Improving TCP Performance Over 802.11x MAC in Wireless Ad-hoc Network by Modified MAC Protocol

Presented by 何德威
MNET Lab, NTHU
eddie.ee87@nctu.edu.tw

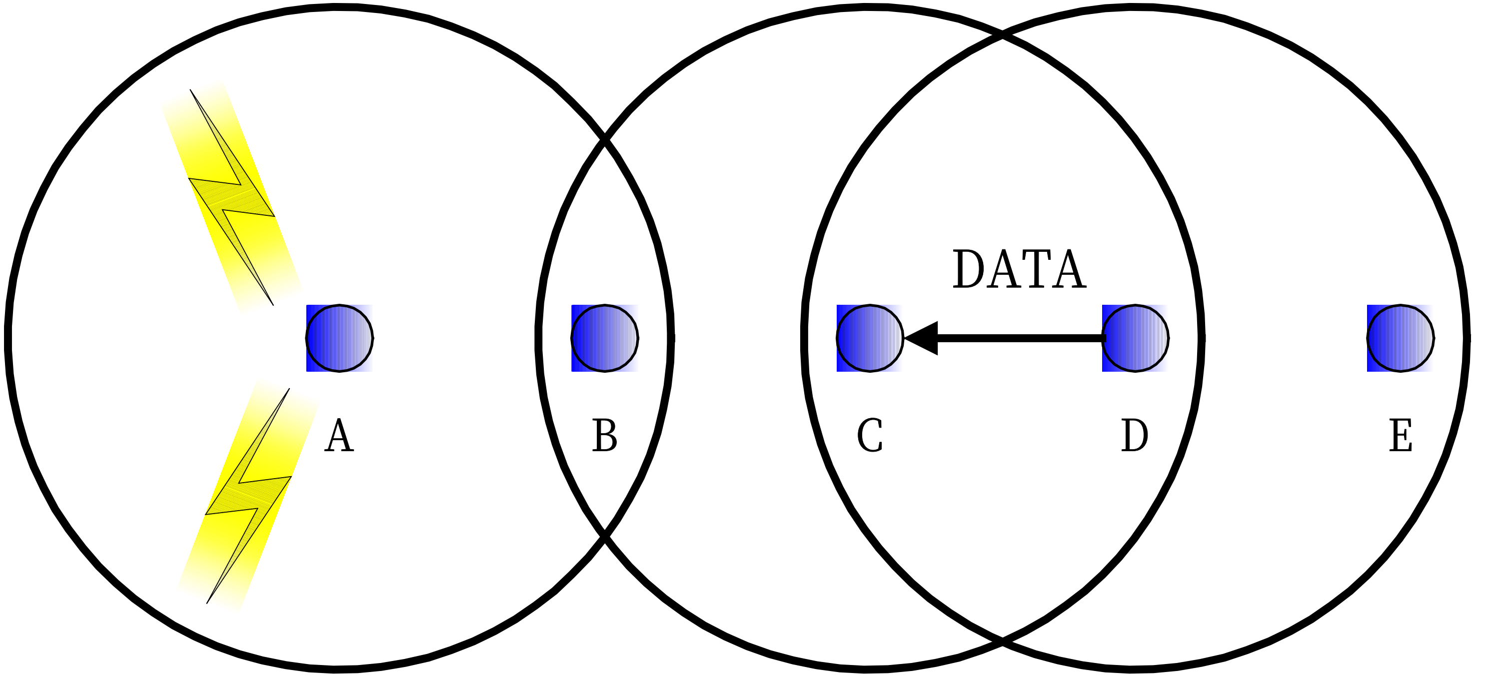
OUTLINE

- Introduction
- Background – Hidden Terminal Problem
- Interactions Between TCP and 802.11x MAC
  - TCP max congestion window size
  - Retry limit in 802.11x MAC
  - Capture effect
  - TCP packet size
- Proposed Techniques to Improve TCP Throughput in INFOCOM 2003
- Simulation Results
- Conclusions

