

Lecture 5

blerina sinaimer

Outline

- Modelling through graphs

- NGS

- (Co)-phylogeny

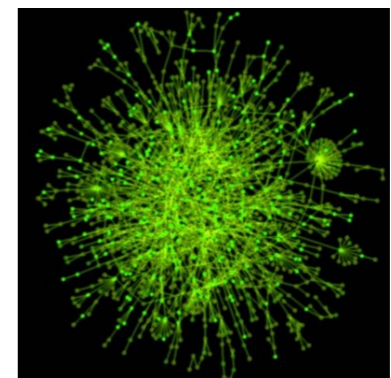
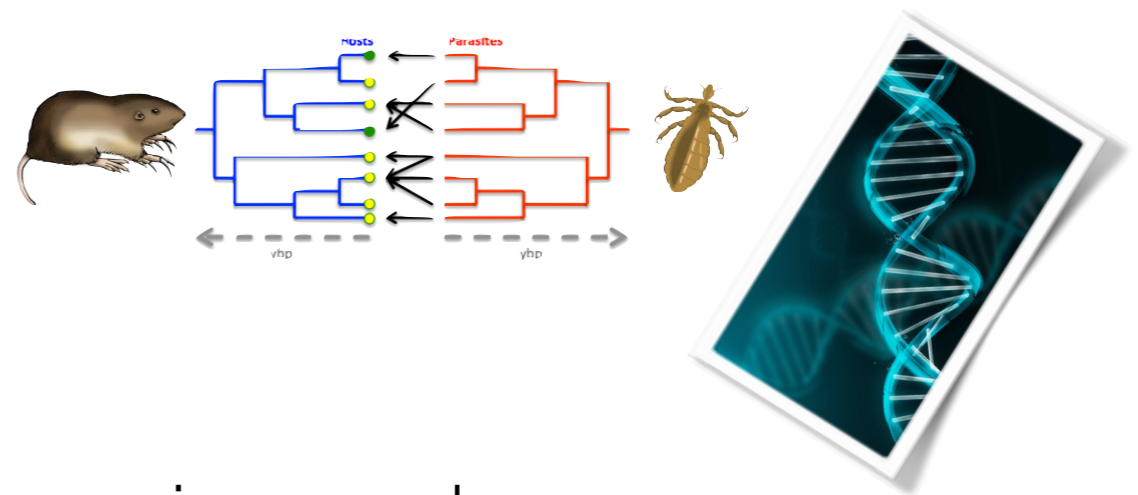
- Two other examples of modelling using graphs

- One solution maybe not informative. Enumerate all the solutions.

- Big data challenge

- Efficient algorithms, efficient data structures

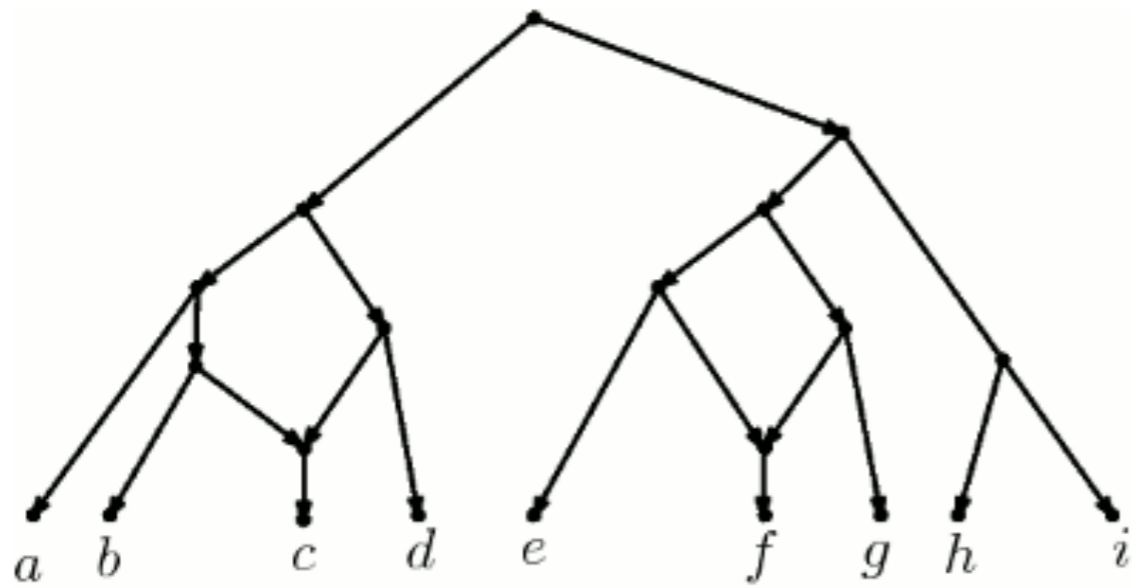
- Local view of large substructures.



phylogenetic networks

Trees?

- Phylogenetic Networks



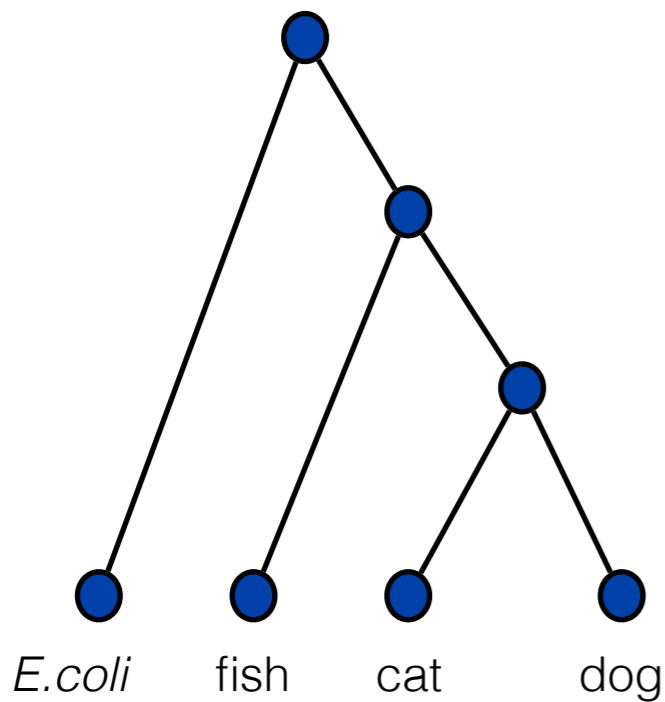
Classical assumption



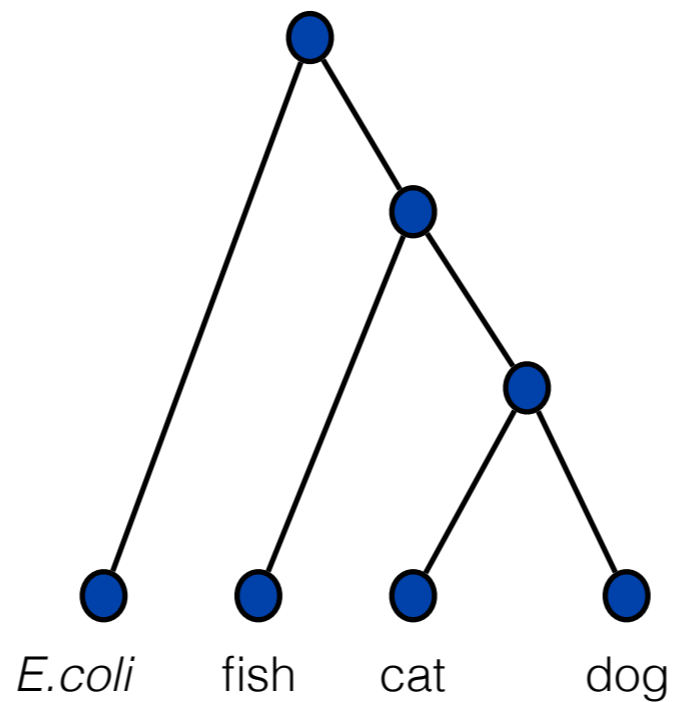
gene 1

gene 2

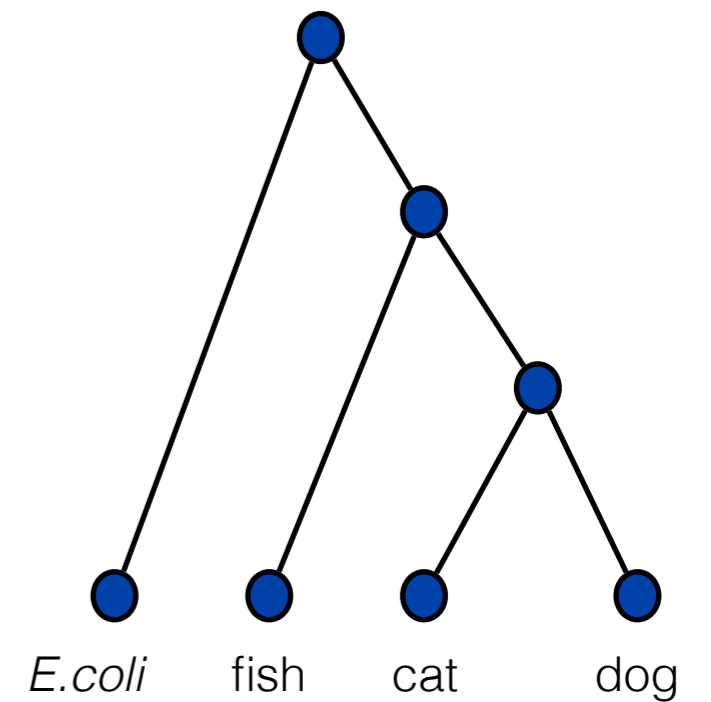
gene 3



gene 1

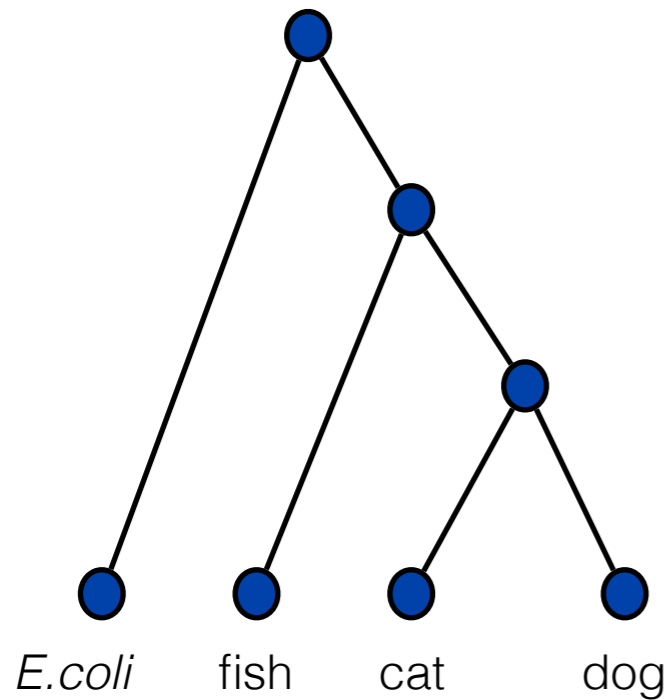


gene 2



gene 3

Classical assumption



Species tree (same as the gene trees)

