Extracting Markov Chain Models from Protocol Execution Traces for End to End Delay Evaluation in WSNs

François Despaux
Université de Lorraine

IoT Lab Conference - Grenoble 2014

November 6, 2014

OUTLINE

Part 1 - Context

Part 2 - Novel Methodology for Modelling WSNs

Part 3 - Results & Contributions

Part 4 - Conclusions & Ongoing Work

Part I

Context

ESTIMATING END TO END (E2E) IN WSNS

- To be able to estimate the e2e delay in WSNs
  - Measurement
    - clock synchronization
    - delay in terms of average delay but not the probability distribution

ESTIMATING END TO END (E2E) IN WSNS

- To be able to estimate the e2e delay in WSNs
  - Measurement
    - clock synchronization
    - delay in terms of average delay but not the probability distribution
  - Simulation
    - Normally not enough accurate (radio model, capture effect, etc)
    - Operating System not taken into account

ESTIMATING END TO END (E2E) IN WSNS

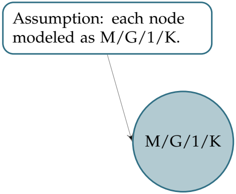
- To be able to estimate the e2e delay in WSNs
  - Measurement
    - clock synchronization
    - delay in terms of average delay but not the probability distribution
  - Simulation
    - Normally not enough accurate (radio model, capture effect, etc)
    - Operating System not taken into account
  - Analytic approach
    - Due to stochastic nature of WSNs and underlying MAC protocols: Markov chains

SOME LIMITATIONS OF EXISTING MARKOV CHAIN

- Misic et al., Park et al.
- Existing models limited to one-hop transmission scenarios.
- Poisson distribution assumptions (arrival rate).
- Why we **cannot** extend existing models to consider multi-hop transmission scenarios ?

SOME LIMITATIONS OF EXISTING MARKOV CHAIN

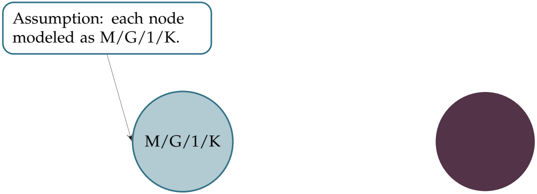
- Misic et al., Park et al.
- Existing models limited to one-hop transmission scenarios.
- Poisson distribution assumptions (arrival rate).
- Why we **cannot** extend existing models to consider multi-hop transmission scenarios ?



Assumption: each node modeled as M/G/1/K.

SOME LIMITATIONS OF EXISTING MARKOV CHAIN

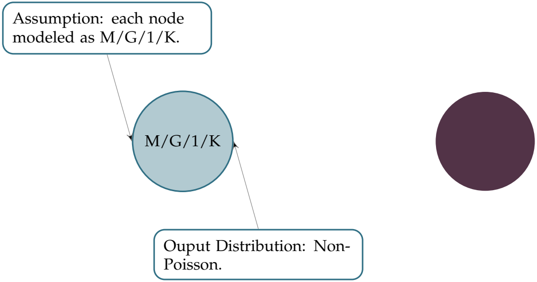
- Misic et al., Park et al.
- Existing models limited to one-hop transmission scenarios.
- Poisson distribution assumptions (arrival rate).
- Why we **cannot** extend existing models to consider multi-hop transmission scenarios ?



Assumption: each node modeled as M/G/1/K.

SOME LIMITATIONS OF EXISTING MARKOV CHAIN

- Misic et al., Park et al.
- Existing models limited to one-hop transmission scenarios.
- Poisson distribution assumptions (arrival rate).
- Why we **cannot** extend existing models to consider multi-hop transmission scenarios ?

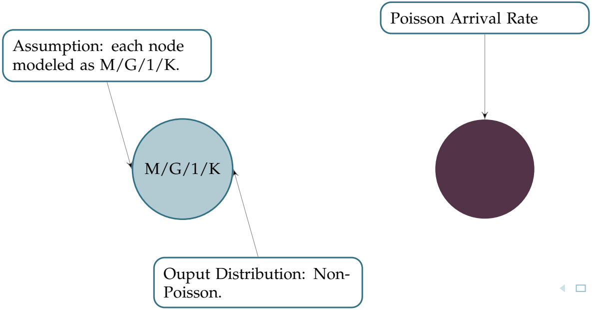


Assumption: each node modeled as M/G/1/K.