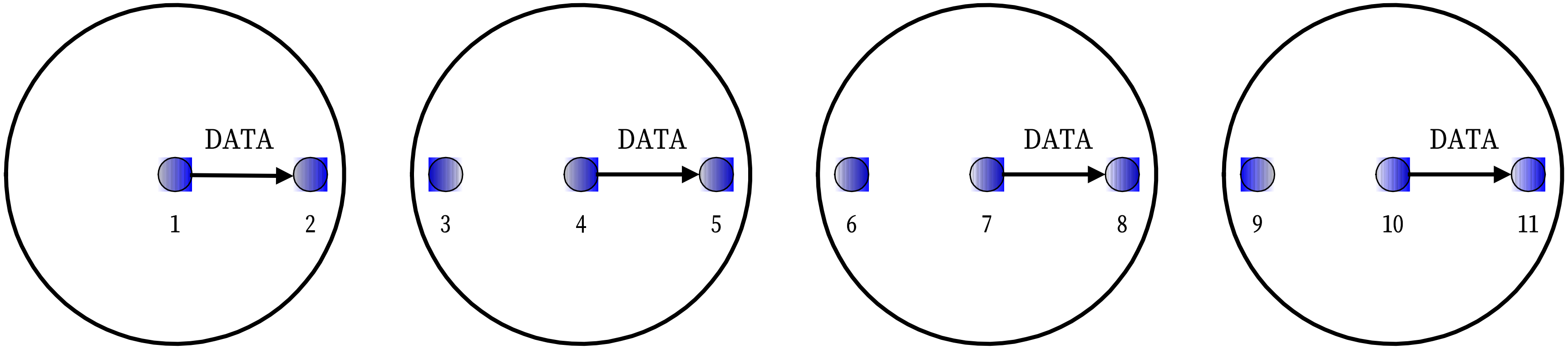
INTRODUCTION

- TCP aspects：
  - TCP assumes a reliable link layer, and packet loss always due to congestion.
  - So TCP does not differentiate between *”congestion-related packet drops”* and *”transmission failures at link layer”*.
- 802.11 MAC aspect：
  - The 802.11x MAC protocol has a *limited view* of the network condition. The information available is *the status of its neighbors*.
  - So hidden terminal problem still persists in 802.11 ad-hoc network.

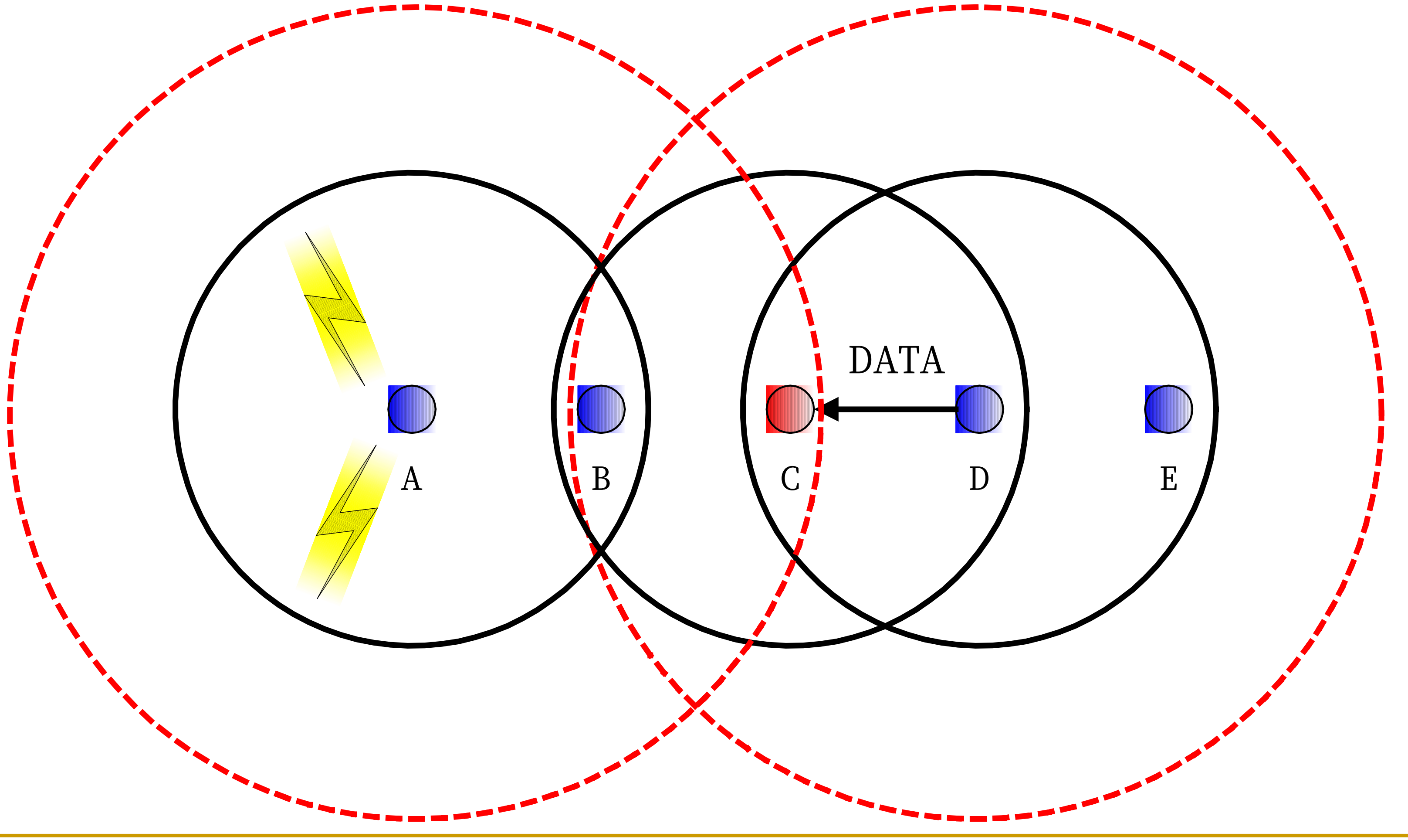
BACKGROUND – HIDDEN TERMINAL PROBLEM(1/2)



