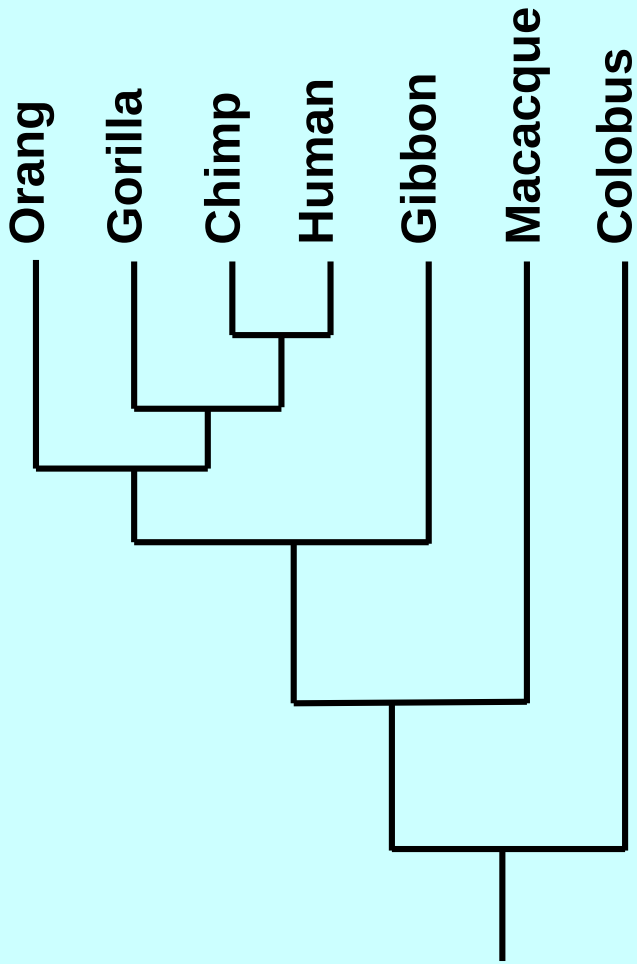
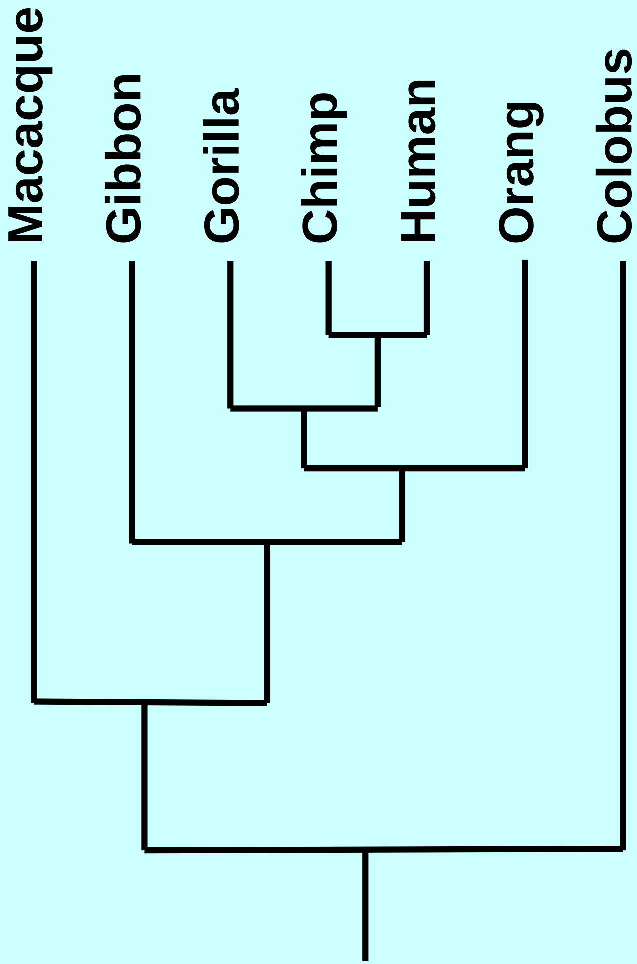
Lecture 2. Tree space and searching tree space

Joe Felsenstein

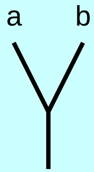
Department of Genome Sciences and Department of Biology

The same tree?



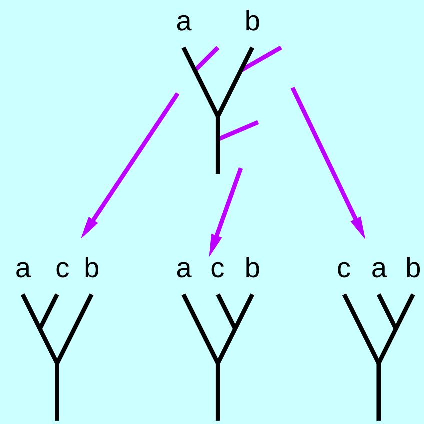


All possible trees



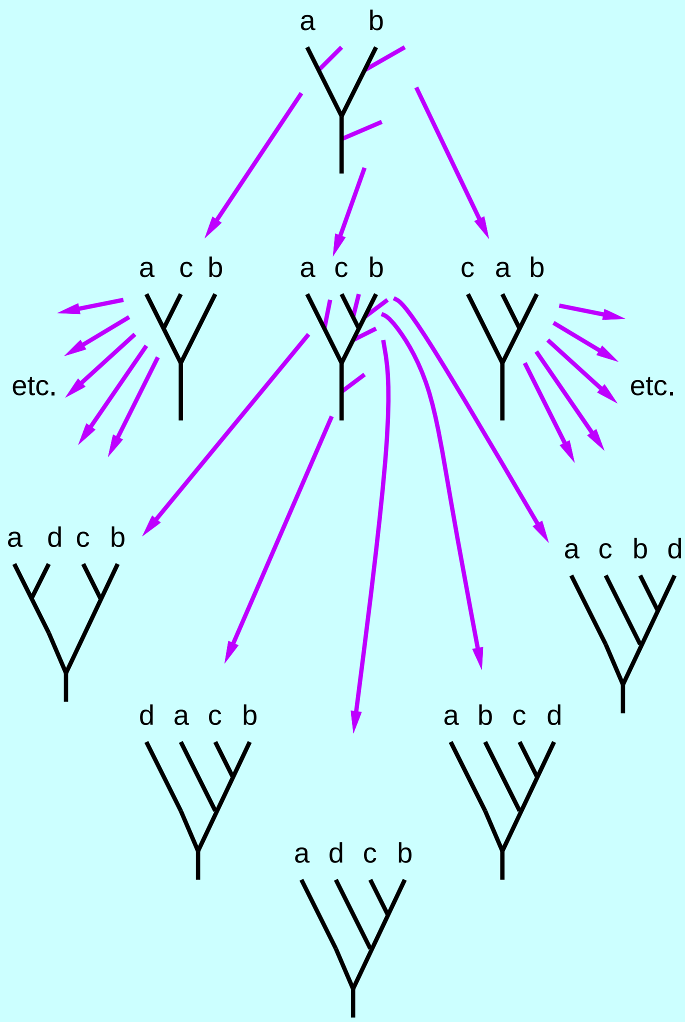
Forming all 4-species trees by adding the next species in all possible places

All possible trees



Forming all 4-species trees by adding the next species in all possible places

All possible trees



Forming all 4-species trees by adding the next species in all possible places

The number of rooted bifurcating trees:

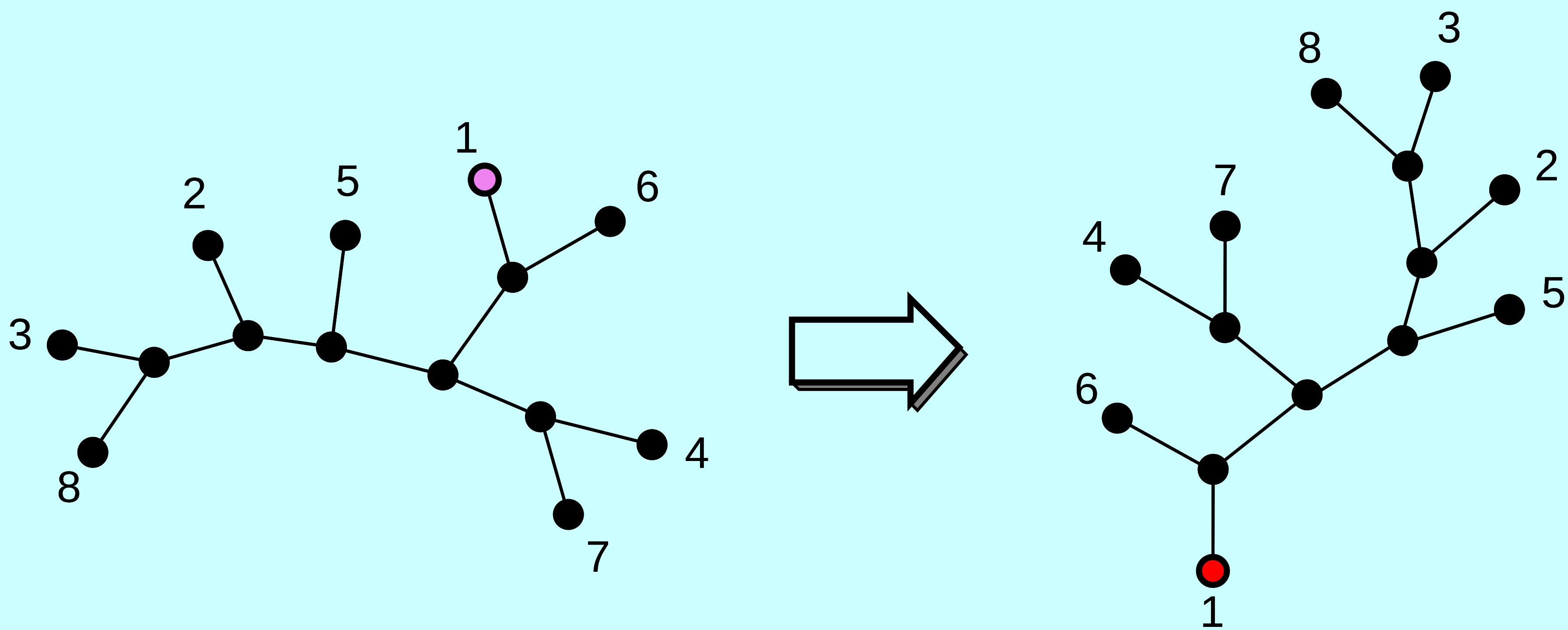
$$1 \times 3 \times 5 \times 7 \times \ldots \times (2n-3)$$

$$= \; (2n-3)!/\left((n-2)!\, 2^{n-2}\right)$$

which is:

species	number of trees
1	1
2	1
3	3
4	15
5	105
6	945
7	10,395
8	135,135
9	2,027,025
10	34,459,425
11	654,729,075
12	13,749,310,575
13	316,234,143,225
14	7,905,853,580,625
15	213,458,046,676,875
16	6,190,283,353,629,375
17	191,898,783,962,510,625
18	6,332,659,870,762,850,625
19	221,643,095,476,699,771,875
20	8,200,794,532,637,891,559,375
30	4.9518 $\times 10^{38}$
40	1.00985 $\times 10^{57}$
50	2.75292 $\times 10^{76}$

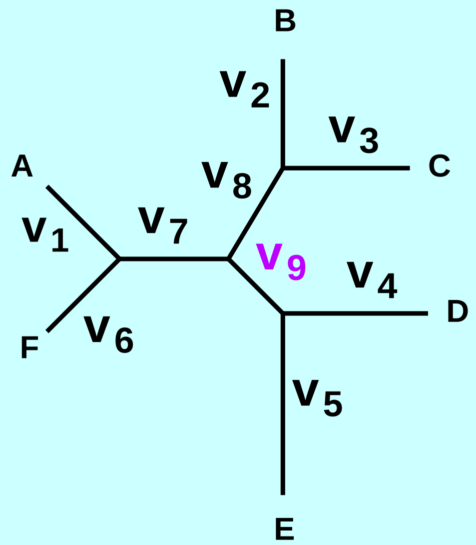
Mapping an unrooted tree into a rooted tree

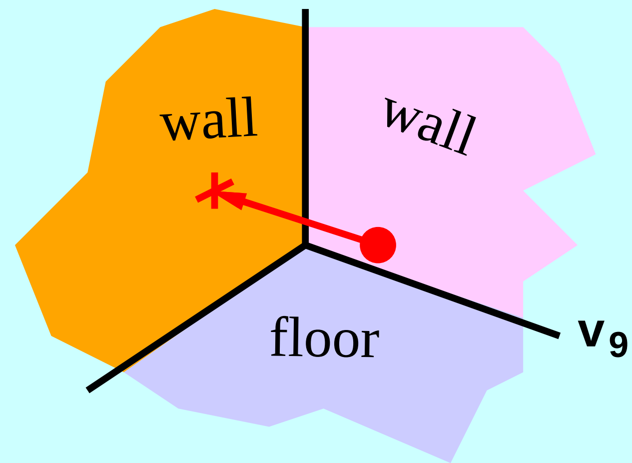


... one with one fewer species.

