IEEE 802.11 Distribute Coordinator Function Discrete–Event Simulator against State–of–the–Art Analytical Model

Simulation and Performance Evaluation project

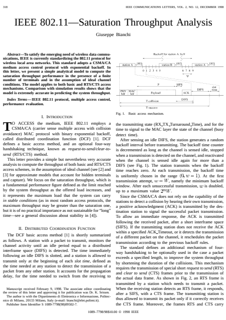
Luca Mosetti, Shandy Darma

November, 2023

Table of Contents
1 Related works

Analytical Models

1 Related works



318 IEEE COMMUNICATIONS LETTERS, VOL. 2, NO. 12, DECEMBER 1998

IEEE 802.11—Saturation Throughput Analysis

Giuseppe Bianchi

IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS, VOL. 18, NO. 3, MARCH 2000 535

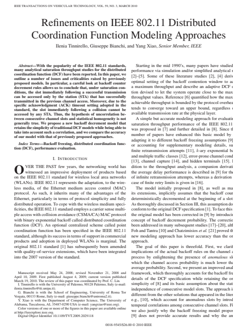
Performance Analysis of the IEEE 802.11 Distributed Coordination Function

Giuseppe Bianchi

IEEE COMMUNICATIONS LETTERS, VOL. 9, NO. 8, AUGUST 2005 765

Remarks on IEEE 802.11 DCF Performance Analysis

Giuseppe Bianchi and Ilenia Tinnirello



IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, VOL. 59, NO. 3, MARCH 2010 1055

Refinements on IEEE 802.11 Distributed Coordination Function Modeling Approaches

Ilenia Tinnirello, Giuseppe Bianchi, and Yang Xiao, *Senior Member, IEEE*

Table of Contents

2 Distribute Coordinator Function

Introduction

- DCF follows Carrier Sense Multiple Access / Collision Avoidance principles
- Stations (STAs) use a "listen-before-talk" approach
- Key mechanisms: carrier sensing, random backoff, ACK

ACK

- Positive acknowledgment after successful reception
- Transmitted at the end of the packet after a SIFS
- If no ACK, reschedule transmission

Carrier Sensing Mechanism

- STAs sense the medium before transmission
- If a STA senses the channel idle for DIFS, it generates a random backoff period

Backoff strategy

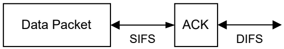
- Backoff period expressed in backoff slots
- Decreases when the medium is sensed idle
- "Frozen" during a detected transmission

- At each transmission, backoff period uniformly chosen in $[0, CW)$
- After unsuccessful transmission, $CW = \min \begin{pmatrix} 2 \cdot CW \\ 2^m \cdot W \end{pmatrix}$
- Resets to W after success or retry limit R reached

BAS & RTS/CTS

2 Distribute Coordinator Function

BAS mode



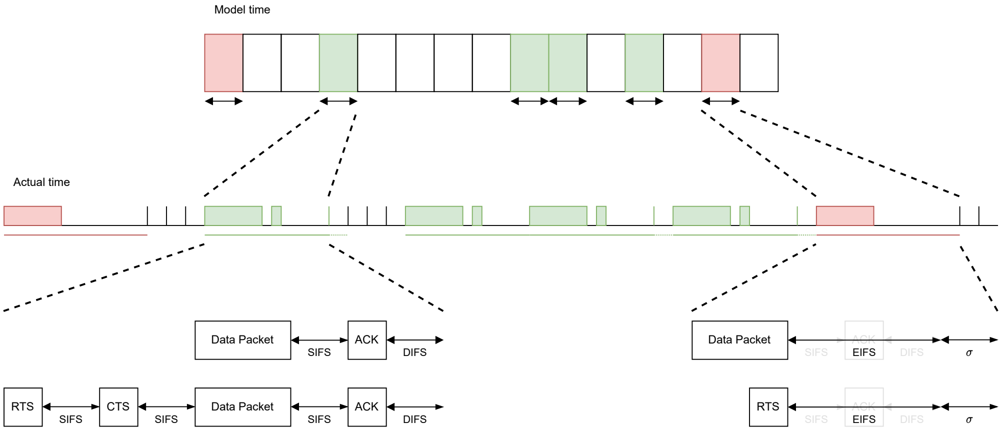
- Default mechanism
- Transmit directly the entire packet

RTS/CTS mode

- Four-way handshaking mechanism
- Request-to-Send (RTS)
- Clear-to-Send (CTS)

Times

2 Distribute Coordinator Function



Times

2 Distribute Coordinator Function

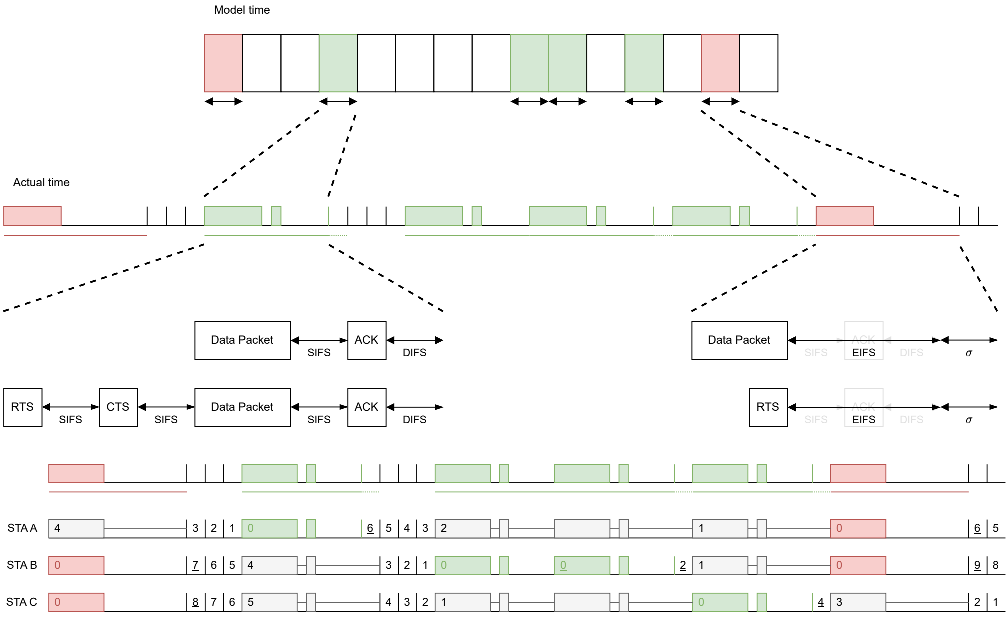
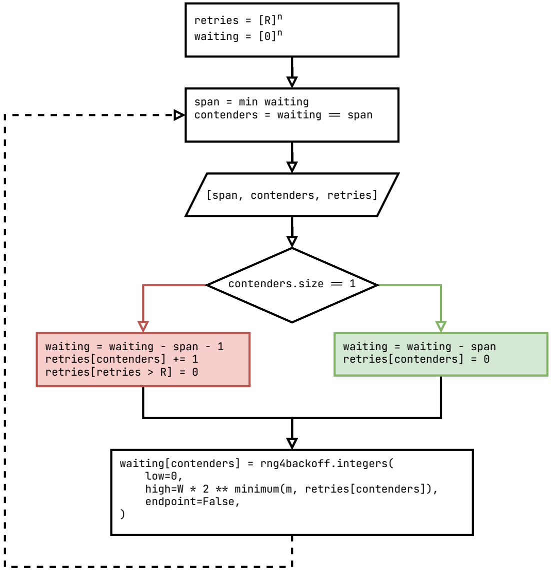


Table of Contents

3 Problem definition & Setup

Simulator

3 Problem definition & Setup



Characteristics:

- Discrete-Event Simulator
- Independent of any temporal specification
- Python 3.11

Assumptions:

