The complexity of computing nearest neighbour interchange distances between ranked phylogenetic trees

Lena Collienne

Biological Data Science Lab
Department of Computer Science
University of Otago

24/04/2020

Phylogenetic Inference

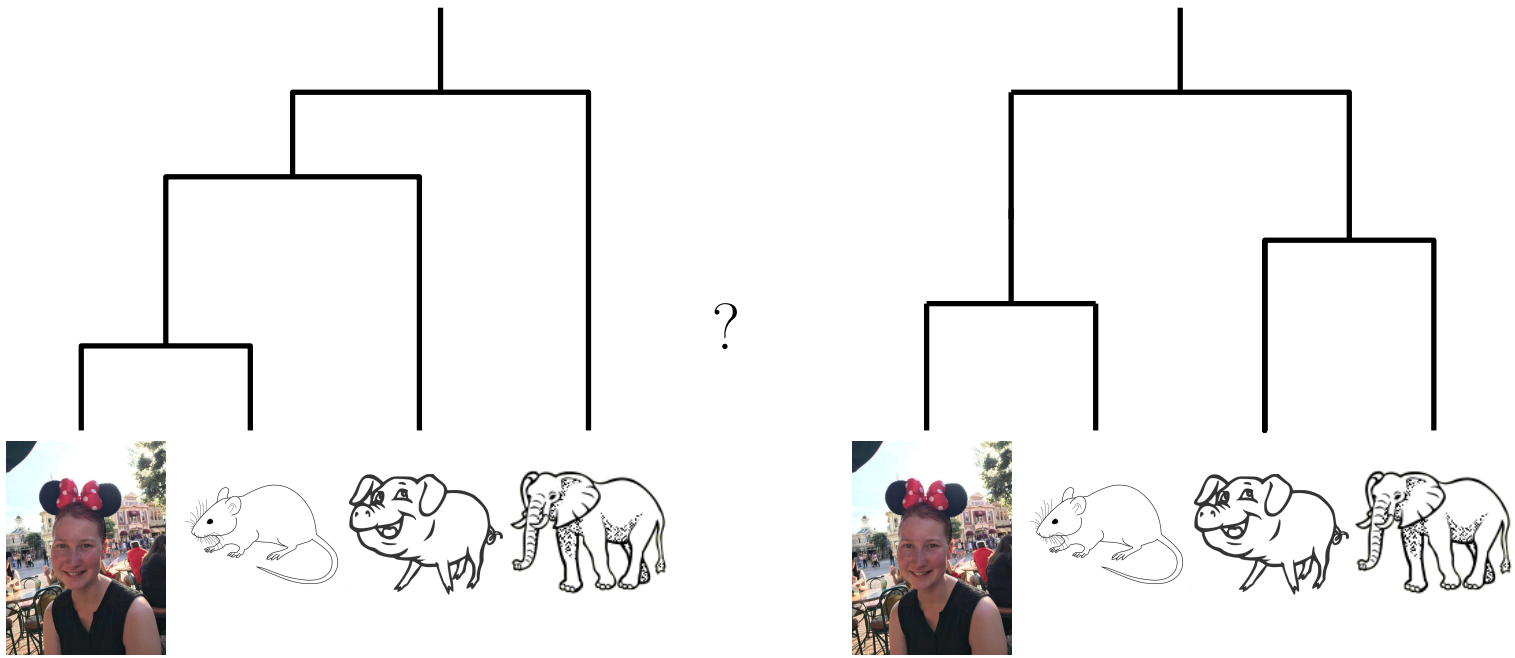
DNA sequences

Mouse	CTCGTATCCCTTGTAACTCCGTCCCACTCCTTTTAT
Elephant	CTCATAGCACTTGTAACTCCGTCCCACGCCTTTTCT
Human	CTCGTATCCCTTGTAACTCCGTCCCACTCCTTTTTT
Pig	CTCCTAGCACTTGTAACTCCGTCCCACCCCTTTTGT

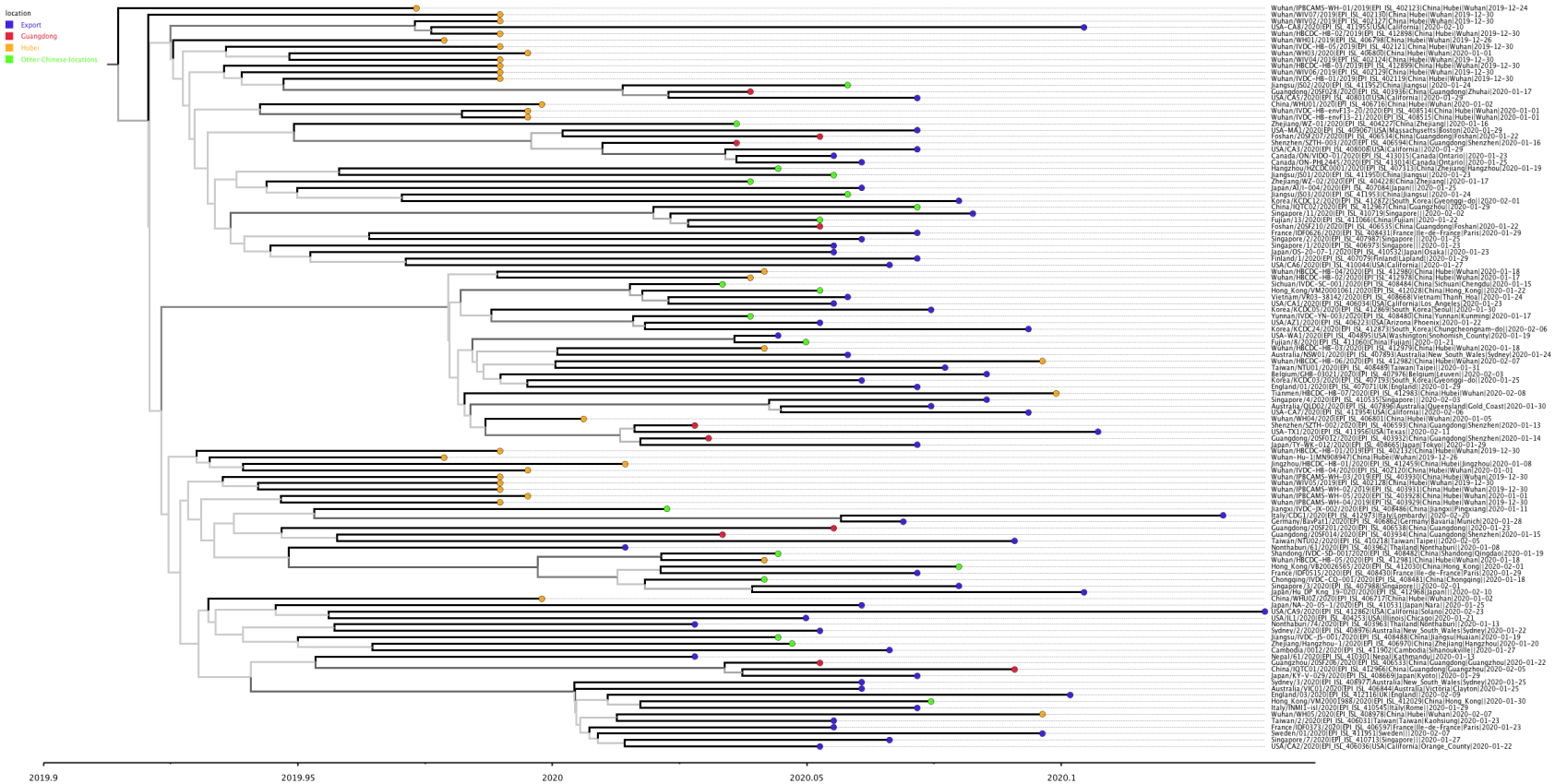
Phylogenetic Inference

DNA sequences

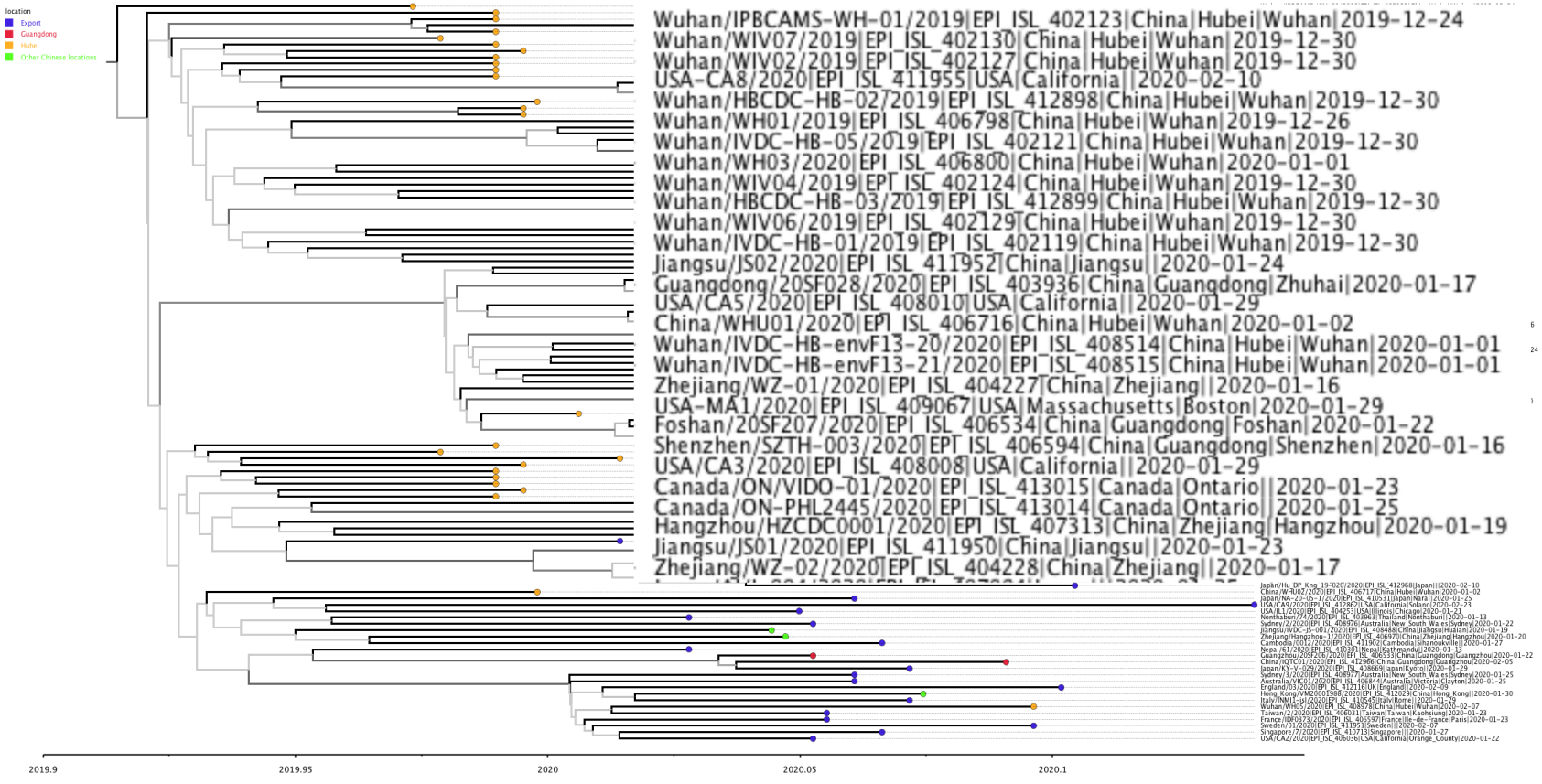
Mouse	CTCGTATCCCTTGTAACTCCGTCCCACTCCTTTTAT
Elephant	CTCATAGCACTTGTAACTCCGTCCCACGCCTTTTCT
Human	CTCGTATCCCTTGTAACTCCGTCCCACTCCTTTTTT
Pig	CTCCTAGCACTTGTAACTCCGTCCCACCCCTTTTGT



Phylogenetic Inference



Phylogenetic Inference



Phylogenetic Inference

Tree search algorithms

MEGA4: molecular evolutionary genetics analysis (**MEGA**) **software** version 4.0
K Tamura, J Dudley, M Nei… - Molecular biology and …, 2007 - academic.oup.com
We announce the release of the fourth version of **MEGA software**, which expands on the existing facilities for editing DNA sequence data from autosequencers, mining Web-databases, performing automatic and manual sequence alignment, analyzing sequence …
☆ 99 Cited by 32728 Related articles All 19 versions

RAxML-VI-HPC: maximum likelihood-based phylogenetic analyses with thousands of taxa and mixed models
A Stamatakis - Bioinformatics, 2006 - academic.oup.com
RAxML-VI-HPC (randomized axelerated maximum likelihood for high performance computing) is a sequential and parallel program for inference of large phylogenies with maximum likelihood (ML). Low-level technical optimizations, a modification of the search …
☆ 99 Cited by 13839 Related articles All 29 versions

New algorithms and methods to estimate maximum-likelihood phylogenies: assessing the performance of **PhyML** 3.0
S Guindon, JF Dufayard, V Lefort… - Systematic …, 2010 - academic.oup.com
PhyML is a phylogeny software based on the maximum-likelihood principle. Early **PhyML** versions used a fast algorithm performing nearest neighbor interchanges to improve a reasonable starting tree topology. Since the original publication (Guindon S., Gascuel O …
☆ 99 Cited by 10154 Related articles All 30 versions

[PDF] **MRBAYES**: Bayesian inference of phylogenetic trees
JP Huelsenbeck, F Ronquist - Bioinformatics, 2001 - 146.6.100.192
The program **MRBAYES** performs Bayesian inference of phylogeny using a variant of Markov chain Monte Carlo. Availability: **MRBAYES**, including the source code, documentation, sample data files, and an executable, is available at http://brahms. biology …
☆ 99 Cited by 21336 Related articles All 21 versions

