ISCBacademy webinar – September 30, 2020

RAxML-NG:
a fast, scalable and user-friendly tool for maximum likelihood phylogenetic inference

Alexandros Stamatakis and Alexey Kozlov

HITS

Computational Molecular Evolution Group,
Heidelberg Institute for Theoretical Studies

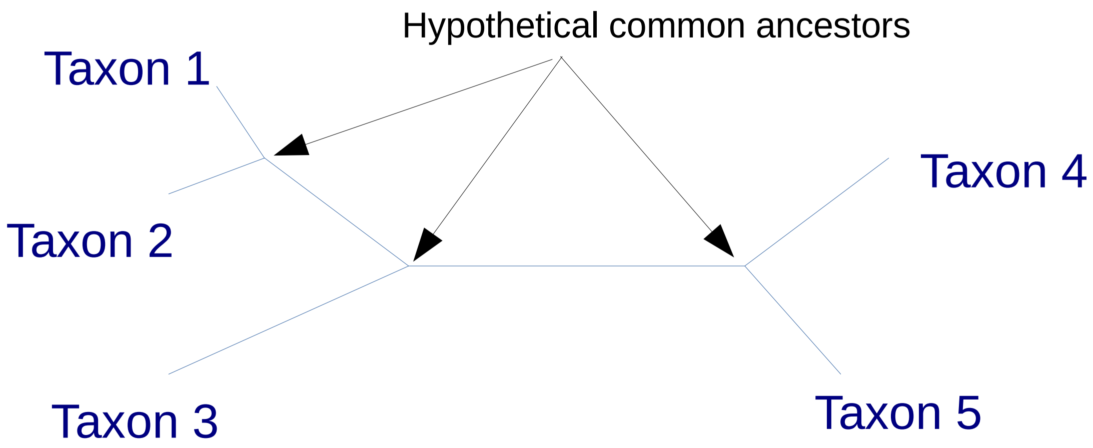
Institute for Theoretical Informatics,
Karlsruhe Institute of Technology

KIT
Karlsruhe Institute of Technology

Outline

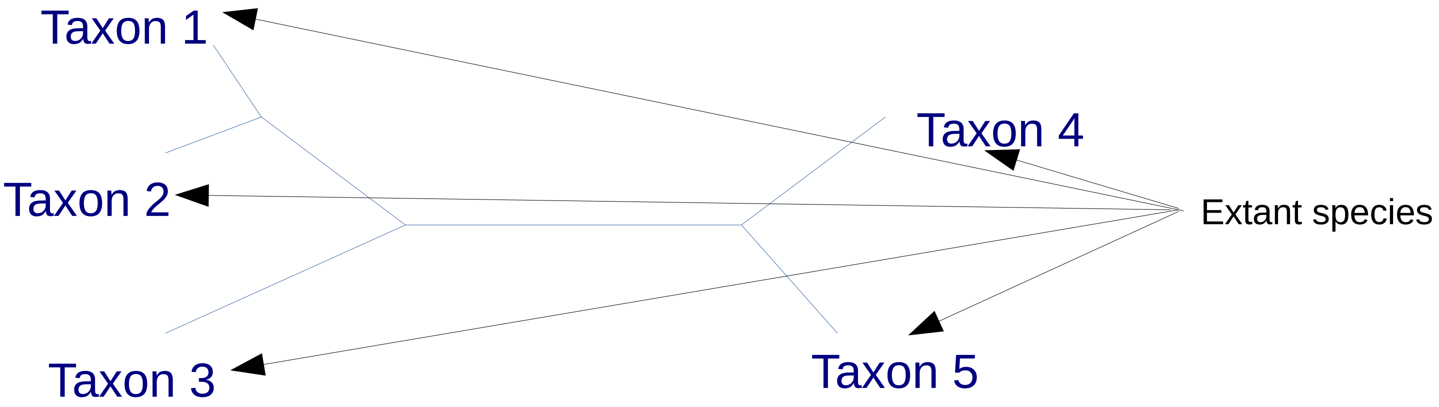
- Introduction to Phylogenetic Inference - *Alexandros*
- The RAxML Search Algorithm - *Alexandros*
- Improvements in RAxML **N**ext **G**eneration - *Alexey*
- Tutorial - *Alexey*

A phylogeny

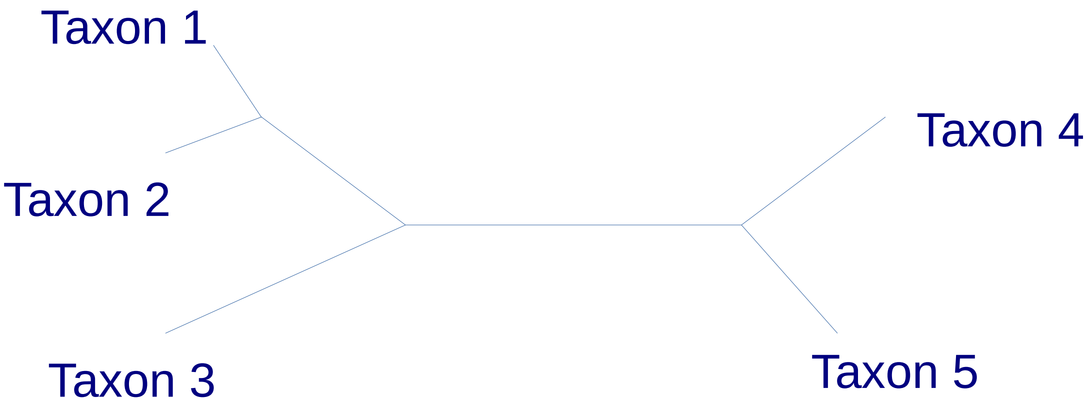


Phylogenies describe evolutionary relationships among species

A phylogeny

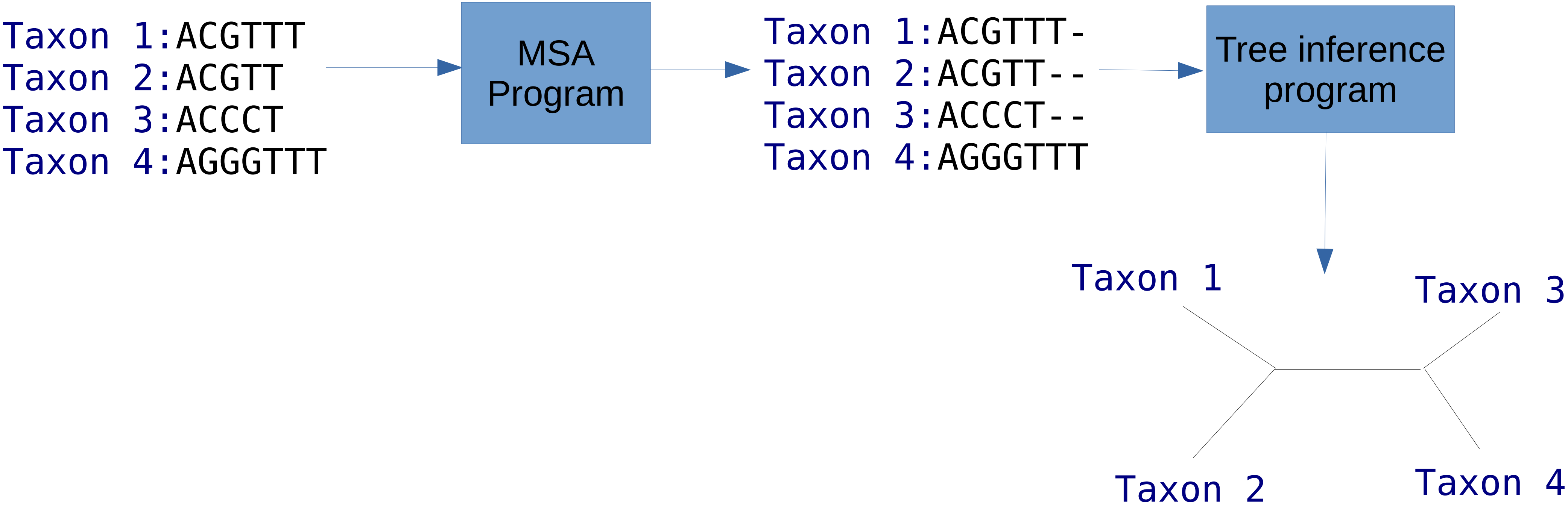


A phylogeny



Phylogenetic trees are unrooted binary trees!!!

Tree Inference Pipeline



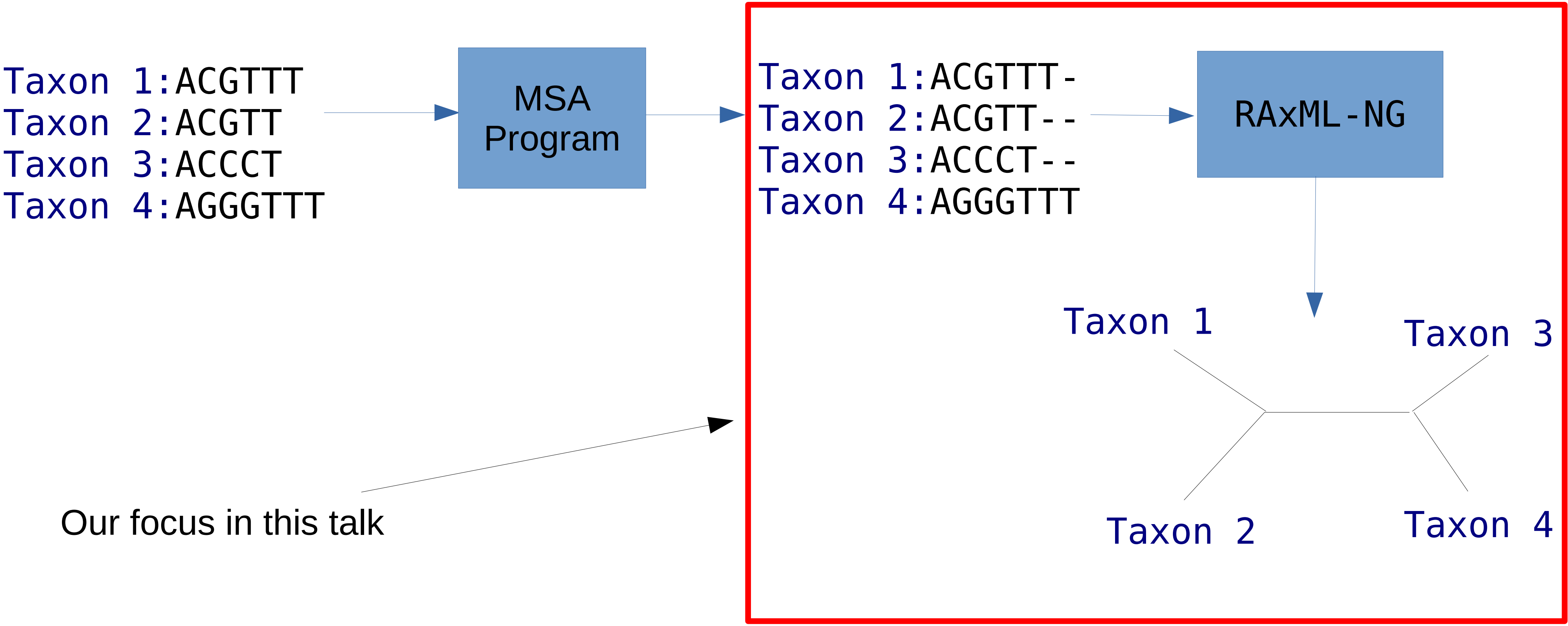
```
Taxon 1:ACGTTT
Taxon 2:ACGTT
Taxon 3:ACCCT
Taxon 4:AGGGTTT
```

MSA Program

```
Taxon 1:ACGTTT-
Taxon 2:ACGTT--
Taxon 3:ACCCT--
Taxon 4:AGGGTTT
```

Tree inference program

Tree Inference Pipeline



Taxon 1:ACGTTT
Taxon 2:ACGTT
Taxon 3:ACCCT
Taxon 4:AGGGTTT

MSA Program

Taxon 1:ACGTTT-
Taxon 2:ACGTT--
Taxon 3:ACCCT--
Taxon 4:AGGGTTT

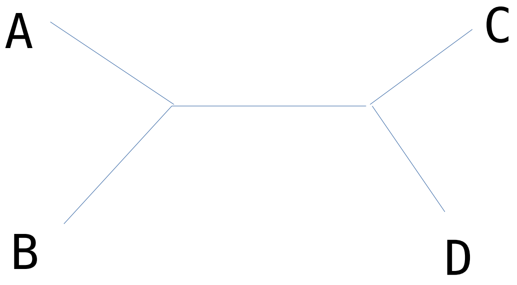
RAxML-NG

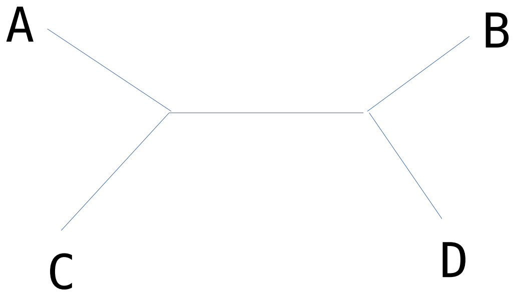
Taxon 1
Taxon 3
Taxon 2
Taxon 4

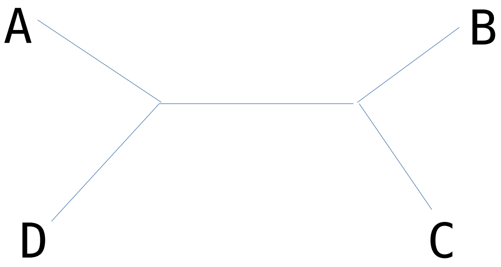
Our focus in this talk

How many unrooted 4-taxon trees exist?

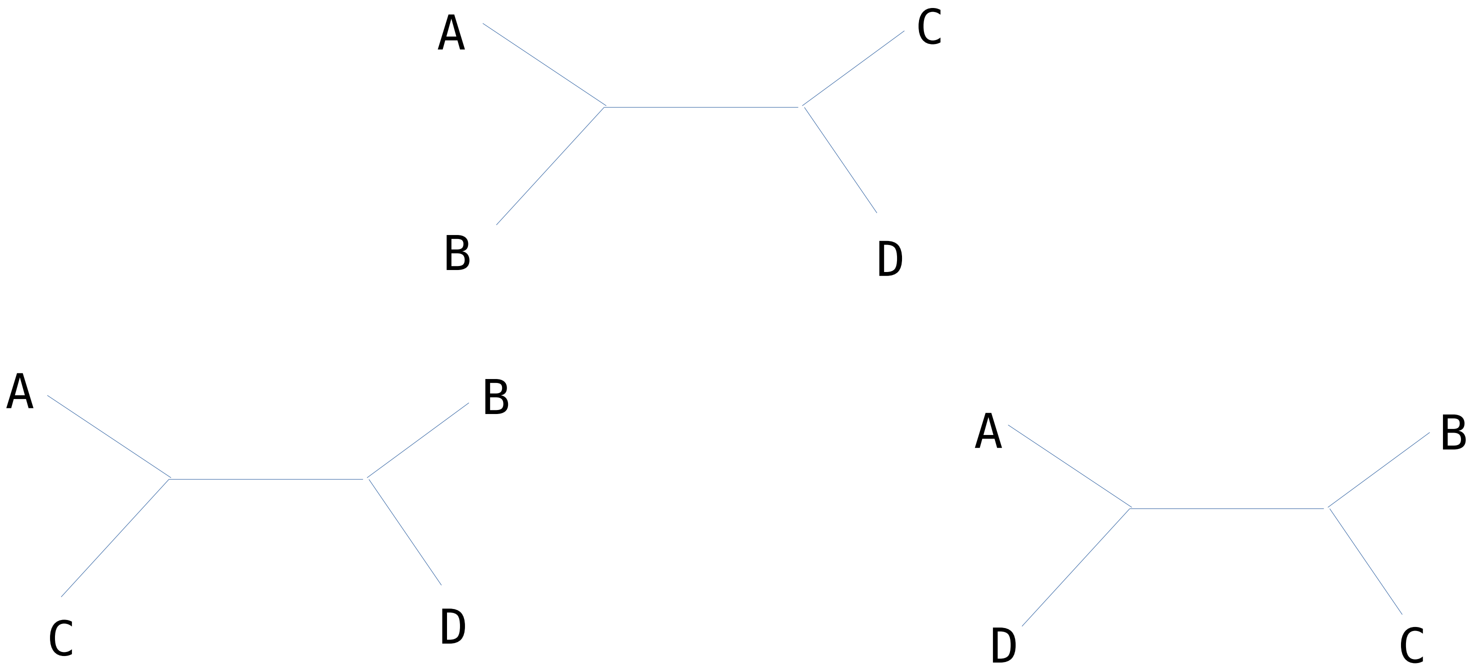
How many unrooted 4-taxon trees exist?



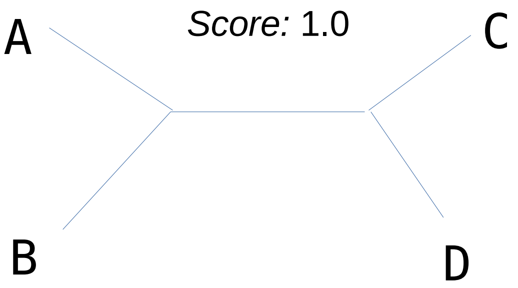




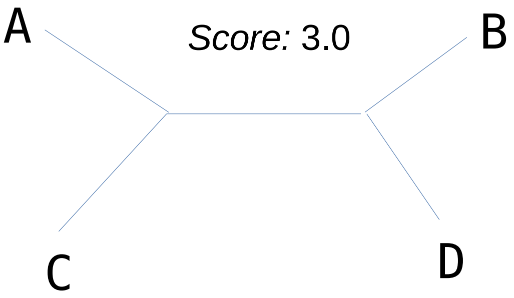
How do we select among them ?



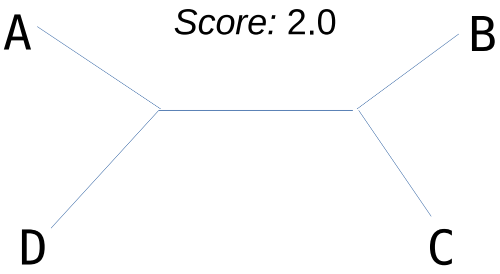
How do we select among them???



Score: 1.0



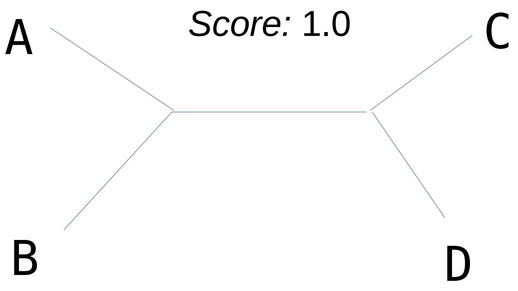
Score: 3.0

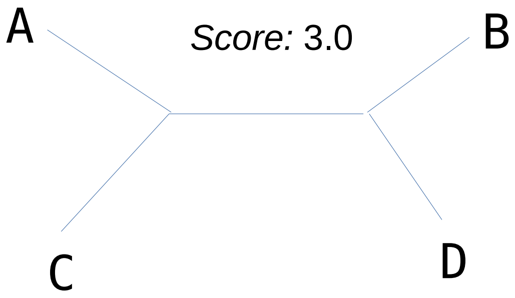


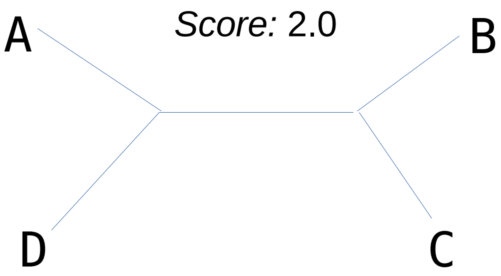
Score: 2.0

We need scoring criteria!!!

How do we select among them???

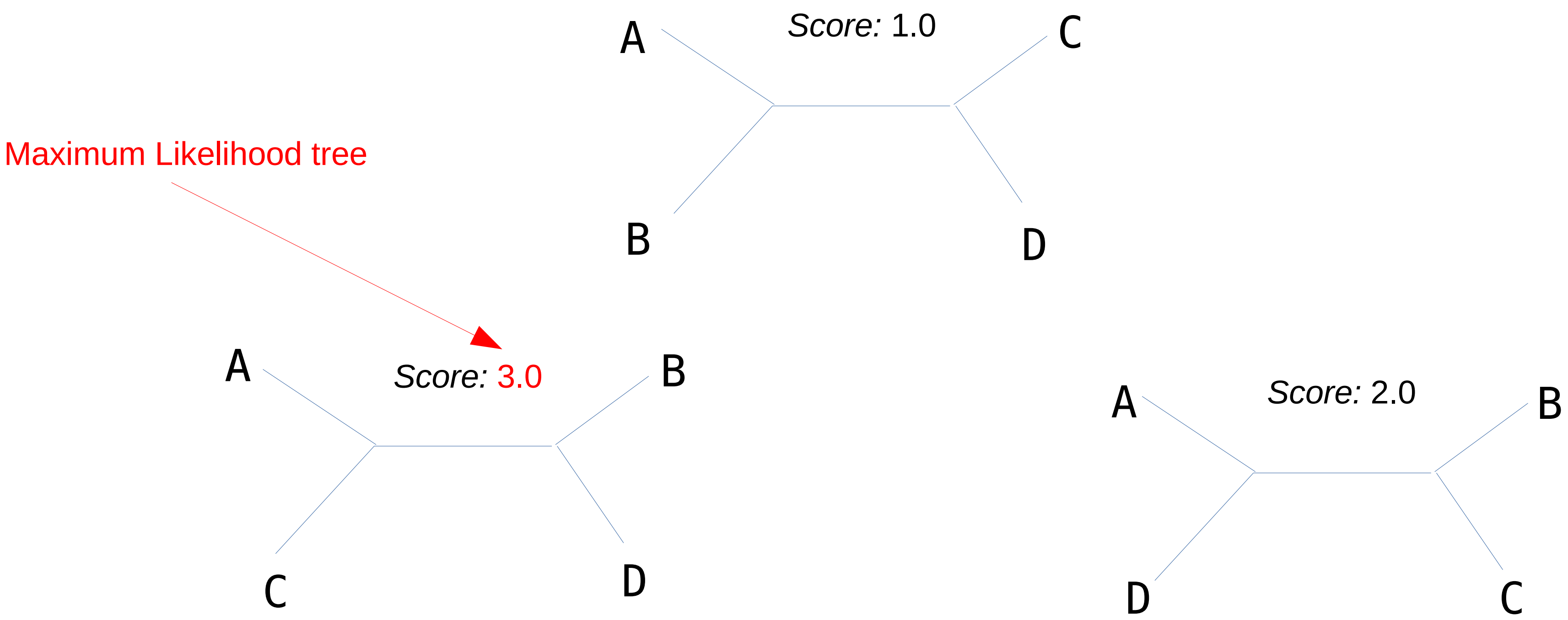






We need scoring criteria!!!
The currently most widely used criterion is maximum likelihood:
How likely is it that the tree, given a model of evolution, generated the observed data?

