Parallel Spawning Strategies for Dynamic-Aware MPI Applications

Authors

Iker Martín-Álvarez

Maribel Castillo

José I. Aliaga

Sergio Iserte

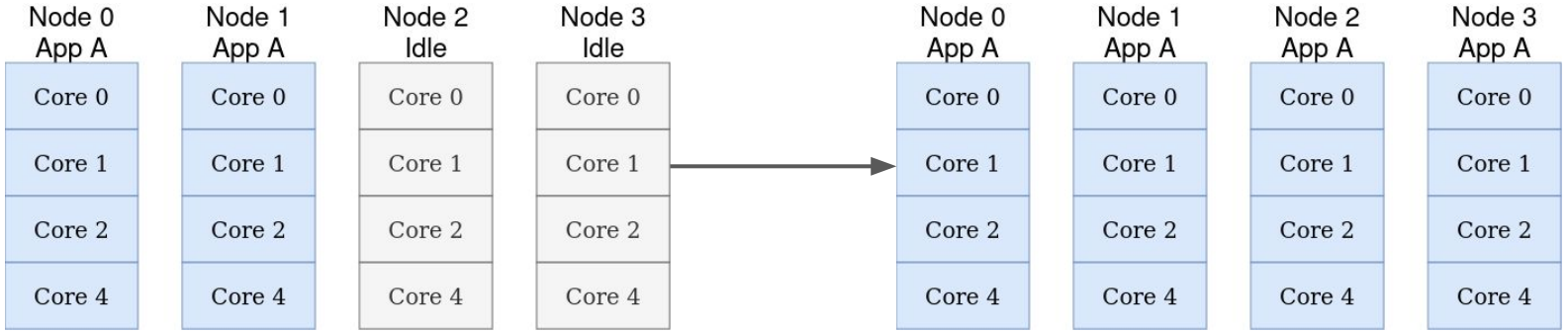
Outline:

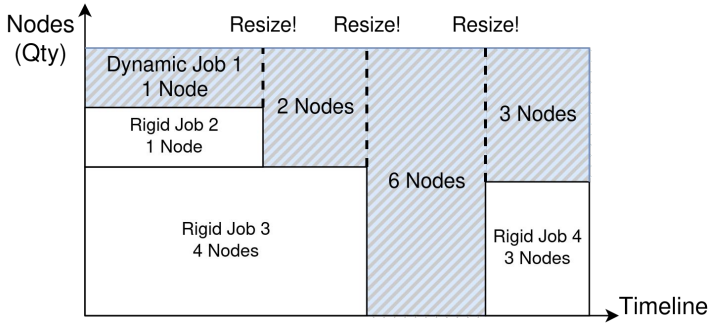
1. Dynamic Resources
2. A Challenge with Traditional MPI
3. A … Solution?
4. The Parallel Strategy
5. Preliminary Results
6. Closing Remarks

Outline:

1. Dynamic Resources
2. A Challenge with Traditional MPI
3. A … Solution?
4. The Parallel Strategy
5. Preliminary Results
6. Closing Remarks

What is Dynamic Resource Management (DRM)?



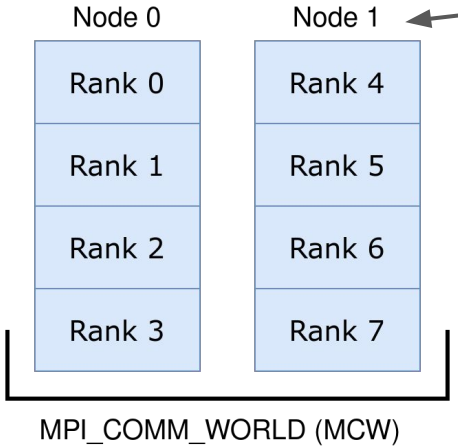


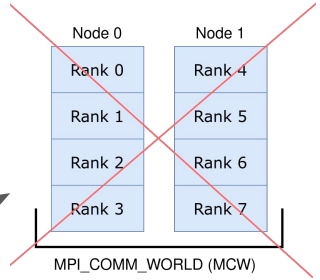
Outline:

1. Dynamic Resources
2. A Challenge with Traditional MPI
3. A … Solution?
4. The Parallel Strategy
5. Preliminary Results
6. Closing Remarks

