Deep kernelization for the Tree Bisection and Reconnnect (TBR) distance in phylogenetics

Steven Kelk

Department of Advanced Computing Sciences ← *new name since last Friday…*
Maastricht University
The Netherlands

Email: steven.kelk@maastrichtuniversity.nl

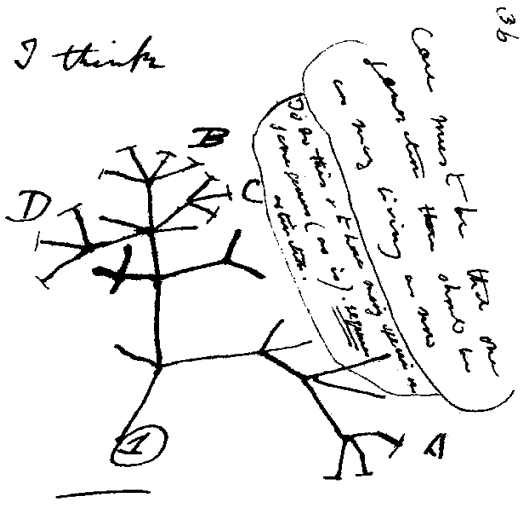
Joint work with Simone Linz (Auckland) and Ruben Meuwese (Maastricht)

Utrecht University, 20th June 2022.

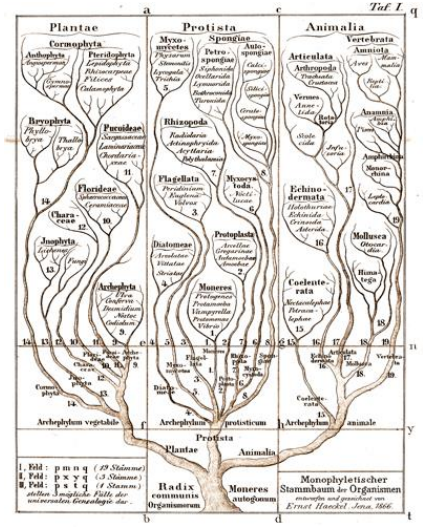
Maastricht University

Phylogenetics

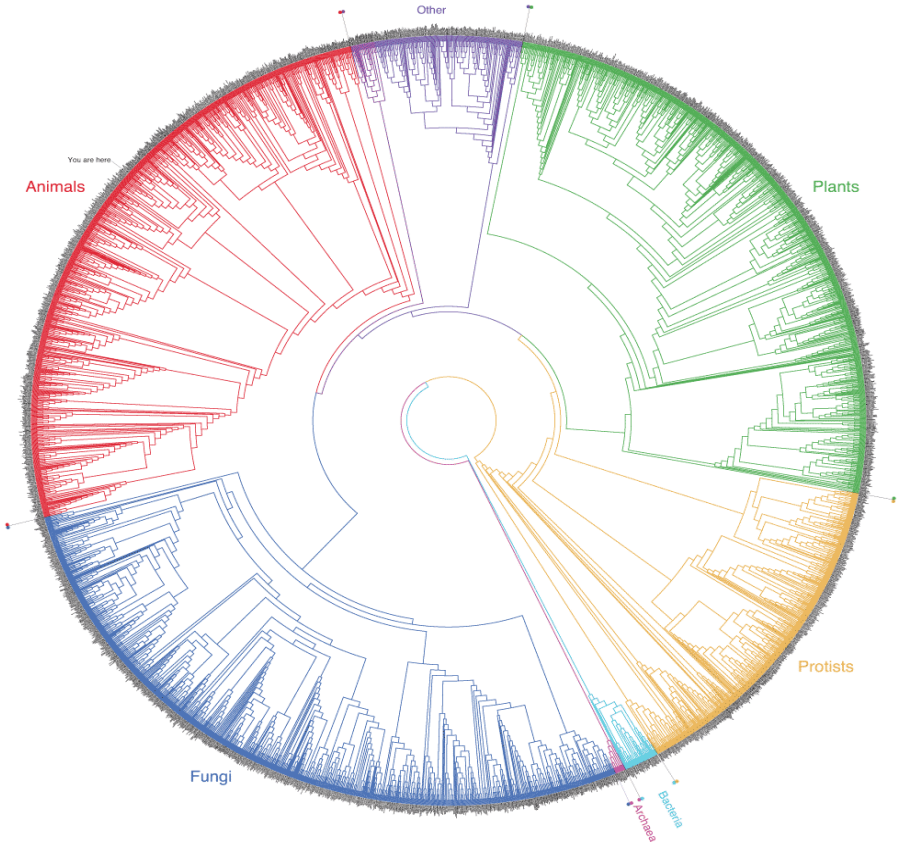
The reconstruction and analysis of evolutionary trees and networks based on molecular sequence data or morphological characters.



Charles Darwin (1837)

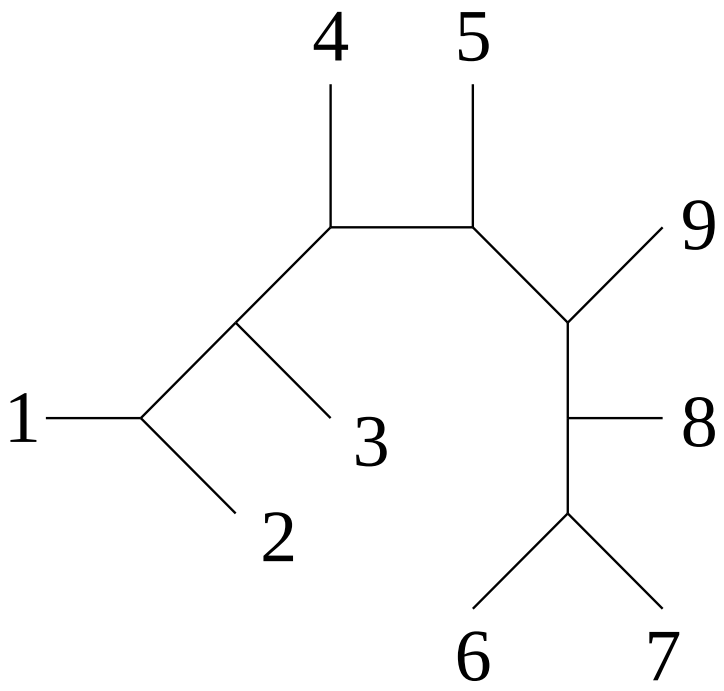


Ernst Haeckel (1866)



Hillis, et al. (2003)

Phylogenetic trees



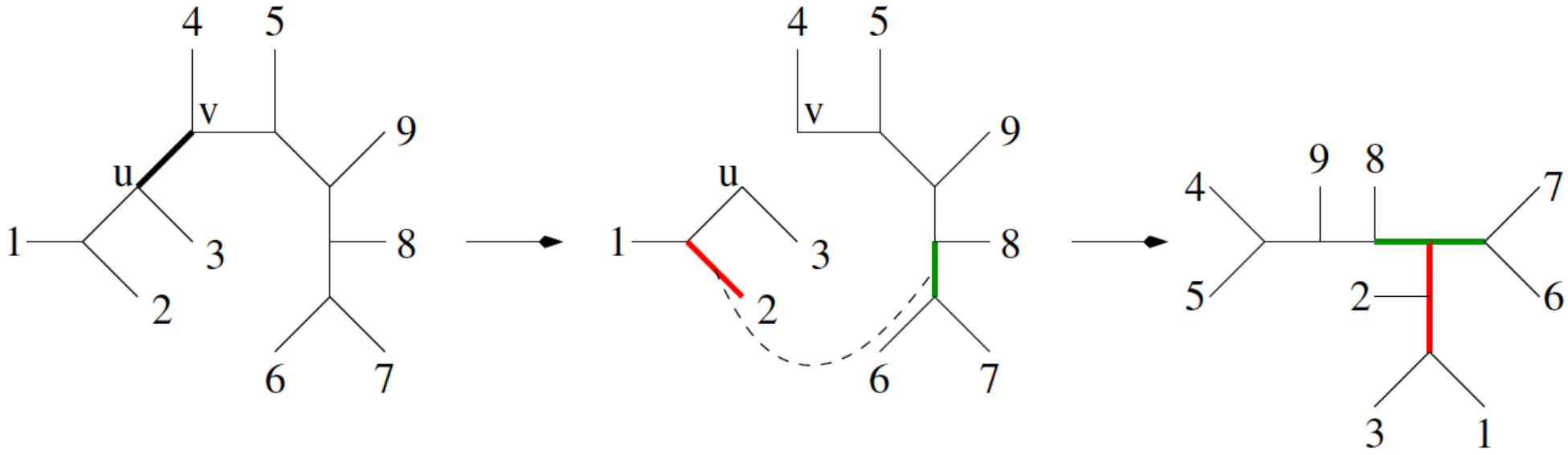
An (*unrooted) phylogenetic tree on X* is a connected acyclic graph whose internal vertices have degree three and whose leaf set is *X*.

Distances between phylogenetic trees

We wish to compare two trees, i.e. to quantify the dissimilarities between them.

Distances between trees provide a lower bound on the number of non-tree-like events, such as hybridization, which can cause the topologies of the trees to differ.

Tree bisection and reconnection (TBR)



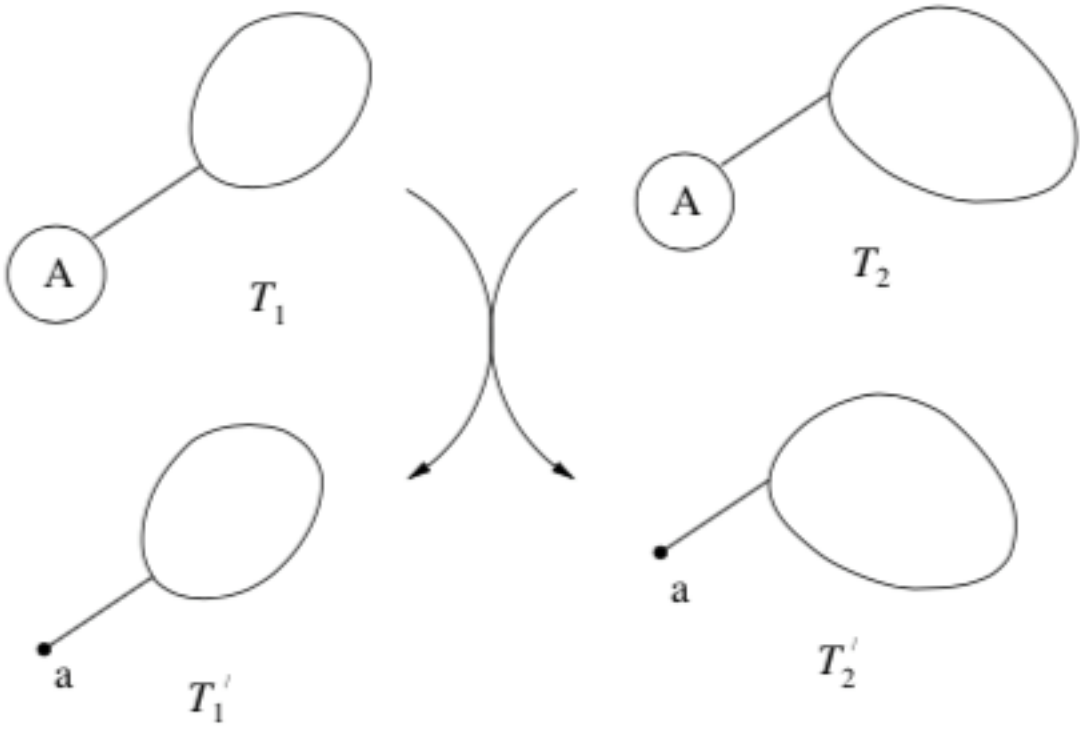
Let $d_{TBR}(T,T')$ denote the minimum number of TBR operations required to transform T into T'. Then, $d_{TBR}(T,T')$ induces a metric on the space of all unrooted phylogenetic trees with n leaves.

(Robinson, 1971; Allen and Steel, 2001).

Computing $d_{TBR}(T,T')$ is NP-hard and fixed-parameter tractable, when parameterized by $k=d_{TBR}$.

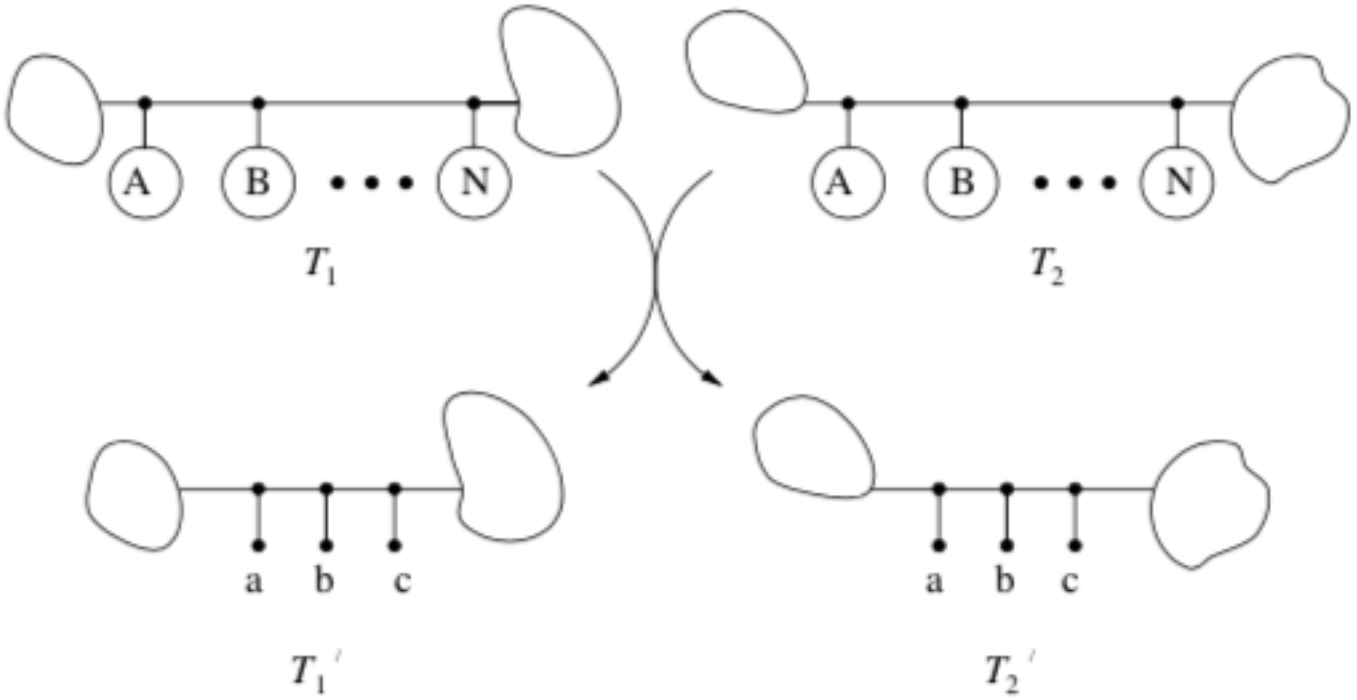
(Hein et al., 1996; Allen and Steel, 2001).

Subtree reduction



Allen and Steel, 2001

Chain reduction



Allen and Steel, 2001

Theorem. (Allen and Steel, 2001).

[Reductions are safe] Let S and S' be two trees obtained from T and T' by applying a single subtree or chain reduction. Then

$$d_{\mathrm{TBR}}(T,T') = d_{\mathrm{TBR}}(S,S').$$

[Linear kernel] Let S and S' be two trees obtained from T and T' by repeated applications of the subtree and chain reduction until no further reduction is possible. Then

$$|X'| \leq 28 d_{\mathrm{TBR}}(T,T'),$$

where X' is the leaf set of S and S'.

Theorem. (Allen and Steel, 2001).

[Reductions are safe] Let S and S' be two trees obtained from T and T' by applying a single subtree or chain reduction. Then

$$d_{\mathrm{TBR}}(T,T') = d_{\mathrm{TBR}}(S,S').$$

[Linear kernel] Let S and S' be two trees obtained from T and T' by repeated applications of the subtree and chain reduction until no further reduction is possible. Then

$$|X'| \leq 28 d_{\mathrm{TBR}}(T,T'),$$

where X' is the leaf set of S and S'.

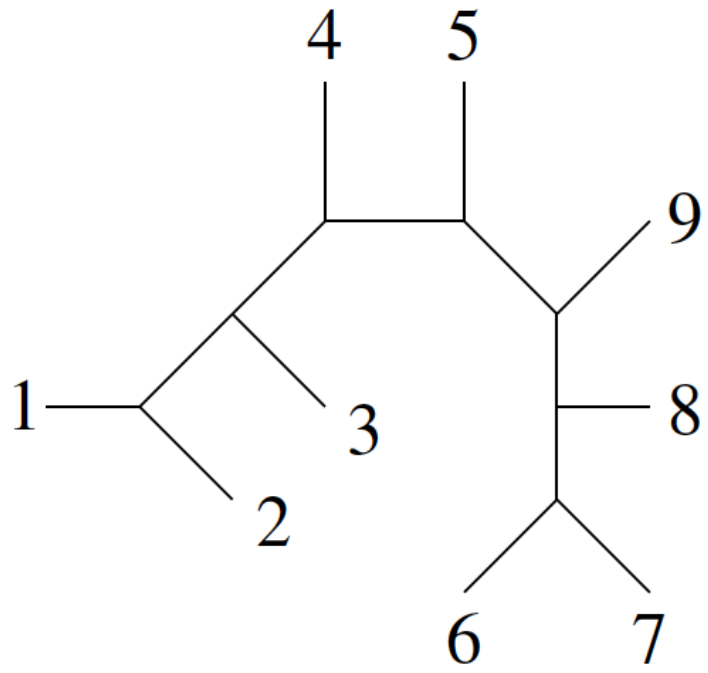
How good is this bound/is it tight/can we do better?

