Automatic Heuristic Construction in a Complete General Game Player

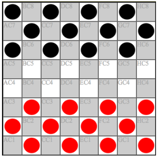
Gregory Kuhlmann Kurt Dresner Peter Stone

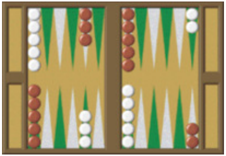
Learning Agents Research Group
Department of Computer Sciences
The University of Texas at Austin

AAAI 2006

Computer Game Playing

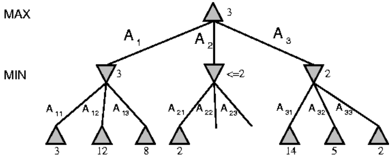
- One of AI’s biggest success stories
  - checkers, chess, scrabble, othello, connect-4





Computer Game Playing

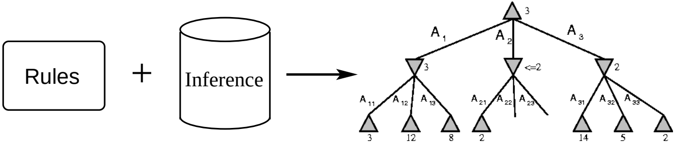
- One of AI’s biggest success stories
  - checkers, chess, scrabble, othello, connect-4
- **Search** is universal in game playing



- Bound search for large state spaces
  - Board evaluation function **(heuristic)**
- Game analysis
  - Traditionally performed by human designers
  - Specific to a single game

General Game Playing

- Single system plays **many games** in a class
- Analysis performed by **system itself**
- Player inputs game rules for unknown game
  - Game description allows simulation
  - Expand game tree

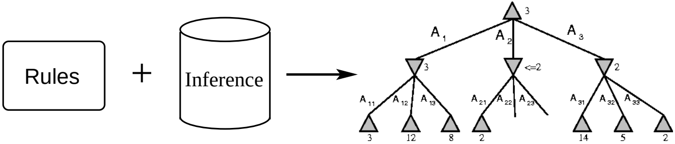


Rules + Inference

- **If not exhaustively searchable, what to do?**
  - Look for hints in game description

General Game Playing

- Single system plays **many games** in a class
- Analysis performed by **system itself**
- Player inputs game rules for unknown game
  - Game description allows simulation
  - Expand game tree

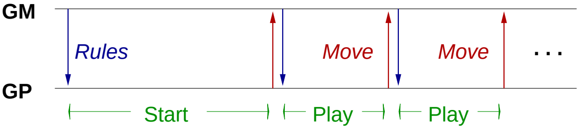


Rules + Inference

- **If not exhaustively searchable, what to do?**
  - Look for hints in game description

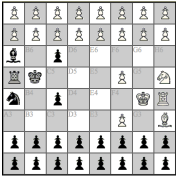
AAAI GGP Competition

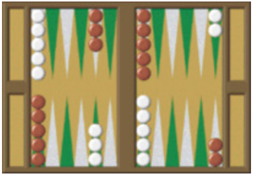
- **Game Players** run as servers
- Game Manager sends rules to players
  - Game Description in **GDL**
  - Start clock
    - Time to analyze description (1–40 minutes)
  - Play clock:
    - Time to make moves (10–120 seconds)

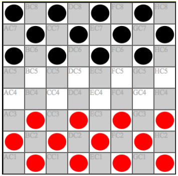


Class of Games

Deterministic, Perfect Information Games







YES

NO

Game Description Language

- First order logic (KIF)
- State: database of provable facts
- Constructs
  - **init:** initial state
  - **legal:** legal moves
  - **next:** state transitions
  - **terminal:** termination conditions
  - **goal:** value of terminal states

