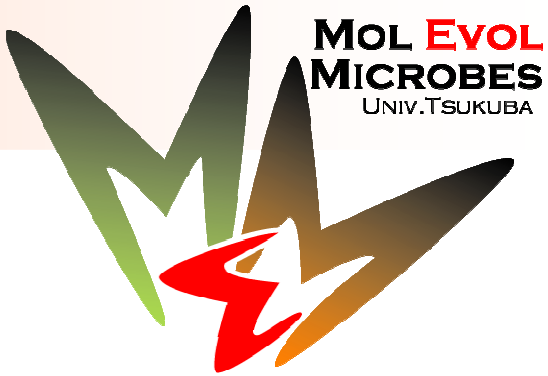
計算科学研究センター平成21年度学際共同利用研究成果発表
@ May 7th, 2010



ヒューリスティツクな樹形探索効率評価

稲垣祐司
（筑波大学大学院生命環境科学研究科）

重点課題推進プログラム（09b-24）
スタート樹形数がヒューリステイックな樹形探索効率に及ぼす影響評価
(Impact of the number of start trees on the efficiency of heuristic tree search)
代表：稲垣祐司（筑波大学大学院生命環境科学研究科）

最尤法(1)

- データを現実化する確率をパラメータの関数とみたもの
  - すべての可能な系統樹の中から尤度を最大化する系統樹を選択する

配列データ

$$X = \begin{pmatrix} X_{11}, X_{12}, \cdots\cdots, X_{1h}, \cdots\cdots, X_{1n} \\ X_{21}, X_{22}, \cdots\cdots, X_{2h}, \cdots\cdots, X_{2n} \\ X_{31}, X_{32}, \cdots\cdots, X_{3h}, \cdots\cdots, X_{3n} \\ X_{41}, X_{42}, \cdots\cdots, X_{4h}, \cdots\cdots, X_{4n} \end{pmatrix}$$

h番目の座位:

$$X_h = \begin{pmatrix} X_{1h} \\ X_{2h} \\ X_{3h} \\ X_{4h} \end{pmatrix}$$

たとえば、

$$X = \begin{pmatrix} \mathrm{A}, \mathrm{C}, \cdots\cdots, \mathrm{T}, \cdots\cdots, \mathrm{G} \\ \mathrm{A}, \mathrm{G}, \cdots\cdots, \mathrm{A}, \cdots\cdots, \mathrm{C} \\ \mathrm{T}, \mathrm{A}, \cdots\cdots, \mathrm{A}, \cdots\cdots, \mathrm{T} \\ \mathrm{A}, \mathrm{G}, \cdots\cdots, \mathrm{T}, \cdots\cdots, \mathrm{C} \end{pmatrix}$$

生物種1
生物種2
生物種3
生物種4

いま、

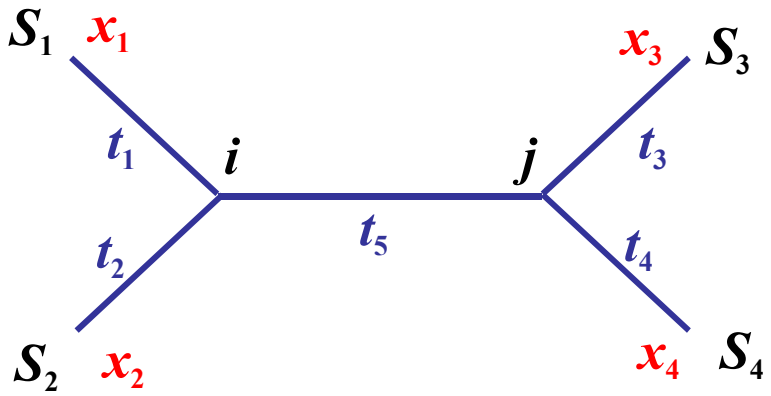
$$X_h = \begin{pmatrix} \boldsymbol{x}_1 \\ \boldsymbol{x}_2 \\ \boldsymbol{x}_3 \\ \boldsymbol{x}_4 \end{pmatrix}$$

であったとする。

最尤法(2)

