60802 Time Sync – Monte Carlo Simulations with RR & NRR Drift Tracking and Compensation – Implementation

David McCall – Intel Corporation

Version 1

References

1 – David McCall "60802 Time Synchronisation – Monte Carlo Analysis: 100-hop Model, "Linear" Clock Drift, NRR Accumulation – Overview & Details, Including Equations – v2", contribution to IEC/IEEE 60802, September 2022

2 – David McCall "60802 Time Sync – RR & NRR Drift Tracking and Compensation – Implications – v02", contribution to IEC/IEEE 60802, June 2023

Contents

- Overview of Simulation
- Recap of dTE Related to RR Drift
- Problems with using NRR Drift Tracking & Error Compensation Approach for RR
- Alternative Approach: Additional Field in New TLV & Additional Calculations
- Implications for Grandmasters, End Stations & Normative Requirements
- Next Steps
  - 802.1ASdm PAR & CSD
- Summary

Overview of Simulation

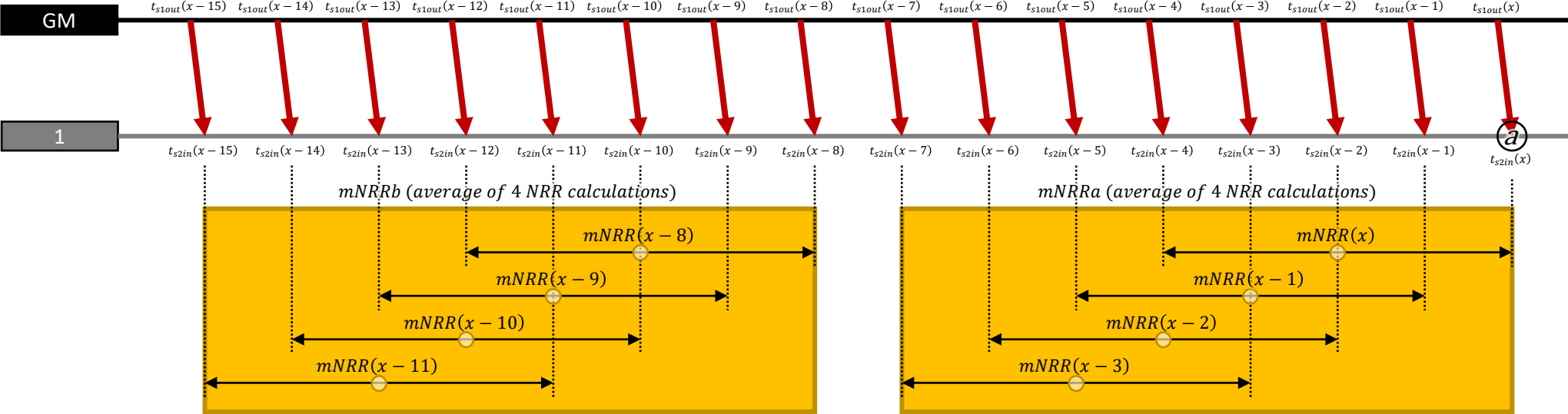Addition of NRR & RR drift tracking and compensation to Monte Carlo Multi-Hop Simulation
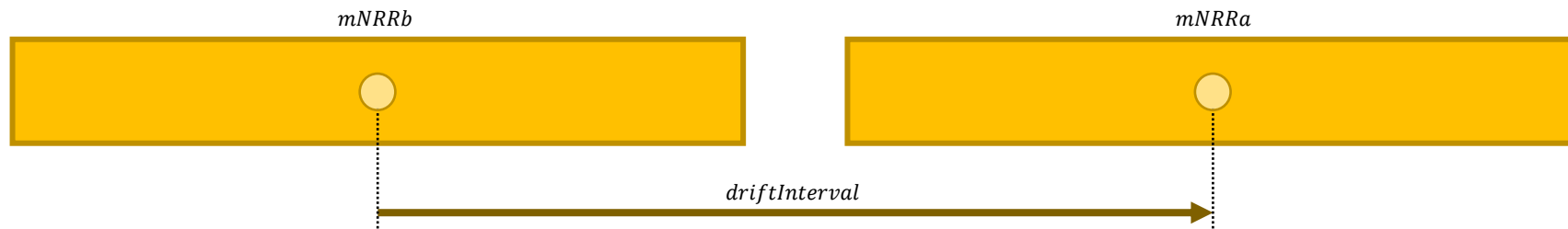
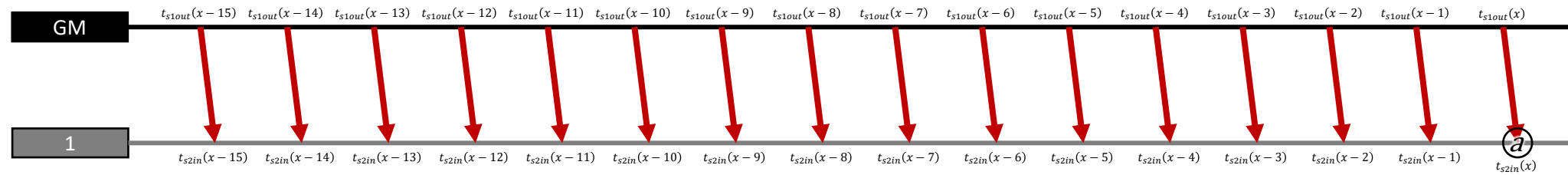# Neighbor Rate Ratio Calculation (NRR)
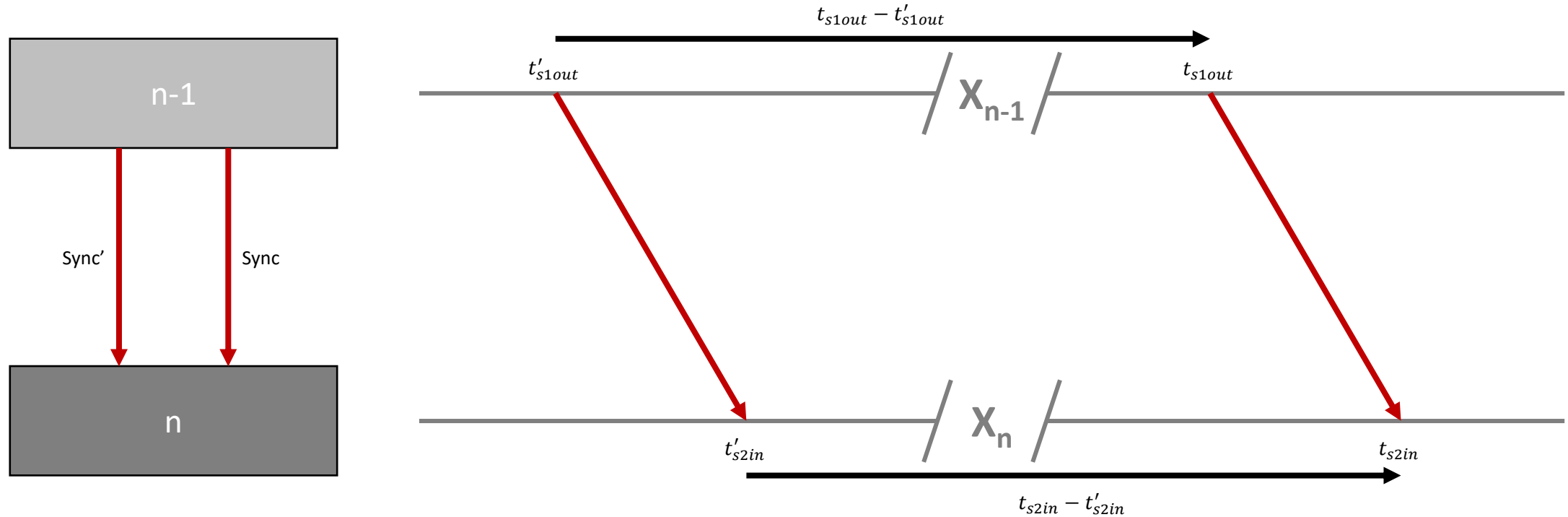
$$mNRR = \left(\frac{t_{s1out} - t'_{s1out}}{t_{s2in} - t'_{s2in}} \right)$$

