Data Locality Aware Strategy for Two-Phase Collective I/O.

Rosa Filgueira, David E.Singh, Juan C. Pichel, Florin Isaila, and Jesús Carretero.

Universidad Carlos III de Madrid (Spain).

Sumary

- Problem description.
- Main objectives.
- Locality Aware strategy for Two Phase I/O:
  - Linear Assignment Problem.
  - LA-Two-Phase I/O (LATP).
- Evaluation.
- Results
- Conclusions.

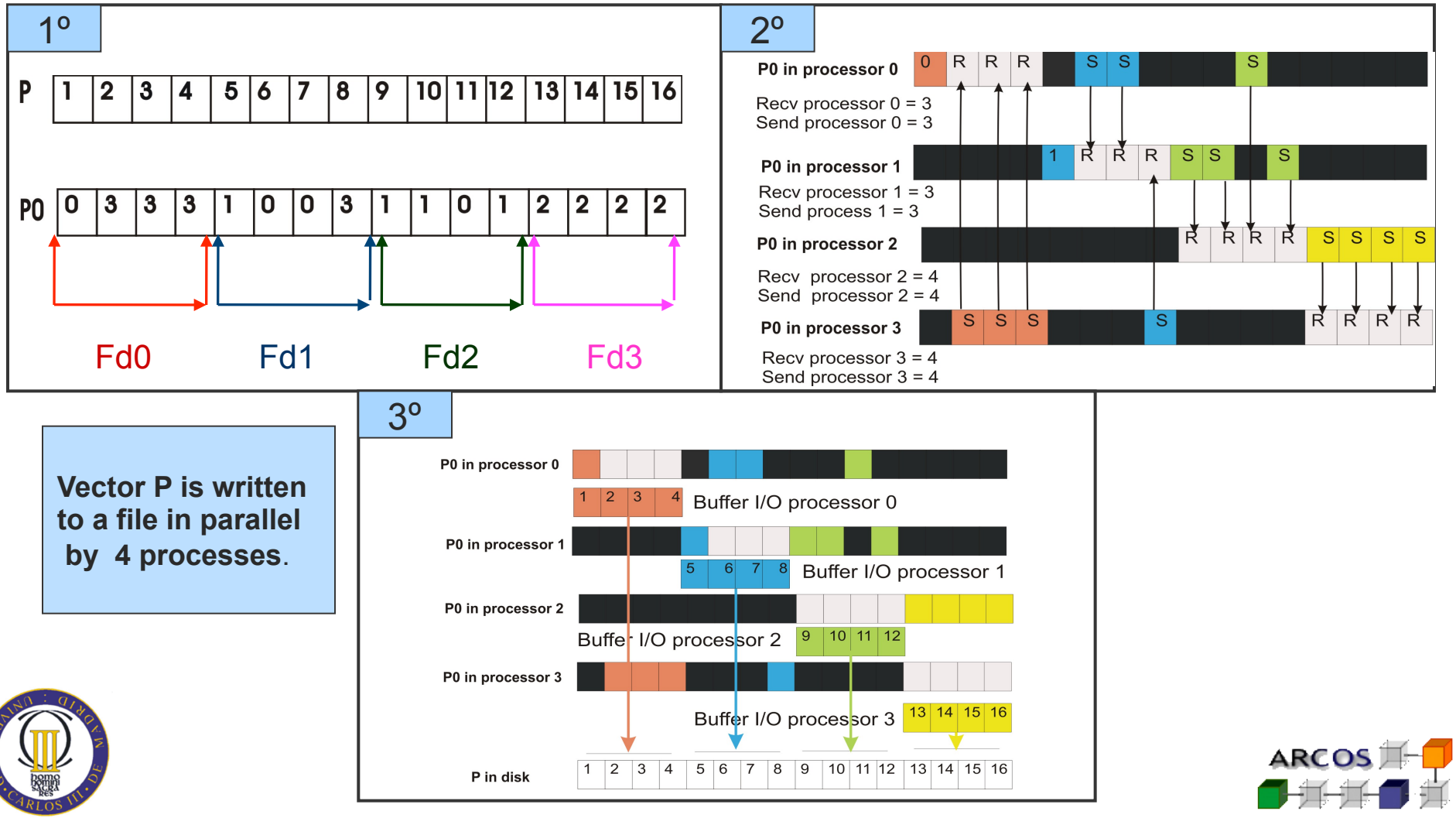
1. Problem description(I)

- Parallel scientific application generate lots of data
- Access pattern:
  - Individual process read/write non-contiguously.
  - Collective access: contiguous.
- Collective I/O: aggregates individual small requests into larger ones
  - Disk-directed I/O (aggregation close to disk).
  - Two-phase I/O (aggregation at compute nodes): our optimization target.

