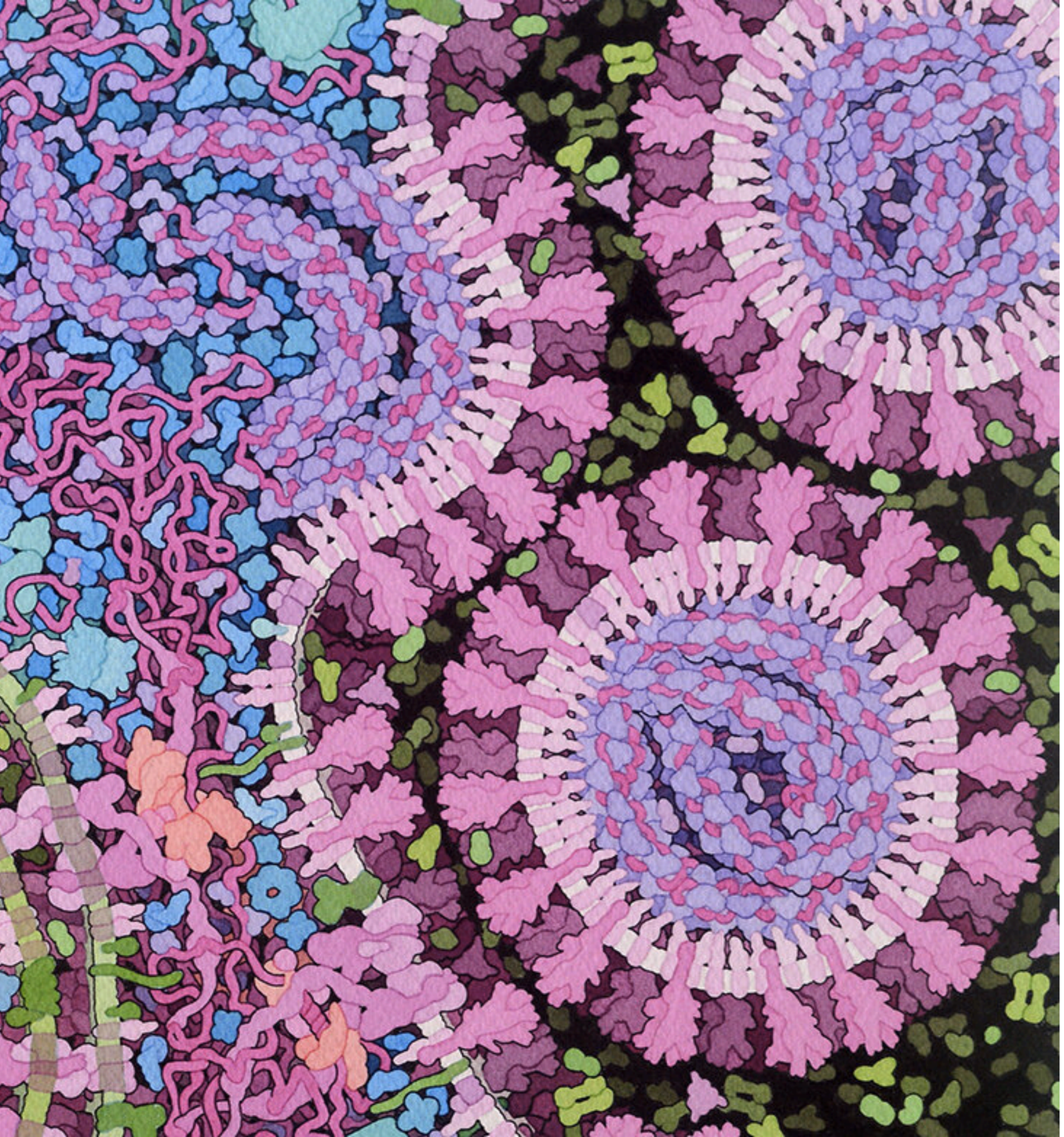


بسم الله الرحمن الرحیم

ژنومیک محاسباتی

جلسه ۹: یک الگوریتم برای بازسازی درخت تبارزایی ویژگی‌مبنا

ترم پاییز ۱۴۰۰-۱۴۰۱

درخت تبارزایی ویژگی_مبنا

- ورودی: ماتریس ویژگی‌ها
  - سطر: گونه‌ها
  - ستون: ویژگی‌ها
- خروجی: یک درخت روی گونه‌ها

Approximate Maximum Parsimony and Ancestral Maximum Likelihood

Publisher: IEEE

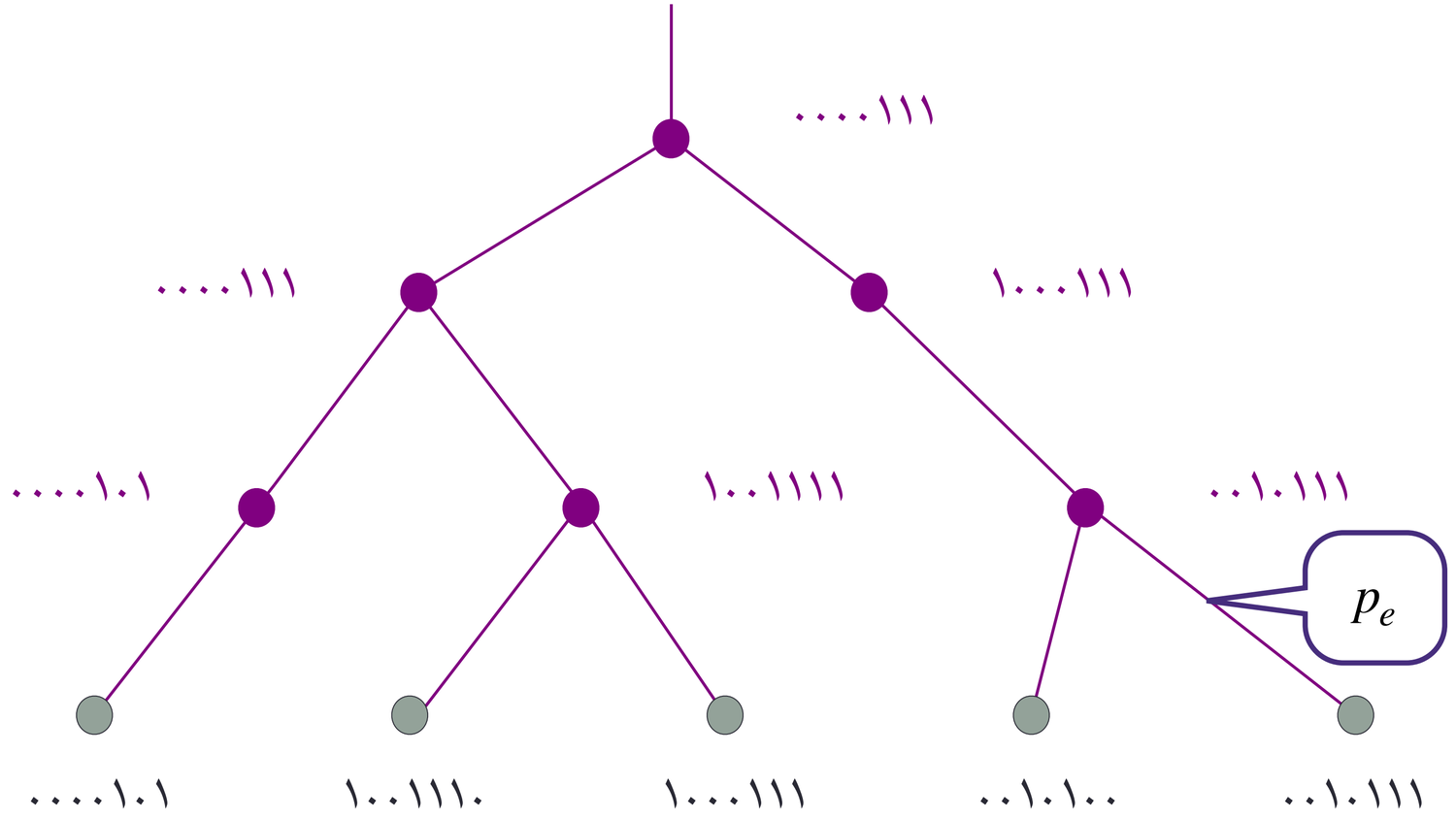
Noga Alon ; Benny Chor ; Fabio Pardi ; Anat Rapoport **All Authors**

Abstract:
We explore the maximum parsimony (MP) and ancestral maximum likelihood (AML) criteria in phylogenetic tree reconstruction. Both problems are NP-hard, so we seek approximate solutions. We formulate the two problems as Steiner tree problems under appropriate distances. The gist of our approach is the succinct characterization of Steiner trees for a small number of leaves for the two distances. This enables the use of known Steiner tree approximation algorithms. The approach leads to a 16/9 approximation ratio for AML and asymptotically to a 1.55 approximation ratio for MP.

Published in: IEEE/ACM Transactions on Computational Biology and Bioinformatics (Volume: 7 , Issue: 1, Jan.-March 2010)

مدل احتمالاتی

- مدل ۲ حالته:
  - ۰ و ۱
- ورودی: رشته‌های برگ‌ها
- خروجی:
  - درخت روی برگ‌ها
  - رشته‌های (۰و۱) روی راس‌های میانی
  - احتمال تغییر روی یال‌ها
- احتمال درخت=



ANCESTRAL MAXIMUM LIKELIHOOD VERSION I

Input: A set S of m binary sequences, each of length n.

Goal: Find a tree T with m leaves, an assignment $e \longmapsto p_e \in [0,1]$ of edge probabilities, and a labelling $\lambda : V(T) \to \{0,1\}^n$ of the vertices such that

1) The m labels of the leaves are exactly the sequences from S, and
2) $\prod_{e \in E(T)} p_e^{d_e}(1-p_e)^{n-d_e}$ (where d_e is the Hamming distance of the two labels across the edge e) is maximized.

ANCESTRAL MAXIMUM LIKELIHOOD VERSION I

Input: A set S of m binary sequences, each of length n.

Goal: Find a tree T with m leaves, an assignment $e \longmapsto p_e \in [0,1]$ of edge probabilities, and a labelling $\lambda : V(T) \to \{0,1\}^n$ of the vertices such that

1) The m labels of the leaves are exactly the sequences from S, and
2) $\prod_{e \in E(T)} p_e^{d_e}(1-p_e)^{n-d_e}$ (where d_e is the Hamming distance of the two labels across the edge e) is maximized.

رشته‌های روی راس‌های میانی

بهترین p_eها

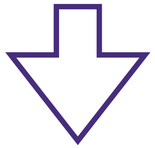
ANCESTRAL MAXIMUM LIKELIHOOD VERSION I

Input: A set S of m binary sequences, each of length n.

Goal: Find a tree T with m leaves, an assignment $e \longmapsto p_e \in [0,1]$ of edge probabilities, and a labelling $\lambda : V(T) \to \{0,1\}^n$ of the vertices such that

1) The m labels of the leaves are exactly the sequences from S, and
2) $\prod_{e \in E(T)} p_e^{d_e}(1-p_e)^{n-d_e}$ (where d_e is the Hamming distance of the two labels across the edge e) is maximized.

