Detecting Anomalies in Very Large Graphs*

Michael M. Wolf† and Benjamin A. Miller

MIT Lincoln Laboratory

Numerous applications focus on the analysis of entities and the connections between them, and such data are naturally represented as graphs. In particular, the detection of a small subset of vertices with anomalous coordinated connectivity is of broad interest, for problems such as detecting strange traffic in a computer network or unknown communities in a social network. These problems become more difficult as the background graph grows larger and noisier and the coordination patterns become more subtle. In this talk, we present a statistical framework addressing this cross-mission challenge and the computational challenges involved when the data sets are very large.

The statistical framework has been developed as part of an effort called Signal Processing for Graphs (SPG), where signal processing concepts are applied to graph data in order to find these anomalies [3]. The SPG framework determines statistically whether, for the observed data, an anomalous subgraph is detected (rejection of the null hypothesis that there is no anomalous subgraph) or not (acceptance of the null hypothesis). Detection algorithms in the SPG framework are based on the concept of graph residuals, formed by subtracting the expected graph data from the observed graph data. The detection framework is designed to detect significant residuals concentrated on a small subset of vertices. The SPG processing chain has several stages including temporal integration of graph data, construction of the expected topology graph, dimensionality reduction, anomalous subgraph detection, and identification of the subgraph's vertices. Dimensionality reduction is a particularly important step, being the most computationally complex, and will be the primary focus of the presentation.

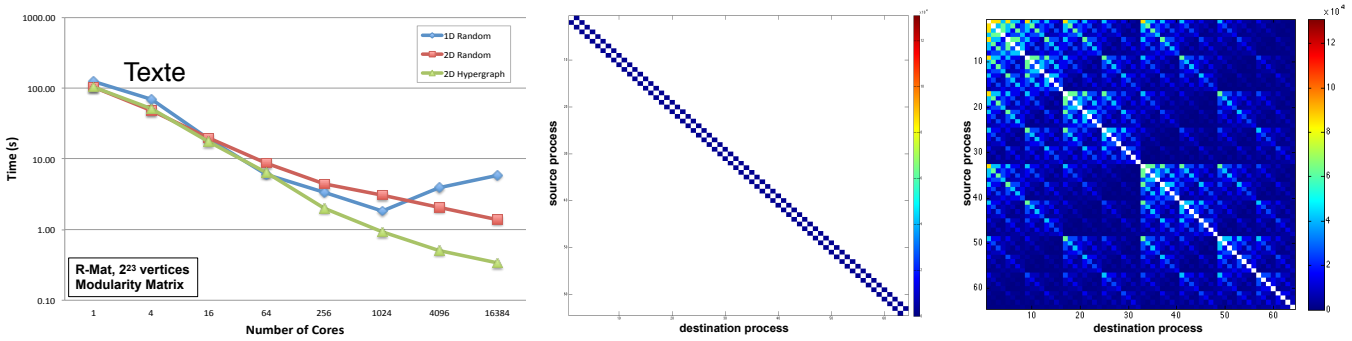
Dimensionality reduction is frequently done by decomposing the residual matrix through eigendecomposition (although SVD would work as well). While relatively strong signals can be detected with only one eigenvector, more powerful detection methods needed to detect more subtle anomalies may require hundreds of eigenvectors. Thus, our initial efforts improve the performance of this step have focused on speeding up the solution of the eigensystem, $Bx_i = \lambda_i x_i$, where $B = A - E[A]$ is the residual matrix. For the purpose of this work, we have focused on the modularity matrix [4], where the expected value is a rank-1 approximation to the observed data, $E[A] = kk^T/(2M)$, where M is the number of edges in the graph and k is the degree vector (assuming undirected edges for simplicity). We use the Anasazi eigensolver [2] to solve our eigensystem, allowing us to find anomalies in very large graphs in a reasonable amount of time. In particular, we use the block Krylov-Schur method to find the eigenvectors corresponding to the eigenvalues with the largest real components (which correspond to the largest residuals). The presentation will demonstrate the use of Anasazi to find eigenpairs in graphs with over four billion vertices.

Figure 1a shows the run time to find the first eigenvalue of a 2^{23}-vertex R-Mat matrix (a=0.5, b=0.125, c=0.125, with an average of 8 nonzeros per row) as the number of cores increases. When a simple one-dimensional distribution (1D, blue curve) is used, the scalability is limited and the run time actually increases for more than 1024 cores. 1D distributions are the de facto standard and tend to work well for traditional computational science and engineering problem. However, for matrices derived from power-law graphs, 1D partitioning tends to result in all-to-all communication in the sparse matrix-vector multiplication (SpMV) operation that dominates the eigensolver run time. Figure 1 shows the communication patterns for 1D distributions of a more traditional finite difference matrix (1b) and the R-Mat matrix (1c). In these illustrations, the color corresponds to amount of data communicated between a pair of processes (white represents no communication, blue represents little communication, and red represents much communication, with the maximum value (red) being set to be the number of rows divided by the number of cores). The 1D distribution works well for the finite difference matrix, with each process communicating with at most two other processes. However, for the R-Mat matrix, each process has to communicate with every other process.