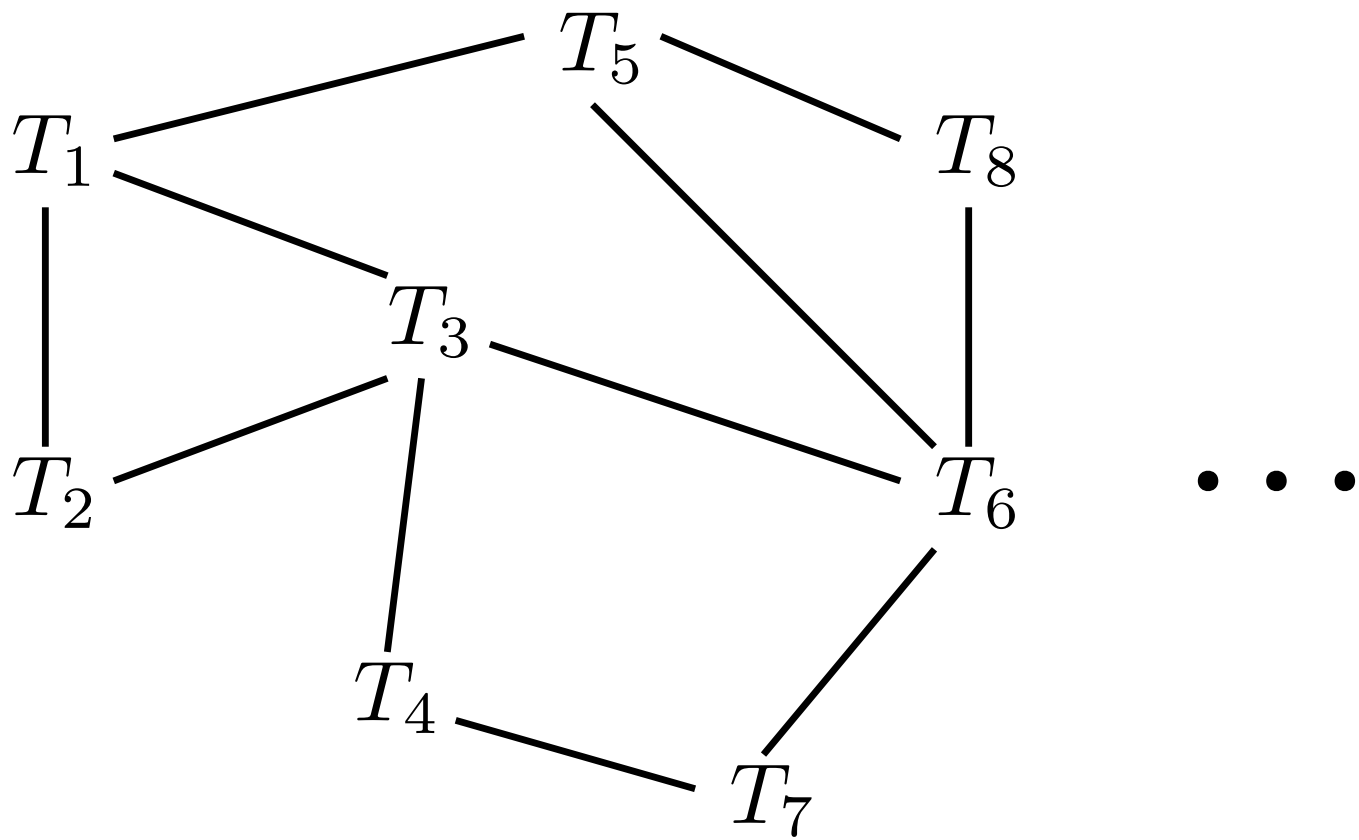
MrBayes 3: Bayesian phylogenetic inference under mixed models
F Ronquist, JP Huelsenbeck - Bioinformatics, 2003 - academic.oup.com
MrBayes 3 performs Bayesian phylogenetic analysis combining information from different data partitions or subsets evolving under different stochastic evolutionary models. This allows the user to analyze heterogeneous data sets consisting of different data types—eg …
☆ 99 Cited by 26317 Related articles All 30 versions

[HTML] **BEAST**: Bayesian evolutionary analysis by sampling trees
AJ Drummond, A Rambaut - BMC …, 2007 - bmcevolbiol.biomedcentral.com
The evolutionary analysis of molecular sequence variation is a statistical enterprise. This is reflected in the increased use of probabilistic models for phylogenetic inference, multiple sequence alignment, and molecular population genetics. Here we present **BEAST**: a fast …
☆ 99 Cited by 11063 Related articles All 26 versions »

Bayesian phylogenetics with BEAUti and the **BEAST** 1.7
AJ Drummond, MA Suchard, D Xie… - Molecular biology and …, 2012 - academic.oup.com
Computational evolutionary biology, statistical phylogenetics and coalescent-based population genetics are becoming increasingly central to the analysis and understanding of molecular sequence data. We present the Bayesian Evolutionary Analysis by Sampling …
☆ 99 Cited by 7681 Related articles All 14 versions

Phylogenetic Inference

Tree search algorithms

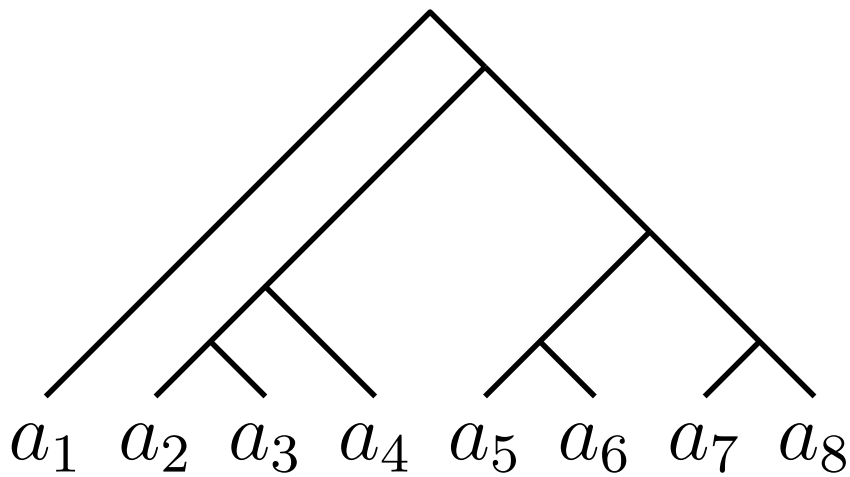


Phylogenetic trees

Problem: There are $(2n-3)!!$ trees on n leaves

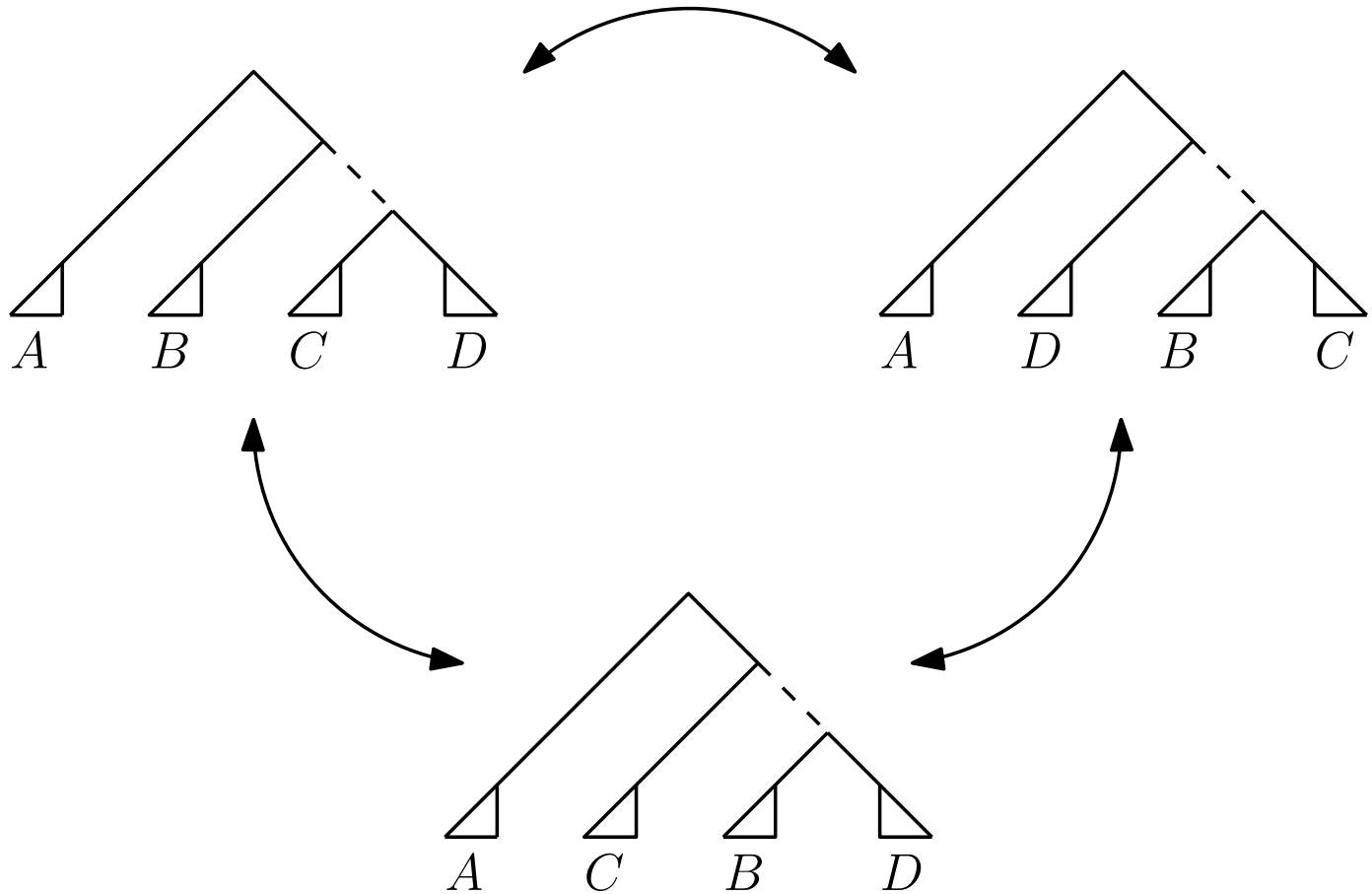
n	Number of trees
4	15
5	105
6	945
7	10395
	...
50	$2.752921 \cdot 10^{76}$