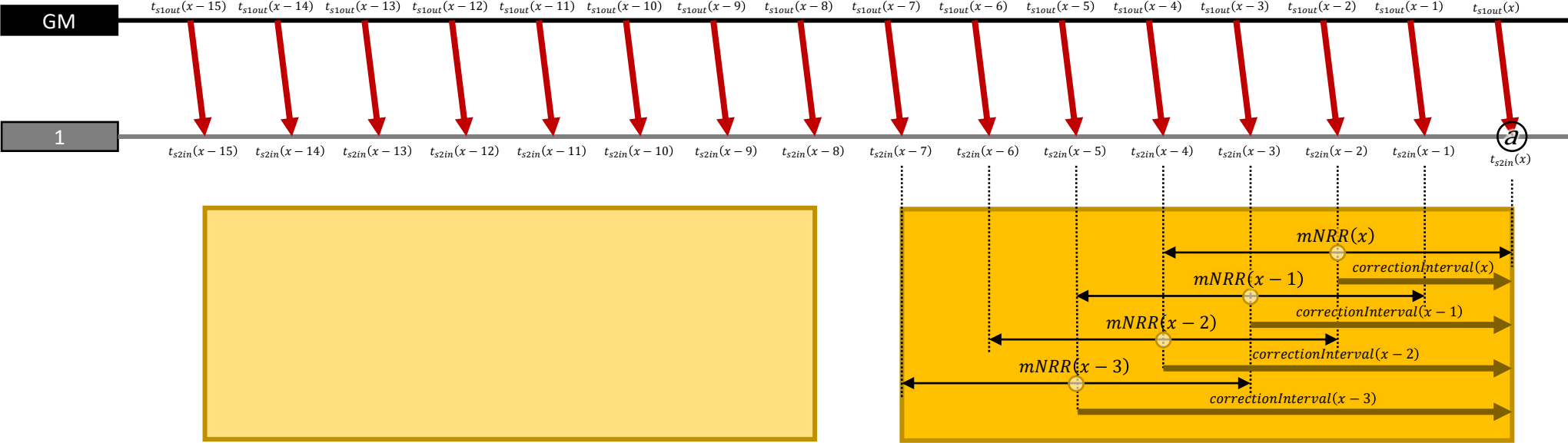**ppm**

NRR Drift Rate Calculation
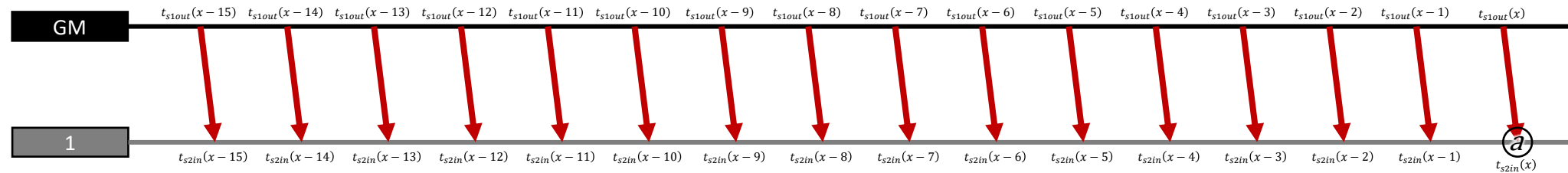
NRR Drift Rate Calculation

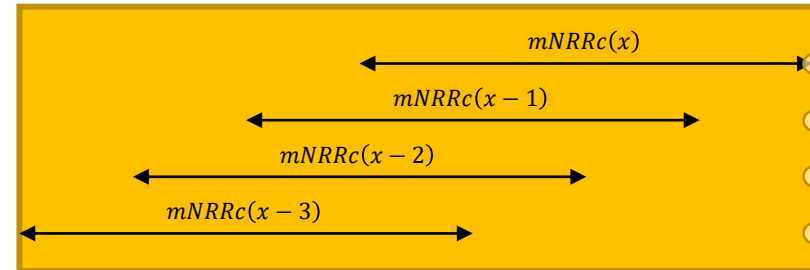$$NRRdriftRate = \frac{mNRRa - mNRRb}{driftInterval}$$

