Graphs in phylogenomics, a few applications

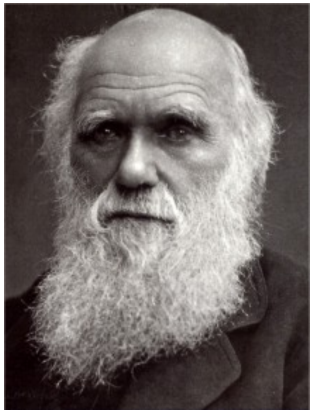
Celine Scornavacca

17/11/2020

From Aristotle to Darwin

Since Aristotle, naturalists have always tried to classify the abundance of creatures that populate the Earth.

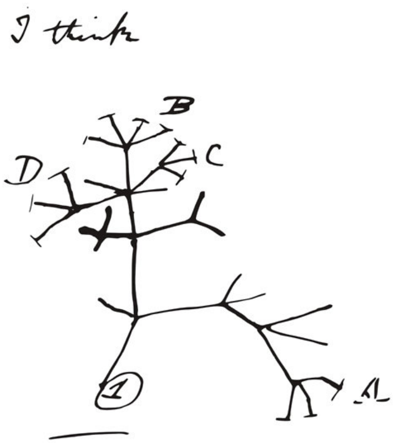
Aristote: the scala naturae;
Carl von Linné: classification of living;
Antoine Laurent de Jussieu;
Leclerc de Buffon: the first to evoke the possibility that species can evolve;
Jean-Baptiste Lamarck: first theory of evolution;
Charles Darwin: The Origins of Species (1859).



From *The Origin of Species*

- It is a truly wonderful fact that all animals and all plants throughout all time and space should be related to each other in groups, subordinate to groups. [...]

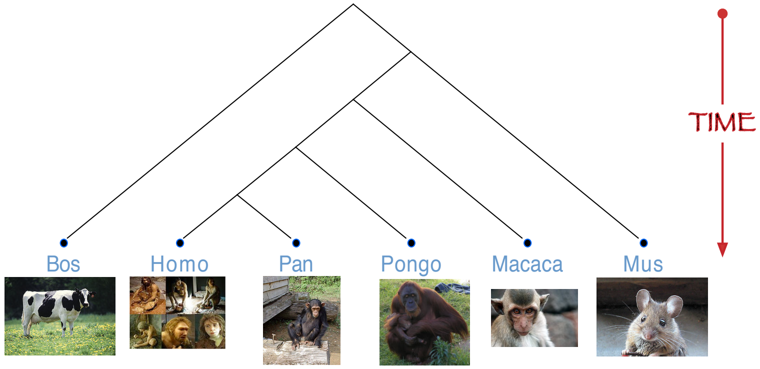
- The affinities of all the beings of the same class have sometimes been represented by a great tree. [...] The green and budding twigs may represent existing species; and those produced during former years may represent the long succession of extinct species.



Charles Darwin, (1872), pp. 170-171. The Origin of Species. Sixth Edition. The Modern Library, New York.

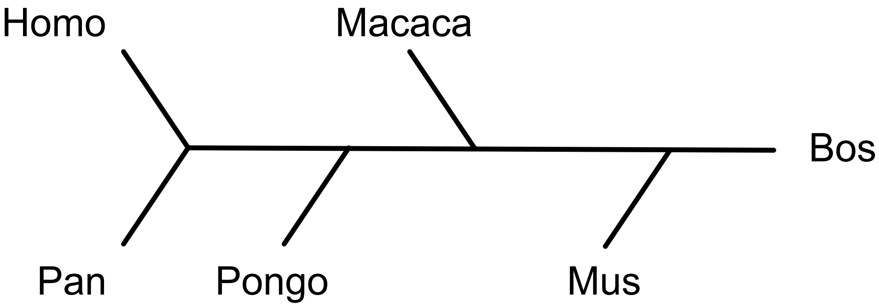
Phylogenetics/phylogenetic trees

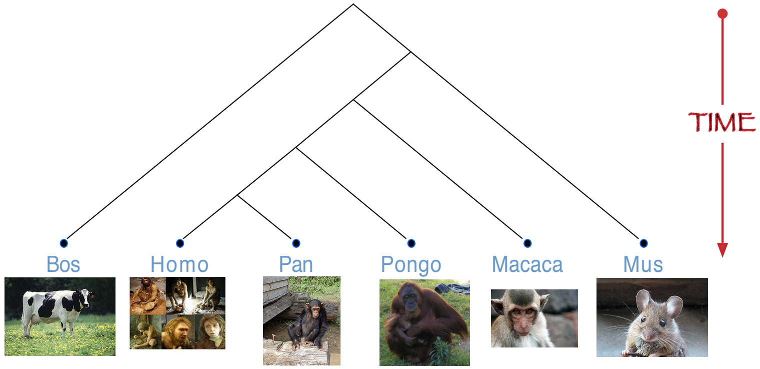
- Phylogenetics aims at clarifying, using molecular and morphological data, the evolutionary relationships that exist among different species
- These relationships can be represented through *phylogenetic trees* or *phylogenies*, out-branching trees with no indegree-1 outdegreee-1 nodes, where sinks are associated to a set of species (often *binary*)
  - the sinks or taxa represent existing organisms
  - the only node with indegree-0 is called root
  - internal nodes represent hypothetical ancestors
  - each internal node represents the lowest common ancestor of all taxa below it (clade)
  - nodes and branches can have several kinds of information associated with them, such as time or amount of evolution estimates



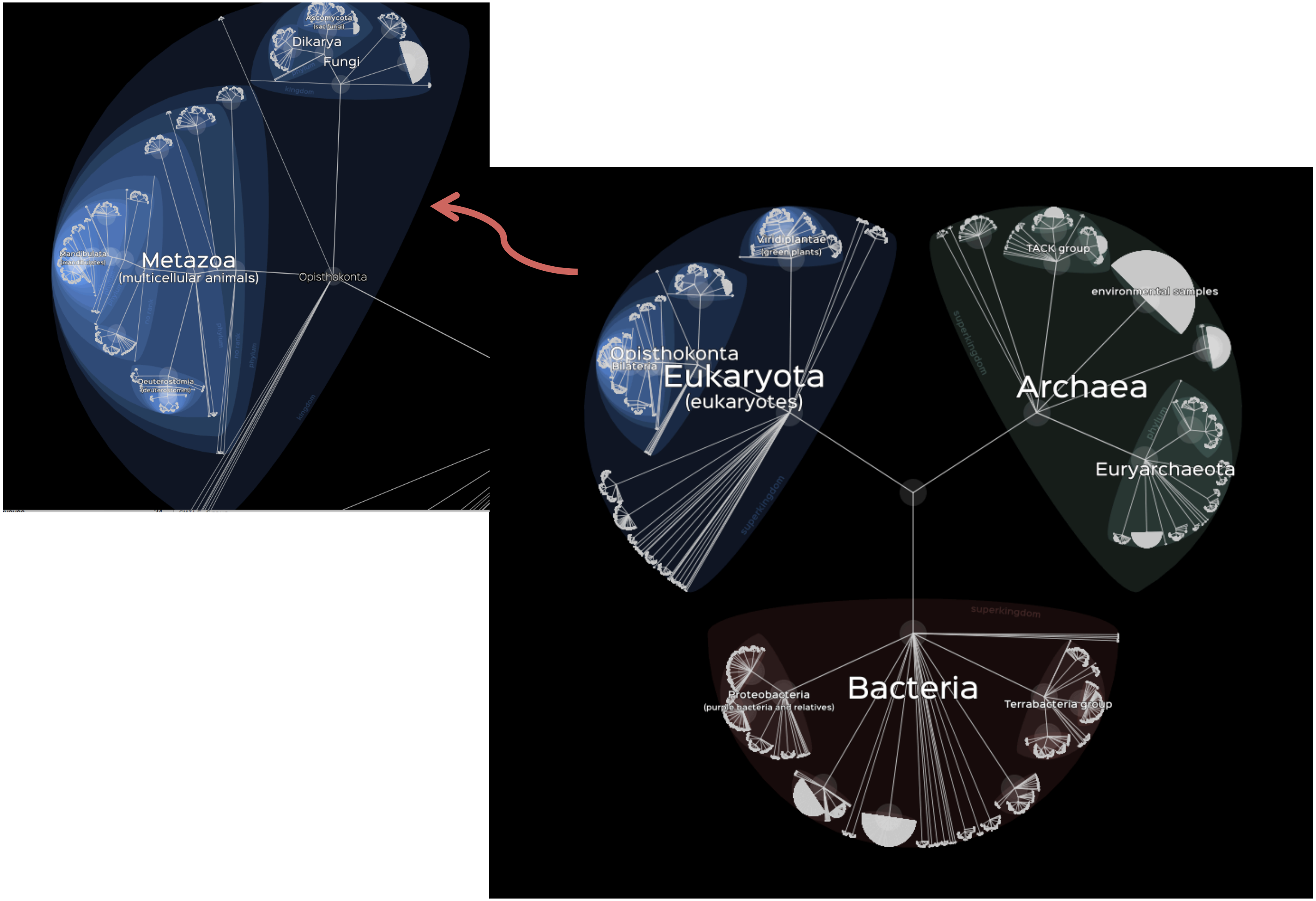
Phylogenetics/phylogenetic trees

- Phylogenetics aims at clarifying, using molecular and morphological data, the evolutionary relationships that exist among different species
- These relationships can be represented through *phylogenetic trees* or *phylogenies*, out-branching trees with no indegree-1 outdegreee-1 nodes, where sinks are associated to a set of species (often *binary*)



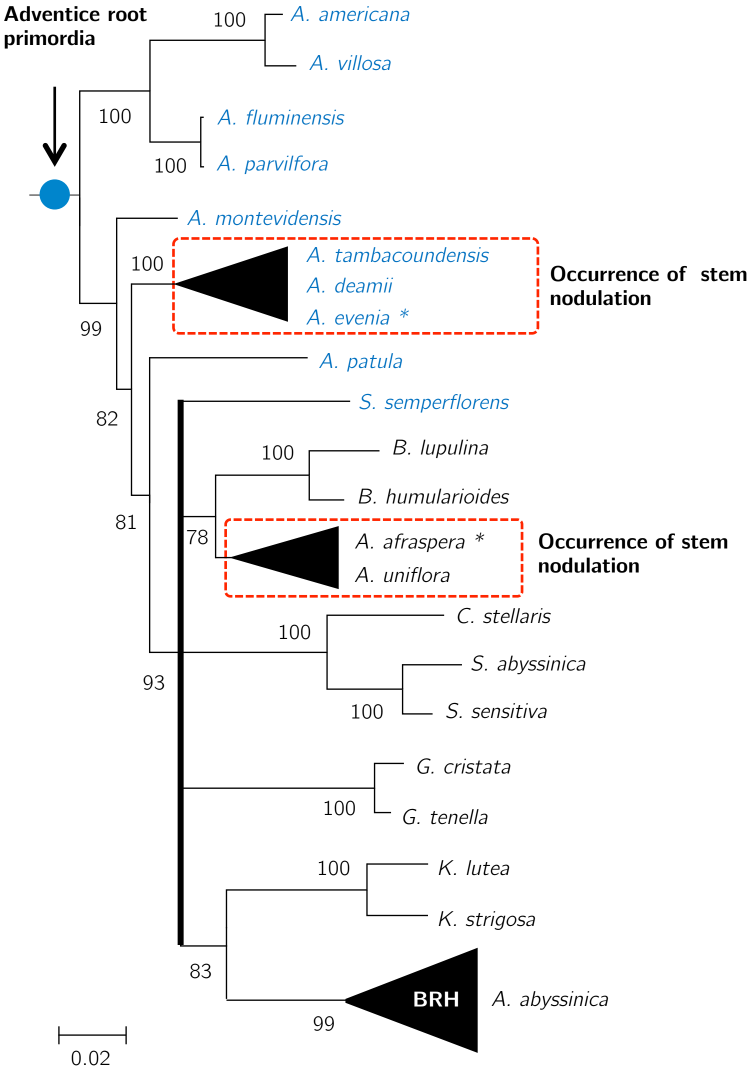


Applications: the TOL – Tree Of Life



de Vienne DM (2016) Lifemap: Exploring the Entire Tree of Life. PLOS Biology.

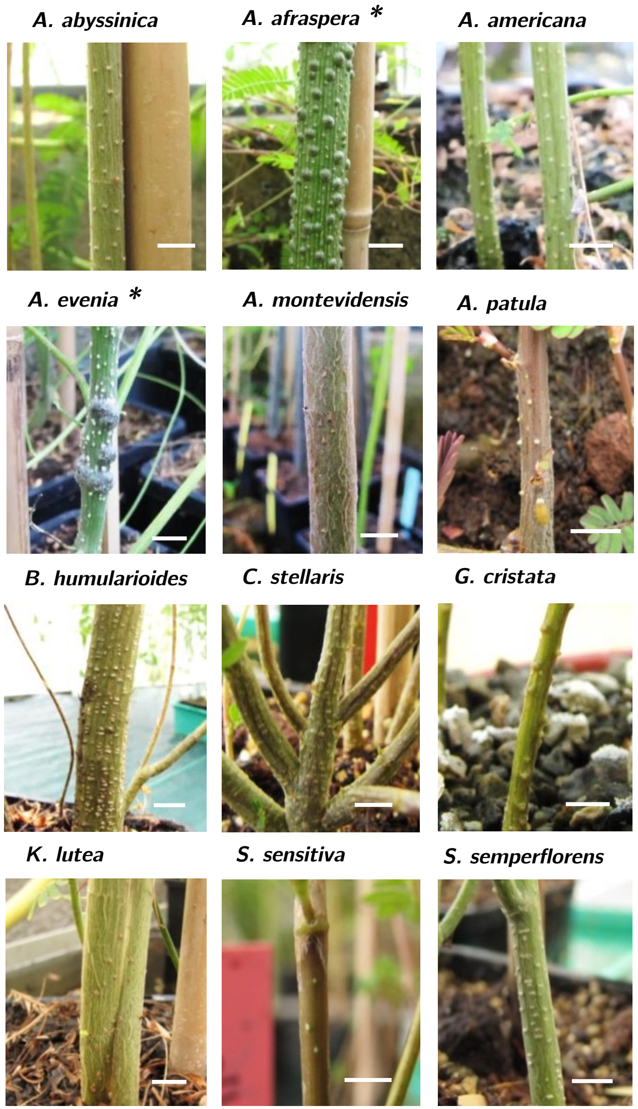
Applications: character evolution



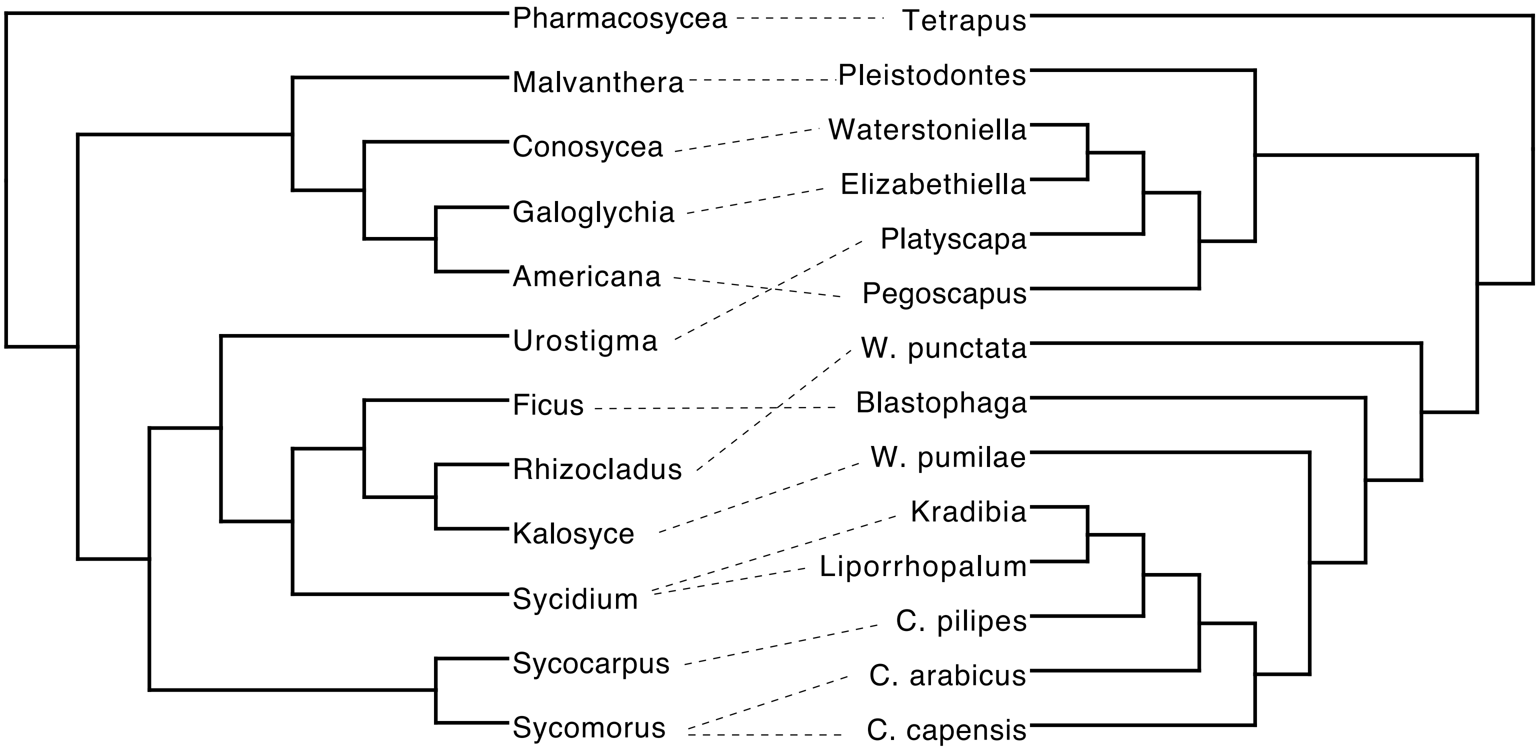
Adventice root primordia

Occurrence of stem nodulation

Occurrence of stem nodulation



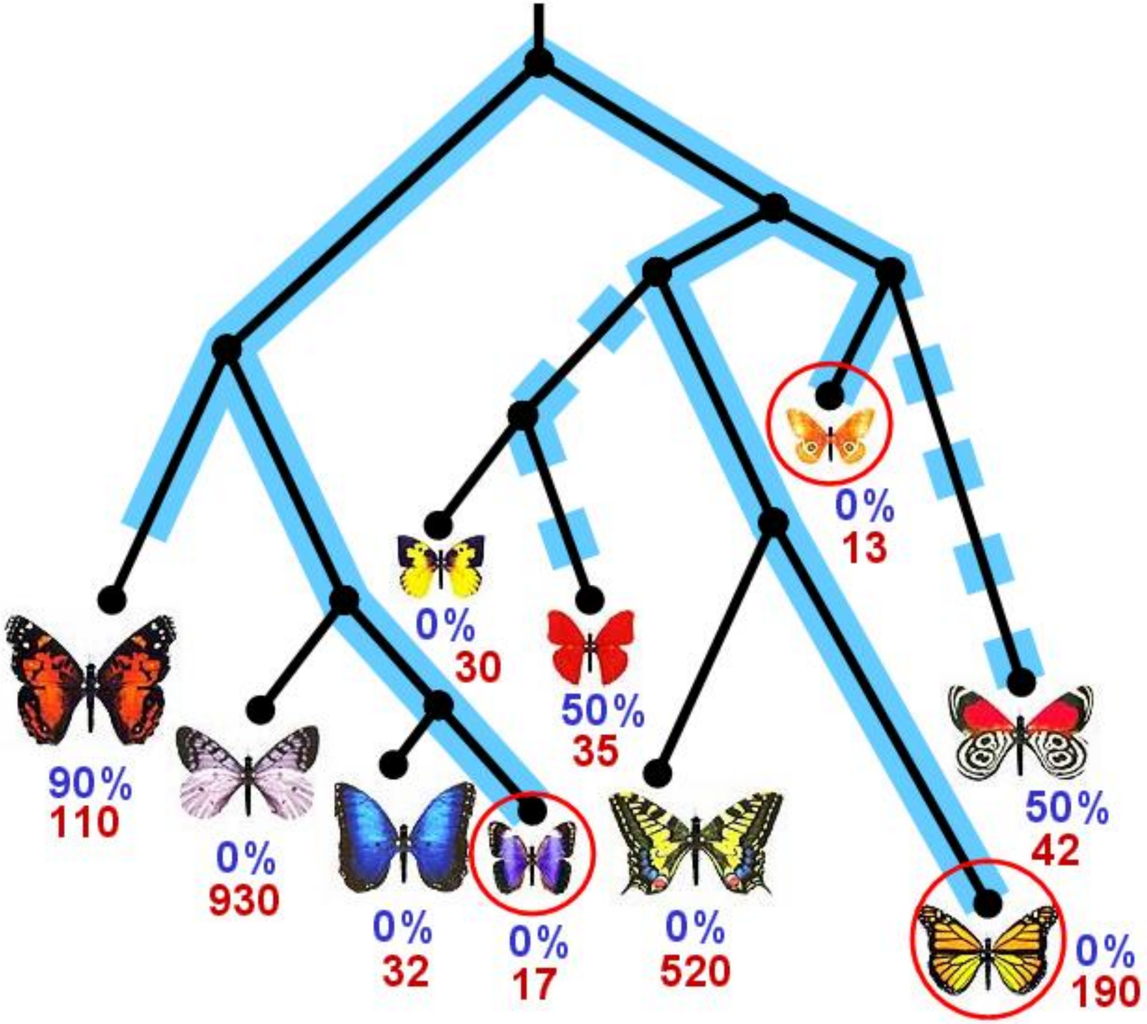
Applications: co-evolution



ficus trees

wasps

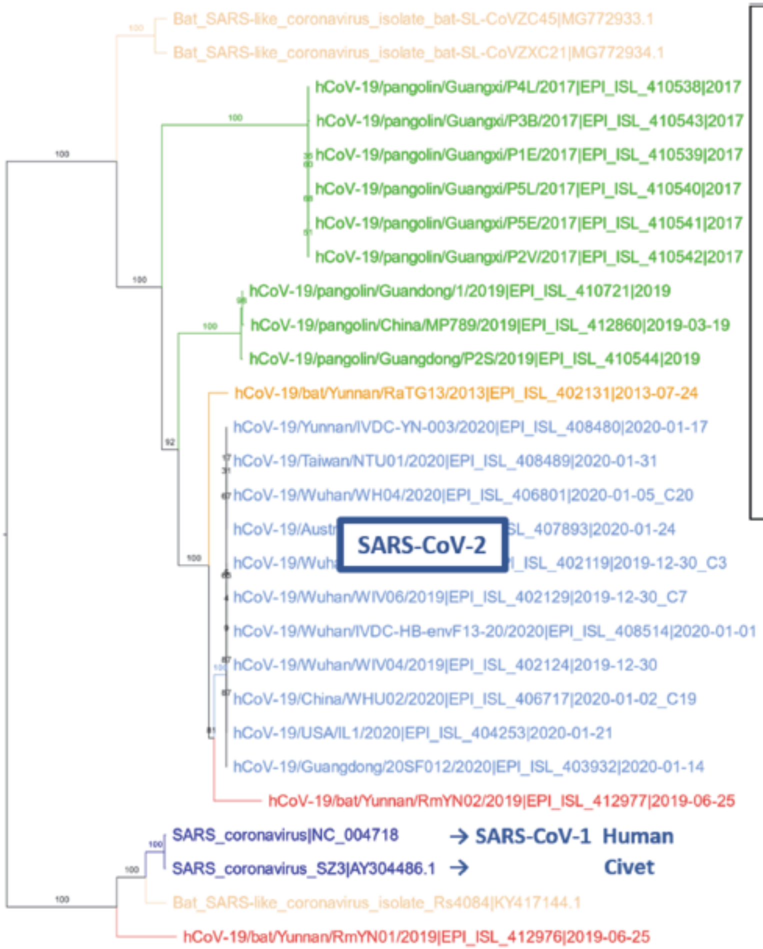
Applications: the Noah's Ark Problem



F. Pardi and N. Goldman (2007). Resource aware taxon selection for maximizing phylogenetic diversity. Systematic Biology.

Applications: disease evolution and spreading

Phylogeny of SARS-CoV-2 related strains (GISAID, 10/5/2020)


Bat_SARS-like_coronavirus_isolate_bat-SL-CoVZC45|MG772933.1
Bat_SARS-like_coronavirus_isolate_bat-SL-CoVZXC21|MG772934.1
hCoV-19/pangolin/Guangxi/P4L/2017|EPI_ISL_410538|2017
hCoV-19/pangolin/Guangxi/P3B/2017|EPI_ISL_410543|2017
hCoV-19/pangolin/Guangxi/P1E/2017|EPI_ISL_410539|2017
hCoV-19/pangolin/Guangxi/P5L/2017|EPI_ISL_410540|2017
hCoV-19/pangolin/Guangxi/P5E/2017|EPI_ISL_410541|2017
hCoV-19/pangolin/Guangxi/P2V/2017|EPI_ISL_410542|2017
hCoV-19/pangolin/Guandong/1/2019|EPI_ISL_410721|2019
hCoV-19/pangolin/China/MP789/2019|EPI_ISL_412860|2019-03-19
hCoV-19/pangolin/Guangdong/P2S/2019|EPI_ISL_410544|2019
hCoV-19/bat/Yunnan/RaTG13/2013|EPI_ISL_402131|2013-07-24
hCoV-19/Yunnan/IVDC-YN-003/2020|EPI_ISL_408480|2020-01-17
hCoV-19/Taiwan/NTU01/2020|EPI_ISL_408489|2020-01-31
hCoV-19/Wuhan/WH04/2020|EPI_ISL_406801|2020-01-05_C20
hCoV-19/Austr
L_407893|2020-01-24
hCoV-19/Wuh
I_ISL_402119|2019-12-30_C3
SARS-CoV-2
hCoV-19/Wuhan/WIV06/2019|EPI_ISL_402129|2019-12-30_C7
hCoV-19/Wuhan/IVDC-HB-envF13-20/2020|EPI_ISL_408514|2020-01-01
hCoV-19/Wuhan/WIV04/2019|EPI_ISL_402124|2019-12-30
hCoV-19/China/WHU02/2020|EPI_ISL_406717|2020-01-02_C19
hCoV-19/USA/IL1/2020|EPI_ISL_404253|2020-01-21
hCoV-19/Guangdong/20SF012/2020|EPI_ISL_403932|2020-01-14
hCoV-19/bat/Yunnan/RmYN02/2019|EPI_ISL_412977|2019-06-25
SARS_coronavirus|NC_004718
→ SARS-CoV-1 Human
SARS_coronavirus_SZ3|AY304486.1 →
Civet
Bat_SARS-like_coronavirus_isolate_Rs4084|KY417144.1
hCoV-19/bat/Yunnan/RmYN01/2019|EPI_ISL_412976|2019-06-25
100
100
100
100
92
100
100
100
100

Light Orange: previous bat CoVs

Orange: closest bat precursor
Yunnan 2013

Red: new bat CoVs
Yunnan 2019

Light blue: hCoV-19 2019-20

Green: pangolin CoV
Southern China 2019

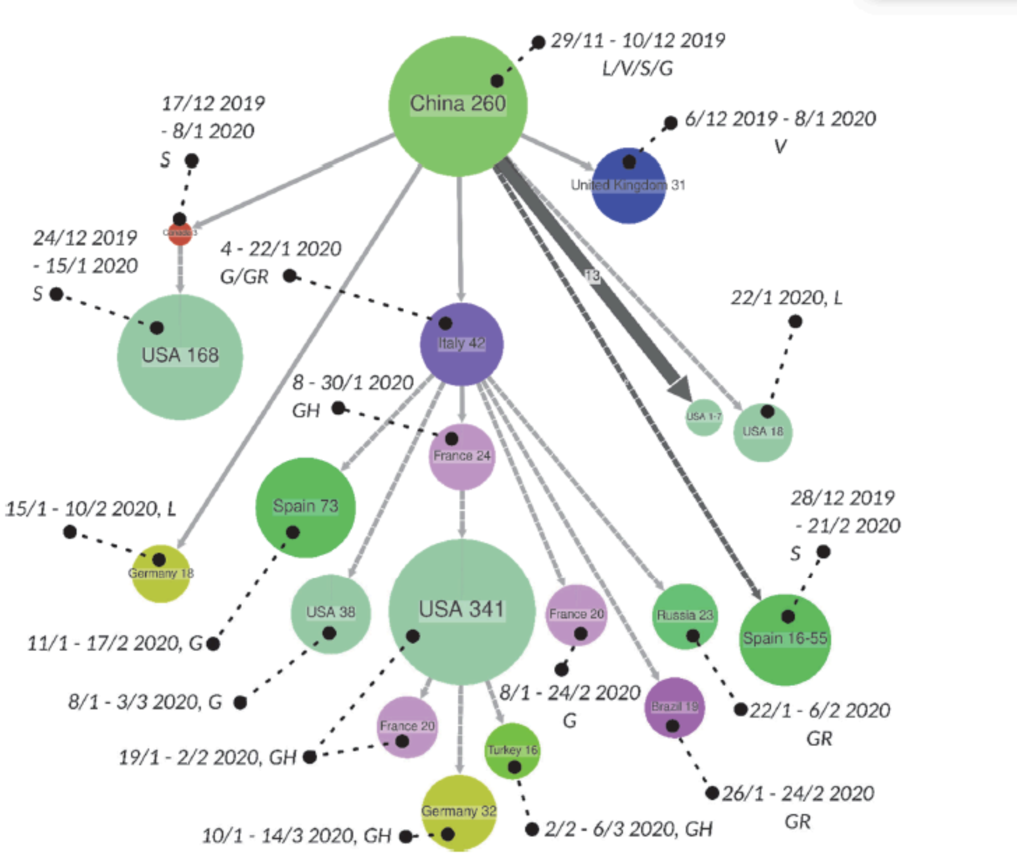
Blue ... SARS CoV

GISAID

Anna Zhukova et al (2020) Origin, evolution and global spread of SARS-CoV-2 To appear in the Comptes Rendus - Biologies of the French Academy of Sciences

Applications: disease evolution and spreading

Phylogenetic scenario showing the main transmission clusters of SARS-CoV-2 until April 25, 2020.

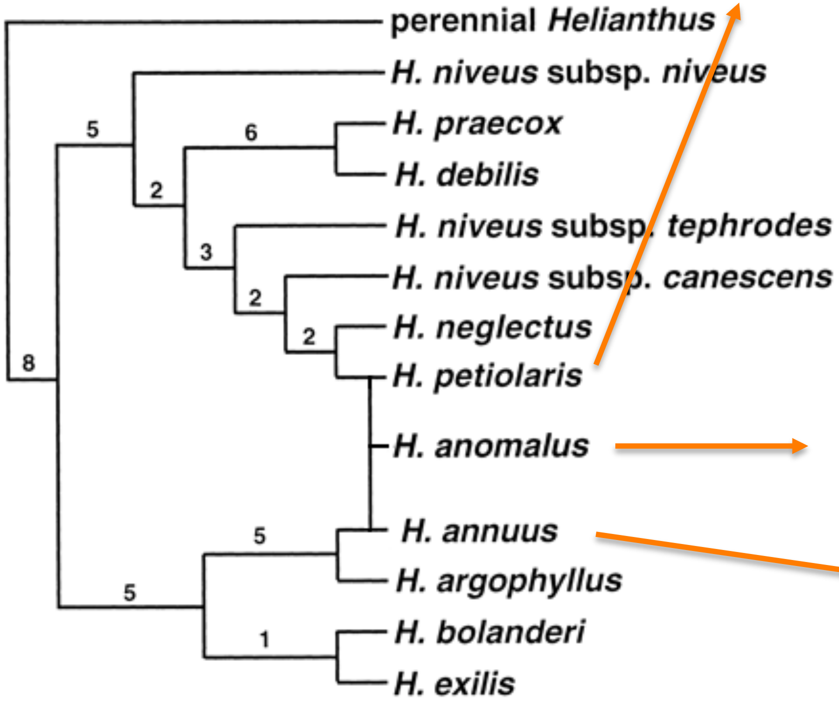


Anna Zhukova et al (2020) Origin, evolution and global spread of SARS-CoV-2 To appear in the Comptes Rendus - Biologies of the French Academy of Sciences

Explicit phylogenetic networks

They represent evolutionary history when inheritance is from multiple ancestors – because of reticulate events, e.g:

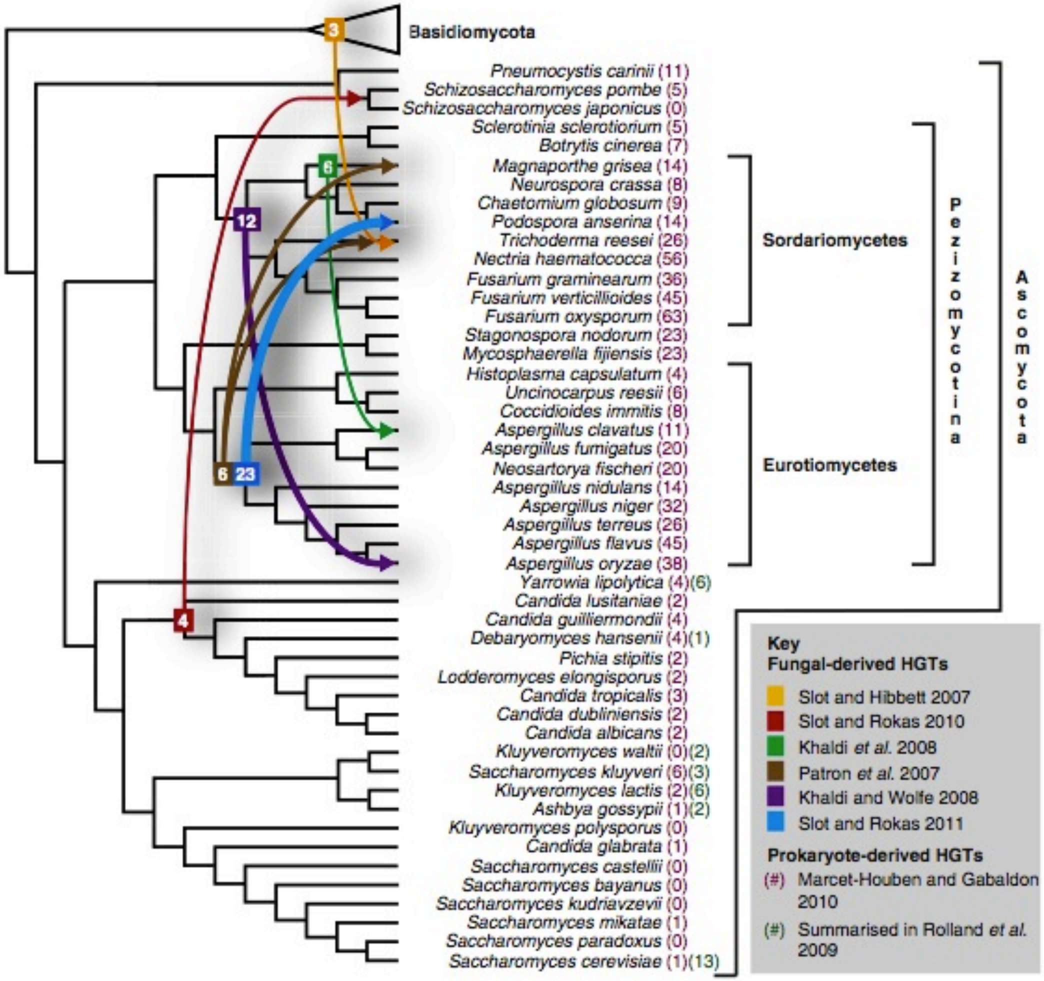
- **Hybrid speciation**
- Lateral gene transfer
- Recombination

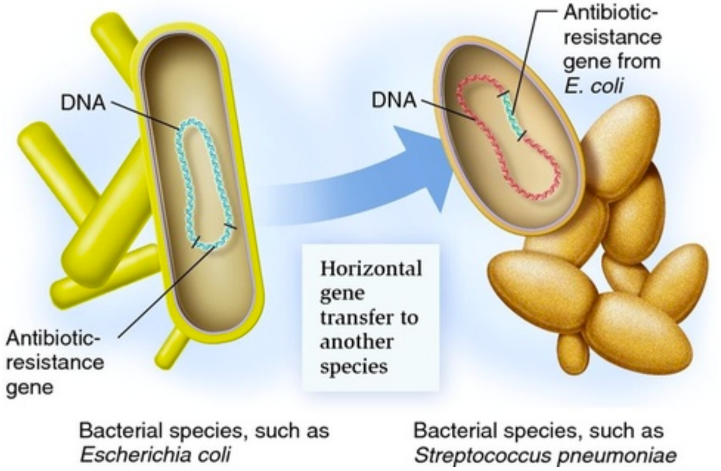


Explicit phylogenetic networks

They represent evolutionary history when inheritance is from multiple ancestors – because of reticulate events, e.g:

