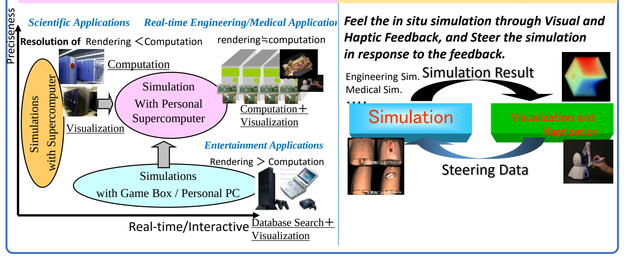
Prototype Implementation of Simulation Caching Framework for Remote Multi-user Interaction

J. Zhang, Y. Yamamoto, S. Fukuma, S. Mori

Univ. of Fukui, JAPAN

Motivation

Collaborative and Interactive Supercomputing — **Human in the Loop Scientific Simulation**



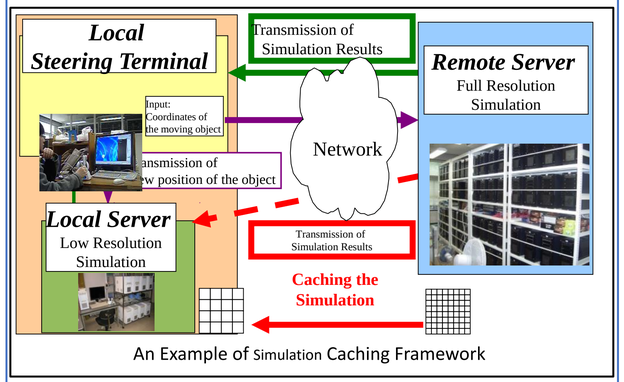
Feel the in situ simulation through Visual and Haptic Feedback, and Steer the simulation in response to the feedback.

Simulation Caching

Latency Hiding

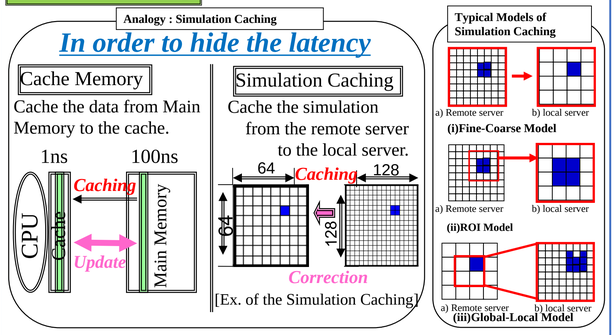
The Simulation Caching

1) utilizes a moderate scale computing server (a local server), associate with local operation terminal, to perform a sort of simulation, low resolution simulation for example, to make an immediate and reasonable response to the operators intervention, and
2) keeps the accuracy of the cached simulation by weakly cooperating with the original simulation running on the remote server.

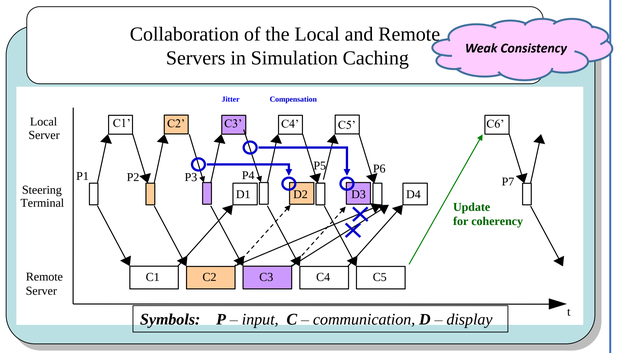


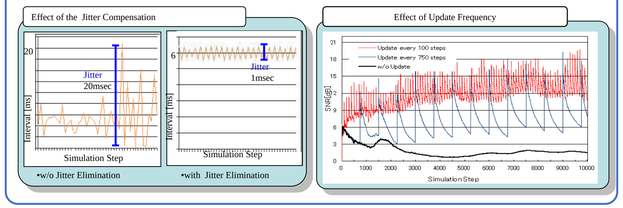
An Example of Simulation Caching Framework

Simulation Caching



Collaboration : Keep Weak Consistency





Simulation Caching Framework for Multi-user Interaction

In order to provide Simulation Caching Environment to simulation users, we have developed a Simulation Caching API with the functionality of both real-time sharing and interactive steering of the simulation among distanced users. The following figures shows the concepts of distanced simulation sharing and the overview of the prototype implementation of Simulation Caching Framework. With this framework, simulation programmers may concentrate on the programming of their own simulation provided that the steering input function and the visualization of simulation result should be separated from the simulation core.

