An invariant of scale-free graphs

Alexei Vernitski

Abstract—In many applications (including biology and the study of computer networks) graphs are found to be scale-free. It has been argued that this property alone does not tell us much about the structure of the graph. In this paper, we introduce a numerical characteristic of a graph, which we call the astral index, and which can be calculated efficiently. We demonstrate that the Barabási-Albert algorithm for generating scale-free graphs produces not just scale-free graphs, but only scale-free graphs with a constant astral index. On some examples of biological graphs, we see that they not only are scale-free, but also share the value of the astral index with Barabási-Albert graphs. For comparison, we demonstrate that the Erdős–Rényi model for generating random graphs also generates only graphs with a constant astral index, whose value significantly differs from that of graphs generated by the Barabási-Albert algorithm.

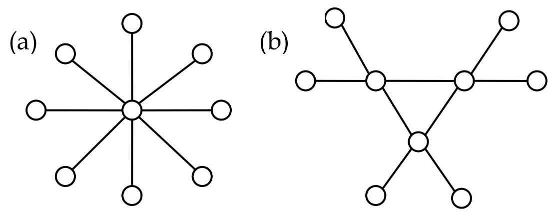
1 Types of Graphs in Applications

THIS paper contributes to the study of types of graphs occurring in real-life applications. It has been observed that the same type of graphs occurs surprisingly often, and a number of theoretical models characterizing such graphs have been suggested.

Before we consider such models, we should note that in this paper, we try to follow the standard graph-theoretical terminology such as 'graphs', 'vertices' and 'edges', even though some readers might be more used to the terminology from applied-sciences papers such as 'networks', 'nodes' and 'connections'. (At the same time, we shall avoid speaking about the 'graph of a function', so, the word 'graph' will mean only one thing within the paper.)

The most important theoretical model in recent publications is that of scale-free graphs; see, for instance, [1] or any of many papers on this topic by Barabási and other researchers. To define this model, let us introduce the degree distribution function. The *degree* of a vertex is the number of vertices adjacent to it. The *degree distribution function* of a graph is defined at each positive integer d as the number of vertices in the graph that have degree d.

Figure 2: examples of small graphs



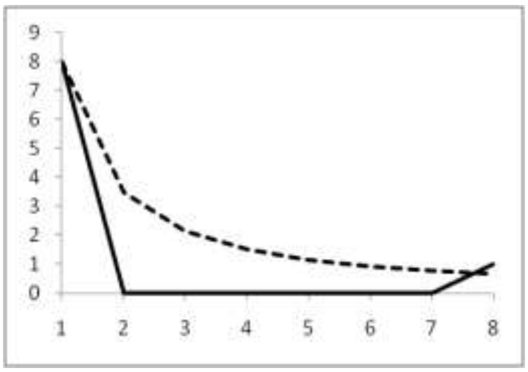
For instance, a *star graph* is a graph whose set of vertices is V, and whose edges connect one fixed vertex $u \in V$ with every other vertex $v \in V \backslash u$. The 9-vertex star graph presented on Figure 2 (a) has 8 vertices with degree 1 and 1 vertex with degree 8.

Another example is the graph on Figure 2 (b). It has 6 vertices with degree 1 and 3 vertices with degree 4. Thus, in both graphs there are fewer vertices with a relatively large degree and more vertices with a relatively small degree.

If we draw a diagram representing the degree distribution function of such a graph, we shall see that we have larger values of the function on the left of the diagram, and smaller values of the function on the right of the diagram. For instance, the solid line on the diagram on Figure 2 is the degree distribution function of the 9-vertex star graph on Figure 2 (a).

One can try to approximate the degree distribution function of a graph by a suitable smooth function, for instance, by a power function. For instance, the dashed line on this diagram is a function with the formula $y = 8x^{-1.2}$, and it could be considered as a rough approximation of the degree distribution function.

Figure 1: degree distribution function of a small scale-free graph. The horizontal axis represents the degree of a vertex. The vertical axis represents the number of vertices with this degree. The dashed line is an approximation by a power function.



