Generator Description

Kyle Nickerson
Tian Wang
Arshad Arafat

June 20, 2018

1 Problem Description

Instance: Two undirected graphs G and H and a natural number k.
Accept if: G and H have a common induced subgraph containing at least k nodes

2 Model

This section describes the model we will for generating the random graphs for our instances. Our graph generator will use a preferential attachment model - similar to the Barabási–Albert (BA) Model [1] - for generating random graphs. The original BA model describes how linear preferential attachment can produce graphs with power law degree distributions. The model used in our generator is a more general version of the BA model, which allows for sub-linear and super-linear preferential attachment [2]. The strength of the preferential attachment is an input parameter to our generator. This will allow us to explore how different types of preferential attachment influence the development of common subgraph structures.

3 Parameters

Our graph generator will take 4 parameters: m_0, m, n, and α, explained below. An instance of our problem $< G, H, k >$ can be described with the following parameters.

- m_{0G} is the initial number of nodes in G.
- m_G is the number of edges added with each additional node in G.
- n_G is the total number of nodes in G.
- α_G is a parameter indicating the strength of preferential attachment for G.

- m_{0H} is the initial number of nodes in H.
- m_H is the number of edges added with each additional node in H.
- n_H is the total number of nodes in H.
- α_H is a parameter indicating the strength of preferential attachment for H.
- k is the size of common subgraph we are looking for between G and H

4 Planned parameter regime

4.1 Motivation

We are most interested in examining how both the density of edges, and the strength of preferential attachment, impact the size of the largest common induced subgraph between two networks. Because of this, we are most interested studying the variation of the parameters k, m_G, α_G, m_H and α_H. In order to simplify the number of different conditions in the experiment, we are only planning on creating instances where G and H have the same number of nodes ($n_G = n_H$). Another simplification to parameters we will use, is to always set $m_{G0} = m_G$ and $m_{H0} = m_H$, which is commonly used with the BA model [1,2].

4.2 Planned Values

- $n_G = n_H = 50$
- $m_{0G} = m_G, m_{0H} = m_H, = \{1, 2, ..10\}$
- $\alpha_G, \alpha_H = \{0.0, 0.5, ..., 2.0\}$ Changes to alpha affect the shape of the degree distribution produced.
- $k = \{3, 4, 5..?\}$. We plan on determining the max value for k experimentally, because we do not know how big the common subgraphs will be on average. By increasing it from small to large, we can insure we have both yes and no instances of our problem.

4.3 Experiment Setup

We plan on first creating a set of graphs using the graph generator, using all combinations of the m and α parameters (likely multiple graphs for each combination of values). Then to create instances of our problem, we will consider all pairs of graphs in the set, and for each pair of graphs, we will consider multiple k values. This will let us observe how looking for larger graphs effect the execution time of the SAT solver, an its ability to find satisfying assignments.

4.4 Justification of Values

We choose to fix the number of nodes in the graphs as 50, because we have many other parameters we are more interested in exploring. The number of variables produced by our reduction, in terms of the above parameters, there will be $2 \cdot k \cdot (n_G + n_H) = 100 \cdot k$ variables in the CNF formula. This will give us a range of around 300 variables for our smallest instances, and depending on how large k gets, likely over 1000 variables for the largest. If we find this n value is to large for the solver, we will experiment will smaller values.

The m values we chosen to be in a similar range to those observed in real world networks [3]. Also as m increases, so will the number of clauses in our CNF formula. This will ensure we produce SAT instances with a wide range in the number of clauses.

The α values where chosen so that graphs with different network structures will be produced. The α values we have chosen will produce networks with the following topologies [2]:

- 0.0 - random
- 0.5 - random (weak preferential attachment)
- 1.0 - scale-free
- 1.5 - hub-and-spoke (multiple winners take all)
- 2.0 - hub-and-spoke (one or multiple winners take all)

5 References

[1] - Barabási, Albert-László Albert, Réka (October 1999). "Emergence of scaling in random networks" (PDF). Science. 286 (5439): 509–512.
[2] - Barabási, Albert-László. "Network Science." Retrieved from http://networksciencebook.com.
[3] - Newman, M. E. J. Networks : An Introduction. Oxford ; Toronto: Oxford UP, 2010.