- Hybrid speciation
- **Lateral gene transfer**
- Recombination

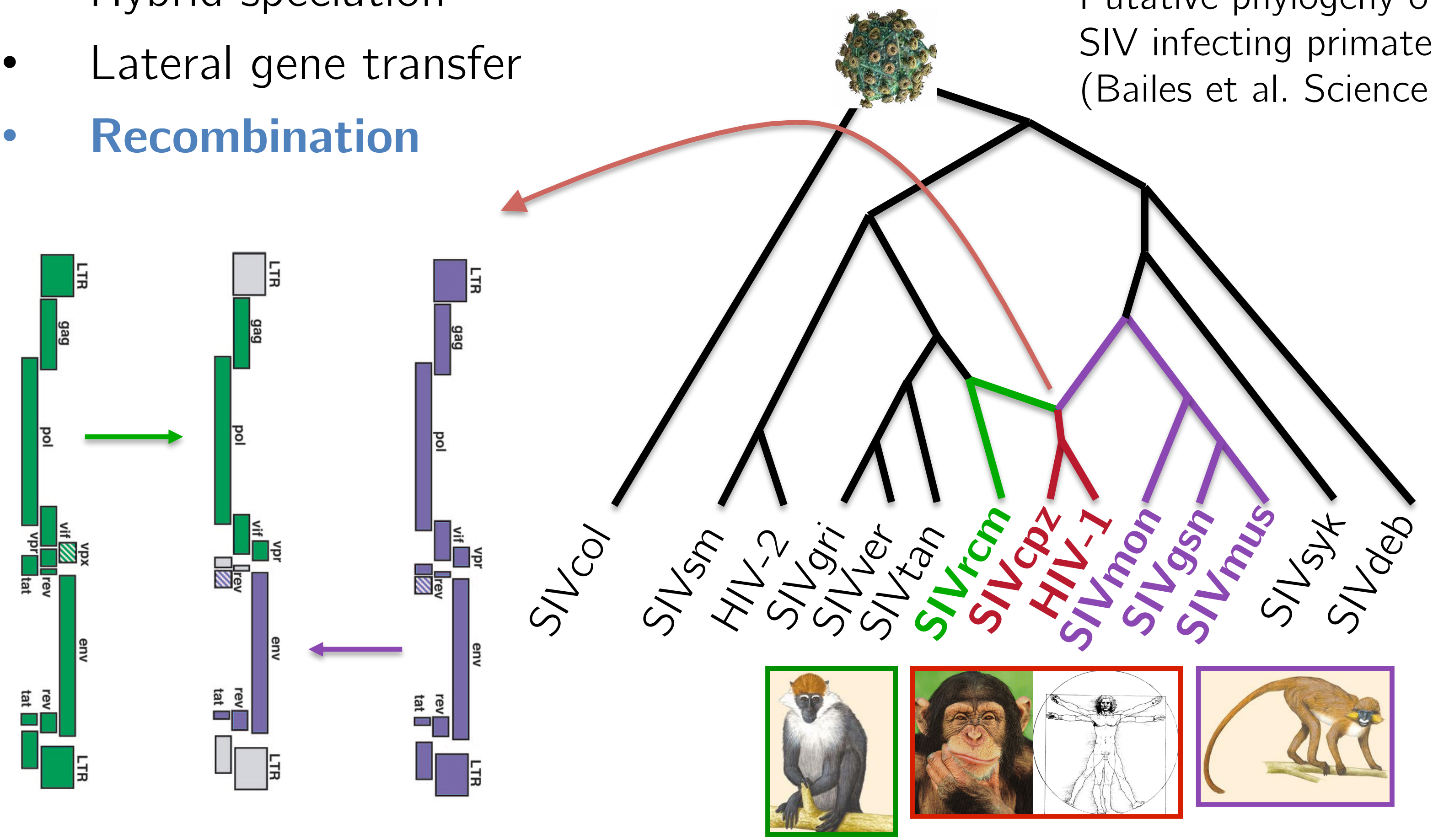

Basidiomycota
3
Pneumocystis carinii (11)
Schizosaccharomyces pombe (5)
Schizosaccharomyces japonicus (0)
Sclerotinia sclerotiorum (5)
Botrytis cinerea (7)
Magnaporthe grisea (14)
Neurospora crassa (8)
Chaetomium globosum (9)
Podospora anserina (14)
Trichoderma reesei (26)
Nectria haematococca (56)
Fusarium graminearum (36)
Fusarium verticillioides (45)
Fusarium oxysporum (63)
Stagonospora nodorum (23)
Mycosphaerella fijiensis (23)
Histoplasma capsulatum (4)
Uncinocarpus reesii (6)
Coccidioides immitis (8)
Aspergillus clavatus (11)
Aspergillus fumigatus (20)
Neosartorya fischeri (20)
Aspergillus nidulans (14)
Aspergillus niger (32)
Aspergillus terreus (26)
Aspergillus flavus (45)
Aspergillus oryzae (38)
Yarrowia lipolytica (4)(6)
Candida lusitaniae (2)
Candida guilliermondii (4)
Debaryomyces hansenii (4)(1)
Pichia stipitis (2)
Lodderomyces elongisporus (2)
Candida tropicalis (3)
Candida dubliniensis (2)
Candida albicans (2)
Kluyveromyces waltii (0)(2)
Saccharomyces kluyveri (6)(3)
Kluyveromyces lactis (2)(6)
Ashbya gossypii (1)(2)
Kluyveromyces polysporus (0)
Candida glabrata (1)
Saccharomyces castellii (0)
Saccharomyces bayanus (0)
Saccharomyces kudriavzevii (0)
Saccharomyces mikatae (1)
Saccharomyces paradoxus (0)
Saccharomyces cerevisiae (1)(13)
6
12
6
23
4
Sordariomycetes
Eurotiomycetes
Pezizomycotina
Ascomycota
Key
Fungal-derived HGTs
Slot and Hibbett 2007
Slot and Rokas 2010
Khaldi et al. 2008
Patron et al. 2007
Khaldi and Wolfe 2008
Slot and Rokas 2011
Prokaryote-derived HGTs
(#) Marcet-Houben and Gabaldon 2010
(#) Summarised in Rolland et al. 2009


DNA
Antibiotic-resistance gene
DNA
Antibiotic-resistance gene from E. coli
Horizontal gene transfer to another species
Bacterial species, such as Escherichia coli
Bacterial species, such as Streptococcus pneumoniae

Explicit phylogenetic networks

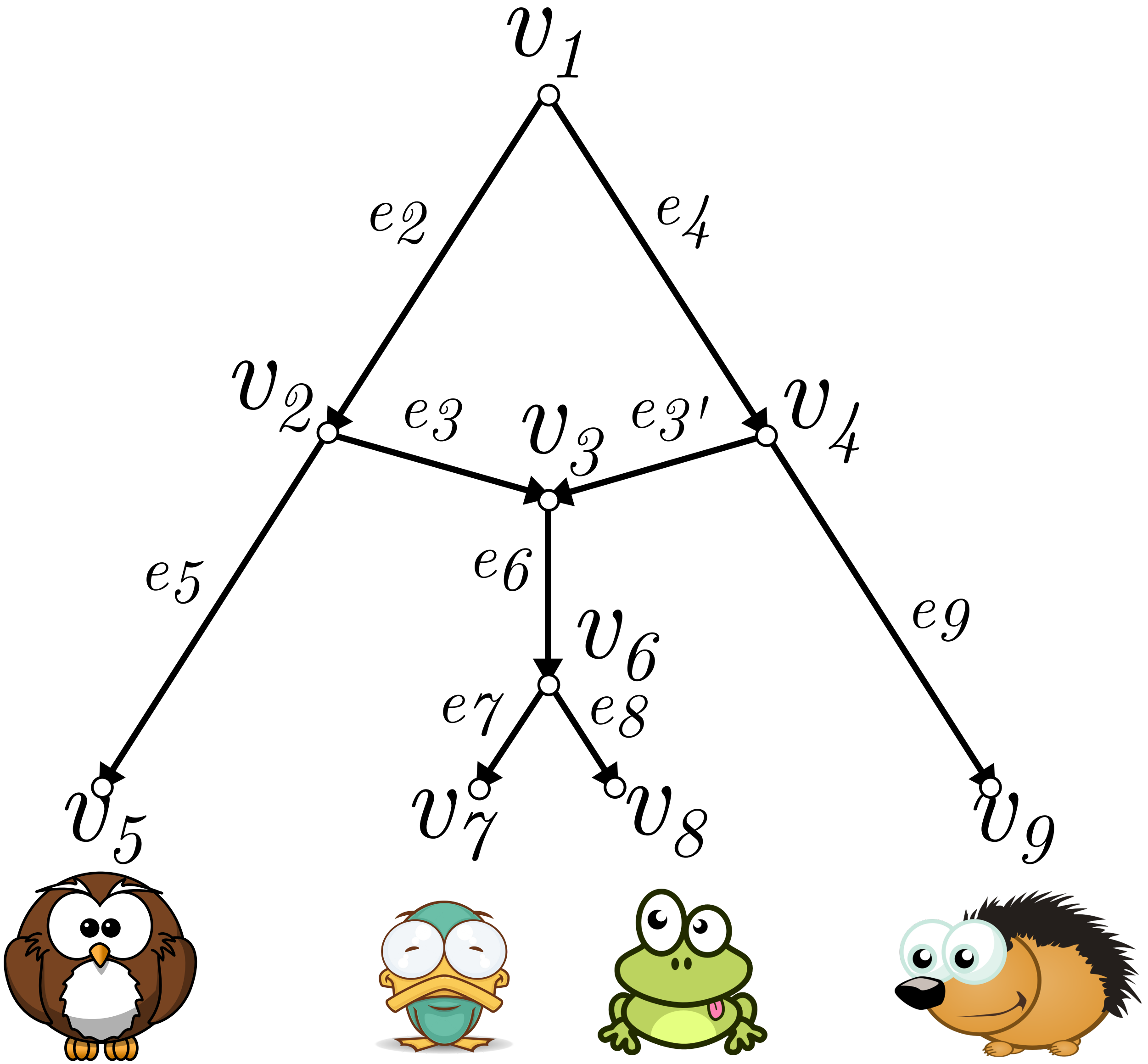
They represent evolutionary history when inheritance is from multiple ancestors – because of reticulate events, e.g:

- Hybrid speciation
- Lateral gene transfer
- **Recombination**

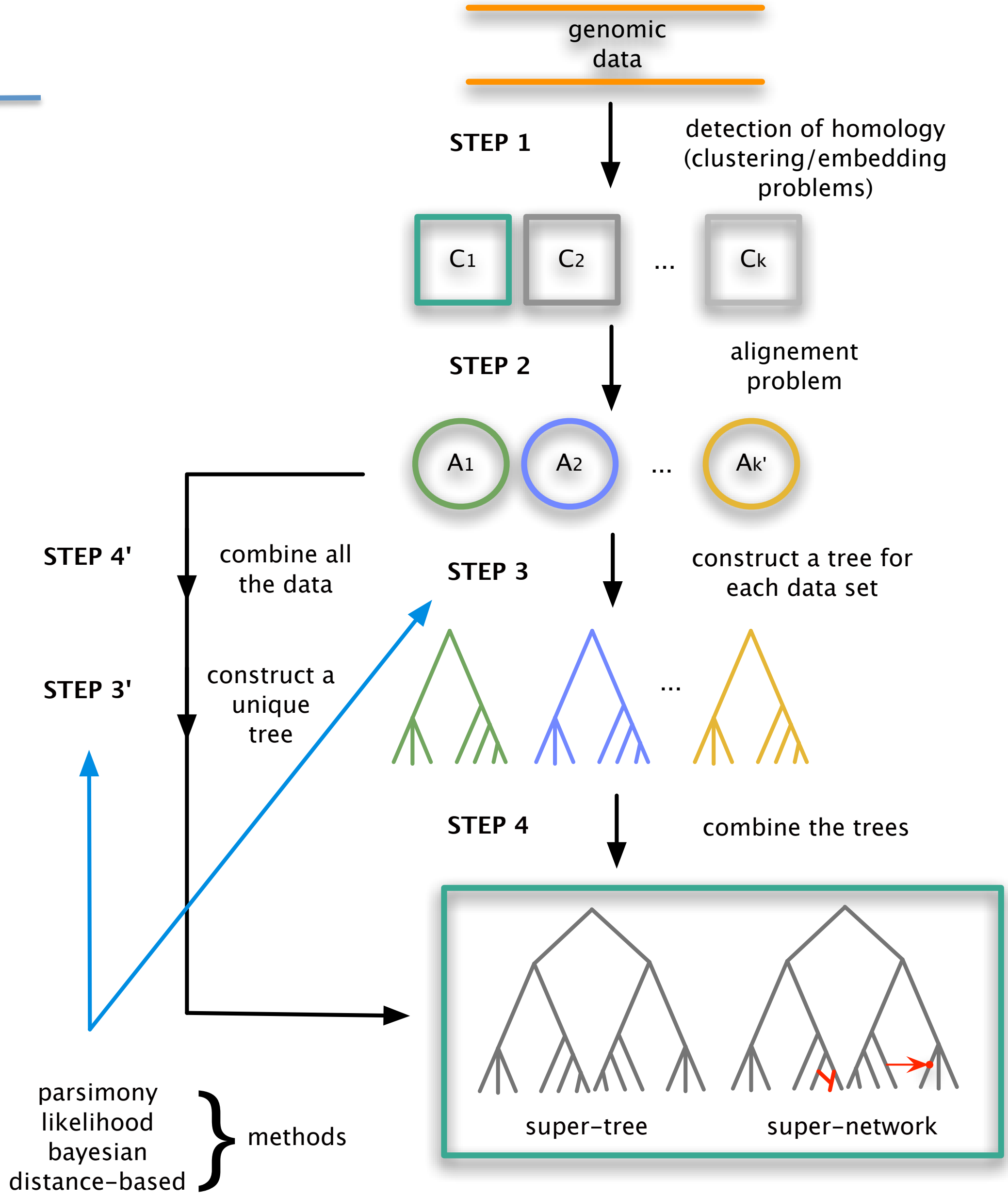


Putative phylogeny of HIV/SIV infecting primates (Bailes et al. Science 2003)

Explicit phylogenetic networks (rDAG)



Phylogenomics



genomic data

STEP 1 — detection of homology (clustering/embedding problems)

C_1 C_2 ... C_k

STEP 2 — alignement problem

A_1 A_2 ... $A_{k'}$

STEP 3 — construct a tree for each data set

STEP 4 — combine the trees

super-tree

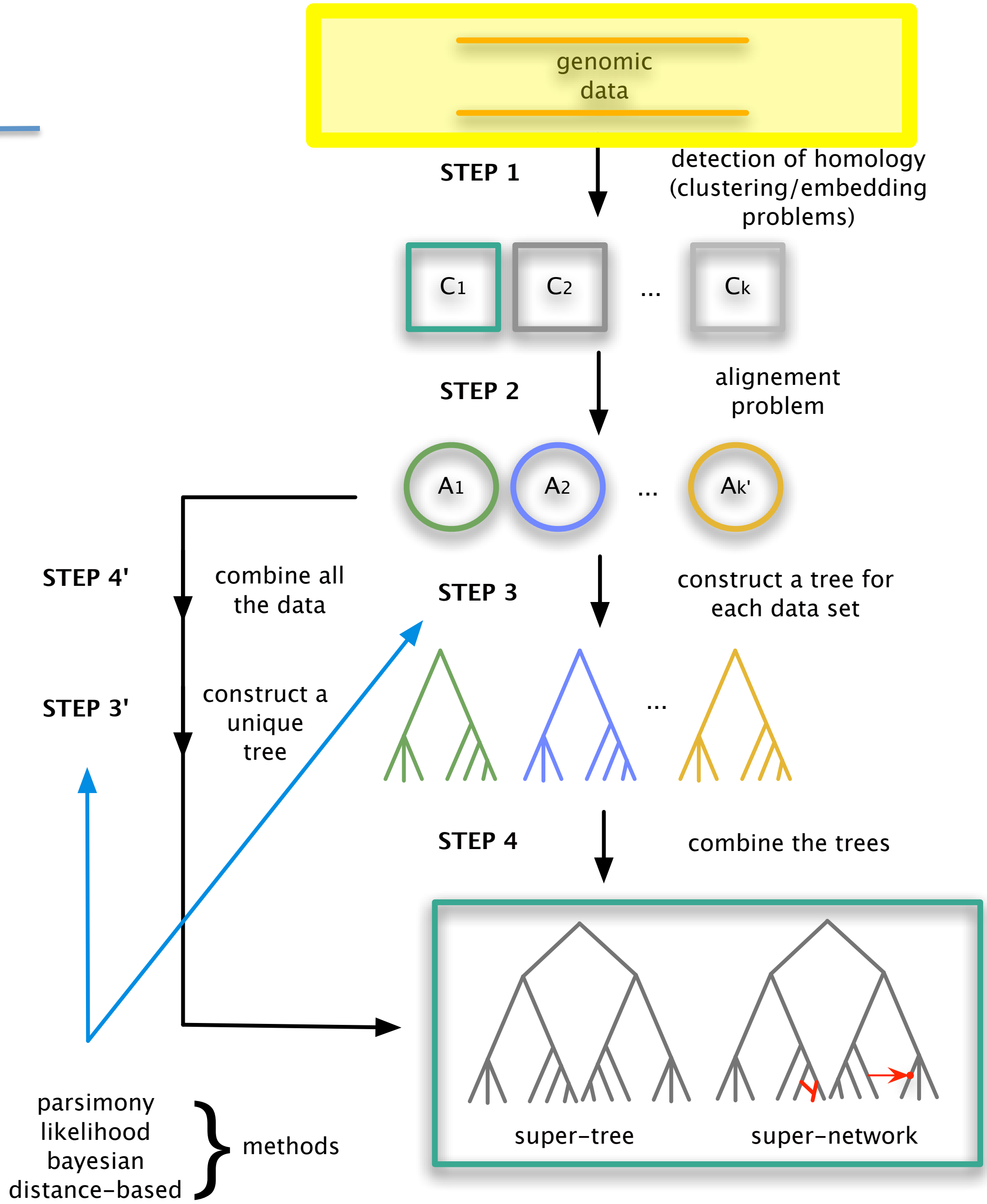
super-network

STEP 4' — combine all the data

STEP 3' — construct a unique tree

parsimony
likelihood
bayesian
distance-based
} methods

Phylogenomics



genomic data

STEP 1 detection of homology (clustering/embedding problems)

STEP 2 alignement problem

STEP 3 construct a tree for each data set

STEP 4 combine the trees

STEP 4' combine all the data

STEP 3' construct a unique tree

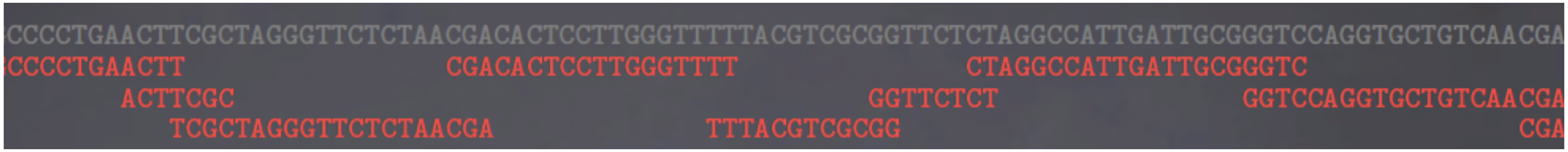
parsimony
likelihood
bayesian
distance-based } methods

Assembly for next generation sequencing –NGS

CCCCTGAACTTCGCTAGGGTTCTCTAACGACACTCCTTGGGTTTTTACGTCGCGGTTCTCTAGGCCATTGATTGCGGGTCCAGGTGCTGTCAACGA

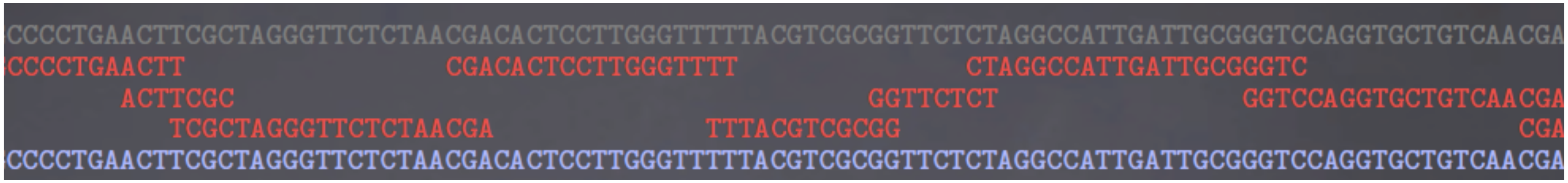
- We want to sequence a genome, a chromosome, a portion of a genome, etc.

Assembly for next generation sequencing NGS



- We want to sequence a genome, a chromosome, a portion of a genome, etc
- The portion of genomic data we want to sequence is chopped into smaller pieces, which can be easily “read”

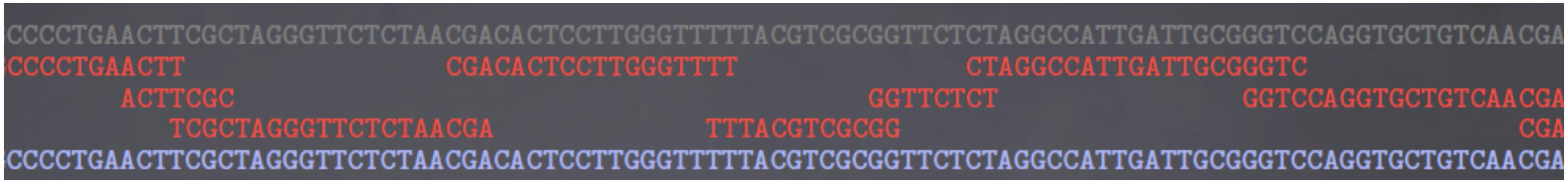
Assembly for next generation sequencing –NGS



```
CCCCTGAACTTCGCTAGGGTTCTCTAACGACACTCCTTGGGTTTTTACGTCGCGGTTCTCTAGGCCATTGATTGCGGGTCCAGGTGCTGTCAACGA
CCCCTGAACTT                CGACACTCCTTGGGTTTT              CTAGGCCATTGATTGCGGGTC
       ACTTCGC                                       GGTTCTCT               GGTCCAGGTGCTGTCAACGA
          TCGCTAGGGTTCTCTAACGA             TTTACGTCGCGG                                      CGA
CCCCTGAACTTCGCTAGGGTTCTCTAACGACACTCCTTGGGTTTTTACGTCGCGGTTCTCTAGGCCATTGATTGCGGGTCCAGGTGCTGTCAACGA
```

- We want to sequence a genome, a chromosome, a portion of a genome, etc
- The portion of genomic data we want to sequence is chopped into smaller pieces, which can be easily “read”
- The assembly step puts all the *reads* together, and we obtain the whole sequence back

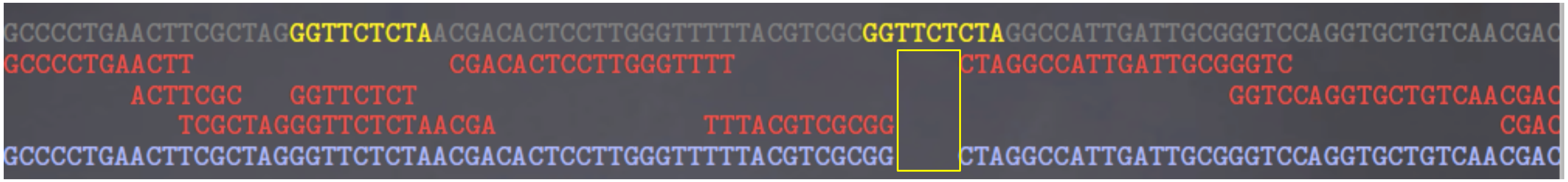
Assembly for next generation sequencing –NGS


CCCCTGAACTTCGCTAGGGTTCTCTAACGACACTCCTTGGGTTTTTACGTCGCGGTTCTCTAGGCCATTGATTGCGGGTCCAGGTGCTGTCAACGA
CCCCTGAACTT
CGACACTCCTTGGGTTTT
CTAGGCCATTGATTGCGGGTC
ACTTCGC
GGTTCTCT
GGTCCAGGTGCTGTCAACGA
TCGCTAGGGTTCTCTAACGA
TTTACGTCGCGG
CGA
CCCCTGAACTTCGCTAGGGTTCTCTAACGACACTCCTTGGGTTTTTACGTCGCGGTTCTCTAGGCCATTGATTGCGGGTCCAGGTGCTGTCAACGA

- We want to sequence a genome, a chromosome, a portion of a genome, etc
- The portion of genomic data we want to sequence is chopped into smaller pieces, which can be easily “read”
- The assembly step puts all the *reads* together, and we obtain the whole sequence back

Easier to say than to do

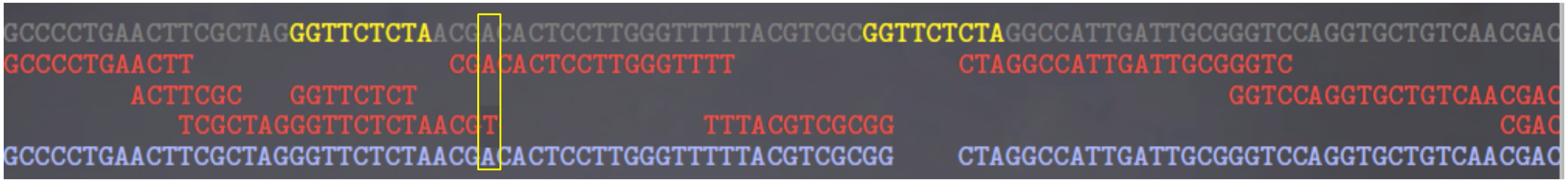
Assembly for next generation sequencing –NGS



```
GCCCCTGAACTTCGCTAGGGTTCTCTAACGACACTCCTTGGGTTTTTACGTCGCGGTTCTCTAGGCCATTGATTGCGGGTCCAGGTGCTGTCAACGAC
GCCCCTGAACTT                CGACACTCCTTGGGTTTT              CTAGGCCATTGATTGCGGGTC
        ACTTCGC   GGTTCTCT                                                   GGTCCAGGTGCTGTCAACGAC
           TCGCTAGGGTTCTCTAACGA             TTTACGTCGCGG                                      CGAC
GCCCCTGAACTTCGCTAGGGTTCTCTAACGACACTCCTTGGGTTTTTACGTCGCGG    CTAGGCCATTGATTGCGGGTCCAGGTGCTGTCAACGAC
```

- Parts of the sequence might not be covered by reads
  - ✓ high coverage

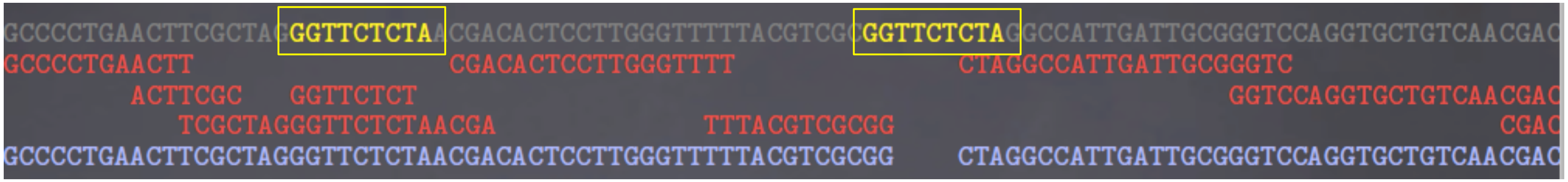
Assembly for next generation sequencing –NGS



```
GCCCCTGAACTTCGCTAGGGTTCTCTAACGACACTCCTTGGGTTTTTACGTCGCGGTTCTCTAGGCCATTGATTGCGGGTCCAGGTGCTGTCAACGAC
GCCCCTGAACTT                CGACACTCCTTGGGTTTT              CTAGGCCATTGATTGCGGGTC
        ACTTCGC   GGTTCTCT                                                   GGTCCAGGTGCTGTCAACGAC
           TCGCTAGGGTTCTCTAACGT             TTTACGTCGCGG                                      CGAC
GCCCCTGAACTTCGCTAGGGTTCTCTAACGACACTCCTTGGGTTTTTACGTCGCGG    CTAGGCCATTGATTGCGGGTCCAGGTGCTGTCAACGAC
```

- Parts of the sequence might not be covered by reads
  - ✓ high coverage
- Errors are possible
  - ✓ high coverage
  - ✓ consensus

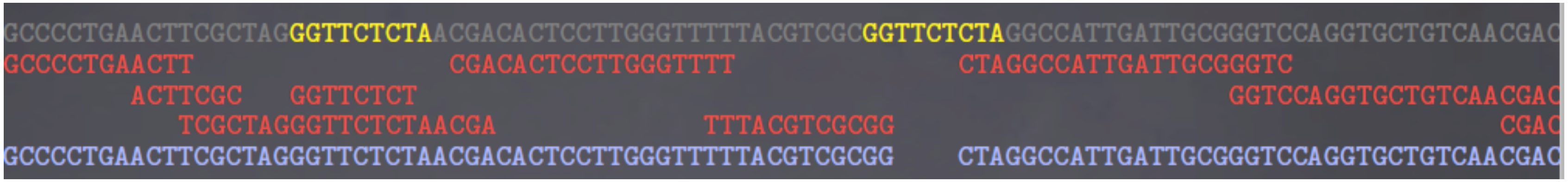
Assembly for next generation sequencing –NGS



```
GCCCCTGAACTTCGCTAGGGTTCTCTAACGACACTCCTTGGGTTTTTACGTCGCGGTTCTCTAGGCCATTGATTGCGGGTCCAGGTGCTGTCAACGAC
GCCCCTGAACTT                CGACACTCCTTGGGTTTT              CTAGGCCATTGATTGCGGGTC
        ACTTCGC   GGTTCTCT                                                   GGTCCAGGTGCTGTCAACGAC
           TCGCTAGGGTTCTCTAACGA             TTTACGTCGCGG                                      CGAC
GCCCCTGAACTTCGCTAGGGTTCTCTAACGACACTCCTTGGGTTTTTACGTCGCGG    CTAGGCCATTGATTGCGGGTCCAGGTGCTGTCAACGAC
```

- Parts of the sequence might not be covered by reads
  - ✓ high coverage
- Errors are possible
  - ✓ high coverage
  - ✓ consensus
- Repeats (common in DNA) make assembly ambiguous

Assembly for next generation sequencing –NGS

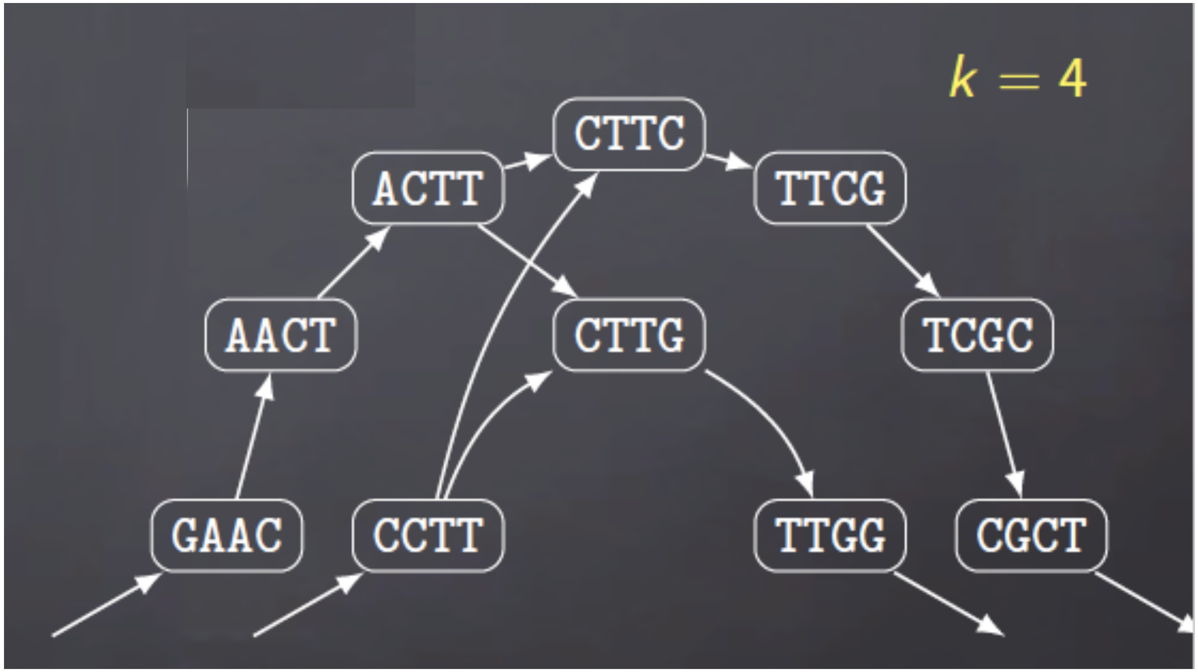


```
GCCCCTGAACTTCGCTAGGGTTCTCTAACGACACTCCTTGGGTTTTACGTCGCGGTTCTCTAGGCCATTGATTGCGGGTCCAGGTGCTGTCAACGAC
GCCCCTGAACTT                CGACACTCCTTGGGTTTT             CTAGGCCATTGATTGCGGGTC
        ACTTCGC   GGTTCTCT                                                  GGTCCAGGTGCTGTCAACGAC
           TCGCTAGGGTTCTCTAACGA            TTTACGTCGCGG                                      CGAC
GCCCCTGAACTTCGCTAGGGTTCTCTAACGACACTCCTTGGGTTTTACGTCGCGG    CTAGGCCATTGATTGCGGGTCCAGGTGCTGTCAACGAC
```

- Parts of the sequence might not be covered by reads
  - ✓ high coverage
- Errors are possible
  - ✓ high coverage
  - ✓ consensus
- Repeats (common in DNA) make assembly ambiguous

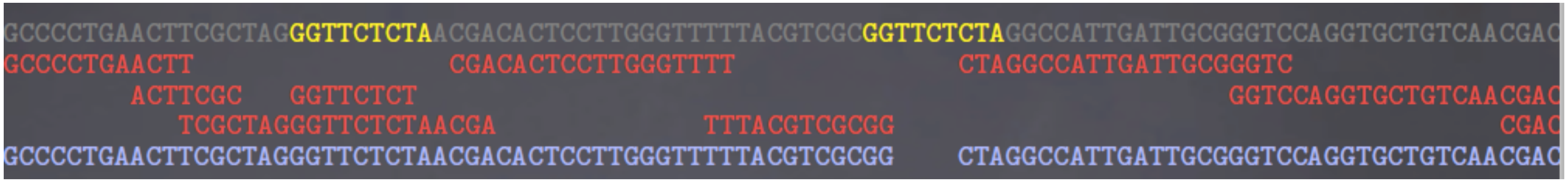
DeBruijn-graph based assembly

DeBruijn-graph based assembly



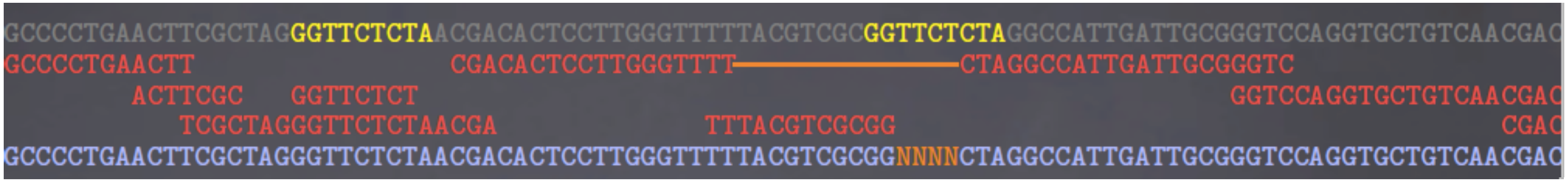
- chop all reads into “k-mers”
- builds overlap graph (“DeBruijn graph”)
- find Eulerian path

Scaffolding



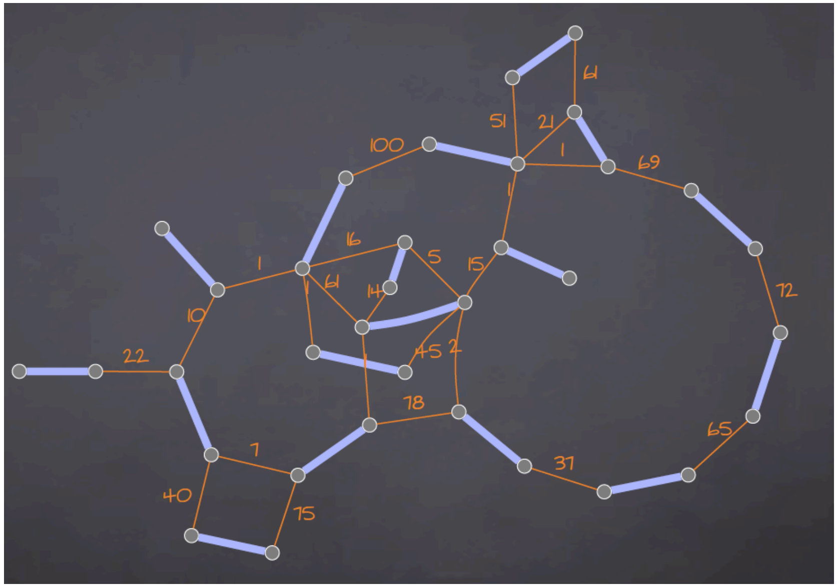
```
GCCCCTGAACTTCGCTAGGGTTCTCTAACGACACTCCTTGGGTTTTACGTCGCGGTTCTCTAGGCCATTGATTGCGGGTCCAGGTGCTGTCAACGAC
GCCCCTGAACTT                CGACACTCCTTGGGTTTT             CTAGGCCATTGATTGCGGGTC
        ACTTCGC   GGTTCTCT                                                  GGTCCAGGTGCTGTCAACGAC
           TCGCTAGGGTTCTCTAACGA            TTTACGTCGCGG                                      CGAC
GCCCCTGAACTTCGCTAGGGTTCTCTAACGACACTCCTTGGGTTTTACGTCGCGG    CTAGGCCATTGATTGCGGGTCCAGGTGCTGTCAACGAC
```

Scaffolding



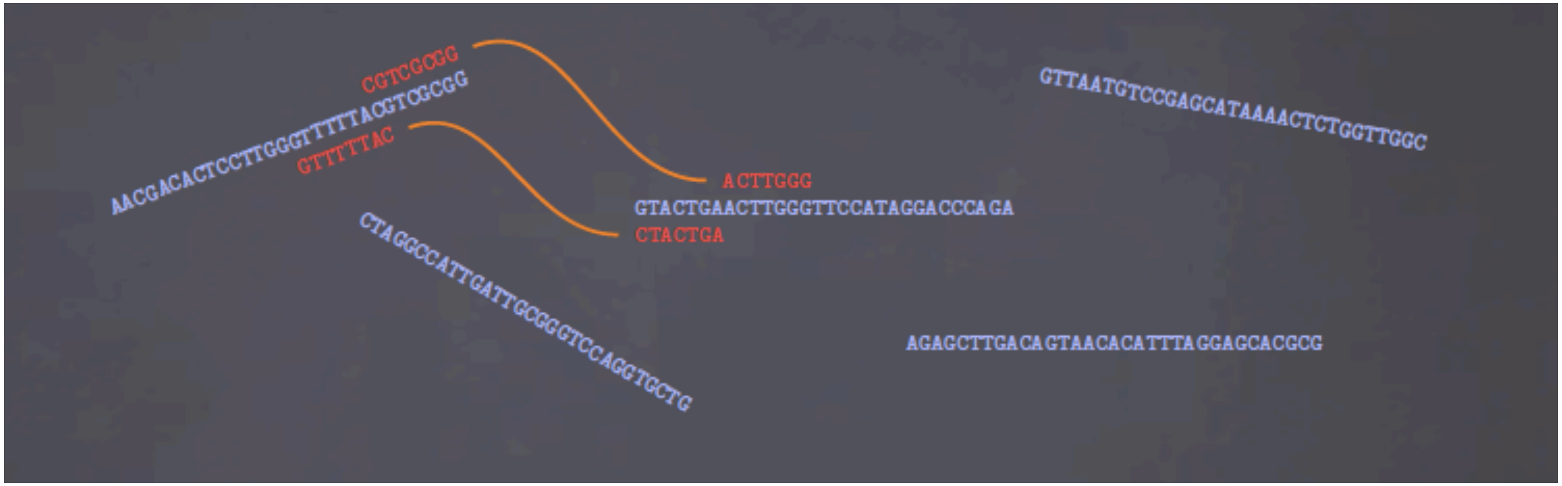
```
GCCCCTGAACTTCGCTAGGGTTCTCTAACGACACTCCTTGGGTTTTTACGTCGCGGTTCTCTAGGCCATTGATTGCGGGTCCAGGTGCTGTCAACGAC
GCCCCTGAACTT                CGACACTCCTTGGGTTTT              CTAGGCCATTGATTGCGGGTC
        ACTTCGC   GGTTCTCT                                                   GGTCCAGGTGCTGTCAACGAC
           TCGCTAGGGTTCTCTAACGA             TTTACGTCGCGG                                      CGAC
GCCCCTGAACTTCGCTAGGGTTCTCTAACGACACTCCTTGGGTTTTTACGTCGCGGNNNNCTAGGCCATTGATTGCGGGTCCAGGTGCTGTCAACGAC
```

Thanks to paired-end information, we can join *contigs* into chromosomes. This step is called scaffolding



