

知識工学

4回目：検索（その1）

- ☐ 検索の定義
- ☐ 検索の必要性：NP 問題
- ☐ 検索方法
- ☐ 横形検索
- ☐ 縦型検索
- ☐ ビーム検索

<http://www.cit.ics.saitama-u.ac.jp/~far/Lectures/KE/04kaime/>

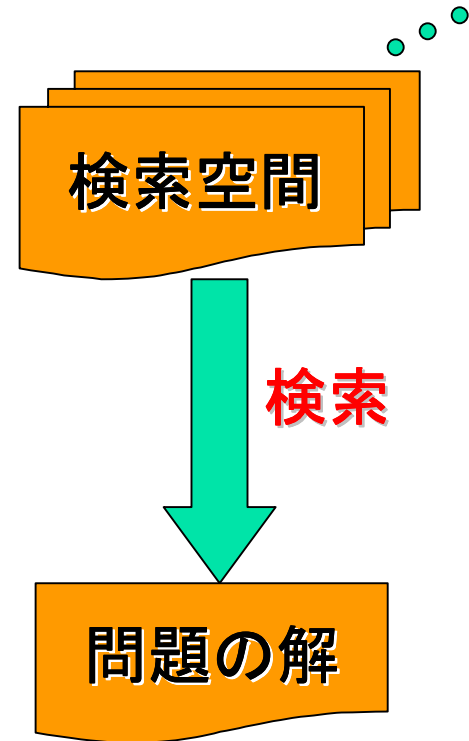
検索の定義

■ 定義

ある問題に対して多くの可能性の中から特定の解を見つけだすための戦略.

■ 概念

- 検索領域 (Search Space)
- 検索コスト (Search Cost)
- 評価関数 (Evaluation Function)





検索の必要性：NP 問題

状態組合わせの爆発問題 (Combinatorial explosion)