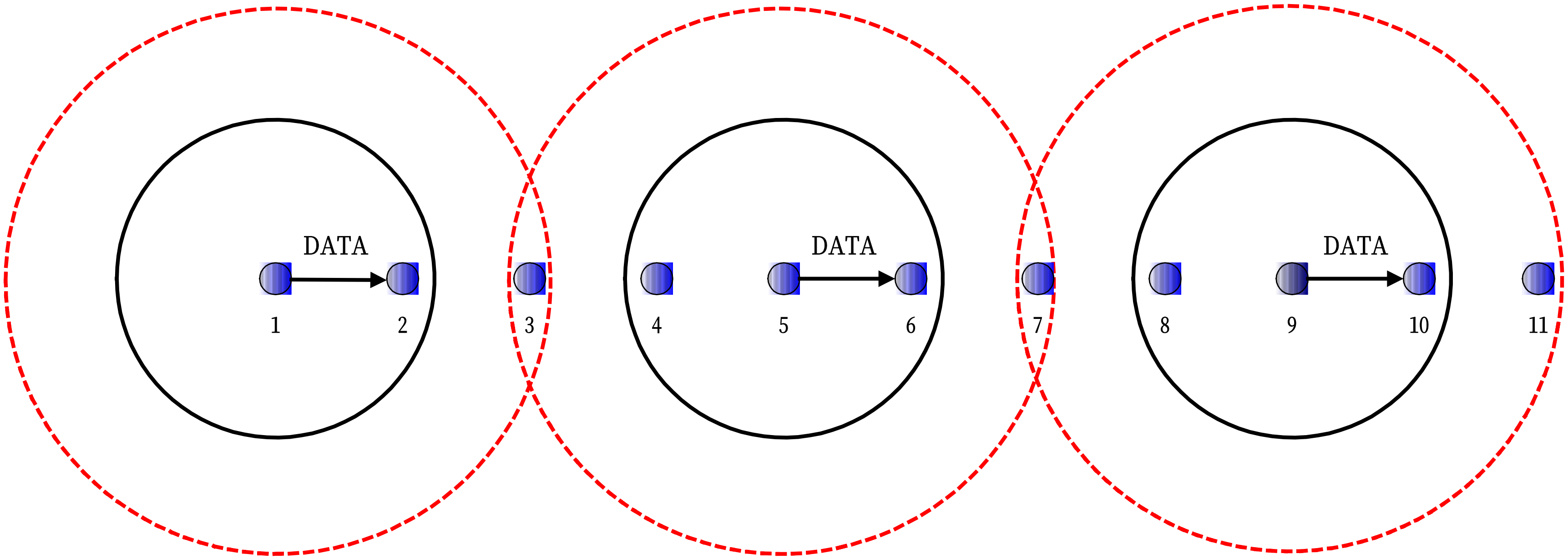
BACKGROUND – HIDDEN TERMINAL PROBLEM(1/2)