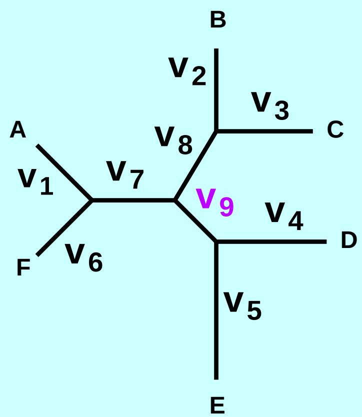
For one tree topology

The space of trees varying all $2n-3$ branch lengths, each a nonegative number, defines an “orthant" (open corner) of a $(2n-3)$-dimensional real space:

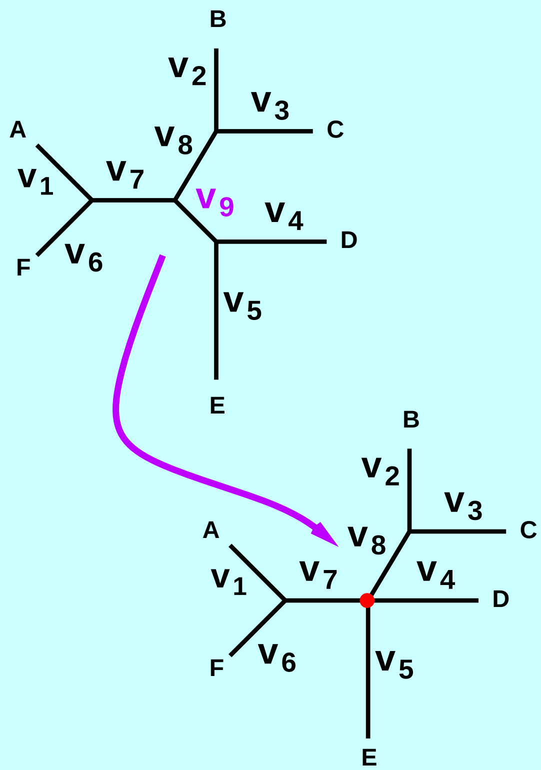




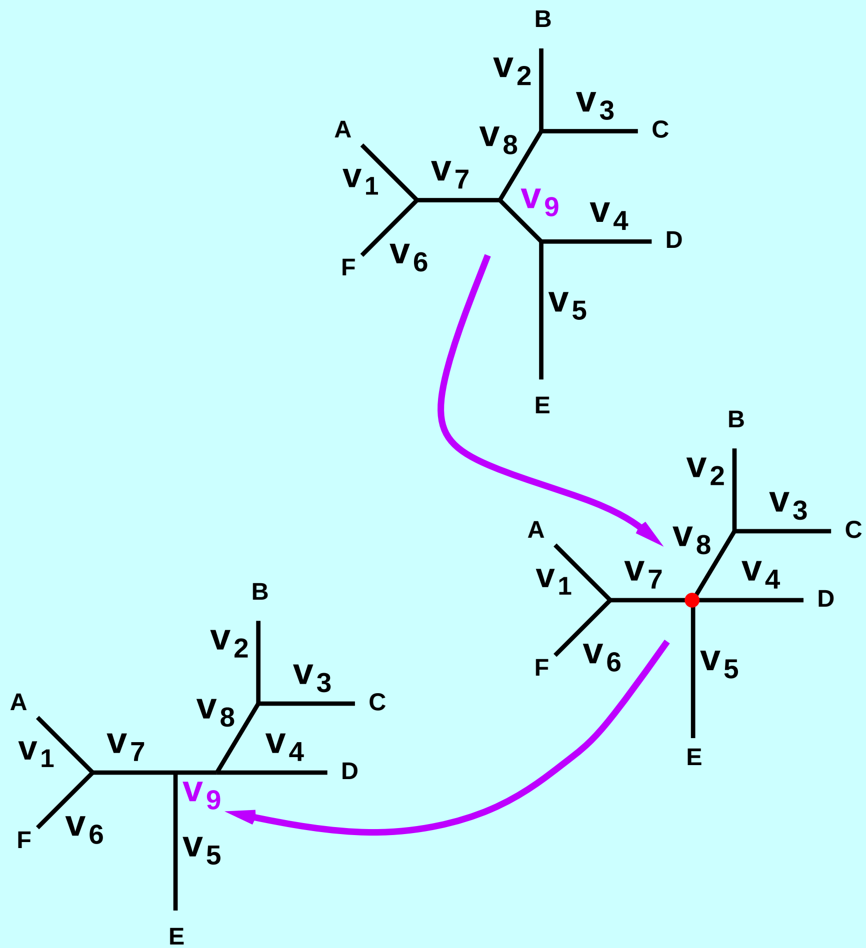
Through the looking glass ...



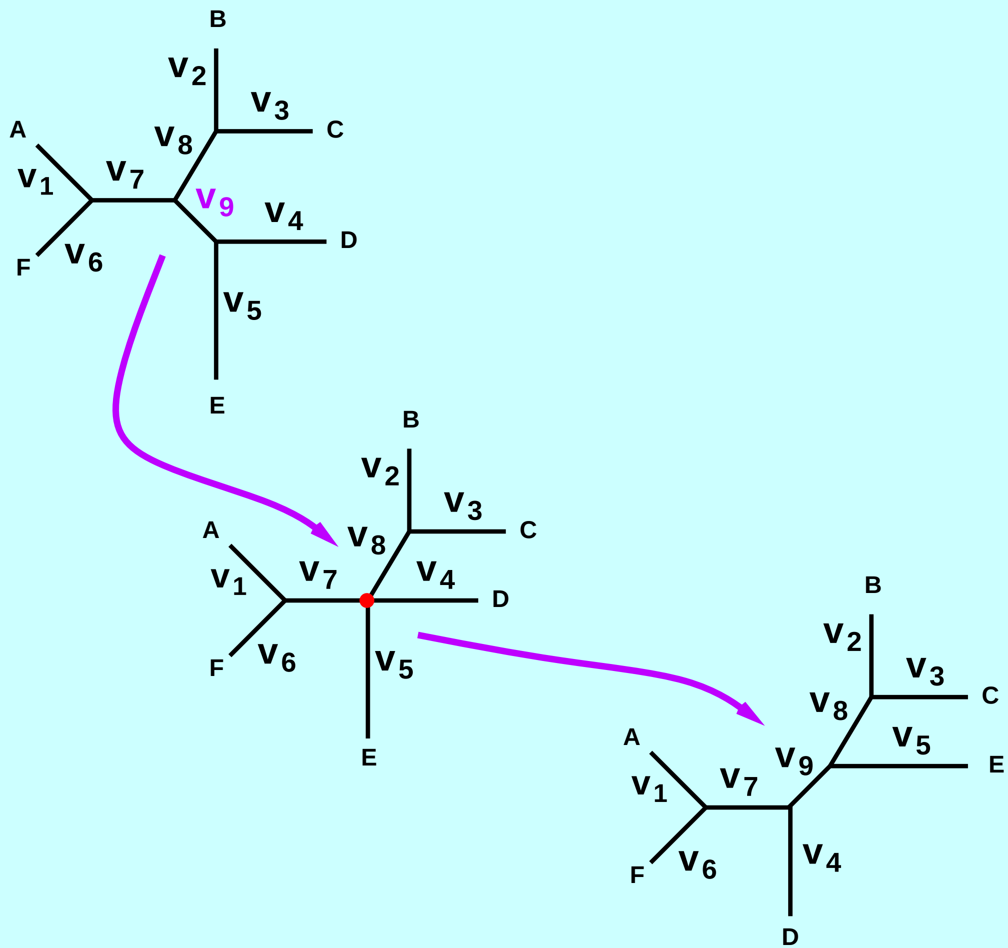
Through the looking glass ...



Through the looking glass ...

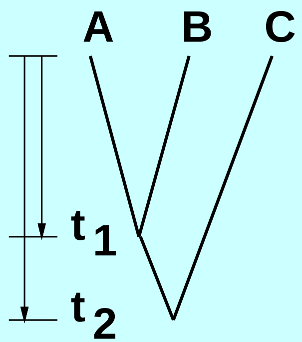


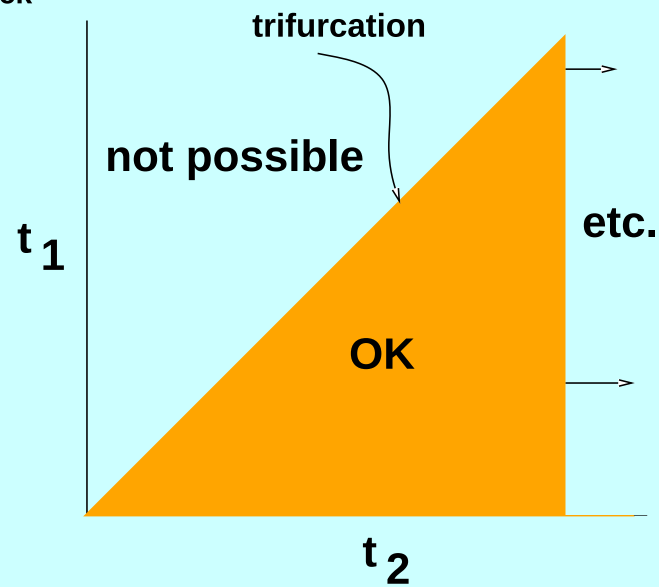
Through the looking glass ...



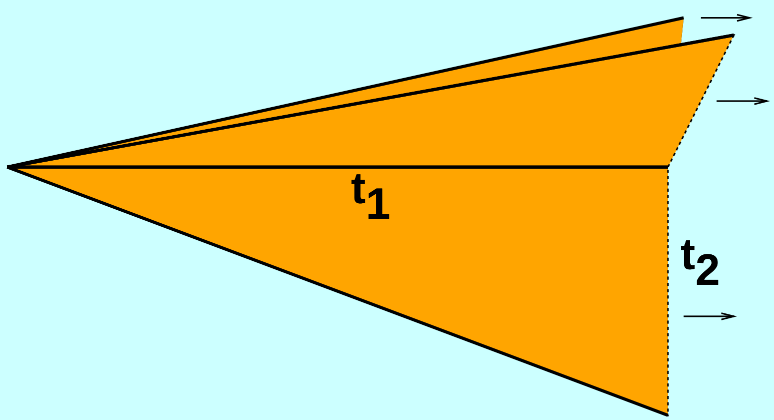
Tree space

an example: three species with a clock

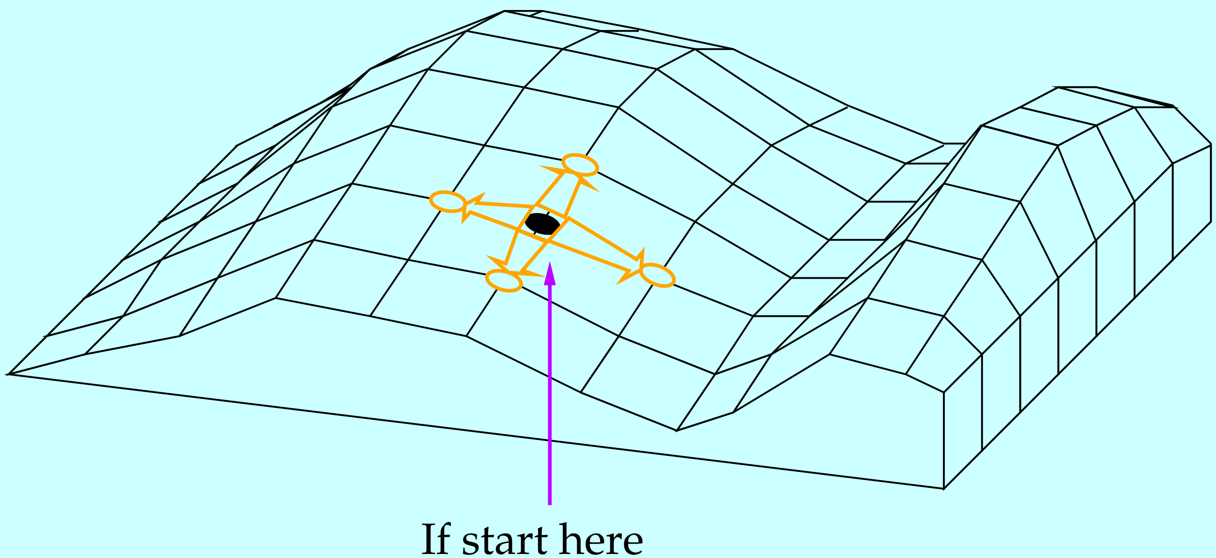




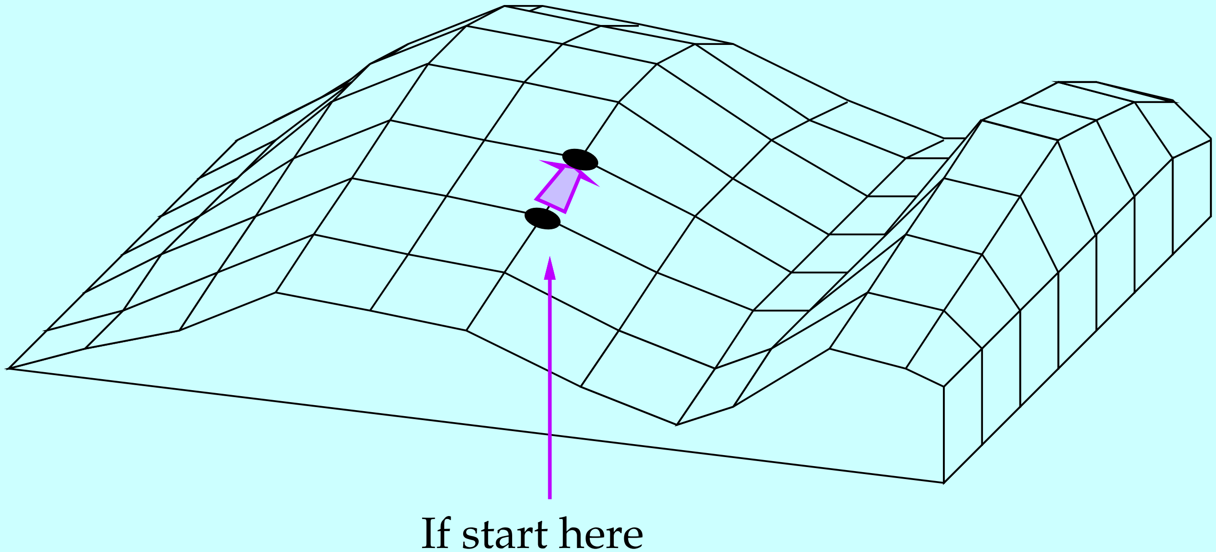
when we consider all three possible topologies, the space looks like:



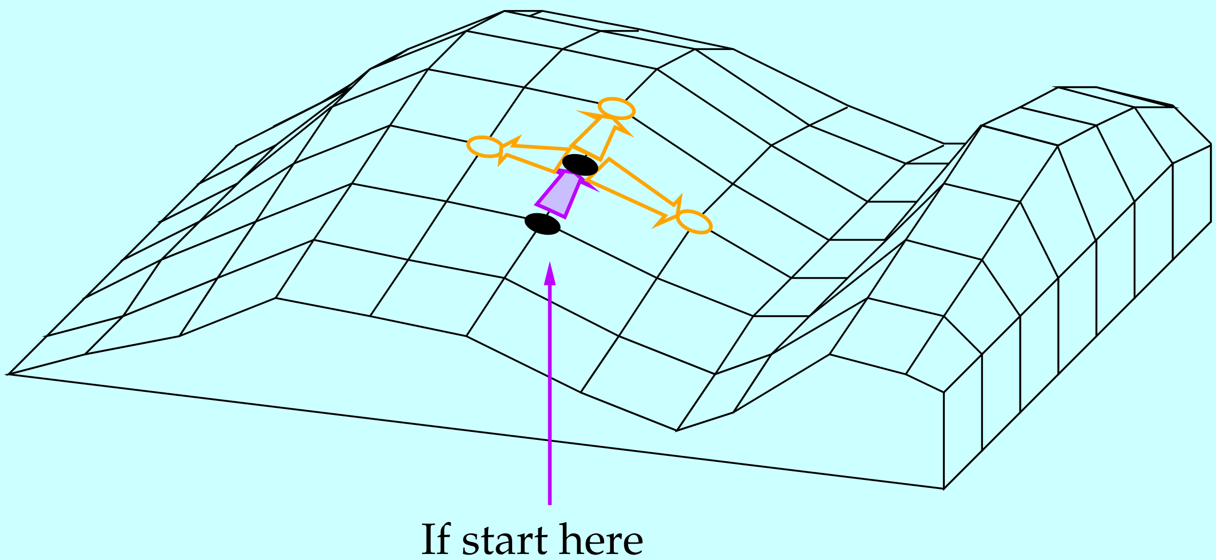
A global maximum is not easy to find



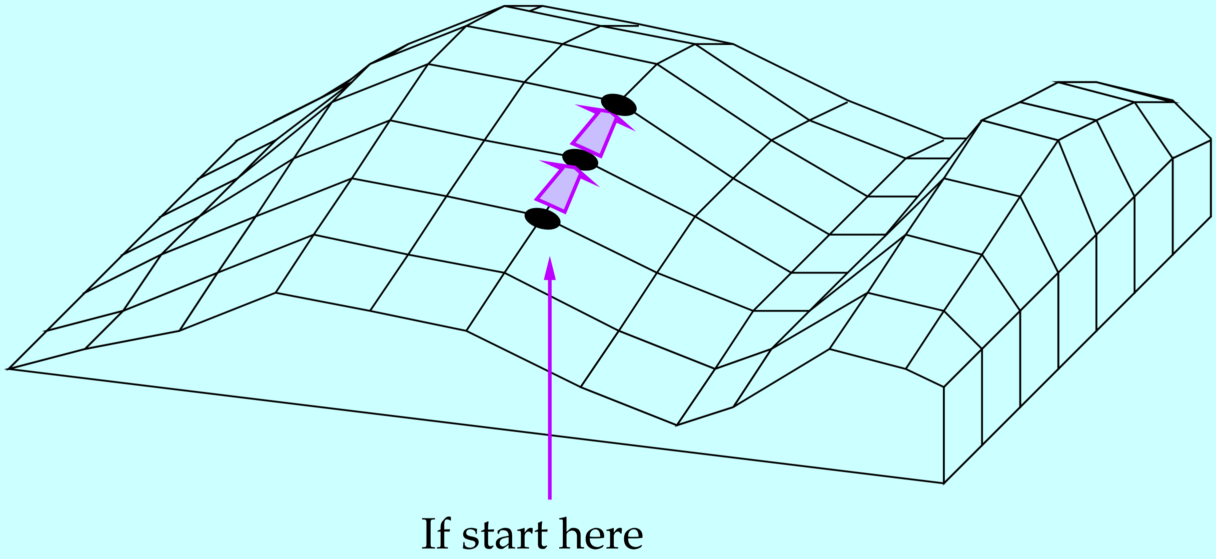
A global maximum is not easy to find



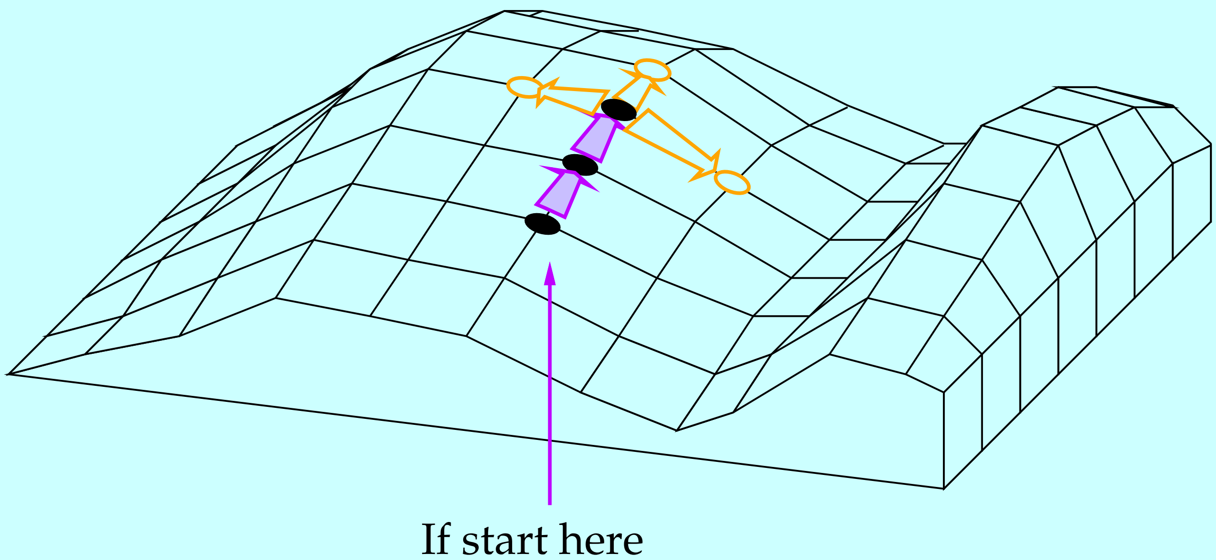
A global maximum is not easy to find



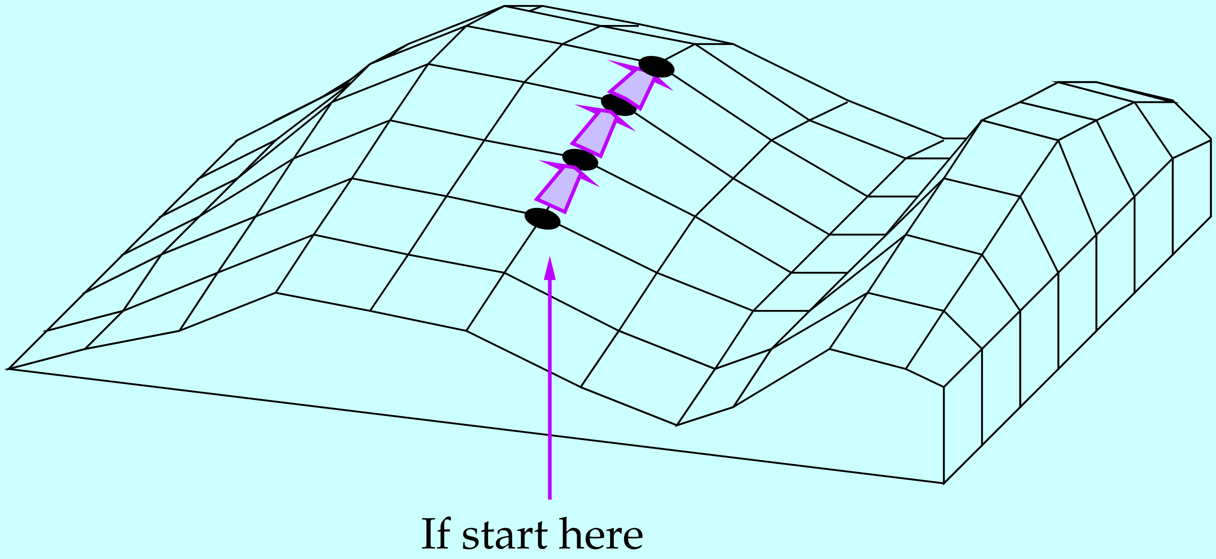
A global maximum is not easy to find



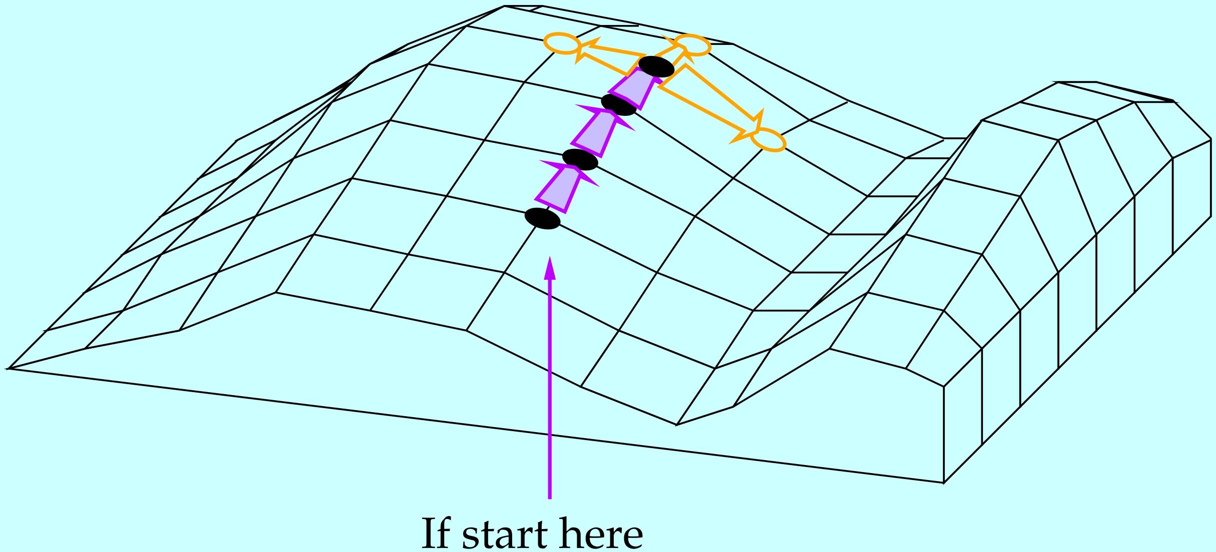
A global maximum is not easy to find



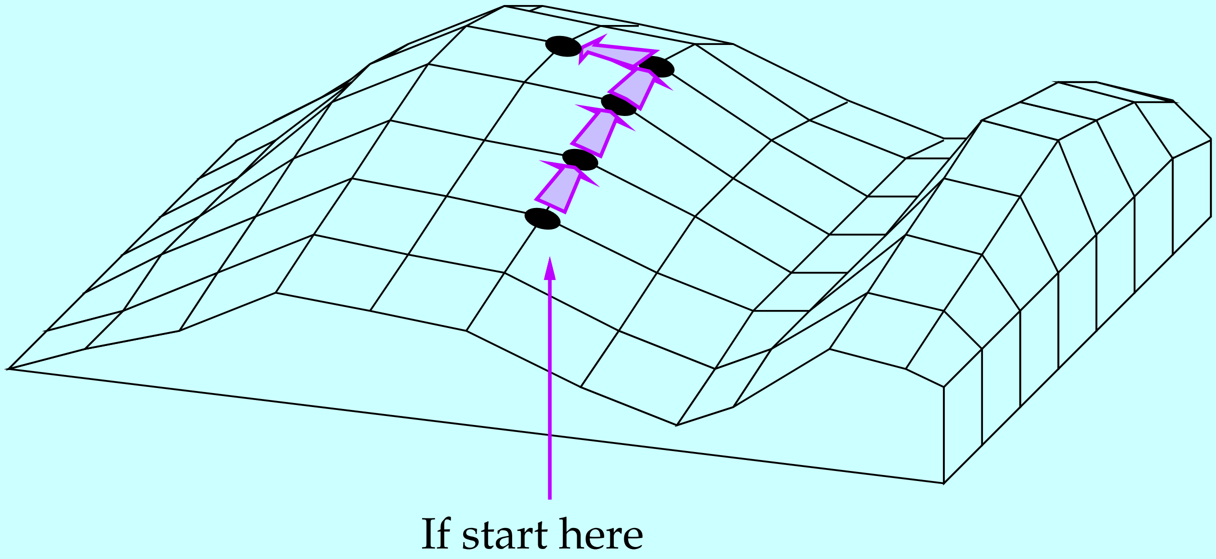
A global maximum is not easy to find



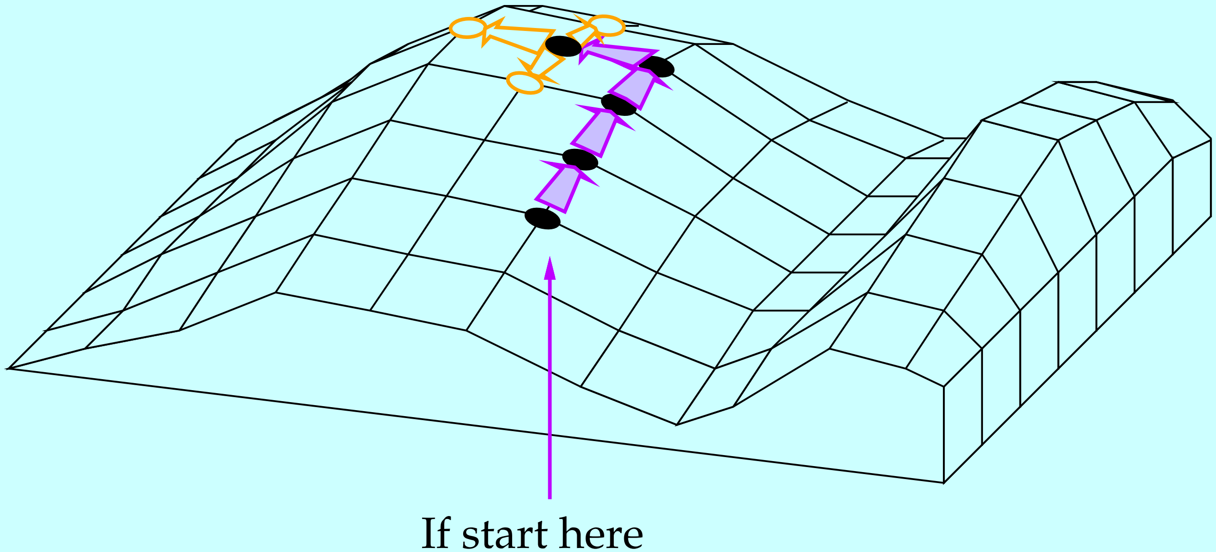
