Fixed-Parameter Algorithms for Finding Agreement Supertrees

David Fernández-Baca, Sylvain Guillemot, Brad Shutters & Sudheer Vakati
Computer Science Department, Iowa State University

Building Phylogenies from Gene Sequences

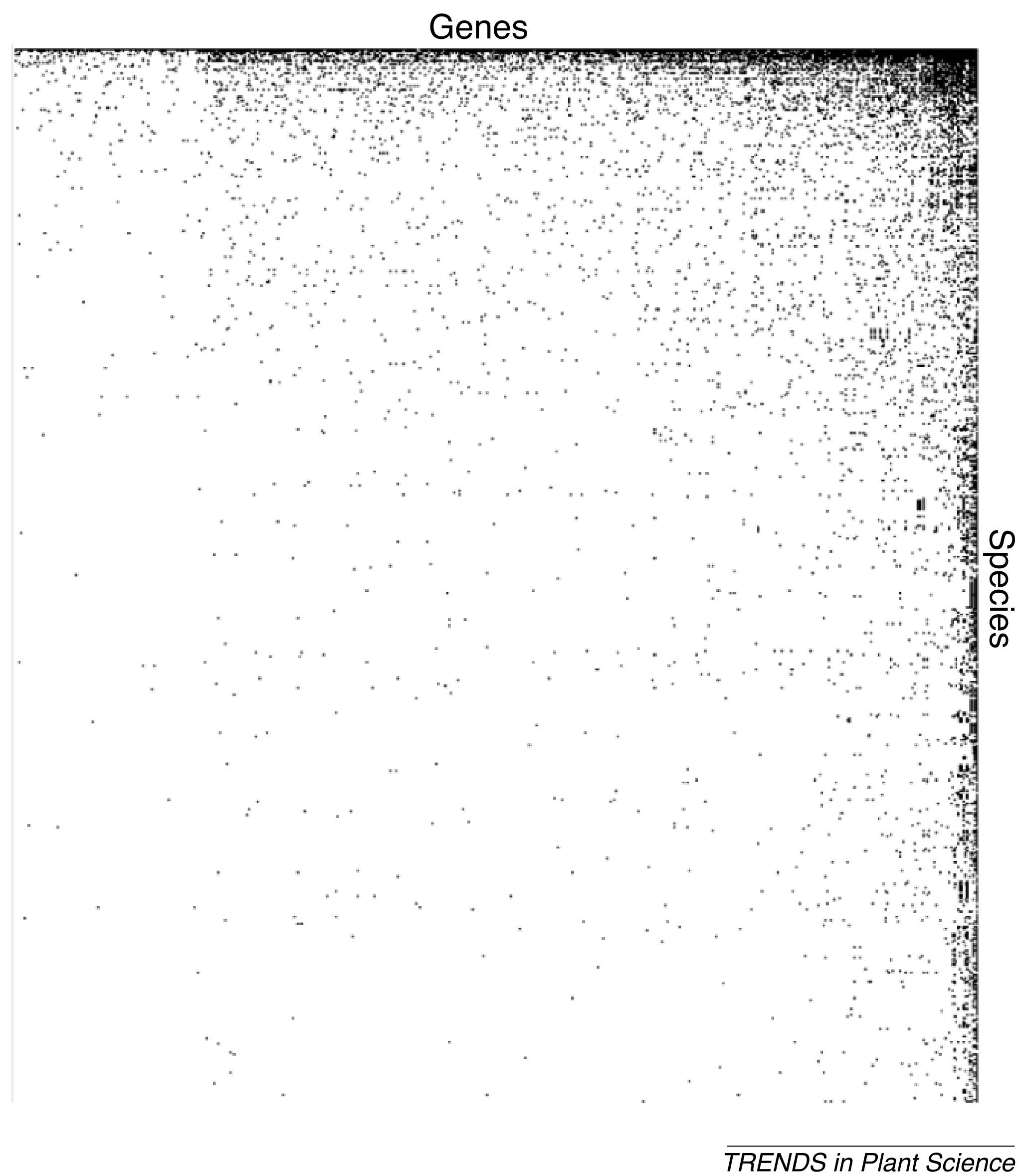
Building Phylogenies from Gene Sequences

- **Ideal case:** Use genes common to all the taxa (species).

Building Phylogenies from Gene Sequences

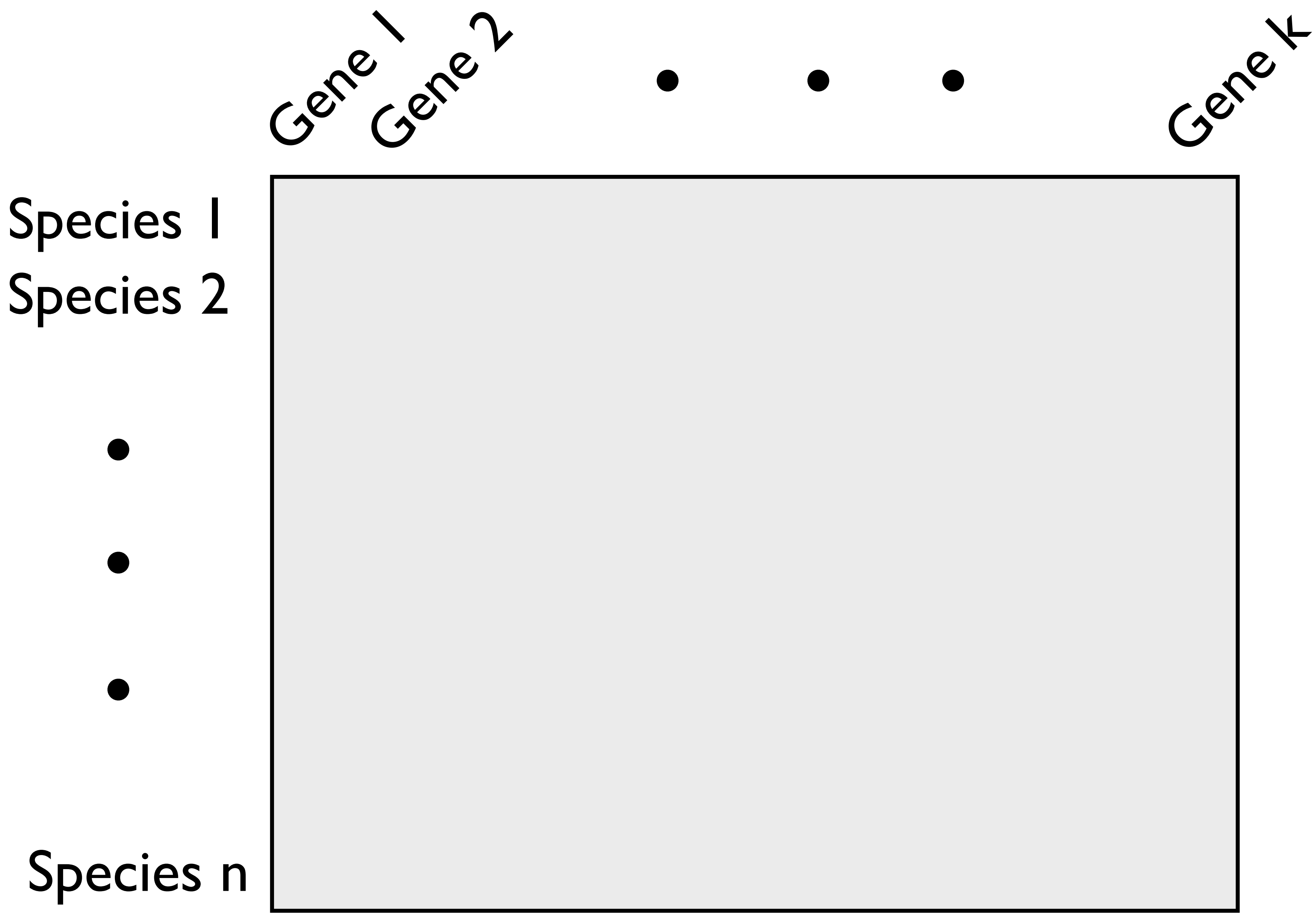
- **Ideal case:** Use genes common to all the taxa (species).
- **Reality:** Low sampling density, number of common genes is very limited.

