CS677 Course Project

Parallel Probabilistic Data Sampling: Implementation & Scaling Study

Course Instructors
Prof. Preeti Malakar
Prof. Soumya Dutta

Group Members
Dasari Charithambika
Divya Gupta
Om Shivam Verma
Palak Mishra
Siddharth Pathak

Contents

Introduction

- Sampling is a data analysis method involving the selection of a subset from a larger dataset.
- Instead of analyzing every piece of data, sampling focuses on a representative group.
- Efficiency, Cost-Effective, Inference, Bias Reduction

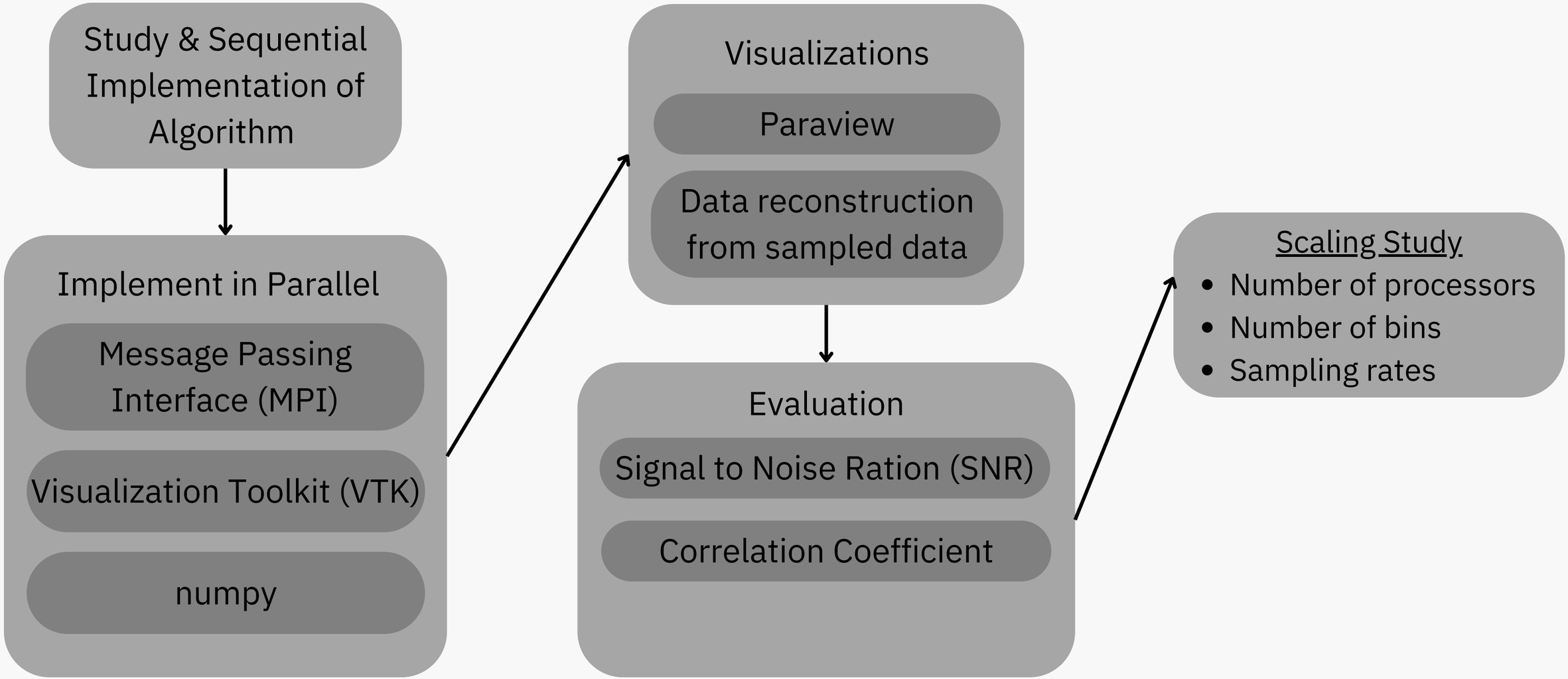
Motivation

- Tech Challenge, Data Reduction, Feature Detection, Data Slimming Down, and Versatile Visualization

Problem Statement

- Feature-based exploration in scientific datasets faces challenges, as existing methods do not effectively prioritize feature regions over non-feature regions.
- Parallel implementation of existing sampling methods - Simple random sampling, Value-based Sampling, and Smoothness-based sampling.
- Reconstruction based visualization and scaling of all implemented methods.

Methodology



Methodology: Evaluation, Visualization

Signal-to-Noise Ratio

- To estimate the quality of the reconstruction
- Larger values indicate better reconstructions.

$$SNR = 20 * log_{10} \frac{\sigma_{raw}}{\sigma_{noise}}.$$

σ raw is the overall standard deviation of the original data.
The quantity **σ** noise is the standard deviation of the error of the reconstruction

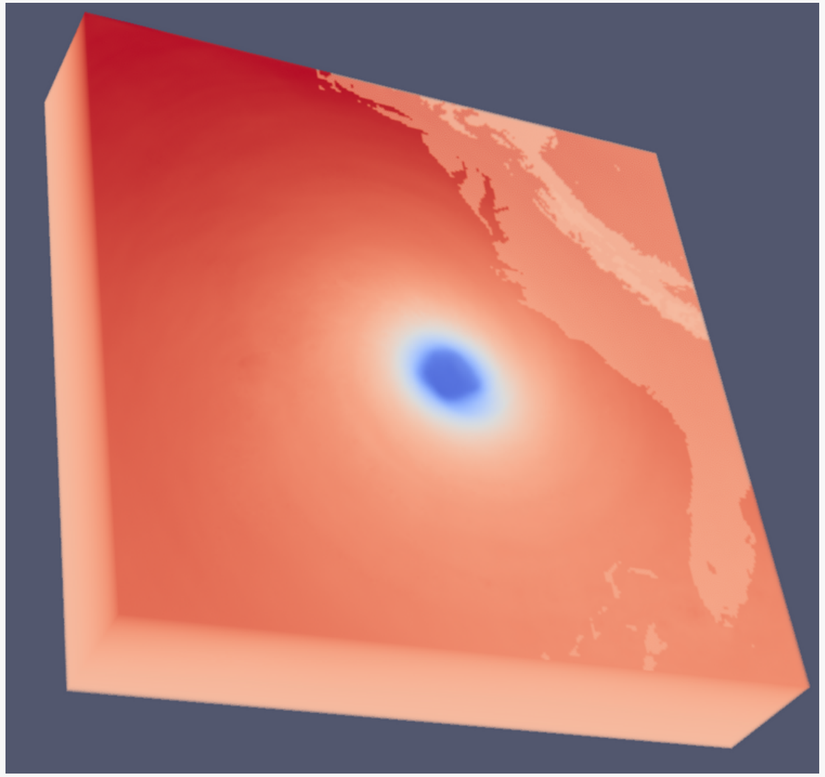
Correlation coefficient Plot

$$r = \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum (x_i - \bar{x})^2 \sum (y_i - \bar{y})^2}}$$

Comparison of correlation coefficient across different sampling methods for three different datasets.

Datasets:

- We are currently leveraging the **Isabel** dataset for the purpose of visualization and implementing parallelization techniques in our computational workflows.
- We are using the two different dimensions of Isabel dataset.
- “Small” dataset: 250 x 250 x 50
- “Large” dimension: 750 x 750 x 150



Scaling Study

- The KD lab computing nodes are used.
- Each computing node runs **4 processes each**.

- Each combination of parameters was run 3-5 times, and the average time was taken
- The computed times were:
  - Input/Output (I/O): Reading and writing to disk
  - Communication time: Time for scattering and gathering the data
  - Computing time: Time required to run computation on each node

1.Simple Random Sampling

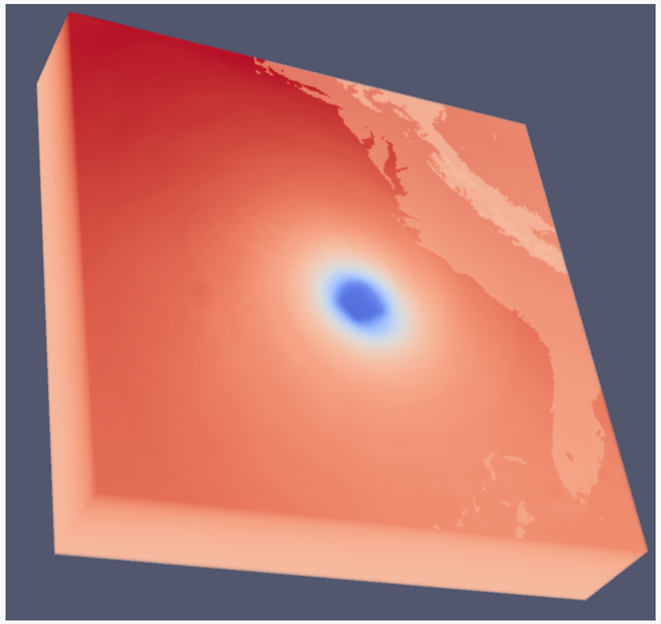
Randomly drawing samples from unknown populations.

- **Methodology**
- Given a regular grid dataset and a user-given sampling fraction, η; a random number, r can be generated from a uniform distribution U~ *unif*(0, 1).
- If r < η then this grid point will be accepted, ensuring each data point has an equal chance of getting selected.
- Do not assign priority to any specific data value that might be of importance to the scientists.

