# hpcscan version 1.1
# Performance benchmarks on Shaheen II (KAUST)

Updated December 21, 2020

# Content

# Content

# Shaheen II (KAUST)

## Machine Shaheen II / Cray XC40

- Computing nodes Intel Haswell 2.3 Ghz dual socket (16 cores / socket)
- RAM 128 GB with Peak memory BW 136.5 GB/s
- Peak performance Single Prec. 2.36 TFLOP/s / Double Prec. 1.18 TFLOP/s
- Interconnect Cray Aries with Dragonfly topology
  - 60 GB/s optical links between groups
  - 8.5 GB/s copper links between chassis
  - 3.5 GB/s backplane within a chassis
  - 5 GB/s PCIe from node to Aries router

# Content

# Test Case Memory - Description

## Benchmark objective

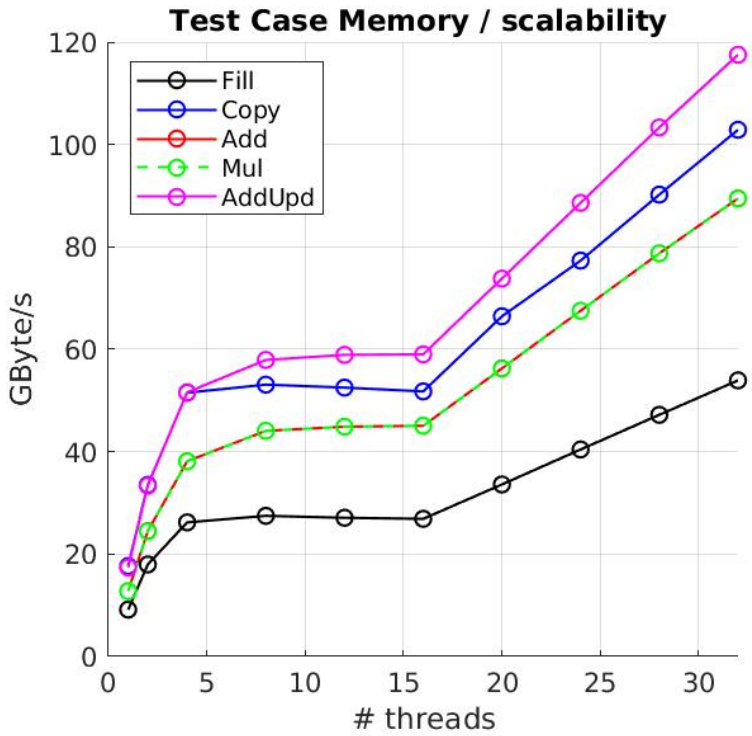
Measure GByte/s and GPoint/s for

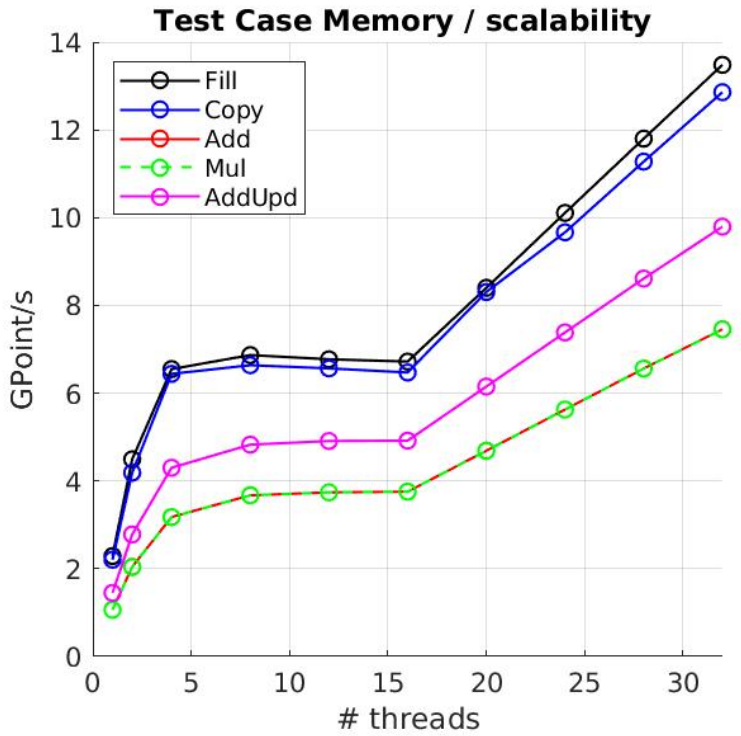
- Fill grid (W = coef)
- Copy grid (W = U)
- Add grids (W = U + V)
- Multiply grids (W = U * V)
- Add and update grids (W = W + U)

