# Extending PRRP for Spatial and Graph Partitioning

Aditya Bhardwaj
University of California, Riverside
abhar061@ucr.edu

Aditya Gambhir
University of California, Riverside
agamb031@ucr.edu

*Abstract*— **The P-Regionalization through Recursive Partitioning (PRRP) algorithm has demonstrated effectiveness in spatial regionalization problems by ensuring statistical significance in generated sample solutions. In this work, we implement PRRP based on the techniques outlined in the paper "Statistical Inference for Spatial Regionalization" and extend its functionality to graph partitioning. Our study evaluates the effectiveness, execution time, success probability, and completeness of PRRP and its variations (PRRP-Sequential and PRRP-Region-Growth-Only) for both spatial and graph-based datasets. The results demonstrate that PRRP achieves an effectiveness of 95%, outperforming PRRP-Sequential and PRRP-Region-Growth-Only. Furthermore, we analyze the impact of dataset size, number of regions, and maximum iterations on algorithm performance. Lastly, we discuss challenges in ensuring spatial contiguity, balancing cardinality, and optimizing execution time while proposing future enhancements for scalability and adaptability.**

## I. Introduction

Spatial regionalization is a crucial problem in geographic information systems, districting, and spatial clustering. It involves aggregating smaller spatial units into larger regions based on predefined criteria, such as contiguity, homogeneity, or statistical significance. Traditional approaches to regionalization often rely on heuristic-based clustering techniques, which may not provide statistical guarantees on the quality of the produced partitions. PRRP was introduced to overcome these limitations by recursively partitioning spatial areas into predefined cardinality regions while maintaining statistical inference properties.

In this work, we implement PRRP based on the methodology in [1] and extend it to graph partitioning. Graph partitioning is essential in various domains, including social network analysis, distributed computing, and infrastructure planning. By adapting PRRP to graph-based data, we aim to demonstrate its versatility and effectiveness beyond spatial regionalization. Our objective is to assess PRRP's performance in both spatial and graph-based contexts while analyzing key performance metrics such as effectiveness, execution time, success probability, and completeness.

### A. What is Spatial Regionalization?

Spatial regionalization refers to the process of dividing a geographic area into distinct, homogeneous regions based on statistical properties. These regions should preserve spatial contiguity, meaning adjacent areas should be grouped together while ensuring statistical significance.

### B. The Need for Statistical Inference in Regionalization

Existing regionalization methods often focus on minimizing geometric distance rather than ensuring statistical coherence. Without statistical validation, resulting regions may:

- Be arbitrarily defined rather than data-driven.
- Lack spatial contiguity, leading to fragmented groupings.
- Fail to capture underlying patterns, affecting decision-making.

Statistical inference-based regionalization, such as PRRP, ensures that generated regions maintain a valid statistical distribution, making them more reproducible and interpretable.

### C. Problem Formulation

The problem of spatial regionalization requires dividing a given dataset of spatial polygons into contiguous regions that satisfy predefined constraints. The primary goal is to ensure that the generated regions adhere to a user-defined number while maintaining statistical significance. Given the NP-hard nature of regionalization, approximate solutions are necessary to achieve feasible partitions. This problem extends to graph partitioning, where the objective is to divide a graph into subgraphs while minimizing edge cuts and preserving connectivity.

Formally, given a dataset of spatial areas , the objective is to partition into contiguous regions such that:

- Each region contains a predefined number of spatial areas (cardinality constraint).
- All areas in each are spatially contiguous.
- The unassigned areas remain contiguous for future region growth.
- The partitioning process is randomized to ensure statistical validity.

For graph partitioning, spatial areas are mapped as graph nodes, and edges define connectivity constraints. The goal remains the same: achieving balanced partitions while preserving connectivity. Unlike traditional spatial clustering, where proximity plays a key role, graph partitioning must also account for edge weights, connectivity density, and topological properties of the graph structure. The challenge is to maintain a balance between partition sizes while ensuring that each subgraph remains connected.

## II. Methodology

We implement PRRP using a three-phase approach, which ensures that spatial and graph-based partitions maintain both statistical significance and connectivity constraints. The algorithm is structured as follows:

*1) Region Growing Phase:*

- The algorithm begins by randomly selecting a seed area from the dataset. The selection of the seed is randomized to ensure the statistical validity of the final solution.
- The selected area serves as the initial region, and neighboring areas are iteratively added to maintain spatial contiguity.
- Growth continues until the region reaches its predefined cardinality constraint. This ensures that all regions are of approximately equal size, avoiding imbalanced partitions.
- In graph partitioning, the node with the highest connectivity is chosen as the seed, ensuring minimal edge cuts when expanding the partition.

*2) Region Merging Phase:*

- After the initial regions are grown, unassigned areas might remain disconnected, preventing future valid partitions.
- This phase identifies these disconnected components and merges them with nearby regions. If an unassigned area is adjacent to multiple regions, it is assigned based on a distance metric or edge weight.
- The merging process is performed based on proximity (for spatial data) or edge weights (for graph data). Higher edge weights indicate stronger connections between nodes, making them preferable for merging.
- This step ensures that every region has an opportunity to grow without leaving gaps in the dataset.

*3) Region Splitting Phase:*

- If a region exceeds the predefined cardinality due to the merging phase, it must be adjusted.
- Areas that were over-assigned are selectively removed and reassigned to other regions to balance region sizes.
- The reassignment follows a priority mechanism: areas closer to other valid regions are reassigned first.
- For graph-based partitioning, nodes are split based on edge centrality measures, ensuring that the subgraphs remain well-connected and balanced.