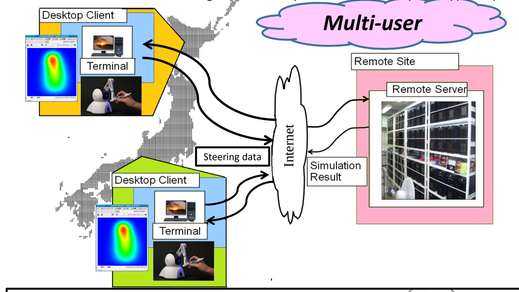
The key components of this framework are listed below:

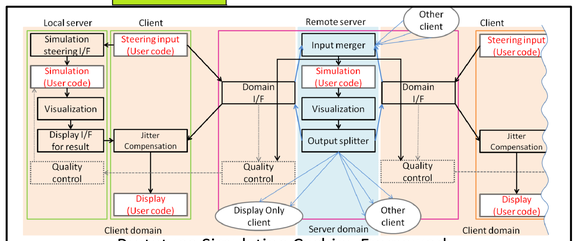
Quality control : assures 1) the quality of communication between distanced users and 2) the weak consistency of local simulation performed locally at user sites.

Input Merger : merges the steering inputs from users and feeds them to the simulation program. Merged inputs are also send to the local servers to weakly maintain the input consistency.

Output Splitter : appropriately splits the simulation result to uses. In order to reduce the network traffic data compression is also take place here.

Domain Interface : which provides interface between the simulation caching domains of each user and the simulation core on remote server, also maintains the security between domains though it is not implemented in the prototype implementation.

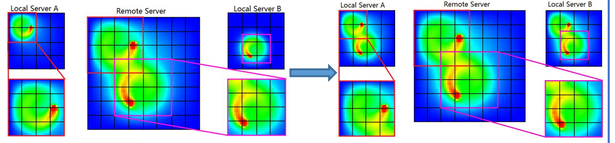




Prototype Simulation Caching Framework

Results

Before consistency, local server A or B don't know what happened by others.
→After consistency, local sever A knows B's latest simulation or input, same to B.



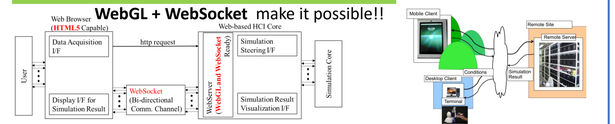
Conclusions

In this poster, we showed our work-in-progress researches toward ubiquitous and interactive supercomputing. The Simulation Caching realizes interactive remote steering of simulation by hiding the network latency while the integrated simulation caching framework realizes remote multi-user collaboration over the on-going simulation keeping weak consistency among the remote user.

Currently, we have been improving our framework so that it could deal with some other collaboration scenarios, for example, the scenario where the local server performs limited-regional finer-grid simulation. We are also developing the Web-based monitoring and steering interface to allow the sharing of the simulation among multiple remote users.

Further Extension: Web-based Simulation Steering

WebGL + WebSocket make it possible!!



"Tiled Touch Pad" using multiple mobile devices

Multi-user extension is also easy to realize.

Bi-Directional and Scalable HCI using mobile devices.

- Dynamically Scalable Tiled Display
  Pixel Count and Frame Rate Scalability
- Simulation/Visualization Steering using multiple touch panels.

References

[1] K. Hashimoto, S. Tezuka, S. Mori : Simulation Caching and Its Application to Remote Interactive Fluid Simulation, IPSJ Trans. on Advanced Computing Systems, Vol.5, No.4. pp.76–86(2012)(in Japanese).

[2] R. Henmi, Y. Nishimura, H. Suzuki, S. Fukuma, S. Mori, A. Yamaguchi, S. Tomita : Prototype Implementation of a GPU-based Interactive Coupled Fluid-Structure Simulation, Proc. of Int'l Conf. on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD 2012), pp.721–725(2012).

[3] J. Zhang, S. Yuasa, S. Fukuma, and S. Mori. A real-time gpu-based coupled fluid-structure simulation with haptic interaction. In Proc. Of Int'l Conf. on Computer and Information Science (ICIS 2016), pages 1–6, June 2016.

[4] F. Arakawa, Y. Yamamoto, R. Henmi, S. Fukuma, S. Mori: Toward Interactive Supercomputing: Simulation Caching & Multi-Client Web-Based Interactive HCI, Int'l Supercomputing Conf. HPC in Asia Poster Session 12(a), 2013.

[5] S. Masui, N. Shibata, F. Arakawa, S. Fukuma, S. Mori. Web-based Ubiquitous Simulation Steering based on Simulation Caching. IEEE Symp. Pacific Visualization, Poster Session, April 2016.

Acknowledgement

This research is supported in part by JSPS KAKENHI Grant No. 25280042.