■ NP (noun-phrase) 問題

- 答えが与えられたら多項式オーダーの計算量で正しいかどうか確認できる問題からなるクラス

■ NP- hard 問題

- NP完全の性質すら持たないNP問題。NP完全問題よりも易しくない

■ NP- complete 問題

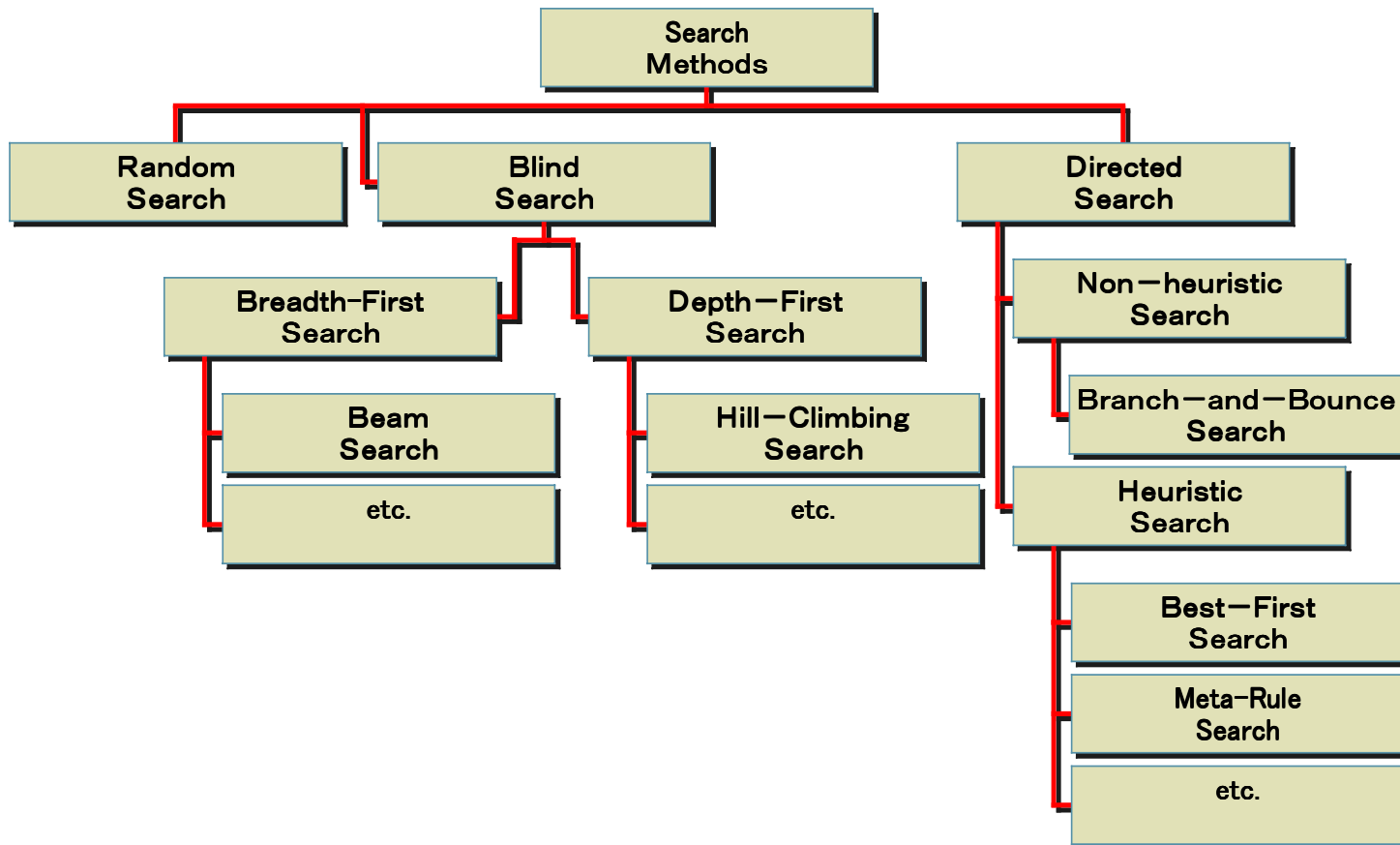
- 全てのNP問題が多項式オーダーの計算量でその問題に還元できるような問題



検索(サーチ)方法 — 1

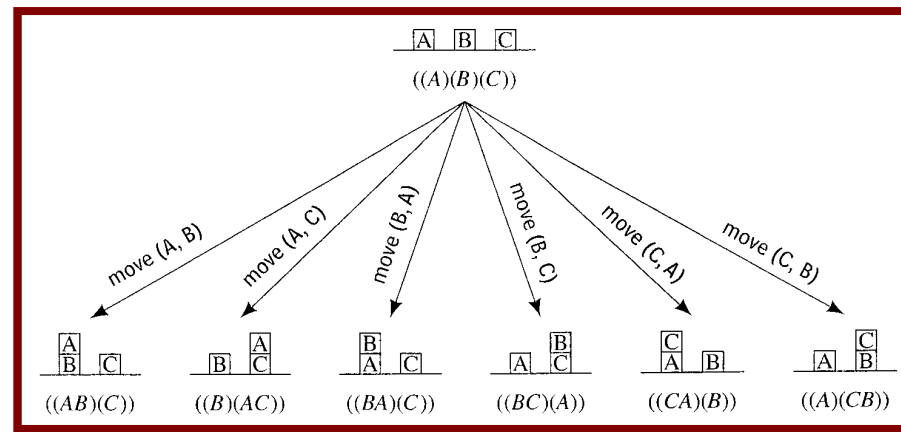
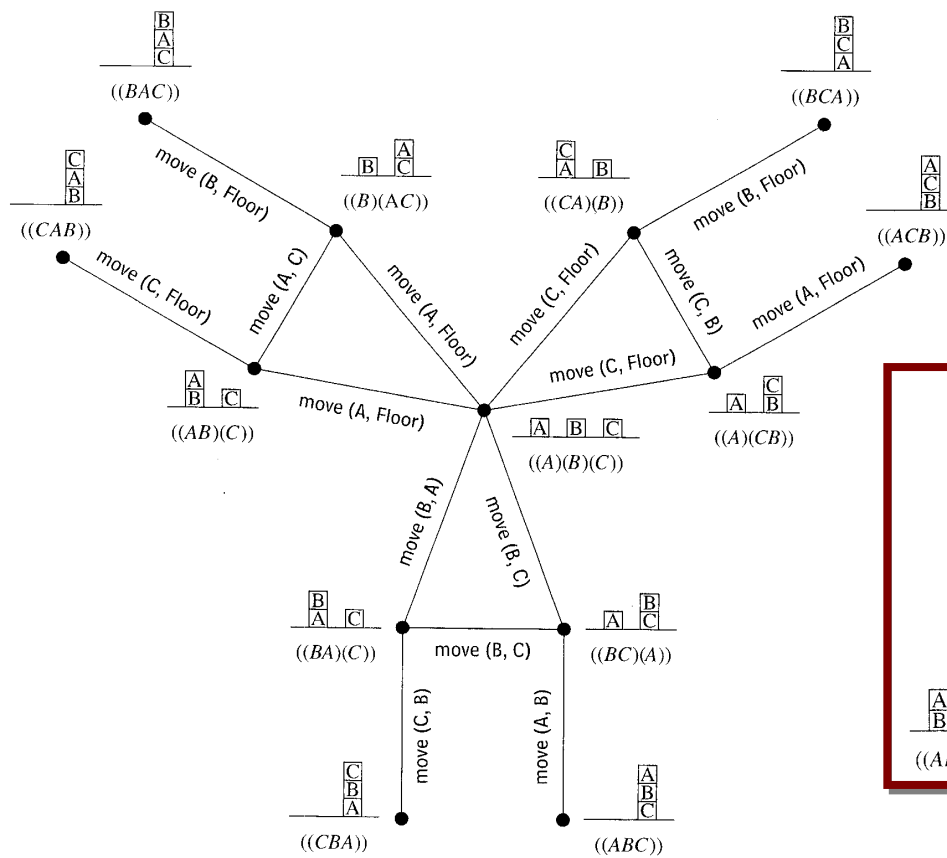
- ランダム検索 (Random Search)
- ブラインド検索 (Blind Search)
- 横形検索 (Breadth-First Search)
- 縦形検索 (Depth-First Search)
- ビーム検索 (Beam Search)
- 山登り検索 (Hill Climbing Search)
- 分岐限定法 (Branch-and-Bound Search)
- 最良優先検索法 (Best-First Search)
- その他

