

# Simple Model Analysis and Performance Tuning of Hybrid TCP Congestion Control

J.Katto, K.Ogura, Y.Akai, T.Fujikawa, K.Kaneko & S.Zhou

Dept. of Computer Science, Waseda University



# [ Outline ]

- Research Backgrounds & Objectives
  - TCP Variants & our TCP-Fusion
- Analytical Models and Evaluations
  - Ideal Models (Loss, Delay, Hybrid)
  - Abstraction of Actual Hybrids (CTCP, ARENO, YeAH, Fusion)
    - Performance Tuning
  - Response Function
  - Evaluations
- Conclusions





# Backgrounds & Objectives



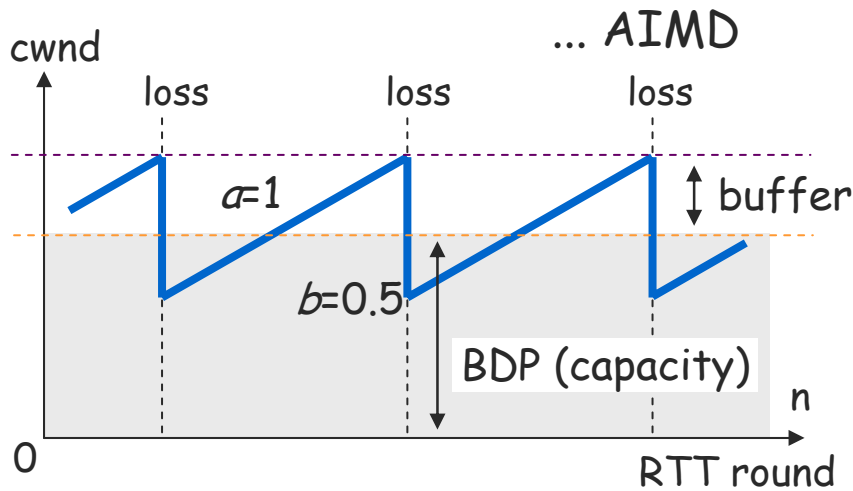
# [ Backgrounds: TCP Variants (1) ]

- Loss-based (AIMD: Additive Increase Multiplicative Decrease)
  - TCP-Reno / NewReno / SACK
  - High-Speed TCP (IETF RFC 3649, Dec 2003)
  - Scalable TCP (PFLDnet 2003)
  - BIC-TCP / CUBIC-TCP (IEEE INFOCOM 2004, PFLDnet 2005)
  - H-TCP (PFLDnet 2004)
  - TCP-Westwood (ACM MOBICOM 2001)
- Delay-based (RTT Observation)
  - TCP-Vegas (IEEE JSAC, Oct 1995)
  - FAST-TCP (INFOCOM 2004)
- Hybrid (Loss & RTT)
  - Gentle High-Speed TCP (PfHSN 2003)
  - TCP-Africa (IEEE INFOCOM 2005)
  - Compound TCP (PFLDnet 2006)
  - Adaptive Reno (PFLDnet 2006)
  - TCP-Illinois (ValueTools 2006)
  - YeAH-TCP (PFLDnet 2007)
  - TCP-Fusion (PFLDnet 2007)
- etc, etc, etc, ...

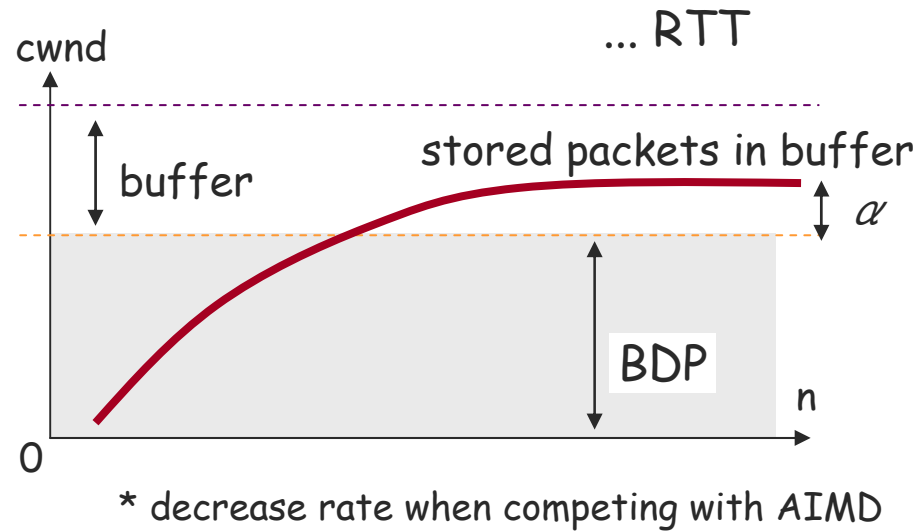


# [ Backgrounds: TCP Variants (2) ]

## ■ Loss-based

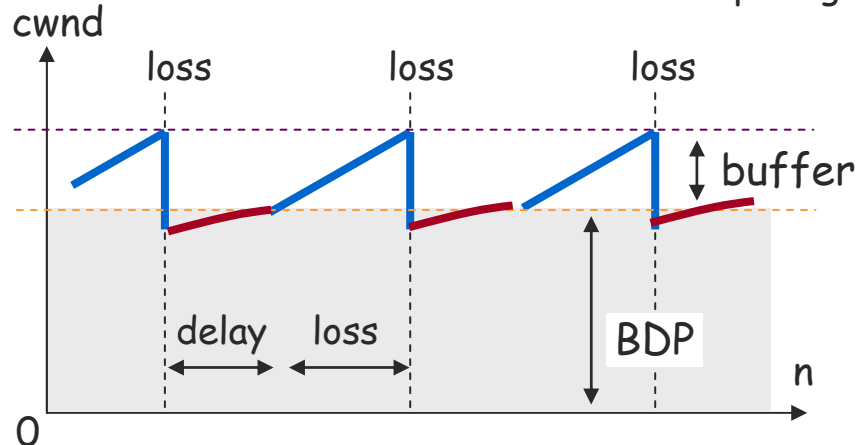


## ■ Delay-based



## ■ Hybrid

- no RTT increase: Delay mode
- RTT increase: Loss mode

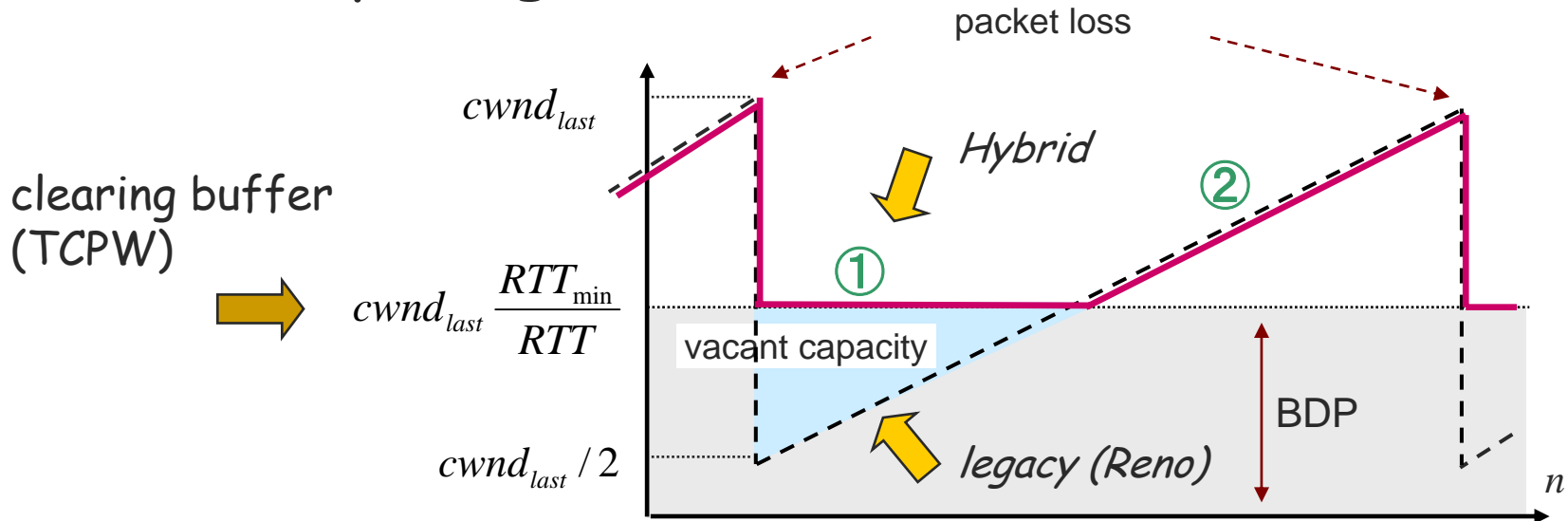


BDP: Bandwidth-Delay Product



# Backgrounds: TCP-Fusion (1)

- Basic mechanism (single flow case: no competing flow, buffer < BDP)



