

Multimode Ring Cohesion Theory

Douglas R. White – work agenda
Contribution to the Social Scaling Workshop
August 28, 2003 V.2; Findings updated Oct 25

ISCOM Network Subgroup Members

Cohesion is a concept long discussed in network analysis, and there have been significant findings on the subject (see Friedkin 1998 for example) but good measures of cohesion have been in short supply (see White and Harary 2001). Predictive Structural Cohesion (PSC) theory is fairly new. The connection between multilevel network analysis and navigability was proposed by White and Houseman (2002). MRC theory is proposed herein (2003).

Program to generate non-redundant fragments

- a. kinship
starting point
- b. biotech

Multimode Ring Cohesion Theory

MRC theory looks at such units of cohesion and the fields in which they are embedded as spectral signatures of types and frequencies of elementary rings. It is rings or cycles that generate cohesion and local clustering in networks. An elementary ring is a subgraph consisting of pure cycles with exactly n nodes and n edges, i.e., there are no cycles within the ring. We do a frequency count of different types of such cycles and study their distributional properties (power-law, exponential). They may be composed of differing kinds of nodes and different kinds of edges or arcs. Because arcs (directed edges) can often be assumed to imply backflow we include fragments that consist of semipaths (ignoring direction to constitute the cycle but keeping track of directions in the labels of the cycles). We can limit length of cycles needed to generate all cycles of whatever length using the counting rule for independent cycles, and keeping all the smaller cycles needed to generate an independence-set basis for the model graph.

The claim for a “predictive theory” of multimode ring cohesion derives from the fact that the independent cycles are necessary and sufficient to generate all of the observed cohesion in all of the k -components observed in a model networks, so the frequency distributions of all possible shorter independent cycles allows us to study how cohesion is generated, e.g., is there any preference gradient or other underlying process (e.g., preference for diversity) that generates the observed model graph structure. A random graph with the same number of edges and nodes of each type would be expected to generate an exponential distributions of ordered ring frequencies. Power-law distributions for frequencies of cycles are the ones that suggest self-organizing properties of cohesion and unit formation in any given network. I.E., this is how we treat the measure of cohesion in networks with multiple kinds of relationships. MRC theory predicts power-law distributions on ring types that are involved in preferential choice or other non-random processes and that are thus constructive of PSC units.

Preliminary results: Domains of datasets and the testing of PSC, Navigability and MRC theories

Kinship. In the case of kinship multiple relations are defined with respect to ego— i.e., Fa, Mo, FF, FM, MF, MM, FFB, and so forth—that are the component relationships of cohesive

marriages. E.g. a FaBrDa marriage is one where ego's FaFa=spouse's FaFa. The component relationships are what makes each such cycle different, i.e., these marriages are rings of multimode relations composed of elementary relations Fa,Mo,Br,Sis,So,Da,Hu,Wi. These relations also have reciprocal classes Sib={Br,Si}, Par={Fa,Mo}, Ch={So,Da} with reciprocals Sib⁻¹=Sib, Par⁻¹=Ch, and Hu⁻¹=Wi.

Preliminary Research Findings and extensions: Examining a dozen cases so far (White and Houseman 2002) finds societies divided into two main clusters with opposing characteristics:

A) Societies with high frequencies of consanguineal (blood) marriages show power-laws distributions on consanguineal rings (blood marriages) and exponential distributions on two- family affinal rings (affinal marriages). **Empirical proposition:** These societies have preference gradients on blood marriages, but none on other types of relinking marriages.

B) Societies with prohibitions on consanguineal (blood) marriages show power-laws distributions on two- family affinal rings (affinal marriages) and exponential distributions on consanguineal rings (blood marriages). **Empirical proposition:** These societies have preference gradients on two-family relinking marriages, possibly in connection with class and status endogamy preference gradients, but no preference gradients on blood marriages.

