

# PhyloNet

Yun Yu

Department of Computer Science  
Bioinformatics Group  
Rice University  
yy9@rice.edu

# Basic Usage

---

- ❖ Command:

> java -jar PhyloNet\_3.4.0.jar script.nex

- ❖ Input NEXUS file:

```
#NEXUS

BEGIN TREES;

Tree g1 = (((a,b),c),d),e);
Tree g2 = ((a,b),((c,e),d));
Tree g3 = ((a,c),((b,e),d));

END;

BEGIN PHYLONET;

Infer_ST_MDC (g1, g2, g3);

END;
```

# Commands

---

## ❖ Likelihood

- ❖ For a phylogenetic phylogeny (tree or network)
  - ❖ Scoring: *CalGTProb*
  - ❖ Inference: *InferNetwork\_ML*

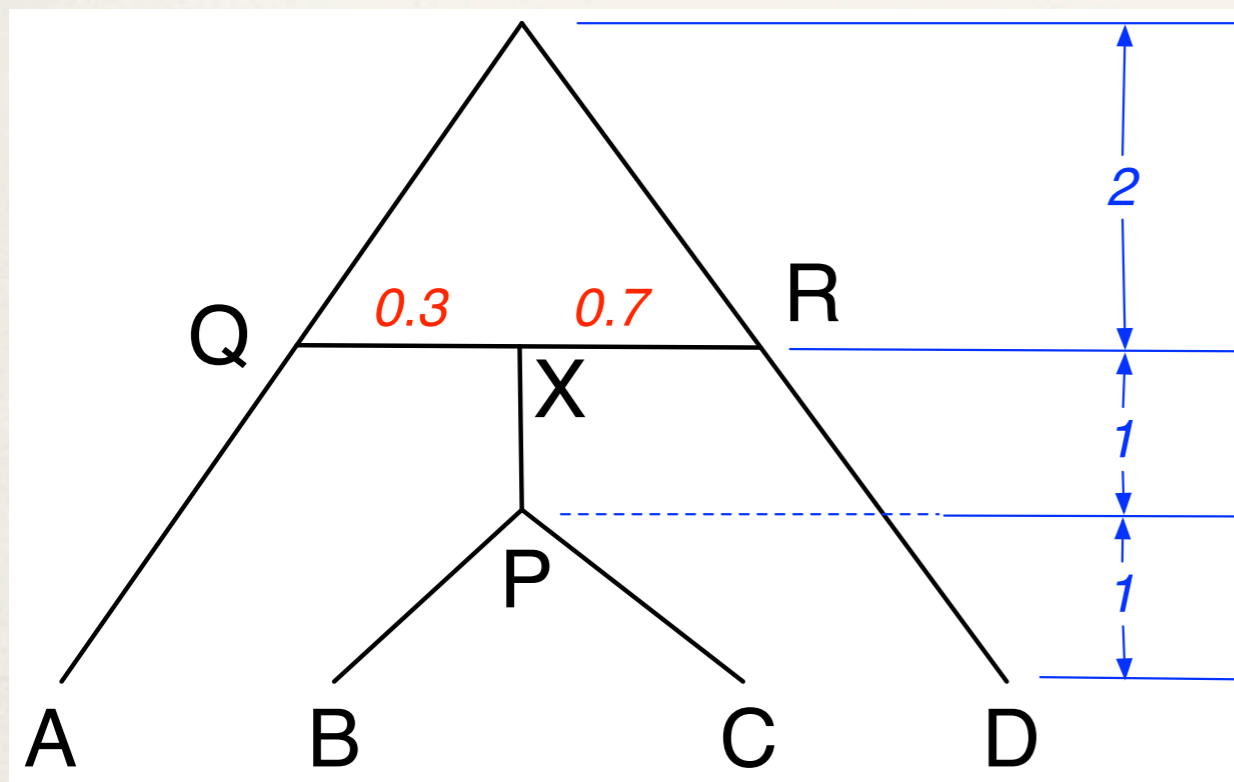
## ❖ Parsimony

- ❖ For a phylogenetic tree (ILS)
  - ★ Scoring: *DeepCoalCount\_Tree*
  - ★ Inference: *Infer\_ST\_MDC, Infer\_ST\_MDC\_UR*
- ❖ For a phylogenetic network (ILS & hybridization)
  - ★ Scoring: *DeepCoalCount\_Network*
  - ★ Inference: *InferNetwork\_Parsimony*

# Phylogenetic Network

## ❖ Rich Newick Format

- ❖ Newick for phylogenetic tree & Extended Newick for phylogenetic network (Cardona et al., 2008)