Generalizing this example, a *scale-free graph* is a graph whose degree distribution function can be approximated by a power function, that is, a function of the form $y = kx^{-m}$, where k, m are some positive numbers. Also, in this case one says that this graph demonstrates a *power law* distribution.

It has been observed that a large number of graphs in various applications, ranging from biological graphs to the graphs of Internet connections, are scale-free.

Let us introduce an informal concept. Suppose a graph is like a scale-free graph in the sense that it has 'few' vertices with 'large' degrees and 'many' vertices with 'small' degrees, but does not necessarily demonstrate a power law distribution; in this case, we shall say that the graph is a declining-degree graph. For example, star graphs are declining-degree graphs; in the next section, we shall introduce astral graphs as an important generalization of star graphs, and they also are declining-degree graphs.

Random graphs (in the Erdős–Rényi sense) are graphs in which for each two vertices, the probability of having an edge between them is the same and does not depend on the presence or absence of other edges. Random graphs are not declining-degree graphs: their degree distribution is the Poisson distribution. This is a classical model in graph theory. We shall use random graphs to compare how our model works on scale-free graphs and on random graphs.

2 Astral graphs

In this section we introduce astral graphs; in the following sections, we shall see that they are a very convenient instrument for analyzing the structure of graphs, and especially the structure of declining-degree graphs.

Figure 3: a small astral graph

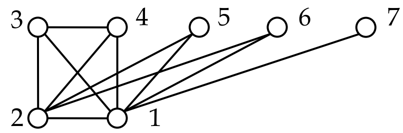


Table 1: the adjacency matrix of the astral graph shown on Figure 3

	1	2	3	4	5	6	7
1	0	1	1	1	1	1	1
2	1	0	1	1	1	1	0
3	1	1	0	1	0	0	0
4	1	1	1	0	0	0	0
5	1	1	0	0	0	0	0
6	1	1	0	0	0	0	0
7	1	0	0	0	0	0	0

Note that in the model considered in this paper, all graphs are undirected without loops. If necessary, it is possible to generalize the model to directed graphs.

We say that a graph is *astral* if there is a linear ordering of the set of its vertices such that whenever there is an edge between vertices a and b, there is also an edge between all vertices c and d such that $c \le a$ and $d \le b$. For example, a small graph presented on Figure 3 is astral, as can be seen easily if we impose on it a numbering of vertices as shown on the diagram.

It is convenient to think about this definition in terms of the adjacency matrices of graphs. If vertices of the graph are in a suitable order, the adjacency matrix of an astral graph has a special form as shown in the example in Table 1. That is, whenever we have 1 in a certain cell of the table, we also have 1 in all cells to the left and above this cell. (Note that the matrix is symmetric because the graph is undirected, and it has 0s on the diagonal because the graph has no loops.)

We call astral graphs this name because they are generalizations of star graphs. Indeed, every star graph is an astral graph. Let us consider a star graph whose set of vertices is V, and whose edges are the edges between a certain fixed vertex $u \in V$ and every other vertex $v \in V \setminus u$. To see that this star graph is an astral graph, it is sufficient to consider a linear order on V such that u is the smallest vertex relative to this order. Alternatively, if we consider adjacency matrices, if we consider a star graph, its adjacency matrix, with a suitable ordering of vertices, contains 1s in the first row and the first column, and 0s everywhere else.

3 Degree distribution in astral graphs

Sparse connected astral graphs are declining-degree graphs. One way of looking at this fact is by looking at the first and the last vertices in the ordering in the definition of the astral graph. Indeed, a few vertices at the beginning of the ordering are adjacent to every or almost every other vertex of the graph, and, on the other hand, comparatively many vertices at the end of the ordering have only one or a few neighbors. For example, suppose we consider any connected astral graph with 100 vertices and 300 edges. It can be calculated that in such a graph, between 3-10 vertices have degree 50 or greater, and between 84-96 vertices have degree 3 or less.

