Lecture Outline 14
Topics in Graph Algorithms (CSCI5320-19S)

CAI Leizhen
Department of Computer Science and Engineering
The Chinese University of Hong Kong
lcai@cse.cuhk.edu.hk

April 12, 2019

Keywords: Vertex covers, indirect certificates, indirect certificating method, random partition, and random selection.

L. Cai, Vertex Covers Revisited: Indirect Certificates and FPT Algorithms, arXiv:1807.11339, 2018.

1 Indirect Certificating Method

Let G be a graph with a k-vertex cover and $N(M)$ denote the open neighbourhood of marked vertices. The following extremely simple algorithm for G returns a k-vertex cover $N(M)$ with probability at least 4^{-k}.

Randomly mark each vertex with probability 1/2 and output $N(M)$.

For the graph in Figure 1, the single dark vertex can be used as an indirect certificate to certify that the graph has an 8-vertex cover. And the existence of such small indirect certificates has an intimate connection with the above algorithm.

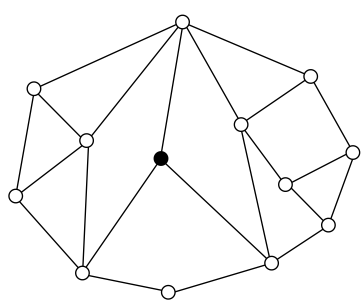

Figure 1: A graph with one marked vertex as a certificate.

The random separation method of Cai, Chan and Chan [1] is a special case of a more general idea of *indirect certificating* that uses small indirect certificates as helpers to find k-solutions. For two disjoint subsets χ and X of vertices V, a partition $(V', V - V')$ of V is a *valid* (χ, X)*-partition* if $\chi \subseteq V'$ and $X \subseteq V - V'$.

> *Discover a small-size structure χ that is disjoint from a k-solution X and can be used as a certificate for a yes-instance. Then use random partition to produce a valid (χ, X)-partition, and find a k-solution with the aid of χ.*

Two examples in Cai, Chan and Chan [1], including the problem of finding a maximum-weight independent set of size k in planar graphs, implicitly used this general idea. There have been a few FPT algorithms based on this idea of indirect certificating: Cygan et al. [3] have designed an FPT algorithm for obtaining an Eulerian graph by deleting at most k edges, and Cai and Ye [2] have presented FPT algorithms for finding two edge-disjoint (s, t)-paths with some length constraints.

1.1 Small indirect certificates for vertex covers

Instead of using a natural certificate X, i.e., a k-vertex cover, to certify a yes-instance of Vertex Cover[k], we can also use an indirect certificate χ disjoint from X, where $N(\chi)$ is the open neighbourhood of χ.

Lemma 1.1 [Cai 2018] *For any minimal vertex cover X of a graph $G = (V, E)$, G contains at most $|X|$ vertices $\chi \subseteq V - X$ such that $N(\chi) = X$.*

Proof. Since X is a minimal vertex cover, every vertex in X covers at least one edge in cut $[X, V - X]$. For each vertex $v \in X$, arbitrarily choose an adjacent vertex $v^* \in V - X$ and let $\chi = \{v^* : \ v \in X\ \}$. Then χ clearly has the property in the lemma as $V - X$ is an independent set. ∎

Application 1: You want to invite friends over for a large fun party. To not let anyone spoil the party, you invite at most one of two persons if they cannot get along with each other. The problem can be formulated as the maximum independent set problem in a graph G. Suppose that there are 100 candidates, and G has an independent set of size 90. Then a list of at most 10 guests will suffice for you to determine the whole list of 90 guests.

Application 2: Consider the problem of destroying conflict pairs in an edge coloured graph G by removing at most k edges, where a *conflict pair* is a pair of adjacent edges with different colours. The *edge-conflict graph* of G, denoted $X(G)$, is an uncoloured graph where each vertex represents an edge of G and each edge corresponds to a conflict pair in G. Then the problem on G is equivalent to Vertex Cover[k] on $X(G)$, and hence has an indirect certificate χ consisting of at most k edges.

