# Transport Protocols for the Internet Interactive Applications

Jae-young Lee

Aug 12, 2009

# Roadmap

- Introduction
- Related Work
- Existing Protocols (TCP and UDP)
- Interactive Application Protocols
- OPNET Simulation Scenario
- OPNET Simulation Results
- Discussion
- Conclusion/Future Work

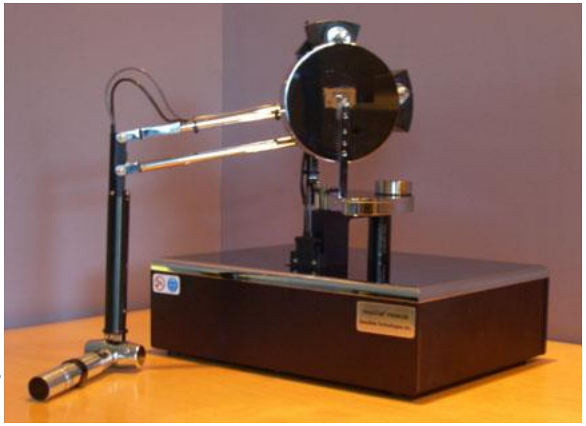
# Introduction: overview (1/2)

- Interactive applications:
  - Internet applications that send/receive real-time information between end systems
  - Teleoperation, internet game, internet telephony, real-time audio, and video applications
- Teleoperation:
  - One of the interactive applications that sends motion data and receives reflecting force data in a distance

Human operator

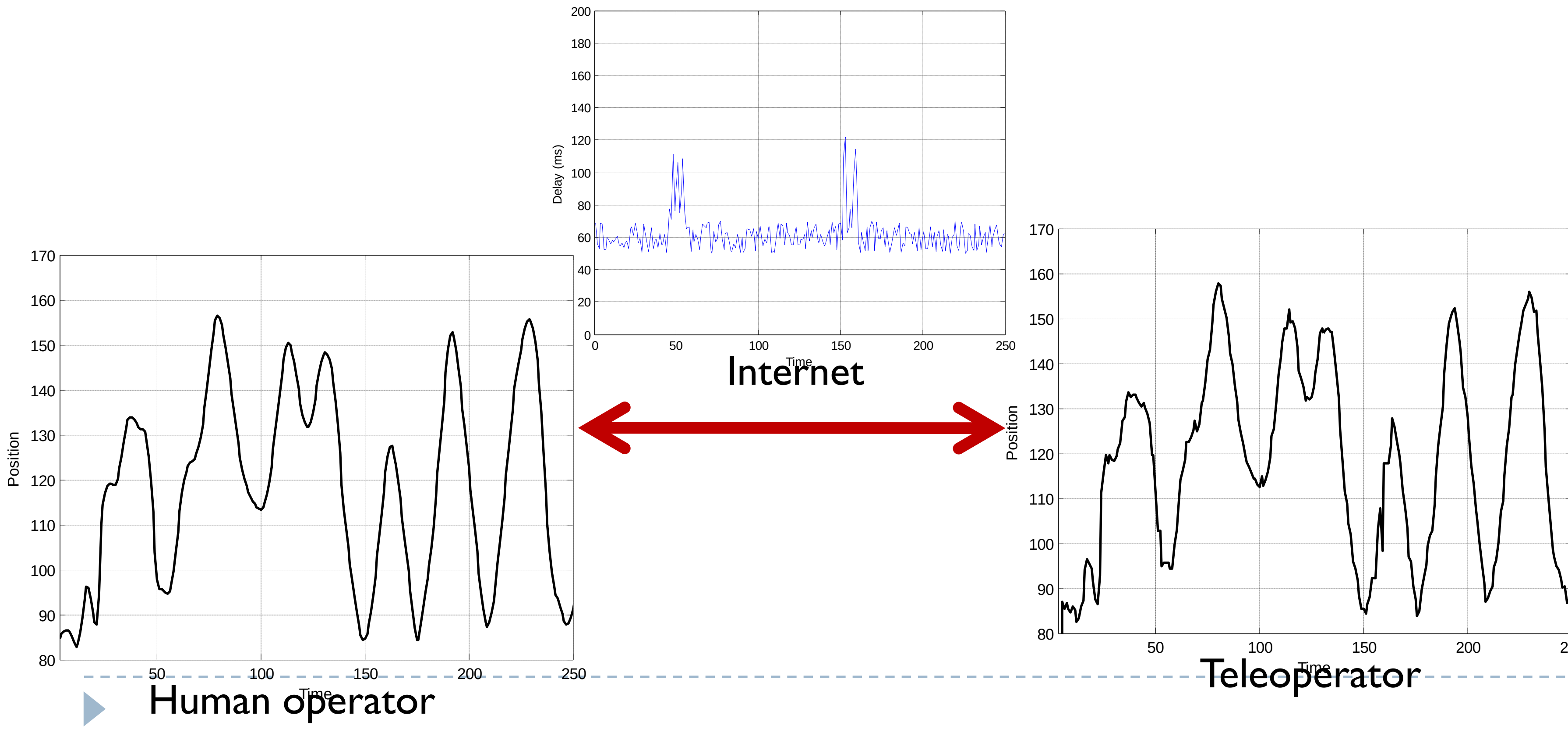
Internet

Teleoperator



# Introduction: motivation (2/2)

- Issues in interactive applications:
  - Time delay – unknown and variable time delay may impair the real-time operation
  - Data loss – open issue



# Related Work (1/2)

- Control system approaches [2], [3]:
  - Based on wave-variable transform for stable operation between two end systems
- Signal processing approaches [4], [5]:
  - Based on prediction/estimation methods to obtain original data

➡ Many other approaches are also done by employing control system and signal processing approaches


- [2] G. Niemeyer and J. Slotine, "Designing force reflecting teleoperators with large time delays to appear as virtual tools," Proc. IEEE International Conference on Robotics and Automation, Albuquerque, NM, Apr. 1997, pp. 2212–2218.
- [3] K. Kawashima, K. Tadano, G, Sankaranarayana, and B. Hannaford, "Bilateral teleoperation with time delay using modified wave variables," Proc. IEEE International Conference on Intelligent Robots and Systems, Sept. 2008.
- [4] S. Clarke, G. Schillhuber, M. Zach, and H. Ulbrich, "The effects of simulated inertia and force prediction on delayed telepresence," Presence, vol. 16, no. 5, pp. 543-558, Oct. 2007.
- [5] S. Munir and W. Book, "Internet-based teleoperation using wave variables with prediction," IEEE Transactions on Mechatronics, vol. 7, no 2, June 2002.

