Collective Operations with Notified Communication in Shared Windows

Muhammed Abdullah Al Ahad, Christian Simmendinger, Roman Iakymchuk, Erwin Laure and Stefano Markidis

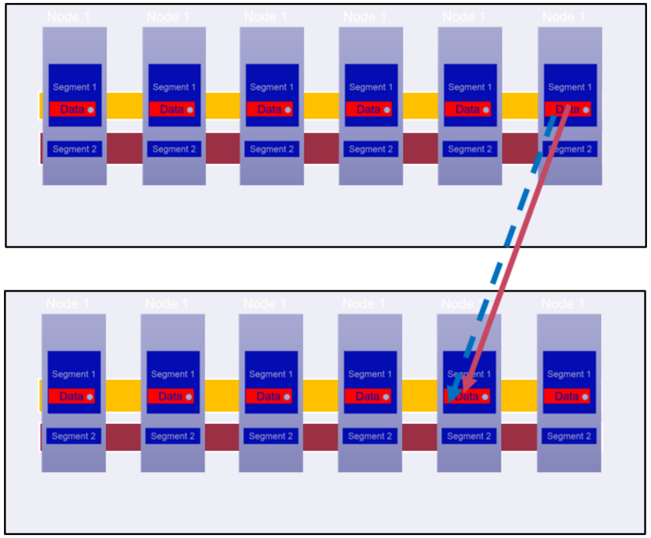
This project is funded from the European Union's Horizon 2020 Research and Innovation programme under Grant Agreement no. 671602.

Notified communication in shared memory

Context: Extension of GASPI notified communication for shared memory.

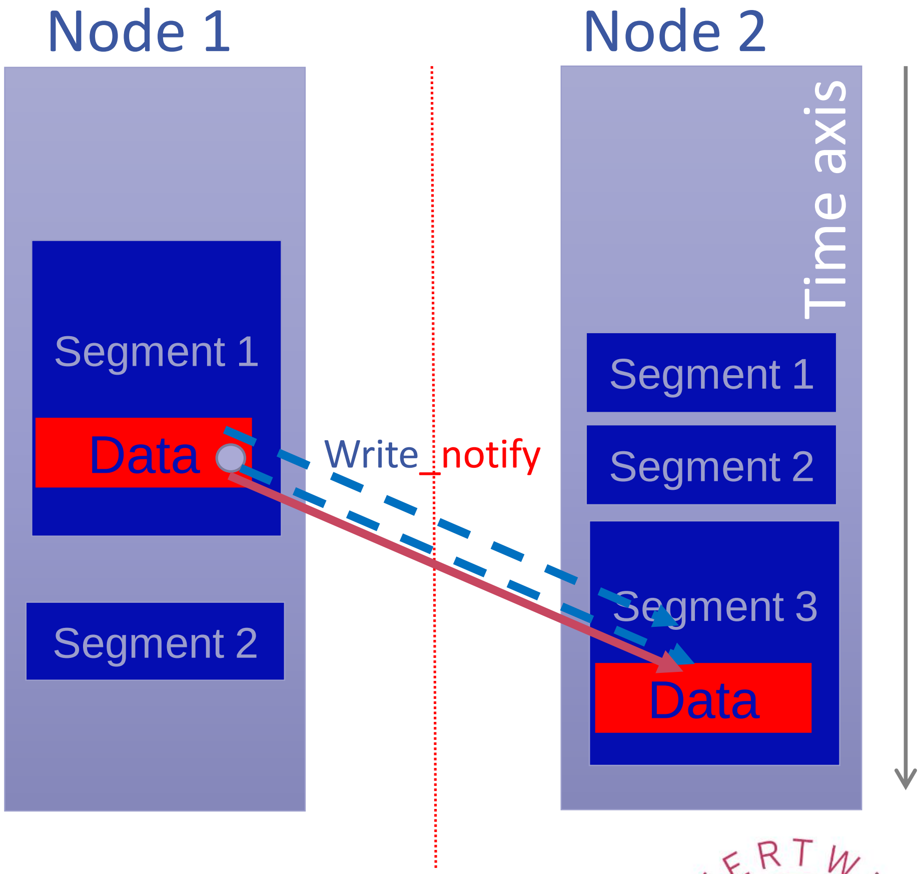
- Collectives as testbed for evaluating programmability and performance.

