Solving the Maximum Agreement Subtree and Maximum Compatible Tree problems on bounded degree trees

Sylvain Guillemot, François Nicolas

LIRMM, Montpellier
France

4th July 2006

Introduction

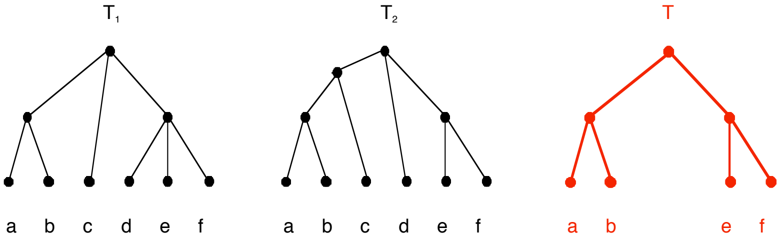
The Mast and Mct problems:
given a set of "evolutionary trees", choose a subset of the "species" on which they agree (for a suitable definition of this notion).

Applications:

1. obtain a consensus between several trees;
2. define similarity measures between trees;
3. other applications.

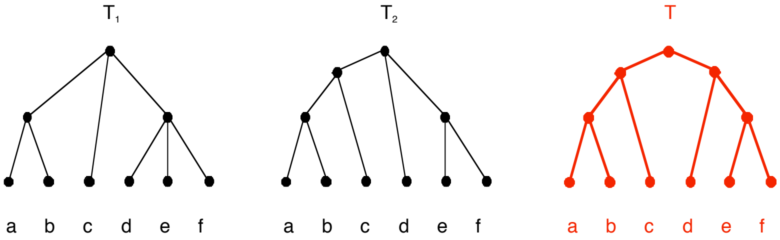
The MAST problem

The Mast problem: [FG85]
Given a collection $\mathcal{T} = \{T_1, ..., T_k\}$ of trees on a common label set L, find a tree T with the largest number of leaves, st T is "included" in every T_i.
$\rightarrow$ we say that T is an *agreement subtree* for $\mathcal{T}$.



The MCT problem

The Mct problem: [HS96]
Given a collection $\mathcal{T} = \{T_1, ..., T_k\}$ of trees on a common label set L, find a tree T with the largest number of leaves, st T can be obtained from every T_i by removing labels and resolving multifurcations.
$\rightarrow$ we say that T is a *compatible tree* for $\mathcal{T}$.



Bounded degree trees

We have the following parameters:

- k is the number of input trees;
- n is the number of labels;
- $N = O(kn)$ is the size of the instance;
- D is a bound on the maximum degree of the trees.

Here we consider the parameterized complexity of the problems wrt D.

Important result: MAST can be solved in time $O(n^D + kn^3)$ ([FPT95, Bry97]).

Questions and results

Questions:

- MAST can be solved in $O(n^D + kn^3)$ time; but...
- can we do better, eg $2^D N$ or $N^{\sqrt{D}}$?
- what is the complexity of MCT?

Results:

- MAST is $\mathrm{W}[1]$-hard wrt D, and cannot be solved in time $\Phi(D)N^{o(D)}$ unless SNP $\subseteq$ SE;
- MCT is $\mathrm{W}[1]$-hard wrt D, and cannot be solved in time $\Phi(D)N^{o(2^{D/2})}$ unless SNP $\subseteq$ SE.

Toolkit

The class Fpt : A parameterized problem Π is in $\textsc{Fpt}$ if on every instance (x, k), it can be solved in time $f(k)|x|^c$ for some function f and some constant c.
For instance, $\textsc{Mast}$ would be $\textsc{Fpt}$ wrt D if we could solve it in time $2^D N$.
Goal of parameterized complexity:

- "positive toolkit": techniques to derive FPT algorithms when possible;
- "negative toolkit": techniques to show that a problem is unlikely to be FPT.

