# On Order Gain of Backoff Misbehaving Nodes in CSMA/CA-based Wireless Networks

Zhuo Lu, Wenye Wang, and Cliff Wang

Department of Electrical and Computer Engineering
North Carolina State University

# Outline

# Random Backoff in CSMA/CA

CSMA/CA (carrier sensing multiple access with collision avoidance) relies on distributed random backoff to resolve channel collisions.

Binary exponential backoff is widely used in existing standards, e.g.,

- 802.3 (Ethernet)
- 802.11 (WiFi)
- 802.15.4 (ZigBee)

# Backoff Misbehavior

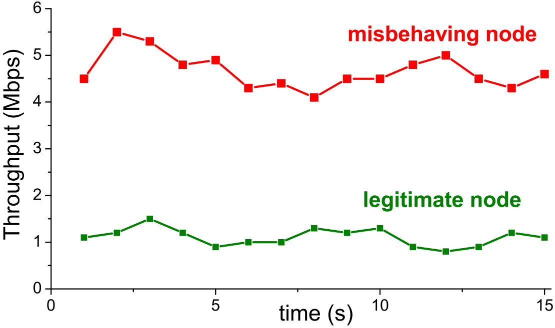
However, a node may not always obey the binary exponential backoff.
Backoff misbehavior: a selfish node purposely reduces its backoff time to gain more access to the channel.

- can lead to severe unfairness, and even denial-of-service.

# Backoff Misbehavior

However, a node may not always obey the binary exponential backoff. Backoff misbehavior: a selfish node purposely reduces its backoff time to gain more access to the channel.

- can lead to severe unfairness, and even denial-of-service.




Snapshot from jperf for comparison between throughputs of misbehaving and legitimate nodes.

# Existing Work in the Literature

Existing Work focused mainly on design of countermeasures to backoff misbehavior.

- Much work assumed a particular backoff misbehavior model and provided countermeasure to it. [Kyasanur'05, Guang'08, Rong'06]

Recent work [Radosavac'08] pointed out that we should focus more on backoff misbehaviors with significant gains.

# Existing Work in the Literature

Existing Work focused mainly on design of countermeasures to backoff misbehavior.

- Much work assumed a particular backoff misbehavior model and provided countermeasure to it. [Kyasanur'05, Guang'08, Rong'06]

Recent work [Radosavac'08] pointed out that we should focus more on backoff misbehaviors with significant gains.

However, little attention has been focused on quantifying the gain of backoff misbehaving nodes in the literature.

We are therefore motivated to address the problem: *how to quantify the gain of backoff misbehavior?*

# Two Classes of Backoff Misbehavior

Continuous misbehavior:

- keeps performing misbehavior, unless disabled by countermeasures.

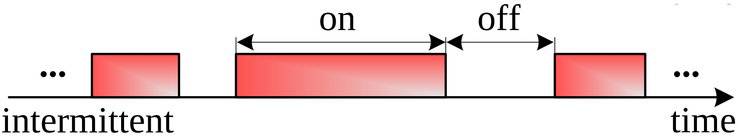
# Two Classes of Backoff Misbehavior

Continuous misbehavior:

- keeps performing misbehavior, unless disabled by countermeasures.

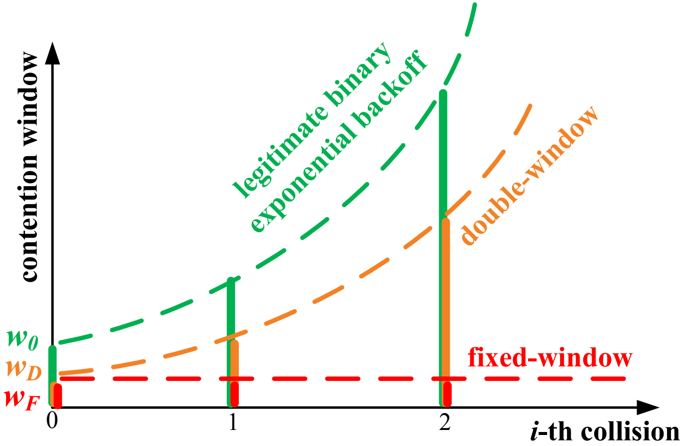

Intermittent misbehavior:

- performs misbehavior only in *on* state.



