Pattern Recognition 58 (2016) 68–82

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

Filtering graphs to check isomorphism and extracting mapping by using the Conductance Electrical Model ☆

Manuel Igelmo, Alberto Sanfeliu *

Institut de Robòtica i Informàtica Industrial, CSIC-UPC, Llorens i Artigas 4-6, 08028 Barcelona, Spain

ARTICLE INFO

Article history:
Received 18 August 2015
Received in revised form
29 January 2016
Accepted 9 March 2016
Available online 17 March 2016

Keywords:
Graph isomorphism
Graph matching
Conductances Equivalent Model
Star Method
Graph filter

ABSTRACT

This paper presents a new method of filtering graphs to check exact graph isomorphism and extracting their mapping. Each graph is modeled by a resistive electrical circuit using the Conductance Electrical Model (CEM). By using this model, a necessary condition to check the isomorphism of two graphs is that their equivalent resistances have the same values, but this is not enough, and we have to look for their mapping to find the sufficient condition. We can compute the isomorphism between two graphs in $O(N^3)$, where N is the order of the graph, if their star resistance values are different, otherwise the computational time is exponential, but only with respect to the number of repeated star resistance values, which usually is very small. We can use this technique to filter graphs that are not isomorphic and in case that they are, we can obtain their node mapping. A distinguishing feature over other methods is that, even if there exists repeated star resistance values, we can extract a partial node mapping (of all the nodes except the repeated ones and their neighbors) in $O(N^3)$. The paper presents the method and its application to detect isomorphic graphs in two well know graph databases, where some graphs have more than 600 nodes.

1. Introduction

It is known that a graph is a powerful and flexible structure which allows modeling many types of objects and systems, due to this, graphs are used in many fields such as chemistry, biochemistry, transport, telephony, computer networks, voice recognition, and computer vision [1]; in many cases the graphs have a high number of nodes and/or edges [2].

In the field of Pattern Recognition, the process of evaluating the similarity of two graphs is referred as graph matching. In this area we can differentiate between two types of the methods: exact and inexact graph matching. The stringent way of defining the exact graph matching is the graph isomorphism, meanwhile the inexact graph matching looks for the best mapping between the graphs through minimizing a matching cost. There are numerous works that deal with the state of the art on the graph matching, such as [3–7]. Other papers ([8,9] among others) perform comparisons between different methods.

Two graphs are isomorphic when any node renumbering preserves adjacencies (unweighted graphs) or weights (weighted graphs). The graph as a data structure has the great drawback that the comparison between them requires computationally prohibitive calculation time [10], i.e. exponential time complexity with respect to the number of nodes. That is why there is a vast and extensive literature[1] to find reasonably quick ways to decide when two graphs are identical, i.e. isomorphic, and also if applicable, to extract the mapping between their nodes. Moreover, it is known that the graph isomorphism problem belongs to **NP**, but not been known to belong to either one of the following subsets: **P** and **NP**-complete [12] (see also [13]).

As we have already commented the graph isomorphism (exact matching) is an open problem, in contrast to other related graph problems whose computational complexity has been shown to be **NP**-complete such as graph homomorphism, subgraph isomorphism and maxim common subgraph[2] of graphs whose proof can be found in [14,10] and [15] respectively, so that all efforts are being dumped in search in polynomial time suboptimal solutions for these problems.

The foregoing is for graphs in general, but there are subsets of graphs for which subexponential solutions to the problem of

☆This work was partially funded by CICYT DPI2013-42458-P.

* Corresponding author.
E-mail addresses: migelmo@xtec.cat (M. Igelmo), sanfeliu@iri.upc.edu (A. Sanfeliu).
URLS: http://www.iri.upc.edu (M. Igelmo), http://www.iri.upc.edu (A. Sanfeliu).

[1] In [11] (published in 2013), the authors assert that there are a few hundreds of algorithms published on the subject.

[2] The maximum common subgraph problem is reducible to the problem of clique and this is **NP**-complete.

http://dx.doi.org/10.1016/j.patcog.2016.03.015
0031-3203/

Table 1
Comparison of features of some methods. Pay special attention to the column "Partial mapping?".

Method	Best case	Worst case	Search tree?	Iterative?	Partial mapping?	Closed form?
Ullmann [30]	$O(N^3)$	$O(N!N^3)$	Yes	Yes	No	No
SD [31]	$O(N^3)$	$O(N!N)$	Yes	Yes	No	No
VF [32]	$O(N^3)$	$O(N!N)$	Yes	Yes	No	No
Nauty [36]	$O(N^2 \log N)$	Exponential	Yes	Yes	No	No
SM (this paper)	$O(N^3)$	$\max(O(N^3), O(J!))$	No	No	Yes	Yes

isomorphism, such as planar graphs [16–19], rooted trees [20], graphs of bounded degree [21], interval graphs [22], circular graphs chords [23,24] and arcs [24], graphs of bounded genus [25–27], graphs of bounded eigenvalue [28] and graphs of bounded treewidth [29], have been shown.

