Domain Decomposition Methods under Constraints in CFD Simulations

MODELING SEMINAR 2024

Team Members:

Pratik Suryawanshi

Akshat Shukla

Arla Gracia Diengdoh

Project Co-ordinator:

Dr. Maximilian Bauer

CONTENT

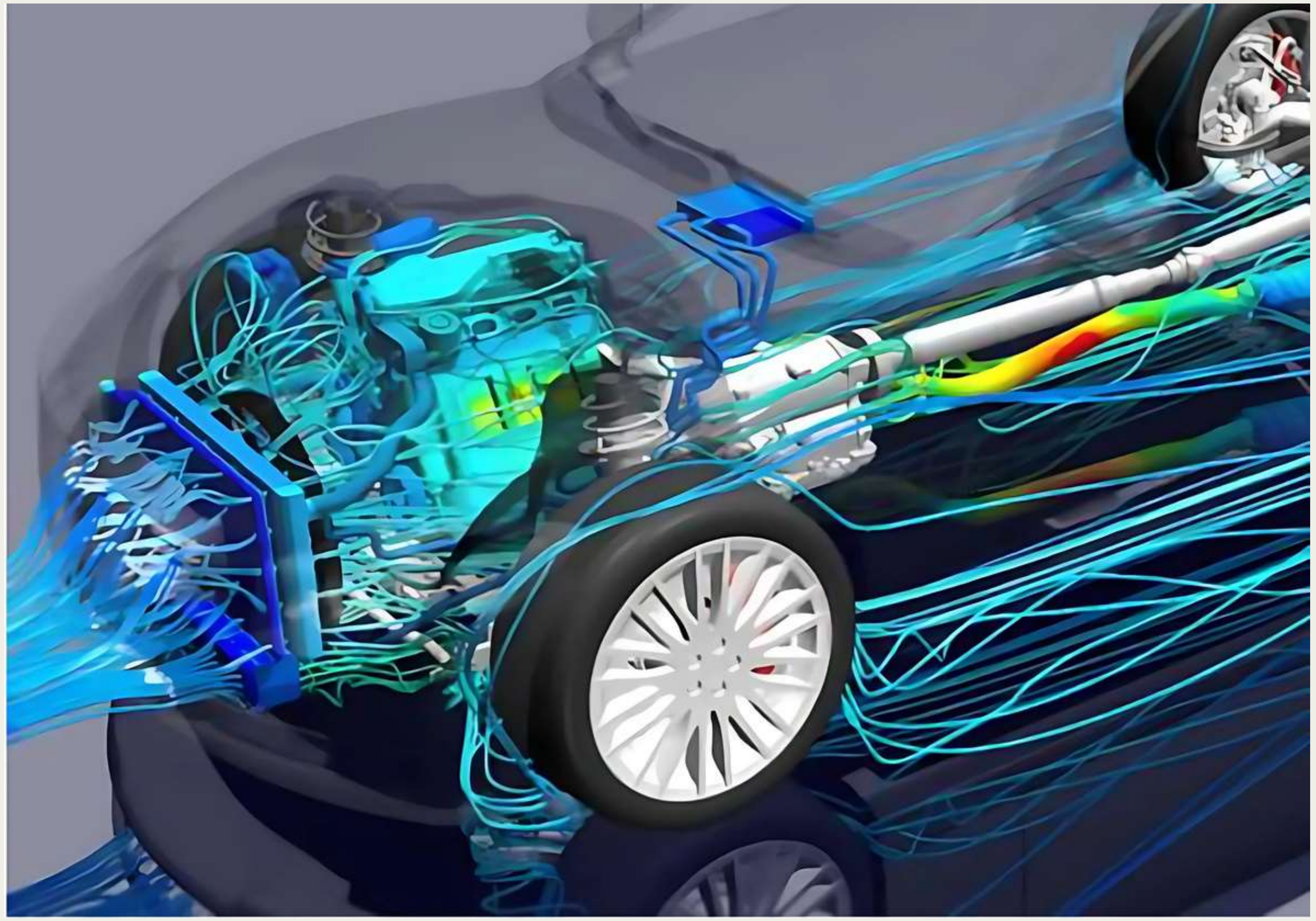
CONTENT

> Problem Statement & Objectives

- Graph Theory & Tools
- Geometric Partitioning
- Conclusion & Scope

THE PROBLEM STATEMENT

The challenge is to partition the mesh while minimizing the communication with the interface cells.



Sample Simulation from STAR-CCM+

OBJECTIVES

- Develop partitioning algorithm with **interface continuity**
- Ensure **computational efficiency**
- Evaluate **existing methods**
- Improve **load distribution** and reduce **communication overhead**

CONTENT

- Problem Statement & Objectives

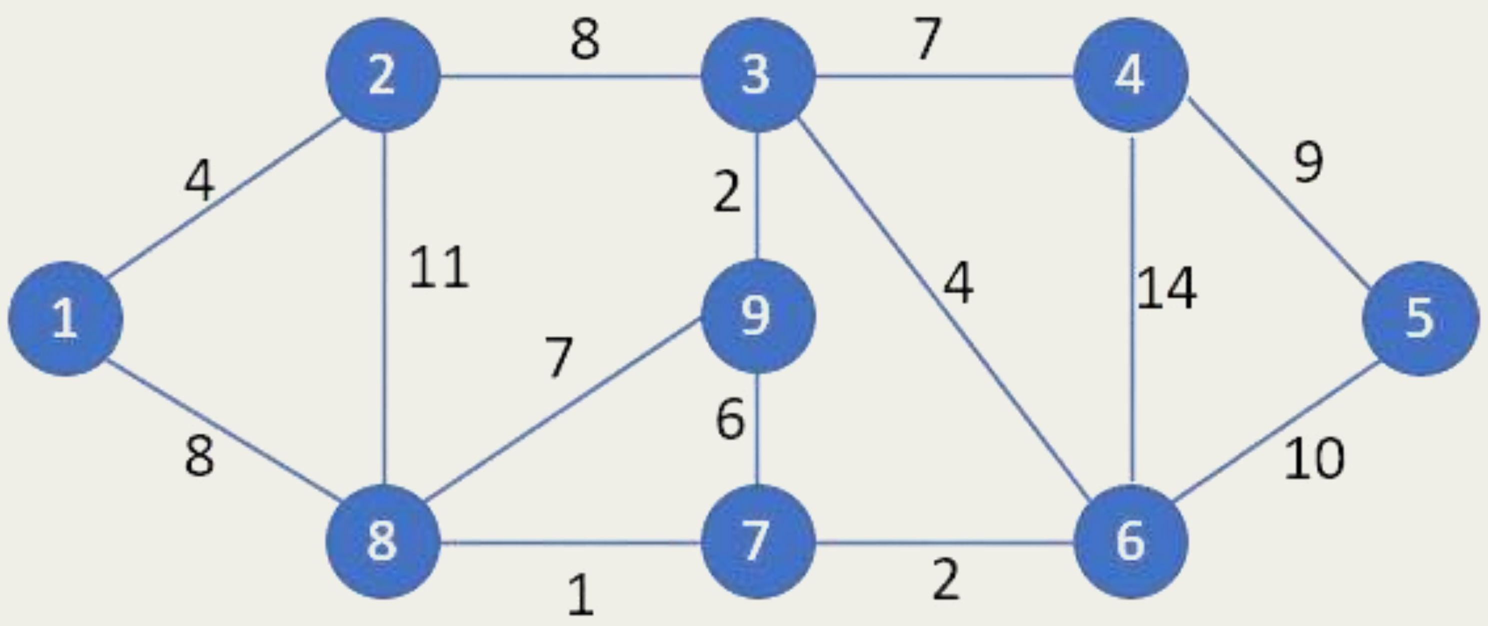
> Graph Theory & Tools

- Geometric Partitioning
- Conclusion & Scope

GRAPH THEORY

A graph is an ordered pair G=(V,E) comprising:
V, a set of vertices (also called nodes or points);

$E \subseteq \{\{x, y\} \mid x, y \in V \text{ and } x \neq y\}$ is a set of edges



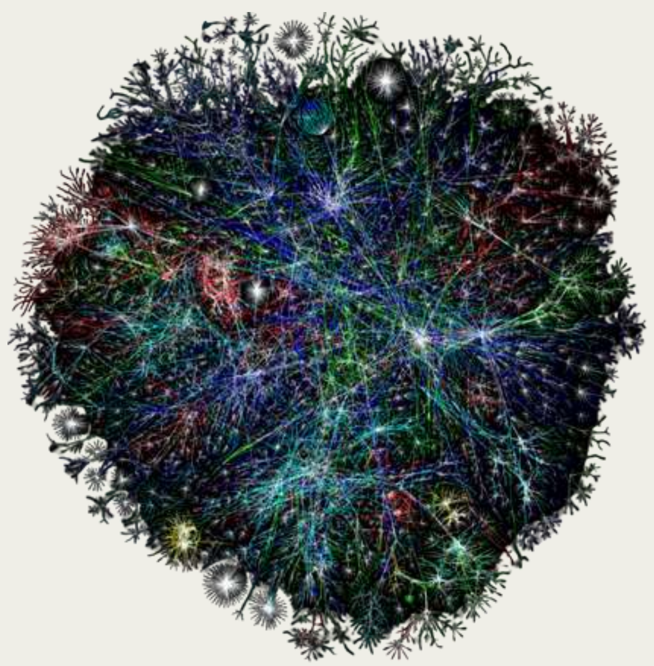


Fig. 1. Graphs and Nodes

GRAPH PARTITIONING

A graph partition is the reduction of a graph to a smaller graph by partitioning its set of nodes into mutually exclusive groups.

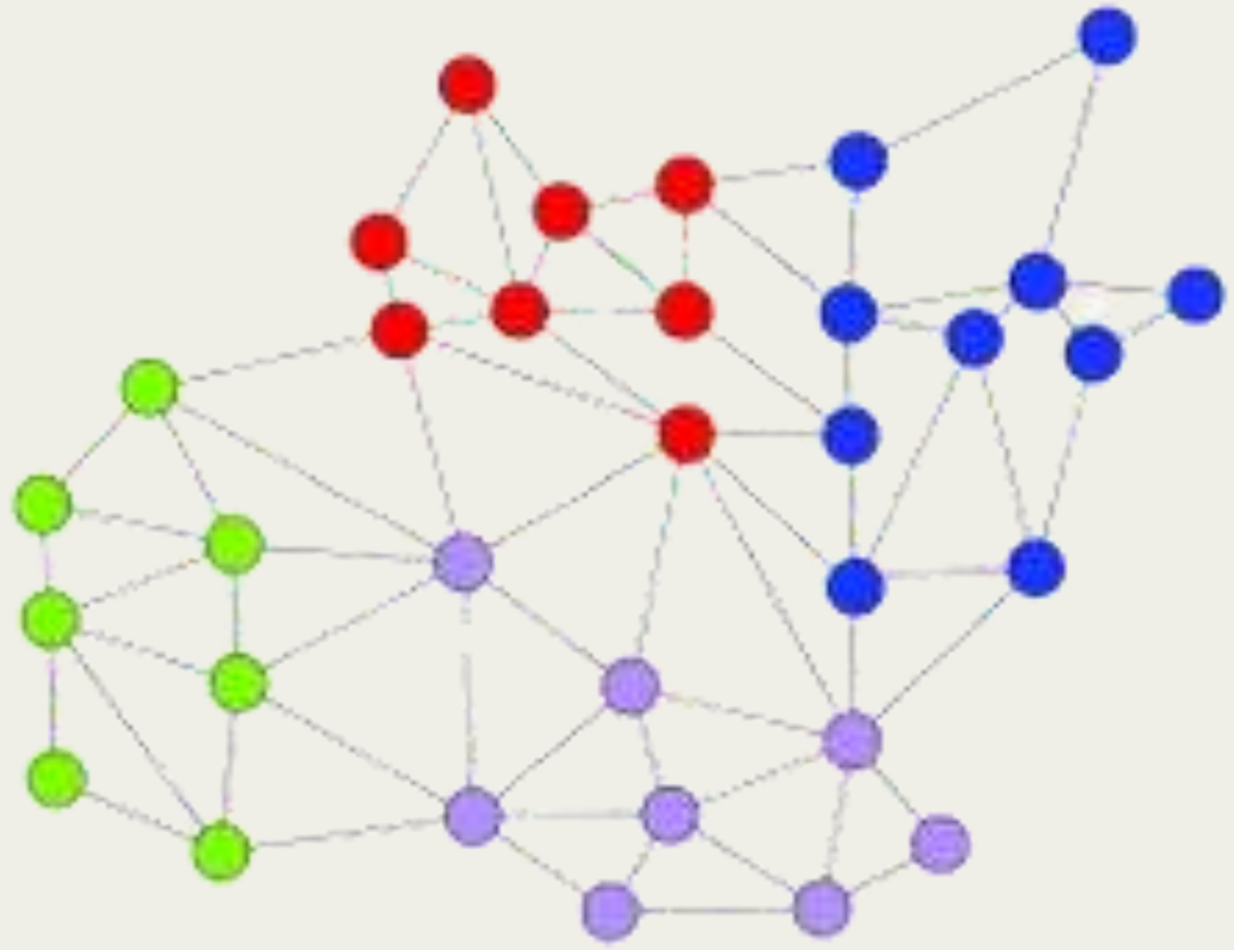


Fig. 2. Graph Partitioning

METIS - Serial Graph Partitioning and Fill-reducing Matrix Ordering

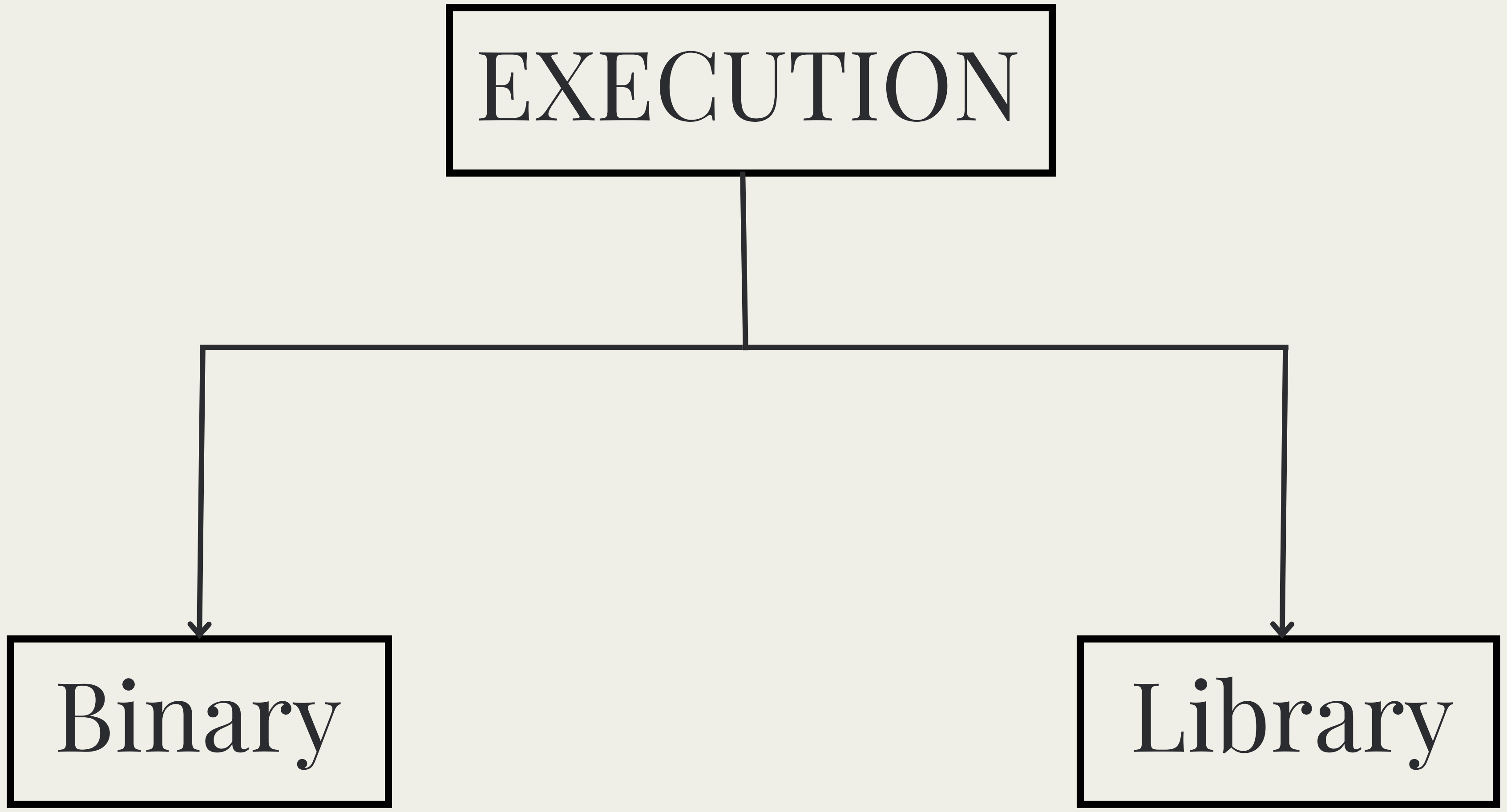


Fig. 3. METIS Execution Schematic

METIS

Multi-level graph partitioning

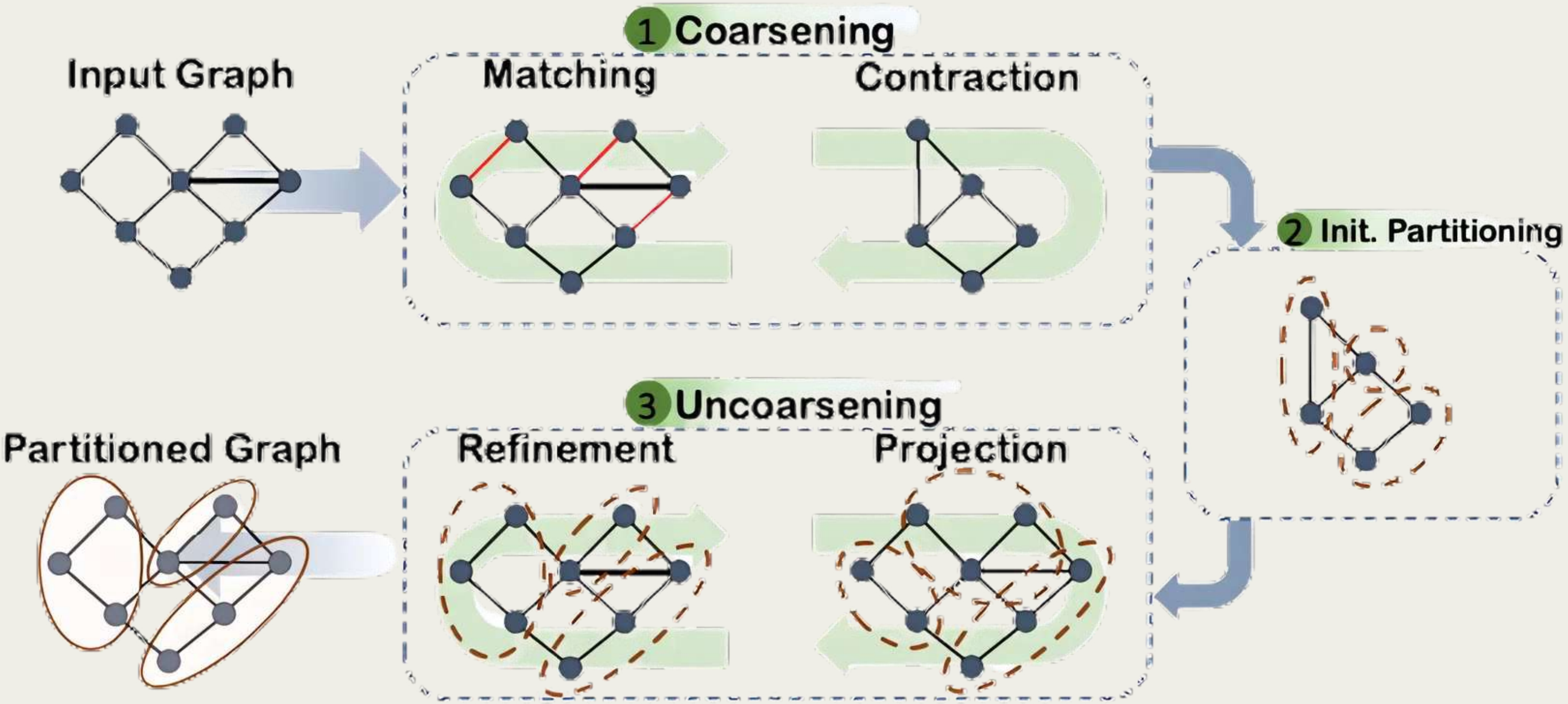


Fig. 4. Multi-level Graph Partitioning Process

GMSH FILE STRUCTURE

“**Gmsh**, an automatic three-dimensional finite element mesh generator with built-in pre- and post-processing facilities.”