A global maximum is not easy to find



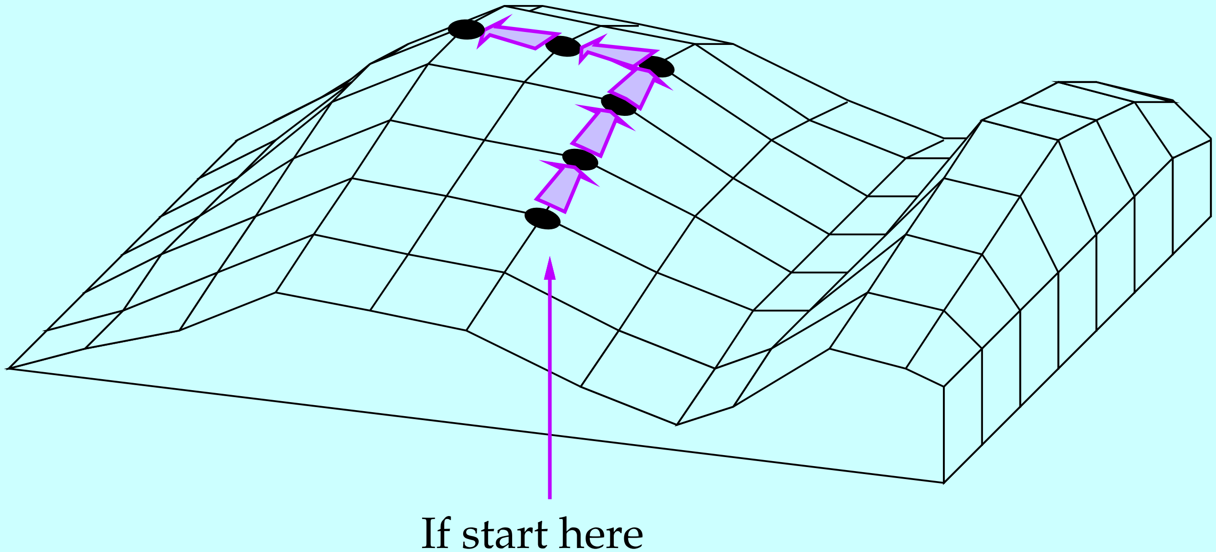
A global maximum is not easy to find



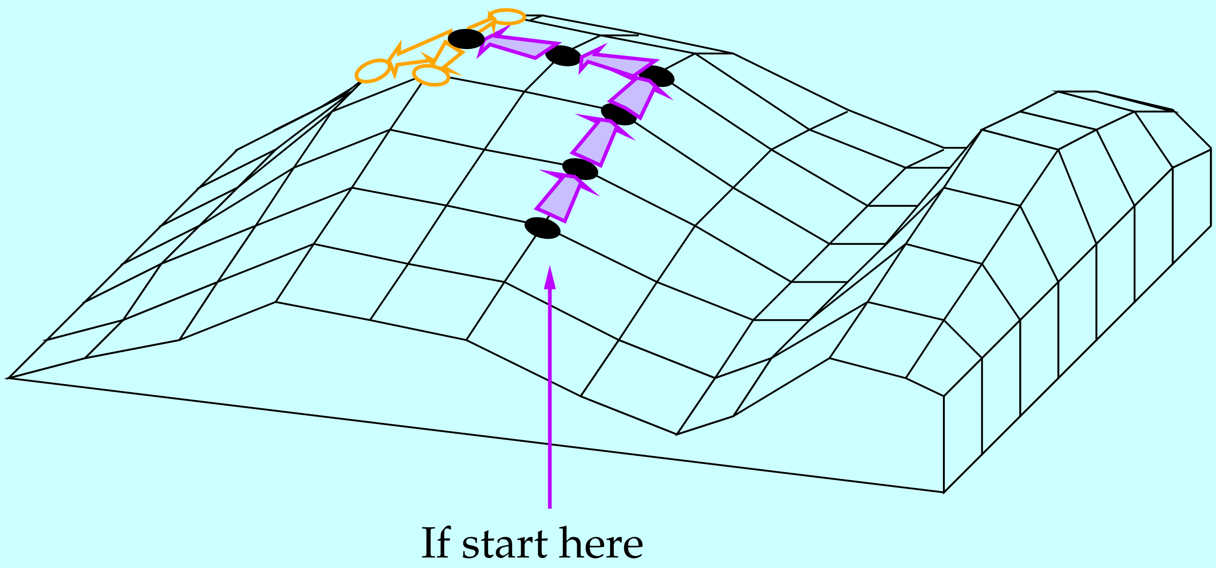
A global maximum is not easy to find



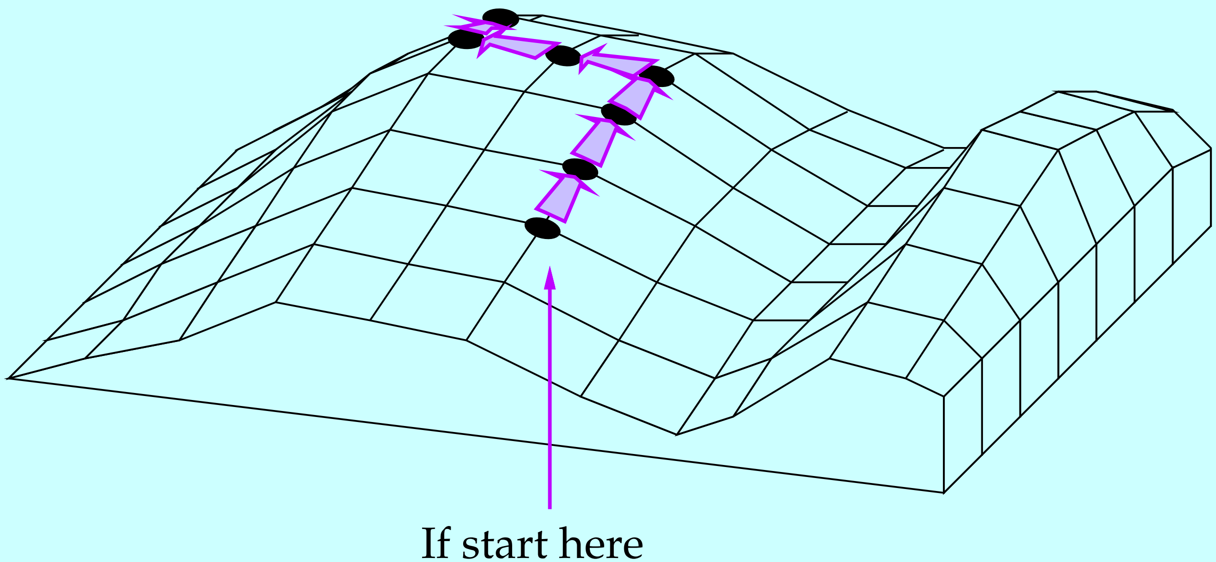
A global maximum is not easy to find



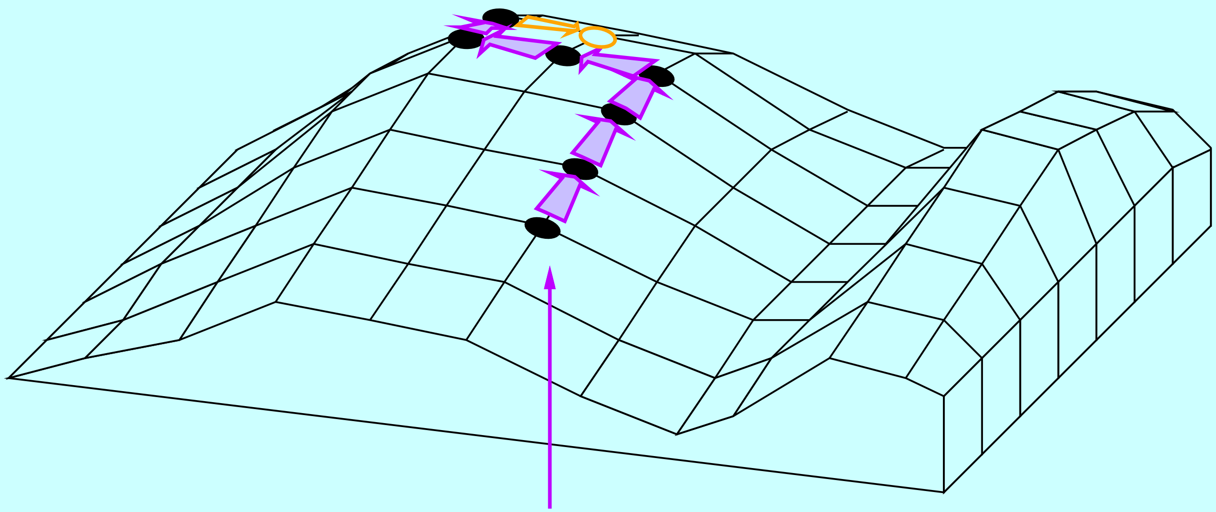
A global maximum is not easy to find



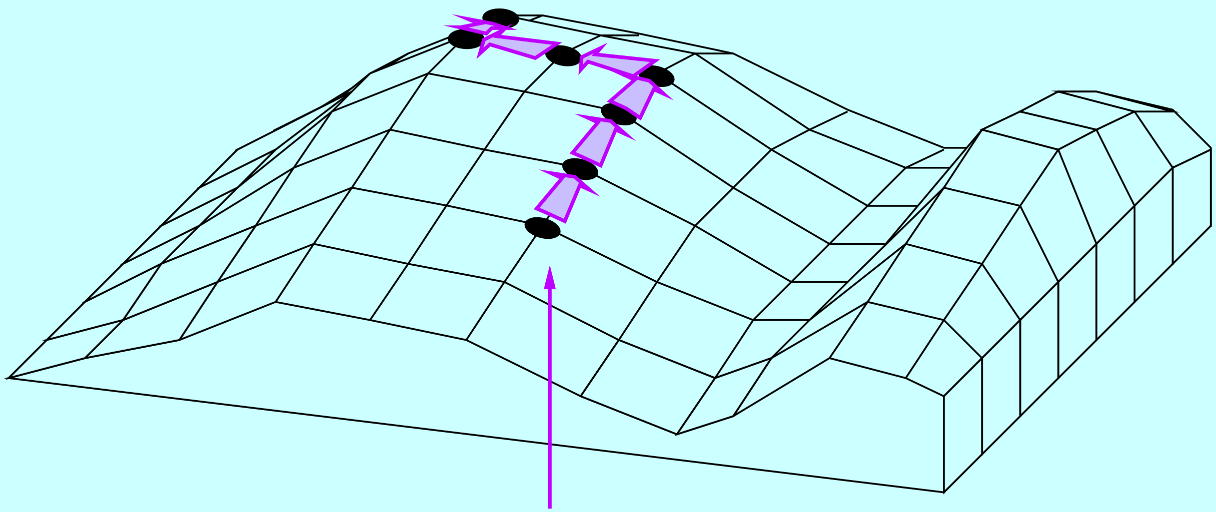
A global maximum is not easy to find



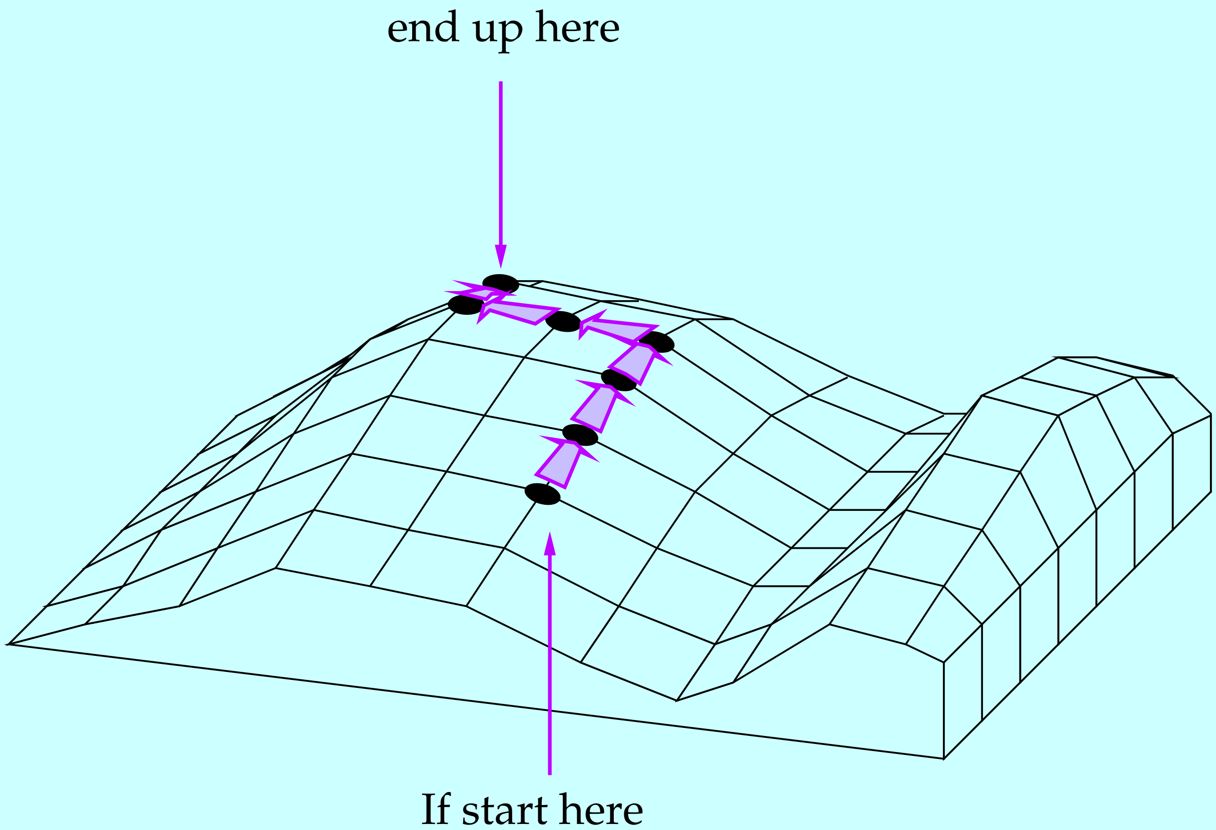
A global maximum is not easy to find



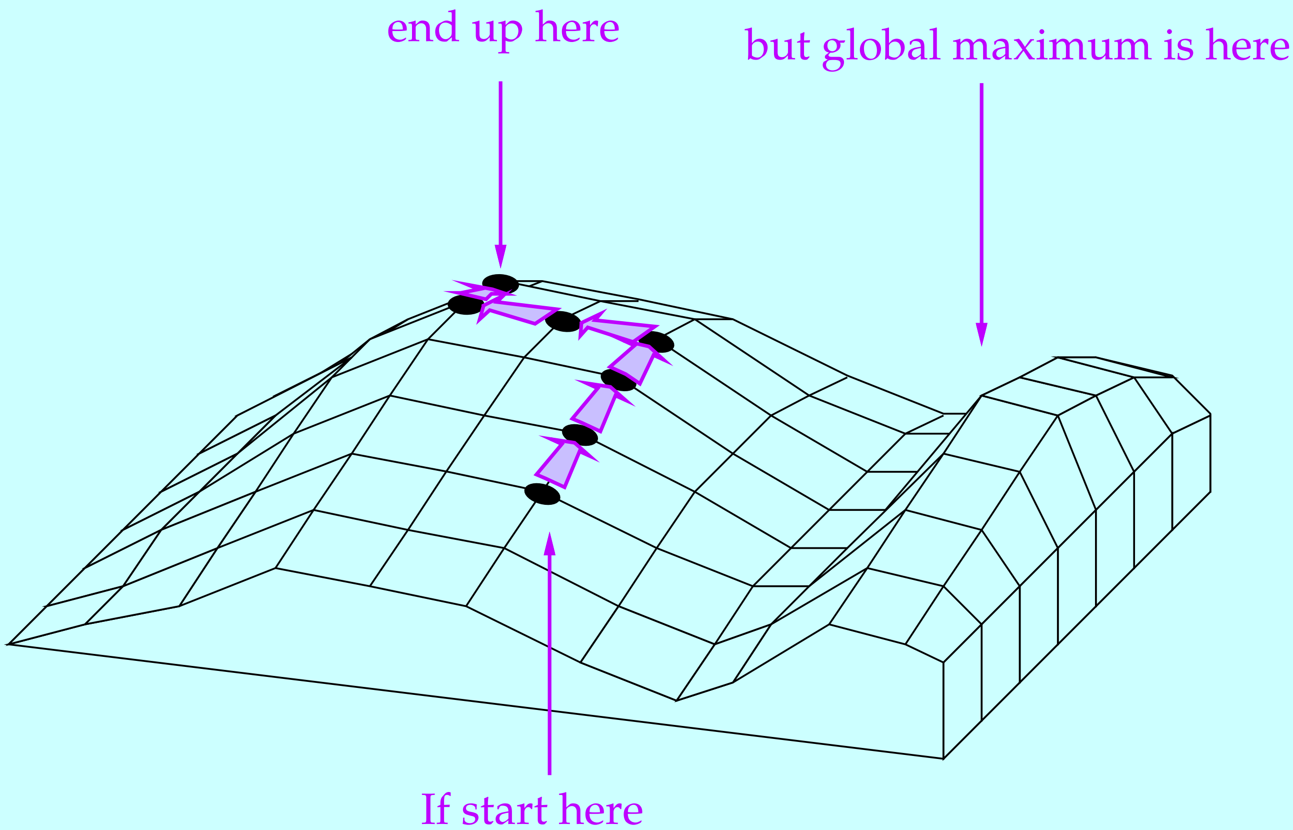