Game Description Language

```
(role white)  (role black)
(init (cell a 1 b))   (init (cell a 2 b))
(init (cell a 1 b))   (init (cell a 2 bk))
(init (cell a 1 wr))  (init (cell a 2 b))
(init (cell a 1 b))   (init (cell a 2 b))
(init (control white))  (init (step 1))
(<= (legal white (move wk ?u ?v ?x ?y))
    (true (control white))
    (true (cell ?u ?v wk))
    (kingmove ?u ?v ?x ?y)
    (true (cell ?x ?y b)))
(<= (next (step ?y))
    (true (step ?x))
    (succ ?x ?y))
(succ 1 2)  (succ 2 3)  (succ 3 4)  (succ 4 5)
(<= (goal white 100)
    checkmate)
(<= terminal
    (true (step 10)))
```

- Simulate with theorem prover (Prolog)
- How can we do better than just legal play?

Game Description Language

```
(role white)  (role black)
(init (cell a 1 b))   (init (cell a 2 b))
(init (cell a 1 b))   (init (cell a 2 bk))
(init (cell a 1 wr))  (init (cell a 2 b))
(init (cell a 1 b))   (init (cell a 2 b))
(init (control white))  (init (step 1))
(<= (legal white (move wk ?u ?v ?x ?y))
    (true (control white))
    (true (cell ?u ?v wk))
    (kingmove ?u ?v ?x ?y)
    (true (cell ?x ?y b)))
(<= (next (step ?y))
    (true (step ?x))
    (succ ?x ?y))
(succ 1 2)  (succ 2 3)  (succ 3 4)  (succ 4 5)
(<= (goal white 100)
    checkmate)
(<= terminal
    (true (step 10)))
```

- Simulate with theorem prover (Prolog)
- How can we do better than just legal play?

Identifying Structures

Identify structures from common game elements

Successor Relations

```
(succ 1 2)      (angel paper table)
(succ 2 3)      (angel table bottom)
(succ 3 4)      (angel bottom mellow)
(succ 4 5)      (angel mellow yard)
```

Tokens will be scrambled. Based on structure alone.

Bridge between logical and **numerical** representations

Identifying Structures (cont.)

Find rules matching templates

Step Counters

```
(<= (next (step ?x))    (<= (next (foo ?u))
    (true (step ?y))        (true (foo ?v))
    (succ ?y ?x))           (bar ?v ?u))
```

Again no lexical clues used.

- Bounds tree depth
- Remove for longer internal games
- Remove from **Transposition Table**

Board Game Structures

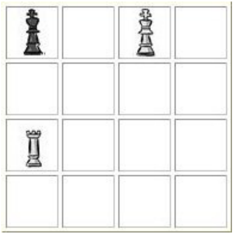
Many games have a board of some type

State

```
(cell 1 1 bk) (cell 1 2 b)
(cell 1 3 wk) (cell 1 4 b)
(cell 2 1 b)  (cell 2 2 b)
(cell 2 3 bk) (cell 2 4 b)
(cell 3 1 wr) (cell 3 2 b)
(cell 3 3 b)  (cell 3 4 b)
(cell 4 1 b)  (cell 4 2 b)
(cell 4 3 b)  (cell 4 4 b)
```

Boards and Pieces

```
cell:0,1->2 ; [b, wk, wr, bk]
cell:0,2->1 ; [1, 2, 3, 4]
cell:1,2->0 ; [1, 2, 3, 4]
```



- Start with all ternary functions
- Divide slots into **inputs** and outputs
- **Refine** through internal simulation

Board Game Structures

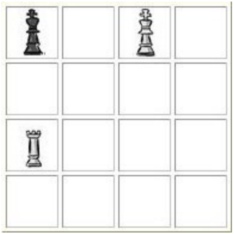
Many games have a board of some type

State

```
(cell 1 1 bk) (cell 1 2 b)
(cell 1 3 wk) (cell 1 4 b)
(cell 2 1 b)  (cell 2 2 b)
(cell 2 3 bk) (cell 2 4 b)
(cell 3 1 wr) (cell 3 2 b)
(cell 3 3 b)  (cell 3 4 b)
(cell 4 1 b)  (cell 4 2 b)
(cell 4 3 b)  (cell 4 4 b)
```