Gmsh file structure

INITIAL APPROACHES AND ATTEMPTS

Conversion of .vtk to .graph

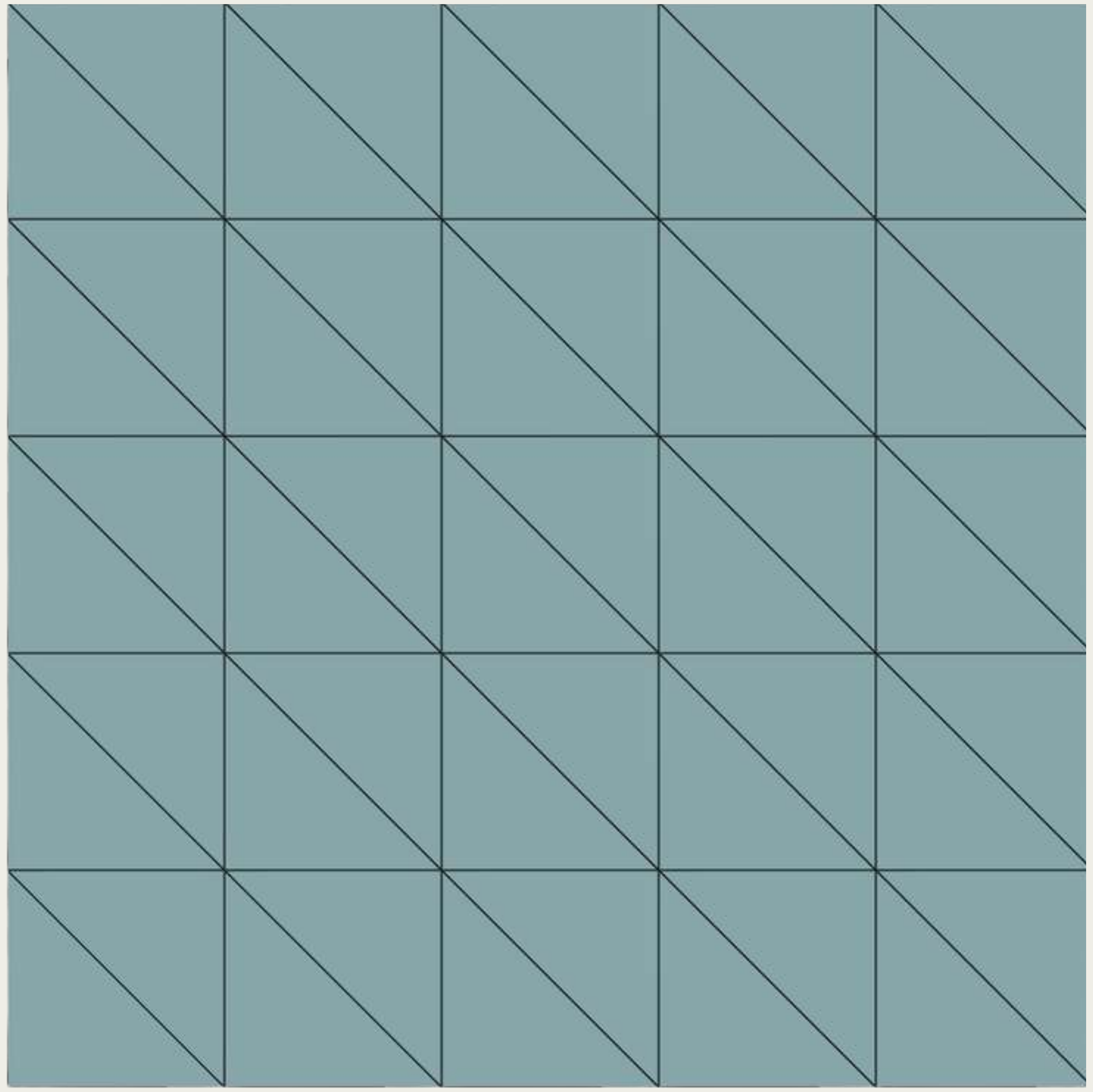


Fig. 5. Mesh - Courant Triangulation

.VTK FILE STRUCTURE

```
# vtk DataFile Version 3.0
Algorithm Sample
ASCII
DATASET POLYDATA
POINTS 36 double
0 0 0
0.2 0 0
0.4 0 0
0.6 0 0
0.8 0 0
...
...
...
...
...
```

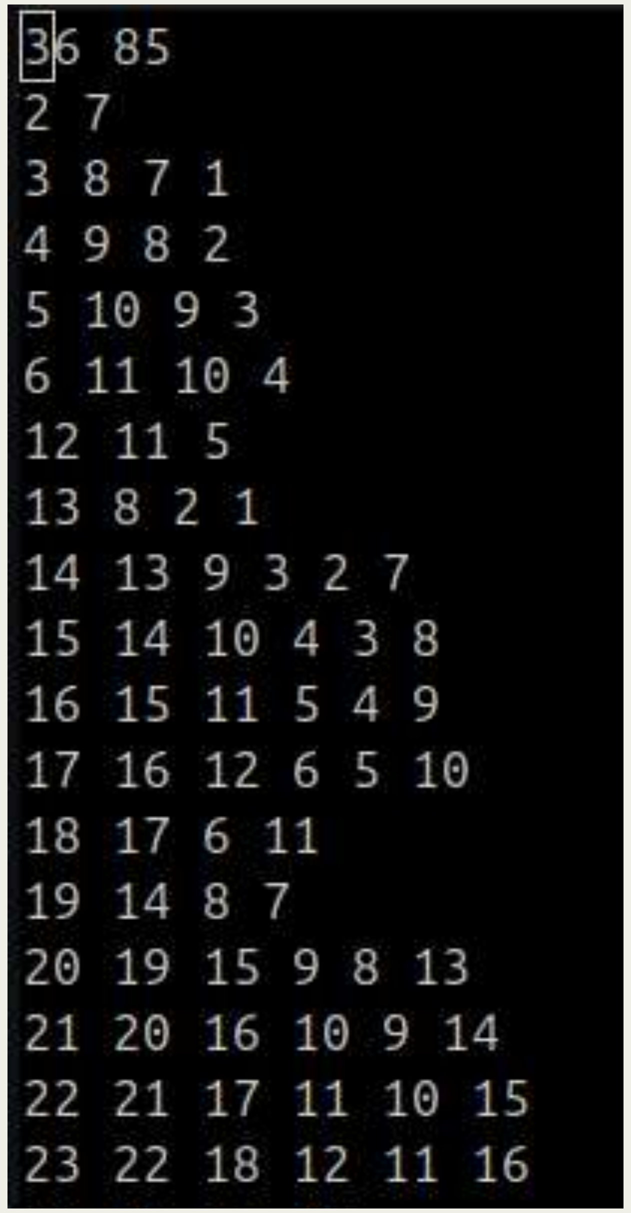
```
POLYGONS 50 200
3 0 6 1
3 6 7 1
3 1 7 2
3 7 8 2
3 2 8 3
3 8 9 3
...
...
...
...
...
```

```
CELLS 169 456
OFFSETS vtktypeint64
0 1 2 3 4 5 6 7 8
10 12 14 16 18 20 22 24 26
28 30 32 34 36 38 40 42 44
46 48 50 52 54 56 58 60 62
64 66 68 70 72 75 78 81 84
87 90 93 96 99 102 105 108 111
...
...
...
CONNECTIVITY vtktypeint64
0 1 2 3 4 5 6 7 0
8 8 9 9 10 10 3 3 11
11 12 12 13 13 2 2 14 14
15 15 16 16 1 1 17 17 18
...
...
...
```

```
CELL_TYPES 168
1
1
1
1
1
...
...
...
CELL_DATA 168
FIELD FieldData 1
PartitionID 1 168 int
61 53 12 108 81 32 41 102 60
68 67 109 111 113 0 14 13 47
46 52 55 54 35 32 31 30 44
...
...
...
```

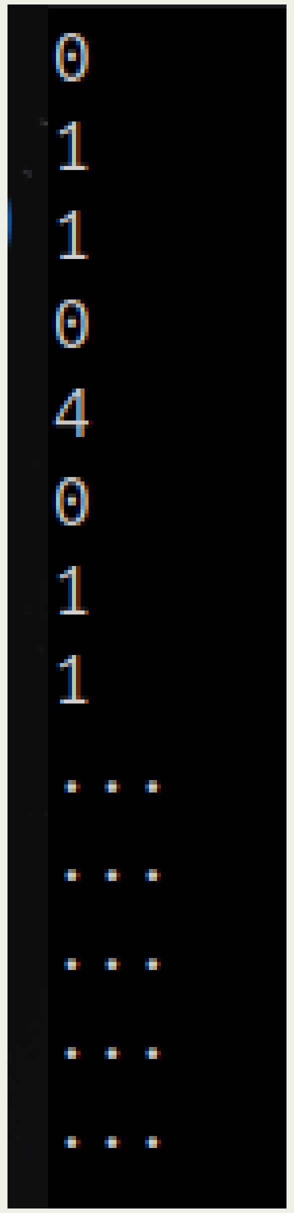
[VTK File Structure]

CONVERSION OF .VTK TO .GRAPH



```
36 85
2 7
3 8 7 1
4 9 8 2
5 10 9 3
6 11 10 4
12 11 5
13 8 2 1
14 13 9 3 2 7
15 14 10 4 3 8
16 15 11 5 4 9
17 16 12 6 5 10
18 17 6 11
19 14 8 7
20 19 15 9 8 13
21 20 16 10 9 14
22 21 17 11 10 15
23 22 18 12 11 16
```

Fig. 6. Graph File



```
0
1
1
0
4
0
1
1
...
...
...
...
...
```

Fig. 7. Part File

.graph File Structure

VISUALIZATION ENGINE AND OUTPUT

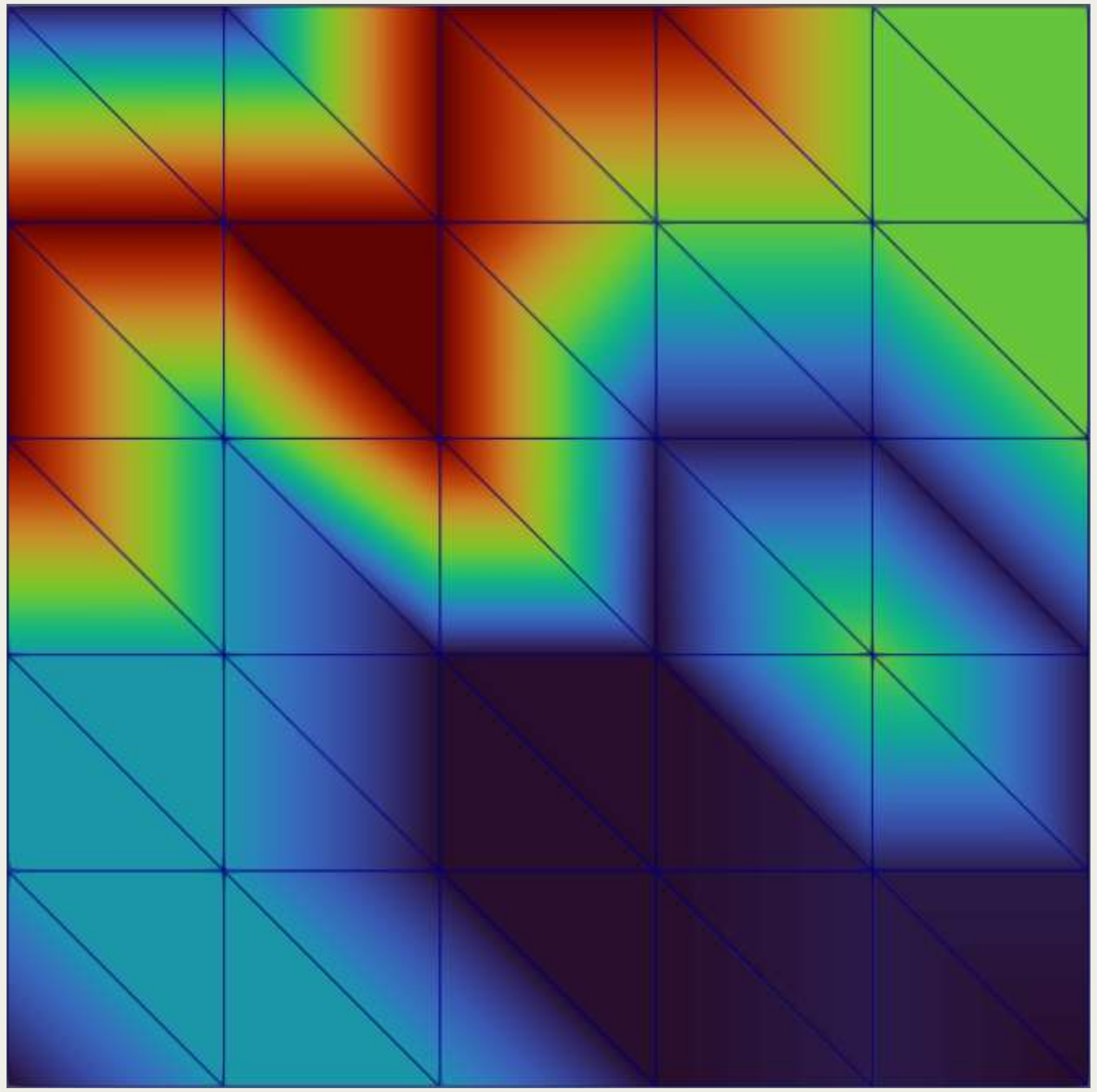


Fig. 8. 5 Partition Mesh Visualization

CONTENT

GEOMETRIC PARTITIONING THEORY

Theorem: Let G = (V,E) be a (α, k)-overlap graph in d dimensions with n nodes. Then there is a vertex separator V_s such that $V = V_1 \cup V_s \cup V_2$ with the following properties:

- $|V_1| \leq \dfrac{n(d+1)}{d+2} \quad \text{and} \quad |V_2| \leq \dfrac{n(d+1)}{d+2}$
- $|V_S| \leq O\left(\alpha k^{1/d} n^{1-1/d}\right)$