1. Problem description (II)

- Two-Phase I/O phases:
  - Shuffle: aggregate data into contiguous buffers.
  - I/O: transfer contiguous buffer to file system.
- Before these two phases:
  - File region is divided into equal contiguous regions called File Domains (FD).
  - Each FD is assigned to a subset of compute nodes (aggregators).
    - Each aggregator is responsible for transferring all data from its FD to the file system.
- Cause of inefficiency: The assignment of FD to aggregators is independent of data distribution.

1. Problem description (III)



Vector P is written to a file in parallel by 4 processes.

2. Main Objectives

- Replacing the rigid assignment of FDs by an assignment dependent of the initial data distribution.
- Our assignment increases the I/O efficiency and reduces:
  - The number of communication operations.
  - The volume of communication.
  - The total execution time.

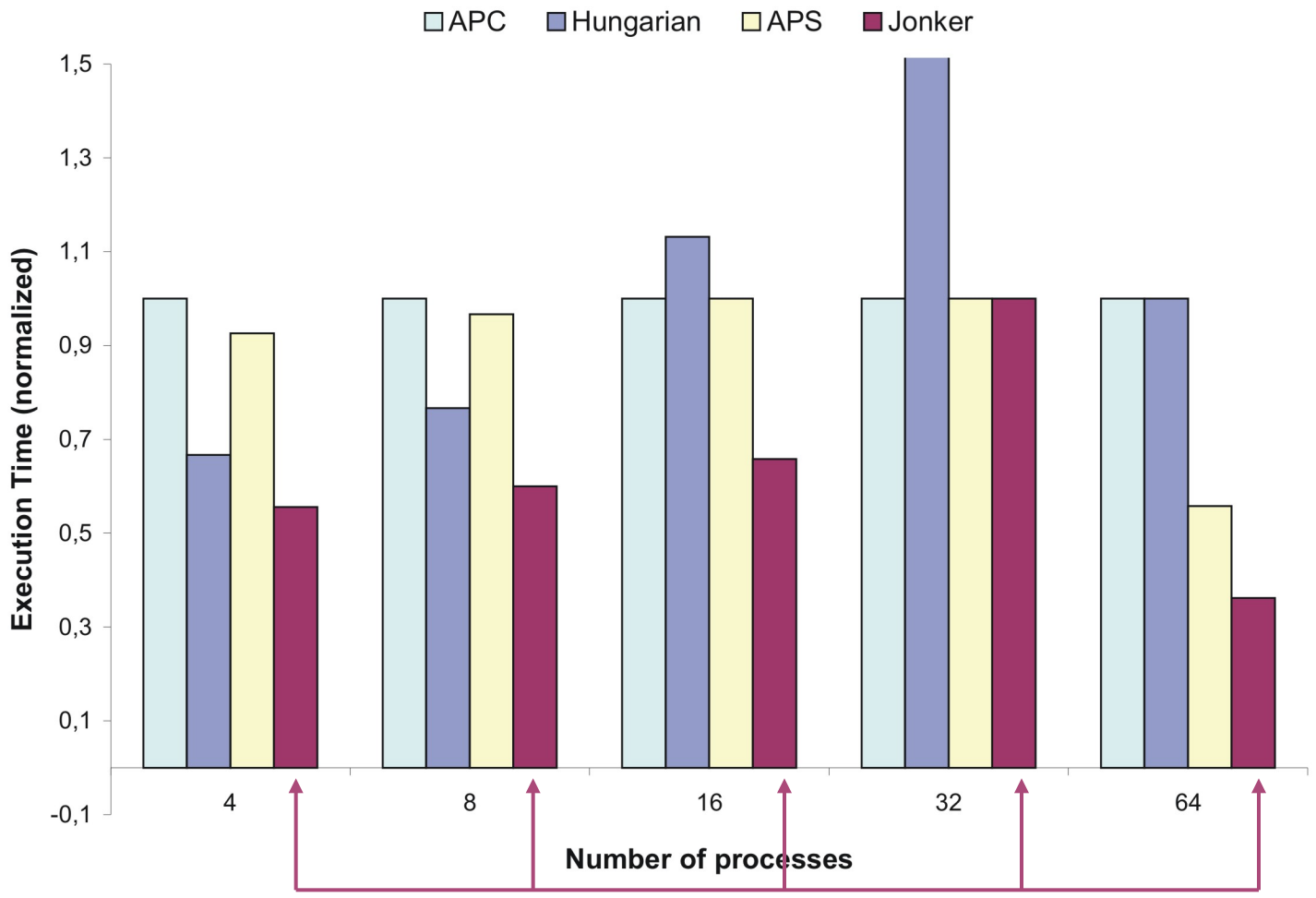
3. Locality aware strategy of Two Phase I/O.

