

Séminaire du groupe BioInformatique et Visualisation
19/10/2017 - Bordeaux

***Finding a gene tree
in a phylogenetic network***

Philippe Gambette



Outline

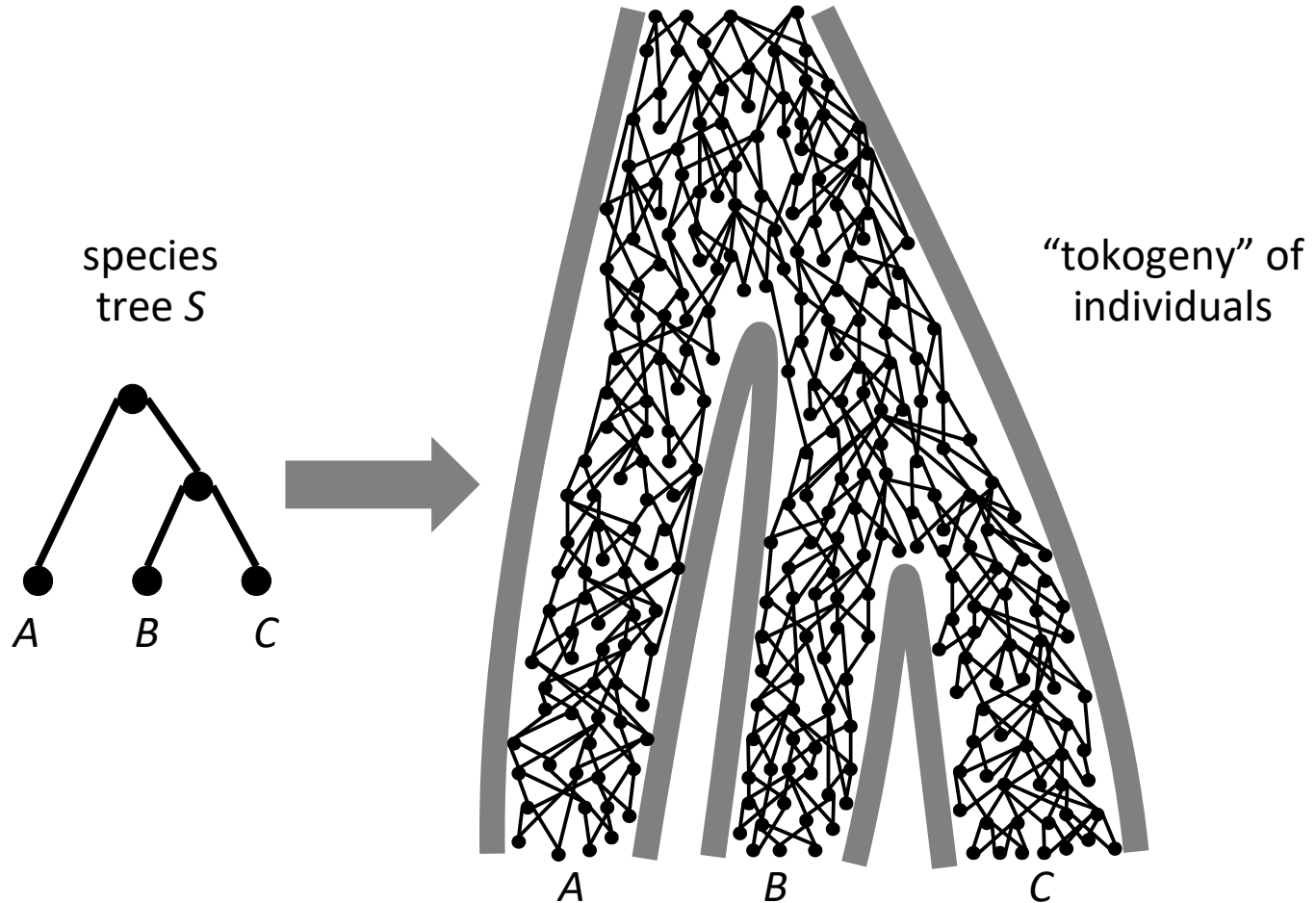
- Phylogenetic networks
- Classes of phylogenetic networks
- The Tree Containment Problem

Outline

- Phylogenetic networks
- Classes of phylogenetic networks
- The Tree Containment Problem

Phylogenetic trees

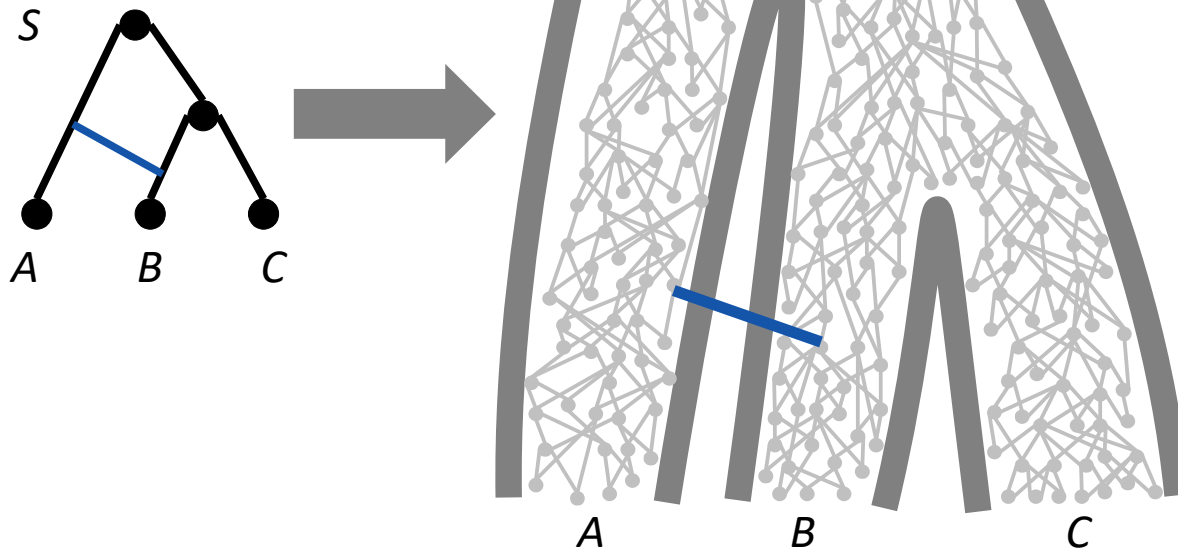
Phylogenetic tree of a set of species



Genetic material transfers

Transfers of genetic material between coexisting species:

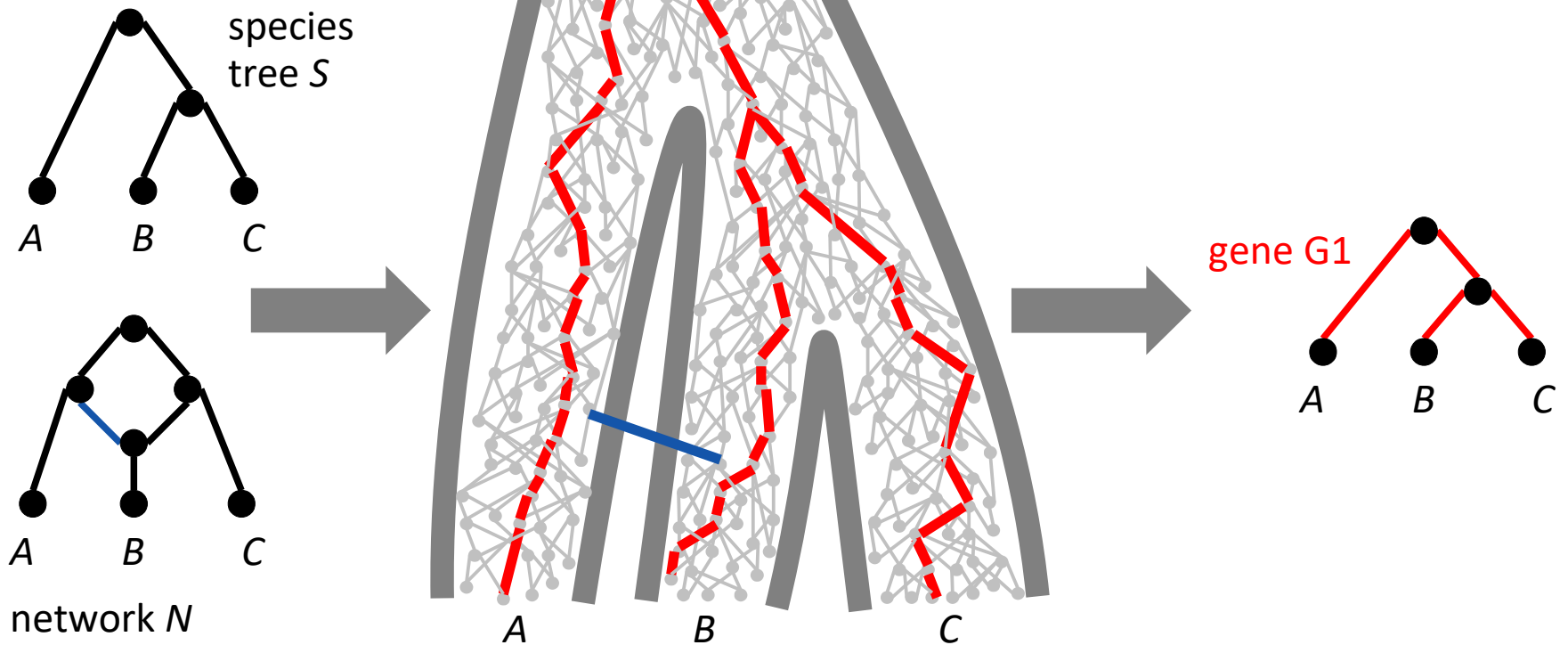
- lateral gene transfer
- hybridization
- recombination



Genetic material transfers

Transfers of genetic material between coexisting species:

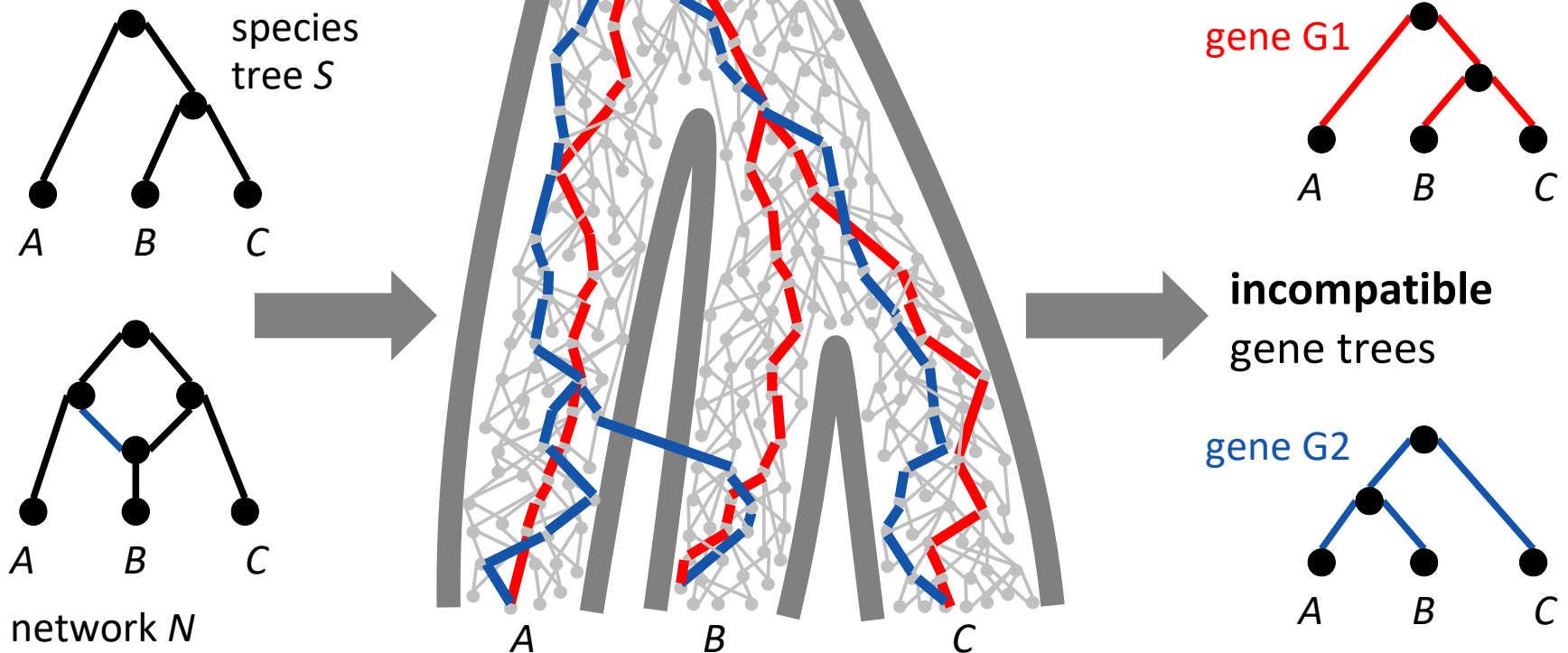
- lateral gene transfer
- hybridization
- recombination



Genetic material transfers

Transfers of genetic material between coexisting species:

- lateral gene transfer
- hybridization
- recombination

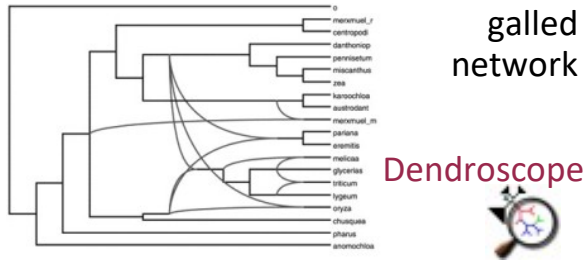


Phylogenetic networks

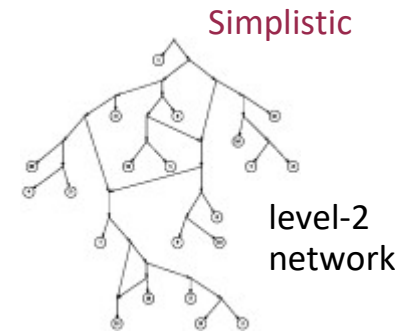
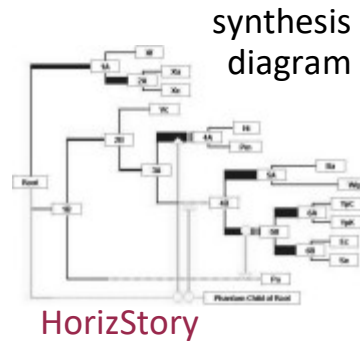
Phylogenetic network: network representing evolution data

