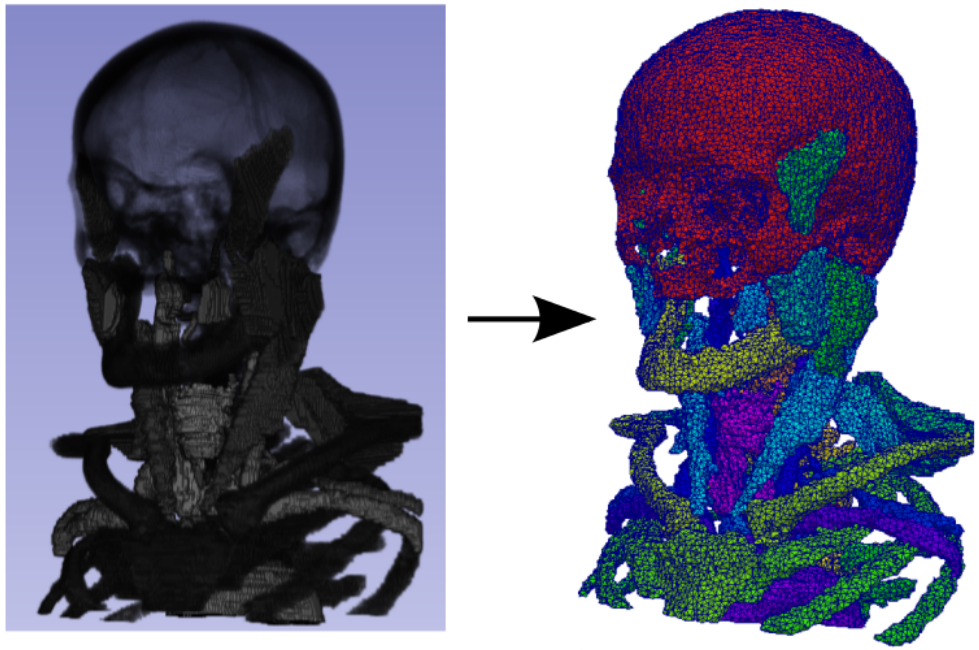
Scalable Parallel Delaunay Image-to-Mesh Conversion for Shared and Distributed Memory Architectures

Daming Feng
Advisors: Dr. Nikos Chrisochoides
Dr. Andrey Chernikov
Department of Computer Science, Old Dominion University
Center for Real-time Computing (https://crtc.cs.odu.edu/)



This work in part is funded by NSF grant no. CCF-1439079, NASA grant no. NNX15AU39A and DoD's PETTT Special Project PP-CFD-KY07-007. In addition, it utilized resources from the Extreme Science and Engineering Discovery Environment (XSEDE), which is supported by National Science Foundation grant number ACI-1053575. Finally, it is partially supported by the Richard T. Cheng Endowment.

Contents

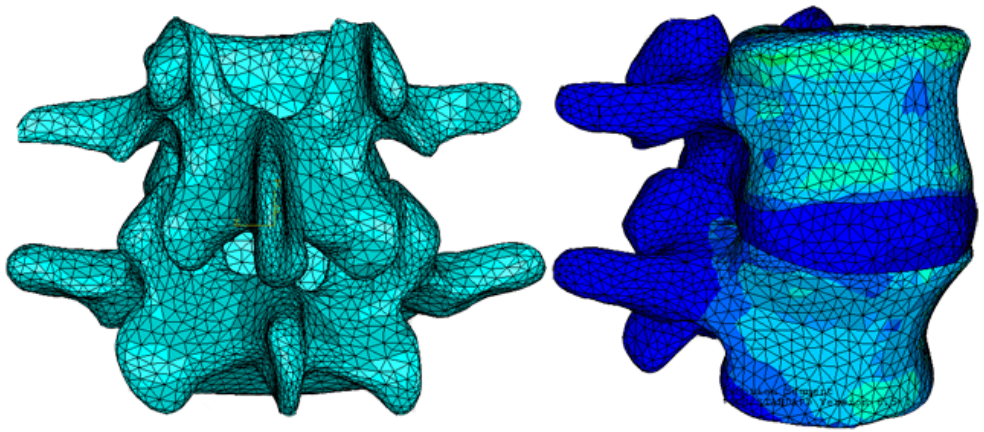
Motivation

Application:

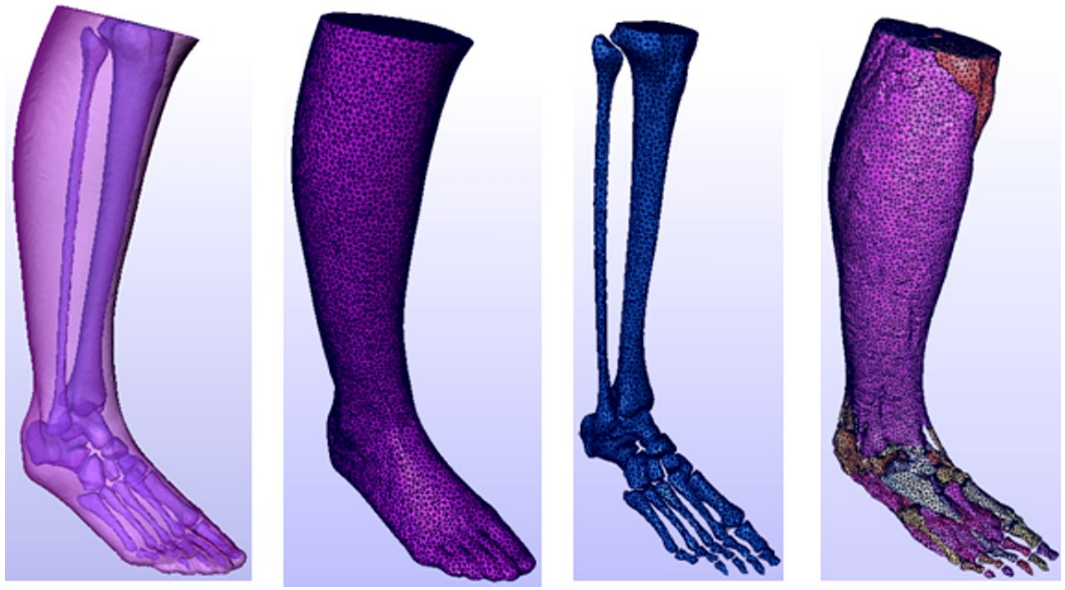
- Finite element modeling
  - Image guided therapy
  - Patient-specific interactive surgery simulation

Image-to-mesh conversion requirement

- Quality: radius-to-edge ratio, dihedral angles, size
- Fidelity: mesh surface should be a correct geometric representation of object boundary
- Scalability: the ability of a parallel algorithm to achieve a speedup proportional to the number of cores.



Midwest Medico-Legal Associates.

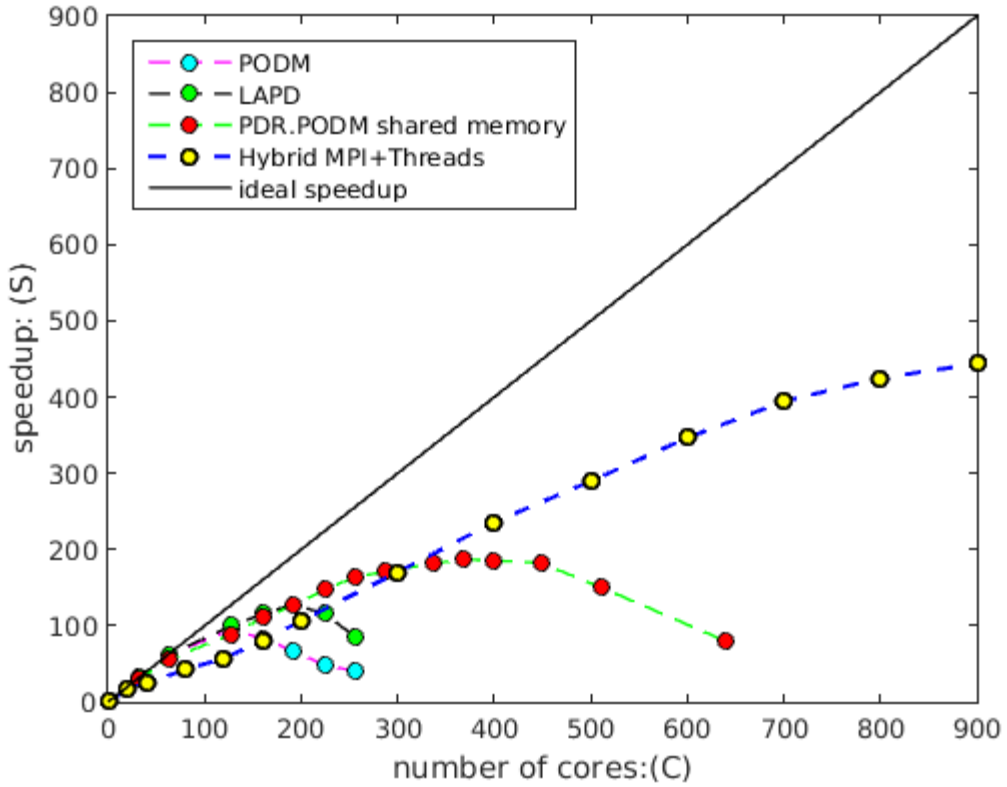


http://ireteth.certh.gr/mechatronics/biomechanics/

Three Parallel Meshing Algorithms

Proposed Parallel Delaunay Image-to-Mesh Algorithms:

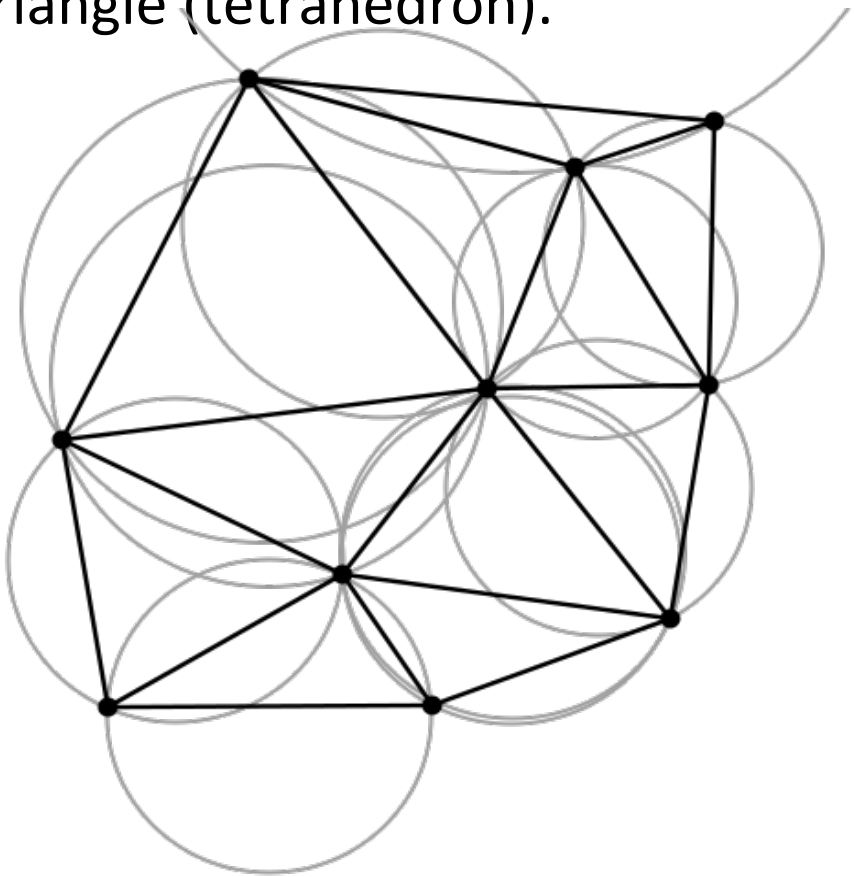
- LAPD:
  - Employs a data locality-aware parallel mesh refinement procedure to reduce the communication overhead caused by the remote memory accesses
- PDR.PODM:
  - Combines the best features of two previous algorithms
- Hybrid MPI+Threads Meshing:
  - Inter-node Communication and Data Exchange (MPI)
  - Intra-node Parallel Mesh Refinement (Pthread)