Low Sampling Density

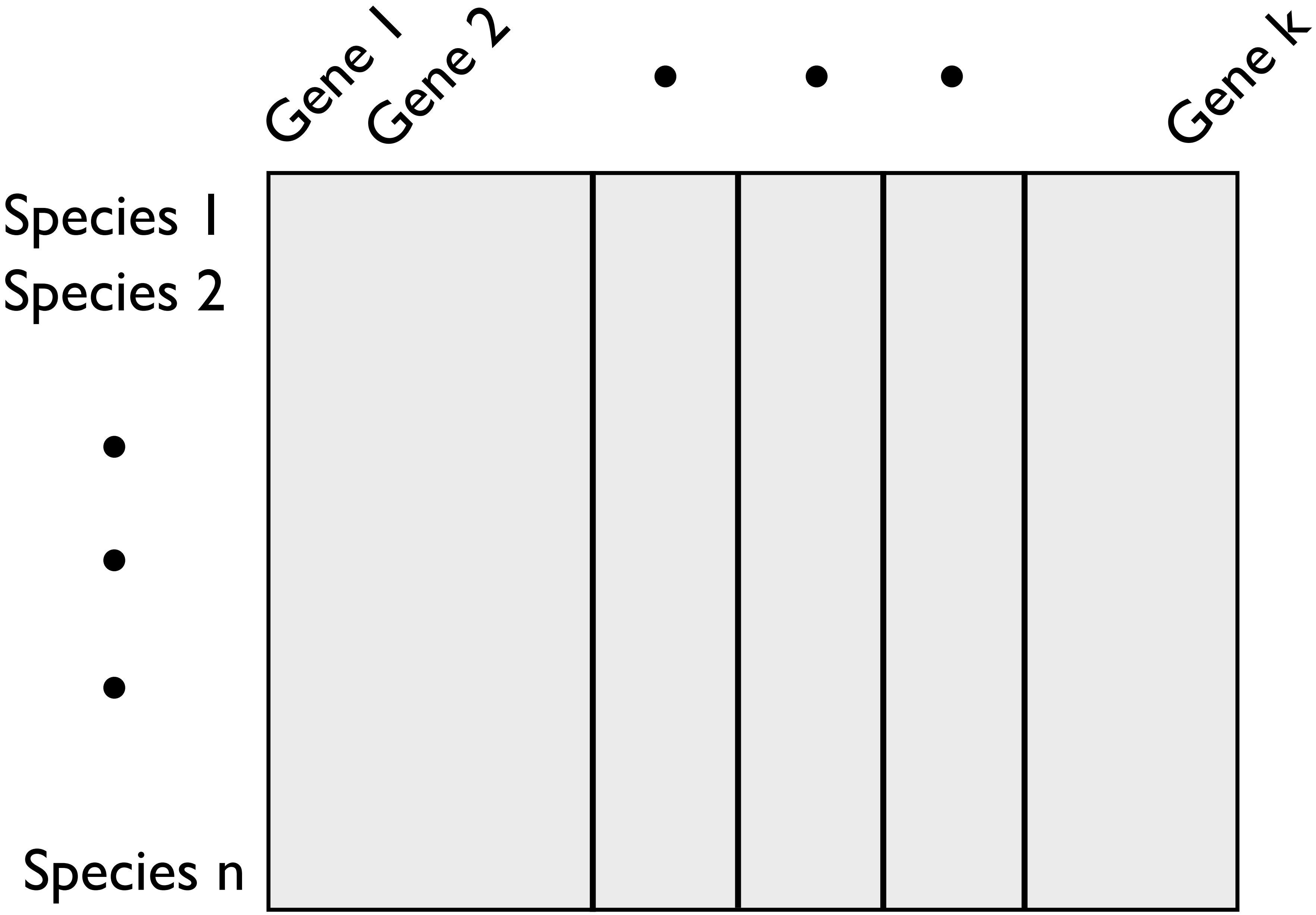


Sanderson & Driskell, 2003

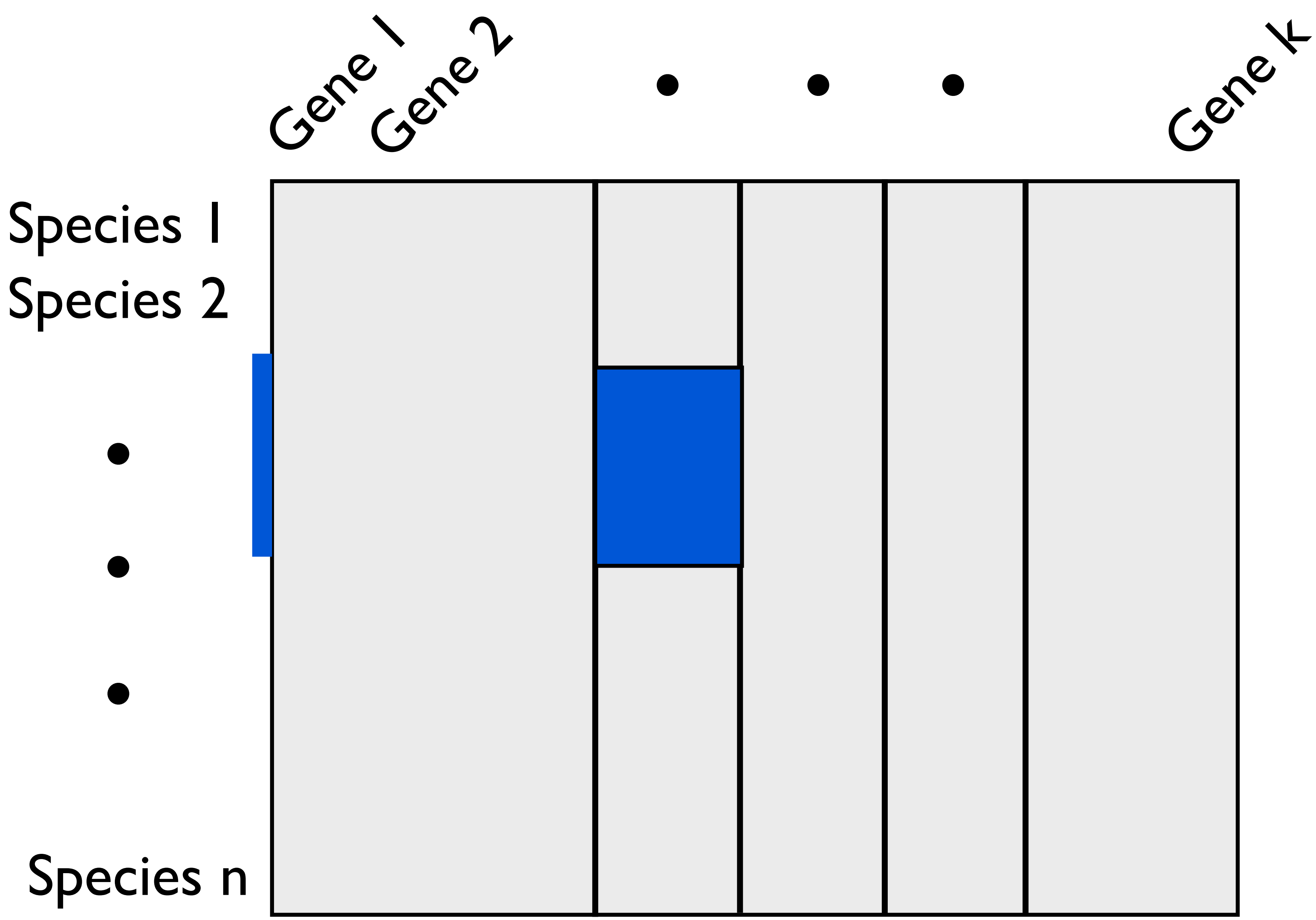
Supertrees



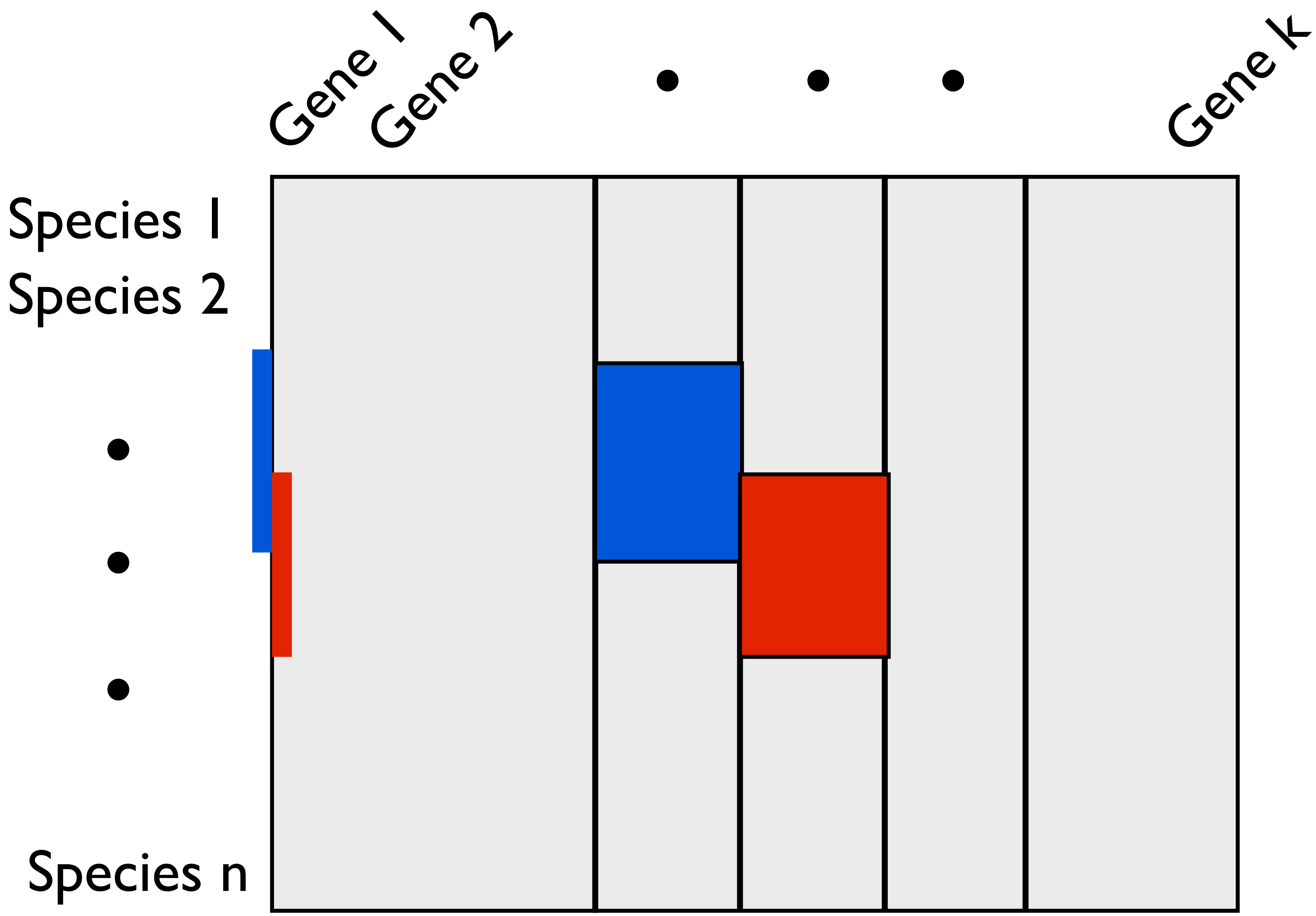
Supertrees



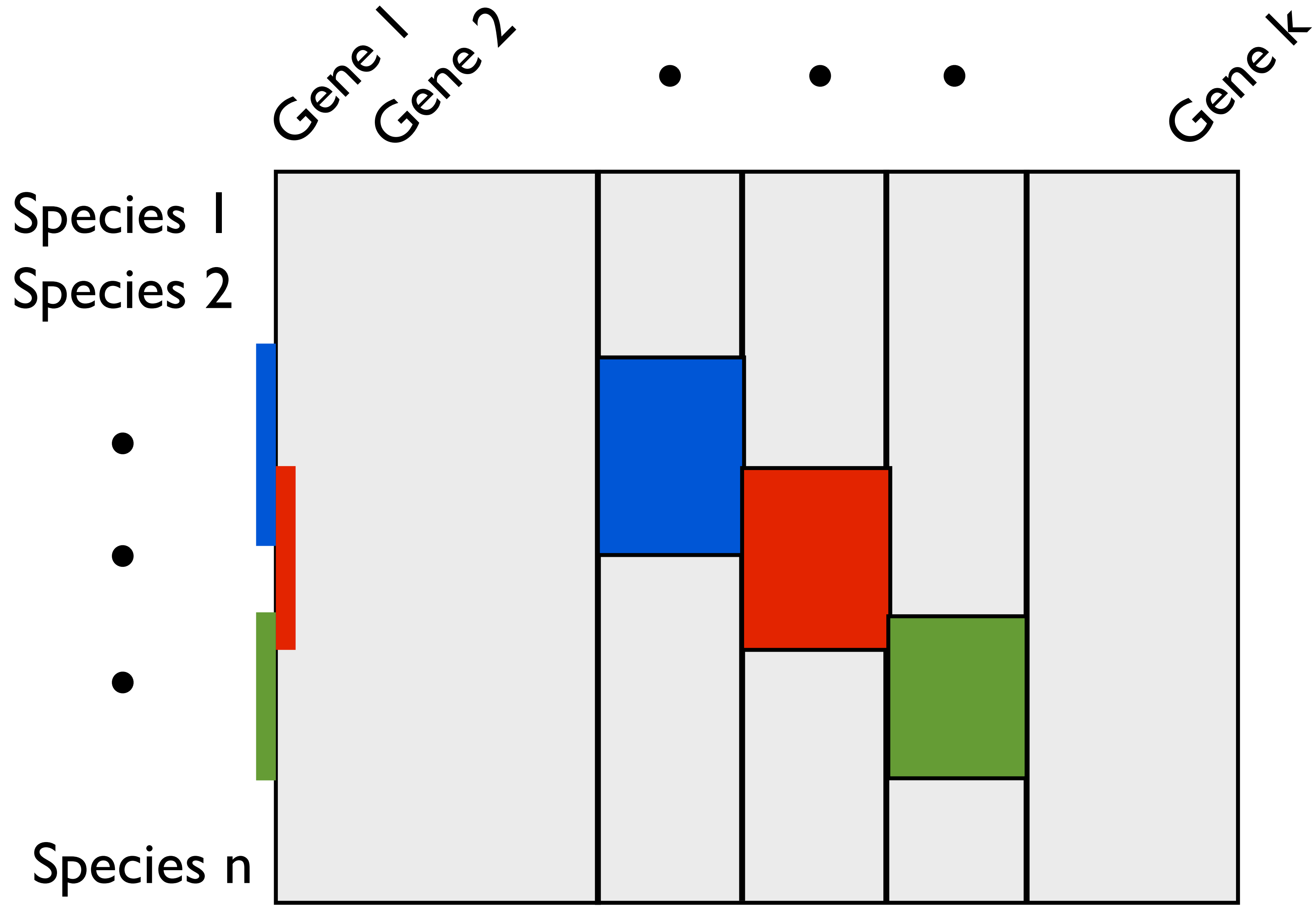
Supertrees



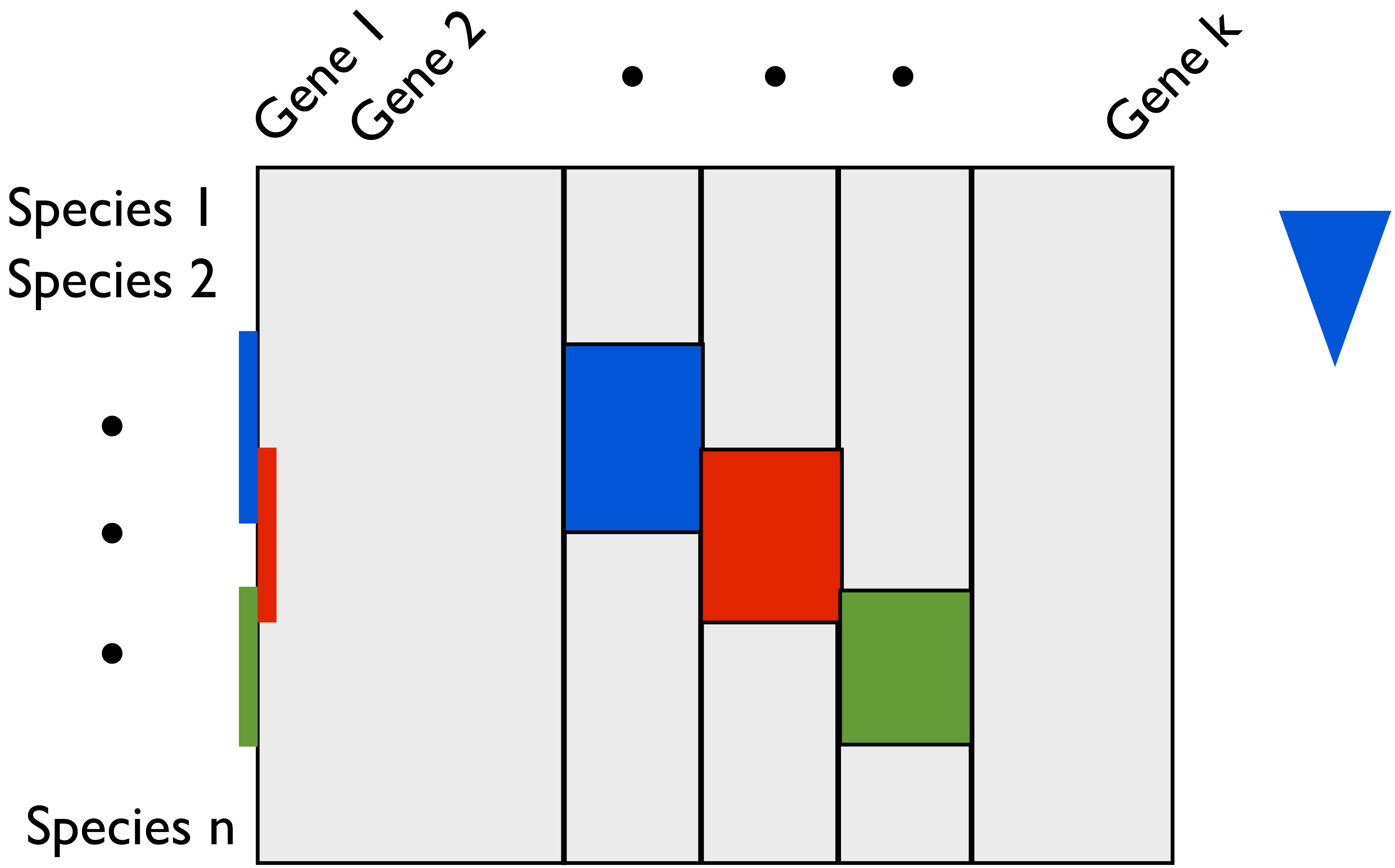
Supertrees



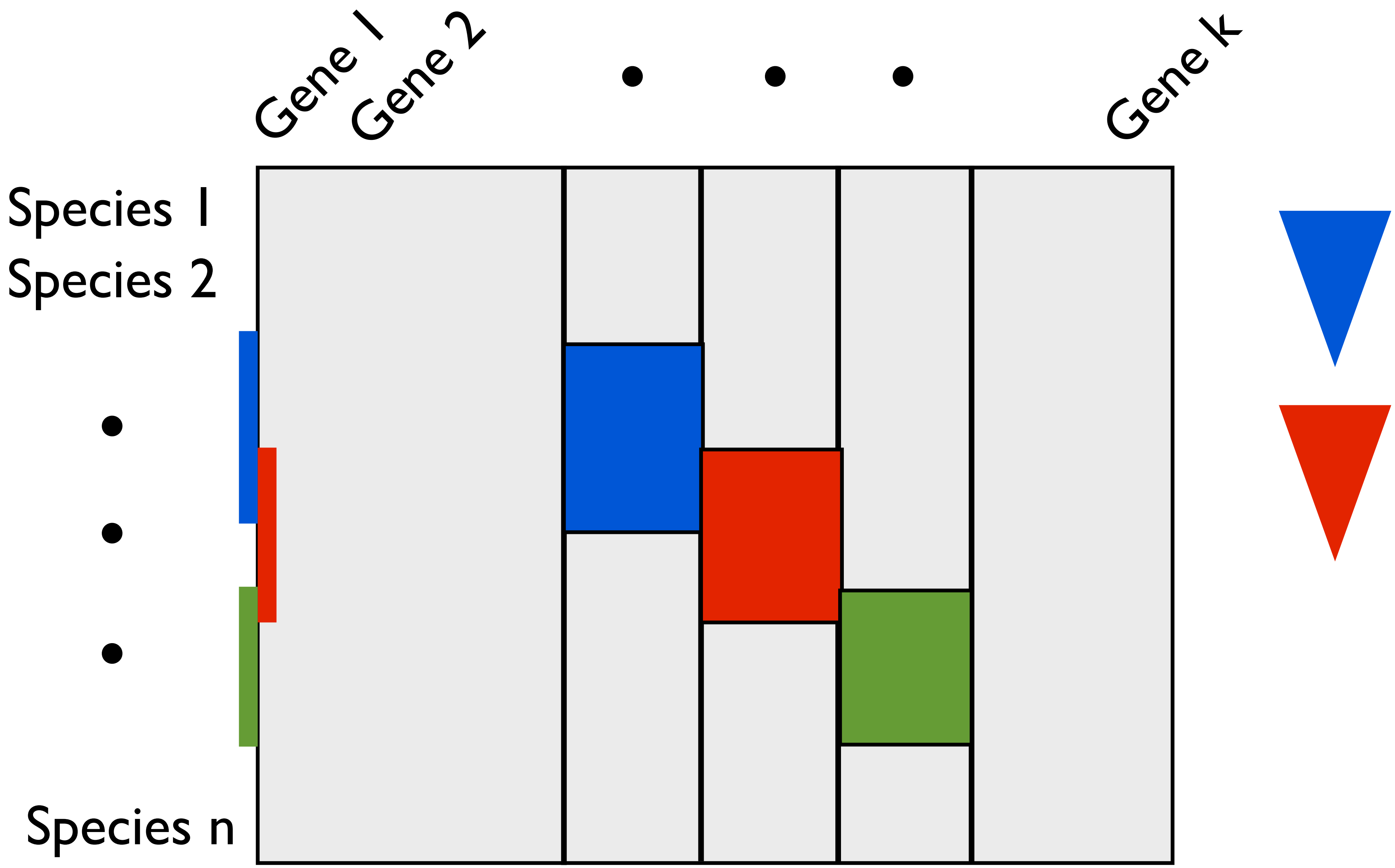
Supertrees



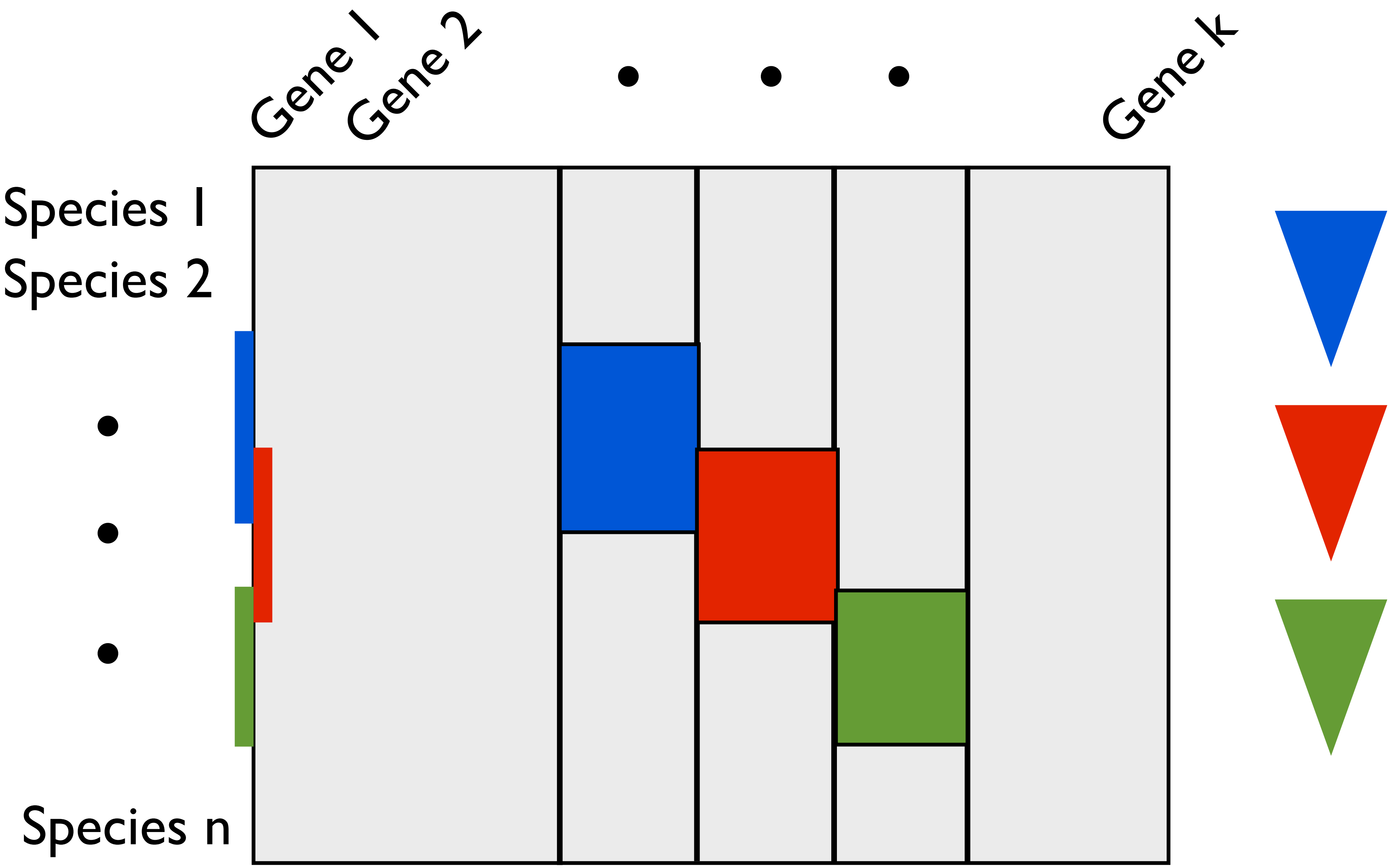
Supertrees



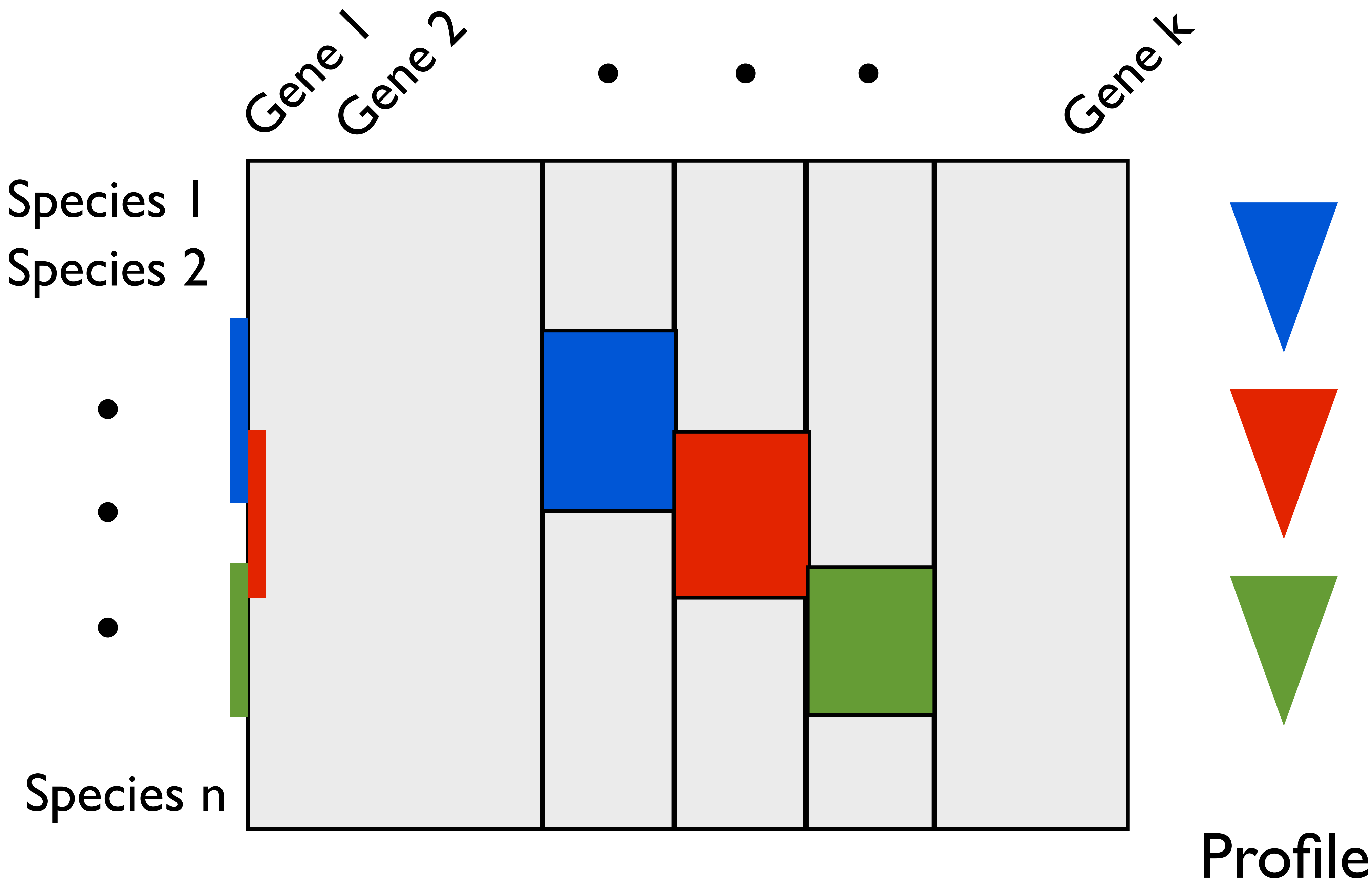
Supertrees



