CSE 599: Interplay between Convex Optimization and Geometry **Winter 2018**

Lecture 11: Sparse Cholesky Decomposition

Lecturer: Yin Tat Lee

Disclaimer: *Please tell me any mistake you noticed.*

Please read the slide first. In the remaining of the lecture, we discuss how we can permute the variable to minimize the number of non-zeros in the Cholesky decomposition. The ordering of the decomposition can be huge difference on the number of non-zeros. In extreme cases, it can makes the number of non-zeros decreases from n^2 to $O(n)$. In particular, for matrices rises from planar graphs, we know that the number of non-zeros is $O(n \log n)$ under some ordering [77].

Remark. For simplicity, we assume all Cholesky decomposition $A = LL^\top$ does not have numerical cancellation. This makes sure the non-zero locations of L satisfy a certain pattern.

Ideally, we want to solve the following problem:.

Problem. Given a sparse symmetric matrix A, find a permutation matrix P such that the Cholesky decomposition of $P^\top AP$ has as little non-zeros as possible.

Unfortunately, solving this problem exactly is NP-hard in general. The goal of the lecture is to give a polylog approximation algorithm of this problem for graph with $O(1)$ degree. Before this, we note that MATLAB used the following two algorithms to find the permutation matrix:

11.1 Nested Dissection

Definition 11.1.1. For any graph $G = (V, E)$, we call S is a $\frac{1}{2}$-balanced node separator if any connected component of $V - S$ has at most $\frac{n}{2}$ vertices. We call S is an optimal $\frac{1}{2}$-balanced separator if it has the minimum number of vertices.

The nested algorithm is as follows:

1. Find an optimal $\frac{1}{2}$-balanced separator S of G_A. ($\tilde{O}(1)$ approximation also suffices.)
2. Let $A_1, A_2, \cdots, A_k$ are connected components of $G_A - S$.
3. Recurse the algorithm on A_i to find the ordering O_i of the vertices in A_i.
4. Output the ordering $[O_1, O_2, \cdots, O_k, O]$ where O is an arbitrary ordering of S.

Note that this algorithm produces a tree where every vertex is a $\frac{1}{2}$-balanced node separator. We call this tree is a nested dissection separator tree.

By inspecting up-looking algorithm (see the slide) for Cholesky decomposition, we have the following:

Theorem 11.1.2. *Define G_A be the graph with edges $\{(i, j)$ such that $A_{ij} \neq 0$ or $A_{ji} \neq 0\}$. If $A = LL^\top$, then $(i, j) \in G_L$ if and only if there is a path $i, v_1, v_2, \cdots, v_k, j$ in G_A such that $v_t < \min(i, j)$ for $t = 1, 2, \cdots, k$.*

To compare the ordering nested dissection gives with the optimum, we need some handle of the structure of the Cholesky decomposition.

Theorem 11.1.3 ([75])**.** *Let L be any lower triangular matrix arise in $A = LL^\top$. Then G_L has a $\frac{1}{2}$-balanced separator S which induces a clique in G_L.*

Before the proof, we give the key corollary of this result:

Corollary 11.1.4. *Let $A = LL^\top$. Let S_i be the separators at any level of a nested dissection separator tree. Then,*

$$|L| \geq \frac{1}{2} \sum_i |S_i|^2 .$$

Proof of Corollary. Let V_i be the set of vertices S_i is separating. Let G_i be G_L restricted on V_i. Note that G_i is still given by Cholesky decomposition. Theorem 11.1.3 shows that $|G_i| \geq \frac{1}{2} |S_i|^2$. Therefore, we have that $|L| \geq \frac{1}{2} \sum_i |S_i|^2$. □

Proof of Theorem 11.1.3. The proof follows from the following algorithm:

1. $S = \{n\}$.
2. While some component U of $G_{L+L^\top} - S$ has more than $\frac{n}{2}$ vertices
   (a) While some $i \in S$ is not adjacent to any vertex in U
       i. $S \leftarrow S - \{i\}$.
   (b) $S \leftarrow S \cup \{\max_{j \in U}\}$

Note that U is decreasing during the algorithm and that in each outer iteration $\max_{j\in U}$ decreased by 1. Therefore, the algorithm must ends. Hence, it suffices to prove S forms a clique. Clearly, it starts with a clique.

Now, we use induction to prove that S is always a clique. Let j be the maximizer of the current U and i be any element in S.

Due to step a, there is $j' \in U$ that $(j', i) \in G_L$. Hence, Theorem 11.1.2 shows that there is a path $j', \cdots, i$ in G_A with intermediate indices less than j'.

Since j' is connected to j in G_L, Theorem 11.1.2 shows that there is a path $j', \cdots, j$ in G_A with intermediate indices less than j.

Combining two paths, there is a path $j, \cdots, i$ in G_A with intermediate indices less than j (using $j' \leq j$ here).

Now, Theorem 11.1.2 shows that $(j, i) \in G_L$.

Hence, we showed that j is adjacent to any $i \in S$. Therefore, $S \cup \{j\}$ is a clique. □

Now, we are ready to prove the approximation ratio of nested dissection.