رشته‌های روی راس‌های میانی



بهترین p_eها

$$\max \quad p_e^{d_e}(1-p_e)^{n-d_e}$$

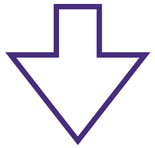
ANCESTRAL MAXIMUM LIKELIHOOD VERSION I

Input: A set S of m binary sequences, each of length n.

Goal: Find a tree T with m leaves, an assignment $e \longmapsto p_e \in [0,1]$ of edge probabilities, and a labelling $\lambda : V(T) \to \{0,1\}^n$ of the vertices such that

1) The m labels of the leaves are exactly the sequences from S, and
2) $\prod_{e \in E(T)} p_e^{d_e}(1-p_e)^{n-d_e}$ (where d_e is the Hamming distance of the two labels across the edge e) is maximized.

رشته‌های روی راس‌های میانی



بهترین p_eها

$$\max \quad p_e^{d_e}(1-p_e)^{n-d_e}$$

$$\nabla = d_e p_e^{d_e-1}(1-p_e)^{n-d_e} - (n-d_e)p_e^{d_e}(1-p_e)^{n-d_e-1}$$

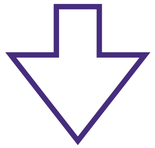
ANCESTRAL MAXIMUM LIKELIHOOD VERSION I

Input: A set S of m binary sequences, each of length n.

Goal: Find a tree T with m leaves, an assignment $e \longmapsto p_e \in [0,1]$ of edge probabilities, and a labelling $\lambda : V(T) \to \{0,1\}^n$ of the vertices such that

1) The m labels of the leaves are exactly the sequences from S, and
2) $\prod_{e \in E(T)} p_e^{d_e}(1-p_e)^{n-d_e}$ (where d_e is the Hamming distance of the two labels across the edge e) is maximized.

رشته‌های روی راس‌های میانی



بهترین p_eها

$$\max \quad p_e^{d_e}(1-p_e)^{n-d_e}$$

$$\nabla = d_e p_e^{d_e-1}(1-p_e)^{n-d_e} - (n-d_e)p_e^{d_e}(1-p_e)^{n-d_e-1} = 0$$

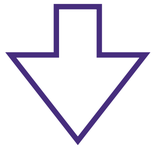
ANCESTRAL MAXIMUM LIKELIHOOD VERSION I

Input: A set S of m binary sequences, each of length n.

Goal: Find a tree T with m leaves, an assignment $e \longmapsto p_e \in [0,1]$ of edge probabilities, and a labelling $\lambda : V(T) \to \{0,1\}^n$ of the vertices such that

1) The m labels of the leaves are exactly the sequences from S, and
2) $\prod_{e \in E(T)} p_e^{d_e}(1-p_e)^{n-d_e}$ (where d_e is the Hamming distance of the two labels across the edge e) is maximized.

رشته‌های روی راس‌های میانی



بهترین p_eها

$$\max \quad p_e^{d_e}(1-p_e)^{n-d_e}$$

$$\nabla = d_e p_e^{d_e-1}(1-p_e)^{n-d_e} - (n-d_e)p_e^{d_e}(1-p_e)^{n-d_e-1} = 0$$

$$\nabla = d_e(1-p_e) - (n-d_e)p_e = 0$$

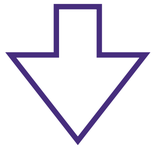
ANCESTRAL MAXIMUM LIKELIHOOD VERSION I

Input: A set S of m binary sequences, each of length n.

Goal: Find a tree T with m leaves, an assignment $e \longmapsto p_e \in [0,1]$ of edge probabilities, and a labelling $\lambda : V(T) \to \{0,1\}^n$ of the vertices such that

1) The m labels of the leaves are exactly the sequences from S, and
2) $\prod_{e \in E(T)} p_e^{d_e}(1-p_e)^{n-d_e}$ (where d_e is the Hamming distance of the two labels across the edge e) is maximized.

رشته‌های روی راس‌های میانی



بهترین p_eها

$$\max \quad p_e^{d_e}(1-p_e)^{n-d_e}$$

$$\nabla = d_e p_e^{d_e-1}(1-p_e)^{n-d_e} - (n-d_e)p_e^{d_e}(1-p_e)^{n-d_e-1} = 0$$

$$\nabla = d_e(1-p_e) - (n-d_e)p_e = 0$$

$$p_e = d_e/n.$$

ANCESTRAL MAXIMUM LIKELIHOOD VERSION I

Input: A set S of m binary sequences, each of length n.

Goal: Find a tree T with m leaves, an assignment $e \longmapsto p_e \in [0,1]$ of edge probabilities, and a labelling $\lambda : V(T) \to \{0,1\}^n$ of the vertices such that

1) The m labels of the leaves are exactly the sequences from S, and
2) $\prod_{e \in E(T)} p_e^{d_e}(1-p_e)^{n-d_e}$ (where d_e is the Hamming distance of the two labels across the edge e) is maximized.

$$\left(\frac{d_e}{n}\right)^{d_e/n}\left(1-\frac{d_e}{n}\right)^{1-d_e/n}$$

رشته‌های روی راس‌های میانی

بهترین p_eها

$$\max \quad p_e^{d_e}(1-p_e)^{n-d_e}$$

$$\nabla = d_e p_e^{d_e-1}(1-p_e)^{n-d_e} - (n-d_e)p_e^{d_e}(1-p_e)^{n-d_e-1} = 0$$

$$\nabla = d_e(1-p_e) - (n-d_e)p_e = 0$$

$$p_e = d_e/n.$$

ANCESTRAL MAXIMUM LIKELIHOOD VERSION I

Input: A set S of m binary sequences, each of length n.

Goal: Find a tree T with m leaves, an assignment $e \longmapsto p_e \in [0,1]$ of edge probabilities, and a labelling $\lambda : V(T) \to \{0,1\}^n$ of the vertices such that

1) The m labels of the leaves are exactly the sequences from S, and
2) $\prod_{e \in E(T)} p_e^{d_e}(1-p_e)^{n-d_e}$ (where d_e is the Hamming distance of the two labels across the edge e) is maximized.

$$\left(\frac{d_e}{n}\right)^{d_e/n}\left(1-\frac{d_e}{n}\right)^{1-d_e/n}$$

ANCESTRAL MAXIMUM LIKELIHOOD VERSION I

Input: A set S of m binary sequences, each of length n.

Goal: Find a tree T with m leaves, an assignment $e \longmapsto p_e \in [0,1]$ of edge probabilities, and a labelling $\lambda : V(T) \to \{0,1\}^n$ of the vertices such that

1) The m labels of the leaves are exactly the sequences from S, and
2) $\prod_{e \in E(T)} p_e^{d_e} (1-p_e)^{n-d_e}$ (where d_e is the Hamming distance of the two labels across the edge e) is maximized.

$$\left(\frac{d_e}{n}\right)^{d_e/n} \left(1-\frac{d_e}{n}\right)^{1-d_e/n}$$

$$\sum_{e \in E(T)} \left(\frac{d_e}{n} \log\left(\frac{d_e}{n}\right) + \left(1-\frac{d_e}{n}\right) \log\left(1-\frac{d_e}{n}\right)\right)$$

ANCESTRAL MAXIMUM LIKELIHOOD VERSION I

Input: A set S of m binary sequences, each of length n.

Goal: Find a tree T with m leaves, an assignment $e \longmapsto p_e \in [0,1]$ of edge probabilities, and a labelling $\lambda : V(T) \to \{0,1\}^n$ of the vertices such that

1) The m labels of the leaves are exactly the sequences from S, and
2) $\prod_{e \in E(T)} p_e^{d_e} (1-p_e)^{n-d_e}$ (where d_e is the Hamming distance of the two labels across the edge e) is maximized.

$$\left(\frac{d_e}{n}\right)^{d_e/n} \left(1-\frac{d_e}{n}\right)^{1-d_e/n}$$

$$\sum_{e \in E(T)} \left(\frac{d_e}{n} \log\left(\frac{d_e}{n}\right) + \left(1-\frac{d_e}{n}\right) \log\left(1-\frac{d_e}{n}\right)\right)$$

$$= \sum_{e \in E(T)} -H_2\left(\frac{d_e}{n}\right),$$

ANCESTRAL MAXIMUM LIKELIHOOD VERSION I

Input: A set S of m binary sequences, each of length n.

Goal: Find a tree T with m leaves, an assignment $e \longmapsto p_e \in [0,1]$ of edge probabilities, and a labelling $\lambda : V(T) \to \{0,1\}^n$ of the vertices such that

1) The m labels of the leaves are exactly the sequences from S, and
2) $\prod_{e \in E(T)} p_e^{d_e}(1-p_e)^{n-d_e}$ (where d_e is the Hamming distance of the two labels across the edge e) is maximized.

$$\left(\frac{d_e}{n}\right)^{d_e/n}\left(1-\frac{d_e}{n}\right)^{1-d_e/n}$$

$$\sum_{e \in E(T)} \left(\frac{d_e}{n} \log\left(\frac{d_e}{n}\right) + \left(1-\frac{d_e}{n}\right)\log\left(1-\frac{d_e}{n}\right)\right)$$

$$= \sum_{e \in E(T)} -H_2\left(\frac{d_e}{n}\right),$$

$$-p\log_2(p) - (1-p)\log_2(1-p)$$

ANCESTRAL MAXIMUM LIKELIHOOD VERSION II

ANCESTRAL MAXIMUM LIKELIHOOD (VERSION II)

Input: A set S of m binary sequences, each of length n.

Goal: Find a tree T with m leaves and a labelling $\lambda : V(T) \to \{0,1\}^n$ of the vertices such that

1) The m labels of the leaves are exactly the sequences from S, and
2) $\sum_{e \in E(T)} H\left(d_e/n\right)$ is minimized.

$-p\log_2(p) - (1-p)\log_2(1-p)$

MAXIMUM PARSIMONY

Input: A set S of m binary sequences, each of length n.

