LEO VAN IERSEL

TU DELFT

LEO VAN IERSEL

TU DELFT

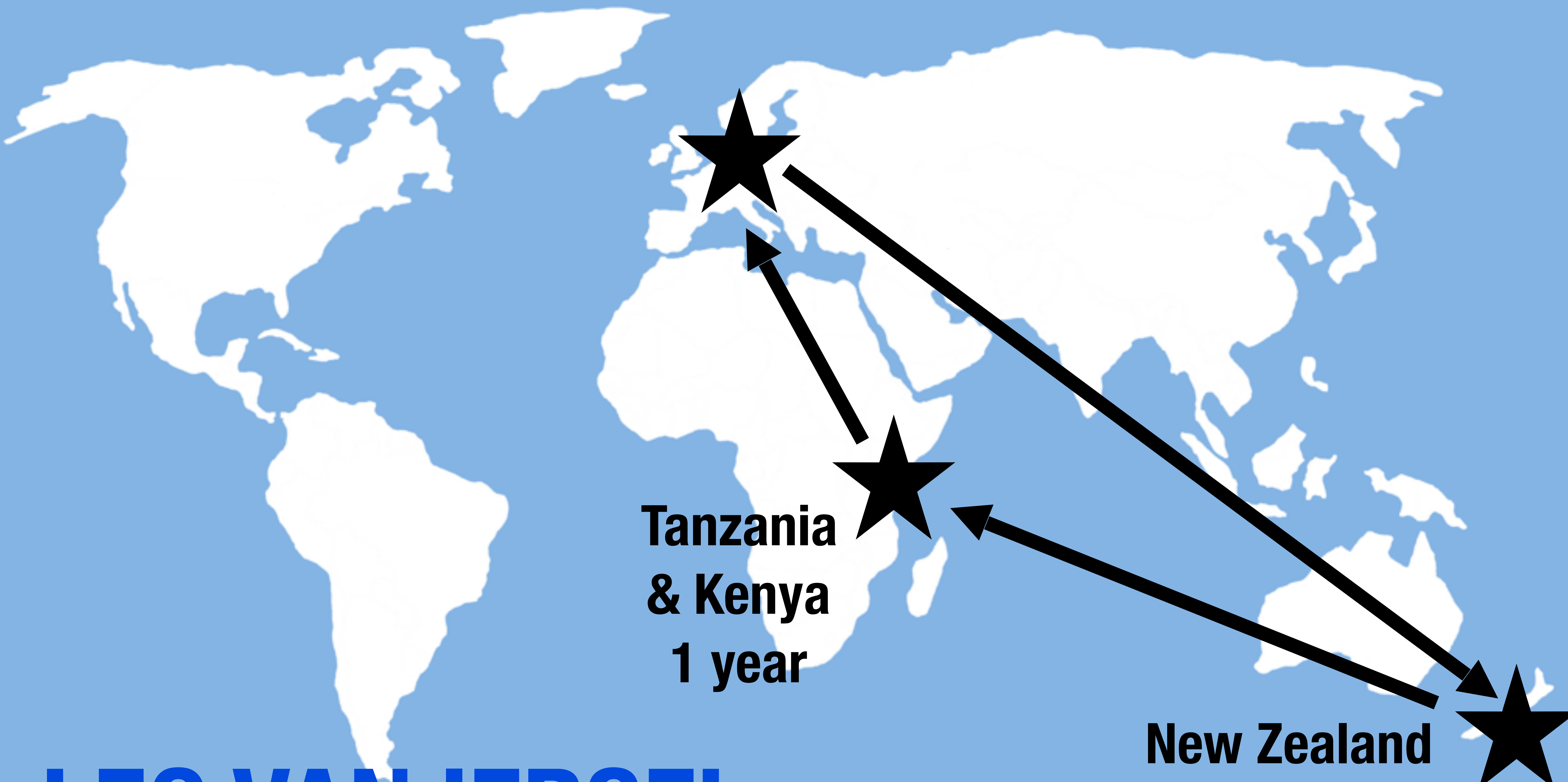
LEO VAN IERSEL

TU DELFT


LEO VAN IERSEL
TU DELFT
CWI
UT
TU/e
NOORDZEE
WADDENZEE
DUITSLAND
Noord-Holland
Friesland
Groningen
Drenthe
Overijssel
Flevoland
Gelderland
Utrecht
Zuid-Holland
Zeeland
Noord Brabant
Limburg

LEO VAN IERSEL

TU DELFT

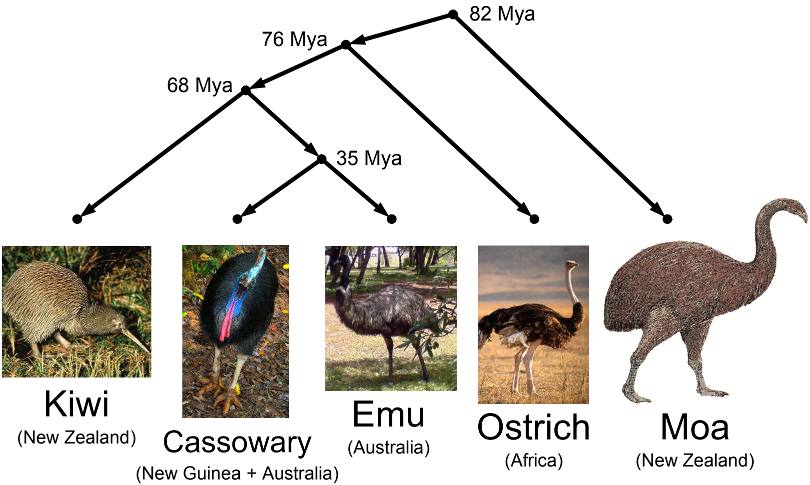


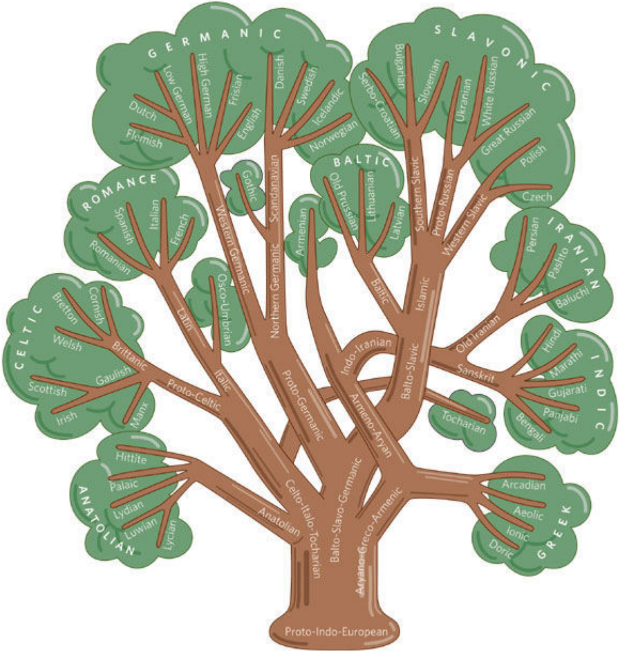
LEO VAN IERSEL

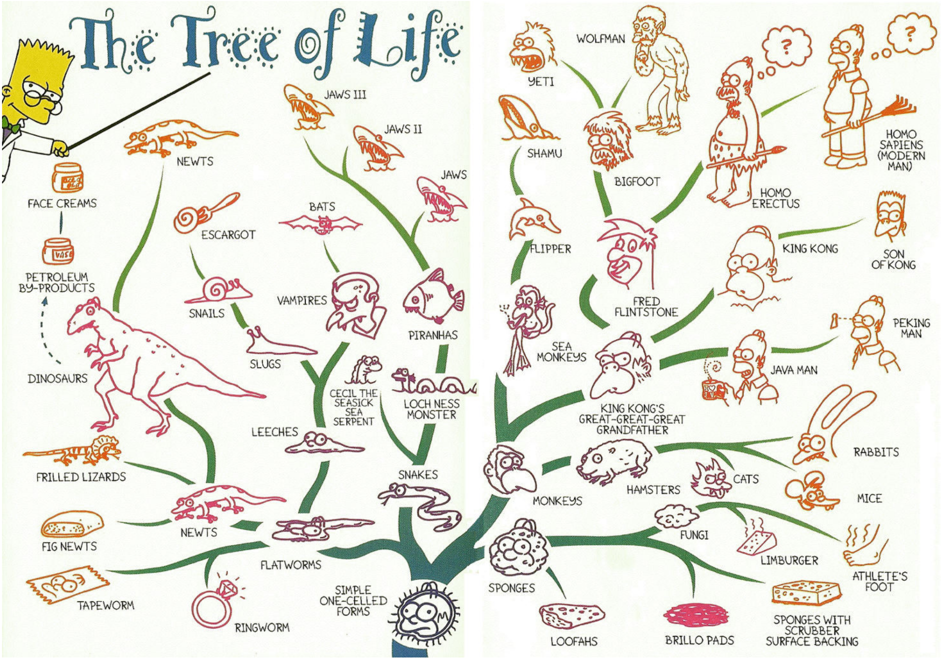
TU DELFT

Definition

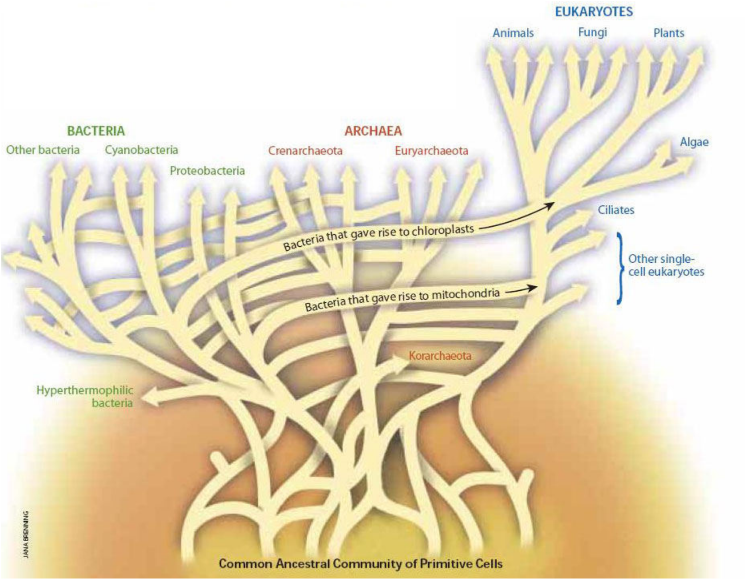
Let X be a finite set. A **(rooted) phylogenetic tree** on X is a rooted tree with no indegree-1 outdegree-1 vertices whose leaves are bijectively labelled by the elements of X.




GERMANIC
Low German
High German
Frisian
Danish
Swedish
Icelandic
Dutch
Flemish
English
Norwegian
SLAVONIC
Bulgarian
Serbo-Croatian
Slovenian
Ukranian
White Russian
Great Russian
Polish
Czech
BALTIC
Old Prussian
Lithuanian
Latvian
ROMANCE
Spanish
Italian
French
Romanian
Gothic
Scandanavian
Western Germanic
Armenian
Southern Slavic
Proto-Russian
Western Slavic
IRANIAN
Persian
Pashto
Baluchi
Osco-Umbrian
Northern Germanic
Baltic
Islamic
CELTIC
Bretton
Cornish
Welsh
Brittanic
Gaulish
Scottish
Irish
Manx
Latin
Italic
Proto-Celtic
Old Iranian
Indo-Iranian
INDIC
Hindi
Marathi
Gujarati
Panjabi
Bengali
Sanskrit
Balto-Slavic
Proto-Germanic
Armeno-Aryan
Tocharian
Hittite
Palaic
Lydian
Luwian
Lycian
ANATOLIAN
Anatolian
Celto-Italo-Tocharian
Balto-Slavo-Germanic
Aryano-Greco-Armenic
GREEK
Arcadian
Aeolic
Ionic
Doric
Proto-Indo-European



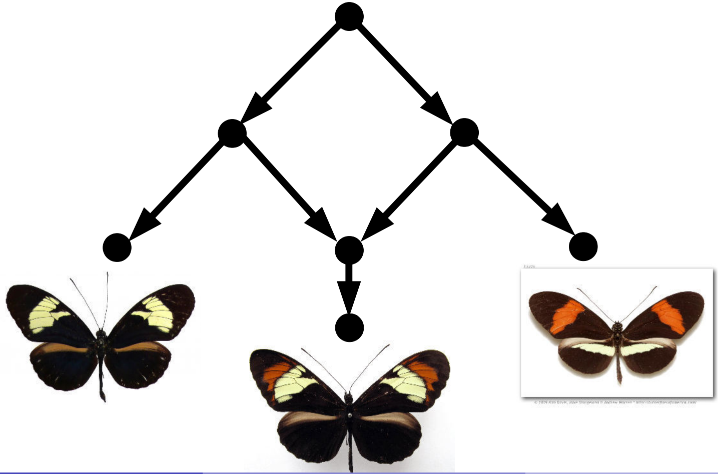
The Tree of Life
JAWS III
JAWS II
JAWS
NEWTS
FACE CREAMS
PETROLEUM BY-PRODUCTS
ESCARGOT
BATS
SNAILS
VAMPIRES
PIRANHAS
SLUGS
DINOSAURS
CECIL THE SEASICK SEA SERPENT
LOCH NESS MONSTER
LEECHES
FRILLED LIZARDS
SNAKES
NEWTS
FIG NEWTS
FLATWORMS
TAPEWORM
RINGWORM
SIMPLE ONE-CELLED FORMS
YETI
WOLFMAN
SHAMU
BIGFOOT
HOMO SAPIENS (MODERN MAN)
HOMO ERECTUS
FLIPPER
FRED FLINTSTONE
KING KONG
SON OF KONG
SEA MONKEYS
PEKING MAN
JAVA MAN
KING KONG'S GREAT-GREAT-GREAT GRANDFATHER
RABBITS
MONKEYS
HAMSTERS
CATS
MICE
FUNGI
LIMBURGER
ATHLETE'S FOOT
SPONGES
LOOFAHS
BRILLO PADS
SPONGES WITH SCRUBBER SURFACE BACKING



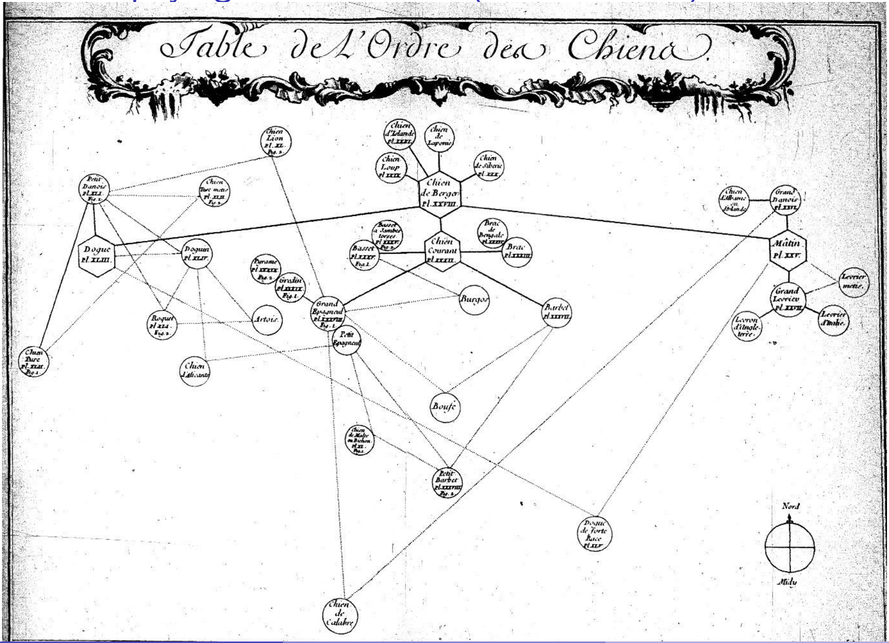
W.F. Doolittle et al. (2000)

Definition

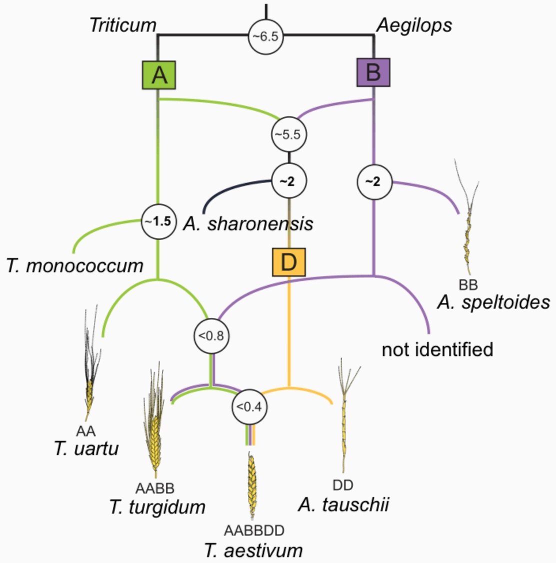
Let X be a finite set. A **(rooted) phylogenetic network** on X is a rooted directed acyclic graph with no indegree-1 outdegree-1 vertices whose leaves are bijectively labelled by the elements of X.



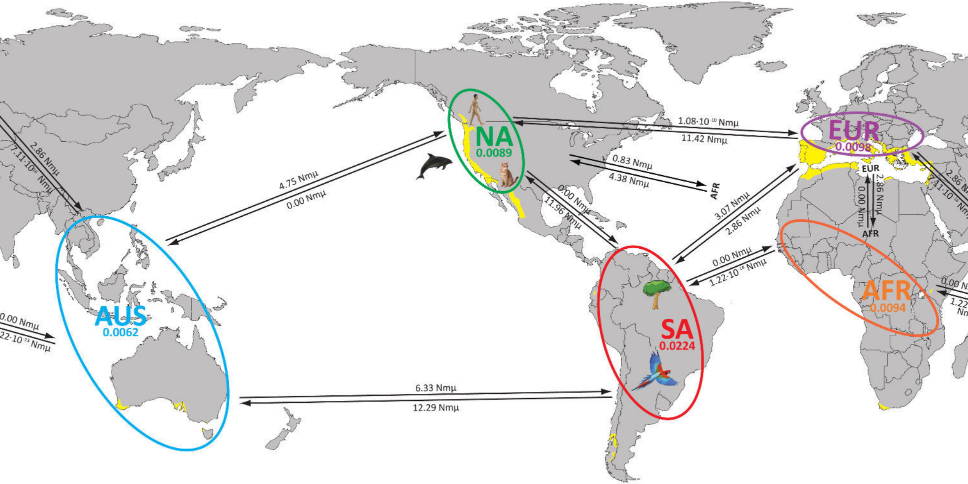
The first phylogenetic network (Buffon, 1755)



Marcussen et al., Ancient hybridizations among the ancestral genomes of bread wheat. Science (2014)



Origin of tropical pathogen C. gattii traced to the Amazon



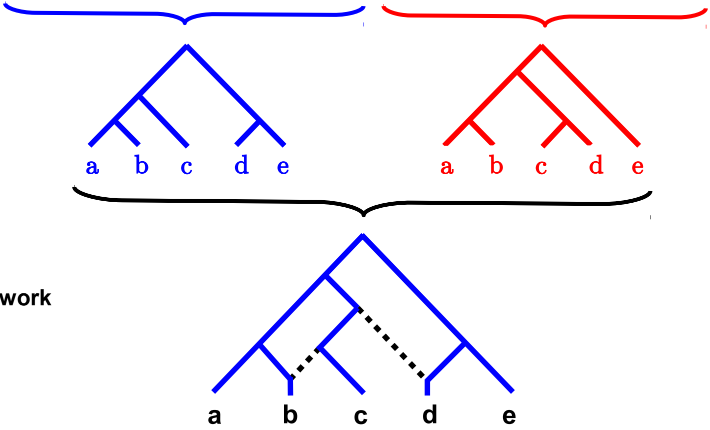
Hagen et al., Ancient dispersal of the human fungal pathogen Cryptococcus gattii from the Amazon rainforest. PLoS ONE (2013).

PART 1: NETWORKS FROM TREES

Species a	ACCCTAG--TC-ATC---AGC-GAC-C	TA-GTATCCCTC---TCTATATAT
Species b	ATACTAGTTTT-ATC-AAAGC-GAC-C	TA-GTAC---TCGGATCT--ATAT
Species c	ATATTAG-TC-GATCTACAGCTGAC-C	TAGGTACCCCTCGGATCCATAT-T
Species d	ACCCTAGTTTCGGATCCAAGC-GAC-C	TA-GTATCCCTC---TCTATATCT
Species e	ACC--TG-TCC-ATCTATG-CTGACTC	TA-GTATCCCTCAGA-CTATAT-A

Gene trees

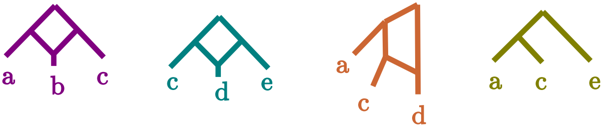
Species network



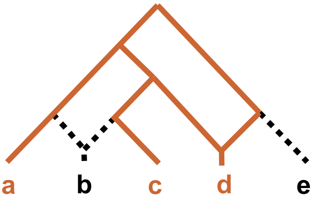
PART 2: NETWORKS FROM SUBNETS

Species a	ACCCTAG--TC--ATC---AGC-GAC-CTA-GTACCCTC---TCTATATAT
Species b	ATACTAGTTTT--ATC-AAAGC-GAC-CTA-GTA---TCGGATCT--ATAT
Species c	ATATTAG--TC-GATCTACAGC-GAC-CTAGGTACCCTCGGATCCATAT-T
Species d	ACCCTAGTTTCGGATCCCAAGC-GAC-CTA-GTACCCTC---TCTATATCT
Species e	ACC--TG--TCC-ATCT--AGC-GAC-CTA-GTACCCTCAGA-CTATAT-A

Trinets

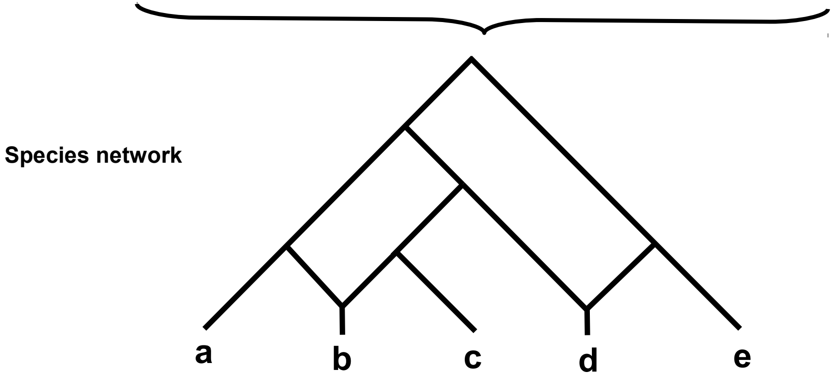


Species network



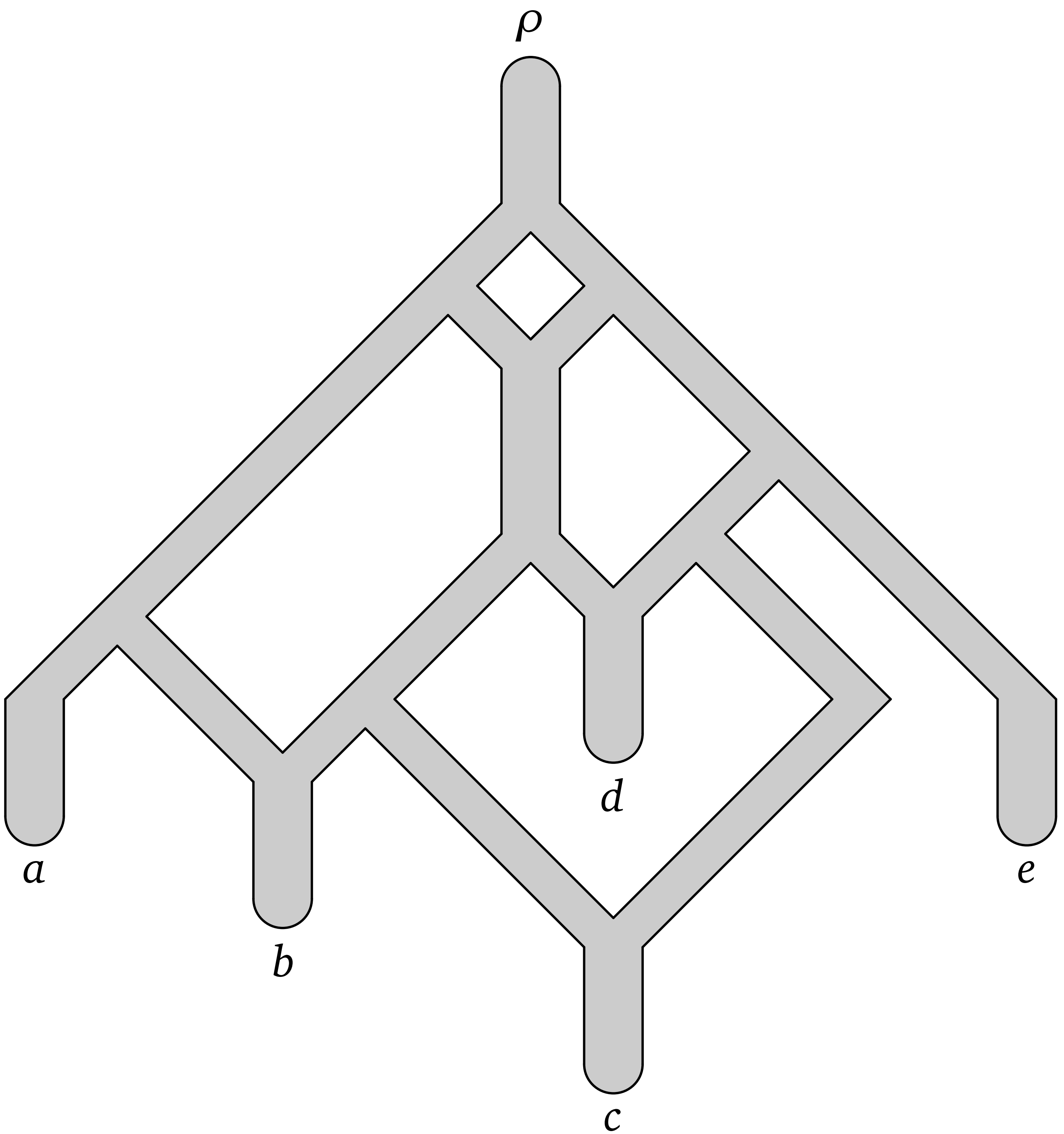
PART 3: NETWORKS FROM SEQUENCES

```
Species a   ACCCTAG--TC--ATC---AGC-GAC-CTA-GTACCCTC---TCTATATAT
Species b   ATACTAGTTTT--ATC-AAAGC-GAC-CTA-GTA---TCGGATCT--ATAT
Species c   ATATTAG--TC-GATCTACAGC-GAC-CTAGGTACCCTCGGATCCATAT-T
Species d   ACCCTAGTTTCGGATCCCAAGC-GAC-CTA-GTACCCTC---TCTATATCT
Species e   ACC--TG--TCC-ATCT--AGC-GAC-CTA-GTACCCTCAGA-CTATAT-A
```