Boards and Pieces

```
cell:0,1->2 ; [b, wk, wr, bk]
```



- Start with all ternary functions
- Divide slots into **inputs** and outputs
- **Refine** through internal simulation

Board Game Structures

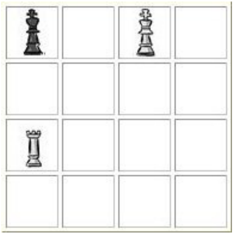
Many games have a board of some type

State

```
(cell 1 1 bk) (cell 1 2 b)
(cell 1 3 wk) (cell 1 4 b)
(cell 2 1 b)  (cell 2 2 b)
(cell 2 3 bk) (cell 2 4 b)
(cell 3 1 wr) (cell 3 2 b)
(cell 3 3 b)  (cell 3 4 b)
(cell 4 1 b)  (cell 4 2 b)
(cell 4 3 b)  (cell 4 4 b)
```

Boards and Pieces

```
cell:0,1->2 ; [wk, wr, bk]
```



- Start with all ternary functions
- Divide slots into **inputs** and outputs
- **Refine** through internal simulation

Features

Identified Structure	Generated Features
Ordered Board w/ Pieces	Each piece's X coordinate Each piece's Y coordinate Manhattan distance between each pair of pieces Sum of pair-wise Manhattan distances
Board w/o Pieces	Number of markers of each type
Quantity	Amount

- Board inputs ordered by successor relation(s)?
- Board has at least one piece?
- Non-board features also identified

From Features to Heuristics

Maximize single feature:

$$H(s) = 1 + R^{-} + (R^{+} - R^{-} - 2) * V(s)$$

Or **minimize** single feature:

$$H(s) = 1 + R^{-} + (R^{+} - R^{-} - 2) * [1 - V(s)]$$

- Example: Maximize white rook's y-coordinate
- Actual win always better than heuristic value
- Actual loss always worse

Complete System

During Start Clock:

- Candidate heuristics constructed from GD
- "Best" heuristic is chosen
  - **Old approach:** parallel search
  - **New approach:** internal tournament

During Play Clock:

- Iterative-deepening Minimax search
  - Minimax search w/ $\alpha\beta$ pruning
  - Transposition table and history heuristic
  - Extensions for $>$ 2 players, simultaneous games

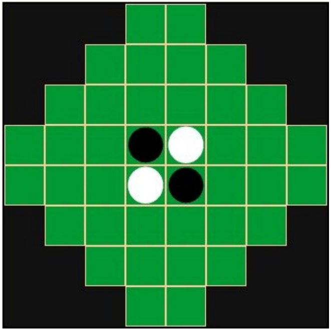
Experiments

- **Goal:** Identify impact of game analysis
- Three different games
  - created by competition organizers
- Heuristic chosen manually
  - simulates good method to choose heuristic
  - no experimentation after initial selection
- Opponent: constant heuristic (exhaustive search)

Nothello

Othello variant

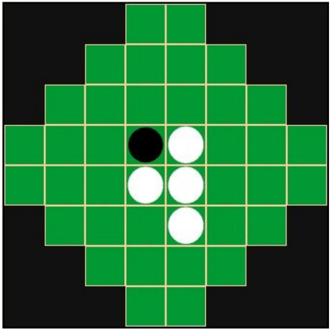
- more corner squares
- **opposite goal:** finish with fewer markers



Nothello

Othello variant

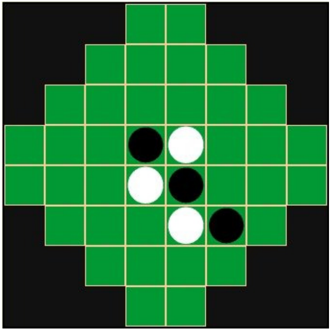
- more corner squares
- **opposite goal:** finish with fewer markers



