

1^{er} Cours : Introduction MPRI 2013–2014

Michel Habib
habib@liafa.univ-Paris-Diderot.fr
<http://www.liafa.univ-Paris-Diderot.fr/~habib>

Sophie Germain, septembre 2013



Academic year : 2013–2014

- ▶ Research team : Distributed algorithms and graphs, LIAFA :
- ▶ Part I
Practical graph algorithms
Michel Habib (Pr Univ. Paris Diderot)
- ▶ Part II
Treewidth
Structure theorems (minors theory and graph decompositions
Robertson and Seymour)
Pierre Charbit (MdC, Univ. Paris Diderot)
- ▶ contact : Prenom.Nom@liafa.univ-Paris-Diderot.fr

Programme détaillé de la partie I

- ▶ Graph Classes and (geometric) representations
- ▶ Partition refinement tools.
- ▶ Chordal graphs.
- ▶ A theory on graph searches
- ▶ Graph decompositions (modular decomposition, join ...).
- ▶ Important graph parameters : **treewidth**, rankwidth, cliquewidth.

Typical questions

- ▶ Recognition algorithms, if possible linear-time and **certifying**.
- ▶ Efficient decomposition algorithms
- ▶ Computations of compact encodings and representation
- ▶ Random generation and enumeration
- ▶ Routing protocols and diameter estimation
- ▶ Computation of some invariant (for example min coloring or max clique).

Schedule of this introduction course

Our research group

Cographs

Graphs are good models

How to define a good graph algorithm ?

Themes

- ▶ Bioinformatics (mainly phylogeny and other graph problems)
- ▶ Networks and Distributed systems
- ▶ Analysis of huge graphs in social sciences (ranking, clustering)

GANG

An INRIA–LIAFA project on graphs an networks, dir. Laurent Viennot

SAE

Systématique, Adaptation, Evolution : A biological group at UPMC with Eric Baptiste and Philippe Lopez

Social sciences

ANR Project AlgoPol (Politics of Algorithms), a collaboration with Dominique Cardon (Orange Lab) and Christophe Prieur (LIAFA)

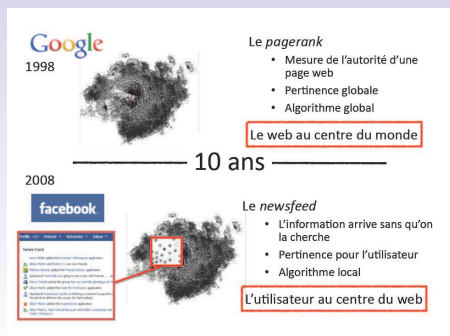
Programme II

- ▶ Algorithms for cocomparability graphs and partially ordered sets
- ▶ Applications
 - ▶ Computations in networks (flow, diameter, pagerank ...)
 - ▶ Parametrized complexity
 - ▶ Phylogenetics and other problems from comparative genomics.

Goals

- ▶ Reach a good level of knowledge on graph algorithms (many students FAQ)
- ▶ Understand the greedy algorithms
- ▶ Try to understand why some simple heuristics works for most practical data
- ▶ Understand how to use structural results to design algorithms
- ▶ Panorama of research

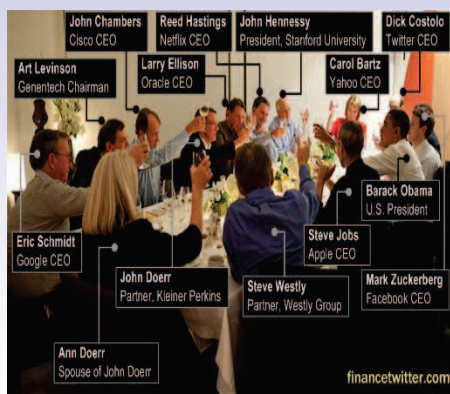
Google et Facebook



Next challenges

- ▶ Merging the Web and social networks ?
- ▶ Recommender Algorithms (bipartite graph clustering) or reasoning with heterogeneous data.

Big Capital !



