2007年度 計算知能論A
ヒューリスティクス探索

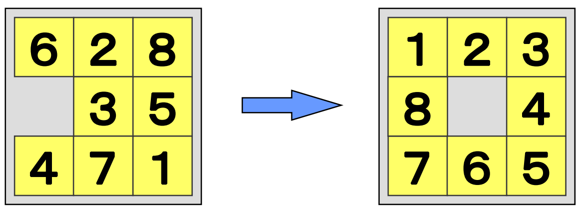
2007年6月11日

上田 和紀
早稲田大学理工学部CS学科

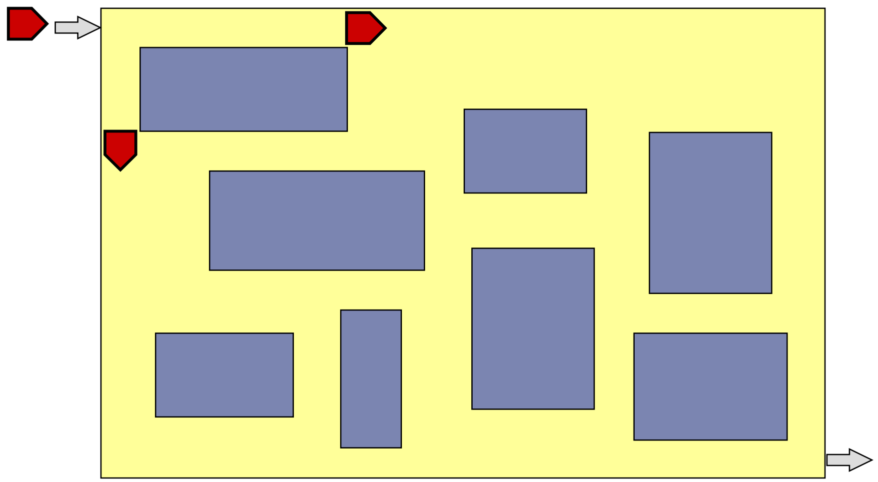
ヒューリスティクス (heuristics)

- ヒューリスティクス ＝ “rule of thumb”
  - （経験則，目の子算，大雑把なやり方）
  - *“Eureka!” (I’ve found it!)　　— Archimedes*
- ヒューリスティクス探索 ＝ 先読みをせずに状況の良さを評価する「静的評価関数」を利用して，開いた節点の展開順序を決めること
- 静的評価関数は正確でなくてもよい
  - それでも役に立てばよいし，実際役に立つことが多い

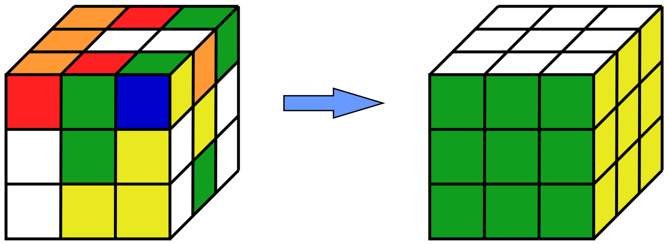
８パズルの静的評価関数



ロボットの歩行経路の静的評価関数



Rubik’s Cube の静的評価関数



最良優先探索 (best-first search)

- すべての開いた節点のうちで目標に最も近そうなものを展開する
  - 開いた節点を，静的評価関数に基づく優先度つき待ち行列（priority queue）として管理
  - Greedy algorithm（貪欲法）の一例
    - 過去は顧みずに良さそうな方向に邁進
  - 最適解（初期状態から目標状態に至るコスト最小のパス（path））を見つける保証はない