Toolkit

Definition (fpt-reduction).
Let Π, Π' be two parameterized problems.
We say that Π fpt-reduces to Π' iff there are functions f, g, a constant c, and an algorithm A which from every instance (x, k) of Π, produces an instance (x', k') of Π' satisfying:

- A has running time $f(k)|x|^c$;
- $k' \leq g(k)$;
- $(x, k) \in \Pi$ iff $(x', k') \in \Pi'$.

An fpt-reduction is *linear* if g is at most linearly increasing (ie if $k' = O(k)$).

Toolkit

The class W[1].

Contains the problems fpt-reducible to *IS*: given a parameter k, and a graph G, decide if G has an independent of size k?
if Π is $\mathrm{W}[1]$-hard, then Π cannot be solved in $f(k)|x|^c$ time unless $\mathrm{FPT} = \mathrm{W}[1]$.

The class W$_l$[1].

Contains the problem which are reducible to *IS* by linear fpt-reductions.
If Π is $\mathrm{W}_l[1]$-hard, then Π cannot be solved in $f(k)|x|^{o(k)}$ time unless $\mathrm{SNP} \subseteq \mathrm{SE}$ [Hua04].

Reduction for MAST

Main result:

Proposition. $\textsc{Mast}[D]$ is $\mathrm{W}_l[1]$-hard.

Consequences:

- $\textsc{Mast}$ cannot be solved in time $\Phi(D)N^c$ unless $\textsc{Fpt} = \mathrm{W}[1]$;
- $\textsc{Mast}$ cannot be solved in time $\Phi(D)N^{o(D)}$ unless $\textsc{Snp} \subseteq \textsc{Se}$.

Reduction for MAST

Let $G = (V, E)$ be a k-partite graph with partition $V_1, ..., V_k$.

A *partitioned independent set* of G is a set I st:

- I is an independent set of G;
- $|I \cap V_i| = 1$ for every i.

The problem PIS asks: given a k-partite graph without isolated vertices, decide if G has a partitioned independent set.

Lemma. PIS[k] is $\mathrm{W}_l[1]$-hard.

Reduction for Mast

Lemma. There exists a linear fpt-reduction from Pis$[k]$ to Mast$[D]$.

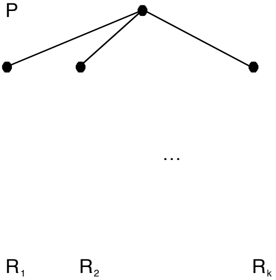
Sketch of the reduction.

Let $G = (V, E)$ be a k-partite graph with partition $V_1, ..., V_k$.
We contruct a collection $\mathcal{T}$ of trees with degree $\leq k + 1$.
$\longrightarrow$ this ensures the linearity.

We set $L := V, \mathcal{T} = \{P\} \cup \{Q_e : e \in E\}$.

Control component. P

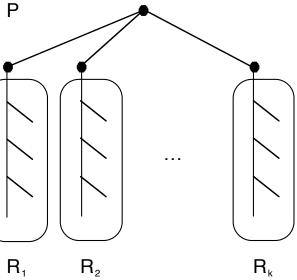
- the root has $R_1, \ldots, R_k$ as child subtrees;



Reduction for MAST

Control component. P

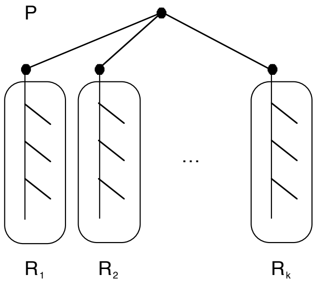
- the root has $R_1, \ldots, R_k$ as child subtrees;
- each R_i is a rake-tree with label set V_i.



Reduction for MAST

Control component. *P*

- the root has $R_1, ..., R_k$ as child subtrees;
- each R_i is a rake-tree with label set V_i.

