BÀI#9 - HÀNH VI SAI TRÁI CỦA MAC;

TS. HOÀNG SỸ TƯƠNG

BÀI#9 - HÀNH VI SAI TRÁI CỦA MAC

- Lớp MAC IEEE 802.11
- Hoạt động sai trong 802.11 MAC
- Một vài mối đe dọa MAC khác (thời gian cho phép)

IEEE802.11

- Chế độ cơ sở hạ tầng
  - Nhiều trạm dùng chung một AP kết nối Internet
    - Chức năng điều phối phân tán (DCF)
    - Chức năng điều khiển điểm (PCF)
      - *Hiếm khi được sử dụng do không hiệu quả, đặc tả tiêu chuẩn mơ hồ và thiếu hỗ trợ khả năng tương tác*
- Chế độ Ad hoc
  - Multi-hop, không hạ tầng, không Internet
  - Chưa bao giờ thực sự được ưa chuộng về mặt thương mại
- Chế độ lưới - mesh mode (sử dụng 802.11s)
- • Wi-Fi Direct

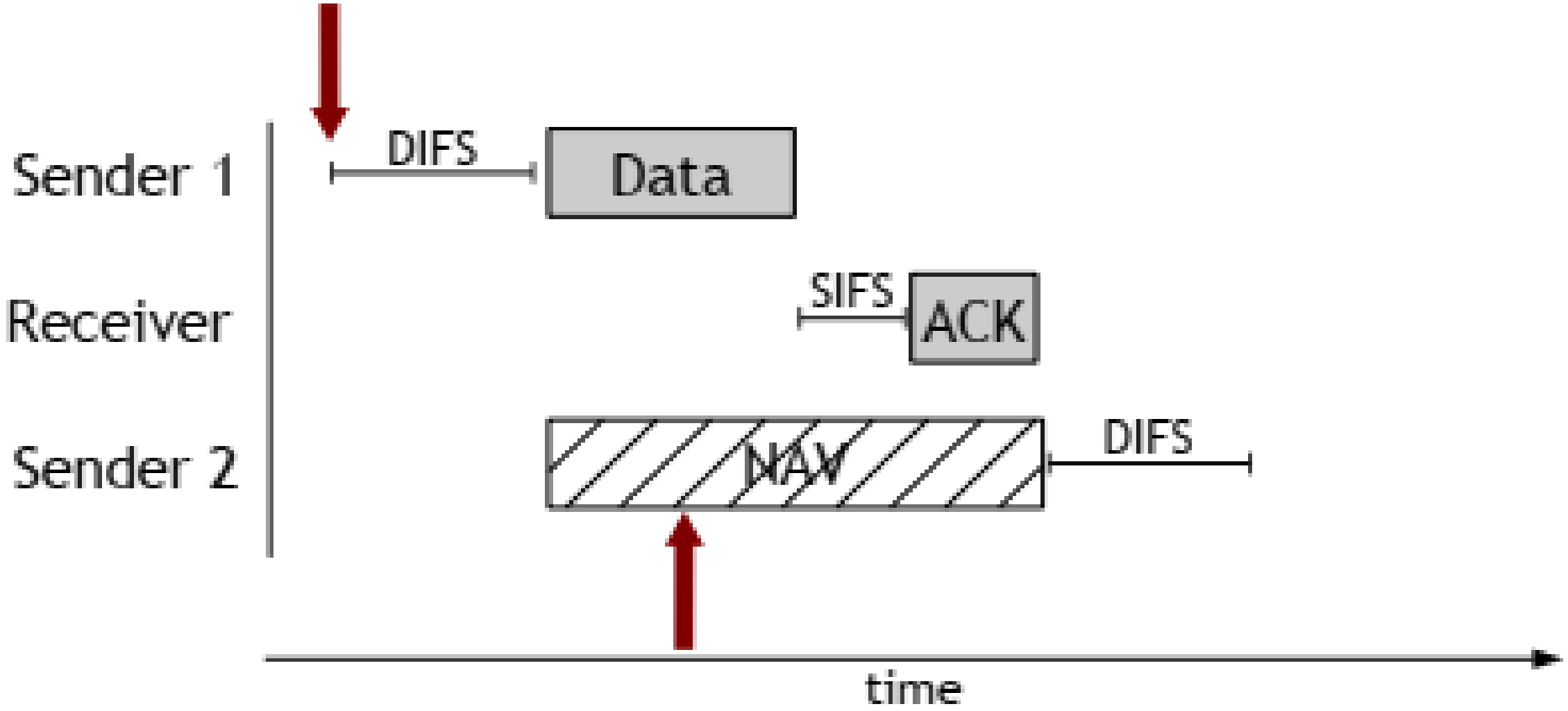
MAC 802.11

- Trách nhiệm của lớp MAC
  - Trách nhiệm logic
    - *xác định Địa chỉ*
    - *Phân mảnh*
    - *Phát hiện, sửa lỗi và quản lý*
  - Trách nhiệm về thời gian
    - *Quản lí kênh*
    - *Kiểm soát luồng liên kết*
    - *Tránh va chạm*
- Hôm nay, chúng ta tập trung vào các lỗ hổng dựa trên thời gian

CSMA - CARRIER SENSE MULTIPLE ACCESS

- Mang ý nghĩa đa truy cập
  - Nghe kênh trước khi truyền
  - Nếu kênh im lặng, hãy truyền
    - *Sau một thời gian trễ ngắn (DIFS = Khoảng cách giữa các khung DCF)*
  - Nếu kênh đang bận:
    - *Chờ cho đến khi nó yên lặng trong một khoảng thời gian DIFS*
    - *Chờ khoảng thời gian dự phòng ngẫu nhiên*
    - *Gửi nếu vẫn im lặng*
  - Chờ ACK hoặc truyền lại bằng cách sử dụng backoff ngẫu nhiên

VẬN HÀNH DCF SỬ DỤNG CSMA



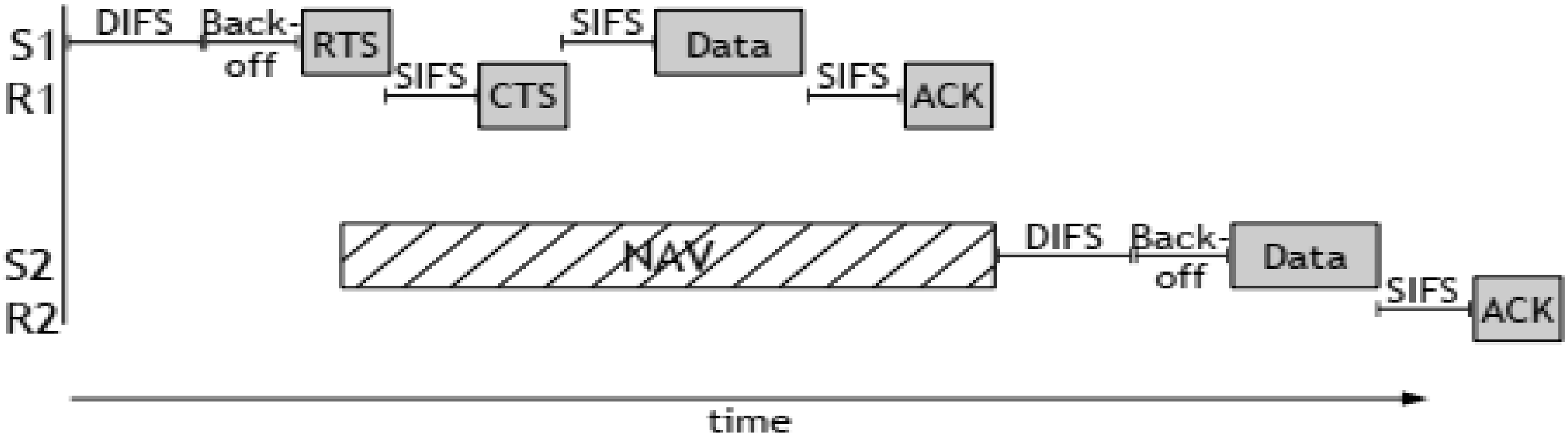
BACKOFF NGẪU NHIÊN

- Giảm khả năng xảy ra va chạm
  - Mỗi thiết bị phải đợi trong một khoảng thời gian ngẫu nhiên tùy thuộc vào sự tranh chấp trong quá khứ - sử dụng "cửa sổ tranh chấp" CW
  - Nếu phương tiện đang bận:
    - *Chờ khoảng thời gian DIFS*
    - *Đặt bộ đếm lùi ngẫu nhiên trong CW*
    - *Truyền sau khi hết thời gian truy cập*
- Sau khi truyền lại không thành công:
  - Tăng CW theo cấp số nhân
    - 2^n-1 from CW_{min} to CW_{max}, e.g., 7 → 15 → 31

TRÁNH VA CHẠM

- Cố gắng đặt trước kênh để tránh xung đột bởi những người gửi khác
  - Yêu cầu gửi (RTS)
    - *Trước khi truyền dữ liệu, bên gửi truyền RTS*
  - Xóa để gửi (CTS)
    - *Bên nhận truyền CTS để thông báo cho bên gửi tiếp tục*
  - RTS và CTS sử dụng IFS ngắn (SIFS < DIFS) để ưu tiên gói dữ liệu

CÁCH SỬ DỤNG RTS/CTS



- RTS/CTS is not required
  - S1-R1 use RTS/CTS, S2-R2 do not

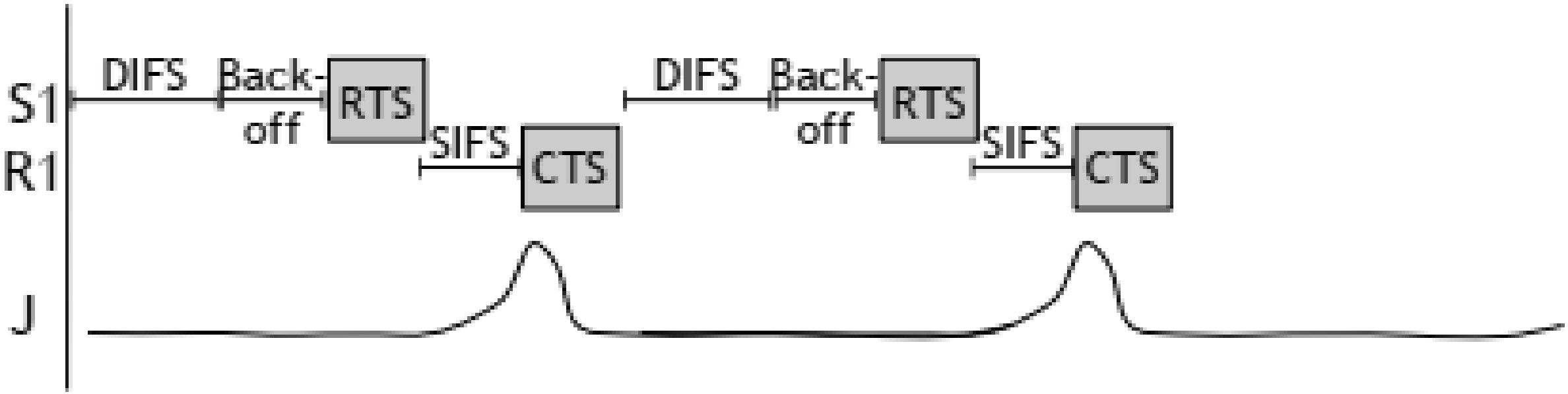
HOẠT ĐỘNG SAI TẠI TẦNG MAC

- 802.11 DCF hoạt động tốt với giả định rằng mọi người tin cậy nhau
  - *Đây có thể là một giả định hợp lý khi các giao thức MAC bị ràng buộc bởi phần cứng*
- Tuy nhiên, các nút ích kỷ và độc hại được tự do phá vỡ các quy tắc
  - *Phần mềm MAC thực hiện việc này rất dễ dàng*

MỘT SỐ CÁCH HOẠT ĐỘNG SAI KHÁC Ở LỚP MAC LÀ GÌ?