Having said that sparse astral graphs are declining-degree graphs, it is important to note that within this restriction, they can approximate any degree distribution. For instance, it is easy to construct astral graphs which are scale-free, or which have the exponential law distribution. Because astral graphs are independent from any particular degree distribution, they can be used to characterize graphs in applications in a way which is complementary to the ones which are used now and which are based on degree distributions.

4 Decomposing into astral graphs

Suppose we consider a graph with the set of vertices V and with the set of edges E. By *decomposing* the graph we mean that we find some graphs, say, m graphs $G_1, \dots, G_m$, whose sets of vertices are subsets of V, whose sets of edges $E_1, \dots, E_m$ are subsets of E, and such that $E = E_1 \cup \dots \cup E_m$.

Decomposing a graph into simple graphs is a standard instrument of graph theory. Moreover, decomposing a graph into star graphs (or, to be more precise, star forests) has been considered; this topic in graph theory is known as *star arboricity*, see, for instance, [1]. However, to the best of our knowledge, decomposition of graphs has never been used for classification of scale-free graphs.

The main new construction we are introducing is *decomposing a graph into astral graphs*. There are many ways of decomposing a graph into astral graphs. For instance, one trivial way of doing this is to decompose a graph into a union of its edges, since every edge by itself is an astral graph. What we are interested in is decomposing a graph into a small number of large astral graphs. However, doing this in an optimal way is a complicated and computationally infeasible task. Instead, we have developed a fast heuristic algorithm.

The algorithm that we use for decomposing a graph into astral graphs is presented as Algorithms 1 and 2 below. It is merely a greedy algorithm which tries to find a large astral subgraph of the graph (this part of the process is Algorithm 1), extracts it, and continues doing this again until a complete decomposition is achieved.

Let us clarify some terminology. When we consider a star graph, we refer to its vertices with degree 1 as *leaves*. The word *largest* in relation to a subgraph refers to the number of vertices.

Algorithm 1: finding a large astral subgraph

Start with the original graph G. Find a largest substar S_1 of G. Then find a largest substar S_2 of G such that all vertices of S_2 are leaves of S_1. Working iteratively, find a largest substar S_{i+1} of S_i such that all vertices of S_{i+1} are leaves of S_i. Continue until you reach a graph S_i such that there are no stars whose vertices are leaves of S_i. Then consider the (necessarily astral) subgraph A of G which is the union of all stars S_i.

(For example, if we apply this algorithm to the graph on Figure 3, then first it will find a star connecting 1 to leaves 2, 3, 4, 5, 6, 7; then it will find a star that con connecting 2 to leaves 3, 4, 5, 6; then it will find a one-edge star connecting 3 to 4; then it will stop.)

Algorithm 2: decomposing into astral graphs

Start with the original graph G. Find in it an astral subgraph A_1, using Algorithm 1. Extract the edges of A_1 from G, thus producing a graph G_1. Working iteratively, find in G_i an astral subgraph A_{i+1}, using Algorithm 1. Extract the edges of A_{i+1} from G_i, thus producing a graph G_{i+1}. Continue until you reach a graph G_i in which there are no edges.

One characteristic of the algorithm might deserve discussion. As you see, this algorithm contains a certain degree of randomness. Indeed, it relies on finding a largest subgraph with certain properties, and in principle, there might be more than one of such subgraphs. The choice between these subgraphs is merely a part of implementation of the algorithm and is a factor which is external and random with respect to the graph. Accordingly, it might be possible that the decompositions generated by the algorithm might vary, and might not always contain the same number of astral graphs. However, fortunately, this variability is only minor; in our experiments, the following situation is typical. If one considers a sparse graph with 1000 vertices (generated by the Barabási-Albert model with the edge/vertex ratio 1, see definitions in the following sections) then such a graph will be decomposed by our algorithm, on average, into 304.7 astral graphs, and the standard deviation of this number will be just 1.0.

5 Astral decomposition number of scale-free graphs: A small example

For brevity, let us call the number of astral graphs into which a graph can be decomposed (by the algorithm in the previous section) the *astral decomposition number* of the graph.

