

人工知能入門

-探索による人工知能-

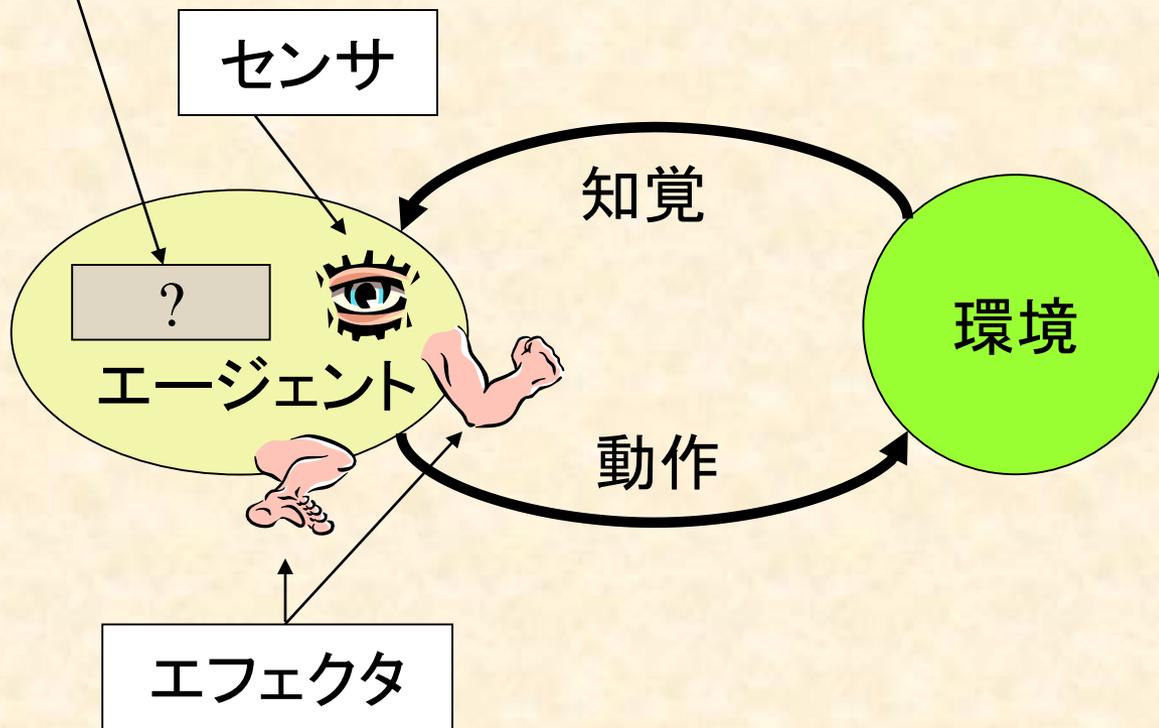
Lecture 12

敵対探索: Alpha-beta枝刈りと評価関数

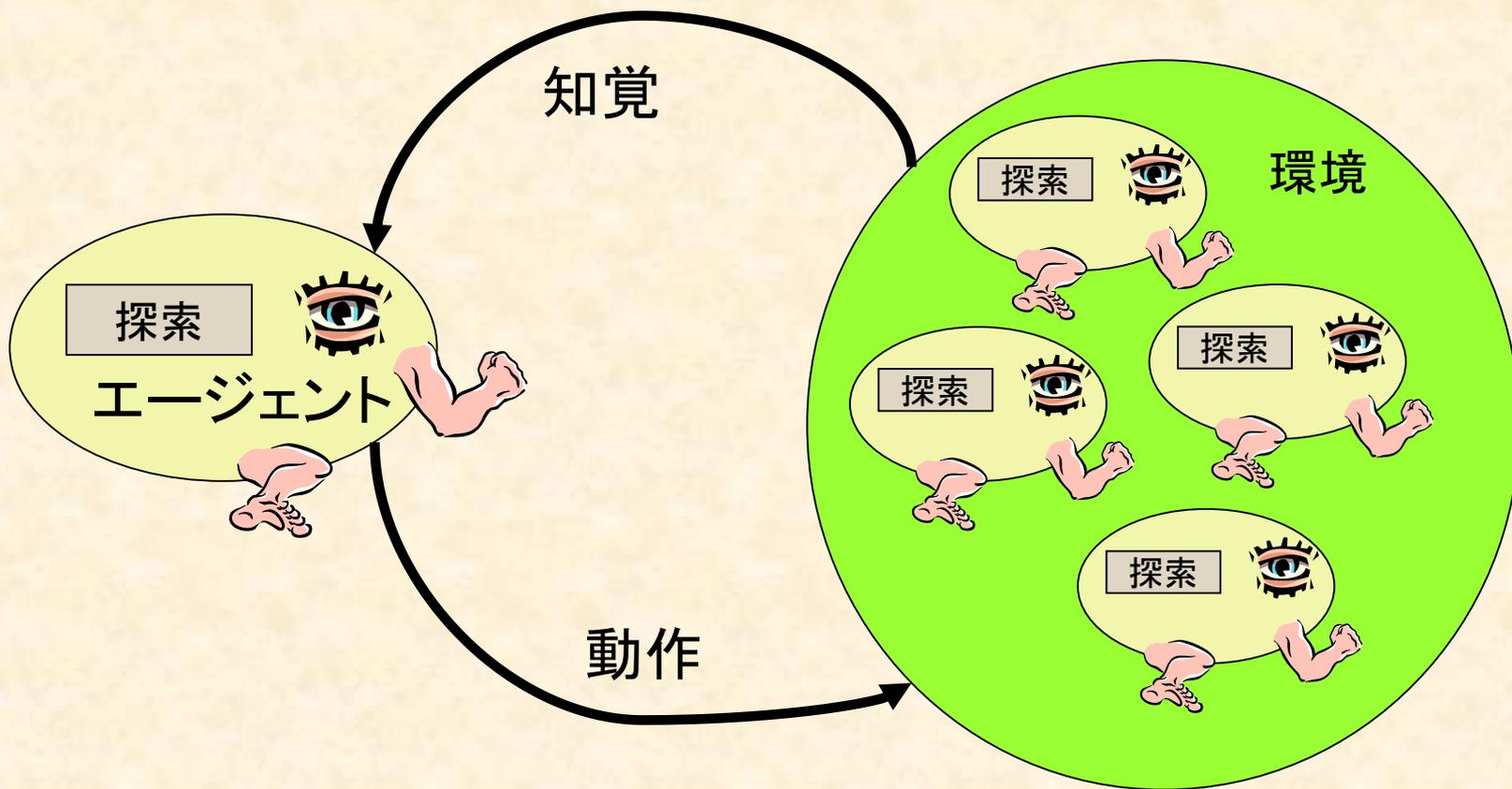
<http://www2.teu.ac.jp/gamelab/LECTURES/ArtificialIntelligence/index.html>

合理的エージェント

この設計方法は？



競合の環境



ゲームの中の敵対探索

- ★ ゲームが持っている抽象性は興味深い研究対象
 - ◆ ゲームの状態は表現が容易
 - ◆ かなり少数の動作しかできないようになっている
 - ◆ 競合的な環境の理想モデル