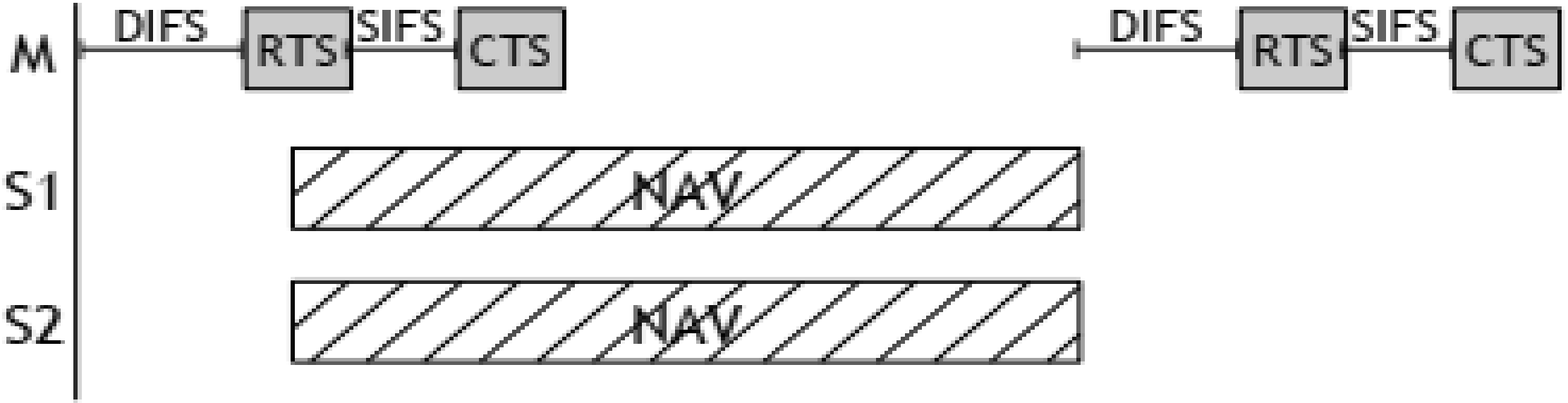
GÂY NHIỄU MAC

- Cấu trúc và hành vi DCF tạo lợi thế cho những kẻ tấn công gây nhiễu
  - *Gây nhiễu sau RTS (và giai đoạn SIFS) chặn CTS (ngăn luồng dữ liệu) và chiếm kênh (ngăn người gửi khác sử dụng kênh)*



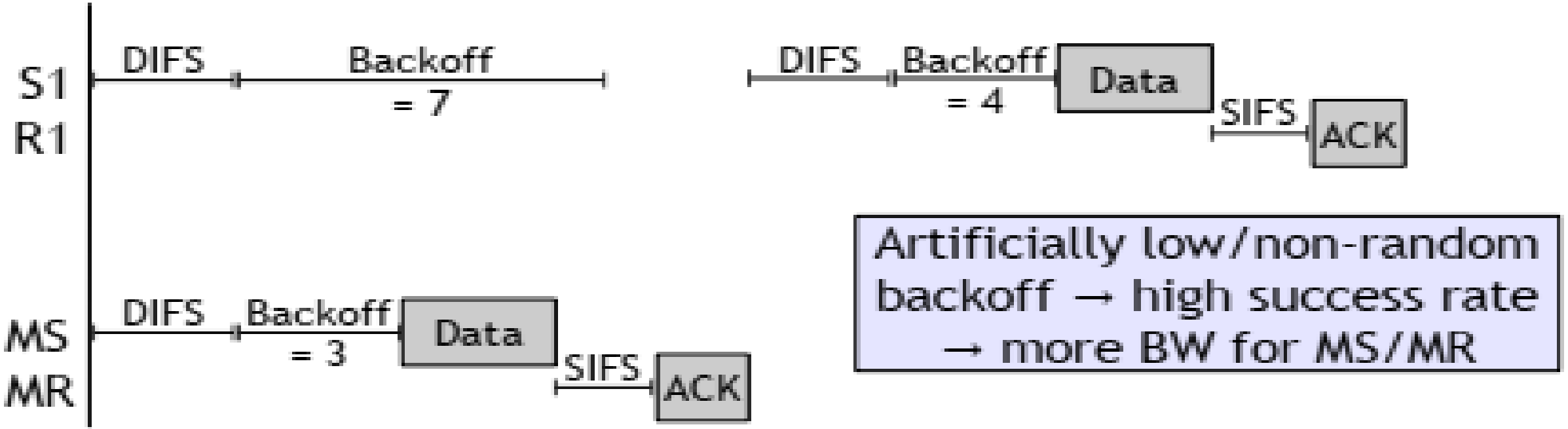
CHẶN MAC

- Cấu trúc và hành vi DCF tạo lợi thế cho những kẻ tấn công DoS khác
  - *RTS/CTS "flooding" - gửi lặp đi lặp lại các trao đổi RTS/CTS trong khi những người gửi khác tuân thủ các quy tắc*



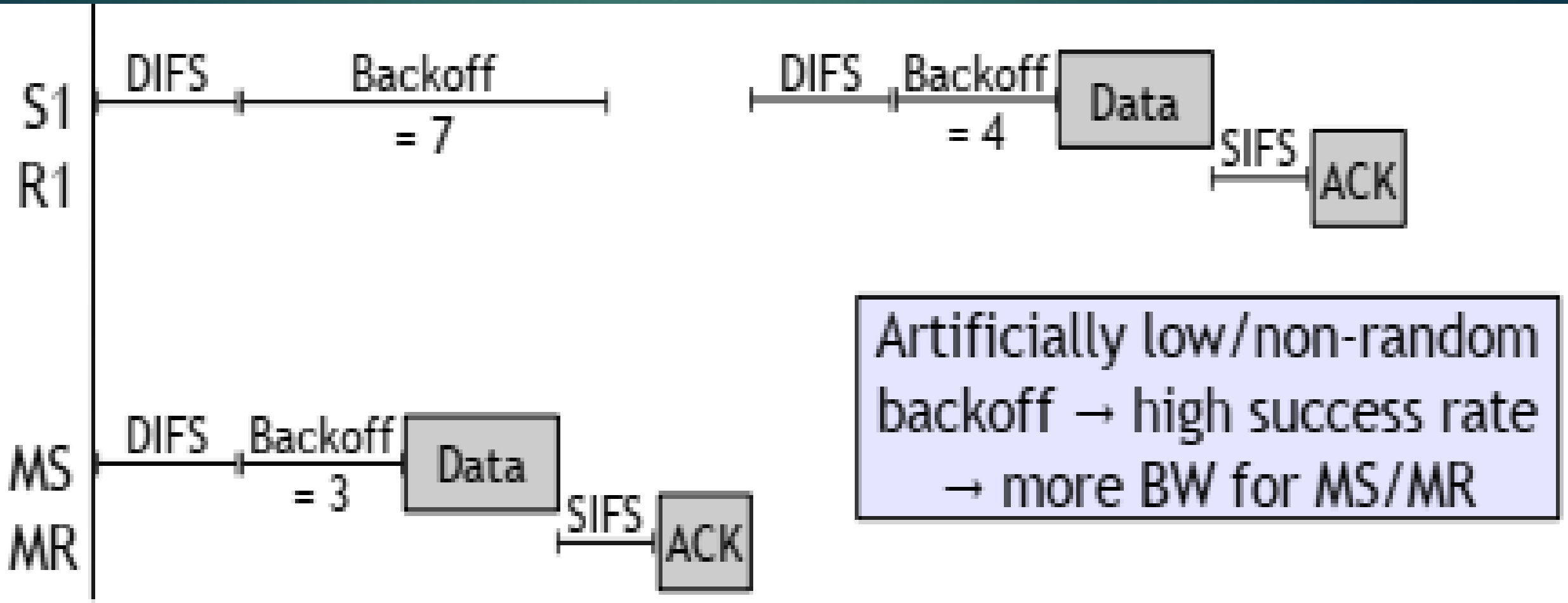
MAC GREED W/ JAMMING

- Các nguồn tham lam/độc hại có thể chặn hoặc xung đột với các nguồn khác, khiến tốc độ gửi của chúng giảm xuống
  - *Tạo thêm cơ hội cho nguồn tham lam*



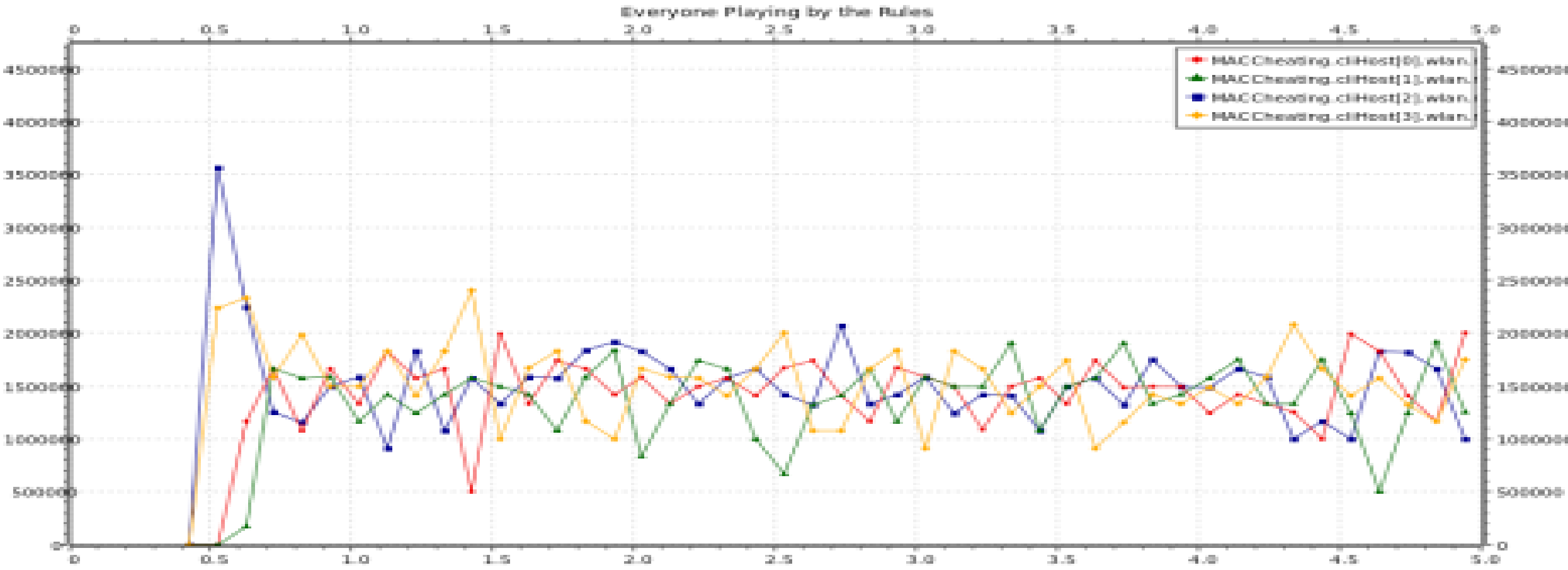
MAC GREED w/ THAM SỐ

- Các nguồn tham lam/độc hại có thể thao túng các tham số giao thức để sử dụng tài nguyên không công bằng



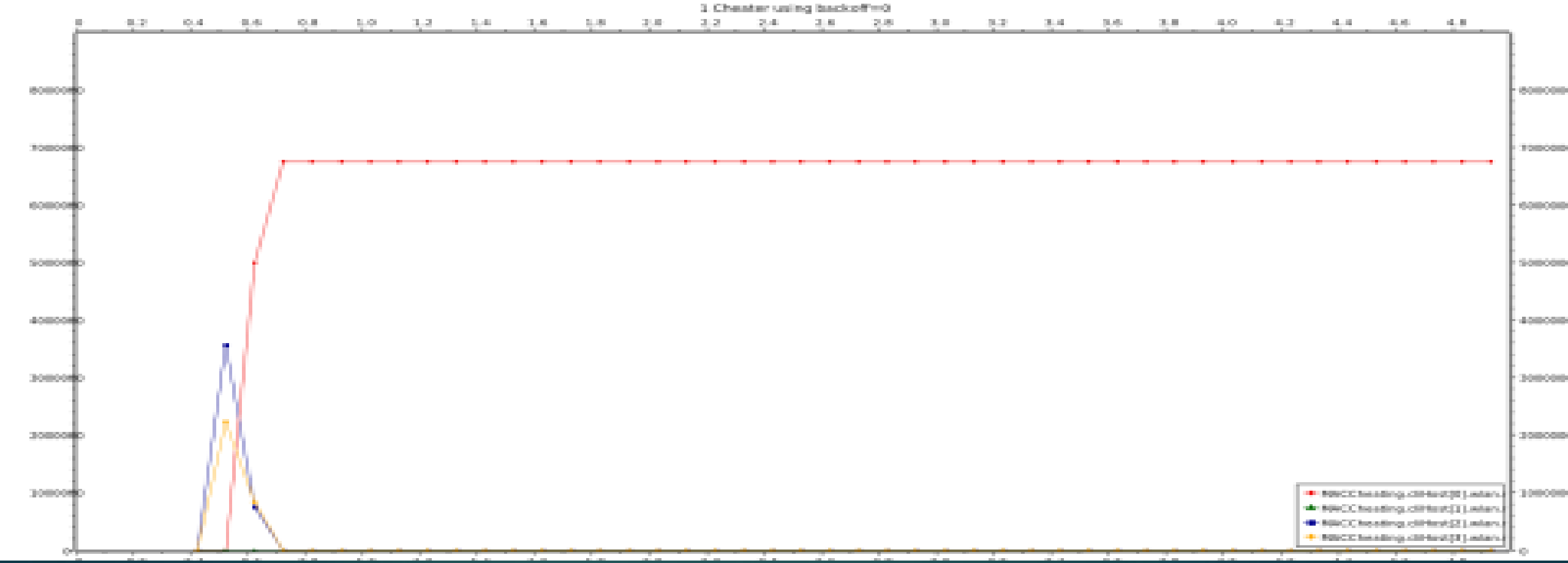
VÍ DỤ

- 4 máy khách, tất cả đều hợp tác (sử dụng OMNET++)



