INFORMATION TECHNOLOGY CENTER
THE UNIVERSITY OF TOKYO

JCAHPC

# Multiplicative Schwartz type Block Multi-color Gauss-Seidel Smoother for Multigrid

Masatoshi Kawai

THE UNIVERSITY OF TOKYO

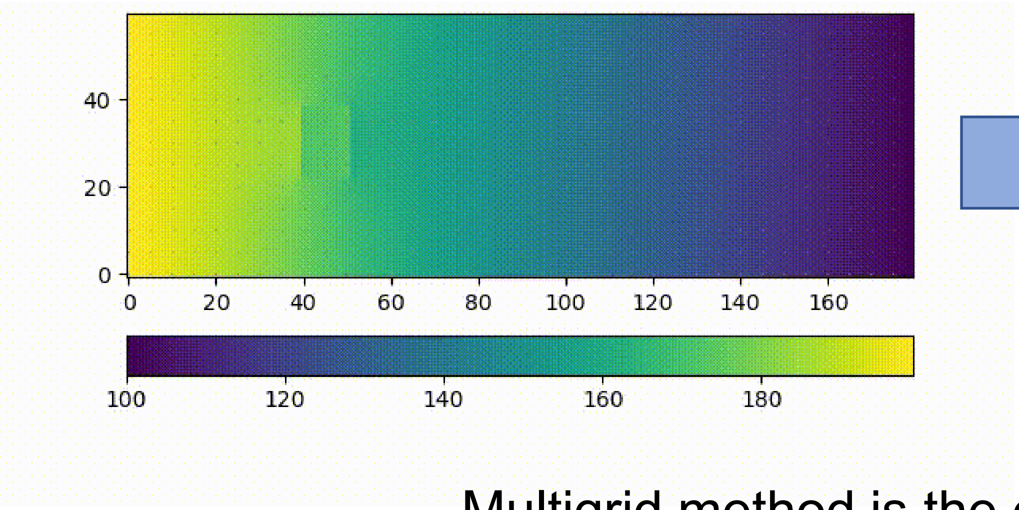
# Multigrid method

A multigrid method is effective iterative method for solving SLEs

Required a fast solver for solving SLEs which has sparse coefficient matrix

→ Such SLEs are derived from various simulations and consumes most of the simulation time.

Example of a CFD




Example of the SLE

$$A^h x = b$$

$A^h$ : Sparse coefficient matrix

$x$ : Vector of Exact solution

$b$ : Vector of right-side values

Multigrid method is the effective method for solving such SLEs

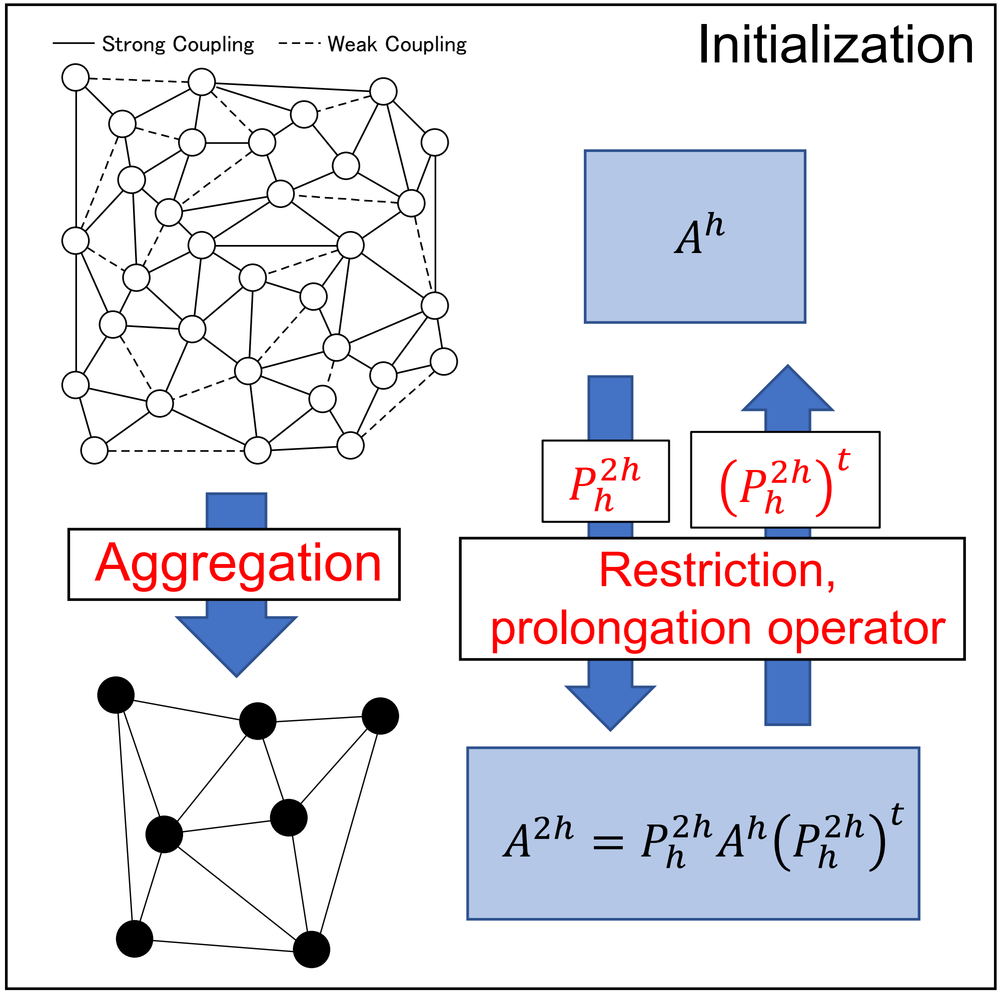
- One of an iterative method
- High convergence rate