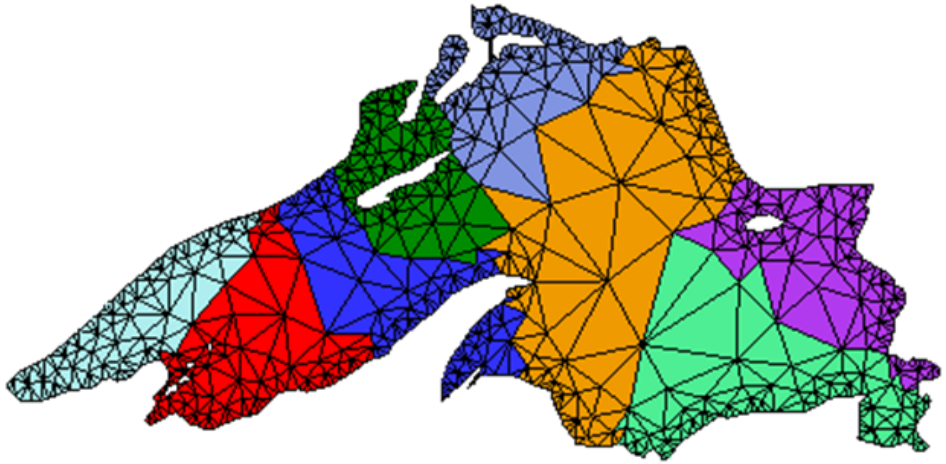


Contents

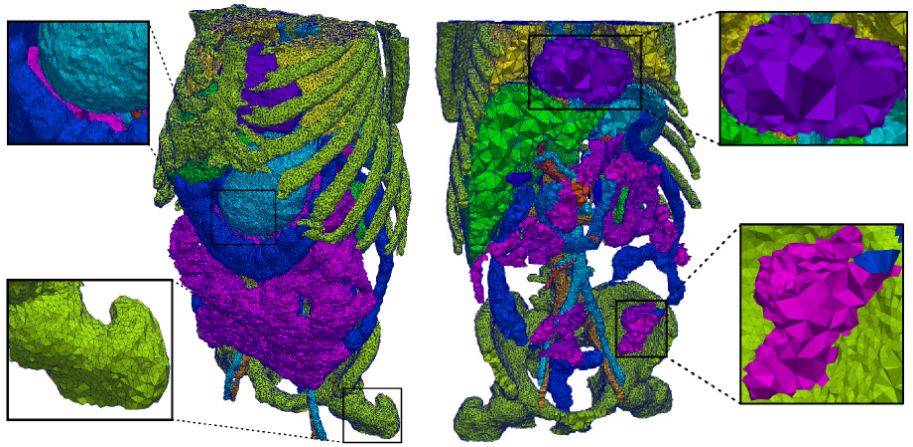
Delaunay Mesh Generation

- Delaunay Mesh Generation creates mesh conforming to Delaunay property, i.e., in a Delaunay mesh, no point (vertex) in the mesh is inside the circumcircle (circumsphere) of any triangle (tetrahedron).



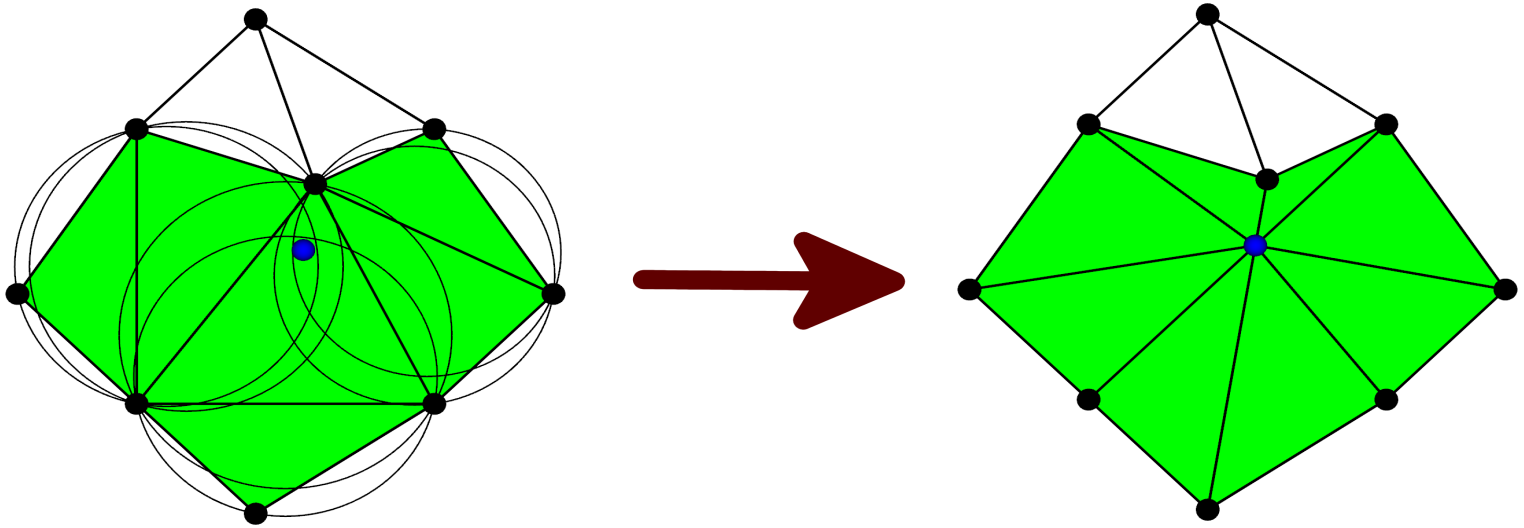


https://www.cs.cmu.edu/~quake/triangle.html

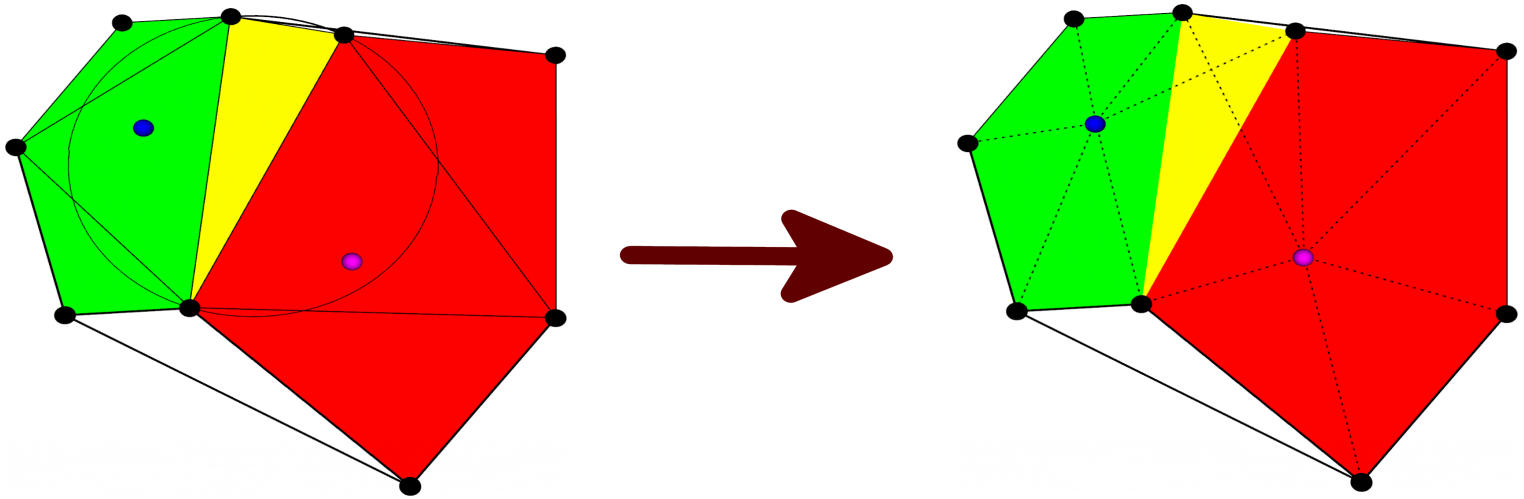


PDR.PODM Delaunay Mesh Example

(Parallel) Delaunay Mesh Refinement



point is inserted



concurrent insertions

Contents

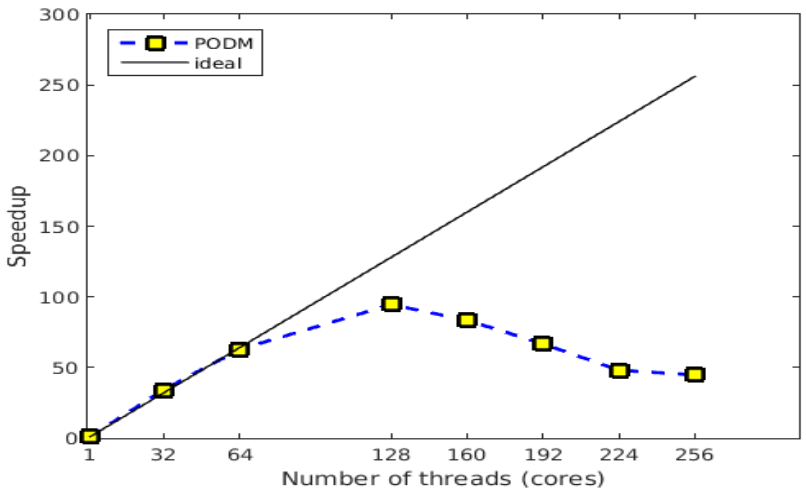
Remote Memory Access

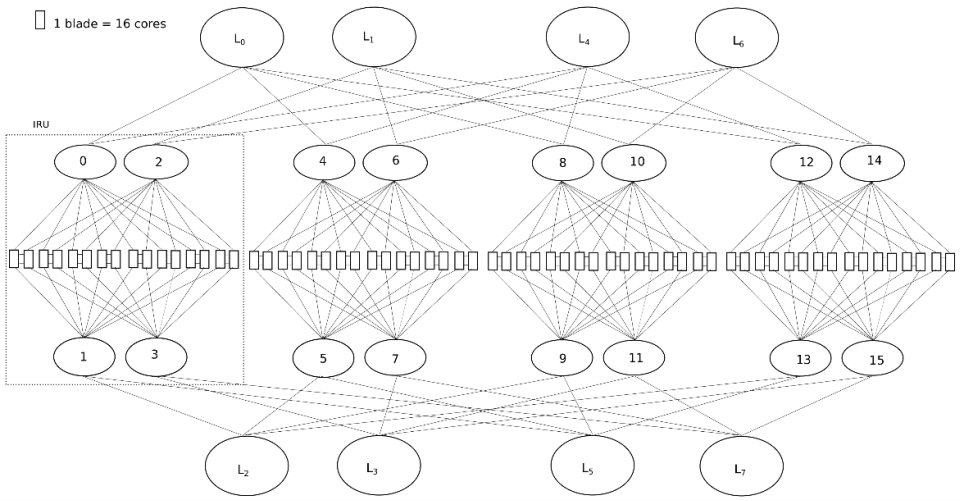
Distributed Shared Memory:

- Distributed Shared Memory (DSM) is a form of memory architecture where the (physically separate) memories can be addressed as one (logically shared) address space.
- Local Memory Access vs Remote Memory Access