Goal: Find a tree T with m leaves and a labelling $\lambda : V(T) \to \{0,1\}^n$ of the vertices such that

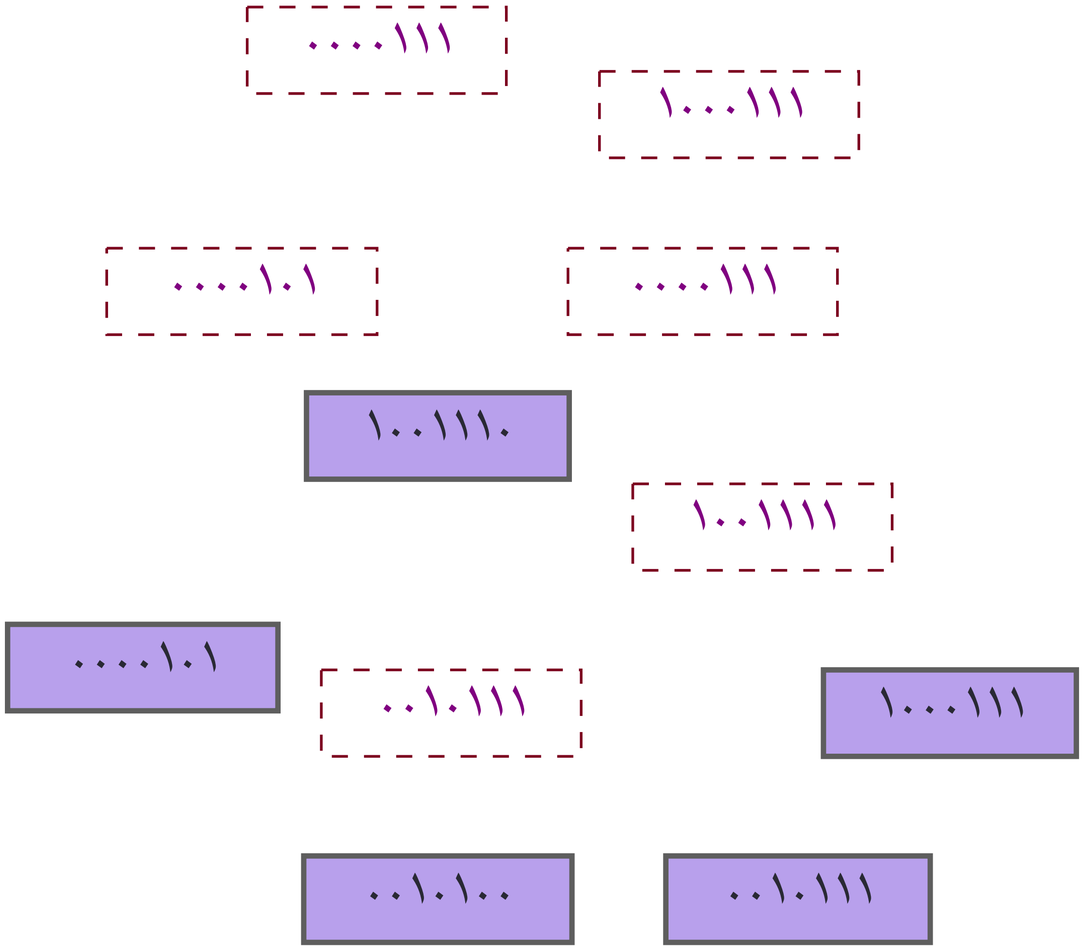
1) The m labels of the leaves are exactly the sequences from S, and
2) $\sum_{e \in E(T)} \left(d_e/n\right)$ is minimized.

MAXIMUM PARSIMONY

Input: A set S of m binary sequences, each of length n.

Goal: Find a tree T with m leaves and a labelling $\lambda : V(T) \to \{0,1\}^n$ of the vertices such that

1) The m labels of the leaves are exactly the sequences from S, and
2) $\sum_{e \in E(T)} (d_e/n)$ is minimized.



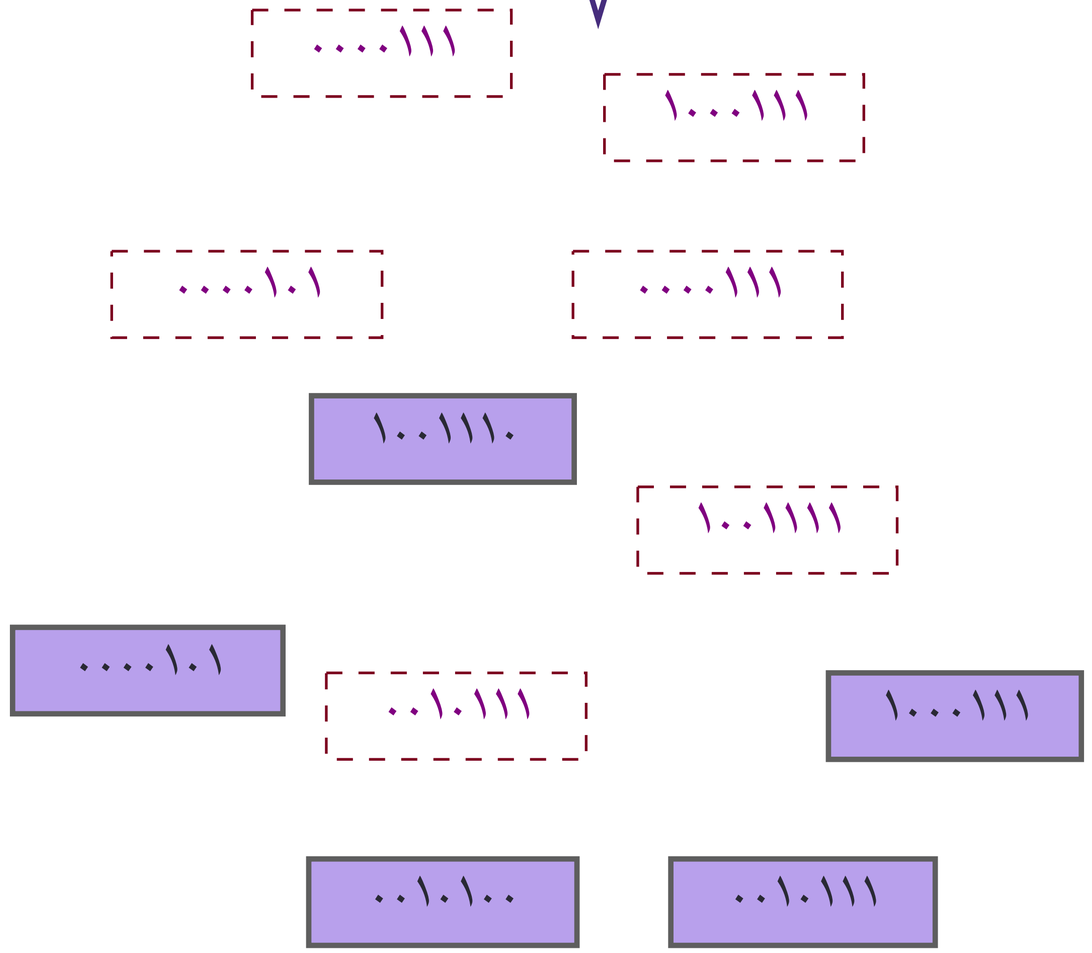
MAXIMUM PARSIMONY

Input: A set S of m binary sequences, each of length n.

Goal: Find a tree T with m leaves and a labelling $\lambda : V(T) \to \{0,1\}^n$ of the vertices such that

1) The m labels of the leaves are exactly the sequences from S, and
2) $\sum_{e \in E(T)} (d_e/n)$ is minimized.

زیرگراف همبند روی راس‌های خاص



MAXIMUM PARSIMONY

Input: A set S of m binary sequences, each of length n.

Goal: Find a tree T with m leaves and a labelling $\lambda : V(T) \to \{0,1\}^n$ of the vertices such that

1) The m labels of the leaves are exactly the sequences from S, and
2) $\sum_{e \in E(T)} (d_e/n)$ is minimized.

مسئله ؟

زیرگراف همبند روی راس‌های خاص

۰۰۰۰۱۱۱

۱۰۰۰۱۱۱

۰۰۰۰۱۰۱

۰۰۰۰۱۱۱

۱۰۰۱۱۱۰

۱۰۰۱۱۱۱

۰۰۰۰۱۰۱

۰۰۱۰۱۱۱

۱۰۰۰۱۱۱

۰۰۱۰۱۰۰

۰۰۱۰۱۱۱

مسئله درخت اشتاینر

مسئله ؟

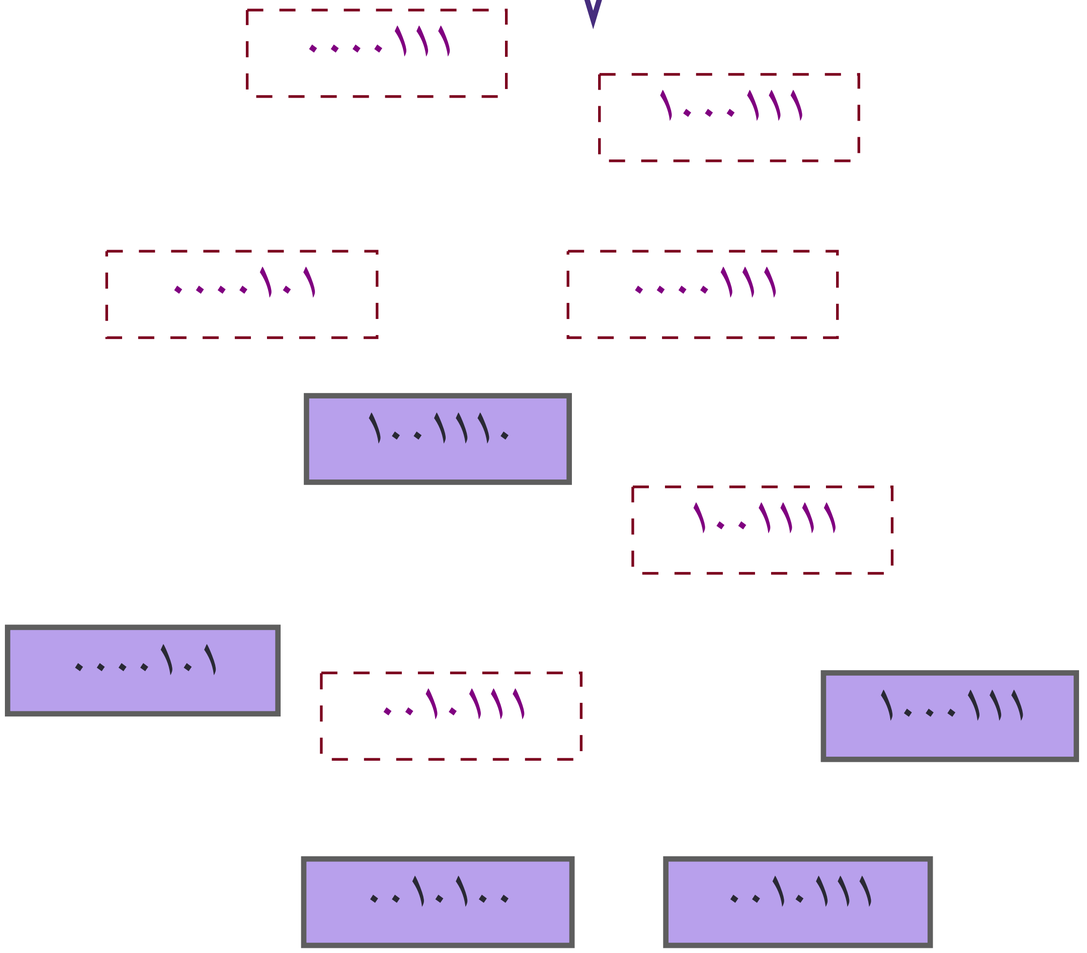
زیرگراف همبند روی راس‌های خاص

MAXIMUM PARSIMONY

Input: A set S of m binary sequences, each of length n.

Goal: Find a tree T with m leaves and a labelling $\lambda : V(T) \to \{0,1\}^n$ of the vertices such that

1) The m labels of the leaves are exactly the sequences from S, and
2) $\sum_{e \in E(T)} (d_e/n)$ is minimized.