検索(サーチ)方法 - 2



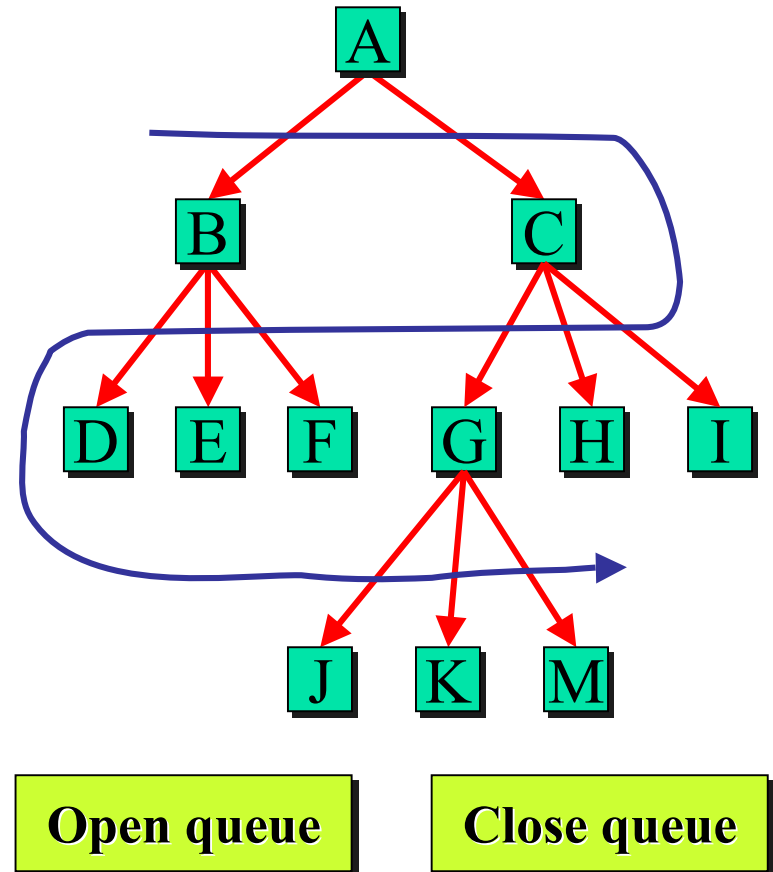
例：積み木問題 その1

- 積み木問題の完全な状態推移グラフ (State-Space Graph)

