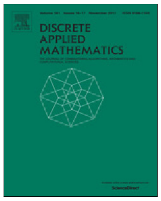
Contents lists available at ScienceDirect

Discrete Applied Mathematics

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



Fast algorithms for indices of nested split graphs approximating real complex networks

Irene Sciriha [a], Johann A. Briffa [b,*], Mark Debono [a]

[a] *Department of Mathematics, University of Malta, Msida MSD 2080, Malta*

[b] *Department of Commun. & Computer Engineering, University of Malta, Msida MSD 2080, Malta*

ARTICLE INFO

Article history:
Received 22 June 2017
Received in revised form 12 March 2018
Accepted 19 March 2018
Available online xxxx

Keywords:
Threshold graph
Nested split graph
Markov chains
Simulated annealing
Hamming distance
Shannon entropy
Randić index
Szeged index
Estrada index
Wiener index
Gutman graph energy
Resolvent energy

ABSTRACT

We present a method based on simulated annealing to obtain a nested split graph that approximates a real complex graph. This is used to compute a number of graph indices using very efficient algorithms that we develop, leveraging the geometrical properties of nested split graphs. Practical results are given for six graphs from such diverse areas as social networks, communication networks, word associations, and molecular chemistry. We present a critical analysis of the appropriate perturbation schemes that search the whole space of nested split graphs and the distance functions that gauge the dissimilarity between two graphs.

1. Introduction

The architecture of real-life networks may not be predictable and involves data which is sometimes difficult to manage. Algorithms known to date for the computation of many parameters associated with complex networks are often exponential-time, blocking further development. To overcome this problem, the strategy implemented here is to use the class of *nested split graphs* (NSG) since a graph in this class can provide a model network which is sufficiently close to a given real-world network and which enables easy computation. The class of NSGs provides an ideal candidate as these graphs are nice to work with, are easily stored, and their structured topology lends itself to polynomial time algorithms for the computation of certain of its invariants. Where this is not possible, their rich mathematical versatility may enable other methods to circumvent the problem.

Nested split graphs, also known as *threshold graphs*, form a subclass of split graphs in which the vertex set is partitioned into a clique (maximal complete subgraph) and a co-clique, that is an independent subset of vertices (with no edges between any pair of the independent vertices). Furthermore, NSGs display a structure in which the clique and the co-clique are partitioned into *cells* which form an *equitable vertex partition*, that is each vertex of a cell has the same number of neighbours in each of the cells. Moreover, the adjacencies among the cells impose a structure on the NSG that induces a partition of the vertices referred to as the *NSG partition*.

[*] Corresponding author.
E-mail addresses: irene.sciriha-aquilina@um.edu.mt (I. Sciriha), johann.briffa@um.edu.mt (J.A. Briffa).

https://doi.org/10.1016/j.dam.2018.03.054
0166-218X/© 2018 Elsevier B.V. All rights reserved.

The questions we ask are: (a) given a network G, is there a NSG $\hat{G}$ that is sufficiently close to G that can be used instead of G for the purposes of computing network parameters and spectral properties? and (b) how can computation time be reduced by drawing on the geometric properties of NSGs?

To search for the optimum $\hat{G}$, we devise two *distance functions*, the scaled walks and the spectral functions, f_{SW} and f_λ, respectively, with the objective to measure the rate of convergence of the original given network to an interim NSG G' picked by the algorithm considered. The techniques used are Markov chains and simulated annealing that enable the space of coded NSGs to be searched. Three perturbation schemes are used for each of the distance functions. These are the 'hamming', 'edge', and 'move' schemes, all of which will be explained in detail in Section 3.2. The process outputs the best network among all NSGs that can replace the given network, up to a prescribed tolerance, for the purpose of computing selected invariants associated with the original network.

The parameters to be computed using the different techniques are derived mainly from sociology, information theory and physical chemistry. They are Entropy, the Randić, Wiener, Szeged, Co-PI and Estrada indices, and also Gutman's Graph Energy and Resolvent Energy. The networks tested come from communication networks, word associations, and molecular chemistry.