GEOMETRIC PARTITIONING THEORY

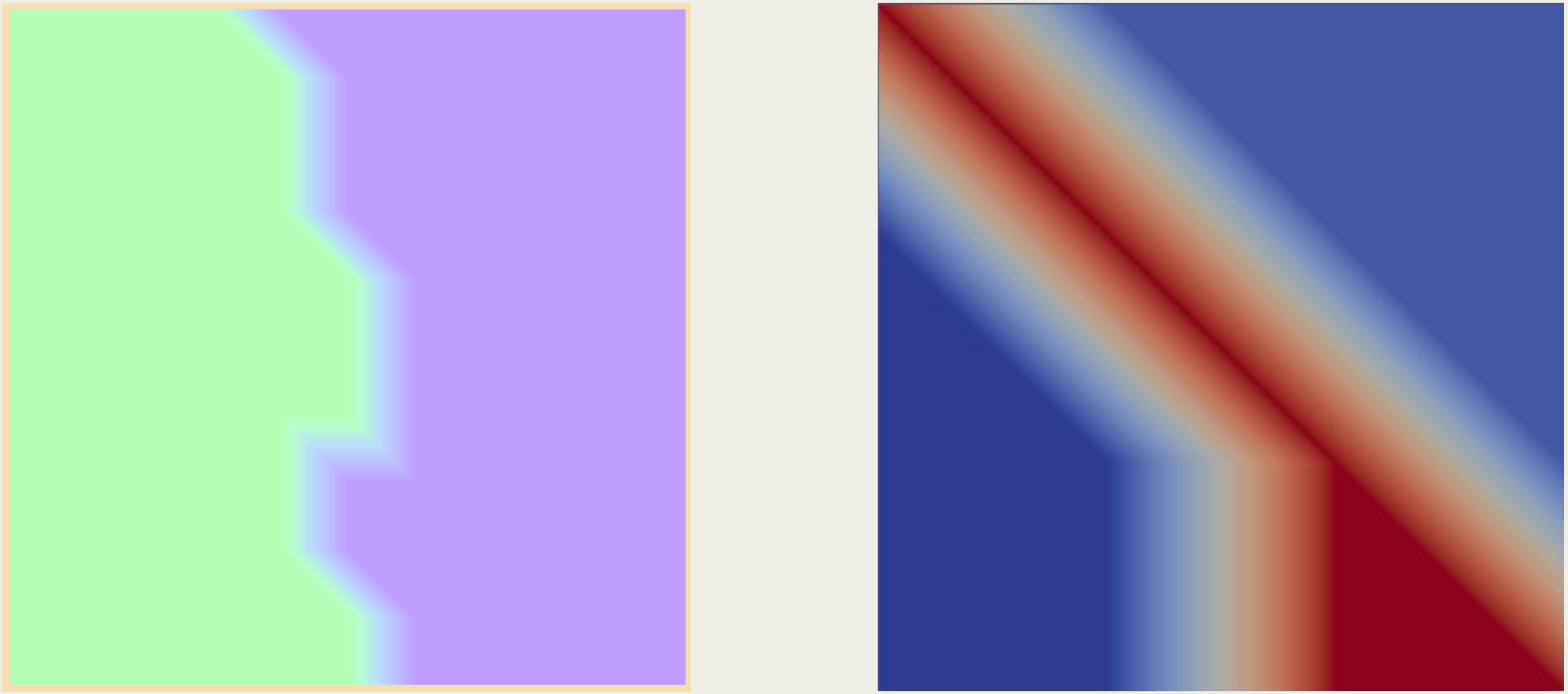


Fig. 9. Partition without and with geometric separator

GEOMETRIC PARTITIONING THEORY

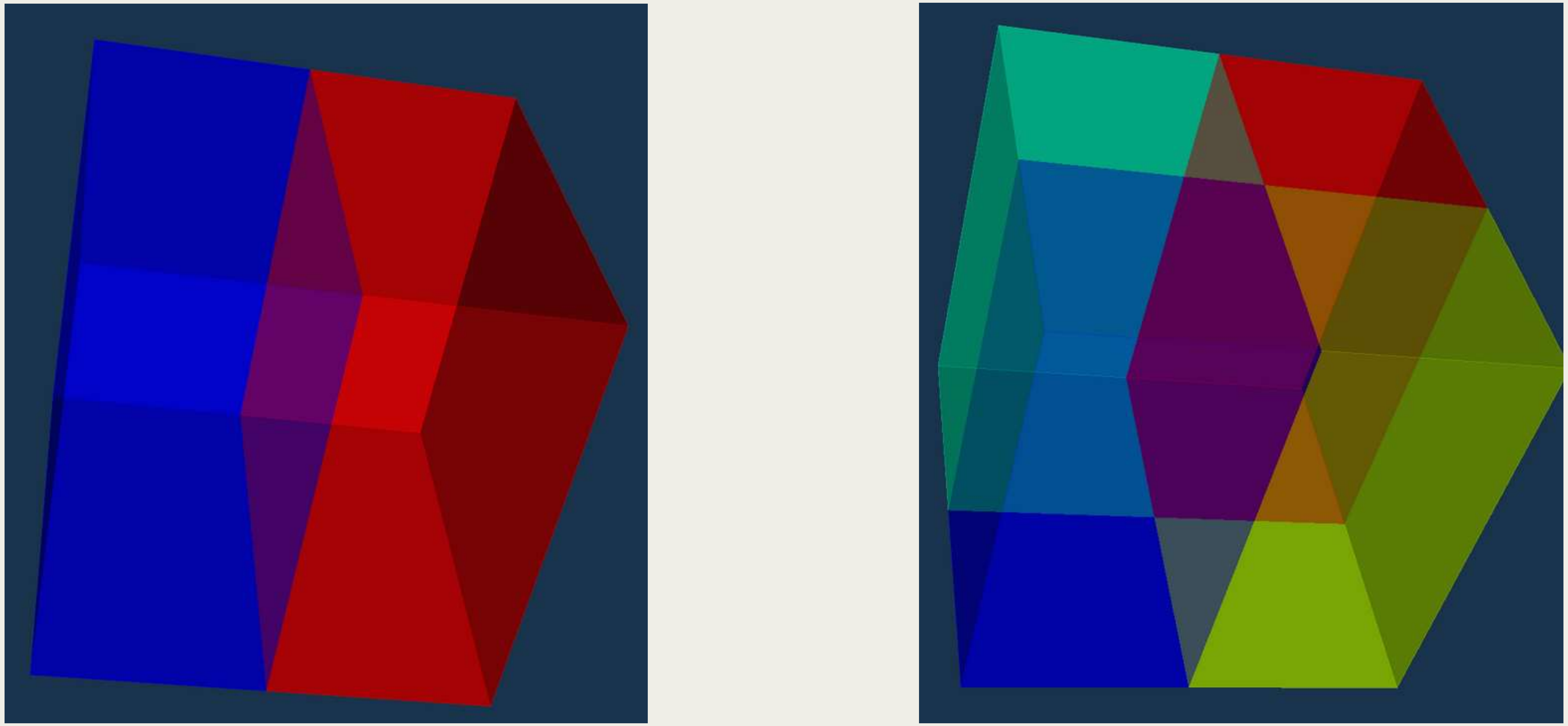


Fig. 10. Geometric separator for 3D Partitioning

THE END?

STRATEGIC PIVOT

Why opt METIS, if the geometric separator of the mesh is already known?

Objective Realignment - Minimizing communication with the geometrically separated sub-domain.

CONTENT

Minimizing the Edgecut

Consider a graph G = (V, E), and let P be a vector of size |V | such that P [i] stores the number of the partition that vertex i belongs to. The edgecut of this partitioning is defined as the number of edges (v, u) for which P [v] ≠ P [u]. If the graph has weights associated with the edges, then the edgecut is defined as the sum of the weight of these straddling edges.

$$\text{Edgecut} = \sum_{\substack{(u,v)\in E \\ P[v]\neq P[u]}} w(u, v).$$

Minimizing the total communication volume

Let $G = (V, E)$ be a graph, and let P be a vector of size $|V|$ where $P[i]$ denotes the partition number to which vertex i belongs. Define $V_b \subset V$ as the set of interface (or border) vertices, where each vertex $v \in V_b$ is connected to at least one vertex in a different partition.

For each vertex $v \in V_b$, let $N_{adj}[v]$ represent the number of distinct partitions (other than $P[v]$) to which the vertices adjacent to v belong. The total communication volume of this partitioning is defined as:

$$\text{total\_v} = \sum_{v \in V_b} N_{\text{adj}}[v].$$

METIS PARTITIONING ALGORITHMS

Multilevel k-way Partitioning:

A k-way partition P_m of the coarsest graph $G_m = (V_m, E_m)$ is computed, which divides V_m into k parts.

Coarsening

Partitioning

Uncoarsening

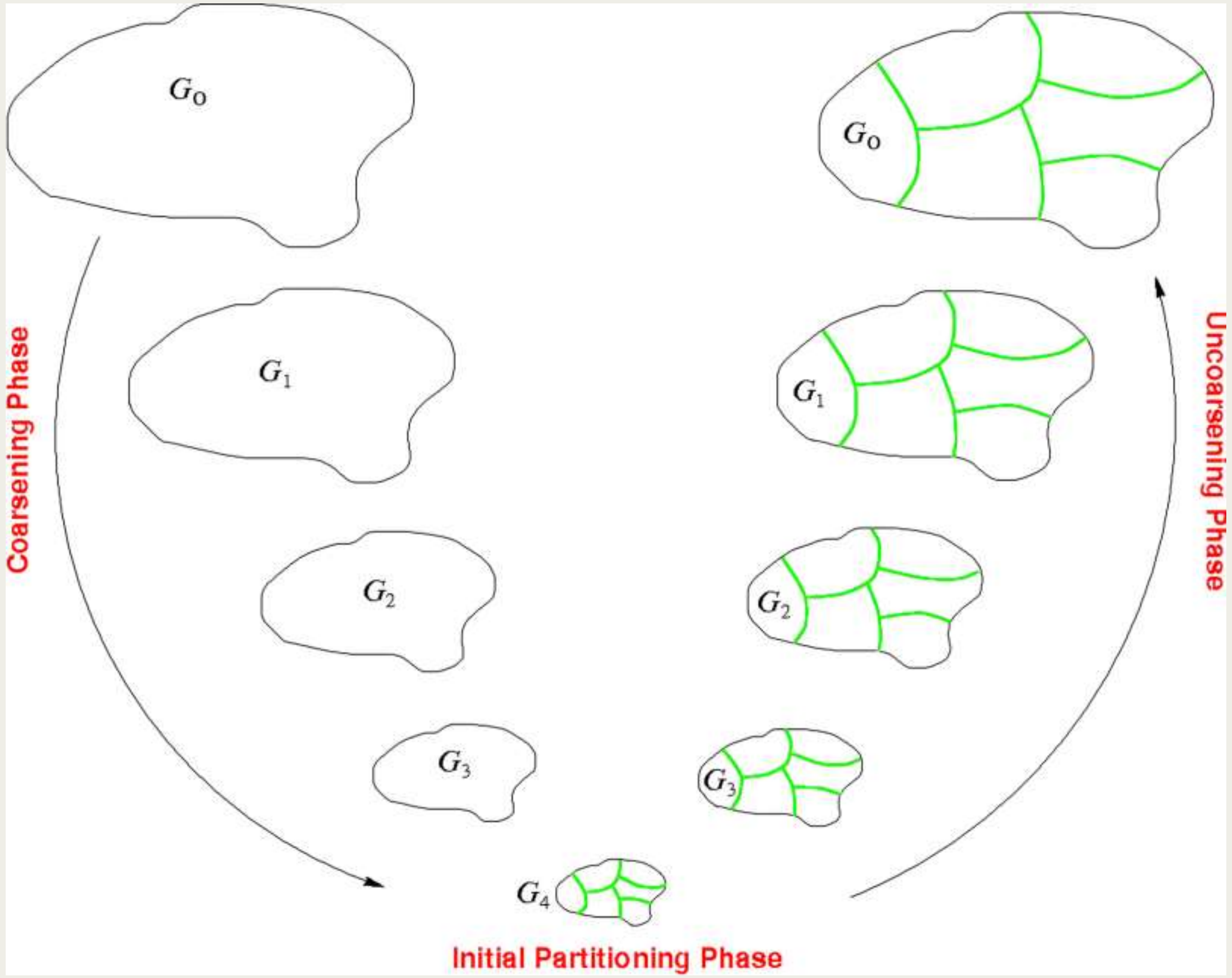


Fig. 11. Multilevel K-way Partitioning

Multilevel R-Bisection:

A 2-way partition P_m of the graph $G_m = (V_m, E_m)$ is computed that partitions V_m into two parts, each containing half the vertices of G_0.

Coarsening

Partitioning

Uncoarsening

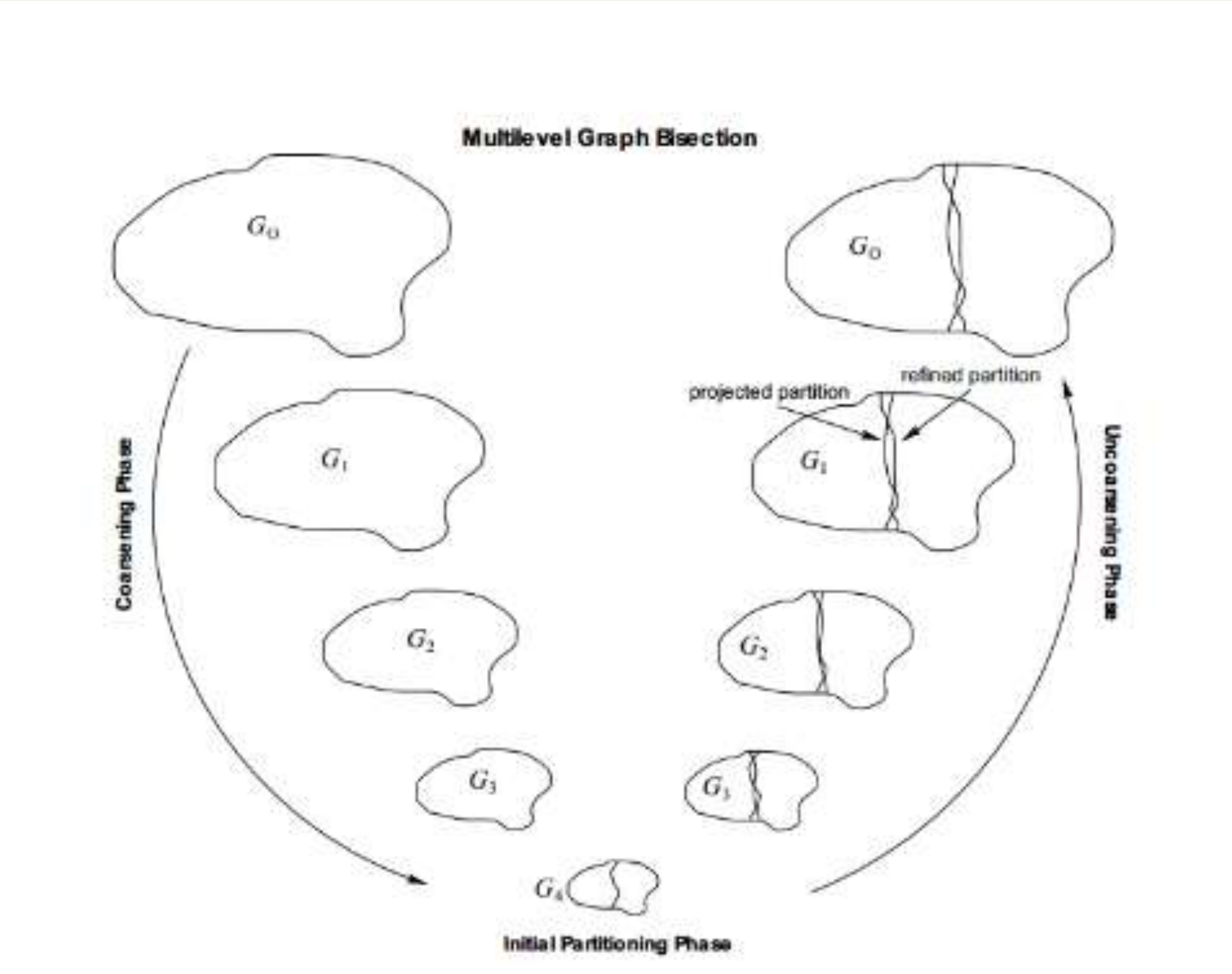


Fig. 11. Multilevel R-Bisection Partitioning

METIS PARTITIONING ROUTINES

`METIS_PartMeshDual`

This function is used to partition a mesh into k parts based on a partitioning of the mesh’s dual graph.

`METIS_PartMeshNodal`

This function is used to partition a mesh into k parts based on a partitioning of the mesh’s nodal graph.