- Allreduce
  - Pipelined ring
  - Dissemination
- AllgatherV



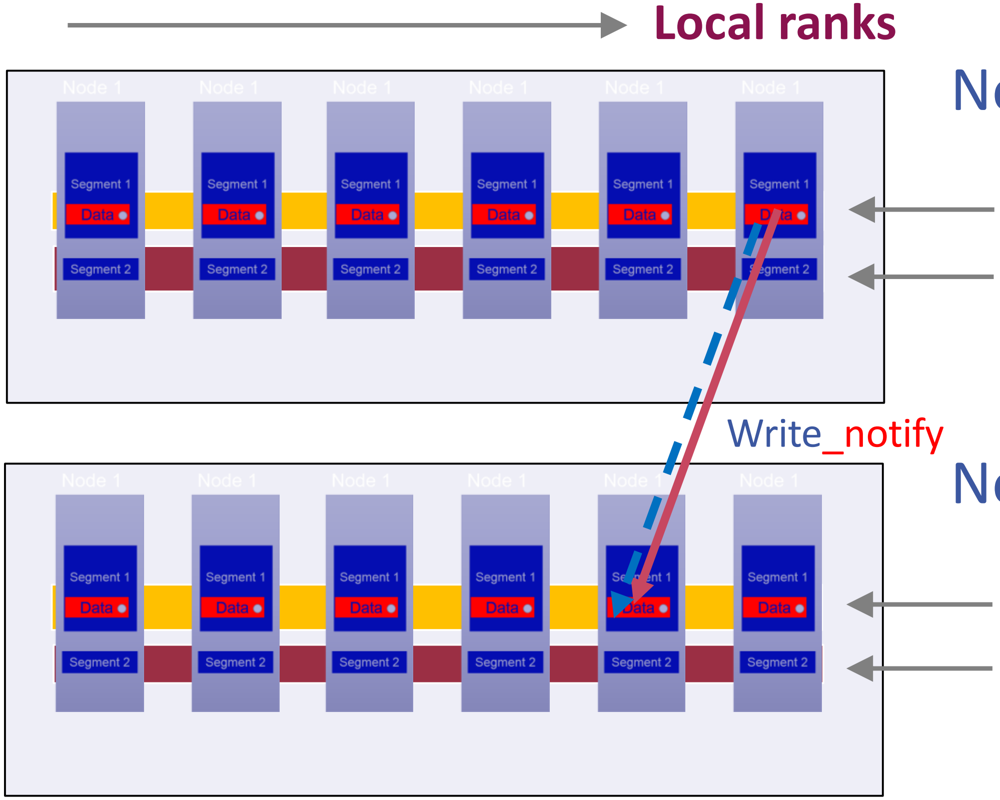
Synchronisation with Notifications

- **Designated communication areas in GASPI are called memory segments.**
- **Segments are freely configurable.**
- **GASPI uses notified write and read (weak synchronization).**
- **GASPI also supports bundled write and notify as a single call: write_notify (and read_notify)**
- **GASPI is interoperable with MPI and allows incremental porting**



Synchronisation with Notifications

- Q: How can we extend notified communication for shared segments ?
  A: Use local notifications in shared data segments and share remote notifications across local ranks.