Phylogenetic trees



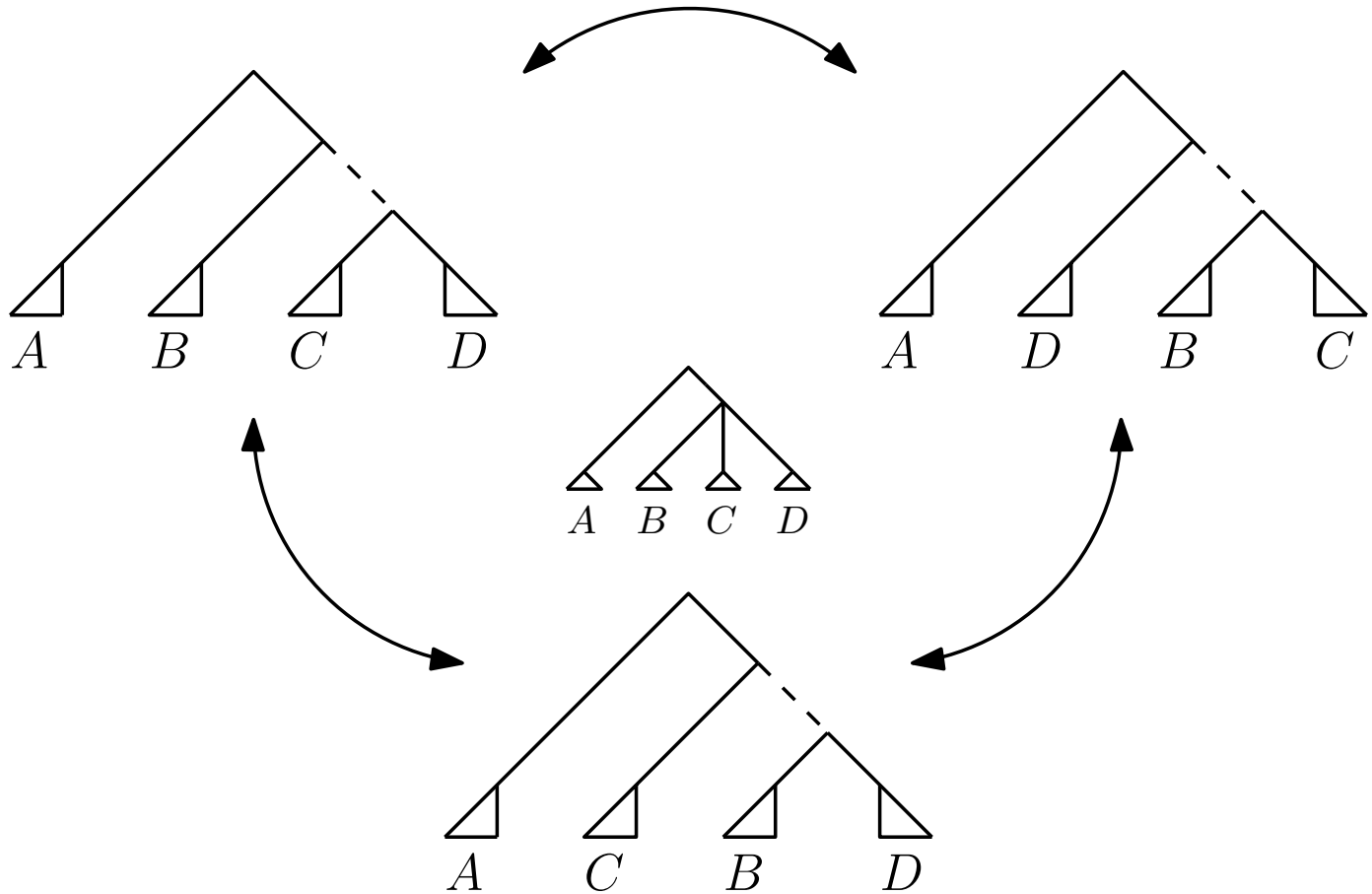
NNI – Nearest Neighbour Interchange

Definition 1



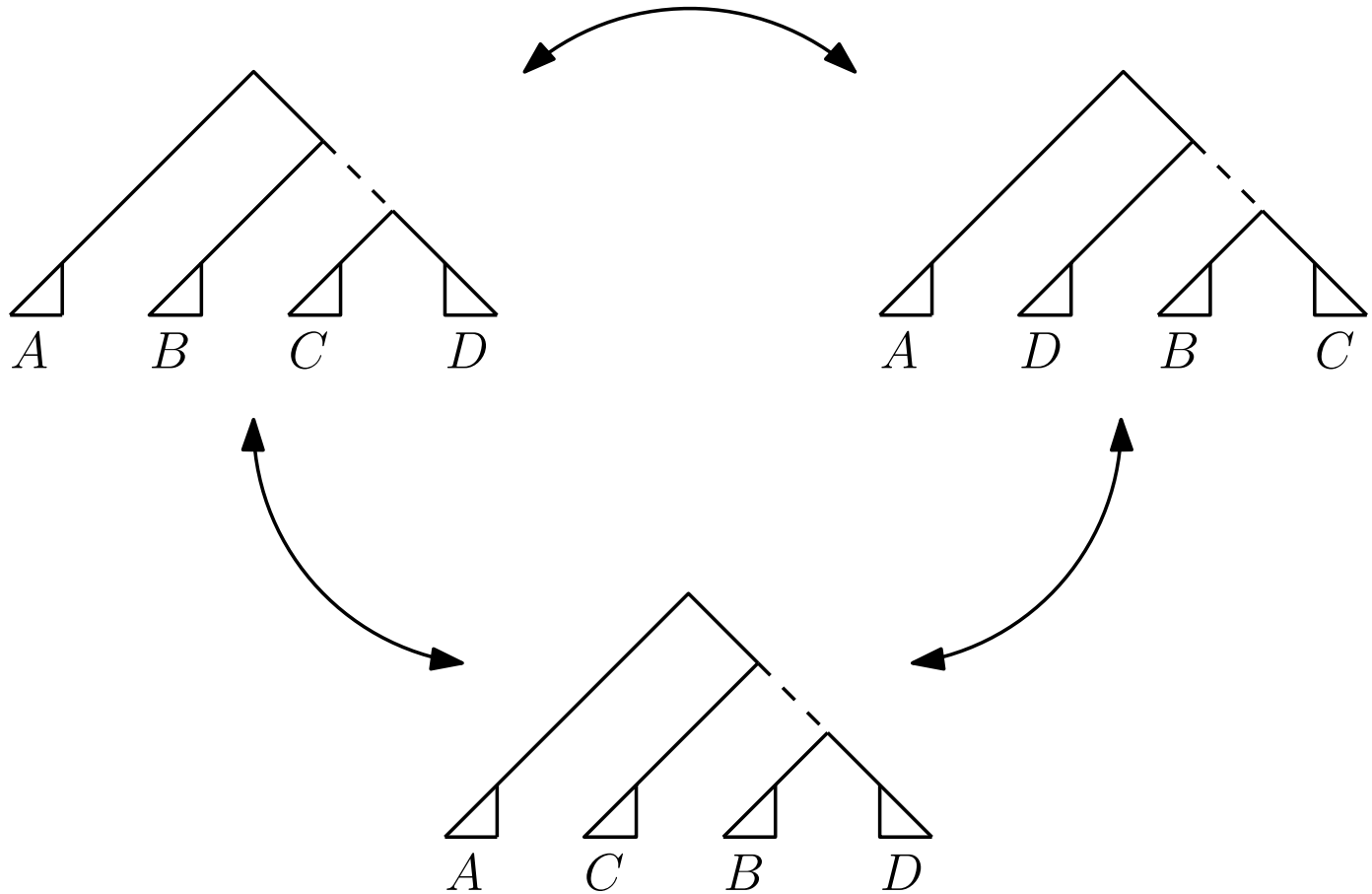
NNI – Nearest Neighbour Interchange

Definition 1



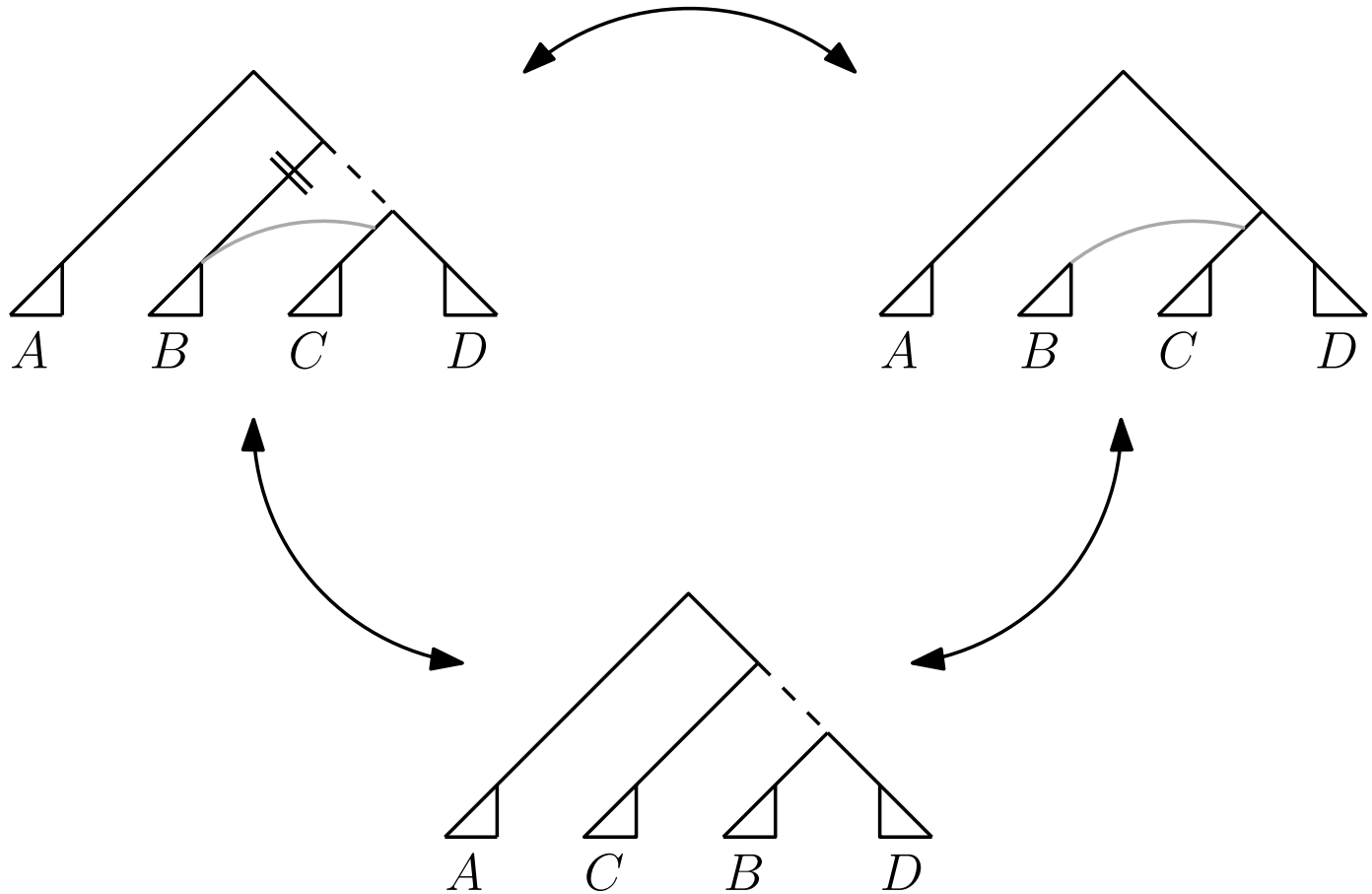
NNI – Nearest Neighbour Interchange

Definition 2



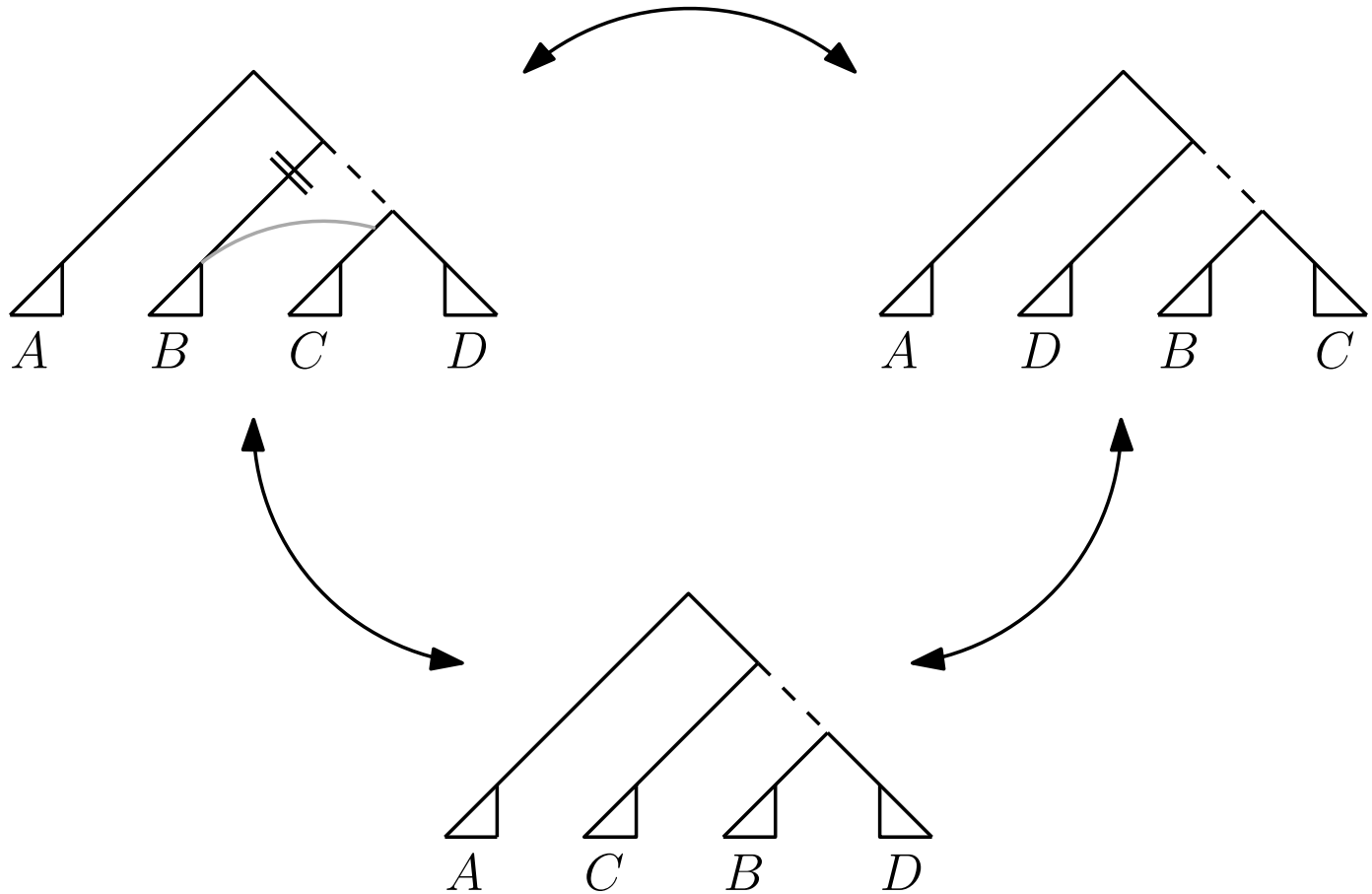
NNI – Nearest Neighbour Interchange

Definition 2



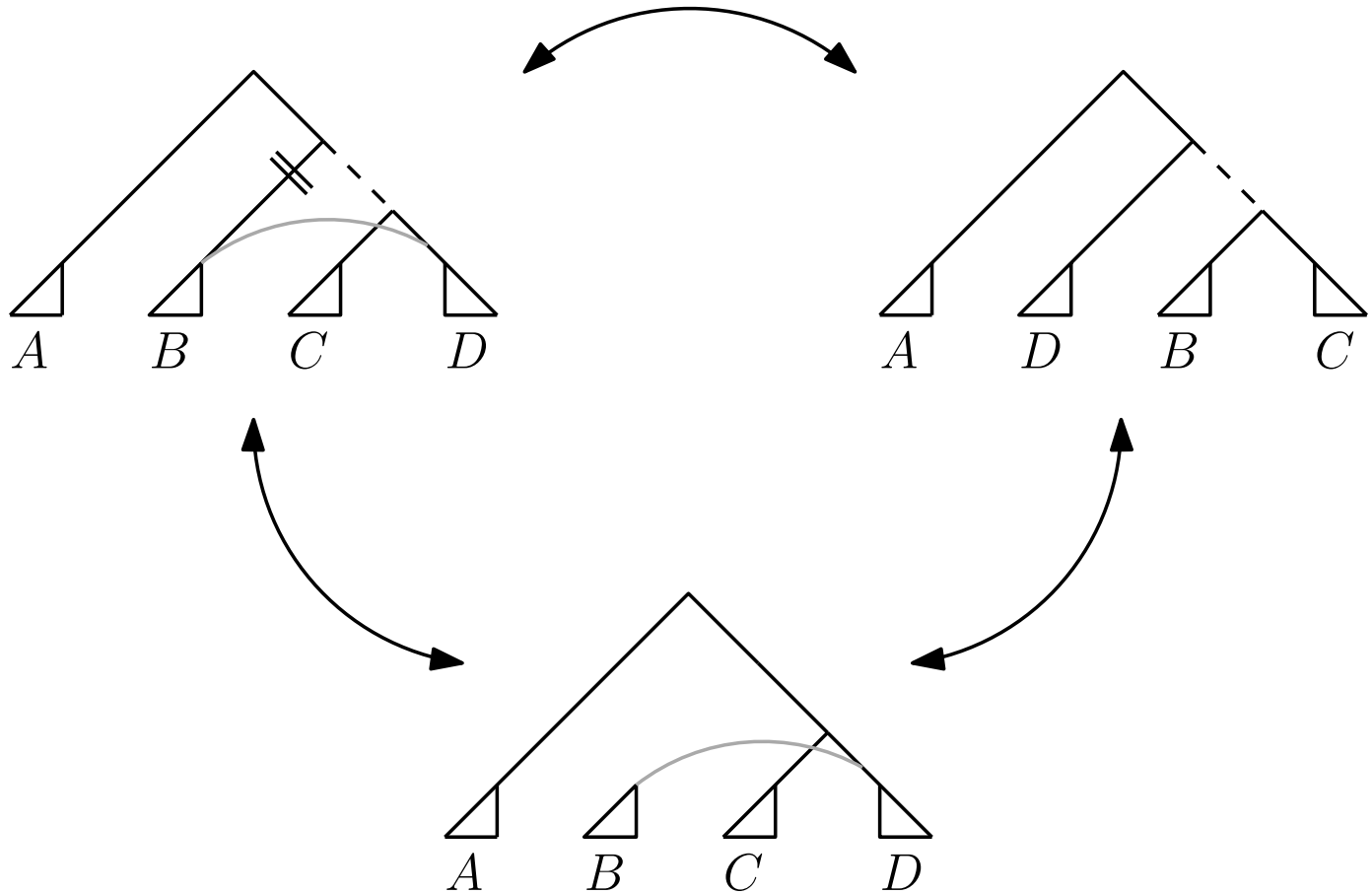
NNI – Nearest Neighbour Interchange

Definition 2



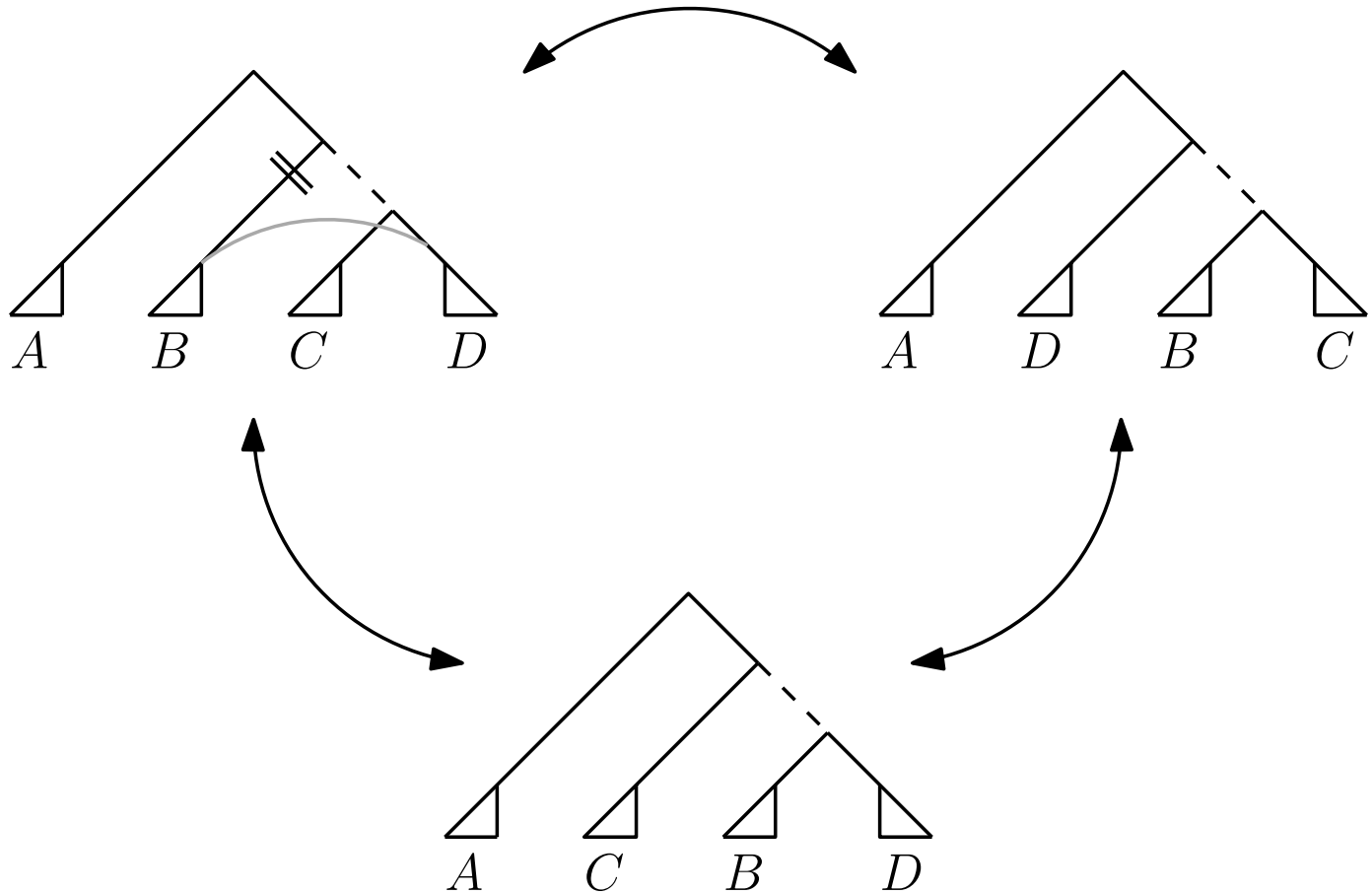
NNI – Nearest Neighbour Interchange

Definition 2



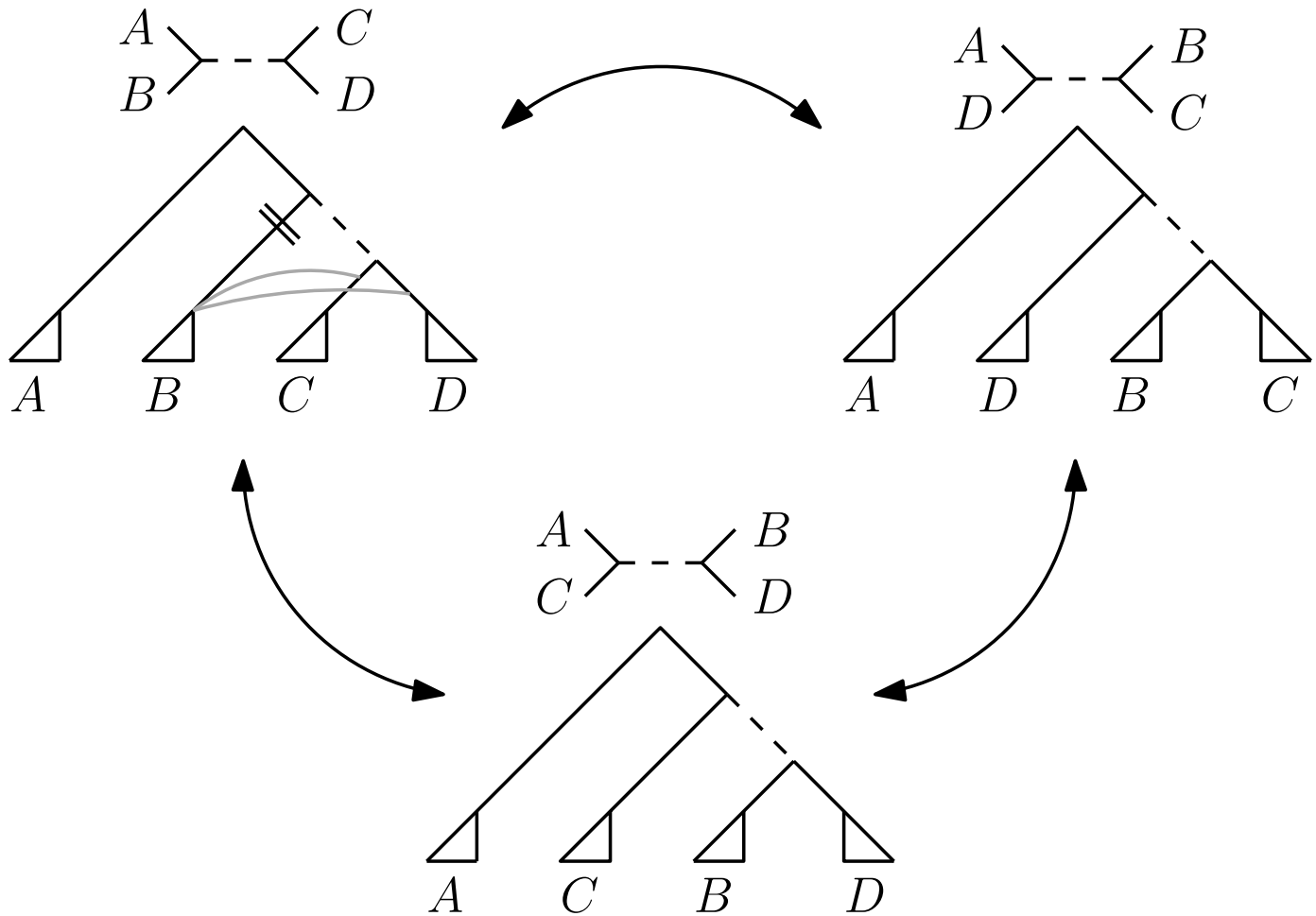
NNI – Nearest Neighbour Interchange

Definition 2

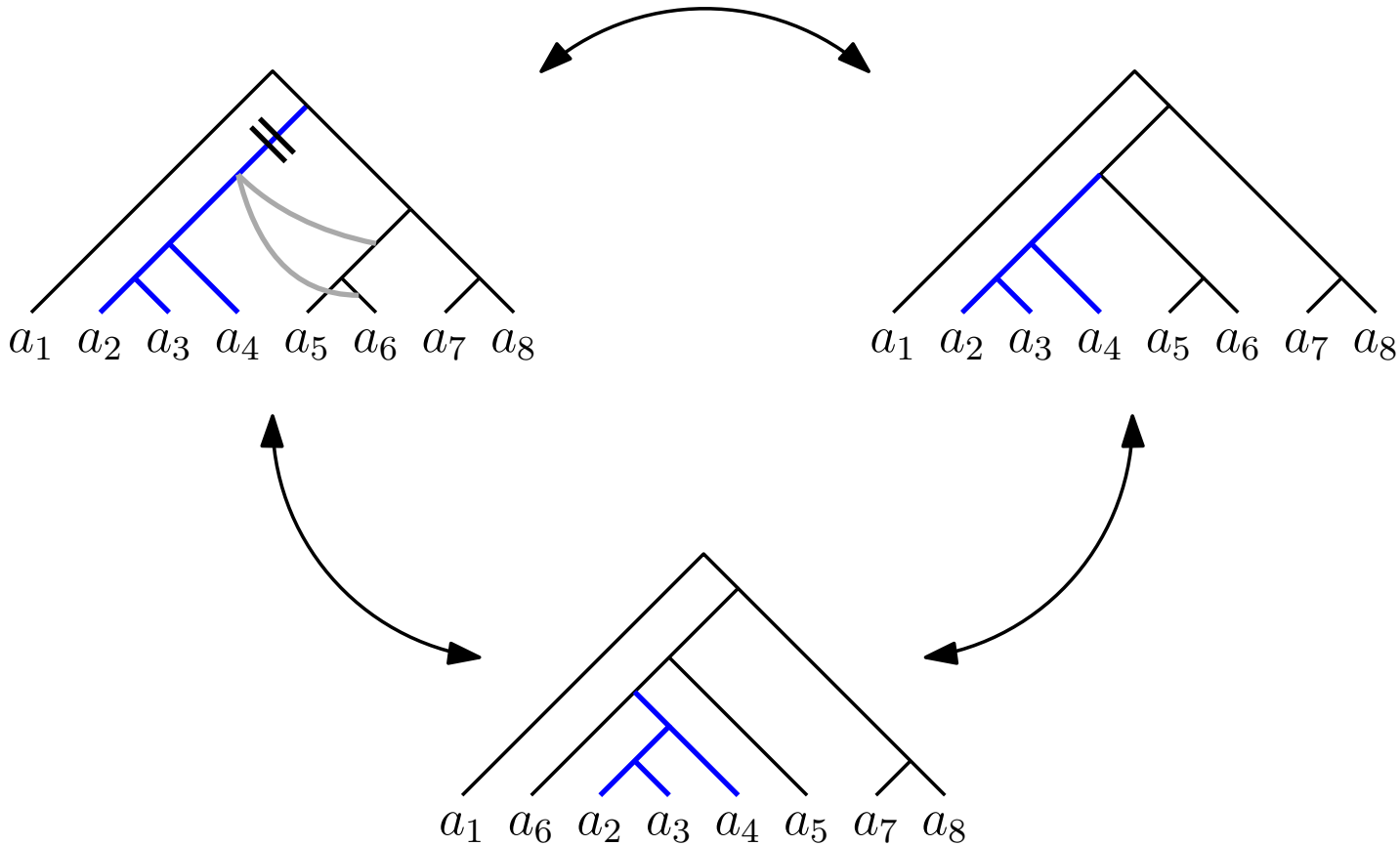


NNI – Nearest Neighbour Interchange

Definition 2



SPR - Subtree Prune and Regraft



Complexity

NNI

NNI-DIST:

INSTANCE: A pair of trees T and R
FIND: Distance between T and R in NNI

Complexity

NNI

NNI-DIST:

INSTANCE: A pair of trees T and R
FIND: Distance between T and R in NNI

- $\mathcal{NP}$-hard

Complexity

NNI

NNI-DIST:

INSTANCE: A pair of trees T and R
FIND: Distance between T and R in NNI

- $\mathcal{NP}$-hard
- BUT: fixed-parameter tractable (FPT)

Complexity