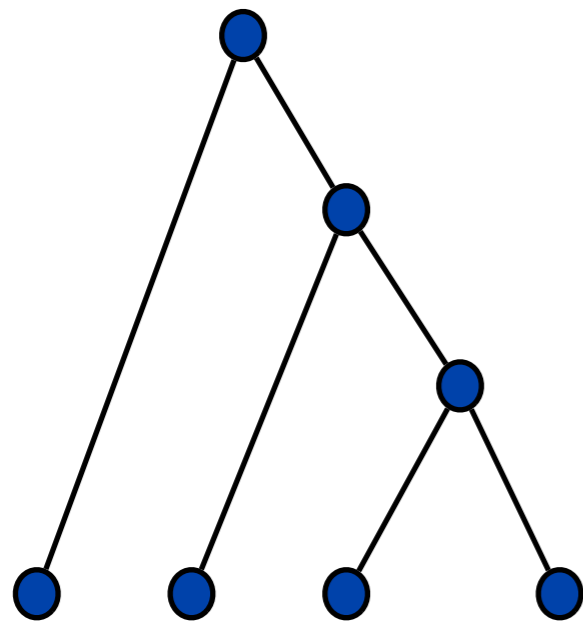
Classical assumption



gene 1

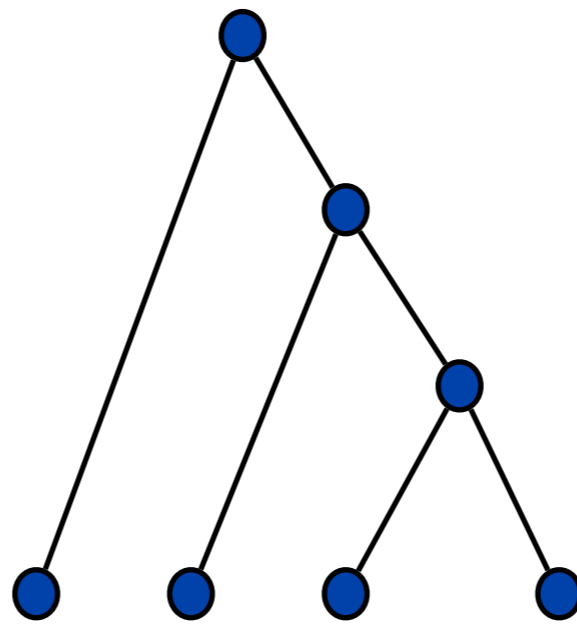
gene 2

gene 3



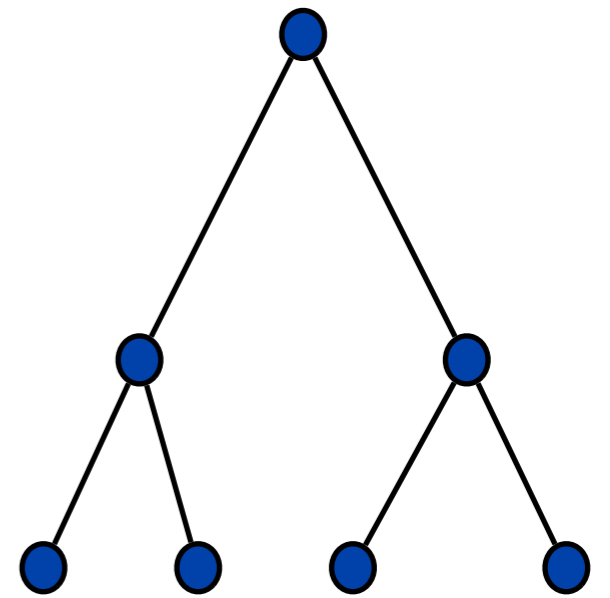
Plant 1 Plant 2 Plant 3 Plant 4

gene 1



Plant 1 Plant 4 Plant 3 Plant 2

gene 2



Plant 1 Plant 2 Plant 3 Plant 4

gene 3

Classical assumption

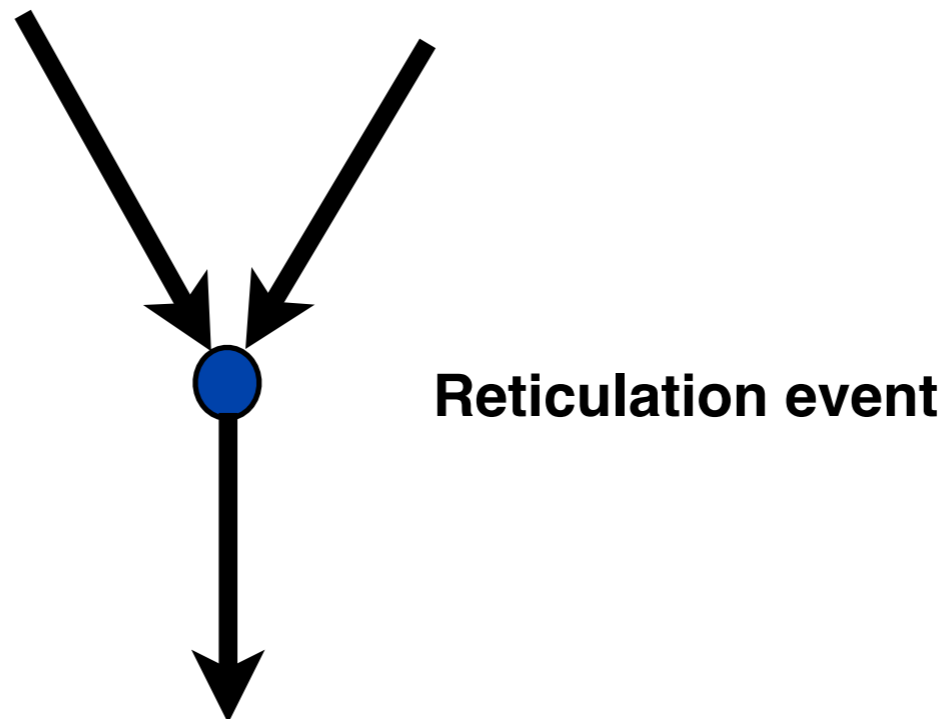


Species tree?

There are often multiple conflicting (“incongruent”) tree signals involved. There are actually many different evolutionary phenomena that can cause multiple conflicting tree signals to arise.

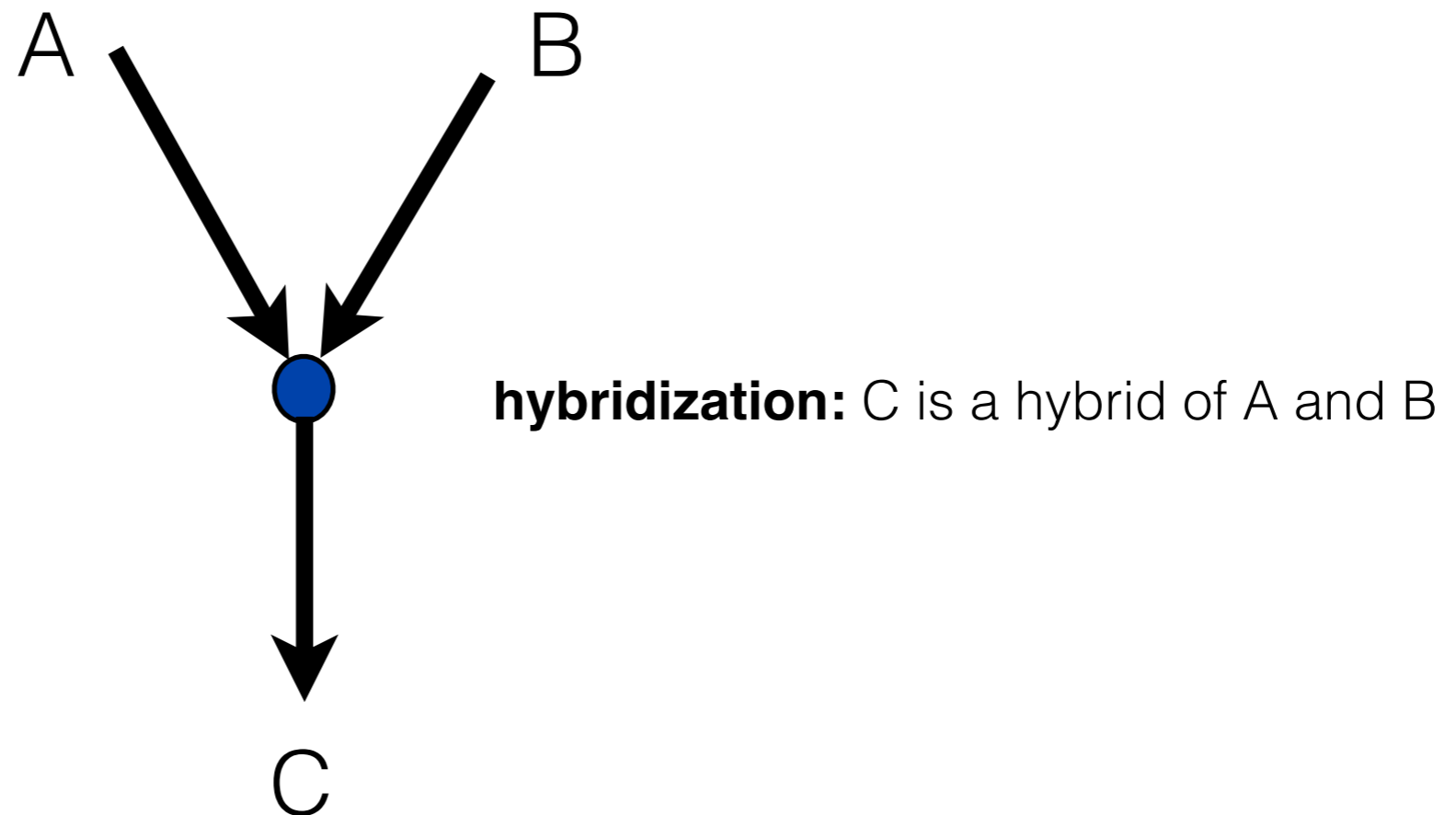
Homology is the existence of shared ancestry between a pair of structures, or genes, in different species. A common example of homologous structures in evolutionary biology are the wings of bats and the arms of primates.

- **recombination**
- **hybridization**
- **duplication/loss**
- ...



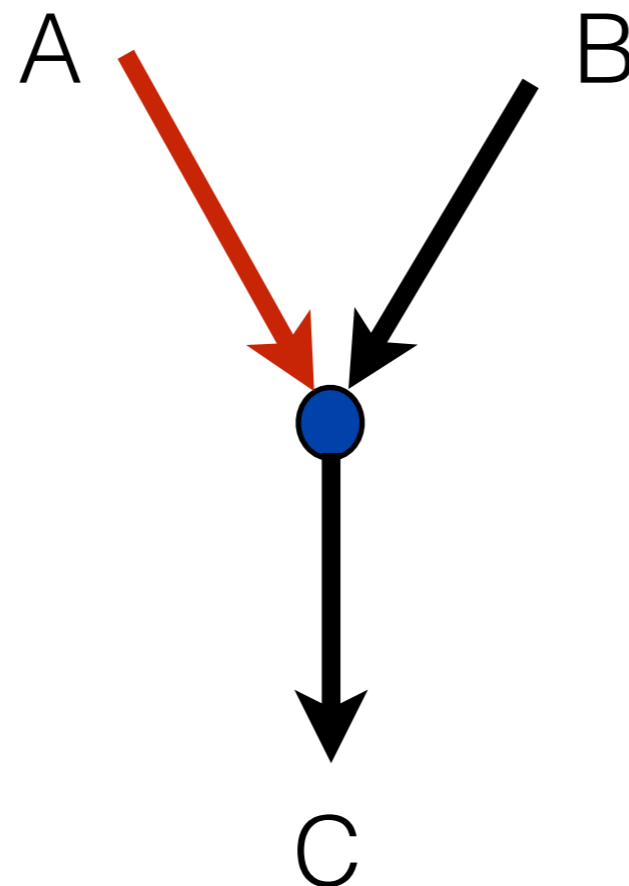
Homology is the existence of shared ancestry between a pair of structures, or genes, in different species. A common example of homologous structures in evolutionary biology are the wings of bats and the arms of primates.

