Analyzing the structure of large graphs

Ravi Kannan
Computer Science Department,
Yale University,
New Haven, CT 06511.
Email: kannan@cs.yale.edu.

V. Vinay
Computer Science and Automation
Indian Institute of Science,
Bangalore, India.
Email : vinay@csa.iisc.ernet.in

July 13, 1999

Abstract

The purpose of this paper is to study the structure of large sparse graphs such as the graph of the web - the graph consisting of one node per document in the web and a directed edge from i to j whenever document i has a hypertext link to document j. This particular graph is of course of interest as being by far the largest "human-made" graph (with millions of vertices) arising from a "natural setting". We develop theoretical tools and algorithms to analyze the density structure of such graphs by relating density to easy to find Linear Algebra quantities.

1 Introduction

In general, for such large graphs, there are a number of simple questions which one may easily answer by making "one-pass" through the graph (visiting say each edge once and doing some minor bookkeeping). Among these are questions like the average in-degree, the distribution of degrees etc. But looking for sub-structures consisting of even a few nodes like $K_{3,3}, K_4$ etc. and estimating their number is very difficult. However, they are of interest because they reveal small "highly connected" sub-structures. More generally, one would like to study which parts of the graph are relatively dense (have a lot of edges) and which are sparse. Such questions have received some attention lately [8],[7] [11], [9]. The motivation for the web-graph is to identify so-called "web communities" - collections of related documents (which form the dense parts of the graph.) Several combinatorial heuristics have been proposed to tackle these problems. But the prohibitive size of the web has meant that while these algorithms do identify some of the communities, they are not able to exhaustively enumerate them; also while enumerative procedures may tackle small sub-structures (with less than, say, 10 nodes), they cannot identify larger dense sub-structures.

The idea of isolating the dense and sparse parts of a graph has also been carried out in Graph Theory beginning with the fundamental Regularity Lemma of Szemerédi [12], [10]. This lemma roughly states that the vertex set of any graph may be partitioned into a small number of pieces so that for most pairs of pieces, the density of subparts of the pieces is close to the density of the whole pieces. This has many applications in Graph Theory and Number Theory. Constructive versions of the lemma (which find the partition) are now available [1],[6].But this lemma applies only to graphs whose number of edges is $\Omega(|V|^2)$, i.e., to dense graphs. Graphs like the web graph are very sparse (in the case of the web graph, the average degree is in the single digits, while the number of vertices is in the tens of millions). Thus this theory is not directly applicable to such graphs. However, we will draw some help from this area.

Here, we will first introduce a different definition of density (quantifying "relatively highly connected") suitable for sparse directed graphs like the web graph.

We develop an algorithm which successively finds and removes the (approximately) densest subgraph until there is no more dense subgraph left. [Thus this is also in a sense a decomposition, like in the setting of the Regularity Lemma, but here, we deal with sparse graphs.]

We will prove a Theorem relating the density of the densest subgraph directly to the linear algebraic quantity - the singular value. Then we may use recently developed Monte Carlo algorithm for the Singular Value Decomposition of a matrix[4],[3]; it is fast enough to be feasible on graphs of the size of the web graph.

The question these algorithms are capable of addressing is the following : The hypertext links in the web are of two types; the first is a link caused by what we may call a "global factor" - i.e., the URL's interest in a popular topic (examples are - programming, surfing, baseball, air travel, Theoretical Computer Science ...). One would expect that there are relatively few such broad topics of interest (perhaps in the several hundreds), but they cause a large number of links in the web (perhaps in the tens of percentage). One would expect these links to be highly correlated. The second type of link is caused by what we may call "local factors" (for example to a cousin's or friend's homepage). One expects such links to be independent and scattered. We can use the methods of Statistics (in particular Principal Component Analysis) to related these considerations to Linear Algebraic quantities like the singular values. We elaborate on this in the final paper; the focus of the current write-up is to formulate the theoretical definitions and results to carry out such an analysis. We plan to run experiments to identify the role of global and local factors based on the algorithms and theorems here.

