


CS-541
Wireless Sensor Networks

Lecture 6: Radio Duty Cycling for Wireless Sensor Networks

Spring Semester 2017 -2018

Prof Panagiotis Tsakalides, Dr Athanasia Panousopoulou, Dr Gregory Tsagkatakis

Objectives

- Why do we need duty cycling

- Categorization of RDC techniques: synchronous – asynchronous

Why do we need duty cycling for WSN

- Sensor Nodes states: **TX, RX, Idle**, Sleep
- Idle Listening: radio is on, channel scanning, but no tx-rx activity.
- Power consumed during idle listening depends on the transceiver / hardware platform characteristics.

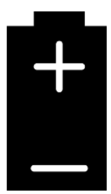
State	Measurements (CC2480+ MSP430)	TI (chipcon) CC2420 [33]	TI CC2480 [34]	TI CC2520 [35]	Ember EM260 [36]	Jennic JN5121 [37]	Freescale MC1322x [38]
Operating Voltage (V)	3.6 V	2.1–3.6 V	2.0–3.6 V	1.8–3.8 V	2.7–3.6 V	2.1–3.6 V	2.0–3.6 V
Output power range (dBm)	0 dBm	−24 to 0 dBm	0 dBm	Up to 5 dBm	0 dBm	−32 to 4.5 dBm	−30 to 5 dBm
Sleep mode	0.75 µA	1 µA (max)	0.5 µA	0.12 µA (max.)	3.5 µA	1 µA	0.4 µA
Transmission	30.5 mA	17.4 mA	27 mA	25.8 mA	49 mA	28–30 mA	29 mA
Channel Listening	32.5 mA	18.8 mA	27 mA	22.3 mA	44 mA	28–34 mA	21 mA

Casilari, Eduardo; Cano-García, Jose M.; Campos-Garrido, Gonzalo. 2010. "Modeling of Current Consumption in 802.15.4/ZigBee Sensor Motes." Sensors 10, no. 6: 5443-5468.

We do we need duty cycling for WSN

- On idle listening: **Energy loss especially on low-traffic conditions.**
- Network becomes dependent on the capacity of the battery and the quality of the battery cells (depth of discharge and how fast it discharges)
- Regardless of the network traffic and sensor nodes activity (e.g. relays or not) **lifetime is the same**

AA (1.5x2) batteries: ~6 days of lifetime



vs.

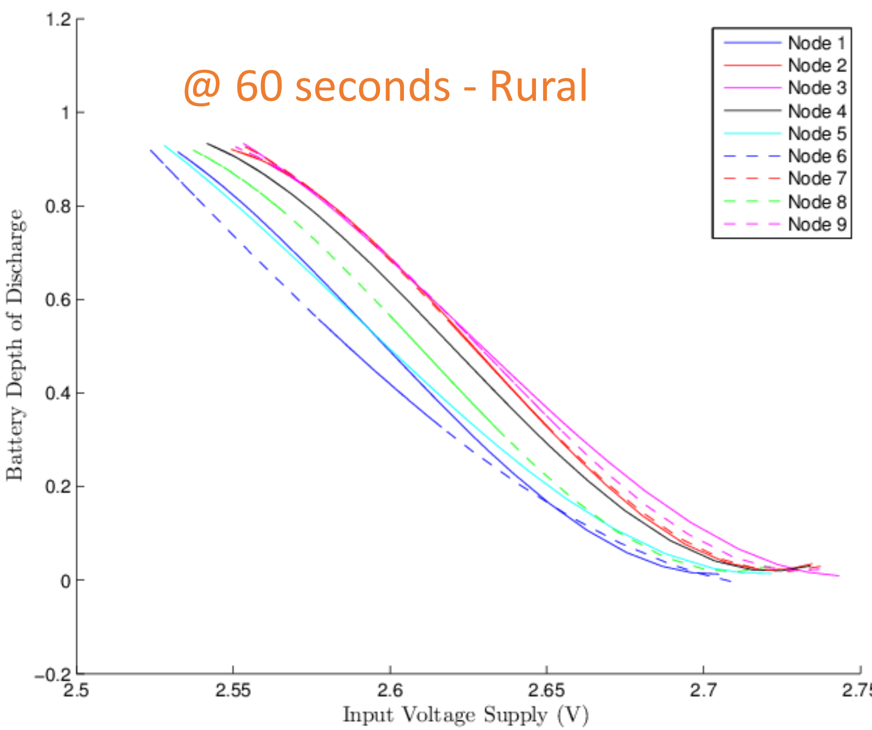
12 V batteries: ~3 weeks of lifetime (GG Bridge project).

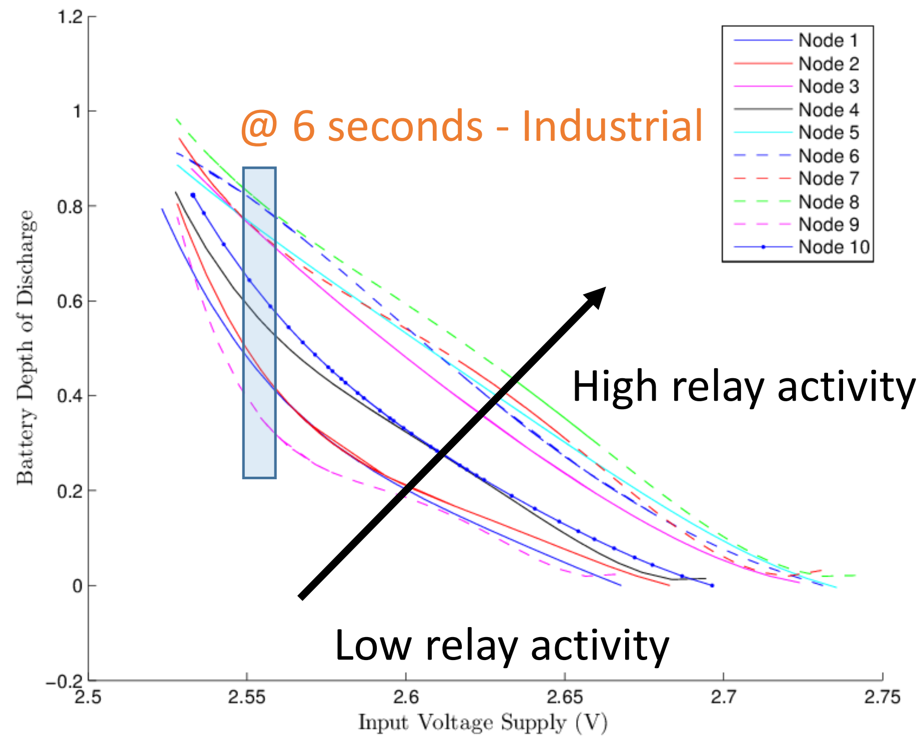
Hydrobionets

We do we need duty cycling for WSN

- Example: 2 small-scale testbeds (10-15 nodes), same protocol stack, operating at different environments (rural and industrial) and having different traffic demands

Empirical characterization of discharge of AA batteries

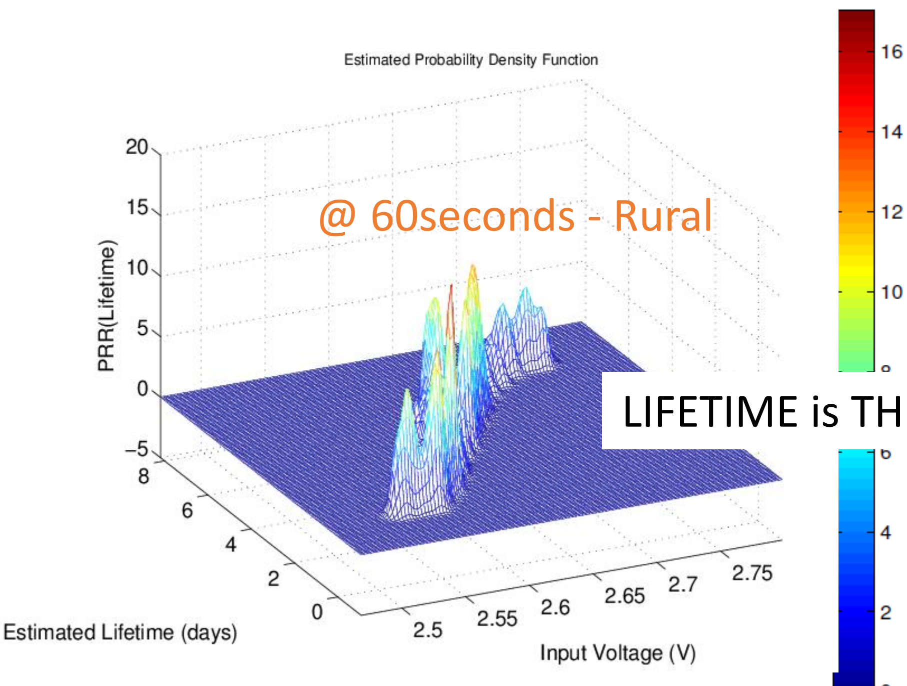




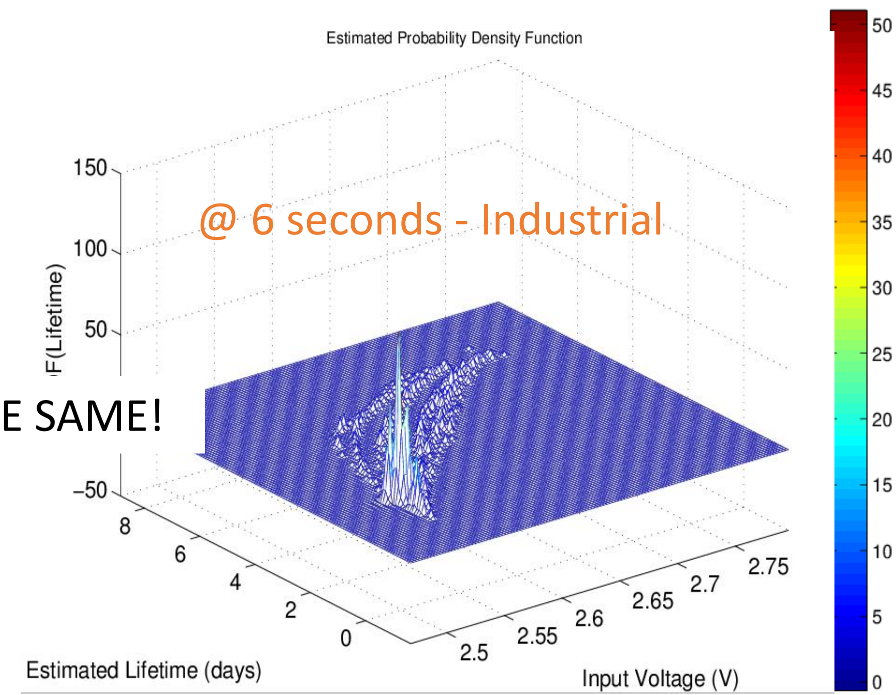
We do we need duty cycling for WSN

- Example: 2 small-scale testbeds (10-15 nodes), same protocol stack, operating at different environments (rural and industrial) and having different traffic demands

Empirical characterization of lifetime (Available Energy / Consumed Power)



LIFETIME is THE SAME!

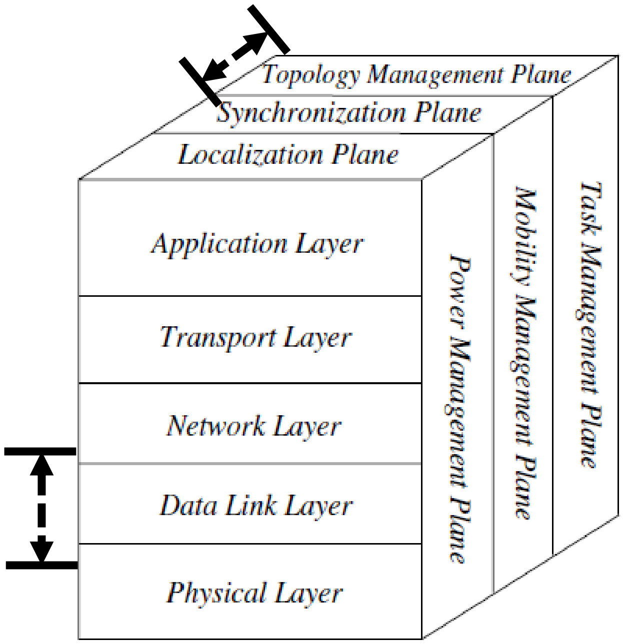


Duty Cycling

- **Radio Duty Cycling** is an energy-saving technique for reducing the amount of energy consumed during **idle listening**.