The main result of this paper is that the astral decomposition number of a graph turns out to be an important structural characteristic of a graph. We shall see this both on examples of graphs from theoretical models and on examples of graphs from applications.

One typical example is the following (except that the graph in consideration is relatively small – only 106 vertices and 96 edges).

Let us consider a graph $YeastSmall$ (protein interaction in yeast) studied in [3], see page 803; it can be found at the URL [4]. Suppose we consider $YeastSmall$ as an undirected graph without loops, as we do in the model described in this paper. Then $YeastSmall$ is a scale-free graph: indeed, its degree distribution can be approximated well by a function $y = 30x^{-1.2}$.

The astral decomposition number of $YeastSmall$ is 29 (We have said above that there is a small degree of randomness in the algorithm. However, this randomness does not have much effect; for example, $YeastSmall$ is always decomposed by our algorithm into exactly 29 graphs.)

Now let us compare the astral decomposition number of $YeastSmall$ with the astral decomposition numbers we get from theoretical models.

The Barabási-Albert model (also known as the 'preferential attachment' model) is the best-known algorithm for producing scale-free graphs [1]; see Appendix A for a short description of the algorithm. Let us call graphs produced by this model *Barabási-Albert graphs*. Suppose we use the Barabási-Albert model to produce scale-free graphs with approximately the same number of vertices and the same number of edges as $YeastSmall$, and then use our algorithm to decompose these graphs into astral graphs. Then we see that the astral decomposition number of Barabási-Albert graphs, is, on average, 29.3, with the standard deviation 2.7. Thus, surprisingly, we see that all Barabási-Albert graphs have approximately the same astral decomposition number. The astral decomposition number 29 of $YeastSmall$ is consistent with the astral decomposition number $29.3 \pm 2.7$of Barabási-Albert graphs. Thus, not only the property of being scale-free, but also the astral decomposition number shows us that the graph $YeastSmall$ is similar to the graphs generated by the Barabási-Albert model.

For comparison, suppose we use the Erdős–Rényi model (see Appendix B for its short description) to produce random graphs with the same number of vertices and the same number of edges as $YeastSmall$, and then use our algorithm to decompose these graphs into astral graphs. Let us call graphs produced by the Erdős–Rényi model *Erdős–Rényi graphs*. Then we see that the astral decomposition number of Erdős–Rényi graphs, is, on average, 41.0, with the standard deviation 2.2. Thus, surprisingly, we see that all Erdős–Rényi graphs have approximately the same astral decomposition number. The astral decomposition number 29 of $YeastSmall$ is four standard deviations away from the astral decomposition number 41.0 ± 2.2 of Erdős–Rényi graphs. Thus, we can conclude that the graph $YeastSmall$ is not a random graph.

6 Astral Index

Whereas the number of astral graphs in the decomposition of a graph is a valid research instrument, as shown in the previous section, we can improve this characteristic further. Let us define the *astral index* of a graph G as the number of astral graphs in its decomposition into astral graphs, divided by the number of vertices in G. By the *edge/vertex ratio* we shall mean the ratio of the number of edges to the number of vertices in a graph.

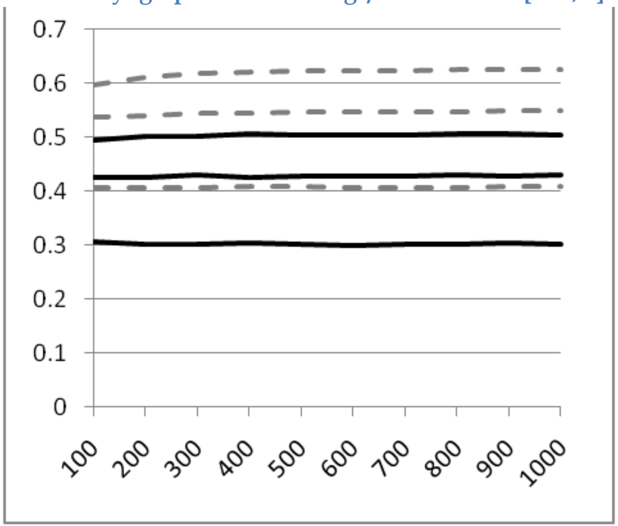
The introduction of the astral index is justified by a surprising fact that the astral index of Barabási-Albert graphs does not depend on the size of graphs, but only on the edge/vertex ratio. This fact makes the astral index a very convenient instrument when we want to decide whether graphs in applications are or are not like Barabási-Albert graphs.