مسئله درخت اشتاینر

- ورودی: گراف + راس‌های خاص (ترمینال)
- خروجی:
  - زیردرخت همبند شامل راس‌های ترمینال
  - هزینه کمینه

مسئله درخت اشتاینر

- ورودی: گراف + راس‌های خاص (ترمینال)
- خروجی:
  - زیردرخت همبند شامل راس‌های ترمینال
  - هزینه کمینه

نسخه هندسی

$N=5, L\approx 3.891$

$cf.L_{\curvearrowright}=4, L_{\bigstar}\approx 4.253$

مسئله درخت اشتاینر

- ورودی: گراف + راس‌های خاص (ترمینال)
- خروجی:
  - زیردرخت همبند شامل راس‌های ترمینال
  - هزینه کمینه

نسخه هندسی

$N=5, L\approx 3.891$

$cf. L\curvearrowright=4, L\bigstar\approx 4.253$

- سختی: اگر P<>NP، نمی‌توان به ازای هر $\epsilon > 0$ الگوریتم $1+\epsilon$ـ تقریب پیدا کرد
  - با ضریب ۹۶/۹۵ ممکن نیست

مسئله درخت اشتاینر

- ورودی: گراف + راس‌های خاص (ترمینال)
- خروجی:
  - زیردرخت همبند شامل راس‌های ترمینال
  - هزینه کمینه

نسخه هندسی

$N=5, L\approx 3.891$

$cf. L\curvearrowright=4, L\bigstar\approx 4.253$

- سختی: اگر P<>NP، نمی‌توان به ازای هر $\epsilon > 0$ الگوریتم $1+\epsilon$ـ تقریب پیدا کرد
  - با ضریب ۹۶/۹۵ ممکن نیست
- الگوریتم‌های تقریبی
  - الگوریتم ۱٫۳۹ـ تقریب

ANCESTRAL MAXIMUM LIKELIHOOD VERSION II

ANCESTRAL MAXIMUM LIKELIHOOD (VERSION II)

Input: A set S of m binary sequences, each of length n.

Goal: Find a tree T with m leaves and a labelling $\lambda : V(T) \to \{0,1\}^n$ of the vertices such that

1) The m labels of the leaves are exactly the sequences from S, and
2) $\sum_{e \in E(T)} H\left(d_e/n\right)$ is minimized.

$-p\log_2(p)-(1-p)\log_2(1-p)$

مسئله درخت اشتاینر

MAXIMUM PARSIMONY

Input: A set S of m binary sequences, each of length n.

Goal: Find a tree T with m leaves and a labelling $\lambda : V(T) \to \{0,1\}^n$ of the vertices such that

1) The m labels of the leaves are exactly the sequences from S, and
2) $\sum_{e \in E(T)} \left(d_e/n\right)$ is minimized.

مسئله درخت اشتاینر

ANCESTRAL MAXIMUM LIKELIHOOD VERSION II

ANCESTRAL MAXIMUM LIKELIHOOD (VERSION II)

Input: A set S of m binary sequences, each of length n.

Goal: Find a tree T with m leaves and a labelling $\lambda : V(T) \to \{0,1\}^n$ of the vertices such that

1) The m labels of the leaves are exactly the sequences from S, and
2) $\sum_{e \in E(T)} H\left(d_e/n\right)$ is minimized.

$-p\log_2(p)-(1-p)\log_2(1-p)$

مسئله درخت اشتاینر

MAXIMUM PARSIMONY

Input: A set S of m binary sequences, each of length n.

Goal: Find a tree T with m leaves and a labelling $\lambda : V(T) \to \{0,1\}^n$ of the vertices such that

1) The m labels of the leaves are exactly the sequences from S, and
2) $\sum_{e \in E(T)} \left(d_e/n\right)$ is minimized.

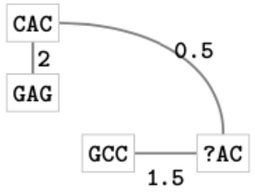
مسئله درخت اشتاینر

مشکل کجاست؟

الگوریتم تقریبی مناسب

1 - Initialize

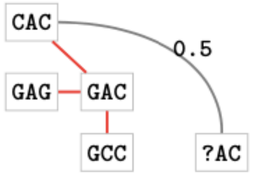
Find the minimum spanning tree (MST) between input nodes



2- Main loop

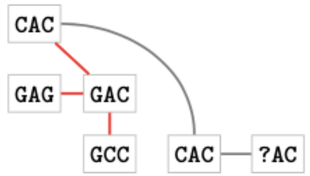
3 - Construct Tree

For every tree t in the stack, if it is compatible with the current tree, update the tree by adding t.



4 - Impute

Replace each node which has missing values with a fully imputed one, which minimizes the cost of the tree.

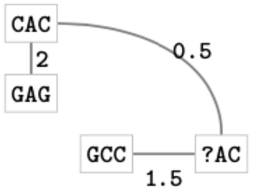


- الگوریتم تقریبی برمن (برای درخت اشتاینر):
- T = یک درخت فراگیر بین راس‌های ترمینال
- به ازای هر زیرمجموعه k_عضوی از ترمینال‌ها K
- به ازای هر توپولوژی درخت t روی این مجموعه
- tau: بهترین درخت با توپولوژی t روی K:
- آیا اضافه کردن این درخت کمکی می‌کند؟
- درخت را به پشته اضافه کن و T را به‌روز کن
- درخت‌های اضافه شده از انتها را یکی یکی به T اضافه کن (اگر هنوز می‌توانستیم)

The algorithm, explained

1 - Initialize

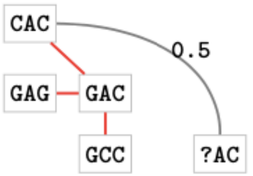
Find the minimum spanning tree (MST) between input nodes



2- Main loop

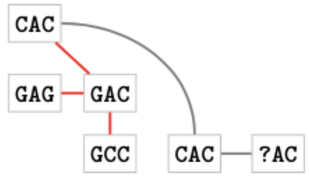
3 - Construct Tree

For every tree t in the stack, if it is compatible with the current tree, update the tree by adding t.



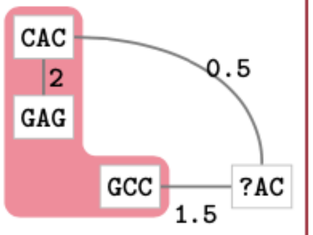
4 - Impute

Replace each node which has missing values with a fully imputed one, which minimizes the cost of the tree.



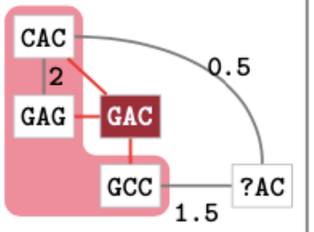
2 - Enumerate Restricted Trees

For every k-subset of the samples K, and every tree topologies t with K as its leaves, perform the following steps:



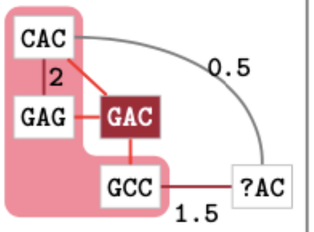
2A - Fill internal nodes of t

Find the best evolutionary tree τ with K as its leaves by a dynamic program



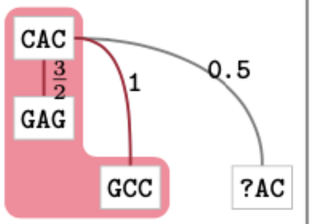
2B - Find bridges

Find bridges β, which are edges that should be removed from the spanning tree between S to avoid cycles after adding τ to the tree.



2C - Project τ onto S

Let gain $=$ cost(t) $-$ cost(β)
If gain > 0,
push t to the stack,
add projection of the bridges on K
weight of new edge $=$ weight of bridge $-$gain



الگوریتم تقریبی مناسب

- الگوریتم تقریبی برمن (برای درخت اشتاینر):
  - T = یک درخت فراگیر بین راس‌های ترمینال
  - به ازای هر زیرمجموعه k_عضوی از ترمینال‌ها K
    - به ازای هر توپولوژی درخت t روی این مجموعه
      - tau: بهترین درخت با توپولوژی t روی K:
      - آیا اضافه کردن این درخت کمکی می‌کند؟
        - درخت را به پشته اضافه کن و T را به‌روز کن
  - درخت‌های اضافه شده از انتها را یکی یکی به T اضافه کن (اگر هنوز می‌توانستیم)

قضیه: ضریب تقریب الگوریتم برمن

- برای k=۳، الگوریتم برمن یک ۱۱/۶
- برای k=۴، الگوریتم برمن یک ۱۶/۹

الگوریتم تقریبی مناسب برای ما

MAXIMUM PARSIMONY

Input: A set S of m binary sequences, each of length n.

Goal: Find a tree T with m leaves and a labelling $\lambda : V(T) \to \{0,1\}^n$ of the vertices such that

1) The m labels of the leaves are exactly the sequences from S, and
2) $\sum_{e \in E(T)} (d_e/n)$ is minimized.

مسئله درخت اشتاینر

- الگوریتم تقریبی برمن (برای درخت اشتاینر):
- T = یک درخت فراگیر بین راس‌های ترمینال
- به ازای هر زیرمجموعه k_عضوی از ترمینال‌ها K
- به ازای هر توپولوژی درخت t روی این مجموعه
- tau: بهترین درخت با توپولوژی t روی K:
- آیا اضافه کردن این درخت کمکی می‌کند؟
- درخت را به پشته اضافه کن و T را به‌روز کن
- درخت‌های اضافه شده از انتها را یکی یکی به T اضافه کن (اگر هنوز می‌توانستیم)

الگوریتم تقریبی مناسب برای ما

گراف عظیم!

MAXIMUM PARSIMONY

Input: A set S of m binary sequences, each of length n.

Goal: Find a tree T with m leaves and a labelling $\lambda : V(T) \to \{0,1\}^n$ of the vertices such that

1) The m labels of the leaves are exactly the sequences from S, and
2) $\sum_{e \in E(T)} (d_e/n)$ is minimized.

مسئله درخت اشتاینر

- الگوریتم تقریبی برمن (برای درخت اشتاینر):
- T = یک درخت فراگیر بین راس‌های ترمینال
- به ازای هر زیرمجموعه k_عضوی از ترمینال‌ها K
- به ازای هر توپولوژی درخت t روی این مجموعه
- tau: بهترین درخت با توپولوژی t روی K:
- آیا اضافه کردن این درخت کمکی می‌کند؟
- درخت را به پشته اضافه کن و T را به‌روز کن
- درخت‌های اضافه شده از انتها را یکی یکی به T اضافه کن (اگر هنوز می‌توانستیم)

الگوریتم تقریبی مناسب برای ما

گراف عظیم!

MAXIMUM PARSIMONY

Input: A set S of m binary sequences, each of length n.

Goal: Find a tree T with m leaves and a labelling $\lambda : V(T) \to \{0,1\}^n$ of the vertices such that

1) The m labels of the leaves are exactly the sequences from S, and
2) $\sum_{e \in E(T)} (d_e/n)$ is minimized.

مسئله درخت اشتاینر

راحت

- الگوریتم تقریبی برمن (برای درخت اشتاینر):
  - T = یک درخت فراگیر بین راس‌های ترمینال
  - به ازای هر زیرمجموعه k_عضوی از ترمینال‌ها K
    - به ازای هر توپولوژی درخت t روی این مجموعه
      - tau: بهترین درخت با توپولوژی t روی K:
        - آیا اضافه کردن این درخت کمکی می‌کند؟
          - درخت را به پشته اضافه کن و T را به‌روز کن
  - درخت‌های اضافه شده از انتها را یکی یکی به T اضافه کن (اگر هنوز می‌توانستیم)

الگوریتم تقریبی مناسب برای ما

گراف عظیم!

MAXIMUM PARSIMONY

Input: A set S of m binary sequences, each of length n.

Goal: Find a tree T with m leaves and a labelling $\lambda : V(T) \to \{0,1\}^n$ of the vertices such that

1) The m labels of the leaves are exactly the sequences from S, and
2) $\sum_{e \in E(T)} (d_e/n)$ is minimized.