A* 探索

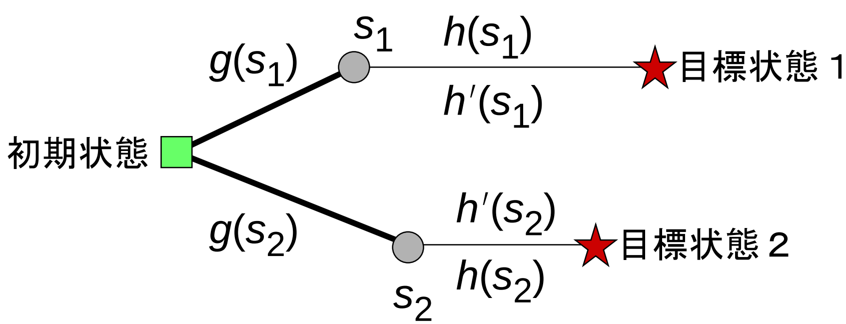
- ヒューリスティクスを利用して最適解を効率よく見つける方法

$g(s_k)$： 初期状態から状態 s_k へ至るコスト（≥ 0）
$h(s_k)$： s_k から目標状態へのコスト（≥ 0）
$h'(s_k)$: s_k から目標状態へのコストの見積り（≥ 0）

  - Open list には同一の s_k が複数個含まれるかも知れないが，各 s_k には経路情報（初期状態からいかに s_k に到達したか）が付随していると解釈

A* 探索

$g(s_k)$： 初期状態から状態 s_k へ至るコスト（≥ 0）
$h(s_k)$： s_k から目標状態へのコスト（≥ 0）
$h'(s_k)$: s_k から目標状態へのコストの見積り（≥ 0）



A* 探索

- 状態 s_k の「良さ」の尺度として

  $f(s_k) = g(s_k) + h'(s_k)$

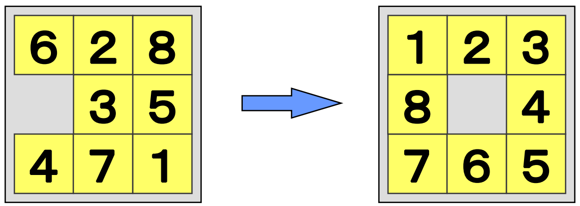
  を採用
- 開いた節点（= open list 内の状態）の中で f の値の最も小さなものを展開
  - $g(s_k)$ だけに基づく探索は幅優先探索（あるいはその一般化）
  - $h'(s_k)$ だけに基づく探索は最良優先探索
  - 【注意】展開で新たに得られた節点が目標状態か否かの検査は，待ち行列から取り出した後に行う

A* 探索

- 目標状態へのコストの見積り h' はどのような関数がいいか？
  - h になるべく近い関数であること
  - 楽観的な見積り（過小評価）であること

    つまり $\forall s\ (0 \le h'(s) \le h(s))$
- なぜ悲観的な見積り（過大評価）は望ましくないのか？

８パズルの A* 探索



- h'_1：最終位置にないタイルの枚数
- h'_2：各タイルの現在位置と最終位置とのマンハッタン距離の和

ヒューリスティクスの効果（8パズル）

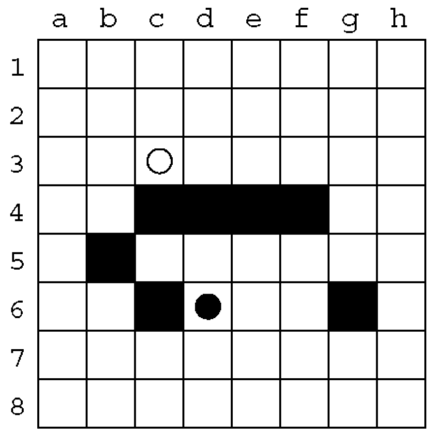
	訪れた節点の数		
d	反復深化	A* (h'_1)	A* (h'_2)
2	10	6	6
4	112	13	12
6	680	20	18
8	6384	39	25
10	47127	93	39
12	364404	227	73
14	3473941	539	113

(Russel and Norvig, 1997)