NRR Drift Rate Calculation

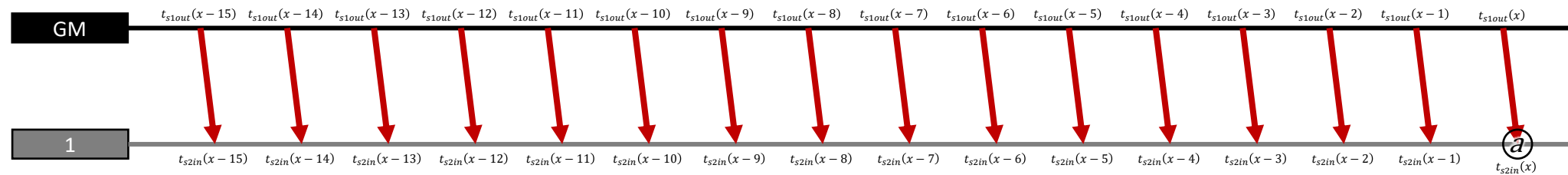$$mNRRc = mNRR + (NRRdriftRate \times correctionInterval)$$

NRR Drift Rate Calculation

$t_{s1out}(x-15)$ $t_{s1out}(x-14)$ $t_{s1out}(x-13)$ $t_{s1out}(x-12)$ $t_{s1out}(x-11)$ $t_{s1out}(x-10)$ $t_{s1out}(x-9)$ $t_{s1out}(x-8)$ $t_{s1out}(x-7)$ $t_{s1out}(x-6)$ $t_{s1out}(x-5)$ $t_{s1out}(x-4)$ $t_{s1out}(x-3)$ $t_{s1out}(x-2)$ $t_{s1out}(x-1)$ $t_{s1out}(x)$

GM

1

$t_{s2in}(x-15)$ $t_{s2in}(x-14)$ $t_{s2in}(x-13)$ $t_{s2in}(x-12)$ $t_{s2in}(x-11)$ $t_{s2in}(x-10)$ $t_{s2in}(x-9)$ $t_{s2in}(x-8)$ $t_{s2in}(x-7)$ $t_{s2in}(x-6)$ $t_{s2in}(x-5)$ $t_{s2in}(x-4)$ $t_{s2in}(x-3)$ $t_{s2in}(x-2)$ $t_{s2in}(x-1)$

a

$t_{s2in}(x)$

mNRRca (corrected to point a; average of 4 mNRRc calculations)

$mNRRc(x)$

$mNRRc(x-1)$

$mNRRc(x-2)$

$mNRRc(x-3)$

NRR Drift Rate Calculation
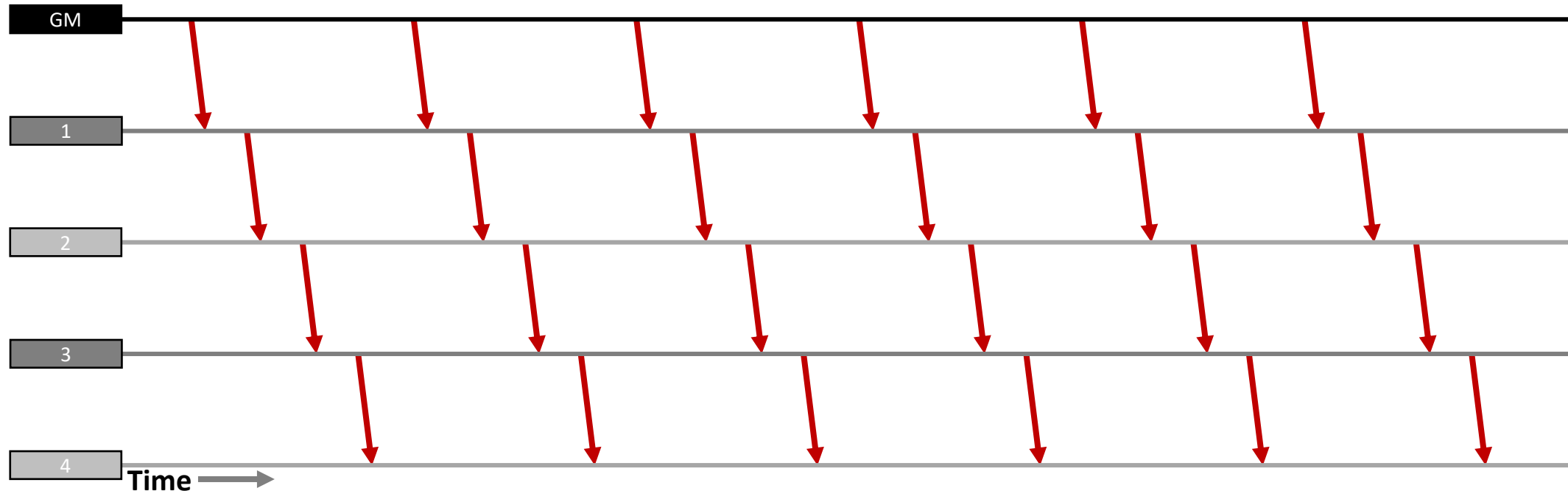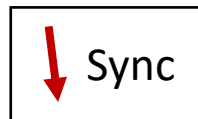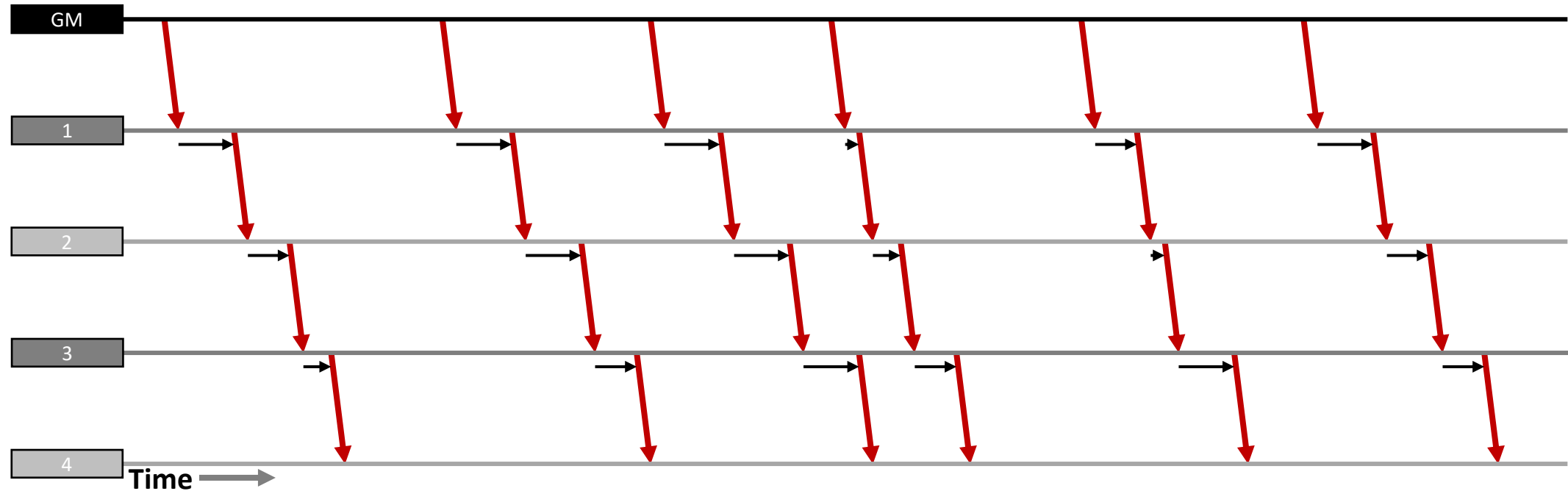
Sync Message Intervals – Idealised*

