Tiny steps towards addressing one part of the DNA rearrangement puzzle

Marie-France Sagot

Inria Rhône-Alpes – Projet HELIX

Laboratoire de Biométrie et Biologie Évolutive

Université Claude Bernard, Lyon I, France

Joint work with Estela Maris Rodrigues and Yoshiko Wakabayashi

Instituto de Matemática e Estatística (IME)

Universidade de São Paulo (USP), Brazil

Listen ! a frog

Jumping in the stillness

Of an ancient pond !

Basho (Spring days)

Listen ! a frog

Jumping in the stillness

Of an ancient pond !

Basho (Spring days)

Listen ! a piece of *E. coli*

Jumping in the stillness

Of my genome !

slight modification of
a poem by Basho

Our general concerns

How to identify, at a low and high level and with a fair degree of confidence and accuracy, the rearrangements that took place ?

How to use rearrangement events to calculate a distance between organisms ?

How to infer a phylogeny when rearrangements have taken place ?

Our concern in this talk

How to identify, at a low and high level and with a fair degree of confidence and accuracy, the rearrangements that took place ?

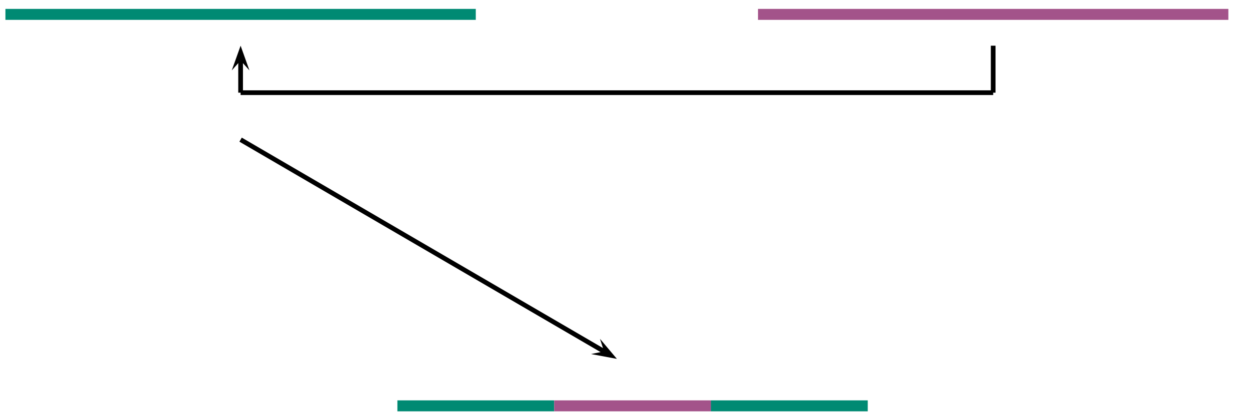
How to use rearrangement events to calculate a distance between organisms ?

How to infer a phylogeny when rearrangements have taken place ?

A rapid word on the rearrangements that will be considered (1)

Horizontal transfer

species or strain X_1 **species or strain X_2**



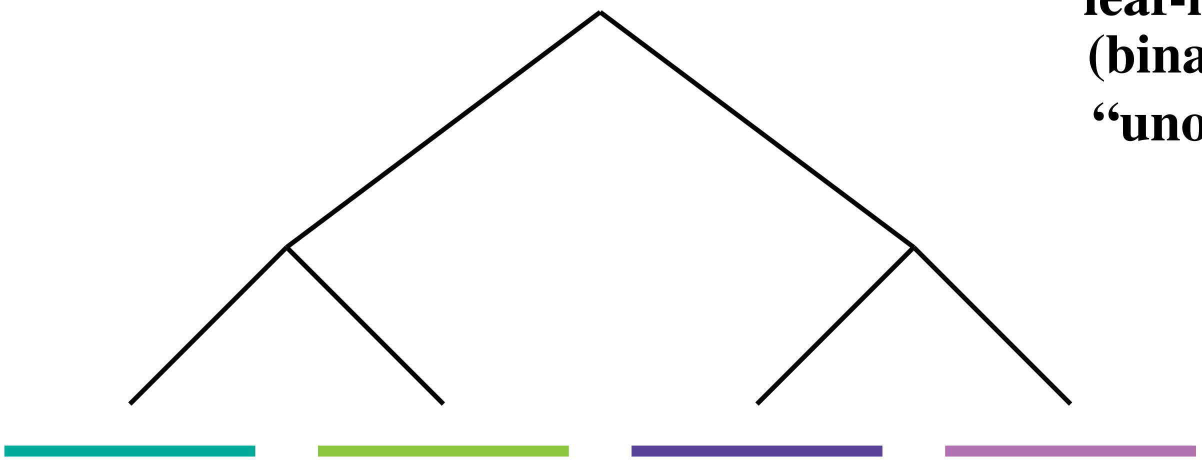
A rapid word on the rearrangements that will be considered (2)

Conversion

Trees representing the evolutionary history of a set of organisms

Data (here): for each organism, a set of characters (DNA sequence positions) over a discrete set of states (A,C,G,T)

(rooted)
leaf-labeled
(binary)
“unordered”



current organisms

Phylogenetic tree

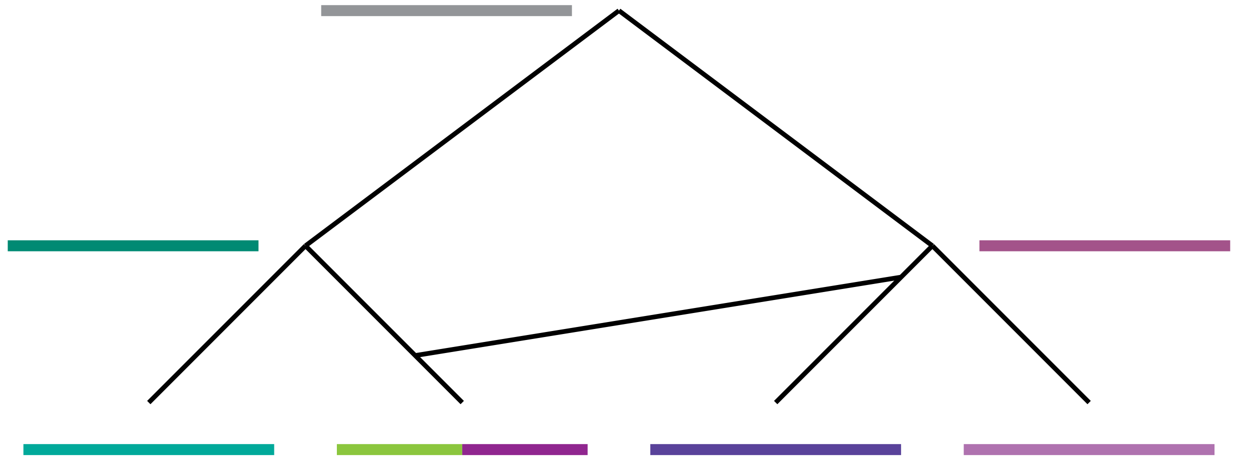
Find most parsimonious tree – “Small Phylogeny Problem”

* ancestors

(rooted)
leaf-labeled
(binary)
“unordered”

Effect of transfer on the history of the organisms

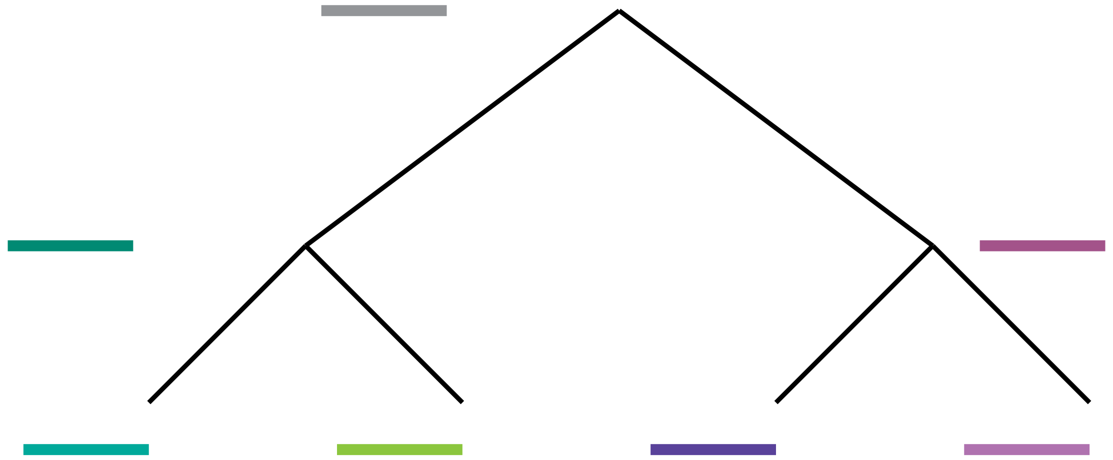
Not a single tree anymore



Effect of transfer on the history of the organisms

Not a single tree anymore but two ! (or more)

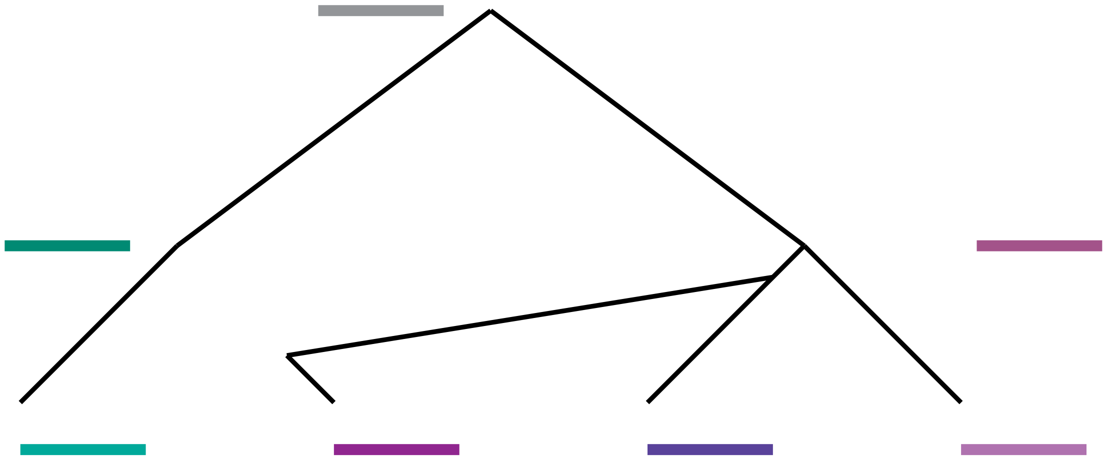
the first:



Effect of transfer on the history of the organisms

Not a single tree anymore but two ! (or more)

the second:



Genetic transfer detection and localization

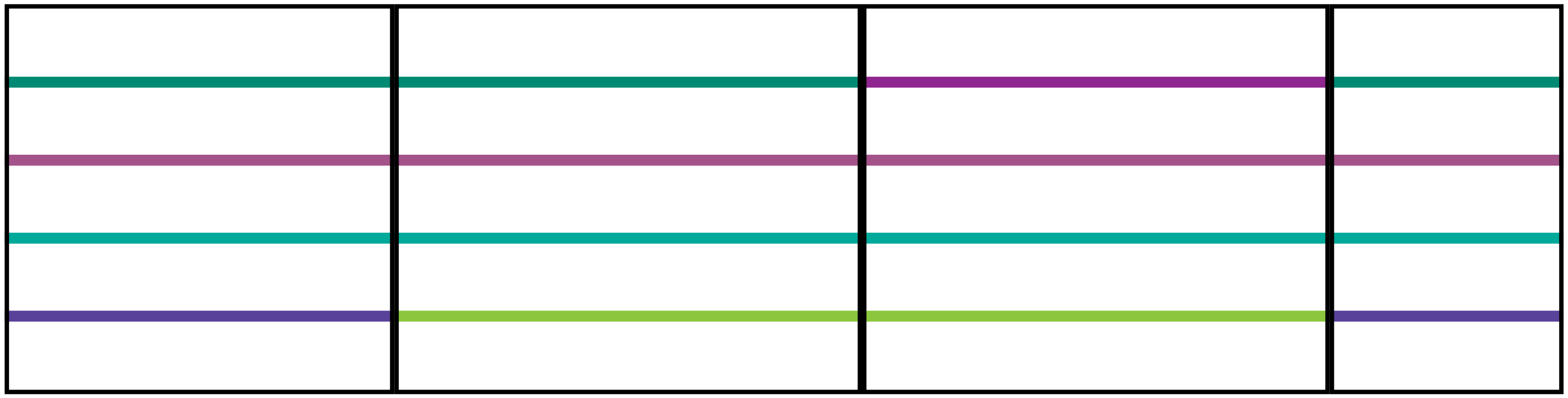
Data

a set of phylogenetically "correctly" aligned sequences from closely related organisms

Genetic transfer detection and localization as an optimisation problem

The (ideal ?) result

alignment



tree 1 tree 2 tree 3 tree 4

"optimal" partition: sum of lengths of trees + penalty for genetic transfers is MINIMUM

Genetic transfer detection and localization as an optimisation problem

The (ideal ?) result

alignment

"optimal" partition: sum of lengths of trees + penalty for genetic transfers is MINIMUM

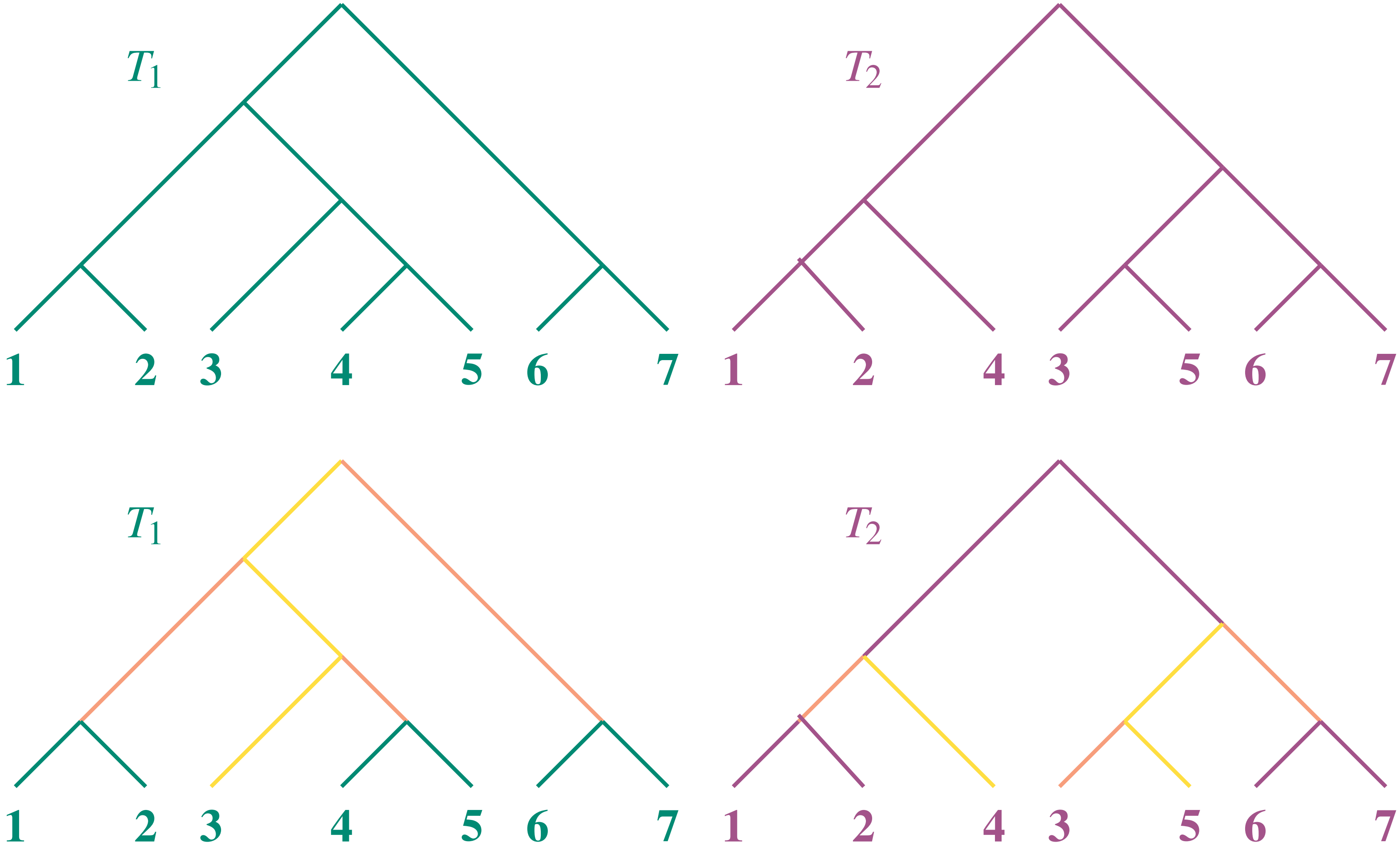
penalty should depend on genetic transfer distance between two trees