# Related Work (2/2)

- Transport protocol approaches [6]-[9]:
  - Only few approaches have been proposed
  - Modifications to existing protocols: TCP and UDP
    - Real time protocol (RTP), Interactive real-time protocol (IRTP), Real-time network protocol (RTNP), Efficient transport protocol (ETP)


- [6] H. Schulzrinne, S. Deering, R. Frederick, and V. Jacobson, "RTP: a transport protocol for real-time applications," RFC 3550, July 2003.
- [7] L. Ping, "Transport layer protocol reconfiguration for network-based robot control system," Proceedings of IEEE International Conference on Networking, Sensing and Control, Tucson, AZ, Mar. 2005, pp. 1049-1053.
- [8] Y. Uchimura, "Bilateral robot system on the real-time network structure," IEEE Transactions on Industrial Electronics, vol. 51, no. 5, pp. 940-946, Oct. 2004.
- [9] R. Wirz, "Efficient transport protocol for networked haptics applications," Proceedings of the 6th International Conference on Haptics: Perception, Devices and Scenarios, Madrid, Spain, June 2008, vol. 5024, pp. 3-12.

# Existing Protocols

- TCP
  - Reliable and connection-oriented
  - e-mail, web, remote terminal access, and file transfer
  - Relatively large variation of time delay
  - For interactive application, it can be used for delivery of crucial information
- UDP
  - Unreliable and conectionless
  - Streaming multimedia and voice over IP
  - Small variation of time delay
  - For interactive application, it can be used for delivery of real-time data that is loss-tolerant

# Interactive application protocols (1/3)

- RTP (real-time protocol) [6]:
  - Designed for multimedia services
  - Use an intermediate buffer, that may lead large overall delay
  - Not appropriate for teleoperation
- RTNP (real-time network protocol) [7]:
  - Time delay depends not only on network, but also on operating system
  - Implemented based on UNIX environment
  - Limitation – not available to other environment (Windows)


- [6] H. Schulzrinne, S. Deering, R. Frederick, and V. Jacobson, "RTP: a transport protocol for real-time applications," RFC 3550, July 2003.
- [7] L. Ping, "Transport layer protocol reconfiguration for network-based robot control system," Proc. IEEE International Conference on Networking, Sensing and Control, Tucson, AZ, Mar. 2005, pp. 1049-1053.

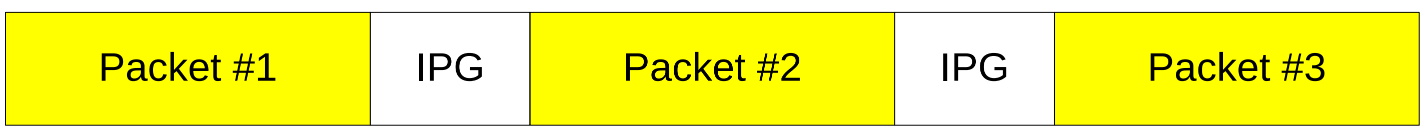
# Interactive application protocols (2/3)

- IRTP (interactive real-time protocol) [8]:
  - Assign priority in packets of real-time data
  - Take advantages of both TCP and UDP
    - TCP - crucial data transmission
    - UDP - real-time data transmission
- ETP (efficient transport protocol) [9]:
  - Based on inter-packet gap (IPG) insertion between packets
  - Congested network can be manages by IPG control
  - IPG control can be applied to both TCP and UDP


- [8] Y. Uchimura, "Bilateral robot system on the real-time network structure," IEEE Transactions on Industrial Electronics, vol. 5 1, no.5, pp. 940-946, Oct. 2004.
- [9] R. Wirz, "Efficient transport protocol for networked haptics applications," Proc. the 6th International Conference on Haptics: Perception, Devices and Scenarios, Madrid, Spain, June 2008, vol. 5024, pp. 3-12.

# Interactive application protocols (3/3)

- IPG (inter-packet gap):
  - Time delay between two successive packets
  - If IPG is increased, then data rate is reduced
    - If IPG control is used with UDP, it manages network congestion like TCP does with window size




IPG: Inter-packet Gap

IPG insertion between packets

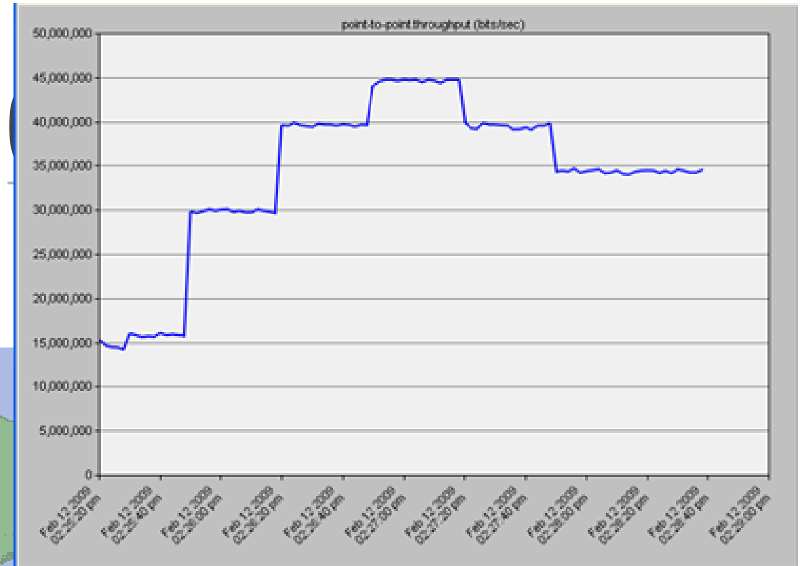
# Simulation Scenario (1/5)