- This work presents Locality-Aware Two-Phase (LATP) I/O.
  - LATP employs the Linear Assignment Problem (LAP) for finding an optimal assignment of FD to processes during the I/O stage.

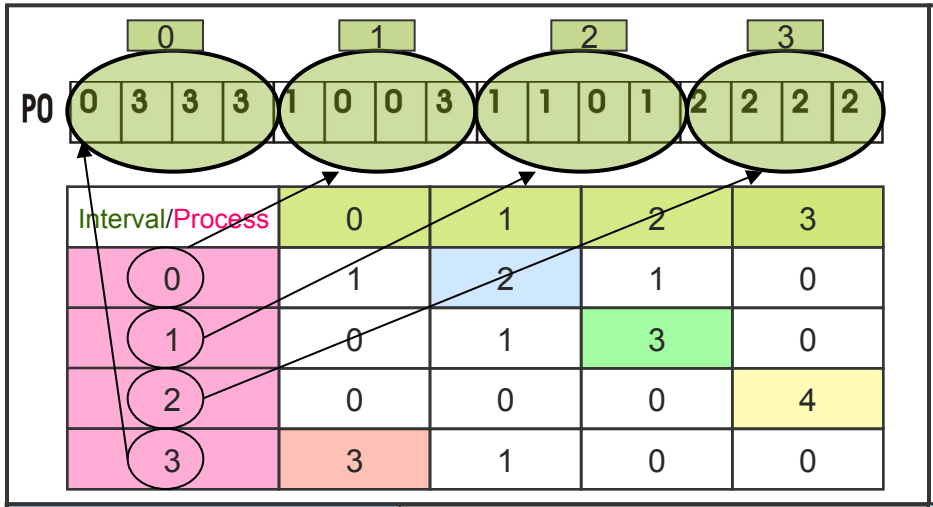
3.1 Linear Assigment Problem (I)

- LAP computes the optimal assignment of m items to n elements given an m x n cost matrix.
- Several algorithms have been developed for LAP:
  - Hungarian algorithm.
  - Jonker and Volgenant algorithm.
  - APC and APS Algorithms.
- All algorithms produce the same assignment.
- The difference is the time to compute the optimal allocation.

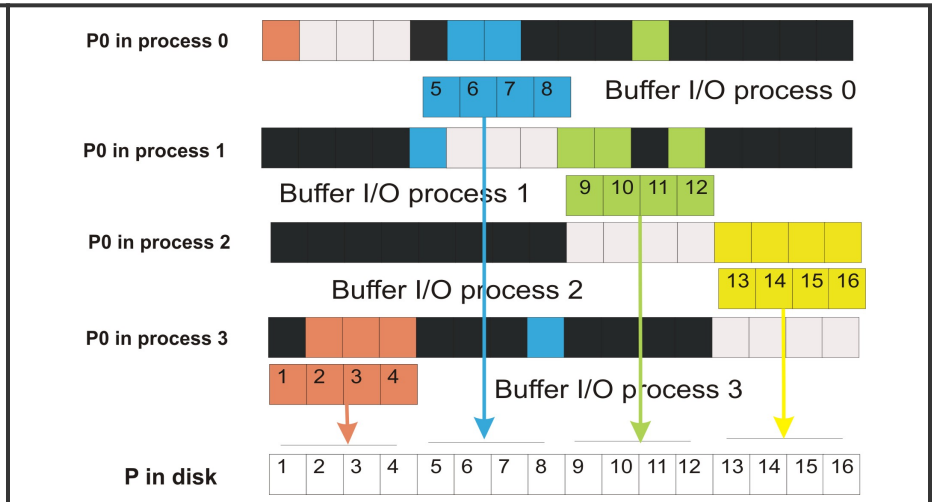
3.1 Linear Assigment Problem (II)