A global maximum is not easy to find



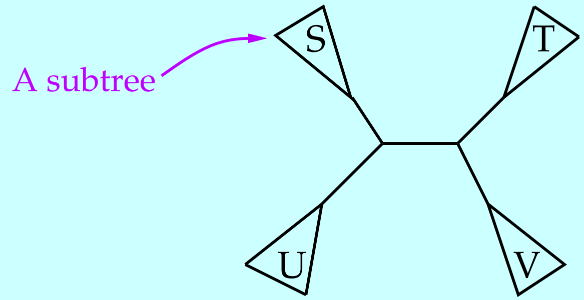
A global maximum is not easy to find



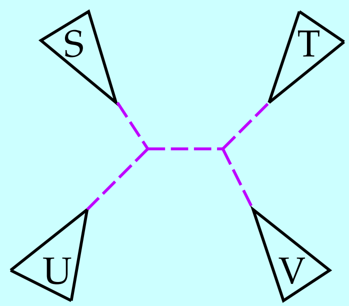
A global maximum is not easy to find



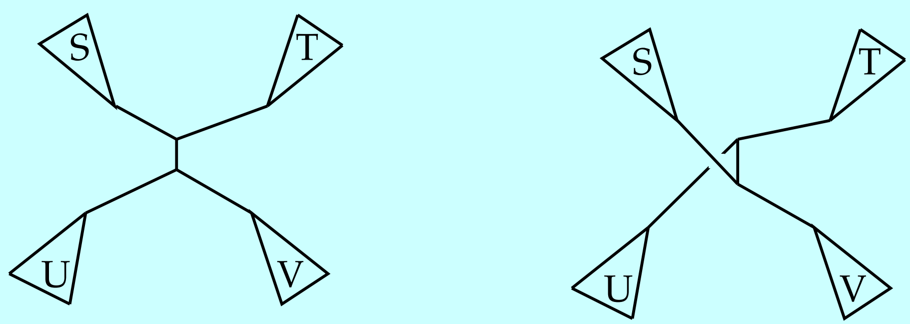
Nearest-neighbor interchanges (NNIs)



is rearranged by dissolving the connections to an interior branch

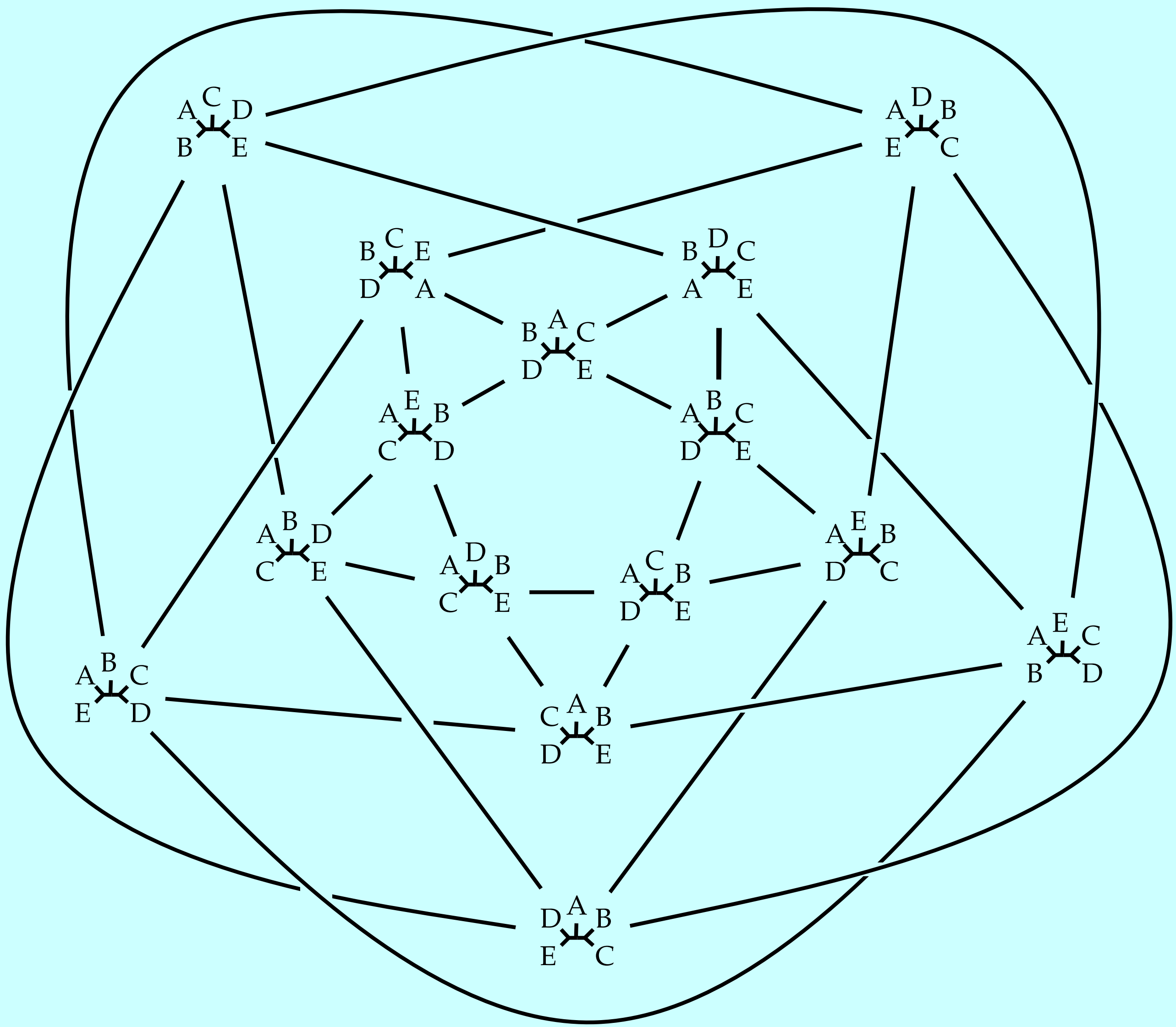


and reforming them in one of the two possible alternative ways:

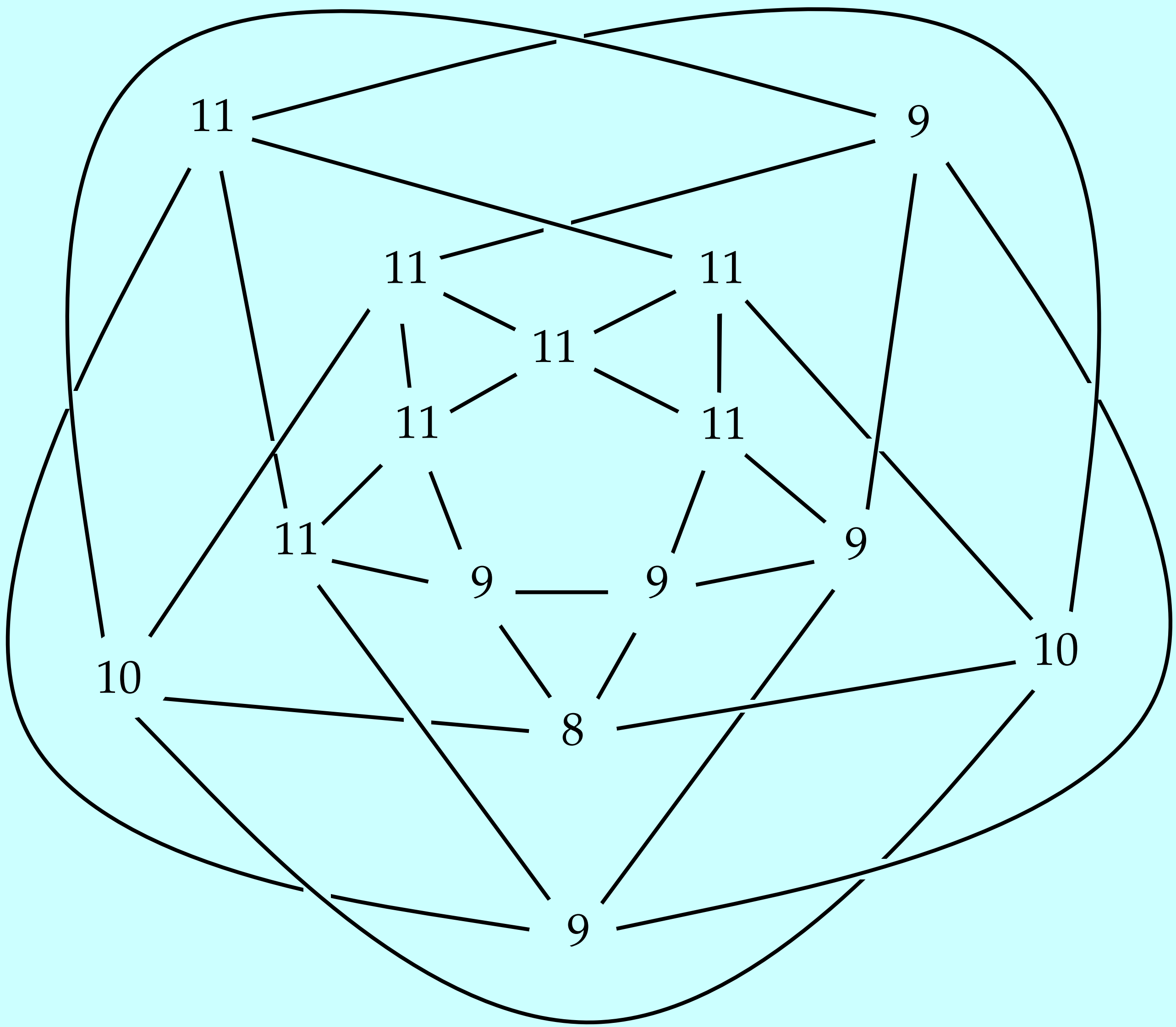


(The triangles are subtrees)