Ouput Distribution: Non-Poisson.

SOME LIMITATIONS OF EXISTING MARKOV CHAIN

- Misic et al., Park et al.
- Existing models limited to one-hop transmission scenarios.
- Poisson distribution assumptions (arrival rate).
- Why we **cannot** extend existing models to consider multi-hop transmission scenarios ?



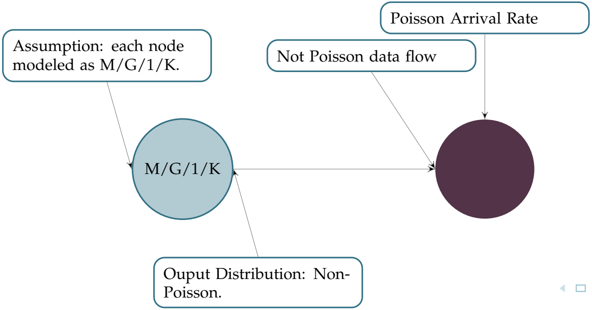
Poisson Arrival Rate

Assumption: each node modeled as M/G/1/K.

Ouput Distribution: Non-Poisson.

SOME LIMITATIONS OF EXISTING MARKOV CHAIN

- Misic et al., Park et al.
- Existing models limited to one-hop transmission scenarios.
- Poisson distribution assumptions (arrival rate).
- Why we **cannot** extend existing models to consider multi-hop transmission scenarios ?



Poisson Arrival Rate

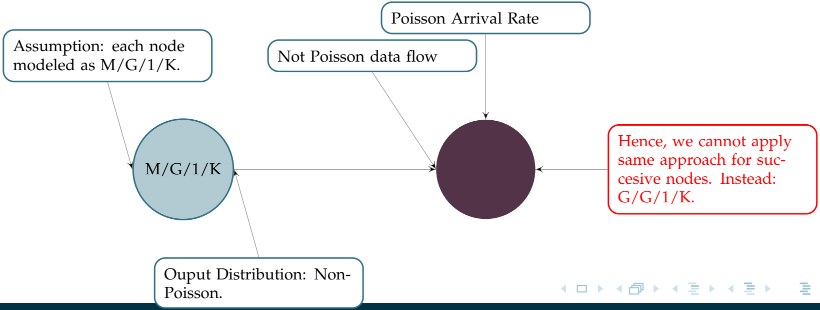
Assumption: each node modeled as M/G/1/K.

Not Poisson data flow

Ouput Distribution: Non-Poisson.

SOME LIMITATIONS OF EXISTING MARKOV CHAIN

- Misic et al., Park et al.
- Existing models limited to one-hop transmission scenarios.
- Poisson distribution assumptions (arrival rate).
- Why we **cannot** extend existing models to consider multi-hop transmission scenarios ?



Poisson Arrival Rate

Not Poisson data flow

Assumption: each node modeled as M/G/1/K.

Hence, we cannot apply same approach for successive nodes. Instead: G/G/1/K.

Ouput Distribution: Non-Poisson.

SOME LIMITATIONS (CONT)

- Underlying Operating System
  - In our previous work [1] we shown that the underlying OS introduces extra delays that affect the whole e2e delay

---

[1] On the Gap Between Mathematical Modelling and Measurement Analysis for Performance Evaluation of the 802.15.4 MAC Protocol - RTN 2013, Paris, France

SOME LIMITATIONS (CONT)

- Underlying Operating System
  - In our previous work [1] we shown that the underlying OS introduces extra delays that affect the whole e2e delay

Conclusions

- Proposed models are normally abstraction of the reality and sometimes not accurate for estimating performance parameters.
- The extension of the proposed model for a real WSN scenario is not straightforward (multi-hop scenario, for instance).

---

[1] On the Gap Between Mathematical Modelling and Measurement Analysis for Performance Evaluation of the 802.15.4 MAC Protocol - RTN 2013, Paris, France

Part II

Novel Methodology for Modelling WSNs

OBJECTIVE

- A novel approach
  - We combine measurement-based and analytic approaches based on process mining techniques for discovering a Markov chain model.
    - We discover a local Markov chain for each node.
    - By analysing the MAC protocol execution **log file**.
    - From this Markov chain we obtain the **one-hop delay distribution function** in one node.