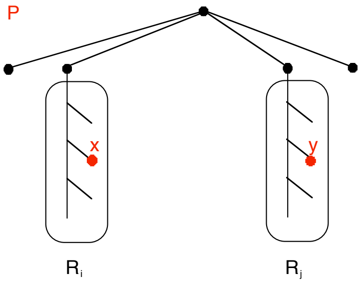


$\Rightarrow \Delta(P) = k.$

Reduction for MAST

Selection component. Q_e for each $e \in E$

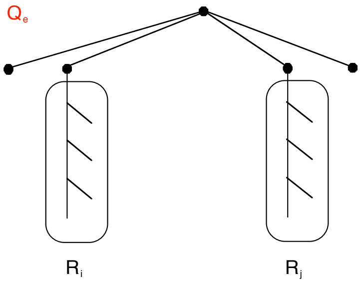
- let x, y be the two endpoints of e; then $x \in V_i, y \in V_j$.
- Q_e is obtained from P, by



Reduction for MAST

Selection component. Q_e for each $e \in E$

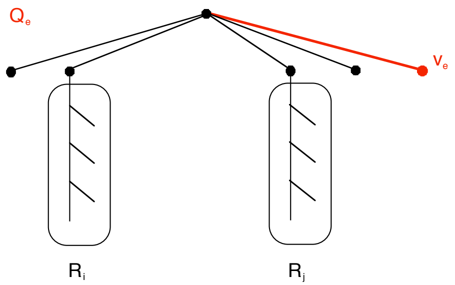
- let x, y be the two endpoints of e; then $x \in V_i, y \in V_j$;
- Q_e is obtained from P, by (i) removing x, y,



Reduction for MAST

Selection component. Q_e for each $e \in E$

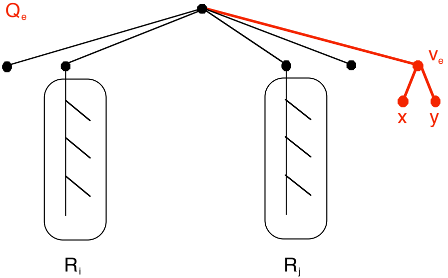
- let x, y be the two endpoints of e; then $x \in V_i, y \in V_j$;
- Q_e is obtained from P, by (i) removing x, y, (ii) creating a new node v_e as child of the root



Reduction for MAST

Selection component. Q_e for each $e \in E$

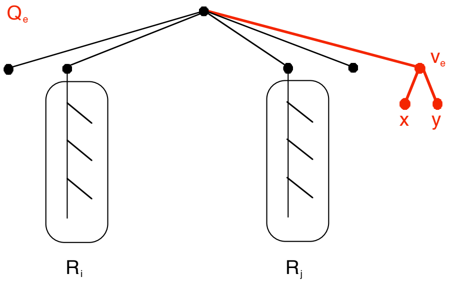
- let x, y be the two endpoints of e; then $x \in V_i, y \in V_j$;
- Q_e is obtained from P, by (i) removing x, y, (ii) creating a new node v_e as child of the root, (iii) regrafting x, y as children of v_e.



Reduction for MAST

Selection component. Q_e for each $e \in E$

- let x, y be the two endpoints of e; then $x \in V_i, y \in V_j$;
- Q_e is obtained from P, by (i) removing x, y, (ii) creating a new node v_e as child subtree of the root, (iii) regrafting x, y as children of v_e.



$$\Rightarrow \Delta(Q_e) = k + 1$$

Reduction for MAST

Lemma. G is a positive instance of PIS iff $\text{MAST}(\mathcal{T}) \geq k$

1. if G has a partitioned independent set $I = \{x_1, ..., x_k\}$ with $x_i \in V_i$, then $T = (x_1, ..., x_k)$ is an agreement subtree for $\mathcal{T}$;

2. if $\mathcal{T}$ has an agreement subtree T of size k, then:
   - Control: T has the form $(x_1, ..., x_k)$ with $x_i \in V_i$;
   - Selection: there is no edge joining x_i, x_j in G.

and thus $I = \{x_1, ..., x_k\}$ is a partitioned independent set of G.