This observation is presented on Figure 4.

The nearly horizontal line at the height 0.3 [at the height 0.42, at the height 0.5] shows the average astral index of Barabási-Albert graphs with the number of vertices ranging from 100 to 1000, and the edge/vertex ratio equal to 1 [equal to 2, equal to 3].

(It should be noted that to generate Barabási-Albert graphs, we use the Barabási-Albert model in its original form, and this algorithm has a degree of randomness and does not allow one to generate graphs with precisely a given number of edges. We approximate a desired edge/vertex ratio by using the model with parameters $m = 1,2,3$ and $m_0 = m + 1$. This is a good approximation, but the edge/vertex ratio of the graphs in consideration is, in fact, slightly less than stated in the previous paragraph.)

Figure 4: astral indices of graphs of varying size generated by two models. The horizontal axis shows the number of vertices in a graph. The vertical axis shows the astral index. Solid lines correspond to Barabási-Albert graphs with the edge/vertex ratio 1 [or 2, 3]. Dashed lines correspond to Erdős–Rényi graphs with the edge/vertex ratio 1 [or 2, 3].



For comparison, the nearly horizontal line at the height 0.4 [at the height 0.54, at the height 0.62] shows the average astral index of Erdős–Rényi graphs with the number of vertices ranging from 100 to 1000, and the edge/vertex ratio equal to 1 [equal to 2, equal to 3]. Thus, the astral index of Erdős–Rényi graphs also does not depend on the size of graphs, but only on the edge/vertex ratio.

Comparing graphs with edge/vertex ratio equal to 1, 2 and 3, we see that as the edge/vertex ratio becomes higher, the astral index grows. This trend would continue if we increased the edge/vertex ratio further. However, one is not likely to meet graphs with high edge/vertex ratio in applications.

It is useful to have in mind that astral index cannot take arbitrary values; one can prove that the astral index of every graph lies in the range between 0 and 1.

7 Astral Index of Scale-Free Graphs: Larger Examples

In this section we consider examples of one biological graph and two non-biological graphs and discover that only the biological graph has a value of astral index coinciding with that predicted by the Barabási-Albert model.

Let us consider a graph $YeastLarge$ (protein interaction in yeast), available at the URL [5]. This graph is scale-free and is studied as a scale-free graph in [6]. The graph $YeastLarge$ has 1870 vertices and 2845 edges, and its edge/vertex ratio is 1.52. Its astral index is 0.36. The Barabási-Albert model in its original form does not allow one to generate Barabási-Albert graphs with this ratio. However, looking at the results in the previous section, it does not seem unreasonable if we approximate the dependency of the astral index on the edge/vertex ratio between 1 and 2 by a linear function. Then we can suppose that the hypothetical Barabási-Albert graphs with the edge/vertex ratio approximately 1.5 should have the astral index approximately 0.36. This number is consistent with the astral index of the graph $YeastLarge$.

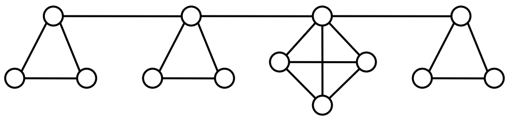
For comparison, let us consider two graph having a completely different, non-biological nature – a graph $PowerGrid$ (a part of the USA power grid) in [7] and $Internet$ (a snapshot of the Internet) accessible on [8]. The graph $PowerGrid$ has 4941 vertices and 6594 edges. The edge/vertex ratio of $PowerGrid$ is 1.33, which is similar to the edge/vertex ratio of $YeastLarge$. The degree distribution function of the graph $PowerGrid$ is similar to the degree distribution function of the graph $YeastLarge$: indeed, both can be approximated by a function with a form $y = kx^{-2.3}$, with different values of k. The graph $Internet$ has 22963 vertices and 48436 edges, thus, its edge/vertex ratio is 2.11; this is somewhat higher than for the other two graphs, and, as discussed in the previous section, if $Internet$ was like a Barabási-Albert graph, one would expect a slightly larger astral index for this graph than for the other two. The degree distribution of $Internet$ can be approximated well by a function