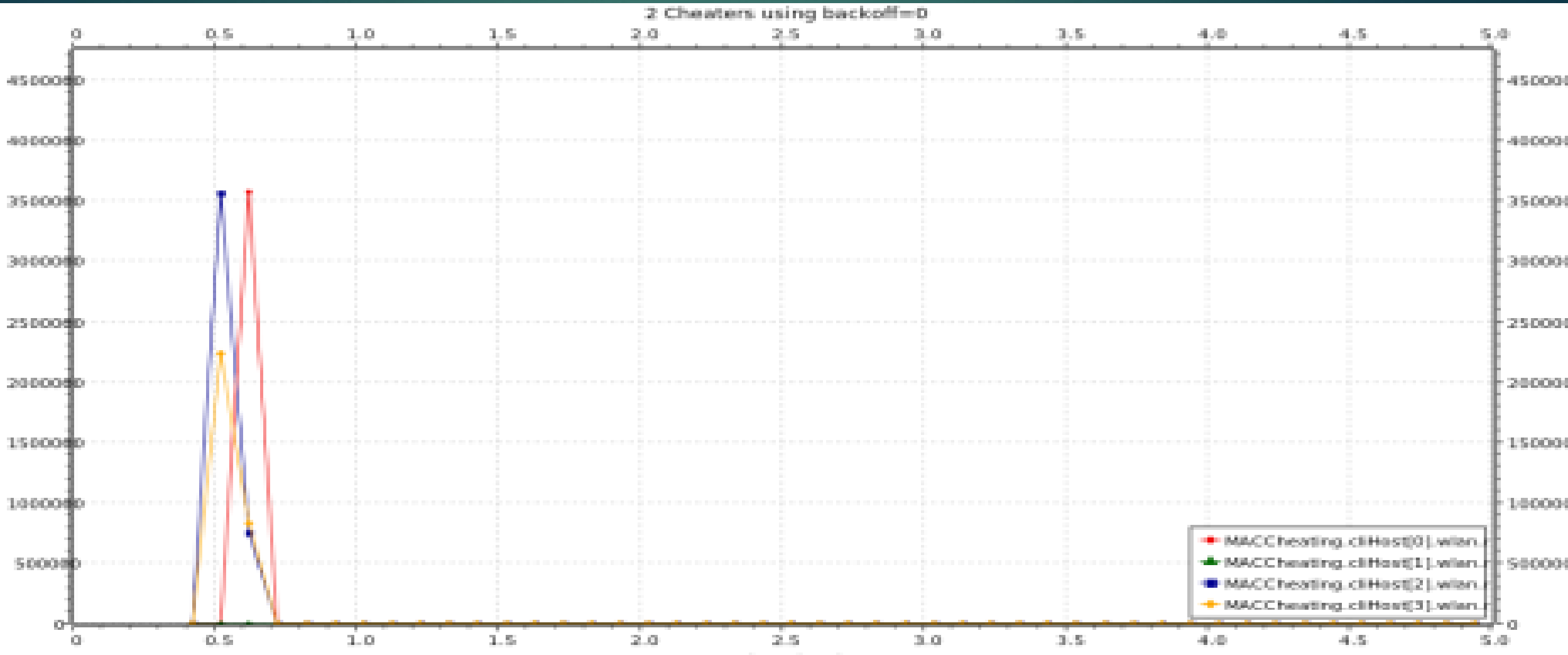
VÍ DỤ

- 4 khách hàng, 1 sử dụng backoff = 0



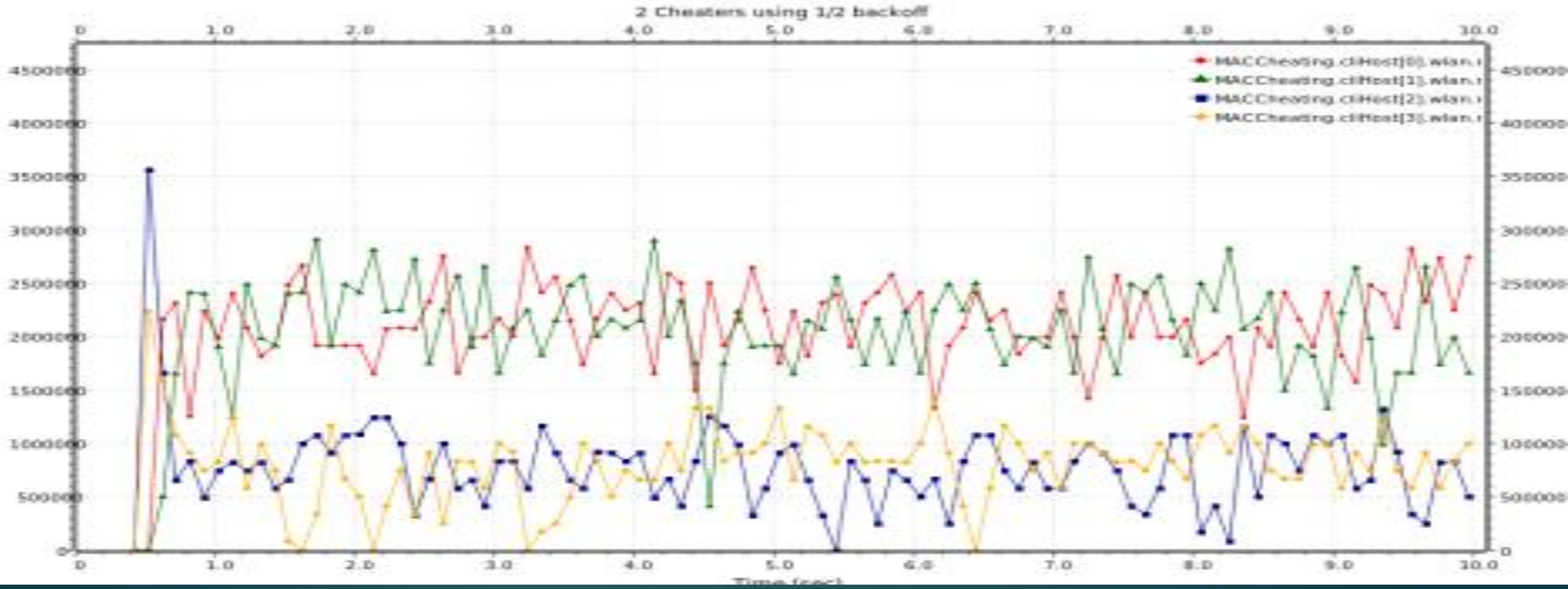
VÍ DỤ

- 4 khách hàng, 2 sử dụng backoff = 0



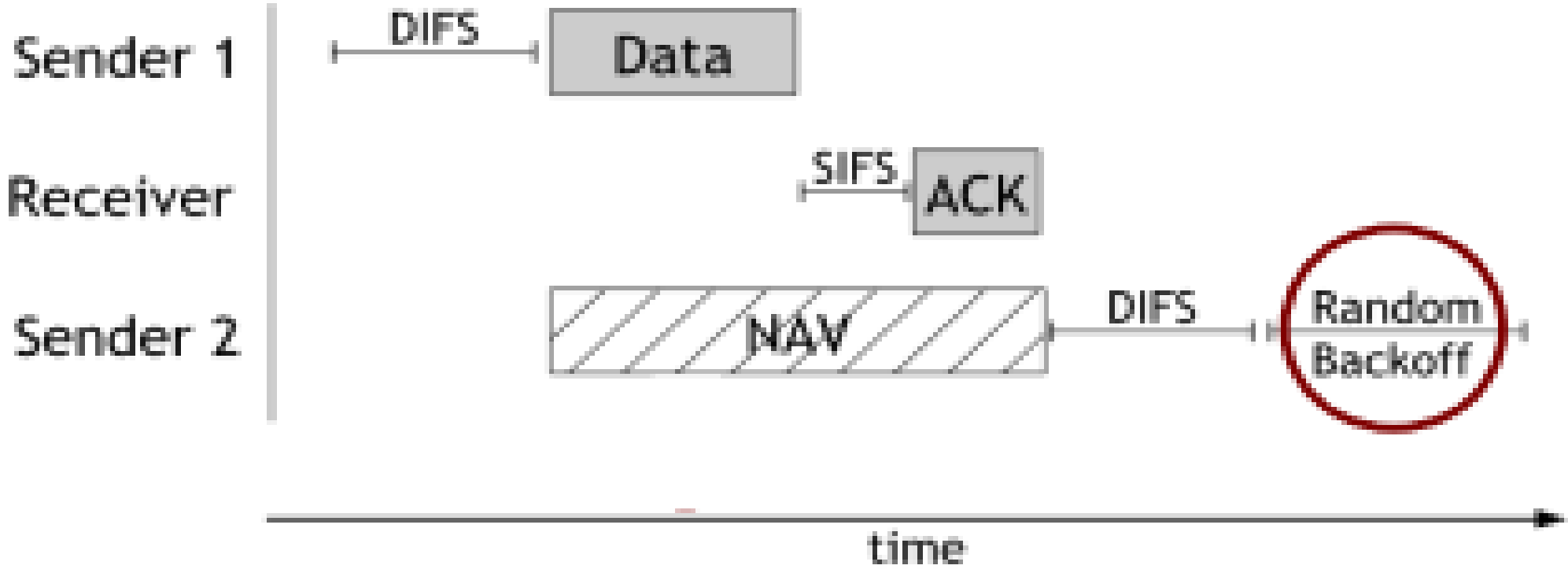
VÍ DỤ

- 4 khách hàng, 1 sử dụng backoff / 2



GIAN LẬN TRONG CSMA/CA

- "CSMA/CA được thiết kế với giả định rằng các nút sẽ chơi theo luật"
  - *Những kẻ gian lận MAC cố tình không tuân theo giao thức IEEE 802.11, đặc biệt là về kích thước cửa sổ tranh chấp và lùi lại*

