Automorphism Group of Graphs

Ritam M Mitra

Indian Statistical Institute

19th April, 2024

Graphs

- G=(V,E); V=set of vertices; E=set of edges
- order of G = V = n

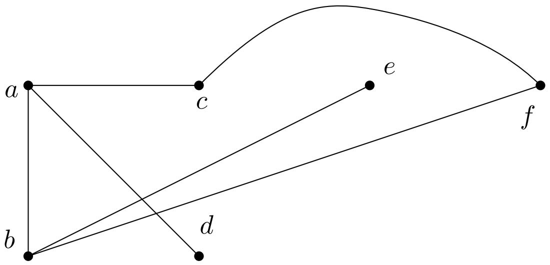


Figure: V={a,b,c,d,e,f} E={ab,ac,ad,be,cf,bf}

Permutation Group

- A *permutation* of the set Ω is a bijective mapping $g : \Omega \to \Omega$
- The composition $g_1 g_2$ of two permutations g_1 and g_2 is

$$v(g_1 g_2) = (v g_1) g_2 \textit{ for each } v \in \Omega$$

- G is closed under composition: if $g_1, g_2 \in G$ then $g_1 g_2 \in G$;
- G contains the *identity* permutation 1, defined by $v1 = v$ for $v \in \Omega$;
- G is closed under inversion, where the inverse of g is the permutation g^{-1} defined by the rule that $vg^{-1} = w$ if $wg = v$;

Automorphism of graphs

Definition

An automorphism on a graph G is a bijection $\phi : V(G) \to V(G)$ such that $uv \in E(G)$ if and only if $\phi(u)\phi(v) \in E(G)$.

Note that graph automorphisms preserve *adjacency*. In layman terms, a graph automorphism is a symmetry of the graph.

Example

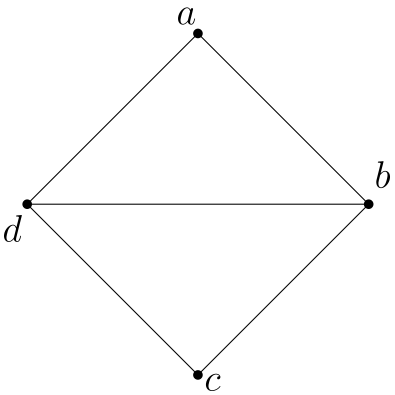


Figure: Graph G

Example

One automorphism switches vertices a and c.

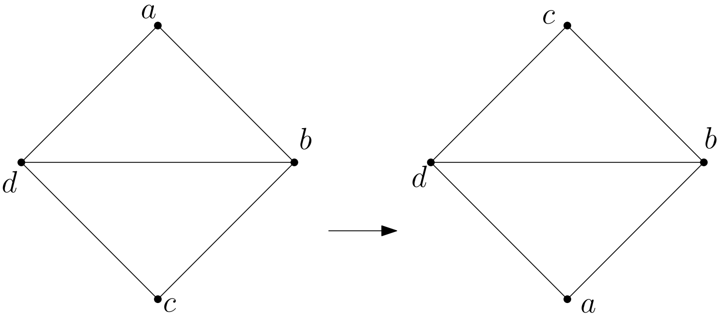


Figure: $\alpha = (a, c)(b)(d)$

Example

One automorphism switches vertices b and d.

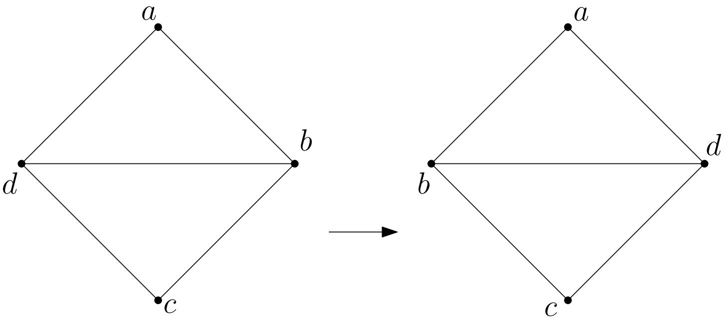


Figure: $\beta = (a)(b, d)(c)$

Example

The final automorphism switches vertices *a* and *c* and also switches *b* and *d*.

