Mining the inner structure of the Web graph

2008 J. Phys. A: Math. Theor. 41 224017

(http://iopscience.iop.org/1751-8121/41/22/224017)

IOP PUBLISHING JOURNAL OF PHYSICS A: MATHEMATICAL AND THEORETICAL

J. Phys. A: Math. Theor. **41** (2008) 224017 (12pp) doi:10.1088/1751-8113/41/22/224017

Mining the inner structure of the Web graph*

Debora Donato[1], **Stefano Leonardi**[2], **Stefano Millozzi**[2] **and Panayiotis Tsaparas**[3]

[1] Yahoo! Research, Barcelona, Spain
[2] Sapienza Universitá di Roma, Italy
[3] University of Helsinki, Finland

E-mail: debora@yahoo-inc.com, leon@dis.uniroma1.it, millozzi@dis.uniroma1.it and tsaparas@cs.helsinki.fi

Received 8 October 2007, in final form 20 December 2007
Published 21 May 2008
Online at stacks.iop.org/JPhysA/41/224017

Abstract

Despite being the sum of decentralized and uncoordinated efforts by heterogeneous groups and individuals, the World Wide Web exhibits a well-defined structure, characterized by several interesting properties. This structure was clearly revealed by Broder *et al* (2000 Graph structure in the web *Comput. Netw.* **33** 309) who presented the evocative *bow-tie* picture of the Web. Although, the bow-tie structure is a relatively clear abstraction of the macroscopic picture of the Web, it is quite uninformative with respect to the finer details of the Web graph. In this paper, we mine the inner structure of the Web graph. We present a series of measurements on the Web, which offer a better understanding of the individual components of the bow-tie. In the process, we develop algorithmic techniques for performing these measurements. We discover that the scale-free properties permeate all the components of the bow-tie which exhibit the same macroscopic properties as the Web graph itself. However, close inspection reveals that their inner structure is quite distinct. We show that the Web graph does not exhibit self similarity within its components, and we propose a possible alternative picture for the Web graph, as it emerges from our experiments.

PACS number: 39.75.−k

(Some figures in this article are in colour only in the electronic version)

* Partially supported by the EU under contract 001907 (DELIS) and 33555 (COSIN), and by the Italian MIUR under contract ALINWEB.

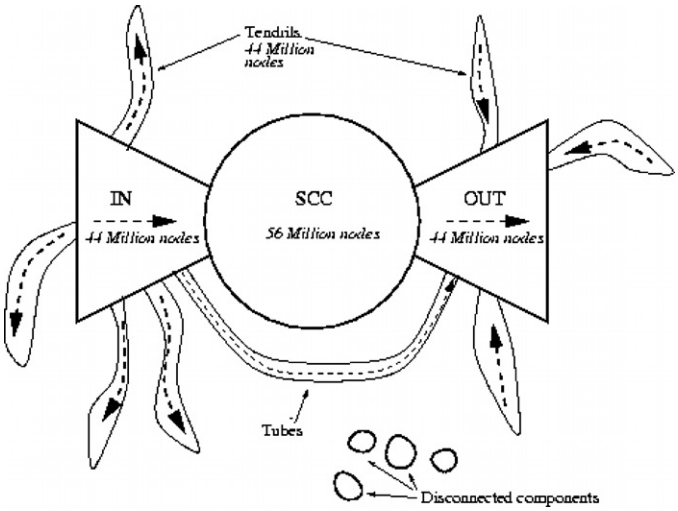


Figure 1. The bow-tie structure of the Web graph. Reprinted from Computer Networks (http://www.sciencedirect.com/science/journal/13891286), volume 33, Andrei Broder *et al*, Graph structure in the Web, pages 309–320, copyright (2000), with permission from Elsevier.

1. Introduction

In the past decade, the world has witnessed the explosion of the World Wide Web from an information repository of a few millions of hyperlinked documents into a massive world-wide 'organism' that serves informational, transactional and communication needs of people all over the globe. Naturally, the Web has attracted the interest of the scientific community, and it has been the subject of intensive research work in various disciplines. One particularly interesting line of research is devoted to analyze the structural properties of the Web, that is, understanding the structure of the *Web graph* [1, 4, 15].