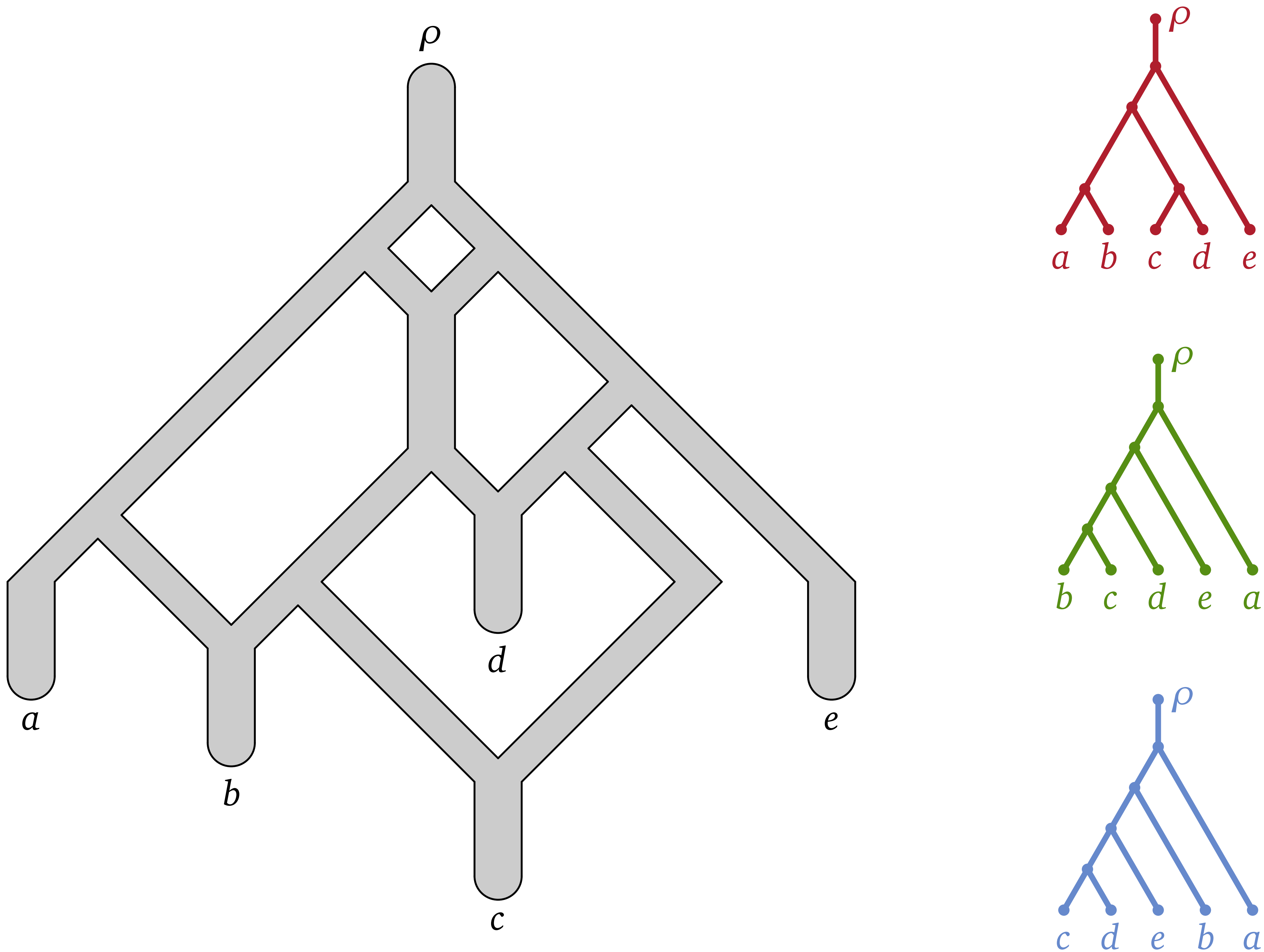


PART 1:
NETWORKS FROM TREES

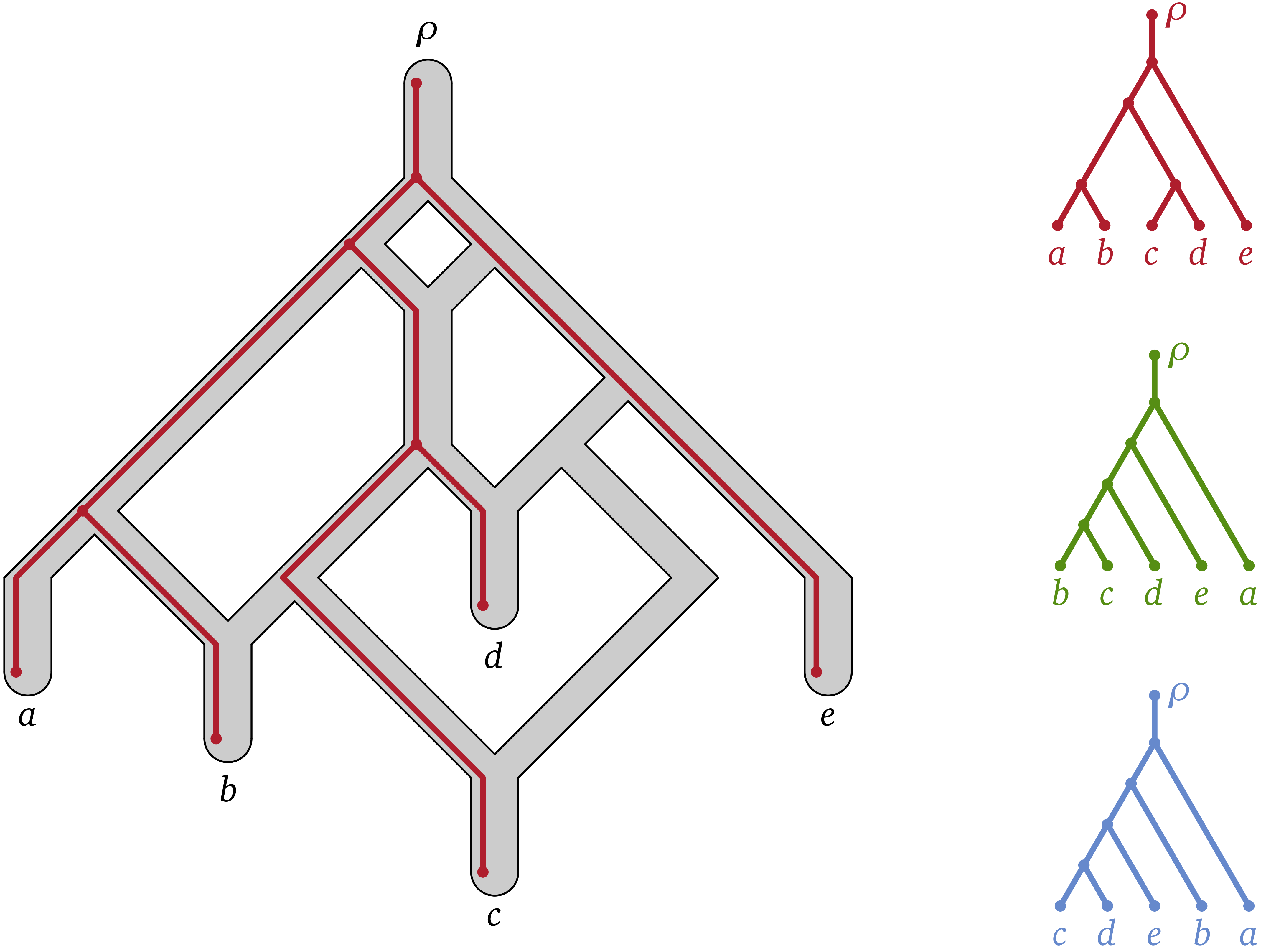
Tree-based Network Reconstruction



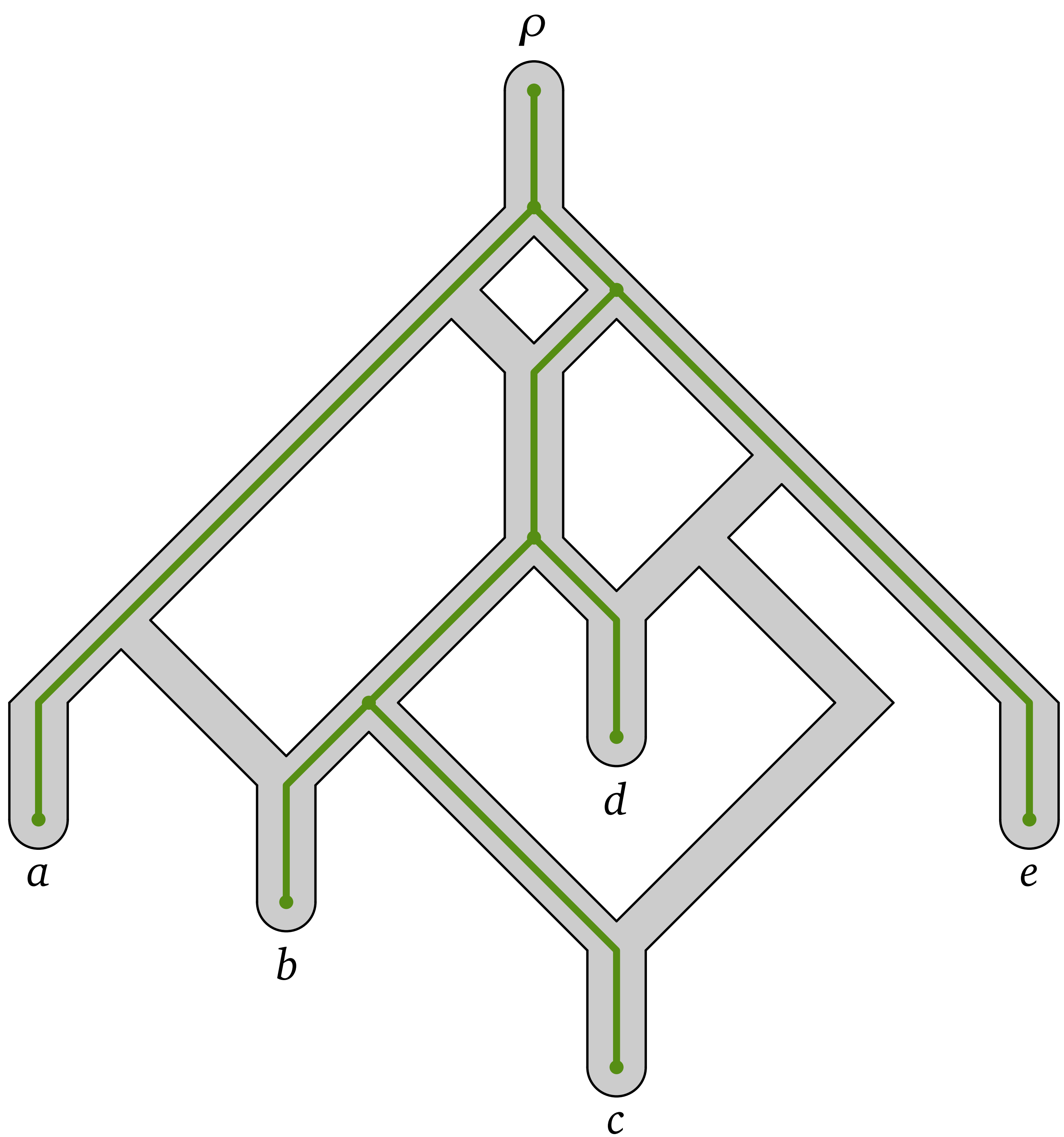
Tree-based Network Reconstruction

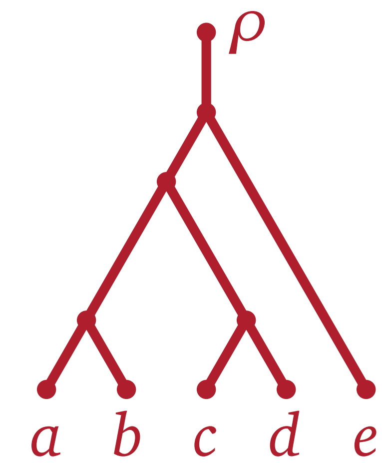


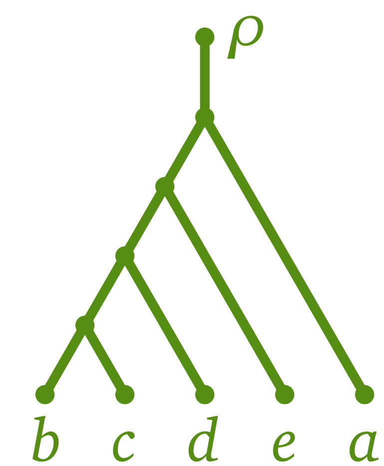
Tree-based Network Reconstruction

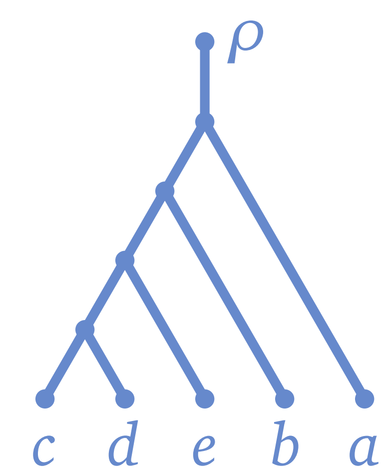


Tree-based Network Reconstruction

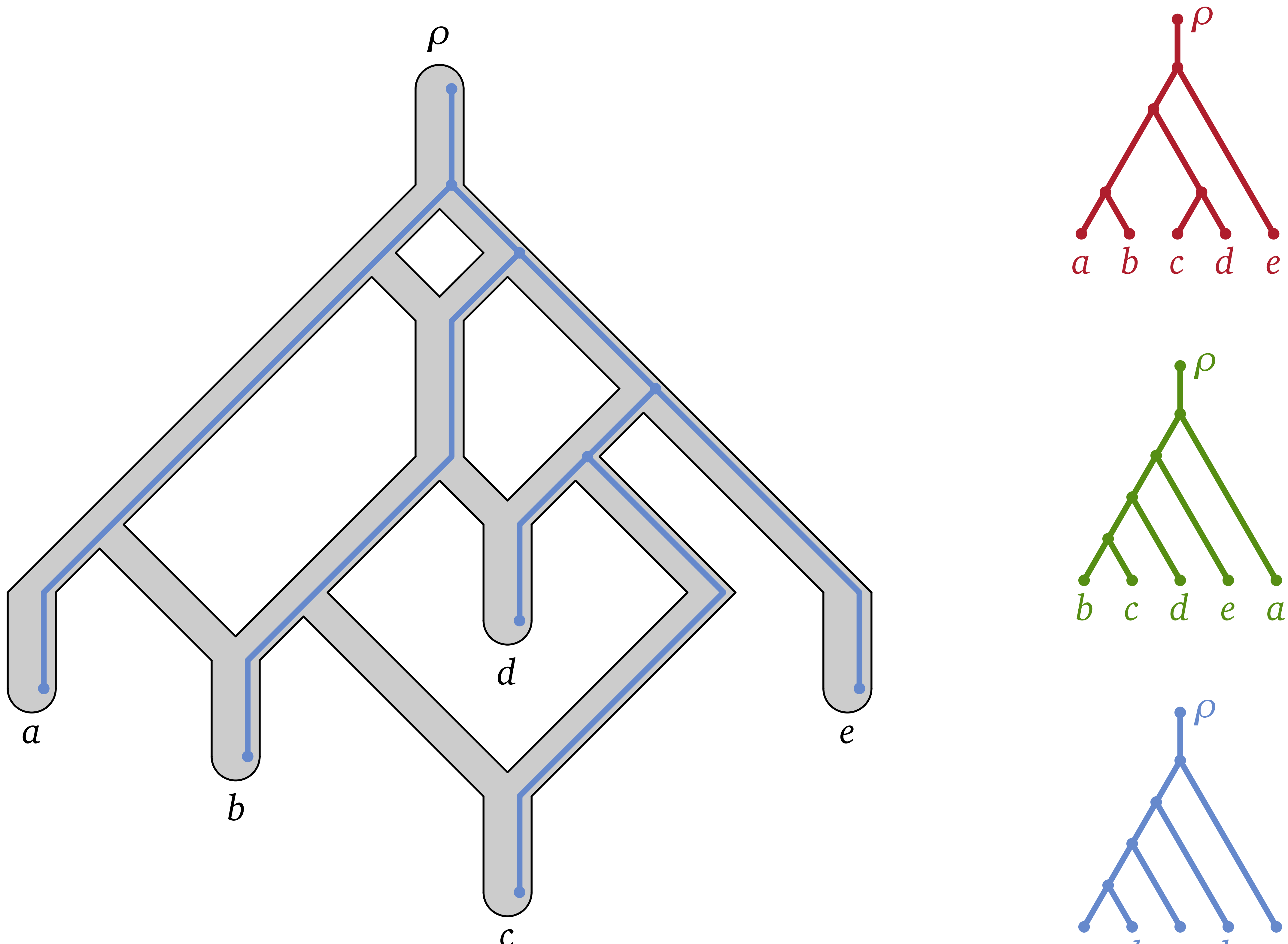




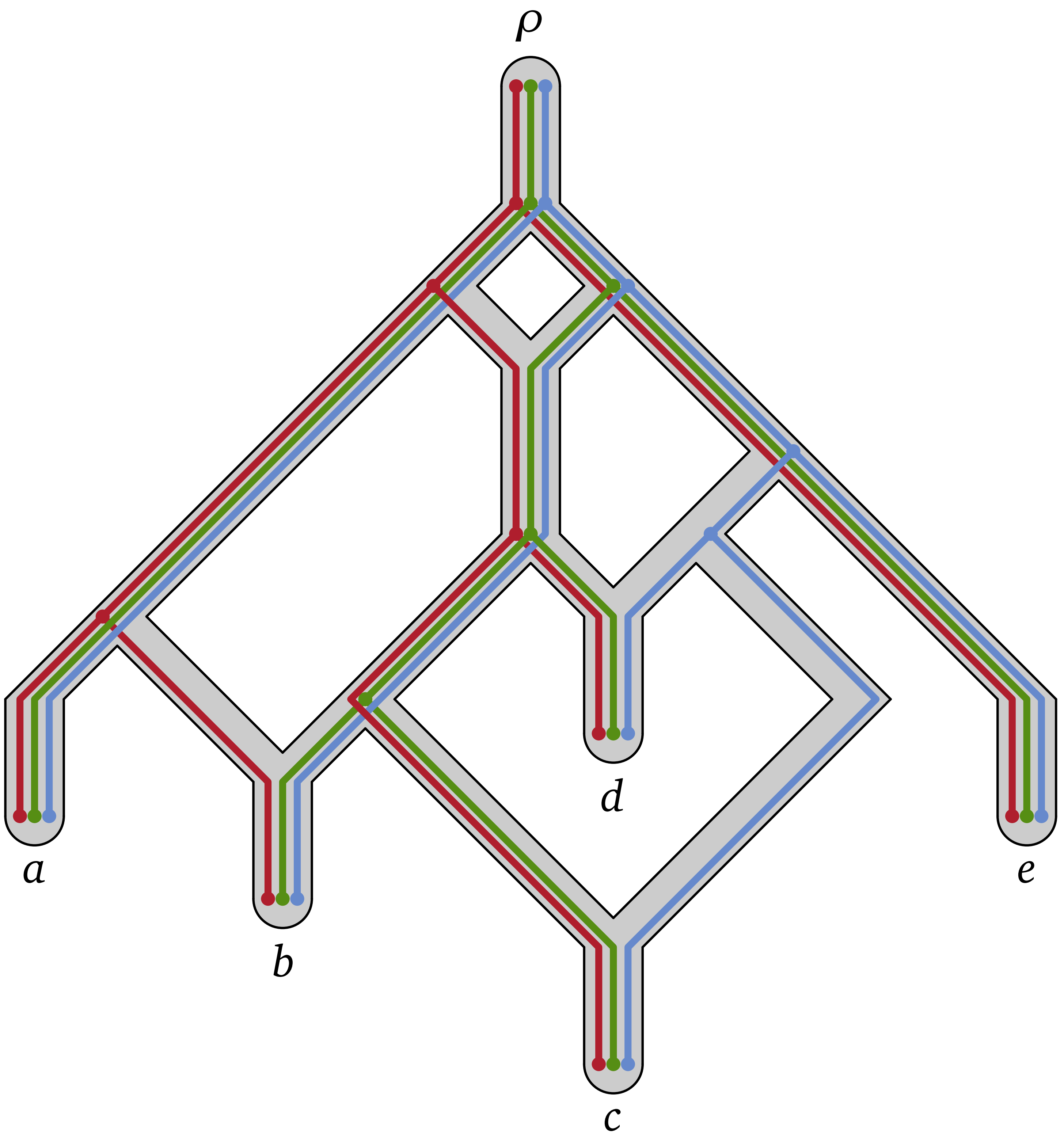


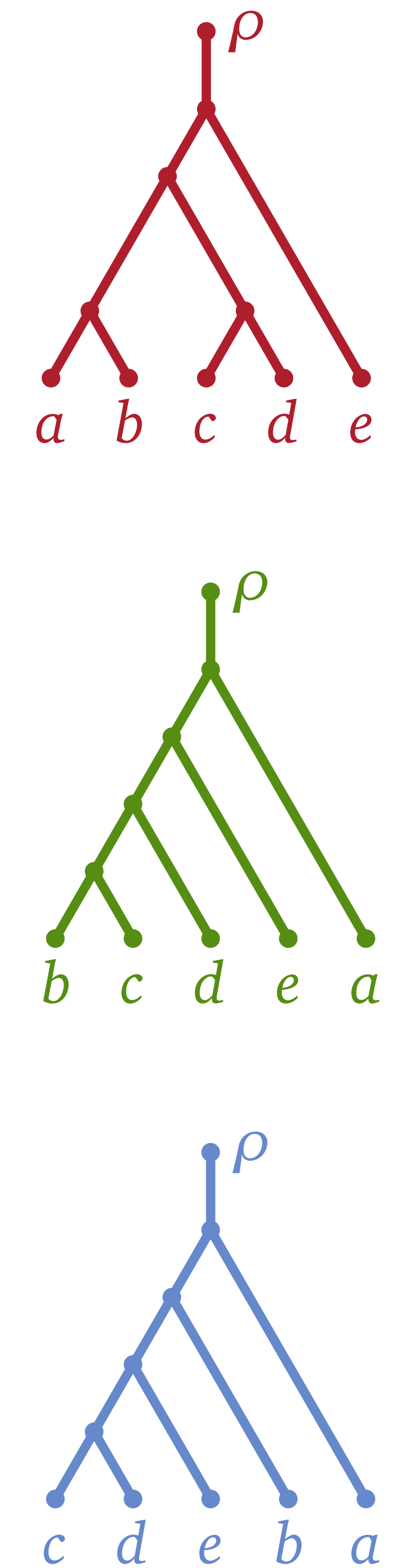


Tree-based Network Reconstruction



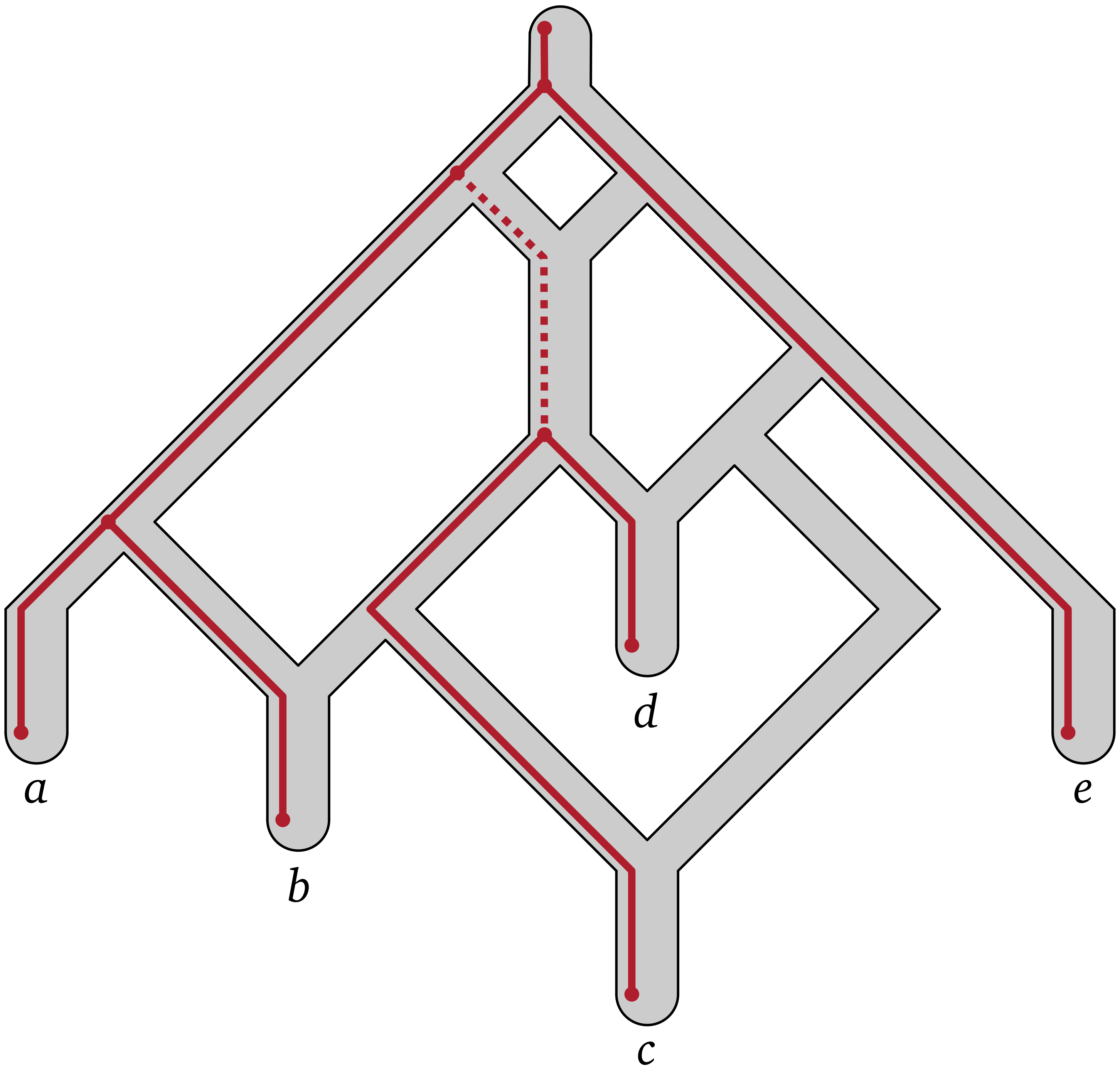
Tree-based Network Reconstruction

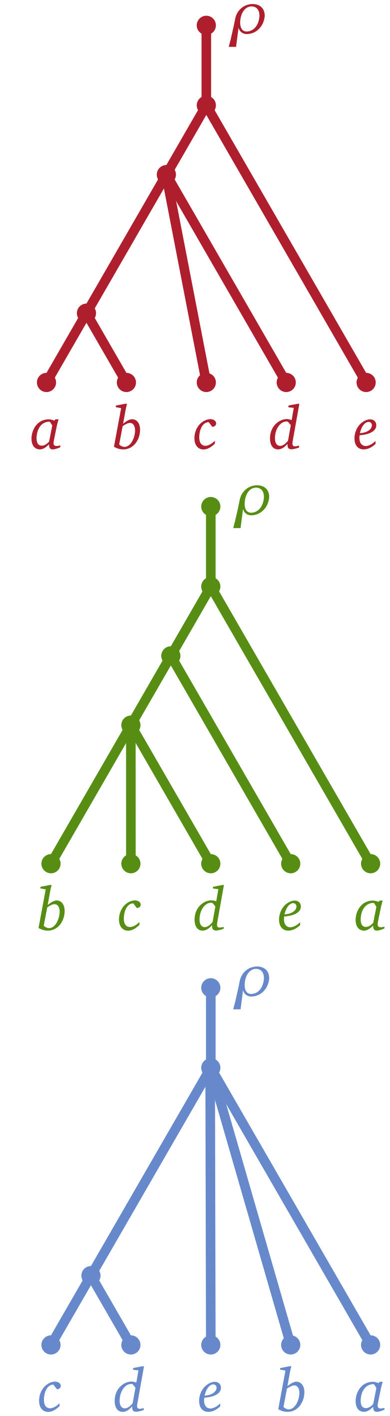




Nonbinary Trees

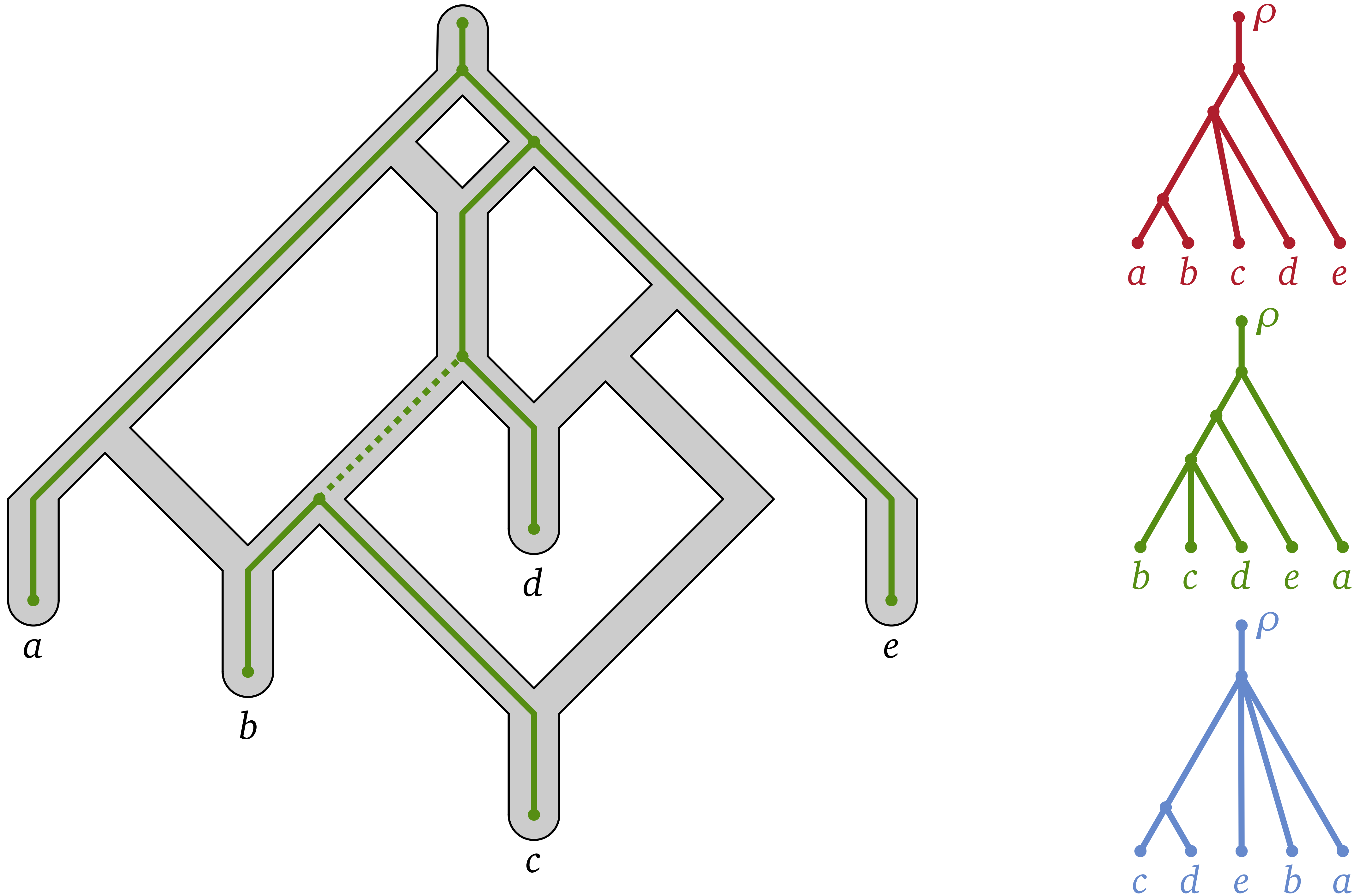
Definition. A phylogenetic tree T is ***displayed*** by a phylogenetic network N if T can be obtained from a subgraph of N by contracting edges.