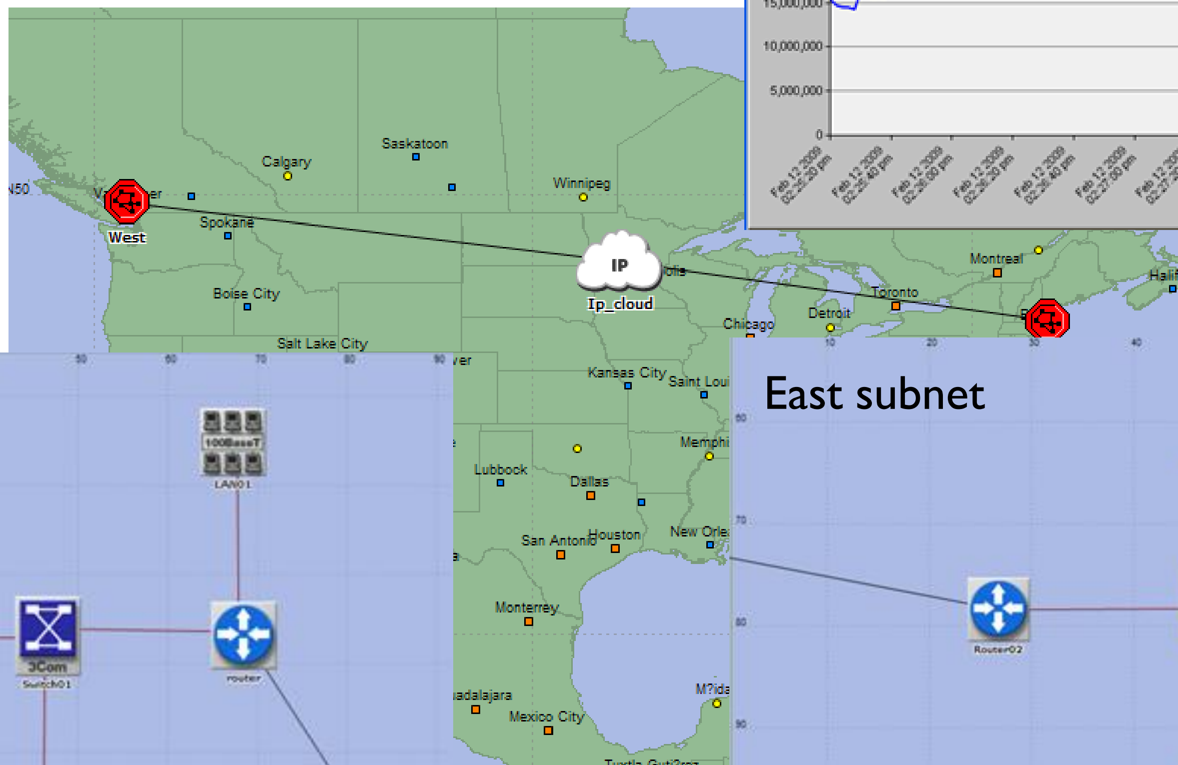
- Simulation tool: OPNET Modeler v. 14.5
- Simulation design:
  - WAN topology is designed with West and East subnets
    - Human operator: located in West subnet
    - Teleoperator: located in East subnet
  - Two subnets are connected with IP clouds
    - Packet discard ratio: 1%
    - Packet latency: 1ms ~ 100ms
  - Each subnet contains LANs with star topology and servers
  - Background load is included