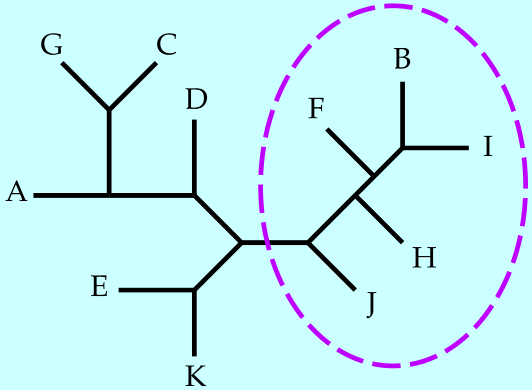
all 15 trees, connected by NNIs



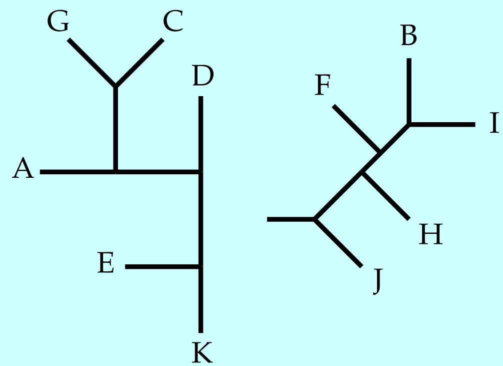
with parsimony scores



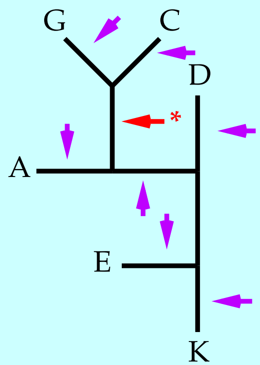
Subtree pruning and regrafting (SPR) rearrangement



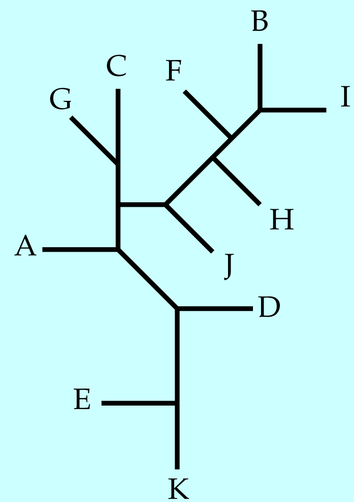
Break a branch, remove a subtree



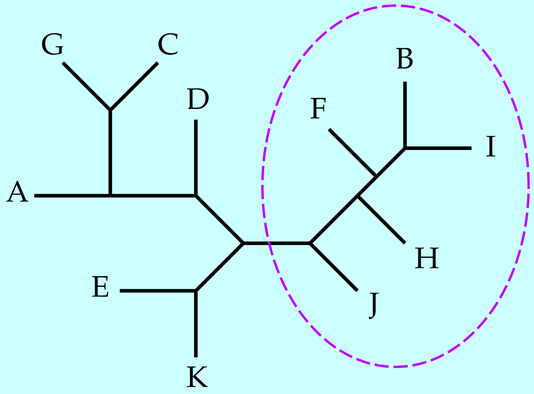
Add it in, attaching it to one (*)
of the other branches



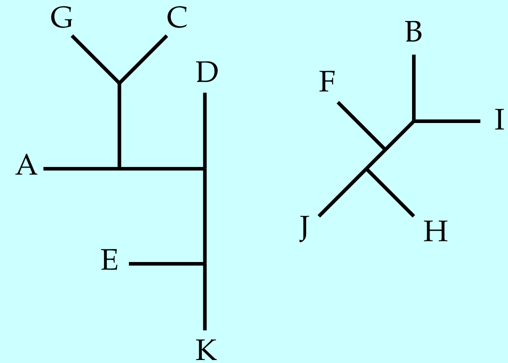
Here is the result:



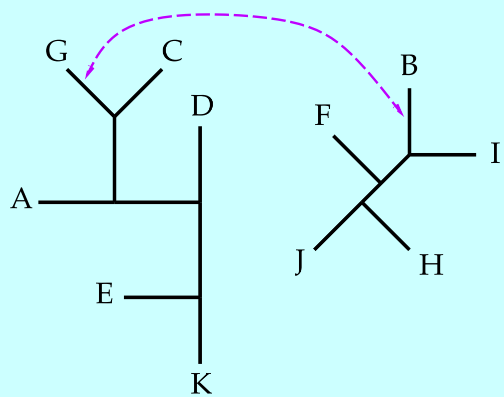
Tree bisection and reconnection (TBR) rearrangement



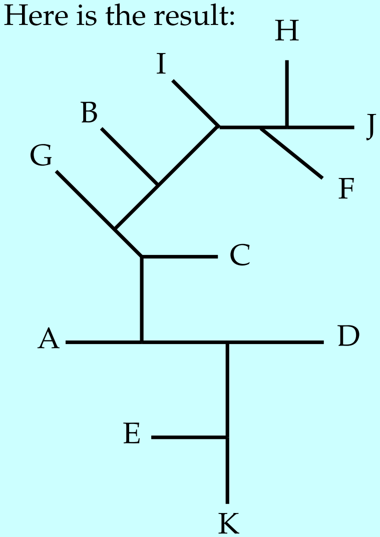
Break a branch, separate the subtrees



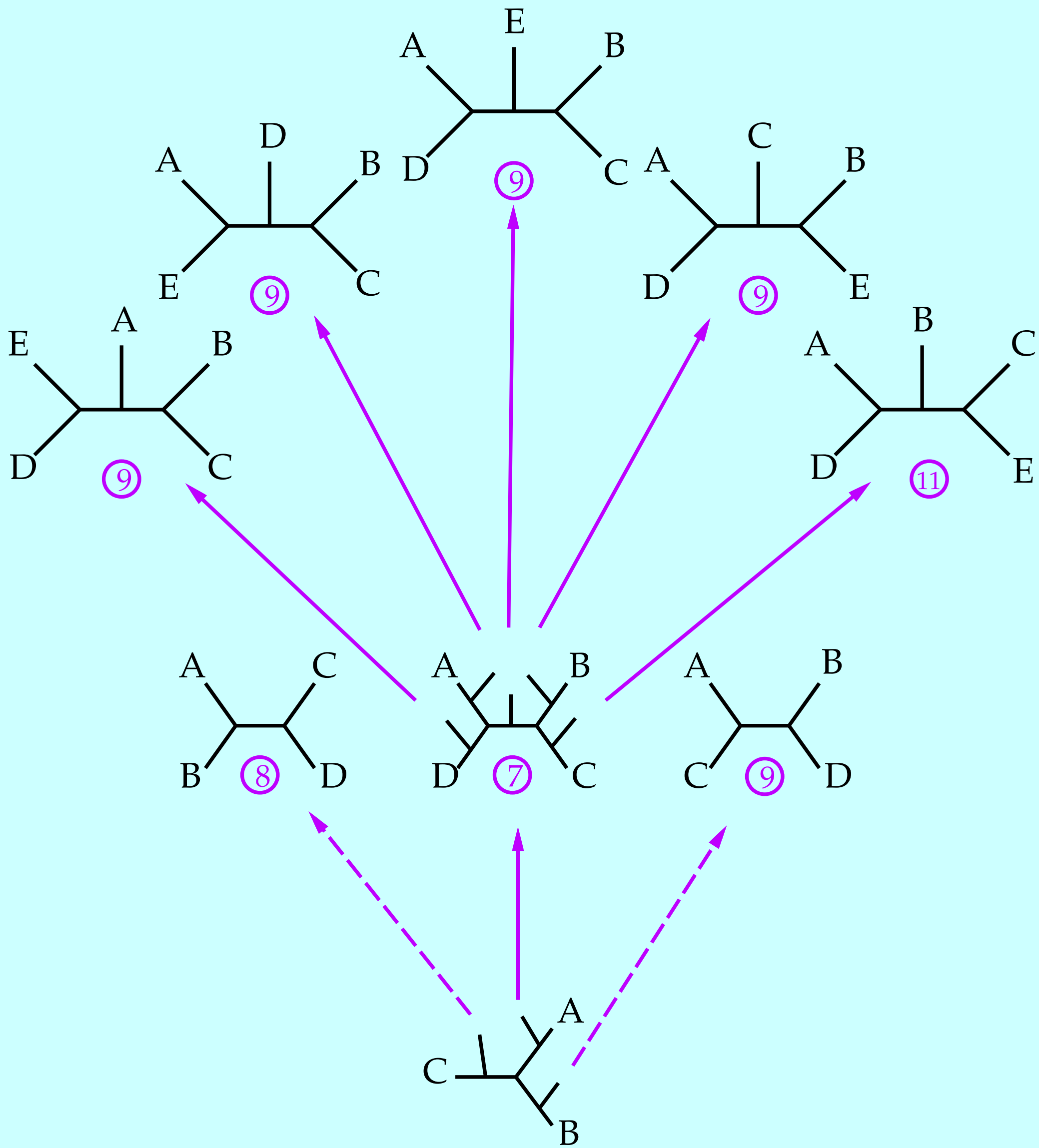
Connect a branch of one
to a branch of the other



Here is the result:

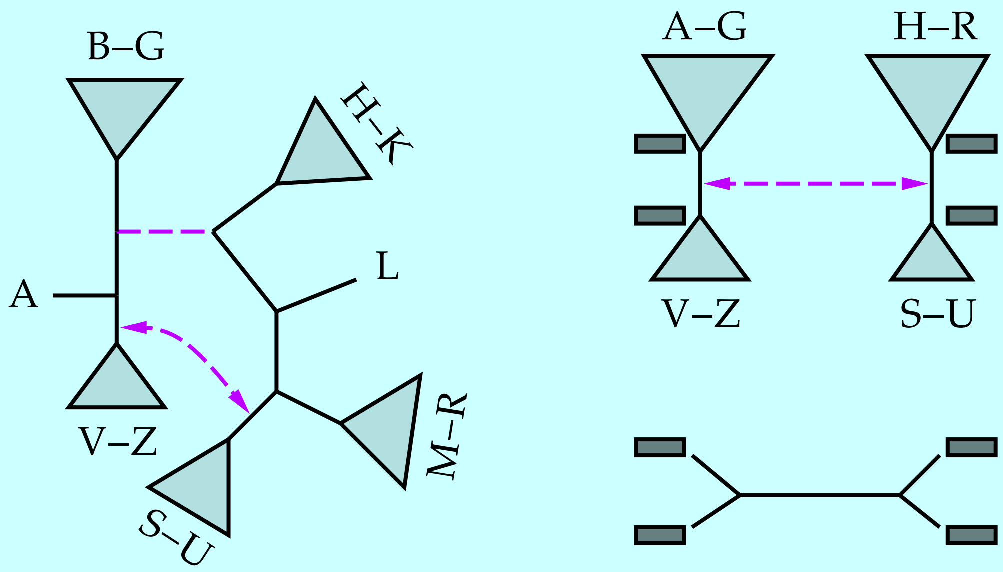


Greedy search by sequential addition



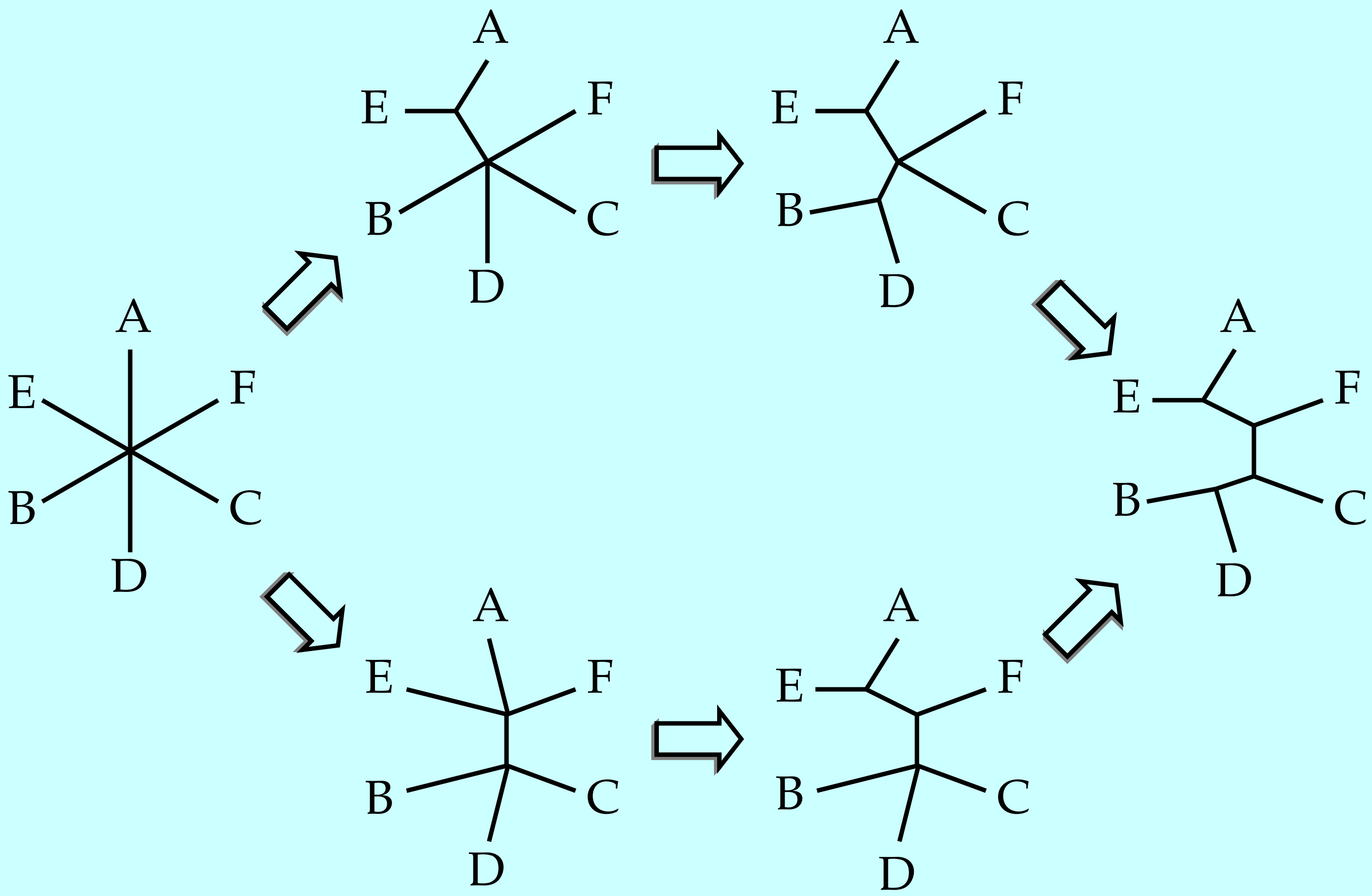
Greedy search by addition of species in a fixed order (A, B, C, D, E) in the best place each time.