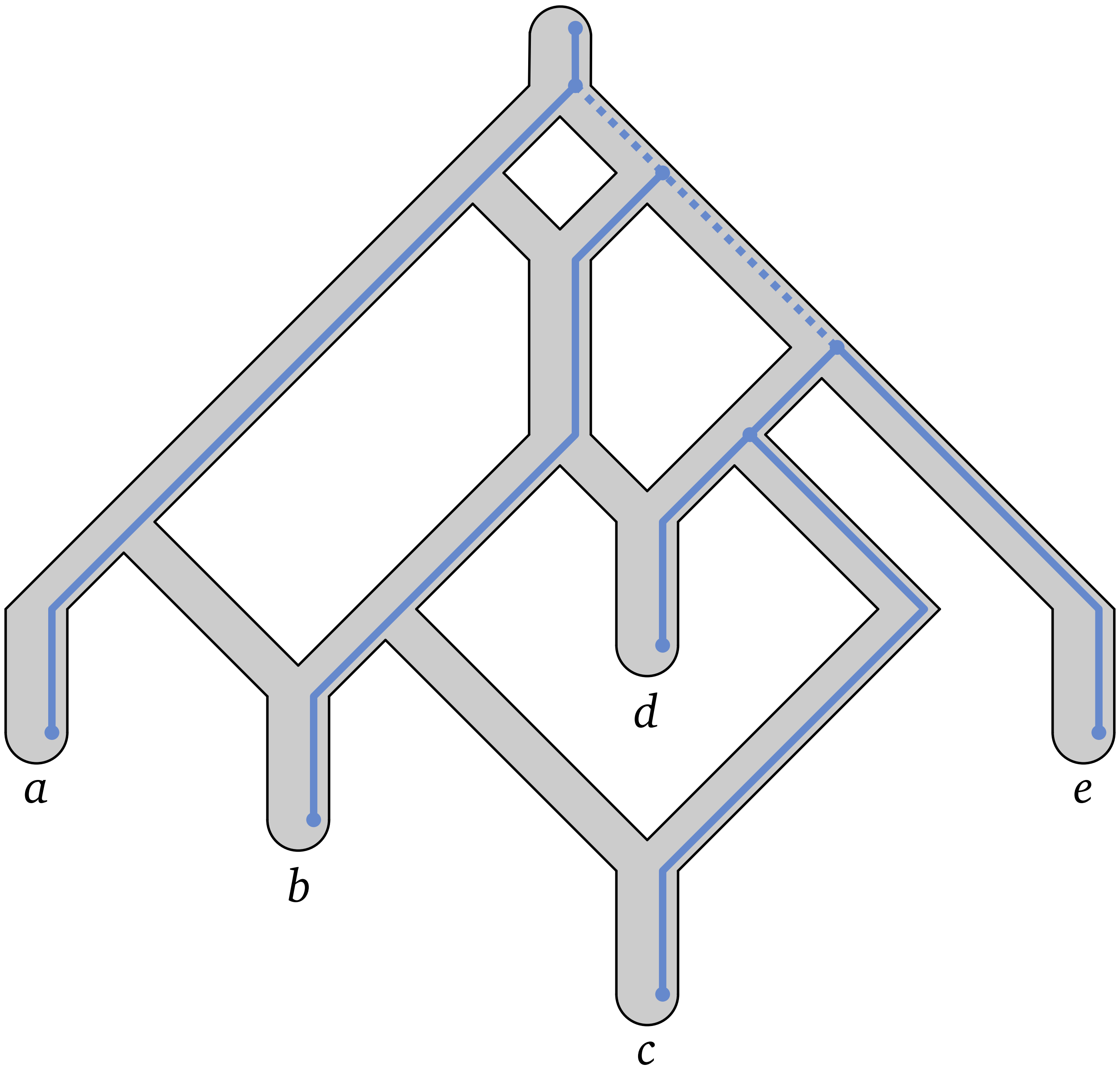
Nonbinary Trees

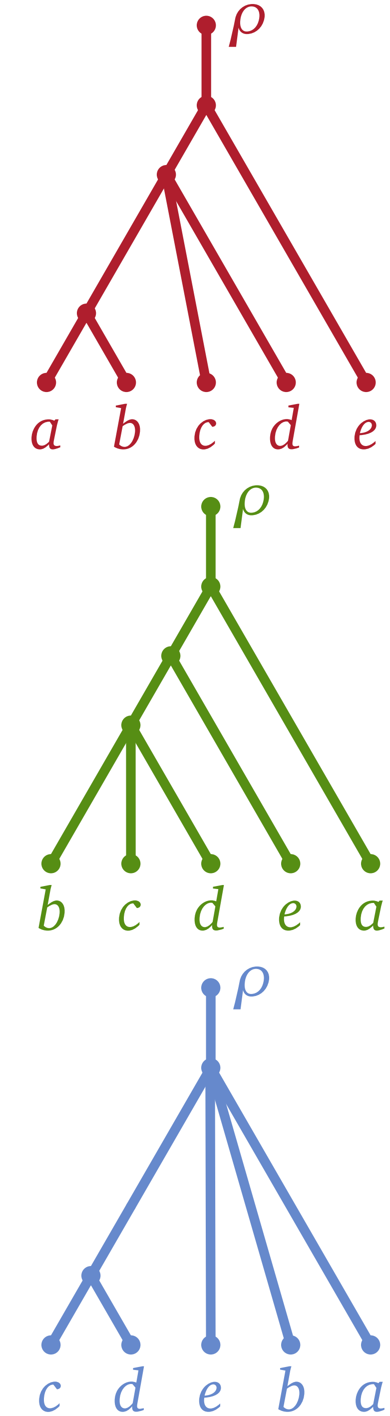
Definition. A phylogenetic tree T is ***displayed*** by a phylogenetic network N if T can be obtained from a subgraph of N by contracting edges.



Nonbinary Trees

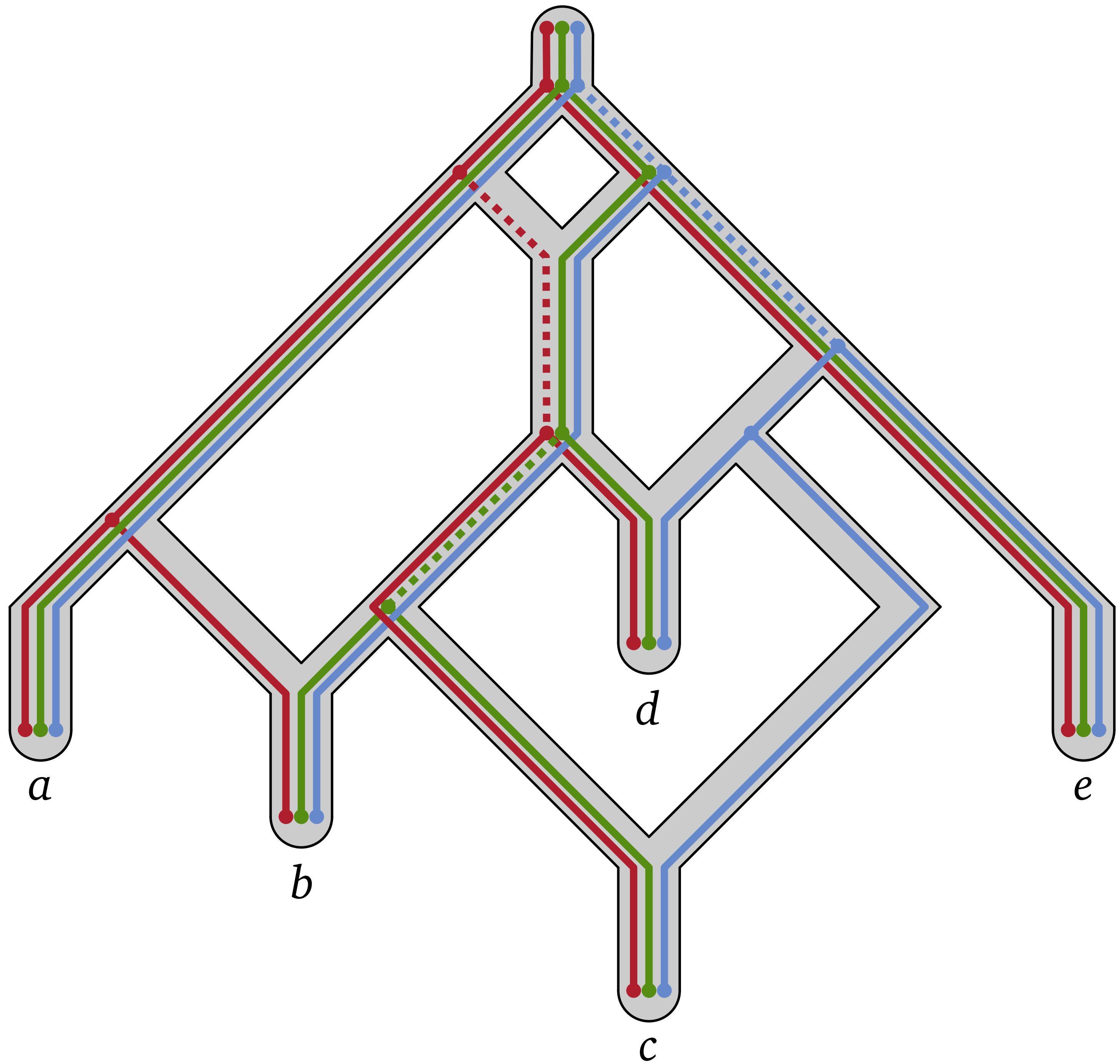
Definition. A phylogenetic tree T is ***displayed*** by a phylogenetic network N if T can be obtained from a subgraph of N by contracting edges.

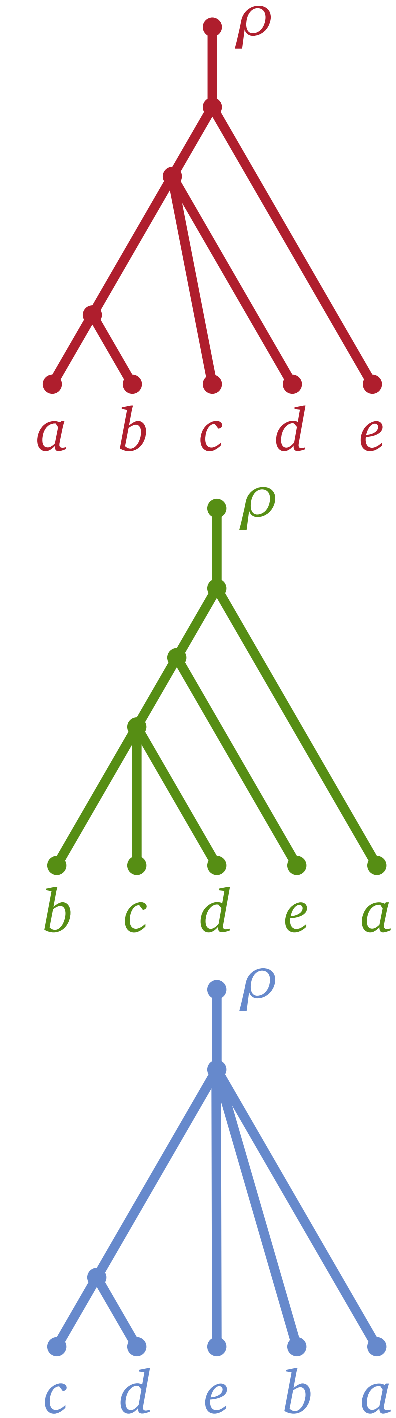




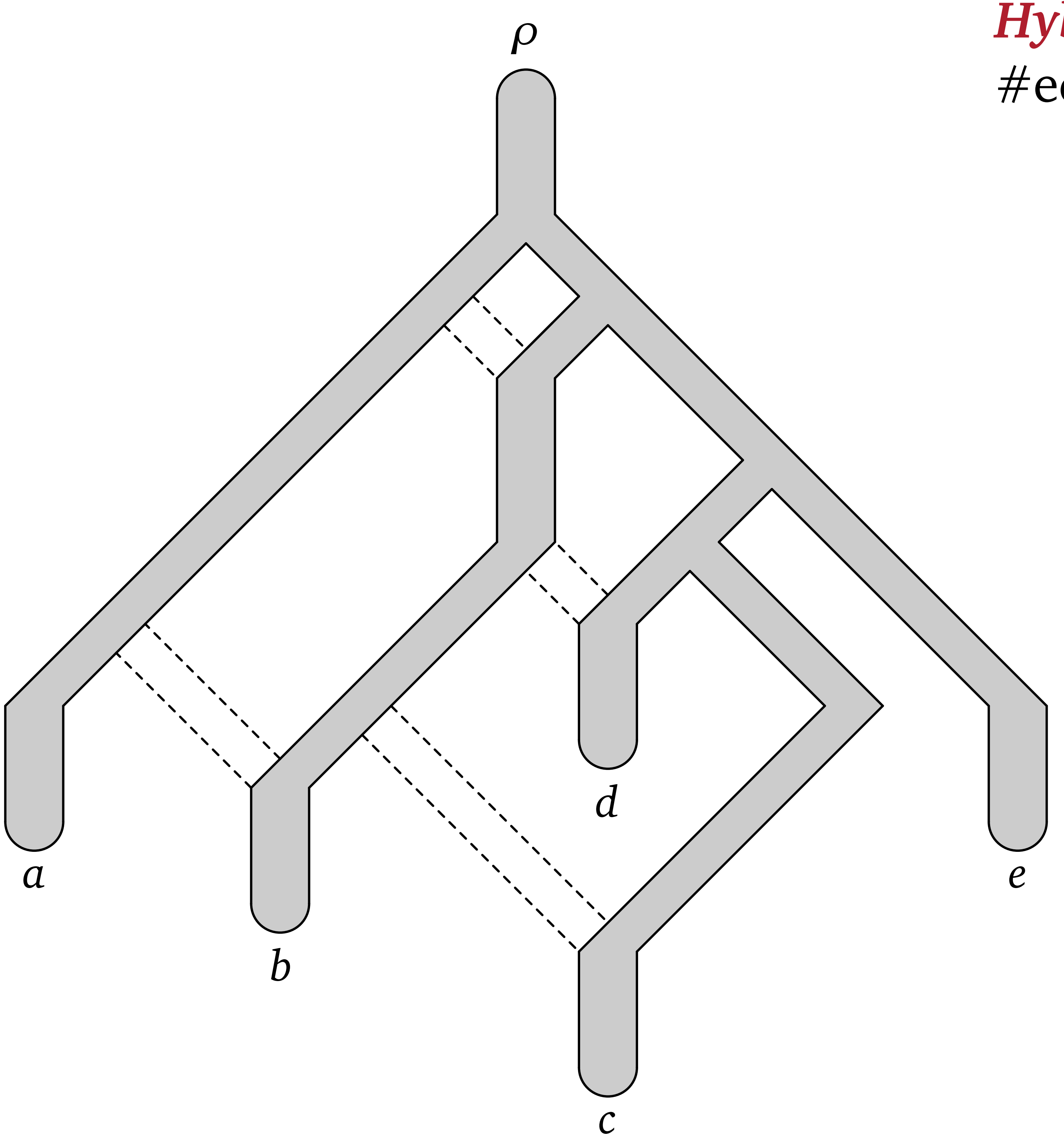
Nonbinary Trees

Definition. A phylogenetic tree T is ***displayed*** by a phylogenetic network N if T can be obtained from a subgraph of N by contracting edges.





Tree-based Network Reconstruction



Hybridization number:
#edges to cut to obtain a tree

Results

Problem: HYBRIDIZATION NUMBER
Given: Collection of phylogenetic trees $\mathscr{T}$, each on the same n leaves, $k \in \mathbb{N}$
Question: Does there exist a phylogenetic network that displays each tree in $\mathscr{T}$ and has hybridization number at most k?

Two binary trees:

- Direct relationship to ***maximum acyclic agreement forest*** (MAAF)
- $O((28k)^k + n^3)$-time algorithm (Bordewich & Semple 2007)
- $O(3.18^k n)$- time algorithm (Whidden, Beiko & Zeh, 2013)
- Same approximability as ***directed feedback vertex set*** (Kelk, vI, Lekic, Linz, Scornavacca, Stougie, 2012)

Any number of nonbinary trees: (vI, Kelk & Scornavacca, 2014)

- Kernel with $4k(5k)^t$ leaves, with t the number of trees
- Kernel with $20k^2(\Delta^+ - 1)$ leaves, with Δ^+ the maximum outdegree
- $n^{f(k)}t$-time bounded-search algorithm, with f astronomical

Three binary trees:

- $c^k \text{poly}(n)$ time algorithm (vI, Lekic, Kelk, Whidden & Zeh, 2014)

Results

Problem: Hybridization Number
Given: Collection of phylogenetic trees $\mathcal{T}$, each on the same n leaves, $k \in \mathbb{N}$
Question: Does there exist a phylogenetic network that displays each tree in $\mathcal{T}$ and has hybridization number at most k?

Two binary trees:

- Direct relationship to ***maximum acyclic agreement forest*** (MAAF)
- $O((28k)^k + n^3)$-time algorithm (Bordewich & Semple 2007)
- $O(3.18^k n)$- time algorithm (Whidden, Beiko & Zeh, 2013)
- Same approximability as ***directed feedback vertex set*** (Kelk, vI, Lekic, Linz, Scornavacca, Stougie, 2012)

Any number of nonbinary trees: (vI, Kelk & Scornavacca, 2014)

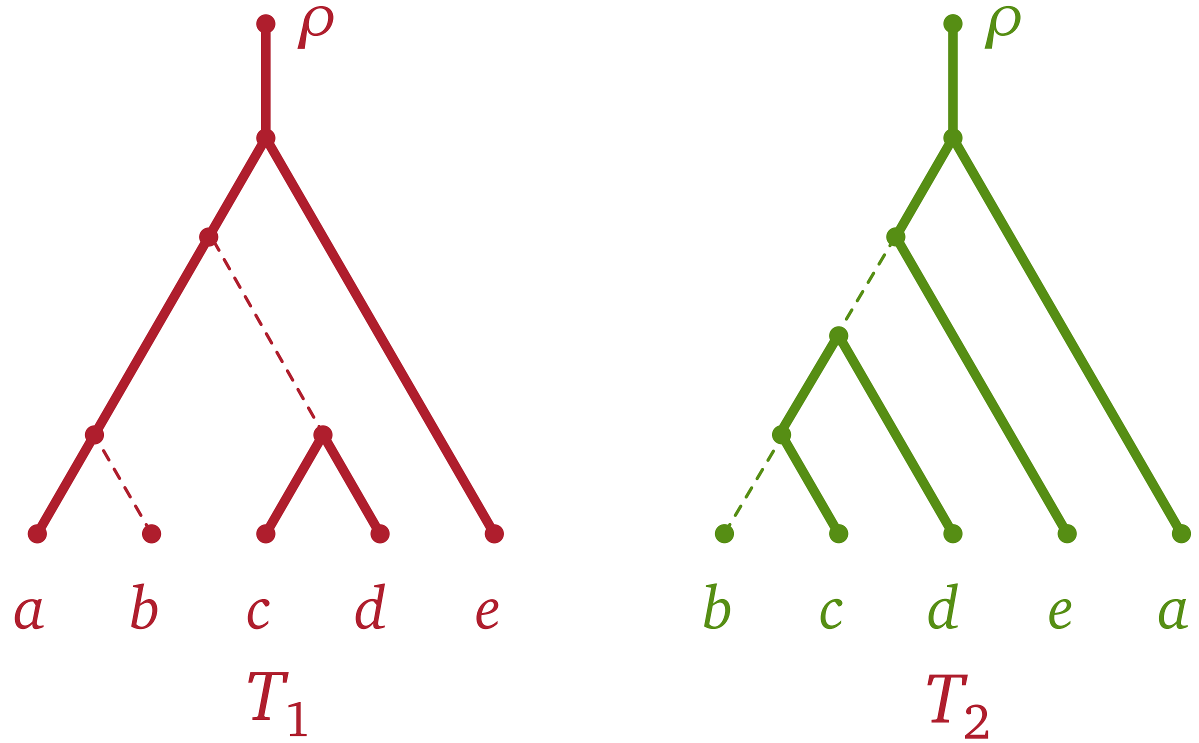
- Kernel with $4k(5k)^t$ leaves, with t the number of trees
- Kernel with $20k^2(\Delta^+ - 1)$ leaves, with Δ^+ the maximum outdegree
- $n^{f(k)}t$-time bounded-search algorithm, with f astronomical

Three binary trees:

- $c^k\text{poly}(n)$ time algorithm (vI, Lekic, Kelk, Whidden & Zeh, 2014) ($c = 1609891840$)

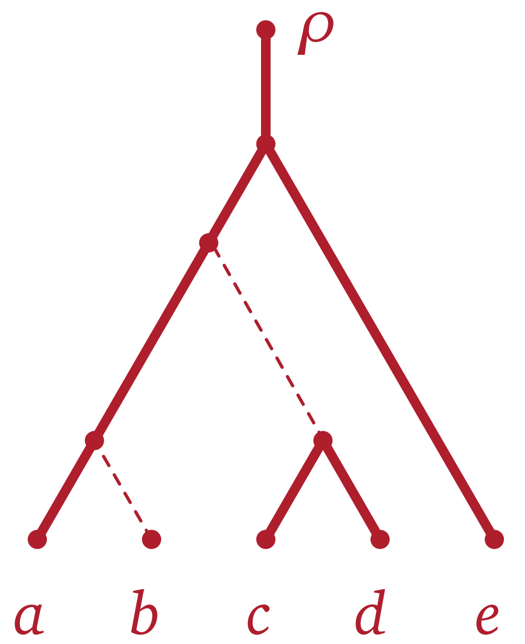
Agreement Forests

An ***agreement forest*** of two binary trees is a forest that can be obtained from either tree by deleting edges and unlabelled vertices and suppressing indegree-1 outdegree-1 vertices

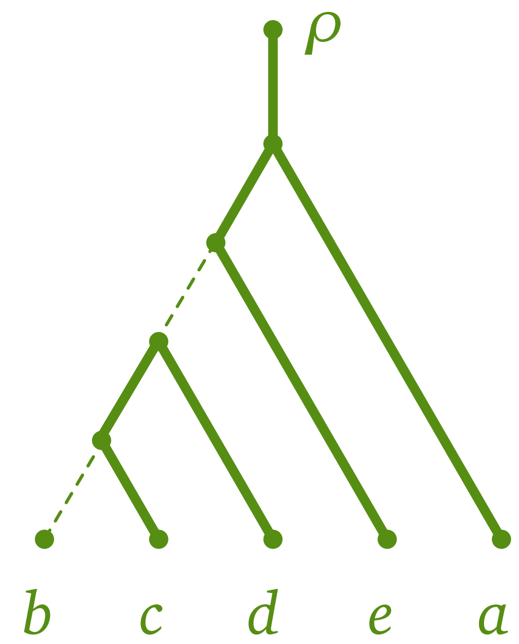


Agreement Forests

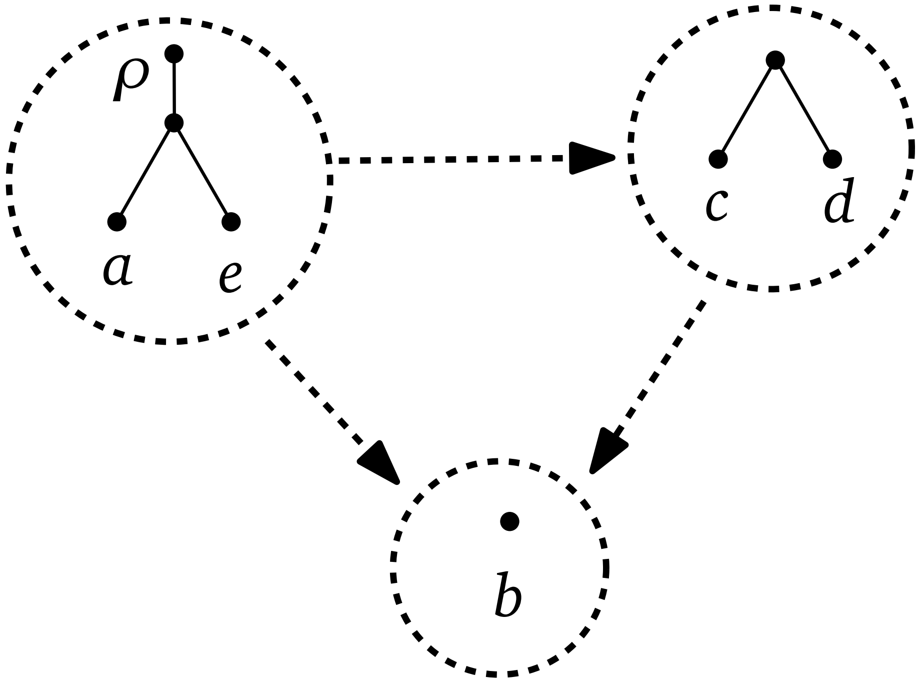
An ***agreement forest*** of two binary trees is a forest that can be obtained from either tree by deleting edges and unlabelled vertices and suppressing indegree-1 outdegree-1 vertices



T_1



T_2

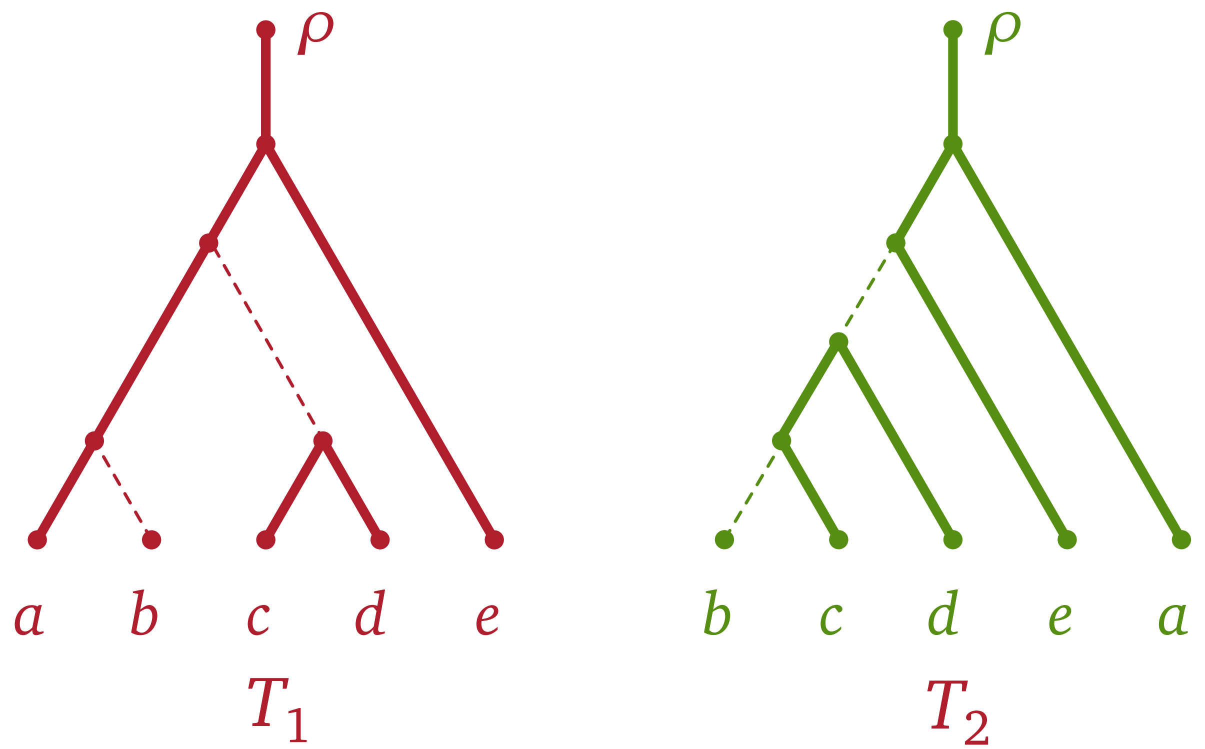


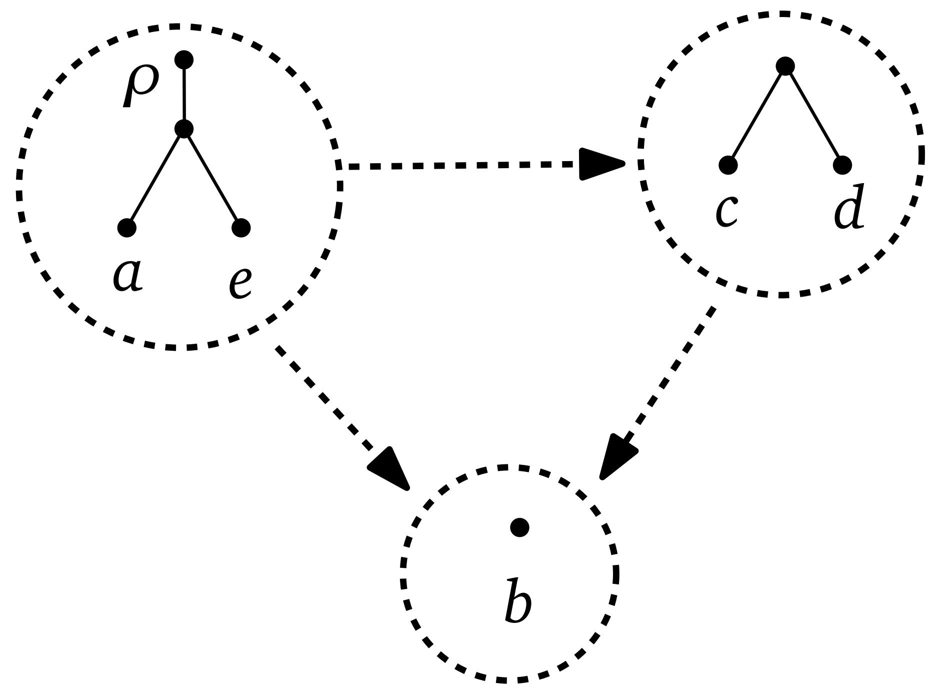
Inheritance Graph

An agreement forest is ***acyclic*** if its inheritance graph is acyclic
An acyclic agreement forest with a minimum number of components is called a ***Maximum Acyclic Agreement Forest (MAAF)***

