# Lecture 10: Imperfect Information

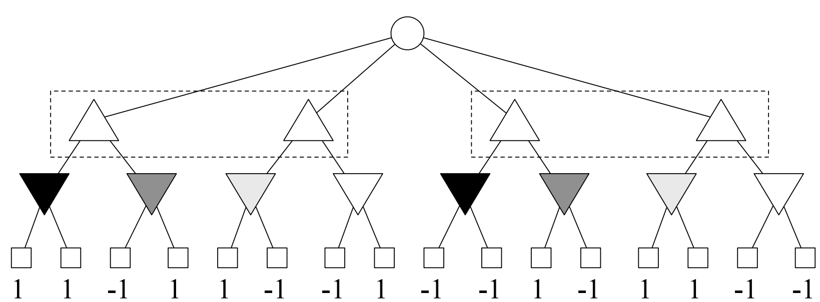
AI For Traditional Games
Prof. Nathan Sturtevant
Winter 2011

UNIVERSITY OF DENVER

## Imperfect Information

- So far, all games we've developed solutions for have perfect information
  - No hidden information such as individual cards
  - Hidden information often represented as chance nodes
    - Could be a play by one player that is hidden until the end of the game

## Example Tree




## What is the size of a game with ii?

- Simple betting game (Kuhn Poker)
  - Ante 1 chip
  - 2-player game, 3-card deck, 1 card each
    - First player can check/bet
    - Second player can bet/check or call/fold
    - If 2nd player bets, 1st player can call/fold
  - 3 hands each / 6 total combinations
  - [Exercise: Draw top portion of tree in class]

## Simple Approach: Perfect-Info Monte-Carlo

- We have good perfect information-solvers
  - How can we use them for imperfect information games?
- Sample all unknown information (eg a world)
  - For each world:
    - Solve perfectly with alpha-beta
  - Take the average best move
- If too many worlds, sample a reasonable subset

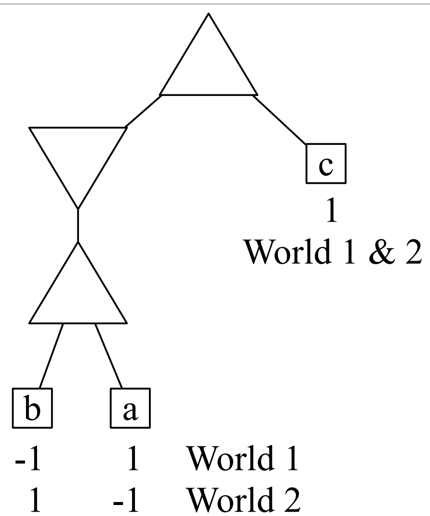


## Drawbacks of Monte-Carlo

- May be too many worlds to sample
- May get probabilities on worlds incorrect
  - World prob. based on previous actions in the game
- May reveal information in actions
  - Good probabilities needed for information hiding
  - Program has no sense of information seeking/hiding moves
- Analysis may be incorrect (see work by Frank and Basin)

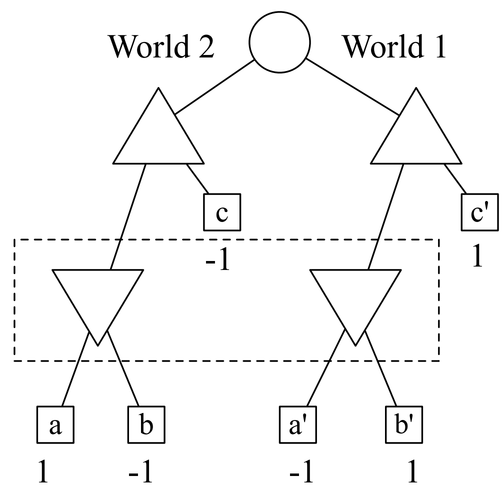


## Strategy Fusion






## Non-locality



## Strengths of Monte-Carlo

- Simple to implement
  - Relatively fast
- Can play some games very well
  - Approximates some games better than others
- How can we measure this?
  - Abstract model of a game

## Analysis of PIMC

- Understanding the Success of Perfect Information Monte Carlo Sampling in Game Tree Search
  - Jeffrey Long and Nathan R. Sturtevant and Michael Buro and Timothy Furtak

## Leaf Correlation

- (lc) With probability lc, each sibling pair of terminal nodes will have the same payoff value (whether it be 1 or -1). With probability (1 – lc), each sibling pair will be anti-correlated, with one randomly determined leaf having value 1 and its sibling being assigned value -1.