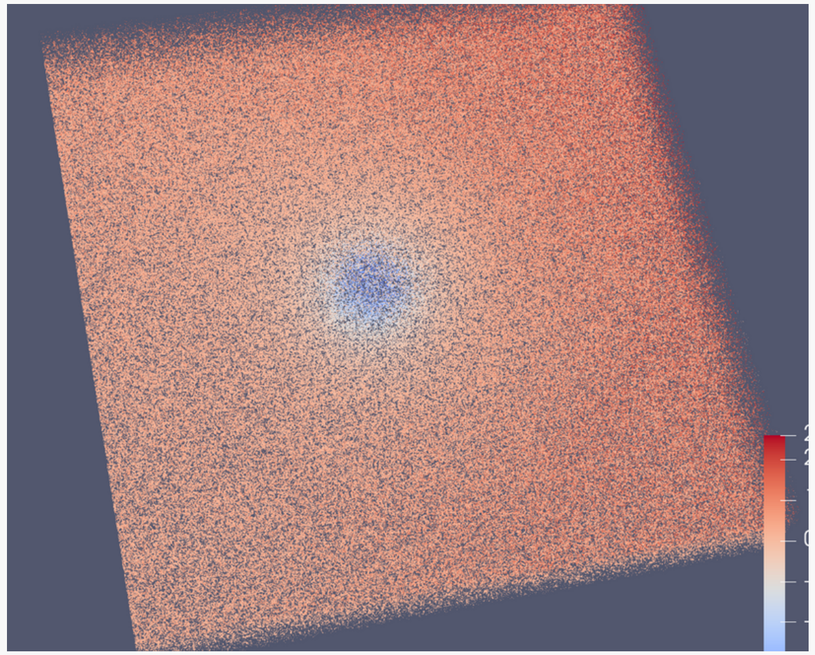
1.1 Visualisation

2% sampled

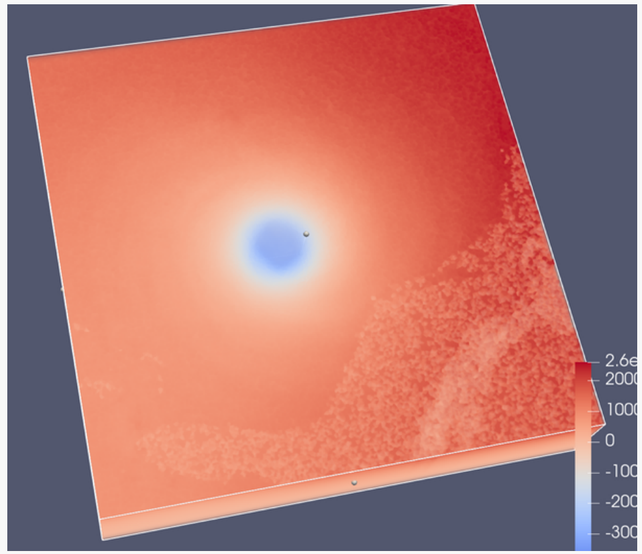
Original Dataset



Sampled Data in Paraview



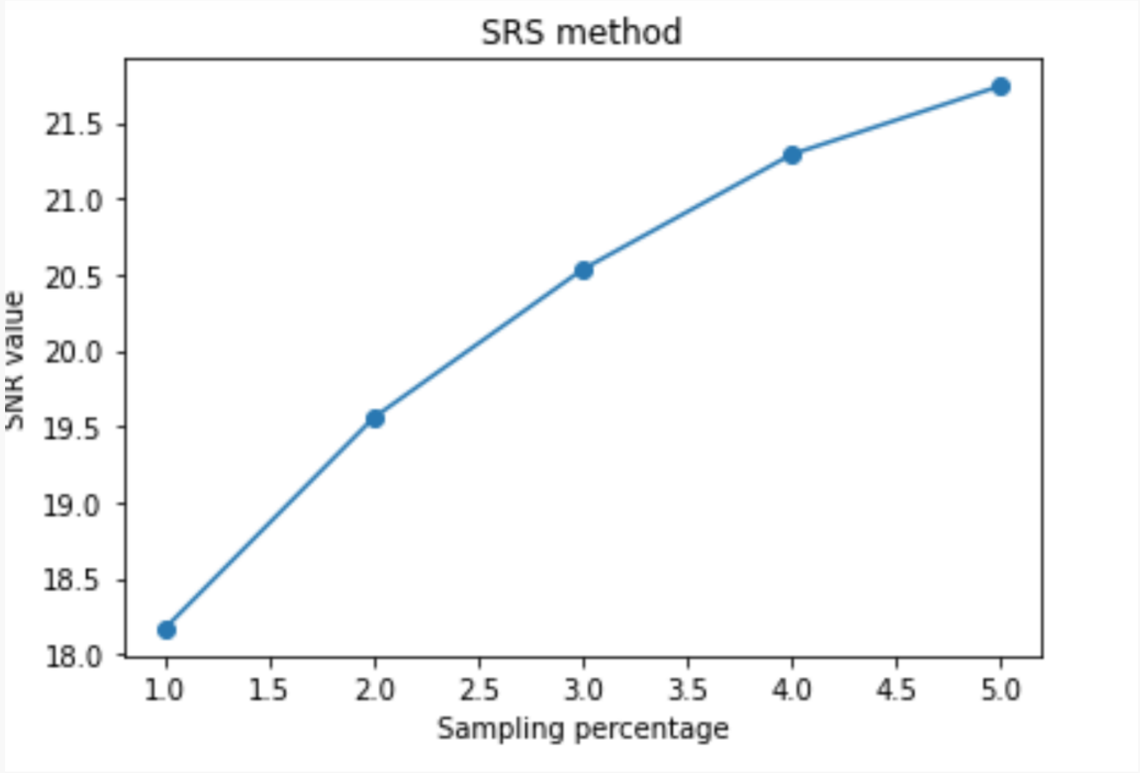
Reconstructed in Paraview



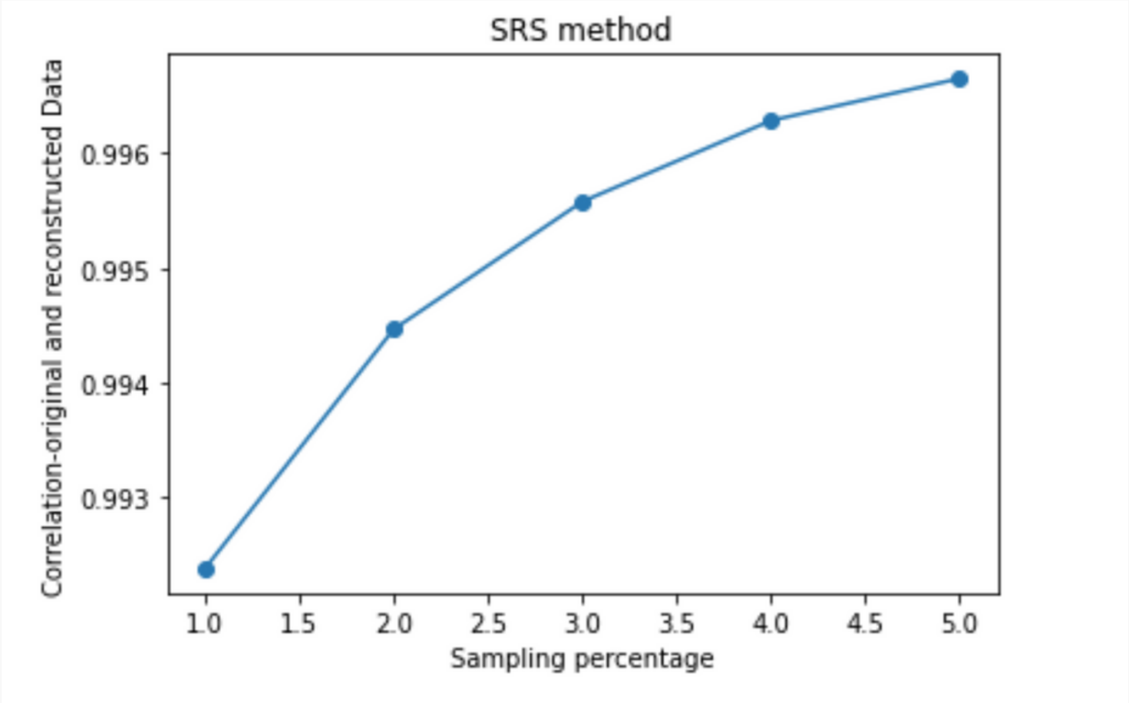
1.2 Evaluation

Correlation coefficient between original and Reconstructed datasets.a

SNR-Sampling Rate

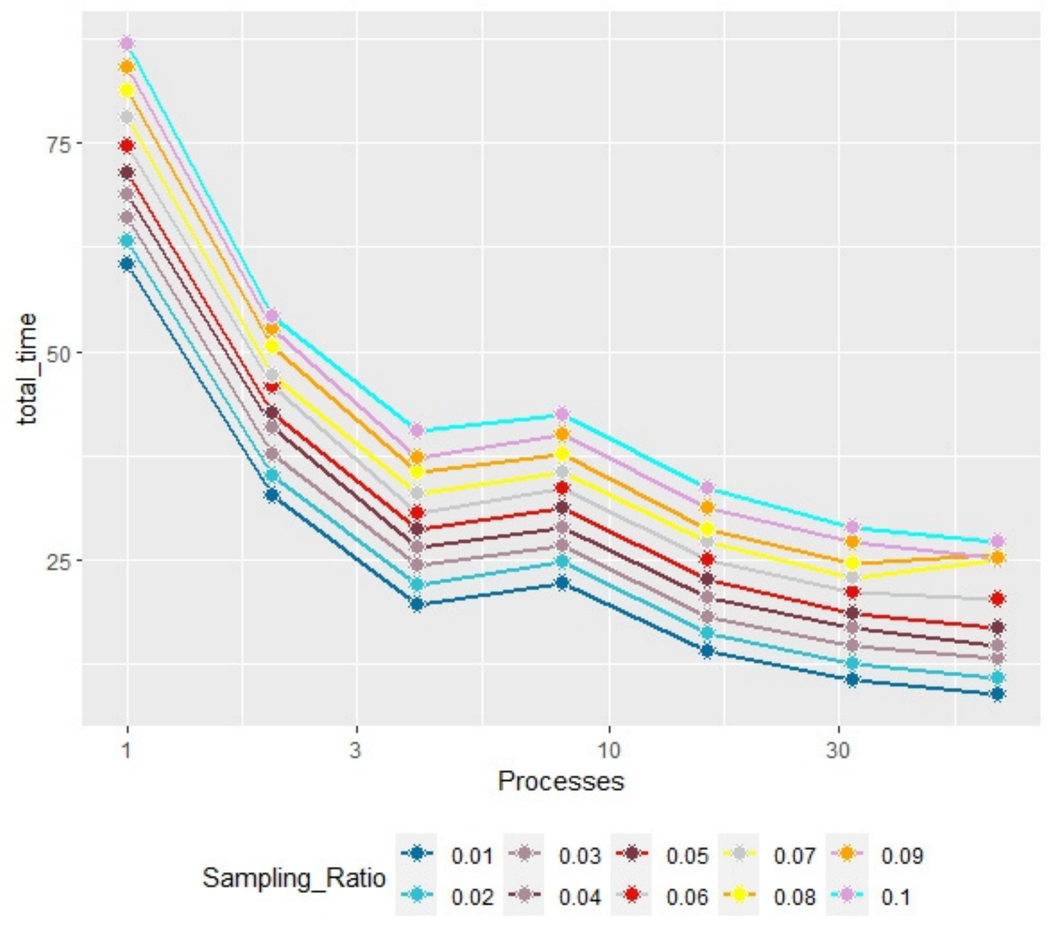


Correlation-Sampling Rate

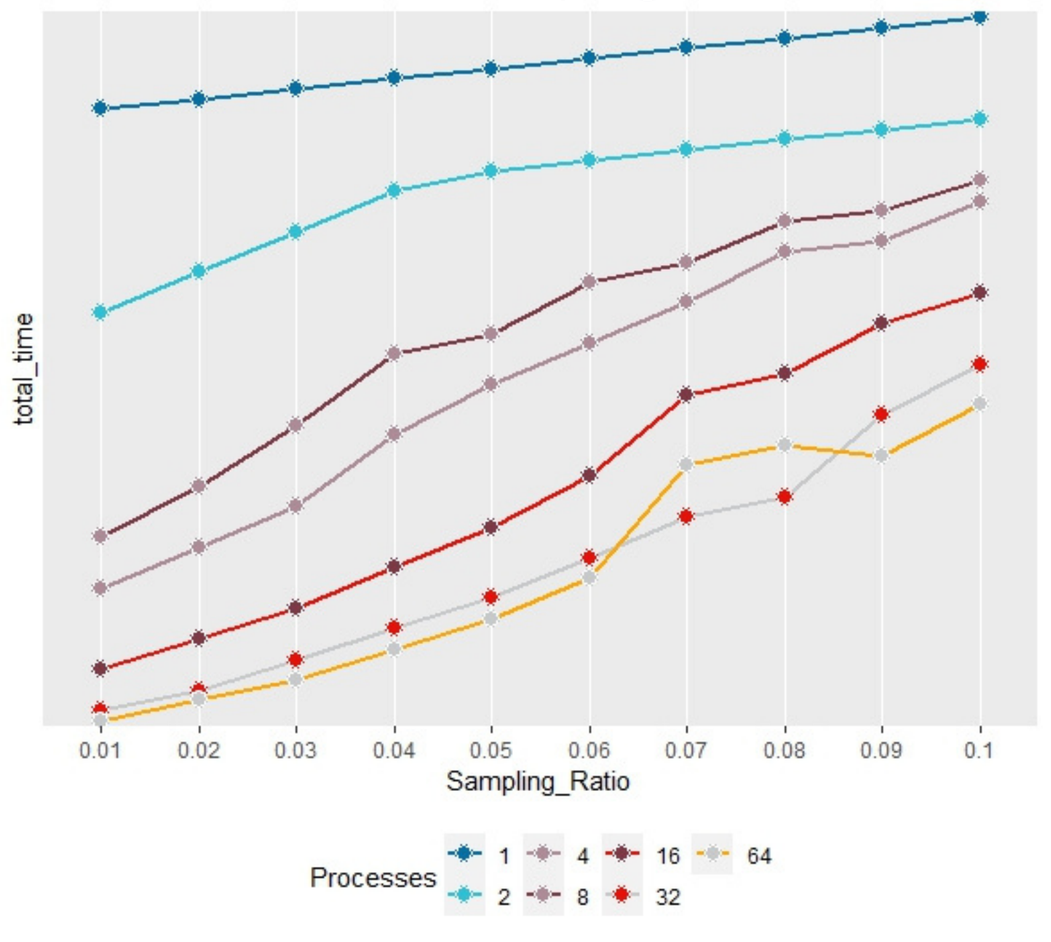