下記の樹形と枝の長さt_1~t_4の元、枝の末端の塩基x_1~x_4 が実現する確率は以下の通り(枝の末端の塩基)

$$f(x_1, x_2, x_3, x_4|\theta) = \sum_i \left\{ \pi_i Pix_1(t_1) Pix_2(t_2) \sum_j Pij(t_5)\, Pjx_3(t_3)\, Pjx_4(t_4) \right\}$$



π_i :データ行列の塩基 i の組成値

ただし、共通祖先の塩基の状態i, jは不明なので、A,C,G,Tの4通りの場合について足し合わせる。

X_hが独立に同一の確率法則に従って進化していると仮定すると、データ行列を得る確率は、各座位の確率の積となる。その式をθの関数と見なしたものL を尤度といい、尤度を最大にするようなパラメータの値を定めて、それを枝の長さの推定値とする。
すなわち、尤度は、

$$L(\theta|\mathrm{X}) = \prod_{h=1}^{n} f(\mathrm{X}h|\theta)$$

ただし、$\theta = (t_1, \ldots, t_5)$

対数尤度は、 $$l(\theta|\mathrm{X}) = \sum_{h=1}^{n} \log f(\mathrm{X}h|\theta)$$

対数尤度lを最大にするθの推定値 を求める

網羅的探索

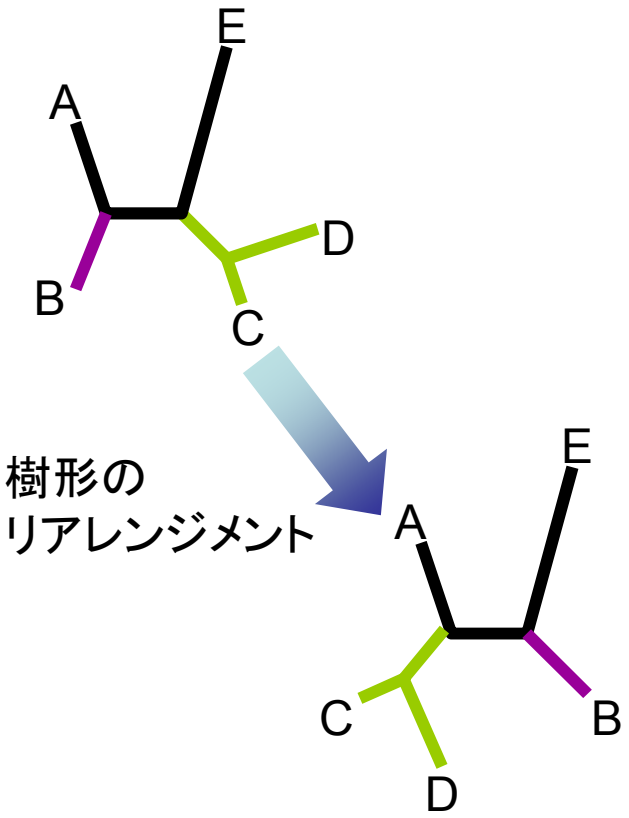
- すべての可能な系統樹の対数尤度を求め、最尤系統樹を選択する ― 網羅的探索
  - 配列数の増加に対し、樹形数は指数関数的に増加する