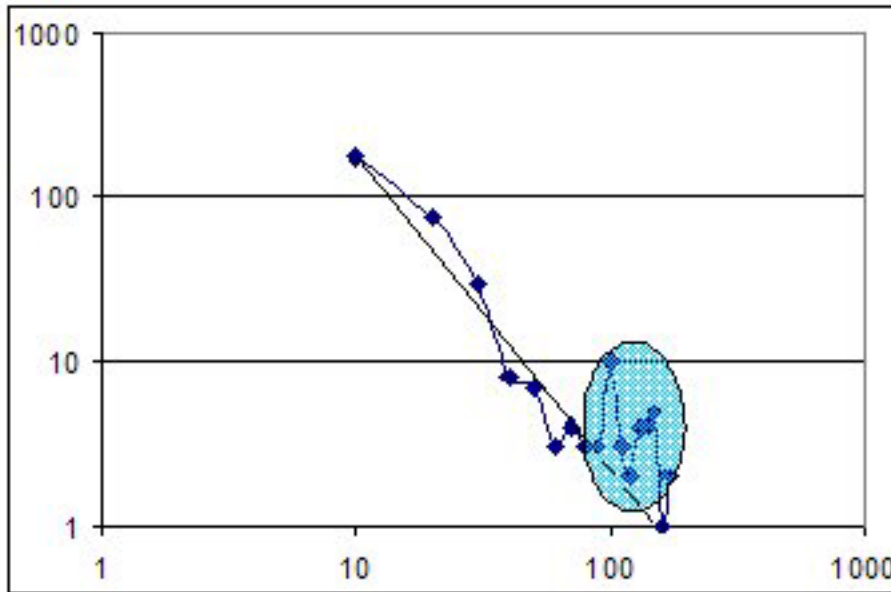
C) Navigability topologies examined for one case, the Turkish nomads. **Empirical proposition:** (1) Societies with high frequencies of blood ties (and power laws on their distributions) have navigable strong tie networks where strong ties are defined by reciprocal marriage exchanges between families (e.g., sublineages). (2) Societies lacking high frequencies of blood ties (and power laws on the distributions of affinal relinking) have navigable weak tie networks where weak ties are defined by nonreciprocal marriage links between families (e.g., kindreds).

Business. In the business world multimode cycles would be defined by different types of ego/partner or ego/alter or ego/corporate or other type of entity, or entity/entity relations. So we would take apart a bipartite corporate interlock network for example in terms of the frequency distributions of different types of elementary (but heterogeneous) rings or cycles. And so forth.

Preliminary Research Findings: Examining a dozen years of inter-corporate contract networks in the worldwide biotechnical industry, the diversity of partner experience and a variety of measures of structural cohesion (often interacting with other variables) are the primary predictors of new ties and of repeat ties. **Empirical proposition:** Presumed preference gradient for ring diversity and relatively short rings.

The 220 independent cycles among 472 biotech organizations and their partners (2700 organizations) were examined. As hypothesized, a power-law distribution was found for the frequency histogram (slope =1.9). Outliers, even to this curve, were found for Biotech-Government-Biotech triads, with all four activity types roughly equal in the contribution of edges. This shows how Government agencies such as NIH are supernumerary hubs of the network. Such hubs also produce an unusually high clustering coefficient. Further analysis will break out the histogram of frequencies of Biotech-Government-Biotech triads by specific agency and type of activities. All 15 combinations occur for the four types of edges, which indicates agency support for all the permutations of activity types.

Figure 1:



Node Color = Organizational Form

- 1 Cyan = Biotechnology Firms
- 2 Orange = Public Research Organizations
- 3 Yellow = Large Pharmaceutical/Chemical Corporations
- 4 Brown = Government Agencies
- 5 Gray = Venture Capital Firms
- 6 White = Other Organization

Tie Color = Activity Type

- 1 Red = Research & Development
- 2 Green = Finance
- 3 Blue = Commercialization
- 5 Magenta = Licensing

Civilizations. Here the entities are cities, states and empires; the primary types of relations involve types of conflict, trade, information flow, and material flow; the attributes of the nodes are longitudinal measures of demographic variables, warfare and regime change. Many of these variables are available within the community of world systems or civilizational scholars. Many of the dynamical processes at the level of cities and states have been studied.

Preliminary Research Findings: Peter Turchin (2003) Examining cohesion impressionistically in world historical dynamics, converges independently to propositions of PSC theory. Ring structure has not yet been examined.

Social Class and Government. Multimode rings are defined by political relationships. See again Turchin (2003).

Preliminary Research Findings: Social class boundaries have been found through relational cohesion methods in numerous cases, studying property and marriage patterns. In several cases we have intercolated property and marriage patterns with officeholding and identified powerholding social classes.

Mixed. Kinship, business, government and civilization networks can be mixed in complex forms.

Multimode Ring Cohesion Method

Ring structure is studied by distributional methods. To compute ring distributions we

1) create network images (e.g., using program Pajek) of every possible type of ring of relatively short circumference or of theoretical interest, but always taking all possible types of ring within the delimited domain.