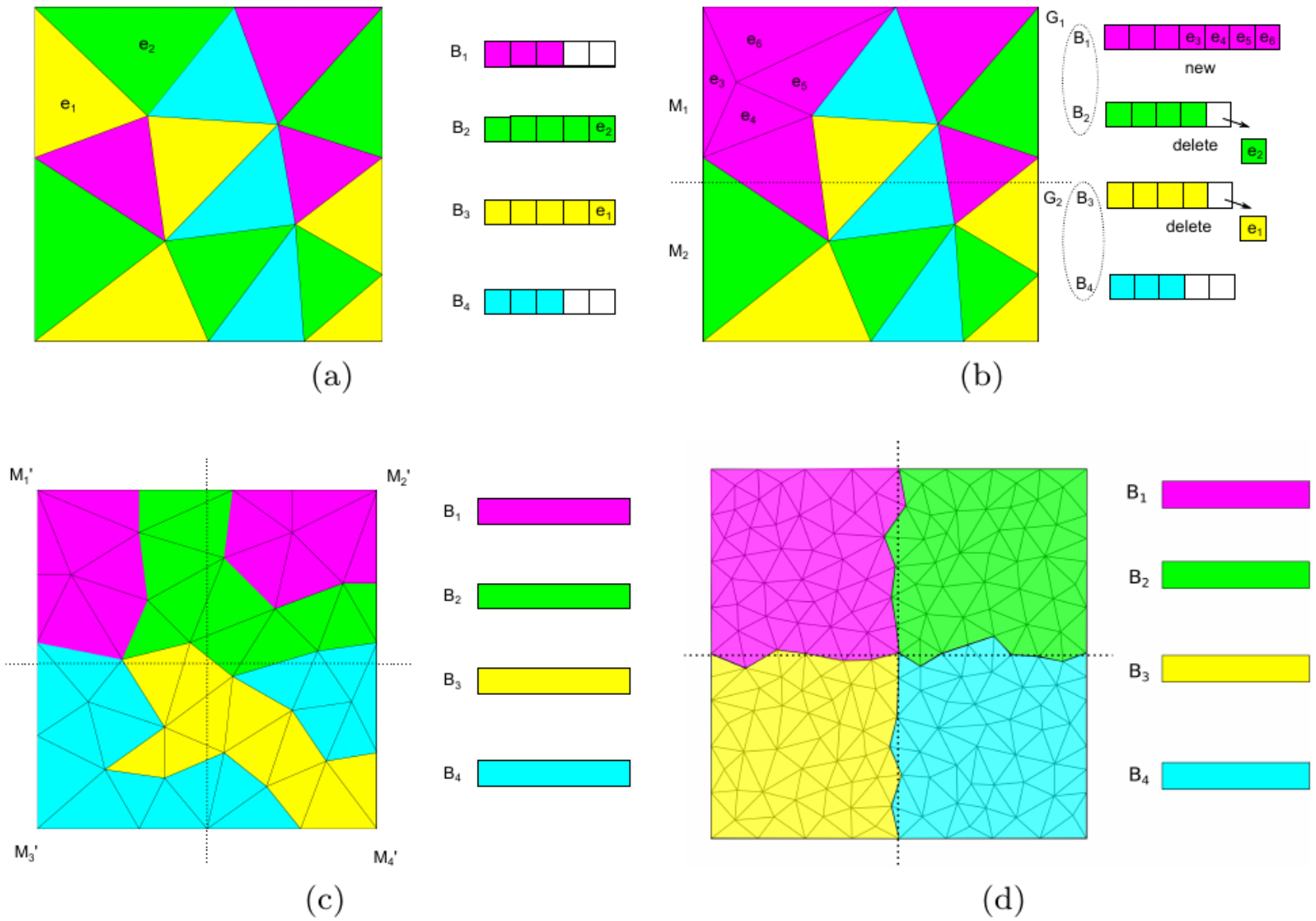
PODM:

- Performance deterioration
- Reasons: remote memory accesses and network congestion

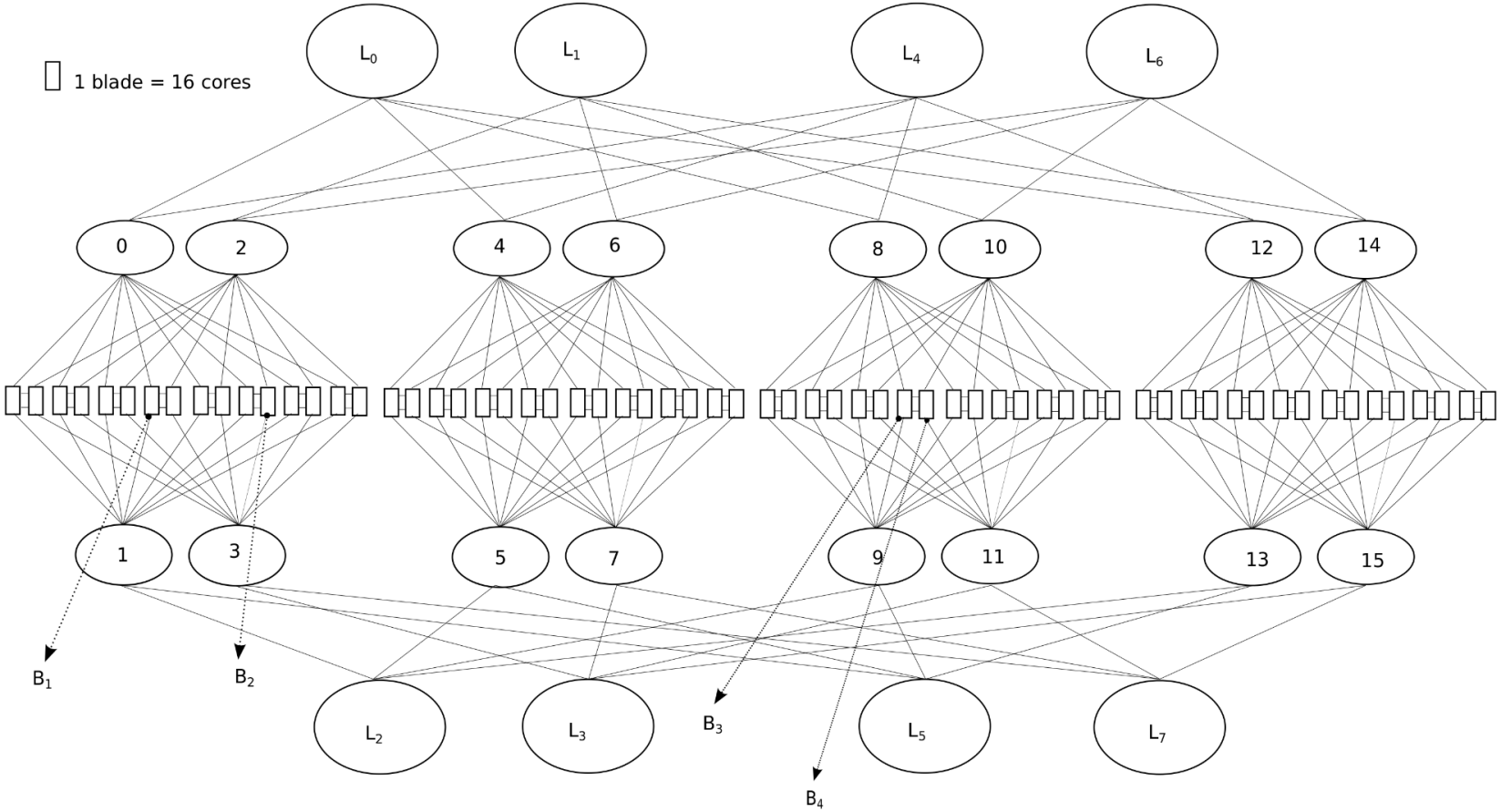




Locality Aware Parallel Delaunay Mesh Generation Algorithm (LAPD)



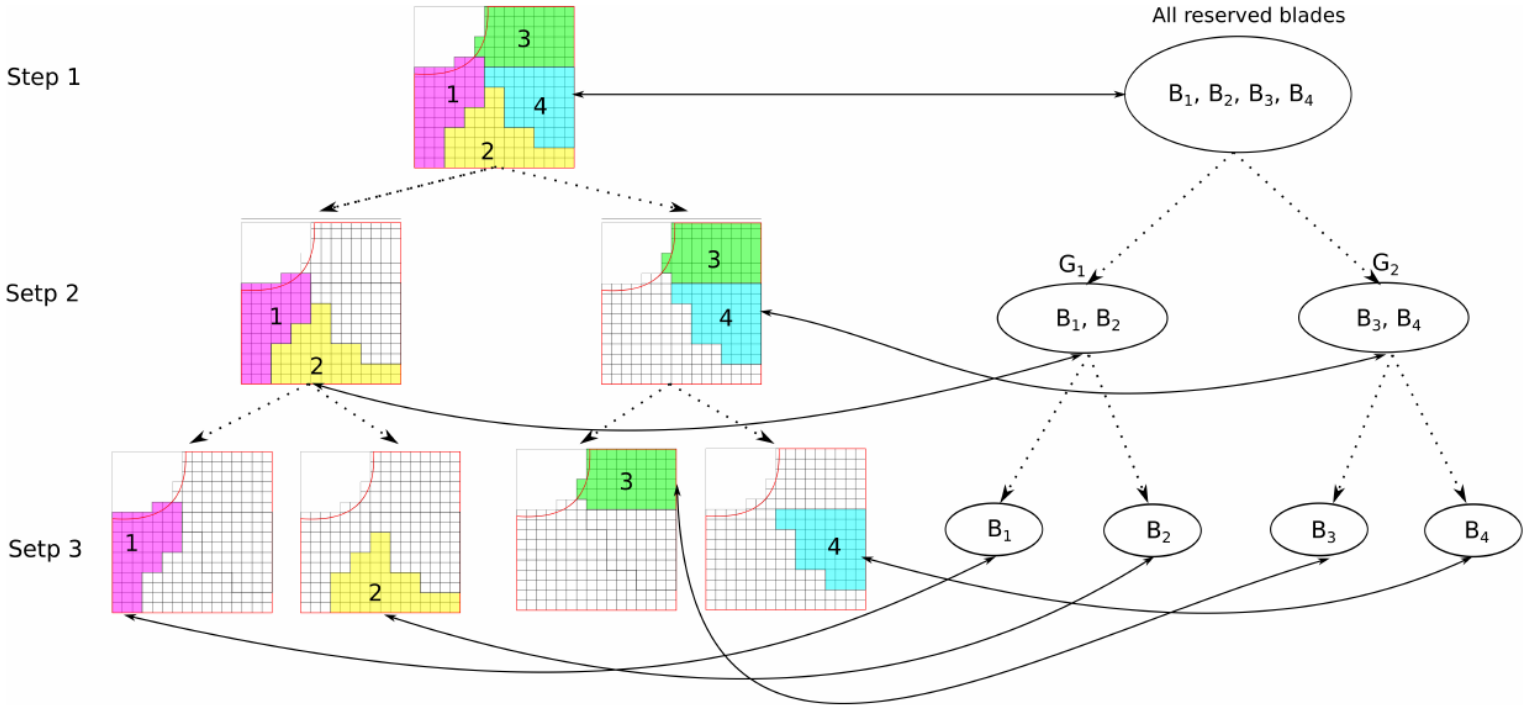
Distributed Shared Memory Architecture



A reservation of 64 cores (4 blades).

