4th Lecture: Modular decomposition
MPRI 2015–2016

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

Sophie Germain, 5 octobre 2015
Schedule

Introduction

Structural aspects of modular decomposition
  Uniqueness decomposition theorem
  Partitive Families
  Structural Aspects of Prime graphs
  Factoring permutation

Algorithmic aspects of modular decomposition
  Historical Notes
  Bottom up Techniques
  Top Down techniques
Pas de cours la semaine prochaine !
le cours du 12 octobre n’aura pas lieu
1. Examples of diameter searches based on the algorithms presented in this course:
   [http://gang.inria.fr/road/](http://gang.inria.fr/road/)

2. OpenStreetMap (OSM): 80 millions of nodes, average degree 3

3. Directed graphs (because of one way roads)

4. Roadmaps graphs a special domain of research interest
   Quasi-planar graph (bridges on the roads)

5. Never forget that computer science has an important experimental part.

6. Many algorithmic ideas come out some experiment.
Structural aspects of modular decomposition

Modules

For a graph $G = (V, E)$, a **module** is a subset of vertices $A \subseteq V$ such that

$$\forall x, y \in A, \quad N(x) - A = N(y) - A$$

The problem with this definition: must we check all subsets $A$?

Trivial Modules

$\emptyset$, $\{x\}$ and $V$ are modules.

Prime Graphs

A graph is **prime** if it admits only trivial modules.
Our main goal is to find good algorithms for modular decomposition.
But we cannot avoid to investigate in details the combinatorial properties of the modules in graphs.

Of course modules can be also defined for directed graphs but also for many discrete structures such as hypergraphs, matroids, boolean functions, submodular functions, automaton, ...
Examples

Characterization of Modules

A subset of vertices $M$ of a graph $G = (V, E)$ is a module iff

$$\forall x \in V \setminus M, \text{ either } M \subseteq N(x) \text{ or } M \cap N(x) = \emptyset$$

Examples of modules

- connected components of $G$
- connected components of $\overline{G}$
- any vertex subset of the complete graph (or the stable)
Modular decomposition (algorithmic aspects)

But also an operation on graphs: Modular composition

A graph grammar with a simple rule: replace a vertex by a graph

Very natural notion, (re)discovered under many names in various combinatorial structures such as: clan, homogeneous set, ...

An important tool in graph theory
Playing with the definition

**Duality**

A is a module of $G$ implies $A$ is a module of $\overline{G}$.

**Easy observations**

- No prime undirected graph with $\leq 3$ vertices (false for directed graphs, as a directed triangle shows it)
- $P_4$ the path with 4 vertices is the only prime on 4 vertices.
- $P_4$ is isomorphic to its complement.
Twins and strong modules

**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 $\overline{G}$.

Classes of twins are particular modules (stable sets for false twins and complete for true twins).

**Strong modules**

A **strong module** is a module that does not strictly overlap any other module.
**Modular partition**

A partition \( \mathcal{P} \) of the vertex set of a graph \( G = (V, E) \) is a **modular partition** of \( G \) if any part is a module of \( G \).

Let \( \mathcal{P} \) be a modular partition of a graph \( G = (V, E) \). The **quotient graph** \( G/\mathcal{P} \) is the induced subgraph obtained by choosing one vertex per part of \( \mathcal{P} \).
Lemma (Mohring Radermacher 1984)

Let $\mathcal{P}$ be a modular partition of $G = (V, E)$. $\mathcal{X} \subseteq \mathcal{P}$ is a module of $G/\mathcal{P}$ iff $\bigcup_{M \in \mathcal{X}} M$ is a module of $G$. 
Modular Decomposition Theorem

Theorem (Gallai 1967)

Let $G = (V, E)$ be a graph with $|V| \geq 4$, the three following cases are mutually exclusive:

1. $G$ is not connected,
2. $\overline{G}$ is not connected,
3. $G/\mathcal{M}(G)$ is a prime graph, with $\mathcal{M}(G)$ the modular partition containing the maximal strong modules of $G$. 
As a byproduct, we notice that a prime graph $G$ satisfies:
$G$ and $\overline{G}$ are connected
Modular decomposition tree

Tree
A recursive application of this theorem yields a tree $T$ in which:

- The root corresponds to $V$
- Leaves are associated to vertices
- Each node corresponds to a strong module

There are 3 types of nodes:
- Parallel
- Series
- Prime
Another explanation

The set of strong modules is nested into an inclusion tree (called the **modular decomposition tree** $MD(G)$ of $G$).
Partitive Families

Lemma

If $M$ and $M'$ are two overlapping modules then

- (i) $M \setminus M'$ is a module
- (ii) $M \cap M'$ is a module
- (iii) $M \cup M'$ is a module
- (iv) $M \Delta M'$ is a module

- A family satisfying (i) - (iv) is called a partitive family
- A family satisfying (i) - (iii)) is called a weak partitive family
Remarks on the module properties

- (ii) is always true, even if $M$ and $M'$ are not overlapping
- (iii) could be written that way:
  If $M \cap M' \neq \emptyset$ then $M \cup M'$ is a module
- (iv) is not a consequence of (i) and (iii) since $M \setminus M'$ and $M' \setminus M$ do not overlap. And overlapping is really needed to prove (i) for modules.
A proof of Gallai’s theorem

We note that the two first cases are exclusive. Either $G$ is not connected or $\overline{G}$ is not connected but not both. Let us consider the third case in which $G$ and $\overline{G}$ are connected.

1. If $G$ is prime, then the third case is trivially obtained.
2. Else $G$ admits some non trivial modules. Let us consider $M_1, \ldots, M_k$ the maximal non trivial modules of $G$. We will prove that they are strong modules.
if $M_i$ and $M_j$ overlap then using the algebraic properties of modules: necessarily $M_i \cup M_j$ is a module.

But using their maximality: $M_i \cup M_j = V(G)$.

Then $(M_i \setminus M_j, M_i \cap M_j, M_j \setminus M_i)$ is a modular partition of $G$. 
Since $G$ is connected, there is at least one edge between these 3 sets. But then using the definition of modules all edges between these 3 sets exist, and therefore $\overline{G}$ is not connected, a contradiction. So the $M'_i$s do not overlap.

To finish the proof, it suffices to notice that the quotient graph is prime, else at least one of the $M_i$ would not be maximal.

At least the $M'_i$s are strong modules since they partition $V(G)$. 
One can notice, that we use very little of graph theory in this proof.
A general combinatorial decomposition

Fact

The set of all modules of an undirected graph (resp. a directed graph) constitutes a partitive family (resp. a weak partitive family).

Uniqueness decomposition theorem, Chein Habib Maurer 1981

Partitive (resp. weakly partitive) families admit a decomposition tree with two (resp. three) types of nodes:

- degenerate (also called fragile)
- prime
- (resp. linear)
This tree representation theorem for partitive (resp. weakly partitive) families $F \subseteq 2^{|X|}$, yields an encoding of these families in $O(|X|)$.

This kind of combinatorial decomposition of a set family has been generalized in many directions.
Application: Modular Decomposition Theorem for Directed Graphs

Theorem (Chein, Habib, Maurer 1981)

Let $G = (V, E)$ be a directed graph with $|V| \geq 4$, the four following cases are mutually exclusive:

1. $G$ is not connected, *Parallel node*
2. $\overline{G}$ is not connected, *Series node*
3. $G^*$ is not strongly connected, *Linear node*
4. $G/\mathcal{M}(G)$ is a prime graph, with $\mathcal{M}(G)$ the modular partition containing the maximal strong modules of $G$, *Prime node*
Prime graphs are nested

**Folklore Theorem**

Let $G$ be a prime graph ($|G| \geq 4$), then $G$ contains a $P_4$.

**Theorem Schmerl, Trotter, Ille 1991**

Let $G$ be a prime graph ($|G| = n \geq 4$), then $G$ contains a prime graph on $n - 1$ vertices or a prime graph on $n - 2$ vertices.
A simple proof

A stronger statement, Cournier, Ille, 1991

For a prime graph there is at most one vertex not contained in a $P_4$.