Consistent Sync Interval at GM; same Residence Time, every time (but…*Not to Scale)

Sync Intervals at the GM (outgoing) and the first node (incoming) are not exactly 125ms, but are modeled as a uniform distribution between 119ms and 131ms.
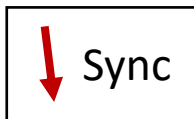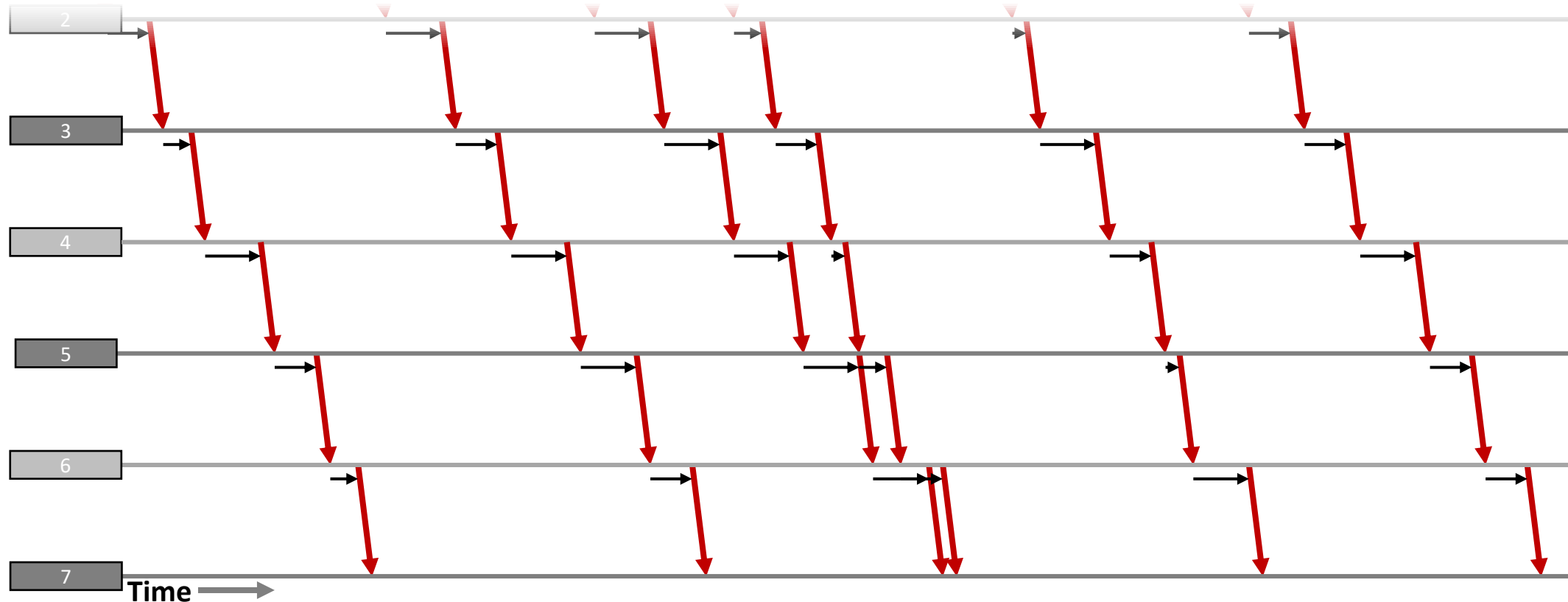
Sync Message Intervals – "Realistic"*

Variable Sync Interval at GM; variable Residence Times (but still...*Not to Scale)

At the subsequent nodes, the intervals between Sync messages (incoming) are the intervals at the previous node modified by the Residence Time at the previous node. Residence Time is modeled as a normal distribution with average 5 ms, standard deviation of 1.8 ms, truncated at 1 ms and 15 ms (outliers are mapped to 1 ms and 15 ms)

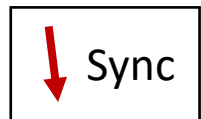# Sync Message Intervals – "Realistic"*

Variable Sync Interval at GM; variable Residence Times (but still...*Not to Scale)

Time

| ↓ Sync |

What happens when Sync messages get close?...