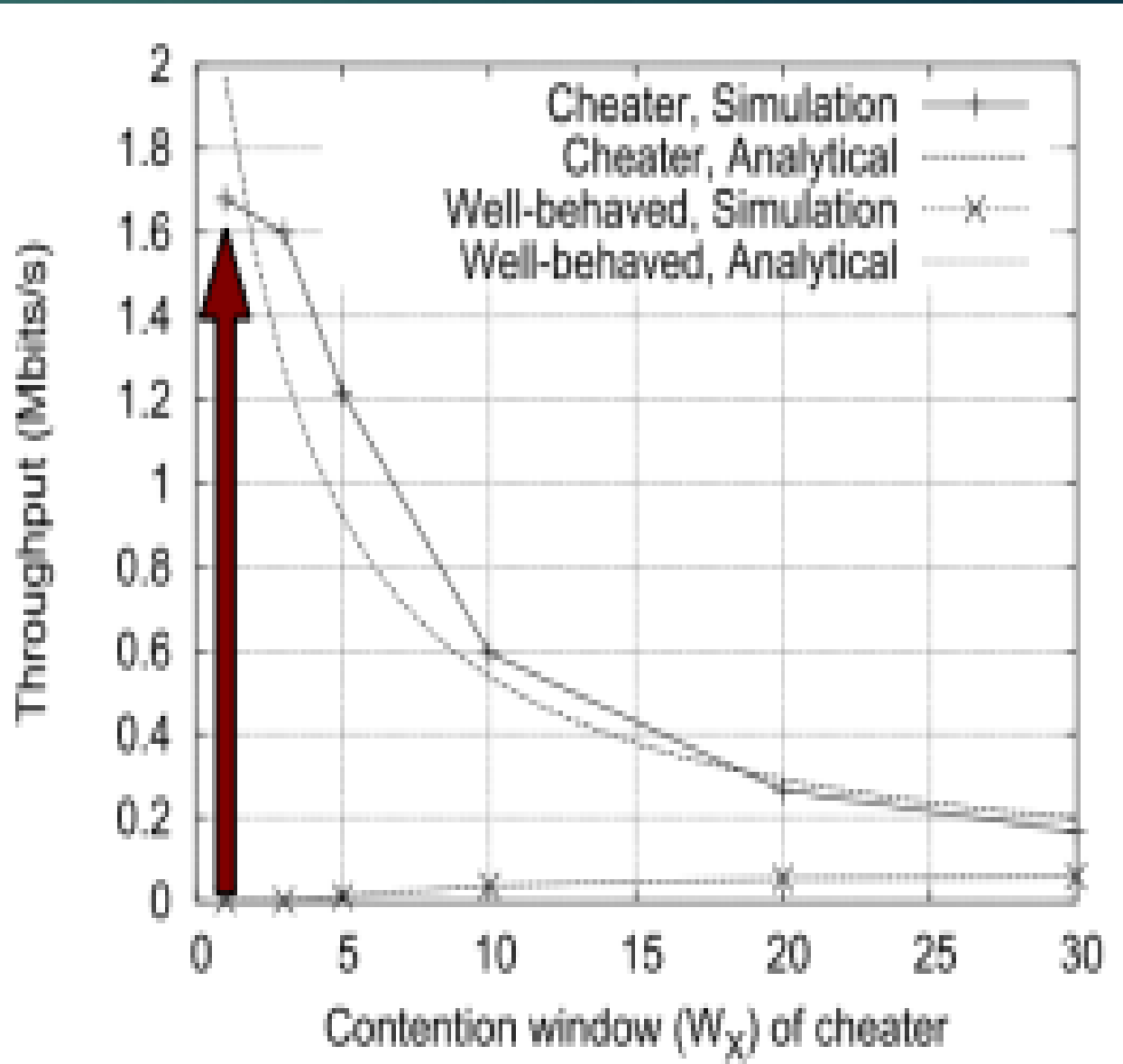


MÔ HÌNH TRÒ CHƠI HỆ THỐNG

- N cặp tx-rx trong một miền xung đột duy nhất, sử dụng 802.11, C của N là những kẻ gian lận có quyền kiểm soát các tham số lớp MAC
- Người gian lận muốn tối đa hóa giá trị trung bình thông lượng r_j
- Là một trò chơi:
  - *Mỗi người chơi (người ăn gian) điều chỉnh kích thước cửa sổ tranh chấp của mình W_j để tối đa hóa tiện ích $U_i = r_i$*
  - *Người chơi phản ứng với những thay đổi của người dùng N-C còn lại, những người chơi theo luật*
- Các tác giả phân tích mối quan hệ giữa thông lượng và kích thước cửa sổ tranh chấp

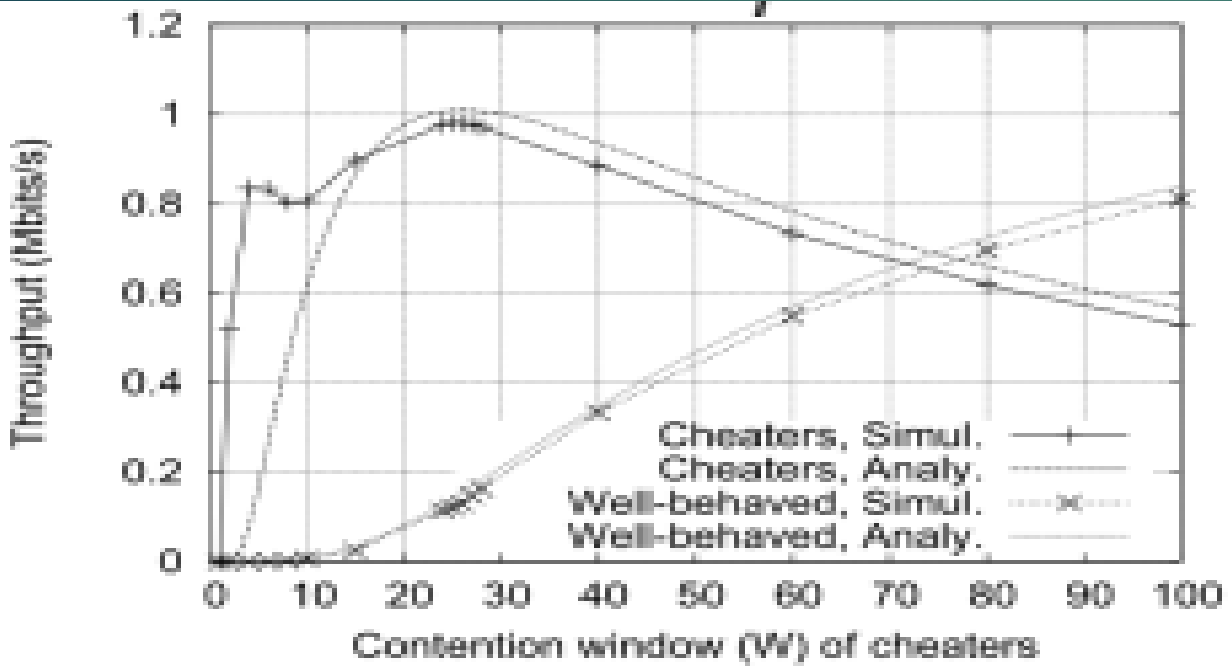
GIAN LẬN TĨNH ĐƠN

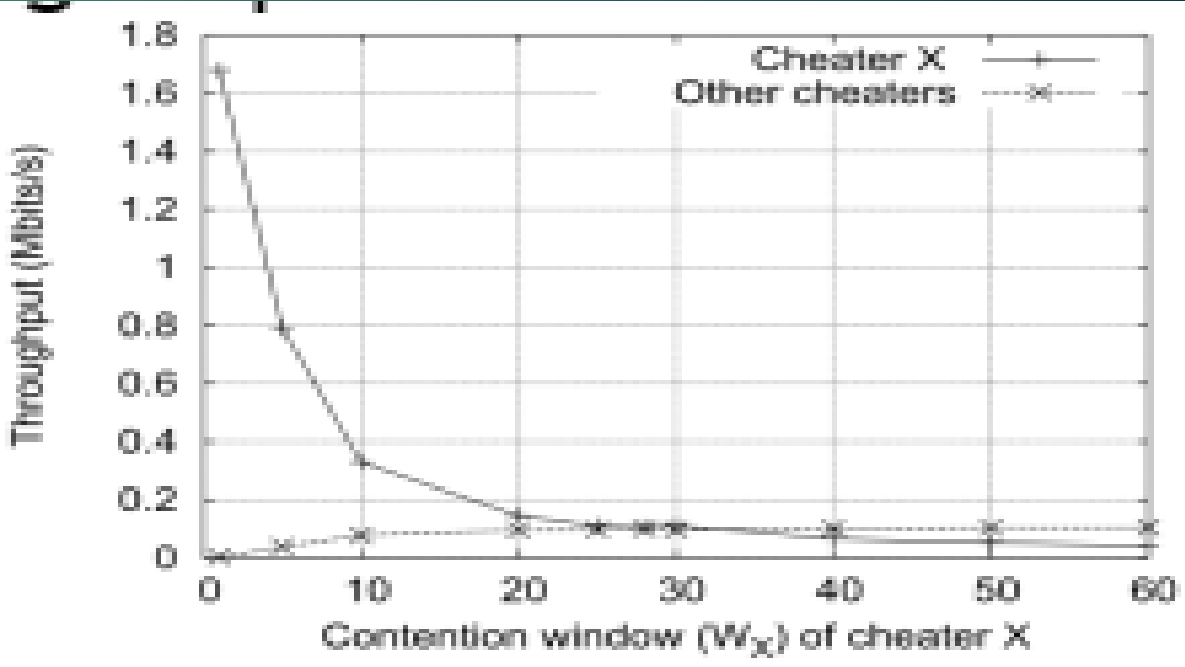
- **Trường hợp đầu tiên:** một kẻ gian lận với một chiến lược cố định (tức là đưa ra quyết định và kiên định với nó)
- Một kẻ gian lận có được thông lượng tốt nhất tại W_j =1
- Trên thực tế, W_j=1 là Cân bằng Nash cho trò chơi tĩnh với C=1



NHIỀU KẺ GIAN LẬN TĨNH

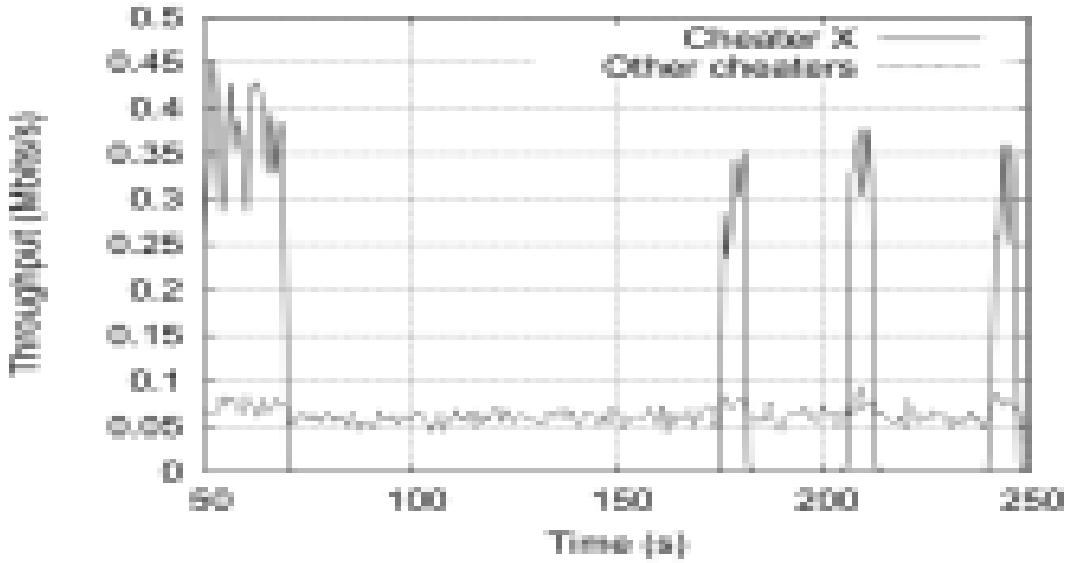
- **Trường hợp thứ hai:** nhiều kẻ gian lận với một chiến lược cố định
  - *2.1 Những kẻ gian lận không biết về nhau*
  - *2.2 Những kẻ gian lận nhận thức được sự cạnh tranh giữa kẻ gian lận và kẻ gian lận trong việc hình thành chiến lược*
- Kích thước cửa sổ W_j=1 không còn tối ưu

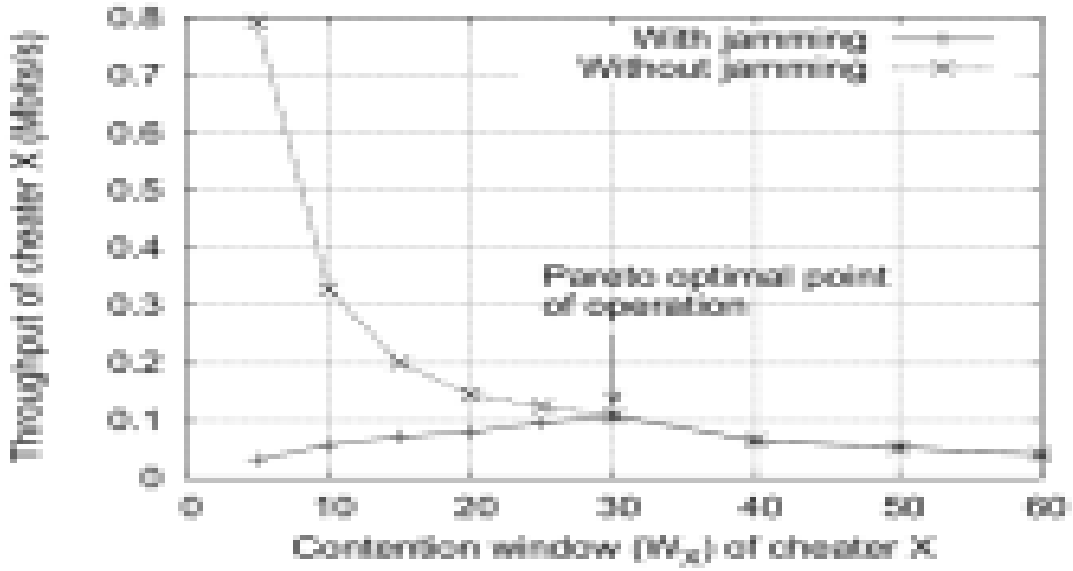




TRÒ CHƠI GIAN LẬN ĐỘNG

- Trong trò chơi động, những người gian lận có thể thay đổi chiến lược của họ để đáp lại những người chơi khác (bao gồm cả những người gian lận khác)
  - *Một hình phạt được thực thi trên chức năng tiện ích, vì vậy những kẻ gian lận hội tụ đến điểm vận hành tối ưu*
  - *"Những kẻ gian lận hợp tác" có thể áp dụng hình phạt đối với "những kẻ gian lận không hợp tác" bằng cách làm nhiễu các gói tin của chúng*