Nothello

Othello variant

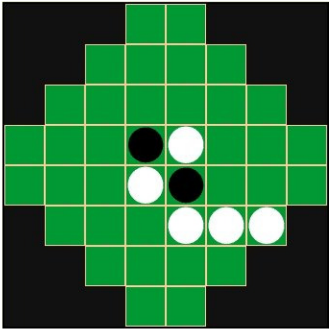
- more corner squares
- **opposite goal:** finish with fewer markers



Nothello

Othello variant

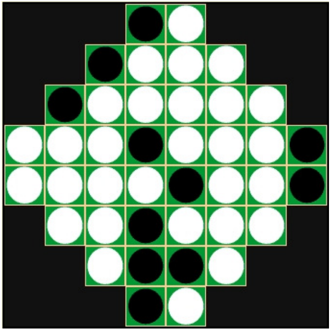
- more corner squares
- **opposite goal:** finish with fewer markers



Nothello

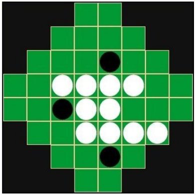
Othello variant

- more corner squares
- **opposite goal:** finish with fewer markers



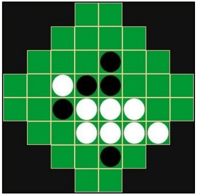
Nothello

Heuristic: minimize number of own markers



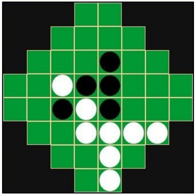
NumMarkers: 10

H(s): 74.25



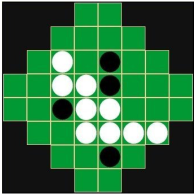
NumMarkers: 8

H(s): 79.20



NumMarkers: 8

H(s): 79.20

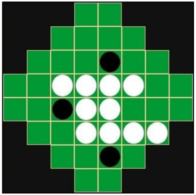


NumMarkers: 9

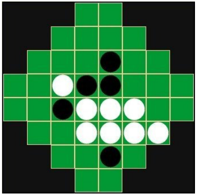
H(s): 76.73

Nothello

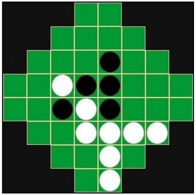
Heuristic: minimize number of own markers