Genetic transfer distance between two phylogenetic trees

Maximum Agreement Forest of two trees

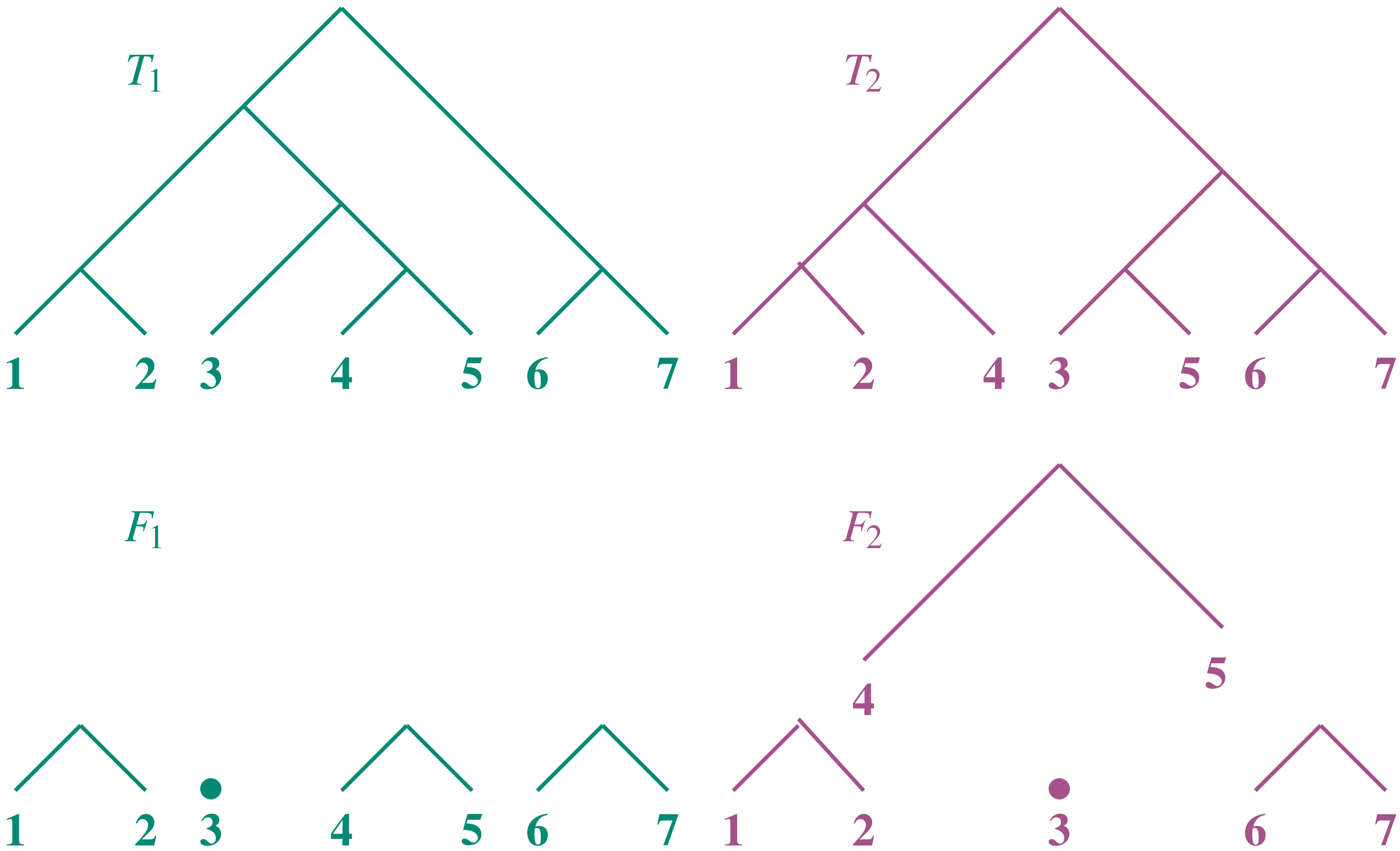
MAFALDA (E. M. Rodrigues *et al*, APPROX'01, LNCS, 2129:159-169)
(E. M. Rodrigues *et al*, The Maximum Agreement Forest Problem: Approximation Algorithms and Computational Experiments, submitted)

Agreement Forest (AF)



Elimination = Removal + Contraction

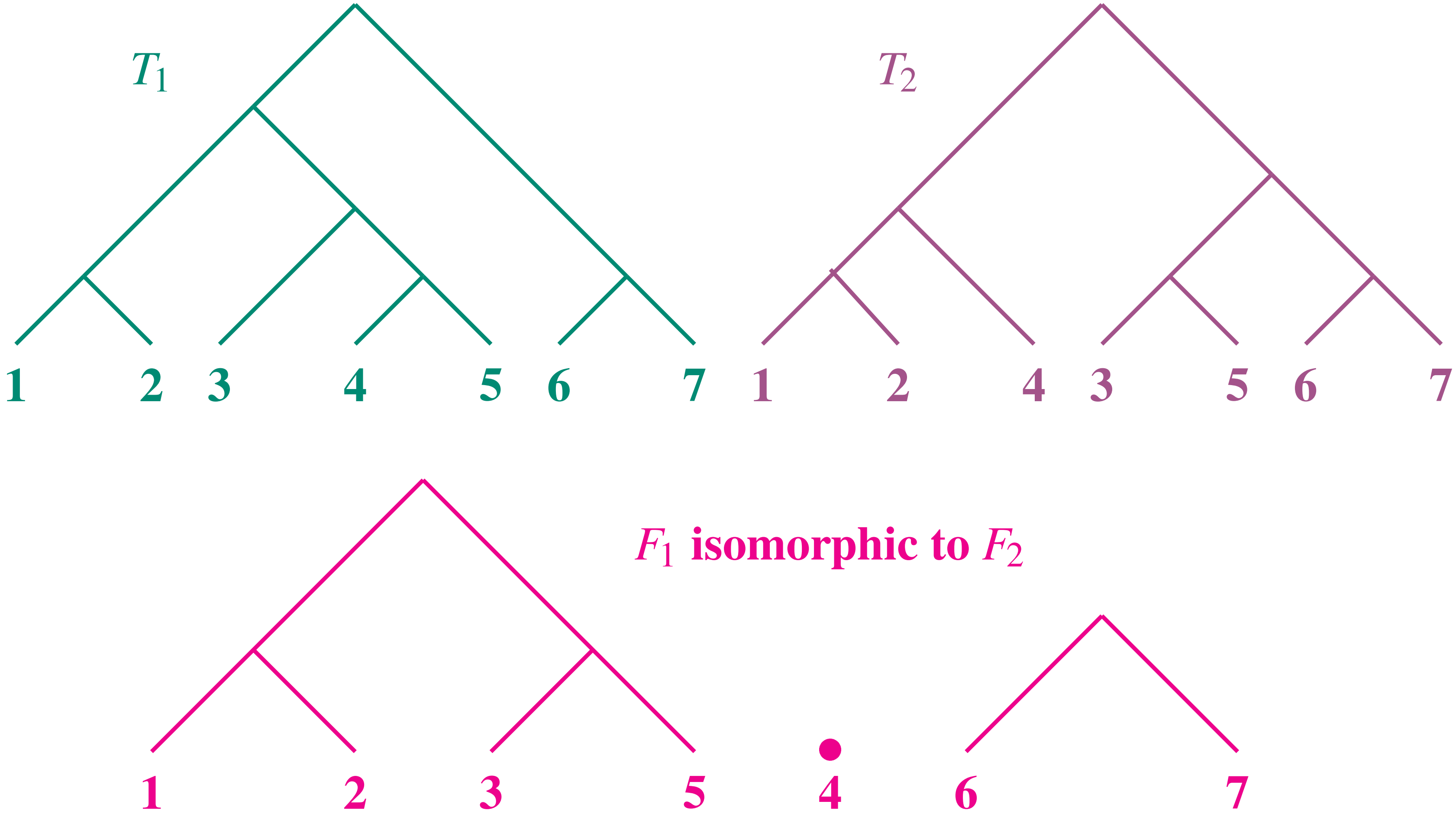
Agreement Forest (AF)



size (*i.e.* number of components) = 4

Maximum Agreement Forest (MAF)

(Maximum Common Minor)



size of MAF = 3

Maximum Agreement Forest problem for binary trees

Shown to be NP-Hard by Hein *et al.*, 1996

(Suggested to be MAX-SNP-Hard)

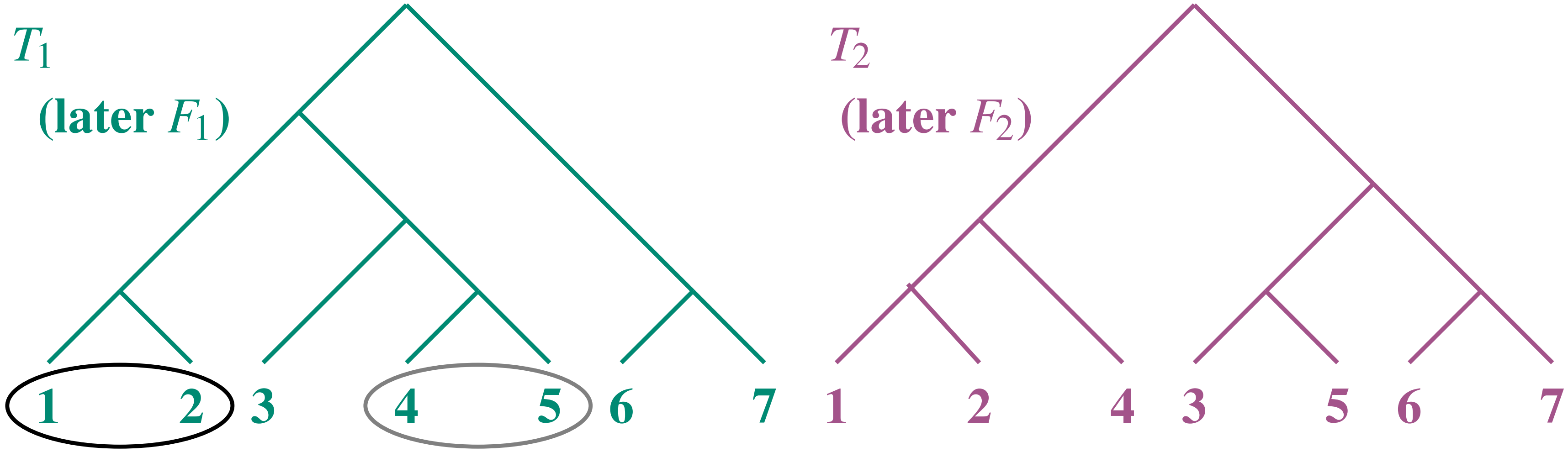
Who also provided a 3-approximation algorithm for the problem

Which, in fact, we showed is a 4-approximation

We introduced three algorithms all of which are 3-approximations

and extended the best to deal with trees of degree d (the ratio is $d+1$)

Our (so far) best algorithm for the MAF problem



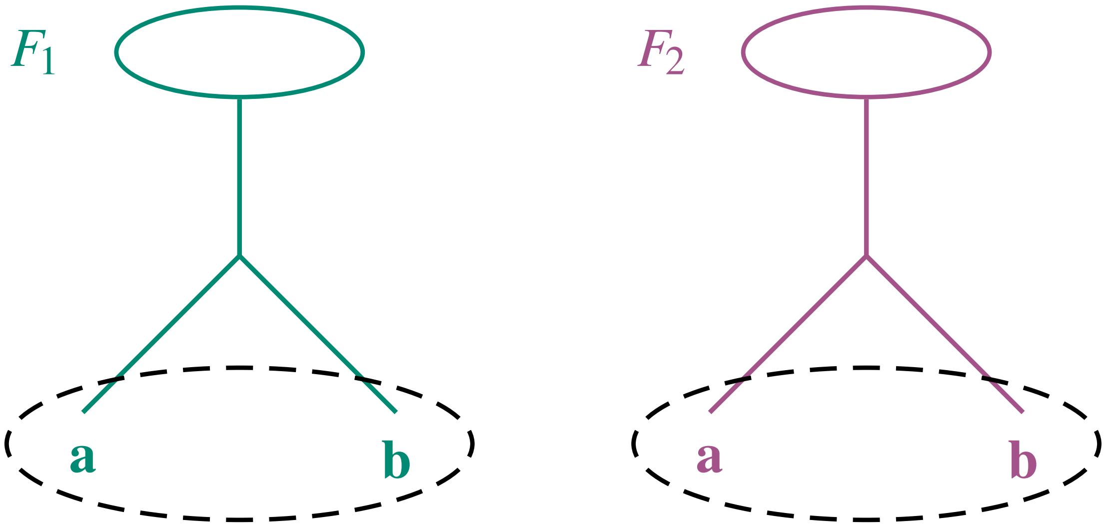
Search for pairs of sibling leaves in T_1 (F_1)

For each pair of sibling leaves in T_1 (F_1)
Consider the positions of the corresponding leaves in T_2 (F_2)

Various cases must be considered

The easiest cases

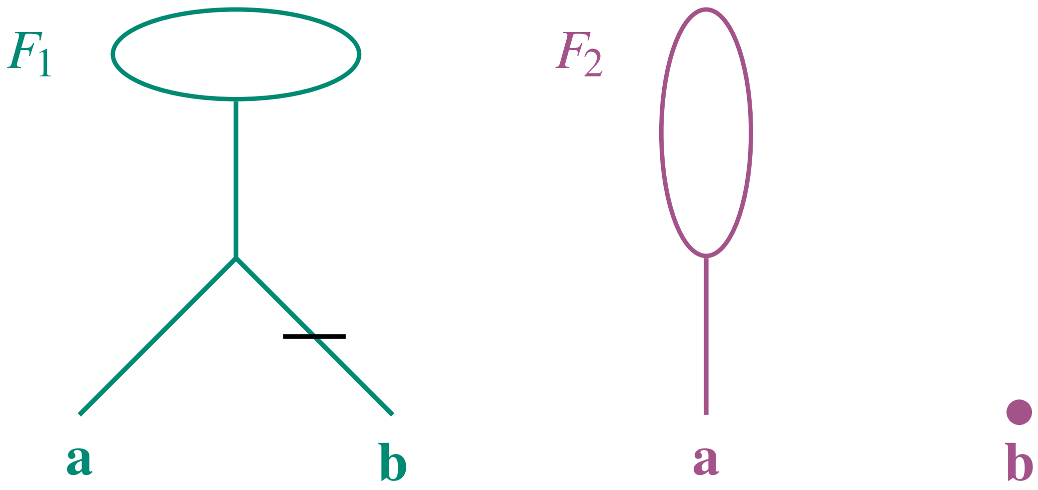
Case 1:



shrink

The easiest cases

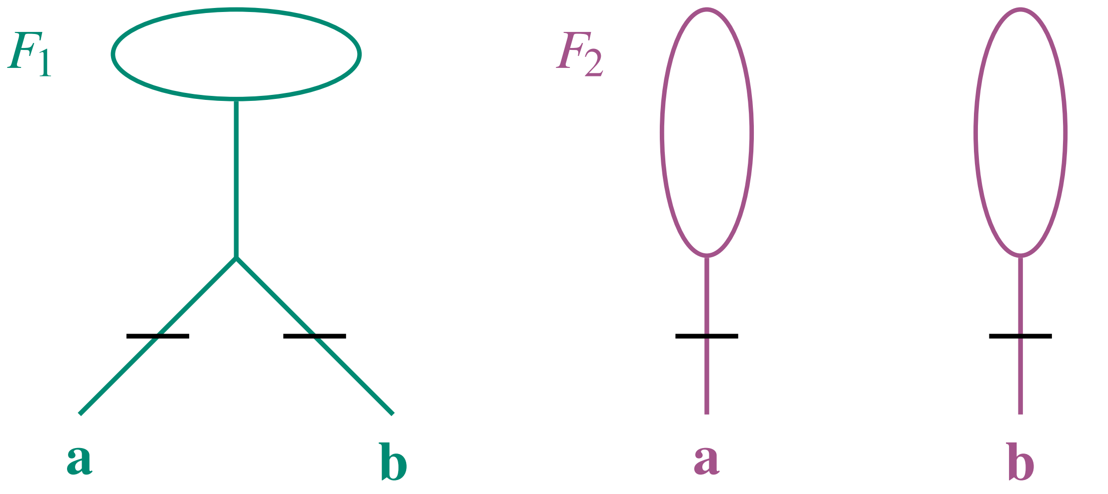
Case 2:



cut an edge in F_1 and contract

The easiest cases

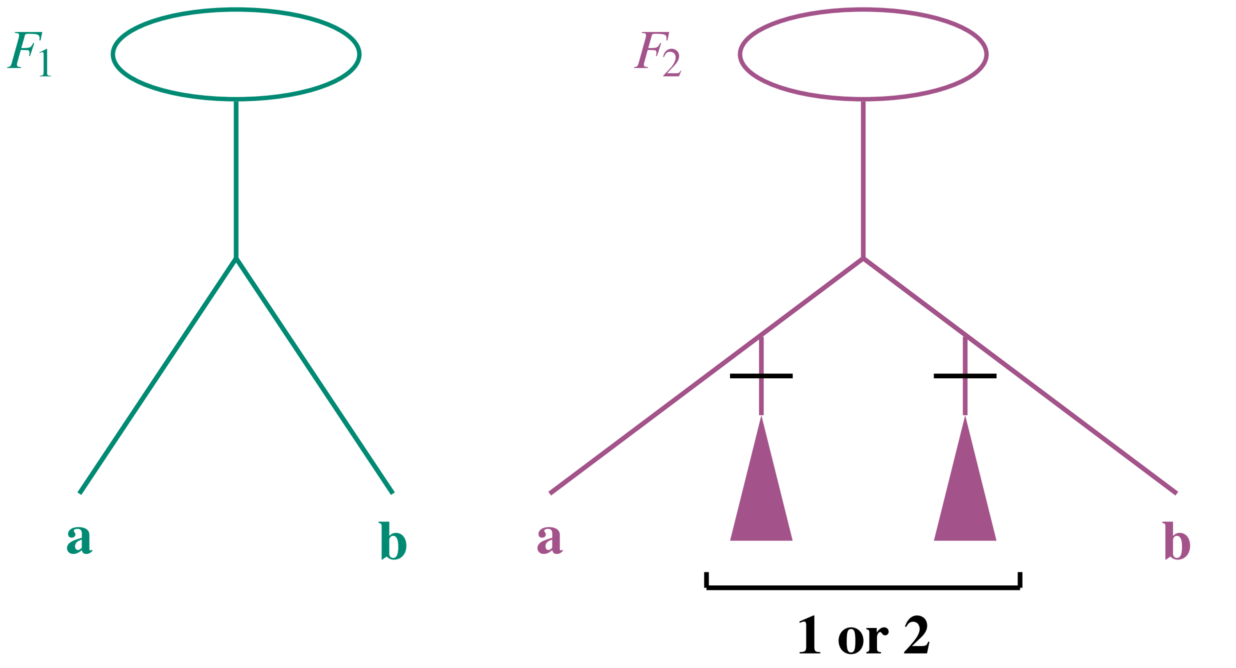
Case 3:



cut two edges each in F_1 and F_2 and contract in F_1

The two more delicate cases

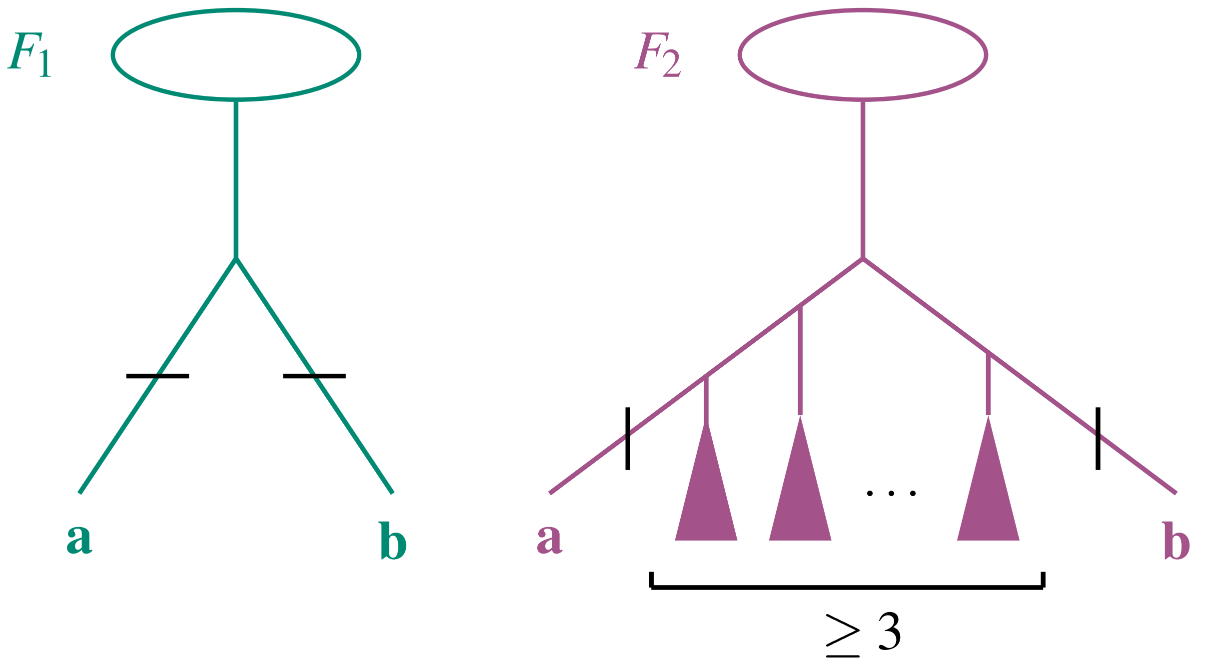
Case 4:



cut the subtrees hanging from the path from a to b in F_2

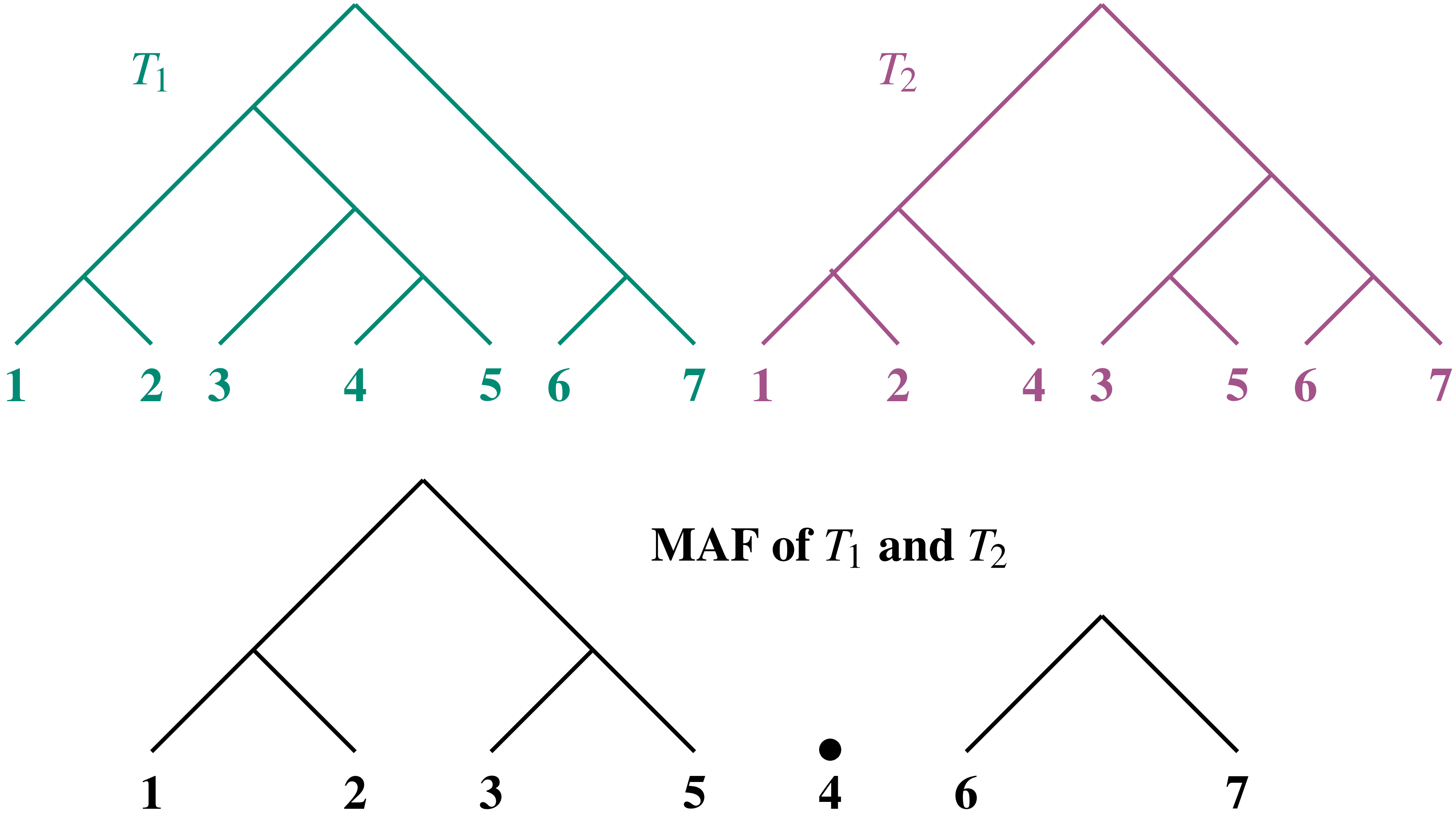
The two more delicate cases

Case 5:

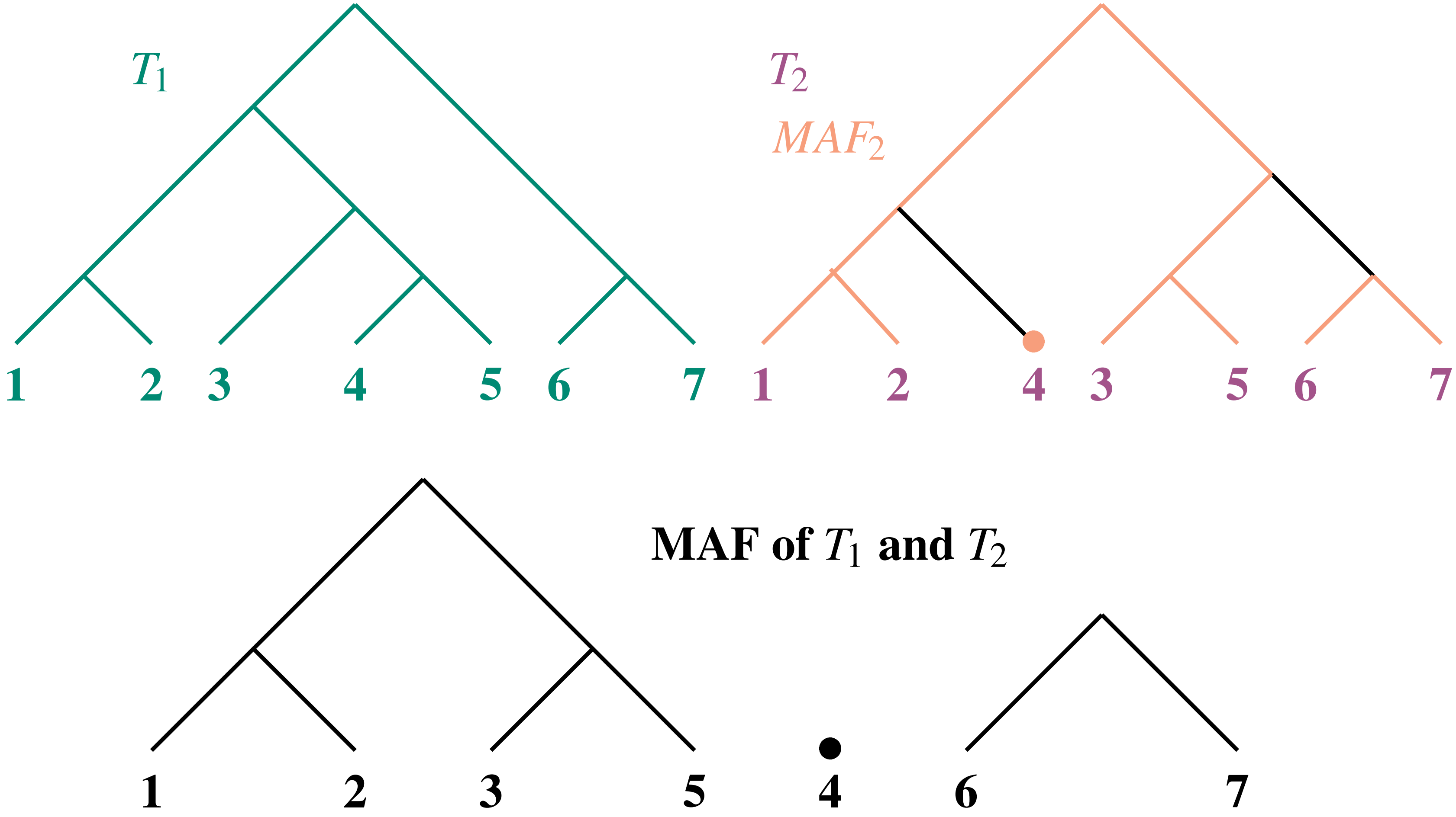


cut two edges each in F_1 and F_2
(those leading to the leaves labelled a and b)
and contract in F_1 and, possibly, F_2

Outline of the ratio analysis

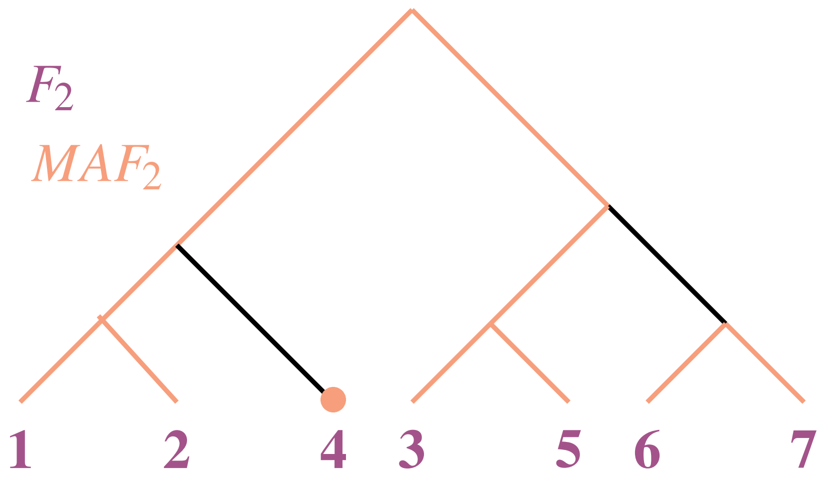


Outline of the ratio analysis



edges NOT belonging to MAF_2 are called links

Outline of the ratio analysis



approximation algorithm does "good" and "bad" eliminations

good elimination: elimination of a link

number of good eliminations is MAF-1

idea: prove at least one third of all eliminations are good

again various cases to analyse

Where is the main difficulty

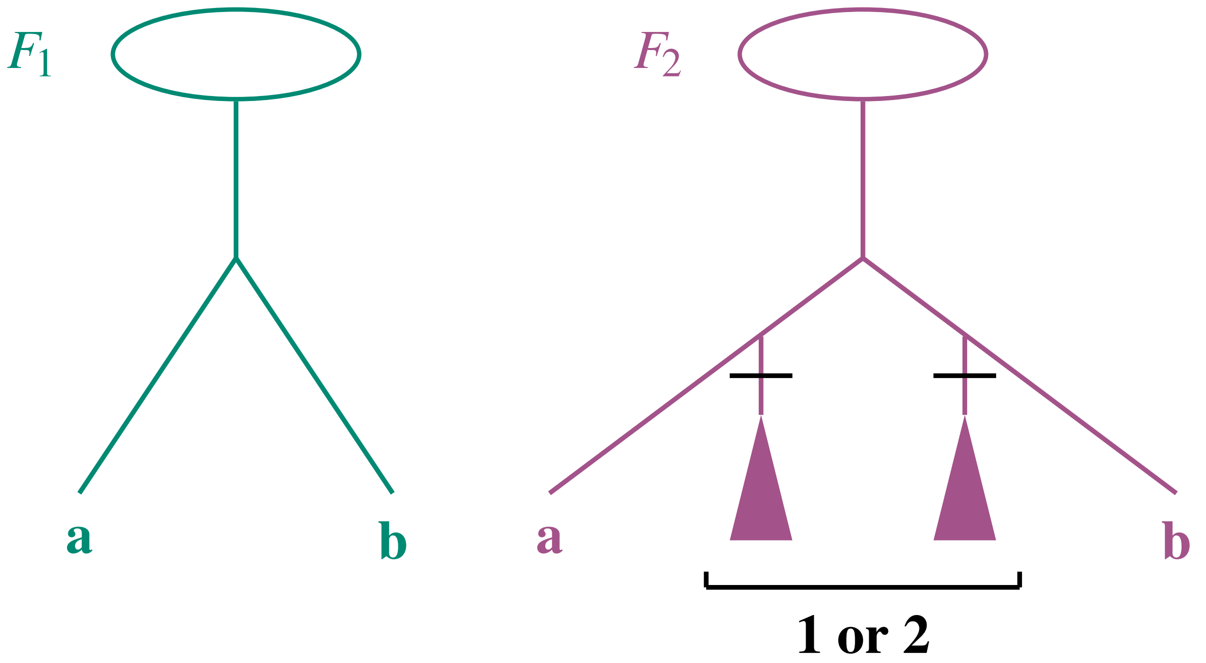
Bad eliminations must be compensated (possibly later only) with the good elimination of an edge