# Simulation Scenario (

- Simulation scenario

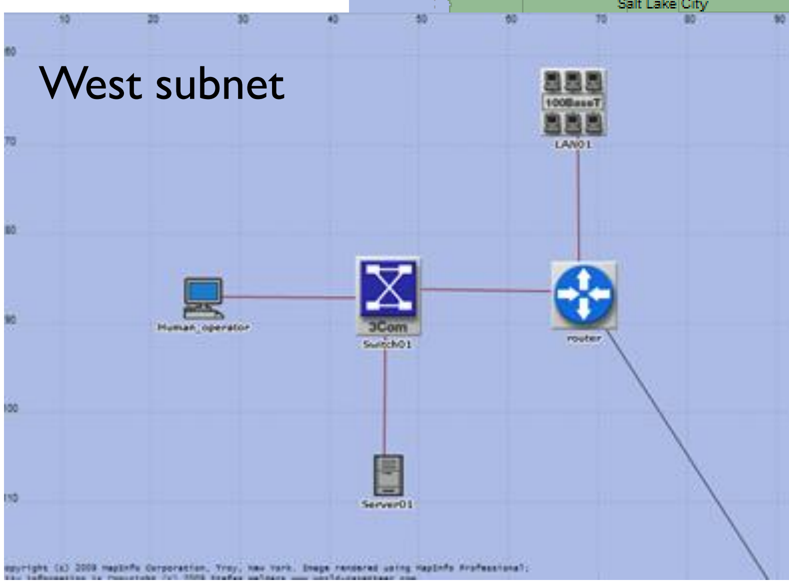
Backload



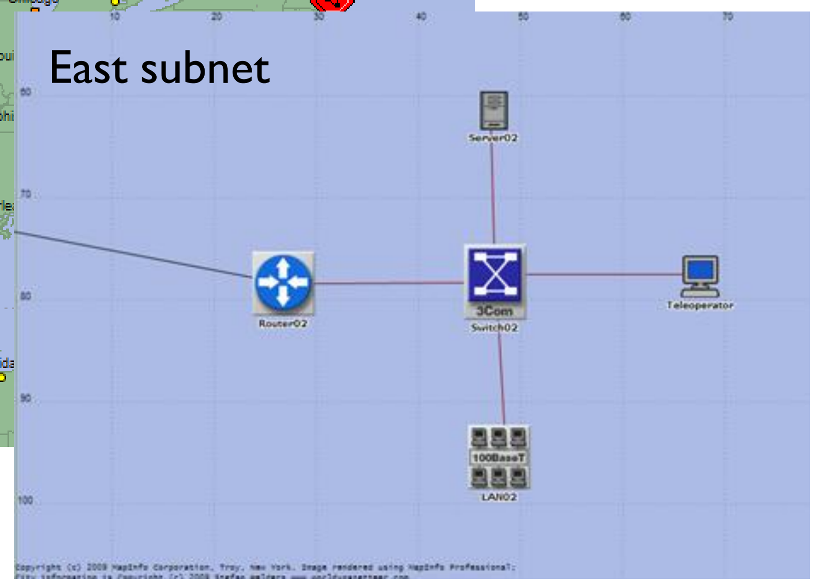




AN topology

West subnet




East subnet



# Simulation Scenario (3/5)

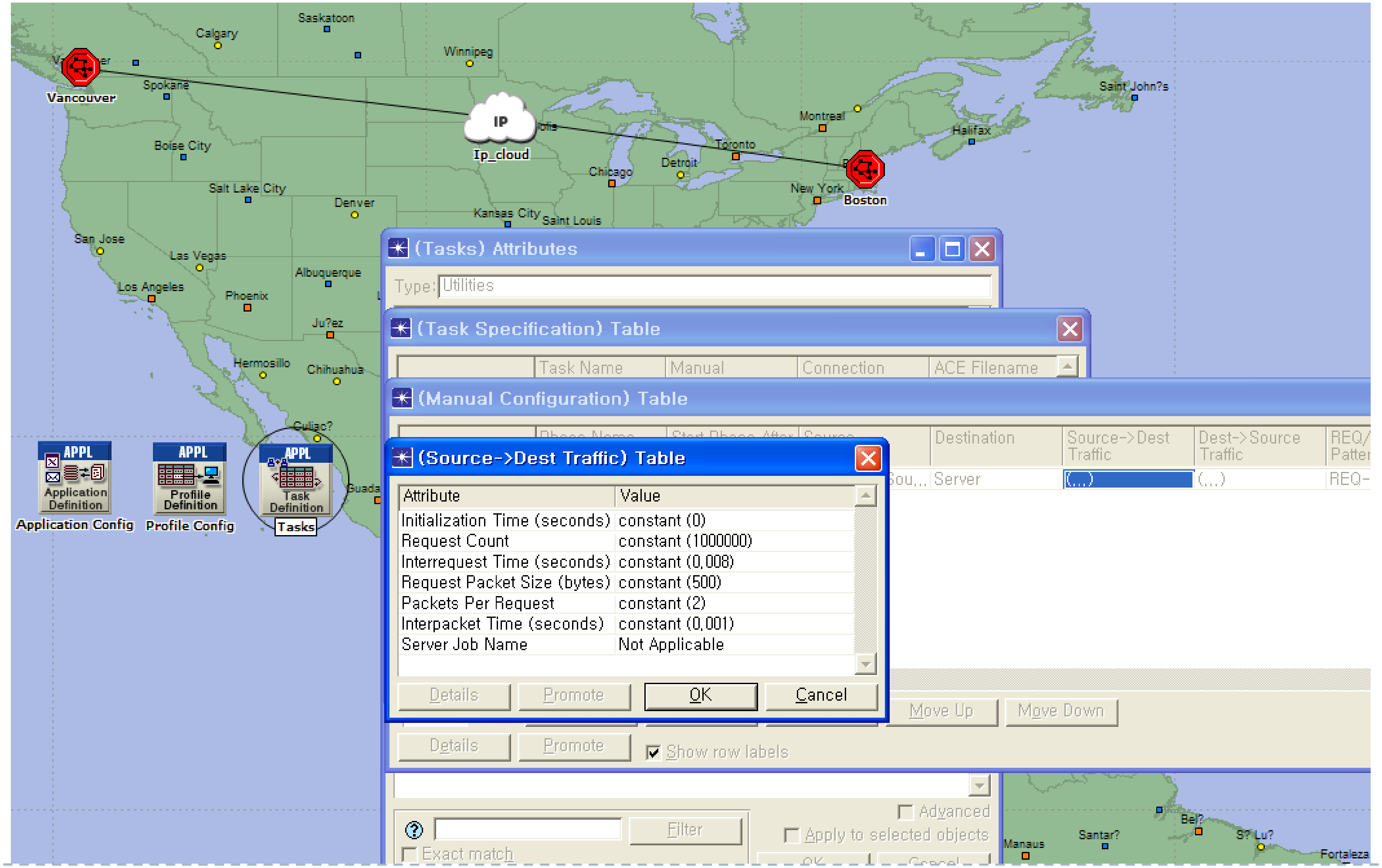
- Implementation details:
  - TCP and UDP:
    - Use "Task Configuration"
    - 1 Mbps data rate: Human operator ⟷ Teleoperator
      - Packet size: 500 bytes
      - Inter-request time: 40 ms
      - Packets per request: 10
      - 500 (bytes) * 1/0.04 * 10 * 8 = 1 Mbps
    - Transport protocols can be selected by using "Application Configuration"
      - TCP Reno version
      - Standard UDP

# Simulation Scenario (4/5)

- Implementation details:
  - ETP (Efficient transport protocol):
    - IPG (inter-packet gap) is inserted using "Task Configuration"
    - Data rate is reduced depending on IPG
      - Packet size: 500 bytes
      - Inter-request time: 8 ms
      - Packets per request: 2
      - IPG: 1 msec
      - 500 (bytes) * 1/(0.008+0.001) * 2 * 8 = 0.9 Mbps
    - IPGs are inserted in both TCP and UDP cases