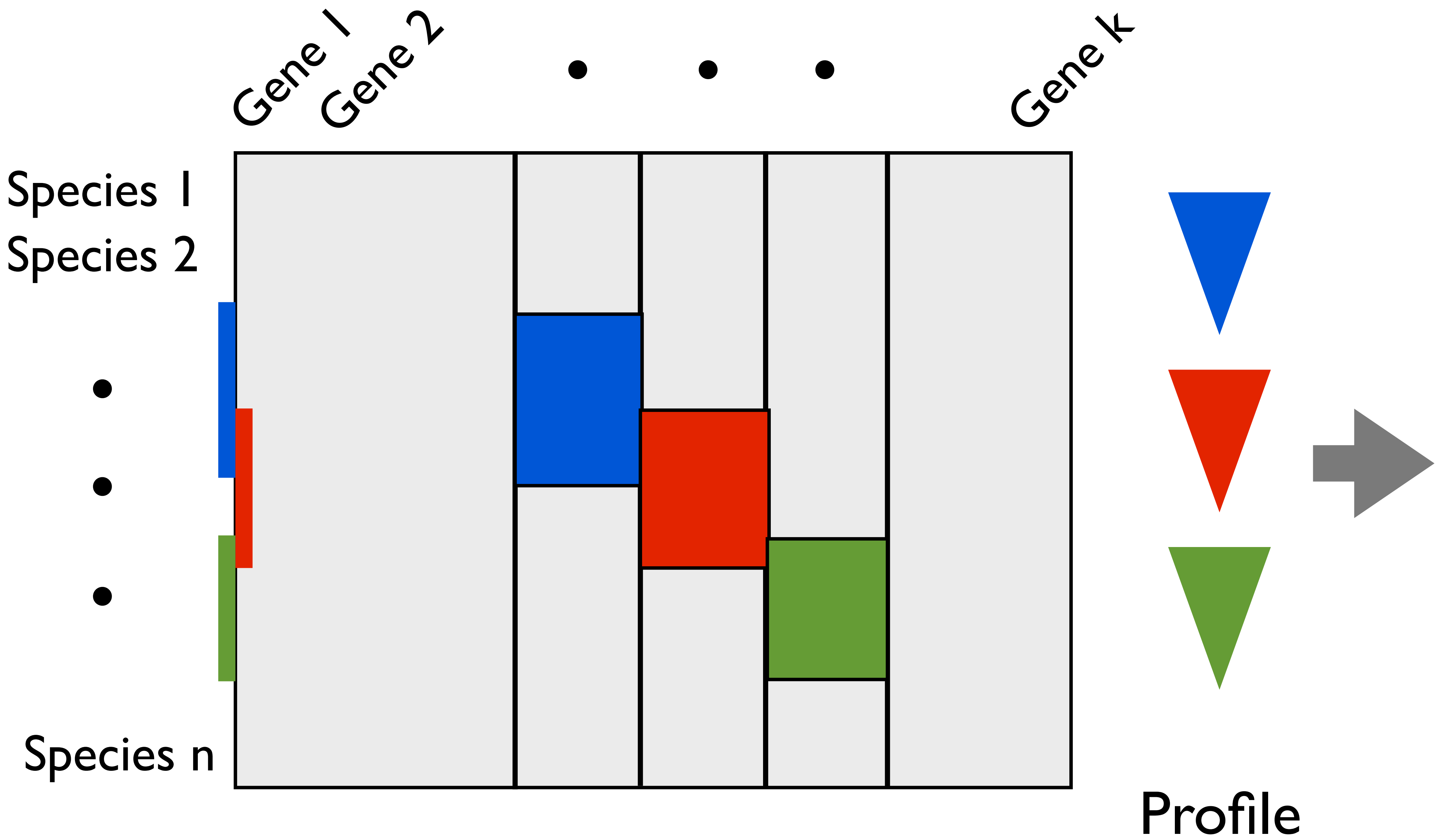
Supertrees



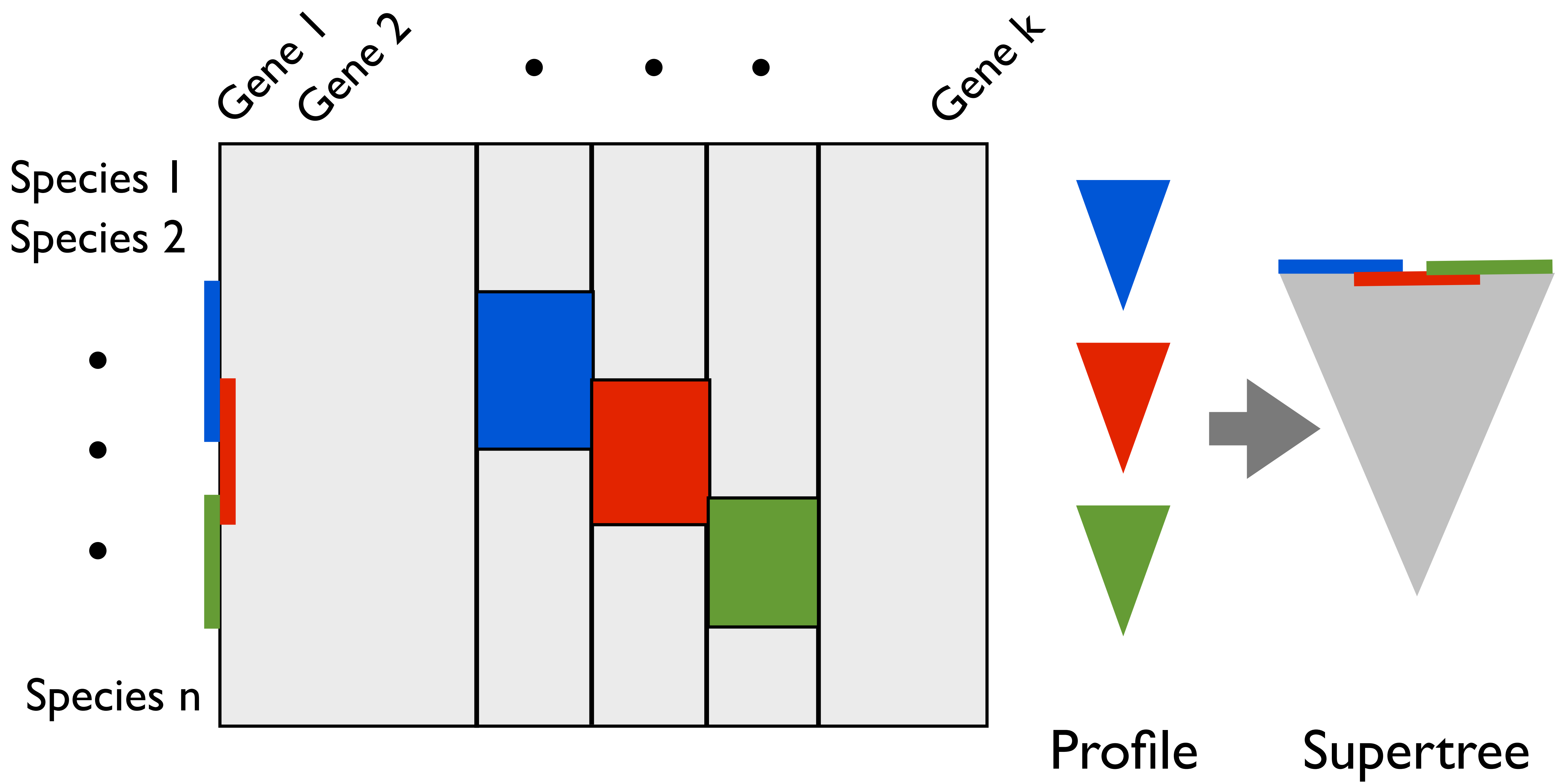
Supertrees



Supertrees



Supertrees


Gene 1
Gene 2
Gene k
Species 1
Species 2
Species n
Profile
Supertree

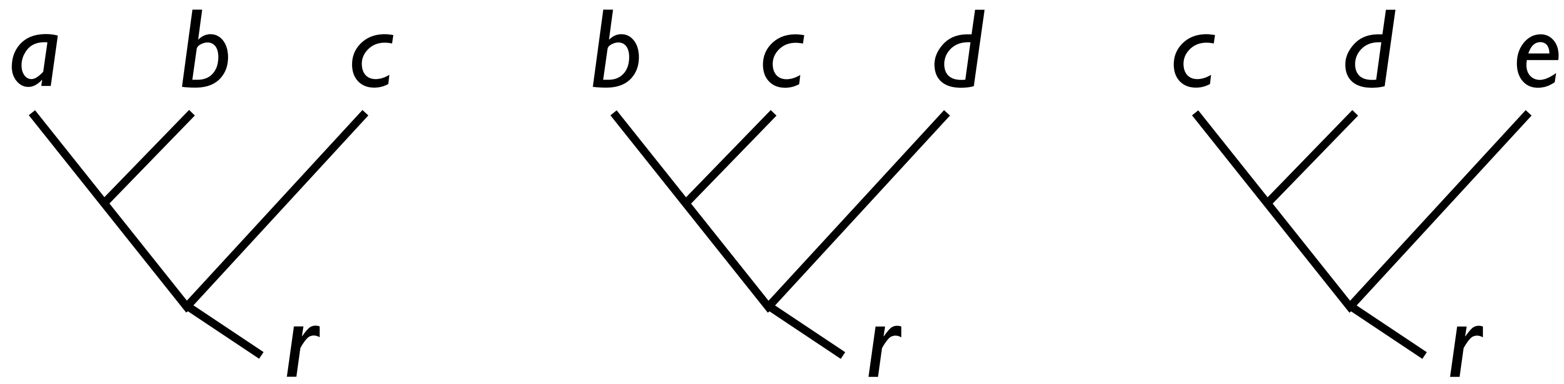
Agreement Supertrees

Agreement Supertrees

Input
Trees

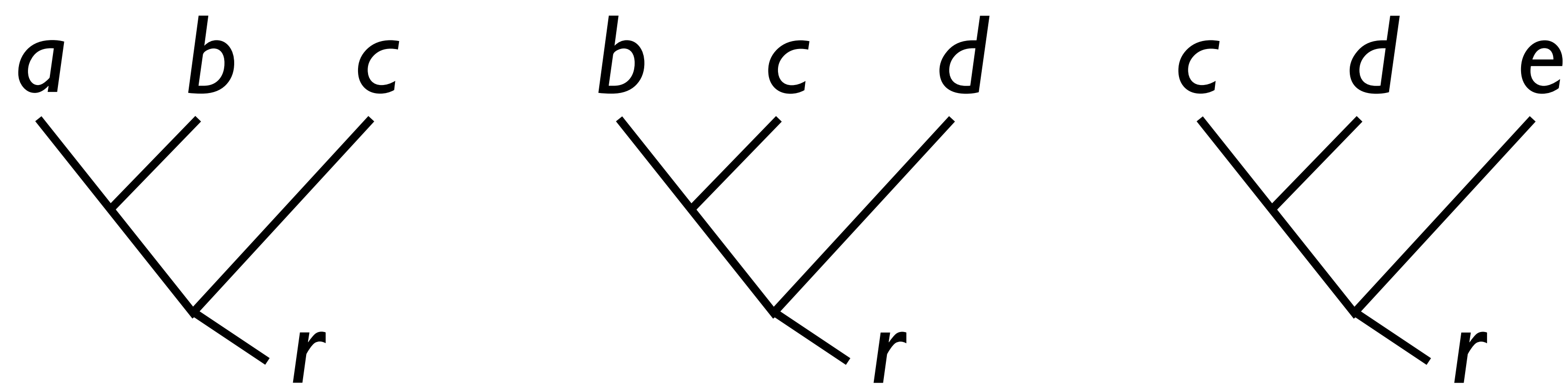
Agreement Supertrees

Input Trees



Agreement Supertrees

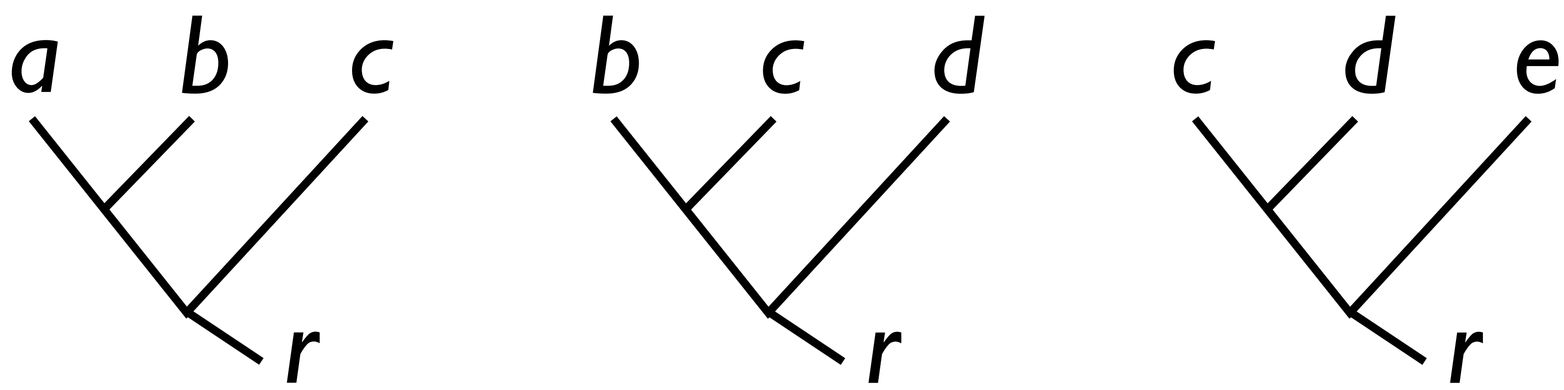
Input Trees



(Profile)