“belonging to the same case as led to the elimination”

because we do not know how many times each case will appear

The two more delicate cases

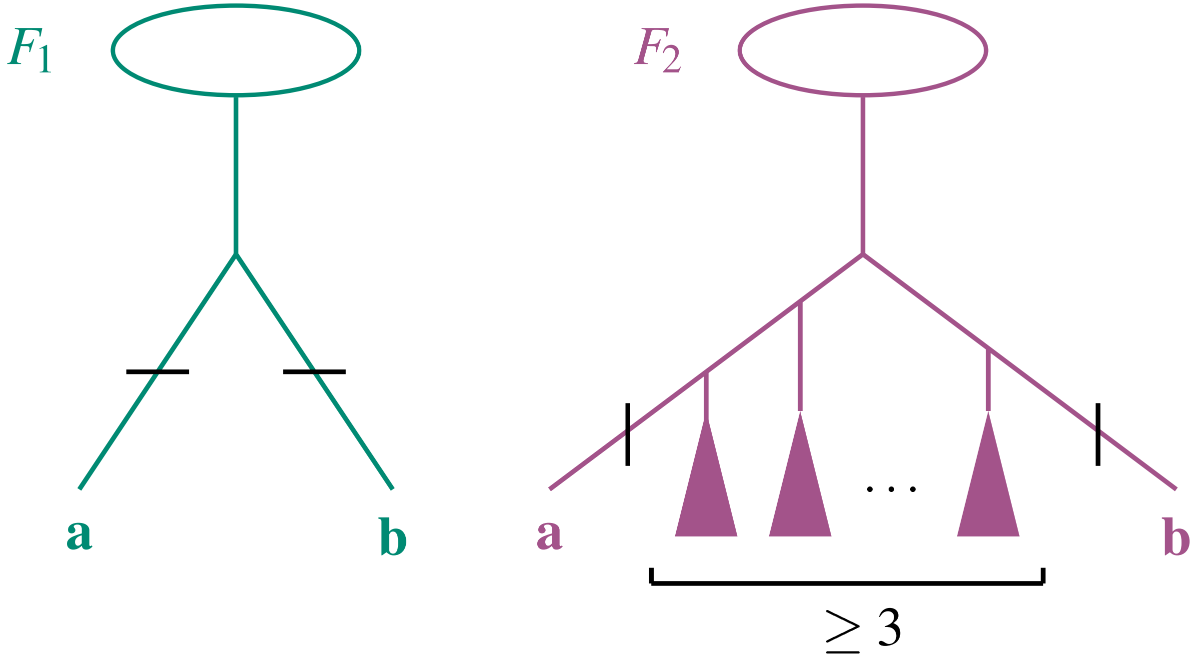
Case 4:



possibly both are bad eliminations but...

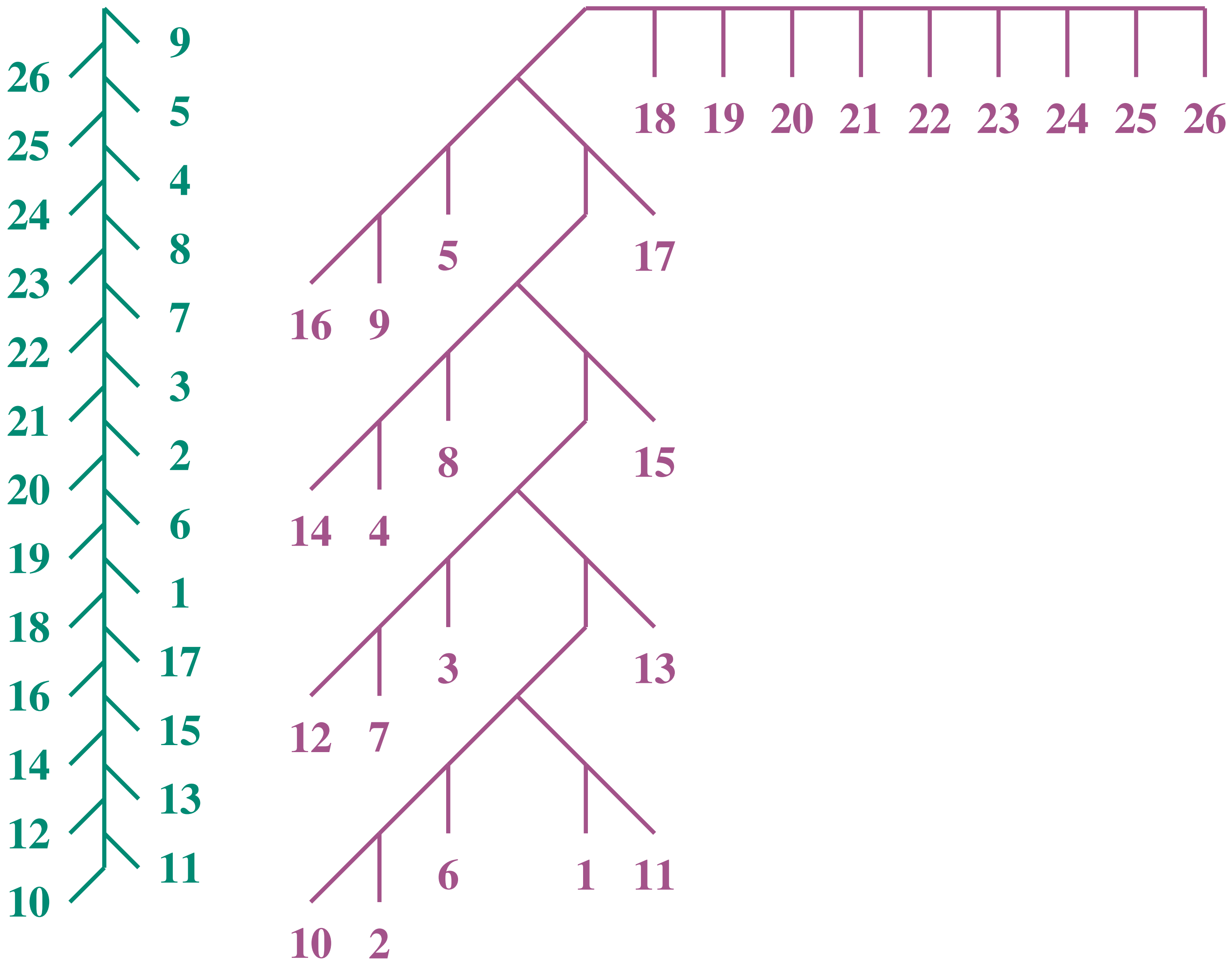
The two more delicate cases

Case 5:

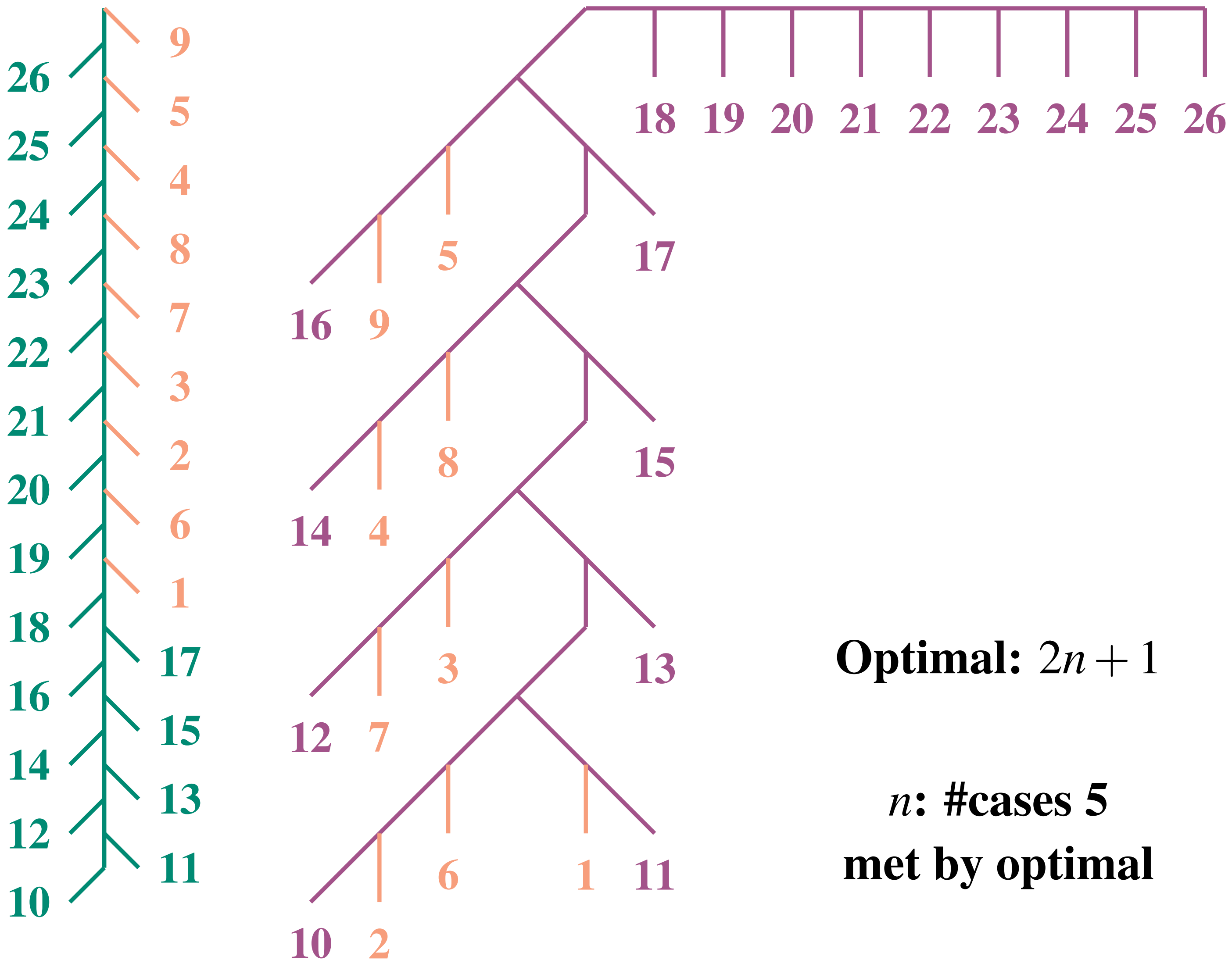


possibly both are bad eliminations but...