1.3 Scaling

Time - #Processes

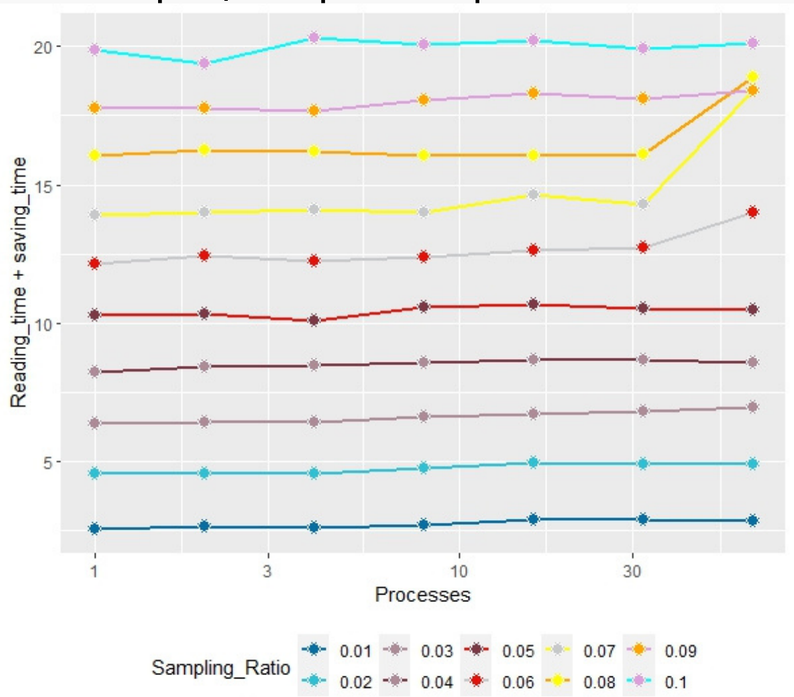


Time - Sampling Rates

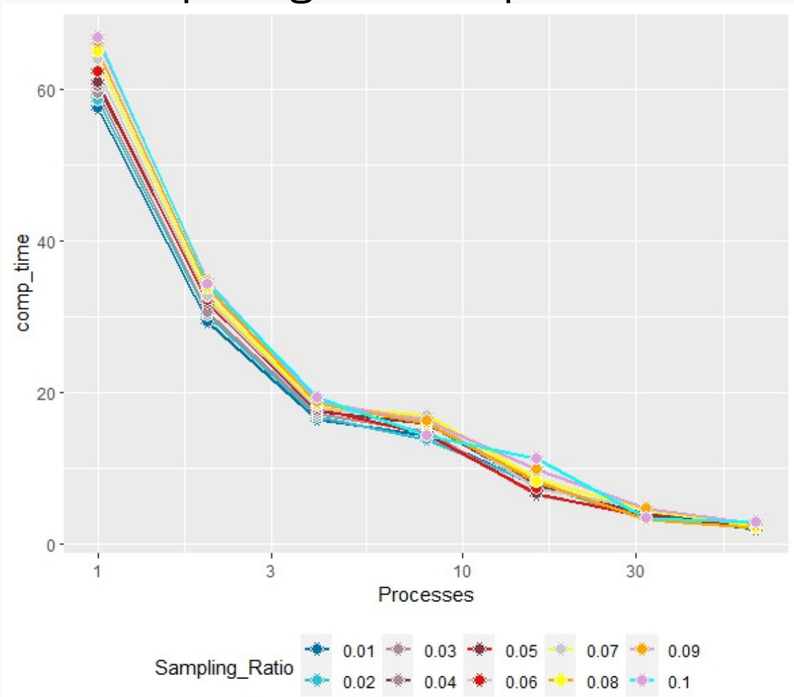


Reason of Sharp Kink in Total Time from going to 4-5 Processes

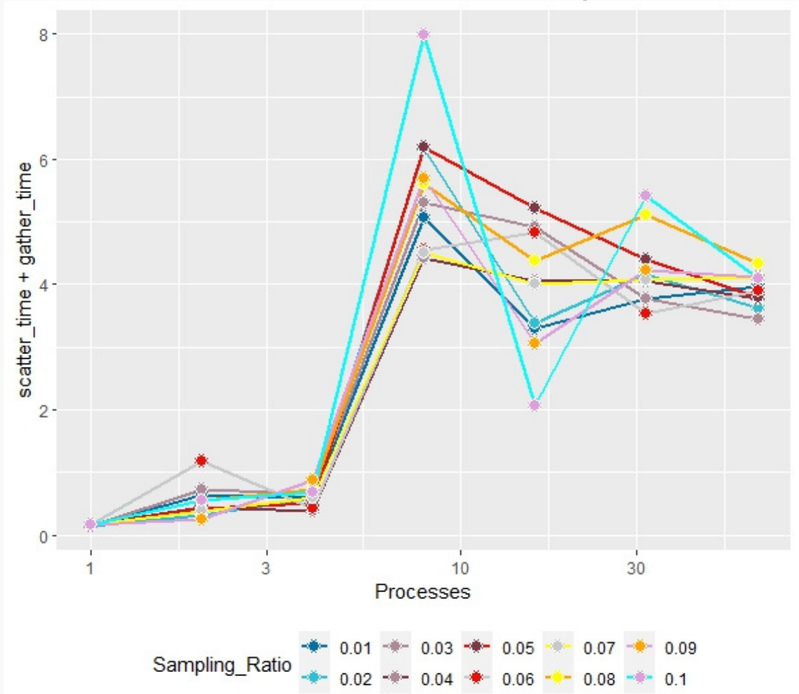
Input/Output - # processes



Computing time - # processes



Communication time - # processes



2. Value-based Importance Sampling

Method

Creating value-based histogram

Calculate of Importance function
such that sparse values are sampled more, i.e.