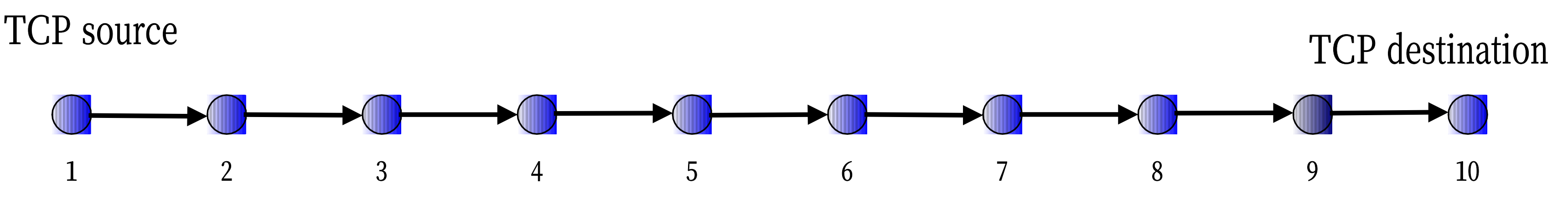
BACKGROUND – HIDDEN TERMINAL PROBLEM(1/2)



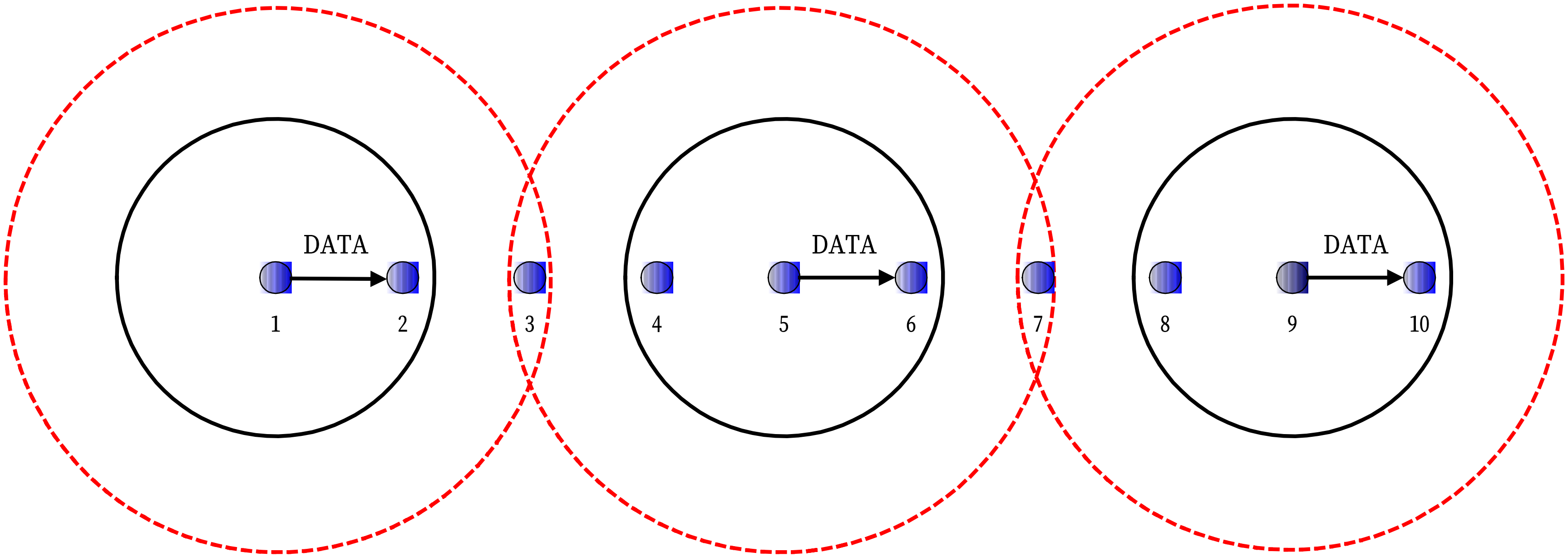
BACKGROUND – HIDDEN TERMINAL PROBLEM(1/2)