The ratio of 3 at which we arrive is tight



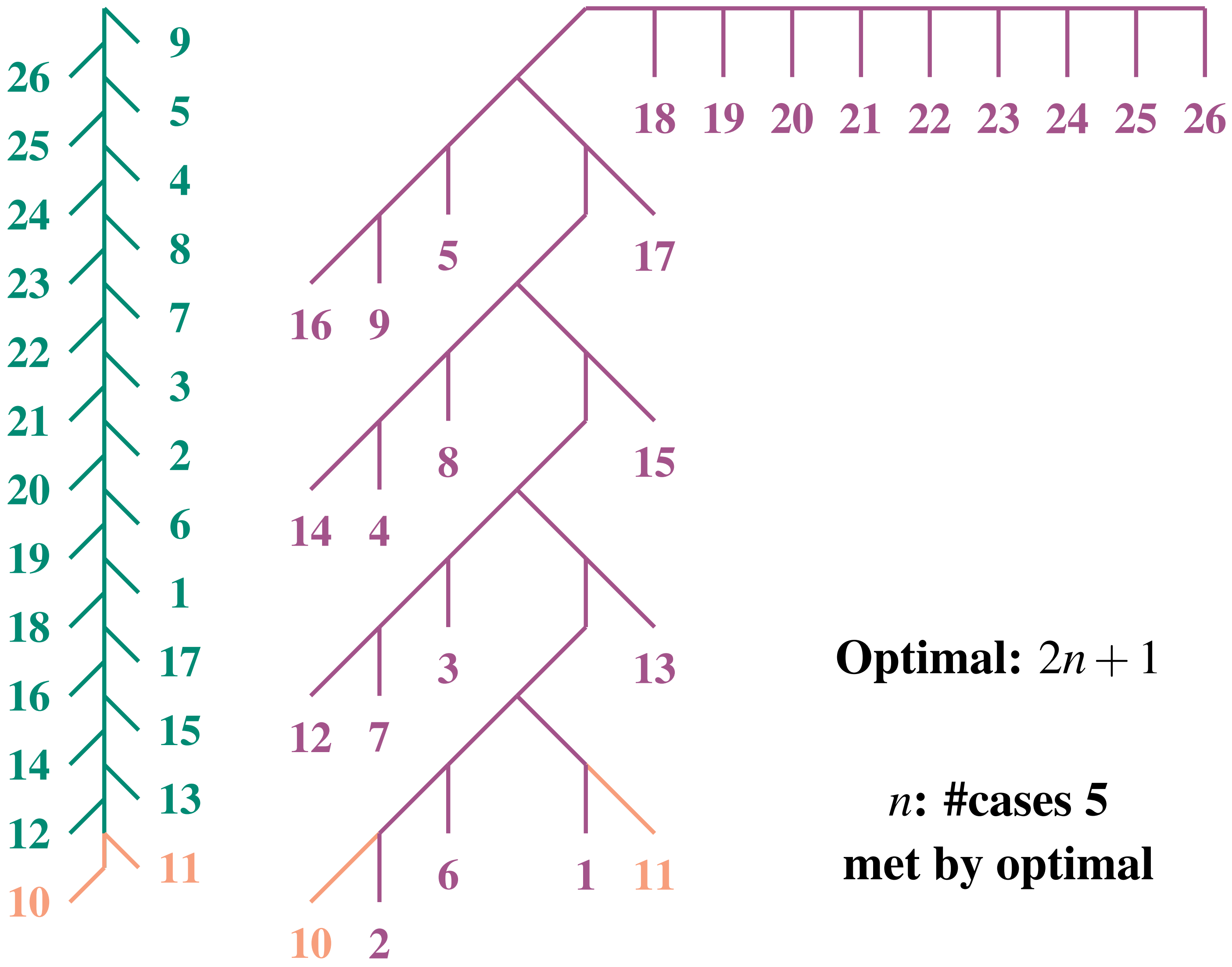
This is a tight ratio



Optimal: $2n+1$

n: **#cases 5**
met by optimal

This is a tight ratio

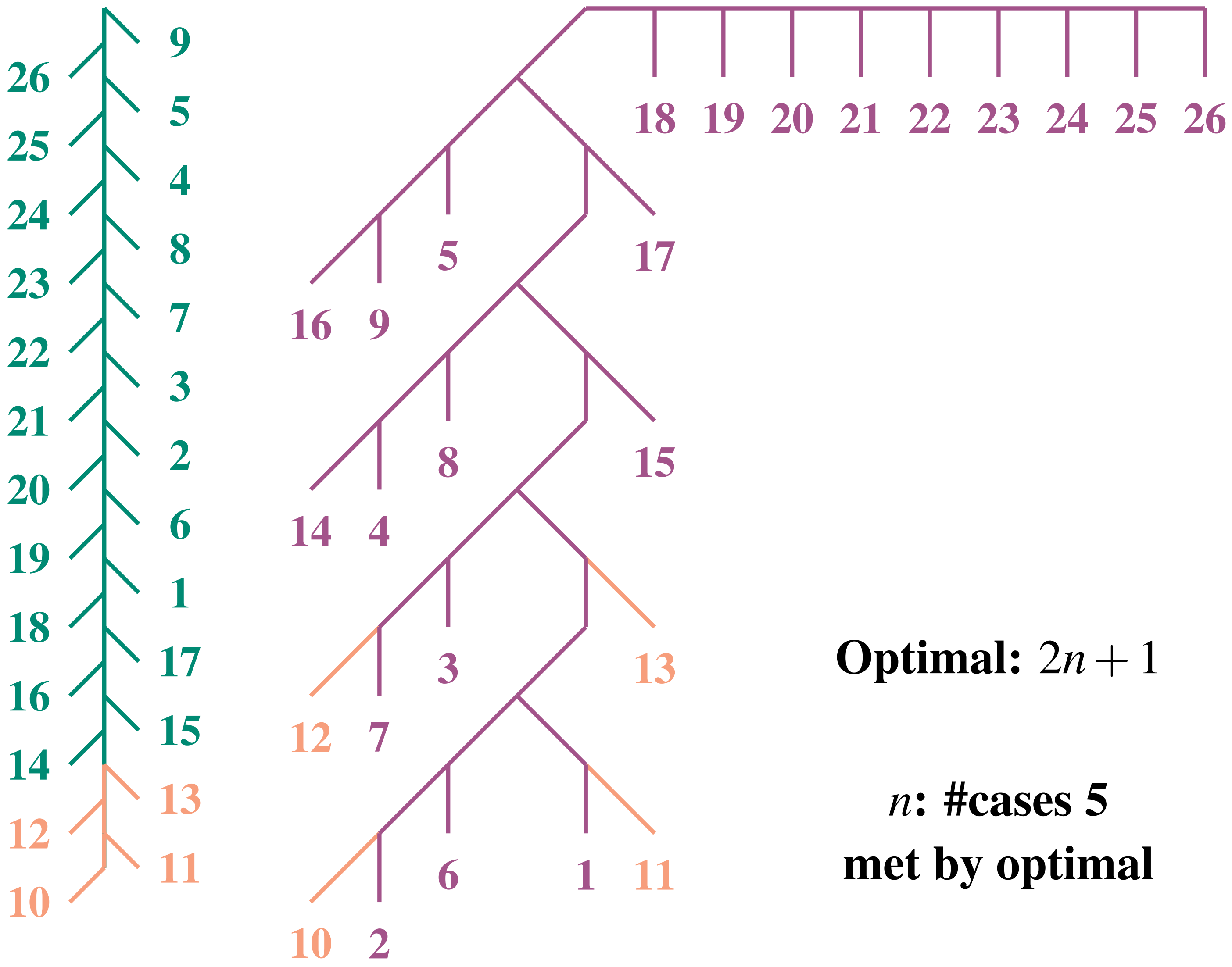


Optimal: $2n+1$

n: **#cases 5**

met by optimal

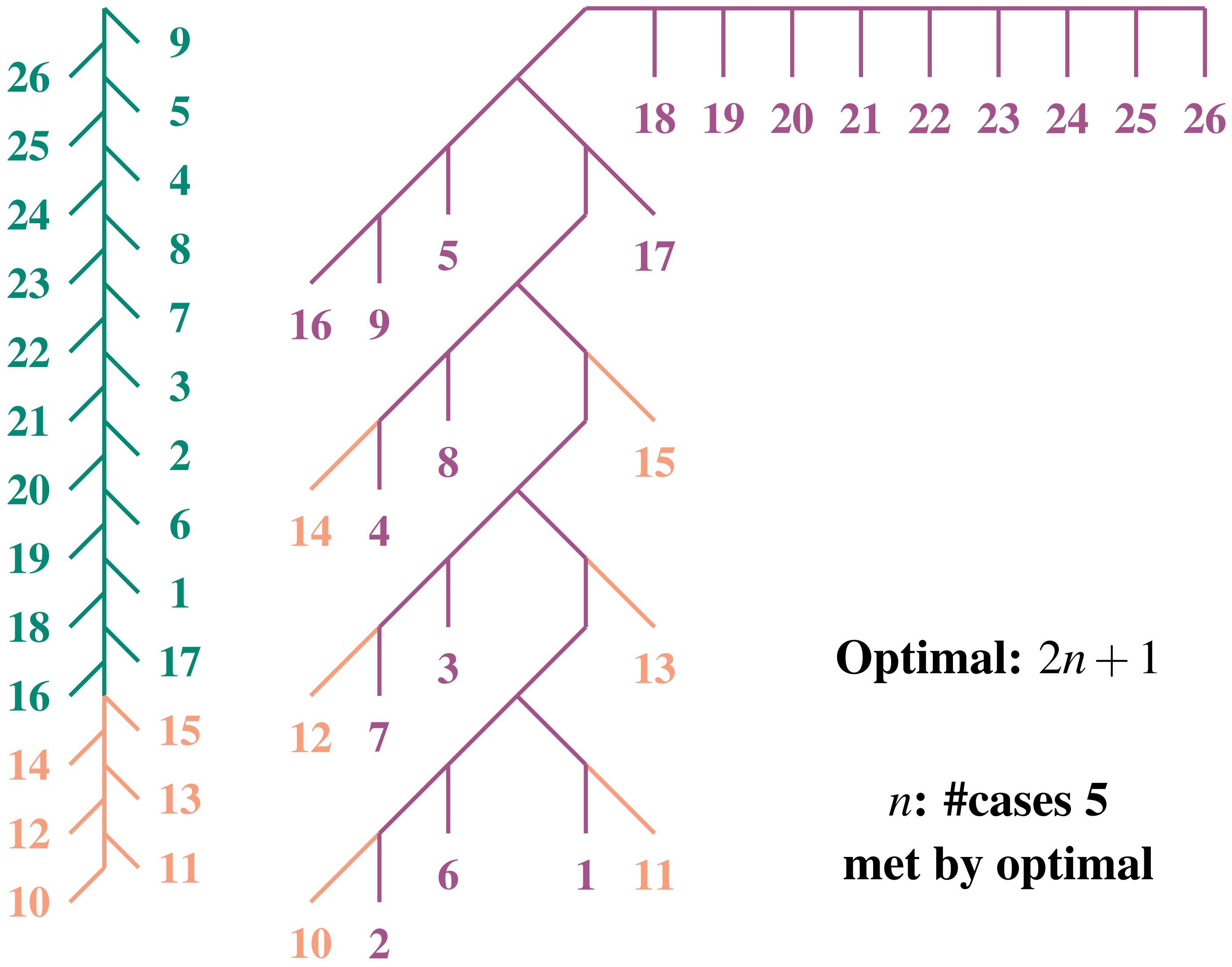
This is a tight ratio



Optimal: $2n+1$

n**: #cases 5**
met by optimal

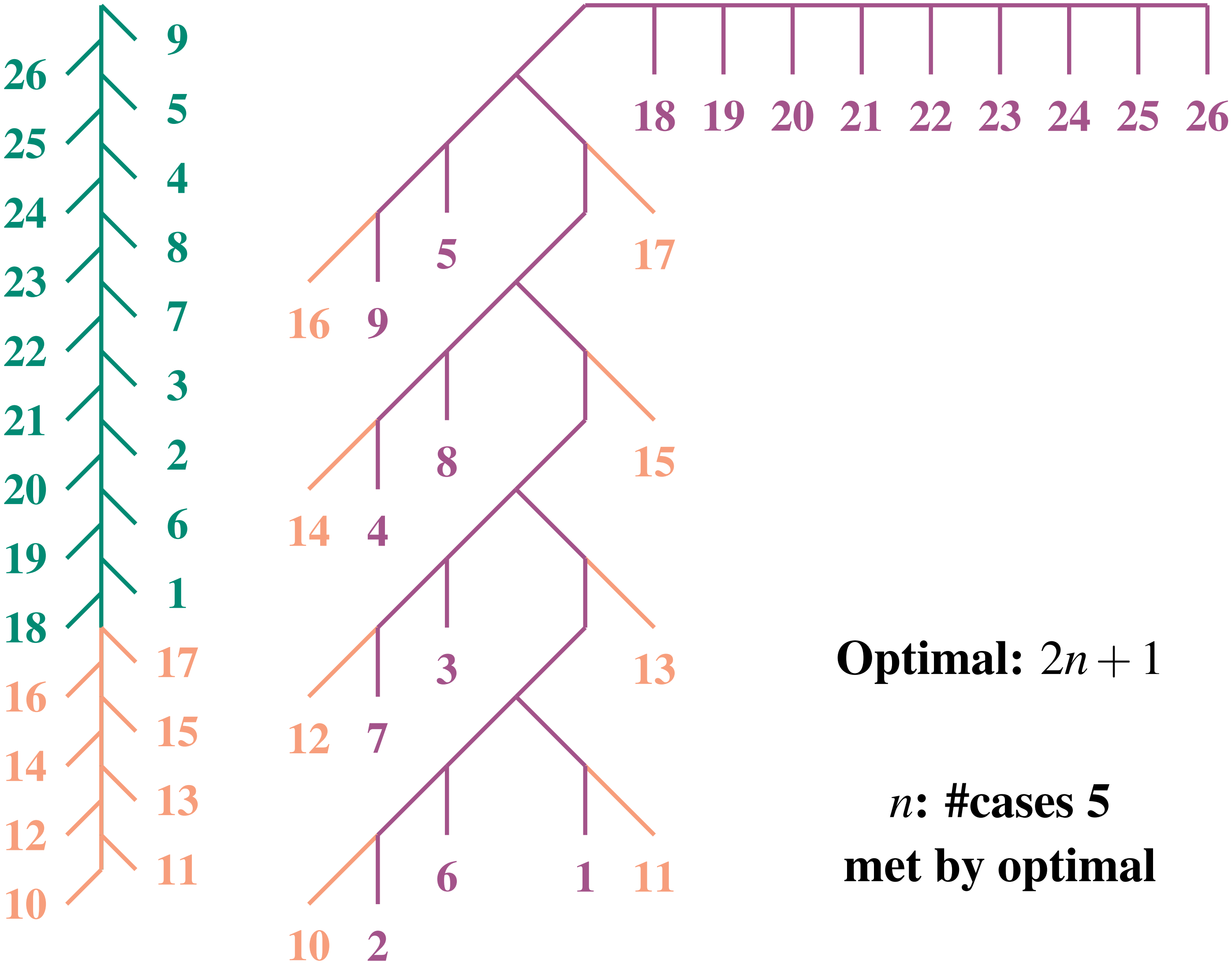
This is a tight ratio



Optimal: $2n+1$

n**: #cases 5**
met by optimal

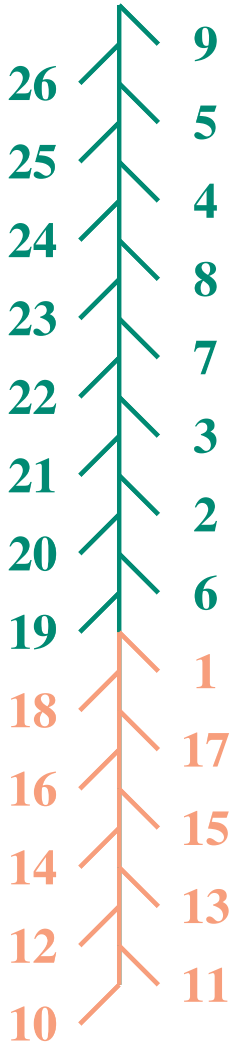
This is a tight ratio

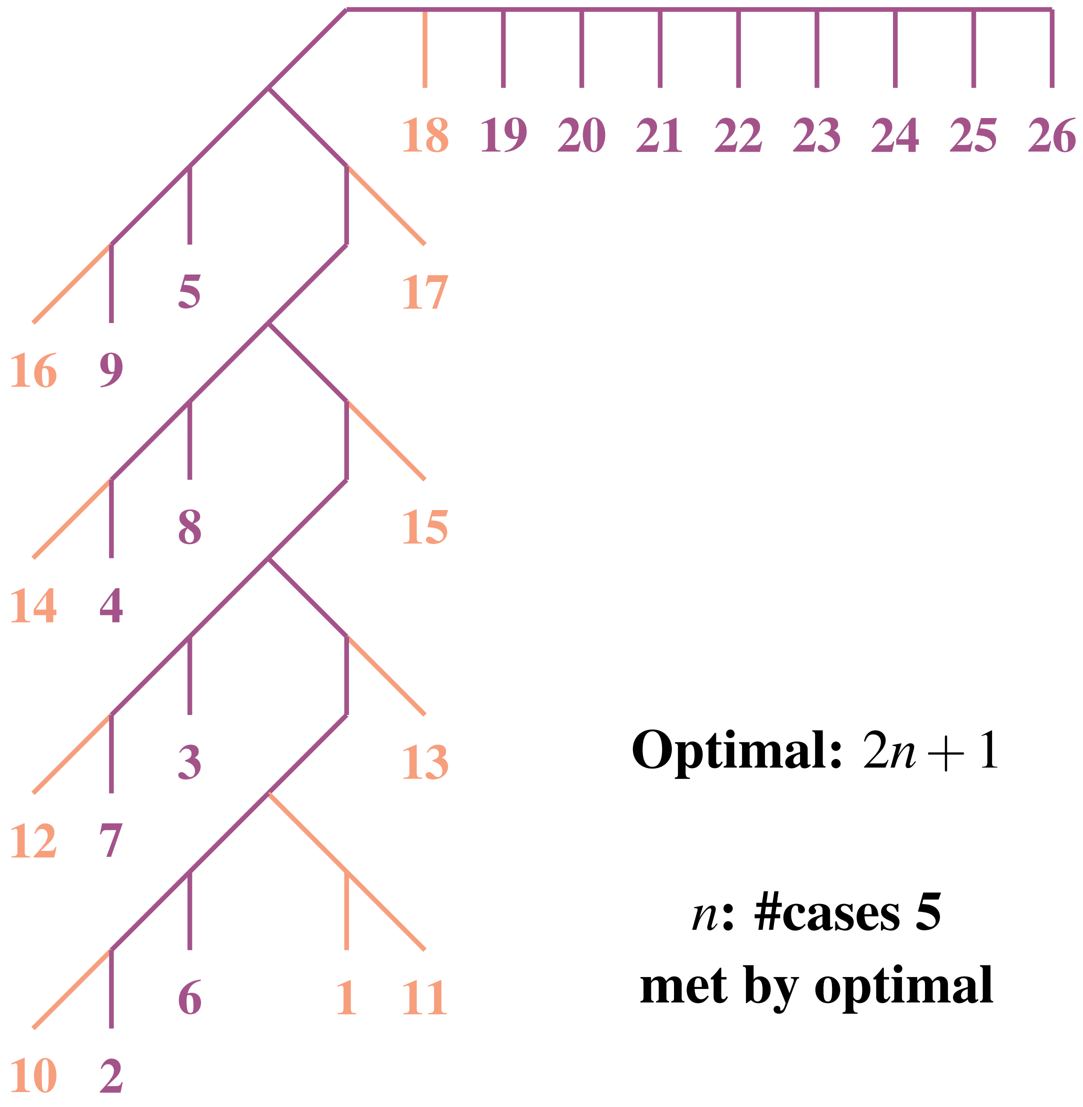


Optimal: $2n+1$

n: **#cases 5**
met by optimal

This is a tight ratio

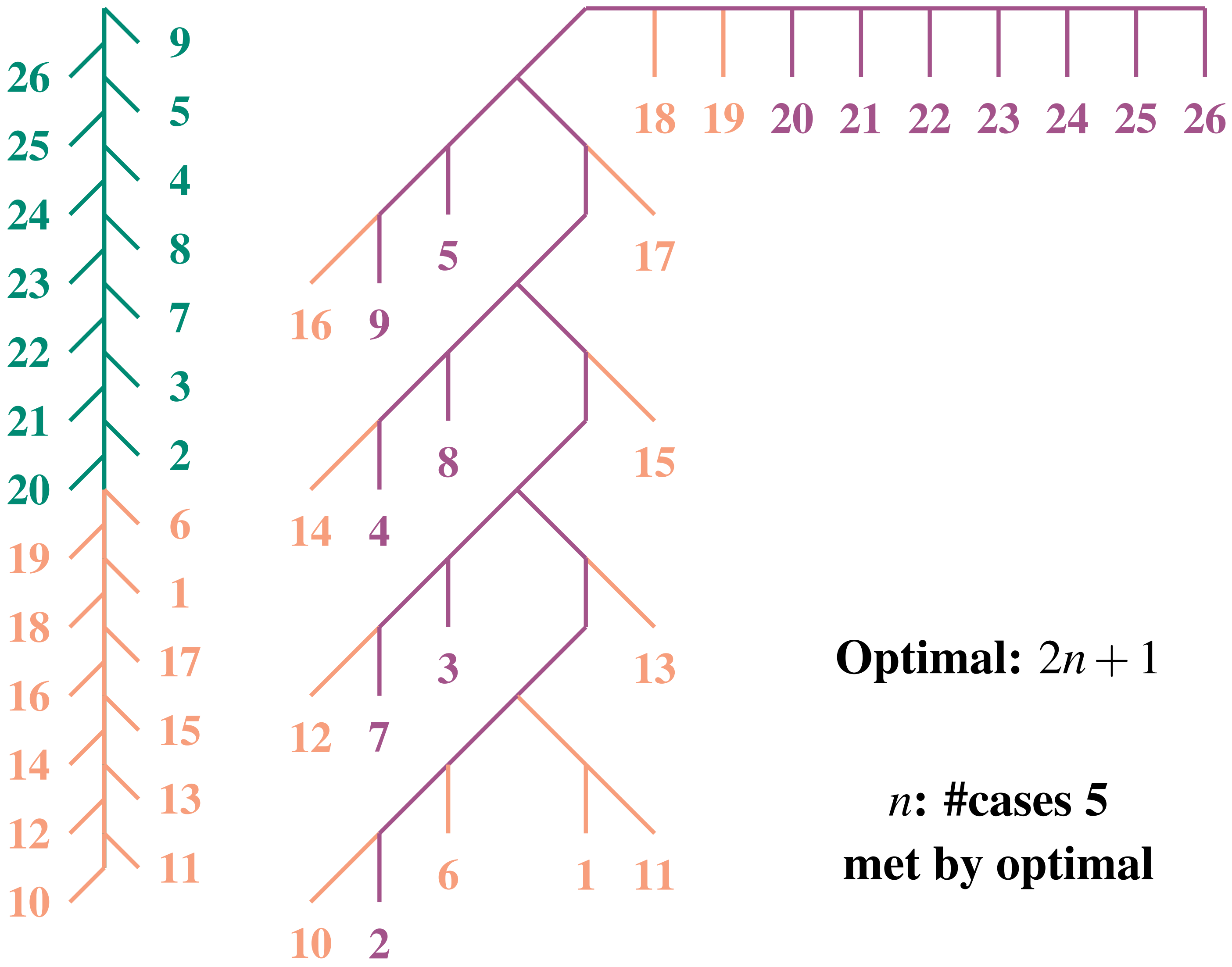




Optimal: $2n + 1$

n: **#cases 5 met by optimal**

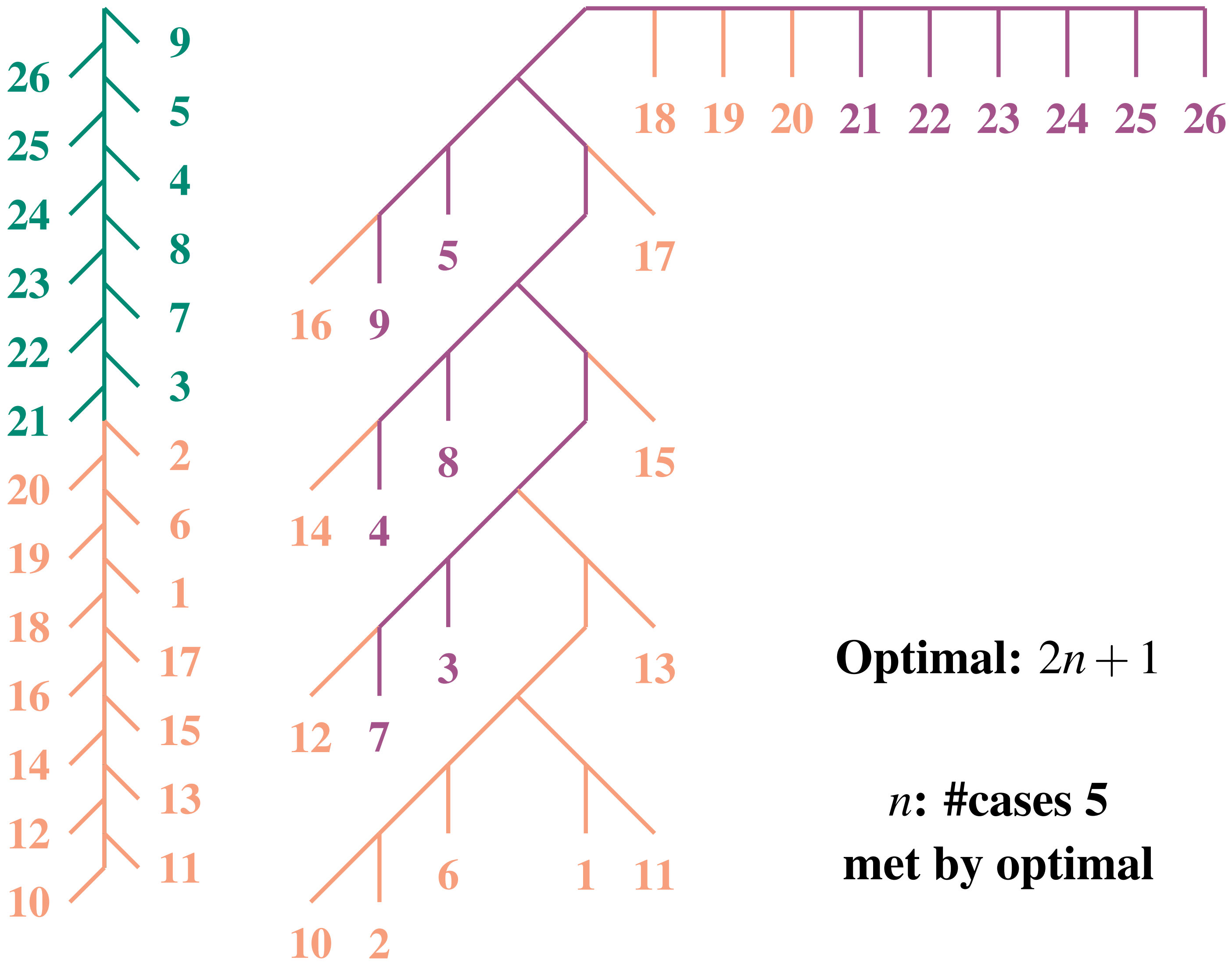
This is a tight ratio



Optimal: $2n + 1$