Local ranks

Node 1

Shared data segment

Shared notification segment

Node 2

Shared data segment

Shared notification segment

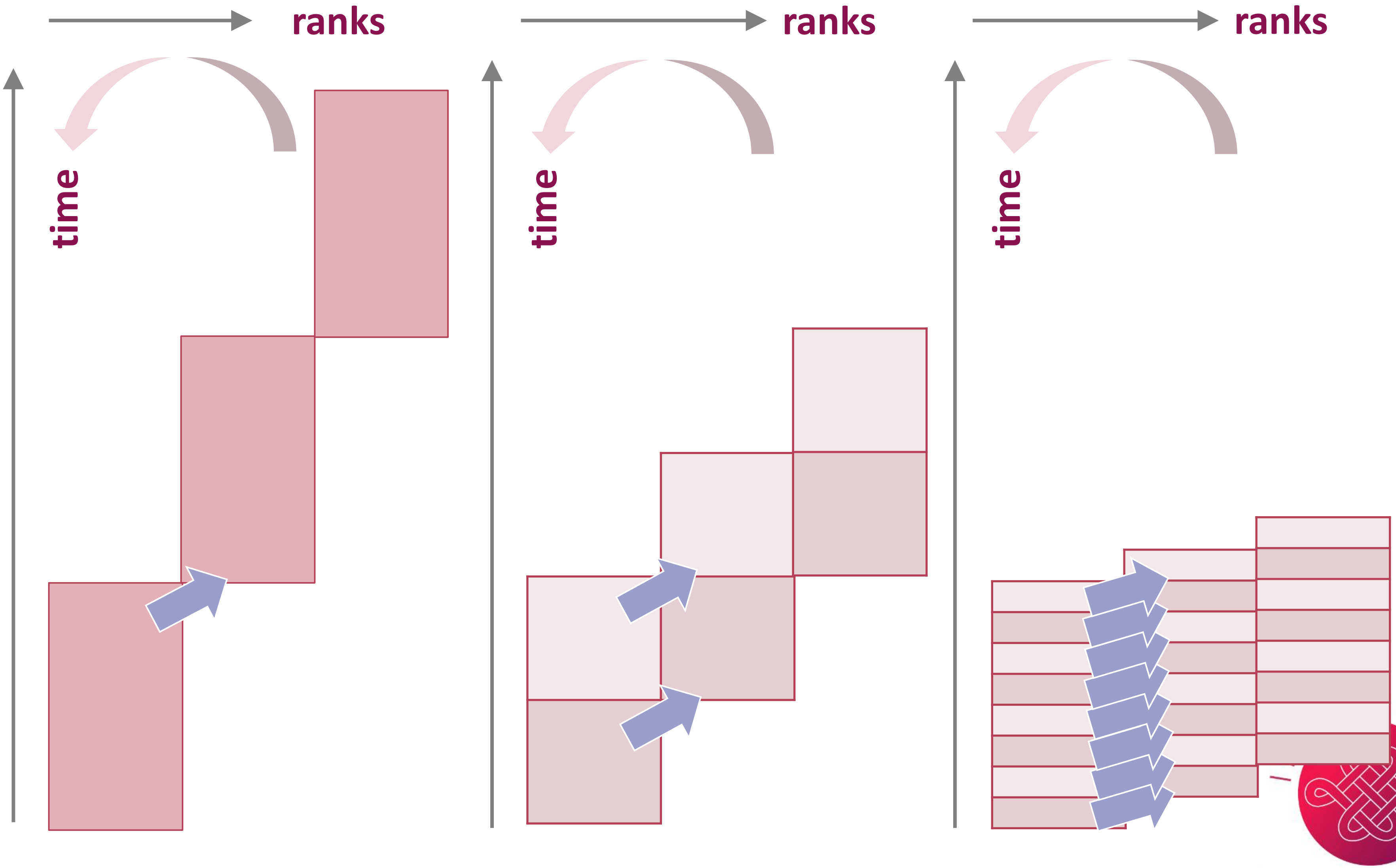
Testbed – Collectives in Shared Windows

Testbed for notified communication - Collectives.

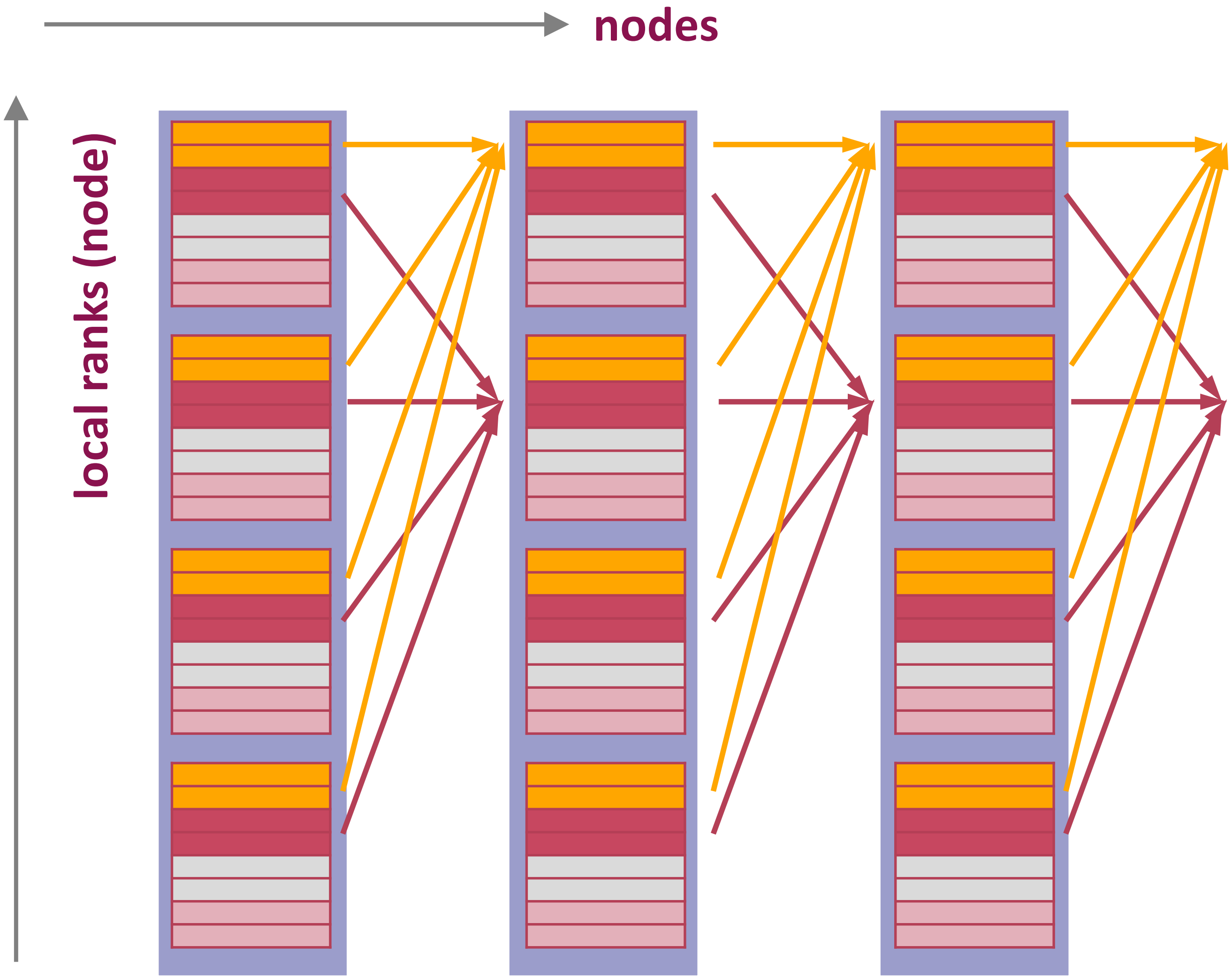
Allreduce with GASPI shared segments

- **Pipelined Ring for large message sizes**
  - **2-stage pipelined process, reduce + broadcast**
  - **Highly parallel pipeline for fast ramp-up (latency ~2(n-1))**
  - **Shared memory for node internal communication**
  - **Leverage available bandwidth / compute for reduce**

