Complexity Insights of the Minimum Duplication Problem

Guillaume Blin Paola Bonizzoni Riccardo Dondi
Romeo Rizzi Florian Sikora

Université Paris-Est Marne-la-Vallée, LIGM - UMR CNRS 8049, France
DISCo, Università degli Studi di Milano-Bicocca, - Milano, Italy
DSLSCC, Università degli Studi di Bergamo, - Bergamo, Italy
DIMI, Università di Udine - Udine, Italy
Lehrstuhl fur Bioinformatik, Friedrich-Schiller-Universitat Jena, Germany

January 2012
Minimum Duplication Problem

- Problem in phylogenetics and comparative genomics related to 2 types of trees: gene trees and species trees
- Evolutionary history of genomes
  - results from a series of evolutionary events producing new species from a common ancestor (speciation)
  - represented as a species tree

Guillaume Blin Complexity Insights of the Minimum Duplication Problem
Minimum Duplication Problem

- Other evolutionary events such as gene duplication, loss, lateral transfer leading to new species
- Focus on duplication: genomic event causing a gene inside a genome to be copied; each copy evolving independently
- Considering a specific gene family, its evolution with regards to extant species is given as a gene tree
Trees reconciliation

- Gene and species trees may present incompatibilities
- A challenging problem is to reconcile them by hypothetical gene duplication
Trees reconciliation

- Gene and species trees may present incompatibilities
- A challenging problem is to reconcile them by hypothetical gene duplication
Trees reconciliation

- Gene and species trees may present incompatibilities
- A challenging problem is to reconcile them by hypothetical gene duplication
Trees reconciliation

- Parsimony principle in finding minimum number of gene duplications
- Inferred by lower common ancestor mapping
Minimum Duplication Problem

Definition

**Input**  a set of gene trees

**Output**  a species tree that induces a minimum number of gene duplications

Known Hardness Results

- Relation with Minimum Triplets Consistency: NP-hard, W[2]-hard,
- inapproximable within factor $O(\log n)$ even for a forest of unbounded number of uniquely leaf-labeled gene trees with three leaves
- We will prove that it is APX-hard even when consisting of 5 uniquely leaf-labelled gene trees with unbounded number of leaves (technical proof not presented here)
Minimum Duplication Problem

Definition

**Input** a set of gene trees

**Output** a species tree that induces a minimum number of gene duplications

Known Results On The Bright Side

- Different heuristics have been proposed
- Among them, Chauve et al proposed to consider a related problem which recursively produces a natural greedy heuristic: \textsc{Minimum Bipartite Duplication Problem}
Minimum Bipartite Duplication Problem

Definition

Input  a set of gene trees
Output a bipartition \((\Lambda_1, \Lambda_2)\) of the species inducing a minimum number of gene duplications

It corresponds to find duplications preceding the first speciation (pre-duplications)
Minimum Bipartite Duplication Problem

Definition

**Input** a set of gene trees

**Output** a bipartition $(\Lambda_1, \Lambda_2)$ of the species inducing a minimum number of gene duplications

It corresponds to find duplications preceding the first speciation (pre-duplications)

Known Results On The Bright Side

- 2-approximable
- ⇒ We show that the problem is Randomized Polynomial for an unbounded number of bounded depth gene trees
Randomized Algorithm

- Definition: Algorithm allowed to do some random decisions as it processes the input.
- We will prove that our algorithm has a polynomial overall running time to get a high probability of success.
- Based on the following correspondence: $\text{MBD} \equiv \text{Min Cut in Colored Hypergraph} \equiv \text{Min Cut in Colored Graph}$.
Randomized Algorithm

- **Definition**: Algorithm allowed to do some random decisions as it processes the input

- We will prove that our algorithm has a polynomial overall running time to get a high probability of success

- Based on the following correspondence: MBD $\equiv$ Min Cut in Colored Hypergraph $\equiv$ Min Cut in Colored Graph

$$C = \{\square, \blacksquare, \square, \blacksquare, \square, \blacksquare\}$$
Randomized Algorithm