How do we select among them???



Maximum Likelihood tree

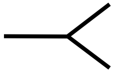
Score: 1.0

Score: 3.0

Score: 2.0

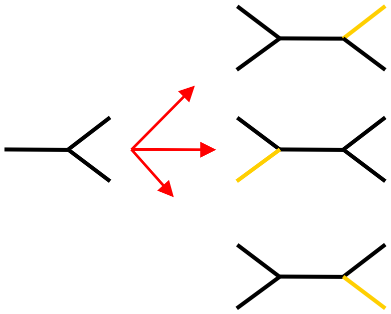
We need scoring criteria!!!
The currently most widely used criterion is maximum likelihood:
How likely is it that the tree, given a model of evolution, generated the observed data?

The number of trees



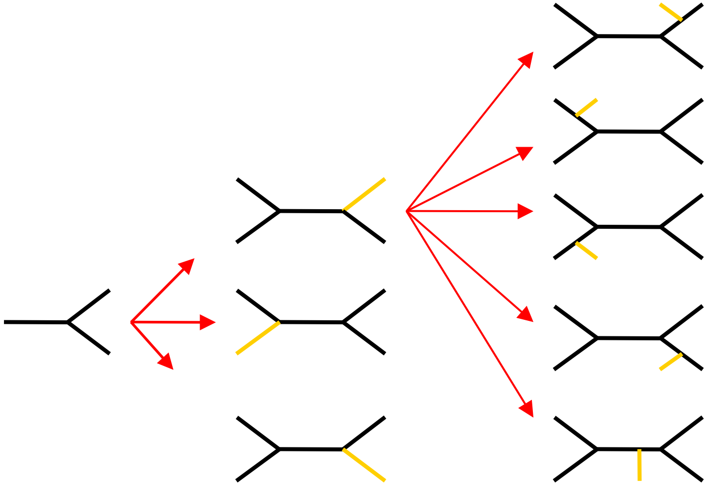
3 taxa → *1 tree*

The number of trees



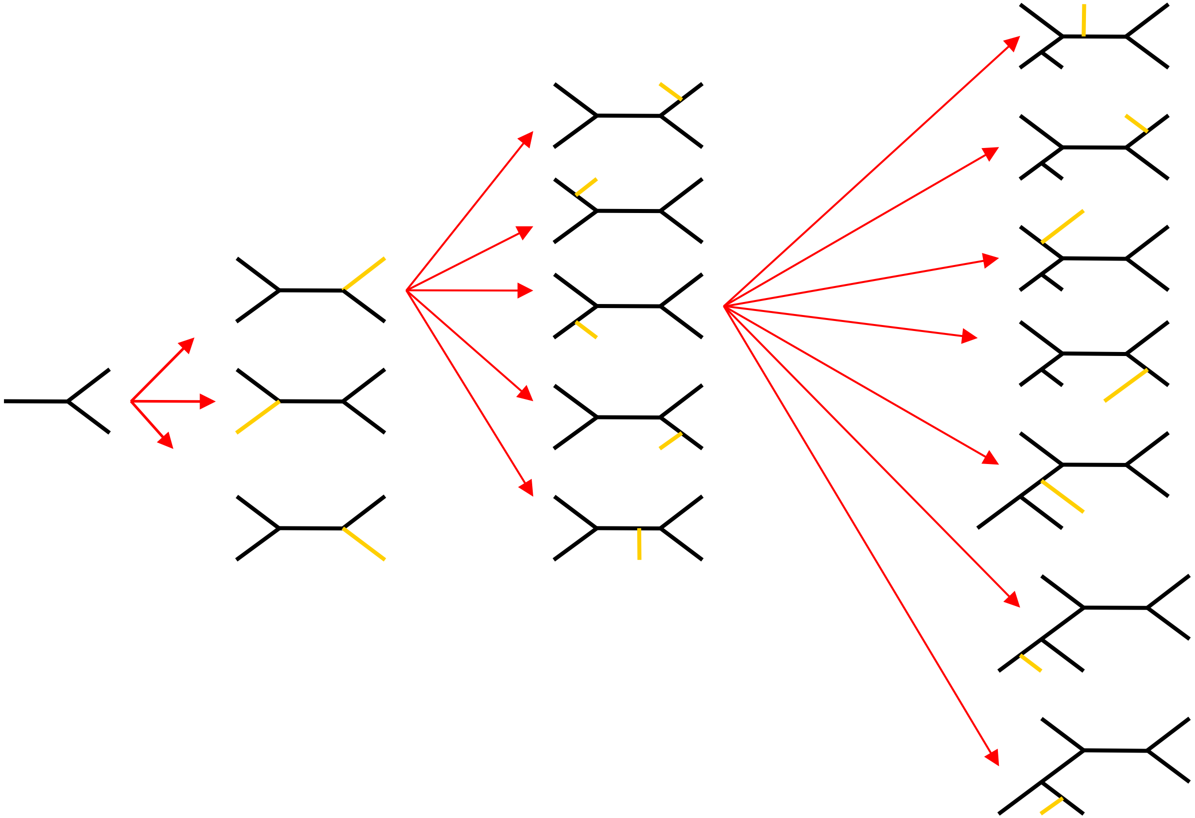
4 taxa → *3 trees*

The number of trees



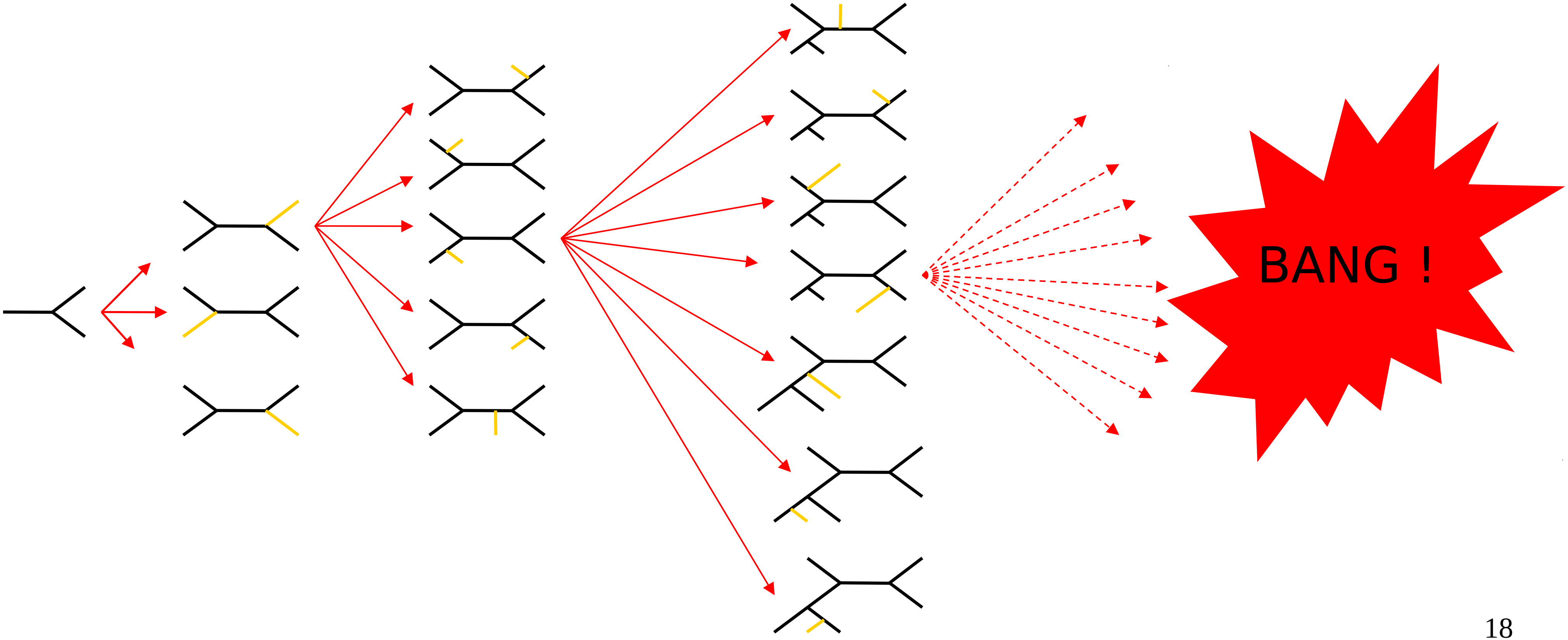
5 taxa → *15 trees*

The number of trees

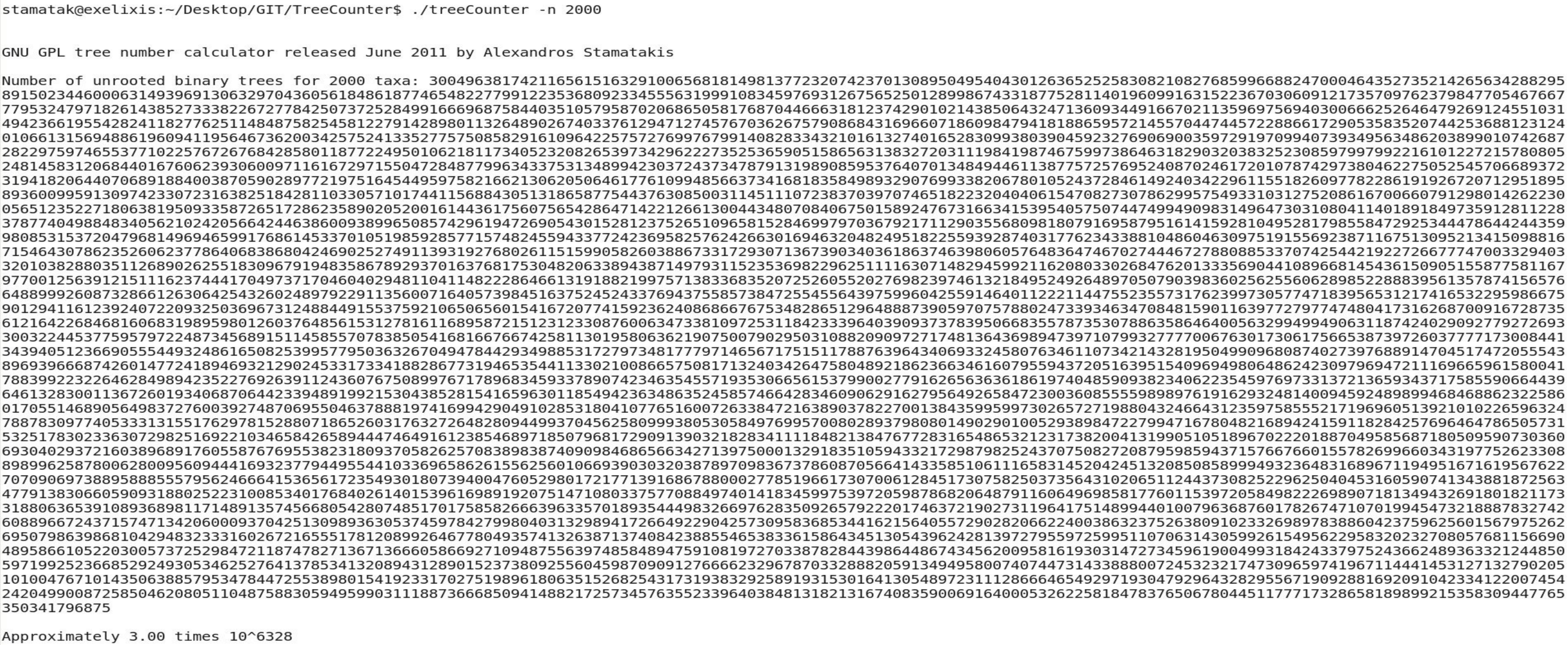


6 taxa → *105 trees*

The number of trees explodes!

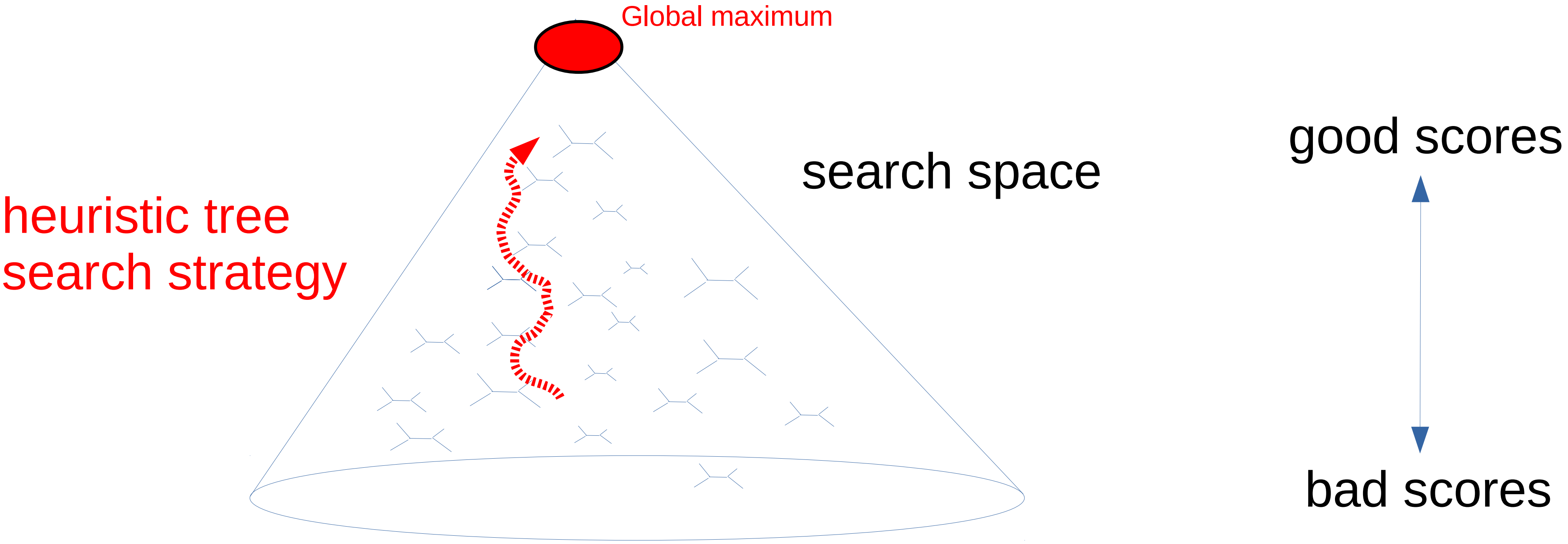


# possible trees with 2000 taxa



```
stamatak@exelixis:~/Desktop/GIT/TreeCounter$ ./treeCounter -n 2000


GNU GPL tree number calculator released June 2011 by Alexandros Stamatakis

Number of unrooted binary trees for 2000 taxa: 30049638174211656151632910065681814981377232074237013089504954043012636525258308210827685996688247000464352735214265634288295
891502344600063149396913063297043605618486187746548227799122353680923345556319991083459769312675652501289986743318775281140196099163152236703060912173570976237984770546 7667
779532479718261438527338226727784250737252849916669687584403510579587020686505817687044666318123742901021438506432471360934491667021135969756940300666252646479269124551031
4942366195542824118277625114848758254581227914289801132648902674033761294712745767036267579086843169660718609847941818865957214557044744572288661729053583520744253688123124
0106613156948861960941195646736200342575241335277575085829161096422575727699767991408283343210161327401652830993803904592327690690035972919709940739349563486203899010742687
2822975974655377102257672676842858011877224950106218117340523208265397342962227352536590515865631383272031119841987467599738646318290320383252308597997992216101227215780805
2481458312068440167606239306009711616729715504728487799634337531348994230372437347879131989085953764070134849446113877572576952408702461720107874297380462275052545706689372
3194182064407068918840038705902897721975164544959758216621306205064617761099485663734168183584989329076993382067801052437284614924034229611551826097782286191926720712951895
8936009959130974233072316382518428110330571017441156884305131865877544376308500311451110723837039707465182232040406154708273078629957549331031275208616700660791298014262230
0565123522718063819509335872651728623589020520016144361756075654286471422126613004434807084067501589247673166341539540575074474994909831496473031080411401891849735912811228
3787740498848340562102420566424463860093899650857429619472690543015281237526510965815284699797036792171129035568098180791695879516141592810495281798558472925344478644244359
9808531537204796814969465991768614533701051985928577157482455943377242369582576242663016946320482495182255939287403177623433881048604630975191556923871167513095213415098816
7154643078623526062377864068386804246902527491139319276802611515990582603886733172930713673903403618637463980605764836474670274446727880885337074254421922726677747003329403
3201038288035112689026255183096791948358678929370163768175304820633894387149793115235369822962511116307148294599211620803302684762013335690441089668145436150905155877581167
9770012563912151116237444170497371704604029481104114822286466131918821997571383368352072526055202769823974613218495249264897050790398360256255606289852288839561357874156576
6488999260873286612630642543260248979229113560071640573984516375245243376943755857384725545564397599604255914640112221144755235573176239973057747183956531217416532295986675
9012941161239240722093250369673124884491553759210650656015416720774159236240868667675348286512964888739059707578802473393463470848159011639772797747480417316268700916728735
6121642268468160683198959801260376485615312781611689587215123123308760063473381097253118423339640390937378395066835578735307886358646400563299499490631187424029092779272693
3003224453775957972248734568915114585570783850541681667667425811301958063621907500790295031088209097271748136436989473971079932777700676301730617566538739726037777173008441
3439405123669055544932486165082539957795036326704947844293498853172797348177797146567175151178876396434069332458076346110734214328195049909680874027397688914704517472055543
8969396668742601477241894693212902453317334188286773194653544113302100866575081713240342647580489218623663461607955943720516395154096949806486242309796947211169665961580041
7883992232264628498942352276926391124360767508997671789683459337890742346354557193530665615379900277916265636361861974048590938234062235459769733137213659343717585590664439
6461328300113672601934068706442339489199215304385281541659630118549423634863524585746642834609062916279564926584723003608555598989761916293248140094592489899468468862322586
0170551468905649837276003927487069550463788819741699429049102853180410776516007263384721638903782270013843599599730265727198804324664312359758555217196960513921010226596324
7887830977405333131551762978152880718652603176327264828094499370456258099938053058497699570080289379808014902901005293898472279947167804821689424159118284257696464786505731
5325178302336307298251692210346584265894447464916123854689718507968172909139032182834111184821384767728316548653212317382004131990510518967022201887049585687180509590730360
6930402937216038968917605587676955382318093705826257083898387409098468656634271397500013291835105943321729879825243707508272087959859437157667660155782699660343197752623308
8989962587800628009560944416932377944955441033696586261556256010669390303203878970983673786087056641433585106111658314520424513208508589994932364831689671194951671619567622
7070906973889588855579562466641536561723549301807394004760529801721771391686788000277851966173070061284517307582503735643102065112443730825229625040453160590741343881872563
4779138306605909318802522310085340176840261401539616989192075147108033757708849740141834599753972059878682064879116064969858177601153972058498222698907181349432691801821173
3188063653910893689811714891357456680542807485170175858266639633570189354449832669762835092657922201746372190273119641751489944010079636876017826747107019945473218887832742
6088966724371574713420600093704251309893630537459784279980403132989417266492290425730958368534416215640557290282066224003863237526380910233269897838860423759625601567975262
6950798639868104294832333160267216555178120899264677804935741326387137408423885546538336158643451305439624281397279559725995110706314305992615495622958320232708057681156690
4895866105220300573725298472118747827136713666058669271094875563974858489475910819727033878284439864486743456200958161930314727345961900499318424337975243662489363321244850
5971992523668529249305346252764137853413208943128901523738092556045987090912766662329678703328882059134949580074074473143388800724532321747309659741967114441453127132790205
1010047671014350638857953478447255389801541923317027519896180635152682543173193832925891931530164130548972311128666465492971930479296432829556719092881692091042334122007454
2420499008725850462080511048758830594959903111887366685094148821725734576355233964038481318213167408359006916400053262258184783765067804451177717328658189899215358309447765
350341796875

Approximately 3.00 times 10^6328
```

Problem Complexity



Global maximum

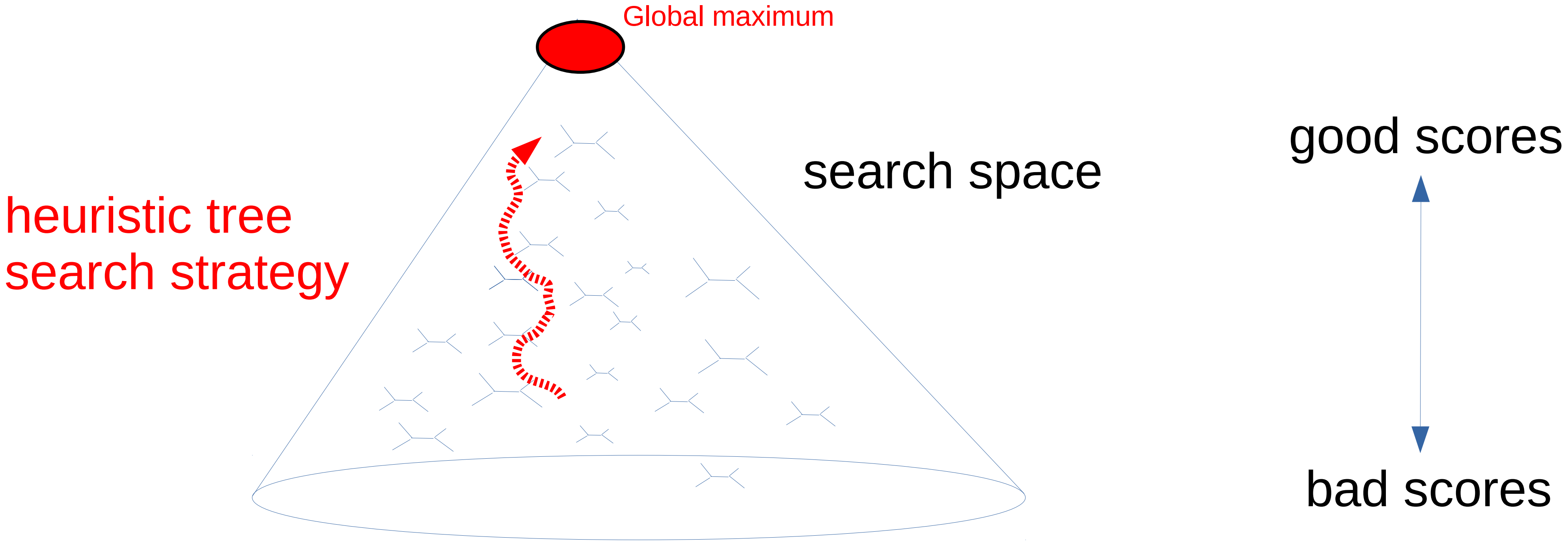
heuristic tree search strategy

search space

good scores

bad scores

Problem Complexity



Global maximum

heuristic tree
search strategy

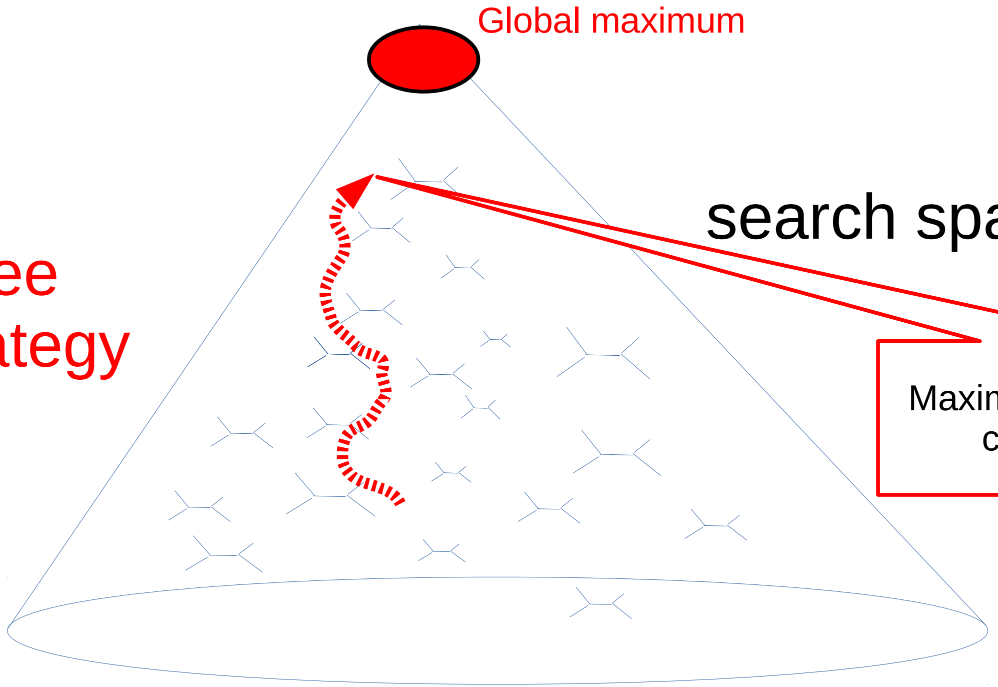
search space

good scores

bad scores

Finding the best tree under Maximum Likelihood is **NP-hard**!

