Implementation Experiments of TCP proxy Mechanism

Osaka University
Kana YAMANEGI

Outline

- Background
  - TCP Overlay Network
  - TCP proxy Mechanism
- Experimental Environment
- Experiments using the Public Network
  - Effect of TCP proxy
  - Effect of High-Speed TCP
- Conclusion and Future Work

Background

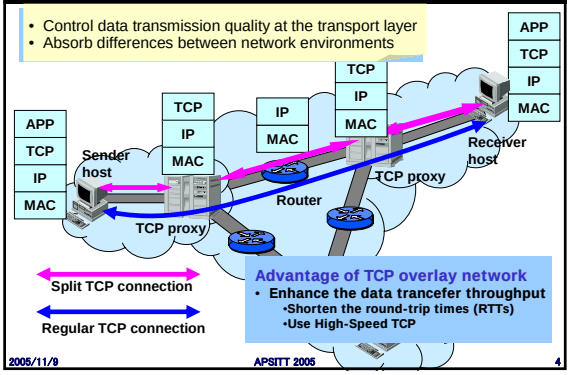
- Development of the access/backbone network technologies
- Rapid growth of the Internet population

→User demands diversified and sophisticated services

  - But can't be assured in the current Internet because of its best-effort basis

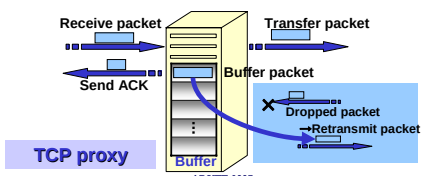
TCP Overlay Network

- Control data transmission quality at the transport layer
- Absorb differences between network environments



TCP proxy Mechanism

- TCP proxy
  - Relay and transfer the TCP session
  - Send back pseudo ACK packet
  - Buffer the received packet
    - →Can retransmit packets from TCP proxy when packet dropped
- Improve the data transfer throughput of connections by shortening the round-trip time
- Other TCP protocol can be used between TCP proxies



Objectives

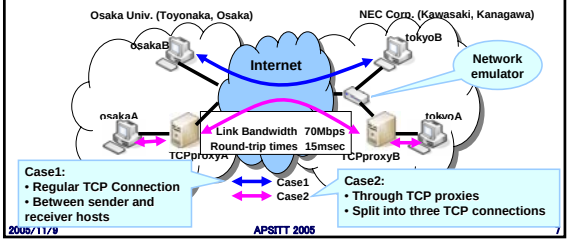
We have logically confirmed the effect of the TCP proxy mechanism [4], but not reveal yet in the actual network

- Implementation Evaluation of the TCP proxy mechanism in the public network
  - Use the public Internet network between Osaka and Tokyo
  - Verify effectiveness in the actual network
- Evaluation of the performance using High-Speed TCP between TCP proxies
  - HSTCP (High-Speed TCP)
  - gHSTCP (gentle HSTCP)

[4] I. Maki, G. Hasegawa, M. Murata, and T. Murase, "Performance analysis and improvement of TCP proxy mechanism in TCP overlay networks," in *Proceedings of IEEE International Conference on Communications, Wireless Networking (ICC 2005)*, May 2005.

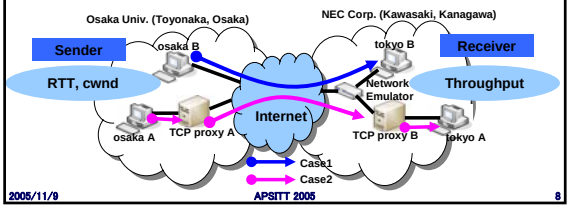
Experimental Environment

- 4 - receiver and sender hosts
- 2 - TCP proxies
- Network emulator
  - Osaka-Tokyo case: not used
  - Okinawa-Tokyo case: add 25-msec delay

