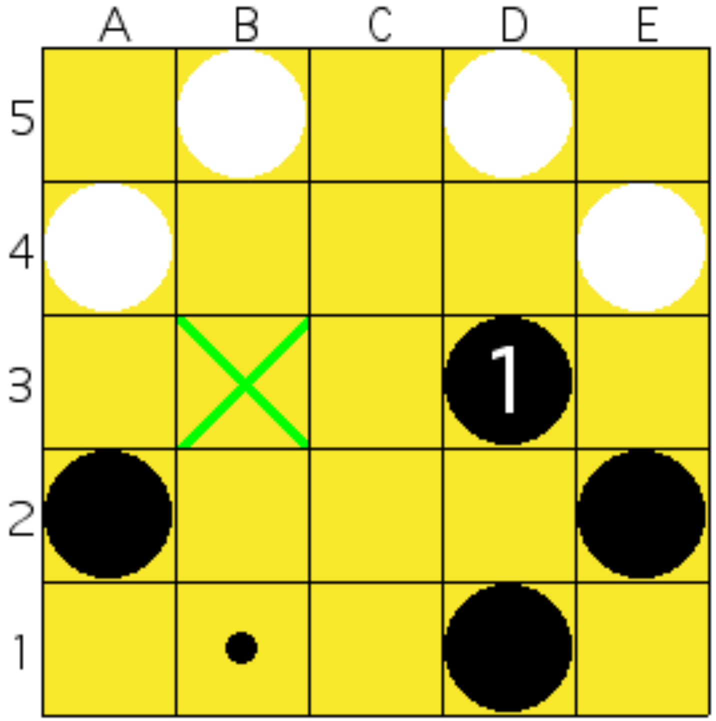
# Solving 5x5 Amazons

Martin Müller
University of Alberta
Edmonton, Canada

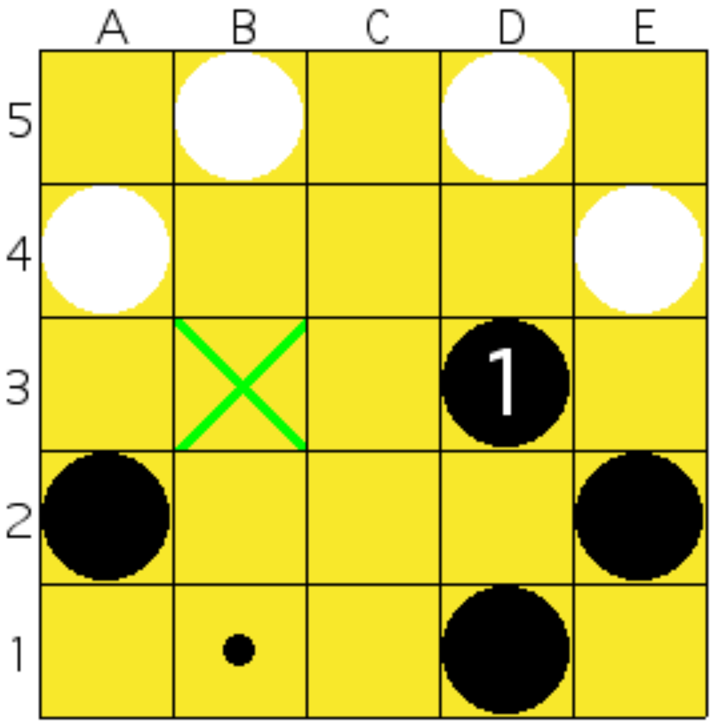


# 5x5 Amazons

- The game of Amazons
- Board partition: active areas and territories
- Upper and lower bounds for an area
- Pruning dominated moves in the search
- Solving 5x5
- How about 6x6?

# The Game of Amazons

- Amazons move and shoot like chess queens
- Cannot move or shoot across *burned-off* squares (X)
- Last player who can make a move wins