مسئله درخت اشتاینر

راحت

- الگوریتم تقریبی برمن (برای درخت اشتاینر):
- T = یک درخت فراگیر بین راس‌های ترمینال
- به ازای هر زیرمجموعه k_عضوی از ترمینال‌ها K

راحت

- به ازای هر توپولوژی درخت t روی این مجموعه
- tau: بهترین درخت با توپولوژی t روی K:
- آیا اضافه کردن این درخت کمکی می‌کند؟
- درخت را به پشته اضافه کن و T را به‌روز کن
- درخت‌های اضافه شده از انتها را یکی یکی به T اضافه کن (اگر هنوز می‌توانستیم)

الگوریتم تقریبی مناسب برای ما

گراف عظیم!

MAXIMUM PARSIMONY

Input: A set S of m binary sequences, each of length n.

Goal: Find a tree T with m leaves and a labelling $\lambda : V(T) \to \{0,1\}^n$ of the vertices such that

1) The m labels of the leaves are exactly the sequences from S, and
2) $\sum_{e \in E(T)} (d_e/n)$ is minimized.

مسئله درخت اشتاینر

- الگوریتم تقریبی برمن (برای درخت اشتاینر):
- T = یک درخت فراگیر بین راس‌های ترمینال
- به ازای هر زیرمجموعه k_عضوی از ترمینال‌ها K
- به ازای هر توپولوژی درخت t روی این مجموعه
- tau: بهترین درخت با توپولوژی t روی K:
- آیا اضافه کردن این درخت کمکی می‌کند؟
- درخت را به پشته اضافه کن و T را به‌روز کن
- درخت‌های اضافه شده از انتها را یکی یکی به T اضافه کن (اگر هنوز می‌توانستیم)

راحت

راحت

ANCESTRAL MAXIMUM LIKELIHOOD VERSION II

ANCESTRAL MAXIMUM LIKELIHOOD (VERSION II)

Input: A set S of m binary sequences, each of length n.

Goal: Find a tree T with m leaves and a labelling $\lambda : V(T) \to \{0,1\}^n$ of the vertices such that

1) The m labels of the leaves are exactly the sequences from S, and
2) $\sum_{e \in E(T)} H\left(d_e/n\right)$ is minimized.

$$-p\log_2(p)-(1-p)\log_2(1-p)$$

مسئله درخت اشتاینر

MAXIMUM PARSIMONY

Input: A set S of m binary sequences, each of length n.

Goal: Find a tree T with m leaves and a labelling $\lambda : V(T) \to \{0,1\}^n$ of the vertices such that

1) The m labels of the leaves are exactly the sequences from S, and
2) $\sum_{e \in E(T)} \left(d_e/n\right)$ is minimized.

مسئله درخت اشتاینر

ANCESTRAL MAXIMUM LIKELIHOOD VERSION II

ANCESTRAL MAXIMUM LIKELIHOOD (VERSION II)

Input: A set S of m binary sequences, each of length n.

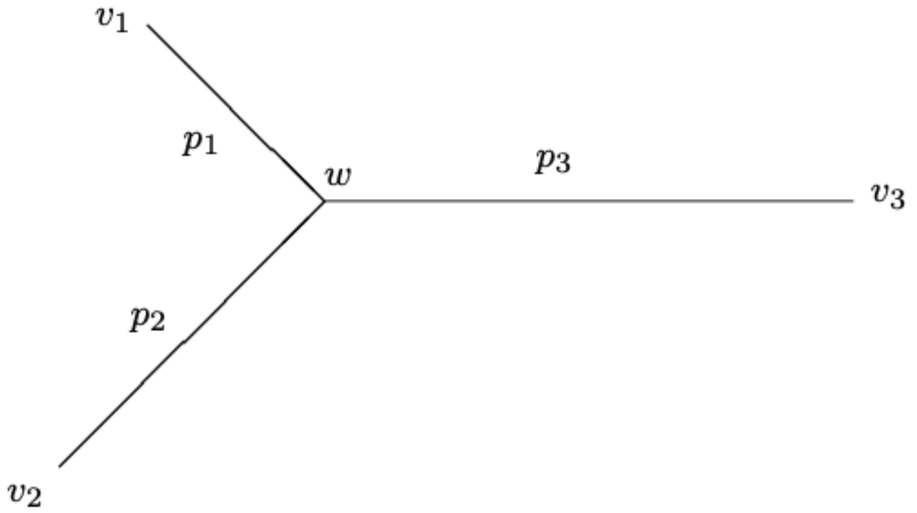
Goal: Find a tree T with m leaves and a labelling $\lambda : V(T) \to \{0,1\}^n$ of the vertices such that

1) The m labels of the leaves are exactly the sequences from S, and
2) $\sum_{e \in E(T)} H\left(d_e/n\right)$ is minimized.

$$-p\log_2(p)-(1-p)\log_2(1-p)$$

مسئله درخت اشتاینر

● برای k=۳ و ۴؟



دو افزودنی

- ۱ـ اگر ورودی‌ها «؟» داشته باشد؟
- ۲ـ اگر مدل احتمالاتی درخت‌ها مارکوف باشد

METHOD (SCELESTIAL ALGORITHM)

RESULTS

COMPARISON ON SIMULATED DATA

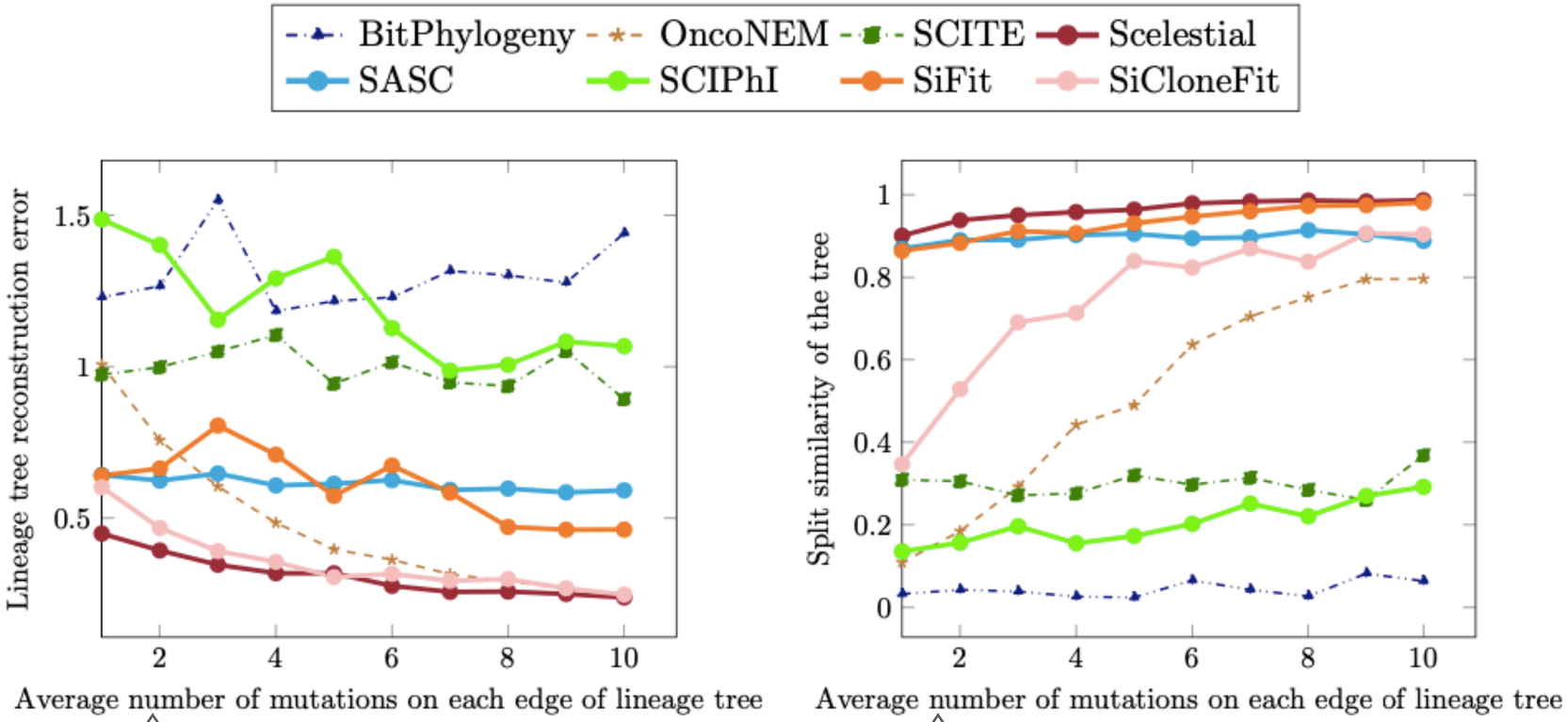
- **Data: OncoNEM's simulator** (false positive: 1.5%, false negatives 10%, missing value rate 7%)

	50 samples				100 samples			
	5 clones		10 clones		5 clones		10 clones	
Method	20 sites	50 sites	20 sites	50 sites	20 sites	50 sites	20 sites	50 sites
OncoNEM	0.93	0.97	0.80	0.76	0.96	0.96	0.84	0.78
Scelestial	**0.84**	**0.86**	**0.73**	**0.68**	0.89	**0.88**	**0.76**	**0.71**
BitPhylogeny	0.96	0.95	0.94	1.03	0.97	1.05	0.94	0.97
SCITE	1.00	0.97	0.93	0.89	0.99	0.96	0.91	0.98
SASC	0.88	0.91	0.80	0.78	0.93	0.92	0.82	0.80
SCIPhI	0.94	1.00	0.92	0.90	0.96	1.01	0.96	0.90
SiFit	0.88	0.90	0.87	0.82	**0.86**	0.91	0.84	0.79
SiCloneFit	0.96	1.01	0.83	0.79	0.99	1.05	0.90	0.81

- Average of 10 simulations.
- Distances between pairs of samples, compared between the ground truth and the tree (the less the better)

