

Sensor Clock Drift Measurement based MAC Protocol Design

2009. 8. 24.

Saewoong Bahk

Network Laboratory

Seoul National University



Contents

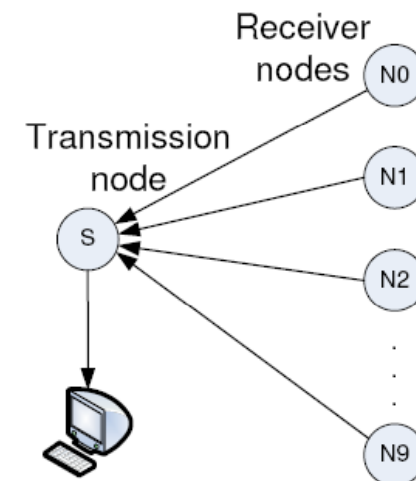
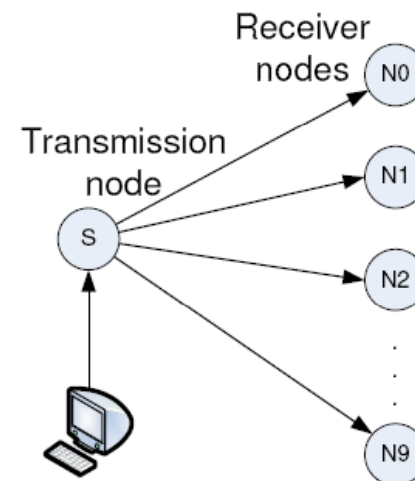
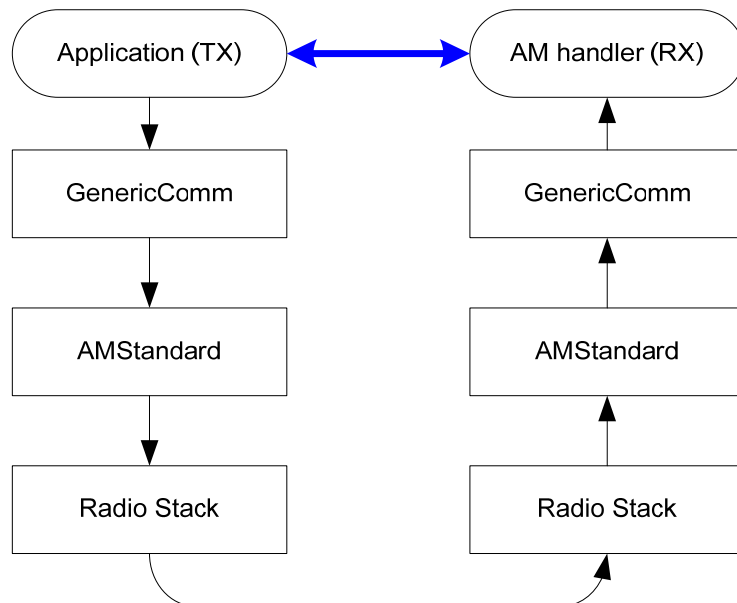
- ◆ Introduction
- ◆ Sensor Clock Drift Measurements
- ◆ Characteristics of Sensor Clock Errors
- ◆ Design of WideMAC protocol
- ◆ Performance Evaluation
- ◆ Conclusion



Sensor Clock Drift Measurements

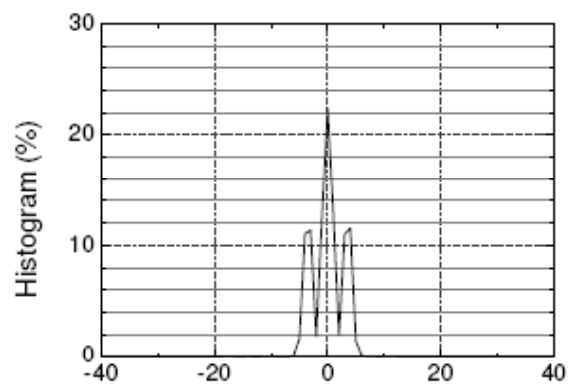
◆ Clock characteristics measurement

- 3 node type
 - TelosB, MicaZ, Hmote
- 11 nodes for each type
- Various timestamp message interval
 - 250msec ~ 60min
- ~14days, outdoor

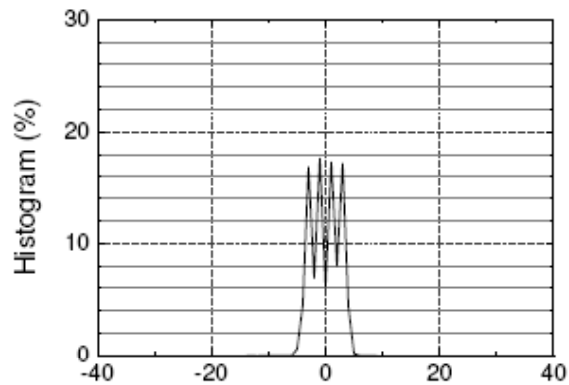


(a) Time-stamp message broadcasting. (b) Data gathering.

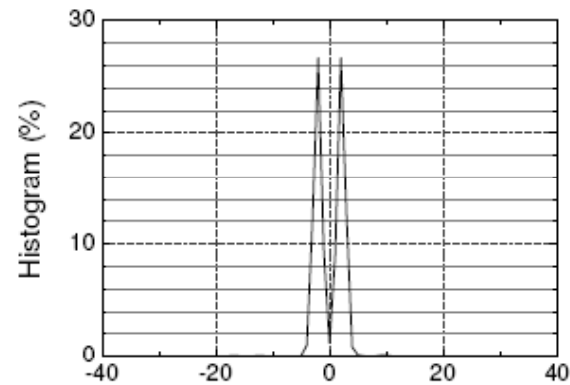
Topology for relative clock speed variation measurement.



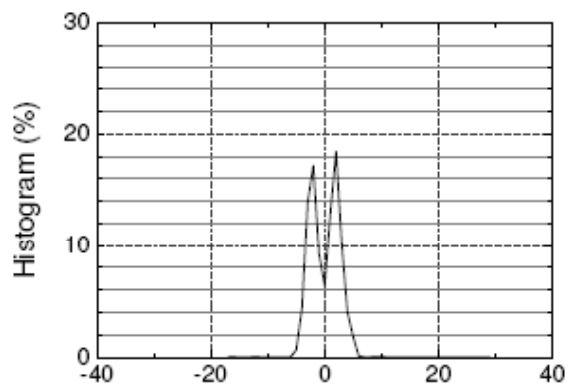
Clock drift (clock)
(a) 250msec.



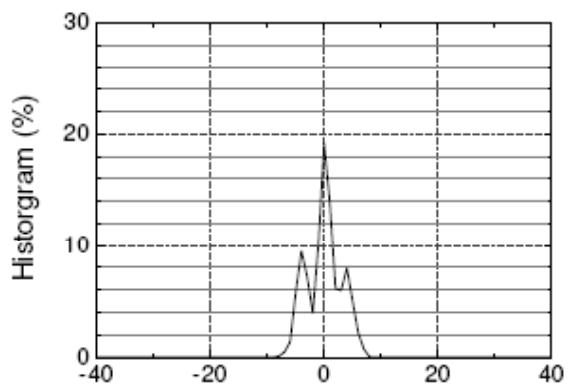
Clock drift (clock)
(b) 500msec.