2) Iteratively take each ring type, called a **ring fragment** of a network, and search for all instances in the total population network, and register its **frequency**. Each set of nodes that constitutes a ring fragment is also registered by its **ring type**.

3) Order the ring types by frequency and plot this **frequency order** by **number of ring types** with this frequency, registering also the ring types. The distribution is then tested as to whether it is **power-law** or **exponential**.

4) Separate **ring classes** (similar sets of ring types) may also be designated on theoretical grounds and tested for distributional characteristics.

5) In the case of power-law distributions, it is of interest to determine whether the **frequency order** of the higher-frequency ring-types (possibly in a ring class) has a **predictable gradient**, e.g., composed by weighted variable such as ring diameter, heterogeneity or composition.

Recall that in a connected graph (possibly with different kinds of nodes and edges that for all sorts of different combinations may make up a ring) with m edges and n nodes, the number of independent cycles is $k = m - n + 1$ ($n-1$ edges are used to make a spanning tree, the cycle theorem is that each additional edge creates an independent cycle). Therefore we need only to find up k undifferentiated cycles of the smallest length L to have a means of finding all differentiated cycles of length L .

Methodological Developments

The Pajek software package for large network analysis has the capability of finding all possible fragments in graphs of a given type specified by different kinds of semicycles in weighted arcs and edges. Although this is an NP-hard problem in exact form, it is done fairly easily in the way that Andrej Mrvar has written the backtracking algorithm. For each node corresponding to a given ego in the fragment ring formula (after eliminating all fragment isomorphisms) follow the path using tree backtracking until all such rings are found, adding one each time a ring is found to the vector for that fragment at the location of this ego.

Pajek generates the frequencies. The input file of fragments marks the fragment ego with label "1" and stacks as *.net files in an overall *.paj file. For this project, Mrvar has made the fragments option iterative, so for a given model graph as input that a whole series of fragments frequencies (and ego identifiers) can be calculated at once and stored to an output file. The output is a series of vectors, one for each type of fragment showing 1s in the vector for marked egos in the model file. From the vector output file we can recover all the fragment frequencies and locations in the model graph.

Bibliography

1. Older Work on Cohesion

Richard D. Alba and Charles Kadushin, The intersection of social circles: A new measure of social proximity in networks, *Sociological Methods & Research* 5 (August, 1976): 77-102.

Bourgeois, M. and Noah E. Friedkin. 2001. The Distant Core: Social Solidarity, Social Distance, and Interpersonal Ties. *Social Networks* 23: 245-260.

- Friedkin, Noah E. 1983. Horizons of Observability and Limits of Informal Control in Organizations. *Social Forces* 62:54-77.
- 1984. Structural Cohesion and Equivalence Explanations of Social Homogeneity. *Sociological Methods & Research* 12: 235-261.
- 1993. Structural Bases of Interpersonal Influence in Groups: A Longitudinal Case Study. *American Sociological Review* 58: 861-872.
- 1998. *A Structural Theory of Social Influence*. New York: Cambridge University Press.
- Harary, Frank, Robert Z. Norman, and Dorwin Cartwright. *Structural Models: An Introduction to the Theory of Directed Graphs*. New York: Wiley, 1965. They define four categories of reachability (connectivity category) in a directed graph: unilateral (one or the other of each pair can reach the other on a directed path), weak (every node can reach any other through a semipath, and strong (every node can reach any other through a directed path), and disconnected (there is some node that cannot reach another through a semipath). In contrast, connectivity level as used in this paper corresponds to the type of connectivity in Menger's Theorem (Harary 1969), e.g., the number of nodes whose removal is needed to disconnect a graph, and the minimum number of node-independent paths between pairs of nodes in the graph (White and Harary 2001).