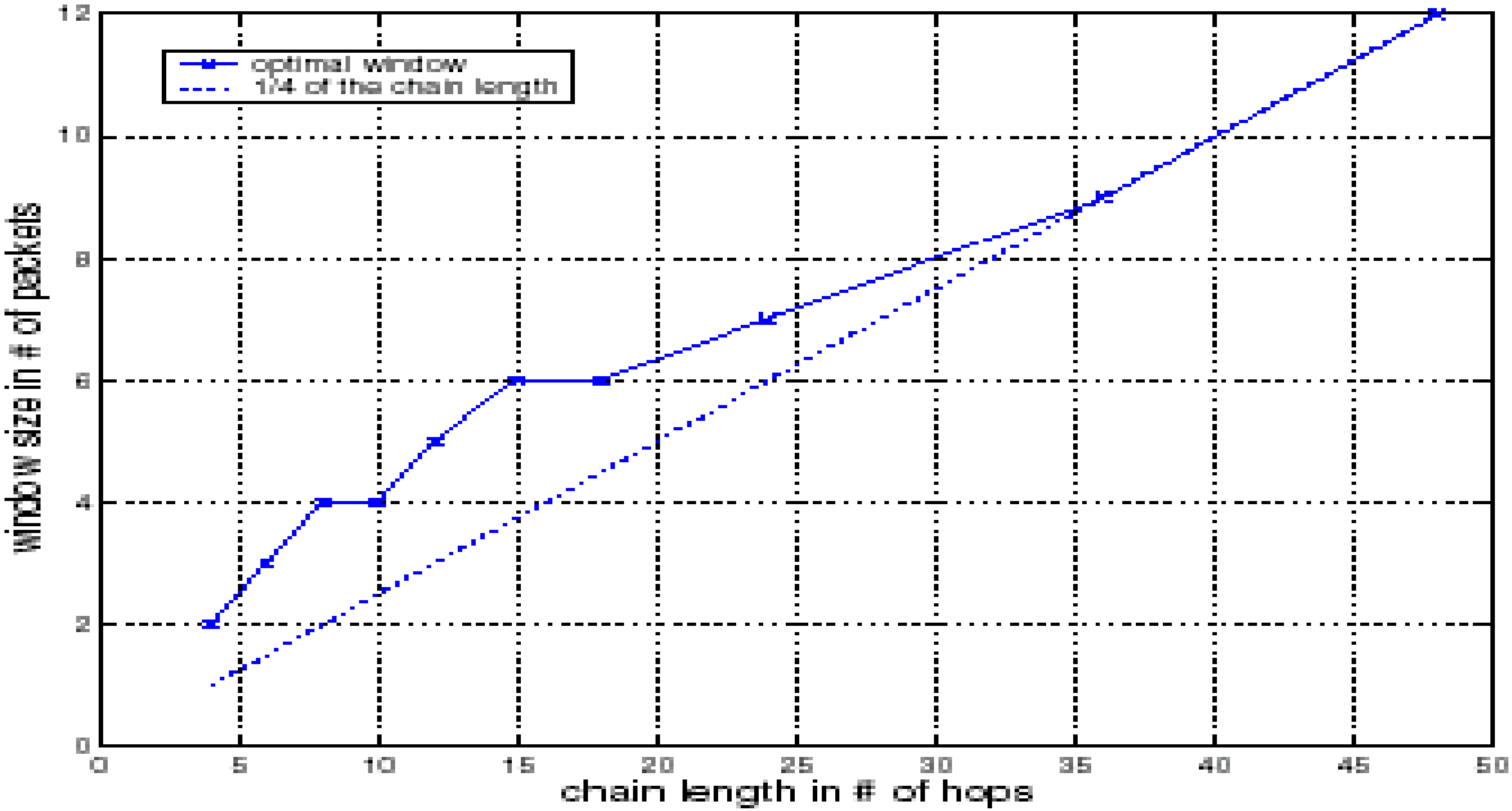
TCP MAX CONGESTION WINDOW SIZE(1/3)