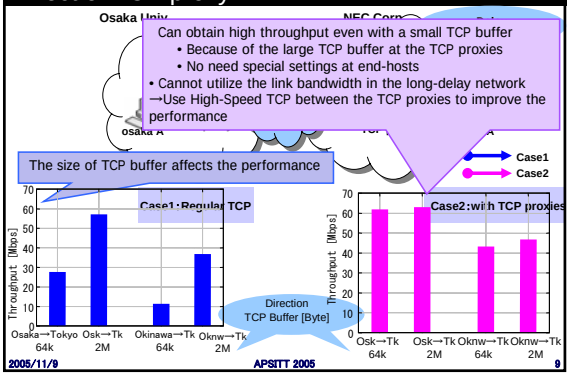


Experimental Settings

- Using the measurement tool *iperf* for injecting TCP traffic
- Protocol at end-hosts: TCP Reno
- TCP socket buffer: 64 KBytes (default) / 2 MBytes
- Experimental direction: Osaka Univ. → NEC Corp.
- Evaluation: Throughput
  - The amount of data arriving at the receiver host per unit time



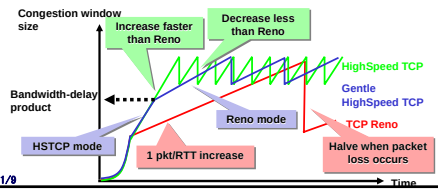
Effect of TCP proxy



Research Background: High-Speed TCP

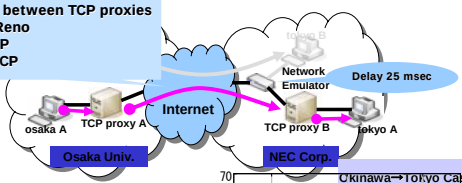
cwnd control

	AI [pkts/RTT]	MD	Feature・Problem
TCP Reno	1	0.5	▪ Be common in the current Internet ▪ Cannot utilize the link bandwidth
HSTCP (High-Speed TCP)	a(w)	b(w)	▪ Obtain high throughput ▪ Occupy the link bandwidth
gHSTCP (gentle High-Speed TCP)	congestion: 1 otherwise: a(w)	b(w)	▪ Observes the RTT to find congestion ▪ Be fair to co-existing TCP Reno connection

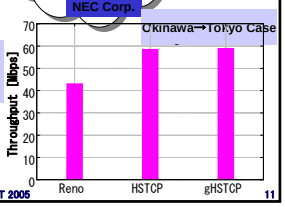


Effect of High-Speed TCP

Protocols between TCP proxies
- TCP Reno
- HSTCP
- gHSTCP



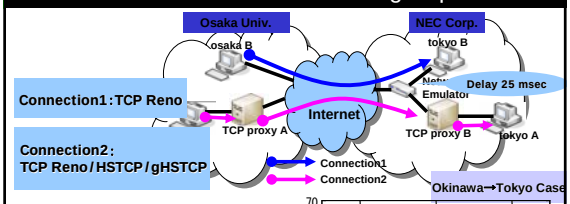
- Enhance the data transfer throughput by using HSTCP or gHSTCP
- No need special setting at the end-hosts



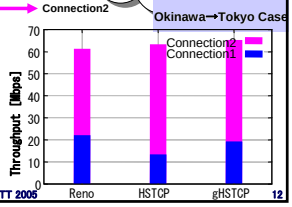
Fairness between TCP Reno and High-Speed TCP

Connection1: TCP Reno

Connection2: TCP Reno / HSTCP / gHSTCP



- HSTCP
  - Make co-existing (TCP Reno) connection's throughput decrease
- gHSTCP
  - Obtain high throughput while not affecting the performance of co-existing connections



Conclusion

- TCP proxy mechanism is verified making the data transfer throughput increase in the actual network
  - Equivalent to the results by simulation
  - No need the special settings of the end hosts
  - Enhance the performance by using High-Speed TCP between the TCP proxies
- Future work:
  - Evaluation in a larger experimental network using more than three networks
  - Evaluation in a high-speed network