- **recombination**
- **hybridization**
- **duplication/loss**
- ...



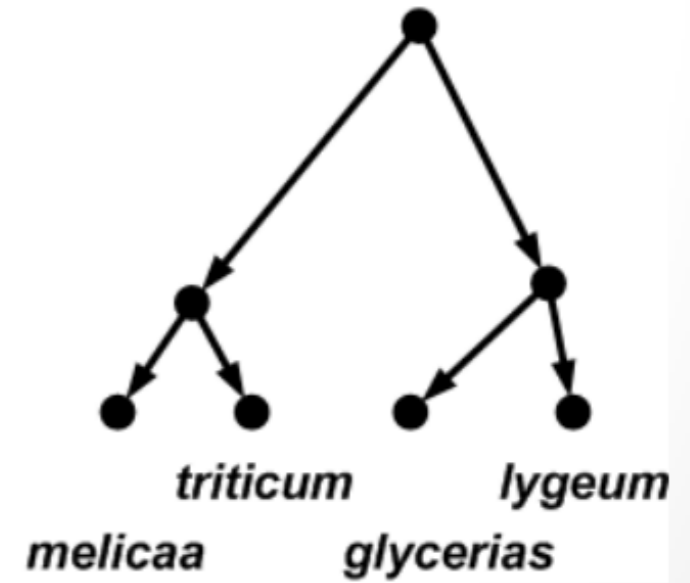
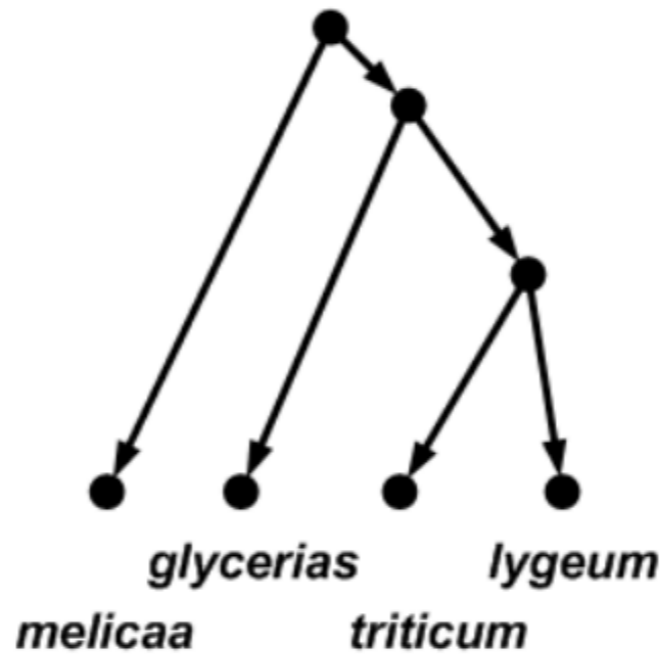
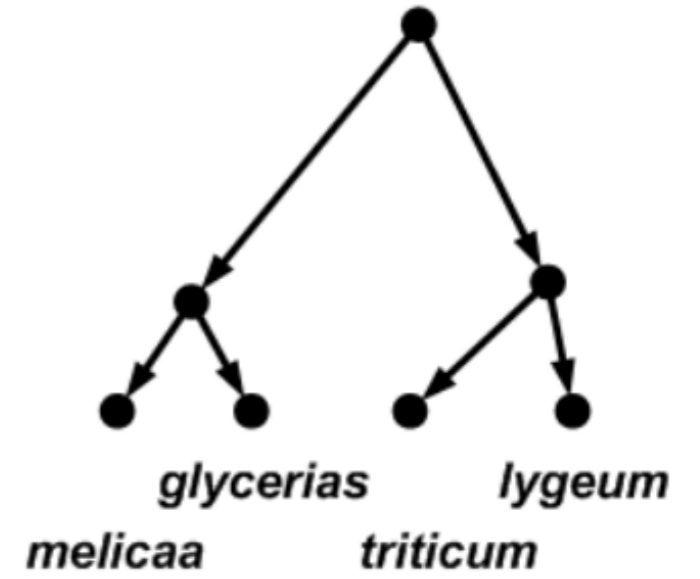
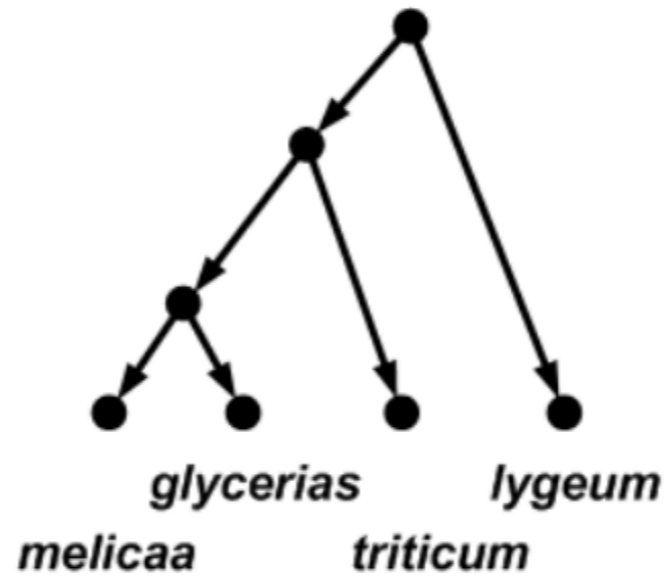
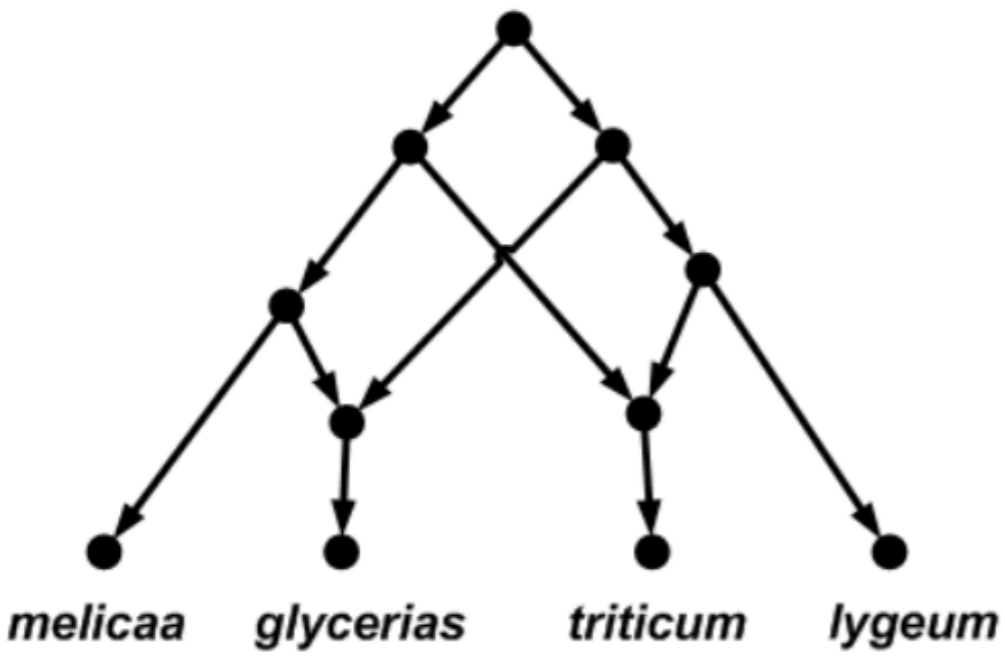
Homology is the existence of shared ancestry between a pair of structures, or genes, in different species. A common example of homologous structures in evolutionary biology are the wings of bats and the arms of primates.

- **recombination**
- **hybridization**
- **duplication/loss**
- ...



Horizontal Gene Transfer:

a transfer of one or more genes from donor A into recipient B (emphasizes asymmetry)



Phylogenetic networks

Definition 0.1. A *phylogenetic \mathcal{X} -network*, or \mathcal{X} -network for short, N is an ordered pair (G, f) , where

- $G = (V, E)$ is a directed, acyclic graph (DAG) with $V = \{r\} \cup V_L \cup V_T \cup V_N$, where
 - $\text{indeg}(r) = 0$ (r is the *root* of N);
 - $\forall v \in V_L, \text{indeg}(v) = 1$ and $\text{outdeg}(v) = 0$ (V_L are the *leaves* of N);
 - $\forall v \in V_T, \text{indeg}(v) = 1$ and $\text{outdeg}(v) \geq 2$ (V_T are the *tree-nodes* of N); and,
 - $\forall v \in V_N, \text{indeg}(v) = 2$ and $\text{outdeg}(v) = 1$ (V_N are the *network-nodes* of N),

and $E \subseteq V \times V$ are the network's edges (we distinguish between *network-edges*, edges whose heads are network-nodes, and *tree-edges*, edges whose heads are tree-nodes).

- $f : V_L \rightarrow \mathcal{X}$ is the *leaf-labeling* function, which is a bijection from V_L to \mathcal{X} .