TCP MAX CONGESTION WINDOW SIZE(2/3)



TCP MAX CONGESTION WINDOW SIZE(3/3)



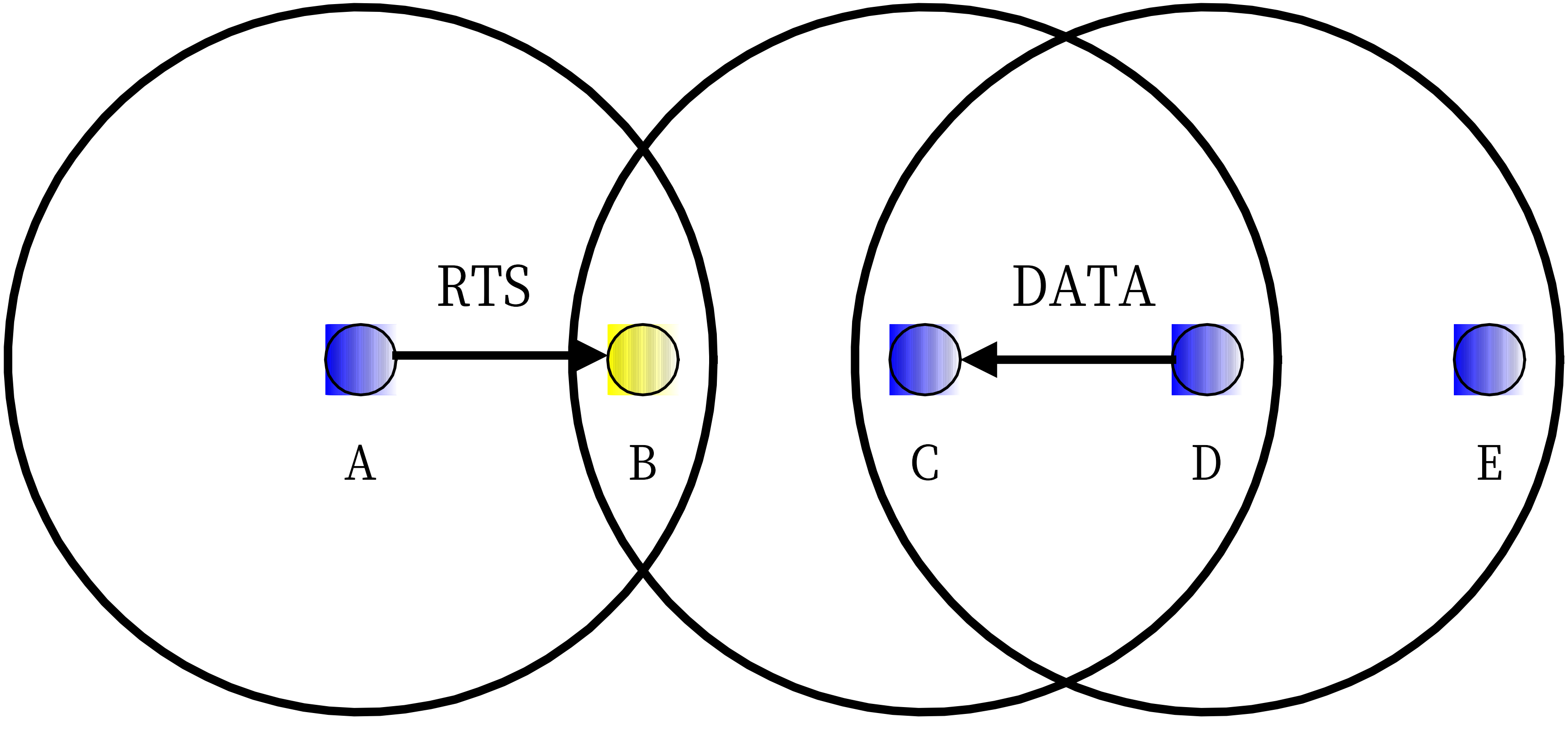
RETRY LIMIT IN 802.11x MAC

- For a *static* topology or a *low mobility* environment, *increasing the retry limits* results in significant improvement of performance.
- For a highly mobile environment, such a increase will result in increased delay in link failure detection.