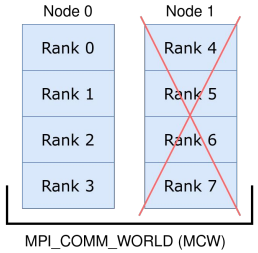
Returning Nodes back to the RMS

RMS requires node 1 back

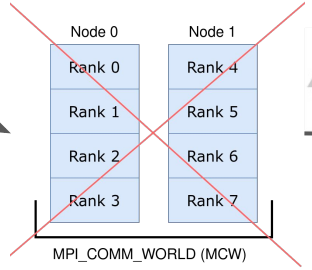


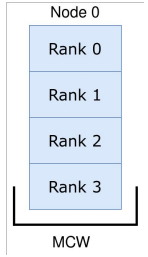


Kill job



Terminate unneeded processes





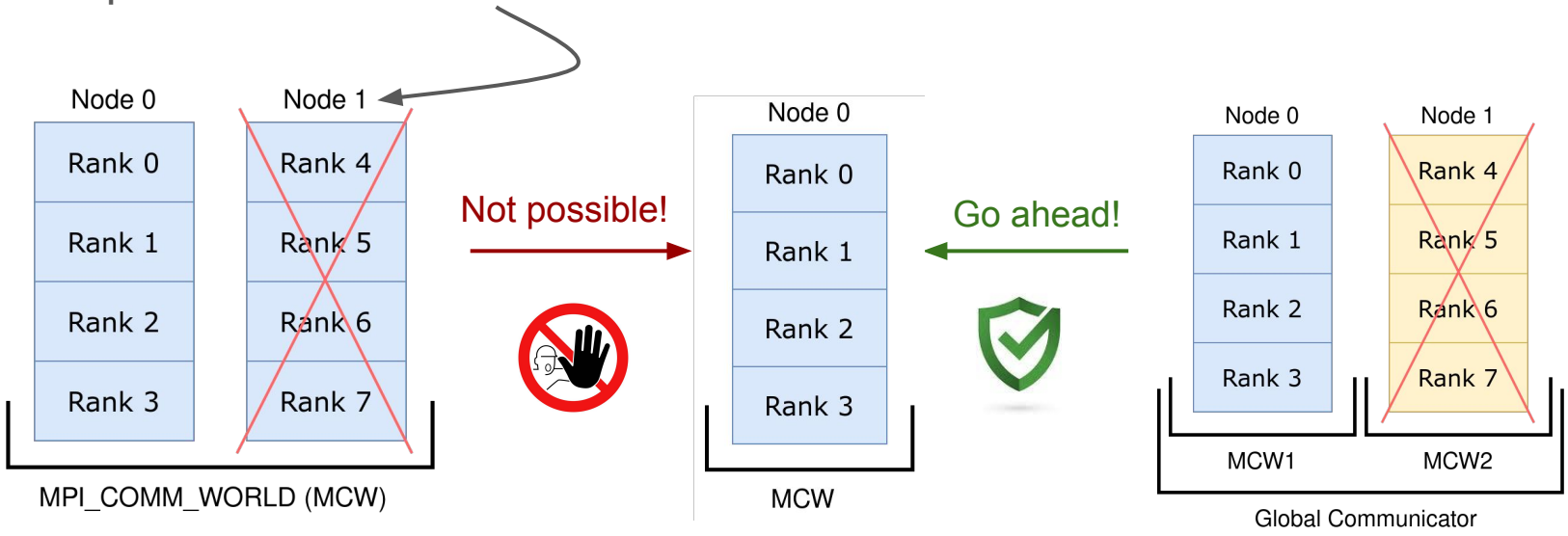
Replace all processes

Outline:

1. Dynamic Resources
2. A Challenge with Traditional MPI
3. A … Solution?
4. The Parallel Strategy
5. Preliminary Results
6. Closing Remarks