NumMarkers: 10

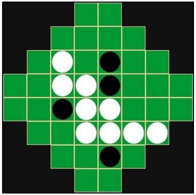
H(s): 74.25


NumMarkers: 8

H(s): 79.20


NumMarkers: 8

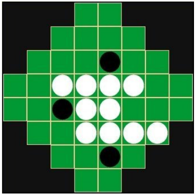
H(s): 79.20


NumMarkers: 9

H(s): 76.73

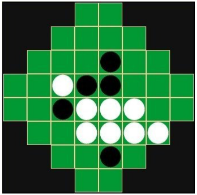
Nothello

Heuristic: minimize number of own markers



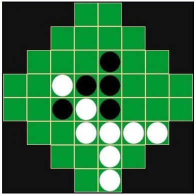
NumMarkers: 10

H(s): 74.25



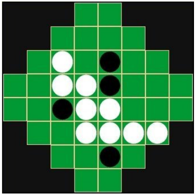
NumMarkers: 8

H(s): 79.20



NumMarkers: 8

H(s): 79.20

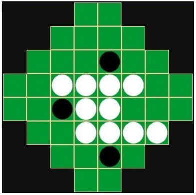


NumMarkers: 9

H(s): 76.73

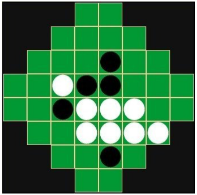
Nothello

Heuristic: minimize number of own markers



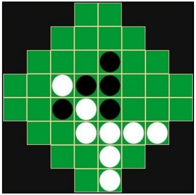
NumMarkers: 10

H(s): 74.25



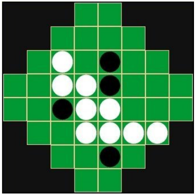
NumMarkers: 8

H(s): 79.20



NumMarkers: 8

H(s): 79.20



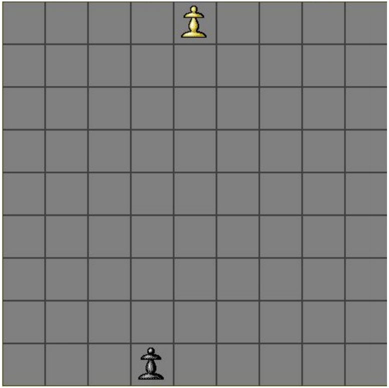
NumMarkers: 9

H(s): 76.73

Hallway

Chess board with two pawns

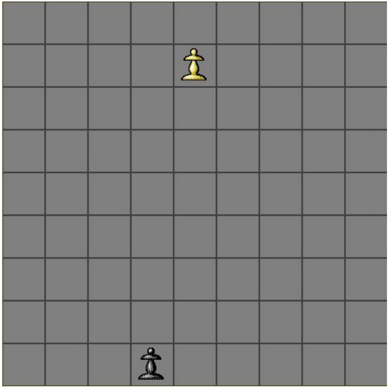
- **Actions:** move pawn or place wall
- **Goal:** reach other side first



Hallway

Chess board with two pawns

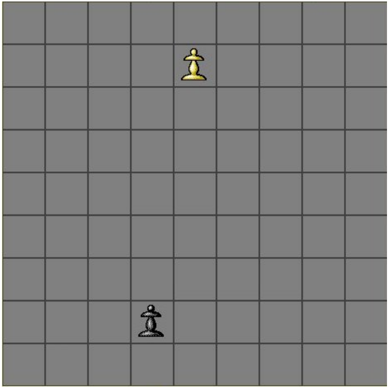
- **Actions:** move pawn or place wall
- **Goal:** reach other side first



Hallway

Chess board with two pawns

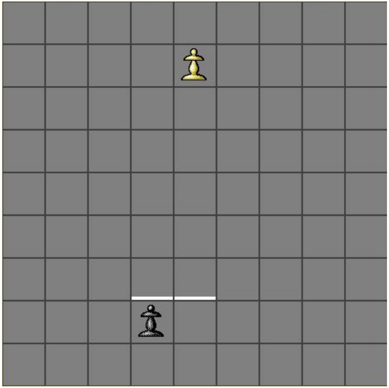
- **Actions:** move pawn or place wall
- **Goal:** reach other side first



Hallway

Chess board with two pawns

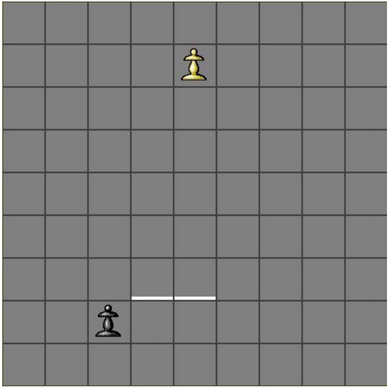
- **Actions:** move pawn or place wall
- **Goal:** reach other side first



Hallway

Chess board with two pawns

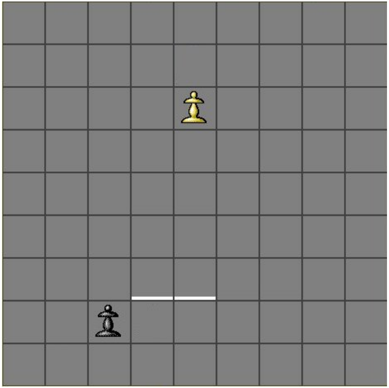
- **Actions:** move pawn or place wall
- **Goal:** reach other side first