$y = kx^{-2.18}$, for some k. Thus, from the point of view of being scale-free, these graphs have approximately the same structure. However, let us look at their astral indices. The astral index of $PowerGrid$ is 0.44, significantly higher than approximately 0.36, which is the astral index of $YeastLarge$ and what could be expected from the Barabási-Albert model. As to $Internet$, its astral index is 0.14. This is abnormally small compared with approximately 0.42, as could be expected from the Barabási-Albert model for graphs with the edge/vertex approximately 2. Thus, we need to conclude that despite being scale-free, graphs $PowerGrid$ and $Internet$ differ considerably from Barabási-Albert graphs.

8 Scale-free vs astral index

What is shown on real-life examples in the previous section can be reinforced by some theoretical examples. It is tempting to misinterpret the main results of this paper in the following way. "Suppose G is a graph with the edge/vertex ratio 1 [or 2, or 3]. Then G is scale-free if and only if its astral index is approximately 0.3 [or 0.42, or 0.5]." Such a statement would be wrong in both directions. Indeed, there are scale-free graphs with the edge/vertex ratio 1 whose astral index is far from 0.3, and, on the other hand, there are graphs with the edge/vertex ratio 1 and the astral index is 0.3 which are far from being scale-free.

Figure 5: a non-scale-free graph with a typical astral index of Barabási-Albert graphs



Indeed, for instance, it is easy to construct examples of astral graphs which are scale-free. The astral decomposition number of such graphs is, obviously, 1, hence, their astral index is close to 0. Thus, there are examples of scale-free graphs with 'atypical' values of the astral index.

On the other hand, let us consider graphs which consist entirely or almost entirely of small complete graphs K_3 and K_4, like the graph on Figure 5.

Such graphs have the astral index close to 0.3, but they are not scale-free; they are not even declining-degree graphs.

So, we have to conclude that, on the one hand, in general, the properties of being scale-free and of having a particular value of astral index are not necessarily related; on the other hand, it would not be unreasonable to conjecture that graphs in a particular type of applications might be all scale-free and also all have a particular value of astral index.

9 Summary of results

It is well-known that very often graphs in applications are scale-free (or, at least, declining-degree graphs).

An algorithm known as the Barabási-Albert model generates examples of scale-free graphs.

In this paper we introduce a numerical characteristic of graphs which we call astral index; an astral index can range from 0 to 1 (in sparse graphs in applications, we are likely to see values of astral index between approximately 0 and 0.5). The astral index can be calculated efficiently, even for large graphs.

We see that all graphs generated by the Barabási-Albert model share approximately the same value of astral index. To be more precise, graphs generated by the Barabási-Albert model with a given number of vertices and a given number of edges share approximately the same value of astral index.

Moreover, we notice that the value of astral index of Barabási-Albert graphs does not depend on their size, but only on their edge/vertex ratio. Namely, Barabási-Albert graphs with the edge/vertex ratio equal to approximately 1 [or 2, or 3] have the value of astral index equal to approximately 0.3 [or 0.42, or 0.5].

On both theoretical and real-life examples, we see that scale-free graphs don't have to have a particular value of astral index. Therefore, we have to conclude that the Barabási-Albert model is not good at generating arbitrary scale-free graphs, but it only generates scale-free graphs with a certain constant value of astral index. We conjecture that the Barabási-Albert model is 'natural' in the biological context, because the value of astral index of graphs generated by this model coincides with the value of astral index we observe in examples of scale-free graphs in some biological applications.