CAPTURE EFFECT

- Capture effect in wireless cellular network：
  - Binary exponential back-off *favors the last successful node*.
  - Thus, the connections *starting early* or *more heavily loaded* ones may have a higher probability to capture the channel.
- Capture effect in wireless multihop network：
  - A *short connection* (in hops) tends to dominate over longer connections and capture more bandwidth.

TCP PACKET SIZE(1/2)



TCP PACKET SIZE(2/2)

- An increase in packet size reduces the fraction of packets delivered successfully.
- It also increases the packet drop probability.
- However, larger packets provide better *channel utilization*.

PROPOSED TECHNIQUES TO IMPROVE TCP THROUGHPUT(1/2)

Algorithm 1 L-RED: LinkLayerSend(Packet p)

Require: avg_retry is the average MAC retries for each packet

if $avg\_retry < min\_th$ **then**
    $mark\_prob \leftarrow 0$
    $pacing \leftarrow ON$
else
    $mark\_prob = min\{\frac{avg\_retry - min\_th}{max\_th - min\_th}, max\_P\}$
    set $pacing$ OFF
end if
mark p with $mark\_prob$
MacLayerSend(p, $pacing$)
$retry$ = GetMacRetries()
$avg\_retry = \frac{7}{8}avg\_retry + \frac{1}{8}retry$

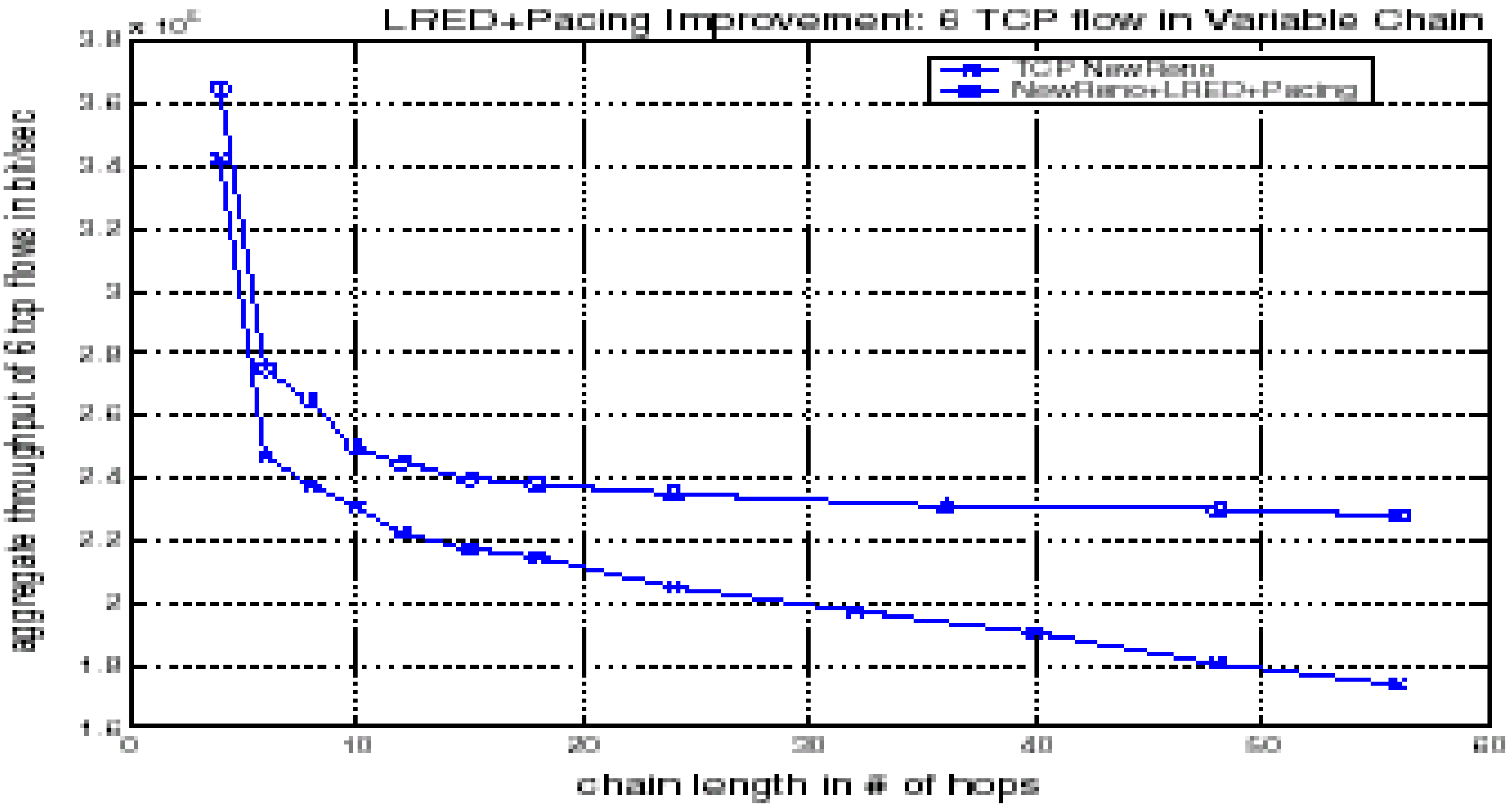
PROPOSED TECHNIQUES TO IMPROVE TCP THROUGHPUT(2/2)