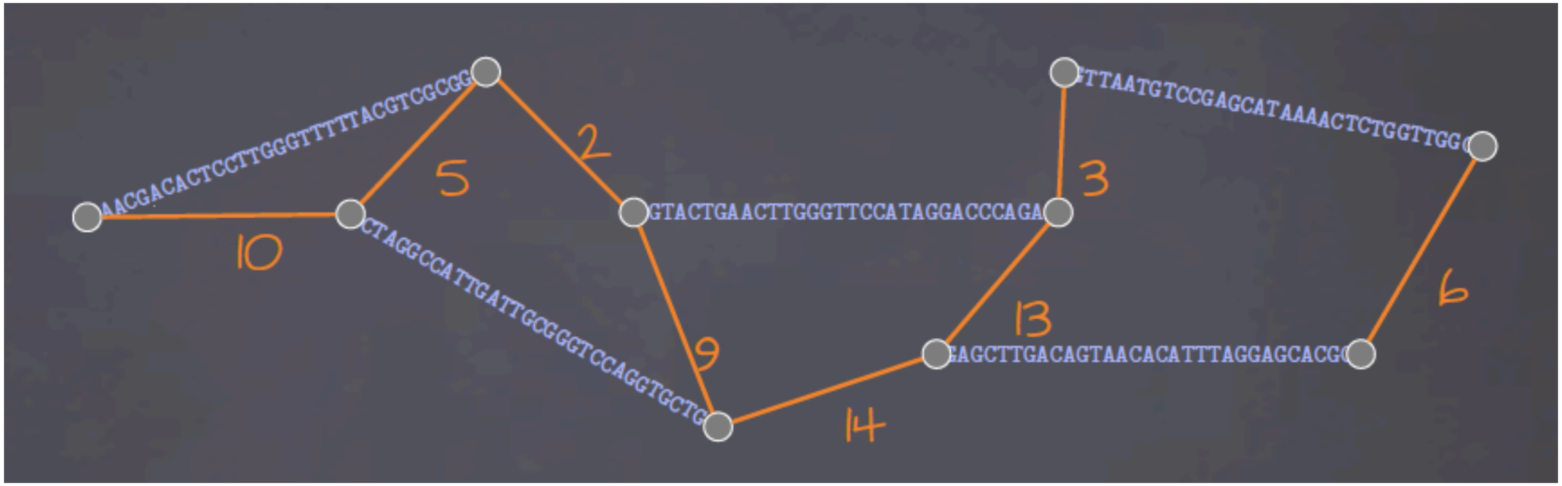
Scaffolding

- map reads into contigs


CGTCGCGG
AACGACACTCCTTGGGTTTTTACGTCGCGG
GTTTTTAC
ACTTGGG
GTACTGAACTTGGGTTCCATAGGACCCAGA
CTACTGA
GTTAATGTCCGAGCATAAAACTCTGGTTGGC
CTAGGCCATTGATTGCGGGTCCAGGTGCTG
AGAGCTTGACAGTAACACATTTAGGAGCACGCG

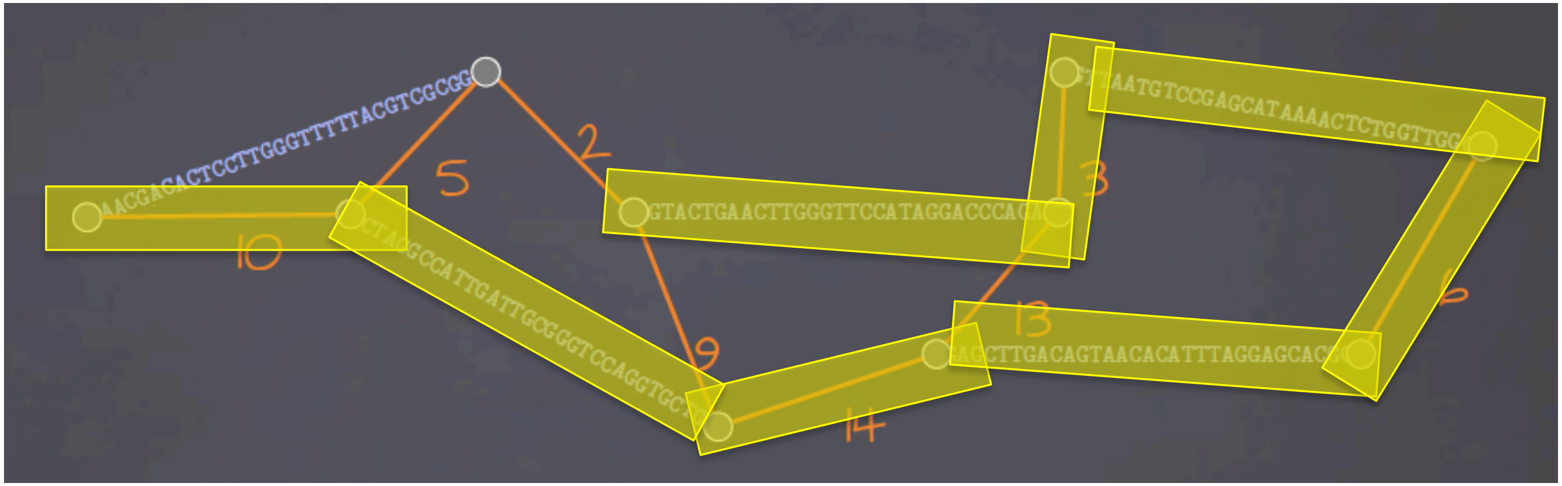
Scaffolding

- map reads into contigs
- pair contigs according to read-pairing (weighted)


AACGACACTCCTTGGGTTTTTACGTCGCGG
2
5
10
GTACTGAACTTGGGTTCCATAGGACCCAGA
TTAATGTCCGAGCATAAAACTCTGGTTGG
3
CTAGGCCATTGATTGCGGGTCCAGGTGCTG
9
6
13
AGCTTGACAGTAACACATTTAGGAGCACG
14

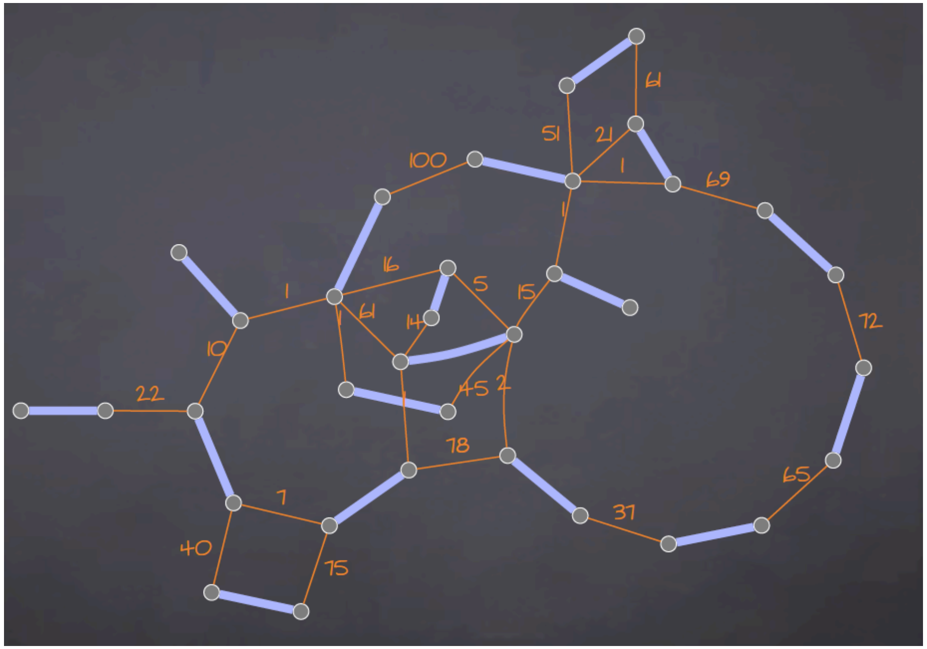
Scaffolding

- map reads into contigs
- pair contigs according to read-pairing (weighted)
- cover “scaffold graph” with (heavy) alternating paths, where each path corresponds to a chromosome


AACGACACTCCTTGGGTTTTTACGTCGCGG
5
2
10
CTAGGCCATTGATTGCGGGTCCAGGTGC
GTACTGAACTTGGGTTCCATAGGACCCACA
TTAATGTCCGAGCATAAAACTCTGGTTGG
3
9
13
AGCTTGACAGTAACACATTTAGGAGCACG
6
14

Scaffolding

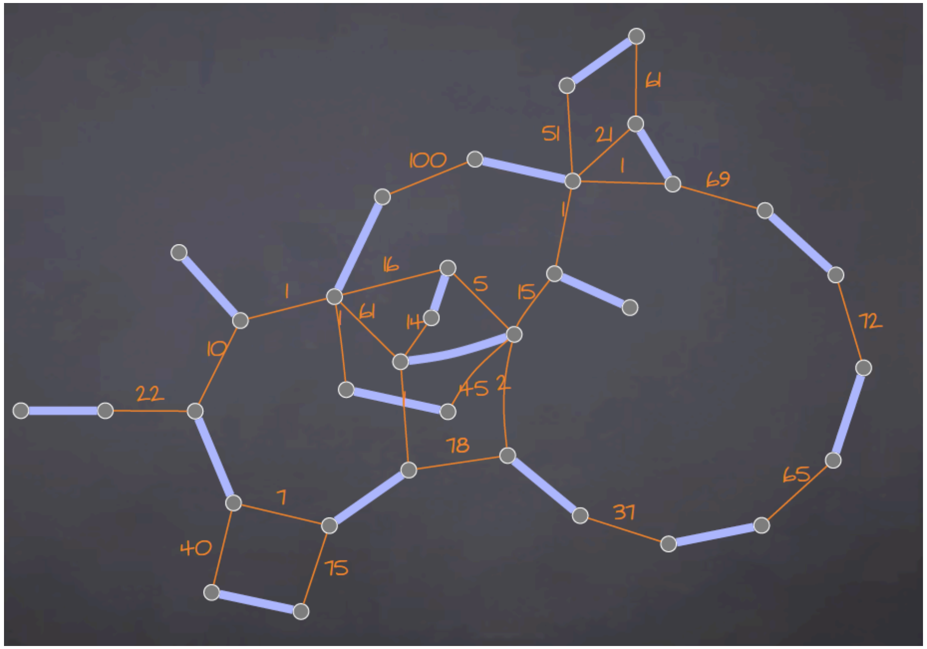
- map reads into contigs
- pair contigs according to read-pairing (weighted)
- cover “scaffold graph” with (heavy) alternating paths, where each path corresponds to a chromosome



- Np alternating paths
- Nc alternating cycles

Scaffolding

- map reads into contigs
- pair contigs according to read-pairing (weighted)
- cover "scaffold graph" with (heavy) alternating paths, where each path corresponds to a chromosome

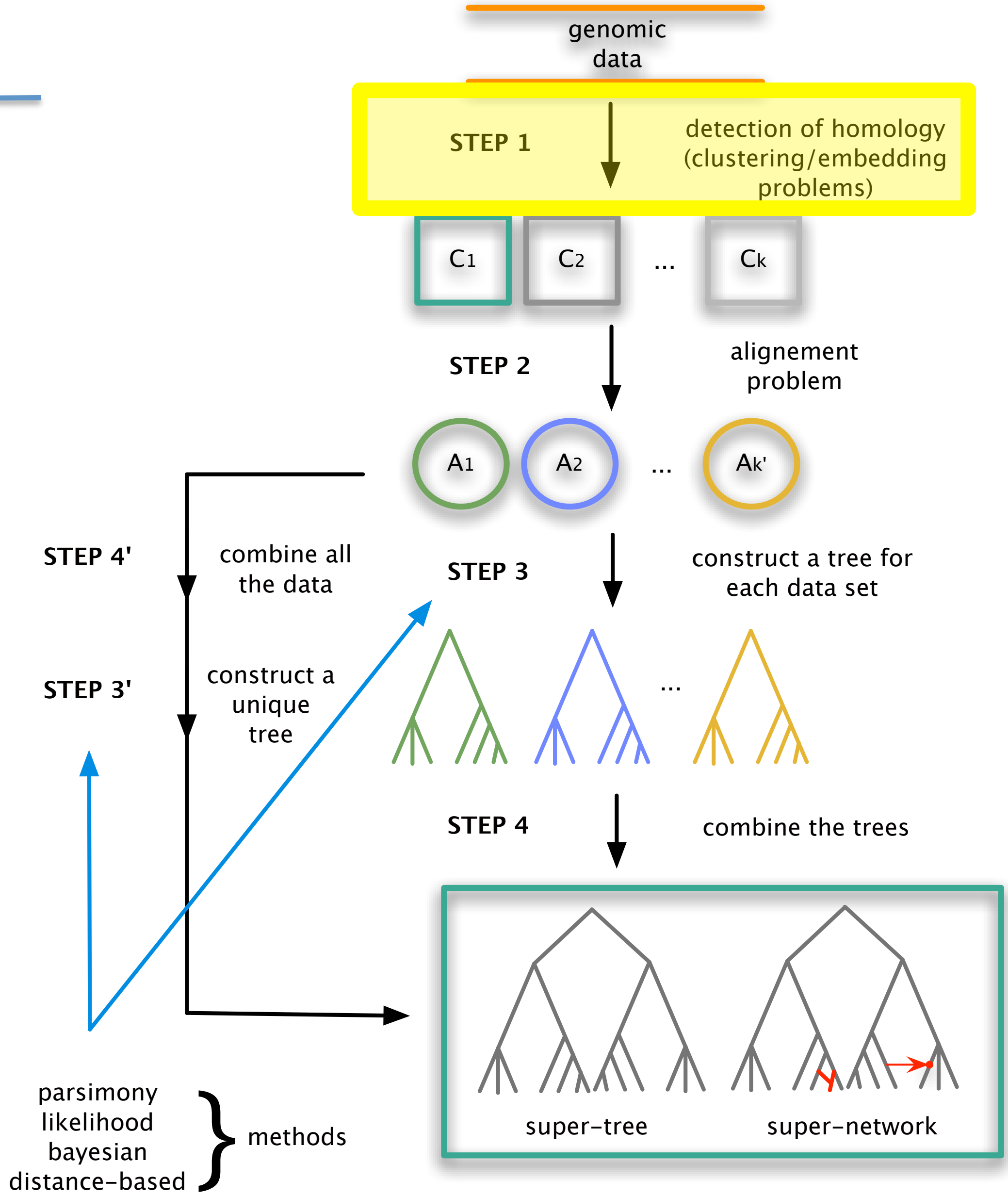


- Contig Jumps
- Multiplicities
- Linearization of solutions

- Np alternating paths
- Nc alternating cycles

Thanks to Mathias Weller for the nice blackboard-like pics

Phylogenomics



genomic data

STEP 1

detection of homology (clustering/embedding problems)

C_1 C_2 ... C_k

STEP 2

alignement problem

A_1 A_2 ... $A_{k'}$

STEP 3

construct a tree for each data set

...

STEP 4

combine the trees

super-tree

super-network

STEP 4'

combine all the data

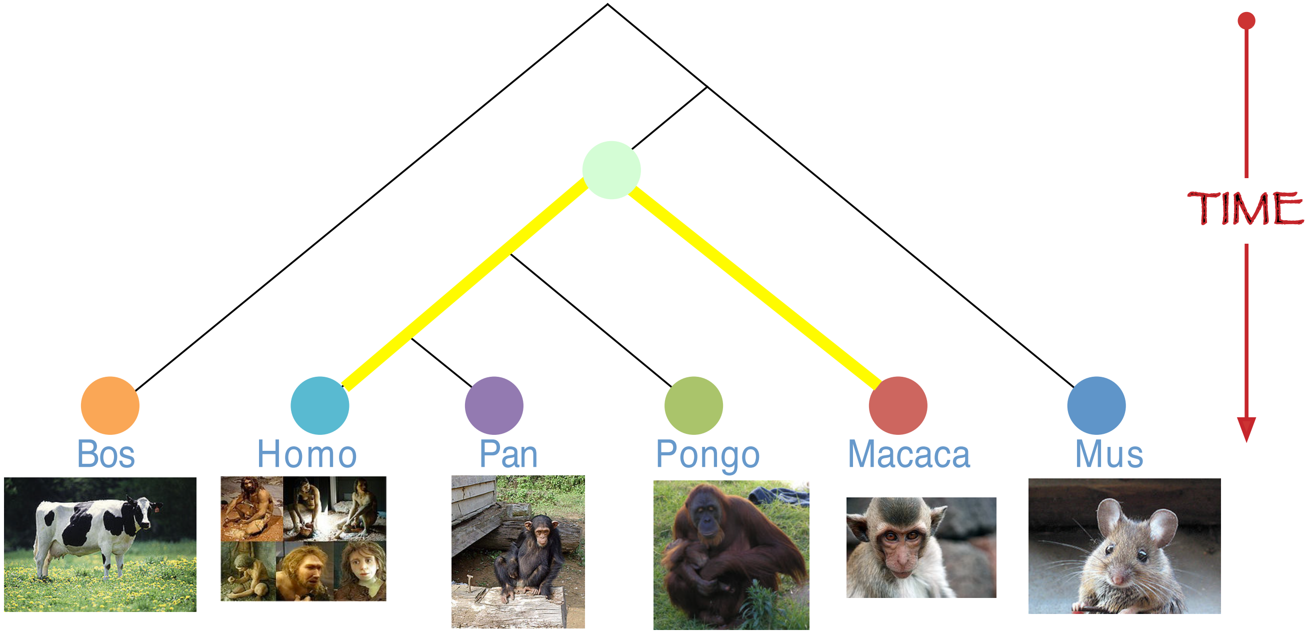
STEP 3'

construct a unique tree

parsimony
likelihood
bayesian
distance-based } methods

Which sequence to compare?

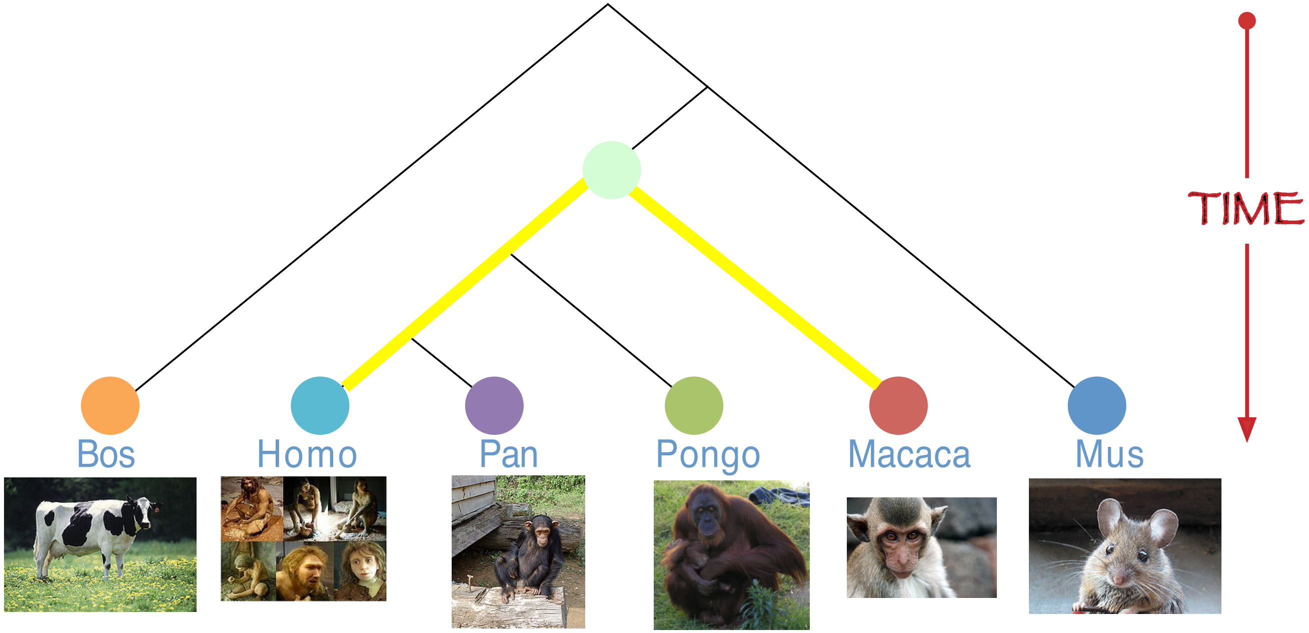
Homologous genes, ie sequences inherited in the species of interest from a common ancestor. Groups of homologous genes form *gene families*.



Which sequence to compare?

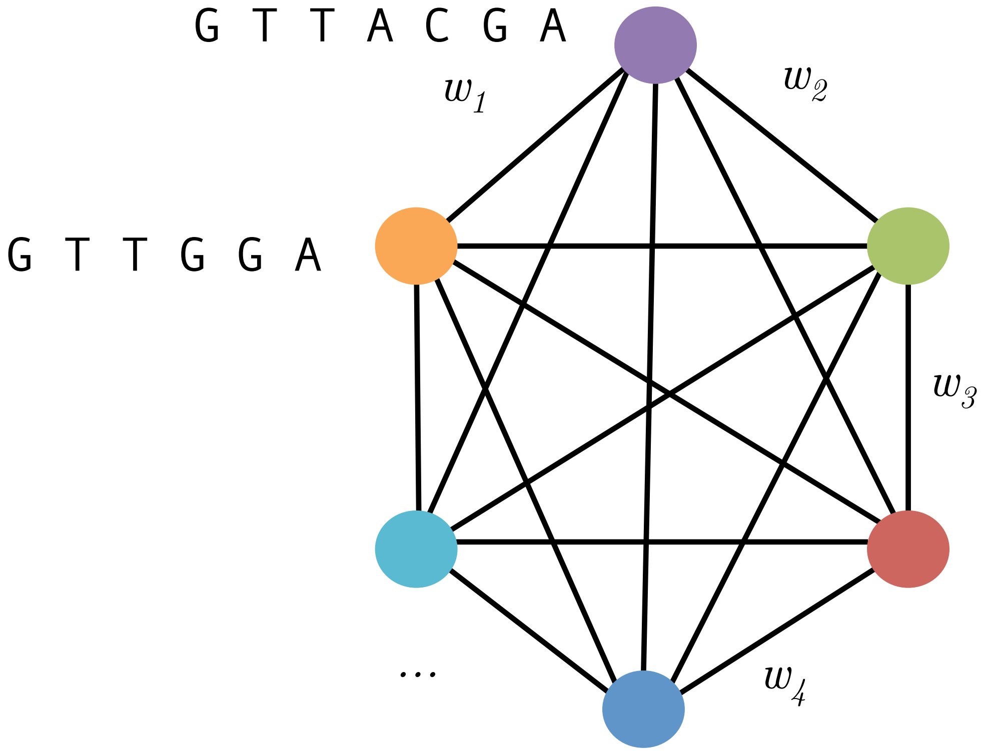
Homologous genes, ie sequences inherited in the species of interest from a common ancestor. Groups of homologous genes form *gene families*.

But sequences do not come with nice labels on them, telling us to which gene family they belong



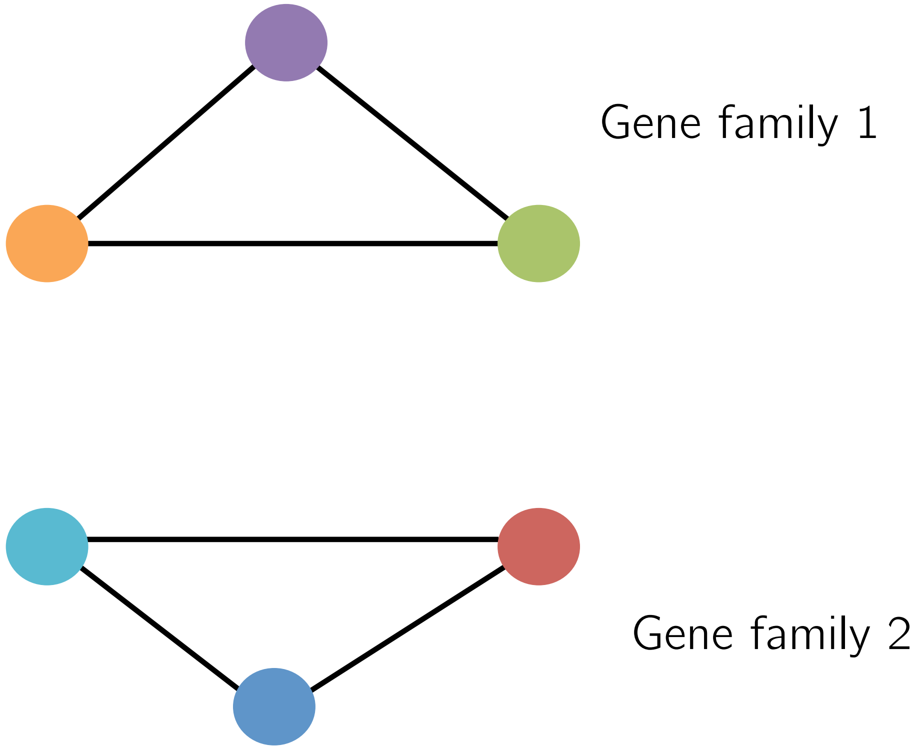
Homology inference

We put all the genes in a pool and we cluster them into gene families using *similarity measures*



Homology inference

After applying a filtering step deleting edges with weights lower than a certain threshold, we would like to get this kind of scenarios...

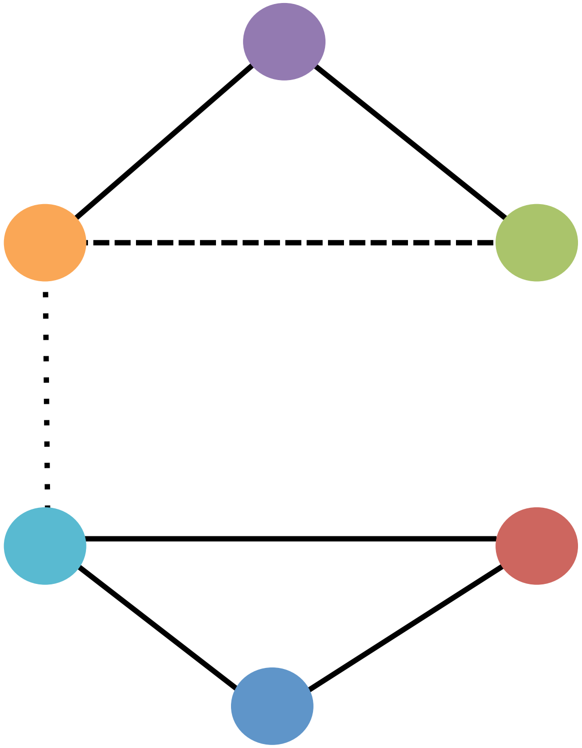


Gene family 1

Gene family 2

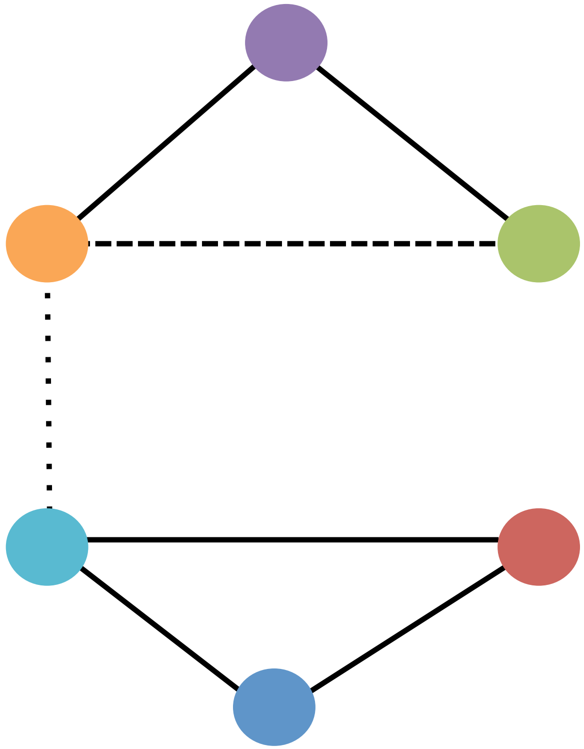
Homology inference

… but we don’t! We often get unclear scenarios where our disconnected cliques are not really cliques and not really disconnected

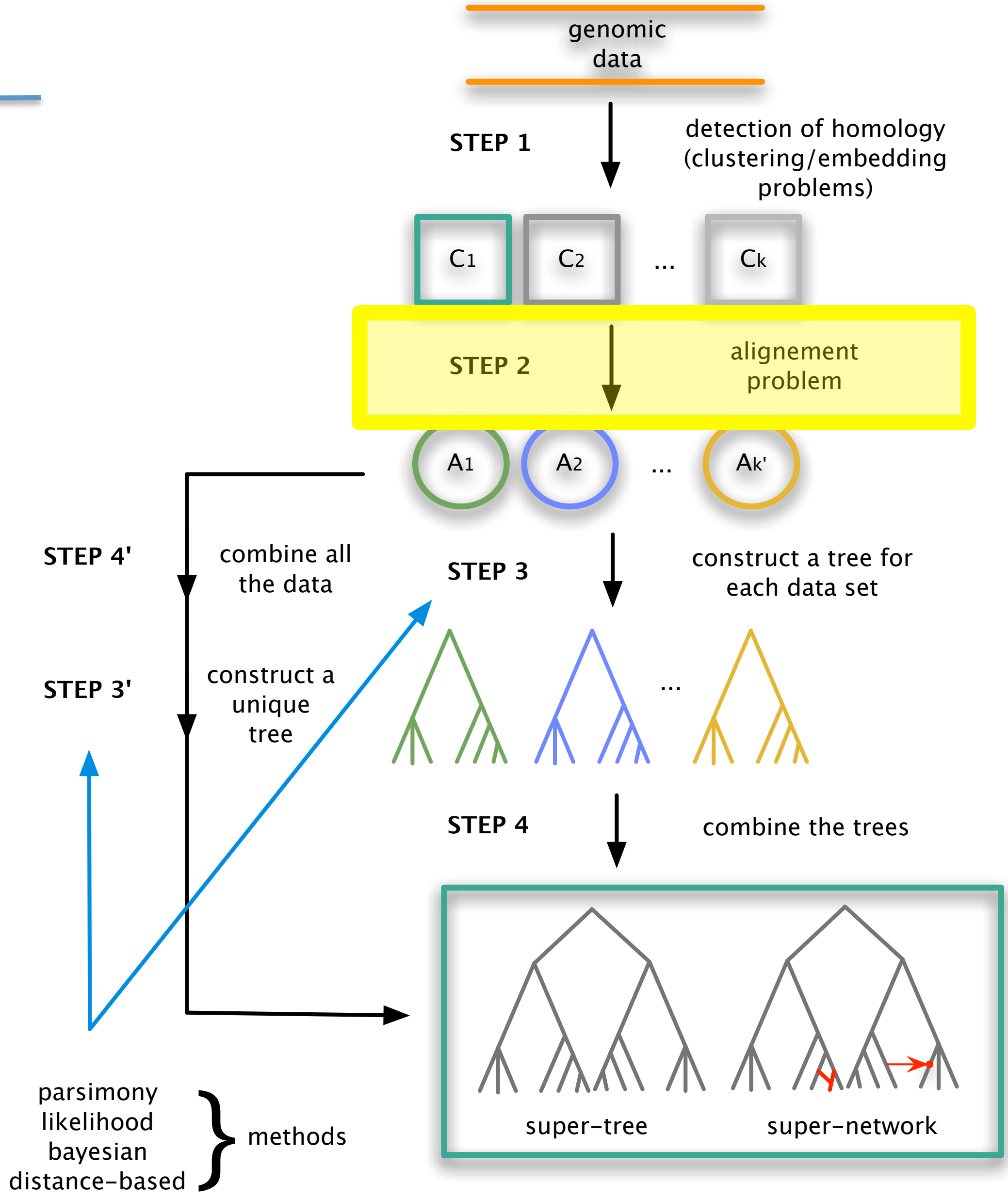


Homology inference

- cluster algorithm for graphs (e.g. MCL)
- graph editing (adding deleting edges to get disconnected cliques)



Phylogenomics



genomic data

STEP 1

detection of homology (clustering/embedding problems)

STEP 2

alignement problem

STEP 3

construct a tree for each data set

STEP 4

combine the trees

STEP 4'

combine all the data

STEP 3'

construct a unique tree

super-tree

super-network

parsimony
likelihood
bayesian
distance-based

} methods

Alignment (aka which characters to compare)

Homologous characters, ie characters inherited in the species of interest from a common ancestor. We need to *align* sequences because no only *mutations* happen on genomic sequences but also *indels* (insertions and deletions)

```
G T T A C G A
 G T T G G A
```

```
G T T A C G A
G T T - G G A

G T T A C G A
G T T G - G A

G T T A C - G A
G T T - - G G A
```

Alignment (aka which characters to compare)

Homologous characters, ie characters inherited in the species of interest from a common ancestor. We need to *align* sequences because no only *mutations* happen on genomic sequences but also *indels* (insertions and deletions)

```
G T T A C G A
 G T T G G A
```

```
G T T A C G A
G T T - G G A

G T T A C G A
G T T G - G A

G T T A C - G A
G T T - - G G A
```

- opening of the gaps
- extension of the gaps

[Affine functions are often used]

Alignment (aka which characters to compare)

Homologous characters, ie characters inherited in the species of interest from a common ancestor. We need to *align* sequences because no only *mutations* happen on genomic sequences but also *indels* (insertions and deletions)

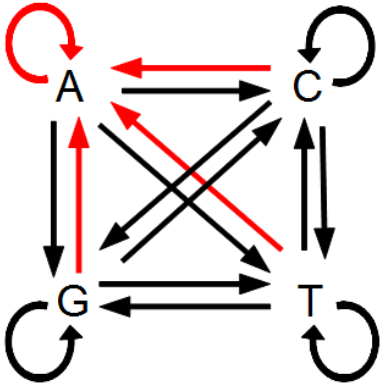
```
G T T A C G A
 G T T G G A
```

```
G T T A C G A
G T T - G G A

G T T A C G A
G T T G - G A

G T T A C - G A
G T T - - G G A
```

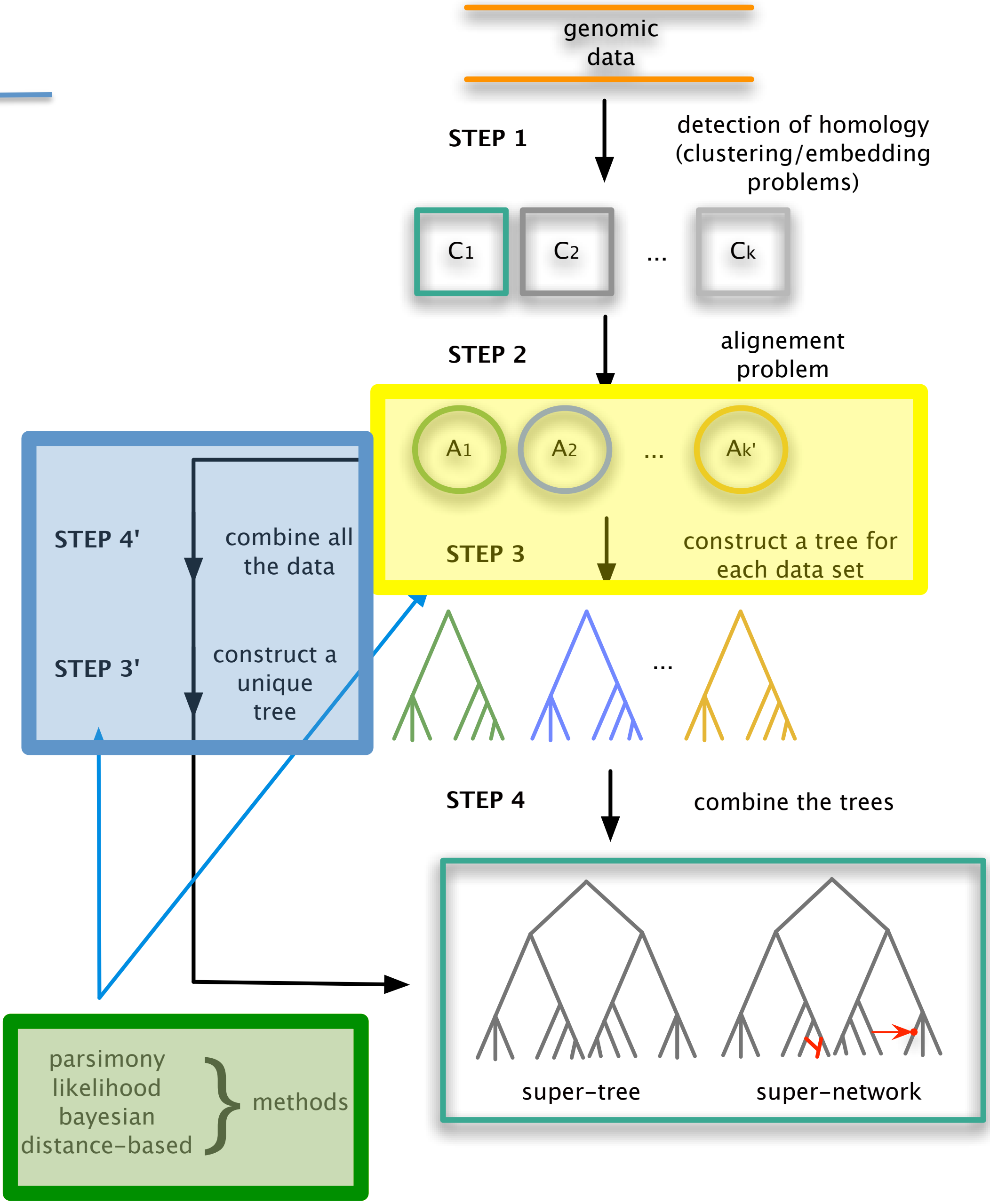
- opening of the gaps
- extension of the gaps

[Affine functions are often used]

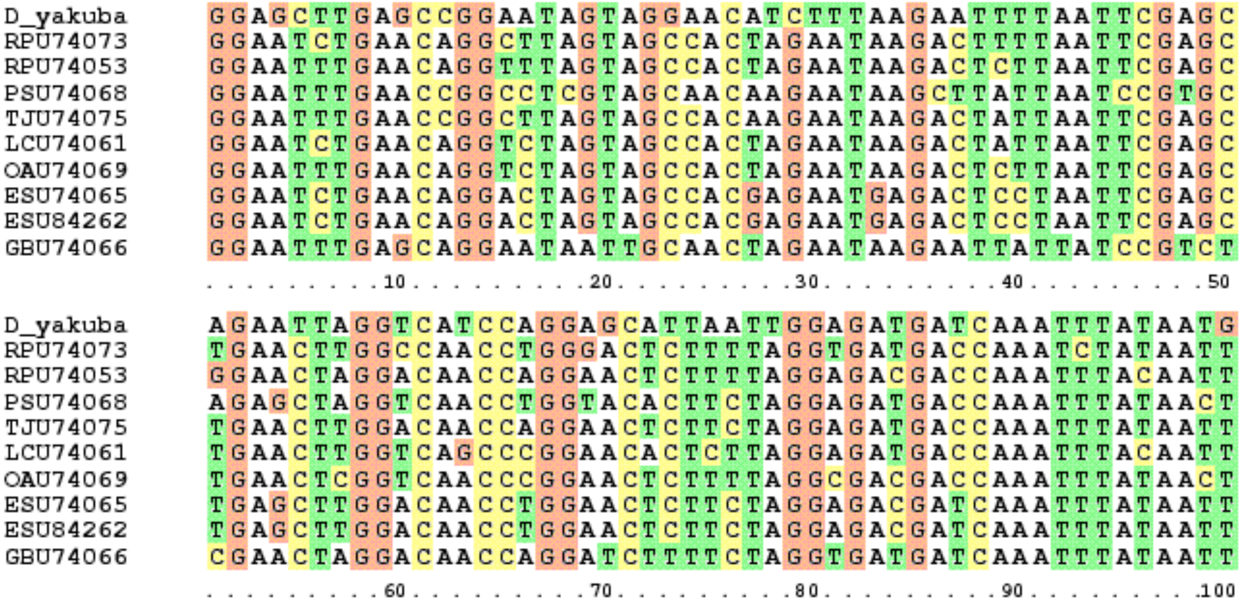
- substitutions (between nucleotides or amino acids)



