2011 Open Science Grid Summer School Final Assignment

Yanli Zhao, July 28th, 2011

With the development of advanced data collection technology, the volume of geographic data that is either explicitly geographic or that can be converted to geographic after it is linked to geographic information (e.g., location), has increased rapidly (Armstrong 2000, Armstrong 2005). For example, LIDAR (Light Detection and Ranging), an optical remote sensing technology, can collect sub-meter resolution elevation data of earth. If the elevation of one point is represented by a float, 4 bytes, there will be up to 3.9 TB data in Utah State, USA. Commensurate increases are also being seen in other types of data record methods such as GPS, individual-level administrative and retail records (e.g., medical information), which are often linked by addresses to locations. Performing simple input-output (I/O) of these massive data files requires a considerable amount of processing time. And it could take even much more when these data resources are used in analyses.

Many types of geographic analysis and models are intrinsically computationally intensive (Armstrong 2000, Armstrong 2005). Some involve combinatorial search strategies, like "traveling salesman problem" that is a spatial-optimization problem and is referred as NP-complete. The computational intractability for such problems has been recognized by geographers for decades. Other types of models such as local clustering analysis to detect statistically significant local-level clusters involve Monte Carlo simulation and statistical methods. These analyses are computationally intensive even for small and moderate size datasets. When they are applied to large geographic datasets on

large spatial and temporal scale, they could require much more computational resources that are often unavailable on a single computer.

Parallel and distributed computing system, i.e. Open Science Grid (OSG), can provide significant computational and storage resources that the data intensive complex geographic models require. It combines computational resources from multiple supercomputing centers, providing an integrated, persistent system for scientific research and education. Authorized users can safely and seamlessly move a large amount of data between different sites fast, divide a complex problem into smaller problems and submit these smaller problems as jobs to OSG, which is responsible for finding the matched resources to run the submitted jobs. By harnessing these resources, a complex or an even intractable problem could be solved in a reasonable time.

In this report, a spatial statistic method, called Gi*(d) statistics (Shaowen Wang 2008), will be researched in the future. The Gi*(d) statistic method is often used to identify local clusters and widely used in Geographic Information System (GIS). As the workflow of this method shown in the Figure 1, there’re 3 loops in the program and thus it is computational intensive. The significant computational intensity could be solved by using the OSG resources.

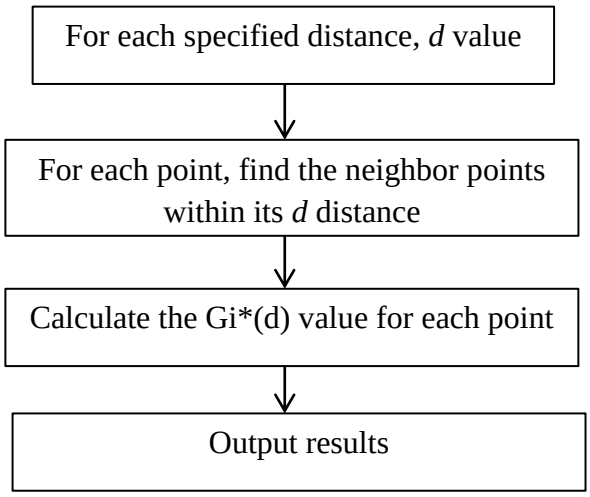

Figure 1. Workflow of Gi*(d) statistics

In the experiments, the input points are produced by random point generator on CyberGIS Gateway (https://gisolve2.cigi.illinois.edu:8443/home/). Each point is recorded as *(x, y, z)*, in which *(x, y)* is the *2D* geographic coordinates of the location, and *z* is the attribute that we are interested in. There are up to *10* billion points calculated, totally *120* GB data if *(x, y, z)* is represented by floats.

Since each point's Gi*(d) value is calculated using the same program, we can decompose the input large dataset into multiple smaller datasets and let each job run on one smaller dataset. If each compute node in the cluster has memory size *M* GB and each job is required to be run in the memory and thus can process at most *M/2* GB input dataset, then there are at least *n*120/(M/2)* jobs needed, where *n* is the number of *d* values. For each job, it processes about *M/2* GB points data and will take approximately *1* hour if *M = 4*. These jobs can be submitted to our local clusters at NCSA super computer center using Condor-G, and the decomposed small datasets are moved to different specified sites using GridFTP. Since the number of small datasets is large, a program will

be written to distribute these datasets automatically using bash and python script languages. Likewise, for job submission, the submitted sites should be from the sites that the datasets are stored, so as to improve the data locality. After finishing all the jobs, the output datasets from all the jobs are combined into a large output dataset for each *d* value, therefore, the DAGMan is used to control the workflow. The workflow of the Gi*(d) statistic on the OSG is illustrated in Figure 2

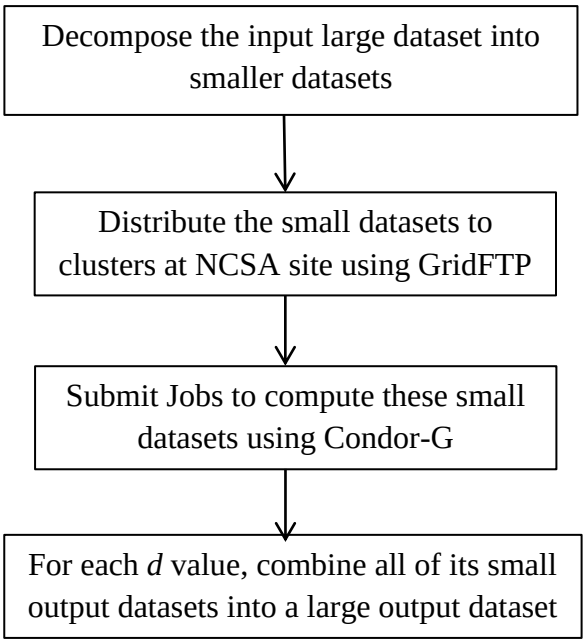

Figure 2. Workflow of the *Gi*(d)* statistic on OSG

If the input dataset is 120GB, and each decomposed small dataset is 2GB. For 10 *d* values, there will be 120/2*10=600 jobs submitted to Grids. For each job, it is estimated to take 1 hour to finish the work. If there're 60 compute nodes used to run these jobs simultaneously, it is expected to finish all the jobs within 15 hours including the input data distribution, jobs synchronization in the last stage and final output results combination. Thus, by using the OSG resources, the Gi*(d) statistic is expected to be

capable of processing the larger datasets and improve the overall computational performances.

Acknowledgement

Thanks a lot for the help from my Ph.D. advisor, Prof. Shaowen Wang (University of Illinois at Urbana-Champaign), OSG summer school mentor, Dr. Tim Cartwright (University of Wisconsin-Madison) and OSG Summer School organizers and Instructors.

References

1. Armstrong, Marc P. (2000). 'Geography and Computational Science', *Annals of the Association of American Geographers*, 90: 1, 146 – 156.
2. Armstrong, M. P., Wang, S., and Cowles, M. (2005). Using a computational grid for geographic information analysis: A Reconnaissance. *Professional Geographer*, 57, pp. 365–375.
3. Wang, S., Cowles, M. K., and Armstrong, M. P. 2008. "Grid Computing of Spatial Statistics: Using the TeraGrid for Gi*(d) Analysis." Concurrency and Computation: Practice and Experience, 20:1697-1720