二人ゲームにおける完全な決定

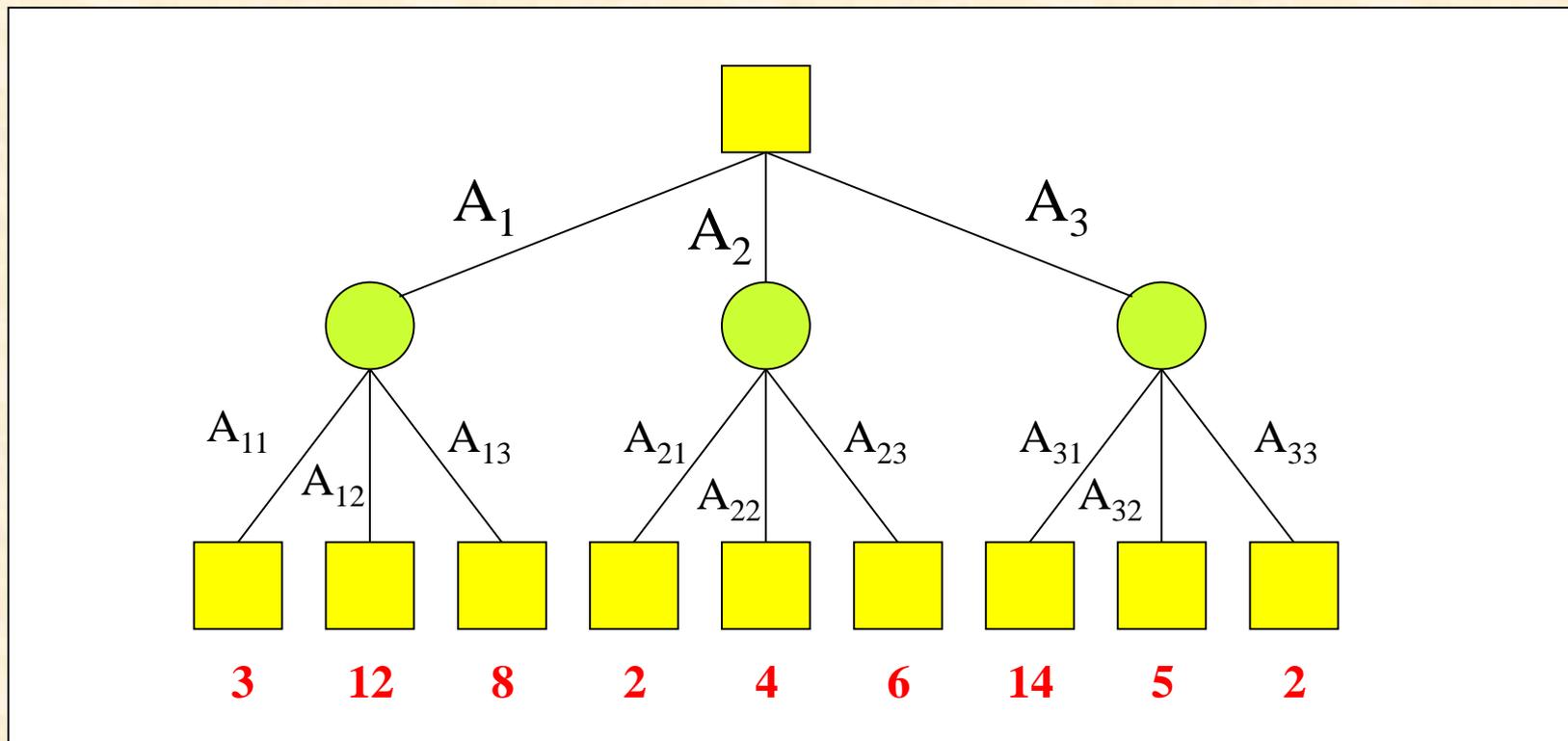
★ 二人でプレイするゲームの一般的な場合

- ◆ 二人をそれぞれをMAXとMINと呼ぶ
- ◆ MAXが先手
- ◆ ゲームが終わるまで交互に指し続ける
- ◆ ゲームが終わったときにある得点が勝者に与えられる

★ ゲームの問題定式化

- ◆ 初期状態: 盤面の状態とどちらの手番かの指示
- ◆ オペレータ: プレイヤーが指すことのできる合法手を定義する
- ◆ 終端テスト: ゲームの終了を決定する
- ◆ 効用関数(報酬関数): ゲームの結果を数値として与える