# Continuous Misbehavior Models

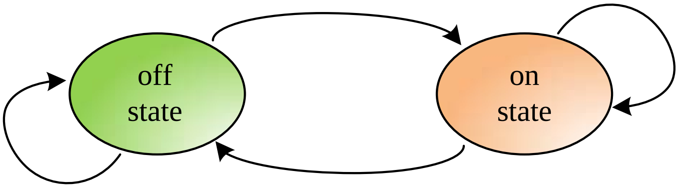
- Double-window backoff misbehaving scheme $\mathcal{B}_D$.
- Fixed-window backoff misbehaving scheme $\mathcal{B}_F$.



# Intermittent Misbehavior Model

A general Markov Chain with two states (on, off).

- on state: use any misbehaving scheme $\mathcal{B}_m$.
- off state: legitimate .
- on-state ratio $\theta \in (0, 100\%)$.



# Choose a performance metric

Our goal is to choose a performance metric to measure the gain of backoff misbehavior for CSMA/CA-based wireless networks.

# Choose a performance metric

Our goal is to choose a performance metric to measure the gain of backoff misbehavior for CSMA/CA-based wireless networks.

Candidates:

- throughput
- delay

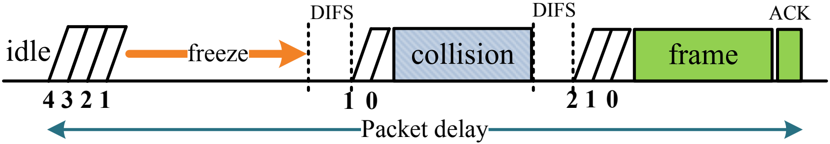
# Choose a performance metric

Our goal is to choose a performance metric to measure the gain of backoff misbehavior for CSMA/CA-based wireless networks.

Candidates:

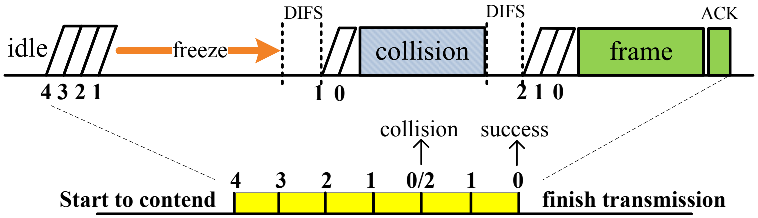
- throughput
- delay

But they are associated highly with protocol signals.



# Waiting Time in a Backoff Process

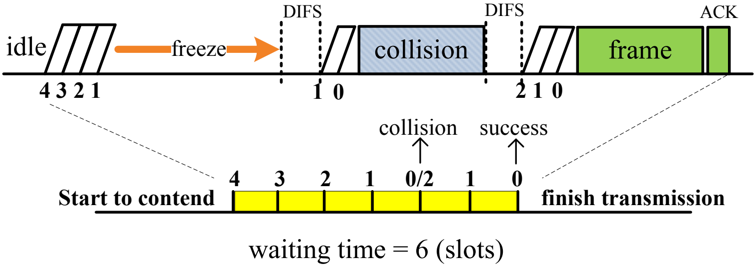
If we delete all protocol signals; then,




waiting time = 6 (slots)

# Waiting Time in a Backoff Process

If we delete all protocol signals; then,



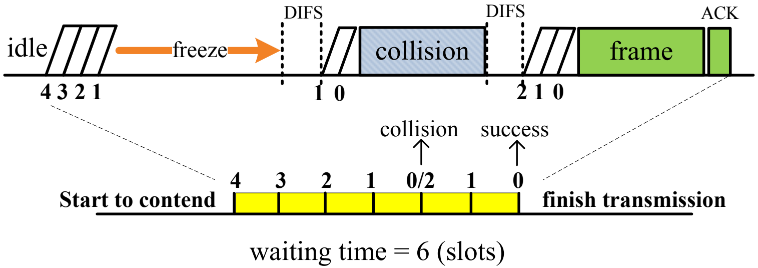

The resultant process consists only idle slots during the period the node contends for the channel.

