Lecture 11: Minimax Search

Artificial Intelligence
CS-GY-6613

Julian Togelius
julian.togelius@nyu.edu

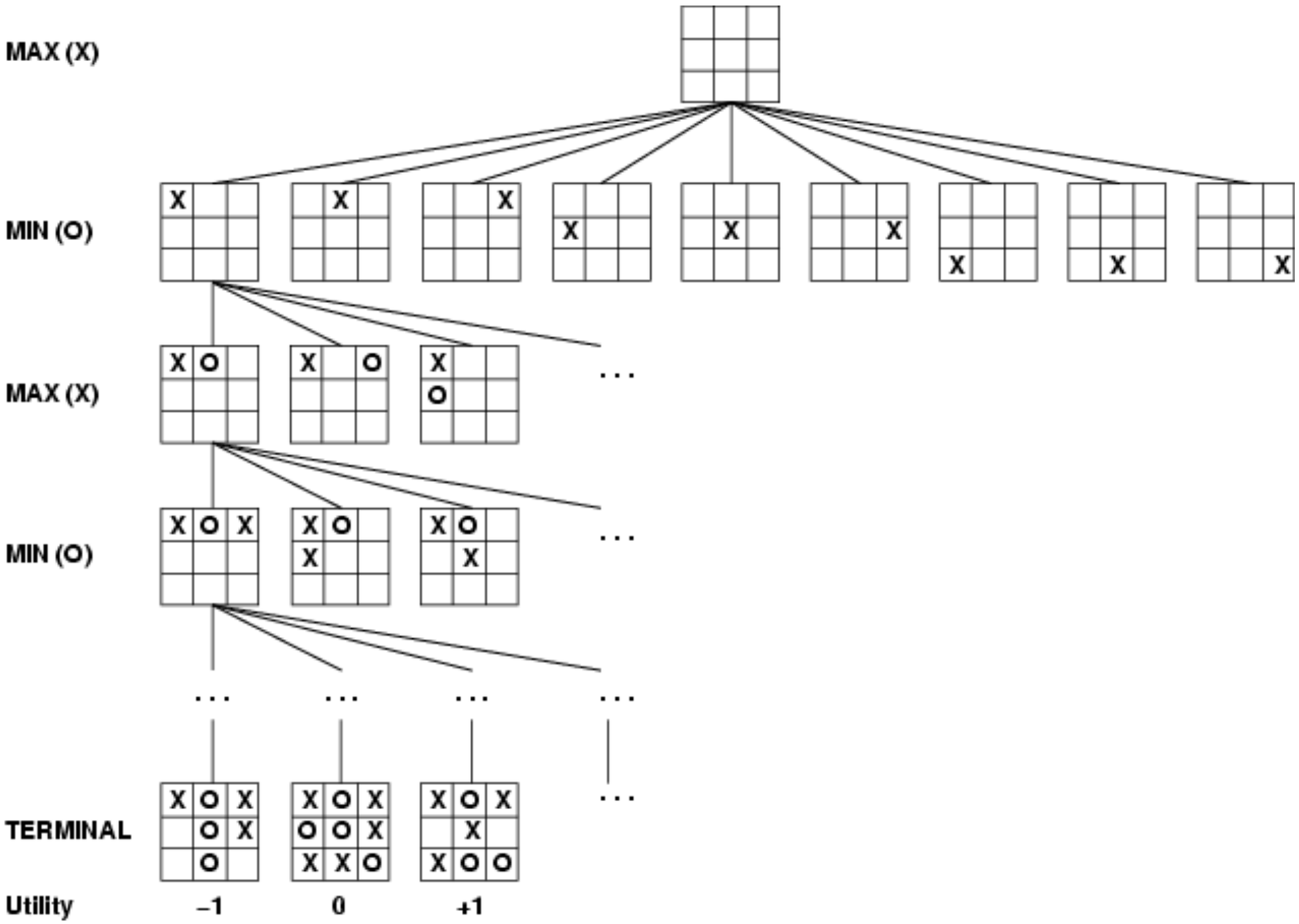
Adversarial problems?

- So far, problems have been deterministic with a known forward model
    - Though the methods would work with noise and/or imperfect models
- Adversarial: there is another agent, which works against you
    - The opponent is not predictable
- Game trees rather than search trees