Problem Complexity

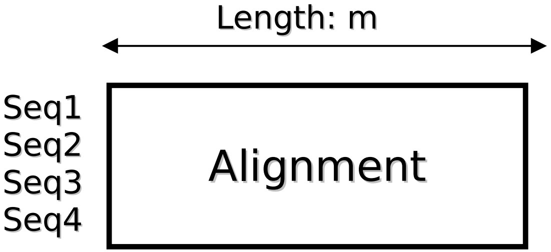


heuristic tree
search strategy

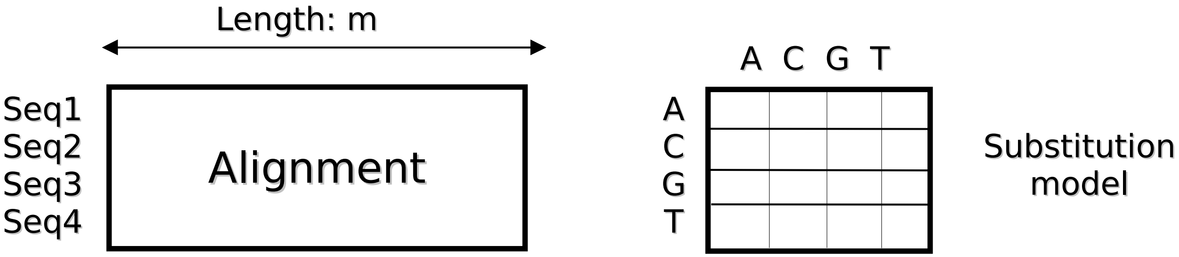
search space

Maximum Likelihood tree searches can end up in local optima

Maximum Likelihood



Maximum Likelihood



Length: m

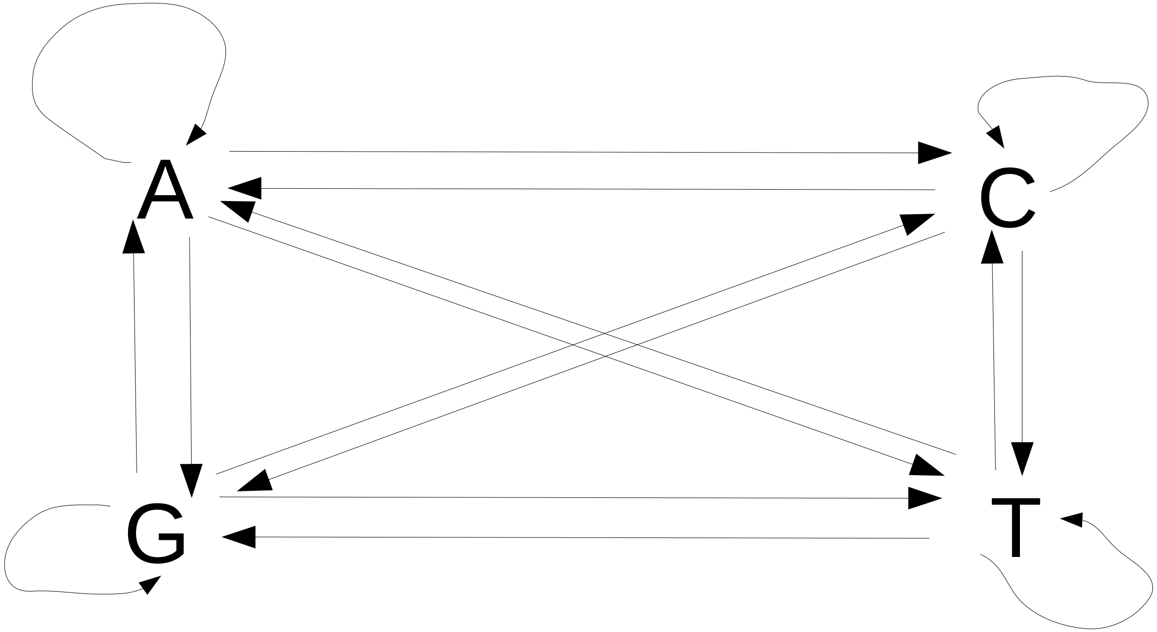
Seq1
Seq2
Seq3
Seq4

Alignment

	A	C	G	T
A				
C				
G				
T				

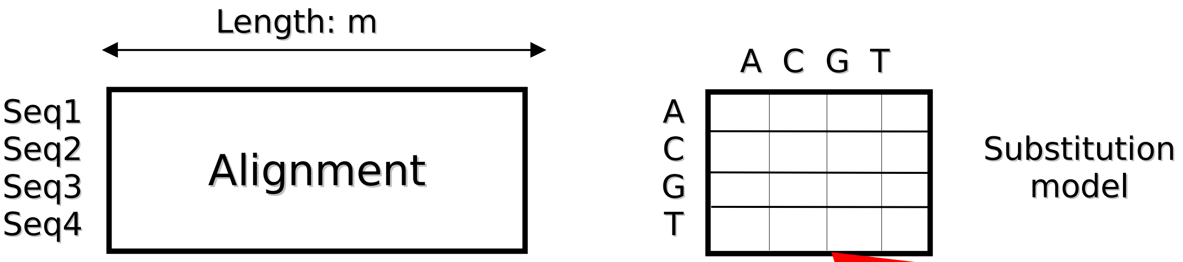
Substitution model

Nucleotide Substitution Models



We model evolution as time-reversible Markov Process!

Maximum Likelihood



Length: m

Seq1
Seq2
Seq3
Seq4

Alignment

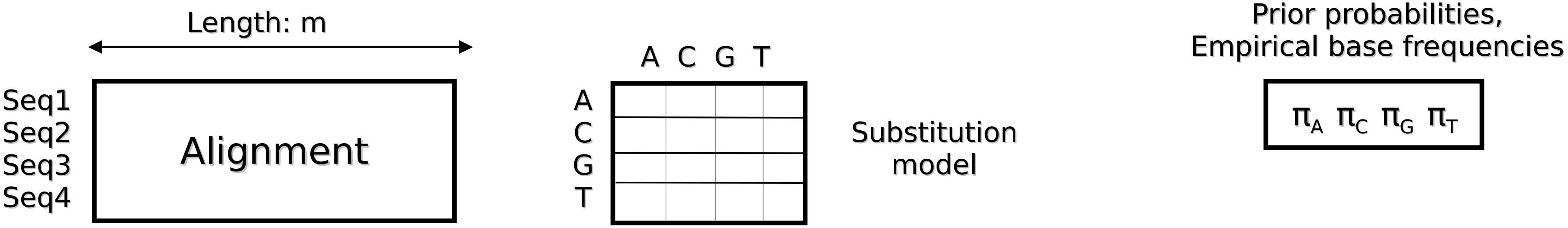
A C G T

A
C
G
T

Substitution model

Commonly denoted as *Q* matrix: transition probs for time *dt*, for time *t*: $P(t)=e^{Qt}$

Maximum Likelihood



Length: m

Seq1
Seq2
Seq3
Seq4

Alignment

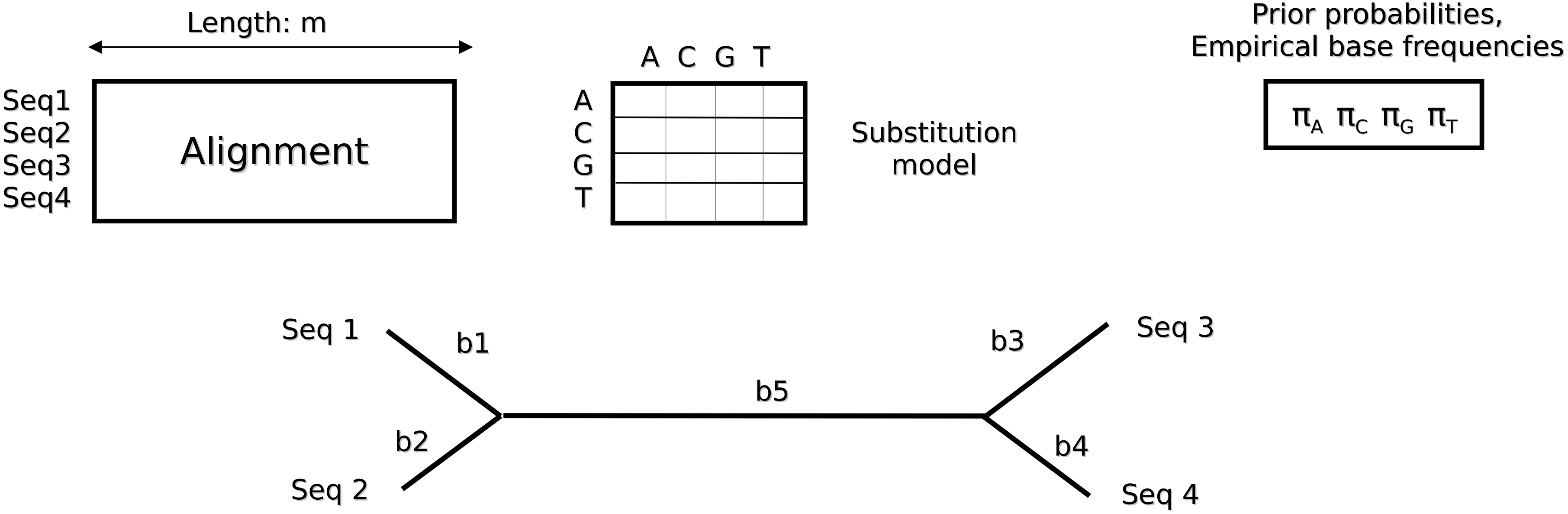
	A	C	G	T
A				
C				
G				
T				

Substitution model

Prior probabilities,
Empirical base frequencies

π_A π_C π_G π_T

Maximum Likelihood



Length: m

Seq1
Seq2
Seq3
Seq4

Alignment

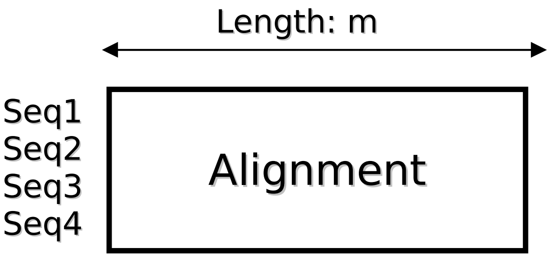
	A	C	G	T
A				
C				
G				
T				

Substitution
model

Prior probabilities,
Empirical base frequencies

π_A π_C π_G π_T

Maximum Likelihood



Length: m

Seq1
Seq2
Seq3
Seq4

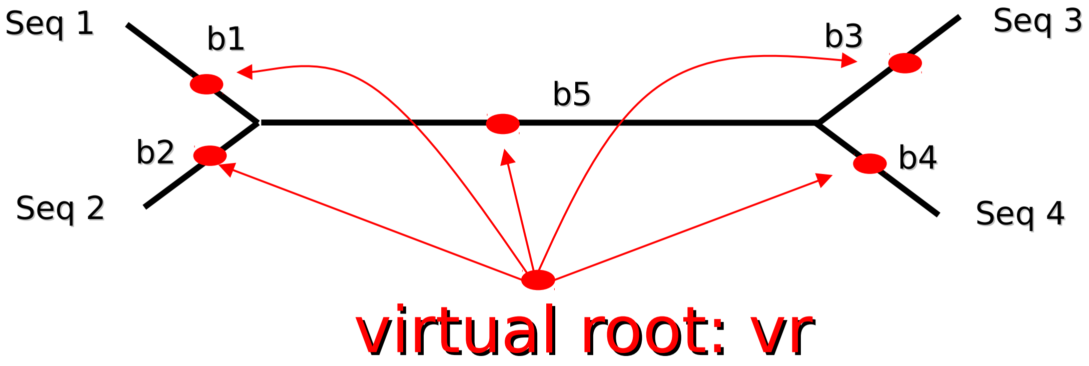
Alignment

	A	C	G	T
A				
C				
G				
T				

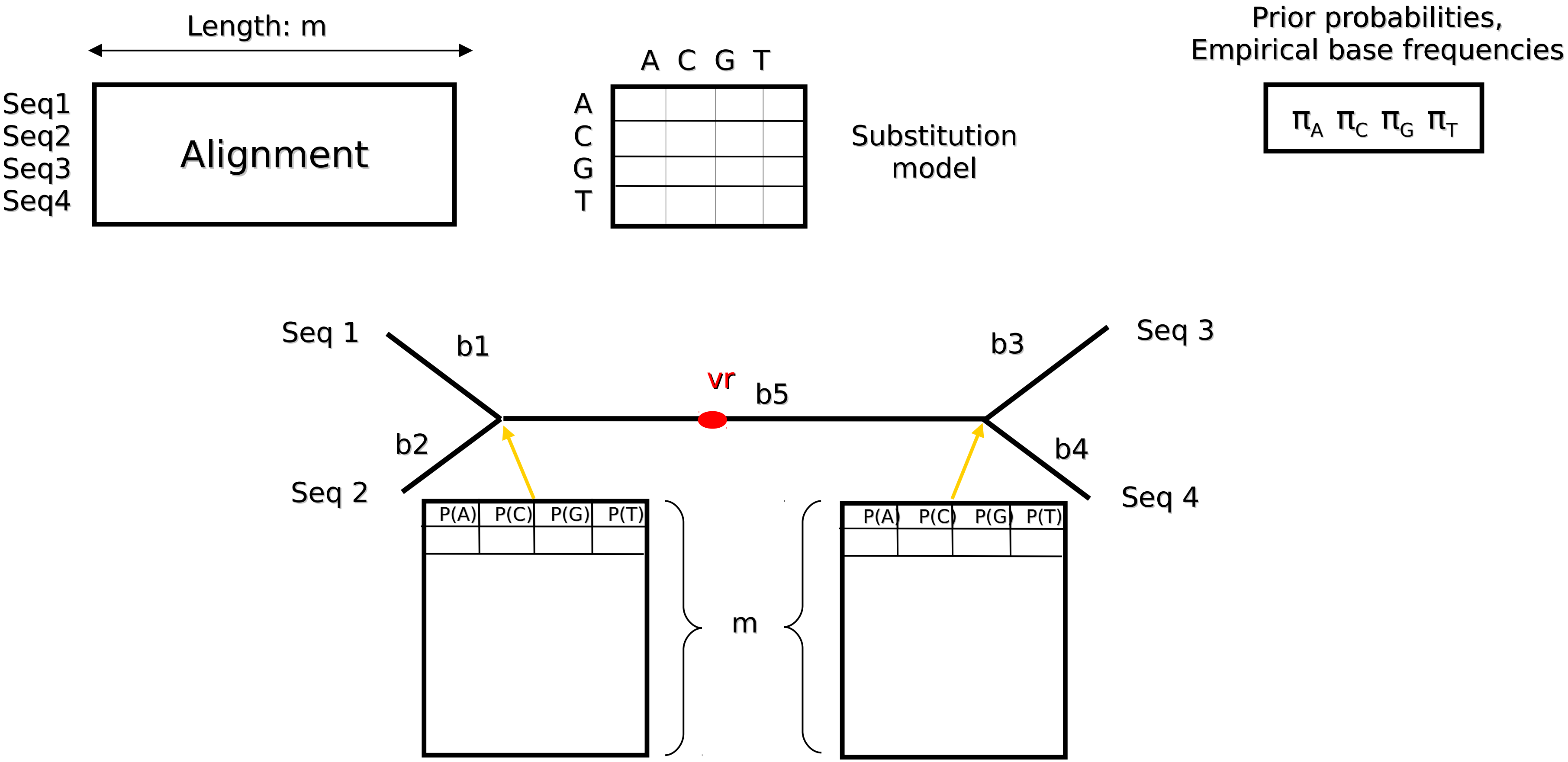
Substitution model

Prior probabilities,
Empirical base frequencies

$\boxed{\pi_A\ \pi_C\ \pi_G\ \pi_T}$



Maximum Likelihood



Length: m

Seq1
Seq2
Seq3
Seq4

Alignment

	A	C	G	T
A				
C				
G				
T				

Substitution model

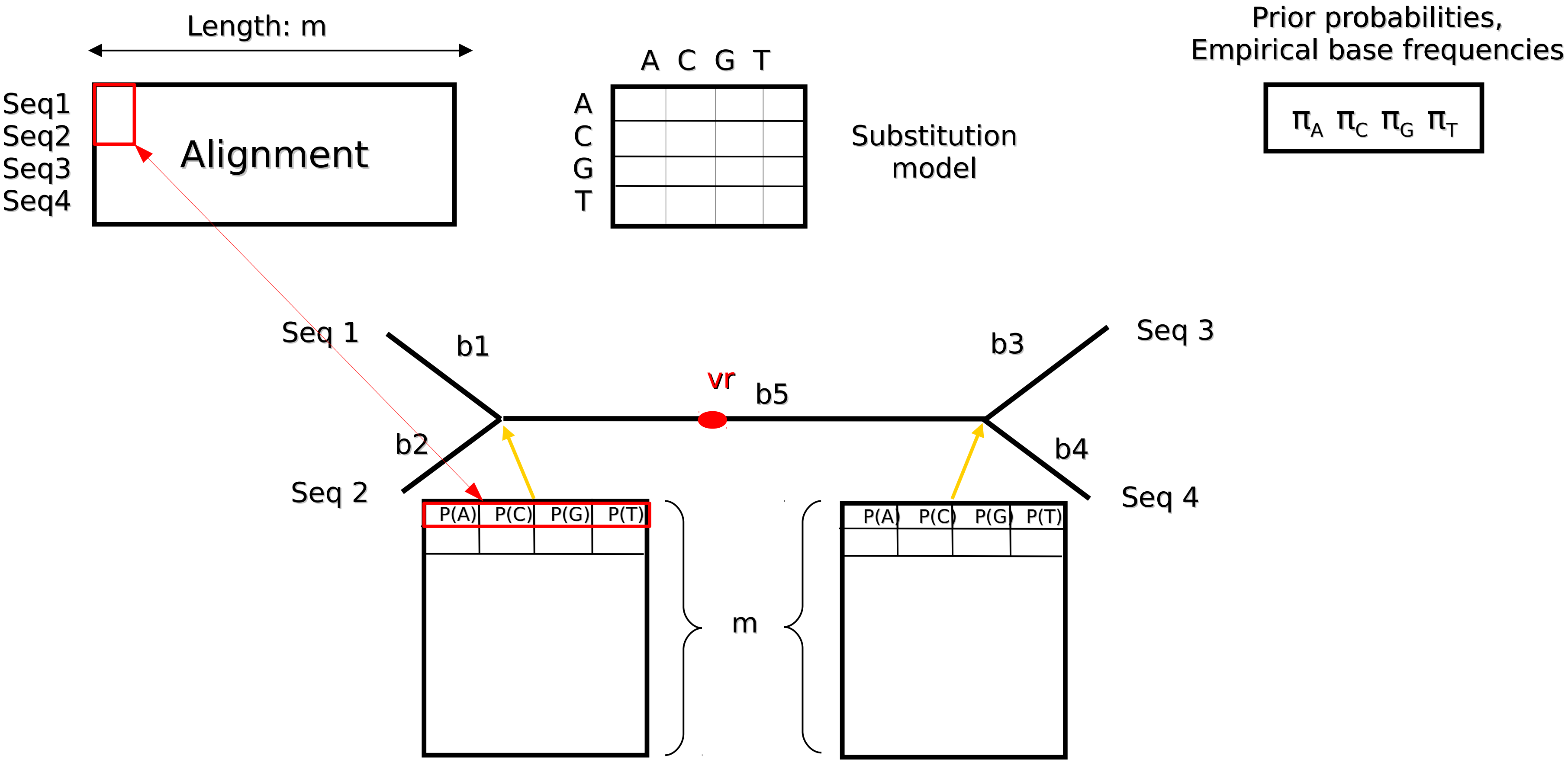
Prior probabilities,
Empirical base frequencies