Goloboff's time-saving trick



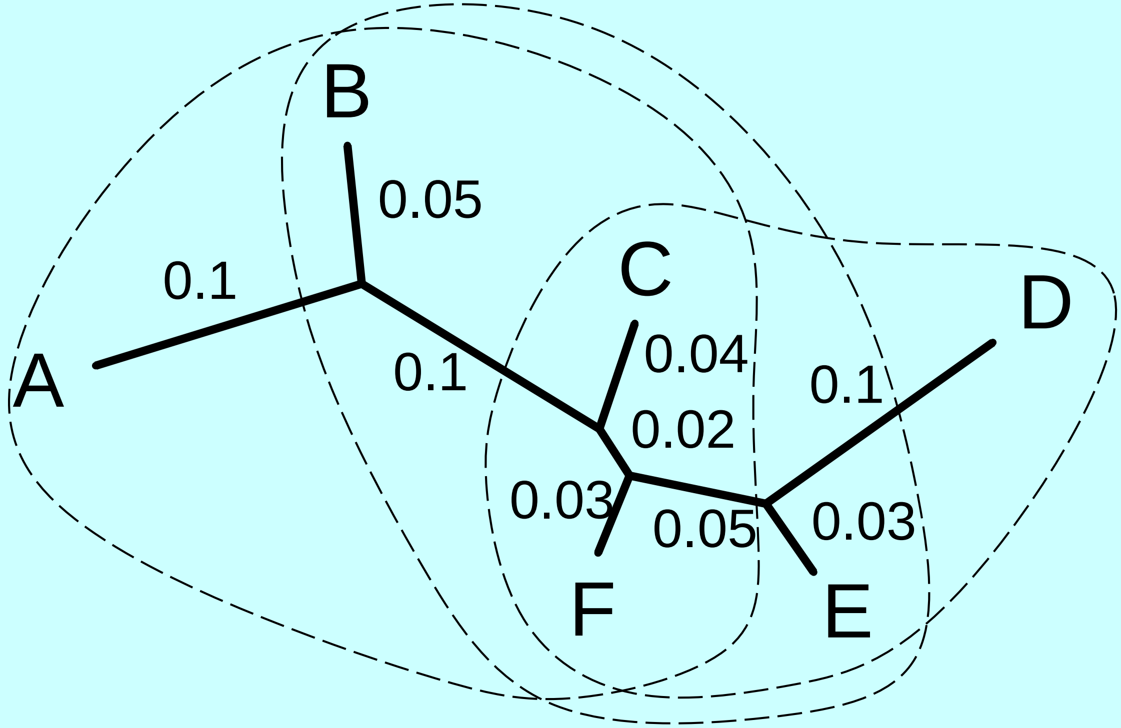
Goloboff's economy in computing scores of rearranged trees
Once the "views" have been computed, they can be taken to represent subtrees, without going inside those subtrees

Star decomposition



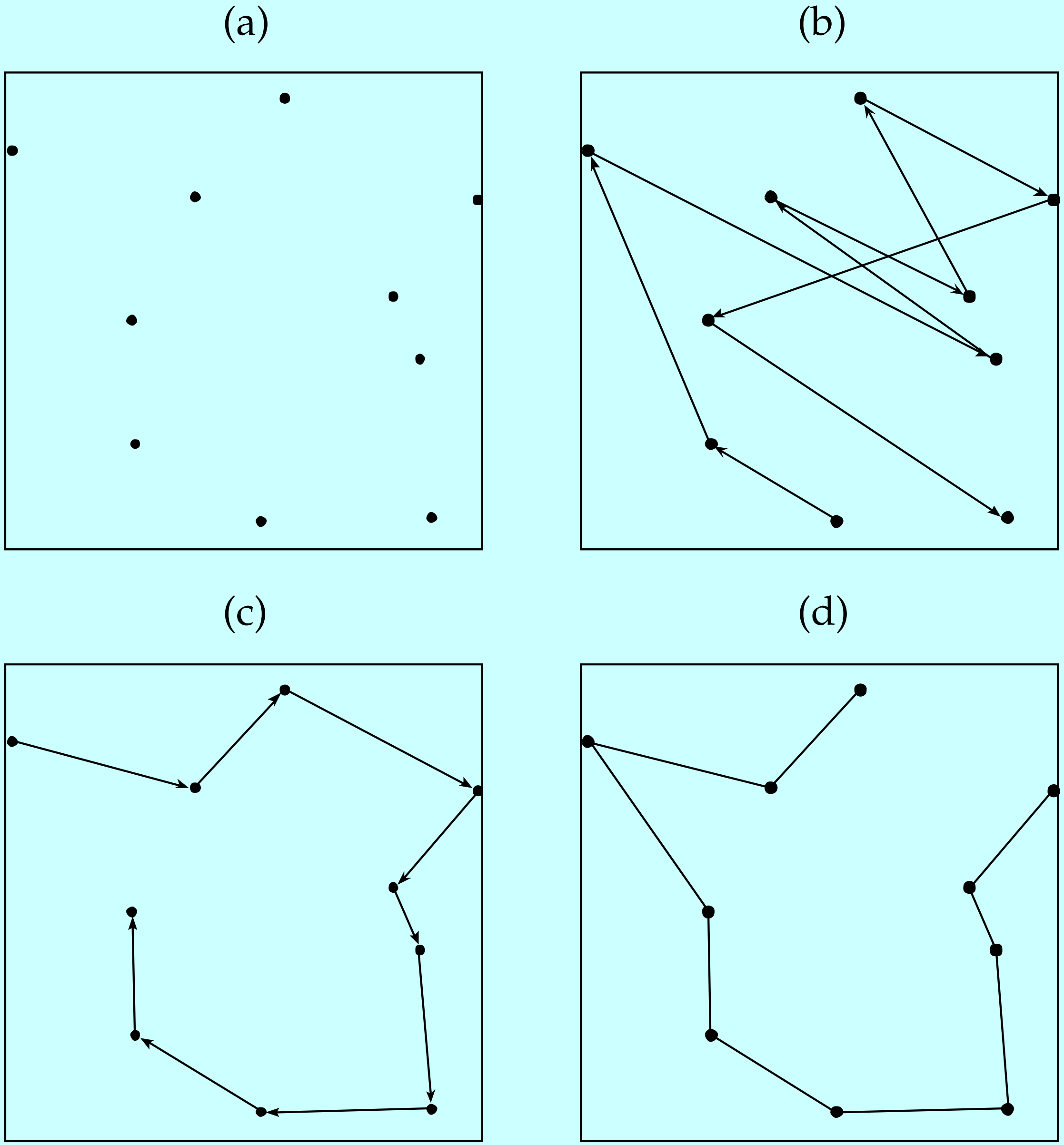
“Star decomposition" search for best tree can happen in multiple ways

