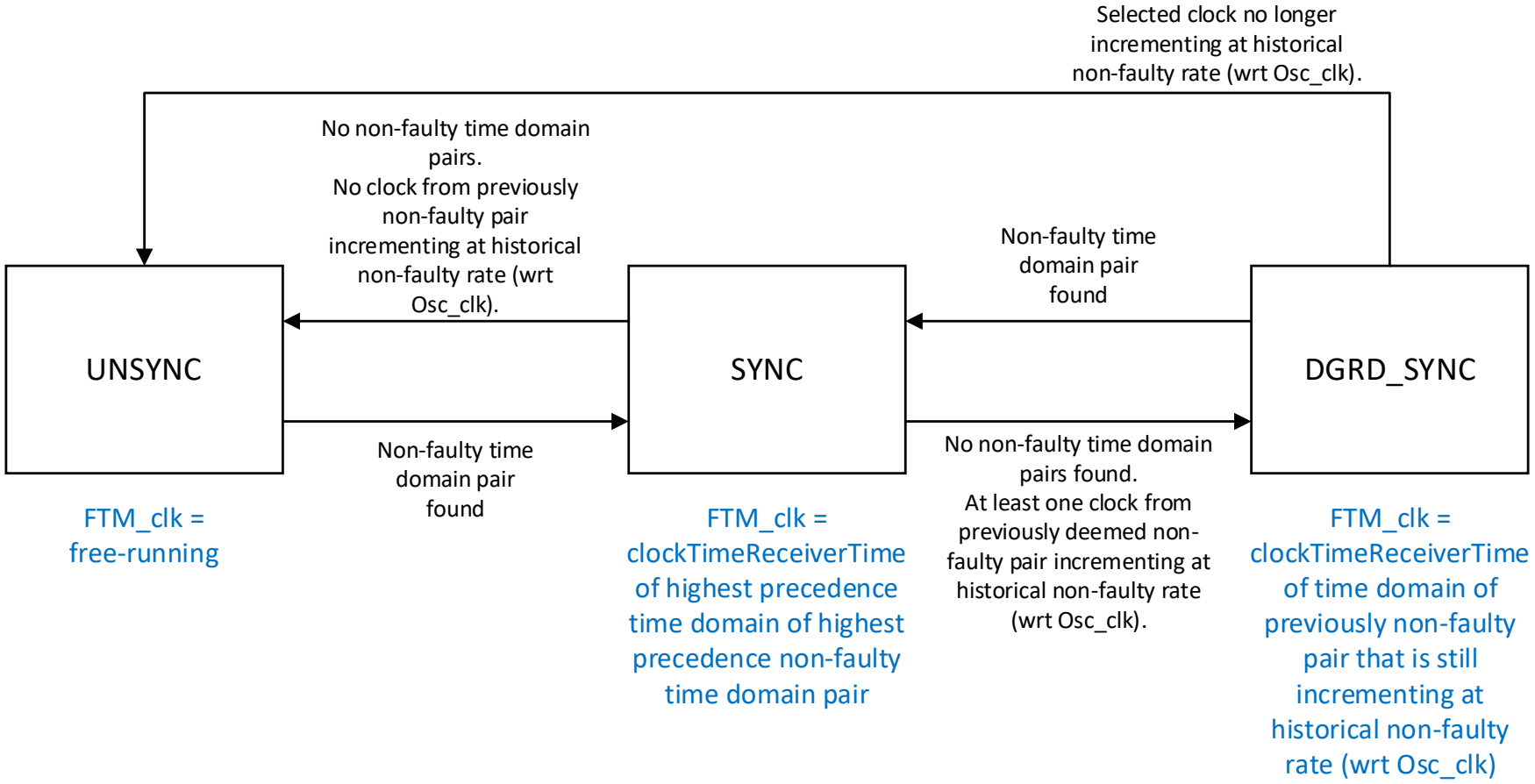
- Each FTM has (at least) three states:
  - **Unsynchronized** – The FTM is not synchronized, increasing its reliance on the validity of the incoming timing information.
    - The FTM is more vulnerable to faults causing system failure.
  - **Synchronized** – The FTM is synchronized, meaning that it can filter out arbitrary values from a faulty GM or plane.
    - Relies on knowledge that non-faulty time domains are aligned to within (Δ + max_accepted_propagation_skew).
  - **Degraded Synchronized** – The FTM has lost redundancy on the incoming timing information but uses its local frequency reference and historical rate information to determine which one, if any, of its incoming time domains is non-faulty.
    - Uses historical relationship of local oscillator to non-faulty time domains

