Towards the real-time local weather forecasting using CReSS on GPU clusters

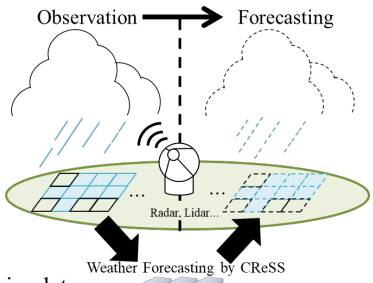
Naoya Nomura, Masato Gocho, Takuya Uesugi, Kei Akama, Tetsutaro Yamada, and Hiroshi Sakamaki
Information Technology R&D Center, Mitsubishi Electric Corporation, Japan

◆ Introduction

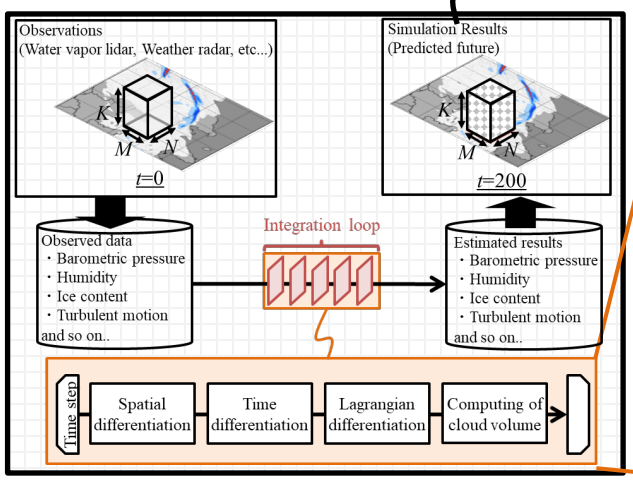
- Predicting heavy rains and storms plays an important role in preventing and mitigating devastating damage to urban life and infrastructure.
  → The real-time weather forecasting system is needed to construct this safety system.
- CReSS (Cloud Resolving Storm Simulator)[1] is the model that can predict local cloud conditions with high accuracy.
  But CReSS takes a huge computation time.
  →Parallelization is required to realize the real-time local weather forecasting system.
- In this study, we demonstrate a developed CReSS implemented on a single GPU system that is a prototype system of GPU clusters.

◆ Weather forecasting flow of CReSS

- We aim to predict the precipitation locations and amounts by simulating cloud movement.

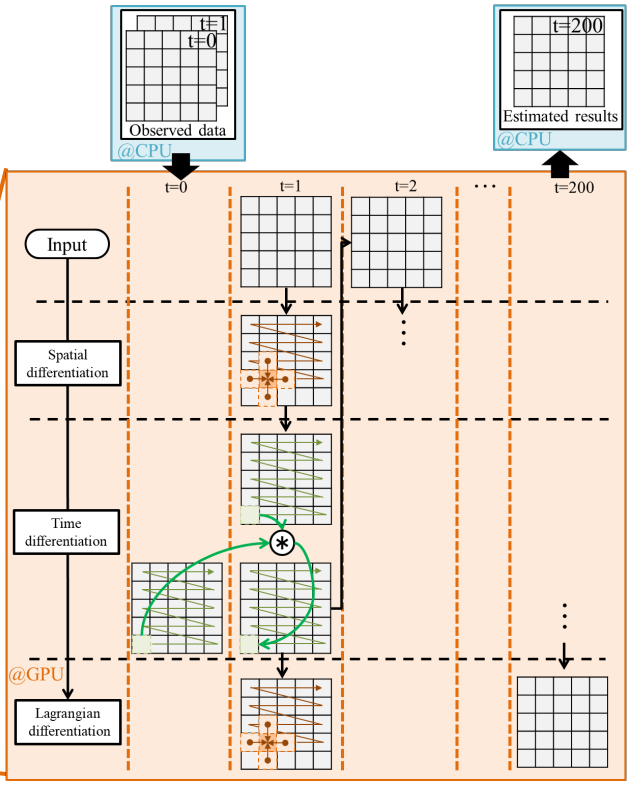


- **CReSS** is a cloud model simulator. In CReSS, there are three types of differential computations for each time integration loop.



◆ Parallelized CReSS by GPU

In this study, distributed parallel CReSS on GPU system is developed using OpenACC. OpenACC is the directive-based framework for parallel programming. In this system, 128 GPU cores is set per 1 thread block, and the entire mesh is divided and distributed to all thread-blocks.



◆ Experimental Results

GPU-based implementation is 19 times faster than CPU.
— In 203 × 203 × 53 mesh size, CPU: 332.6 [s], GPU: 17.5 [s]

CPU		Intel Xeon Gold 6142 (2.6[GHz]) × 16 Cores
GPU		NVIDIA Tesla V100-SXM2 (1.37[GHz]) × 5,120 Cores
OS		Ubuntu 16.04
Compiler	CPU	GNU Fortran 5.4.0 (Option: -O2 -fopenmp)
	GPU	NVIDIA HPC SDK 20.7, CUDA 11.0 (Option: -O2 -acc -ta=tesla,cc70 -lm)

- Integration loop is processed in 200 steps.
- In "CPU", the original code of CReSS is used and parallelized in 16 threads.

Execution time[s]	Mesh size 43 × 43 × 53		Mesh size 203 × 203 × 53	
	CPU	GPU	CPU	GPU
Spatial differentiation	0.8	0.1	3.0	0.6
Time differentiation	49.9	1.4	145.2	8.3
Lagrangian differentiation	18.7	0.6	104.6	4.1
Computing cloud volume	6.7	2.8	32.3	4.4
Others	18.9	0.1	47.5	0.1
Whole execution time	95.1	**5.1**	332.6	**17.5**

◆ Conclusion

- We demonstrated GPU-accelerated CReSS towards real-time local weather forecasting on GPU clusters.
- To evaluate the efficiency of the calculation speed, we measured the execution time of the developed CReSS for a single GPU system.
  — The time of weather prediction is 19 times faster than CPU (CPU: 332.6 [s], GPU: 17.5 [s]).
- We will implement parallel CReSS for GPU cluster system and evaluate the efficiency of the computational time as a future work.
  → We aim to implement on CPU/GPU cluster computational system and analysis the calculation time of CReSS.

◆ Acknowledgements

We deeply appreciate for the cooperation of National Research Institute for Earth Science and Disaster Resilience.

◆ References

[1] K. Tsuboki and A. Sakakibara, "Large-scale parallel computing of Cloud Resolving Storm Simulator", High Performance Computing, Springer, pp.243–259, 2002.