Some applications

1. Shortest paths computations in Google maps
2. Community detections in Big data (NSA, CIA, FBI ...but also Google, Facebook, Carrefour, ...)
3. (Un)Fortunately this course provides tools to solve these two practical problems

Practical issues

Graphs made up with vertices and edges (or arcs) provide a very powerful tool to model real-life problems. Weights on vertices or edges can be added.

Many applications involve graph algorithms, in particular many facets of computer science !

Also many new leading economical applications **Google PageRank** and **FaceBook** are graph based

Theoretical side

Why such a course ?

Fagin's theorems in structural complexity

Characterizations of P and NP using graphs and logics fragments.

NP

The class of all graph-theoretic properties expressible in existential second-order logic is precisely NP.

P

The class of all graph-theoretic properties expressible in Horn existential second-order logic with successor is precisely P.

Which class of graphs? First example the Cographs

Complement-reducible graphs

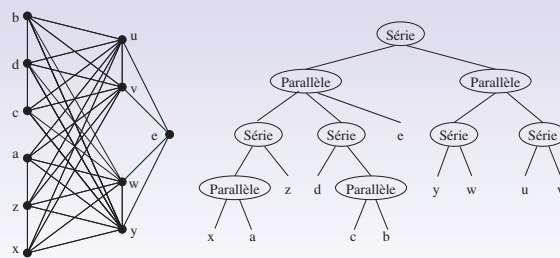
S. Seinsche. On a property of the class of n-colorable graphs. Journal on Combinatorial Theory (B), 16 :191-193, 1974.

Recursive Definition

The class of cographs is the smallest class of graphs containing $G_0 = K_1$ and closed under series and parallel compositions.

They can be represented via a tree (called a cotree) using these operations, the leaves being the vertices.

An example



Cographs properties

Hereditary

Every induced subgraph of a cograph is a cograph.

Characterisation Theorem

A graph is a cograph iff it does not contain a P_4 (path of length 3 with 4 vertices) as an induced subgraph.

Easy induction

With the cotree representation.^a

a. Proof valid only in the finite case!

Cographs an interesting class of graphs

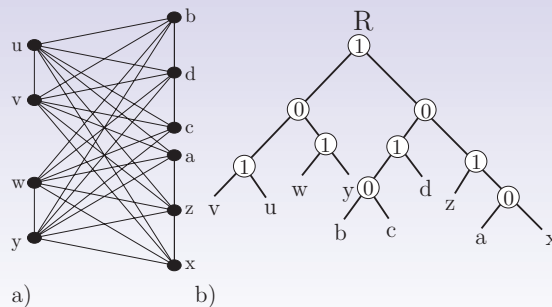


FIGURE: a) A cograph G . b) An embedding of the cotree T_G of G .

Properties of the cotree

- ▶ Vertices of the cotree can be labelled with 0 (parallel) and 1 (series).
- ▶ From G to \bar{G} just exchange 0's and 1's in the cotree. So one extra bit is enough to encode both of them.
- ▶ $xy \in E$ iff $LCA(x, y)$ in the cotree is labelled with 1
- ▶ The cotree provides an exact coding of the graph in $O(|V(G)|)$. And the query $xy \in E(G)?$ can be answered in $O(1)$ using LCA operations.

Twins

Twins

$x, y \in V$ are false- (resp. true-) **twins** if $N(x) = N(y)$ (resp. $N(x) \cup \{x\} = N(y) \cup \{y\}$).

x, y are false twins in G iff x, y are true twins in \bar{G} .

Elimination scheme

G is a cograph iff it exists an ordering of the vertices s.t. x_i has a twin (false or true) in $G\{x_{i+1}, \dots, x_n\}$

Cograph applications

- ▶ Fork, Join operations.
- ▶ Series-parallel electrical networks
- ▶ Series-parallel orders (applications to scheduling)
- ▶ Quantum physics : "Two-colorable graph states with maximal Schmidt measure" Simone Severini, University of York, U.K. 2005

Other applications

1. Redundancy elimination in graphs
2. Applications of quasi-twins :
 data compression in bipartite graphs,
 Identifying customers : if you change your phone card but keep the same set of correspondants (FBI ...)