We also show in this paper that Linear Algebra can be used to solve the following clustering problem: Suppose there are m data objects, each with n attributes presented by an $m \times n$ matrix A (where m, n are large), where A_{ij} is the value of the j th attribute of object i. Define the correlation of objects i and j as $\sum_{k=1}^{n} A_{ik}A_{jk}$. (This is sometimes called the similarity. If it each feature is is either "on" or "off", then this is the number of common features turned on.) The average correlation of a subset of objects is the sum of the pairwise correlations divided by the number of objects. (It may appear at first sight that the more natural thing is to divide by the number of pairs of objects in the subset, but we will later argue that the division by the number of objects is more appropriate.) An important problem is to find the subset of objects with the maximum average correlation. We will show that our Linear Algebra based method solves this problem too approximately and that the densest subgraph problem above and this problem are related in a natural way.

[This problem of finding the maximum average correlation can in fact be solved exactly by flow techniques in polynomial time ; we outline this well- known algorithm in Remark (2) below. But of course a polynomial time flow algorithm is far too slow for our purpose here. So, we will develop a simple Linear Algebra based approximation algorithm. But, we suspect that the problem of finding $d(A)$ exactly is NP-hard.]

Finally, we also show a relation between these quantities and higher order correlations among objects where we take the correlations of triples, quadruples and so on of objects.

Notation We will prove our results for a general matrices with nonnegative entries. After scaling, we will assume that $0 \leq A_{ij} \leq 1$. In the case of graphs, A can be taken to be the adjacency matrix. R will denote the set of rows of the matrix A and C the set of columns; we call A an $R \times C$ matrix. For any $S \subseteq R$ and $T \subseteq C$, we let $A(S,T) = \sum_{i \in S, j \in T} A_{ij}$. For each $i \in R$, we denote by $\deg(i)$ the sum $\sum_{j \in C} A_{ij}$; if A is the adjacency matrix of a directed graph, then $\deg(i)$ is the out-degree of the node i. [We may define in-degrees as column sums; but we do not use this.] We also define for $i, j \in R$ (not necessarily distinct) co-deg(i,j) to be the sum $\sum_{k \in C} A_{ik}A_{jk}$; in the case of a graph, this is the "co-degree" of the nodes i, j, i.e., the number of nodes to which both i and j have edges. In the case that the matrix A represents the object-attribute matrix (as described above), the co-deg(i,j) is the correlation between the objects i and j.

We also need to recapitulate some Linear Algebra. Every real matrix A can be expressed as

$$A = \sum_{t=1}^{r} \sigma_t(A) u^{(t)} {v^{(t)}}^T$$

where $\sigma_1(A) \geq \sigma_2(A) \geq \ldots \geq \sigma_r(A) \geq 0$ are called the singular values of A and the $u^{(t)}$ form an orthonormal set of column vectors and so do the $v^{(t)}$. (For any matrix P, we denote by P^T the transpose of P.) Also ${u^{(t)}}^T A = \sigma_t {v^{(t)}}^T$ and $Av^{(t)} = \sigma_t u^{(t)}$ for $1 \leq t \leq r$. This is the *singular value decomposition* of A.

We denote by $||A||$ the square root of the sum of squares of the entries of A. It is also known from standard Linear Algebra that

$$||A||^2 = \sum_t \sigma_t(A)^2.$$

2 Main Definitions, Theorems

Definition 1 Let $S \subseteq R$ and $T \subseteq C$. We define the density of the pair of sets S, T by

$$d(S,T) = \frac{A(S,T)}{\sqrt{|S||T|}}. \tag{1}$$

We define the density $d(A)$ of the whole matrix A by

$$d(A) = \max_{S \subseteq R, T \subseteq C} d(S,T).$$

Remark 1. It may seem more natural at first sight to have $|S||T|$ rather than the square root in the denominator. But if we defined it this way, a pair of vertices with an edge between them would have the highest density, namely 1. This trivial problem may be avoided by taking the maximum of $\frac{|\{(u,v):u\in S, v\in T\}|}{|S||T|}$ over all sufficiently large subsets S, T. Besides being inelegant, such restrictions on sizes of sets make the computational problem of finding the approximately densest subsets NP-hard, even if we allow for very large approximation factors. [2], [5].