- Radio Duty Cycling corresponds to scheduling sensor nodes between active (tx, rx, idle) and sleep operational modes

- Challenges:
  - How to cope with extra latency (sampling on sleep mode -> queued until active period
  - Relay node wakes up to forward the packet – packet is lost or delayed
  - Communication connectivity – network partitioning



"Wireless Sensor Networks", Ian F. Akyildiz, Mehmet Can Vuran, 2010, Willey, Ch 1

Duty Cycling Categorization

Synchronous

- **Synchronization between sensor nodes to accommodate the same sleep/active schedule (how do we do that?)**
- **All nodes share the same clock (not the same as TDMA – they can still compete for accessing the medium)**
- **Communication is always feasible during the active periods**
- **Link disconnections are avoided**
- **Not efficient for not fully connected networks or scalable multi-hop topologies (latency)**
- **Duty cycle is selected before network deployment - how to deal with traffic fluctuations**

Asynchronous

- **No need for synchronization between sensor nodes**
- **Each node preserve its own clock and independent active/sleep periods**
- **More resilient to network dynamics**
- **More appropriate for large-scale networks & best effort traffic**
- **Communication might not be feasible**
- **Nodes may suffer from intense interference due to excessive retransmissions**

Synchronous Duty Cycling

- If data sampled by a source node during its sleep period have to be queued until the active period
- A relay node has to wait until the next hop wakes up to forward the packet

Approaches:

A. Interference is not considered as a problem

B. Interference aspects are addressed as is part of the solution – resulting to a contention-based MAC protocol

Synchronous Duty Cycling

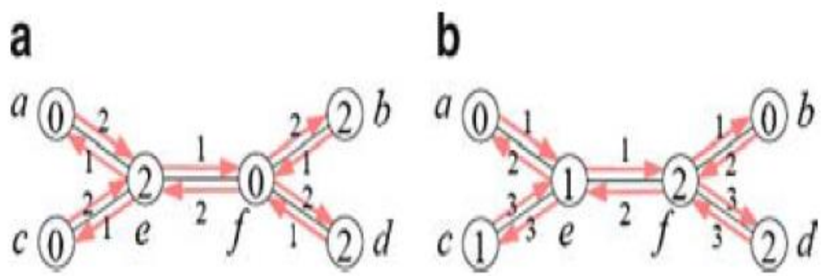
A. Interference is not considered part of the problem

- Time is slotted and interference can be resolved within each slot (contention-based)
- Goal: small communication delay between any pair of nodes.

“Guide to Wireless Sensor Networks”, S. Misra, I. Woungang, S. C. Misra, 2009, Springer, Ch 15

G: (V, E)

Each sensor node is active in exactly one of *k* slot, even if it does not have data to transmit



slot assignment function ***f(u): V(G) -> [0, k-1]*** :

- Each sensor node *u* is active only in slots *i*k + f(u), i = 0,1,2,...*
- If a sensor node has a packet to forward to a neighbor, it can wake up in the active slot of **that neighbor** and transmit the packet.

Synchronous Duty Cycling

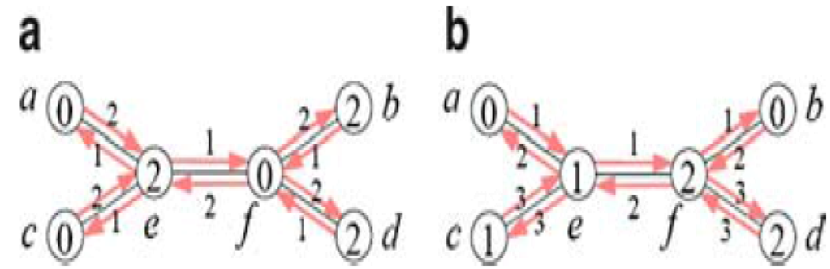
A. Interference is not considered part of the problem

- Time is slotted and interference can be resolved within each slot (contention-based)
- Goal: small communication delay between any pair of nodes.

G(V) = (V, E)

Example (a): $f(u) = \begin{cases} 0, if\ u\ \in\ \{a, c, f\} \\ 2\ if\ u\ \in\ \{e, d, b\} \end{cases}$

Example (b): $f(u) = \begin{cases} 0, if\ u\ \in\ \{a, b\} \\ 1, if\ u\ \in\ \{e, c\} \\ 2\ if\ u\ \in\ \{d, f\} \end{cases}$



“Guide to Wireless Sensor Networks”, S. Misra, I. Woungang, S. C. Misra, 2009, Springer, Ch 15

Synchronous Duty Cycling

A. Interference is not considered part of the problem

- This means that: (a) time is slotted and interference can be resolved within each slot (contention-based)
- Goal: small communication delay between any pair of nodes.

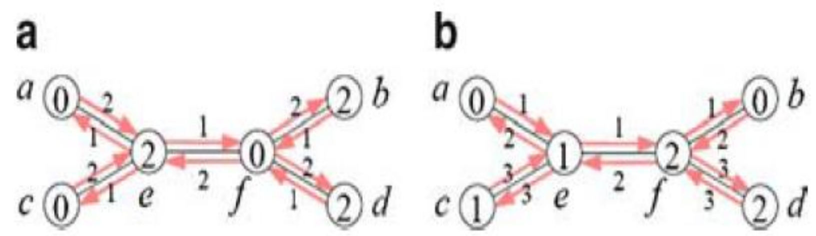
Delay between two nodes **u↔v** :

$$d_G^f(u,v) = \begin{cases} k & \text{if } f(u) = f(v) \\ (f(v) - f(u)) \mod k & \text{otherwise} \end{cases}$$

Can you think of why?

"Guide to Wireless Sensor Networks", S. Misra, I. Woungang, S. C. Misra, 2009, Springer, Ch 15

G(V) = (V, E)



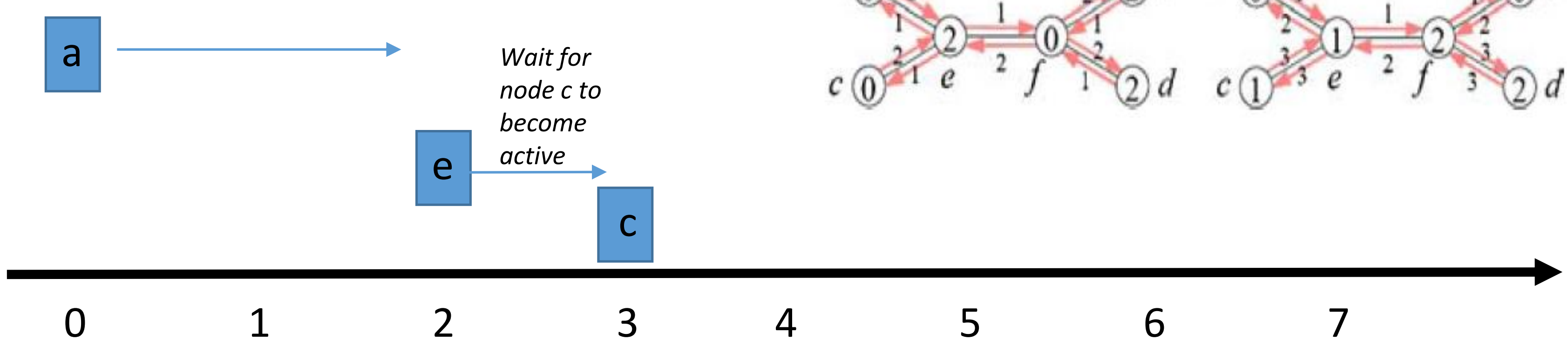
Synchronous Duty Cycling

A. Interference is not considered part of the problem

- This means that: (a) time is slotted and interference can be resolved within each slot (contention-based)
- Goal: small communication delay between any pair of nodes.

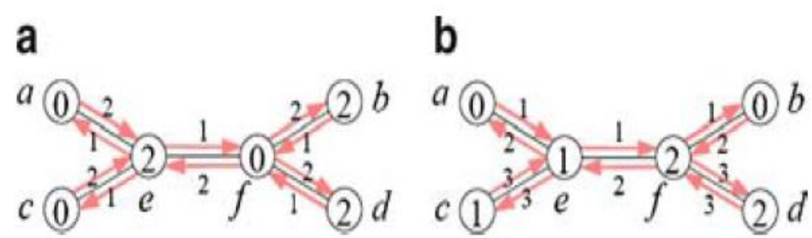
Delay between two nodes **u↔v** :

$$d_G^f(u,v) = \begin{cases} k & \text{if } f(u) = f(v) \\ (f(v) - f(u)) \mod k & \text{otherwise} \end{cases}$$



"Guide to Wireless Sensor Networks", S. Misra, I. Woungang, S. C. Misra, 2009, Springer, Ch 15

G(V) = (V, E)



Synchronous Duty Cycling

A. Interference is not considered part of the problem

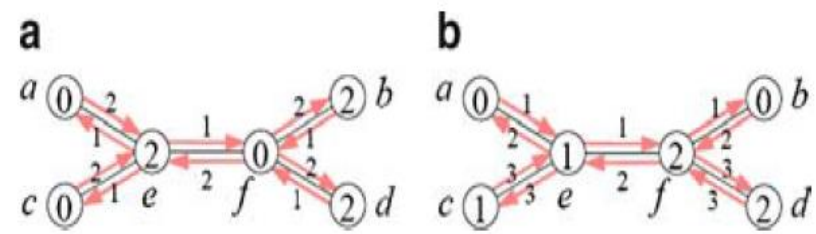
- This means that: (a) time is slotted and interference can be resolved within each slot (contention-based)
- Goal: small communication delay between any pair of nodes.

Delay between two nodes u ↔ v:

“Guide to Wireless Sensor Networks”, S. Misra, I. Woungang, S. C. Misra, 2009, Springer, Ch 15

G(V) = (V, E)

$$d_G^f(u,v) = \begin{cases} k & \text{if } f(u) = f(v) \\ (f(v) - f(u)) \mod k & \text{otherwise} \end{cases}$$



Delay along a path P:

$$\sum_{u \to v \in P} d_G^f(u,v).$$

a

a-> b: ??

b

a->b ??

Synchronous Duty Cycling

A. Interference is not considered part of the problem

- This means that: (a) time is slotted and interference can be resolved within each slot (contention-based)
- Goal: small communication delay between any pair of nodes.