NNI

NNI-DIST:

INSTANCE: A pair of trees T and R
FIND: Distance between T and R in NNI

- $\mathcal{NP}$-hard
- BUT: fixed-parameter tractable (FPT)

FPT:

Parameter k such that problem is solvable in $\mathcal{O}(f(k) * n^{\mathcal{O}(1)})$
$\Rightarrow$ efficiently solvable for small k

Complexity

NNI

NNI-DIST:

INSTANCE: A pair of trees T and R
FIND: Distance between T and R in NNI

- $\mathcal{NP}$-hard
- BUT: fixed-parameter tractable (FPT):

  distance computable in $\mathcal{O}(2^{\frac{21k}{2}} * n)$ where $d(T, R) \leq k$

Complexity

NNI

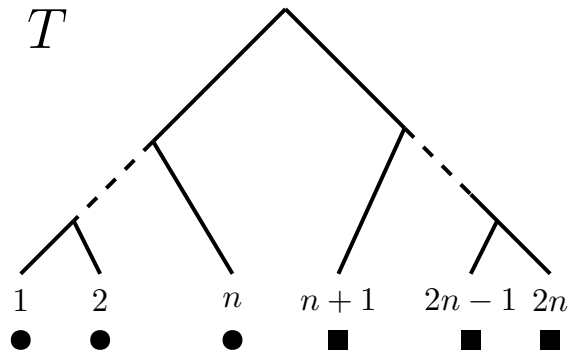
NNI-DIST:

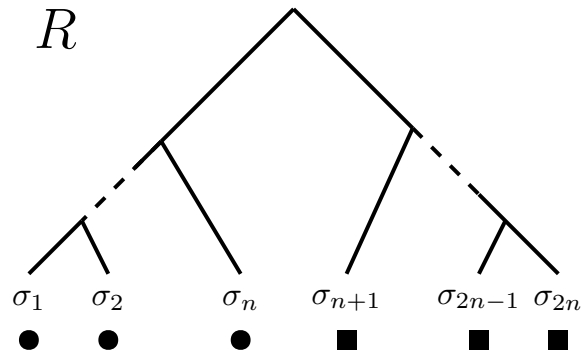
INSTANCE: A pair of trees T and R
FIND: Distance between T and R in NNI

- $\mathcal{NP}$-hard
- BUT: fixed-parameter tractable (FPT):
- Approximation algorithm: ratio $\mathcal{O}(log(n))$

Complexity

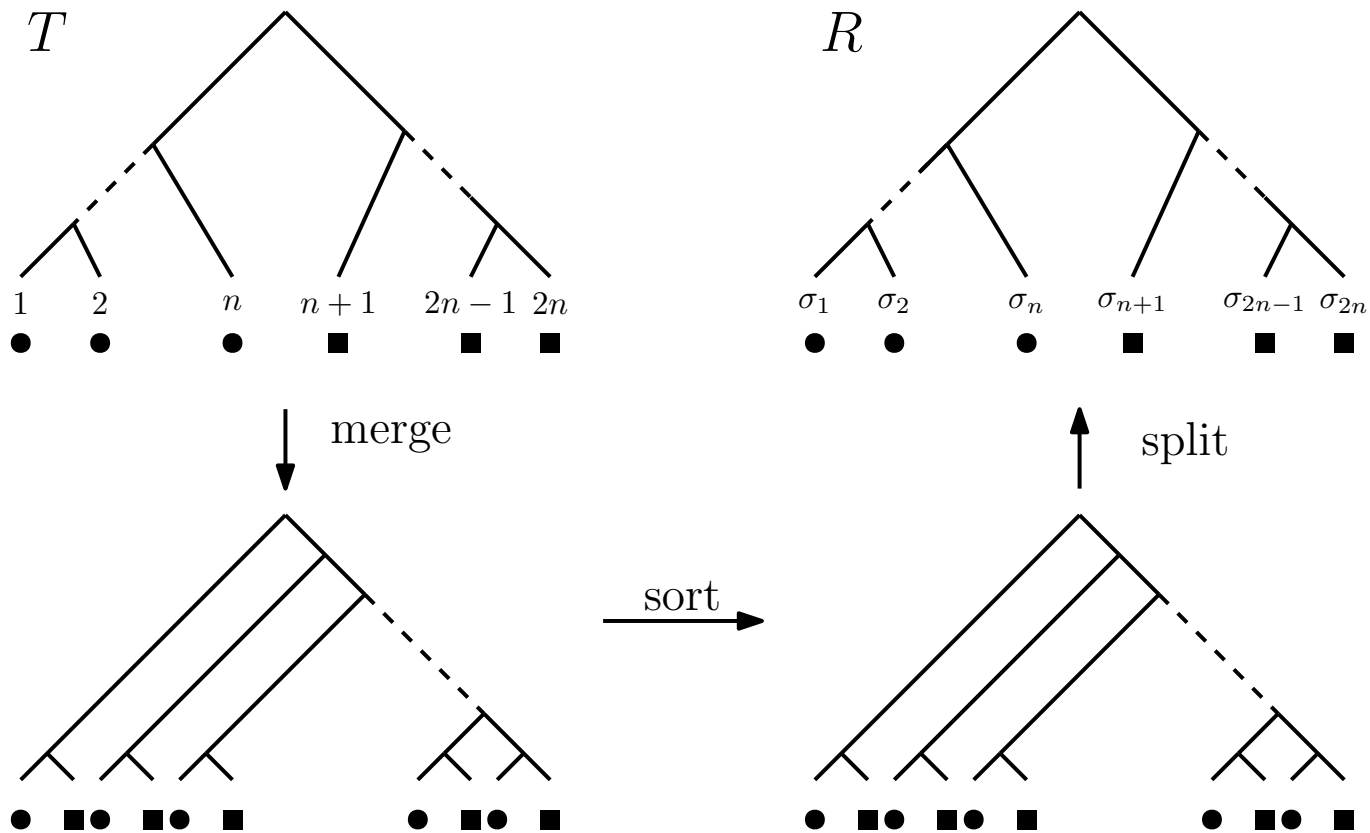
NNI - Cluster Property





Complexity

NNI - Cluster Property



Complexity

SPR

SPR-DIST:

INSTANCE: A pair of trees T and R
FIND: Distance between T and R in SPR

Complexity

SPR

SPR-DIST:

INSTANCE: A pair of trees T and R
FIND: Distance between T and R in SPR

- $\mathcal{NP}$-hard

Complexity

SPR

SPR-DIST:

INSTANCE: A pair of trees T and R
FIND: Distance between T and R in SPR

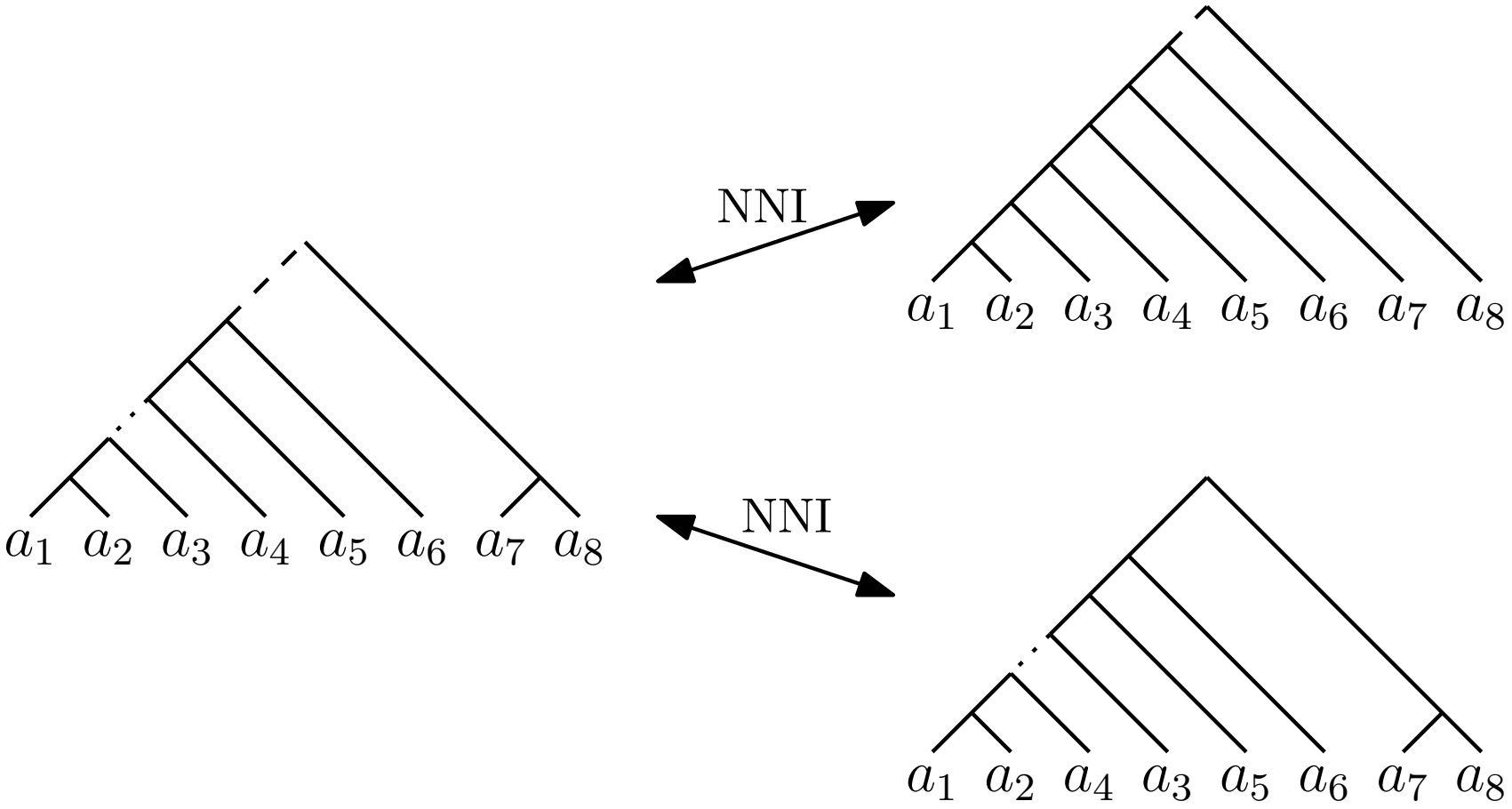
- $\mathcal{NP}$-hard
- BUT: fixed-parameter tractable (FPT)

  distance computable in $\mathcal{O}(2.42^k * k + n^3)$ where $d(T, R) \leq k$

Complexity

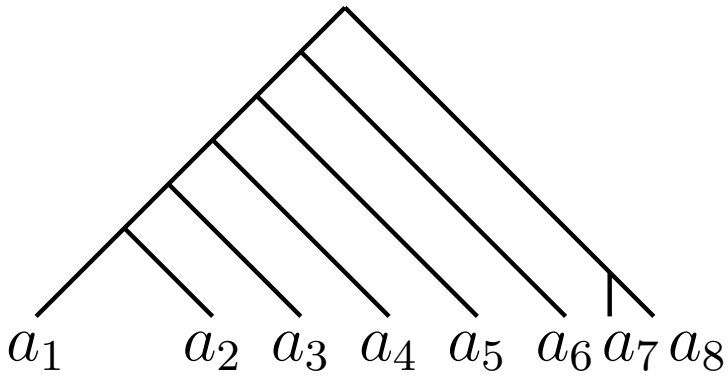
Is there a different parameter that makes $\mathrm{NNI\text{-}DIST}$ easier?

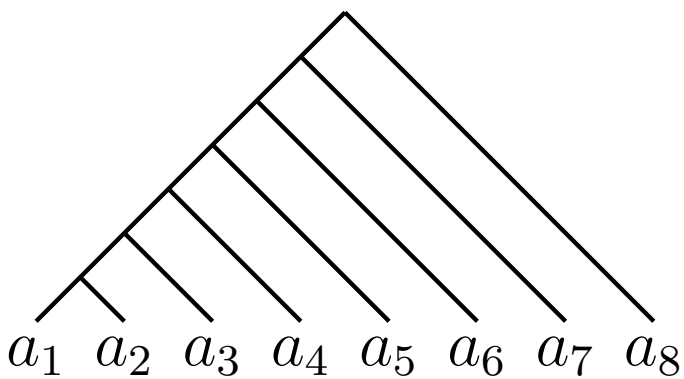
Parameterising NNI



Parameterising NNI

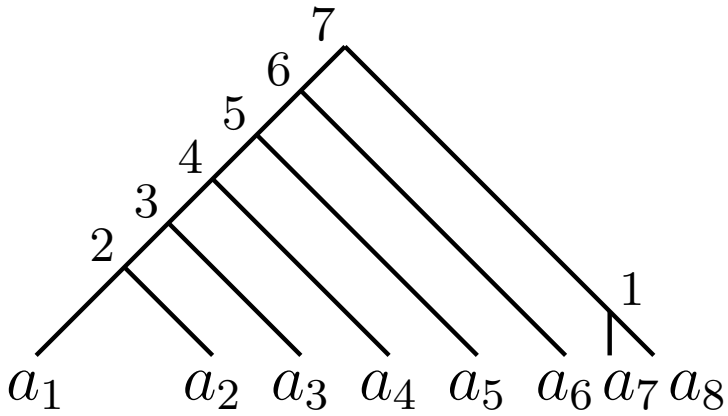
Ranked trees

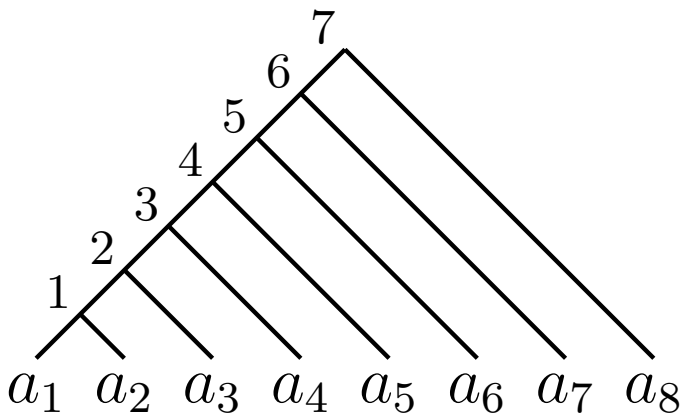




Parameterising NNI

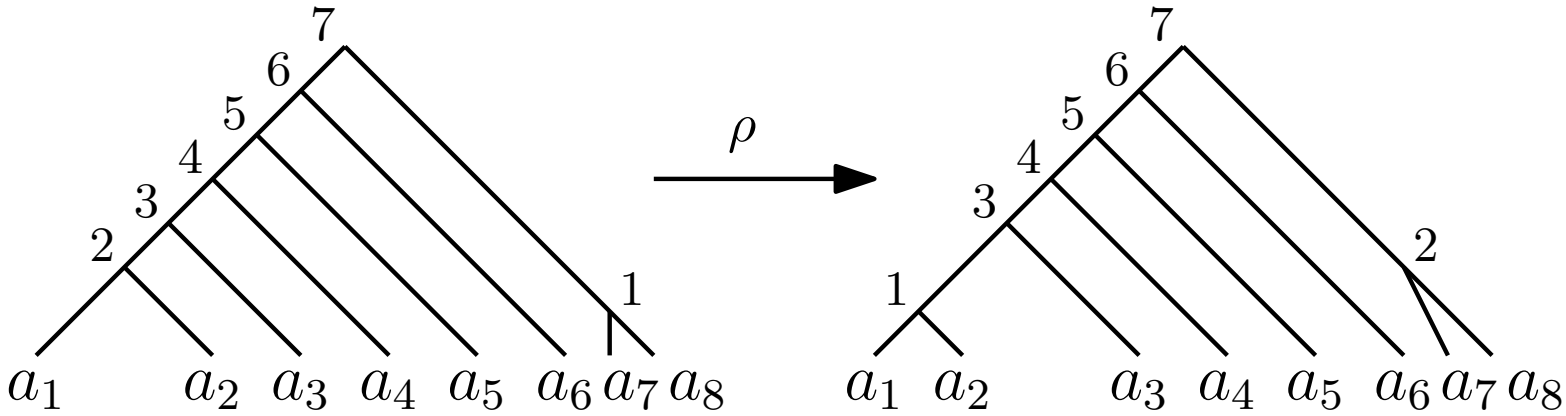
Ranked trees

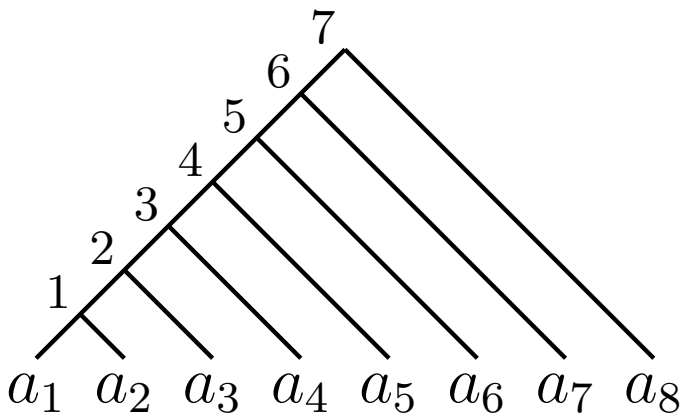




Parameterising NNI

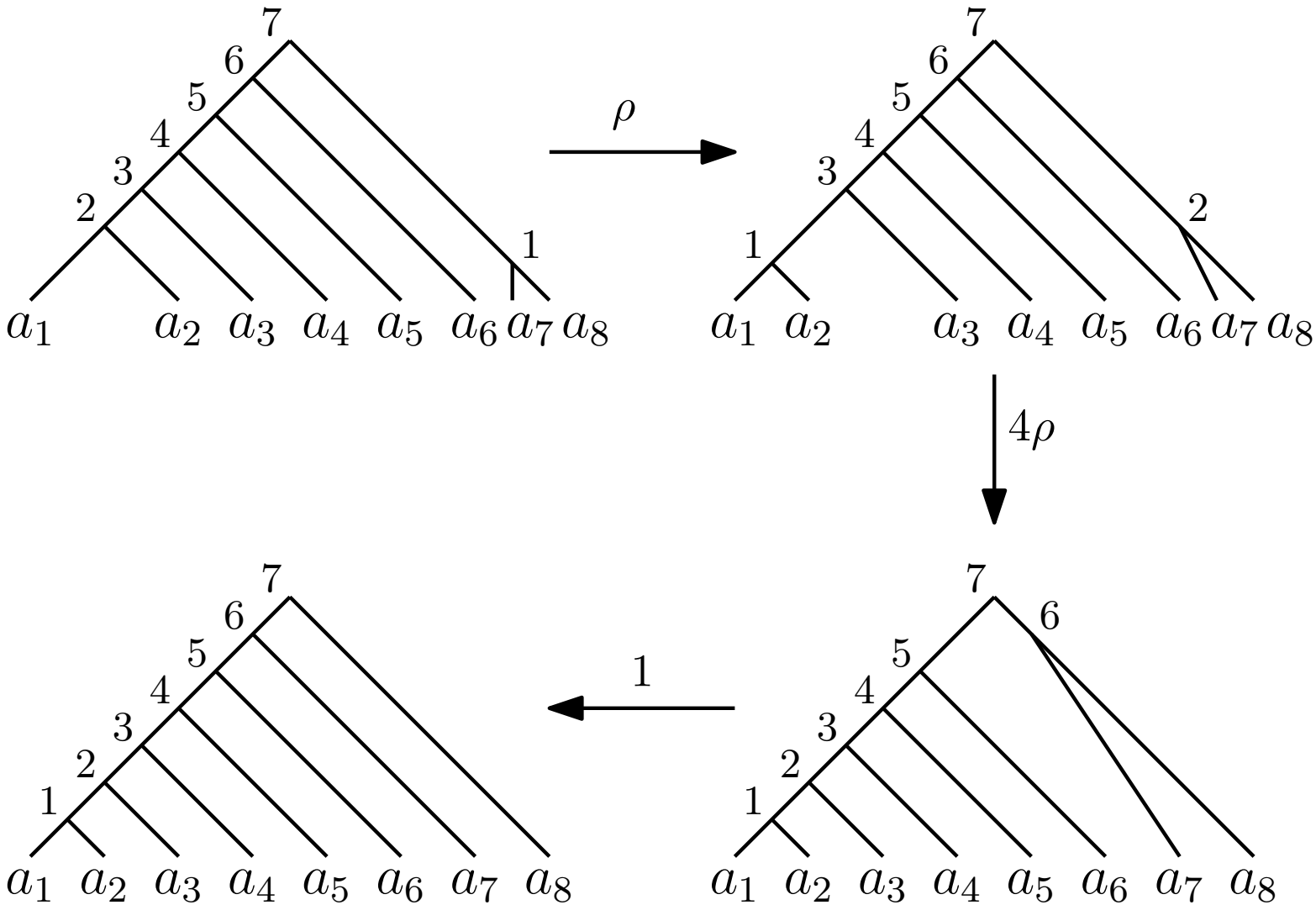
Ranked trees





Parameterising NNI – RNNI(ρ)

Ranked trees



Parameterising NNI – RNNI(ρ)

Complexity

RNNI(ρ)-SP:

INSTANCE: A pair of trees T and R
FIND: A path of minimal weight between T and R in RNNI(ρ)

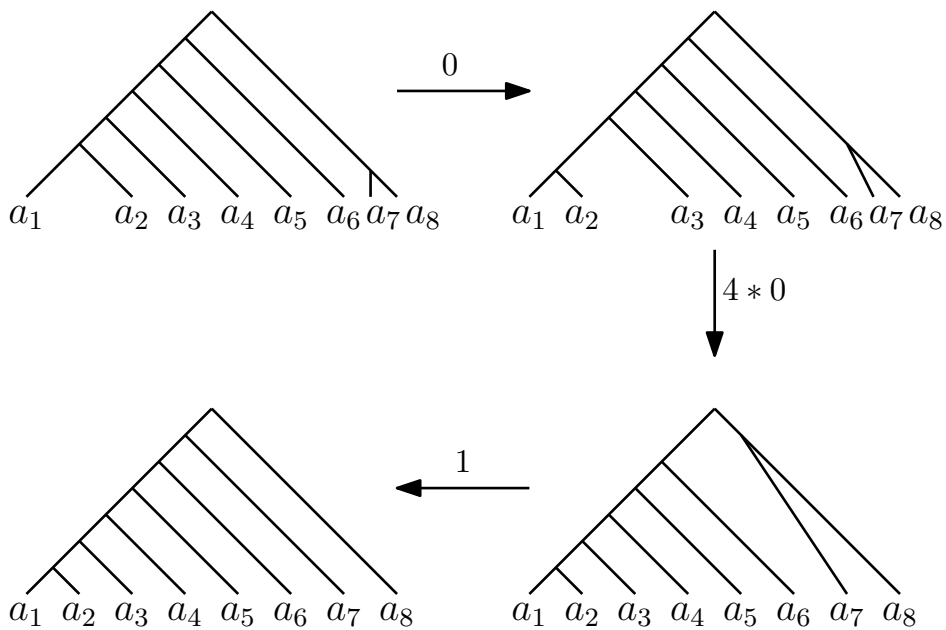
Parameterising NNI – RNNI(ρ)

