Algebra and Computation Course Instructor: V. Arvind

Lecture 8,9: General Bounded Degree GRAPH-ISO

Lecturer: V. Arvind *Scribe: Ramprasad Saptharishi*

1 Recap

In the last few classes, we saw that we could reduce trivalent graph isomorphism to SET-STAB for 2-groups. The algorithm is identical to a more general problem, which could be used to solve GRAPH-ISO with degree of each vertex bounded by a constant d.

The idea is more or less the same, we have two graphs with max degree bounded by a constant d. We shall, in polynomial time, reduce this problem to a restricted SET-STAB problem that could be solved in polynomial time.

2 Generalization to $(\leq d)$-degree graphs

Given two graphs X_1 and X_2 with the additional promise that the max degree in both of them is bounded by a constant d.

Just as in the trivalent case, add a new bridge between two edges of the graph. Then problem the reduces to finding the automorphism group of the graph that fixes this distinguished edge e. Let X_i be the subgraph consisting of edges that are reachable by paths of length atmost i through e. As before, any e-automorphism must preserve these layers and hence we have a natural map $\pi_i : Aut_e(X_{i+1}) \longrightarrow Aut_e(X_i)$. Using these maps, we are going to find a generating set for $Aut_e(X_n) = Aut_e(X)$.

The claim is that the groups in discussion are special, all their composition factors are $\leq d$. We shall prove this by induction just as in the trivalent case (where each of the factors were atmost 2, thus giving us a 2-group).

2.1 $\ker \pi_i$ has small factors

The kernel is the set of e-automorphisms of X_{i+1} that fix X_i. And since the degree is bounded by d, every vertex in X_i can reach out to atmost d vertices in X_{i+1}, and all automorphisms in the kernel can only permute the vertices with common parents.

Let A be the set of subsets of $V(X_i)$ of size atmost d. Then we have the neighbourhood map $\Gamma : V(X_{i+1}) \setminus V(X_i) \longrightarrow A$ such that $\Gamma(u) = a$ if a is the set of neighbours of u. Note that $\left\{\Gamma^{-1}(a)\right\}$ partitions the vertices in $X_{i+1} \setminus X_i$ and $|\Gamma^{-1}(a)| \leq d$.

Thus clearly, $\ker \pi_i = \bigotimes \mathrm{Sym}\left(\Gamma^{-1}(a)\right)$ and thus has composition factors atmost d.

This hence proves the claim that each of the automorphism has small factors. The above argument also yields a generating set for the kernel. Once we have a generating set for the image as well, we can construct the automorphism group of X_{i+1} from X_i.

2.2 Image of π_i reduces to restricted SET-STAB

Similar to the trivalent case, the image is just the set of all automorphisms that stabilize the following sets.

Edges within X_i:

$$A' = \{\text{2-subsets of } A \text{ that are new edges inside } X_i\}$$

Parent with same number of siblings:

$$A_s = \left\{a \in A \ : \ |\Gamma^{-1}(a)| = s\right\} \qquad 1 \leq s \leq d$$

Claim 1. *The image is precisely the subgroup of automorphisms of $Aut_e(X_i)$ that stabilize each of the A_s and A'.*

The proof is identical to the trivalent case, we shall hence skip it.

3 The restricted SET-STAB

Given a group $G = \langle A \rangle \leq \mathrm{Sym}(\Omega)$ and $\Delta \subseteq \Omega$. We are interested in finding $G_\Delta = \{g \in G \ : \ \Delta^g = \Delta\}$.

We do not hope to solve the general problem in polynomial time. However, a restricted version where G is a group with small compositional factors can be done efficiently. Divide-and-conquer is the method adopted in the algorithm.

Definition 2. *A finite group $G \in \mathcal{B}_d$ if every composition factor is isomorphic to a subgroup of S_d.*

From the earlier sections, we saw that $Aut_e(X) \in \mathcal{B}_d$ if the degree of X is bounded by d. Hence, we need to solve the SET-STAB problem for groups in $\mathcal{B}_d$.