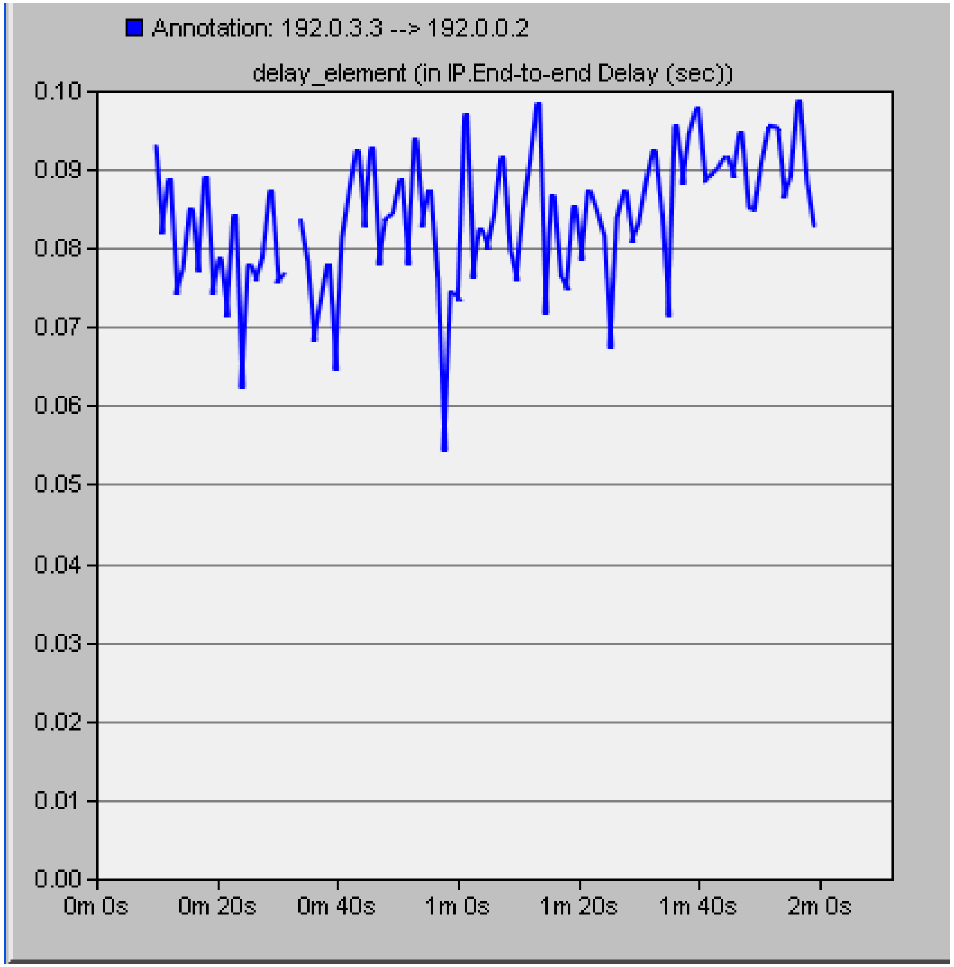
# Simulation Scenario (5/5)

- OPNET project view



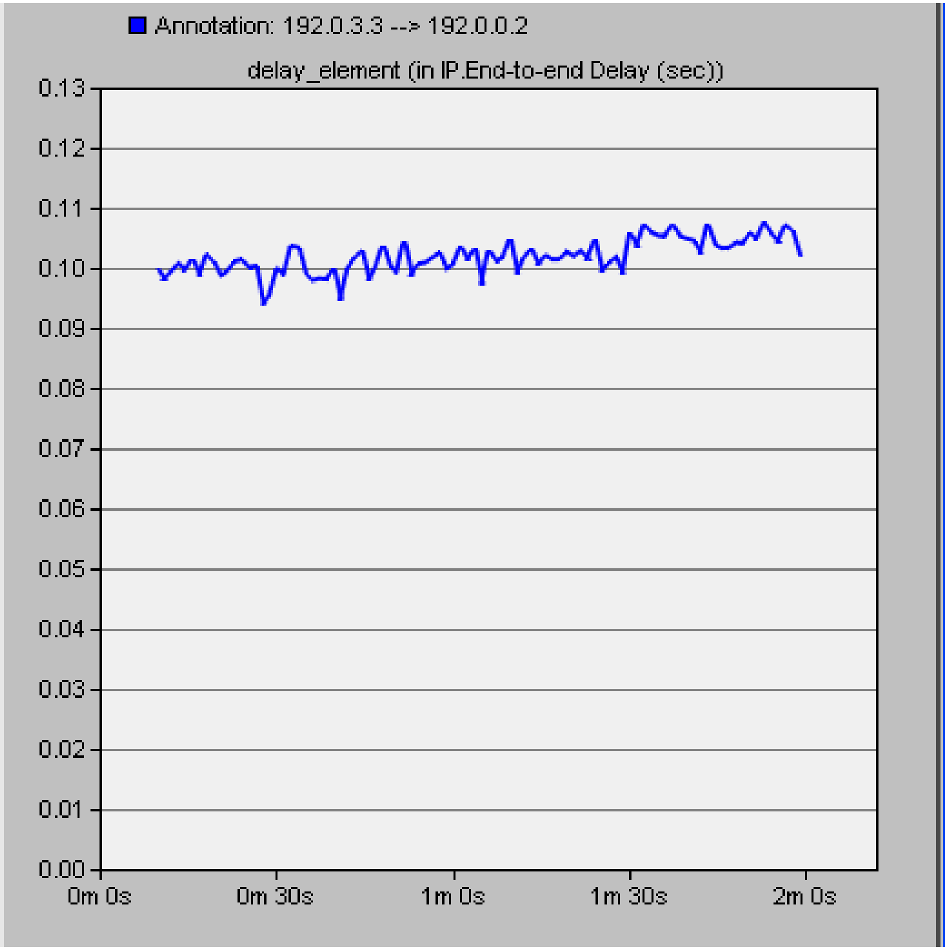
# Simulation Results (1/5)

- TCP (Reno)
  - End-to-end delay between Human operator and Teleoperator
    - Avg: 83.3 ms
    - Min: 54.4 ms
    - Max: 98.9 ms
    - Std. dev: 8.4
  - Variation of end-to-end delay is relatively large



# Simulation Results (2/5)

- UDP
  - End-to-end delay
    - Avg: 101.9 ms
    - Min: 93.9 ms
    - Max: 107.6 ms
    - Std. dev: 2.8
  - Compared with TCP, variation of end-to-end delay is small



