Extension of Simulation Caching Framework for Large-scale Simulation

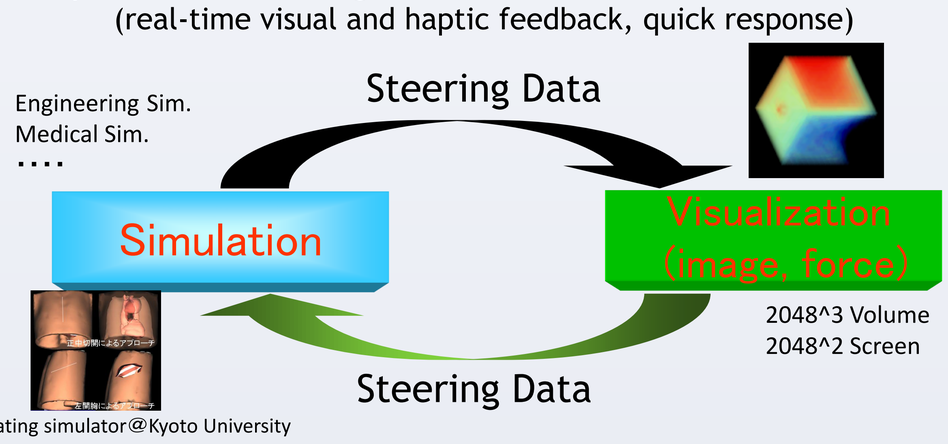
Yoshitaka Kumada, Jiachao Zhang, Shinji Fukuma, Shin-ichiro Mori

University of Fukui

1. Motivation: Interactive Simulation

Interactive Simulation: Human/Machine-In-The-Loop Scientific Simulation.

Computational Steering of In situ Simulation and Visualization
(real-time visual and haptic feedback, quick response)



Combine: Computing World ----------Real World

Merits: In situ Simulation & Visualization, Real-time Interaction, Dynamic Analysis, On-going phenomena in real world.

Interesting and **Expected**: e.g. surgery simulation, cyber physical system.

Tradeoff: usually, it's a tradeoff of interactivity and computing accuracy.

Real-time: requires a high performance of whole system.

(However, HPC resources are usually located far from **users**)

(Men or machines)

2. Simulation Caching Framework

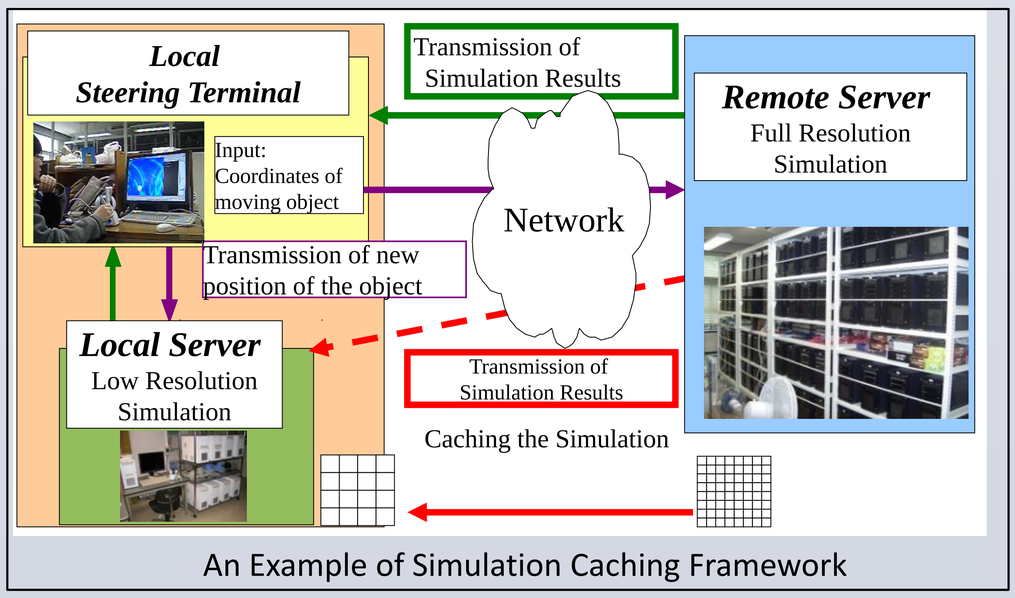
Real-time Interaction & Improved Accuracy by Edge/Fog-Like Computing.

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' simulation steering, 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

Effects of quality consistency control

Weak Consistency

Improve simulation result on local server through quality consistency control.

A simple case of 2D heat diffusion simulation

3. Extension

Problems:

So far, we only investigate small-scale simulation… Thus…

1) Not utilize the power of highly parallel computing on computing nodes.
2) Not investigate and consider side effects of large-scale transferred data.