Sync Message Intervals – "Realistic"*
Variable Sync Interval at GM; variable Residence Times (but still...*Not to Scale)
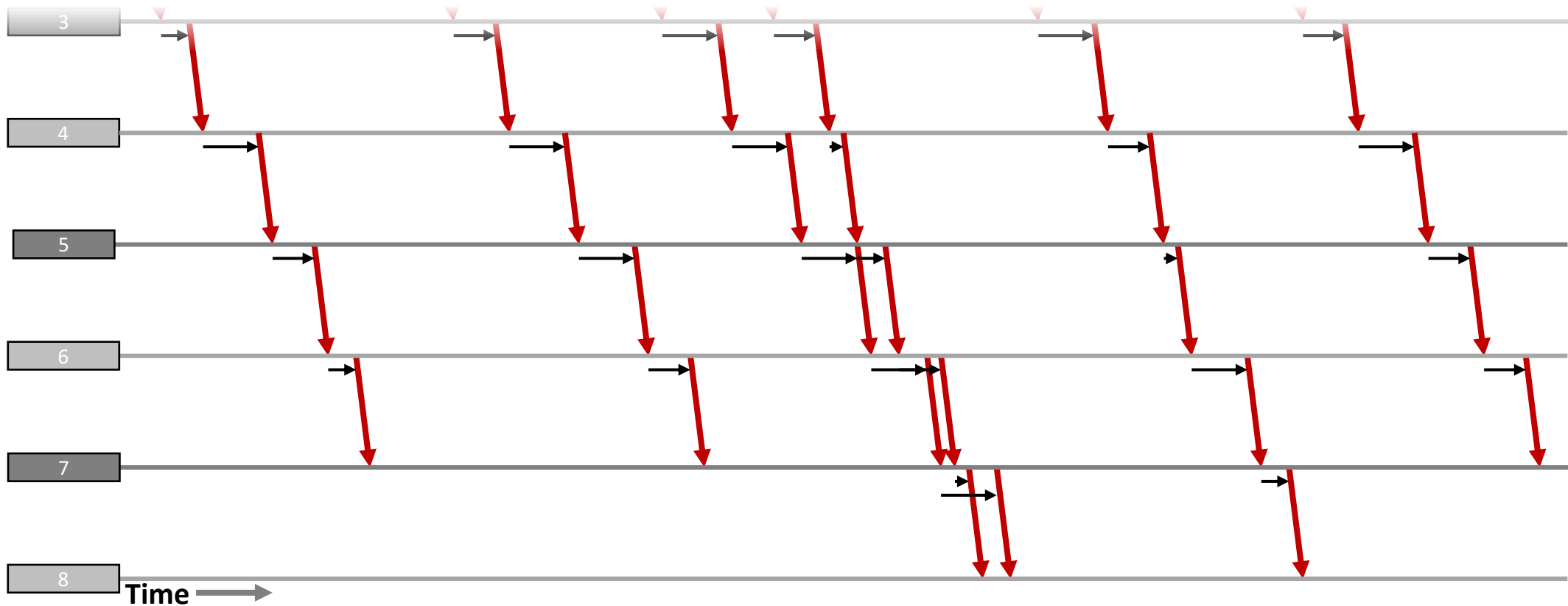
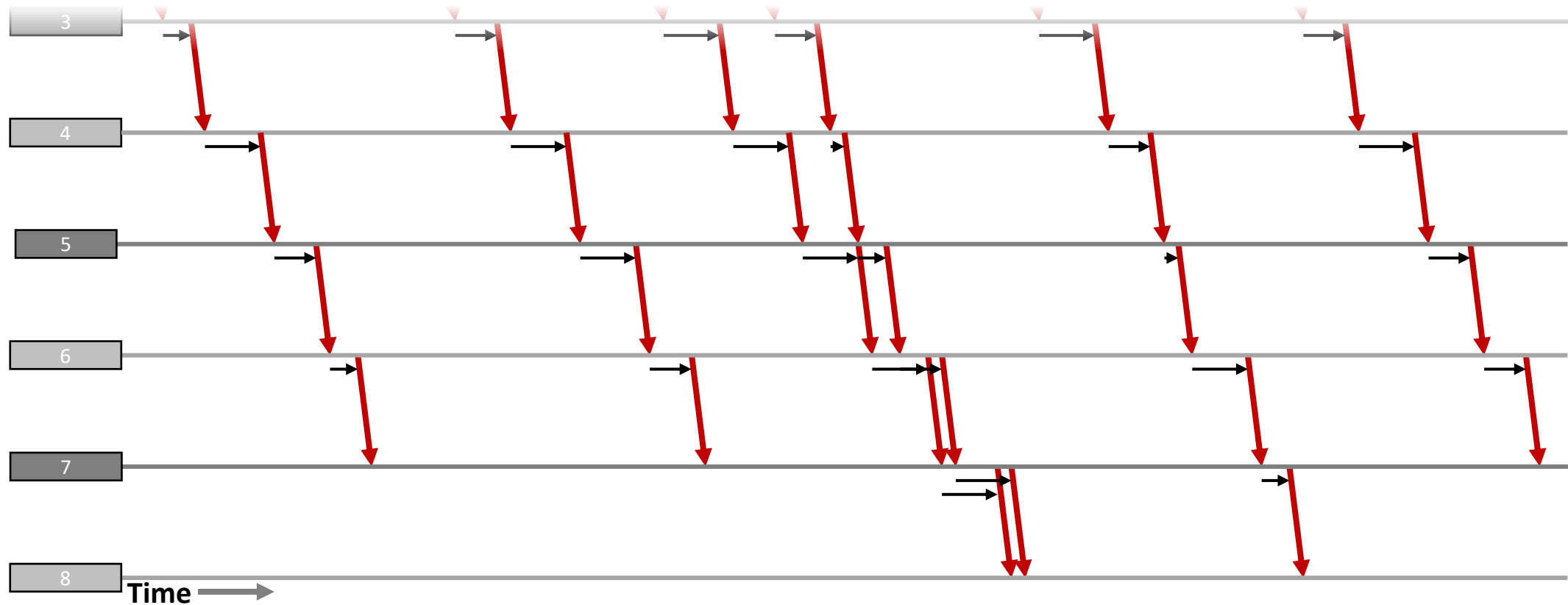| ↓ Sync | And then swap positions? And do measures to address it affect subsequent Sync messages? This is an issue for simulations and potentially 1-step sync. See [2] for details. |