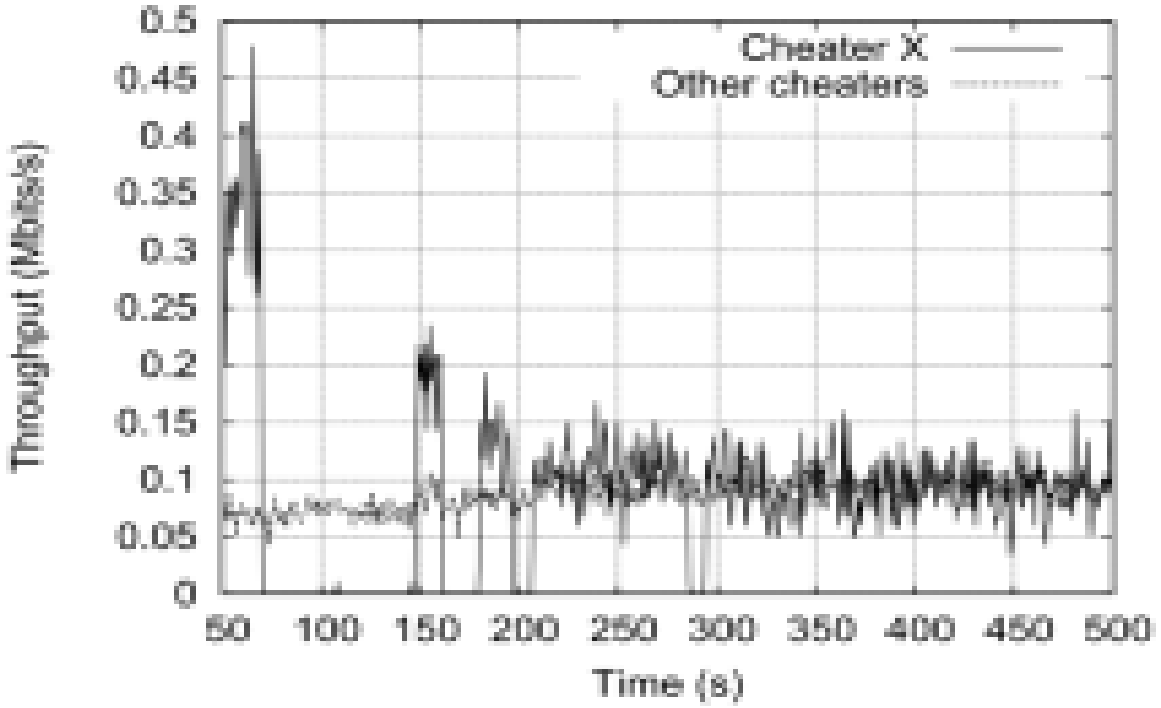


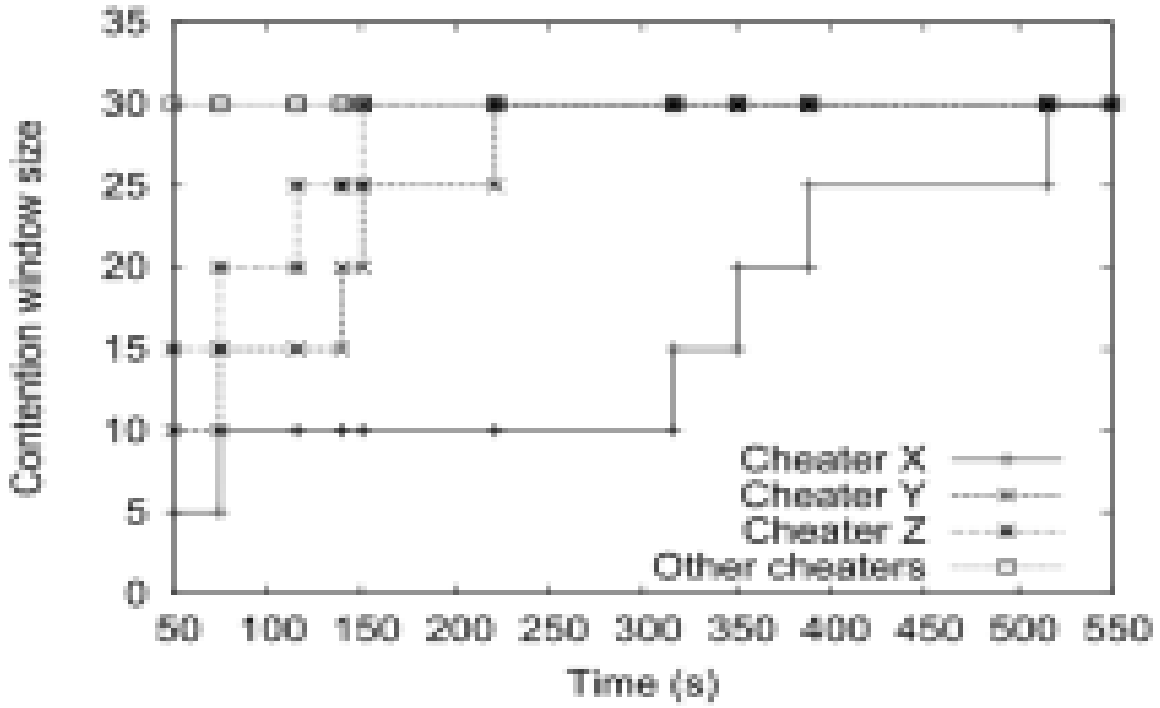


GIAN LẬN PHÂN TÁN/THÍCH ỨNG

- Những kẻ gian lận có thể quan sát thông lượng thực tế và gây nhiễu để điều chỉnh kích thước cửa sổ tranh chấp

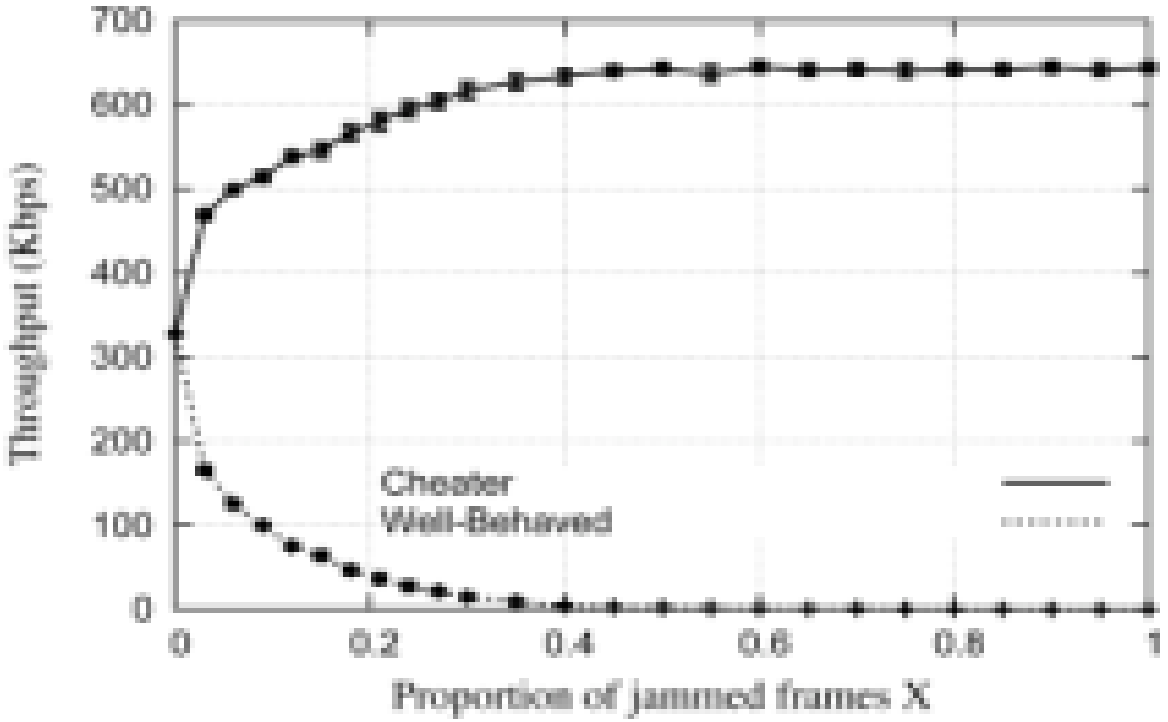
- Những kẻ gian lận buộc phải hợp tác hoặc nhận được thông lượng thấp hơn do bị phạt từ những kẻ gian lận khác