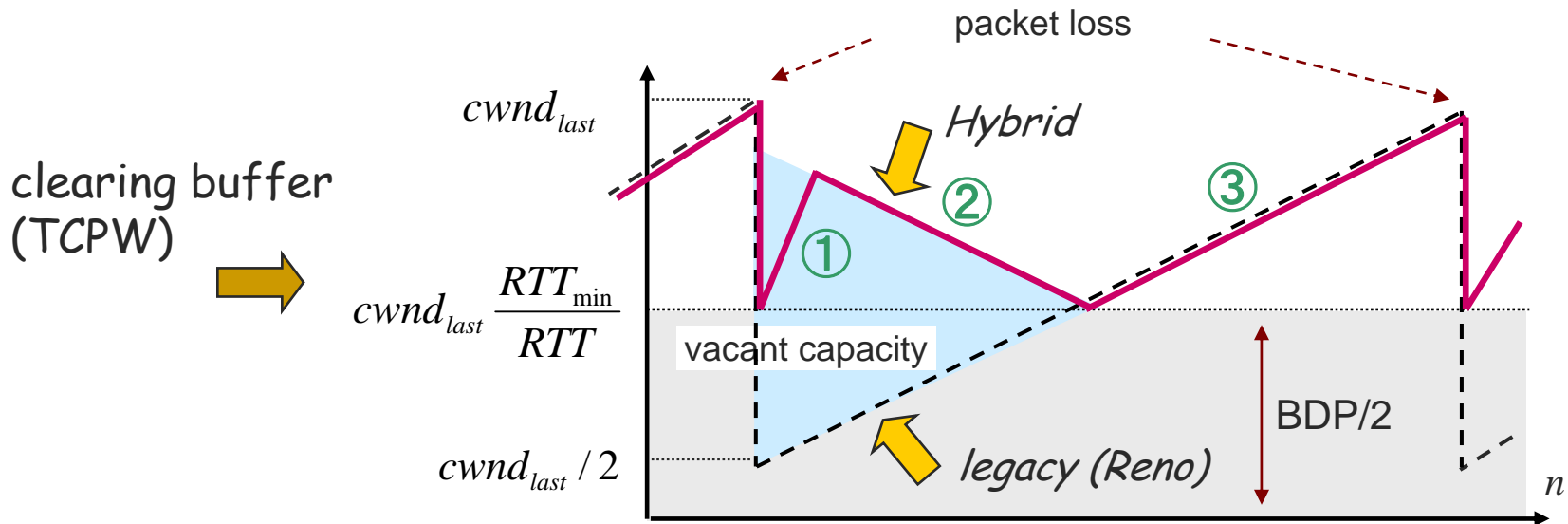
Window control by switching two modes (loss & delay):

- Keeps constant rate until RTT increases (delay mode)
- Performs as TCP-Reno (loss mode, when triggered by virtual reno-cwnd)



# Backgrounds: TCP-Fusion (2)

- Basic mechanism (two flow case: when competing with TCP-Reno, buffer < BDP)



Window control by switching two modes (loss & delay):

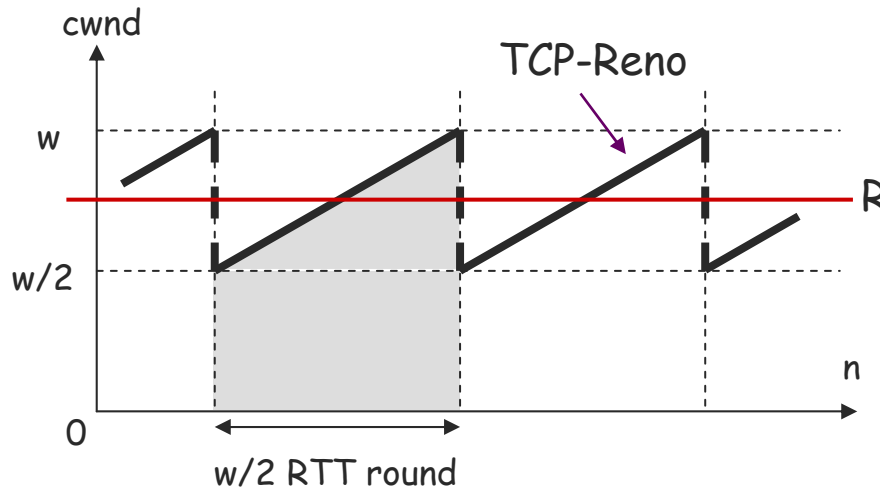
- ① Rapid increase of  $cwnd$  (delay ... throughput efficiency)
- ② Gradual decrease of  $cwnd$  (delay ... avoids congestion)
- ③ Performs as TCP-Reno (loss ... friendliness to legacy TCP)



# [ Backgrounds: TCP Modeling ]

## ■ Congestion Avoidance Behavior

\* equation based rate control



$w$ : cwnd when packet losses happen

$p$ : packet loss ratio

RTT: round trip time

$b$ : delayed ACK counts

$t_{RTO}$ : retransmission timeout

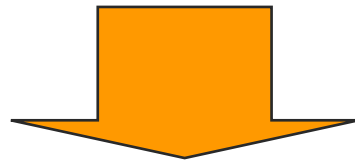
$R$ : (equivalent) rate

$$\begin{aligned} &\text{with timeout consideration} \\ &\left\{ \begin{aligned} p &= \frac{8}{3w^2} \\ R &= \frac{PS}{RTT} \cdot \sqrt{\frac{3}{2p}} \end{aligned} \right. \quad \Rightarrow \quad R_{loss} = \frac{PS}{RTT \sqrt{\frac{2bp}{3}} + t_{RTO,loss} \cdot 3 \sqrt{\frac{3bp}{8}} \cdot p(1+32p^2)} \end{aligned}$$



# [Problems & Objectives]

- On Hybrid TCP Congestion Control,
  - Theoretical background is not sufficient (best performance is not clear)
  - Too many control parameters (we have to tune them by many simulations and many implementations)



- Build analytical models for hybrid TCP congestion control
- Theoretically validate its throughput efficiency and friendliness to legacy TCPs
- Enable parameter tuning
- Derive a response function



# Analytical Models & Evaluations

1. Ideal models
2. Abstraction of actual hybrids
  - Parameter tuning
3. Response function



# [ Ideal Model (1) ]

## ■ Definition

- Loss-based (AIMD: TCP-Reno) :
  - $\text{cwnd} += 1$  (per RTT round)
  - $\text{cwnd} *= 1/2$  (when packet losses happen)
- Delay-based :
  - always fills a pipe (BDP) without causing RTT increase
- Hybrid :
  - performs in delay-mode when RTT stays at its minimum
  - performs in loss-mode when RTT increase is observed



# [ Ideal Model (2) ]

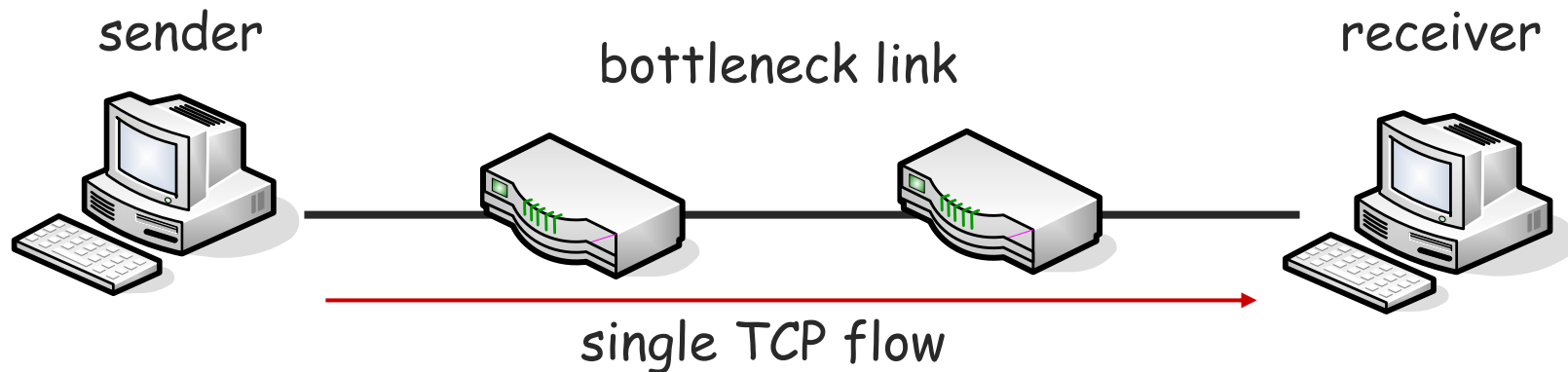
- Parameter definition
  - $w$ : cwnd when packet losses happen
  - $W$ : number of packets corresponding to BDP
  - $p$ : packet loss rate
- Assumption (from TCP modeling)
  - Next equation holds for random loss case (e.g. RED router, small buffer and wireless) as well as for buffer overflow case

$$p = \frac{8}{3w^2} \quad (\text{in case of TCP-Reno})$$



# [ Single Flow Model (1) ]

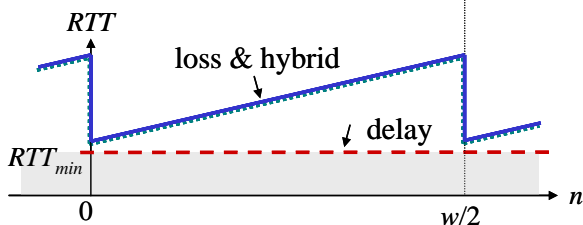
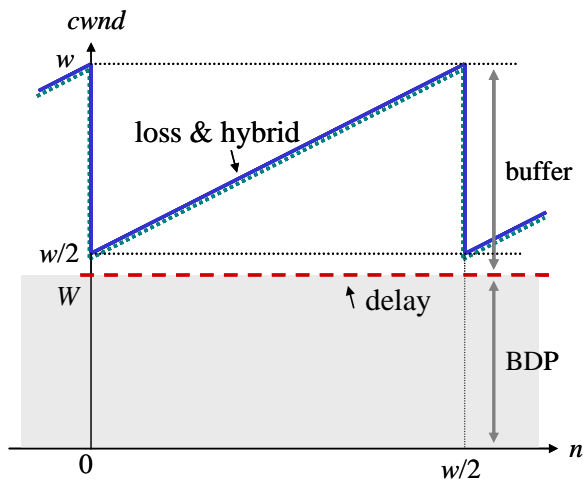
- Connection Topology