Sync Message Intervals – "Realistic"*
Variable Sync Interval at GM; variable Residence Times (but still...*Not to Scale)

Monte Carlo simulation increases Residence Time of message that fell behind (or any Sync message that is closer that 1 ms to the previous one) so that it is at least 1 ms after previous Sync message. This does not affect subsequent Sync messages...unless increasing the Residence Time reduces the time between the adjusted Sync message and the subsequent Sync message is now less than 1ms.

Relevant Intervals – Hop 1
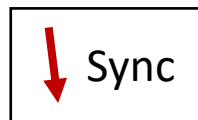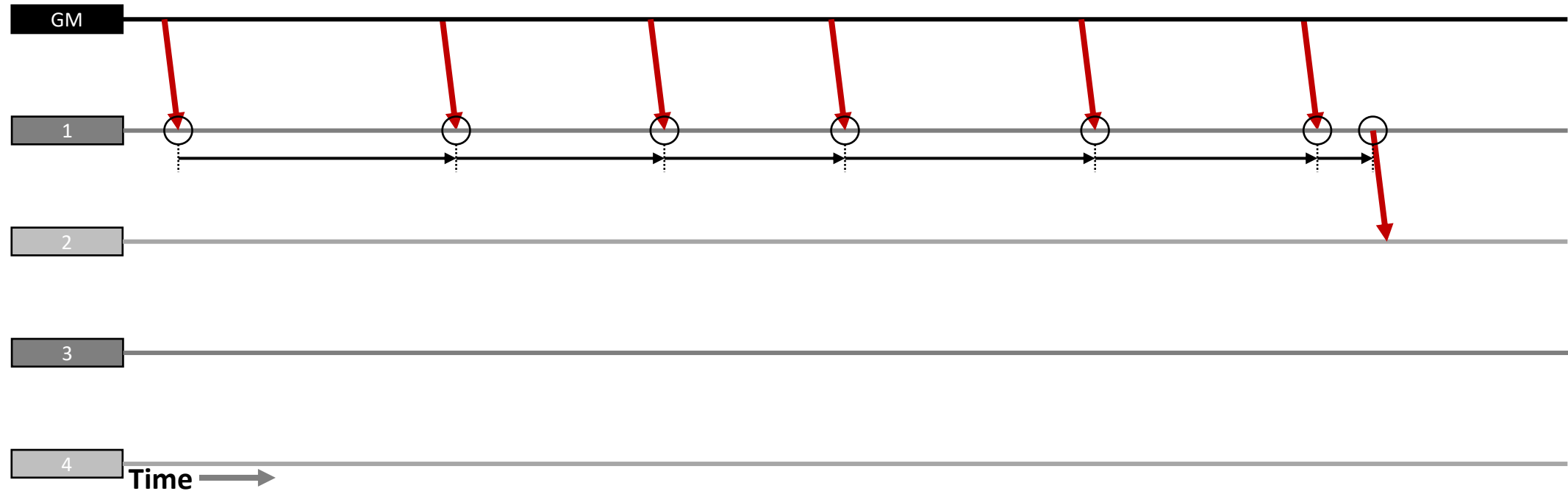
| | Sync |

Hop 1: generate Sync Intervals and all Residence Times. Only use final Residence Time for calculations, until end of calculation loop, then calculate new Sync Intervals for next hop.

Relevant Intervals – Hop 2
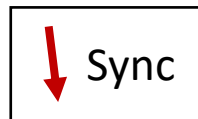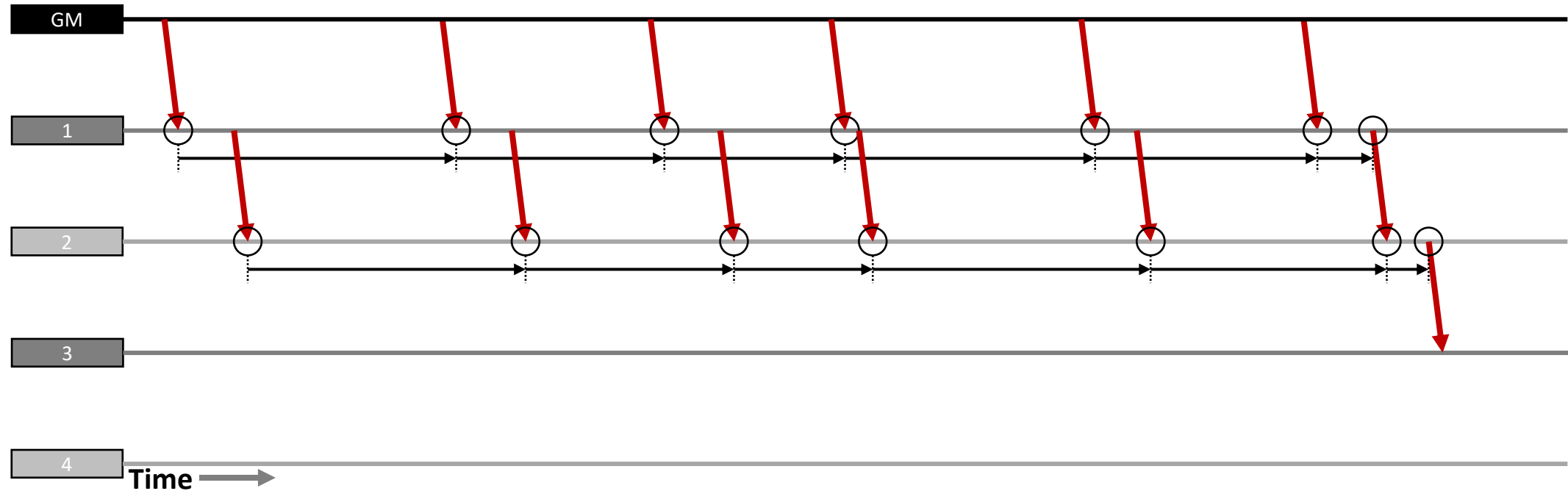
| | Sync | Hop 2: Calculate new residence times. Only use final Residence Time for calculations, until end of calculation loop, then calculate new Sync Intervals for next hop. |