The Web graph is the directed graph induced by the hyperlinks of the Web: the nodes are the (static) HTML pages, and the edges are the hyperlinks between them, directed from the page that contains the link to the target of the link. Understanding the structure and the evolution of the Web graph is a fascinating problem for the community of theoretical computer science. At the same time it has many practical implications. Knowledge of the Web structure can be used to devise better crawling strategies [17], perform clustering and classification [15], improve browsing [5]. Furthermore, it can help in improving the performance of search engines, one of the major driving forces in the development of the Web. The celebrated HITS [13] and PageRank [3] algorithms rely on the link structure of the Web to produce improved rankings of the search results. The knowledge of the macroscopic structure of the Web has been used in devising efficient algorithms for the computation of PageRank [10, 12].

The first large-scale study of the Web graph was performed by Broder *et al* [4] and it revealed that the Web graph contains a giant component that consists of three distinct components of almost equal size: the CORE, made up of a single strongly connected component; the IN set, comprised by nodes that can reach the CORE but cannot be reached by it; the OUT set, consisting of nodes that can be reached by the CORE but cannot reach it. These three components form the well-known *bow-tie* structure of the Web graph, shown in figure 1.[4]

[4] The figure is reproduced from [4].

The bow-tie picture describes the macroscopic structure of the Web. However, very little is known about the inner structure of the components that comprise it. Broder *et al* [4] pose it as an open problem to study further the structure of those components. Understanding the finer details of the Web graph is an interesting problem on its own, but it is also important in practice for improving the performance of algorithms that rely on the link structure of the Web. Furthermore, it could be useful for refining the existing stochastic models for the Web [1, 14, 18].

The study of the Web graph poses additional challenges. Typically, the Web graph consists of millions of nodes and billions of edges. Performing standard graph algorithms (such as BFS and DFS) on a graph of this size is a non-trivial task since data cannot be stored in main memory. It is therefore necessary to devise external-memory algorithms [6] that can work on massive graphs. The challenge is to customize the algorithms to the Web graph, taking advantage of the specific structure of the Web.

In this paper, we study the finer structure of the Web graph, addressing the open question raised by Broder *et al* [4]. We refine the bow-tie picture by providing details for its individual components. In the process, we develop a suite of algorithms for handling massive graphs. Our contributions can be summarized as follows:

- We implement a number of external and semi-external memory graph theoretic algorithms for handling massive graphs, which can run on computers with limited resources. Our algorithms have the distinct feature that they exploit the structure of the Web in order to improve their performance.
- We experiment with four different crawls and we observe the same macroscopic properties previously reported in the literature: the degree distributions follow a power law, and the graph has a bow-tie structure, although (depending on the crawler) a little different in shape.
- We study in detail the inner structure of the bow-tie graph. We perform a series of measurements on the CORE, IN and OUT components. Our measurements reveal the following surprising fact: although the individual components share the same macroscopic statistics with the whole Web graph, they have substantially different structure. We suggest a refinement of the bow-tie picture, the *daisy structure* of the Web graph, that takes our findings into account.

The rest of the paper is structured as follows. In section 2, we review some of the basic graph theoretic definitions, and some of the previous work. In section 3, we outline the algorithms for handling the Web graph. In section 4, we present our experimental findings. We conclude in section 5 with a discussion on the implications of our findings, and possible future experiments.

2. Background

2.1. Graphs and power laws

We will be using various basic graph theoretic definitions and algorithms that can be found in any graph theory textbook (e.g., [7]). Here, we only remind the reader of the definitions of strongly and weakly connected components.

A set of nodes S forms a *strongly connected component (SCC)* in a directed graph, if and only if for every pair of vertices $u, v \in S$, there exists a path from u to v, *and* from v to u. A set of nodes S forms a *weakly connected component (WCC)* in a directed graph G, if and only if the set S is a connected component of the undirected graph G_u that is obtained by removing the directionality of the edges in G.

We will often talk about power-law distributions which are characteristic of the Web. A discrete random variable X follows a power-law distribution if the probability of taking value i is $P[X = i] \propto 1/i^{\gamma}$, for a constant $\gamma \geqslant 0$. The value γ is the exponent of the power law.

2.2. Related work

The study of the structure of the Web graph has recently been the subject of a large body of literature. A well-documented characteristic of the Web graph is the ubiquitous presence of power-law distributions. Kleinberg *et al* [14] and Barabasi and Albert [1] demonstrated that the in-degree of the Web graph follows a *power-law* distribution. Later experiments by Broder *et al* [4] on a crawl of 200M pages from 1999 by AltaVista confirmed it as a basic property: the in-degree of a vertex is distributed according to a power law with exponent $\gamma \approx 2.1$. The sizes of the SCC components also follow a power law. The out-degree distribution follows an imperfect power-law distribution.

