An optimization-inspired approach to parallel sorting

Team Metropolis:
James Farzi, JJ Lay, and Graham West

COMS 7900, Capstone

Abstract

We present a novel method influenced by ideas in the field optimization to efficiently sort large amounts of data in parallel on a cluster of computing nodes. We describe in depth the different challenges which beset such a method, including distributing/importing files, locally sorting the data on each node, uniformly binning the data, and exchanging data between the nodes. We also present the results of several different timing tests applied to the method. These tests demonstrate how the method scales when the number of files and/or the number of nodes is increased. Finally, we summarize the the greatest challenges in implementing the method, as well as the components of the method which were the most successful. We conclude with a discussion on ways in which the method could be improved.

Contents

1 The Method

Introduction:

We used C++ with C MPI calls
Used GitHub
Workflow description

1.1 File I/O

1.1.1 Distributing files

1.1.2 Importing files

1.2 Sorting

1.2.1 Linked list merge sort

1.2.2 Bubble sort

1.3 Binning

1.3.1 Adapting bin edges

1.3.2 Data binning w/ binary search

1.3.3 Stopping criterion: the uniformity metric

1.4 Exchanging data

1.4.1 Data swap

1.4.2 Cleanup

2 Testing

2.1 Adaptive binning

2.1.1 Prototyping in MATLAB

2.1.2 C++ runs

2.2 Everything Else

3 Conclusions

We will now conclude with two discussions on 1) the most difficult and most successful aspects of our method and 2) ways of improving the both the method's performance/efficiency and our workflow as a group.

3.1 Challenges and successes

3.2 Future work