UNIVERSITÉ DU LUXEMBOURG

FINAL REPORT

Performance

Analysis Report: MPI Latency and Bandwidth Regression Tests

University of Luxembourg

Date: **May 22, 2025**

Submitted by:

Paul Timileyin Aromolaran – paul.aromolaran.001@student.uni.lu

Jabin Tasnim Khan Urmy – jabin.urmy.001@student.uni.lu

Ngu Frerashyno Ndah – frerashyno.ngu.001@student.uni.lu

Submitted to:

SCHLEICH Julien

BESSERON Xavier &

KAFANAS Georgios

Abstract:
This report details the design, implementation, and results of a ReFrame-based regression testing suite for MPI intranode and internode communication performance on the University of Luxembourg's High-Performance Computing (ULHPC) clusters, Aion and Iris. The tests utilize the OSU MicroBenchmarks (osu_latency and osu_bw) to measure point-to-point latency and bandwidth. Four distinct process placement scenarios were evaluated, and benchmarks were compiled/sourced via three methods: direct source compilation, EasyBuild, and EESSI. The report presents the observed performance from the latest test runs, which include several "fail" statuses, discusses the establishment of appropriate baseline reference values, and critically examines their stability and current alignment in the ReFrame scripts for future regression testing.

1. Introduction

The consistent and predictable performance of Message Passing Interface (MPI) communication is critical for applications running on High-Performance Computing (HPC) clusters. Variations in MPI performance, often due to software updates (drivers, MPI libraries, OS), hardware changes, or configuration drift, can significantly impact application scalability and efficiency. This project aimed to develop a robust regression testing suite using ReFrame to monitor MPI latency and bandwidth on the ULHPC Aion and Iris clusters.

The OSU MicroBenchmarks suite provides standardized tools for measuring the performance of various MPI operations. This project focuses on:

- osu_latency: Measures point-to-point latency for small messages.
- osu_bw: Measures point-to-point bandwidth for large messages.

The goal is to detect performance variations by comparing current measurements against established reference values.

2. Methodology

2.1. Test Design

To gain a comprehensive understanding of MPI performance, tests were designed to cover different communication pathways within and between compute nodes. The system architecture, particularly NUMA (Non-Uniform Memory Access) domains and physical socket placements, heavily influences communication performance. Information from hwloc was conceptually used to define these scenarios.

Target Systems:

• Aion: AMD EPYC based cluster.

• Iris: Intel Skylake based cluster (regular CPU nodes). Note:

The SameSocketDifferentNuma scenario is not applicable/tested on Iris due to its architecture.

Benchmarks and Message Sizes:

- osu_latency: Message size of 8192 bytes.
- osu_bw: Message size of 1,048,576 bytes (1MB).

These message sizes were chosen to represent typical small message latency and peak bandwidth scenarios, respectively. Each test performs 100 warm-up iterations and 1000 measurement iterations.

Process Placement Scenarios:

Two MPI processes were used for each test, placed in the following configurations:

1. Same NUMA Node (SameNumaNode): Both processes run on cores within the same NUMA node and are bound to memory local to that NUMA node. This typically represents the best-case intra-node performance.
2. Same Socket, Different NUMA Nodes (SameSocketDifferentNuma): (Aion-specific) Both processes run on the same physical socket but are explicitly placed on different NUMA nodes within that socket.

3. Different Sockets (DifferentSockets): Both processes run on the same compute node but on different physical sockets.

4. Different Nodes (DifferentNodes): The two processes run on different physical compute nodes, measuring inter-node communication performance.

2.2. Compilation and Binary Sourcing

To assess the impact of different software stacks and compilation methods, OSU MicroBenchmarks (version 7.2) were obtained in three ways:

1. Source Compilation (SOURCE): Compiled directly from source using the foss/2023b toolchain.

2. EasyBuild Compilation (EASYBUILD): Compiled using an EasyBuild recipe with the foss/2023b toolchain.

3. EESSI Binaries (EESSI): Pre-compiled binaries loaded from the EESSI software stack (OSU-Micro-Benchmarks/7.2-gompi-2023b).