Agreement Forests

An ***agreement forest*** of two binary trees is a forest that can be obtained from either tree by deleting edges and unlabelled vertices and suppressing indegree-1 outdegree-1 vertices



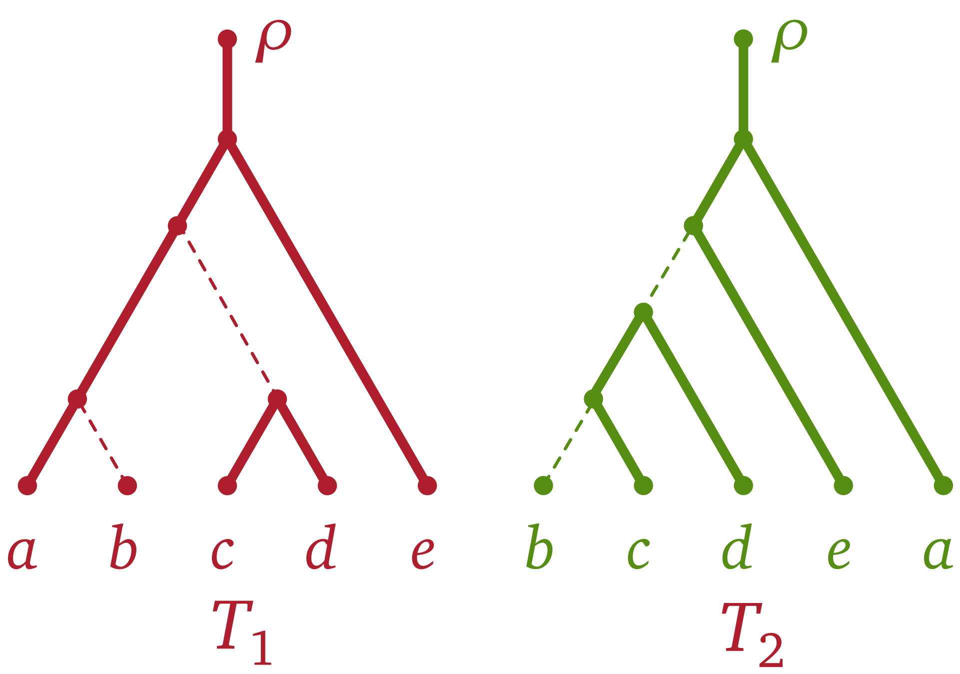


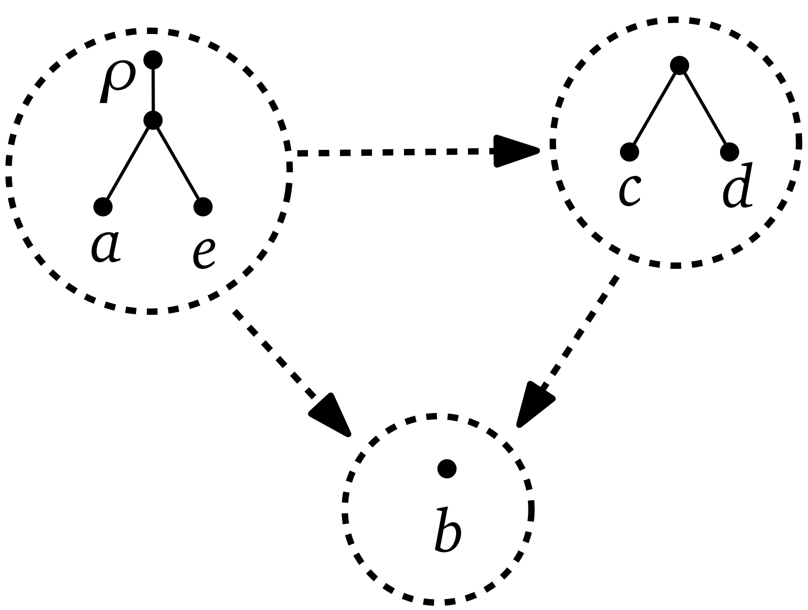
Inheritance Graph

An agreement forest is ***acyclic*** if its inheritance graph is acyclic
An acyclic agreement forest with a minimum number of components is called a ***Maximum Acyclic Agreement Forest (MAAF)***

For **two binary trees:** HYBRIDIZATION NUMBER = |MAAF| - 1
(Bordewich & Semple 2007)

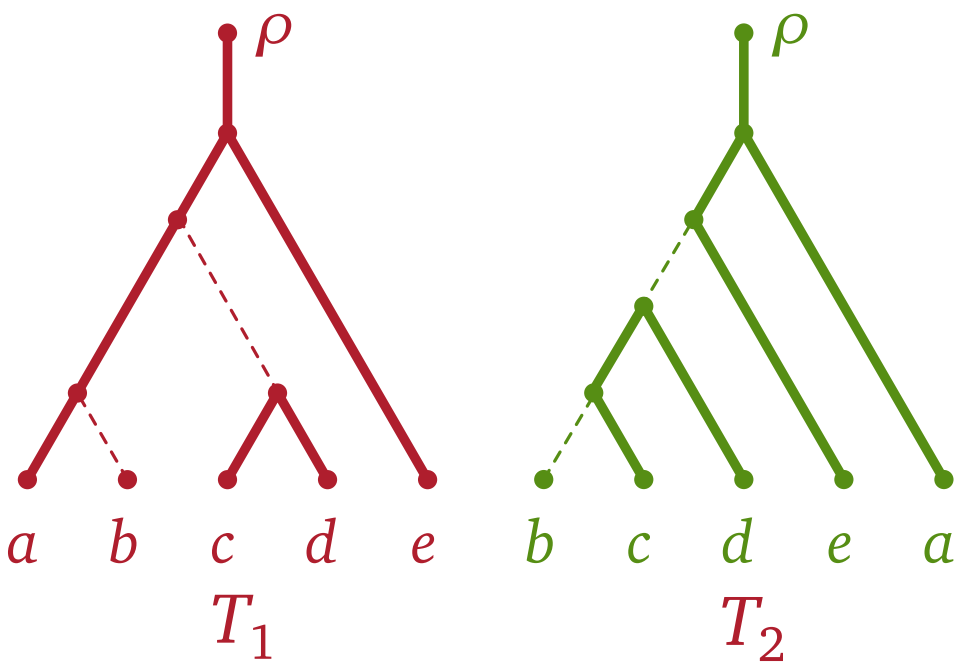
Agreement Forests vs Hybridization Networks

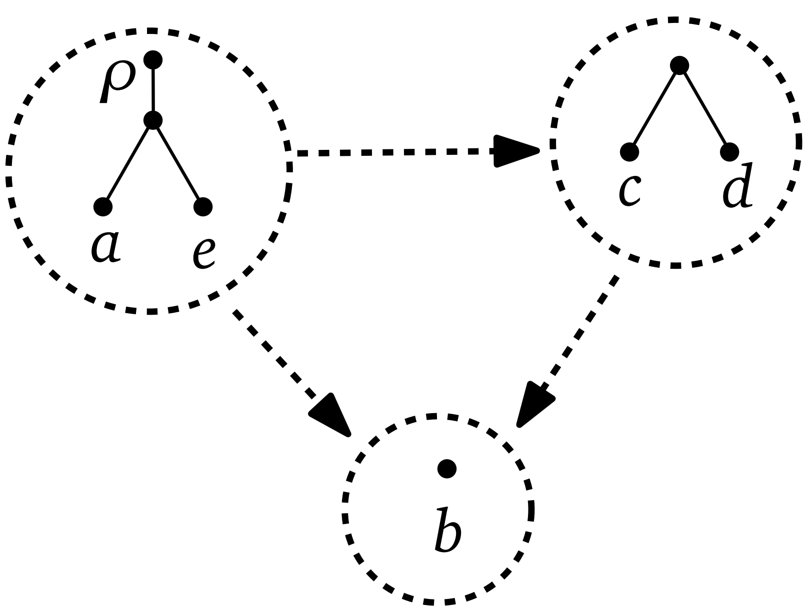




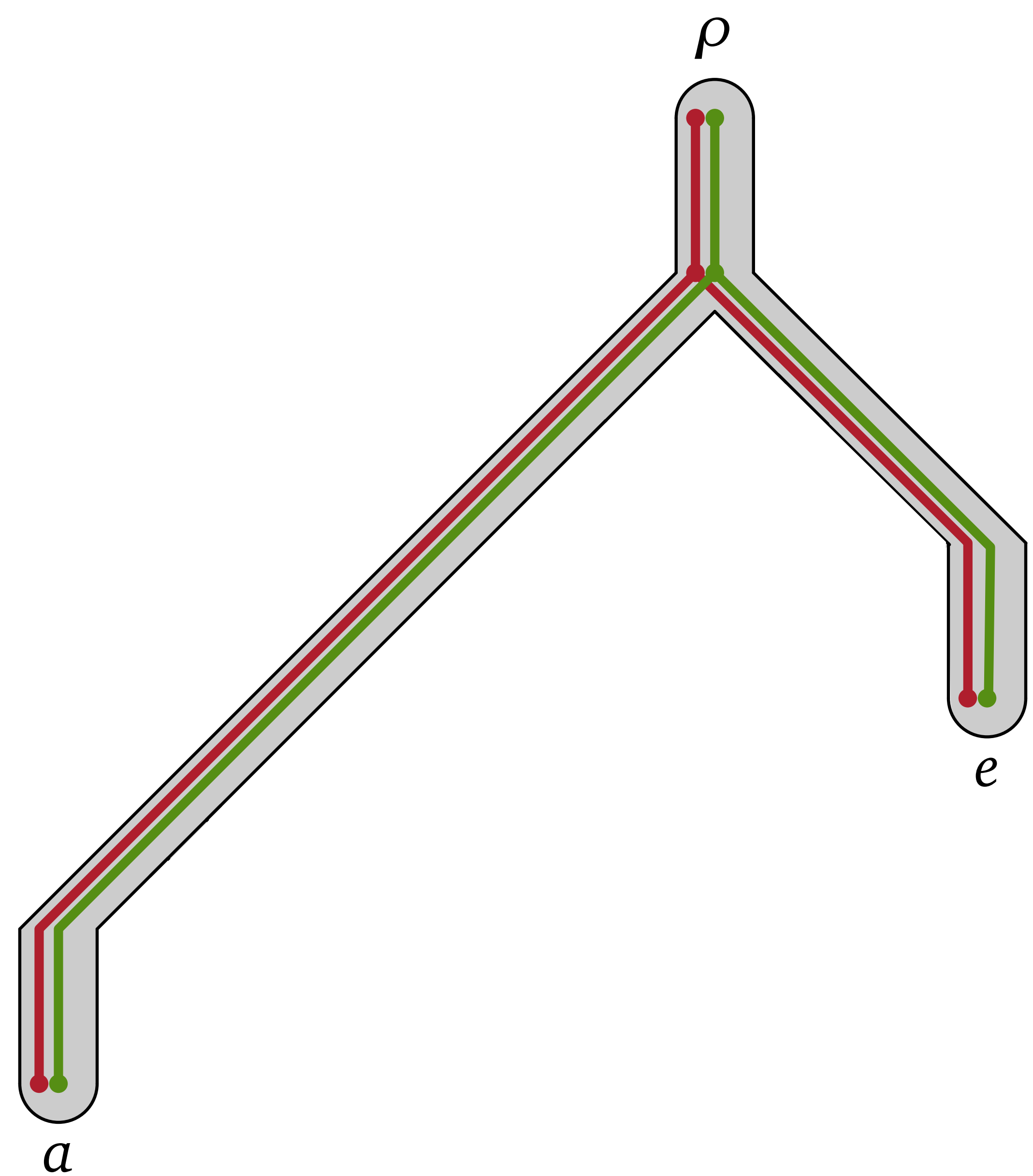
Inheritance Graph

Agreement Forests vs Hybridization Networks

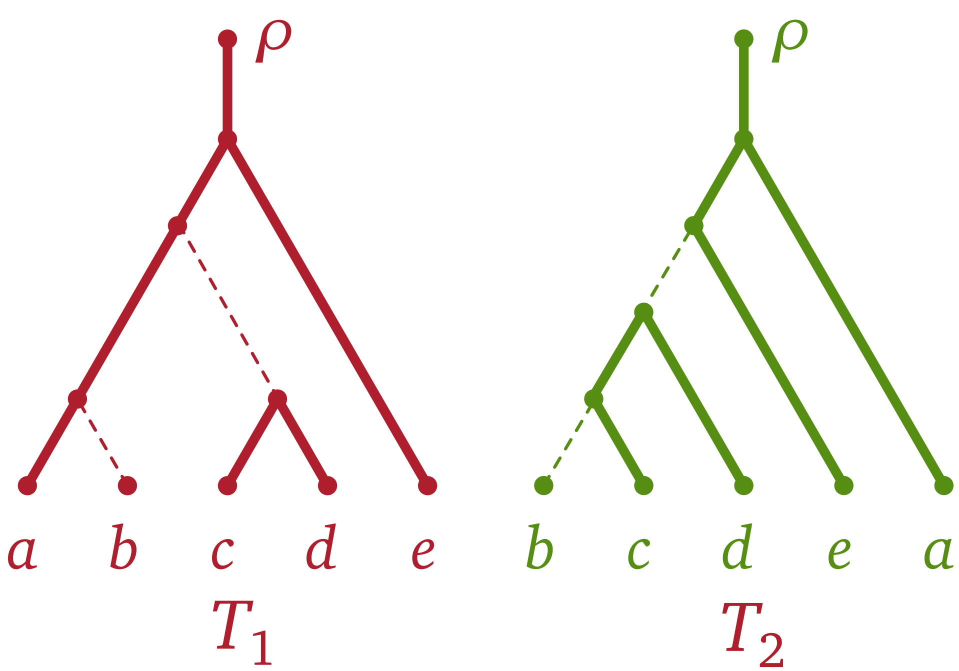


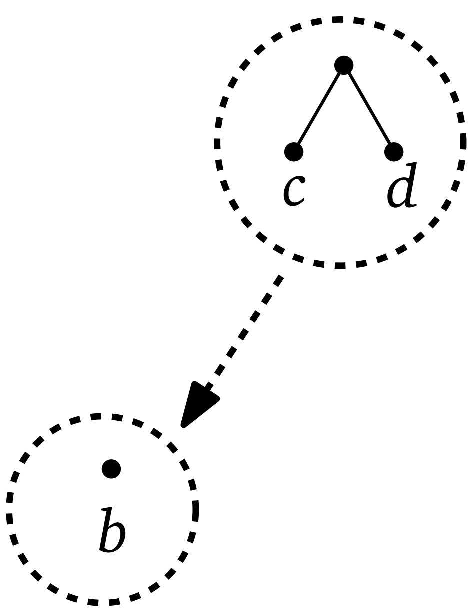


Inheritance Graph

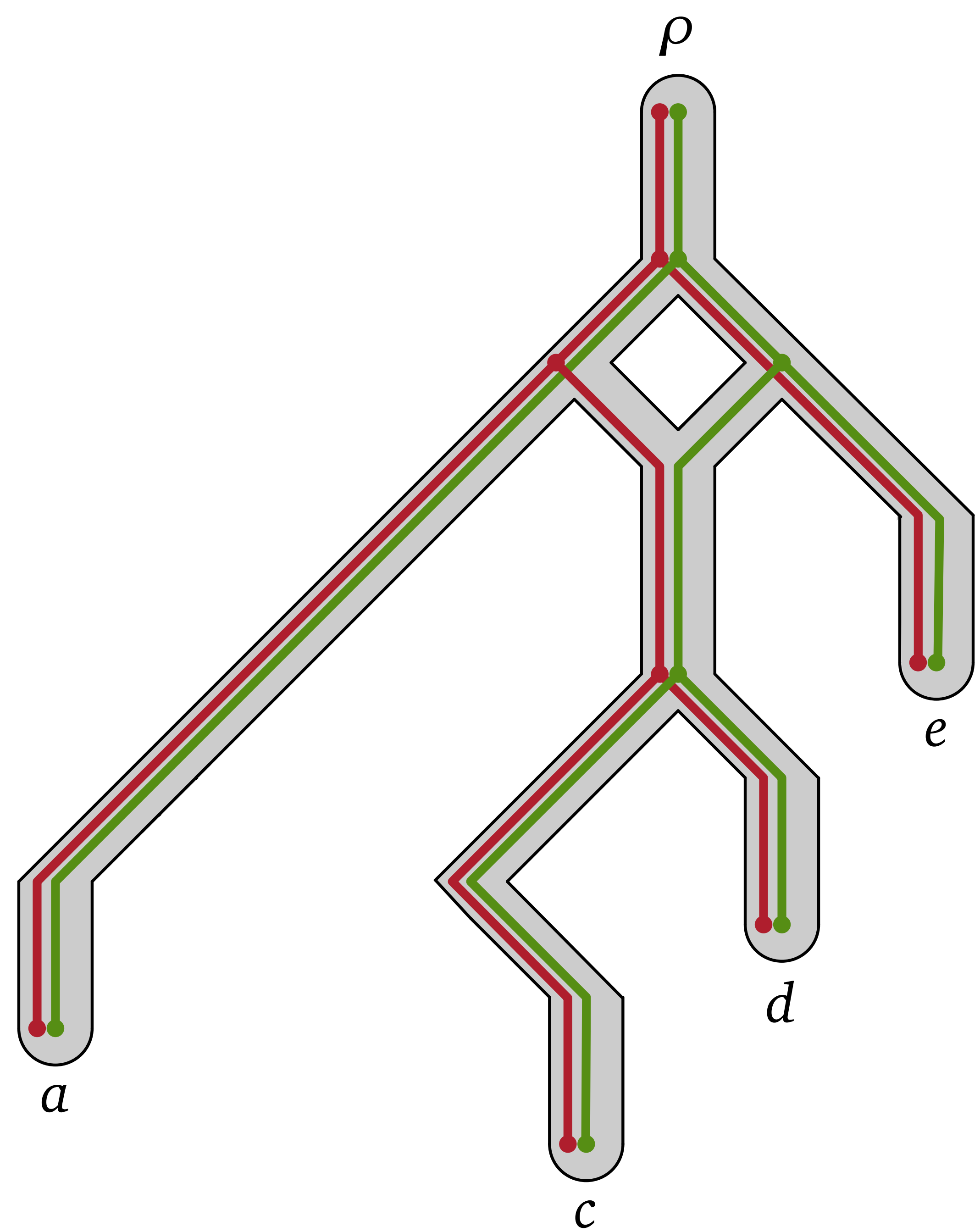


Agreement Forests vs Hybridization Networks

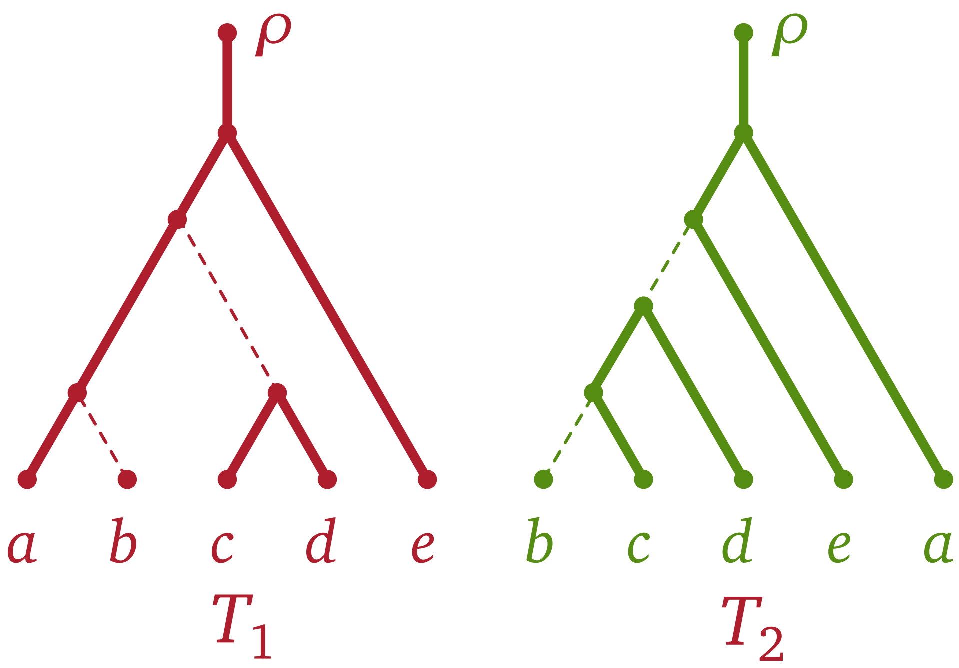


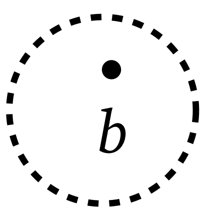


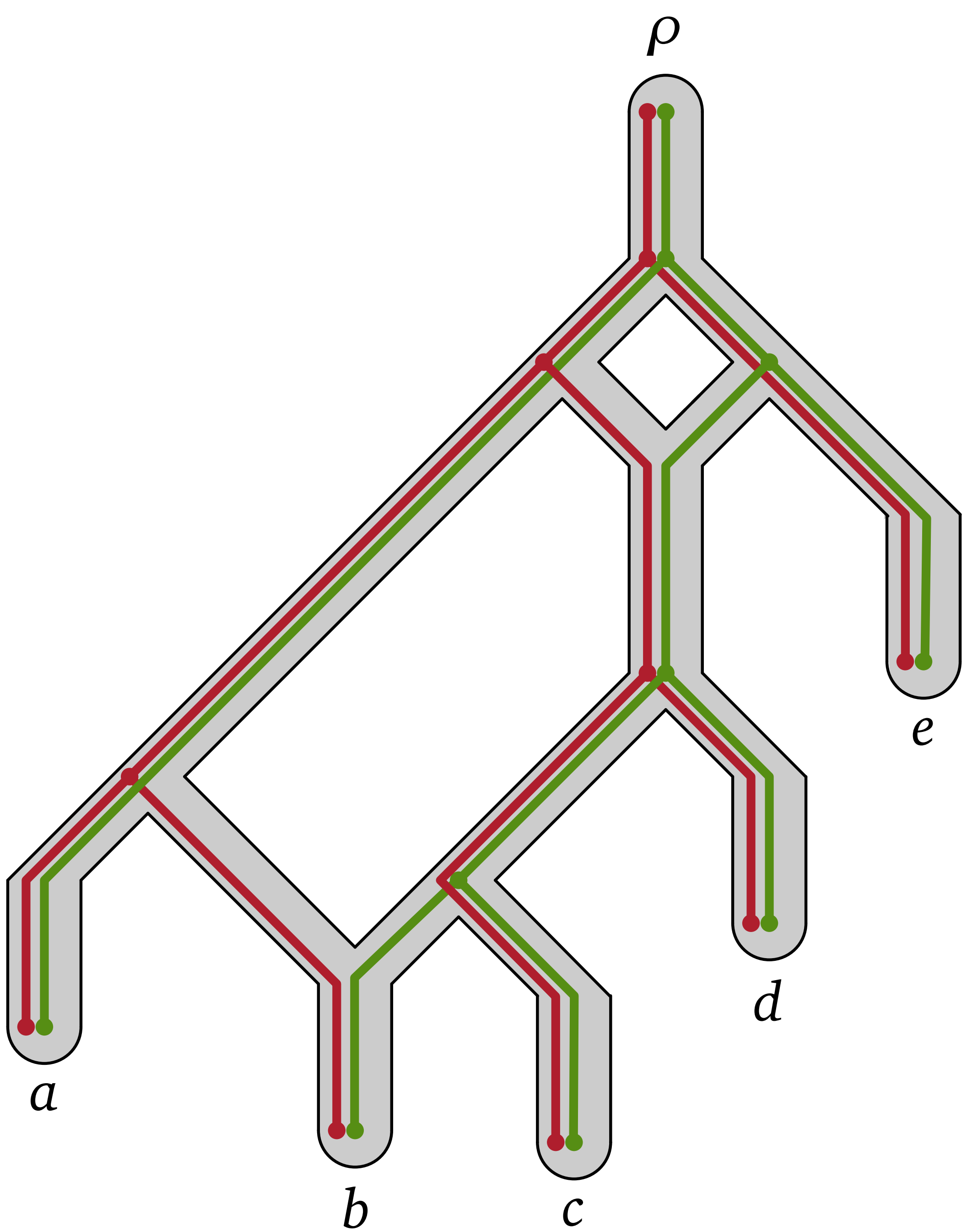
Inheritance Graph



Agreement Forests vs Hybridization Networks

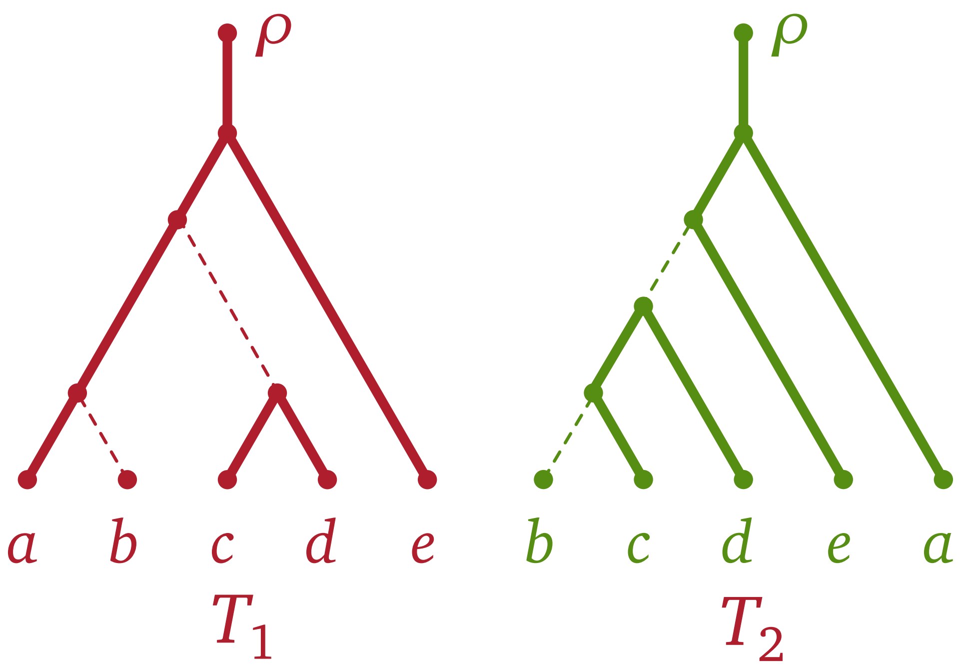


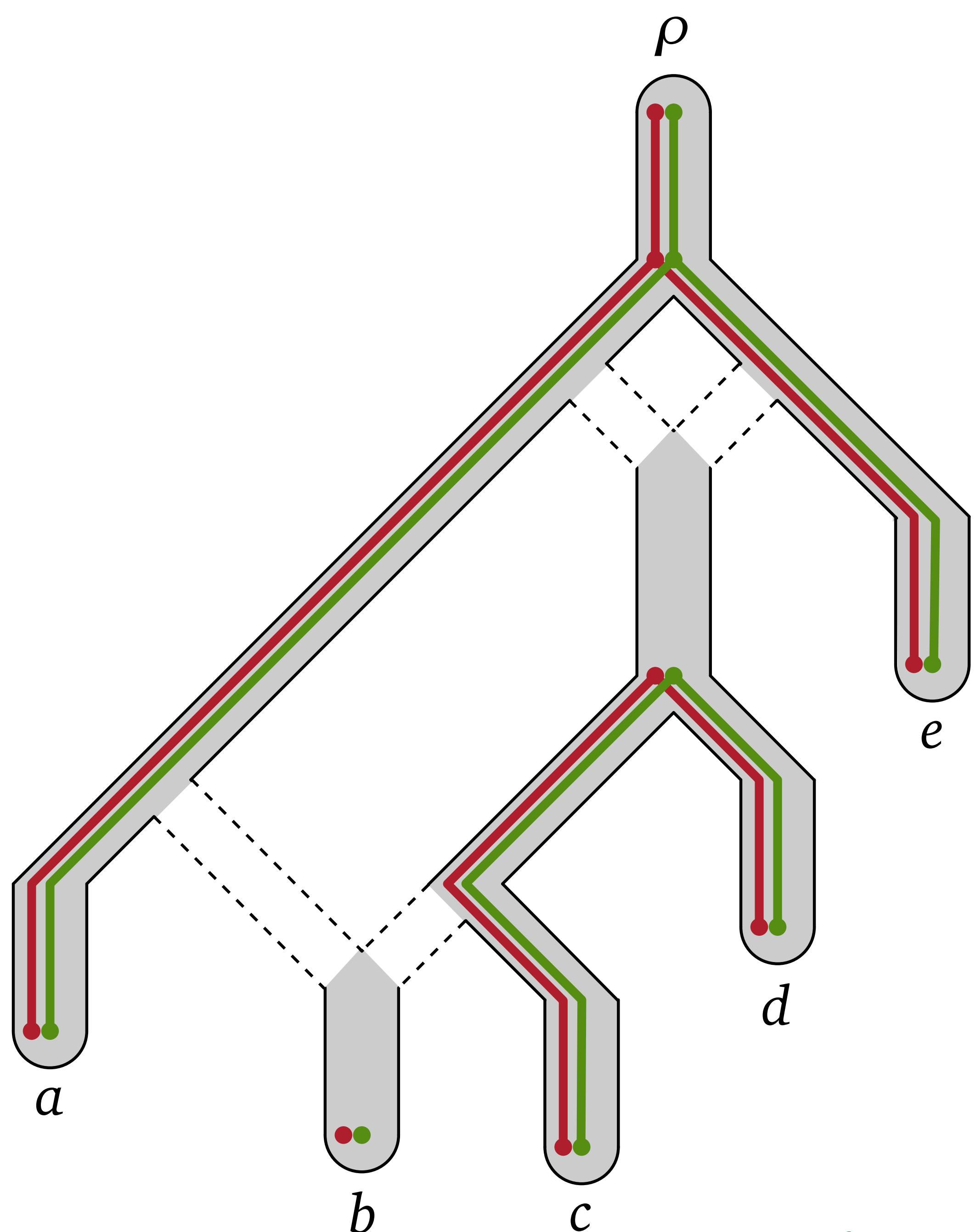




Inheritance Graph

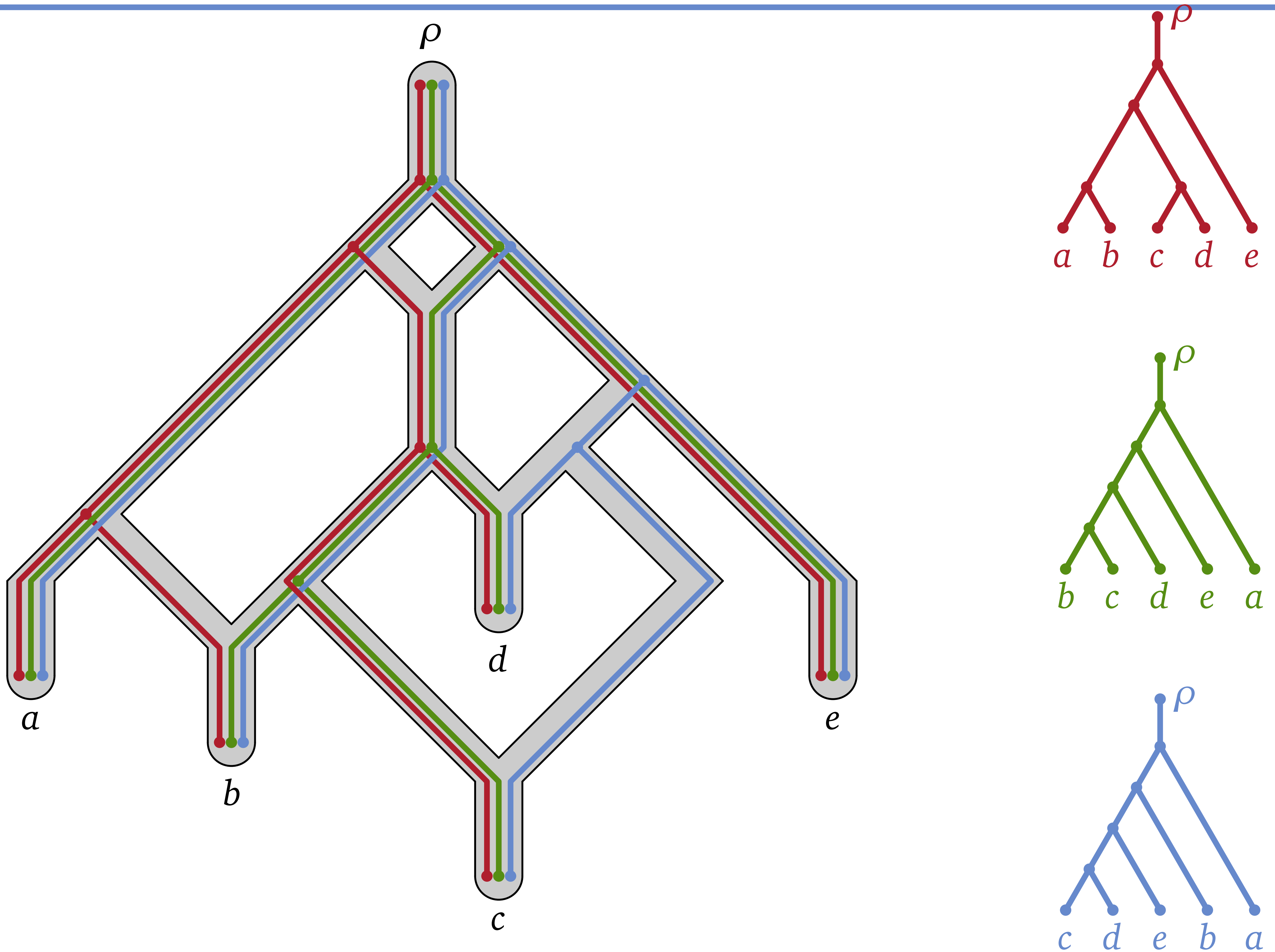
Agreement Forests vs Hybridization Networks