The waiting time, which is the total number of counted idle slots when a node contends for the channel

- only measured by slots, therefore protocol-independent.

# Waiting Time in a Backoff Process

If we delete all protocol signals; then,




The resultant process consists only idle slots during the period the node contends for the channel.

The waiting time, which is the total number of counted idle slots when a node contends for the channel

- only measured by slots, therefore protocol-independent.

> Waiting time is a good metric to quantify the performance of one node in a generic CSMA/CA network.

# Order Gain Based on Waiting Time

- Waiting time: quantify performance of one node.
- Goal: performance difference between misbehaving node and legitimate node.

We need to introduce a new metric. First, define some rules on our metric $G$.

# Order Gain Based on Waiting Time

- Waiting time: quantify performance of one node.
- Goal: performance difference between misbehaving node and legitimate node.

We need to introduce a new metric. First, define some rules on our metric $G$.

1. Based on the waiting time.
2. If two nodes use the same backoff scheme, the metric should be 0.
3. The metric follows the additive rule:
   - If the gain of node A over node B is $G_1$ and the gain of node B over node C is $G_2$,
   - Then the gain of node A over node C: $G_1 + G_2$.

# Order Gain Based on Waiting Time

- Waiting time: quantify performance of one node.
- Goal: performance difference between misbehaving node and legitimate node.

We need to introduce a new metric. First, define some rules on our metric $G$.

1. Based on the waiting time.
2. If two nodes use the same backoff scheme, the metric should be 0.
3. The metric follows the additive rule:
   - If the gain of node A over node B is $G_1$ and the gain of node B over node C is $G_2$,
   - Then the gain of node A over node C: $G_1 + G_2$.

**Definition of Order Gain: The Gain of Node B over Node A**

$$G(t) = \log_t \frac{P(W_A > t)}{P(W_B > t)}$$

# Order Gain Based on Waiting Time (cont'd)

## Definition of Order Gain

$$G(t) = \log_t \frac{P(W_A > t)}{P(W_B > t)}$$

# Order Gain Based on Waiting Time (cont'd)

## Definition of Order Gain

$$G(t) = \log_t \frac{P(W_A > t)}{P(W_B > t)}$$

- Rule 1: Based on tail distribution $P(W > t)$, e.g.,
  - $P(W > t) = \lambda e^{-\lambda t}$, $P(W \leq t) = 1 - \lambda e^{-\lambda t}$

# Order Gain Based on Waiting Time (cont'd)

## Definition of Order Gain

$$G(t) = \log_t \frac{P(W_A > t)}{P(W_B > t)}$$

- Rule 1: Based on tail distribution $P(W > t)$, e.g.,
  - $P(W > t) = \lambda e^{-\lambda t}$, $P(W \leq t) = 1 - \lambda e^{-\lambda t}$
- Rule 2 ($W_A$ and $W_B$ follow the same distribution) :

$$\log_t \frac{P(W_A > t)}{P(W_B > t)} = 0.$$

# Order Gain Based on Waiting Time (cont'd)

## Definition of Order Gain

$$G(t) = \log_t \frac{P(W_A > t)}{P(W_B > t)}$$

- Rule 1: Based on tail distribution $P(W > t)$, e.g.,
  - $P(W > t) = \lambda e^{-\lambda t}$, $P(W \leq t) = 1 - \lambda e^{-\lambda t}$
- Rule 2 ($W_A$ and $W_B$ follow the same distribution) :

$$\log_t \frac{P(W_A > t)}{P(W_B > t)} = 0.$$

- Rule 3 (Additive rule):

$$\log_t \frac{P(W_A > t)}{P(W_C > t)} = \log_t \frac{P(W_A > t)}{P(W_B > t)} + \log_t \frac{P(W_B > t)}{P(W_C > t)}$$