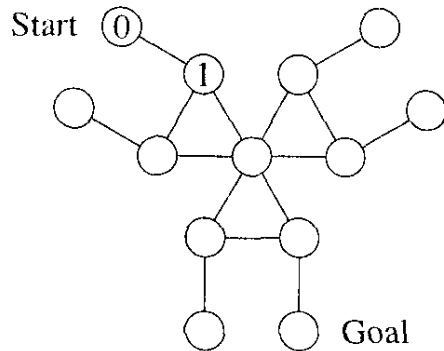


横形検索 (Breadth-First Search)

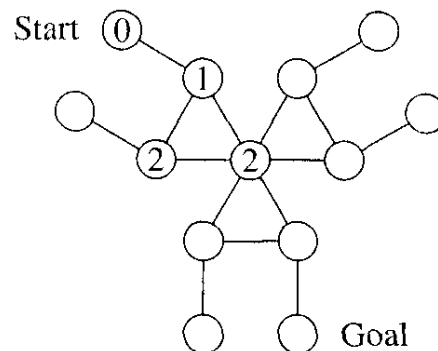
- A から出発しにAの子節点すべてを展開する。
- 死節点が見付かるとき止まる。目的節点が見付かるまでにこれを繰り返す。
- 待ち行列 (queue) による実現。



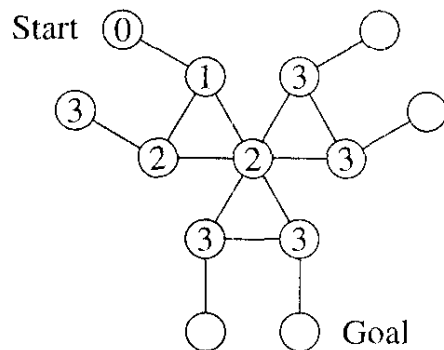
例：積み木問題 その2



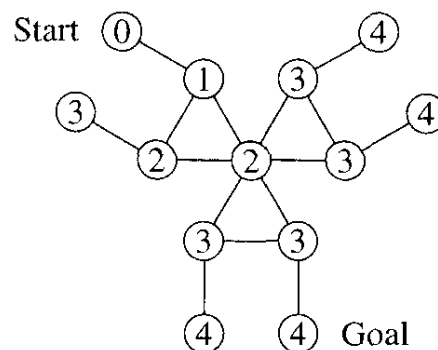
(a)



(b)



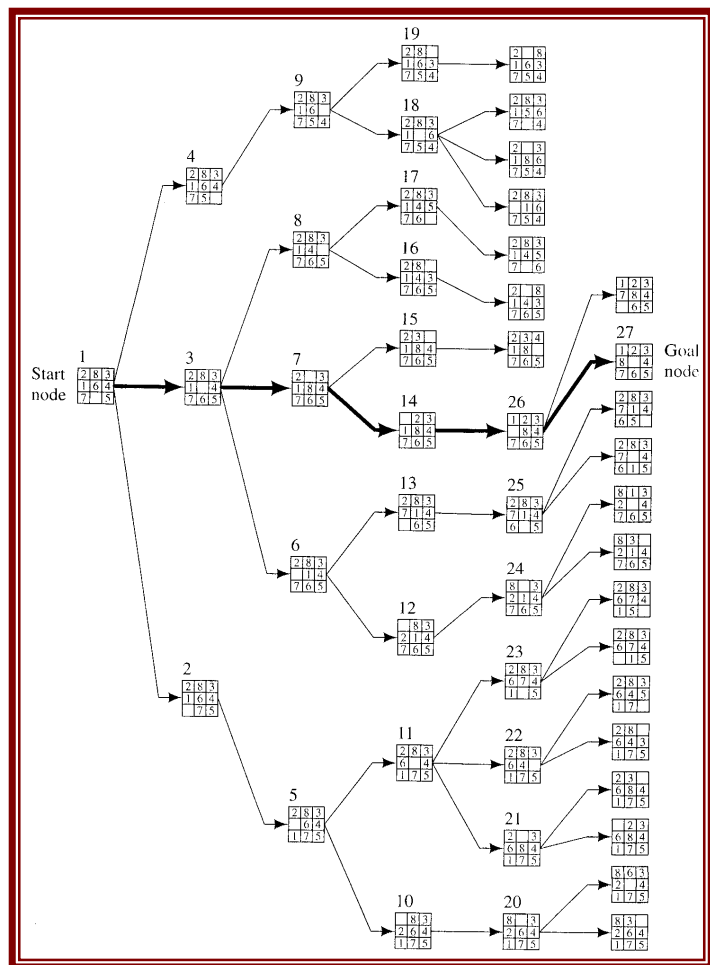
(c)