- **explicit** phylogenetic networks

model evolution



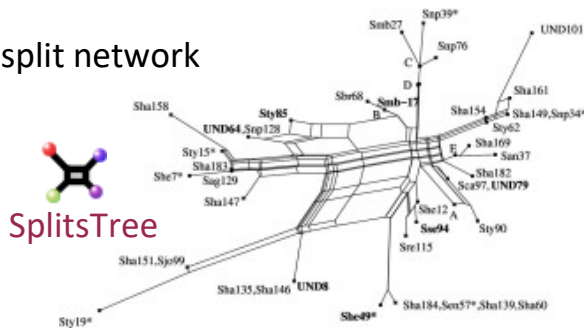
Dendroscope



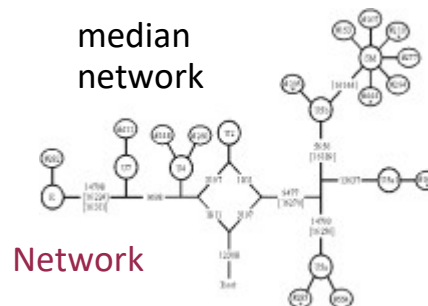
- **abstract** phylogenetic networks

classify, visualize data

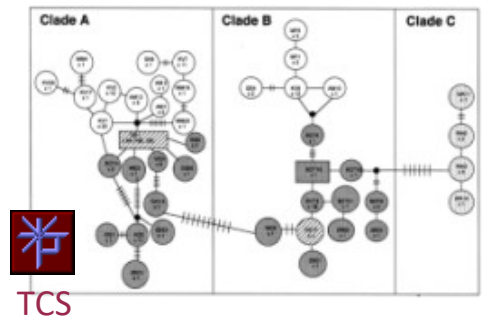
split network



median network

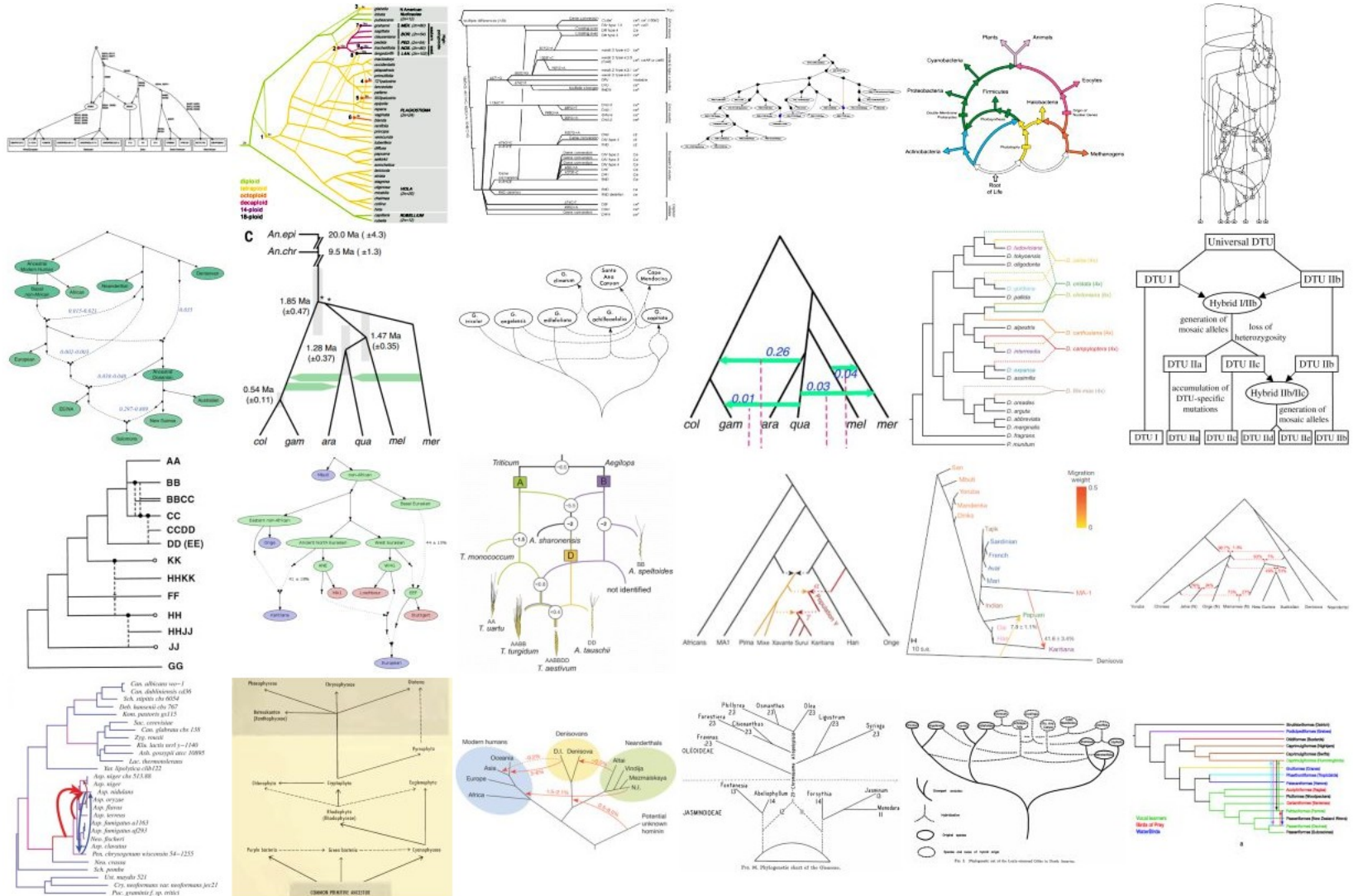


minimum spanning network

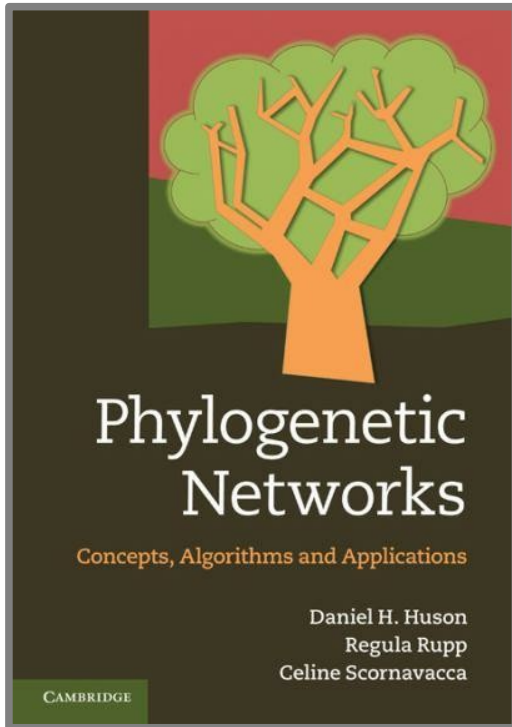


Today: focus on explicit phylogenetic networks

A gallery of explicit phylogenetic networks :

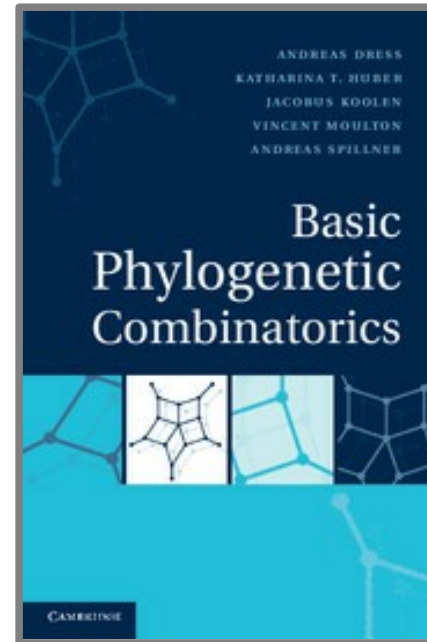


Books about phylogenetic networks

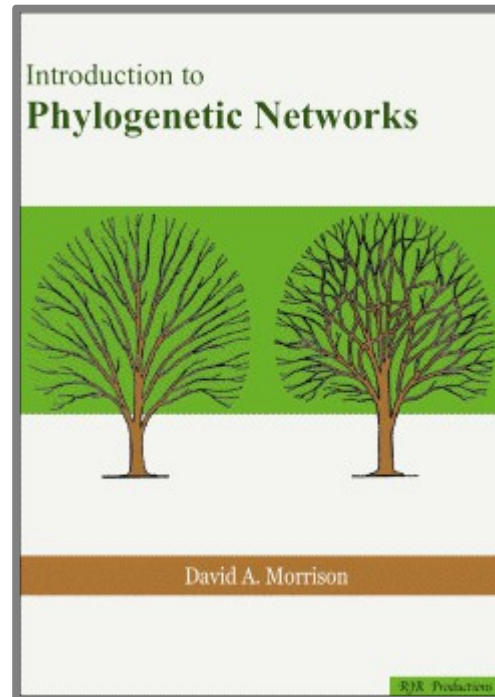


Huson, Rupp,
Scornavacca, 2011

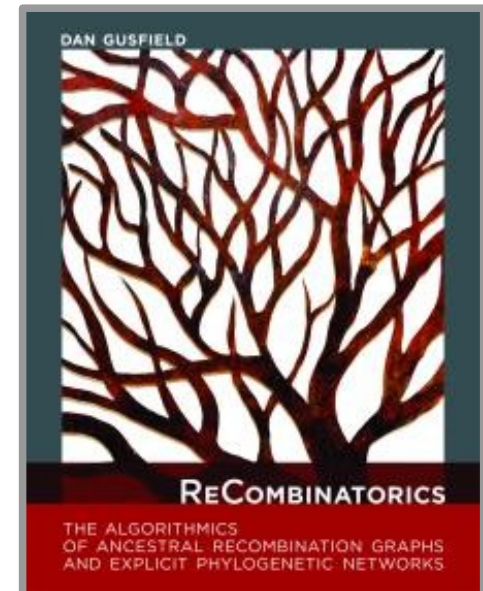
Dress, Huber,
Koolen, Moulton,
Spillner, 2012



Gusfield, 2014



Morrison, 2011



Workshops about phylogenetic networks



The Future of Phylogenetic Networks, 15-19 October **2012**, Lorentz Center (Leiden, The Netherlands)

Utilizing Genealogical Phylogenetic Networks in Evolutionary Biology: Touching the Data, 7-11 July **2014**, Lorentz Center (Leiden, The Netherlands)



The Phylogenetic Network Workshop, 27-31 July **2015**, Institute for Mathematical Science (National University of Singapore)

Distinguishability in Genealogical Phylogenetic Networks, 13-17 August **2018** (Leiden, The Netherlands)

Who is who in Phylogenetic Networks?

Who is Who in Phylogenetic Networks

🏠 Authors Community Keywords Publications Software Browse Basket Account Contribute! About Help 📡 🔍

FIND EXPERTS

Find researchers working on a specific topic, in a given country, and find where (journals, conferences) the community publishes or meets.

EXPLORE RESEARCH

Browse publications, access keyword definitions and find trends in publications on phylogenetic network methods and methodologies.

DISCOVER SOFTWARE

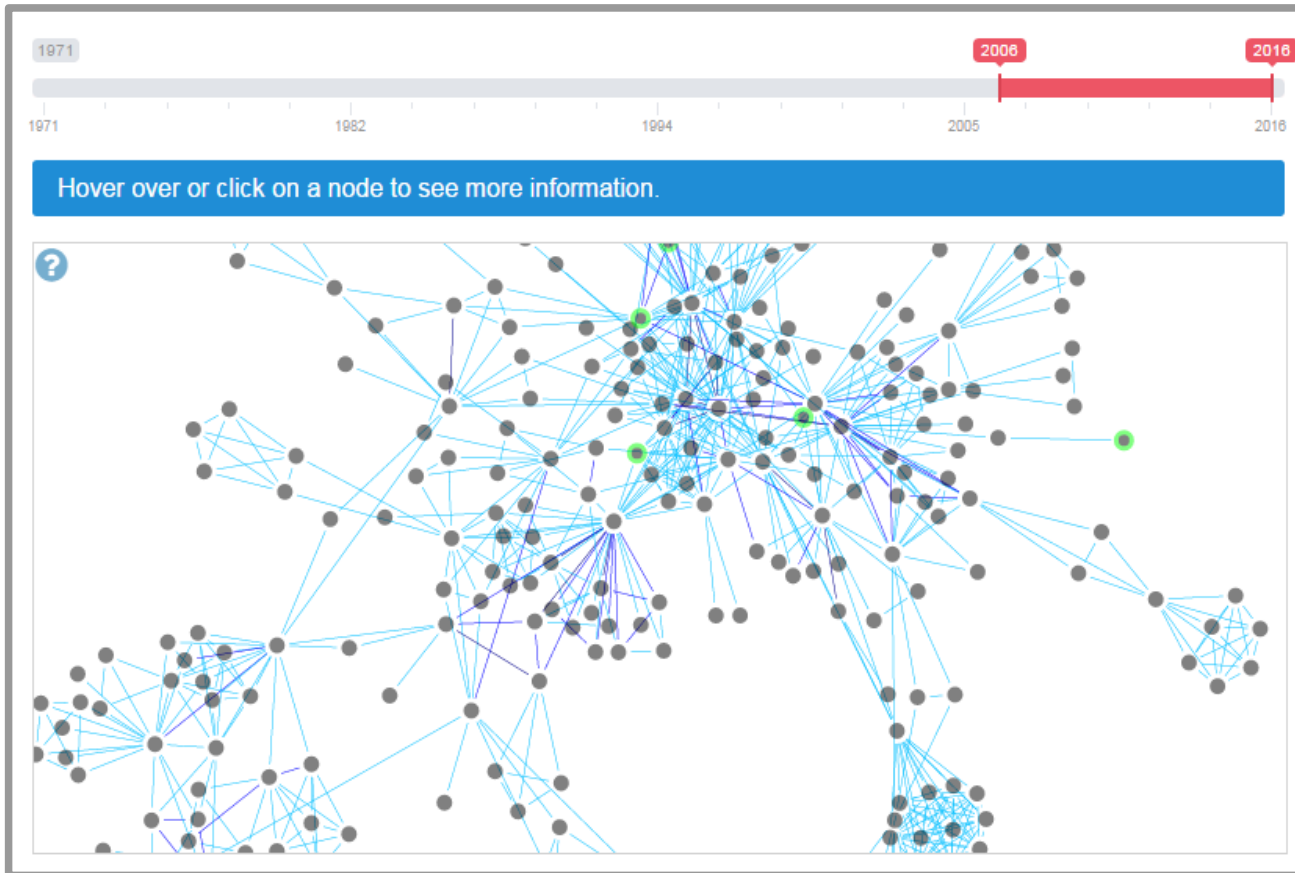
Locate programs to compute, evaluate, compare or visualize phylogenetic networks, and view how these are linked with each other and input data.

FOLLOW COMMUNITY

Follow an author, publications tagged with a keyword, or the entire database using the 📡 icon in the menu, on an author's page, or on a keyword's page.

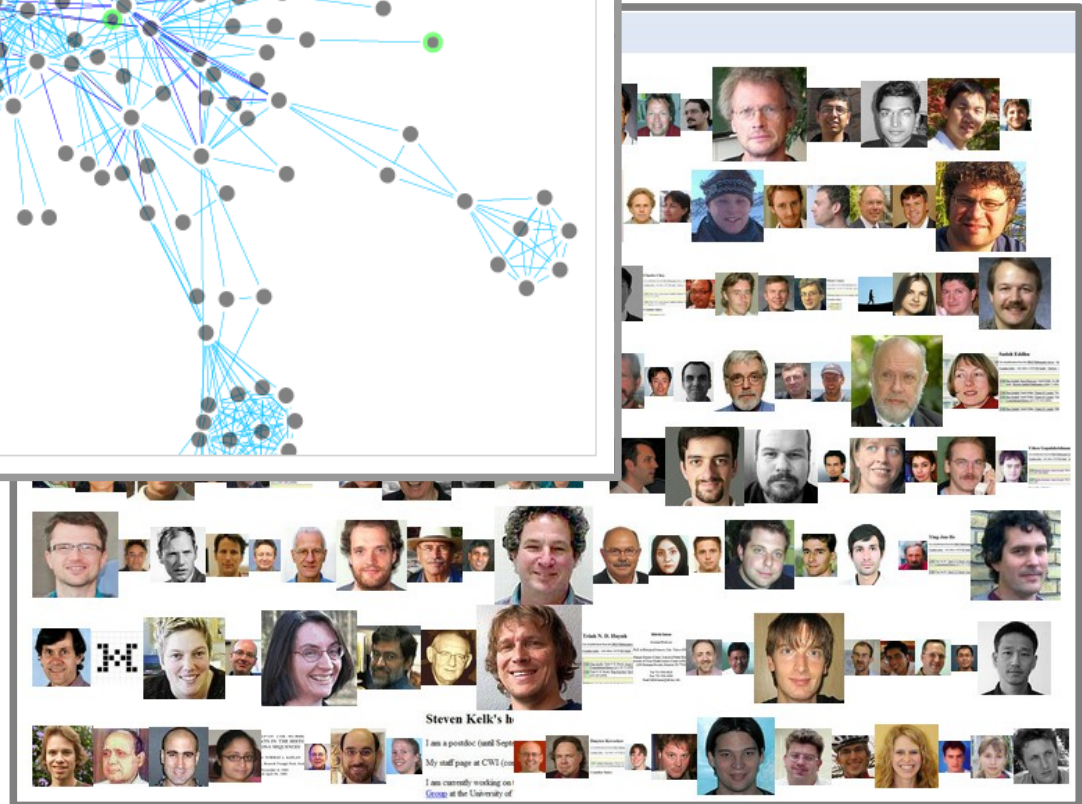
Based on BibAdmin by Sergiu Chelcea
+ tag clouds, date histograms, journal lists,
keyword definitions, co-author graphs

Who is who in Phylogenetic Networks?



