# Evaluation of the 6TiSCH Network Formation

**Dario Fanucchi**[1] Barbara Staehle[2] Rudi Knorr [1,3]

[1]Department of Computer Science
University of Augsburg, Germany

[2]Department of Computer Science
University of Applied Sciences Konstanz, Germany

[3]Fraunhofer Institute for Embedded Systems and
Communication Technologies, Munich, Germany

Universität
Augsburg
University

Hochschule Konstanz
Department of Computer Science

Fraunhofer
ESK

# Industrial Wireless Sensor Networks (IWSNs)

Targeted applications:
Process monitoring and control

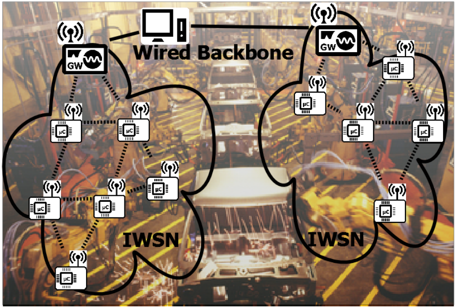
Strict requirements:

- Reliability up to 99,999%
- Lifetime > 5 years
- Latency: tens of milliseconds

Main characteristics of a IWSN:

- mesh, multi-hop, lossy network
- harsh environment
- a gateway and up to 100 resource constrained nodes
- specific designed communication protocols

# Industrial Wireless Sensor Networks (IWSNs)




Targeted applications:
Process monitoring and control

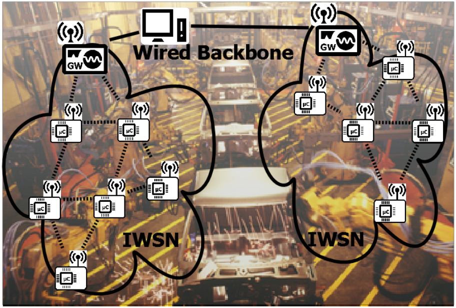
Strict requirements:

- Reliability up to 99,999%
- Lifetime $>$ 5 years
- Latency: tens of milliseconds

Main characteristics of a IWSN:

- mesh, multi-hop, lossy network
- harsh environment
- a gateway and up to 100 resource constrained nodes
- specific designed communication protocols

# Industrial Wireless Sensor Networks (IWSNs)




Targeted applications:
Process monitoring and control

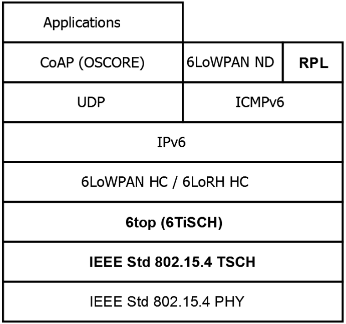
Strict requirements:

- Reliability up to 99,999%
- Lifetime $>$ 5 years
- Latency: tens of milliseconds

Main characteristics of a IWSN:

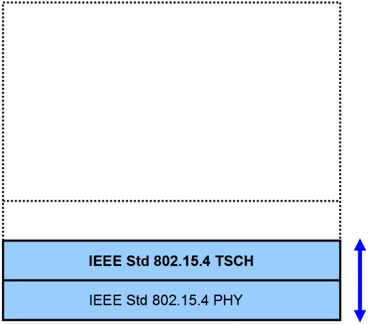
- mesh, multi-hop, lossy network
- harsh environment
- a gateway and up to 100 resource constrained nodes
- specific designed communication protocols

# 6TiSCH-Stack: IETF Suite of Protocols for Industrial IoTs



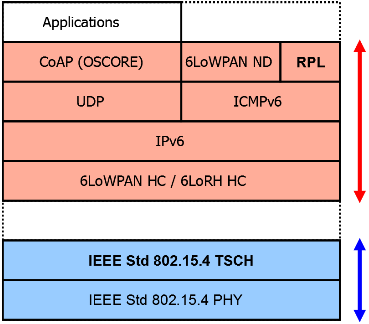

- Upper layers: IPv6-connectivity
  - 6LoWPAN, IPv6, CoAP etc.
  - *RPL as distributed routing protocol*
- Glueing together: 6top protocol proposed by *IETF 6TiSCH*
  - assignment of communication links
  - *definition of bootstrapping procedures*
- At the bottom: IEEE 802.15.4-2015
  - 2.4 GHz low-power radio
  - *MAC: Time Slotted Channel Hopping* for industrial performance

