HPC Exam Project

Scaling Study of the Stencil Method

Giovanni Lucarelli

October 18, 2025

UNIVERSITÀ
DEGLI STUDI
DI TRIESTE

Introduction

Goals

1. **Optimize** the stencil method for the 2d heat equation
2. **Parallelize** using hybrid approach
3. Perform **scalability** study:
   - 3.1 Thread scaling
   - 3.2 Strong scaling
   - 3.3 Weak scaling

Algorithm

Heat equation (2d)

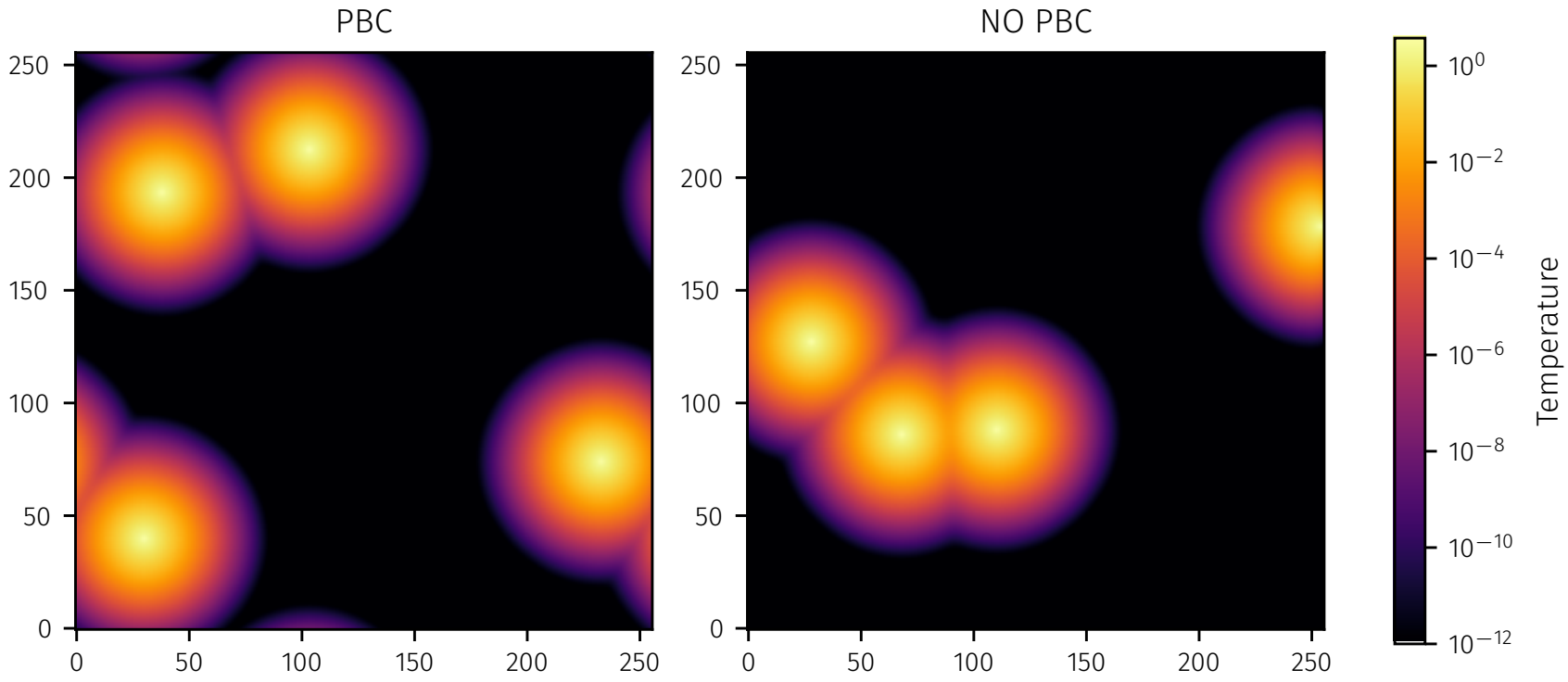
$$\partial_t u = \alpha(\partial_x^2 u + \partial_y^2 u)$$

