High-Performance Terrain Analysis and Democratic Algorithms

Richard Barnes

The availability of high-resolution, continent-scale digital elevation models will enable new science. However, analyzing such data requires the development of new algorithms that can handle long-range spatial dependencies. Such algorithms should ideally benefit users with supercomputers as well as users with modest resources. Here, I present evidence that this can be accomplished and highlight possible paths forward.

Digital elevation models (DEMs) are widely used in geospatial terrain analysis for estimating hydrologic and geomorphic properties. These include soil moisture, terrain stability, erosive potential, rainfall retention, and stream power. Knowing these properties aids in the prediction of landslides, flooding, river avulsion, and the geological stability of nuclear waste storage sites. [3, 6]

The size and resolution of the datasets that can be used to make such predictions has been growing. In 2017, the Polar Geospatial Center (PGC) released ArcticDEM6: a DEM covering 14% of the planet at 2 m resolution. [7] The PGC is partnering with Planet Labs to expand the spatial and temporal scope of their dataset. Ultimately, it will be possible to produce high-resolution DEMs for most of the planet on a daily basis.

If each cell in a DEM can be considered independently or as part of a local neighborhood, then computation is embarrassingly parallel and it is possible to process datasets of this size efficiently using existing algorithms and infrastructure, such as Google Earth. However, hydrological calculations cannot be performed in this way. The flow of water from one cell to the next creates long-range interactions that cannot yet be effectively parallelized.

My work shows that it is possible to address some of the challenges in terrain analysis at scale. I have developed effective parallel algorithms for depression-filling and D8 flow accumulation, two core operations of terrain analysis. These permit the processing of datasets 1000 x larger than was previously possible, up to several trillions cells. [1, 2] On a dataset small enough for the industry-standard TauDEM software to process, the depression-filling algorithm ran 6.3 x faster, used 19 x less bandwidth, 70 x less communication time, and 7 x less RAM. I have leveraged GPUs to perform similar operations rapidly on smaller scales, which is useful for landscape evolution modeling. [3]

Challenges remain. Foremost, although my algorithms are efficient, they are not yet general. It is unclear whether similar algorithms exist for other common terrain analysis techniques. "Flow algebras" conceptualize a general approach [8] to this problem, but have not yet been realized in practice. Graph algorithm building blocks (GABB) [5] have been realized and may suffice, but are likely to be *too* general given that many geospatial graphs are embedded-planar while GABB is suitable for any topology.

My approach ensures that, regardless of the DEM's size each DEM cell is accessed a fixed number of times, all compute nodes remain fully utilized, and only a fixed number of low-cost communication events are required. These properties imply that the algorithms can run effectively on laptops or supercomputers.

Such "democratic algorithms" are necessary to ensure that the benefits of a Geospatial Software Institute are realized by all users, not just those who are well-resourced. Algorithmic democracy is important for novel geospatial applications. Cloud environments such as Google Earth enable users to perform analyses that might otherwise be impossible, but can also be traps that prevent users from finishing their work if the necessary algorithms are not available. While open source software is often thought of as a necessary prerequisite to innovation, the theoretical advances which make that software effective and broadly usable encode values which must also be considered.

The GraphBLAS project [4] offers a model of how a Geospatial Software Institute could contribute to solving problems in scalable terrain analysis among other geospatial challenges. By helping coordinate an effort to develop standardized mathematical foundations (the GABB), efficient implementations, and hardware tuned to accelerate these implementations, the GraphBLAS effort achieved many of its goals within five years.

Developing holistic understanding of big raster data is a grand geospatial challenge. My work suggests that this challenge can be met. A Geospatial Software Institute can accelerate innovation in this area coordinating efforts while advocating for solutions that benefit all users of geospatial software.

References

[1] Barnes, R., Nov. 2016. Parallel priority-flood depression filling for trillion cell digital elevation models on desktops or clusters. Computers & Geosciences 96, 56–68. doi: 10.1016/j.cageo.2016.07.001

[2] Barnes, R., Jun. 2017. Parallel non-divergent flow accumulation for trillion cell digital elevation models on desktops or clusters. Environmental Modelling & Software 92, 202–212. URL `http://linkinghub.elsevier.com/retrieve/pii/S1364815216304984` doi: 10.1016/j.envsoft.2017.02.022

[3] Barnes, R., 2018. Accelerating a fluvial incision and landscape evolution model with parallelism. arXiv:1803.02977.

[4] Graph BLAS Forum, 2018. Graph BLAS Forum. `http://graphblas.org/index.php?title=Graph_BLAS_Forum`.

[5] Kepner, J., Gilbert, J., 2011. Graph algorithms in the language of linear algebra. SIAM.

[6] Montgomery, D. R., Dietrich, W. E., Apr. 1994. A physically based model for the topographic control on shallow landsliding. Water Resources Research 30 (4), 1153–1171. URL `http://doi.wiley.com/10.1029/93WR02979` doi: 10.1029/93WR02979

[7] Polar Geospatial Center, 2018. Arcticdem. `https://www.pgc.umn.edu/data/arcticdem/`.

[8] Tarboton, D. G., Baker, M. E., 2008. Towards an algebra for terrain-based flow analysis. Representing, modeling and visualizing the natural environment: innovations in GIS 13, 167–194.