# 6TiSCH-Stack: IETF Suite of Protocols for Industrial IoTs



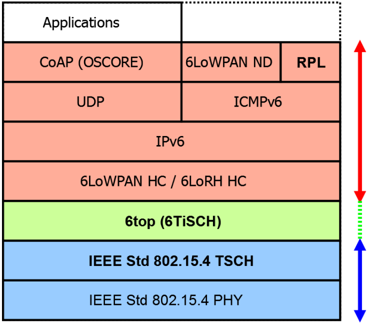

- Upper layers: IPv6-connectivity
  - 6LoWPAN, IPv6, CoAP etc.
  - *RPL as distributed routing protocol*
- Glueing together: 6top protocol proposed by *IETF 6TiSCH*
  - assignment of communication links
  - *definition of bootstrapping procedures*
- At the bottom: IEEE 802.15.4-2015
  - 2.4 GHz low-power radio
  - *MAC: Time Slotted Channel Hopping* for industrial performance

# 6TiSCH-Stack: IETF Suite of Protocols for Industrial IoTs




- Upper layers: IPv6-connectivity
  - 6LoWPAN, IPv6, CoAP etc.
  - *RPL as distributed routing protocol*
- Glueing together: 6top protocol proposed by *IETF 6TiSCH*
  - assignment of communication links
  - *definition of bootstrapping procedures*
- At the bottom: IEEE 802.15.4-2015
  - 2.4 GHz low-power radio
  - *MAC: Time Slotted Channel Hopping* for industrial performance

# 6TiSCH-Stack: IETF Suite of Protocols for Industrial IoTs




- Upper layers: IPv6-connectivity
  - 6LoWPAN, IPv6, CoAP etc.
  - *RPL as distributed routing protocol*
- Glueing together: 6top protocol proposed by *IETF 6TiSCH*
  - assignment of communication links
  - *definition of bootstrapping procedures*
- At the bottom: IEEE 802.15.4-2015
  - 2.4 GHz low-power radio
  - *MAC: Time Slotted Channel Hopping* for industrial performance

Scenario:

- Mesh, multi-hop network for industrial wireless
- 6TiSCH-Stack as IETF proposal for industrial IoT

Problem statement:

- Interplay of MAC and Routing protocols affects network performance
- Initial network formation is challenging

Our simulative study hightlights...

- why a blind adoption of IETF 6TiSCH proposal is risky
- how to tune the MAC and Routing protocol for a successful network formation (situation dependent)

Scenario:

- Mesh, multi-hop network for industrial wireless
- 6TiSCH-Stack as IETF proposal for industrial IoT

Problem statement:

- Interplay of MAC and Routing protocols affects network performance
- Initial network formation is challenging

Our simulative study hightlights...

- why a blind adoption of IETF 6TiSCH proposal is risky
- how to tune the MAC and Routing protocol for a successful network formation (situation dependent)

Scenario:

- Mesh, multi-hop network for industrial wireless
- 6TiSCH-Stack as IETF proposal for industrial IoT

Problem statement:

- Interplay of MAC and Routing protocols affects network performance
- Initial network formation is challenging

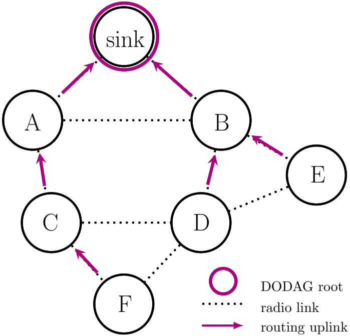
Our simulative study hightlights...