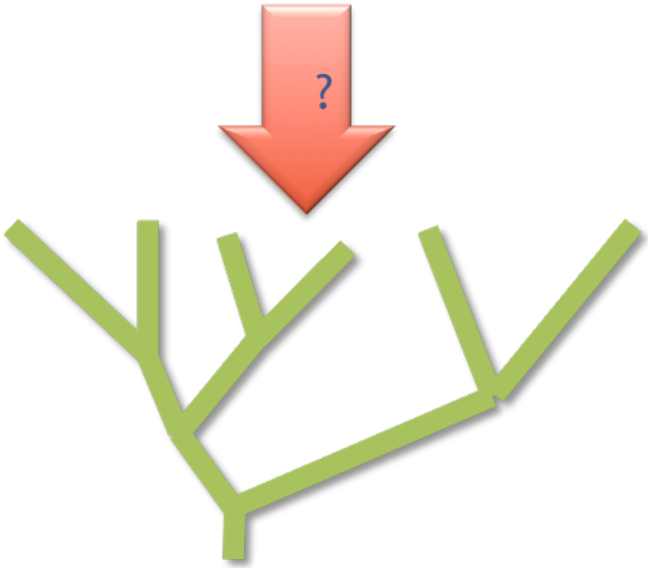
Phylogenomics



Phylogenetic inference



```
D_yakuba    GGAGCTTGAGCCGGAATAGTAGGAACATCTTTAAGAATTTTAATTCGAGC
RPU74073    GGAATCTGAACAGGCTTAGTAGCCACTAGAATAAGACTTTTAATTCGAGC
RPU74053    GGAATTTGAACAGGTTTAGTAGCCACTAGAATAAGACTCTTAATTCGAGC
PSU74068    GGAATTTGAACCGGCCTCGTAGCAACAAGAATAAGCTTATTAATCCGTGC
TJU74075    GGAATTTGAACCGGCTTAGTAGCCACAAGAATAAGACTATTAATTCGAGC
LCU74061    GGAATCTGAACAGGTCTAGTAGCCACTAGAATAAGACTATTAATTCGAGC
OAU74069    GGAATTTGAACAGGTCTAGTAGCCACTAGAATAAGACTCTTAATTCGAGC
ESU74065    GGAATCTGAACAGGACTAGTAGCCACGAGAATGAGACTCCTAATTCGAGC
ESU84262    GGAATCTGAACAGGACTAGTAGCCACGAGAATGAGACTCCTAATTCGAGC
GBU74066    GGAATTTGAGCAGGAATAATTGCAACTAGAATAAGAATTATTATCCGTCT
            .........10.........20.........30.........40.........50

D_yakuba    AGAATTAGGTCATCCAGGAGCATTAATTGGAGATGATCAAATTTATAATG
RPU74073    TGAACTTGGCCAACCTGGGACTCTTTTAGGTGATGACCAAATCTATAATT
RPU74053    GGAACTAGGACAACCAGGAACTCTTTTAGGAGACGACCAAATTTACAATT
PSU74068    AGAGCTAGGTCAACCTGGTACACTTCTAGGAGATGACCAAATTTATAACT
TJU74075    TGAACTTGGACAACCAGGAACTCTTCTAGGAGATGACCAAATTTATAATT
LCU74061    TGAACTTGGTCAGCCCGGAACACTCTTAGGAGATGACCAAATTTACAATT
OAU74069    TGAACTCGGTCAACCCGGAACTCTTTTAGGCGACGACCAAATTTATAACT
ESU74065    TGAGCTTGGACAACCTGGAACTCTTCTAGGAGACGATCAAATTTATAATT
ESU84262    TGAGCTTGGACAACCTGGAACTCTTCTAGGAGACGATCAAATTTATAATT
GBU74066    CGAACTAGGACAACCAGGATCTTTTCTAGGTGATGATCAAATTTATAATT
            .........60.........70.........80.........90.........100
```



Reconstructing phylogenies

- distance-based methods, which use pairwise distances to quantify the amount of evolution separating species

- character-based methods, which retrieve similarities comparing the states taken by species at different characters:
    - parsimony methods
    - likelihood methods
    - bayesian methods

Reconstructing phylogenies

- **distance-based methods**, which use pairwise distances to quantify the amount of evolution separating species

- character-based methods, which retrieve similarities comparing the states taken by species at different characters:

    - parsimony methods
    - likelihood methods
    - bayesian methods

Distance estimation

First thing to do is to define distances between genomic sequences. The usual way (no genome rearrangement here) is to compute them from the alignments

```
G T T A C G A C
G T T - G G A A
```

Distance estimation

First thing to do is to define distances between genomic sequences. The usual way (no genome rearrangement here) is to compute them from the alignments, after having removed the gaps

```
G T T C G A C
G T T G G A A
```

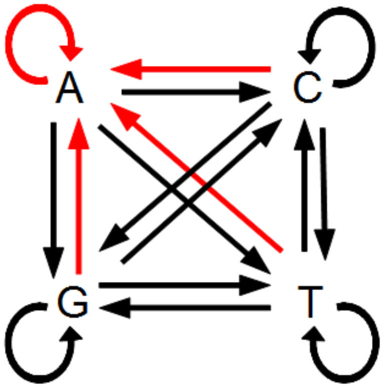
- Hamming distance 1+1

Distance estimation

First thing to do is to define distances between genomic sequences. The usual way (no genome rearrangement here) is to compute them from the alignments, after having removed the gaps

```
G T T C G A C
G T T G G A A
```

1. Hamming distance: 1+1
2. Accounting for the biology:
   - $C_{C\text{->}G} + C_{C\text{->}A}$

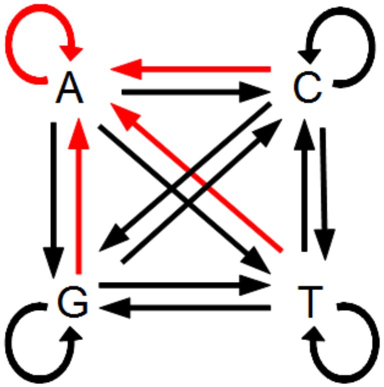


Distance estimation

First thing to do is to define distances between genomic sequences. The usual way (no genome rearrangement here) is to compute them from the alignments, after having removed the gaps

```
G T T C G A C
G T T G G A A
```

1. Hamming distance: 1+1
2. Accounting for the biology:
   - $C_{C\text{->}G} + C_{C\text{->}A}$
   - accounting for multiple, parallel, convergent, coincidental and back substitutions

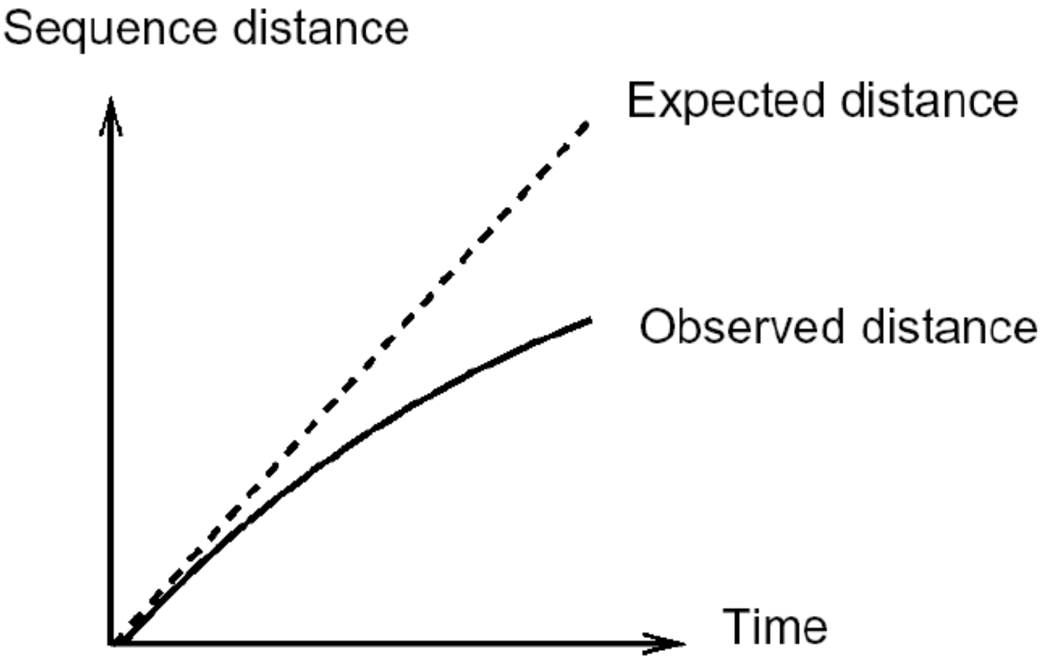


```
G T T C G A C
G T A G G A A
G T T G G A A
```

3 more substitutions!

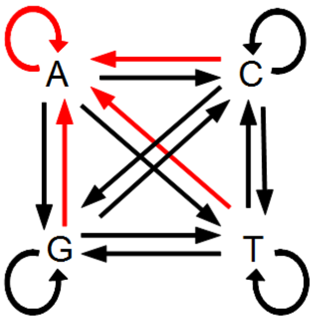
Distance estimation

We correct the Hamming distance (d_0) using a substitution model (a probabilistic model of sequence evolution). The corrected distance aims at estimating the true distance.



G T T C G A C
G T T G G A A

+



G T T C G A C
G T T G G A A

Observed distance: $d_0(t \rightarrow \infty) = 3/4$

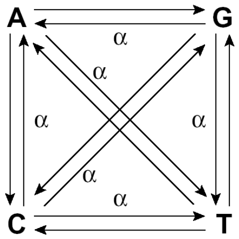
Corrected distance: $d = -\frac{3}{4} \ln(1 - \frac{4}{3} d_0)$

Examples of substitution models

Aka probabilistic models of sequence evolution

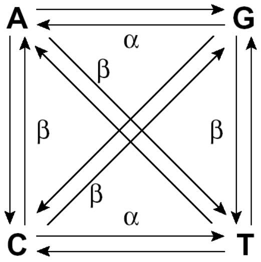
JC

$$Q = \begin{pmatrix} -3\alpha & \alpha & \alpha & \alpha \\ \alpha & -3\alpha & \alpha & \alpha \\ \alpha & \alpha & -3\alpha & \alpha \\ \alpha & \alpha & \alpha & -3\alpha \end{pmatrix}$$



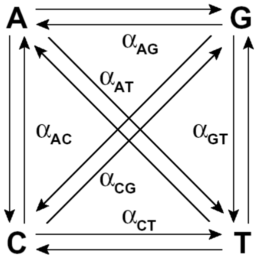
K2P

$$Q = \begin{pmatrix} -\alpha-2\beta & \beta & \alpha & \beta \\ \beta & -\alpha-2\beta & \beta & \alpha \\ \alpha & \beta & -\alpha-2\beta & \beta \\ \beta & \alpha & \beta & -\alpha-2\beta \end{pmatrix}$$



GTR

$$Q = \begin{pmatrix} \lambda_A & \pi_C R_{AC} & \pi_G R_{AG} & \pi_T R_{AT} \\ \pi_A R_{AC} & \lambda_C & \pi_G R_{CG} & \pi_T R_{CT} \\ \pi_A R_{AG} & \pi_C R_{CG} & \lambda_G & \pi_T R_{GT} \\ \pi_A R_{AT} & \pi_C R_{CT} & \pi_G R_{GT} & \lambda_T \end{pmatrix}$$



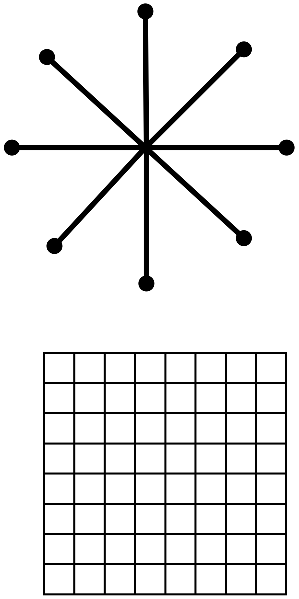
Distance methods

- Estimate pairwise distances between sequences (mean number of substitutions per site, see previous slides)
- Reconstruct a tree that corresponds well to the estimated distances

Distance methods

- Estimate pairwise distances between sequences (mean number of substitutions per site)
- Reconstruct a tree that corresponds well to the estimated distances

- An agglomerative algorithm: Neighbor Joining (NJ)

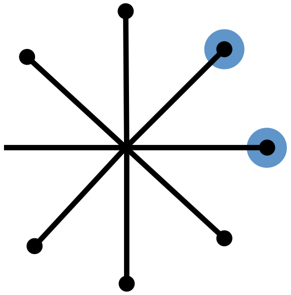


Thanks to Fabio Pardi for the nice pics

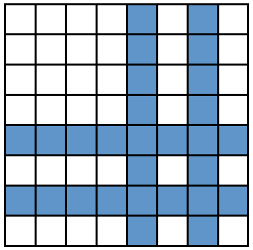
Distance methods

- Estimate pairwise distances between sequences (mean number of substitutions per site)
- Reconstruct a tree that corresponds well to the estimated distances

- An agglomerative algorithm: Neighbor Joining (NJ)



Selection step: which nodes to choose

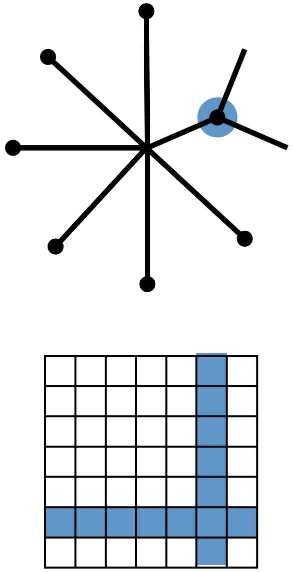


Reduction step: how to update the distances

Distance methods

- Estimate pairwise distances between sequences (mean number of substitutions per site)
- Reconstruct a tree that corresponds well to the estimated distances

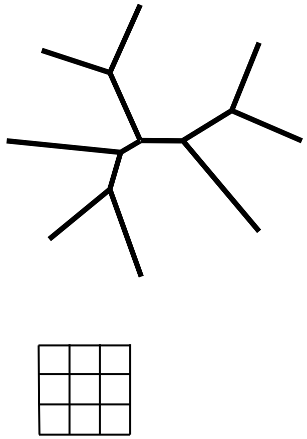
- An agglomerative algorithm: Neighbor Joining (NJ)



Distance methods

- Estimate pairwise distances between sequences (mean number of substitutions per site)
- Reconstruct a tree that corresponds well to the estimated distances

- An agglomerative algorithm: Neighbor Joining (NJ)



until the tree
is binary

UPGMA (1958), WPGMA (1973), ADDTREE (1977), NJ (1987), BIONJ (1997), UNJ (1997), MVR (2000), Weighbor (2000)

Distance methods

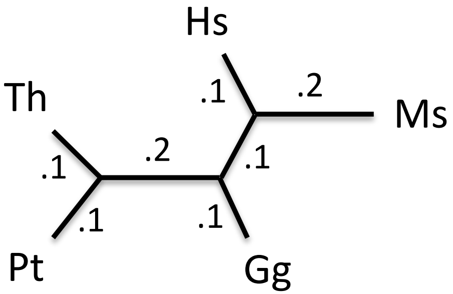
- Estimate pairwise distances between sequences (mean number of substitutions per site)
- Reconstruct a tree that corresponds well to the estimated distances

- An agglomerative algorithm: Neighbor Joining (NJ)

- Optimization principles
  - Least Squares (LS): given the estimated distances δ_{ij}, find T s.t $\delta_{ij} \approx d_{ij}^T$ where d_{ij}^T are the distances between the leaves of T

	Th	Pt	Gg	Ms	Hs
Th	0	.23	.38	.61	.50
Pt		0	.42	.57	.48
Gg			0	.41	.29
Ms				0	.30
Hs					0

(δ_{ij})



	Th	Pt	Gg	Ms	Hs
Th	0	.2	.4	.6	.5
Pt		0	.4	.6	.5
Gg			0	.4	.3
Ms				0	.3
Hs					0

$T \longrightarrow (d_{ij}^T)$

Distance methods

- Estimate pairwise distances between sequences (mean number of substitutions per site)
- Reconstruct a tree that corresponds well to the estimated distances

- An agglomerative algorithm: Neighbor Joining (NJ)

- Optimization principles
  - Least Squares (LS): given the estimated distances δ_{ij}, find T s.t $\delta_{ij} \approx d_{ij}^T$ where d_{ij}^T are the distances between the leaves of T

$$\min_T \sum_{i<j} w_{ij}(d_{ij}^T - \delta_{ij})$$

OLS when $w_{ij}=1$
WLS otherwise, where w_{ij} gives the confidence we have in the distance entry δ_{ij}

Distance methods

- Estimate pairwise distances between sequences (mean number of substitutions per site)
- Reconstruct a tree that corresponds well to the estimated distances

- An agglomerative algorithm: Neighbor Joining (NJ)

- Optimization principles
  - Least Squares (LS): given the estimated distances δ_{ij}, find T s.t $\delta_{ij} \approx d_{ij}^T$ where d_{ij}^T are the distances between the leaves of T

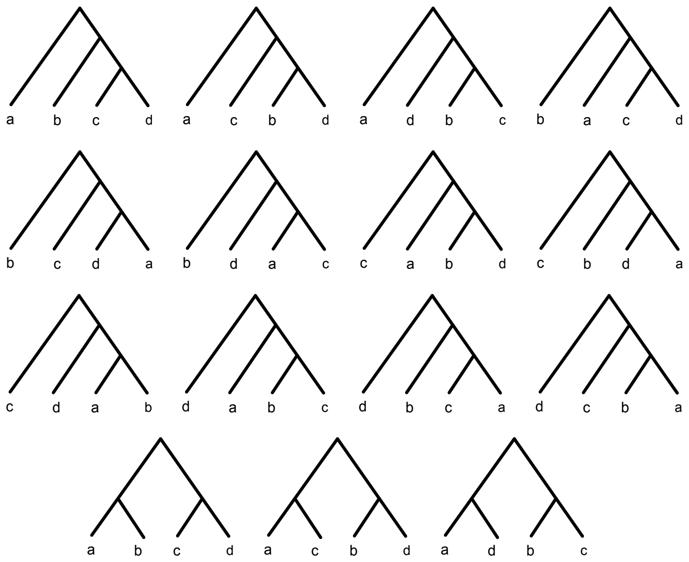
$$\min_T \sum_{i<j} w_{ij}(d_{ij}^T - \delta_{ij})$$

SMALL PROBLEM

OLS when $w_{ij}=1$ $O(n^2)$ $O(n^4)/O(n^2)$

WLS otherwise, where w_{ij} gives the confidence we have in the distance entry δ_{ij}

Distance methods



(2n-3)!!
trees

$$\min_{T} \sum_{i<j} w_{ij}(d_{ij}^{T} - \delta_{ij})$$

BIG
PROBLEM
NP-hard

Heuristics:

- Sequential insertion
- Star decomposition
  - Hill-climbing

Distance methods

- Estimate pairwise distances between sequences (mean number of substitutions per site)
- Reconstruct a tree that corresponds well to the estimated distances

- An agglomerative algorithm: Neighbor Joining (NJ)

- Optimization principles
  - Least Squares (LS): given the estimated distances δ_{ij}, find T s.t $\delta_{ij} \approx d_{ij}^T$ where d_{ij}^T are the distances between the leaves of T
  - Balanced Minimum Evolution (BME): $\min_T \sum_{k \in E(T)} b_k$

$$q(b) = \sum_{i<j} w_{ij}(d_{ij}^T - \delta_{ij})^2$$

BIG PROBLEM NP-hard

Heuristics
(such as NJ)

Reconstructing phylogenies

- distance-based methods, which use pairwise distances to quantify the amount of evolution separating species

- **character-based methods**, which retrieve similarities comparing the states taken by species at different characters:

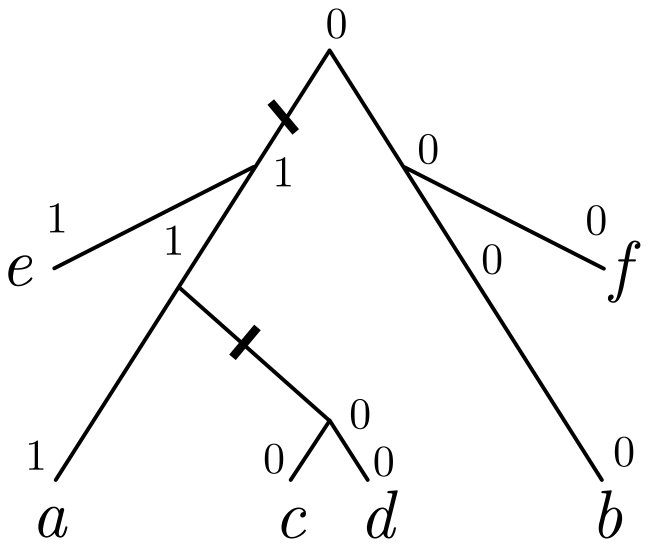
  - parsimony methods
  - likelihood methods
  - bayesian methods

Parsimony methods

- The main hypothesis of parsimony sequence-based methods is that character changes are not frequent and thus the phylogenies that best explain the data are those requiring the fewest evolutionary changes
- Each character can be analyzed independently from the others

$$PS(T|A) = \sum_{j=1}^{m} w_j PS(T|a_{\star,j})$$

$$\begin{array}{c|c|cccc}
a & 1 & 0 & 1 & 0 & 0 \\
b & 0 & 0 & 1 & 1 & 1 \\
c & 0 & 1 & 1 & 0 & 1 \\
d & 0 & 0 & 1 & 1 & 0 \\
e & 1 & 1 & 1 & 0 & 1 \\
f & 0 & 1 & 1 & 1 & 1
\end{array}$$

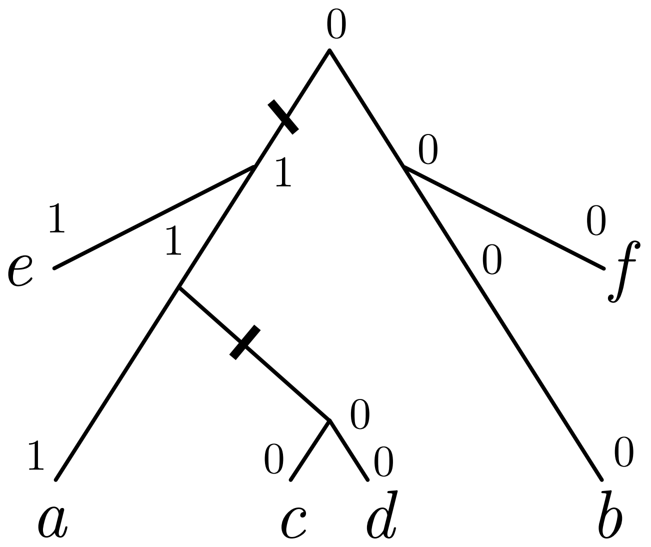


Parsimony methods

- The main hypothesis of parsimony sequence-based methods is that character changes are not frequent and thus the phylogenies that best explain the data are those requiring the fewest evolutionary changes
- Each character can be analyzed independently from the others

$$PS(T|a_{\star,j}) = \min_{\mathcal{T}} \sum_{uv \in E(T)} c_{\mathcal{T}}(uv)$$

variations of this definition are possible

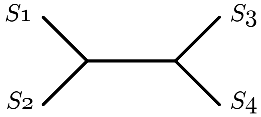


Parsimony methods

- The main hypothesis of parsimony sequence-based methods is that character changes are not frequent and thus the phylogenies that best explain the data are those requiring the fewest evolutionary changes
- Each character can be analyzed independently from the others

$$PS(T|a_{\star,j}) = \min_{\tau} \sum_{uv\in E(T)} c_{\tau}(uv)$$

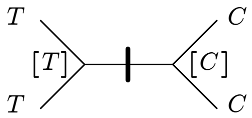
T



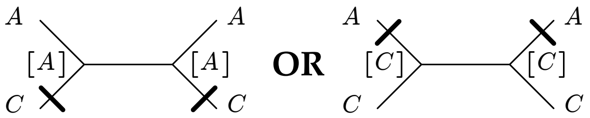
S

S_1	=	T	A	T	T	A
S_2	=	T	C	G	T	A
S_3	=	C	A	G	T	G
S_4	=	C	C	G	T	G

Site 1



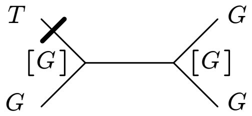
Site 2



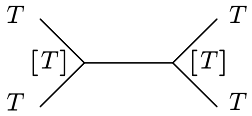
SMALL PROBLEM

O(nm)

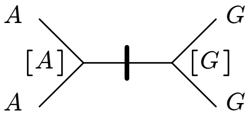
Site 3



Site 4



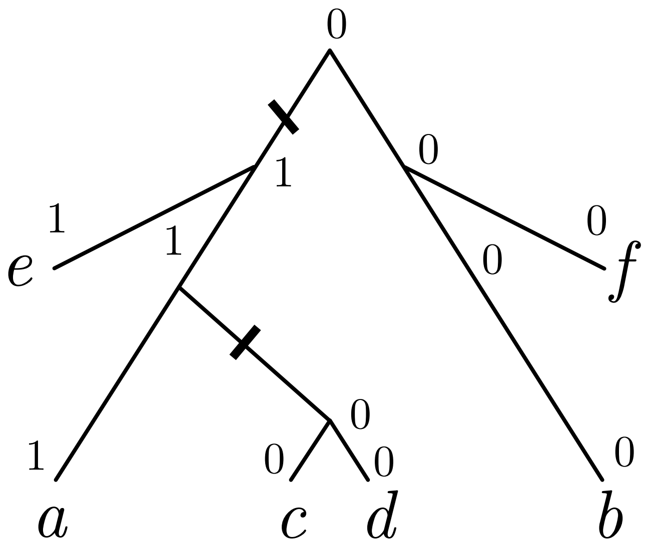
Site 5



Parsimony methods

- The main hypothesis of parsimony sequence-based methods is that character changes are not frequent and thus the phylogenies that best explain the data are those requiring the fewest evolutionary changes
- Each character can be analyzed independently from the others

$$PS(A) = \min_{T} PS(T|A)$$

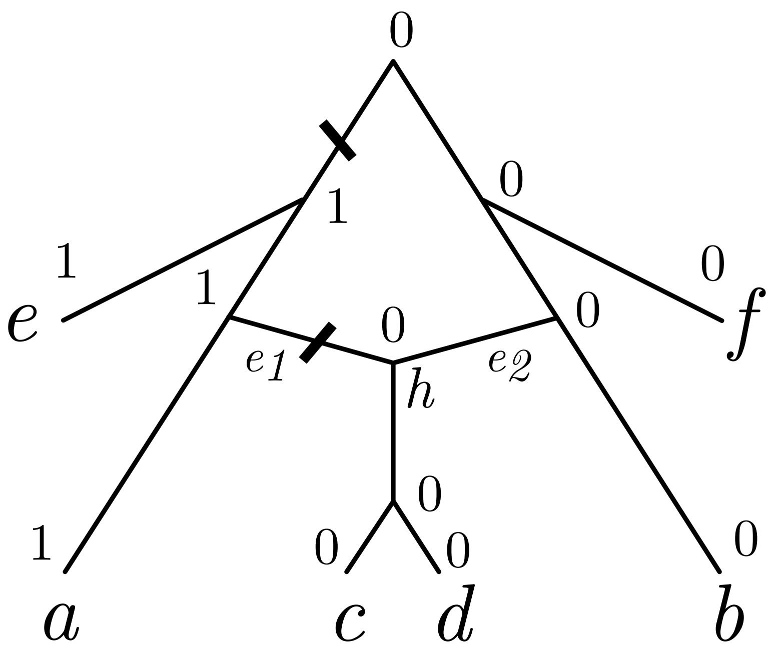


BIG PROBLEM NP-hard

Heuristics
such as hill-climbing

Hardwired parsimony score

SMALL PROBLEM

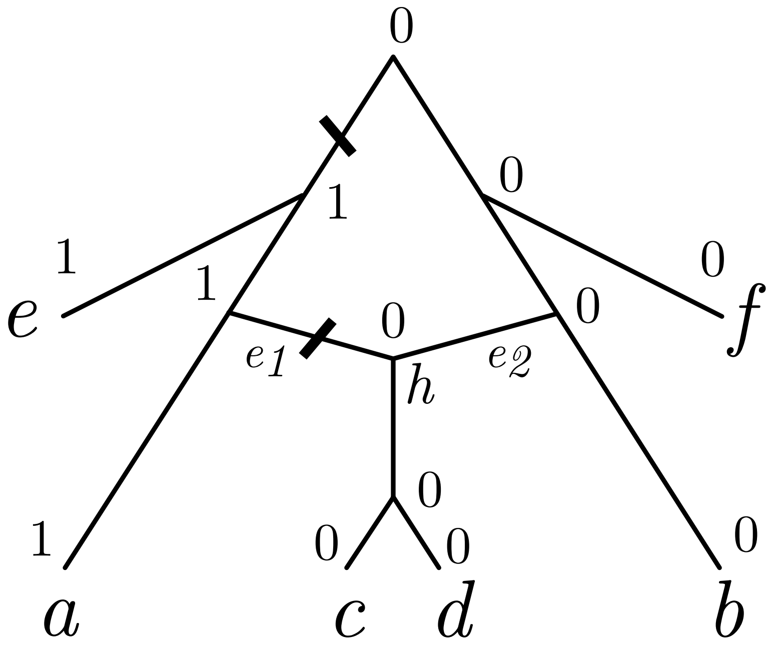


- find the assignment of states to internal nodes of the network such that the total number of edges that connect nodes in different states is minimized (the same definition used for trees!)

$$PS_{hw}(N|a_{\star,j}) = \min_{\tau} \sum_{uv \in E(N)} c_{\tau}(uv)$$

- conjectured to be NP-hard

Hardwired parsimony score - issue

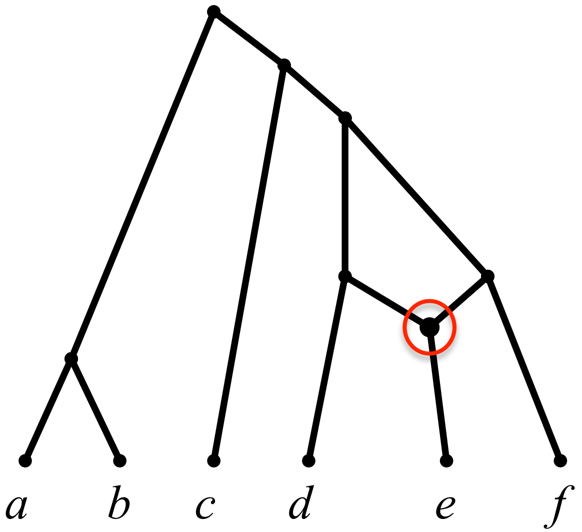


This definition counts a state-change when a reticulation node has the same state as one of its parents, if the other parent has a different state, see for example the reticulation h.

Hence, hardwired parsimony counts more state-changes than necessary since h could very well have inherited its state from its same-state parent.

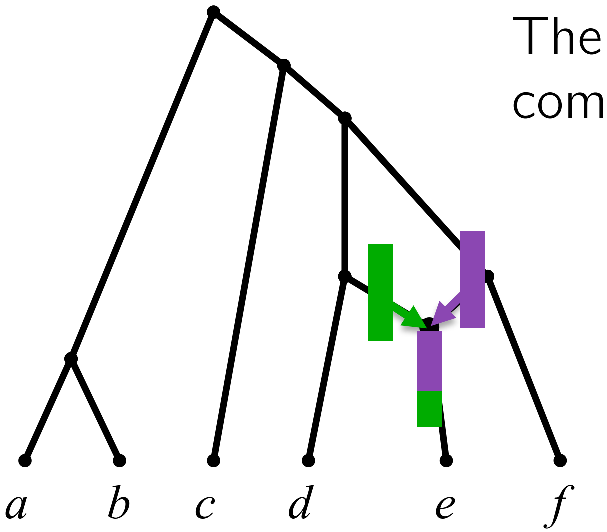
Trees displayed by a network

In a phylogenetic network, a reticulate event is represented as a reticulation, where branches converge to give rise to a new lineage:



Trees displayed by a network

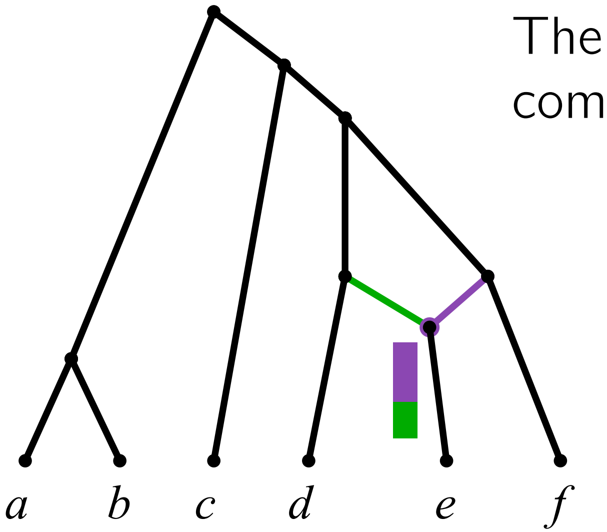
In a phylogenetic network, a reticulate event is represented as a reticulation, where branches converge to give rise to a new lineage:



The genome at the start of the new lineage is a composition of those of the parent lineages.

Trees displayed by a network

In a phylogenetic network, a reticulate event is represented as a reticulation, where branches converge to give rise to a new lineage:

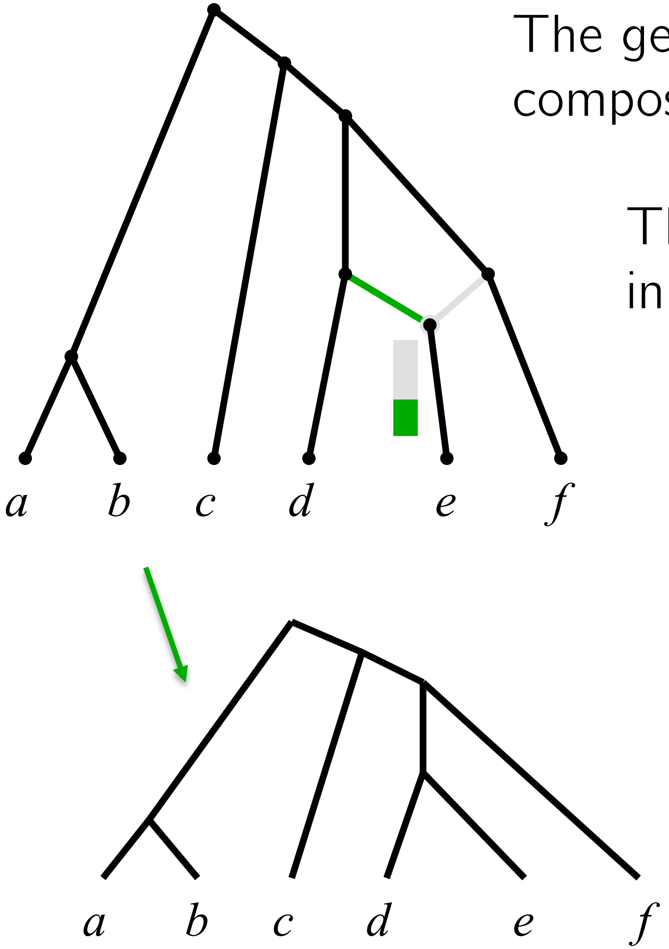


The genome at the start of the new lineage is a composition of those of the parent lineages.

The evolution of each part independently inherited is described by a *"gene" tree*

Trees displayed by a network

In a phylogenetic network, a reticulate event is represented as a reticulation, where branches converge to give rise to a new lineage:

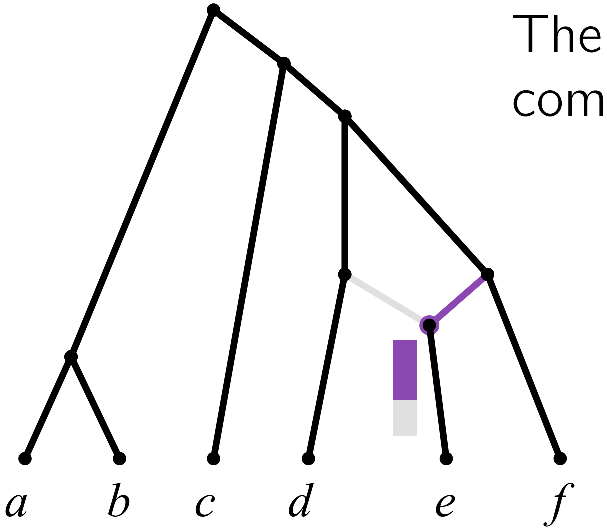


The genome at the start of the new lineage is a composition of those of the parent lineages.

The evolution of each part independently inherited is described by a *"gene" tree*

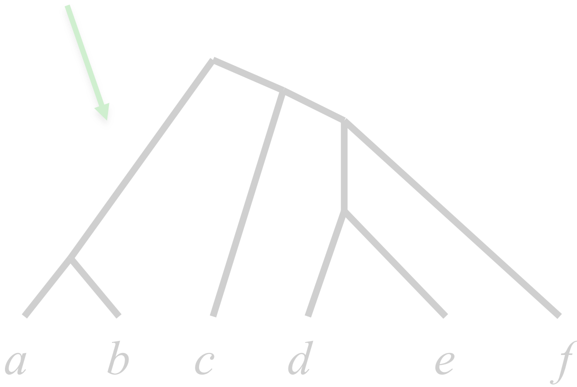
Trees displayed by a network

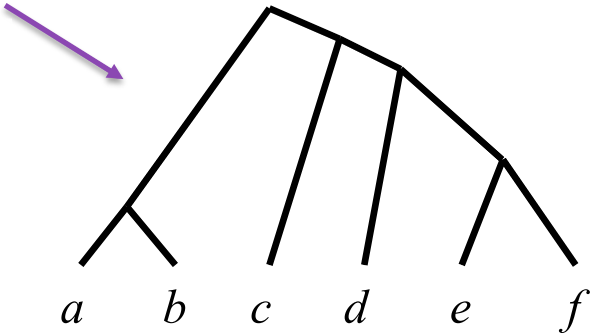
In a phylogenetic network, a reticulate event is represented as a reticulation, where branches converge to give rise to a new lineage:



The genome at the start of the new lineage is a composition of those of the parent lineages.

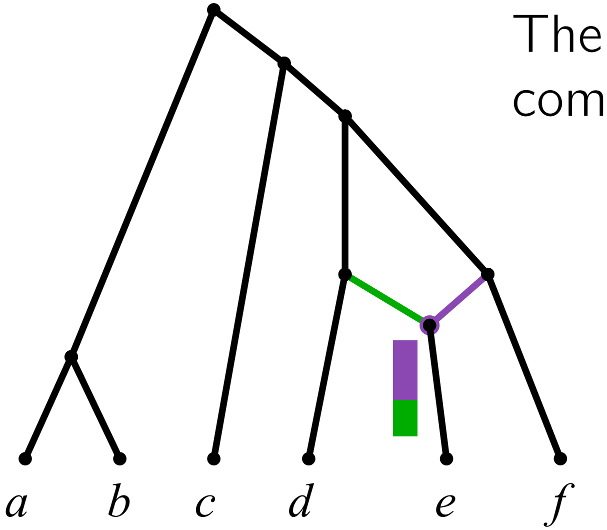
The evolution of each part independently inherited is described by a *“gene” tree*





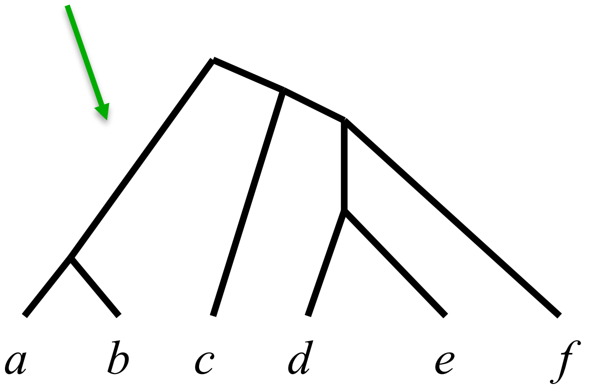
Trees displayed by a network

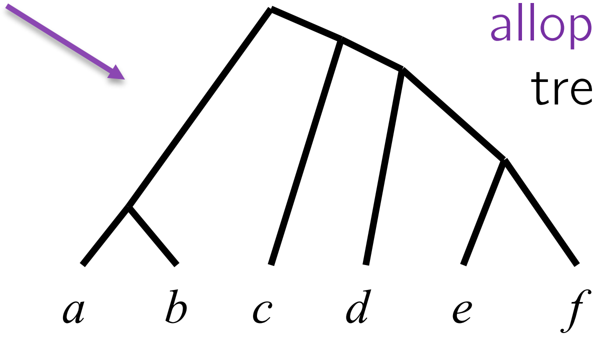
In a phylogenetic network, a reticulate event is represented as a reticulation, where branches converge to give rise to a new lineage:



The genome at the start of the new lineage is a composition of those of the parent lineages.