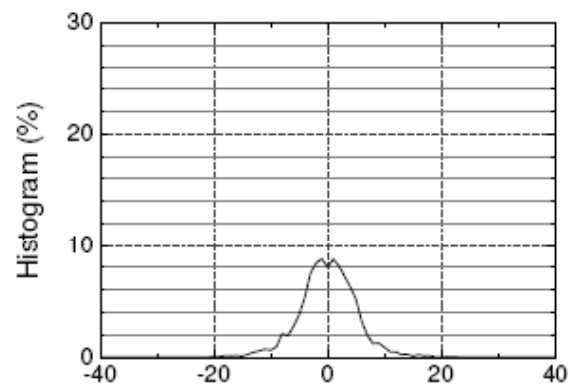
Clock drift (clock)
(c) 1sec.



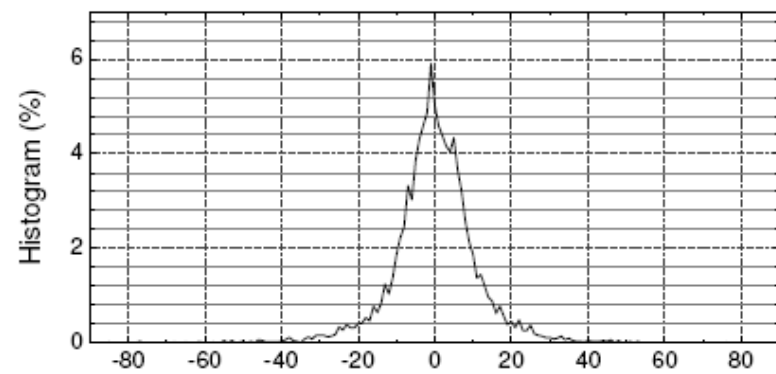
Clock drift (clock)
(d) 16sec.



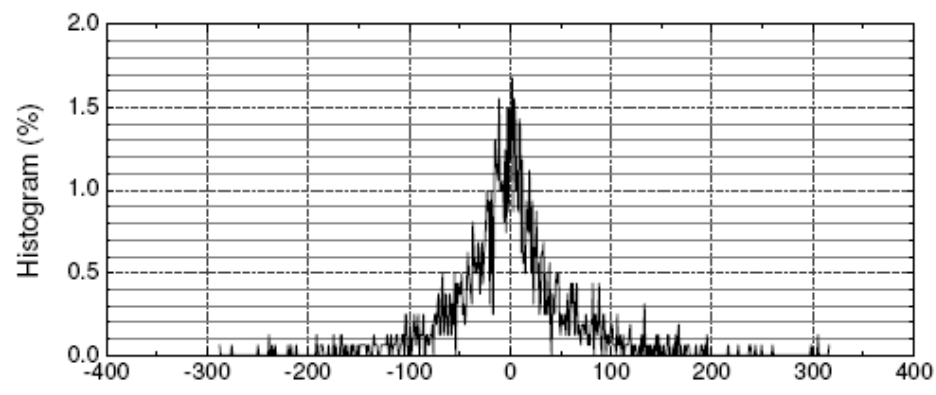
Clock drift (clock)
(e) 1min.



Clock drift (clock)
(f) 4min.



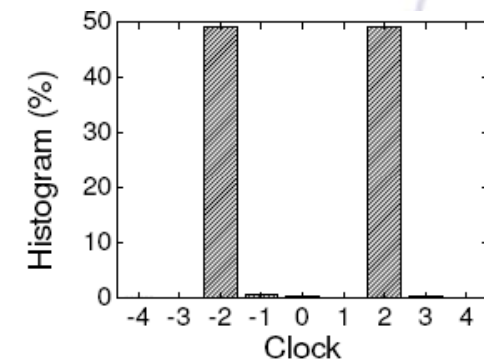
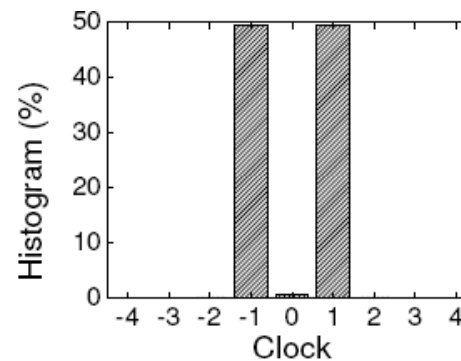
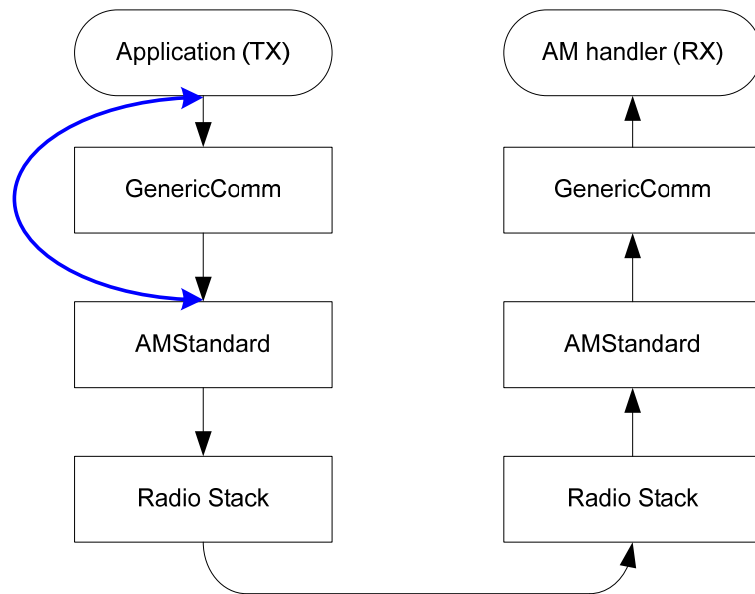
Clock drift (clock)
(g) 16min.



Clock drift (clock)
(h) 1hour.

Error source : Clock granularity

- ◆ Clock measure between Packet TX and Active message layer
 - TelosB : 32KHz-oscillator
 - min granularity : 30.5usec

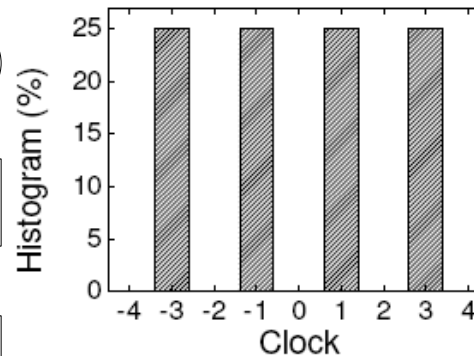
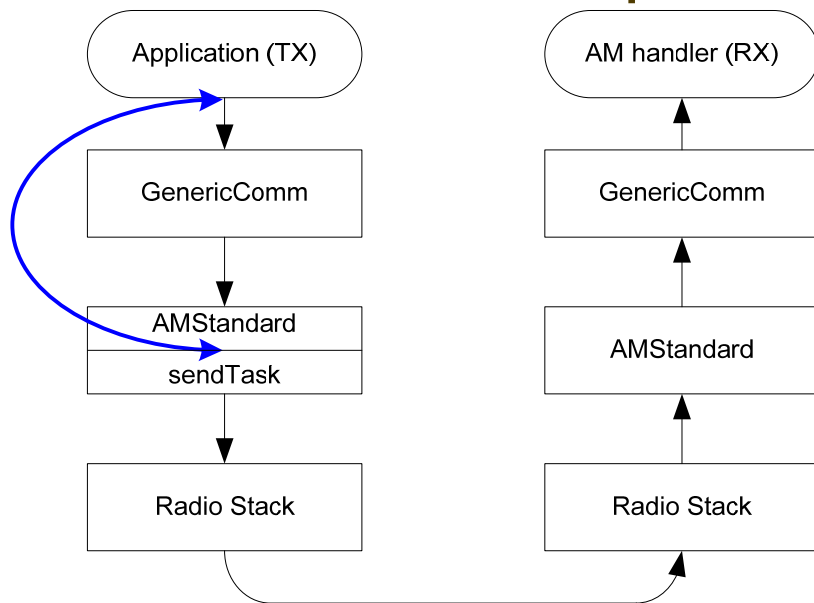


(a) Clock counter difference. (b) Variation of clock counter difference.