- **Definition**: Algorithm allowed to do some random decisions as it processes the input.
- We will prove that our algorithm has a polynomial overall running time to get a high probability of success.
- Based on the following correspondence: $\text{MBD} \equiv \text{Min Cut in Colored Hypergraph} \equiv \text{Min Cut in Colored Graph}$

![Diagram of random graph and cut]

$C = \{ \text{vertices} \}$
Randomized Algorithm

- **Definition**: Algorithm allowed to do some random decisions as it processes the input.
- We will prove that our algorithm has a polynomial overall running time to get a high probability of success.
- Based on the following correspondence: $\text{MBD} \equiv \text{Min Cut in Colored Hypergraph} \equiv \text{Min Cut in Colored Graph}$.
Min Cut in Colored Graph

- Randomized algorithm using colored contraction algorithm inspired by folklore algorithm \(^1\):

\(^1\) J. Kleinberg and E. Tardos
Min Cut in Colored Graph

- Randomized algorithm using colored contraction algorithm inspired by folklore algorithm\(^1\):

Random choice of a color and contract all edges of this color.

\(^1\)J. Kleinberg and E. Tardos
Min Cut in Colored Graph

- Randomized algorithm using colored contraction algorithm inspired by folklore algorithm \(^1\):

\(^1\) J. Kleinberg and E. Tardos

Guillaume Blin

Complexity Insights of the Minimum Duplication Problem
Min Cut in Colored Graph

Randomized algorithm using colored contraction algorithm inspired by folklore algorithm $^1$:

Until you reach only two super-vertices

$^1$J. Kleinberg and E. Tardos
Min Cut in Colored Graph

Randomized algorithm using colored contraction algorithm inspired by folklore algorithm $^1$:

At each step $mul(c)$ contractions = $|V|$ decreases from $mul(c)$

$^1$J. Kleinberg and E. Tardos
Simple randomized algorithm, but what about performance analysis?
⇒ It returns opt with probability \( \geq (|V|^{2k})^{-1} \) where
\[ k = \max_{c \in C} \text{mul}(c) \]

Let OPT = \# colors in optimal cut set
Min Cut in Colored Graph

- Simple randomized algorithm, but what about performance analysis?
  \[ \Rightarrow \text{It returns opt with probability } \geq \left( |V|^{2k} \right)^{-1} \text{ where } k = \max_{c \in C} \text{mul}(c) \]

- Let \( OPT = \# \text{ colors in optimal cut set} \)
- \( Rk1: \forall v \in V, d(v) \geq OPT \)

otherwise \( (\{v\}, \{V \setminus v\}) \) would be better solution
Min Cut in Colored Graph

- Simple randomized algorithm, but what about performance analysis?
  \[
  \Rightarrow \text{It returns opt with probability } \geq \left(\frac{|V|^{2k}}{2}\right)^{-1} \text{ where } \\
k = \max_{c \in C_{\text{mul}}(c)}
\]

- Let \( \text{OPT} = \# \text{ colors in optimal cut set} \)
- Rk1: \( \forall v \in V, \ d(v) \geq \text{OPT} \)
- Rk2: \( \frac{\text{OPT} \cdot |V|}{2} \leq |E| \)

\[
\sum_{v \in V} \frac{(d(v))}{2} \leq |E|
\]
Min Cut in Colored Graph

- Simple randomized algorithm, but what about performance analysis?
  ⇒ It returns opt with probability $\geq (|V|^{2k})^{-1}$ where $k = \max_{c \in C} \text{mul}(c)$

- Let $\text{OPT} = \# \text{colors in optimal cut set}$
- Rk1: $\forall v \in V, d(v) \geq \text{OPT}$
- Rk2: $\frac{\text{OPT} \cdot |V|}{2} \leq |E|$
- Rk3: $|E| \leq k \cdot |C|$

since each color cannot be used more than $k$ edges in $E$
Min Cut in Colored Graph

