A Conductance Electrical Model for Representing and Matching Weighted Undirected Graphs [*]

Manuel Igelmo
Universitat Politècnica de Catalunya (UPC)
migelmo@xtec.cat

Alberto Sanfeliu, Miquel Ferrer
Universitat Politècnica de Catalunya (UPC)
Institut de Robòtica i Informàtica Industrial (UPC–CSIC)
(sanfeliu,mferrer)@iri.upc.edu

Abstract

In this paper we propose a conductance electrical model to represent weighted undirected graphs that allows us to efficiently compute approximate graph isomorphism in large graphs. The model is built by transforming a graph into an electrical circuit. Edges in the graph become conductances in the electrical circuit. This model follows the laws of the electrical circuit theory and we can potentially use all the existing theory and tools of this field to derive other approximate techniques for graph matching. In the present work, we use the proposed circuital model to derive approximated graph isomorphism solutions.

1. Introduction

Graphs have been successfully applied in various fields such as chemistry and biochemistry, transportation, telephony and computers networks, speech recognition and computer vision [2]. In this paper we concentrate in those graphs coming from the field of computer vision. In this case, graphs have labeled nodes and/or edges [13] and they usually have a large number of nodes and/or edges. The methods for graph and sub-graph matching are based on enumerative techniques [15, 7], edit operations [14, 10, 9], spectral methods [16], expectation-maximization [12], random walks [5], genetics algorithms [3] and probabilistic approximations [17]. The time complexity in the enumerative

and edit operation methods is NP–complete while in the other methods it is polynomially bounded. Only in the enumerative solutions we have the exact solution, in the other cases we can get only graph and sub–graph matching approximations. The focus of this paper lies in providing a model for undirected weighted graphs and allowing them to apply theories, methods and procedures which are well known in the model. Specifically, we propose a circuit model without generators and formed exclusively with resistors. As an illustration of the model gives an approximate solution (through the circuit model) to graph isomorphism.

2. Conductance Electrical Model (CEM)

To represent weighted undirected graphs we propose to use a Conductance Electrical Model (CEM), extracted from the circuit theory, formed exclusively with resistors without generators. The two stages for constructing the model are explained in the following subsections.

2.1. Step function

The first stage is to assign conductance values (in what follows we will use the letter c for conductances instead of the usual g for which the latter is used to represent graphs) in the circuit from the edge weights. To this end, we define a step function $\phi(\omega_{ij}) = c_{ij}$ where ω_{ij} is the weight of the edge joining nodes i and j, and c_{ij} is the conductance (in siemens) of the resistor joining nodes i and j in the CEM. The step function must satisfy two conditions. First, the step function must be injective. This condition is absolutely necessary if

we want to retrieve a graph from its model. Second, $\phi(0) = 0$ must be. It is known that when two nodes are not connected by any edge, it corresponds to a zero in the adjacency matrix. These two nodes, in its circuit counterpart, have no resistor so that its conductance is zero. In summary, the step function can be any transformation which fulfils the following requirements:

1. ϕ is injective.
2. $\phi(0) = 0$.

Once the conditions are set, the decision to choose a step function depends strongly on the physical meaning of the weights of the edges and, therefore, it depends on the context of the problem. In other words, the step function is a parameter design.

2.2. Indefinite Conductance Matrix (ICM)

The second and last stage for the model is to obtain the Indefinite Conductance Matrix (ICM) in the usual sense as of circuit theory (for example [4]). It is well known that this matrix is obtained putting in row i and column j the value of the conductance (changing the sign) joining the nodes i and j for $i \neq j$; otherwise must be such that the row sum zero for $i = j$. As an example (see Fig. 1 and 2), if we have an adjacency matrix A of a graph g with degree N like,

$$
A = \begin{pmatrix}
0 & \omega_{12} & \cdots & \omega_{1N} \\
\omega_{12} & 0 & \cdots & \omega_{2N} \\
\vdots & \vdots & \ddots & \vdots \\
\omega_{1N} & \omega_{2N} & \cdots & 0
\end{pmatrix}
$$

then, the CEM model (C matrix) is obtained, for any step function, by the ICM, that is