Allreduce / Allgather(V) - Pipelined Ring

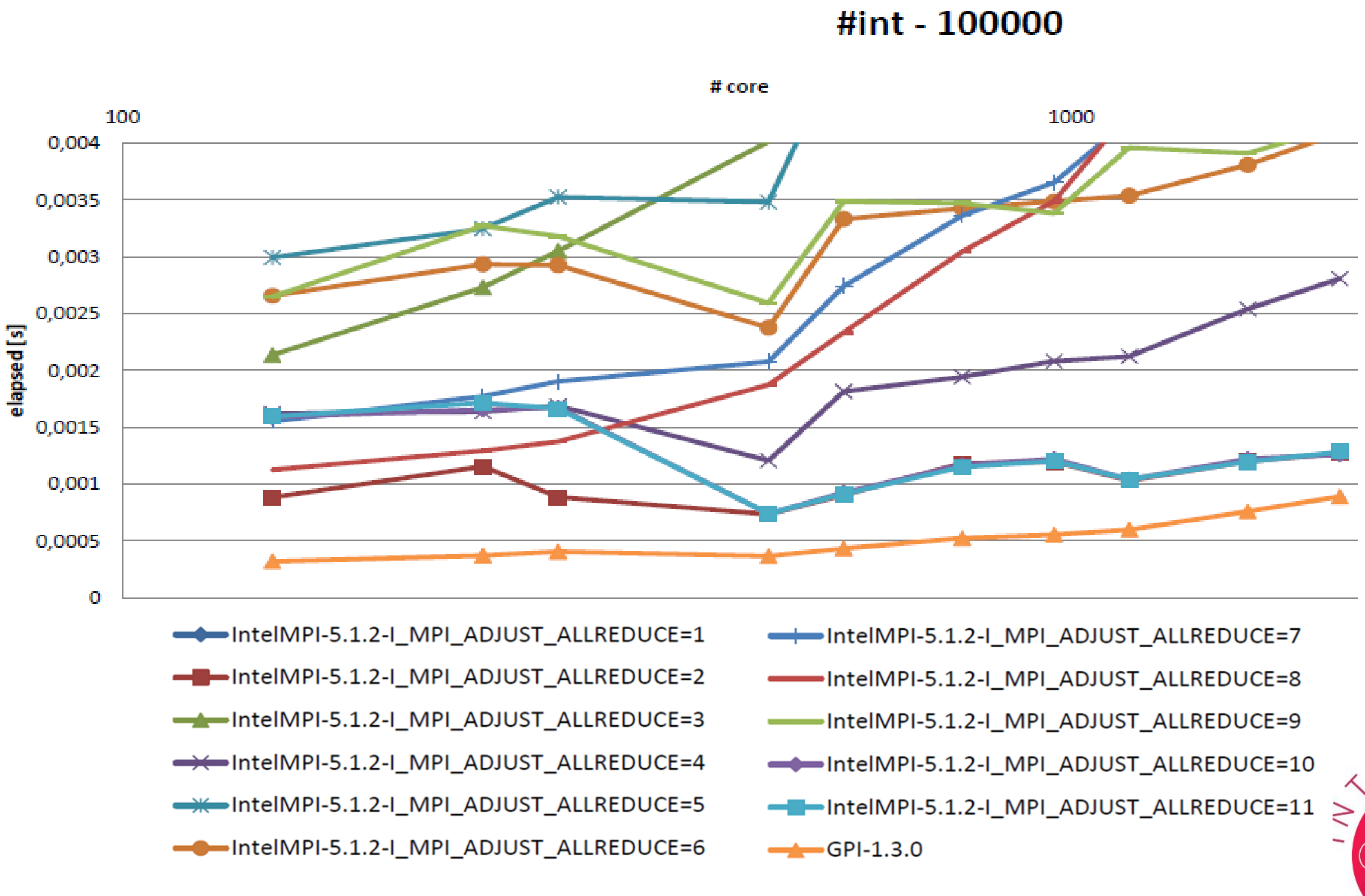


Allreduce / Allgather(V) - Load Distribution across Local Ranks

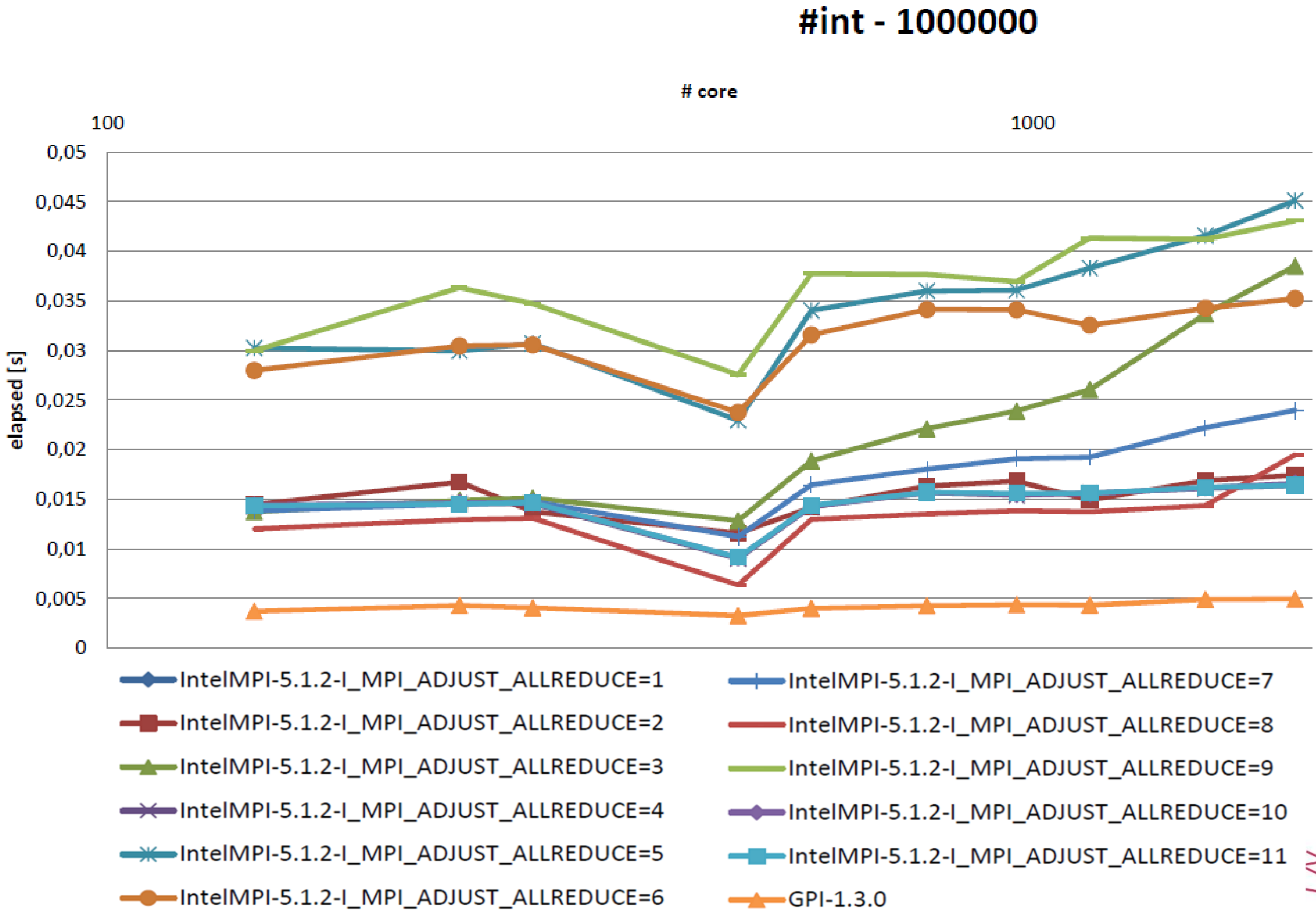


All ranks participate in reducing and forwarding data at any point in time

Allreduce – Pipelined Ring



Allreduce – Pipelined Ring



Testbed – Collectives in Shared Windows

Testbed for notified communication - Collectives.

Allreduce with GASPI shared segments

- Dissemination algorithm
  - Log2(m) dissemination steps, where m <= 2^n.
  - Hide communication of additionally required dissemination steps for m != 2^n
  - Decompose number of ranks as

$$n = \sum_{k=l}^{0} c_k 2^k, \text{ with } c_k \in \{0, 1\}$$

  - In step j, rank i writes additional out of band msg to target t where

$$t = (i + \sum_{k=l}^{j+1} c_k 2^k)$$

Allreduce – Async Dissemination Algorithm

Step 0

ranks

state

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1

Allreduce - Async Dissemination Algorithm

Step 0

ranks

state

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
14	0	1	2	3	4	5	6	7	8	9	10	11	12	13
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
1	2	3	4	5	6	7	8	9	10	11	12	13	14	0

Allreduce - Async Dissemination Algorithm

Step 1

ranks

state

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
14	0	1	2	3	4	5	6	7	8	9	10	11	12	13
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
1	2	3	4	5	6	7	8	9	10	11	12	13	14	0

Allreduce - Async Dissemination Algorithm

Step 1

ranks

state

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
14	0	1	2	3	4	5	6	7	8	9	10	11	12	13
13	14	0	1	2	3	4	5	6	7	8	9	10	11	12
12	13	14	0	1	2	3	4	5	6	7	8	9	10	11
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
3	4	5	6	7	8	9	10	11	12	13	14	0	1	2
2	3	4	5	6	7	8	9	10	11	12	13	14	0	1
1	2	3	4	5	6	7	8	9	10	11	12	13	14	0