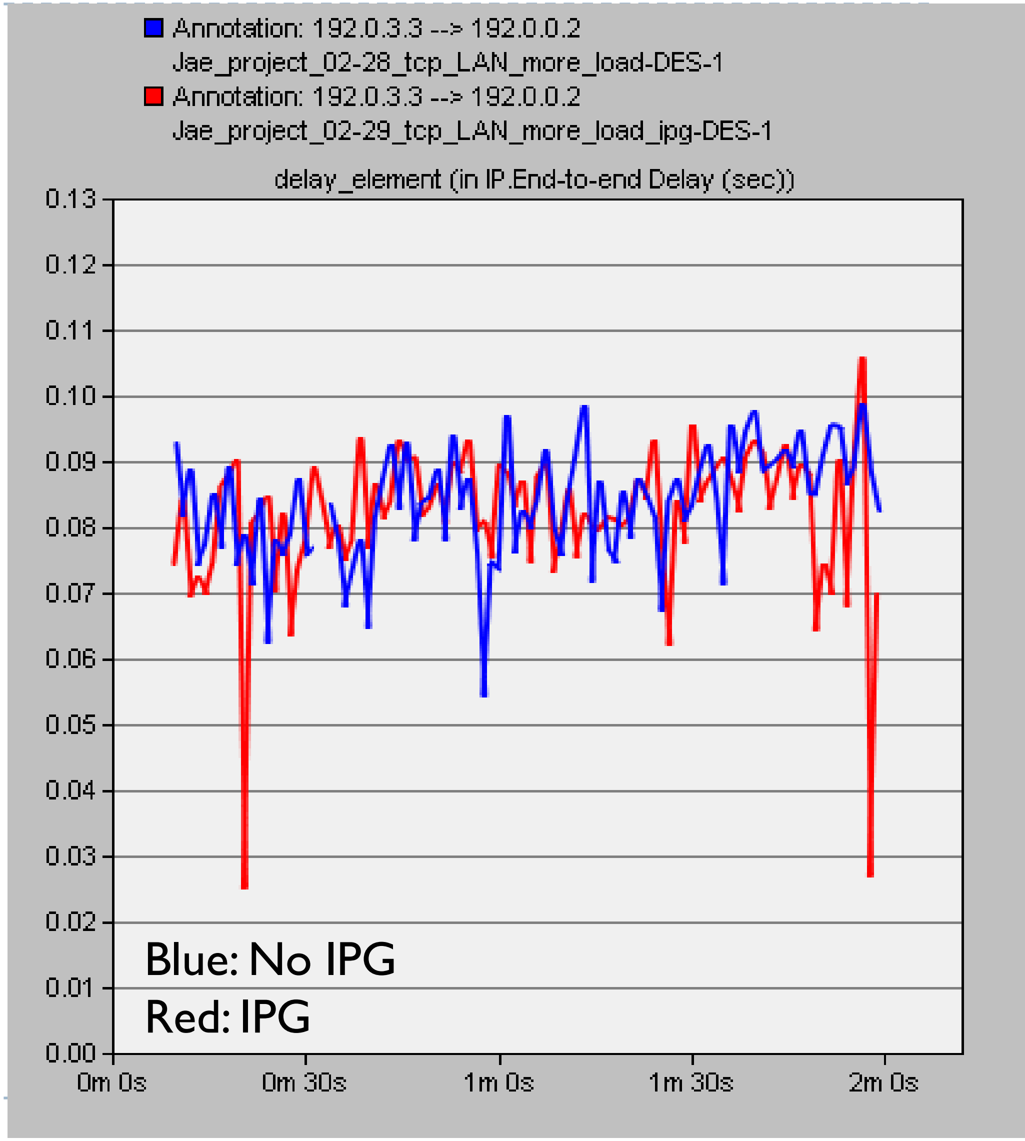
# Simulation Results (3/5)

- IPG in TCP
  - IPG: 4ms
  - End-to-end delay

| | No IPG (ms) | IPG (ms) |
|---|---|---|
| Avg. | 83.3 | 81.4 |
| Min. | 54.4 | 25.3 |
| Max. | 98.9 | 105.9 |
| Std. dev | 8.4 | 11.3 |