Broder *et al* [4] also studied the structure of the Web graph, and presented the bow-tie picture. They decomposed the Web graph into the following components (figure 1): the CORE, consisting of the largest SCC in the graph; the IN, consisting of nodes that can reach the CORE; the OUT, consisting of nodes that are reachable from the CORE; the TENDRILS, consisting of nodes not in the CORE that are reachable from the nodes in IN, or can reach the nodes in OUT; the DISC, consisting of the remaining nodes.

Dill *et al* [9] demonstrated that the Web exhibits self-similarity when considering 'thematically unified clusters' (TUCs), that is, sets of pages that are brought together due to some common trait. Thus the Web graph can be viewed as the outcome of a number of similar and independent stochastic processes. Pennock *et al* [18] also argue that the Web is the sum of stochastic independent processes that share a common (fractal) structure.

The findings about the structure of the Web generated a flurry of research in the field of random graphs. Given that the standard graph theoretic model of Erdös and Rèny [11] is not sufficient to capture the generation of the Web graph, various stochastic models were proposed [1, 14, 18]. Most of them address the fact that the in-degrees must follow a power-law distribution [1]. The *copying model* [14] generates graphs with multiple bipartite cliques [15].

3. Algorithmic techniques for handling the Web graph

This study has required the development of a complete algorithmic methodology for handling very large Web graphs. As a first step we need to identify the individual components of the Web graph. For this we need to be able to perform graph traversals. The link structure of the Web graph takes several gigabytes of disk space, making it prohibitive to use traditional graph algorithms designed to work in main memory. Therefore, we implemented algorithms that achieve remarkable performance improvements when processing data that are stored on external memory. We implemented *semi-external* algorithms, that use only a small constant amount of memory for each node of the graph, as well as *fully-external* algorithms that use an amount of main memory that is independent of the graph size.

We implemented the following algorithms:

- A semi-external graph traversal for determining vertex reachability using only 2 bits per node. The one bit is set when the node is first visited, and the other when all its neighbors have been visited (we say that the node is 'completed'). The algorithm operates on the principle that the order in which the vertices are visited is not important. Starting from

Table 1. Sizes and bow-tie components for the different crawls and the AltaVista graph.

	Italy	Indochina	UK	WebBase	AltaVista
Nodes	41.3M	7.4M	18.5M	135.7M	203.5M
Edges	1.15G	194.1M	298.1M	1.18G	1.46G
CORE	29.8M (72.3%)	3.8M (51.4%)	1.2M (65.3%)	44.7M (32.9%)	56.4 (27.7%)
IN	13.8K (0.03%)	48.5K (0.66%)	312.6K (1.7%)	14.4M (10.6%)	43.3 (21.3%)
OUT	11.4M (27.6%)	3.4M (45.9%)	5.9M (31.8%)	53.3M (39.3%)	43.1 (21.2%)
TENDRILS	6.4K (0.01%)	50.4K (0.66%)	139.4K (0.8%)	17.1M (12.6%)	43.8 (21.5%)
DISC	1.25K (0%)	101.1K (1.4%)	80.2K(0.4%)	6.2M (4.6%)	16.7 (8.2%)

an initial set of nodes, it performs multiple passes over the data, each time visiting the neighbors of the non-completed nodes.

- A semi-external breadth first search that computes blocks of reachable nodes and splits them up in layers according to their distance from the root. In a second step, these layers are sorted to produce the standard BFS traversal of the graph.
- A semi-external depth first search (DFS) that needs 12 bytes plus one bit for each node in the graph. This traversal has been developed following the approach suggested by Sibeyn *et al* [19].
- An algorithm for computing the largest SCC of the Web graph. The algorithm exploits the fact that the largest SCC is a sizable fraction of the Web graph. Thus, by sampling a few nodes of the graph, we can obtain a node of the largest SCC with high probability. We can then identify the nodes of the SCC using the reachability algorithm. As an end product we obtain the bow-tie regions of the Web graph, and we are able to compute all the remaining SCCs of the graph efficiently using the semi-external DFS algorithm.

A software library containing a suite of algorithms for generating and processing massive Web graphs is available online[5]. A detailed presentation of some of these algorithms and a study of their efficiency has been presented in [16]. A complete description of these algorithms is available in the extended version of this work [8].