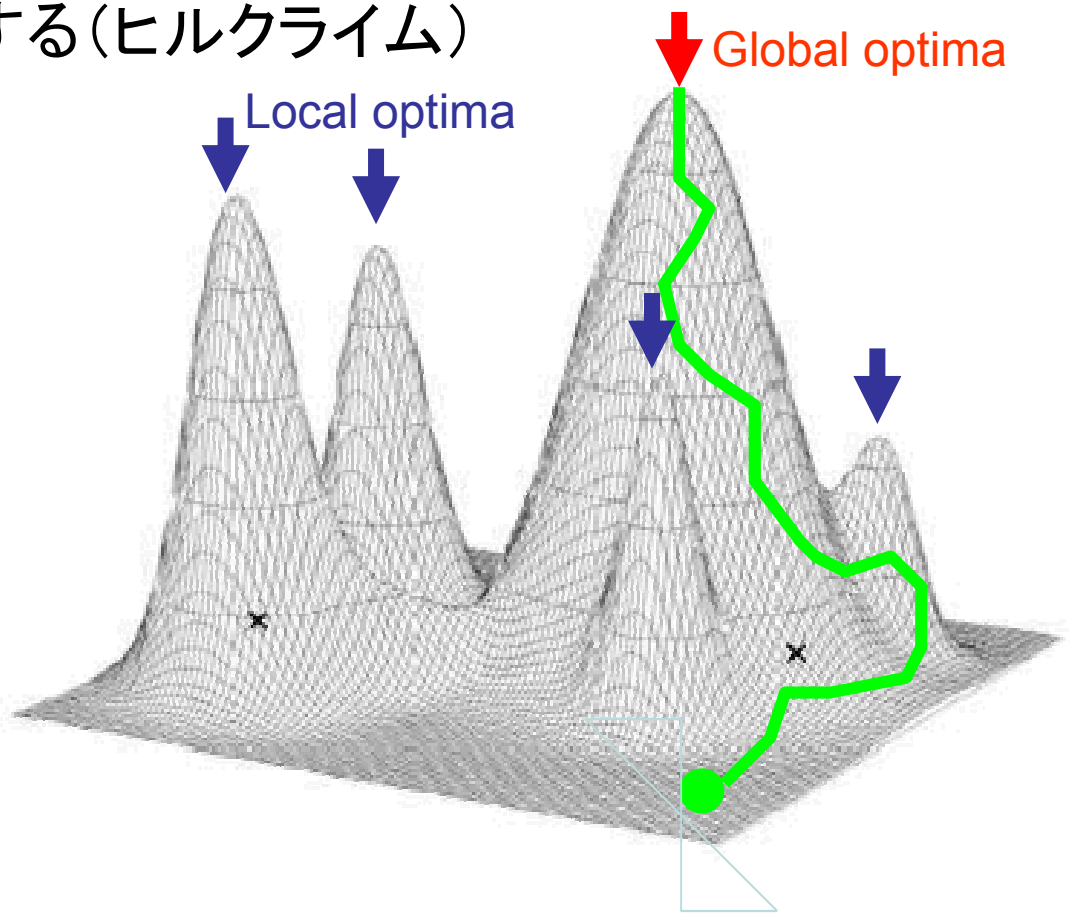
配列数	可能な樹形数
3	1
4	3
5	15
6	105
7	945
8	10395
9	135135
10	2027025
11	34459425
12	654729075
13	15058768725
14	376469218125
20	8200794532637891559375

もはや網羅的探索は不可能

発見的探索

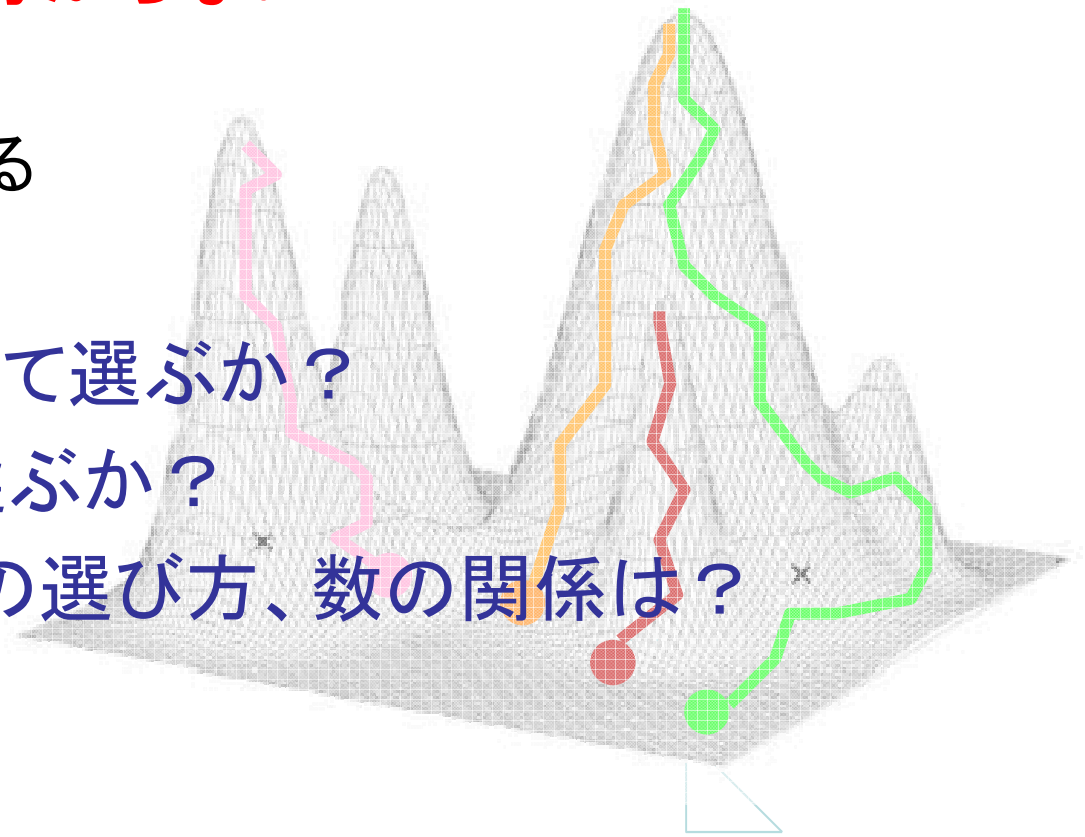
- 実データ解析で網羅的探索は不可能、発見的探索を行う
  - 初期樹形からリアレンジメントを繰り返し、より尤度の大きい樹形を探索する（ヒルクライム）