Remark 2. Suppose Δ is the maximum row or column sum of the matrix A. We have trivially :

$$d(S,T) \leq \Delta \min(|S|, |T|)/(\sqrt{|S||T|}) \leq \Delta.$$

Also, if A is a 0-1 matrix and row i has row sum equal to Δ, then we may take $S = \{i\}$ and $T = \{j : A_{ij} = 1\}$ and we therefore get $d(S,T) \geq \sqrt{\Delta}$.

Remark 3. If $S = T$, and A is the adjacency matrix of an undirected graph, then note that $d(S,T)$ is the average degree of the induced subgraph on S. [We note that by using flow techniques, it is possible to find the induced subgraph of an undirected graph of maximum average degree (see Remark (2) below) in polynomial time. However, the running time is not feasible for graph sizes that we are considering.

Remark 4. For a regular undirected graph, it is easy to see that we maximize $d(S,T)$ by putting $S = T = V$. In this case, $d(.,.)$ does not yield much information. This is as it should be since no subpart of the graph is really denser than the whole.

Remark 5. For a sparse directed graph with dn edges, where d is $O(1)$ (as in the case of the web graph) , we have that $d(V,V)$ is $O(1)$. There may be subparts which are denser. Note that if instead of $\sqrt{|S||T|}$ in the denominator, we put $(|S||T|)^{\lambda}$, where $\lambda < 1/2$, then $d(S,T)$ grows with n and so such a criterion is more likely to choose $S = T = V$ than ours.

Remark 6. If we had two disjoint subsets S_1, S_2 of the rows both of size l and also two disjoint subsets T_1, T_2 of columns both of size l', and the number of edges between S_i and T_j are equal for all 4 possibilities - $1 \leq i, j \leq 2$, then our criterion would prefer $S = S_1 \cup S_2$ and $T = T_1 \cup T_2$ over any individual piece. This is desirable since we want to find as large "web communities" as possible.

We now define another quantity which measures the average correlation and later we will show that this is related to the density above.

Definition 2 For any subset S of the rows, define the average correlation of S by :

$$f(S) = \frac{1}{|S|} \sum_{u,v \in S} co - deg(u,v).$$

(where we take the sum over all ordered pairs u, v, so that u, v as well as v, u appear separately in the sum.)

Also define

$$f(A) = \max_S f(S).$$

Remark 7. It is easy to see that

$$f(S) = \frac{1}{|S|}(AA^T)(S,S).$$

Remark 8. It again may appear that the more natural thing would be to divide by $|S|^2$, the number of (ordered) pairs of rows for which we sum the co-degrees. But if we use this definition, it is easy to argue that the singleton set S containing the element i with the maximum $\sum_j A_{ij}^2$ would maximize the function. It may be argued that this pathology arises from our including the correlation of an object with itself in the sum; but even if we exclude such self-correlations, it is easy to see that the set of two (or a small number of) objects with the maximum co degree is likely to be the answer. This is of course not desirable. What our definition does is to allow the inclusion of more and more objects in the set S as long as they are somewhat correlated with the objects already present. Since our aim is to find as large sets as possible which are well-correlated, this is exactly what we would like.