- why a blind adoption of IETF 6TiSCH proposal is risky
- how to tune the MAC and Routing protocol for a successful network formation (situation dependent)

# How a 6TiSCH network is formed (1)

At least two processes, before network is operational:

1. TSCH synchronisation
   - Goal: build a globally synchronized mesh network
   - Exchanging Enhanced Beacon (EB) frames with time information
2. RPL DODAG construction
   - Goal: organize nodes as a directed tree rooted at the sink
   - Exchanging DODAG Information Object (DIO) packets



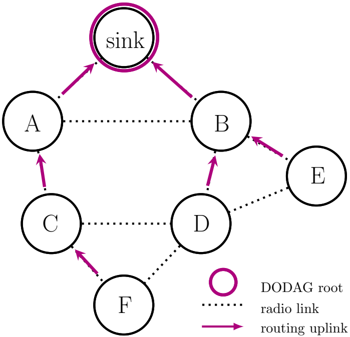

How can we coordinate these two interplaying processes?

# How a 6TiSCH network is formed (1)

At least two processes, before network is operational:

1. TSCH synchronisation
   - Goal: build a globally synchronized mesh network
   - Exchanging Enhanced Beacon (EB) frames with time information

2. RPL DODAG construction

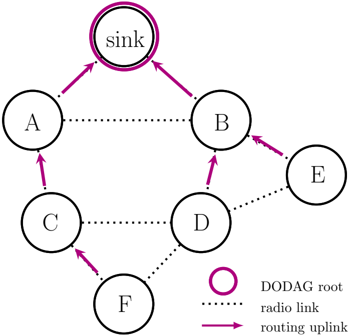



How can we coordinate these two interplaying processes?

# How a 6TiSCH network is formed (1)

At least two processes, before network is operational:

1. TSCH synchronisation
   - Goal: build a globally synchronized mesh network
   - Exchanging Enhanced Beacon (EB) frames with time information
     - emitted every $t_{eb}$
2. RPL DODAG construction
   - Goal: organize nodes as a directed tree rooted at the sink
   - Exchanging DODAG Information Object (DIO) packets

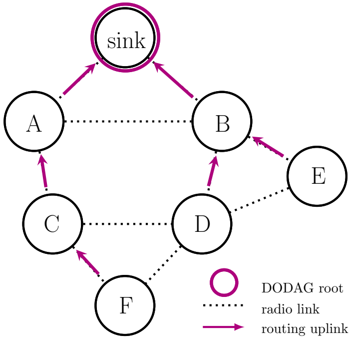



How can we coordinate these two interplaying processes?

# How a 6TiSCH network is formed (1)

At least two processes, before network is operational:

1. TSCH synchronisation
   - Goal: build a globally synchronized mesh network
   - Exchanging Enhanced Beacon (EB) frames with time information
     - emitted every $t_{eb}$
2. RPL DODAG construction
   - Goal: organize nodes as a directed tree rooted at the sink
   - Exchanging DODAG Information Object (DIO) packets

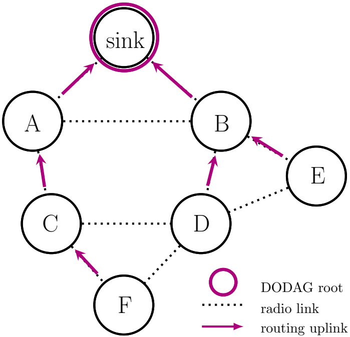



How can we coordinate these two interplaying processes?

# How a 6TiSCH network is formed (1)

At least two processes, before network is operational:

1. TSCH synchronisation
   - Goal: build a globally synchronized mesh network
   - Exchanging Enhanced Beacon (EB) frames with time information
     - emitted every $t_{eb}$
2. RPL DODAG construction
   - Goal: organize nodes as a directed tree rooted at the sink
   - Exchanging DODAG Information Object (DIO) packets
     - Trickle algorithm for adaptive generation

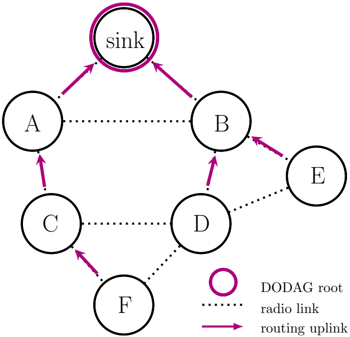



How can we coordinate these two interplaying processes?