Paradigm case

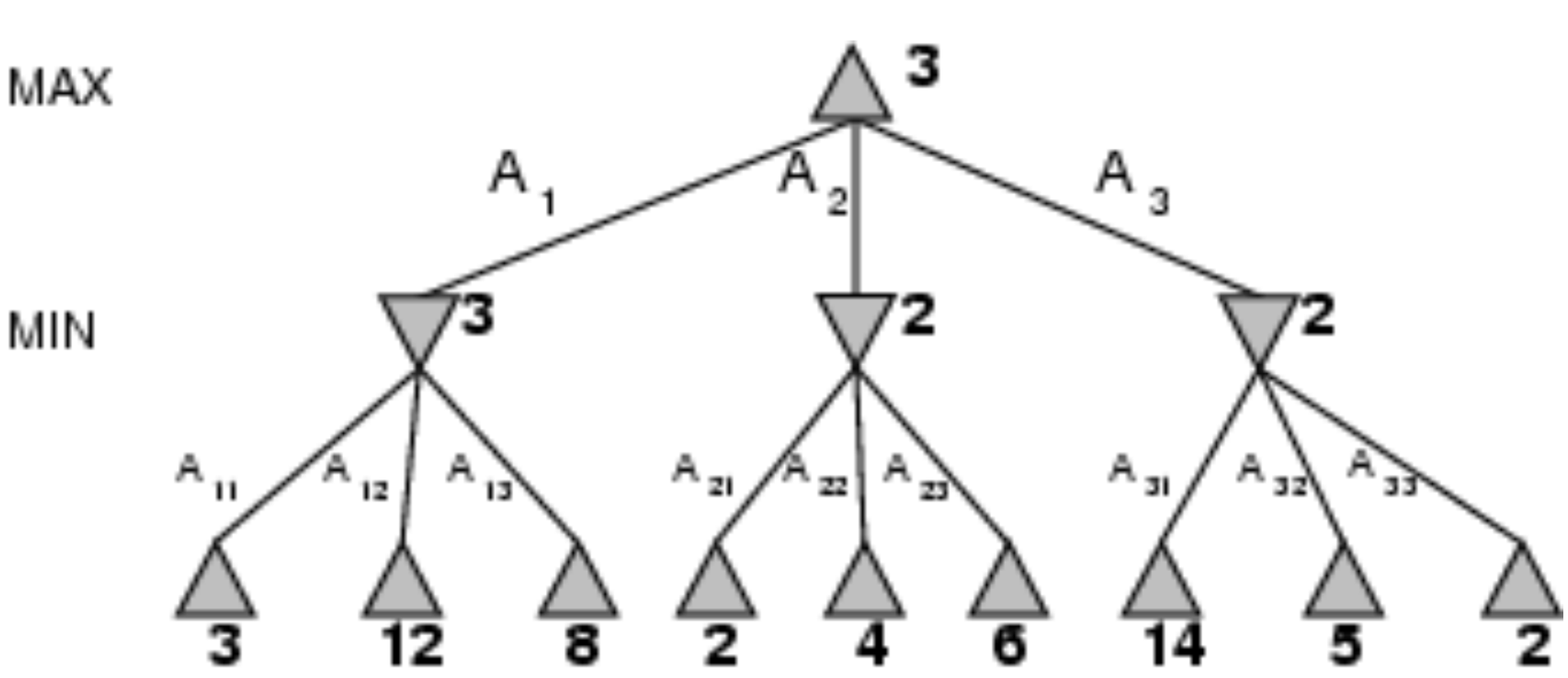
Other cases

Two-player game tree



Minimax

- Perfect play for deterministic games
- Choose move to state with highest minimax value (best payoff against perfect opponent)



```
function MINIMAX-DECISION(state) returns an action
    v ← MAX-VALUE(state)
    return the action in SUCCESSORS(state) with value v

function MAX-VALUE(state) returns a utility value
    if TERMINAL-TEST(state) then return UTILITY(state)
    v ← −∞
    for a, s in SUCCESSORS(state) do
        v ← MAX(v, MIN-VALUE(s))
    return v

function MIN-VALUE(state) returns a utility value
    if TERMINAL-TEST(state) then return UTILITY(state)
    v ← ∞
    for a, s in SUCCESSORS(state) do
        v ← MIN(v, MAX-VALUE(s))
    return v
```

```
MinMax (GamePosition game) {
  return MaxMove (game);
}

MaxMove (GamePosition game) {
  if (GameEnded(game)) {
    return EvalGameState(game);
  }
  else {
    best_move <- {};
    moves <- GenerateMoves(game);
    ForEach moves {
       move <- MinMove(ApplyMove(game));
       if (Value(move) > Value(best_move)) {
          best_move <- move;
       }
    }
    return best_move;
  }
}

MinMove (GamePosition game) {
  best_move <- {};
  moves <- GenerateMoves(game);
  ForEach moves {
     move <- MaxMove(ApplyMove(game));
     if (Value(move) > Value(best_move)) {
        best_move <- move;
     }
  }

  return best_move;
}
```

Properties of Minimax

- *Complete?* Yes (if tree is finite)
- *Optimal?* Yes (against an optimal opponent)
- *Time complexity?* $O(b^m)$
- *Space complexity?* O(bm) (depth-first exploration)

For chess, b ≈ 35, m ≈100 for "reasonable" games
–> exact solution completely infeasible