Remark 9. $f(A)$ can be found (exactly) in polynomial time using flow techniques. We thank Uri Feige and Guy Kortsarz for pointing out this algorithm to us. It uses Remark2 and is as follows : We construct a directed network with a source s, sink t. The vertex set of the network is $R \cup (R \times R) \cup \{s, t\}$. There is an edge in the network from s to each vertex in $R \times R$ with capacity equal to the corresponding entry in the matrix AA^T. From each vertex (i, j), there are two edges - one to i and one to j, both of infinite capacity. There is one edge from each vertex in R to t, each such edge of capacity λ. [λ is a variable here; later it will turn out that the maximum value of λ will yield the answer.] Now suppose the minimum $s - t$ cut in the network is $P, \bar{P}$ where $s \in P$ and $t \in \bar{P}$. Then, clearly, $(i, j) \in P \cap (R \times R)$ implies that $i, j \in P \cap R$. Also, the capacity of the cut is

$$\sum_{(i,j)\in(R\times R)\setminus P} (AA^T)_{ij} + \lambda|P \cap R| = \sum_{(i,j)\in(R\times R)} (AA^T)_{ij} - [\sum_{(i,j)\in P} (AA^T)_{ij} - \lambda|P \cap R|].$$

So, the quantity in square brackets is maximized by the cut. Let $h(\lambda) = [\sum_{(i,j)\in P}(AA^T)_{ij} - \lambda|P \cap R)|]$. Clearly, $h(\lambda)$ is a non-increasing function of λ; we find by binary search the maximum value of λ for which $h(\lambda)$ is nonnegative. It is now easy to see that this λ is exactly the answer we seek.

We now prove results connecting the quantities above and the singular value of the matrix A which is easy to find by standard Linear Algebra in polynomial time.

Theorem 1 *Suppose A is an $R \times C$ matrix with $0 \le A_{ij} \le 1$ for all i, j. Let $|C| = n$. We have*

$$\sigma_1(A) \ge d(A) \ge \frac{1}{\frac{5}{2}\log n + \log 10}\sigma_1(A).$$

Further, subsets S, T satisfying $d(S, T) \ge \frac{1}{\frac{5}{2}\log n + \log 10}\sigma_1(A)$ may be easily found from the top singular vectors of A.

Proof The first inequality is easy to see. We prove the second. Suppose

$$\sigma_1(A) = x^T Ay = \sum_{i,j} x_i A_{ij} y_j, \qquad |x| = |y| = 1.$$

Since the entries of A are nonnegative, we know that $x_i \geq 0$ and $y_j \geq 0$ for all i, j. Let $M = \lceil \log_2(10\sqrt{n}) \rceil$. For $t = 0, 1, 2, \dots M$, define

$$S_t = \{i \in R : 2^t \frac{1}{10\sqrt{n}} < x_i \leq 2^{t+1}\frac{1}{10\sqrt{n}}\} \qquad T_t = \{j \in C : 2^t \frac{1}{10\sqrt{n}} < y_j \leq 2^{t+1}\frac{1}{10\sqrt{n}}\}.$$

Then we have (using the Cauchy-Schwartz inequality in the third step and the fact that for any unit length vector x, we have $|Ax| \leq \sigma_1(A)$),

$$\sum_{i,j} x_i A_{ij} y_j \leq \sum_{t,t'=0}^{M} 2^{t+t'+2}\frac{1}{100n} A(S_t, T_{t'}) + \sum_{i,j : x_i \leq \frac{1}{10\sqrt{n}}} x_i y_j A_{ij} + \sum_{i,j : y_j \leq \frac{1}{10\sqrt{n}}} x_i y_j A_{ij}$$

$$\leq \sum_{t,t'} \left(\frac{1}{100n} 2^{t+t'+2} \sqrt{|S_t||T_{t'}|} \frac{A(S_t, T_{t'})}{\sqrt{|S_t||T_{t'}|}} \right) + \frac{1}{10\sqrt{n}} \sum_i \sum_j A_{ij} y_j + \frac{1}{10\sqrt{n}} \sum_j \sum_i A_{ij} x_i$$

$$\leq \left(\sum_{t,t'} \frac{1}{10^4 n^2} 2^{2t+2t'+4} |S_t||T_{t'}| \right)^{1/2} \left(\sum_{t,t'} \frac{A(S_t, T_{t'})^2}{|S_t||T_{t'}|} \right)^{1/2} + \frac{1}{5}\sigma_1(A)$$