OBJECTIVE

- A novel approach
  - We combine measurement-based and analytic approaches based on process mining techniques for discovering a Markov chain model.
    - We discover a local Markov chain for each node.
    - By analysing the MAC protocol execution **log file**.
    - From this Markov chain we obtain the **one-hop delay distribution function** in one node.
  - A mathematical technique for estimating the **e2e delay distribution function**.
    - Based on one-hop delay distributions found previously.

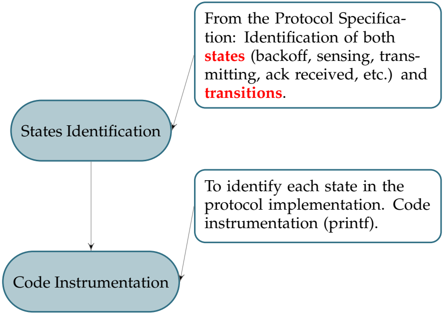
METHODOLOGY (MODELLING ONE NODE)

METHODOLOGY (MODELLING ONE NODE)

From the Protocol Specification: Identification of both **states** (backoff, sensing, transmitting, ack received, etc.) and **transitions**.

States Identification

METHODOLOGY (MODELLING ONE NODE)



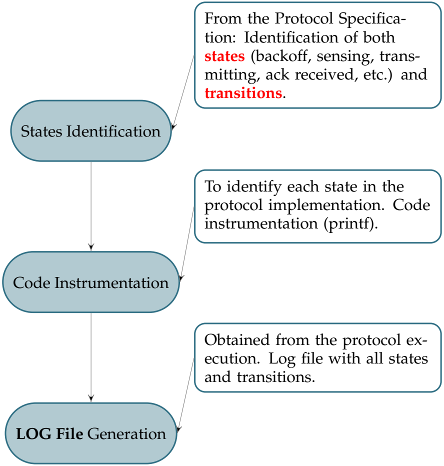
From the Protocol Specification: Identification of both **states** (backoff, sensing, transmitting, ack received, etc.) and **transitions**.

States Identification

To identify each state in the protocol implementation. Code instrumentation (printf).

Code Instrumentation

METHODOLOGY (MODELLING ONE NODE)



States Identification

From the Protocol Specification: Identification of both **states** (backoff, sensing, transmitting, ack received, etc.) and **transitions**.

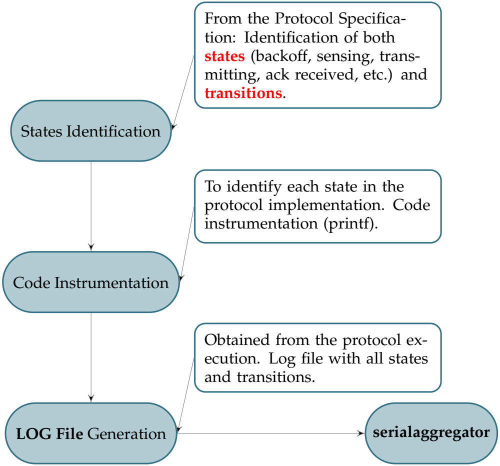
Code Instrumentation

To identify each state in the protocol implementation. Code instrumentation (printf).

LOG File Generation

Obtained from the protocol execution. Log file with all states and transitions.

METHODOLOGY (MODELLING ONE NODE)

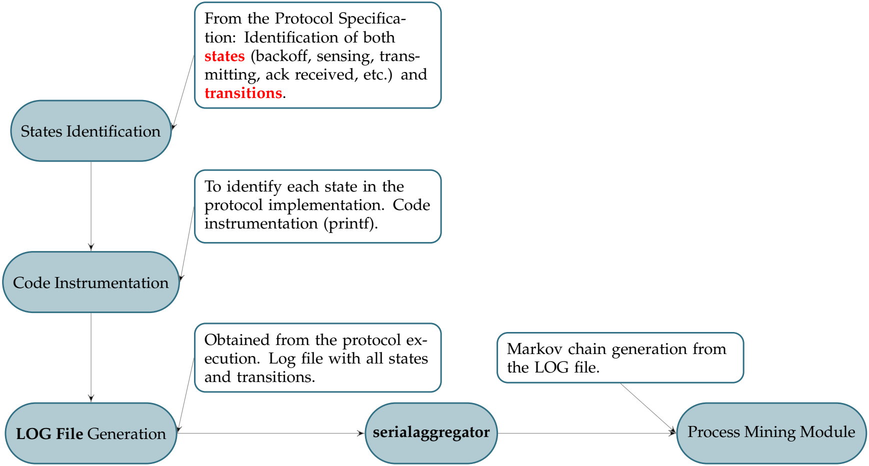


From the Protocol Specification: Identification of both **states** (backoff, sensing, transmitting, ack received, etc.) and **transitions**.