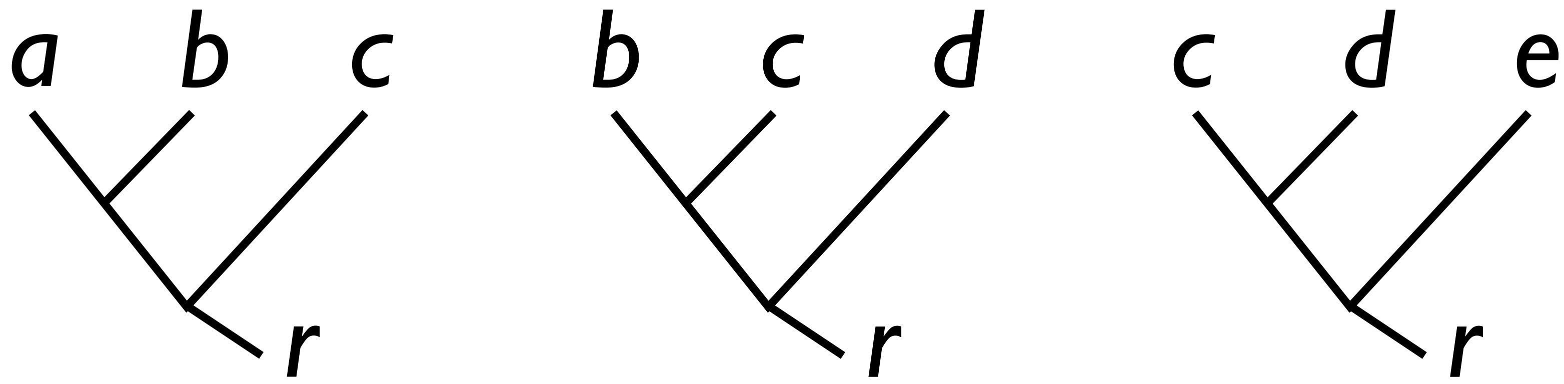
Agreement Supertrees

Input Trees



Agreement Supertrees

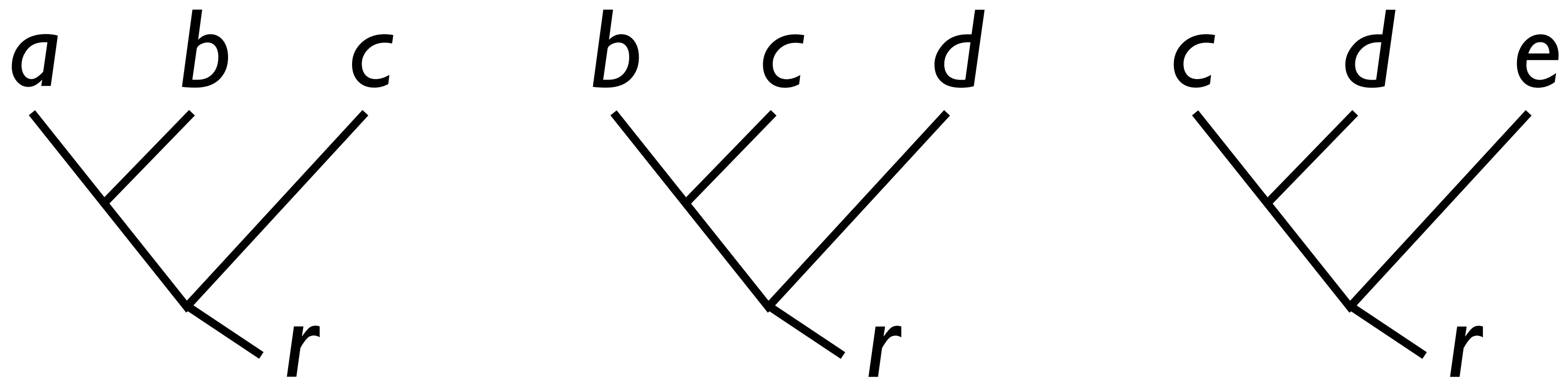
Input Trees



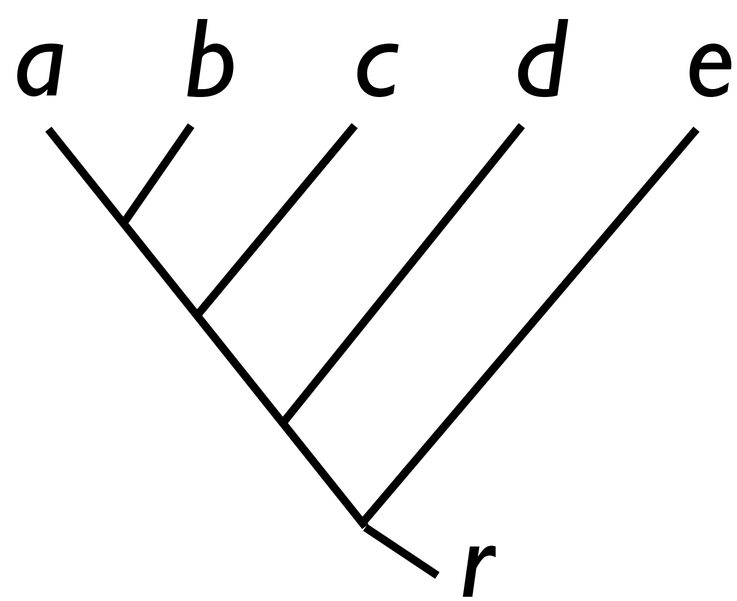
Supertree

Agreement Supertrees

Input Trees

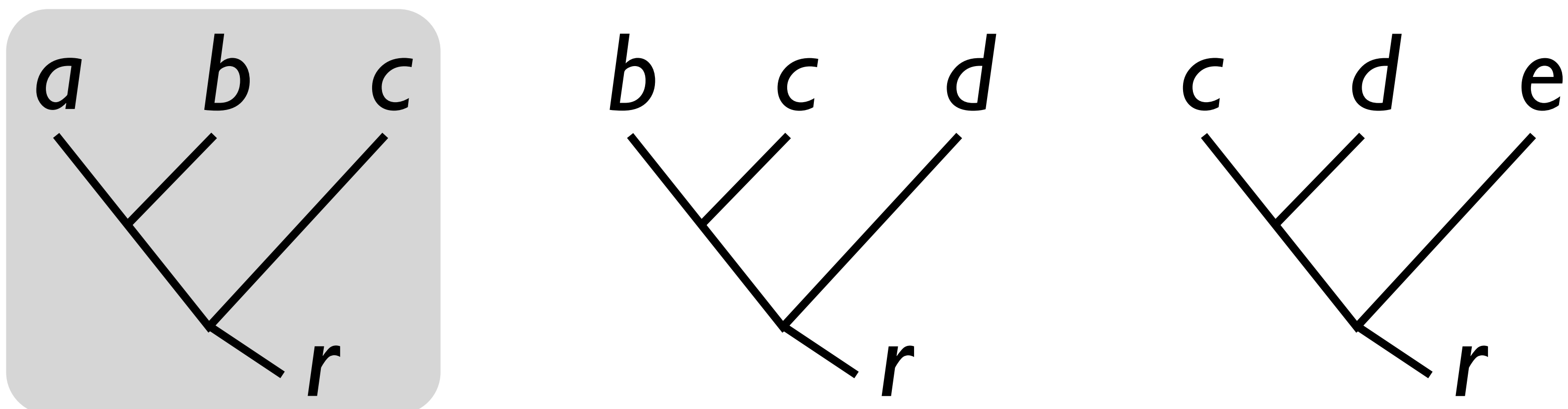


Supertree

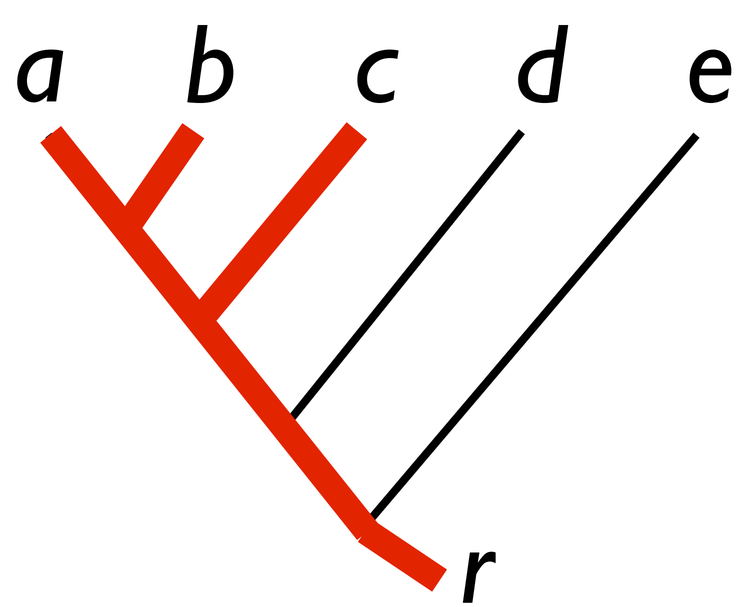


Agreement Supertrees

Input Trees

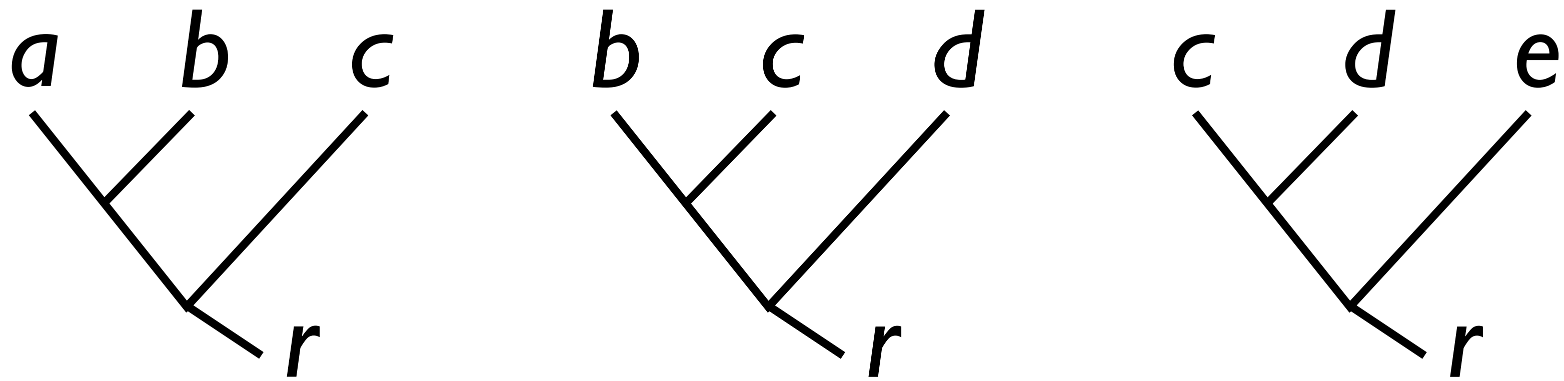


Supertree

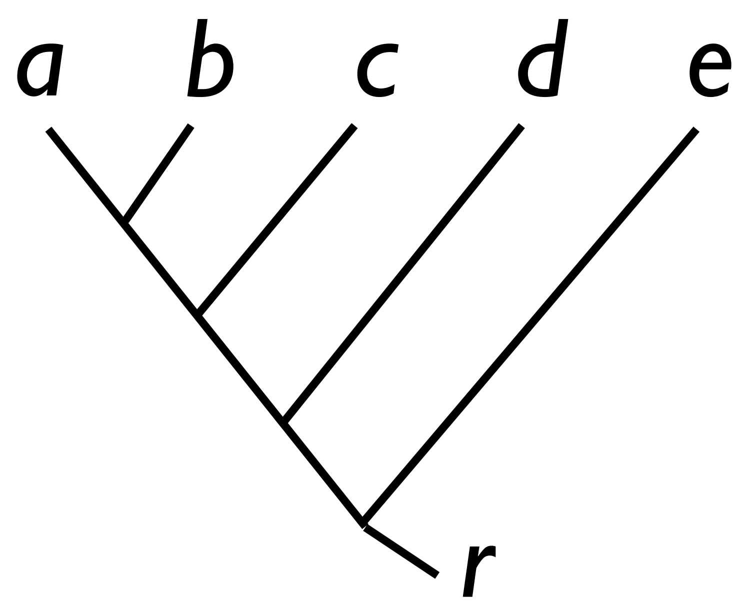


Agreement Supertrees

Input Trees

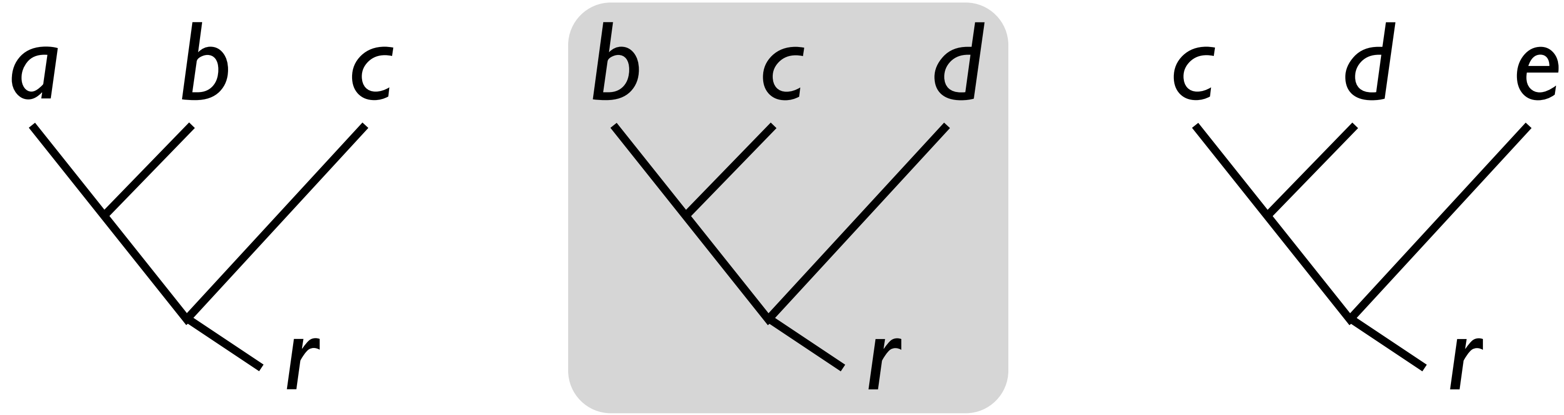


Supertree

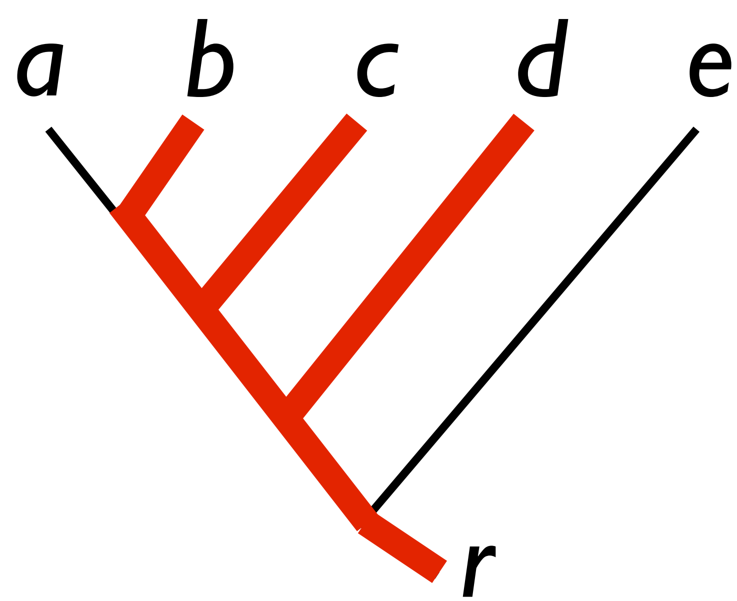


Agreement Supertrees

Input Trees

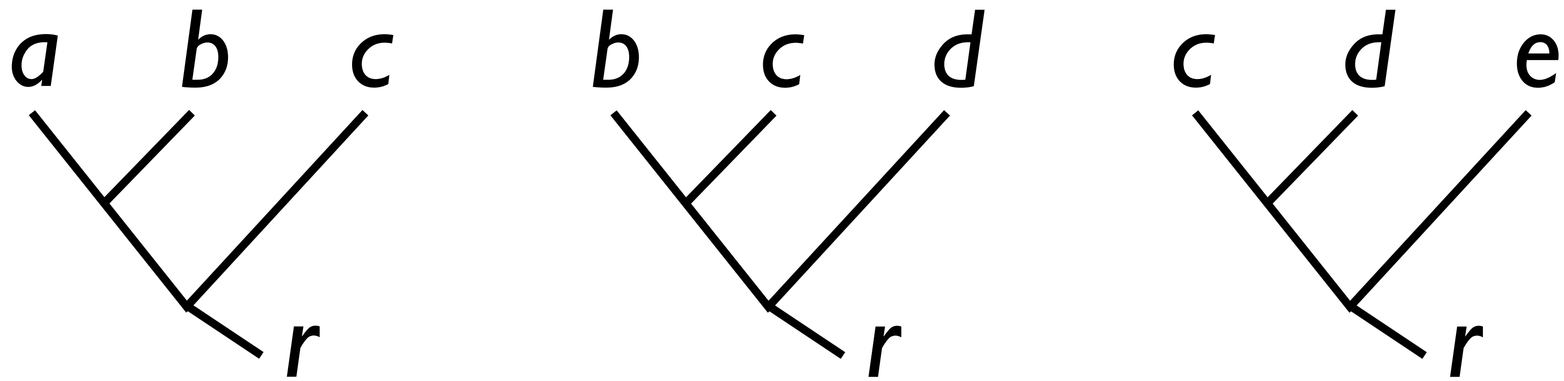


Supertree

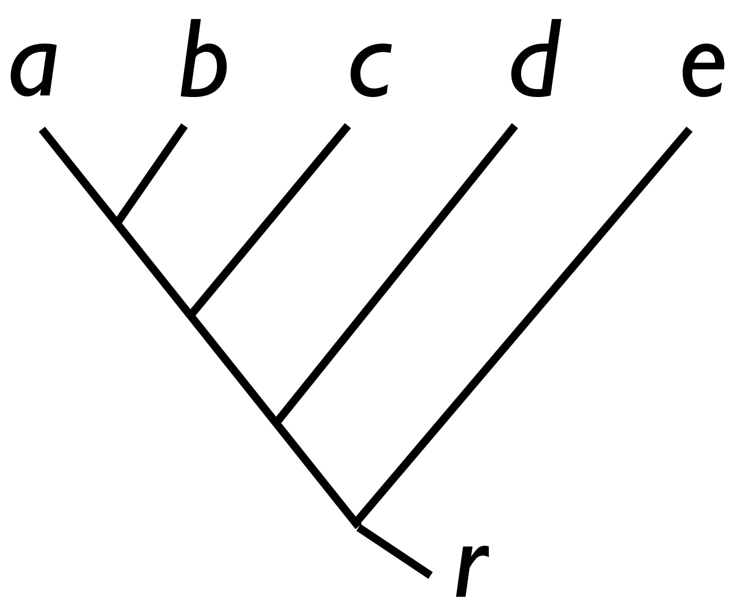


Agreement Supertrees

Input Trees

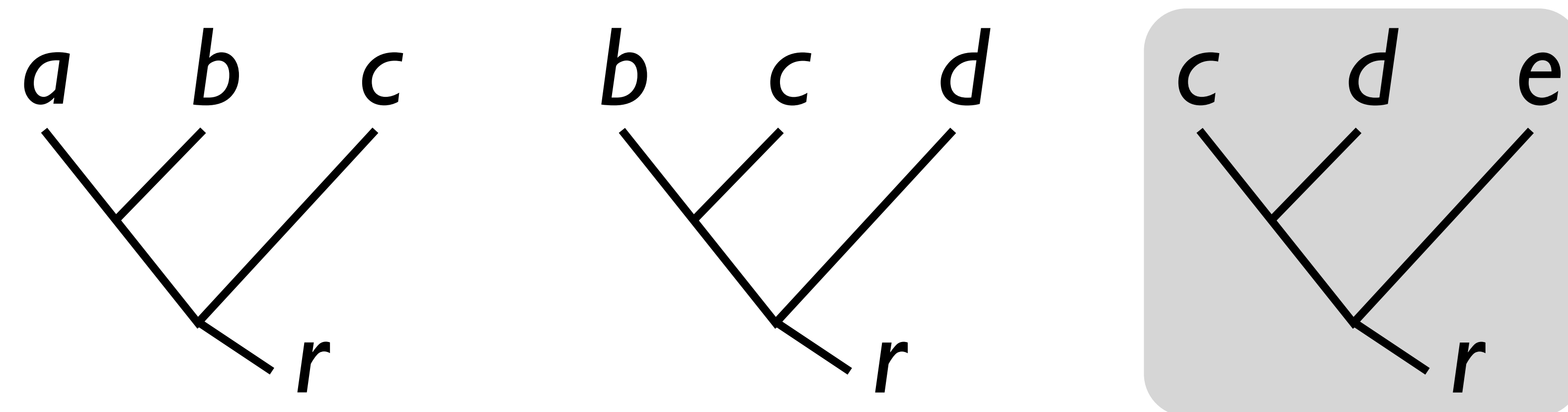