Proof

As a prime graph $G$ is necessarily connected and if $\exists x \in V$ that does not belong to a $P_4$. Every connected component of $\overline{N(x)}$ is a module, therefore $\overline{N(x)}$ must be a stable set.
If $\exists x \neq y \in V$ that does not belong to a $P_4$,
  wlog assume $xy \notin E$
But then $y \in \overline{N(x)}$ and therefore : $N(y) \subseteq N(x)$.
By symmetry $x, y$ must be false twins, a contradiction.
Nota Bene: this result also holds for infinite graphs! So does our previous proof.
In fact if there is such a vertex $x \in V$, $x$ is adjacent to the middle vertices of a $P_4$. (Such a subgraph is called a bull).

A bull is isomorphic to its complement.
Factoring permutation

Definition

A **factoring permutation** of a graph $G = (V, E)$ is a permutation of $V$ in which any strong module of $G$ is a factor. [CH 97]
Structural aspects of modular decomposition

Factoring permutation
Structural aspects of modular decomposition

Factoring permutation
4th Lecture: Modular decomposition MPRI 2015–2016

- Structural aspects of modular decomposition
- Factoring permutation

![Graph Diagram]

1 2 3 4 5 6 7 8 9 10 11
Algorithms

- From $G$ to factoring permutation: $O(n + m \log n)$ [HPV99]
- From factoring permutation to $MD(G)$: $O(n + m)$ [CdMH01] [UY00] [BXHP05]
The set of strong modules is nested into an inclusion tree (called the modular decomposition tree $MD(G)$ of $G$).

A factoring permutation is simply a left-right ordering of the leaves of a plane drawing of $MD(G)$.
Consequence: it always exists factoring permutations. There are easier to compute than the modular decomposition tree.
Historical notes

The big list of published algorithms for modular decomposition
(N.B. Perhaps some items are missing . . . please give me the missing references)

- Cowan, James, Stanton 1972 $O(n^4)$
- Maurer 1977 $O(n^4)$ directed graphs
- Blass 1978 $O(n^3)$
- Habib, Maurer 1979 $O(n^3)$
- Habib 1981 $O(n^3)$ directed graphs
- Corneil, Perl, Stewart 1981, $O(n + m)$ cograph recognition.
- Cunningham 1982 $O(n^3)$ directed graphs
- Buer, Mohring 1983 $O(n^3)$
- McConnell 1987 $O(n^3)$
- McConnell, Spinrad 1989 $O(n^2)$ incremental
- Adhar, Peng 1990 $O(\log^2 n), O(nm)$ proc. parallel, cographs, CRCW-PRAM
- Lin, Olariu 1991 $O(\log n), O(nm)$ proc. parallel, cographs, EREW-PRAM
- Spinrad 1992 $O(n + m\alpha(m, n))$
- Cournier, Habib 1993 $O(n + m\alpha(m, n))$
- Ehrenfeucht, Gabow, McConnell, Spinrad 1994 $O(n^3)$ 2-structures
- Ehrenfeucht, Harju, Rozenberg 1994 $O(n^2)$ 2-structures, incremental
- McConnell, Spinrad 1994 $O(n + m)$
- Cournier, Habib 1994 $O(n + m)$
- Bonizzoni, Della Vedova 1995 $O(n^{3k-5})$ Committee decomposition for hypergraphs
- Dahlhaus 1995 $O(\log^2 n), O(n + m)$ proc. parallel, cographs, CRCW-PRAM
- Dahlhaus 1995 $O(\log^2 n), O(n + m)$ proc. parallel, CRCW-PRAM
Habib, Huchard, Sprinrad 1995 $O(n + m)$ inheritance graphs
McConnell 1995 $O(n^2)$ 2-structures, incremental
Capelle, Habib 1997 $O(n + m)$ if a factoring permutation is given
Dahlhaus, Gustedt, McConnell 1997 $O(n + m)$
Dahlhaus, Gustedt, McConnell 1999 $O(n + m)$ directed graphs
Habib, Paul, Viennot 1999 $O(n + m \log n)$ via a factoring permutation
McConnell, Spinrad 2000 $O(n + m \log n)$
Habib, Paul 2001 $O(n + m)$ cographs via a factoring permutation
Capelle, Habib, Montgolfier 2002 $O(n + m)$ directed graphs if a factoring permutation is provided.
Shamir, Sharan 2003 $O(n + m)$ cographs, fully-dynamic
McConnell, Montgolfier 2003 $O(n + m)$ directed graphs
Habib, Montgolfier, Paul 2003 $O(n + m)$ computes a factoring permutation
Simpler Linear-Time Modular Decomposition via Recursive Factorizing Permutation

Tedder, Corneil, Habib, Paul, ICALP (1) 2008 : 634-646.
Why it is so important?

