Problem Solving in Computer Science
Course Notes - Lecture 13 (April 26, 2005)

Scribe: Marc A. Schaub
marc.schaub@epfl.ch

1 Groups Presentations

1.1 Group BGS (*presented by Wojciech*)

They considered the graph as a scale free network, a problem which has been widely studied, and in particular they used an algorithm modeling epidemic spreads. In such a model, the probability for an uninfected node to get infected is linearly proportional to the number of infected nodes it is connected to. They applied this heuristic to the current problem by considering the clusters (starting from the anchor nodes) as being competing epidemics. Since this approach turned out to be slow, they then considered an approach based on random walks with for each step a probability of 5% of going back to the seed and a probability of 95% of continuing the walk by choosing a random neighbor. This method is a Markovian process and has been implemented in a fast way (running time of less than 4 minutes on the whole graph). The advantage of these approaches is that do not only provide discrete clustering but also a measure of the probability for a paper to be in each of cluster.

They then showed an SVG plot representing the 2000 most visited nodes of the graph and the clustering found using the aforementioned algorithm. The representations shows that highly connected nodes span clusters, which is a property of scale free networks. Furthermore, the connections and clusters seem to make sense; there are for example several links crossing cluster boundaries while going from computer graphics papers to wavelet theory papers. They are currently able to scale this algorithm to approximately 6000 nodes but they think that this limitation is solely due to the plotting package they have been using.

Nir suggests to maybe make a webpage report instead of a Postscript file.

In parallel, this group is still working on the compact algorithm presented in earlier lectures.

1.2 Group MST (*presented by Grégory M.*)

They noticed that Citeseer seems to be in the process rebuilding the OAI database and show evidence that most records are currently unavailable but are put back online in order. The *IsReferencedBy* links appear to have been removed from the new version even though nobody contacted Citeseer about this issue.

They presented plots of the distribution of node degree both before and after the cleanup steps (as presented during the previous lecture). Both distributions are more extreme than power-law.

They are working on implementing the same compact algorithm as the BGS group.

Tom mentions that the approach taken in the contract algorithm is close to the epidemic algorithm.

They indicate that there is a problem with the normalized cut metric introduced during the last lecture and show a new metric proposed by Nir:

$$c_p^n = \sum_{\substack{(u,v)\in E \\ u\in V_i, v\in V_j \\ i\neq j}} \frac{1}{|V_i||V_j|} \tag{1}$$

Tom says that the previously proposed metric doesn't work and proposes to use the geometric mean as denominator instead. Marc says that he thinks that Nir's metric would do better in certain cases, for example when the graph is split into two unconected components. Tom answers that Nir's metric is designed for a problem that can be seprated in a binary way and is biased, whereas the geometric mean is an unbiased metric. Dirk asks why Tom wants to use the geometric mean. Tom replies that when taking only the product of all cluster sizes, this denominator dominates the metric.

The new official metric to use is:

$$c_p^n = \sum_{\substack{(u,v)\in E \\ u\in V_i, v\in V_j \\ i\neq j}} \frac{1}{\sqrt[K]{|V_1||V_2|...|V_K|}} \tag{2}$$

They present the results obtained runing the local optimization algorithm they had previously presented. It is interesting to note that a random partition already outperforms the naive solution (as presented during last lecture) both using the obsolete normalized and Nir's metric even though there are more than 4 million cut edges. Several local optimizazion runs starting from such random partitions converge to local minimums, with a best unnormalized cut of 48'000. They used random shuffle to escape local minimums, which allowed a single run to improve to an unnormalized minimal cut of 30'000. They think that this result is far from the optimum and

that the method can still be improved. They consider combining the results of this algorithm and the contract using a genetic algorithm.

Tom suggest trying to obtain a cut using a breadth first algorithm which clusters nodes with the anchor to which they are closest. Tom asks the MST group to also look at their solution semantically.

1.3 Group GGS (*presented by Abishek*)

They considered an algorithm [1] which starts with each node being one cluster and then grows them bigger. *Tom suggests that this approach is similar to the contract.* They face the problem of ending with most nodes in one single cluster. They propose to avoid this problem by considering the probability of getting this big cluster in a random graph when choosing which clusters to group together. The original complexity of the algorithm is $O(n^2)$, but by using an AVLG representation for the graph, this complexity is reduced to $O(n \cdot log(n))$. They are currently implementing this algorithm.