Supertree

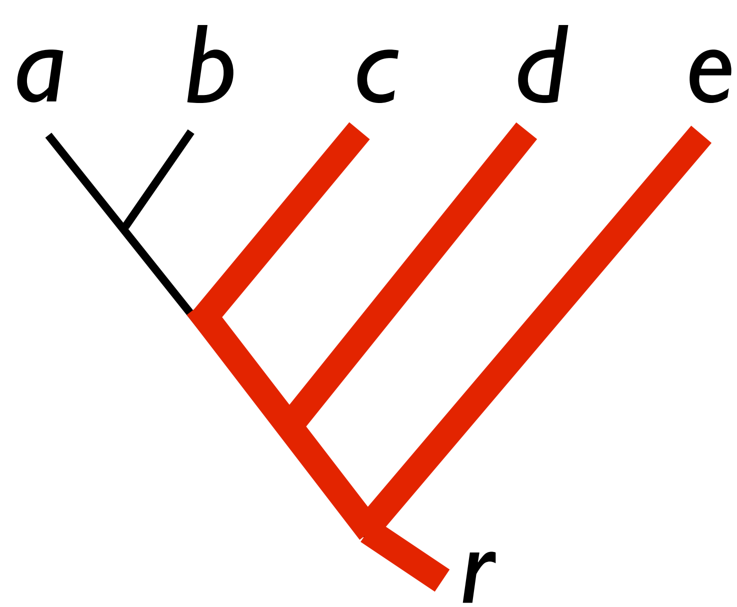


Agreement Supertrees

Input Trees

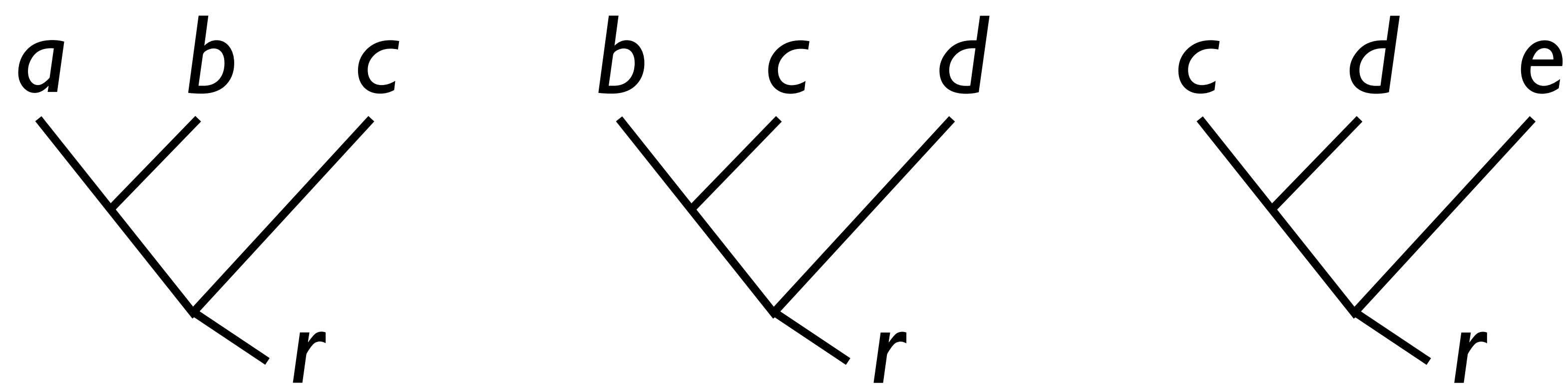


Supertree

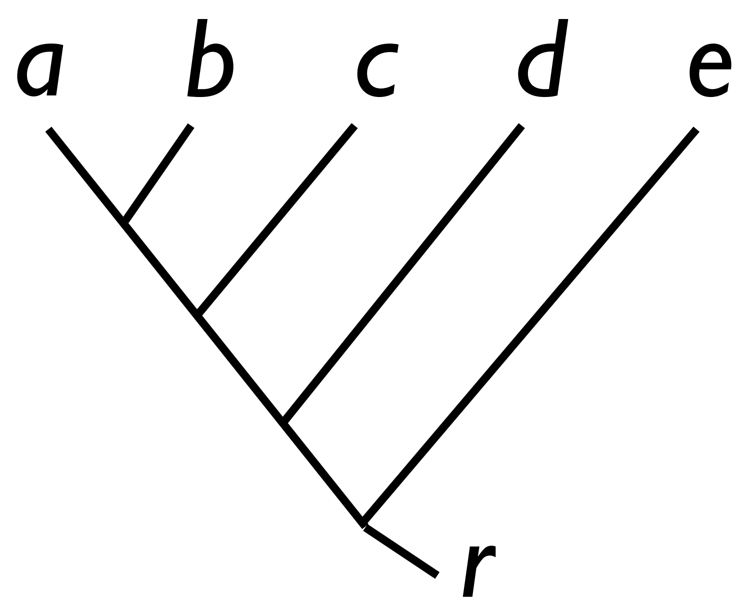


Agreement Supertrees

Input Trees

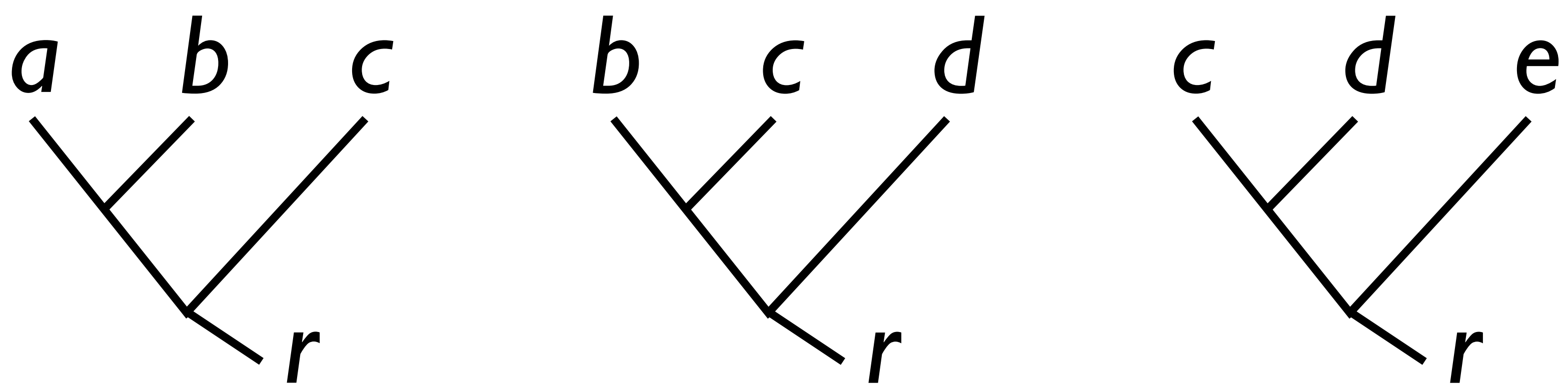


Supertree

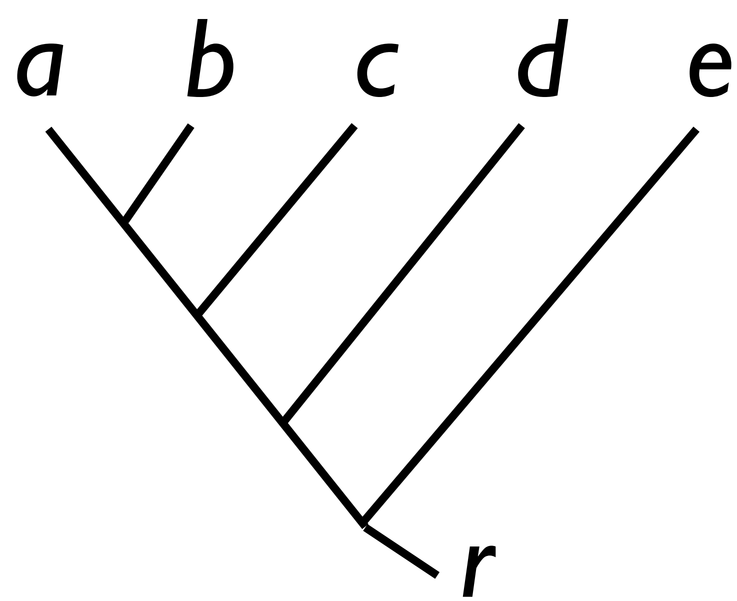


Agreement Supertrees

Input Trees



Supertree



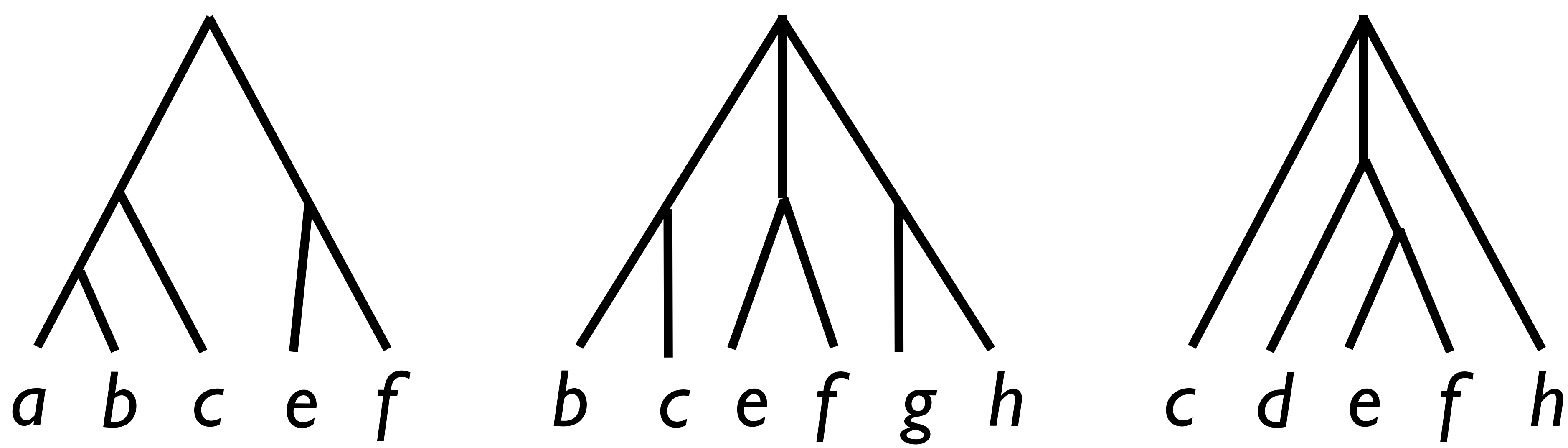
The input trees agree.

Agreement Supertrees: Definition

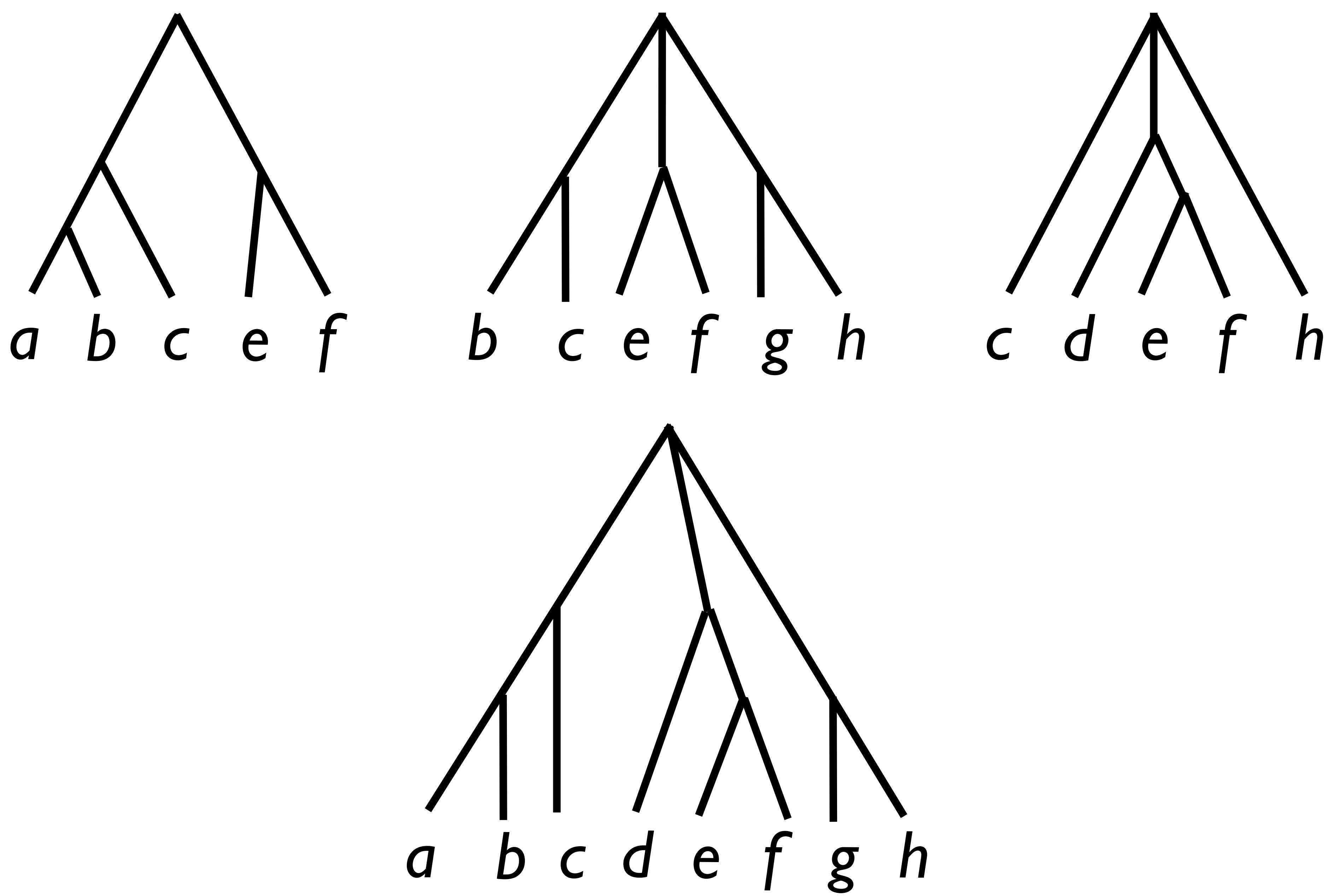
An agreement supertree for a profile $P = (T_1, T_2, \ldots, T_k)$ of rooted phylogenetic trees on n taxa is a tree S such that

- the leaf set of S is the union of the leaf sets of the trees in P and
- for each i, the restriction of S to the leaf set of T_i is T_i itself.

