Eric Buras Thesis: Background

February 23, 2016

1   Graph Laplacian Problem

Information about a weighted, undirected graph G with vertex list V and edge list E can be stored in a matrix for computation. This is called the Laplacian which contains edge connectivity and vertex degree information for the graph. For any vertex $u \in V$, the degree of u is:

$$d(u) = \sum_{v \in V} w_{u,v} \ [6]$$

Let D be the diagonal matrix of degree information for V. The adjacency matrix, A of the graph G is defined as:

$$A(u, v) = w_{u,v} \text{ if } (u, v) \in E \text{ and } 0 \text{ otherwise. } [6]$$

The Laplacian of G is:

$$L = D - A \ [6]$$

A common problem is solving a linear system over the graph Laplacian. This is integral in solving problems related to graph regression, spectral graph theory, maximum and minimum cost flow, resistor networks, and partial differential equations. [6] For simplicity, we will refer to the graph Laplacian problem as solving the linear system:

$$(L + I)x = b$$

for x given L and b, a vector of information for the graph. We must shift by the identity because L has row sum equal to 0, thus it is singular. Solutions to this linear system have been studied extensively.

2 Current Solution Approaches

Current Solutions for linear systems can be divided into direct methods and iterative methods. Standard direct methods such as LU or Cholesky factorization are accurate and suitable for graphs with small numbers of edges/vertices, however become useless for graphs of a certain size and complexity. Fast matrix inversion can be applied with order $O(n^{2.376})$ [6], yet this can be improved upon still. In contrast to direct solvers, iterative methods compute better and better approximate solutions to the linear system. A standard iterative method is the Conjugate Gradient method. To speed up these iterative methods, it is possible to introduce a preconditioner that creates an approximation linear system that is much easier to solve than the original. [6]. Yet none of these basic solvers take into account attributes of the graph Laplacian that can drastically improve method performance.

2.1 Spielman-Teng, Koutis et. al.

In many cases, we want to find an approximation to a graph G for easier computing. Spielman and Teng introduce the idea of a spectral sparsifier. The Laplacian for this approximation is similar to the Laplacian for the original graph, thus resulting in a good preconditioner for the original linear system. Multiple cycles of sparsification and factorization combine to solve the original problem. Spielman and Teng (S-T) were thus able to solve symmetric diagonally dominant systems in nearly linear time [7]. This line of work combined with Vaidya's work on subgraph preconditioners resulted in Koutis and Miller's work solving linear systems based on planar Laplacians [2, 9]. Koutis, et. al. were then able to further decrease the time complexity of S-T for general symmetric diagonally dominant systems [3].

2.2 Multigrid Approaches

Algebraic Multigrid utilizes a Galerkin operator in multiple graph coarsening cycles to solve a Laplacian system in linear time. It is mostly used to solve discretized partial differential equations, but has become more popular to solve graph Laplacian systems. AMG is an optimal solver and is parallelizable thus it is useful for solving incredibly large problems [5]. Three current multigrid approaches are combinatorial multigrid from Koutis et. al., Cascadic multigrid from Urschel et. al., and lean algebraic multigrid from Livne and Brandt. All three propose coarsening over the entire graph, but an important question to ask is: how do you coarsen a graph with edges of varying degrees? How do you know which edges or vertices can be aggregated to preserve information?

2.3 Combinatorial Multigrid (CMG)

Koutis, Miller, and Tolliver propose a combinatorial multigrid solver to solve computer vision problems. This method creates a two-level iterative approach

combining the previously mentioned subgraph preconditioning work of Vaidya with algebraic multigrid. For a set of increasingly larger three-dimensional images, CMG required less iterations to converge than a standard multigrid solver in Matlab [4].