  - IPG does not improve end-to-end delay performance in TCP

* IPG: inter-packet gap

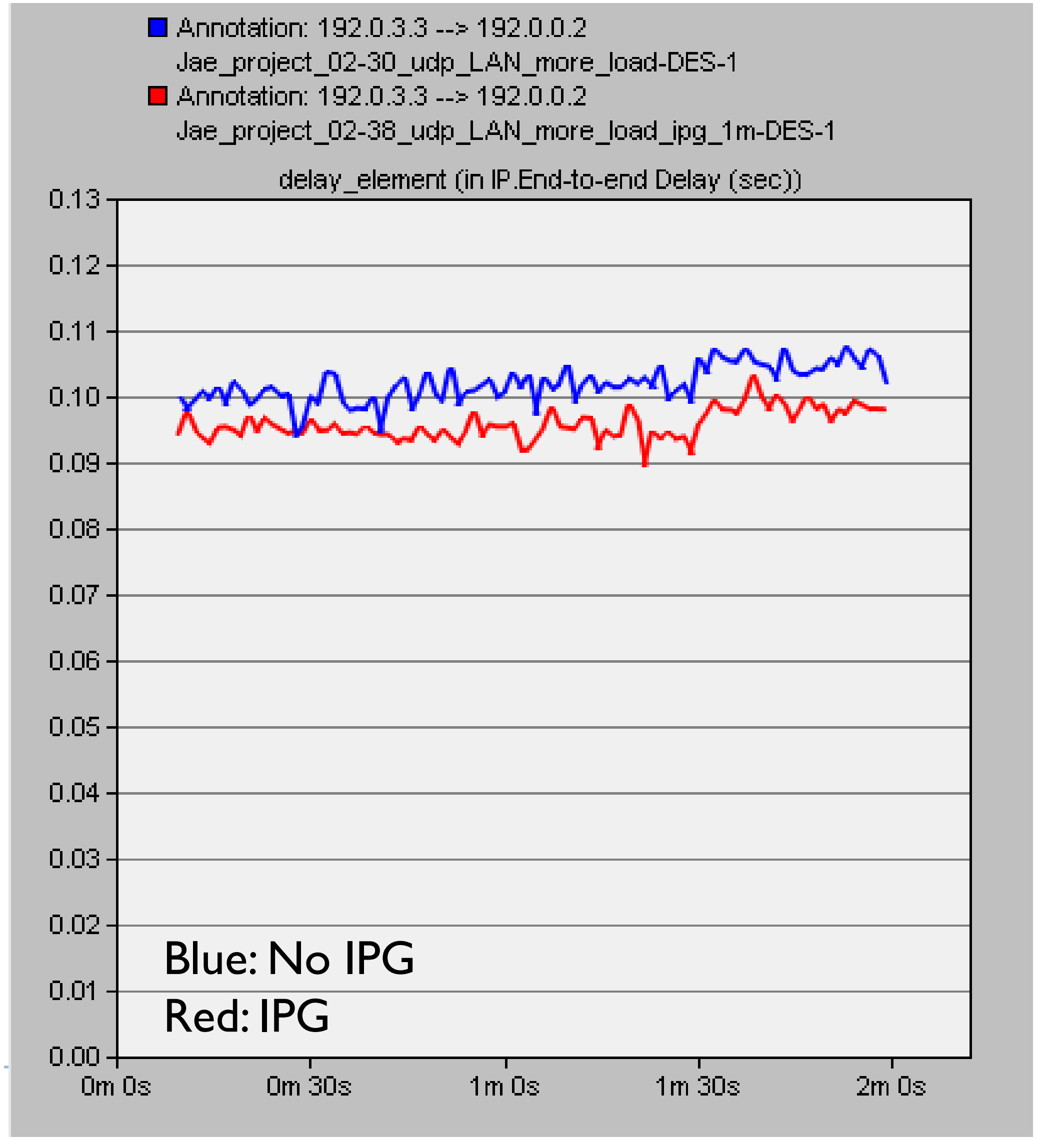


# Simulation Results (4/5)

- IPG in UDP
  - IPG: 1ms
  - End-to-end delay

| | No IPG (ms) | IPG (ms) |
|---|---|---|
| Avg. | 101.9 | 95.7 |
| Min. | 93.9 | 89.5 |
| Max. | 107.6 | 103.2 |
| Std. dev | 2.8 | 2.3 |

  - IPG improves end-to-end delay performance in UDP



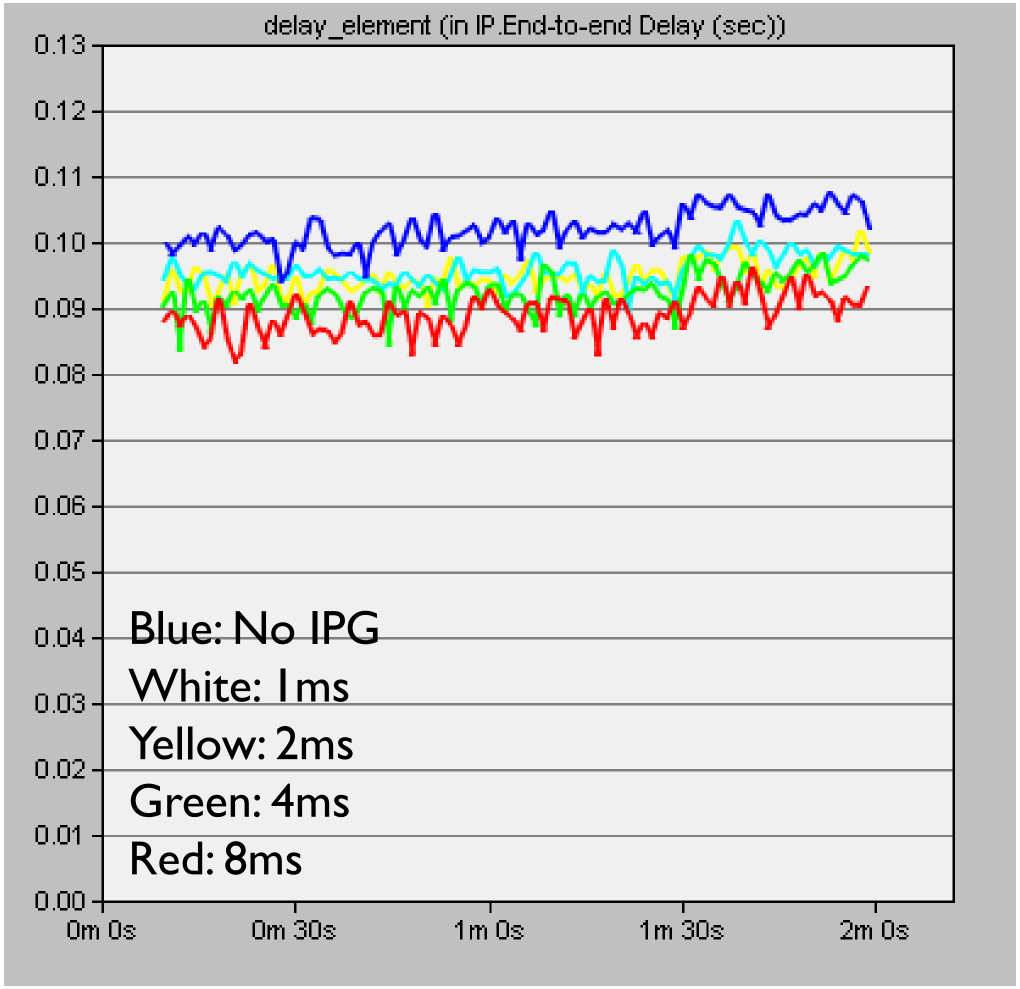

* IPG: inter-packet gap

# Simulation Results (5/5)

- More IPG in UDP
  - IPG: 1~ 8 ms
  - End-to-end delay

| | No IPG | IPG 1 ms | IPG 2 ms | IPG 4 ms | IPG 8 ms |
|---|---|---|---|---|---|
| Avg. | 101.9 | 95.7 | 94.6 | 92.3 | 88.9 |
| Min. | 93.9 | 89.5 | 89.9 | 83.6 | 81.9 |
| Max. | 107.6 | 103.2 | 101.7 | 98.4 | 96.2 |
| Std. dev | 2.8 | 2.3 | 2.3 | 2.9 | 3.0 |