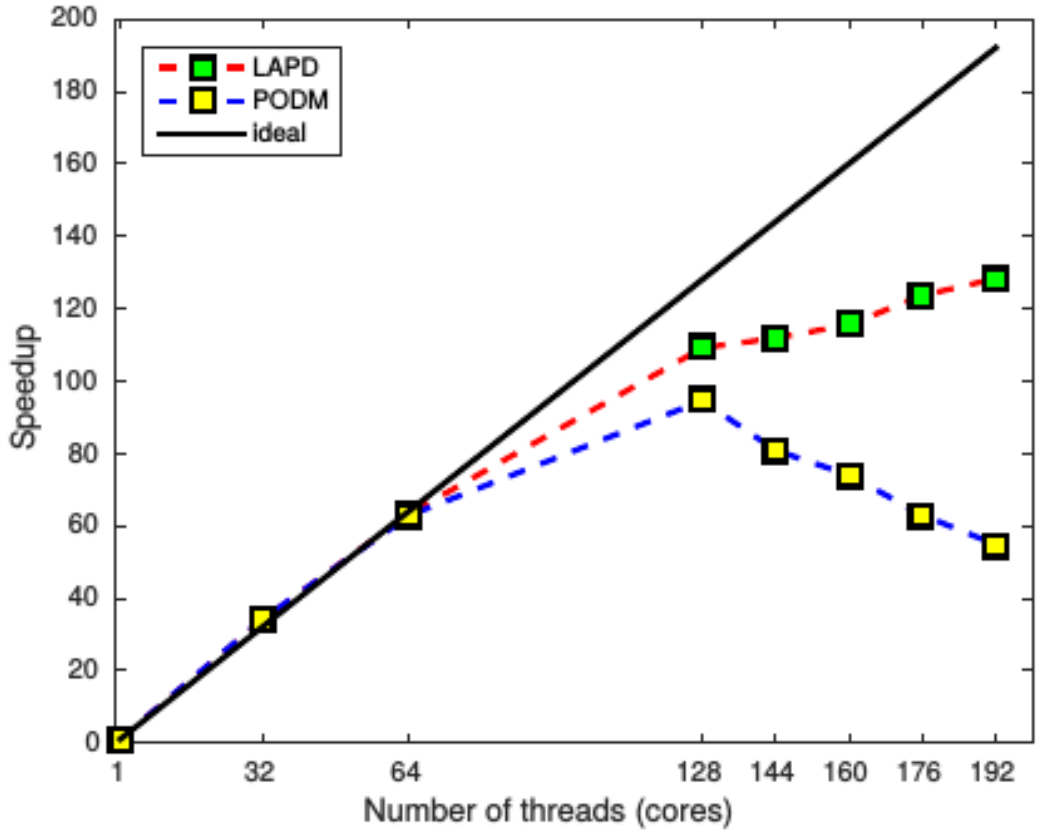
Load Balancing

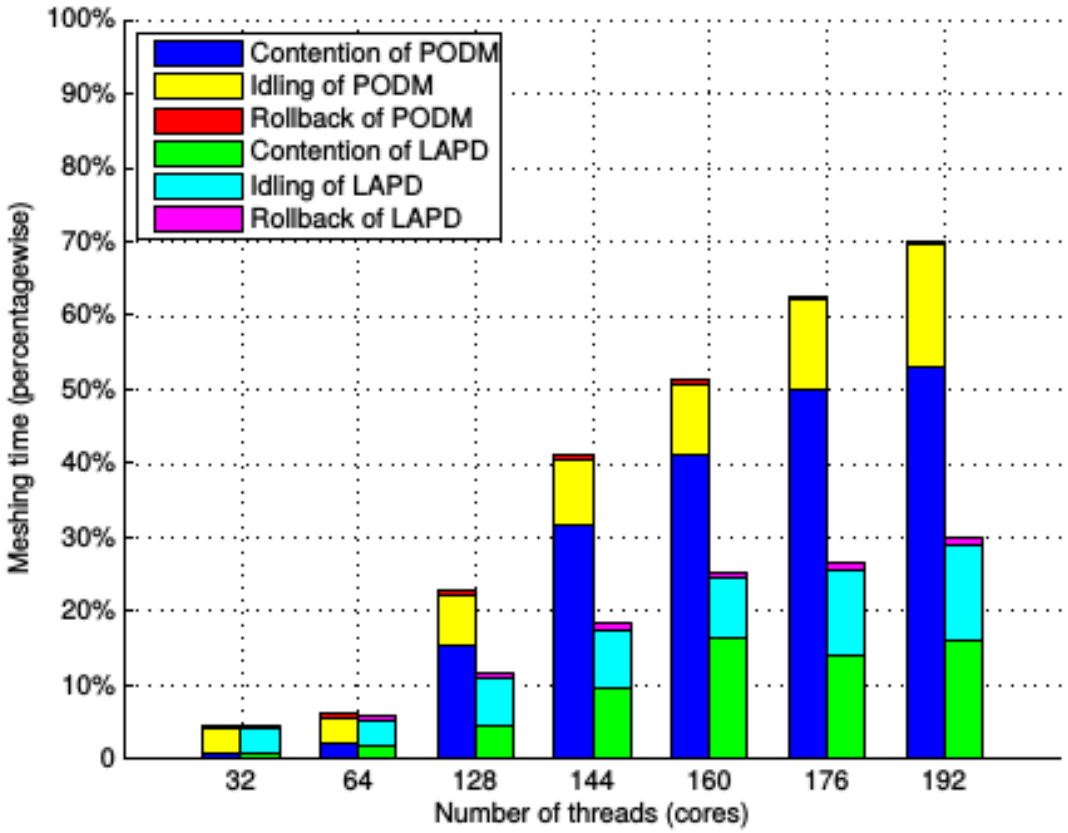
- Over decomposed Block Partition



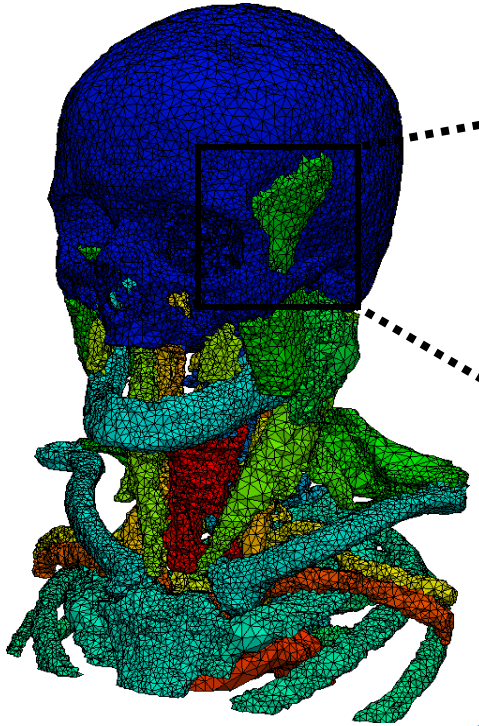
Not a perfect scheme, but solve the problem.

LAPD Speedup and Overhead

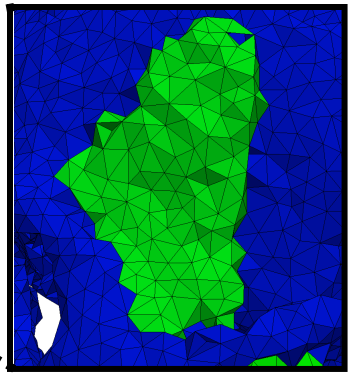


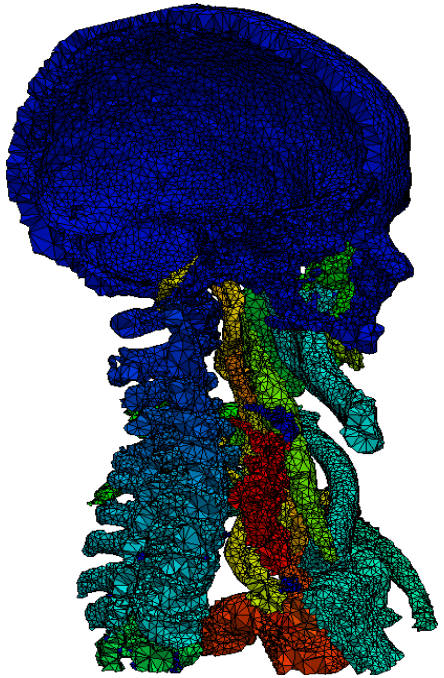


Example Meshes of LAPD



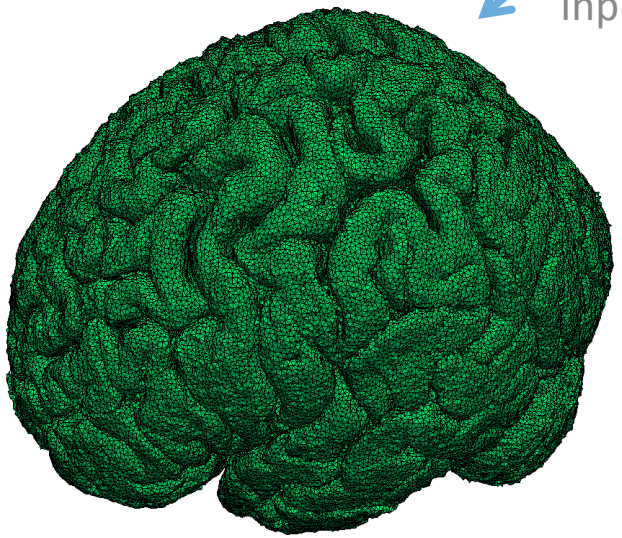
CT Head-neck atlas
Input: Courtesy of SPL





