hpcscan version 1.2
Performances on various architectures

December 2022

Content

Content

Introduction

This document presents a characterization of the computing nodes and interconnect on various systems. The full set of test cases embedded in `hpcscan` is used in various configurations.

List of test cases in this study

Test Case	Objectives	Remark
Memory	Assess memory bandwidth	Scalability analysis on a single node
Grid	Assess bandwidth of grid operations	Analyse effect of the grid size
Comm	Assess inter-node communication bandwidth	Analyse effect of subdomain decomposition
FD_D2	Assess FD spatial derivative computation bandwidth	Analyse effect of FD stencil order
Propa	Find optimal configuration for the wave propagator	Explore range of parameters
Propa	Scalability analysis of wave propagator on multiple nodes	Analyse effect of the FD stencil order

General settings

- All tests are performed in single precision
- Best performance is reported over 10 tries for each case (unless stated otherwise)
- Grids dimensions are 3D
- Grids sizes range from 500 MB up to 4 GB per node
- At maximum, 5 grids are allocated (test case Propa) $\Rightarrow$ **max. memory per node is 20 GB**
- At maximum, **8 computing nodes (with one MPI proc per node) are used**

Content

NEC SX-Aurora

- XX

Content

Test Case Memory - Description

Benchmark objective

Assess memory bandwidth

- Measure GByte/s and GPoint/s for simple operations on memory arrays
- Scalability analysis on a single node
- Get a reference to compare with for the following tests

Benchmark configuration

- Scalability on 1 node with 1 to 32 threads
- Baseline kernel
- Array size 4 GB
- Reproduce results with `./script/testCase_Memory/hpcscanMemory.sh`
- Total 10 configurations, Elapsed time about 4 minutes

Test Case Memory - NEC SX-Aurora TSUBASA XX [1]

[1] Results updated Dec XX, 2022

Content

Test Case Grid - Description

Benchmark objective

Assess bandwidth of grid operations

- Measure GByte/s and GPoint/s for simple and complex operations on 3D grids
- Analyse effect of the grid size

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`
- Total 2 configurations, Elapsed time less than 1 minute

Test Case Grid - NEC SX-Aurora TSUBASA XX [2]

ApplyBoundaryCondition performs at XXX/XXX GBytes (XX/XXX Gpoint/s)

[2] Results updated Dec XX, 2022

Content

Test Case Comm - Description

Benchmark objective

Assess inter-node communication bandwidth

Point to point communication

- Half-duplex BW with MPI_Send from proc X to proc 0
- Full-duplex BW with MPI_Sendrecv between proc X and proc 0

Collective communication

- Grid halos exchange (used in FD kernels) with MPI_Sendrecv
- Analyse effect of subdomain decomposition geometry

Benchmark configuration

- 8 nodes with 1 MPI/node & 32 threads/node
- Baseline kernel
- Grid size 4 GB (1000 x 1000 x 1000 points)
- FD order O8
- Subdomain decomposition: 1x4x2, 1x2x4 & 2x2x2
- Reproduce results with `./script/testCase_Comm/hpcscanComm.sh`
- Total 3 configurations: Elapsed time less than 1 minute

Test Case Comm - NEC SX-Aurora TSUBASA XX [3]

[3] Updated Dec XX, 2022

Content

Test Case FD_D2 - Description

Benchmark objective

Assess FD spatial derivative computation bandwidth

- Directionnal derivatives
  - Axis 1, $W = \partial^2_{x1}(U)$
  - Axis 2, $W = \partial^2_{x2}(U)$
  - Axis 3, $W = \partial^2_{x3}(U)$
- Laplacian $W = \Delta(U)$
- Analyse effect of FD stencil order
- Try different implementations of FD computation

Benchmark configuration

- 1 node with 32 threads
- 2 test modes: Baseline & CacheBlk
- Grid size 4 GB (1000 x 1000 x 1000 points)
- FD orders 2, 4, 8, 12 & 16
- Reproduce results with `./script/testCase_FD_D2/hpcscanFD_D2.sh`
- Total 10 configurations: Elapsed time about 2 minutes

Test Case FD_D2 - NEC SX-Aurora TSUBASA XX [4]

[4] Updated Dec XX, 2022

Content

Test Case Propa - Description

Benchmark objective

Find optimal configuration for the wave propagator regarding accuracy/cost

- Physical problem: 3D domain, size 130 λ in all direction, tmax = 11 periods
- Explore range of grid sampling and time step
- Explore range of FD order
- Try different implementations of the propagator

Benchmark configuration

- 1 node with 32 threads
- Test mode CachBlk
- 2 propagator implementations: Ac2Standard and Ac2SplitComp
- FD orders 4, 8 & 12
- Time step 100, 50 and 10% of stability time step
- Grid size from 500x500x500 (500 MB) to 1000x1000x1000 (4 GB)
- nt from 101 to 2311 (depending of the configuration) & ntry = 4
- Reproduce results with `./script/testCase_Propa/paramAnalysis/hpcscanPropaParamAnalysis.sh`
- Total 108 configurations: Elapsed time about 12 hours