[Jerry Spinrad’ book 03]

The new [linear time] algorithm [MS99] is currently too complex to describe easily [...] The first $O(n^2)$ partitioning algorithms were similarly complex; I hope and believe that in a number of years the linear algorithm can be simplified as well.
Applications of modular decomposition

- A very natural operation to define on discrete structures, searching regularities.
- A structure theory for comparability graphs
- A compact encoding using module contraction and if we keep at each prime node the structure of the prime graph.
- Divide and conquer paradigm can be applied to solve optimization problems. For example to test isomorphism.
A very basic graph algorithmic problem (similar to graph isomorphism problem).

A better understanding of graph algorithms and their data structures.
Minimal Modules

Minimal module containing a set

For every $A \subseteq V$ there exists a unique minimal module containing $A$

Proof:
Since the module family is partitive and therefore closed under $\cap$. 

Splitters

**Definition**

A splitter for a \( A \subseteq V \), is a vertex \( z \notin A \) s.t. \( \exists x, y \in A \text{ with } zx \in E \text{ and } zy \notin E \).

**Modules**

\( A \subseteq V \) is a module iff \( A \) does not have any splitter.

**Useful lemma**

If \( z \) is a splitter for a \( A \subseteq V \), then any module containing \( A \) must also contain \( z \).
Submodularity

Let us denote by $s(A)$ the number of splitters of a set $A$, then $s$ is a submodular function.

**Definition**

A function is submodular if

\[
\forall A, B \subseteq E \\
 f(A \cup B) + f(A \cap B) \leq f(A) + f(B)
\]

This is the basic idea of Uno and Yagura’s algorithm for the modular decomposition of permutation graphs in $O(n)$.
Bottom-up Techniques

Sketch of the algorithm
For each pair of vertices $x, y \in V$
Compute the minimal module $m(x, y)$ containing $x$ and $y$.

Closure with splitters
While there exists a splitter add it to the set.

Complexity
$O(n^2(n + m))$
One can derive a primality test since if there exists a non trivial module, it contains at least two vertices.
For some problems Bottom-Up techniques are the best known.
Origins: Golumbic, Kaplan, Shamir 1995

**Input**: $G_1 = (V, E_1)$ and $G_2 = (V, E_2)$ 2 undirected graphs such that $E_1 \subseteq E_2$ and $\Pi$ be a graph property.

**Results**: a sandwich graph $G_s = (V, E_s)$ satisfying property $\Pi$ and such that $E_1 \subseteq E_s \subseteq E_2$.

Edges of $E_1$ are forced, those of $E_2$ are optional ones, but those of $E_3 = \overline{E_2}$ are forbidden.

Unfortunately most cases are NP-complete, as for example of $\Pi$

- $G_s$ being comparability, chordal, strongly chordal, . . .
Only few polynomial cases

- cographs Golumbic, Kaplan, Shamir (1995)
- sandwich module Cerioli, Everett, de Figueiredo, Klein (1998)

Natural question
Find efficient algorithms for these polynomial cases.
Sandwich module problem

**Input**: $G_1 = (V, E_1)$ and $G_2 = (V, E_2)$ 2 undirected graphs such that $E_1 \subseteq E_2$.

**Result**: a sandwich graph $G_s = (V, E_s)$ having a non trivial module and such that $E_1 \subseteq E_s \subseteq E_2$. 
Minimal Sandwich Module

**Splitter**

For a subset $A \subseteq V$, a splitter is a vertex $z \notin A$ s.t. $\exists x, y \in A$ with $zx \in E_1$ and $zy \notin E_2$ (or equivalently $zy \in E_3$)

A splitter is also called **bias vertex**.

**Algorithm**

The computation of a minimal sandwich module can be done in $O(n^2(n + m_1 + m_3))$.