2.4   Lean Algebraic Multigrid (LAMG)

Livne and Brandt ran a "lean" multigrid algorithm on graph Laplacian systems for almost 4000 real world graphs of varying size and in vastly different fields from the natural sciences to social networks. Their method has three key parts: first, vertices with low degree are eliminated before the graph is coarsened. Second: they aggregate vertices for the coarsening by a proximity heuristic. And third: they apply an energy correction to the Galerkin operator and combine iterate solutions for more accuracy. They test their algorithm against CMG, and find that it requires slightly more work, however is more robust overall [5].One potential downside of LAMG is the vertex aggregation step of multigrid. How can vertices be evenly aggregated over graphs with uneven degree distribution?

2.5   Cascadic Multigrid

A final alternative form of multigrid was proposed by Urshel et. al. to solve a related problem to the graph Laplacian linear system; they wanted to calculate the Fiedler vector (eigenvector corresponding to the second smallest eigenvalue) of a graph Laplacian. This cascadic multigrid utilizes heavy edge maching to quickly coarsen a graph [8]. It remains to be seen whether this approach will be successful for solving an entire Laplacian linear system accurately.

3   Novel Graph Splitting

I will now mention a different graph decomposition technique that is crucial in my work. In studying 'small world' networks proposed by Watts and Strogatz [10] Chung and Lu propose an algorithm that separates a graph into a locally connected component and a global component. Finding a local subgraph is akin to finding a locally connected portion of a graph. Given integers $k \geq 2$ and $l \geq 2$, a (k, l) locally connected graph will have at least l paths connecting any given two nodes with distinct edges in each path. The length of each path can be at most k edges for this pair. A grid network can be described locally with $k = 3, l = 3$, and $k = 5, l = 9$ and is a good example of how connected these types of graphs are. Given graph G, its maximum locally connected subgraph is the union of all locally connected subgraphs within the entire graph. It is important that this maximum is unique, and can be found through edge deletion [1]. Thus we are able to split a Laplacian matrix into two matrices with information about a locally connected portion of the graph and information about global edges of the graph for further use in solving the Laplacian linear system.

References

[1] Fan Chung and Linyuan Lu. *Complex Networks*, chapter The Small World Phenomenon in Hybrid Power Law Graphs, pages 89–104. Springer Berlin Heidelberg, Berlin, Heidelberg, 2004.

[2] Ioannis Koutis and Gary L Miller. A linear work, o (n 1/6) time, parallel algorithm for solving planar laplacians. In *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 1002–1011. Society for Industrial and Applied Mathematics, 2007.

[3] Ioannis Koutis, Gary L. Miller, and Richard Peng. Approaching optimality for solving SDD systems. *CoRR*, abs/1003.2958, 2010.

[4] Ioannis Koutis, Gary L Miller, and David Tolliver. Combinatorial preconditioners and multilevel solvers for problems in computer vision and image processing. *Computer Vision and Image Understanding*, 115(12):1638–1646, 2011.

[5] Oren E. Livne and Achi Brandt. Lean algebraic multigrid (lamg): Fast graph laplacian linear solver. *SIAM Journal on Scientific Computing*, 34(4):B499–B522, 2012.

[6] Daniel A Spielman. Algorithms, graph theory, and linear equations in laplacian matrices. In *Proceedings of the International Congress of Mathematicians*, volume 4, pages 2698–2722, 2010.

[7] Daniel A. Spielman and Shang-Hua Teng. Spectral sparsification of graphs. *CoRR*, abs/0808.4134, 2008.

[8] John C Urschel, Xiaozhe Hu, Jinchao Xu, and Ludmil T Zikatanov. A cascadic multigrid algorithm for computing the fiedler vector of graph laplacians. *arXiv preprint arXiv:1412.0565*, 2014.

[9] Pravin Vaidya. Solving linear equations with symmetric diagonally dominant matrices by constructing good preconditioners. IMA Workshop on Graph Theory and Sparse Matrix Computation, 1991.

[10] Duncan J Watts and Steven H Strogatz. Collective dynamics of 'small-world' networks. *nature*, 393(6684):440–442, 1998.