## Benchmark configuration

- Scalability on 1 node with 1 to 32 threads
- Baseline kernel
- Grid size 4 GB (1000 x 1000 x 1000 points)
- Reproduce results with `./script/testCase_Memory/hpcscanMemory.sh`
- Elapsed time about 4 minutes

# Test Case Memory - Results [1]




[1] Updated Dec 22, 2020

# Test Case Memory - Summary

- Measured memory BW between 91 to 122 GB/s (67-90 % of peak BW)
- Low BW 59 GB/s for Fill (43 % of peak BW)
- Multiply (= imaging condition) performs at 7.6 Gpoint/s

# Content

# Test Case Grid - Description

## Benchmark objective

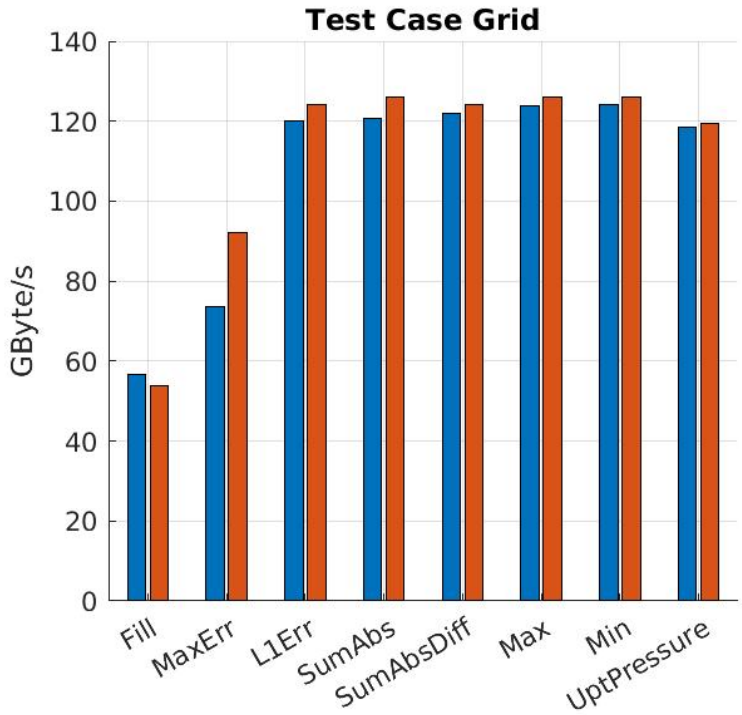
Measure GByte/s and GPoint/s for

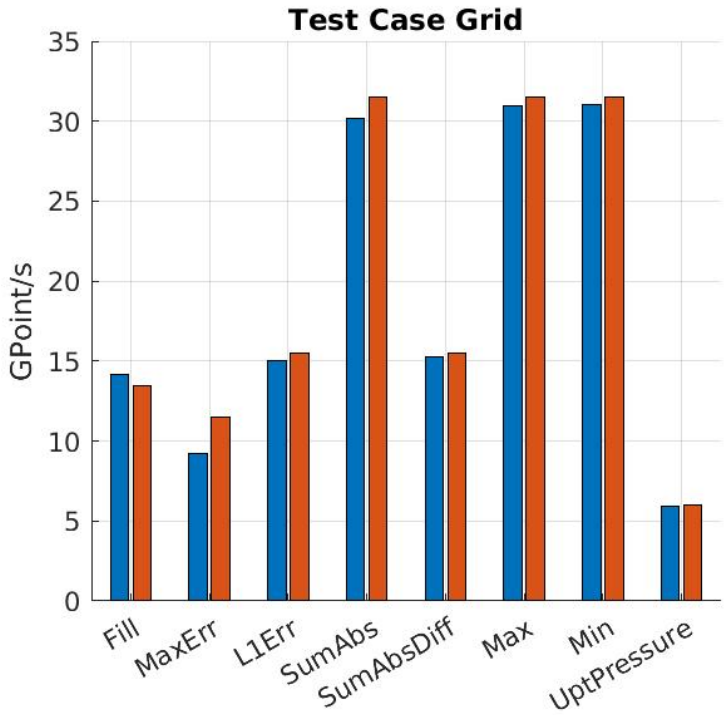
- Fill grid (U = coef)
- Max. diff (U-V)
- L1 norm between U and V
- Sum Abs(U) & Sum Abs(U-V)
- Get max. & Get min. grid U
- Update pressure (used in propagator)
- Boundary condition (free surface at all edges)