# [ Single Flow Model (2) ]

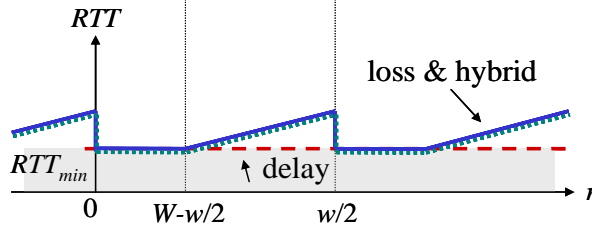
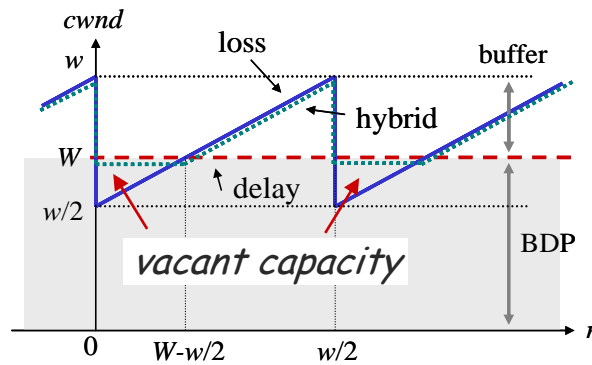
## ■ Classification to three cases

$$w(\text{PLR}) \sim W(\text{BDP})$$



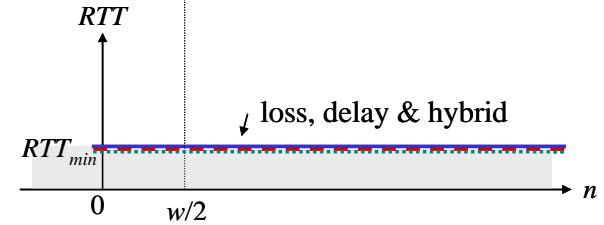
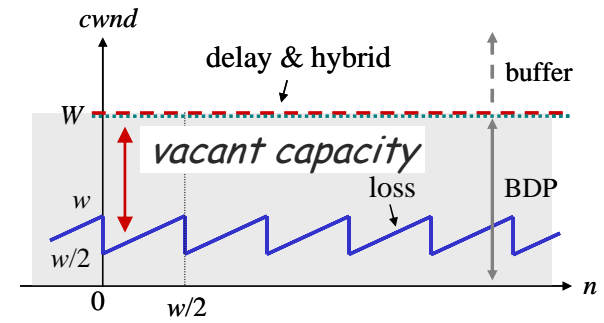
(i)  $W < w/2$  (low PLR)

always buffered  
(loss-mode)



(ii)  $w/2 < W < w$  (medium PLR)

vacant  $\rightarrow$  buffered  
(delay  $\rightarrow$  loss)



(iii)  $w < W$  (high PLR)

always vacant  
(delay-mode)



# [ Single Flow Model (3) ]

- Transmitted packets and elapsed time per single congestion avoidance round

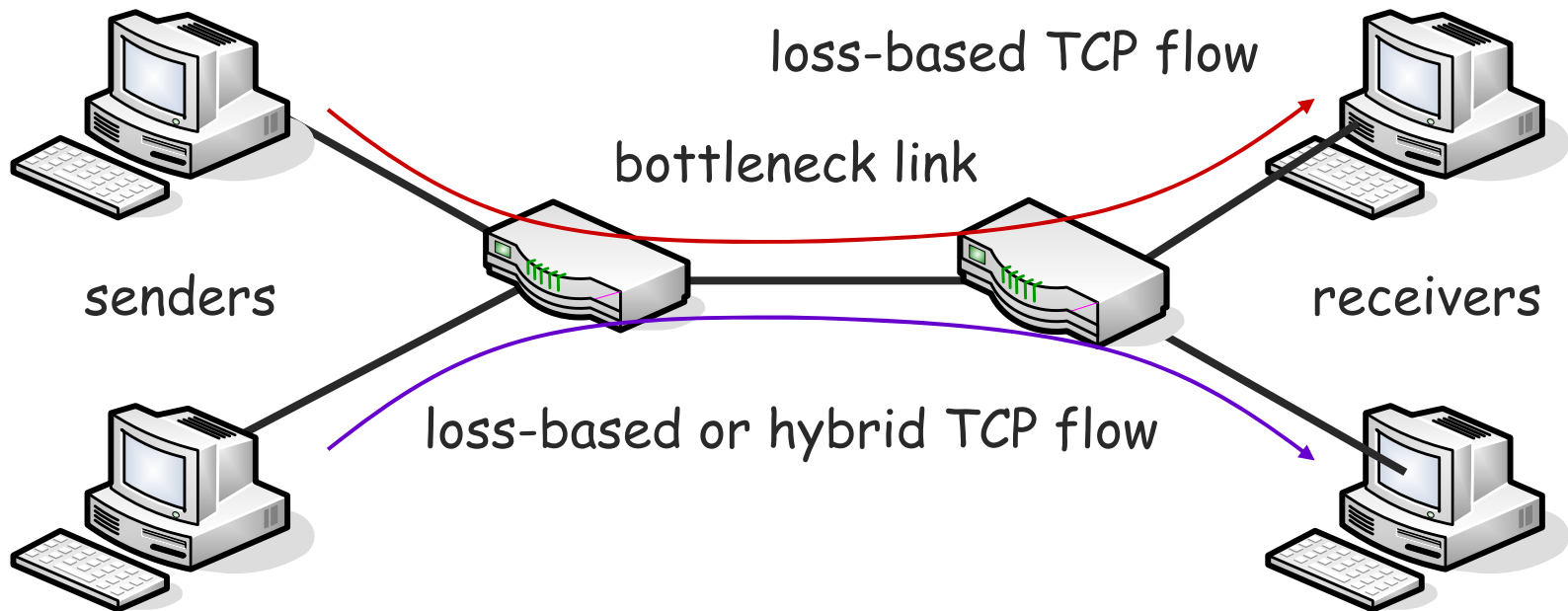
TCP	CA round	(i) $W < w/2$	(ii) $w/2 \leq W < w$	(iii) $w \leq W$
Loss	transmitted packets	$\frac{3}{8}w^2$	$\frac{3}{8}w^2$	$\frac{3}{8}w^2$
	elapsed time	$\frac{1}{2}w \cdot RTT_{\min} + \frac{1}{8}(3w^2 - 4wW) \cdot \frac{PS}{B}$	$\frac{1}{2}w \cdot RTT_{\min} + \frac{1}{2}(w-W)^2 \cdot \frac{PS}{B}$	$\frac{1}{2}w \cdot RTT_{\min}$
Delay	transmitted packets	$\frac{1}{2}w \cdot W$	$\frac{1}{2}w \cdot W$	$\frac{1}{2}w \cdot W$
	elapsed time	$\frac{1}{2}w \cdot RTT_{\min}$	$\frac{1}{2}w \cdot RTT_{\min}$	$\frac{1}{2}w \cdot RTT_{\min}$
Hybrid	transmitted packets	$\frac{3}{8}w^2$	$\frac{1}{2}w \cdot W + \frac{1}{2}(w-W)^2$	$\frac{1}{2}w \cdot W$
	elapsed time	$\frac{1}{2}w \cdot RTT_{\min} + \frac{1}{8}(3w^2 - 4wW) \cdot \frac{PS}{B}$	$\frac{1}{2}w \cdot RTT_{\min} + \frac{1}{2}(w-W)^2 \cdot \frac{PS}{B}$	$\frac{1}{2}w \cdot RTT_{\min}$