Who is Who in
Phylogenetic
Networks, Articles,
Authors &
Programs

Analysis of the co-author and
keyword graphs: internship of
Tushar Agarwal
Agarwal, Gambette &
Morrison, arXiv, 2017



Who is who in Phylogenetic Networks?

abstract-network(88) agreement-forest(29) approximation(18) APX-hard(4) ARG(6) bayesian(4) block-realization(1) bootstrap(4) bound(4) branch-and-bound(1) cactus-graph(1) characterization(11) circular-split-system(13) clustering(3) coalescent(10) consensus(8) consistency(2) cophylogeny(1) counting(3) database(1) distance-between-networks(30) diversity(5) duplication(33) dynamic-programming(10) enumeration(4) **evaluation(25)** **explicit-network(191)** exponential-algorithm(6) FPT(31) from-clusters(16) from-continuous-characters(1) from-distances(45) from-gene-order(1) from-multilabeled-tree(9) from-network(39) from-NGS-data(1) from-quartets(16) **from-rooted-trees(122)** from-sequences(51) from-species-tree(46) from-splits(13) from-trees(7) from-trinets(2) from-triplets(24) from-unrooted-trees(16) galled-network(7) galled-tree(38) generation(12) haplotype-network(2) haplotyping(1) heuristic(32) HMM(2) hybridization(52) inapproximability(5) integer-linear-programming(3) isomorphism(3) k-reticulated(1) kernelization(2) labeling(4) lateral-gene-transfer(56) level-k-phylogenetic-network(30) likelihood(16) lineage-sorting(10) loss(16) MASN(4) median-network(16) MedianJoining(2) Minimal-lateral-network(1) minimum-contradiction(2) **minimum-number(33)** minimum-spanning-network(2) model-selection(2) mu-distance(2) **NeighborNet(14)** nested-network(2) netting(3) normal-network(9) NP-complete(33) optimal-realization(3) parsimony(39) pedigree(1) perfect(5) **phylogenetic-network(402)** **phylogeny(387)** **polynomial(70)** population-genetics(5) Program-AdmixTools(1) Program-ALE(1) Program-Angst(2) Program-Arlequin(5) Program-Beagle(3) Program-BIMLR(1) Program-Bio-PhyloNetwork(4) Program-Clustistic(2) Program-CMPT(1) Program-CombineTrees(2) Program-ConsensusNetwork(1) Program-constNJ(1) Program-CycleKiller(3) **Program-Dendroscope(14)** Program-EEEP(3) Program-FastHN(1) Program-FlatNJ(1) Program-Fylogenetica(2) Program-GalledTree(1) Program-GraphDTL(2) Program-HapBound(1) Program-HGT_simul(1) Program-HiDe(1) Program-HorizStory(2) Program-Hybrid-Lambda(1) Program-HybridInterleave(5) Program-HybridNET(2) Program-HybridNumber(3) Program-Hybroscale(3) Program-JML(1) Program-LatTrans(5) Program-LEV1ATHAN(1) Program-Level1Generator(1) Program-Level2(2) Program-lingpy(1) Program-LNetwork(2) Program-MaafB(2) Program-Marlon(3) Program-MC-Net(1) Program-McKITsch(1) Program-Mowgli(5) Program-MowgliNNI(2) Program-MPNet(1) Program-MY-CLOSURE(1) Program-Nepal(7) Program-NetGen(3) Program-NetTest(1) Program-NetView(1) Program-Network(5) Program-Notung(1) Program-PADRE(7) Program-Phangorn(2) Program-PhippsNetwork(2) **Program-PhyloNet(11)** Program-PhyloNet-HMM(1) Program-PIRN(3) Program-Prunier(2) Program-Pyramids(3) Program-QNet(4) Program-Quartet(1) Program-Quartet-Decomposition(1) Program-QuartetMethods(1) Program-QuartetNet(2) Program-QuasiDec(1) Program-QuickCass(1) Program-RANGER-DTL(4) Program-RecMin(1) Program-Recodon(3) Program-RecPars(1) Program-Reticlad(2) Program-SAGE(2) Program-SAQ-Net(1) Program-Serial-NetEvolve(1) Program-SHRUB(3) Program-Simplistic(3) Program-Sliding-MinPD(1) Program-SNSA(2) Program-Spectronet(4) **Program-SplitsTree(36)** Program-SPNet(5) Program-SPRDist(1) Program-SuperQ(1) **Program-T-REX(13)** Program-TCS(8) Program-TERA(2) Program-TerminusEst(2) Program-TreeFix-DTL(1) Program-TreeMix(1) Program-Treevolve(2) Program-TripNet(2) Program-ultra-Net(1) Program-Ultratnet(1) Program-WeakHierarchies(2) Program-Xscape(1) pyramid(8) quasi-median-network(3) realization(4) **recombination(29)** recombination-detection(4) **reconstruction(255)** regular-network(7) reticulogram(10) serial-evolutionary-networks(1) simulated-annealing(4) simulation(5) site-consistency(1) **software(66)** split(27) split-decomposition(13) split-network(54) SPR-distance(13) spread(2) **statistical-model(32)** statistical-parsimony(3) supernet(4) **survey(31)** tanglegram(1) time-consistent-network(12) tree-child-network(15) tree-sibling-network(10) tripartition-distance(9) triplet-distance(3) unicyclic-network(3) **visualization(30)** weak-hierarchy(8) weakly-compatible(3)

Who is who in Phylogenetic Networks?

abstract-network(88) agreement-forest(29) approximation(18) APX-hard(4) ARG(6) bayesian(4) block-realization(1) bootstrap(4) bound(4) branch-and-bound(1) cactus-graph(1) characterization(11) circular-split-system(13) clustering(3) coalescent(10) consensus(8) consistency(2) cophylogeny(1) counting(3) database(1) distance-between-networks(30) diversity(5) duplication(33) dynamic-programming(10) enumeration(4) **evaluation(25)** **explicit-network(191)** exponential-algorithm(6) **FPT(31)** from-clusters(16) from-continuous-characters(1) from-distances(45) from-gene-order(1) from-multilabeled-tree(9) from-network(39) from-NGS-data(1) from-quartets(16) **from-rooted-trees(122)** from-sequences(51) from-species-tree(46) from-splits(13) from-trees(7) from-trinets(2) from-triplets(24) from-unrooted-trees(16) galled-network(7) galled-tree(38) generation(12) haplotype-network(2) haplotyping(1) heuristic(32) HMM(2) hybridization(52) inapproximability(5) integer-linear-programming(3) isomorphism(3) k-reticulated(1) kernelization(2) labeling(4) lateral-gene-transfer(56) level-k-phylogenetic-network(30) likelihood(16) lineage-sorting(10) loss(16) MASN(4) median-network(16) MedianJoining(2) Minimal-lateral-network(1) minimum-contradiction(2) **minimum-number(33)** minimum-spanning-network(2) model-selection(2) mu-distance(2) NeighborNet(14) nested-network(2) netting(3) normal-network(9) NP-complete(33) optimal-realization(3) parsimony(39) pedigree(1) perfect(5) **phylogenetic-network(402)** **phylogeny(387)** **polynomial(70)** population-genetics(5) Program-AdmixTools(1) Program-ALE(1) Program-Angst(2) Program-Arlequin(5) Program-Beagle(3) Program-BIMLR(1) Program-Bio-PhyloNetwork(4) Program-Clustistic(2) Program-CMPT(1) Program-CombineTrees(2) Program-ConsensusNetwork(1) Program-constNJ(1) Program-CycleKiller(3) **Program-Dendroscope(14)** Program-EEEP(3) Program-FastHN(1) Program-FlatNJ(1) Program-Fylogenetica(2) Program-GalledTree(1) Program-GraphDTL(2) Program-HapBound(1) Program-HGT_simul(1) Program-HiDe(1) Program-HorizStory(2) Program-Hybrid-Lambda(1) Program-HybridInterleave(5) Program-HybridNET(2) Program-HybridNumber(3) Program-Hybroscale(3) Program-JML(1) Program-LatTrans(5) Program-LEV1ATHAN(1) Program-Lev1Generator(1) Program-Level2(2) Program-lingpy(1) Program-LNetwork(2) Program-MaafB(2) Program-Marlon(3) Program-MC-Net(1) Program-McKITsch(1) Program-Mowgli(5) Program-MowgliNNI(2) Program-MPNet(1) Program-MY-CLOSURE(1) Program-Nepal(7) Program-NetGen(3) Program-NetTest(1) Program-NetView(1) Program-Network(5) Program-Notung(1) Program-PADRE(7) Program-Phangorn(2) Program-PhippsNetwork(2) **Program-PhyloNet(11)** Program-PhyloNet-HMM(1) Program-PIRN(3) Program-Prunier(2) Program-Pyramids(3) Program-QNet(4) Program-Quartet(1) Program-Quartet-Decomposition(1) Program-QuartetMethods(1) Program-QuartetNet(2) Program-QuasiDec(1) Program-QuickCass(1) Program-RANGER-DTL(4) Program-RecMin(1) Program-Recodon(3) Program-RecPars(1) Program-Reticlad(2) Program-SAGE(2) Program-SAQ-Net(1) Program-Serial-NetEvolve(1) Program-SHRUB(3) Program-Simplistic(3) Program-Sliding-MinPD(1) Program-SNSA(2) Program-Spectronet(4) **Program-SplitsTree(36)** Program-SPNet(5) Program-SPRDist(1) Program-SuperQ(1) **Program-T-REX(13)** Program-TCS(8) Program-TERA(2) Program-TerminusEst(2) Program-TreeFix-DTL(1) Program-TreeMix(1) Program-Treevolve(2) Program-TripNet(2) Program-ultra-Net(1) Program-Ultranet(1) Program-WeakHierarchies(2) Program-Xscape(1) pyramid(8) quasi-median-network(3) realization(4) **recombination(29)** recombination-detection(4) **reconstruction(255)** regular-network(7) reticulogram(10) serial-evolutionary-networks(1) simulated-annealing(4) simulation(5) site-consistency(1) **software(66)** split(27) split-decomposition(13) split-network(54) SPR-distance(13) spread(2) statistical-model(32) statistical-parsimony(3) supernet(4) survey(31) tanglegram(1) time-consistent-network(12) tree-child-network(15) tree-sibling-network(10) tripartition-distance(9) triplet-distance(3) unicyclic-network(3) visualization(30) weak-hierarchy(8) weakly-compatible(3)

input

software

Who is who in Phylogenetic Networks?

abstract-network(88) agreement-forest(29) approximation(18) APX-hard(4) ARG(6) bayesian(4) block-realization(1) bootstrap(4) bound(4) branch-and-bound(1) cactus-graph(1) characterization(11) circular-split-system(13) clustering(3) coalescent(10) consensus(8) consistency(2) cophylogeny(1) counting(3) database(1) distance-between-networks(30) diversity(5) duplication(33) dynamic-programming(10) enumeration(4) evaluation(25) **explicit-network(191)** exponential-algorithm(6) **FPT(31)** from-clusters(16) from-continuous-characters(1) from-distances(45) from-gene-order(1) from-multilabeled-tree(9) from-network(39) from-NGS-data(1) from-quartets(16) **from-rooted-trees(122)** from-sequences(51) from-species-tree(46) from-splits(13) from-trees(7) from-trinets(2) from-triplets(24) from-unrooted-trees(16) galled-network(7) **galled-tree(38)** generation(12) haplotype-network(2) haplotyping(1) **heuristic(32)** HMM(2) hybridization(52) inapproximability(5) integer-linear-programming(3) isomorphism(3) k-reticulated(1) kernelization(2) labeling(4) lateral-gene-transfer(56) **level-k-phylogenetic-network(30)** likelihood(16) lineage-sorting(10) loss(16) MASN(4) **median-network(16)** MedianJoining(2) Minimal-lateral-network(1) minimum-contradiction(2) **minimum-number(33)** minimum-spanning-network(2) model-selection(2) mu-distance(2) NeighborNet(14) **nested-network(2)** netting(3) **normal-network(9)** **NP-complete(33)** optimal-realization(3) parsimony(39) pedigree(1) perfect(5) **phylogenetic-network(402)** **phylogeny(387)** **polynomial(70)** population-genetics(5) Program-AdmixTools(1) Program-ALE(1) Program-Angst(2) Program-Arlequin(5) Program-Beagle(3) Program-BIMLR(1) Program-Bio-PhyloNetwork(4) Program-Clustistic(2) Program-CMPT(1) Program-CombineTrees(2) Program-ConsensusNetwork(1) Program-constNJ(1) Program-CycleKiller(3) **Program-Dendroscope(14)** Program-EEEP(3) Program-FastHN(1) Program-FlatNJ(1) Program-Fylogenetica(2) Program-GalledTree(1) Program-GraphDTL(2) Program-HapBound(1) Program-HGT_simul(1) Program-HiDe(1) Program-HorizStory(2) Program-Hybrid-Lambda(1) Program-HybridInterleave(5) Program-HybridNET(2) Program-HybridNumber(3) Program-Hybroscale(3) Program-JML(1) Program-LatTrans(5) Program-LEV1ATHAN(1) Program-Level1Generator(1) Program-Level2(2) Program-lingpy(1) Program-LNetwork(2) Program-MaafB(2) Program-Marlon(3) Program-MC-Net(1) Program-McKITsch(1) Program-Mowgli(5) Program-MowgliNNI(2) Program-MPNet(1) Program-MY-CLOSURE(1) Program-Nepal(7) Program-NetGen(3) Program-NetTest(1) Program-NetView(1) Program-Network(5) Program-Notung(1) Program-PADRE(7) Program-Phangorn(2) Program-PhippsNetwork(2) **Program-PhyloNet(11)** Program-PhyloNet-HMM(1) Program-PIRN(3) Program-Prunier(2) Program-Pyramids(3) Program-QNet(4) Program-Quartet(1) Program-Quartet-Decomposition(1) Program-QuartetMethods(1) Program-QuartetNet(2) Program-QuasiDec(1) Program-QuickCass(1) Program-RANGER-DTL(4) Program-RecMin(1) Program-Recodon(3) Program-RecPars(1) Program-Reticlad(2) Program-SAGE(2) Program-SAQ-Net(1) Program-Serial-NetEvolve(1) Program-SHRUB(3) Program-Simplistic(3) Program-Sliding-MinPD(1) Program-SNSA(2) Program-Spectronet(4) **Program-SplitsTree(36)** Program-SPNet(5) Program-SPRDist(1) Program-SuperQ(1) **Program-T-REX(13)** Program-TCS(8) Program-TERA(2) Program-TerminusEst(2) Program-TreeFix-DTL(1) Program-TreeMix(1) Program-Treevolve(2) Program-TripNet(2) Program-ultra-Net(1) Program-Ultratnet(1) Program-WeakHierarchies(2) Program-Xscape(1) **pyramid(8)** **quasi-median-network(3)** realization(4) recombination(29) recombination-detection(4) **reconstruction(255)** **regular-network(7)** reticulogram(10) serial-evolutionary-networks(1) simulated-annealing(4) simulation(5) site-consistency(1) **software(66)** split(27) split-decomposition(13) split-network(54) **SPR-distance(13)** **spread(2)** **statistical-model(32)** statistical-parsimony(3) supernet(4) survey(31) tanglegram(1) **time-consistent-network(12)** **tree-child-network(15)** **tree-sibling-network(10)** tripartition-distance(9) triplet-distance(3) unicyclic-network(3) **visualization(30)** weak-hierarchy(8) weakly-compatible(3)