Test Case Propa - NEC SX-Aurora TSUBASA XX [5]

Blue=FD O4, Pink=FD O8,Red=FD O12 / Square=Ac2Standard, Cross=Ac2SplitComp

[5] Updated Dec XX, 2022

Test Case Propa - Description

Benchmark objective

Scalability analysis of the wave propagator on multiple nodes

- Strong and weak scalability
- Analyse effect of the FD stencil order

Benchmark configuration

- From 1 node to 8 nodes with 32 threads/node
- Test mode CachBlk (best config from test case FD_D2)
- Propagator implementation Ac2Standard (best config from previous test case)
- FD orders 4, 8 & 12
- Subdomain decomposition geometry is best config obtained in test case Comm
- Strong scalability: Grid size 1000x1000x1000 (4 GB)
- Weak scalability: Grid size from 1000x1000x1000 (4 GB) to 1000x4000x2000 (32 GB)
- nt = 100
- Reproduce results with `./script/testCase_strongWeakScalability/hpcscanPropaStrongWeakScalability.sh`
- Total 30 configurations: Elapsed time about 1h 15min

Test Case Propa - NEC SX-Aurora TSUBASA XX [6]

[6] Updated Dec XX, 2022

Content

Summary

Test Case Memory

- Highest memory BW for Add+Update 118 GB/s [9.8 GPoint/s] (86 % of peak BW)
- Lowest memory BW for Fill 54 GB/s [13.5 GPoint/s] (40 % of peak BW)
- All cores (32 threads) are required to reach best perf. on the node

Test Case Grid

- Highest memory BW for L1Err, SumAbs, SumAbsDiff, GetMin & GetMax 125 GB/s (92 % of peak BW)
- In terms of GPoint/s: 15 for L1Err & SumAbsDiff and 30 for SumAbs, GetMin & GetMax
- Lowest memory BW for Fill 54 GB/s [13.5 GPoint/s] (40 % of peak BW)
- Minor effect of the grid size on the performance
- Results are consistent with the test case Memory

Summary

Test Case Comm

- Half-duplex BW about 8.6 GByte/s [2.1 GPoint/s]
- Full-duplex BW about 13.4 GByte/s [3.3 GPoint/s]
- Communication BW is similar between 8 nodes (no unbalance observed)
- Grid halos exchange: large effect of subdomain decomposition geometry
  - Best BW: 1x4x2 or 1x2x4 with 13.6 GByte/s [3.4 GPoint/s]
  - Worst BW: 2x2x2 with 1.8 GByte/s [0.5 GPoint/s]

Summary

Test Case FD_D2

- Max. 10 GPoint/s for ∂^2_{x1} for all FD orders
- Large effect of Cache Blocking
  - Max 950 GByte/s for Δ O12 with Cache Blocking (x7 memory BW)
  - Max 300 GByte/s for Δ O12 Baseline (speed-up 3.2 with Cache Blocking)
  - GFlop/s increases for all derivatives with the FD order (except for Δ O16)
  - Cache Block size is n2=4, n3=16 and full size for n1 (optimized for Shaheen [a])
- Max. 390 GFlop/s for Δ O12 with Cache Blocking (17 % peak GFlop/s)
- Lowest GFlop/s observed for ∂^2_{x3}

[a] Etienne, V., et al. "High-performance seismic modeling with finite-difference using spatial and temporal cache blocking." Third EAGE Workshop on High Performance Computing for Upstream. Vol. 2017. No. 1. European Association of Geoscientists & Engineers, 2017.

Summary

Test Case Propa - Parametric analysis

- Accuracy
  - Increases with FD orders, number of grid points and number of time steps
  - Stability time step is too large for high order FD stencils to reach optimal convergence
- Optimal config to reach error less than 1% with minimal elapsed time
  - FD O8 - Cache Blocking - Standard propagator implementation with N=500 and 10% stability time step
  - This schemes perform at 6.2 GPoint/s and 720 GByte/s (elapsed time 22 s)
  - To reach similar accuracy, FD O4 requires N=900 and an elapsed time 203 s (9.2 times longer than O8)
- Implementation algorithm
  - Splitting computation with a separate Laplacian computation slows down the propagator by 2

Summary

Test Case Propa - Scalability

- Strong scalability
  - Speed-up is sub-linear and gets worst as the number of nodes increases
  - Parallel efficiency decreases with FD order
  - Parallel efficiency on 8 nodes: 77% for O4, 64% for O8 and 60% for O12
- Weak scalability
  - Speed-up is linear
  - Parallel efficiency decreases with FD order
  - Parallel efficiency on 8 nodes: 90% for O4, 83% for O8 and 80% for O12

Content

Acknowledgements

- XX