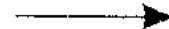
(d)

- 積み木問題の横型検索
- 出発点 ((BAC))
- ゴール ((ABC))

例: 8-Puzzle問題 その1



2	8	3
1	6	4
7		5

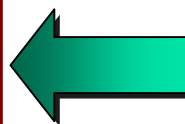


1	2	3
8		4
7	6	5

初期状態

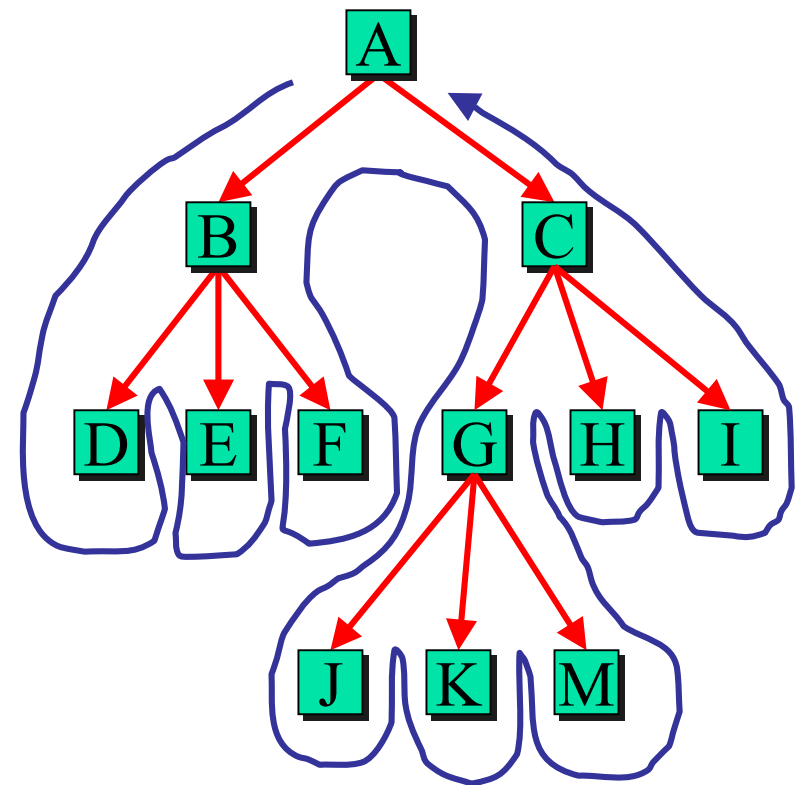
目的状態

横型検索の状態推移
グラフ



縦形検索 (Depth-First Search)

- A から出発しにAの子節点のうち1つを選んで展開する。
- 死節点が見つかるとき後戻り(backtrack)制御を行う。目的節点が見つかるまでにこれを繰り返す。
- スタック(stack)による実現。

