Data Science

A HEURISTIC APPROACH TO THE VERIFICATION OF ISOMORPHIC GRAPHS

E.F. Sayfullina

Togliatti State University, Togliatti, Russia

Abstract. A heuristic approach to the verification of isomorphic graphs is described in this work. The approach is a sequential verification of the graph characteristics which are invariants. The results of computational experiments are described. The aim of experiments is to determine which of the comparative sequences (for graph invariants values) are more efficient for graph isomorphism verification.

Keywords: isomorphic graphs, invariants, heuristic approach.

Introduction

An isomorphism between two undirected and unweighted graphs is a bijection between the vertex sets of these graphs such that two vertices of the first graph are adjacent if and only if corresponding vertices of the second graph are adjacent [1]. In case of isomorphism between two directed or weighted graphs additional restrictions are imposed on directions and weights of the edges.
The graph isomorphism problem is one of computational complexity theory standard problems belonging to NP, but not known to belong to P (in assumption that P $\neq$ NP). It's not known yet whether this problem is belonging to NP-complete [2], but it's known that subgraph isomorphism problem is belonging to NP-complete. Thus, the recent researches for both random and specific graphs are relevant.
Graph invariant is any graph property which is equal for isomorphic graphs [3, 4].
Graph invariant is complete if its value is equal for two graphs if and only if these graphs are isomorphic. Currently known complete graph invariants are min-code and max-code code of adjacency matrix obtained by writing out binary adjacency matrix's values in place followed by subsequent transfer of the resulting binary number to decimal form. Man-code corresponds to such order of adjacency matrix's rows and columns when the result value is the largest. Currently complete graph invariant which can be calculated in polynomial time is not known. But it's not proved that such invariant does not exist.

The most obvious graph invariants are number of vertices $n(G)$ and number of edges $m(G)$.

For graph $G = (V, E)$ number of vertices adjacent with vertex v or number of rows incident with vertex v is called degree s of vertex v. Apparently that for any isomorphic graphs L and L' corresponding vertices are having the same degree.

For graph G an arranged system of degrees $(s_1, s_2, ..., s_n)$ of its vertices is called degree sequence $s(G)$. It also called graphic sequence.

A degree sequence $s(G) = (s_1, s_2, ..., s_n)$ produces two more numeric graph invariants: $min(s_i)$ and $max(s_i)$ $(i = 1,2,...n)$. The second one is also called graph degree.

Description of heuristic approach to the verification of isomorphic graphs and results of computational experiments

This paper examines the following graph invariants [1, 3, 4].

Chromatic number, the smallest number of colors for the vertices in a proper coloring.

Diameter, the longest of the shortest path lengths between pairs of vertices.

Wiener index – $\omega = \sum d(v_i, v_j)$, where $d(v_i, v_j)$ – shortest path between vertices v_i and v_j.

Randic index – $r = \sum\limits_{(v_i,v_j)} \dfrac{1}{\sqrt{d(v_i)d(v_j)}}$, where v_i and v_j are two adjacent vertices,

$d(v_k)$ – degree of vertex k.

Determinant of adjacency matrix.

Number of connected components.

Cyclomatic number – minimal number of edges which need to be removed in order to graph became acyclic. $p_1(G) = p_0(G) + |E(G)| - |V(G)|$, where $p_1(G)$ – cyclomatic number, $p_0(G)$ – number of connected components, $|E(G)|$ – number of edges, $|V(G)|$ – number of vertices.

We describe a new graph invariant (property) second-level degree sequence (such invariant can be described based on more general models [5]). Every element of this sequence is a list of degrees of the vertices adjacent to the current vertex of graph. It's obvious that this property is a graph invariant.

There are two graphs on the pictures below and the values of the invariants for these graphs (Table 1).

The values of these invariants are equal for Graph1 and Graph2. But these two graphs have different second-level degree sequences: [[2, 3, 3, 4, 5, 5, 6], [3, 3, 4, 4, 5, 7], [2, 3, 5, 6, 7], [3, 4, 4, 5, 7], [3, 5, 6, 7], [3, 3, 5, 6], [5, 6, 7], [4, 5, 7], [4, 4, 6], [5, 7]] – for Graph1, and [[2, 3, 3, 4, 5, 5, 6], [2, 3, 4, 4, 5, 7], [3, 3, 5, 6, 7], [3, 4, 4, 5, 7], [3, 5, 6, 7], [3, 3, 5, 6], [4, 5, 7], [4, 5, 7], [4, 5, 6], [6, 7]] – for Graph2. Thus, it can be concluded that Graph1 and Graph2 are nonisomorphic.

For graph isomorphism problem (in case of random graph) all know algorithms ensuring the correct answer are exponential. But for almost every discrete optimization problem there several different approaches for construction of algorithms that are solved this problem. Each of these approaches is more effective for specific set of

incoming data. Regarding this statement in 1997 NoFreeLunchTheorem [6] was formulated and proved. There several different interpretations of this theorem. One of these interpretations is described below.