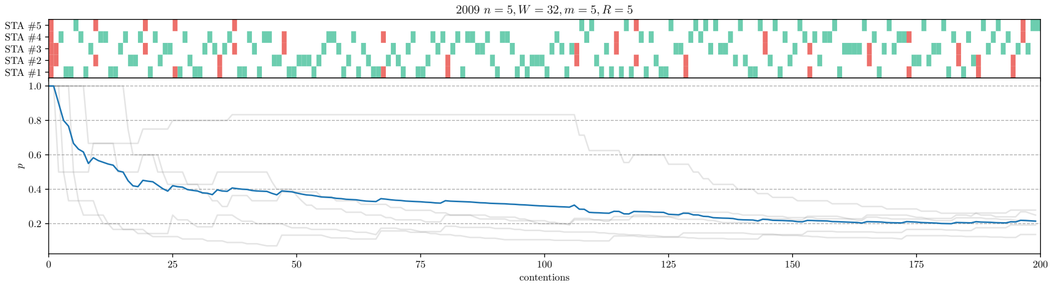
- Constant # of STAs
- Constant payload size
- Transmission queue of each STA is always nonempty
- W/out hidden terminals
- W/out capture effect

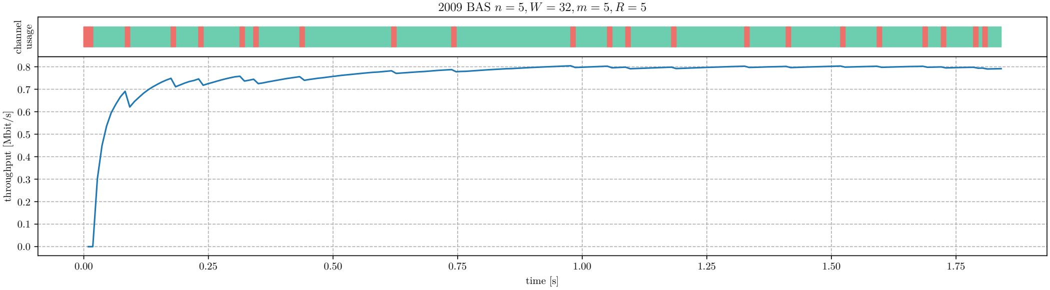
Table of Contents

4 Metrics

p and S

4 Metrics





Probability of collision

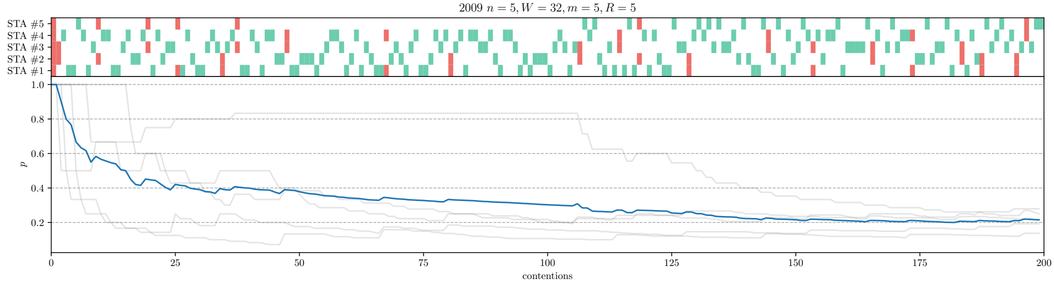


Figure: Measurement of collision probability p. Upper figure shows whether a transmission or a collision occurred in each contention round. Lower figure shows the collision probability, with grey lines describe probability of each stations while blue line describes the final mean of all probabilities.

Saturated throughput

4 Metrics

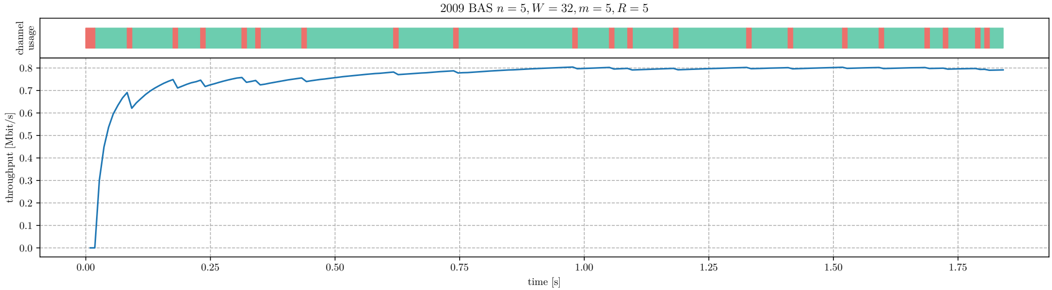
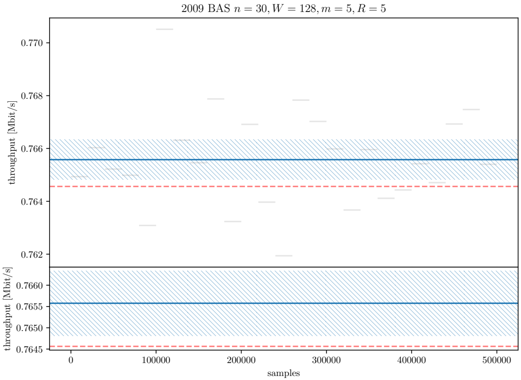


Figure: Measurement of saturated throughput S. Upper figure shows the occurrences of transmissions and collisions. Lower figure shows the saturated throughput of the channel. Each transmission improves throughput while each collision worsens throughput.

Non-overlapping batch over means

4 Metrics



$$\underbrace{Y_1, \ldots, Y_s}_{B_1}, \underbrace{Y_{s+1}, \ldots, Y_{2s}}_{B_2}, \cdots, \underbrace{Y_{(b-1)s+1}, \ldots, Y_{bs}}_{B_b}$$

$$\overline{Z}_b = \frac{1}{b} \sum_i^b Z_i \qquad \left[\overline{Z}_b \pm t_{b-1, \frac{1+\gamma}{2}} \sqrt{\frac{\hat{V}}{b}} \right]_\gamma$$

$$\hat{V} = \frac{1}{b-1} \sum_i^b (Z_i - \overline{Z}_b)^2$$

Table of Contents

5 Analytical Model

Formulae

σ	slot time
L	payload size
T_{MPDU}	the time to transmit the MPDU
T_{ACK}	the time to transmit an ACK
T_{RTS}	the time to transmit a RTS frame
T_{CTS}	the time to transmit a CTS frame
SIFS	the SIFS time
DIFS	the DIFS time
n	number of stations
W	minimum contention window
m	maximum backoff stage, where maximum contention window is $2^m W$
R	maximum number of retries, after which the transmission is dropped
p	probability of collision
τ	probability that a station will transmit
S	saturation throughput

Formulae

$$p = 1 - (1-\tau)^{n-1}$$

$$\boldsymbol{\tau} = \frac{1}{1 + \frac{1-p}{2(1-p^{R+1})}\left[\sum_{j=0}^{R} p^j \cdot (2^j W - 1) - (1 - p^{R+1})\right]}$$

$$\mathbf{S} = \frac{\mathbf{P_s}\overline{\mathbf{E}[L]}}{(1-\mathbf{P_b})\sigma + \mathbf{P_s}\overline{\mathbf{T_s}} + (\mathbf{P_b} - \mathbf{P_s})\overline{\mathbf{T_c}}}$$

$$\mathbf{P_b} = 1 - (1-\boldsymbol{\tau})^n \qquad \mathbf{P_s} = n\boldsymbol{\tau}(1-\boldsymbol{\tau})^{n-1}$$

$$\overline{\mathbf{E}[L]} = L\frac{W}{W-1} \qquad \overline{\mathbf{T_s}} = \mathbf{T_s}\frac{W}{W-1} + \sigma \qquad \overline{\mathbf{T_c}} = \mathbf{T_c} + \sigma$$

$$\mathbf{T_s^{BAS}} = T_{\mathrm{MPDU}} + \mathrm{SIFS} + T_{\mathrm{ACK}} + \mathrm{DIFS}$$

$$\mathbf{T_c^{BAS}} = \mathbf{T_s^{BAS}}$$

$$\mathbf{T_s^{RTS/CTS}} = T_{\mathrm{RTS}} + \mathrm{SIFS} + T_{\mathrm{CTS}} + \mathrm{SIFS} + T_{\mathrm{MPDU}} + \mathrm{SIFS} + T_{\mathrm{ACK}} + \mathrm{DIFS}$$

$$\mathbf{T_c^{RTS/CTS}} = T_{\mathrm{RTS}} + \mathrm{SIFS} + T_{\mathrm{ACK}} + \mathrm{DIFS}$$