MESHES

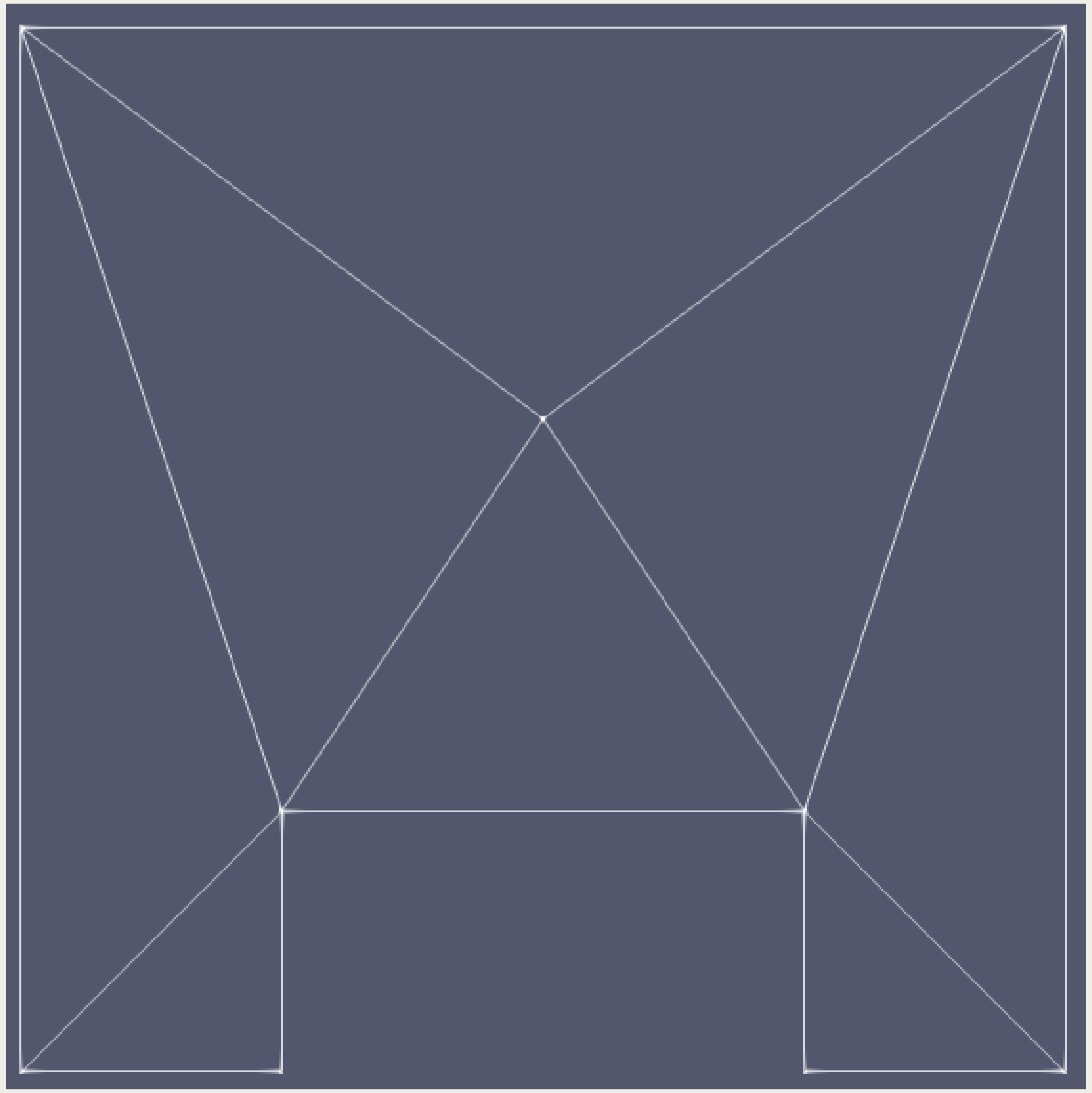


Fig. 13. L1 Geometrically Separated 2D Mesh

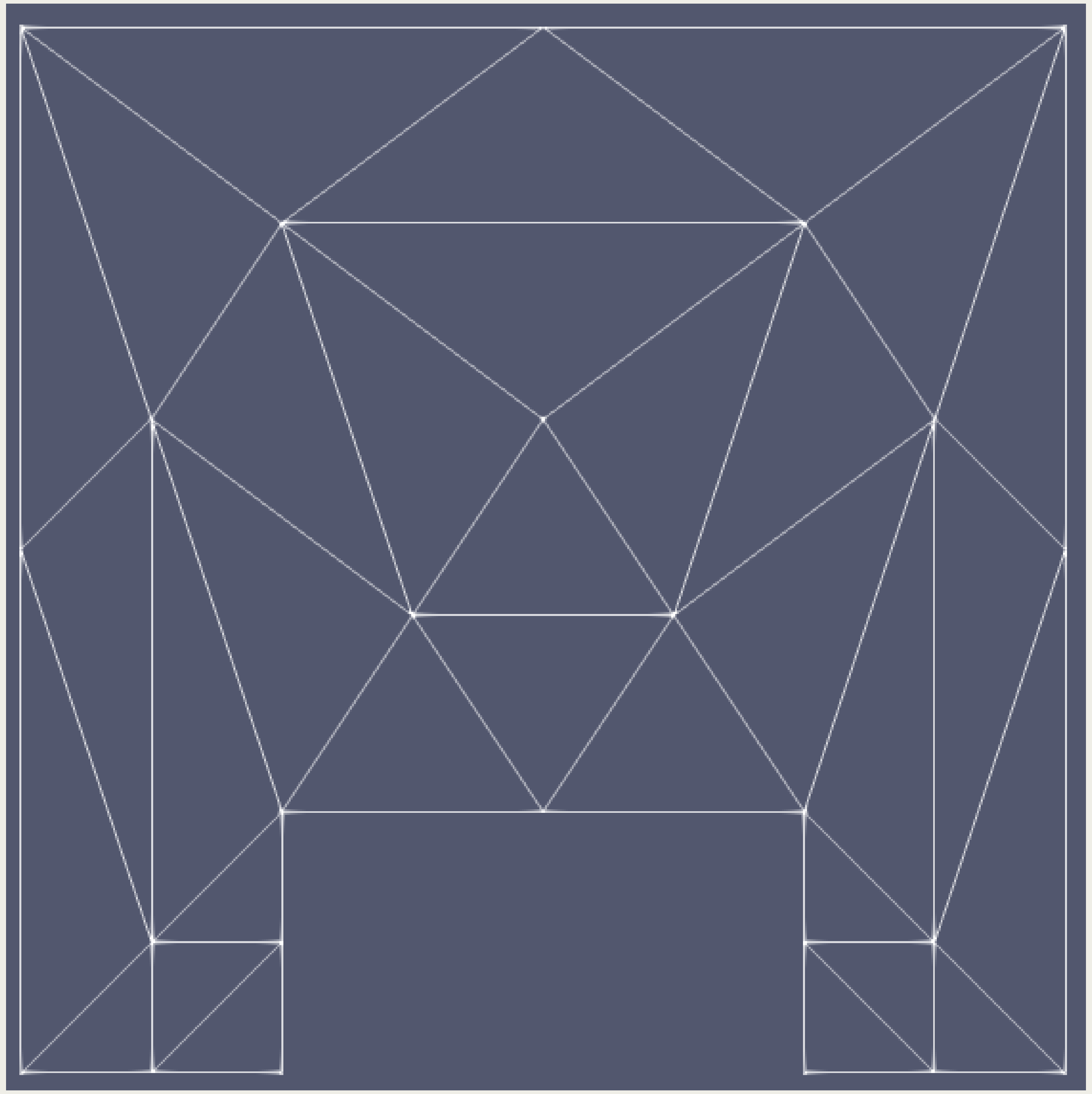


Fig. 14. L2 Geometrically Separated 2D Mesh

MESHES

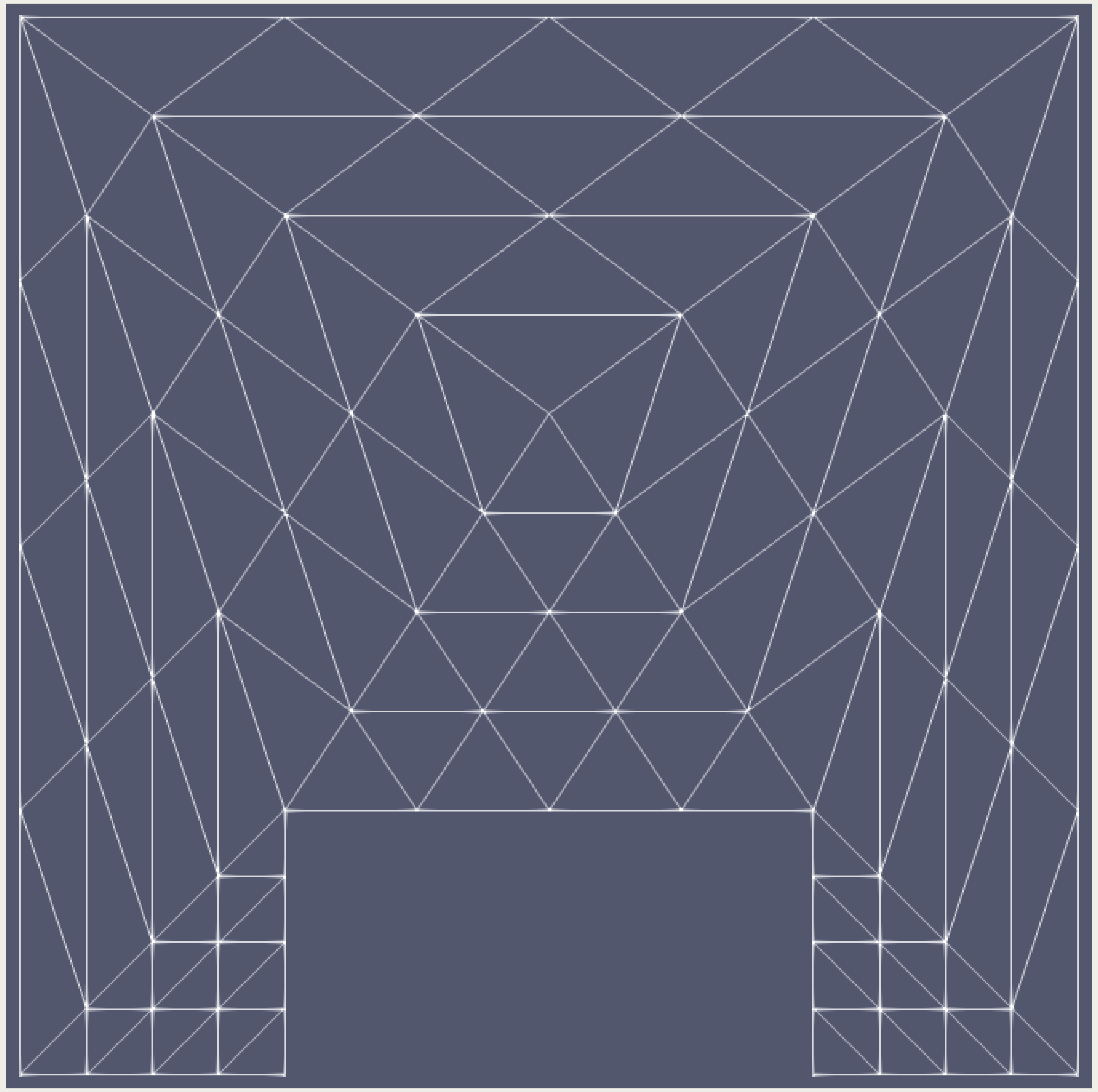


Fig. 15. L3 Geometrically Separated 2D Mesh

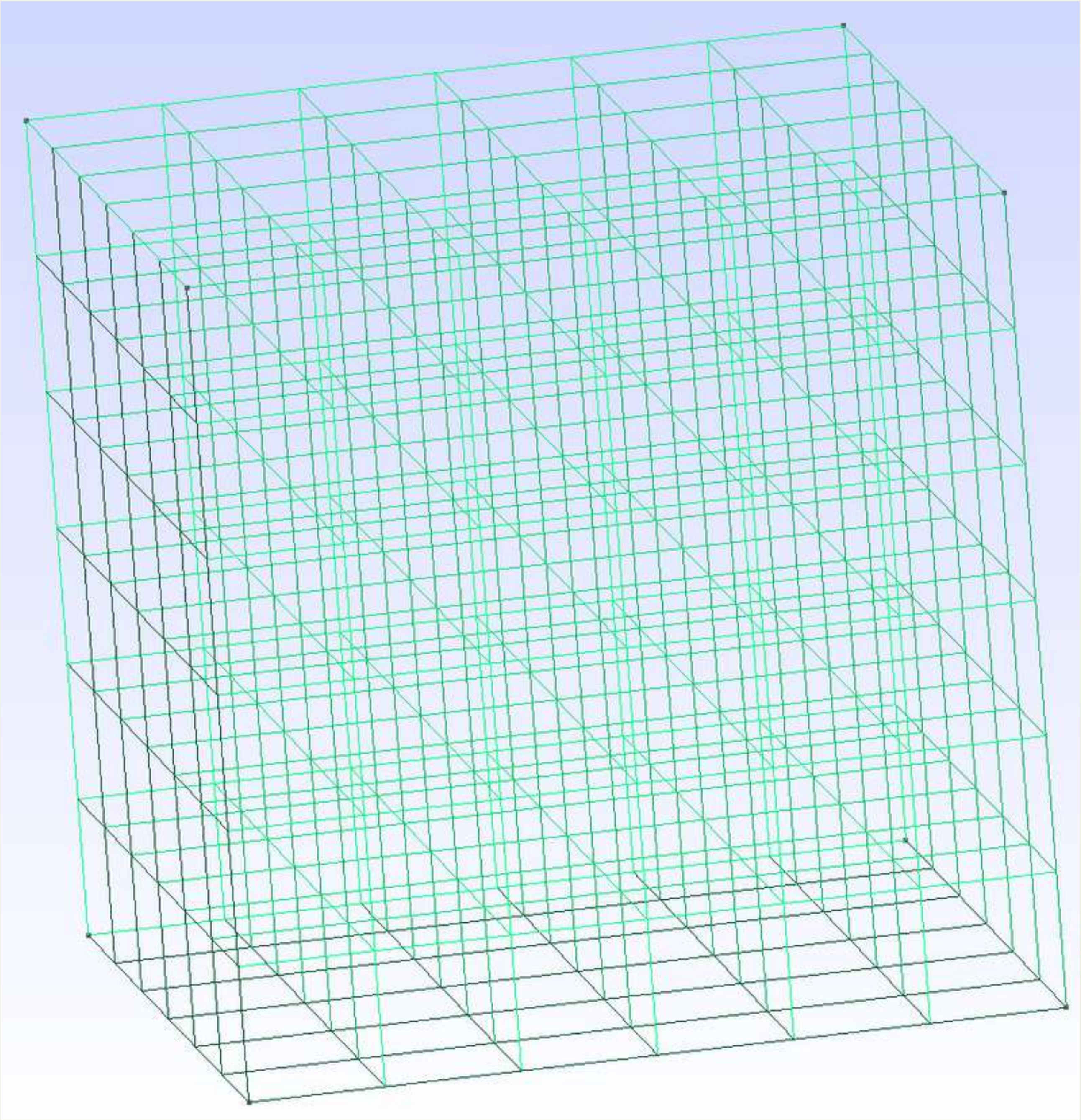


Fig. 16. Transfinite 3D Cube Mesh

PARTITIONING

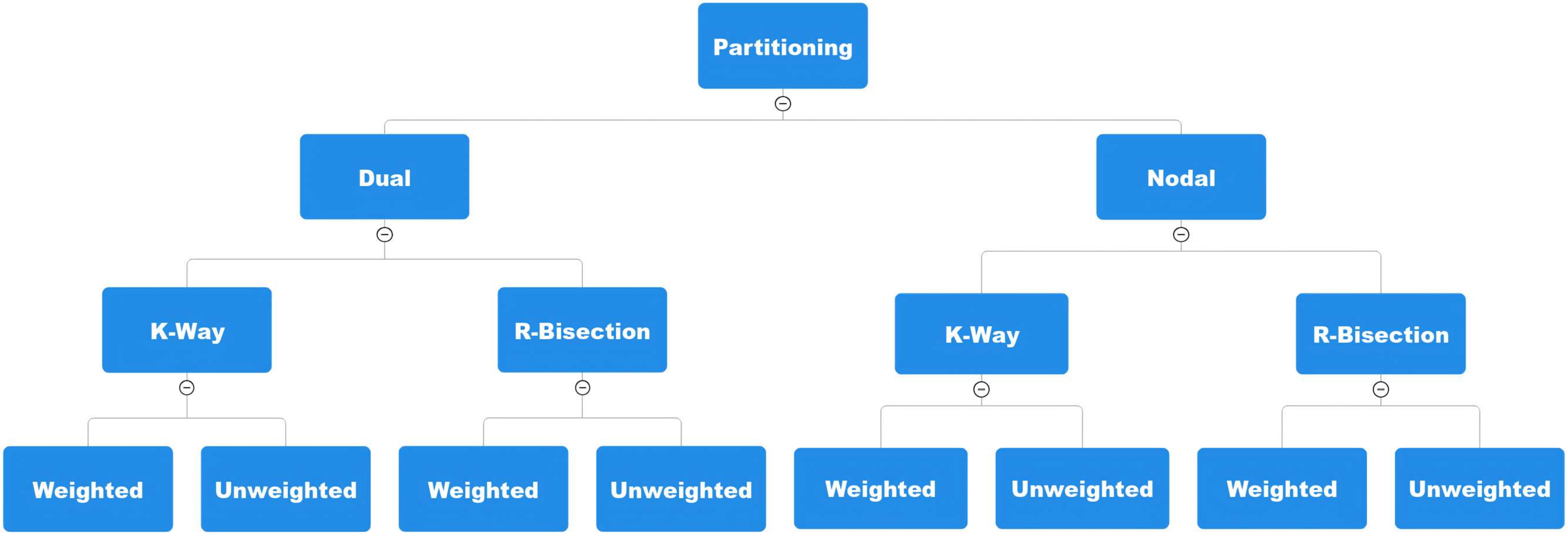


Fig. 17. Algorithmic Combinations for Partitioning in Single Partitioning Scheme
300+ files were generated per mesh