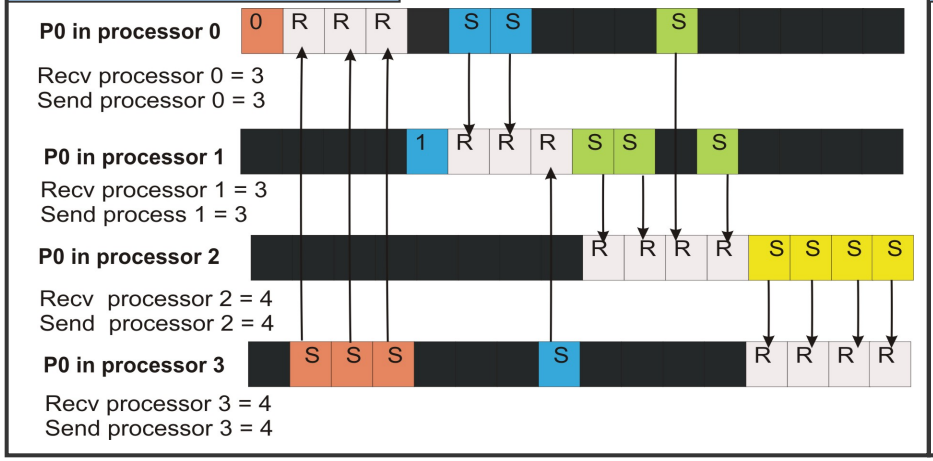
3.2 LA-Two-Phase I/O



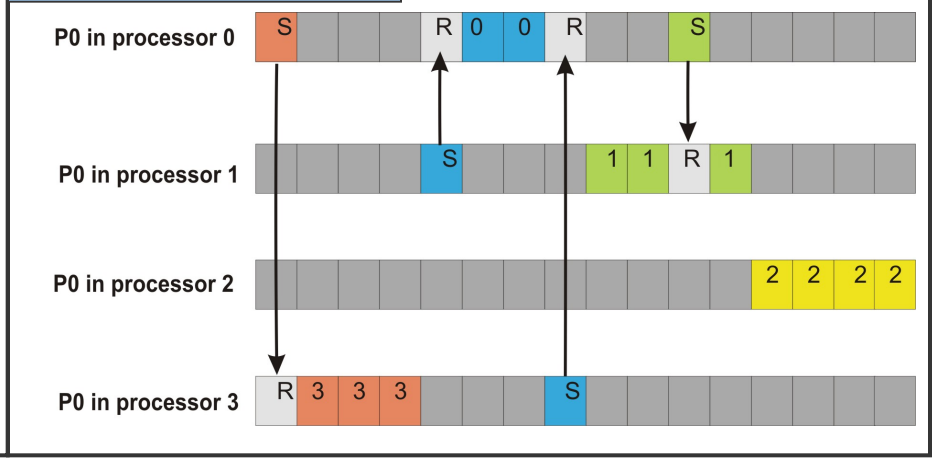
Interval/Process	0	1	2	3
0	1	2	1	0
1	0	1	3	0
2	0	0	0	4
3	3	1	0	0



Original communication



LATP communication



4. Evaluation (I)

- Platform → Magerit Cluster (CESVIMA), 1200eServer BladeCenter nodes.
  - Node→ 2 processor IBM 64 bits, 64 GB RAM adn 40 GB HD.
  - Interconnection → Myrinet.
  - MPICH version →MPICHGM 2.7.15NOGM.
  - File system →PVFS 1.6.3 with 1 metadata server and 8 I/O (64KB striping factor).

4. Evaluation (II)

- Application→BISP3D:
  - Semiconductor devices simulator based on finite element methods.
  - Problem input: an unstructured mesh
    - The mesh is divided into several sub-domains (METIS library).
    - Each sub-domain is assigned to one process.
    - Each process makes calculations on assigned data.
    - The results are written to a file.

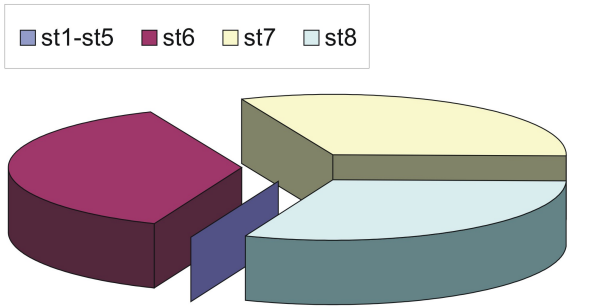
4. Evaluation (III)

- Performed evaluations:
  - Different meshes.
  - Different load.
- The file size (in MB) of each file based on the mesh and load.