To identify each state in the protocol implementation. Code instrumentation (printf).

Obtained from the protocol execution. Log file with all states and transitions.

METHODOLOGY (MODELLING ONE NODE)



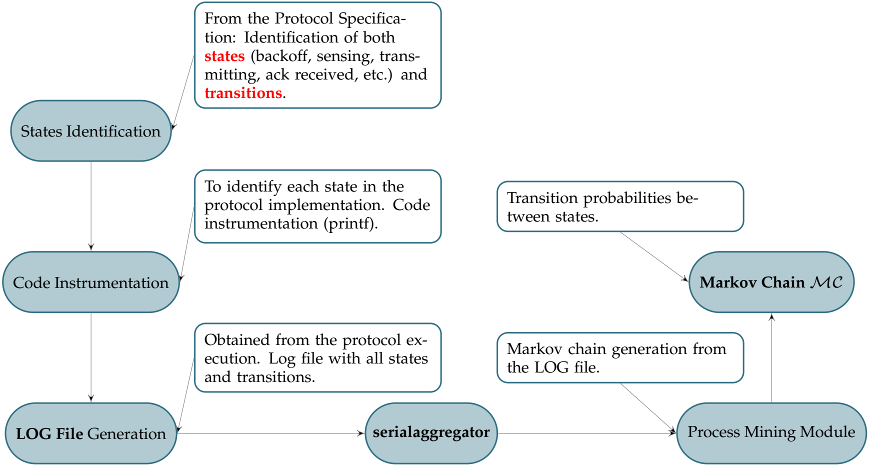
From the Protocol Specification: Identification of both **states** (backoff, sensing, transmitting, ack received, etc.) and **transitions**.

To identify each state in the protocol implementation. Code instrumentation (printf).

Obtained from the protocol execution. Log file with all states and transitions.

Markov chain generation from the LOG file.

METHODOLOGY (MODELLING ONE NODE)



From the Protocol Specification: Identification of both **states** (backoff, sensing, transmitting, ack received, etc.) and **transitions**.

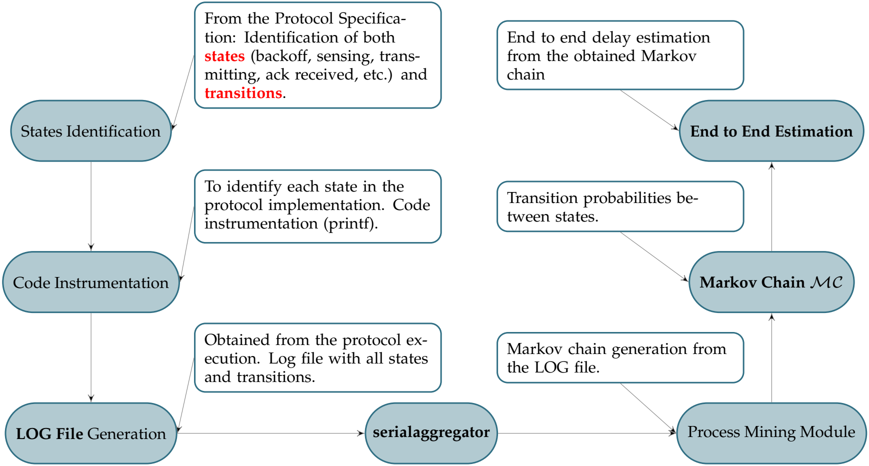
To identify each state in the protocol implementation. Code instrumentation (printf).

Obtained from the protocol execution. Log file with all states and transitions.

Transition probabilities between states.

Markov chain generation from the LOG file.

METHODOLOGY (MODELLING ONE NODE)



States Identification

From the Protocol Specification: Identification of both **states** (backoff, sensing, transmitting, ack received, etc.) and **transitions**.

Code Instrumentation

To identify each state in the protocol implementation. Code instrumentation (printf).

LOG File Generation

Obtained from the protocol execution. Log file with all states and transitions.

serialaggregator

Process Mining Module

Markov chain generation from the LOG file.

