Graph Isomorphism: Notes

Prajwal Neupane

July 27, 2025

Background and Motivation

- Initiated contact with Prof. Schroeder regarding graph isomorphism (GI); received Grohe & Neuen's 2021 survey.
- Aim: Understand both the combinatorial (color refinement, Weisfeiler-Leman) and group-theoretic (Luks, Babai) approaches to GI, implement algorithms, and explore complexity and open research directions.

Foundational Definitions

- **Graph Isomorphism**: Given two graphs G_1, G_2, an isomorphism is a bijection $\sigma : V(G_1) \to V(G_2)$ preserving adjacency: $(u, v) \in E(G_1) \iff (\sigma(u), \sigma(v)) \in E(G_2)$.
- **Permutations and Groups**: $\Omega = \{1, 2, \ldots, n\}$; $\mathrm{Sym}(\Omega)$ is the group of all bijections $\Omega \to \Omega$. Subgroups and group actions underpin the group-theoretic approach.
- **Complexity Classes**: *Polynomial time* ($O(n^k)$), *Non-polynomial* ($O(2^n), O(n!)$), *Quasi-polynomial* ($O(n^{\log^c n})$).

Color Refinement (1-WL) Algorithm

- **Idea**: Iteratively refines vertex coloring by aggregating the multiset of neighbor colors and arc colors.

- **Mathematical formulation**:

$$\begin{aligned} \chi^{(0)}(G) &:= \chi_V \\ \chi^{(i)}(G)(v) &:= (\chi^{(i-1)}(G)(v), M_i(v)) \\ M_i(v) &:= \text{multiset}\left\{(\chi^{(i-1)}(G)(w), \chi_E(v,w), \chi_E(w,v)) : w \in N_G(v)\right\} \end{aligned}$$

- **Properties**:
  - Stabilizes after at most n iterations.
  - Efficient: $O((n+m)\log n)$ time.
  - Not complete: Some non-isomorphic graphs are not distinguished by 1-WL.

- **Implementation progress**: Coded basic structure for node and arc coloring using NetworkX. Refinement loop and color compression (mapping multisets to integers) in progress.

Group-Theoretic Approach (Luks, Babai)

- **String Isomorphism Formulation**: Given $x, y : \Omega \to \Sigma$ and $G \leq \mathrm{Sym}(\Omega)$, does $\exists \sigma \in G$ such that $x^\sigma = y$?

- **Automorphisms and Isomorphisms**:

$$\begin{aligned} \mathrm{Iso}_G(x,y) &= \{\sigma \in G \mid x^\sigma = y\} \\ \mathrm{Aut}_G(x) &= \mathrm{Iso}_G(x,x) \end{aligned}$$

- **Structure theorem**: If $\mathrm{Iso}_G(x,y) \neq \emptyset$, then $\mathrm{Iso}_G(x,y) = \mathrm{Aut}_G(x) \cdot \sigma_0$ for any isomorphism σ_0.

- **Complexity breakthrough** (Babai 2016): General GI solved in quasi-polynomial time, $O(n^{\log^c n})$, via deep group-theoretic recursion.

Complexity Insights

- 1-WL: Practical, fast for many graphs, but incomplete.

- Group-theoretic: Handles all graphs, previously only bounded-degree in polynomial time (Luks); Babai's work extends to all graphs in quasi-polynomial time.

- GI is not known to be in P or NP-complete; its precise complexity remains one of the great open questions.

Coding and Experiments

- Implemented directed colored graphs in `networkx`.
- Set up Jupyter notebooks for iterative refinement and visualization.
- Explored visualization of node and arc colorings; debugged issues with color data types and plotting.
- Next steps: Implement multiset-based color refinement loop, test on small nontrivial examples, compare to networkx built-in isomorphism tools.

Theory/Practice Links and Further Exploration

- Surveyed literature: Grohe & Neuen 2021 (main survey), Babai 2016 (quasi-poly), group action properties.
- Explored the connection between string isomorphism, adjacency matrices, and group actions on graph labels.
- Noted that practical GI solvers (e.g., nauty, Traces) blend coloring and group-theoretic methods.
- Open questions: Automorphism group structure for small colored digraphs, practical detection of 1-WL counterexamples, role of parameterized complexity.

Next Steps

(a) Finalize and test color refinement implementation on directed, colored graphs.

(b) Explore group-theoretic tools (e.g., permutation group libraries) and attempt small-scale automorphism group calculations.