# How a 6TiSCH network is formed (1)

At least two processes, before network is operational:

1. TSCH synchronisation
   - Goal: build a globally synchronized mesh network
   - Exchanging Enhanced Beacon (EB) frames with time information
     - emitted every $t_{eb}$
2. RPL DODAG construction
   - Goal: organize nodes as a directed tree rooted at the sink
   - Exchanging DODAG Information Object (DIO) packets
     - Trickle algorithm for adaptive generation

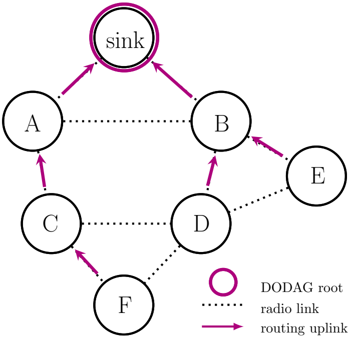



How can we coordinate these two interplaying processes?

# How a 6TiSCH network is formed (1)

At least two processes, before network is operational:

1. TSCH synchronisation
   - Goal: build a globally synchronized mesh network
   - Exchanging Enhanced Beacon (EB) frames with time information
     - emitted every $t_{eb}$
2. RPL DODAG construction
   - Goal: organize nodes as a directed tree rooted at the sink
   - Exchanging DODAG Information Object (DIO) packets
     - Trickle algorithm for adaptive generation

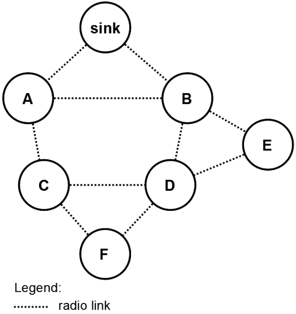



## How can we coordinate these two interplaying processes?

# How a 6TiSCH network is formed (2)

IETF 6TiSCH *minimal configuration* (6TiSCH-MC, RFC 8180):

1. Sink sets TSCH-schedule with one shared slot and sends EBs and DIOs
2. Joining Nodes keep their radio on and listen for EB
3. After hearing an EB: Node learns the *minimal schedule* and is synchronised
4. After hearing a DIO: Node selects a preferred parent and broadcasts EBs and DIOs messages on its turn.




At the end: every node knowns the *minimal schedule* and is in the DODAG

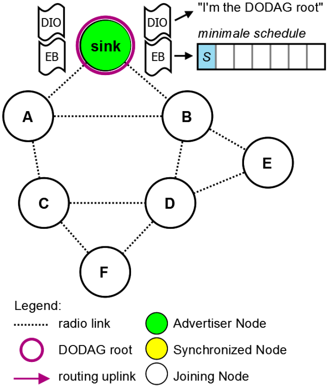
How good is 6TiSCH-MC?

# How a 6TiSCH network is formed (2)

IETF 6TiSCH *minimal configuration* (6TiSCH-MC, RFC 8180):

1. Sink sets TSCH-schedule with one shared slot and sends EBs and DIOs
2. Joining Nodes keep their radio on and listen for EB
3. After hearing an EB: Node learns the *minimal schedule* and is synchronised
4. After hearing a DIO: Node selects a preferred parent and broadcasts EBs and DIOs messages on its turn.

At the end: every node knowns the *minimal schedule* and is in the DODAG



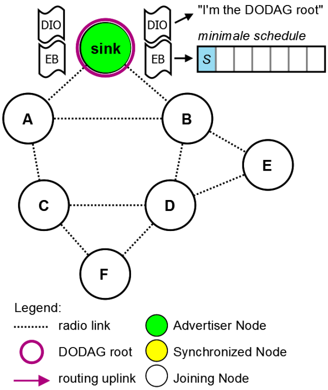

How good is 6TiSCH-MC?