COMPARISON ON SIMULATED DATA

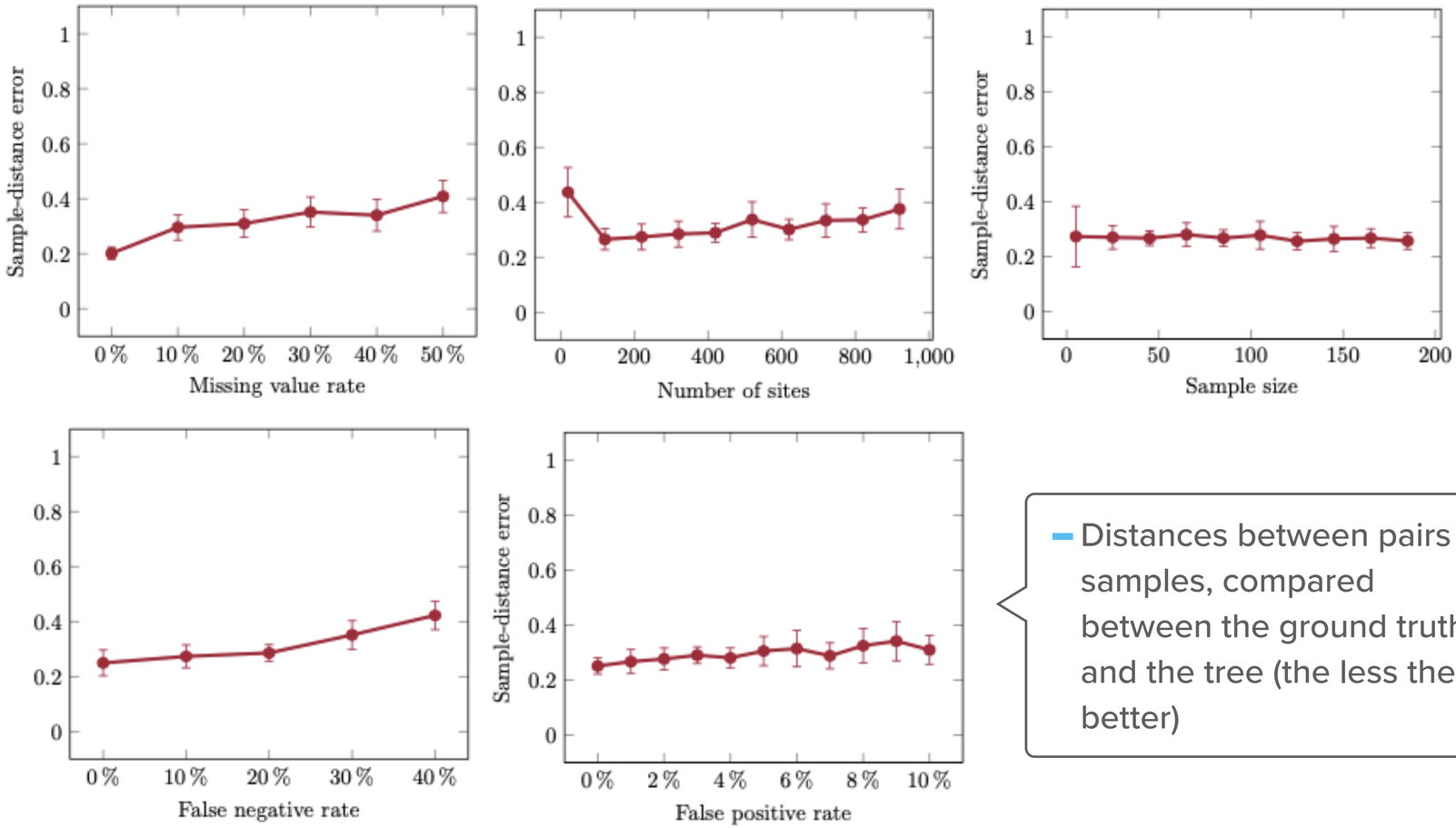
- On simulated data: (false positive: 1.5%, false negatives 10%, missing value rate 7%)



- Average of 10 simulations.
- Distances between pairs of samples, compared between the ground truth and the tree (the less the better)

- Average of 10 simulations.
- Similarity between partitions generated by the ground truth and the tree (the more the better)

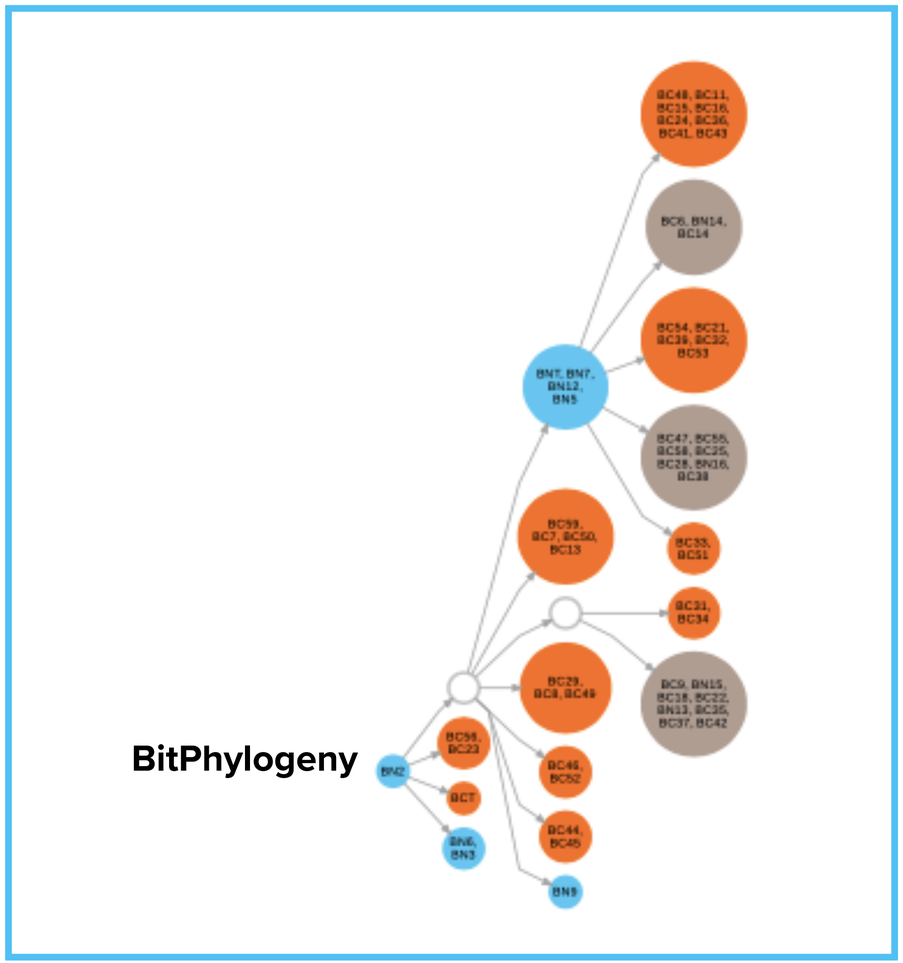
ROBUSTNESS

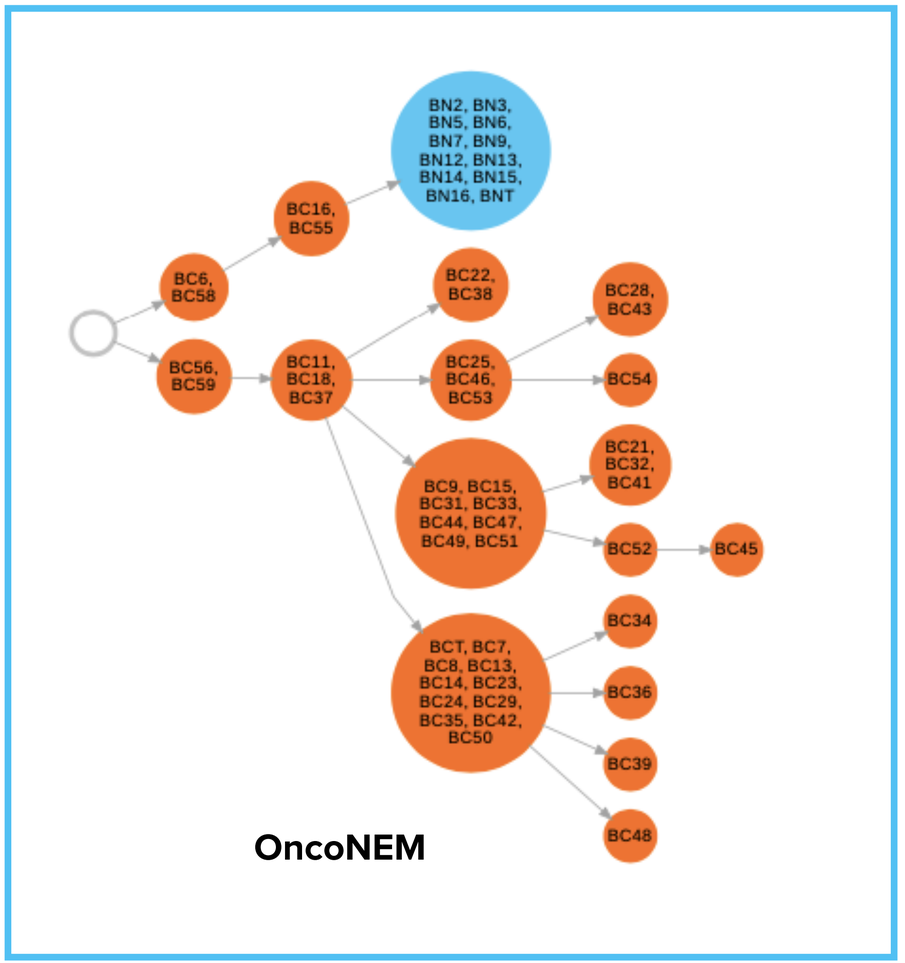


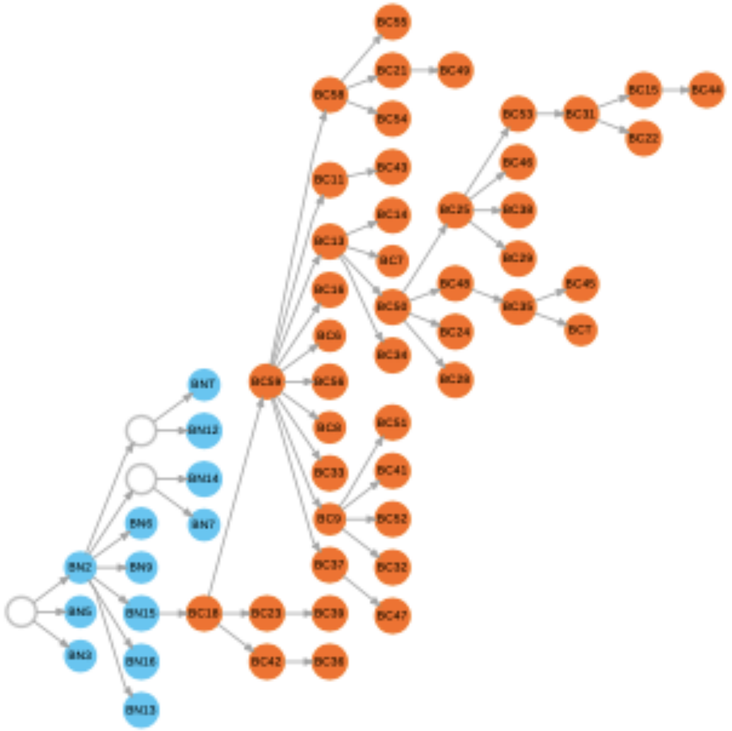
- Distances between pairs of samples, compared between the ground truth and the tree (the less the better)

ON REAL SINGLE-CELL DATASETS

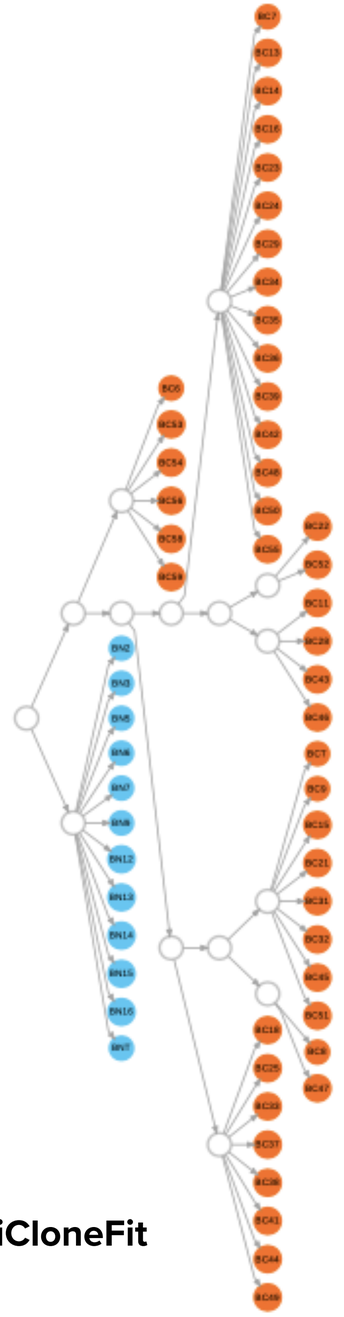
- Single-cell genomic data from muscle-invasive bladder tumor