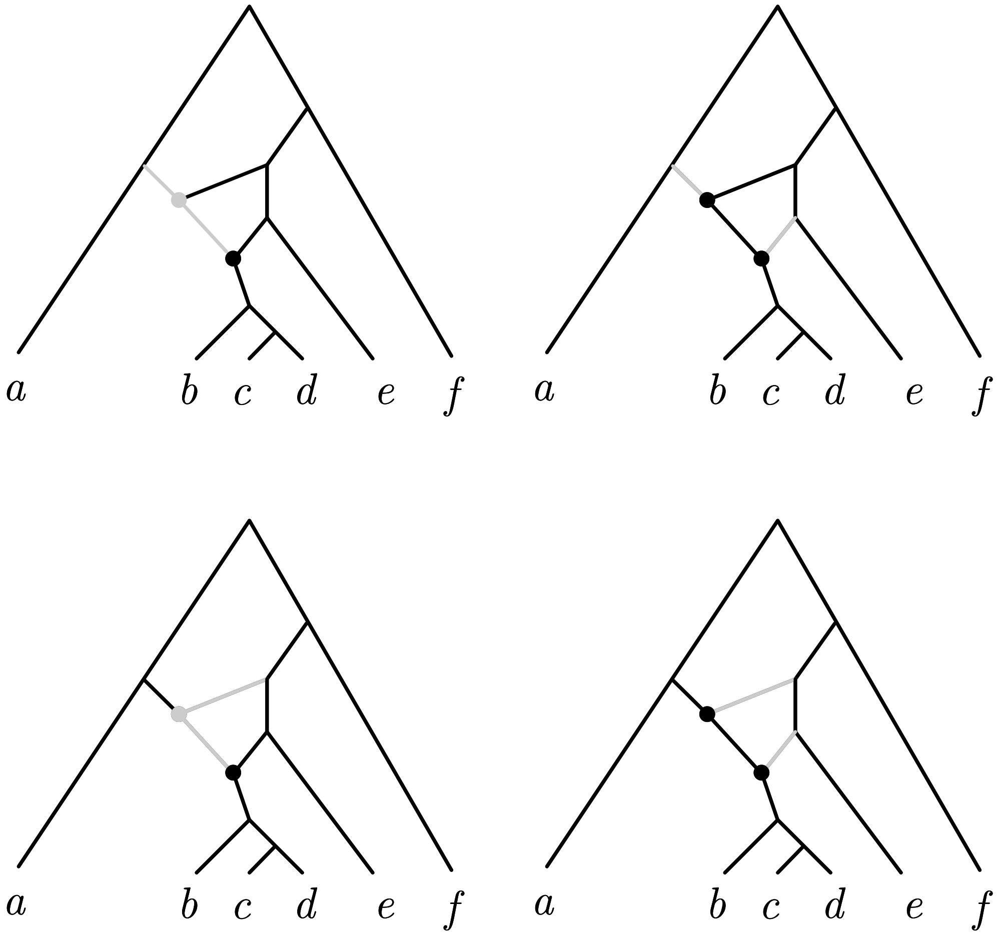
The evolution of each part independently inherited is described by a *"gene" tree*





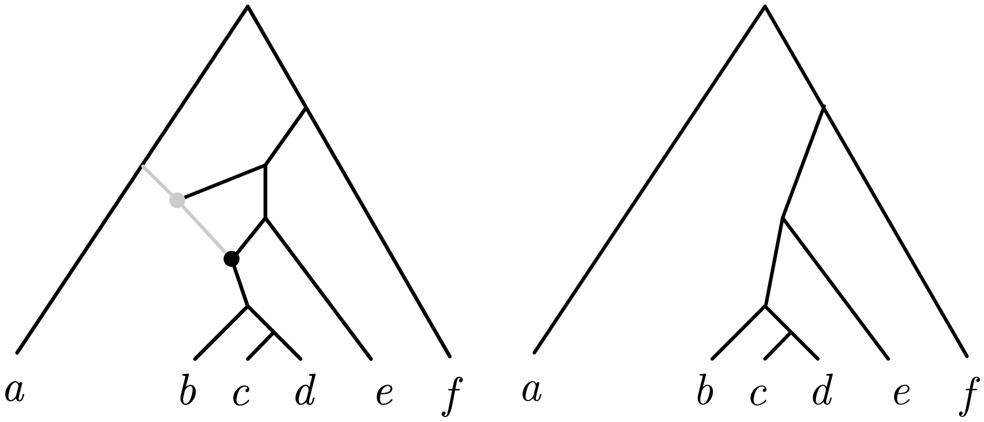
In the absence of deep coalescence and allopolyploidy, the gene trees are *displayed* by the network

Trees displayed by a network



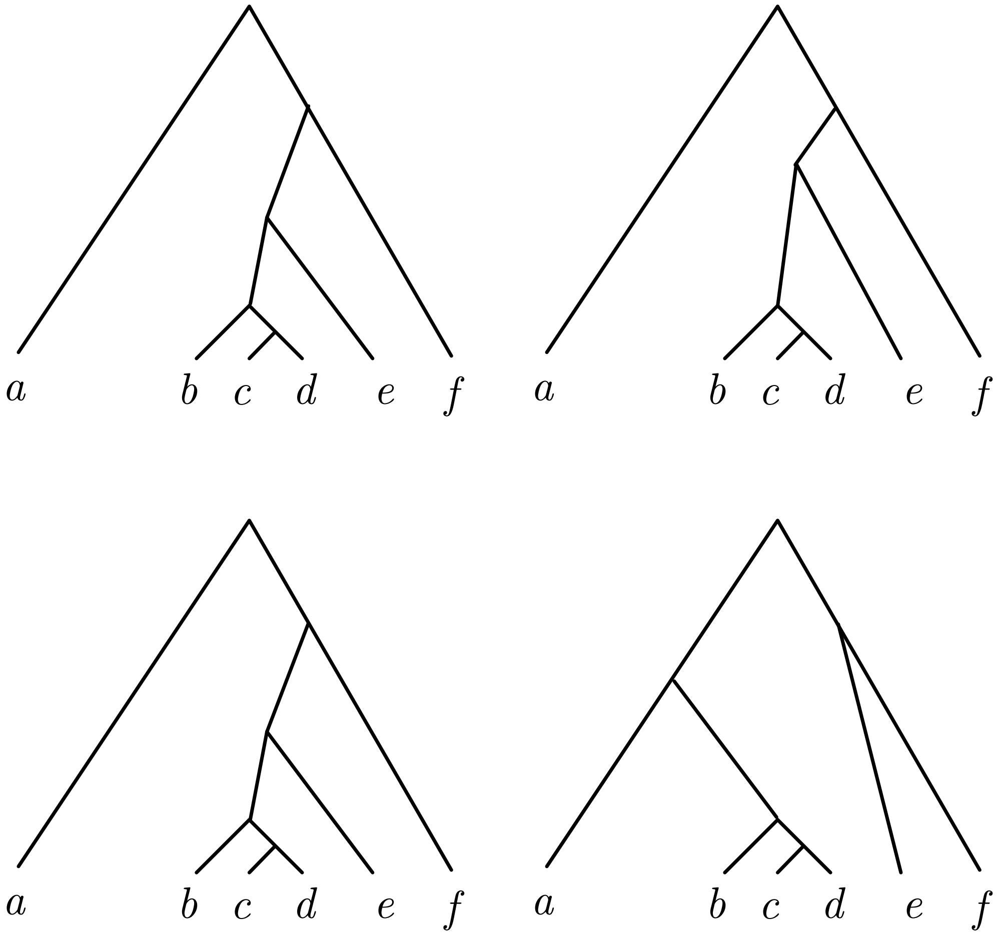
Switch on and off reticulated edges

Trees displayed by a network



Delete switched off edges and unlabelled leaves and suppress outdgree-1 indegree-1 nodes

Trees displayed by a network



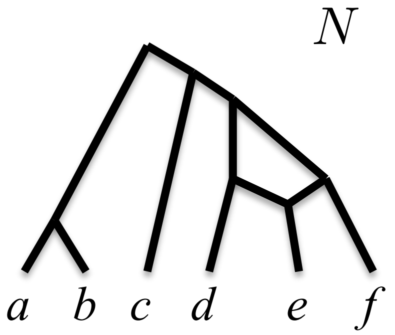
2^r possible trees

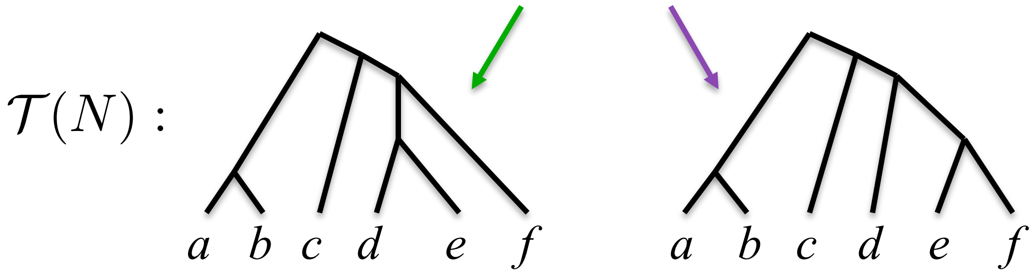
Softwired parsimony score

We evaluate a candidate network *on the basis of how well the trees it displays fit the data:*

$$PS_{sw}(N|a_{\star,j}) = \min_{T\in\mathcal{T}(N)} \min_{\tau} \sum_{uv\in E(T)} c_{\tau}(uv)$$

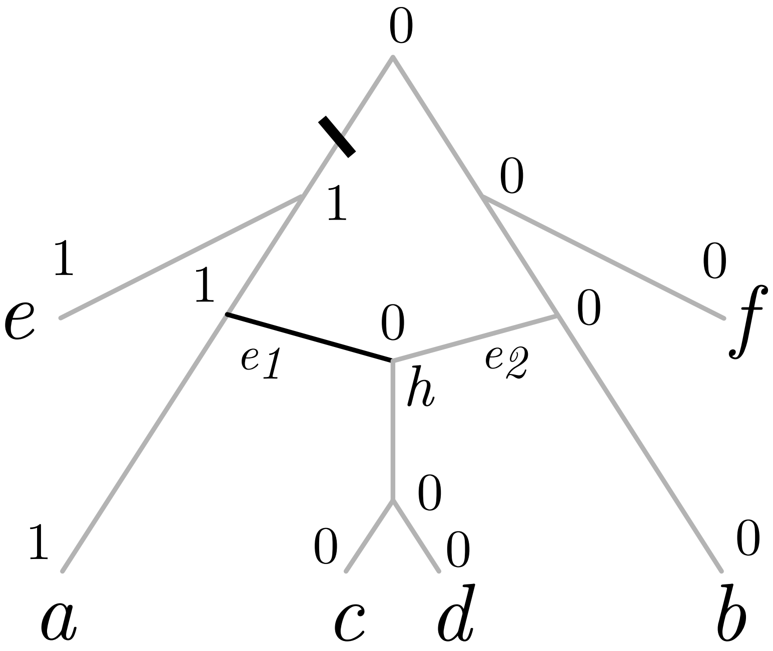
score of a character on a network
= score of the best tree inside the network





Softwired parsimony score - results

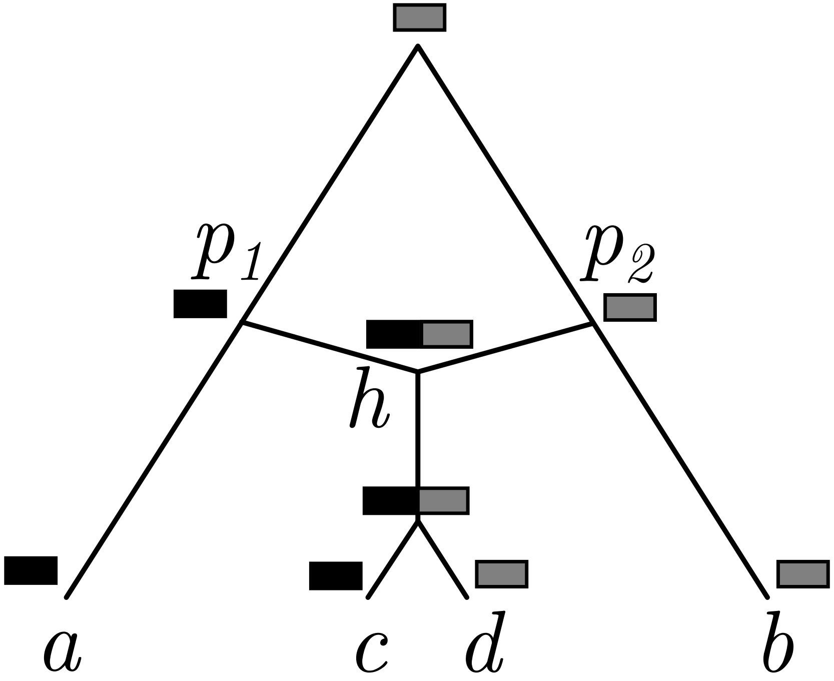
SMALL PROBLEM



- NP-hard for tree-child time-consistent networks and binary characters
- for any constant $\varepsilon > 0$, an approximation factor of $|X|^{1-\varepsilon}$ is not possible in poly time ($|X|^{1/3-\varepsilon}$ for binary networks) unless P = NP
- non-FPT in the parsimony score (NP-hard to know if PS=1!)
- FPT in the level of the network
- fast ILP (simulations)

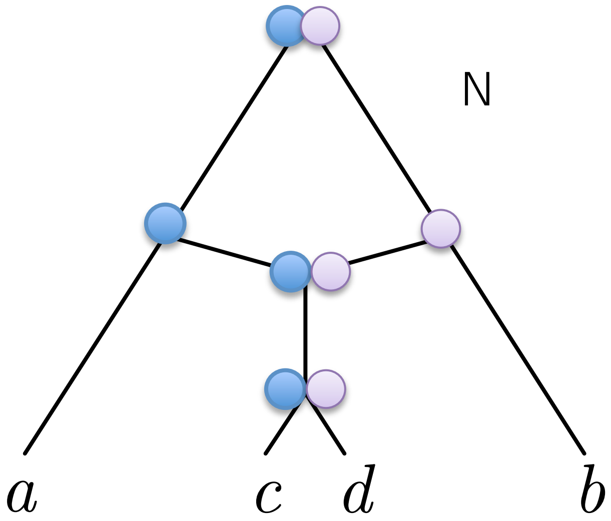
Fischer et al. On computing the maximum parsimony score of a phylogenetic network, 2015

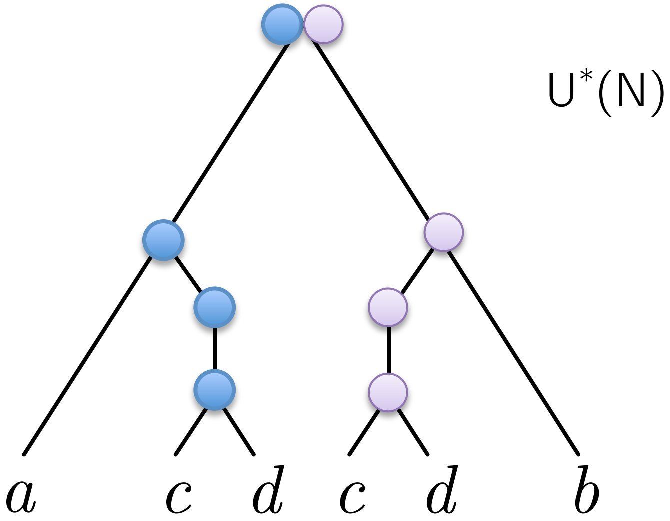
A modeling problem: the allopolyploidy example



The true gene tree is **not displayed** by the network because it needs to *use* both edges entering the hybrid node

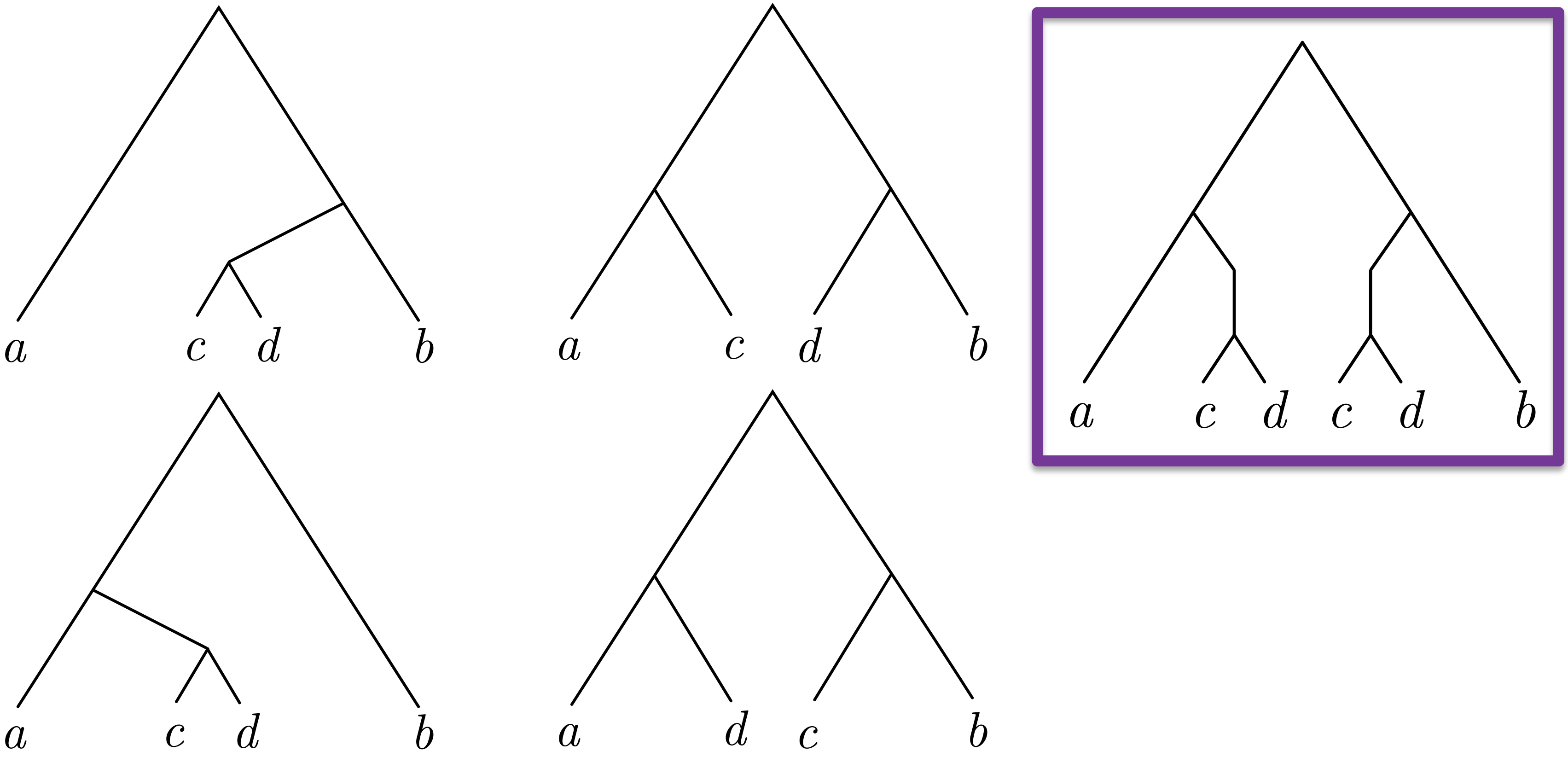
The multi-labelled tree U*(N)





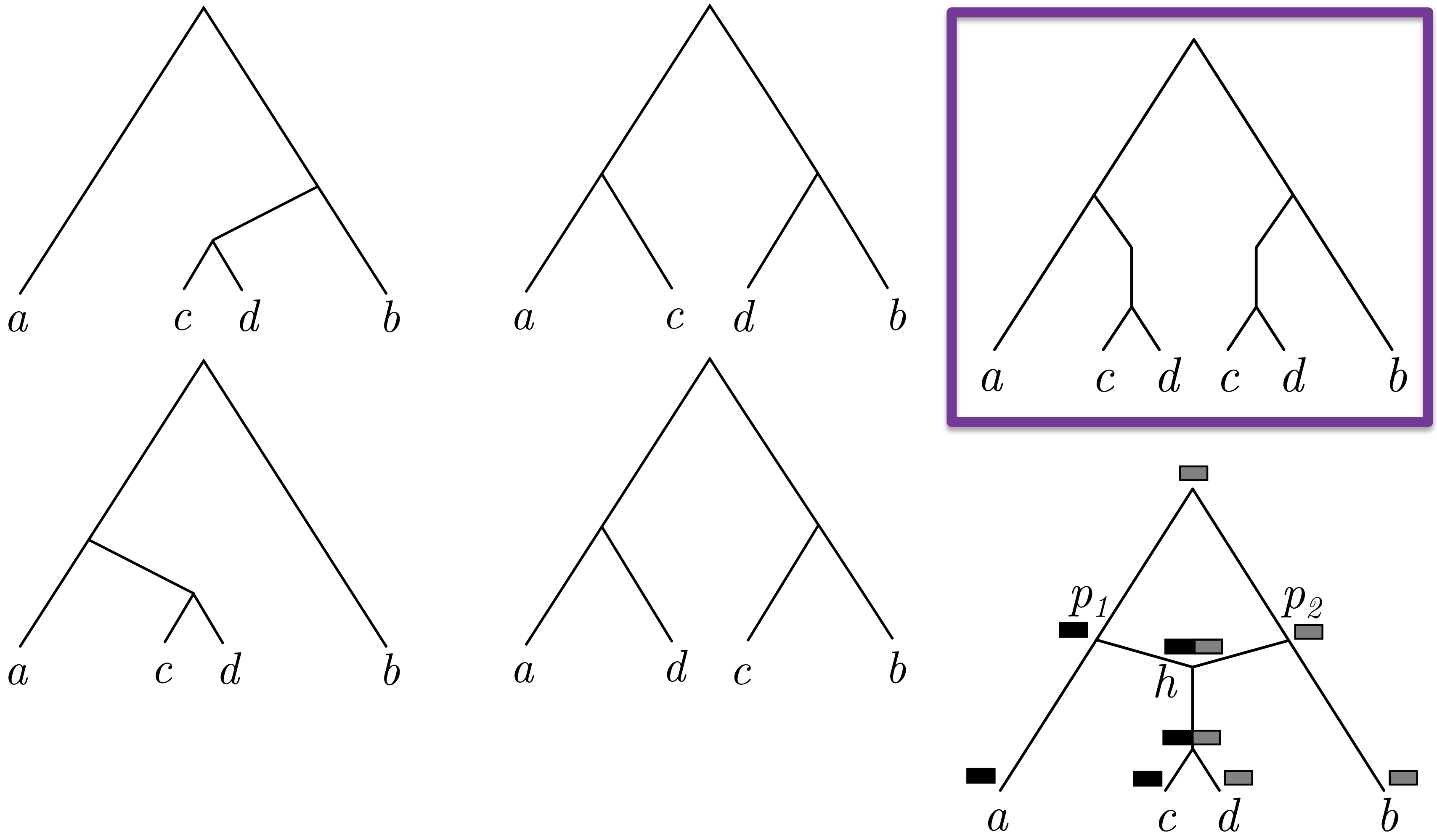
- nodes are the directed paths in N starting at $r(N)$
- for each pair of paths p,p' in N, there is an edge in $U^*(N)$ from p to p' if and only if p=$p'e$ for some edge e in N
- each node in $U^*(N)$ corresponding to a path in N that starts at $r(N)$ and ends at x in X is labelled by x

Parental trees

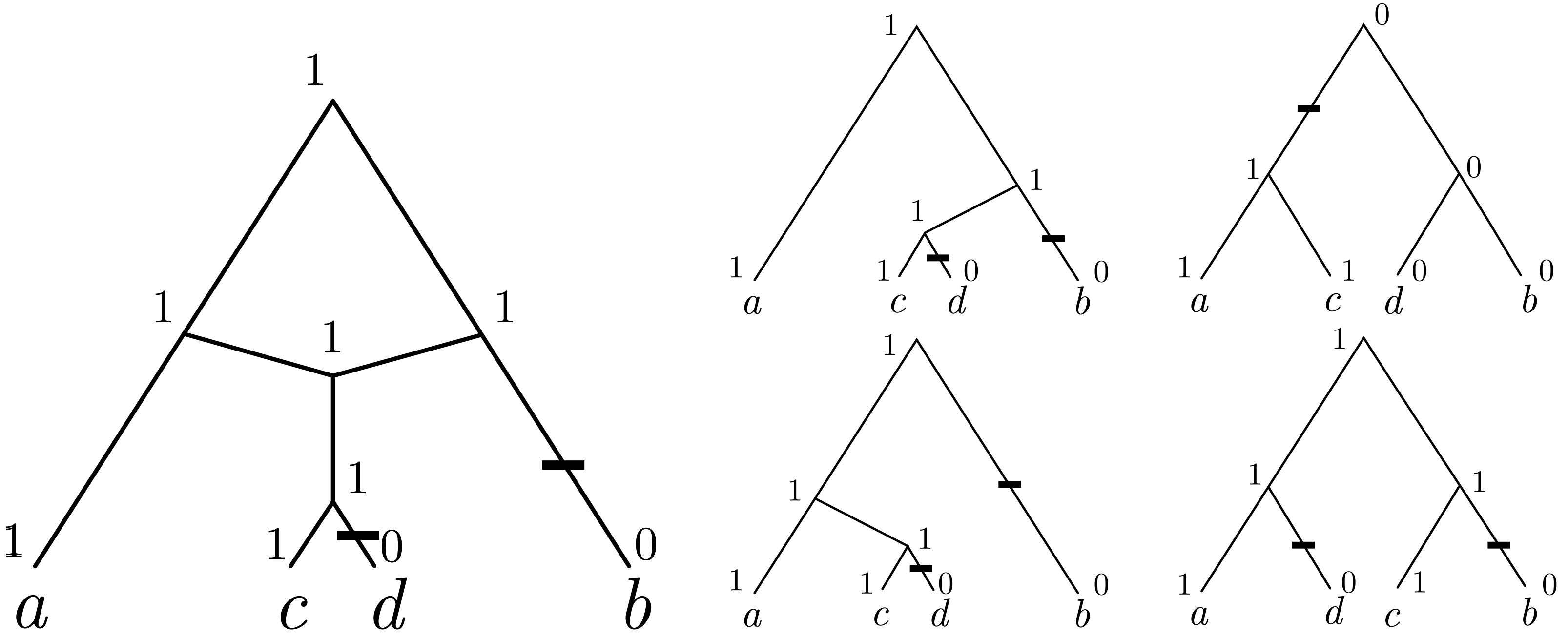


A phylogenetic tree T on X is a parental tree of N if it is displayed by $U^*(N)$

Parental trees

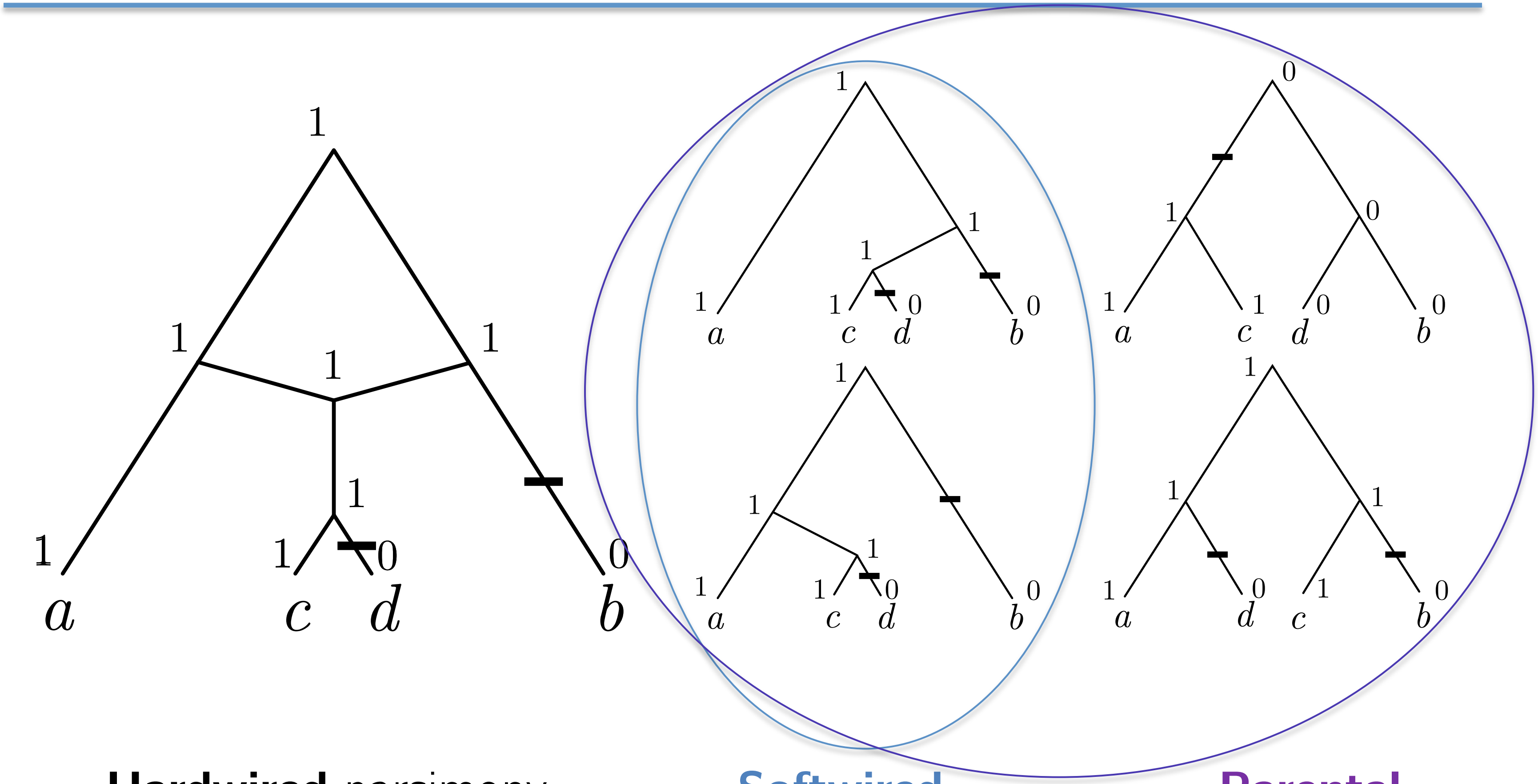


Parental parsimony score



$$PS_{pt}(N|a_{\star,j}) = \min_{T\in\mathcal{PT}(N)} \min_{\tau} \sum_{uv\in E(t)} c_{\tau}(uv)$$

The parsimony scores, an example



Hardwired parsimony

Score =2

$$\min_{\tau} \sum_{uv\in E(N)} c_{\tau}(uv)$$

Softwired

parsimony

Score =2

$$\min_{T\in\mathcal{T}(N)} \min_{\tau} \sum_{uv\in E(T)} c_{\tau}(uv)$$

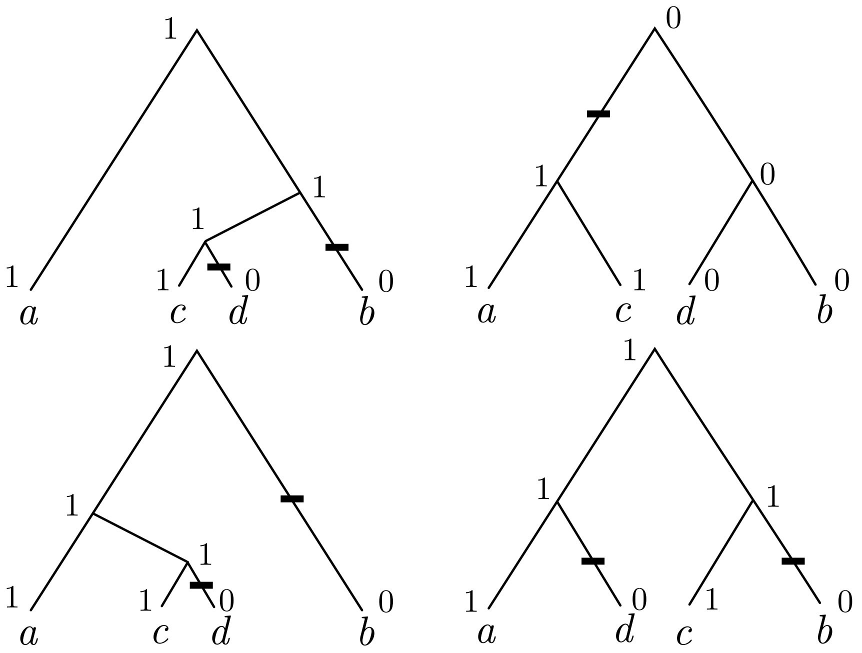
Parental

parsimony

Score =1

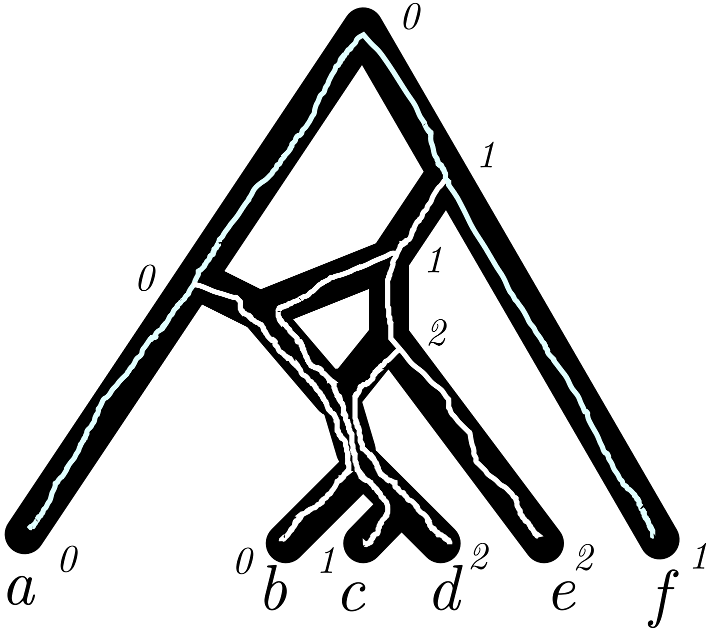
$$\min_{T\in\mathcal{PT}(N)} \min_{\tau} \sum_{uv\in E(T)} c_{\tau}(uv)$$

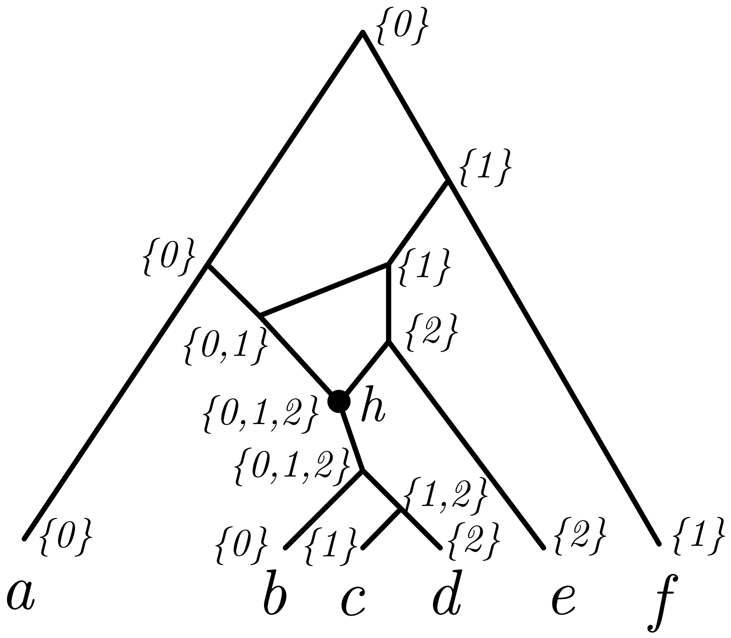
Parental parsimony score - results



- NP-hard even if the network is tree-child and has reticulation depth at most 1 and binary characters
- FPT in the reticulation number of the network
- FPT in the level of the network

Lineage functions





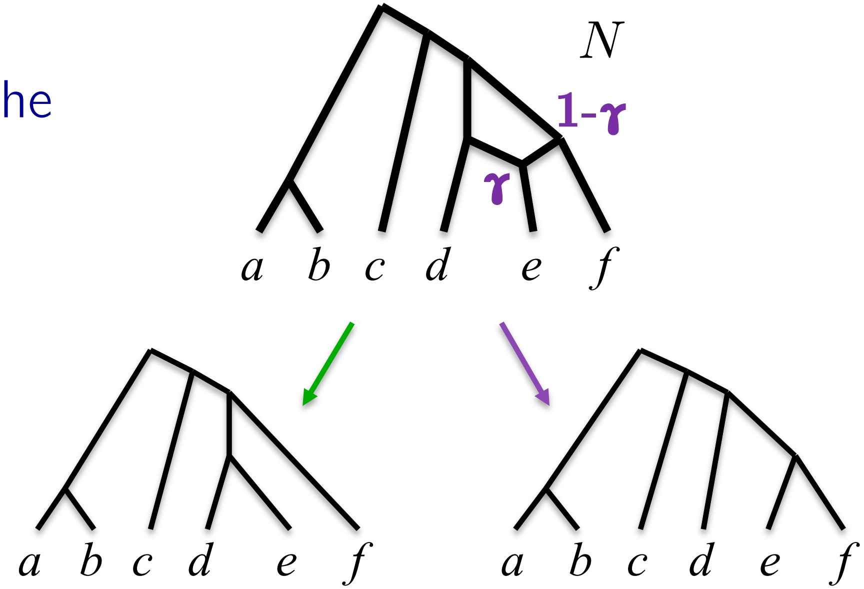
A **lineage function** maps every node in a network to a set of states. Informally, this is a way of tracking how many branches of a parental tree travel through each node of the network, and what states are assigned to each of those branches.

van Iersel et al. Improved maximum parsimony models for phylogenetic networks. Syst Biol. 2018

ML phylogenetic network inference

An optimization problem where a candidate network is evaluated on the basis of how well the trees it ("parentally") displays fit the data:

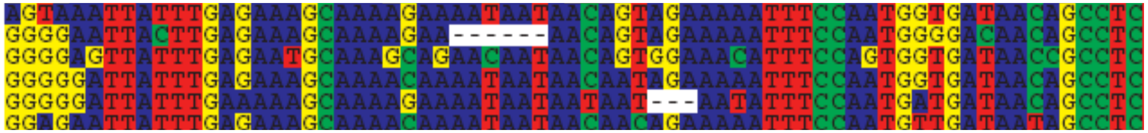
$\mathcal{T}(N)$:



Many possible formulations:

Data:

Sequence alignments:
(typically given in blocks)



```
AGTAAATTATTTGAGAAAGCAAAAGAAAATAATAACAGTAGAAAAATTTCCAATGGTGATAACAGCCTC
GGGGAATTACTTGAGAAAGCAAAAGAA------AACAGTAGAAAAATTTCCAATGGGGACAACAGCCTC
GGGGAGTTATTTGAGAATGCAAAGCAGAACAATAACAGTGGAAACATTTCCAGTGGTGATAACCGCCTC
GGGGGATTATTTGAGAAAGCAAAACAAAATAATAACAATAGAAAAATTTCCAATGGTGATAACAGCCTC
GGGGGATTATTTGAAAAAGCAAAAGAAAATAATAATAAT---AATATTTCCAATGATGATAACAGCCTC
GGAGAATTATTTGAGAAAGCAAAACAAAATAATAACAACAGAAAAATTTCCAATGTTGATAATAGCCTC
```

A_1 $\qquad$ A_2 $\qquad$ $\cdots$ $\qquad$ A_m

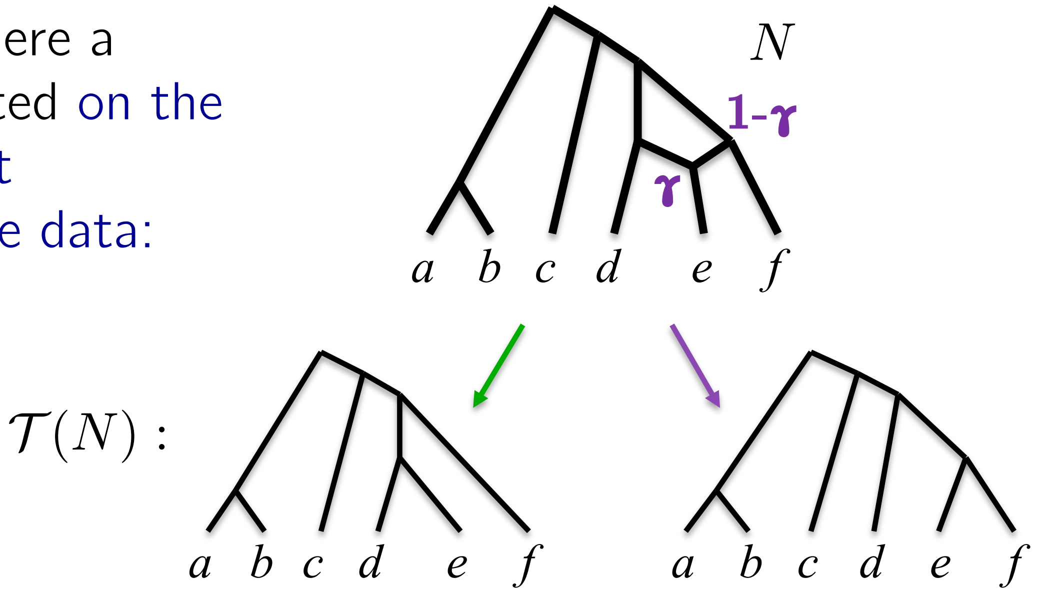
Goal:

Find N that maximises $\mathbf{Pr}(A_1, A_2, \ldots, A_m|N) = \prod_{i=1}^{m} \mathbf{Pr}(A_i|N) = \prod_{i=1}^{m} \left(\sum_{T \in \mathcal{T}(N)} \mathbf{Pr}(A_i|T)\mathbf{Pr}(T|N) \right)$

Jin et al.Maximum likelihood of phylogenetic networks. Bioinformatics 2006.
Yu et al. The Probability of a Gene Tree Topology within a Phylogenetic Network with Applications to Hybridization Detection, 2012

ML phylogenetic network inference

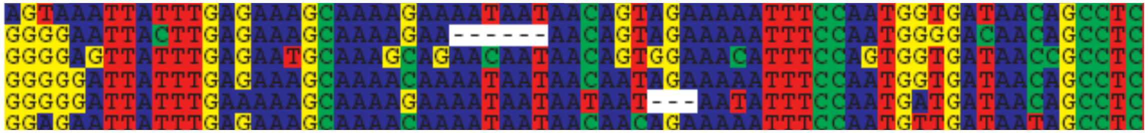
An optimization problem where a candidate network is evaluated on the basis of how well the trees it (“parentally”) displays fit the data:



Many possible formulations:

Data:

Sequence alignments:
(typically given in blocks)


AGTAAATTATTTGAGAAAGCAAAAGAAAATAATAACAGTAGAAAAATTTCCAATGGTGATAACAGCCTC
GGGGAATTACTTGAGAAAGCAAAAGAA------AACAGTAGAAAAATTTCCAATGGGGACAACAGCCTC
GGGGAGTTATTTGAGAATGCAAAGCAGAACAATAACAGTGGAAACATTTCCAGTGGTGATAACCGCCTC
GGGGGATTATTTGAGAAAGCAAAACAAAATAATAACAATAGAAAAATTTCCAATGGTGATAACAGCCTC
GGGGGATTATTTGAAAAAGCAAAAGAAAATAATAATAAT---AATATTTCCAATGATGATAACAGCCTC
GGAGAATTATTTGAGAAAGCAAAACAAAATAATAACAACAGAAAAATTTCCAATGTTGATAATAGCCTC

A_1 A_2 $\cdots$ A_m

Goal:

Find N that maximises

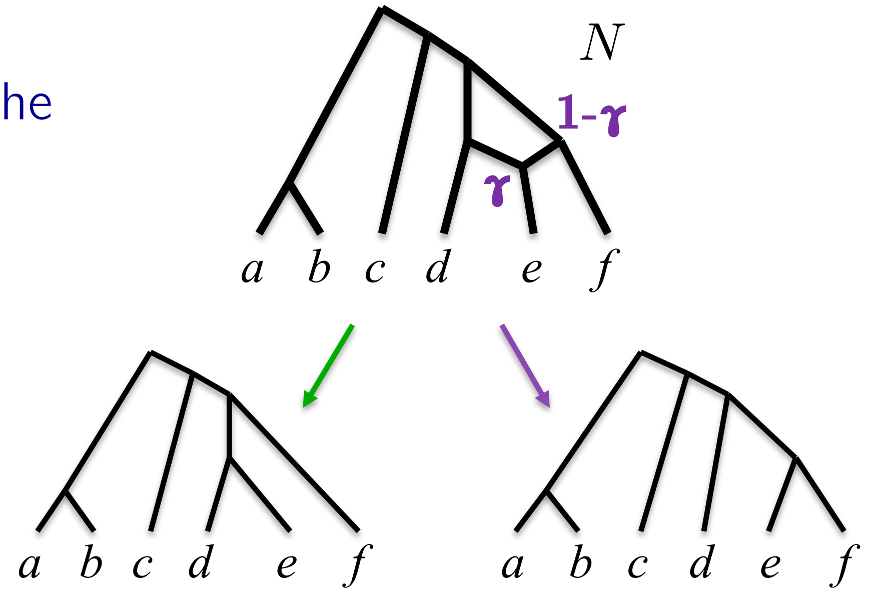
$$\mathbf{Pr}(A_1, A_2, \ldots, A_m|N) = \prod_{i=1}^{m} \mathbf{Pr}(A_i|N) = \prod_{i=1}^{m} \left(\sum_{T\in\mathcal{T}(N)} \mathbf{Pr}(A_i|T)\mathbf{Pr}(T|N) \right)$$

Jin et al.Maximum likelihood of phylogenetic networks. Bioinformatics 2006.
Yu et al. The Probability of a Gene Tree Topology within a Phylogenetic Network with Applications to Hybridization Detection, 2012

ML phylogenetic network inference

An optimization problem where a candidate network is evaluated on the basis of how well the trees it ("parentally") displays fit the data:

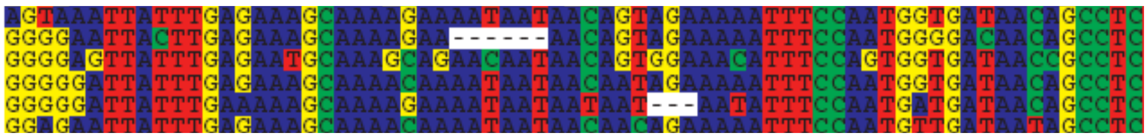
$\mathcal{T}(N)$:



Many possible formulations:

Data:

Sequence alignments:
(typically given in blocks)



```
AGTAAATTATTTGAGAAAGCAAAAGAAAATAATAACAGTAGAAAAATTTCCAATGGTGATAACAGCCTC
GGGGAATTACTTTGAGAAAGCAAAAGAA------AACAGTAGAAAAATTTCCAATGGGGACAACAGCCTC
GGGGAGTTATTTGAGAATGCAAAGCAGAACAATAACAGTGGAAACATTTCCAGTGGTGATAACCGCCTC
GGGGGATTATTTGAGAAAGCAAAACAAAATAATAACAATAGAAAAATTTCCAATGGTGATAACAGCCTC
GGGGGATTATTTGAAAAAGCAAAAGAAAATAATAATAAT---AATATTTCCAATGATGATAACAGCCTC
GGAGAATTATTTGAGAAAGCAAAACAAAATAATAACAACAGAAAAATTTCCAATGTTGATAATAGCCTC
```

A_1 A_2 $\cdots$ A_m

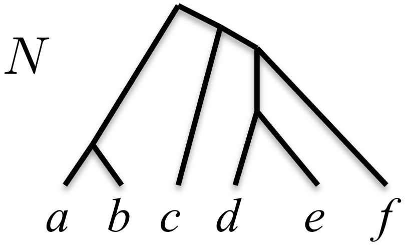
Goal:

Find N that maximises $\mathbf{Pr}(A_1, A_2, \ldots, A_m|N) = \prod_{i=1}^{m} \mathbf{Pr}(A_i|N) = \prod_{i=1}^{m} \left(\sum_{T \in \mathcal{T}(N)} \mathbf{Pr}(A_i|T)\mathbf{Pr}(T|N) \right)$

Jin et al.Maximum likelihood of phylogenetic networks. Bioinformatics 2006.
Yu et al. The Probability of a Gene Tree Topology within a Phylogenetic Network with Applications to Hybridization Detection, 2012

ML phylogenetic tree inference

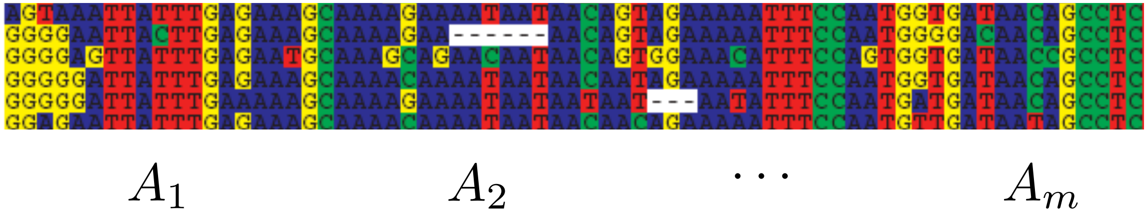
An optimization problem where a candidate network is evaluated on the basis of how well the trees it ("parentally") displays fit the data:



Many possible formulations:

Data:

Sequence alignments:
(typically given in blocks)



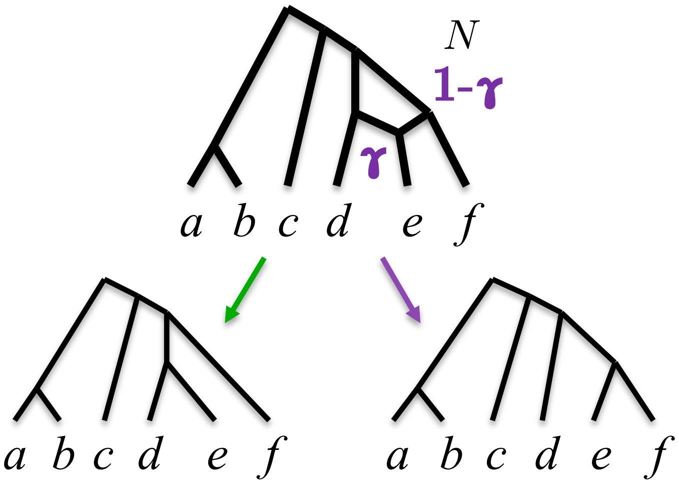
```
AGTAAATTATTTGAGAAAGCAAAAGAAAATAATAACAGTAGAAAAATTTCCAATGGTGATAACAGCCTC
GGGGAATTACTTGAGAAAGCAAAAGAA------AACAGTAGAAAAATTTCCAATGGGGACAACAGCCTC
GGGGAGTTATTTGAGAATGCAAAGCAGAACAATAACAGTGGAAACATTTCCAGTGGTGATAACCGCCTC
GGGGGATTATTTGAGAAAGCAAAACAAAATAATAACAATAGAAAAATTTCCAATGGTGATAACAGCCTC
GGGGGATTATTTGAAAAAGCAAAAGAAAATAATAATAAT---AATATTTCCAATGATGATAACAGCCTC
GGAGAATTATTTGAGAAAGCAAAACAAAATAATAACAACAGAAAAATTTCCAATGTTGATAATAGCCTC
```

Goal:

Find N that maximises $\mathbf{Pr}(A_1, A_2, \ldots, A_m | N) = \prod_{i=1}^{m} \mathbf{Pr}(A_i | N)$:

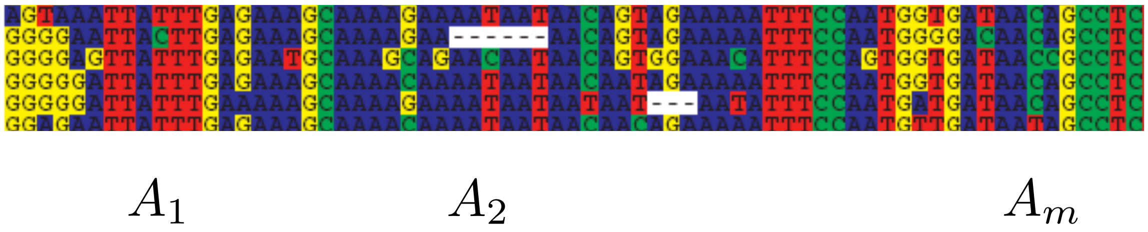
ML under the NMSC

PhyloNet



Data:

Sequence alignments:
(typically given in blocks)



```
AGTAAATTATTTGAGAAAGCAAAAGAAAATAATAACAGTAGAAAAATTTCCAATGGTGATAACAGCCTC
GGGGAATTACTTGAGAAAGCAAAAGAA------AACAGTAGAAAAATTTCCAATGGGGACAACAGCCTC
GGGGAGTTATTTGAGAATGCAAAGCAGAACAATAACAGTGGAAACATTTCCAGTGGTGATAACCGCCTC
GGGGGATTATTTGAGAAAGCAAAACAAAATAATAACAATAGAAAAATTTCCAATGGTGATAACAGCCTC
GGGGGATTATTTGAAAAAGCAAAAGAAAATAATAATAAT---AATATTTCCAATGATGATAACAGCCTC
GGAGAATTATTTGAGAAAGCAAAACAAAATAATAACAACAGAAAAATTTCCAATGTTGATAATAGCCTC
```

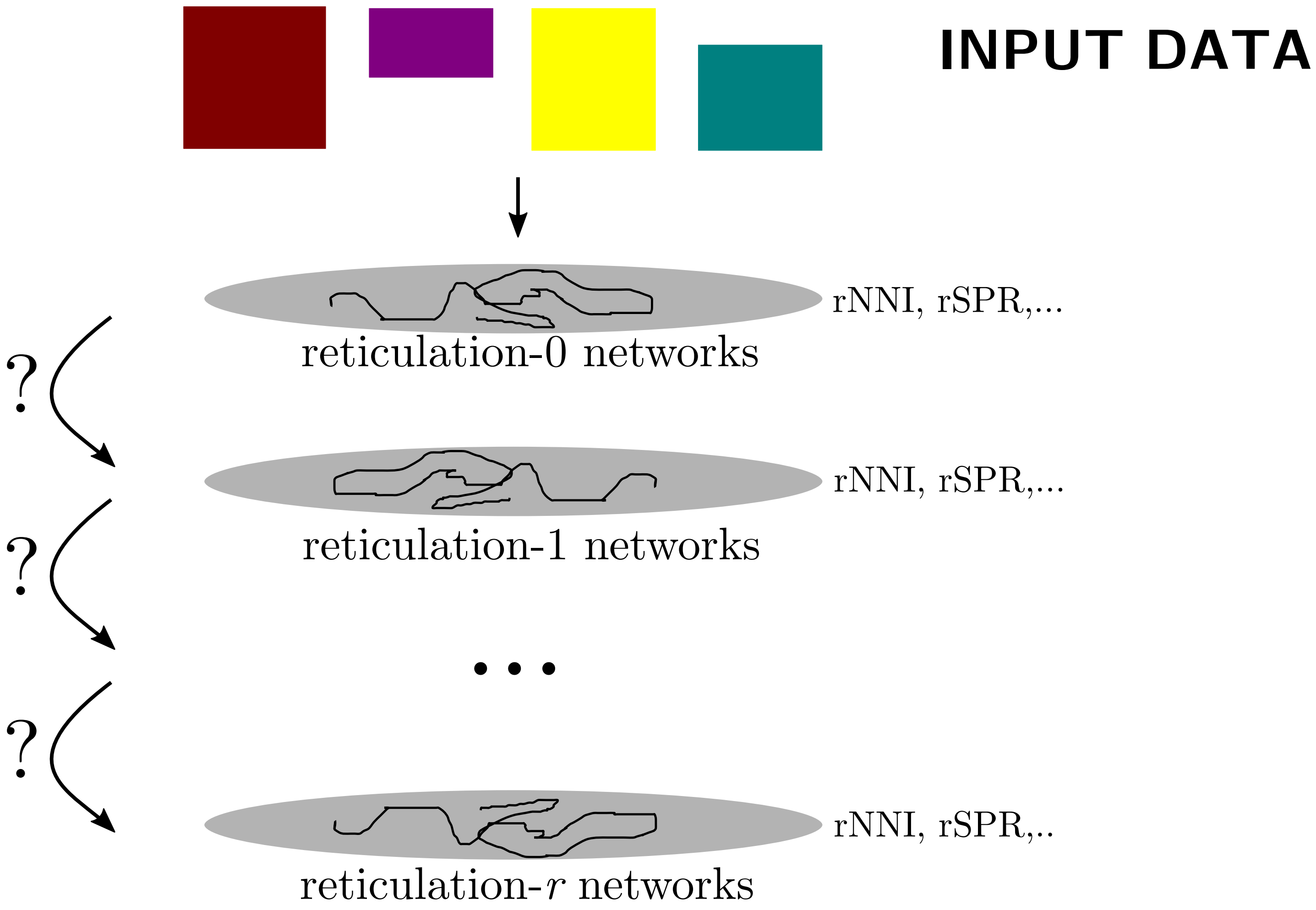
Goal:

Find N that maximises

$$\mathbf{Pr}(A_1, A_2, \ldots, A_m|N) = \prod_{i=1}^{m} p(G_i|N).$$

Zhu and Degnan. Displayed trees do not determine distinguishability under the network multispecies coalescent, 2016
Yu et al. Maximum likelihood inference of reticulate evolutionary histories, 2014
Wen el al. PLOS Genetics 2016 (Bayesian method)

The strategy (hill-climbing, MCMC...)



INPUT DATA

rNNI, rSPR,...

reticulation-0 networks

?

rNNI, rSPR,...

reticulation-1 networks

?

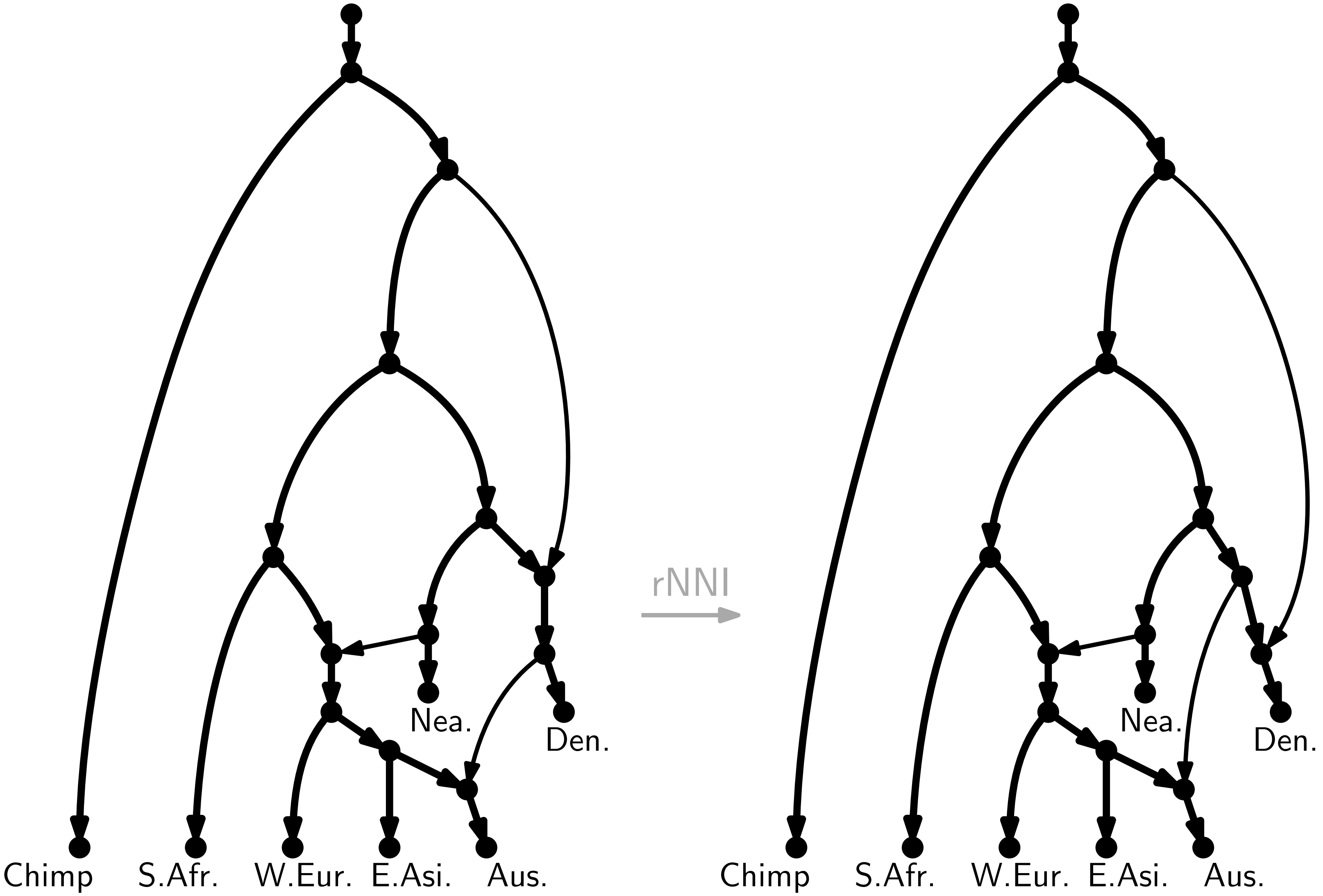
...

?

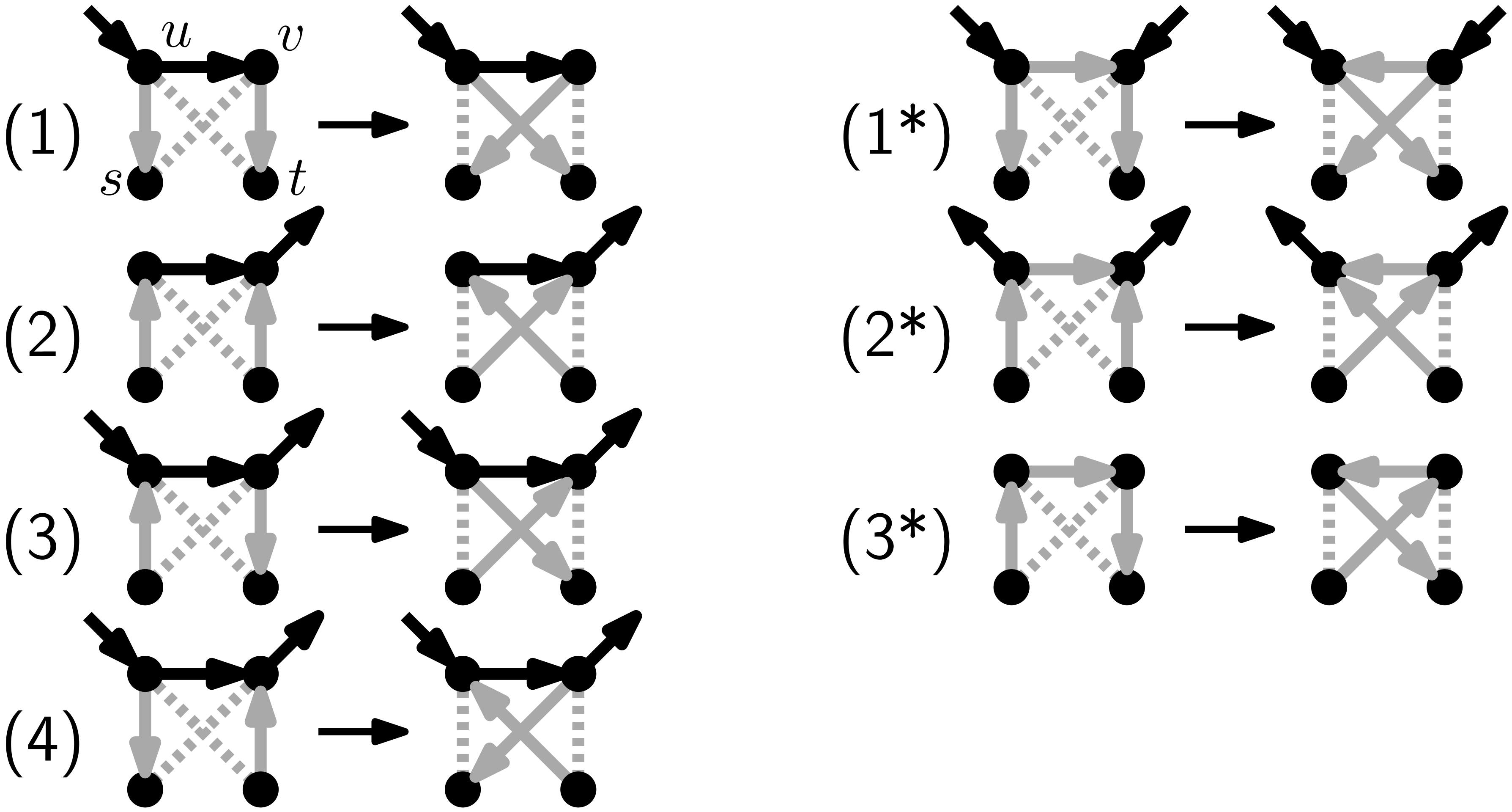
rNNI, rSPR,..

reticulation-r networks

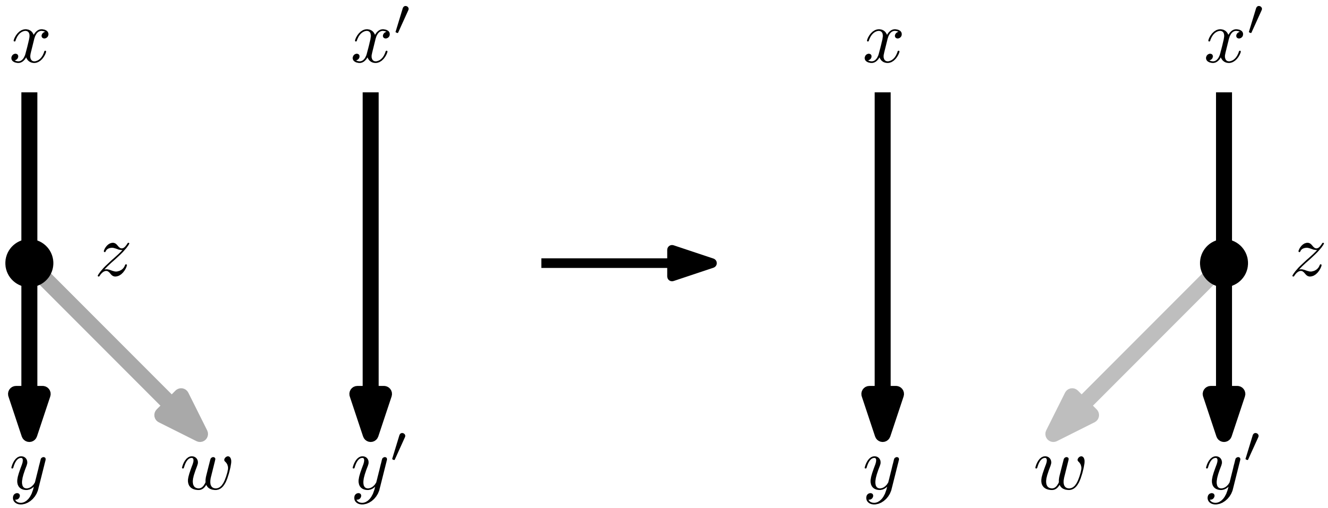
Searching the space of phylogenetic networks

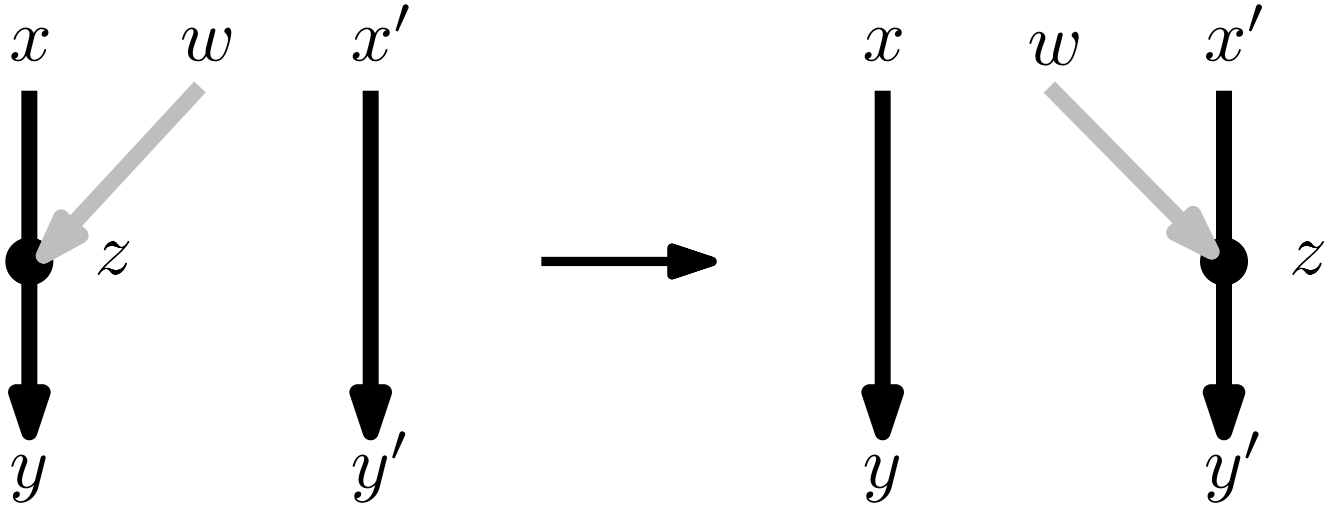


Searching the space of phylogenetic networks (rNNI)



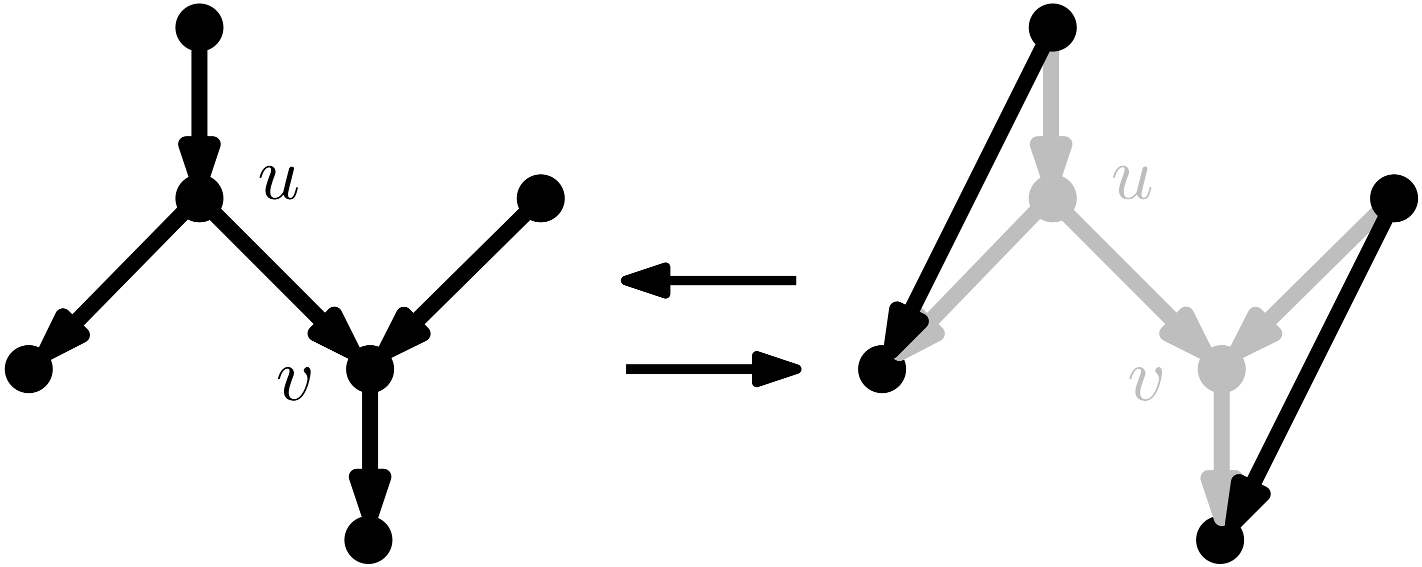
Searching the space of phylogenetic networks (rSPR)



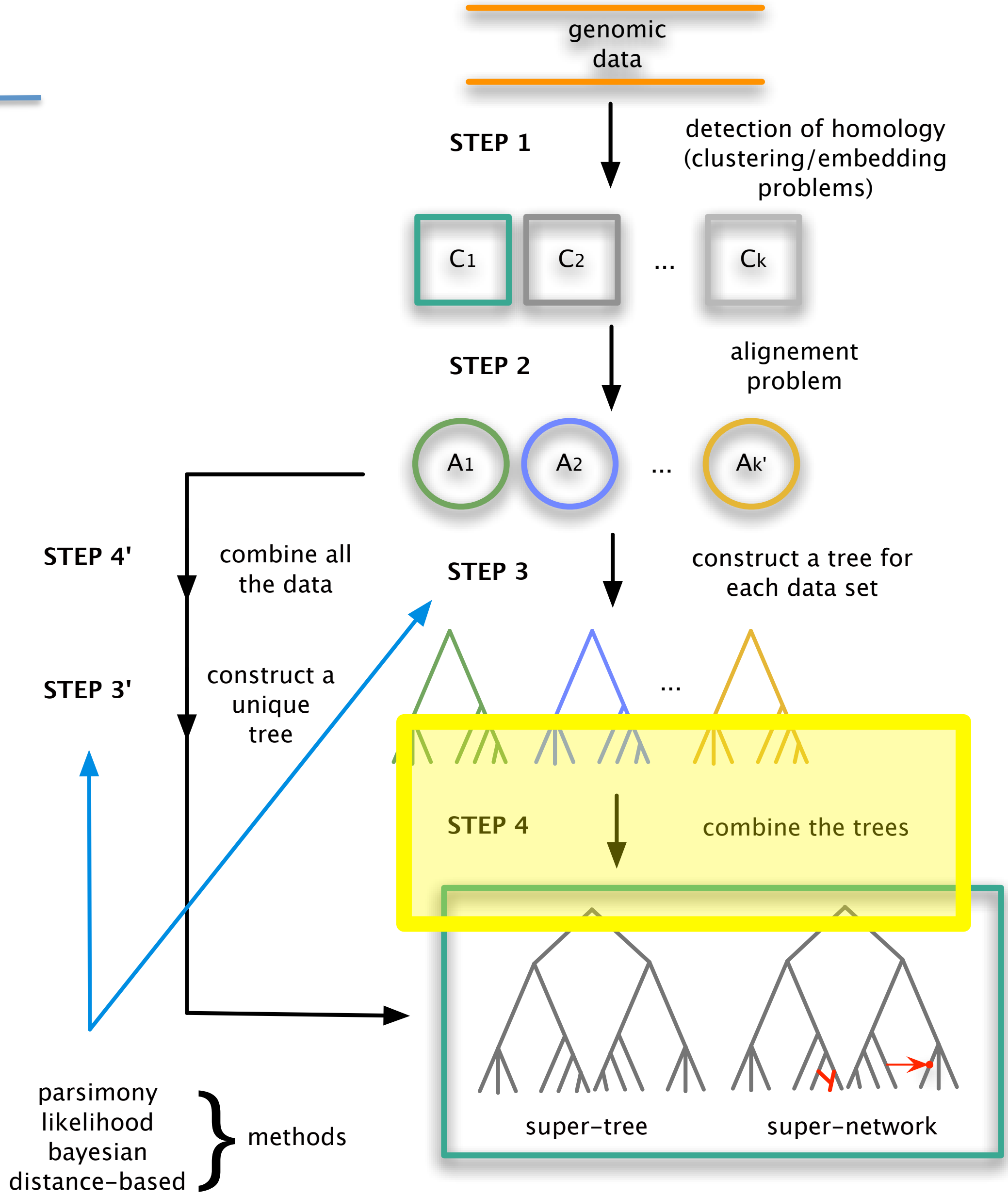


Searching the space of phylogenetic networks

Arc insertion/deletion



Phylogenomics



genomic data

STEP 1 detection of homology (clustering/embedding problems)

