Circuits as a Classifier for Small-World Network Models

A. Leist and K. A. Hawick
Institute of Information and Mathematical Sciences
Massey University – Albany, North Shore 102-904, Auckland, New Zealand
Email: {a.leist, k.a.hawick}@massey.ac.nz
Tel: +64 9 414 0800 Fax: +64 9 441 8181
February 2009

Abstract

The number and length distribution of circuits or loops in a graph or network give important insights into its key characteristics. We discuss the circuit properties of various small-world or scale-free network models generated with different small-world probability parameter values. The small-world properties usually manifest themselves in terms of reduced path-length properties or the set of inter-node distances present in a graph. We show how the number of circuits present can increase or decrease with a larger probability of small-world shortcut links applied, depending upon which model is used. Circuit properties are computationally expensive and we consider counting only a partial circuit distribution and thus being able to use circuits as a classifier for these models in practical cases.

Keywords: circuits; graphs; small-world; scale-free networks.

1 Introduction

Many important models of social and physical phenomena are based on graphs or networks. Small-world and scale-free networks have attracted a great deal of recent attention in the graph literature as their path-length distributions can be controlled in a useful manner employing an arc or edge rewiring probability parameter. The path-lengths are not however the only signature properties of small-world models, and in this paper we consider the circuit or loop properties of graphs and their potential role as a classifier to categorise network models.

“Given any two people in the world, person X and person Z, how many intermediate acquaintance links are needed before X and Z are connected?” This question encouraged Milgram in 1967 [1] to perform a now infamous experiment to see how quickly a letter could get from one side of the USA to a (random) recipient on the other side by only forwarding the letter to acquaintances known on a personal basis. What Milgram showed was that while we may maintain a small group of very good friends, we often have acquaintances who move in different social and professional circles from ourselves, thus the average number of hops that a letter might need to make to get from one person to another is statistically less than if it were simply handed to a random person in the hope that it might eventually arrive.

Milgram’s small-world is thus the phenomenon of being able to find common links between two seemingly disparate entities. In a graph context a small-world effect might reduce the average number of hops that it takes to get from one node to another node in the graph. In real life we can view such things as redundant high-speed Internet backbones that connect major cities as small-world short-cuts through the network: data is able to take a direct route to its destination without having to traverse many multi-hop local-area networks.

Although some doubts have been cast on the quantitative nature of Milgram’s work, the recent work by Watts and Strogatz [2] and by Newman [3] places the small world or network shortcut concept on a firmly quantitative foundation. In this paper we explore the implications of the small-world effect in a variety of graph and network models that might arise in physical and mathematical models.

The properties of networks can be used to assign them to one or more categories of complex networks, most interesting to us are the categories of small-world and scale-free networks which sometimes overlap. Many social networks have been shown to fit into either or both of these categories [4, 5], for example the network of scientific collaborations [6, 7] or even the structure...
of criminal organisations [8]. But complex networks with these properties can also be found in other disciplines. Metabolic networks [9–11] studied by biologists and computer networks like the World-Wide Web [12] or certain peer-to-peer overlay networks [13, 14] have been shown to share properties that are used to put them into either of these categories.

In order to be classified as a small-world network, the shortest paths between any two nodes—also called the mean geodesic distance—must scale logarithmically or slower with the network size for fixed degree \( k \) [15].

Small-world networks are highly clustered, like a regular graph, but retain the small mean geodesic distance characteristic of random graphs [16]. Scale-free networks, on the other hand, have a power-law degree distribution for which the probability of a node having \( k \) links follows \( P(k) \sim k^{-\gamma} \). The exponent \( \gamma \) typically lies between 2.5 and 3.0 [17,18].

Networks are commonly represented by graphs. A graph \( G = (V, E) \) consists of a set of \( V \) vertices and a set of \( E \) edges. The edges can be either undirected (bidirectional) or directed (unidirectional). Directed edges are also called arcs. The vertices represent the nodes of the network and the edges represent the connections between these nodes.

In section 2 we discuss algorithms that can be used to count the elementary circuits in a graph. Section 3 introduces four small-world or scale-free network models, which we analyse in the following sections. The results of this analysis are provided in section 4 and discussed in section 5 where we also offer some conclusions and suggested areas for further study of circuits.

2 Circuits Analysis

An elementary circuit is defined as a closed loop, visiting any given node only once, so there are no crossings. A number of different algorithms for counting the elementary circuits in a graph have been reported in the literature, but these generally either use infeasible amounts of memory or are time exponential [19, 20] with a time bound of \( O((N + e)(c + 1)) \).

We use two approaches to count the number of circuits in a graph. The first one uses a variation of Johnson’s algorithm [21] and counts all circuits of arbitrary lengths. For graphs of \( N \) vertices, \( e \) edges, \( c \) circuits and 1 fully connected component, Johnson’s algorithm is bounded in time by \( O((N + e)(c + 1)) \) and space bounded by \( O(N + e) \). Unlike Johnson’s algorithm our code copes with partially connected graphs without resorting to the need to treat each of the possible \( N_c > 1 \) components separately [22, 23]. This is still a highly expensive process since the number of circuits \( c \) itself grows very rapidly with \( (N, e) \). Some savings in the computation time can be made however, by recognising that we do not always require to know the full circuit-length distribution and so we can truncate the distribution and the enumeration algorithm accordingly. In this second approach, we only consider circuits of lengths 2 to 10. This allows us to analyse networks of larger size than the first approach.