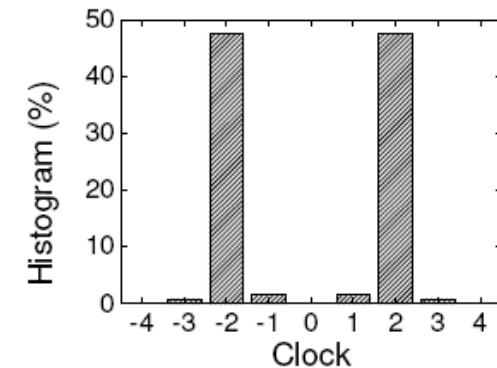
Distribution of difference between packet TX clock and Active msg layer RX clock (interval=500msec).

Error source : Scheduling delay

- ◆ Clock measurement between Packet TX and 1st task
- ◆ Scheduling delay variation when the load is high
 - Small timestamp message interval
- ◆ 3-4 local max points



(a) Interval=500msec.

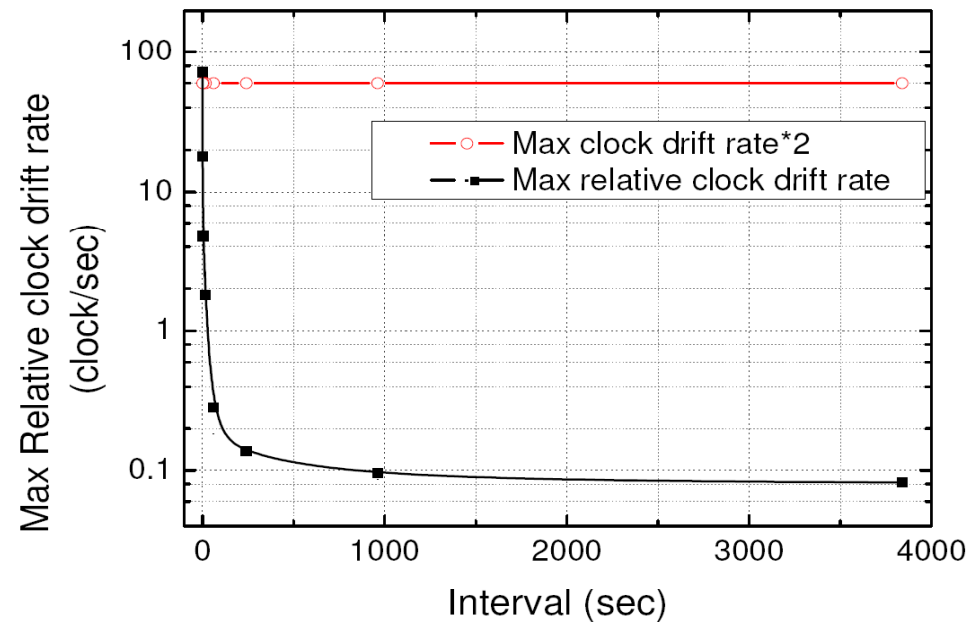


(b) Interval=1sec.

Distribution of difference between packet TX clock and first task's RX clock.

Error source : Relative clock drift

- ◆ Clock measurement between Packet TX and Packet RX
- ◆ Relative clock drift
- ◆ Clock drift is caused by environmental change



Maximum relative clock drift rate.

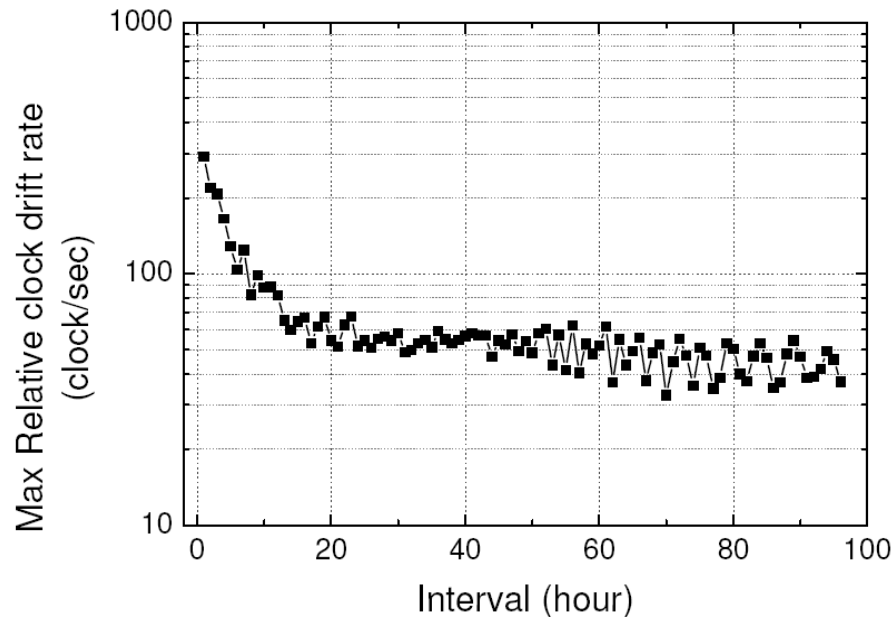


Relative Clock drift caused by Temperature

◆ Clock drift

- Temperature, humidity, aging
- Clock drift vs temperature^[24]

$$f = f_0 \cdot \{1 - 0.04 \times 10^{-6} \cdot (T - T_0)^2\},$$



f is current frequency
 T is temperature
 f_0 is base frequency
 T_0 is base temperature
usually set to 25°C .

Maximum clock drift rate based on temperature-clock drift model.

Relative Clock Drift characteristics

- ◆ *Characteristic 1:* Energy consumption **decreases** with the synchronization interval.
- ◆ *Characteristic 2:* The maximum relative clock drift rate according to the synchronization interval can be **approximated by a linear function**.
- ◆ *Characteristic 3:* **2θ can be smaller** than the maximum relative clock drift rate if the synchronization interval is small, and the difference between 2θ and the maximum relative clock drift rate increases with the synchronization interval.
- ◆ *Characteristic 4:* If the synchronization interval is small, scheduling delay affects the relative clock drift much, and its distribution doesn't follow the Gaussian any more.