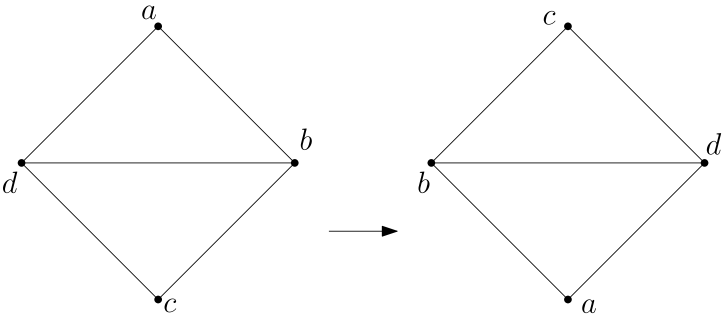


Figure: $\alpha\beta = (a, c)(b, d)$

The Automorphism Group

Definition

Let $Aut(G)$ denote the set of all automorphisms on a graph G. Note that this forms a group under function composition.

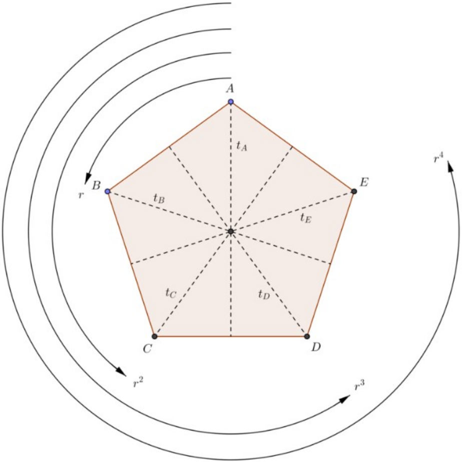
- $Aut(G)$ is closed under function composition.
- Function composition is associative on $Aut(G)$.
- There is an identity element in $Aut(G)$. This is mapping $e(v) = v$ for all $v \in V(G)$.
- For every $\sigma \in Aut(G)$, there is an inverse element $\sigma^{-1} \in Aut(G)$. Since σ is a bijection, it has an inverse. By definition, this is an automorphism.

Facts about graph automorphisms

- Graph automorphisms are degree preserving. In other words, for all $u \in V(G)$ and for all $\phi \in Aut(G)$, $deg(u) = deg(\phi(u))$.
- Graph automorphisms are distance preserving. In other words, for all $u, v \in V(G)$ and for all $\phi \in Aut(G)$, $d(u, v) = d(\phi(u), \phi(v))$.
- The automorphism group of G is equal to the automorphism group of the complement $\overline{G}$.

Dihedral Group

The 5-cycle C_5 has ten automorphisms, realized geometrically as the rotations and reflections of a regular pentagon.



Graph Isomorphism

Definition

There exists a function $'f'$ from vertices of G_1 to vertices of G_2
$[f : V(G_1) \rightarrow V(G_2)]$, such that
(i) f is a bijection (both one-one and onto)
(ii) f preserves adjacency of vertices, i.e., if the edge $\{u, v\} \in G_1$, then the edge $\{f(u), f(v)\} \in G_2$,
then $G_1 \equiv G_2$.

Example

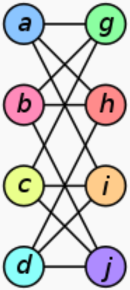
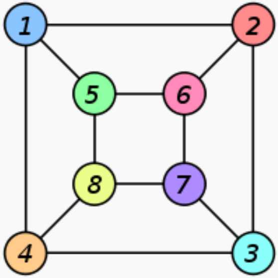
Graph G	Graph H	An isomorphism between G and H
		$f(a) = 1$ $f(b) = 6$ $f(c) = 8$ $f(d) = 3$ $f(g) = 5$ $f(h) = 2$ $f(i) = 4$ $f(j) = 7$

Figure: f is an isomorphism

Algorithmic questions

Question 1

Graph isomorphism

Instance: Graphs G and H

Question: Is $G \cong H$?

Question 2

Automorphism group

Instance: A graph G

Output: generating permutations for $\mathrm{Aut}(G)$.

Reduction

Theorem

With **Graph-Iso** as an oracle, there is a polynomial time algorithm for **Graph-Aut** and vice-versa.

vice-versa part

- given two graphs G_1 and G_2
- Create $G = G_1 \cup G_2$
- G_1 and G_2 are connected
- if any of the generators of $\mathrm{Aut}(G)$ interchanged a vertex in G_1 with one in G_2, then connnectivity should force $G_1 \cong G_2$
- if not G_1 and G_2 connected
- $G_1 \cong G_2 \iff \overline{G_1} \cong \overline{G_2}$
- either G_1 or $\overline{G_1}$ has to be connected

Conclusion

- Constructing the automorphism group is at least as difficult (in terms of its computational complexity) as solving the graph isomorphism problem.
- Similar to the graph isomorphism problem, it is unknown whether it has a polynomial time algorithm or it is NP-complete.

Questions...?