Theorem 11.1.5 ([73])**.** *Nested dissection ordering is at most $O(\sqrt{d} \log^2 n)$ times optimal where d is the maximum degree of the graph.*

Proof. Consider the nested dissection separator tree T. (One can think this as the elimination tree but some vertices are collapsed.). For any fixed vertices v, one can prove that $(u, v) \in G_L$ implies u lies on the path from some neighbors n_v of v (in G_A) to v in the tree T.

Let c_v be the number of non-zeros in column v and let T_v be the subtree in T containing v and all neighbors n_v (in G_A). Then, we have that

$$c_v \leq \sum_{S \in T_v} |S|.$$

Let T_l be all separators in level l in T. Then, we have that

$$\begin{aligned}|L| &\leq \sum_l \sum_{v \in T_l} c_v \\ &\leq \sum_l \sum_{S \in T_l} \sum_{v \in S} c_v.\end{aligned}$$

One difficulty of bounding c_v is that some of them belongs to same separator and hence their descendants in T overlap. Hence, for each S, we pick a v_S that maximize c_v. Hence, we have

$$\begin{aligned}|L| &\leq \sum_l \sum_{S \in T_l} |S| c_{v_S} \\ &\leq \sum_l \sum_{S \in T_l} \sum_{S' \in T_{v_S}} |S||S'| \\ &\leq \sum_{l,l'} \sum_{S \in T_l} \sum_{S' \in T_{v_S} \cap T_{l'}} |S||S'| \\ &\leq \sum_{l,l'} \sqrt{d \sum_{S \in T_l} |S|^2} \sqrt{\sum_{S \in T_l} \sum_{S' \in T_{v_S} \cap T_{l'}} |S'|^2}\end{aligned}$$

where we used that $|T_{v_S} \cap T_{l'}| \leq d$ because the leaf of T_{v_S} is the neighbors of v (in G_A) and there are at most d of them. Now, we note that S in the first sum are separated. Similarly for the second sum. Hence, Corollary 11.1.4 shows that

$$|L| \leq 2 \sum_{l,l'} \sqrt{d \text{OPT}} \sqrt{\text{OPT}} = 2 \log_2^2 n \text{OPT}$$

where we used that the depth of T is at most $\log_2 n$. □

In general, computing optimal $\frac{1}{2}$-balanced node separator is NP-hard. Any α-approximation of such would gives a $O(\sqrt{d}\alpha^2 \log^2 n)$ approximate algorithm for fill-in.

11.2 Open Problems

In practice, MATLAB uses two algorithms.

If the matrix A is very sparse, they use a greedy method. One can think it is a left-looking algorithm and each step choose the new variable that minimize the number of fill in it added in that iteration. That ordering is called minimum-degree. Unfortunately, that algorithm was still too expensive and hence they find a approximate number of fill in instead. It is only known very recently (2017) that minimum-degree ordering can be found in nearly linear time [74]. It would be an interesting to know if that works well in practice compared to MATLAB.

Otherwise, they use nested dissection. Instead of solving SDP to find the $\frac{1}{2}$-balanced node separator (the current best approximation algorithm), they use a multi-level local search to find the $\frac{1}{2}$-balanced separator, called METIS [76]. It seems that the algorithm has not changed too much after these 2 decades despite our knowledge on spectral graph theory and balanced node separator. For example, we did not know the use of SDP on separator at that time and we thought LP relaxation and SDP relaxation is very expensive at that time.

As a purely theory question, we may want to answer this first:

Problem 11.2.1. Can we get a $\tilde{O}(1)$ approximation ratio on fill-in without the d dependence?

One idea here is to avoid the degree problem by splitting vertices and edges. This is a general technique to handle degree in general and it in fact works well in practice to handle dense rows/columns in interior point method. Speaking of this, one may ask

Problem 11.2.2. What is the right extended formulation for this problem?

Problem 11.2.3. Another problem is to remove $\log n$ terms by looking at the problem of fill-in directly or understanding the problem of finding hierarchy of separators directly.

Problem 11.2.4. Given our knowledge on spectral graph theory, do we have a better node separator algorithm in practice than METIS?

These are problems with potential huge impacts. When the Cholesky decomposition is used in convex optimization, usually we reuse the ordering that tens or hundreds of time because the ordering does not care about numerical value of the problem. Therefore, it makes sense to design a more expensive algorithm. Finally, we have this open-ended problem:

Problem 11.2.5. What is the non-linear version of Cholesky decomposition?

References

[73] Ajit Agrawal, Philip Klein, and R Ravi. Cutting down on fill using nested dissection: provably good elimination orderings. In *Graph Theory and Sparse Matrix Computation*, pages 31–55. Springer, 1993.

[74] Matthew Fahrbach, Gary L Miller, Richard Peng, Saurabh Sawlani, Junxing Wang, and Shen Chen Xu. On computing min-degree elimination orderings. *arXiv preprint arXiv:1711.08446*, 2017.

[75] John R Gilbert, Donald J Rose, and Anders Edenbrandt. A separator theorem for chordal graphs. *SIAM Journal on Algebraic Discrete Methods*, 5(3):306–313, 1984.

[76] George Karypis and Vipin Kumar. Metis–unstructured graph partitioning and sparse matrix ordering system, version 2.0. 1995.

[77] Richard J Lipton, Donald J Rose, and Robert Endre Tarjan. Generalized nested dissection. *SIAM journal on numerical analysis*, 16(2):346–358, 1979.