Formulae

$$\tau = \frac{1}{1 + \frac{1-p}{2(1-p^{R+1})}\left[\sum_{j=0}^{R} p^j \cdot (2^j W - 1) - (1 - p^{R+1})\right]}$$

$$\downarrow$$

$$\tau = \frac{1}{1 + \frac{1-p}{2(1-p^{R+1})}\left[\sum_{j=0}^{R} p^j \cdot (\mathfrak{W}_{\mathsf{j}} - 1) - (1 - p^{R+1})\right]}$$

$$\mathfrak{W}_{\mathsf{j}} = W \cdot 2^{\min(m,j)} \quad j \in [0, R]$$

Table of Contents

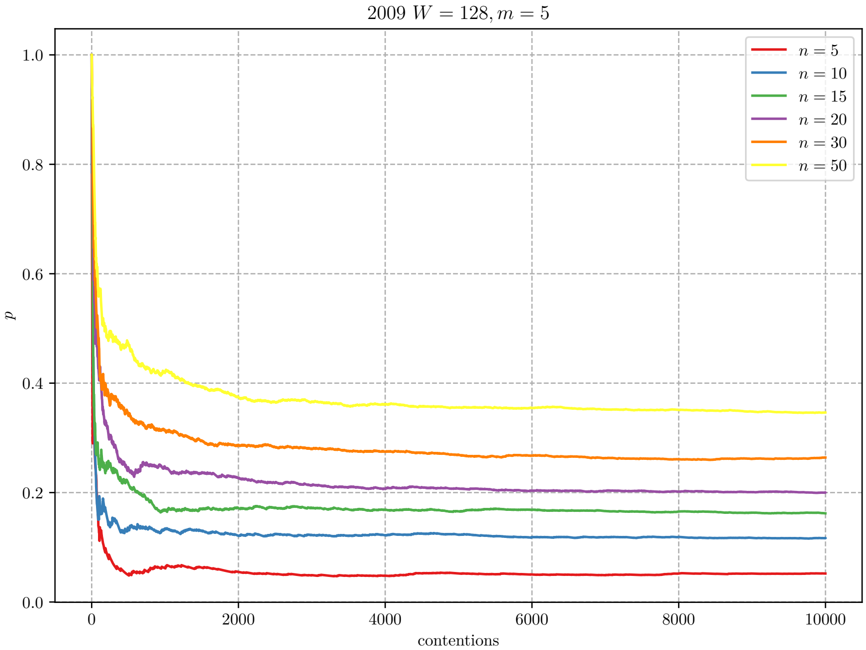
6 Results

Parameters

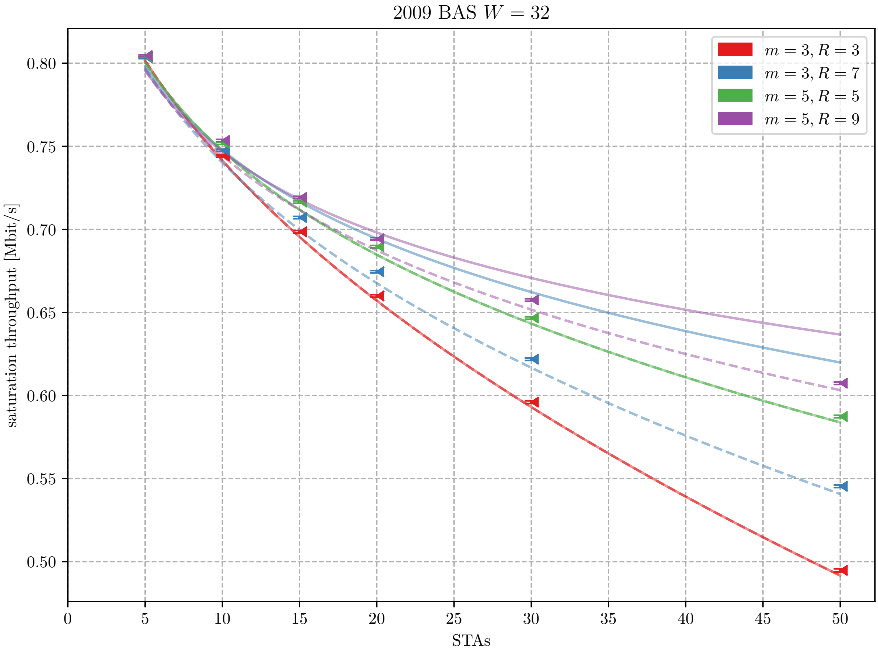
Parameter	2009 **Simulator**
L	1023 Byte
MAC header	34 Byte
PHY header	16 Byte
ACK	14 Byte + PHY header
RTS	20 Byte + PHY header
CTS	14 Byte + PHY header
Channel Bit Rate	1 Mbit/s
σ	50 μs
SIFS	28 μs
DIFS	SIFS $+ 2 \cdot \sigma$