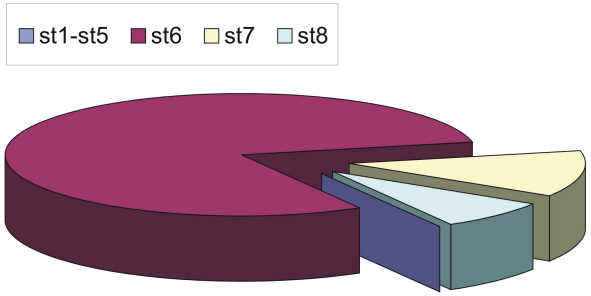
Load	Mesh 1	Mesh 2	Mesh 3	Mesh 4
100	18	12	28	110
200	36	25	56	221
500	90	63	140	552

4. Evaluation (IV)

- Two-Phase I/O stages:
  - File offsets and lengths calculation (st1).
  - File offsets and lengths communication (st2).
  - Interval assignment (st3).
  - File domain calculation (st4).
  - Access request calculation (st5).
  - Metadata transfer (st6).
  - Buffer writting (st7).
  - File writting (st8).



Mesh1 with load 100 and **16 processes**



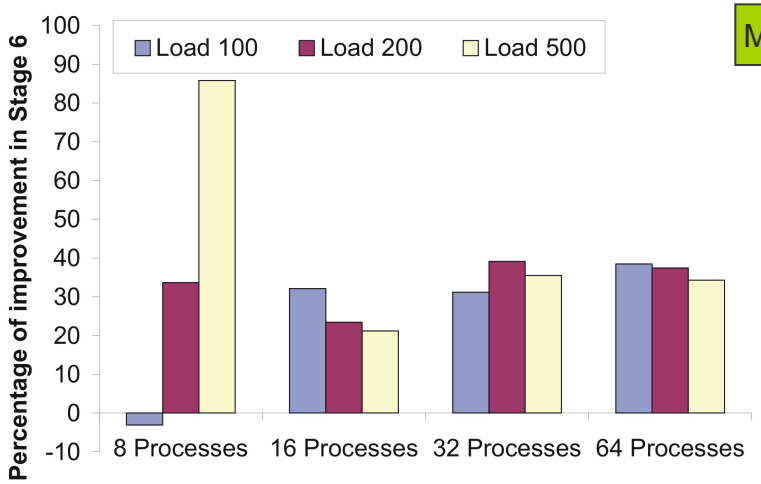
Mesh1 with load 100 and **64 processes**

5. Results

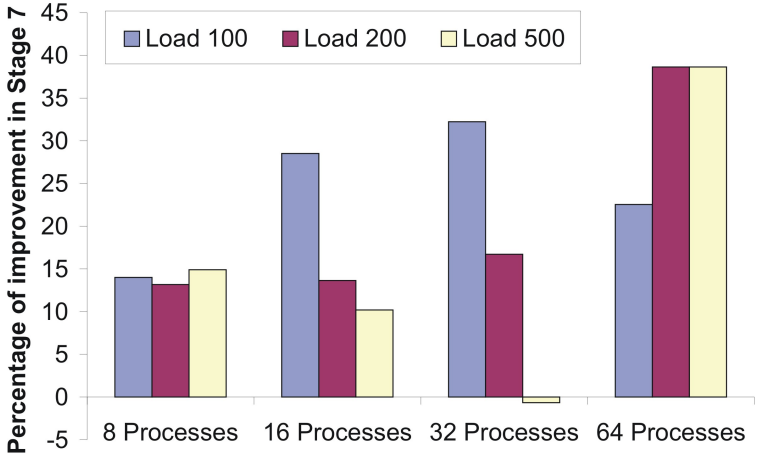
- Percentage of improvement in stages 6 and 7.
- Reduction of transfered data volume.
- Overall Improvement.