WideMAC Protocol



Limitations of existing protocols

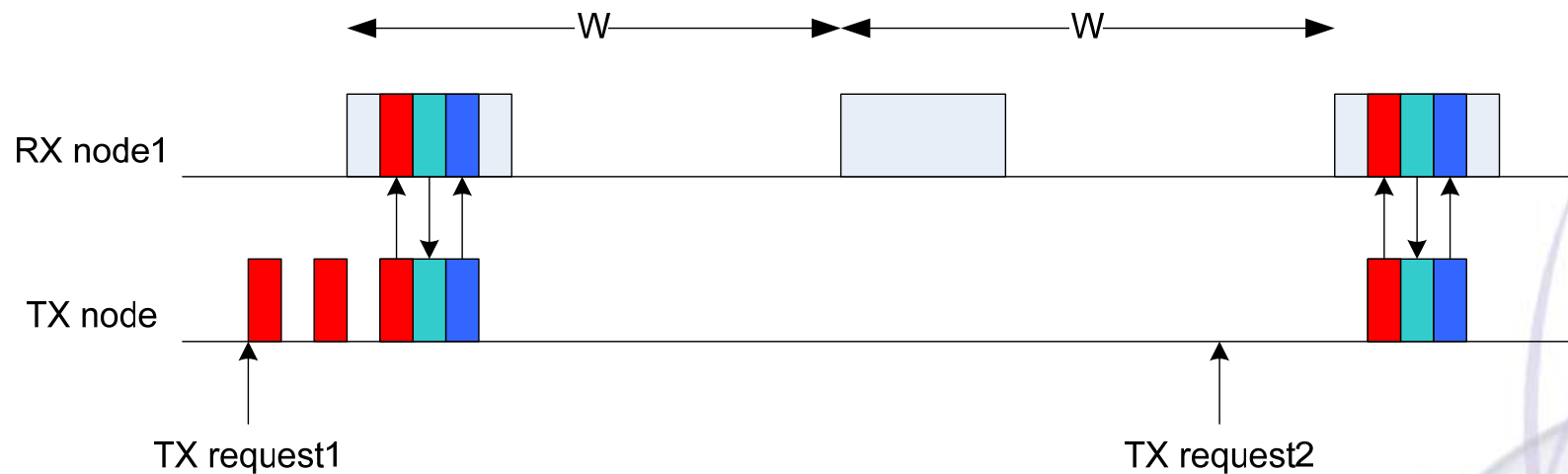
- ◆ No ultra low duty cycle Sensor MAC protocol
 - 0.1 ~ 10%
 - ➔ Previous MAC protocols w/ ultra low duty cycle($\ll 0.1\%$) mode consumes much TX energy
- ◆ High TX overhead
 - To minimize RX power consumption
 - ➔ Because source nodes' packets can be relayed by multiple nodes, source nodes can be suffered from battery run-out.
- ◆ No protocol for high reliability and low energy consumption
 - High reliability protocol consumes much energy
 - ➔ Ultra low duty cycled MAC protocols occupy channels for nothing because it takes much time to find relay node's wakeup time or synchronization ➔ high collision probability

Proposed Algorithm: WideMAC^[21]

- ◆ X-MAC based
- ◆ Ultra low duty cycle
- ◆ Clock drift characteristics
- ◆ Ultra low communication power consumption
- ◆ Ultra low TX power consumption
- ◆ High reliability



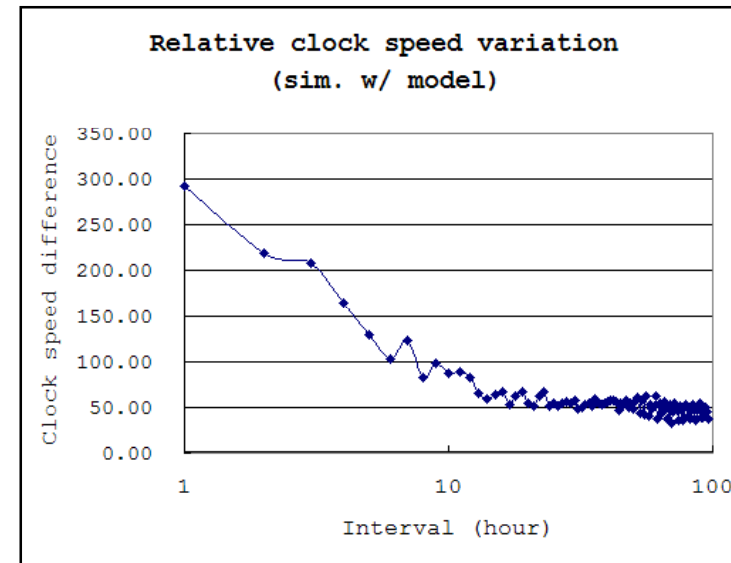
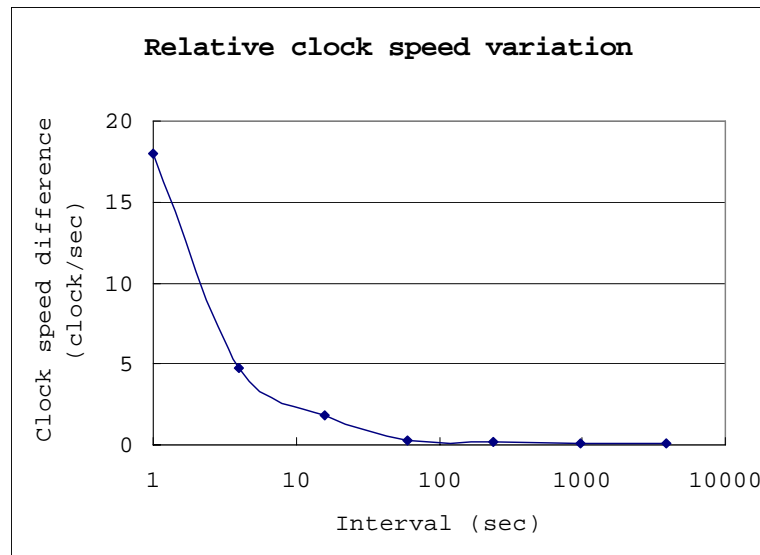
- ◆ Wakeup time estimation



- ◆ Wakeup time estimation is not easy
 - RX node and TX node clocks differ

Clock drift characteristics

- ◆ Longer synchronization interval means lower energy consumption

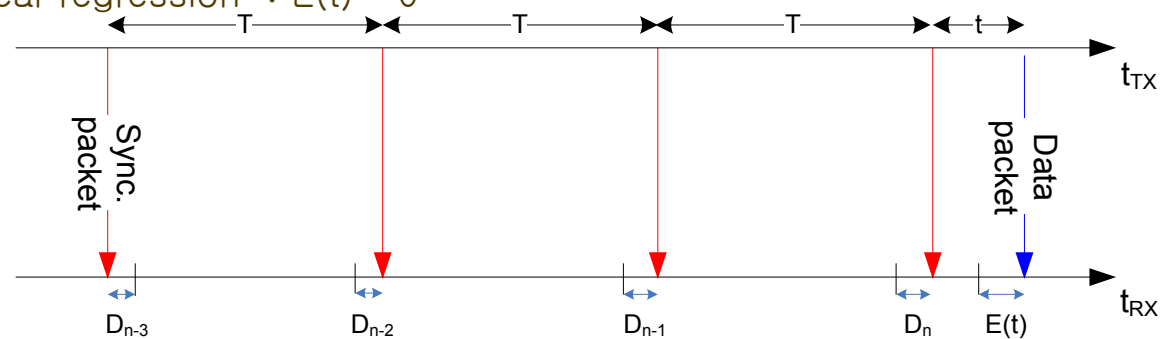


- ◆ Longest synchronization interval
 - ➔ Synchronization when TX packet
 - ➔ **Aperiodic synchronization**