PARTITIONING

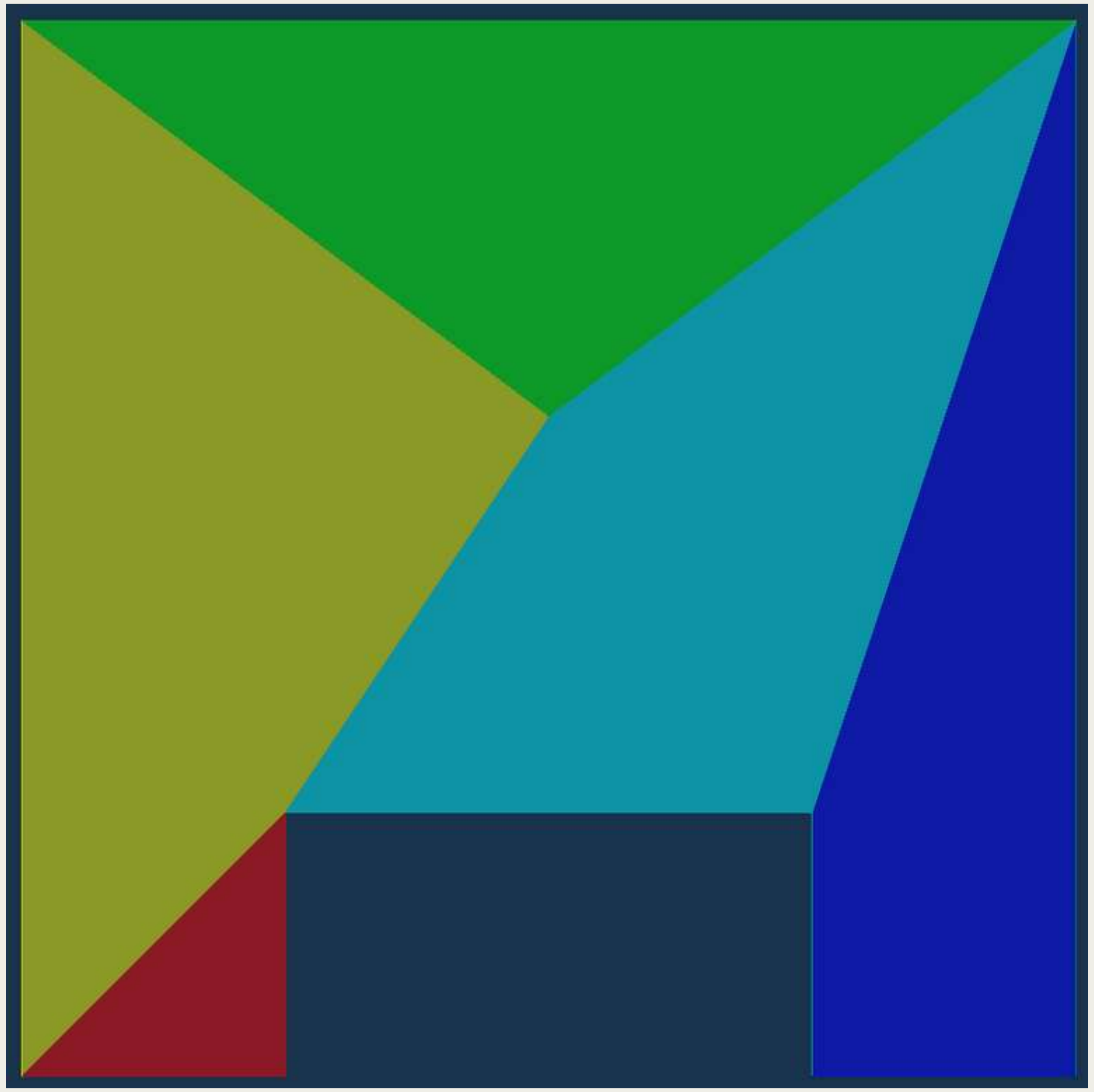


Fig 18. L1, Dual_Kway, 5 Partitions

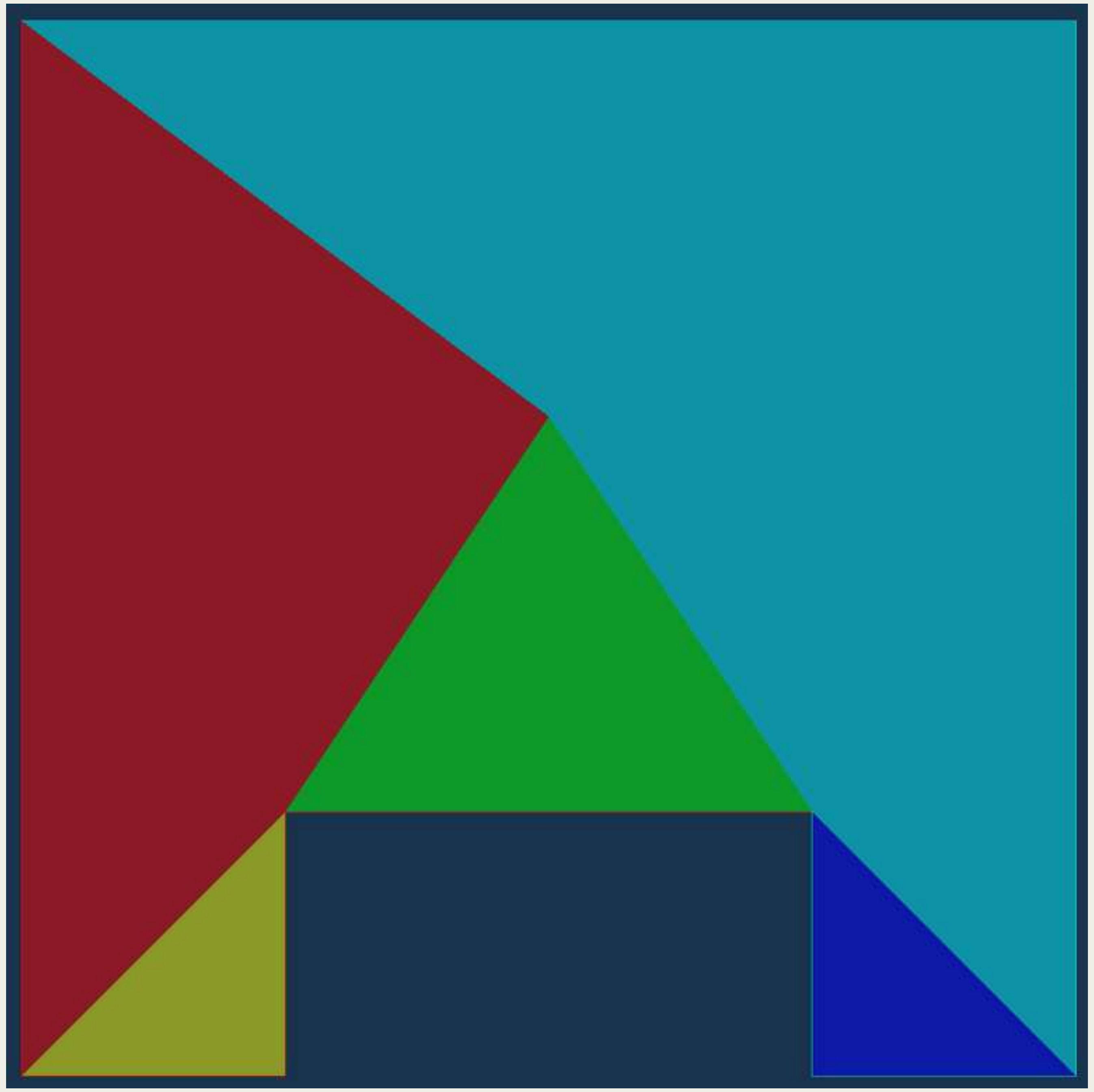


Fig. 19. L1, Dual_RBi, 5 Partitions

PARTITIONING

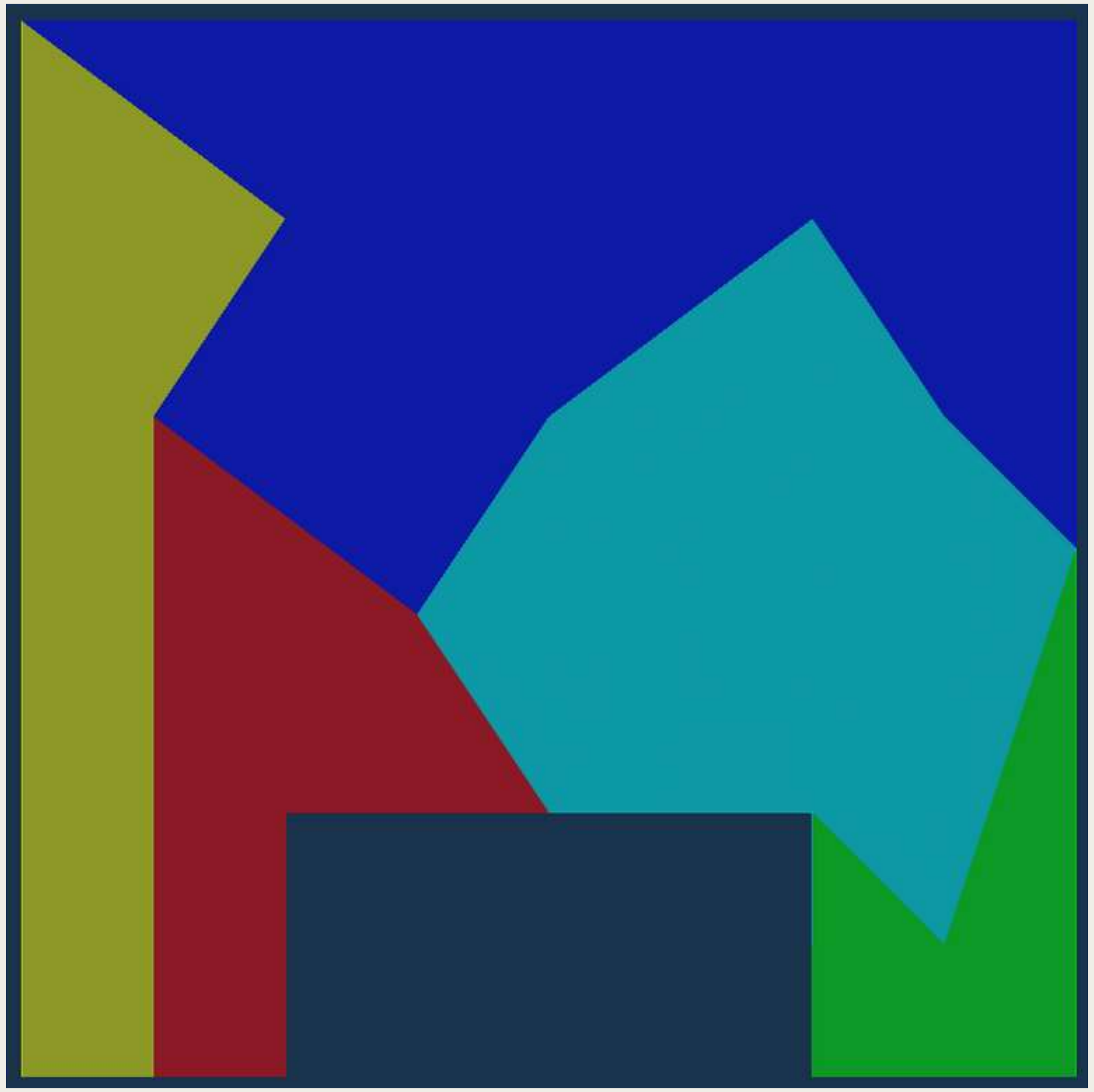


Fig. 20. L2, Dual_Kway, 5 Partitions, Unweighted

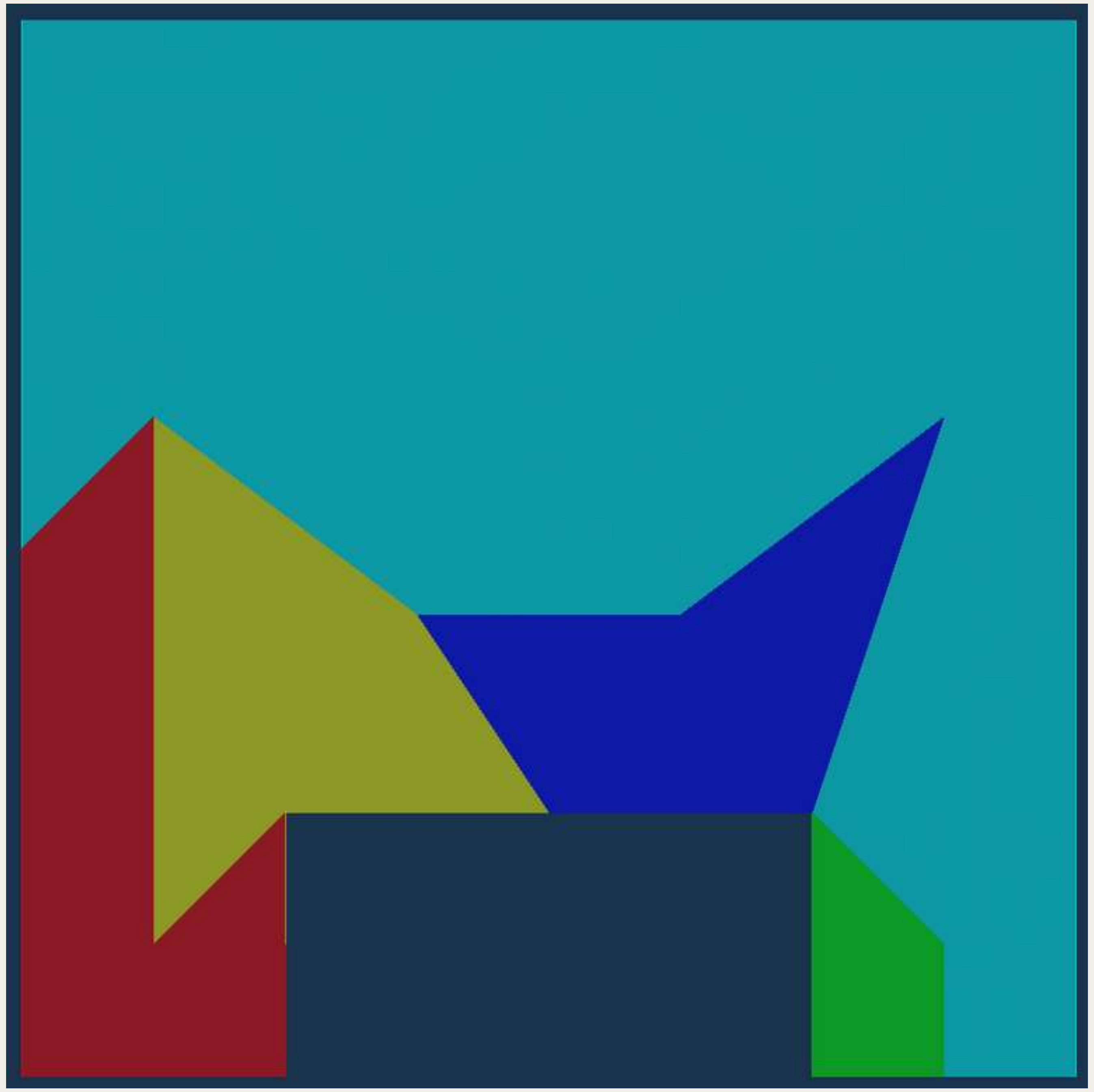


Fig. 21. L2, Dual_Kway, 5 Partitions, Weighted

PARTITIONING

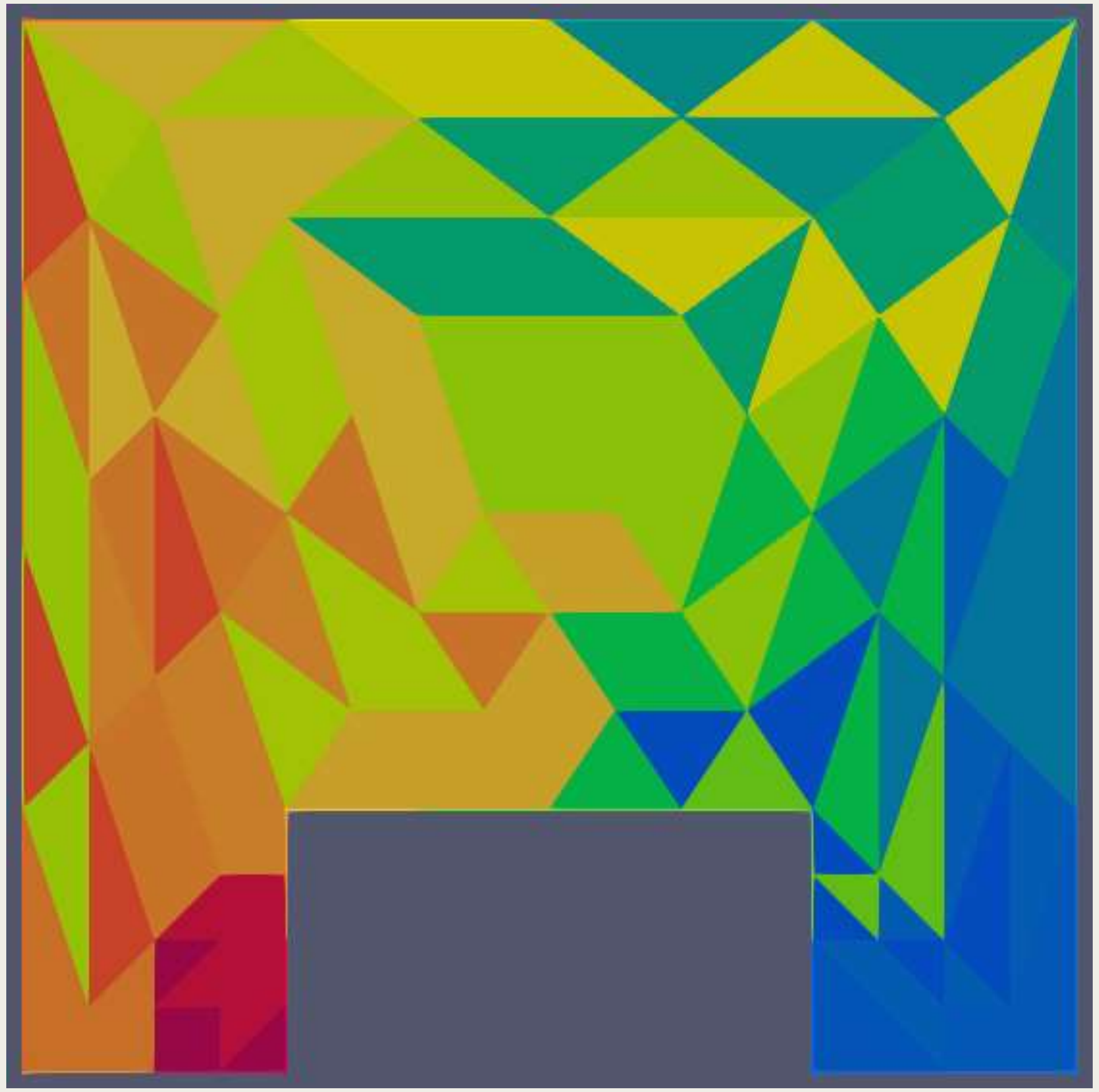


Fig. 22. L3, Dual_Kway, 90 Partitions, Unweighted

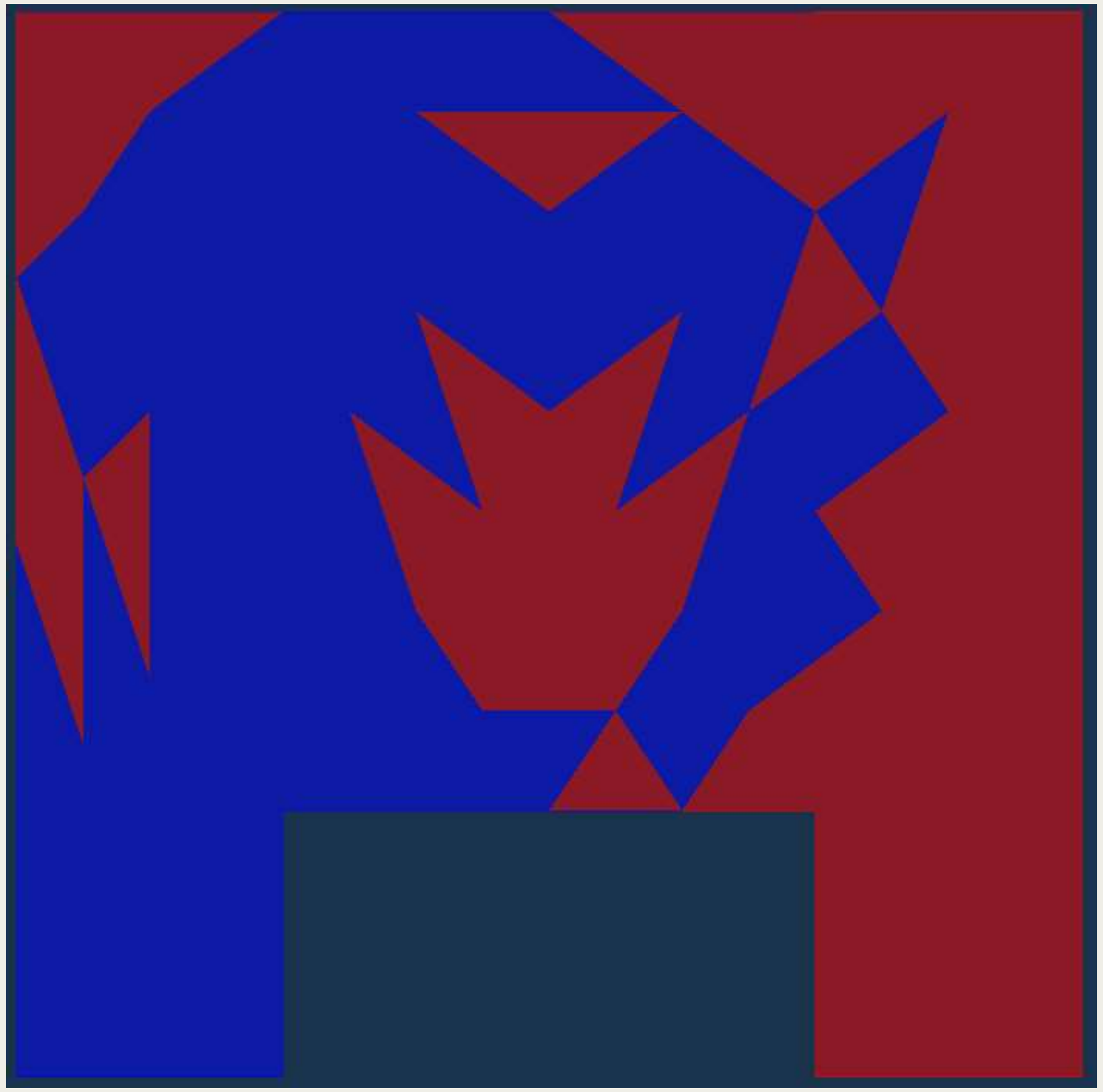


Fig. 23. L3, Nodal_Kway, 100 Partitions, Unweighted

PARTITIONING

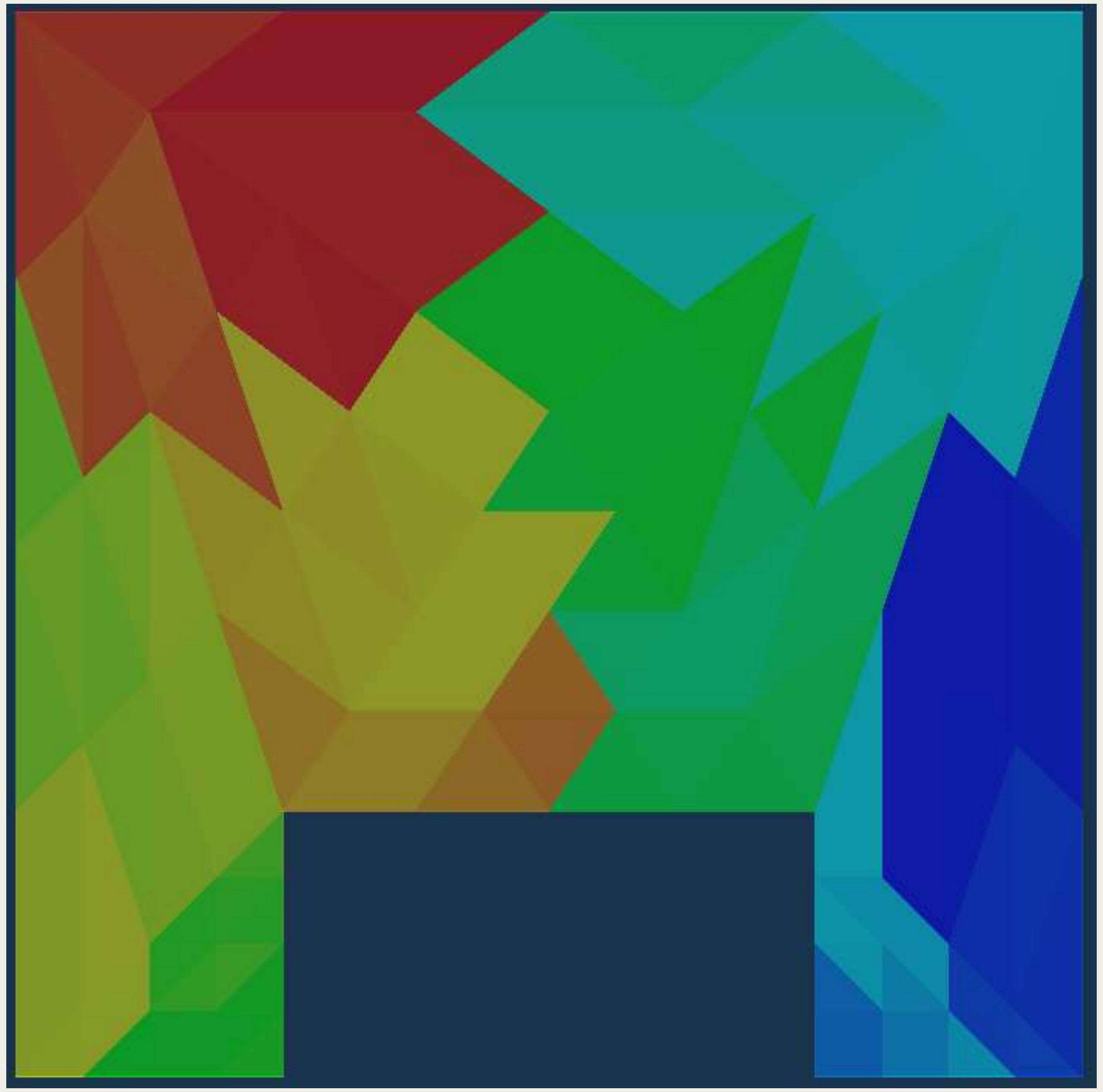


Fig. 24. L3, Dual_RBi, 120 Partitions, Unweighted

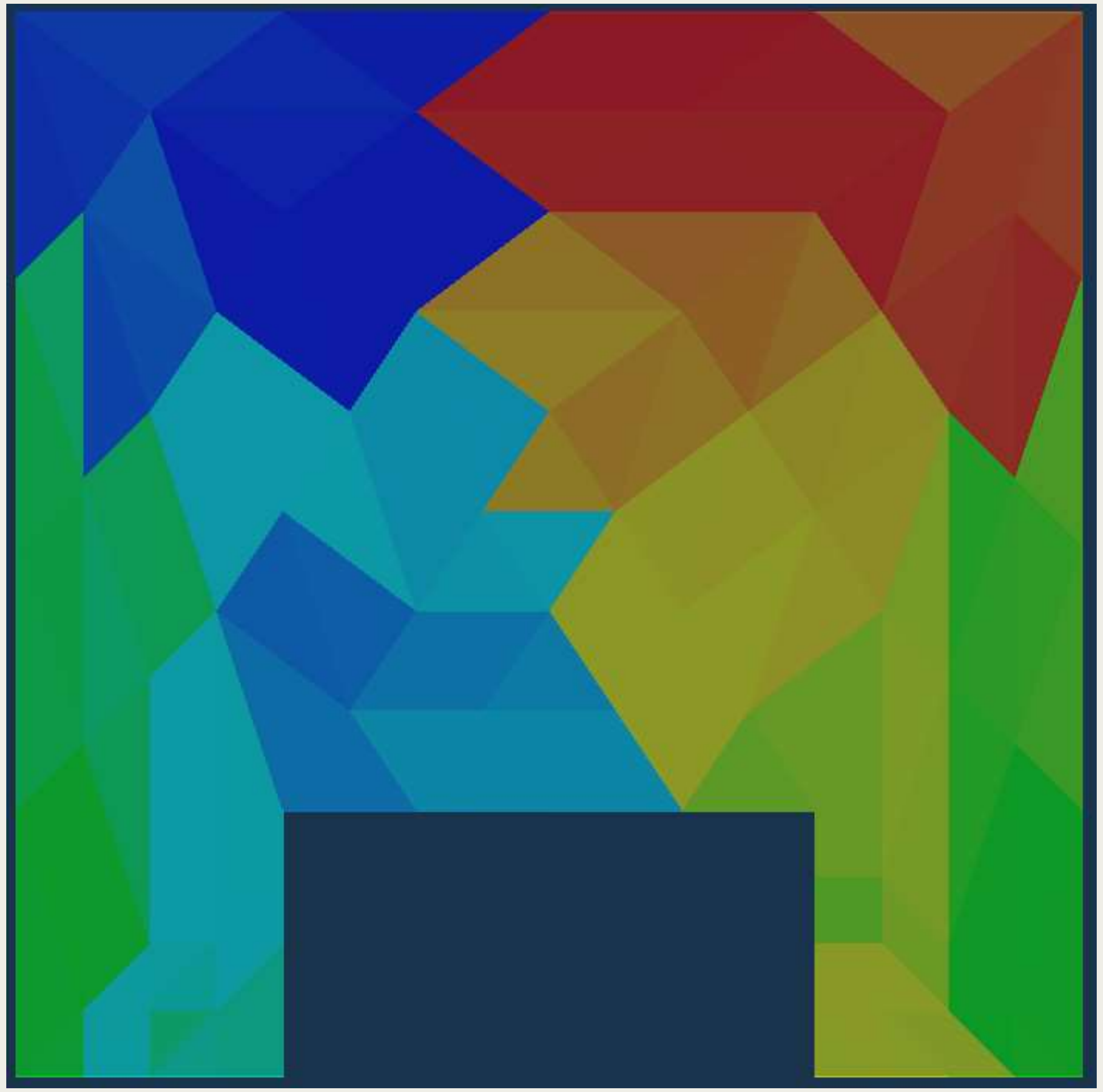


Fig. 25. L3, Dual_RBi, 120 Partitions, weighted