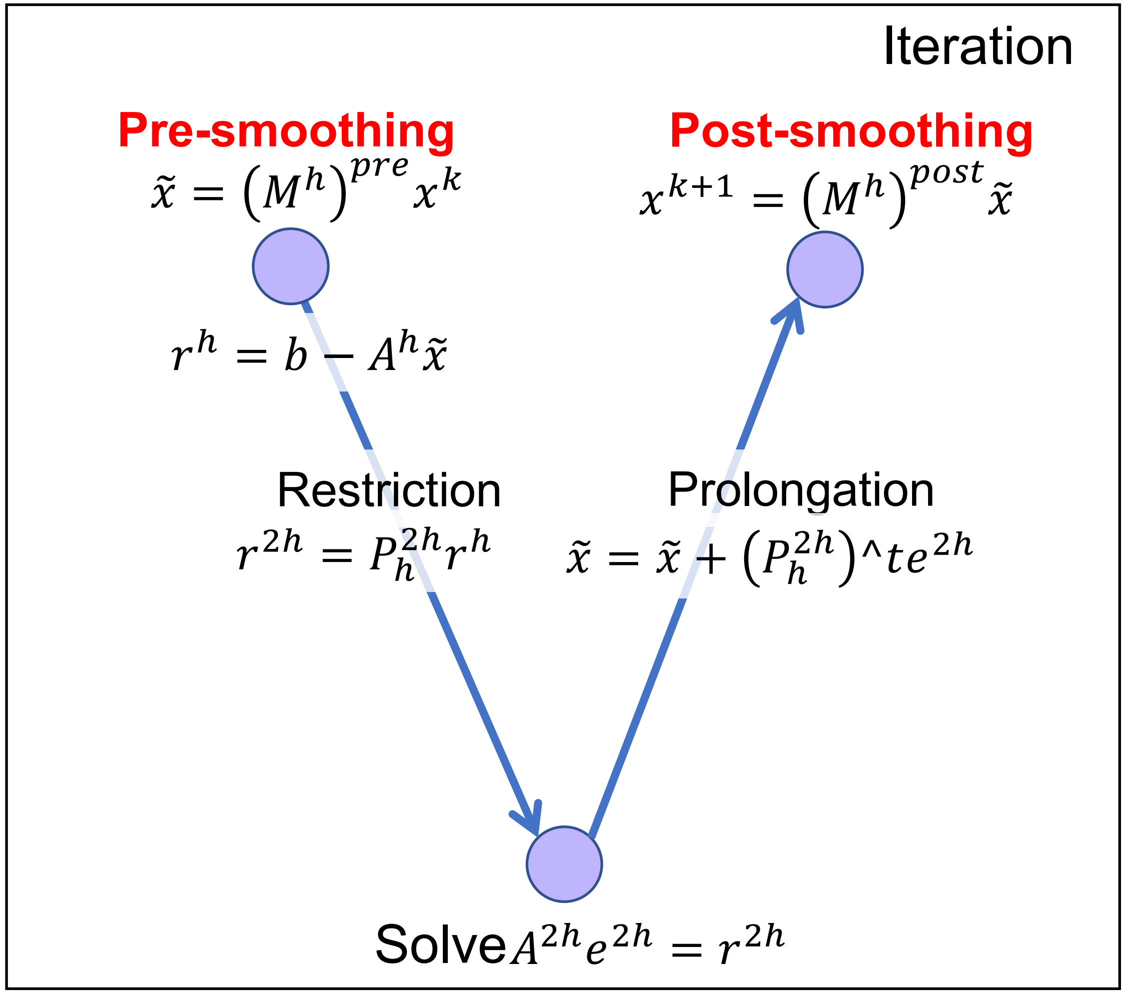
Focusing on developing fast multigrid method