PS: Packet size, B: Link bandwidth



# [ Two Flow Model (1) ]

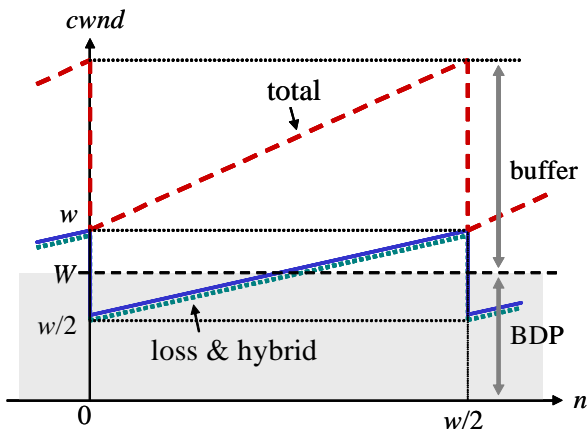
- Connection Topology



# [ Two Flow Model (2) ]

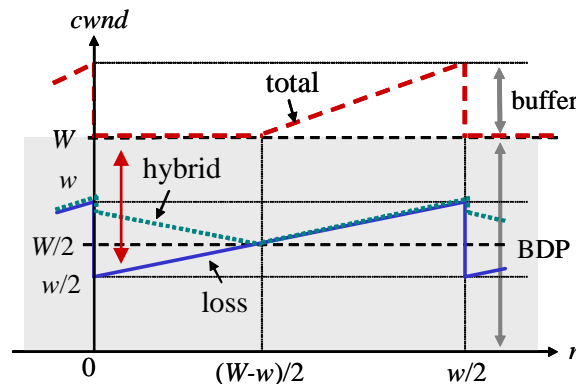
## ■ Classification to three cases

$$w(\text{PLR}) \sim W(\text{BDP})$$



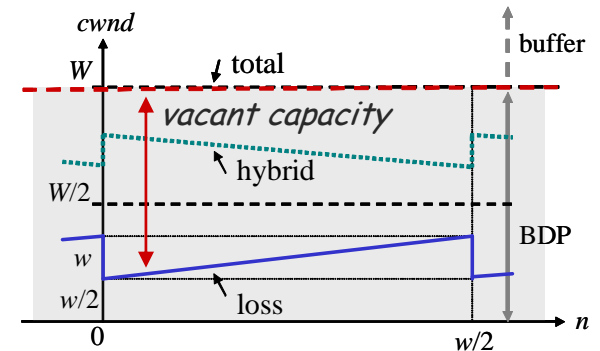
(i)  $W < w$  (low PLR)

always buffered  
(loss mode)



(ii)  $w < W < 2 \cdot w$  (medium PLR)

(vacant)  $\rightarrow$  filled  $\rightarrow$  buffered  
(delay  $\rightarrow$  loss)



(iii)  $2 \cdot w < W$  (high PLR)

always filled (vacant)  
(delay mode)

(1)  $w$  is scaled to half value

(2) delay-based TCP is omitted



# [ Two Flow Model (3) ]

- Transmitted packets and elapsed time per single congestion avoidance round

TCP	CA round	(i) $W < w$	(ii) $w \leq W < 2w$	(iii) $2w \leq W$
Loss	transmitted packets	$\frac{3}{8}w^2$	$\frac{3}{8}w^2$	$\frac{3}{8}w^2$
Hybrid	transmitted packets	$\frac{3}{8}w^2$	$\frac{3}{8}w^2 + \frac{1}{4}(W - w)^2$	$\frac{1}{2}w \cdot W - \frac{3}{8}w^2$
(common)	elapsed time	$\frac{1}{2}w \cdot RTT_{\min} + \frac{1}{4}w(3w - 2W) \cdot \frac{PS}{B}$	$\frac{1}{2}w \cdot RTT_{\min} + \frac{1}{4}(2w - W)^2 \cdot \frac{PS}{B}$	$\frac{1}{2}w \cdot RTT_{\min}$



# [ Abstraction of Hybrids (1) ]

- Window increase/decrease mechanisms

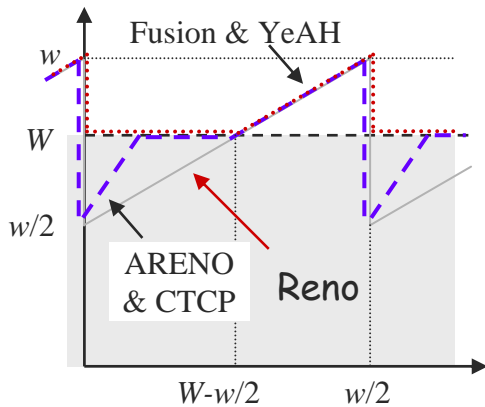
Hybrids	Window increase	Window decrease
CTCP [14]	$0.125 * cwnd^{0.75}$	1/2
ARENO [15]	B/10Mbps	1/2~1
YeAH-TCP [17]	STCP(1.01)	1/2, $RTT_{min}/RTT$ , 7/8
TCP-Fusion [18]	$B * D_{min} / (N * PS)$	$RTT_{min} / RTT$

$D_{min}$ : timer resolution, N: # of flows

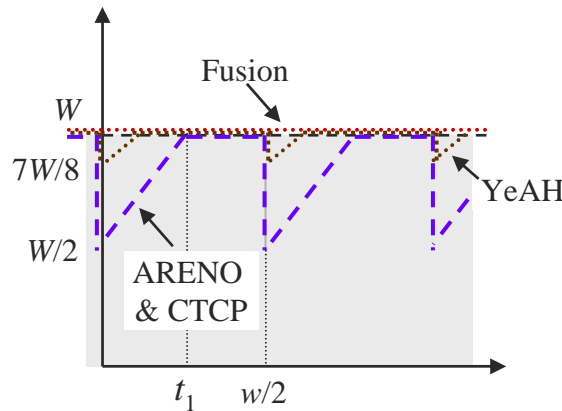


# [ Abstraction of Hybrids (2) ]

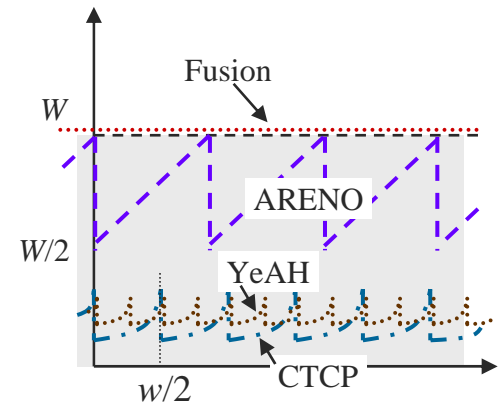
- Single flow case



(ii)  $w/2 < W < w$   
(medium PLR)



(iii)  $w < W, t_1 \leq w/2$   
(high PLR)



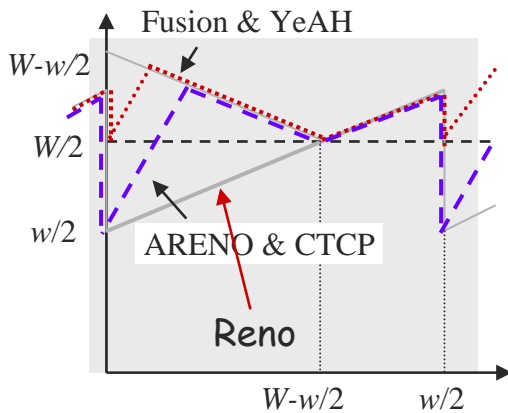
(iii)  $w < W, t_1 > w/2$   
(high PLR)

“Slow” window increase and “constant” window decrease result in periodic behavior in region (iii) → throughput efficiency would be degraded.

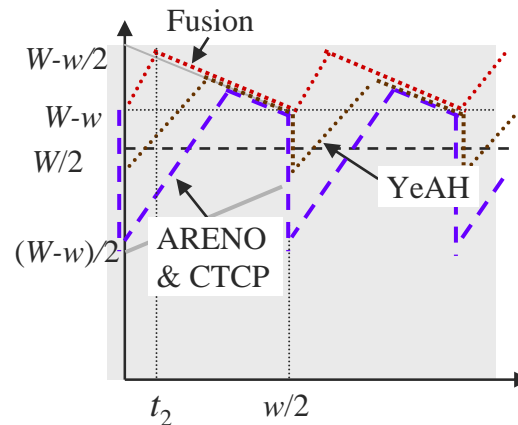


# [ Abstraction of Hybrids (3) ]

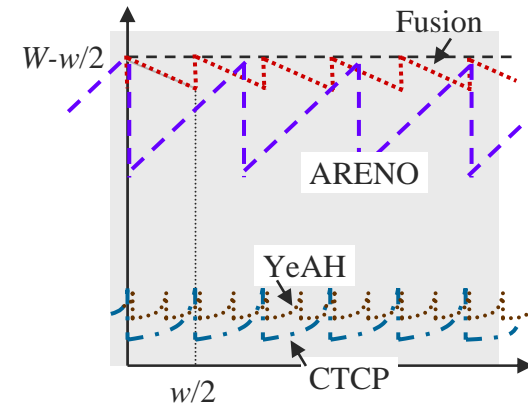
## ■ Two flow case



(ii)  $w < W < 2w$   
(medium PLR)



(iii)  $2w < W, t_2 \leq w/2$   
(high PLR)