n: **#cases 5**
met by optimal

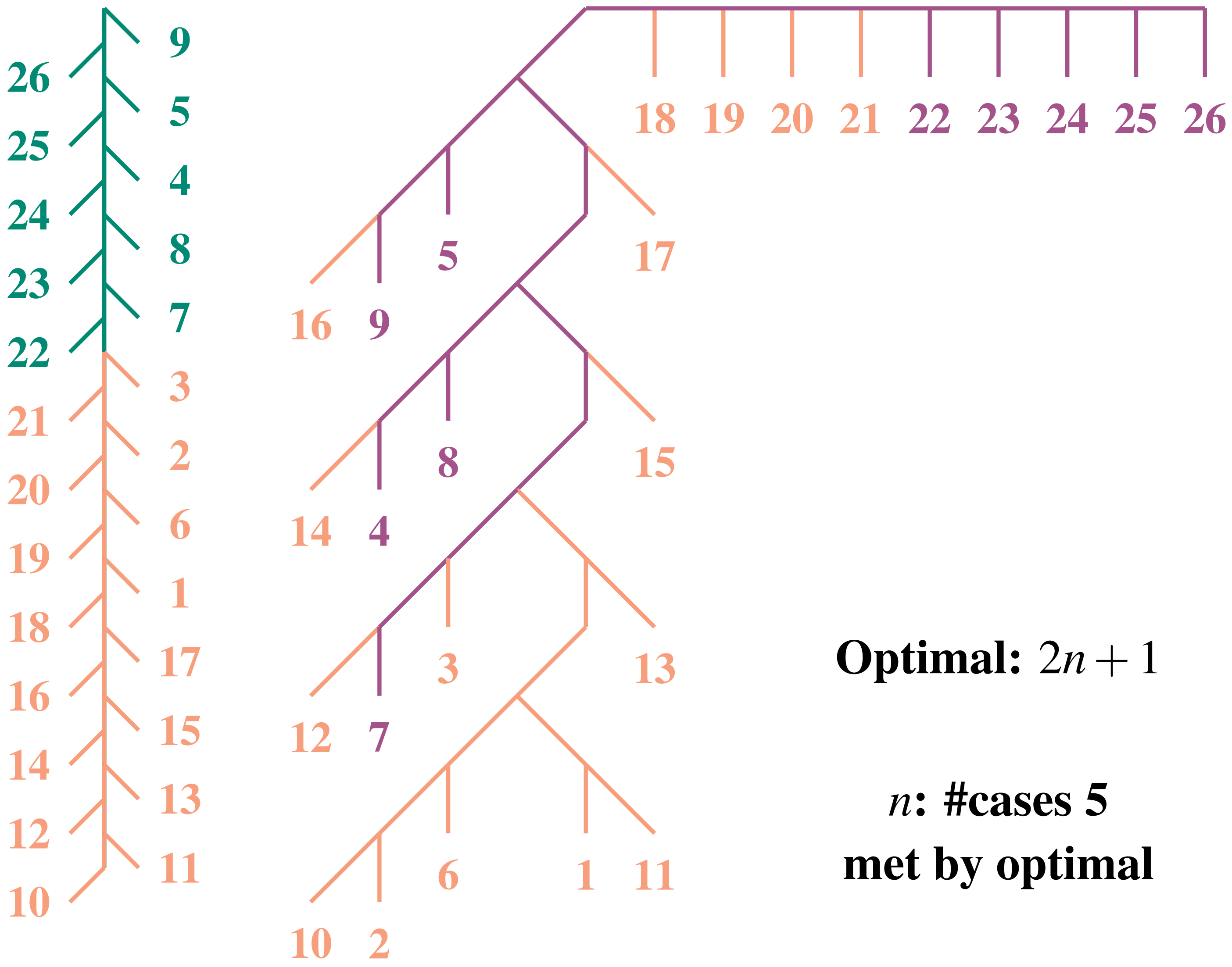
This is a tight ratio



Optimal: $2n+1$

n: **#cases 5**
met by optimal

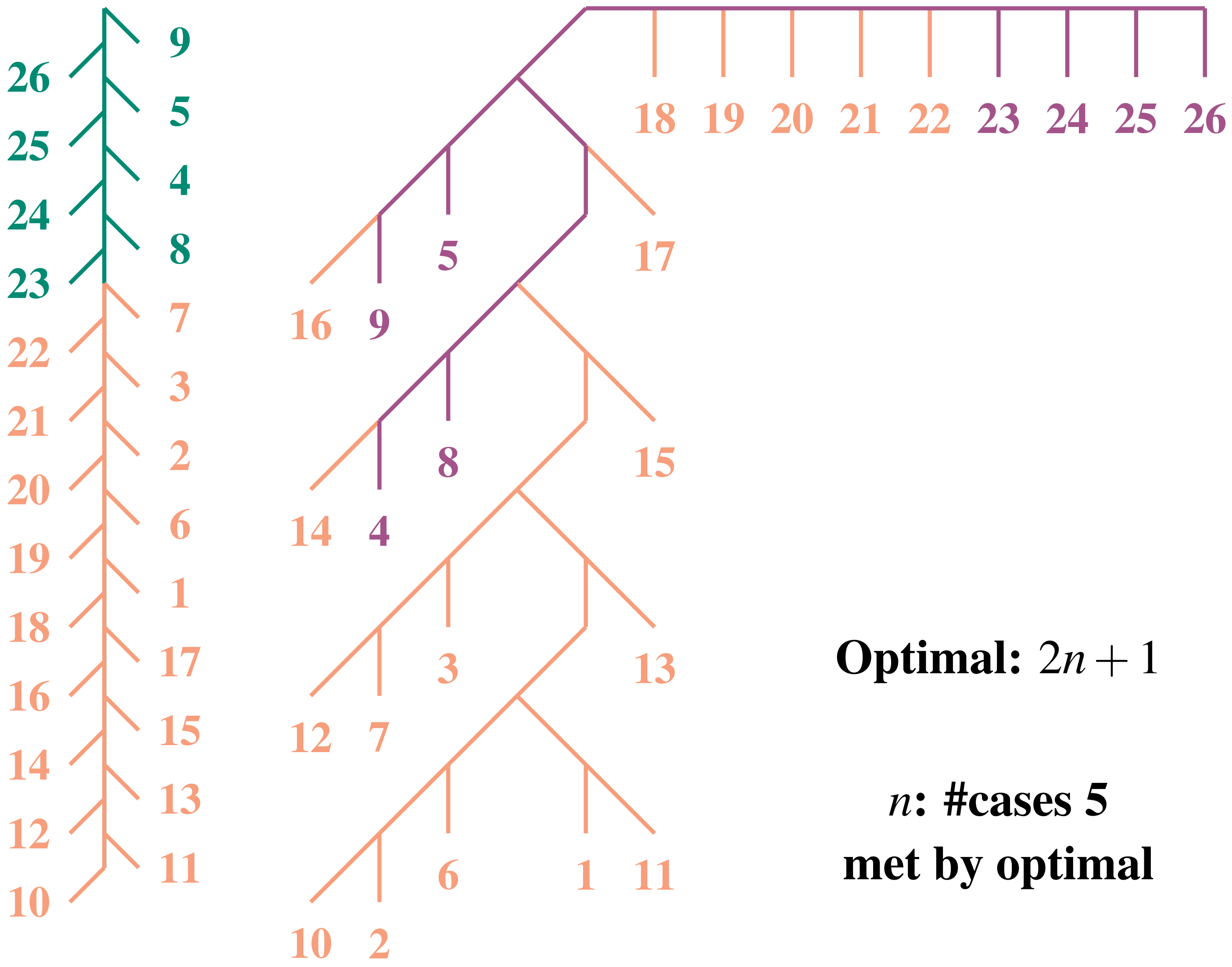
This is a tight ratio



Optimal: $2n+1$

n: **#cases 5**
met by optimal

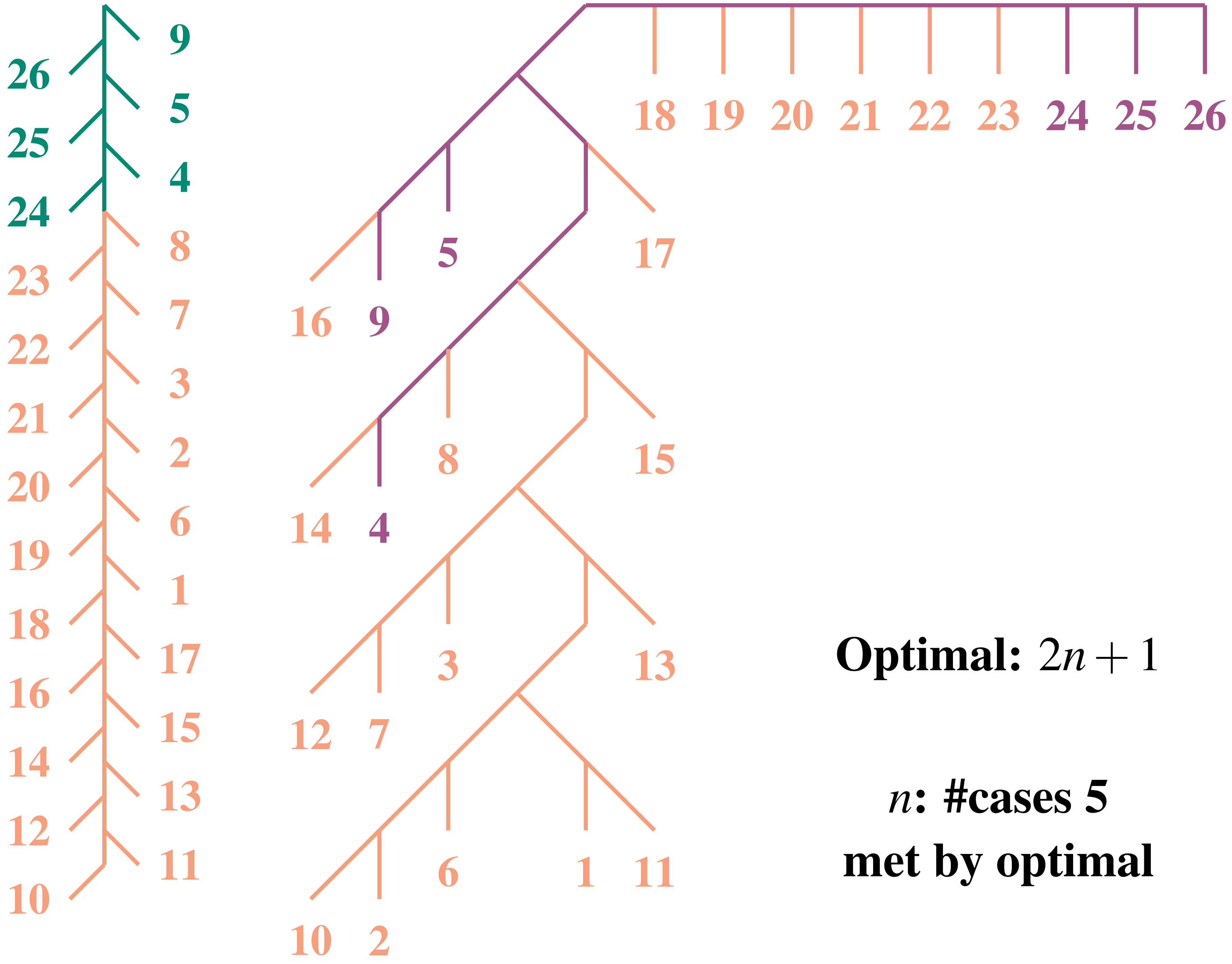
This is a tight ratio



Optimal: $2n+1$

n**: #cases 5**
met by optimal

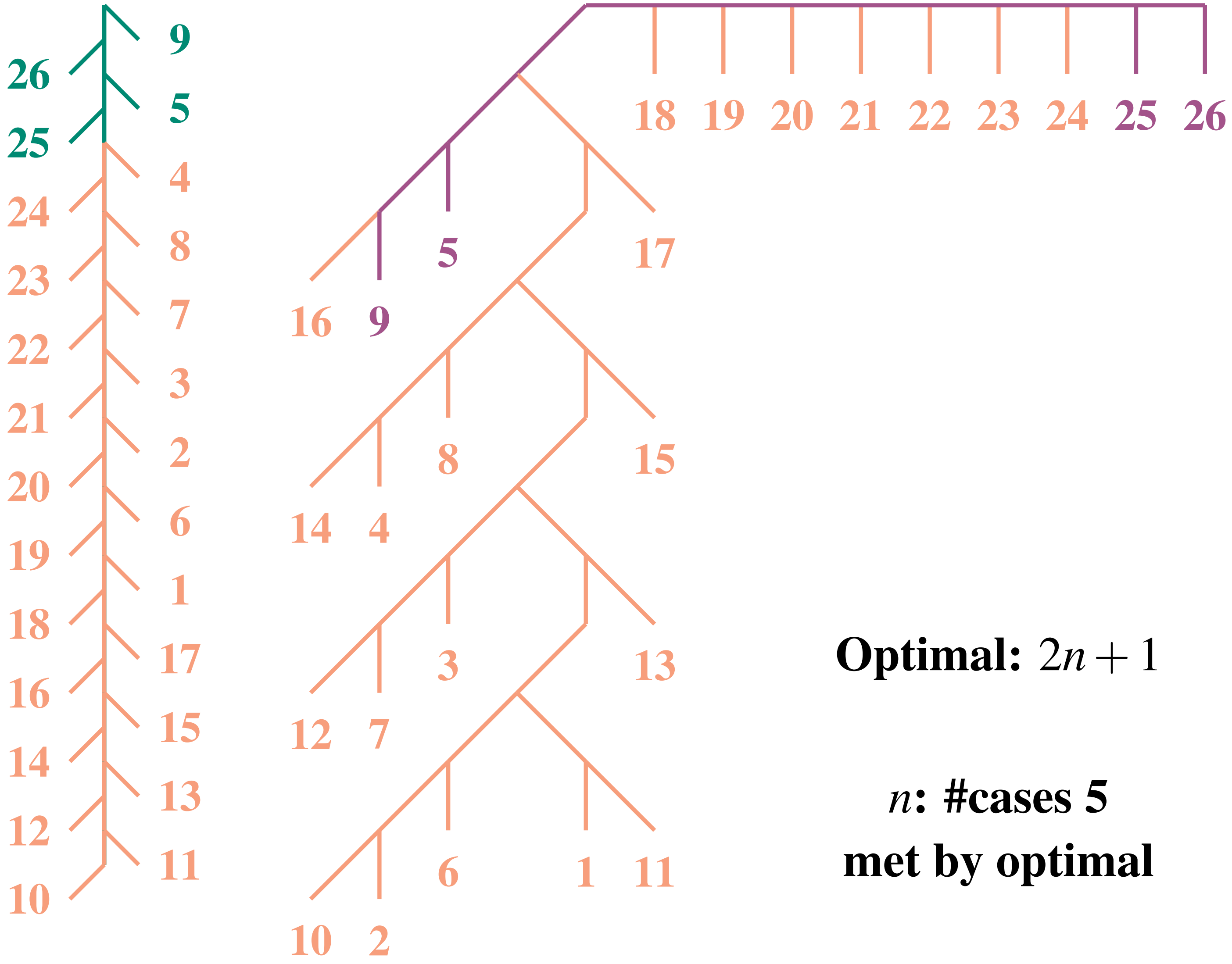
This is a tight ratio



Optimal: $2n + 1$

n**: #cases 5**
met by optimal

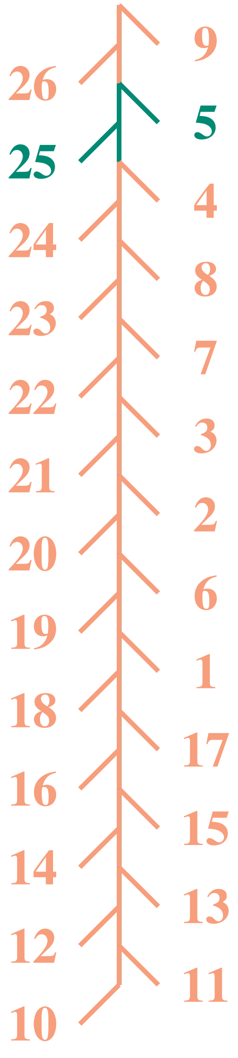
This is a tight ratio

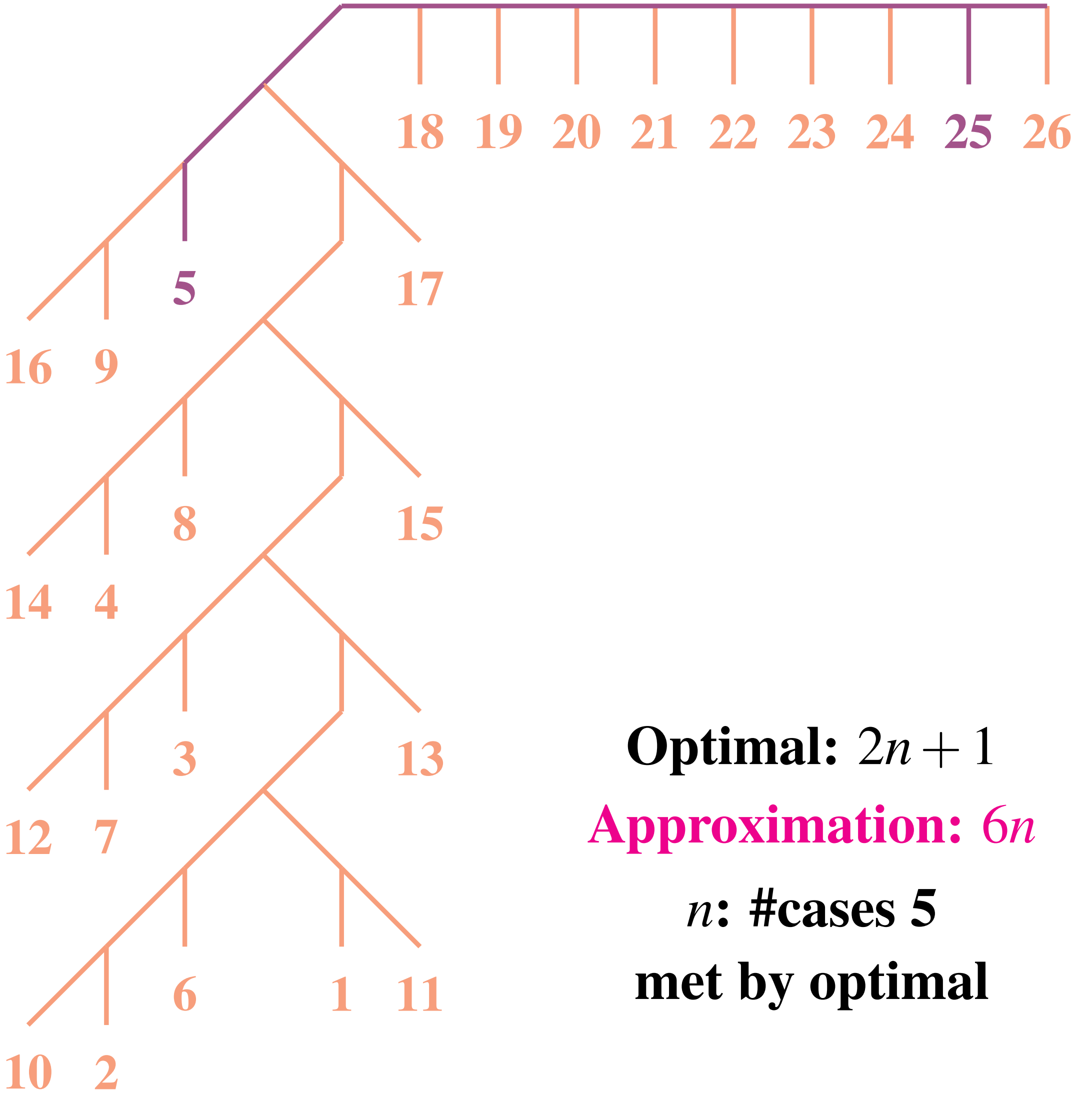


Optimal: $2n+1$

n: **#cases 5**
met by optimal

This is a tight ratio





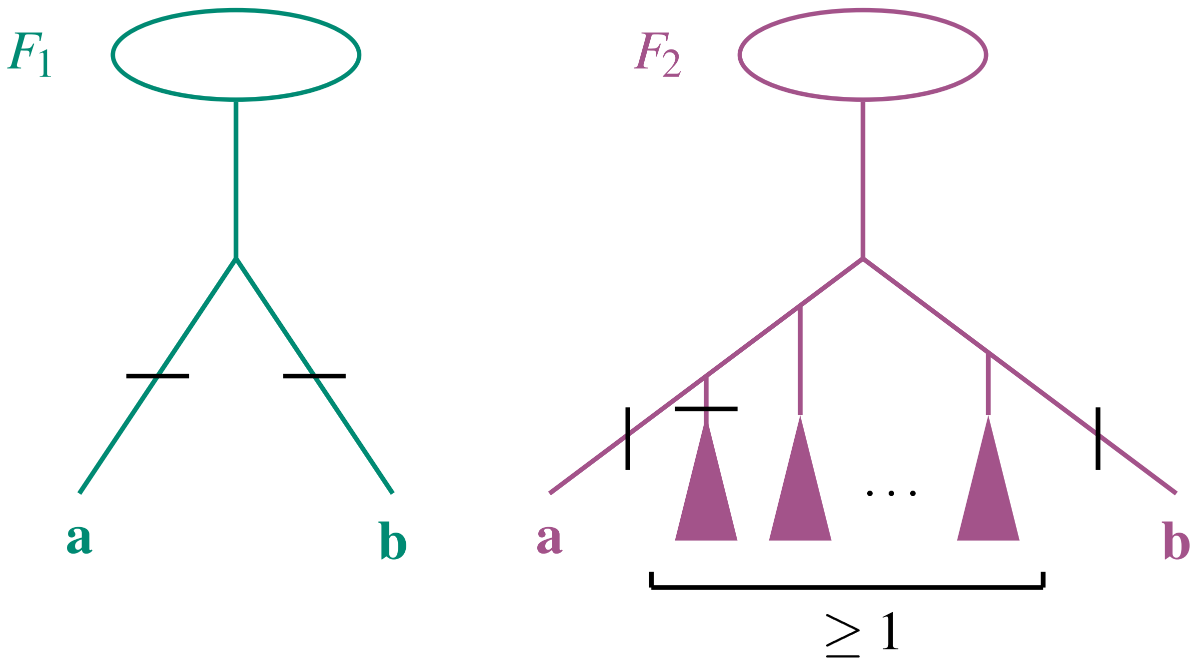
Optimal: $2n+1$

Approximation: $6n$

n**: #cases 5 met by optimal**

A simpler algorithm with the same ratio

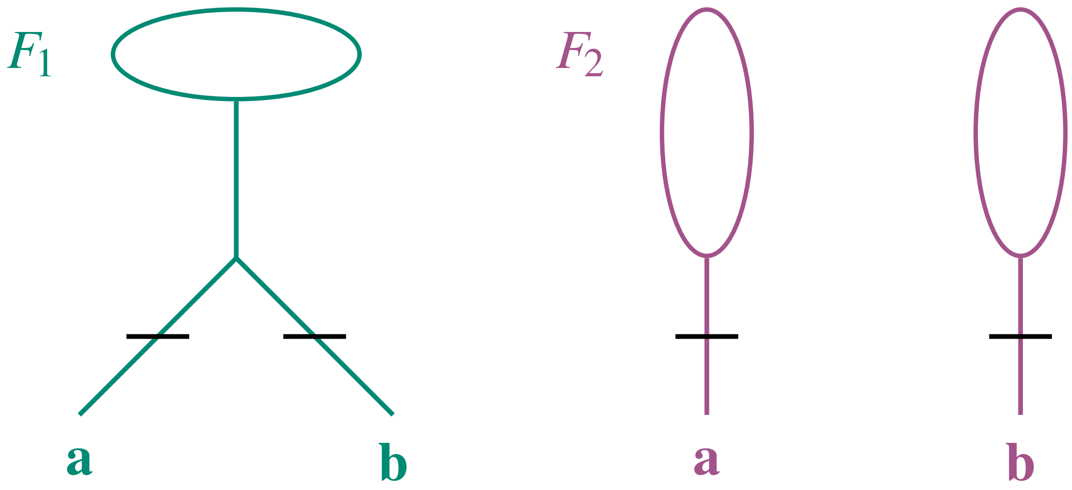