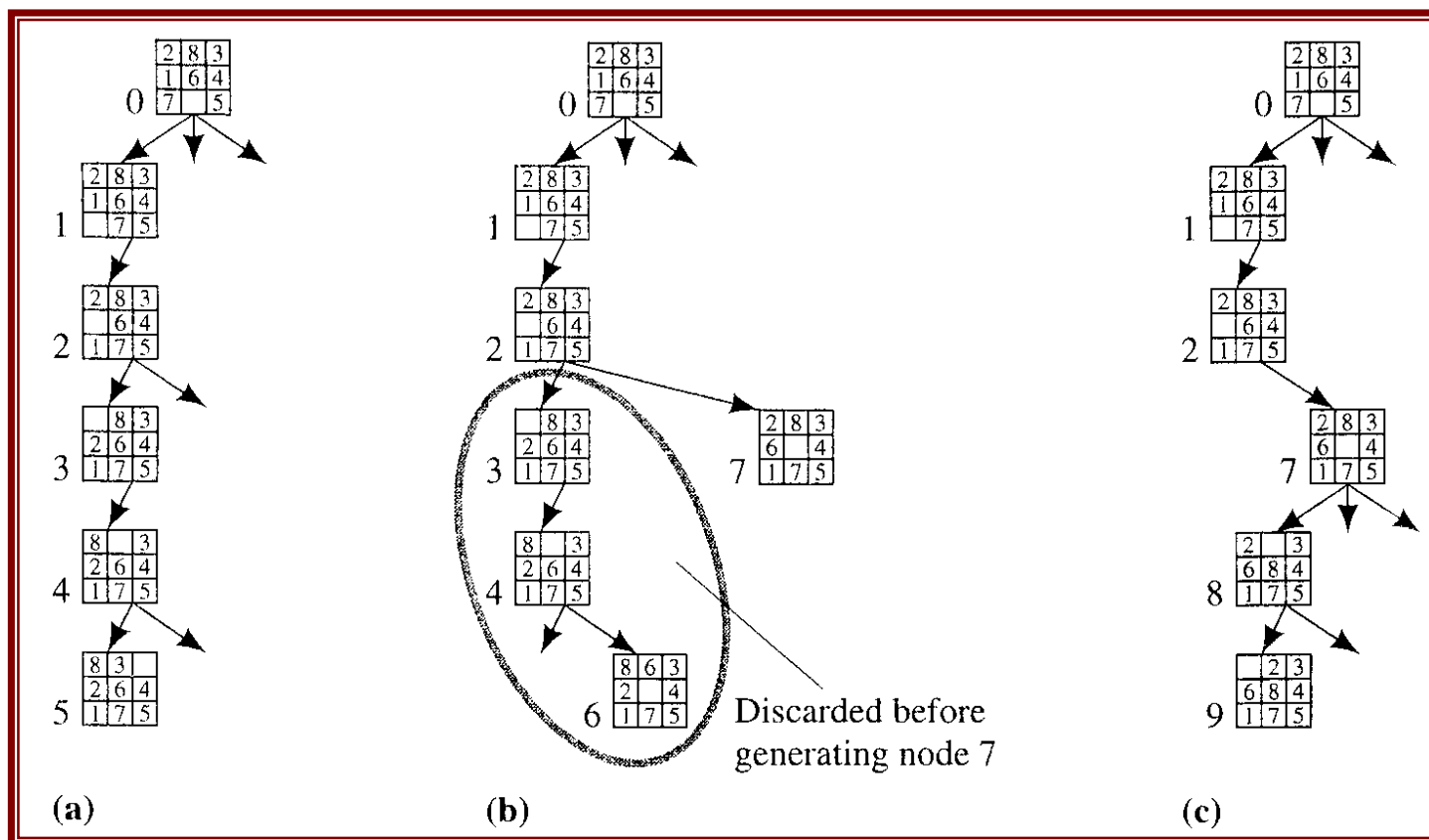


Open stack

Close
queue or stack

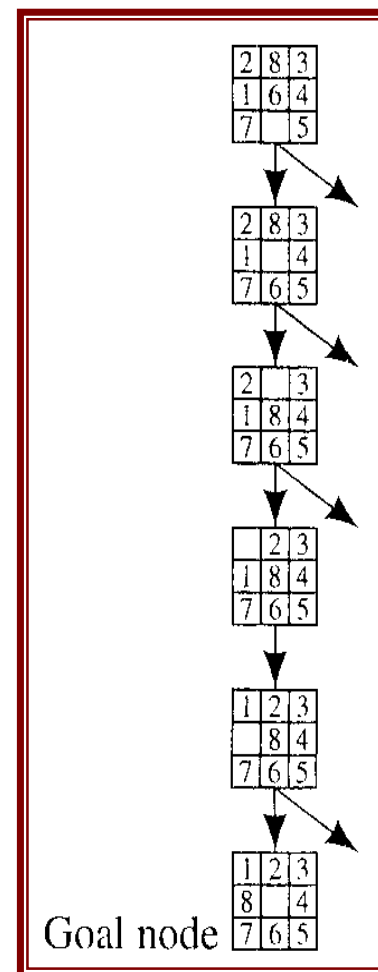
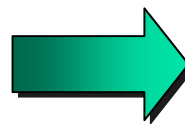
例: 8-Puzzle問題 その2

縦型検索の状態推移(一部)



例：8-Puzzle問題 その3

- 出発点から目的までの縦型検索の状態推移グラフ
- Backtrack直前の枝はすべて削除される



ビーム検索 (Beam Search)

- 横形検索の一種類
- A から出発しにAの子節点一部だけを展開する.
- 死節点が見付かるとき止まる.
- 目的節点が見付かるまでにこれを繰り返す.