Finite difference integration

$$u_{i,j}^{(t+1)} = (1 - 4\alpha)u_{i,j}^{(t)} + \alpha \sum_{\langle i,j \rangle} u_{i,j}^{(t)}$$

$$x \in [0, L_x] \to i \in \{1, \ldots, N_x - 1\}$$
$$y \in [0, L_y] \to j \in \{1, \ldots, N_y - 1\}$$

Code Correctness



Optimization

- Compiler flags:
  `-O3 -Wall -march=native`
- Preprocessor directive:
  `#pragma GCC unroll`

Parallelization: shared memory

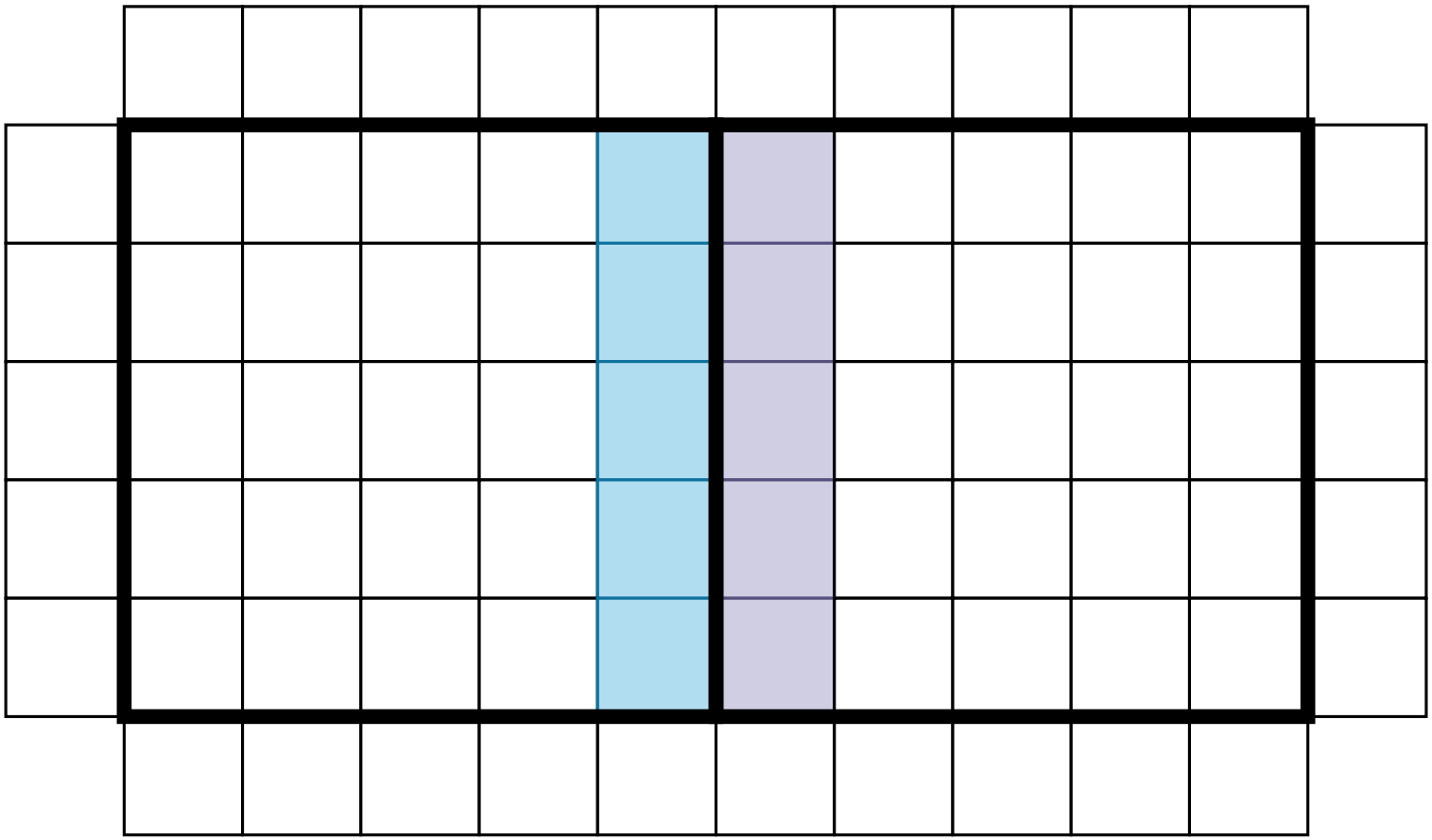
Implementation

```
1     #pragma omp parallel for schedule(static)
2     for (uint j = 1; j <= ysize; j++){
3         for ( uint i = 1; i <= xsize; i++){
4
5             // update rule
6
7         }
8     }
```

