A HPC Framework for Big Spatial Data Processing and Analytics

Anmol Paudel, Marquette University
Satish Puri, Marquette University

I. Introduction

Recently there has been a huge outburst in the availability of spatial data due to the abundance of GPS enabled devices and satellite imagery. Processing such large volumes of data and running analytics require a lot of time if done sequentially or in a single machine.

Spatial data processing and analytics is a highly data- and compute-intensive task. Doing this in a HPC environment has remained a challenge due to implementation and scalability issues.

In our work, we present a framework for processing and analyzing big spatial data in a HPC environment by overcoming or diminishing the above mentioned issues.

II. Background

Data

How spatial data looks?

1. The data is usually available in text based files like XML or CSV.
2. The data is usually shapes that are represented with points, lines and polygons
3. There is a very huge variability in the size of a single shape since they can be very complex
4. The size of single shape doesn't depend on the area it spans but rather on the number of vertices it has
5. There may be no correlation between the size and spatial distribution of the data

II. Background (contd.)

Implementation

How are existing implementations?

1. Most algorithms in this domain are sequential.
2. Code and libraries for most sequential algorithms already exists.
3. Current system for processing and analytics are sequential.
4. Current systems fail to leverage the full potential of the machine.
5. Efficiently running current implementation on distributed systems is still difficult.

III. Known Problems

What are some of the problems we are tackling?

1. Reading big spatial data is time intensive due to its volume.
2. Splitting the reading of spatial data into parallel is challenging due to its non-uniformity.
3. Spatial data usually requires parsing to spatial datatypes because it is stored in a text like file.
4. There can be large imbalance in the load distribution among nodes due to lack of spatial correlation.
5. Large existing codebase is already sequential so converting it parallel can be extremely costly and time consuming

IV. Proposed Solutions

How we intend to tackle some of the know problems?

1. MPI-Vector-IO: To read spatial data in parallel to a spatial data aware MPI environment
2. ADLB: To handle the issues of load imbalance during compute
3. OpenMP: To maximize the use of each node by spawning threads
4. OpenACC: To enable utilization of GPUs if the nodes have them attached

V. A parallel implementation

Our proposal and aim is to make a new system that uses MPI-IO for data loading, MPI for internode communication, ADLB for load balancing, OpenMP to accelerate in-node computing with threads and OpenACC to utilize GPUs if available.

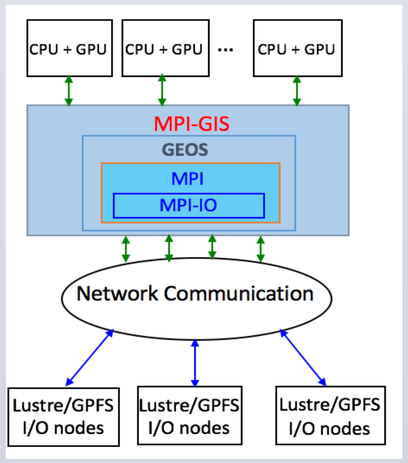

Fig.1. MPI-GIS is built on top of MPI-IO. It uses GEOS open-source library for geometric computations.

VI. Uses

Where will the speedup matter?

1. In forecasting disasters or predicting how they will spread. Here every second is crucial for managing evacuation, recovery or relief efforts.
2. In epidemiology, where early action can save thousands of lives.
3. In managing ground troops, where finding alternative routes in real-time can be mission critical.
4. In locating lost airplanes, where every passing moment makes it even more difficult

In general, in any place where the results are time sensitive and any delay can have huge costs associated with it.

VII. Project Roadmap

How far along are we?

During the first year of PhD research, each of the individual approaches like MPI-Vector-IO, ADLB and directive based computing have been experimented with. Further research into their limitations and overhead in the domain of spatial computing would be required before integrating them into a singular HPC System for Big Spatial Data Processing and Analytics.