Table 1. Invariants of Graph 1 and Graph 2

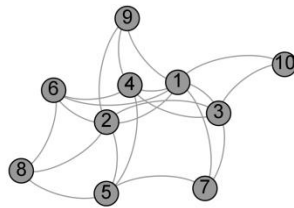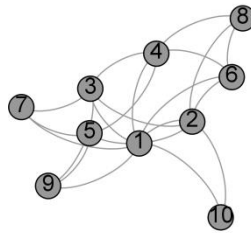|	Graph1	Graph2
Number of vertices	10	10
Number of edges	21	21
Degree sequence	[2, 3, 3, 3, 4, 4, 5, 5, 6, 7]	[2, 3, 3, 3, 4, 4, 5, 5, 6, 7]
Determinant of adjacency matrix	0	0
Number of connected components	1	1
Chromatic number	3	3
Graph diameter	3	3
Wiener index	142	142

Fig. 1. Graph1

Fig. 2. Graph2

There is set of incoming data for the considered discrete optimization problem, instead of the set of incoming data an algorithm for generation of the incoming data is often specified. If for some set of incoming data (specified or generated) some discrete optimization problem solvation algorithm is optimal regarding some criteria (i.e. execution time), then there is other incoming data set for which this algorithm is not optimal regarding the same criteria.

During the research algorithms for random generation of graphs were described and implemented. Based on this algorithms a heuristics approach (depending on a given generation algorithm) for graph isomorphism problem solvation was described and implemented.

For computational experiments 1000 algorithms were randomly generated from the eight graph invariants:
- Chromatic number;
- Determinant of adjacency matrix;
- Graph diameter;
- Number of connected components;
- Cyclomatic number;
- Wiener index;
- Randic index;
- Second-level degree sequence.

For two validated graphs during each of these algorithms the values of these properties are calculated and then then comparing in particular order. If the value of one of these invariants is different for the two graphs, then the algorithm is stopped with the conclusion that these graphs are not isomorphic. If for all eight invariants the value are equal for two graphs then these graph are expected to be isomorphic.

The Markov chain was used for computational experiments. This chain consists of five states (probability of each state is 0.2). Each state is one of the following algorithms for random generation of graphs with given degree sequence.
- Markov chain Monte Carlo (MCMC) algorithm [7]
- A Sequential Importance Sampling Algorithm for Generating Random Graphs with Prescribed Degrees [8]
- Algorithm developed by A. Steger and N. Wormald [9]
- Algorithm for generation of graphs with given degree sequence developed by author [10]
- Algorithm for generation of graphs with given second-level degree sequence developed by author [10]

The algorithm of computational experiments is described below.

Input: number of vertices and distribution function of degree sequence.
- Generate degree sequence according to the given distribution function (such as Binominal distribution, Poison distribution, Zipf distribution).
- Generate initial graph with this degree sequence using Havel-Hakimi criteria [11]
- Generate graph with this degree sequence using the algorithm from the current state of the Markov chain. 1000 iterations were executed for Markov chain. Om each iteration one of the 1000 algorithms (one of the 1000 generated sequences of graph invariants comparison) were executed to validate whether initial graph and generated graph are isomorphic.

Based on the results of the computational experiment it can be concluded that the most effective invariants are Wiener index and second-level degree sequence and the least effective is determinant of adjacency matrix.

Conclusion

Described approach is one of the possible ways to choose the most efficient from several algorithms for solving some discrete optimization problem. In this case this approach was applied for graph isomorphism problem. Moreover, the algorithm is chosen for the given or generated input data.

References

1. Ore O. Graphs and their application. Moscow: "URSS", 2006. [In Russian]
2. Gromkovich Yu. Theoretical computer sciences. Introduction into automata theory, theory of calculability, complexity theory, algorithm theory, randomization, communication and cryptography theory. Saint Petersburg: "BKhV-Peterburg", 2010. [In Russian]
3. Zykov A. Fundamentals of graph theory. Moscow: "Nauka", 1986. [In Russian]
4. Harary F. Graph theory. Moscow: "Mir", 1973. [In Russian]
5. Ostroumova L. Proceedings of Moscow Institute of Physics and Technology, 2012; 4 N 1(13): 29-40. [In Russian]
6. Wolpert D, Macready W. No free lunch theorems for optimization. IEEE Transactions on Evolutionary Computation, 1997; 1(1): 67-82.
7. Berg B. Markov Chain Monte Carlo Simulations and Their Statistical Analysis. Singapore, World Scientific Publ, 2004.
8. Blitzstein J. A Sequential Importance Sampling Algorithm for Generating Random Graphs with Prescribed Degrees. URL : http://statweb.stanford.edu/~cgates/PERSI/papers/ degrees.pdf.
9. Steger A, Wormald N. Generating random regular graphs quickly. Combinatorics, Probab. and Comput., 1999; 8: 377-396.
10. Melnikov B, Sayfullina E. Applying multiheuristic approach to randomly generating graphs with a given degree sequence. University proceedings. Volga region. Physical and mathematical sciences. Mathematics, 2013; 3(27): 69-82.
11. Frank H, Hakimi S. IEEE Transactions on, 1965; 12(3): 44-51.