Simple randomized algorithm, but what about performance analysis?
→ It returns opt with probability \( \geq (|V|^{2k})^{-1} \)
where
\( k = \max_{c \in C} \text{mul}(c) \)

Let \( OPT = \# \) colors in optimal cut set

\( \text{Rk1: } \forall v \in V, d(v) \geq OPT \)
\( \text{Rk2: } \frac{OPT \cdot |V|}{2} \leq |E| \)
\( \text{Rk3: } |E| \leq k \cdot |C| \)

\( \Rightarrow OPT \cdot |V| \leq 2 \cdot |E| \leq 2k \cdot |C| \)
Min Cut in Colored Graph

- The probability $P_r[F_j]$ of failing at $j^{th}$ contraction considering we are left with $C'$ colors, and $|V'| = |V| - i$ vertices
Min Cut in Colored Graph

- The probability $P_r[F_j]$ of failing at $j^{th}$ contraction considering we are left with $C'$ colors, and $|V'| = |V| - i$ vertices
- = choosing a color among the OPT ones
The probability $P_r[F_j]$ of failing at $j^{th}$ contraction considering we are left with $C'$ colors, and $|V'| = |V| - i$ vertices

- = choosing a color among the OPT ones

$P_r[F_j] \leq \frac{OPT}{|C'|}$
Min Cut in Colored Graph

- The probability $P_r[F_j]$ of failing at $j^{th}$ contraction considering we are left with $C'$ colors, and $|V'| = |V| - i$ vertices
- $= \text{choosing a color among the OPT ones}$
- $P_r[F_j] \leq \frac{OPT}{|C'|} \leq \frac{2k.|C'|}{|V'|.|C'|}$ since $OPT.|V| \leq 2.|E| \leq 2k.|C|$
The probability $P_r[F_j]$ of failing at $j^{th}$ contraction considering we are left with $C'$ colors, and $|V'| = |V| - i$ vertices

= choosing a color among the OPT ones

$P_r[F_j] \leq \frac{\text{OPT}}{|C'|} \leq \frac{2k}{|V'|}$
Min Cut in Colored Graph

- The probability \( P_r[F_j] \) of failing at \( j^{th} \) contraction considering we are left with \( C' \) colors, and \( |V'| = |V| - i \) vertices
- = choosing a color among the OPT ones
- \( P_r[F_j] \leq \frac{OPT}{|C'|} \leq \frac{2k}{|V'|} \)
- \( P_r[\text{Success}] \geq \prod_{j=0}^{\infty} (1 - P_r[F_j]) \)
Min Cut in Colored Graph

- The probability $P_r[F_j]$ of failing at $j^{th}$ contraction considering we are left with $C'$ colors, and $|V'| = |V| - i$ vertices
- $= \text{choosing a color among the OPT ones}$
- $P_r[F_j] \leq \frac{\text{OPT}}{|C'|} \leq \frac{2k}{|V'|}$
- $P_r[\text{Success}] \geq \prod_{j=0} \left(1 - P_r[F_j] \right) \geq \prod_{j=0} \left(1 - \frac{2k}{|V| - i} \right)$
The probability $P_{r[F_j]}$ of failing at $j^{th}$ contraction considering we are left with $C'$ colors, and $|V'| = |V| - i$ vertices

= choosing a color among the OPT ones

$P_{r[F_j]} \leq \frac{OPT}{|C'|} \leq \frac{2k}{|V'|}$

$P_{r[Success]} \geq \prod_{j=0}^{\infty}(1 - P_{r[F_j]}) \geq \prod_{j=0}^{\infty}(1 - \frac{2k}{|V'| - i})$

$\geq \prod_{j=0}^{\infty}(\frac{|V| - i - 2k}{|V| - i})$
Min Cut in Colored Graph