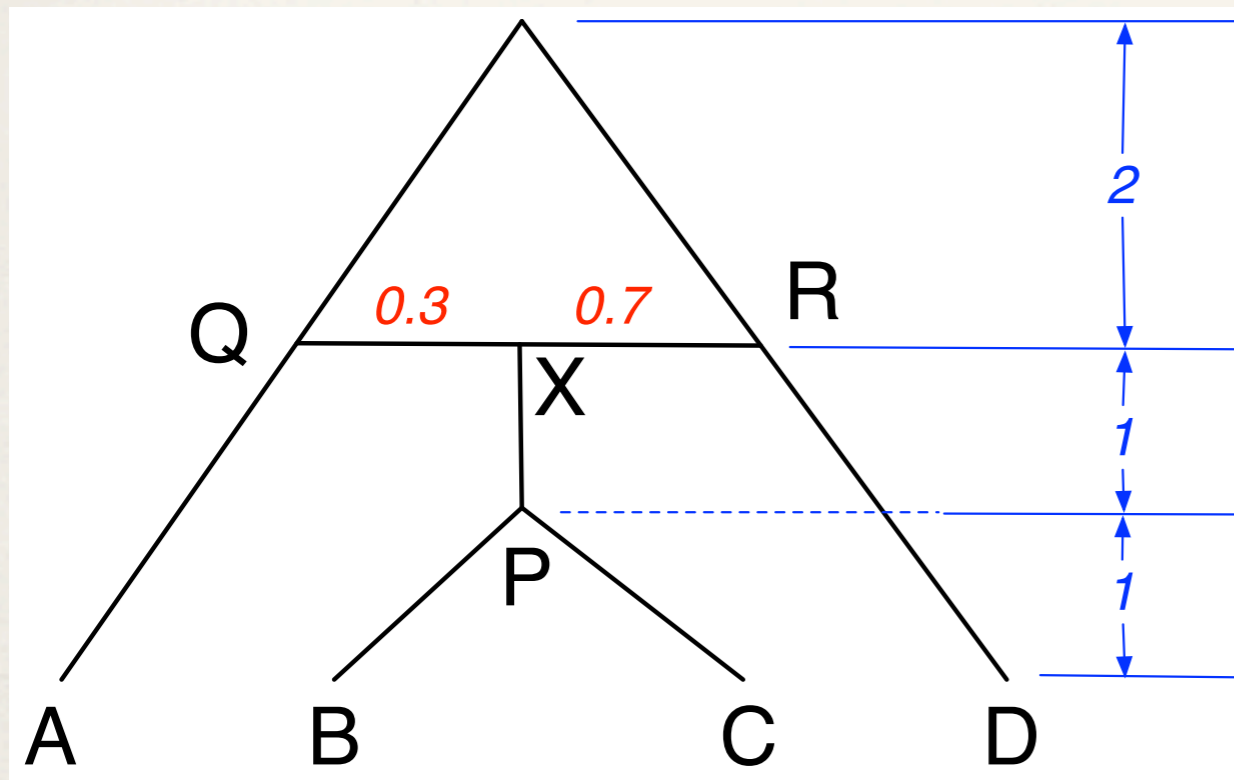
```
((A:2,((B:1,C:1)P:1)X#H1:0::0.3)Q:2,  
(D:2,X#H1:0::0.7)R:2);
```

:branch length:support:inheritance probability

# Phylogenetic Network

## ❖ Rich Newick Format

- ❖ Newick for phylogenetic tree & Extended Newick for phylogenetic network (Cardona et al., 2008)



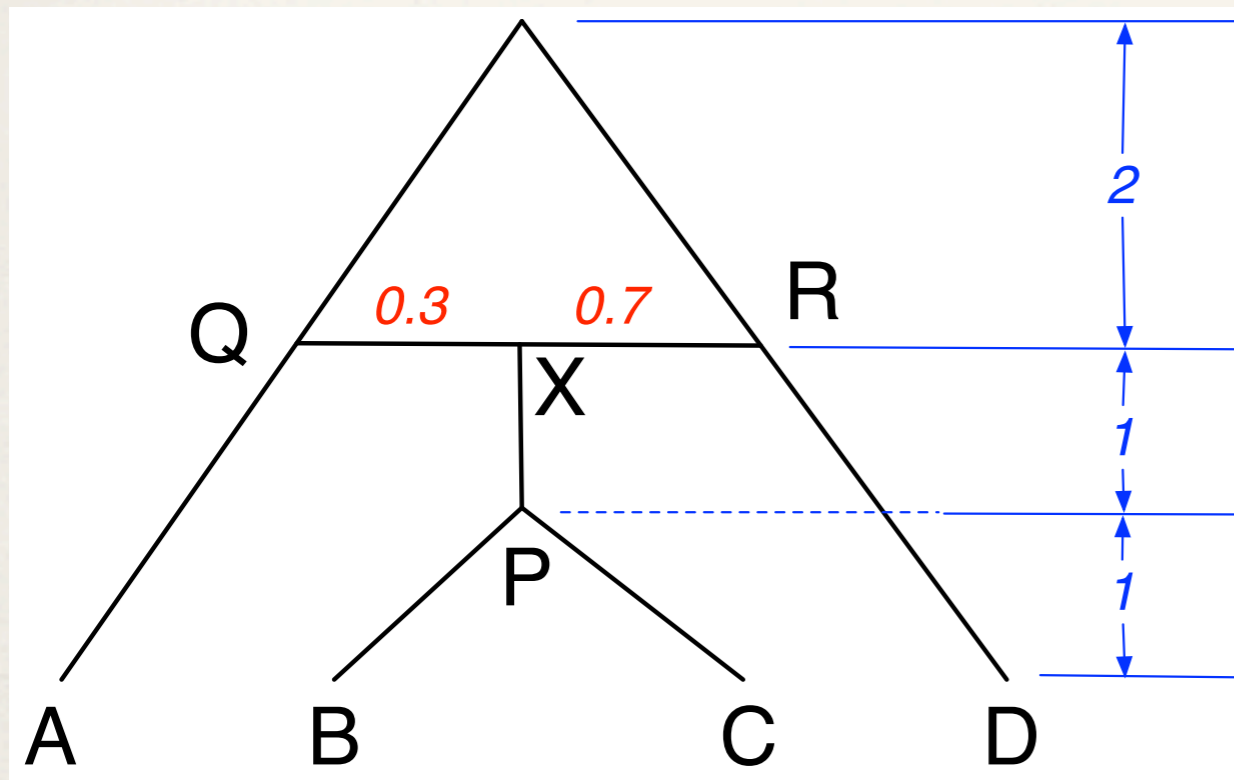
```
((A:2,((B:1,C:1)P:1)X#H1:0::0.3)Q:2,  
(D:2,X#H1:0::0.7)R:2);
```

:branch length:support:inheritance probability

# Phylogenetic Network

## ❖ Rich Newick Format

- ❖ Newick for phylogenetic tree & Extended Newick for phylogenetic network (Cardona et al., 2008)