Allreduce - Async Dissemination Algorithm

Step 2

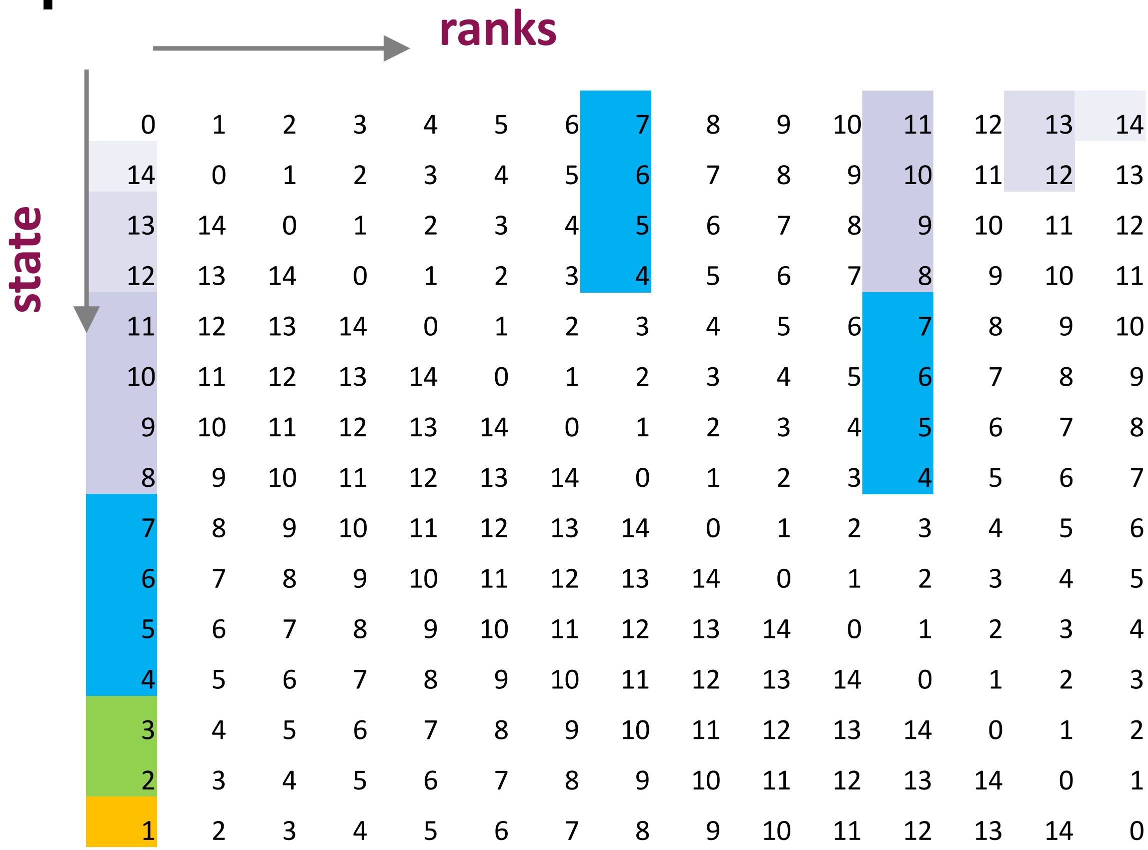
ranks

state

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
14	0	1	2	3	4	5	6	7	8	9	10	11	12	13
13	14	0	1	2	3	4	5	6	7	8	9	10	11	12
12	13	14	0	1	2	3	4	5	6	7	8	9	10	11
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
3	4	5	6	7	8	9	10	11	12	13	14	0	1	2
2	3	4	5	6	7	8	9	10	11	12	13	14	0	1
1	2	3	4	5	6	7	8	9	10	11	12	13	14	0