# Smoother

## Considering to develop a fast and effective smoother

The smoother has strong impact on the convergence rate and computational cost of the multigrid method



# Existing smoothers

## Introducing the existing smoothers

- Sequential Gauss-Seidel smoother
  - Basic smoother
  - Difficult to parallelize
- Multi-color Gauss-Seidel (MC-GS) smoother
  - Well-known approach to parallelize GS
  - Low-data locality
- Block multi-color Gauss-Seidel (BMC-GS) smoother
  - Applying blocking approach to MC-GS
  - Improving data-locality compared with MC-GS

| | Convergence | Data locality | Parallelization |
|---|---|---|---|
| **Sequential GS** | ○ | ○ | ✕ |
| **MC-GS** | ○ | ✕ | ○ |
| **BMC-GS** | ○ | ○ | ○ |

# Multiplicative Schwartz type block multi-color Gauss-Seidel smoother

**Superiority of a multiplicative Schwartz type block multi-color Gauss-Seidel smoother (MS-BMC-GS) is high convergence rate and data locality.**

Normal case including BMC-GS

$\beta$ times iteration for all grid points → Computational time in 1 multigrid cycle is also increased by factor of $\beta$

↓

To reduce the total computational time, the number of multigrid cycles must be decreased by a factor of $\beta$.

**In the MS-BRB-GS**

$\alpha$ times GS steps **in each block (Set the block size less than cache size)**

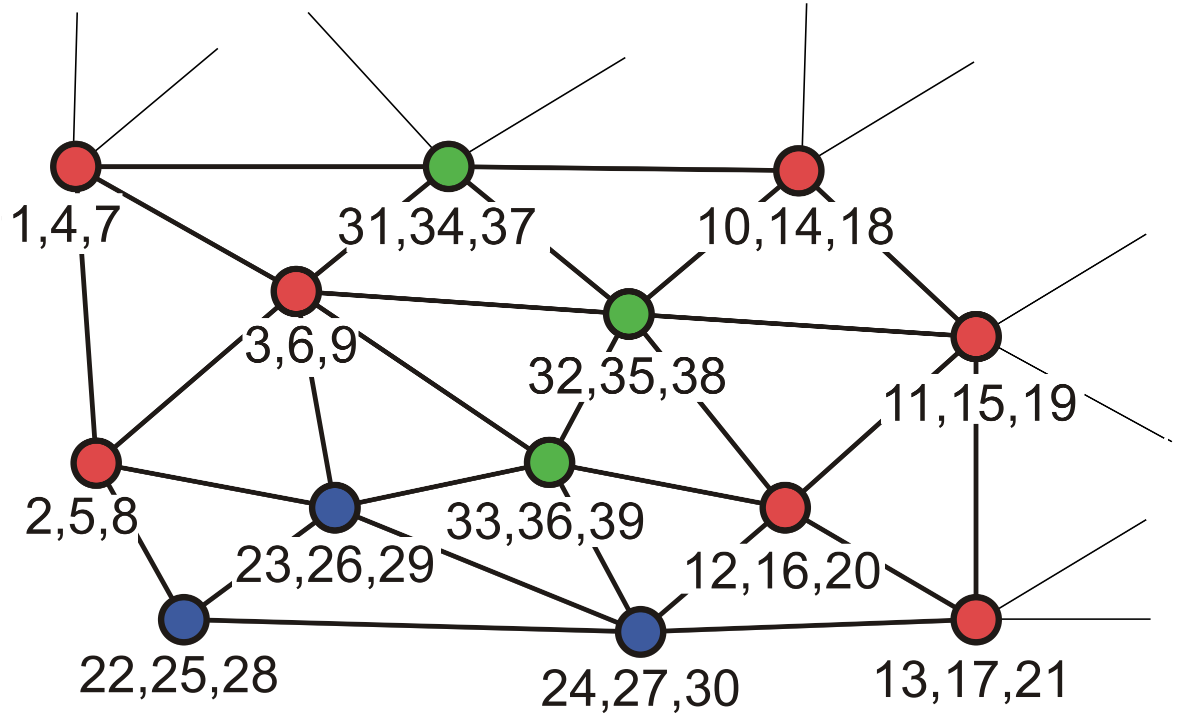
Thanks to the cache, the computational time for one smoothing step is not increased by a factor of $\alpha$.