Markov Chain $\mathcal{MC}$

Transition probabilities between states.

End to End Estimation

End to end delay estimation from the obtained Markov chain

ONE-HOP DELAY FROM MARKOV CHAIN (1/4)

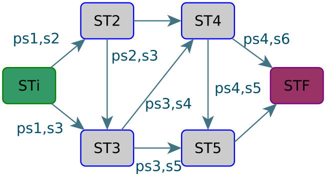


Figure: Markov chain (one node)

- **From the empirical $\mathcal{MC}$ we can obtain:**
  - States and transitions of the protocol.
  - Probability transitions between states.
  - Sojourn time on each state S_k, γ_{S_k}

ONE-HOP DELAY FROM MARKOV CHAIN (1/4)

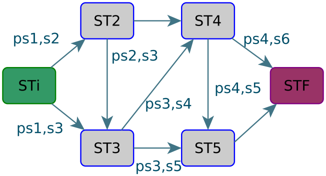


Figure: Markov chain (one node)

- **From the empirical $\mathcal{MC}$ we can obtain:**
  - States and transitions of the protocol.
  - Probability transitions between states.
  - Sojourn time on each state S_k, γ_{S_k}
- Then is it possible to create the Adjacency matrix $\mathcal{A}$

$$\mathcal{A} = \begin{pmatrix} 0 & p_{s_1 s_2} e_{s_1} & 0 & \cdots & 0 \\ 0 & 0 & p_{s_2 s_3} e_{s_2} & \cdots & 0 \\ \vdots & \vdots & 0 & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 0 \end{pmatrix} \quad (1)$$

ONE-HOP DELAY FROM MARKOV CHAIN (2/4)

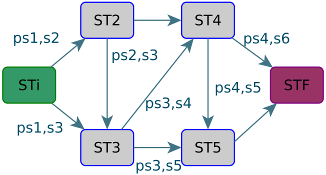


Figure: Markov chain (one node)

$$\mathcal{A} = \begin{pmatrix} 0 & p_{s_1 s_2} e_{s_1} & 0 & \cdots & 0 \\ 0 & 0 & p_{s_2 s_3} e_{s_2} & \cdots & 0 \\ \vdots & \vdots & 0 & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 0 \end{pmatrix}$$

- **Adjacency matrix $\mathcal{A}$**
  - e_{s_k} is the sojourn time distribution γ_{s_k} of state S_k in frequency domain[1] (Laplace Transform). Negative exponential distribution,
    - $$e_{s_k} = \frac{\gamma_{s_k}}{\gamma_{s_k} + s} \qquad (2)$$
  - p_{s_k,s_l} is the probability transition between states S_k and S_l.

---

[1] To avoid calculating convolutions over per-hop delay distribution

ONE-HOP DELAY FROM MARKOV CHAIN (3/4)

- Being s_f the final state (ACK RECEIVED), we compute

$$\vec{\mathcal{A}}^{r}_{s_k,s_f} = \mathcal{A} \cdot \vec{\mathcal{A}}^{r-1}_{s_k,s_f} \tag{3}$$

  - where $\vec{\mathcal{A}}^{1}_{s_k,s_f}$ is the vector containing the delay distribution from state s_k to s_f, path length = 1.

ONE-HOP DELAY FROM MARKOV CHAIN (3/4)

- Being s_f the final state (ACK RECEIVED), we compute

$$\vec{\mathcal{A}}^{r}_{s_k,s_f} = \mathcal{A} \cdot \vec{\mathcal{A}}^{r-1}_{s_k,s_f} \tag{3}$$

  - where $\vec{\mathcal{A}}^{1}_{s_k,s_f}$ is the vector containing the delay distribution from state s_k to s_f, path length = 1.

- Begin s_i the initial state, $\mathcal{A}^{r}_{s_i,s_f}$ gives the delay distribution from source to destination, path length = r, $r = \{1, 2, ...\}$.
- Then, the whole delay distribution in frequency domain can be computed as follows:

$$D_{f-dom}(s) = \sum_{r=1} \mathcal{A}^{r}_{s_i,s_f} \tag{4}$$

ONE-HOP DELAY FROM MARKOV CHAIN (4/4)