Synchronization

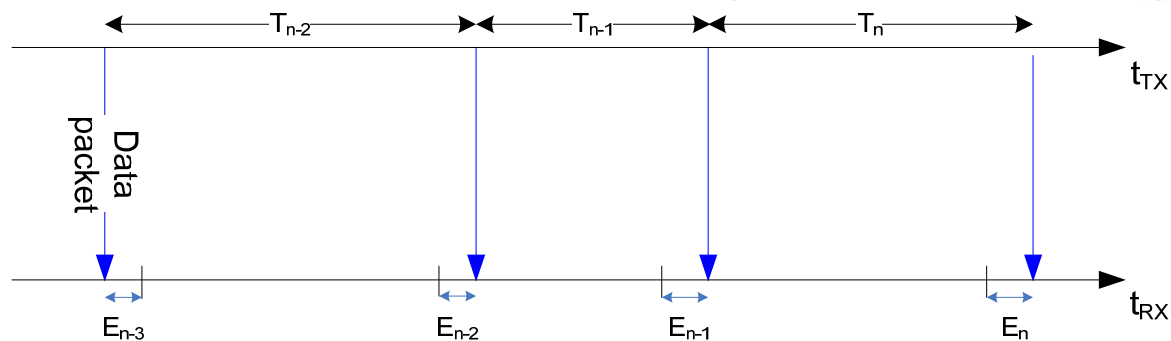
◆ Periodic synchronization

- Periodic data samples (D_k)
 - Easy to estimate RX node's wakeup time
 - e.g) linear regression $\because E(t) \approx 0$



◆ Aperiodic synchronization

- Aperiodic data samples (E_k)
 - e.g.) WiseMAC^[1]
 - Difficult to estimate RX node's wakeup time $\because E_n \gg 0$



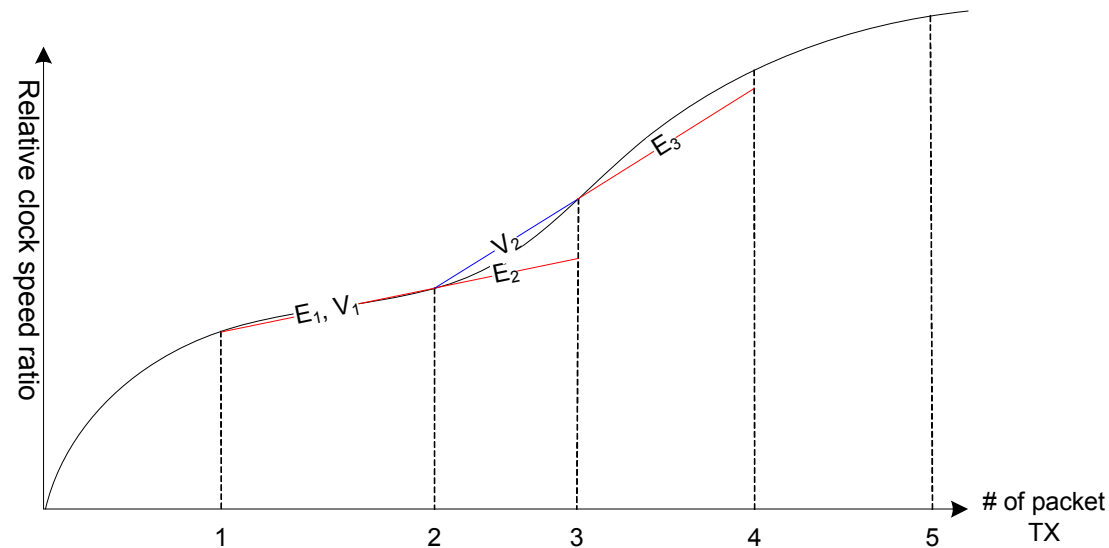
Relative clock speed estimation

- ◆ Relative clock speed estimation algorithm
 - Very insensitive to the current measured relative clock speed because of the communication delay

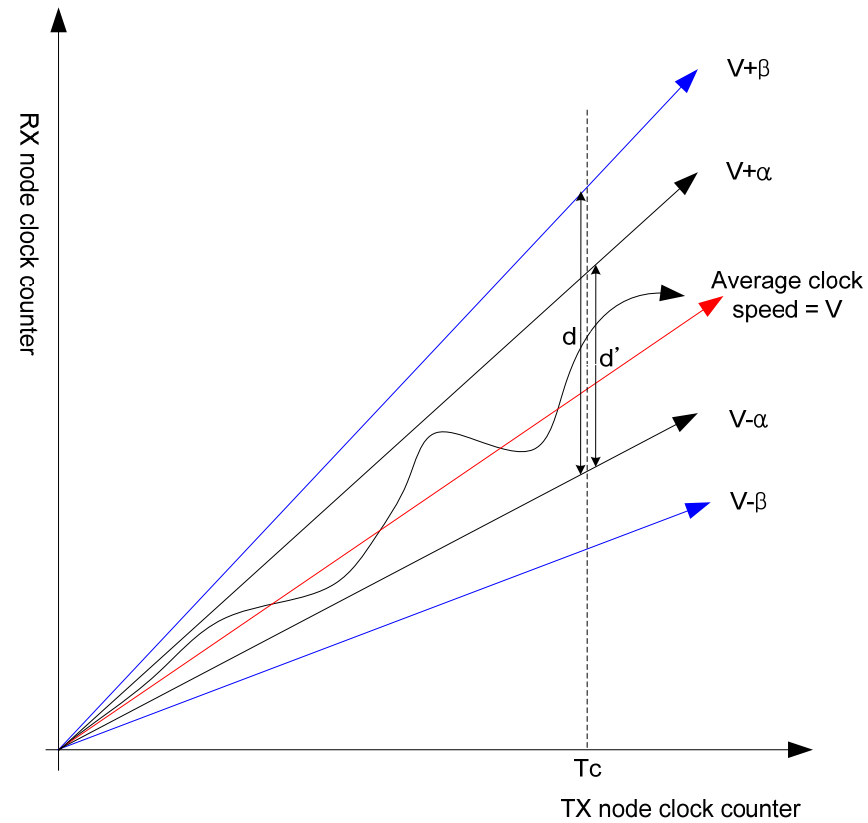
$$E_{n+1} = \alpha V_n + (1 - \alpha) E_n, \quad \alpha = 0.1$$