練習

- ● から ○ への経路を
  - 幅優先探索
  - 最良優先探索
  - A*

  でそれぞれ見つけよ．ただし塗りつぶした区画は通れない．



A* 探索の基本定理

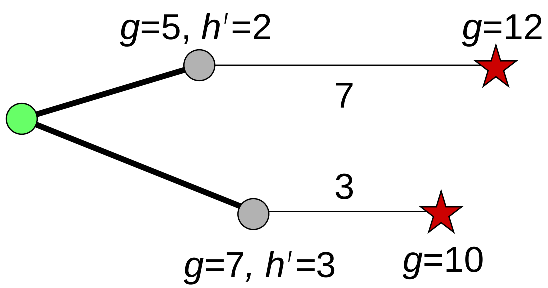
(i) 探索木のどの枝のコストも非負，かつ h' が楽観的な見積りならば，A* 探索が最初に見つける解は最適解である

(ii) どんなコスト c についても， $g(s) < c$ であるような状態 s がたかだか有限個しかなければ (*)，問題に解がある限り，A* 探索は必ず最適解を見つけて停止する

- 探索木の分岐係数が有限で，かつ各枝のコストが ε (>0) を下回らないという「普通の」状況では，前提条件 (*) は満たされる

A* 探索の基本定理

- 注意： A* 探索は，展開した節点を優先度つき待ち行列に入れる時点ではその節点が目標状態かどうかは検査しない．待ち行列から取り出した時点で検査する．



反復深化 A* 探索（IDA*）

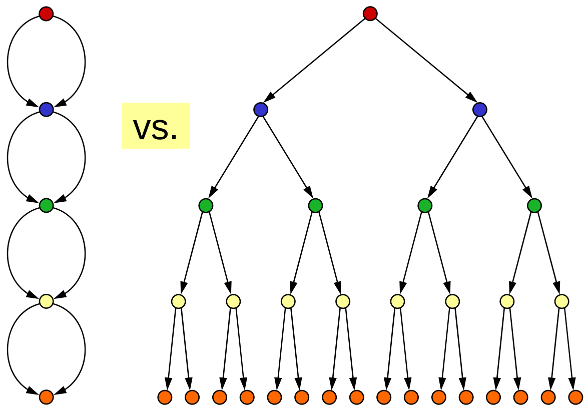
- A* 探索は，開いた節点の待ち行列を使う点で幅優先探索と同じ欠点をもつ
- 代案： $f(s) = g(s) + h'(s)$ の値に上限を設けた深さ優先探索を繰返し行なう