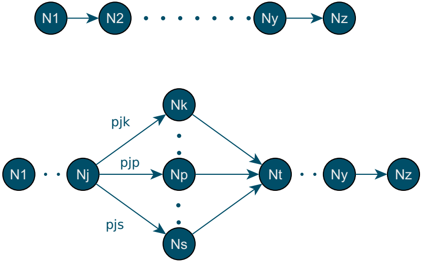
- By derivating $D_{f-dom}(s)$, we can obtain the average delay in time domain

$$\bar{D} = \left.\frac{\partial D_{f-dom}(s)}{\partial s}\right|_{s=0} \quad (5)$$

- By means of the *Inverse Laplace Transform (ILT)*, we obtain the delay distribution in time domain

$$D_{t-dom}(t) = ILT(D_{f-dom}(s)) \quad (6)$$

ESTIMATING END TO END DELAY



- The e2e delay distribution in frequency domain (serial)
  -
    $$D_{e2e(f-dom)}(s) = \prod_{i=1}^{y} D_{f-dom}^{(N_i)}(s)$$

- The e2e delay distribution in frequency domain (parallel)
  -
    $$D_{e2e(f-dom)}(s) = \prod_{i=1}^{j} D_{f-dom}^{(N_i)}(s) \cdot \left(\sum_{i=k}^{s} p_{j,i} \cdot D_{f-dom}^{(N_i)}(s) \right) \cdot \prod_{i=t}^{y} D_{f-dom}^{(N_i)}(s)$$

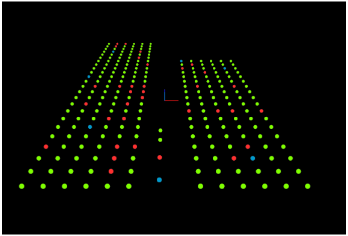
Part III

Results & Contributions

CONTRIBUTIONS

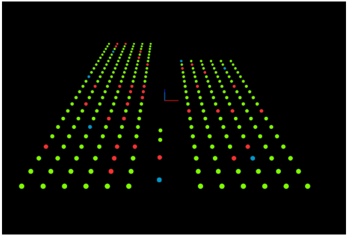
- **X-MAC & RPL**
  - Dynamic routing.
  - Tested in a Large-scale infrastructure (IoT-Senslab)
  - Comparition between routing strategies in terms of e2e delay.
- Two more contributions
  - **ContikiMAC**
    - IEEE DCOSS 2014, Marina del Rey, Californie, May 26 - 28.
  - **Standard IEEE 802.15.4** (slotted version)
    - IEEE ISCC 2014, Madeira, Portugal, June 23-26.

X-MAC & RPL



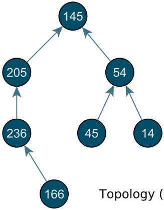
- **X-MAC & RPL Scenario**
  - INRIA Rennes with 256 WSN430 open nodes.
  - Seven nodes from the testbed were selected to carry out the experiments.
  - Buffer size = 8.
    Poisson arrival rate $\lambda = 0.5, 1, 2, 4$ p/s.
    42 bytes (25 bytes of payload + 17 bytes of header).

X-MAC & RPL

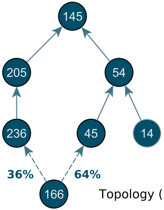


- **X-MAC & RPL Scenario**
    - INRIA Rennes with 256 WSN430 open nodes.
    - Seven nodes from the testbed were selected to carry out the experiments.
    - Buffer size = 8.
      Poisson arrival rate $\lambda = 0.5, 1, 2, 4$ p/s.
      42 bytes (25 bytes of payload + 17 bytes of header).
- Two metrics of the RPL were considered
    - RPL objective function 0 (OF0) (number of hops)
    - RPL objective function ETX (Expected number of transmissions)

X-MAC & RPL



Topology (OF0)



Topology (ETX)

- **Average e2e delay (OF0 & ETX).**

OF0	Emp. Global Av. Delay		Theo. Global Av. Delay	
λ	One-Hop	e2e	One-Hop	e2e
0.5	0,1409	0,2406	0,1408	0,2507
1	0,1507	0,2439	0,1488	0,2580
2	0,1556	0,2531	0,1518	0,2644
4	0,1864	0,3189	0,1846	0,3202

ETX	Emp. Global Av. Delay		Theo. Global Av. Delay	
λ	One-Hop	e2e	One-Hop	e2e
0.5	0,1503	0,2497	0,1482	0,2516
1	0,1467	0,2427	0,1472	0,2530
2	0,1613	0,2605	0,1569	0,2717
4	0,1685	0,2981	0,1627	0,2870