Returning Nodes back to the RMS

RMS requires node 1 back

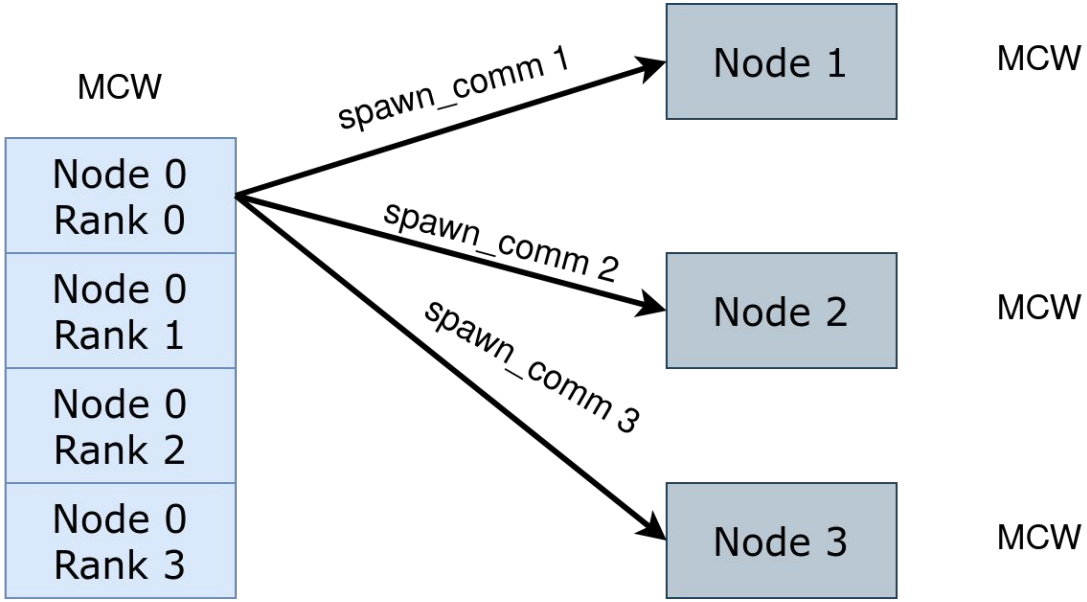


Outline:

1. Dynamic Resources
2. A Challenge with Traditional MPI
3. A … Solution?
4. The Parallel Strategy
   - 4.1 Basic Strategy
   - 4.2 Iterative Diffusion
   - 4.4 Binary Connection
5. Preliminary Results
6. Closing Remarks

Basic Strategy - Get goal, performance later

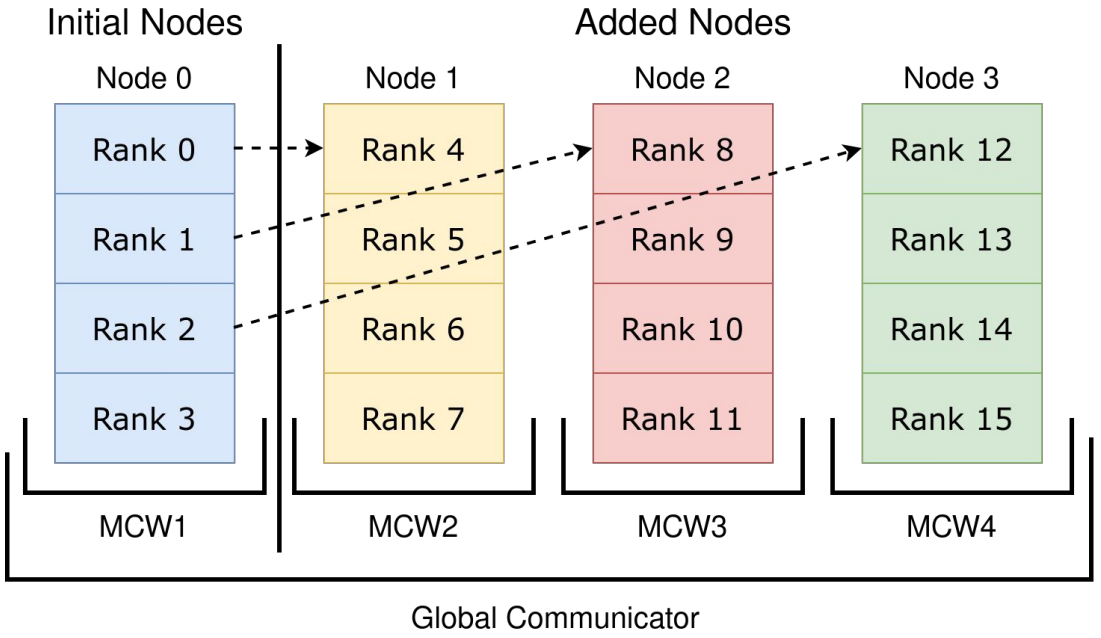
Goal: Remove MPI Comm World limitation for removing ranks



Requires an additional step!
Connect all new and old ranks into a single communicator

Iterative Diffusion (I)

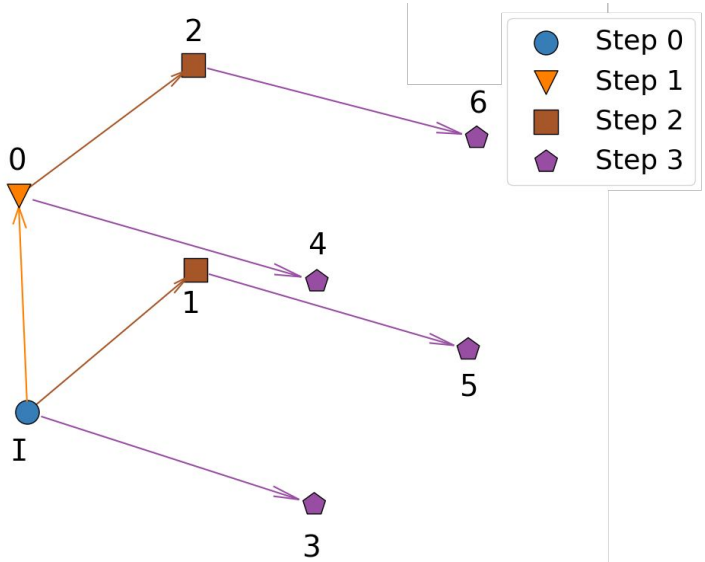
Improvement:
Make all available processes help in the spawn.