- We reanalysed Allen and Steel's kernel, and show that it is considerably smaller than they claimed: $15d_{TBR} - 9$. Moreover, this is tight. (K. & Linz, SIAM Journal on Discrete Mathematics 2019)

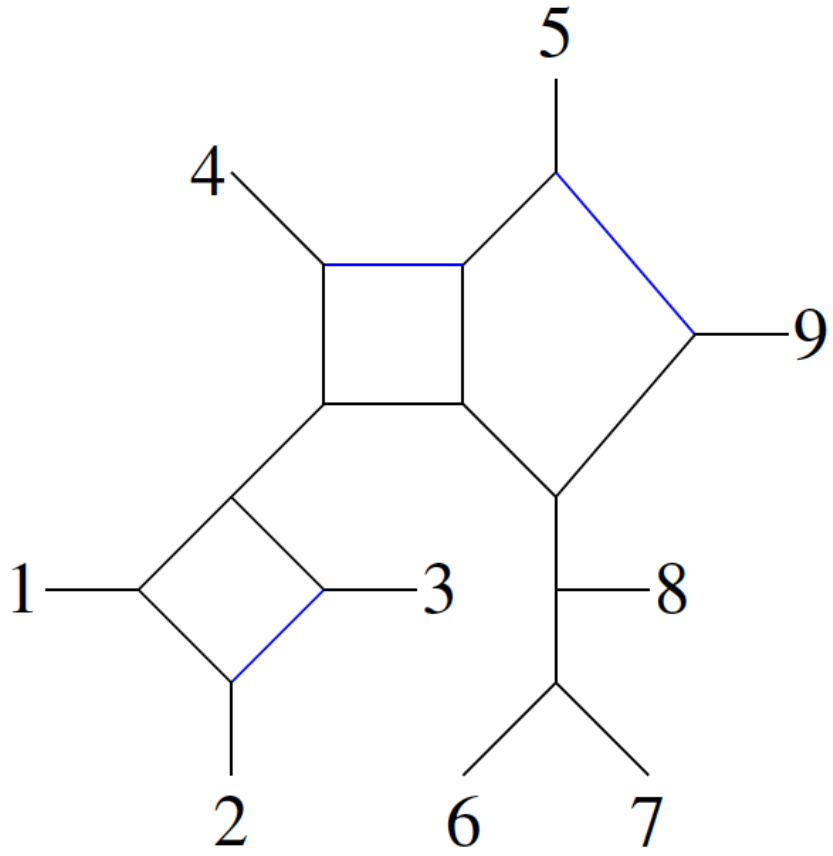
- We devised five new reduction rules which, when combined with Allen and Steel's reduction rules, yield a kernel of size: $11d_{TBR} - 9$. This is also tight. (K. & Linz, Annals of Combinatorics, 2020)

- Today: We introduce a number of 'third generation' reduction rules which reduce the kernel size to $9d_{TBR} - 8$. And, yes, *essentially* tight ☺

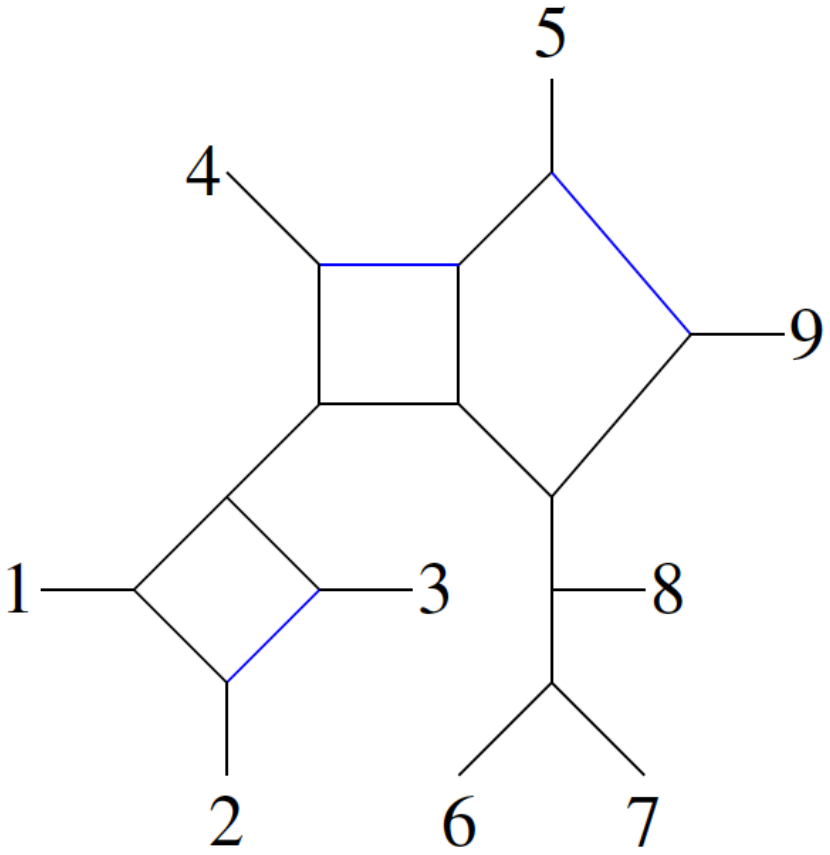
From trees to networks



An *unrooted phylogenetic tree on X* is a connected acyclic graph whose internal vertices have degree three and whose leaf set is *X*.



An *unrooted phylogenetic network N on X* is a simple graph whose internal vertices have degree three and whose leaf set is *X*.



Reticulation number of *N* is

$$r(N) = |E| - (|V|-1).$$

(equal to cyclomatic number).

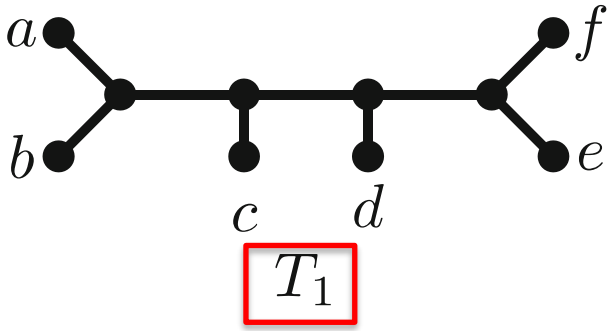
Example. $r(N) = 3$

For two trees *T* and *T'*, define the *hybridization number*

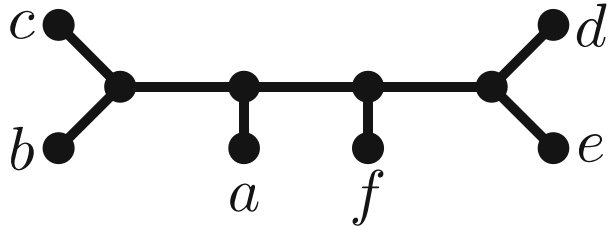
$$h^u(T, T') = \min_N \{r(N)\}$$

Where the minimum is taken over all N that embed *T* and *T'*.

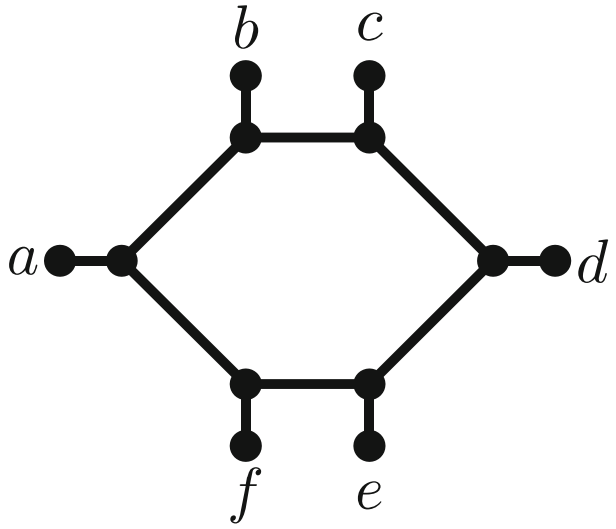
Example.



T_1



T_2



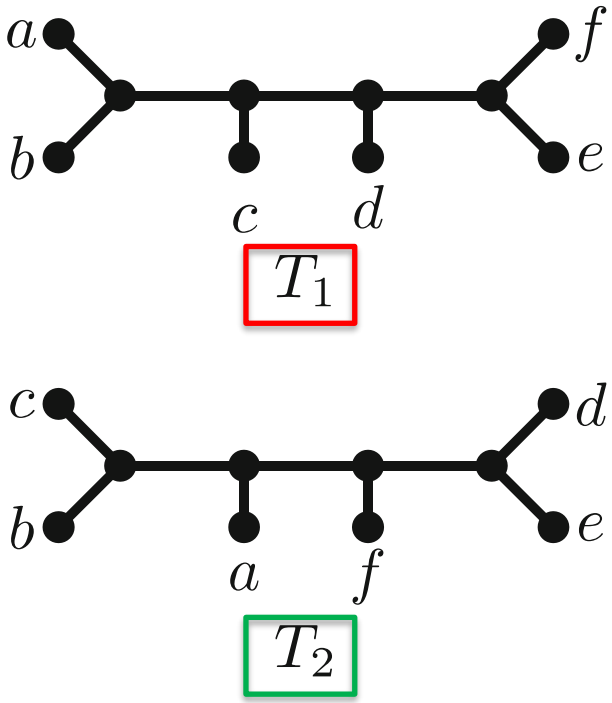
(van Iersel et al., 2018).

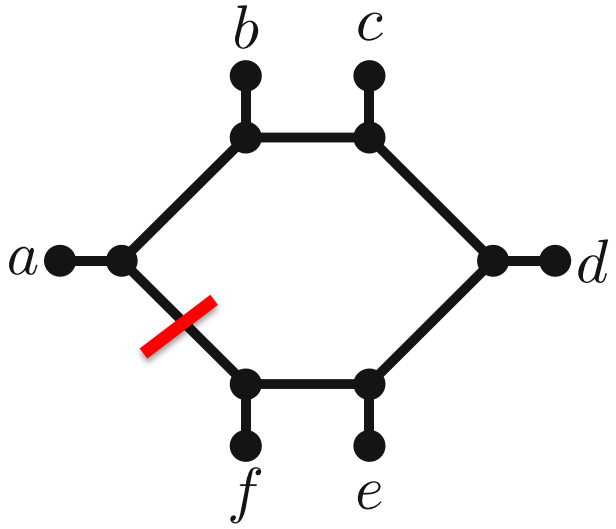
For two trees *T* and *T'*, define the *hybridization number*

$$h^u(T, T') = \min_N \{r(N)\}$$

Where the minimum is taken over all N that embed *T* and *T'*.

Example.





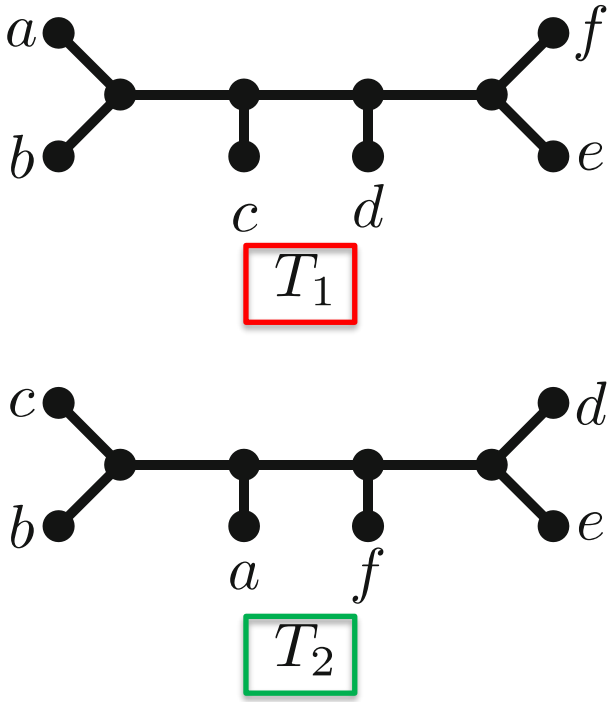
(van Iersel et al., 2018).

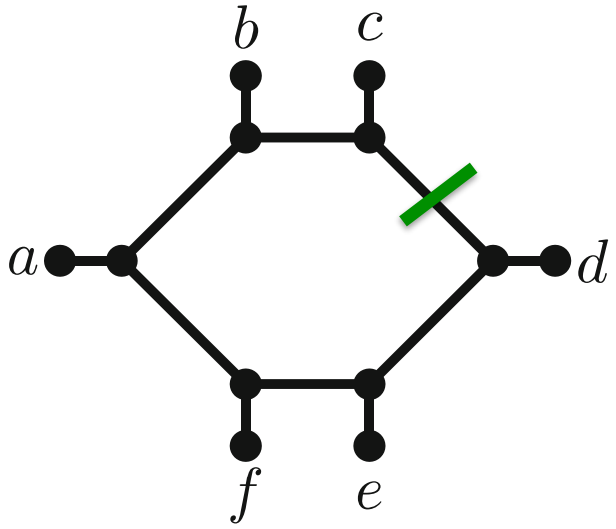
For two trees *T* and *T'*, define the *hybridization number*

$$h^u(T, T') = \min_N \{r(N)\}$$

Where the minimum is taken over all N that embed *T* and *T'*.

Example.





(van Iersel et al., 2018).

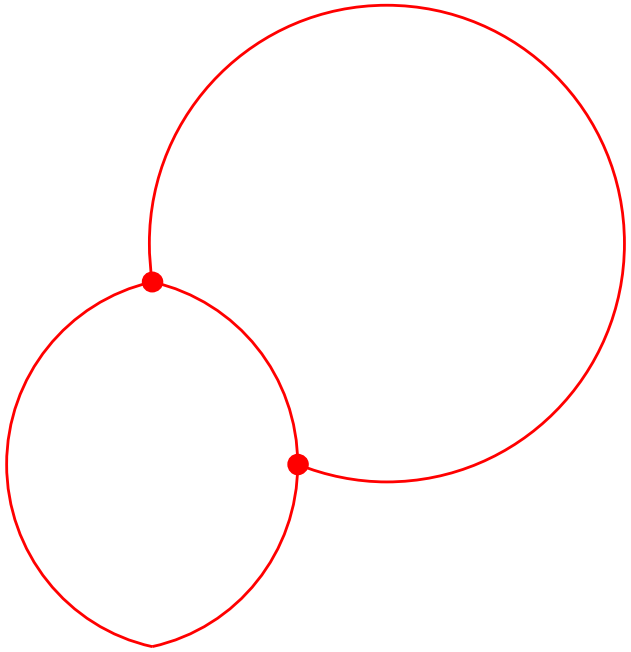
Computing $d_{TBR} \approx$ combining trees into networks

Theorem. (van Iersel et al., 2018)

Let T and T' be two trees. Then

$$d_{\mathrm{TBR}}(T,T') = h^u(T,T')$$

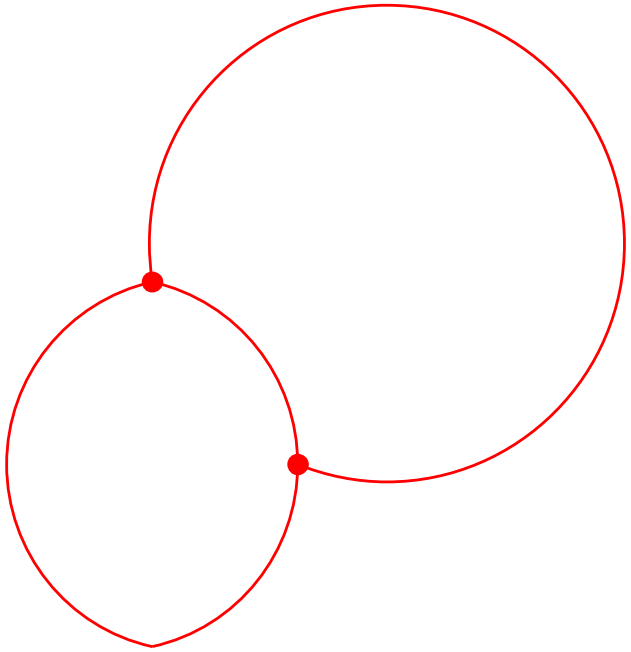
Backbones of phylogenetic networks



2-generator *G* with three edges (*sides*)

For $k \geq 2$, a *k-generator* is a connected cubic multigraph such that $k = |E|-(|V|-1)$.

Backbones of phylogenetic networks

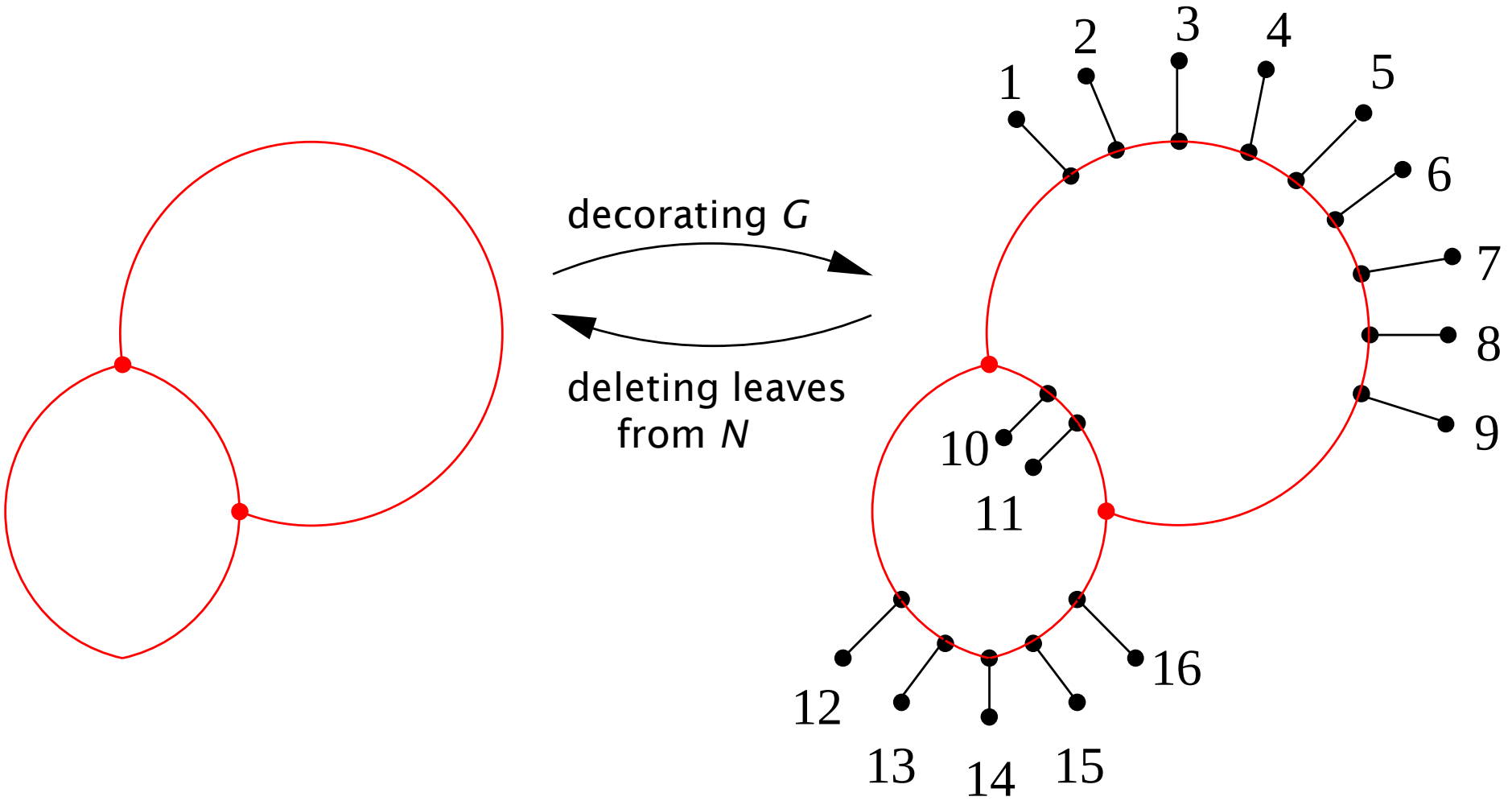


For $k \geq 2$, a *k-generator* is a connected cubic multigraph such that $k = |E|-(|V|-1)$.

2-generator *G* with three edges (*sides*)

More generally: *k*-generator *G* has *3(k-1)* edges (*sides*)

Backbones of phylogenetic networks



2-generator *G* with three edges (*sides*)

network *N* with *r(N)=2* with no pendant subtree

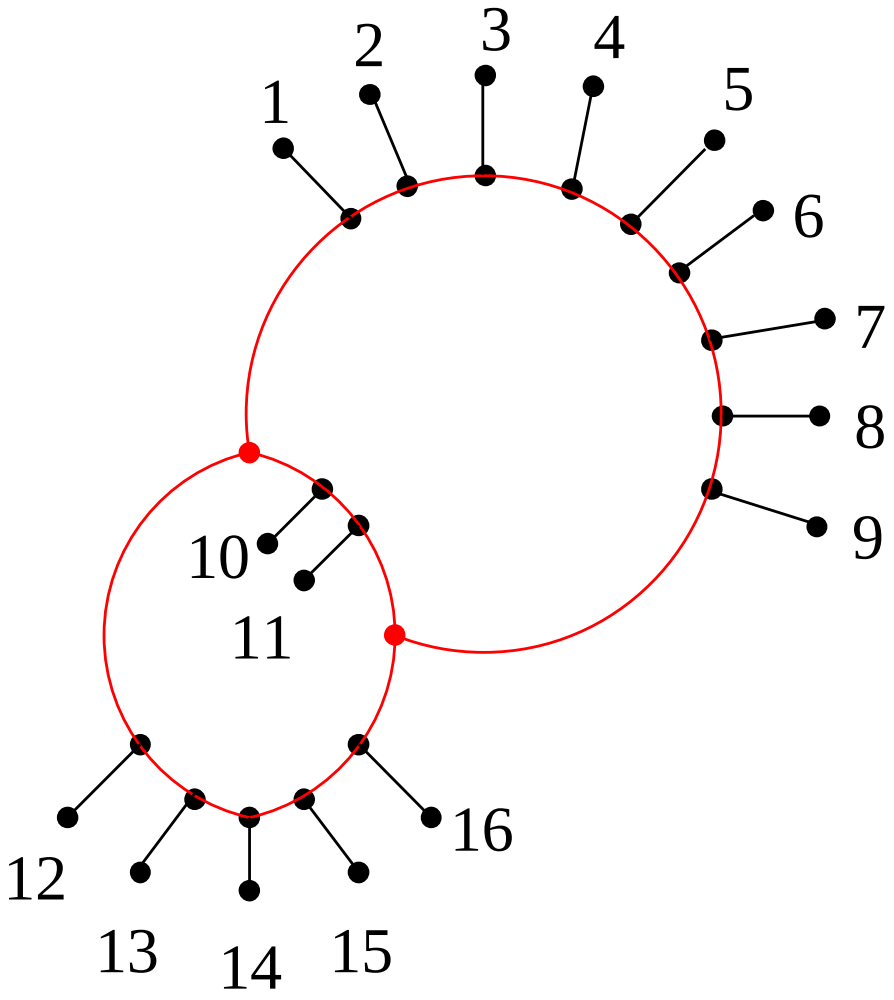
Backbones of phylogenetic networks

Can we bound the number of leaves that decorate a single side of *G*?