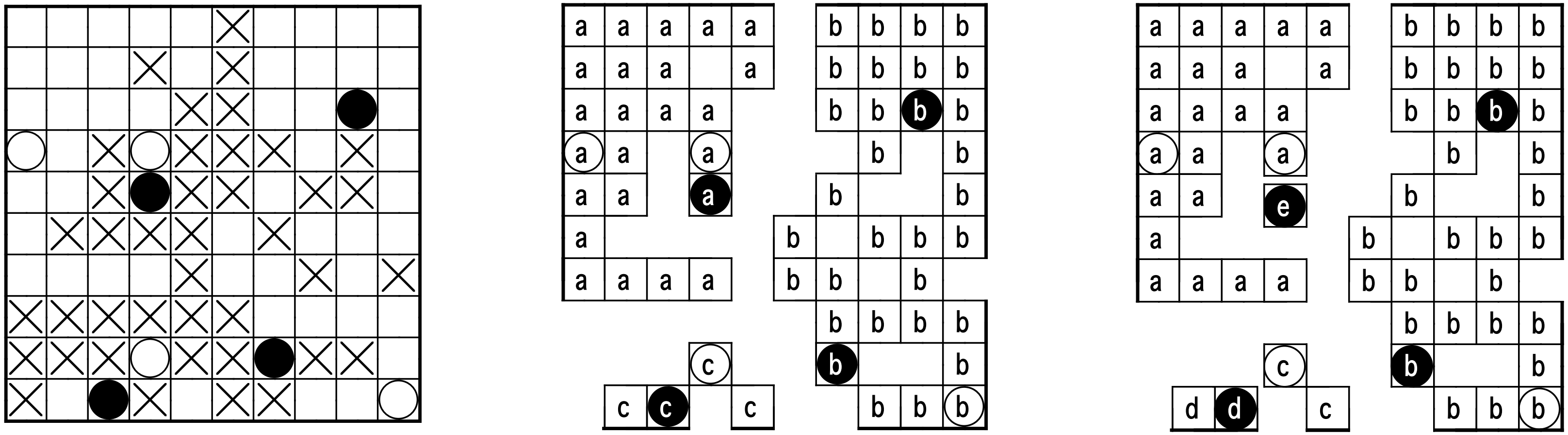


# Why work on 5x5?

- 4x4 too easy, 2nd player win
  - Proof needs only about 6000 nodes
- 6x6 too hard?
- 5x5 search space:
  - 25 points, 8 occupied - at most 17 ply deep
  - Branching factor 262 at root,
    about 80 after 5 moves
  - Hard for a basic brute force search, but…
    …solvable with a bit of work

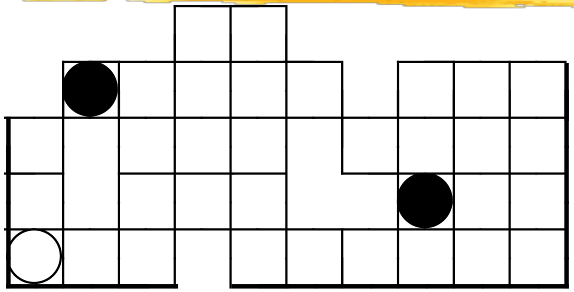
# Board Partition

- Find sets of points connected horizontally or diagonally *(8-connected)*
- Identify *connected components*
- Improve partitioning by *blocking* queens

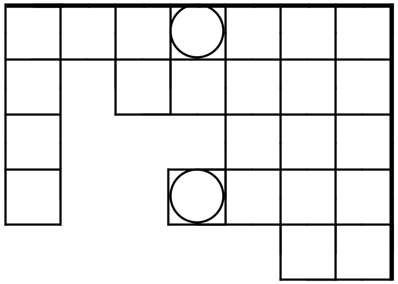


# Territories and Active Areas

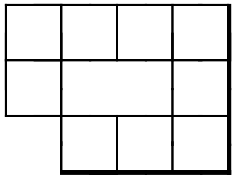
- *Active* area:
  queens of both players



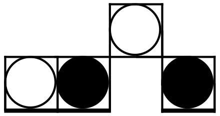
- *Territory*:
  only queens of one player