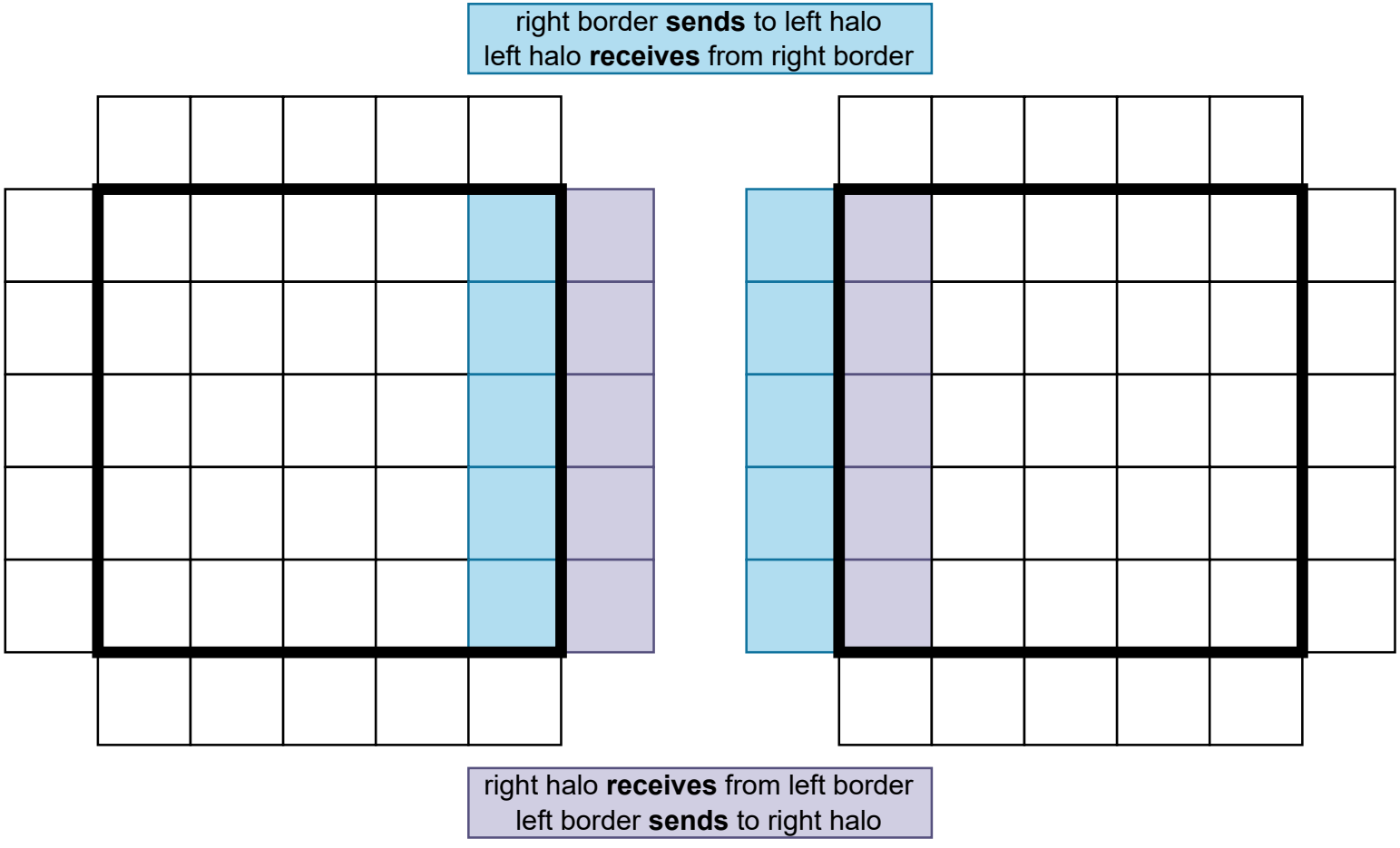
Thread placement and affinity

```
1     export OMP_PLACES=cores
2     export OMP_PROC_BIND=close
```