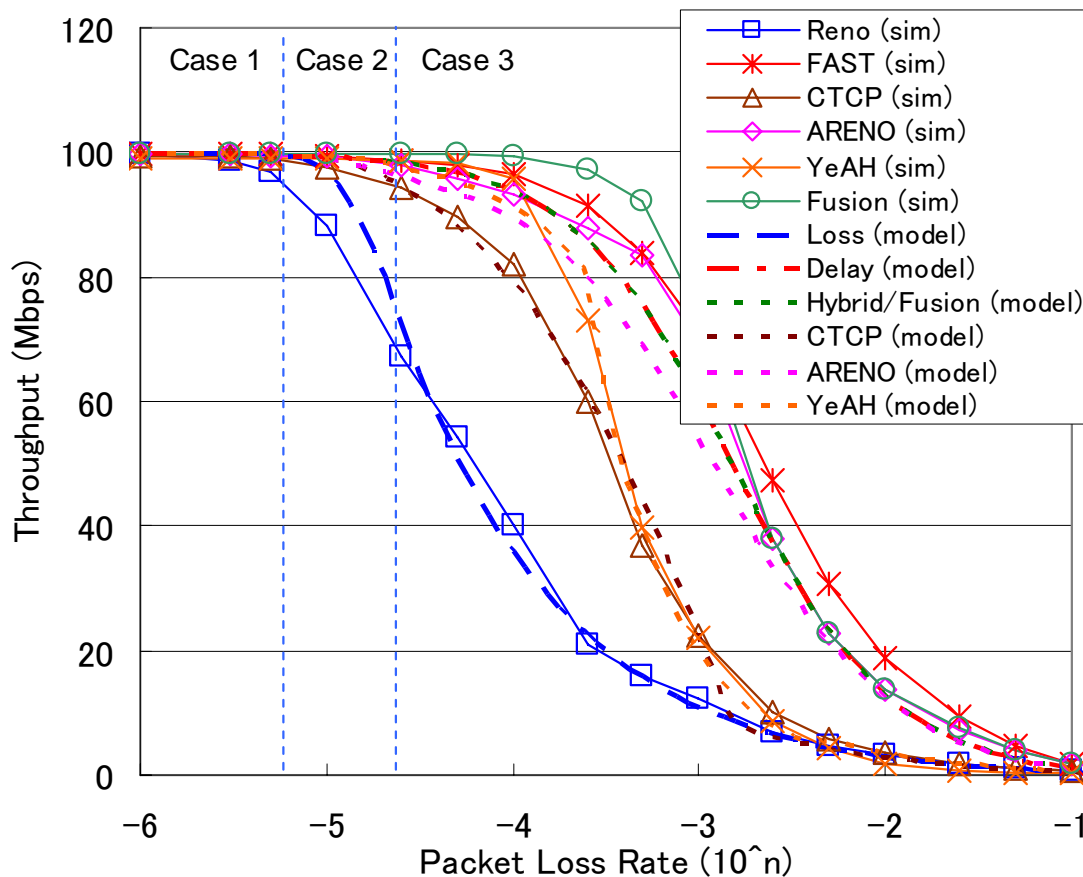
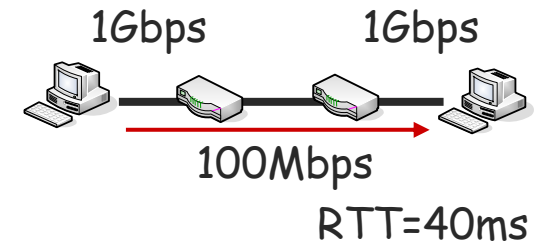
(iii)  $2w < W, t_2 > w/2$   
(high PLR)

"Slow" window increase and "constant" window decrease result in periodic behavior in region (iii) → throughput efficiency would be degraded.



# Evaluations (1)

## Single flow model



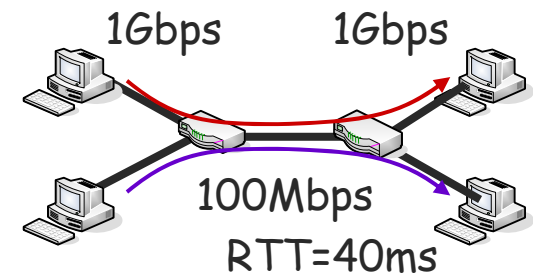
buffer size = BDP (constant)  
PLR : variable

When PLR is large (regions (ii) & (iii)), delay based and hybrid TCPs drastically outperform loss based TCP.

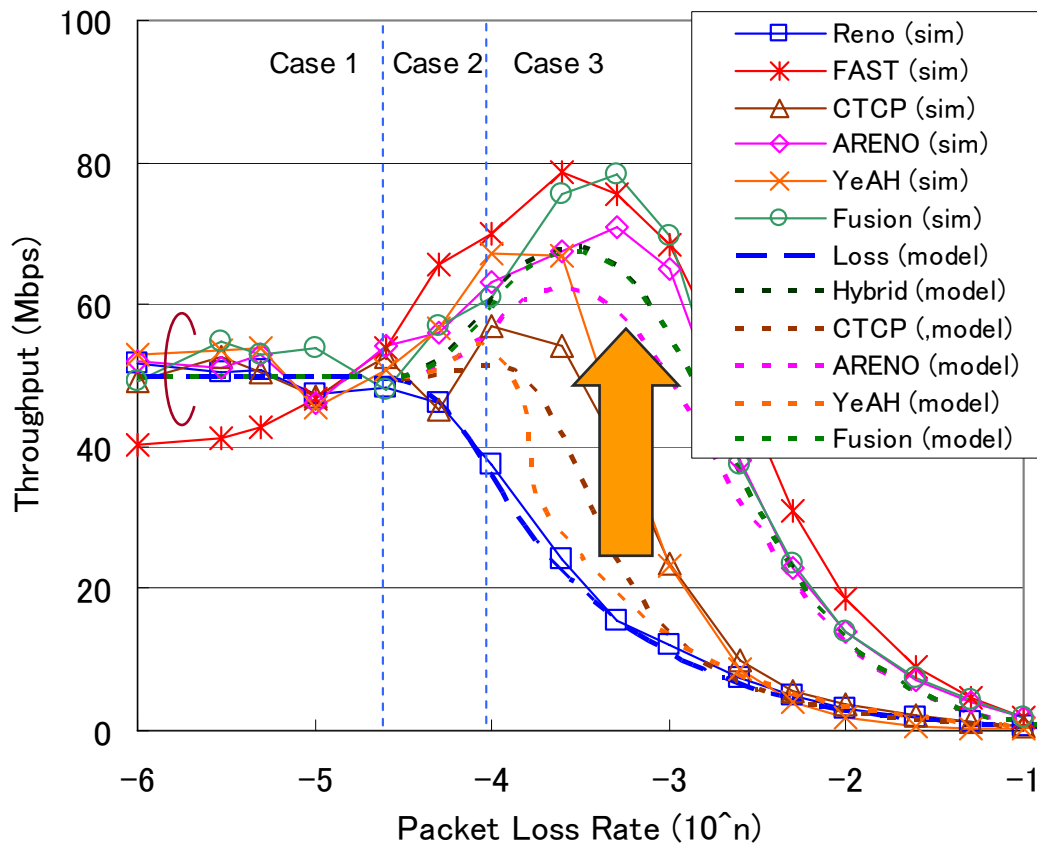
Performances of CTCP and YeAH are worse than the others.

# Evaluations (2)

## Two flow model



buffer size = BDP (constant)  
PLR : variable



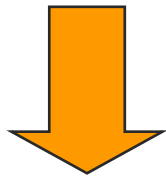
When PLR is low (region (i)), hybrid TCP shows good friendliness to legacy TCP.

When PLR is high (regions (ii) & (iii)), hybrid TCP shows high throughput efficiency, though CTCP and YeAH are worse.



# [ Performance Tuning ]

- “Slow” window increase and “constant” window decrease result in throughput degradations.