Phylogenetic networks

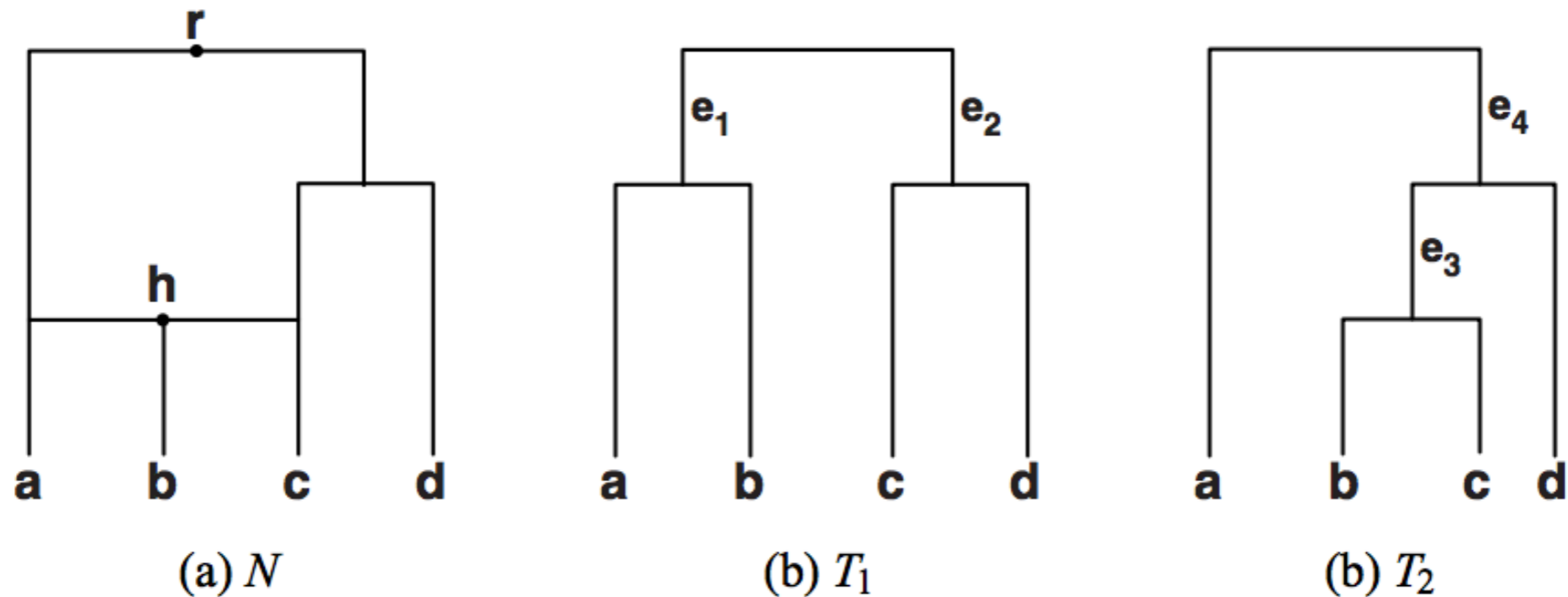


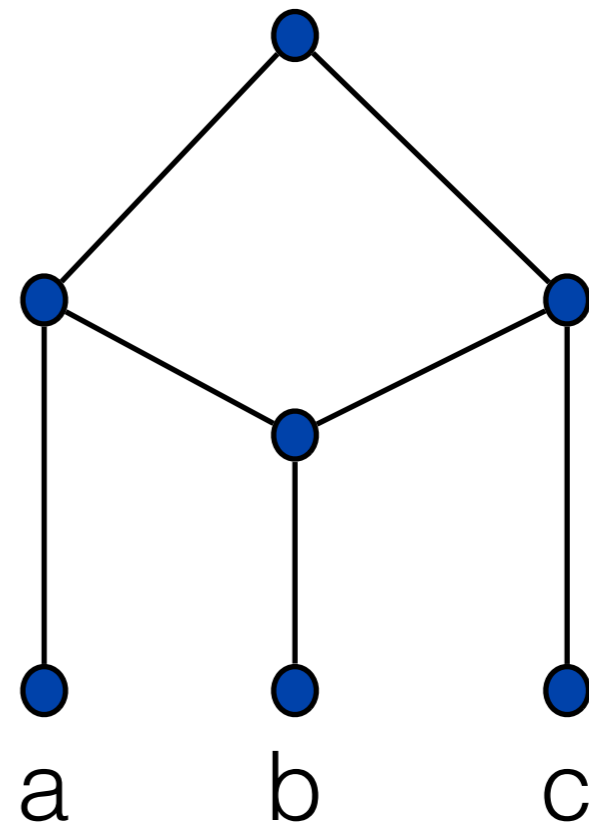
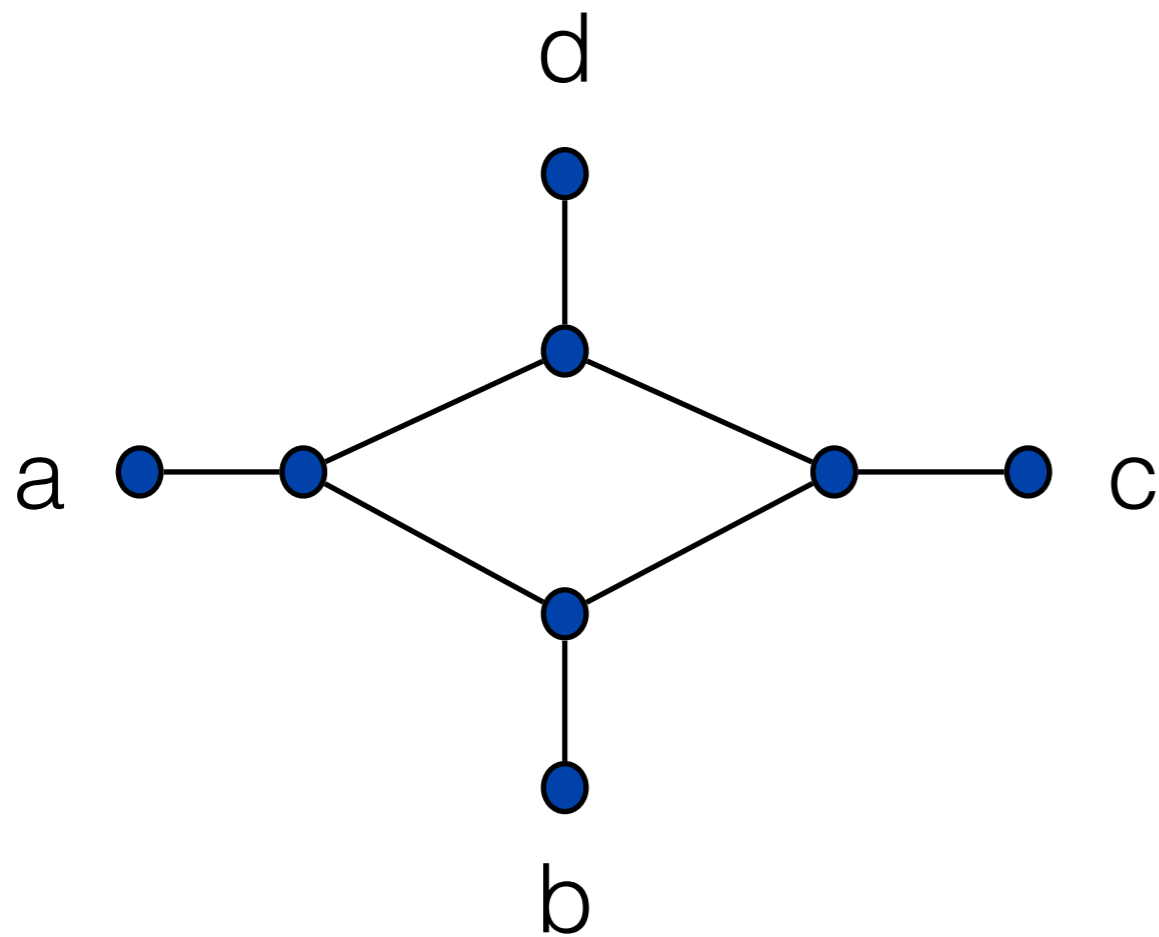
Fig. 1 (a) A phylogenetic \mathcal{X} -network, rooted at node r , with a single network-node, h , and with $\mathcal{X} = \{a, b, c, d\}$. The trees T_1 (b) and T_2 (c) are the elements of $\mathcal{T}(N)$.

Use of phylogenetic networks

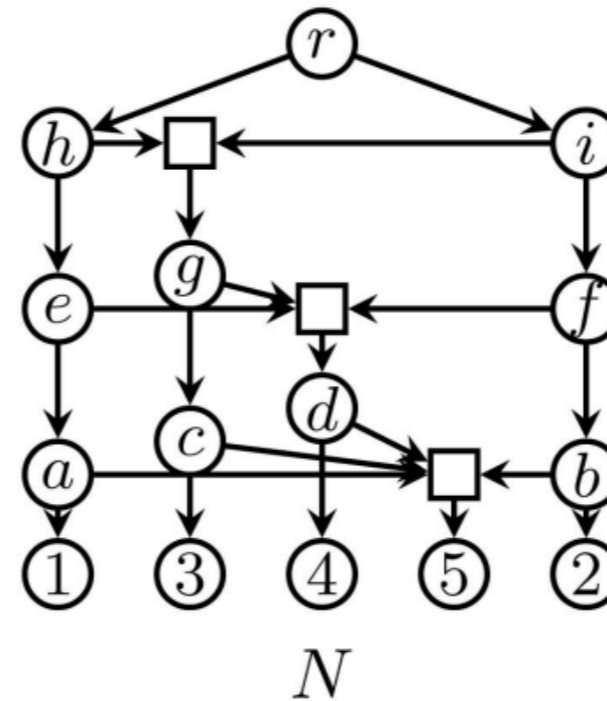
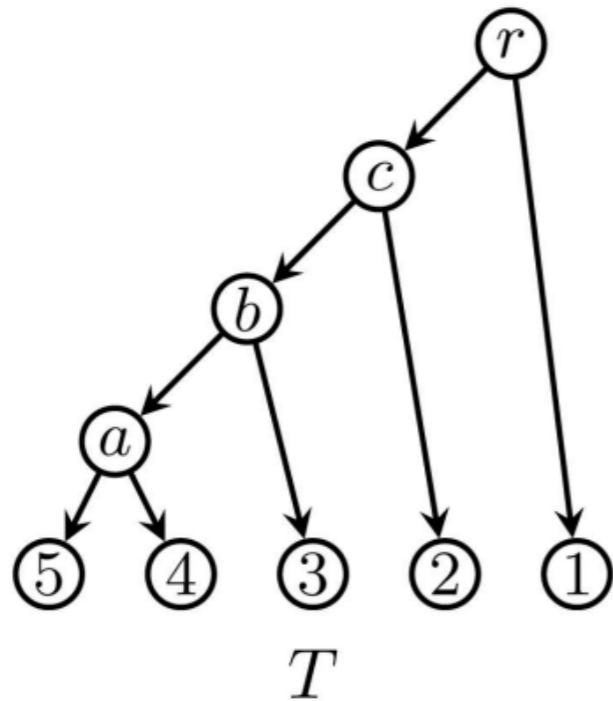
Phylogenetic networks can be used in two different ways.

- As a tool for visualizing incompatible data sets in a helpful manner, in which case we speak of an “*abstract*” phylogenetic network.
- As a representation of a putative evolutionary history involving reticulate events, in which case, the network is called “*explicit*.”

Rooted and unrooted phylogenetic networks



Robinson-Foulds distance



$$C(T) = \left\{ \{1\}, \{2\}, \{3\}, \{4\}, \{5\}, \{4, 5\}, \{3, 4, 5\}, \right. \\ \left. \{2, 3, 4, 5\}, \{1, 2, 3, 4, 5\} \right\},$$

$$C(N) = \left\{ \{1\}, \{2\}, \{3\}, \{4\}, \{5\}, \{1, 5\}, \{2, 5\}, \{3, 5\}, \right. \\ \{4, 5\}, \{1, 4, 5\}, \{2, 4, 5\}, \{3, 4, 5\}, \{1, 3, 4, 5\}, \\ \left. \{2, 3, 4, 5\}, \{1, 2, 3, 4, 5\} \right\}.$$