Application 3: Lemma 1.1 lays the foundation of an FPT algorithm for VERTEX COVER[k] based on random partition. We randomly and independently colour each vertex by either red or blue with equal probability to form a random partition $\{V_r, V_b\}$ of vertices of G, where V_r and V_b respectively are red and blue vertices. There are two key points to note:

- A random red-blue colouring has probability at least 4^{-k} to produce a valid (χ, X)-partition.
- Once we have a valid (χ, X)-partition, the neighbourhood of red vertices yields a required vertex cover.

The following is the algorithm at the beginning in the language of red-blue colouring.

Algorithm VC-IC[k]
Input: A graph $G = (V, E)$.
Output: A k-vertex cover X of G, if it exists.

1. Randomly and independently colour each vertex red or blue with probability 1/2 to generate a random vertex-partition (V_r, V_b) of G.
2. Output $N(V_r)$ as X.

Theorem 1.2 [Cai 2018] *Let G be a graph that admits a k-vertex covers.* Algorithm VC-IC[k] *finds a k-vertex cover of G with probability at least 4^{-k} in $O(kn)$ time, and hence finds a k-vertex cover of G in $O(4^k kn)$ expected time.*

The algorithm can be made into a deterministic FPT algorithm with running time $4^k k^{O(\log k)} n \log n$ by a family of $(n, 2k)$-universal sets for derandomization.

HITTING SET[k]: Cai has shown recently that the indirect certificating method works when every set contains at most t elements, which implies that the method works for k-vertex deletion to obtain H-free graphs for any fixed H.

VERTEX BIPARTIZATION[k]: Does G contain at most k vertices X such that $G - X$ is bipartite?

The problem admits indirect certificates of size at most $2k$: Let X be a minimal solution and (L, R) a bipartition of graph $G - X$. Then every vertex $v \in X$ has a neighbour v_L in L and a neighbour v_R in R, and $\chi = \{(v_L, v_R) : v \in X\}$ is an indirect certificate with at most $2k$ vertices. However we don't know how to use χ to obtain FPT algorithms.

1.2 Smaller indirect certificates for vertex covers

The following is a general property regarding a small number of vertices in connection with minimum vertex covers to lay the foundation of certificates of size $k/3$ for Vertex Cover. For any integer $d \geq 0$, let V_d denote the set of vertices of degree at least d.

Lemma 1.3 [Cai 2018] *Every graph G with a k-vertex cover contains at most k/d vertices χ, where d is any positive integer, such that for every minimum vertex cover X^* of $G - (N[\chi] \cup V_d(G - N[\chi]))$, vertices $N(\chi) \cup V_d(G - N[\chi]) \cup X^*$ is a minimum vertex cover of G.*

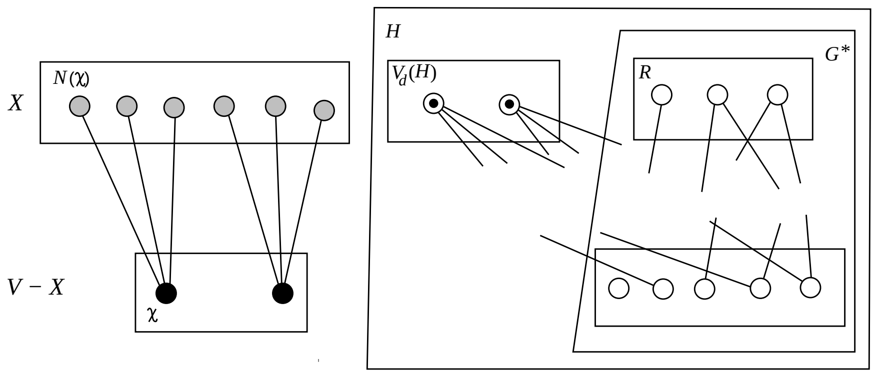

Figure 2: The structure of G with respect to vertex cover X and indirect certificate χ.

Since we can easily find a minimum vertex cover in graphs of maximum degree at most 2 in linear time, the above lemma gives us the following smaller indirect certificates.

Theorem 1.4 [Cai 2018] *Every yes-instance (G, k) of* Vertex Cover[k] *admits a certificate χ with at most $k/3$ vertices.*

With Lemma 1.3 at hand, we can easily improve the success probability of Algorithm VC-IC[k] from 4^{-k} to $2^{-\frac{4}{3}k} > 2.5199^{-k}$. In fact, we can further improve it to 2.1166^{-k} by optimizing the probabilities of colouring vertices red or blue.