  - ETE delay is reduced as IPG increases



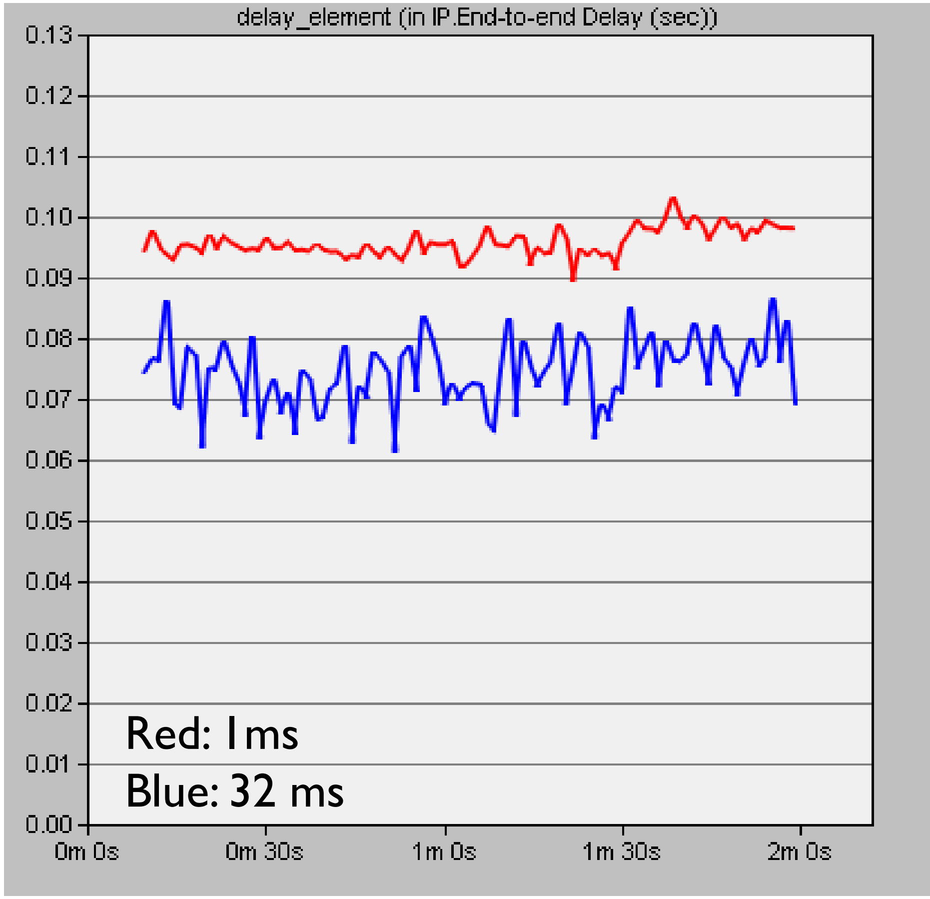

* IPG: inter-packet gap

# Discussion (1/2)

- IPG cannot be too large:
  - Increasing IPG gives larger variation of end-to-end delay

| | 1ms IPG | 32 ms IPG |
|---|---|---|
| Avg. | 95.7 | 74.2 |
| Min. | 89.5 | 61.6 |
| Max. | 103.2 | 86.8 |
| Std. dev | 2.3 | 5.6 |

  - Optimal value of IPG needs to be selected to avoid large variation




* IPG: inter-packet gap

# Discussion (2/2)

- IPG cannot be too large:
  - In teleoperation, haptic data must maintain certain frequency to avoid discontinuity
    - Motion data (human operator) > 30 Hz
    - Force data (teleoperator) > 1000 Hz
    - It is an open question what should be the maximum value of IPG to avoid the discontinuity of force data
  - Compression scheme for force data
    - Allows lower sampling rate
    - Maximum value of IPG can be increased

* IPG: inter-packet gap

# Conclusion/Future Work

- TCP and UDP protocols were investigated for interactive applications
- ETP based on IPG insertion was implemented and compared with existing protocols
- IPG in TCP did not improve end-to-end delay performance
- IPG in UDP improved end-to-end delay performance
- Optimal value of IPG needs to be defined:
  - Avoid variation of end-to-end delay
  - Prevent discontinuity of haptic data
- Haptic data compression scheme is expected to increase IPG

* IPG: inter-packet gap
* ETP: efficient transport protocol

# References


- [1] E. Kamrani, H. Momeni, and A. Sharafat, "Modeling Internet delay dynamics for teleoperation," *Proc. IEEE International Conference on Control Applications*, Aug. 2005.
- [2] G. Niemeyer and J. Slotine, "Designing force reflecting teleoperators with large time delays to appear as virtual tools," *Proc. IEEE International Conference on Robotics and Automation*, Albuquerque, NM, Apr. 1997, pp. 2212–2218.
- [3] K. Kawashima, K. Tadano, G, Sankaranarayana, and B. Hannaford, "Bilateral teleoperation with time delay using modified wave variables," *Proc. IEEE International Conference on Intelligent Robots and Systems*, Sept. 2008.
- [4] S. Clarke, G. Schillhuber, M. Zach, and H. Ulbrich, "The effects of simulated inertia and force prediction on delayed telepresence," *Presence*, vol. 16, no. 5, pp. 543-558, Oct. 2007.
- [5] S. Munir and W. Book, "Internet-based teleoperation using wave variables with prediction," *IEEE Transactions on Mechatronics*, vol. 7, no 2, June 2002.
- [6] H. Schulzrinne, S. Deering, R. Frederick, and V. Jacobson, "RTP: a transport protocol for real-time applications," *RFC 3550*, July 2003.
- [7] L. Ping, "Transport layer protocol reconfiguration for network-based robot control system," *Proc. IEEE International Conference on Networking, Sensing and Control*, Tucson, AZ, Mar. 2005, pp. 1049-1053.
- [8] Y. Uchimura, "Bilateral robot system on the real-time network structure," *IEEE Transactions on Industrial El ectronics*, vol. 51, no. 5, pp. 940-946, Oct. 2004.
- [9] R. Wirz, "Efficient transport protocol for networked haptics applications," *Proc. The 6th International Conference on Haptics: Perception, Devices and Scenarios*, Madrid, Spain, June 2008, vol. 5024, pp. 3-12.

# Thank you