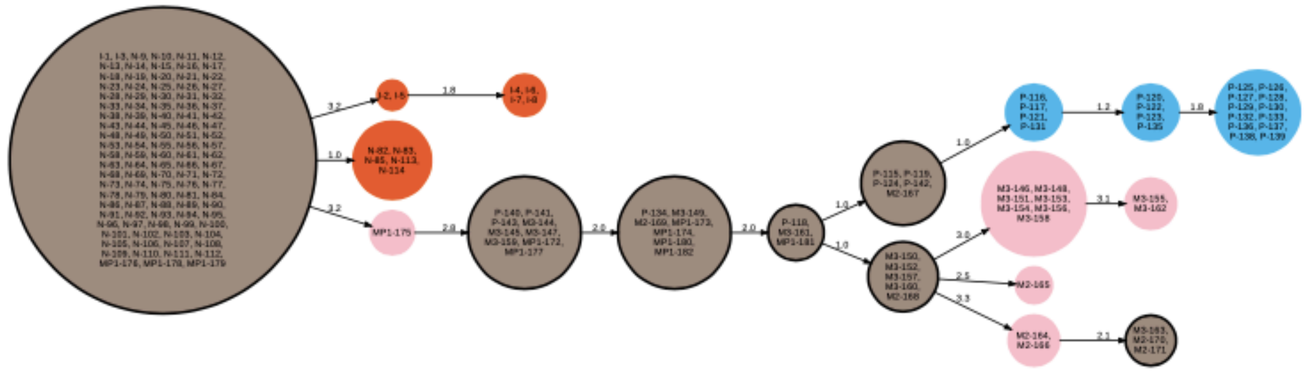
(c) Scelestial



SiCloneFit

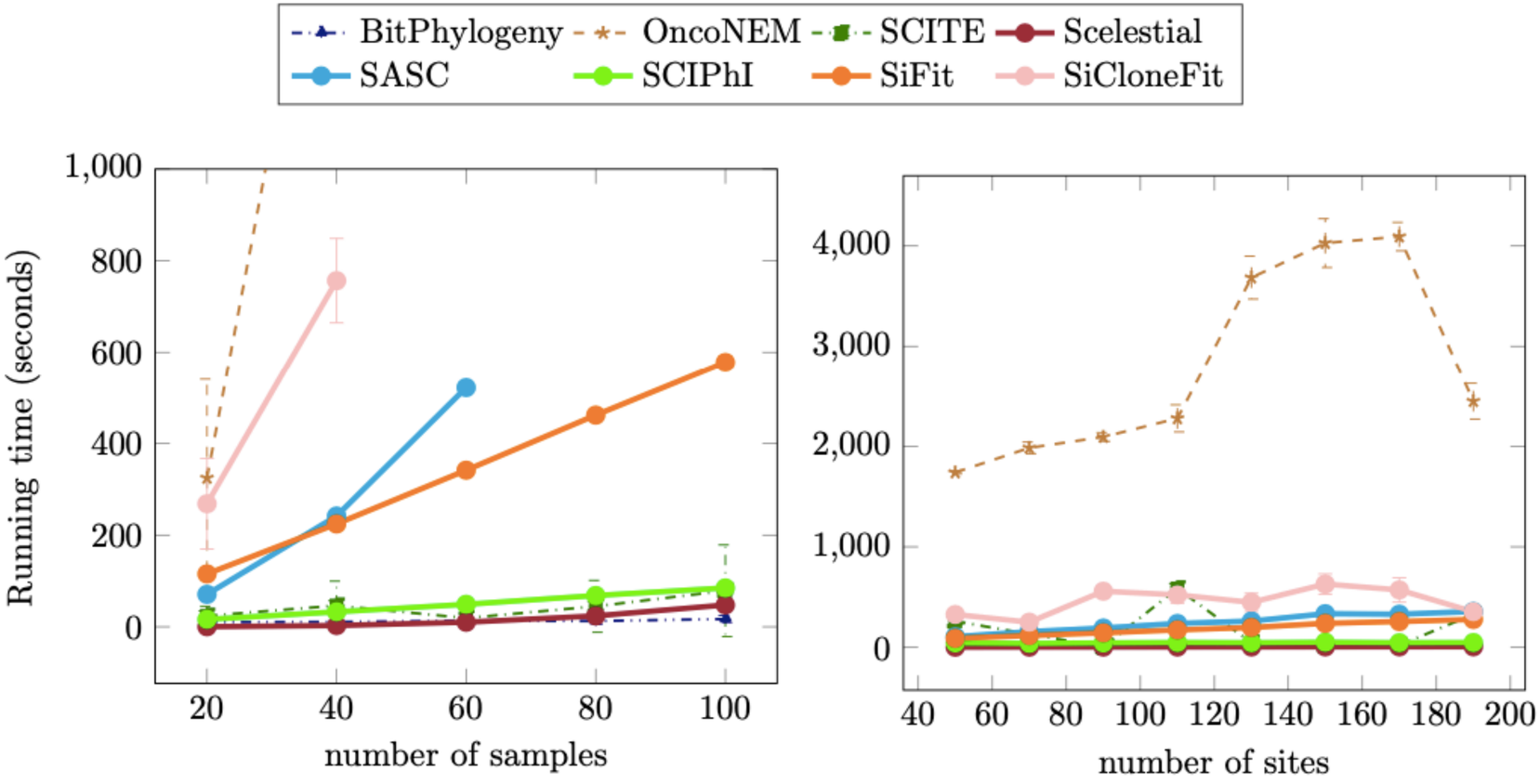
THREE REAL SINGLE-CELL DATASETS

- Scelestial and SiCloneFit perform the best
  - Criteria: Misplacing cancer cells and normal cells
  - Scelestial and SiCloneFit: 3 misplaced cancer cells
  - Others:



(g) OncoNEM

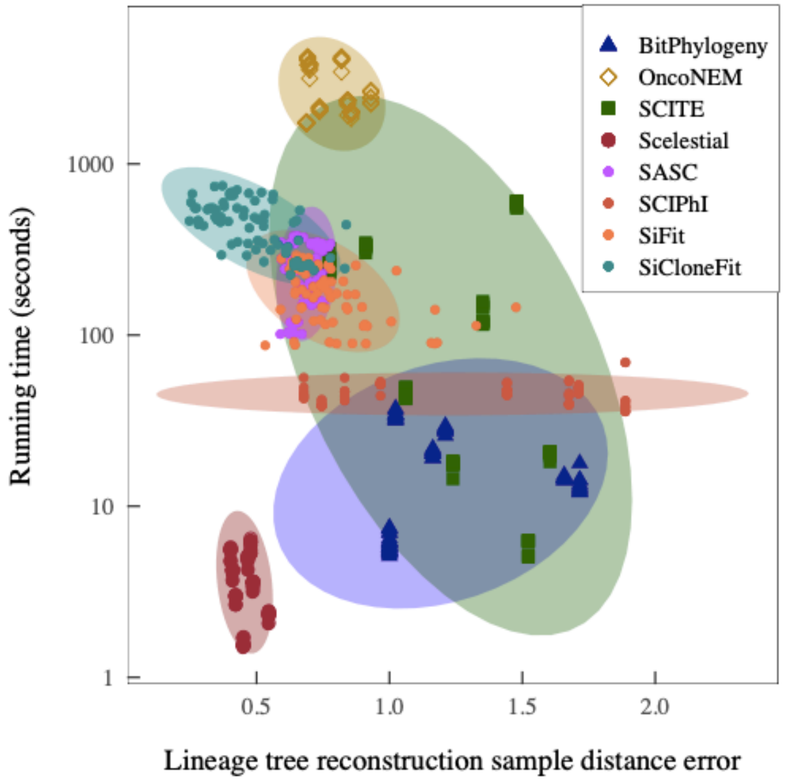
RUNNING TIME PERFORMANCE



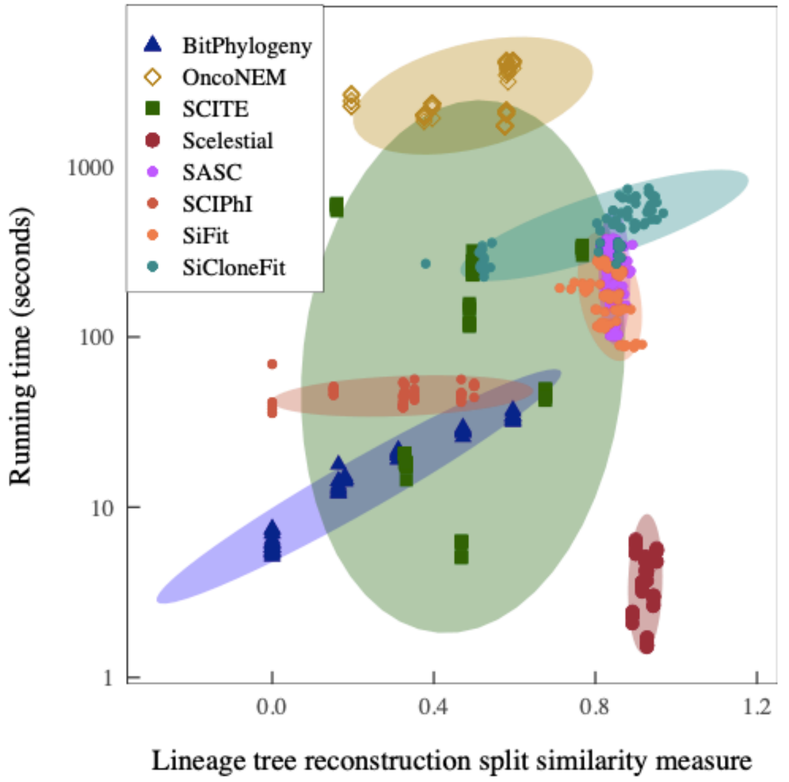
(a) Run time comparisons for varying sample sizes.

(b) Relation of run time to the number of sites

ERROR AND RUNNING TIME



(a) Run time and lineage reconstruction error of four lineage tree inference methods on datasets with different numbers of sites with respect to the pair distance error.



(b) Run times and lineage reconstruction errors of four methods on datasets with different numbers of sites with respect to the split similarity measure.