Definitions Used by the FTM Algorithms

- FTM_clk
  - Time clock produced by the FTM
- Osc_clk
  - Local oscillator clock at the end station or bridge
- clockTimeReceiverTime
  - Synchronized time maintained at the timeReceiver of a PTP End Instance
- Configured hierarchy of time domains
  - A time domain, TD_i, with a lower numbered label i has a higher precedence than one with a higher numbered label i
  - Time domain pairs, TD_{ij}:
    - With a lower numbered label i has a higher precedence than one with a higher numbered label i
    - If i is equal, with a lower numbered label j has a higher precedence than one with a higher numbered label j

Closest-Paired Selection – Concept

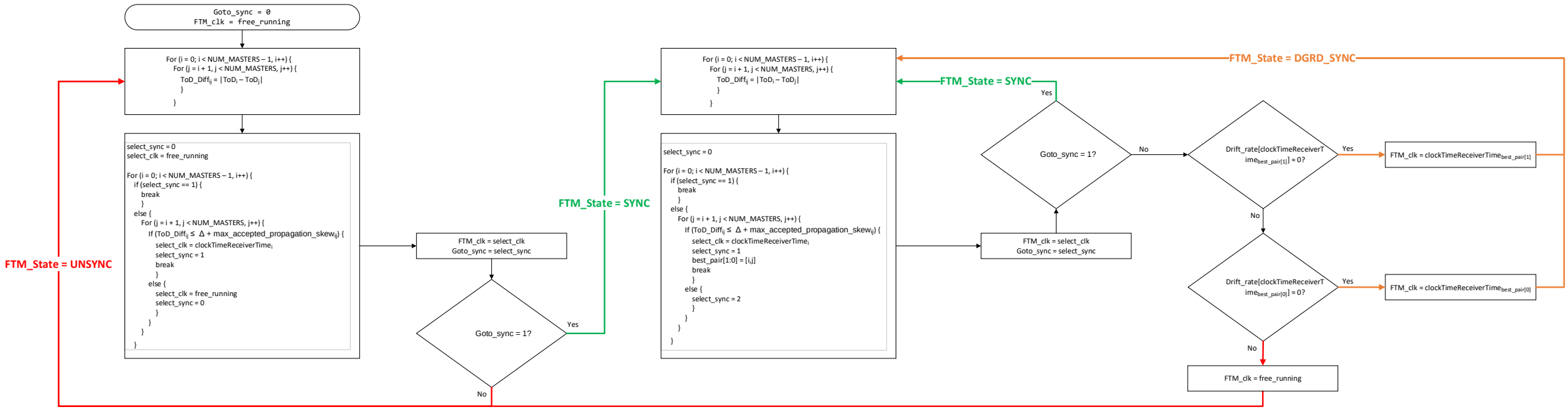
- This algorithm selects the clock from the non-faulty time domain with the highest precedence from the non-faulty time domain pair with the highest precedence