- *Neutral*:
  no queens at all



- *Dead*:
  only queens, no empty spaces



# Search

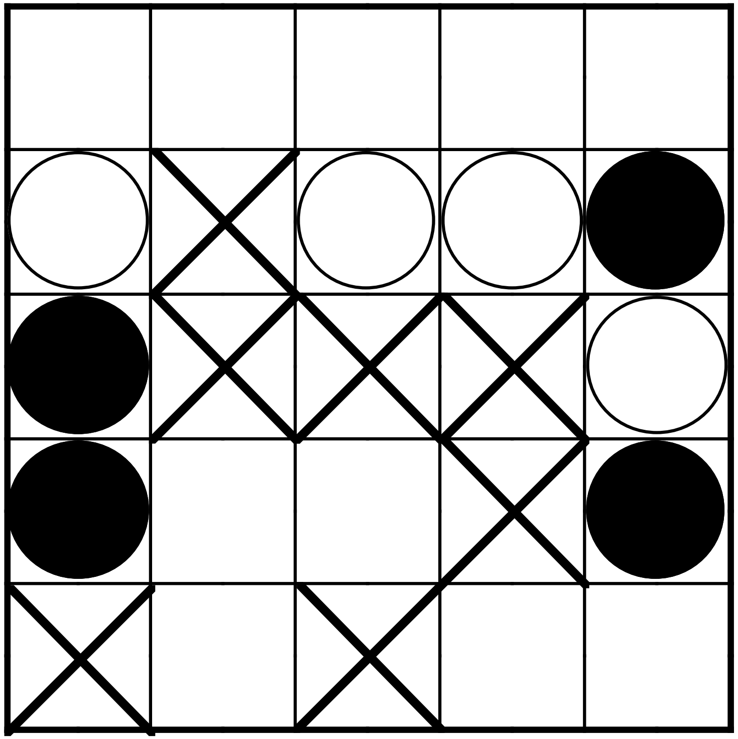
- Brute force search tree too big
- Reduce depth: recognize wins and losses early
- Reduce width: prune dominated moves
- Added rules to my program *Arrow 0.09*
- Disabled *Arrow's* inexact pruning heuristics

# Recognizing Wins and Losses Early

- Basic idea:
  player with more moves wins
  - If both have same number of moves,
    the second player wins
- Method:
  - compute *upper and lower bounds*
    on number of moves in each area
  - Add up estimates, compare with 0
  - Similar to Nathan's work in *Domineering*

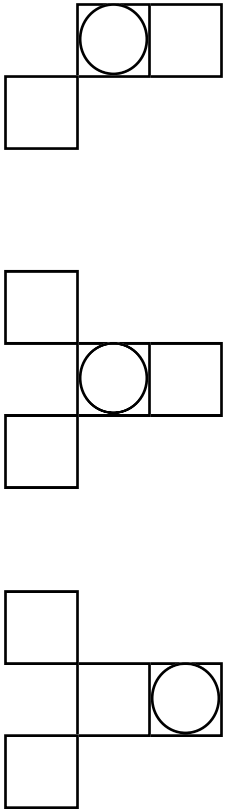
# Example:
# A Winning Position for Black

- W to play here
- W is already lost!
- Proof:
  - B territory is worth 5 moves
  - The active area at the top is worth at most 5 moves
  - W must play first



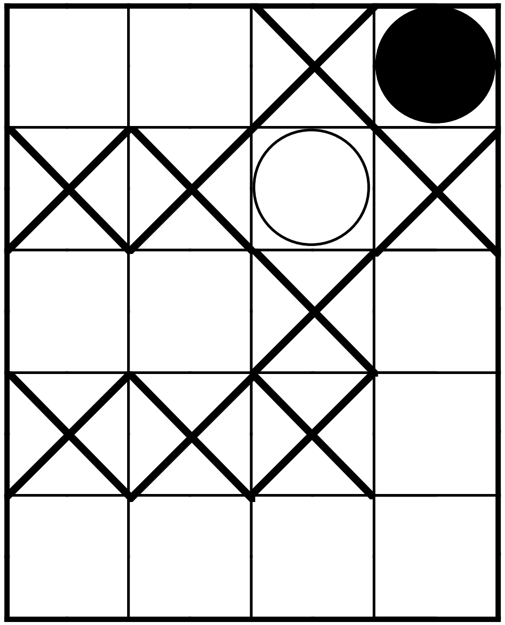
# Finding the Size of Territories

- How many moves can player make in a territory?
- *Defective* areas: cannot fill all empty points
- Simple lower bound: *plod*
  - now used in solver
- Better: 1 ply search with heuristic
  - now used in heuristic player
- Best: use Theo's databases - future



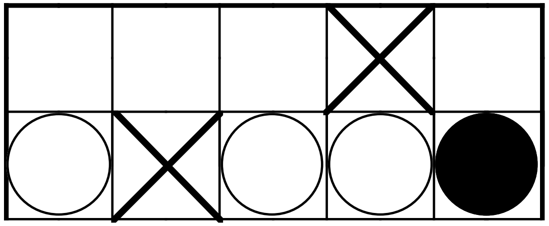
# Handling Multi-territory Blockers

- Problem:
  blocker that blocks off two or more territories
- Cannot fill all territories
- Safe:
  assume she can only fill the biggest one