Allreduce - Async Dissemination Algorithm

Step 2

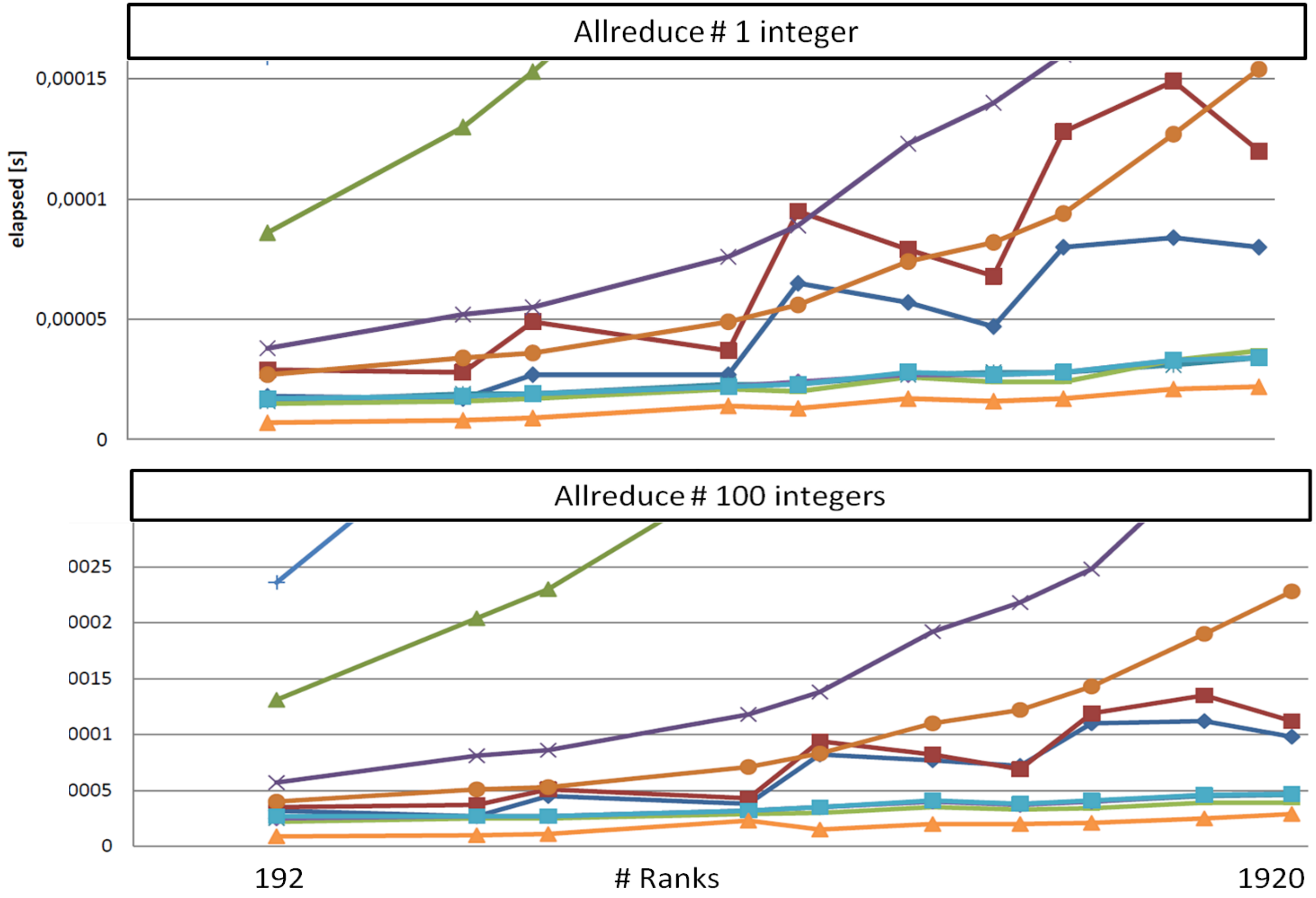


ranks

state

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
14	0	1	2	3	4	5	6	7	8	9	10	11	12	13
13	14	0	1	2	3	4	5	6	7	8	9	10	11	12
12	13	14	0	1	2	3	4	5	6	7	8	9	10	11
11	12	13	14	0	1	2	3	4	5	6	7	8	9	10
10	11	12	13	14	0	1	2	3	4	5	6	7	8	9
9	10	11	12	13	14	0	1	2	3	4	5	6	7	8
8	9	10	11	12	13	14	0	1	2	3	4	5	6	7
7	8	9	10	11	12	13	14	0	1	2	3	4	5	6
6	7	8	9	10	11	12	13	14	0	1	2	3	4	5
5	6	7	8	9	10	11	12	13	14	0	1	2	3	4
4	5	6	7	8	9	10	11	12	13	14	0	1	2	3
3	4	5	6	7	8	9	10	11	12	13	14	0	1	2
2	3	4	5	6	7	8	9	10	11	12	13	14	0	1
1	2	3	4	5	6	7	8	9	10	11	12	13	14	0