4. Experiments and results

We experiment with four different crawls. The first three crawls are samples from the Italian Web (the `.it` domain), the Indochina Web (the `.vn`, `.kh`, `.la`, `.mm` and `.th` domains) and the UK Web (the `.uk` domain) collected by the 'Language Observatory Project'[6] and the 'Istituto di Informatica e Telematica'[7] using UbiCrawler [2]. The fourth crawl is a sample of the whole Web, collected by the WebBase project at Stanford[8] in 2001. This sample contains 360 millions of nodes and 1.5 billion of edges. In order to eliminate non-significant data, we pruned the frontier nodes (i.e. the nodes with in-degree 1 and out-degree 0, on which the crawler has been arrested). The sizes of the crawls are shown in table 1.

4.1. Macroscopic measurements

As a first step in our analysis of the Web graph, we repeat the experiments of Broder *et al* [4] on the macroscopic analysis of the graph. We computed the in-degree, out-degree and

[5] http://www.dis.uniroma1.it/~cosin/.
[6] www.language-observatory.org.
[7] www.itt.cnr.it.
[8] http://www-diglib.stanford.edu/testbed/doc2/WebBase/.

SCC size distributions. As expected, the in-degrees, and the sizes of SCCs follow a power-law distribution, while the out-degree distribution follows an imperfect power law. All our measurements are in agreement with the respective measurements of Broder *et al* [4] for the AltaVista crawl. More detailed results on the various distributions for the WebBase crawl are reported in [16].

We also computed the macroscopic structure of the Web graph. We observe a bow-tie structure. The relative sizes of the components of the bow-tie are shown in table 1, where we also present the numbers for the AltaVista crawl [4], for the purpose of comparison. The first observation is that for the Italian, Indochina and UK crawls, the IN and TENDRILS components are almost non-existent. As a result either the CORE is overgrown (for the Italian and UK crawls), or the nodes are equally distributed between the CORE and the OUT component. For the WebBase crawl we observe that the relative size of IN (11%) is significantly smaller than that observed in the AltaVista crawl, while the OUT component (39%) is now the largest component of the bow-tie. These discrepancies with the AltaVista crawl can most likely be attributed to different crawling strategies and capabilities, rather than to the evolution of the Web. The first three crawls are relatively recent, and all crawls are generated using a small number of starting points. Unfortunately, large-scale crawls are not publicly available.

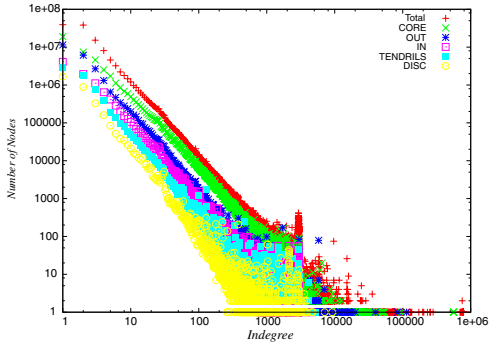
4.2. The inner structure of the bow-tie graph

We now study the fine-grained structure of the Web graph. We are interested in understanding not only the characteristics of each component individually, but also how the components relate to each other. For this purpose we label each node with the name of the component to which it belongs. This gives us five sets of nodes (CORE, IN, OUT, TENDRILS, DISC). For each such subset we obtain the induced subgraph, resulting in five different subgraphs. For example, when referring to the IN graph, we mean the graph that consists of the nodes in IN and all the edges between these nodes.

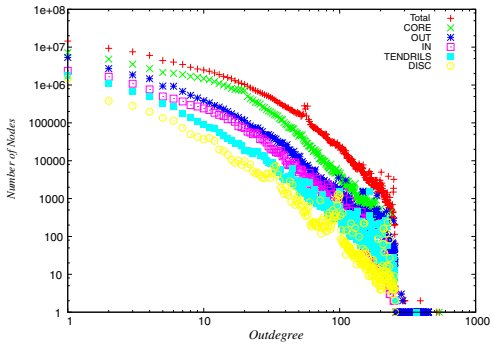
As a first step in the understanding of the individual components we compute the same macroscopic measures as for the whole Web graph. We compute the in-degree, out-degree and SCC size distributions for each of the IN, OUT, TENDRILS and DISC graphs. Figure 2 shows the plots of the distributions for each component and for the whole graph, for the case of the WebBase crawl. It is obvious that the same macroscopic laws that are observed on the whole graph are also present in the individual components.

4.2.1. The structure of the IN and OUT components. Given the fact that the in-degree, out-degree and SCC size distributions in the IN and OUT components are the same as for the whole Web graph, it is tempting to conjecture that the Web has a *self-similar* structure. That is, the bow-tie structure repeats itself inside the IN and OUT components. Dill *et al* [9] demonstrated that the web exhibits self-similarity when considering 'thematically unified' sets of web pages. These subsets are structurally similar to the whole Web. Similar observations are made by Pennock *et al* [18]. However, the subsets considered by these previous works are composed of nodes that may belong to any of the components of the bow-tie graph. The question we are interested in is, whether such self-similarity appears when considering the individual components of the bow-tie graph.