For comparison, we consider graphs generated by the Erdős–Rényi model. We see that graphs generated by the Erdős–Rényi model with a given number of vertices and the same number of edges share approximately the same value of astral index.

Moreover, we notice that the value of astral index of Erdős–Rényi graphs does not depend on their size, but only on their edge/vertex ratio. Namely, Erdős–Rényi graphs with the edge/vertex ratio equal to approximately 1 [or 2, or 3] have the value of astral index equal to approximately 0.4 [or 0.54, or 0.62].

Comparing these two models, we see that the astral index of Barabási-Albert graphs is considerably different from that of Erdős–Rényi graphs.

As to scale-free graphs in applications, we see that for some of them, the astral index is consistent with the Barabási-Albert model, whereas for some others, it might be considerably smaller.

Overall, we see that the astral index of graphs might be a useful additional characteristic when analyzing the structure of graphs in biological and other applications:

- It provides a simple numerical measure of whether a graph is more like a scale-free graph or like a random graph.
- It can reveal whether a scale-free graph is like a graph generated by the Barabási-Albert model or not.

10 Novelty

Our approach is novel: we introduce astral graphs, which are a new concept, and use decomposition into astral graphs as a method of understanding the structure of the graph.

Our approach might be more convenient than other techniques suggested recently to improve our understanding of the structure of graphs in applications.

For instance, motifs are small subgraphs (e.g. triangles) of a graph. When a particular motif is found in a graph more often than would be expected if the graph was random, this tells us something about the structure of the graph. Developments in this direction of research include considering larger motifs, see, for instance, [9]. However, motifs remain very small graphs, whereas we can consider astral graphs of an arbitrarily large size, if necessary.

Subgraphs with certain properties are found in graphs, for example, cliques; see, for instance, [10]. Finding such subgraphs is computationally intractable.

Clusters are found in graphs using various methods; see, for instance, [10]. However, declining-degree graphs, due to their nature, don't necessarily decompose into clusters.

A semi-bipartite graph is a graph whose set of vertices is $V \cup W$, and such that all (or almost all) pairs of vertices from V are connected with an edge, whereas no vertices from W are connected with an edge [12]. Astral graphs are semi-bipartite; thus, semi-bipartite graphs could be considered as a partial step in the direction of research considered in this paper. However, the model of semi-bipartite graphs is too general and does not immediately provide us with any useful characteristics of the graph.

None of these approaches aims at developing a unified numerical characteristic which can be calculated efficiently and which can tell between graphs coming from various applications or generated by various models. This is what we have achieved in this paper.

11 Representing graphs in the computer

It might be useful to describe how we represented graphs in the computer to achieve the optimal time-memory trade-off when calculating the astral index of graphs.

Since these graphs are sparce, it is not reasonable to represent them by the adjacency matrix; instead of this, it is natural to represent a graph as a set of its vertices, and with each vertex, to store the set of vertices adjacent to this vertex. (The word 'set' is used here informally and does not imply a particular implementation.) Let us look at the specific data types used.

How adjacency sets are stored

Let N be the number of vertices in the graph. We assume that vertices are labeled by binary strings of length up to $\log N$. This is a suitable situation for using a special data type known as *trie*; see, for instance, [13] or [14].

The adjacency set of a given vertex v (that is, the set of vertices adjacent to v) is represented as a trie.

Let d be the degree of v. Thus, the adjacency set of v has size up to $O(d \log N)$, and deciding whether a vertex is adjacent to v takes time up to $O(\log N)$. More importantly, finding the degree of a vertex is fast and takes time up to $O(d \log N)$. Note that we cannot pre-calculate and store degrees of vertices because as a part of Algorithm 1, we need to be able to calculate degrees of vertices not only within the whole graph, but also within any given subgraph.

How vertices are stored

The set of all vertices is also represented as a trie, which has the same structure as the adjacency list of an individual vertex. Thanks to this, the adjacency set of a given vertex has the same data type as the set of all vertices; this arrangement considerably simplifies the implementation of Algorithm 1.