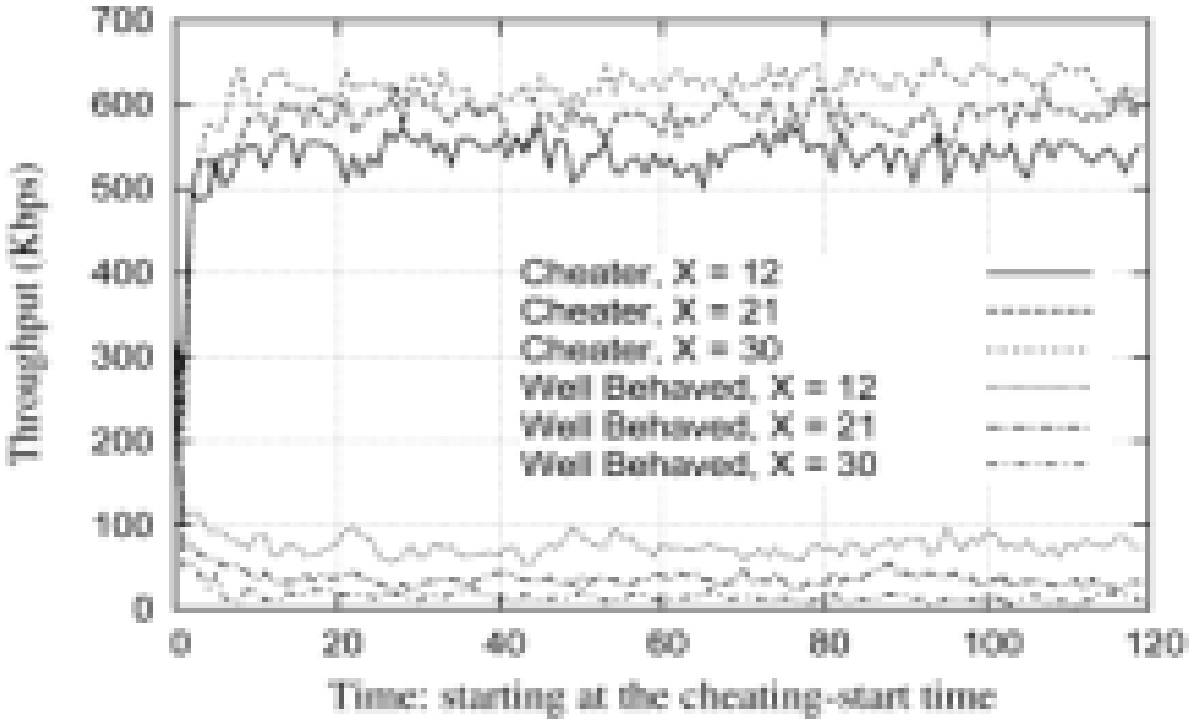




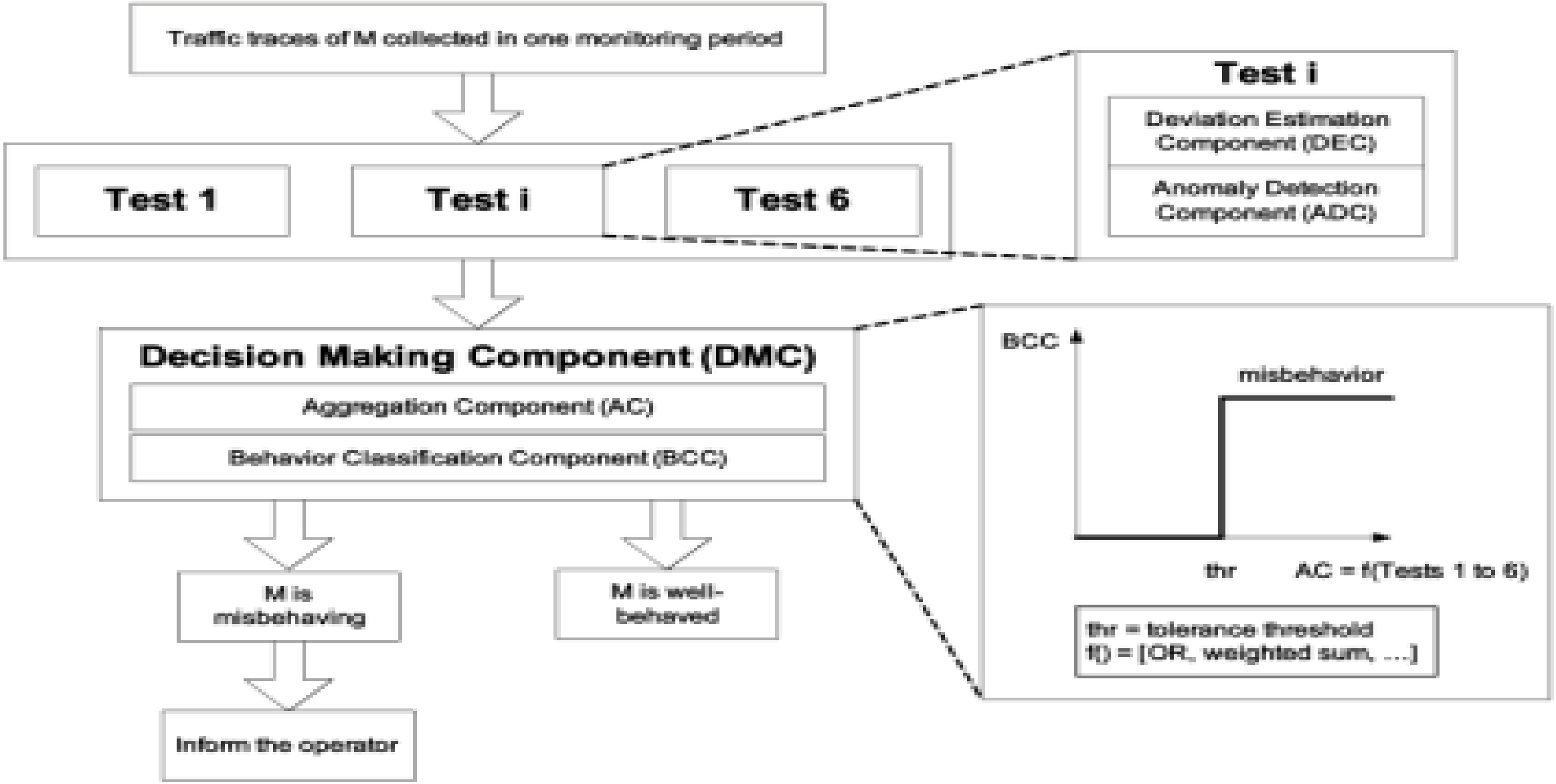
PHÁT HIỆN HÀNH VI THAM LAM

- Phát hiện hành vi tham lam trong lớp Mac của mạng IEEE 802.11 (DOMINO)
  - *Phần mềm được cài đặt tại/gần điểm truy cập có thể phát hiện và xác định những người chơi tham lam*
  - *Không thay đổi phần mềm của người chơi lành tính*



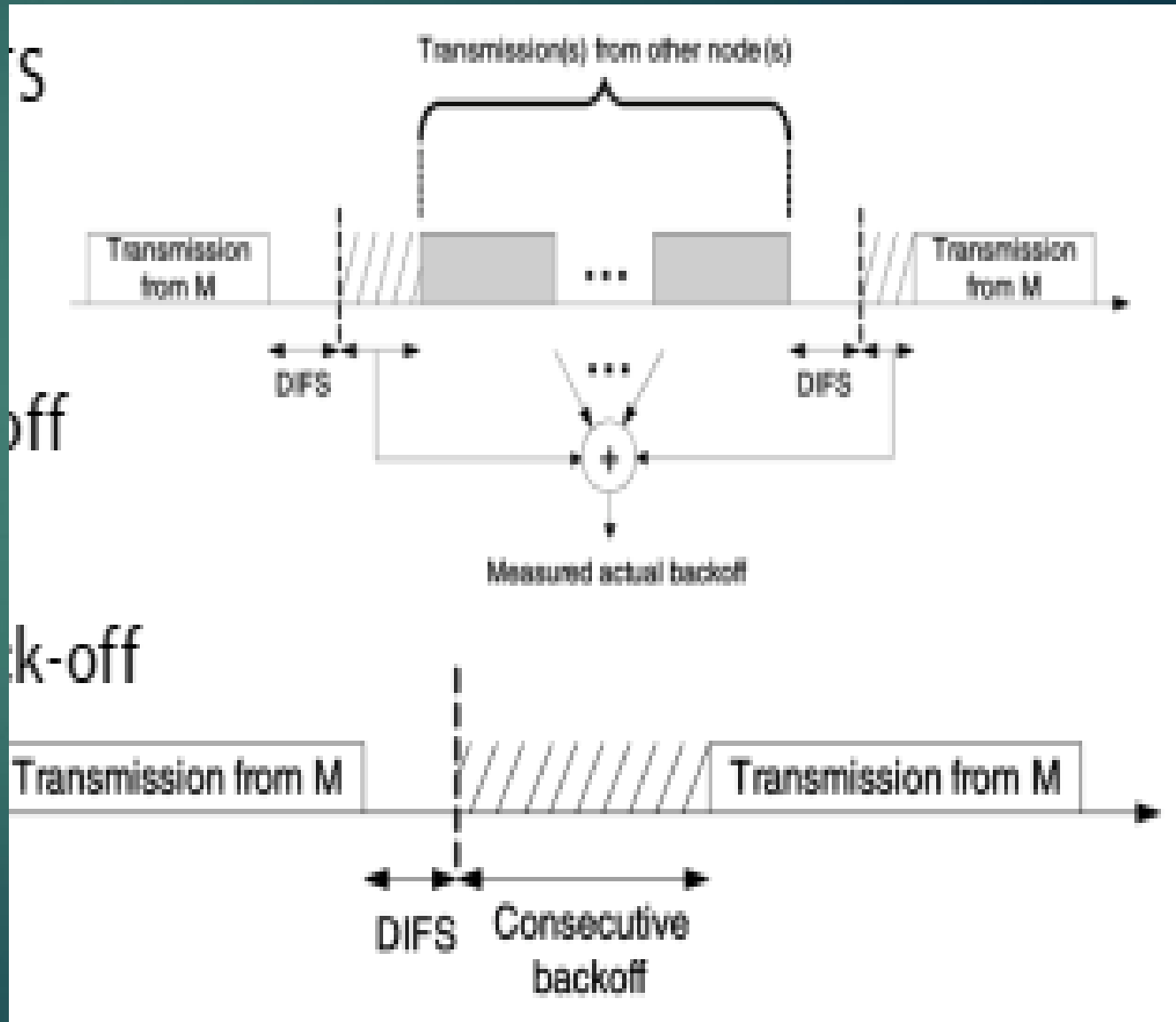


KIẾN TRÚC DOMINO



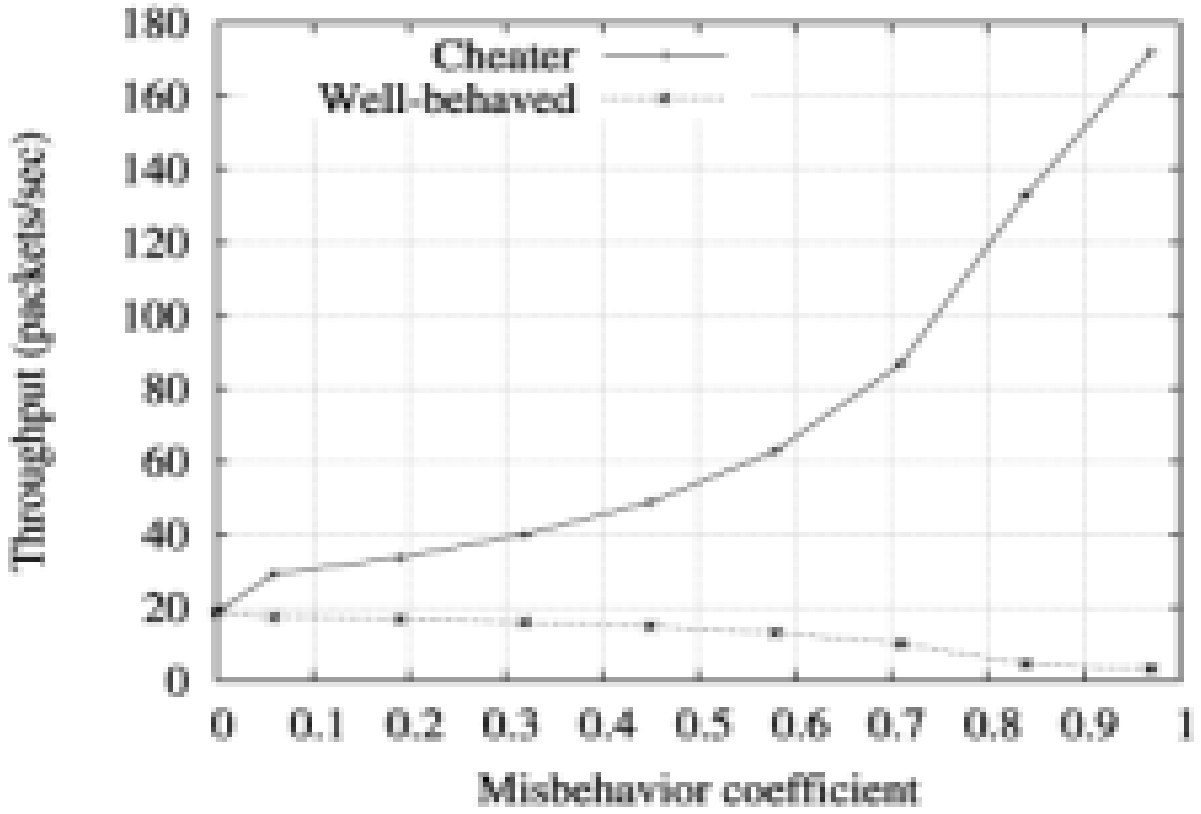
KIỂM TRA HÀNH VI

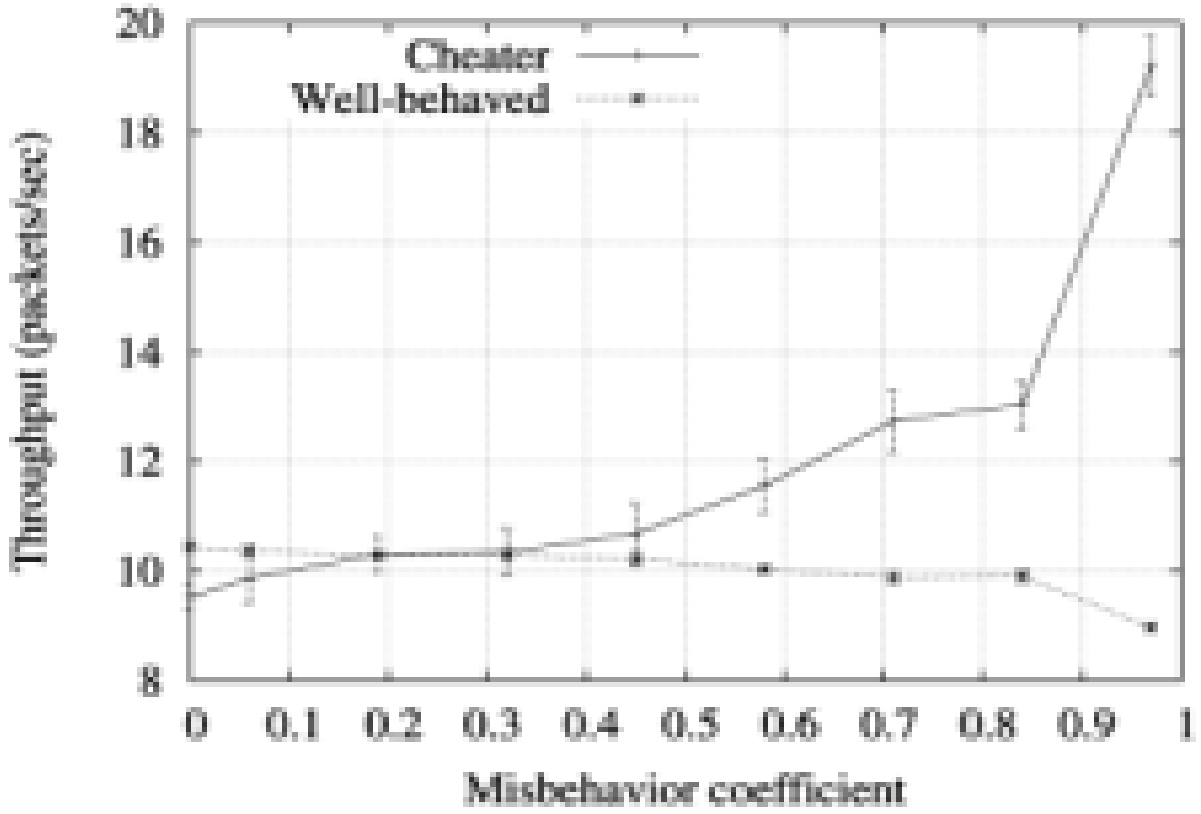
- AP kích hoạt DOMINO thực hiện một số bài kiểm tra hành vi làm cơ sở ra quyết định
  - Khung xáo trộn / truyền lại
  - Ngắn hơn DIFS
  - NAV ngoại cỡ
  - Khoảng thời gian chờ Back-of quan sát back-off