↓

The improvement of convergence caused by increasing the number of GS steps in the block may reduce the total computational time.

# Difference between BMC-GS and MS-BMC-GS

Difference is a computational order with more than twice updating.

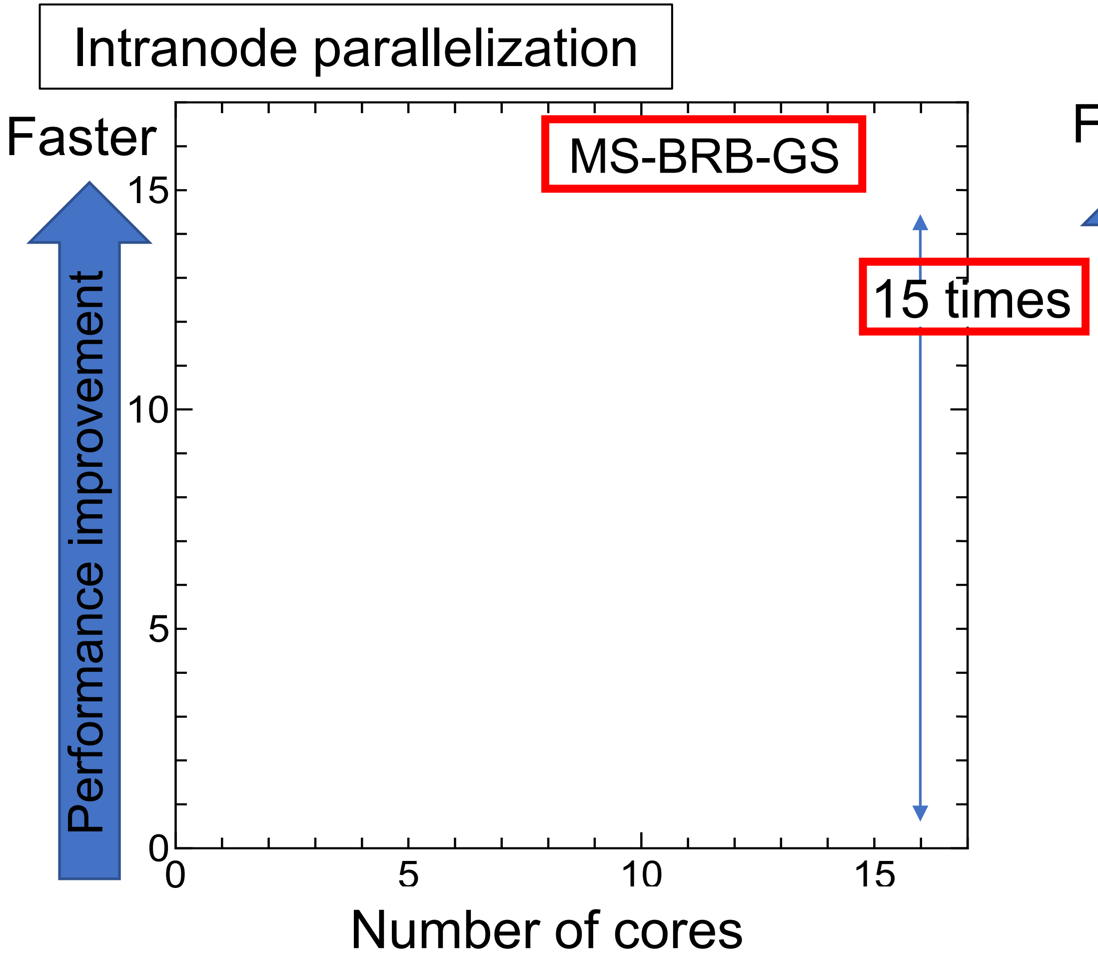


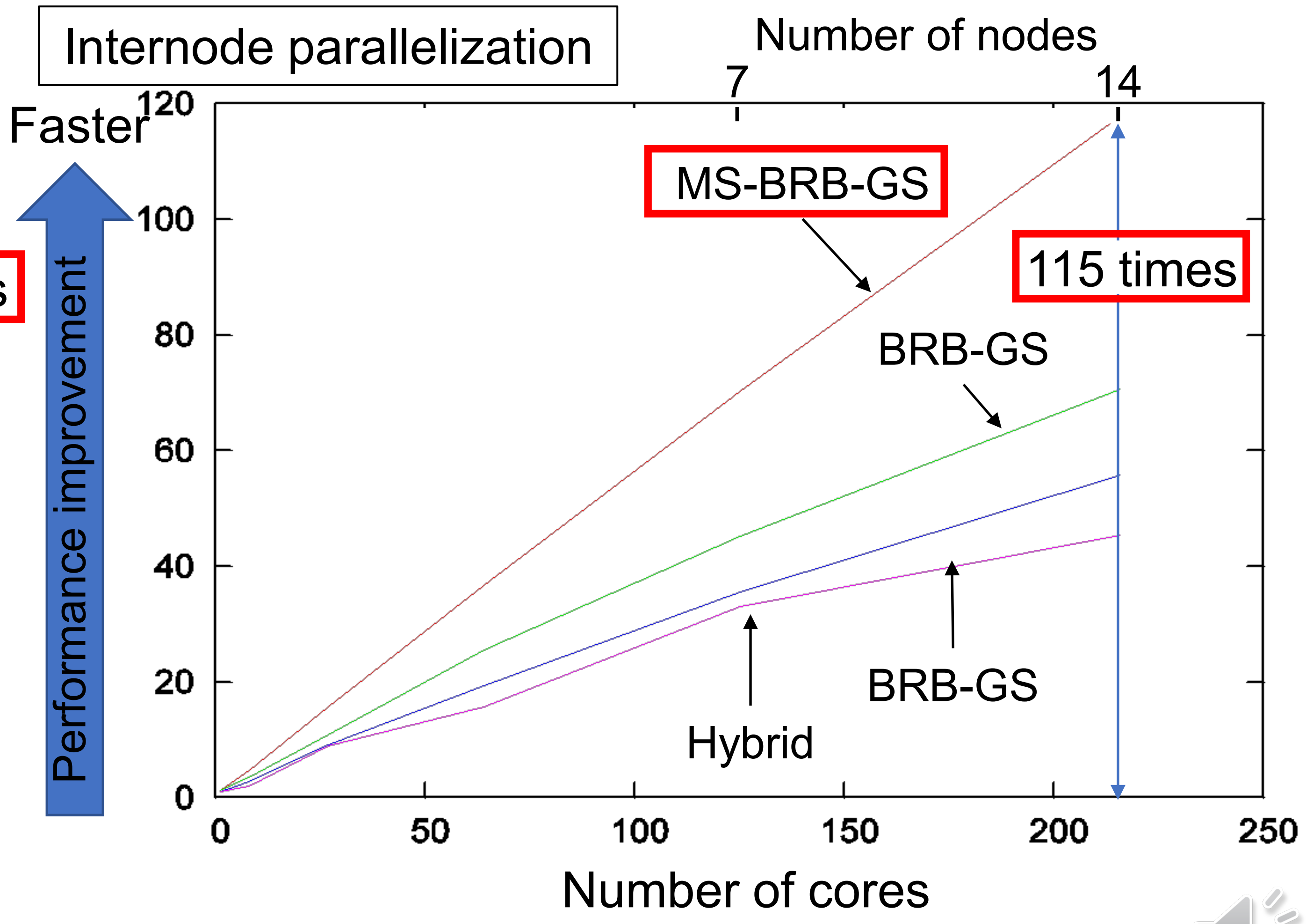

- The example of the BMC-GS
- Three times updating all nodes on the BMC-GS(3) smoothers
  → Three times the computational time
- Three times updating each block, continuously
  → Shorter computational time

Then, required amount of memory size in each block must be smaller than the cache size.