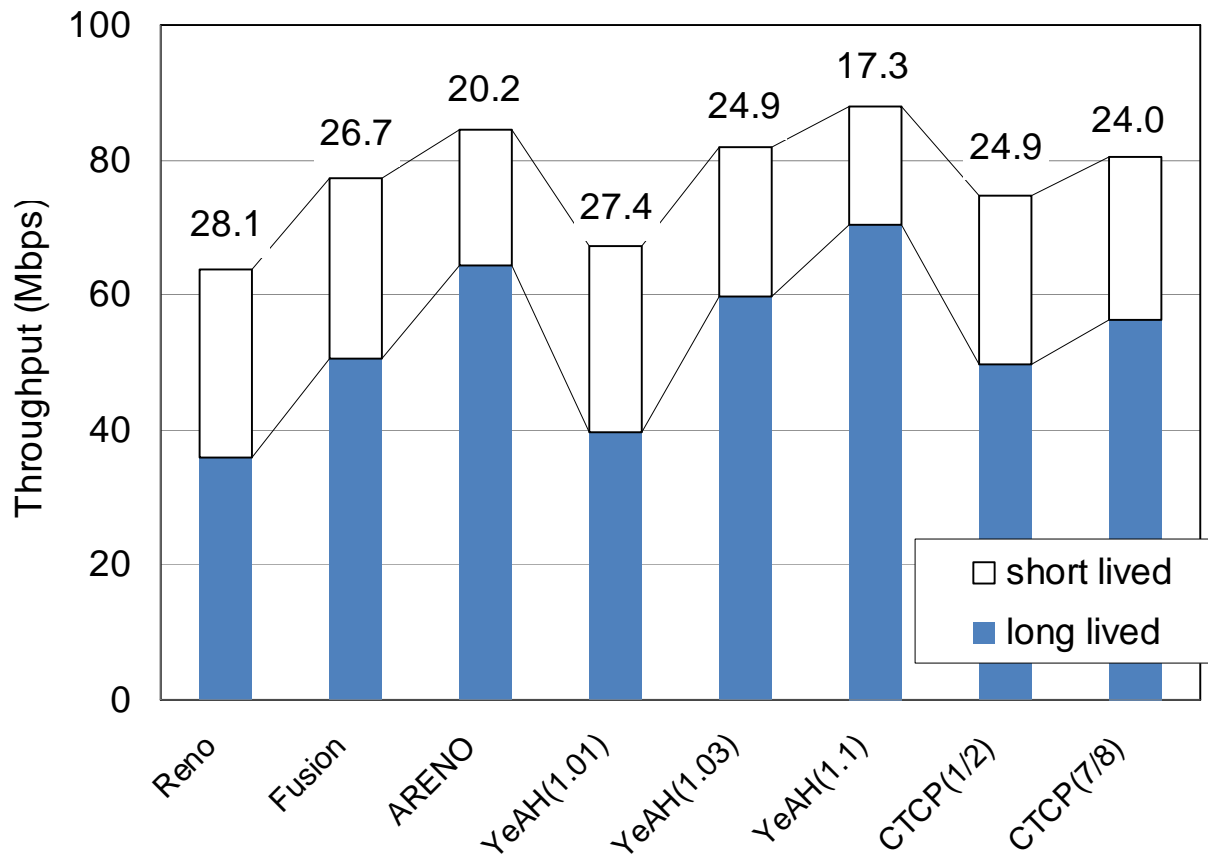


- CTCP:
  - window decrease:  $1/2(\text{original}) \sim 7/8$
- Yeah-TCP:
  - window increase:  $1.01(\text{original}) \sim 1.1$



# [ Evaluations (3) ]

## ■ Performance tuning



one long-lived flow  
100 short-lived flows

TCP-Fusion, YeAH(1.03) and CTCP(7/8) seem to give good balance between throughput and inter-protocol fairness.



# [ Response Function (1) ]

- Average number of transmitted packets per RTT round

$$W_{\text{Reno}} = \frac{1.22}{p^{0.5}}$$

$$W_{\text{HSTCP}} = \frac{0.12}{p^{0.835}}$$

$$W_{\text{STCP}} = \frac{0.08}{p}$$

TCP	CA round	(i) $W < w/2$	(ii) $w/2 \leq W < w$	(iii) $w \leq W$
Loss	transmitted packets	$\frac{3}{8}w^2$	$\frac{3}{8}w^2$	$\frac{3}{8}w^2$
	elapsed time	$\frac{1}{2}w \cdot RTT_{\min} + \frac{1}{8}(3\sqrt{4wW}) \cdot \frac{PS}{B}$	$\frac{1}{2}w \cdot RTT_{\min} + \frac{1}{2}(w) \cdot \frac{PS}{B}$	$\frac{1}{2}w \cdot RTT_{\min}$
Delay	transmitted packets	$\frac{1}{2}w \cdot W$	$\frac{1}{2}w \cdot W$	$\frac{1}{2}w \cdot W$
	elapsed time	$\frac{1}{2}w \cdot RTT_{\min}$	$\frac{1}{2}w \cdot RTT_{\min}$	$\frac{1}{2}w \cdot RTT_{\min}$
Hybrid	transmitted packets	$\frac{3}{8}w^2$	$\frac{1}{2}w \cdot W + \frac{1}{2}(w-W)^2$	$\frac{1}{2}w \cdot W$
	elapsed time	$\frac{1}{2}w \cdot RTT_{\min} + \frac{1}{8}(3\sqrt{4wW}) \cdot \frac{PS}{B}$	$\frac{1}{2}w \cdot RTT_{\min} + \frac{1}{2}(w) \cdot \frac{PS}{B}$	$\frac{1}{2}w \cdot RTT_{\min}$

$$W_{\text{Hybrid/Fusion}} =$$

$$\begin{cases} \sqrt{3/2p} \\ \sqrt{8/3p} + \sqrt{3p/8} \cdot W^2 - W \\ W \end{cases}$$

(i)

(ii)

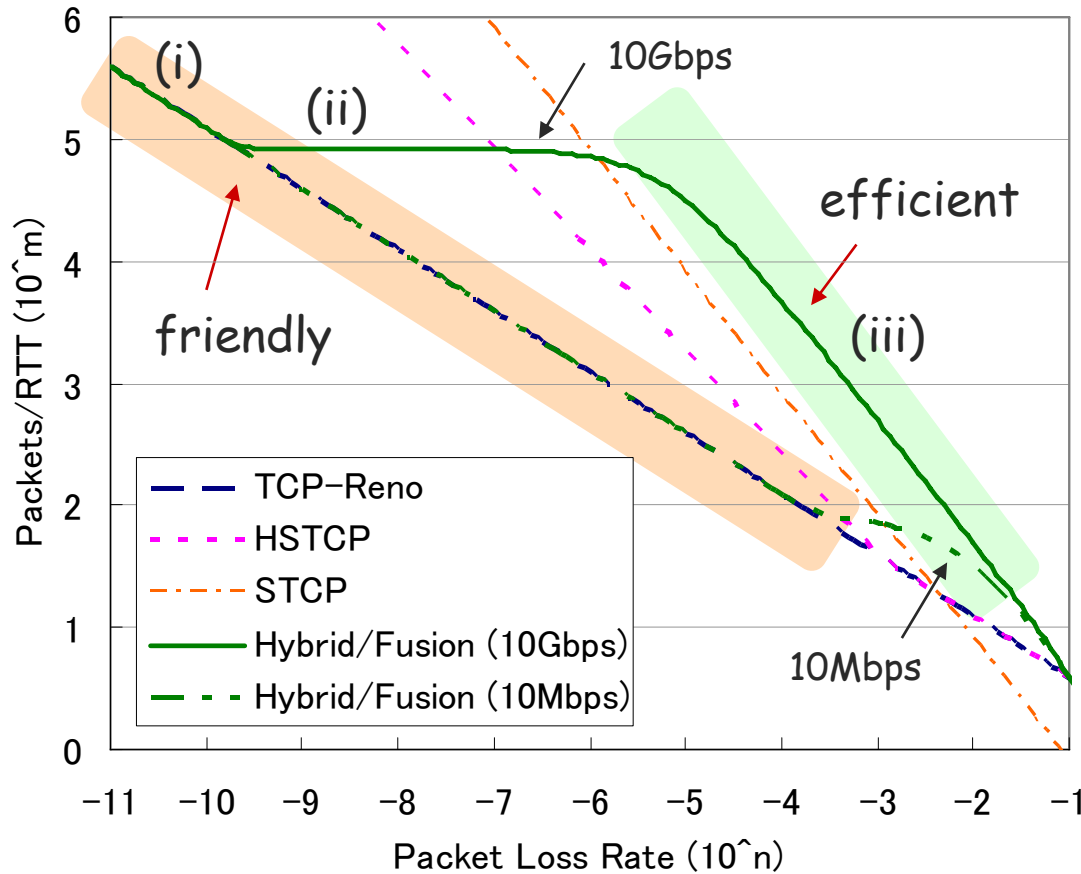
(iii)



*ignore buffering time*



# [ Response Function (2) ]



$$\begin{cases}
 \sqrt{3/2p} & (i) \\
 \sqrt{8/3p} + \sqrt{3p/8} \cdot W^2 - W & (ii) \\
 W & (iii)
 \end{cases}$$

$\frac{\text{transmitted packets}}{\text{elapsed time} + \text{timeout penalty}}$

Hybrid TCP becomes more efficient than loss-based ones as bandwidth ( $W$ ) increases (i.e. region (iii) moves to left).

- region (i) loss-mode
- region (ii) hybrid
- region (iii) delay-mode



[

]

# Conclusions



# [ Conclusions (1) ]

- Performance analysis of hybrid TCP
  - Ideal behavior models
    - Three models : loss-based, delay-based and hybrid
    - Single flow model and two flow (competing flow) model
    - Hybrids can perform well when PLR is high and/or router buffer is small (i.e. when vacant capacity exists)
  - Abstraction models of actual hybrids
    - Four hybrids: CTCP, ARENO, YeAH and Fusion
    - Slow window increase & constant window decrease result in throughput degradation
    - Parameter tuning of CTCP and YeAH-TCP
  - Response function of hybrid TCP
    - Comparison with Reno, HSTCP and STCP
  - Experiments
    - Verifications by analysis & simulations
    - For implementation results, see [18]



# [ Conclusions (2) ]

- Future work
  - Extension to buffered case (ICCCS 2008)
  - Extension to RTT fairness
  - Extension to multiple flows & reverse traffic
  - Extension to wireless (e.g. incorporation of layer 2 retransmission)
  - Extension to recent loss-based TCPs (e.g CUBIC and H-TCP)
  - Promotion of hybrid TCP by theories, implementations & applications
- URL (Linux/ns-2/MATLAB codes)
  - <http://www.katto.comm.waseda.ac.jp/TCP-Fusion/>
  - MATLAB code will be uploaded this weekend. Enjoy !