The underlying data structure used to represent graphs in our implementations is based on arcs. A bidirectional edge is modelled by two arcs going in opposite directions. Every operation that adds, removes or rewire an edge modifies both of the arcs that represent this particular edge accordingly. The circuit counting algorithms operate on this data structure and count a circuit of length two for every pair of arcs used to represent an edge. This makes the implementations more flexible, but the implications have to be kept in mind when interpreting the results.

3 Network Models

We have analysed graphs generated with four different algorithms that are described in this section. Depending on the input parameters, the resulting networks all possess the characteristic properties of either small-world or scale-free graphs. All of them use bidirectional edges for the connections between the vertices.
Model 1 is a simple network model that initially creates an n-dimensional nearest-neighbours-only neighbours structure (e.g. a ring in one-dimension and a square lattice in two-dimensions). Every vertex has \( n \times 2 \) neighbours. In the next step, every edge is rewired with probability \( p \). The new destination vertex is chosen randomly from all the vertices in the network (excluding self-edges). See figure 1 for an example graph.

Model 2 is Watts’ and Strogatz’s \( \beta \)-model [16, 24]. The algorithm for this model starts with a perfect 1-lattice, in which every vertex is connected to its \( \frac{n}{2} \) nearest neighbours on either side. It then considers every vertex \( i \) in turn, along with the edge \( \{i, i+1\} \) that connects it to its nearest neighbour in a clockwise sense and rewires the edge randomly with probability \( p \). Once every vertex has been considered once, it does the same for the edges \( \{i, i+2\} \) and so on, until \( \frac{n}{2} \) turns have been completed. In the end, all edges have been considered exactly once and a fraction of them has been rewired to vertices chosen uniformly at random from the entire graph. Except for very small or large values of \( p \), the resulting networks show the characteristic small-world properties.

Model 3 is Barabasi’s and Albert’s gradual growth and preferential attachment model [17]. The algorithm starts with a small number of vertices, the fully connected core of the network, and adds a new vertex at every time step. A number of bidirectional edges connect the new vertex to the existing network. The neighbours are chosen based on their current degree, following the rich-get-richer principle which leads to an aristocratic network structure [25]. The resulting graphs follow a power-law degree distribution, have a small mean shortest path length and are always fully connected.

Model 4 is Watts’ \( \alpha \)-model [24]. This is a preferential attachment algorithm in which the propensity of a vertex \( i \) connecting to a vertex \( j \) increases with the number of neighbours shared among the two. Depending on the input parameter \( \alpha \), which regulates how strong the effect of this propensity is (the lower the value the stronger the effect, the higher the value the more random the resulting network becomes), it tends to generate strongly clustered small-world networks that may or may not be fully connected.

4 Results

In this section we present the outcomes of our network analysis. All simulations have been performed 50 times and the data points in the following graphs represent the mean values. Every data point also shows the standard deviation from the mean as error bars, although sometimes they are too small to be visible.

![Figure 2: Model 1 with network size \( N = 200 \) (one-dimensional). This graph illustrates the total number of circuits found for all circuit lengths that occur in the graph for varying values of the rewiring probability \( p \). For each dataset except the first one (\( p = 0.00 \)), every three successive values on the x-axis have been averaged to improve the readability of the graph. Furthermore, all error bars have been rescaled by a factor of 0.2 for the same reason. Although these deviations from the mean values indicate a distinct uncertainty of the results, the smooth curves lead us to believe that the averages are of value nevertheless. The peaks of the curves increase with \( p \) up to \( p = 0.9 \). Surprisingly, the curve and peak of \( p = 1.0 \) is similar to the curves of \( p = 0.7 \) and 0.8 and lower than \( p = 0.9 \).

The distributions of the number of circuits of different lengths for model 1 with one- and two-dimensions, as illustrated in figures 2 and 3, are similar except for \( p = 0.0 \). With no rewiring, all the circuits in the one-dimensional model are either of length 2 or 200, the minimum and maximum lengths for this graph instance. This is not the case in two-dimensions, where the curve of \( p = 0.0 \) appears similar to all the other probabilities. However, figure 4 shows that this is not the case, as it does not contain any circuits of odd length. Figure 5 shows how the total number of circuits changes with \( p \) for both cases.

Figure 8 shows that this shape of the circuit length distribution is not special to model 2, but can also be found in the scale-free model 3. It appears to be common for different types of networks.

As mentioned before, an edge is represented by two arcs in the underlying data structure. And even though the models operate on bidirectional edges, the circuits algorithm sees two arcs going in opposite directions instead of one bidirectional edge. Therefore, each of these edges is counted as a circuit of length 2. This is the reason why the number of loops of length 2 is constant in
Figure 3: **Model 1** with network size $N = 6 \times 6 = 36$ (two-dimensional). This graph illustrates the total number of circuits found for all circuit lengths that occur in the graph for varying values of the rewiring probability $p$.