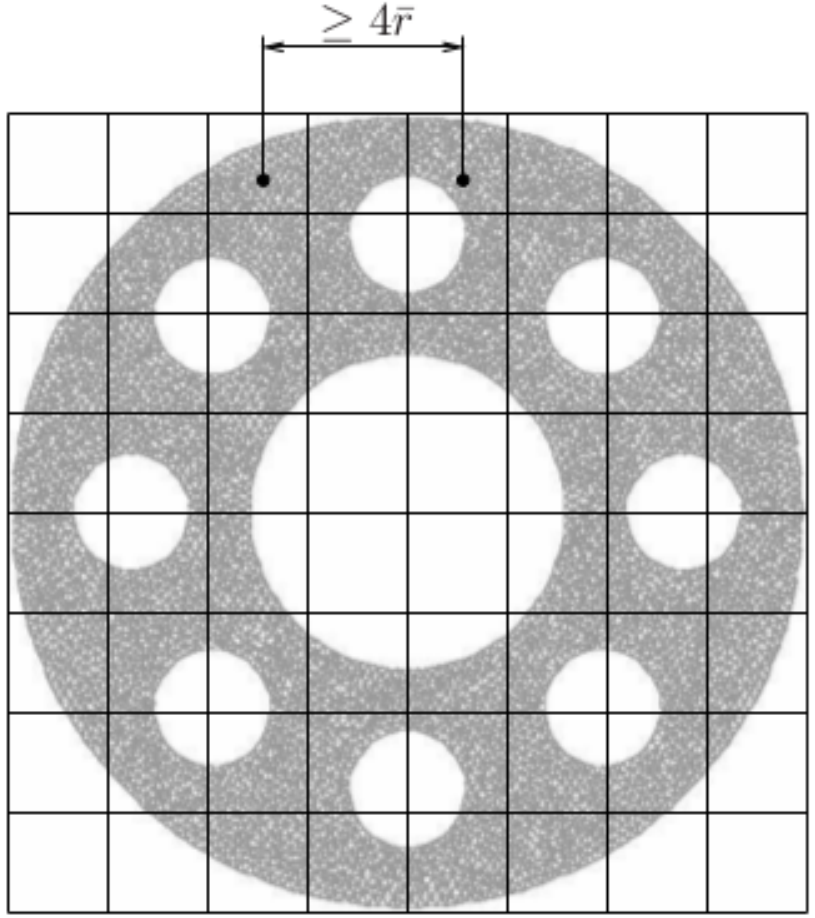
Complex geometries

Big Brain
Input: Courtesy of BigBrain project

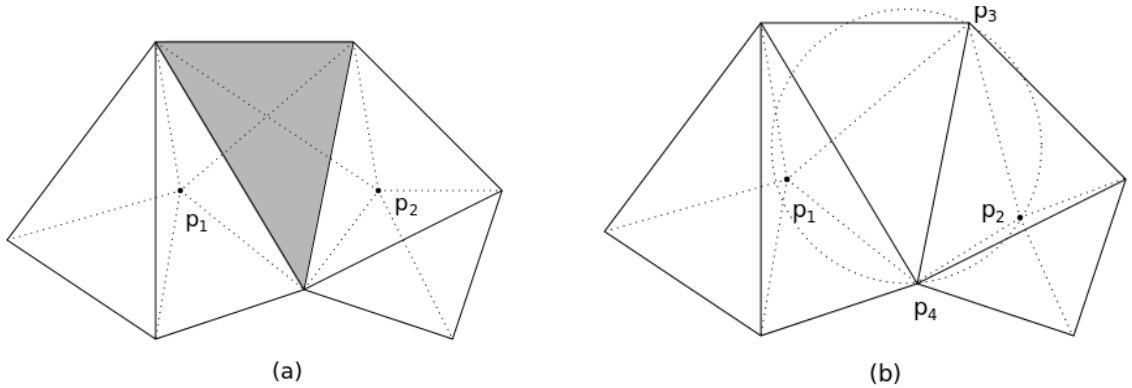


Contents

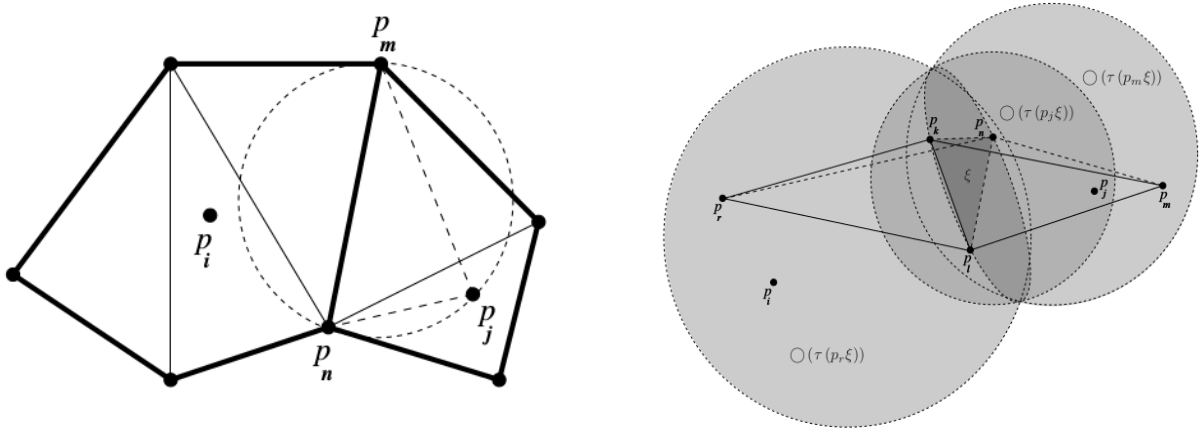
Parallel Delaunay Refinement (PDR)



Concurrent points insertion between two regions



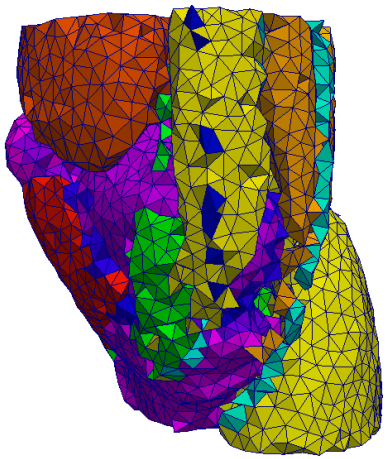
Parallel points insertion results in non-conformal mesh



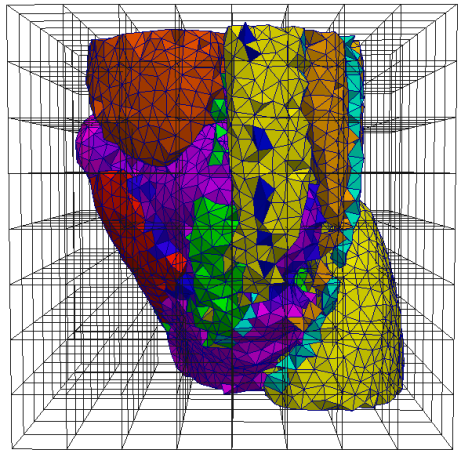
Delaunay-independency of two inserted points

A. Chernikov and N. Chrisochoides, "Parallel guaranteed quality Delaunay uniform mesh refinement," SIAM Journal on Scientific Computing, vol. 28, pp. 1907-1926, 2006.
A. Chernikov and N. Chrisochoides, "Three-dimensional Delaunay refinement for multi-core processors," ACM International Conference on Supercomputing, pp. 214-224, 2008