  ➔ 反復深化 A* 探索

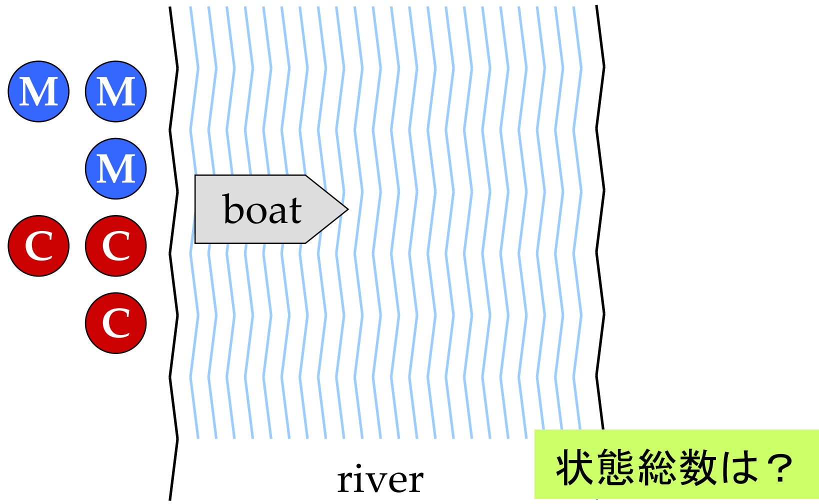
  - 状態数が多いヒューリスティクス探索問題には効果的

グラフの探索と vs. 木の探索

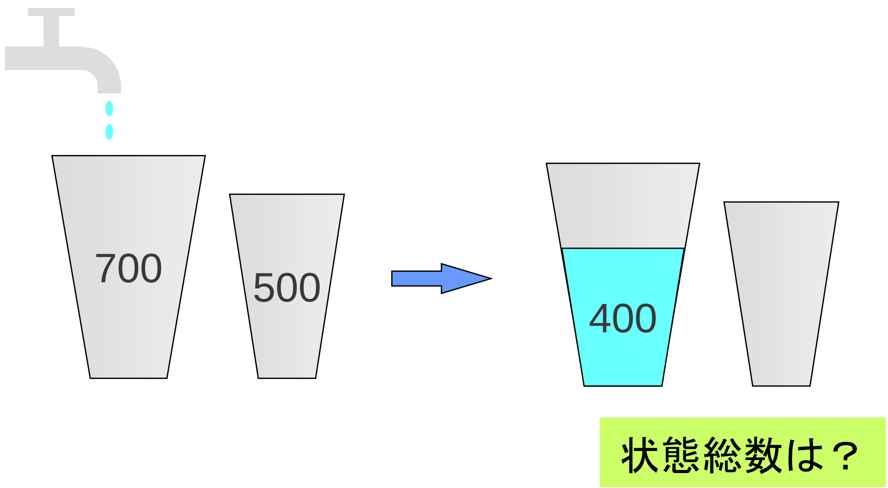
- グラフを木に展開すると，指数関数的に節点数が増加することがある
  - 極端な例：



Missionaries and Cannibals



水差しの問題



探索木における状態の重複

- 探索グラフを探索木に展開すると（＝ 状態の重複を気にせずに探索を進めると）…

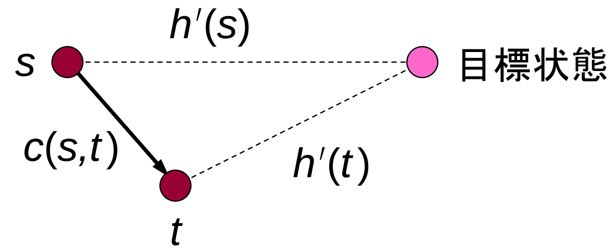
- 問題点１：同じパス上に同一状態が複数回現れる（探索グラフが閉路をもつ場合）
  - 検出は容易 — 初期状態から展開しようとしている節点に至るパスを覚えておけばよい

探索木における状態の重複

- 問題点２：異なるパス上に同一状態が現れる（同一状態に至る複数の方法が存在）➔ 二つのケース
  - a. 新たに見つけた節点が既存の開いた節点と同一
    - ➔ open list を探索すれば検出可能
  - b. 新たに見つけた節点が既存の閉じた節点と同一
    - ➔ 過去に展開したすべての節点をハッシュ表などに覚えておけば検出可能
    - しかしグラフの A* 探索では，同一節点を複数回見つけた場合，（h' が楽観的だという条件だけでは）最初に見つけたときの経路が最適であることは保証されない

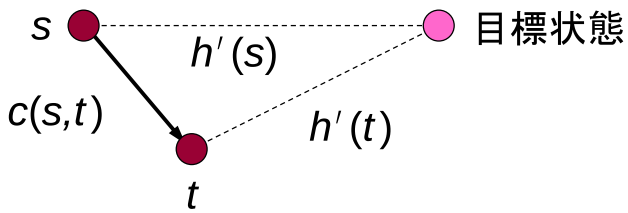
A* 探索ふたたび

- 静的評価関数 $h'(s)$ が単調とは：
  - 節点 s から隣接節点 t への操作コストを $c(s,t)$ とすると，$h'(s) - h'(t) \leq c(s,t)$ であること（cf. 三角不等式）



A* 探索ふたたび

- 静的評価関数 $h'(s)$ が単調だと：
  - k 番目に訪れる節点を s_k とすると，$f(s_k) = g(s_k) + h'(s_k)$ は k について単調非減少
  - グラフの A* 探索で節点 s を展開しようとするとき，s に付随する経路は最適経路である（最適でないコストをもつ節点を展開することはない）



A* 探索ふたたび

- $h'(s)$ が単調でないと
  - 遠回り経路で見つかった節点が展開されてしまうことがある