$$\text{Hence we have } \frac{4}{5}\sigma_1(A) \leq \left(16 \sum_{i,j} x_i^2 y_j^2 \right)^{1/2} \left(\sum_{t,t'} \frac{A(S_t, T_{t'})^2}{|S_t||T_{t'}|} \right)^{1/2}$$

$$\leq 4 (\sum_i x_i^2)^{1/2} (\sum_j y_j^2)^{1/2} \left(\sum_{t,t'} \frac{A(S_t, T_{t'})^2}{|S_t||T_{t'}|} \right)^{1/2} = 4 \left(\sum_{t,t'} \frac{A(S_t, T_{t'})^2}{|S_t||T_{t'}|} \right)^{1/2}.$$

This clearly proves that there is some pair t, t' for which we have the lower bound of the Theorem.

Also note that if we have the singular vectors x, y a t, t' attaining the lower bound may be easily found.

□

Theorem 2

$$(d(A))^2 \leq f(A) \leq (d(A))^2 \ln n.$$

Proof For any subset S define an $|R|-$ vector $u(S)$ by letting $u(S)_i = 1/\sqrt{|S|}$ for $i \in S$ and 0 otherwise. Then we have

$$d(S,T) = u(S)^T A u(T) \leq |u(S)^T A| \quad \text{since } |u(T)| = 1$$

$$\text{But, } |u(S)^T A|^2 = u(S)^T A A^T u(S) = \sum_{i,j} u(S)_i \text{co-deg}(i,j) u(S)_j = \frac{1}{|S|} \sum_{i,j \in S} \text{co-deg}(i,j),$$

and from this the first inequality in the Theorem follows. For the other inequality, consider any fixed $S \subseteq R$. For convenience, let us number the columns such that $\deg_S(j) = \sum_{i \in S} A_{ij}$ satisfy $\deg_S(1) \geq \deg_S(2) \geq \deg_S(3) \dots$. By the definition of $d(A)$, we see that for every $t, t \in \{1, 2, \dots |C|\}$, we have

$$\sum_{j=1}^{t} \deg_S(j) \leq d(A)\sqrt{|S|t}. \tag{2}$$

By Calculus, it is easy to see that subject to (2), the expression $\sum_{j\in C}\deg_S(j)^2$ is maximized if we had $\deg_S(j) = d(A)\sqrt{|S|}(\sqrt{j}-\sqrt{j-1})$ for $j = 1, 2, \ldots |C|$. Thus we always have

$$\sum_{j\in C}\deg_S(j)^2 \le (d(A))^2|S|\left(\sum_{j=1}^{|C|}(\sqrt{j}-\sqrt{j-1})^2\right)$$
$$\le (d(A))^2|S|\sum_{j=2}^{|C|}(j-1)(1/4(j-1)^2) \quad \text{using } \sqrt{1+x} \le 1+\frac{x}{2}$$
$$\le (d(A))^2|S|\ln|C|.$$

But, we note that

$$\sum_{i,i'\in S}\text{co-deg}(i,i') = \sum_{j\in C}\deg_S(j)^2,$$

each $j \in C$ appears precisely $\deg_S(j)^2$ times when we add up the co-degrees of all pairs of nodes in S. Hence the Theorem follows.

□

By similar arguments, we can also prove the following Theorem. [We defer the proof to the final paper.]

Theorem 3

$$\sigma_1(A)^2 \ge f(A) \ge \frac{\sigma_1(A)^2}{\frac{5}{2}\log_2 n + 5\log_2 10}.$$

Higher order correlations For rows $u_1, u_2, \ldots u_k$, define

$$\text{co-deg}(u_1, u_2, \ldots u_k) = \sum_{j\in C} A_{u_1,j}A_{u_2,j}\ldots A_{u_k,j}.$$

Then for any sunset S of rows and any $k \ge 3$ (using a similar Calculus argument as above),

$$\sum_{u_1,u_2,\ldots u_k\in S}\text{co-deg}(u_1, u_2, \ldots u_k) = \sum_{j\in C}(\deg_S(j))^k$$
$$\le (d(A))^k|S|^{k/2}\sum_j(\sqrt{j}-\sqrt{j-1})^k \le 2(d(A))^k|S|^{k/2}.$$

However for general k, a clean lower on the sum of correlations is difficult to obtain.