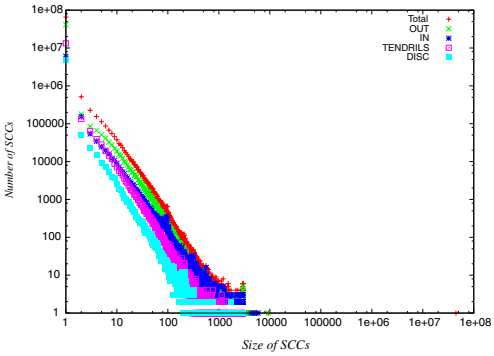
The first indication that the self-similarity conjecture is not true comes from the fact that there is no large SCC in the IN and OUT components. For the OUT component, in all crawls, the largest SCC is only a few thousands of nodes. Given that the size of the OUT component is in the order of millions, the largest SCC is staggeringly small. Furthermore, this is also the



(*a*) In-degree distributions



(*b*) Out-degree distributions



(*c*) SCC size distributions

Figure 2. Macroscopic measures for all components. (*a*) In-degree distributions, (*b*) Out-degree distributions and (*c*) SCC size distributions.

second largest SCC in the graph, which, compared to the largest one (the CORE), is minuscule. We observe a similar phenomenon for the IN component. For the WebBase graph (which is the most interesting case, since the IN component is a non-trivial fraction of the graph) the largest SCC in the IN component is less than 6000 nodes. Detailed numbers about the size of the largest SCC in the IN and OUT components are given in table 2.

Therefore, it appears that there exists no sizable SCC in the IN and OUT components that could play the role of the CORE in a potential bow-tie. However, it is still possible that there exists a giant weakly connected component (WCC) in each component. We therefore computed the WCCs of the two sets. Surprisingly, we discovered that there is no giant WCC in either of the two components. In fact, there is a large number of WCCs per component and their sizes follow a power-law distribution. Figure 3(*a*) shows the WCC size distribution for the WebBase graph. Statistics for all graphs are reported in table 2. Most of the WCCs are of size one. The singleton WCCs comprise 10–22% of the IN component (with the exception of Indochina), and 20–45% of the OUT component. On the other hand, the largest WCC is never more than 30% of the component it belongs to, which is small compared to the giant WCC in the Web graph, which contains more than 90% of the nodes. For the WebBase graph, the largest WCC in the IN component consists of just 1% of the nodes, while the largest WCC in the OUT component consists of 28% of the nodes.

We also investigate how the nodes in the largest WCCs in the IN and OUT components are connected to see if they organized in a bow-tie shape. Our investigation revealed that

Table 2. Statistics for the IN, OUT and CORE components for each crawl.

	Italy	Indochina	UK	WebBase
The IN component				
Nodes in IN	13.8K (0.03%)	48.5K (0.66%)	312.6K (1.69%)	14.4M (11%)
Max SCC	1590	7867	4171	5876
Number of WCCs	1633	117	62K	3.68M
Max WCC	4085 (29.5%)	13.2K (27.2%)	8246 (2.7%)	197.5K (1.3%)
Singleton WCCs	1543 (11.15%)	63 (0.13%)	56K (17.89%)	3.2M (22.46 %)
The OUT component				
Nodes in OUT	11.4M (27.6%)	3.4M (45.9%)	5.9M (31.8%)	53.3M (39%)
Max SCC	19,170	39 283	26 525	9349
Number of WCCs	3.73M	7296K	1.97M	25.4M
Max WCC	1.43M (12.52%)	335.9K (9.85%)	457.4K (7.75%)	14.94M (28.01%)
Singleton WCCs	3.49M (30.6%)	672K (19.71%)	1.84M (31.11%)	24.48M (45.91%)
The CORE component				
Nodes in CORE	29.8M (72.3%)	3.8M (51.4%)	1.2M (65.28%)	44.7M (33%)
Entry points	10.2K (0.03%)	2.3K (0.06%)	106.3K (0.88%)	2.6M (5,87%)
Exit points	15.6M (52.2%)	2.3M (59.6%)	4.8M (39.8%)	29.6M (72.03%)
Bridges	6.25K(0.02%)	1.5K (0.04%)	61.8K (0.51%)	2M (4.58%)
Connectors	1.7M (5.71%)	164.2K (4.32%)	537.9K (4.45%)	2.96M (6.63%)
Petals	325.3K (1.09%)	52.5K (1.38%)	138K (1.14%)	1.4M (3.14%)