# Order Gain Based on Waiting Time (cont'd)

## Definition of Order Gain

$$G(t) = \log_t \frac{P(W_A > t)}{P(W_B > t)}$$

- Rule 1: Based on tail distribution $P(W > t)$, e.g.,
  - $P(W > t) = \lambda e^{-\lambda t}$, $P(W \leq t) = 1 - \lambda e^{-\lambda t}$
- Rule 2 ($W_A$ and $W_B$ follow the same distribution) :

$$\log_t \frac{P(W_A > t)}{P(W_B > t)} = 0.$$

- Rule 3 (Additive rule):

$$\log_t \frac{P(W_A > t)}{P(W_C > t)} = \log_t \frac{P(W_A > t)}{P(W_B > t)} + \log_t \frac{P(W_B > t)}{P(W_C > t)}$$

- A valid metric to indicate the gain of backoff misbehavior.

# Order Gain of Continuous Misbehavior

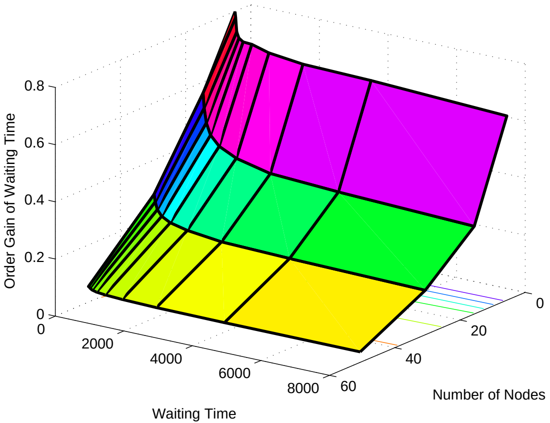
## Analytical Results for Continuous Misbehavior

Double-window: $$G_D(t) = \log_2 \frac{p}{p_{_D}} + \Theta\left(\frac{1}{\ln t}\right),$$

Fixed-window: $$G_F(t) = \Theta\left(\frac{t}{\ln t}\right),$$

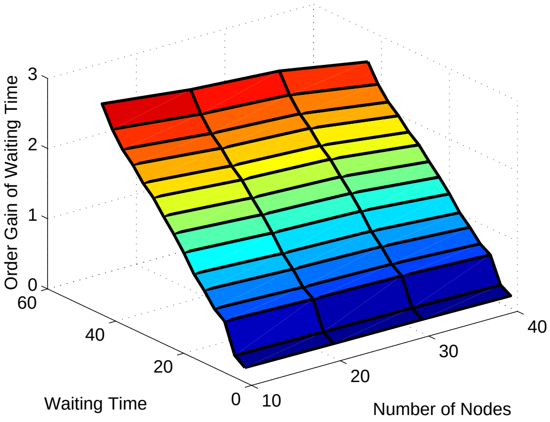
where $p$ and $p_{_D}$ are the collision probabilities of legitimate and double-window misbehaving nodes, respectively.

# Simulation Results for Double-Window Misbehavior




Legitimate nodes have minimum contention window $w_0 = 16$. The double-window misbehaving node has minimum contention window $w_D = 6$.

# Simulation Results for Fixed-Window Misbehavior




Legitimate nodes have minimum contention window $w_0 = 16$. The double-window misbehaving node has minimum contention window $w_D = 8$.

