Feedback Vertex Set

Uriel Feige

Lectures 1 and 2, March 27 - April 3, 2019

Abstract

These are note that I wrote for my own use, and are rather sketchy.

1 Introduction

Feedback vertex set (FVS) is the problem of hitting all cycles of a vertex weighted graph by vertices. Namely, the input is a graph $G(V, E)$ with nonnegative vertex weights $w : V \to R^+$, and the goal is to select a minimum weight set $F \subset V$ whose removal from the graph makes the remaining graph a forest.

The related problem feedback edge set (in undirected graphs), in which edges rather than vertices have weights, and the goal is to pick a minimum set of edges that hits all cycles, can be solved in polynomial time. Use a greedy algorithm to find a maximum weight forest (or spanning tree, if the graph is connected), and the remaining edges form the solution.

Another related problem is that of **triangle hitting by edges**, which was a question in last semester's take home exam. For it, the following is known.

- Factor 3 approximation using local ratio.
- Factor 2 algorithm based on LP relaxation. The analysis used duality and complementary slackness, together with fact that in every graph there is a solution of no more than half the total weight (because bipartite graphs have no triangles).
- Factor 2 hardness, unless Vertex Cover (VC) can be approximated within a ratio better than 2.

We return now to FVS. It is not even immediately clear how to get an $O(\log n)$ approximation ratio. For the local ratio technique, the problem is that there may be cycles larger than $\log n$. For randomized rounding of an LP relaxation (even though there are exponentially many cycles, one can use the ellipsoid algorithm to solve a natural LP relaxation of the problem), the standard analysis based on the union bound does not give much, because the number of cycles is in general exponential in n, rather than polynomial.

2 Logarithmic approximation

We present an $O(\log n)$ approximation algorithm based on the local ratio approach. The same algorithm can be viewed as a primal dual algorithm, as done in Chapter 7.2 in [4]. (There are other small differences in the way we present the algorithm compared to [4].)

In general, the ratio between what the primal and dual pay for a cycle is upper bounded by the number of cycle vertices that end up in the solution. In order to get a good ratio,the problem that needs to be addressed is that of long cycles. For this we introduce the following **cleaning phase** that performs the following operations as long as possible:

- Remove vertices of degree less than 2.
- Contract induced paths into a single vertex of minimum weight on the path (other vertices will not be in the solution).
- Parallel edges (cycle of length 2): reduce the weight of both vertices at the same rate until one (or both) of them has weight 0. The local ratio of this step is 2, because every solution needs to contain at least one of the vertices.
- Include in F all vertices of weight 0.

Lemma 1 *After the cleaning phase ends, either the graph is a forest, or the shortest cycle has length at most* $4 \log n$.

Proof. It the graph is not a forest, start BFS from an arbitrary vertex. As long as no cycles are discovered, in every two levels the number of vertices doubles, because there are no two adjacent degree 2 vertices. Hence this can go on for at most $2 \log n$ steps. ∎

Hence we have the following algorithm. Repeat as long as possible:

- Perform a cleaning phase whenever possible.
- Find the shortest cycle. Subtract the weight of the lowest weight cycle vertices from all cycle vertices. (Hence now the cleaning phase can be resumed.)

Analysis by local ratio. Alternatively, the weight reduced from the shortest cycle is the weight given to the dual variable (the cycle), and the primal solution pays at most $4 \log n$ times dual.

Having established an $O(\log n)$ approximation ratio, we ask whether we can do better.

A difficulty arises because the natural covering LP has an $\Omega(\log n)$ integrality gap. Consider a d-regular (unweighted) *Ramanujan graph.* Its girth (length of shortest cycle) is $\Omega(\log n)$. Hence the LP has a fractional solution of weight

$O(\frac{n}{\log n})$. The minimum FVS has cost $\Omega(n)$, because every subgraph with $\frac{3n}{4}$ vertices has at least $\frac{dn}{2} - \frac{dn}{4} = \frac{dn}{4} \geq n$ edges, and hence has a cycle.

The above analysis of the Ramanujan graph shows that the standard covering LP misses constraints that give lower bounds on optimal solution (in the whole graph and in various subgraphs) in terms of the number of edges. One can incorporate such constraints in the LP and improve the approximation ratio to 2. This is done in Chapter 14.2 of [4], were a primal dual algorithm that is based on the stronger LP is presented. However, we prefer to show here a local ratio algorithm.

3 A factor 2 approximation

The algorithm and analysis presented here are taken from [1]. The first hint that a factor 2 approximation may be possible comes from the following proposition.