Not so easy algorithmic questions

- ▶ How to recognize and certify in linear time, if a graph is a cograph?
 Yes case, build a cotree.
 No case, exhibit a P_4 .
- ▶ How to compute in linear time the classes of (false) twins?

Eventually the class of cograph has :

- ▶ A forbidden induced subgraph characterization
- ▶ A recursive definition and a tree structure
- ▶ An efficient encoding
- ▶ An elimination scheme

Using cotrees one can polynomially solve on cographs, NP-complete problems in the general case :

- ▶ Maximum clique
- ▶ Coloration
- ▶ If G is connected then $Diameter(G) \leq 2$
- ▶ Eigenvalues ...

Using the cotree in a bottom up way

- ▶ Max clique
Consider the cotree as an expression to evaluate with the following rules :
put a 1 on a leaf
interpret a 1 (resp. 0) node of the cotree as a + (resp. max)
- ▶ Min coloration
same rules
- ▶ Therefore Max clique = Min Coloration
 $\omega(G) = \chi(G)$ and cographs are **perfect graphs**

If the cotree is given, Max clique and Min coloration can be computed in $O(|V(G)|)$ for a cograph G .
Else we need to compute the cotree and the algorithm requires $O(|V(G)| + |E(G)|)$.

But they are not so simple (a cograph may have an exponential number of maximal cliques!).
This is why last year we had 2 internships introducing and studying extensions of cographs, namely "switch cographs" and "k-cographs".
Keeping the tree-structure but allowing new operations.

Research problem

1. Find efficient algorithms to compute quasi-twins and generalize to community detections in social networks, in a dynamic settings.
2. More difficult !
Find a polynomial algorithm for graph isomorphism

The great importance of the right model

It is crucial to use all the characteristics of the problem you want to solve and find a good model. Two examples from Peter Winkler's nice book on Mathematical Puzzles :

All the discrete structures considered here are supposed to be finite.

1. An odd number of soldiers in a field in such a way that all pairwise distances are different. Each soldier is told to keep an eye on the nearest other soldier.
Show that at least one soldier is not being watched.
2. A problem on rectangles :
A large rectangle of the plane is partitioned into smaller rectangles, each of which has either integer height or integer width (or both). Prove that the large rectangle also has this property.

Uses of geometric representations

- ▶ Computing a maximum clique (of maximum size) or an efficient representation of an interval graph.
- ▶ Numerous algorithms on planar graphs use the existence of a dual graph. (Ex : flows transform into paths)

Boxicity dimension

To each species one can associate the k principal parameters needed for its survival. Each parameter has a range of possible values. For human beings these parameters could be : temperature, air pollution, air quality (% of CO (carbon monoxide), ...).
To each species one can associate a box in the k -dimensional space. Therefore two species may have lived together iff their boxes intersect.

Which algorithms ?

- ▶ **Optimal ?**
- ▶ **Linear ?**
- ▶ **Efficient ?**
- ▶ **Simple ?**
- ▶ **Easy to program ?**

In our work on optimization problems on graphs, we will met several complexity barriers :

- ▶ Polynomial versus NP hard
In this case an algorithm running in $O(n^{75}.m^{252})$ **could be a nice result !**
- ▶ Linear versus Boolean matrix multiplication
Algorithms running in $O(n.m)$, **hard to find a lower bound !**
- ▶ Practical issues need linear algorithms in order to be applied
 - ▶ on huge graphs such as Web graph
 - ▶ many times such as inheritance in object oriented programming.
 - ▶ **Need for heuristics !**

Brute Force or naive algorithms

For some problem

1. Enumerate the set S of solutions
2. Select the optimal one
3. Show that $|S|$ is polynomial in the size of the problem

Variations

1. Find the best way to enumerate S (for example a Gray code in $O(|S|)$ constant time per element)
2. Many variations . . .
3. Sometimes the problem is NP-hard and the the game is to find the exponential algorithm with the lowest exponent.