Closest-Paired Selection – Example

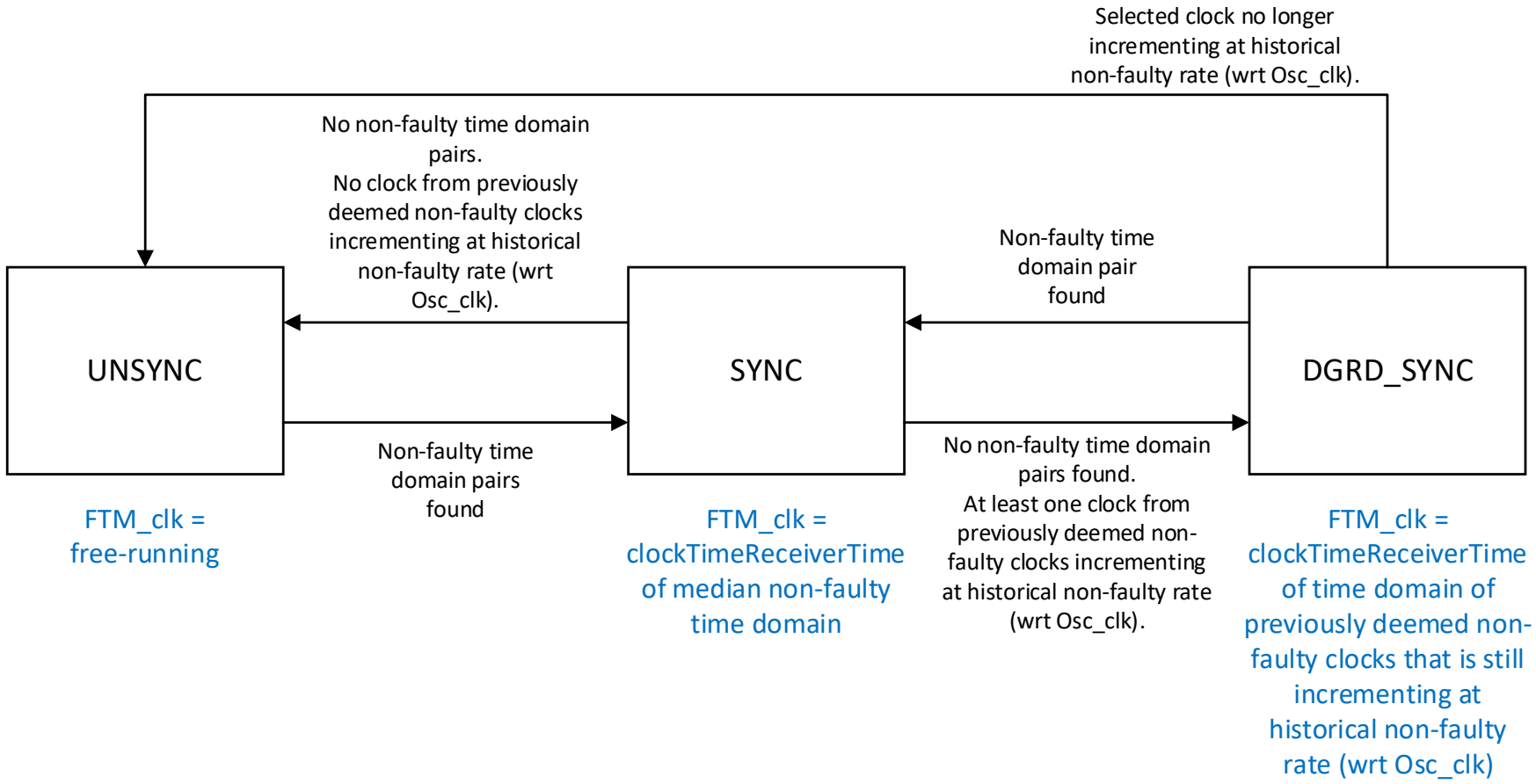
- Time domains 0, 1, and 2 and configured precedence:
  - Time domain 0 has highest precedence
  - Time domain 1 has 2nd precedence
  - Time domain 2 has lowest precedence
- Compare difference magnitudes:
  - $|ToD_0 - ToD_1|$
  - $|ToD_0 - ToD_2|$
  - $|ToD_1 - ToD_2|$
- Select best clock:
  - $clockTimeReceiverTime_0$ if either of first two difference magnitudes ≤ (Δ + max_accepted_propagation_skew(
  - $clockTimeReceiverTime_1$ if only the last difference magnitude ≤ (Δ + max_accepted_propagation_skew)
- For simplicity, DGRD_SYNC functions not shown