This is closely related to the concept of *breakpoints.*

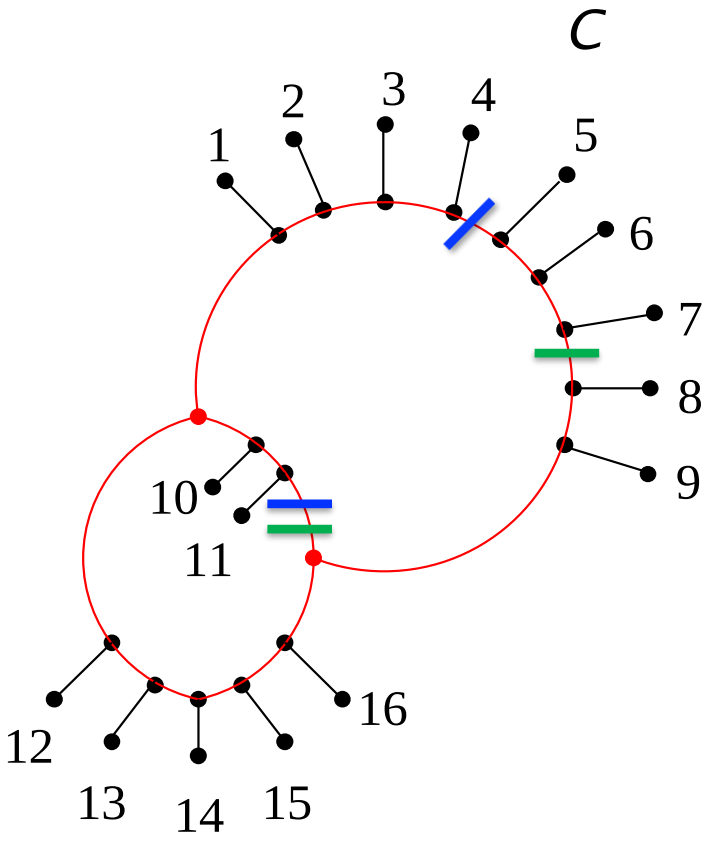
In a network with *r(N)=k*, any phylogenetic tree embedded in it can be retrieved by cutting in *k* places: these are the breakpoints.

So if there are two trees embedded in the network, there will be 2k breakpoints in total (i.e. k per tree).



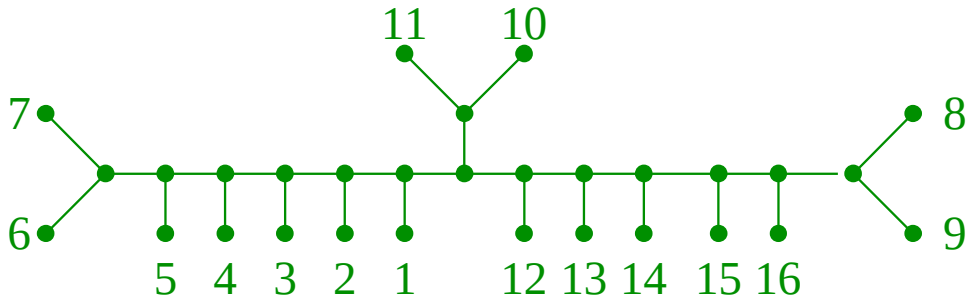
network *N* with *r(N)=2* with no pendant subtree

Breakpoints

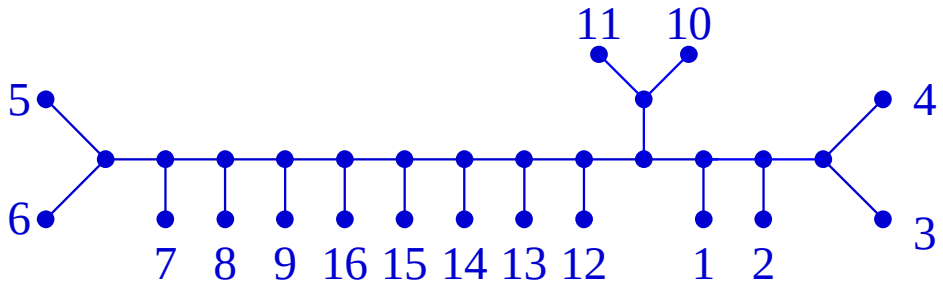


r(N)=2

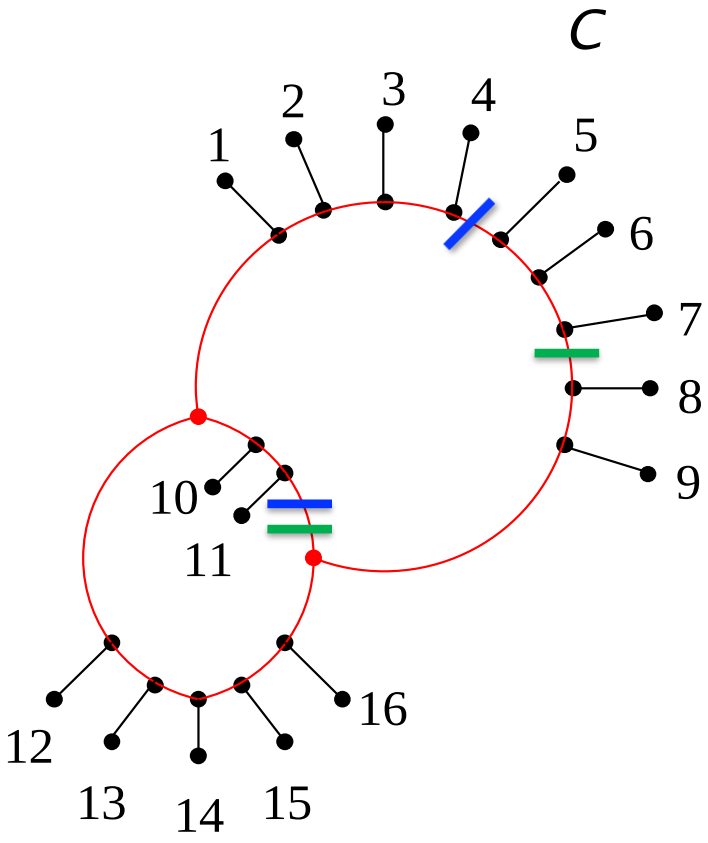
T



T'

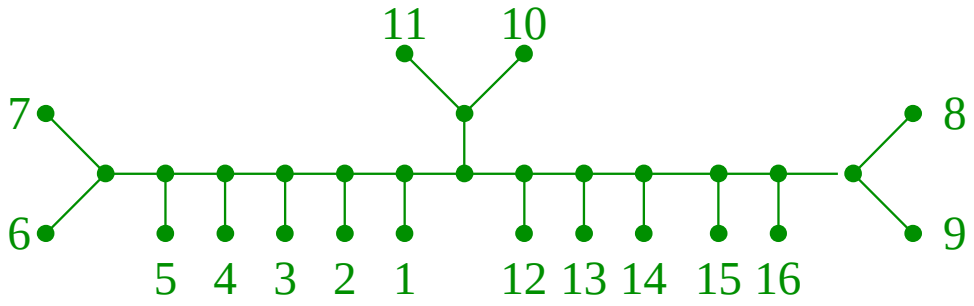


Assuming the chain reduction had been applied to exhaustion, this would not be possible!

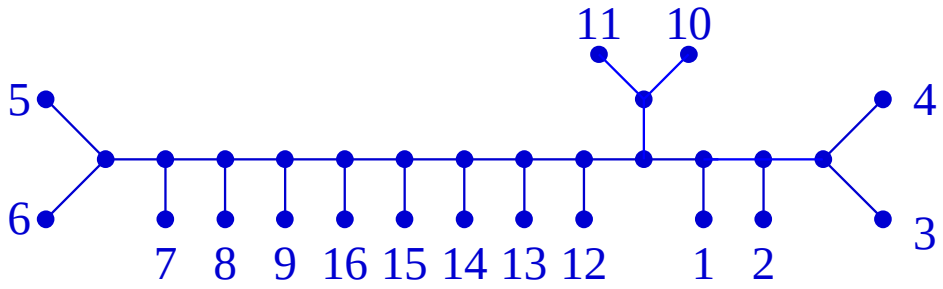


r(N)=2

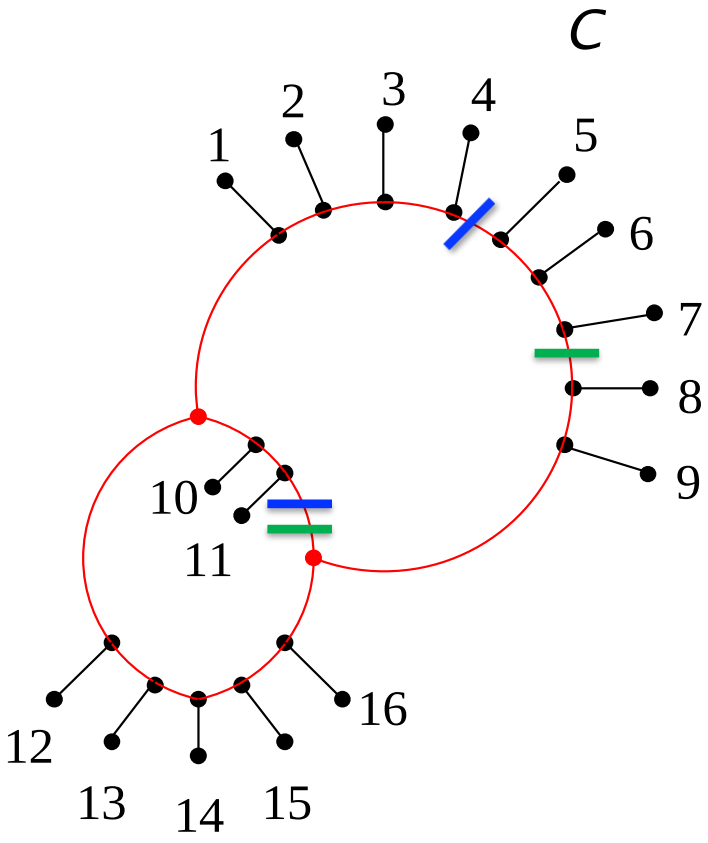
T



T'

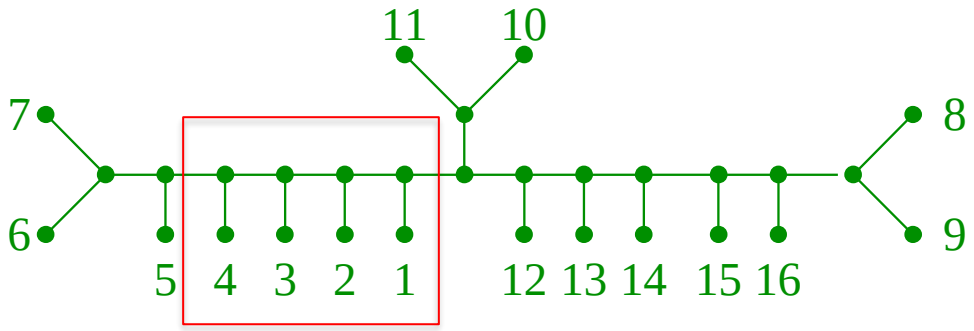


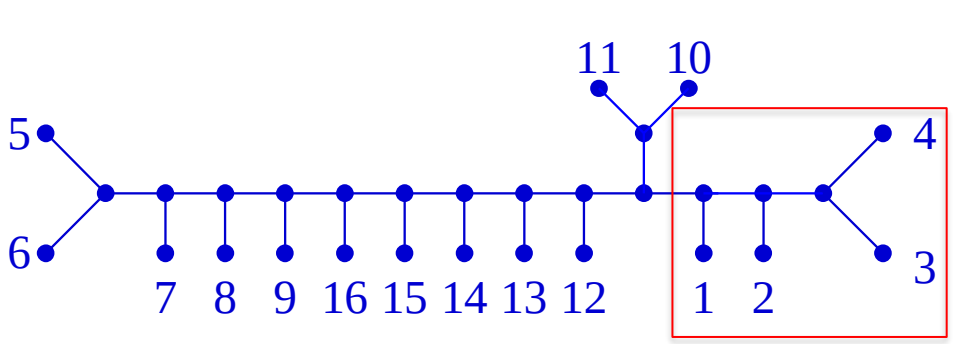
Assuming the chain reduction had been applied to exhaustion, this would not be possible!



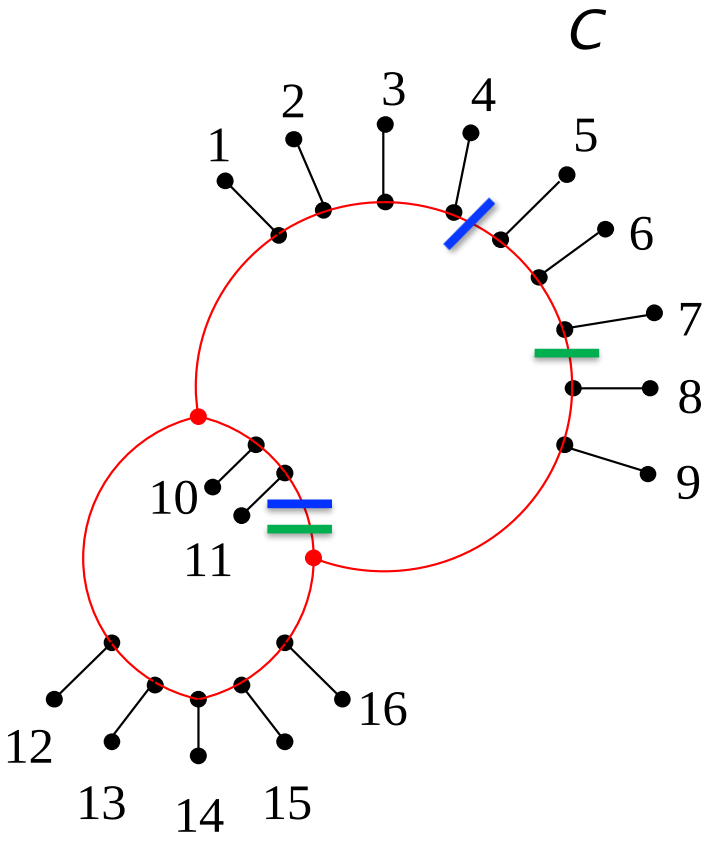
r(N)=2

Unreduced common 4-chain





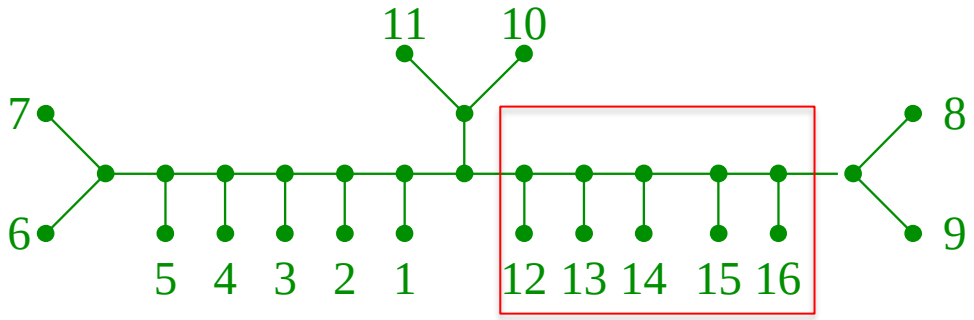
Assuming the chain reduction had been applied to exhaustion, this would not be possible!



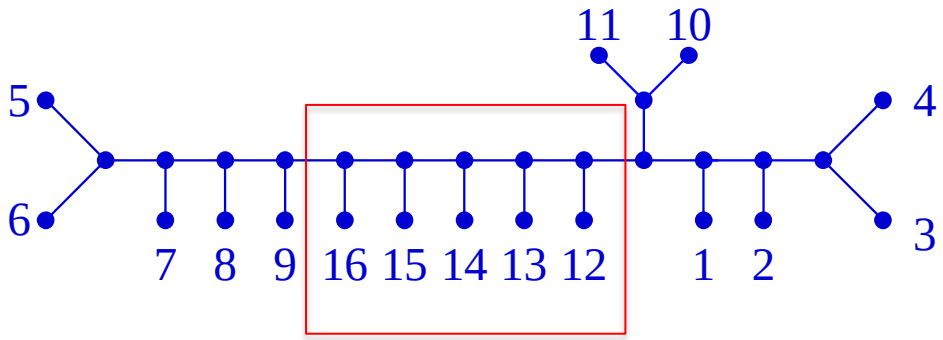
r(N)=2

Unreduced common 5-chain

T



T'



Breakpoint Lemma. (K. and Linz, 2018).

Let S and S' be two trees with no common pendant subtree of size at least 2 and no common chain of length at least 4. Let N be a network that embeds S and S', and let C be a side of N. Let n denote the number of leaves on C.

Then,

- $n \leq 3$ if C has no breakpoints relative to S and S',
- $n \leq 6$ if C has one breakpoint relative to S and S',
- $n \leq 9$ if C has two breakpoints relative to S and S'.

Lemma. (K, and Linz, 2018).

Let S and S' be two trees on X' with no common pendant subtree of size at least 2 and no common chain of length at least 4. If $d_{TBR}(S,S') \geq 2$, then

$$|X'| \leq 15 d_{TBR}(S,S') - 9.$$

Proof sketch. There are $2k$ breakpoints, to distribute across $3(k-1)$ sides.

Sides with 0, 1, 2 breakpoints can have at most 3, 6, 9 leaves respectively.

The maximum of the counting equation is $15k-9$.

From *15k* to *11k…*

Idea. We described 5 (!) new reduction rules which were engineered to reduce the critical numbers in our counting argument:

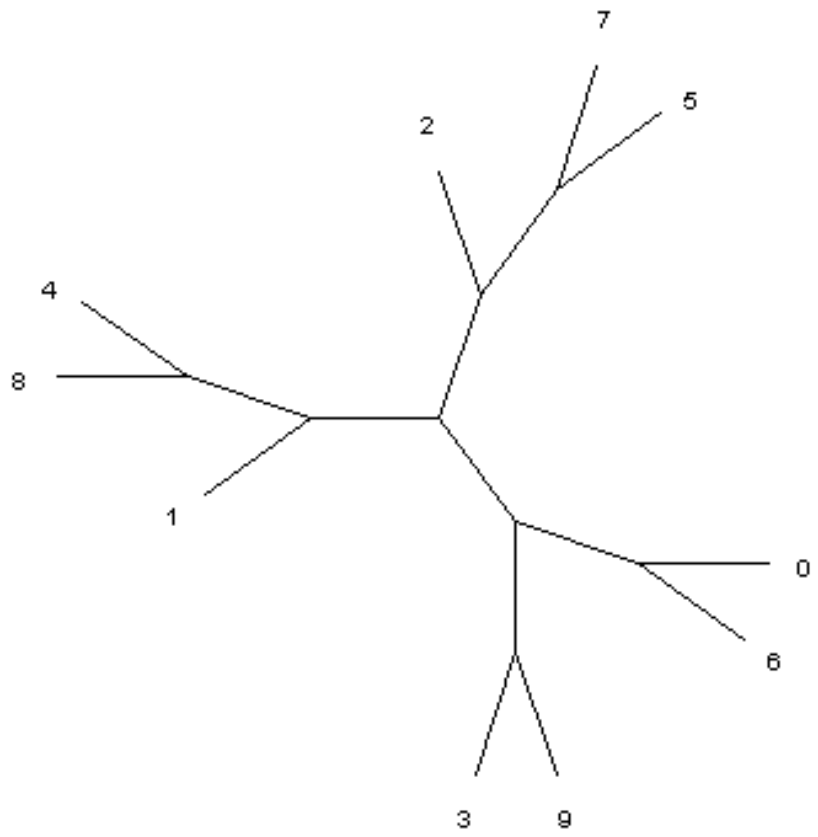
- $n \leq 3$ if C has no breakpoints,
- $n \leq 6 \rightarrow 4$ if C has one breakpoint,
- $n \leq 9 \rightarrow 4$ if C has two breakpoints.