Relevant Intervals – Hop 3

Sync

Hop 3: Calculate new residence times. Only use final Residence Time for calculations, until end of calculation loop, then calculate new Sync Intervals for next hop.
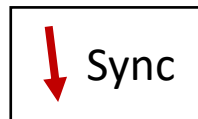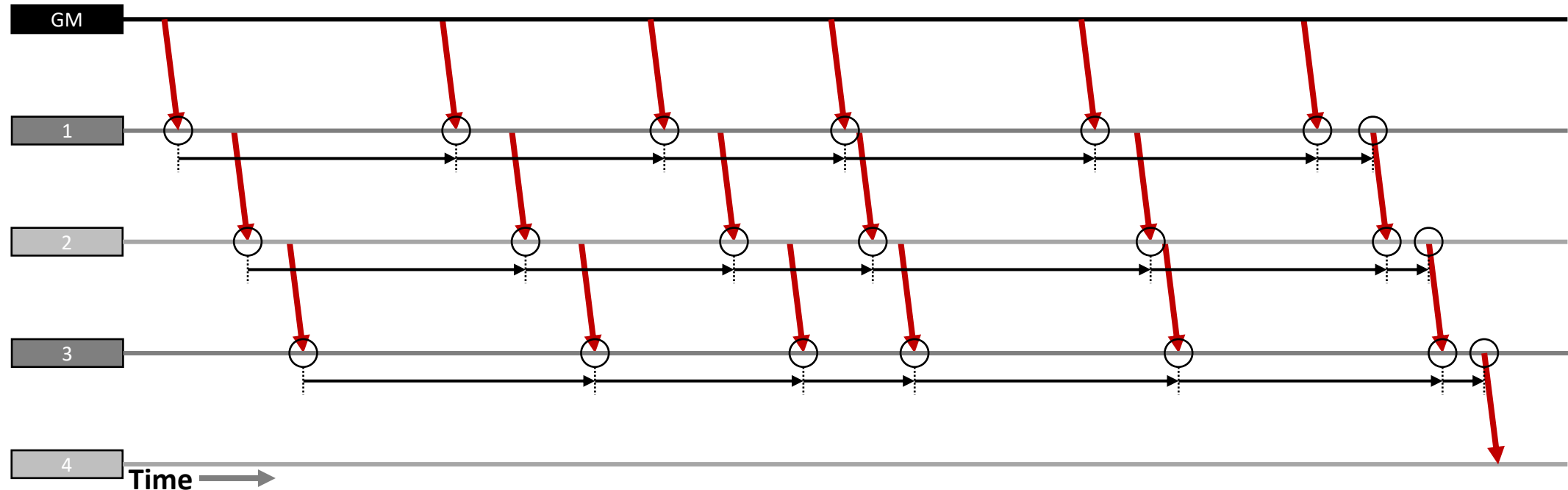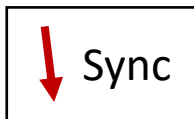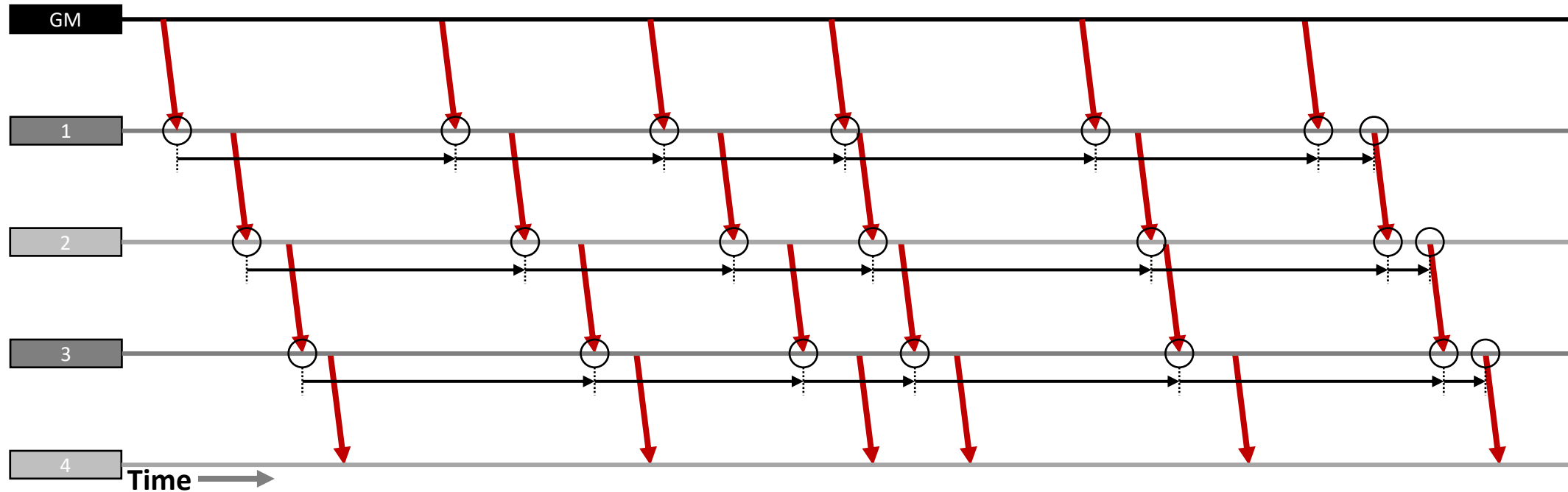
Relevant Intervals – Hop 4

Sync

Hop 4: No residence times.