Delay between two nodes u ↔ v:

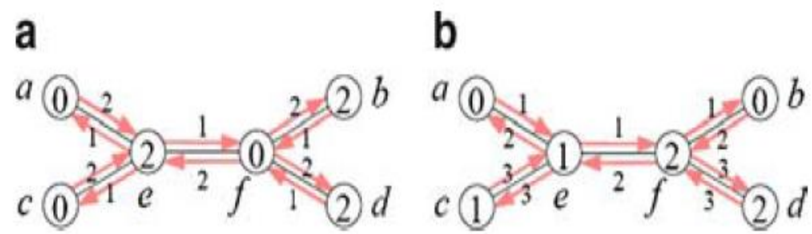
$$d_G^f(u,v) = \begin{cases} k & \text{if } f(u) = f(v) \\ (f(v) - f(u)) \mod k & \text{otherwise} \end{cases}$$

Delay along a path P:

$$\sum_{u \to v \in P} d_G^f(u,v).$$

“Guide to Wireless Sensor Networks”, S. Misra, I. Woungang, S. C. Misra, 2009, Springer, Ch 15

G(V) = (V, E)



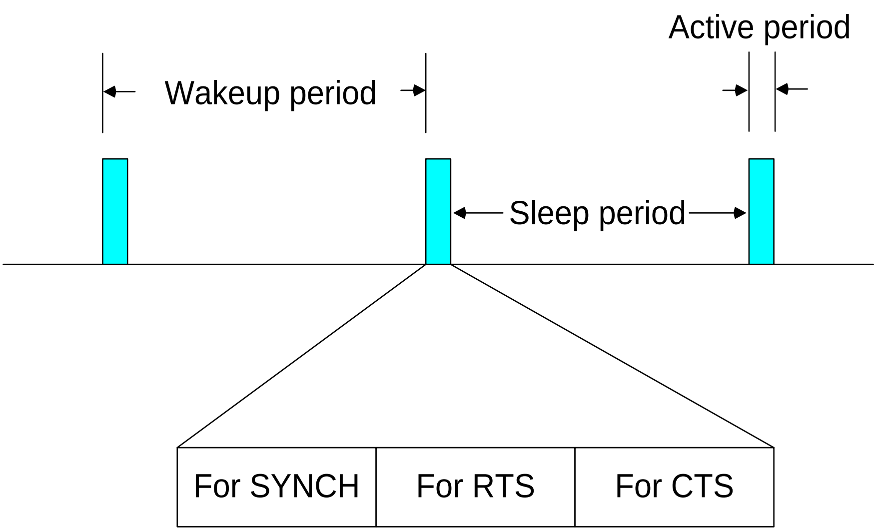
Delay diameter of a duty cycled network is defined as the largest delay distance between any two nodes

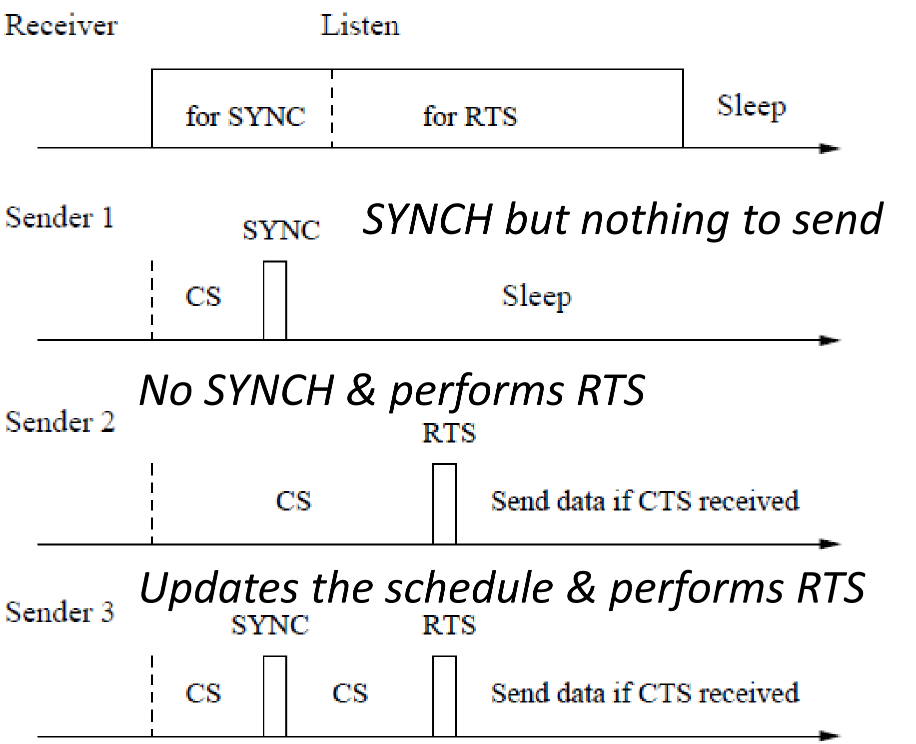
Synchronous Duty Cycling

B. Interference aspects are addressed as is part of the solution – resulting to a contention-based MAC protocol

- **S-MAC**: exploits the synch-based RTS/CTS scheme

Wei Ye; Heidemann, J.; Estrin, D., "An energy-efficient MAC protocol for wireless sensor networks," INFOCOM 2002.





Synchronous Duty Cycling

B. Interference aspects are addressed as is part of the solution – resulting to a contention-based MAC protocol

- **S-MAC**: exploits the synch-based RTS/CTS scheme

How to go from SYNCH packets to synchronized nodes: Creation of virtual clusters w.r.t. to wakeup schedule

- First-come, first served: Nodes try to follow a schedule from 1-hop neighbors.
- If not found, they start their own and let other nodes to follow
- Nodes having "followers" and pick up a new schedule, have to adopt both schedules.
- Time-synchronization packet ~per 15 s

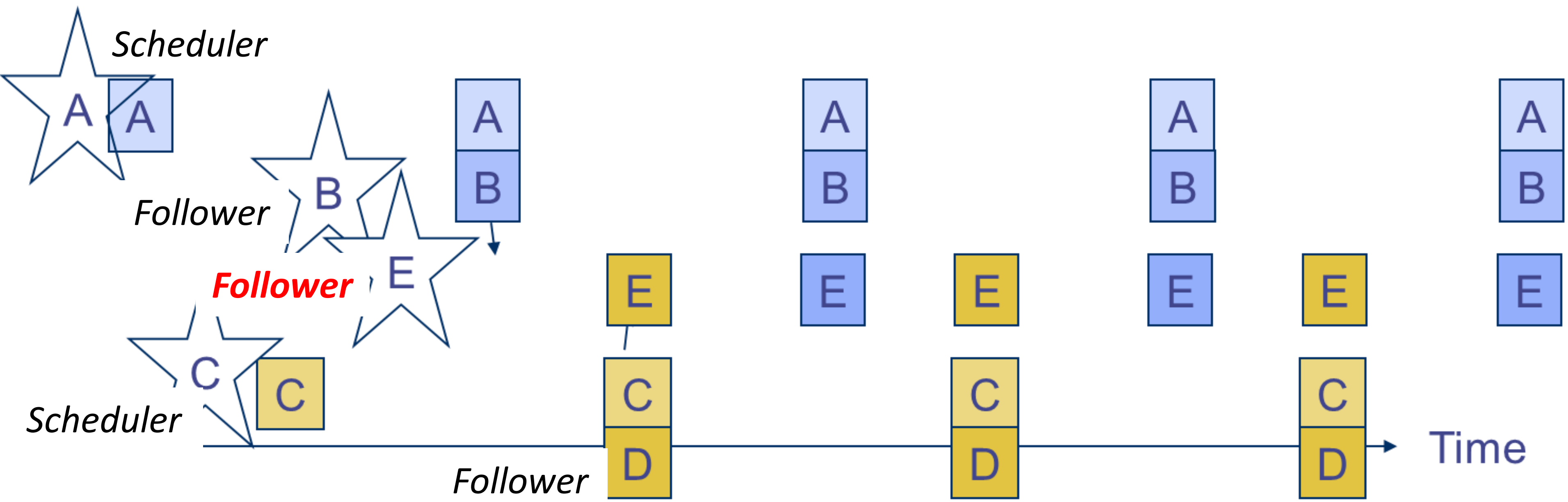
Synchronous Duty Cycling

B. Interference aspects are addressed as is part of the solution – resulting to a contention-based MAC protocol

- **S-MAC**: exploits the synch-based RTS/CTS scheme

Holger Karl, Andreas Willig, Protocols and Architectures for Wireless Sensor Systems, 2005, Willey, Ch. 5

Multi-hop topologies – some nodes will have to follow more than one schedules



Synchronous Duty Cycling

B. Interference aspects are addressed as is part of the solution – resulting to a contention-based MAC protocol

- **S-MAC**

@ large scale, multi-hop networks: nodes following more than one schedule consume more energy

@ all nodes on the same schedule: per-hop latency equals to the sleep period

Dynamic traffic: nodes don't change their wake-up / sleep intervals – if the traffic becomes lighter nodes will spend time in idle mode

Synchronous Duty Cycling

B. Interference aspects are addressed as is part of the solution – resulting to a contention-based MAC protocol

- **T-MAC**

An improvement of S-MAC - Retaining the creation of virtual clusters

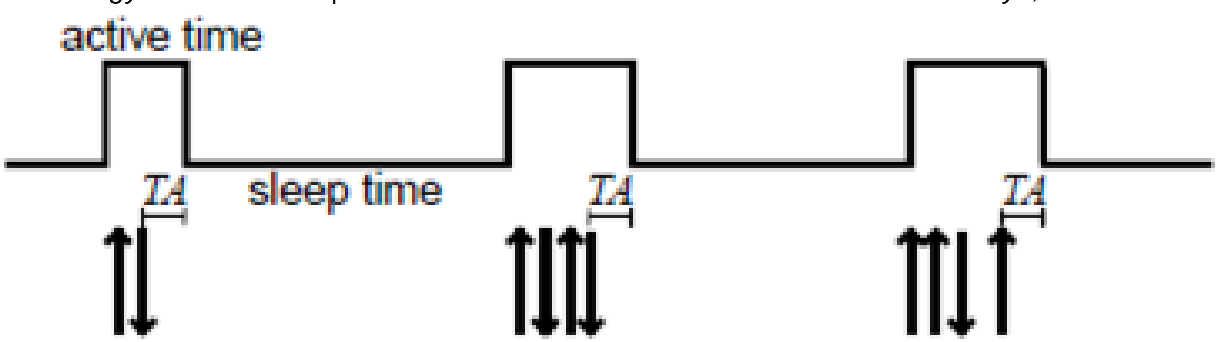
Adaptive duty cycling: adaptive duration of active (& sleep) time.

Burst all messages during the active time –

Go back to sleep mode when no traffic has happened for a certain time (TA)

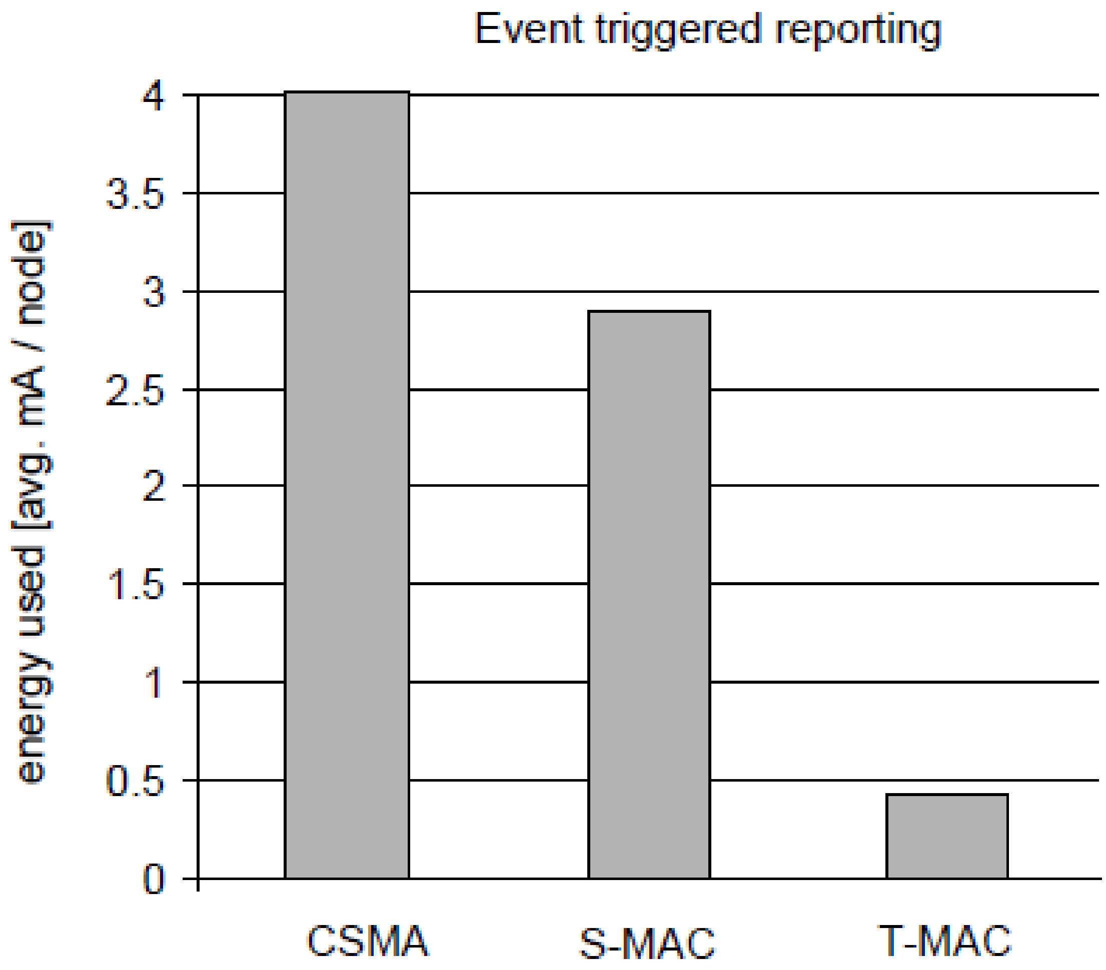
T. van Dam and K. Langendoen. An adaptive energy-efficient MAC protocol for wireless sensor networks. In ACM Sensys, 2003.