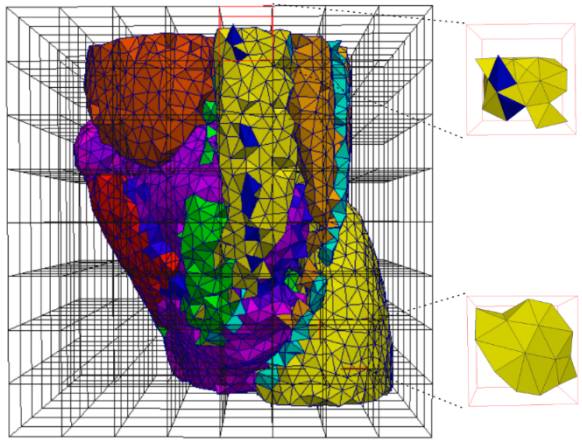
PDR.PODM Algorithm



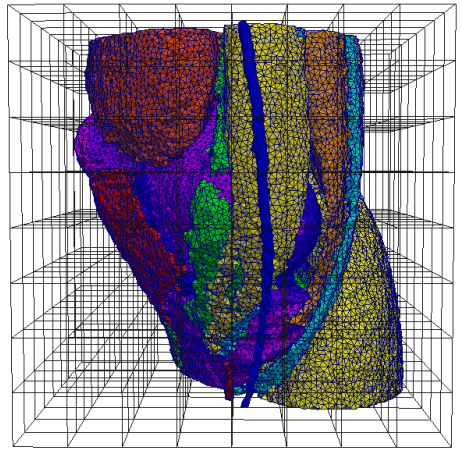
(a) Parallel coarse mesh construction



(b) Coarse mesh distribution

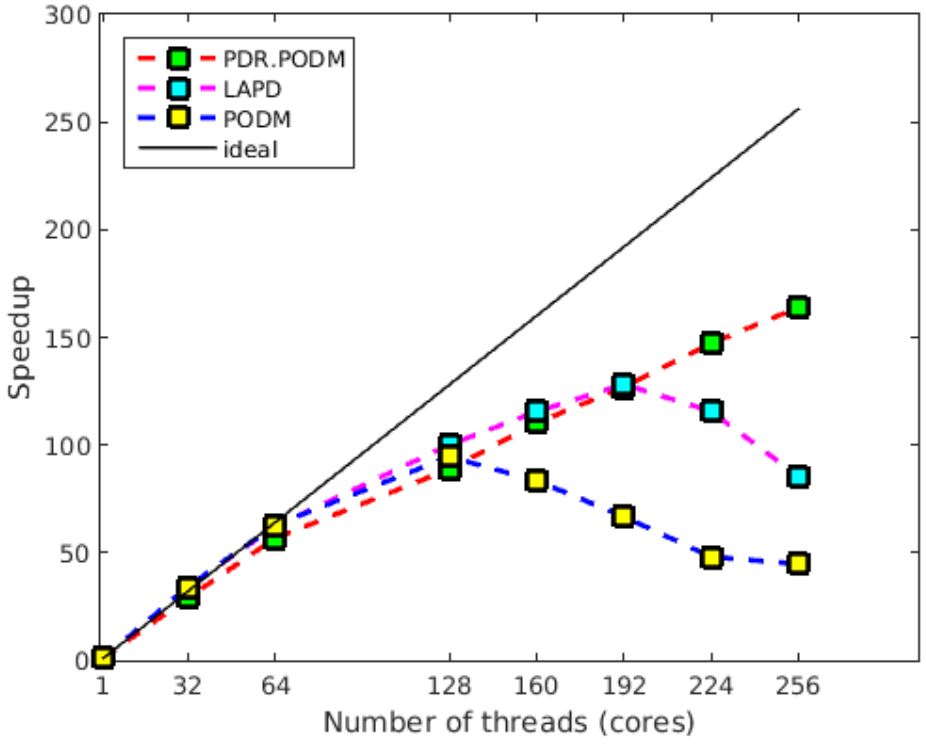


(c) PDR.PODM parallel refinement

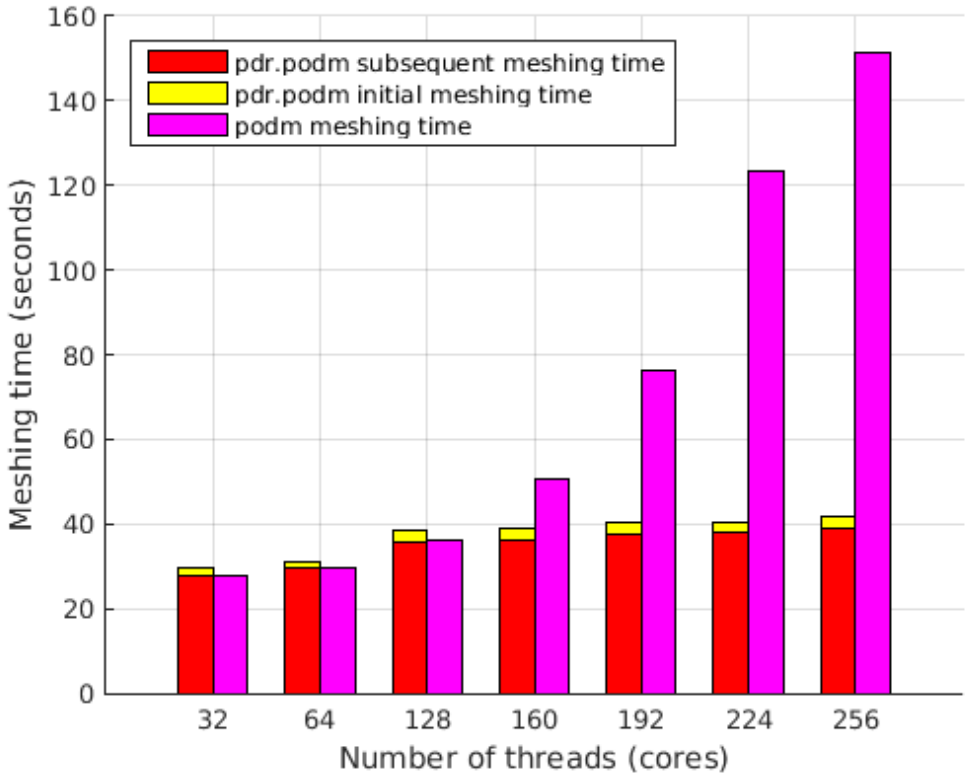


(d) Final Mesh

Speedup and Meshing Time Comparison

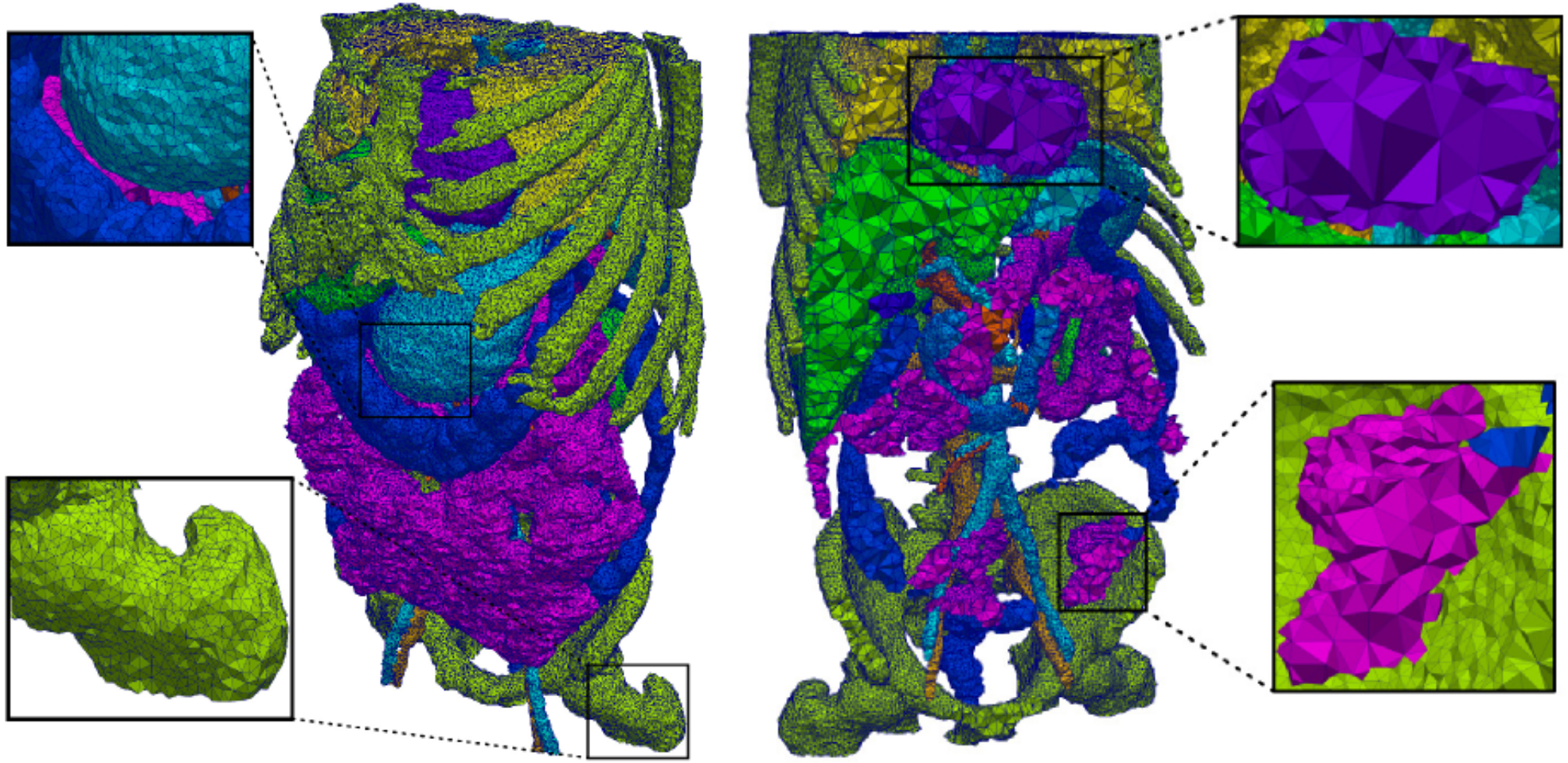


Speedup Comparison



Meshing time of PODM and PDR.PODM

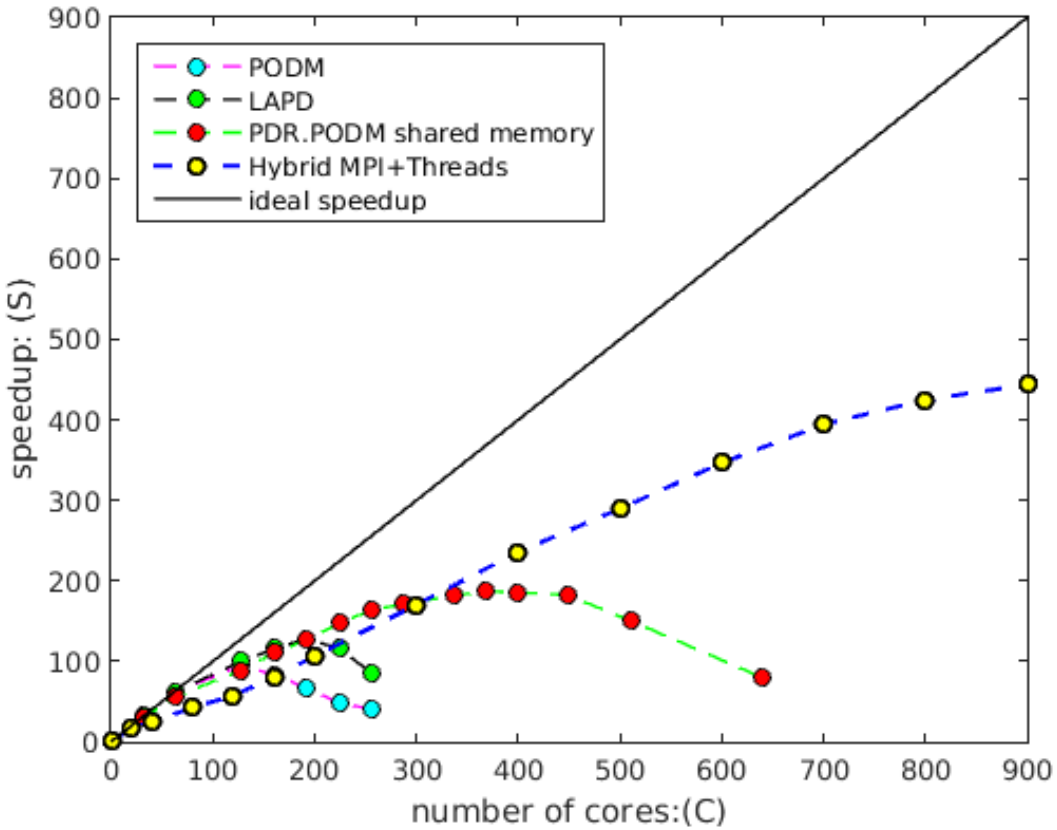
Example Meshes of PDR.PODM



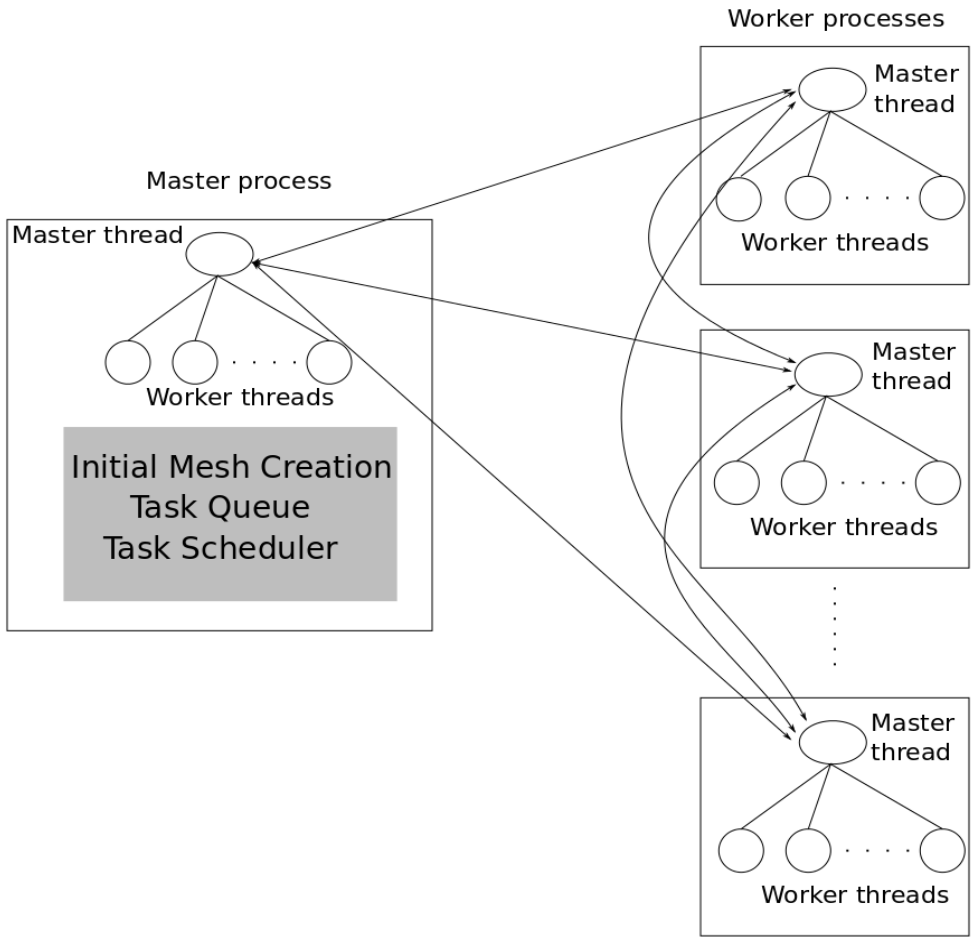
Contents

Hybrid MPI+Threads Meshing Algorithm

- Hybrid Parallel I2M Conversion Algorithm
  - Hybrid MPI+Threads I2M conversion algorithm which takes 3D multi-labeled images as input.
  - The algorithm exploits two levels of parallelization: process level parallelization (which is mapped to a node with multiple cores) and thread level parallelization (each thread is mapped to a single core in a node).
  - It scales well on distributed memory clusters.
  - Nested Master-Worker Model:
    - separates and overlaps communication with computation.

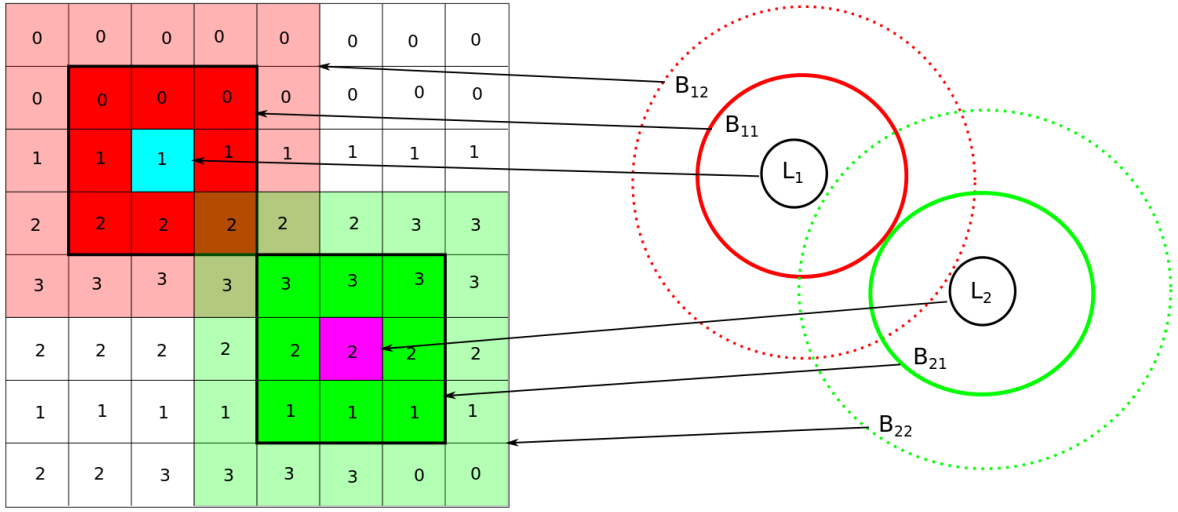


MPI+Threads Parallel I2M Conversion

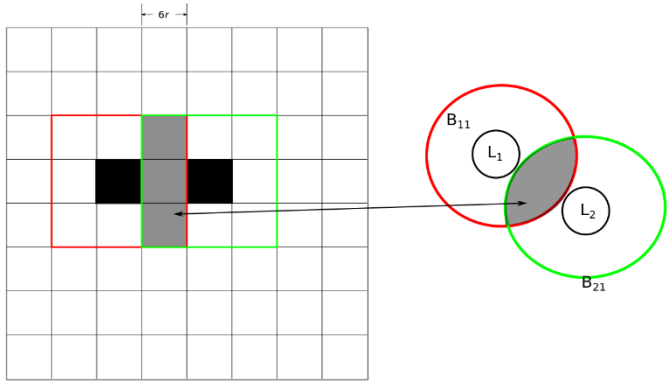


- Nested master-worker model and two level parallelization
  - The master process running on a node creates the coarse mesh, manages and schedules the tasks (subregions) and the worker processes running on other nodes communicate with each other for task request and data migration.
  - Inside each node, the master thread creates multiple worker threads and each of the worker thread runs on one core.
- Overlapping communication with computation
  - The master thread initializes the MPI environment and communicates with the master thread of other processes (communication).
  - The worker threads of each process do not make MPI calls and are only responsible for the local mesh refinement work in the shared memory of each node (computation).