- The probability $P_r[F_j]$ of failing at $j^{th}$ contraction considering we are left with $C'$ colors, and $|V'| = |V| - i$ vertices
- = choosing a color among the OPT ones
- $P_r[F_j] \leq \frac{OPT}{|C'|} \leq \frac{2k}{|V'|}$
- $P_r[Success] \geq \prod_{j=0}^{\infty} (1 - P_r[F_j]) \geq \prod_{j=0}^{\infty} (1 - \frac{2k}{|V| - i})$
- $\geq \prod_{j=0}^{\infty} (\frac{|V| - i - 2k}{|V| - i}) \geq \frac{1}{|V|^{2k}}$
The probability $P_r[F_j]$ of failing at $j^{th}$ contraction considering we are left with $C'$ colors, and $|V'| = |V| - i$ vertices

= choosing a color among the OPT ones

$P_r[F_j] \leq \frac{OPT}{|C'|} \leq \frac{2k}{|V'|}$

$P_r[Success] \geq \prod_{j=0} \left(1 - P_r[F_j] \right) \geq \prod_{j=0} \left(1 - \frac{2k}{|V| - i} \right)$

$\geq \prod_{j=0} \left(\frac{|V| - i - 2k}{|V| - i} \right) \geq \frac{1}{|V|^{2k}}$

A single run of the algorithm fails to find the optimal with probability at most $(1 - (|V|^{2k})^{-1})$
Min Cut in Colored Graph

The probability $P_r[F_j]$ of failing at $j^{th}$ contraction considering we are left with $C'$ colors, and $|V'| = |V| - i$ vertices

= choosing a color among the OPT ones

$P_r[F_j] \leq \frac{OPT}{|C'|} \leq \frac{2k}{|V'|}$

$P_r[Success] \geq \prod_{j=0} |V| - i - 2k \geq \prod_{j=0} (1 - \frac{2k}{|V| - i})$

$\geq \prod_{j=0} (\frac{|V| - i - 2k}{|V| - i}) \geq \frac{1}{|V|^{2k}}$

A single run of the algorithm fails to find the optimal with probability at most $(1 - (|V|^{2k})^{-1})$

Running the algorithm $|V|^{2k} ln|V|$ will lead to having no success with a probability at most $\frac{1}{|V|}$; implying a bounded $k$
Min Cut in Colored Graph

- The probability $P_r[F_j]$ of failing at $j^{th}$ contraction considering we are left with $C'$ colors, and $|V'| = |V| - i$ vertices
- $= \text{choosing a color among the OPT ones}$
- $P_r[F_j] \leq \frac{\text{OPT}}{|C'|} \leq \frac{2k}{|V'|}$
- $P_r[\text{Success}] \geq \prod_{j=0}^{i} (1 - P_r[F_j]) \geq \prod_{j=0}^{i} (1 - \frac{2k}{|V|})$
  $\geq \prod_{j=0}^{i} (\frac{|V| - i - 2k}{|V| - i}) \geq \frac{1}{|V|^{2k}}$
- A single run of the algorithm fails to find the optimal with probability at most $(1 - (|V|^{2k})^{-1})$
- Running the algorithm $|V|^{2k} ln|V|$ will lead to having no success with a probability at most $\frac{1}{|V|}$; implying a bounded $k$
- $\Rightarrow$ MBD is randomized polynomial when the gene trees are of bounded depth
Open problem

- What is the complexity of Minimum Colored Cut?
- What is the complexity of MDB considering unbounded depth gene trees?
Complexity Insights of the Minimum Duplication Problem

Guillaume Blin    Paola Bonizzoni    Riccardo Dondi
Romeo Rizzi   Florian Sikora

Université Paris-Est Marne-la-Vallée, LIGM - UMR CNRS 8049, France
DISCo, Università degli Studi di Milano-Bicocca, - Milano, Italy
DSLCSC, Università degli Studi di Bergamo, - Bergamo, Italy
DIMI, Università di Udine - Udine, Italy
Lehrstuhl fur Bioinformatik, Friedrich-Schiller-Universität Jena, Germany

January 2012