They also considered a different approach based on eigenvalues and eigenvectors[2]. Considering the adjacency matrix A, and the probability matrix $P = A\lambda A^T$, it is possible to express the probability matrix after t iterations: $P^t = A\lambda^t A^T$. The problem of this approach is that the computation of the eigenvectors is in $O(n^3)$ and they estimate that they cannot use this method for more than 200'000 nodes.

References

[1] Aaron Clauset, M. E. J. Newman, and Cristopher Moore. Finding community structure in very large networks. *Physical Review E (Statistical, Nonlinear, and Soft Matter Physics)*, 70(6):066111, 2004.

[2] M. Latapy and P. Pons. Computing communities in large networks using random walks. *ArXiv Condensed Matter e-prints*, December 2004.

1.4 Group GGS (*presented by Alex*)

They introduce the notion of bridge and articulation point between two biconnected components. They get a biggest connected component containing all anchors but two.

Tom asks if there are still differences between the graphs. The MST group points out that they are unable to retrieve the records that are not in the OAI archive due to the rebuild mentioned earlier. . In their cleaned up graph, they also have two anchors that are not connected to the other anchors. The GGS group had retrieved these records earlier. Tom decides that all group should only use the graph built based on the archive only, with

the modification regarding IsReferencedBy links that was introduced during previous lectures.

They looked at cylces and found a semi-connecred component with 10 cycles. They didn't look at the actual papers in the cycles.

They are trying to collapse semi-connected components and put them into a partition. They think that it would be interesting to look at the year of publication and cluster old papers first then add newer one using a local optimization method.

Tom says that looking at additional information is fair game. Nir says that it is fine to look at other information, but reminds them to keeps the final measure in mind.

2 Resources for searching the litterature *(presented by Tom)*

- Commercial Metasearch *(available from within the EPFL network)*
  - ISI Web of Knowledge. *(Probably doesn't index conferences, not optimal for Computer Science but is a reference for other sciences where it is used to measure impact.)*
  - Scopus by Elsevier *(More precise than Citeseer and apparently more complete than ISI. Certain publishers might not have a contract with Elsevier.)*
- Publishers *Full indexes of the corresponding journals / conference proceedings. Available within the EPFL network.)*
  - Non-profit
    - ACM
    - IEEE Xplore
    - SIAM
  - Commercial
    - SpringerLink *Often used for conference proceedings*
    - Elsevier ScienceDirect
- Search engines *(Based on website contents. Less precise.)*
  - Google Scholar

3 How to write papers *(presented by Tom)*

Tom introduces this new part of the course which will be emphasized in the upcoming lectures. He presents several useful books:

- A thesaurus, eg. *Webster Collegiate Thesaurus* (online version). All students claim to be already familiar with the concept of a thesaurus. *Nir points out that it is better to use an English-English dictionary than a French-English dictionary, since it will allow to better understand the meaning of a word. He is looking for a good online French-French dictionnary.*

- A book about style, eg. *The Chicago Manual of Style* (Official webpage). There is no standard English style (this is the case for French or German), but many sets of Style manuals (the New York Times has its own for example).

- A short book everybody should read: *The elements of style. Strunk and White*. Online versions available here and here.

3.1 Elementary Principles of Composition (Strunk and White)

12. Choose a design and stick to it

13. The unit of composition is the paragraph

14. Use **active** voice (always use *We* in papers, even if you're a single author. Use *I* only for acknowledgements.)

15. Put statements in **positive** form.

16. Use definite, specific, concrete language

17. Omit needless words

18. Avoid the succession of loose sentences

19. Express coordinated ideas in similar form

20. Keep related words together (eg. *consider a set B of integers* instead of *consider a set of integers B.*)

21. Keep to one tense as much as possible. In papers stick to present: *Paper A shows* instead of *Paper A showed*. This arguable. However do not jump back and forth like in *In the following paragraph we will show blah In sect S we did show bluh.*

22. Place the emphatic word of a sentence at the end.