input

software

classes

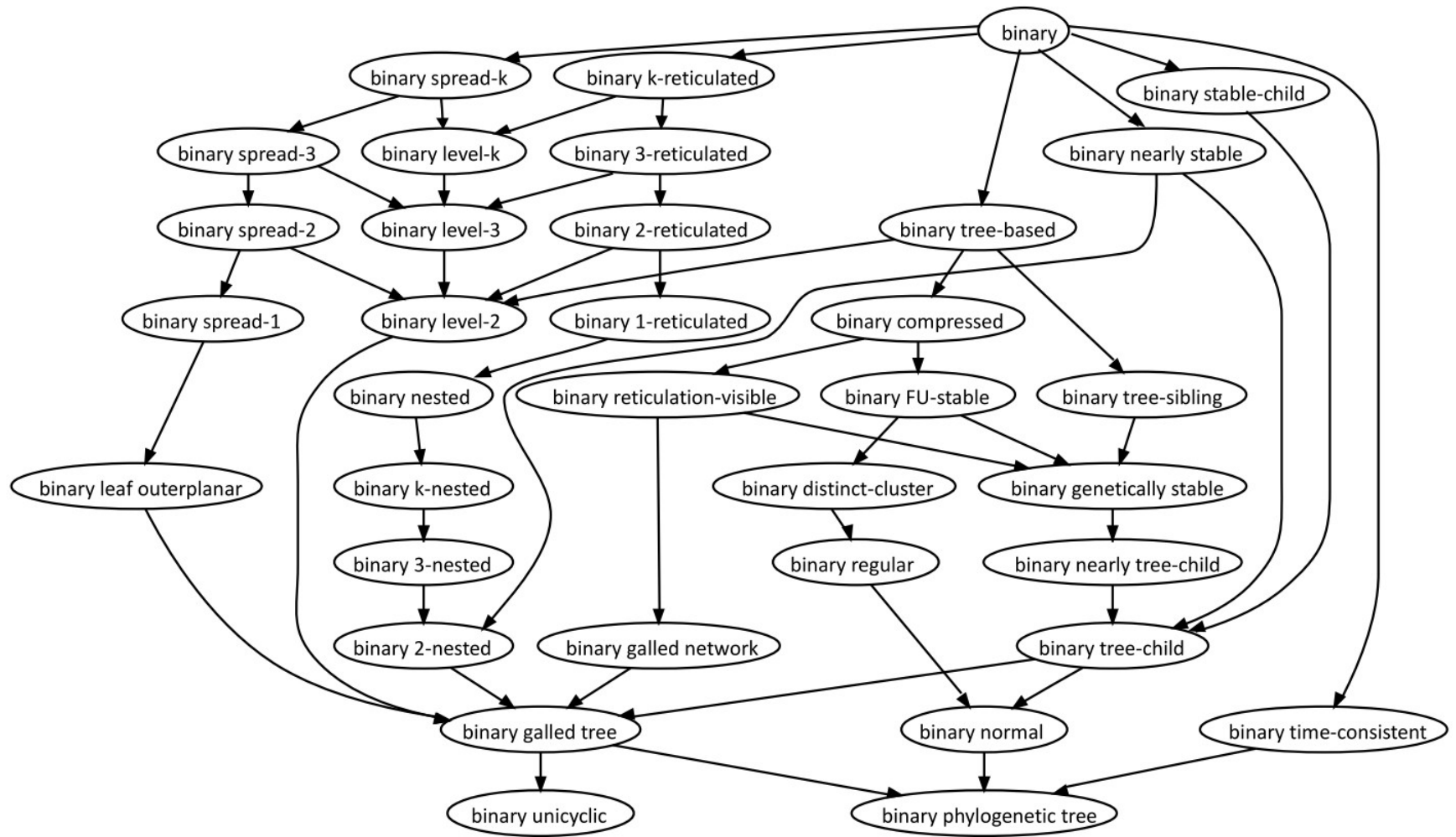
problems

algorithmic properties

Outline

- Phylogenetic networks
- Classes of phylogenetic networks
- The Tree Containment Problem

Classes of phylogenetic networks



Classes of phylogenetic networks: inclusions

<i>binary</i>	unic	gall	tree	nea	gall	gen	reti	tree	con	non	reg	dist	FU	nea	tree	nes	2-n	3-n	level	level	leaf	spr	spr	spr	tim
<u>unicyclic</u>	=	c	c	c	c	c	c	c	c	5	5	5	c	c	c	c	c	c	c	c	c	c	c	c	5
<u>galled tree</u>	6	=	c	c	c	c	c	c	c	5	5	5	c	c	c	c	c	c	c	c	c	c	c	c	5
<u>tree-child</u>	6	7	=	c	c	c	c	c	c	5	5	5	c	c	7	7	7	7	15	15	7	7			5
<u>nearly tree-child</u>	6	7	8	=	8	c	c	c	c	5	5	5	c	8	c	7	7	7	8	15	7	7			5
<u>galled network</u>	2	2	1	1	=	1	c	1	c	1	1	1	14	21	c	7	7	7	18	18	7	7			5
<u>genetically stable</u>	4	4	4	4	4	=	c	c	c	5	5	5	c	8	c	4	4	4	8	15	7	7			4
<u>reticulation-visible</u>	2	2	1	1	4	1	=	1	c	1	1	1	14	8	c	4	4	4	8	15	7	7			4
<u>tree-sibling</u>	4	4	4	4	4	12	12	=	12	5	5	5	12	8	c	4	4	4	8	15	7	7			4
<u>compressed</u>	2	2	1	1	4	1	11	1	=	1	1	1	14	8	c	4	4	4	8	15	7	7			4
<u>normal</u>	7	7	c	c	13	c	c	c	c	=	c	c	c	c	c	7	7	7	15	15	7	7			22
<u>regular</u>	2	2	2	2	13	2	24	2	c	2	=	c	c	8	c	7	7	7	8	15	7	7			22
<u>distinct-cluster</u>	2	2	2	2	13	2	24	2	c	2	9	=	c	8	c	7	7	7	8	15	7	7			22
<u>FU-stable</u>	2	2	1	1	4	1	11	1	c	1	1	1	=	8	c	4	4	4	8	15	7	7			4
<u>nearly stable</u>	2	2	2	2	3	2	3	2	3	2	3	3	3	=	3	7	7	7	3	15	3	3			5
<u>tree-based</u>	2	2	1	1	4	1	12	1	12	1	1	1	12	8	=	4	4	4	8	15	7	7			4
<u>nested</u>	3	3	3	3	3	3	3	3	3	3	3	3	3	19	3	=	20	20	3	16	3	3			5
<u>2-nested</u>	3	3	3	3	3	3	3	3	3	3	3	3	3	c	3	c	=	c	3	16	3	3			5
<u>3-nested</u>	3	3	3	3	3	3	3	3	3	3	3	3	3	19	3	c	19	=	3	16	3	3			5
<u>level-2</u>	4	4	4	4	4	12	12	14	12	4	4	4	12	17	c	4	4	4	=	c	7	7	c	c	4
<u>level-3</u>	2	2	2	2	3	2	3	2	3	2	3	3	3	8	3	4	4	4	3	=	3	3		c	4
<u>leaf outerplanar</u>	4	4	4	4	4	10	10	10	10	4	4	4	10	8	10	4	4	4	8	16	=	c	c	c	4
<u>spread 1</u>	4	4	4	4	4	10	10	10	10	4	4	4	10	8	10	4	4	4	8	16	9	=	c	c	4
<u>spread 2</u>	3	3	3	3	3	3	3	3	3	3	3	3	3	8	3	4	4	4	3	16	3	3	=	c	4
<u>spread 3</u>	3	3	3	3	3	3	3	3	3	3	3	3	3	8	3	4	4	4	3	16	3	3		=	4
<u>time-consistent</u>	3	3	3	3	3	3	3	3	3	3	3	3	3	8	3	7	7	7	3	23	3	3			=

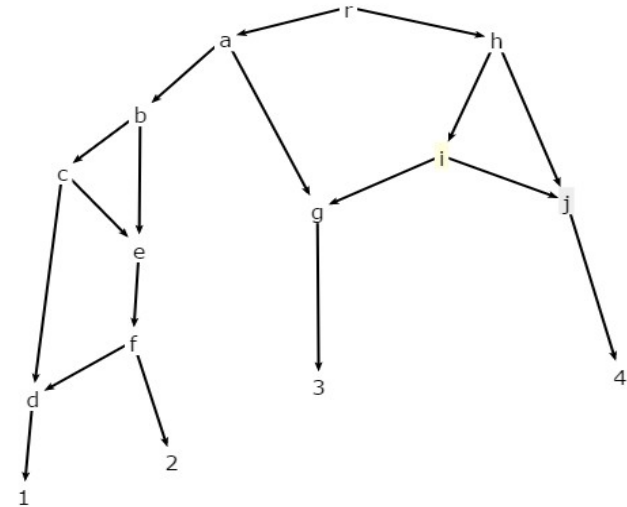
joint work with Maxime Morgado and Narges Tavassoli

Classes of phylogenetic networks: inclusions

<i>binary</i>	unic	gall	tree	nea	gall	gen	reti	tree	con	non	reg	dist	FU	nea	tree	nes	2-n	3-n	level	level	leaf	spr	spr	spr	tim
<u>unicyclic</u>	=	c	c	c	c	c	c	c	c	5	5	5	c	c	c	c	c	c	c	c	c	c	c	c	5
<u>galled tree</u>	6	=	c	c	c	c	c	c	c	5	5	5	c	c	c	c	c	c	c	c	c	c	c	c	5
<u>tree-child</u>	6	7	=	c	c	c	c	c	c	5	5	5	c	c	7	7	7	7	15	15	7	7			5
<u>nearly tree-child</u>	6	7	8	=	8	c	c	c	c	5	5	5	c	8	c	7	7	7	8	15	7	7			5
<u>galled network</u>	2	2	1	1	=	1	c	1	c	1	1	1	14	21	c	7	7	7	18	18	7	7			5
<u>genetically stable</u>	4	4	4	4	4	=	c	c	c	5	5	5	c	8	c	4	4	4	8	15	7	7			4
<u>reticulation-visible</u>	2	2	1	1	4	1	=	1	c	1	1	1	14	8	c	4	4	4	8	15	7	7			4
<u>tree-sibling</u>	4	4	4	4	4	12	12	=	12	5	5	5	12	8	c	4	4	4	8	15	7	7			4
<u>compressed</u>	2	2	1	1	4	1	11	1	=	1	1	1	14	8	c	4	4	4	8	15	7	7			4
<u>normal</u>	7	7	c	c	13	c	c	c	c	=	c	c	c	c	c	7	7	7	15	15	7	7			22
<u>regular</u>	2	2	2	2	13	2	24	2	c	2	=	c	c	8	c	7	7	7	8	15	7	7			22
<u>distinct-cluster</u>	2	2	2	2	13	2	24	2	c	2	9	=	c	8	c	7	7	7	8	15	7	7			22
<u>FU-stable</u>	2	2	1	1	4	1	11	1	c	1	1	1	=	8	c	4	4	4	8	15	7	7			4
<u>nearly stable</u>	2	2	2	2	3	2	3	2	3	2	3	3	3	=	3	7	7	7	3	15	3	3			5
<u>tree-based</u>	2	2	1	1	4	1	12	1	12	1	1	1	12	8	=	4	4	4	8	15	7	7			4
<u>nested</u>	3	3	3	3	3	3	3	3	3	3	3	3	3	19	3	=	20	20	3	16	3	3			5
<u>2-nested</u>	3	3	3	3	3	3	3	3	3	3	3	3	3	c	3	c	=	c	3	16	3	3			5
<u>3-nested</u>	3	3	3	3	3	3	3	3	3	3	3	3	3	19	3	c	19	=	3	16	3	3			5
<u>level-2</u>	4	4	4	4	4	12	12	14	12	4	4	4	12	17	c	4	4	4	=	c	7	7	c	c	4
<u>level-3</u>	2	2	2	2	3	2	3	2	3	2	3	3	3	8	3	4	4	4	3	=	3	3		c	4
<u>leaf outerplanar</u>	4	4	4	4	4	10	10	10	10	4	4	4	10	8	10	4	4	4	8	16	=	c	c	c	4
<u>spread 1</u>	4	4	4	4	4	10	10	10	10	4	4	4	10	8	10	4	4	4	8	16	9	=	c	c	4
<u>spread 2</u>	3	3	3	3	3	3	3	3	3	3	3	3	3	8	3	4	4	4	3	16	3	3	=	c	4
<u>spread 3</u>	3	3	3	3	3	3	3	3	3	3	3	3	3	8	3	4	4	4	3	16	3	3		=	4
<u>time-consistent</u>	3	3	3	3	3	3	3	3	3	3	3	3	3	8	3	7	7	7	3	23	3	3			=

joint work with Maxime Morgado and Narges Tavassoli

Classes of phylogenetic networks: inclusions



Classes containing this network or not

Classes which contain this network (with direct proof)

- binary genetically stable: All reticulation vertices are stable (e for leaf 2, d for leaf 1, g for leaf 3, j for leaf 4) and all have at least one parent which is stable (f , parent of d , stable for leaf 2; b , parent of e , stable for leaves 1 and 2; a , parent of g , stable for leaves 1 and 2; h , parent of j , stable for leaf 4)
- binary nearly stable: All vertices except c and i (whose unique parent is in both cases stable) are stable.
- binary level-2: Easy to check.
- binary leaf outerplanar: Easy to check.
- binary FU-stable: Easy to check.

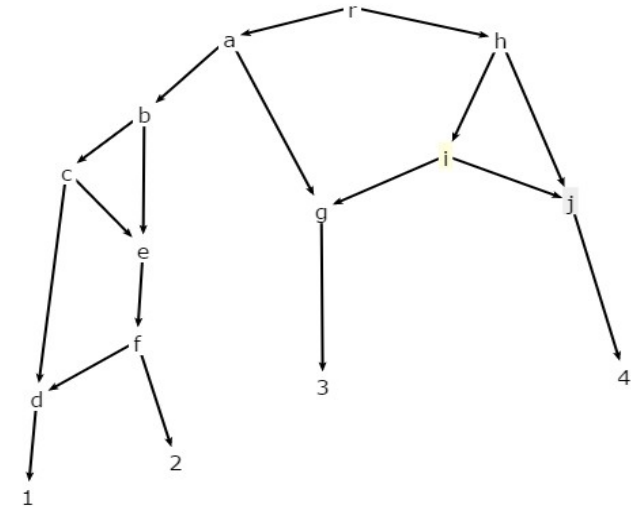
Classes which do not contain this network (with direct proof)

- binary nearly tree-child: Reticulation vertex g is stable but none of its parents a and i have the tree-path property.
- binary nested: Easy to check.
- binary time-consistent: The redundant arc from b to e makes it impossible to build a time-consistent labeling of the vertices.
- binary galled network: Reticulation arc (c,e) is a common arc of reticulation cycles from b to e and from c to d .
- binary distinct-cluster: Vertices h and i both have the same cluster: $\{3,4\}$
- binary 1-reticulated: Tree vertex b can reach 2 reticulation vertices by 2 directed internally vertex-disjoint paths: d and e .

Classes of phylogenetic networks: inclusions

level = maximum number of reticulation vertices among all bridgeless components in the network

cluster = set of leaves below a vertex



Classes containing this network or not

Classes which contain this network (with direct proof)

- binary genetically stable: All reticulation vertices are stable (e for leaf 2, d for leaf 1, g for leaf 3, j for leaf 4) and all have at least one parent which is stable (f , parent of d , stable for leaf 2; b , parent of e , stable for leaves 1 and 2; a , parent of g , stable for leaves 1 and 2; h , parent of j , stable for leaf 4)
- binary nearly stable: All vertices except c and i (whose unique parent is in both cases stable) are stable.
- binary level-2: Easy to check.
- binary leaf outerplanar: Easy to check.
- binary FU-stable: Easy to check.