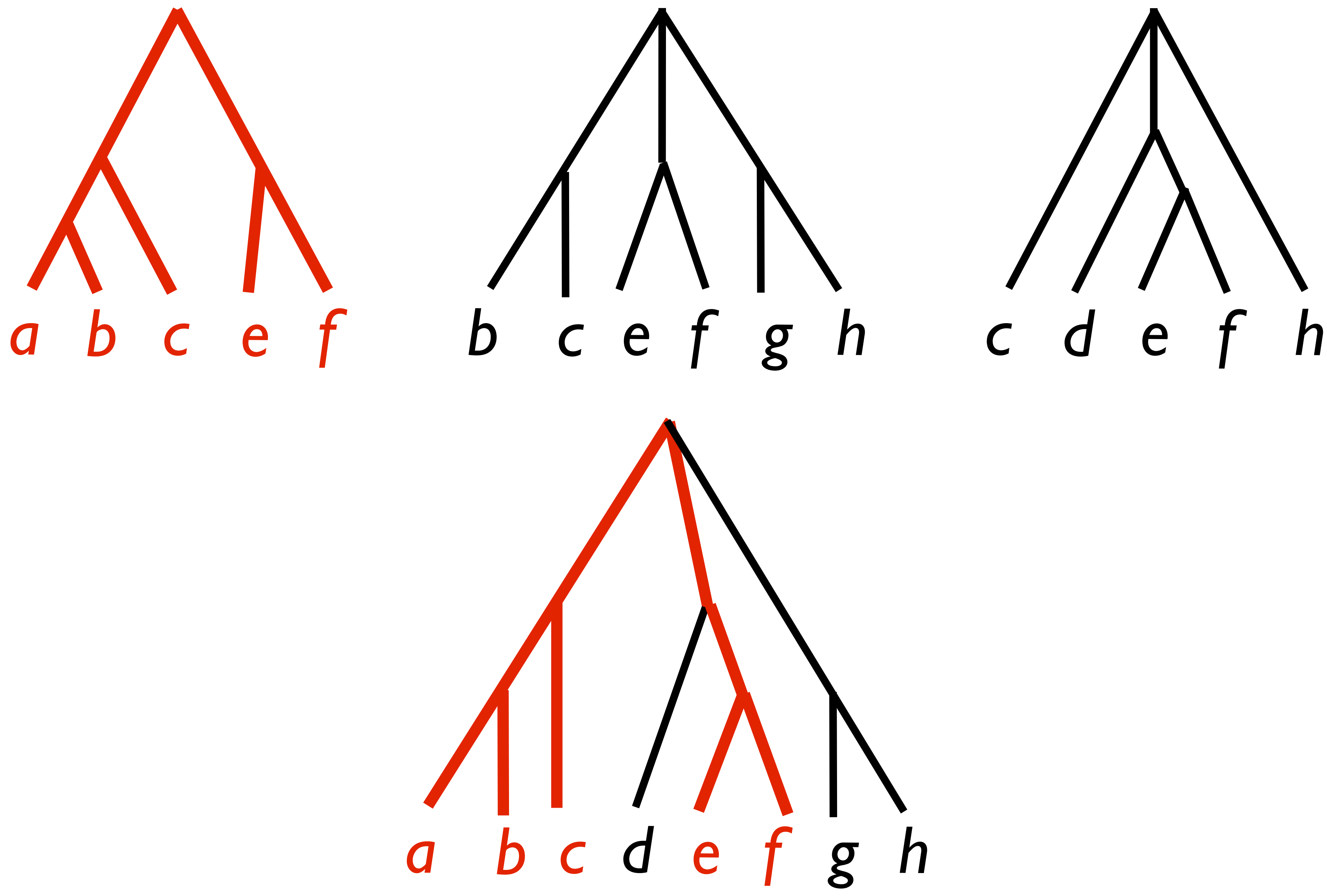
Agreement Supertrees



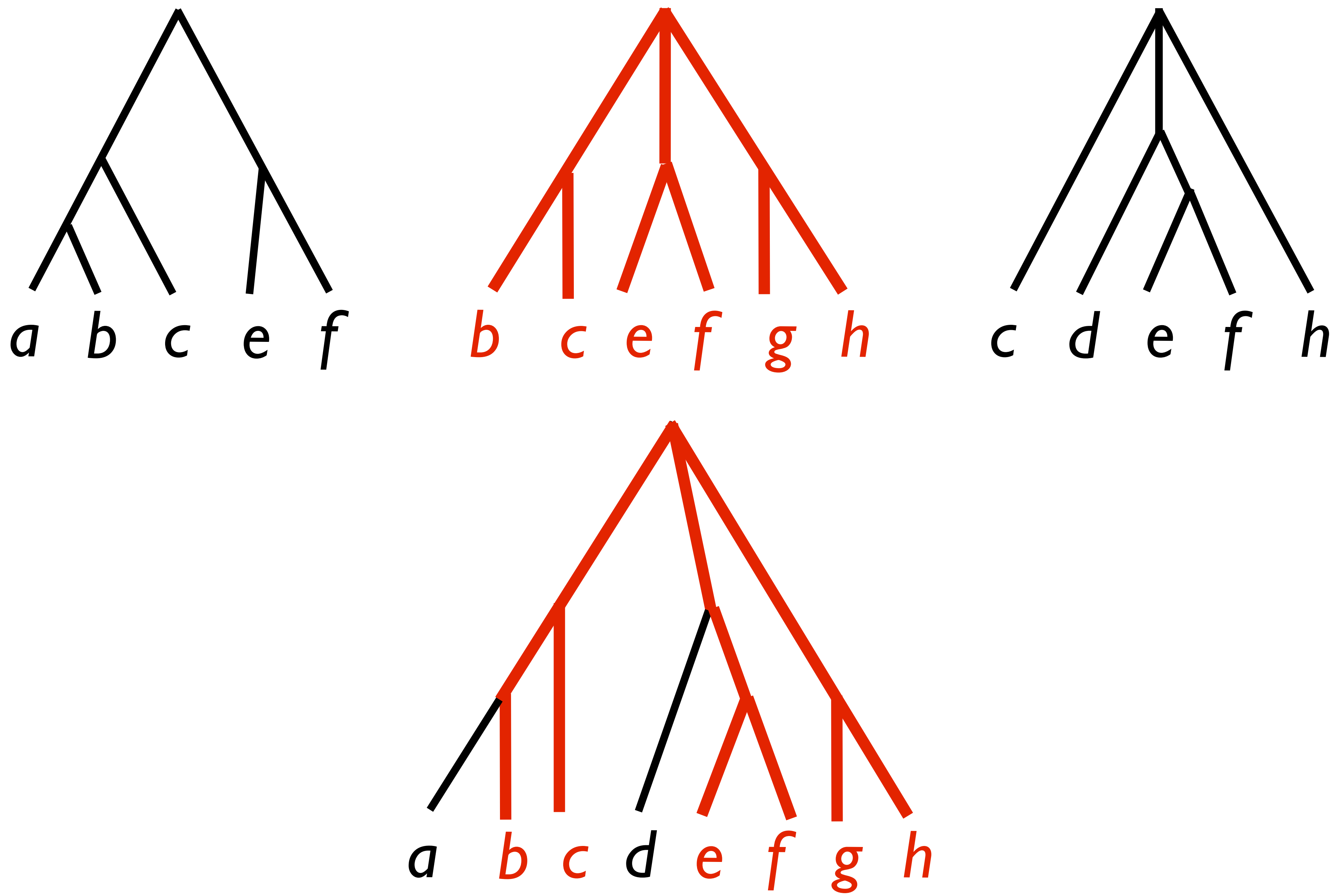
Agreement Supertrees



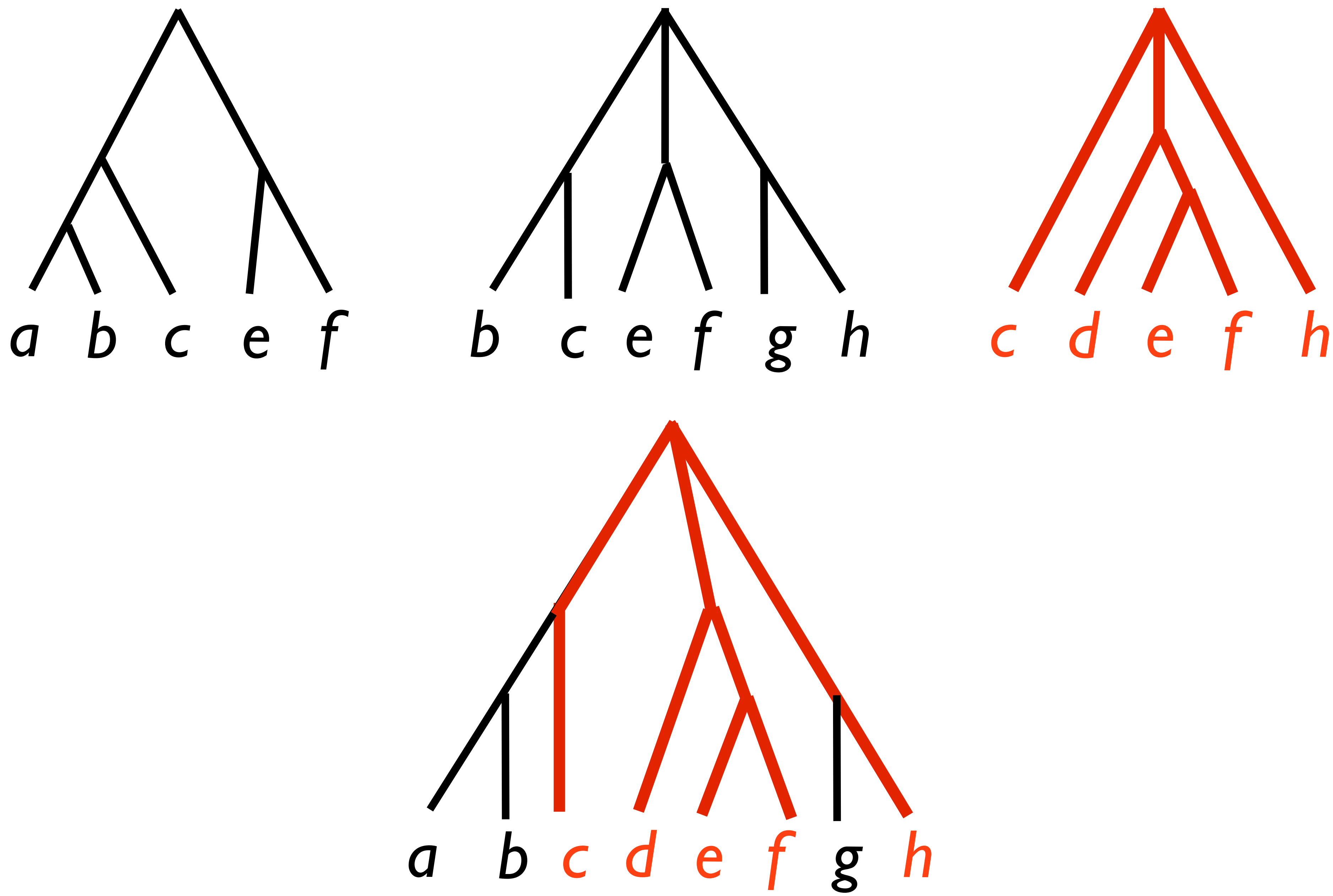
Agreement Supertrees



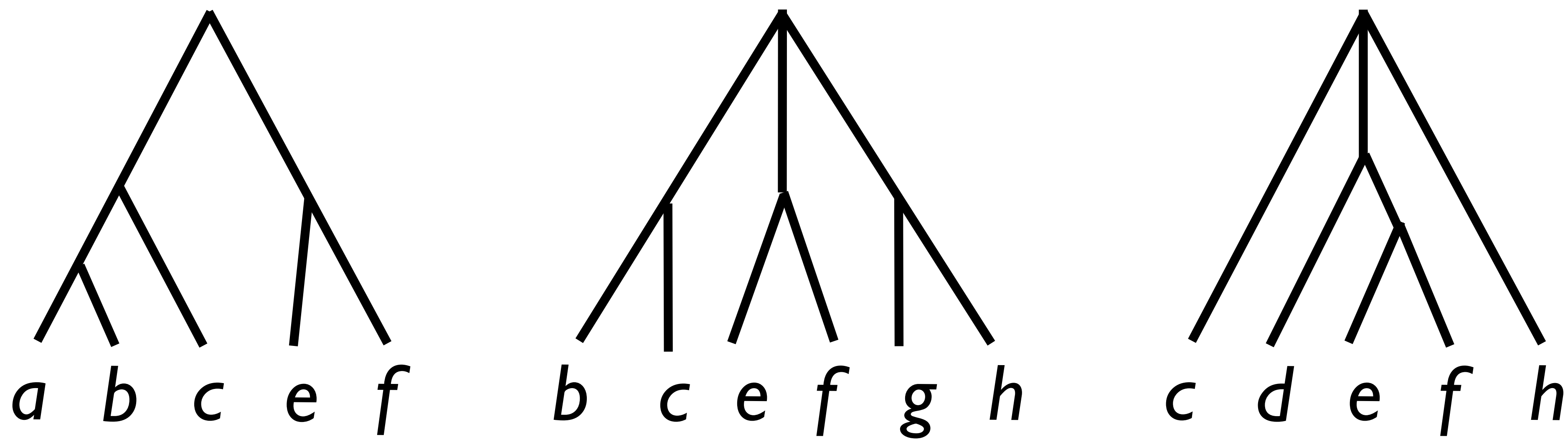
Agreement Supertrees

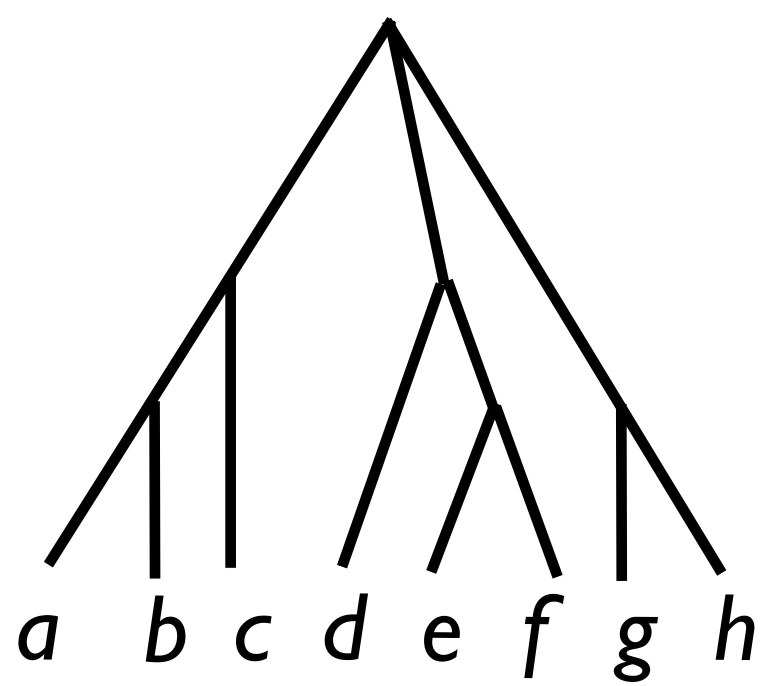


Agreement Supertrees



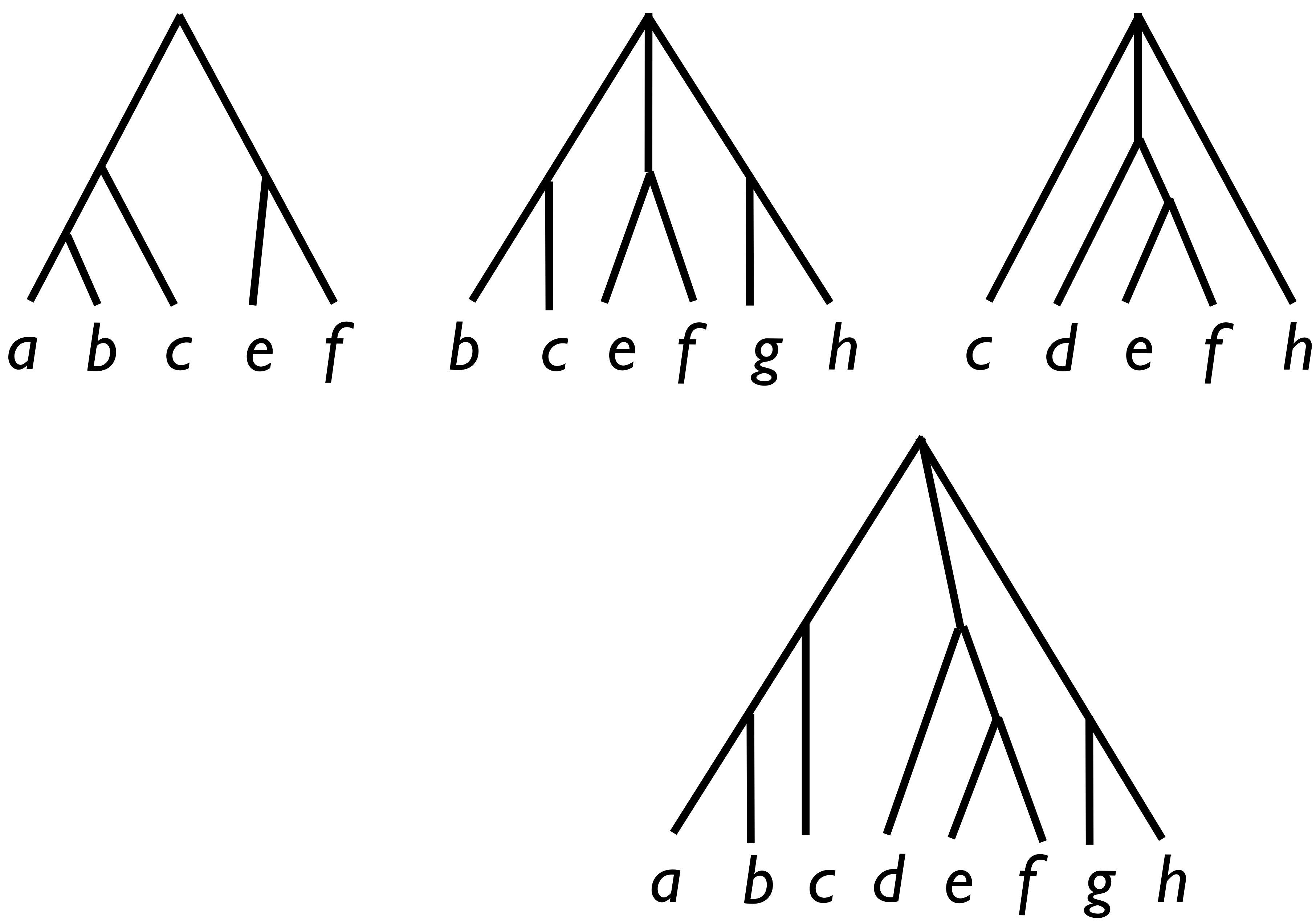
Agreement Supertrees



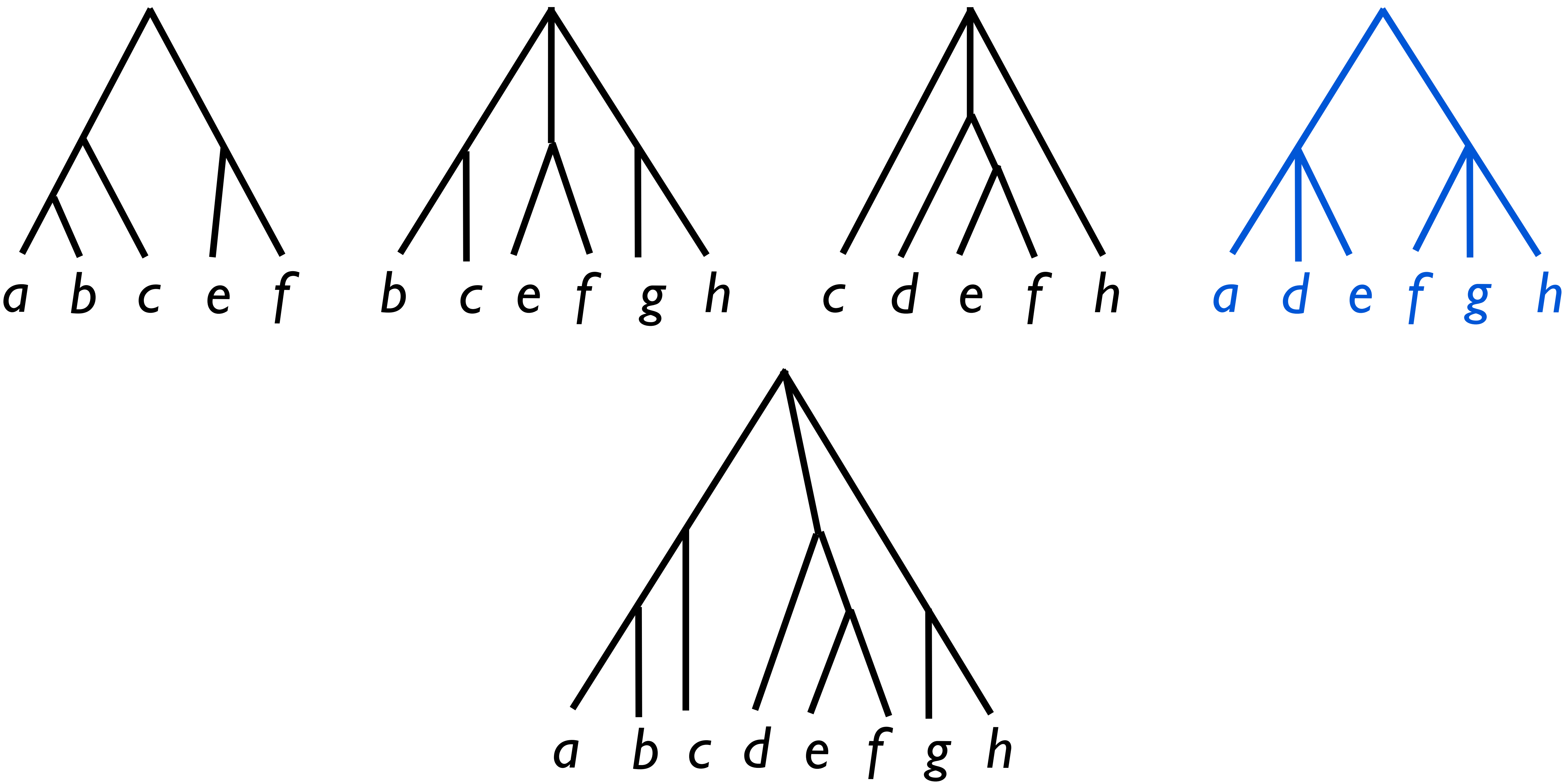


The trees agree

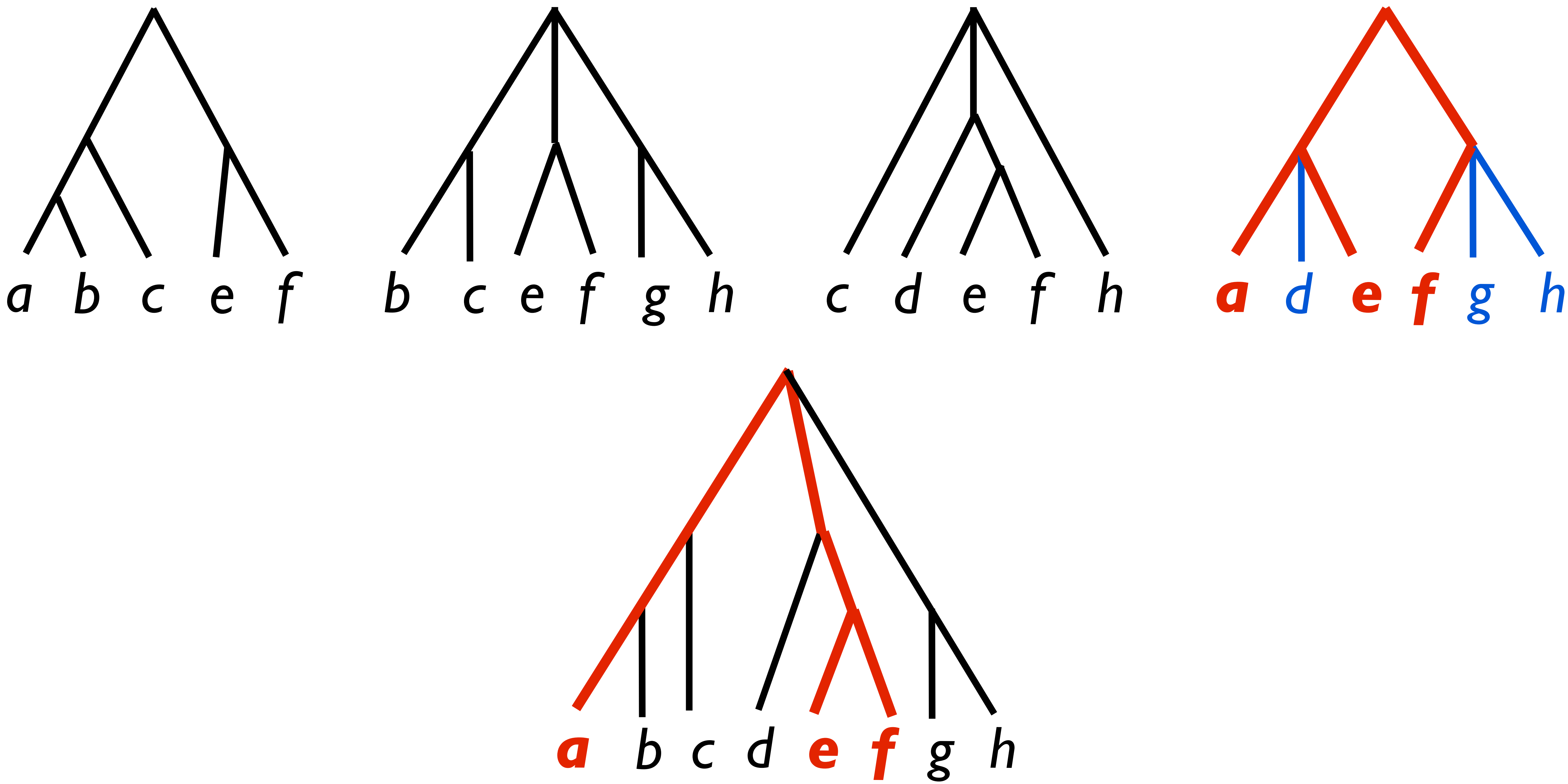
Agreement Supertrees



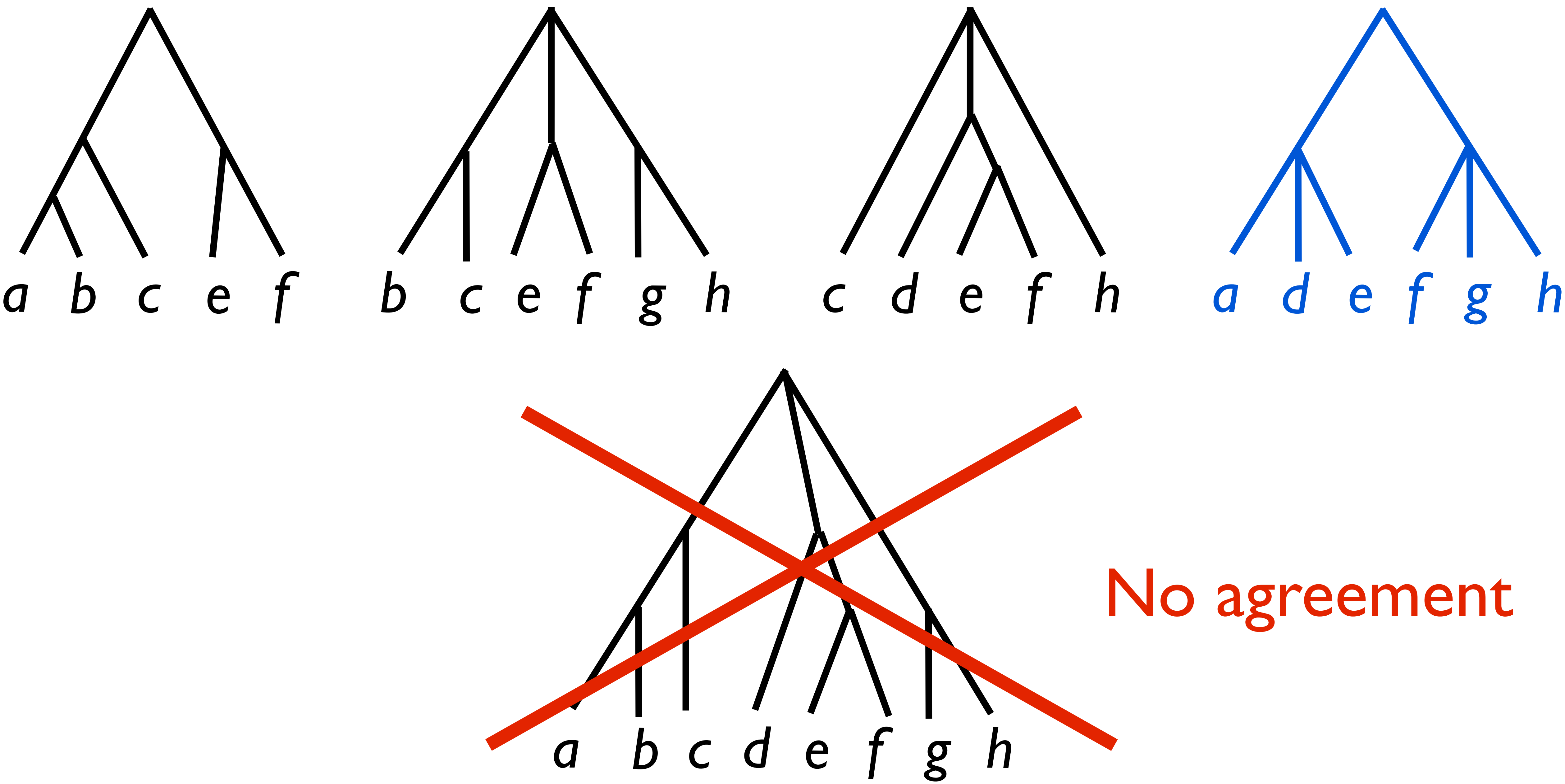
Agreement Supertrees



Agreement Supertrees



Agreement Supertrees



Achieving Agreement

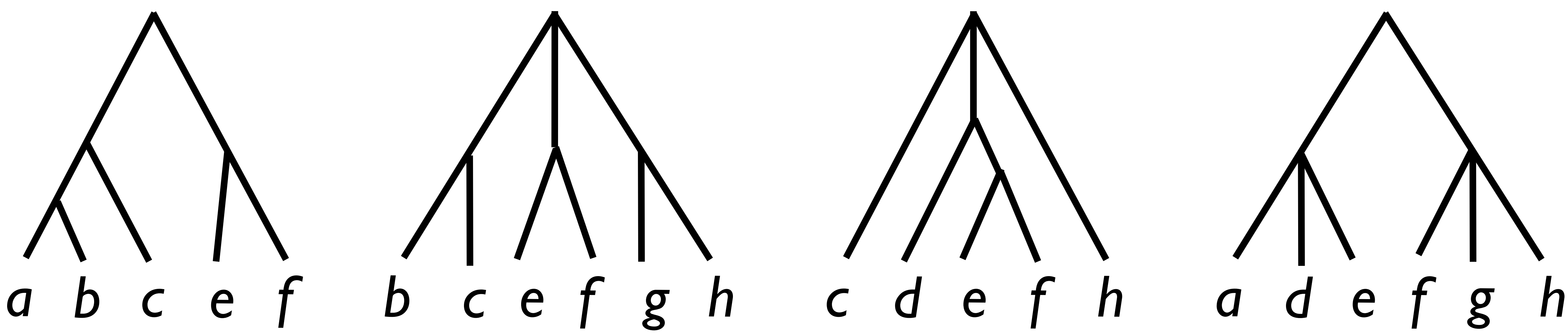
Achieving Agreement

- **Edge contraction (AST-EC):** Can we contract q internal edges from $P = (T_1, T_2, \ldots, T_k)$ so that P has an agreement supertree?

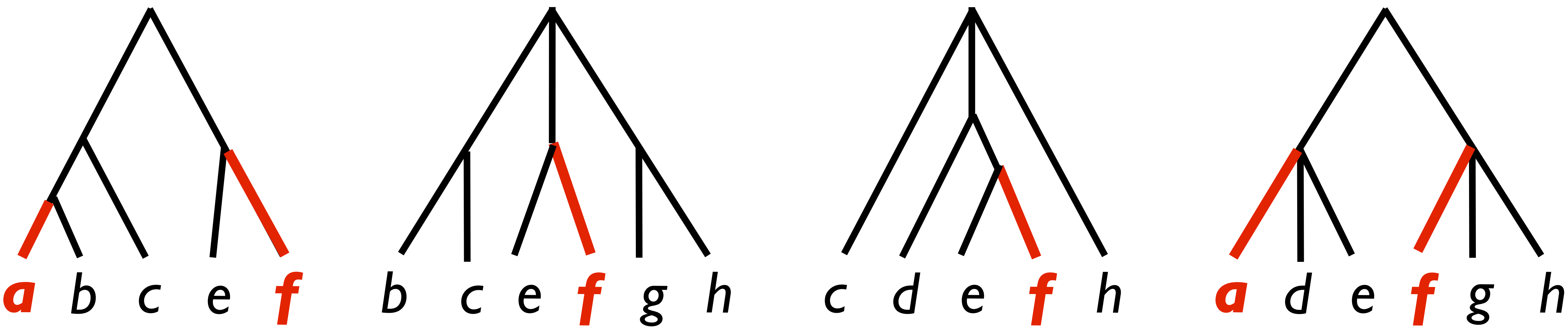
Achieving Agreement

- **Edge contraction (AST-EC):** Can we contract q internal edges from $P = (T_1, T_2, \ldots, T_k)$ so that P has an agreement supertree?
- **Taxon removal (AST-TR):** Can we remove q taxa from $P = (T_1, T_2, \ldots, T_k)$ so that P has an agreement supertree?