Iterative Diffusion (II)

Improvement:

New processes collaborate in further spawns, if required.



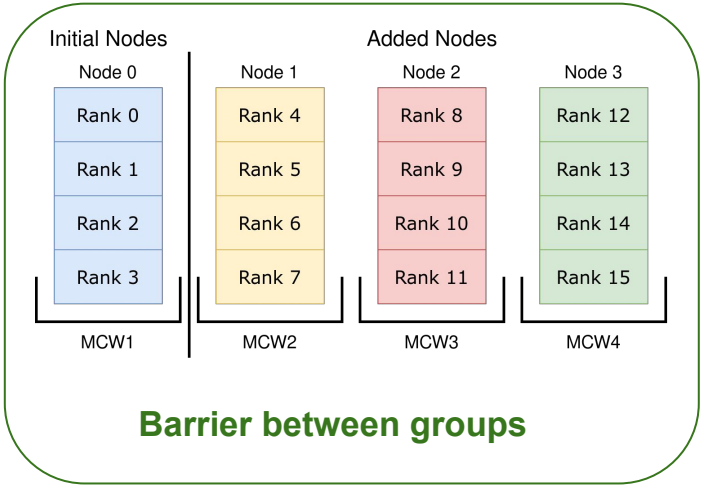
Example: 1 cpu per node
Obj: Get to 64 nodes

Example: 2 cpu per node
Obj: Get to 64 nodes

Binary Connection

Synchronize before merging!

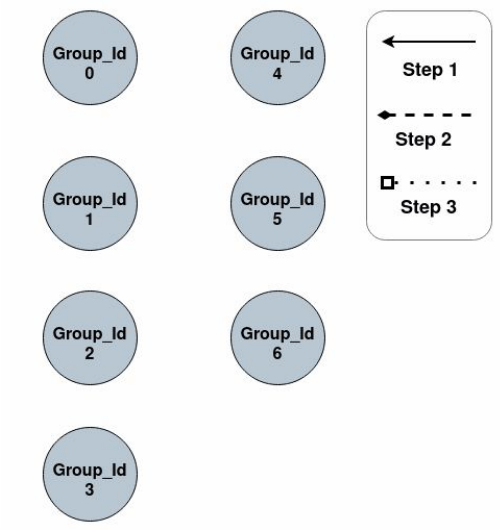
To connect is used MPI_Comm_connect and MPI_Comm_accept, but require that everyone is aware that the ports are open.



Barrier between groups

Connect in steps to its mirror group!

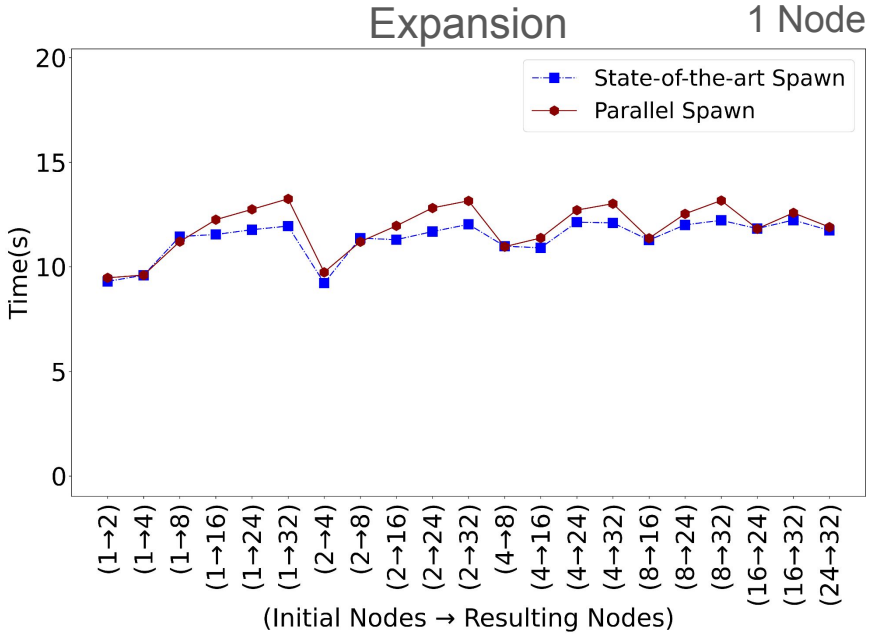
Functions MPI_Comm_connect and MPI_Comm_accept can only merge two MCW for each call.

