Graph Isomorphism

Franklin van Nes

Tuesday 12th November, 2017

1. Formally define and describe the topic (whether technique, algorithm, model or class) in detail, and discuss its significance.
   Graph Isomorphism is very much defined in its name: Iso (same) morphism (shape) of graphs. In more detail, two finite graphs are isomorphic if they share the same number of vertices connected in the same way. Formally, two finite graphs G and H with graph vertices $G.V$ and $H.V$ labelled $V_n = \{1, 2, 3, 4, ..., n\}$, and graph edges $G.E$ and $H.E$ are said to be isomorphic if $f : G.V \to H.V$ is an edge-preserving bijection. Meaning that there exists a permutation p of V_n so that edges $(u, v) \in G.E \iff (p(u), p(v)) \in E.V$ [1]. Figure 1 shows how two isomorphic graphs can look very different while still maintaining the same structure.

2. How do you show a problem is in the class? While graph isomorphisms are quite clear in their classification as defined above, it should be noted that Graph Isomorphism falls into its own category of problem complexity, called *Graph Isomorphism Complete*. According to Lubiw, it has yet to fall into a typical classification, and is neither P nor NP-complete [2]. There exists no known P algorithm, yet graph isomorphism has not be shown to be NP-complete. This means that either a P algorithm must exist for graph isomorphism, graph isomorhpism is a problem outside of P and NP, or, as Schning argues, Graph isomorphism problems are in the low hierarchy of NP, which "does not equal NP unless the polynomial hierarchy collapses to the second level" [4]. Deeper explanation of this exceeds the reaches of this paper, though Schning's research is well explained with the proper background.

3. Which are representative or classic problems in this class?

4. How does this class compare to other classes?

5. What techniques are used to solve problems in this class?

1 Figures

References

[1] G. Chartran. *Isomorphic Graphs*. Dover, 1985.

[2] A. Lubiw. Some np-complete problems similar to graph isomorphism. In *SIAM Journal on Computing*, pages 11–22. Society for Industrial and Applied Mathematics, February 1981.

[3] C. Martin.

[4] U. Schning. A low and a high hierarchy within np. *Journal of Computer and System Sciences*, 27(1):14 – 28, 1983.

Figure 1: Two isomorphic graphs. The colors indicate matching vertices, even if their labels do not match. [3]