There are other approaches to the problem of graph isomorphism for general case. Many of them use a tree search of solutions, these algorithms use brute force but with pruning to nonviable solutions and backtracking techniques. These differ essentially in the criteria for pruning, thus they have the algorithms of Ullmann [30], SD [31] and VF [32,33].

Other methods use the Theory of Groups seeking a canonical labeling of graphs allowing to discern whether they are isomorphic through their respective canonical equality [34]. These techniques also make use of a search tree and automorphisms of graphs. However, as is affirmed in [35], in terms of computational complexity, the theoretical state of canonical labeling is still unsolved. All these algorithms have been computationally implemented giving rise to (in chronological order) "nauty" [36], "saucy" [37], "Bliss" [38,39], "Traces" [11] and "conauto" [40].

Other inexact methods can also be applied to match graphs, not to solve the isomorphism problem, which finds a cost to map one graph to another one. There is an extensive literature on this topic which have already been mentioned [3–7]. We are not going mention these methods, because it is out of the scope of this article.

In this paper we present a completely new method for filtering graph isomorphism and at the same time, extract their node mapping that neither derives nor inspired by any of the aforementioned methods. This method can be applied to attributed graphs with only one numeric attribute (weight) in each edge, and for connected and undirected graphs. It also serves to unweighted graph if these are taken to each edge a unit weight. It cannot be applied to graphs with symbolic labels.

Our method, denominated the Star Method (hereinafter SM) is based on the Conductance Electrical Model (hereinafter CEM) [41]. It models weighted graphs where its weighted edges are transformed in conductances values (S) (we use conductances instead of resistance values (Ω)). The method can also be applied to unweighed graphs, where the value of the edge weight is equal to 1 in this case. By assuming serial connection of an ideal diode with a resistor, the method can be extended to directed graphs. Unfortunately this extension brings nonlinearities making the analysis much more complex (in terms of the circuit).

Using an electric model, we can apply the theories, methods and procedures that are well known in Electrical Circuit Theory (see among others [42]). In the literature, we have only find a work [43] that uses also electrical circuit representation, but oriented to define a resistance distance to match graph models.

Although the method that we propose is oriented to solve exact graph isomorphism in an efficient way, from the point of view of computational time complexity, reducing from exponential to cubic time complexity in most of the cases, we present the work as a filtering technique to eliminate the graphs that are not isomorphic, and detecting and extracting the node mapping of the graphs that are isomorphic. The reason is that in this way, the method can be applied to solve problems where checking isomorphism is the key issue. The proposed method uses the Conductance Electrical Model (CEM) and has two filtering processes. The first filtering process eliminates the graphs when the equivalent resistances do not match. The second filtering process, either detect that there exist an isomorphism and in this case extract the correct node mapping, or detect the graphs that are not isomorphic. The important difference is that the first process is cubic, $O(N^3)$, and the second process can be quadratic or exponential, but in this case only with respect to the star resistance values that are identical. This implies that in most cases, the graph isomorphism can be done in cubic time complexity, making this filtering process very efficient.

In order to compare our method with other well known methods, Table 1 shows a comparison using the following features: best time complexity case; worst time complexity case; if the method uses tree search; if it is an iterative method; if it can be obtained a partial matching; and if the method has a closed form.

We have selected these features to show that our method has some strengths. First, the best time complexity case is the same as the other methods. Second, the worst time complexity case is better than the other methods $\max(O(N^3), O(J!))$, because $J \ll N$. Third, it has a closed form, it is not probabilistic, not iterative neither recursive. Fourth, a partial mapping in $O(N^3)$ time complexity can be obtained, a feature that no other exact methods have. Finally, we have to underline that our method is based on a well known electrical circuit theory.

The rest of the paper is devoted to present the proposed model and method (CEM and SM) for filtering, analyze its characteristics and present experiments to verify the performance of them.

2. Filtering graphs to check isomorphism by using SM

Fig. 1 shows a block diagram of the SM using CEM, which will be used for the description of the method. The graphs modeled as CEM are characterized by having one numerical attribute in each edge (we will call them weights and they can be any non-negative value) and no attributes in their nodes. We can also treat unweighted graphs by assigning value 1 to the attribute of all edges. Hereinafter the two graphs modeled by the CEM will be denoted by g and h, and we will assume that both have the same order N.

2.1. First filter phase: obtaining CEM and equivalent resistances

Consider two undirected and connected graphs (weighted or unweighted[3]) g and h both of order N and size M. These graphs come characterized by their adjacency matrices A^g and A^h, respectively (input and 1st line of block A of Fig. 1). The CEM

[3] In this case we consider that they have unit weights in all edges, i.e., for a undirected graph we always consider in this paper that if nodes i and j are connected, then $\omega_{ij} = 1$.