STEP 2 alignement problem

STEP 3 construct a tree for each data set

STEP 4 combine the trees

STEP 4' combine all the data

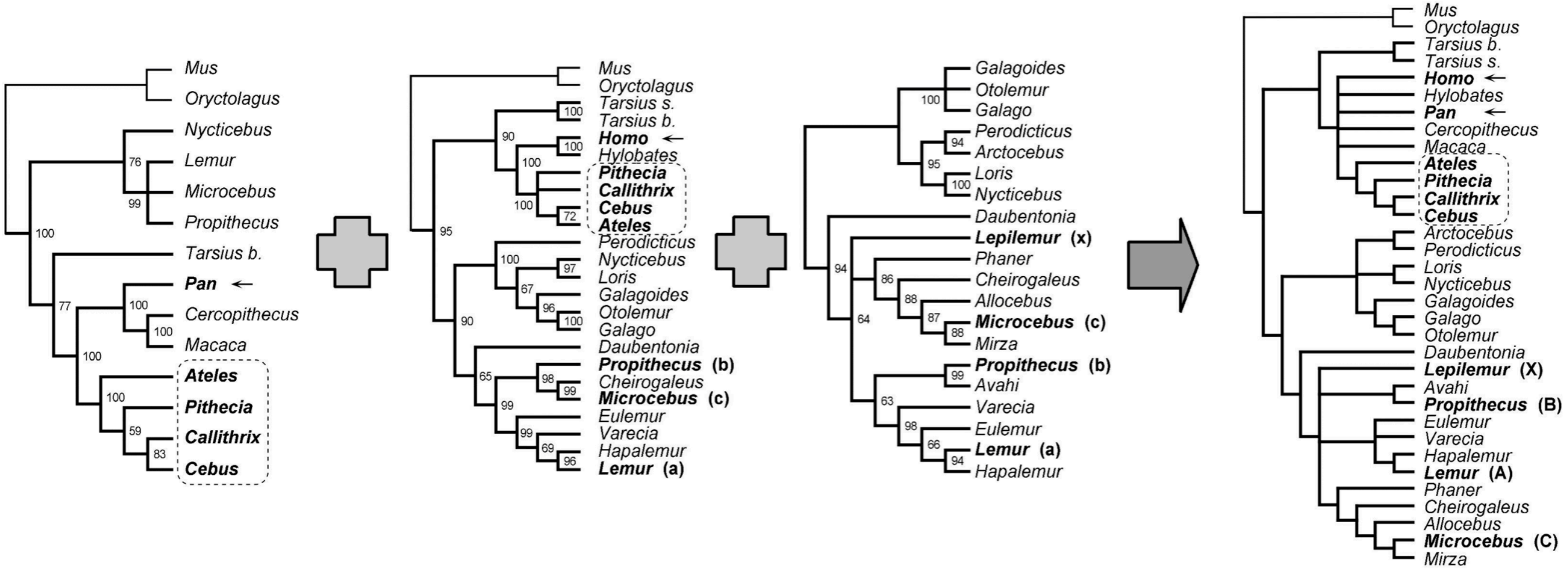
STEP 3' construct a unique tree

super-tree

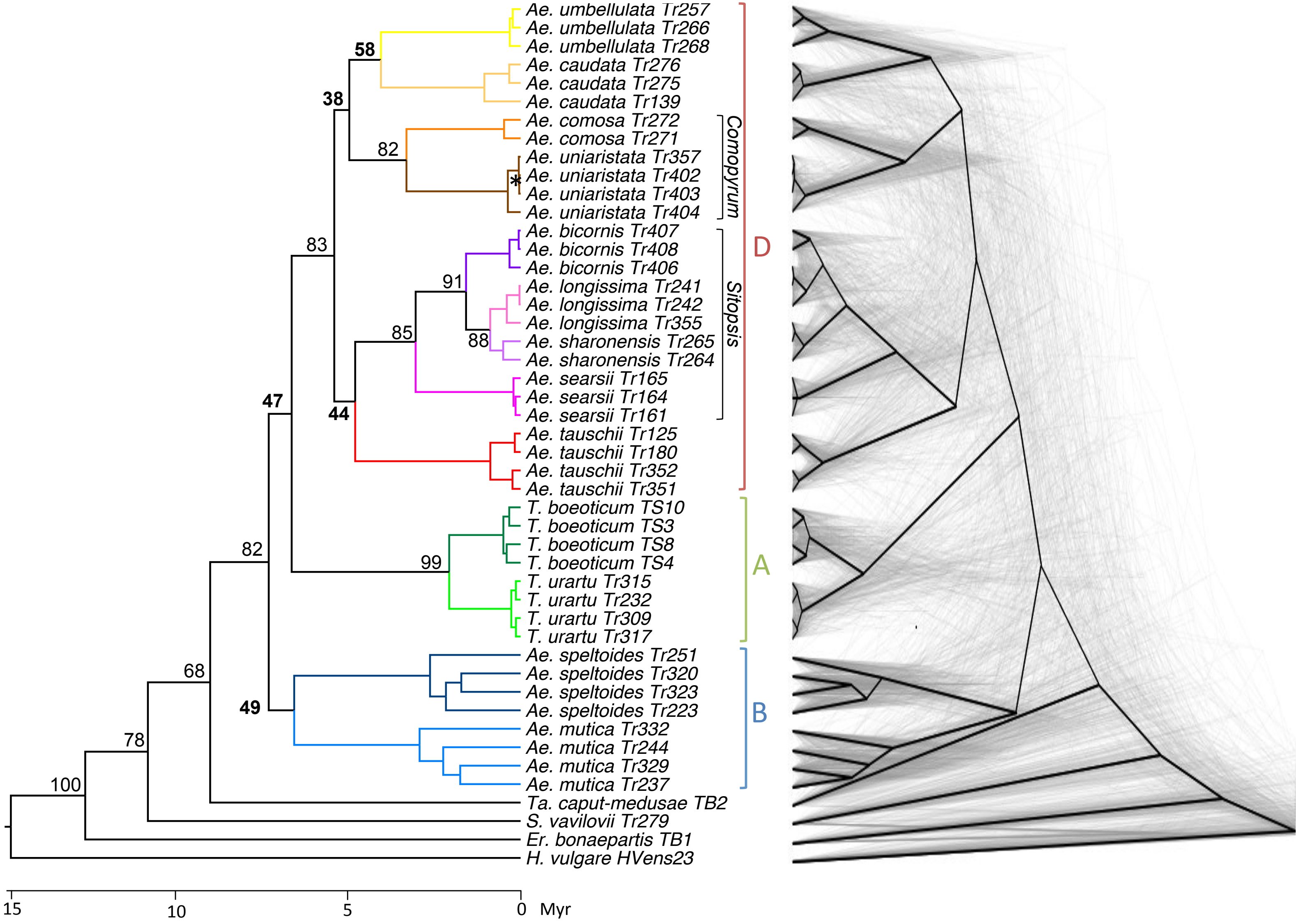
super-network

parsimony
likelihood
bayesian
distance-based } methods

Combining trees



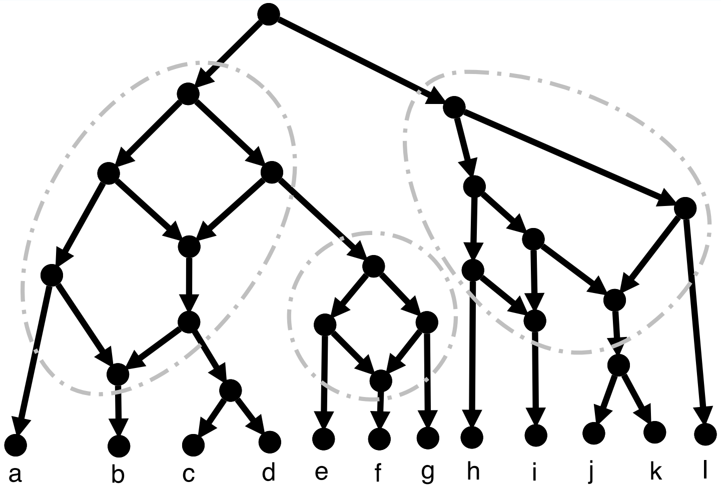
Combining trees



The underlying approach

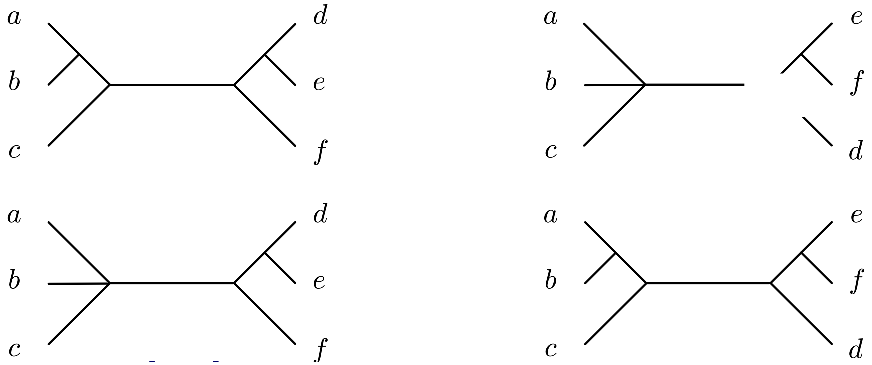
1. • Combinatorial objects such as phylogenetic *trees*, hierarchical *clusters* or *triplets* or *trinets* are constructed from the data of the species under study

2. • These combinatorial objects are combined into a phylogenetic ***network***. The way they are combined and the parameters to optimise (e.g. minimizing the *hybridization number*, i.e. the number of reticulations of the network, or the *level*, i.e. the maximum number of reticulations in each biconnected component) give a large range of different problems

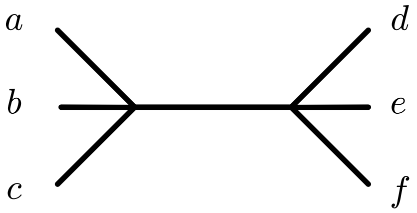


Consensus methods

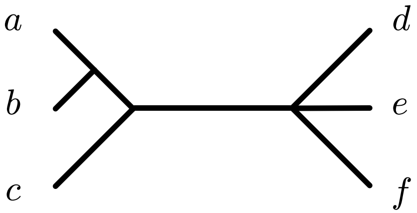
All trees have the **same** taxa



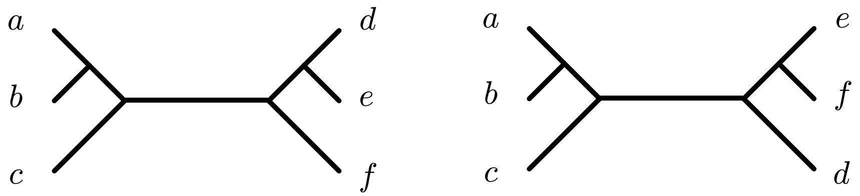
strict consensus, majority consensus



semistrict consensus

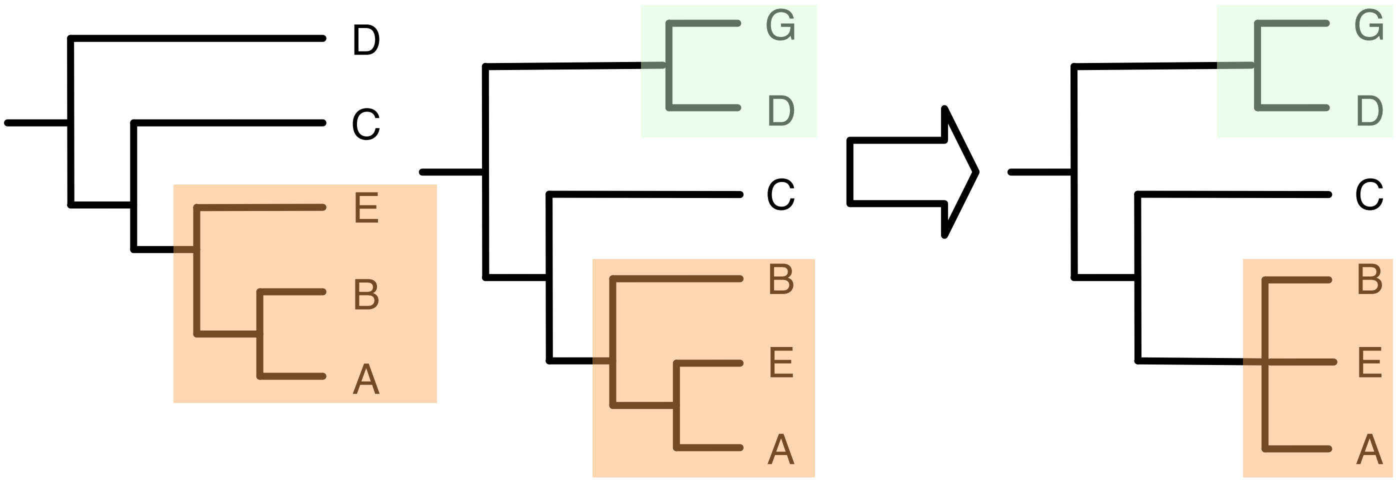


greedy consensus



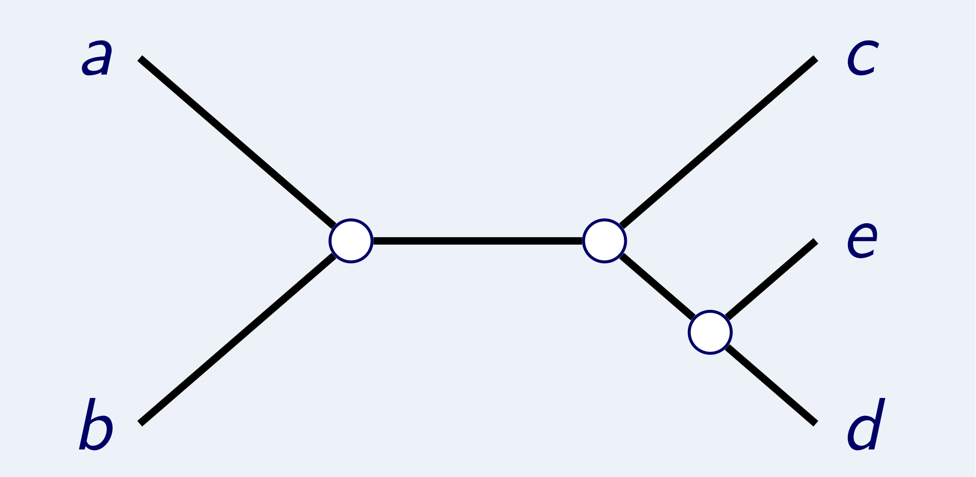
Supertree methods

Trees **do not** have the **same** taxon sets



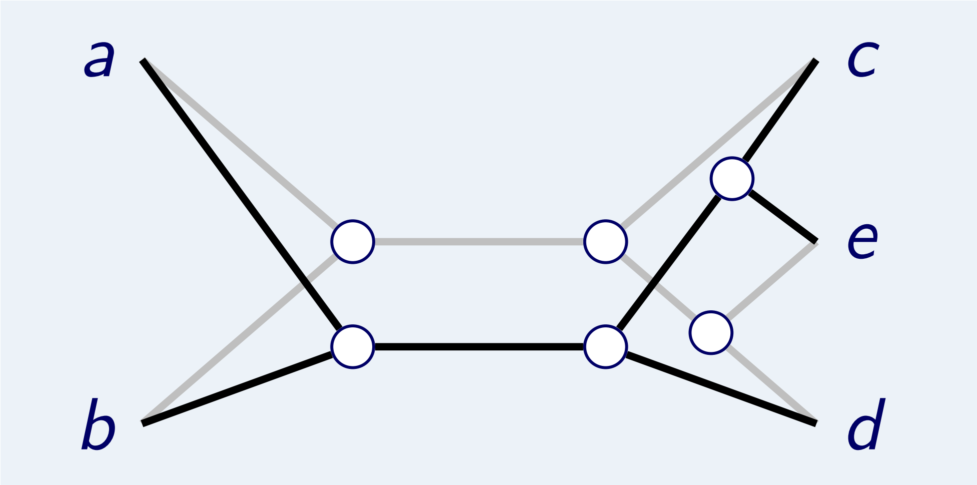
Supertree methods

Display graph



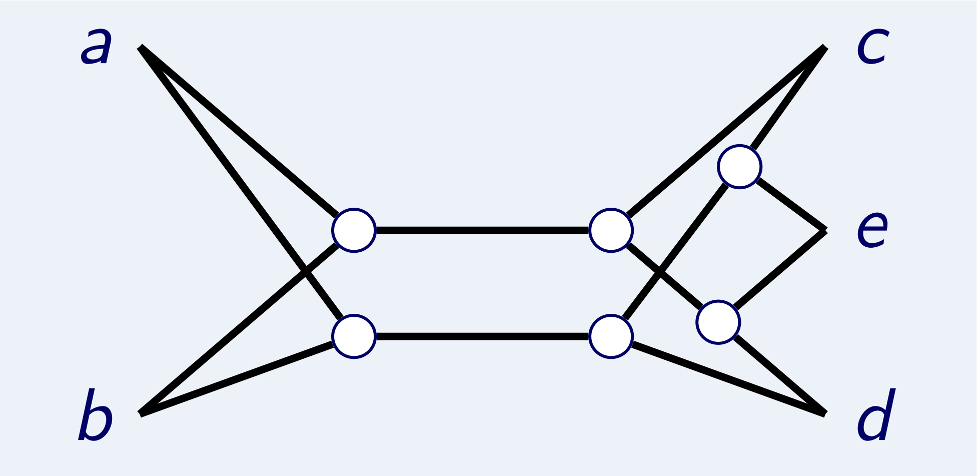
Supertree methods

Display graph



Supertree methods

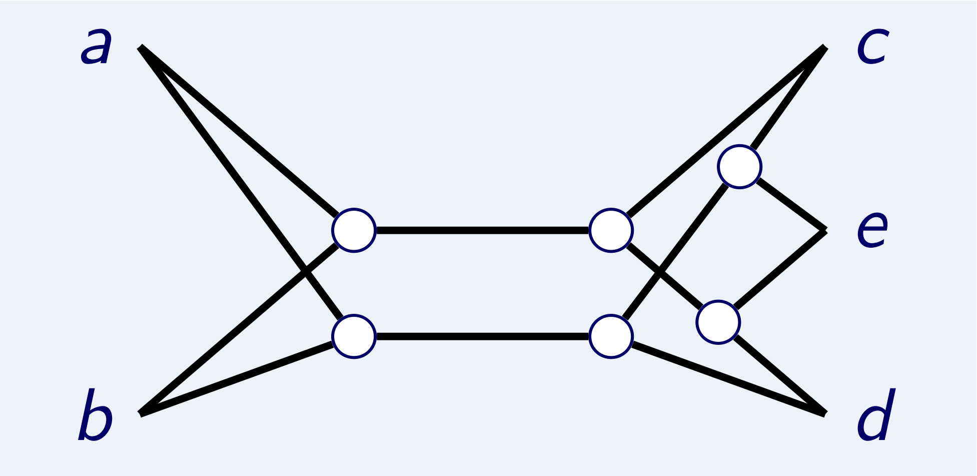
Display graph



The compatibility and the strict compatibility problems for unrooted phylogenetic trees, strongly related, respectively, to the notions of containing as a minor and containing as a topological minor, Both problems are FTP in the number of input trees k, by using their expressibility in MSOL.
But the dependency on k of these algorithms is **prohibitively large.**

Supertree methods

Display graph

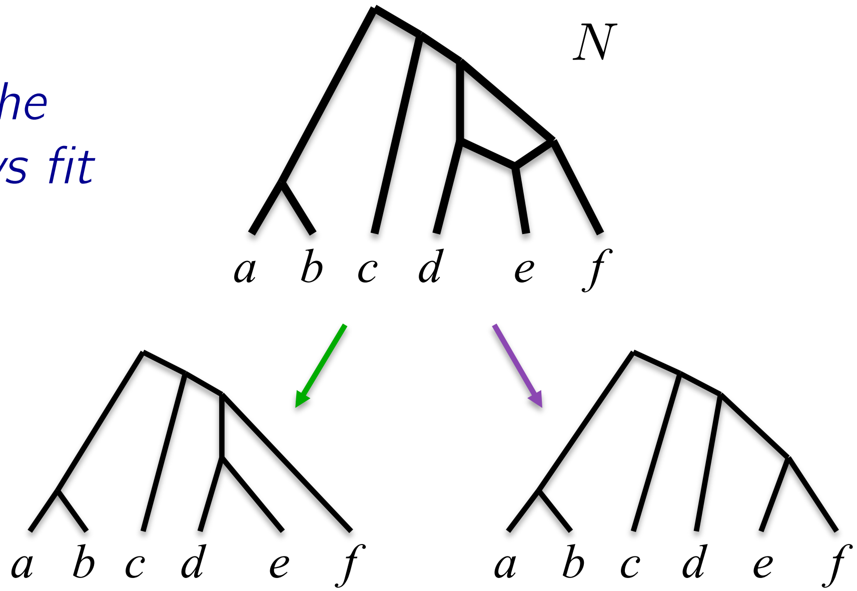


We gave the first explicit dynamic programming algorithms for solving these problems, both runningin time 2 $^{O(k\hat{}2)}$ n, where n is the total size of the input.

Baste el al (2017) Efficient FPT Algorithms for (Strict) Compatibility of Unrooted Phylogenetic Trees. Bulletin of Mathematical Biology.

Phylogenetic supernetwork inference

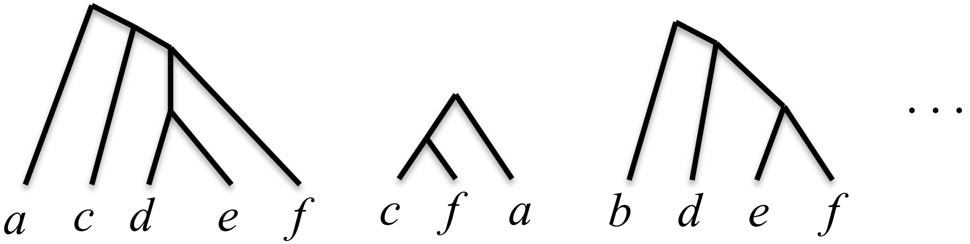
An optimization problem where a candidate network is evaluated *on the basis of how well the trees it displays fit the data:*



Many possible formulations:

Data:

Any trees on the same taxa:



Goal:

Find the network N with the lower hybridization number such that the input trees are `consistent' with one of the trees displayed by N

subject to constraints on the complexity of N

The hybridization number problem

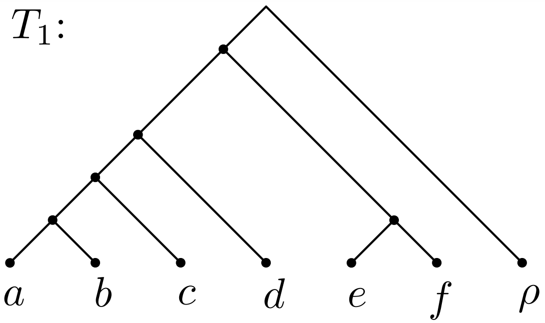
Given: Two rooted binary trees on the same taxon set but different topology.

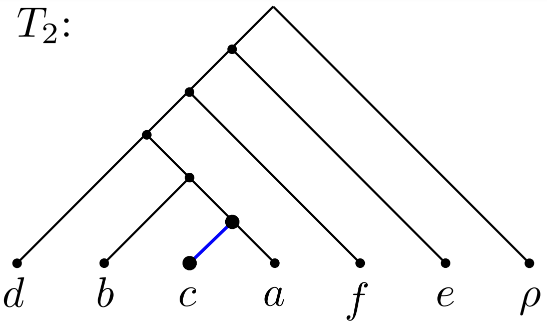
Question: What is the most probable evolutionary history?

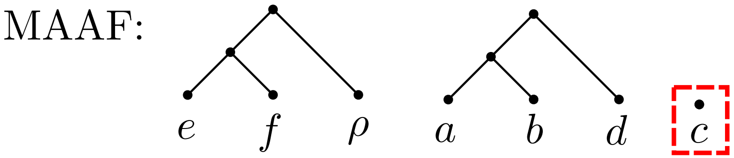
Assumptions: Difference is caused by hybridizations, parsimony framework

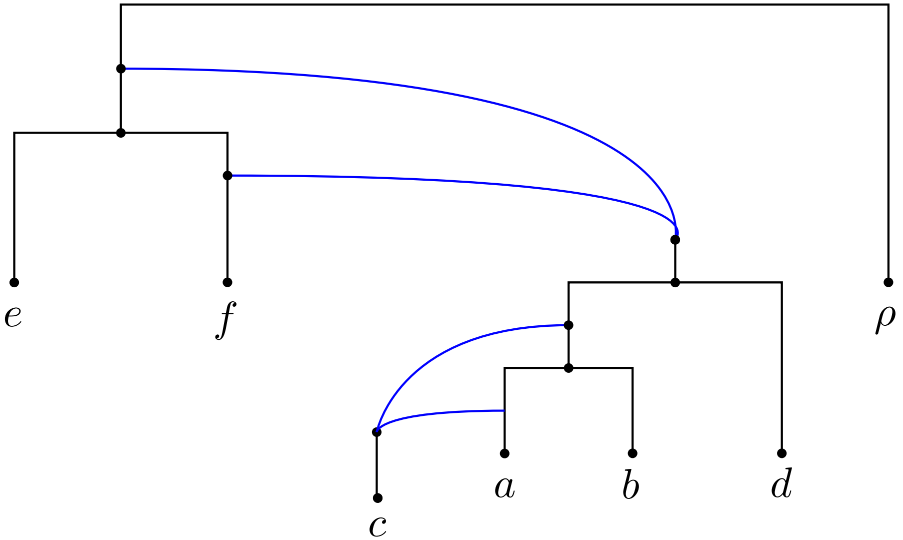
Answer: Network *displaying* both trees with a minimal number of hybridization (reticulation) nodes: **hybridization network**

Using MAAFs to construct hybridization networks





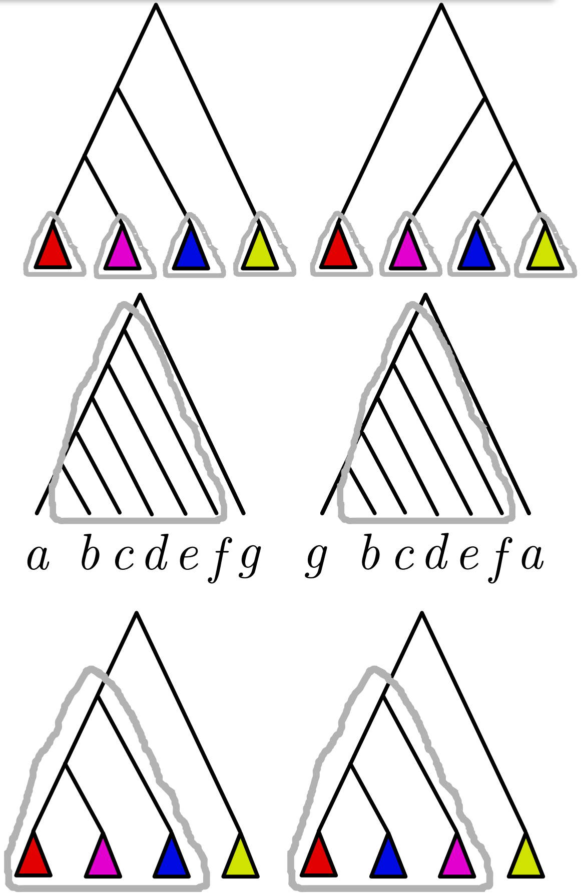




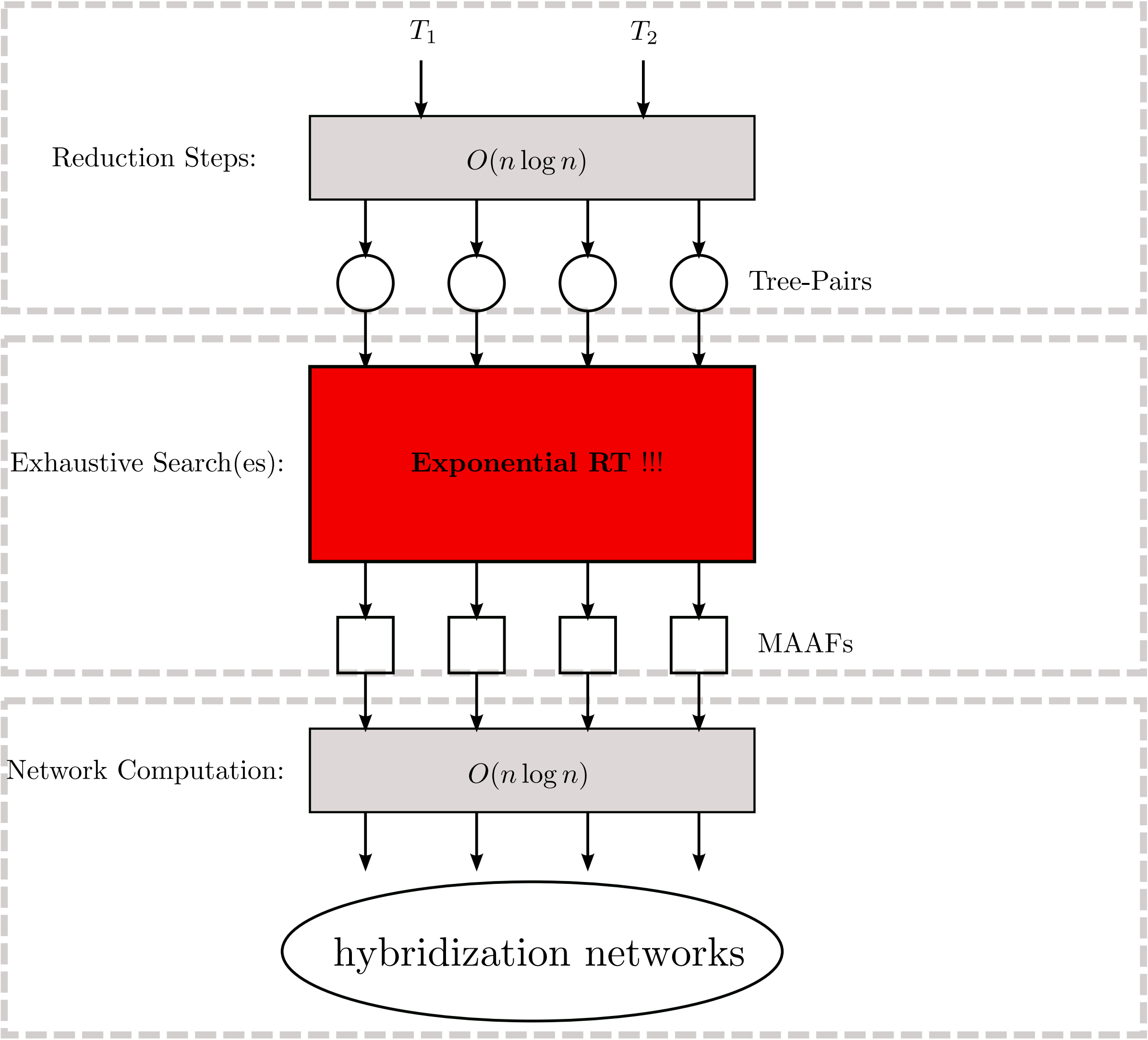
Results

- NP-hard
- FPT in the reticulation number r of the network $O(3.18^r\ n)$
- FPT in the level k of the network $O(3.18^k\ n)$

> Reduction steps:
  - Subtree reduction
  - Chain reduction
  - Cluster reduction



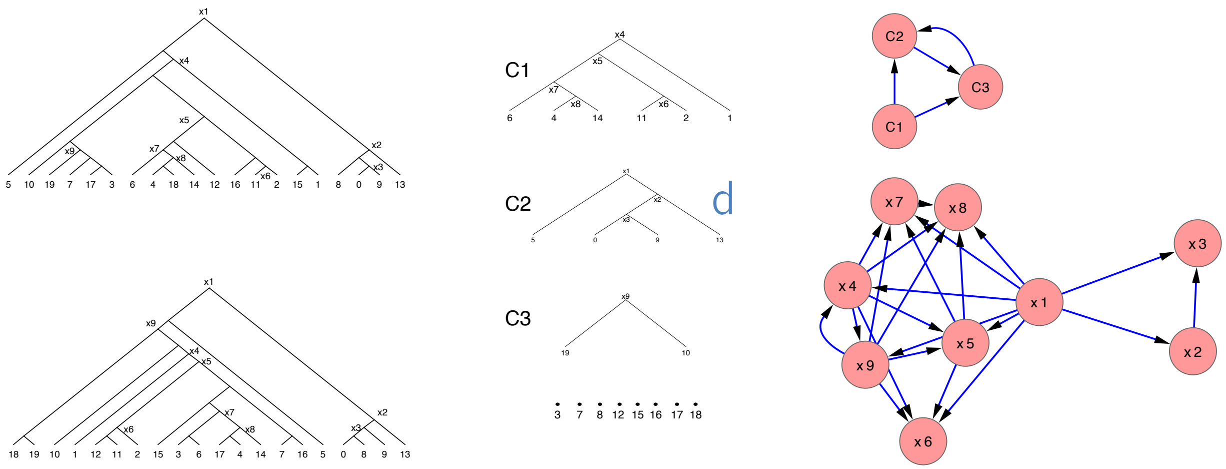
Using MAAFs to construct hybridization networks



Results – approx (connection with the DFVS)

- no 1.36-approximation, unless P=NP
- no (2 − ε)-approximation, unless the unique games conjecture fails
- O(log(r)loglog(r))- approximation
- d(c+1)-approximation

AAF = AF **c** + DFVS **d**



Kelk et al. Cycle killer...qu'est-ce que c'est? On the comparative approximability of hybridization number and directed feedback vertex set 2012
van Iersel et al. A practical approximation algorithm for solving massive instances of hybridization number. 2012

Results – approx (connection with the DFVS)

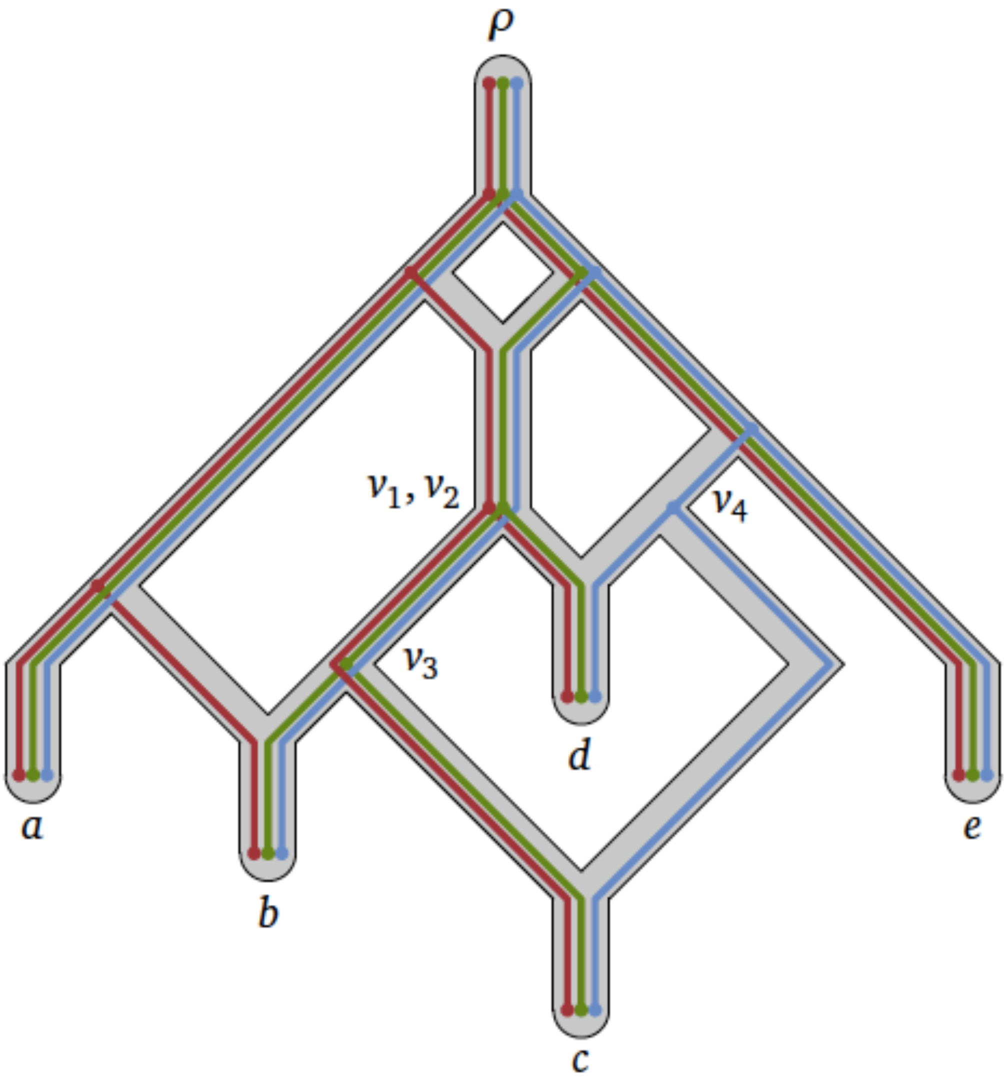
- no 1.36-approximation, unless P=NP
- no (2 – ε)-approximation, unless the unique games conjecture fails
- O(log(r)loglog(r))- approximation
- **d(c+1)-approximation**

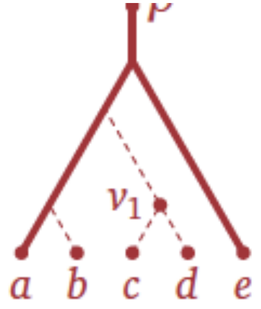
$$\text{AAF} = \text{AF}\ \boxed{c}^{3} + \text{DFVS}\ \boxed{d}^{1}$$

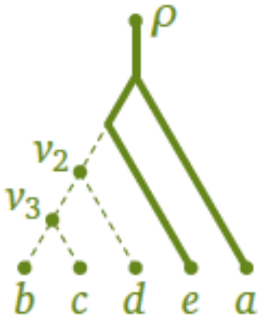
Using the 4-approximation on a normal laptop, we managed to construct networks with up to 10,000 leaves and up to 10,000 reticulations within 10 minutes!