2. Newer Work on Cohesion

- Bearman, Peter, Robert Faris and James Moody. 1999. Blocking the Future: New Solutions for Old Problems in Historical Social Science." *Social Science History*. 23:4:501-533,
- 2002. Networks and History. Special Issue on Networks and Complexity. *Complexity* 8(1):61-71. Uses bicomponents as an elementary measure of cohesion.
<<http://eclectic.ss.uci.edu/~drwhite/Complexity/SpecialIssue.htm>>
- Brudner, Lilyan A., and Douglas R. White. 1997. Class, property and structural endogamy: Visualizing networked histories. *Theory and Society* 26:161-208. Uses bicomponents as an elementary measure of cohesion in class structure and marital (structural) endogamy.
- Moody, James, and Douglas R. White. 2003. Social Cohesion and Embeddedness: A hierarchical conception of social groups. *American Sociological Review* 68(1):1-25.
- Powell, Walter W. Douglas R. White, Kenneth W. Koput and Jason Owen-Smith. Network Dynamics and Field Evolution: The Growth of Interorganizational Collaboration in the Life Sciences. Submitted to: *American Journal of Sociology*.
- Roehner, Bertrand.
- Turchin, Peter. 2003. *Historical Dynamics: Why States Rise and Fall*. Book Ms. Princeton: Princeton University Press, Princeton Studies in Complexity.
<<http://www.eeb.uconn.edu/faculty/turchin/hd.html>>.
- White, Douglas R. 1997. Structural endogamy and the network *graphe de parenté*. *Mathématiques, informatique et sciences humaines* 137:101-125.
- 1999. Controlled Simulation of Marriage Systems, *Journal of Artificial Societies and Social Simulation* 3(2) <<http://jasss.soc.surrey.ac.uk/2/3/5.html>>
- 2002a. Emergence, transformation and decay in pastoral nomad socio-natural systems. To appear in *Emergence, Transformation and Decay in Socio-Natural Systems*, edited by Sander van der Leeuw, Uno Svedin, Tim Kohler, and Dwight Read.
- ed. 2002c. Special Issue on Networks and Complexity. *Complexity* 8(1).
<<http://eclectic.ss.uci.edu/~drwhite/Complexity/SpecialIssue.htm>>.
- 2003. Ties, Weak and Strong. Encyclopedia of Community, forthcoming. Edited by Karen Christensen and David Levinson. Thousand Oaks, CA: Sage Reference.
- White, Douglas R., Vladimir Batagelj and Andrej Mrvar. 1999. Analyzing Large Kinship and Marriage Networks with Pgraph and Pajek. *Social Science Computer Review* 17(3):245-274.
- White, Douglas R., and Frank Harary. 2001. The Cohesiveness of Blocks in Social Networks: Connectivity and Conditional Density. *Sociological Methodology* 2001 vol. 31:305-359. Boston: Blackwell Publishers.
- White, Douglas R., and Michael Houseman. 2002. The Navigability of Strong Ties: Small Worlds, Tie Strength and Network Topology. *Complexity* 8(1):72-81. Special Issue on Networks and Complexity.
- White, Douglas R., and Paul Jorion. 1992. Representing and analyzing kinship: A Network Approach. *Current Anthropology* 33:454-462.
- 1996 Kinship Networks and Discrete Structure Theory: Applications and Implications. *Social Networks* 18:267-314.
- White, Douglas R., Walter W. Powell, Jason Owen-Smith and James Moody, 2003. Network Models and Organizational Theory: From embeddedness to ridge structures. In preparation for *Computational and Mathematical Organization Theory*, special issue on Mathematical Representations for the Analysis of Social Networks within and between Organizations, guest edited by Alessandro Lomi and Phillipa

Pattison.

White, Douglas R., and Thomas Schweizer. 1998. Kinship, Property Transmission, and Stratification in Javanese Villages. pp. 36-58, in, Schweizer, Thomas, and Douglas R. White, Eds. 1998. *Kinship, Networks and Exchange*. Cambridge: Cambridge University Press.

3. Navigability

Dodds, Peter S., R. Muhamad, and Duncan J. Watts. 2003. An experimental study of social search and the small world problem. *Science* 301:827-829. <www.sciencemag.org>

Kleinberg, Jon. 2000. Navigation in a small world. *Nature* 406:845.

OTHER:

White, Douglas R., and Karl Reitz. 1983 Graph and Semigroup Homomorphisms. *Social Networks* 5:193-234.

White, Douglas R., and Patricia Skyhorse. 1998. Parenté Suite: User's Manual for Analysis of Kinship and Marriage Networks. In, V. Burton, T. Finnigan, D. Herr, Eds., *Multimedia Renaissance in Social Science Computing*. CD-ROM. Urbana, IL.

An example (test version)

From andrej.mrvar@Uni-Lj.si Fri Aug 8 15:25:45 2003
Date: Sun, 08 Aug 2004 22:10:10 +0200
From: Andrej Mrvar <andrej.mrvar@Uni-Lj.si>
To: Douglas R. White <drwhite@orion.oac.uci.edu>
Subject: Try this

Hello!

I made modification in Pajek and prepared file for running all fragment searching.