$$I_F(p_i) \propto \frac{1}{H(p_i)} = \frac{C}{H(p_i)}$$

Importance function:
Probability with which a point is sampled

Sampled according to importance function

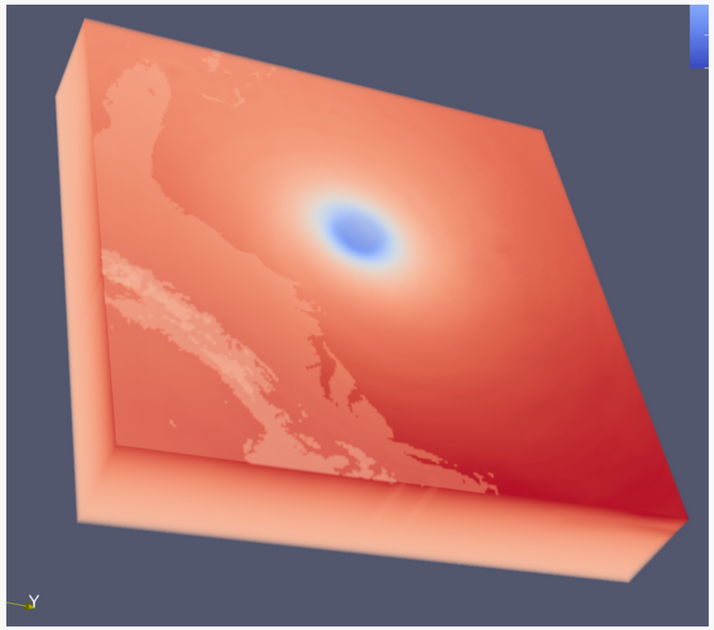
In parallel

- Read data and Scatterv to all nodes
- Allreduce the min and max values
- Count frequencies in each bin
- Allreduce to get global histogram
- Create the importance funtion
- Sample points at each node, as coordinates and values
- Gatherv to collect sampled points
- Write the points to a vtp file