[

]



# [ Single Flow Model (4) ]

- Introducing timeout penalty

- Timeout penalty of TCP-Reno

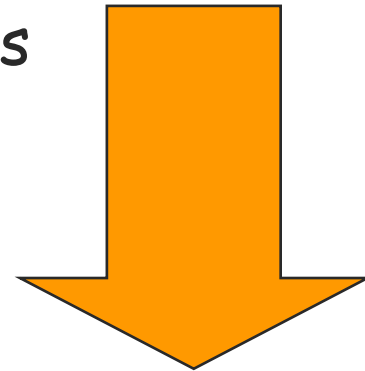
$$t_{RTO,loss} = T_0 \cdot (1 + p + 2p^2 + 4p^3 + 8p^4 + 16p^5 + 32p^6) / (1 - p)$$

- Ratio of transmitted packets

$$t_{RTO,delay} = \frac{K_{delay}}{K_{loss}} \cdot t_{RTO,loss}$$

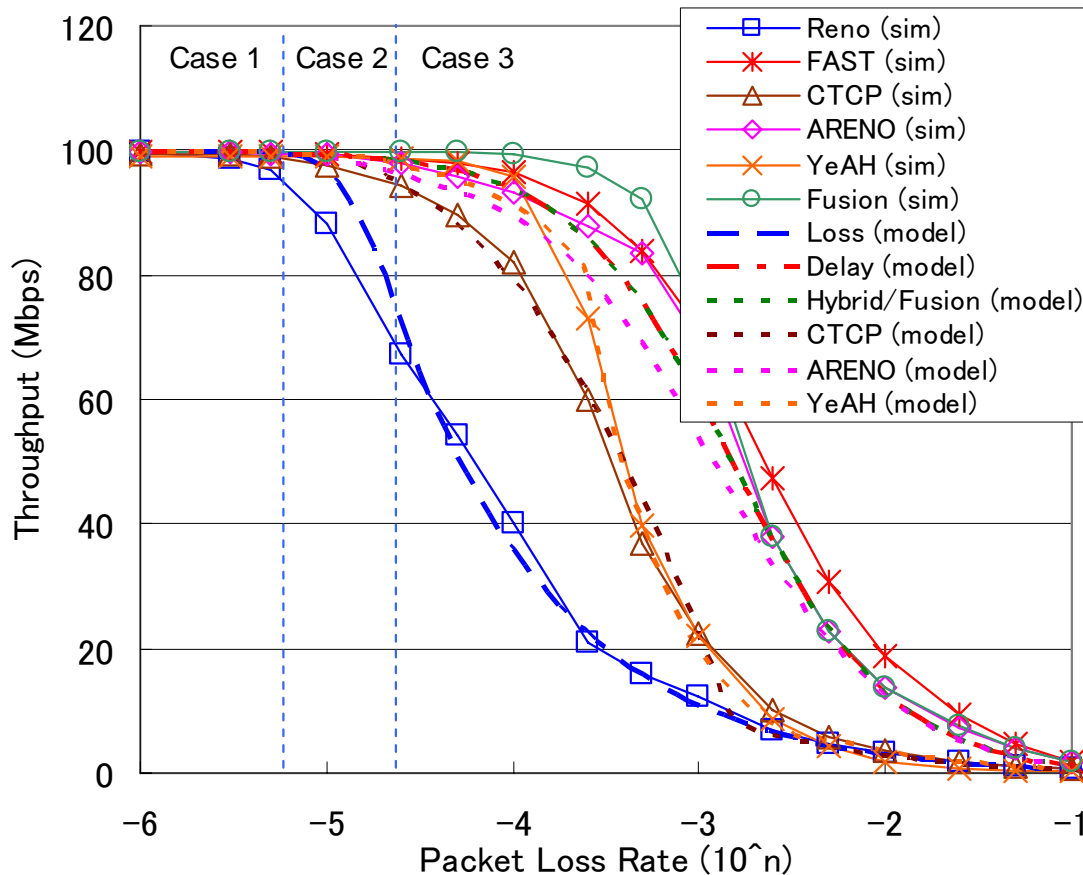
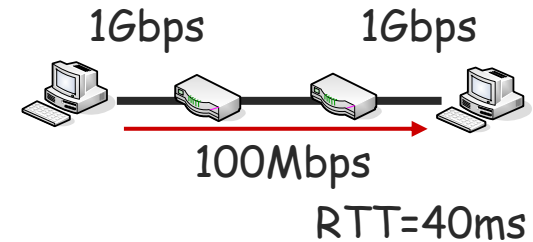
- Estimated throughput

$$\frac{\text{transmitted packets}}{\text{elapsed time} + \text{timeout penalty}}$$



# Evaluations (1) 100Mbps

## Single flow model



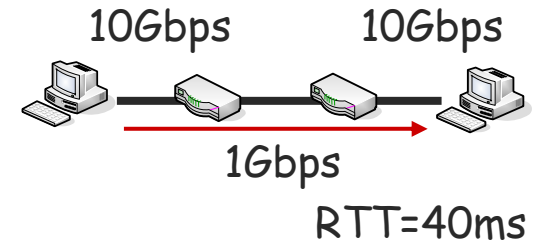
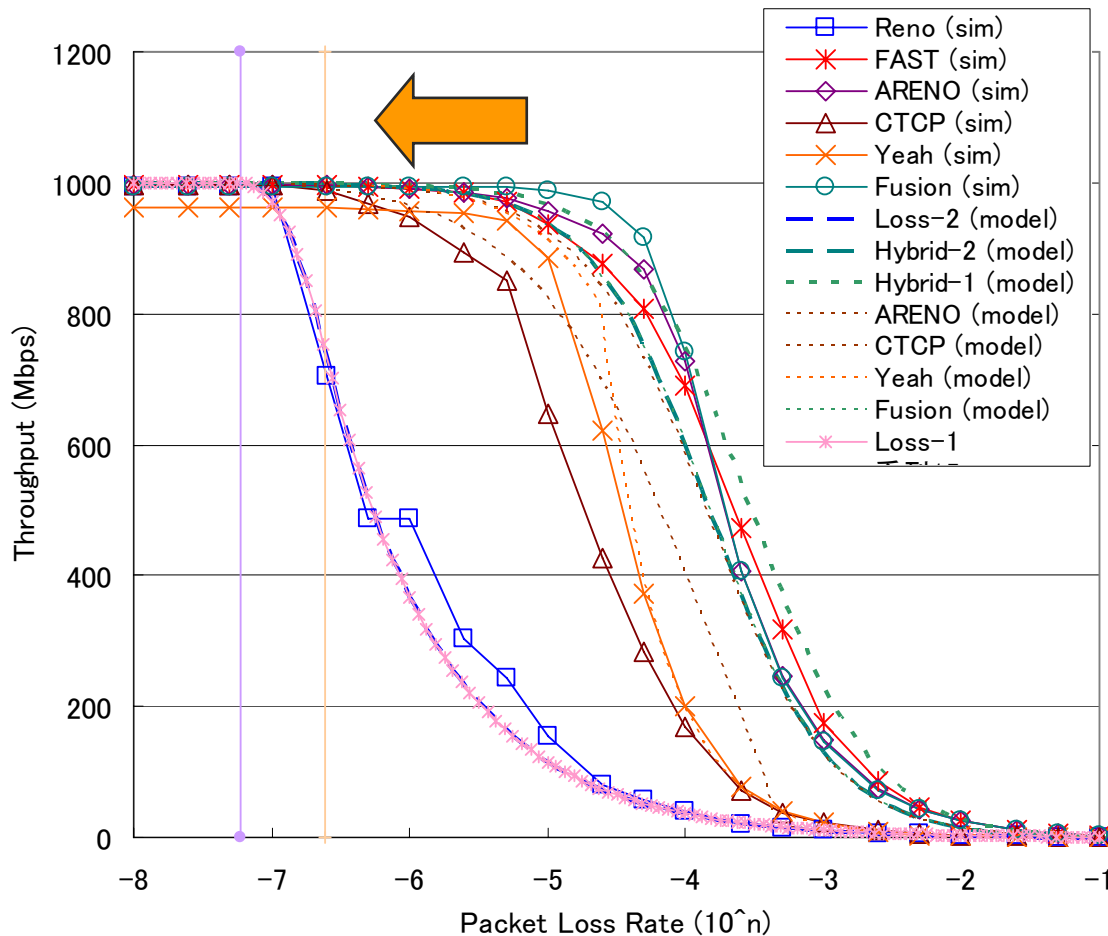
buffer size = BDP (constant)  
PLR : variable

When PLR is large (regions (ii) & (iii)), delay based and hybrid TCPs drastically outperform loss based TCP.

Performances of CTCP and YeAH are worse than the others.

# Evaluations (1') 1Gbps

## Single flow model



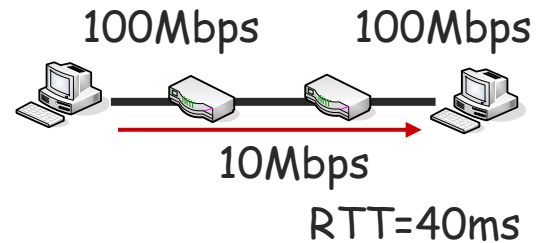
buffer size = BDP (constant)  
 PLR : variable

When PLR is large (regions (ii) & (iii)), delay based and hybrid TCPs drastically outperform loss based TCP.