Some instructions (this is for a kinship example):

1. Run Pajek

1. Select appropriate options to read genealogy as Ore graph:

Options/ReadWrite/ GEDCOM p-graph - not marked
Options/ReadWrite/ Ore 1-male, 2-female links - marked

2. Run logfile allfragmen.log that is attached, using:
File/RepeatSession and select allfragmen.log

3. On the first question you must select Pajek project file with fragments, that you prepared and I made some modifications (pjk0001.paj).

4. On the second question select network with genealogy stored in GEDCOM file (like p-tur.ged).

5. After some running you will get a lot of vectors in Pajek.

All vectors are also written to text file VECTORS.VEC in the directory where you selected the GEDCOM file (the first four columns are INDI, BIRTH, MARR, DEATH and than 227 vectors for your fragments).

There is one more request: since in all your fragments the first person is male (triangle) also in the gedcom file the first person must be male, otherwise you will get only zeros in all vectors. If the first person in GEDCOM file is a woman, run allfragwom.log instead of allfragmen.log.

I hope the explainaion is clear enough, and you will get the results you need. In the attached file you will find

- new Pajek.exe
- paj0001.paj - your list of fragments
- allfragmen and allfragwom: sessions for several fragment searching

Best,

Andrej

[Part 2, Application/X-ZIP-COMPRESSED 852KB.]

Pseudocode for Ring-Fragment Finder

Possibly a Java program can be written for JUNG, the newly released open-source Java Universal Network/Graph Framework (see email below), which will utilize Pajek graph formats and a *.paj file with ring fragments, and perform these analyses. In any case, the rectangular data output (a series of $n \times m$ vectors for n nodes and m fragments) can be used by other programs using the network *.net file being modeled to do further investigation if needed.

Read Model graph G ; Record Fragment F as ring of length L ;

```
for each  $i$  in  $G$ 
  do DFS depth  $k$  to  $L$ 
  if  $DFS(k) \neq F(k)$  back in DFS
  if  $(k)$  already traversed in this  $i$ - $k$  path back in DFS
  found  $i$  at depth  $L$ 
  get induced subgraph; if  $\#edges > L$  backtrack
  if not geodesic backtrack
   $V(i) = V(i) + 1$ 
  end DFS
end for.
```

The following email is to a candidate for the Post-Doc position to work on this project in Paris.

To: Joseph WEHBE <j.wehbe@wanadoo.fr>,
Michael Houseman <houseman@attglobal.net>

Cc: Sander van der Leeuw <vanderle@santafe.edu>,
Henry Wright <hwright@umich.edu>, David Lane <lane@unimo.it>,
Geoffrey West <gbw@santafe.edu>, Douglas R White <drwhite@uci.edu>,
Andrej Mrvar <andrej.mrvar@UNI-LJ.SI>

Dear Joseph (if I may - the help is needed from Andrej re: previous msg below; cc: other members of projects re: various applications)

In case you are wondering what we will be doing together during the year if you are appointed (which I think is likely) there are a variety of things that are methodologically related but substantively different. One (see below) is on large scale population networks via genealogical, spatial and occupational data). Another is business and organizational data. A third is the network evolution of the world's cities, states and empires, together with trade and warfare networks. Sander (one of the leaders of our project) may see other applications as well. From these analyses the group of researchers involved in these studies believes there will come a groundbreaking series of studies.

It is time that you read (if not already) <http://eclectic.ss.uci.edu/~drwhite/Complexity/K&C-a.pdf> to get a sense of this methodology applied to large scale population networks via genealogical links, and some of the discoveries we have made there to date.

The extension of Pajek that I propose for testing Multimode Ring Cohesion (MRC) theory will enable us to do 100 or so comparative studies in rapid time of different large-population studies that we have already done. I.e., each analysis consists of loading a dataset, punching in the option for fragment finding, and offloading the results. Not difficult. Then stage two consists of reading in the say, 30 most interesting cases into excel and doing the sums to get the frequencies and then doing the univariate distributions of those frequencies.

Michael Houseman, my colleague in Paris, is used to doing this latter step and can go through that step with you directly. It is also fairly quick and straightforward.

Other similar applications will be made in other domains of study in the ISCOM group and the SFI working conferences on macro evolution (cities, civilizations).

Best,

Doug White <http://eclectic.ss.uci.edu/>

"Every belief must be weighed against the cost of its consequences."