Hard to do better with this idea, using a bottom up approach.
Brute Force Algorithm

Using the decomposition theorem, we only have to compute at most $n$ times some connected components of $G$ or its complement. $O(n.(n + m))$ complexity.
Three explored directions

- Ehrenfeucht et al approach
- Using Factoring Permutation
- Using LexBFS (as for cographs) Next lecture.
Ehrenfeucht et al approach

\[ \mathcal{M}(G, v) \]

\[ \mathcal{M}(G, v) \] is the partition of \( V(G) \) composed by \( \{v\} \) and the maximal modules of \( G \) that do not contain \( v \).

Principle of the Ehrenfeucht et al.’s algorithm

1. Compute \( \mathcal{M}(G, v) \)
2. Compute \( MD(G/\mathcal{M}(G,v)) \)
3. For each part \( \mathcal{X} \in \mathcal{M}(G, v) \) compute \( MD(G[\mathcal{X}]) \)
Computing $\mathcal{M}(G, v)$ via Partition Refinement

Splitter again

If $z$ is a splitter of $A \subseteq V(G)$ then any strong module contained in $A$ is either contained in $N(z) \cap A$ or in $A - N(z)$.
Computation of $\mathcal{M}(G, v)$

$\Rightarrow O(n + m \log n)$ time using vertex partitioning algorithm.
1. Particular partition refinement rule:
   Do not refine its part
   Just to maintain the invariant:
   Modular partition \( \leq \) Current partition

2. To obtain a \( \log n \)
   Avoid the biggest part
How to reconstruct the modular decomposition tree from the partition $\mathcal{M}(G, v)$?
The most difficult step in many algorithms.
Algorithmic aspects of modular decomposition

Top Down techniques

**Computation of** $\text{MD}(G/\mathcal{M}(G,v))$

- The modules of $G/\mathcal{M}(G,v)$ are linearly nested: any non-trivial module contains $v$
- The *forcing graph* $\mathcal{F}(G, v)$ has edge $\overrightarrow{xy}$ iff $y$ separates $x$ and $v$
The strong connected components of the forcing graph $\mathcal{F}(G, v)$ provides the modules of $G/M(G, v)$.

Recurse on each module.
Algorithmic aspects of modular decomposition

Top Down techniques

Complexity

- [Ehrenfeucht et al.'94] gives a $O(n^2)$ complexity. It is quite tricky to efficiently compute the forcing graph $F(G, v)$.
- [MS00] gives a very simple $O(n + m \log n)$ algorithm based on vertex partitioning.
- [DGM'01] proposes a $O(n + m.\alpha(n, m))$ and a more complicated $O(n + m)$ implementation.

Other algorithms

- [CH94] and [MS94] present the first linear algorithms.
- [MS99] present a new linear time algorithm which extends to transitive orientation.
Factoring permutations

The set of strong modules is nested into an inclusion tree (called the modular decomposition tree $MD(G)$ of $G$).

A factoring permutation is simply a left-right ordering of the leaves of a plane drawing of $MD(G)$. 
Consequence: it always exists factoring permutations. There are easier to compute than the modular decomposition tree.
Invariant

Any strong module is a factor of the partition.
Splitter interpretation

Starting with the partition \( \{ \mathcal{N}(x), \{x\}, \overline{\mathcal{N}(x)} \} \), we maintain the following invariant:
There exists a factoring permutation smaller than the current partition.
A hierarchy of graph models

1. Undirected graphs (graphes non orientés)
2. Tournaments (Tournois), sometimes 2-circuits are allowed.
3. Signed graphs (Graphes signés) each edge is labelled + or - (for example friend or enemy)
4. Oriented graphs (Graphes orientés), each edge is given a unique direction (no 2-circuits)
   An interesting subclass are the DAG Directed Acyclic Graphs (graphes sans circuit), for which the transitive closure is a partial order (ordre partiel)
5. Directed graphs or digraphs (Graphes dirigés)
Second Neighborhoods Conjecture
P.D. Seymour 1990
Every digraph without 2-circuits has a vertex with at least as many second neighbors as first neighbors.
Second neighbors, $SN(x)$ is the set of vertices at exact distance 2 of $x$.
Therefore we are looking for $x$ such that $|SN(x)| \geq |N(x)|$. 