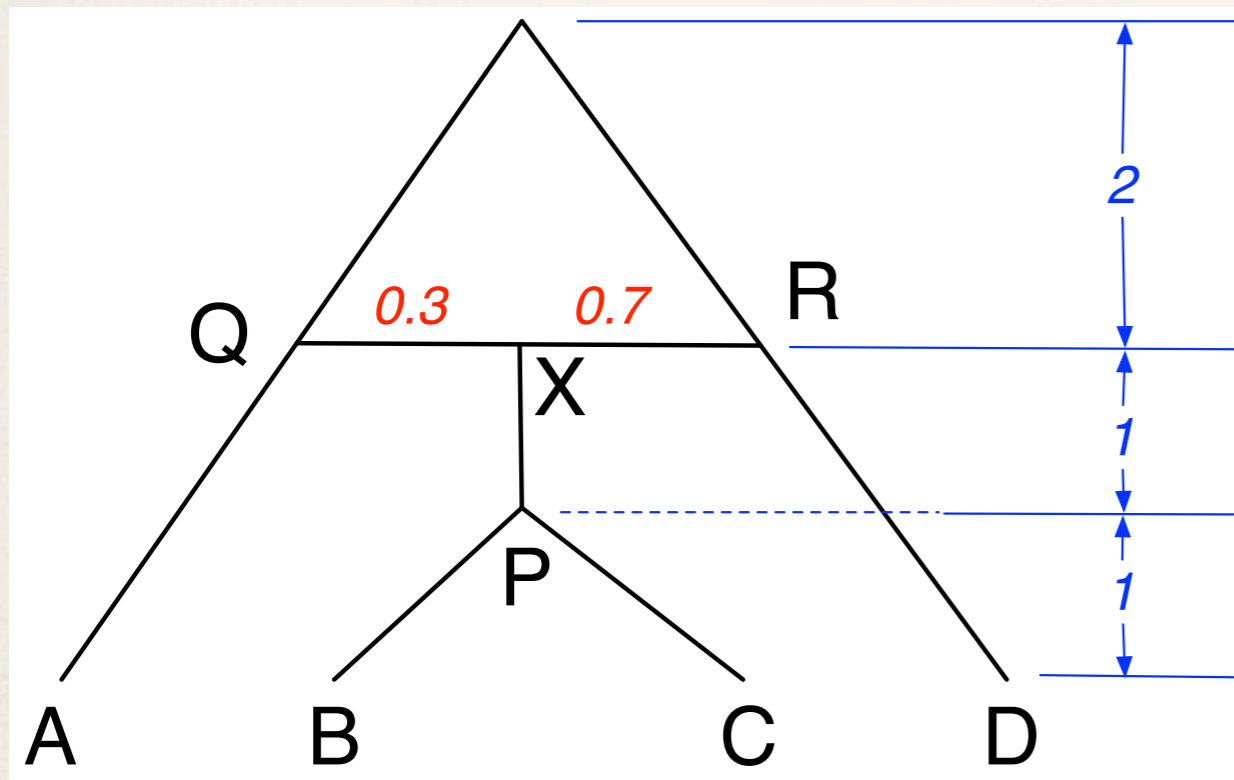
```
((A:2,((B:1,C:1)P:1)X#H1:0::0.3)Q:2,  
(D:2,X#H1:0::0.7)R:2);
```

:branch length:support:inheritance probability

# Phylogenetic Network

## ❖ Rich Newick Format

- ❖ Newick for phylogenetic tree & Extended Newick for phylogenetic network (Cardona et al., 2008)



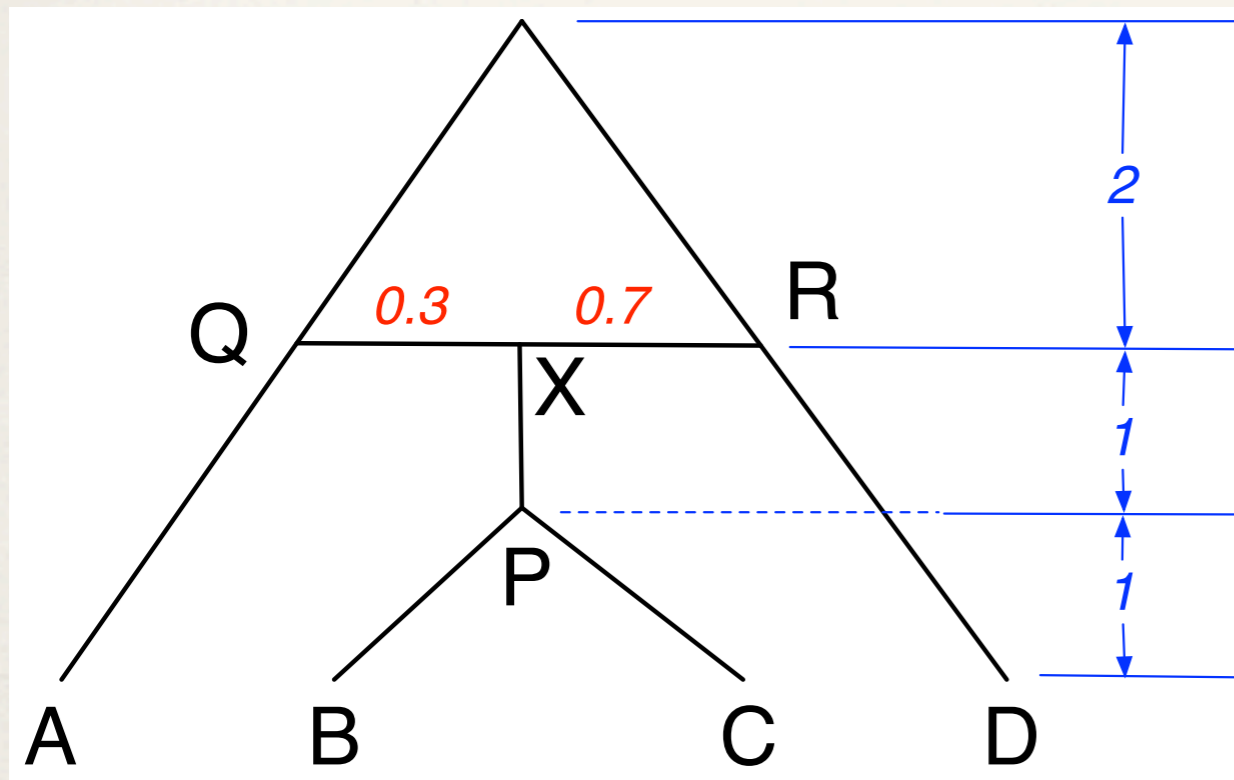
```
((A:2,((B:1,C:1)P:1)X#H1:0::0.3)Q)2,  
(D:2,X#H1:0::0.7)R:2);
```