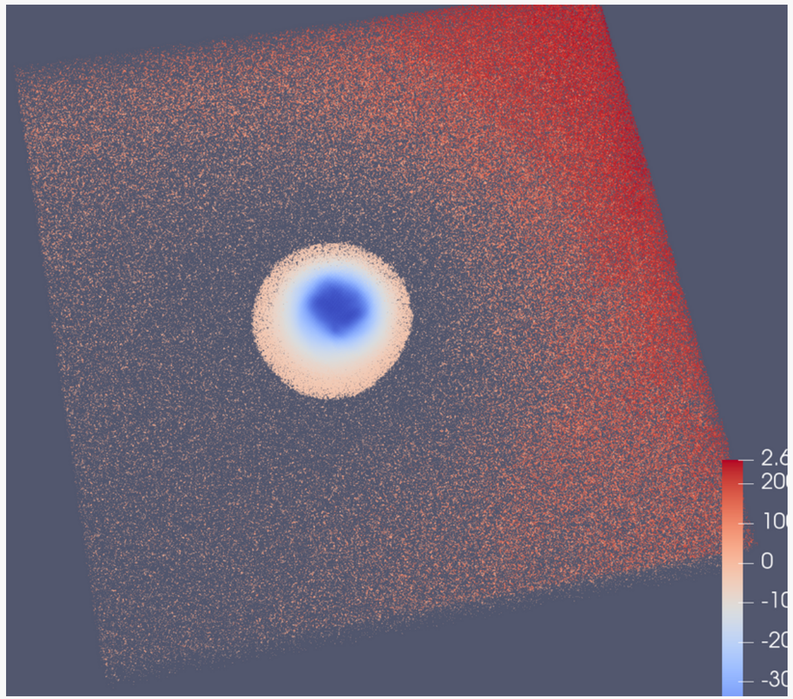
2.1 Visualisation

2% sampled

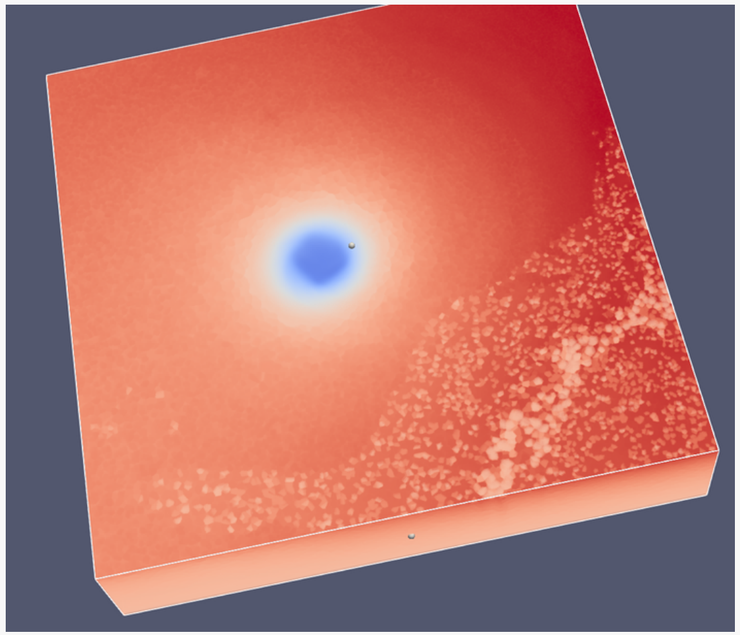
Original dataset



Sampled data in paraview



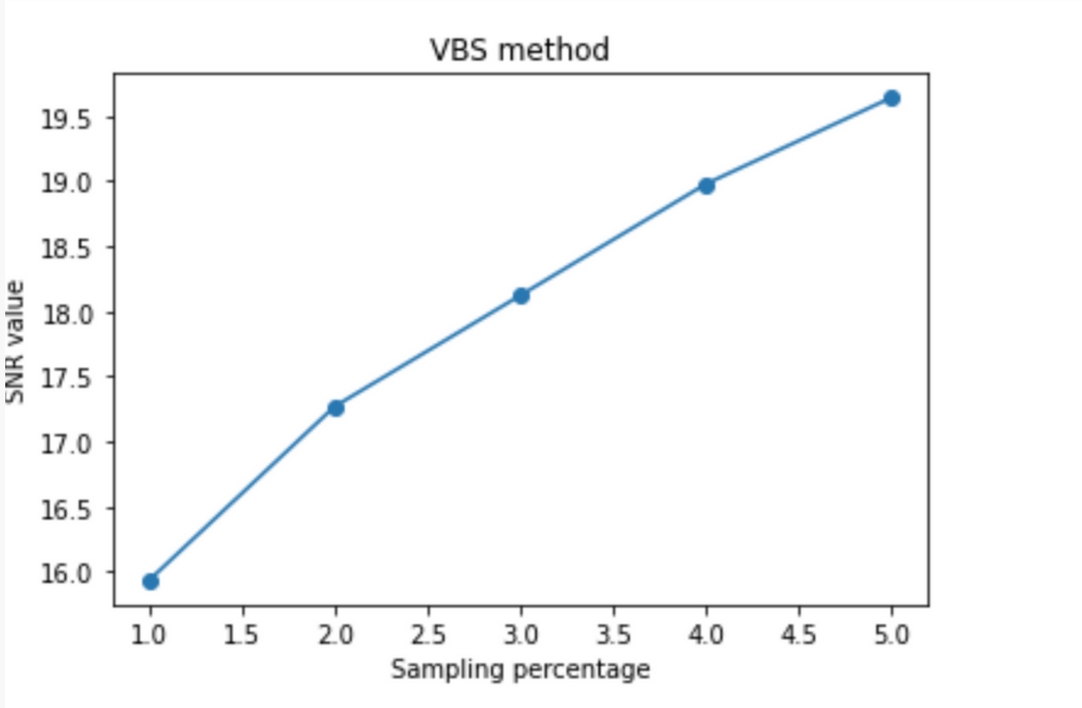
Reconstructed in Paraview



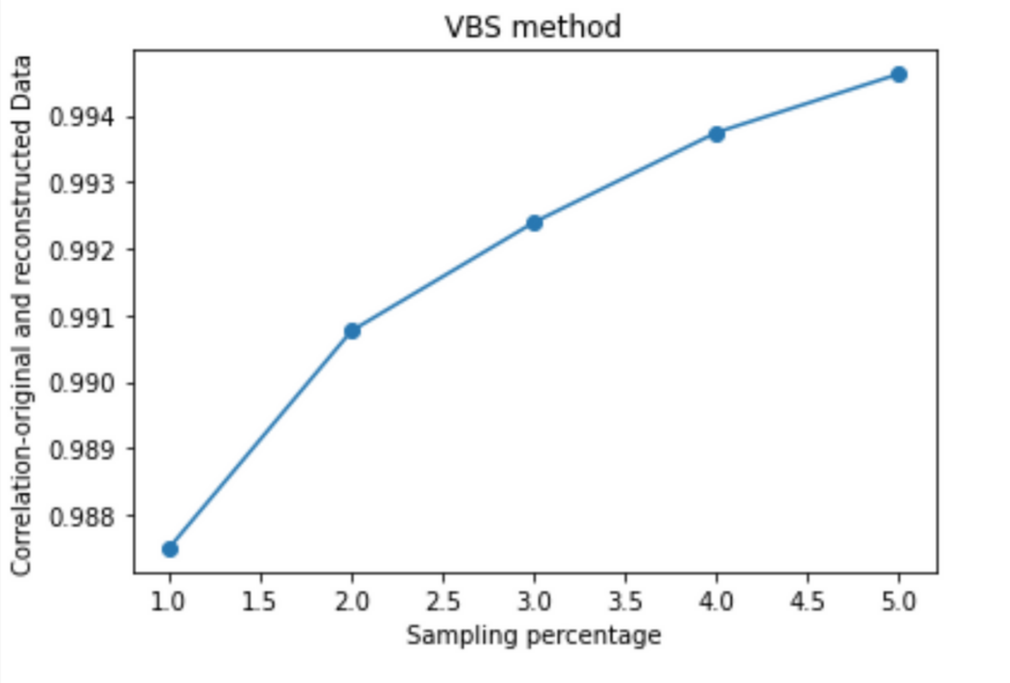
2.2 Evaluation

Correlation coefficient between original and Reconstructed datasets.a

SNR-sampling rate

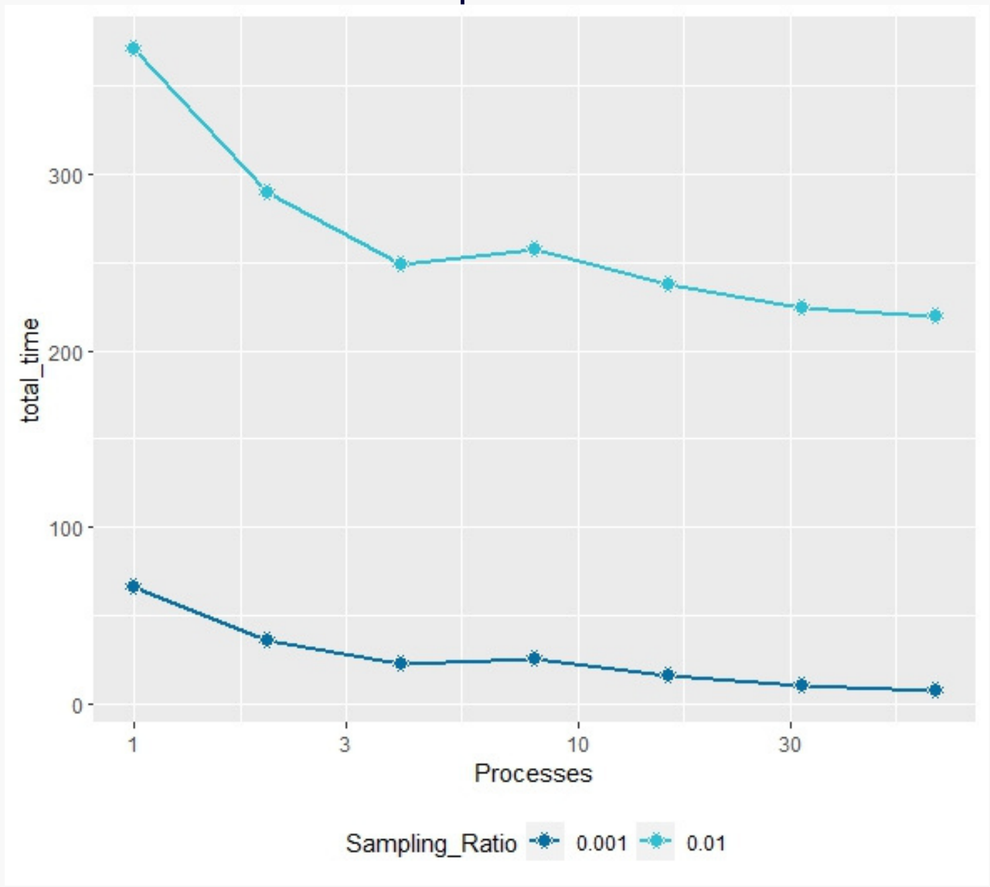


Correlation-sampling rate

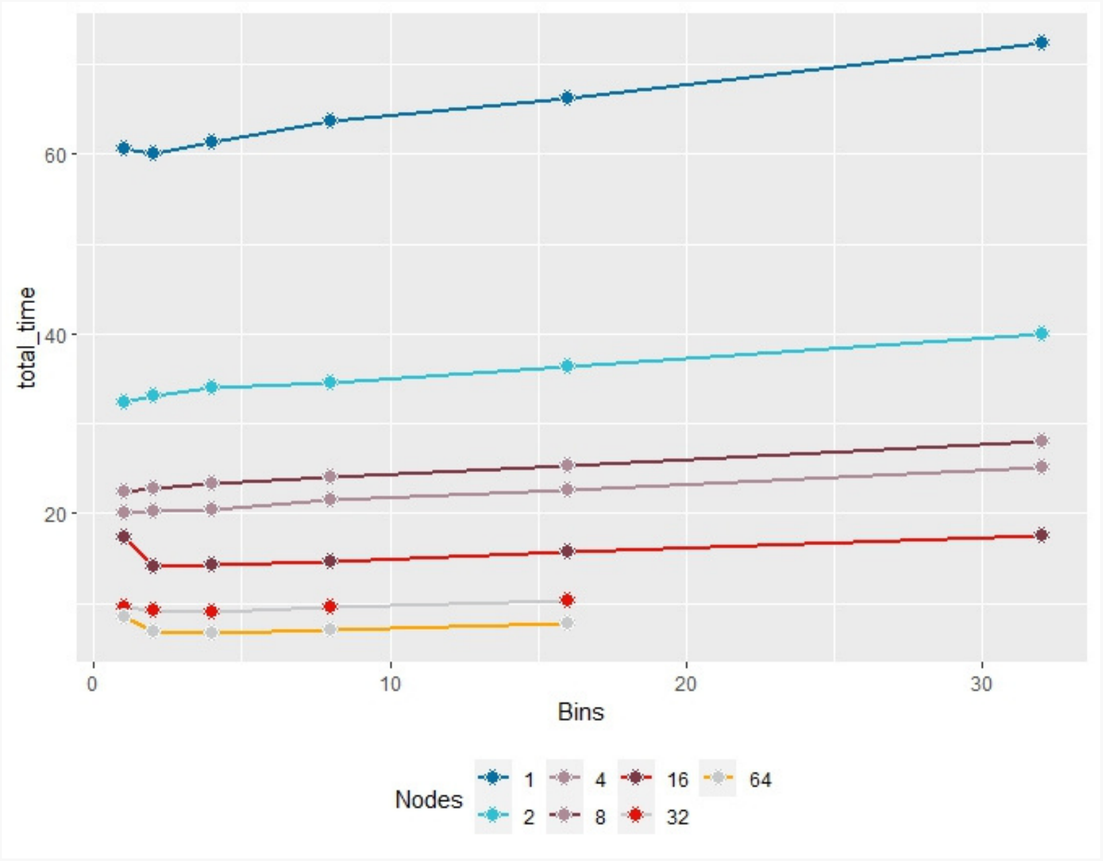


2.3 Scaling

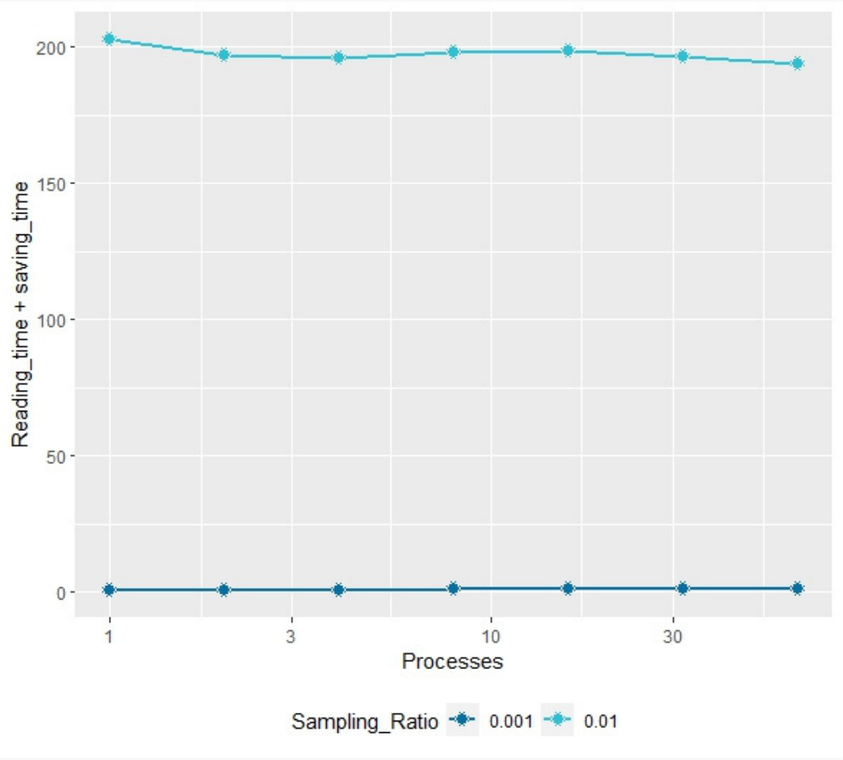
Time-#processes



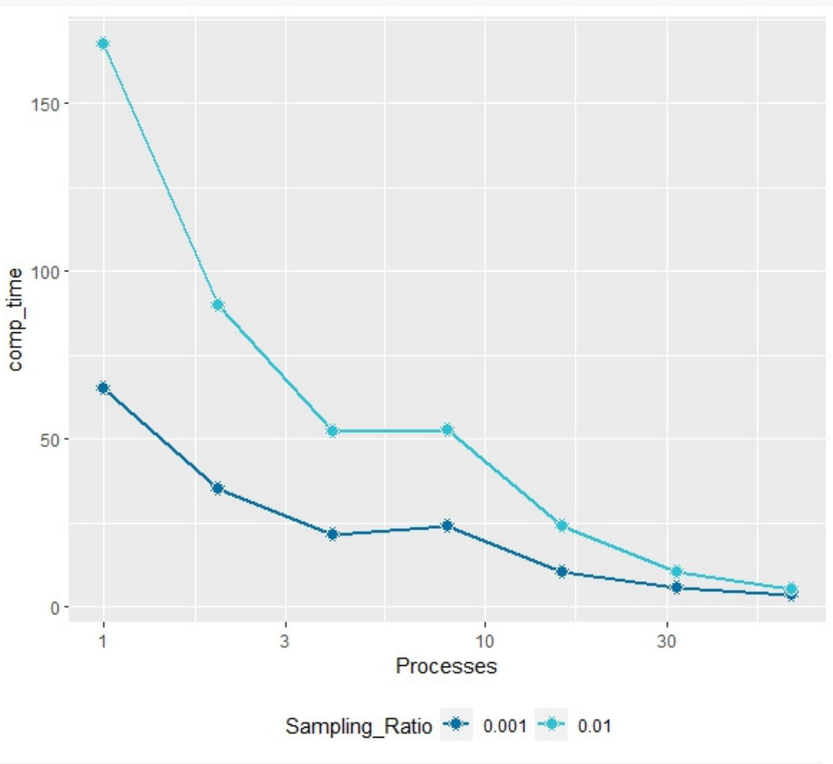
Time-#bins



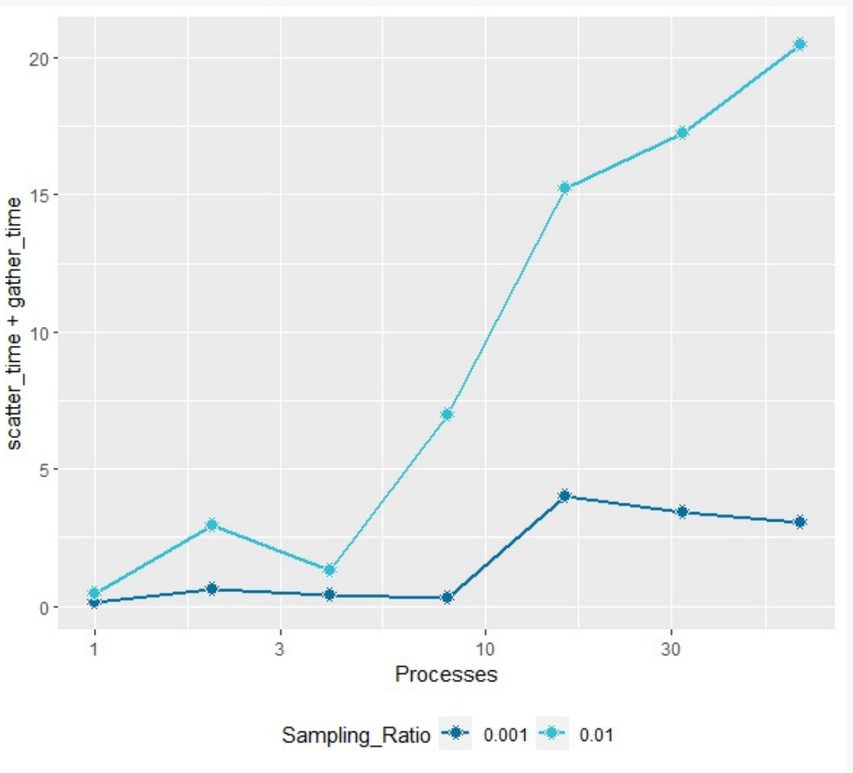
Input/Output - # processes



Computing time - # processes