Proposition 2 *On unweighted d-regular graphs,*

- *FVS can be approximated within a ratio not worse than 2.*
- *Moreover, every minimal solution (one for which the removal of any vertex from it makes the solution infeasible) obtains this approximation ratio.*

Proof. The proposition is clearly true for $d = 2$. We prove it for $d \geq 3$.

The optimal solution F^* satisfies $\frac{nd}{2} - d|F^*| < n - k$, implying

$$|F^*| > n(\frac{1}{2} - \frac{1}{2d-2}).$$

Any minimal solution $|F|$ satisfies $2|F| \leq (n-k)(d-1)$ (because every vertex in FVS has at least two edges into non-isolated vertices in the remaining graph). This implies

$$|F| \leq n(1 - \frac{2}{1+d}).$$

For $d \geq 3$ the approximation ratio is better than 2. ■

The above proposition gives hope that a factor of 2 is achievable also in the general case. What is the appropriate statement for general weighted graphs? What relation between weight and degree insures factor 2 approximation?

Under the condition that the graph is *clean* (passed a cleaning phase), we have such a relation. We shall call a cycle C in a graph G *independent* if all vertices of C have degree 2 in G, and *semi-independent* if all vertices of C but one have degree 2 in G. Note that after the cleaning phase, G does not have any independent or semi-independent cycles.

Lemma 3 *Let G be a* **clean** *graph in which for every vertex, $w(v) = d(v) - 1$.*

- *Then FVS can be approximated within a ratio not worse than 2.*

- *Moreover, every minimal solution obtains this.*

Proof. Let us define:

$$p(v) = \frac{d(v)}{2} - 1.$$

The optimal solution F^* satisfies $m - \sum_{v \in F^*} d(v) < n - |F^*|$, implying

$$w(F^*) > m - n = \sum_{v} (\frac{d_v}{2} - 1) = \sum_{v \in V} p(v) = p(V).$$

Any minimal solution F satisfies $2|F| \leq w(V - F) = 2m - n - w(F)$ implying

$$w(F) \leq 2m - n - 2|F|.$$

This proves the lemma when $|F| \geq \frac{n}{2}$, but not in general. (We do get partial results. As the graph is clean, contracting degree 2 vertices we have that $m \geq \frac{3n}{2}$, and after putting them back in we have $m \geq \frac{5n}{6}$. This gives a ratio no worse than 11.)

We now improve over the above analysis. Consider a minimal solution F. Each vertex $v \in F$ is *blocked* by two edges into a connected component of $V - F$ (as removing v from F will then close a cycle). We refer to the corresponding two edges as the blocking edges for v (if there is more than one way of choosing the two blocking edges – we pick one such pair arbitrarily).

Consider any tree T in $V - F$ with t vertices that blocks a vertex of F. A key observation is that $E(T, F) \geq 3$. This can be seen as follows. Every vertex of T has degree at least 2 in G. If the tree has at least 3 leaves, then $E(T, F) \geq 3$. If it is an isolated vertex then together with the blocked vertex we have two parallel edges. If T is a path, then either $E(T, F) \geq 3$ or G has two adjacent vertices whose sum of degrees is 4. All options except for $E(T, F) \geq 3$ are excluded because G is clean.

We also have

$$E(T, F) = \sum_{v \in T} d_v - 2(|T| - 1) = 2p(T) + 2.$$

Let $F(T)$ be the set of vertices that are blocked by T. Then:

$$|F(T)| \leq \lfloor \frac{E(T, F)}{2} \rfloor \leq E(T, F) - 2 = 2p(T).$$

The second inequality holds because $E(T, F) \geq 3$.

As all vertices of F are blocked, we have that $|F| \leq 2 \sum_T p(T) \leq 2p(V - F)$.

Combining the above we have:

$$w(F) = 2p(F) + |F| \leq 2p(F) + 2p(V - F) = 2p(V) \leq 2w(F^*).$$

■

Turning the Lemma 3 into a local ratio algorithm requires an extra idea beyond those that we have seen so far. The third step of the following algorithm is introduced because the lemma guarantees a factor 2 approximation only for minimal solutions. Had the guarantee been given for all solutions (or in other words, if we had $w(F^*) \geq \frac{W(V)}{2}$), then step 3 would not be needed.