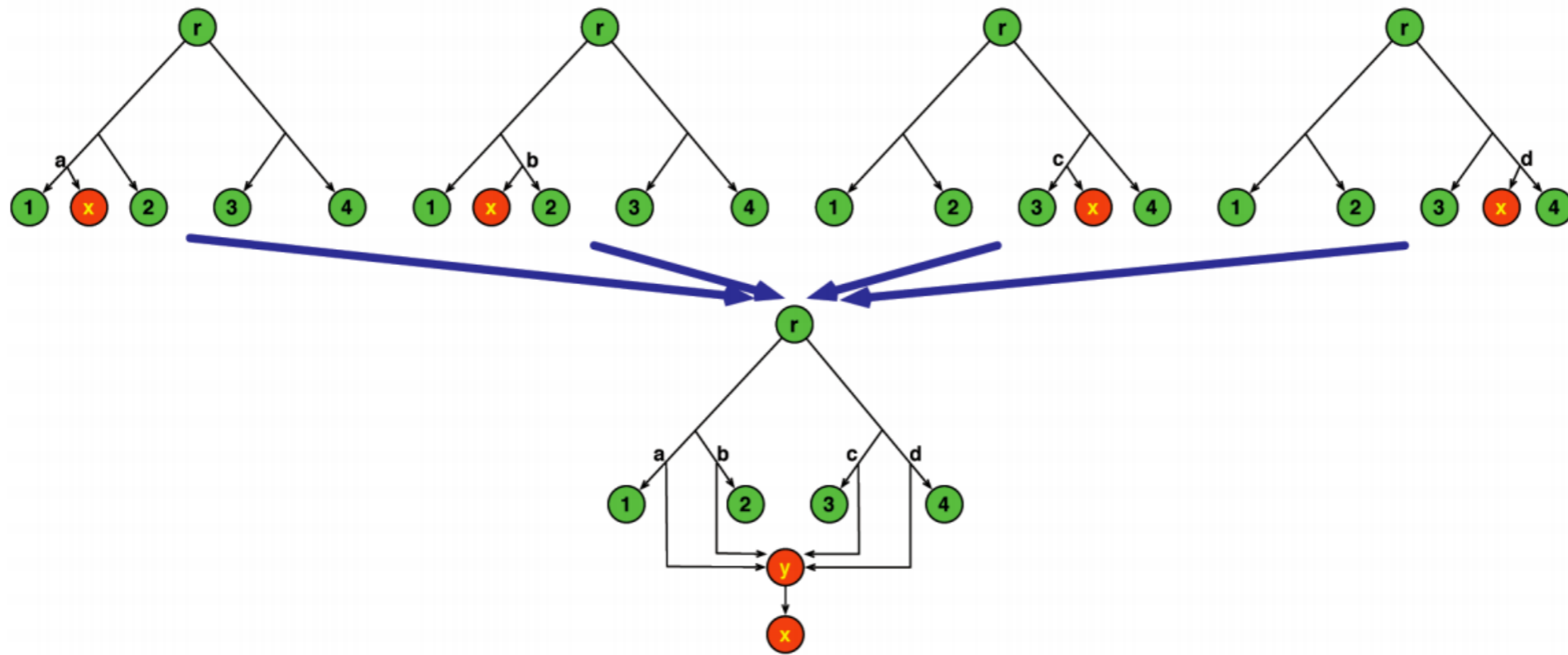
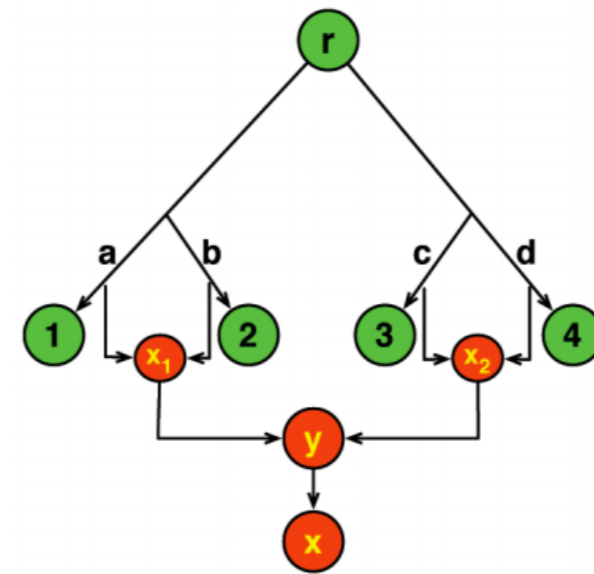


TABLE 1

The Possible Refinements of Node y in the Phylogenetic Network in Fig. 3, Which Result in Networks in Which Each Nodes Has at Most Two Parents

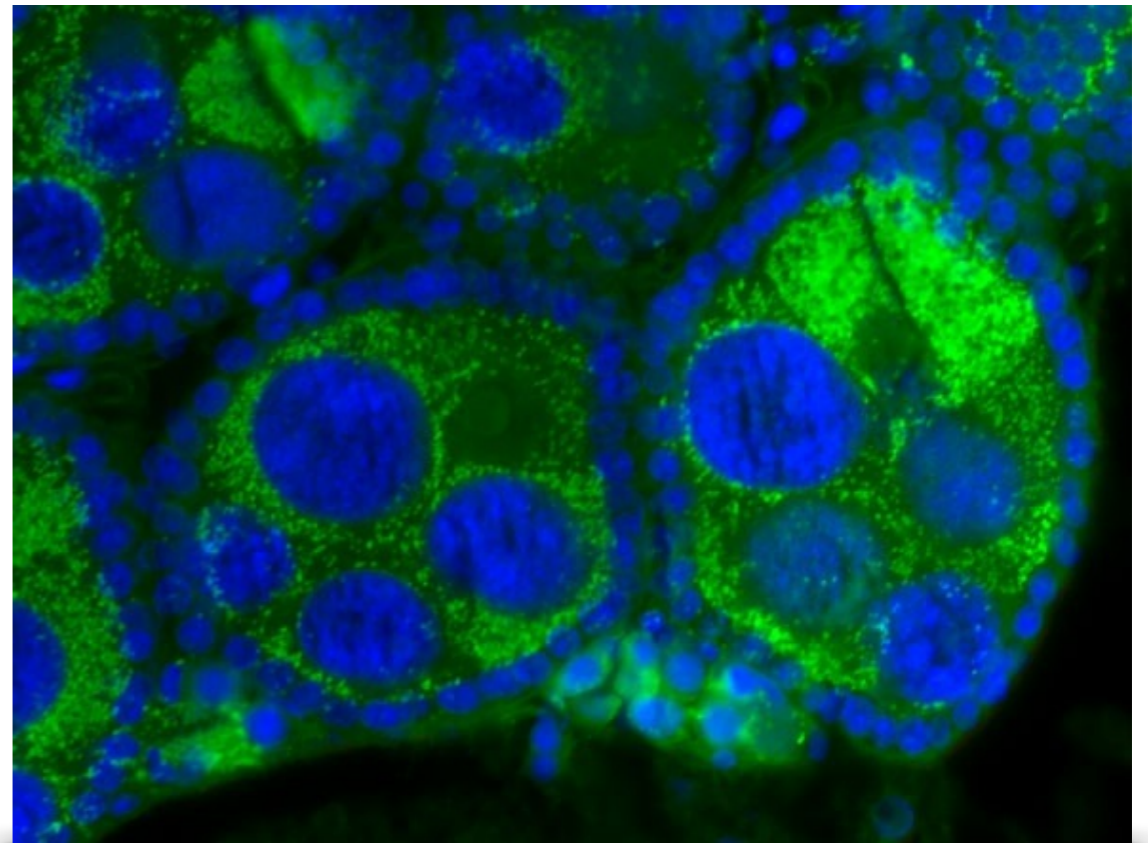
Refinement	Parents of x_1	Parents of x_2	Parents of y
1	a, b	c, d	x_1, x_2
2	a, c	b, d	x_1, x_2
3	a, d	b, c	x_1, x_2
4	a, b	x_1, c	x_2, d
5	a, b	x_1, d	x_2, c
6	a, c	x_1, b	x_2, d
7	a, c	x_1, d	x_2, b
8	a, d	x_1, b	x_2, c
9	a, d	x_1, c	x_2, b



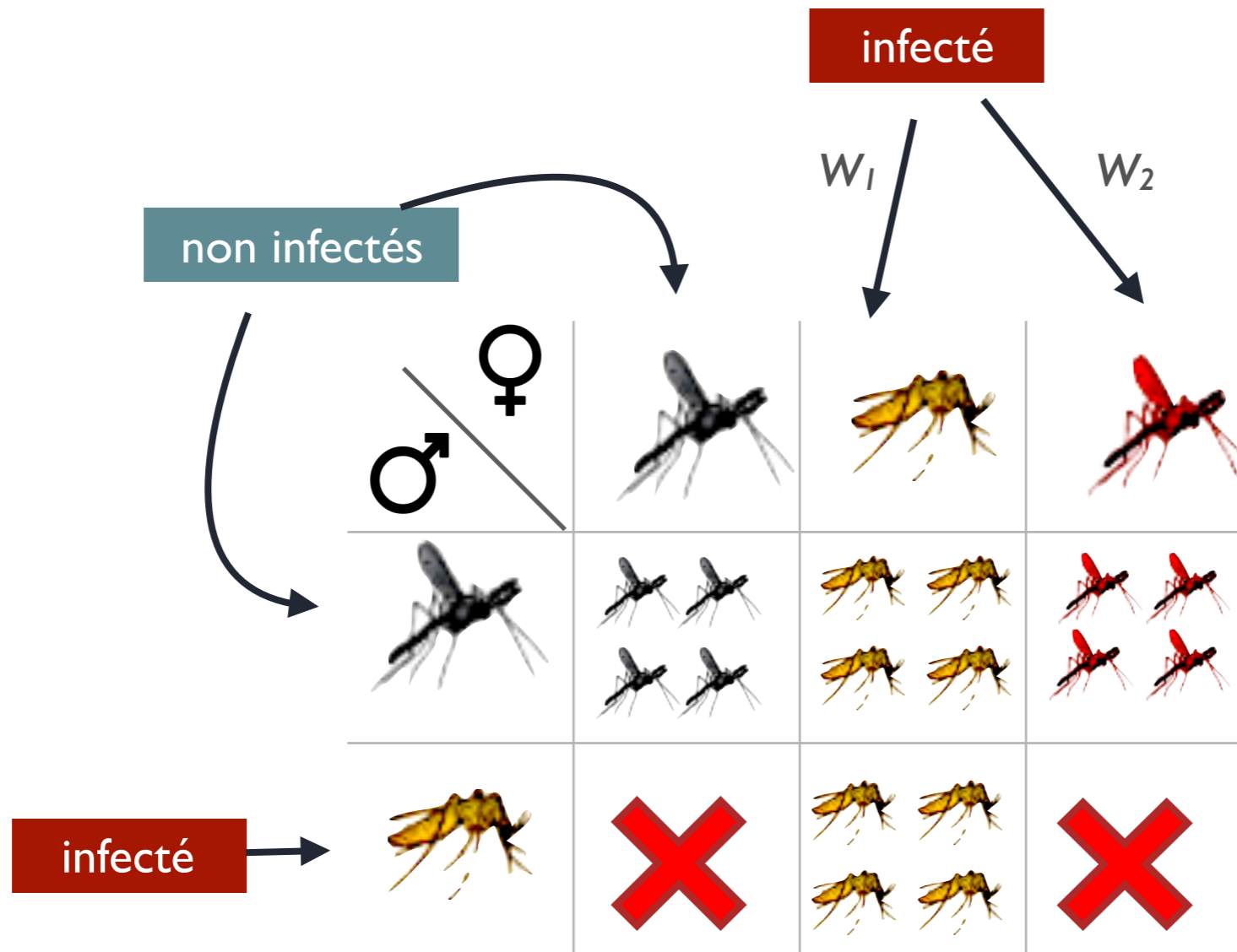
Wolbachia infection and cytoplasmic incompatibility

Wolbachia: a bacteria that manipulates the reproductive system of its host










- It is one of the world's most common parasites infecting around 60% of insects.
- *Cytoplasmic incompatibility*: the inability of *Wolbachia*-infected males to successfully reproduce with uninfected females or females infected with another *Wolbachia* strain.



Compatibility matrix



The problem

	?	?	?
?			
?			
?			