Firstly, if $\Omega_1, \Omega_2, \cdots, \Omega_m$ are the G orbits then it is equivalent compute the stabilizer of $\Omega_i \cap \Delta$ for all i. Thus, we can assume that G acts transitively on Ω.

How do we further divide? Consider the block structure! Recall that blocks (Δ) are subsets of Ω such that they move as a whole ($\Delta^g \cap \Delta \neq \phi \Rightarrow \Delta = \Delta^g$). A block system naturally induces a tree structure on Ω. We shall call this the structure forest of G. And since we've already broken down Ω into G-orbits, we infact have just a structure tree.

picture might be useful

Each node is labelled by a block, that is the union of its children. The leaves are the trivial singleton blocks.

Now consider the top-most level, the root is labelled by Ω, which is the union of its children say $\Gamma_1, \cdots, \Gamma_m$. Now, considering each Γ_i as a point, G's action on them is primitive (there are no non-trivial blocks). The kernel of this action is the group $H = \{g \in G \ : \ \Gamma_i^g = \Gamma_i \quad \forall i\}$.

Let $G = \bigcup_{i=1}^r H\tau_i$, then $G_\Delta = \bigcup (H\tau_i)_\Delta$. But in order to talk about a stabilizer in a coset, we need to generalize the stabilizer problem.

3.1 Generalized SET-STAB

One could think of Δ as a 2-colouring of the set Ω. Thus a natural generalization would be the following.

Given a colouring C of Ω, we wish to find

$$G_C = \{g \in G \ : \ a^g \text{ has the same colour as } a \quad \forall a \in \Omega\}$$

Let us extend this a little further to capture the coset structure as well.

Given a coloured Ω , $G \leq \mathrm{Sym}(\Omega)$, $\sigma \in \mathrm{Sym}(\Omega)$ and $\Omega' \subseteq \Omega$ that is G-stable[1], we wish to find

$$stab(\Omega', G\sigma) = \left\{g \in G\sigma \ : \ \omega^g \sim \omega \quad \forall \omega \in \Omega'\right\}$$

Then we have the following easy claim, the proof is left to the reader (simple though)

[1] stable under action of G, union of orbits

Claim 3. *$stab(\Omega', G\sigma)$ is a right coset of $stab(\Omega', G)$.*

We can hence ask the following question: Given $G = \langle A \rangle \leq \mathrm{Sym}(\Omega)$ that is coloured and a Ω' that is a G-stable set and a $\sigma \in \mathrm{Sym}(\Omega)$. How do we compute $stab(\Omega', G\sigma)$?

With this generalization, we can do the divide and conquer.

4 The Divide and Conquer

Given a group $G \in \mathcal{B}_d$, we want to solve the stabilizer problem. The first step is to break Ω into orbits and work on each of them separately. Once we do that, we look at the block structure of G and at the topmost level.

We have a single tree, with Ω at the root with $\Gamma_1, \Gamma_2, \cdots, \Gamma_m$ as the primitive blocks on which G acts. The kernel of this action is $H = \{g \in G \,:\, \Gamma_i^g = \Gamma_i \quad \forall i\}$. Let

$$G\sigma = \bigcup_{i=1}^{r} H\tau_i\sigma$$

Now all that we need to do is compute $stab(\delta \cap \Gamma_i, H\tau_j\sigma)$ for all i and j and we have a recursive algorithm.

The catch here is that we need to know how many recursive calls to make. It would be completely useless if r (the index of H in G) was exponential; we need a decent bound on r. Fortunately, we do have such a bound.

Theorem 4 (Babai, Cameron, Pálfy)**.** *If $G \leq S_n$ belongs to $\mathcal{B}_d$ and is primitive, then $|G| \leq n^{cd}$ where c is an absolute constant.*

Thus, $r \leq m^{cd}$ and the algorithm would then go through. It can be shown that we then have an $O(n^d)$ algorithm for the restricted SET-STAB, and hence a $O(n^{d^2})$ algorithm for bounded degree graph isomorphism.