π_A π_C π_G π_T

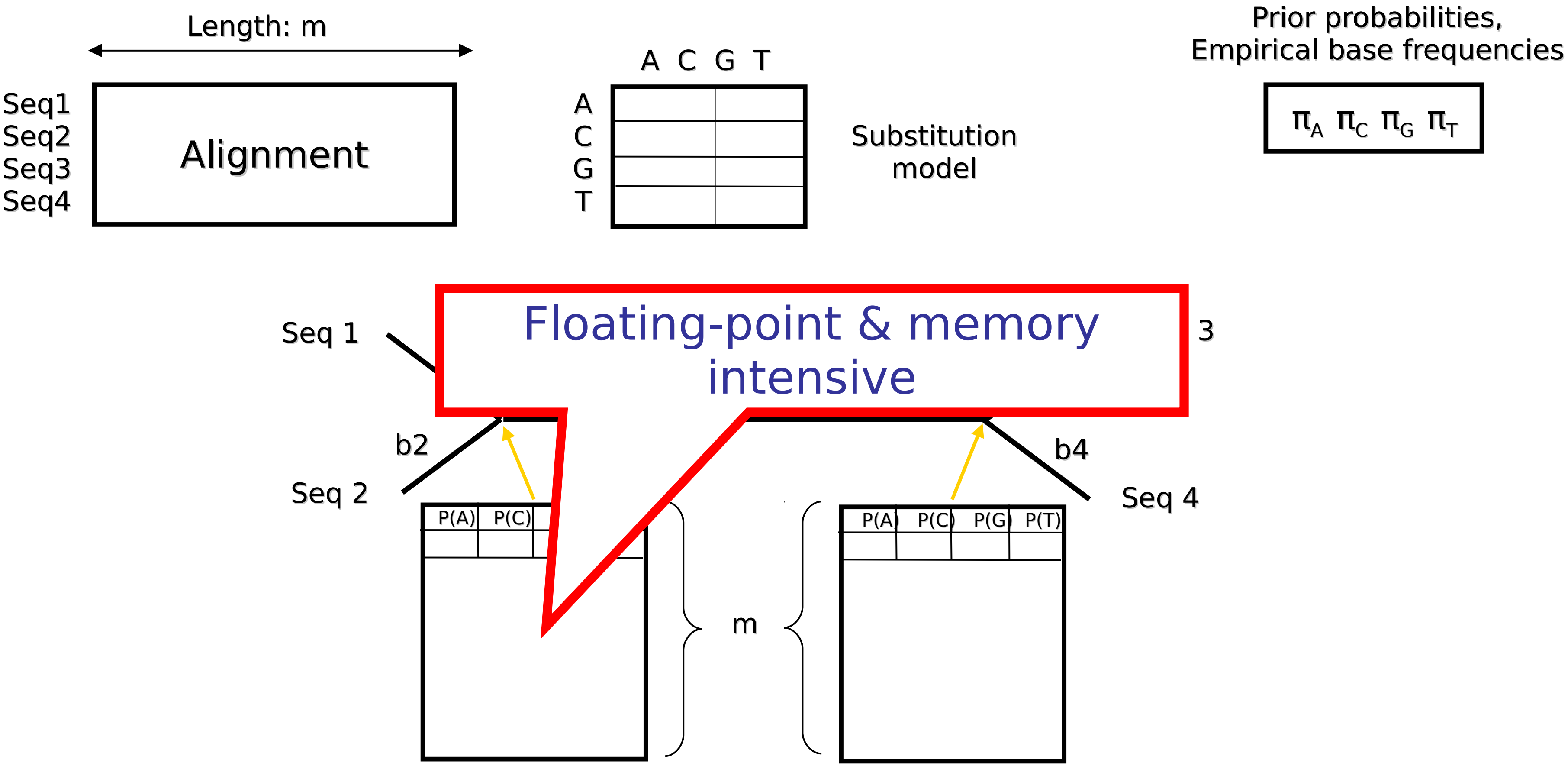
P(A)	P(C)	P(G)	P(T)

P(A)	P(C)	P(G)	P(T)

Maximum Likelihood

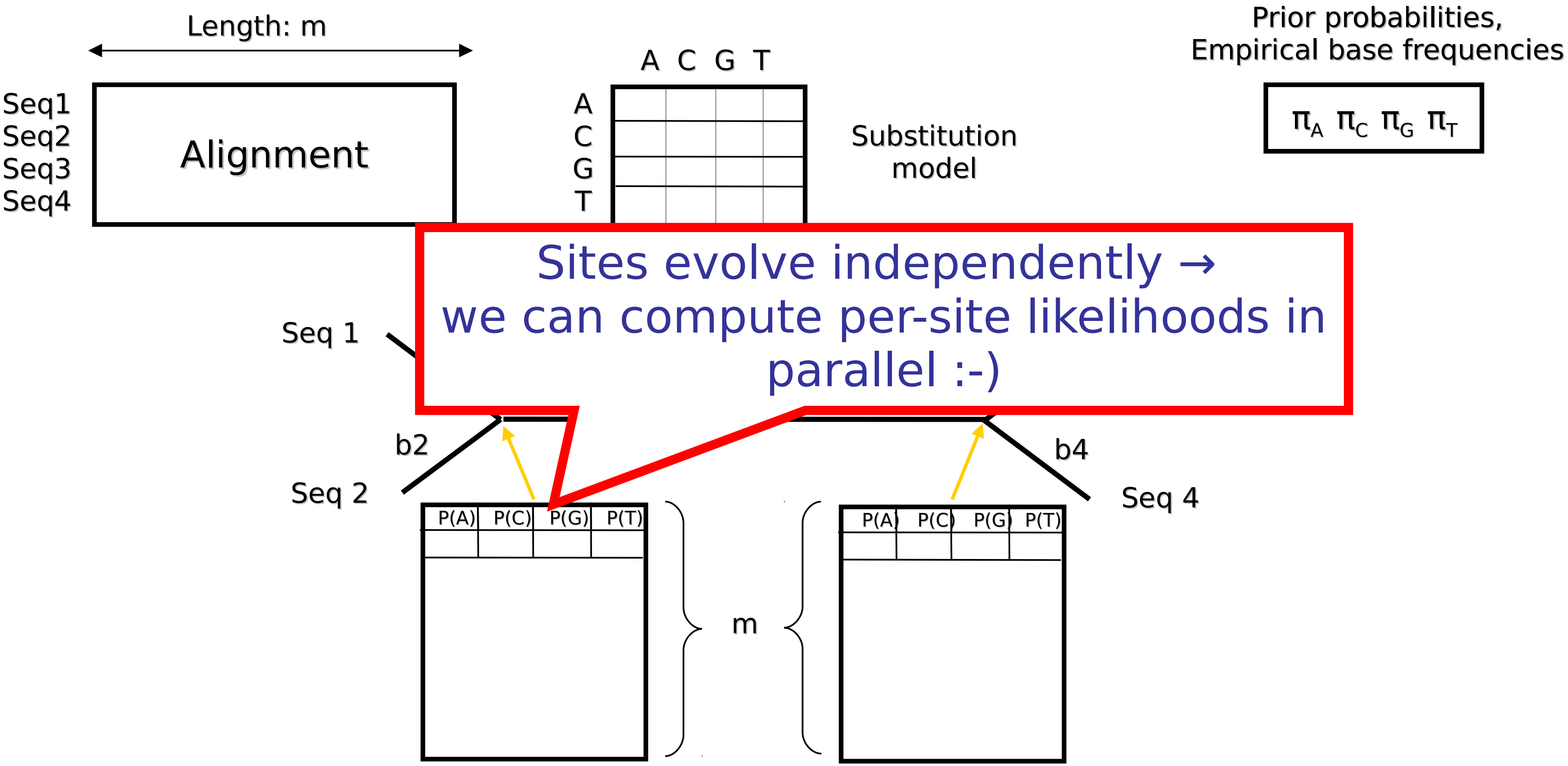


Maximum Likelihood



Floating-point & memory intensive

Maximum Likelihood



Sites evolve independently →
we can compute per-site likelihoods in parallel :-)

Outline

- Introduction to Phylogenetic Inference - *Alexandros*
- **The RAxML Search Algorithm -** ***Alexandros***
- Improvements in RAxML **N**ext **G**eneration - *Alexey*
- Tutorial - *Alexey*

How does it work?

Compute randomized stepwise addition order
Maximum Parsimony tree

How does it work?

Compute randomized stepwise addition order Maximum Parsimony tree

Advantage of RAxML: search starts from distinct point in search space every time

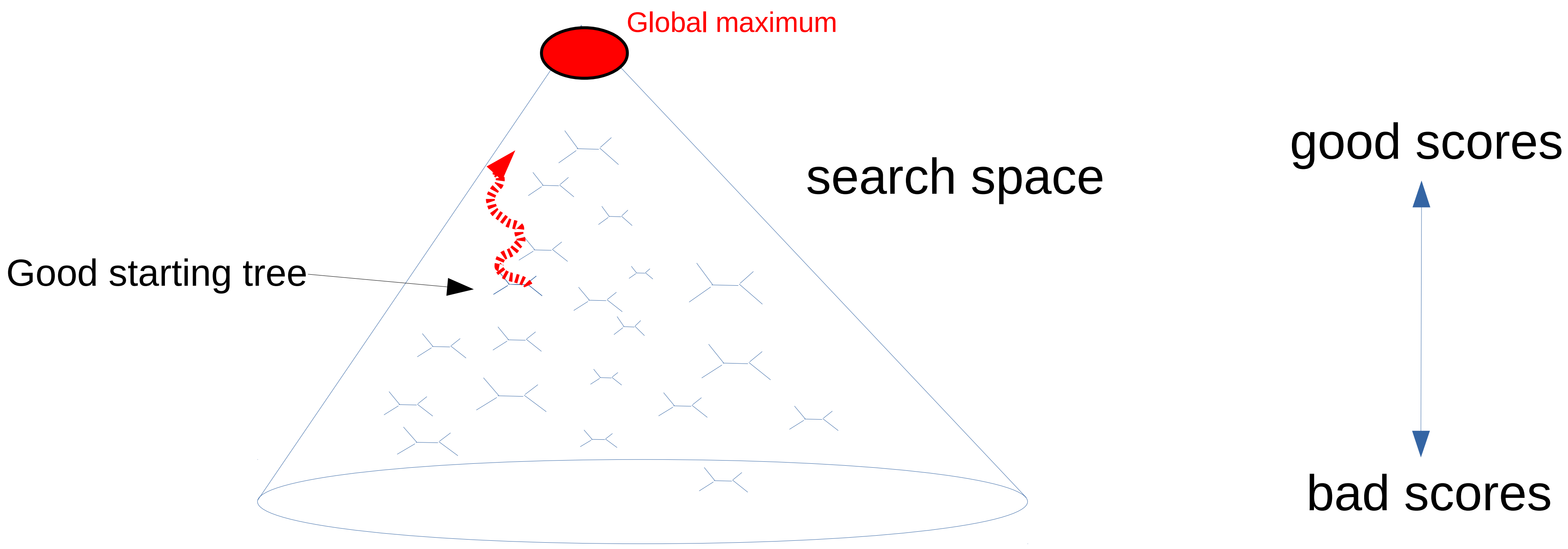
How does it work?

Compute randomized stepwise addition order Maximum Parsimony tree

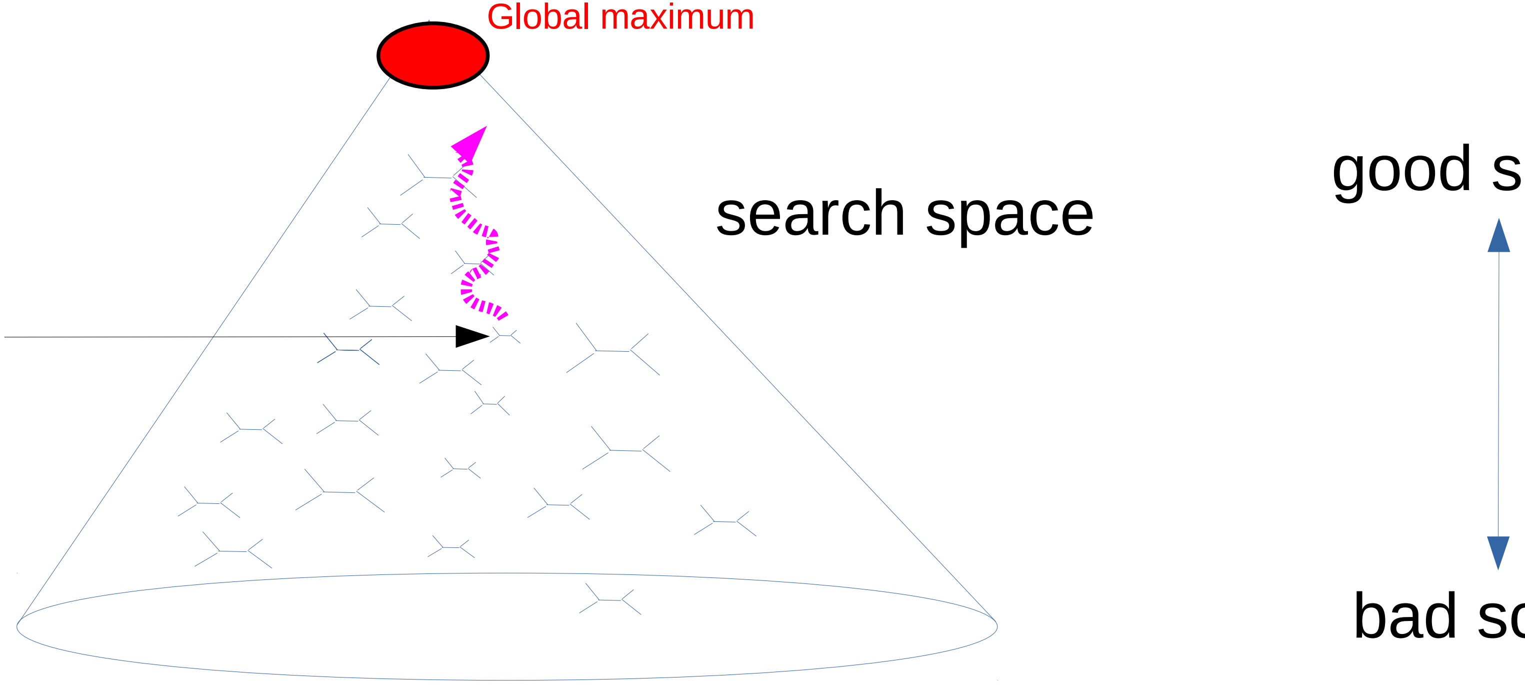
Advantage of RAxML: search starts from distinct point in search space every time

Alternatively, we can start from a completely random tree

Starting Trees



Starting Trees



Global maximum

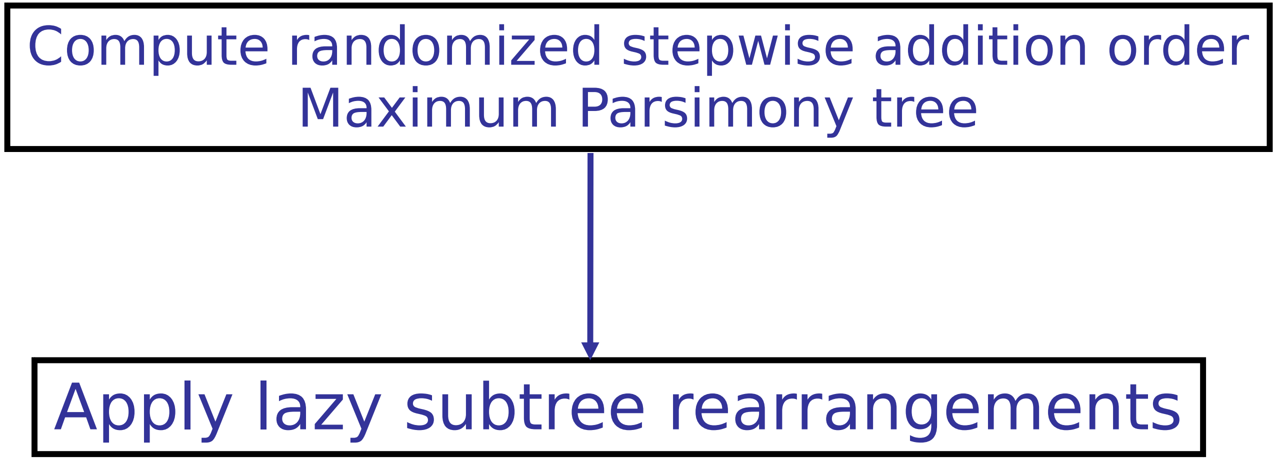
search space

Another good starting tree

good scores

bad scores

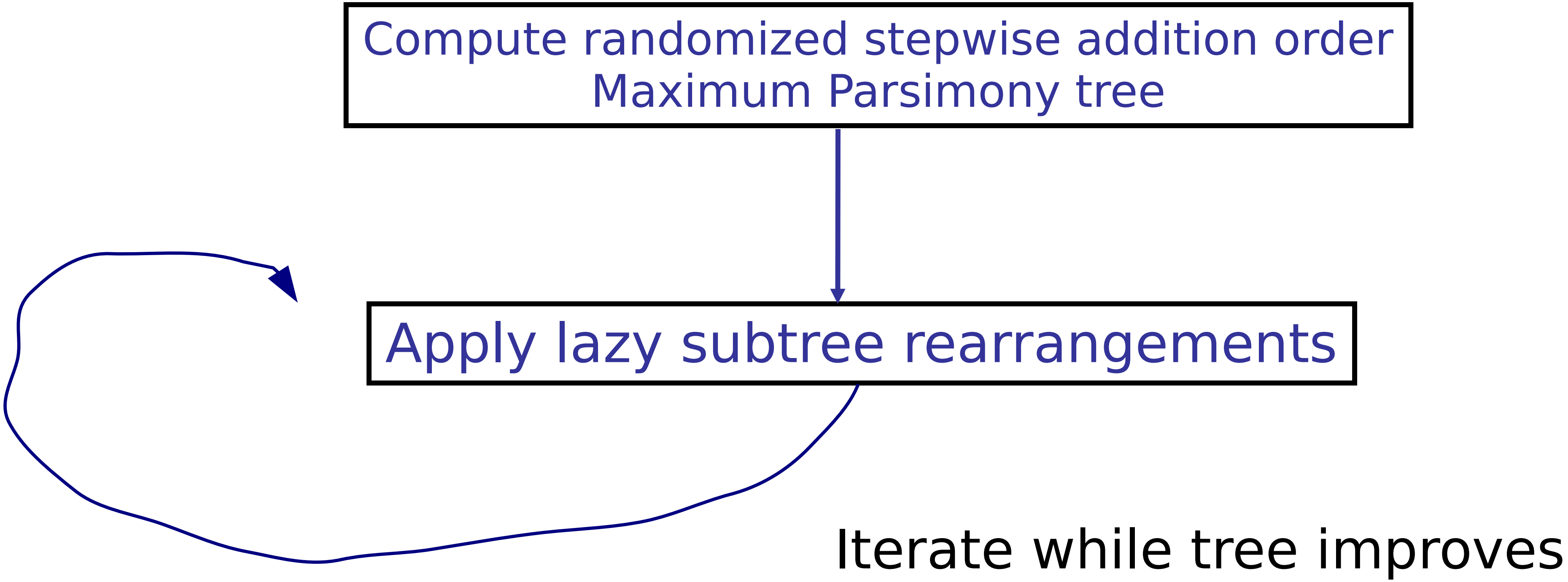
How does it work?



Compute randomized stepwise addition order Maximum Parsimony tree

Apply lazy subtree rearrangements

How does it work?