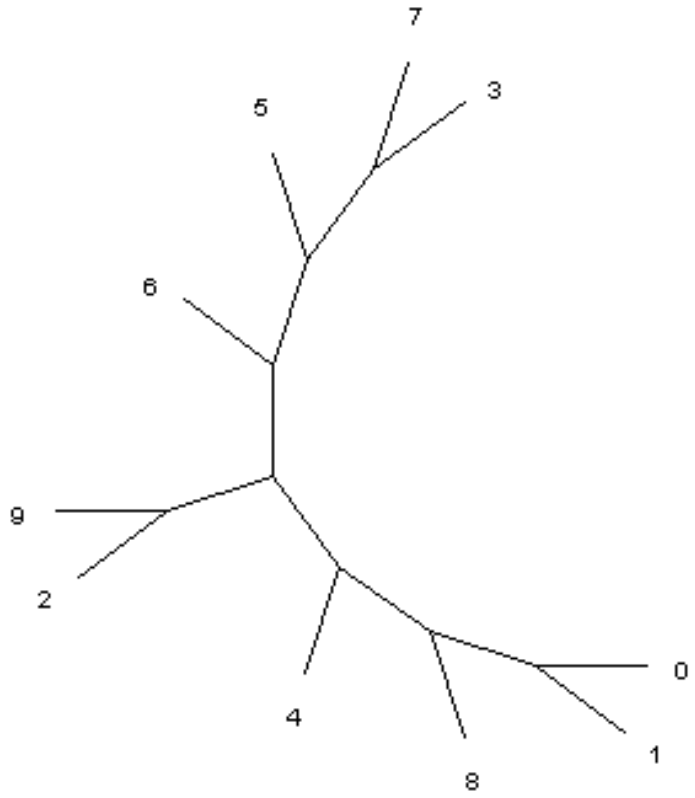
By dividing *2k* breakpoints across *3(k-1)* sides, we concluded that the size of the new kernel is at most…

$$4*2k + 3*(k-3) = 11k-9.$$

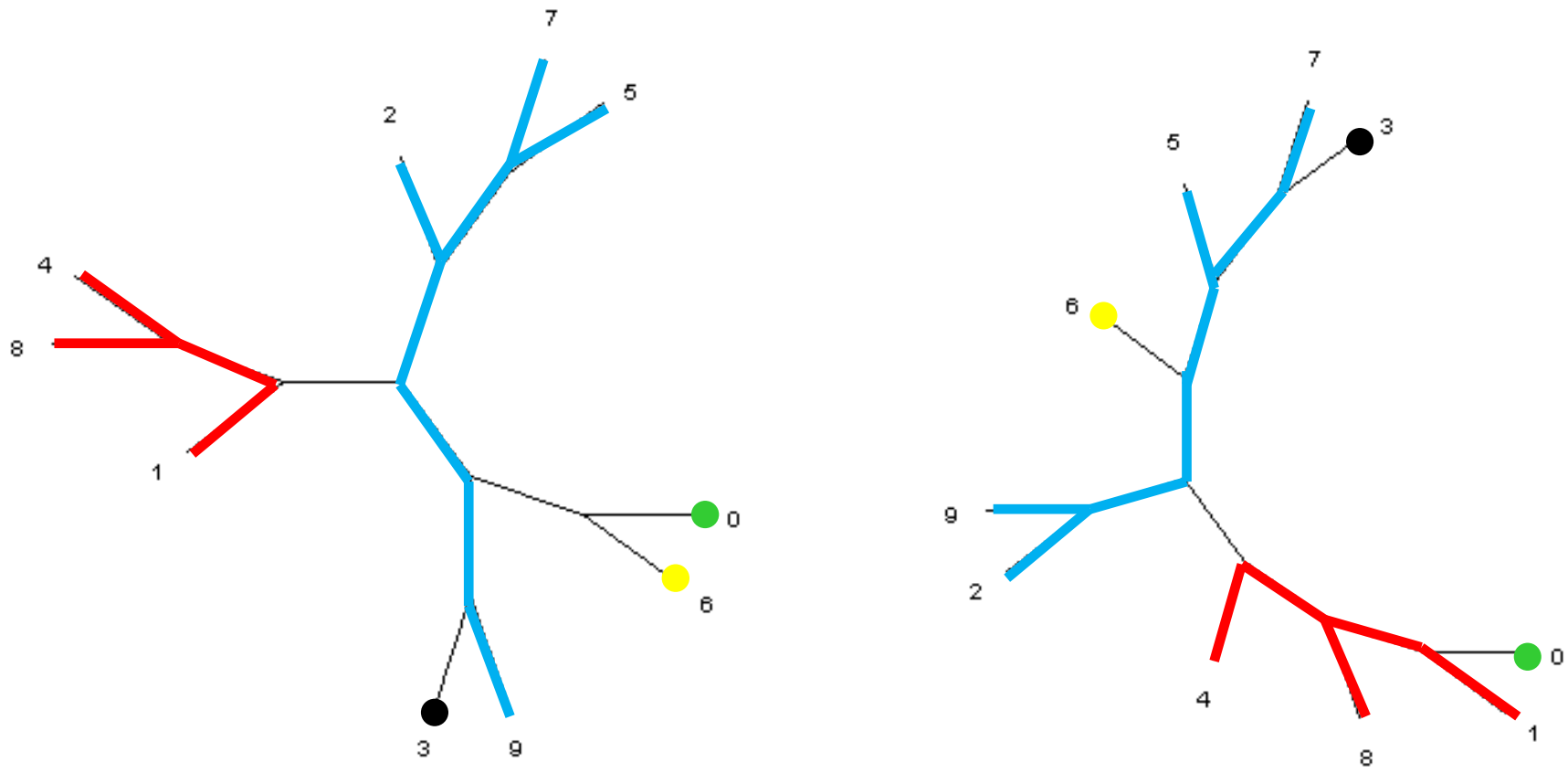
The correctness of these new rules requires use of the *agreement forest* characterization of d_{TBR}.

A third characterization of d_{TBR}



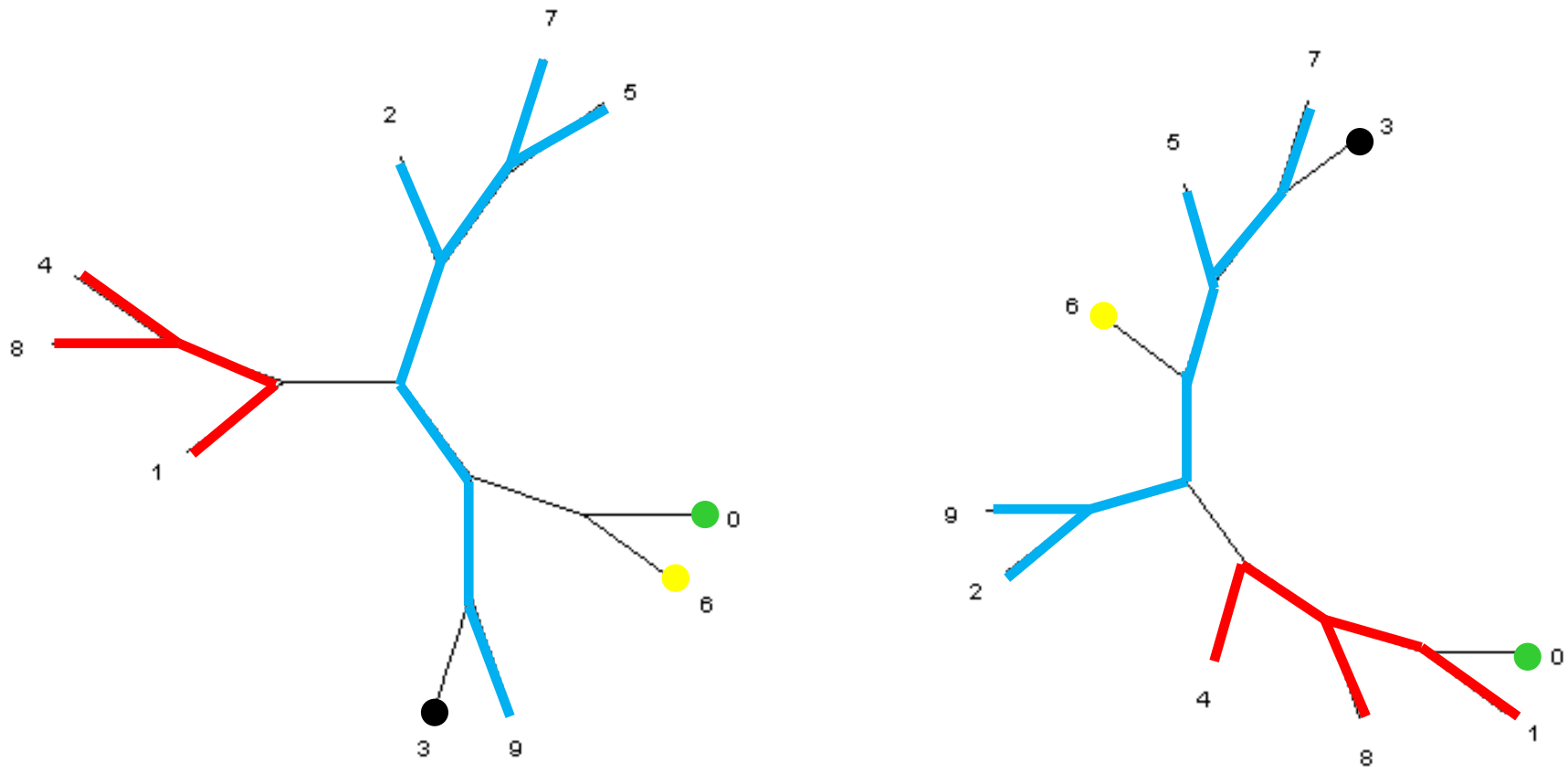


A third characterization of d_{TBR}



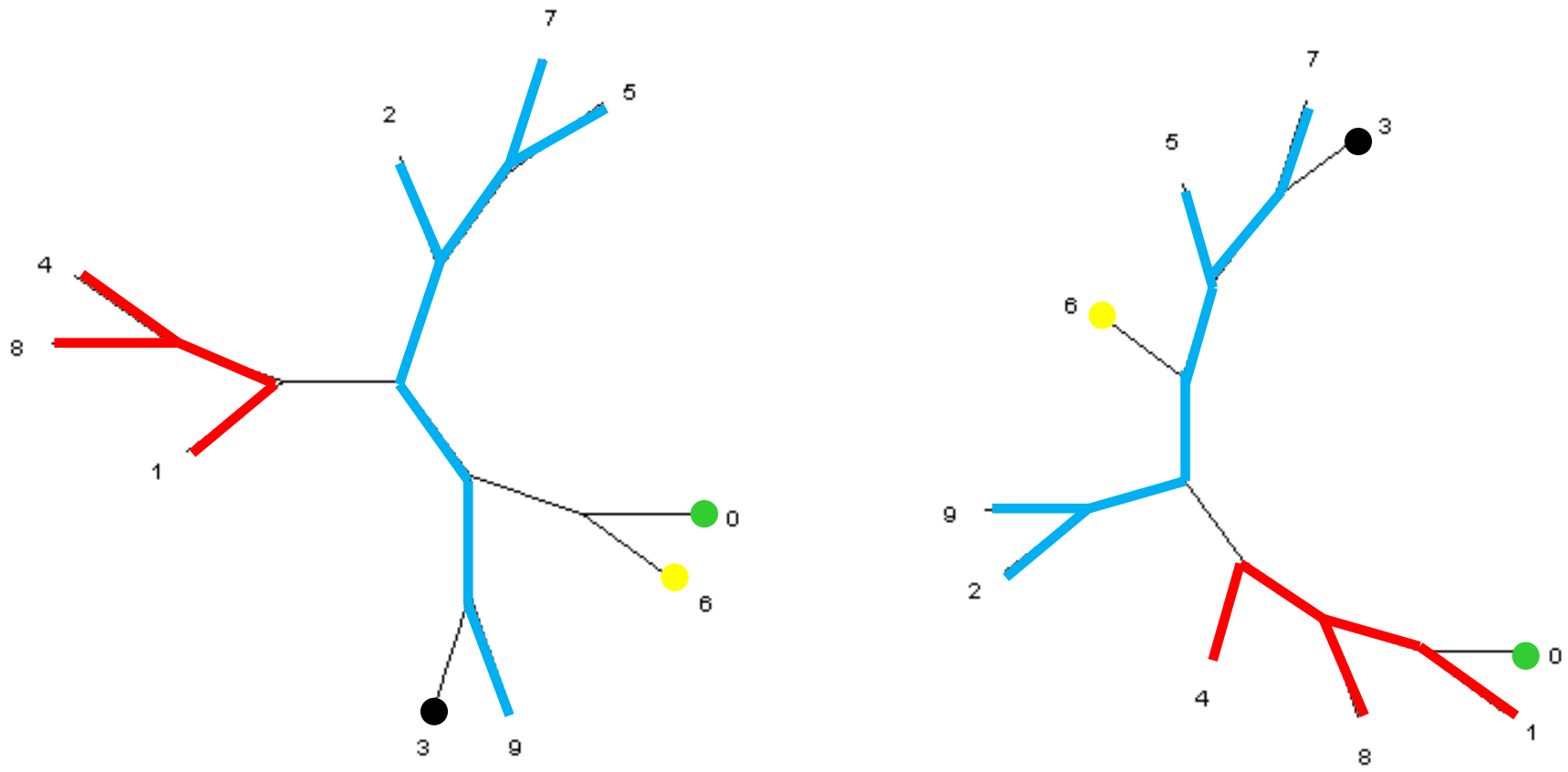
Agreement forest with 5 components

A third characterization of d_{TBR}



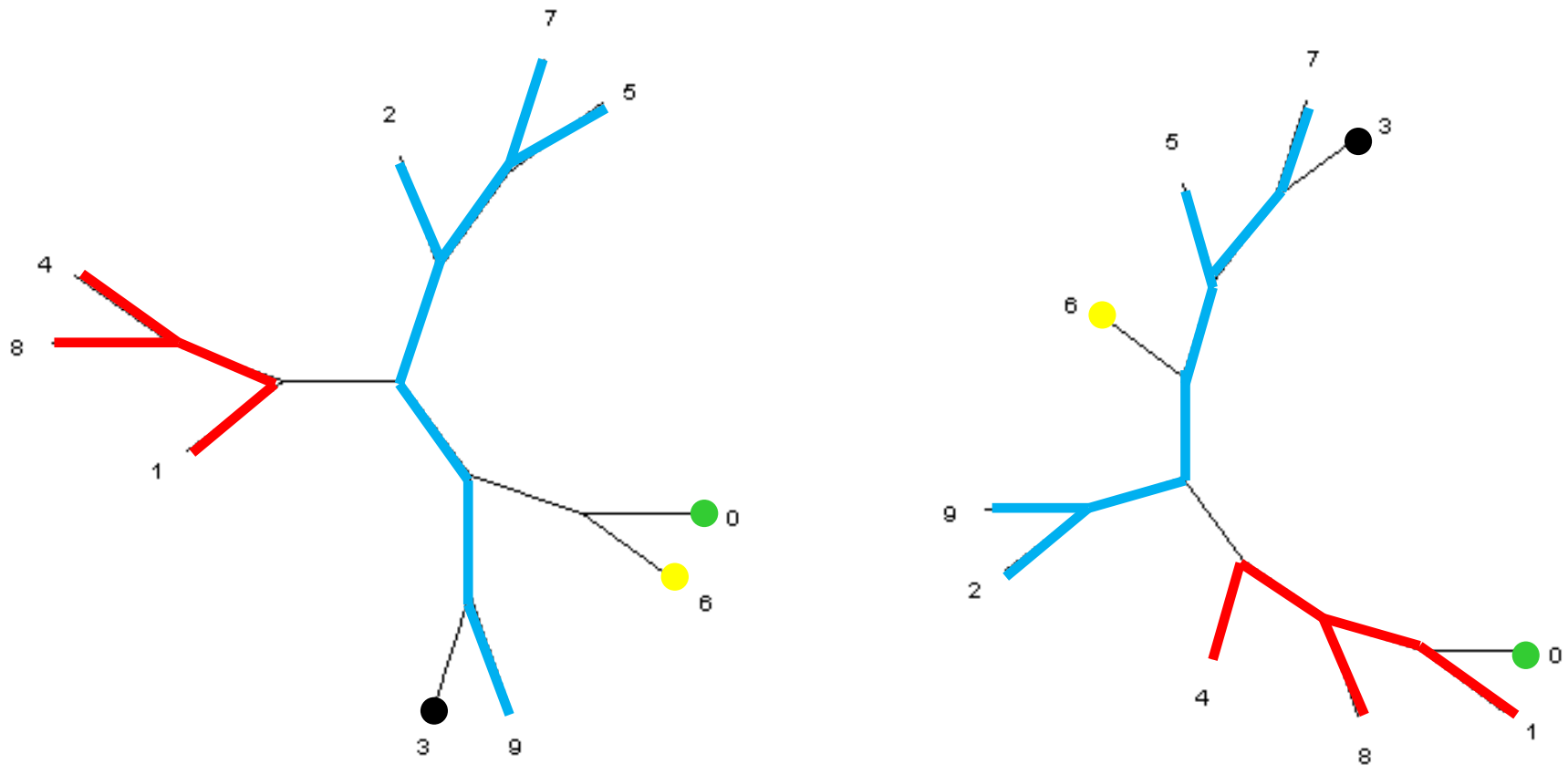
Fewer components are not possible: this is a *maximum agreement forest* (MAF)

A third characterization of d_{TBR}



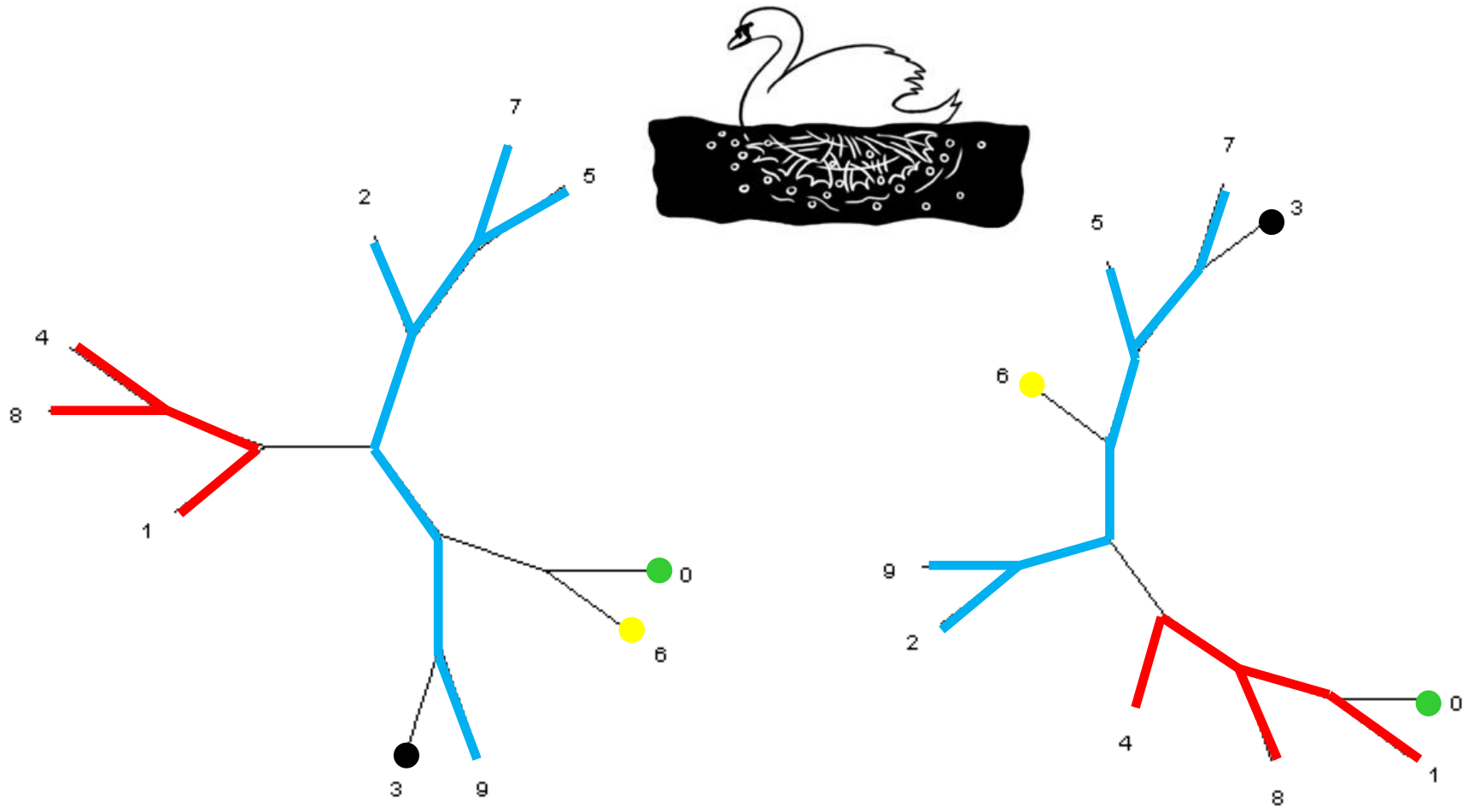
Allen and Steel 2001:
d_{TBR} is equal to the number of components in a MAF, minus 1.