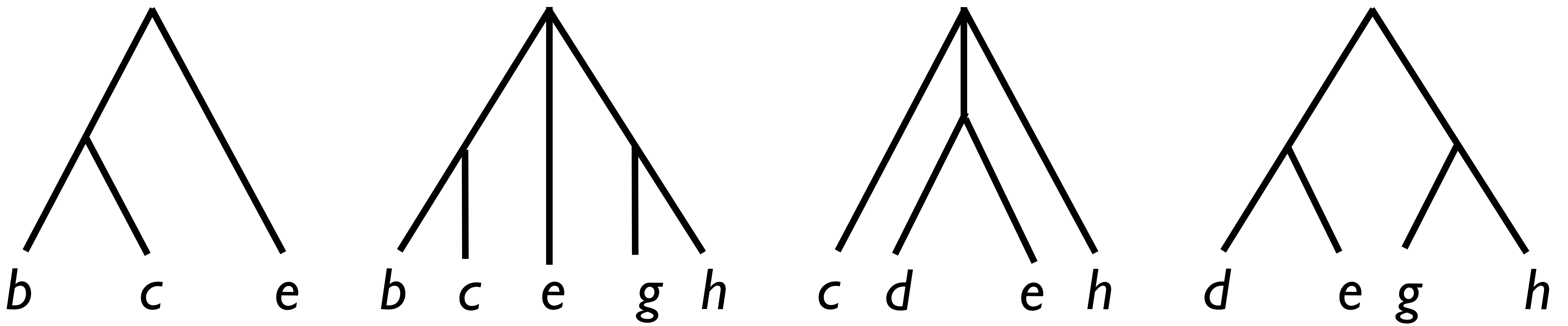
Taxon Removal



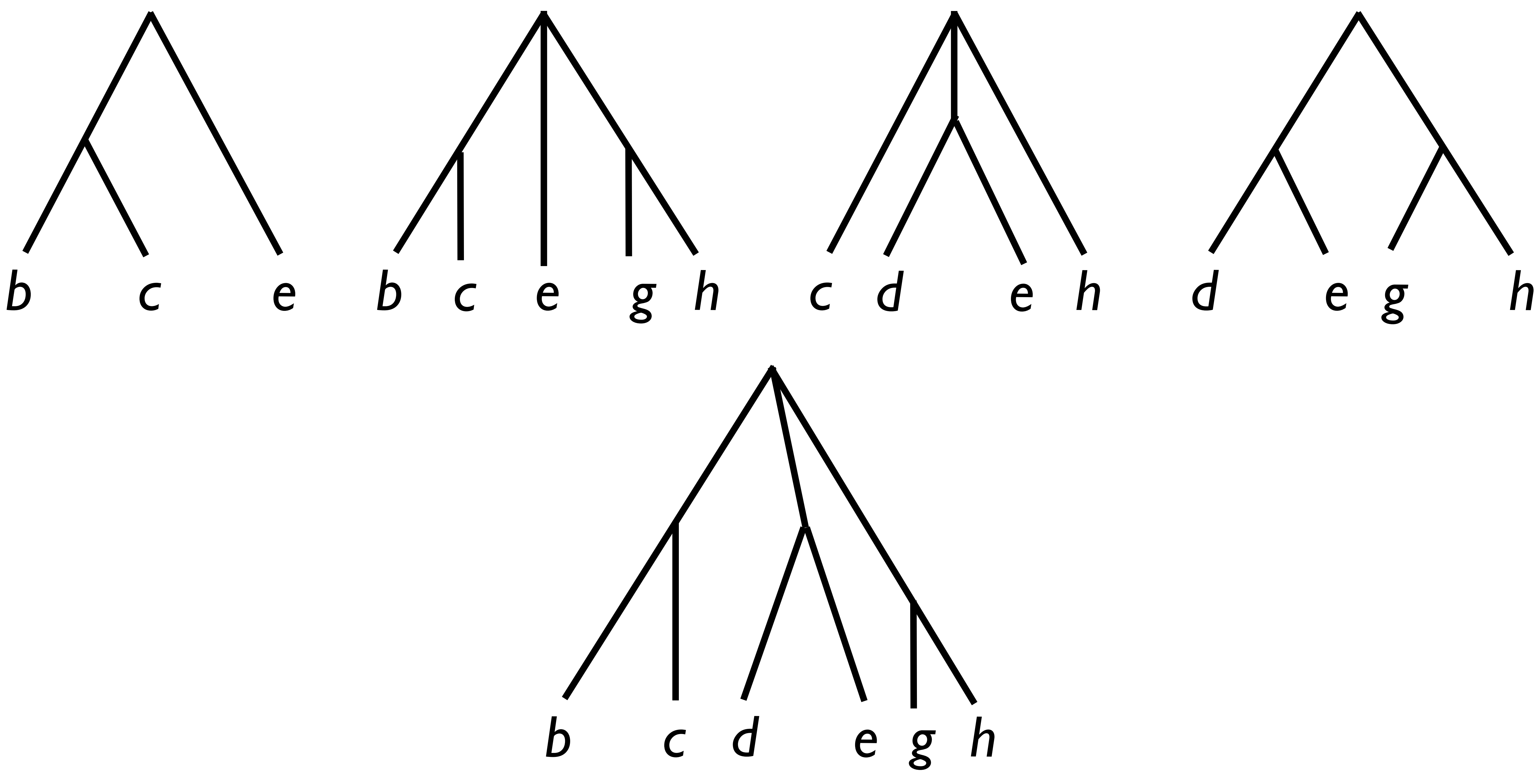
Taxon Removal



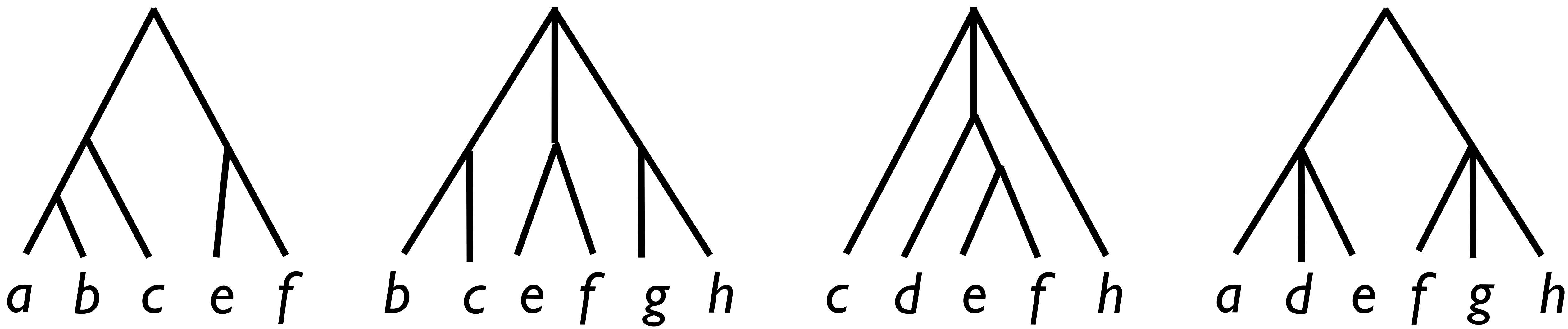
Taxon Removal



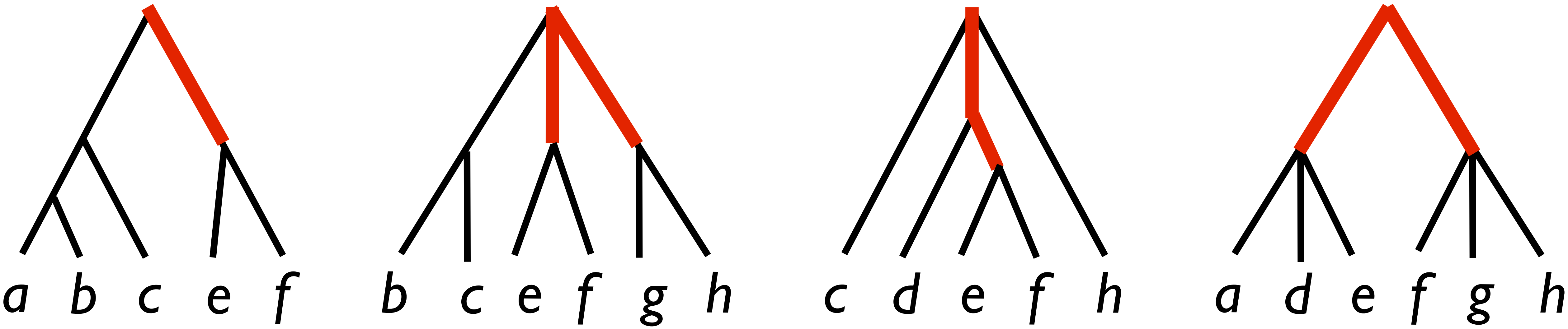
Taxon Removal



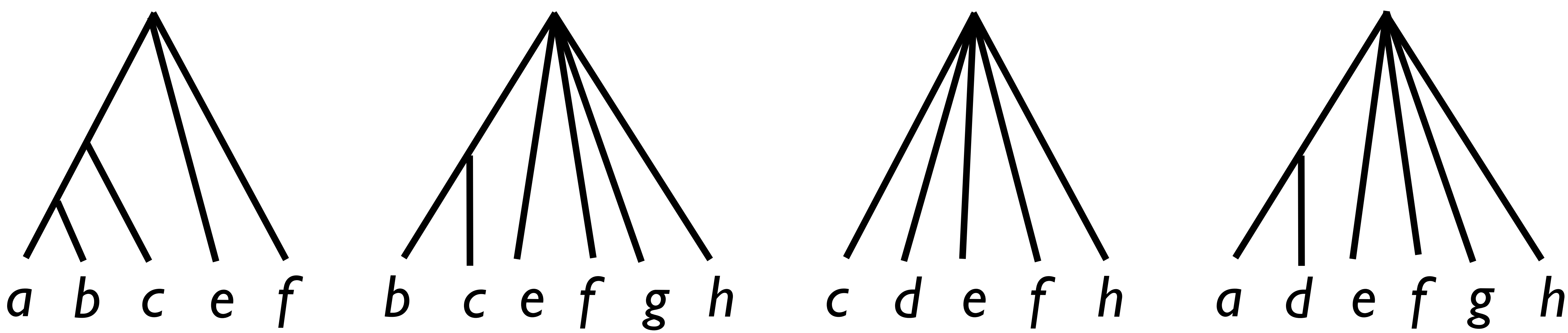
Edge Contraction



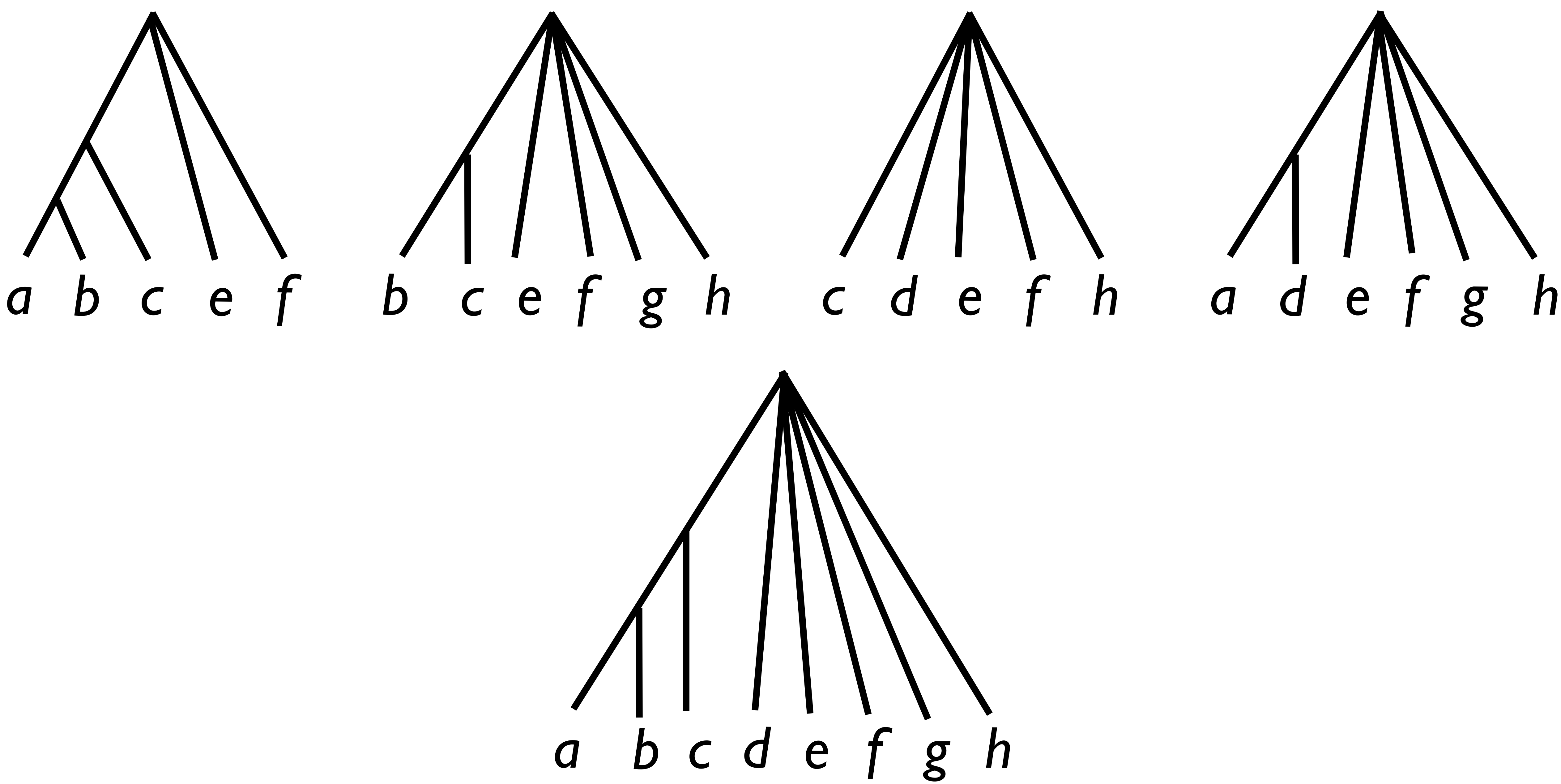
Edge Contraction



Edge Contraction



Edge Contraction



Our Results

Our Results

- AST-EC is NP-hard.

Our Results

- AST-EC is NP-hard.
- A characterization of agreement.

Our Results

- AST-EC is NP-hard.
- A characterization of agreement.
- A $O(kn^2)$ algorithm for testing agreement.

Our Results

- AST-EC is NP-hard.
- A characterization of agreement.
- A $O(kn^2)$ algorithm for testing agreement.
- If answer is "no", the algorithm returns an obstruction, a subset of internal nodes encapsulating the taxa on which the trees disagree.

Our Results

- AST-EC is NP-hard.
- A characterization of agreement.
- A $O(kn^2)$ algorithm for testing agreement.
- If answer is "no", the algorithm returns an obstruction, a subset of internal nodes encapsulating the taxa on which the trees disagree.
- AST-TR and AST-EC with input trees of arbitrary degree are fixed-parameter tractable in k and q.

Our Results

- AST-EC is NP-hard.
- A characterization of agreement.
- A $O(kn^2)$ algorithm for testing agreement.
- If answer is "no", the algorithm returns an obstruction, a subset of internal nodes encapsulating the taxa on which the trees disagree.
- AST-TR and AST-EC with input trees of arbitrary degree are fixed-parameter tractable in k and q.
  - $O((2k)^p \, kn^2)$-time algorithms.

Previous Work: AST-TR

- NP-hard (Jansson et al., 2005, Berry & Nicolas 2007).
- Fixed-parameter tractable in k and q for binary input trees (Guillemot & Berry 2010).
- Fixed-parameter intractable when parameterized by only k or q (Berry & Nicolas 2007)
- Maximum Agreement Supertree: Find largest subset of taxa on which trees agree (Berry & Nicolas 2007, Jansson et al. 2005, Kao 2007).
    - Dual of optimization version of AST-TR.
    - Exact algorithms:
        - Binary: $O(6^k n^k)$ (Guillemot & Berry 2010, Hoang & Sung 2011).
        - Maximum degree d: $O((kd)^{kd+3} 2^k n^k)$ (Hoang & Sung 2011).

Complexity of AST-EC

AST-EC is NP-complete

Reduction from Multicut:

Multicut

Input: $G = (V, E)$, $R \subseteq V \times V$, and integer q.

Question: Does there exist $S \subseteq E$, with $|S| \leq q$ s.t. for every $uv \in R$, u and v are in different components of $G \setminus S$?

Reduction from Multicut

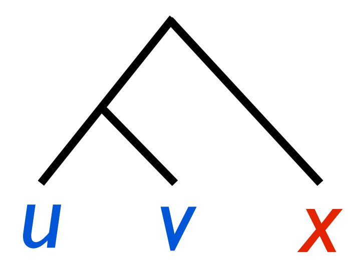
Reduction from Multicut

Given $G = (V, E)$, $R \subseteq V \times V$, and q, build an instance (P, q) of AST-EC, with two kinds of trees:

Reduction from Multicut

Given $G = (V, E)$, $R \subseteq V \times V$, and q, build an instance (P, q) of AST-EC, with two kinds of trees:

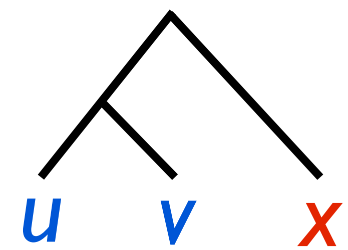
(i) For every $uv \in E$:



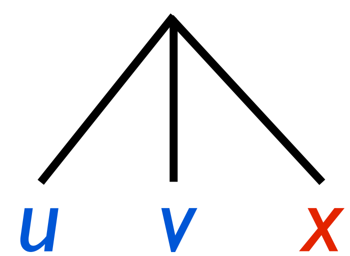
Reduction from Multicut

Given $G = (V, E)$, $R \subseteq V \times V$, and q, build an instance (P, q) of AST-EC, with two kinds of trees:

(i) For every $uv \in E$:



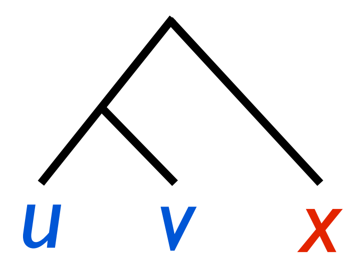
(ii) For every $uv \in R$:



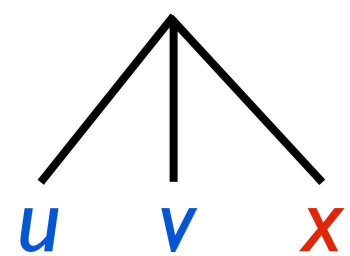
Reduction from Multicut

Given $G = (V, E)$, $R \subseteq V \times V$, and q, build an instance (P, q) of AST-EC, with two kinds of trees:

(i) For every $uv \in E$:

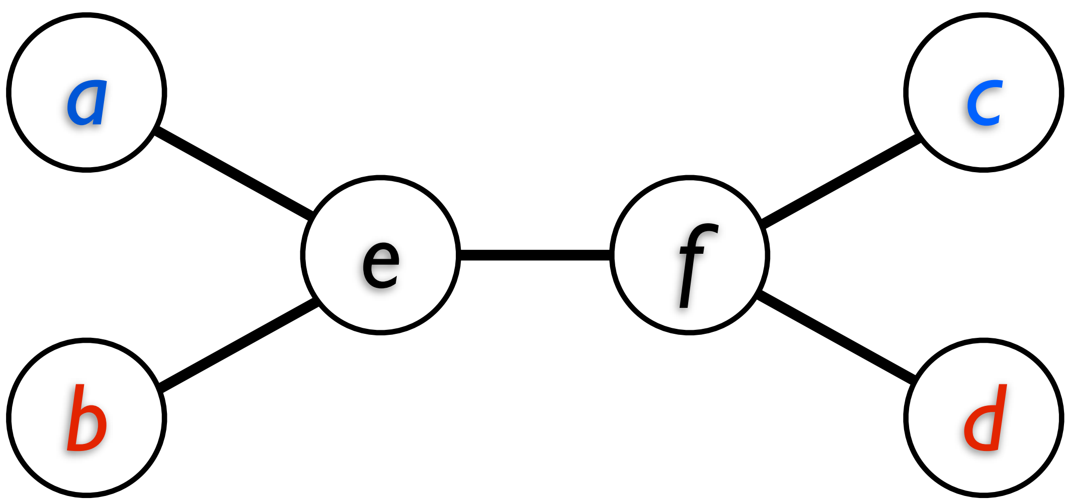


(ii) For every $uv \in R$:



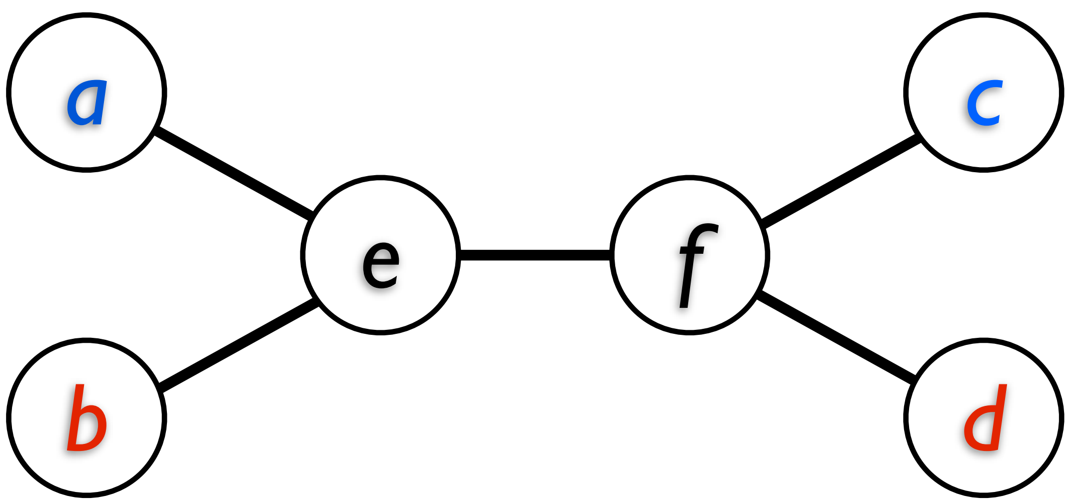
Claim. G has a cut of size $\leq q$, iff (P, q) has a "yes" answer