CONTENT

COMMUNICATION METRIC

Calculates the following communication quantities within the mesh:

1. Communication Metric for The ‘Stitched’ Mesh
2. Total Communication Load
3. Average Communication per Partition (normalized)
4. Maximum Communication Load

STITCHING

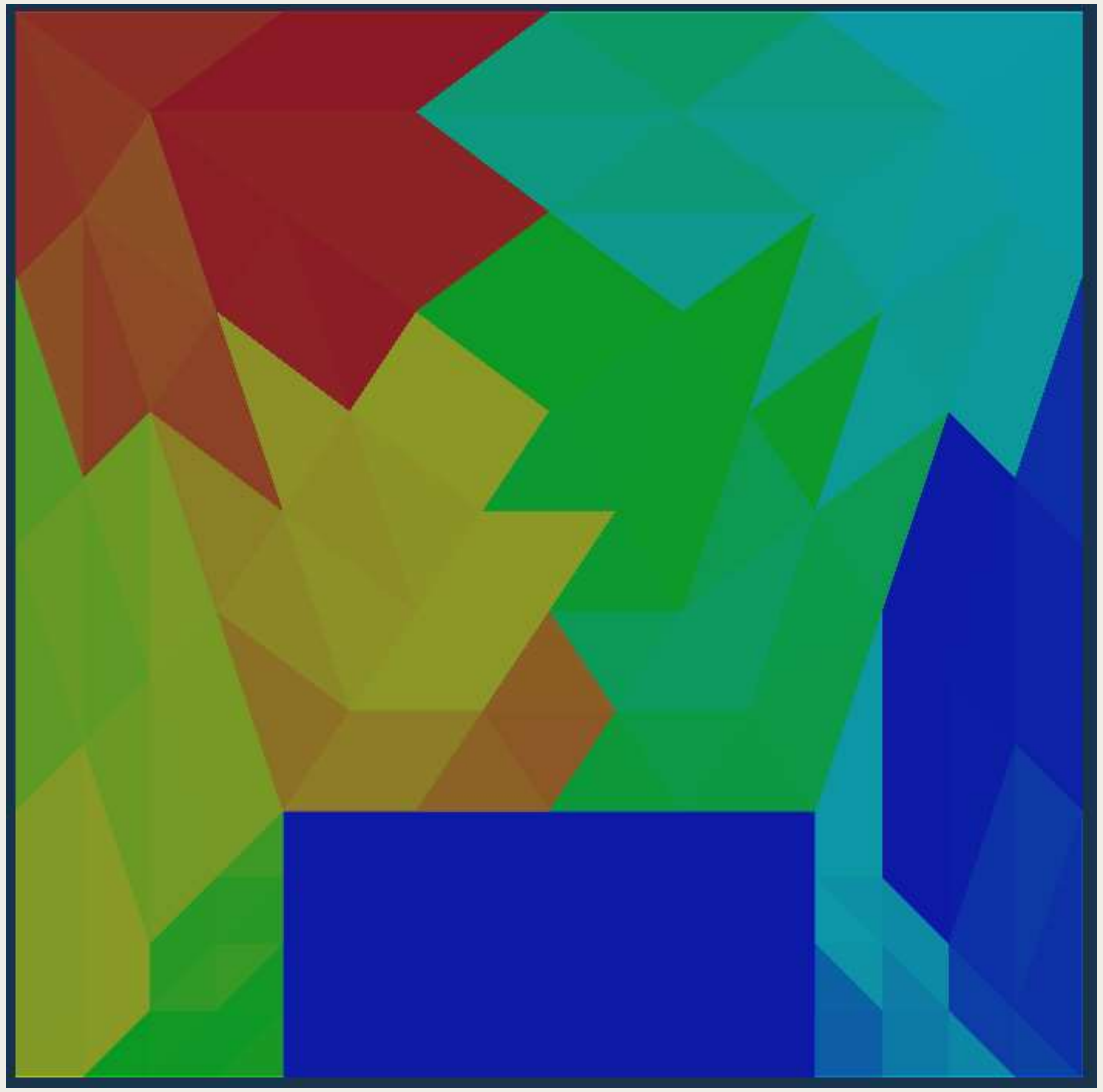


Fig. 26. L3, Dual_RBi, 120 Partitions, Unweighted, Stitched

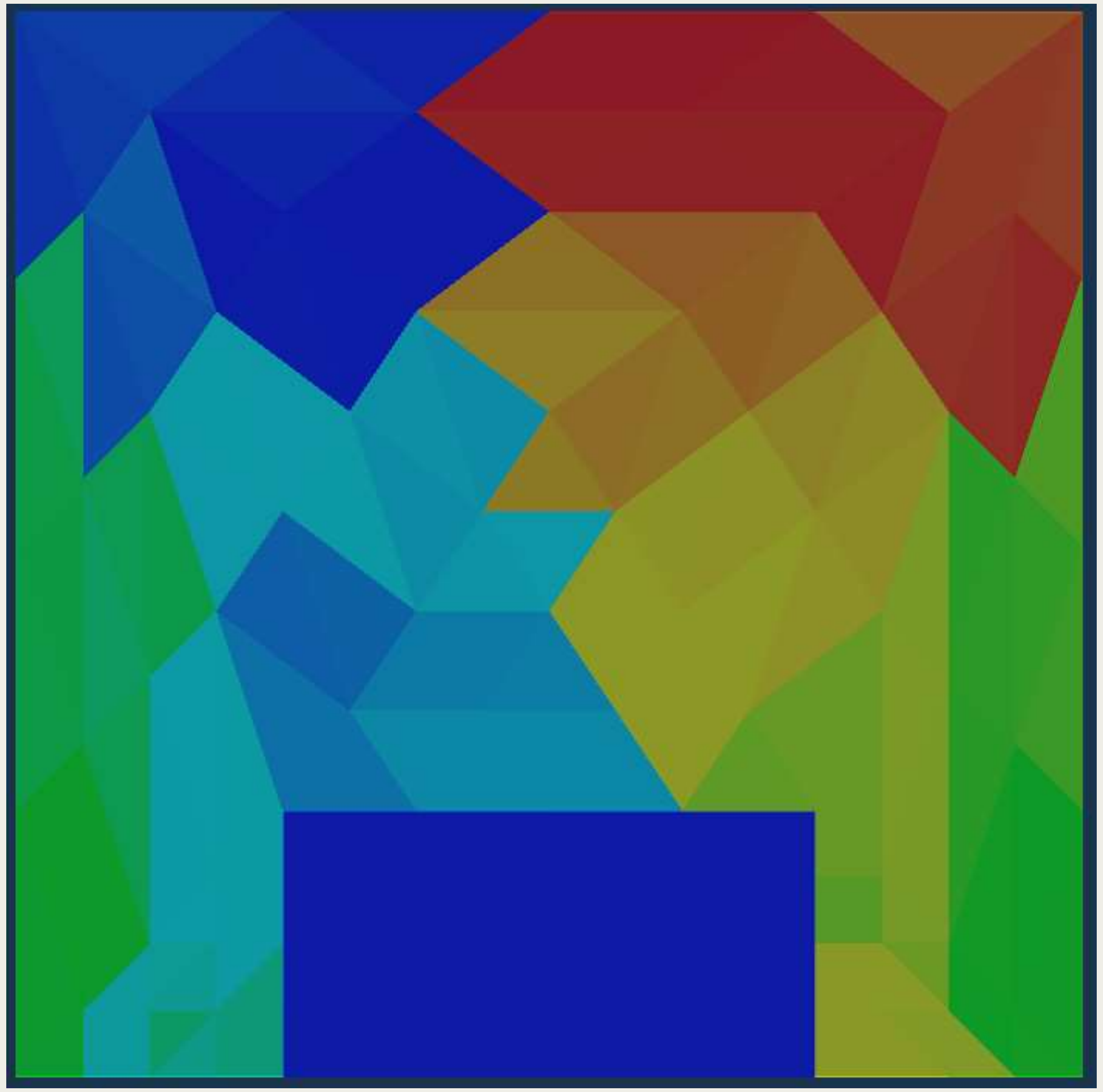
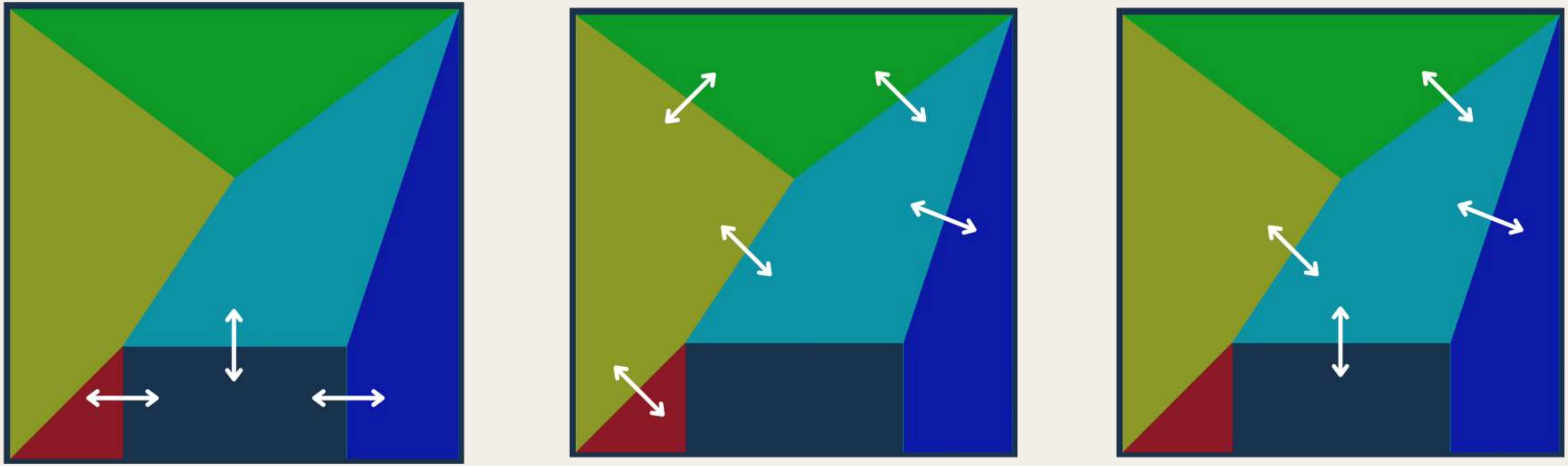


Fig. 27. L3, Dual_RBi, 120 Partitions, weighted, Stitched



METRIC

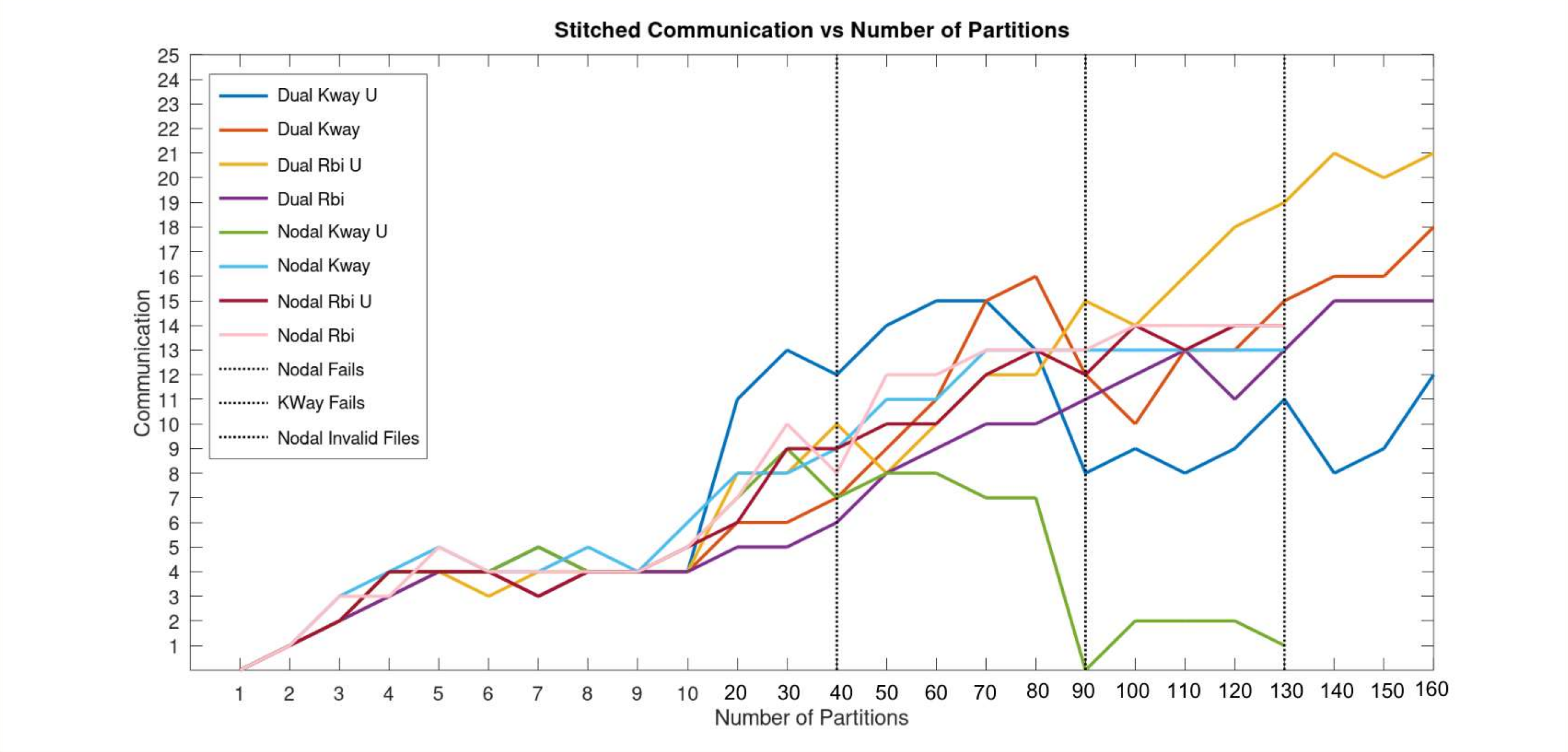
CONTENT

- Problem Statement & Objectives
- Graph Theory & Tools
- Geometric Partitioning & Objective Realignment
- Partitioning Theory & Outcomes
- Communication Metric Development & Implementation