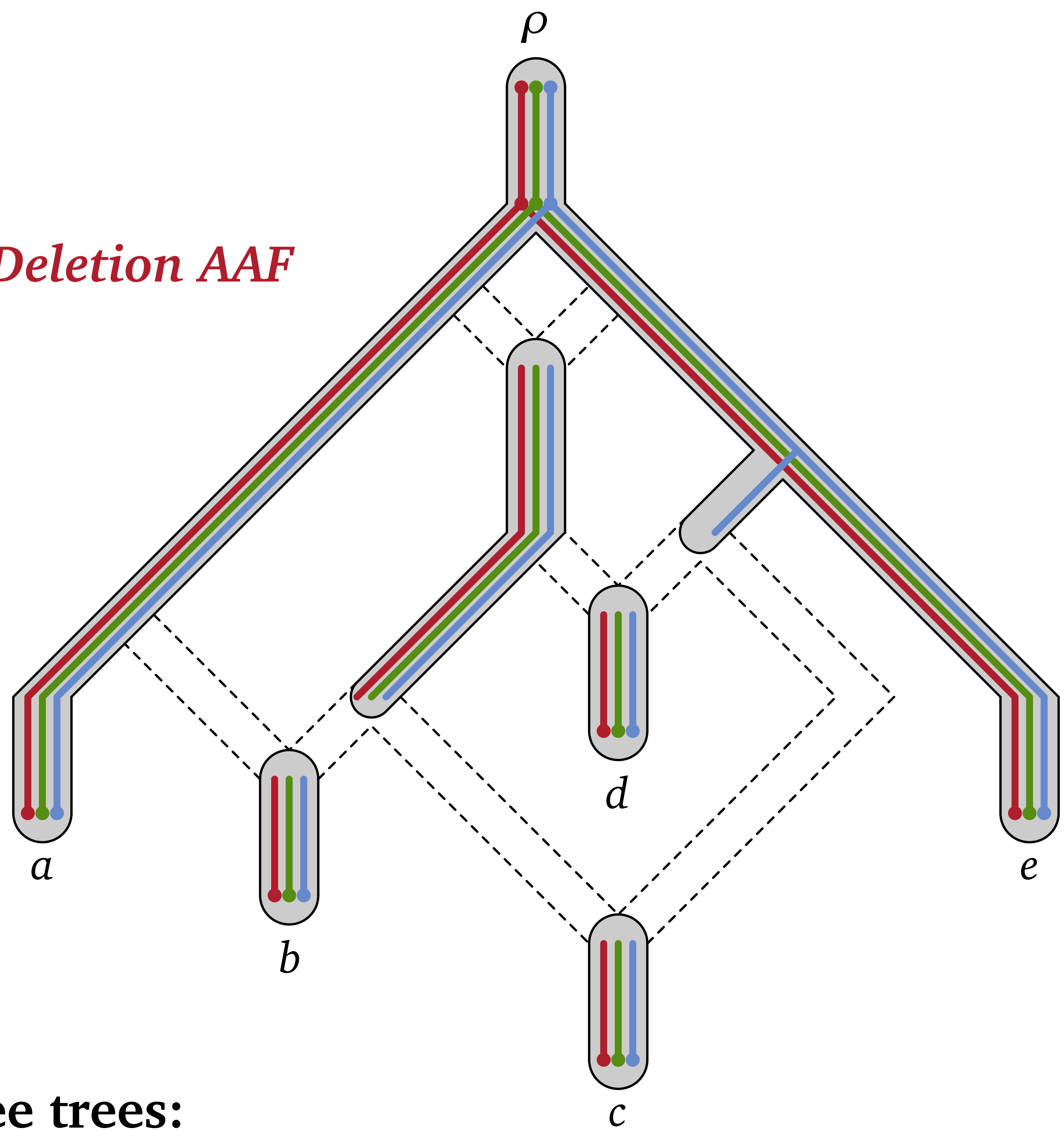
Deletion AAF

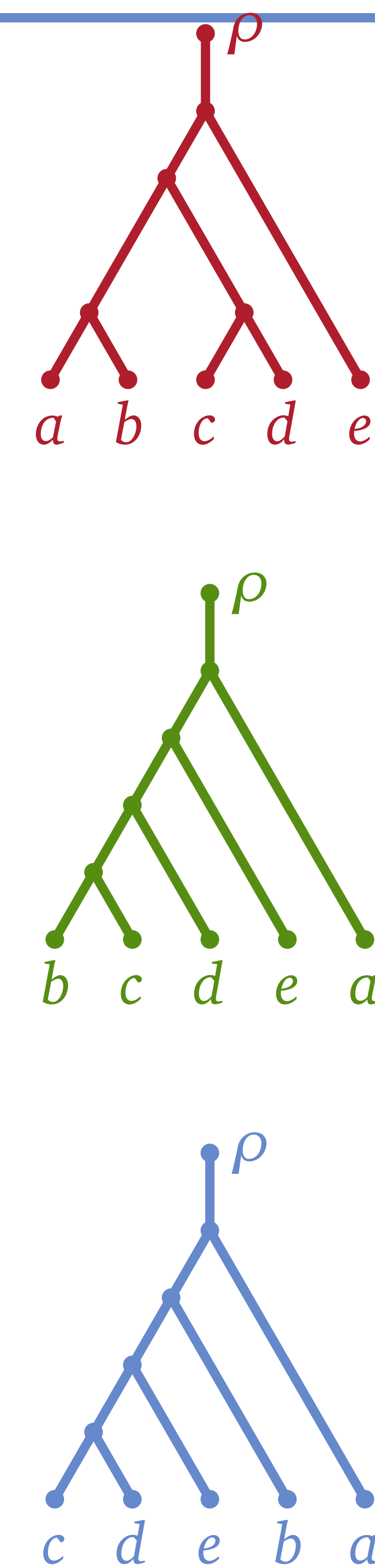
Hybridization Number on three trees in c^kpoly(n) time



Hybridization Number on three trees in c^kpoly(n) time

Deletion AAF

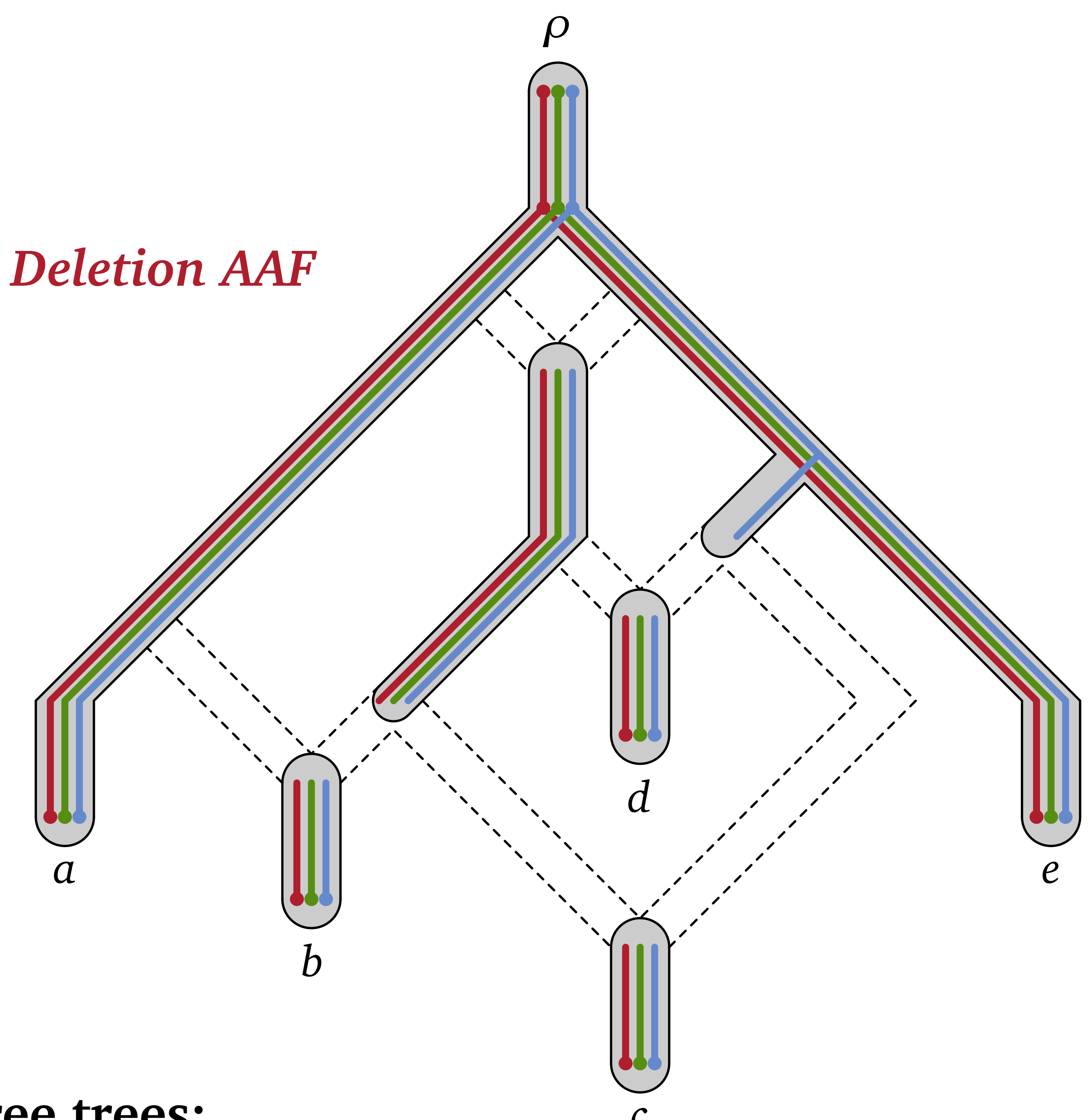


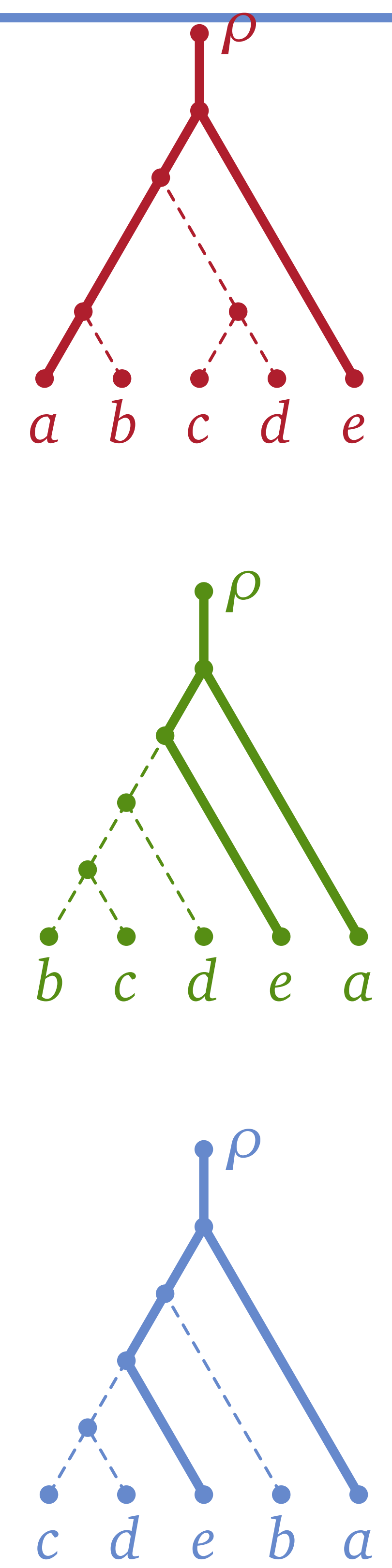


Three trees:

HYBRIDIZATION NUMBER $\geq$ |MAAF| - 1

Hybridization Number on three trees in c^kpoly(n) time

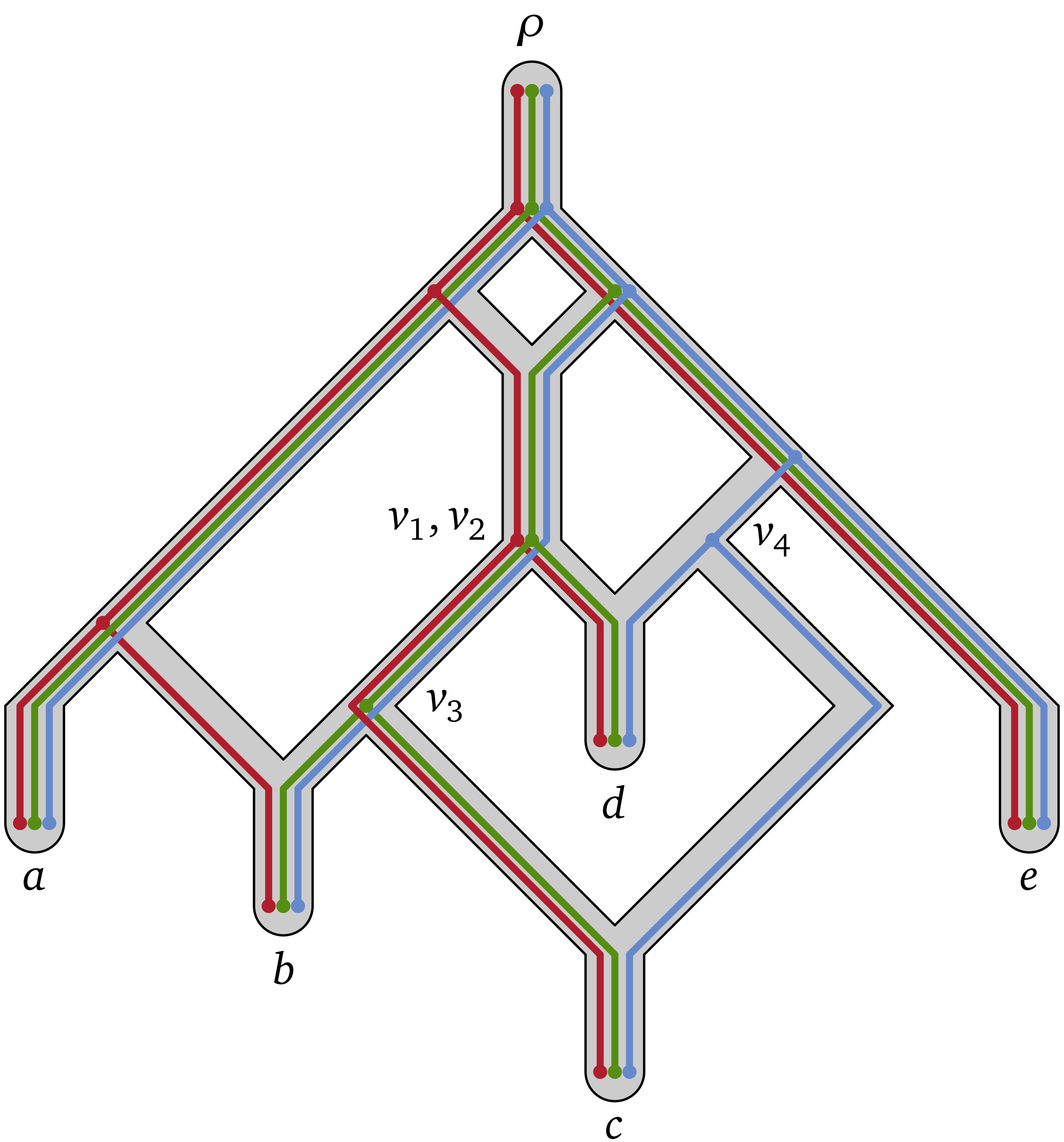


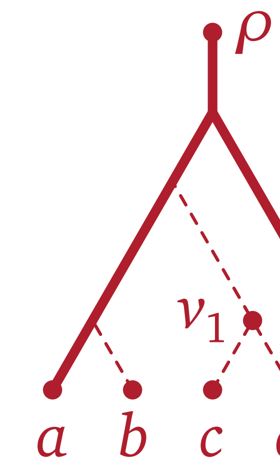


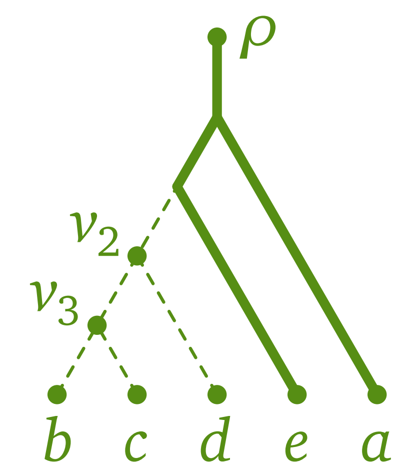
Three trees:

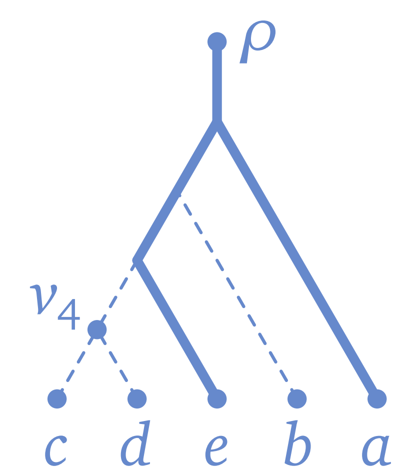
HYBRIDIZATION NUMBER ≥ |MAAF| - 1

Invisible Components and the Extended AAF

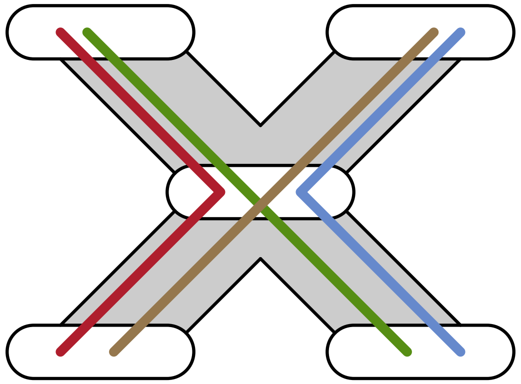




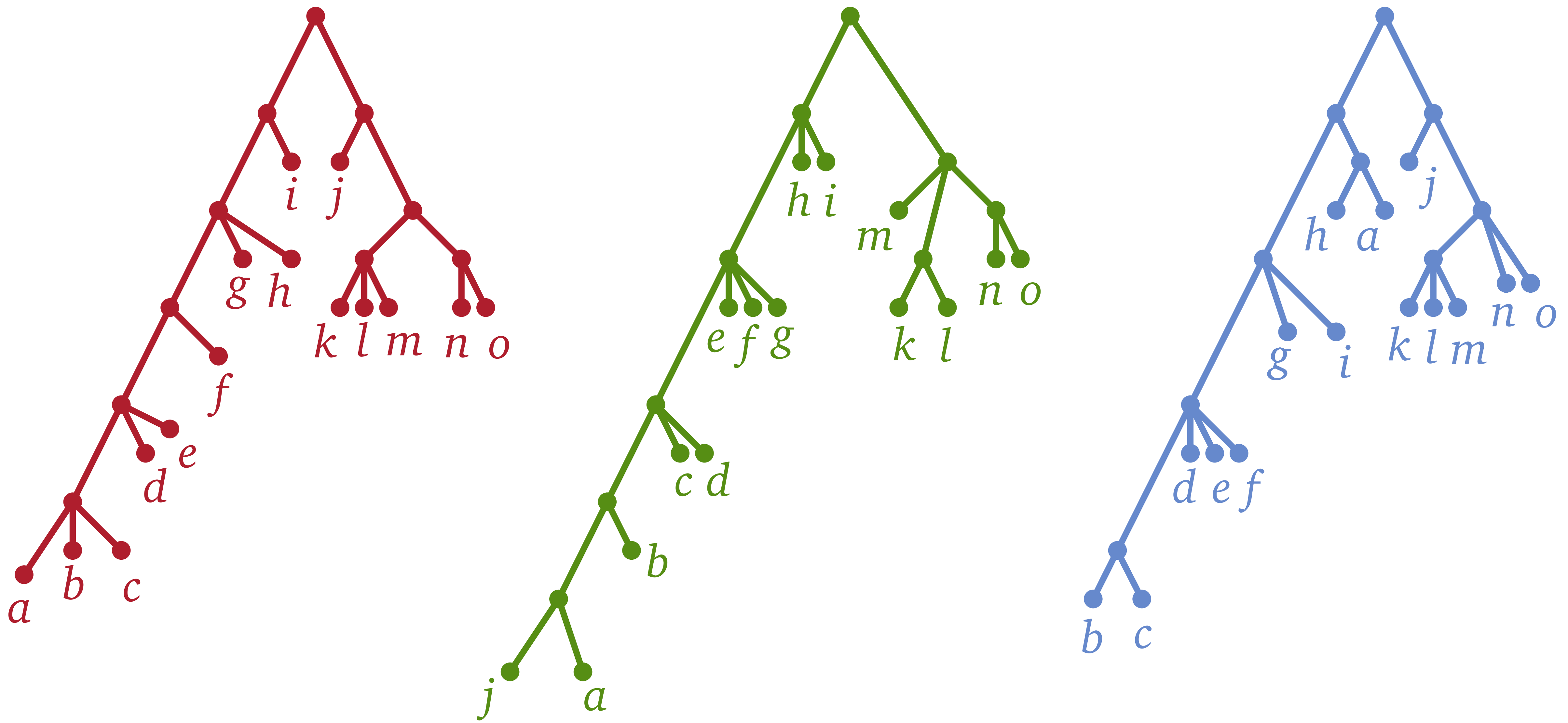




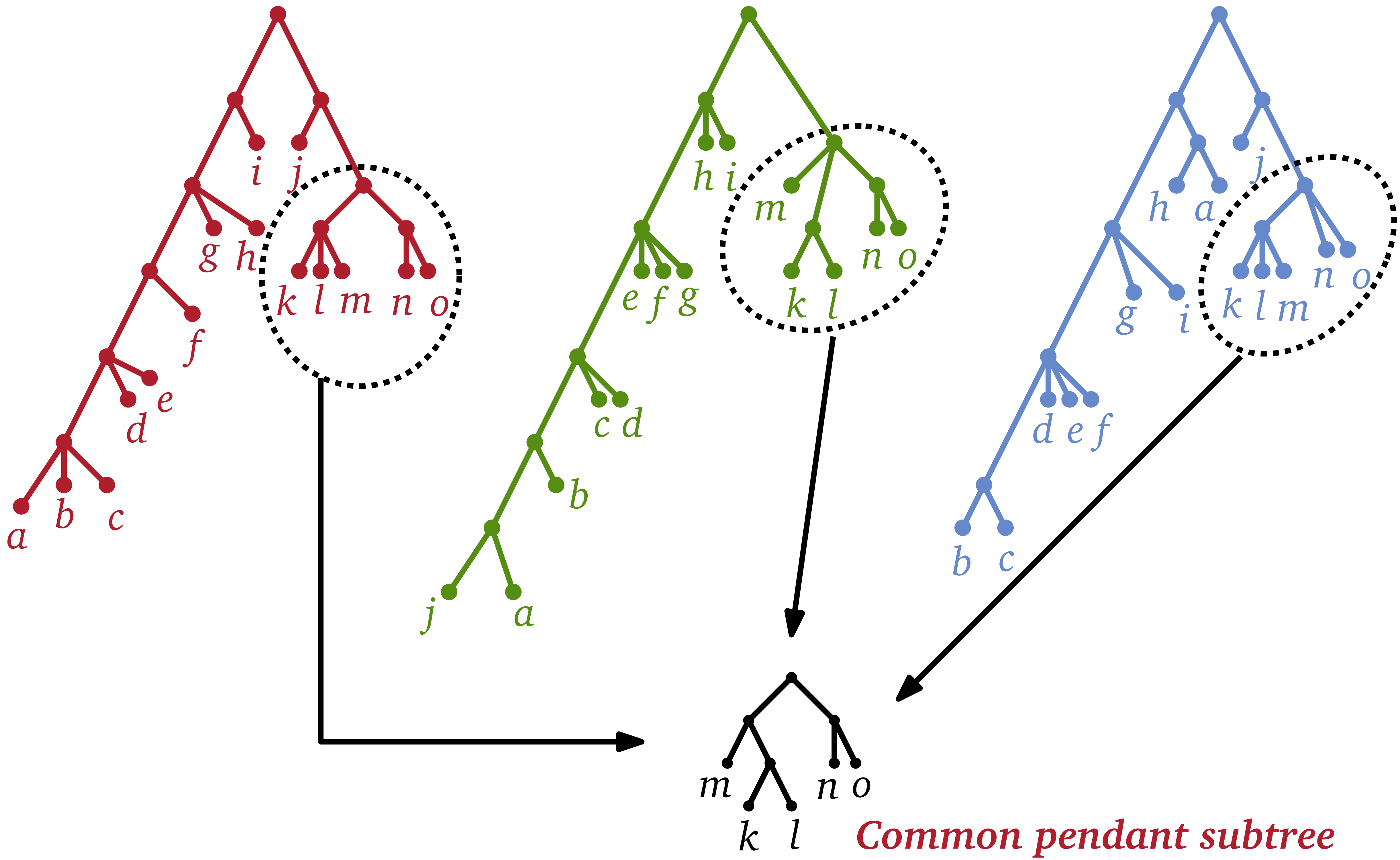
Four Trees May Not Have an Optimal Canonical Network