RRdriftRate, mRR & Correction Field Calculations at PTP Relays

GM

1

$$RRdriftRate(1) = NRRdriftRate(1)$$

$$correctionField(1) = \left(1 + \frac{mRR_b(1)}{10^6}\right) \times \left(meanLinkDelay(1) + residenceTime(1)\right)$$

$$mRR_c(1) = mRR_a(1) + RRdrift_{1/1}(1a \rightarrow 1c) = mNRRc_a(1) + NRRdrift(1a \rightarrow 1c)$$

c

residenceTime + meanLinkDelay

meanLinkDelay

residenceTime + meanLinkDelay / 2

$$RRdriftRate(2) = \frac{RRdriftRate(1)}{\left(1 + \frac{mNRRc_a(2)}{10^6}\right)} + NRRdriftRate(2)$$

$$mRR_a(2) = mRR_c(1) + RRdrift_{1/2}(1c \rightarrow 2a) + mNRRc_a(2)$$

$$RRdriftRate(2)$$
$$correctionField(2)$$
$$mRR_c(2) = mRR_a(2) + RRdrift_{2/2}(2a \rightarrow 2b)$$

t$_{1out}$

2

a b c

t$_{2in}$

$$mRR_b(2) = mRR_a(2) + RRdrift_{2/2}(2a \rightarrow 2b)$$

$$correctionField(2) = correctionField(1) + \left(1 + \frac{mRR_b(2)}{10^6}\right) \times \left(meanLinkDelay(2) + residenceTime(2)\right)$$

residenceTime - meanLinkDelay / 2

residenceTime

3

Time

RRdriftRate, mRR & Correction Field Calculations at PTP Relays

$$RRdriftRate(1) = NRRdriftRate(1)$$

$$correctionField(1) = \left(1 + \frac{mRR_b(1)}{10^6}\right) \times \left(meanLinkDelay(1) + residenceTime(1)\right)$$

$$mRR_c(1) = mRR_a(1) + RRdrift_{1/1}(1a \rightarrow 1c) = mNRRc_a(1) + NRRdrift(1a \rightarrow 1c)$$

meanLinkDelay

$$RRdriftRate(2) = \frac{RRdriftRate(1)}{\left(1 + \frac{mNRRc_a(2)}{10^6}\right)} + NRRdriftRate(2)$$

$$mRR_a(2) = mRR_c(1) + RRdrift_{1/2}(1c \rightarrow 2a) + mNRRc_a(2)$$

Time

t_{2in}

Tsync2sync
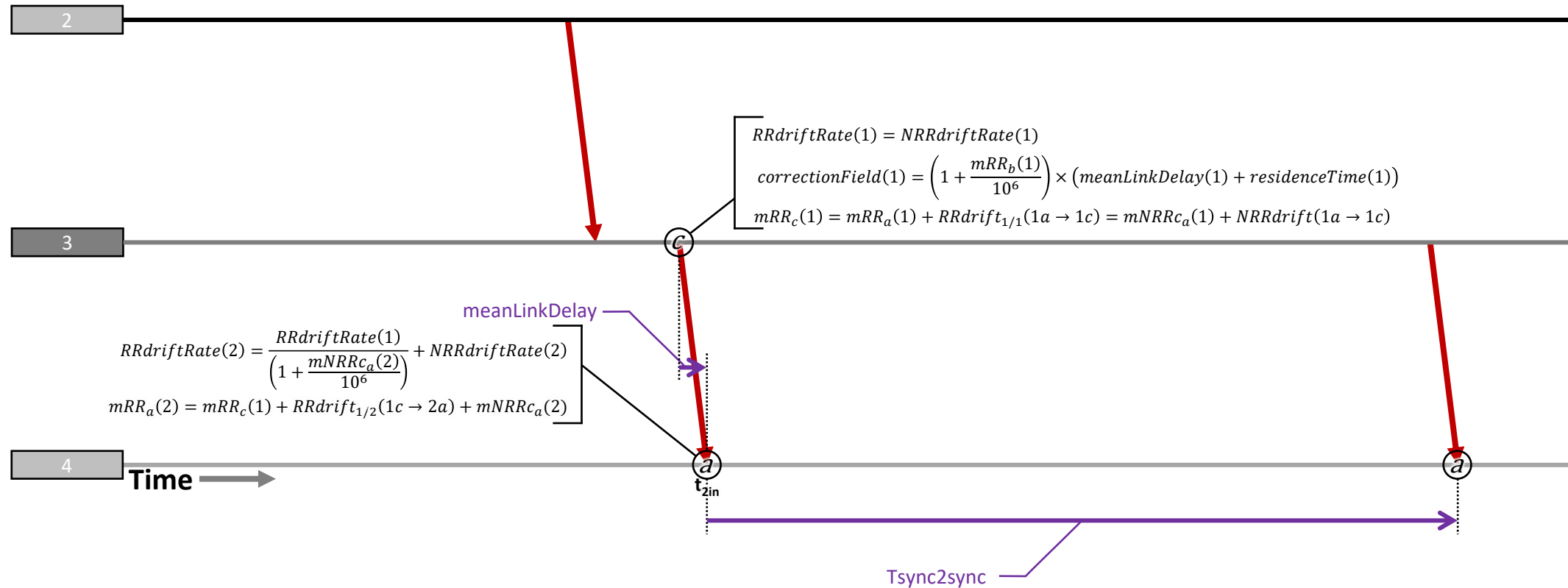
Errors at End Station are modelled via a four-step piece-wise linear model (four intervals) with two errors:
- Error due to Error in mRR at [4a] due errors in mNRR down the chain of devices
- Error due to RR drift between arrival of Sync messages (after compensation via RRdriftRate)

Thank you!