Complexity

RNNI(ρ)-SP:

INSTANCE: A pair of trees T and R
FIND: A path of minimal weight between T and R in RNNI(ρ)

- RNNI(0)-SP is $\mathcal{NP}$-hard



Parameterising NNI – RNNI(ρ)

Complexity

RNNI(ρ)-SP:

INSTANCE: A pair of trees T and R
FIND: A path of minimal weight between T and R in RNNI(ρ)

- RNNI(0)-SP is $\mathcal{NP}$-hard
- RNNI(ρ)-SP is $\mathcal{NP}$-hard for $0 < \rho < \frac{1}{\Delta(\mathrm{RNNI})}$

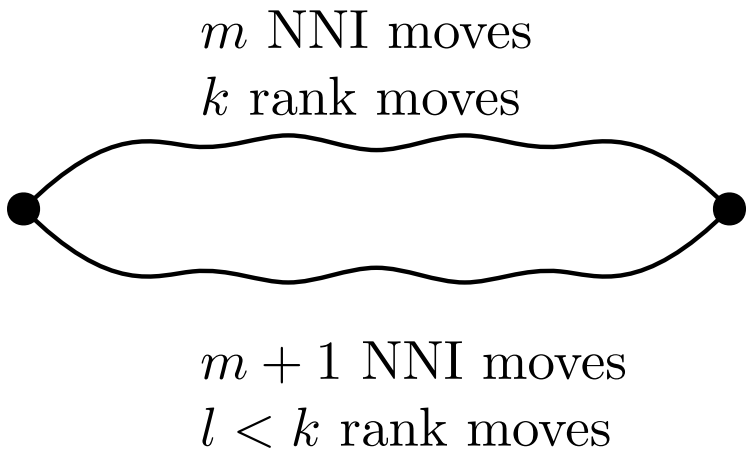
Parameterising NNI – RNNI(ρ)

Complexity

RNNI(ρ)-SP:

INSTANCE: A pair of trees T and R
FIND: A path of minimal weight between T and R in RNNI(ρ)

- RNNI(0)-SP is $\mathcal{NP}$-hard
- RNNI(ρ)-SP is $\mathcal{NP}$-hard for $0 < \rho < \frac{1}{\Delta(\mathrm{RNNI})}$



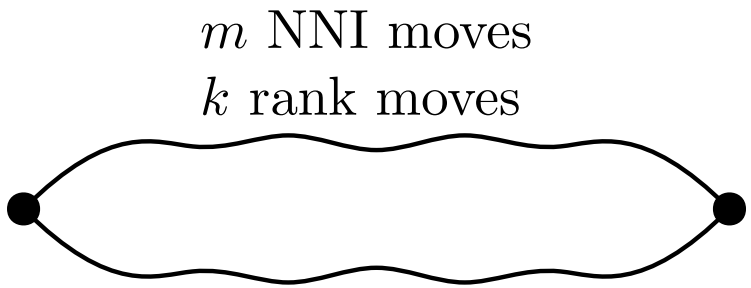
Parameterising NNI – RNNI(ρ)

Complexity

RNNI(ρ)-SP:

INSTANCE: A pair of trees T and R
FIND: A path of minimal weight between T and R in RNNI(ρ)

- RNNI(0)-SP is $\mathcal{NP}$-hard
- RNNI(ρ)-SP is $\mathcal{NP}$-hard for $0 < \rho < \frac{1}{\Delta(\mathrm{RNNI})}$



$\rho * (k - l) < 1$

Parameterising NNI – RNNI(ρ)

Complexity

RNNI(ρ)-SP:

INSTANCE: A pair of trees T and R
FIND: A path of minimal weight between T and R in RNNI(ρ)

- RNNI(0)-SP is $\mathcal{NP}$-hard
- RNNI(ρ)-SP is $\mathcal{NP}$-hard for small $\rho > 0$
- RNNI(1)-SP is

Parameterising NNI – RNNI(ρ)

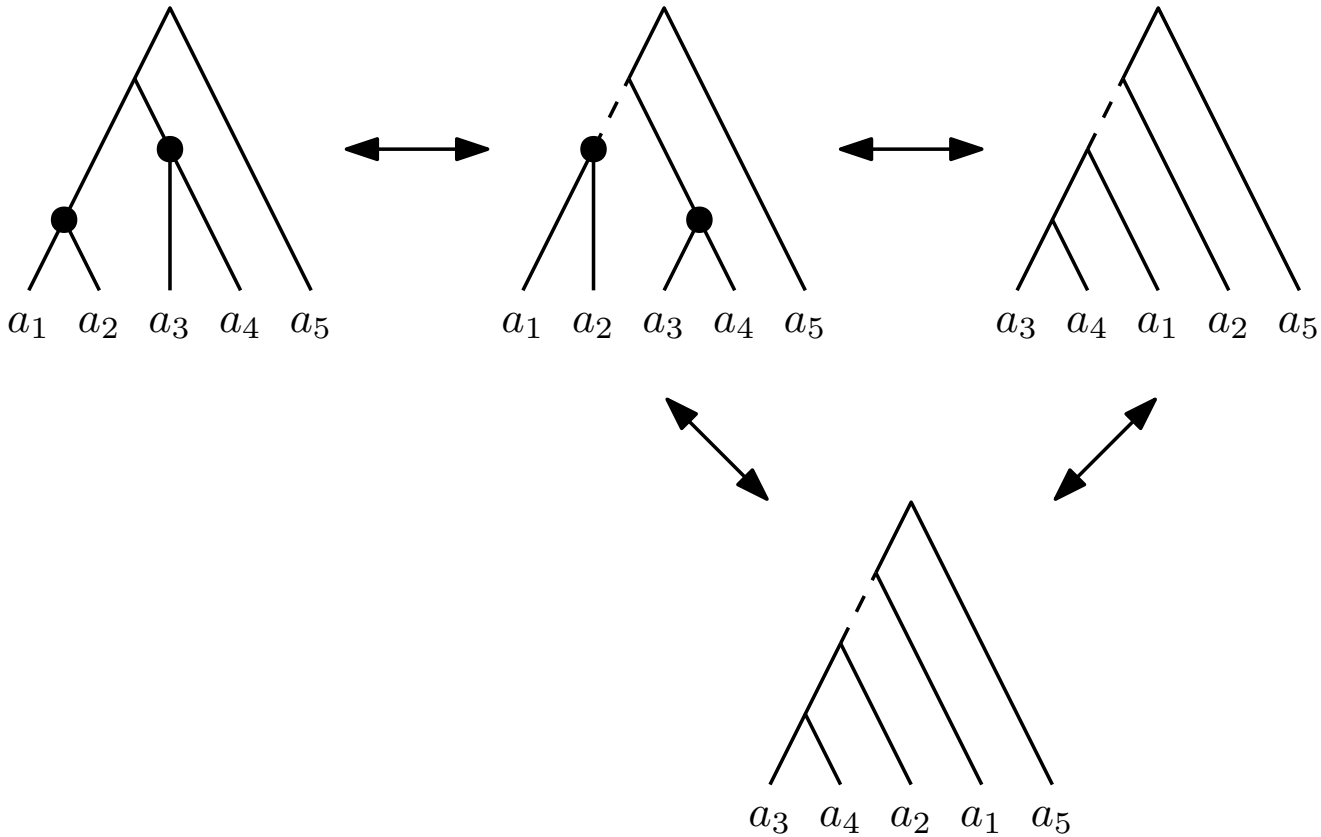
Complexity

RNNI(ρ)-SP:

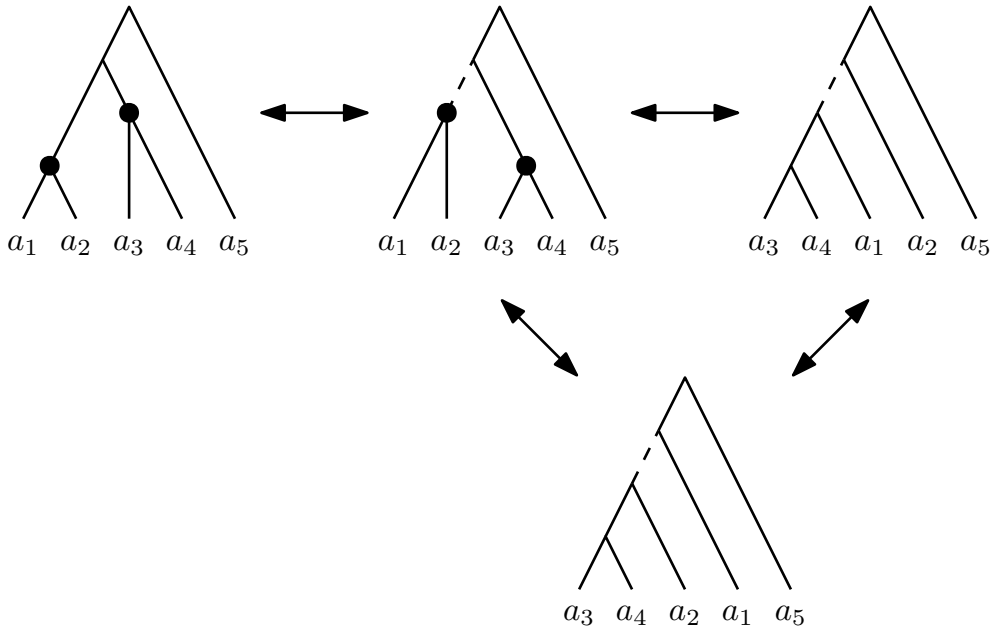
INSTANCE: A pair of trees T and R
FIND: A path of minimal weight between T and R in RNNI(ρ)

- RNNI(0)-SP is $\mathcal{NP}$-hard
- RNNI(ρ)-SP is $\mathcal{NP}$-hard for small $\rho > 0$
- RNNI(1)-SP is polynomial

RNNI(1)

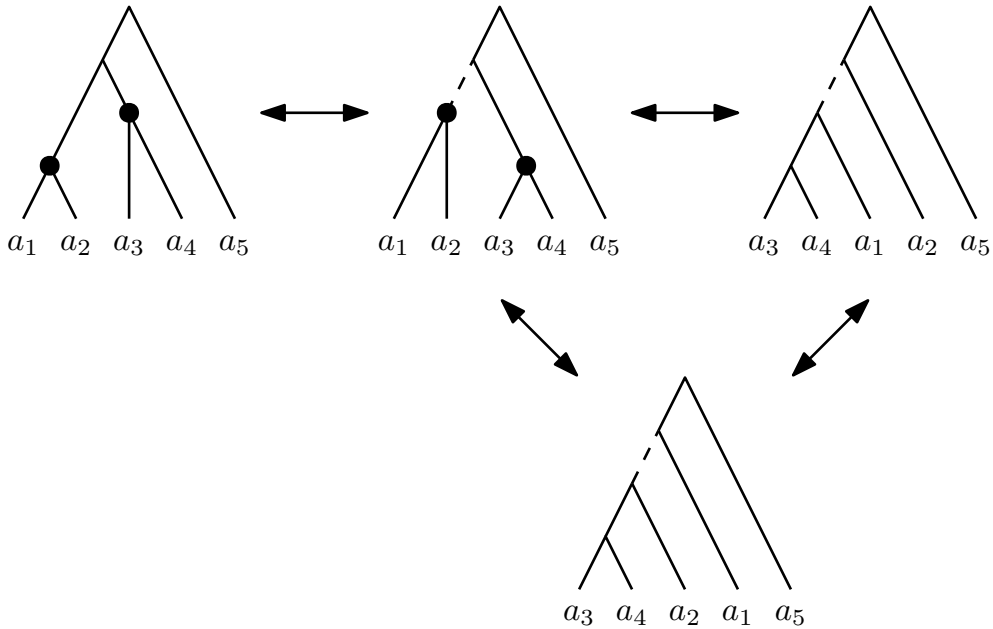


RNNI



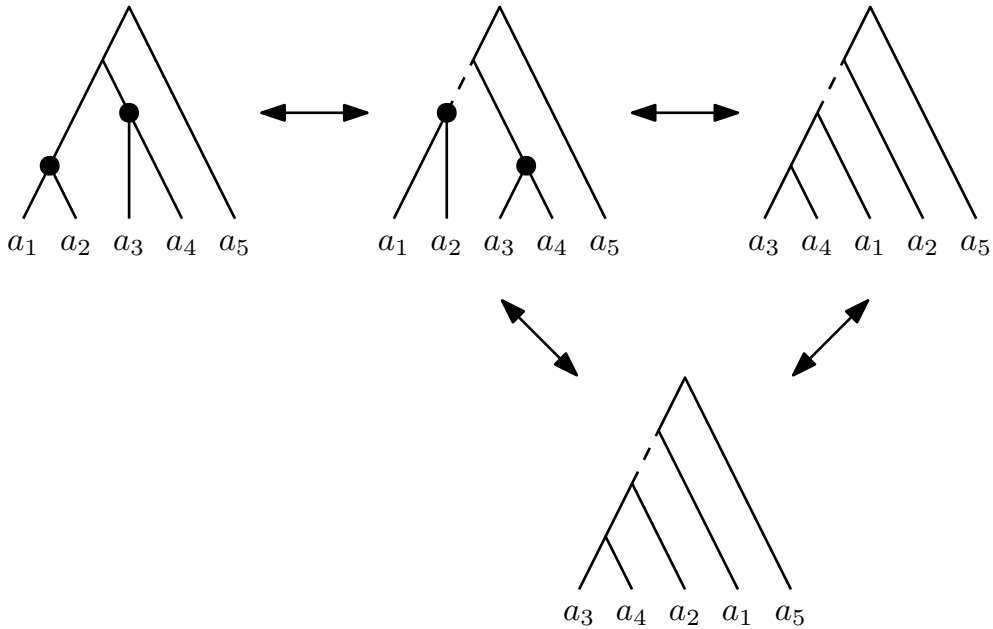
- Diameter $\frac{(n-1)(n-2)}{2}$

RNNI



- Diameter $\frac{(n-1)(n-2)}{2}$
- Radius $\frac{(n-1)(n-2)}{2}$

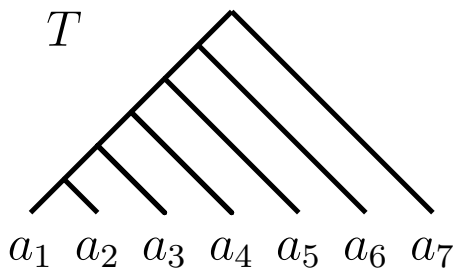
RNNI

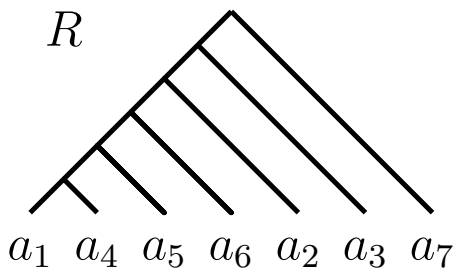


- Diameter $\frac{(n-1)(n-2)}{2}$
- Radius $\frac{(n-1)(n-2)}{2}$
- The set of caterpillar trees is convex