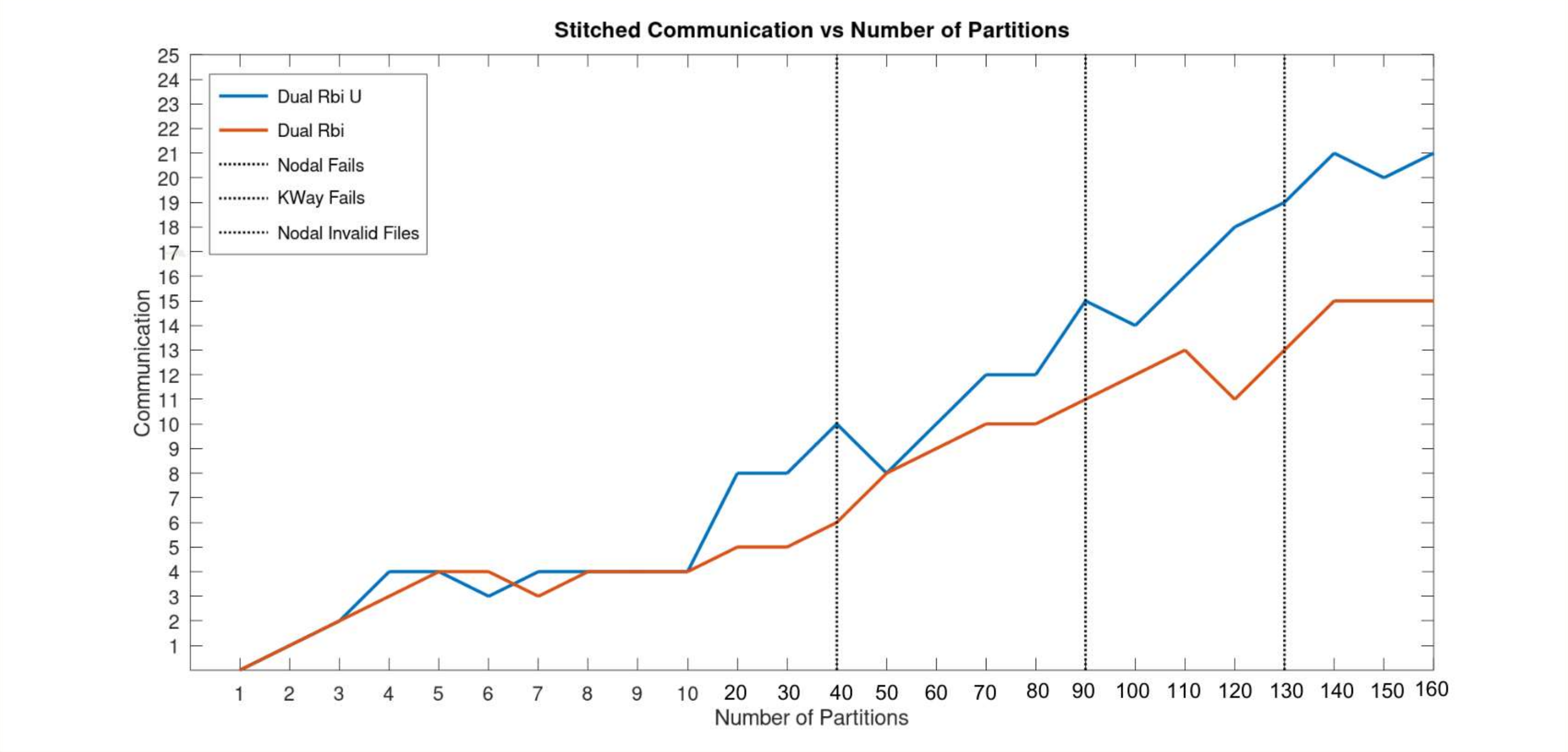
> Results

- Troubleshooting
- Conclusion & Scope

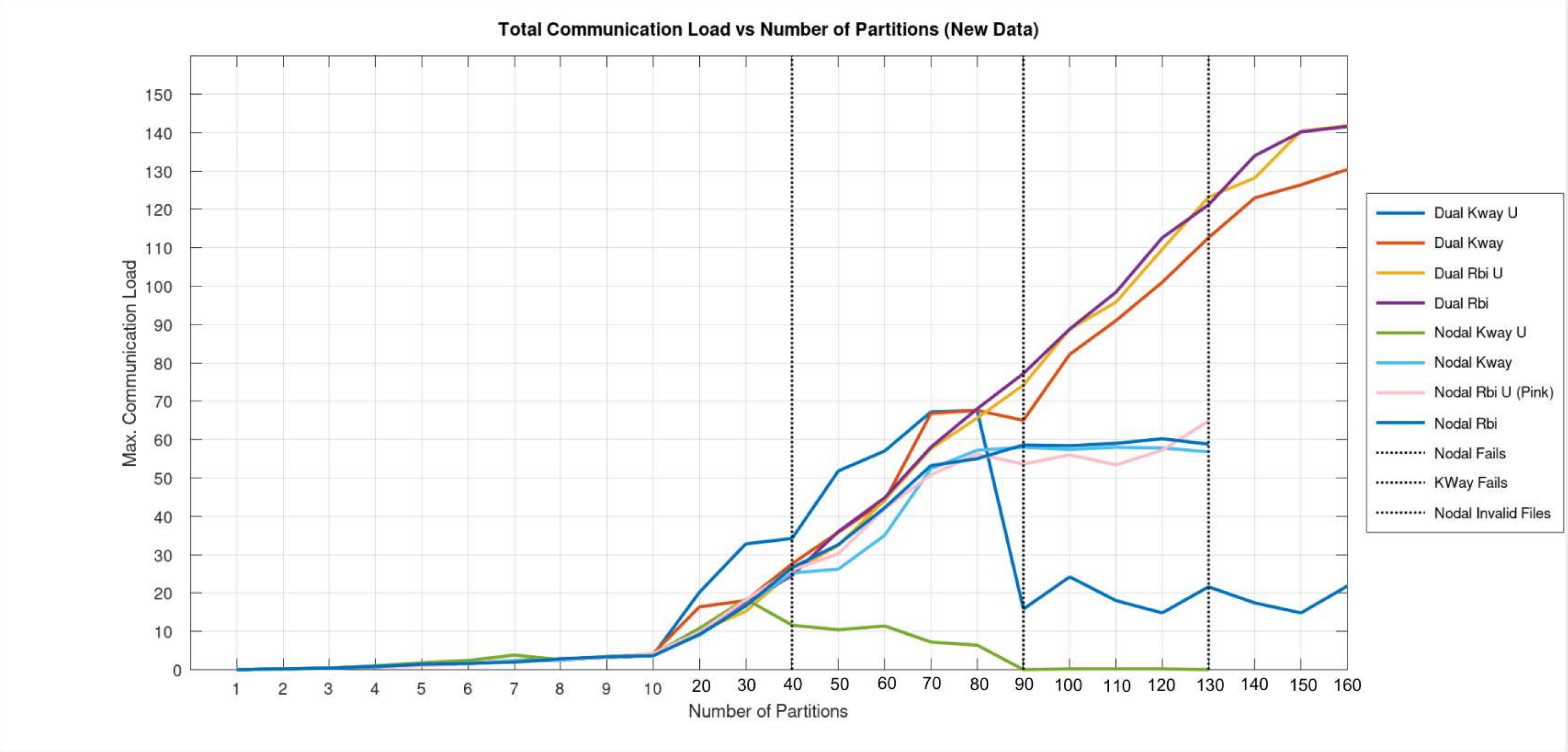
COMMUNICATION LOAD GRAPH FOR STITCHED MESH



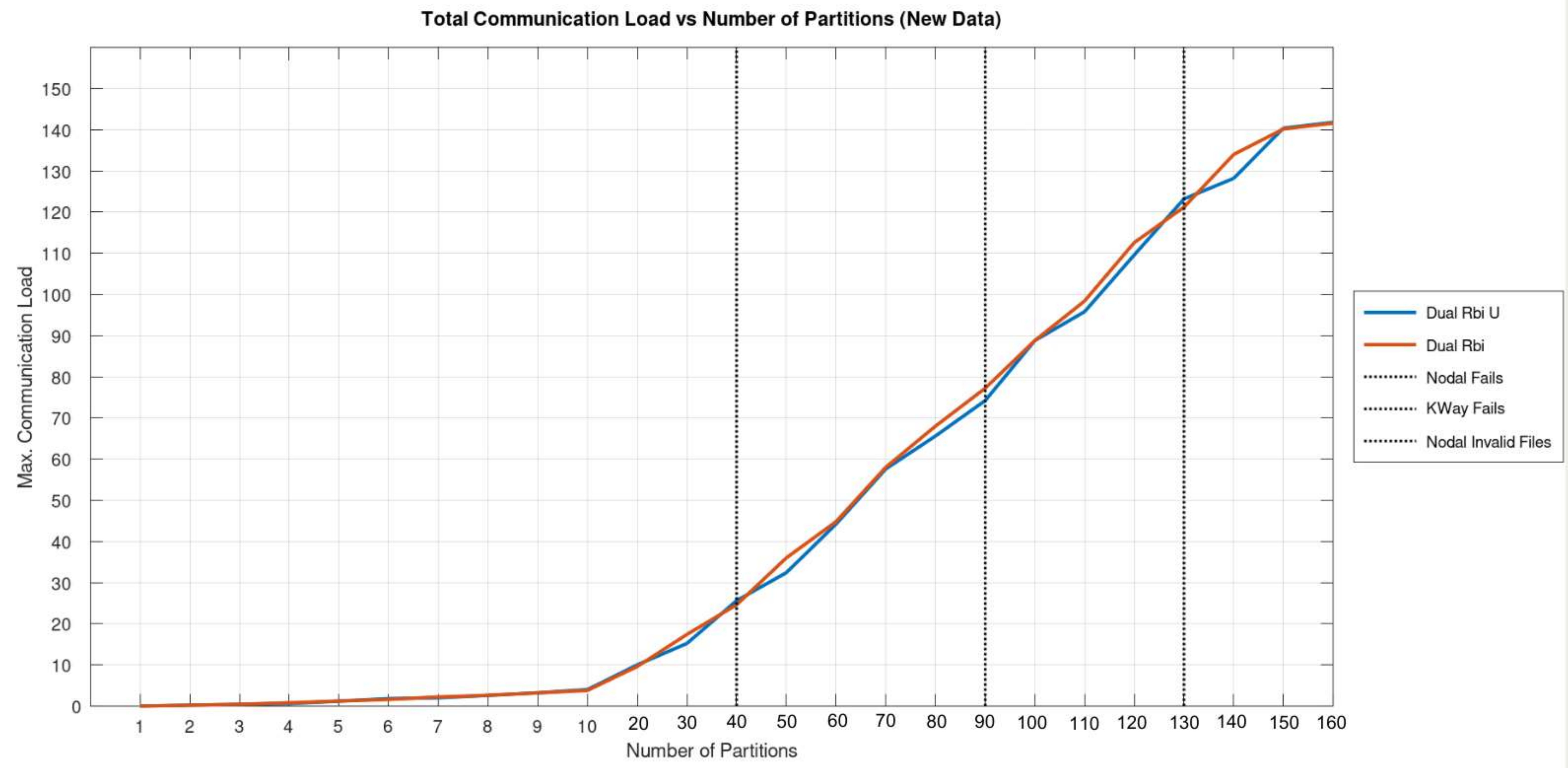
COMMUNICATION LOAD GRAPH FOR STITCHED MESH



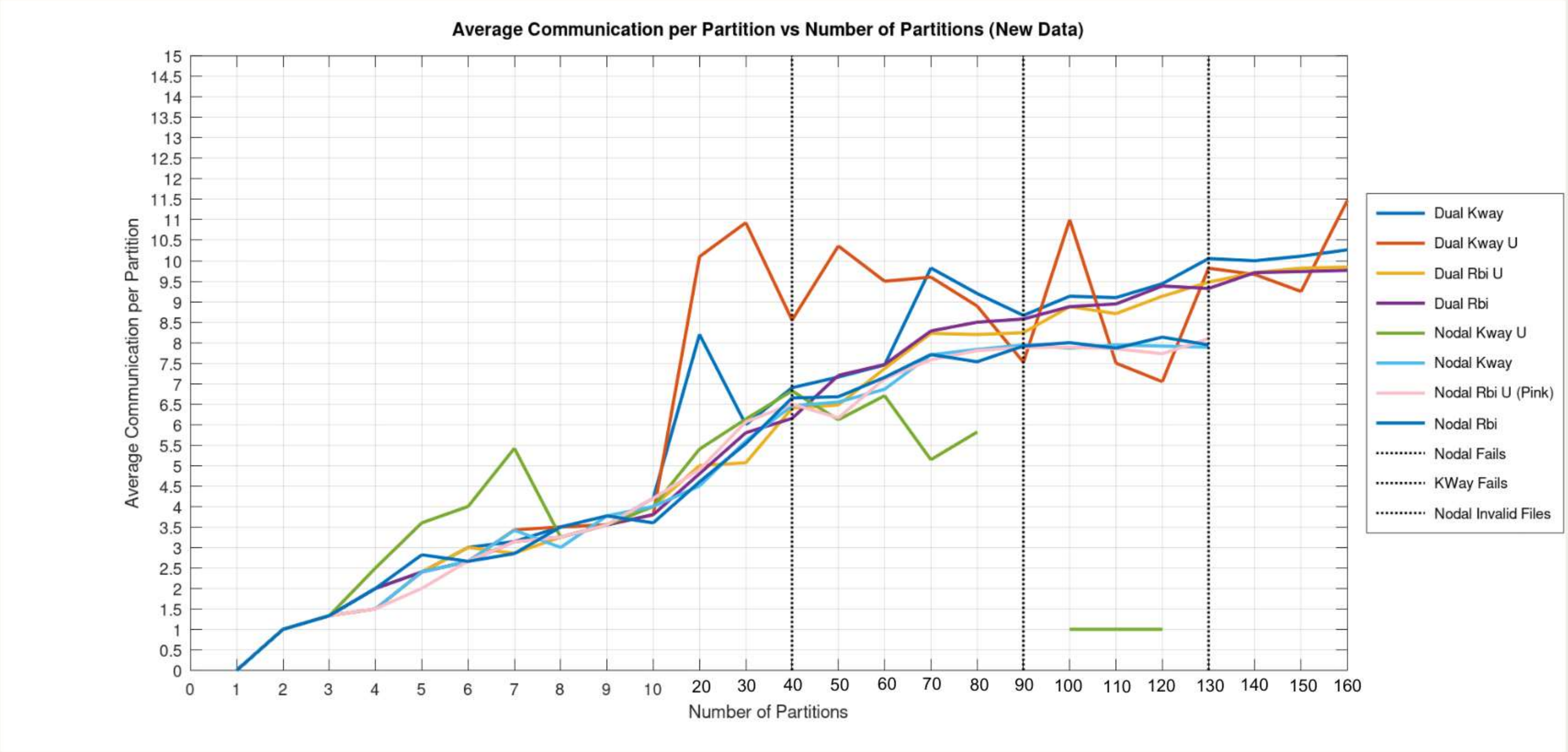
TOTAL COMMUNICATION LOAD GRAPH



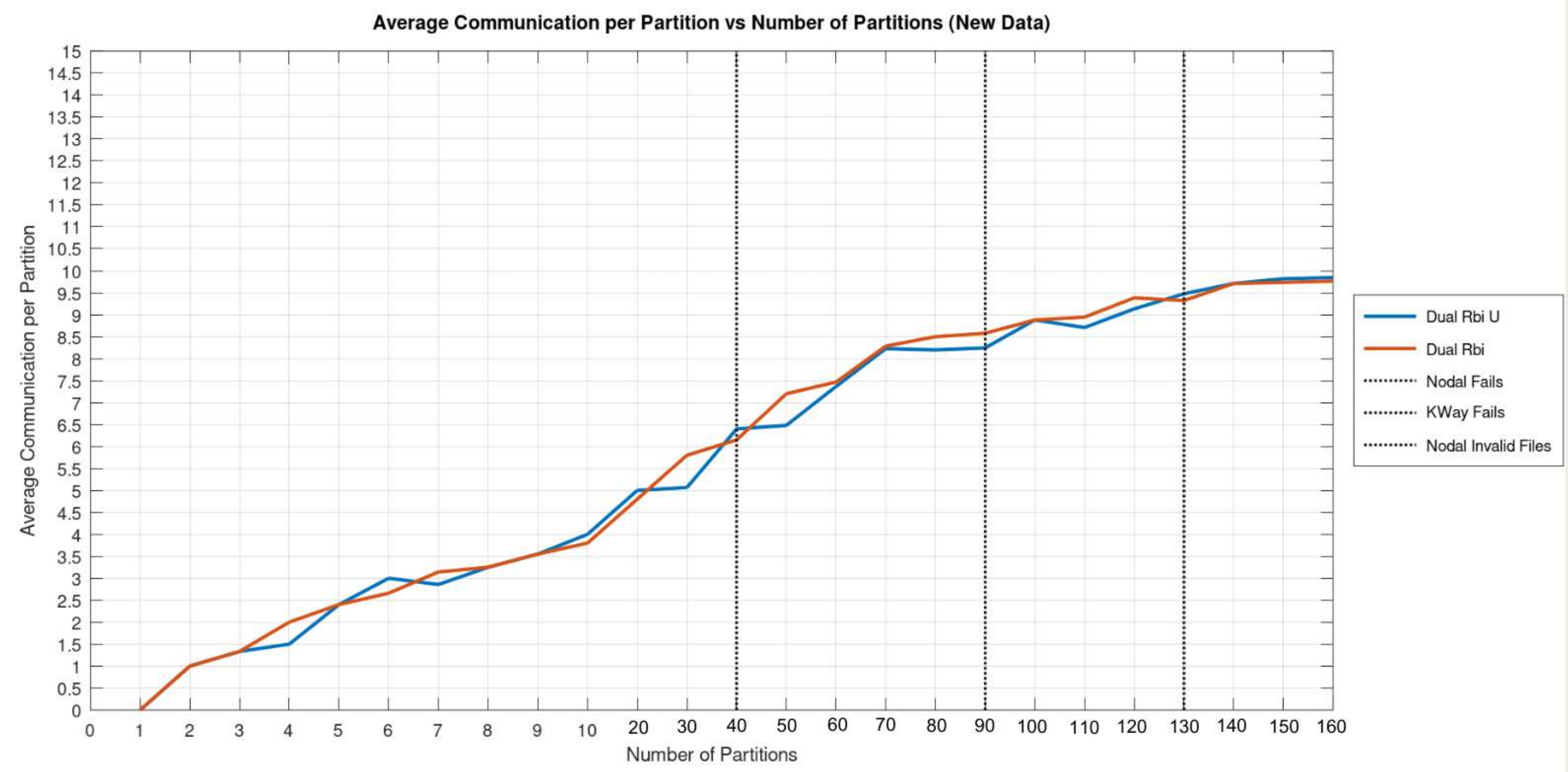
TOTAL COMMUNICATION LOAD GRAPH



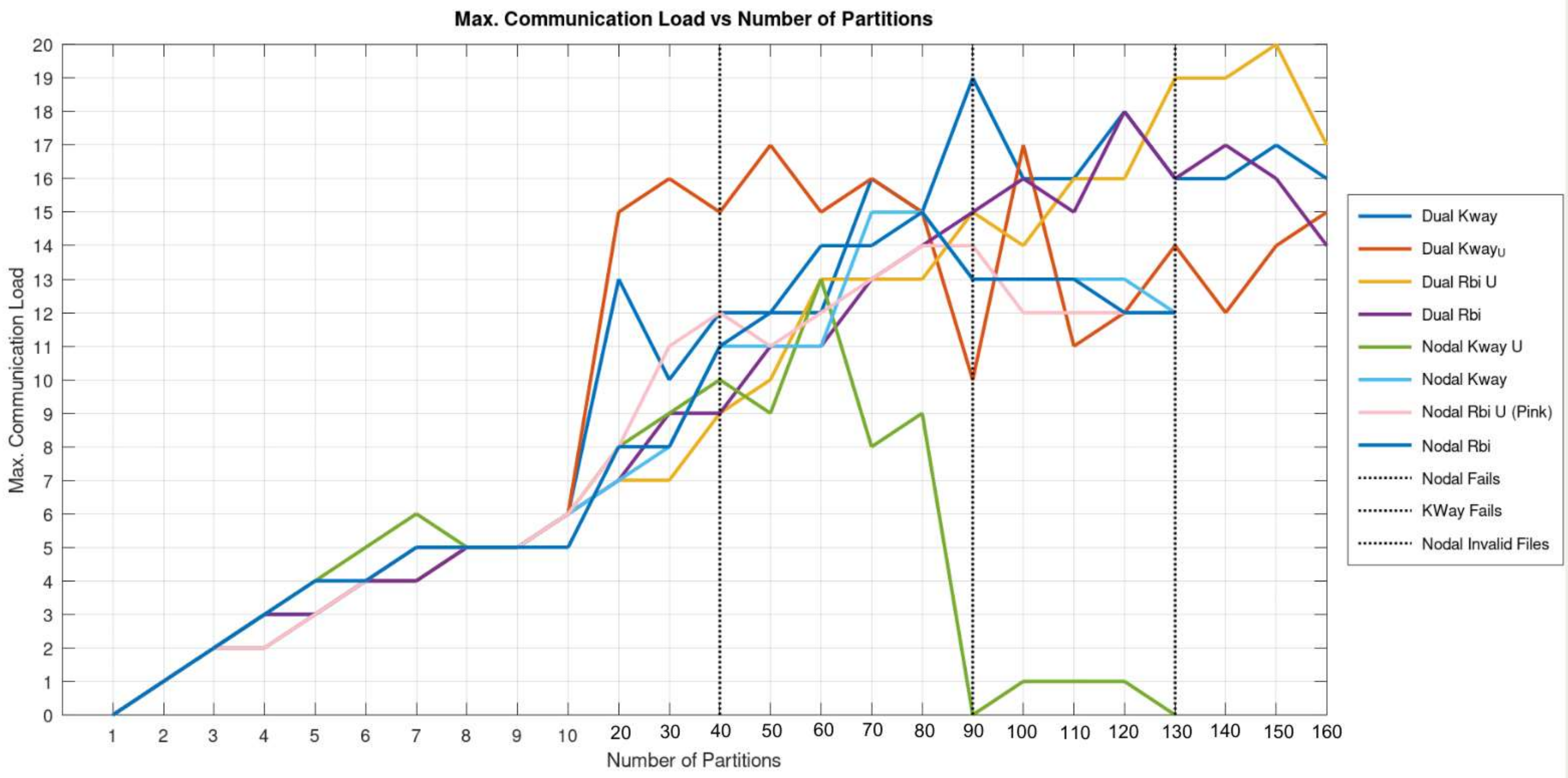
AVERAGE COMMUNICATION LOAD GRAPH



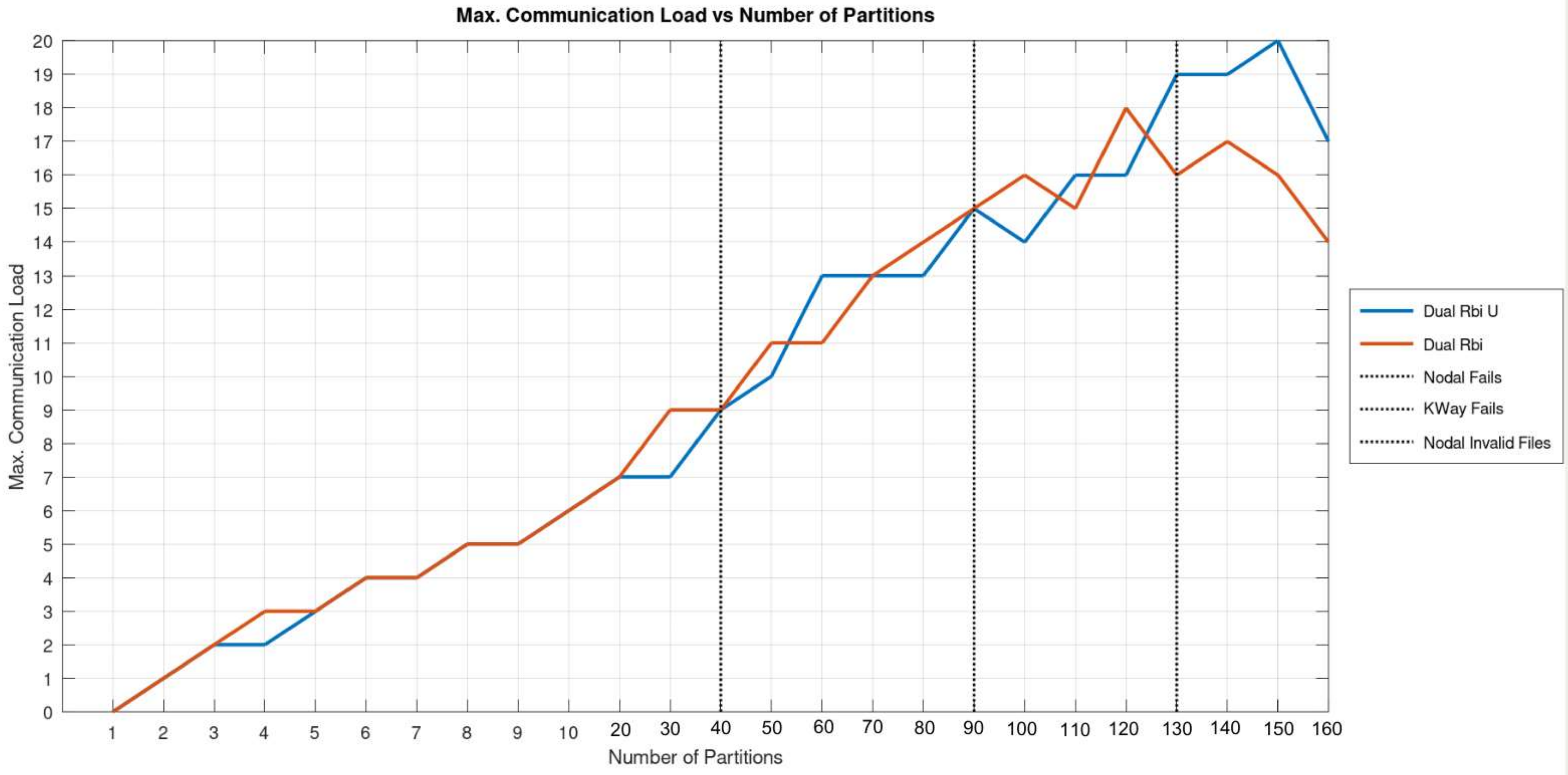
AVERAGE COMMUNICATION LOAD GRAPH



MAXIMUM COMMUNICATION LOAD GRAPH



MAXIMUM COMMUNICATION LOAD GRAPH



DRAWBACKS OF CURRENT METHODOLOGY

- **Difficulty** - Mesh-to-Graph Mapping
- **Time Consumption** - Multiple commands
- **File Handling** - Multiple File Generation

DATA FORMATS AND EASE OF PARSING

Ways to ease data extraction:

Format	Easy Data Parsing
.vtk	✓
.obj	✗
.blend	✗
.msh	✗
.stl	✗
.vtu(XML)	✗

CONTENT

TROUBLESHOOTING

```
Analyzing file: ../../../JEDI/Opti_parti3.vtk50Nodal_kway_Unweighted.vtk
Partition Communication Analysis:
Partition 2 communicates with partitions: 4 11 30 50
Partition 4 communicates with partitions: 2 11 17 30 50
Partition 11 communicates with partitions: 2 4 17 30 33
Partition 17 communicates with partitions: 4 11 24 29 30 33 34 36 50
Partition 19 communicates with partitions: 29 30 33 34 36 40
Partition 24 communicates with partitions: 17 27 30 34 36 45 50
Partition 27 communicates with partitions: 24 29 34 36 40 45 47 48
Partition 29 communicates with partitions: 17 19 27 30 33 34 36 40
Partition 30 communicates with partitions: 2 4 11 17 19 24 29 33 34 50
Partition 33 communicates with partitions: 11 17 19 29 30 34 40
Partition 34 communicates with partitions: 17 19 24 27 29 30 33 36
Partition 36 communicates with partitions: 17 19 24 27 29 34 45 50
Partition 40 communicates with partitions: 19 27 29 33 45 47 48
Partition 41 communicates with partitions: 45 47 48
Partition 45 communicates with partitions: 24 27 36 40 41 47 48 50
Partition 47 communicates with partitions: 27 40 41 45
Partition 48 communicates with partitions: 27 40 41 45 50
Partition 50 communicates with partitions: 2 4 17 24 30 36 45 48
Partition Communication Analysis for file: ../../../JEDI/Opti_parti3.vtk50Nodal_kway_Unweighted.vtk
Total communications: 120
Average communications per partition: 6.66667
Maximum communication load (max number of communications a partition has): 10

The file with the minimum communication is: ../../../JEDI/Opti_parti3.vtk50Nodal_kway_Unweighted.vtk