Cases 4 and 5:



cut the two edges in F_1 and F_2
leading to the leaves labelled a and b
plus one subtree (any one) hanging from the path from a to b

A 3-approximation algorithm providing a lower bound on the optimal

Case 3:



cut two edges each in F_1 and F_2, contract in F_1
and cut one more edge incident to a (non-isolated) leaf in F_2

$$|\text{sol. this algo.} - 1| \leq 3\,(|\text{sol. optimal}| - 1)$$

Results on simulated data

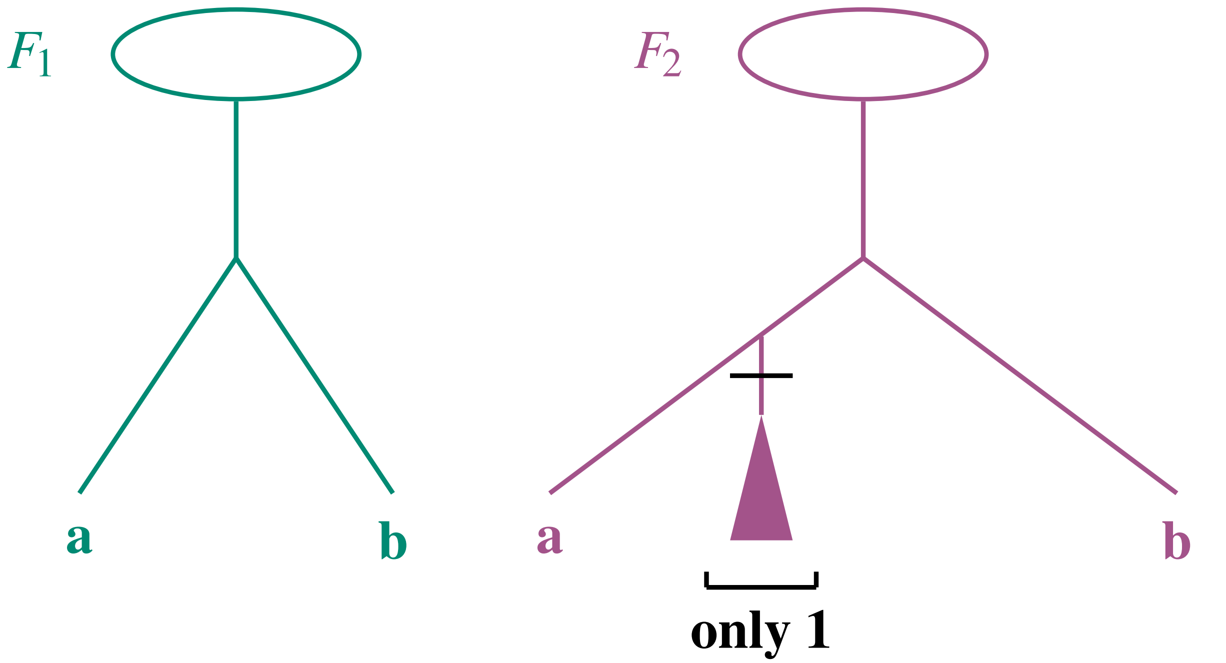
File	Alg.2	Alg.1	Alg.3	lb	(Alg.2)/lb
Optimum $\leq$ 100 (99 subtree tranfers)					
001-100	139	150	265	89	1.57
002-100	150	153	270	91	1.65
003-100	152	156	277	93	1.65
004-100	137	141	255	86	1.60
005-100	148	151	267	90	1.65
006-100	142	148	250	84	1.70
007-100	142	148	265	89	1.60
008-100	144	147	266	90	1.60
009-100	146	146	264	89	1.65
010-100	147	153	259	87	1.69