Initialisation bias

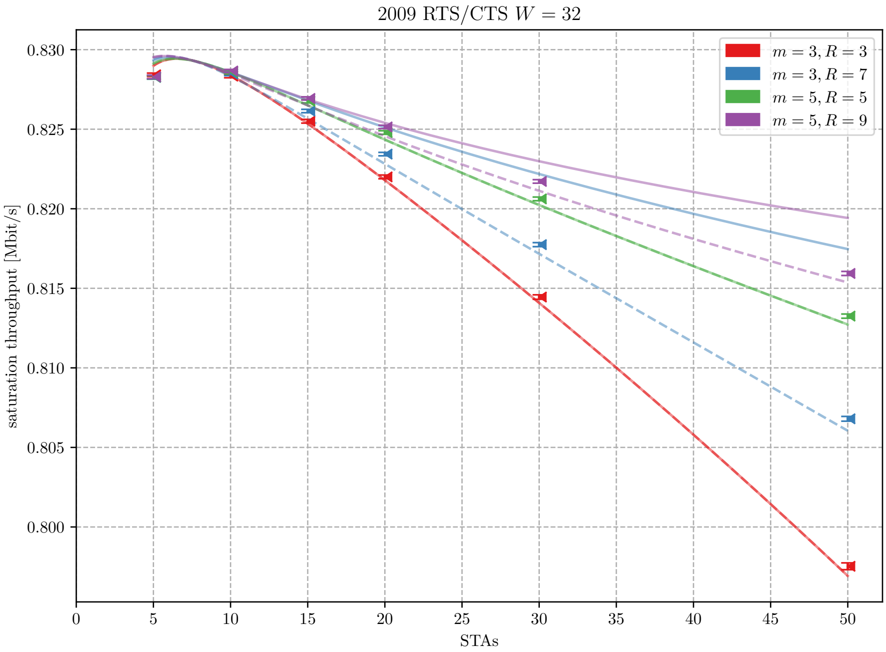
6 Results



Saturation throughput

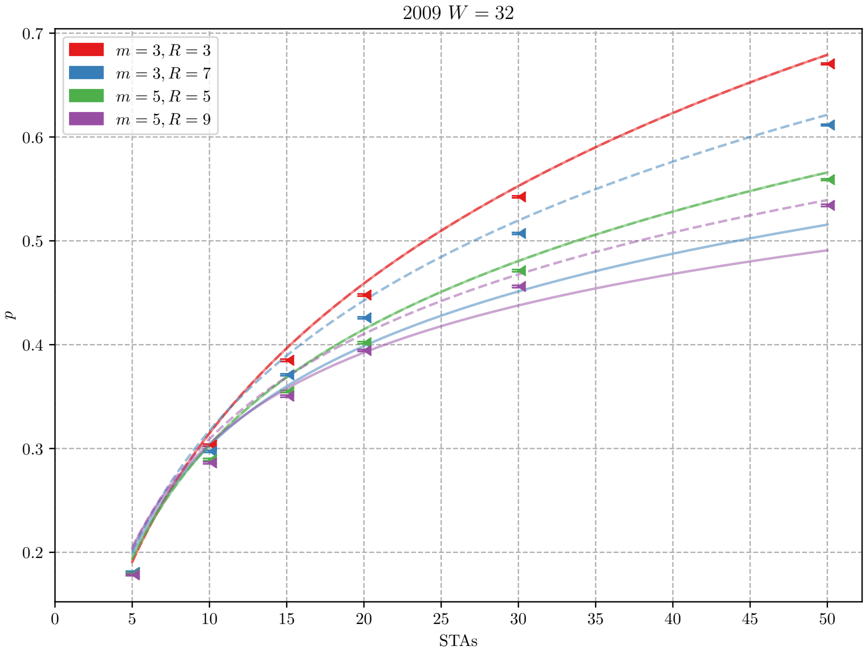


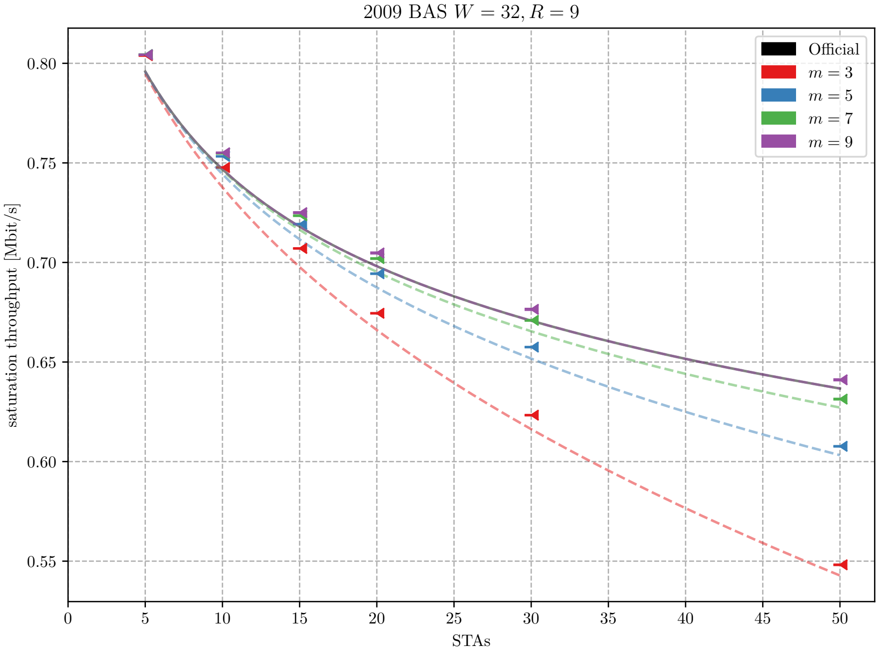
Saturation throughput

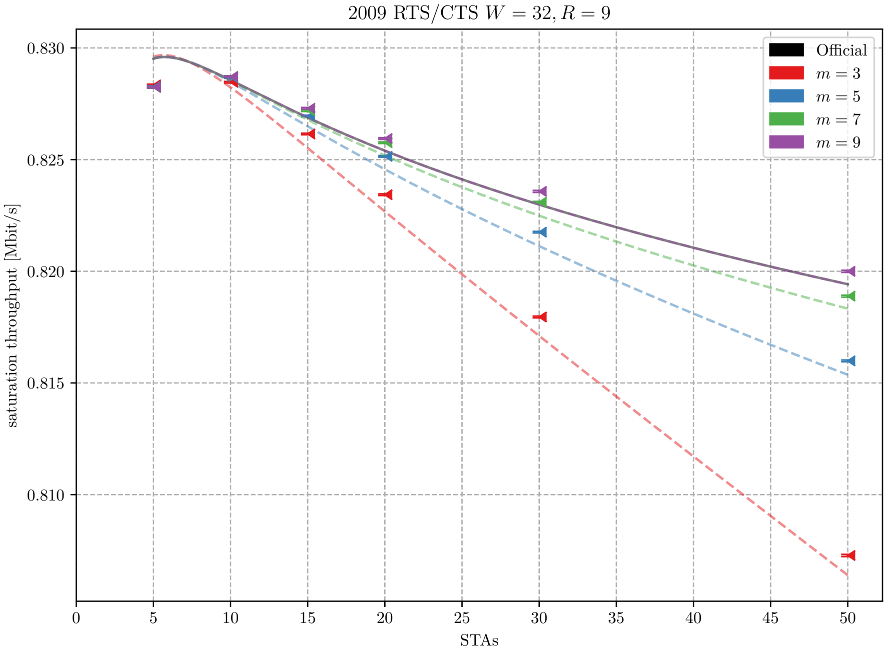


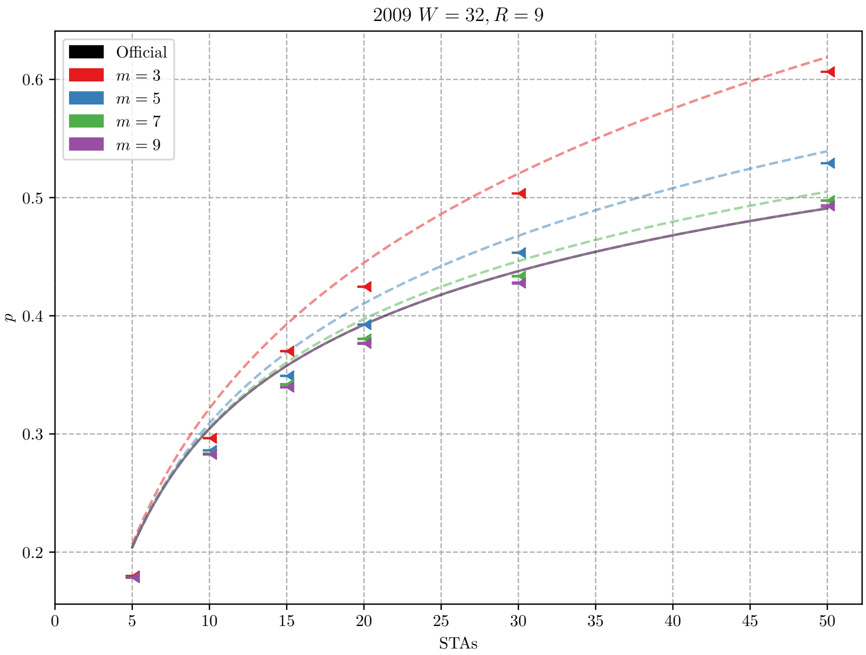
Probability of collision

6 Results



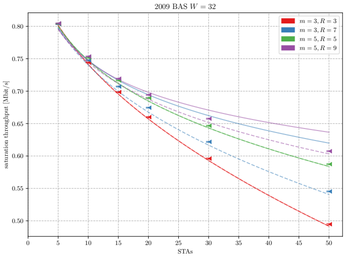

2009 BAS $W = 32, R = 9$
Official
$m = 3$
$m = 5$
$m = 7$
$m = 9$
saturation throughput [Mbit/s]
0.80
0.75
0.70
0.65
0.60
0.55
0
5
10
15
20
25
30
35
40
45
50
STAs