- **Packet reception rate for OF0 & ETX.**

	OF0	ETX
λ	PRR (%)	PRR (%)
0.5 p/sec	97	98
1 p/sec	96	95
2 p/sec	33	35
4 p/sec	2.8	9.9

X-MAC & RPL

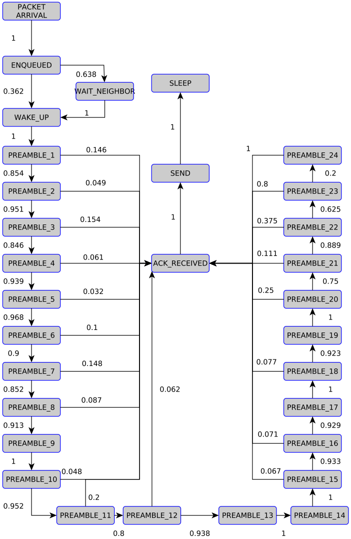
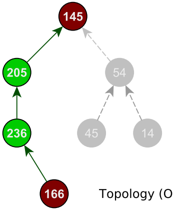
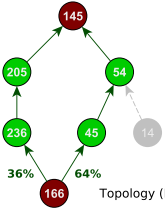


Figure: Markov chain: $\lambda = 4$

X-MAC & RPL



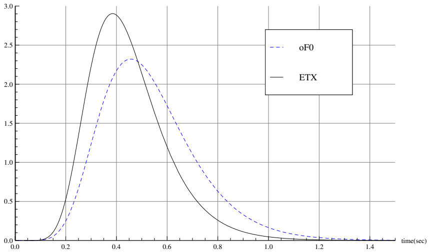
Topology (OF0)



Topology (ETX)

- **PDF of the e2e delay**
  - From node 166 to the sink (145)
  - For both RPL objective functions

E2e delay distribution: OF0 and ETX for node 166 (lambda = 4p/s)



LIMITATIONS OF OUR METHODOLOGY

- Obtained model depends strictly on the input parameters (traffic).
  - To generate traces for several scenarios with different traffic patterns to draw more general conclusions.

LIMITATIONS OF OUR METHODOLOGY

- Obtained model depends strictly on the input parameters (traffic).
    - To generate traces for several scenarios with different traffic patterns to draw more general conclusions.

- It is imperative to have the source code of the protocol to be able to instrument it.

LIMITATIONS OF OUR METHODOLOGY

- Obtained model depends strictly on the input parameters (traffic).
  - To generate traces for several scenarios with different traffic patterns to draw more general conclusions.
- It is imperative to have the source code of the protocol to be able to instrument it.
- Code instrumentation using printf-like instructions affects the execution timing and non-intrusive approaches are not easy to implement.

Part IV

Conclusions & Ongoing Work

CONCLUSIONS

- Contributions
  - A novel methodology to obtain a Markov chain model from any MAC protocol by analysing execution traces.

CONCLUSIONS

- Contributions
    - A novel methodology to obtain a Markov chain model from any MAC protocol by analysing execution traces.
    - A mathematical technique for estimating the **e2e delay distribution** in both one-hop and multi-hop transmission scenarios.

CONCLUSIONS

- Contributions
  - A novel methodology to obtain a Markov chain model from any MAC protocol by analysing execution traces.
  - A mathematical technique for estimating the **e2e delay distribution** in both one-hop and multi-hop transmission scenarios.
  - Our methodology is suitable for modelling WSNs in real testbed scenarios taking into account, not only the underlying duty-cycled MAC protocol, but also a dynamic routing protocol.

ONGOING WORK

- To study the influence of the arrival rate in the Markov chain (generalization).

ONGOING WORK

- To study the influence of the arrival rate in the Markov chain (generalization).
- Considering the independence of our methodology with regard to the overlying routing protocol
  - To compare routing protocol's performance in terms of end to end delay (RPL vs OC-Routing).

QUESTIONS ?


tak
hvala
Баярлалаа
dank u
thank you
arigatô
danke
ačiū
благодарю
gràcies
nandri
gracias
grazzi
choukran
asante
Благодарам
Дякую
diolch
mamnun
ngiyabonga
hvala
hvala
khob chai
djiere dieuf
multumesc
akun
merci
takk
shukriya
kop khun
köszönöm
Mh'gōi
obrigado
terima kasih
Xièxiè