> Algorithm VC-IC[$k/3$]
> Input: A graph $G = (V, E)$.
> Output: A k-vertex cover X of G, if it exists.
>
> 1. Randomly and independently colour each vertex red with probability $1/4$ and blue with probability $3/4$ to generate a random vertex-partition (V_r, V_b) of G.
> 2. Compute a minimum vertex cover X^* of $G - (N[V_r] \cup V_3(G - N[V_r])$, and output $N(V_r) \cup V_3(G - N[V_r]) \cup X^*$ as X.

Theorem 1.5 [Cai 2018] Algorithm VC-IC[$k/3$] *finds, with probability at least* 2.1166^{-k}, *a* k*-vertex cover of* G*, if it exists, in* $O(kn)$ *time.*

1.3 Semi-random partition

First we note that in Algorithm VC-IC[k], the colour of a vertex is totally independent of colours of other vertices, and the order we process vertices is immaterial. This is in a sense wasteful as a red vertex actually forces all its neighbours to be blue in order to obtain a valid (χ, X)-partition. We now incorporate this forcing step into our algorithm to make the colouring stage semi-random, which increases the chance of obtaining a valid (X, χ)-partition and hence the chance of success for the algorithm.

> Procedure Semi-Random-Partition
> Input: Graph $G = (V, E)$.
> Output: Red-blue partition (V_r, V_b) of V.
>
> Repeat the following until all vertices of G are coloured: Randomly choose an uncoloured vertex v, colour it red or blue with probability p for red and probability $1 - p$ for blue, and colour all neighbours of v blue if v is coloured red.

We remark that the above procedure does not recolour neighbours of v as they are either uncoloured or blue when the procedure colours v. To see this, we note that if any vertex $u \in N(v)$ is red, then u is coloured red before v, which would have forced v blue.

Lemma 1.6 [Cai 2018] *Let* p *be the probability that a vertex is coloured red. For any vertex* v *of degree* d*, the probability* $P(v)$ *that* Semi-Random-Partition *colours* v *red and all vertices in* $N(v)$ *blue is at least*

$$\frac{1 - (1-p)^{d+1}}{d+1}$$

for $p < 1$ *and* $1/(d+1)$ *for* $p = 1$*.*

We can replace Step 1 in Algorithm VC-IC[k] by Semi-Random-Partition to obtain a new algorithm that finds in linear time a k-vertex cover of G, if it exists, with probability at least $(3/8)^{-k} > 2.667^{-k}$. In fact we can fine-tune the probability p to maximize the success probability of our algorithm, and it turns out that the optimal value is $p = 1$. This is surprising, as it suggests that there is no need for randomness in a red-blue colouring, and it is the ordering of vertices that determines the success probability of our algorithm. Indeed, this perspective gives us another unexpected algorithm based on random selection: *Randomly choose a vertex and declare it to be not in solution.*

Algorithm VC-SRP
Input: A graph $G = (V, E)$.
Output: A k-vertex cover X of G, if it exists.

1. Repeat the following until all vertices are coloured: Randomly and uniformly choose an uncoloured vertex v, colour v red and all neighbours of v blue to form a (V_r, V_b)-partition of V.
2. Output $N(V_r)$ as X.

We can use Lemma 1.1 and Lemma 1.6 to obtain the following result, where the probability can be improved to 1.6633^{-k} by using indirect certificates of size $k/3$.

Theorem 1.7 [Cai 2018] Algorithm VC-SRP *finds, with probability at least* 2^{-k}*, a* k*-vertex cover of* G*, if it exists, in* $O(kn)$ *time.*

References

[1] Cai, L., Chan, S. M., and Chan, S. O., Random separation: A new method for solving fixed-cardinality optimization problems. In International Workshop on Parameterized and Exact Computation, LNCS 4169, pp. 239-250 (2006).

[2] Cai, L. and Ye, J., Finding two edge-disjoint paths with length constraints. In International Workshop on Graph-Theoretic Concepts in Computer Science, LNCS 9941 pp. 62-73 (2016).

[3] Cygan, M., Marx, D., Pilipczuk, M., Pilipczuk, M., and Schlotter, I., Parameterized complexity of Eulerian deletion problems. Algorithmica, 68(1), 41-61 (2014).