Given a matrix of compatibilities representing the results of crossings, what is the minimum number of different strains of Wolbachia that are necessary to explain the result?

Toxine/Antitoxine model
Idea: For a crossing to be successful the female must carry the antitoxins for all the toxins that the male carries.

Problem definition

Incompatibility matrix

	F0	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10	F11	F12	F13	F14	F15	F16	F17	F18
M0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0
M1	0	0	1	0	0	0	0	1	0	0	1	0	0	0	0	1	0	0	0
M2	1	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0
M3	0	0	1-0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
M4	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
M5	0	0	1	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0
M6	1	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0
M7	1	1	1	1	1	1	1	0	1	0	1	1	1	1	1	1	1	1	1
M8	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
M9	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
M10	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
M11	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0
M12	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0
M13	1	0	1	0	0	1-0	0	1	1	1	1	0	0	0	0	0	0	0	0
M14	1	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0
M15	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0
M16	1	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0
M17	1	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0
M18	1	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0

Find minimum k

Male toxines

	T1	T2	...	T_k
M1				
M2				
...				
M19				

Female antitoxines

	A1	A2	...	A_k
F1				
F2				
...				
F19				

Problem definition

Definition 1. The \otimes vectors multiplication is an operation between two boolean vectors $U, V \in \{0, 1\}^k$ such that:










$$U \otimes V := \begin{cases} 1 & U[i] > V[i] \text{ for some } i \in \{1, \dots, k\} \\ 0 & \text{otherwise} \end{cases}$$

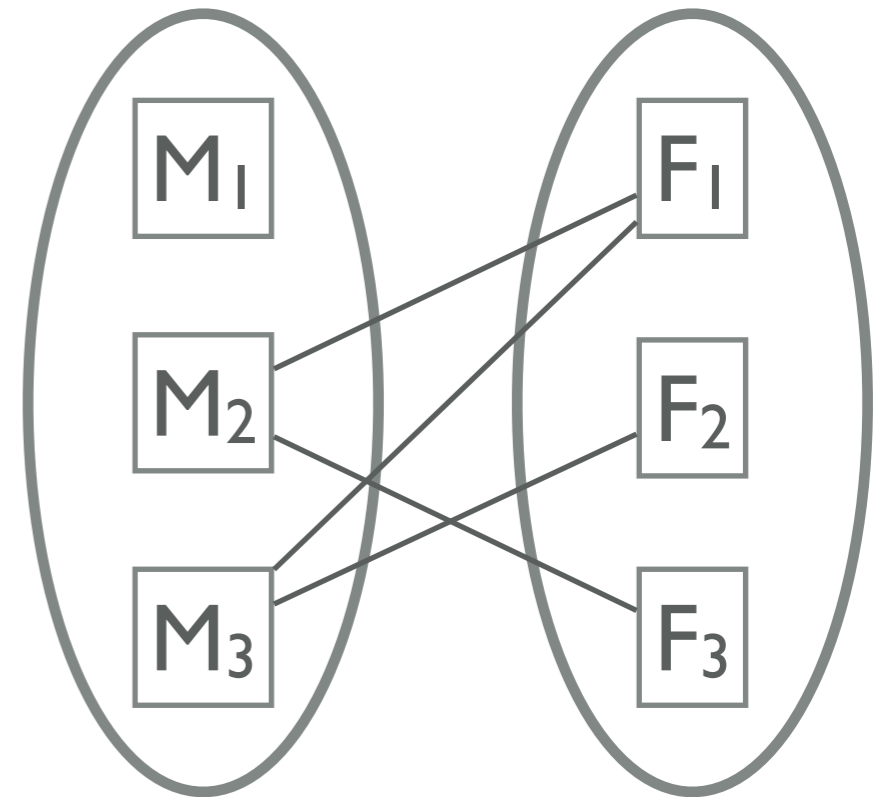
In other words, the result of the \otimes multiplication is 0 if, for all corresponding locations, the value in the second vector is not less than in the first.

Definition 2. The \otimes row-by-row matrix multiplication is a function $\{0, 1\}^{n \times k} \times \{0, 1\}^{m \times k} \rightarrow \{0, 1\}^{n \times m}$ such that $C = M \otimes R$ iff $C_{i,j} = M_i \otimes R_j$ for all $i \in \{1, \dots, n\}$ and $j \in \{1, \dots, m\}$. (Here M_i and R_j respectively denote the i 'th and j 'th rows of M and R .)

Definition 3. In the MOD/RESC PARSIMONY INFERENCE problem, the input is a boolean matrix $C \in \{0, 1\}^{n \times m}$, and the goal is to find two boolean matrices $M \in \{0, 1\}^{n \times k}$ and $R \in \{0, 1\}^{m \times k}$ such that $C = M \otimes R$ and with k minimum.

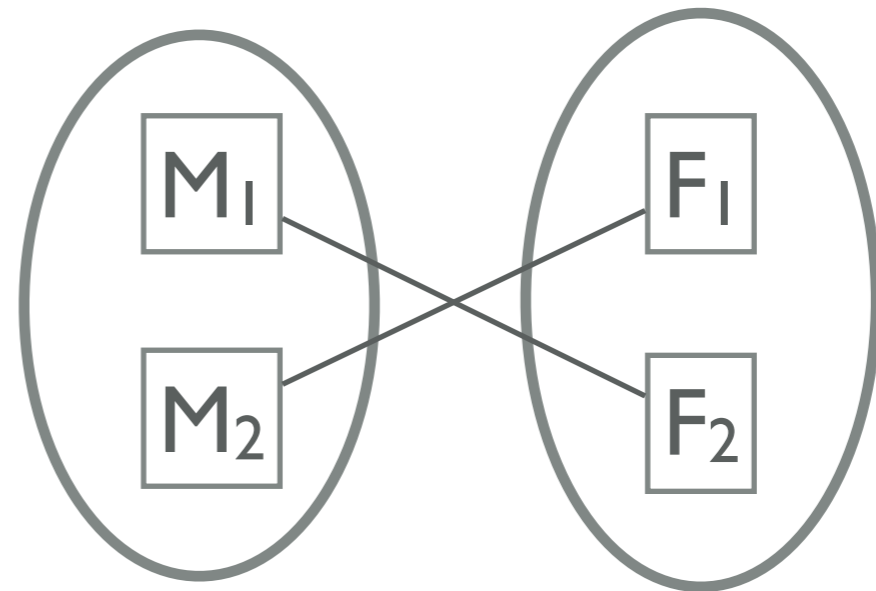
The problem

?	?	?	?
?			
?			
?			



Example

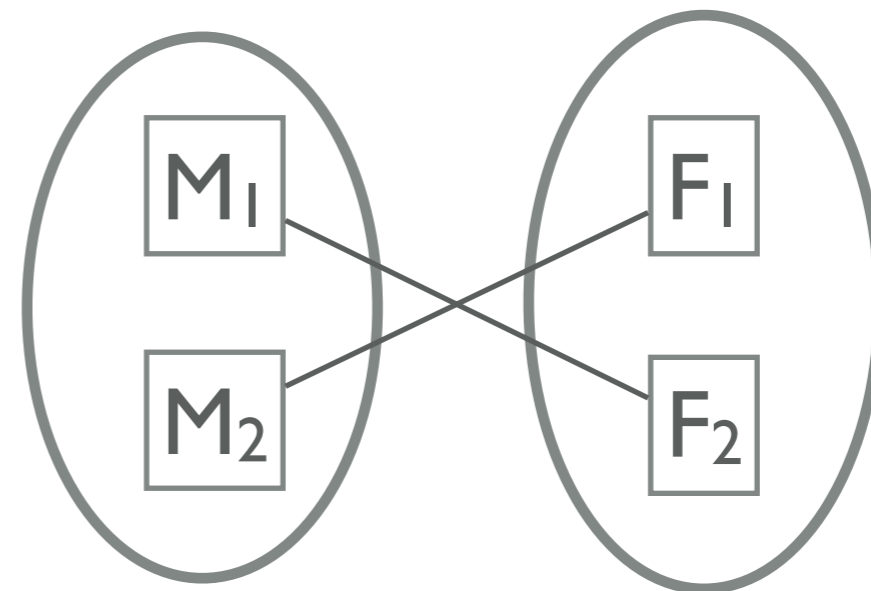
	F1	F2
M1	1	0
M2	0	1



How many pairs toxin/antitoxine we need?

Example

	F1	F2
M1	1	0
M2	0	1

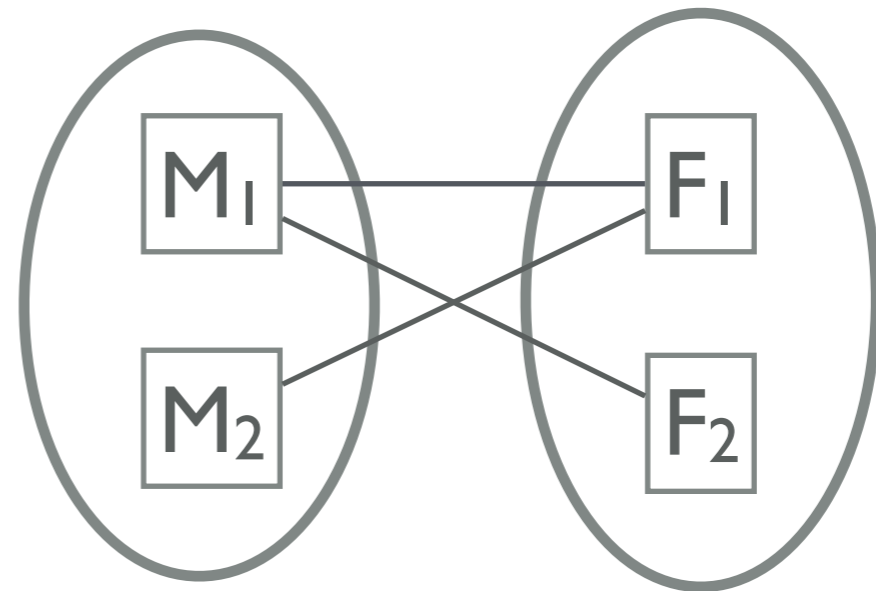


How many pairs toxin/antitoxine we need?

2 different Toxine/Antitoxine

Example

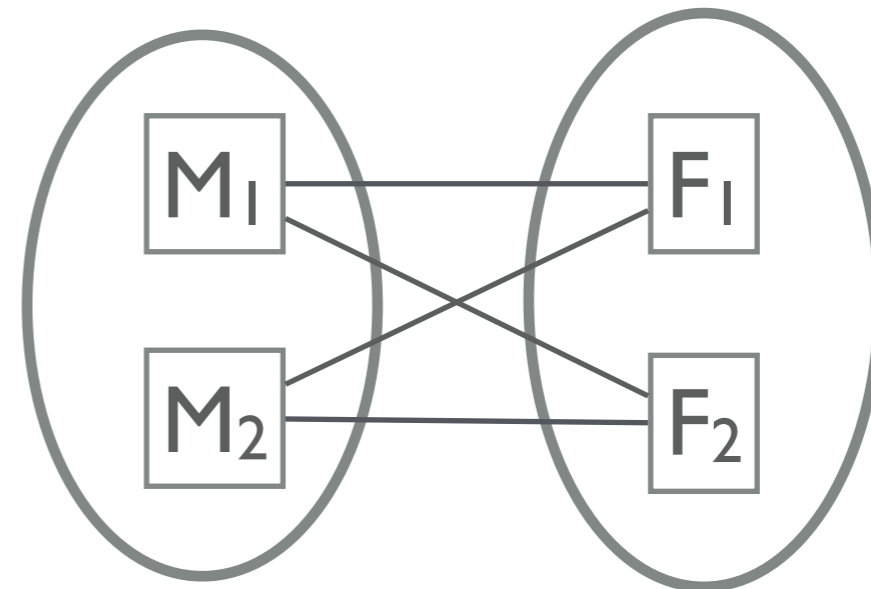
	F1	F2
M1	1	1
M2	0	1



How many pairs toxin/antitoxine we need?