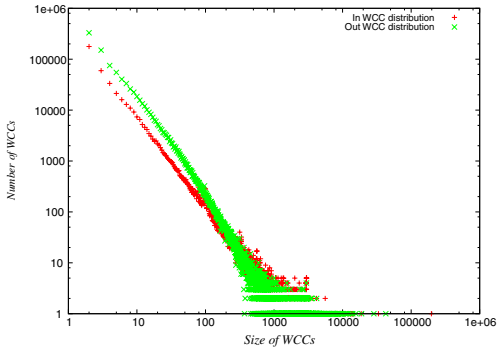
Table 3. IN and OUT depth.

	Italy	Indochina	UK	WebBase
Depth IN	2	11	15	8
Depth OUT	26	21	25	580

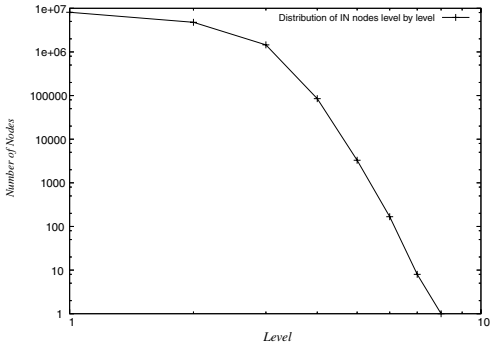
starting from the largest SCC in the WCC, we can create a bow-tie that is no more than 15% of the WCC (for the Italian Web), and usually less than 5%. The rest belongs to the DISC component. (Note that a node that points to the tendrils coming out of IN, or is pointed to by those going into OUT, belongs to DISC, although it is still weakly connected to the graph). This suggests that the WCC consists of multiple small atrophic bow-ties that are sparsely interconnected with each other.

In order to better understand how the nodes in IN and OUT are arranged with respect to the CORE, we performed the following experiment. We condensed the CORE in a single node and we performed a forward and a backward BFS. This allows us to split the nodes in the IN and OUT components in *levels* depending on their distance from the CORE. The depths of the components are shown in table 3. In all graphs, the depths of the components are relatively small. Furthermore, most nodes are concentrated close to the CORE. Typically, about 80–90% of the nodes in the OUT component are found within the first five layers. For the WebBase graph, although the OUT is much deeper, with 580 levels, more than 58% of its nodes are at distance 1 from the CORE, and 93% are within distance 5. Furthermore, after level 305 there exists only a single chain of nodes that extends until level 580, making the effective depth of the OUT 305. The node distributions, level by level, for the WebBase graph are shown in figures 3(*b*) and (*c*), for the IN and OUT sets respectively. The plots are in logarithmic scale.

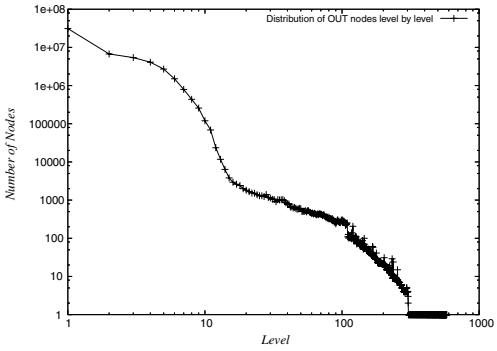
Therefore, we conclude that the IN and OUT components are shallow and highly fragmented. They are comprised of several sparse weakly connected components of low depth. Most of their volume consists of nodes that are directly linked to the CORE.



(*a*) Distribution of WCC sizes per component



(*b*) Distribution of IN nodes level by level



(*c*) Distribution of OUT nodes level by level

Figure 3. Characteristics of the IN and OUT components. (*a*) Distribution of WCC sizes per component, (*b*) Distribution of IN nodes level by level and (*c*) Distribution of OUT nodes level by level.

4.2.2. The structure of the CORE. As a first step in the study of the CORE graph, we examine its relation with the IN and OUT components. We define an *entry point* to the CORE to be a node that is pointed to by at least one node in the IN component, and an *exit point* to be a node that points to at least one node in the OUT component. A *bridge* is a node that is both an entry and an exit point. The number of entry and exit points is shown in table 2. It is interesting to observe that a large fraction of the entry points act like bridges. Furthermore, with the exception of the UK crawl, the majority of the nodes in the CORE is connected to the 'outside' world. In the WebBase crawl, this number is around 80% of the whole CORE, while the 'deep CORE' consists of a little more than 20%.