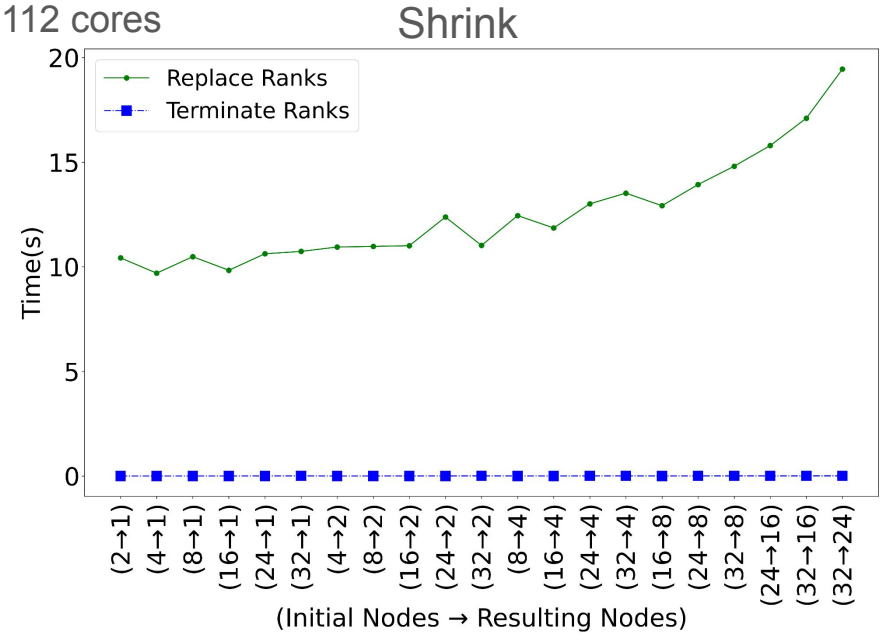


Outline:

1. Dynamic Resources
2. A Challenge with Traditional MPI
3. A … Solution?
4. The Parallel Strategy
5. Preliminary Results
6. Closing Remarks

MarenostrumV - BSC





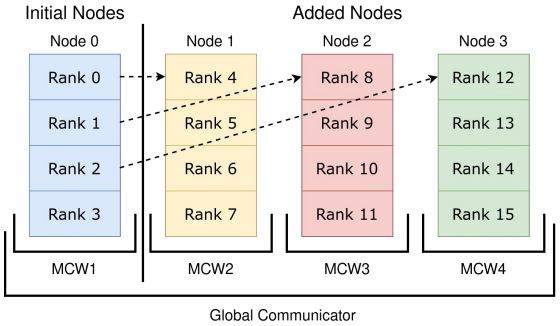
Parallel spawn provides a slight overhead (maximum of 1.25x more time).

→

But allows to perform resizes in immediate time (minimum speedup of 1387x).

Outline:

1. Dynamic Resources
2. A Challenge with Traditional MPI
3. A … Solution?
4. The Parallel Strategy
5. Preliminary Results
6. Closing Remarks



New parallel technique to enable termination of ranks without replacement for resize operations.

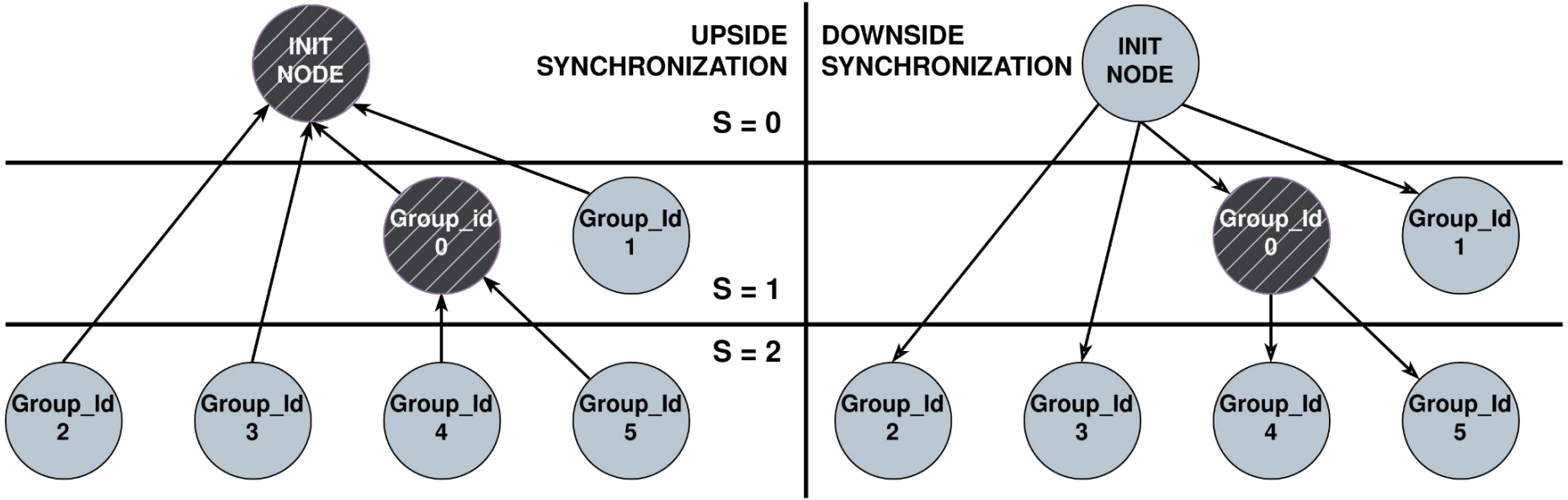
Parallel spawn maintains similar performance to previous methods. (1,25x max)