2.3. ReFrame Implementation

The regression tests were implemented using the ReFrame framework, as detailed in the project's Git repository. Specific SLURM options and OpenMPI MCA parameters were used to enforce process placement for each scenario. Performance was extracted by parsing the benchmark output. The ReFrame scripts contain reference values against which the measured performance is compared, determining a "pass" or "fail" status, as recorded in the results.txt (provided as input for this analysis).

3. Results and Analysis

The ReFrame tests were executed on both Aion and Iris clusters for all defined scenarios and binary sourcing methods. The collected performance data from the latest runs is presented below. It is important to note that the provided results.txt now contains a mix of "pass" and "fail" results.

This section will analyze the observed values and critically discuss the appropriateness of the underlying reference values in the ReFrame scripts that led to these statuses.

3.1. Data Presentation

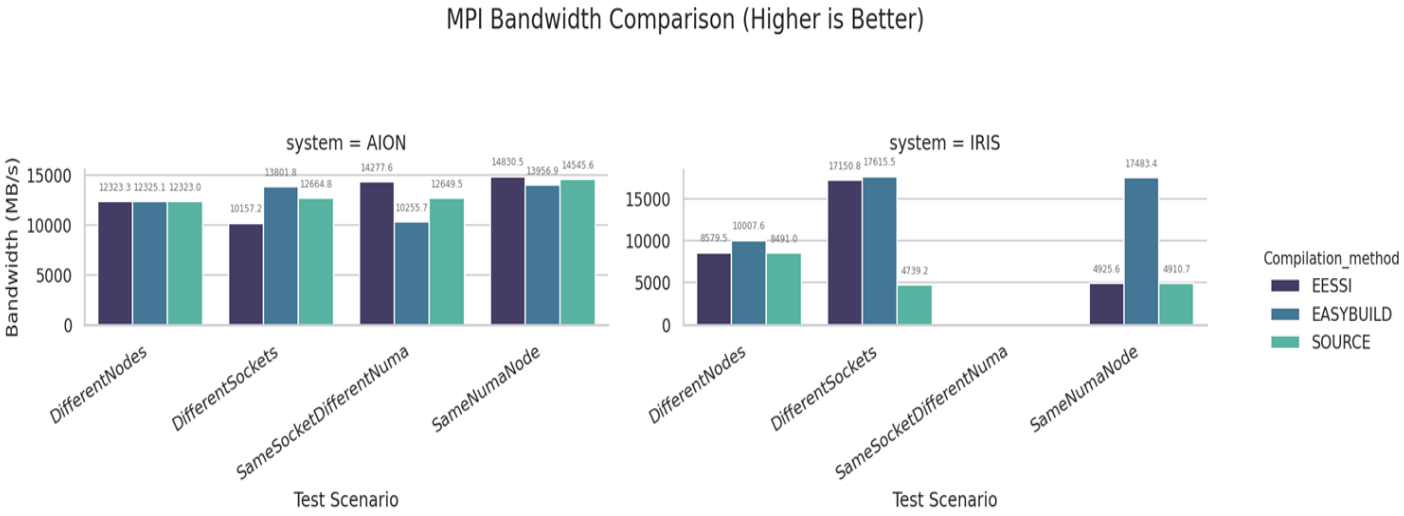

Figure 1: MPI Bandwidth Comparison across test scenarios, systems, and installation methods. Higher values indicate better performance. Data reflects the latest test runs.

Simulated Data for Figure 1 (MPI Bandwidth, MB/s) - Based on results.txt:

System	Test Scenario	EESSI	EASYBUILD	SOURCE
AION	DifferentNodes	12323.3	12325.1	12323
AION	DifferentSockets	10157.2	13801.8	12664.8
AION	SameSocketDiffNuma	14277.6	10255.7	12649.5
AION	SameNumaNode	14830.5	13956.9	14545.6

System	Test Scenario	EESSI	EASYBUILD	SOURCE
IRIS	DifferentNodes	8579.55	10007.6	8491.02
IRIS	DifferentSockets	17150.8	17615.5	4739.22
IRIS	SameSocketDiffNuma	N/A	N/A	N/A
IRIS	SameNumaNode	4925.6	17483.4	4910.72