TA determines the minimal amount of idle listening per frame.



TA > fixed contention time + length of RTS + turnaround time

Synchronous Duty Cycling



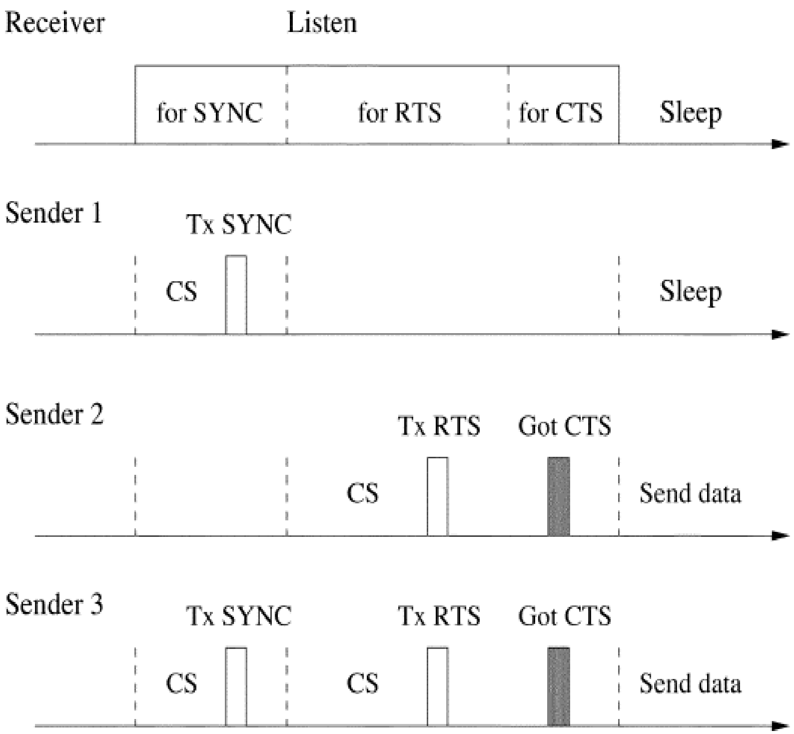
T. van Dam and K. Langendoen. An adaptive energy-efficient MAC protocol for wireless sensor networks. In ACM Sensys, 2003.

Synchronous Cycling

B. Interference aspects are addressed as is part of the solution – resulting to a contention-based MAC protocol

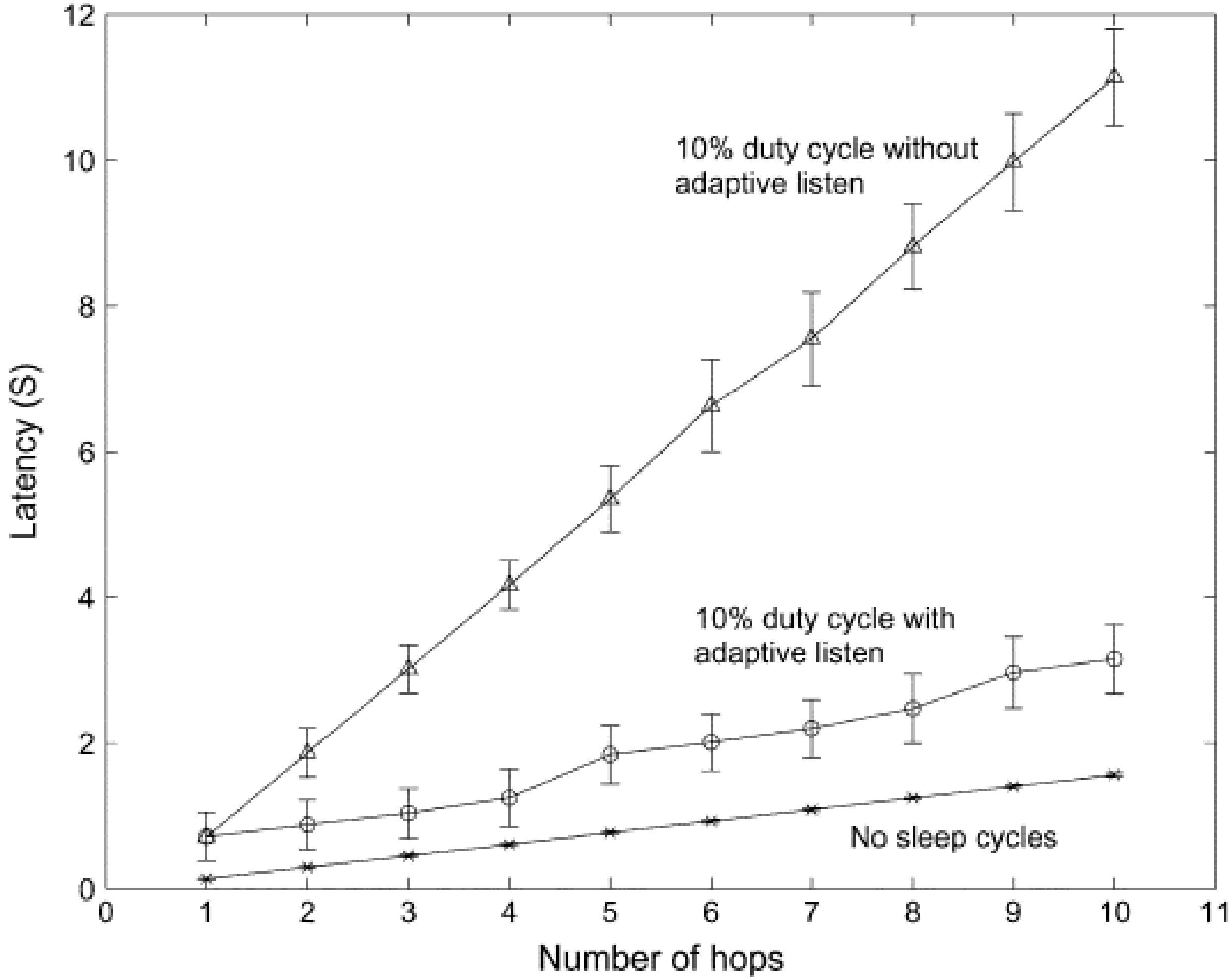
- **Adaptive listening** – to improve delay over hop transmissions (S-MAC)

- Nodes overhearing neighbors' transmissions stay awake for a short period of time at the end of the transmission.
- If they are the next-hop relays then they are able to immediately pass the data to it instead of waiting for its scheduled listen time.
- If the node does not receive anything during the adaptive listening, it will go back to sleep until its next scheduled listen time.



W. Ye, J. Heidemann, and D. Estrin. Medium access control with coordinated adaptive sleeping for wireless sensor networks. IEEE/ACM Transactions on Networks, 12(3):493–506, 2004.

Synchronous Cycling

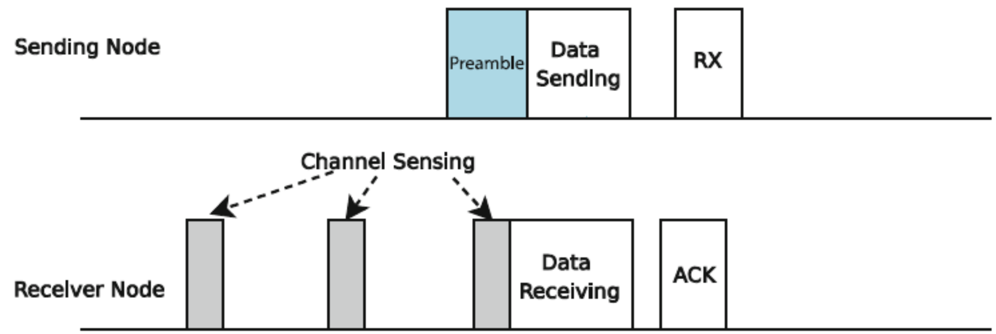


if not combined with the routing policy, more energy cost due to extended activation nodes that lie one or more steps ahead in the routing path stay awake for additional time.

W. Ye, J. Heidemann, and D. Estrin. Medium access control with coordinated adaptive sleeping for wireless sensor networks. IEEE/ACM Transactions on Networks, 12(3):493–506, 2004.

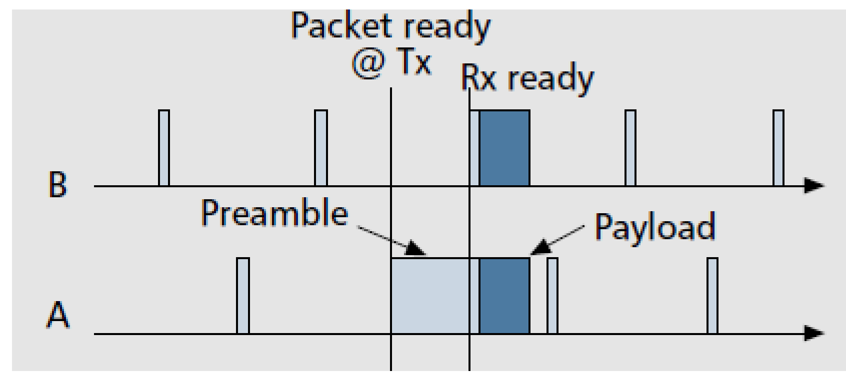
Asynchronous Duty Cycling

- Preamble Sampling



The preamble transmission time should be at least equal to the duty-cycling period!!

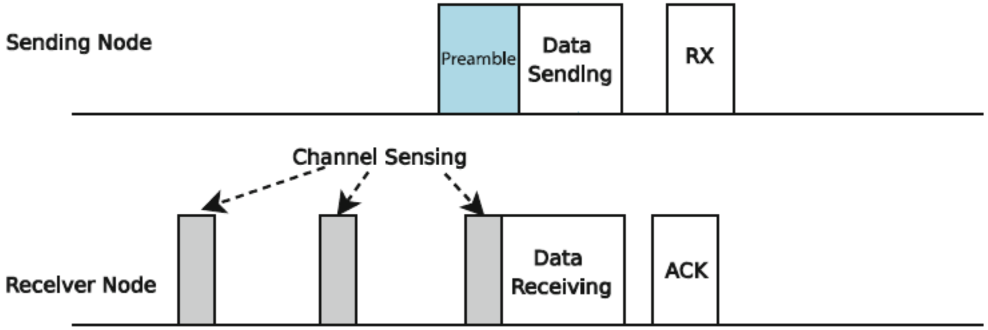
Typical preamble sizes implemented >> Theoretical values



Adapting preamble duration with respect to traffic conditions:
Light traffic: longer preambles
Heavy traffic: short preambles

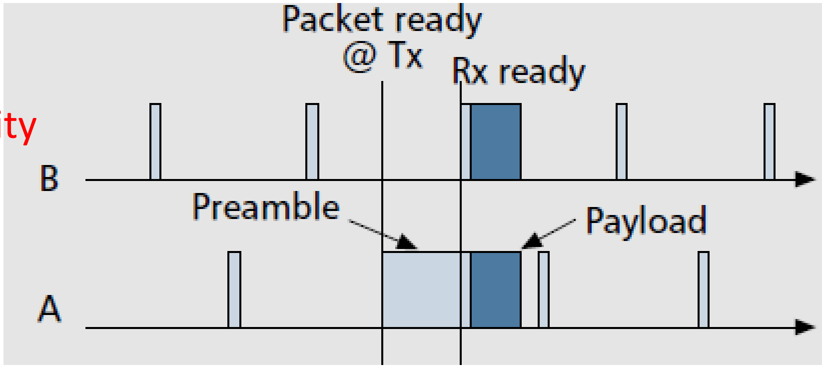
Raghunathan, V.; Ganeriwal, S.; Srivastava, M., "Emerging techniques for long lived wireless sensor networks," Communications Magazine, IEEE , vol.44, no.4, pp.108,114, April 2006

Asynchronous Duty Cycling



- Preamble Sampling

The preamble transmission
time should be at least equal to the
duty-cycling period!!
Typical preamble sizes implemented >> Theoretical values



- Long preambles decrease the effective channel capacity
- Increase the receiving / listening overhead.