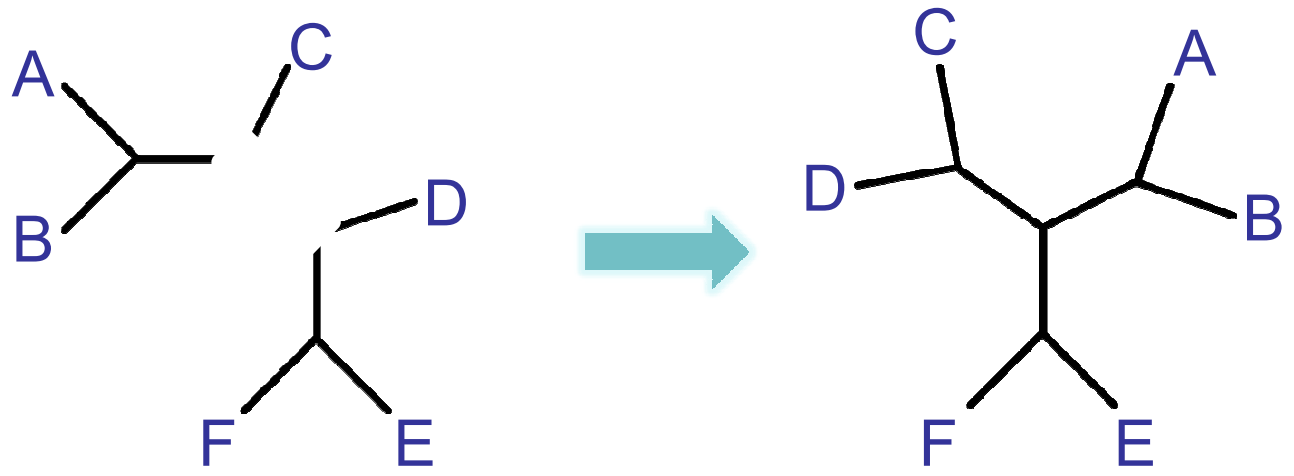
発見的探索のリスク

- すべての樹形を探索しないため
  Local optimaにトラップされる可能性がある
  - Global optimaがわからない
    → 間違えても、分からない
    - 初期点を増やす
    - 探索効率を上げる
  - 初期点をどうやって選ぶか？
  - 初期点をいくつ選ぶか？
  - 探索法と初期点の選び方、数の関係は？