Reduction Rules

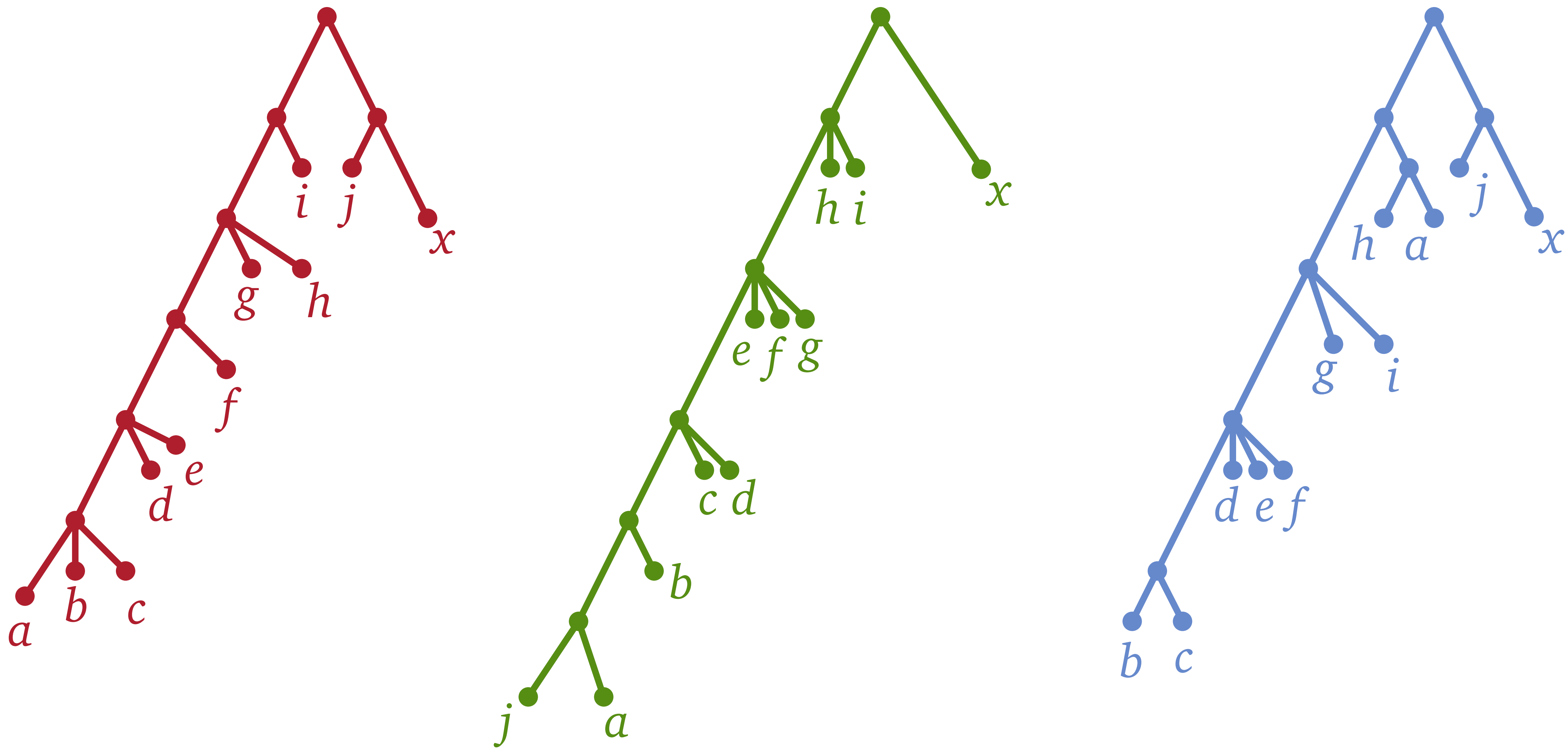


Reduction Rules



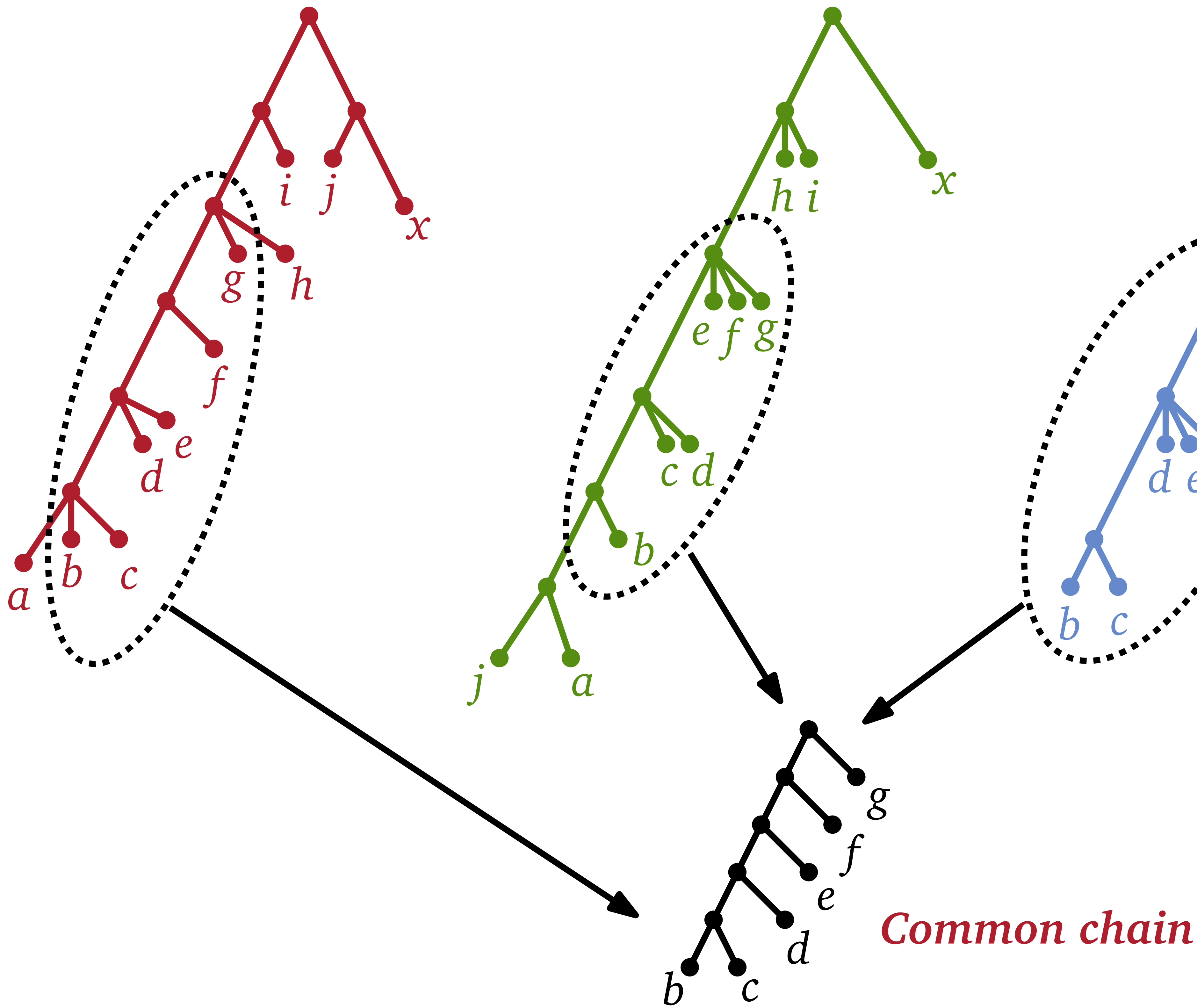
Common pendant subtree

Reduction Rules

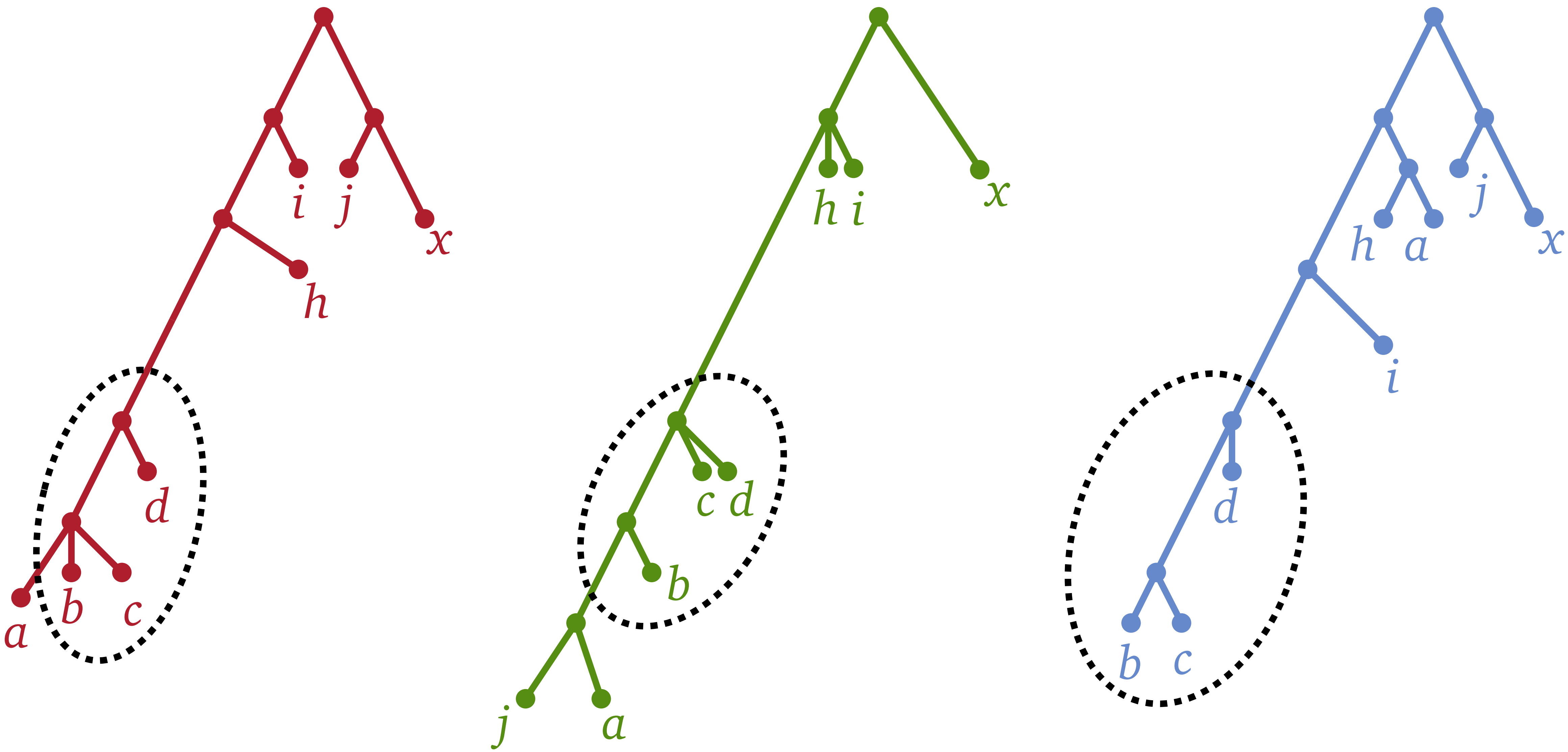


Reduce subtree to a single leaf

Reduction Rules



Reduction Rules



Reduce chain to a certain length

Results

Problem: Hybridization Number
Given: Collection of phylogenetic trees $\mathcal{T}$, each on the same n leaves, $k \in \mathbb{N}$
Question: Does there exist a phylogenetic network that displays each tree in $\mathcal{T}$ and has hybridization number at most k?

Two binary trees:

- Direct relationship to ***maximum acyclic agreement forest*** (MAAF)
- $O((28k)^k + n^3)$-time algorithm (Bordewich & Semple 2007)
- $O(3.18^k n)$- time algorithm (Whidden, Beiko & Zeh, 2013)
- Same approximability as ***directed feedback vertex set*** (Kelk, vI, Lekic, Linz, Scornavacca, Stougie, 2012)

Any number of nonbinary trees: (vI, Kelk & Scornavacca, 2014)

- Kernel with $4k(5k)^t$ leaves, with t the number of trees
- Kernel with $20k^2(\Delta^+ - 1)$ leaves, with Δ^+ the maximum outdegree
- $n^{f(k)}t$-time bounded-search algorithm, with f astronomical

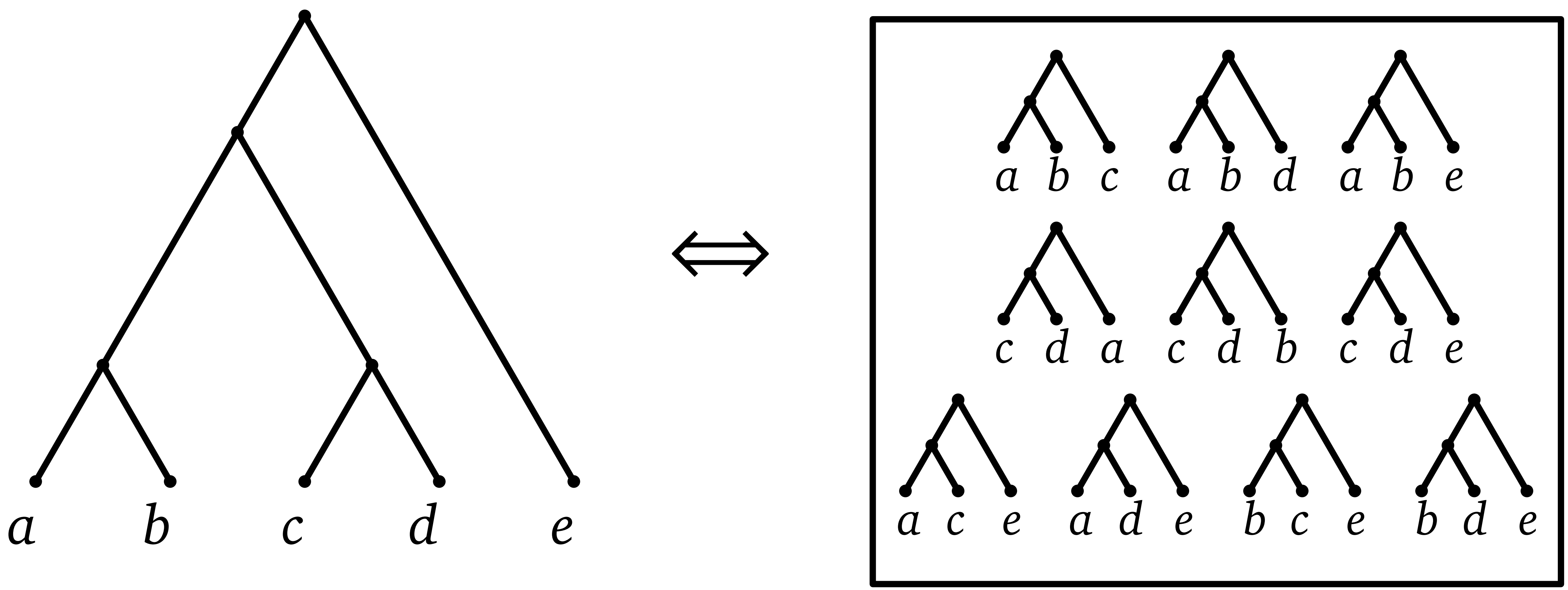
Three binary trees:

- $c^k\text{poly}(n)$ time algorithm (vI, Lekic, Kelk, Whidden & Zeh, 2014) ($c = 1609891840$)

PART 2:
NETWORKS FROM SUBNETWORKS

Encoding Trees

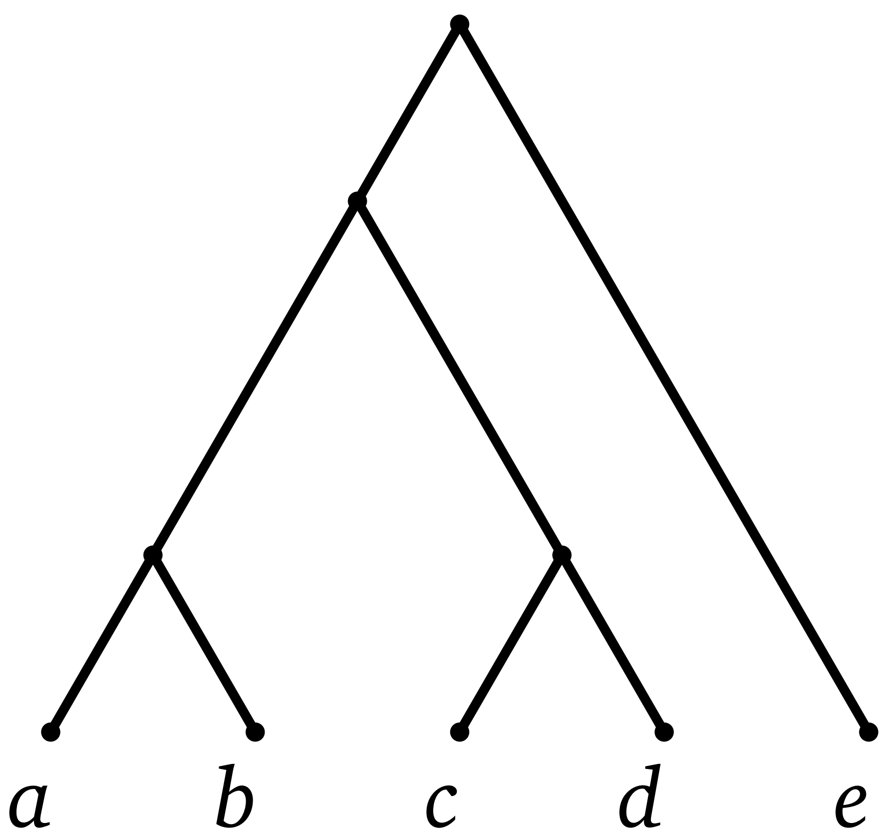
Trees are ***encoded*** by their ***triplets***.



Encoding Trees

Trees are ***encoded*** by their ***triplets***.

Trees are ***encoded*** by their ***clusters***.



$\Longleftrightarrow$

$\{a\}$ $\{b\}$ $\{c\}$

$\{d\}$ $\{a, b\}$

$\{c, d\}$ $\{a, b, c, d\}$

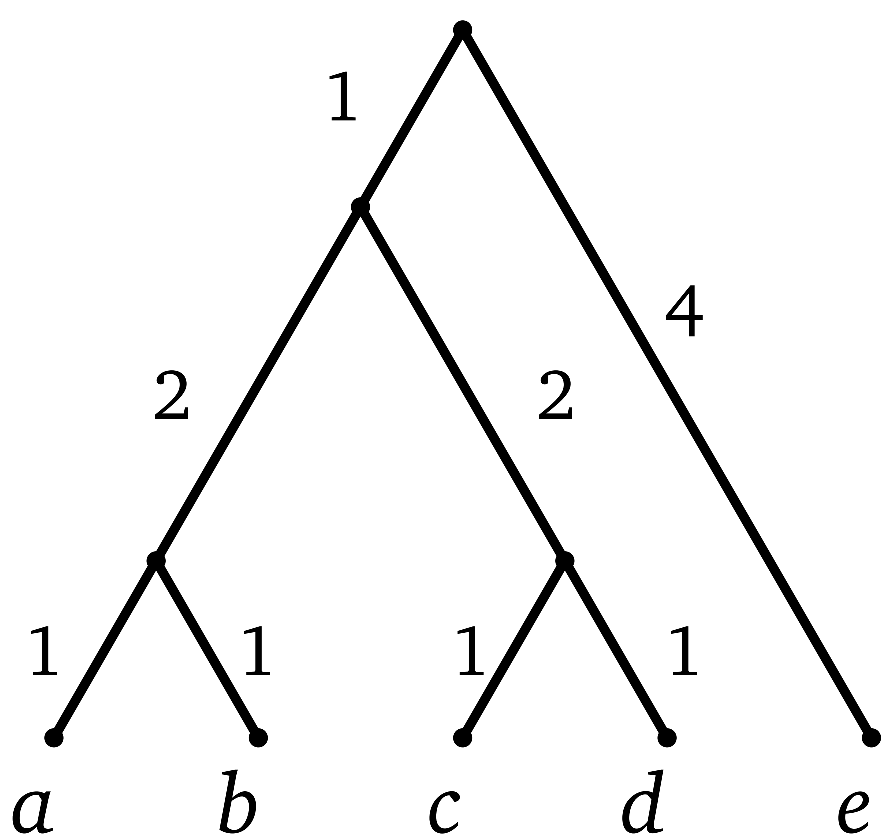
$\{a, b, c, d, e\}$

Encoding Trees

Trees are ***encoded*** by their ***triplets***.

Trees are ***encoded*** by their ***clusters***.

Trees are ***encoded*** by their ***distances***.



$\Longleftrightarrow$

	a	b	c	d	e
a	0	2	6	6	8
b	2	0	6	6	8
c	6	6	0	2	8
d	6	6	2	0	8
e	8	8	8	8	0

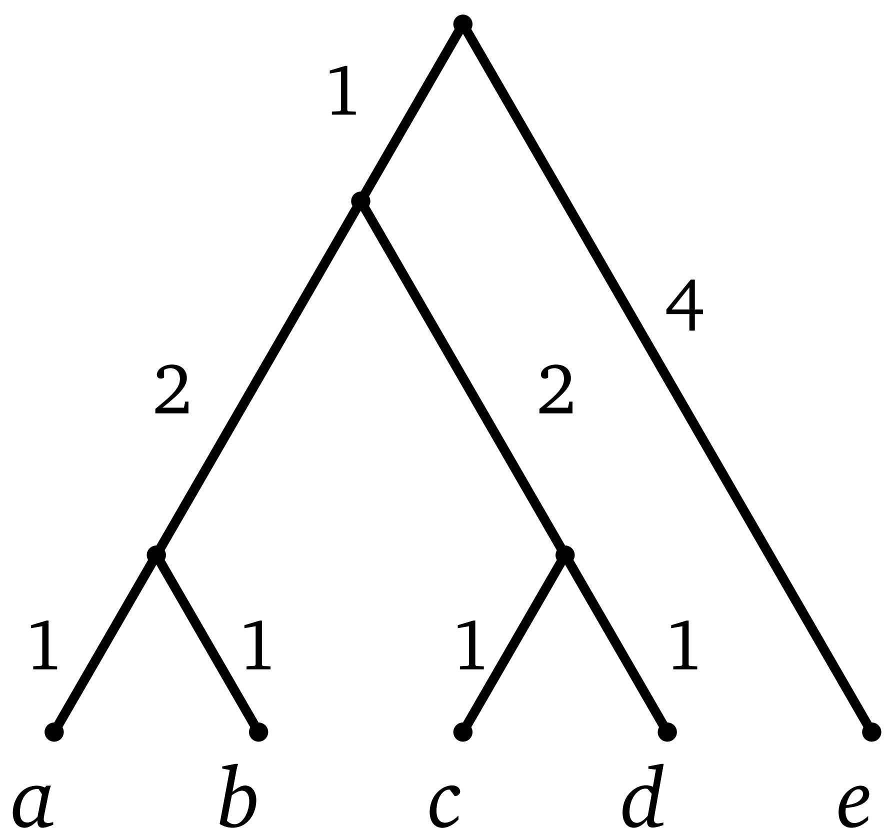
Encoding Trees

Trees are ***encoded*** by their ***triplets***.

Trees are ***encoded*** by their ***clusters***.

Trees are ***encoded*** by their ***distances***.

Can we encode ***networks***?



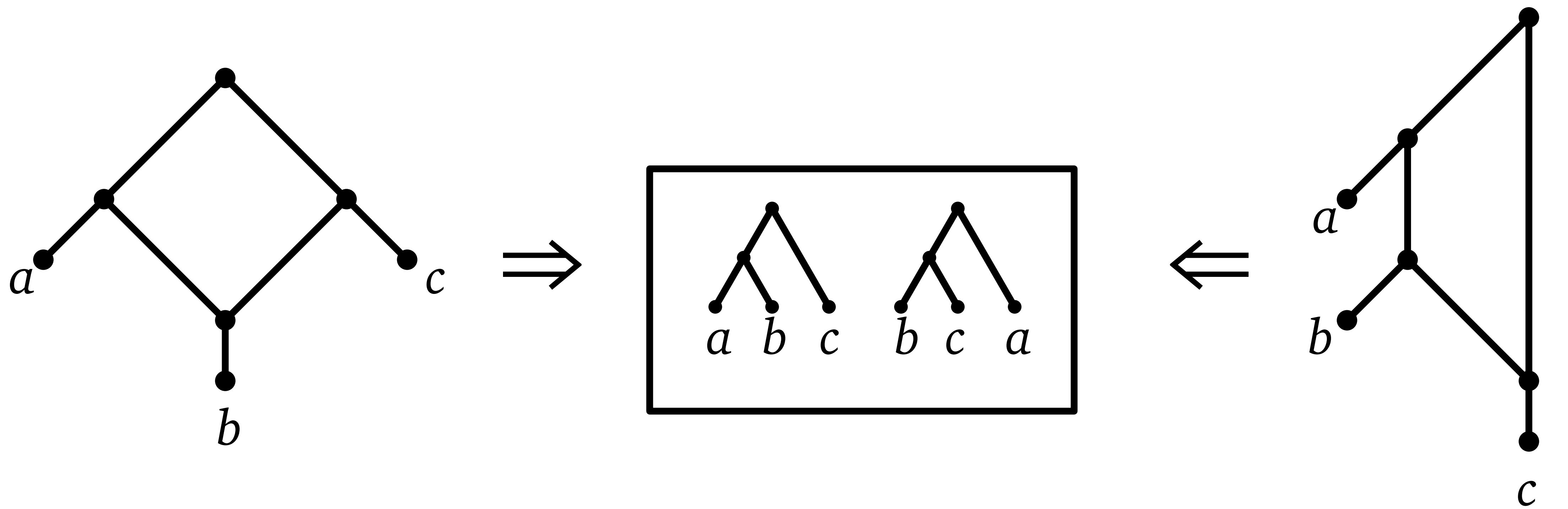
$\Longleftrightarrow$

$$
\begin{array}{c c}
 & \begin{array}{ccccc} a & b & c & d & e \end{array} \\
\begin{array}{c} a \\ b \\ c \\ d \\ e \end{array} &
\begin{bmatrix}
0 & 2 & 6 & 6 & 8 \\
2 & 0 & 6 & 6 & 8 \\
6 & 6 & 0 & 2 & 8 \\
6 & 6 & 2 & 0 & 8 \\
8 & 8 & 8 & 8 & 0
\end{bmatrix}
\end{array}
$$

Encoding Networks

Trees are encoded by their ***triplets***.

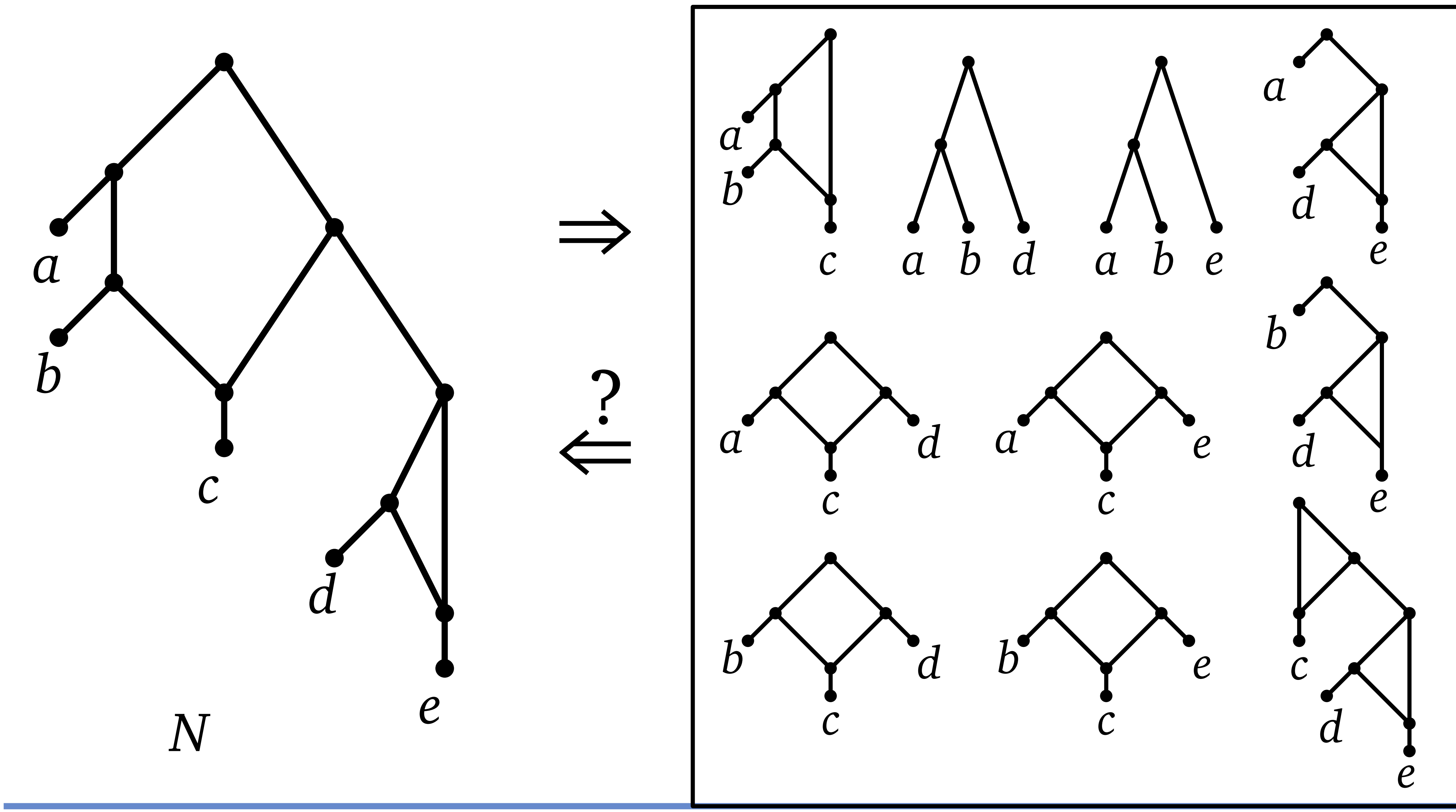
Networks are ***not*** encoded by their ***triplets***.



Trinets and Subnets

Trees are encoded by their ***triplets***.

Are ***networks*** encoded by their ***trinets***?