A third characterization of d_{TBR}



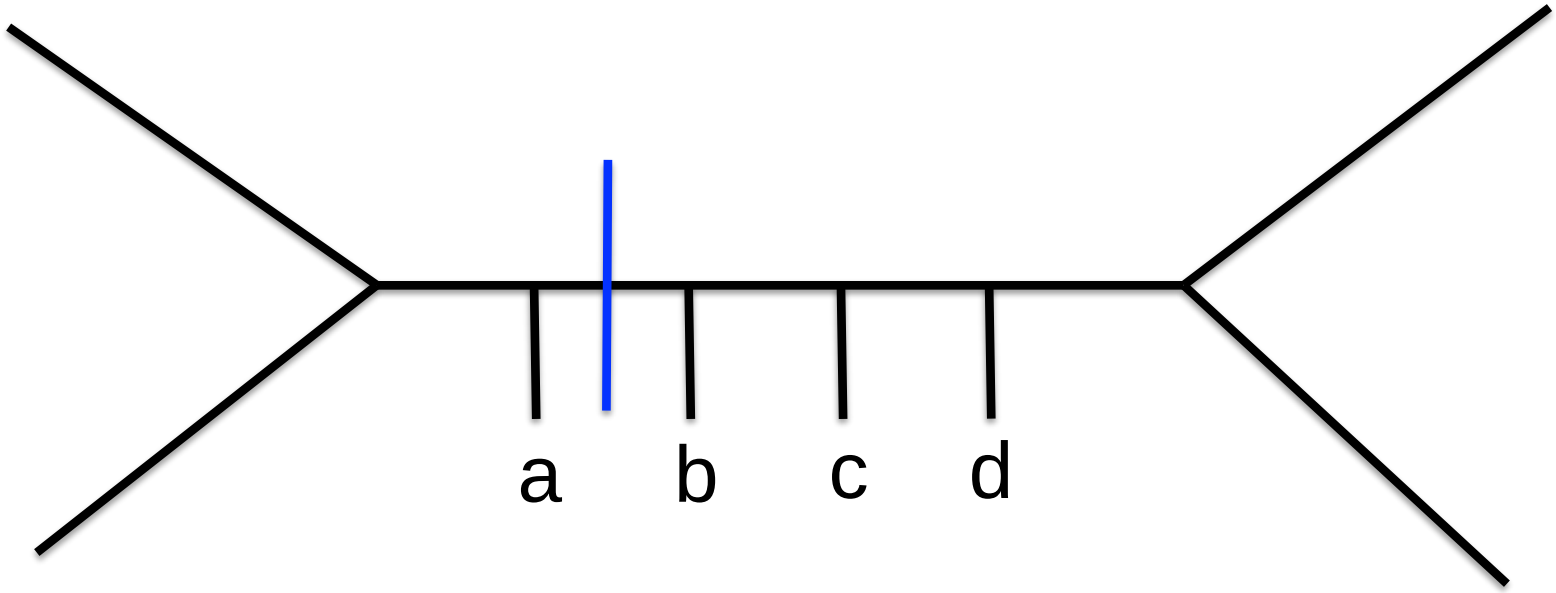
This characterization of d_{TBR} is used extensively in the new reductions that follow, but due to lack of time it will be largely hidden in this talk.

A third characterization of d_{TBR}

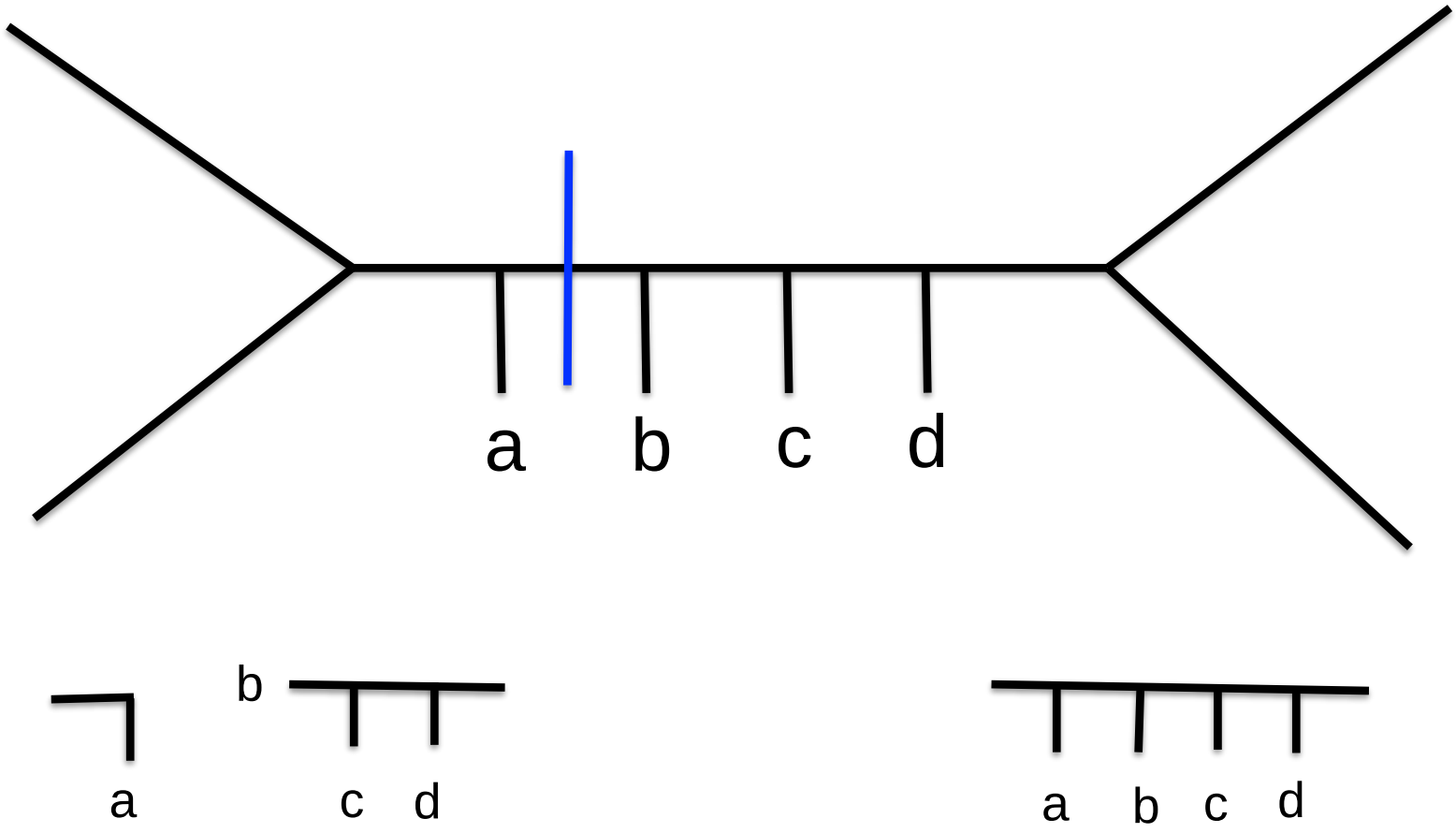


This characterization of d_{TBR} is used extensively in the new reductions that follow, but due to lack of time it will be largely hidden in this talk.

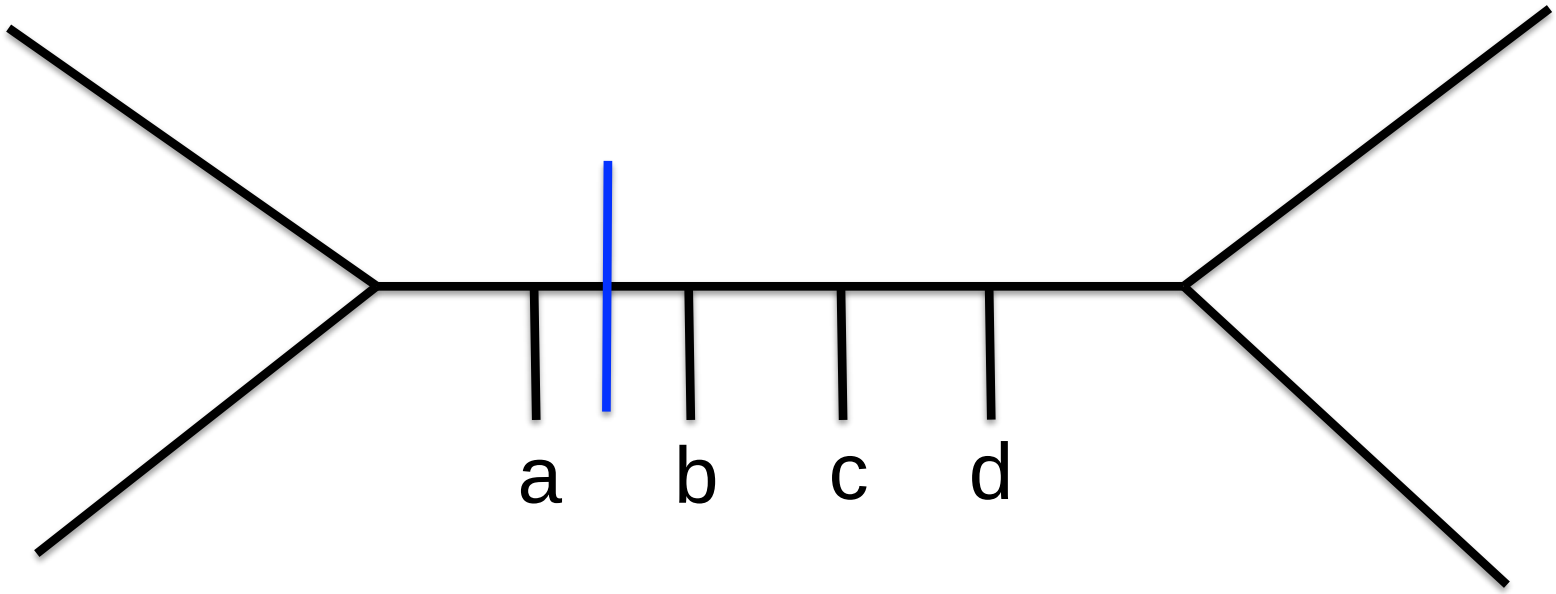
First bottleneck: a “1|3” side

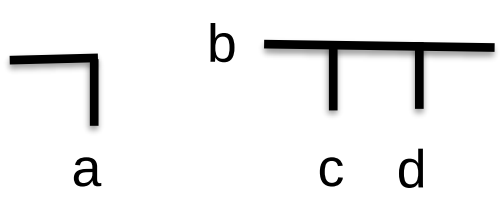


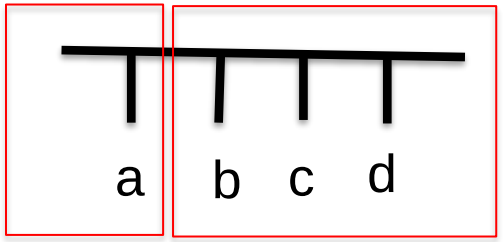
First bottleneck: a “1|3” side



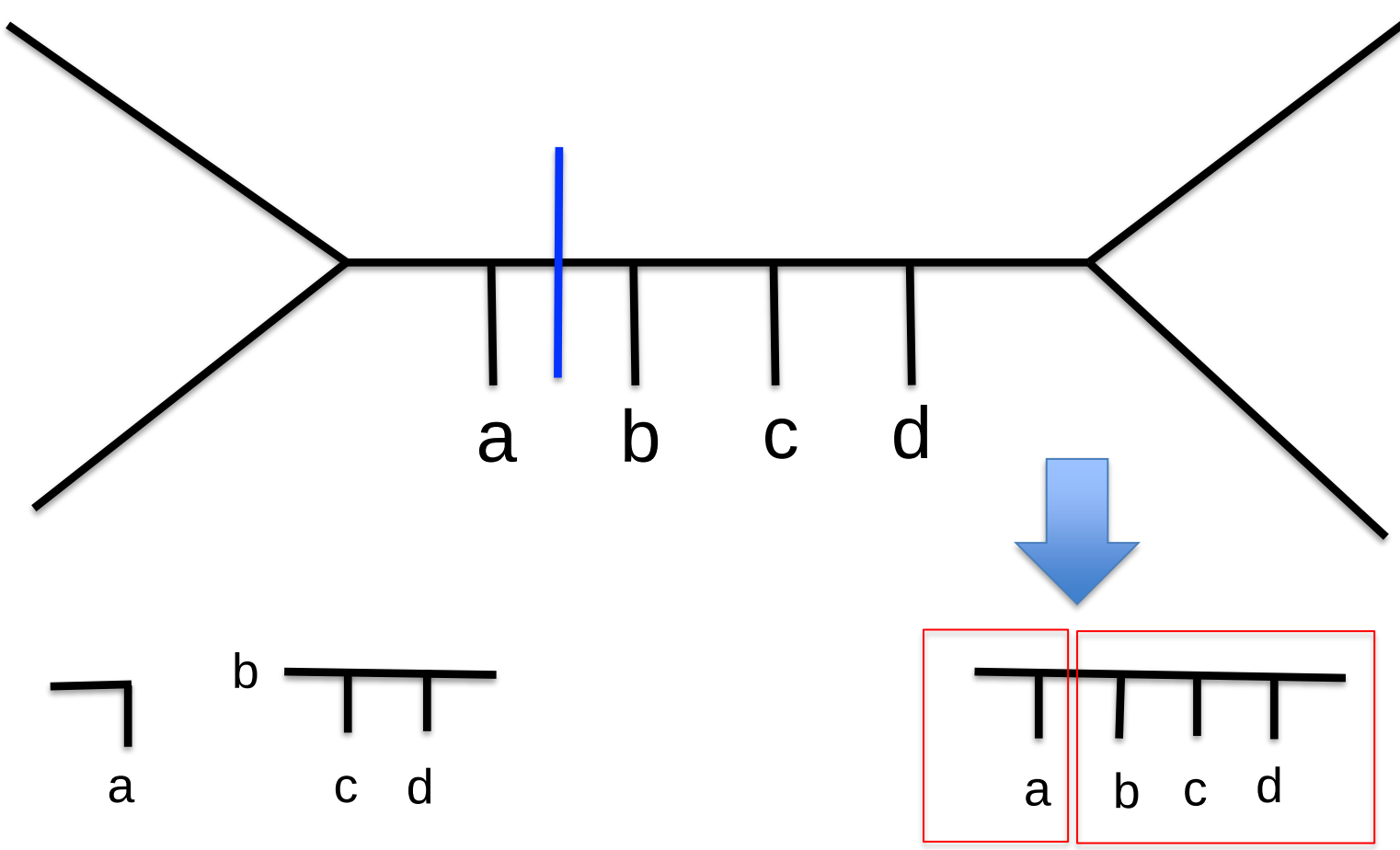
First bottleneck: a “1|3” side



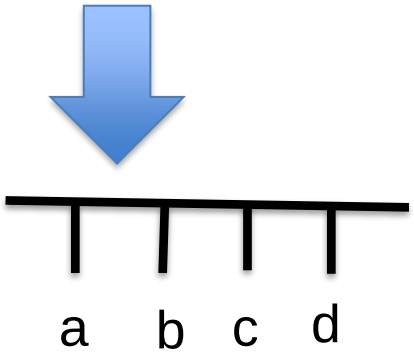


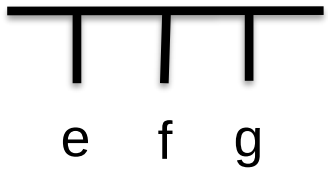


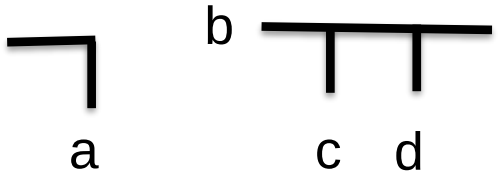
We can prove that no blocks of some maximum agreement forest cross this edge, but deleting it would disconnect the trees and produce a different type of problem ☹

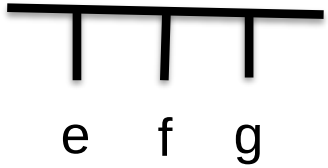


Solution: find another 3-chain (e,f,g) common to both trees….

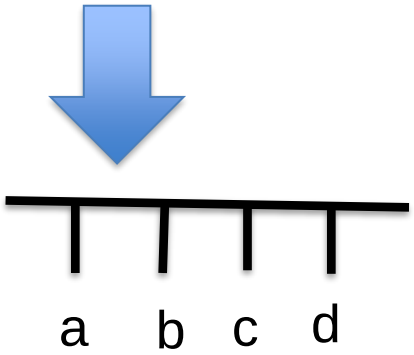


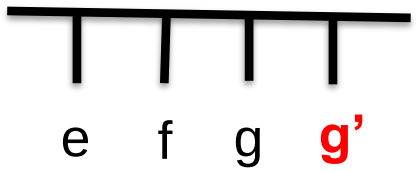


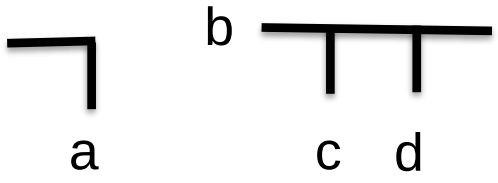


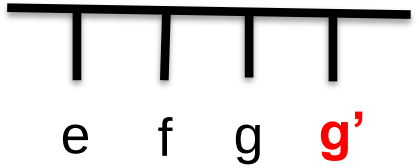


Extend it to length 4…..