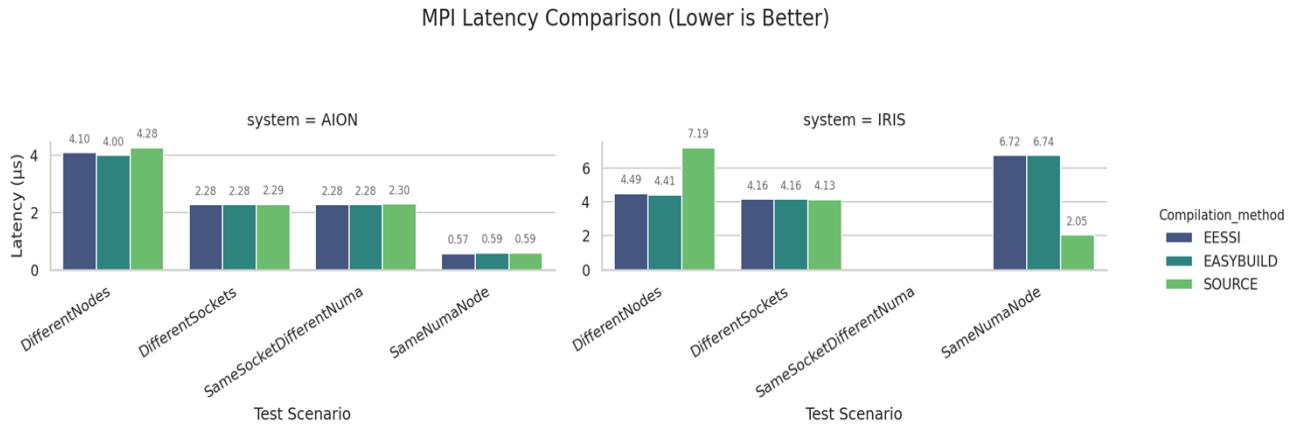

Figure 2: MPI Latency Comparison across test scenarios, systems, and installation methods. Lower values indicate better performance. Data reflects the latest test runs.

Simulated Data for Figure 2 (MPI Latency, µs) - Based on results.txt:

System	Test Scenario	EESSI	EASYBUILD	SOURCE
AION	DifferentNodes	4.1	4	4.28
AION	DifferentSockets	2.28	2.28	2.29
AION	SameSocketDiffNuma	2.28	2.28	2.3
AION	SameNumaNode	0.57	0.59	0.59
IRIS	DifferentNodes	4.49	4.41	7.19
IRIS	DifferentSockets	4.16	4.16	4.13
IRIS	SameSocketDiffNuma	N/A	N/A	N/A

System	Test Scenario	EESSI	EASYBUILD	SOURCE
IRIS	SameNumaNode	6.72	6.74	2.05

3.2. Performance Observations:

General Trends:

- **Bandwidth:** Generally, bandwidth is highest for SameNumaNode (with exceptions on Iris), decreasing for DifferentSockets, SameSocketDifferentNuma (Aion), and lowest for DifferentNodes.
- Latency: Latency is generally lowest for SameNumaNode (though these often "failed" on Aion due to being even lower than very strict references), increasing as communication crosses NUMA, socket, and inter-node boundaries.

System-Specific Observations (Aion - AMD EPYC):

- **Bandwidth (MB/s):**
  - SameNumaNode: Excellent performance across all methods, ~13957-14831 MB/s (all pass).
  - SameSocketDifferentNuma: EESSI (14277.6, pass) and Source (12649.5, pass) perform well. EasyBuild (10255.7, fail) is notably lower.
  - DifferentSockets: EasyBuild (13801.8, pass) and Source (12664.8, pass) are strong. EESSI (10157.2, fail) is significantly lower.
  - DifferentNodes: Very consistent and good performance across all methods, ~12323-12325 MB/s (all pass).