## Benchmark configuration

- 1 node with 32 threads
- Baseline kernel
- 2 grid sizes
  - Small size 500 MB (500 x 500 x 500 points)
  - Medium size 4 GB (1000 x 1000 x 1000 points)
- Reproduce results with `./script/testCase_Grid/hpcscanGrid.sh`
- Elapsed time less than 1 minute

# Test Case Grid - Results [2]




Blue small grid / Red medium grid
ApplyBoundaryCondition performs at 713/846 GBytes (89/105 Gpoint/s)

[2] Updated Dec 23, 2020

# Test Case Grid - Summary

- L1 Err., Get Min & Max: 125 GB/s close to peak BW (92 % Peak Mem. BW)
- Low perf for Fill: 54-58 GB/s (40-43 % Peak Mem. BW)
- Max Err. 72-91 GB/s (53-67 % Peak Mem. BW)
- Pressure update 6 GPoint/s (120 GB/s, 88 % Peak Mem. BW)

# Content

# Test Case Comm - Description

## Benchmark objective

Measure GByte/s and GPoint/s for

MPI point to point communication

- Send with MPI_Send from proc X to proc 0 (Half-duplex BW)
- Send and receive with MPI_Sendrecv between proc X and proc 0 (Full-duplex BW)

MPI collective communication

- Exhange of halos used in FD kernel with MPI_Sendrecv
- Domain decomposition with N1 x N2 x N3 subdomains

## Benchmark configuration

- 8 nodes with 32 threads
- Baseline kernel
- Grid size 4 GB (1000 x 1000 x 1000 points)
- Subdomain decomposition: 1x4x2 / 1x2x4 & 2x2x2
- Reproduce results with `./script/testCase_Comm/hpcscanComm.sh`
- Elapsed time less than 1 minute

# Test Case Comm - Results

Table: Bandwidth GB/s [3]

| MPI#1 | MPI#2 | Send | Sendrecv | Halo exch. | Comm. size | Subdomains |
|---|---|---|---|---|---|---|
| 0 | 1 | 8.5 | 15.3 | - | 47 MB | - |
| 0 | 2 | 8.3 | 15.3 | - | 47 MB | - |
| 0 | 3 | 8.6 | 15.3 | - | 47 MB | - |
| 0 | 4 | 8.5 | 15.3 | - | 47 MB | - |
| 0 | 5 | 8.2 | 15.3 | - | 47 MB | - |
| 0 | 6 | 8.5 | 15.3 | - | 47 MB | - |
| 0 | 7 | 8.6 | 15.3 | - | 47 MB | - |
| All | All | - | - | 5.0 | 128 MB | 1 4 2 |
| All | All | - | - | 5.1 | 128 MB | 1 2 4 |
| All | All | - | - | 2.0 | 96 MB | 2 2 2 |