Two-Level Buffer Zones



Two-level buffer zones to avoid the synchronization



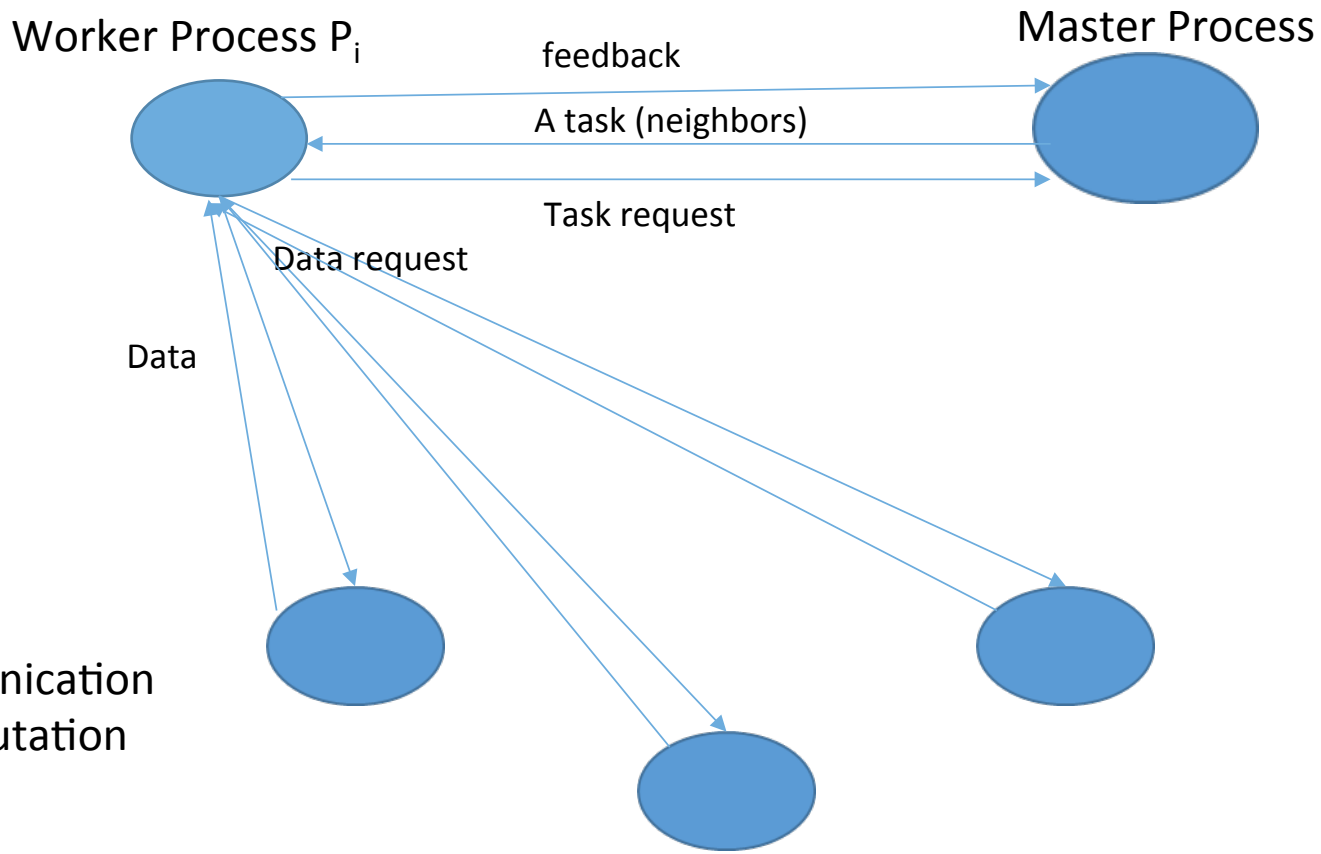
Synchronization is needed in the overlap buffer zones

Two level neighbors

- First level buffer:
  - The submeshes inside first level neighbors need to be moved to the local memory of working process.
- Second level buffer:
  - Select and schedule independent subregions to multiple processes to avoid synchronization during refinement.
- Integer Flag
  - The process rank (node ID): where the submesh inside the subregion is stored.

Inter-node Communication and Data Exchange (MPI)

Task: subregion
Data: submesh

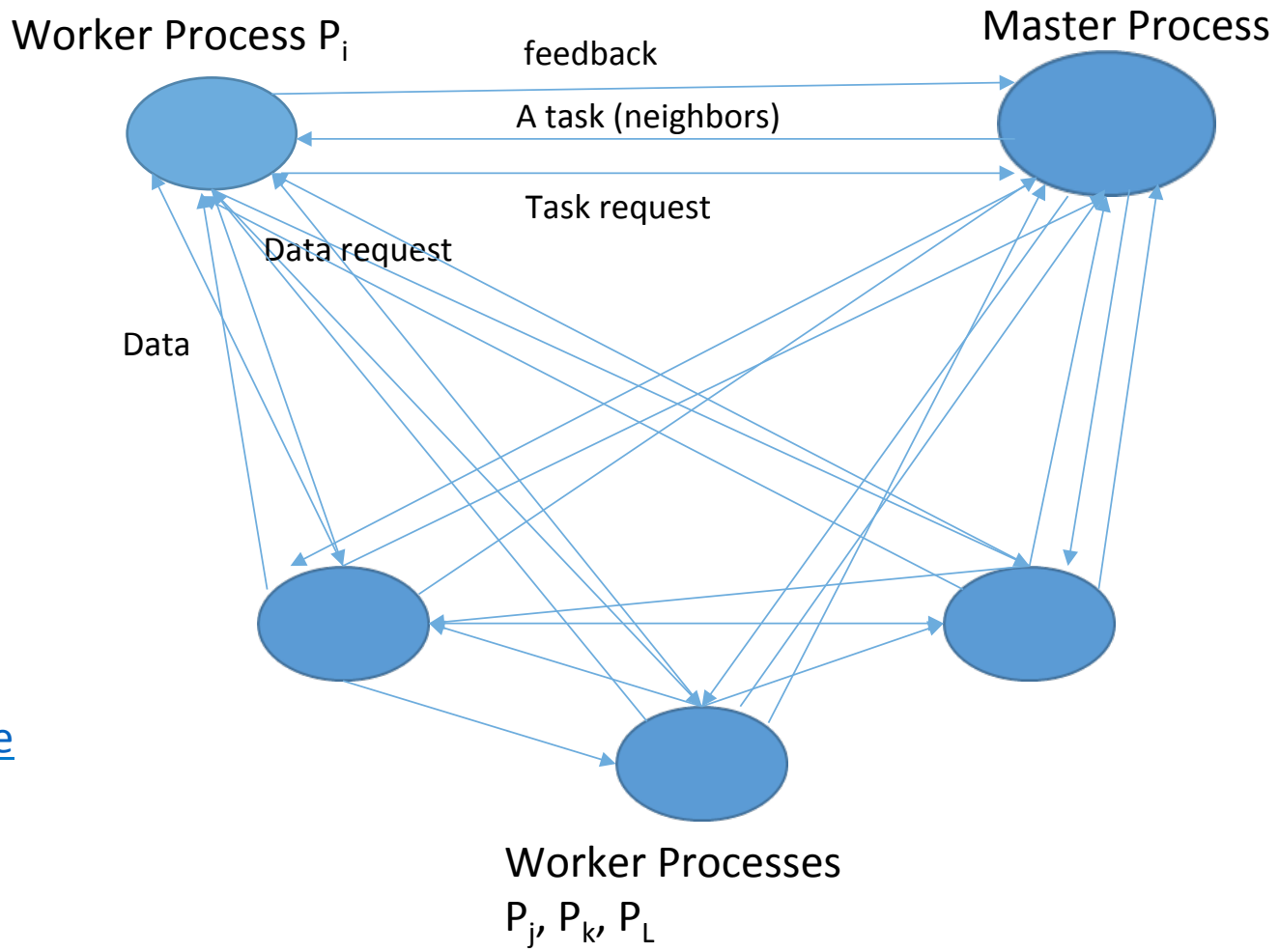


Master thread: communication
Worker threads: computation

Worker Processes
P_j, P_k, P_L

Inter-node Communication and Data Exchange (MPI)

Task: subregion
data: submesh



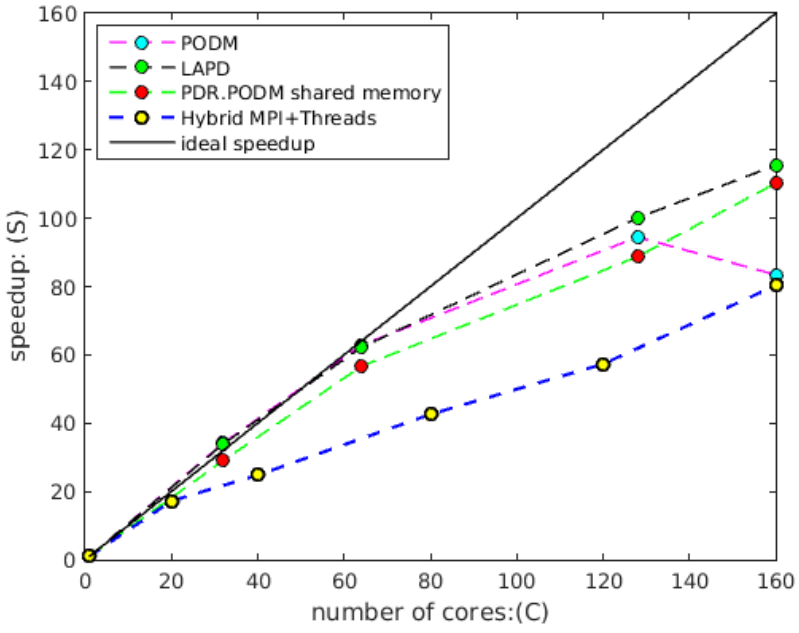
[Pseudo-code]