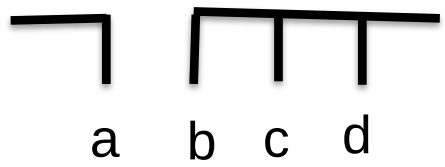


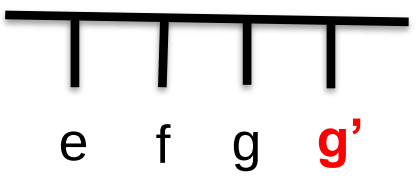


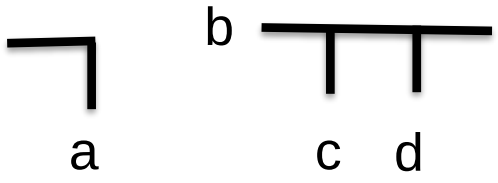


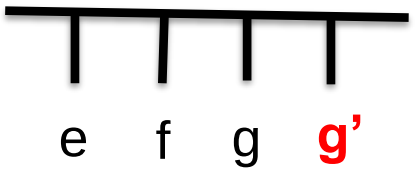


And move the edge we wanted to cut…







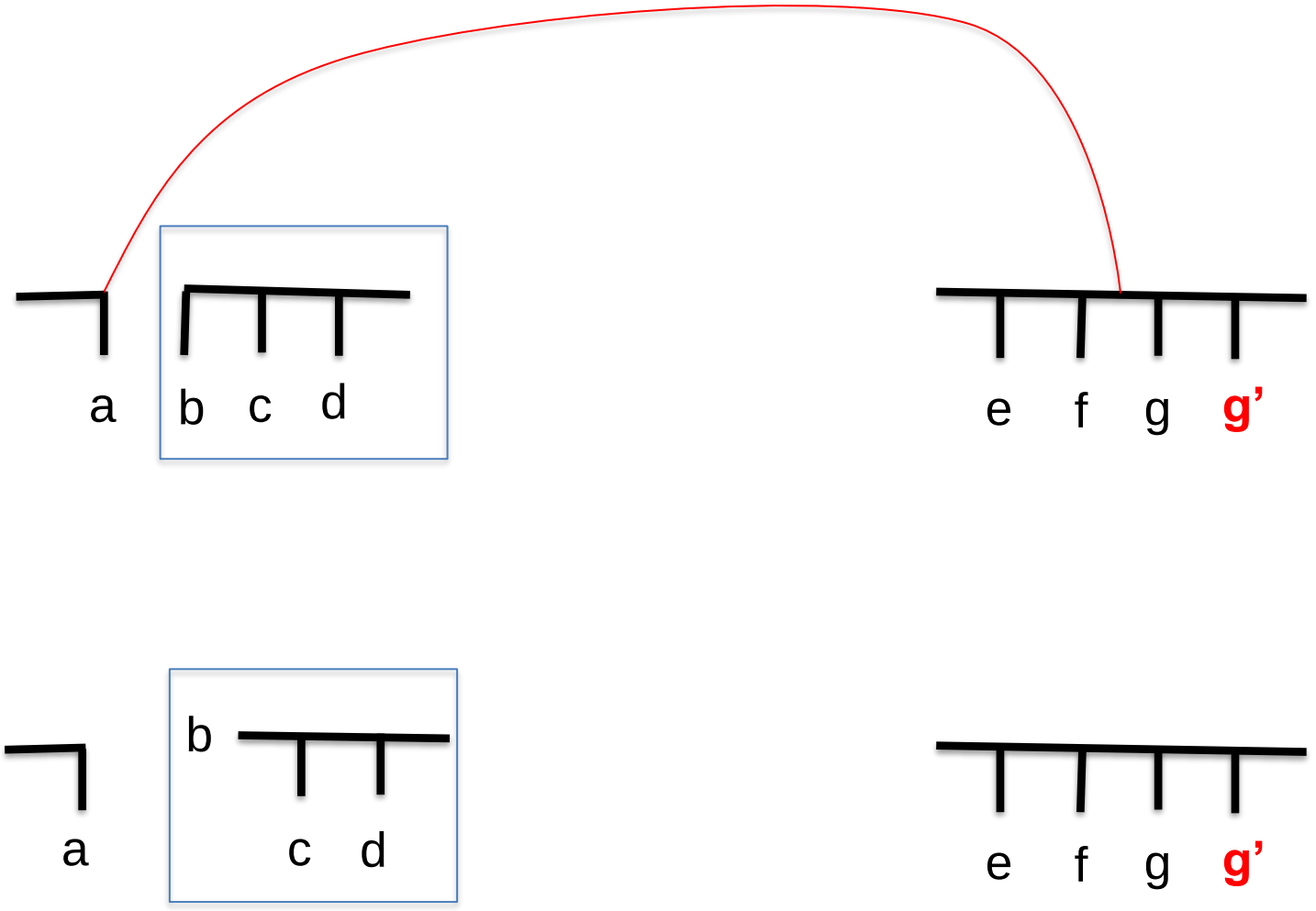



a
b
c
d
e
f
g
g’


a
b
c
d
e
f
g
g’

This is distance preserving…but creates a common subtree that we can reduce!

This is distance preserving…but creates a common subtree that we can reduce!



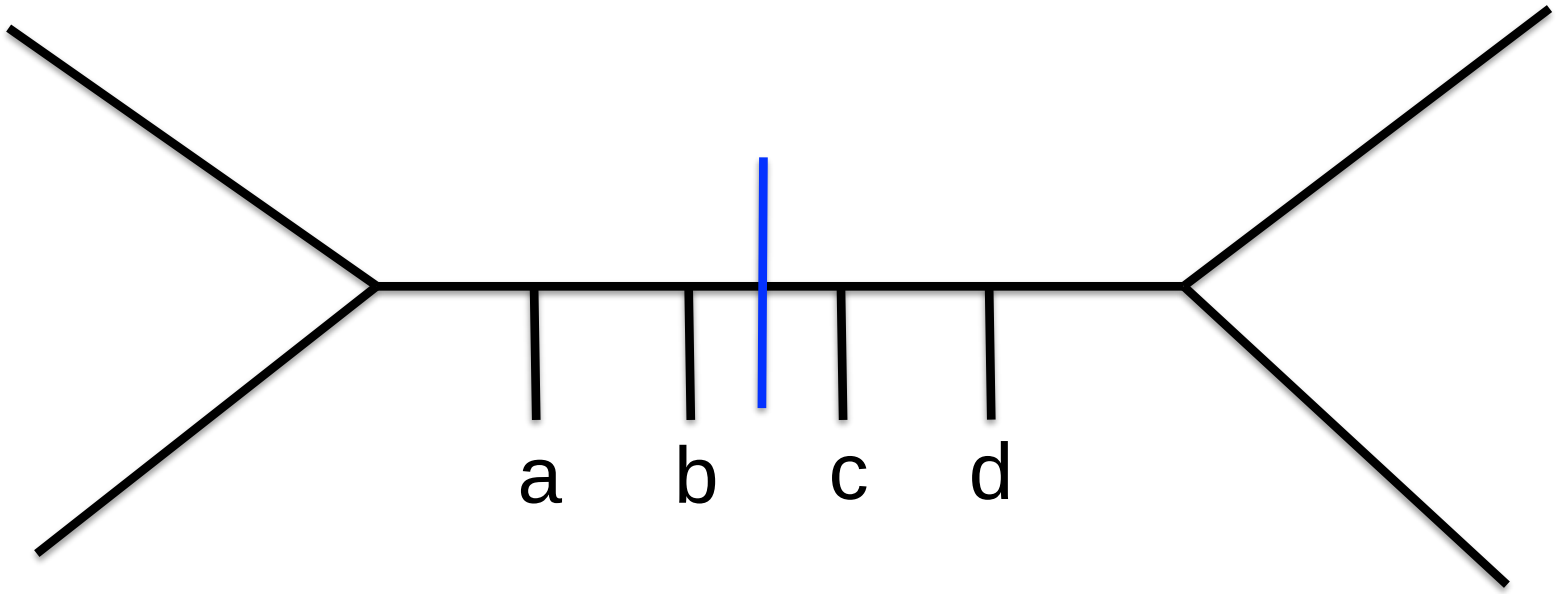
This is distance preserving…but creates a common subtree that we can reduce!

1|3 sides can eat each other…

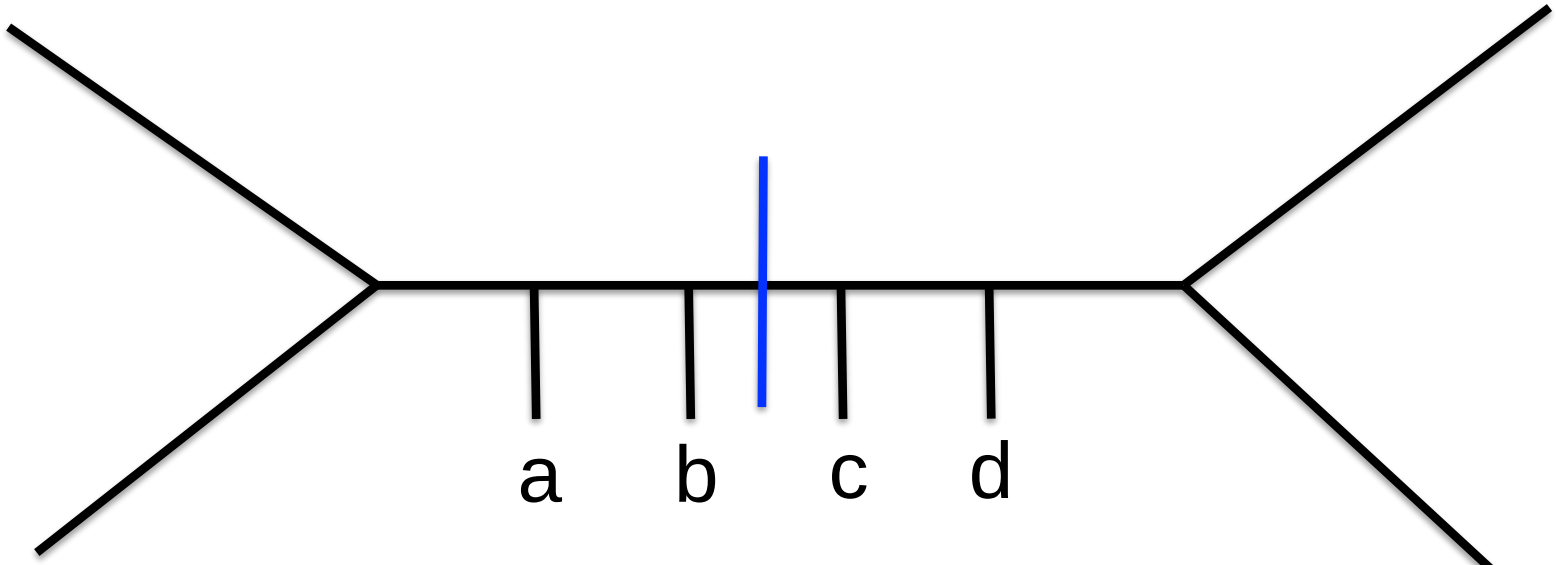
Idea. Each of these 1|3 sides contains a common chain of length 3, so you can use the chain in one 1|3 side to trigger the reduction of another 1|3 side!

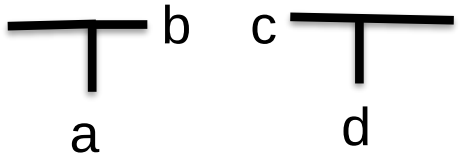
After doing this to exhaustion, there can be at most one 1|3 side.

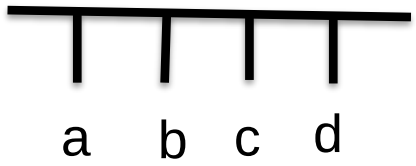
Second bottleneck: a “2|2” side



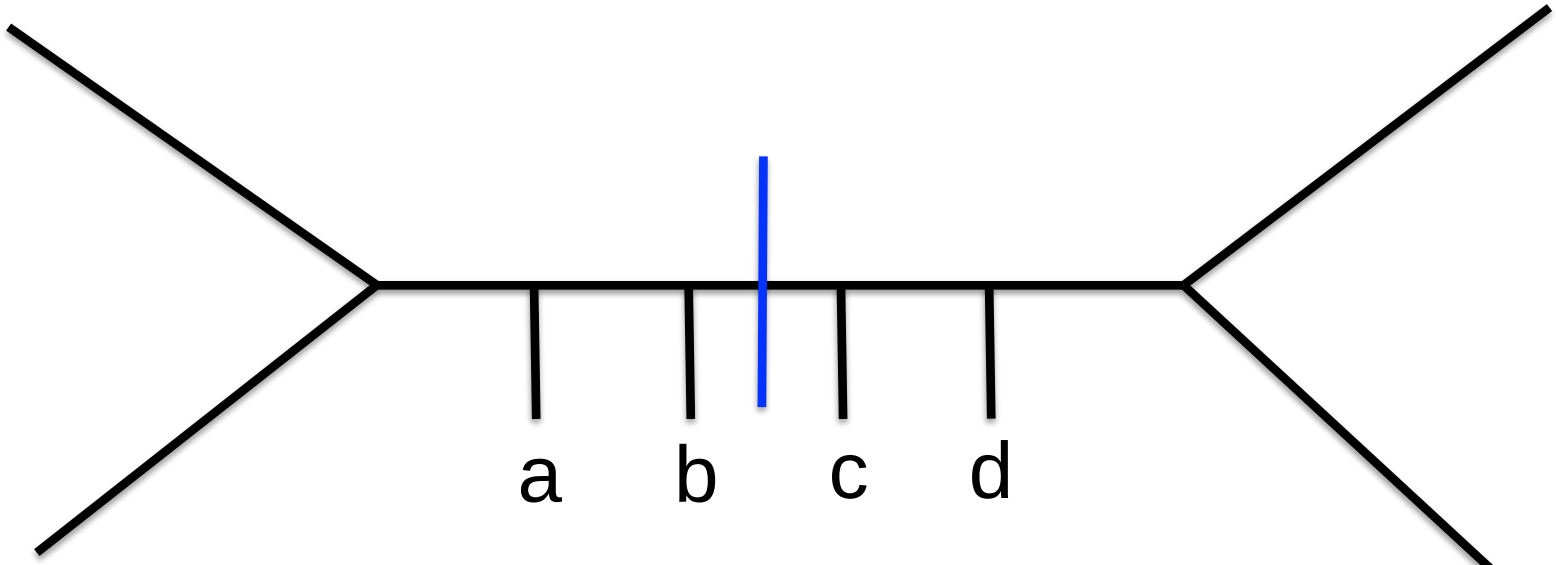
Second bottleneck: a “2|2” side

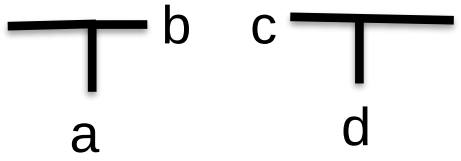


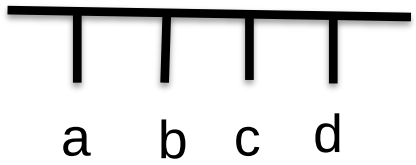




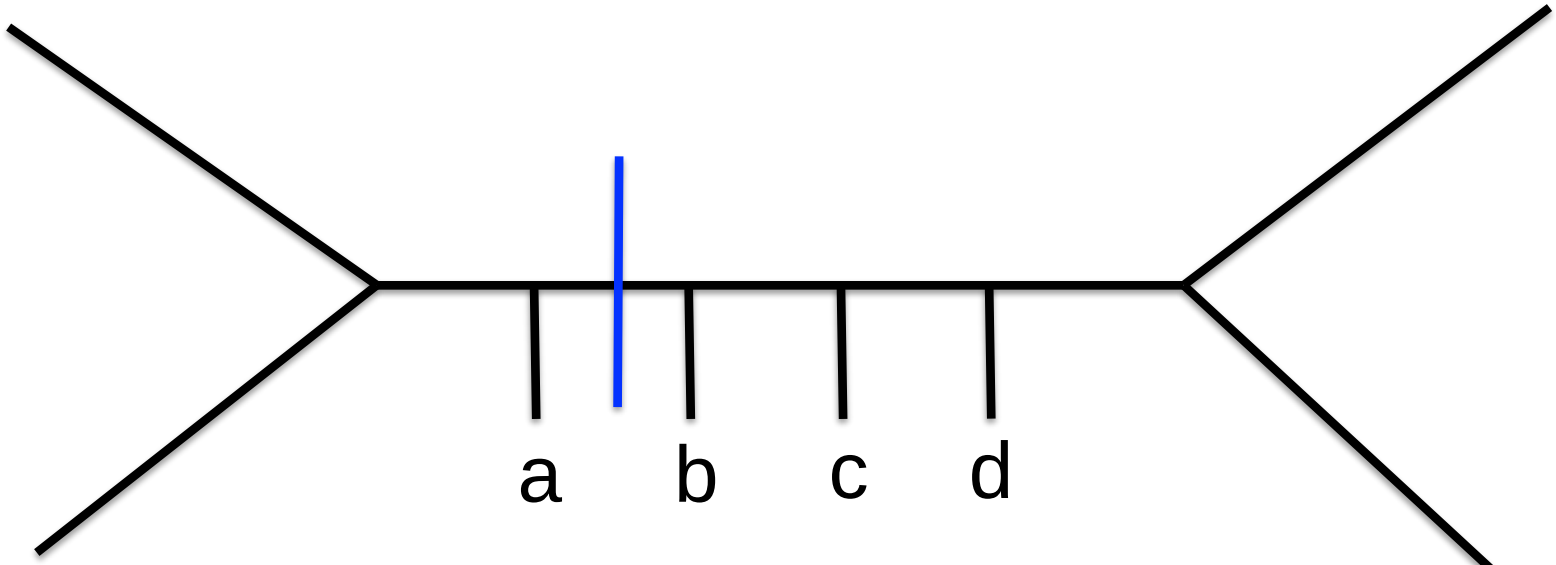
Suppose we could turn this into a 1|3 side…