Example

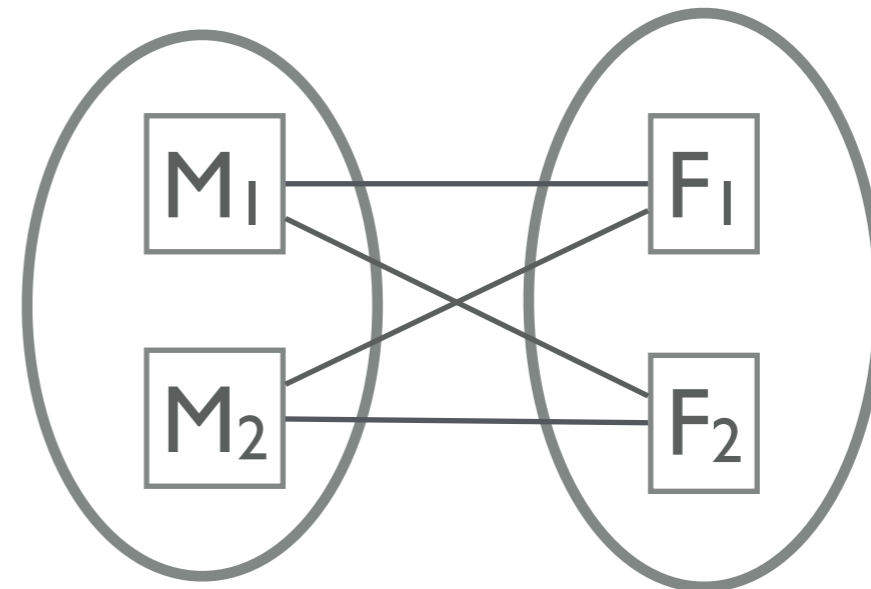
	F1	F2
M1	1	1
M2	1	1



How many pairs toxin/antitoxine we need?

Example

	F1	F2
M1	1	1
M2	1	1



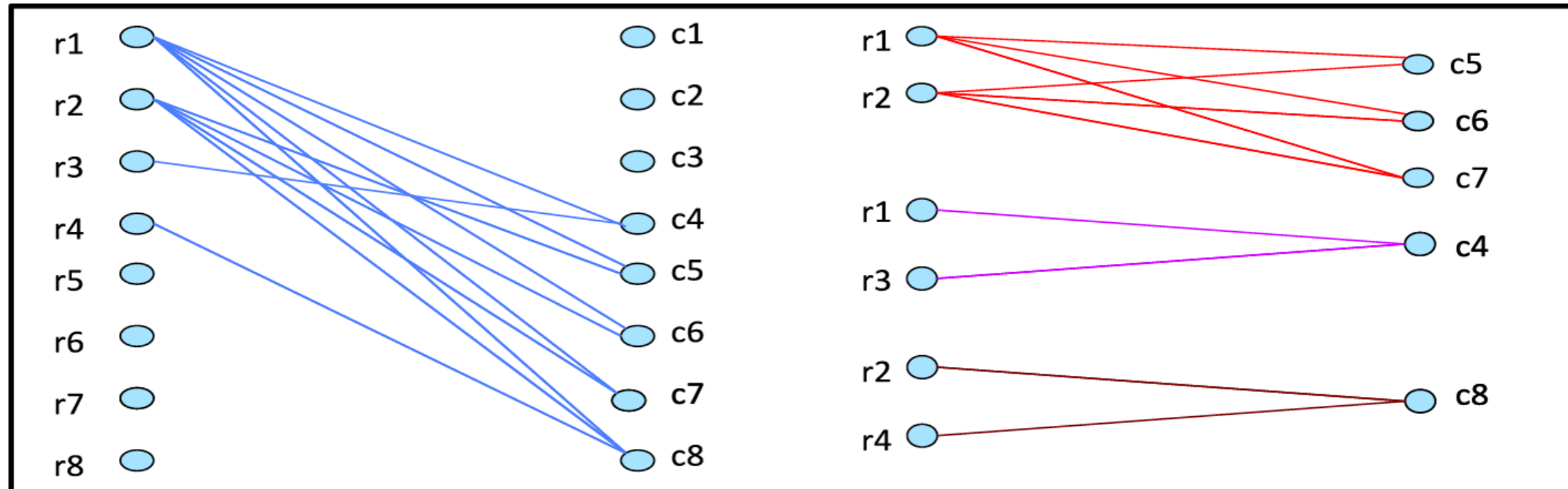
How many pairs toxin/antitoxine we need?

Only 1 pair Toxine/Antitoxine is enough to explain the situation

Definition 4. In the BICLIQUE EDGE COVER problem, the input is a bipartite graph G , and the goal is to find the minimum number of bicliques B_1, \dots, B_k of G such that $E(G) := \bigcup_{\ell} E(B_{\ell})$.

Theorem 1. Let C be a boolean matrix of size $n \times m$. Then there are two matrices $M \in \{0, 1\}^{n \times k}$ and $R \in \{0, 1\}^{m \times k}$ with $C = M \otimes R$ iff the bipartite graph G with $A(G) := C$ has a biclique edge cover with k bicliques.

Reduction



G	1	2	3	4	5	6	7	8
1	0	0	0	1	1	1	1	0
2	0	0	0	0	1	1	1	1
3	0	0	0	1	0	0	0	0
4	0	0	0	0	0	0	0	1
5	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0

M	1	2	3
1	1	1	0
2	1	0	1
3	0	1	0
4	0	0	1
5	0	0	0
6	0	0	0
7	0	0	0
8	0	0	0

R	1	2	3
1	1	1	1
2	1	1	1
3	1	1	1
4	1	0	1
5	0	1	1
6	0	1	1
7	0	1	1
8	1	1	0

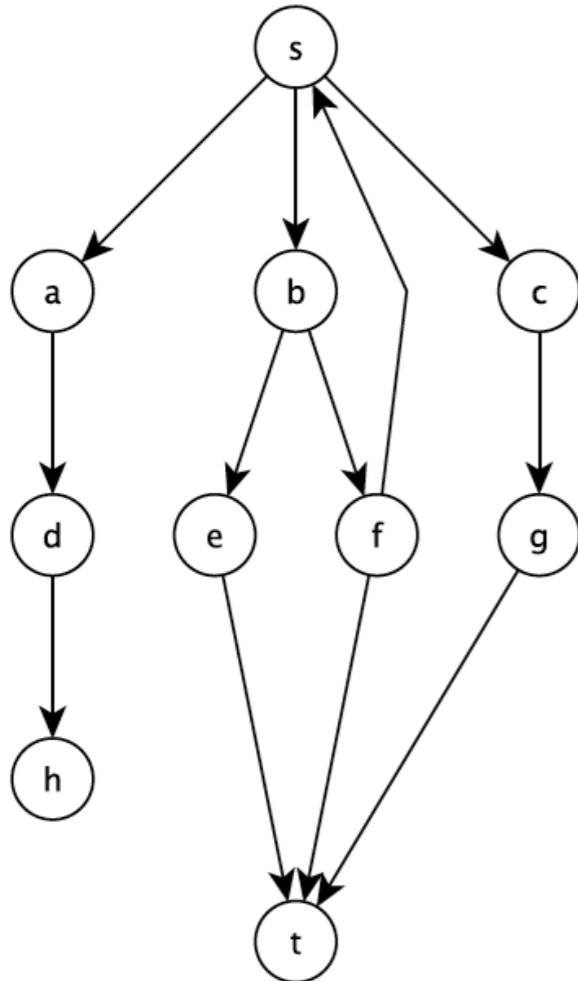
enumerate (listing) all
the solutions

Listing all (s,t)-paths

The problem

Given a directed graph G list all the (s,t)-paths in G .

Idea: Partition the set of solutions



The set of paths $s \rightsquigarrow t$ in G can be partitioned in:

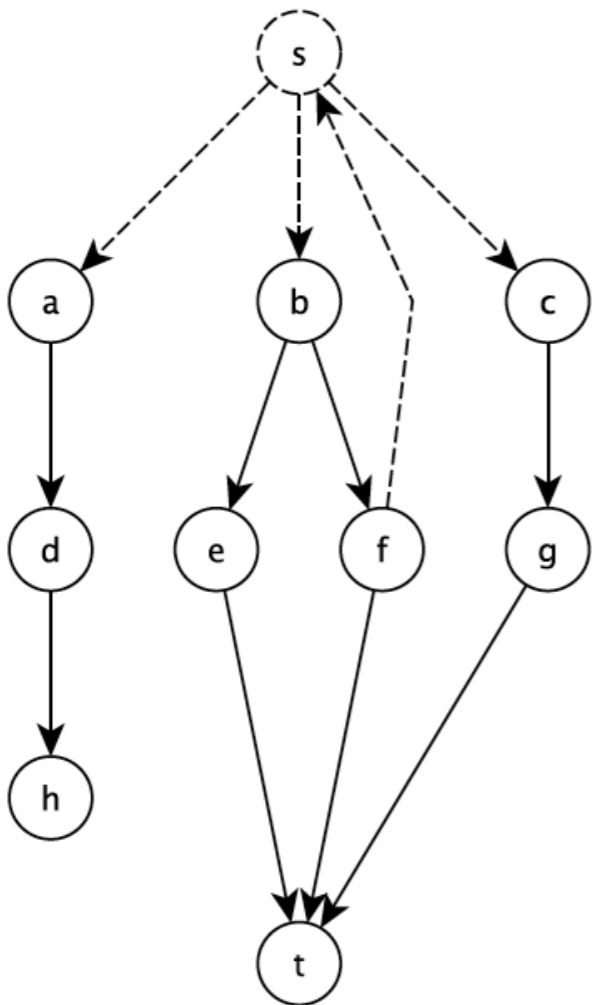
- ▶ paths that use (s, a) ;
- ▶ paths that use (s, b) ;
- ▶ paths that use (s, c) .

Listing all (s,t)-paths

The problem

Given a directed graph G list all the (s,t)-paths in G .