Min-max探索

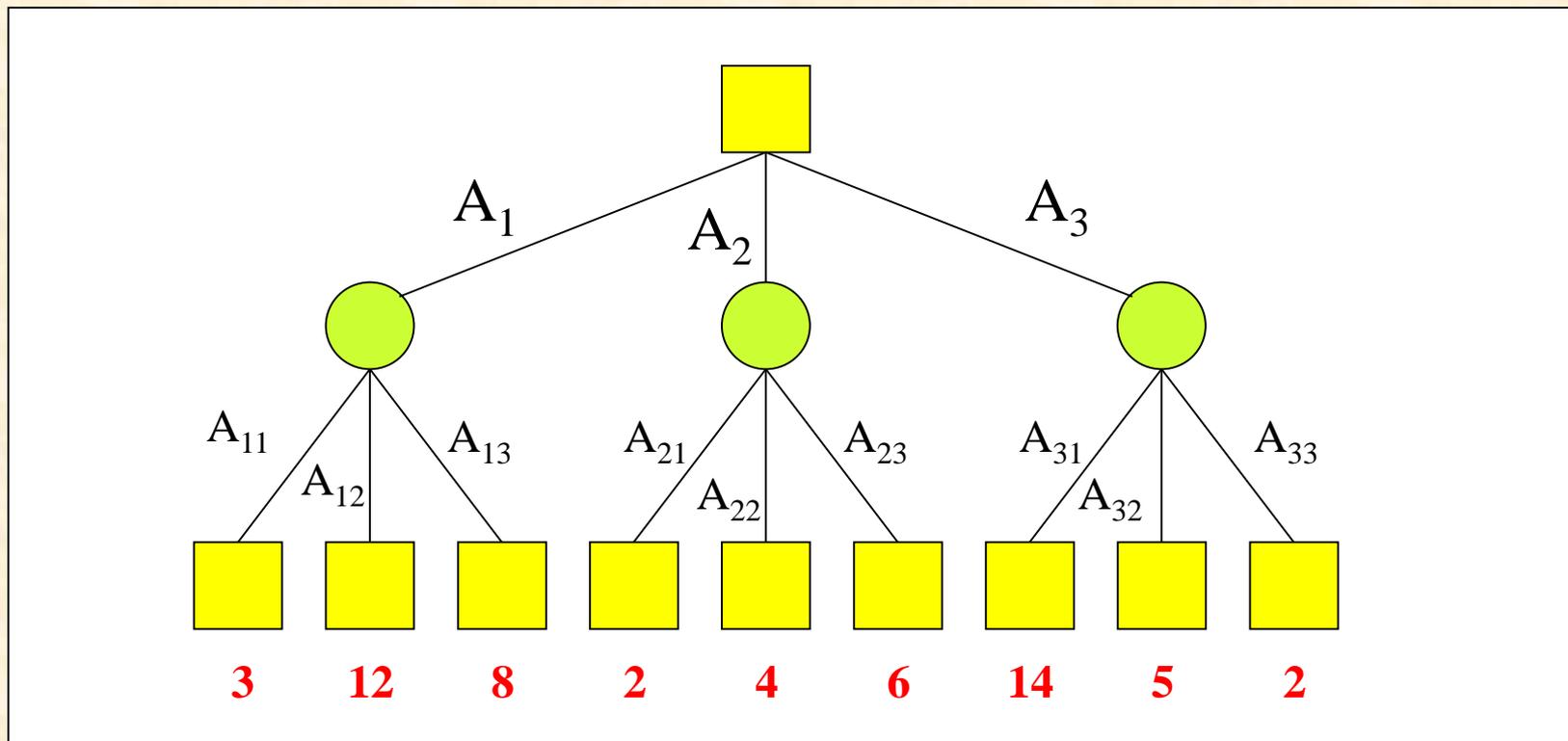


Alpha-beta枝刈り

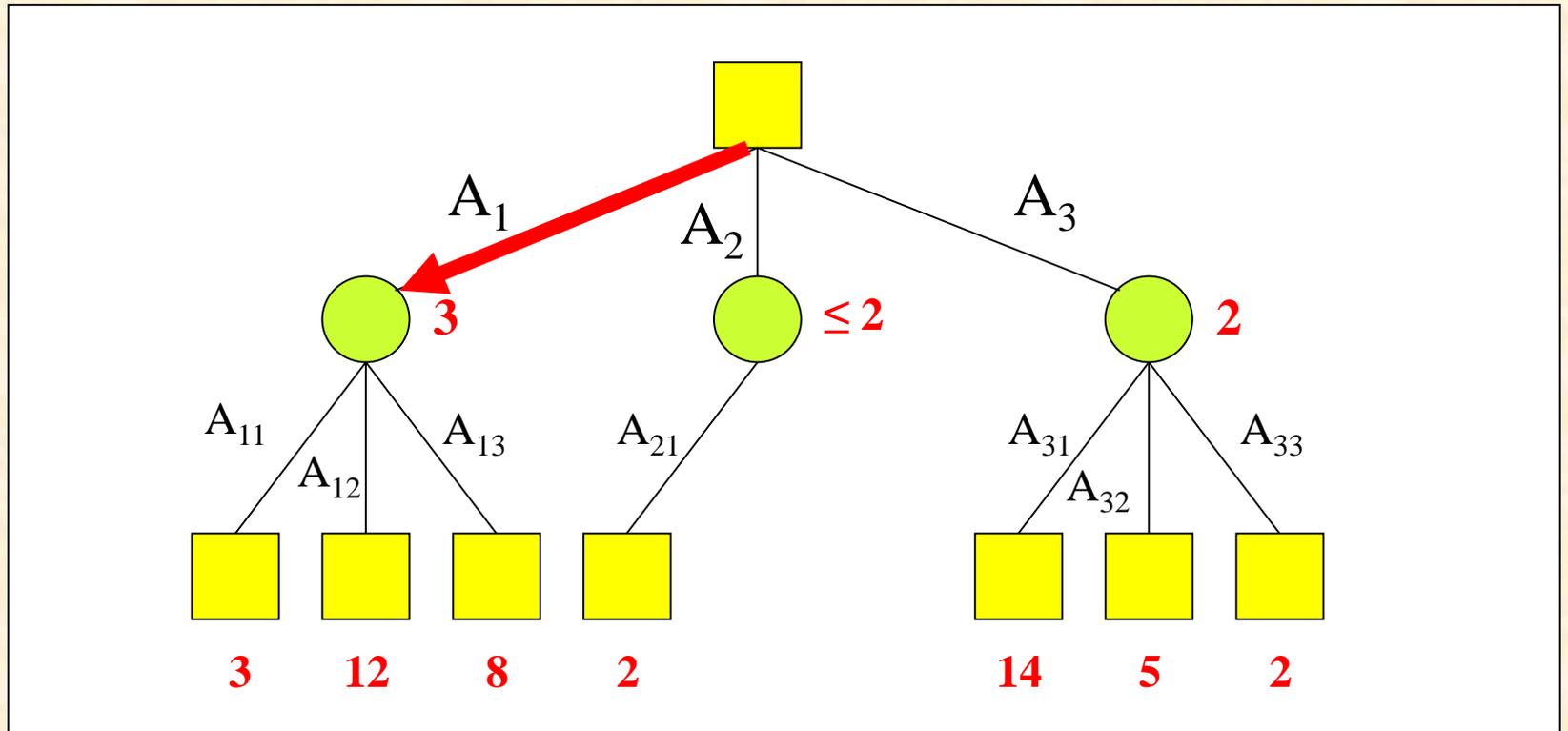
★ Min-max探索と同じ結果が早く終わる

◆ アイデア: 最終結果に関係ないノードを「刈る」

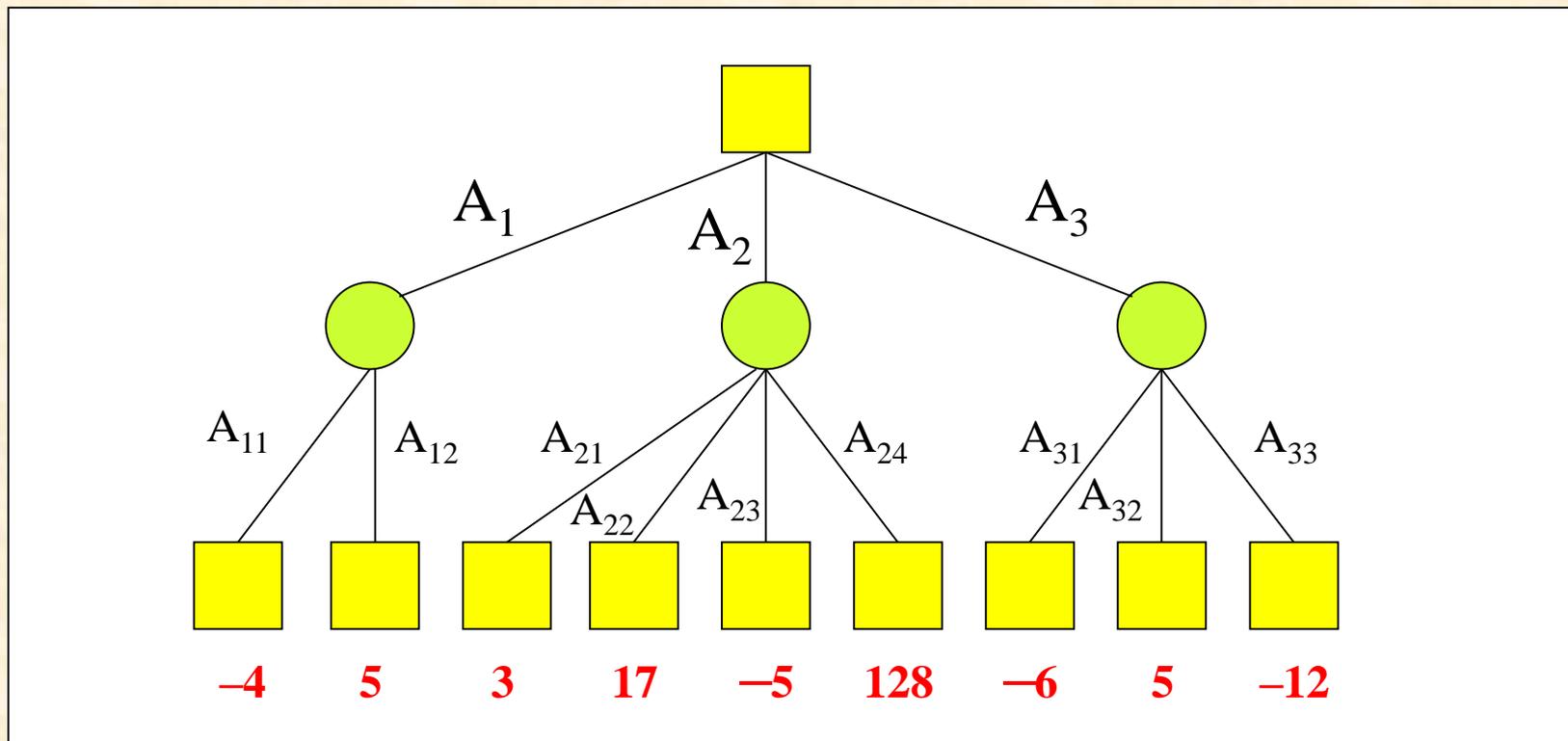
Min-max探索



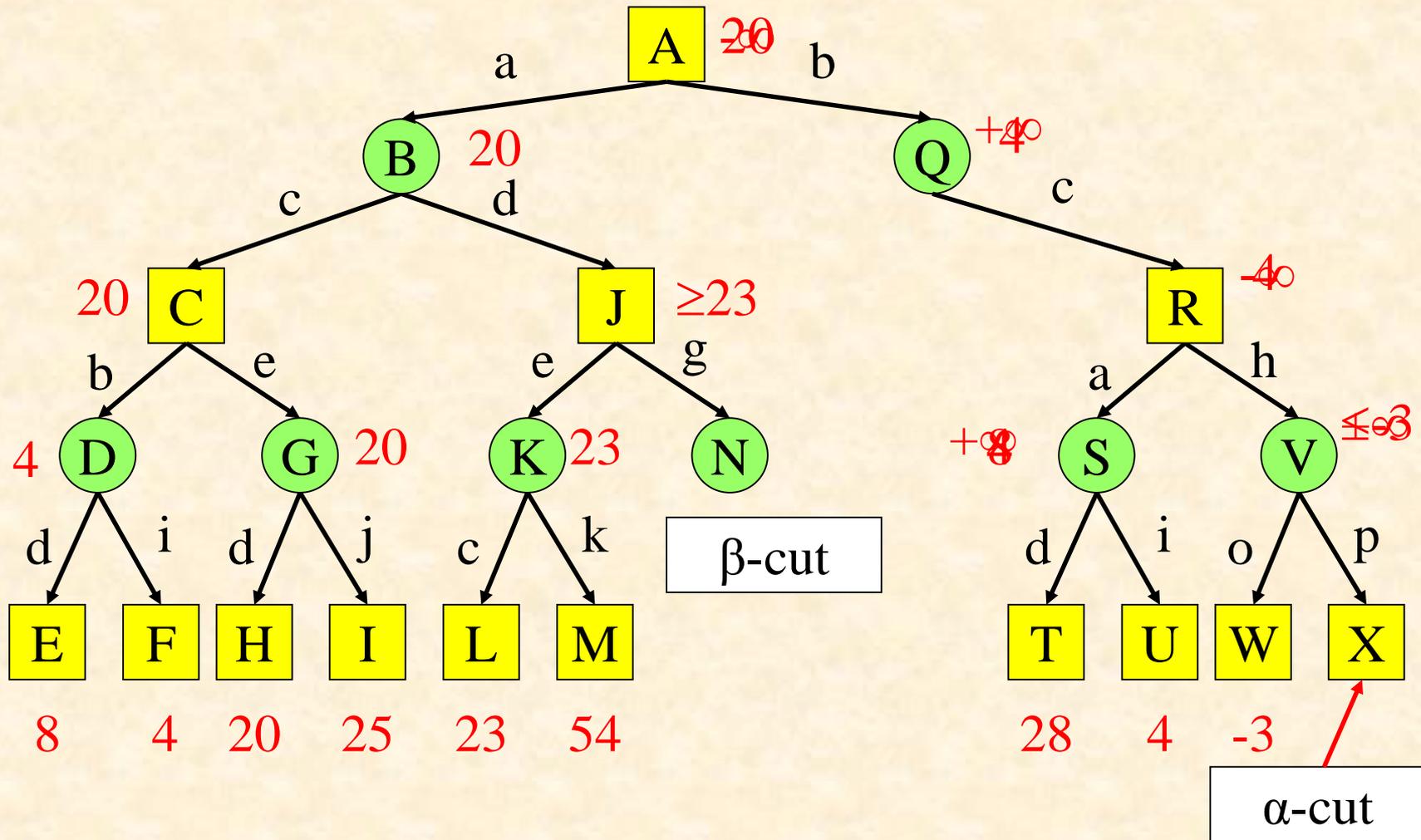
Alpha-beta枝刈り