Overhearing nodes will have to be awake for

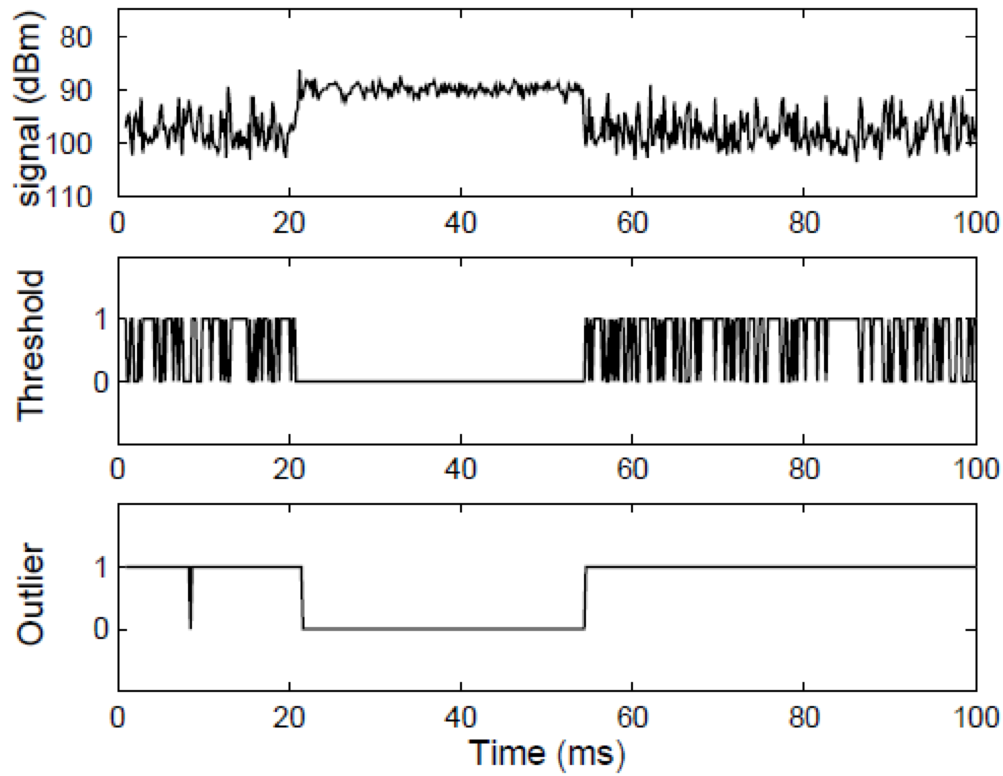
half of the preamble transmission time before receiving the destination address information in the packet header and going back to sleep if they are not the intended receivers.

Raghunathan, V.; Ganeriwal, S.; Srivastava, M., "Emerging techniques for long lived wireless sensor networks," Communications Magazine, IEEE , vol.44, no.4, pp.108,114, April 2006

Asynchronous Duty Cycling

B-MAC

- Combines Preamble sampling (Low Power Listening) with CCA
- No synch phase, no RTS/CTS (default)
- CCA: for accessing the medium and for determining when the channel is active during LPL
- Preamble length = channel checking period for ensuring successful RX.
  - *Typical values: 10 – 1600 ms*
- Node goes back to sleep after successfu packet reception or time out period.
- Interval between LPL samples is maximized so that the time spent sampling the channel is minimized.



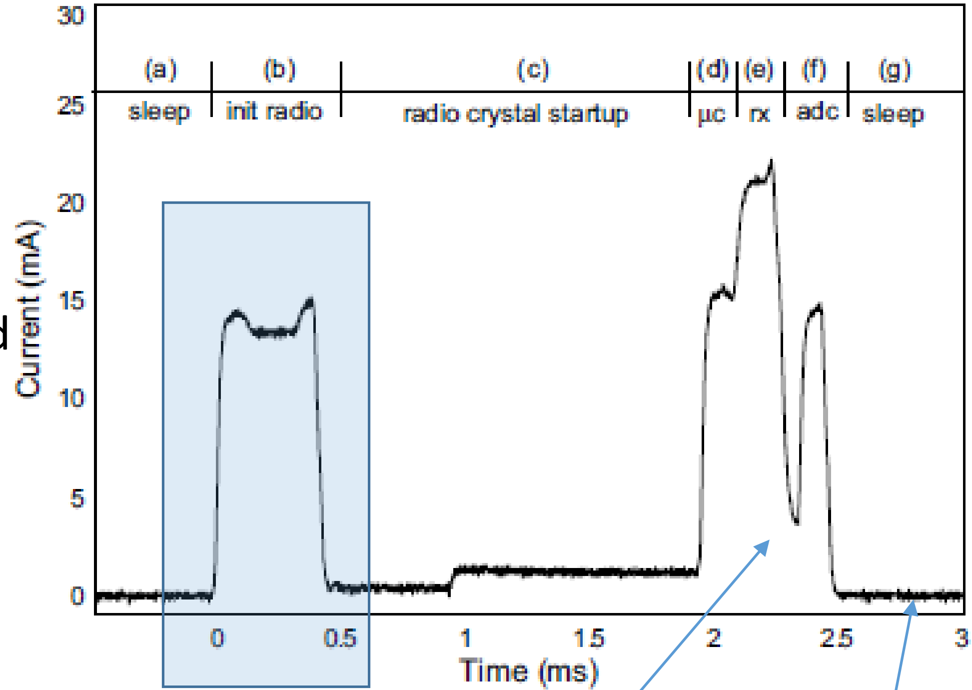
Polastre, Joseph, Jason Hill, and David Culler. "Versatile low power media access for wireless sensor networks." SENSYS, ACM, 2004.

Asynchronous Duty Cycling

B-MAC

TRX cylces Vs Power consumption

- Initial configuration
- Starts the radio and its oscillator
- Switch the radio to receive mode (d) and perform the actions of the protocol.



ADC starts acquisition and analysis of the incoming signal

Going back to sleep if no activity (LPL mode)

The cost for powering up the radio is the same for all protocols.

What makes the difference:

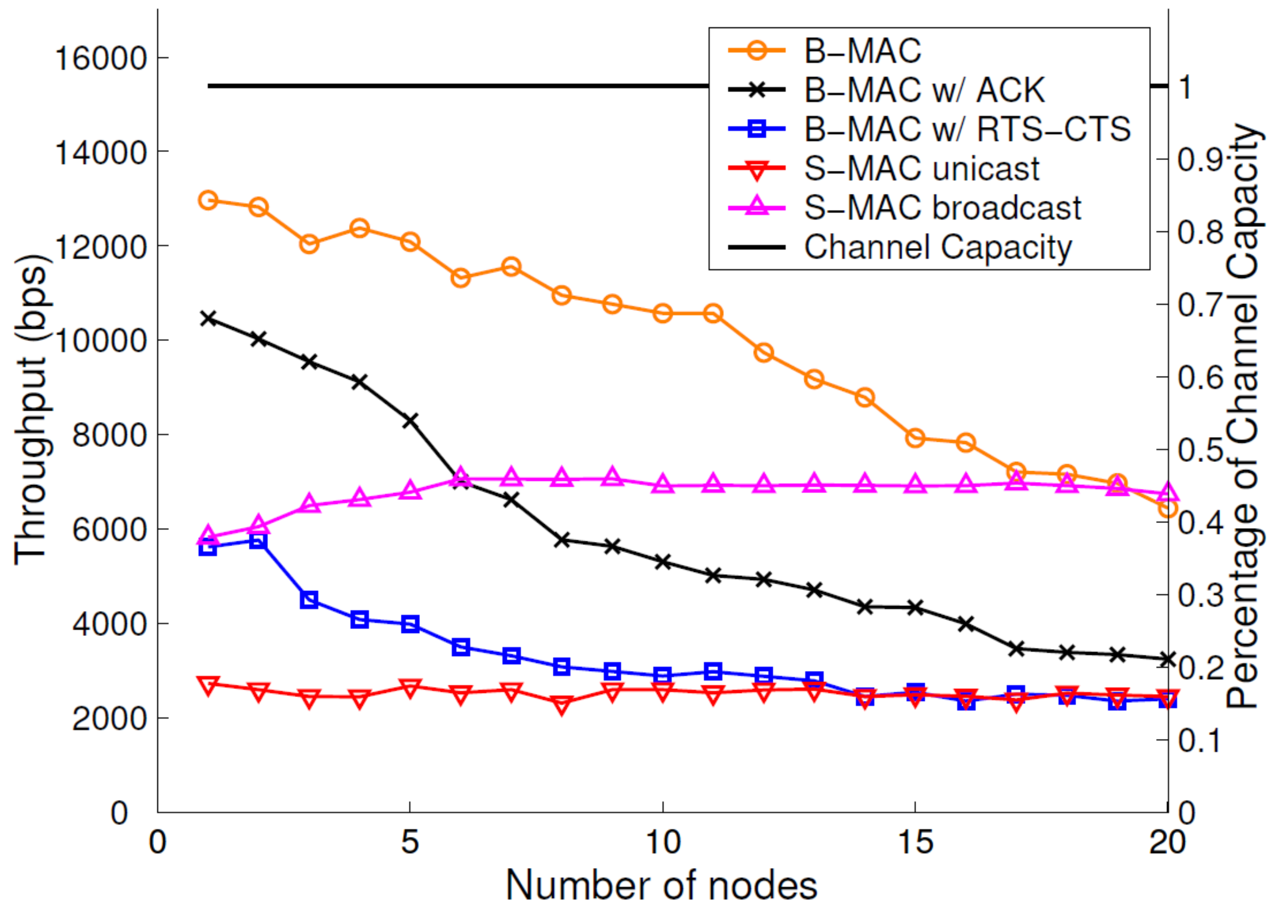
- How long the radio is on after it has been started
- how many times the radio is started.

Polastre, Joseph, Jason Hill, and David Culler. "Versatile low power media access for wireless sensor networks." SENSYS, ACM, 2004.

Polastre, Joseph, Jason Hill, and David Culler. "Versatile low power media access for wireless sensor networks." SENSYS, ACM, 2004.