Allreduce - Dissemination

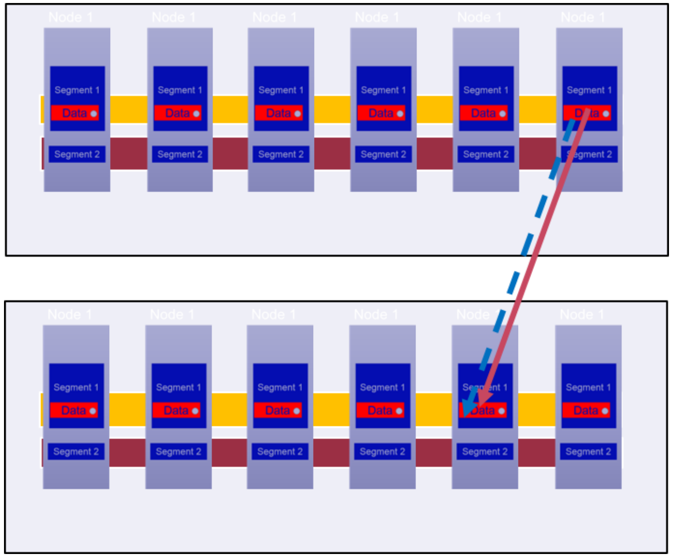


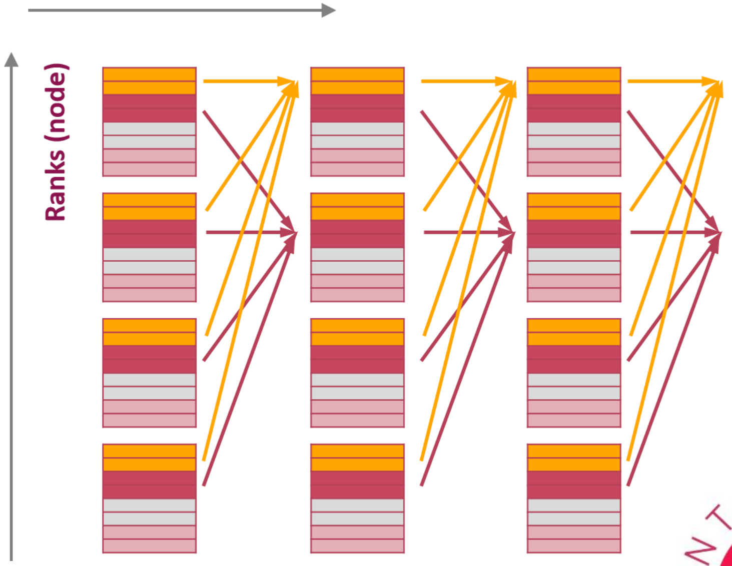
Allgather(V)

Testbed for notified communication - Collectives.

Allgather(V) with GASPI shared segments

- **Irregular data distribution**
- **Pipelined ring with dynamic load distribution**

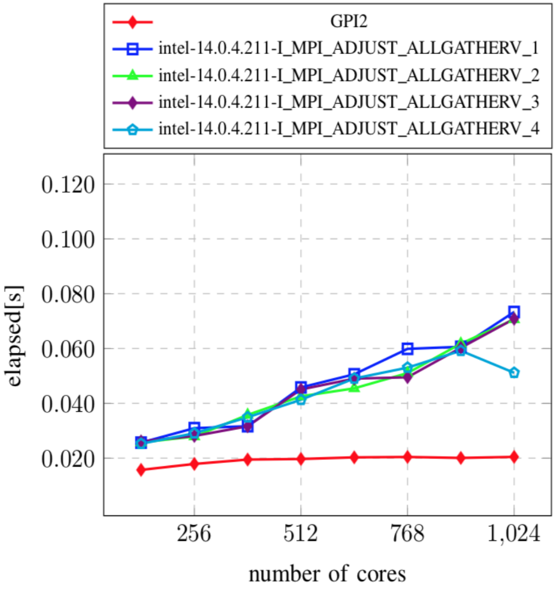




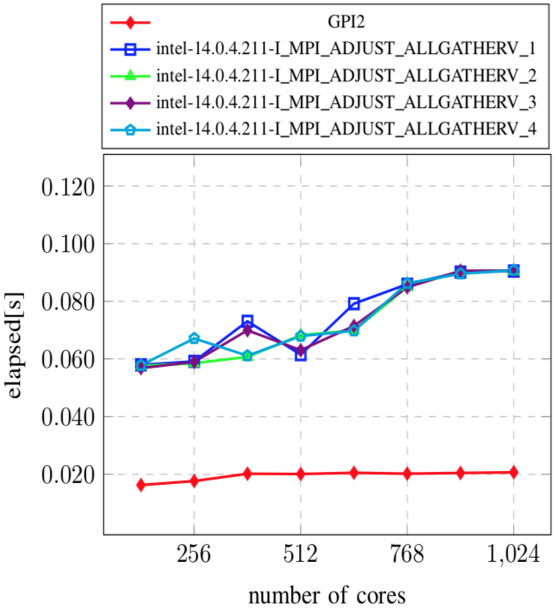
Allgather(V) : Performance results

Data distribution

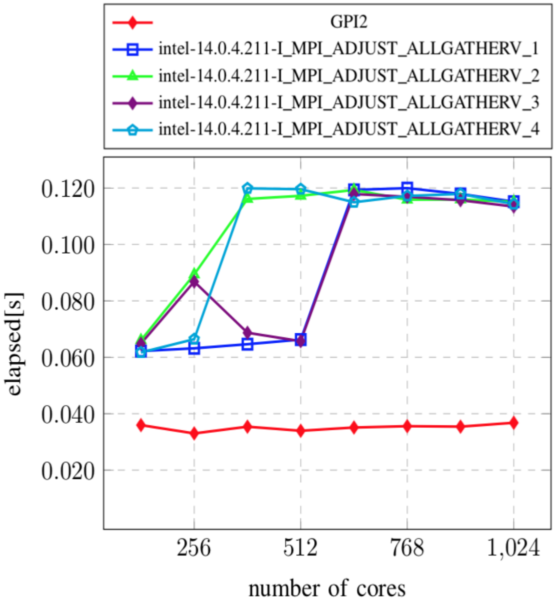
- **Regular, linear decreasing, broadcast**



(f): Regular data distribution



(g): Linearly decreasing data distribution



(h): Broadcast

Questions?

Thank you for your attention

This project is funded from the European Union's Horizon 2020 Research and Innovation programme under Grant Agreement no. 671602.