[11] have done some experiments to estimate the number of complete bipartite subgraphs with s vertices on one side and t on the other for various values of s, t (with the motivation that such bipartite subgraphs represent one form of highly connected subgraphs.) These numbers as they observe are hard to compute. Observe that the number of such complete subgraphs is given by

$$\sum_{i_1,i_2,\ldots i_s\in R}\binom{\text{co-deg}(i_1, i_2, \ldots i_s)}{t}.$$

Obviously then these numbers can be expressed as combinations of the numbers

$$\sum_{i_1, i_2, \ldots i_s \in R} (\text{co-deg}(i_1, i_2, \ldots i_s))^t.$$

These are in a sense higher moments of the co-degree numbers. Thus studying the co-degrees will also give us the numbers of complete subgraphs. Of course, the difficulty is there are too many such co-degrees so that it is not possible to compute each one. We outline a way of achieving some savings in the running time in the next paragraph.

In [4],[3], simple randomized algorithms have been developed which estimate the singular values and vectors of the large matrix A by finding such quantities for randomly chosen sub-matrices. At the heart of their proofs is a lemma which says that if we choose at random a small number of columns of A according to a certain probability distribution, then the co-degrees of pairs of rows with respect to A will be close to the co-degrees of those pairs of rows with respect to S (in the sense of sum of squared errors.) We believe this lemma can be extended to a similar statement about higher order co-degrees; so their computation can also be sped up; this will be analyzed in the final paper.

3 Algorithms using Sampling

The Algorithm

The general idea for decomposing the matrix A will be :

Find $\sigma_1(A)$ and the corresponding singular vectors. Then as in Theorem 1, find a dense S, T pair and remove edges between this pair and repeat.

To find the singular values and vectors, we could use the recent randomized algorithms of [3]. Their approach is the following : we pick a row of A with probabilities proportional to the length squared of the row. We repeat this experiment s times independently to get a $s \times n$ matrix S, where we then scale each row to be of length 1. Their result is that from a singular value decomposition of S, we may infer an "approximate" singular value decomposition of A itself. It can be shown that the choice of probabilities - proportional to the length squared- is optimal; also as they point out, by making one pass through the entire matrix A, we may compute the length squared of each row and set up a sampler which will then take only $O(1)$ time per sample.

There is one difficulty in applying this algorithm to our situation here. Our matrix A has about 600 million rows and as many columns (at last count). So the one pass is itself difficult (it seems to involve wading through the 600 million URL's unless one has access to the web graph).

We circumvent this problem by sampling the rows of the matrix uniformly at random. This is feasible since for example, Google allows us to read in a random URL. We first derive from simple first principles, theoretical bounds on the variance of the uniform sampler below (on lines similar to the papers above). As one may intuitively expect, the uniform sampler is not bad compared to the length squared sampler if the lengths of the rows are not too disparate. We will quantify this and then derive actual numerical guarantees on our estimates in the final paper.

The following basic lemma give us a bound on the variance. Note that S is a random sub-matrix of s rows of A (picked independently uniformly). The first trick which makes the bounds work is to compare S^TS and A^TA (entry by entry) which are both $n \times n$ matrices. The second trick is to just sum the variances of all the entries which simplifies the calculations. While both these tricks are very simple, they keep the expressions from getting out of hand.