Closest-Paired Selection – Algorithm



Mid-Value Selection – Concept

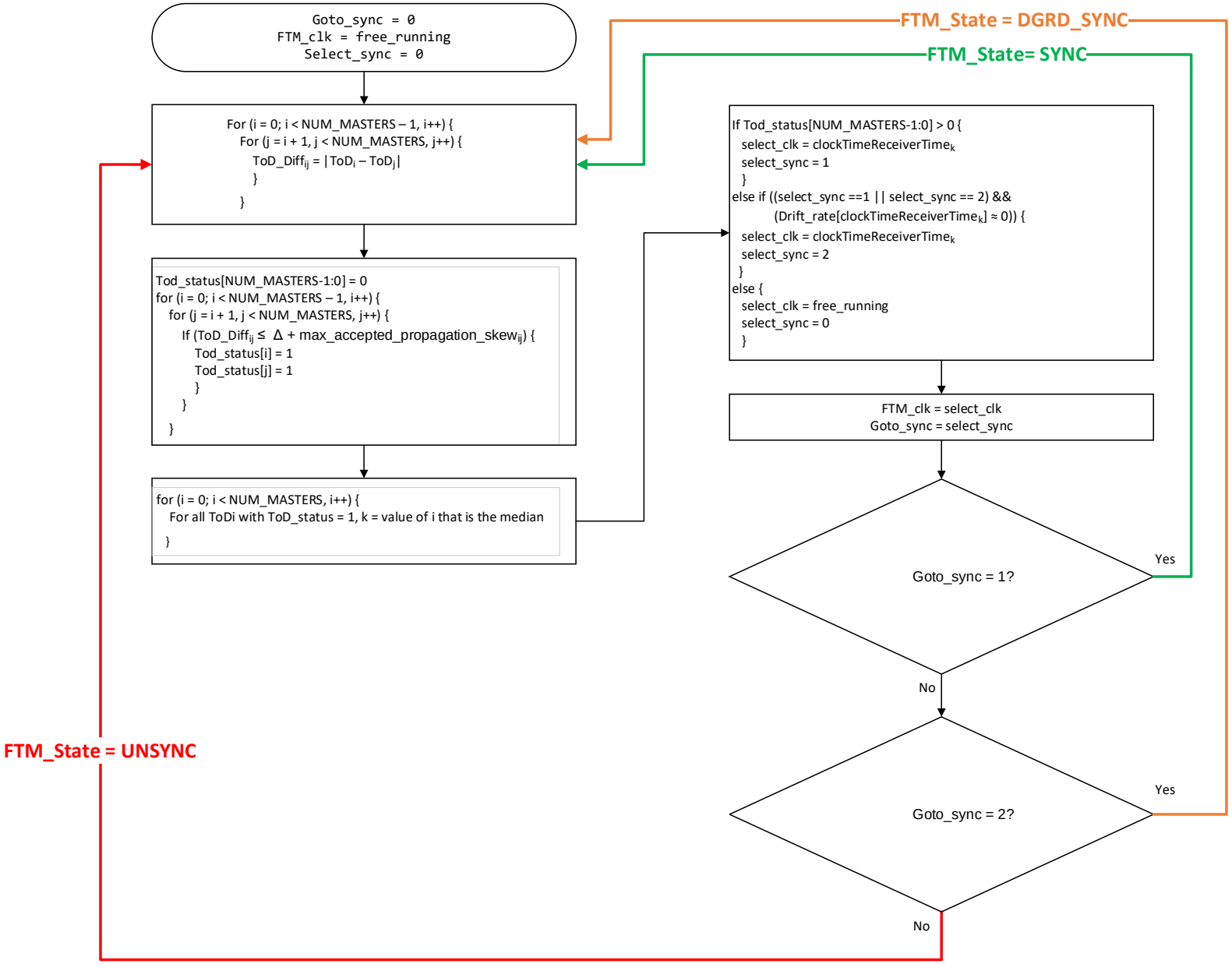
- FTM selects the median of all the clocks that are available and that are deemed to be non-faulty



Mid-Value Selection – Example

- Time domains 0, 1, and 2 and configured precedence:
  - Time domain 0 has highest precedence
  - Time domain 1 has 2nd precedence
  - Time domain 2 has lowest precedence
- Compares difference magnitudes:
  - $|ToD_0 - ToD_1|$
  - $|ToD_0 - ToD_2|$
  - $|ToD_1 - ToD_2|$
- Validate clocks:
  - $clockTimeReceiverTime_0$ non-faulty if either of first two difference magnitudes ≤ (Δ + max_accepted_propagation_skew)
  - $clockTimeReceiverTime_1$ non-faulty if either the first or the last difference magnitude ≤ (Δ + max_accepted_propagation_skew)
  - $clockTimeReceiverTime_2$ non-faulty if either of the last two difference magnitudes ≤ (Δ + max_accepted_propagation_skew)
- Select best clock:
  - Select median of non-faulty clockTimeReceiverTimes if odd number of non-faulty clocks
  - Select the first of the middle two non-faulty clockTimeReceiverTimes if even number of non-faulty clocks
- For simplicity, DGRD_SYNC functions not shown

Mid-Value Selection – Algorithm



Algorithm Mapping

- Possible to apply either algorithm at end station or bridge
  - For example
    - End-station may perform FT-Mid-Value Selection
    - Bridge may perform Closest-Pair Selection
      - May reduce software complexity on relay node
- Effect of FTM algorithm on time is local to the end station or to the bridge
  - FTMs in different end stations and bridges do not need to use the same algorithm
  - FTMs can use proprietary algorithms instead of the default algorithm

What's Next?

- Clean-up the FTM algorithms:
  - Improve clarity and fix bugs
  - Propose options for determining max_accepted_propagation_skew
  - Write clause 7.1.4 of P802.1DP
- Potential enhancements for FTM algorithms:
  - Time domain merging (e.g., averaging)?
- Potential informative material for annexes of P802.1DP:
  - Time agreement generation (i.e., synchronizing the GMs)?
  - Example network topologies?
  - New network topologies?

Summary

- Fault tolerant time propagation without fault tolerant time generation agreement is not terribly complicated
  - Not a Byzantine problem
  - FTM only affects local End Station/Bridge
  - No changes needed for IEEE 802.1AS
- Two FTM algorithms proposed
  - Closest Pair Selection
  - Mid-Value Selection

References

1. Driscoll K, Hall B, et al., Byzantine Fault Tolerance, from Theory to Reality, SpringerLink.
2. Osder S, Generic faults and design solutions for flight-critical systems, ARC Aerospace Research Central (arc.aiaa.org).
3. Minor P, et al., A Unified Fault-Tolerance Protocol, NASA Technical Reports

Questions?