Parallelization: distributed memory



Parallelization: distributed memory



Parallelization: distributed memory

For each task:

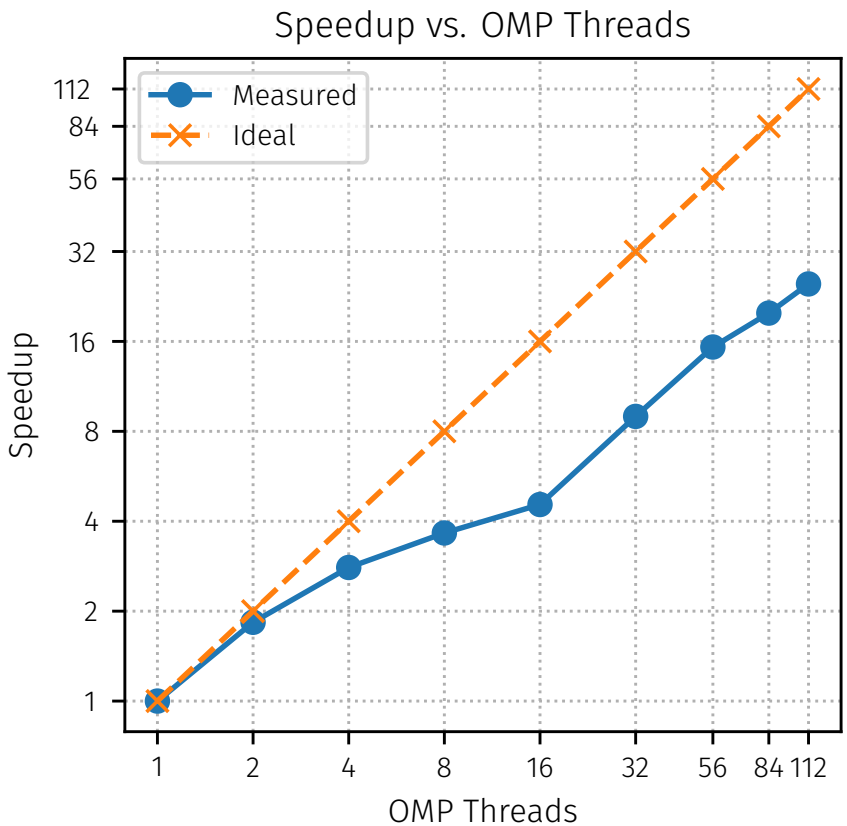
```
1   // pack buffers
2
3   MPI_Irecv(...);
4
5   MPI_Isend(...);
6
7   update_internal();
8
9   MPI_Waitall();
10
11  // unpack buffers
12
13  update_border();
```