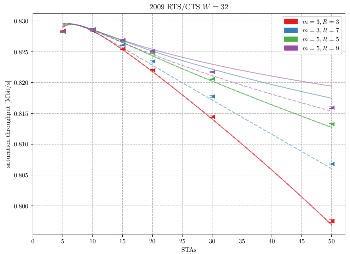

2009 RTS/CTS $W = 32, R = 9$
saturation throughput [Mbit/s]
STAs
Official
$m = 3$
$m = 5$
$m = 7$
$m = 9$
0.830
0.825
0.820
0.815
0.810
0
5
10
15
20
25
30
35
40
45
50

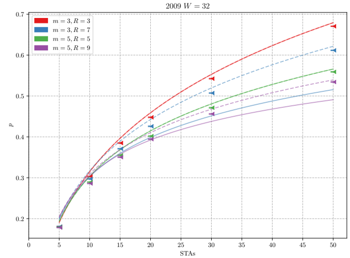


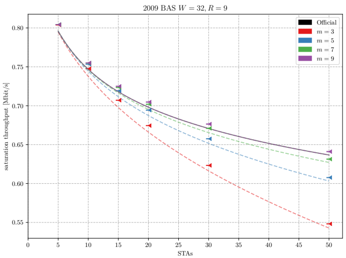
2009 $W = 32, R = 9$
Official
$m = 3$
$m = 5$
$m = 7$
$m = 9$
p
0.2
0.3
0.4
0.5
0.6
0
5
10
15
20
25
30
35
40
45
50
STAs

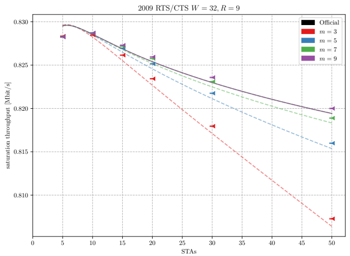
All in all
6 Results











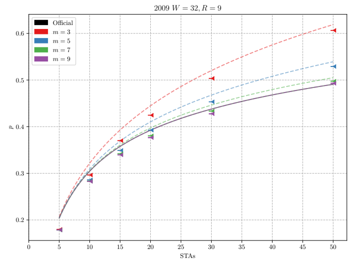


Table of Contents

7 Appendix

Differences between analytical models

$$p = 1 - (1-\tau)^{n-1}$$

$$\tau = \frac{2}{1 + W + pW\sum_{i=0}^{m-1}(2p)^i}$$

$$\boldsymbol{\tau} = \frac{1}{1 + \frac{1-p}{2(1-p^{R+1})}\left[\sum_{j=0}^{R} p^j \cdot (2^j W - 1) - (1 - p^{R+1})\right]}$$

$$\mathsf{S} = \frac{\mathsf{P_s}\mathsf{P_{tr}}L}{(1-\mathsf{P_{tr}})\sigma + \mathsf{P_{tr}}\mathsf{P_s}\mathsf{T_s} + \mathsf{P_{tr}}(1-\mathsf{P_s})\mathsf{T_c}}$$

$$\mathbf{S} = \frac{\mathbf{P_s}\overline{\mathbf{E}[L]}}{(1-\mathbf{P_b})\sigma + \mathbf{P_s}\overline{\mathbf{T_s}} + (\mathbf{P_b} - \mathbf{P_s})\overline{\mathbf{T_c}}}$$

$$\mathsf{P_{tr}} = 1 - (1-\tau)^n \qquad \mathsf{P_s} = \frac{n\tau(1-\tau)^{n-1}}{\mathsf{P_{tr}}}$$

$$\mathbf{P_b} = 1 - (1-\boldsymbol{\tau})^n \qquad \mathbf{P_s} = n\boldsymbol{\tau}(1-\boldsymbol{\tau})^{n-1}$$

$$\overline{\mathbf{E}[L]} = L\frac{W}{W-1} \qquad \overline{\mathbf{T_s}} = \mathbf{T_s}\frac{W}{W-1} + \sigma \qquad \overline{\mathbf{T_c}} = \mathbf{T_c} + \sigma$$

Differences between analytical models

$$
\begin{aligned}
\mathsf{T}_{\mathsf{s}}^{\mathsf{BAS}} &= T_{\text{MPDU}} + \delta + \text{SIFS} + T_{\text{ACK}} + \delta + \text{DIFS} \\
\mathsf{T}_{\mathsf{c}}^{\mathsf{BAS}} &= T_{\text{MPDU}} + \delta + \text{DIFS}
\end{aligned}
$$

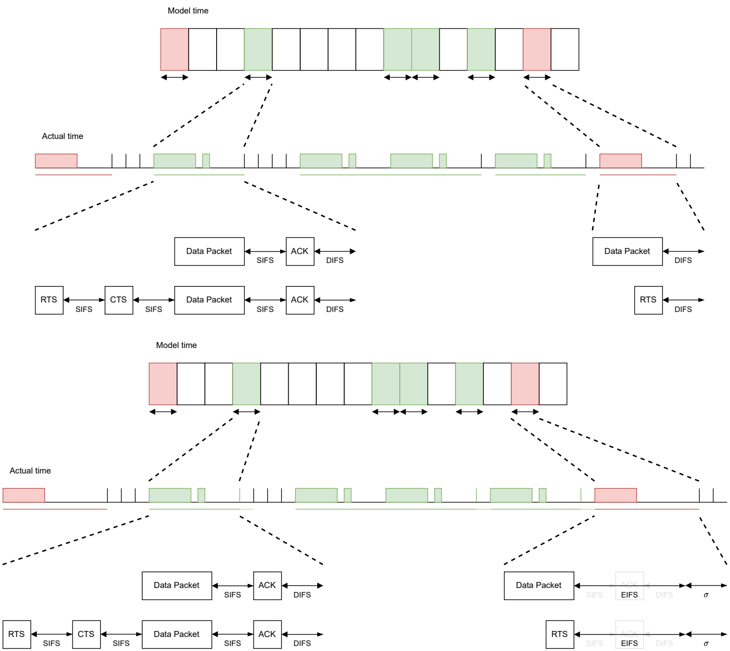
$$
\begin{aligned}
\mathbf{T}_{\mathbf{s}}^{\mathbf{BAS}} &= T_{\text{MPDU}} + \text{SIFS} + T_{\text{ACK}} + \text{DIFS} \\
\mathbf{T}_{\mathbf{c}}^{\mathbf{BAS}} &= \mathbf{T}_{\mathbf{s}}^{\mathbf{BAS}}
\end{aligned}
$$

$$
\begin{aligned}
\mathsf{T}_{\mathsf{s}}^{\mathsf{RTS/CTS}} &= T_{\text{RTS}} + \delta + \text{SIFS} + T_{\text{CTS}} + \delta + \text{SIFS} \\
&\quad + T_{\text{MPDU}} + \delta + \text{SIFS} + T_{\text{ACK}} + \delta + \text{DIFS} \\
\mathsf{T}_{\mathsf{c}}^{\mathsf{RTS/CTS}} &= T_{\text{RTS}} + \delta + \text{DIFS}
\end{aligned}
$$