E_n : n th estimated relative clock speed of RX

V_n : n th measured relative clock speed of RX



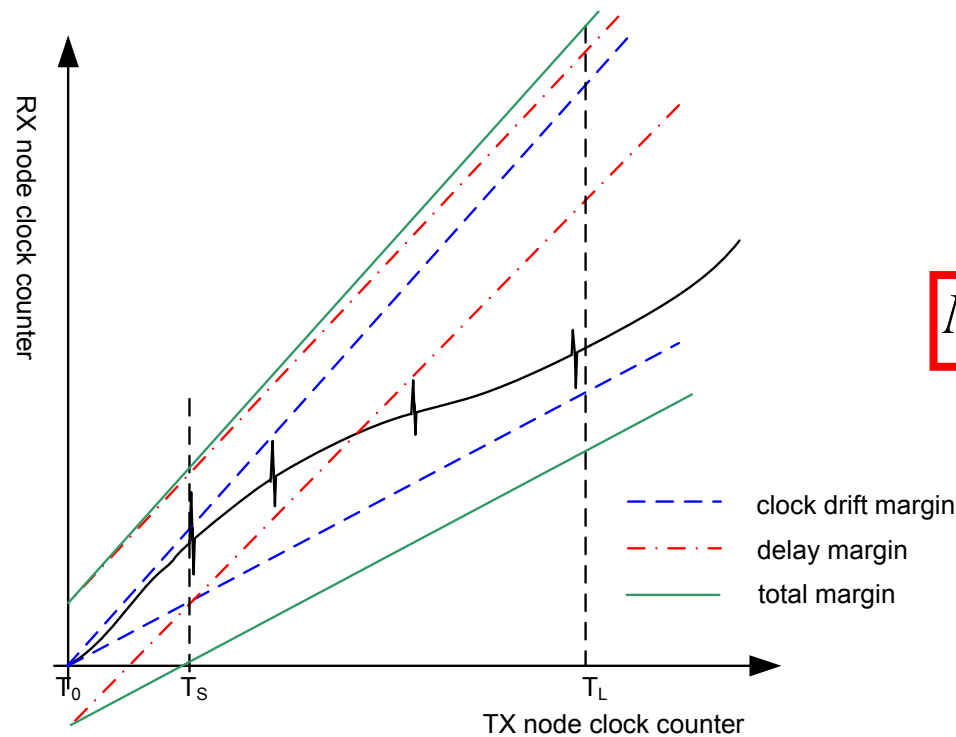
Clock drift margin



$$C_{n+1} = \max\left(C_n, \frac{d_n}{T_n}\right) \cdot \varepsilon$$

Total margin

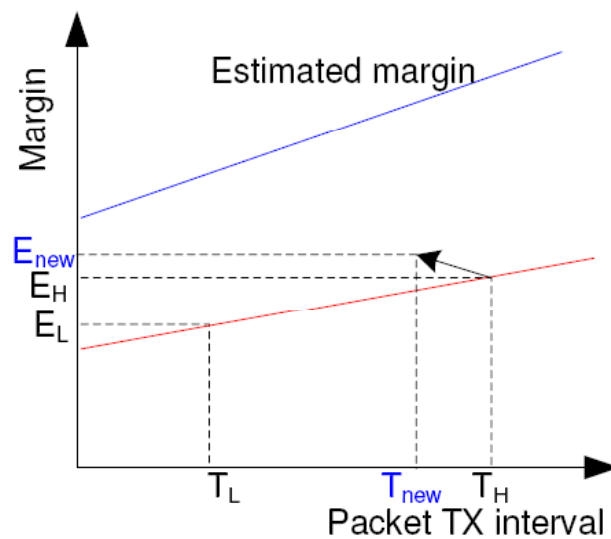
- ◆ Delay margin is effective for short inter-tx time
- ◆ Clock drift margin is effective for long inter-tx time



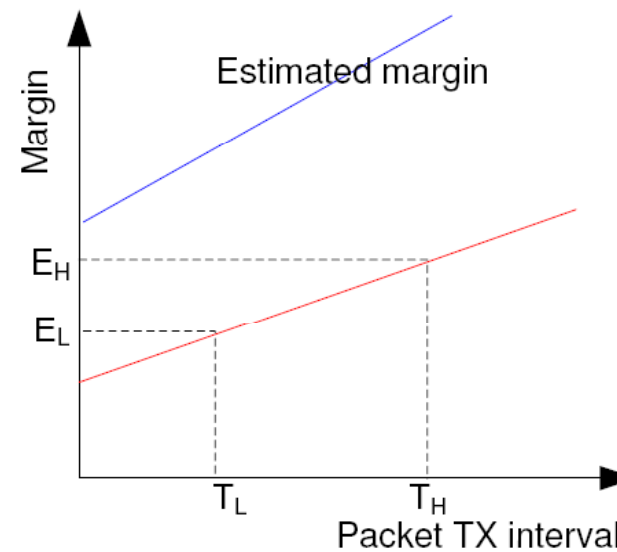
$$M_{total}(T_n) = M_{delay} + M_{clock}$$

Linear margin

- ◆ Uses the data of two different packet TX intervals
- ◆ Can support a wide range of packet TX interval



(a) Before updating margin.






(b) After updating margin

Algorithm

Algorithm 1 : Calculate T_{rx_wakeup} & M_{total}

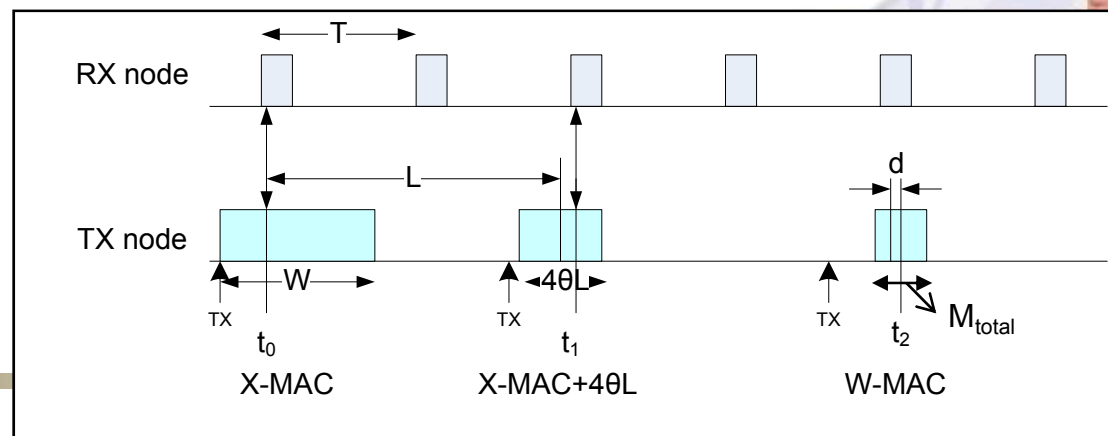
- 1: **if** E_{n+1} is not available
- 2: $E_{n+1} = 1$
- 3: **endif**
- 4: Find min k s.t. $kT \cdot E_{n+1} > T_n$
- 5: $T_{rx_wakeup} = kT \cdot E_{n+1} + \text{previous } T_{rx_wakeup}$
- 6: **if** D_n, C_n is not available
- 7: $M_{total} = 4\theta \cdot kT \cdot E_{n+1}$
- 8: **else**
- 9: $M_{total} = \alpha D_n + \beta C_n \cdot kT \cdot E_{n+1}$
- 10: **endif**

Algorithm 2 : Update estimation model

- 1: 
 - 2: 
 - 3: 
- $$d_{d,n} = d_n \cdot \frac{\frac{T_n}{DER}}{1 + \frac{T_n}{DER}} \quad d_n = d_n \cdot \frac{1}{1 + \frac{T_n}{DE}}$$

Algorithm 3 : Wakeup time estimation

- 1: **if** no information for estimation
- 2: Send preamble packet now during T
- 3: **else**
- 4: Call **Algorithm 1**
- 5: Send preamble packet at $T_{rx_wakeup} - M_{total}$ during $2M_{total}$
- 6: **endif**
- 7: **if** estimation failed
- 8: Initialize estimation model
- 9: **else**
- 10: Call **Algorithm 2**
- 11: **endif**



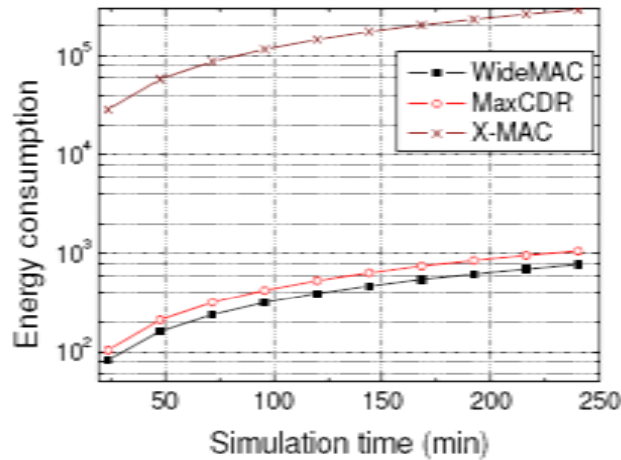
Simulation

- ◆ WideMAC, X-MAC, MaxCDR (X-MAC+4 θ L)
- ◆ Topology: 25 nodes, grid or random
5 sources, 1 sink
- ◆ TX range: 65m, Interference range: 130m