演習問題 12-1: Alpha-beta 枝刈り



Alphaとbetaの違い



不完全な決定

✦ Min-max探索は無理

- ◆ ほとんどの場合、終端状態まですべての道筋を探索する時間がない

✦ Min-max探索を利用できる方法

- ◆ 早めに探索を打ち切る
- ◆ 探索木の葉に**評価関数**を適用する
- ◆ 効用関数を評価関に置き換える
- ◆ 終端テストは**打ち切りテスト**に置き換える

評価関数

✦ 評価関数

- ◆ ゲームのある局面から得ることが期待される効用の見積もり
- ◆ 将棋の例
 - 駒の価値:「歩」は1点、「金」は6点、「飛」は10点
 - 他の特徴:「玉の安全度」など

✦ ゲームプログラムの性能が評価関数の質に極めて存在している

- ◆ 終端ノードに対して効用関数と合致していなければならない
- ◆ あまり長い時間かかってはいけない
- ◆ 実際に勝つ可能性を正確に反映していなければならない
 - 同じ勝つ可能性がある局面は同じ評価になる

演習問題12-2:五目並べの評価関数

★ 演習問題:五目並べの評価関数には何が
必要かを討論してください

まとめ

✦ Alpha-beta枝刈り

- ◆ Min-max探索と同じ結果が早く終わる
- ◆ 最終結果に関係ないノードを「刈る」
- ◆ α 枝刈りと β 枝刈りの2種類がある

✦ 評価関数

- ◆ 終端状態まですべての道筋を探索する時間がない
- ◆ 途中で探索を打ち切って、効用関数の代わりに評価関数を利用
- ◆ ゲームのある局面から得ることが期待される効用の見積もり

ミニテストについて

★ 明日の朝は3回目のミニテストを行う

- ◆ 持ち込み可
- ◆ 遅刻しないように
- ◆ **内容**: ヒューリスティック探索と敵対探索 (第9回～第12回)

最終日についてのお願い

- ✦ 五目並べの大会をスムーズに実施するためにフラッシュメモリを持ってきて下さい