Reduction from Multicut



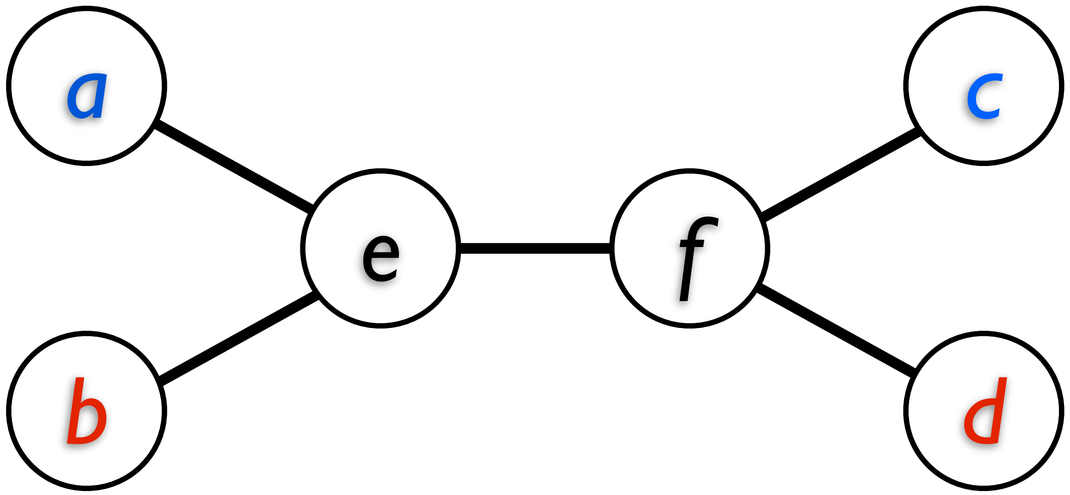
$R = \{ac, bd\}$

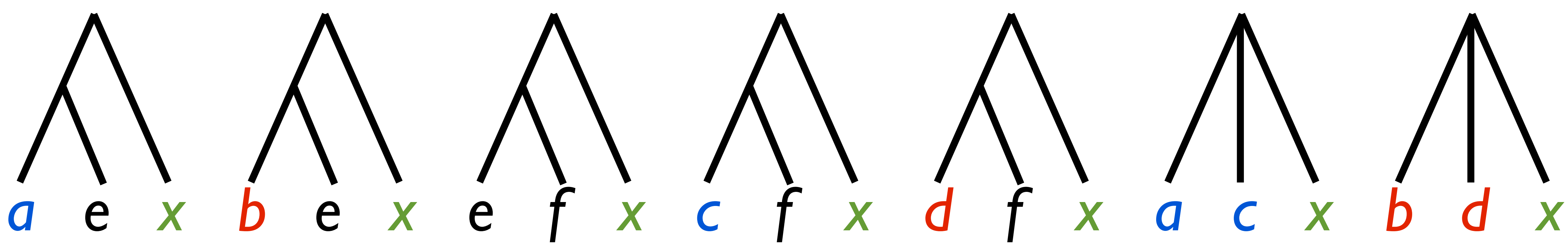
Reduction from Multicut



Edges

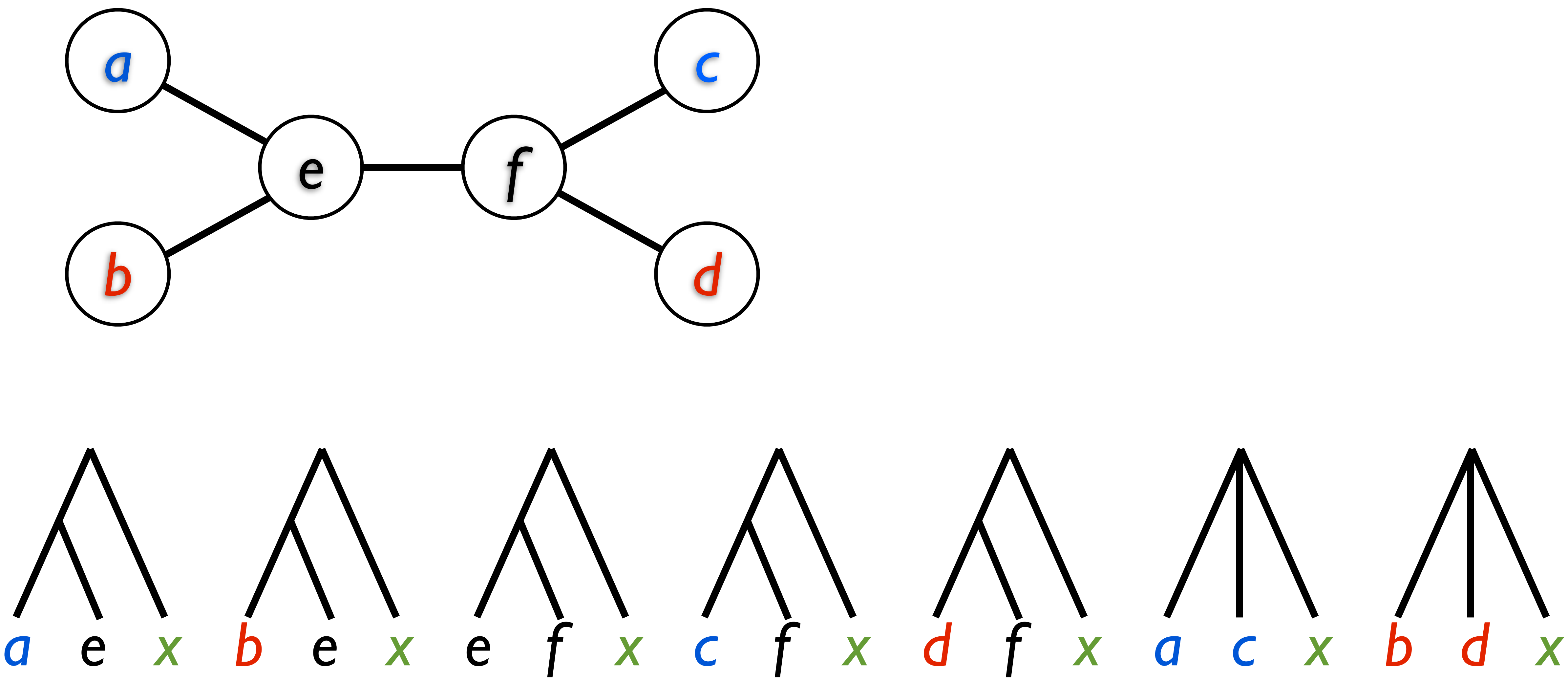
Reduction from Multicut



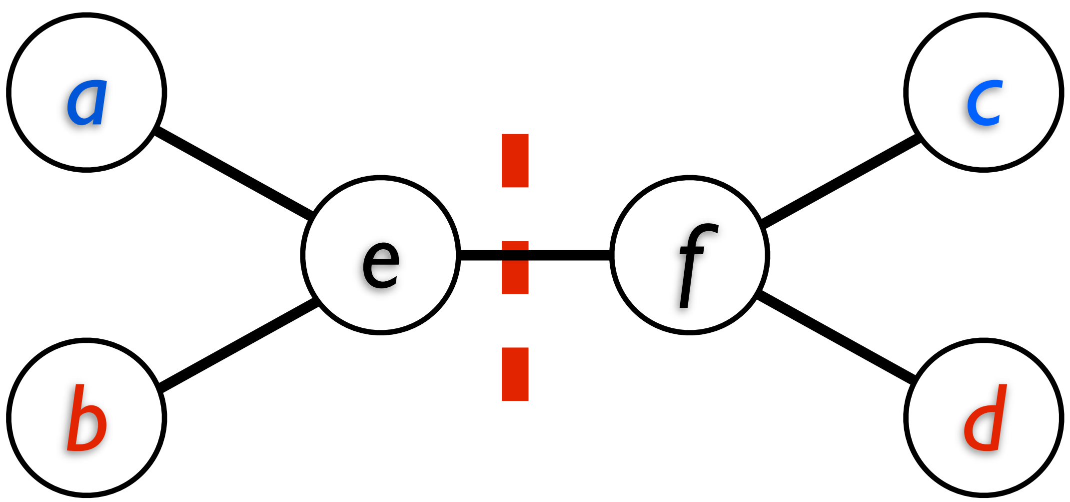


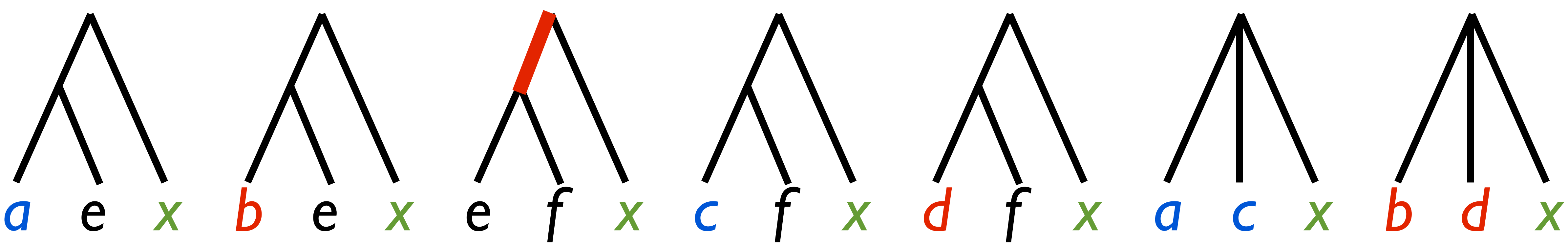
Requests

Reduction from Multicut

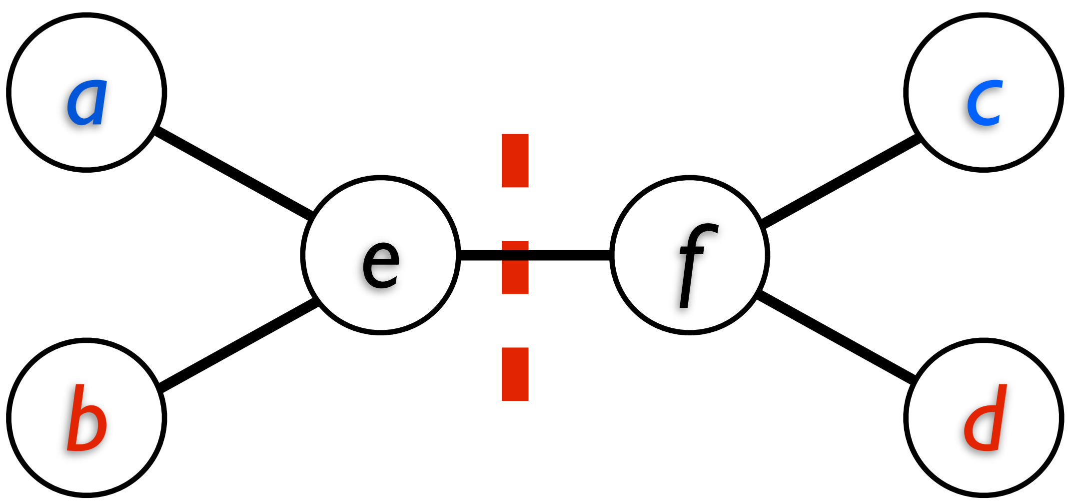


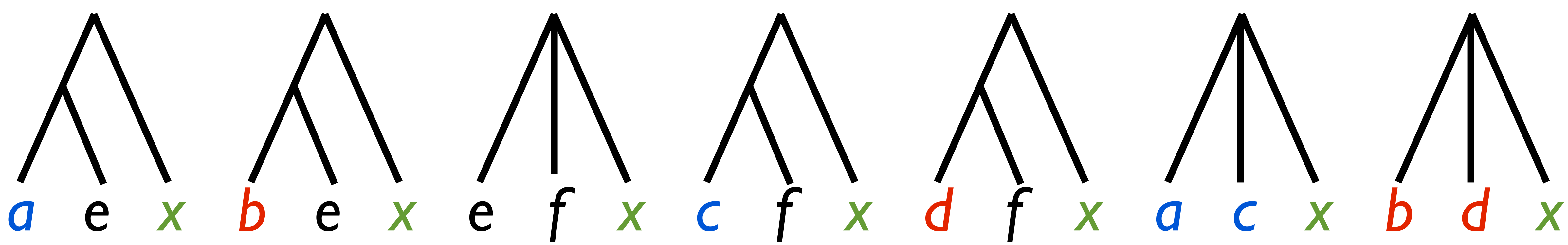
Reduction from Multicut



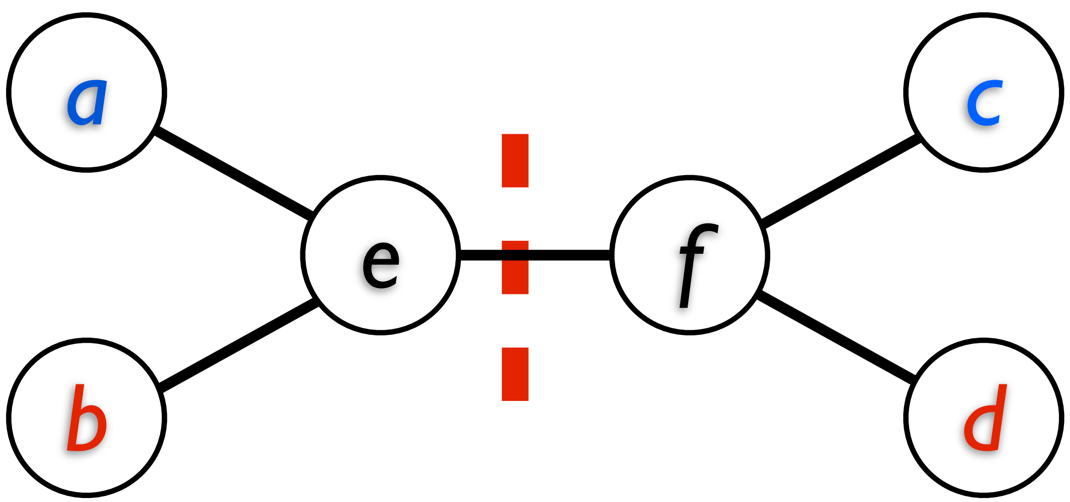


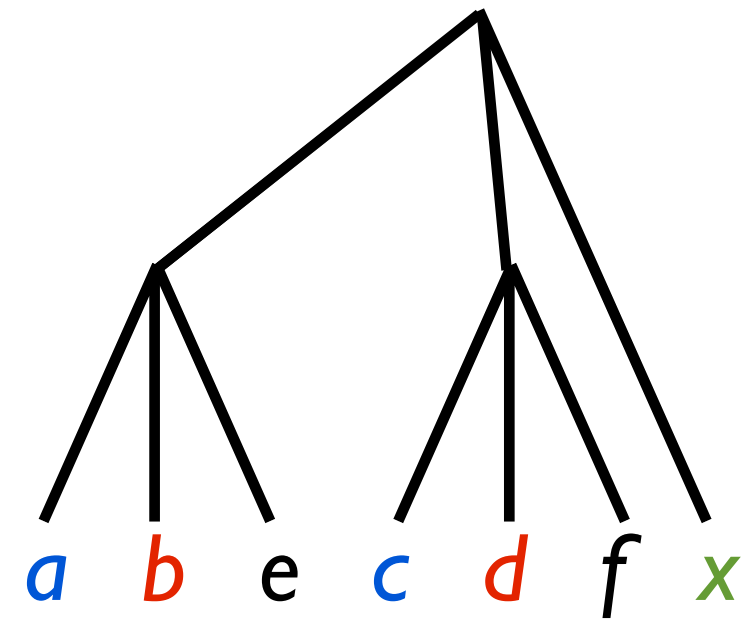
Reduction from Multicut

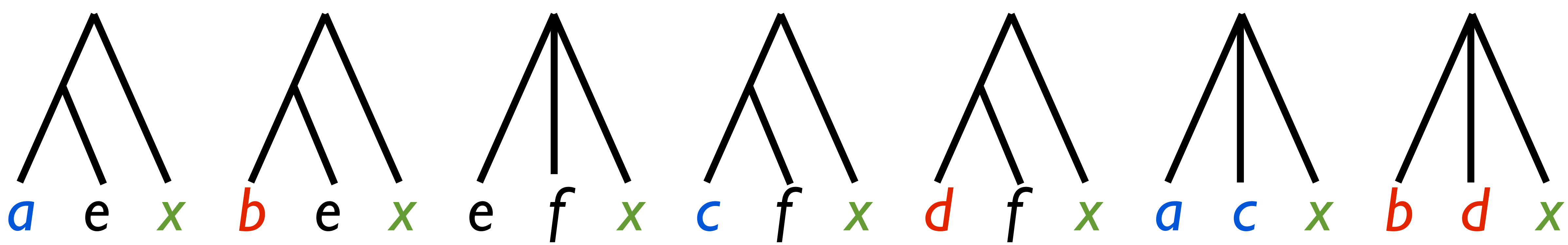




Reduction from Multicut







Editing Problems

Editing and Obstructions

Editing and Obstructions

- Π is some property

Editing and Obstructions

- Π is some property
  - E.g., agreement.

Editing and Obstructions

- Π is some property
  - E.g., agreement.
- **Decision problem:** Does P satisfy Π?

Editing and Obstructions

- Π is some property
  - E.g., agreement.
- **Decision problem:** Does P satisfy Π?
- Typically, Π is too restrictive to be satisfied by most instances.

Editing and Obstructions

- Π is some property
  - E.g., agreement.
- **Decision problem:** Does P satisfy Π?
- Typically, Π is too restrictive to be satisfied by most instances.
- **Approach:** **Edit** P to make it satisfy Π.

Editing and Obstructions

- Π is some property
  - E.g., agreement.
- **Decision problem:** Does P satisfy Π?
- Typically, Π is too restrictive to be satisfied by most instances.
- **Approach:** **Edit** P to make it satisfy Π.
  - E.g., contract edges, delete taxa.

Editing and Obstructions

Editing and Obstructions

- **Editing problem:** Is there a small (size $\leq q$) piece Y of P that can be "deleted" so that $P \backslash Y$ satisfies Π?

Editing and Obstructions

- **Editing problem:** Is there a small (size $\leq q$) piece Y of P that can be "deleted" so that $P \setminus Y$ satisfies Π?
- **Small obstruction property:** If P does not satisfy Π, there is a small subset of X of P that also fails to satisfy Π.

Editing and Obstructions

- **Editing problem:** Is there a small (size $\leq q$) piece Y of P that can be "deleted" so that $P \setminus Y$ satisfies Π?
- **Small obstruction property:** If P does not satisfy Π, there is a small subset of X of P that also fails to satisfy Π.
  - X is an ***obstruction*** for Π.

GenericEdit(P,q):

if P satisfies Π **then return** "yes"

if $q == 0$ **then return** "no"

find an obstruction X

if $\exists\ e \in X$ s.t. *GenericEdit*$(P \backslash e, q-1)$ == "yes" **then**
    return "yes"

return "no"

GenericEdit(P,q):

if P satisfies Π **then return** "yes"

if $q == 0$ **then return** "no"

find an obstruction X

if ∃ e ∈ X s.t. *GenericEdit(P\e,q−1)* == "yes" **then**

return "yes"

return "no"

Fact. If Π can be decided in $n^{O(1)}k^{O(1)}$ time & $|X| \leq ck$, then *GenericEdit* runs in time $(ck)^q \cdot n^{O(1)}k^{O(1)}$.

Strategy

Strategy

1. Prove that AST-EC and AST-TR satisfy the small obstruction property:
   If $P = (T_1, T_2, \ldots, T_k)$ has no agreement supertree, there is a "subset" X of P of size $\leq 2k$, such that P restricted to X has no agreement supertree either.

Strategy

1. Prove that AST-EC and AST-TR satisfy the small obstruction property:
   If $P = (T_1, T_2, \ldots, T_k)$ has no agreement supertree, there is a "subset" X of P of size $\leq 2k$, such that P restricted to X has no agreement supertree either.
2. Give a $O(kn^2)$ time algorithm to find an obstruction.

Strategy

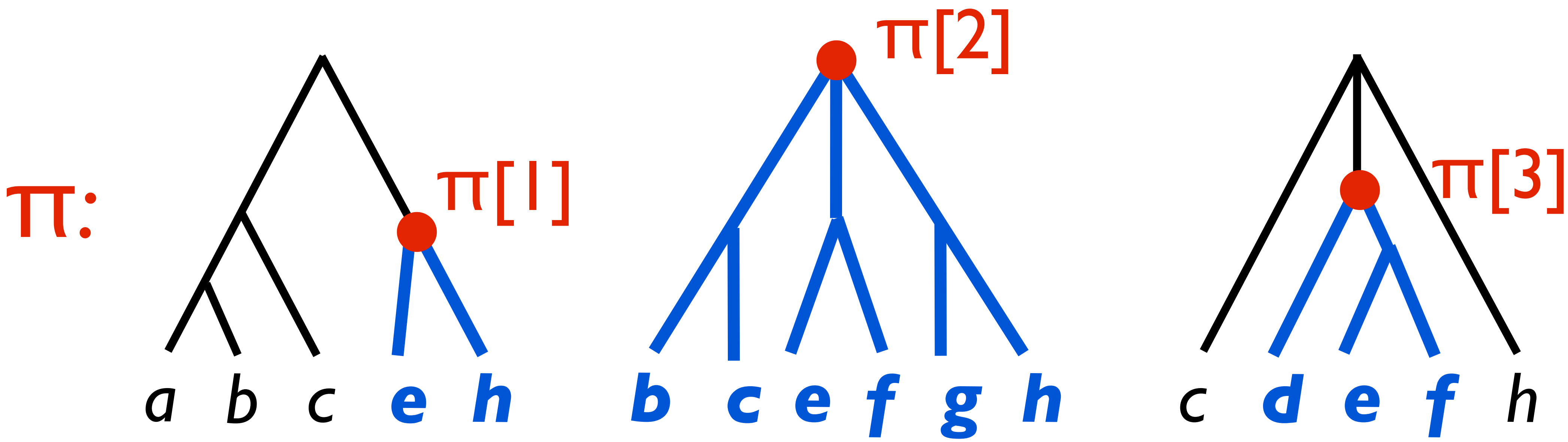
1. Prove that AST-EC and AST-TR satisfy the small obstruction property:
   If $P = (T_1, T_2, \ldots, T_k)$ has no agreement supertree, there is a "subset" X of P of size $\leq 2k$, such that P restricted to X has no agreement supertree either.

2. Give a $O(kn^2)$ time algorithm to find an obstruction.

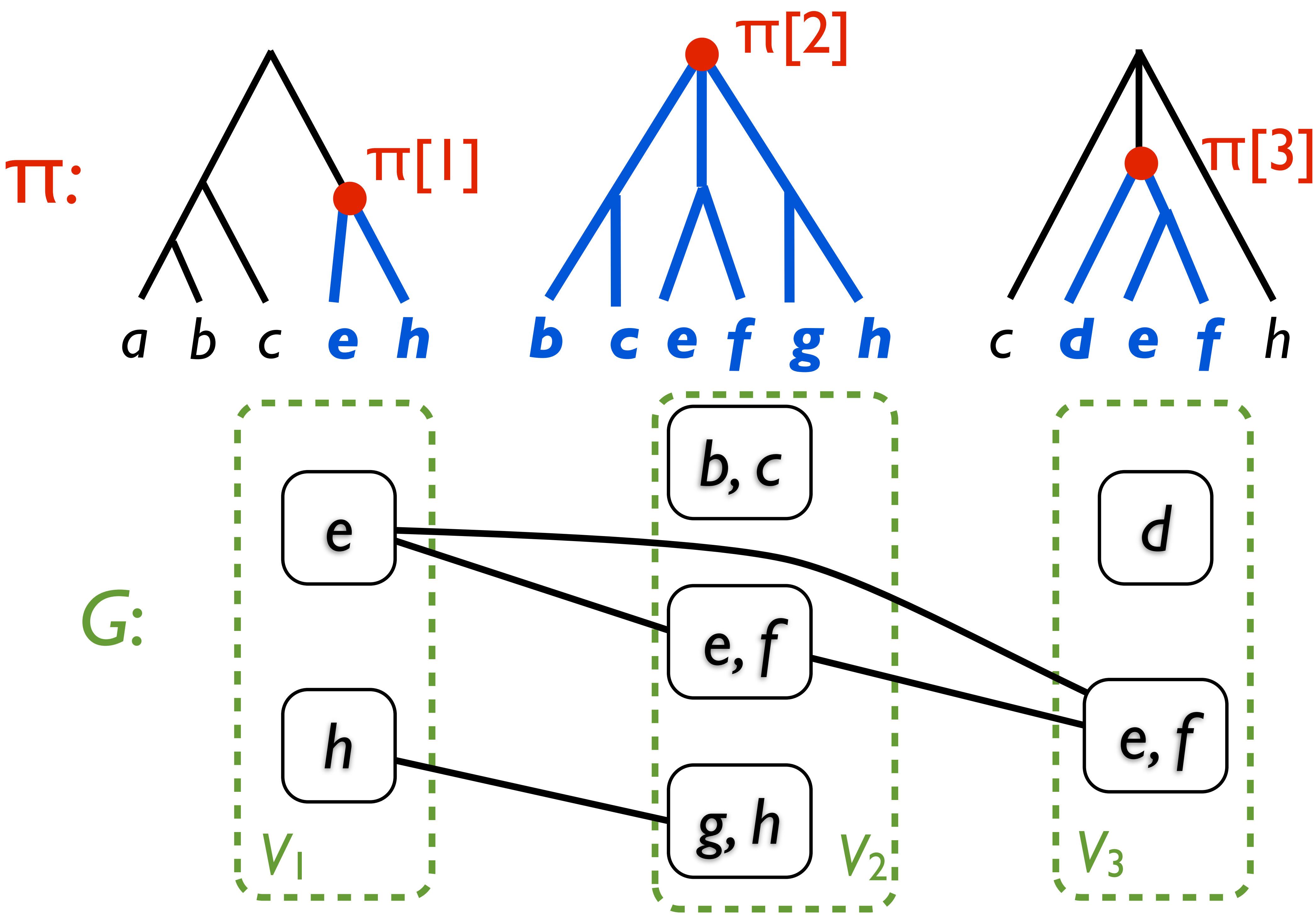
3. Use *GenericEdit* to get $O((2k)^q kn^2)$ algorithm.

Obstructions to Agreement

Position



Position



Positions & Agreement

- **Lemma.** *P* has an agreement supertree iff there is an agreement supertree for every position π.
- Algorithm (sketch):
  - Initially π is the root position.
  - If *G* decomposes nicely, find a successor position for each part of the decomposition and recurse.
  - If *G* does not decompose nicely, π encapsulates the taxa on which the trees disagree.

Nice Sets & Successors

- $Y \subseteq V$ is nice if, for each $i \in [k]$, Y contains either zero, one, or all the elements of V_i.
- The successor of $\pi = (\pi[1], \ldots, \pi[k])$ w.r.t. a nice set Y is the vector $\pi_Y = (\pi_Y[1], \ldots, \pi_Y[k])$ where
  - $\pi_Y[i] = \perp$ if $V_i \cap Y = \emptyset$
  - $\pi_Y[i] = p$ if $V_i \cap Y = \{p\}$
  - $\pi_Y[i] = \pi[i]$ if $|V_i \cap Y| \geq 2$

Nice Partitions

- Partition R of V is nice if every set of R is nice, and, for every $\{A, B\} \subseteq R$, A and B are disconnected.
- **Lemma.** The following are equivalent:
  - $\exists$ an agreement supertree for π.
  - $\exists$ a nice partition R of G where $|R| \geq 2$, and, for every $Y \in R$, π_Y has an agreement supertree.
- Partition R is finer than partition Q, denoted $R \leq Q$, iff, for every $C \in R$, $\exists\, D \in Q$ s. t. $C \subseteq D$.
- **Lemma.** The ordering of all nice partitions of G under $\leq$ has a unique minimal element.

Obstructions

- π is an obstructing position if there exists $X \subseteq V$ s.t.
  - $|X \cap V_i| = 2$ for each i, and
  - there is a set $F \subseteq E$ s. t.:
    - $|F| \leq 2|K| - 1$;
    - for each $v \in X$, $\exists\ e \in F$ s. t. $v \in e$; and
    - the minimum nice partition of the subgraph of G induced by F is a singleton.
- X is an obstruction set for π.

Algorithms

- An obstruction set X can be found in $O(kn^2)$ time.
- For AST-EC, can show that at least one edge $\{v, parent(v)\}$ with $v \in X$ must be contracted to achieve agreement.
  - At most $2k$ edges must be considered.
- For AST-TR, can use X to obtain a conflict set of at most $2k - 1$ taxa.
- Solve AST-EC and AST-TR using *GenericEdit*.

Summary

- AST-EC and AST-TR are fixed-parameter tractable in k and p.
  - $O((2k)^p\ kn^2)$-time algorithms.
- Bounds of $2k$ and $2k - 1$ on the respective sizes of the obstruction sets for AST-EC and AST-TR
- Both bounds are tight.

Open Problems

Open Problems

- NP-hardness proof for AST-EC is by reduction from Multicut.

Open Problems

- NP-hardness proof for AST-EC is by reduction from Multicut.
  - Multicut is fixed-parameter tractable (Marx & Razgon 2011),

Open Problems

- NP-hardness proof for AST-EC is by reduction from Multicut.
  - Multicut is fixed-parameter tractable (Marx & Razgon 2011),
  - Is AST-EC fixed-parameter tractable in p only?

Open Problems

- NP-hardness proof for AST-EC is by reduction from Multicut.
  - Multicut is fixed-parameter tractable (Marx & Razgon 2011),
  - Is AST-EC fixed-parameter tractable in p only?
- Are AST-EC and/or AST-TR fixed-parameter tractable for unrooted trees?

Funding

- CCF-106029
- DEB-0829674

Thank you!