5.1 Improvement in stages 6 and 7.

Mesh1



Mesh1

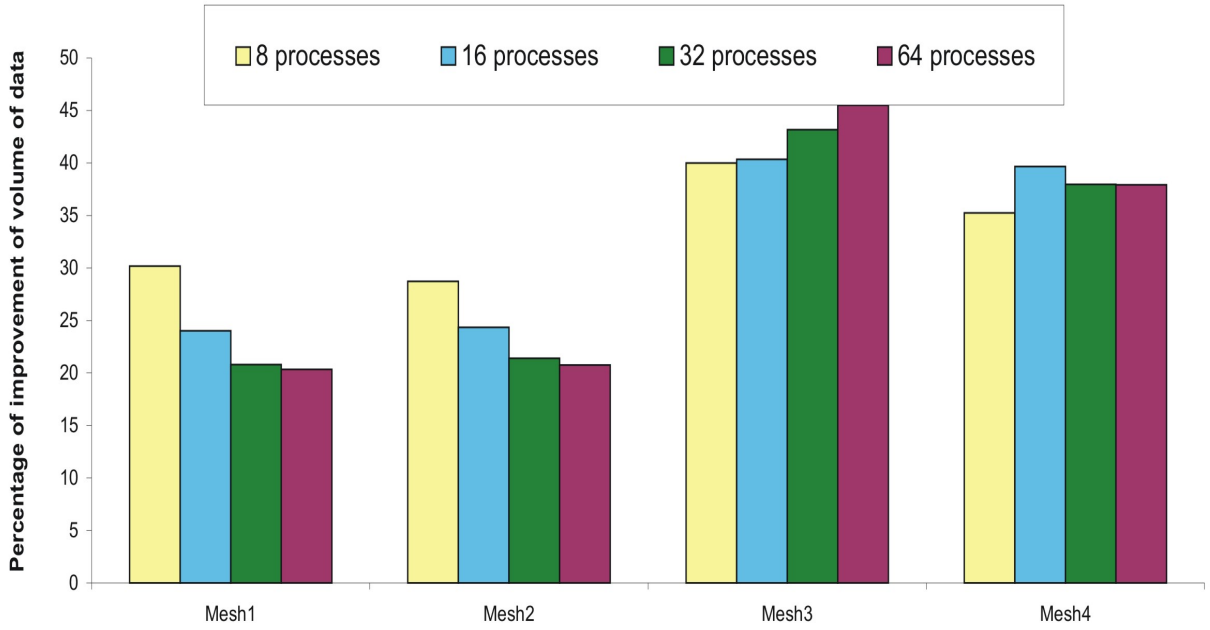


In St6 each process:
-calculates what request of other processes lie in its FD.
-creates a list of offsets and lengths for each process.
-sends the lists to the rest process

In St7 each process:
-sends the data calculated in St6 stage.

LATP:
- reduces the time of st6 and st7 in most cases.
- increases the locality (maximizes data stored in local FD):
  -Sends less data to the other processes
  -Reduces volume and number of communication operations.