Parallel Spawning Strategies for Dynamic-Aware MPI Applications

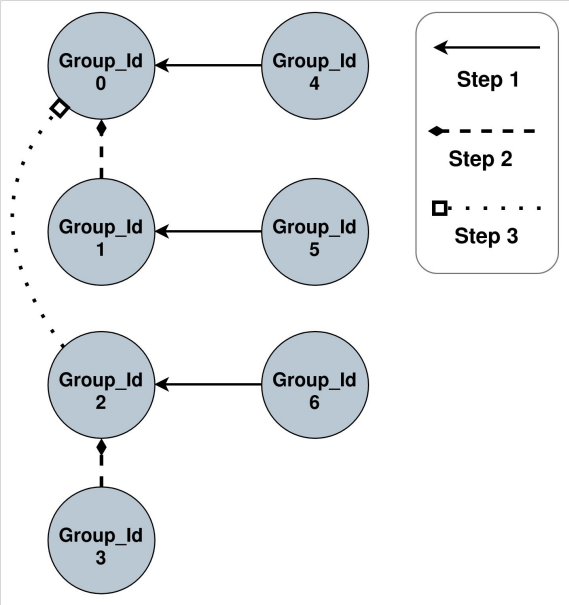
Preprint

Authors
Iker Martín-Álvarez (martini@uji.es)
Maribel Castillo
José I. Aliaga
Sergio Iserte

Synchronizing before Connecting



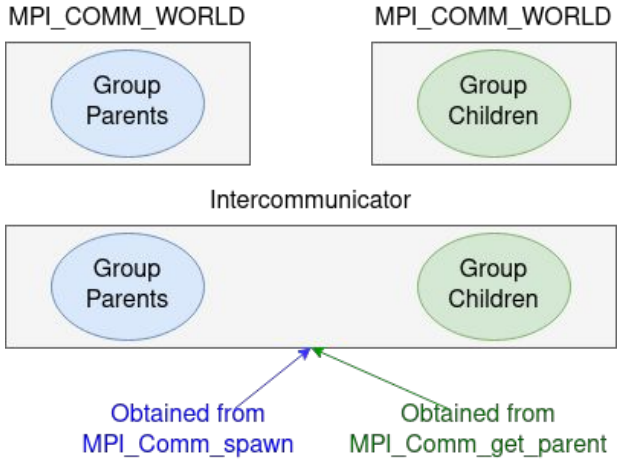
Connecting


Group_Id 0
Group_Id 4
Group_Id 1
Group_Id 5
Group_Id 2
Group_Id 6
Group_Id 3
Step 1
Step 2
Step 3