Lemma 1 *Suppose A is an $m \times n$ matrix with $|A_{ij}| \leq 1$ and we pick s rows uniformly at random from A to form the $s \times n$ matrix S. Then we have*

$$E\left(||\frac{m}{s}S^TS - A^TA||^2\right) \leq \frac{m}{s}\sum_i \left(\sum_j A_{ij}\right)^2 = \frac{m}{s}\sum_i (deg(i)^2).$$

Perhaps of more direct interest is the following Corollary of the lemma which follows directly by applying the Hoffmann-Wielandt Theorem of Linear Algebra to the real symmetric matrices - S^TS and A^TA.

Corollary 1

$$E\left(\sum_t \left((\sigma_t(A))^2 - (\sigma_t(S))^2\right)^2\right) \leq \frac{m}{s}\sum_i \left(\sum_j A_{ij}\right)^2 .$$

These only bound how far apart the singular values of A and S are. We can also argue that if we have the singular vectors of S, we can derive from them, a good low-rank approximation to A. This we will do in the final paper, where we will describe further interpretations of the singular vectors.

Proof of lemma : We first sample with replacement (rather than without) to get S - i.e., for each of the m rows of A, we independently choose a random variable X_i which is 1 with probability $\frac{s}{m}$ and 0 with probability $1 - \frac{s}{m}$ and include row i of A in S if X_i is 1.

Then for each fixed $j, l, 1 \leq j, l \leq n$, we have

$$(S^TS)_{jl} = \sum_{i=1}^{m} A_{ij}A_{il}X_i.$$

So,

$$E((S^TS)_{jl}) = \frac{s}{m}(A^TA)_{jl}$$

and then the independence implies that

$$Var((S^TS)_{jl}) \leq \sum_{i=1}^{m} A_{ij}A_{il}\frac{s}{m}.$$

Summing over all j, l and using $\sum_{j,l=1}^{n} A_{ij}A_{il} = (\sum_j A_{ij})^2$, we get the lemma.

□

References

[1] N. Alon, R. Duke, L. Lefmann, V. Rödl, Yuster, "The algorithmic aspects of the regularity lemma", Journal of Algorithms, **16** (1994), pp 80-109.

[2] Y. Asahiro, and K. Iwama, "Finding Dense Subgraphs" Proceedings of the International Symposium on Algorithms and Computation,1995, Lecture Notes in Computer Science 1004, pp 102-111, 1995.

[3] P. Drineas, A. Frieze, R. Kannan, S. Vempala, V. Vinay, "Clustering in large graphs and matrices" in Symposium on Discrete Algorithms, 1999.

[4] A. Frieze, R. Kannan, S. Vempala, "Fast Monte-Carlo Algorithms for finding low rank approximations" in Foundations of Computer Science, 1998.

[5] U. Feige and M. Seltser, "On the densest $k-$subgraph problem" Weizmann Institute, Technical Report, CS 97-16, Sept. 1997

[6] A. Frieze and R. Kannan, "Quick approximation to matrices and applications" To appear in Combnatorica.

[7] D. Gibson, J. Kleinberg, P. Raghavan, "Inferring web communities from Web topology", Proc. of HYPERTEXT, 1998 pp 225-234.

[8] J. Kleinberg, "Authoritative sources in hypertext linked environments", Proc. of the ACM-SIAM Symposium on Discrete Algorithms, 1998, pp 668-677.

[9] J. Kleinberg, R. Kumar, P. Raghavan, S. Rajagopalan, A. Tomkins, " The web as a graph : measurements, models, and methods", Manuscript ???

[10] J. Komlós and M. Simonovits, "Szemerédi's Regularity Lemma and its applications in Graph Theory" in Combinatorics, Paul Erdös is Eighty (vol. 2), Bolyai Society, Budapst, 1996 pp 295-352.

[11] S. R. Kumar, P. Raghavan, S. Rajagopalan, A. Tomkins, "Trawling emerging cyber-communities automatically", Proceedings of the 8 th WWW Conference, 1999.

[12] E. Szemerédi, "Regular partitions of graphs", in Colloques Internationaux, C.N.R.S. No 260, Orsay, 1976 pp 399-401.