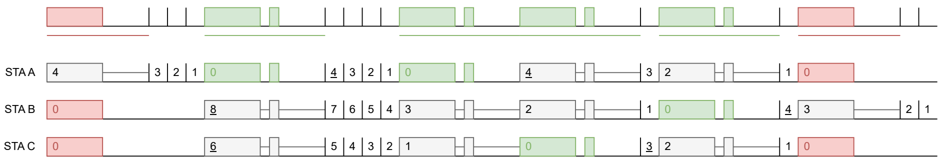
$$
\begin{aligned}
\mathbf{T}_{\mathbf{s}}^{\mathbf{RTS/CTS}} &= T_{\text{RTS}} + \text{SIFS} + T_{\text{CTS}} + \text{SIFS} \\
&\quad + T_{\text{MPDU}} + \text{SIFS} + T_{\text{ACK}} + \text{DIFS} \\
\mathbf{T}_{\mathbf{c}}^{\mathbf{RTS/CTS}} &= T_{\text{RTS}} + \text{SIFS} + T_{\text{ACK}} + \text{DIFS}
\end{aligned}
$$

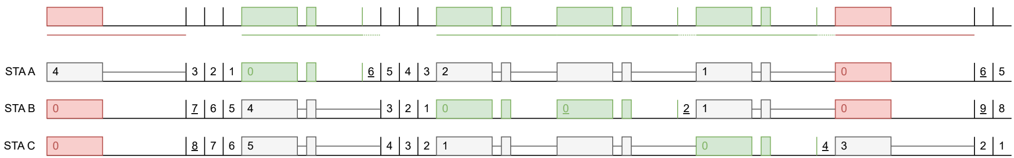
Differences between simulators

7 Appendix

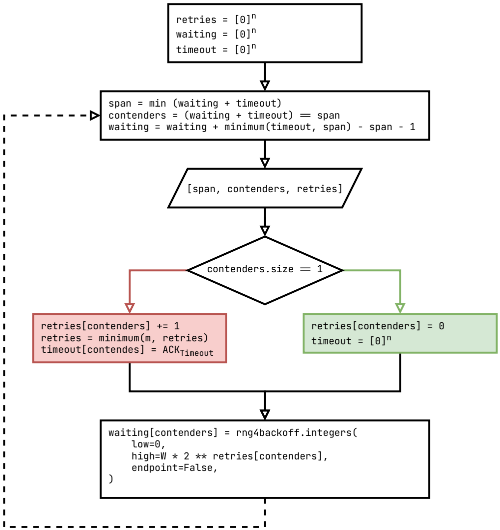


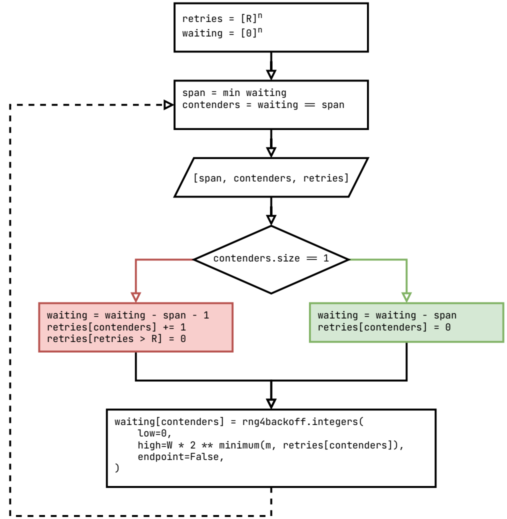
Differences between simulators





Differences between simulators





simulation()

```python
        seeds: t.Iterator[NDArray[np.uint32]],
        n: int,
        W: int,
        m: int,
        ack_timeout: int,
) -> t.Iterator[Log]:
    rng4backoff: np.random.Generator = np.random.default_rng(next(seeds))

    timeouts: NDArray[bool] = np.full(n, fill_value=False)
    retries: NDArray[int] = np.zeros(n, dtype=int)
    waiting: NDArray[int] = np.zeros(n, dtype=int)

    while True:

        span: int = np.amin(waiting + timeouts * ack_timeout)
        mask: NDArray[bool] = (waiting + timeouts * ack_timeout) == span
        contenders: NDArray[int] = np.flatnonzero(mask)

        yield Log(
            span=span,
            contenders=mask,
            attempt=retries.copy(),
        )

        waiting = waiting + np.minimum(timeouts * ack_timeout, span) - span - 1

        match contenders.size:
            case 1:
                timeouts = np.full(n, fill_value=False)
                retries[contenders] = 0
            case _:
                timeouts = mask
                retries[contenders] += 1
                retries.clip(max=m, out=retries)

        waiting[contenders] = rng4backoff.integers(
            low=0,
            high=W * 2 ** retries[contenders],
            endpoint=False,
        )
```

simulation()

```python
        seeds: t.Iterator[NDArray[np.uint32]],
        n: int,
        W: int,
        m: int,
        R: int,
) -> t.Iterator[Log]:
    rng4backoff: np.random.Generator = np.random.default_rng(next(seeds))

    retries: NDArray[int] = np.full(n, R, dtype=int)
    waiting: NDArray[int] = np.zeros(n, dtype=int)

    while True:

        span: int = np.amin(waiting)
        mask: NDArray[bool] = waiting == span
        contenders: NDArray[int] = np.flatnonzero(mask)

        yield Log(
            span=span,
            contenders=mask,
            attempt=retries.copy(),
        )

        waiting = waiting - span

        match contenders.size:
            case 1:
                retries[contenders] = 0
            case _:
                waiting -= 1
                retries[contenders] += 1
                retries[retries > R] = 0

        waiting[contenders] = rng4backoff.integers(
            low=0,
            high=W * 2 ** np.minimum(m, retries[contenders]),
            endpoint=False,
        )
```

Table of Contents

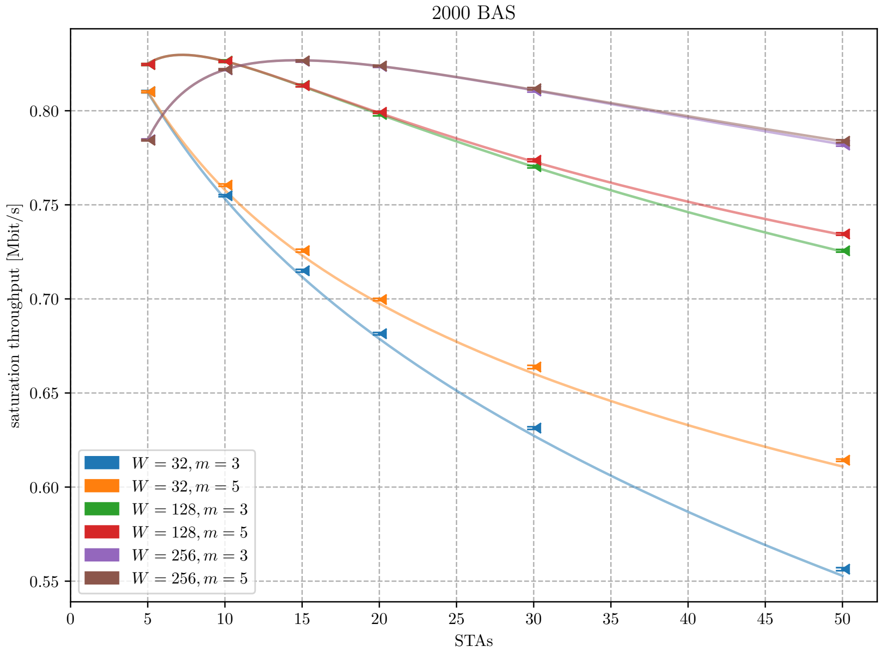
8 Results

Parameters

Parameter	2000 **Simulator**	2009 **Simulator**
L	1023 Byte	1023 Byte
MAC header	34 Byte	34 Byte
PHY header	16 Byte	16 Byte
ACK	14 Byte + PHY header	14 Byte + PHY header
RTS	20 Byte + PHY header	20 Byte + PHY header
CTS	14 Byte + PHY header	14 Byte + PHY header
Channel Bit Rate	1 Mbit/s	1 Mbit/s
σ	50 μs	50 μs
SIFS	28 μs	28 μs
DIFS	SIFS $+ 2 \cdot \sigma$	SIFS $+ 2 \cdot \sigma$
δ	1 μs	–
$\mathsf{ACK}_{\mathsf{Timeout}}$	300 μs	–