When using Markov chains and simulated annealing to determine a NSG close enough to a given G, the problem centres on the best distance function and Markov chain perturbation to use for reasonable computation times and reliable values of graph invariants of the NSG to which the process converges. The different techniques are analysed to see which distance functions and perturbations correspond to optimized output and well behaved algorithms.

The paper will be organized as follows. In Section 2 we start by presenting various characterization of NSGs. This is followed in Section 3 by descriptions of the graph parameters to be estimated, and the distance functions to be used as a measure of closeness between the interim NSG G' in the simulated annealing process and the original graph G. The simulated annealing process is presented in Section 4, together with the perturbation schemes used to traverse the whole space of NSGs. In this section we also develop efficient algorithms to determine the graph parameters of Section 3 by taking advantage of the geometric properties of NSGs. Estimates of selected parameters of G are obtained by calculating them for $\hat{G}$, the limiting NSG. Practical results are given in Section 5 for a selection of six graphs, where we consider the effect of the simulated annealing parameters on the NSGs obtained, and also compare the indices for the limiting NSGs with those of the original graphs. In Section 6 pointers for further work from these seminal ideas are suggested.

2. Characterizations of a nested split graph

A complete graph on n vertices is K_n and a graph on n vertices with no edges is $\overline{K_n}$. The path and the cycle on n vertices are denoted by P_n and C_n, respectively. A disconnected graph consisting of r copies of a graph G is denoted by rG. The number of edges incident to a vertex v of a graph G on n vertices is the degree ρ_v. A graph has an isolated vertex if it has a vertex with no edges incident to it, while a dominating vertex corresponds to a vertex with edges to all the other vertices of the graph. A vertex v is a *duplicate* of a vertex u in a graph G if u and v are not adjacent and they have the same neighbouring vertices in G. Two vertices u and v are *co-duplicates* if they are adjacent and have the same neighbours.

Different characterizations of NSGs which we shall use in computing the different NSG parameters are now presented.

Theorem 1. *The following statements are equivalent for a graph G [18,22]:*

(i) *G is a NSG;*
(ii) *G is P_4, C_4 and $2K_2$ free;*
(iii) *G is constructed from K_1 by successive additions of an isolated or dominating vertex.*

From Theorem 1(iii), the following result gives the construction of a connected NSG from a binary code.

Proposition 1. *A connected NSG on n vertices is uniquely coded as a binary string of $(n-2)$ bits. The rth bit is 0 if the $(r+1)$th vertex added is an isolated vertex and it is 1 if it is a dominating vertex. This string, which is referred to as the* minimum representation *of the NSG, assumes that the first vertex in the construction is an isolated vertex and that the last vertex is a dominating vertex.*

Observe that $(n-2)$ bits are sufficient because in the canonical construction the first vertex is always isolated and the last vertex has to be dominating. It also follows from Theorem 1(iii) that each intermediate subgraph in the construction of the NSG G is itself a NSG. If the first bit of the encoding string (reading from the left) is 0, then the intermediate NSG G_2 on two vertices is $2K_1$, whereas if it is 1, then G_2 is K_2.

Definition 1. The *creation sequence* for a connected NSG on n vertices is the n-tuple $\mathbf{c} = (c_1, c_2, \ldots, c_n)$ with the first element $c_1 = 0$, the last element $c_n = 1$, and intermediate elements equal to the bits of the minimum representation.

It is convenient to make use of a more concise form of the creation sequence by encoding repetitions of successive bits.

Definition 2. The *compact creation sequence* for a connected NSG on n vertices is the r-tuple $\mathbf{a} = (a_1, a_2, \ldots, a_r)$, where $\sum a_i = n$, r is even, and for all i, $a_i \geq 1$ is the length of the ith successive sequence of entries in the creation sequence of the same value (i.e. with each entry equal to 0 or each entry equal to 1).

Please cite this article in press as: I. Sciriha, et al., Fast algorithms for indices of nested split graphs approximating real complex networks, Discrete Applied Mathematics (2018), https://doi.org/10.1016/j.dam.2018.03.054.