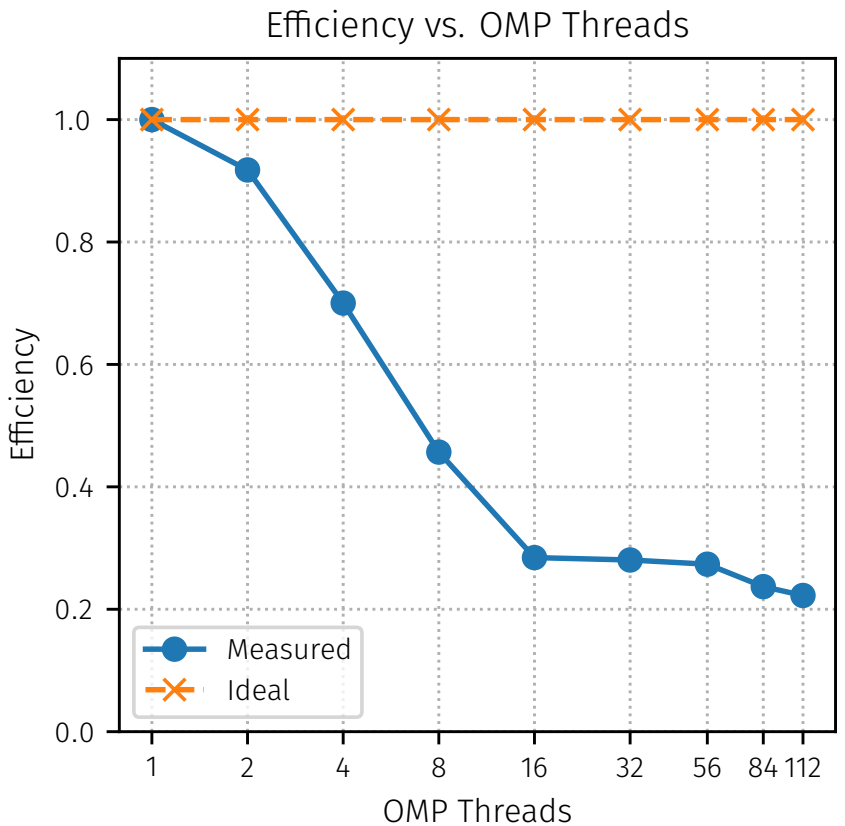
Results

Thread Scaling

```
GRID_SIZE_X=16384
GRID_SIZE_Y=16384
N_STEPS=500

NODES=1
N_TASKS_PER_NODE=1

THREAD_LIST="1 2 4 8 16 32 56 84 112"
```