$$
C = \begin{pmatrix}
\sum_{\substack{j=1 \\ j\neq 1}}^{N} c_{1j} & -c_{12} & \cdots & -c_{1N} \\
-c_{12} & \sum_{\substack{j=1 \\ j\neq 2}}^{N} c_{2j} & \cdots & -c_{2N} \\
\vdots & \vdots & \ddots & \vdots \\
-c_{1N} & -c_{2N} & \cdots & \sum_{\substack{j=1 \\ j\neq N}}^{N} c_{Nj}
\end{pmatrix}
$$

where it has already been taken into account that both matrix are symmetrics ($\omega_{ji} = \omega_{ij}$ and $c_{ji} = c_{ij}$). Note that the ICM has some similarities but it is not a Laplacian matrix in general. Only if $c_{ij} = \omega_{ij}$ ICM becomes a Laplacian.

2.3. Benefits of CEM

CEM is a linear model: One of the most natural representation for weighted graphs is through its
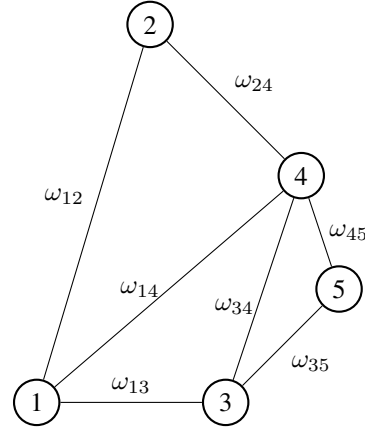

Figure 1. An example of a weighted undirected graph.
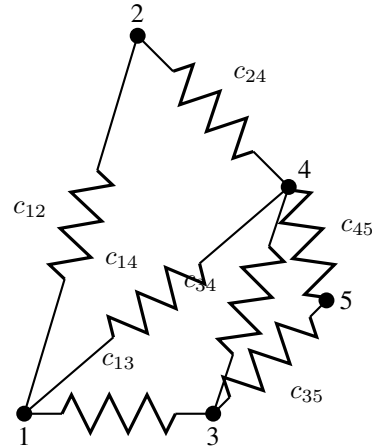

Figure 2. The CEM model of the example graph of the Fig. 1.

adjacency matrix. The CEM representation of a graph continues to be a matrix (C) but with the added value that is a linear application that means we can use all the tools of linear applications. The physical meaning of this matrix can be seen in the formula $I = CV$ where C is the CEM representation of the graph, V and I are the vector column of the voltages and currents respectively.

Computational complexity of CEM: The computational complexity to obtain the CEM is $O(N^2)$ where N is the graph size.

3. Approximate graph isomorphism

We will use the proposed model to approximately compute graph isomorphism between two given graphs. This procedure is carried out in three steps:

1. Transform both graphs g and h into their CEM models C_g and C_h.
2. Approximate CEM models C_g and C_h into star circuits C'_g and C'_h.
3. Decide whether there is an isomorphism comparing the star circuits C'_g and C'_h.

Using afore mentioned procedure, we cannot confirm exactly whether two graphs are isomorphic, because we transform a CEM model in a star circuit and we are not able to know if two different CEM models has the same star circuit. However it can be useful to perform a first filtering within a large database, keeping only those pairs of graphs which are said to be isomorphic and the apply an exact (or a more accurate) method to make a final decision.

3.1. Obtaining the CEM model

The two graphs to be compared g and h (with N nodes and M edges) are transformed into their CEM models C_g and C_h using the following step function,

$$c_{ij} = \phi(\omega_{ij}) = \begin{cases} 0; & \omega_{ij} = 0 \\ 1/\omega_{ij}; & otherwise \end{cases}$$

3.2. Computing the star circuit

Using circuit theory, we extracted from the original circuit the value of equivalent resistance seen from the $N(N-1)/2$ pairs of nodes: $r_{eq_{ij}}$. With this data we compute their star circuits, C'_g and C'_h. These circuits have $N+1$ nodes (the central node will be the reference) and N branches (resistors) that their behavior approximates the original CEM circuit (see Figure 3 for an example). With this new representation we will be able to perform approximate graph isomorphism.
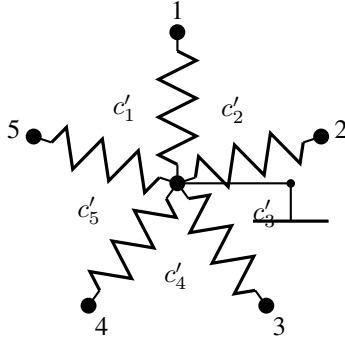

Figure 3. star circuit as approach of the example original circuit of the Fig. 2.