Results on simulated data

File	Alg.2	Alg.1	Alg.3	lb	(Alg.2)/lb
Optimum $\leq$ 400 (399 subtree tranfers)					
001-400	515	556	744	249	2.07
002-400	526	562	731	245	2.15
003-400	513	553	731	245	2.10
004-400	492	546	723	242	2.04
005-400	514	562	733	245	2.10
006-400	526	569	708	237	2.22
007-400	508	558	758	254	2.00
008-400	501	551	711	238	2.11
009-400	529	570	747	250	2.12
010-400	539	582	751	251	2.15

Hein's algorithm

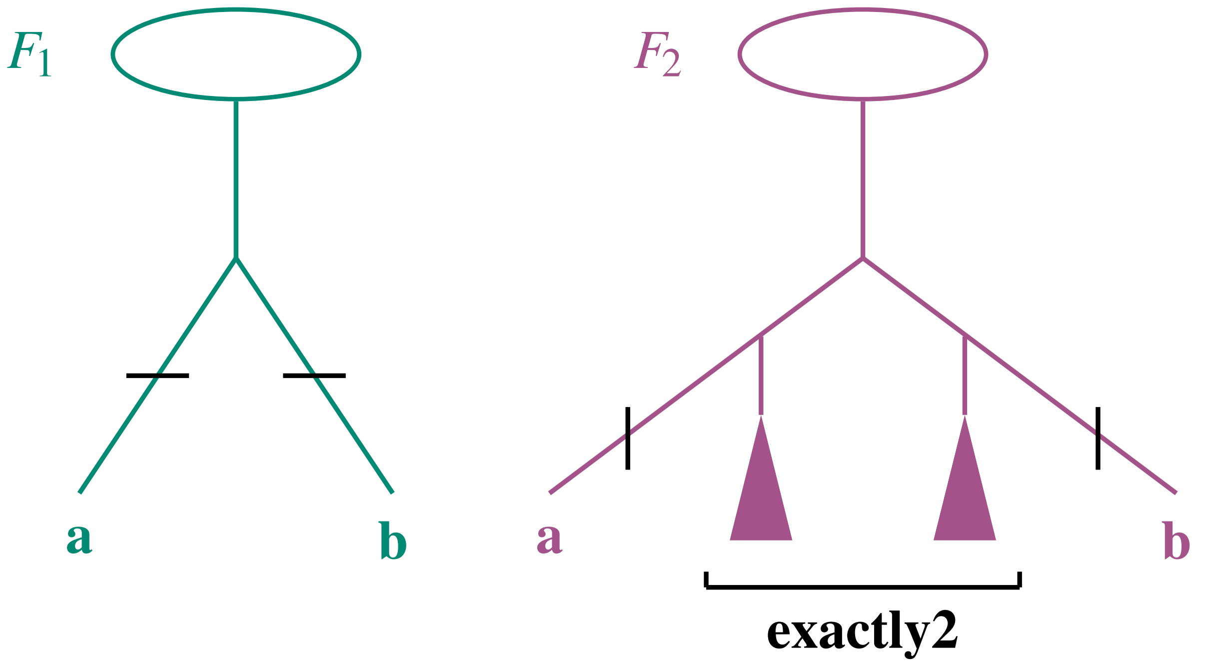
Case 4:



cut the subtree hanging from the path from a to b in F_2

Hein's algorithm

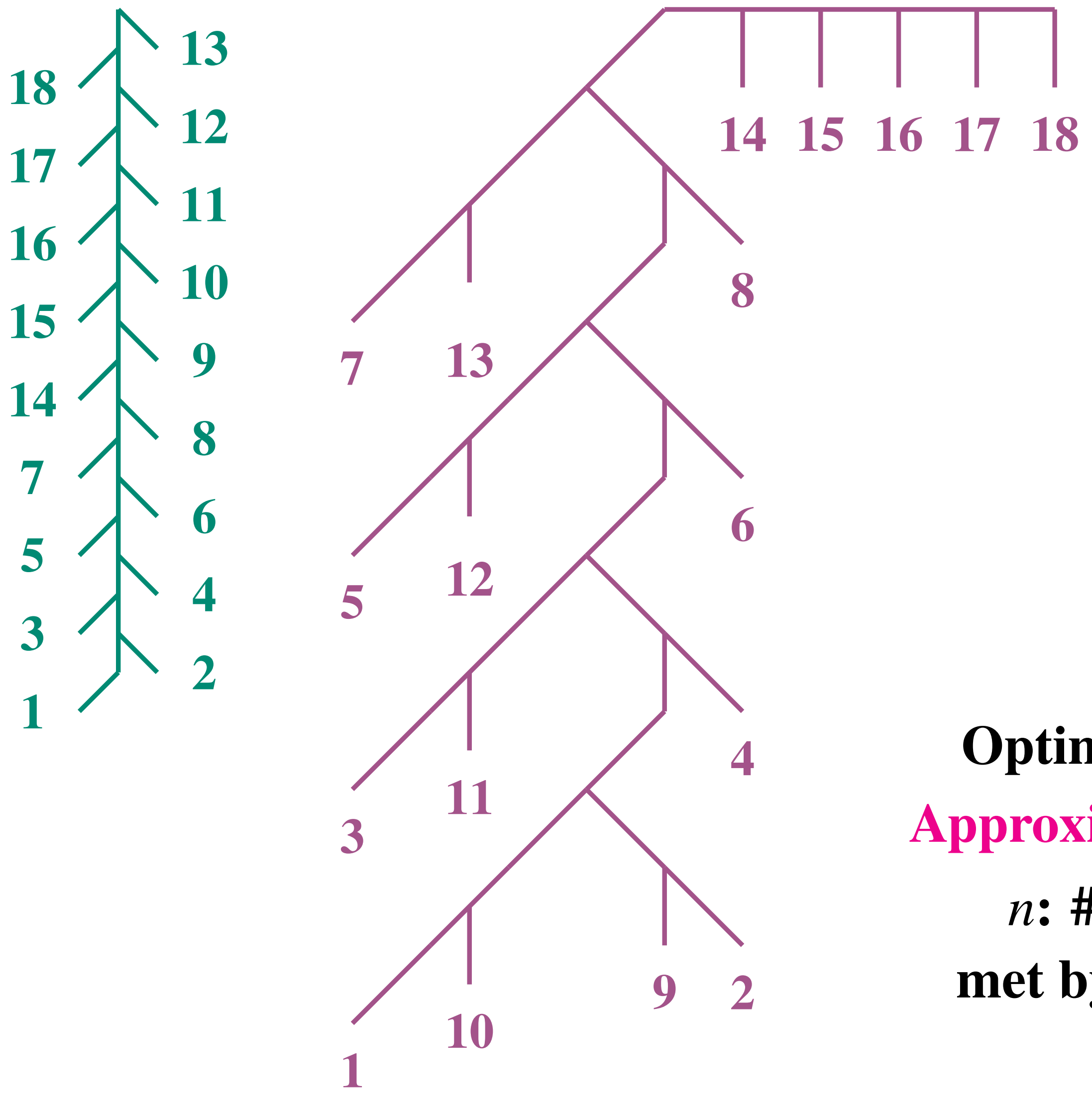
Case 4 which becomes part of case 5:



cut edges leading to the leaves labelled a and b

and contract in F_1

Hein's ratio of 4 is also tight



Optimal: $n+1$

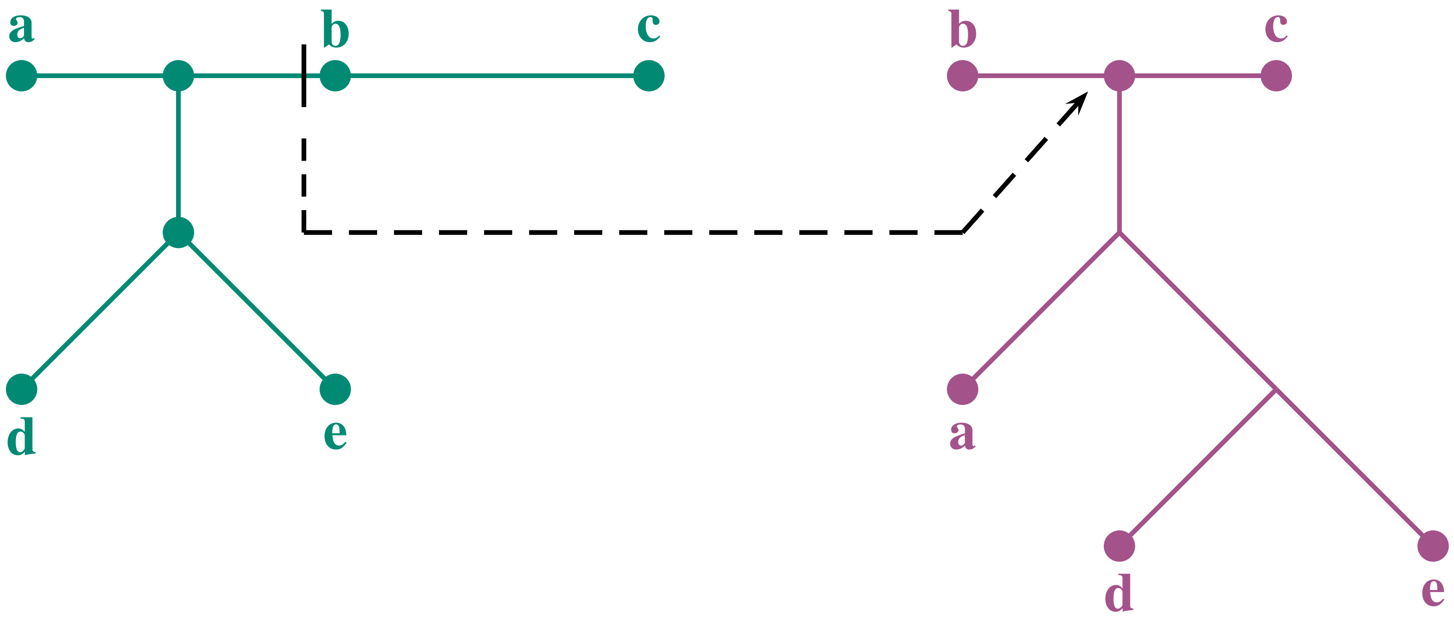
Approximation: $4n$

n**: #cases 5**

met by optimal

A (better?) way of modelling genetic transfer distance between two rooted trees

Subtree Prune and Regraft (SPR)



Relation between MAF and SPR

For any two binary trees T_1 and T_2 labelled at the leaves

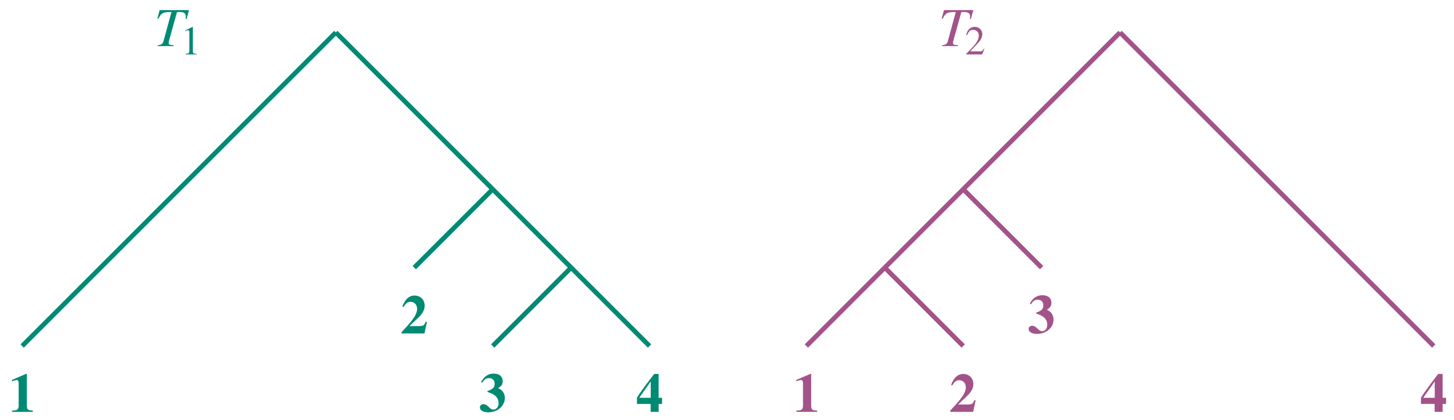
$$D_{SPR}(T_1, T_2) = MAF(T_1, T_2) - 1$$

(Claimed by J. Hein *et al*, 1997)

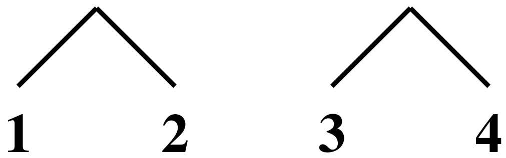
$$D_{SPR}(T_1, T_2) \neq MAF(T_1, T_2) - 1$$

(B. Allen and M. Steel, 2001)

Example



MAF of T_1 and T_2 **size 2**



SPR of T_1 and T_2 **2 moves required**

Relation between MAF and SPR revisited

For any two binary trees T_1 and T_2 labelled at the leaves

$$D_{SPR}(T_1,T_2) = \begin{array}{l} \quad MAF(T_1,T_2) \\ \textbf{or } MAF(T_1,T_2)-1 \end{array}$$

(E. M. Rodrigues *et al.*, PhD thesis, in portuguese, 2003)

Our conjecture

This depends on whether the root is preserved or not

(E. M. Rodrigues *et al.*, in progress)

Some of our perspectives

Prove the conjecture and improve the bounds

Obtain a better than 3-approximation for the MAF

Examine the "fixed parameter tractability" of these and other distances

> **Ex: Can we obtain an algorithm with $O(n^{O(1)} f(d))$ where d is the distance, n the number of leaves and $f(d)$ can be any function (even exponential, but the exponential function involves only d) ?**
>
> **(work in preparation)**

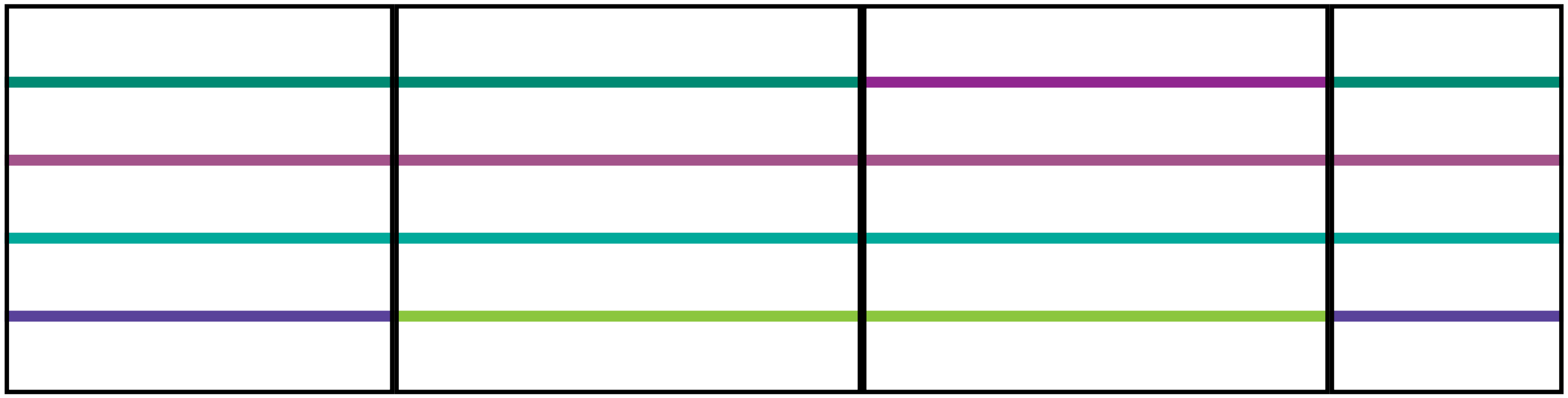
Examine further what is (perhaps ?) the most interesting genetic transfer distance: weighted SPR

The most difficult ? $\rightarrow$ go back to where we came from

Genetic transfer detection and localization
as an optimisation problem

The (ideal ?) result

alignment



tree 1 **tree 2** **tree 3** **tree 4**

"optimal" partition: sum of lengths of trees + penalty for genetic transfers is MINIMUM

So far

A more "pragmatic" approach using windows and the ILD test

(DRUID, M. Zelwer *et al.*, paper in preparation)