Thread Scaling



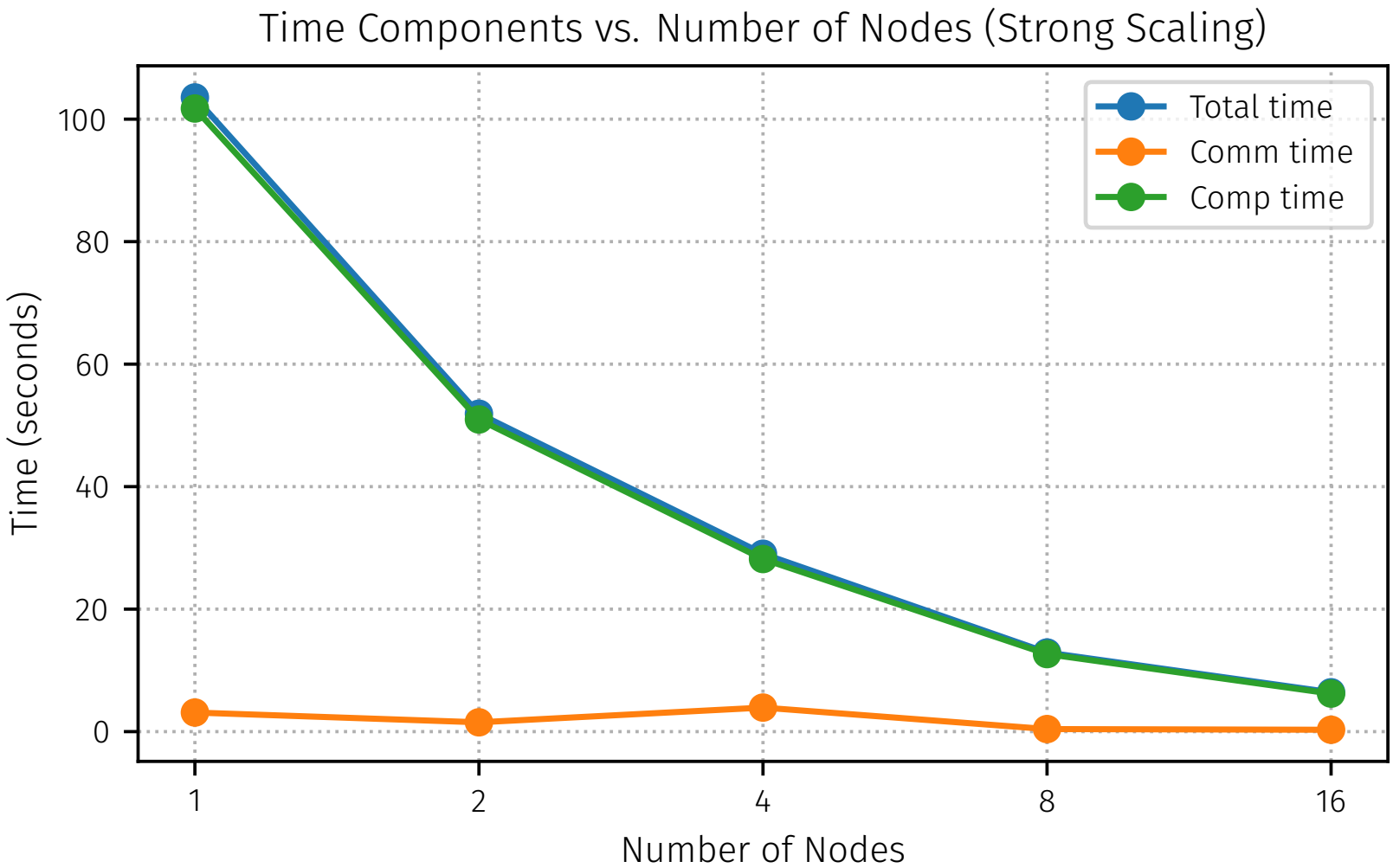


Node Topology

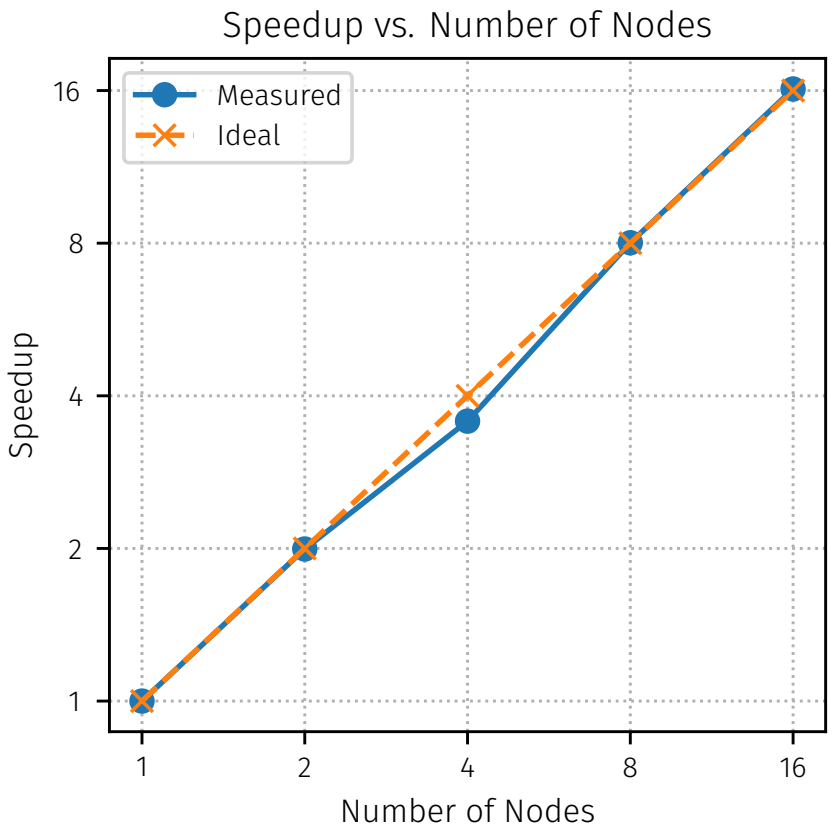
Strong Scaling