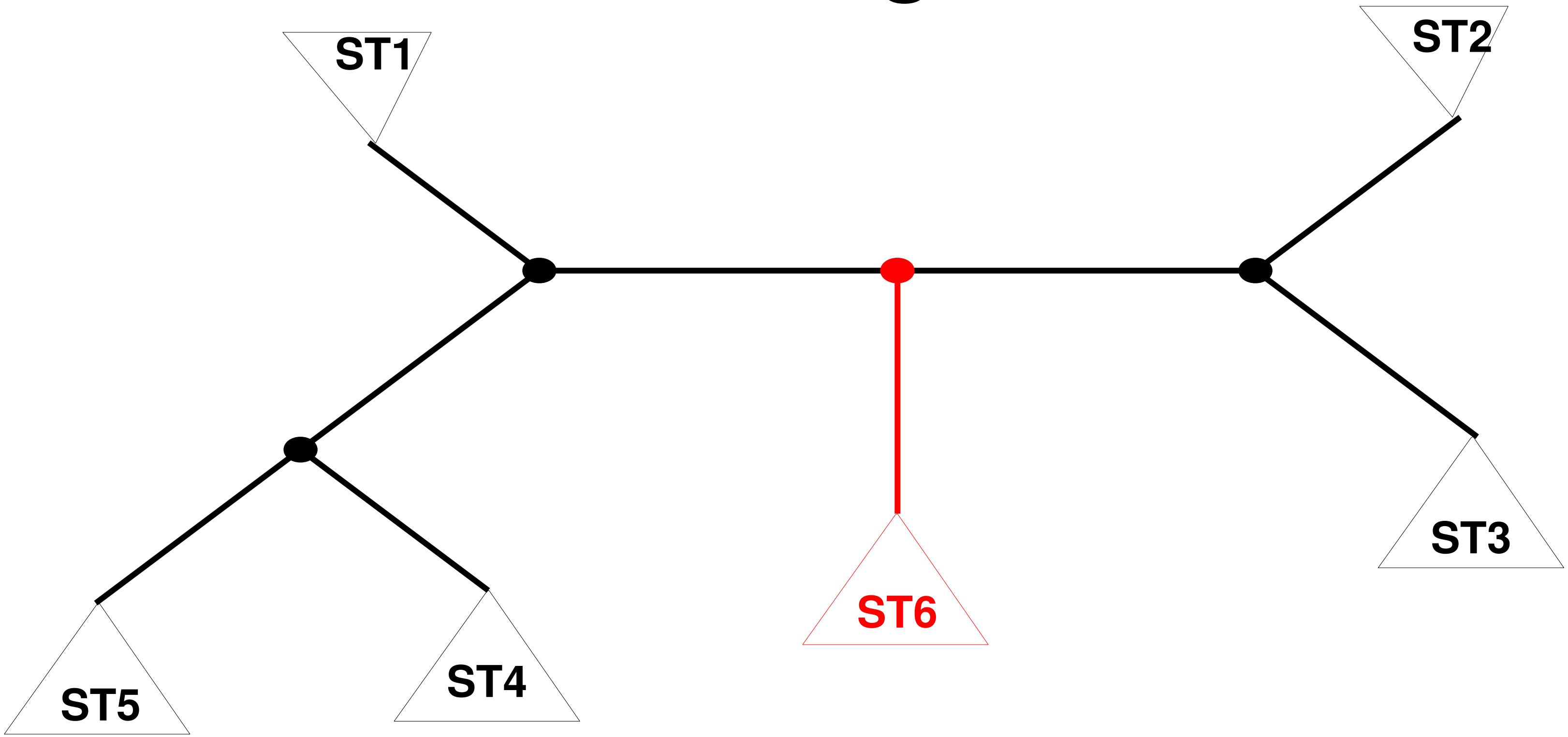


Compute randomized stepwise addition order Maximum Parsimony tree

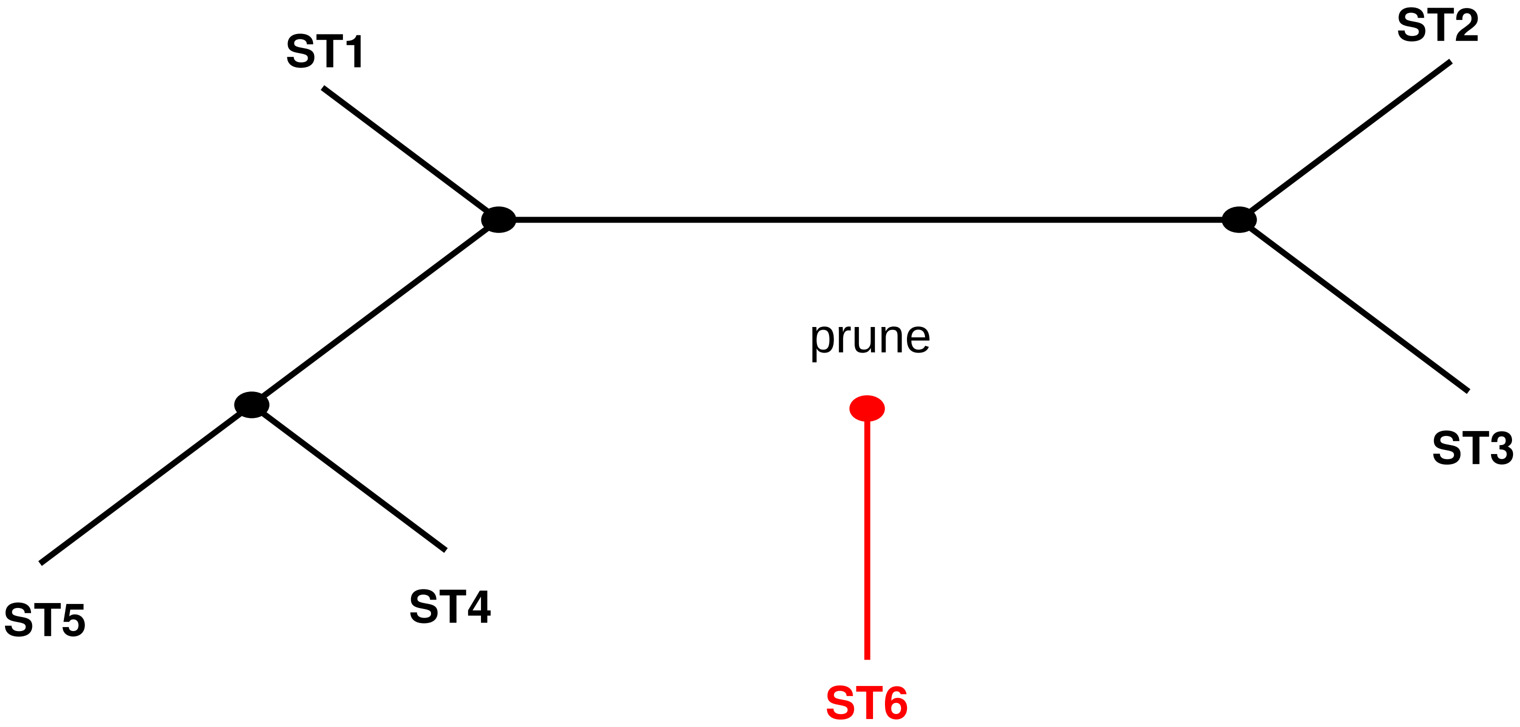
Apply lazy subtree rearrangements

Iterate while tree improves

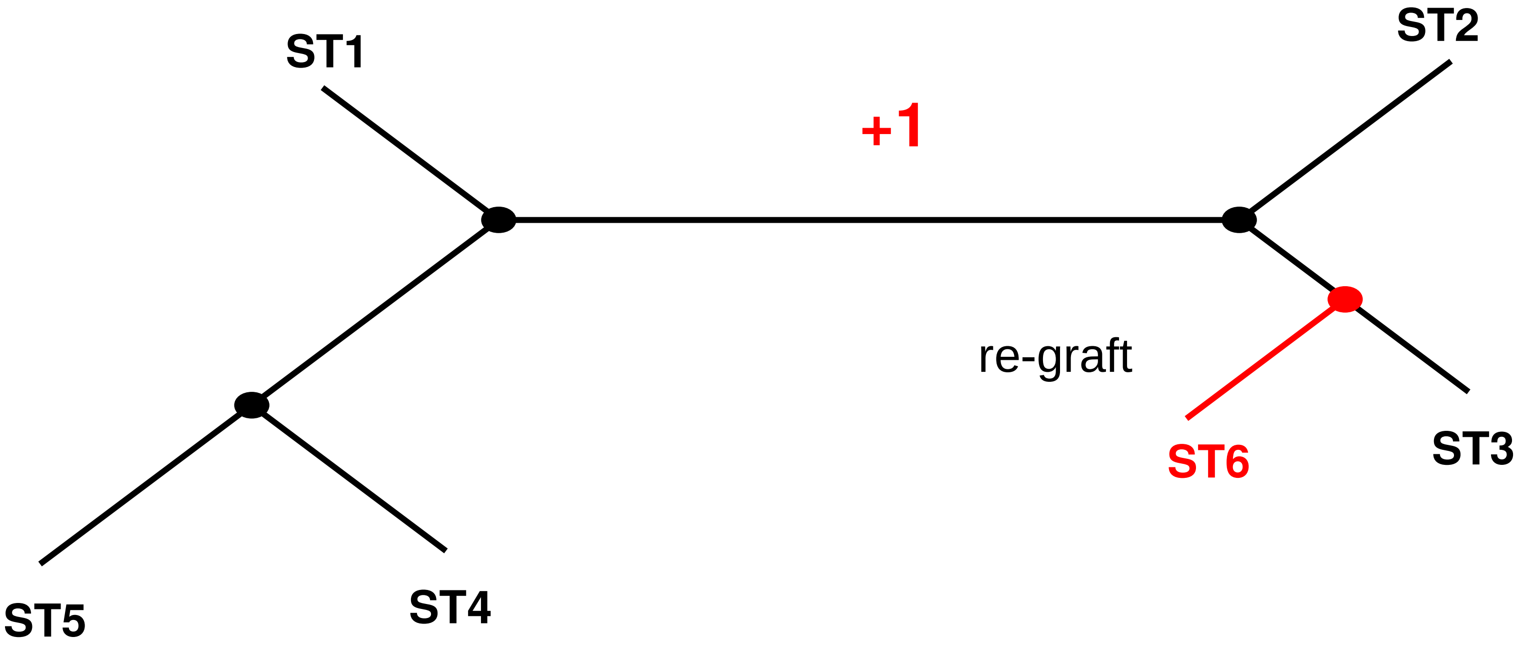
Subtree Pruning & Re-Grafting



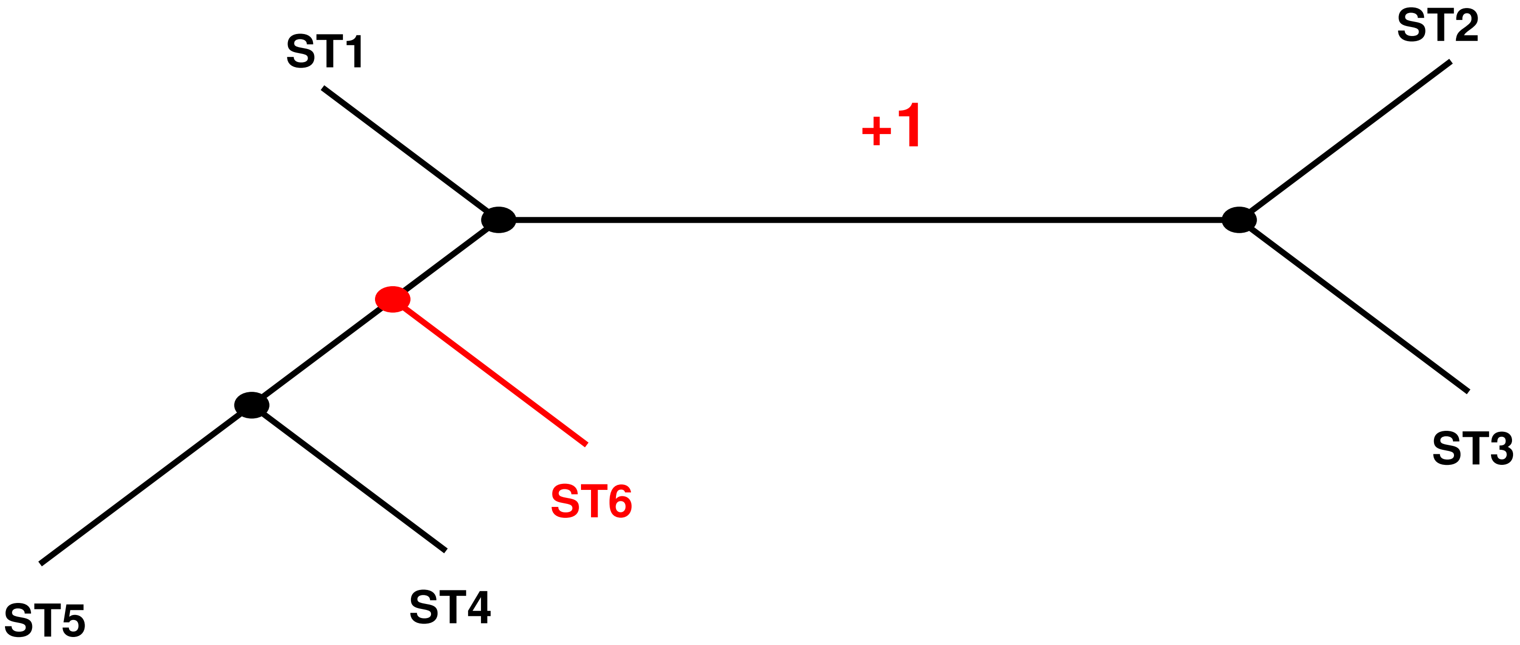
Subtree Pruning & Re-Grafting



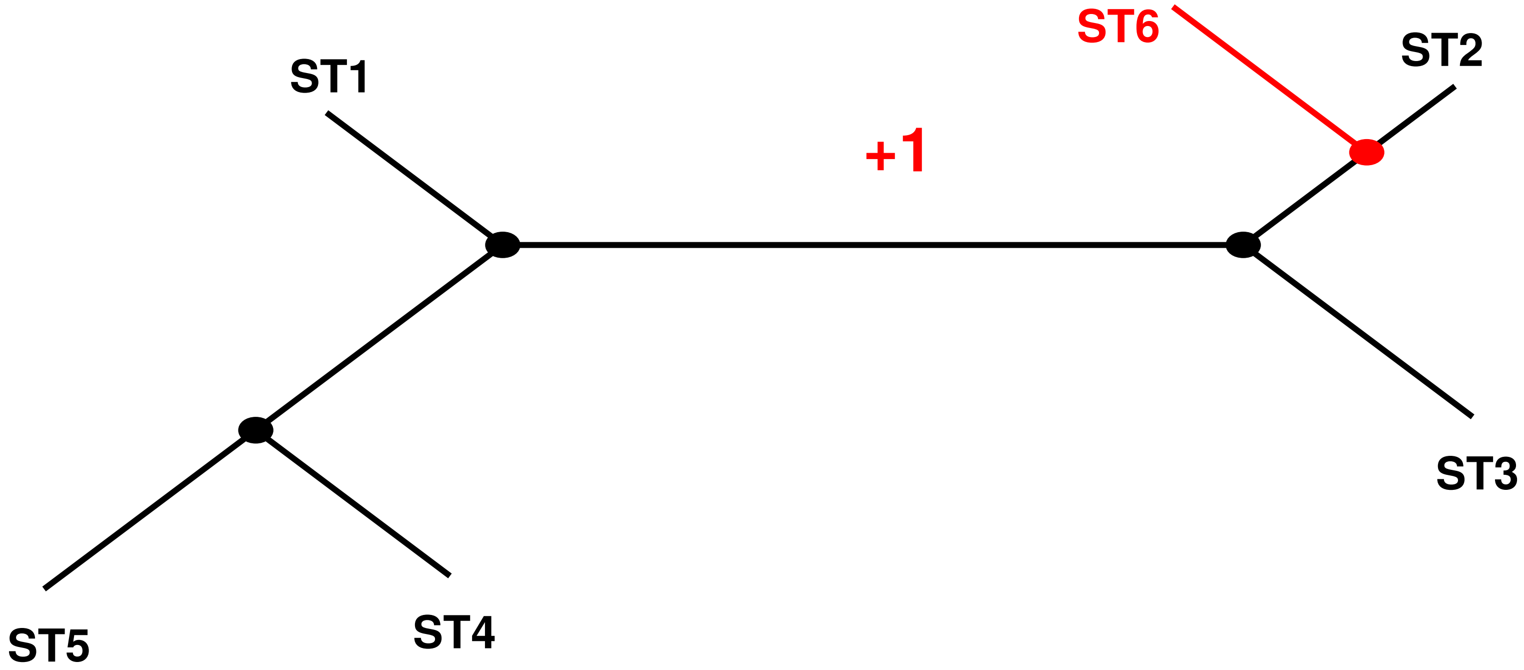
Subtree Pruning & Re-Grafting



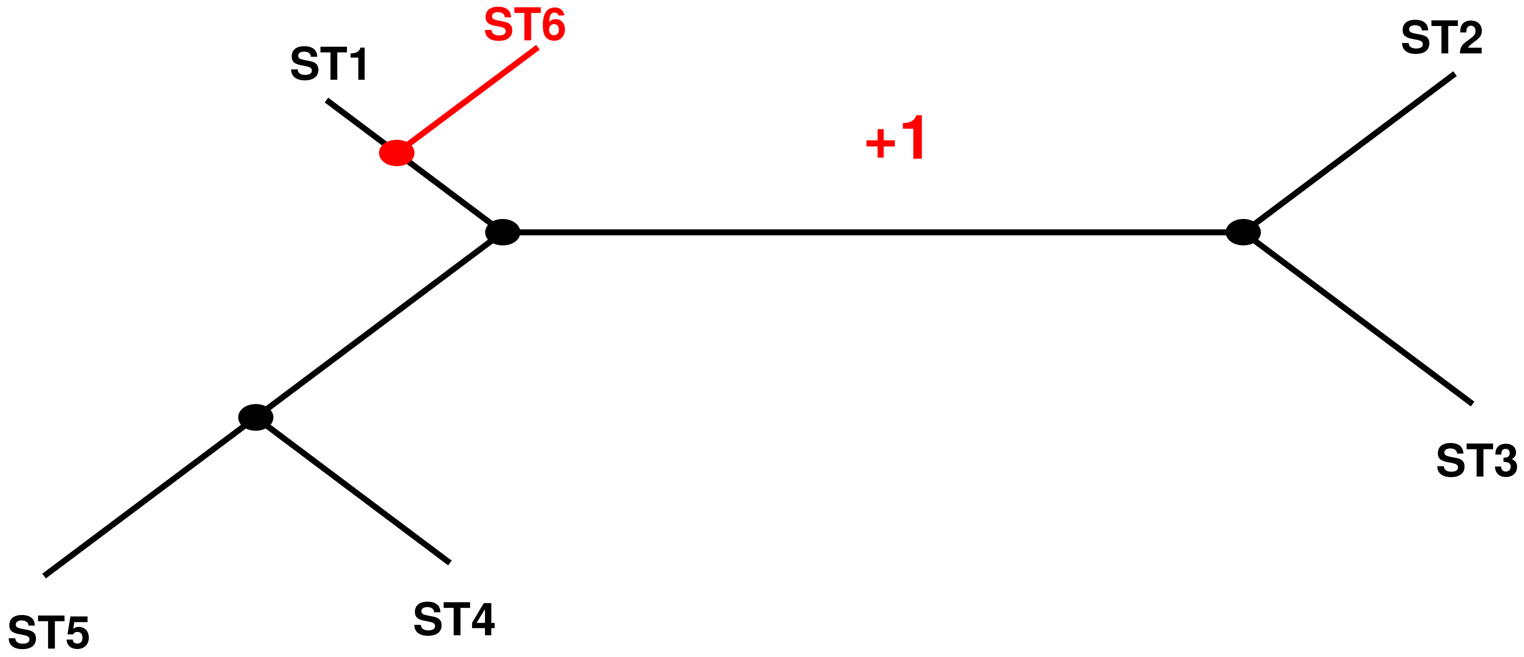
Subtree Pruning & Re-Grafting



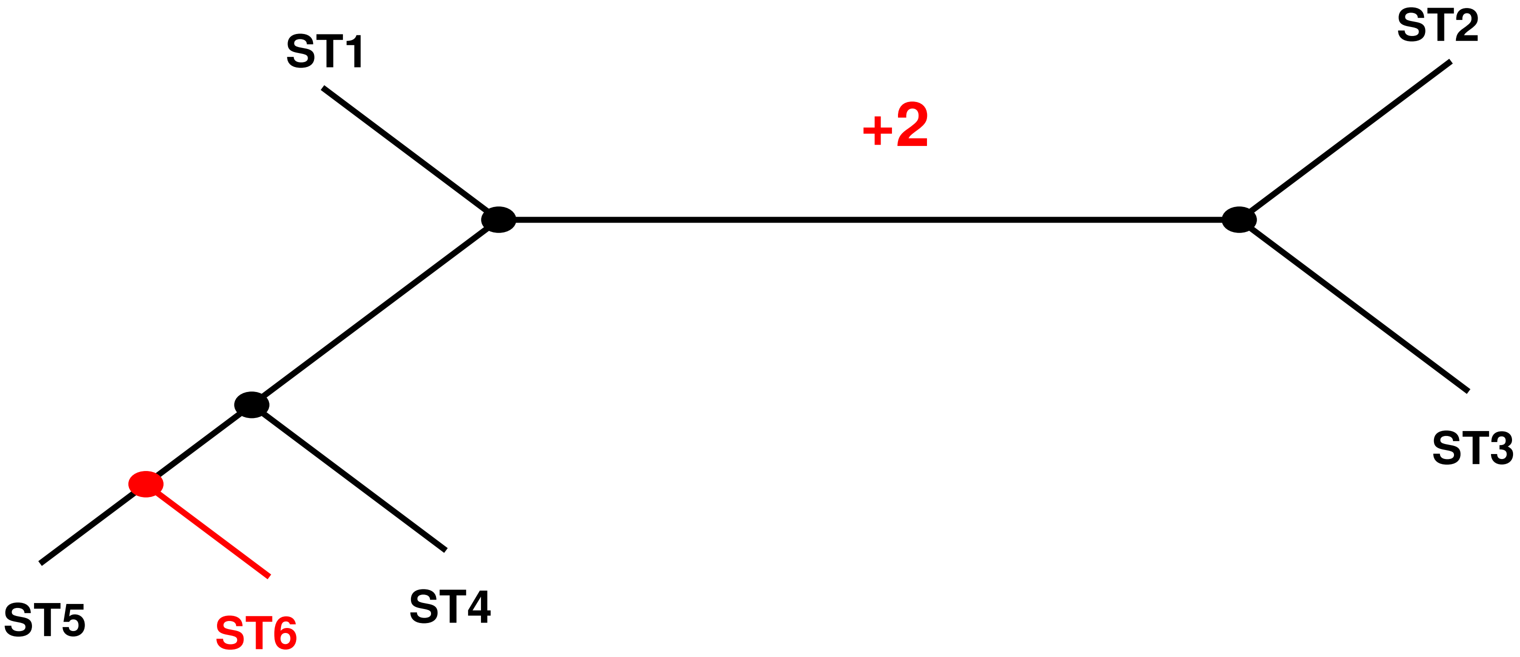
Subtree Pruning & Re-Grafting



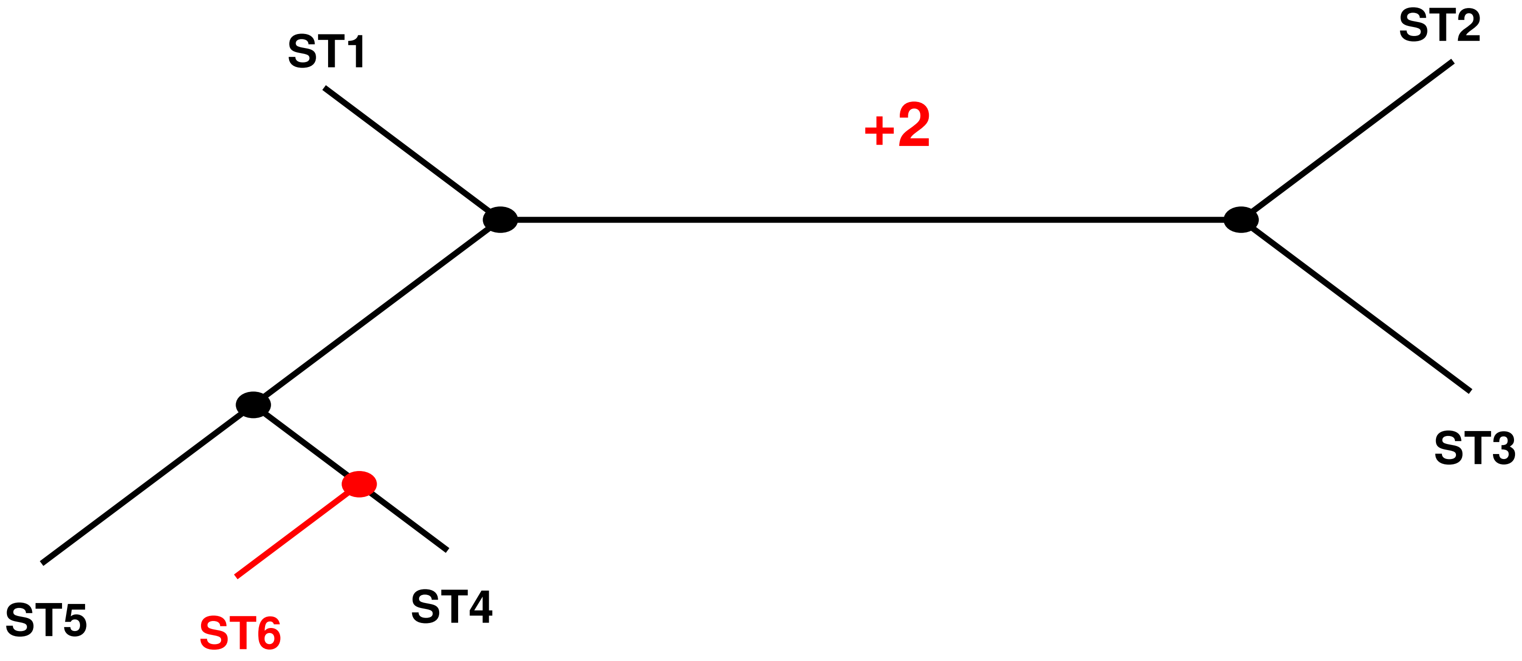
Subtree Pruning & Re-Grafting