Communication time - # processes



3. Smoothness-based Importance Sampling

Method

Calculate gradients

Create gradient-based histogram

Calculate of Importance function such that highest gradients are sample most, i.e.

$$I_F(p_i) \propto G(p_i)^k, \text{ where } k \in \mathbb{R}$$
$$k \to \infty$$

Sampled according to importance function

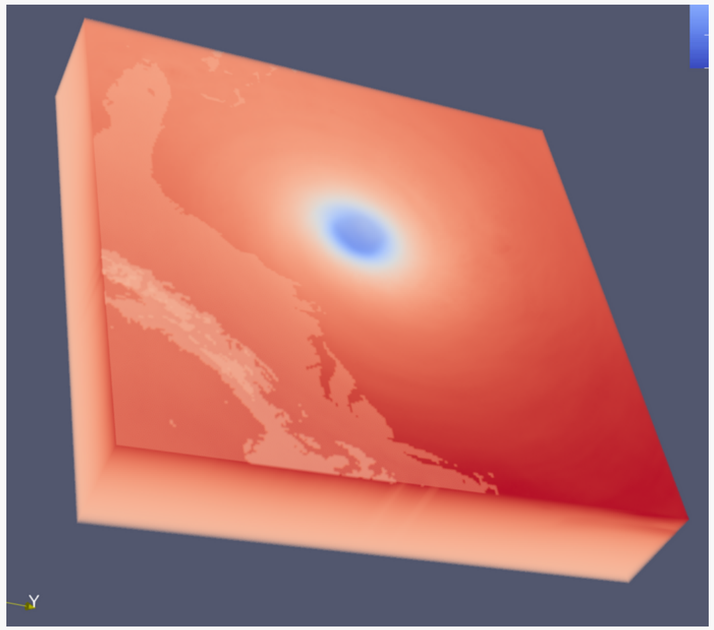
In parallel

- Read data and Scatterv to all nodes
- Calculate gradient at each node
- allreduce the min and max values
- Count frequencies in each bin
- Allreduce to get global histogram
- Create the importance funtion
- Sample points at each node, as coordinates and values
- Gatherv to collect sampled points
- Write the points to a vtp file