Disk-covering

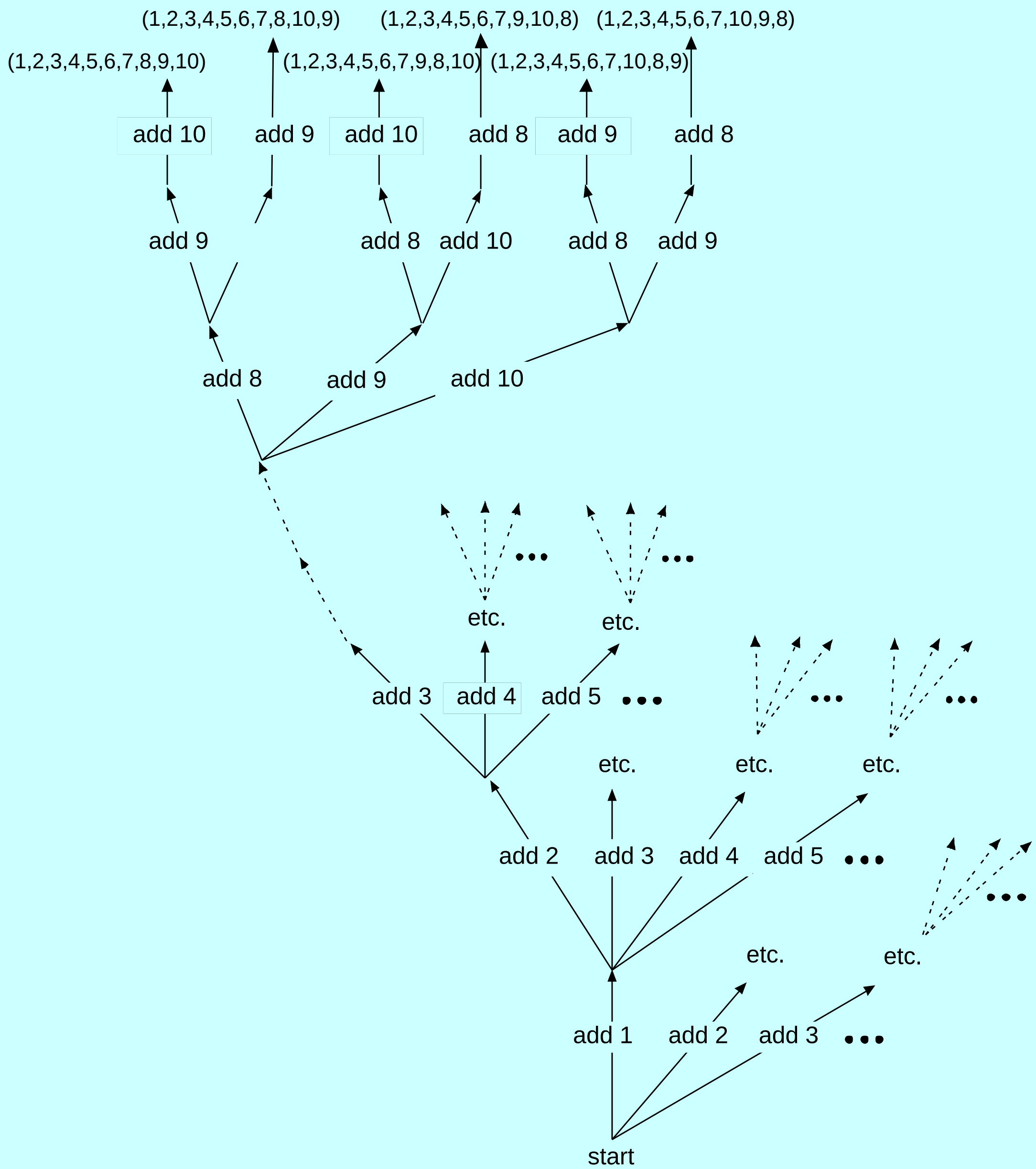


“Disk covering" – assembly of a tree from overlapping estimated subtrees

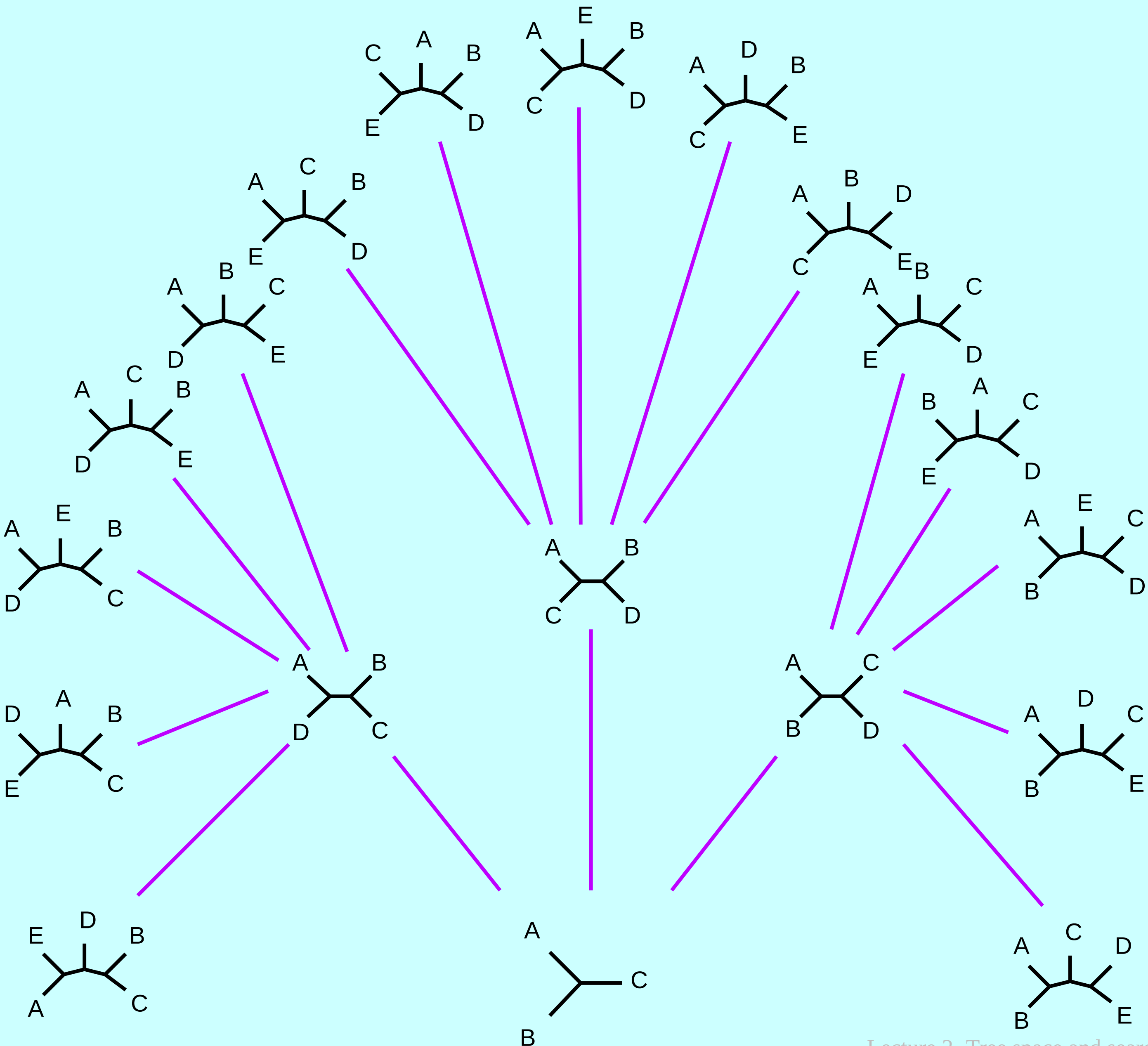
Shortest Hamiltonian path problem



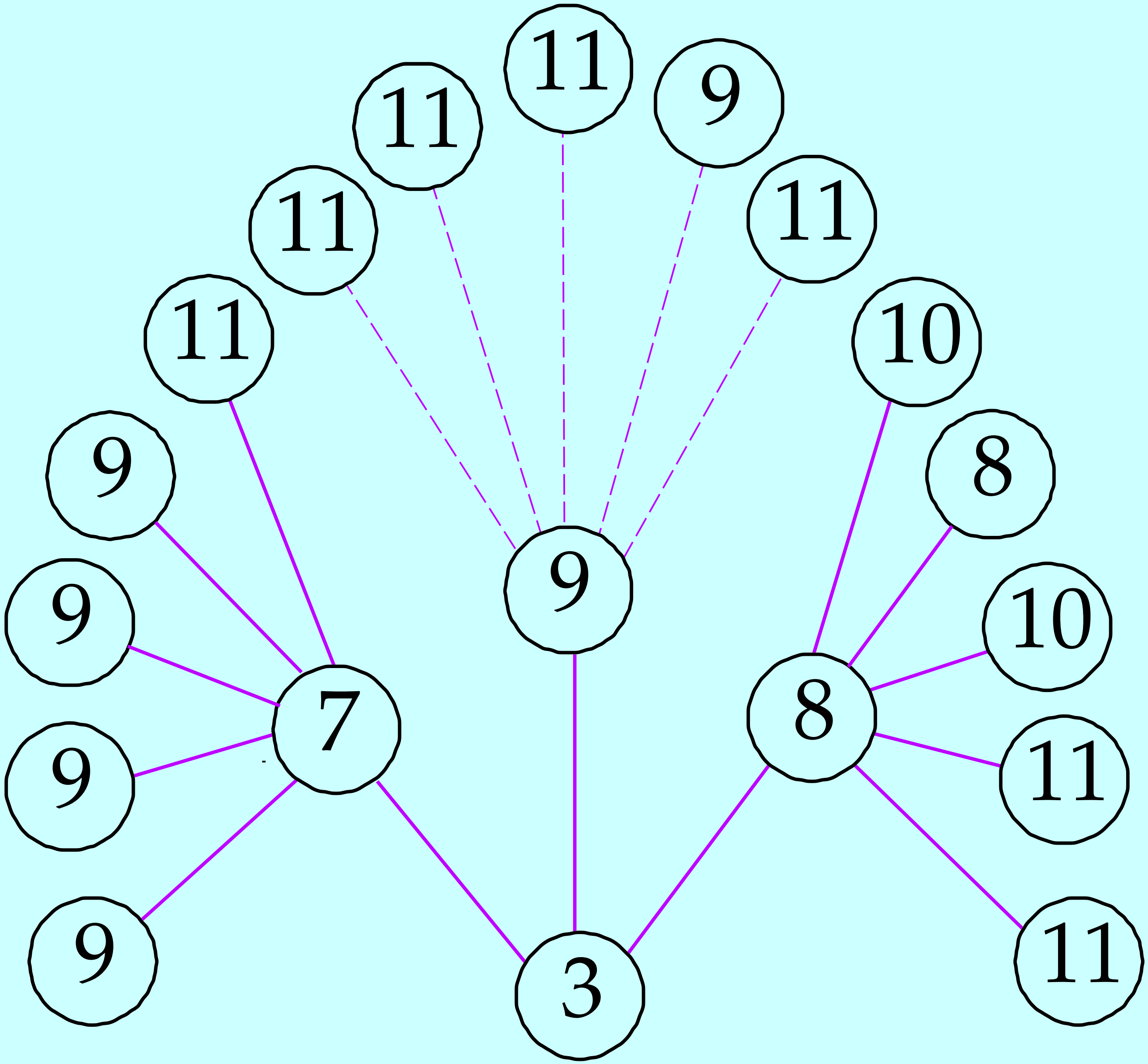
Search tree for this problem



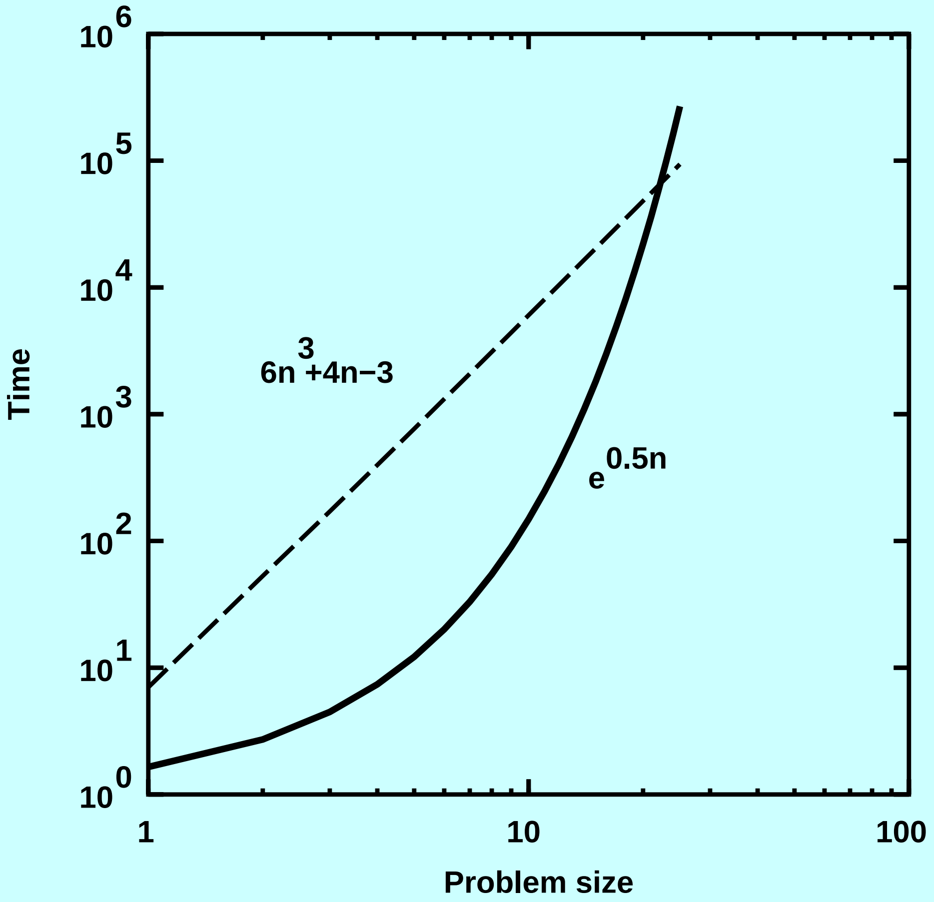
Search tree of trees



same, with parsimony scores in place of trees

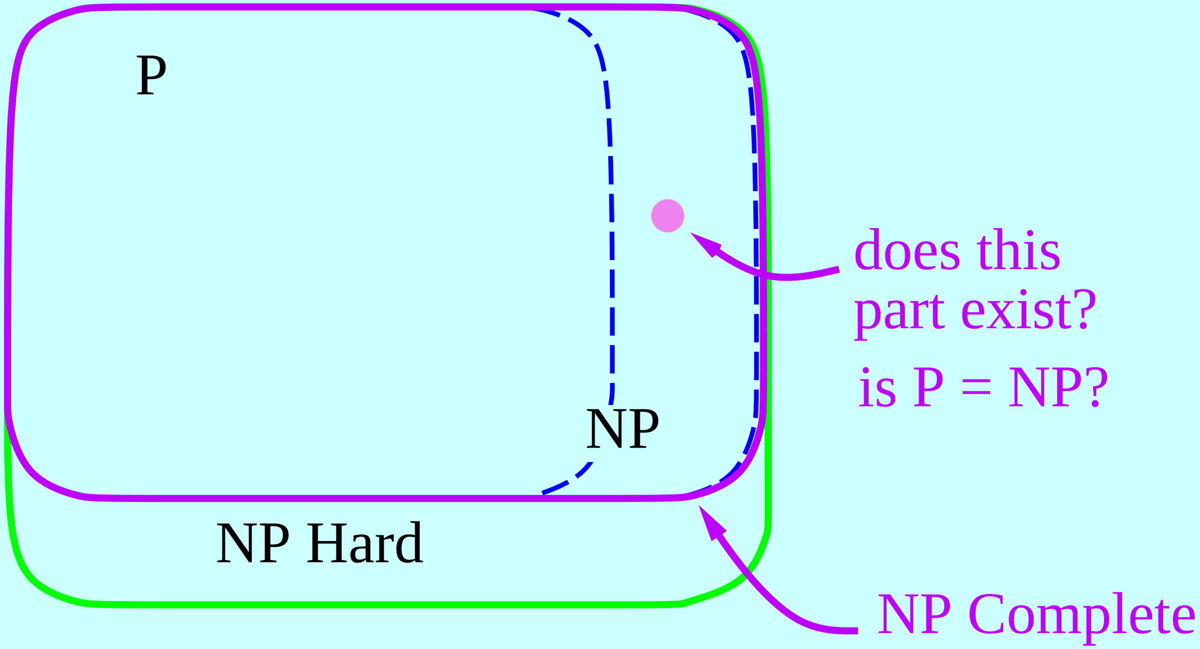


Polynomial time and exponential time



How does the time taken by an algorithm depend on the size of the problem? If it is a polynomial (even one with big coefficients), with a big enough case it is faster than one that depends on the size exponentially.

NP completeness and NP hardness



(This diagram is not quite correct – see the diagrams on the Wikipedia page for “NP-hard”).

P = problems that can be solved by a polynomial time algorithm

NP complete = problems for which a proposed solution can be checked in polynomial time but for which it can be proven that if one of them is in P, all are.

NP hard = problems for which a solution can be checked in polynomial time, but might be not solvable in polynomial time.

Some references

Felsenstein, J. 1978. The number of evolutionary trees. *Systematic Zoology* **27:** 27-33. (Correction, vol. 30, p. 122, 1981) **[Review of counting tip-labelled trees, recursion for counting multifurcating case]**

Cavalli-Sforza, L. L. and A. W. F. Edwards. 1967. Phylogenetic analysis: models and estimation procedures. *American Journal of Human Genetics* **19:** 233-257. also *Evolution* **21:** 550-570. **[Includes counting and tree shapes]**

Camin, J. H. and R. R. Sokal. 1965. A method for deducing branching sequences in phylogeny. *Evolution* **19:** 311-326. **[Early parsimony paper includes rearrangement of trees]**

Waterman, M. S. and T. F. Smith. 1978. On the similarity of dendrograms. *Journal of Theoretical Biology* **73:** 789-800. **[Defines NNIs. Uses them to get a distance between trees.]**

Maddison, D. R. 1991. The discovery and importance of multiple islands of most-parsimonious trees. *Systematic Zoology* **40:** 315-328. **[Discusses heuristic search strategy involving ties, multiple starts]**

Farris, J. S. 1970. Methods for computing Wagner trees. *Systematic Zoology* **19:** 83-92. **[Early parsimony algorithms paper is one of first to mention sequential addition strategy]**

continued

Saitou, N., and M. Nei. 1987. The neighbor-joining method: a new method for reconstructing phylogenetic trees. *Molecular Biology and Evolution* **4:** 406-425. **[First mention of star-decomposition search for best trees, sort of]**

Strimmer, K., and A. von Haeseler. 1996. Quartet puzzling: a quartet maximum likelihood method for reconstructing tree topologies. *Molecular Biology and Evolution* **13:** 964-969. **[Assembles trees out of quartets]**

Huson, D., S. Nettles, L. Parida, T. Warnow, and S. Yooseph. 1998. The disk-covering method for tree reconstruction. pp. 62-75 in *Proceedings of "Algorithms and Experiments" (ALEX98), Trento, Italy, Feb. 9-11, 1998*, ed. R. Battiti and A. A. Bertossi. **["Disk-covering method" for long stringy trees]**

Swofford, D. L. and G. J. Olsen. 1990. Phylogeny reconstruction. Chapter 11, Pp. 411-501 in *Molecular Systematics,* ed. D. M. Hillis and C. Moritz. Sinauer Associates, Sunderland, Massachusetts. **[Review that discusses strategies, names SPR and TBR rearrangement methods]**

Foulds, L. R. and R. L. Graham. 1982. The Steiner problem in phylogeny is NP-complete. *Advances in Applied Mathematics* **3:** 43-49. **[Parsimony is NP-hard]**

Graham, R. L. and L. R. Foulds. 1982. Unlikelihood that minimal phylogenies for a realistic biological study can be constructed in reasonable computat ional time. *Mathematical Biosciences* **60:** 133-142. **[... and more]**

continued

Hendy, M. D. and D. Penny. 1982. Branch and bound algorithms to determine minimal evolutionary trees. *Mathematical Biosciences* **60:** 133-142 **[Introduced branch-and-bound for phylogenies]**

Felsenstein, J. 2004. *Inferring Phylogenies.* Sinauer Associates, Sunderland, Massachusetts. **[For this lecture the material is chapters 3, 4, and 5]**

Semple, C. and M. Steel. 2003. *Phylogenetics.* Oxford University Press, Oxford. **[Also covers search strategies]**