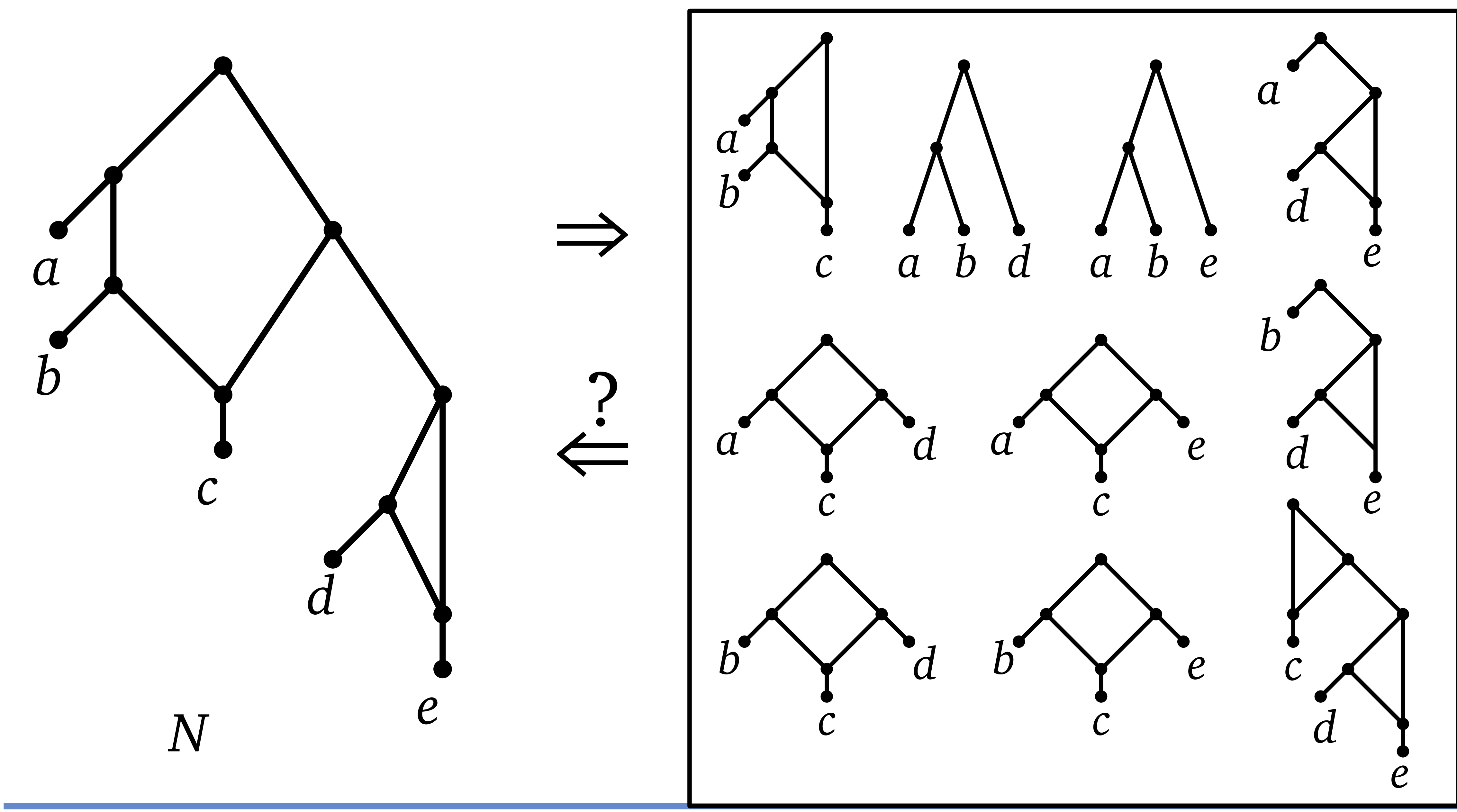


$\mathscr{T}(N)$

Trinets and Subnets

The ***subnet*** $N|X'$ is obtained from N by

1. deleting all vertices that are not on ***any*** path from the root to a leaf in X';
2. deleting all vertices that are on ***all*** paths from the root to a leaf in X';
3. suppressing indegree-1 outdegree-1 vertices and parallel arcs.

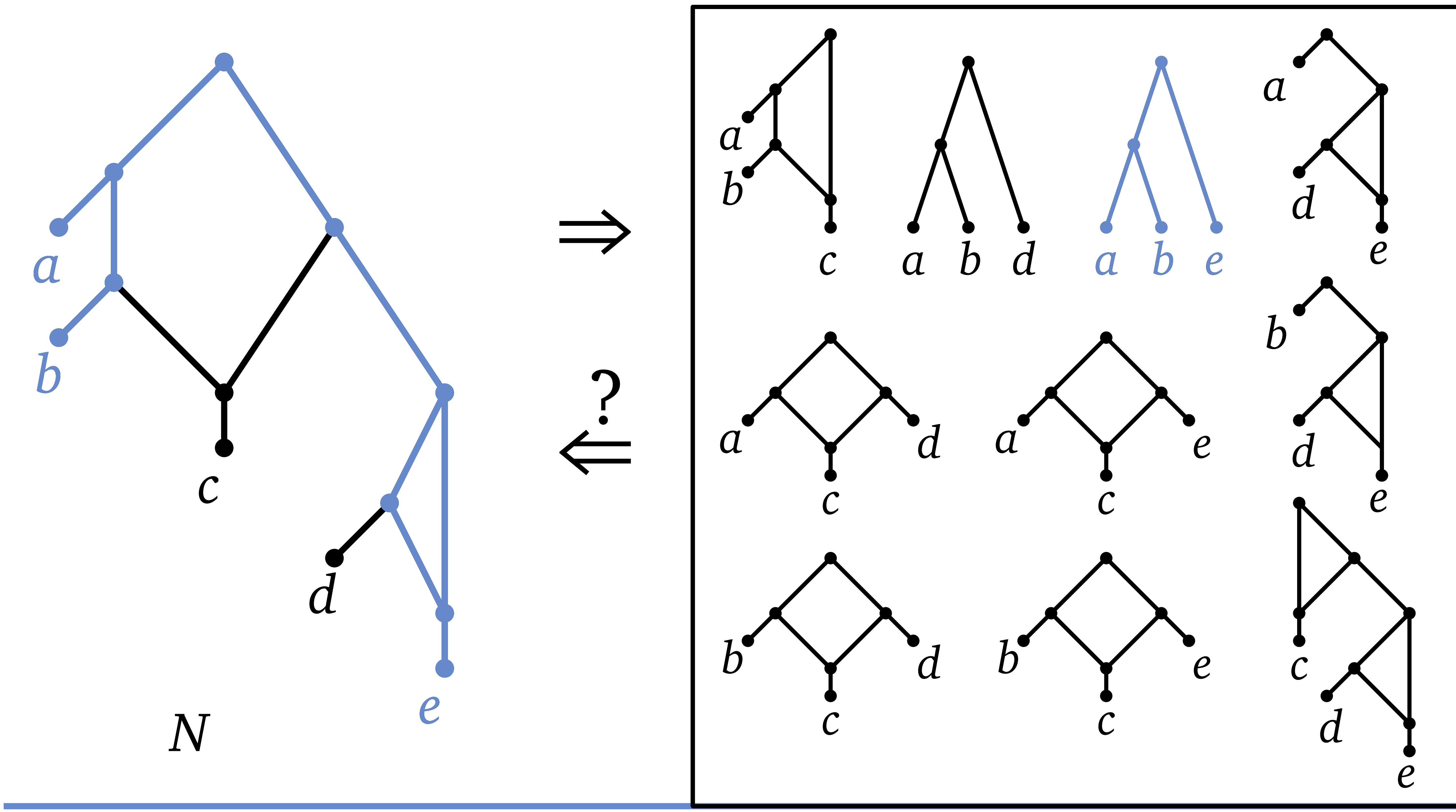


$\mathscr{T}(N)$

Trinets and Subnets

The ***subnet*** $N|X'$ is obtained from N by

1. deleting all vertices that are not on ***any*** path from the root to a leaf in X';
2. deleting all vertices that are on ***all*** paths from the root to a leaf in X';
3. suppressing indegree-1 outdegree-1 vertices and parallel arcs.

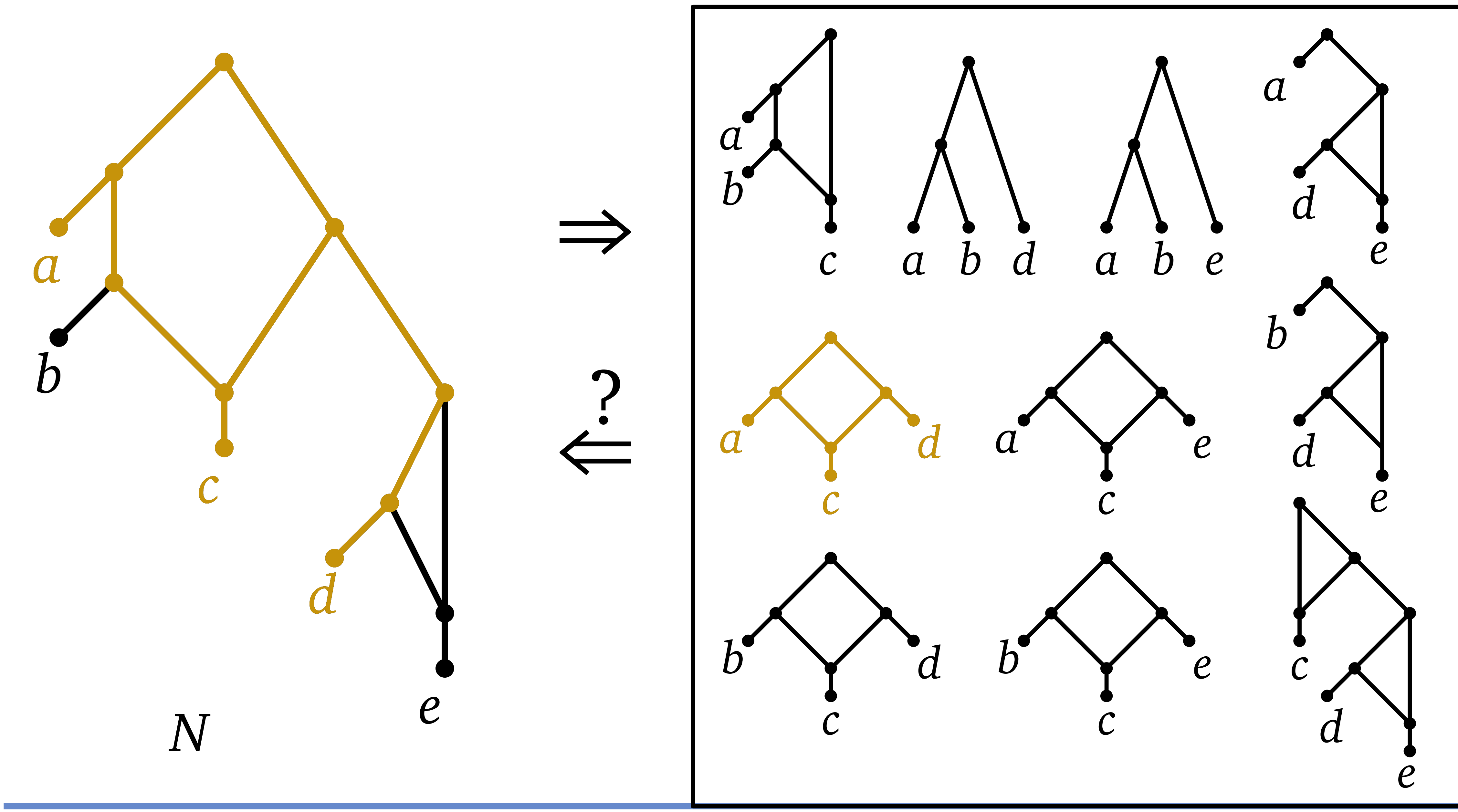


$\mathscr{T}(N)$

Trinets and Subnets

The ***subnet*** $N|X'$ is obtained from N by

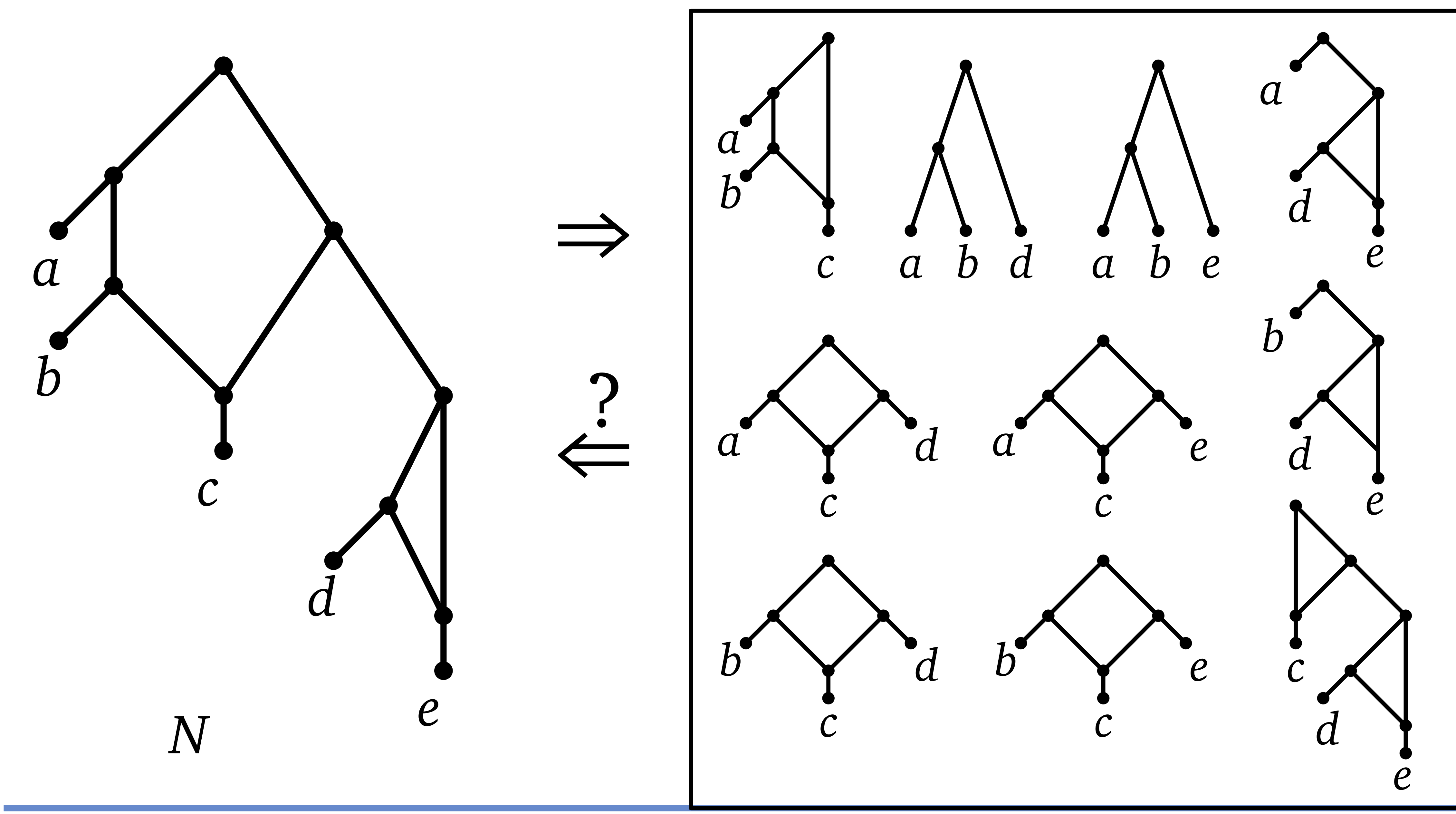
1. deleting all vertices that are not on ***any*** path from the root to a leaf in X';
2. deleting all vertices that are on ***all*** paths from the root to a leaf in X';
3. suppressing indegree-1 outdegree-1 vertices and parallel arcs.



$\mathscr{T}(N)$

Trinets and Subnets

A ***trinet*** is a subnet with 3 leaves.
A ***binet*** is a subnet with 2 leaves.

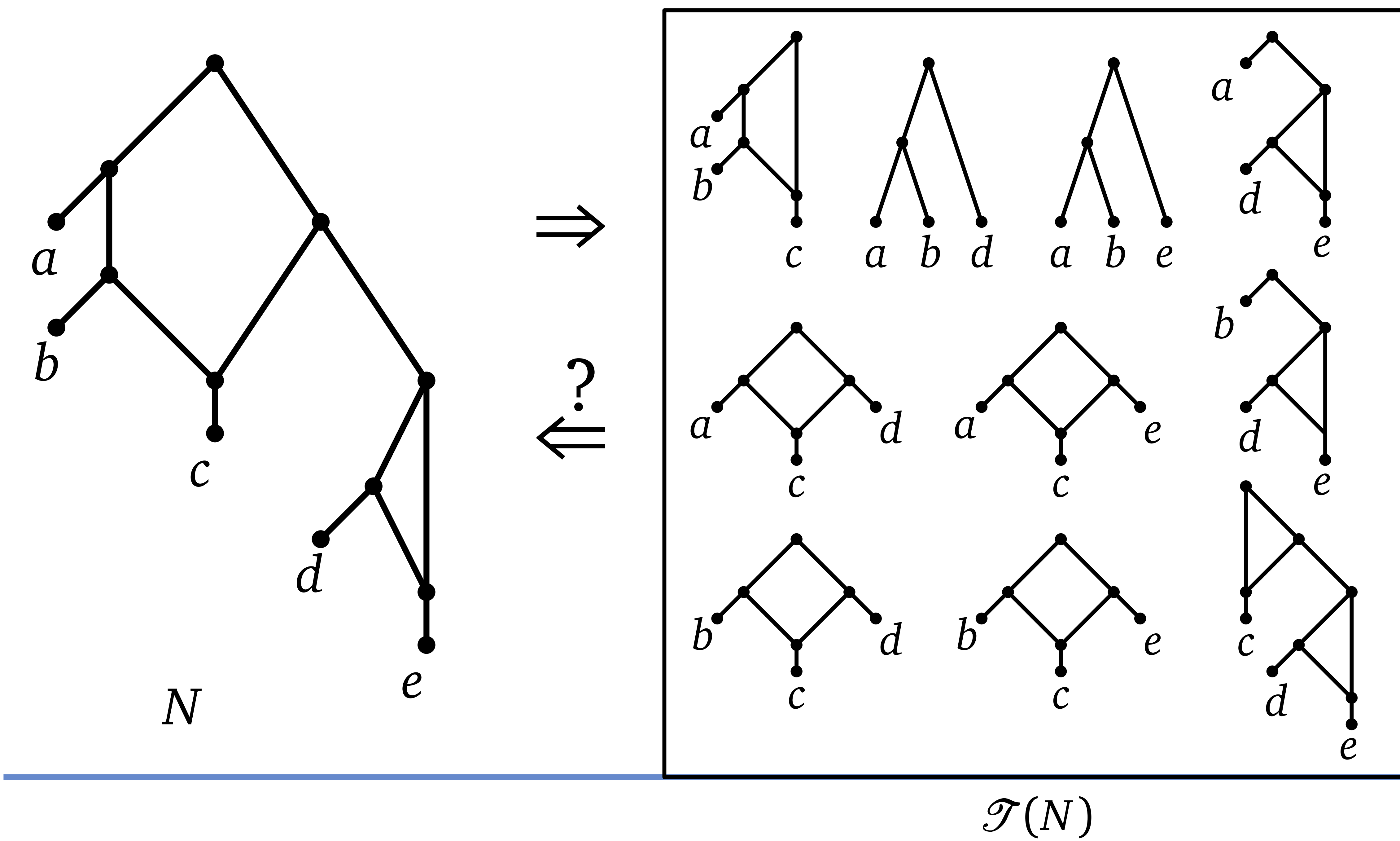


$\mathscr{T}(N)$

Trinets and Subnets

Definition.

- ***level**-k*: each biconnected component has hybridization number $\leq k$;
- ***tree-child***: each non-leaf vertex has a child with indegree-1.

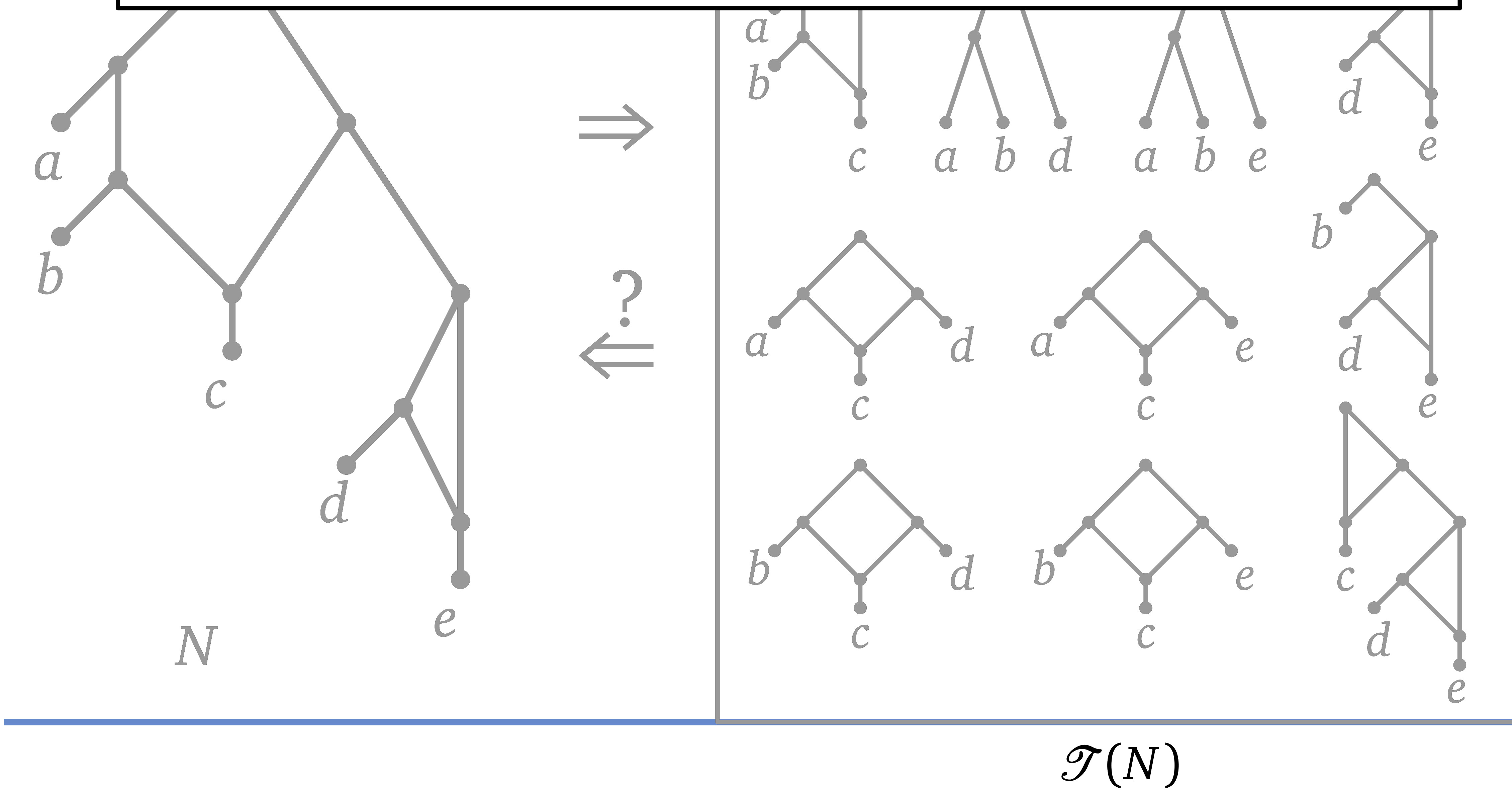


Trinets and Subnets

Definition.

- ***level**-k*: each biconnected component has hybridization number $\leq k$;
- ***tree-child***: each non-leaf vertex has a child with indegree-1.

> **Theorem.** (Huber, vI & Moulton) Binary level-1, level-2 and tree-child networks are all encoded by their trinets.



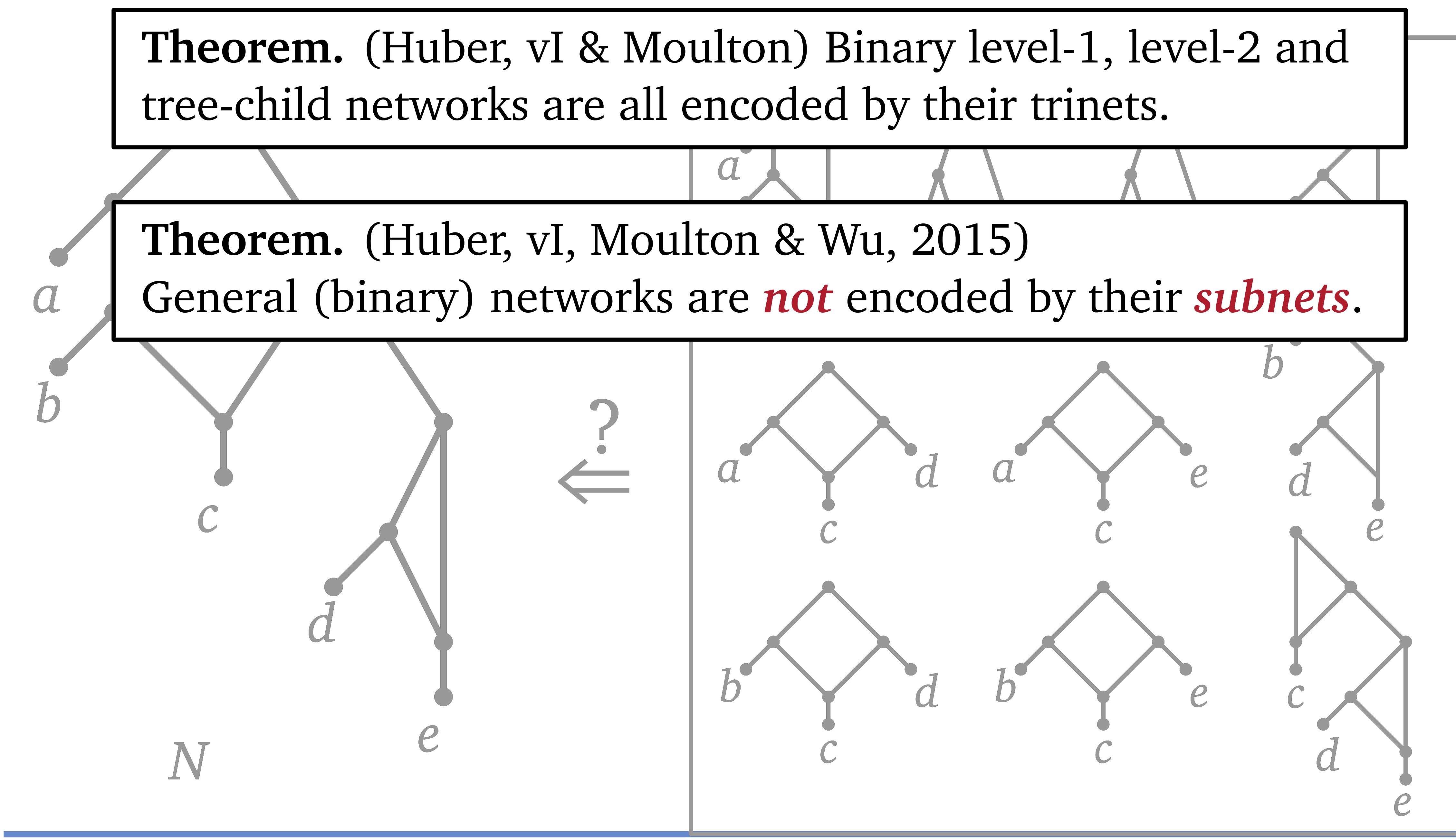
Trinets and Subnets

Definition.

- ***level***-k: each biconnected component has hybridization number $\leq k$;
- ***tree-child***: each non-leaf vertex has a child with indegree-1.

Theorem. (Huber, vI & Moulton) Binary level-1, level-2 and tree-child networks are all encoded by their trinets.

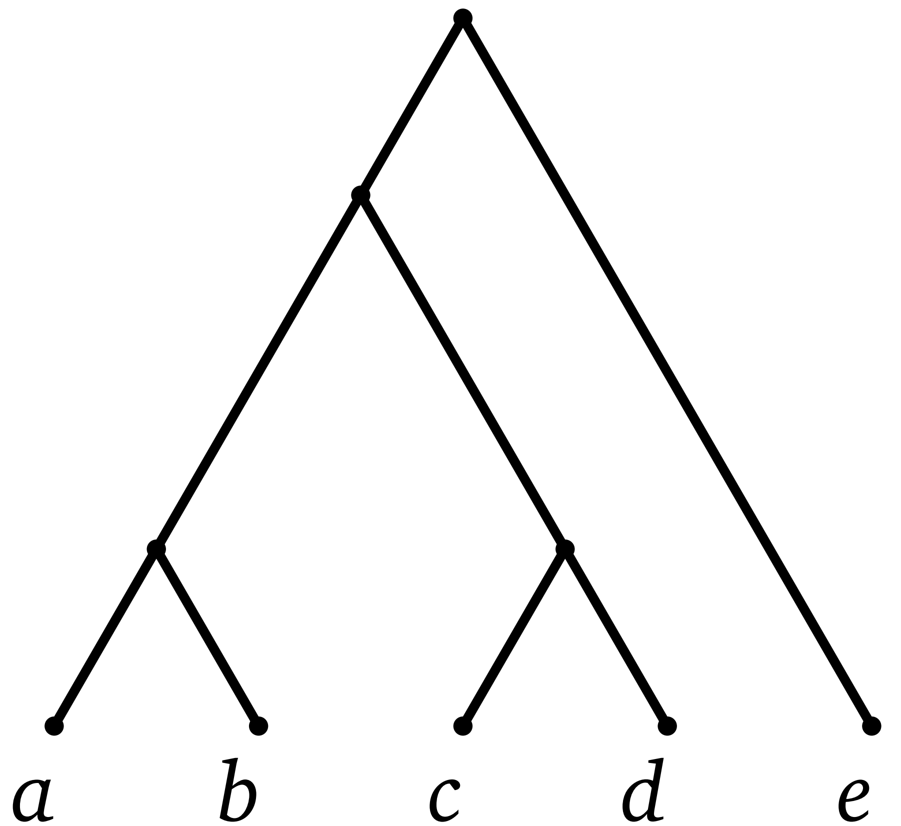
Theorem. (Huber, vI, Moulton & Wu, 2015) General (binary) networks are ***not*** encoded by their ***subnets***.