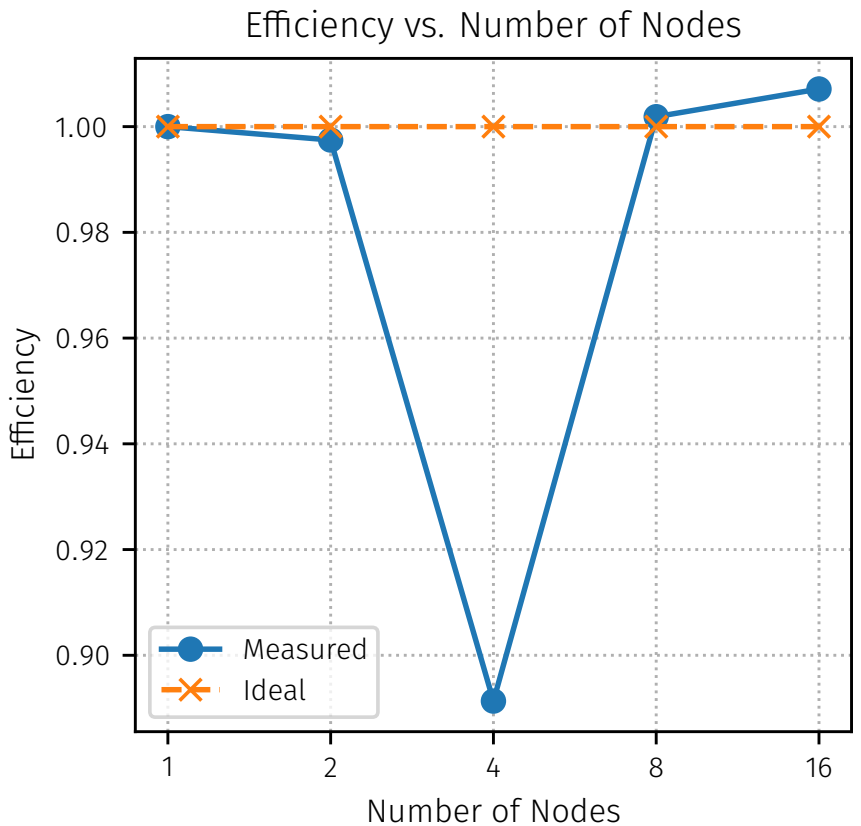
```
GRID_SIZE_X=65536
GRID_SIZE_Y=65536
N_STEPS=500d

OMP_THREADS=14
N_TASKS_PER_NODE=8

NODE_LIST="1 2 4 8 16"
```