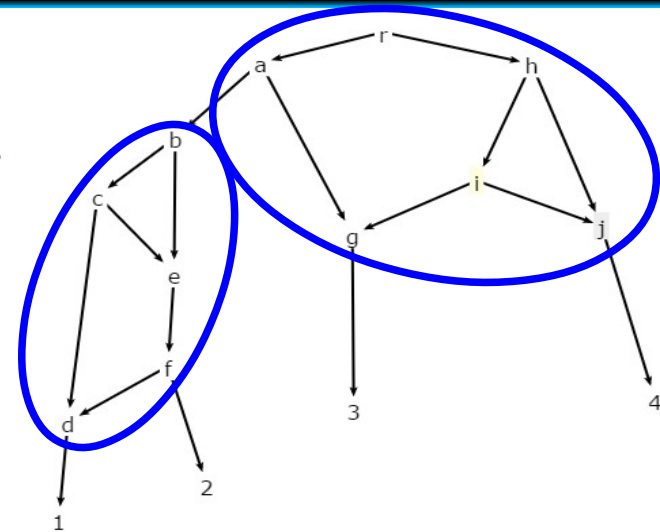
Classes which do not contain this network (with direct proof)

- binary nearly tree-child: Reticulation vertex g is stable but none of its parents a and i have the tree-path property.
- binary nested: Easy to check.
- binary time-consistent: The redundant arc from b to e makes it impossible to build a time-consistent labeling of the vertices.
- binary galled network: Reticulation arc (c,e) is a common arc of reticulation cycles from b to e and from c to d .
- binary distinct-cluster: Vertices h and i both have the same cluster: $\{3,4\}$
- binary 1-reticulated: Tree vertex b can reach 2 reticulation vertices by 2 directed internally vertex-disjoint paths: d and e .

Classes of phylogenetic networks: inclusions

level = maximum number of reticulation vertices among all **bridgeless components** in the network

cluster = set of leaves below a vertex



Classes containing this network or not

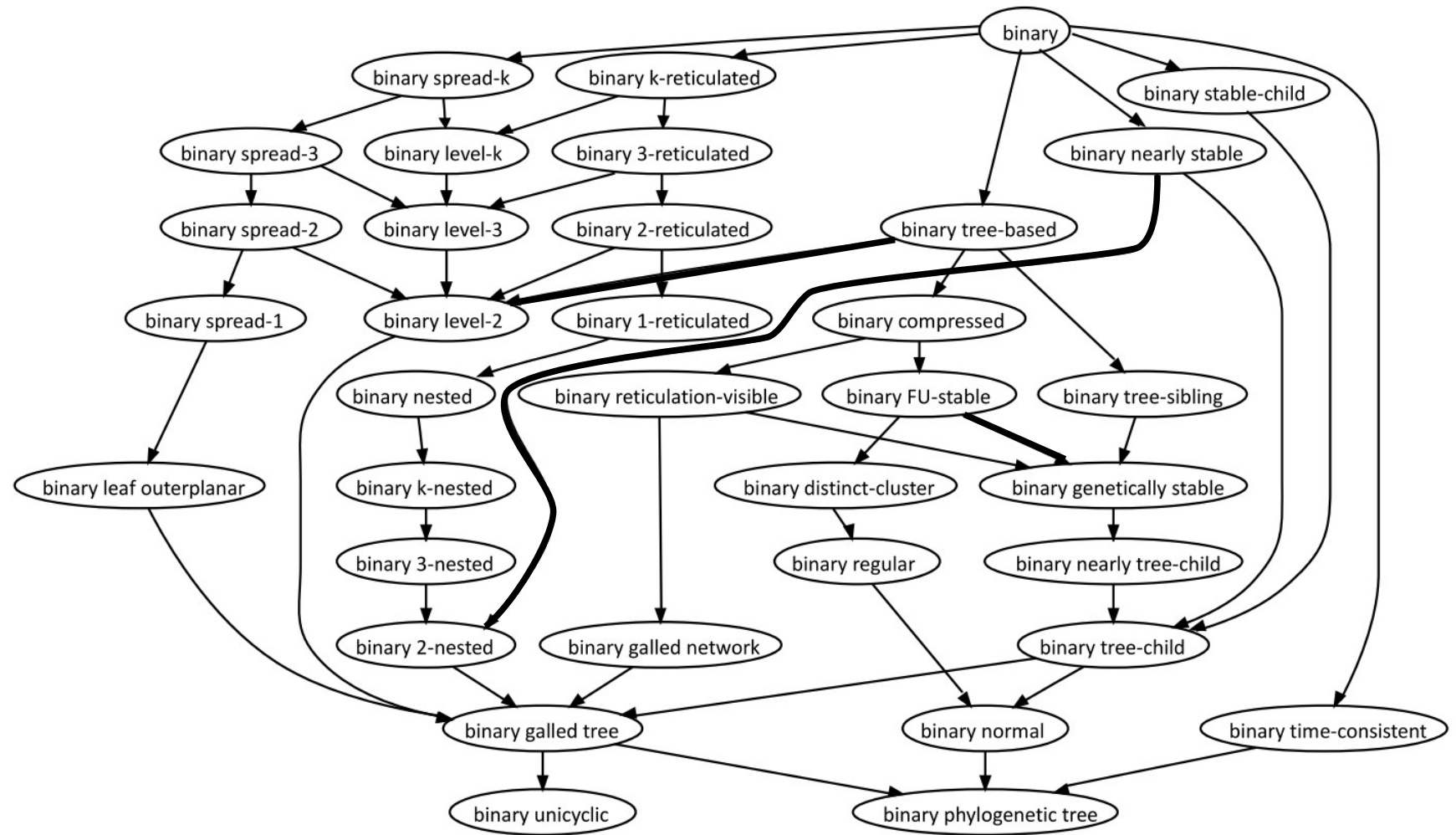
Classes which contain this network (with direct proof)

- binary genetically stable: All reticulation vertices are stable (e for leaf 2, d for leaf 1, g for leaf 3, j for leaf 4) and all have at least one parent which is stable (f , parent of d , stable for leaf 2; b , parent of e , stable for leaves 1 and 2; a , parent of g , stable for leaves 1 and 2; h , parent of j , stable for leaf 4)
- binary nearly stable: All vertices except c and i (whose unique parent is in both cases stable) are stable.
- binary level-2: Easy to check.
- binary leaf outerplanar: Easy to check.
- binary FU-stable: Easy to check.

Classes which do not contain this network (with direct proof)

- binary nearly tree-child: Reticulation vertex g is stable but none of its parents a and i have the tree-path property.
- binary nested: Easy to check.
- binary time-consistent: The redundant arc from b to e makes it impossible to build a time-consistent labeling of the vertices.
- binary galled network: Reticulation arc (c,e) is a common arc of reticulation cycles from b to e and from c to d .
- binary distinct-cluster: Vertices h and i both have the same cluster: $\{3,4\}$
- binary 1-reticulated: Tree vertex b can reach 2 reticulation vertices by 2 directed internally vertex-disjoint paths: d and e .

Classes of phylogenetic networks: new inclusions



Classes of phylogenetic networks

Problems and properties studied on these classes

Problems

- **Tree Containment:** Does the input network contain the input tree on the same set of leaves?
- **Cluster Containment:** Does the input network contain the input subset of leaves as a softwired cluster?
- **Phylogenetic Network Isomorphism:** Are the two input phylogenetic networks isomorphic?

Properties

- **Upper bound on the number of vertices:** The number of vertices is bounded by the number of leaves.
- **Unbounded number of vertices:** The number of vertices is not bounded by the number of leaves.
- **Completeness for reconstruction from trees:** There exists a network in this class which displays all binary rooted phylogenetic trees on n leaves.

Database content

73 classes of phylogenetic networks including 35 classes of binary phylogenetic networks (defined in a total of 20 bibliographic references), 51 inclusion relationships proved directly between classes (including some found in a total of 9 bibliographic references), 21 networks (65 memberships to a class, 53 non-memberships to a class), 3 problems considered, 3 properties considered, 33 theorems proved directly (including some found in a total of 15 bibliographic references) including 23 positive results (which can be extended to subclasses) and 10 negative results (which can be extended to superclasses).

Classes of phylogenetic networks

Problems and properties studied on these classes

Problems

- Tree Containment: Does the input network contain the input tree on the same set of leaves?
- Cluster Containment: Does the input network contain the input subset of leaves as a softwired cluster?
- Phylogenetic Network Isomorphism: Are the two input phylogenetic networks isomorphic?

Properties

- Upper bound on the number of vertices: The number of vertices is bounded by the number of leaves.
- Unbounded number of vertices: The number of vertices is not bounded by the number of leaves.
- Completeness for reconstruction from trees: There exists a network in this class which displays all binary rooted phylogenetic trees on n leaves.

Database content

73 classes of phylogenetic networks including 35 classes of binary phylogenetic networks (defined in a total of 20 bibliographic references), 51 inclusion relationships proved directly between classes (including some found in a total of 9 bibliographic references), 21 networks (65 memberships to a class, 53 non-memberships to a class), 3 problems considered, 3 properties considered, 33 theorems proved directly (including some found in a total of 15 bibliographic references) including 23 positive results (which can be extended to subclasses) and 10 negative results (which can be extended to superclasses).

Polynomial time algorithm available on class A \rightarrow works on **subclass B**
NP-completeness on class B \rightarrow NP-completeness on **superclass A**
(similar to ISGCI)

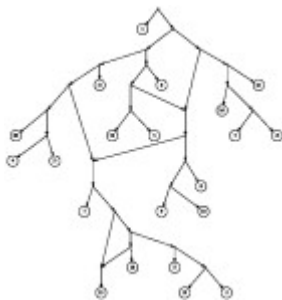
Outline

- Phylogenetic networks
- Classes of phylogenetic networks
- The Tree Containment Problem

Phylogenetic network reconstruction

espèce 1 : AATTGCAG TAGCCCAAAAT
espèce 2 : ACCTGCAG TAGACCAAT
espèce 3 : GCTTGCCG TAGACAAGAAT
espèce 4 : ATTTGCAG AAGACCAAAT
espèce 5 : TAGACAAGAAT
espèce 6 : ACTTGCAG TAGCACAAAAT
espèce 7 : ACCTGGTG TAAAAT

G1 G2



{gene sequences}

distance methods

*Bandelt & Dress 1992 - Legendre & Makarenkov 2000 -
Bryant & Moulton 2002 - Chan, Jansson, Lam & Yiu 2006*

parsimony methods

*Hein 1990 - Kececioglu & Gusfield 1994 - Jin, Nakhleh,
Snir, Tuller 2009 - Park, Jin & Nakhleh 2010 - Kannan &
Wheeler, 2012*

likelihood methods

*Snir & Tuller 2009 - Jin, Nakhleh, Snir, Tuller 2009 -
Velasco & Sober 2009 - Meng & Kubatko 2009*

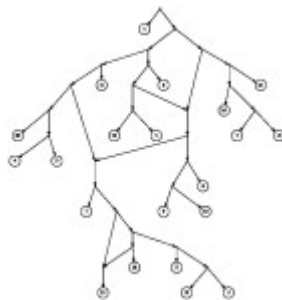
network *N*

Phylogenetic network reconstruction

**Problem: methods are usually slow,
especially with rapidly increasing sequence length.**

espèce 1 : AATTGCAG TAGCCCAAAAT
espèce 2 : ACCTGCAG TAGACCAAT
espèce 3 : GCTTGCCG TAGACAAGAAT
espèce 4 : ATTTGCAG AAGACCAAAT
espèce 5 : TAGACAAGAAT
espèce 6 : ACTTGCAG TAGCACAAAAT
espèce 7 : ACCTGGTG TAAAAT

G1 **G2**



{gene sequences}

distance methods

*Bandelt & Dress 1992 - Legendre & Makarenkov 2000 -
Bryant & Moulton 2002 - Chan, Jansson, Lam & Yiu 2006*

parsimony methods

*Hein 1990 - Kececioglu & Gusfield 1994 - Jin, Nakhleh,
Snir, Tuller 2009 - Park, Jin & Nakhleh 2010 - Kannan &
Wheeler, 2012*

likelihood methods

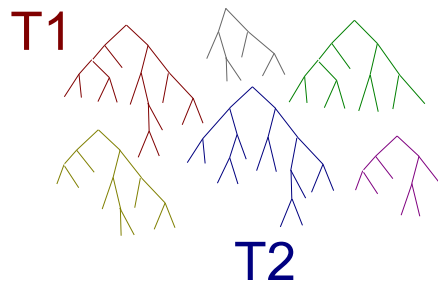
*Snir & Tuller 2009 - Jin, Nakhleh, Snir, Tuller 2009 -
Velasco & Sober 2009 - Meng & Kubatko 2009*

network *N*

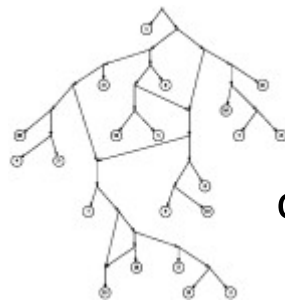
Phylogenetic network reconstruction

espèce 1 : AATTGCAG TAGCCCAAAAT
espèce 2 : ACCTGCAG TAGACCAAT
espèce 3 : GCTTGCCG TAGACAAGAAT
espèce 4 : ATTTGCAG AAGACCAAT
espèce 5 : TAGACAAGAAT
espèce 6 : ACTTGCAG TAGCACAAAAT
espèce 7 : ACCTGGTG TAAAAT

G1 G2



explicit network



{gene sequences}

Reconstruction of a tree for each gene present in several species

Guindon & Gascuel, SB, 2003

{trees}

HOGENOM Database
Dufayard, Duret, Penel, Gouy,
Rechenmann & Perrière, BioInf, 2005



Tree reconciliation or consensus

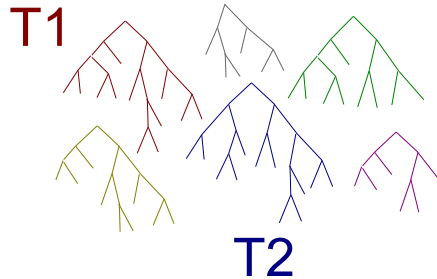
optimal super-network N:

- contains the input trees
- has the smallest number of reticulations

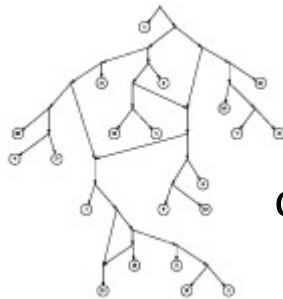
Phylogenetic network reconstruction

espèce 1 : AATTGCAG TAGCCCAAAAT
espèce 2 : ACCTGCAG TAGACCAAT
espèce 3 : GCTTGCCG TAGACAAGAAT
espèce 4 : ATTTGCAG AAGACCAAAAT
espèce 5 : TAGACAAGAAT
espèce 6 : ACTTGCAG TAGCACAAAAT
espèce 7 : ACCTGGTG TAAAAT

G1 G2



explicit network



{gene sequences}

Reconstruction of a tree for each gene present in several species

Guindon & Gascuel, SB, 2003

{trees}

HOGENOM Database



Dufayard, Duret, Penel, Gouy, Rechenmann & Perrière, BioInf, 2005

1470 species, >290 000 trees

Tree reconciliation or consensus

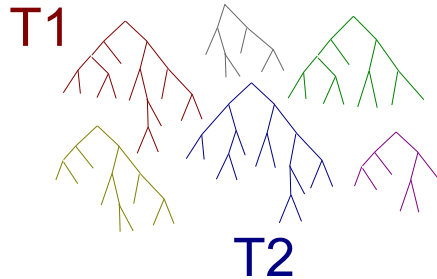
optimal super-network N:

- contains the input trees
- has the smallest number of reticulations

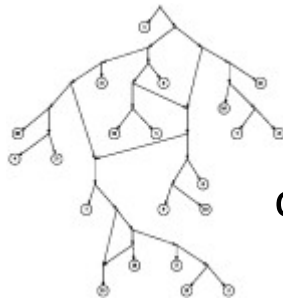
Phylogenetic network reconstruction

espèce 1 : AATTGCAG TAGCCCAAAAT
espèce 2 : ACCTGCAG TAGACCAAT
espèce 3 : GCTTGCCG TAGACAAGAAT
espèce 4 : ATTTGCAG AAGACCAAAT
espèce 5 : TAGACAAGAAT
espèce 6 : ACTTGCAG TAGCACAAAAT
espèce 7 : ACCTGGTG TAAAAT

G1 G2



explicit network



{gene sequences}

Reconstruction of a tree for each gene present in several species

Guindon & Gascuel, SB, 2003

{trees}

HOGENOM Database



Dufayard, Duret, Penel, Gouy, Rechenmann & Perrière, BioInf, 2005

1470 species, >290 000 trees

Tree reconciliation or consensus

Tree Containment Problem

optimal super-network N:

- contains the input trees
- has the smallest number of reticulations

The Tree Containment Problem (T.C.P.)