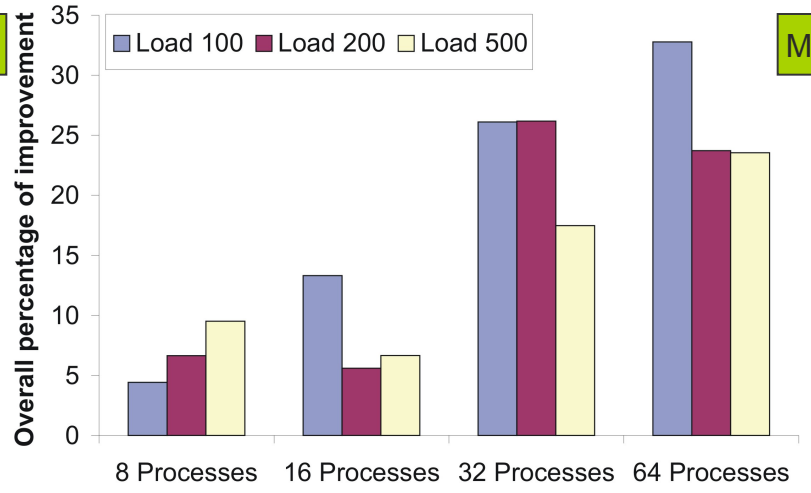
5.1 Reduction of communications



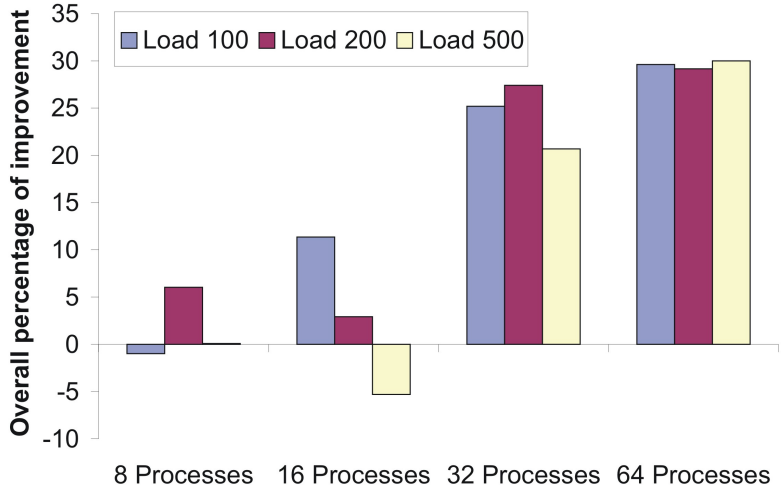
When **LATP** is applied, the transferred data volume is reduced.

5.3 Overall Improvement

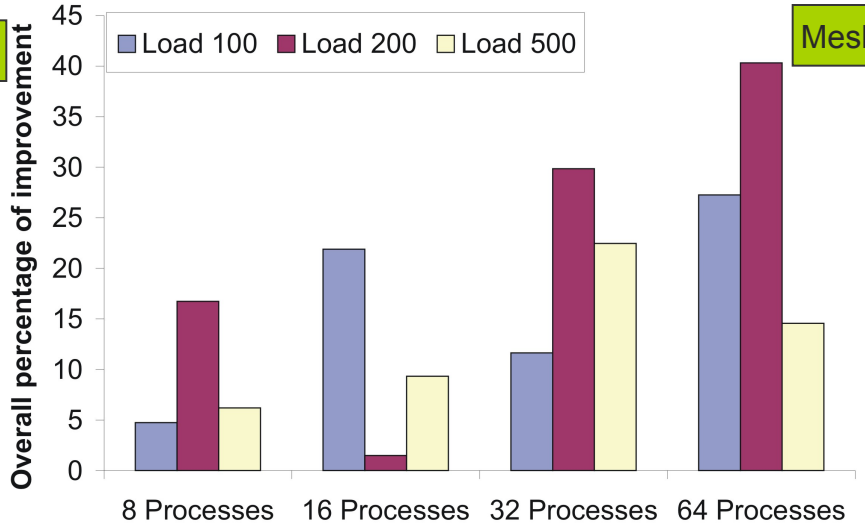
Mesh1



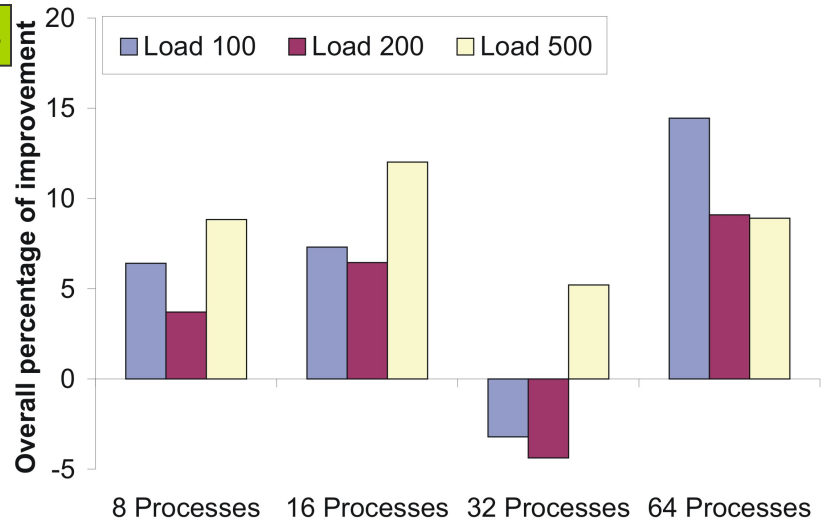
Mesh2



Mesh3



Mesh4



6. Conclusions

- LATP is an optimization of two-phase collective I/O.
- Uses Linear Assignment problem for maximizing the locality.
- Improves overall performance.
- The new stage (st3) has insignificant overhead.
- Scales well: the greater the number of processes, the larger improvement.