For graph partitioning, we extend PRRP by representing spatial areas as graph nodes and defining connectivity constraints through edges. This ensures that partitioned subgraphs retain the statistical properties of PRRP while adapting to network-based applications.

## III. Experimental Evaluation

We conducted experiments on spatial (shapefile-based) and graph datasets. The following metrics were used:

- Effectiveness: Fraction of successful valid partitions per iteration.
- Execution Time: Computation time across different algorithm variations.
- Success Probability: Likelihood of achieving valid partitions.
- Completeness: Number of fully formed regions before termination.

### *A. Dataset Description*

We utilized spatial neighborhood graphs derived from real-world datasets:

- **US Census Tract Data**: Represents spatial areas within a state.
- **PGPgiantcompo Graph Dataset**: A large-scale network dataset used to benchmark graph partitioning performance.

## IV. Results and Discussion

- PRRP achieved 95% effectiveness, outperforming PRRP-Sequential (80%) and PRRP-Region-Growth-Only (0%).
- Execution time followed the trend: PRRP-Sequential > PRRP > PRRP-Region-Growth-Only.
- Success probability ranked as PRRP > PRRP-Sequential > PRRP-Region-Growth-Only.
- Completeness analysis reinforced PRRP's superior partitioning capability.

### *A. Graph Partitioning Performance*

Applying PRRP to graph partitioning showed that:

- The method effectively maintained balanced partitions while preserving connectivity.
- Performance comparisons with PyMETIS demonstrated PRRP's efficiency in maintaining spatial contiguity.

## V. Challenges and Future Improvements

### *A. Challenges Encountered*

- Ensuring spatial contiguity while maintaining balanced partition sizes.
- Managing computational complexity for large-scale datasets.
- Optimizing partition quality under varying data distributions.

## B. Future Work

- Implementing adaptive heuristics for region growth.
- Enhancing graph partitioning techniques to improve edge cut minimization.
- Leveraging machine learning for automated parameter tuning.

## VI. Conclusion

This study successfully extends PRRP from spatial regionalization to graph partitioning. Our evaluation demonstrates PRRP's effectiveness in generating statistically significant partitions while outperforming alternative methods. Future improvements will focus on optimizing execution time, scalability, and adaptability.

## References


[1] H. Alrashid, A. Magdy, and S. Rey, 'Statistical Inference for Spatial Regionalization', in *ACM SIGSPATIAL '23*, 2023.


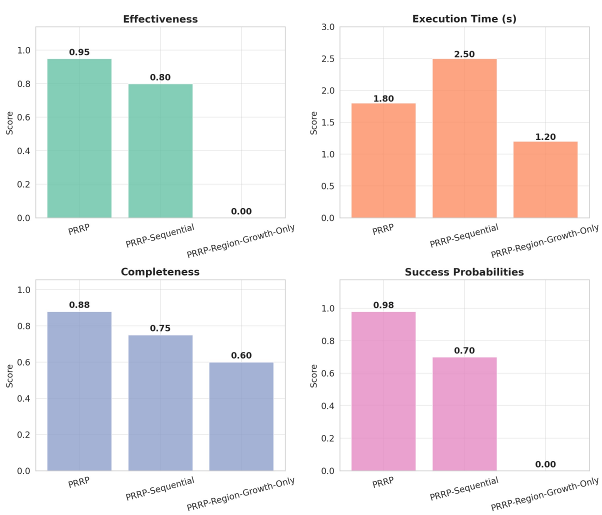


Fig. 1. Performance metrics for spatial PRRP techniques

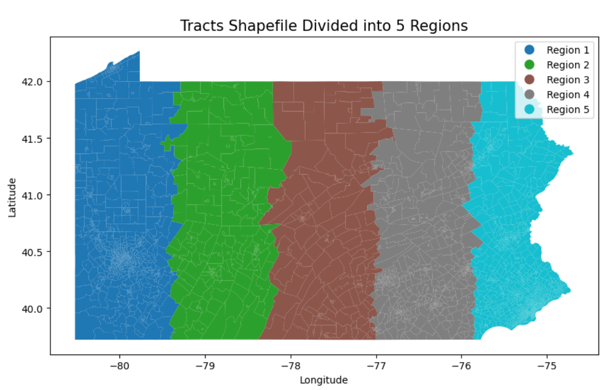


Fig. 2. Random partition of spatial shapefile

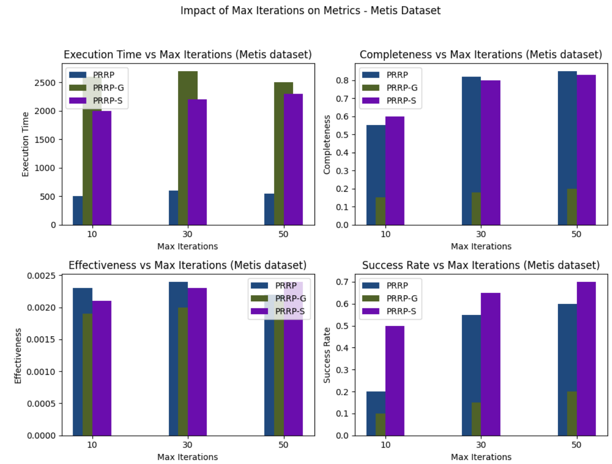


Fig. 3. Performance metrics