BMAC is about 4.5 faster than SMAC-unicast

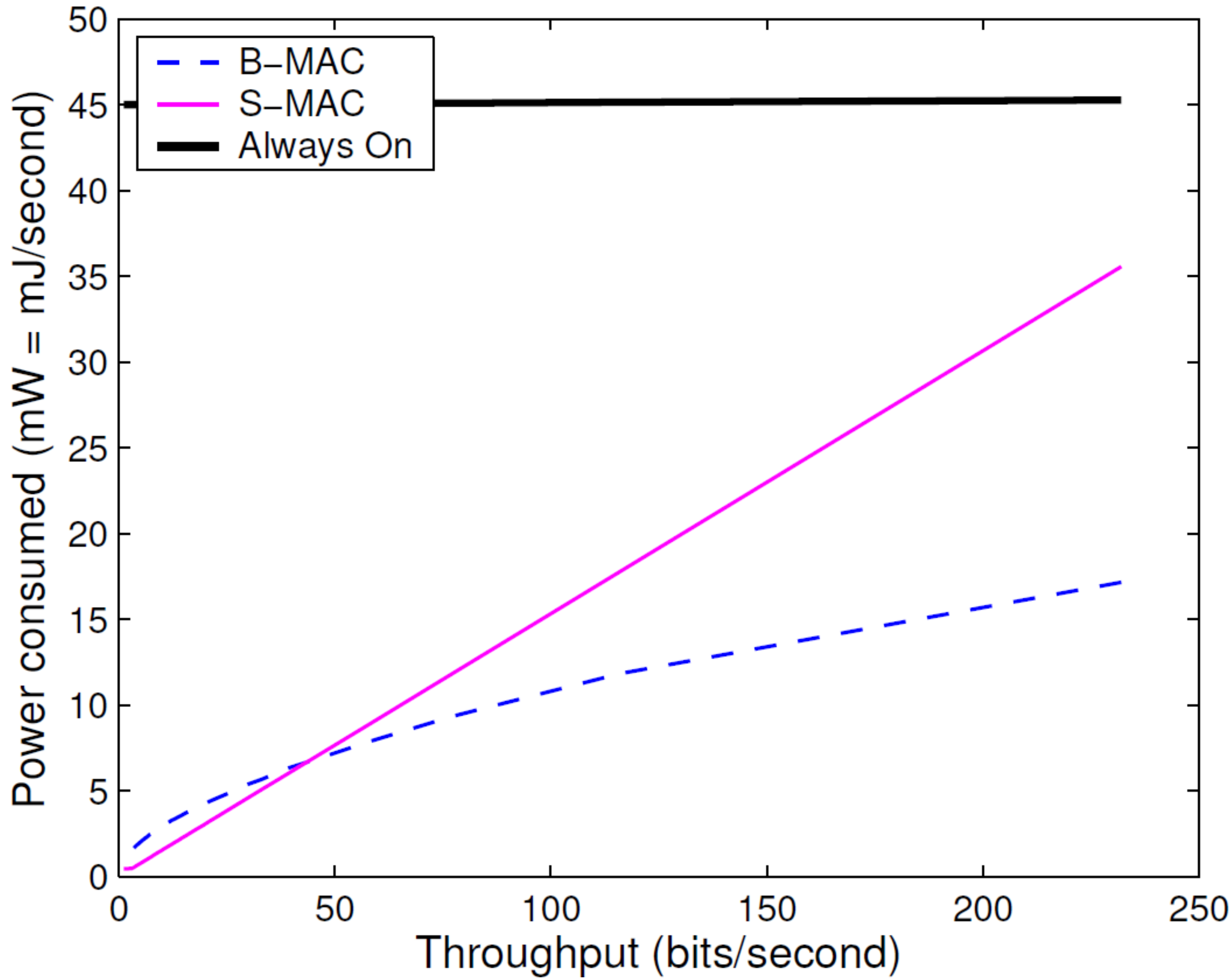
- Not as fast when ACK or RTS/CTS is used
- Differences less pronounced as # of nodes increases
- What about hidden terminal without RTS/CTS?



Polastre, Joseph, Jason Hill, and David Culler. "Versatile low power media access for wireless sensor networks." SENSYS, ACM, 2004.

Low data rates: SMAC is better

BMAC: larger preambles at low throughput, progressively becoming smaller



Asynchronous Duty Cycling

- ContikiMAC

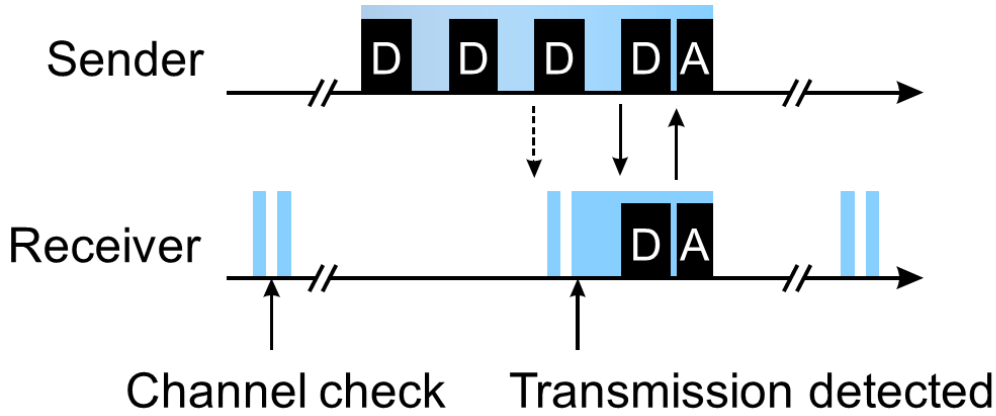
Best effort RDC….

Periodical wake-ups to listen for packet transmissions from neighbors.
Using CCA for checking the channel

If a packet transmission is detected during a wake-up, the receiver is kept on

Link layer acknowledgments from the receiver

Repeated transmissions until a link layer acknowledgment from the receiver.
Or within the full wake-up ensure (broadcast)

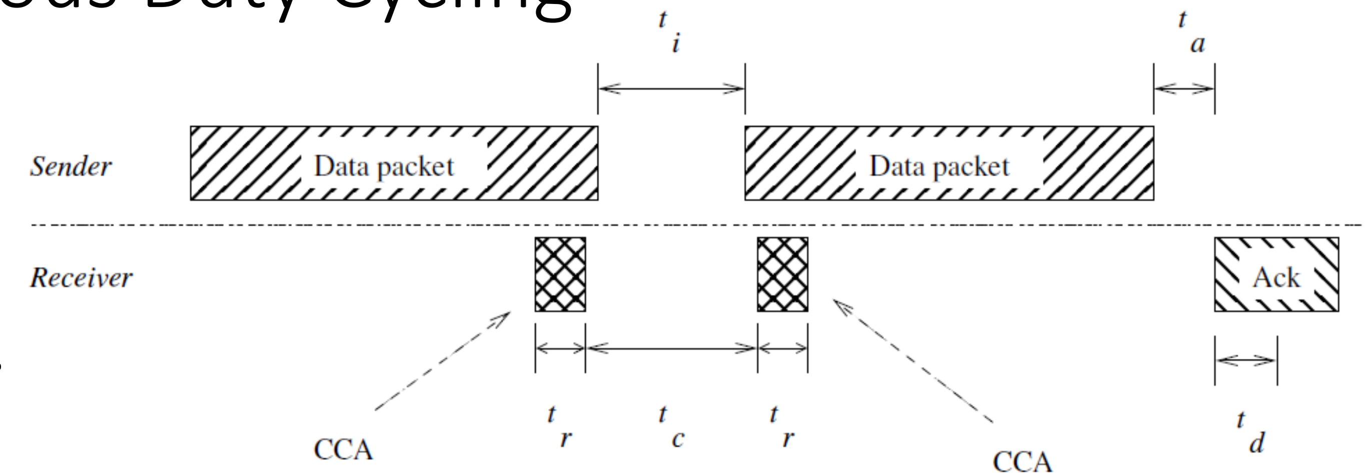


Dunkels, Adam. "The contikimac radio duty cycling protocol." (2011).

Asynchronous Duty Cycling

- ContikiMAC

Timing Aspects…



Dunkels, Adam. "The contikimac radio duty cycling protocol." (2011).

t_i: the interval between each packet transmission.

t_r : the time required for a stable RSSI, needed for a stable CCA indication.

t_c: the interval between each CCA.

t_a: the time between receiving a packet and sending the acknowledgment packet.

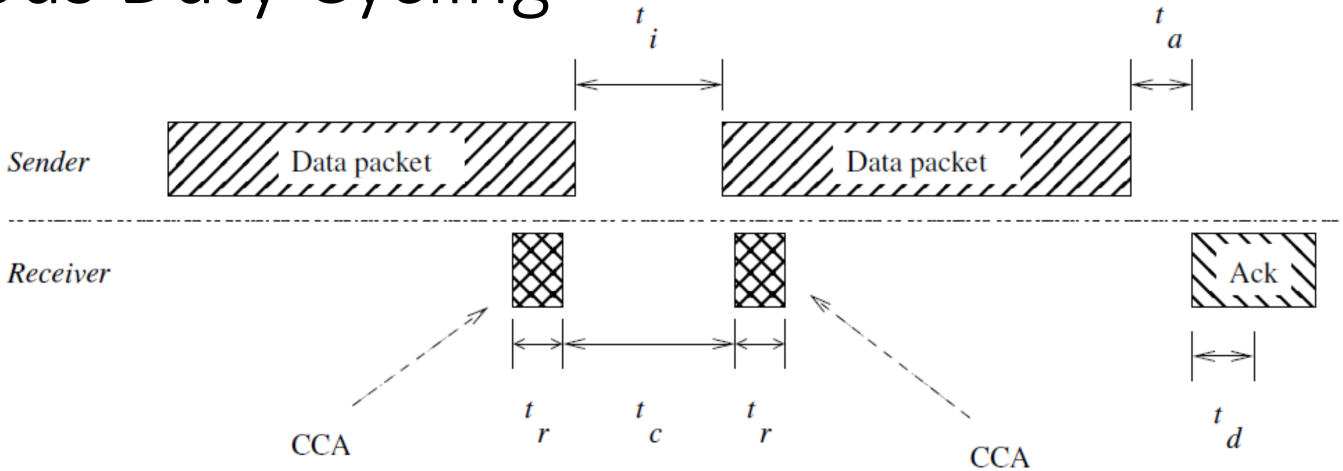
t_d: the time required for successfully detecting an ACK from the receiver.

t_s: transmission time of the shortest packet (depending also on the under. PHY characteristics)

Asynchronous Duty Cycling

- ContikiMAC

Timing Aspects…



Dunkels, Adam. "The contikimac radio duty cycling protocol." (2011).

$t_i < t_c$ To ensure that at least one of the two CCA will catch the transmission attempt

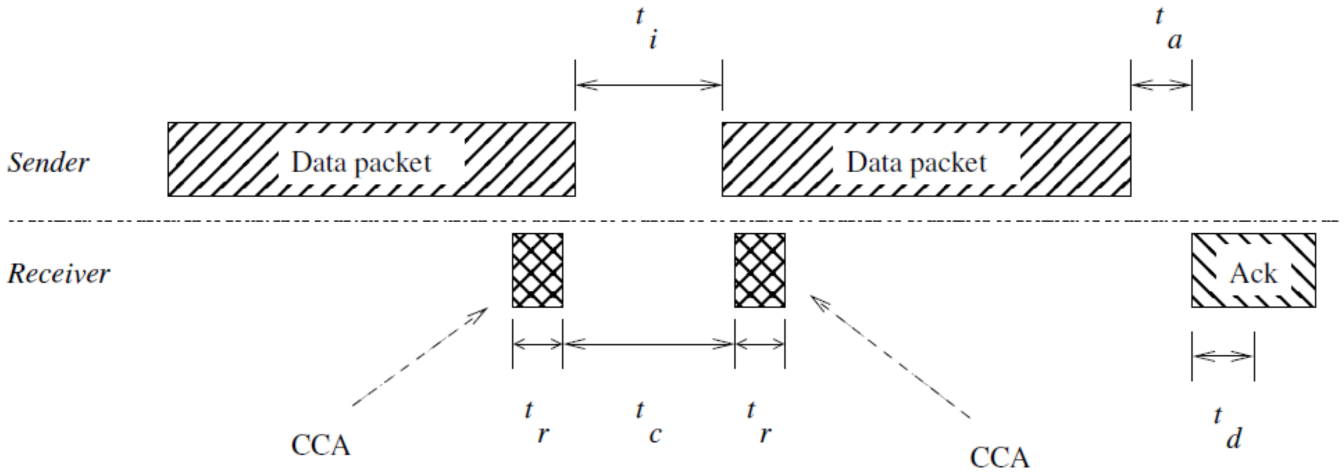
$t_s > 2t_r + t_c$ Data packet should be longer than 2 successive CCA detection periods

$t_a + t_d < t_i$ So that we will avoid a third TX attempt and will not create intra-network interference

What would their combination give????