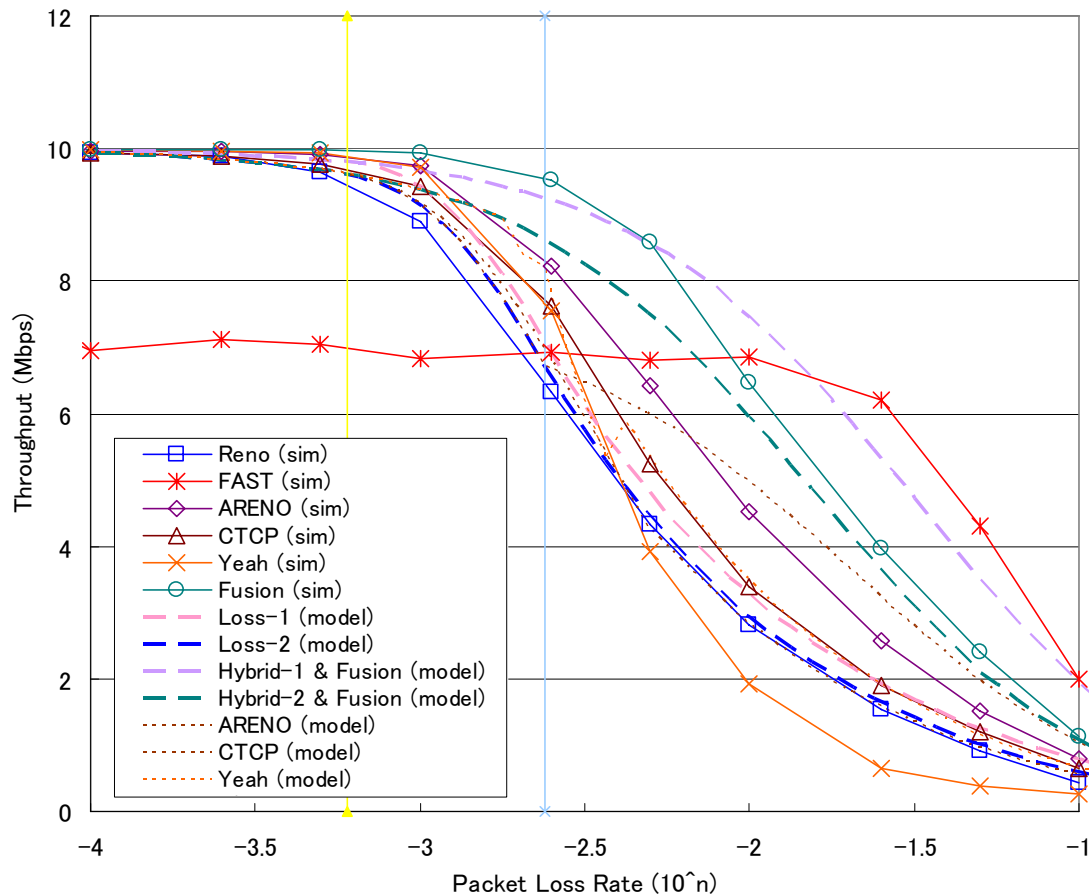
As bandwidth increases from 100Mbps to 1Gbps, regions (ii) & (iii) move to left.

# Evaluations (1") 10Mbps

## Single flow model

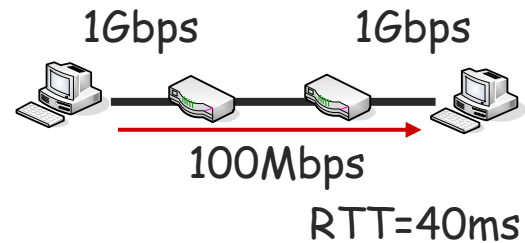
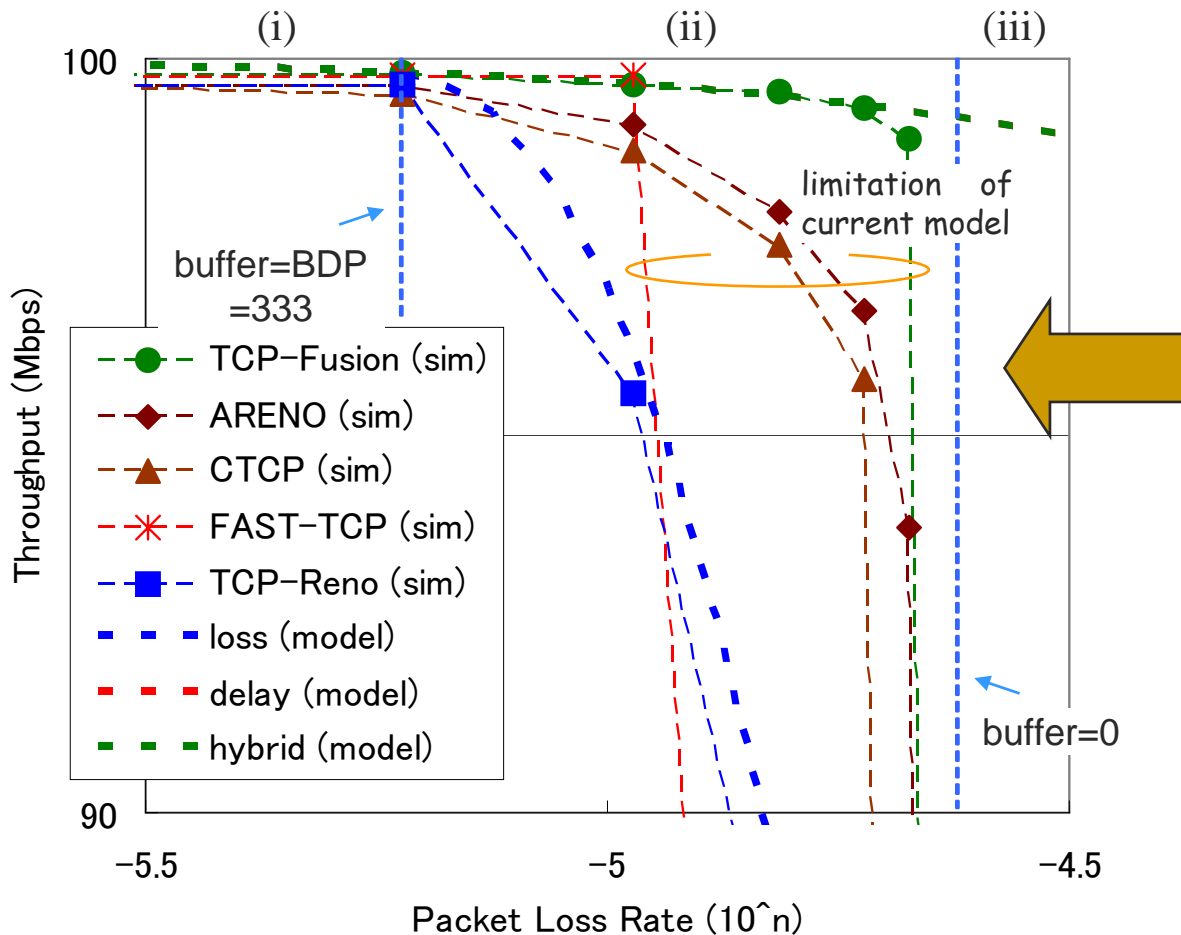


buffer size = BDP (constant)  
PLR : variable

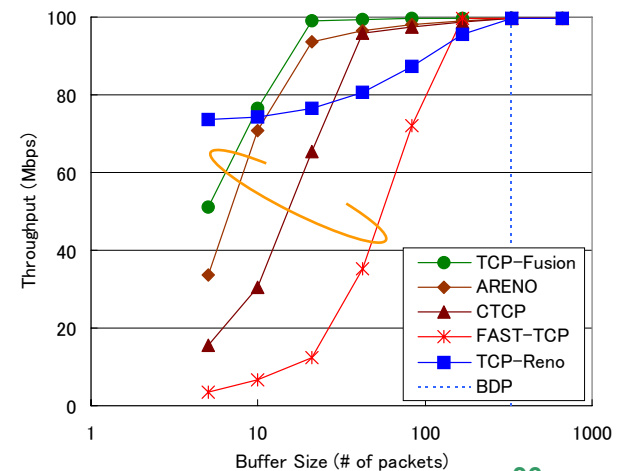


# [ Single Flow Model (6) ]

## ■ Result (2) buffer size

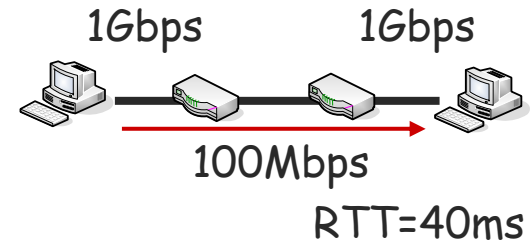


PLR =  $10^{-6}$  (constant)  
buffer size : variable



# Evaluations (1") YeAH-TCP

## Single flow model



buffer size = BDP (constant)  
 PLR : variable