We also compute the in-degree distribution of the entry points when we restrict the source of the links to be in the IN component, and, as expected, we observe a power law. This implies that most nodes 'serve' as entry points to just a few nodes in the IN component, while there exist a few nodes that serve as entry points to a large number of IN nodes. Similar distributions are obtained when we consider the out-degree distribution of the exit points, restricted to the OUT component.

We then study the connectivity of the CORE. We first look for nodes that are loosely connected to the CORE. We define a *connector* to be a node of the CORE that has a single in-coming and out-going link. A connector forms a *petal* if the source of the incoming link, and the target of the out-going link are the same node. Large number of connectors would

Table 4. Sensitivity of the CORE under targeted attacks.

(a) Deleting nodes with high total degree					(b) Deleting nodes with high in-degree and out-degree							
Deg	Del	Max SCC	Max SCC (%)	SCC num	In-deg	Del	Out-deg	Del	Total del	Max SCC	Max SCC (%)	SCC num
50 000	9	44.2M	98.9	81K	4000	1.1K	233	1,154	2,263	42.2M	94.4	595K
21 500	39	43.7M	97.9	175K	2600	9.9K	185	10K	20.6K	39.8M	89.0	1.75M
10 000	199	43.2M	96.6	285K	1750	26K	158	25K	51K	37M	82.9	3M
4000	1.1K	42.3M	94.7	505K	1000	52K	130	54K	105K	33.7M	75.5	4.75M
1000	55K	35.1M	78.6	3.7M	500	112K	105	108K	219K	29.4M	66.1	7M
500	120K	31M	69.6	5.7M	225	259K	82	227K	487K	23.5M	53.3	10M
100	1.03M	14.8M	34.6	14.7M	120	518K	62	499K	949K	17.8M	40.8	13M

imply weak connectivity of the CORE. The number of connectors is shown in table 2, and it is on average around 5%. Of these 20–45% are petals. Therefore, connectors are only a small part of the CORE.

In order to further understand the connectivity of the CORE, we test the resilience of the CORE to targeted attacks by performing the following experiment. For some k we delete all nodes from the CORE that have total degree (in-degree plus out-degree) at least k. We then compute the size of the largest SCC in the resulting graph. Table 4(a) shows how the size of the largest SCC changes as we decrease k, and we increase the number of deleted nodes for the case of the WebBase graph. Similar trends are observed in the other crawls. We observe that the threshold on the total degree must become as low as 100 in order to obtain an SCC of size less than 50% of the CORE.

We note that there is a large discrepancy between the values of the in-degrees and out-degrees in the Web graph. The highest in-degree is close to 566K, while the highest out-degree is just 536. Note that an upper-bound on the out-degree may be imposed by the crawler, if it limits the number of outgoing links of a page that it explores. Therefore, it may be the case that when deleting the nodes with high total degree, we only delete nodes with high in-degree. We experiment with a different kind of attack that removes (approximately) k nodes with the highest in-degree and k nodes with the highest out-degree. The results are shown in table 4(*b*). The CORE remains resilient even against this combined attack. An interesting observation while performing this experiment was that the nodes with the highest in-degree and the nodes with the highest out-degree are quite distinct. Actually, the correlation between the in-degree and out-degree is close to zero. It appears that nodes that are strong hubs in the CORE are not also strong authorities.

There are two ways to interpret these results. The first is that there are no obvious *failure points* in the CORE, that is, strong hubs or authorities that pull the rest of the nodes together, and whose removal from the graph causes the immediate collapse of the network. In order to disconnect the CORE you need to remove nodes with sufficiently low degree. On the other hand, note that we managed to reduce the largest SCC to 35–40% of the original by removing about 1M nodes. However this is less than 1% of the total nodes. In that sense the CORE is vulnerable to targeted attacks.

5. Discussion and future work

In this paper, we undertook a study of the Web graph at a finer level. We observed that the ubiquitous presence of power laws describing several properties at a macroscopic level does not necessarily imply self-similarity in the individual components of the Web graph. Indeed, the different components have quite distinct structure, with the IN and OUT being highly fragmented, while the CORE being well interconnected.

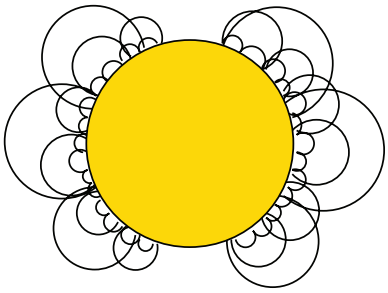


Figure 4. The daisy structure of the Web.