## Bias

- b: At each correlated pair of leaf nodes, the nodes' values will be set to 1 with probability b and -1 otherwise. Thus, with bias of 1, all correlated pairs will have a value of 1, and with bias of 0.5, all correlated pairs will be either 1 or -1 at uniform random (and thus biased towards neither player). Note that anti-correlated leaf node pairs are unaffected by bias.

## Disambiguation factor

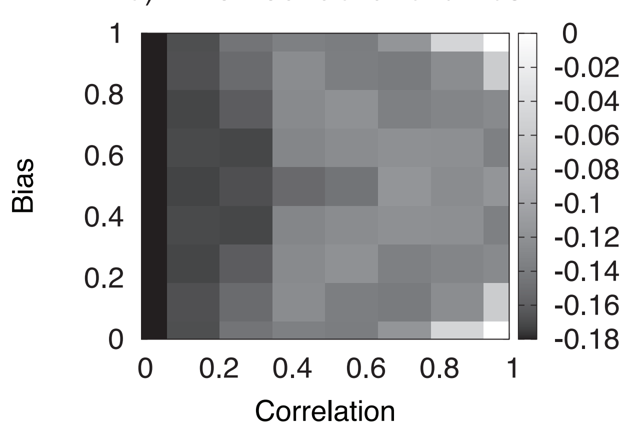
- (df): Each time p is to move, we recursively break each of his information sets in half with probability df (thus, each set is broken in two with probability df; and if a break occurs, each resulting set is also broken with probability df and so on). If df is 0, then p never gains any direct knowledge of his opponent's private information. If df is 1, the game collapses to a perfect information game, because all information sets are broken into sets of size one immediately.

## Measurements in practice

- Trick-based card games
  - Leaf-correlation: tends to be correlated
  - Bias: tend to have bias based on cards
  - Disambiguation: lots of disambiguation (each action provides some information)

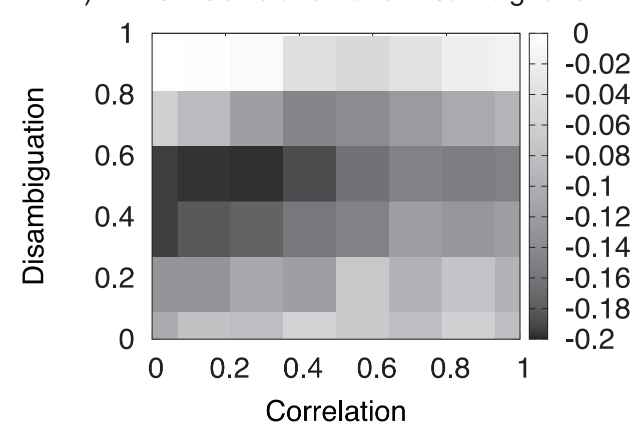
## Abstract model results

a) PIMC - Correlation and Bias

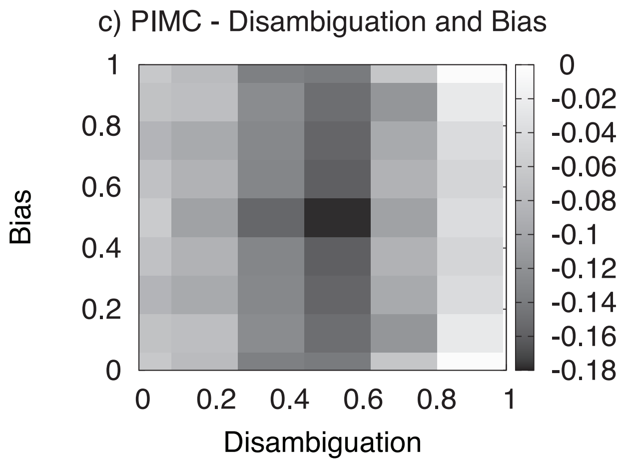



## Abstract model results

b) PIMC - Correlation and Disambiguation



# Abstract model results




# Measurements in practice

- Kuhn Poker
  - Leaf-correlation: mixed (0.5)
    - You can sometimes fold and give the payoff to the other player (anti-correlated)
  - Bias: tend to have bias based on cards, but averages out over all cards (0.5)
  - Disambiguation: no disambiguation (actions give no direct information about the cards you hold)

# Kuhn Poker

| | Opponent | |
|---|---|---|
| Player: | Nash | Best-Response |
| Random (p1) | -0.161 | -0.417 |
| Random (p2) | -0.130 | -0.500 |
| PIMC (p1) | -0.056 | -0.083 |
| PIMC (p2) | 0.056 | -0.166 |

Table 1: Average payoff achieved by random and PIMC against Nash and best-response players in Kuhn poker.