# How a 6TiSCH network is formed (2)

IETF 6TiSCH *minimal configuration* (6TiSCH-MC, RFC 8180):

1. Sink sets TSCH-schedule with one shared slot and sends EBs and DIOs
2. Joining Nodes keep their radio on and listen for EB
3. After hearing an EB: Node learns the *minimal schedule* and is synchronised
4. After hearing a DIO: Node selects a preferred parent and broadcasts EBs and DIOs messages on its turn.

At the end: every node knowns the *minimal schedule* and is in the DODAG



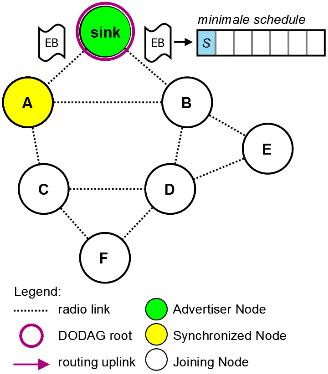

How good is 6TiSCH-MC?

# How a 6TiSCH network is formed (2)

IETF 6TiSCH *minimal configuration* (6TiSCH-MC, RFC 8180):

1. Sink sets TSCH-schedule with one shared slot and sends EBs and DIOs
2. Joining Nodes keep their radio on and listen for EB
3. After hearing an EB: Node learns the *minimal schedule* and is synchronised
4. After hearing a DIO: Node selects a preferred parent and broadcasts EBs and DIOs messages on its turn.

At the end: every node knowns the *minimal schedule* and is in the DODAG

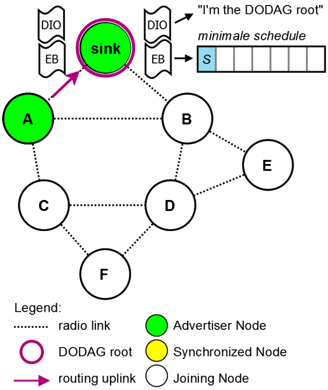



How good is 6TiSCH-MC?

# How a 6TiSCH network is formed (2)

IETF 6TiSCH *minimal configuration* (6TiSCH-MC, RFC 8180):

1. Sink sets TSCH-schedule with one shared slot and sends EBs and DIOs
2. Joining Nodes keep their radio on and listen for EB
3. After hearing an EB: Node learns the *minimal schedule* and is synchronised
4. After hearing a DIO: Node selects a preferred parent and broadcasts EBs and DIOs messages on its turn.



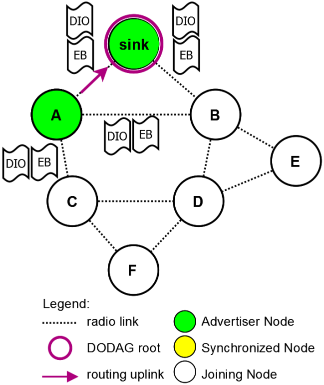

At the end: every node knowns the *minimal schedule* and is in the DODAG

How good is 6TiSCH-MC?

# How a 6TiSCH network is formed (2)

IETF 6TiSCH *minimal configuration* (6TiSCH-MC, RFC 8180):

1. Sink sets TSCH-schedule with one shared slot and sends EBs and DIOs
2. Joining Nodes keep their radio on and listen for EB
3. After hearing an EB: Node learns the *minimal schedule* and is synchronised
4. After hearing a DIO: Node selects a preferred parent and broadcasts EBs and DIOs messages on its turn.




At the end: every node knowns the *minimal schedule* and is in the DODAG

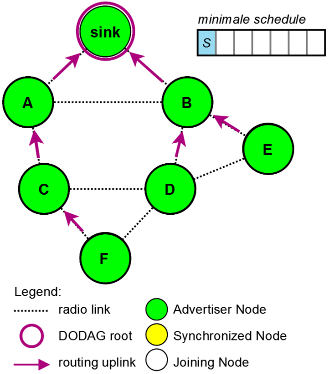
How good is 6TiSCH-MC?

# How a 6TiSCH network is formed (2)

IETF 6TiSCH *minimal configuration* (6TiSCH-MC, RFC 8180):

1. Sink sets TSCH-schedule with one shared slot and sends EBs and DIOs
2. Joining Nodes keep their radio on and listen for EB
3. After hearing an EB: Node learns the *minimal schedule* and is synchronised
4. After hearing a DIO: Node selects a preferred parent and broadcasts EBs and DIOs messages on its turn.