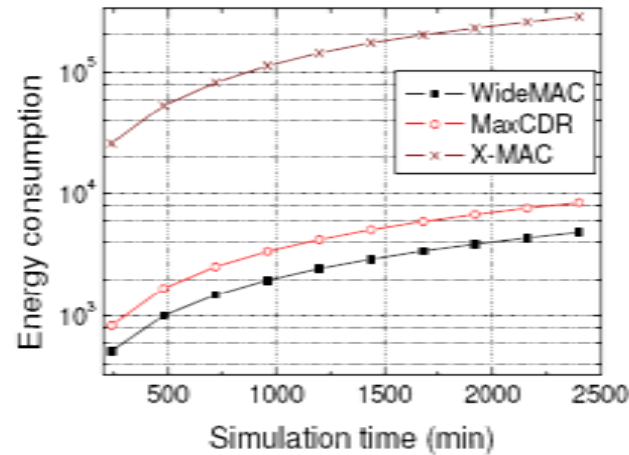
Simulation configuration.

Test Environment Name	Test 1	Test 2	Test 3
Wakeup interval (<i>msec</i>)	0.52	0.52	0.52
Sleep interval (<i>sec</i>)	10	10	300
Packet TX interval (<i>min</i>)	3	30	90
Duty cycle ($\times 10^{-5}$)	5.2	5.2	0.17
Total Sim. Time (<i>hour</i>)	4	40	120

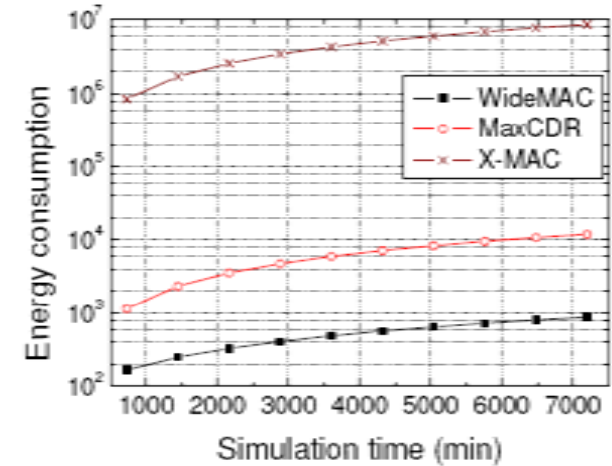
Simulation results (1)



(a) Test 1.

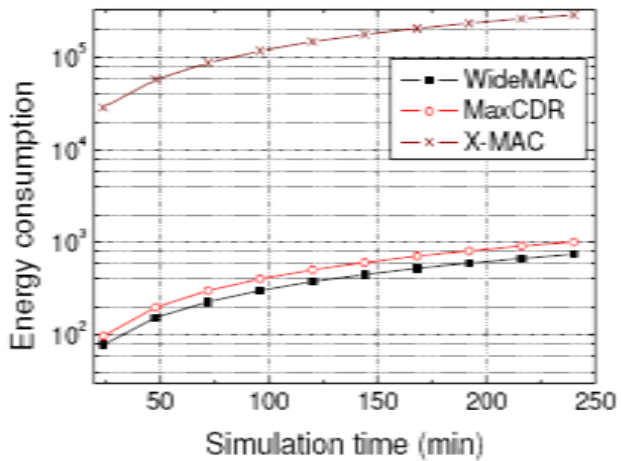


(b) Test 2.

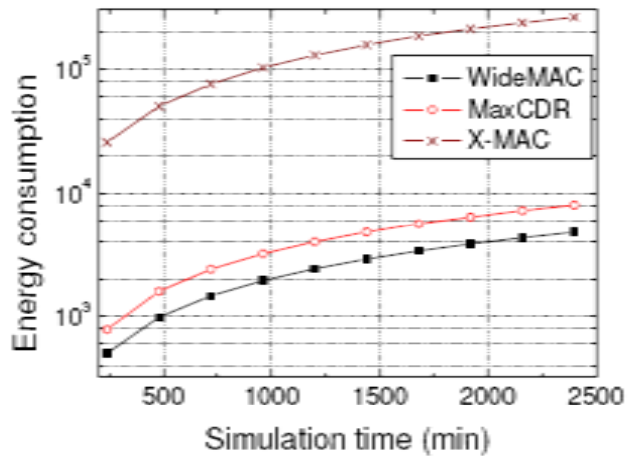


(c) Test 3.

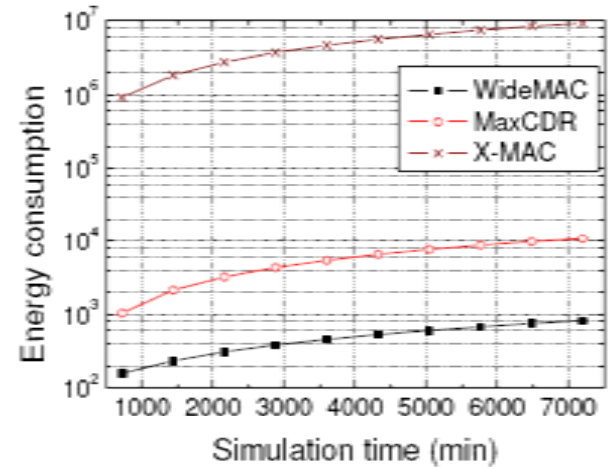
Energy consumption of grid topology.



(a) Test 1.



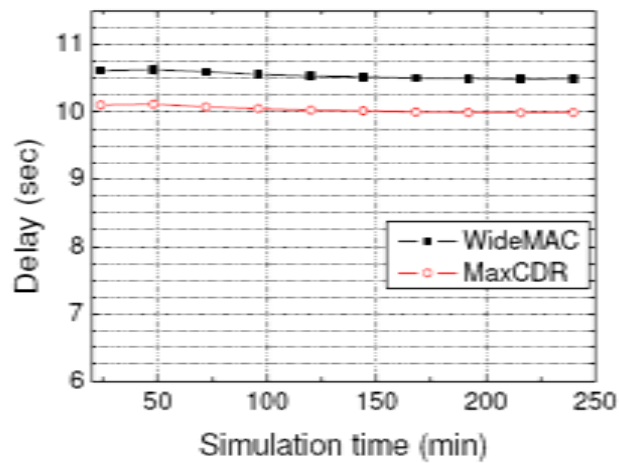
(b) Test 2.



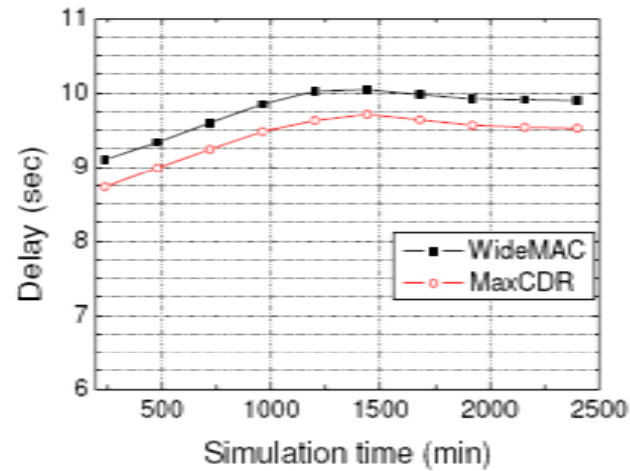
(c) Test 3.