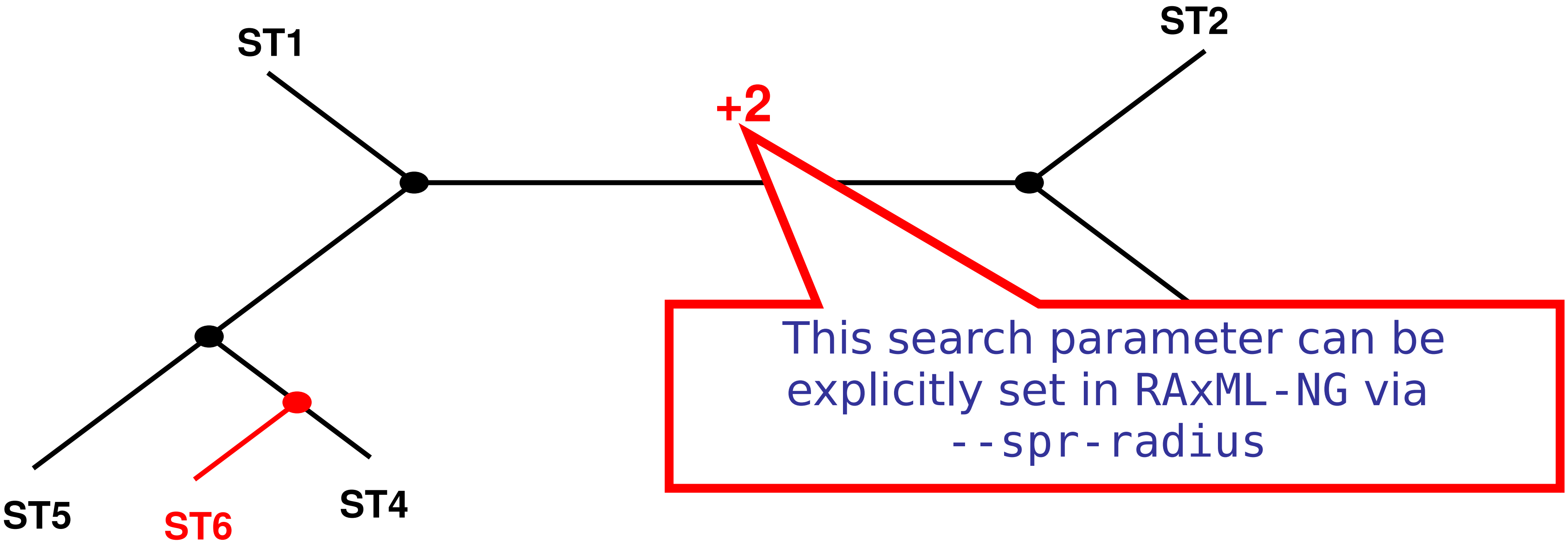
Subtree Pruning & Re-Grafting



Subtree Pruning & Re-Grafting

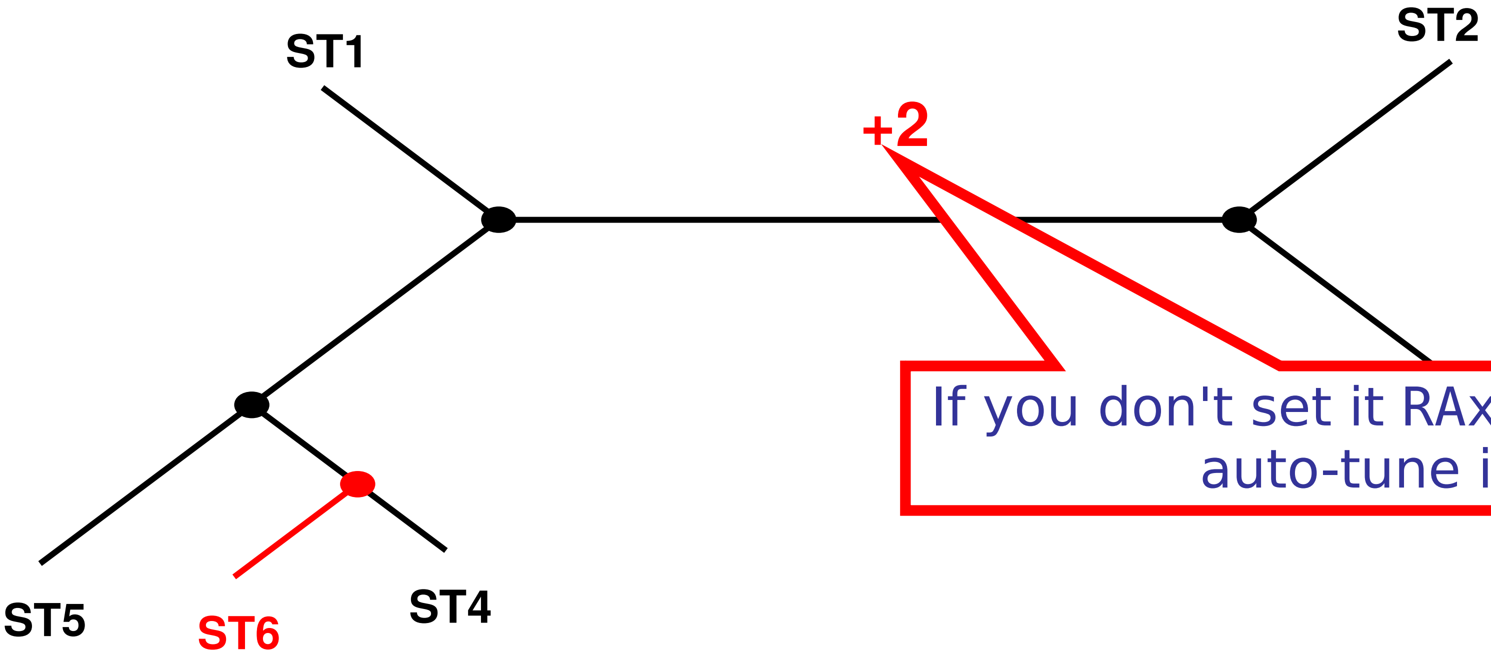


Subtree Pruning & Re-Grafting



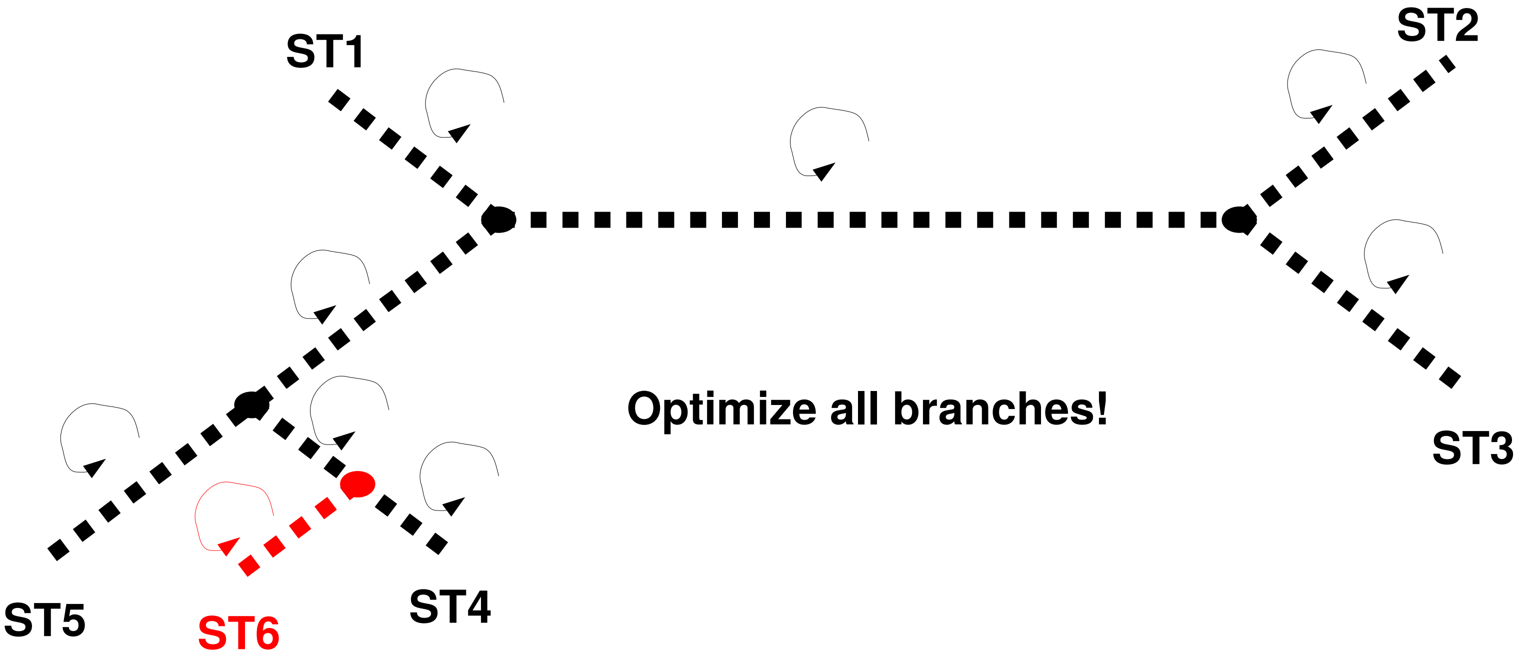
This search parameter can be explicitly set in RAxML-NG via `--spr-radius`

Subtree Pruning & Re-Grafting



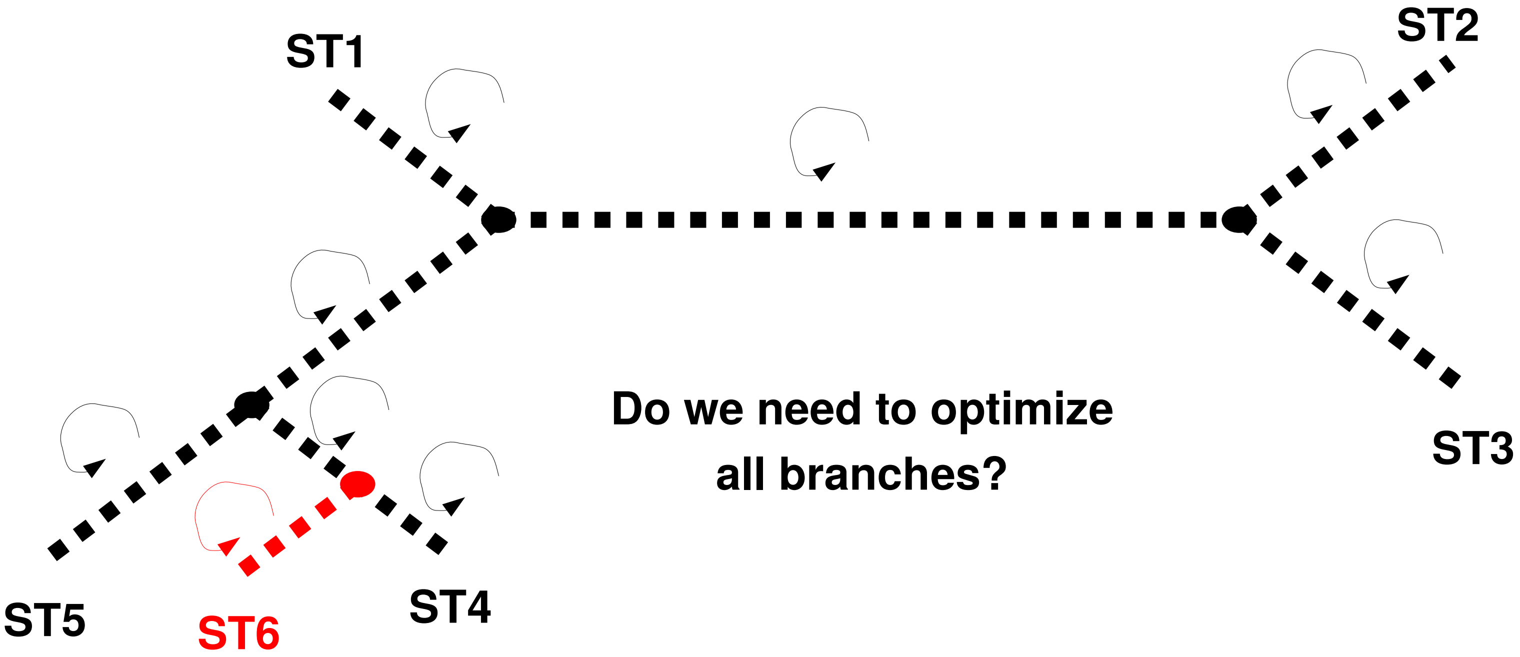
If you don't set it RAxML-NG will auto-tune it

Subtree Pruning & Re-Grafting



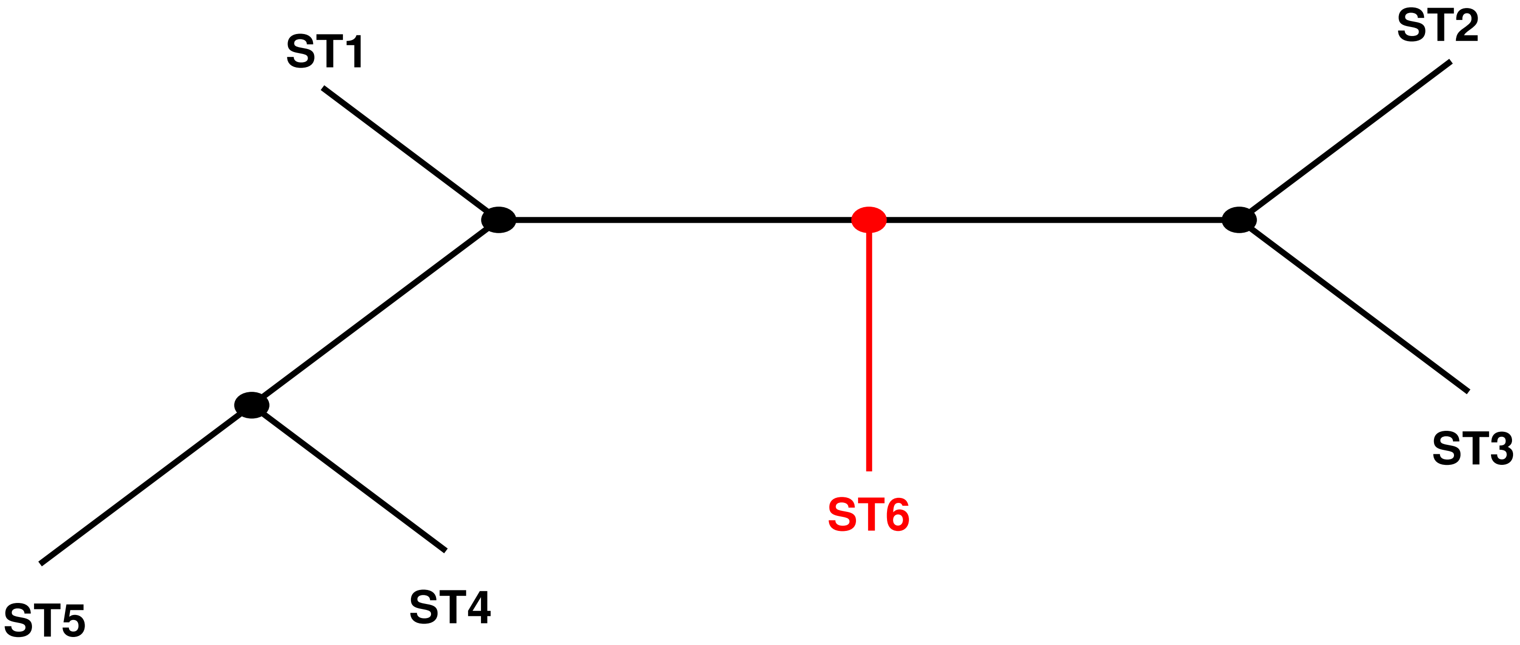
Optimize all branches!

Subtree Pruning & Re-Grafting

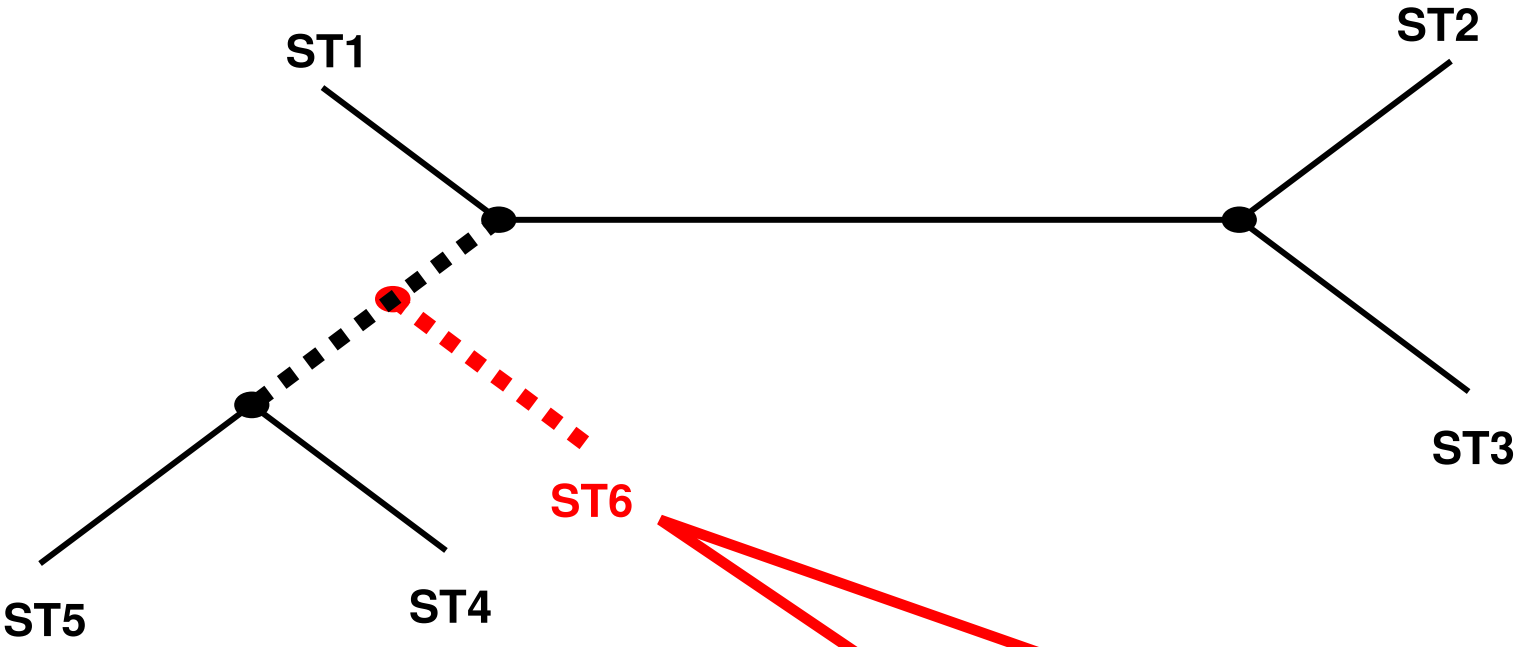


Do we need to optimize all branches?