:branch length:support:inheritance probability

# Phylogenetic Network

## ❖ Rich Newick Format

- ❖ Newick for phylogenetic tree & Extended Newick for phylogenetic network (Cardona et al., 2008)



```
((A:2,((B:1,C:1)P:1)X#H1:0::0.3)Q:2,  
(D:2,X#H1:0::0.7)R:2);
```

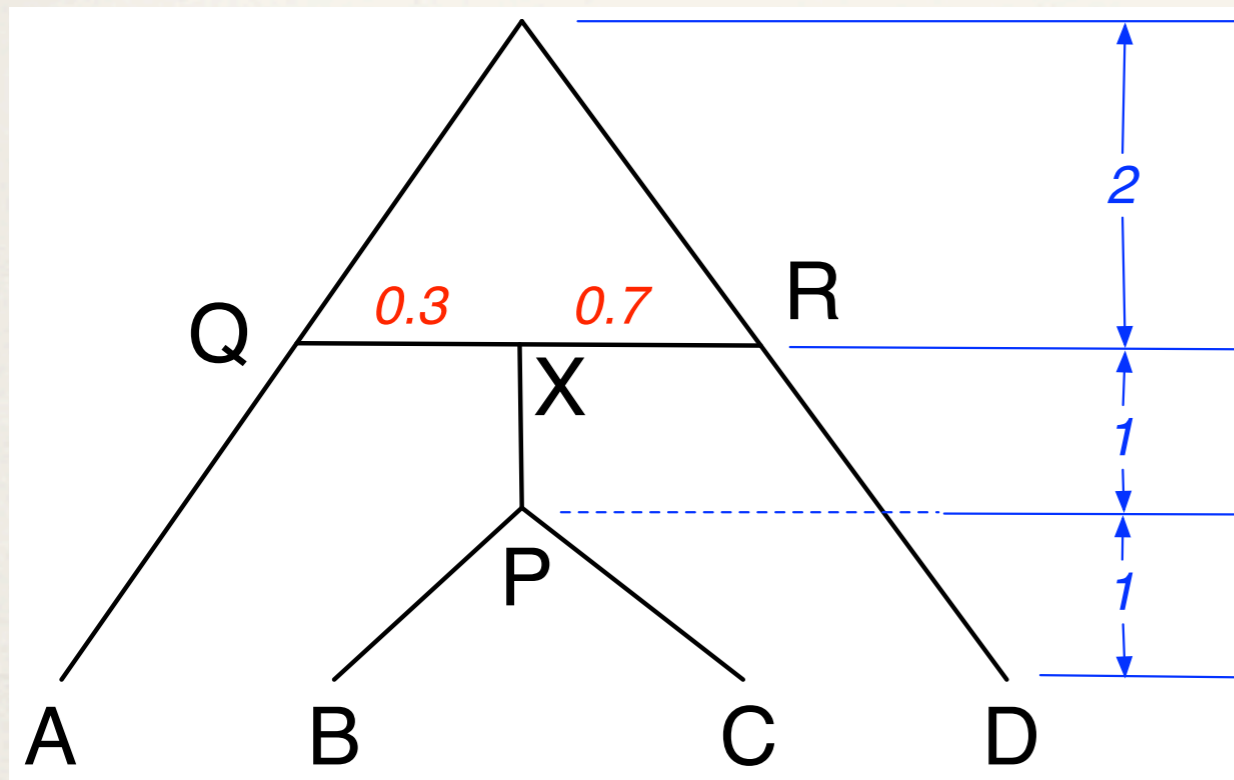
:branch length:support:inheritance probability



# Phylogenetic Network

## ❖ Rich Newick Format

- ❖ Newick for phylogenetic tree & Extended Newick for phylogenetic network (Cardona et al., 2008)