Thus, the whole graph is implemented as 'a trie of tries'. Namely, the set of vertices is a trie such that with each vertex (represented by a node of the trie), a trie can be stored which represents the adjacency set of the vertex. To our knowledge, this implementation of sparse graphs is new.

Let a be the average degree of a vertex in the graph. Thus, the whole representation of the graph has size up to $O(aN \log N)$. Deciding whether two vertices are adjacent takes time up to $O(\log N)$. Adding or removing an edge also takes time up to $O(\log N)$.

Appendix A: Barabási-Albert model

Start with a small complete graph of size m_0. Acting recursively, add a new vertex u and connect it to up to m existing vertices, using the following probabilistic rule: the probability of creating an edge between u and one of existing vertices v should be directly proportional to the current degree of v. Continue adding new vertices, as described above, until the desired number of vertices is reached.

(The edge/vertex ratio of the obtained graph will be approximately equal to m.)

Appendix B: Erdős–Rényi model

Start with an empty graph with as many vertices as required. Acting recursively, choose two vertices at random and connect them with an edge. Continue adding new edges, as described above, until the desired number of edges is reached.

Acknowledgment

The author wishes to thank Dr Andrew Harrison and Dr Edd Codling for their advice.

References

[1] A.-L. Barabási and R. Albert. Emergence of scaling in random networks. Science 286: 509–512.

[2] N. Alon, C. McDiarmid and B. Reed, Star arboricity, *Combinatorica*, Volume 12, Number 4, 1992.

[3] Lee, T.I., Rinaldi, N.J., Robert, F., Odom, D.T., Bar-Joseph, Z.,

Gerber, G.K., Hannett, N.M., Harbison, C.R., Thompson, C.M., Simon I., Zeitlinger J., Jennings, E.G., Murray, H.L., Gordon, D.B., Ren, B., Wyrick, J.J., Tagne, J., Volkert T.L., Fraenkel, E., Gifford D.K., and Young, R.A., Transcriptional Regulatory Networks in Saccharomyces cerevisiae. Science 298: 799-804 (2002).

[4] http://jura.wi.mit.edu/young_public/regulatory_network/reg-reg_ordered.xls

[5] http://www.nd.edu/~networks/resources/protein/bo.dat.gz

[6] H. Jeong, S. P. Mason, A.-L. Barabási and Z. N. Oltvai, Lethality and centrality in protein networks, *Nature* 411, 41-42 (3 May 2001).

[7] Watts, D.J.; Strogatz, S.H. (1998). Collective dynamics of 'small-world' networks, *Nature* 393 (6684): 409–10.

[8] http://www-personal.umich.edu/~mejn/netdata/

[9] J. A. Grochow, M. Kellis, Network Motif Discovery Using Subgraph Enumeration and Symmetry-Breaking, Research in Computational Molecular Biology, Lecture Notes in Computer Science 4453, 92-106, 2007.

[10] B. Balasundaram, S. Butenko and S. Trukhanov, Novel Approaches for Analyzing Biological Networks, Journal of Combinatorial Optimization, 23-39, Volume 10, Number 1 / August, 2005.

[11] P. Pons and M. Latapy, Communities in Large Networks, Journal of Graph Algorithms and Applications, vol. 10, no. 2, pp. 191–218 (2006).

[12] Xu, Kai and Williams, Rohan and Hong, Seok-Hee and Liu, Qing and Zhang, Ji (2009) Semi-bipartite graph visualization for gene ontology networks. In: 17th International Symposium on Graph Drawing (GD'09), 22-25 Sept 2009, Chicago, USA.

[13] K. Mehlhorn, *Data Structures and Algorithms 1: Sorting and Searching*, Springer-Verlag, Berlin, 1984.

[14] M. T. Goodrich, R. Tamassia, *Algorithm Design: Foundation, Analysis and Internet Examples*, Wiley, 2002.

Alexei Vernitski is a Senior Lecturer in the Department of Mathematical Sciences at the University of Essex, UK