----- Forwarded message -----

Date: Sat, 2 Aug 2003 13:16:17 -0700 (PDT) From: Douglas R. White <drwhite@orion.oac.uci.edu> To: Andrej Mrvar <andrej.mrvar@UNI-LJ.SI>, Batagelj <Vladimir.Batagelj@uni-lj.si> Cc: Michael Houseman <houseman@attglobal.net>, Laurent Barry <barry@ehess.fr>, Scott White <scott@insegic.com>, Douglas R White <drwhite@uci.edu>
Subject: here is my latest project - need a bit of help

Dear Andrej and Vlado

If you have had a look at my article with houseman at <http://eclectic.ss.uci.edu/~drwhite/Complexity/SpecialIssue.htm> you will see we have discovered some power-law organizational differences in different types of kinship system.

I am now using Pajek to search for fragments (the enclosed file-gives all blood marriages up to seventh cousins) in a large Ore graph using line value matching. It would be nice in cursing through these graphs if the label for each (e.g. the first is FBD) would show somewhere (they are entered as text after *Vertices N [i.e., title here]).

What I want is a single command in Nets that lets me take a series of First Networks and find all the subgraphs of each type in the main Second Network. I think this is a natural thing to do since most problems do not concern simple one type of fragment but many different fragments. It is simple an iterative generalization of the existing option.

In the case of blood marriages there is a single lowest node in each digraph. the natural output is a k vectors where k is the number of subgraphs, and where in each vector for a given subgraph, all nodes that are the single lowest node in a matching fragment are coded 1 and the rest 0. One would simple output those vectors (say into Excel) for further analysis of the results.

We are going to be doing this recurrently for about 100 main graphs.

In the next phase we will have two lowest nodes (the lowest nodes are those with no descendants) and two sets of k vectors will be needed as output (or else 2 adjacent vectors for each fragment). Note that as a natural result a user could use this vector in a graphic display to see where the fragments occur (with by converting to a partition and coloring or else by Graph-Vector showing up as different sizes).

Conceivably one could do three lowest nodes, or even four... though there is a limit to which such analyses are meaningful, although we have names for these things and they turn out to be highly significant. Blood marriages have one, 2-family relinking have 2 and so forth. I now have a little FORTRAN program that converts strings like FBD (father's brother's daughter marriages) into

Pajek graphs for the study of the frequencies of such fragments. I can extract those frequencies from the vectors of course, must summing the 1s.

I have discovered that Pajek's fragment-finder is a much better way do the kinship-cycle analysis (of such fragments, and their power-law or exponential distributions). This is because in my earlier Par-Calc you were allow to find a marriage type even if it was a cycle which contained two or more subcycles. Pajek's method is much more refined, as it does not allow this, thus you get a marriage type classification in pure form.

I would much appreciate if you could conceive a way for me to do this as a standard option or even with a macro, which is hard for me even to envision.

You can try this file out for example with and GEDCOM file read as an Ore graph.

thanks. I guess we will be seeing one another again in Ljubljana at the van der Leeuw conference. I recommended the three of you for that conference and think that you and Pajek will be of great help in that endeavor as well.

----- Forwarded message -----

From scottw@eml.cc Mon Aug 4 16:20:06 2003
Date: Mon, 04 Aug 2003 14:42:55 -0800
From: scott white <scottw@eml.cc>
To: drwhite@uci.edu
Subject: Fwd: JUNG 1.0 released

On Sat, 02 Aug 2003 07:27:17 -0800, from "scott white" <scottw@eml.cc>: I would like to announce the release 1.0 of the open-source Java Universal Network/Graph Framework described below. Our website can be visited at <http://jung.sourceforge.net>. It contains more information about the project.

Note: We are currently looking for help in the area of social networks algorithms and routines. Please see our FAQ (<http://jung.sourceforge.net/faq.php#contribute>) for more information on how one might go about contributing code to our project.

best,
scott

Scott White
PhD Student
Information & Computer Science, UCI

JUNG — the Java Universal Network/Graph Framework--is a software library that provides a common and extendible language for the modeling, analysis, and visualization of data that can be represented as a graph or network. It is written in Java, which allows JUNG-based applications to make use of the extensive built-in capabilities of the Java API, as well as those of other existing third-party Java libraries.

The JUNG architecture is designed to support a variety of representations of entities and their relations, such as directed and undirected graphs, multi-modal graphs, graphs with parallel edges, and hypergraphs. It provides a mechanism for annotating graphs, entities, and relations with

metadata. This facilitates the creation of analytic tools for complex data sets that can examine the relations between entities as well as the metadata attached to each entity and relation.

The current distribution of JUNG includes implementations of a number of algorithms from graph theory, data mining, and social network analysis, such as routines for clustering, decomposition, optimization, random graph generation, statistical analysis, and calculation of network distances, flows, and importance measures (centrality, PageRank, HITS, etc.).