Hallway

Chess board with two pawns

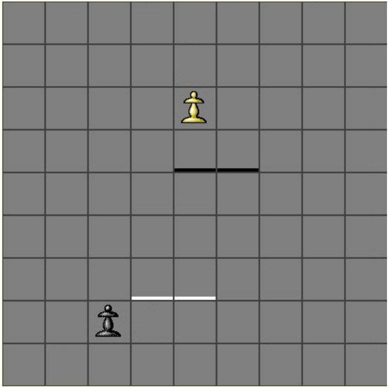
- **Actions:** move pawn or place wall
- **Goal:** reach other side first



Hallway

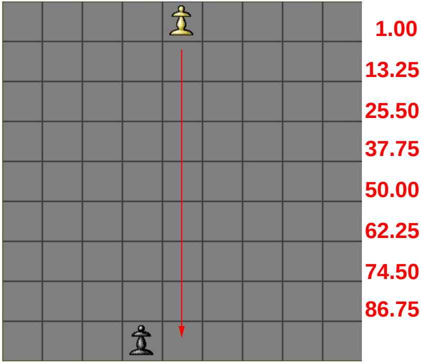
Chess board with two pawns

- **Actions:** move pawn or place wall
- **Goal:** reach other side first



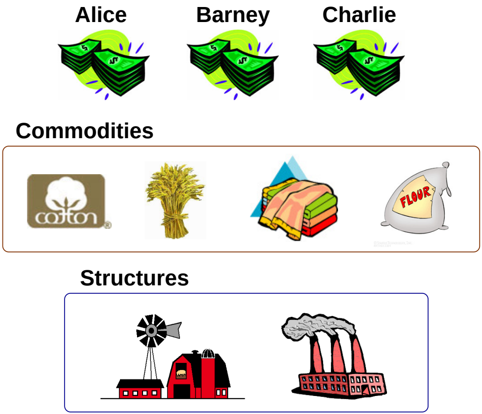
Hallway

Heuristic: maximize own pawn’s y-coordinate



Farmers

Commodities trading game w/ three simultaneous players



Alice

Barney

Charlie

Commodities

Structures

Heuristic: maximize own money

Farmers

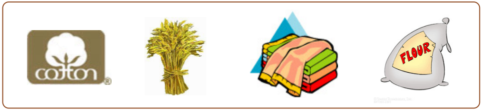
Commodities trading game w/ three simultaneous players

Alice

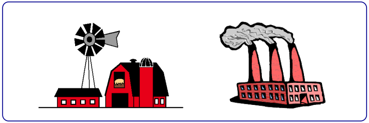
Barney

Charlie

Commodities



Structures



Heistic: maximize own money

Results

- **Experimental results**

Game	Matches	Expected Wins	Empirical Wins	p
Nothello	15	7.5	15	10^{-5}
Hallway	15	3	15	10^{-11}
Farmers	25	8.3	11	0.234

- **Competition Results**
  - 2005: competitive but technical difficulties
  - 2006: very competitive (3rd place)
    - after 72 matches, gap with first: $\sim$ 3 games

UTCS

Results

- **Experimental results**

Game	Matches	Expected Wins	Empirical Wins	p
Nothello	15	7.5	15	10^{-5}
Hallway	15	3	15	10^{-11}
Farmers	25	8.3	11	0.234

- **Competition Results**
  - 2005: competitive but technical difficulties
  - 2006: very competitive (3rd place)
    - after 72 matches, gap with first: $\sim$ 3 games

Conclusion and Future Work

- **General Game Playing**
  - Automate game analysis
- **Automatic Heuristic Construction**
  - Structures → Features → Heuristics
- Method incorporated into **complete agent**
- **Future Work**
  - Learn more complex evaluation functions
  - Understand game similarity
  - Transfer knowledge between games

Conclusion and Future Work

- **General Game Playing**
  - Automate game analysis
- **Automatic Heuristic Construction**
  - Structures → Features → Heuristics
- Method incorporated into **complete agent**
- **Future Work**
  - Learn more complex evaluation functions
  - Understand game similarity
  - Transfer knowledge between games