# Result 1 : MS-BMC-GS with a geometric multigrid method

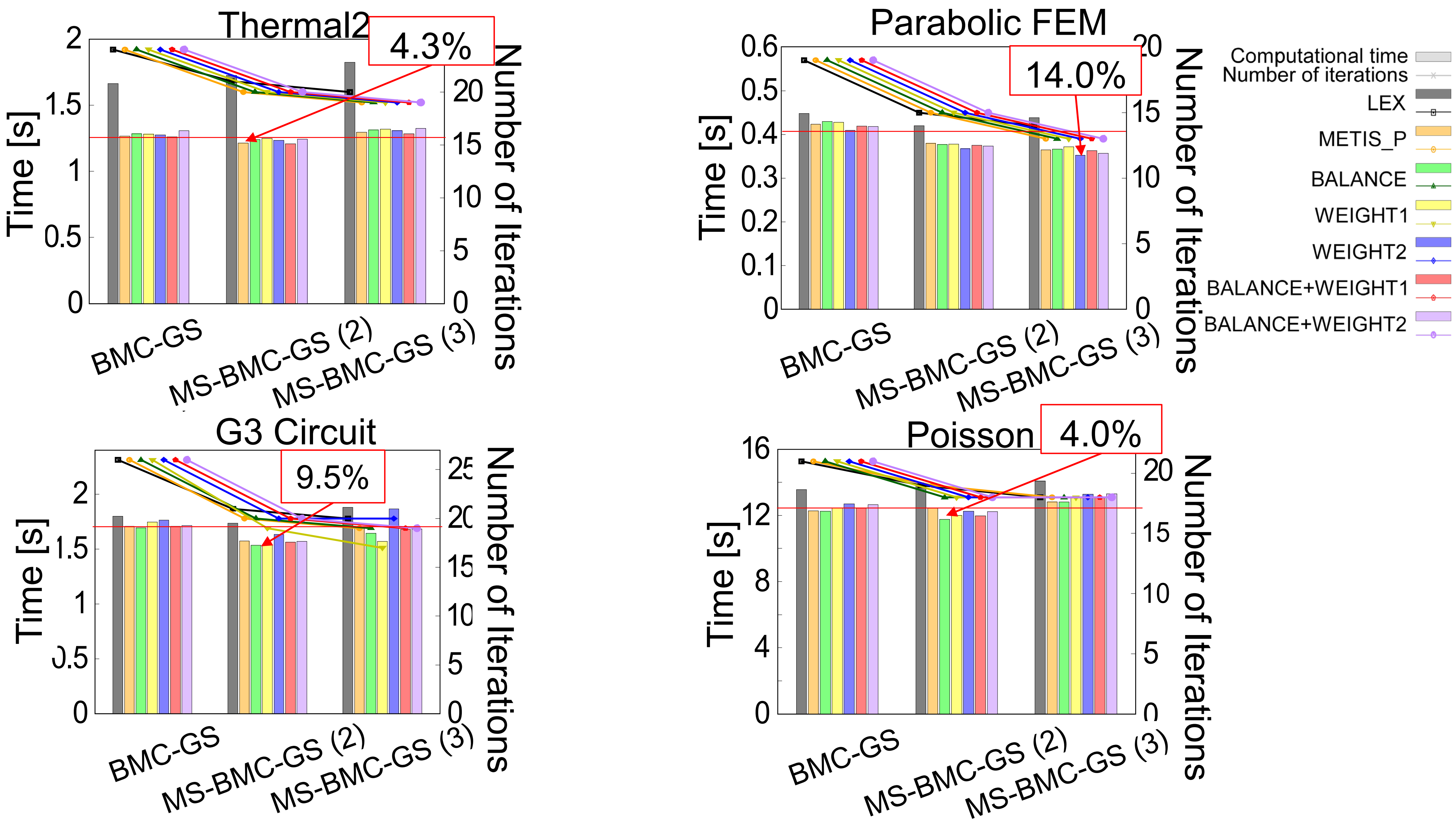
Performance improvement of MS-BMC-GS with a geometric multigrid method



# Result 1 : MS-BMC-GS with an algebraic multigrid method

MS-BMC-GS shows better performance with all problems.



# Conclusion

- We propose a multiplicative Schwartz type block multi-color Gauss-Seidel (MS-BMC-GS) smoother for multigrid method.
- The superiority of the MS-BMC-GS is  higher convergence rate and data locality compared with existing smoothers
- Repeatedly Gauss-Seidel iterations to each block achieves the superiority.
- With a geometric and an algebraic multigrid method, MS-BMC-GS shows better performance than existing smoothers.

Future works

- Evaluating MS-BMC-GS with larger problems
- Automatically deciding number of repeatedly GS iterations to each block with an auto-tuning approach

Thank you for watching