Strong Scaling



Strong Scaling

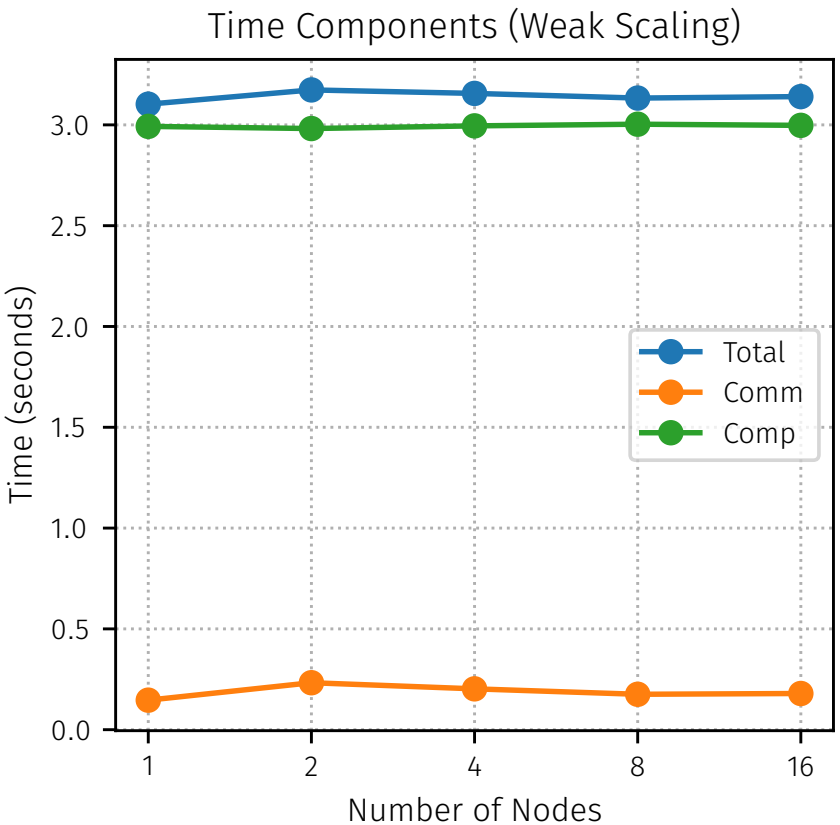


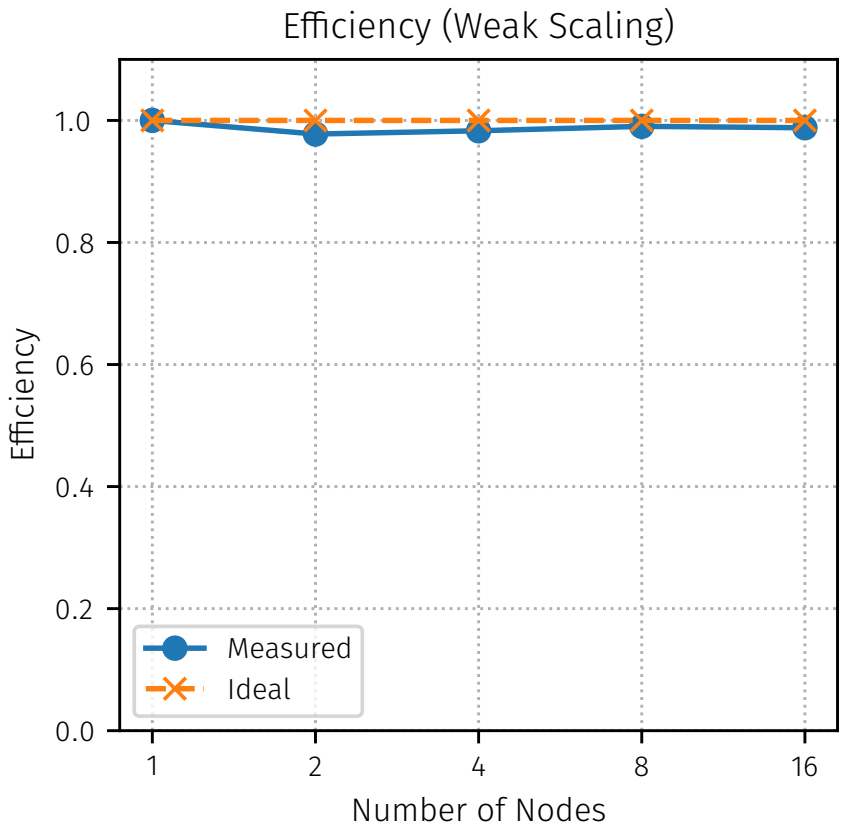


Weak Scaling

```
LOCAL_X=4096
LOCAL_Y=4096
N_STEPS=500

OMP_THREADS=14
TASKS_PER_NODE=8
NODE_LIST="1 2 4 8 16"

for NODES in NODE_LIST; do
    TOTAL_TASKS=$(( NODES * TASKS_PER_NODE ))

    case "${TOTAL_TASKS}" in
    8)    PX=4;  PY=2  ;;   # 1 node  (8 ranks)
    ...
    esac

    GRID_SIZE_X=$(( LOCAL_X * PX ))
    GRID_SIZE_Y=$(( LOCAL_Y * PY ))
    ...
done
```

Weak Scaling





Conclusion

- Stencil method is memory bound!

Thank You!