[3] Updated Sep 19, 2020

# Content

# Test Case FD_D2 - Description

## Benchmark objective

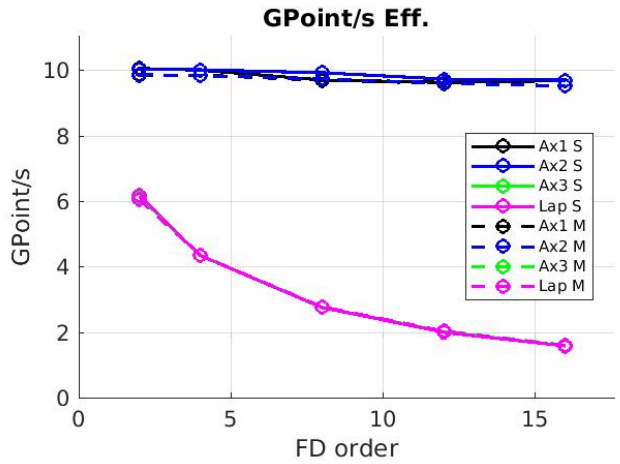
Measure GByte/s, GPoint/s & Gflop/s for

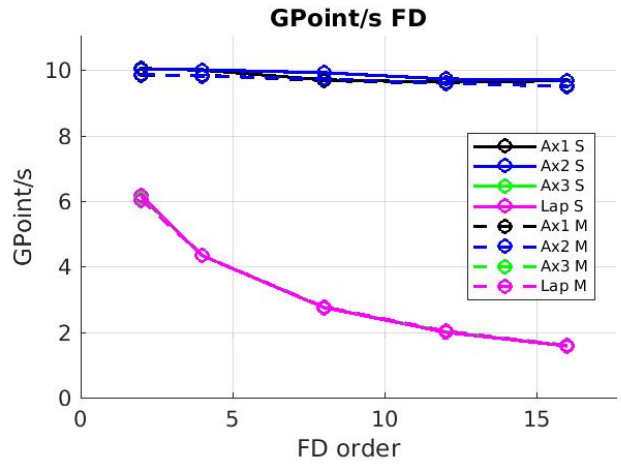
- Computation of second order derivatives with finite-difference stencil
- Directionnal derivatives
  - Axis 1, $W = \partial^2_{x1}(U)$
  - Axis 2, $W = \partial^2_{x2}(U)$
  - Axis 3, $W = \partial^2_{x3}(U)$
- Laplacian $W = \Delta(U)$
- Stencil orders 2, 4, 8, 12 & 16

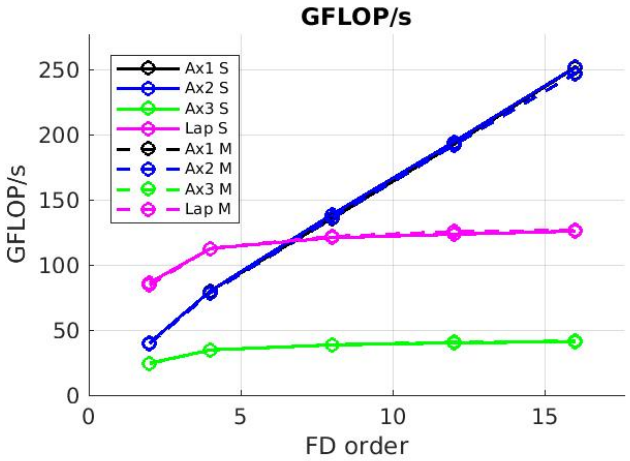
## Benchmark configuration

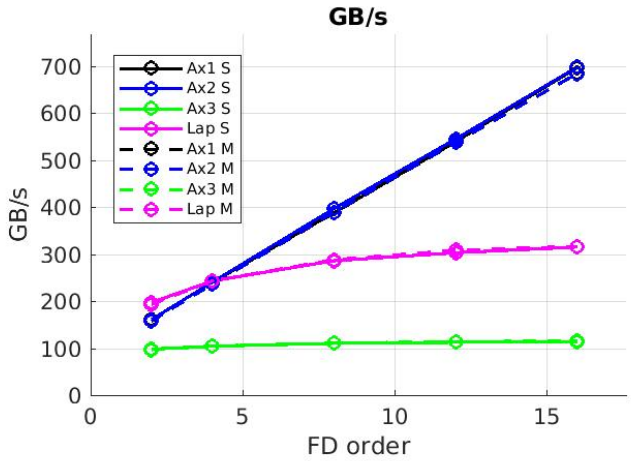
- 1 node with 32 threads
- 2 test modes
  - Baseline
  - CacheBlk
- Grid size 4 GB (1000 x 1000 x 1000 points)
- Reproduce results with `./script/testCase_FD_D2/hpcscanFD_D2.sh`
- Elapsed time about 2 minutes