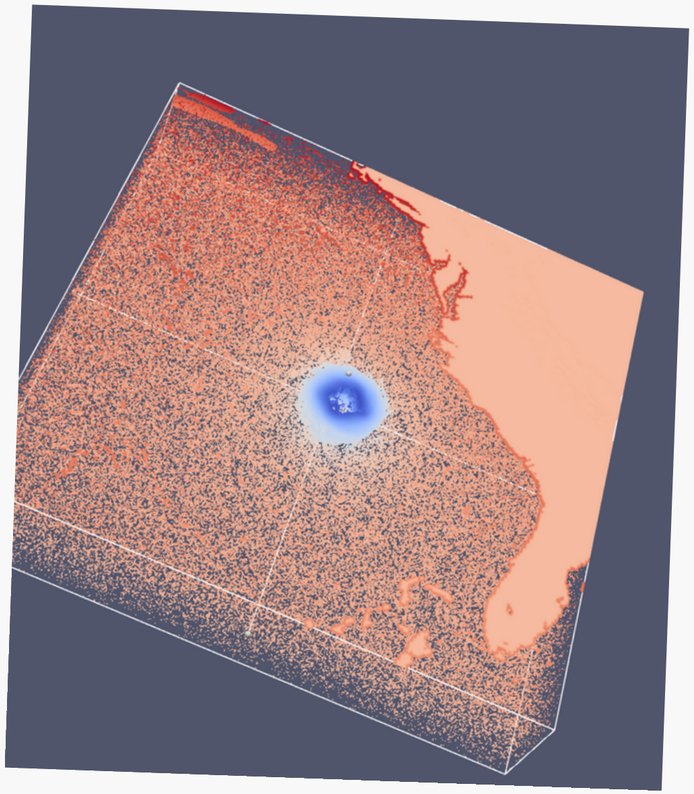
3.1 Visualisation

2% sampled

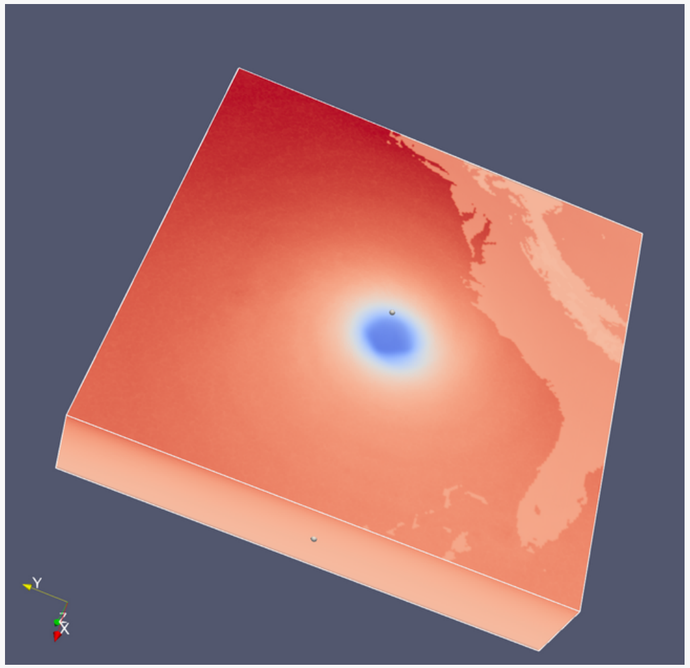
Original dataset



Sampled data in paraview

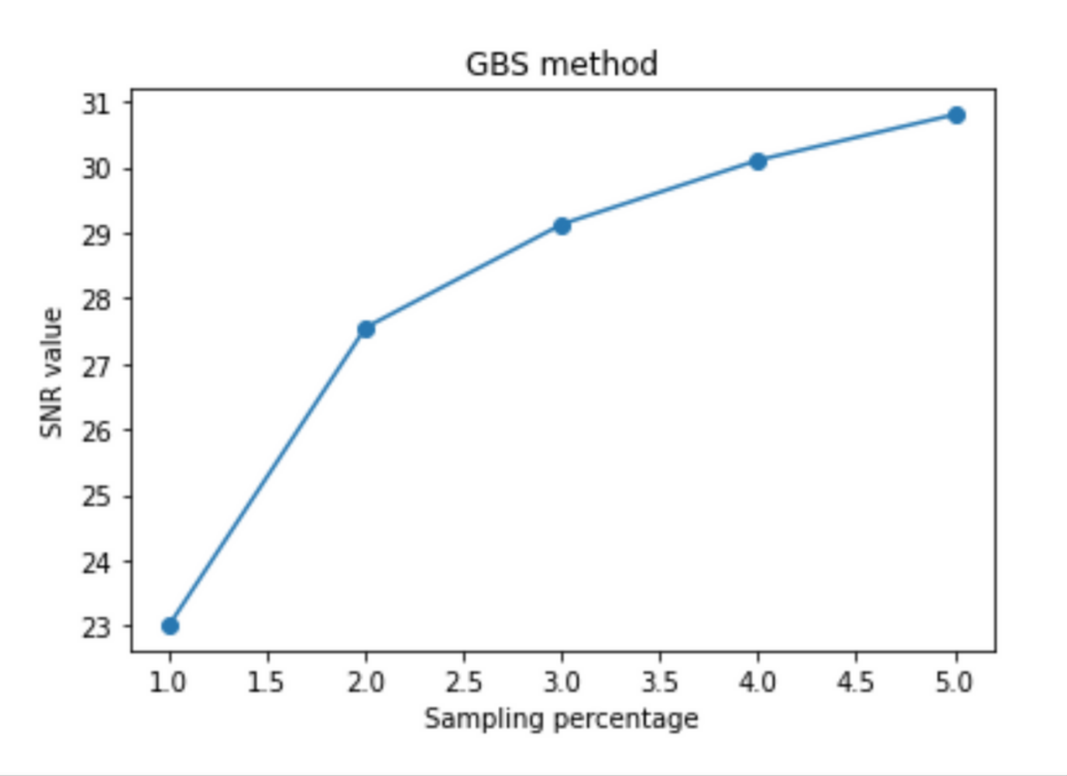


Reconstructed in Paraview

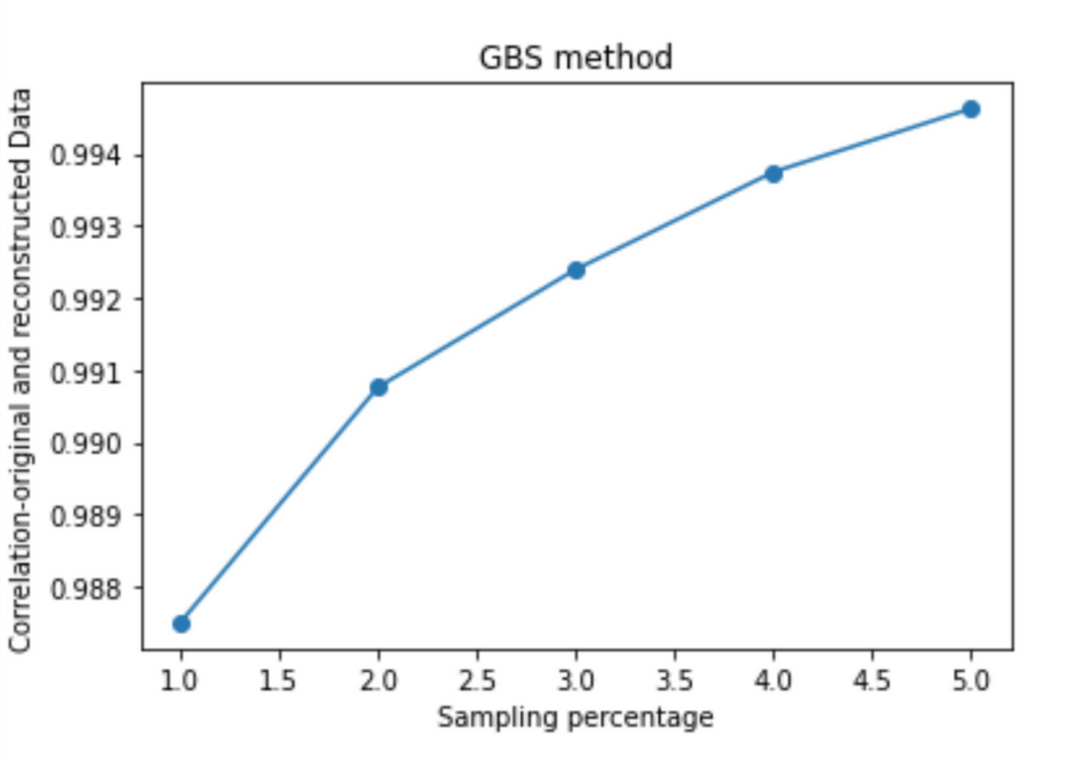


3.2 Evaluation

SNR-sampling rate

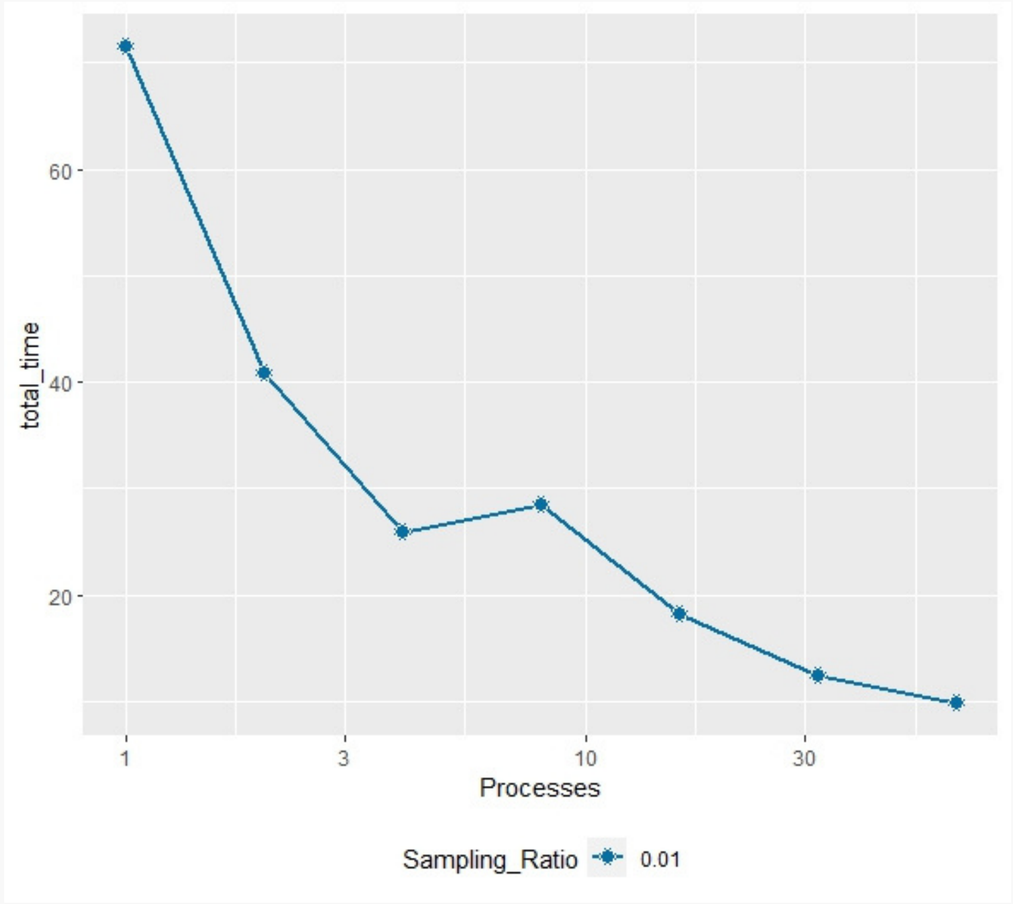


Correlation-sampling rate

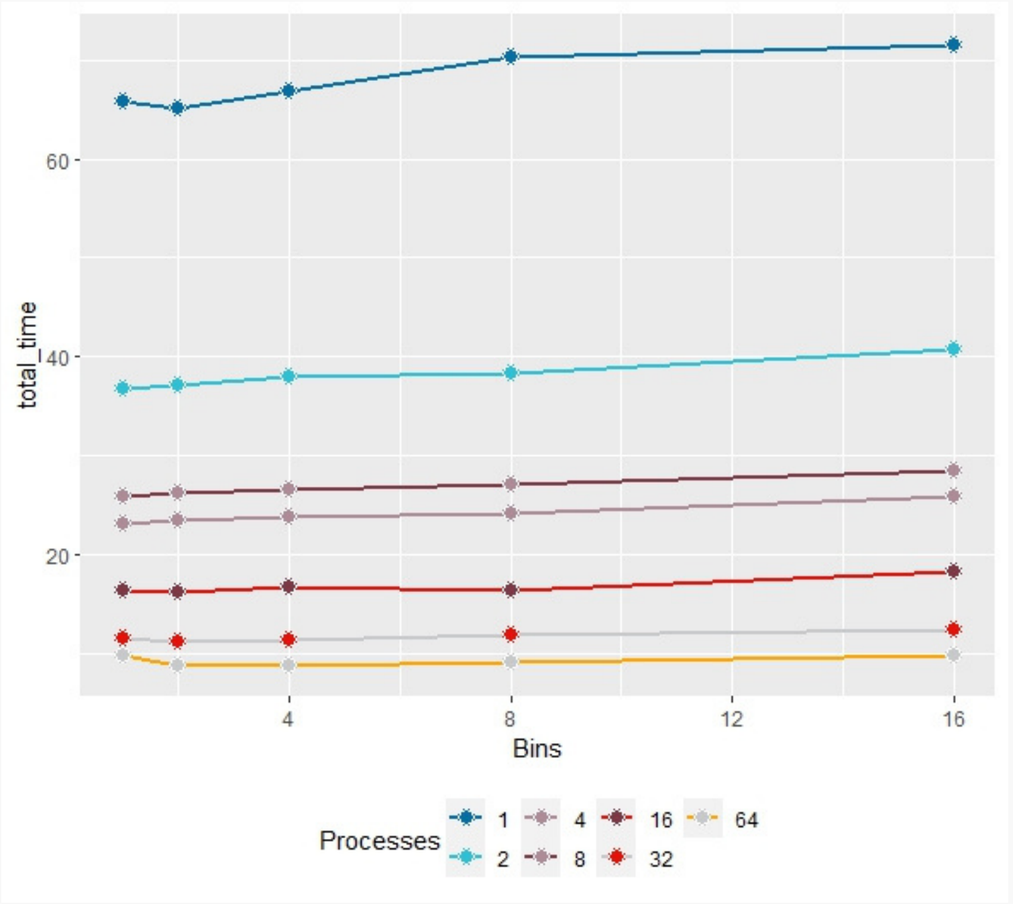


3.3 Scaling

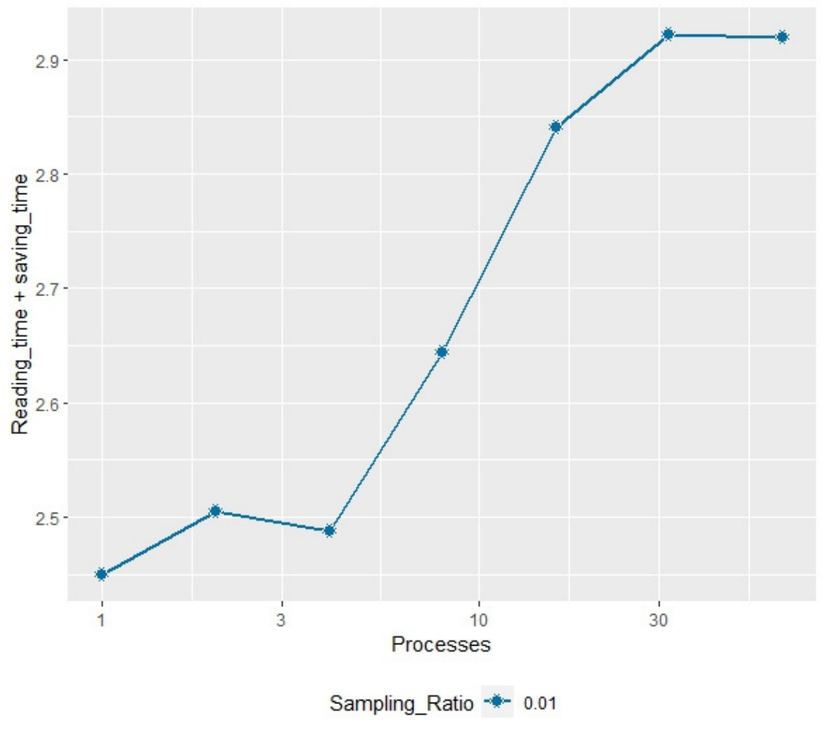
Time-#processes



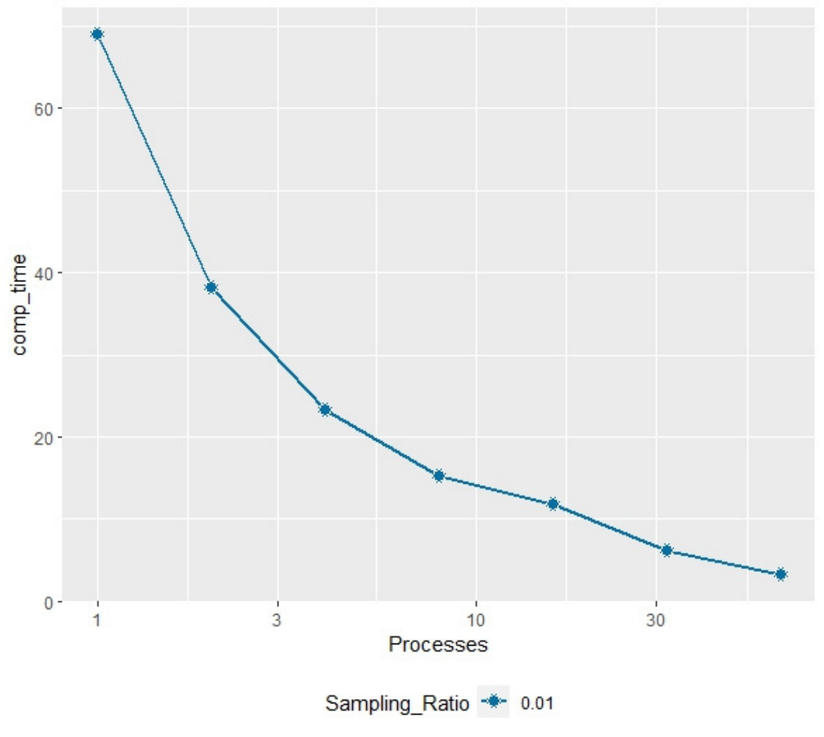
Time-#bins



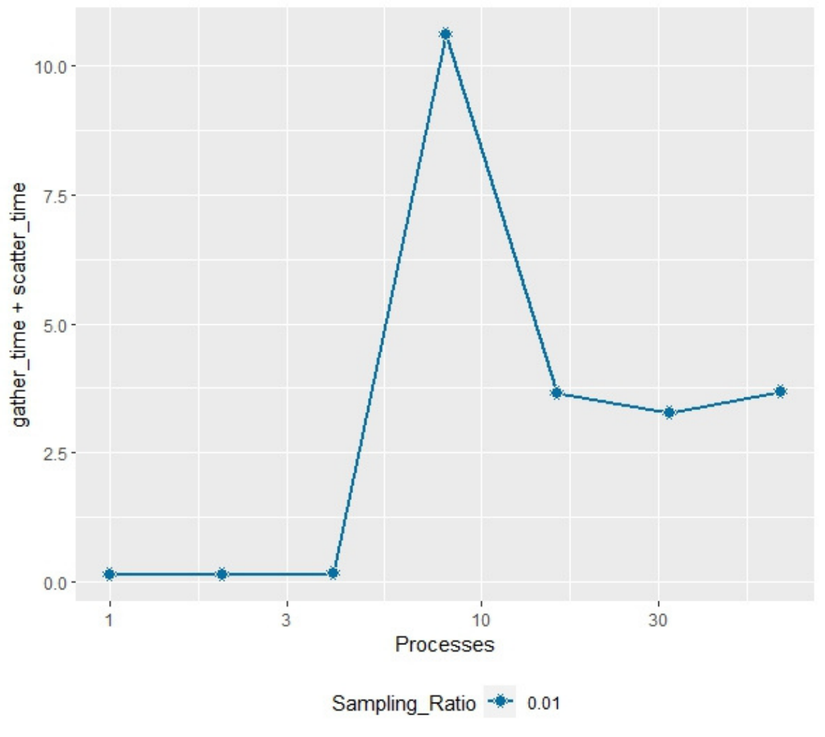
Input/Output - # processors



Computing time - # processors



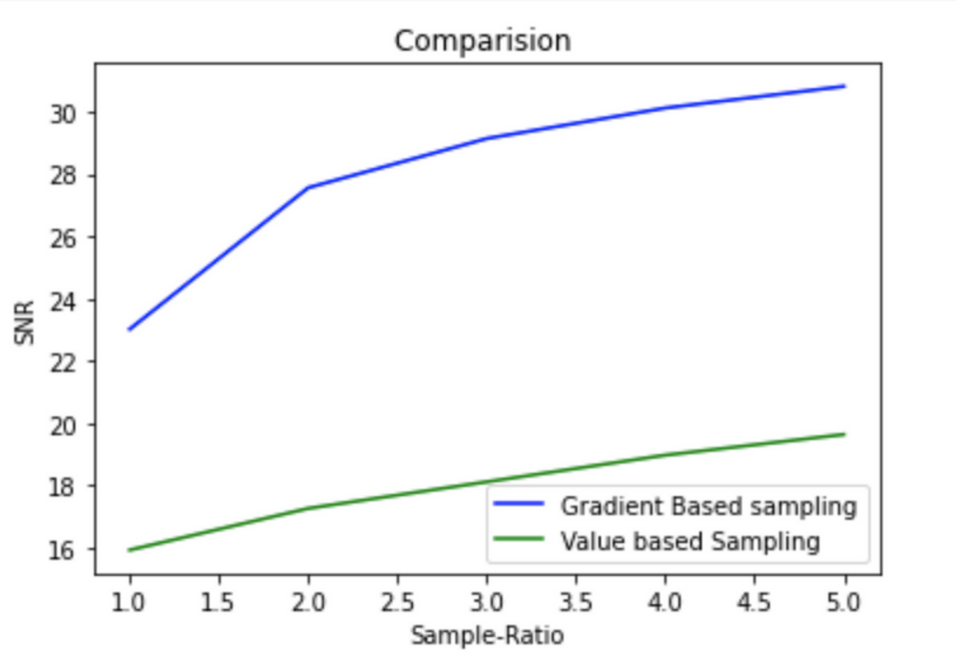
Communication time - # processors



Evaluation

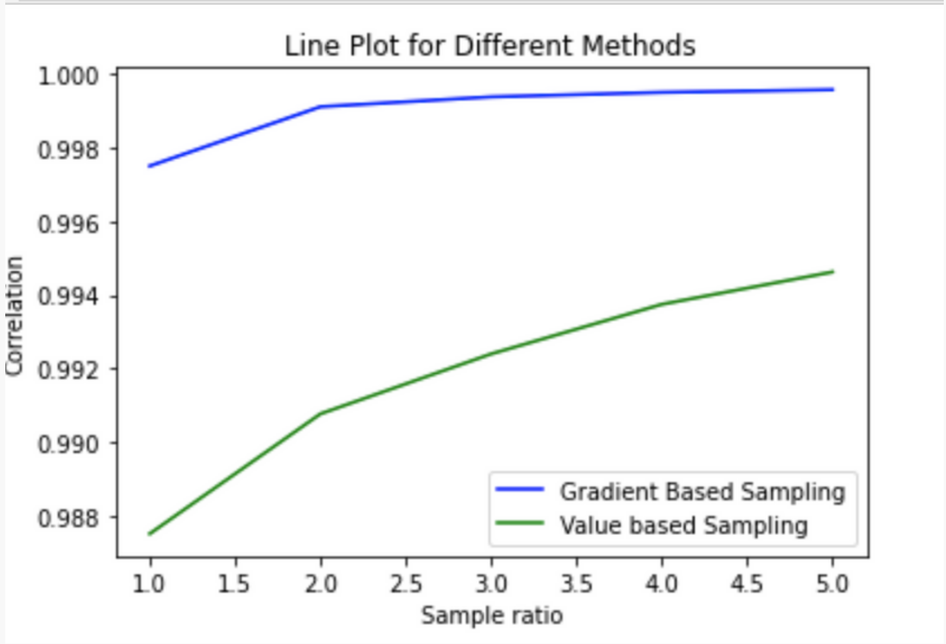
Signal-to-noise ratio (SNR)

- Comparison of SNR across different sampling methods for varying sampling ratios.
- It can be observed that for a given sampling rate, our proposed method generally performs better



Correlation Plot

- Comparison of correlation coefficient across different sampling methods for various sampling ratio



Conclusion

1. We have implemented sampling techniques
   - Simple random sampling
   - Value-based importance sampling
   - Smoothness-based Importance Sampling
2. We have utilized mpi4py for parallelization, employed to distribute the computational workload across multiple nodes or processors
3. We have used linear interpolation based reconstruction-based visualization
4. We have used these evaluation metrics to compare the existing methods to our method
   - Signal to Noise Ratio (SNR)
   - Correlation Coefficient

Conclusion

5. Implementation Tools:
   - Employed a combination of programming languages and libraries suitable for implementing the sampling techniques, mpi4py, and visualization algorithms
   - Ensured compatibility and optimization for parallel computing using mpi4py
6. Objective:
   - The overarching goal was to enhance the efficiency and accuracy of data sampling and visualization through the integration of diverse sampling techniques and parallelization strategies.

Work distribution

Student's Name	Work
Dasari Charithambika	Reconstruction based visualization, Evaluation,Report, PPT
Divya Gupta	Reconstruction based visualization, Evaluation, Report, PPT
Om Shivam Verma	Correction in our VBS Implementation, Parallelisation using MPI, VTK file handling, Report, PPT
Palak Mishra	Implementation of Sampling Algorithms in series, Report, PPT
Siddharth Pathak	Scaling study, Report, PPT

References:

- https://ieeexplore.ieee.org/document/9130956t
- https://mpi4py.readthedocs.io/
- https://examples.vtk.org/site/Python/

THANK YOU SO MUCH!

thank you so much