Input: A binary phylogenetic network N and a tree T over the same set of taxa.

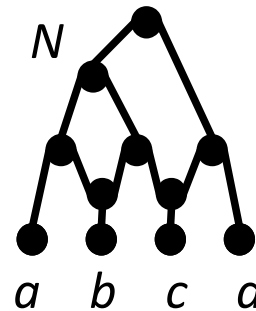
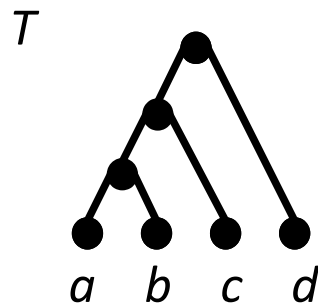
Question: Does N display T ?

The Tree Containment Problem (T.C.P.)

Input: A binary phylogenetic network N and a tree T over the same set of taxa.

Question: Does N display T ?

→ Can we remove one incoming arc, for each vertex with >1 parent in N , such that the obtained tree is equivalent to T ?

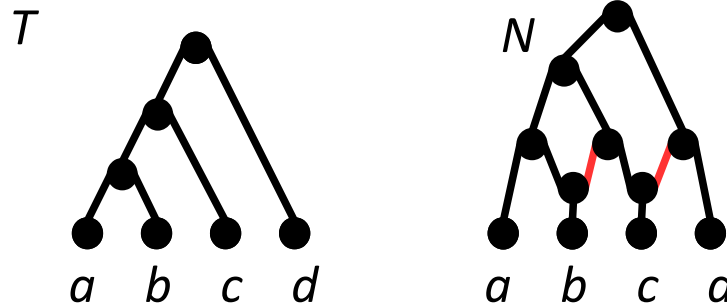


The Tree Containment Problem (T.C.P.)

Input: A binary phylogenetic network N and a tree T over the same set of taxa.

Question: Does N display T ?

→ Can we **remove one incoming arc**, for each vertex with >1 parent in N , such that the obtained tree is equivalent to T ?



The Tree Containment Problem (T.C.P.)

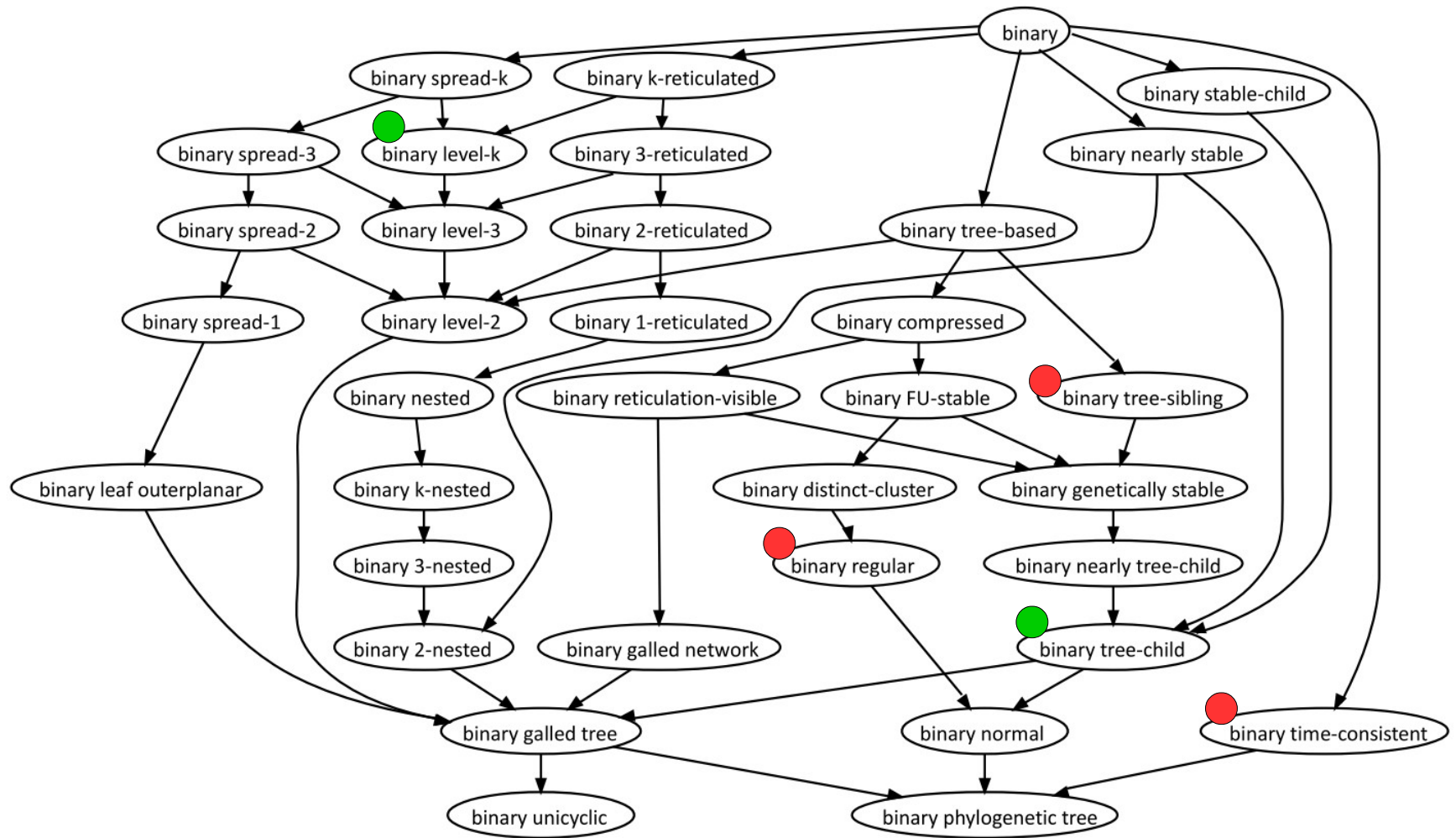
Input: A binary phylogenetic network N and a tree T over the same set of taxa.

Question: Does N display T ?

- **NP-complete** in general (Kanj, Nakhleh, Than & Xia, 2008)
- **NP-complete** for tree-sibling, time-consistent, regular networks (Iersel, Semple & Steel, 2010)
- **Polynomial-time solvable** for normal networks, for binary tree-child networks, and for level- k networks (Iersel, Semple & Steel, 2010)

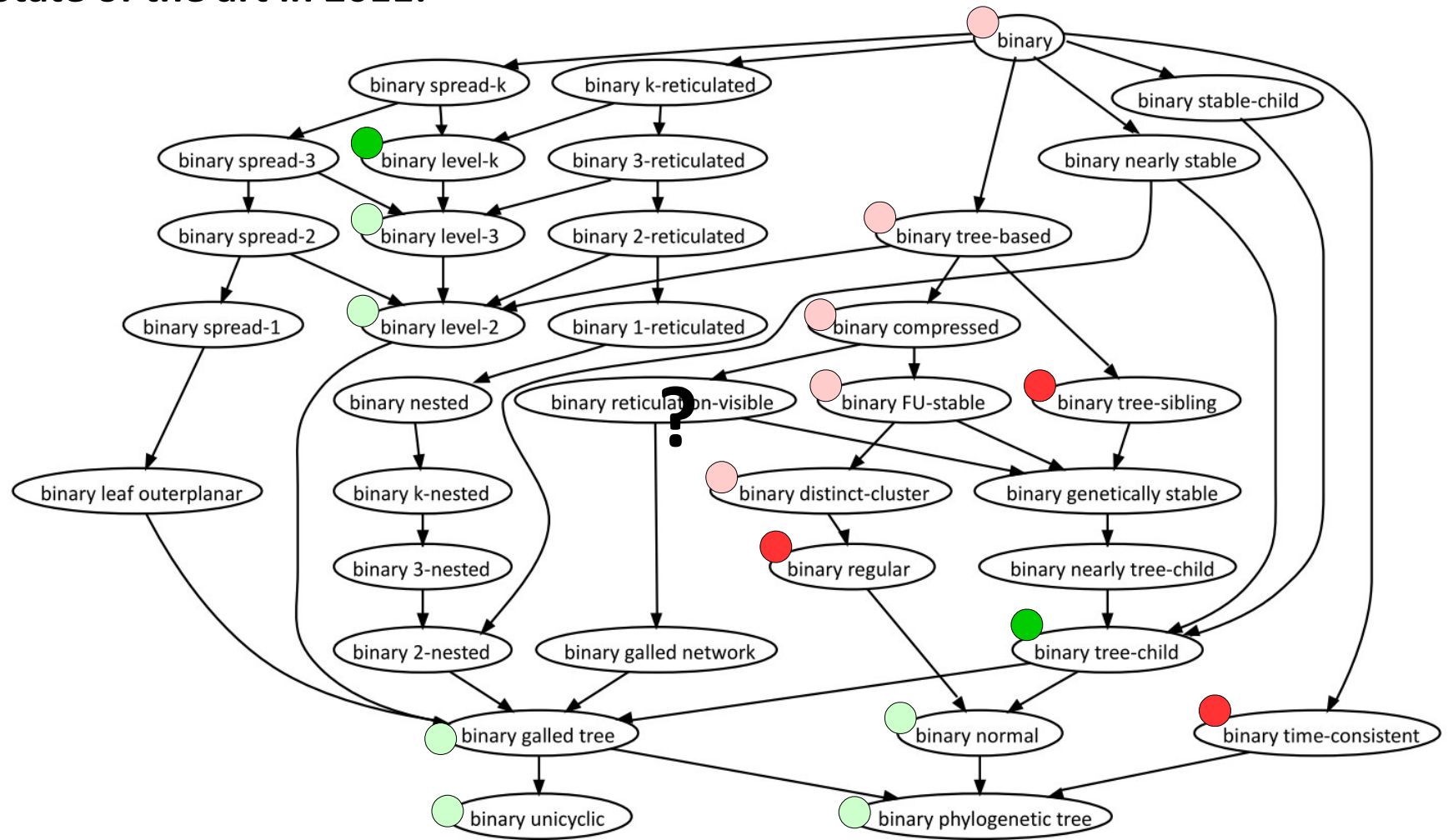
Classes of phylogenetic networks and the T.C.P.

State of the art in 2011:



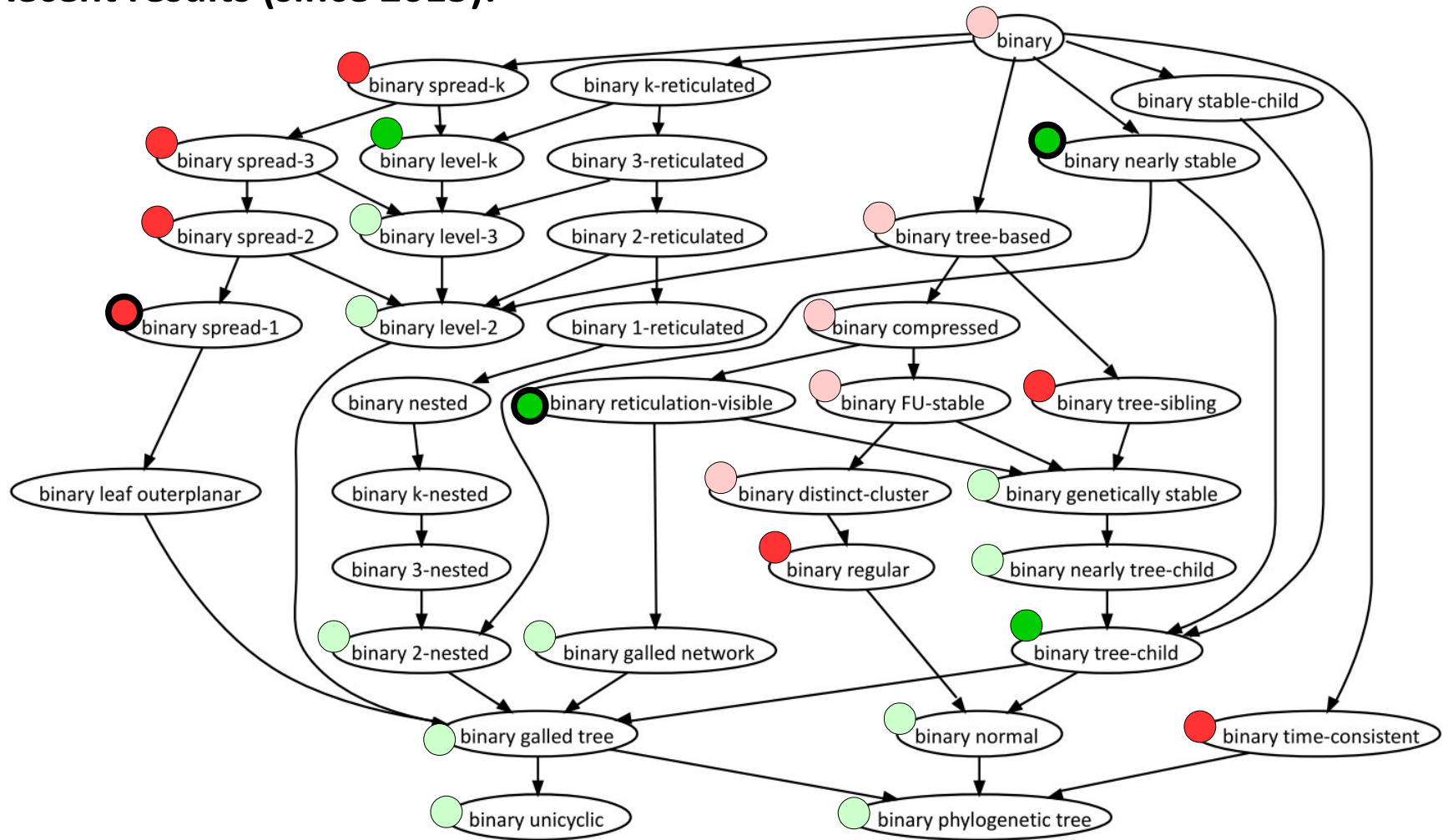
Classes of phylogenetic networks and the T.C.P.

State of the art in 2011:



Classes of phylogenetic networks and the T.C.P.