JUNG also provides a visualization framework that makes it easy to construct tools for the interactive exploration of network data. Users can use one of the layout algorithms provided, or use the framework to create their own custom layouts. In addition, filtering mechanisms are provided which allow users to focus their attention, or their algorithms, on specific portions of the graph.

As an open-source library, JUNG provides a common framework for graph/network analysis and visualization. We hope that JUNG will make it easier for those who work with relational data to make use of one another's' development efforts, and thus avoid continually re-inventing the wheel.

— The JUNG Framework Development Team

Places to start programming iterative Subgraph Isomorphism Problem?

Subgraph Isomorphism in Polynomial Time (1995) (4 citations)

B.T. Messmer, H. Bunke

<http://citeseer.nj.nec.com/update/87918>

citeseer.nj.nec.com/cache/papers/cs/77/ftp:zSzzSziampftp.unibe.chzSzpubzSzTechReportszSz1995zSziamp-95-003.pdf/messmer95subgraph.pdf

Abstract: In this paper, we propose a new approach to the problem of subgraph isomorphism detection. The new method is designed for systems which differentiate between graphs that are a priori known, so-called model graphs, and unknown graphs, so-called input graphs. The problem to be solved is to find a subgraph isomorphism from an input graph, which is given on-line, to any of the model graphs. The new method is based on an intensive preprocessing step in which the model graphs are used to create a...

See David Eppstein bibliography at

<http://www.ics.uci.edu/~eppstein/bibs/subiso.bib>

less relevant?: B.T. Messmer, H. Bunke

Fast Error-correcting Graph Isomorphism Based on Model Precompilation (1996) (6 citations)

<http://citeseer.nj.nec.com/cache/papers/cs/15293/ftp:zSzzSzftp.iam.unibe.chzSzpubzSzTechReportszSz1996zSziamp-96-012.pdf/messmer96fast.pdf>

In this paper we present a fast algorithm for the computation of errorcorrecting graph isomorphisms. The new algorithm is an extension of a method for exact subgraph isomorphism detection from an input graph to a set of a priori known model graphs, which was previously developed by the authors. Similarly to the original algorithm, the new method is based on the idea of creating a decision tree from the model graphs. This decision tree is compiled off-line in a preprocessing step. At run...

Gemini

Gemini is a program that compares two circuit netlists and reports whether they are exactly the same, pinpointing differences if there are any. Gemini is typically used to determine whether a VLSI circuit layout is correct by comparing the wirelist extracted from the layout to the specification wirelist. This is also known as LVS - layout vs. schematic. Gemini is a program based on a well-known graph isomorphism algorithm that is very efficient for comparing circuits encountered in practice. Gemini's running time is almost linear in the size of the circuits: about $K \cdot (N/1000)^{1.07}$ seconds, where N is the number of transistors, and only slightly worse for very symmetric circuits. (K depends on your processor, but is substantially less than 1 for current workstations.) The latest version of Gemini has the following features:
Circuit input is in standard sim format with an extension that allows user-defined devices. This allows Gemini to accommodate 4-terminal devices and technologies other than digital CMOS.
Collapses series (chains) and parallel (fingers) transistors.
Creates a Magic feedback file that marks differences in the layout itself.
The Gemini program is available via anonymous ftp. For details, contact larry@cs.washington.edu

SubGemini

SubGemini is a program that uses a general subgraph isomorphism algorithm to find subcircuits in a larger circuit. While the subgraph isomorphism problem is known to be NP-complete in general, the SubGemini algorithm can efficiently solve this problem for most circuits encountered in practice by making use of their inherent structure. The SubGemini program exists only as a prototype proof-of-concept and thus is not generally available. Please contact us (ebeling@cs.washington.edu) if you have interest in obtaining the source code.

[Home](#) | [Notice Board](#) | [Instructor](#) | [TAs](#) | [Students](#) | [Assignments](#) | [Schedule](#) | [Lectures](#) | [Resources](#)
Department of Computer Science and Engineering, IIT Kanpur
CS245: Algorithms
Dr. R. K. Ghosh

Details of Project on subgraph isomorphism.

The subgraph isomorphism problem is a well known problem for algorithm engineers. It is, given a graph G and a graph g , to verify whether " G has a subgraph which is isomorphic to g " or not. It is an NP-complete problem.

But for some special cases (if G is a tree or planar), it had been shown that, subgraph isomorphism problem can be solved in polynomial time. But in real world applications graphs may not be trees (or planar) always. So the current research interest is to develop some heuristic algorithms for such NP-complete problems, which have high probability of success.