Algorithm 2 Adaptive Pacing

Require: *extra_Backoff* = 0

```
if received ACK then
    random_Backoff ← ran_backoff(cong_win) {DATA transmission succeeded. Setup the backoff timer}
    if pacing is ON then
        extra_Backoff = TX_Time(DATA) + overhead
    end if
    backoff ← random_Backoff + extra_Backoff
    start backoff_timer
end if
```

SIMULATION RESULTS(1/4)



SIMULATION RESULTS(2/4)

	TCP NewReno w/standard LL	TCP NewReno w/LL+LRED+PACING
flow 1	532 Kbps	85512 Kbps
flow 2	126229 Kbps	90459 Kbps
flow 3	115554 Kbps	70334 Kbps
flow 4	1608 Kbps	47946 Kbps
Aggregate	242923	294251
Fairness	0.51	0.95

TABLE VI

Throughput and Fairness Comparisons between NewReno and NewReno+LRED+PACING. 4 flows in 13×13 Grid.

SIMULATION RESULTS(3/4)

	TCP NewReno w/standard LL	TCP NewReno w/LL+LRED+PACING
flow 1	532 Kbps	85512 Kbps
flow 2	126229 Kbps	90459 Kbps
flow 3	115554 Kbps	70334 Kbps
flow 4	1608 Kbps	47946 Kbps
Aggregate	242923	294251
Fairness	0.51	0.95

TABLE VI

Throughput and Fairness Comparisons between NewReno and NewReno+LRED+PACING. 4 flows in 13×13 Grid.

SIMULATION RESULTS(4/4)

	NR Aggregate	NR Fairness	LRED+ Aggregate	LRED+ Fairness
2 flows	203K bps	0.502	252K bps	0.921
4 flows	241K bps	0.508	294K bps	0.952
8 flows	824K bps	0.524	963K bps	0.527
12 flows	690K bps	0.455	880K bps	0.56

TABLE VII

Aggregate throughput and fairness comparisons between NewReno and NewReno+LRED+PACING with 2, 4, 8 and 12 flows in grid topology.

CONCLUSIONS

- I have shown that both TCP and 802.11x MAC protocol are needed to be modified for better performance in the wireless ad-hoc network.
- The authors propose two link layer techniques, LRED and Adaptive Pacing, which can improve the throughput of standard TCP flows by as much as 30%.