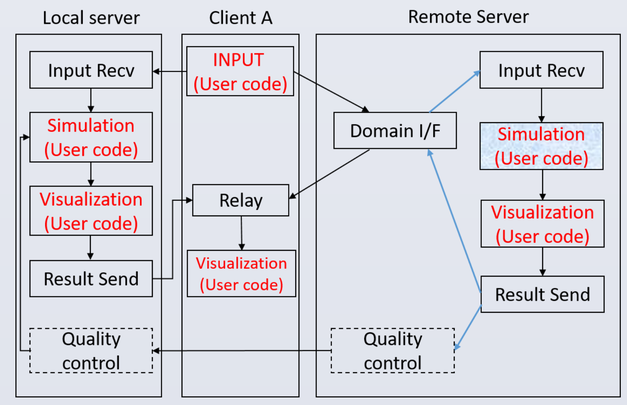
So Extension:

tries to endow parallel computing ability for framework w/o introducing security issue to computing nodes.

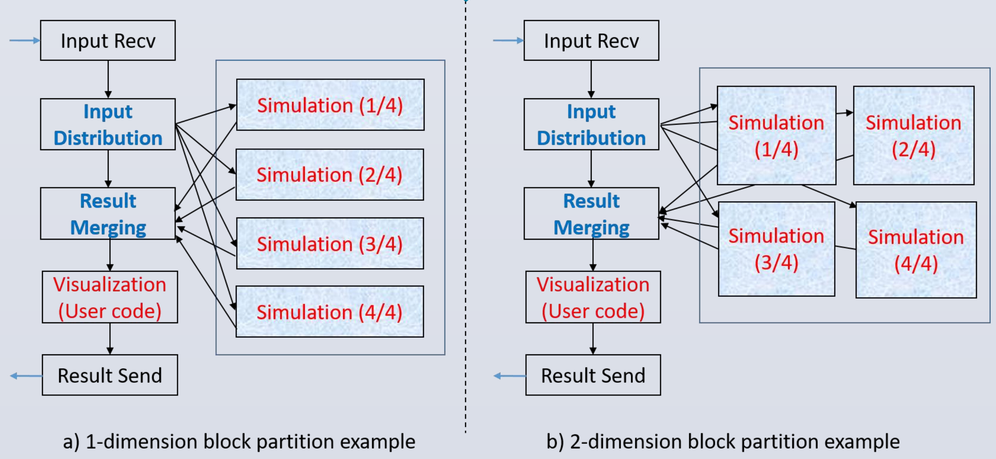
What we did (library extension as the framework developer):

1) Inter-node parallel computing on back-end computing nodes.
2) Data scattering and gathering for typical domain decomposition.
3) Utilize data compression to reduce transferred consistency data size.

- Using an open-source electronic-magnetic simulation **OpenFDTD**.
- Assumption: Login node can communicate with computing nodes.
- Between Domain/IF, Quality control and Input/Result processes, we use local socket to loosely couple two separate MPI world.



Overview of original Simulation Caching Framework



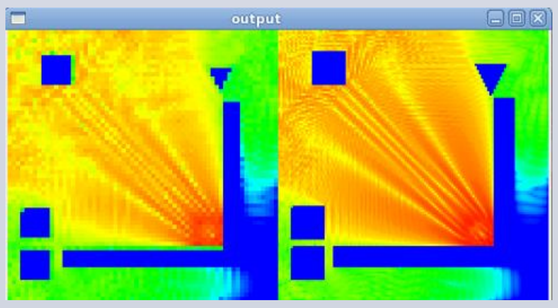
a) 1-dimension block partition example

b) 2-dimension block partition example

Extension of parallel computing for large scale simulation

4. Conclusion

1) Implement extended framework (using a large-scale electronic-magnetic simulator OpenFDTD as studied case).
2) Utilize power of inter-node parallel computing on many nodes system.
3) Provides user interface for data scattering and gathering.
4) Optimize by endeavors like parallel data compress/decompression.



A snapshot of implemented OpenFDTD simulation.
left) Local Simulation, right) Remote Simulation

References

1. EEM Company 2017. OpenFDTD-Open Source FDTD Method Program.
2. Yu Yamamoto et al. Implementation of Simulation Caching Framework, IPSJ Journal, Vol.57(3), pp.823-835, 2016
3. Jiachao Zhang et al. A Real-time GPU-based Coupled Fluid-Structure Simulation with Haptic Interaction, Int'l Journal of Computer & Information Science(IJCIS), vol.17, No.4, pp.1-10, 2016.
4. Jiachao Zhang et al. Evaluation of Three Quads using matrix transpose, Journal of Information Processing(JIP), vol.26, 2018(In Press)

Contact

Jiachao Zhang, PhD Candidate
University of Fukui
zhang@sylph.fuis.u-fukui.ac.jp