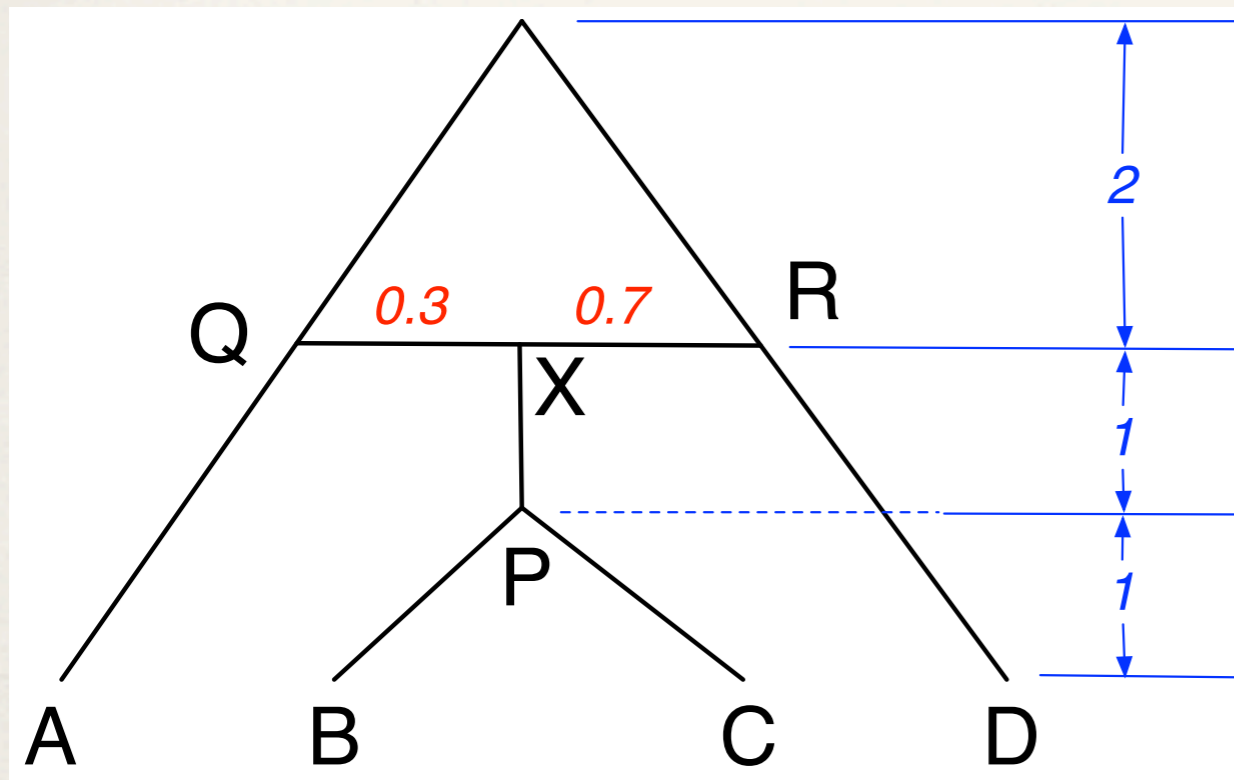
```
((A:2,((B:1,C:1)P:1)X#H1:0::0.3)Q:2,  
(D:2,X#H1:0::0.7)R:2);
```

:branch length:support:inheritance probability

# Phylogenetic Network

## ❖ Rich Newick Format

- ❖ Newick for phylogenetic tree & Extended Newick for phylogenetic network (Cardona et al., 2008)



```
((A:2,((B:1,C:1)P:1)X#H1:0::0.3)Q:2,  
(D:2,X#H1:0::0.7)R:2);
```

:branch length:support:inheritance probability

# Phylogenetic Network

---

- ❖ Rich Newick Format

- ♦ Newick for phylogenetic tree & Extended Newick for phylogenetic network (Cardona et al., 2008)
- ♦ <https://wiki.rice.edu/confluence/display/PHYLONET/Rich+Newick+Format>

# Visualizing a Phylogenetic Network

---

- ❖ Dendroscope: <http://ab.inf.uni-tuebingen.de/software/dendroscope/>
- ❖ Removing the inheritance probabilities from the Rich Newick string PhyloNet output

```
(((((Skud, (Sbay)#H1:::0.36082474226804123),  
(Spar, Scer), Smik)), #H1:::0.6391752577319587), Scas), Sklu), Calb);
```



```
(((((Skud, (Sbay)#H1), (Spar, Scer), Smik)), #H1), Scas), Sklu), Calb);
```

ILS

---

# Scoring a Phylogenetic Tree Under Parsimony

---

## ❖ Command Usage

```
DeepCoalCount_tree {speciesTree1 [, speciesTree2...]}  
(geneTree1 [, geneTree2...]) [-u] [-b threshold] [-a taxaMap]  
[resultOutputFile]
```

<i>speciesTree1</i> [, <i>speciesTree2</i> ...]	Comma delimited set of species trees	mandatory
<i>geneTree1</i> [, <i>geneTree2</i> ...]	Comma delimited collection of gene trees.	mandatory
-u	Gene trees should be treated as unrooted.	optional
-b <i>threshold</i>	Gene trees bootstrap threshold.	optional
-a <i>taxaMap</i>	Gene trees / species tree taxa association.	optional
<i>resultOutputFile</i>	Optional file destination for command output.	optional

# Scoring a Phylogenetic Tree Under Parsimony

---

## ❖ Input NEXUS File

```
#NEXUS

BEGIN TREES;
Tree st1 = ((((((Scer,Spar),Smik),Skud),Sbay),Scas),Sklu),Calb);
Tree st2 = ((((((Scer,Spar),Smik),(Skud,Sbay)),Scas),Sklu),Calb);
Tree gt0 = ((((((Scer,Spar),Smik),Skud),Sbay),Scas),Sklu),Calb);
...
Tree gt105 = ((((((Scer,Spar),Smik),Skud),Sbay),(Scas,Sklu)),Calb);
END;

BEGIN PHYLONET;
DeepCoalCount_tree {st1,st2} (gt0,...,gt105);
END;
```

## ❖ Output

```
Species_Tree#1 = ((((((Scer:0,Spar:0):2,Smik:0):21,Skud:0):54,Sbay:0):0,Scas:0):35,Sklu:0):0,Calb:0):0;
Total number of extra lineages: 112
Species_Tree#2 = ((((((Scer:0,Spar:0):2,Smik:0):21,(Skud:0,Sbay:0):64):0,Scas:0):35,Sklu:0):0,Calb:0):0;
Total number of extra lineages: 122
```