So, the goal of this project is to come up with some heuristic approximation (polynomial) algorithm for the problem of subgraph isomorphism, which runs with high probability of success.

Materials

An introduction to subgraph isomorphism and related problems (slides) pdf

[Home](#) | [Notice Board](#) | [Instructor](#) | [TAs](#) | [Students](#) | [Assignments](#) | [Schedule](#) | [Lectures](#) | [Resources](#)

From andrej.mrvar@Uni-Lj.si Wed Aug 6 12:32:28 2003
Date: Thu, 06 Aug 1998 20:37:37 +0200
From: Andrej Mrvar <andrej.mrvar@Uni-Lj.si>
To: Douglas R. White <drwhite@orion.oac.uci.edu>
Subject: The same message again

Hello!

>I am now using Pajek to search for fragments (the enclosed file-gives all
>blood marriages up to seventh cousins) in a large Ore graph using line
>value matching. It would be nice in cursing through these graphs if the
>label for each (e.g. the first is FBD) would show somewhere (they are
>entered as text after *Vertices N [i.e., title here]).

You should put label to the *Network statment, e.g.

```
*Network pjk0001.net FBD  
*Vertices 5
```

.....

>In the case of blood marriages there is a single lowest node
>in each digraph. the natural output is a k vectors where k is the number
>of subgraphs, and where in each vector for a given subgraph, all nodes
>that are the single lowest node in a matching fragment are coded 1 and the
>rest 0. One would simple output those vectors (say into Excel) for further
>analysis of the results.

If I understand you correctly, you would like to put 1 in vector
for vertices that match the vertex that you mark in Pajek file as:

```
3 " " triangle x_fact 5 y_fact 5
```

I would suggest to mark this vertex with number and put it as label of
vertex. This label (number) is later used in vector. Therefore I would suggest
to use:

```
3 "1" triangle
```

Than I can write extension of Pajek fragment searching, that will mark matching vertices
using this labels (numbers) in vector. What do you think?
The suggested (changed) Pajek file is attached.

Andrej

THIS MUCH WILL WORK FINE

***in the output vector for each fragment, please ADD a 1 to each column found
to be the EGO of a new fragment.***

I DONT NEED THIS:

In this case you can mark several vertices, e.g. in the case of two lowest vertices, one
lowest vertex with 1, the second with 2.

REASON:

your backtracking algorithm with the frequency counter in the vector for each
fragment type will take care of my counting problem.

I can write my own little program to take your vectors and look, person by
person, for example, at how many fragments (and how many of each type) and
reconstruct from the FORMULA for the fragment using a depth first for that
formula, keeping the cycles that match. Can do lots of other things as well.

As Kleinberg (2000) was the first to show, only a limited range of network topologies will satisfy the designation of navigability. His model, in fact, required a perfect fit between the distance decay parameter k for probability of ties in a space of dimension k' , where navigability is possible only where $k=k'$. Watts, Dodds and Newman (2002) showed a more robust range of network topologies (*unpack – explain why important*). White and Houseman (2002:73-74):

“sought to identify a family of realistic social network models for complex small worlds with strong upper limits on how many links an individual may possess. They imbue the actors in their network models with social identities. Social distance between pairs of individuals is then defined by differences in the taxonomically organized categories of identity. Like Kleinberg, they find that the ability to search and find specific targets depends on the network having not only short network distances, but also links constructed with probabilities that decay exponentially with social distance. By tuning the exponential parameter for social-distance decay of link probability, their family of models generate networks that have searchability as well as short average network distances, and they match up to describe the results of Milgram’s original SW experiment”

The SSW networks of Watts, Dodds and Newman demonstrate some further properties. Searchability increases when the hierarchies of identity are multiple rather than singular, and when these multiple identities cross-cut one another, in the sense of statistical independence. This allows one step in a search to be taken on the basis of one aspect of the target’s identity, while a next step might be taken on the basis of another aspect. Such cross-cuts move much more quickly towards the target since they move the search out of a cluster of ties in the network that reflects similarities on only one attribute (for which there may be many independent clusters) towards clusters that have many of the target’s

attributes, and in which local ties in the cluster are more likely to lead directly to or close to the target. The introduction of multiple social dimensions leads to a much more robust result—i.e., networks are searchable for a broad range of parameters— which is very different from Kleinberg's singular condition. For navigability they require not just social identities, however, but a network that is constructed with distance decay across the proximities defined by similarities in identities.