LƯU LƯỢNG UDP SO VỚI TCP

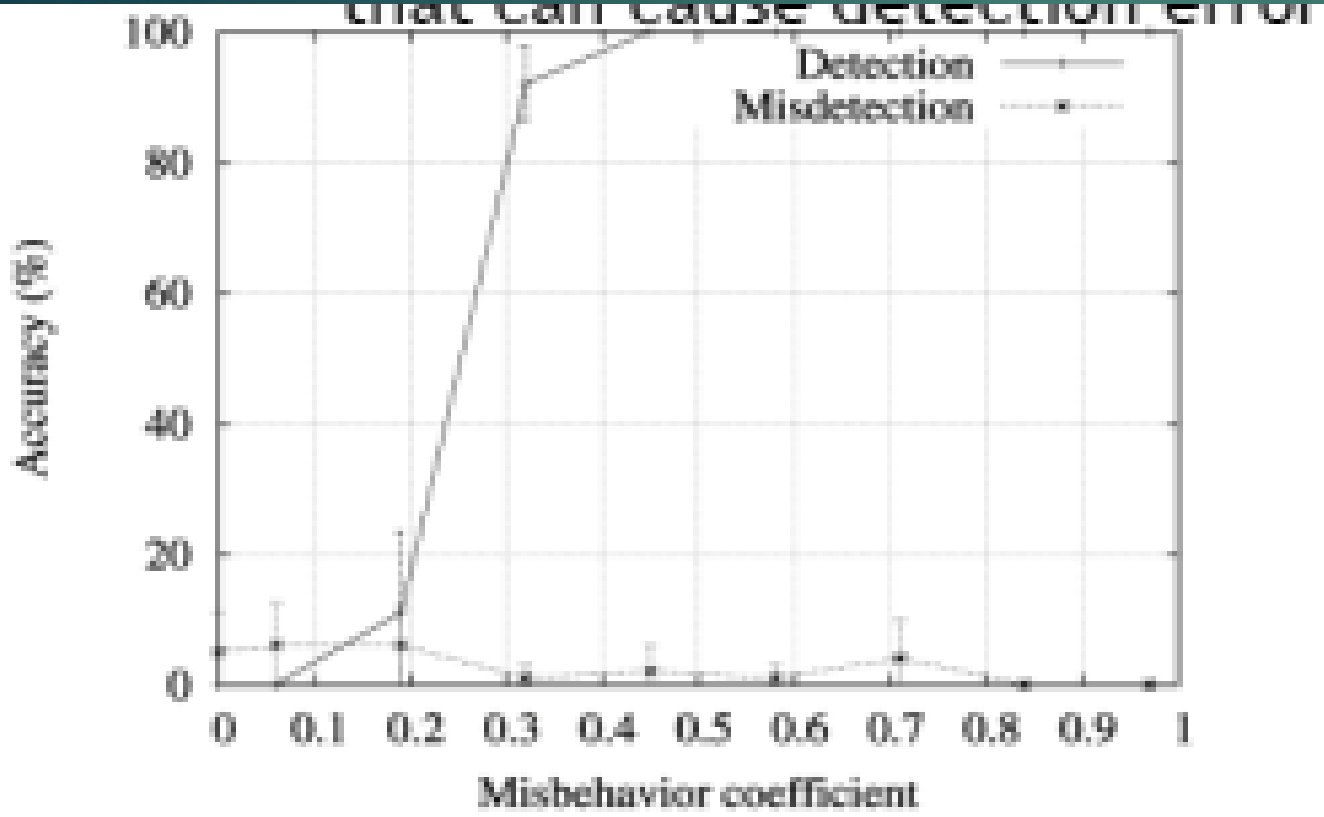
- Tác động của hành vi sai trái khác nhau đối với các loại lưu lượng truy cập mục tiêu khác nhau
  - *Chênh lệch lưu lượng giữa người gian lận và người dùng lành tính cao hơn trong trường hợp UDP*

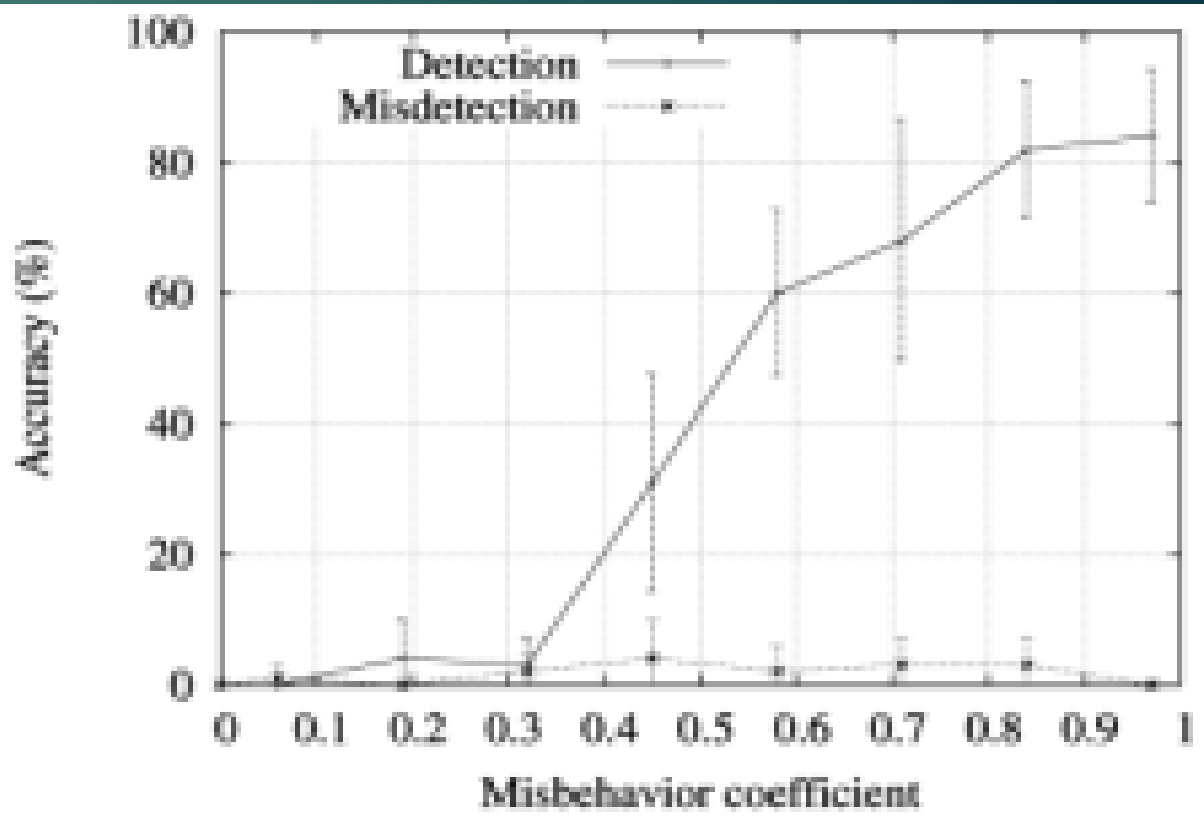




PHÁT HIỆN UDP SO VỚI TCP

- Loại lưu lượng cũng có tác động đáng kể đến khả năng phát hiện của DOMINO
  - Kiểm tra back-off thực tế trong UDP so với Kiểm tra back-off liên tục trong TCP
    - *Kiểm soát tắc nghẽn TCP gây ra các hành vi liên quan đến thời gian bổ sung có thể là nguyên nhân phát hiện lỗi*





THẢO LUẬN THÊM THÔNG QUA CÁC BÀI BÁO

- DOMINO nói về rất nhiều loại hành vi sai trái khác nhau
  - Các cuộc tấn công gây nhiễu, hành vi sai thời gian, v.v.
- Thiết kế hệ thống có thể triển khai
  - Rất nhiều thông số thiết kế để lựa chọn
  - Phân tích nhiều loại hành vi sai trái
  - Kết hợp các cơ chế bảo mật, chất lượng dịch vụ, các kịch bản lỗi không dây (ví dụ: thiết bị đầu cuối ẩn)

CÔNG BẰNG TRONG 802.11

- 802.11 kết hợp nhiều cơ chế công bằng khác nhau
  - Cung cấp sự công bằng bất kể chất lượng kết nối
  - Cho phép các kết nối chất lượng thấp chiếm phương tiện lâu hơn nhiều so với các kết nối chất lượng cao