With 1D distributions, solving our eigendecomposition problem for very large power-law graphs is infeasible with our runtime requirements. However, there has been recent work on using two-dimensional (2D) distributions that bound the number of messages to be communicated in the resulting SpMV operation. In Figure 1a, we show the improved results for two of these 2D distributions: one based on a random partitioning of rows/columns with an imposed 2D block Cartesian structure (2DR, red line, [5]) and one that uses hypergraph partitioning (with the Zoltan 1D hypergraph partitioner) instead of random partitioning to improve the communication volume (2DH, green line, [1]). Using these 2D methods, we are able to get more reasonable performance scaling and find eigenvectors for large graphs, with the 2DH method performing particularly well. In this talk, we present these results and further analysis of how these 2D methods can be used in practice.

*This work is sponsored by the Intelligence Advanced Research Projects Activity (IARPA) under Air Force Contract FA8721-05-C-0002. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon. Disclaimer: The views and conclusions contained herein are those of the author and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of IARPA or the U.S. Government.

†Currently at Sandia National Laboratories



(a) Time to find one eigenpair. (b) 9-point 2D finite difference. (c) R-Mat matrix.

Figure 1: Strong scaling results for R-Mat matrix (a). Communication patterns (b,c) for 1D distributions over 64 cores. Row represents source process and column represents the destination process. Color corresponds to amount of data communicated (with maximum value being number of rows divided by number of cores). Jobs run on National Energy Research Scientific Computing Center (NERSC) machine Hopper.

The challenge with 2DH is its relative complexity in computing the distribution. For an 8-million-row R-Mat problem, approximately 40,000 SpMV operations are required to amortize the additional cost of calculating the 2DH distribution rather than 2DR. Since we typically need at most a few thousand SpMV operations in our eigensolver, the 2DH distribution must be effective for multiple observed graphs to be useful. We explored this effectiveness using a simple dynamic graph model in which we partition an initial graph and use our R-Mat generator to add more edges (using the defined distribution) to generate a sequence of additional graphs. Figure 2 shows the runtime (normalized by the number of edges) of the SpMV operations for the 2DR and 2DH distributions applied to several graphs that evolve from an initial graph containing 30% of the edges in the final graph. Although the 2DH distribution loses some of its advantage over the 2DR distribution, it still has a significant advantage at the final graph. Thus, it may be possible to effectively amortize the expensive 2DH distribution over several graphs and use this distribution to our computational advantage. We plan to explore this further for additional dynamic graph models.

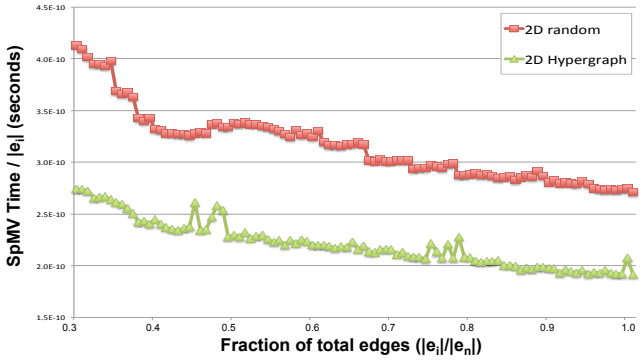


Figure 2: SpMV time in seconds (normalized by number of nonzeros) as graph evolves.

References

[1] E. G. Boman, K. D. Devine, and S. Rajamanickam. Scalable matrix computations on large scale-free graphs using 2D graph partitioning. In *Proc. Supercomputing*, pages 50:1–50:12, New York, NY, USA, 2013. ACM.

[2] C. G. Baker et al. Anasazi software for the numerical solution of large-scale eigenvalue problems. *ACM Trans. Math. Softw.*, 36(3):13(1–23), July 2009.

[3] B. A. Miller, N. T. Bliss, P. J. Wolfe, and M. S. Beard. Detection theory for graphs. *Lincoln Laboratory Journal*, 20(1):10–30, 2013.

[4] M. E. J. Newman. Finding community structure in networks using the eigenvectors of matrices. *Phys. Rev. E*, 74(3), 2006.

[5] A. Yoo, A. H. Baker, R. Pearce, and V. E. Henson. A scalable eigensolver for large scale-free graphs using 2D graph partitioning. In *Proc. Supercomputing*, pages 63:1–63:11, New York, NY, USA, 2011. ACM.