Figure 4: This graph shows three of the datasets already illustrated in figure 3, but instead of taking the natural logarithm of the $y$-values, they were normalised so that the $y$-values add up to 1.0. Except for the first dataset ($p = 0.00$), all datasets have the same basic shape and only the peak is in slightly different positions on the x-axis. With $p = 0.00$ however, the results show a zig-zag-pattern because no circuits of odd length were found in the graph.

Figure 5: This graph illustrates how the total number of circuits changes with increasing $p$ for both the one- and two-dimensional cases of **model 1**. The degree $k = 4$ for $N = 6 \times 6$ and $k = 2$ for $N = 200$.

Figure 6: **Model 2** with network size $N = 1000$. This graph illustrates how the number of circuits of lengths 2-10 changes with the rewiring probability $p$.

Figures 6 and 11 show how two different small-world models behave with increasing randomness. The number of short circuits decreases considerably with increasing $p$ and $\alpha$ respectively (with the exception of circuit length 2). However, in model 2 the number of circuits of length 8 and greater increases after a short decrease for small $p$, whereas in model 4 it initially decreases slightly and then remains constant.

The degree $k$ in model 3 is approximately $2 \times$ the number of new edges added with every vertex. The fully connected core of $c$ vertices is the reason why $k$ is not exactly twice this number. The value of core size $c$ is kept equal to the number of edges added with every new vertex.
Both figures 8 and 9 show that the number of circuits of length 2 is slightly larger than that of circuits of length 3 and 4 when only 2 to 3 edges are added with every new vertex. The former graph illustrates this by the little dip in the first dataset for small circuit lengths, whereas the latter graph shows it in the crossover of the result values for circuits of length 2 to 4.

The algorithms for model 2, 3 and 4 allow to specify the degree of the final graphs. Figures 7, 9 and 10 illustrate how the number of circuits of different lengths changes with the degree for these network models. For the lowest measured degree $k = 2$ ($k \approx 2$ for model 3), no circuits containing more than 2 (model 3) or 3 (model 4) vertices exist in the graphs. For higher degrees, the
result values for all three models increase with the degree. One difference between the scale-free model and the small-world models exists though. With the exception of circuit length 2, for every measured data point in the scale-free model there are more circuits of length \( x + 1 \) than of length \( x \). In the small-world models, on the other hand, there is a crossover at \( k \sim 4 \). The result values are smaller the longer the circuit length for degrees smaller than this crossover value, but for higher degrees there are more longer circuits than shorter ones. Of course, this only holds up to the peaks illustrated in figures 2, 3 and 8, after which the number of circuits of length \( x + 1 \) is smaller than that for circuits of length \( x \).

**Figure 11:** Model 4 with \( N = 1000 \) and \( k = 10 \). This graph illustrates how the number of circuits of lengths 2-10 changes with the parameter \( \alpha \). Length 2 is not affected at all due to the reason described in the main text. For the other lengths, the results show that shorter circuits are more affected by the randomness of the network than longer ones. Even though the probability that the network is joined together into a single component increases with \( \alpha \) (the clustering decreases), the number of longer circuits is more consistent than the number of shorter circuits. This may be due to the fact that longer circuits benefit more when disjoined clusters merge together than shorter ones, and therefore they do not suffer as much from the decreasing clustering in the network.

Figure 12 compares how the number of circuits of length 10 changes with the network size in all four models.

### 5 Discussion and Conclusions

In the previous section we have shown how circuits can be used to illustrate differences or similarities between various network models. For instance, when comparing the number of circuits of different lengths found in both small-world and scale-free networks, these curves seem to typically peak in the range \( \frac{1}{2}N < \text{peak} < N \), where the network size \( N \) is the largest possible circuit.

The analysed small-world and scale-free models show different results in the number of circuits found in comparable graph instances. They also differ in the way the degree and size of the graphs influences these numbers. As far as the available data suggests, scale-free graphs contain more circuits than small-world graphs and this gap increases further with the network size.

These results suggest that measuring the circuits in a graph can provide valuable insights into its structure. The main problem, however, is the complexity of the circuit counting algorithms. Restricting the length of the circuits under consideration enables us to analyse larger graphs than we would otherwise be able to. And while this also restricts the way circuits can be used to classify the networks to a certain extent, shorter circuits still provide valuable information about the graph.

A combination of the two approaches may give even better results. Small instances of a certain graph type could be used to obtain information about the complete circuit distributions, while larger instances are
only analysed in a restricted way and then use the previously obtained distributions to draw conclusions about the numbers of longer circuits.

In summary, we have discussed the complexity of counting the circuits in a graph, described 4 small-world or scale-free network models and then analysed the usability of circuits as a classifier for these networks.

We believe that the number and length distributions of circuits can give valuable insights into the structure of networks. Further analysis of different network models will show in how far this assumption holds for other network types.

Acknowledgements

Thanks to H.A.James for helpful comments on this work and assistance in coding the circuits analysis program.

References