Reduction for $\textsc{Mct}$

Main result:

Proposition. $\textsc{Mct}[2^{\lceil D/2 \rceil}]$ is $\mathrm{W}_l[1]$-hard.

Consequences:

- $\textsc{Mct}$ cannot be solved in $\Phi(D)N^c$ time unless $\textsc{Fpt} = \mathrm{W}[1]$;
- $\textsc{Mct}$ cannot be solved in $\Phi(D)N^{o(2^{D/2})}$ time unless $\textsc{Snp} \subseteq \textsc{Se}$.

Reduction for MCT

Let $G = (V, E)$ be a k-partite graph with partition $V_1, ..., V_k$.

A *2-partitioned independent set* of G is a set I st:

- I is an independent set of G;
- $|I \cap V_i| = 2$ for every i.

The problem PIS-2 asks: given a k-partite graph G, does G have a 2-partitioned independent set?

Lemma. The problem PIS-2[k] is $W_l[1]$-hard.

Reduction for MCT

Lemma. There exists a linear fpt-reduction from PIS-$2[k]$ to MCT$[2^{\lceil D/2 \rceil}]$.

Sketch of the reduction.

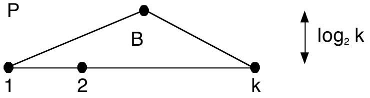
Let $G = (V, E)$ be a k-partite graph with partition $V_1, \ldots, V_k$.
We construct a collection $\mathcal{T}$ of trees with degree $\leq 2\lceil \log_2 k \rceil + 1$.
$\rightarrow$ this ensures the linearity since with $D \leq 2\lceil \log_2 k \rceil$, we have $2^{\lceil D/2 \rceil} = O(k)$.

We set $L := V$, $\mathcal{T} = \{P, P'\} \cup \{Q_e : e \in E\}$.

Reduction for MCT

Control component: P

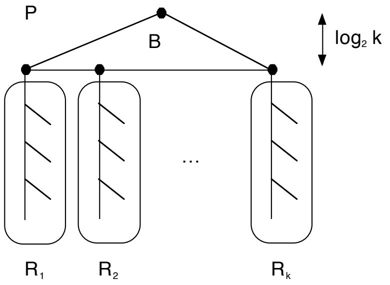
- B binary tree, with leaves $1, ..., k$ and height $\leq \lceil \log_2 k \rceil$;



Reduction for MCT

Control component: P

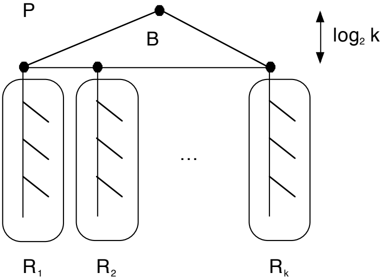
- B binary tree, with leaves $1, \ldots, k$ and height $\leq \lceil \log_2 k \rceil$;
- on leaf i of B, we graft a rake-tree R_i with label set V_i, ie $P = B[R_1, \ldots, R_k]$.



Reduction for MCT

Control component: P

- B binary tree, with leaves $1, ..., k$ and height $\leq \lceil \log_2 k \rceil$;
- on leaf i of B, we graft a rake-tree R_i with label set V_i, ie $P = B[R_1, ..., R_k]$.

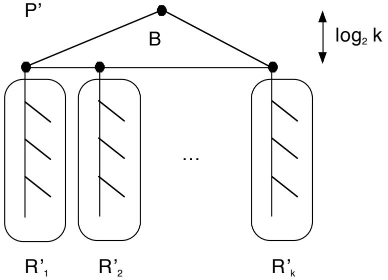


$\Rightarrow \Delta(P) = 2.$

Reduction for MCT

Control component: P'

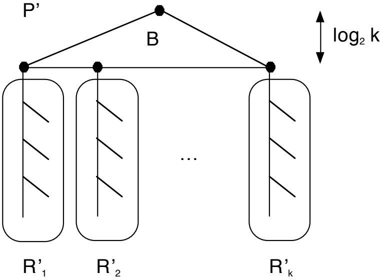
- let R'_i be the symmetric rake-tree of R_i;
- on leaf i of B, we graft the rake-tree R'_i, ie $P' = B[R'_1, ..., R'_k]$.