# Experimental Results in a WiFi Network

Experiment Setups:

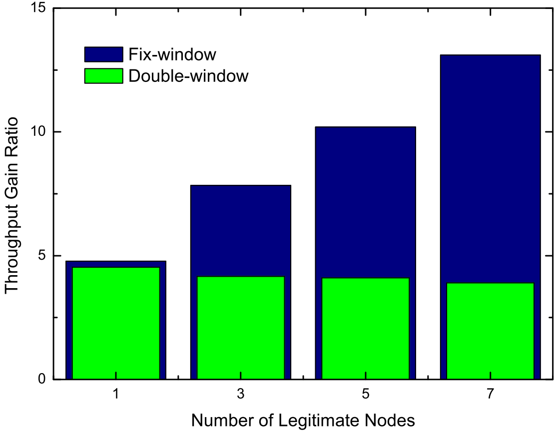
1. Off-the-shelf 802.11b network adapter.
2. Metric:

$$\text{Throughput gain ratio} = \frac{\text{throughput of misbehaving node}}{\text{throughput of legitimate node}}.$$

# Experimental Results in a WiFi Network

One continuous misbehaving node chooses one of the following.

1. Double-window backoff misbehavior with $W_D = 8$.
2. Fix-window backoff misbehavior with $W_F = 8$.



# Order Gain of Intermittent Misbehavior

## Analytical Results for Intermittent Misbehavior

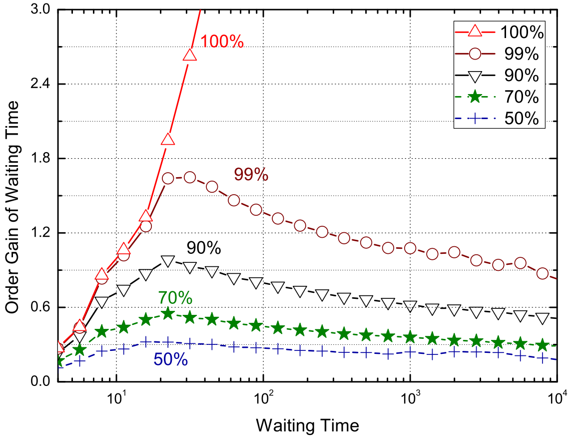
For an intermittent misbehaving node with on-state backoff scheme $\mathcal{B}_m$ and on-state ratio $\theta$, its order gain satisfies

$$G_I(t) = \log_2 \frac{p_{_{on}}}{p_{_{off}}} + \Theta\left(\frac{1}{\ln t}\right),$$

$p_{_{on}}$ and $p_{_{off}}$ are collision probabilities of legitimate nodes in *on* and *off* states, respectively.

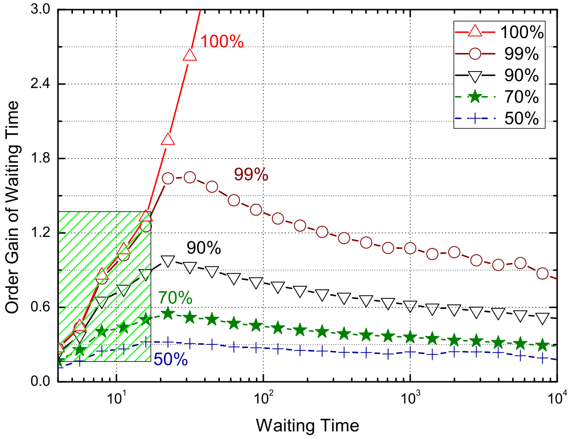
# Simulation Results for Intermittent Misbehavior

On-state misbehavior scheme $\mathcal{B}_m$ is a fixed-window misbehaving scheme with $w_F = 8$.



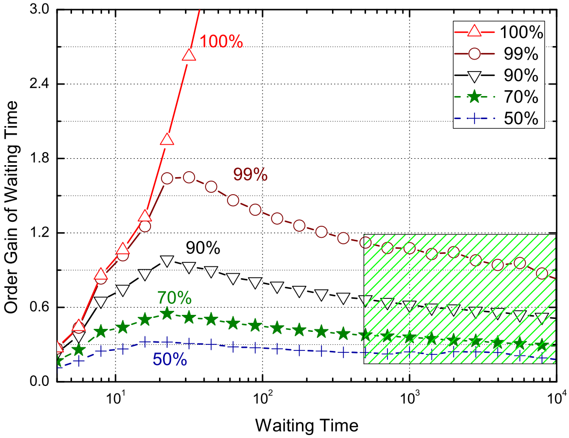
# Simulation Results for Intermittent Misbehavior

On-state misbehavior scheme $\mathcal{B}_m$ is a fixed-window misbehaving scheme with $w_F = 8$.