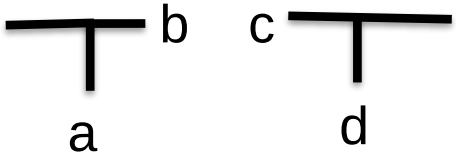


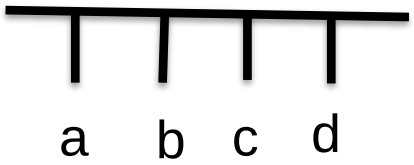




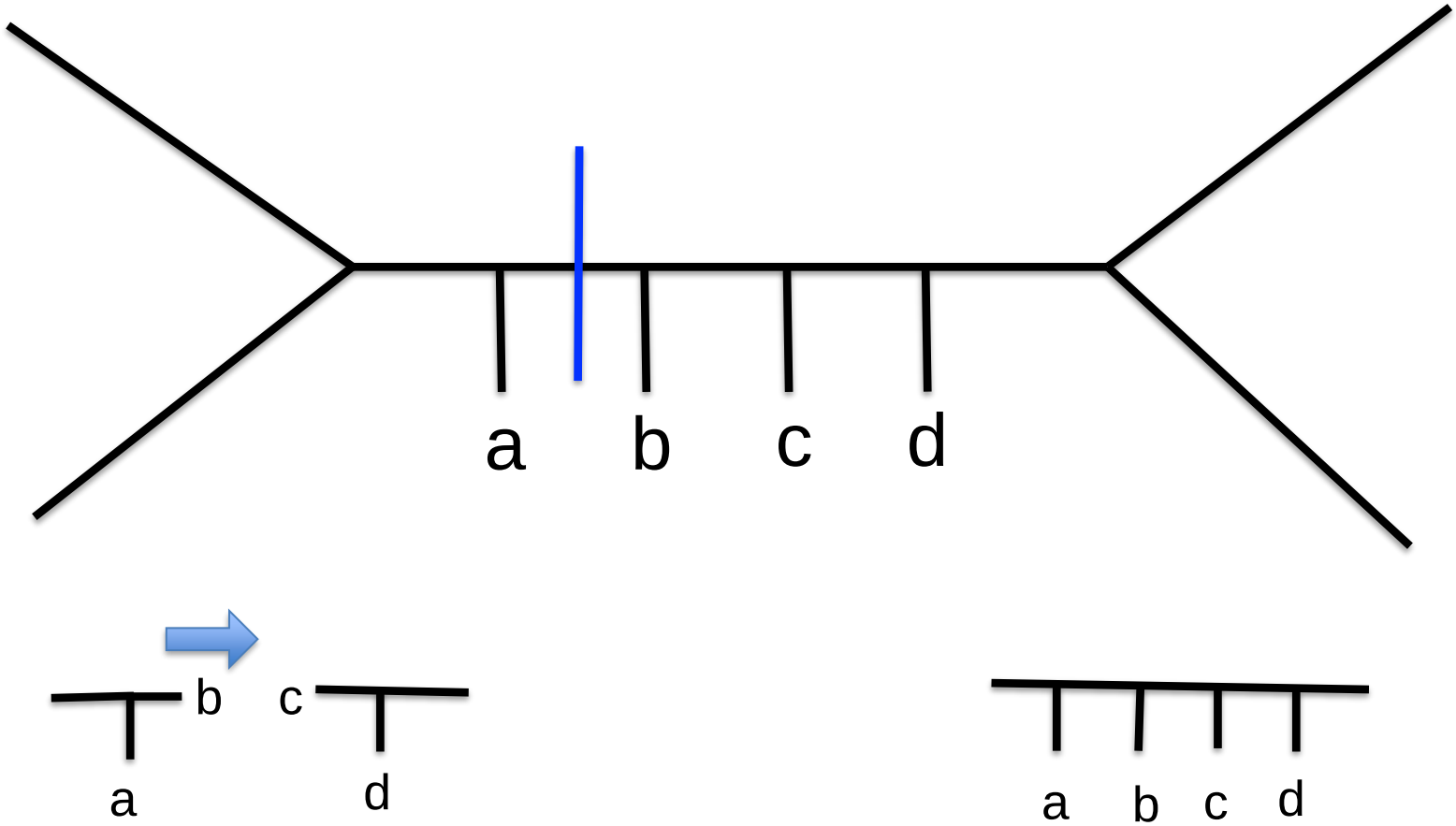
Suppose we could turn this into a 1|3 side…



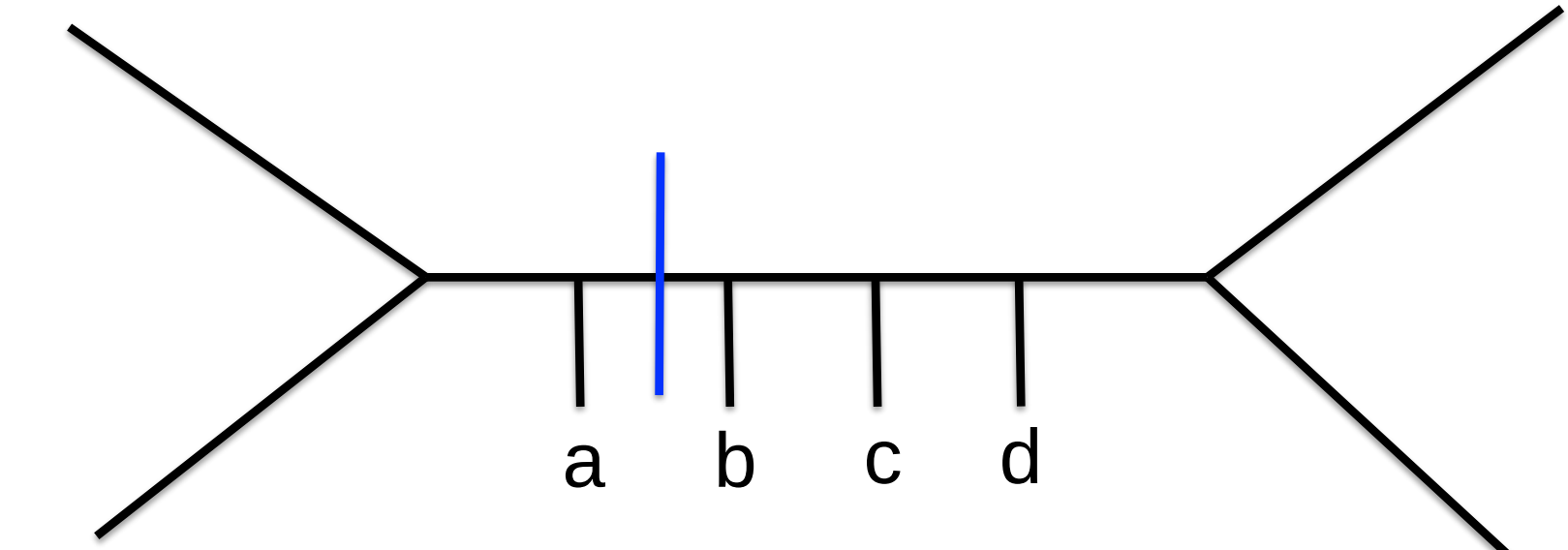


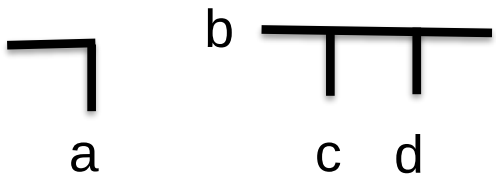


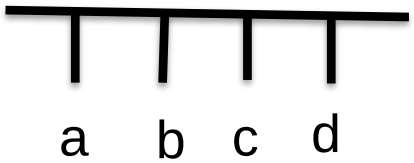
Suppose we could turn this into a 1|3 side…



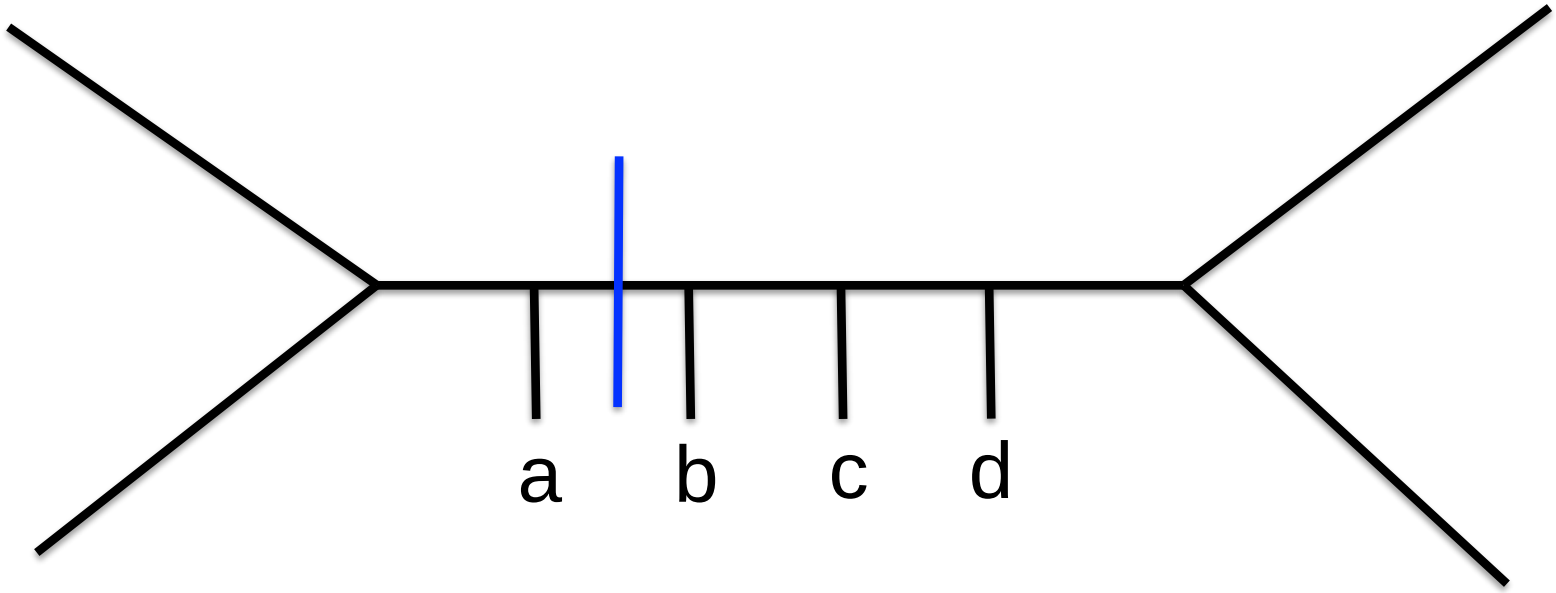
Suppose we could turn this into a 1|3 side…

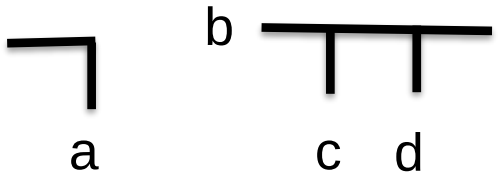


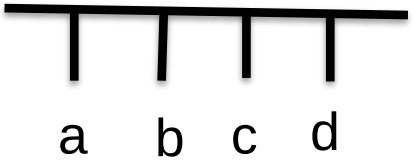




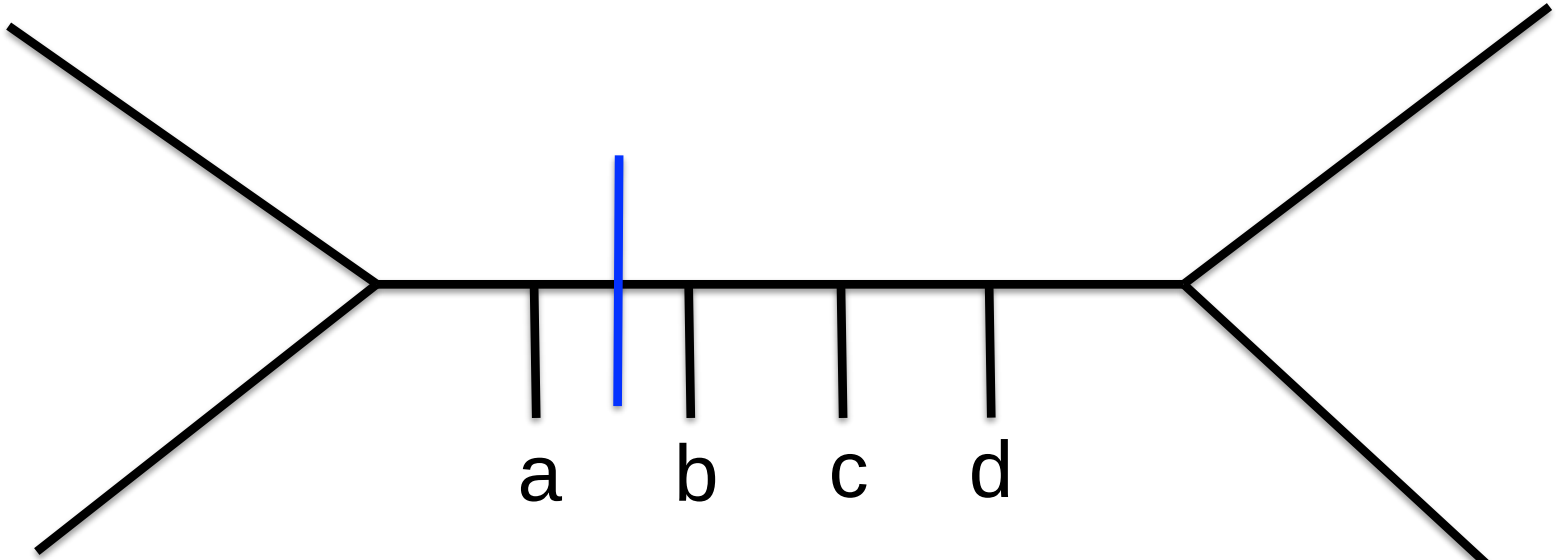
Then we could turn 2|2 sides into 1|3 sides and then use the 1|3 sides to eat each other. BUT….

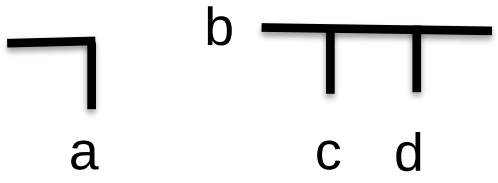


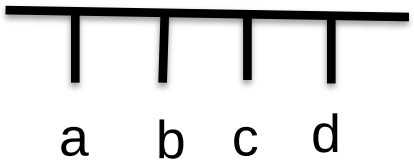


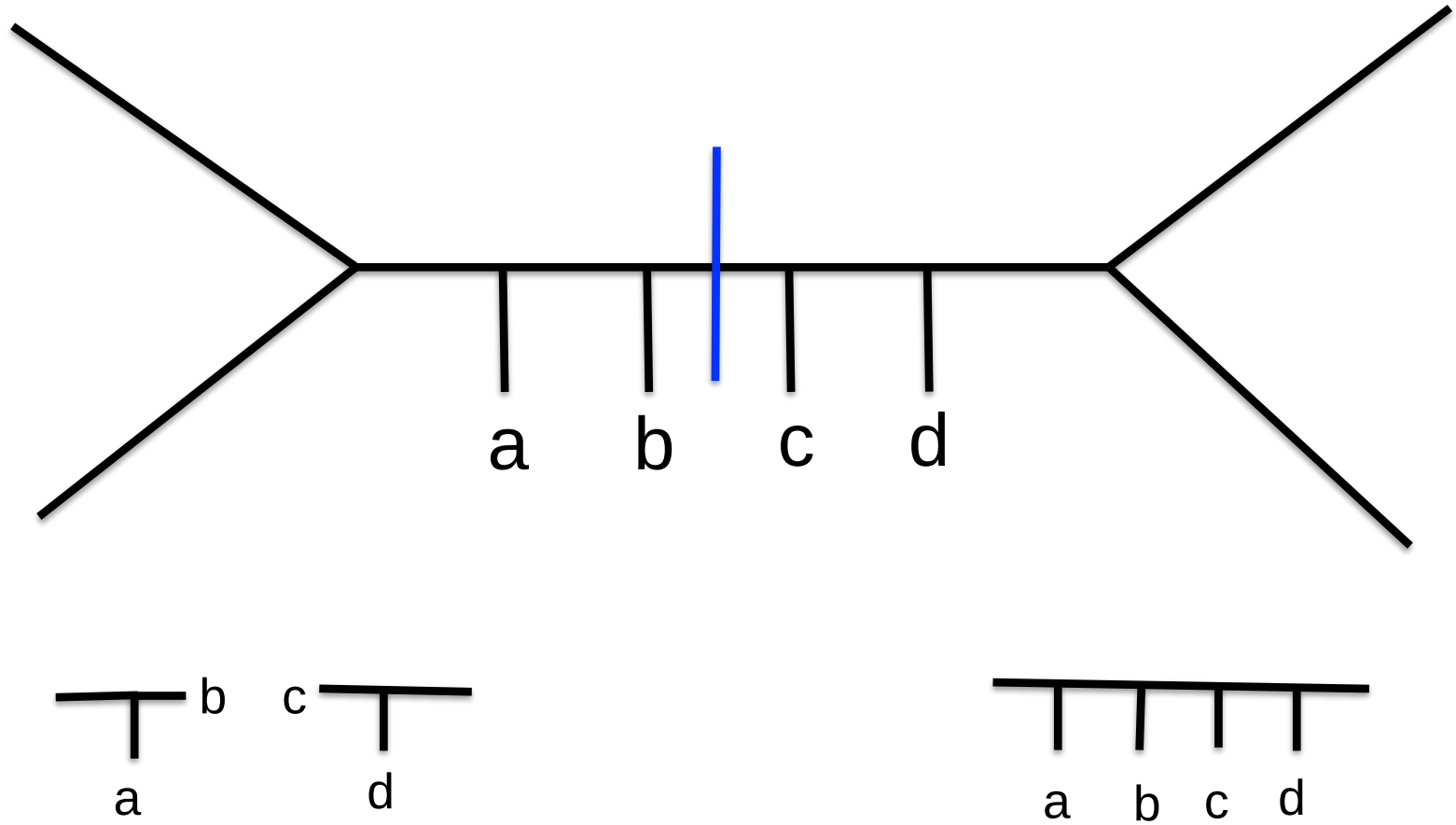


…it is not always allowed to “flip” a 2|2 side into a 1|3 side. So when is it allowed?