Lazy Subtree Pruning & Re-Grafting

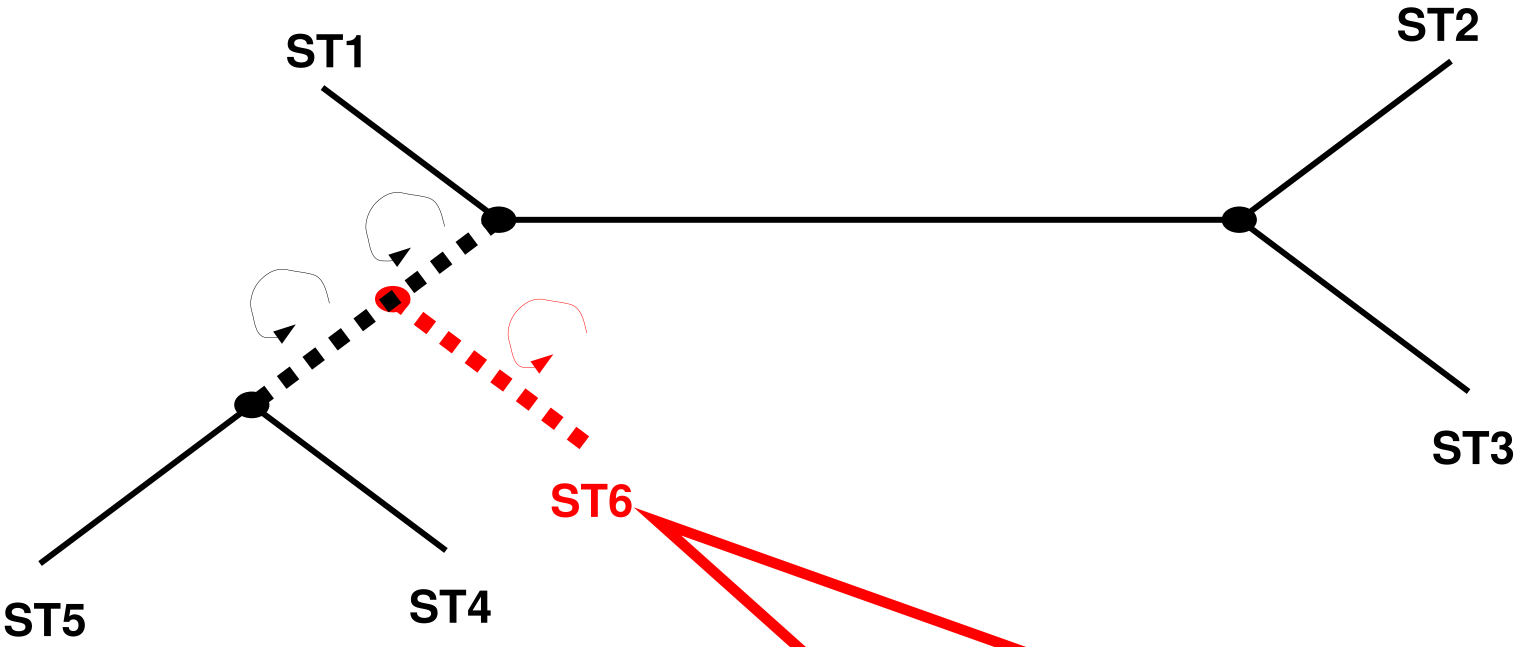


Lazy Subtree Pruning & Re-Grafting



“FAST” SPR round:
no branches are optimized

Lazy Subtree Pruning & Re-Grafting

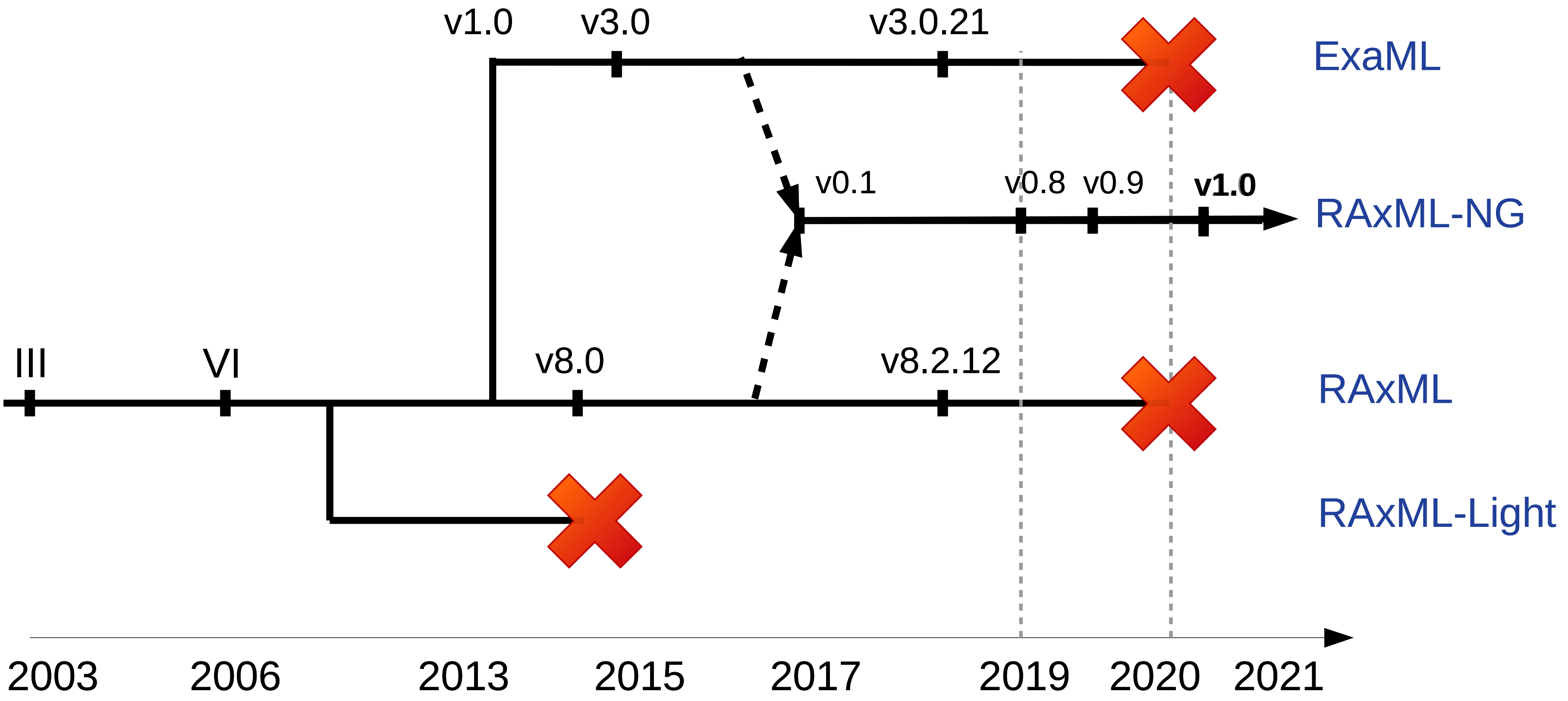


“SLOW” SPR round:
three adjacent branches are optimized

Outline

- Introduction to Phylogenetic Inference - *Alexandros*
- The RAxML Search Algorithm - *Alexandros*
- **Improvements in RAxML Next Generation - *Alexey***
- Tutorial - *Alexey*

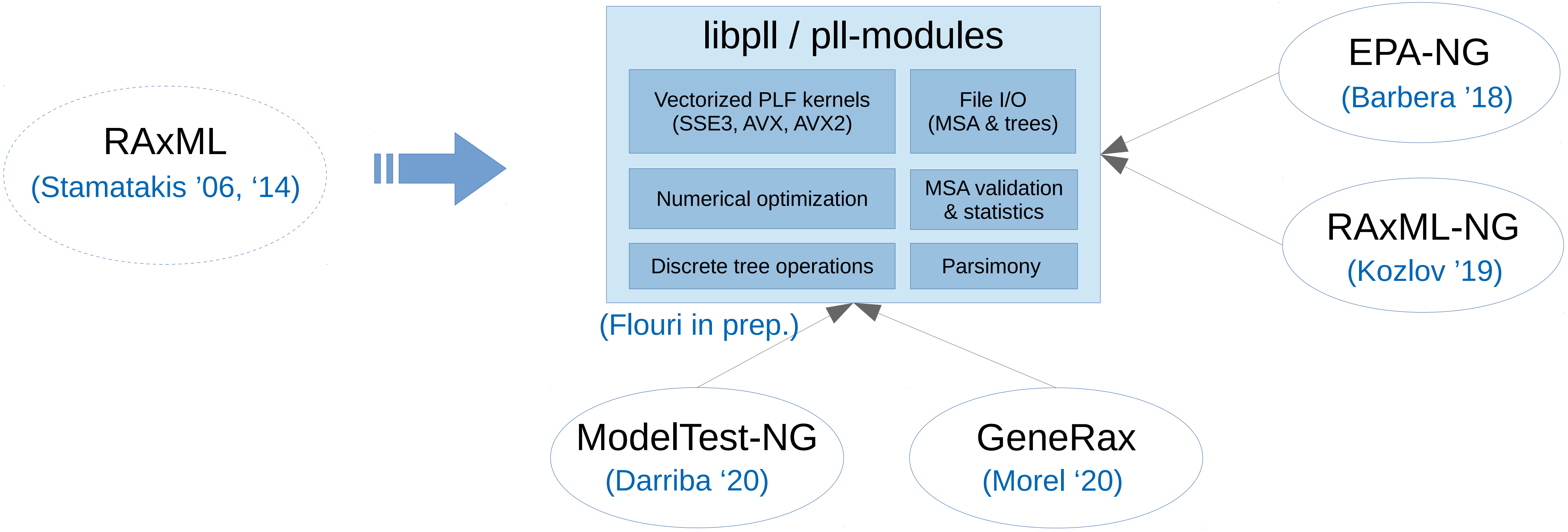
Evolution of RAxML(-NG)



RAxML-NG design goals

- Full rewrite of RAxML
  - Search heuristic largely unchanged (as of v1.0)
- Improve maintainability & enable code reuse
- Eliminate known bugs & bottlenecks
- Improve user experience
  - by default, “do the right thing”

RAxML-NG & family



RAxML
(Stamatakis ’06, ‘14)

libpll / pll-modules

Vectorized PLF kernels
(SSE3, AVX, AVX2)

File I/O
(MSA & trees)

Numerical optimization

MSA validation
& statistics

Discrete tree operations

Parsimony

(Flouri in prep.)

EPA-NG
(Barbera ’18)

RAxML-NG
(Kozlov ’19)

ModelTest-NG
(Darriba ‘20)

GeneRax
(Morel ‘20)

Improvements & new features

- Flexible evolutionary models
  - All “classical” + custom DNA models (eg. DNA010010 = HKY)
  - Per-partition rate heterogeneity (incl. FreeRate)
  - Proportional branch lengths
- Phylogenetic terrace detection (Biczok ‘17)
- Transfer bootstrap support metric (Lemoine ‘18)
- Energy monitoring

Performance & scalability

- Checkpointing
- Advanced load balancing
- Binary alignment format

from ExaML

- “Site repeats” optimization (Kobert ‘17)
  - 10-60% speedup + memory savings
- Flexible and user-friendly parallelization

Parallelization: hardware

raxmlHPC

raxmlHPC-SSE

raxmlHPC-AVX

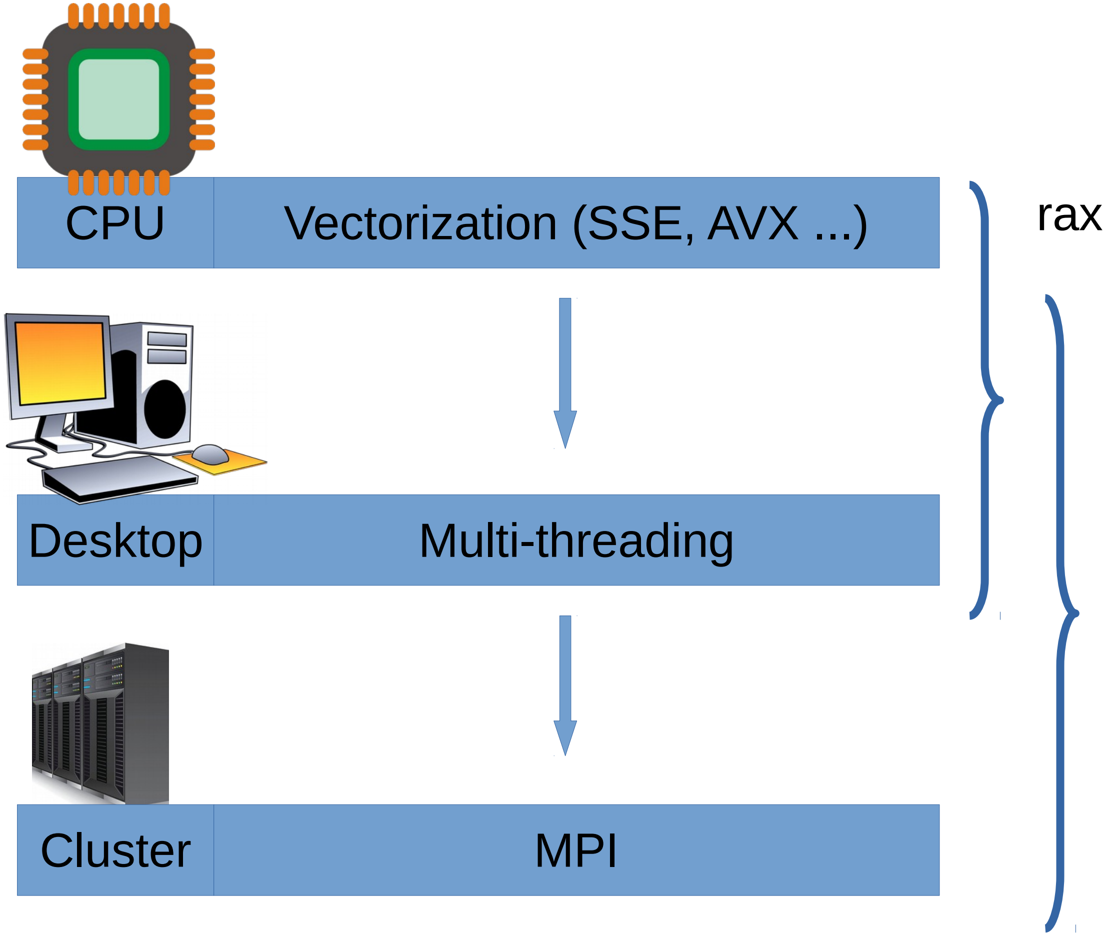
raxmlHPC-PTHREADS

raxmlHPC-PTHREADS-SSE

raxmlHPC-HYBRID

raxmlHPC-MPI

raxmlHPC-MPI-AVX2



CPU | Vectorization (SSE, AVX ...)

Desktop | Multi-threading

Cluster | MPI

raxml-ng

raxml-ng-mpi

Parallelization: software

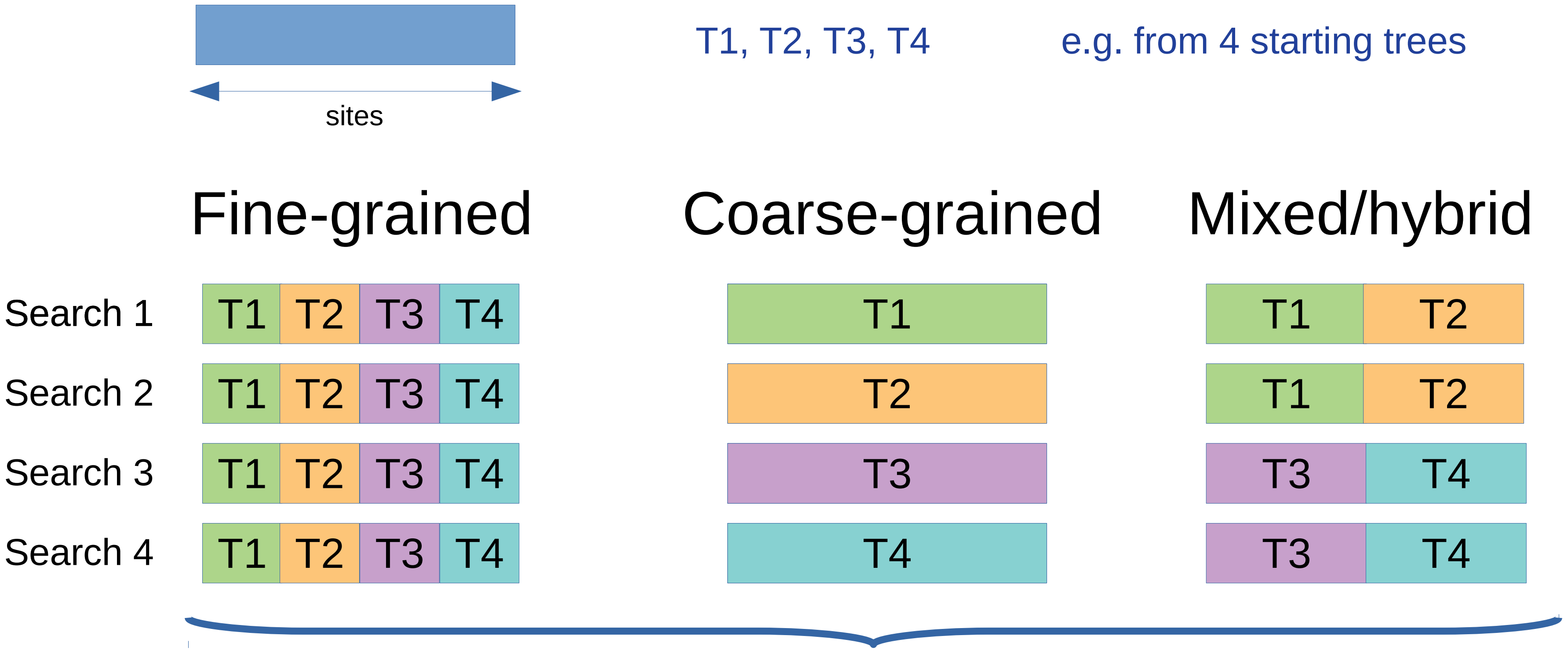
Alignment

4 threads

4 searches

T1, T2, T3, T4

e.g. from 4 starting trees



New in v1.0: Full native support and automatic configuration!

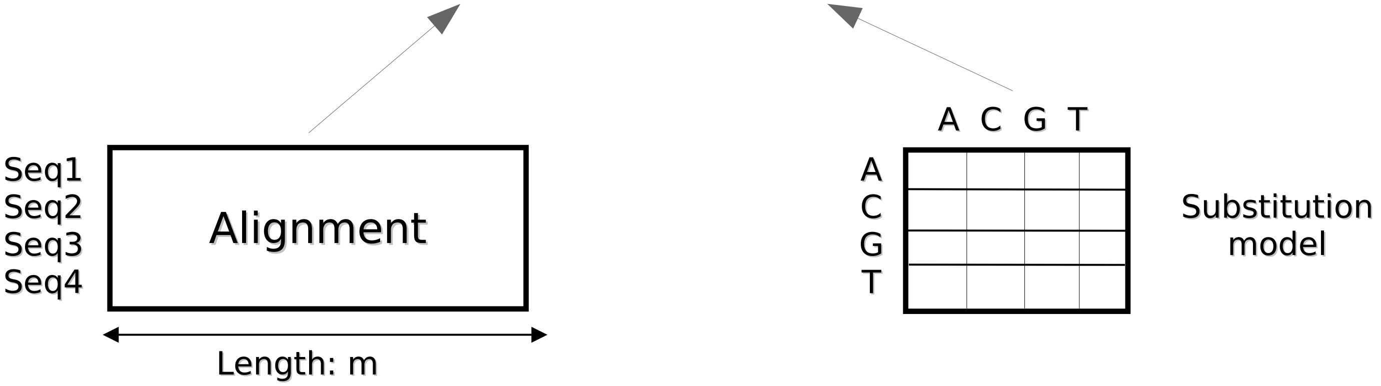
Outline

- Introduction to Phylogenetic Inference - *Alexandros*
- The RAxML Search Algorithm - *Alexandros*
- Improvements in RAxML Next Generation - *Alexey*
- **Tutorial - *Alexey***

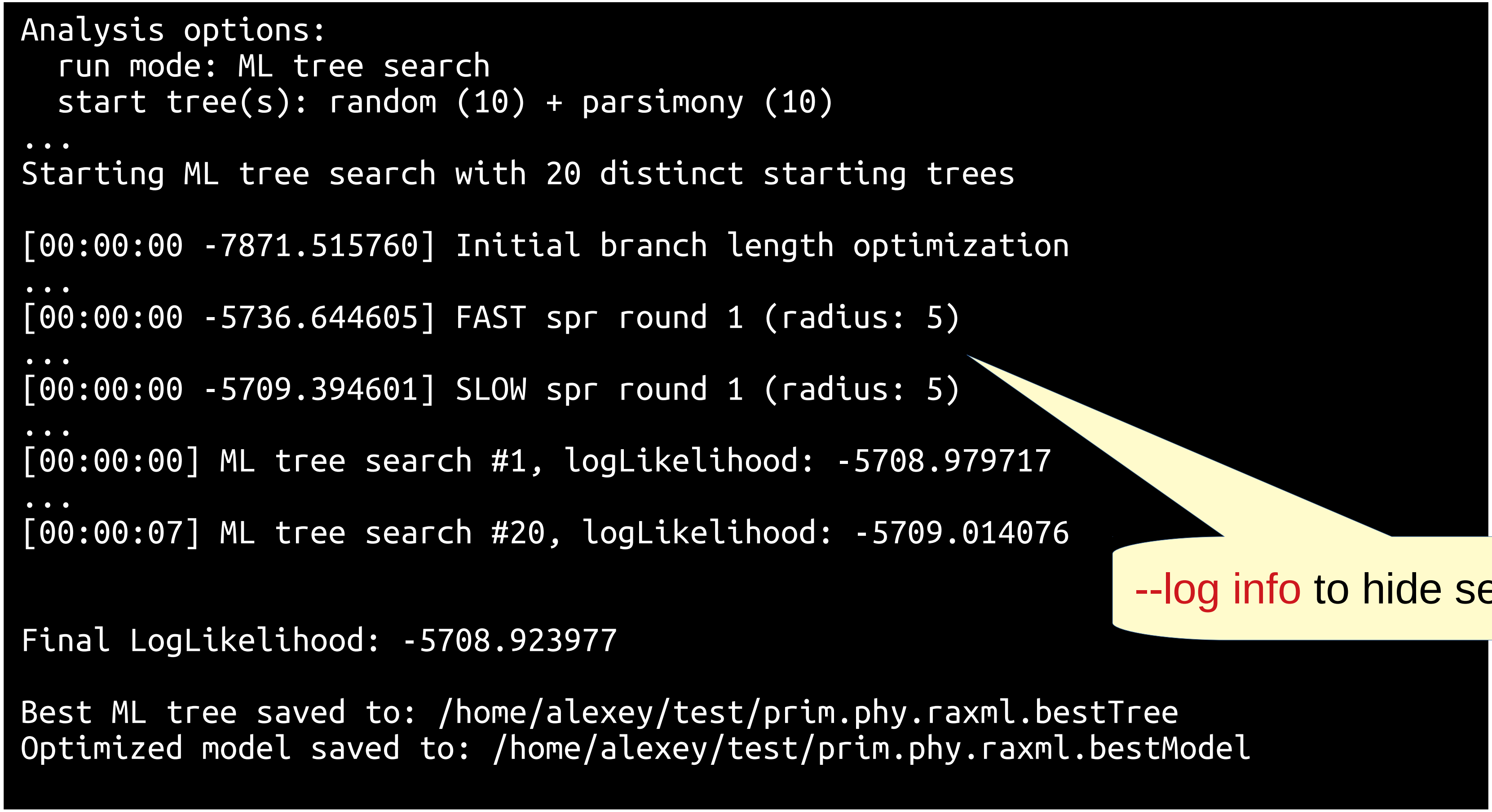
Quick start: ML tree search

- Default command: --search
  - 20 starting trees (10 random + 10 parsimony)
  - Pick the best-scoring one

```
$ raxml-ng --msa prim.phy --model GTR+G
```



ML tree search: Output



```
Analysis options:
  run mode: ML tree search
  start tree(s): random (10) + parsimony (10)
...
Starting ML tree search with 20 distinct starting trees

[00:00:00 -7871.515760] Initial branch length optimization
...
[00:00:00 -5736.644605] FAST spr round 1 (radius: 5)
...
[00:00:00 -5709.394601] SLOW spr round 1 (radius: 5)
...
[00:00:00] ML tree search #1, logLikelihood: -5708.979717
...
[00:00:07] ML tree search #20, logLikelihood: -5709.014076


Final LogLikelihood: -5708.923977

Best ML tree saved to: /home/alexey/test/prim.phy.raxml.bestTree
Optimized model saved to: /home/alexey/test/prim.phy.raxml.bestModel
```