CONCLUSION

CONCLUSION

- We designed and implemented Scelestial
  - Single-cell phylogeny reconstruction method
  - Based on Steiner tree approximation algorithm
  - Supports missing values

CONCLUSION

- We designed and implemented Scelestial
  - Single-cell phylogeny reconstruction method
  - Based on Steiner tree approximation algorithm
  - Supports missing values
- Evaluation (in comparison with other methods)
  - Scelestial performs best in phylogeny reconstruction on simulated data
  - Scelestial and SiCloneFit perform best on real data-sets
  - Scelestial is fastest

CONCLUSION

- We designed and implemented Scelestial
  - Single-cell phylogeny reconstruction method
  - Based on Steiner tree approximation algorithm
  - Supports missing values
- Evaluation (in comparison with other methods)
  - Scelestial performs best in phylogeny reconstruction on simulated data
  - Scelestial and SiCloneFit perform best on real data-sets
  - Scelestial is fastest
- Available as
  - RScelestial (an R package)
  - C++ codes (not public yet)

REFERENCES

- [Alon08] Alon N, Chor B, Pardi F, Rapoport A. Approximate maximum par- simony and ancestral maximum likelihood. IEEE/ACM Transactions on Computational Biology and Bioinformatics. 2008;7(1):183–187.
- [Berman04] Berman P, Ramaiyer V. Improved approximations for the Steiner tree problem. Journal of Algorithms. 1994;17(3):381–408.
- Yuan K, Sakoparnig T, Markowetz F, Beerenwinkel N. BitPhylogeny: a probabilistic framework for reconstructing intra-tumor phylogenies. Genome Biology. 2015;16(1):36.
- Ross EM, Markowetz F. OncoNEM: inferring tumor evolution from single-cell sequencing data. Genome Biology. 2016;17(1):69.
- Jahn K, Kuipers J, Beerenwinkel N. Tree inference for single-cell data. Genome Biology. 2016;17(1):86.
- Zafar H, Tzen A, Navin N, Chen K, Nakhleh L. SiFit: inferring tumor trees from single-cell sequencing data under finite-sites models. Genome Biology. 2017;18(1):178.
- Zafar H, Navin N, Chen K, Nakhleh L. SiCloneFit: Bayesian in- ference of population structure, genotype, and phylogeny of tumor clones from single-cell genome sequencing data. Genome Research. 2019;29(11):1847–1859.

REPORT 28 JAN 2021

APPROXIMATION ALGORITHM FOR PHYLOGEOGRAPHY PROBLEM

INTRODUCTION: MOTIVATION

ALGORITHMIC PERSPECTIVE OF PHYLOGENY INFERENCE

- A lot of heuristics
  - NJ, RaXML, Fastree, …
- Can we have a guarantee?
  - An approximation algorithm (Alon, et al)
  - Inapproximability result (???)

AN APPROXIMATION ALGORITHM FOR PHYLOGENY INFERENCE

- Reduction of the phylogeny problem to Steiner tree
  - On a huge graph
  - With strange weights
- Using established Steiner tree approximation algorithm
  - With approximation ratio 1.7
-

MOTIVATION

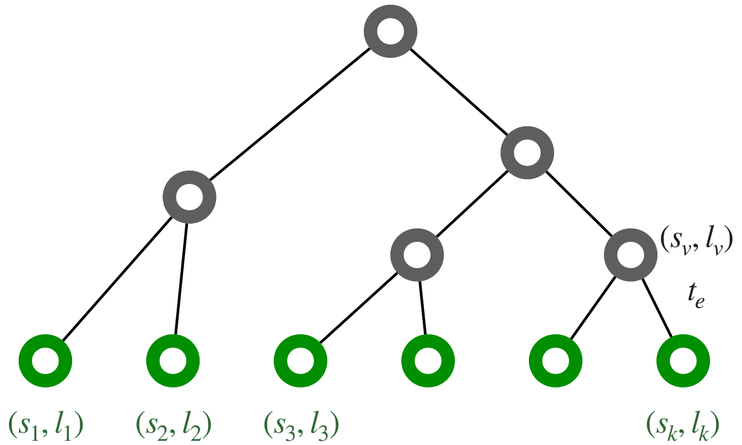
- Main **limitation** of Alon, et al's work
  - Only for two-state (wild/mutated)
  - Not applicable on geography
- Phylogeography
  - JC69, …
  - Our pas work "Scelestial": A heuristic on the shoulder of a guaranteed algorithm —> a new one for phylogeography

PROBLEM DEFINITION (REVISED)

PROBLEM STATEMENT

- **Phylogeography** inference problem:
  - Input: a set of sequences + geolocation
  - Output: A tree annotated with sequences + geolocations
  - Objective function: maximum likelihood, or minimum parsimony
    - A model of evolution
    - A model of transportations

OUTPUT



OBJECTIVE

Probability of root

Prob. of sequence of root

Prob. of location of root

For each edge of tree $v \to u$

$$\max_{T,t,s,l} \mathbb{P}[s_{\text{root}(T)}]\mathbb{P}[l_{\text{root}(T)}] \cdot \prod_{v \to u} \mathbb{P}[s_u \mid s_v] \cdot \mathbb{P}[l_u \mid l_v]$$

Prob. of $s_v \to s_u$

Prob. of $l_v \to l_u$

Probability of edge $v \to u$

Find best

T: tree

t_e: duration for edge

s_v: sequence for internal nodes

l_v: geolocation for internal nodes

GENERAL ISSUES

- Old methods do not work!
  - Even for geolocation only case (without sequences)
- Tree is directed
  - Very hard

OUR CONTRIBUTION

OUR ALGORITHM

BASIC OPERATION

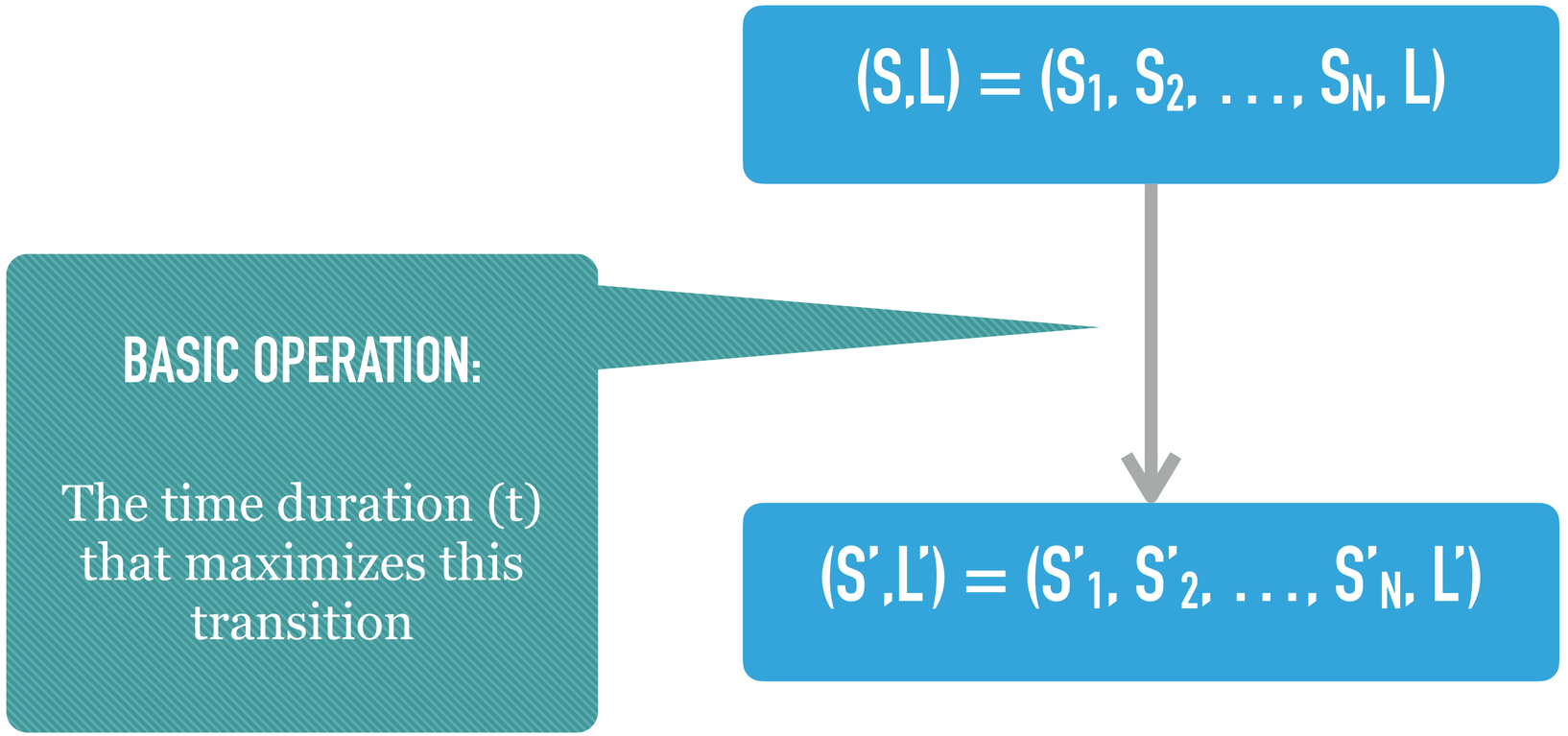
Algorithm 1 Method for phylogeny reconstruction of Markov reversible model of sequences

Require: $S = \{s_i\}$: a set of k sequences of length n

1: **function** $c'(v, u)$ **return** $w(v, u) - w(u)$

2: $T \leftarrow \{(L_i := \{s_i\}, E_i := \emptyset)\}_i$ for $i = 1, \ldots, k$.
3: **while** $|T| \geq 2$ **do**
4: $A[i, j] \leftarrow c'(L_i[1], L_j[1])$ for all $1 \leq i, j \leq \lfloor |T|/2 \rfloor$
5: $M \leftarrow$ Minimum perfect matching of A
6: **for** $(i, j) \in M$ **do**
7: $T \leftarrow T + (L_i \cup L_j, E_i \cup E_j + \{L_i[1], L_j[1]\}) - (L_i, E_i) - (L_j, E_j)$
8: $(L, E) \leftarrow$ the only member of T
9: **return** Tree with edges E rooted arbitrarily

BASIC OPERATION



RESULT 1: PERFORMANCE OF OUR ALGORITHM

Theorem 7. *The method presented in Algorithm 1 with oracle access to $w(x, y)$ and $w(x)$ (see Definition 1) is a polynomial time $\lceil \log_2 k \rceil$-approximation algorithm for the reversible Markov phylogeny problem.*

Our result $\leq \lceil \log_2 k \rceil \cdot$ OPT

$\lceil \log_2 k \rceil$ -approximation algorithm for phylogeny inference

- For reversible markov models
- For phylogeography inference

RESULT 2: INAPPROXIMABILITY

Theorem 9. *The phylogeny problem on reversible Markov model with input of size k does not admit any $(\ln k - c \cdot (\ln \ln k)^2) \ln k$ factor approximation algorithm for some constant c, unless NP has quasi-polynomial time algorithms.*

Our result is almost the best

No (ln k)-approximation algorithm

IMPLEMENTING BASIC OPERATION FOR JC69 AND PHYLOGEOGRAPHY

- JC69
  - Numeric algorithms
- Phylogeography
  - Simulation-like algorithm
    - Fast for real-world graphs

CONCLUSION

- Algorithm with guarantee for “Phylogeography” and “Phylogeny” on reversible Markov models (e.g. JC69)
  - $\lceil \log_2 k \rceil$-approximation algorithm
  - No (ln k)-approximation algorithm
- Future:
  - Submitted to “IEEE/ACM Transactions on Computational Biology and Bioinformatics”
  - Other problems
  - Implementation (maybe)


سوال؟