Reduction for MCT

Control component: P'

- let R'_i be the symmetric rake-tree of R_i;
- on leaf i of B, we graft the rake-tree R'_i, ie $P' = B[R'_1, ..., R'_k]$.

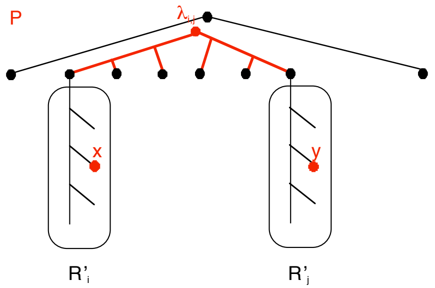


$\Rightarrow \Delta(P') = 2.$

Reduction for MCT

Selection component: Q_e for each $e \in E$

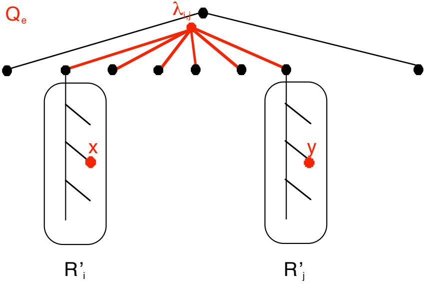
- let x, y be the two endpoints of e; then $x \in V_i, y \in V_j$. Let $\lambda_{i,j}$ the least common ancestor of leaves i, j in B.
- Q_e is obtained from P by



Reduction for MCT

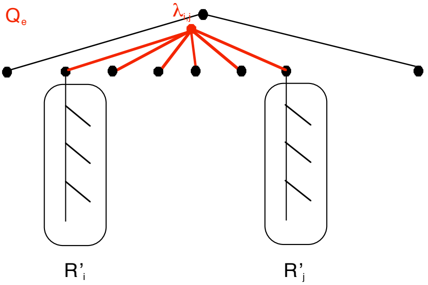
Selection component: Q_e for each $e \in E$

- let x, y be the two endpoints of e; then $x \in V_i, y \in V_j$. Let $\lambda_{i,j}$ the least common ancestor of leaves i, j in B.
- Q_e is obtained from P by (i) collapsing the edges joining i, j in P,



Selection component: Q_e for each $e \in E$

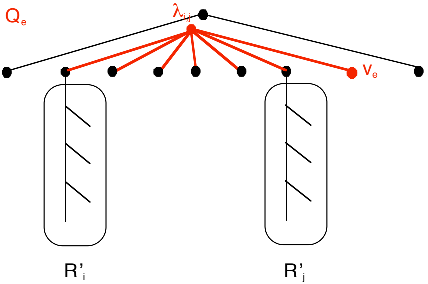
- let x, y be the two endpoints of e; then $x \in V_i, y \in V_j$. Let $\lambda_{i,j}$ the least common ancestor of leaves i, j in B.
- Q_e is obtained from P by (i) collapsing the edges joining i, j in P, (ii) removing x, y,



Reduction for MCT

Selection component: Q_e for each $e \in E$

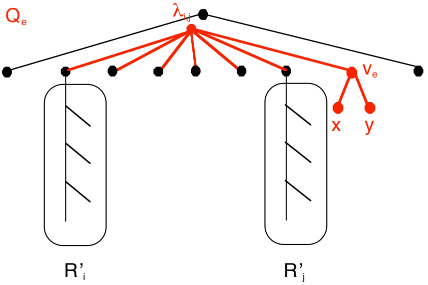
- let x, y be the two endpoints of e; then $x \in V_i, y \in V_j$. Let $\lambda_{i,j}$ the least common ancestor of leaves i, j in B.
- Q_e is obtained from P by (i) collapsing the edges joining i, j in P, (ii) removing x, y, (iii) creating a new node v_e as child of $\lambda_{i,j}$