# Inferring a Phylogenetic Tree Under Parsimony

---

## ❖ Command Usage

```
infer_ST_MDC/infer_ST_MDC_UR (geneTree1 [, geneTree2...]) [-e proportion] [-x] [-b threshold] [-a taxaMap] [-ur] [-t time] [resultOutputFile]
```

<i>geneTree1 [, geneTree2...]</i>	Comma delimited collection of gene trees.	mandatory
<i>-e proportion</i>	Get optimal and sub-optimal species trees whose scores are less than <i>proportion%</i> worse than the optimal score.	optional
<i>-x</i>	Use all clusters instead of clusters in gene trees.	optional
<i>-b threshold</i>	Gene trees bootstrap threshold.	optional
<i>-a taxaMap</i>	Gene trees / species tree taxa association.	optional
<i>-ur</i>	Allow returning non-binary species tree.	optional
<i>-t timeLimit</i>	Limit search time to <i>timeLimit</i> minutes.	optional
<i>resultOutputFile</i>	Optional file destination for command output.	optional



# Inferring a Phylogenetic Tree Under Parsimony

---

## ❖ Input NEXUS File

```
#NEXUS

BEGIN TREES;
Tree gt0 = ((((((Scer,Spar),Smik),Skud),Sbay),Scas),Sklu),Calb);
...
Tree gt105 = ((((((Scer,Spar),Smik),Skud),Sbay),(Scas,Sklu)),Calb);
END;

BEGIN PHYLONET;
Infer_ST_MDC (gt0,...,gt105);
END;
```

## ❖ Output

```
(Calb:0,(Sklu:0,(Scas:0,(Sbay:0,(Skud:0,(Smik:0,(Scer:0,Spar:0):2):21):54):0):35):0):0;
Total number of extra lineages:112
```

# ILS + Hybridization

---

---

# Scoring a Phylogenetic Network Under Parsimony

---

## ❖ Command Usage

```
DeepCoalCount_network speciesNetwork (geneTree1 [,  
geneTree2...]) [-m ac|mul] [-b threshold] [-a taxaMap]  
[resultOutputFile]
```

<i>speciesNetwork</i>	The candidate species network	mandatory
<i>geneTree1</i> [, <i>geneTree2</i> ...]	Comma delimited collection of gene trees.	mandatory
<i>-m ac mul</i>	Specify the algorithm used for computation. The default value is <i>ac</i> .	optional
<i>-b threshold</i>	Gene trees bootstrap threshold.	optional
<i>-a taxaMap</i>	Gene trees / species tree taxa association.	optional
<i>resultOutputFile</i>	Optional file destination for command output.	optional

# Scoring a Phylogenetic Network Under Parsimony

---

## ❖ Input NEXUS File

```
#NEXUS

BEGIN NETWORKS;
Network sn = (((X#H1,((Smik,(Scer,Spar)),((Sbay)X#H1,Skud))),Scas),Sklu),Calb);
END;

BEGIN TREES;
Tree gt0 = ((((((Scer,Spar),Smik),Skud),Sbay),Scas),Sklu),Calb);
...
Tree gt105 = ((((((Scer,Spar),Smik),Skud),Sbay),(Scas,Sklu)),Calb);
END;

BEGIN PHYLONET;
DeepCoalCount_network sn (gt0,...,gt105);
END;
```

## ❖ Output

```
[x41] ((((((Scer,Spar),Smik),Skud),Sbay),Scas),Sklu),Calb);: 0
[x2] (((((Scer,Spar),Smik,(Skud,Sbay)),Scas),Sklu),Calb);: 0
[x2] ((((((Scer,Spar),(Skud,Sbay)),Smik),Sklu),Scas),Calb);: 2
...
Total number of extra lineages: 60
```

# Inferring a Phylogenetic Network Under Parsimony

---

## ❖ Command Usage

```
inferNetwork_parsimony (geneTree1 [, geneTree2...]) numReticulations  
[-b threshold] [-a taxaMap] [-s startingNetwork] [-n numNetReturned]  
[-m maxNetExamined] [-d maxDiameter] [resultOutputFile]
```

<i>geneTree1 [, geneTree2...]</i>	Comma delimited collection of gene trees.	mandatory
<i>numReticulations</i>	Maximum number of reticulation nodes to added.	mandatory
<i>-b threshold</i>	Gene trees bootstrap threshold.	optional
<i>-a taxaMap</i>	Gene trees / species tree taxa association.	optional
<i>-s startingNetwork</i>	Specify the network to start search. Default value is the optimal MDC tree.	optional
<i>-n numNetReturned</i>	Number of optimal networks to return. Default value is 1.	optional
<i>-m maxNetExamined</i>	Maximum number of network topologies to examined. Default value is infinity.	optional
<i>-d maxDiameter</i>	Maximum diameter to make an arrangement during network search. Default value is infinity.	optional
<i>resultOutputFile</i>	Optional file destination for command output.	optional

# Inferring a Phylogenetic Network Under Parsimony

## ❖ Command Usage

```
inferNetwork_parsimony (geneTree1 [, geneTree2...]) numReticulations  
[-b threshold] [-a taxaMap] [-s startingNetwork] [-n numNetReturned]  
[-m maxNetExamined] [-d maxDiameter] [resultOutputFile]
```

<i>geneTree1</i> [, <i>geneTree2</i> ...]	Comma delimited collection of gene trees.	mandatory
<i>numReticulations</i>	Maximum number of reticulation nodes to added.	mandatory
<i>-b threshold</i>	Gene trees bootstrap threshold.	optional
<i>-a taxaMap</i>	Gene trees / species tree taxa association.	optional
<i>-s startingNetwork</i>	Specify the network to start search. Default value is the optimal MDC tree.	optional
<i>-n numNetReturned</i>	Number of optimal networks to return. Default value is 1.	optional
<i>-m maxNetExamined</i>	Maximum number of network topologies to examined. Default value is infinity.	optional
<i>-d maxDiameter</i>	Maximum diameter to make an arrangement during network search. Default value is infinity.	optional
<i>resultOutputFile</i>	Optional file destination for command output.	optional