--log info to hide search progress

Tree with support values

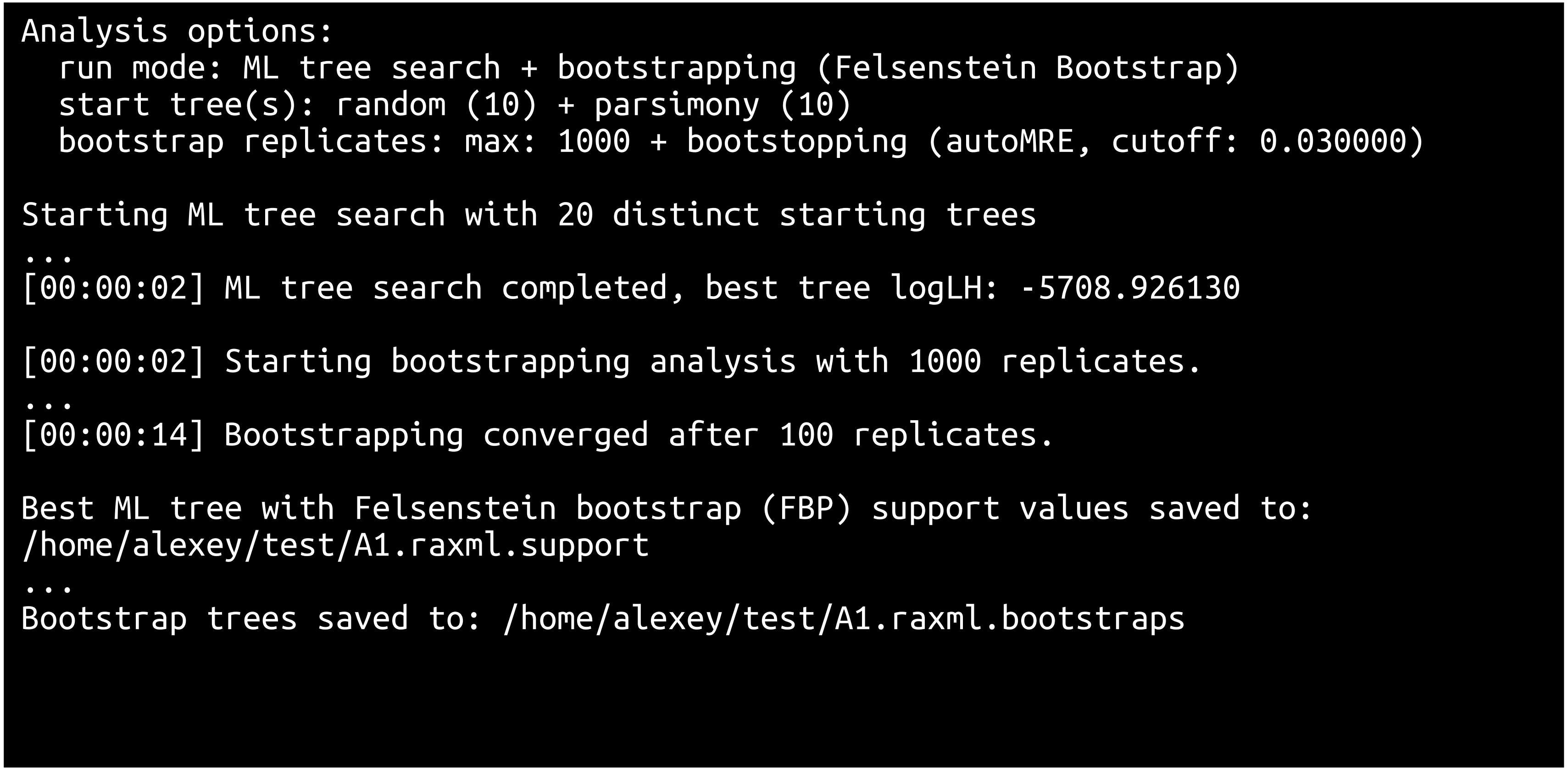
- All-in-one mode: --all
  - ML tree search (as before)
  - Bootstrapping with autoMRE convergence test
  - Compute support values + map on ML tree

```
$ raxml-ng --all --msa prim.phy --model GTR+G --prefix A1 --seed 1
```

Output files: A1.raxml.*

Fixed RNG seed
for better reproducibility

Tree with support values: Output



```
Analysis options:
  run mode: ML tree search + bootstrapping (Felsenstein Bootstrap)
  start tree(s): random (10) + parsimony (10)
  bootstrap replicates: max: 1000 + bootstopping (autoMRE, cutoff: 0.030000)

Starting ML tree search with 20 distinct starting trees
...
[00:00:02] ML tree search completed, best tree logLH: -5708.926130

[00:00:02] Starting bootstrapping analysis with 1000 replicates.
...
[00:00:14] Bootstrapping converged after 100 replicates.

Best ML tree with Felsenstein bootstrap (FBP) support values saved to:
/home/alexey/test/A1.raxml.support
...
Bootstrap trees saved to: /home/alexey/test/A1.raxml.bootstraps
```

Customize analysis

- Starting trees: --tree

```
--tree rand{1}
```

```
--tree pars{50},rand{50}
```

```
--tree user.nw
```

- Bootstrap replicates: --bs-trees

```
--bs-trees 100
```

```
--bs-trees autoMRE{500}
```

```
--bs-trees bs.bw
```

- Branch length linkage mode: --brlen

```
--brlen linked
```

```
--brlen scaled
```

```
--brlen unlinked
```

- Branch length limits: --blmin / --blmax

```
--blmin 1e-9
```

```
--blmax 10
```

Tree likelihood evaluation

- Optimize free model parameters and branch lengths on a fixed topology: --evaluate

```
$ raxml-ng --evaluate --msa prim.phy --tree A1.raxml.bestTree --model GTR+G --prefix E1
```

```
Evaluating 1 trees

[00:00:00] Tree #1, initial LogLikelihood: -6419.996676

[00:00:00 -6419.996676] Initial branch length optimization
[00:00:00 -6276.983451] Model parameter optimization (eps = 0.100000)

[00:00:00] Tree #1, final logLikelihood: -5709.005148
...
Best ML tree saved to: /home/alexey/test/E1.raxml.bestTree
Optimized model saved to: /home/alexey/test/E1.raxml.bestModel
```

Tree likelihood evaluation (2)

- Compute and print tree log-likelihood: --loglh
  - No branch length optimization
  - No model optimization
  - No output files created

```
$ raxml-ng --loglh --msa prim.phy --tree A1.raxml.bestTree --model GTR{1/2/3/4/5/6}+G{0.5}

Final LogLikelihood: -6334.023267
```

Check & parse

- Check alignment for format errors: --check

```
$ raxml-ng --check --msa prim.phy –model GTR+G
```

- Compress alignment into binary file: --parse

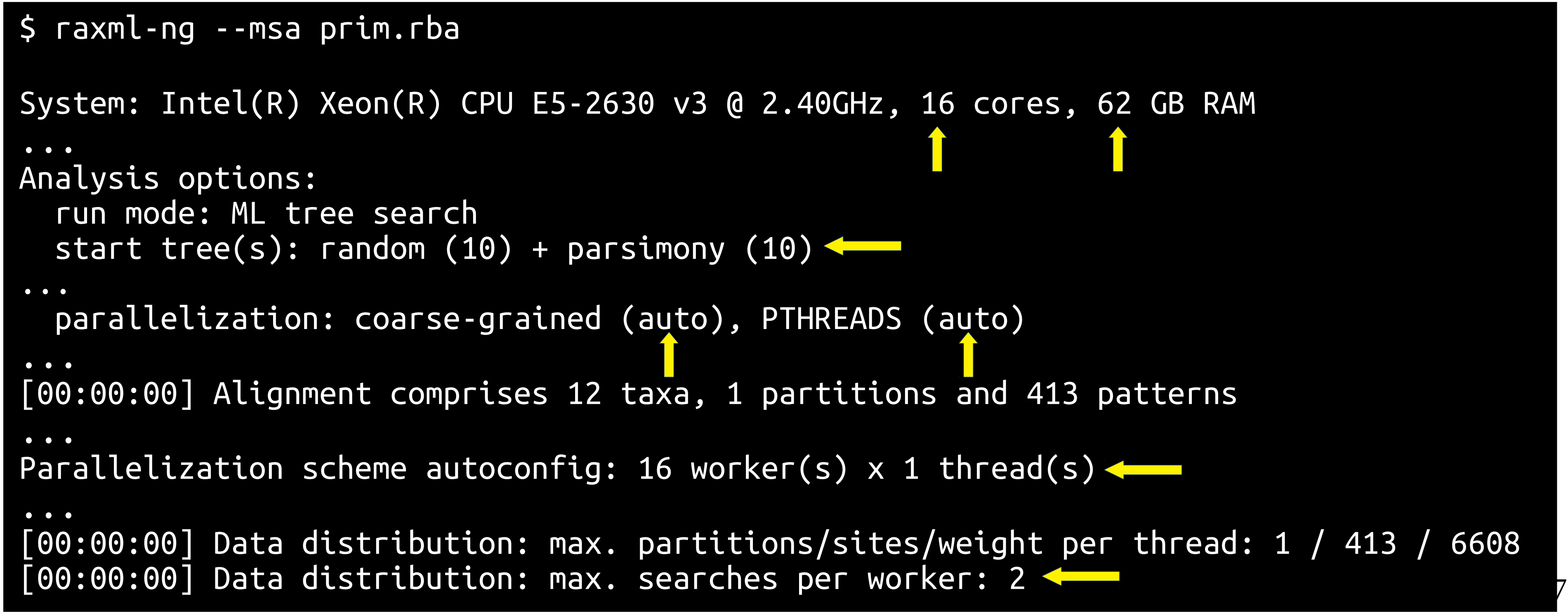
```
$ raxml-ng --parse --msa prim.phy --model GTR+G --prefix prim
```

- ...which can be then used in parallel jobs

```
$ raxml-ng --search --msa prim.raxml.rba --prefix S1
```

Parallelization tuning

- Fully automatic (default) → heuristic-based



```
$ raxml-ng --msa prim.rba

System: Intel(R) Xeon(R) CPU E5-2630 v3 @ 2.40GHz, 16 cores, 62 GB RAM
...
Analysis options:
  run mode: ML tree search
  start tree(s): random (10) + parsimony (10)
...
  parallelization: coarse-grained (auto), PTHREADS (auto)
...
[00:00:00] Alignment comprises 12 taxa, 1 partitions and 413 patterns
...
Parallelization scheme autoconfig: 16 worker(s) x 1 thread(s)
...
[00:00:00] Data distribution: max. partitions/sites/weight per thread: 1 / 413 / 6608
[00:00:00] Data distribution: max. searches per worker: 2
```

Parallelization tuning

- Automatic with upper limits

```
$ raxml-ng --msa prim.rba --threads auto{16} --workers auto{2}
```

- Manual

```
$ raxml-ng --msa prim.rba --threads 16 --workers 2
```

- Also works with MPI

```
$ mpirun -n 4 raxml-ng-mpi --msa prim.rba --threads 16 --workers 8
```

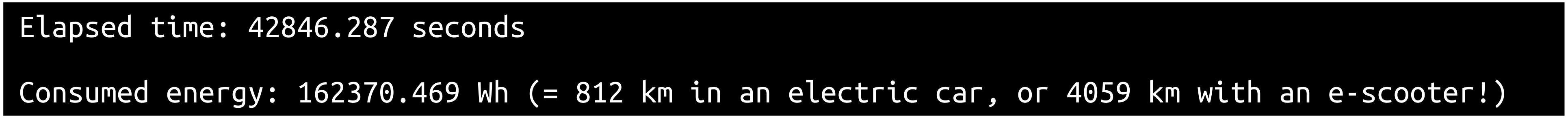
4 ranks * 16 threads = 64 = 8 workers * 8 threads

Energy monitoring

- New in RAxML-NG v1.0: energy usage report
  - Measured with Intel RAPL → CPU+DRAM only
  - Supported on Linux systems only
  - To disable, add: --extra energy-off

Energy monitoring

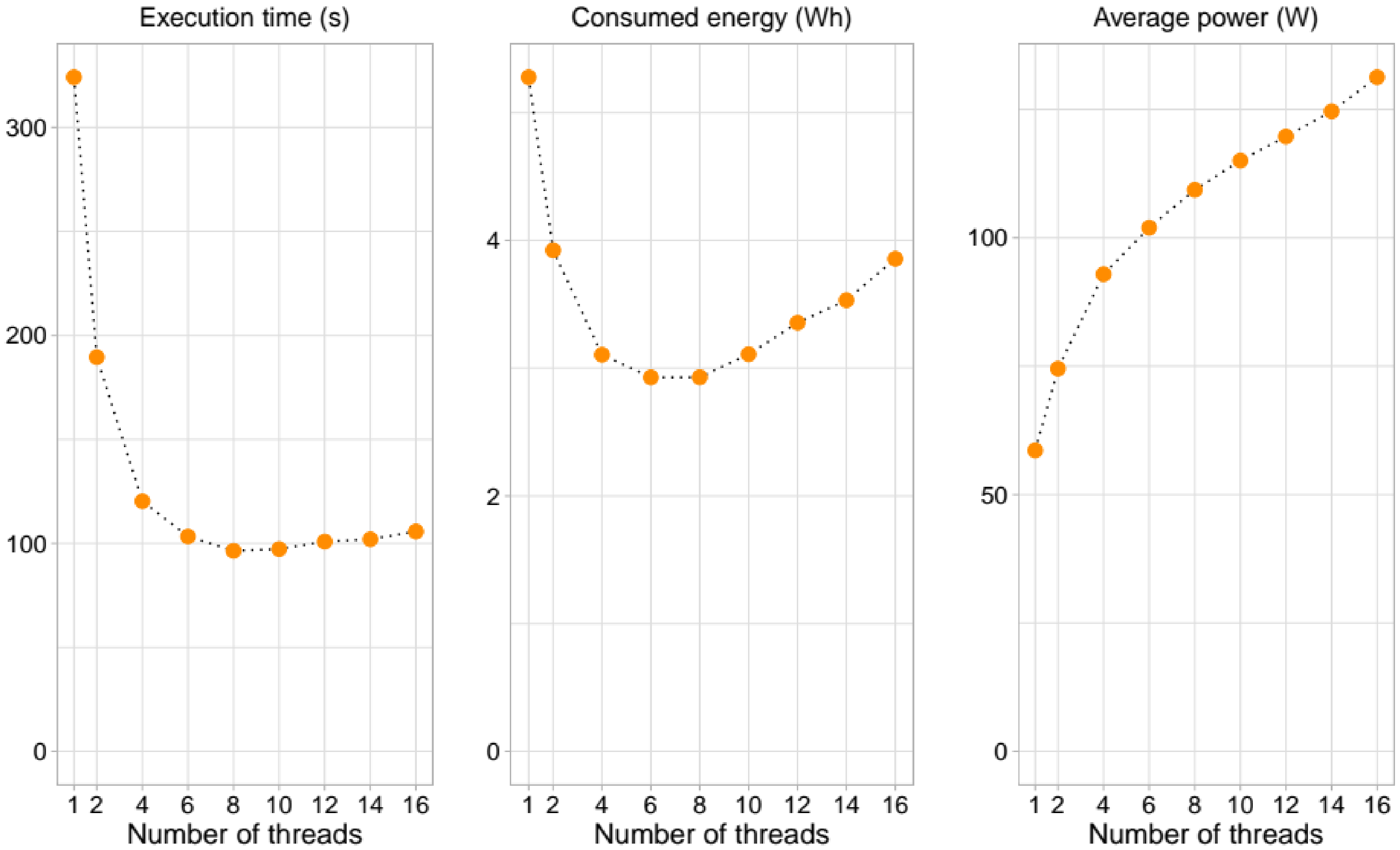
- New in RAxML-NG v1.0: energy usage report
  - Measured with Intel RAPL → CPU+DRAM only
  - Supported on Linux systems only
  - To disable, add: --extra energy-off



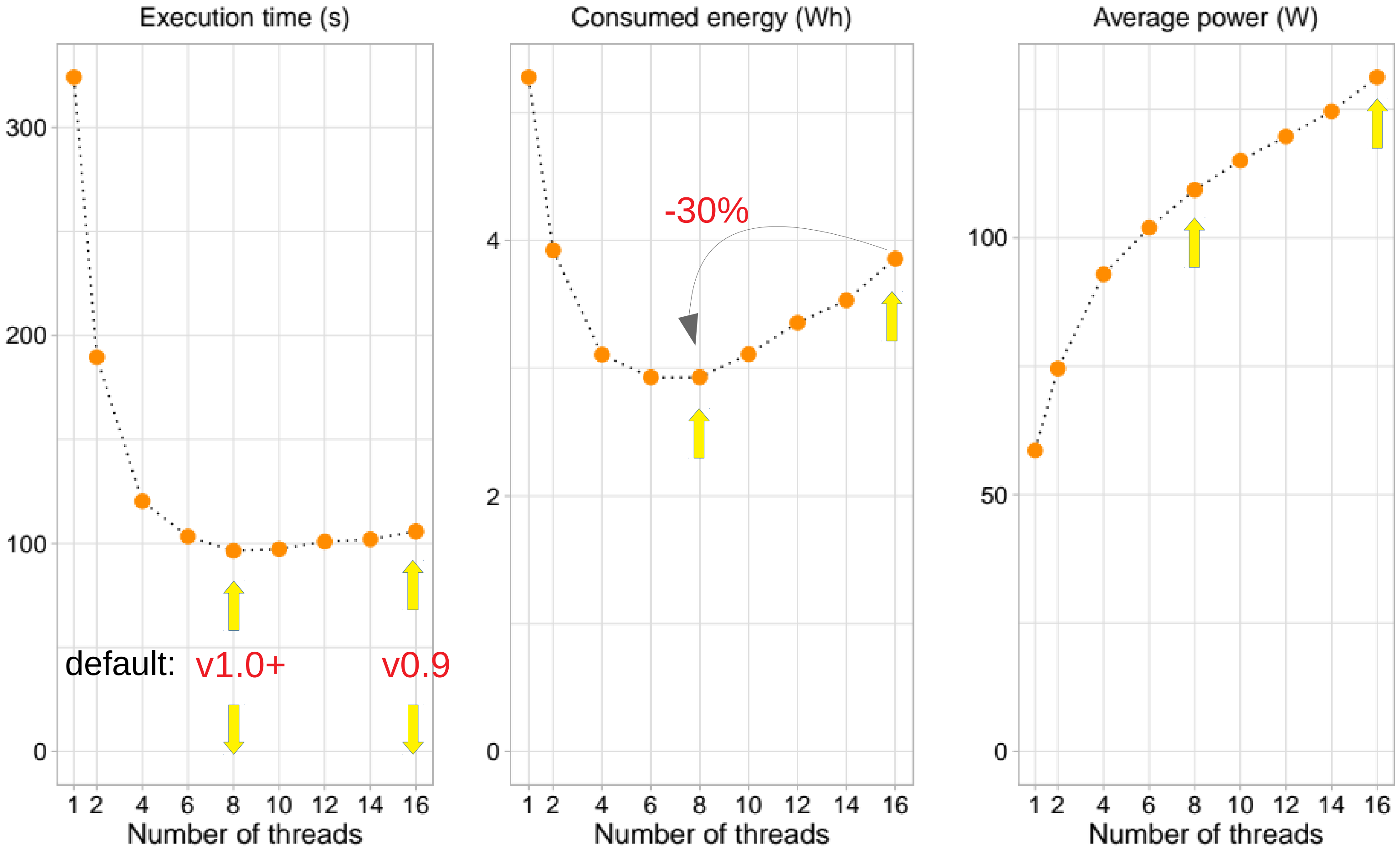
```
Elapsed time: 42846.287 seconds

Consumed energy: 162370.469 Wh (= 812 km in an electric car, or 4059 km with an e-scooter!)
```

Single tree search (96 nodes x 12h): >160 kWh
My apartment per month: ~100 kWh

You can't improve what you can't measure!



You can't improve what you can't measure!



RAxML-NG availability

- Web services
  - Vital-IT: https://raxml-ng.vital-it.ch/#/ → free, for small datasets
  - CIPRES: http://www.phylo.org/ → registration required
- Source+binaries for Linux & macOS
  - GitHub: https://github.com/amkozlov/raxml-ng
- Conda: https://anaconda.org/bioconda/raxml-ng
- GUI: https://github.com/AntonelliLab/raxmlGUI

Where to get help?

- Documentation

  https://github.com/amkozlov/raxml-ng/wiki

- Tutorial

  https://github.com/amkozlov/raxml-ng/wiki/Tutorial

- User support group

  https://groups.google.com/forum/#!forum/raxml

Q & A

References

- Barbera et al. (2018) **EPA-ng: Massively Parallel Evolutionary Placement of Genetic Sequences.** *Systematic Biology*, syy054, https://doi.org/10.1093/sysbio/syy054
- Biczok et al. (2017) **Two C++ libraries for counting trees on a phylogenetic terrace.** Bioinformatics. https://doi.org/10.1093/bioinformatics/bty384
- Kobert et al. (2017) Efficient detection of repeating sites to accelerate phylogenetic likelihood calculations. *Syst. Biol* *https://doi.org/10.1093/sysbio/syw075*
- Kozlov et al. (2019) **RAxML-NG: A fast, scalable, and user-friendly tool for maximum likelihood phylogenetic inference**. Bioinformatics*.* https://doi.org/10.1093/bioinformatics/btz305
- Kozlov, Aberer, Stamatakis (2015) **ExaML version 3: a tool for phylogenomic analyses on supercomputers**. *Bioinformatics*. https://doi.org/10.1093/bioinformatics/btv184
- Lemoine et al. (2018) **Renewing Felsensteinen phylogenetic bootstrap in the era of big data.** *Nature.* *https://doi.org/10.1038/s41586-018-0043-0*
- Morel, Kozlov, Stamatakis (2019) **ParGenes: a tool for massively parallel model selection and phylogenetic tree inference on thousands of genes.** *Bioinformatics.* https://doi.org/10.1093/bioinformatics/bty839
- Morel et al (2020). **GeneRax: A tool for species tree-aware maximum likelihood based gene tree inference under gene duplication, transfer, and loss**. *MBE*. https://doi.org/10.1093/molbev/msaa141
- Stamatakis (2006) **RAxML-VI-HPC: maximum likelihood-based phylogenetic analyses with thousands of taxa and mixed models**. *Bioinformatics*, https://doi.org/10.1093/bioinformatics/btl446
- Stamatakis (2014) **RAxML version 8: a tool for phylogenetic analysis and post-analysis of large phylogenies.** *Bioinformatics*, https://doi.org/10.1093/bioinformatics/btu033
- Stamatakis and Aberer (2013) **Novel parallelization schemes for large-scale likelihood-based phylogenetic inference.** *In Parallel Distributed Processing (IPDPS)* https://doi.org/10.1109/IPDPS.2013.70