This will require, for example, that the equivalent resistance seen from each terminal pair $(r'_{eq_{ij}})$ is identical or approximate to the original circuit counterparts. In star circuit this value is simply

$$r'_{eq_{ij}} = r'_i + r'_j$$

where r'_i and r'_j are the resistors that join nodes i and j with the reference node respectively. So in the above equation the $r'_{eq_{ij}}$ is replaced by $r_{eq_{ij}}$. This leads to obtain a system of $N(N-1)/2$ linear equations with N unknowns (the N values of star circuit). The matrix form is

$$\begin{pmatrix} 1 & 1 & 0 & 0 & \cdots & 0 & 0 \\ 1 & 0 & 1 & 0 & \cdots & 0 & 0 \\ 1 & 0 & 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 1 & 0 & 0 & 0 & \cdots & 1 & 0 \\ 1 & 0 & 0 & 0 & \cdots & 0 & 1 \\ 0 & 1 & 1 & 0 & \cdots & 0 & 0 \\ 0 & 1 & 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 1 & 0 & 0 & \cdots & 1 & 0 \\ 0 & 1 & 0 & 0 & \cdots & 0 & 1 \\ 0 & 0 & 1 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 1 & 0 & \cdots & 1 & 0 \\ 0 & 0 & 1 & 0 & \cdots & 0 & 1 \\ 0 & 0 & 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 1 & \cdots & 1 & 0 \\ 0 & 0 & 0 & 1 & \cdots & 0 & 1 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & 1 & 1 \end{pmatrix} \begin{pmatrix} r'_1 \\ r'_2 \\ r'_3 \\ r'_4 \\ \vdots \\ r'_{N-1} \\ r'_N \end{pmatrix} = \begin{pmatrix} r_{eq12} \\ r_{eq13} \\ r_{eq14} \\ \vdots \\ r_{eq1,N-1} \\ r_{eq1,N} \\ r_{eq23} \\ r_{eq24} \\ \vdots \\ r_{eq2,N-1} \\ r_{eq2,N} \\ r_{eq34} \\ \vdots \\ r_{eq3,N-1} \\ r_{eq3,N} \\ r_{eq45} \\ \vdots \\ r_{eq4,N-1} \\ r_{eq4,N} \\ \vdots \\ \vdots \\ r_{eqN-1,N} \end{pmatrix}$$

This linear system has more equations than unknowns that usually have no solution unless there are sufficient linearly dependent equations. It is therefore an oversized system which we obtain an approximate solution that minimizes the euclidean norm. If we denote as A the matrix of coefficients, as b the column vector of independent terms, as r' the exact solution and as a $\hat{r}'$ the approximate solution that minimizes the euclidean norm then we have

$$r' \approx \hat{r}' = (A^t A)^{-1} A^t b$$

where the $(A^t A)^{-1} A^t$ is known pseudo-inverse of A. Remember that the resulting $\hat{r}'$ link the corresponding node in the original graph.

3.3. Comparison between the two star circuits

Once we have the values of N resistors, $\hat{r}'_i$, in the star circuit of the two graphs, we have to do node labeling assignment in order to find the isomorphism matching. If two star circuits are identical that means isomorphic,

they have the same N resistor values in the corresponding nodes. The node labeling assignment is done by arranging the values of the resistors using ascendant or descendant criteria. Then the comparison between both star circuits is done node by node, if they are the same, then both star circuits are identical. The computation complexity is linear with respect to the number of resistors, which is equal to the number of nodes N.

4. Experiments

In our experiments we used the graph-based version [11] of the COIL-100 database [8]. This dataset is composed by 7,200 graphs comming from 72 different objects. The nodes of the graphs in this database are the corners detected in the COIL objects using the Harris algorithm [6]. From these database we picked up randomly 100 graphs with the same degree, and we computed the graph isomorphism between all the possible unordered pairs $(100(100 + 1)/2 = 5,050)$. As a result we can say that al the isomorphic pairs were detected and no false positives were found. In short, the algorithm performed perfectly.