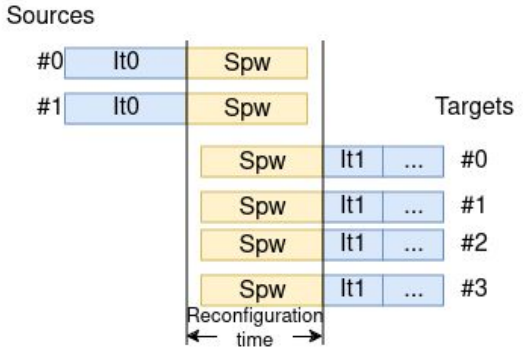
Baseline Method

2 Methods to spawn processes:

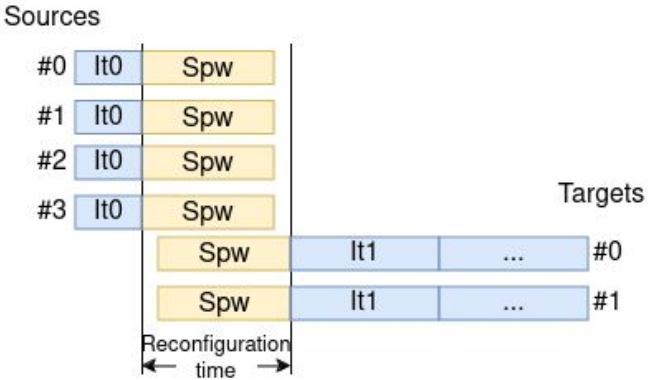
- **Baseline**
- Merge



Expand

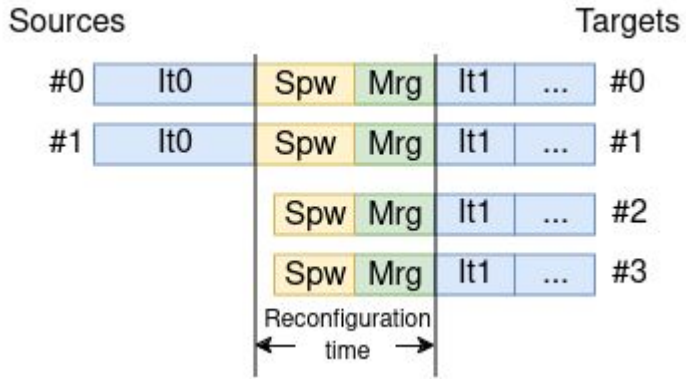


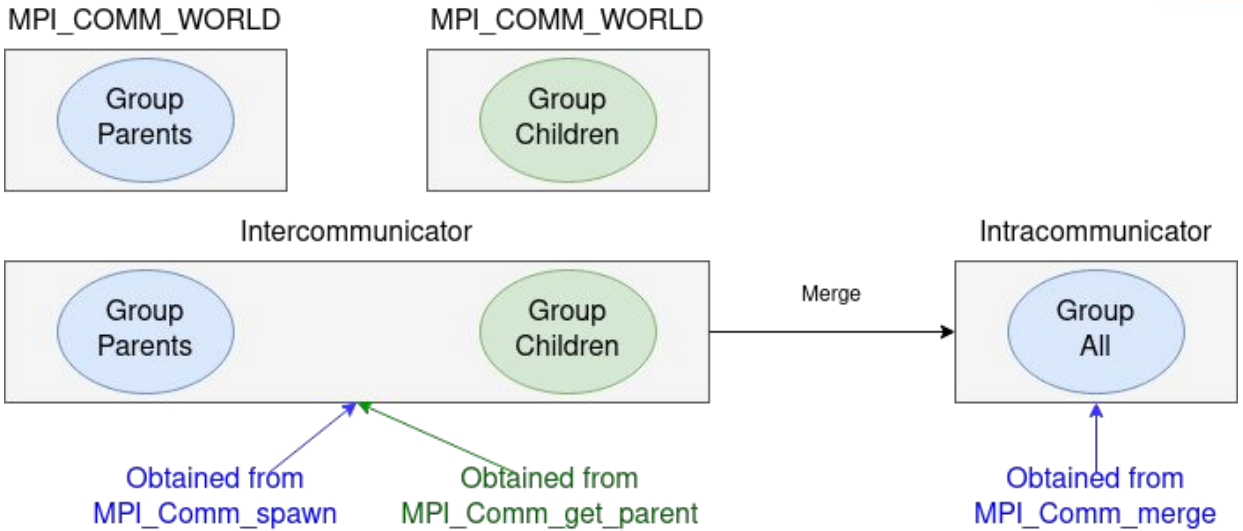
Shrink



Merge Method - Expand

2 Methods to spawn processes:
- Baseline
- **Merge**

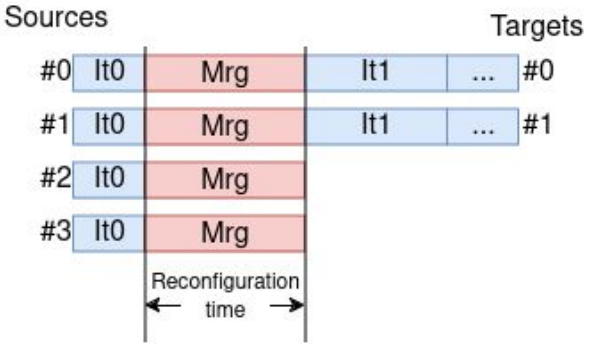


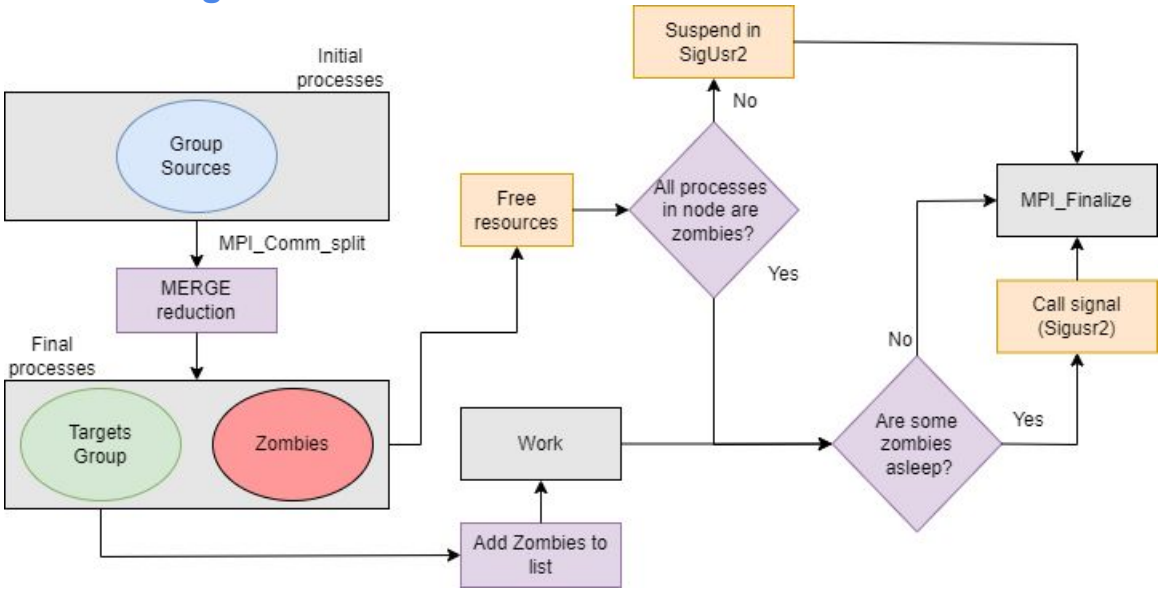


Merge Method - Shrink

2 Methods to spawn processes:

- Baseline
- **Merge**

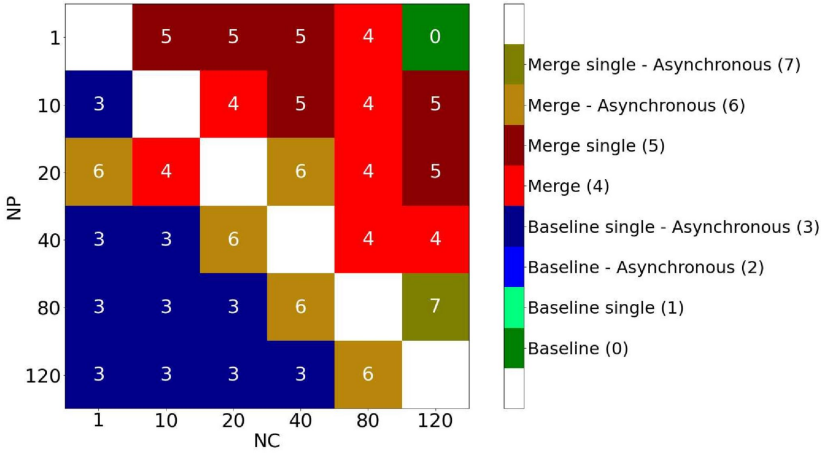




Process management results

- Which combination of methods/strategies perform better?
- Executed application is Conjugate Gradient (CG)

Preferred methods



Expansion times

NP	NC	RT_S(s) Baseline	Baseline single	Merge	Merge single
1	10	0,316	0,313	**0,284**	0,289
	20	0,861	1,035	**0,716**	0,721
	40	0,861	0,995	**0,799**	0,809