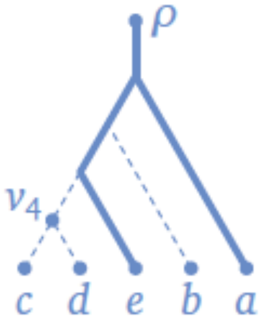
Kelk et al. Cycle killer...qu'est-ce que c'est? On the comparative approximability of hybridization number and directed feedback vertex set 2012
van Iersel et al. A practical approximation algorithm for solving massive instances of hybridization number. 2012

More than 2 trees



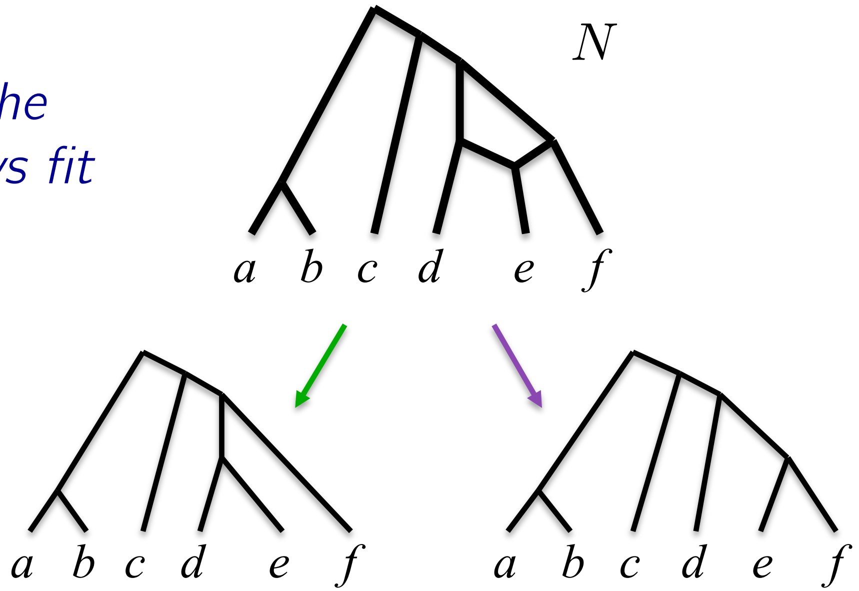






Phylogenetic supernetwork inference

An optimization problem where a candidate network is evaluated *on the basis of how well the trees it displays fit the data:*



Many possible formulations:

Data:

Clusters of taxa: $\{a,b\},\{d,e\},\{d,e,f\},\{a,b,c,d,e,f\},\{e,f\},\{c,d,e,f\},\ldots$

Goal:

Find the network N with the lower hybridization number such that the input clusters are `explained' by one of the trees displayed by N

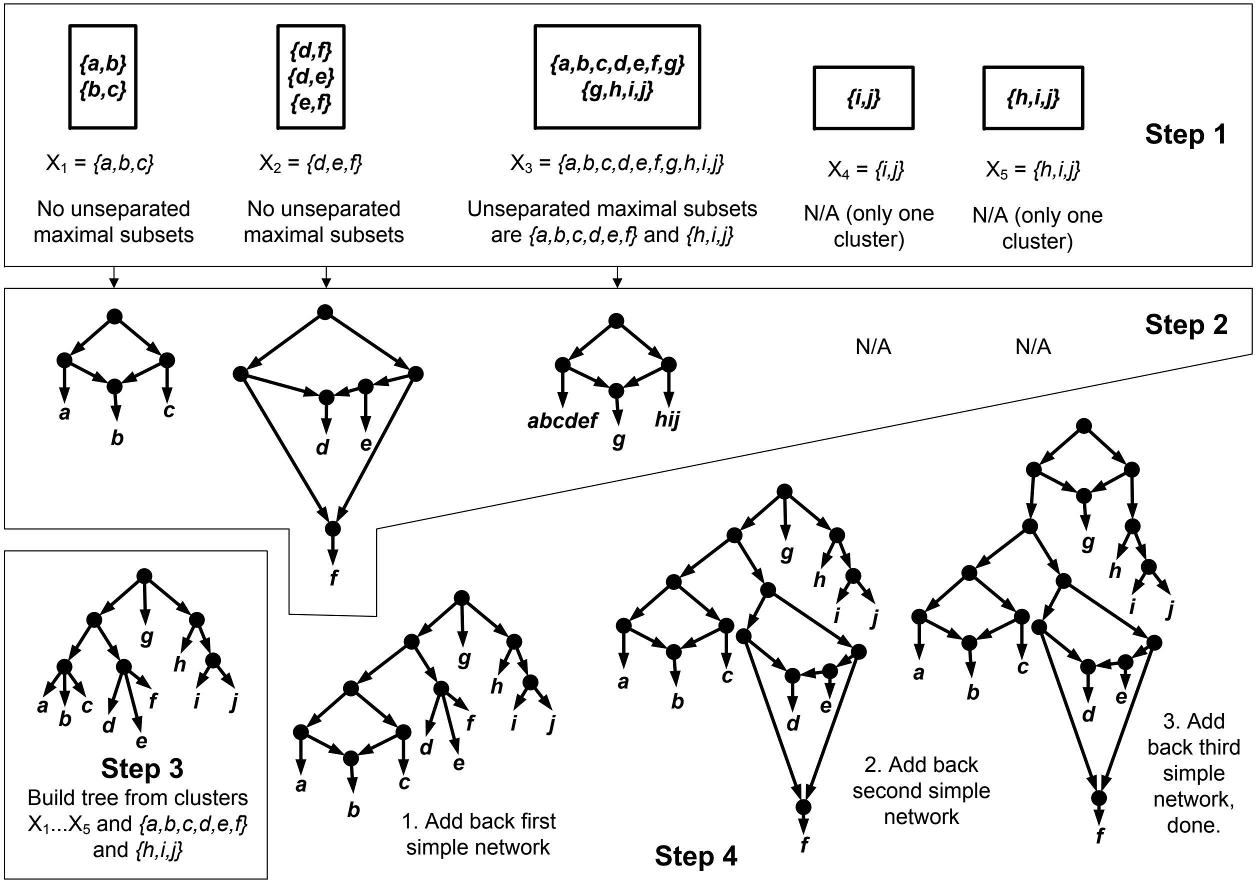
subject to constraints on the complexity of N

Clusters

- cluster containment: NP-hard
- minimization NP-hard, APX-hard
- A possible solution … topological constraints:
  - **galled trees** (level-1 networks)… it does not always exist
  - **galled networks** (if every reticulation in N has a *tree cycle*)… still NP-hard
  - **level-k networks** … still NP-hard

Clusters

CASS algorithm : search for the level-k network containing a set of clusters (exact for level-1 and level-2 networks)



Step 1

{a,b}
{b,c} — X_1 = *{a,b,c}* — No unseparated maximal subsets

{d,f}
{d,e}
{e,f} — X_2 = *{d,e,f}* — No unseparated maximal subsets

{a,b,c,d,e,f,g}
{g,h,i,j} — X_3 = *{a,b,c,d,e,f,g,h,i,j}* — Unseparated maximal subsets are *{a,b,c,d,e,f}* and *{h,i,j}*

{i,j} — X_4 = *{i,j}* — N/A (only one cluster)

{h,i,j} — X_5 = *{h,i,j}* — N/A (only one cluster)

Step 2

N/A

N/A

Step 3

Build tree from clusters X_1...X_5 and *{a,b,c,d,e,f}* and *{h,i,j}*

Step 4

1. Add back first simple network

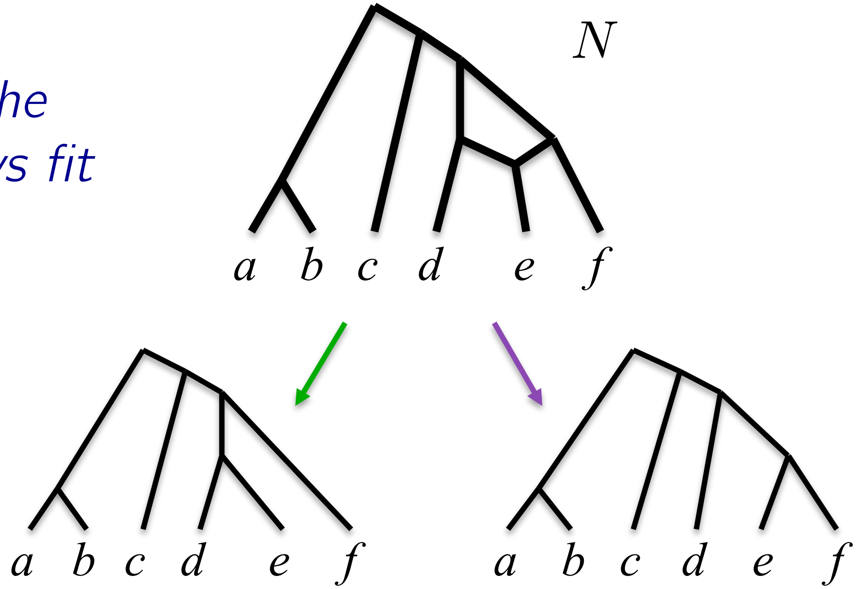
2. Add back second simple network

3. Add back third simple network, done.

van Iersel et al. Phylogenetic networks do not need to be complex: using fewer reticulations to represent conflicting clusters. 2010

Phylogenetic network inference

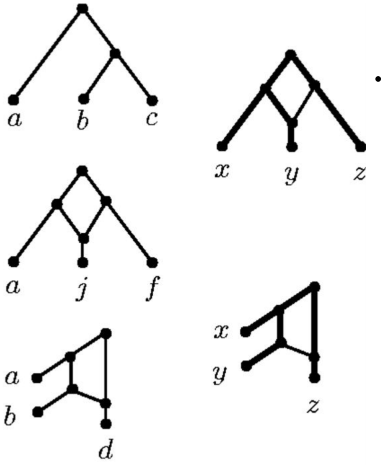
An optimization problem where a candidate network is evaluated *on the basis of how well the trees it displays fit the data:*



Many possible formulations:

Data:

Any trinets on the same taxa:
(inferred from other data)

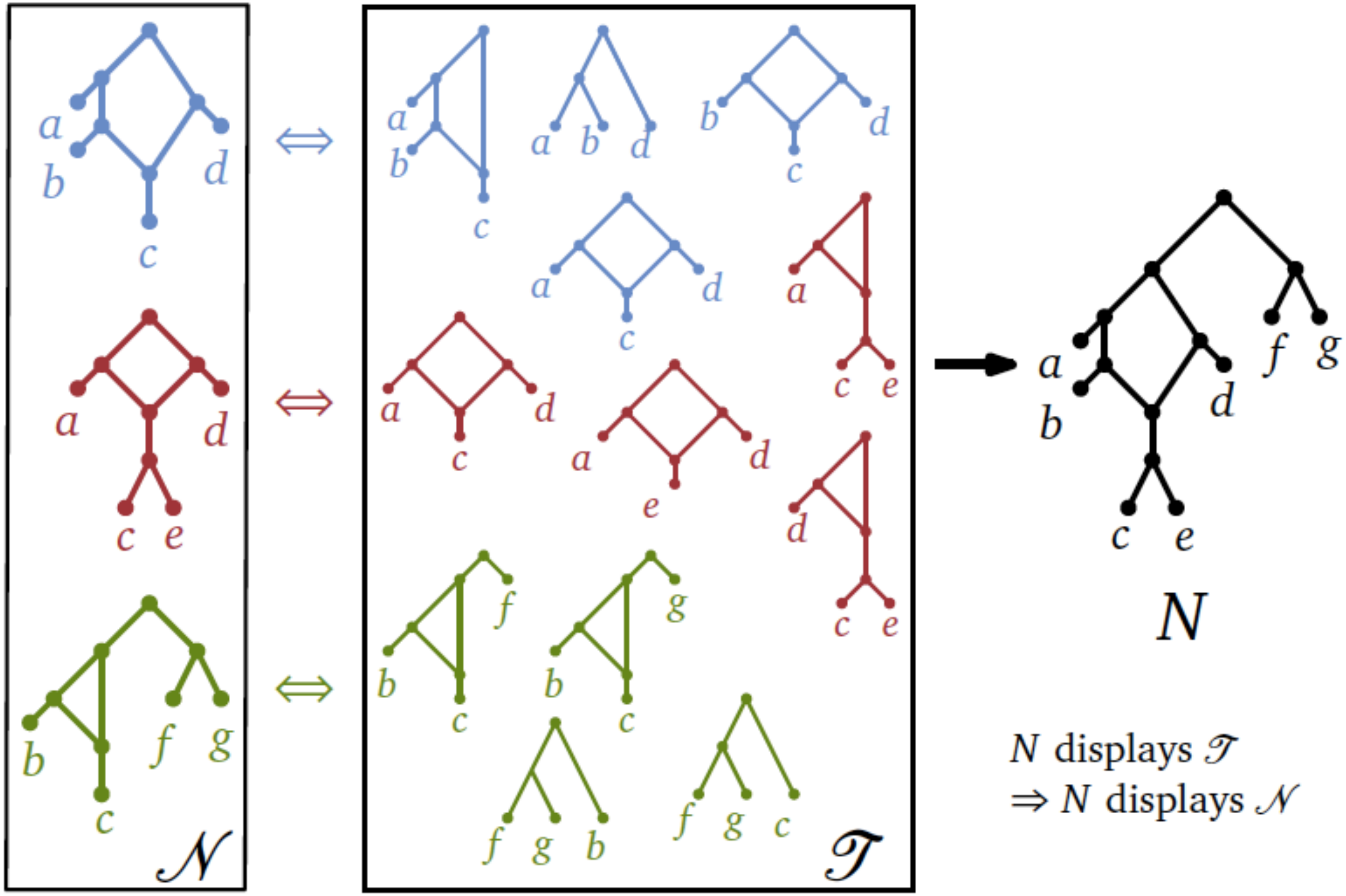


Goal:

Find the network N with the lower hybridization number such that the input trees are `consistent’ with the N

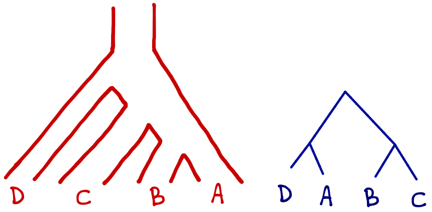
subject to constraints on the complexity of N

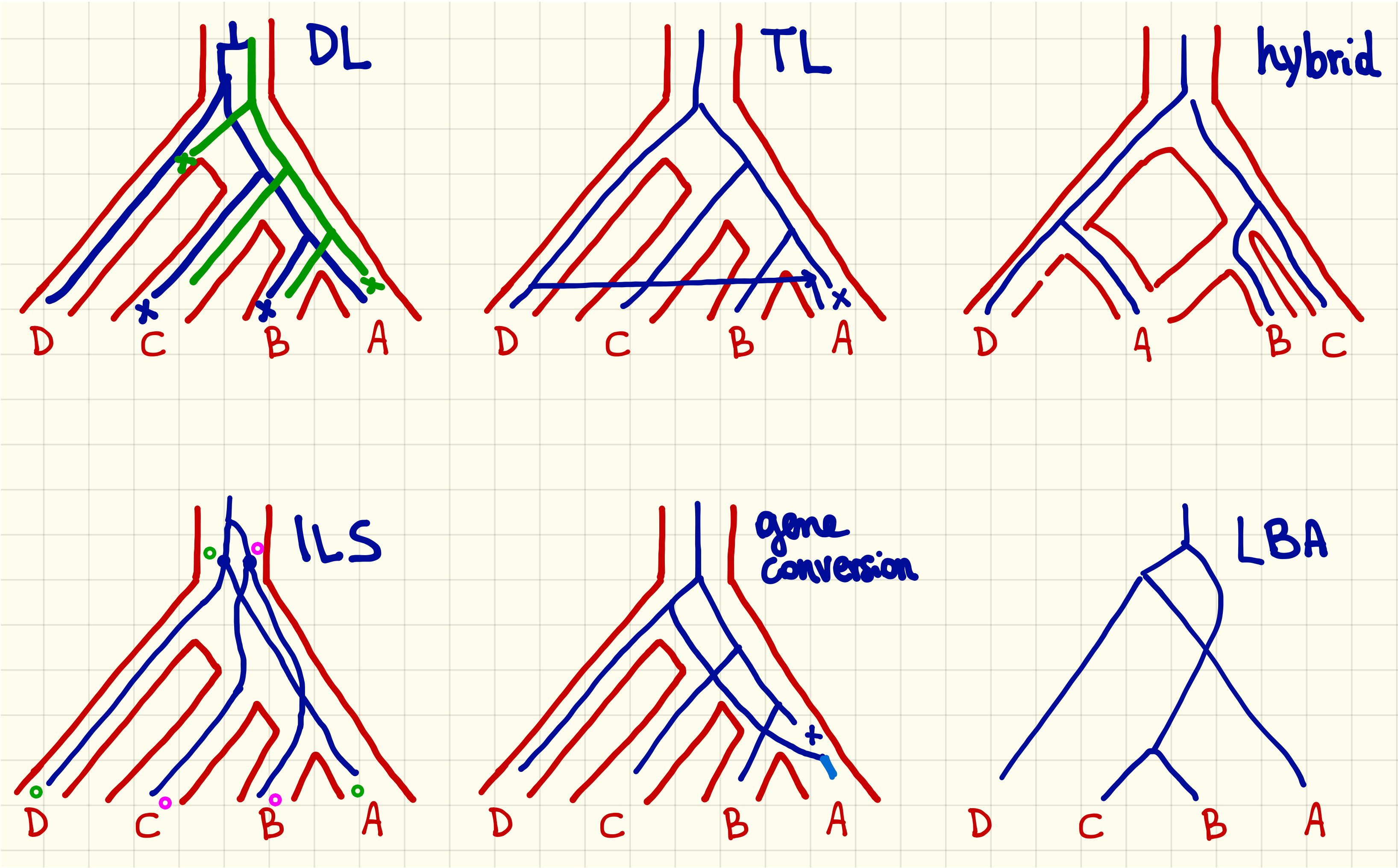
Trinets



N displays $\mathcal{T}$
$\Rightarrow N$ displays $\mathcal{N}$

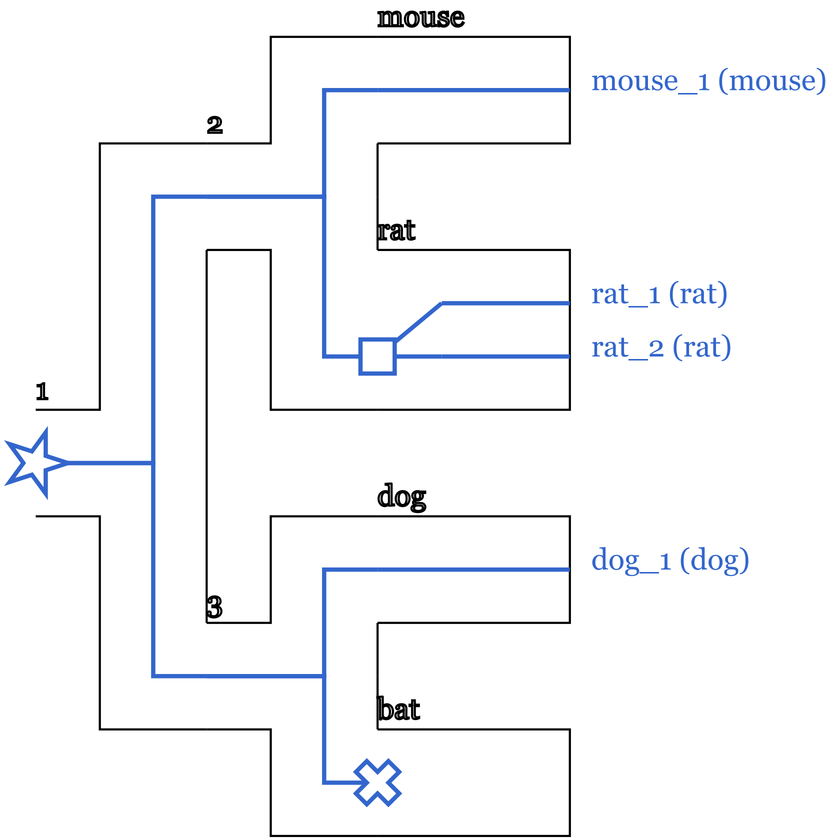
Species/gene trees





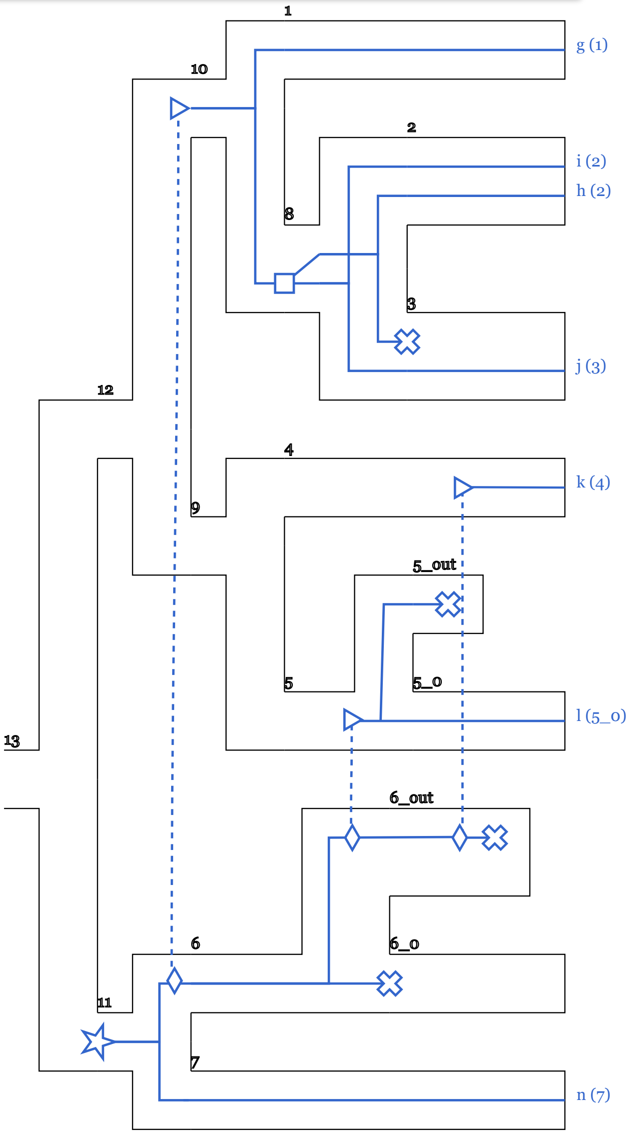
DL model

- Speciation (S) are the only possible events shaping species histories
- Speciation (S), duplication (D) and loss (L) are the possible events shaping gene histories
- Each contemporary gene is a leaf of G and is associated to the corresponding species of S in which this gene is collected
- Each S in G happens at S in S
- Each S and D event gives birth to exactly two genes
- The evolution of G along S goes forward in time
- L events in G are supposed to happen at a S in S



DTL model

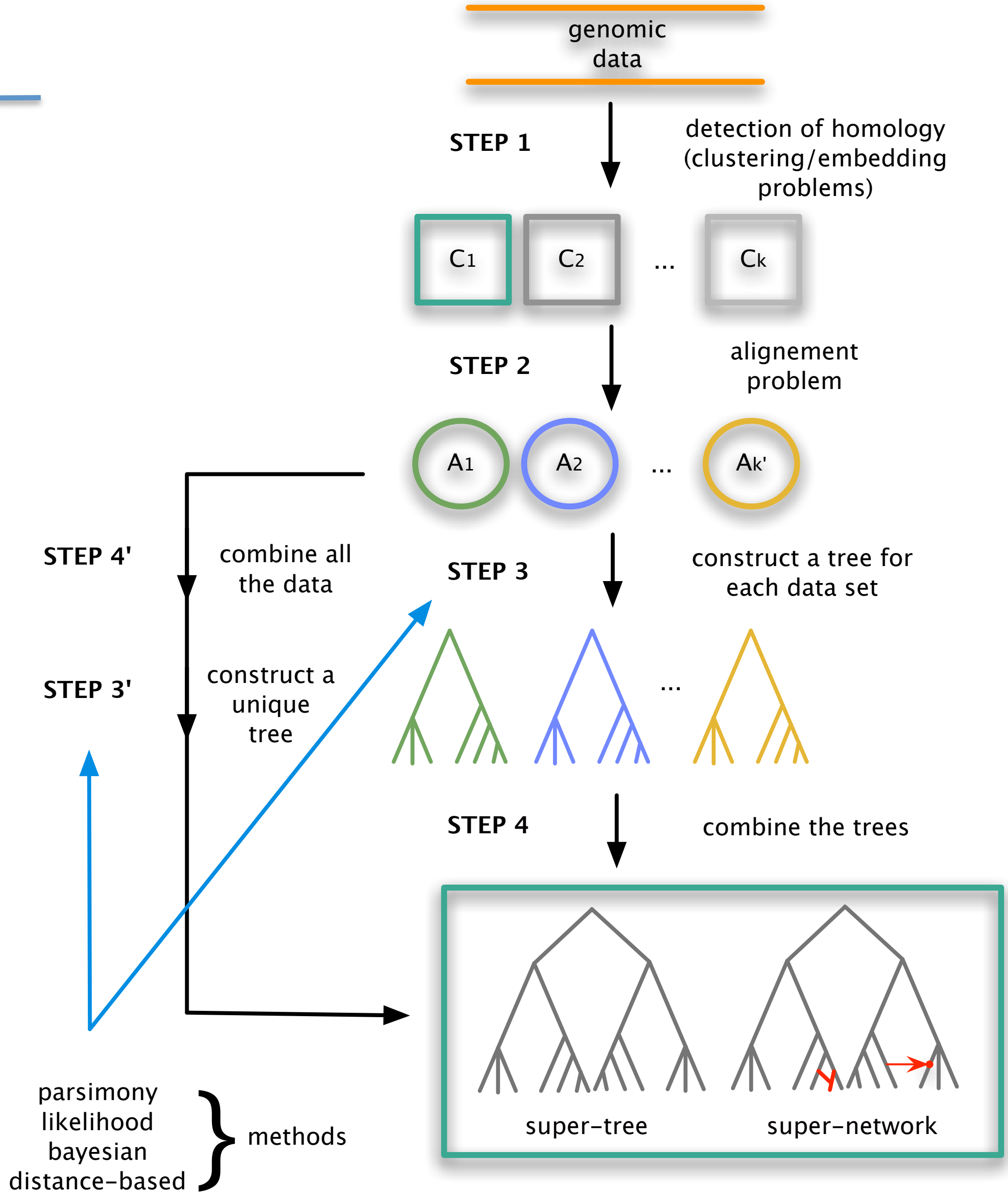
- Speciation (S) are the only possible events shaping species histories
- Speciation (S), duplication (D) loss (L) and transfers (T) between sampled/ unsampled species are the possible events shaping gene histories
- Each contemporary gene is a leaf of G and is associated to the corresponding species of S in which this gene is collected
- Each S in G happens at S in S
- Each S and D event gives birth to exactly two genes
- The evolution of G along S goes forward in time
- Each T event is happens between two co-existing species.



Evolution of applications

- Find one of the "*good*" scenarios (e.g. to detect homology/ paralogy)
  - DTL The best-performing parsimony-based algorithm to date for ranked species trees (i.e. we suppose to have knowledge of the relative order in which nodes appear in the tree) $O(n^2 m)$
  - DTL A modification of the algorithm can be used to reconcile against undated species trees $O(n m)$
  - DTL Unrooted/non-binary gene trees as input $O(m n^2 (3^d - 2^{d+1}))$ where d is the maximum out-degree of any node in G
  - DTLI A algorithm for ranked species trees $O(m(n^2 +n n_k 2^k) 2^k)$ where k is the maximum number of ILS branches that are connected in S and n_k is the number of sets of connected ILS branches of S (e.g., if we have a group of three adjacent ILS branches, k = 3 while nk = 1)
  - DL on networks $O(h^2 m n)$ where h is the number of nodes with 2 parents in the network
  - DTL on LGT networks $O(n m)$

Phylogenomics



genomic
data

STEP 1

detection of homology
(clustering/embedding
problems)

C_1 C_2 ... C_k

STEP 2

alignement
problem

A_1 A_2 ... $A_{k'}$

STEP 4'

combine all
the data

STEP 3'

construct a
unique
tree

STEP 3

construct a tree for
each data set

...

STEP 4

combine the trees

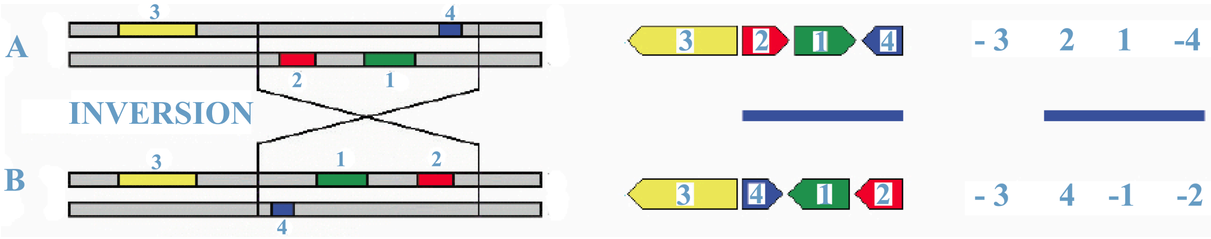
super-tree

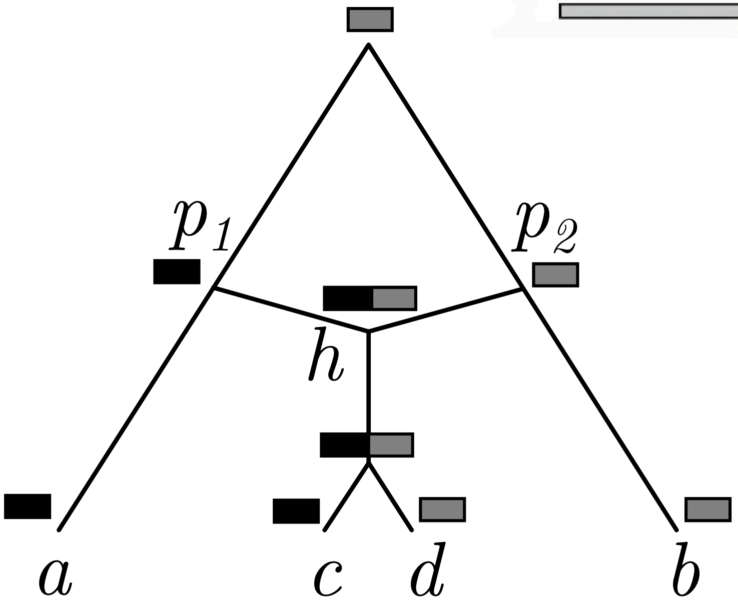
super-network

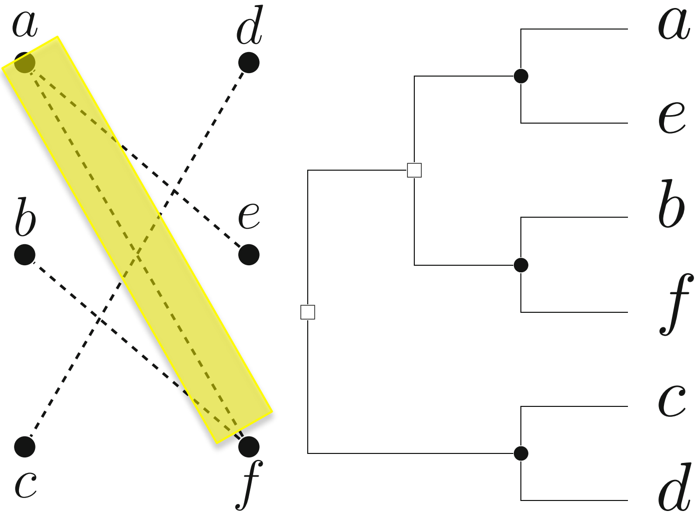
parsimony
likelihood
bayesian
distance-based
} methods

What I did not even mention

- sequence analyses (recombination detection, genome rearrangements such as sorting by reversals, or DCJ, orthology detection)

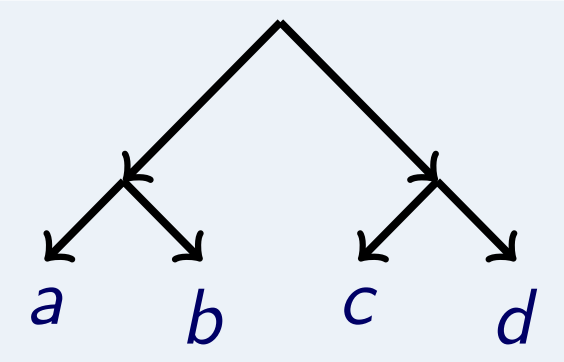






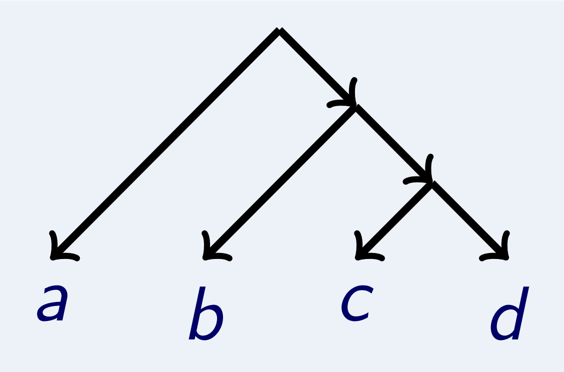
What I did not even mention

- sequence analyses (recombination detection, genome rearrangements such as sorting by reversals, or DCJ, orthology detection)
- comparing trees/networks (edit distances, confidence value...)



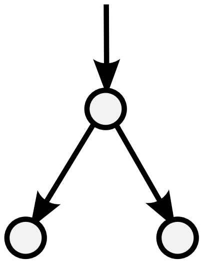
What I did not even mention

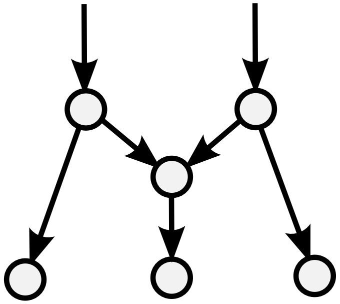
- sequence analyses (recombination detection, genome rearrangements such as sorting by reversals, or DCJ, orthology detection)
- comparing trees/networks (edit distances, confidence value...)

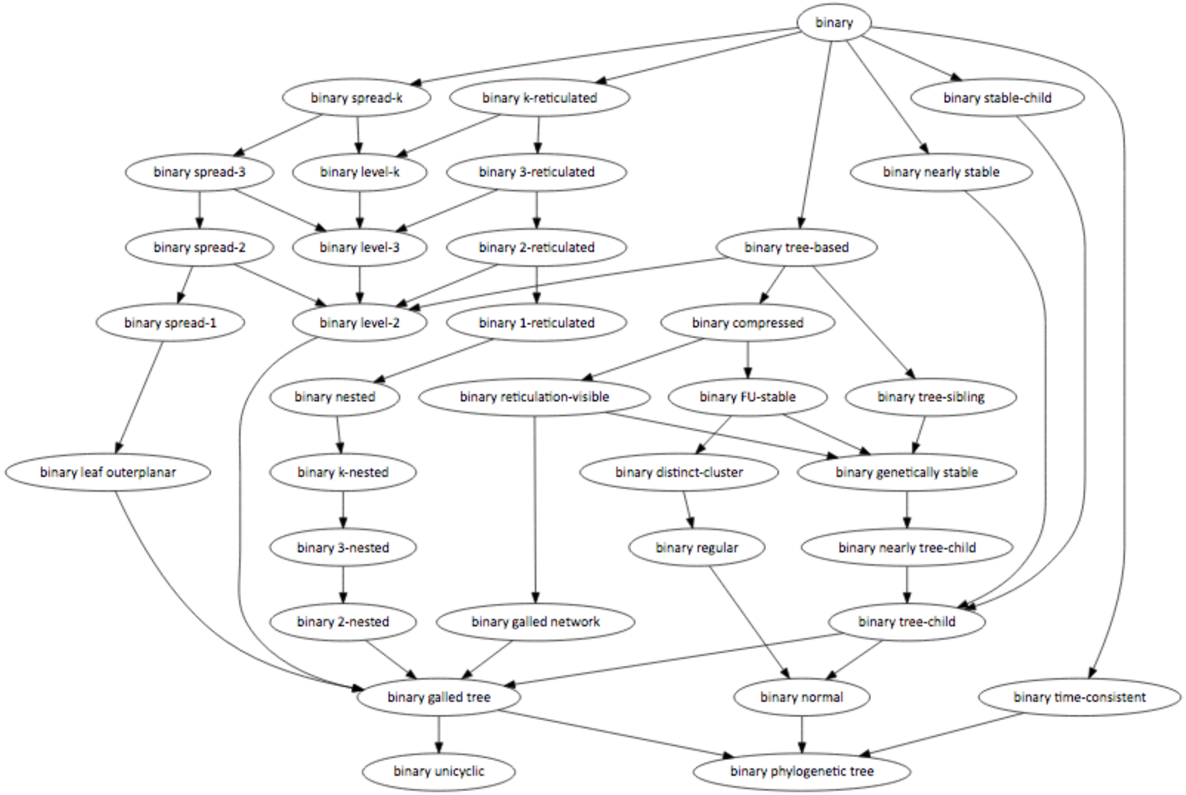


What I did not even mention

- sequence analyses (recombination detection, genome rearrangements such as sorting by reversals, or DCJ, orthology detection)
- comparing trees/networks (edit distances, confidence value...)
- generating/counting/studying classes of trees/networks

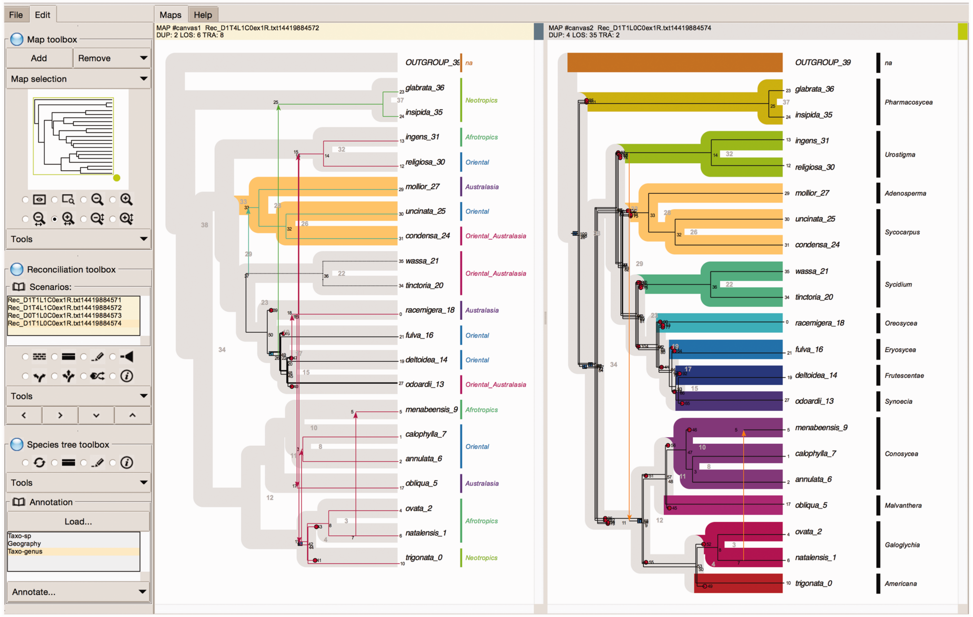






What I did not even mention

- sequence analyses (recombination detection, genome rearrangements such as sorting by reversals, or DCJ, orthology detection)
- comparing trees/networks (edit distances, confidence value...)
- generating trees/networks
- drawing trees and networks
- ...

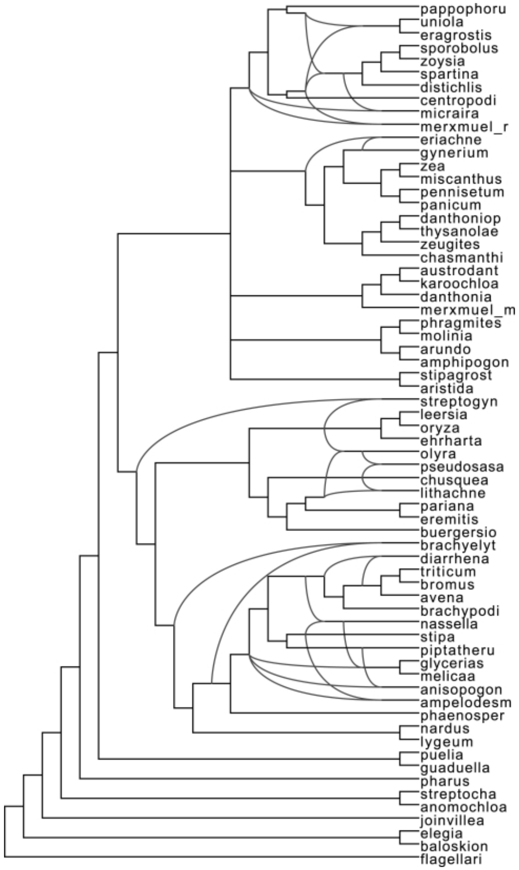


MAP #canvas1 Rec_D1T4L1C0ex1R.txt14419884572
DUP: 2 LOS: 6 TRA: 8

MAP #canvas2 Rec_D1T1L0C0ex1R.txt14419884574
DUP: 4 LOS: 35 TRA: 2

Scenarios:
Rec_D1T1L1C0ex1R.txt14419884571
Rec_D1T4L1C0ex1R.txt14419884572
Rec_D0T1L0C0ex1R.txt14419884573
Rec_D1T1L0C0ex1R.txt14419884574

Annotation:
Taxo-sp
Geography
Taxo-genus



Peer Community In

Looking for a way of publishing that is

- **transparent**,
- **made** by and for **researchers**,
- **independent** of publishing companies and
- **totally free** for authors and readers?

Winner of the
LIBER Award for Library Innovation
supported by
OCLC
LIBER 2020 ONLINE
08 Jun 2020
www.liberconference.eu
@LIBERconference | #LIBER2020

Check us out at https://peercommunityin.org
and submit to **PCI Math Comp Biol** https:/mcb.peercommunityin.org/