At the end: every node knowns the *minimal schedule* and is in the DODAG

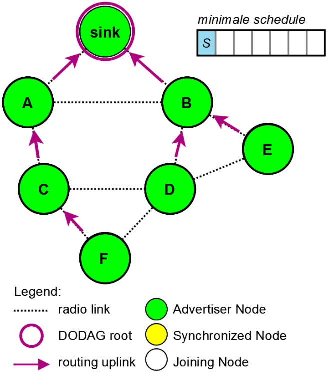



How good is 6TiSCH-MC?

# How a 6TiSCH network is formed (2)

IETF 6TiSCH *minimal configuration* (6TiSCH-MC, RFC 8180):

1. Sink sets TSCH-schedule with one shared slot and sends EBs and DIOs
2. Joining Nodes keep their radio on and listen for EB
3. After hearing an EB: Node learns the *minimal schedule* and is synchronised
4. After hearing a DIO: Node selects a preferred parent and broadcasts EBs and DIOs messages on its turn.




At the end: every node knowns the *minimal schedule* and is in the DODAG

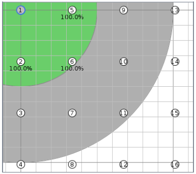
## How good is 6TiSCH-MC?

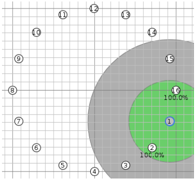
# Simulation Setup

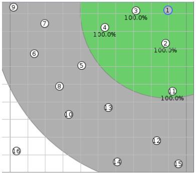
- Contiki OS and Cooja simulator
  (1) open-source, (2) popular and (3) compliance with 6TiSCH-stack
- Different topologies and three network sizes $N_{size} \in \{9, 16, 25\}$

# Simulation Setup

- Contiki OS and Cooja simulator
  (1) open-source, (2) popular and (3) compliance with 6TiSCH-stack
- Different topologies and three network sizes $N_{size} \in \{9, 16, 25\}$

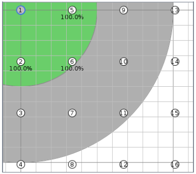


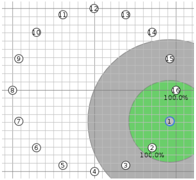


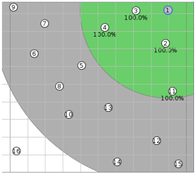


# Simulation Setup

- Contiki OS and Cooja simulator
  (1) open-source, (2) popular and (3) compliance with 6TiSCH-stack
- Different topologies and three network sizes $N_{size} \in \{9, 16, 25\}$





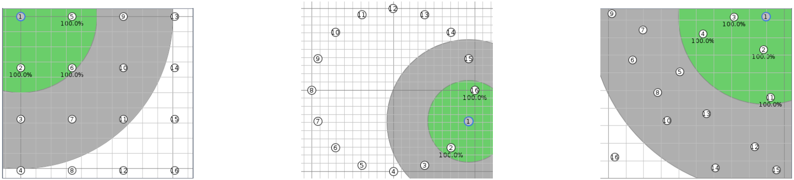


- Varying crucial parameter of TSCH and RPL Trickle:

| Parameter | Symbol | Value |
|---|---|---|
| TSCH number of channels | $N_c$ | $\{4, 16\}$ |
| TSCH EB period | $t_{eb}$ | $\{2048, 4096, 8192, 16384\}$ ms |
| RPL minimal interval | $I_{min}$ | $\{128, 256, ..., 4096\}$ ms |

# Simulation Setup

- Contiki OS and Cooja simulator
  (1) open-source, (2) popular and (3) compliance with 6TiSCH-stack
- Different topologies and three network sizes $N_{size} \in \{9, 16, 25\}$