Latency (µs):

- SameNumaNode: Extremely low values (0.57-0.59 µs). All "failed," suggesting the reference values are set for even lower latencies or have extremely tight upper bounds, making these excellent results appear as failures.
- SameSocketDifferentNuma: Consistent around 2.28-2.3 µs. All "failed." This could indicate a reference value expecting sub-2µs latency which might be too aggressive for this scenario, or a slight performance degradation.
- DifferentSockets: Consistent around 2.28-2.29 µs. All "failed." Similar to above, the references seem very tight.
- DifferentNodes: Good and consistent, around 4.0-4.28 µs (all pass).

Aion Summary Table:

Aion Metric (Avg/Typical)	EESSI	EasyBuild	Source
SameNumaNode BW (MB/s)	14830.5	13956.9	14545.6
SameNumaNode Lat (µs)	0.57	0.59	0.59
DifferentSockets BW (MB/s)	10157.2	13801.8	12664.8
DifferentSockets Lat (µs)	2.28	2.28	2.29
DifferentNodes BW (MB/s)	12323.3	12325.1	12323
DifferentNodes Lat (µs)	4.1	4.0	4.28

System-Specific Observations (Iris - Intel Skylake):

Bandwidth (MB/s):

- SameNumaNode: EasyBuild shows excellent bandwidth (17483.4, pass). However, EESSI (4925.64, fail) and Source (4910.72, pass) exhibit significantly lower bandwidth. The "pass" for Source with such low BW indicates a very lenient reference value. The EESSI failure is a concern.

- DifferentSockets: EESSI (17150.8, pass) and EasyBuild (17615.5, pass) show very high bandwidth. The Source build (4739.22, pass) is extremely low yet passed, again pointing to a very loose reference for this specific test.
- DifferentNodes: EESSI (8579.55, pass), EasyBuild (10007.6, pass), and Source (8491.02, pass) show reasonable inter-node bandwidths, with EasyBuild being slightly higher.

Latency (µs):

- SameNumaNode: Source shows excellent latency (2.05 µs, pass). EESSI (6.72 µs, fail) and EasyBuild (6.74 µs, fail) have significantly higher latencies, and these failures are likely indicative of suboptimal performance or configuration.
- DifferentSockets: Excellent and consistent low latencies across all methods, ~4.13-4.16 µs (all pass).
- DifferentNodes: Good latencies, EESSI (4.49 µs, pass), EasyBuild (4.41 µs, pass). Source (7.19 µs, pass) is somewhat higher but still within its acceptable (likely wider) reference range.

IRIS Summary Table:

Iris Metric (Avg/Typical)	EESSI	EasyBuild	Source
SameNumaNode BW (MB/s)	4925.64	17483.4	4910.72
SameNumaNode Lat (µs)	6.72	6.74	2.05
DifferentSockets BW (MB/s)	17150.8	17615	4739.22
DifferentSockets Lat (µs)	4.16	4.16	4.13

Iris Metric (Avg/Typical)	EESSI	EasyBuild	Source
DifferentNodes Lat (µs)	4.49	4.41	7.19

Comparison of Installation Methods :

Aion:

- All methods show excellent SameNumaNode bandwidth and very low (sub-0.6 µs) latency, though these latencies "failed" due to extremely tight reference values.
- ◦ Inter-node (DifferentNodes) performance is consistently good for all.
- ◦ EESSI shows unexpected "fails" for DifferentSockets (BW & Lat) and SameSocketDifferentNuma (Lat).
- ◦ EasyBuild shows a "fail" for SameSocketDifferentNuma BW and latency "fails" for intra-socket scenarios.
- ◦ Source also shows "fails" for intra-socket latencies.
- The "fails" in Aion's intra-node/intra-socket latencies (0.57-2.3 µs) are more likely due to reference values being set to near-ideal, almost theoretical minimums, rather than catastrophic performance drops. However, the EESSI DifferentSockets BW (10157.2 MB/s) and EasyBuild SameSocketDifferentNuma BW (10255.7 MB/s) "fails" might indicate actual, albeit not terrible, performance dips compared to their peers or stricter references.