Recent results (since 2015):



Fête de la Science at UPEM

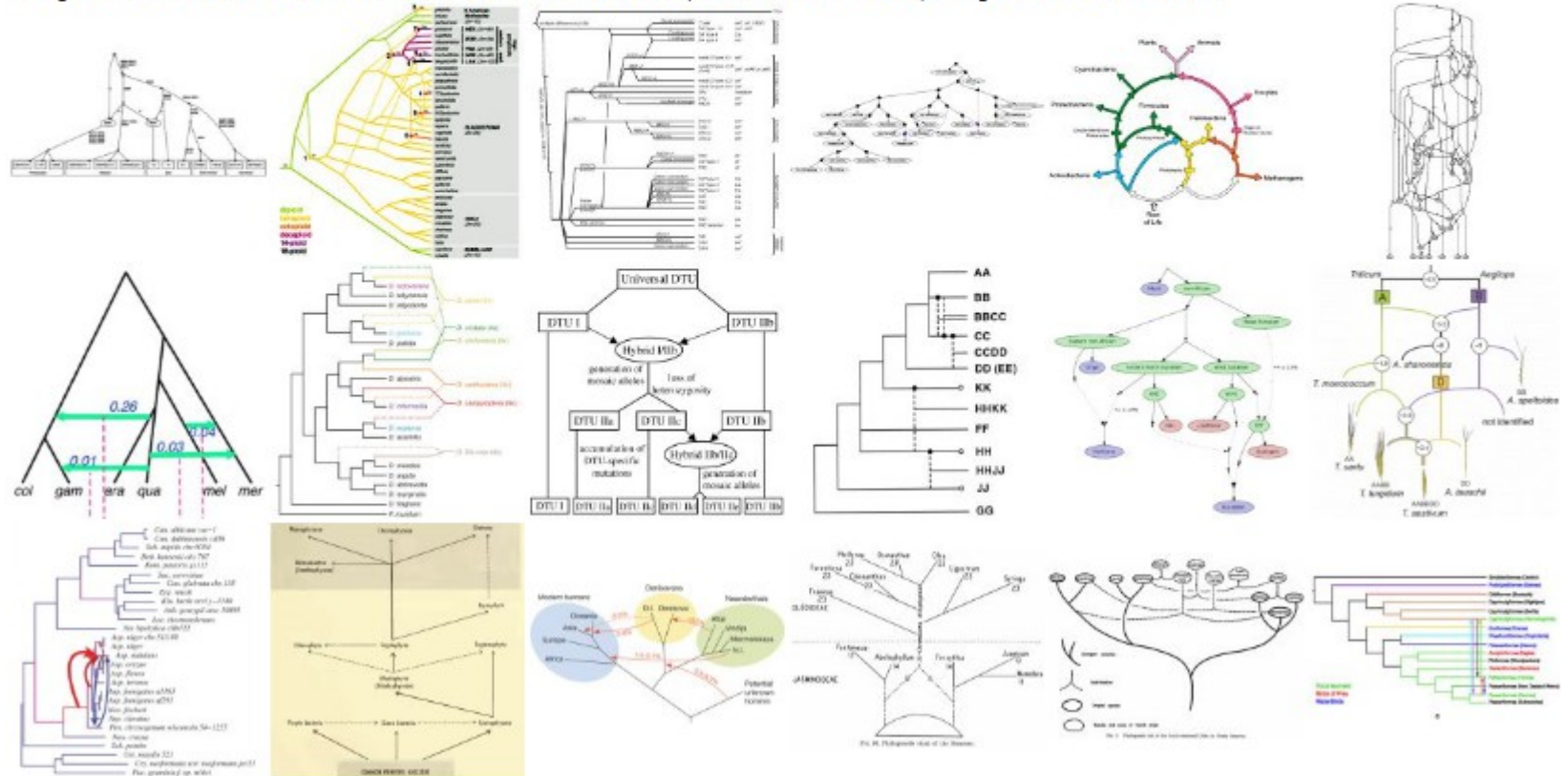


Photos : Campus numérique de l'UPEM

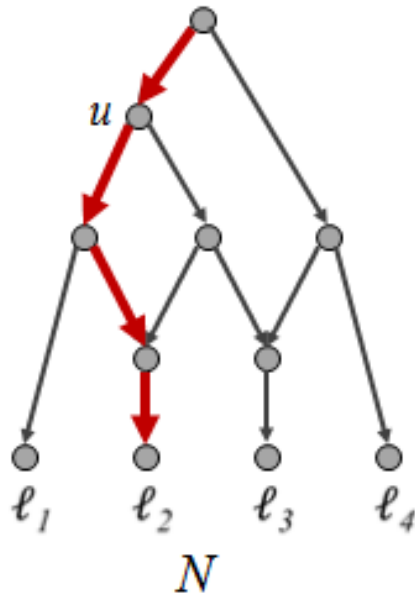
Science académie at UPEM

15. Figure 5 of Sessa EB, Zimmer EA, Givnish TJ (2012). [Unraveling reticulate evolution in North American Dryopteris \(Dryopteridaceae\)](#). *BMC Evolutionary Biology* 12:104 → [load this net](#)
16. Figure 4 of Marcussen, T., Jakobsen, K.S., Danihelka, J., Ballard, H. E., Blaxland K., Brysting, A. K., Oxelman, B. (2012). [Inferring Species Networks from Gene Trees in High-Polyploid Nor](#)
17. Figure 3 of Reich, D., Patterson, N., Kircher, M., Delfin, F., Nandineni, M. R., Pugach, I., ... & Saitou, N. (2011). [Denisova admixture and the first modern human dispersals into Southeast](#)
18. Figure 3 of Charlton, N. D., Carbone, I., Tavantzis, S. M., Cubeta, M. A. (2008). [Phylogenetic relatedness of the M2 double-stranded RNA in *Rhizoctonia fungi*](#). *Mycologia* 100(4):555-56
19. Figure 5 of Westenberger, S. J., Barnabé, C., Campbell, D. A., Sturm, N. R. (2005). [Two Hybridization Events Define the Population Structure of *Trypanosoma cruzi*](#). *Genetics* 171(2):527.
20. Figure 4 of Ge, S., To, S., Lu, B. R. (1999). [Phylogeny of rice genomes with emphasis on origins of allotetraploid species](#). *PNAS* 96(25):14400-14405 → [load this network!](#)
21. Figure 56 of Grant, V. (1971). *Plant Speciation*, Columbia University Press → [load this network!](#)
22. Figure 14.1 of Alston, R. E., Turner, B. L. (1963). *Biochemical systematics*, Englewood Cliffs, N.J., Prentice-Hall → [load this network!](#)
23. Figure 5 of Grant, V. (1953). [The Role of Hybridization in the Evolution of the Leafy-Stemmed *Gilias*](#), *Brittonia* 5(4):337-367 → [load this network!](#)
24. Figure 96 of Taylor, H. (1945). [Cyto-Taxonomy and Phylogeny of the Oleaceae](#), *Evolution* 7(1):51-64 → [load this network!](#)

The figures extracted from the article cited above are available below: click directly on them to load the corresponding network in the form above!

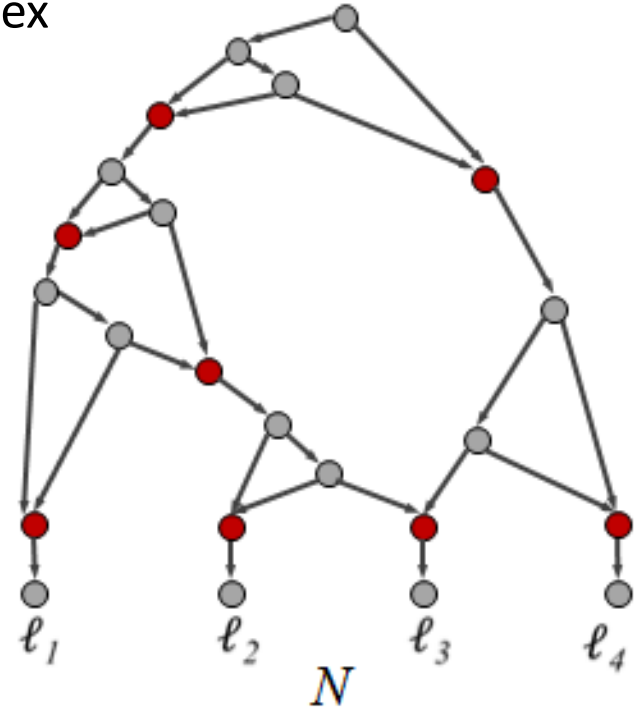


Reticulation-visible and nearly-stable networks



A vertex u is **stable** if there exists a leaf l such that all paths from the root to l go through u .

A phylogenetic network is **reticulation-visible** if every reticulation vertex is stable.



A phylogenetic network is **nearly-stable** if for each vertex, either it is stable or its parents are.

Strategy to get a quadratic time algorithm for T.C.P.

Given N , a phylogenetic network with n leaves and the input tree T of the T.C.P.

Theorem 1: If N is reticulation-visible then: $\#\{\text{reticulation vertices of } N\} \leq 4(n-1)$
 $\#\{\text{vertices of } N\} \leq 9n$

Theorem 2: If N is nearly-stable then $\#\{\text{reticulation vertices of } N\} \leq 12(n-1)$

Theorem 3: Considering a longest path in N (nearly stable), it is possible, in constant time:

- either to realize that T is not contained in N
- or to build a network N' with less arcs than N such that T contained in N if and only if T contained in N'

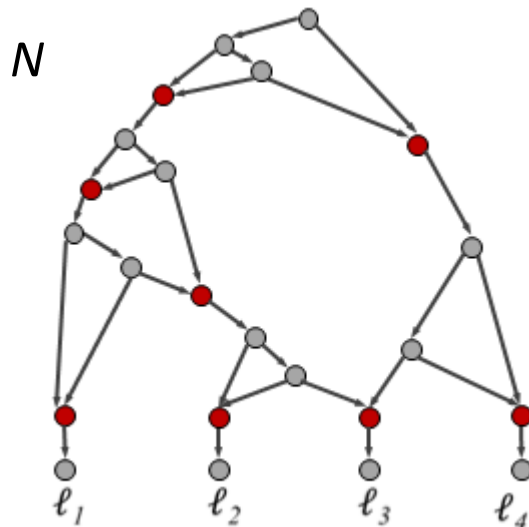
Number of reticulations of a reticulation-visible network

Decompose N into $2n-2$ paths:

- remove one reticulation arc per reticulation, ensuring we get no « dummy leaf », to get a tree T with n leaves
- summarize T into a rooted binary tree T' with n leaves... and $2n-2$ arcs

We can prove (technical) that:

each path contains at most 2 reticulation vertices



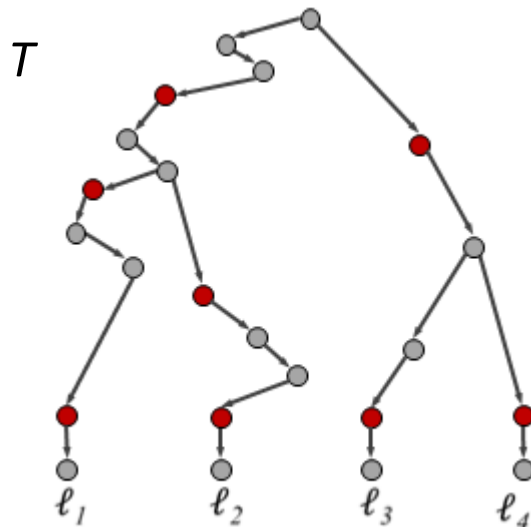
Number of reticulations of a reticulation-visible network

Decompose N into $2n-2$ paths:

- remove one reticulation arc per reticulation, ensuring we get no « dummy leaf », to get a tree T with n leaves
- summarize T into a rooted binary tree T' with n leaves... and $2n-2$ arcs

We can prove (technical) that:

each path contains at most 2 reticulation vertices



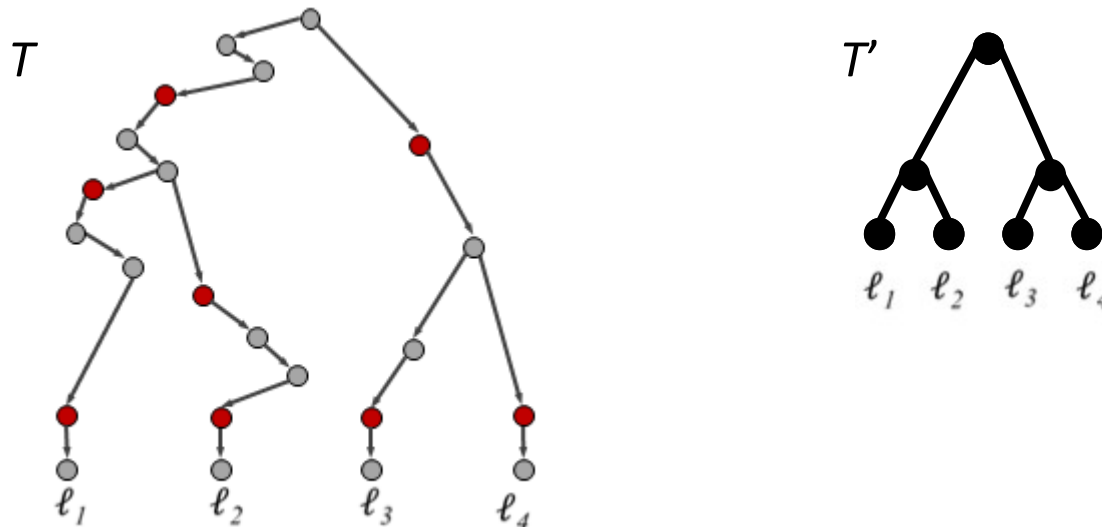
Number of reticulations of a reticulation-visible network

Decompose N into $2n-2$ paths:

- remove one reticulation arc per reticulation, ensuring we get no « dummy leaf », to get a tree T with n leaves
- summarize T into a rooted binary tree T' with n leaves... and $2n-2$ arcs

We can prove (technical) that:

each path contains at most 2 reticulation vertices



Number of reticulations of a reticulation-visible network

Decompose N into $2n-2$ paths:

- remove one reticulation arc per reticulation, ensuring we get no « dummy leaf », to get a tree T with n leaves
- summarize T into a rooted binary tree T' with n leaves... and $2n-2$ arcs

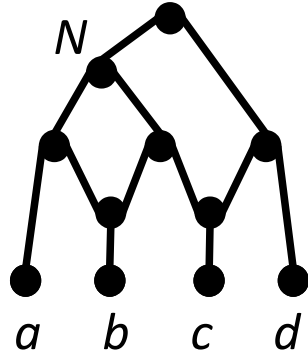
We can prove (technical) that:

each path contains at most 2 reticulation vertices

→ N contains at most $4(n-1)$ reticulation vertices

Number of reticulations of a reticulation-visible network

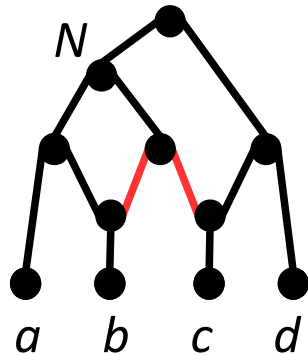
« Dummy leaves »?



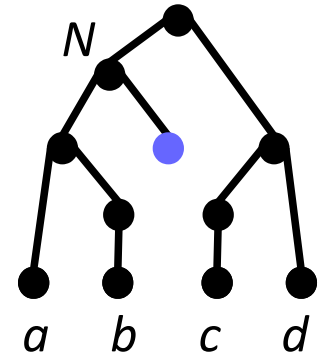
Deleting reticulation arcs can create « **dummy leaves** »

Number of reticulations of a reticulation-visible network

« Dummy leaves »?



Deleting **reticulation arcs** can create « **dummy leaves** »



Number of reticulations of a reticulation-visible network

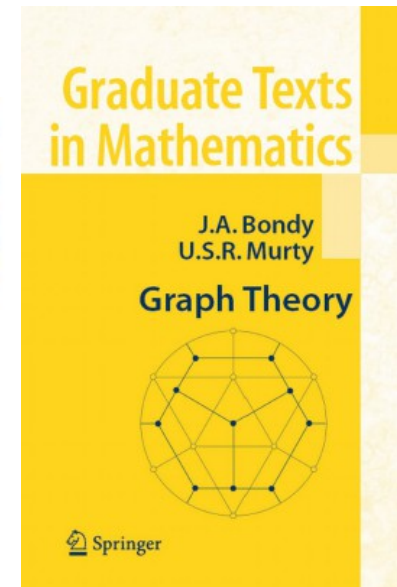
Possible to avoid creating « dummy leaves »?

Number of reticulations of a reticulation-visible network

Possible to avoid creating « dummy leaves »?