Energy consumption of random topology.

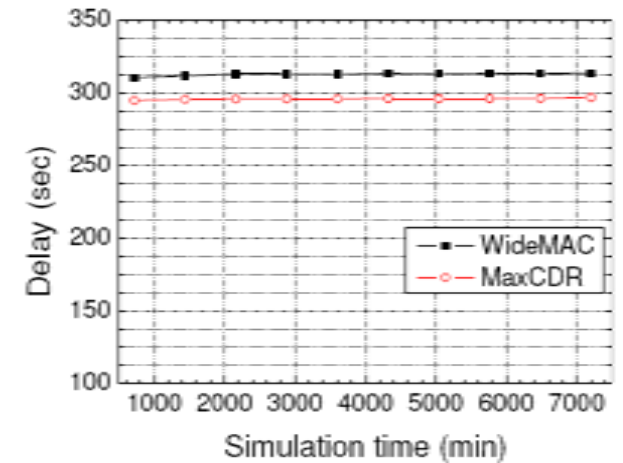
Simulation results (2)



(a) Test 1.

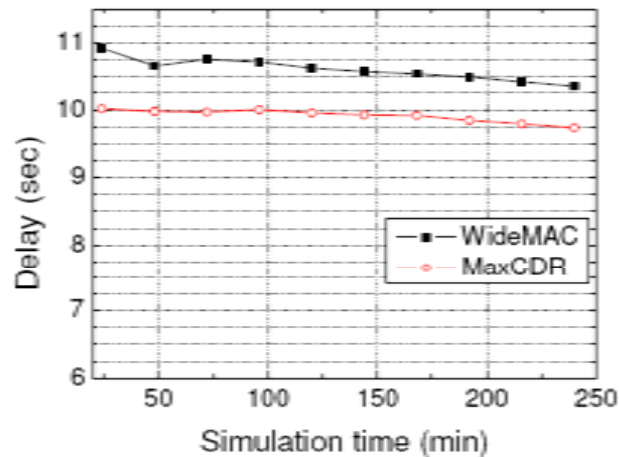


(b) Test 2.

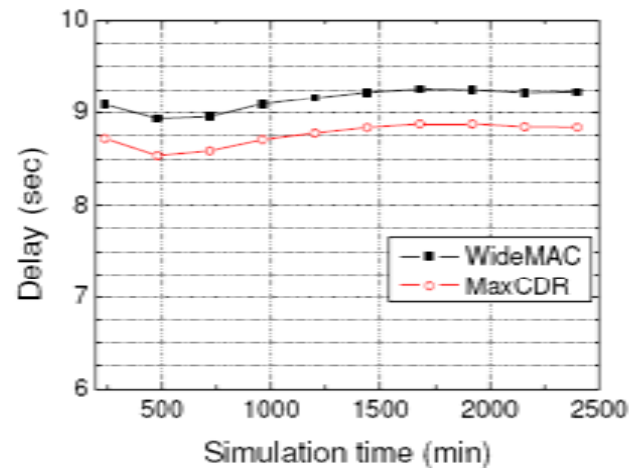


(c) Test 3.

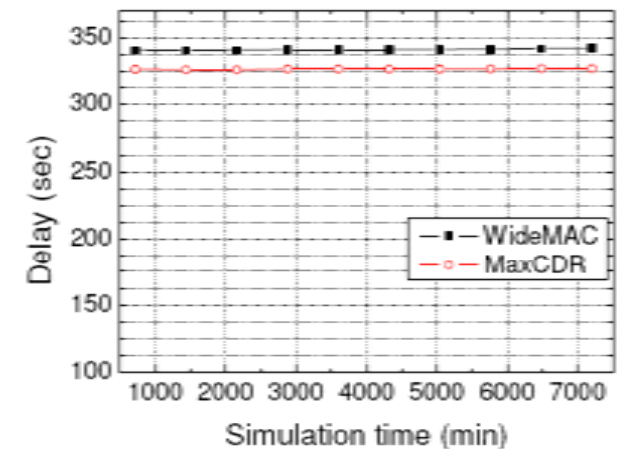
Average end-to-end delay of grid topology.



(a) Test 1.



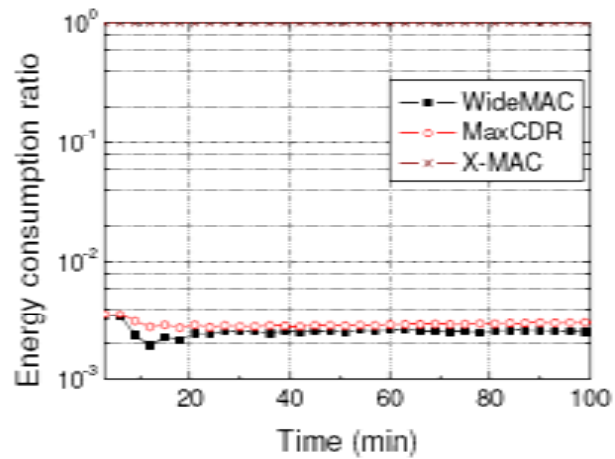
(b) Test 2.



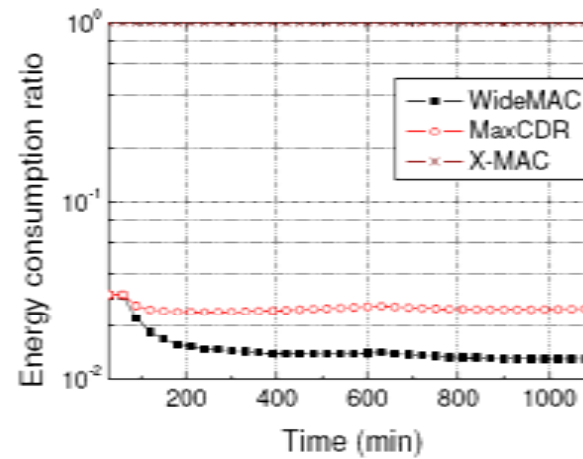
(c) Test 3.

Average end-to-end delay of random topology.

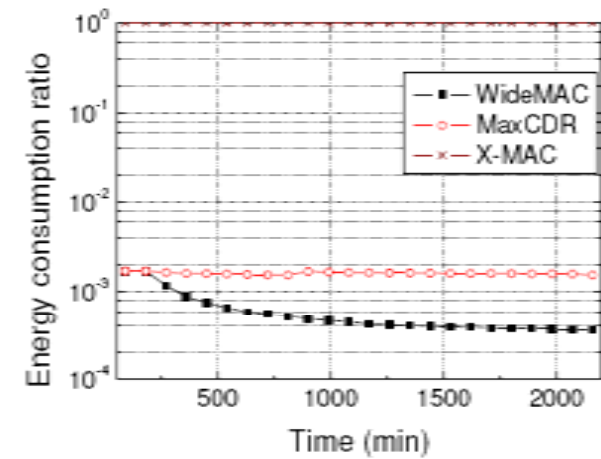
Simulation results (3)



(a) Test 1.



(b) Test 2.



(c) Test 3.

Energy consumption of testbed.

Conclusion

- ◆ Clock drift measurement according to various TX intervals
- ◆ WideMAC supports an ultra low duty cycled network thanks to the clock drift estimation algorithm
- ◆ It achieves a longer network lifetime by ultra low TX power consumption
- ◆ Reliable communication