RNNI

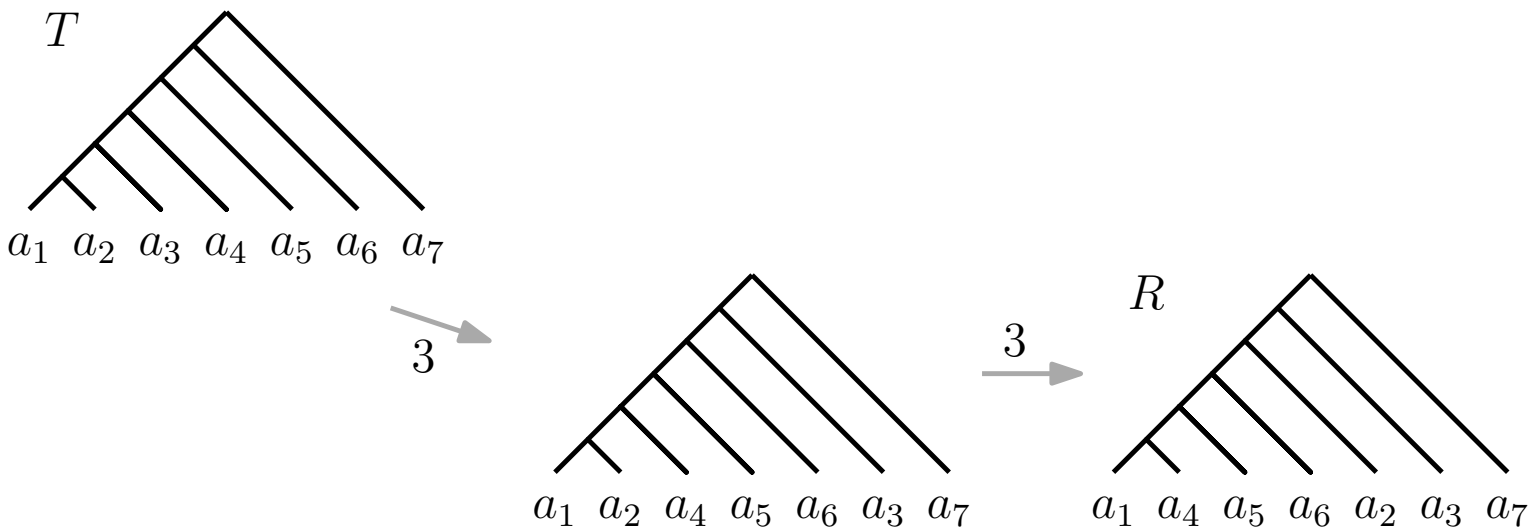
- Diameter $\frac{(n-1)(n-2)}{2}$
- Radius $\frac{(n-1)(n-2)}{2}$
- The set of caterpillar trees is convex





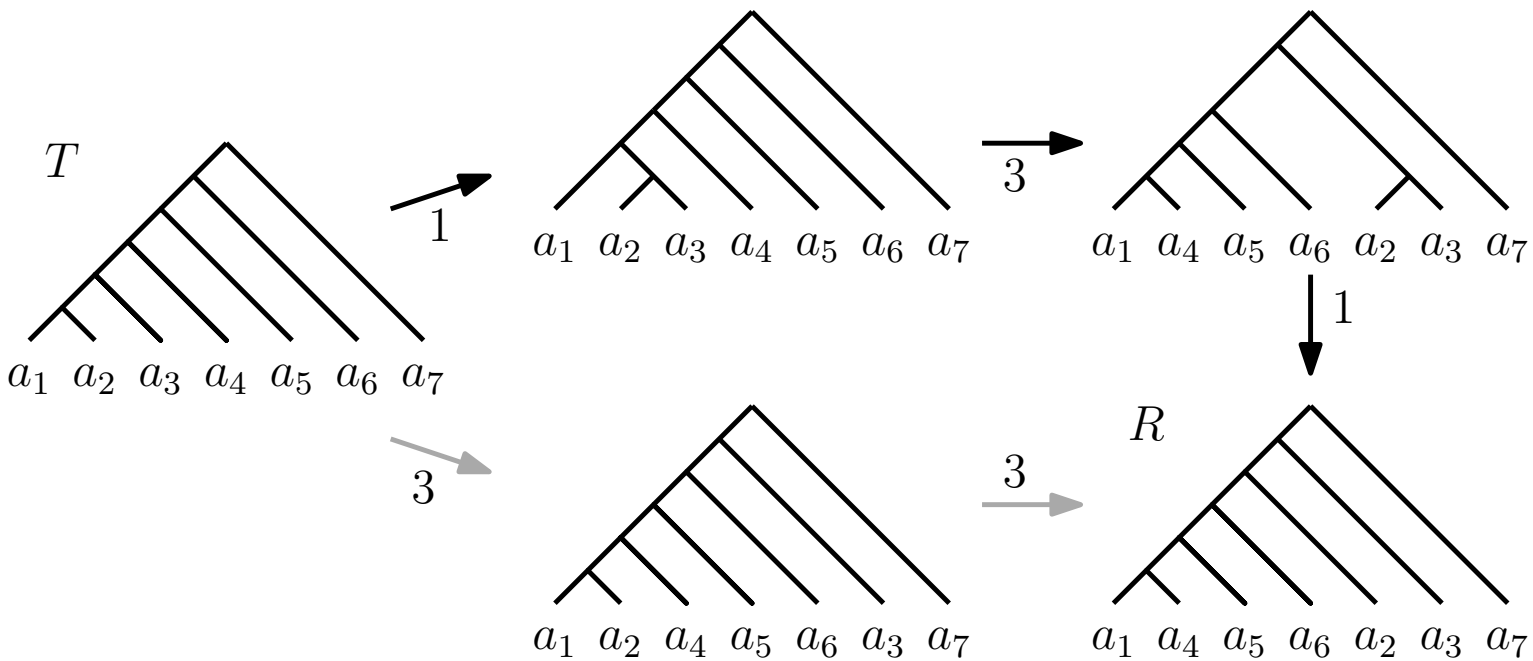
RNNI

- Diameter $\frac{(n-1)(n-2)}{2}$
- Radius $\frac{(n-1)(n-2)}{2}$
- The set of caterpillar trees is convex

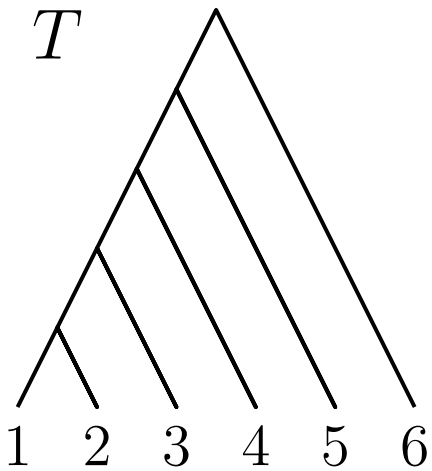


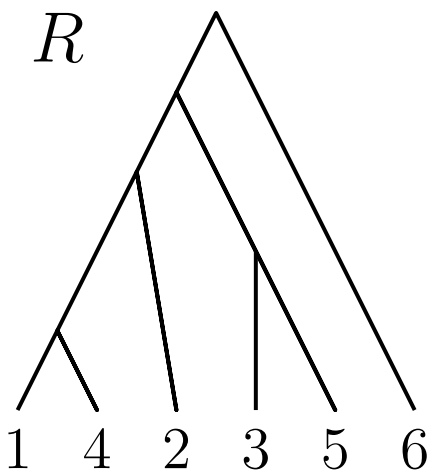
RNNI

- Diameter $\frac{(n-1)(n-2)}{2}$
- Radius $\frac{(n-1)(n-2)}{2}$
- The set of caterpillar trees is convex