# Test Case FD_D2 - Results [4]



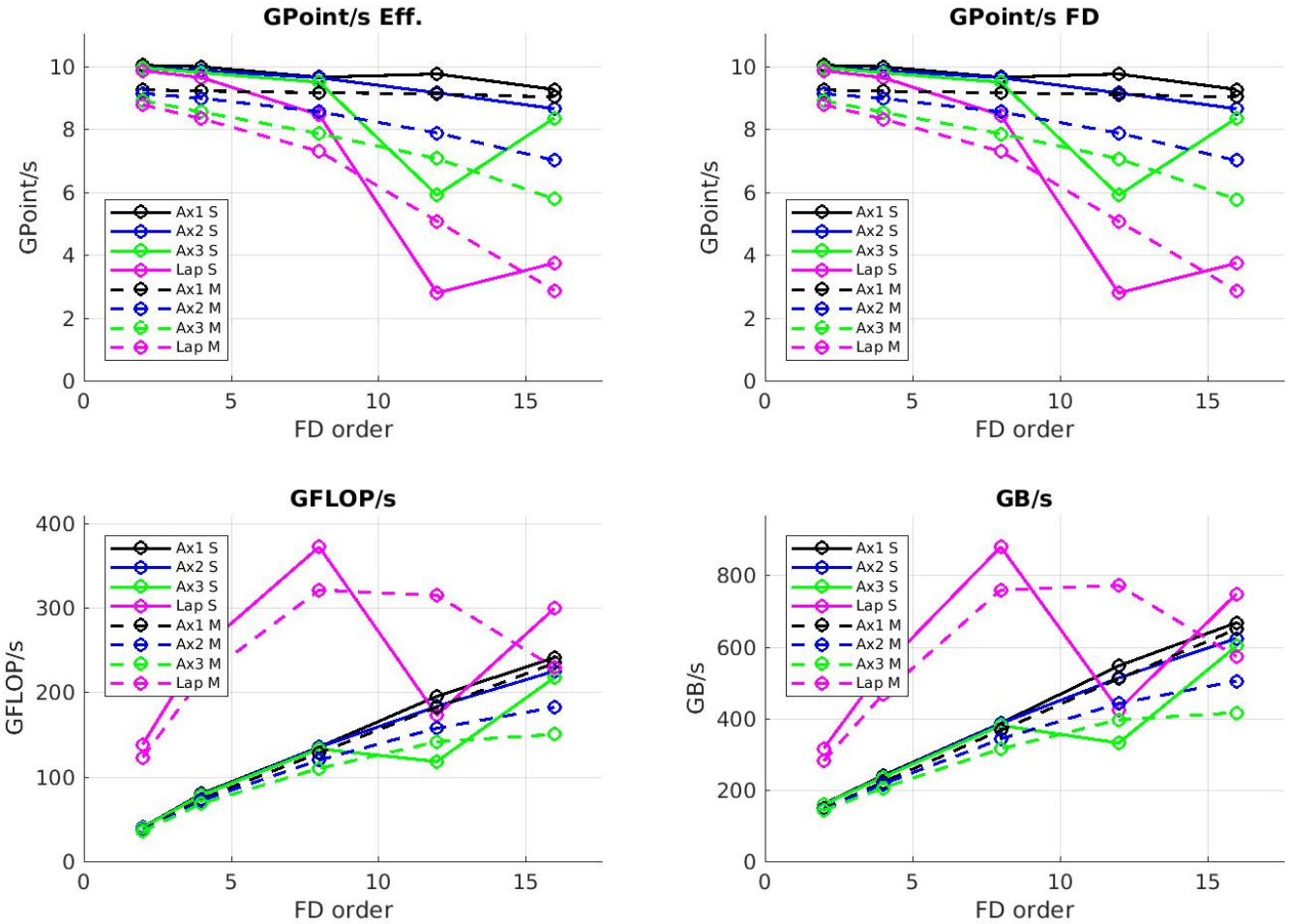




[4] Updated Sep 26, 2020

# Test Case FD_D2 - Results

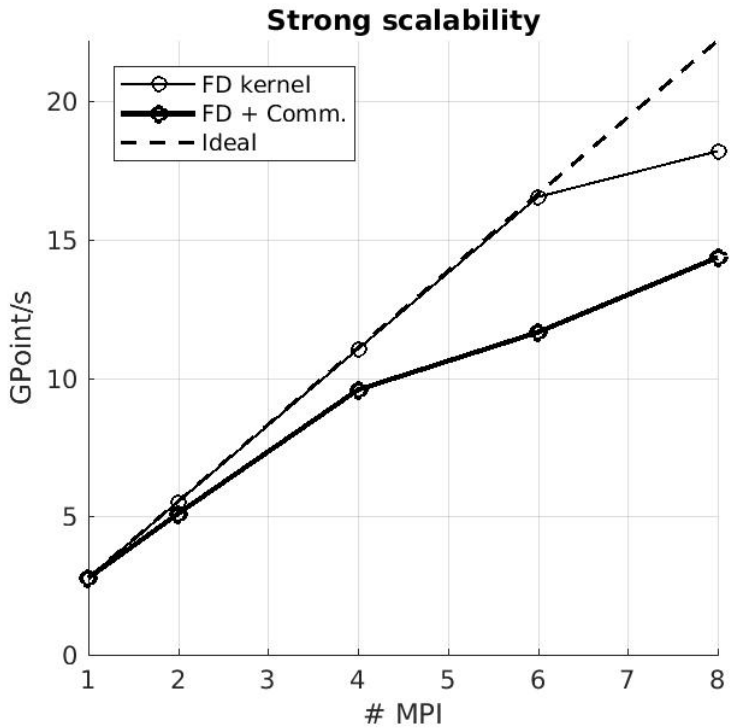
- 1 node with 32 threads / Cache blocking kernel [5]
- `./script/testCase_FD_D2/runSmallGridShaheen.sh & runMediumGridShaheen.sh`