- Varying crucial parameter of TSCH and RPL Trickle:

| Parameter | Symbol | Value |
|---|---|---|
| TSCH number of channels | $N_c$ | $\{4, 16\}$ |
| TSCH EB period | $t_{eb}$ | $\{2048, 4096, 8192, 16384\}$ ms |
| RPL minimal interval | $I_{min}$ | $\{128, 256, ..., 4096\}$ ms |

- Performance metrics:
  (1) time, (2) charge consumed and (3) number of control frames exchanged until completed network formation

# Results: Limits of 6TiSCH-MC

In dense network or with improper setting of TSCH and RPL parameters:

1. some nodes are not yet operational after 30 minutes
2. high battery consumption in several nodes

Table: Successful DODAG formations within 30 min

| | | Grid | | | Ellipse | | | Random | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | $N_{size}$ | | | $N_{size}$ | | | $N_{size}$ | | |
| $t_{eb}$ | $N_c$ | 9 | 16 | 25 | 9 | 16 | 25 | 9 | 16 | 25 |
| 2048 ms | 4 | 0% | 0% | 0% | 70% | 16% | 0% | 0% | 0% | 0% |
| 4096 ms | 4 | 84% | 0% | 0% | 100% | 100% | 86% | 46% | 0% | 0% |
| 8192 ms | 4 | 100% | 20% | 0% | 100% | 100% | 100% | 100% | 2% | 0% |
| 16384 ms | 4 | 100% | 70% | 0% | 100% | 100% | 100% | 100% | 14% | 0% |

With slotframe duration $T_{sf} = 1.01\,s$ (i.e. $N_s = 101$, $t_s = 10\,ms$) and $I_{min} = 1024\,ms$

Due to...

- collisions of control frame
- queuing delay of DIO packets

# Results: Limits of 6TiSCH-MC

In dense network or with improper setting of TSCH and RPL parameters:

1. some nodes are not yet operational after 30 minutes
2. high battery consumption in several nodes

Table: Successful DODAG formations within 30 min

| | | Grid | | | Ellipse | | | Random | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | $N_{size}$ | | | $N_{size}$ | | | $N_{size}$ | | |
| $t_{eb}$ | $N_c$ | 9 | 16 | 25 | 9 | 16 | 25 | 9 | 16 | 25 |
| 2048 ms | 4 | 0% | 0% | 0% | 70% | 16% | 0% | 0% | 0% | 0% |
| 4096 ms | 4 | 84% | 0% | 0% | 100% | 100% | 86% | 46% | 0% | 0% |
| 8192 ms | 4 | 100% | 20% | 0% | 100% | 100% | 100% | 100% | 2% | 0% |
| 16384 ms | 4 | 100% | 70% | 0% | 100% | 100% | 100% | 100% | 14% | 0% |

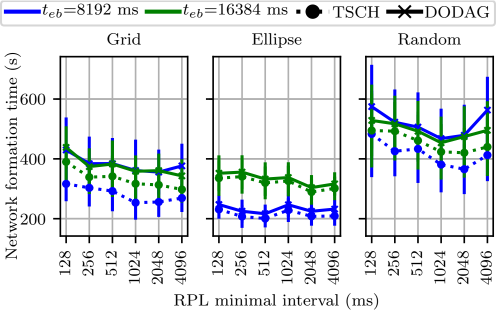
With slotframe duration $T_{sf} = 1.01\,s$ (i.e. $N_s = 101$, $t_s = 10\,ms$) and $I_{min} = 1024\,ms$

Due to...

- collisions of control frame
- queuing delay of DIO packets

# Main Observations and Recommendations (1)

Network formation time:




In dense topologies:

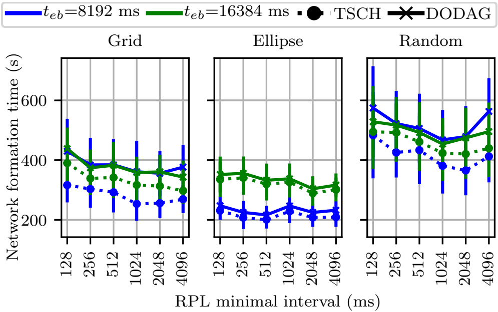
1. Time gap between TSCH-synchronisation and DODAG completion
2. RPL minimal interval $I_{min}$ matters