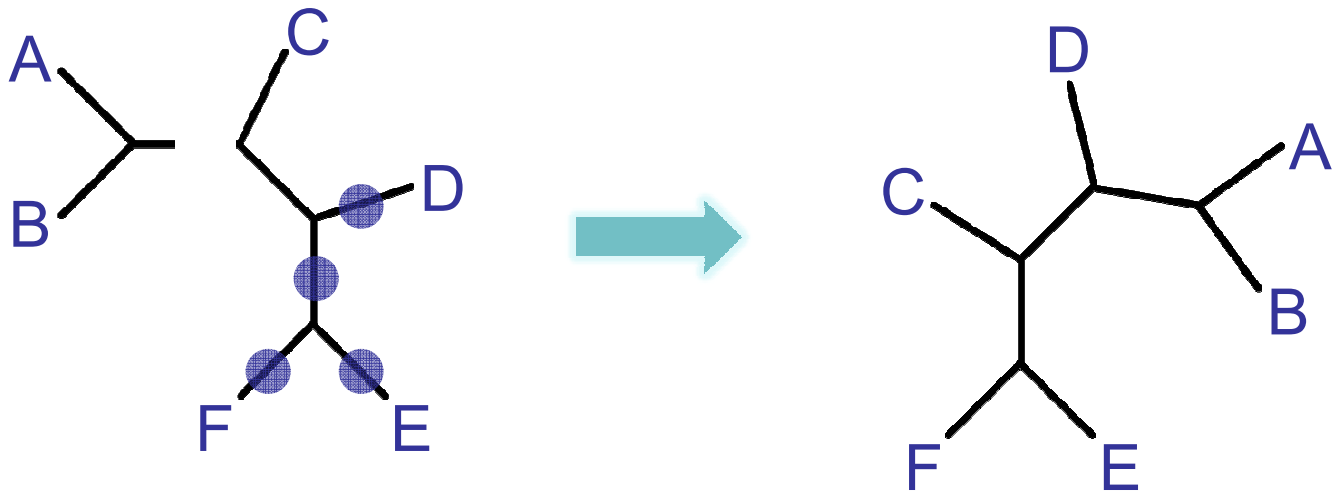


Heuristic tree search

- NNI — nearest neighbor interchanges



- SPR — subtree pruning & regrafting

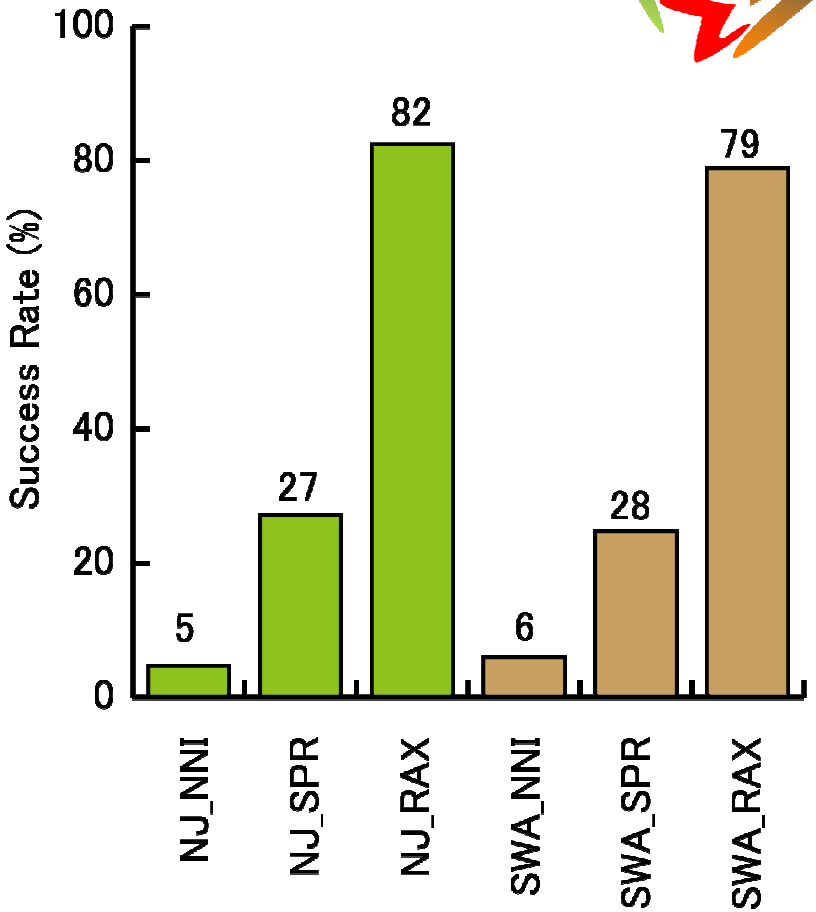


準実データによる解析

- シミュレーションデータ
  - あらかじめ「正しい」系統樹が分かっている
    - 発見的探索が「正解」に到達したか判定可能
  - 実データでの複雑な配列進化を再現できない
- 実データからブートストラップ法で解析するデータを作成
  - 実データと同様の複雑な配列進化をもつ
  - 前もって「正解」が分からない
    - 各データから前もってML系統樹を網羅的探索
    - 1データあたり2027025樹形の対数尤度を計算
  - 発見的探索結果と網羅的探索結果を比較し、「正解」に到達したか判定

樹形探索効率の評価（の一部）

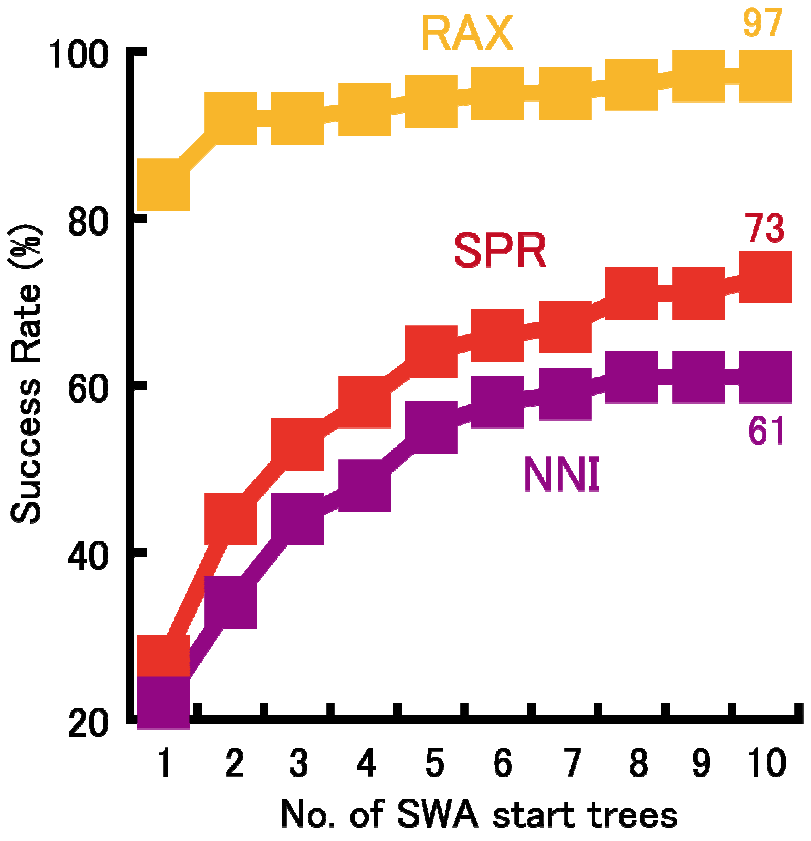
- Styデータ
  - 89個のブートストラップデータ
  - 202万樹形／データ
- 発見的探索
  - 初期樹形
    - NJ or SWA法
    - １樹形からの探索
  - 発見的探索
    - NNI, SPR, and “advanced” SPR (RAX)



探索効率：NNI < SPR << RAX

樹形探索効率：初期点数の影響

- 発見的探索
  - 初期樹形：SWA法
    - 10点まで増やす
  - 発見的探索
    - NNI, SPR, and “advanced” SPR (RAX)



初期点増加により探索効率増加

SPRは~70%、NNIは~60%で頭打ち？

今後の課題

- Styデータの他、Mic、Arcデータの解析を総合し、投稿論文を作成する
  - (準)実データに対し、網羅的探索と発見的探索を組み合わせた、世界初の厳密な効率評価
  - 発見的探索効率が思って以上に低い
- 通常実データには11配列以上が含まれる
  - 網羅的探索は、10配列・202万樹形が限界
  - 実データサイズでの発見的探索の効率評価には、シミュレーションデータを用いるしかない
    - より複雑な配列進化モデルを使用したシミュレーションが必要