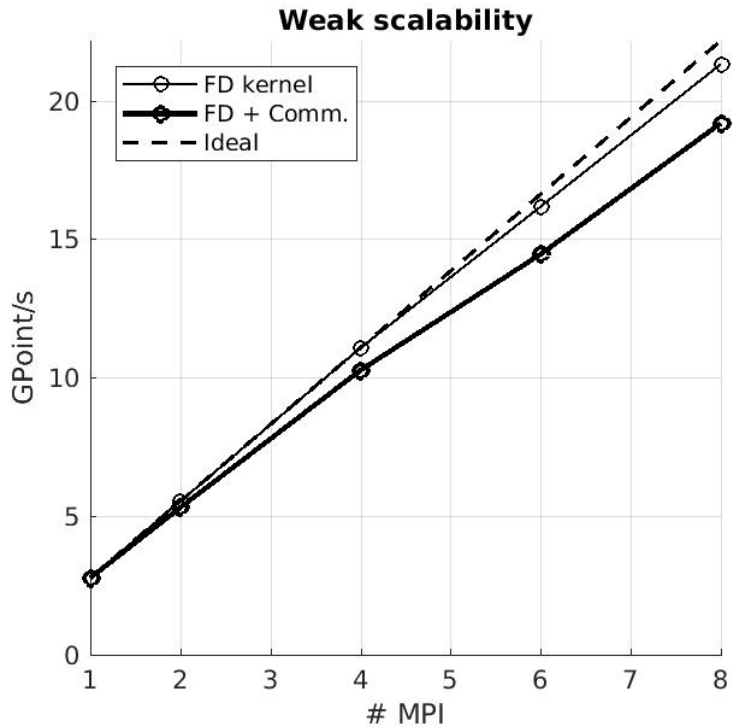



[5] Updated Sep 26, 2020

# Test Case FD_D2 - Results

- 1 to 8 nodes with 32 threads/node
- Baseline kernel [6]
- Strong scalabity: Grid 1000 x 1000 x 1000 (4 GB)
- Weak scalabity: Grids from 4 GB (1 proc) to 32 GB (8 proc)
- 3D Laplacian O8