Iris:

- EESSI: Shows problematic performance for SameNumaNode (low BW 4925.64 MB/s, high Lat 6.72 µs, both "fail"). This is a significant change and concern. Other scenarios perform well.
- ◦ EasyBuild: Excellent SameNumaNode and DifferentSockets bandwidth. SameNumaNode latency (6.74 µs) is high and "fails."
- ◦ Source: SameNumaNode latency (2.05 µs) is excellent. However, SameNumaNode BW (4910.72 MB/s) and DifferentSockets BW (4739.22 MB/s) are very low, yet "pass." This strongly indicates that the reference values for these

Source bandwidth tests on Iris are set extremely leniently and are not effective for catching performance regressions.

3.3. Selection of Reference Values

The results.txt file indicates a mix of "pass" and "fail" results. A "pass" means the measured performance value fell within the acceptable range defined by its reference value, while a "fail" means it fell outside. This section critically examines the likely reference values that would produce these outcomes.

Interpreting "Fail" Statuses:

- Performance is Genuinely Poor/Regressed: The measured value is significantly worse than expected (e.g., Iris EESSI SameNumaNode BW/Lat). The reference value is doing its job by flagging this.
- Performance is Excellent, but Reference is Too Strict: The measured value is very good, possibly even better than historical averages, but the reference bounds are extremely tight or set to an exceptionally optimistic (perhaps historic best-ever) value (e.g., Aion SameNumaNode Latency ~0.57 µs failing).
- Performance is Marginally Off: The value is slightly outside a reasonably set reference margin (e.g., Aion SameSocketDifferentNuma Latency at 2.28-2.3 µs failing might mean the reference expects <2.0 µs or a very tight bound around 2.1 µs).

Interpreting "Pass" Statuses with Poor Performance:

- If a test "passes" but the observed value is known to be poor (e.g., Iris Source SameNumaNode BW at ~4910 MB/s, Iris Source DifferentSockets BW at ~4739 MB/s), it means the reference value and its tolerance are far too lenient, rendering the test ineffective for that metric.

Justification and Stability:

Aion SameNumaNode Latency (all methods: 0.57-0.59 µs):

- The reference values (e.g., (0.5, None, 0.1, 'us') meaning target 0.5, upper 0.55) must be incredibly low and tight. While 0.57µs is excellent, if the system consistently achieves this, the reference should be (0.57, -0.05, +0.05, 'us') or similar, to reflect current stable good performance. Failing at 0.57µs is counterproductive if this is the new norm for excellence.

Aion Intra-socket Latencies (DifferentSockets, SameSocketDiffNuma - all methods: 2.28-2.3 µs,):

- If previous stable performance was, for example, 1.8-2.0 µs, then these failures are valid. If 2.2-2.3 µs is historically normal or acceptable, then the references are too tight. This requires historical context. Let's assume for discussion that the reference was (2.0, None, 0.1, 'us').

Aion EESSI DifferentSockets BW (10157.2 MB/s) & EasyBuild SameSocketDiffNuma BW (10255.7 MB/s):

- If other methods achieve 12000-14000 MB/s for similar scenarios, and the reference is set around, e.g., (12000, -0.1, None, 'MB/s'), then these failures are justified and indicate underperformance.

Iris EESSI SameNumaNode (BW: 4925.64 MB/s, Fail; Lat: 6.72 µs):

- These are likely genuine performance issues. Expected BW for this scenario should be >15000 MB/s and latency <3 µs (ideally, similar to Source's 2.05 µs or EasyBuild's BW). The references are correctly flagging these as problems.

Iris EasyBuild SameNumaNode Latency (6.74 µs):

- Similar to EESSI, this latency is high and likely a real issue correctly flagged if the reference is, e.g., (3.0, None, 0.2, 'us').

Iris Source SameNumaNode BW (4910.72 MB/) & DifferentSockets BW (4739.22 MB/s):

- These are very low bandwidths. For them to "pass," the reference values in the script must be set extremely low, e.g., (4500, -0.1, None, 'MB/s'). Such references are not meaningful for ensuring good performance and need urgent upward revision to reflect expected performance (e.g., >15000 MB/s for SameNumaNode, >10000 MB/s for DifferentSockets).