- ContikiMAC

Timing Aspects…



Dunkels, Adam. "The contikimac radio duty cycling protocol." (2011).

$t_i < t_c$ To ensure that at least one of the two CCA will catch the transmission attempt

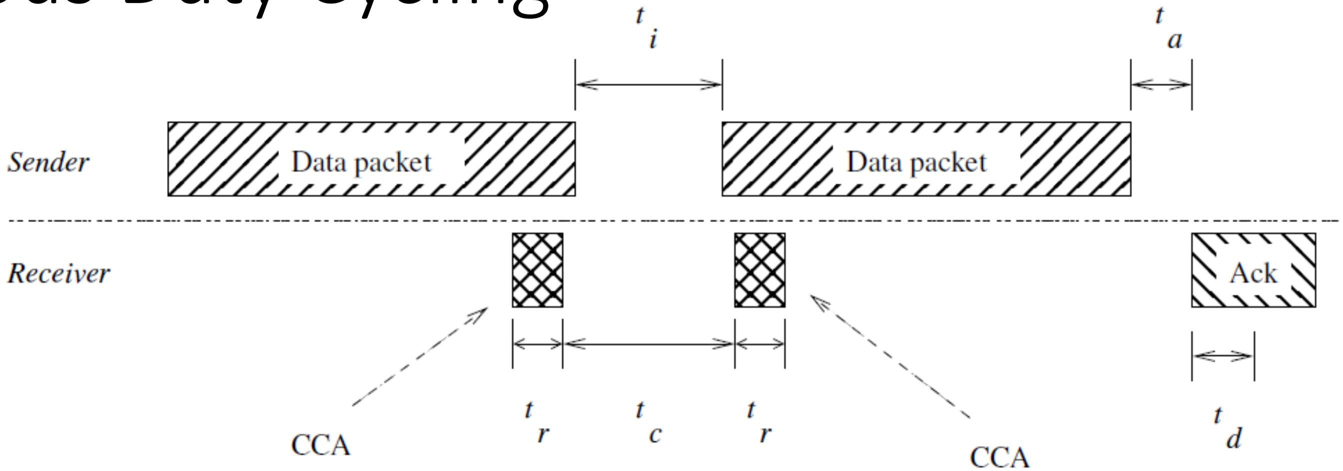
$t_s > 2t_r + t_c$ Data packet should be longer than 2 successive CCA detection periods

$t_a + t_d < t_i$ So that we will avoid a third TX attempt and will not create intra-network interference

Asynchronous Duty Cycling

- ContikiMAC

Timing Aspects…



Dunkels, Adam. "The contikimac radio duty cycling protocol." (2011).

$$t_a + t_d < t_i < t_c < t_c + 2t_r < t_s.$$

Based on symbol duration & channel bit rate you can calculate the min PHY payload size

E.g., IEEE 802.15.4 link layer:

t_a = 12 symbols (1 symbol = 4/250 ms)

An IEEE 802.15.4 receiver can reliably detect the reception of an ACK after the 4-byte long preamble and the 1-byte start of frame delimiter, which totals to 10 symbols

t_r = 0.192 ms (HW dependent)

Asynchronous Duty Cycling

$$t_a + t_d < t_i < t_c < t_c + 2t_r < t_s.$$

E.g., IEEE 802.15.4 2.4GHz / DSSS link layer (DATA RATE 250kbps) :

t_a = 12 symbols (1 symbol = 4/250 ms)

An IEEE 802.15.4 receiver can reliably detect the reception of an ACK after the 4-byte long preamble and the 1-byte start of frame delimiter, which totals to 10 symbols

t_r = 0.192 ms (HW dependent)

Based on symbol duration & channel bit rate you can calculate the min PHY payload size

ta = (12 * 4 /250) ms
tr = 0.192 ms
td = (10 * 4/250) ms

Asynchronous Duty Cycling

$t_a = (12 * 4 / 250)\ ms = 0.192ms$

$t_r = 0.192\ ms$

$t_d = (10 * 4/250) = 0.16ms$

$t_a + t_d < t_i < t_c < 2t_r + t_c < t_s$

$0.192 + 0.16 < t_i < t_c < 0.384 + t_c < t_s$

$0.352 < t_i < t_c \Rightarrow 0.352 < t_c$

$t_c < 0.384 + t_c < t_s \Rightarrow 0.384 + t_c < t_s$

$0.352 + 0.384 + t_c < t_c + t_s$

$0.736 < t_s\ (ms)$

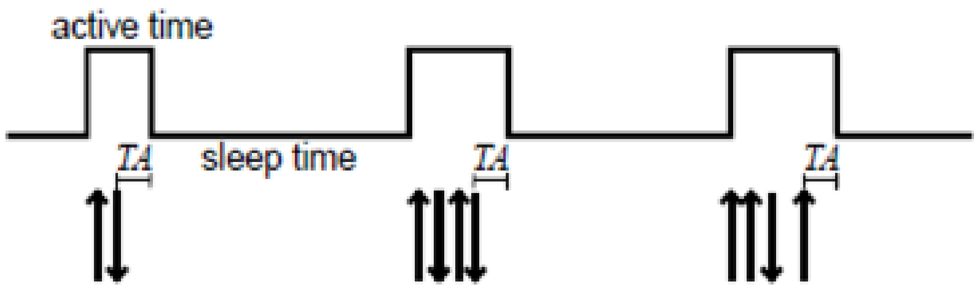
250 kbps (PHY Header 6 bytes)

250kbps => 1 sec -> 250kb

0.736ms -> (0.736 / 1000)(s)*(250 *1000) b/s = 0.736 * 250b = (184 / 8)B = 23 Bytes

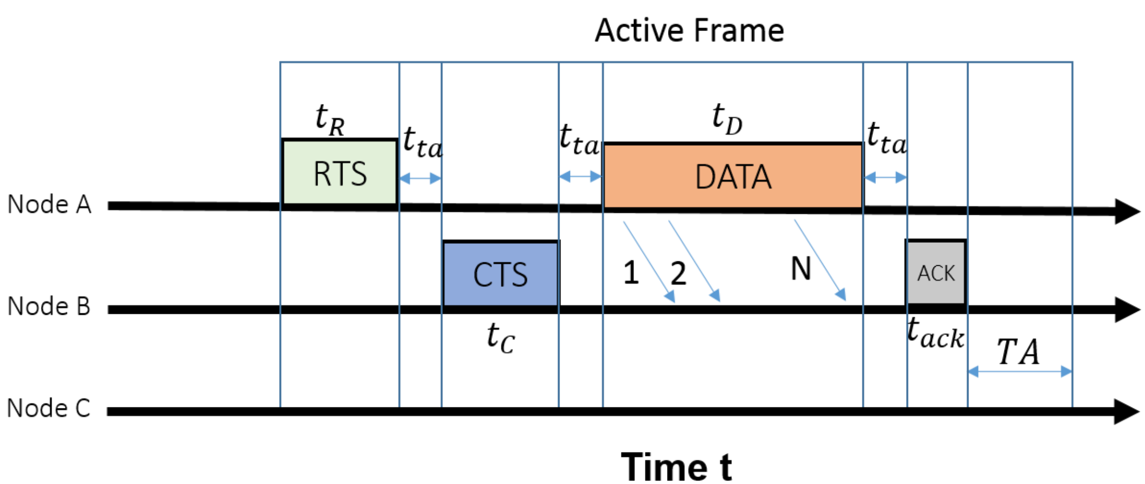
17 Bytes

Exercise on TMAC…



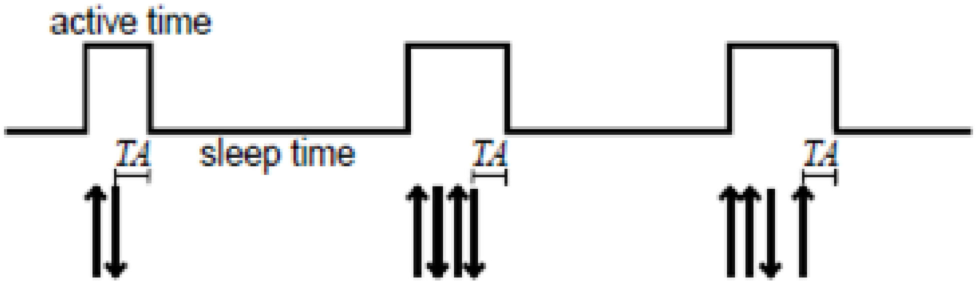
$$TA > t_{con} + t_R + t_{ta}$$

In practical problems, we use $TA = 1.5\,(t_{con} + t_R + t_{ta})$, while the data exchange over time within an active frame is as follows:



(a) t_R is the duration of the RTS packet, (b) t_D is duration of the data transmission, (c) and t_{ack} is the ACK packet duration.

Exercise on TMAC…



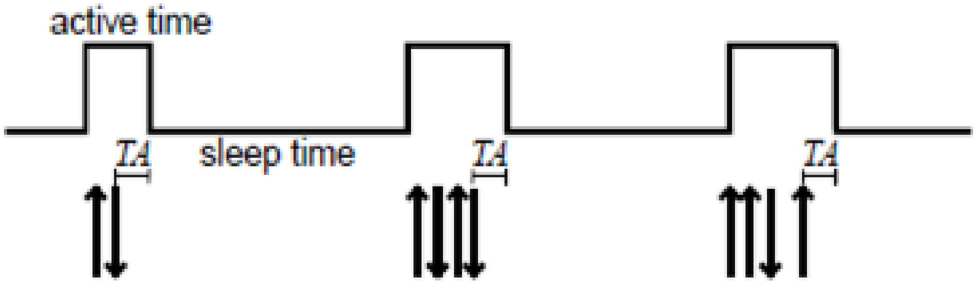
$$TA > t_{con} + t_R + t_{ta}$$

$$TA = 1.5\,(t_{con} + t_R + t_{ta})$$

A. Considering that a RTS packet has 5 bytes PHY payload, and $t_{con} = 20ms$ calculate the value of TA for an IEEE 802.15.4-compliant network that operates at 2.4GHz using DSSS (t_{ta}= 0.192ms)

HINT: The PHY header for IEEE 802.15.4 is 6 Bytes long

Exercise on TMAC…



$$TA > t_{con} + t_R + t_{ta}$$

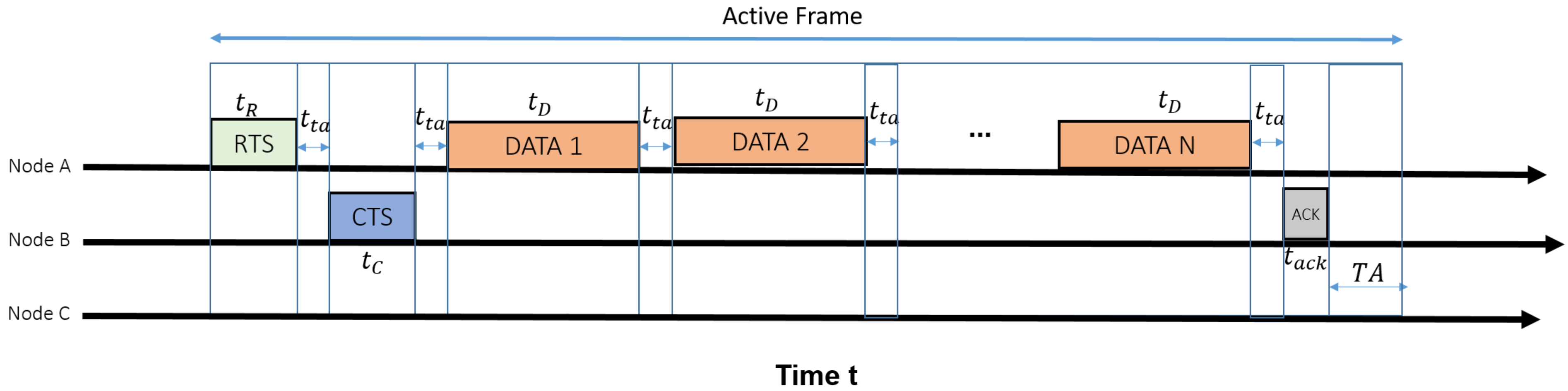
$$TA = 1.5\,(t_{con} + t_R + t_{ta})$$

A. Considering that a RTS packet has 5 bytes PHY payload, and $t_{con} = 20ms$ calculate the value of TA for an IEEE 802.15.4-compliant network that operates at 2.4GHz using DSSS (t_{ta}= 0.192ms)