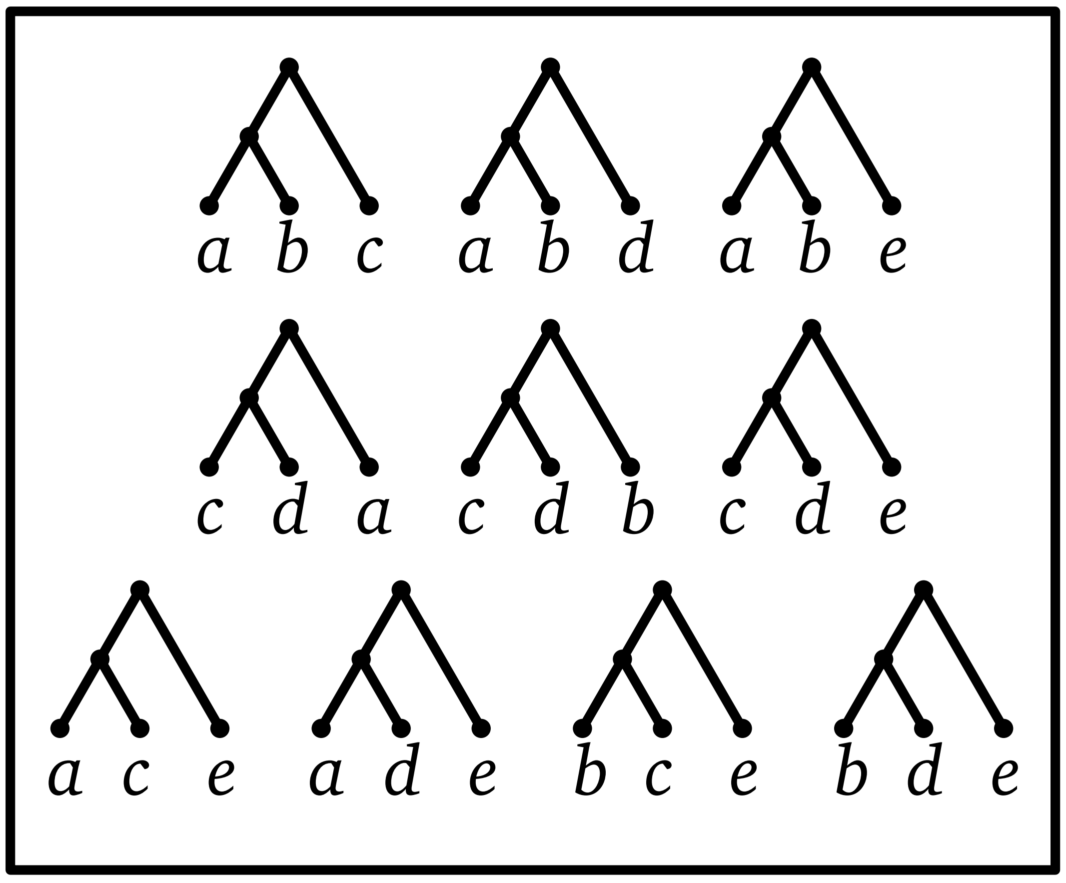
$\mathcal{T}(N)$

Reconstructing trees from triplets

Trees are ***encoded*** by their ***triplets***



$\Longleftrightarrow$

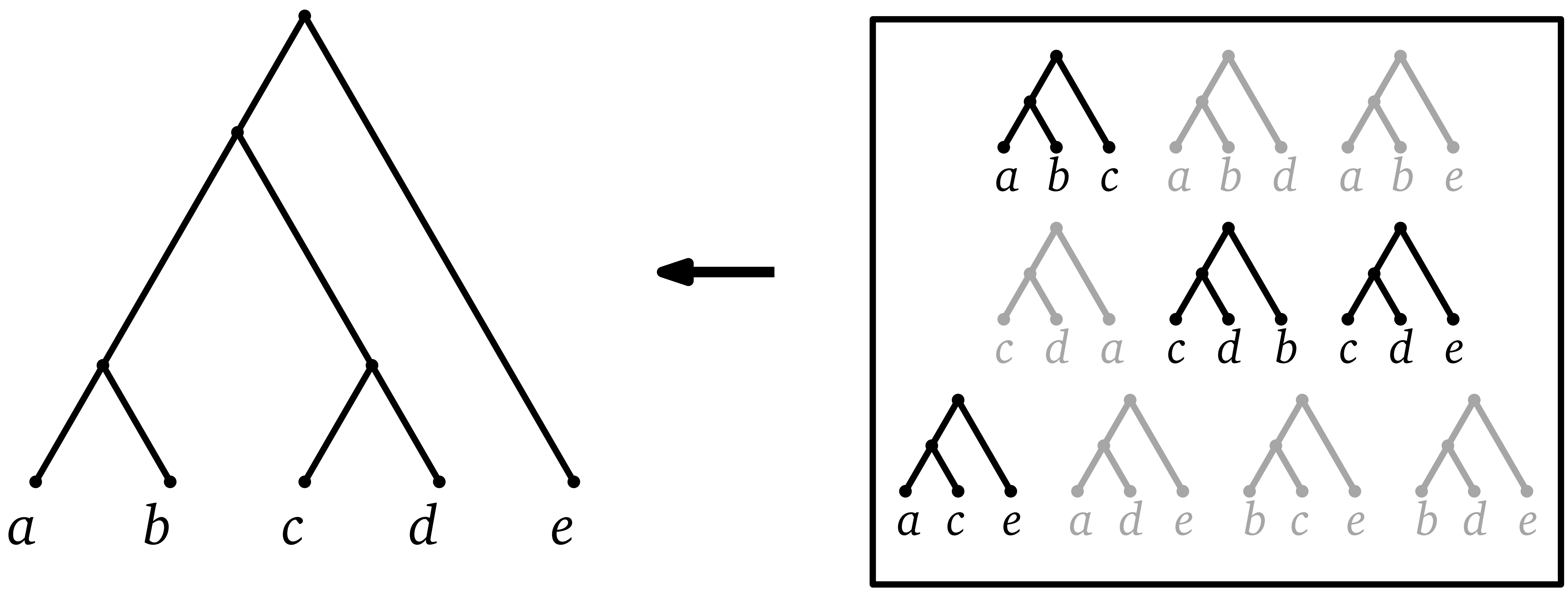


Reconstructing trees from triplets

Trees are ***encoded*** by their ***triplets***

and given any set of triplets, we can ***construct*** a tree displaying them, if one exists, in polynomial time.

(Aho, Sagiv, Szymanski, Ullman, 1981)



Reconstructing networks from trinets

Level-1 networks are encoded by their ***trinets***

- and given a ***complete*** set of trinets, we can construct a level-1 network displaying them, if one exists, in polynomial time.

(Huber & Moulton, 2013)

Reconstructing networks from trinets

Level-1 networks are encoded by their ***trinets***

- and given a ***complete*** set of trinets, we can construct a level-1 network displaying them, if one exists, in polynomial time.

(Huber & Moulton, 2013)

- for an ***arbitrary*** set of trinets, this is ***NP-hard***

but solvable in $O(3^n \text{poly}(n))$ time

(Huber, vI, Moulton, Scornavacca & Wu 2014)

Reconstructing networks from trinets

Level-1 networks are encoded by their ***trinets***

- and given a ***complete*** set of trinets, we can construct a level-1 network displaying them, if one exists, in polynomial time.

(Huber & Moulton, 2013)

- for an ***arbitrary*** set of trinets, this is ***NP-hard***

  but solvable in $O(3^n \text{poly}(n))$ time

- for an arbitrary set of ***binets***, this is polynomial-time solvable

(Huber, vI, Moulton, Scornavacca & Wu 2014)

Reconstructing networks from trinets

Level-1 networks are encoded by their ***trinets***

- and given a ***complete*** set of trinets, we can construct a level-1 network displaying them, if one exists, in polynomial time.

(Huber & Moulton, 2013)

- for an ***arbitrary*** set of trinets, this is ***NP-hard***

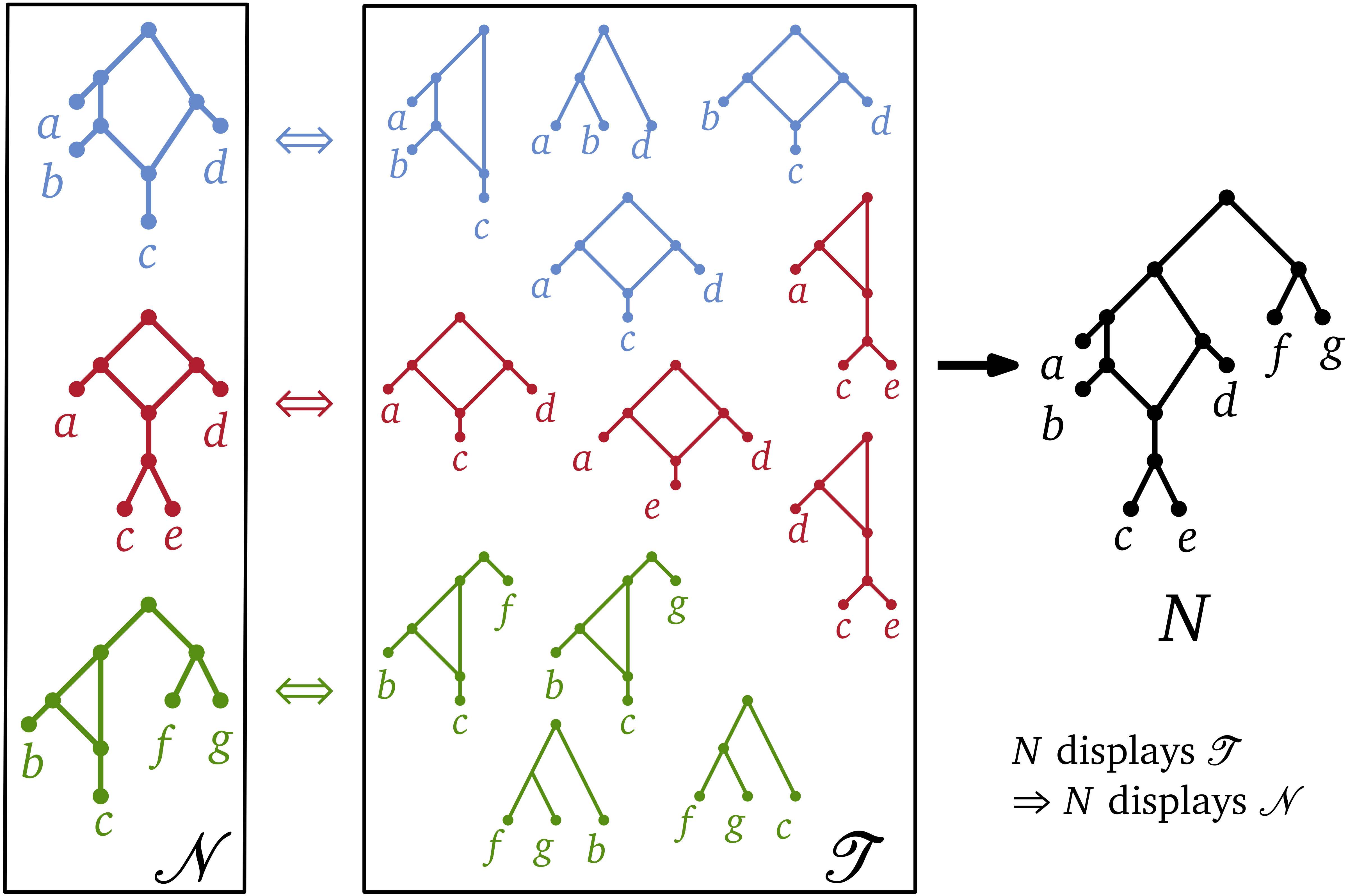
  but solvable in $O(3^n \text{poly}(n))$ time

- for an arbitrary set of ***binets***, this is polynomial-time solvable

- and also for ***subnets*** in which all cycles have size 3.

(Huber, vI, Moulton, Scornavacca & Wu 2014)

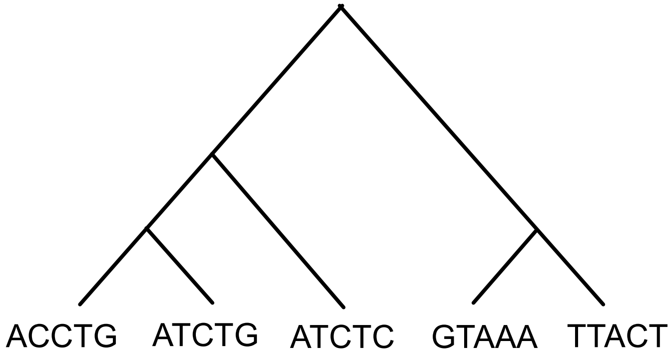
Supernetwork Methods



PART 3:
NETWORKS FROM SEQUENCES

Maximum Parsimony for **trees**

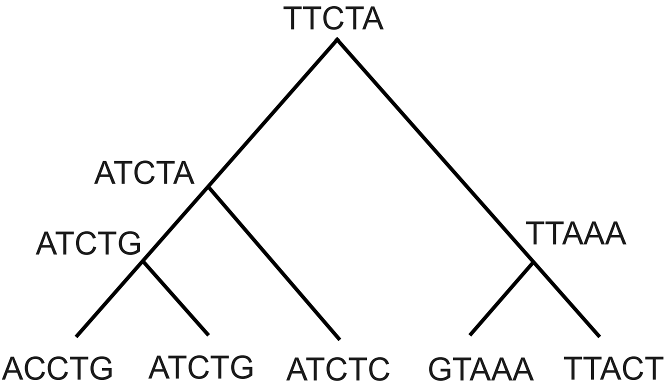
Small parsimony problem: given a tree and a sequence for each leaf, assign sequences to the internal vertices in order to **minimize** the total number of **changes**.



Example input

Maximum Parsimony for trees

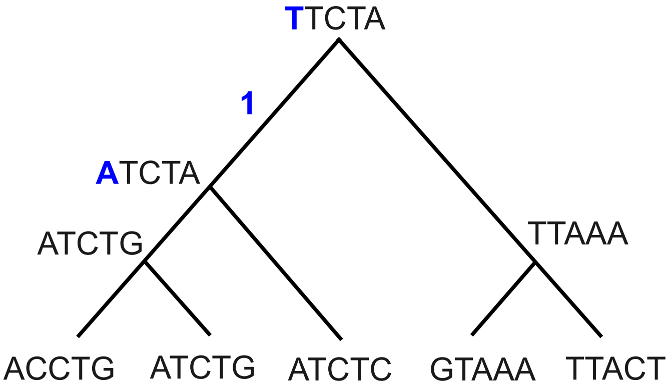
Small parsimony problem: given a tree and a sequence for each leaf, assign sequences to the internal vertices in order to **minimize** the total number of **changes**.



Example labelling of internal vertices

Maximum Parsimony for trees

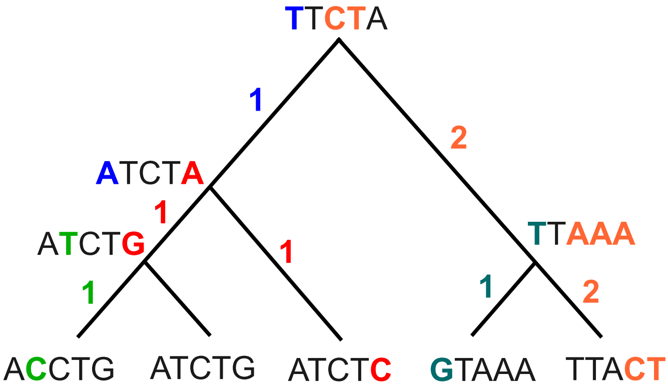
Small parsimony problem: given a tree and a sequence for each leaf, assign sequences to the internal vertices in order to **minimize** the total number of **changes**.



Example of one change

Maximum Parsimony for trees

Small parsimony problem: given a tree and a sequence for each leaf, assign sequences to the internal vertices in order to **minimize** the total number of **changes**.



The parsimony score is 9.

Maximum Parsimony for trees

Small parsimony problem: given a tree and a sequence for each leaf, assign sequences to the interior vertices in order to **minimize** the total number of **changes**.

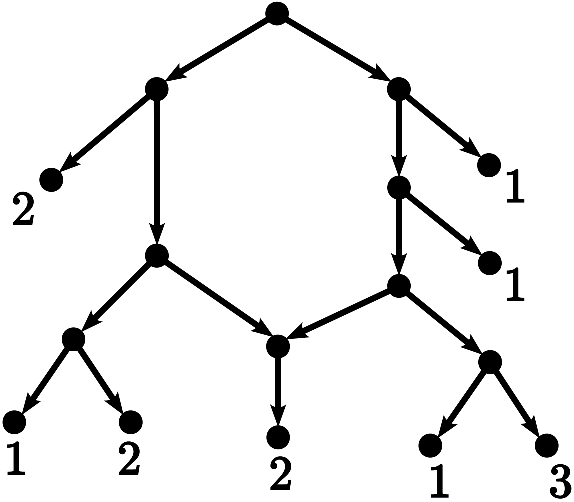
- Polynomial-time solvable:
  - Consider each character (position in the sequences) separately.
  - Use dynamic programming (Fitch, 1971).

Small Parsimony Problem on **Networks**

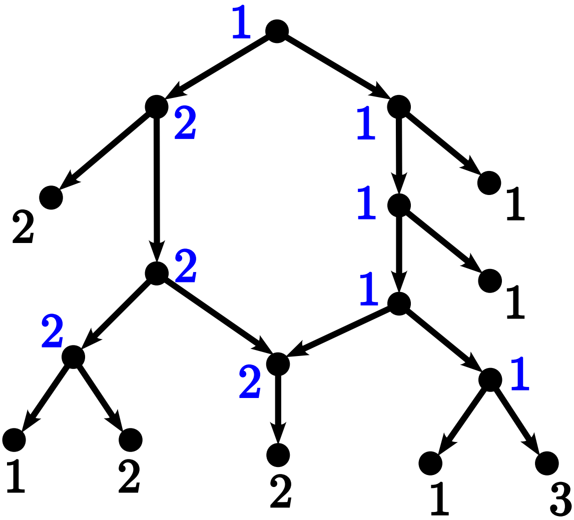
Given a network and a state for each leaf.

- **Hardwired** Parsimony Score: the minimum number of state-changes over all possible assignments of states to internal vertices.
- **Softwired** Parsimony Score: the minimum parsimony score of a tree displayed by the network.

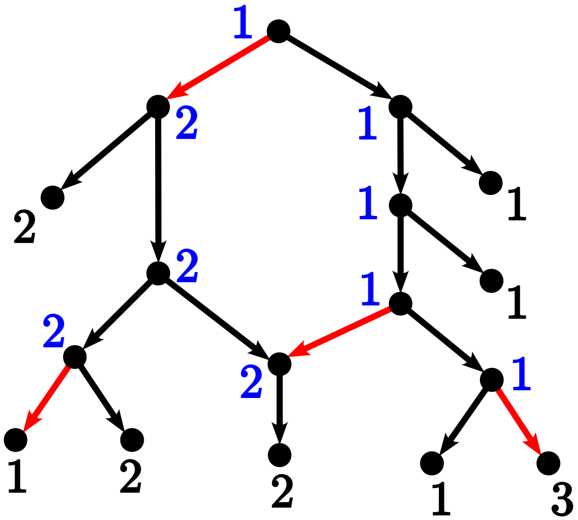
Example: Input



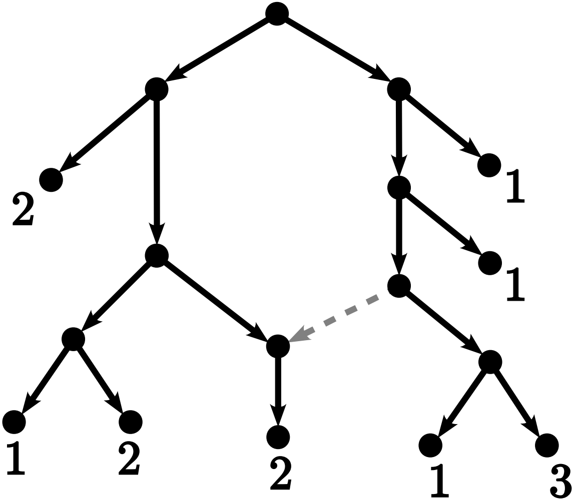
Possible asignment of states to internal vertices



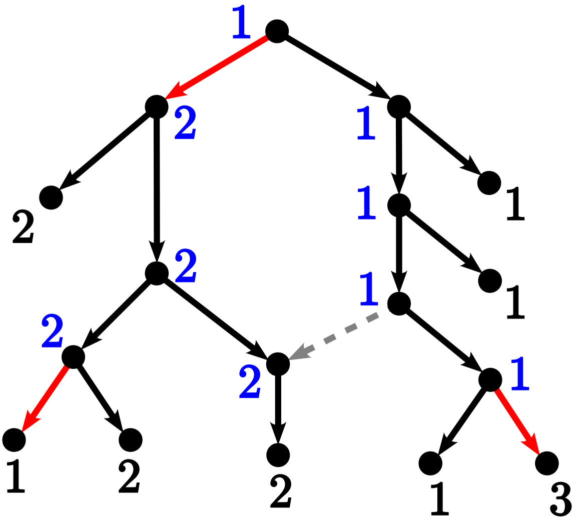
Hardwired Parsimony Score = 4



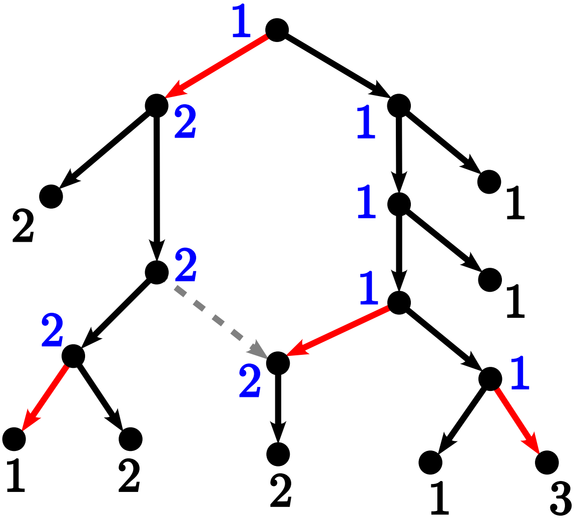
One of the two trees displayed by the network



The parsimony score of this tree is 3

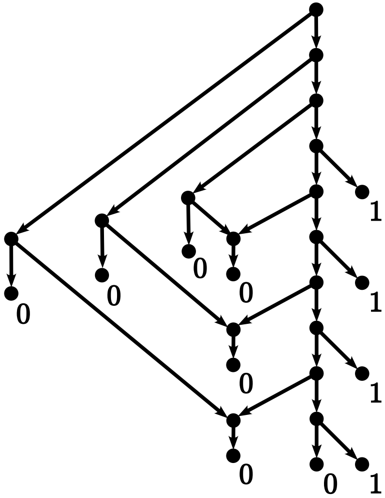


The parsimony score of the other tree is 4

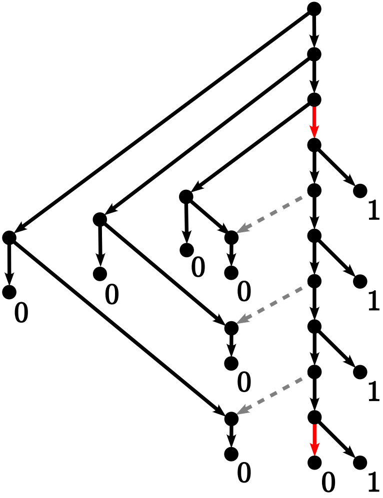


The softwired parsimony score of the network is $\min\{3, 4\} = 3$

Hardwired and Softwired scores can be arbitrarily far apart

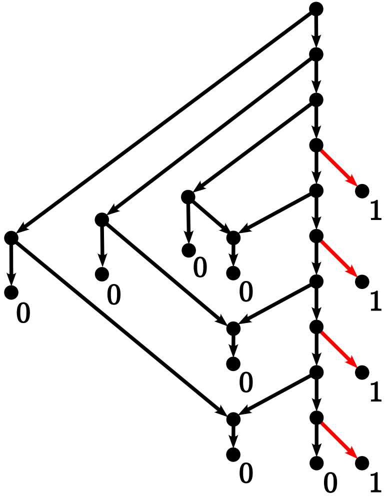


Hardwired and Softwired scores can be arbitrarily far apart



Softwired Parsimony Score = 2

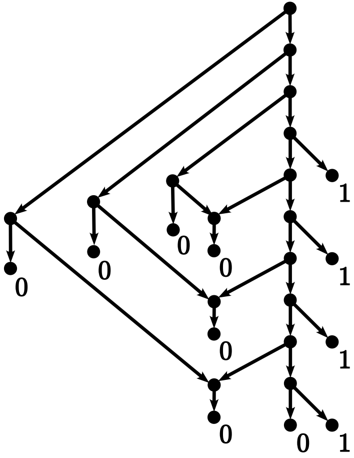
Hardwired and Softwired scores can be arbitrarily far apart



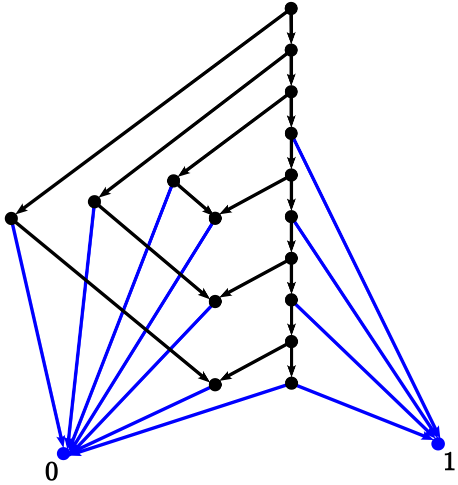
Softwired Parsimony Score = 2

Hardwired Parsimony Score = Hybridization Number + 1

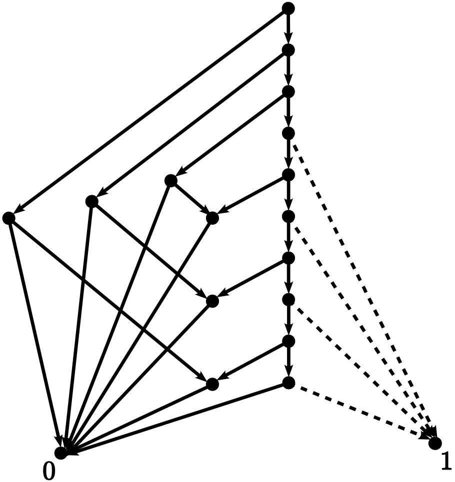
The **hardwired** parsimony score equals the size of a **minimum multiterminal cut** in the graph obtained by merging all leaves with the same state into a single vertex, and letting the merged vertices be the terminals.



The hardwired parsimony score equals the size of a **minimum multiterminal cut** in the graph obtained by merging all leaves with the same state into a single vertex, and letting the merged vertices be the terminals.



The hardwired parsimony score equals the size of a **minimum multiterminal cut** in the graph obtained by merging all leaves with the same state into a single vertex, and letting the merged vertices be the terminals.



- The **hardwired** parsimony score can be computed in polynomial time when there are two states,

- The **hardwired** parsimony score can be computed in polynomial time when there are two states,
- and approximated well when there are more than two states.

- The **hardwired** parsimony score can be computed in polynomial time when there are two states,
- and approximated well when there are more than two states.

Theorem (Fischer, vI, Kelk & Scornavacca, 2015)

For every constant $\epsilon > 0$ *there is* **no polynomial-time approximation algorithm** *that approximates the* **softwired** *parsimony score to a factor* $n^{1-\epsilon}$ *for a network and a binary character, unless* $P = NP$.

- The **hardwired** parsimony score can be computed in polynomial time when there are two states,
- and approximated well when there are more than two states.

Theorem (Fischer, vI, Kelk & Scornavacca, 2015)

For every constant $\epsilon > 0$ there is **no polynomial-time approximation algorithm** *that approximates the* **softwired** *parsimony score to a factor $n^{1-\epsilon}$ for a network and a binary character, unless $P = NP$.*

Luckily, the softwired parsimony score can be computed efficiently when the hybridization number (or "level") of the network is small.

Main open questions (from all parts)

- Is there is an FPT algorithm for HYBRIDIZATION NUMBER on multiple nonbinary trees and the hybridization number as only parameter.

Main open questions (from all parts)

- Is there is an FPT algorithm for HYBRIDIZATION NUMBER on multiple nonbinary trees and the hybridization number as only parameter.
- Which classes of networks are encoded by trinets?

Main open questions (from all parts)

- Is there is an FPT algorithm for HYBRIDIZATION NUMBER on multiple nonbinary trees and the hybridization number as only parameter.
- Which classes of networks are encoded by trinets?
- How can we search for a network with optimal softwired parsimony score, over all networks with hybridization number at most k?

Thanks

- Mareike Fischer (Greifswald)
- Katharina Huber (Norwich)
- Steven Kelk (Maastricht)
- Nela Lekić (Maastricht)
- Simone Linz (Christchurch)
- Vincent Moulton (Norwich)
- Celine Scornavacca (Montpellier)
- Leen Stougie (Amsterdam)
- Taoyang Wu (Norwich)