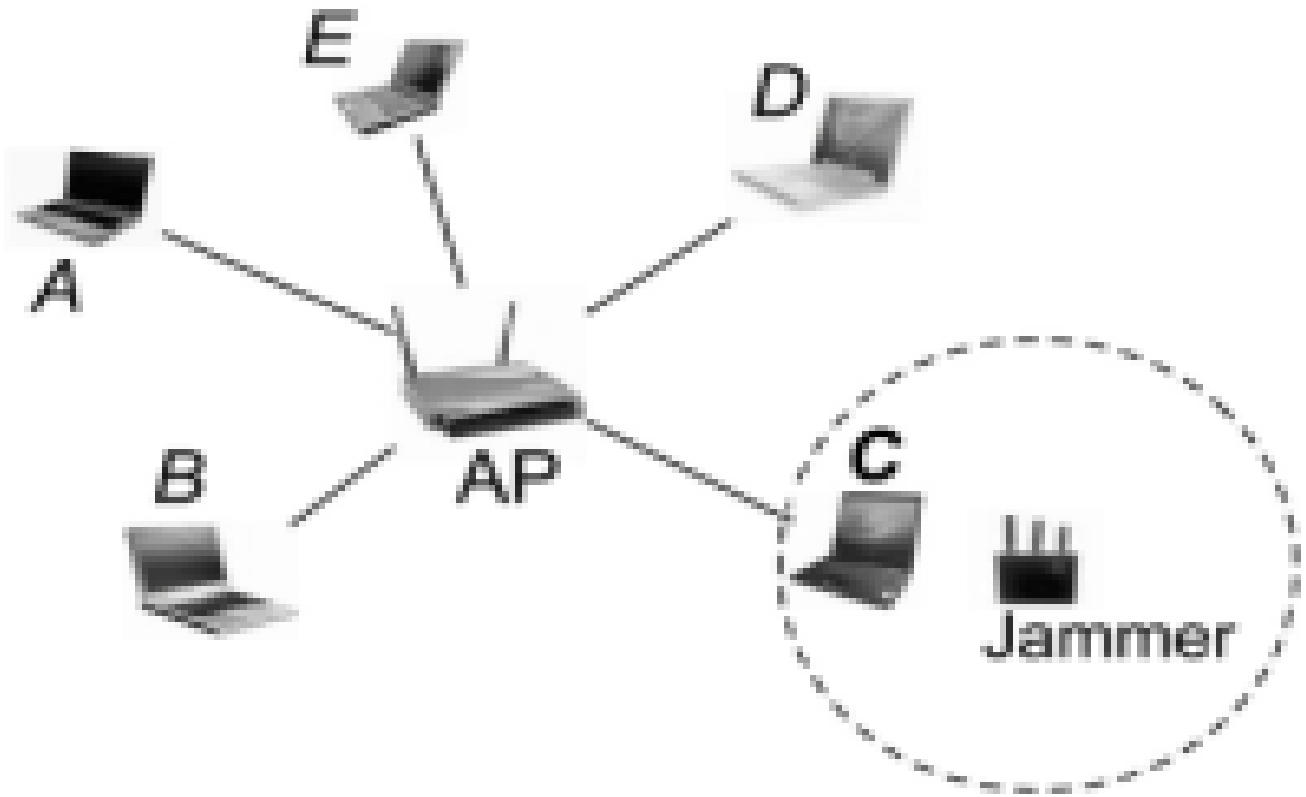
GÂY NHIỄU TIỀM ẨN TRONG 802.11

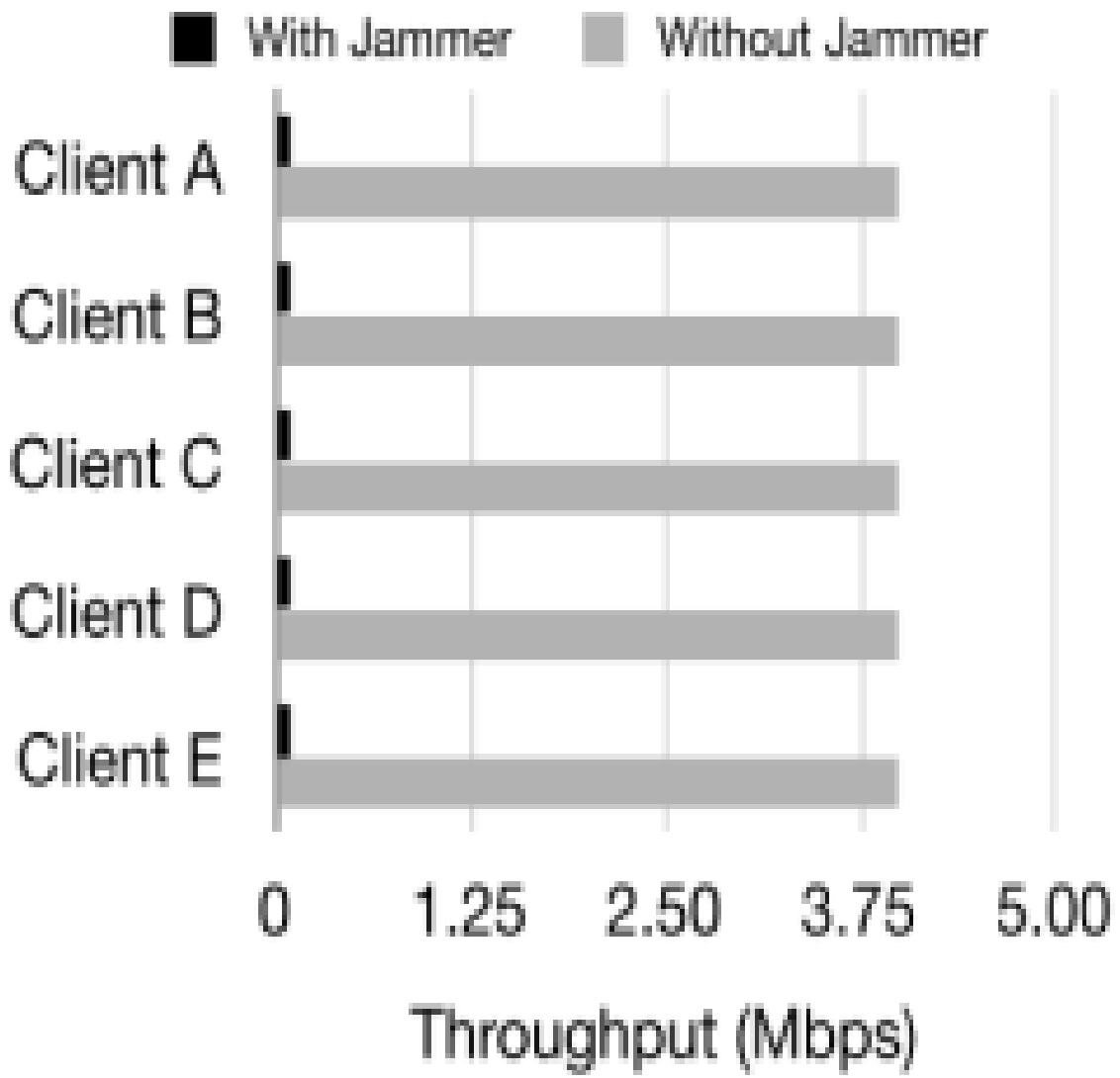
[Broustis và cộng sự, 2009]

- 802.11 có một cơ chế cân bằng tích hợp, về cơ bản cho phép tất cả người dùng nhận được cùng một thông lượng dài hạn
  - Kẻ tấn công thông minh có thể lợi dụng thuộc tính này để từ chối dịch vụ của người khác bằng cách gây nhiễu một người dùng
  - Sự hạ cấp của một người dùng khiến những người dùng khác cạn kiệt tài nguyên
  - Việc gây nhiễu một nút cuối không nhất thiết phải được AP quan sát, vì vậy việc phát hiện khó hơn nhiều

GÂY NHIỄU TIỀM ẨN

- Thiết bị gây nhiễu công suất thấp tấn công một nút gần nó, làm giảm thông lượng cho mọi người dùng sử dụng cùng một AP





GIẢM THIỂU GÂY NHIỄU TIỀM ẨN

- FIJI: giảm thiểu chống nhiễu của cuộc tấn công gây nhiễu ngầm
  - **Mục tiêu 1**: đảm bảo rằng các nút không bị tấn công không bị ảnh hưởng gián tiếp bởi cuộc tấn công
  - **Mục tiêu 2:** đảm bảo lượng lưu lượng tối đa được gửi đến nút bị tấn công, với điều kiện là nút đó đang bị tấn công
  - Cả hai mục tiêu đều dựa vào việc phát hiện rõ ràng cuộc tấn công gây nhiễu

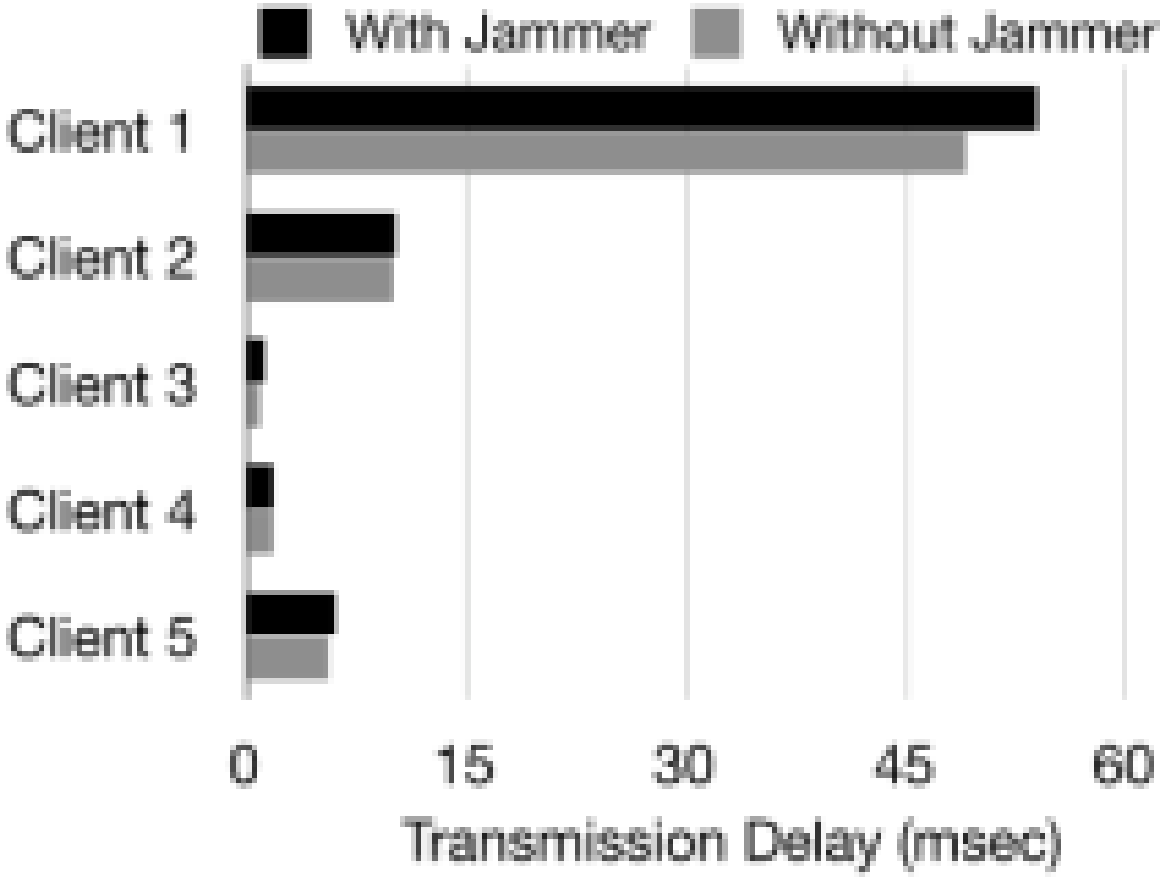
THÀNH PHẦN PHÁT HIỆN FIJI

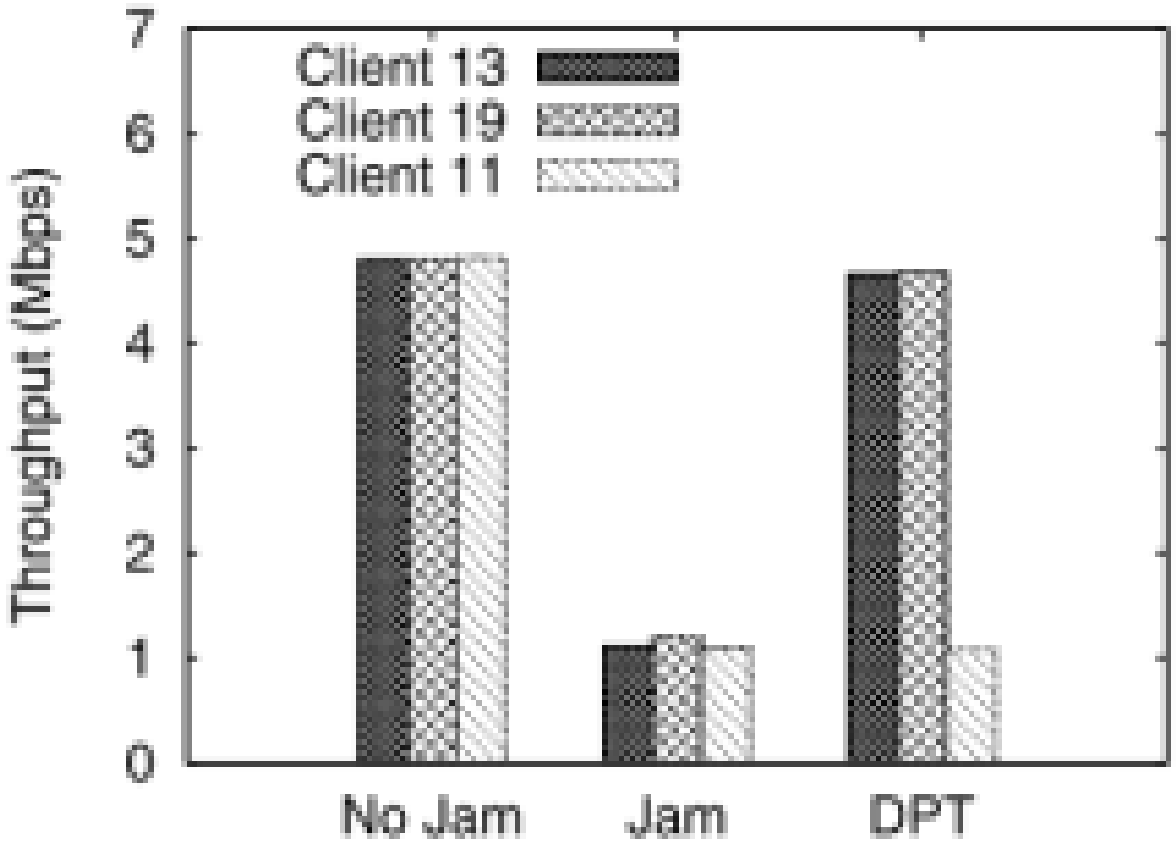
- Vì FIJI được điều hành/quản lý hoàn toàn tại AP, nên việc phát hiện cũng phải diễn ra tại đó; phát hiện tấn công gây nhiễu không điển hình
- Cơ chế phát hiện gây nhiễu tiêu chuẩn (ví dụ: sử dụng RSSI+PDR) không áp dụng, cần các số liệu khác
- Thay vào đó, hãy tìm những thay đổi về độ trễ truyền
  - *Thời gian giao dịch đo được tăng rất lớn cho thấy nút đang bị tấn công*

THÀNH PHẦN GIAO THÔNG FIJI

- Điều chỉnh các mẫu lưu lượng truy cập cho tất cả khách hàng dựa trên việc phát hiện các sự kiện (events)
  - **Giải pháp đơn giản:** không gửi bất kỳ dữ liệu nào cho các máy khách bị đang bị chèn, nhưng điều này là không công bằng và có thể dẫn đến các vấn đề lớn nếu phát hiện ra bất kỳ lỗi nào.
  - Chấp nhận hạ lưu lượng cho nút bị tấn công, nhưng giữ nguyên lưu lượng cho các nút khác
  - Hai cách tiếp cận để xử lý nút bị tấn công:
    - *Điều chỉnh kích thước gói dữ liệu: các gói phân mảnh ngắn hơn có nhiều khả năng đi qua*
    - *Điều chỉnh tốc độ dữ liệu: gửi đến các nút ít bị kẹt hơn*

ĐÁNH GIÁ FIJI





BUỔI 10:

CÁC MỐI ĐE DỌA Ở LỚP MẠNG;

QUẢN LÝ ĐỊNH DANH