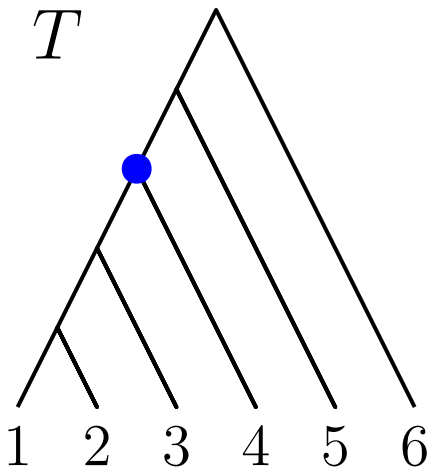


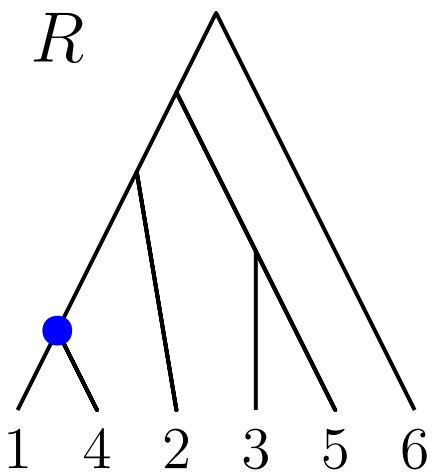
FindPath



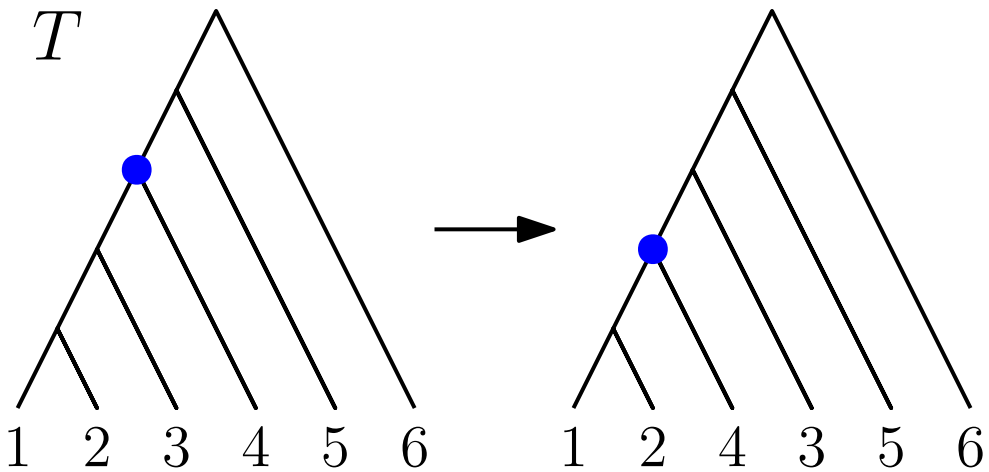


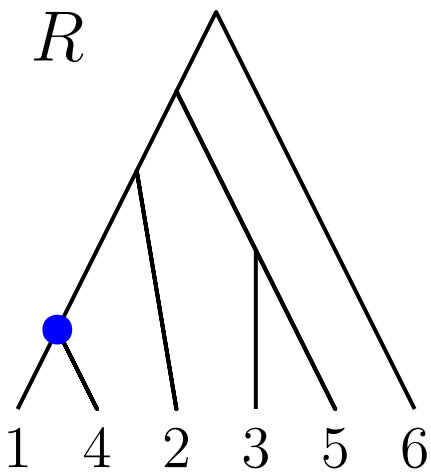
FindPath



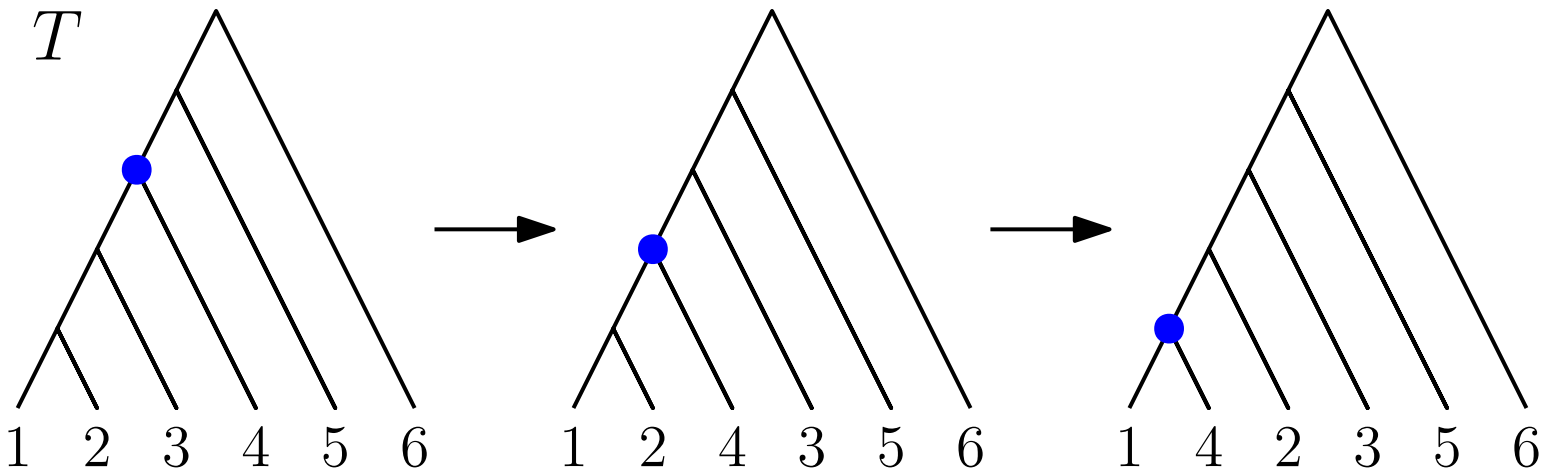


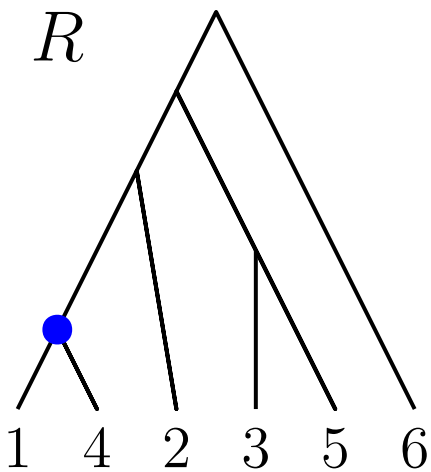
FindPath



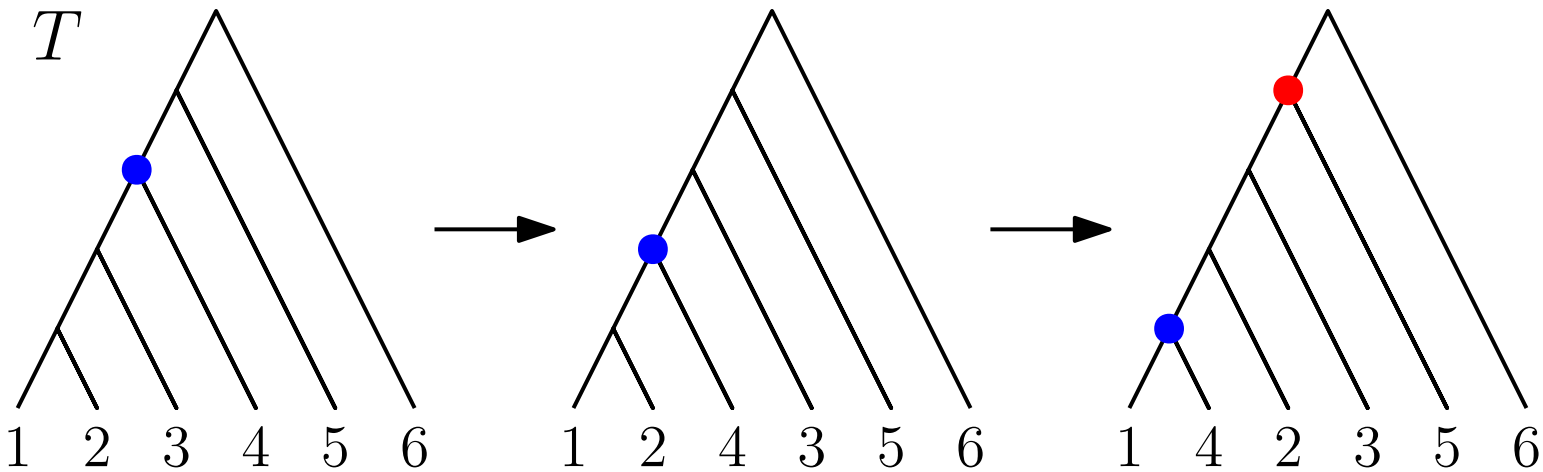


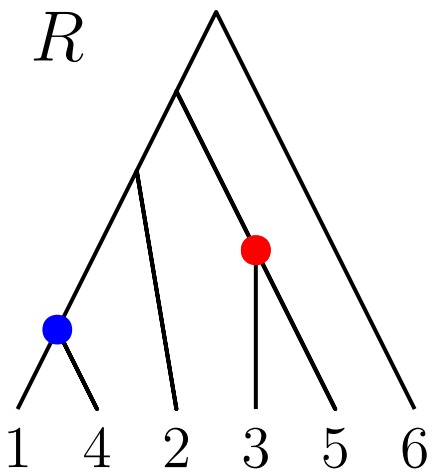
FindPath



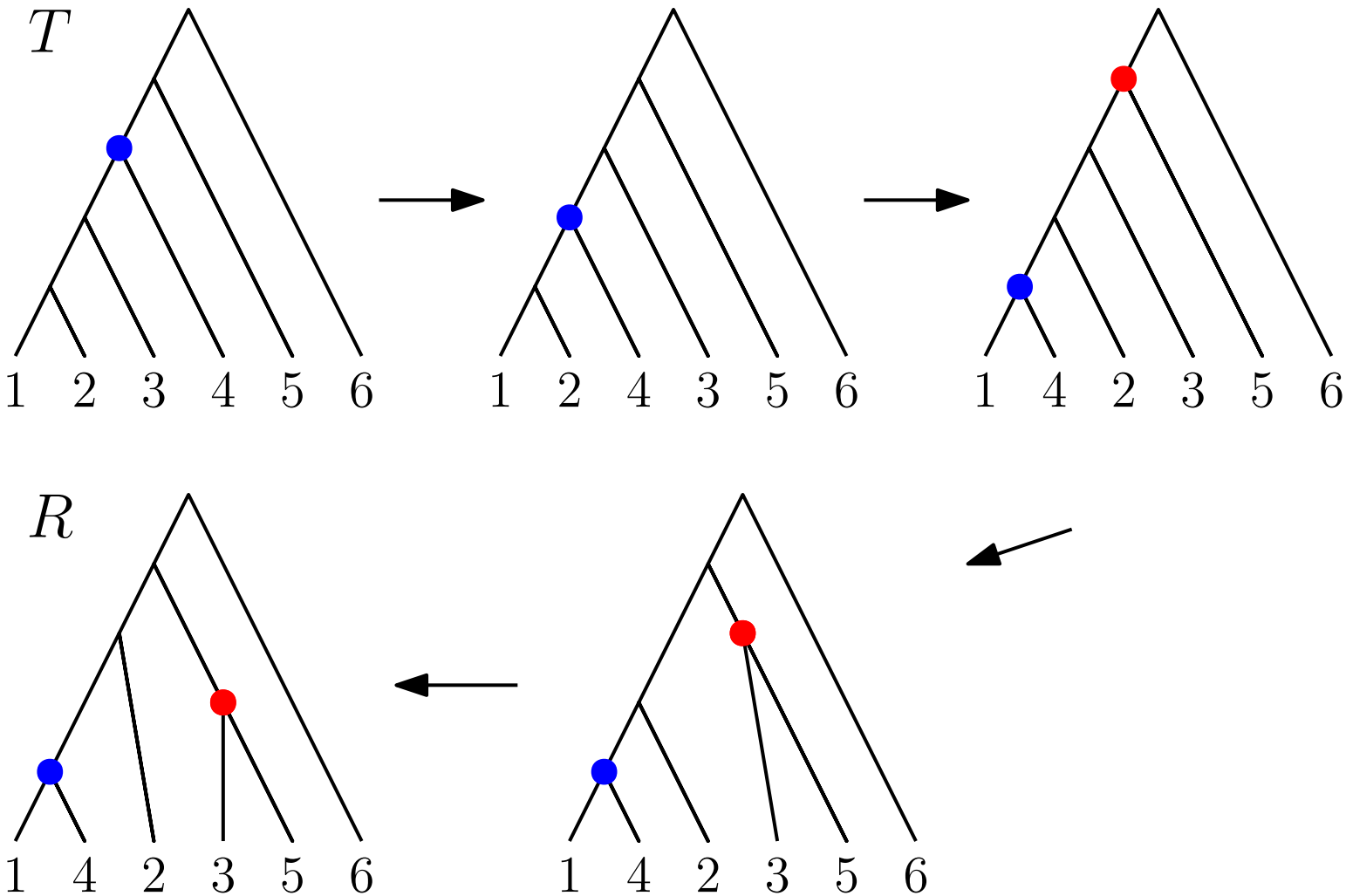


FindPath





FindPath



FindPath

- Greedy algorithm for approximating $\mathrm{RNNI}(1)$-SP

FindPath

- Greedy algorithm for approximating $\mathrm{RNNI}(1)$-SP
- Running time $\mathcal{O}(n^2)$

FindPath

- Greedy algorithm for approximating $\mathrm{RNNI}(1)$-SP
- Running time $\mathcal{O}(n^2)$
- Shortest paths for up to 7 leaves

FindPath

Theorem

FindPath *computes shortest paths in* RNNI.

FindPath

Theorem

FindPath *computes shortest paths in* RNNI.

Idea for proof

$\mathrm{FP}(T, R) :=$ *path between* T *and* R *computed by* FindPath

FindPath

Theorem

FindPath *computes shortest paths in* RNNI.

Idea for proof

$\mathrm{FP}(T, R) :=$ *path between* T *and* R *computed by* FindPath

Lemma

If for all trees T, R *and neighbour* T' *of* T *it is*

$$|\mathrm{FP}(T', R)| \geq |\mathrm{FP}(T, R)| - 1$$

,then

$$|\mathrm{FP}(T, R)| = d(T, R)$$

for all trees T *and* R

FindPath

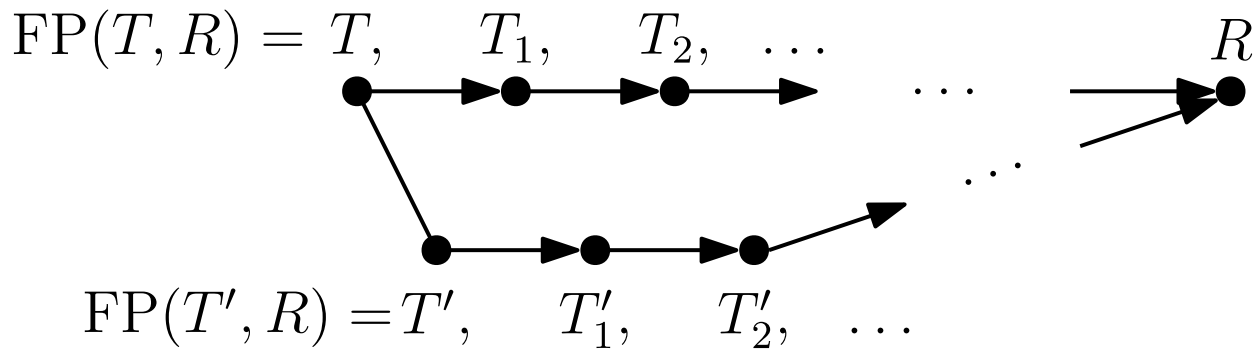
Theorem

FindPath *computes shortest paths in* RNNI.

Idea for proof

$\mathrm{FP}(T, R) :=$ *path between* T *and* R *computed by* FindPath

$$|\mathrm{FP}(T', R)| \geq |\mathrm{FP}(T, R)| - 1$$

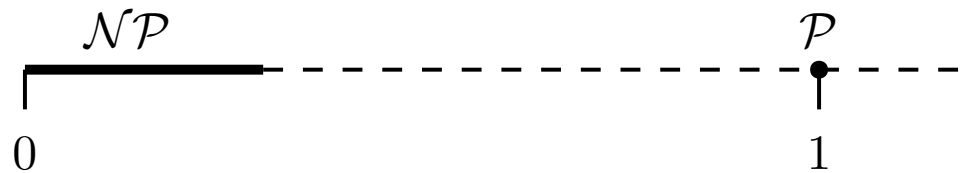


FindPath

RNNI(ρ)-SP:

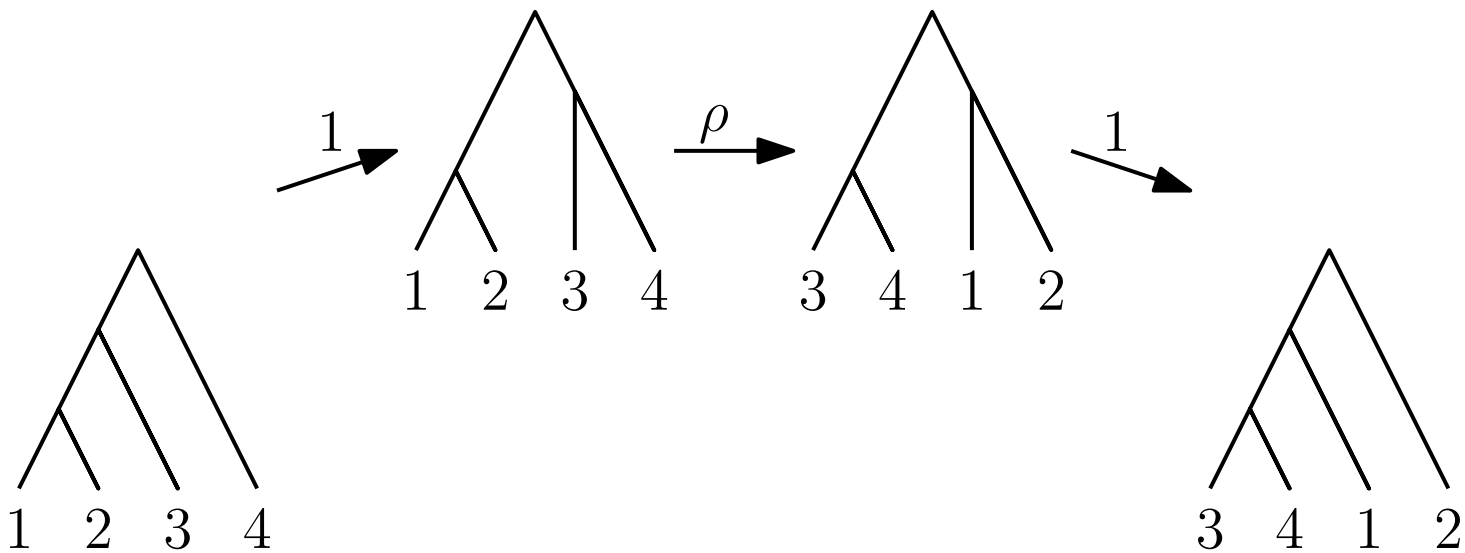
INSTANCE: A pair of trees T and R
FIND: A path of minimal weight between T and R in RNNI(ρ)

- RNNI(0)-SP is $\mathcal{NP}$-hard
- RNNI(ρ)-SP is $\mathcal{NP}$-hard for small $\rho > 0$
- RNNI(1)-SP is polynomial



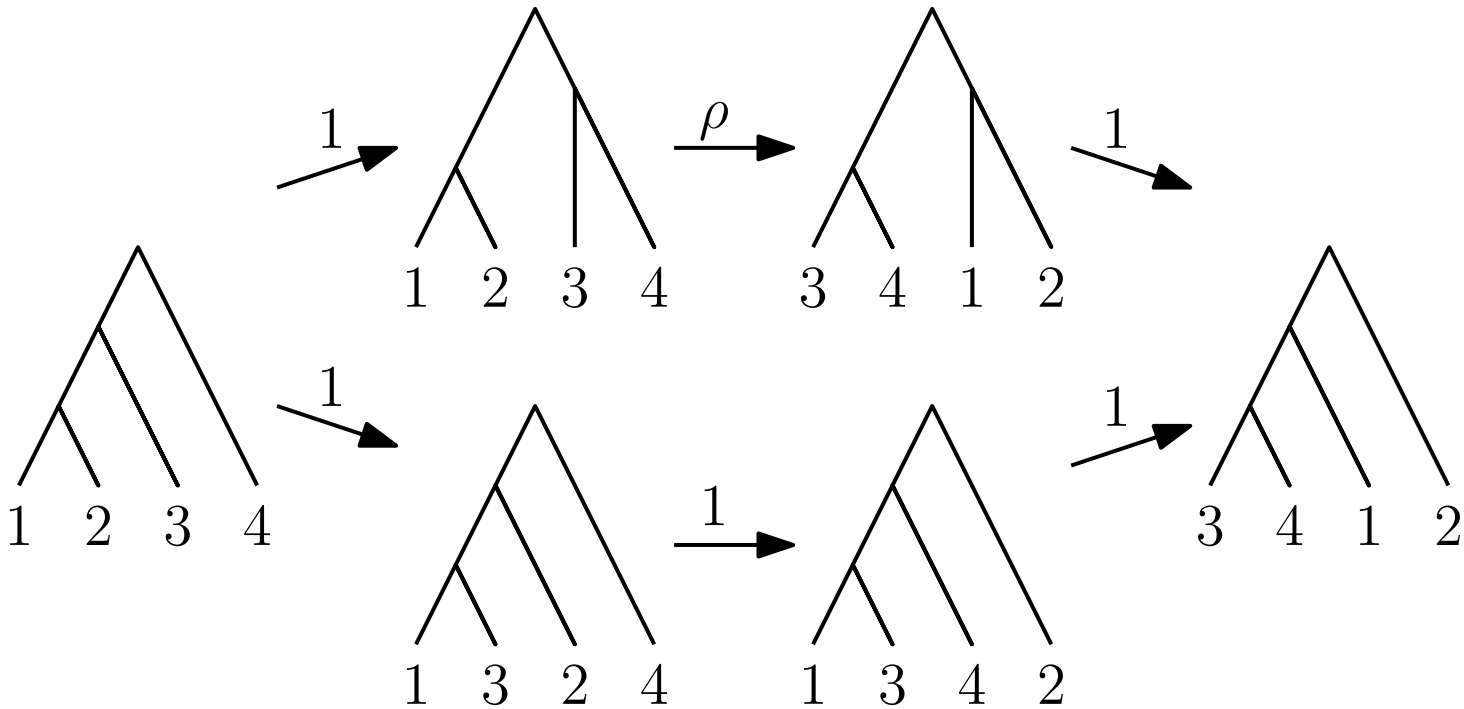
RNNI(ρ) for $\rho > 1$

FindPath does not work:



RNNI(ρ) for $\rho > 1$

FindPath does not work:

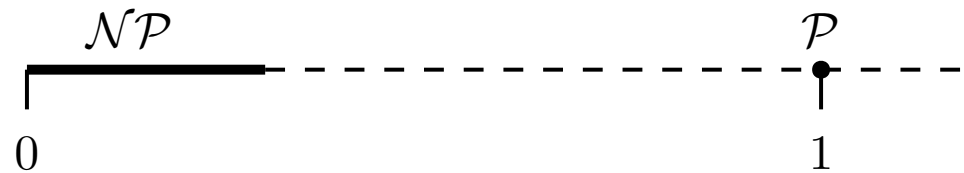


$\mathrm{RNNI}(\rho)$ for $\rho > 1$

$\mathrm{RNNI}(\rho)$-SP:

INSTANCE: A pair of trees T and R
FIND: A path of minimal weight between T and R in $\mathrm{RNNI}(\rho)$

- $\mathrm{RNNI}(0)$-SP is $\mathcal{NP}$-hard
- $\mathrm{RNNI}(\rho)$-SP is $\mathcal{NP}$-hard for small $\rho > 0$
- $\mathrm{RNNI}(1)$-SP is polynomial



$\Rightarrow$ What about $\mathrm{RNNI}(\rho)$-SP for other values of ρ?

Thank you