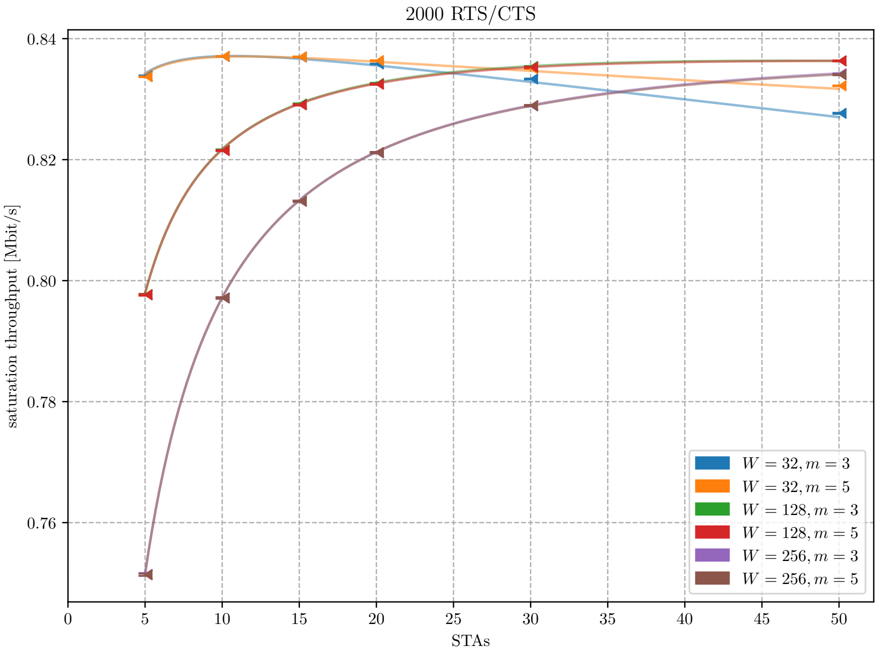
Saturation throughput

8 Results



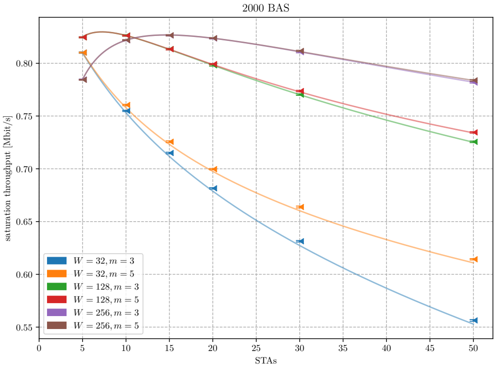
Saturation throughput

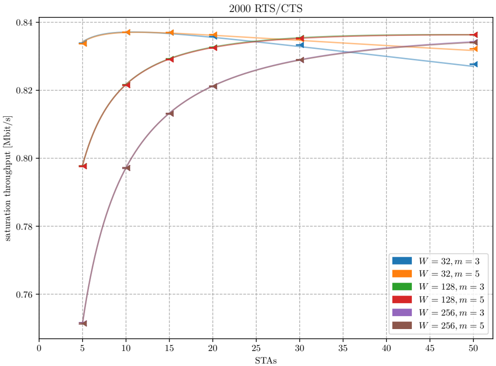
8 Results

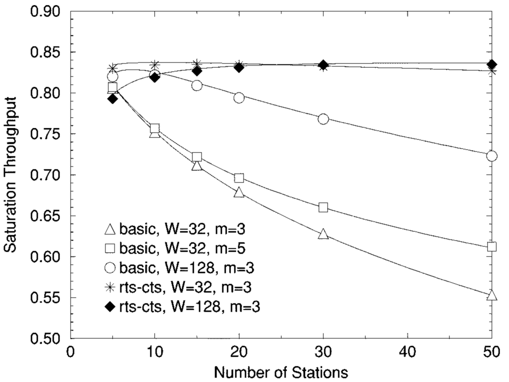


Saturation throughput

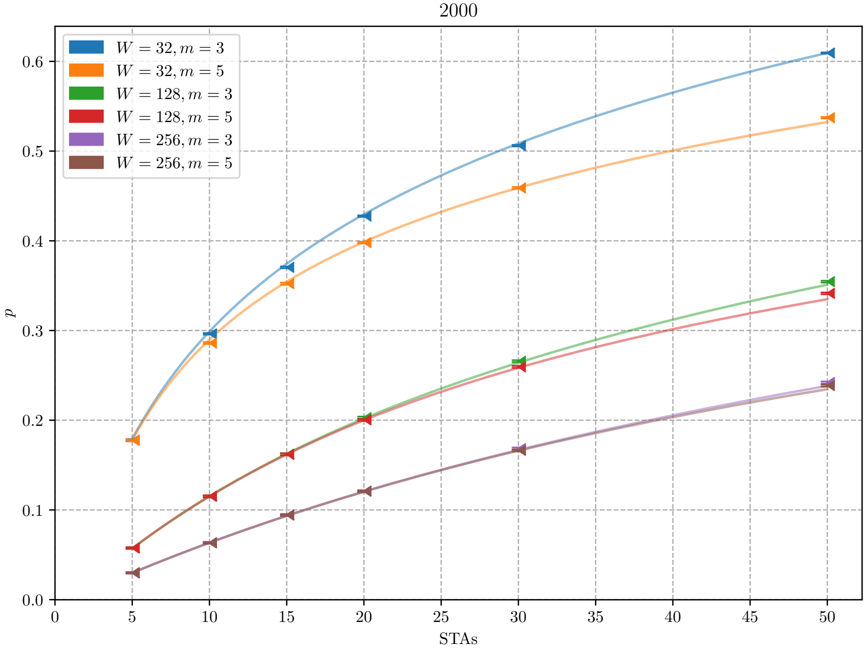
8 Results





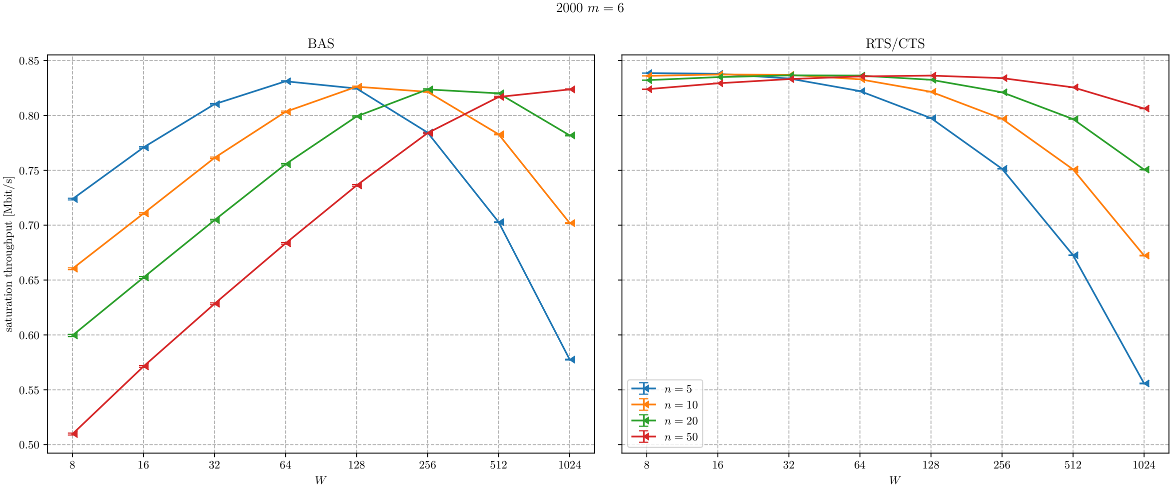


Probability of collision



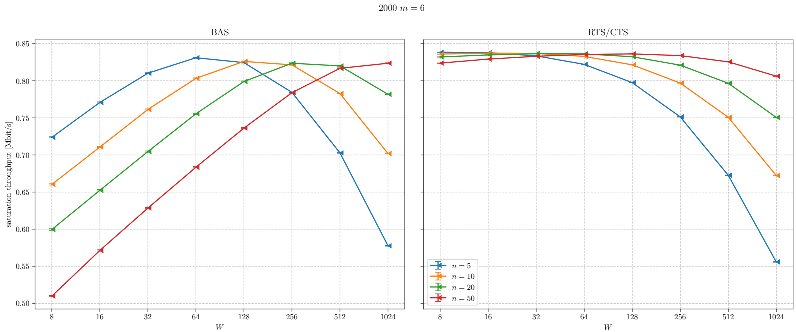
Experiment