# Inferring a Phylogenetic Network Under Parsimony

## ❖ Command Usage

```
inferNetwork_parsimony (geneTree1 [, geneTree2...]) numReticulations  
[-b threshold] [-a taxaMap] [-s startingNetwork] [-n numNetReturned]  
[-m maxNetExamined] [-d maxDiameter] [resultOutputFile]
```

<i>geneTree1</i> [, <i>geneTree2</i> ...]	Comma delimited collection of gene trees.	mandatory
<i>numReticulations</i>	Maximum number of reticulation nodes to added.	mandatory
<i>-b threshold</i>	Gene trees bootstrap threshold.	optional
<i>-a taxaMap</i>	Gene trees / species tree taxa association.	optional
<i>-s startingNetwork</i>	Specify the network to start search. Default value is the optimal MDC tree.	optional
<i>-n numNetReturned</i>	Number of optimal networks to return. Default value is 1.	optional
<i>-m maxNetExamined</i>	Maximum number of network topologies to examined. Default value is infinity.	optional
<i>-d maxDiameter</i>	Maximum diameter to make an arrangement during network search. Default value is infinity.	optional
<i>resultOutputFile</i>	Optional file destination for command output.	optional

# Inferring a Phylogenetic Network Under Parsimony

## ❖ Command Usage

```
inferNetwork_parsimony (geneTree1 [, geneTree2...]) numReticulations  
[-b threshold] [-a taxaMap] [-s startingNetwork] [-n numNetReturned]  
[-m maxNetExamined] [-d maxDiameter] [resultOutputFile]
```

<i>geneTree1</i> [, <i>geneTree2</i> ...]	Comma delimited collection of gene trees.	mandatory
<i>numReticulations</i>	Maximum number of reticulation nodes to added.	mandatory
<i>-b threshold</i>	Gene trees bootstrap threshold.	optional
<i>-a taxaMap</i>	Gene trees / species tree taxa association.	optional
<i>-s startingNetwork</i>	Specify the network to start search. Default value is the optimal MDC tree.	optional
<i>-n numNetReturned</i>	Number of optimal networks to return. Default value is 1.	optional
<i>-m maxNetExamined</i>	Maximum number of network topologies to examined. Default value is infinity.	optional
<i>-d maxDiameter</i>	Maximum diameter to make an arrangement during network search. Default value is infinity.	optional
<i>resultOutputFile</i>	Optional file destination for command output.	optional



# Inferring a Phylogenetic Network Under Parsimony

## ❖ Command Usage

```
inferNetwork_parsimony (geneTree1 [, geneTree2...]) numReticulations  
[-b threshold] [-a taxaMap] [-s startingNetwork] [-n numNetReturned]  
[-m maxNetExamined] [-d maxDiameter] [resultOutputFile]
```

<i>geneTree1</i> [, <i>geneTree2</i> ...]	Comma delimited collection of gene trees.	mandatory
<i>numReticulations</i>	Maximum number of reticulation nodes to added.	mandatory
<i>-b threshold</i>	Gene trees bootstrap threshold.	optional
<i>-a taxaMap</i>	Gene trees / species tree taxa association.	optional
<i>-s startingNetwork</i>	Specify the network to start search. Default value is the optimal MDC tree.	optional
<i>-n numNetReturned</i>	Number of optimal networks to return. Default value is 1.	optional
<i>-m maxNetExamined</i>	Maximum number of network topologies to examined. Default value is infinity.	optional
<i>-d maxDiameter</i>	Maximum diameter to make an arrangement during network search. Default value is infinity.	optional
<i>resultOutputFile</i>	Optional file destination for command output.	optional

# Inferring a Phylogenetic Network Under Parsimony

---

## ❖ Input NEXUS File

```
#NEXUS

BEGIN TREES;
Tree gt0 = ((((((Scer,Spar),Smik),Skud),Sbay),Scas),Sklu),Calb);
...
Tree gt105 = ((((((Scer,Spar),Smik),Skud),Sbay),(Scas,Sklu)),Calb);
END;

BEGIN PHYLONET;
InferNetwork_Parsimony (gt0,...,gt105) 1;
END;
```

## ❖ Output

```
((((((((Sbay)#H1:::0.36082474226804123,Skud),
((Spar,Scer),Smik)),#H1:::0.6391752577319587),Scas),Sklu),Calb);
Total number of extra lineages: 60
```

# The Likelihood of a Phylogenetic Tree/Network

---

## ❖ Command Usage

```
CalGTProb speciesNetwork (geneTree1 [, geneTree2...]) [-m ac|mul] [-b threshold] [-a taxaMap] [resultOutputFile]
```

<i>speciesNetwork</i>	The candidate species network	mandatory
<i>geneTree1 [, geneTree2...]</i>	Comma delimited collection of gene trees.	mandatory
<i>-m ac mul</i>	Specify the algorithm used for computation. The default value is <i>ac</i> .	optional
<i>-b threshold</i>	Gene trees bootstrap threshold.	optional
<i>-a taxaMap</i>	Gene trees / species tree taxa association.	optional
<i>resultOutputFile</i>	Optional file destination for command output.	optional

# The Likelihood of a Phylogenetic Tree/Network

---

## ❖ Input NEXUS File

```
#NEXUS

BEGIN NETWORKS;
Network sn = (Calb:1.0,(Sklu:1.0,(Scas:1.0,((Sbay:1.0)618961#H1:1.0::0.6365,((Smik:
1.0,(Spar:1.0,Scer:1.0)i7:3.5682)i6:1.1771,(618961#H1:1.0::0.3635,Skud:1.0)
617975:2.1601)i5:5.9384)i4:5.9094)i3:0.7001)i2:5.9079)i1;
END;

BEGIN TREES;
Tree gt0 = (((((((Scer,Spar),Smik),Skud),Sbay),Scas),Sklu),Calb);
...
Tree gt105 = (((((((Scer,Spar),Smik),Skud),Sbay),(Scas,Sklu)),Calb);
END;

BEGIN PHYLONET;
CalGTProb sn (gt0,...,gt105);
END;
```

