Journal of Symbolic Computation 59 (2013) 81–112

Contents lists available at SciVerse ScienceDirect

Journal of Symbolic Computation

www.elsevier.com/locate/jsc

The Graph Isomorphism Problem and approximate categories

Harm Derksen [1]

University of Michigan, Department of Mathematics, 530 Church Street, Ann Arbor, MI 48109-1043, United States

ARTICLE INFO

Article history:
Received 6 May 2013
Accepted 1 June 2013
Available online 1 July 2013

Keywords:
Graph Isomorphism Problem
Weisfeiler–Lehman algorithm
Complexity Theory
Categories

ABSTRACT

It is unknown whether two graphs can be tested for isomorphism in polynomial time. A classical approach to the Graph Isomorphism Problem is the d-dimensional Weisfeiler–Lehman algorithm. The d-dimensional WL-algorithm can distinguish many pairs of graphs, but the pairs of non-isomorphic graphs constructed by Cai, Fürer and Immerman it cannot distinguish. If d is fixed, then the WL-algorithm runs in polynomial time. We will formulate the Graph Isomorphism Problem as an Orbit Problem: Given a representation V of an algebraic group G and two elements $v_1, v_2 \in V$, decide whether v_1 and v_2 lie in the same G-orbit. Then we attack the Orbit Problem by constructing certain approximate categories $\mathcal{C}_d$, $d \in \mathbb{N} = \{1, 2, 3, \ldots\}$ whose objects include the elements of V. We show that v_1 and v_2 are not in the same orbit by showing that they are not isomorphic in the category $\mathcal{C}_d$ for some $d \in \mathbb{N}$. For every d this gives us an algorithm for isomorphism testing. We will show that the WL-algorithms reduce to our algorithms, but that our algorithms cannot be reduced to the WL-algorithms. Unlike the Weisfeiler–Lehman algorithm, our algorithm can distinguish the Cai–Fürer–Immerman graphs in polynomial time.

1. Introduction and main results

1.1. The Graph Isomorphism Problem

Suppose that Γ_1 and Γ_2 are two graphs on n vertices. The *Graph Isomorphism Problem* asks whether they are isomorphic or not. In Computational Complexity Theory, the Graph Isomorphism Problem

E-mail address: hderksen@umich.edu.

[1] The author is supported by the NSF, grant DMS 0901298.

http://dx.doi.org/10.1016/j.jsc.2013.06.002

plays an important role, because it lies in the complexity class **NP**, but it is not known whether it lies in **P** or **NP-complete**. See Köbler et al. (1993) for more details. Based on Valiant's algebraic version of the **P** versus **NP** problem (Valiant, 1979), Mulmuley and Sohoni reformulated Valiant's **P** versus **NP** problem into a question about orbits of algebraic groups in Mulmuley and Sohoni (2001, 2008). In this paper, we will study the Graph Isomorphism Problem in terms of orbits of algebraic groups, but our approach is not closely related to the work of Mulmuley and Sohoni.

For special families of graphs there are polynomial time algorithms for the Graph Isomorphism Problem. Polynomial time algorithms were found for trees (Edmonds' algorithm, see Busacker and Saaty, 1965, p. 196), planar graphs (Hopcraft and Tarjan, 1973; Hopcroft and Wong, 1974) and more generally for graphs of bounded genus (Filotti and Mayer, 1980; Miller, 1980), for graphs with bounded degree (Luks, 1982), for graphs with bounded eigenvalue multiplicity (Babai et al., 1982), and for graphs with bounded color class size (Luks, 1986).

A general approach to the Graph Isomorphism Problem was developed by Weisfeiler and Lehman in the 1960s. The d-dimensional Weisfeiler–Lehman algorithm $\mathbf{WL}_d$ systematically colors e-tuples of vertices ($e \leqslant d$) until a stable coloring is obtained (see Weisfeiler and Lehman, 1968; Weisfeiler, 1976). The d-dimensional WL-algorithm terminates with a proof that the two graphs are not isomorphic, or it terminates with an inconclusive result. If $d \geqslant n$, then the d-dimensional Weisfeiler–Lehman algorithm will distinguish all non-isomorphic graphs with n vertices. For fixed d, the Weisfeiler–Lehman algorithm runs in polynomial time. The higher-dimensional Weisfeiler–Lehman algorithm can distinguish graphs in many families of graphs. However, Cai, Fürer and Immerman showed in Cai et al. (1992) that for every d, there exists a pair of non-isomorphic graphs with degree 3 and $O(d)$ vertices which cannot be distinguished by the d-dimensional Weisfeiler–Lehman algorithm. The set of Weisfeiler–Lehman algorithms $\mathbf{WL} = \{\mathbf{WL}_d\}_{d\in\mathbb{N}}$ is an example of what we will call a *family of GI-algorithms*:

Definition 1.1. A family of GI-algorithms is a collection of algorithms $\mathbf{A} = \{\mathbf{A}_d\}_{d\in\mathbb{N}}$ such that

(1) The input of $\mathbf{A}_d$ consists of two graphs with the same number of vertices. The value of the output is either "`non-isomorphic`" or "`inconclusive`". If the output is "`non-isomorphic`" then the graphs are not isomorphic and we say that $\mathbf{A}_d$ distinguishes the two graphs.
(2) If the graphs are not isomorphic, then $\mathbf{A}_d$ distinguishes them for some d.
(3) For fixed d, $\mathbf{A}_d$ runs in polynomial time.

Besides the Weisfeiler–Lehman algorithm, there are other families of polynomial time algorithms for the Graph Isomorphism Problem. In order to compare various algorithms, we make the following definition (see also Evdokimov et al., 1999, §6):

Definition 1.2. For two families of GI-algorithms $\mathbf{A} = \{\mathbf{A}_d\}_{d\in\mathbb{N}}$ and $\mathbf{B} = \{\mathbf{B}_d\}_{d\in\mathbb{N}}$ we say that $\mathbf{A}$ is reducible to $\mathbf{B}$ if there exists a function $f : \mathbb{N} \to \mathbb{N}$ such that for every d and every pair of graphs which $\mathbf{A}_d$ distinguishes, the pair can be distinguished by $\mathbf{B}_{f(d)}$. We say that $\mathbf{A}$ and $\mathbf{B}$ are equivalent if $\mathbf{A}$ is reducible to $\mathbf{B}$ and $\mathbf{B}$ is reducible to $\mathbf{A}$.

The Weisfeiler–Lehman algorithm is combinatorial in nature. There are also more algebraic approaches to the Graph Isomorphism Problem. The 2-dimensional Weisfeiler–Lehman algorithm can be formulated in terms of *cellular algebras* (see Weisfeiler, 1976).[2] These algebras were introduced by Weisfeiler and Lehman, and independently by D. Higman under the name *coherent algebras* (see Higman, 1987; Friedland, 1989). In Evdokimov et al. (1999), Evdokimov, Karpinski and Ponomarenko introduced the d-closure of a cellular algebra. One may view d-closed cellular algebras as higher-dimensional analogs of the cellular algebras. The algorithm based on this d-closure will be denoted by $\mathbf{CA}_d$. In Evdokimov et al. (1999) it was shown that the algorithm $\mathbf{CA}_d$ distinguishes any two graphs which can be distinguished by $\mathbf{WL}_d$. In Evdokimov and Ponomarenko (1999, Theorem 1.4)

[2] These cellular algebras should not be confused with a different, seemingly unrelated notion of cellular algebras introduced in Graham and Lehrer (1996).