5. Conclusions and future research

The main contribution of this paper is a novel graph model, the CEM model, inspired on the circuit theory, which allows us to approximately test for graph isomorphism in only $O(N^2)$ time complexity, where N is the size of the graphs. Combining the CEM model with star circuit transformation, we are able to perform approximate graph isomorphism detection. Simple experiments done on a real database show that the method performs good in approximate graph isomorphism detection.

However, there are still some directions to extend this model. For instance, in the illustrative example presented in this paper we have chosen to approximate the original circuit by a star circuit, but this is not the only possibility, other settings fit provided that can detect special nodes to facilitate the comparison of approximate circuits. It should be emphasized to obtain the model itself is accurate in the sense that it is always possible to obtain the graph from the model, the approaches that enable the reduction of the problem are computational approaches that are performed on the model already obtained.

It is also of interest to use this model in other problems, such as graph distance measure, node reduction [1] and [18] and subgraph isomorphism.

References

[1] T. Asano and T. Hirata. Edge-deletion and edge-contraction problems. *Proceedings of the fourteenth an-*
nual ACM symposium on Theory of computing, pages 245–254, 1982.

[2] H. Bunke and A. Sanfeliu. *Syntactic and Structural Pattern Recognition - Theory and Applications*. World Scientific Publishing Co. Pte. Ltd.. Series in Computer Science - Vol. 7, Singapore, New Jersey, London, Hong Kong, ISBN 9971-50-566-5., 1990.

[3] A. D. J. Cross, R. C. Wilson, and E. R. Hancock. Inexact graph matching using genetic search. *Pattern Recognition*, 30(6):953–970, 1997.

[4] R. A. DeCarlo and P. Lin. *Linear Circuit Analysis*. Prentice Hall, 1995.

[5] M. Gori, M. Maggini, and L. Sarti. Exact and approximate graph matching using random walks. *Pattern Anal. and Mach. Intelligence*, 27(7):1100–1111, 2005.

[6] C. Harris and M. Stephens. A combined corner and edge detector. *Proceedings of the 4th Alvey Vision Conference*, pages 147–151, 1988.

[7] B. T. Messmer and H. Bunke. A new algorithm for error-tolerant subgraph isomorphism detection. *IEEE Transactions on Pattern Analysys and Machine Intelligence*, 20(5):493–504, May 1998.

[8] S. A. Nene, S. K. Nayar, and H. Murase. Columbia object image library (coil-100). *Technical Report CUCS-006-96*, February 1996.

[9] M. Neuhaus and H. Bunke. Edit distance-based kernel functions for structural pattern classification. *Pattern Recognition*, 39(10):1852–1863, 2006.

[10] M. Neuhaus and H. Bunke. Automatic learning of cost function for graph edit distance. *Information Sciences*, (177):239–247, 2007.

[11] K. Riesen and H. Bunke. Iam graph database repository for graph based pattern recognition and machine learning. *SSPR*, 2008.

[12] A. Robles-Kelly and E. R. Hancock. An expectation-maximisation framework for segmentation and grouping. *Image and Vision Computing*, (20):725–738, 2002.

[13] A. Sanfeliu, R. Alquézar, J. Andrade, J. Climent, Serratosa, F., and J. Vergs. Graph-based representations and techniques for image processing and image analysis. *Pattern Recognition*, (35):639–650, 2002.

[14] A. Sanfeliu and K. S. Fu. A distance measure between attributed relational graphs for pattern recognition. *IEEE Trans. Syst. Man Cybern.*, SMC-13(3):353–362, 1983.

[15] J. R. Ullman. An algorithm for subgraph isomorphism. *Journal of the Association for Computing Machinery*, 23(1):31–42, January 1976.

[16] S. Umeyama. An eigendecomposition approach to weighted graph matching problems. *IEEE Trans. PAMI 10*, pages 695–703, 1998.

[17] R. Wilson and E. R. Hancock. Structural matching by discrete relaxation. *Pattern Analysis and Machine Intelligence*, 19(6):634–648, 1997.

[18] M. Yannakakis. Node-and edge-deletion np-complete problems. *Proceedings of the tenth annual ACM symposium on Theory of computing*, pages 253–264, 1978.