1. *Cleaning step:* Whenever possible, clean G.
2. *Reduce step:* When cleaning is not possible, reduce $\gamma(d_{v_i} - 1)$ from the weight of every vertex v_i, for largest γ that does not create negative weights. Take weight 0 vertices into F.
3. *Reverse removal:* When the graph becomes empty, process F in reverse order. From every group of edges that was added at a step, keep a **minimal** set (minimal, not minimum, and not just greedy as greedy need not produce a minimal FVS) that preserves an FVS. Call the resulting solution B.

It is not difficult to see that F is a FVS, but in need not provide a factor 2 approximation, because for graphs G_i at the beginning of reduce steps, $F \cap G_i$ need not be a minimal FVS for G_i.

Lemma 4 *Reverse removal ensures that on each step we do not pay a local ratio greater than 2. Moreover, B is a feedback vertex set.*

Proof. Let F be the FVS found by the algorithm in its forward phase, and let $B \subset F$ be the set that remains after the backward phase. Let G_i (with $G_1 = G$) be the sequence of graphs encountered in the beginnings of iterations. For every vertex $v \in B$, we can trace how its weight $w(v)$ decreases from $w_1(v) = w(v)$ to 0 along the different iterations, where iteration i is charged for the value $w_i(v) - w_{i+1}(v)$.

We prove by backward induction that for every i, $B \cap G_i$ is a minimal FVS for G_i. Hence the associated step is charged with a local ratio no worse than 2 (compared to whatever the optimal solution is charged at the same step).

For G_t that was followed by a cleaning step, we do not pay a local ratio larger than 2. Paying a factor of 2 may happen only if the cleaning step included handling two parallel edges such that both vertices in their end points have degree larger than 2. (In fact, such cleaning steps are not needed for the proof of Lemma 3. They are only needed if one of the endpoints has degree 2, and then reverse removal ensures that from this semi-independent cycle we take only one vertex and not two. Hence cleaning can be changed so that the local ratio for cleaning steps is reduced to 1.)

Consider now G_t that was followed by a reduce step in which the weight of some vertices V_0 were reduced to 0, giving a graph G_{t+1} on the remaining vertices, and adding V_0 to F. By induction, the vertices of $B \cap G_{t+1}$ are a minimal FVS for G_{t+1}. In G_t, the only cycles that V_0 needs to cover are those that involve at least one vertex of V_0. V_0 cannot contribute towards covering any of the cycles in G_{t+1}, hence they cannot make any vertex of $B \cap G_{t+1}$ redundant. As every vertex of V_0 that is redundant is removed from F we have that $B \cap G_t$ is a minimal FVS for G_t. Observe that future backward steps cannot add vertices of G_t to the F (because only vertices of weight 0 are put into F). ■

The approximation ratio of 2 proved for FVS is optimal, under standard assumptions.

Proposition 5 *The approximation ratio for FVS is not better than that of VC.*

Proof. Reduction from VC: add to every edge $e = (i, j)$ a distinct auxiliary vertex v_e and the edges (i, v_e), (j, v_e). ■

It is interesting to note that in (unweighted?) graphs of maximum degree 3, FVS can be solved in polynomial time [3] (by reduction to the Matroid parity problem), whereas VC is NP-hard.

Related question: comes in four versions. Find smallest number of edges (or vertices) that hit all cycles of length exactly (or at most) k.

If $k = n$, the exact version is not approximable as it implies deciding Hamiltonicity.

For constant k, strong hardness results appear in [2]. Still, for the edge version and even k, it is not known if an approximation better than k is possible, and for any k, potentially a ratio of $k/2$ is possible.

References

[1] Vineet Bafna, Piotr Berman, Toshihiro Fujito: A 2-Approximation Algorithm for the Undirected Feedback Vertex Set Problem. SIAM J. Discrete Math. 12(3): 289–297 (1999).

[2] Venkatesan Guruswami, Euiwoong Lee: Inapproximability of Feedback Vertex Set for Bounded Length Cycles. Electronic Colloquium on Computational Complexity (ECCC) 21: 6 (2014)

[3] Shuichi Ueno, Yoji Kajitani, Shin'ya Gotoh: On the nonseparating independent set problem and feedback set problem for graphs with no vertex degree exceeding three. Discrete Mathematics 72(1-3): 355–360 (1988).

[4] David P. Williamson, David B. Shmoys: The Design of Approximation Algorithms. Cambridge University Press 2011. https://www-cambridge-org.ezproxy.weizmann.ac.il/core/books/design-of-approximation-algorithms/88E0AEAEFF2382681A103EEA572B83C6