**Recommendations for implementers**

Set $t_{eb} \geq 4 \cdot (\text{slotframe duration})$ and $I_{min} = (\text{slotframe duration}) + \epsilon$

# Main Observations and Recommendations (1)

Network formation time:




In dense topologies:

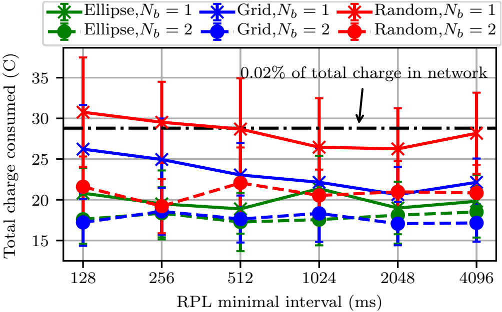
1. Time gap between TSCH-synchronisation and DODAG completion
2. RPL minimal interval $I_{min}$ matters

## Recommendations for implementers

Set $t_{eb} \geq 4 \cdot$ (slotframe duration) and $I_{min} =$ (slotframe duration) $+ \epsilon$

# Main Observations and Recommendations (2)

Extending 6TiSCH-MC with $N_b = 2$ shared slots:



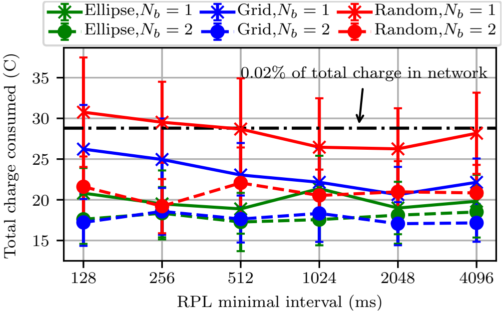

1. theoretical twice duty-cycle, but reduced charge consumed
2. reduction of the time spent for network formation

Recommendation for implementers:

If dense topologies: Add additional shared slots for time and energy savings

# Main Observations and Recommendations (2)

Extending 6TiSCH-MC with $N_b = 2$ shared slots:




1. theoretical twice duty-cycle, but reduced charge consumed
2. reduction of the time spent for network formation

**Recommendation for implementers:**

If dense topologies: Add additional shared slots for time and energy savings

# Summary

Herein:

- Overview of the IETF 6TiSCH *minimal configuration* (6TiSCH-MC)
- Extensive simulations to characterize its behaviour

Conclusions:

- Potential downsides of 6TiSCH-MC with dense topologies
- Recommendations for setting TSCH and RPL parameters

Future work:

- Validate the results with testbeds/realistic channel
- Develop an algorithm for allocation of broadcast links in TSCH

# Summary

Herein:

- Overview of the IETF 6TiSCH *minimal configuration* (6TiSCH-MC)
- Extensive simulations to characterize its behaviour

Conclusions:

- Potential downsides of 6TiSCH-MC with dense topologies
- Recommendations for setting TSCH and RPL parameters

Future work:

- Validate the results with testbeds/realistic channel
- Develop an algorithm for allocation of broadcast links in TSCH

# Summary

Herein:

- Overview of the IETF 6TiSCH *minimal configuration* (6TiSCH-MC)
- Extensive simulations to characterize its behaviour

Conclusions:

- Potential downsides of 6TiSCH-MC with dense topologies
- Recommendations for setting TSCH and RPL parameters

Future work:

- Validate the results with testbeds/realistic channel
- Develop an algorithm for allocation of broadcast links in TSCH

# Summary

**Herein:**

- Overview of the IETF 6TiSCH *minimal configuration* (6TiSCH-MC)
- Extensive simulations to characterize its behaviour

**Conclusions:**

- Potential downsides of 6TiSCH-MC with dense topologies
- Recommendations for setting TSCH and RPL parameters

**Future work:**

- Validate the results with testbeds/realistic channel
- Develop an algorithm for allocation of broadcast links in TSCH

Thank you!

Any questions?