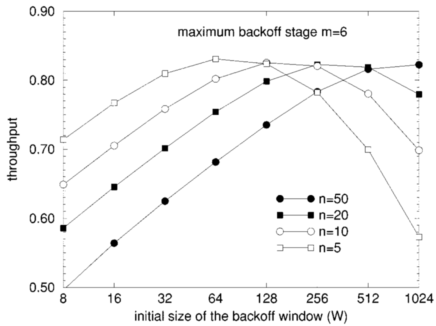
8 Results

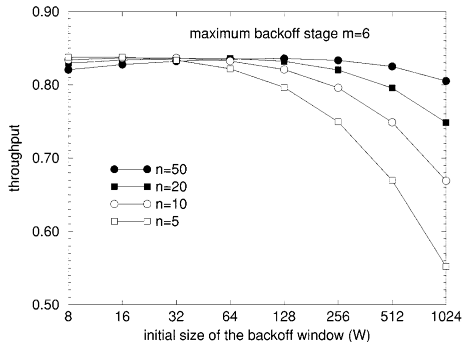


Experiment

8 Results

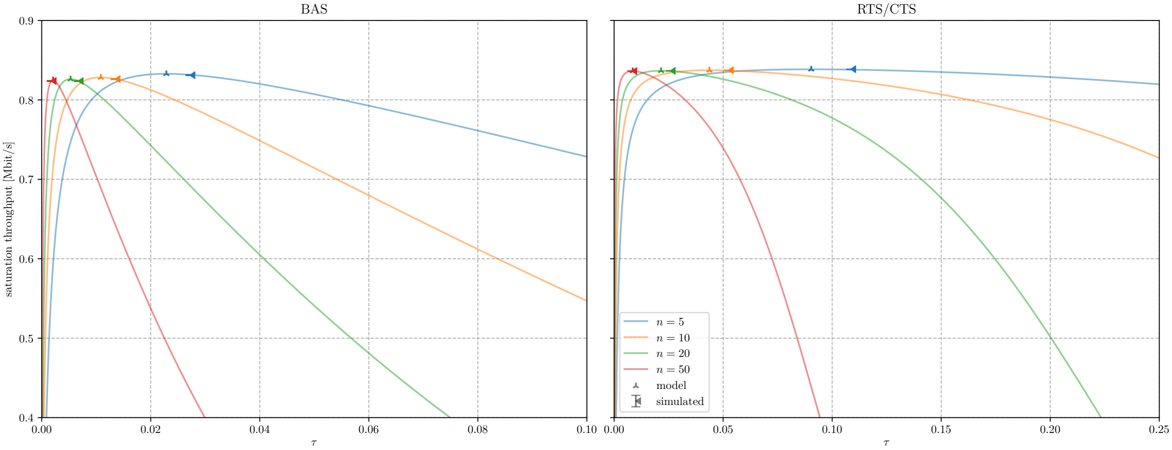






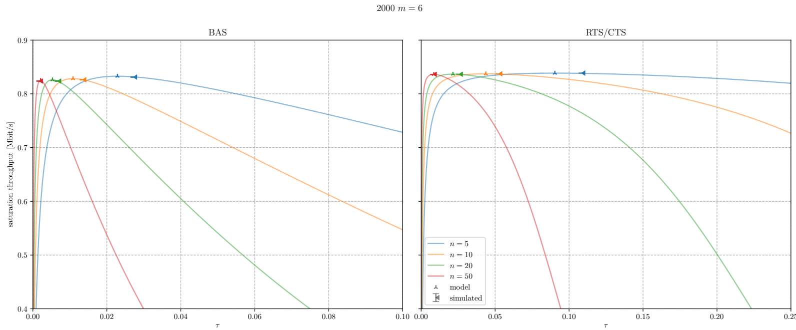
Experiment
8 Results

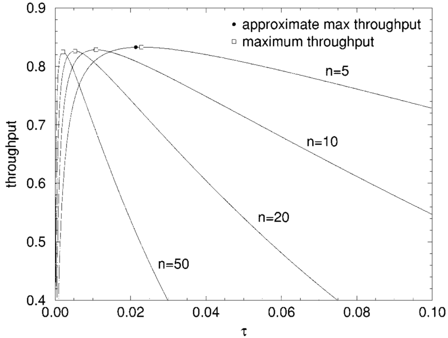
2000 $m = 6$

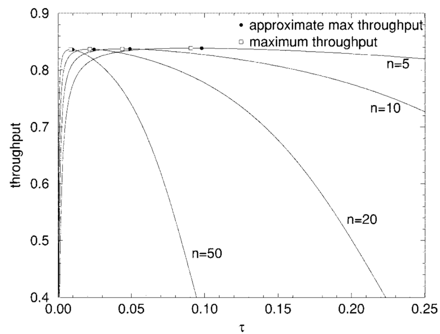


Experiment

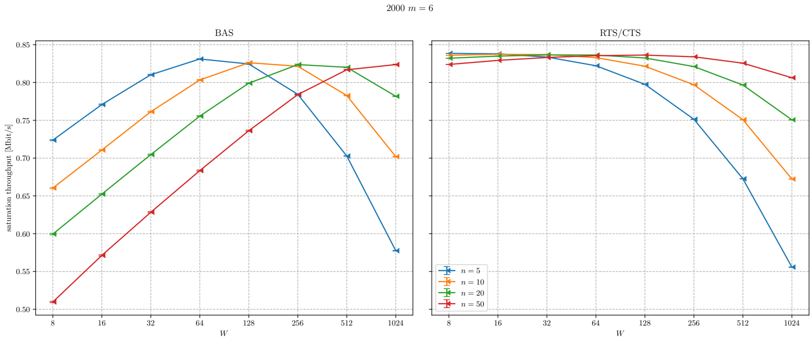
8 Results

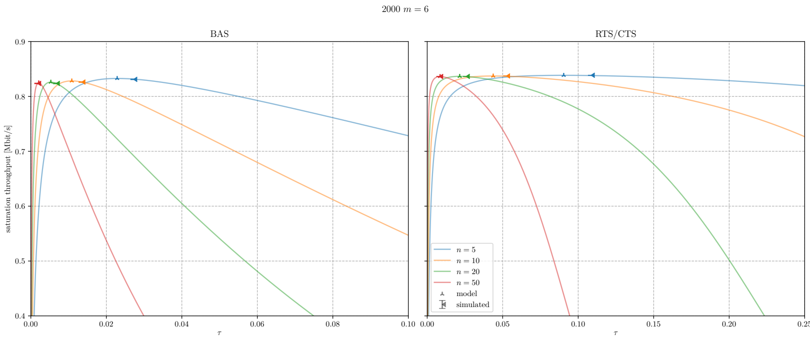






Experiment
8 Results





Q&A