	80	0,989	1,075	**0,932**	1,211
	120	**0,912**	1,029	0,992	1,023
10	20	1,286	1,654	**0,477**	0,486
	40	1,213	1,635	0,766	**0,743**
	80	1,293	1,693	**0,861**	0,888
	120	1,315	1,636	**0,891**	0,915
20	40	1,304	1,991	**0,790**	0,821
	80	1,407	1,932	**0,864**	0,958
	120	1,413	1,870	1,089	**1,071**
40	80	1,429	2,022	0,894	**0,877**
	120	1,526	2,113	**0,923**	0,942
80	120	1,522	2,316	**0,906**	0,996

Shrinking times

NP	NC	RT_S(s) Baseline	Baseline single	Merge
10	1	0,200	0,205	**0,001**
20	1	0,400	0,427	**0,001**
	10	0,933	1,220	**0,001**
40	1	0,400	0,423	**0,030**
	10	0,883	1,165	**0,025**
	20	1,271	1,740	**0,116**
80	1	0,388	0,418	**0,217**
	10	0,881	1,189	**0,181**
	20	1,262	1,826	**0,149**
	40	1,415	2,039	**0,148**
120	1	0,375	0,424	**0,231**
	10	0,953	1,251	**0,148**
	20	1,229	1,776	**0,178**
	40	1,300	2,102	**0,351**
	80	1,529	2,235	**0,156**

Related article