HINT: The PHY header for IEEE 802.15.4 is 6 Bytes long

Case 1 time needed for RTS: 11*8/250kpbs = 0.352ms – 1.5*(20+0.352+0.192)

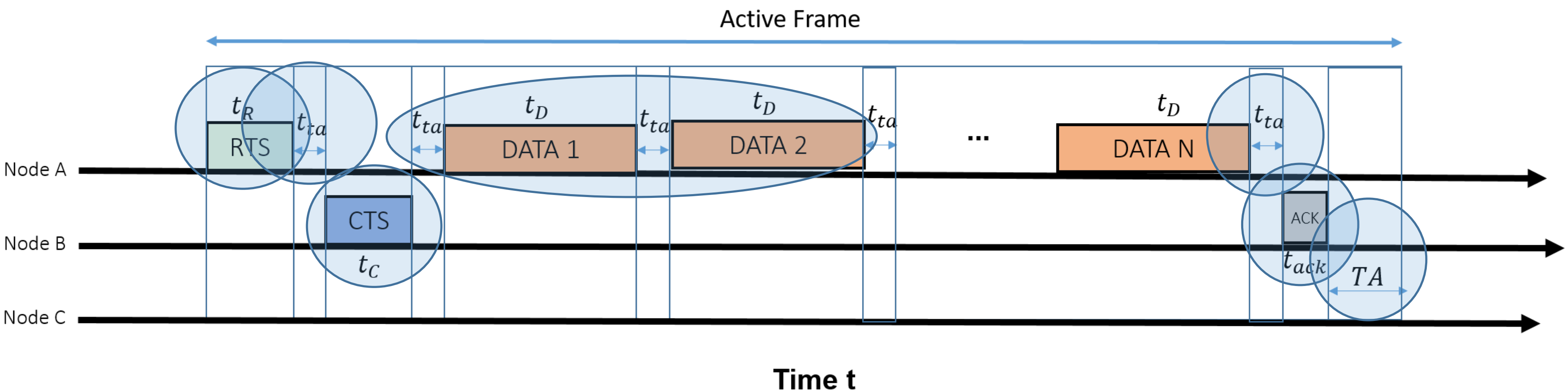
Exercise on TMAC…



B. calculate the total duration of the active frame, considering that both the PHY payload of the CTS and the ACK packet equals to 5 Bytes, and for total data size 160 Bytes.

HINT: The PHY header for IEEE 802.15.4 is 6 Bytes long, and the maximum value of the PHY frame is 128Bytes.

Exercise on TMAC…



B. calculate the total duration of the active frame, considering that both the PHY payload of the CTS and the ACK packet equals to 5 Bytes, and for 160 bytes per second.

HINT: The PHY header for IEEE 802.15.4 is 6 Bytes long, and the maximum value of the PHY frame is 128Bytes.

160bytes – 2 packets: 86 bytes each (incl. PHY header) $t_{D=}\left(86 * \frac{8}{250}\right) ms$ = 2.752ms

$$t_R + t_{ta} + t_C + t_{ta} + 2 * (t_D + t_{ta}) + t_{ack} + TA$$

Practical Aspects

- Side effects of bad RDC

A single TX-RX pair, scanning different channels, and different tx power

On weak links & low data rate…:

Repeatedly retransmitted packets due to asynchronous policy can be perceived as noise (multi-path effect…)

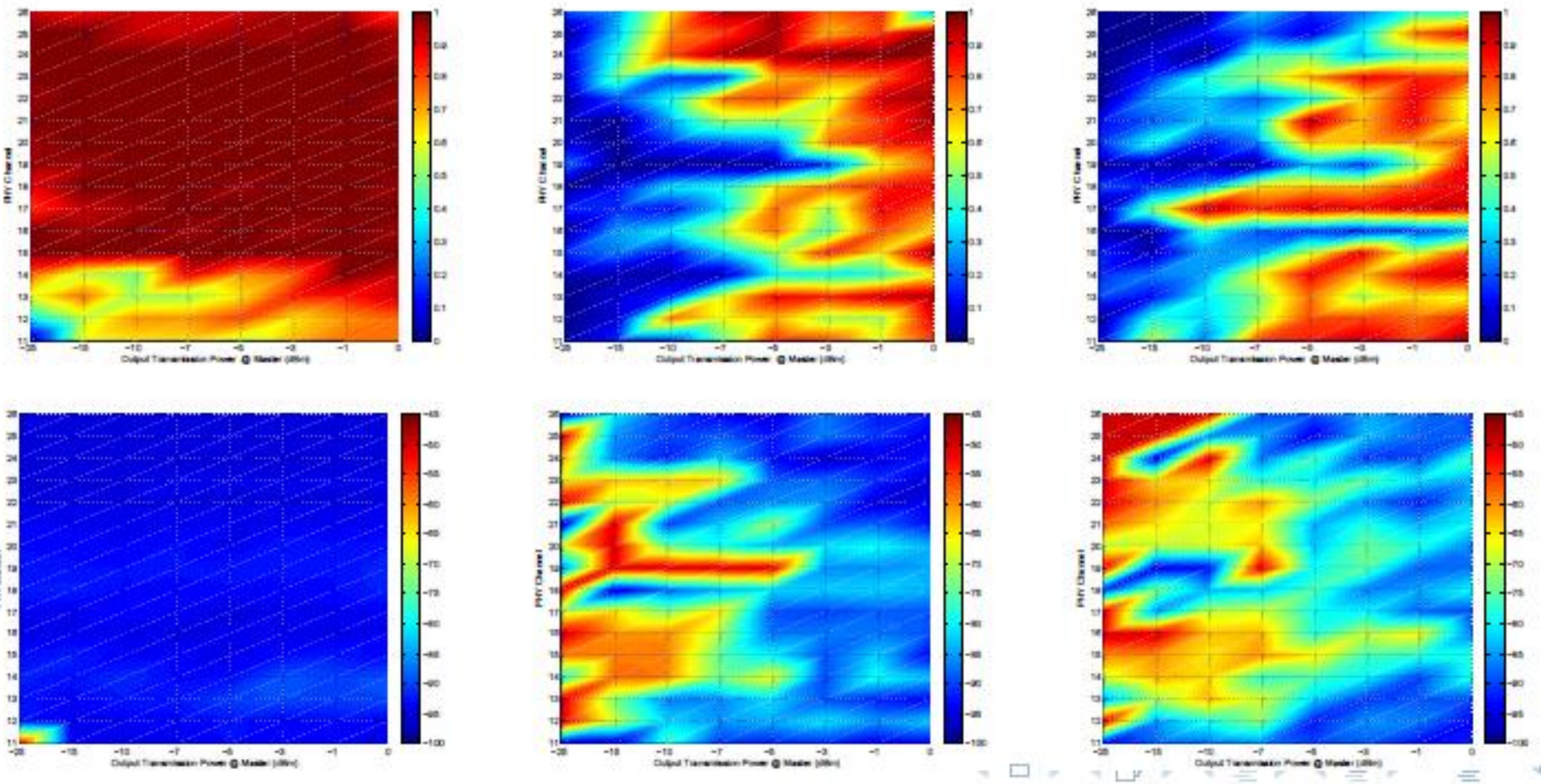
Intra-network interference

Practical Aspects

No RDC | ACK-based RDC | ACK-based RDC – poor synch

PRR

NF



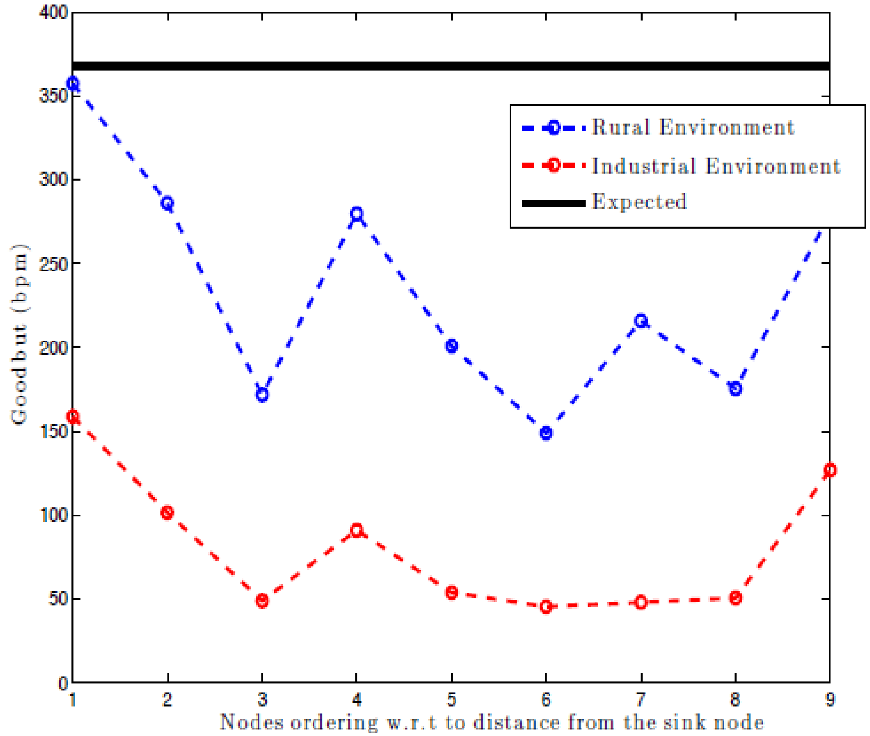
Practical Aspects

- Side effects of bad RDC

Multi-hop networks with low data rate

Destabilizing network behavior, due to low data rate (nodes that are not relaying packets)

Keeping the control / sync messages could alleviate this issue



References and Material for Reading

1. Casilari, Eduardo; Cano-García, Jose M.; Campos-Garrido, Gonzalo. 2010. "Modeling of Current Consumption in 802.15.4/ZigBee Sensor Motes." Sensors 10, no. 6: 5443-5468.
2. "Guide to Wireless Sensor Networks", S. Misra, I. Woungang, S. C. Misra, 2009, Springer, Ch 15
3. W. Ye, J. Heidemann, and D. Estrin. Medium access control with coordinated adaptive sleeping for wireless sensor networks. IEEE/ACM Transactions on Networks, 12(3):493–506, 2004.
4. Dunkels, Adam. "The contikimac radio duty cycling protocol." (2011).
5. Raghunathan, V.; Ganeriwal, S.; Srivastava, M., "Emerging techniques for long lived wireless sensor networks," Communications Magazine, IEEE , vol.44, no.4, pp.108,114, April 2006
6. T. van Dam and K. Langendoen. An adaptive energy-efficient MAC protocol for wireless sensor networks. In ACM Sensys, 2003.
7. Holger Karl, Andreas Willig, Protocols and Architectures for Wireless Sensor Systems, 2005, Willey, Ch. 5
8. Wei Ye; Heidemann, J.; Estrin, D., "An energy-efficient MAC protocol for wireless sensor networks," INFOCOM 2002. Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE , vol.3, no., pp.1567,1576 vol.3, 2002
9. Polastre, Joseph, Jason Hill, and David Culler. "Versatile low power media access for wireless sensor networks." SENSYS, ACM, 2004.