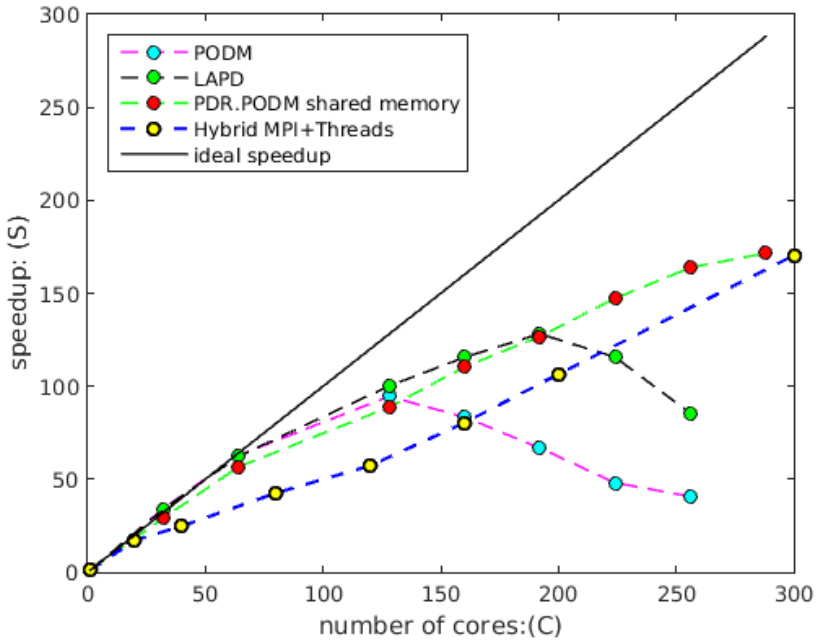
Performance

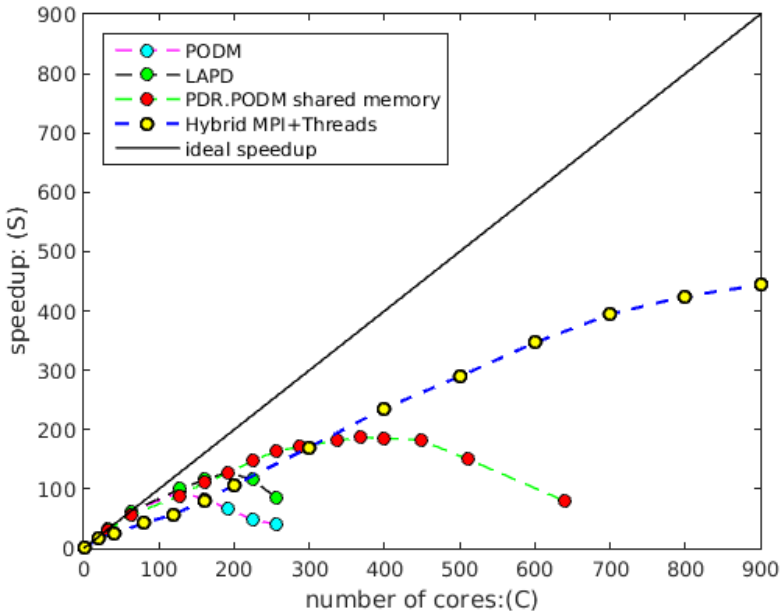
# Cores	# Elements (million)	Running Time (s)	Elements/s (m/s)	Speedup	Efficiency	depth
1	6.63	65.13	0.10	1.0	100.00%	-
20	133.87	76.33	1.75	17.21	86.03%	-
40	261.56	103.64	2.52	24.76	61.90%	3
80	520.30	119.88	4.34	42.58	53.22%	3
120	780.99	133.72	5.84	57.30	47.75%	4
160	1034.82	126.44	8.18	80.29	50.18%	4
200	1292.20	119.05	10.85	106.49	53.24%	4
300	1937.04	111.57	17.36	170.33	56.78%	4
400	2577.94	107.19	24.05	235.95	58.99%	4
500	3217.94	109.02	29.51	289.57	57.91%	4
600	3858.97	109.22	35.33	346.61	57.77%	4
700	4501.39	111.97	40.19	394.97	56.34%	4
800	5143.09	118.77	43.30	424.82	53.10%	4
900	5776.34	127.66	45.25	445.89	50.10%	4

It creates 45.25 million elements per second when runs with 900 cores.

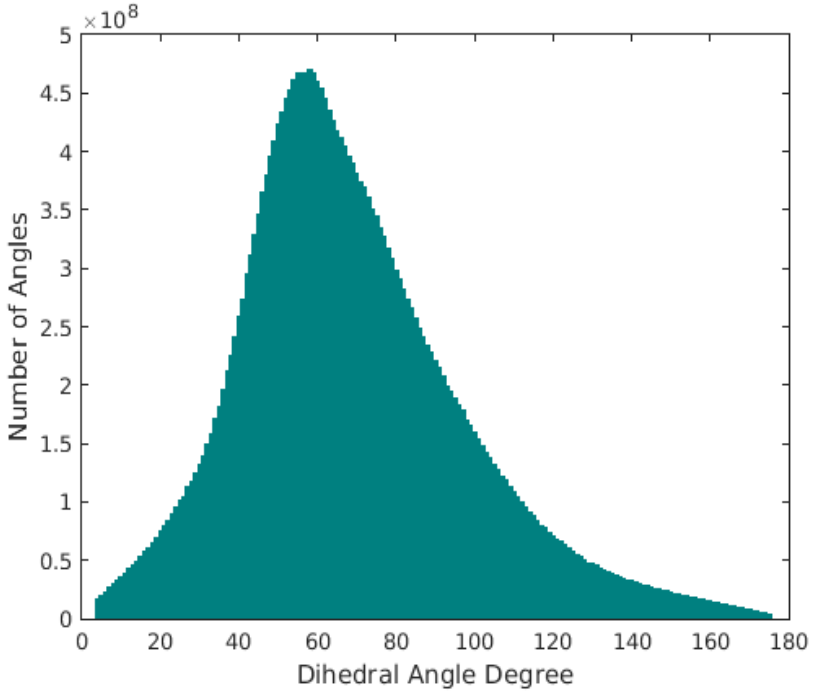
Performance Comparison

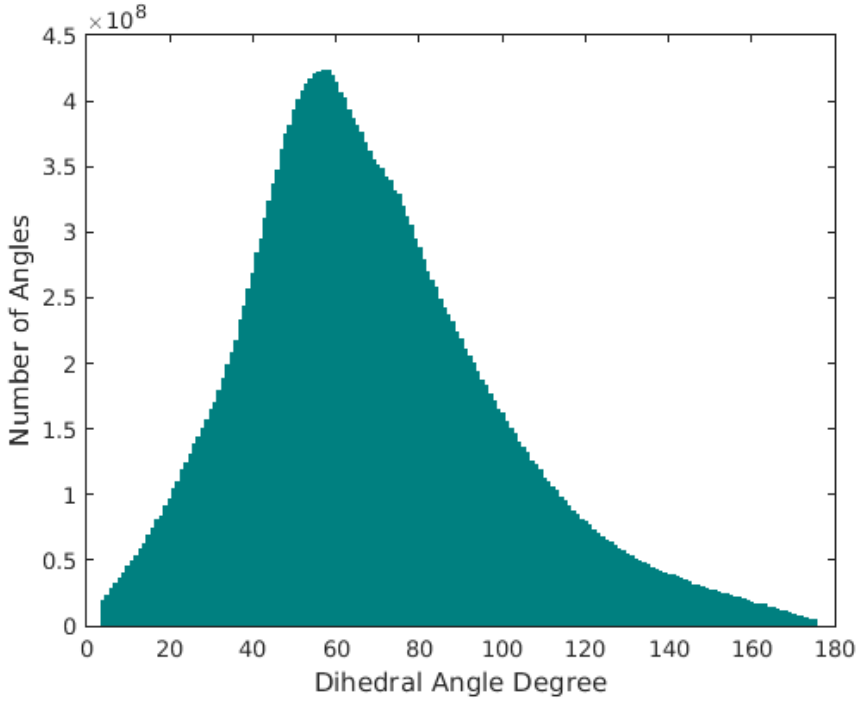






Quality





Dihedral angle range abdominal:
Hybrid Method: (4.7°, 172.1°)

Dihedral angle range knee:
Hybrid Method: (4.8°, 171.7°)

Contents

Conclusion

Conclusion:

- LAPD:
  - Employs a data locality-aware parallel mesh refinement procedure to reduce the communication overhead caused by the remote memory accesses.
- PDR.PODM:
  - Combines the best features of two previous algorithms.
- Hybrid MPI+Threads
  - Explores two levels of parallelization: process level parallelization among nodes and thread level parallelization inside node.
  - Separate the MPI communication and local shared memory mesh refinement.

Future Work:

- Optimize current implementation and improve the scalability.
  - Utilize the current mesh generation software (like MOAB and MeshKit).
  - Explore the Index-based customized layer to avoid the explicit mesh format translation.
- Explore the hybrid MPI+Thread implementation of parallel mesh generation algorithm on heterogeneous node cluster.

Publications

Journal Papers:

Daming Feng, Andrey Chernikov and Nikos Chrisochoides, "A Hybrid Parallel Delaunay Image-to-Mesh Conversion Algorithm Scalable on Distributed-Memory Clusters", Invited in Computer-Aided Design, Feb. 2017.

Daming Feng, Christos Tsolakis, Andrey Chernikov and Nikos Chrisochoides, "Scalable 3D Hybrid Parallel Delaunay Image-to-Mesh Conversion Algorithm for Distributed Shared Memory Architectures", Computer-Aided Design, July 2016.

Daming Feng, Andrey Chernikov and Nikos Chrisochoides, "Two-level Locality-Aware Parallel Delaunay Image-to-Mesh Conversion", Parallel Computing, February 2016.

Wenlu Zhang, Daming Feng, Rongjian Li, et. al, "A mesh generation and machine learning framework for Drosophila gene expression pattern image analysis", Journal BMC Bioinformatics, 2013.

Wenlu Zhang, Rongjian Li, Daming Feng, Andrey Chernikov, Nikos Chrisochoides, Christopher Osgood and Shuiwang Ji, "Evolutionary soft co-clustering: formulations, algorithms, and applications", Data Mining and Knowledge Discovery, August 2014.

Conference Papers and Posters:

Daming Feng, Andrey Chernikov and Nikos Chrisochoides, "A Hybrid Parallel Delaunay Image-to-Mesh Conversion Algorithm Scalable on Distributed-Memory Clusters", 2016 International Meshing Roundtable, Washington, D.C., USA, Sep 27-30, 2016.

Daming Feng, Christos Tsolakis, Andrey Chernikov and Nikos Chrisochoides, "Scalable 3D Hybrid Parallel Delaunay Image-to-Mesh Conversion Algorithm for Distributed Shared Memory Architectures", 2015 International Meshing Roundtable (IMR24), Austin, TX, October 11-14, 2015.

Daming Feng, Andrey Chernikov, and Nikos Chrisochoides, “Preparing for exascale parallel mesh generation”, Modeling, Simulation, and Visualization Student Capstone Conference, pages 14-20. Suffolk, VA, April 2015.

Nikos Chrisochoides, Andrey Chernikov, Daming Feng, Christos Tsolakis and Scott Pardue, "Toward Exascale-Era Parallel Mesh Generation", 13th U.S. National Congress on Computational Mechanics, July, 2015

Nikos Chrisochoides, Andrey Chernikov, Daming Feng and Christos Tsolakis "Extreme-Scale Parallel Mesh Generation: Telescopic Approach", 2015 International Meshing Roundtable (IMR24), Austin, TX, October 11-14, 2015.

Panagiotis Foteinos, Daming Feng, Andrey Chernikov and Nikos Chrisochoides, "Multi-Layered Unstructured Mesh Generation", 27th ACM International Conference on Supercomputing (ICS'13), June, 2013.

Kevin Garner, Daming Feng, Fotios Drakopoulos, Yixun Liu and Nikos Chrisochoides, "Image-to-Mesh Conversion Tool", 2015 International Meshing Roundtable (IMR24), Austin, TX, October 11-14, 2015

Thank you!