Reduction for MCT

Selection component: Q_e for each $e \in E$

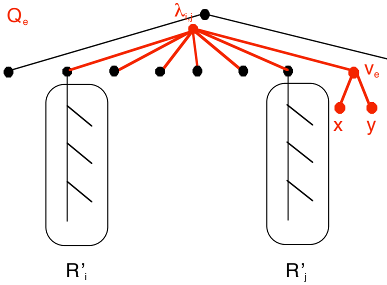
- let x, y be the two endpoints of e; then $x \in V_i, y \in V_j$. Let $\lambda_{i,j}$ the least common ancestor of leaves i, j in B.
- Q_e is obtained from P by (i) collapsing the edges joining i, j in P, (ii) removing x, y, (iii) creating a new node v_e as child of $\lambda_{i,j}$, (iv) regrafting x, y as children of v_e.



Reduction for MCT

Selection component: Q_e for each $e \in E$

- let x, y be the two endpoints of e; then $x \in V_i, y \in V_j$. Let $\lambda_{i,j}$ the least common ancestor of leaves i, j in B.
- Q_e is obtained from P by (i) collapsing the edges joining i, j in P, (ii) removing x, y, (iii) creating a new node v_e as child of $\lambda_{i,j}$, (iv) regrafting x, y as children of v_e.



$\Rightarrow \Delta(Q_e) \leq 2\lceil \log_2 k \rceil + 1.$

Reduction for MCT

Lemma. G positive instance of PIS-2 iff $\text{MCT}(\mathcal{T}) \geq 2k$

1. if G has a 2-partitioned independent set $I = \{x_1^1, x_1^2, ..., x_k^1, x_k^2\}$ where $x_i^1, x_i^2 \in V_i$, then $T = B[(x_1^1, x_1^2), ...(x_k^1, x_k^2)]$ is a compatible tree for $\mathcal{T}$;

2. if $\mathcal{T}$ has a compatible tree T of size $2k$, then:
   - Control: T has the form $B[(x_1^1, x_1^2), ..., (x_k^1, x_k^2)]$ with $x_i^1, x_i^2 \in V_i$;
   - Selection: there is no edge joining vertices x_i^p, x_j^q in G.

and thus $I = \{x_1^1, x_1^2, ..., x_k^1, x_k^2\}$ is a 2-partitioned independent set of G.

Other results

Other results on the parameterized complexity of MAST and MCT [GN06]:

- an algorithm solving MCT in time $O(n^{2^{d+2}} + kn^3)$;
- algorithms solving MAST and MCT in time $2^{2kd}n^3$;
- more results.

D. Bryant.
Building trees, hunting for trees and comparing trees: theory and method in phylogenetic analysis.
PhD thesis, University of Canterbury, Department of Mathemathics, 1997.

C. R. Finden and A. D. Gordon.
Obtaining common pruned trees.
Journal of Classification, 2:255–276, 1985.

M. Farach, T. M. Przytycka, and M. Thorup.
On the agreement of many trees.
Information Processing Letters, 55(6):297–301, 1995.

S. Guillemot and F. Nicolas.
Parameterized complexity of the maximum agreement subtree and maximum compatible tree problems, to appear.
2006.

A. M. Hamel and M. A. Steel.
Finding a maximum compatible tree is NP-hard for sequences and trees.
Applied Mathematics Letters, 9(2):55–59, 1996.

X. Huang.
Parameterized complexity and polynomial-time approximation schemes.
PhD thesis, Texas AM University, Department of Computer Science, 2004.