Our work suggests a refinement of the bow-tie pictorial view of the Web graph [4]. The bow-tie picture seems too coarse to describe the details of the Web. The picture that emerges from our work can better be described by the shape of a *daisy* (figure 4): the IN and OUT regions are fragmented into large number of small and shallow *petals* (the WCCs) hanging from the central dense CORE.

It would be interesting to obtain larger, and more 'realistic' crawls, and perform the same measurements to verify our hypothesis. Our current results are sensitive to the choices and limitations of the crawlers, and it is not clear if the available crawls are representative of the actual Web graph. Unfortunately, there are no publicly available crawls that have been collected with the aim of validating our hypothesis on the structure of the Web graph. We plan in the future to collect crawls with this goal in mind.

A deeper understanding of the structure of the Web graph may also have several consequences on designing more efficient crawling strategies. The fact that IN and OUT are highly fragmented may help in splitting the load between different robots without much overlapping. Moreover, the fact that most of the vertices are at few hops from the CORE may explain why breadth first search crawling is more effective than other crawling strategies [17].

Our work motivates further experiments on the Web graph. It would be interesting to devise efficient algorithms for estimating the clustering coefficient, a commonly used measure for connectivity. Furthermore, further exploration of the structure of the CORE is necessary to gain a deeper understanding. Possible measurements could include spectral properties, or clustering and community discovery. As a concluding remark, we observe that we are still very far from devising a theoretical model that is able to capture the finer connectivity properties of the Web graph.

References

[1] Barabasi A L and Albert A 1999 Emergence of scaling in random networks *Science* **286** 509–12
[2] Boldi P, Codenotti B, Santini M and Vigna S 2004 Ubicrawler: a scalable fully distributed web crawler *Softw. Pract. Exp.* **34** 711–26
[3] Brin S and Page L 1998 The anatomy of a large-scale hypertextual Web search engine *WWW*
[4] Broder A, Kumar R, Maghoul F, Raghavan P, Rajagopalan S, Stata S, Tomkins A and Wiener J 2000 Graph structure in the web *Comput. Netw.* **33** 309–20
[5] J Carrìere and R Kazman Webquery: searching and visualizing the web through connectivity *6th WWW Conf.*
[6] Chiang Y, Goodrich M T, Grove E F, Tamassia R, Vengroff D E and Vitter J S 1995 External-memory graph algorithms *SODA*
[7] Cormen T H, Leiserson C E and Rivest R L 1992 *Introduction to Algorithms* (Cambridge, MA: MIT Press)
[8] Millozzi S, Donato D, Leonardi S and Tsaparas P 2005 Mining the inner structure of the web graph *Technical report* DELIS-TR-157 http://delis.upb.de/docs/

[9] Dill S, Kumar R, McCurley K, Rajagopalan S, Sivakumar D and Tomkins A 2001 Self-similarity in the web *Proc. 27th VLDB Conf.*
[10] Eiron N, McCurley K S and Tomlin J A 2004 Ranking the web frontier *WWW*
[11] Erdös P and Rèny A 1960 On the evolution of random graphs *Publ. Math. Inst. Hung. Acad. Sci.* **5** 17–61
[12] Kamvar S, Haveliwala T, Manning C and Golub G 2003 Exploiting the block structure of the web for computing pagerank *Technical report* Stanford University
[13] Kleinberg J 1997 Authoritative sources in a hyperlinked environment *J. ACM* **46** 604–32
[14] Kleinberg J, Kumar R, Raghavan P, Rajagopalan S and Tomkins A 1999 The web as a graph: measurements, models and methods *Proc. Intl. Conf. Combinatorics and Computing*
[15] Kumar R, Raghavan P, Rajagopalan S and Tomkins A 1999 Trawling the web for emerging cyber communities *WWW*
[16] Laura L, Leonardi S, Millozzi S, Meyer U and Sibeyn J F 2002 Algorithms and experiments for the webgraph *European Symposium on Algorithms (ESA)*
[17] Najork M and Wiener J L 2001 Breadth-first crawling yields high-quality pages *WWW Conf.*
[18] Pennock D M, Flake G W, Lawrence S, Glover E J and Giles C L 2002 Winners don't take all: characterizing the competition for links on the web *Proc. Natl Acad. Sci.* **99** 5207–11
[19] Sibeyn J F, Abello J and Meyer U 2002 Heuristics for semi-external depth first search on directed graphs *SPAA*