16.2 Matchings in Bipartite Graphs 423

16.2.13 Let $G := G[X, Y]$ be a bipartite graph such that $d(x) \geq 1$ for all $x \in X$ and $d(x) \geq d(y)$ for all $xy \in E$, where $x \in X$ and $y \in Y$. Show that G has a matching covering every vertex of X .
(N. ALON)

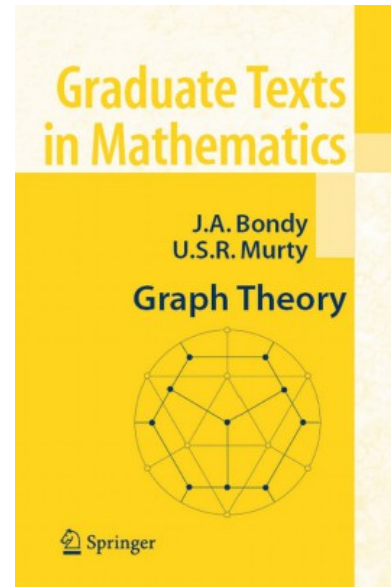


Number of reticulations of a reticulation-visible network

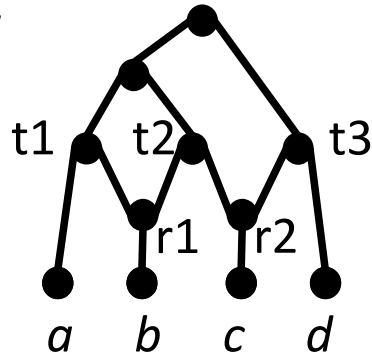
Possible to avoid creating « dummy leaves »?

16.2 Matchings in Bipartite Graphs 423

16.2.13 Let $G := G[X, Y]$ be a bipartite graph such that $d(x) \geq 1$ for all $x \in X$ and $d(x) \geq d(y)$ for all $xy \in E$, where $x \in X$ and $y \in Y$. Show that G has a matching covering every vertex of X . (N. ALON)

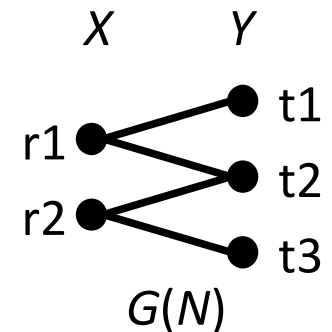


Given N



Build $G(N)$, bipartite graph such that:

- X = reticulation vertices of N
 → all vertices in X have degree 2
- Y = tree vertices of N with at least one reticulation child
 → all vertices in Y have degree 1 or 2
- edge between x and y iff x is a child of y

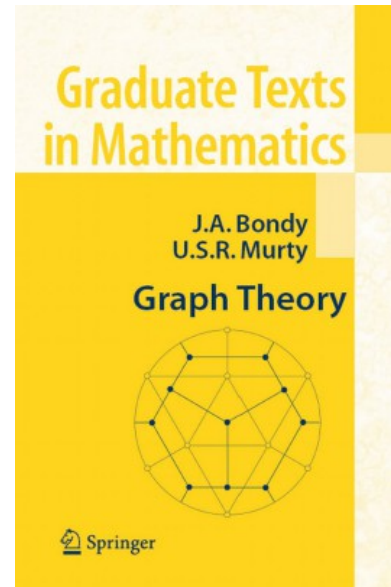


Number of reticulations of a reticulation-visible network

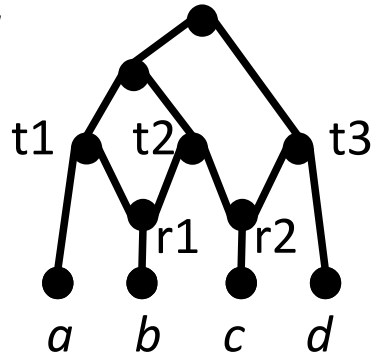
Possible to avoid creating « dummy leaves »?

16.2 Matchings in Bipartite Graphs 423

16.2.13 Let $G := G[X, Y]$ be a bipartite graph such that $d(x) \geq 1$ for all $x \in X$ and $d(x) \geq d(y)$ for all $xy \in E$, where $x \in X$ and $y \in Y$. Show that G has a matching covering every vertex of X . (N. ALON)



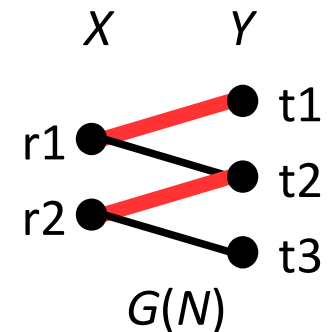
Given N



Build $G(N)$, bipartite graph such that:

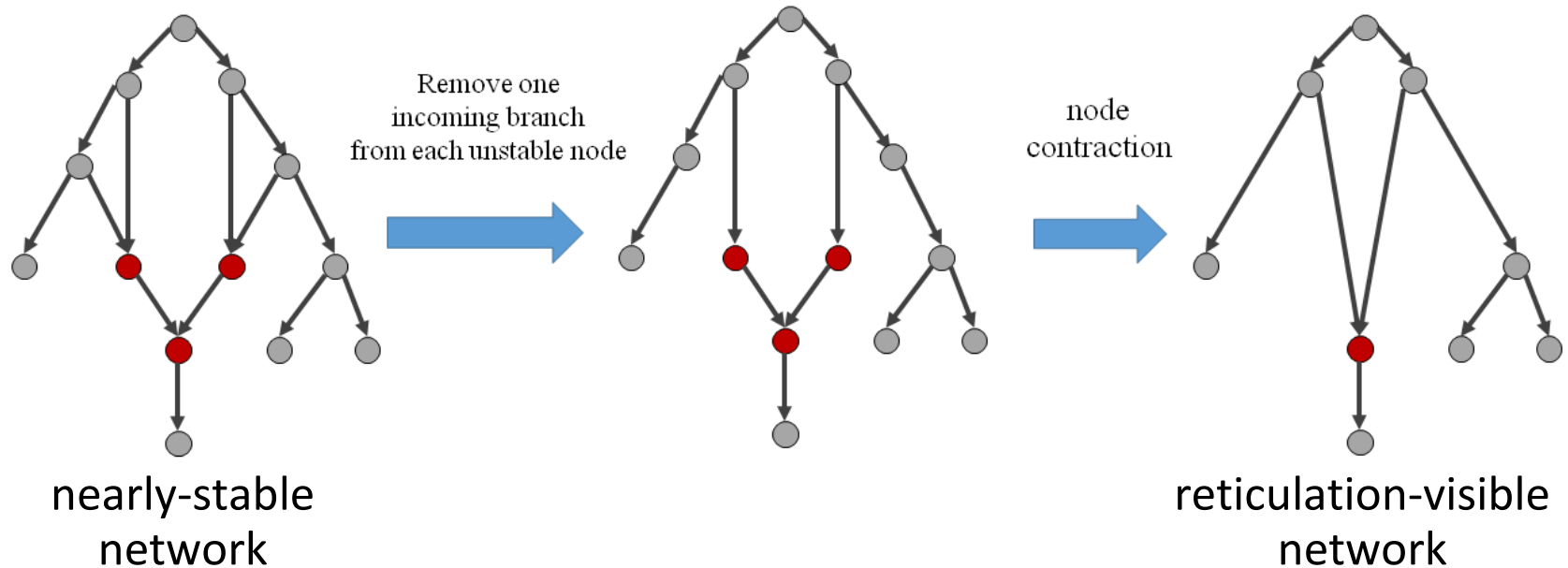
- X = reticulation vertices of N
 → all vertices in X have degree 2
- Y = tree vertices of N with at least one reticulation child
 → all vertices in Y have degree 1 or 2
- edge between x and y iff x is a child of y

→ **matching covering every vertex of X** → **edges to remove from N**



Number of reticulations of a nearly-stable network

Reduce nearly-stable networks to stable networks



Claim: $\#unstableReticulations \leq 2 \#stableReticulations$

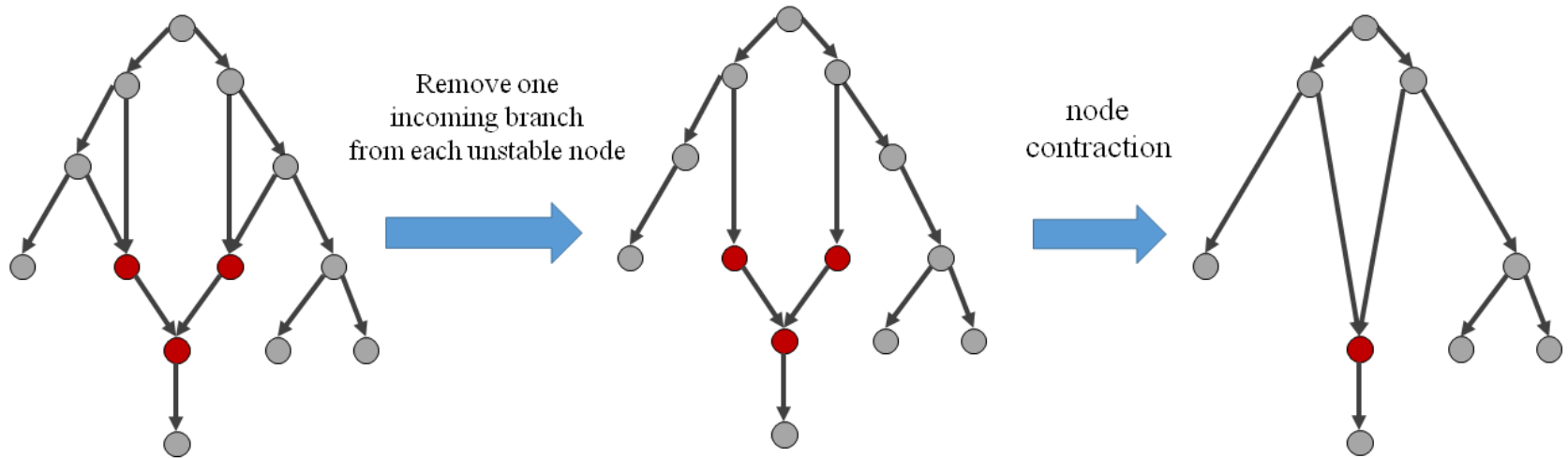
Proof:

Each unstable reticulation must have a stable reticulation child.

At most two unstable reticulations may have the same reticulation child.

Number of reticulations of a nearly-stable network

Reduce nearly-stable networks to stable networks



$\#unstableReticulations \leq 2 \#stableReticulations$

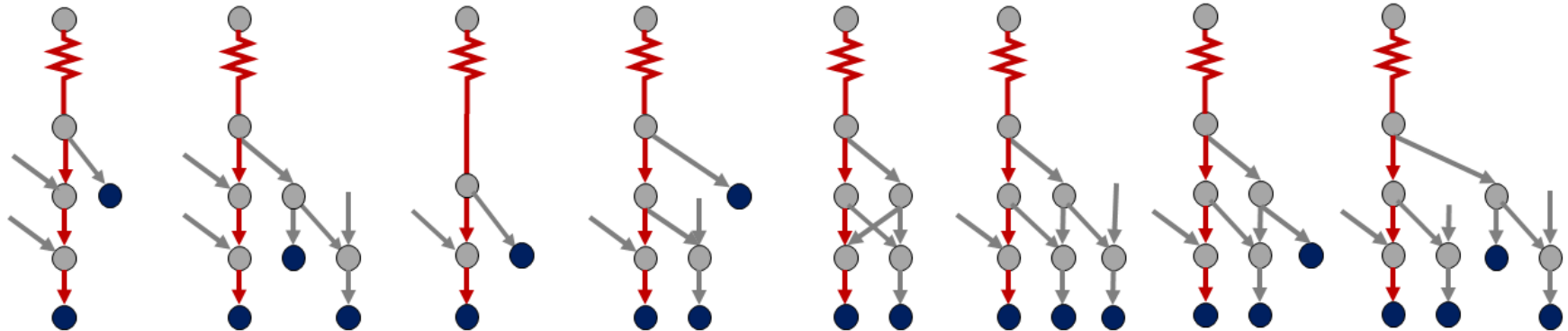
$\#stableReticulations \leq 4(n-1)$

$\rightarrow \#reticulations \leq 12(n-1)$

Deleting reticulation arcs to simplify the question

Simplify N by removing an arc near the end of a **longest path P** .

Case analysis (8 cases):



Algorithm for nearly-stable networks:

Repeat:

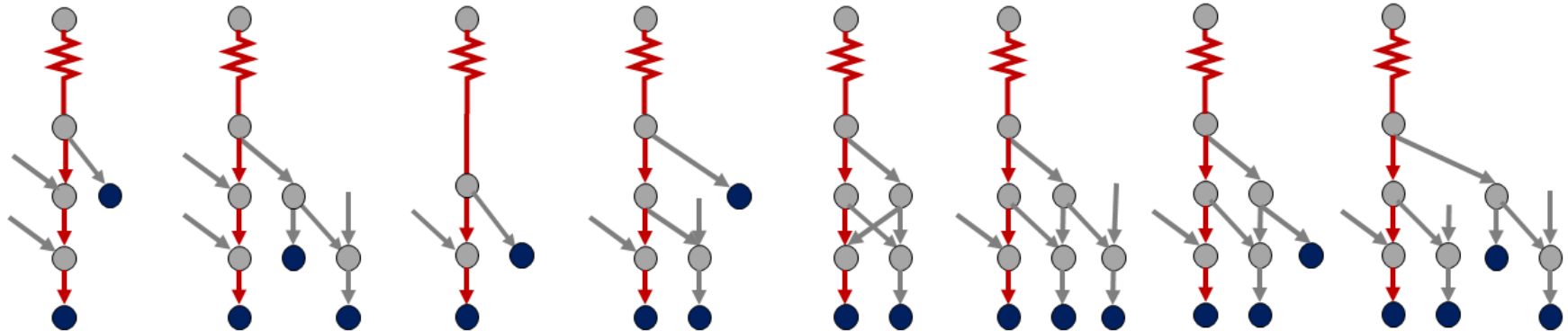
- Compute a longest path P in N
- Simplify N by considering the subnetwork at the end of P according to the 8 cases above (arc removal + contraction)
- Replace a «cherry» (2 leaves + their common parent) appearing in both N and T by a new leaf

Finally check that the obtained network is identical to T

Deleting reticulation arcs to simplify the question

Simplify N by removing an arc near the end of a **longest path P** .

Case analysis (8 cases):



Algorithm for nearly-stable networks:

Repeat:

- Compute a longest path P in N
- Simplify N by considering the subnetwork at the end of P according to the 8 cases above (arc removal + contraction)
- Replace a «cherry» (2 leaves + their common parent) appearing in both N and T by a new leaf

Finally check that the obtained network is identical to T

$O(n)$ times:

→ $O(n)$

→ $O(1)$

→ $O(1)$

→ $O(n)$

Improved algorithms

$O(n \log n)$ -time algorithm for nearly-stable networks

Fakcharoenphol, Kumpijit & Putwattana, JCSSE 2015

Linear time algorithm for nearly-stable networks

Gambette, Gunawan, Labarre, Vialette & Zhang, DAM 2017

Cubic (and then linear) time algorithm for reticulation-visible networks

Gunawan, DasGupta & Zhang, RECOMB 2016

Bordewich & Semple, Advances in Applied Mathematics, 2016

Gunawan, arXiv, 2017

#reticulations $\leq 3(n-1)$ for reticulation-visible networks

Bordewich & Semple, Advances in Applied Mathematics, 2016

Linear time algorithm for a superclass of reticulation-visible networks

Weller, arXiv, 2017

Perspectives...

- Provide a practical algorithmic toolbox using network classes
 - <http://phylnet.info/tools/>
- Combine the combinatorial and statistical approaches to reconstruct phylogenetic networks
 - CNRS PICS project with L. van Iersel, S. Kelk, F. Pardi & C. Scornavacca
- Develop interactions with related fields:
 - host/parasite relationships
 - population genetics
 - stemmatology