[6] Updated Sep 26, 2020

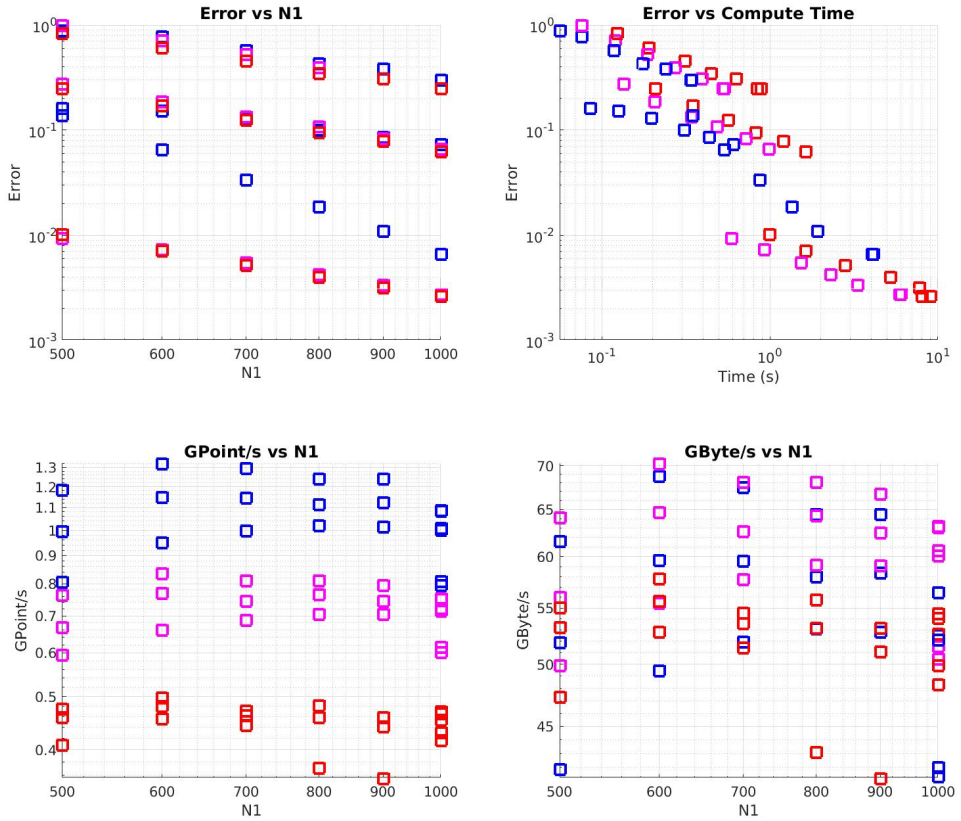
# Test Case FD_D2 - Summary

- Large benefit of cache blocking
- Significant effect of grid dimnsion and index (very bad performance for n3 without cache blocking)
- Min BW 50 GFLOP/s ($\partial^2_{x3}$ O2) = 2 % peak BW [apparent Mem. BW 150 GB/s]
- Max BW 370 GFLOP/s ($\Delta$ O8) = 16 % peak BW [apparent Mem. BW 900 GB/s]
- Apparent Mem. BW 150-900 GB/s (110-660 % Peak Mem. BW) = shows data in-cache effect
- Typical stencils of interest for geophysical applications
  - $\Delta$ O4 BW = 8-10 GPoint/s
  - $\Delta$ O8 BW = 7-9 GPoint/s
  - $\Delta$ O12 BW = 3-5 GPoint/s
- Parallel efficiency with 8 nodes 55 to 86 % (depends on workload on Shaheen)

# Content

# Test Case PropaAc2 - Results

- preliminary results [7]
- Eigen mode - 1D model
- FD: Black O2, Blue O4, Pink O8, Red O12 / Square=Baseline
- `./paramAnalysis/propaAccuracy/runMars.sh`




[7] Updated Nov 5, 2020

# Content

# Acknowledgements


- KAUST ECRC and KSL for access and support on Shaheen II & Ibex