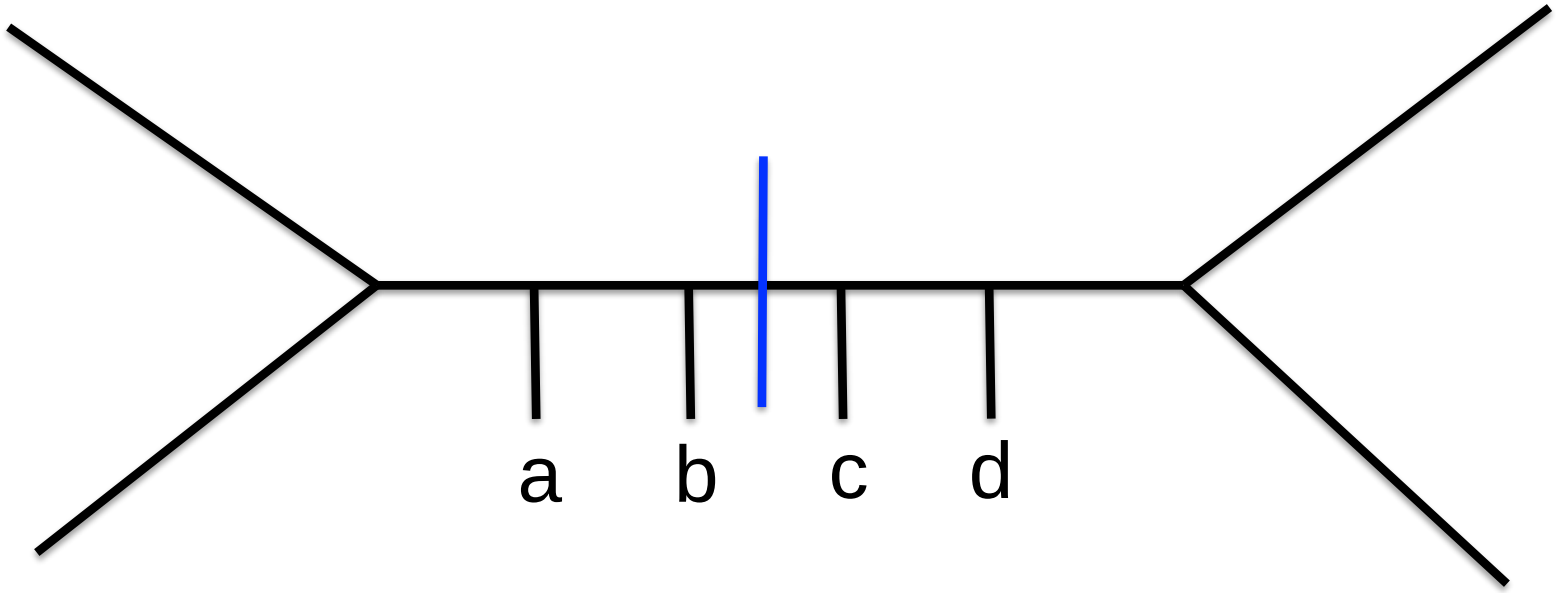


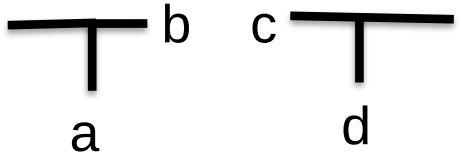


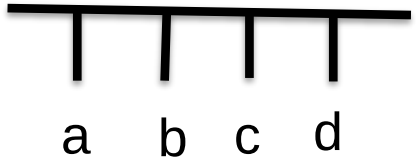



a
b
c
d
b
c
a
d
a
b
c
d

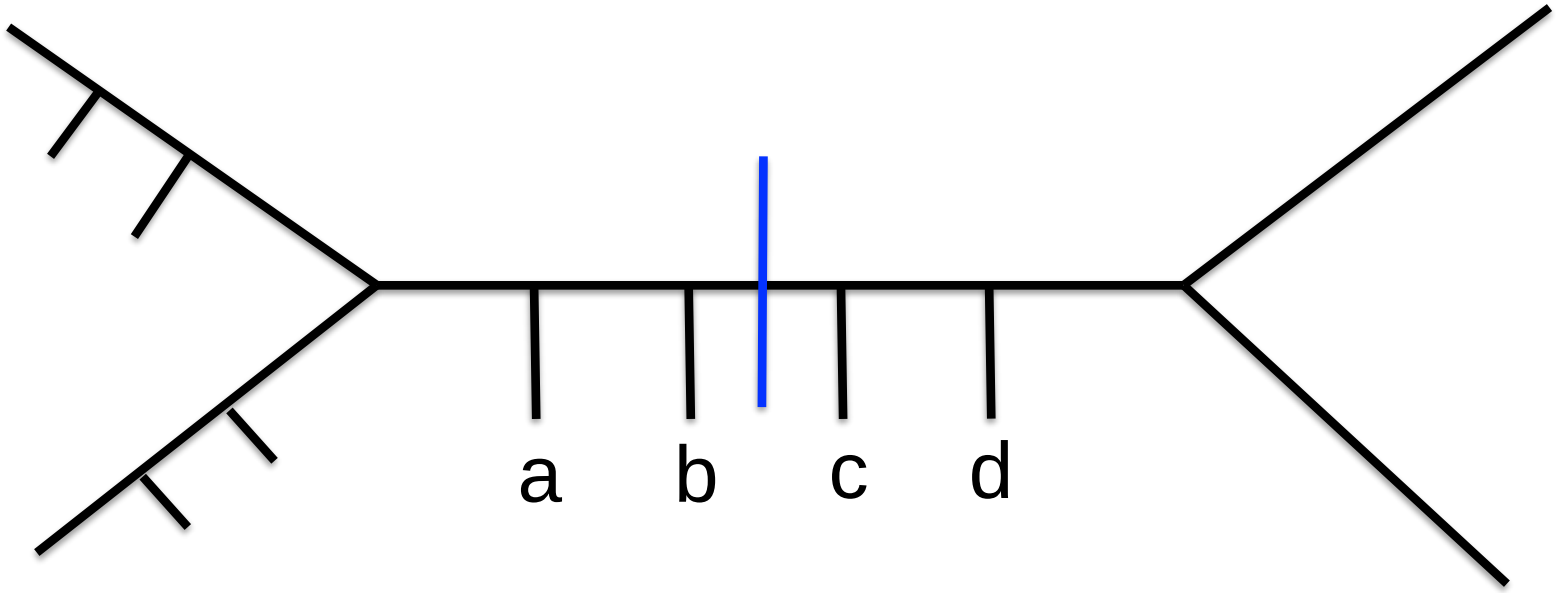
Answer: it is safe to turn a 2|2 side into a 1|3 side when there are "many" leaves on adjacent sides.

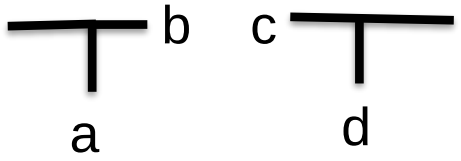


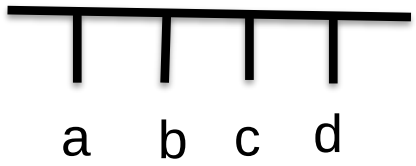




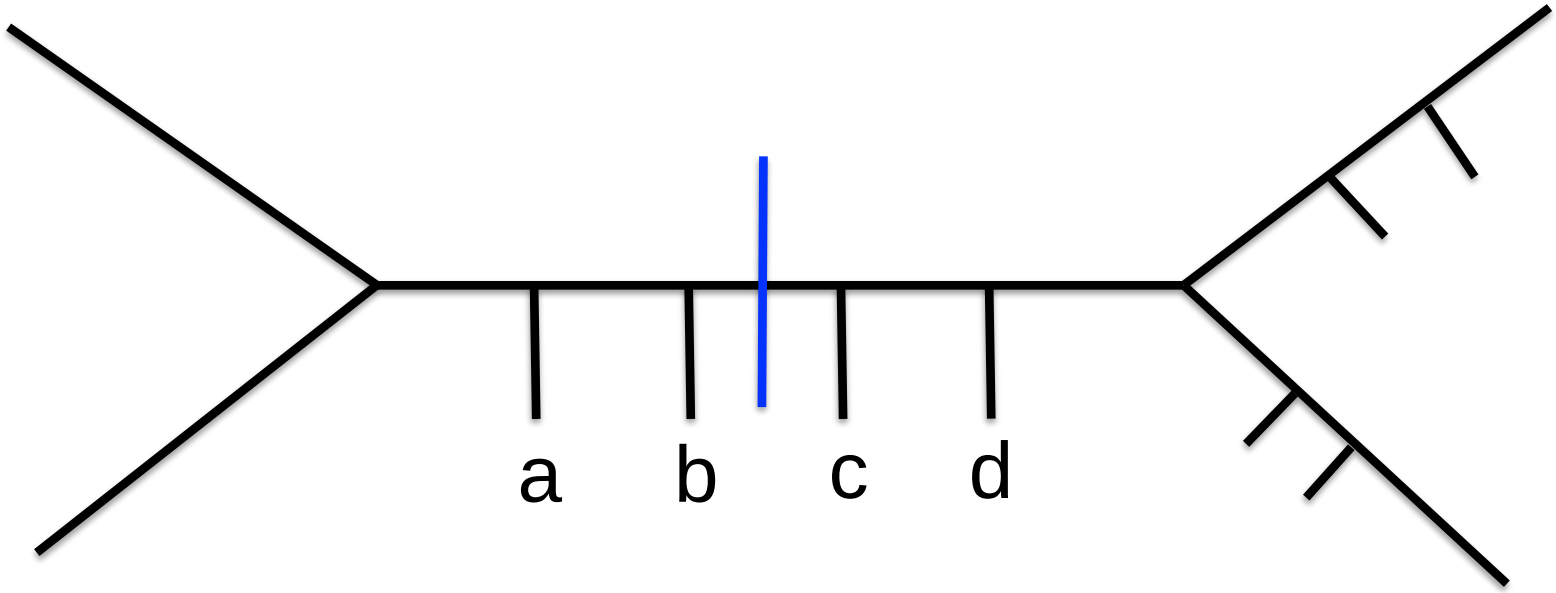
Answer: it is safe to turn a 2|2 side into a 1|3 side when there are "many" leaves on adjacent sides.

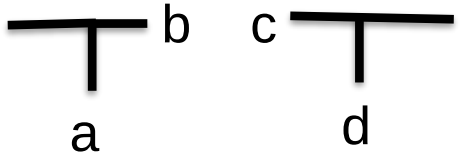


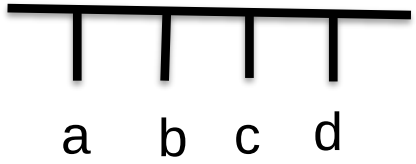




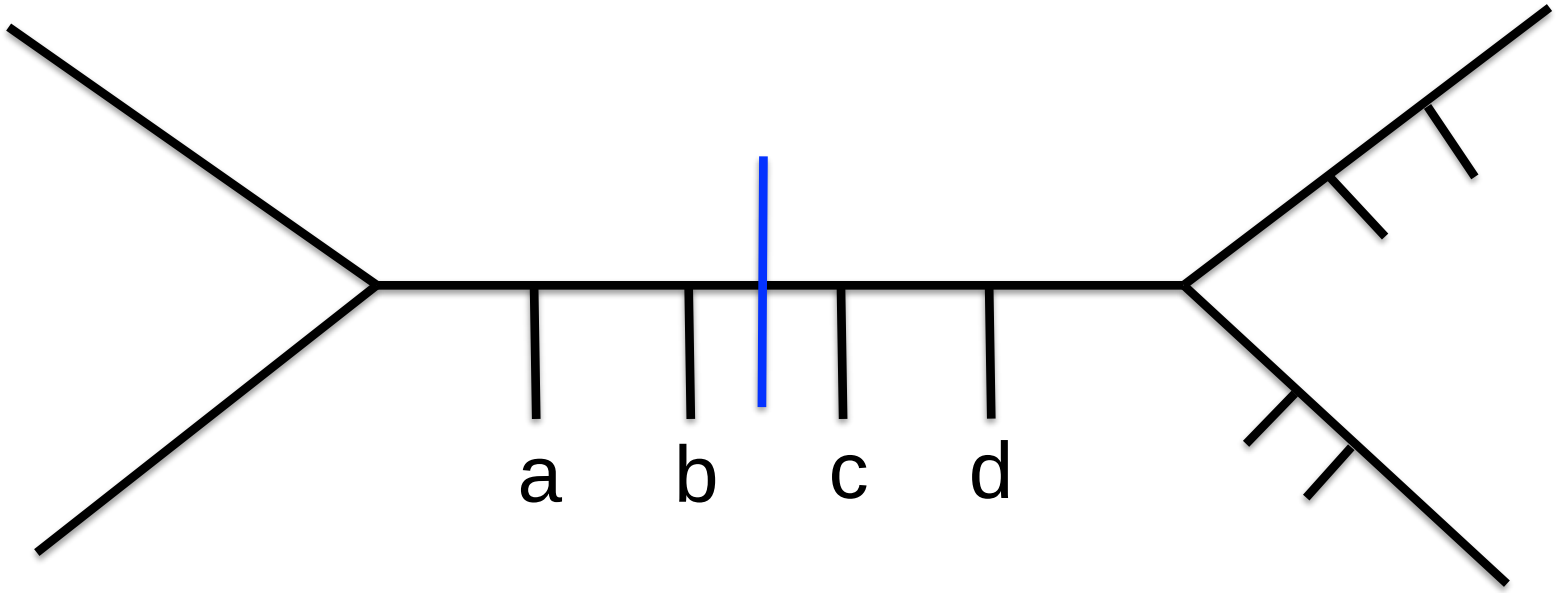
Answer: it is safe to turn a 2|2 side into a 1|3 side when there are "many" leaves on adjacent sides.

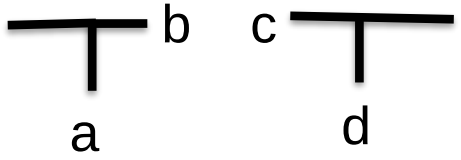


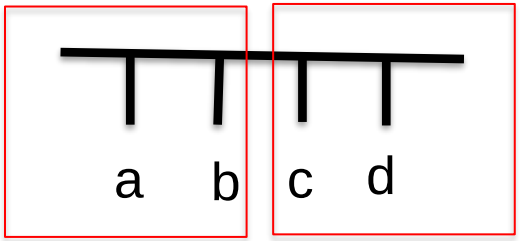




Answer: it is safe to turn a 2|2 side into a 1|3 side when there are "many" leaves on adjacent sides.







Proof uses agreement forests

1|3 and densely flanked 2|2 sides obliterate each other!

Idea… 2|2 sides that have many leaves on adjacent sides (“densely flanked 2|2 sides”) can be turned into 1|3 sides, which can then eat themselves.

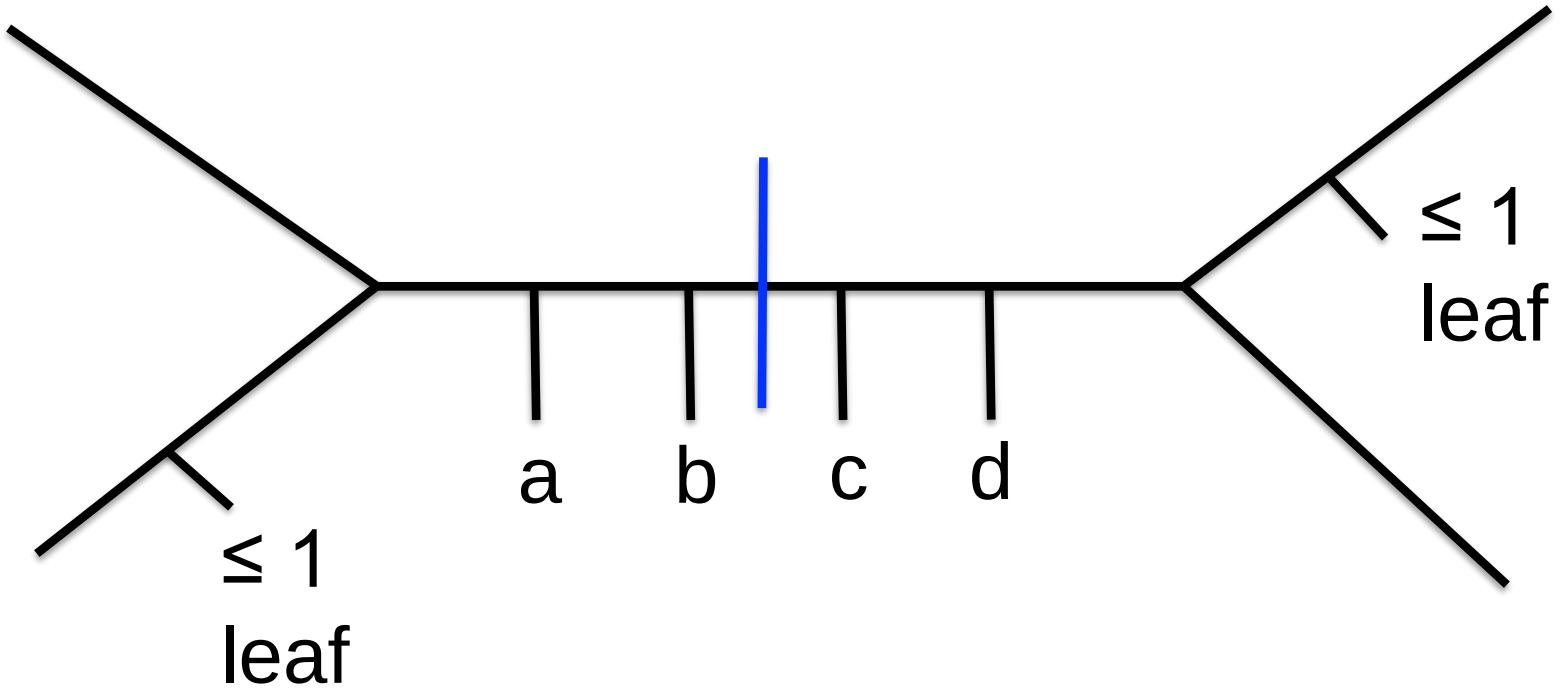
At the point that this process cannot continue anymore, all but 1 of the 1|3 sides and the densely flanked 2|2 sides have gone.

(A similar type of dense-flanking argument can be used to prove that 2|1|1 sides, a third type of bottleneck, can also be destroyed, but I won’t talk about that today.)

Insight… Apart from 1 possible exception, the only surviving sides with 4 leaves are “sparsely flanked” i.e. have relatively few leaves on adjacent sides.

So viewed together they contribute on average fewer than 4 leaves per side.

The only surviving sides
with 4 leaves are sparsely flanked:



Sketch of upper bounding argument

- We have *2k* breakpoints to divide across *3(k-1)* sides.
- We can safely assume there are no sides with 0 or 2 leaves.
- Let p, q, r be the number of sides with 4, 3 or 1 leaves.
- Crucially: all except ≤1 sides with 4 leaves are "sparsely flanked", which means they have at least two adjacent sides with 1 leaf.
- But each side with one leaf can be shared by at most 4 sides with 4 leaves, so $r \geq (2/4)p$.

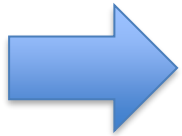
```
Maximize 4p + 3q + 1r + 1
subject to
p+q+r = 3k - 3
p <= 2k
r >= (2/4)p and
p,r,q >= 0 (and integer)
```

```
Maximize 4p + 3q + 1r + 1
subject to
p+q+r = 3k - 3
p <= 2k
r >= (2/4)p and
p,r,q >= 0 (and integer)
```

```
Maximize 4p + 3q + 1r + 1
subject to
p+q+r = 3k - 3
p <= 2k
r >= (2/4)p and
p,r,q >= 0 (and integer)
```

→ *q = (3k-3)-p-r*

Maximize 4p + 3q + 1r + 1
subject to
p+q+r = 3k - 3
p <= 2k
r >= (2/4)p and
p,r,q >= 0 (and integer)



Maximize 9k + p - 2r - 8
subject to
p <= 2k
r >= (1/2)p and
p,r,q >= 0 (and integer).

Maximize 4p + 3q + 1r + 1
subject to
p+q+r = 3k - 3
p <= 2k
r >= (2/4)p and
p,r,q >= 0 (and integer)

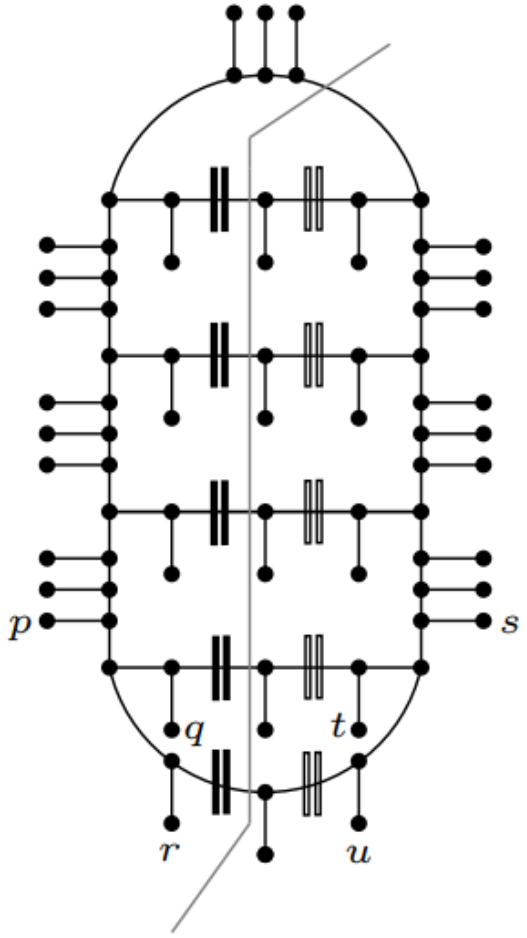
Maximize 9k + p - 2r - 8
subject to
p <= 2k
r >= (1/2)p and
p,r,q >= 0 (and integer).

So the kernel has size at most *9k-8….*

```
Maximize 4p + 3q + 1r + 1
subject to
p+q+r = 3k - 3
p <= 2k
r >= (2/4)p and
p,r,q >= 0 (and integer)
```

```
Maximize 9k + p - 2r - 8
subject to
p <= 2k
r >= (1/2)p and
p,r,q >= 0 (and integer).
```

…and this is essentially tight ☺



This “k-ladder” construction (here k=5) induces two irreducible trees that have d_{TBR}=k and exactly 9k-9 leaves.

Conclusions and future work

We achieved the improvement from 11k-9 to 9k-8 by introducing three new powerful reduction rules.

Can we go below *9k-8* ? Probably, but…

…auxiliary proofs and lemmas are already extremely technical ☹

Can we analytically and/or computationally (semi-)automate the search for new reduction rules, proofs of correctness and bounding arguments to keep proof complexity under control?

Can the new reduction rules be used elsewhere?

Do the new reduction rules have added value in practice? (Probably: the 11k-9 rules already work better in practice than the 15k-9 rules: Wersch, K., Linz, Stamoulis, Annals of Operations Research 2022)

Thank you for listening!

More details at:

- **Deep kernelization for the Tree Bisection and Reconnnect (TBR) distance in phylogenetics,** https://arxiv.org/abs/2206.04451 (K., Linz and Meuwese, 2022)
- **New reduction rules for the tree bisection and reconnection distance** (K. and Linz, Annals of Combinatorics 24(3), 2020)