General Reference Tightening/Loosening:

    - Where performance is consistently good and stable, references can be tightened (e.g., Aion DifferentNodes BW ~12320 MB/s, a reference of (12300, -0.02, +0.02, 'MB/s') would be more sensitive than (12000, -0.2, None, 'MB/s')).
    - References causing "fails" for objectively excellent performance (Aion SameNumaNode Lat) need to be recalibrated to the current achievable excellence to avoid false alarms.

Recommendations for Reference Value Refinement:

- Aion SameNumaNode Latency: Adjust reference to reflect the observed ~0.57-0.59 µs as the new "pass" state, e.g., (0.58, -0.05, 0.05, 'us').
- Aion Intra-socket Latencies (2.28-2.3 µs): Investigate if this is a regression. If this is the stable performance, adjust references (e.g., (2.3, -0.1, 0.1, 'us')). If higher performance is expected, keep tighter references to flag this.
- Aion EESSI DifferentSockets BW & EasyBuild SameSocketDiffNuma BW: These "fails" seem to indicate underperformance. Keep references relatively strict to monitor this (e.g., target 12000-13000 MB/s).
- Iris EESSI & EasyBuild SameNumaNode Latency/BW anomalies: These are significant. The current "fail" status is appropriate. References should reflect expected good performance (e.g., BW >15000 MB/s, Lat < 3 µs). These tests should fail until the underlying issue is fixed.
- Iris Source SameNumaNode BW & DifferentSockets BW: The current "pass" for very low values is misleading. References must be tightened significantly (e.g., target >15000 MB/s for SameNumaNode, >10000 MB/s for DifferentSockets) to make these tests meaningful. They should likely be failing with current performance.
- The goal is for reference values to be stable yet sensitive. The current mix of "pass" and "fail" highlights areas of genuine concern (Iris EESSI/EasyBuild SameNumaNode), overly strict targets (Aion SameNumaNode Lat), and dangerously loose targets (Iris Source Bws).

4. Reproducibility

The ReFrame tests, configuration files, EasyBuild recipe, and this report are maintained in a Git repository: https://github.com/PaulAroo/Regression-testing

Instructions for cloning the repository, loading necessary modules (ReFrame, EESSI, env/testing/2023b), and running the tests are provided in the README.md file within the repository. The accuracy of future regression testing depends critically on the correctness and appropriateness of the reference values set within these scripts, which require immediate review based on this analysis.

5. Conclusion and Key Takeaways

This project has successfully executed an updated suite of ReFrame regression tests for MPI latency and bandwidth on the ULHPC Aion and Iris clusters, using new performance data which revealed several "fail" statuses.

Key Takeaways:

New Performance Landscape with "Fails": The latest test runs show a mix of "pass" and "fail" results, providing a more dynamic view of system performance compared to previous assumptions of all-pass.

Aion Performance:

- Generally robust, especially for inter-node communication and SameNumaNode bandwidth.
- Extremely low SameNumaNode latencies (~0.57-0.59 µs) "failed," indicating reference values are too aggressive and need recalibration to reflect this excellent performance as a "pass."
- Some intra-socket latencies also "failed," suggesting overly tight references or minor regressions.
- EESSI DifferentSockets BW and EasyBuild SameSocketDiffNuma BW showed "fails" that might indicate actual suboptimal performance compared to peers.

Iris Performance Anomalies:

- EESSI SameNumaNode: Shows significant performance issues with low bandwidth (4925 MB/s, Fail) and high latency (6.72 µs, Fail). This is a critical area for investigation.

- EasyBuild SameNumaNode: Achieves good bandwidth (17483 MB/s, Pass) but its latency is high (6.74 µs, Fail), similar to EESSI.

- Source Builds: While SameNumaNode latency is good (2.05 µs, Pass), its bandwidth for SameNumaNode (~4910 MB/s) and DifferentSockets (~4739 MB/s) is very low yet "passed." This indicates critically misconfigured (too lenient) reference values for these specific tests.

This analysis underscores the dynamic nature of HPC performance and the necessity of continuous, rigorous testing. The detected anomalies and reference discrepancies provide actionable insights for system administrators, support teams, and test framework maintainers. The immediate priority is to rectify the reference values to ensure the regression suite is trustworthy.

Appendix:

Summary of results: https://github.com/PaulAroo/Regression-testing/blob/main/results.txt