```

Nodal Failure after 30 Partitions

TROUBLESHOOTING

```
Analyzing file: ../../../JEDI/Opti_parti3.vtk100Nodal_kway_Unweighted.vtk
Partition Communication Analysis:
Partition 48 communicates with partitions: 95 100
Partition 95 communicates with partitions: 48 100
Partition 100 communicates with partitions: 48 95
Partition Communication Analysis for file: ../../../JEDI/Opti_parti3.vtk100Nodal_kway_Unweighted.vtk
Total communications: 6
Average communications per partition: 2
Maximum communication load (max number of communications a partition has): 2

The file with the minimum communication is: ../../../JEDI/Opti_parti3.vtk100Nodal_kway_Unweighted.vtk
```

K-Way Failure after 90 Partitions

TROUBLESHOOTING

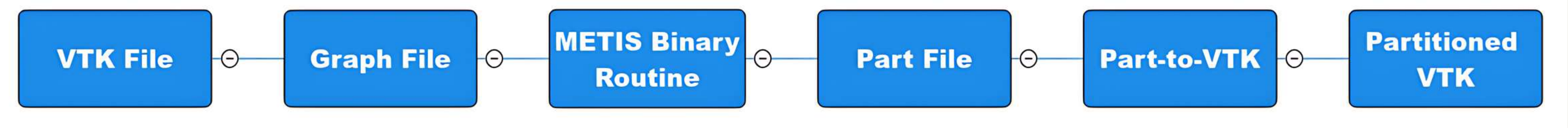
```
        ***Cannot bisect a graph with 0 vertices!
        ***You are trying to partition a graph into too many parts!
        ***Cannot bisect a graph with 0 vertices!
        ***You are trying to partition a graph into too many parts!
        ***Cannot bisect a graph with 0 vertices!
        ***You are trying to partition a graph into too many parts!
        ***Cannot bisect a graph with 0 vertices!
        ***You are trying to partition a graph into too many parts!
        ***Cannot bisect a graph with 0 vertices!
        ***You are trying to partition a graph into too many parts!
        ***Cannot bisect a graph with 0 vertices!
        ***You are trying to partition a graph into too many parts!
Partitioned mesh has been written to test.vtk
```

Nodal Generates Invalid Files

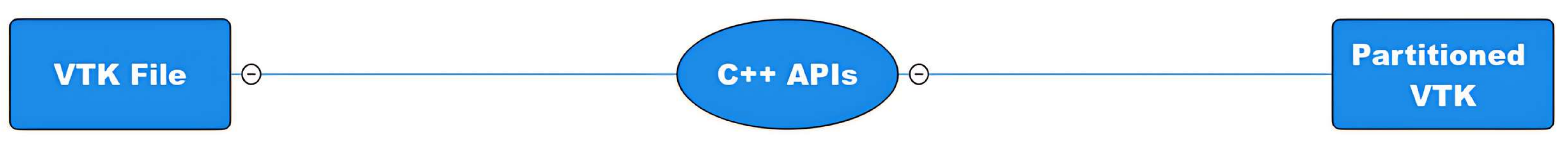
TROUBLESHOOTING

Runtime errors

TROUBLESHOOTING



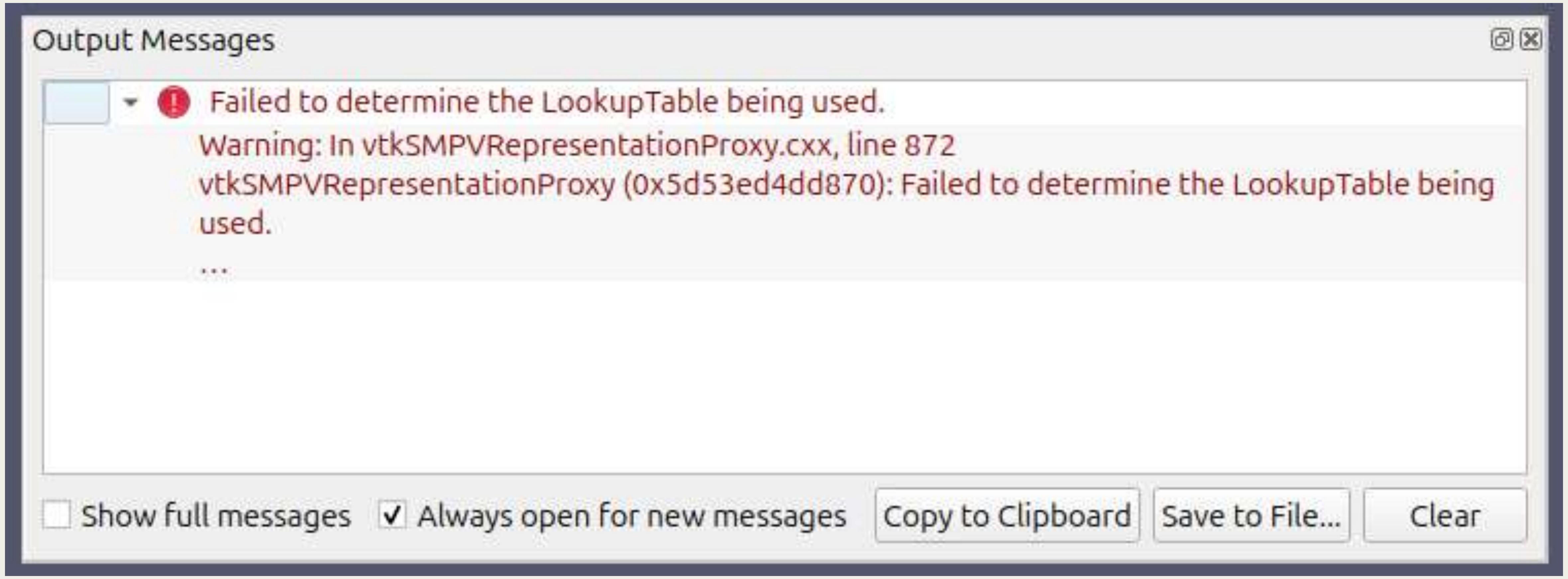
Initial Pipeline



Improved Pipeline

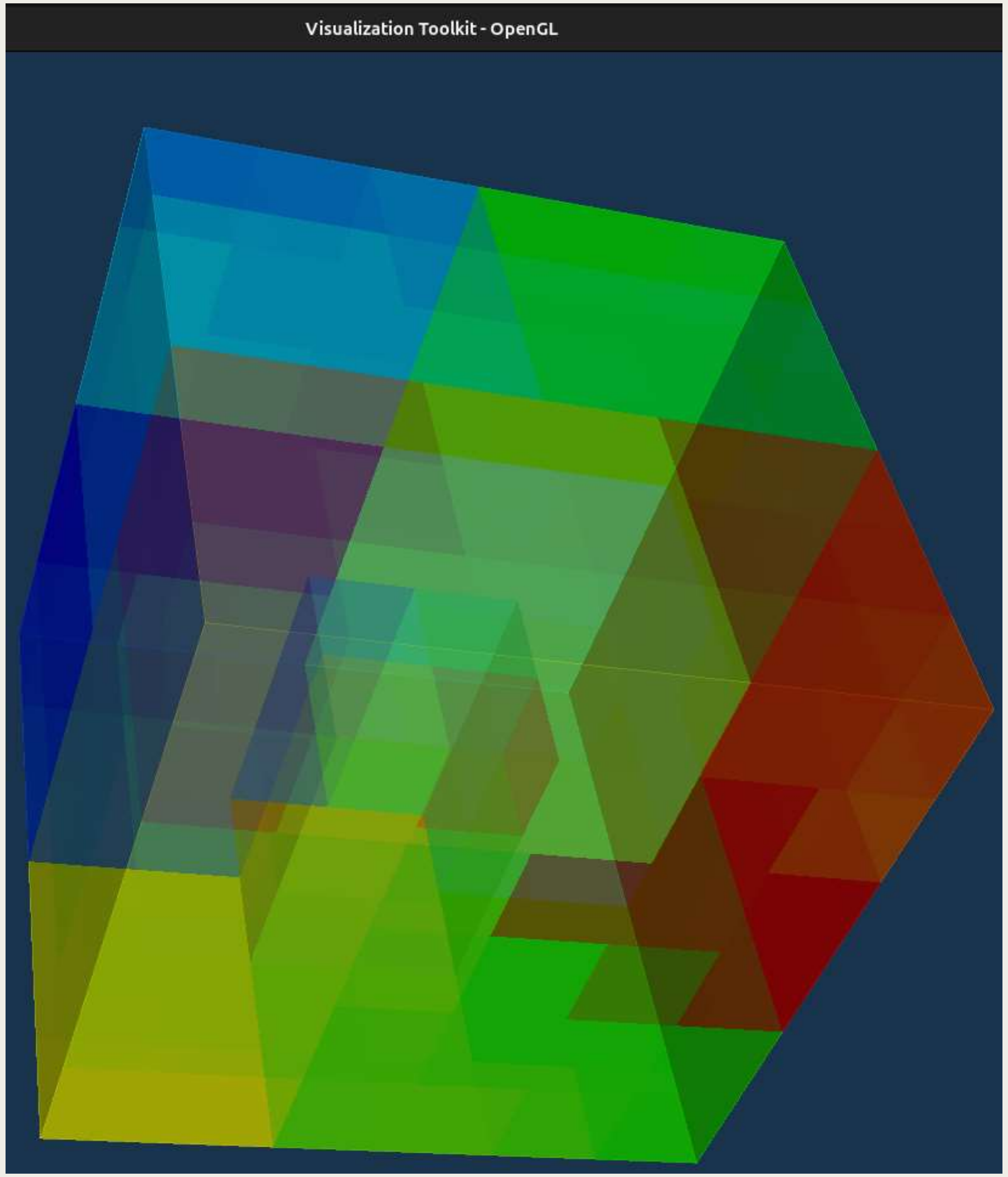
TROUBLESHOOTING

Need for Custom Renderer



The error message

TROUBLESHOOTING



Custom Renderer

CONTENT

CONCLUSION

Optimized Communication: Dual, R-Bisection, and Weighted methods minimize the communication between the geometrically separated domain and the other sub-domain.

Performance Gains: Significant reduction in inter-domain communication compared to other partitioning methods.

Algorithm Outcome: The algorithms effectively generate valid partitions for the specific meshes, meeting the required constraints.

SCOPE

Multiple Geometrically Separated Partition’s Analysis

Slotted 3D Mesh Generation

Renderer Optimization

Solver’s Runtime Analysis

Thank you!

The End, for real