Idea: **Recursively** partition the set of solutions



The set of paths $s \rightsquigarrow t$ in G can be partitioned in:

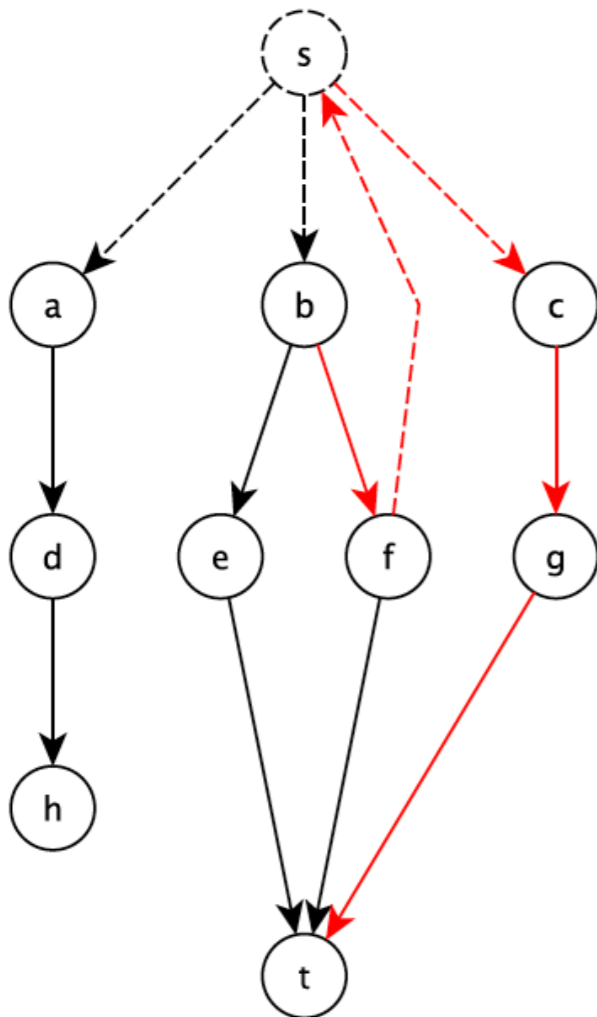
- ▶ (s, a) plus $a \rightsquigarrow t$ in $G - s$;
- ▶ (s, b) plus $b \rightsquigarrow t$ in $G - s$;
- ▶ (s, c) plus $c \rightsquigarrow t$ in $G - s$.

Listing all (s,t)-paths

The problem

Given a directed graph G list all the (s,t)-paths in G .

Idea: **Recursively** partition the set of solutions



The set of paths $s \rightsquigarrow t$ in G can be partitioned in:

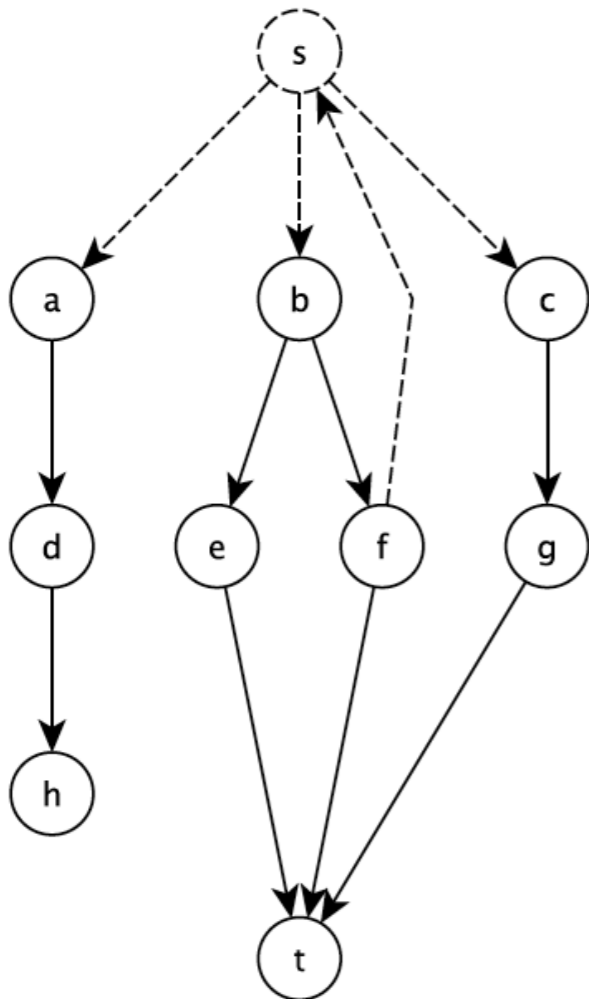
- ▶ (s, a) plus $a \rightsquigarrow t$ in $G - s$;
- ▶ (s, b) plus $b \rightsquigarrow t$ in $G - s$;
- ▶ (s, c) plus $c \rightsquigarrow t$ in $G - s$.

Listing all (s,t)-paths

The problem

Given a directed graph \mathbf{G} list all the (s,t)-paths in \mathbf{G} .

Idea: Explore only non-empty partitions



- ▶ There is no $s \rightsquigarrow t$ path using (s, a) .
- ▶ Before exploring a partition, test if it contains at least one solution.

Listing all (s,t)-paths

The algorithm

Algorithm 1.2: $stPATHS(G, s, t, \pi)$

Input: An undirected graph G , vertices s and t , and a path π (initially empty).

Output: The paths from s to t in G .

```
1 if  $s = t$  then
2   |   output S
3   |   return
4 choose an edge  $e = (s, v)$ 
5 if there is a vt-path in  $G - s$  then
6   |    $stPATHS(G - s, v, t, \pi(s, v))$ 
7 if there is a st-path in  $G - e$  then
8   |    $stPATHS(G - e, s, t, \pi)$ 
```

Listing all (s,t)-paths

The algorithm

Algorithm 1.2: $stPATHS(G, s, t, \pi)$

Input: An undirected graph G , vertices s and t , and a path π (initially empty).

Output: The paths from s to t in G .

1 **if** $s = t$ **then**

2 output S

3 **return**

4 choose an edge $e = (s, v)$

5 **if** *there is a vt-path in $G - s$* **then**

6 $stPATHS(G - s, v, t, \pi(s, v))$

7 **if** *there is a st-path in $G - e$* **then**

8 $stPATHS(G - e, s, t, \pi)$



$O(|V| + |E|)$ using DFS

Listing all (s,t)-paths

The algorithm

Algorithm 1.2: $stPATHS(G, s, t, \pi)$

Input: An undirected graph G , vertices s and t , and a path π (initially empty).

Output: The paths from s to t in G .

1 **if** $s = t$ **then**

2 output S

3 **return**

4 choose an edge $e = (s, v)$

5 **if** *there is a vt-path in $G - s$* **then**

6 $stPATHS(G - s, v, t, \pi(s, v))$

7 **if** *there is a st-path in $G - e$* **then**

8 $stPATHS(G - e, s, t, \pi)$



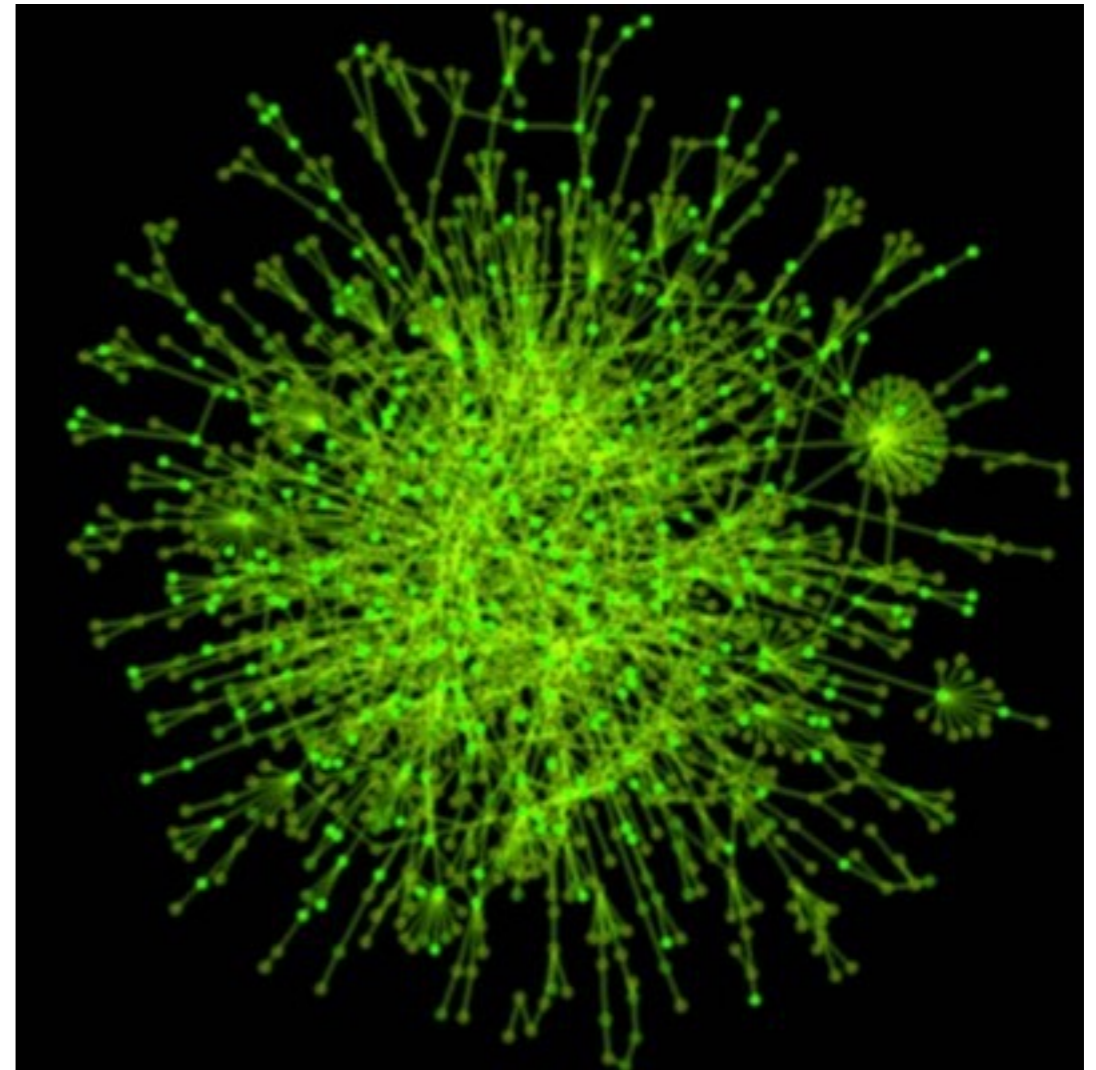
$O(|V| + |E|)$ using DFS

Delay: $O((|V| + |E|)^2)$.

from local to global

Local view of large structures

- Graphs in input can be large
- Number of solutions can be large

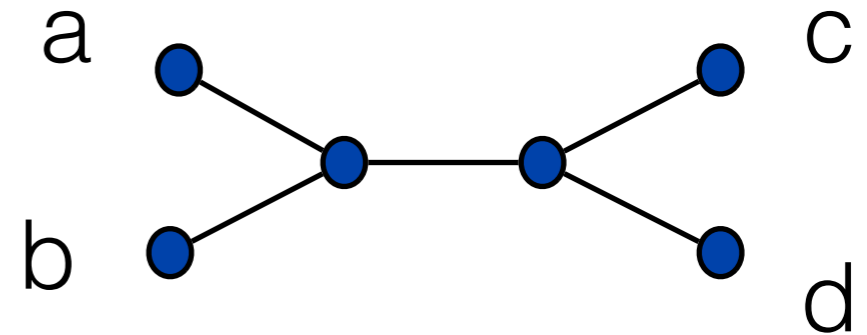


Approach (Local view): Sample substructures of size k .

From a set of quartets to phylogenetic trees

- The base unit of information of a unrooted phylogenetic tree is a quartet.

Quartet based reconstruction:



- Given a set S of quartets find the tree on the full set of species that satisfies most of the quartets
- Even deciding whether there is a tree T that satisfies all the quartets is NP-complete [Steel '92]
- Approximation algorithm: Random labeling gives $1/3$ expected approximation ratio.
- Possible direction: Check a “small” subset of quartets if it is compatible. How much can we infer about a quartet set just by examining its constituting subsets? [Alon et al. SODA '14]