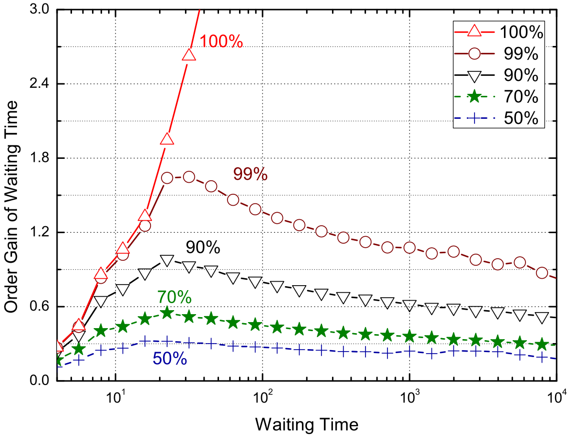
# Simulation Results for Intermittent Misbehavior

On-state misbehavior scheme $\mathcal{B}_m$ is a fixed-window misbehaving scheme with $w_{_F} = 8$.



# Simulation Results for Intermittent Misbehavior

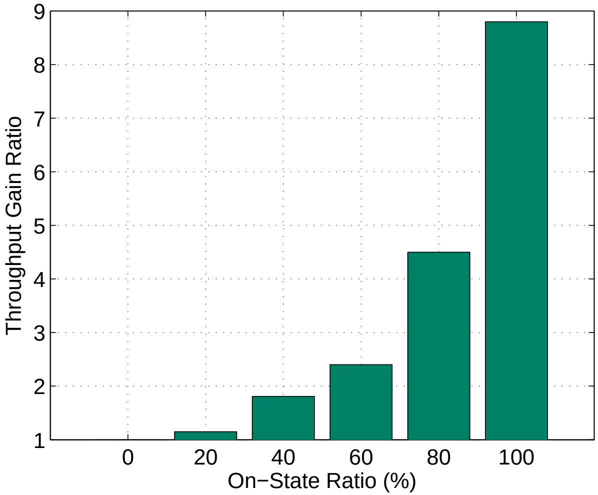
On-state misbehavior scheme $\mathcal{B}_m$ is a fixed-window misbehaving scheme with $w_{_F} = 8$.




Key Observation: For small $\theta$, the order gain is not significant.

# Experimental Results in a WiFi Network

15 legitimate nodes, 1 intermittent misbehaving node.



# Conclusions

- We introduce a metric, order gain to investigate the gain of continuous misbehavior and intermittent misbehavior and perform simulations and experiments to validate our analysis.

# Conclusions

- We introduce a metric, order gain to investigate the gain of continuous misbehavior and intermittent misbehavior and perform simulations and experiments to validate our analysis.
- The number of nodes is a critical factor to the evaluation of countermeasures to backoff misbehaviors.
  - e.g., in a network with a large amount of users, countermeasure can omit double-window misbehavior and focus more on fixed-window misbehavior.

# Conclusions

- We introduce a metric, order gain to investigate the gain of continuous misbehavior and intermittent misbehavior and perform simulations and experiments to validate our analysis.
- The number of nodes is a critical factor to the evaluation of countermeasures to backoff misbehaviors.
  - e.g., in a network with a large amount of users, countermeasure can omit double-window misbehavior and focus more on fixed-window misbehavior.
- An intermittent misbehaving node with a small $\theta$ can not achieve much gain from the network.

Thank you!

1. Off-the-shelf 802.11b network adapter.
2. All devices are placed in the same lab so that they have the same channel condition.
3. Madwifi driver to modify the contention window.
    - Legitimate node: minimum contention window = 32, binary exponential backoff.
    - Misbehaving node: minimum contention window smaller than 32, binary exponential or fixed-window backoff.
4. Iperf to generate saturated traffic.