## ❖ Output

```
[x41] (((((((Scer,Spar),Smik),Skud),Sbay),Scas),Sklu),Calb); : 0.33635215497586934
[x2] (((((Scer,Spar),Smik,(Skud,Sbay)),Scas),Sklu),Calb); : 0.17475146005978076
...
Total log probability: -249.61245619098128
```

# Maximum Likelihood Inference of a Phylogenetic Tree/Network

## ❖ Command Usage

```
inferNetwork_ML (geneTree1 [, geneTree2...]) numReticulations [-b threshold]
[-a taxaMap] [-s startingNetwork] [-n numNetReturned] [-m maxNetExamined] [-d maxDiameter]
[-p (rel,abs)] [-r maxRounds] [-t maxTryPerBranch] [-i improveThreshold]
[-l maxBranchLength][resultOutputFile]
```

<i>geneTree1</i> [, <i>geneTree2</i> ...]	Comma delimited collection of gene trees.	mandatory
<i>numReticulations</i>	Maximum number of reticulations to added.	mandatory
<i>-b threshold</i>	Gene trees bootstrap threshold.	optional
<i>-a taxaMap</i>	Gene trees / species tree taxa association.	optional
<i>-s startingNetwork</i>	Specify the network to start search. Default value is the optimal MDC tree.	optional
<i>-n numNetReturned</i>	Number of optimal networks to return. Default value is 1.	optional
<i>-m maxNetExamined</i>	Maximum number of network topologies to examined. Default value is infinity.	optional
<i>-d maxDiameter</i>	Maximum diameter to make an arrangement during network search. Default value is infinity.	optional
<i>-p (rel, abs)</i>	The original stopping criterion of Brent's algorithm. Default value is (0.01, 0.001).	optional
<i>-r maxRound</i>	Maximum number of rounds to optimize branch lengths for a network topology. Default value is 100.	optional
<i>-t maxTryPerBranch</i>	Maximum number of trial per branch in one round to optimize branch lengths for a network topology. Default value is 100.	optional
<i>-i improveThreshold</i>	Minimum threshold of improvement to continue the next round of optimization of branch lengths. Default value is 0.001.	optional
<i>-l maxBranchLength</i>	Maximum branch lengths considered for optimization. Default value is 6.	optional
<i>resultOutputFile</i>	Optional file destination for command output.	optional

# Maximum Likelihood Inference of a Phylogenetic Tree/Network

## ❖ Command Usage

```
inferNetwork_ML (geneTree1 [, geneTree2...]) numReticulations [-b threshold]
[-a taxaMap] [-s startingNetwork] [-n numNetReturned] [-m maxNetExamined] [-d maxDiameter]
[-p (rel,abs)] [-r maxRounds] [-t maxTryPerBranch] [-i improveThreshold]
[-l maxBranchLength][resultOutputFile]
```

<i>geneTree1</i> [, <i>geneTree2</i> ...]	Comma delimited collection of gene trees.	mandatory
<i>numReticulations</i>	Maximum number of reticulations to added.	mandatory
<i>-b threshold</i>	Gene trees bootstrap threshold.	optional
<i>-a taxaMap</i>	Gene trees / species tree taxa association.	optional
<i>-s startingNetwork</i>	Specify the network to start search. Default value is the optimal MDC tree.	optional
<i>-n numNetReturned</i>	Number of optimal networks to return. Default value is 1.	optional
<i>-m maxNetExamined</i>	Maximum number of network topologies to examined. Default value is infinity.	optional
<i>-d maxDiameter</i>	Maximum diameter to make an arrangement during network search. Default value is infinity.	optional
<i>-p (rel, abs)</i>	The original stopping criterion of Brent's algorithm. Default value is (0.01, 0.001).	optional
<i>-r maxRound</i>	Maximum number of rounds to optimize branch lengths for a network topology. Default value is 100.	optional
<i>-t maxTryPerBranch</i>	Maximum number of trial per branch in one round to optimize branch lengths for a network topology. Default value is 100.	optional
<i>-i improveThreshold</i>	Minimum threshold of improvement to continue the next round of optimization of branch lengths. Default value is 0.001.	optional
<i>-l maxBranchLength</i>	Maximum branch lengths considered for optimization. Default value is 6.	optional
<i>resultOutputFile</i>	Optional file destination for command output.	optional

# Maximum Likelihood Inference of a Phylogenetic Tree/Network

---

## ❖ Input NEXUS File

```
#NEXUS

BEGIN TREES;
Tree gt0 = ((((((Scer,Spar),Smik),Skud),Sbay),Scas),Sklu),Calb);
...
Tree gt105 = ((((((Scer,Spar),Smik),Skud),Sbay),(Scas,Sklu)),Calb);
END;

BEGIN PHYLONET;
InferNetwork_ML (gt0,...,gt105) 1;
END;
```

## ❖ Output

```
(Calb:1.0,(Sklu:1.0,(Scas:1.0,((Sbay:1.0)#H1:1.0::0.6365,((Smik:1.0,(Spar:1.0,Scer:1.0):
3.5682):1.1771,(#H1:1.0::0.3635,Skud:1.0):2.1601):5.9384):5.9094):0.7001):5.9079);
Total log probability: -249.61245619098128
```

# Try It Yourself!

---

- ❖ <http://www.cs.utexas.edu/users/tandy/workshop-schedule.html>