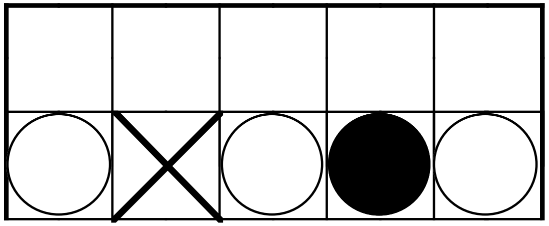


# Improved Bounds for Active Areas

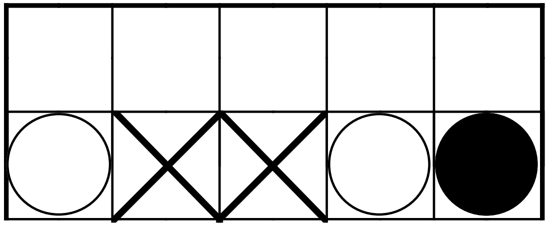
- n empty: *-n* <= *value* <= +*n*
- Improve: guaranteed moves
  - Single move: *if we play first*, improve bound by 2
  - Pair of *independent* moves: improve bound by 2
- Combine several areas
  - Two single moves: improve bound by 2



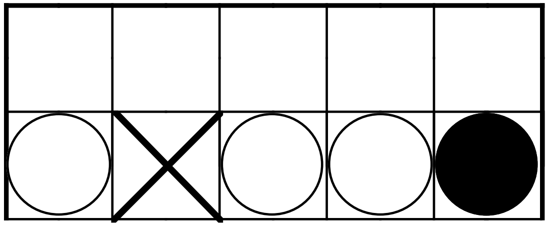
2



2



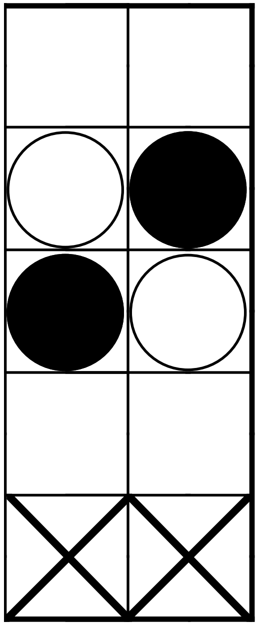
2(!)



0(!)

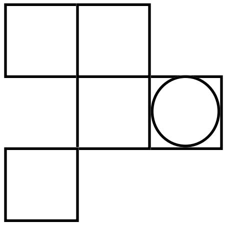
# Example: Improving the Bounds can Lead to an Exact Value

- Starting bounds: [-4,4]
- Each queen has a safe move
- Both players can improve their bounds by 2*2
- Final bounds: [0,0]
- Exact value: 0

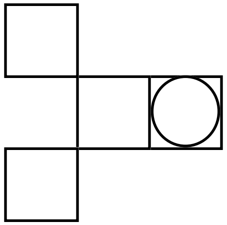


# Pruning Moves in the Search

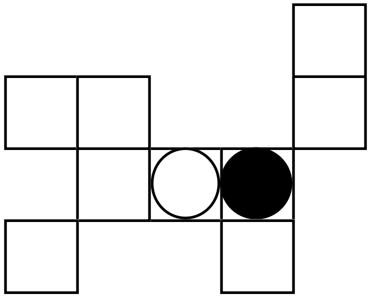
- Exact solver: may prune only irrelevant or equivalent moves



- Main reduction: filling territories

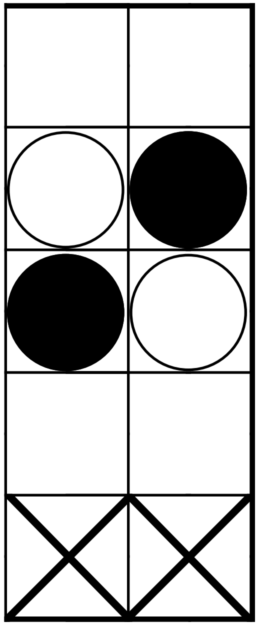


- If can fill completely, generate only one optimal move
- Problem: blockers

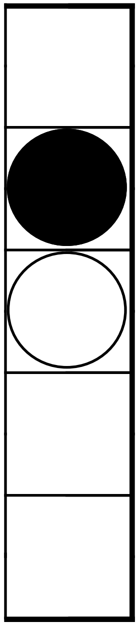


# Pruning Moves (2)

- Rule:
  If value of area is constant and *value* <= *0* for us, then do not generate any moves there.



0



-1 for Black

# Move Ordering

- Embedded prover in my normal Amazons program *Arrow*
- Used *Arrow's* heuristic evaluation function
- Move sorting for new node: 1 ply lookahead, call static evaluation function
- Move sorting for old node: get old search value from iterative deepening

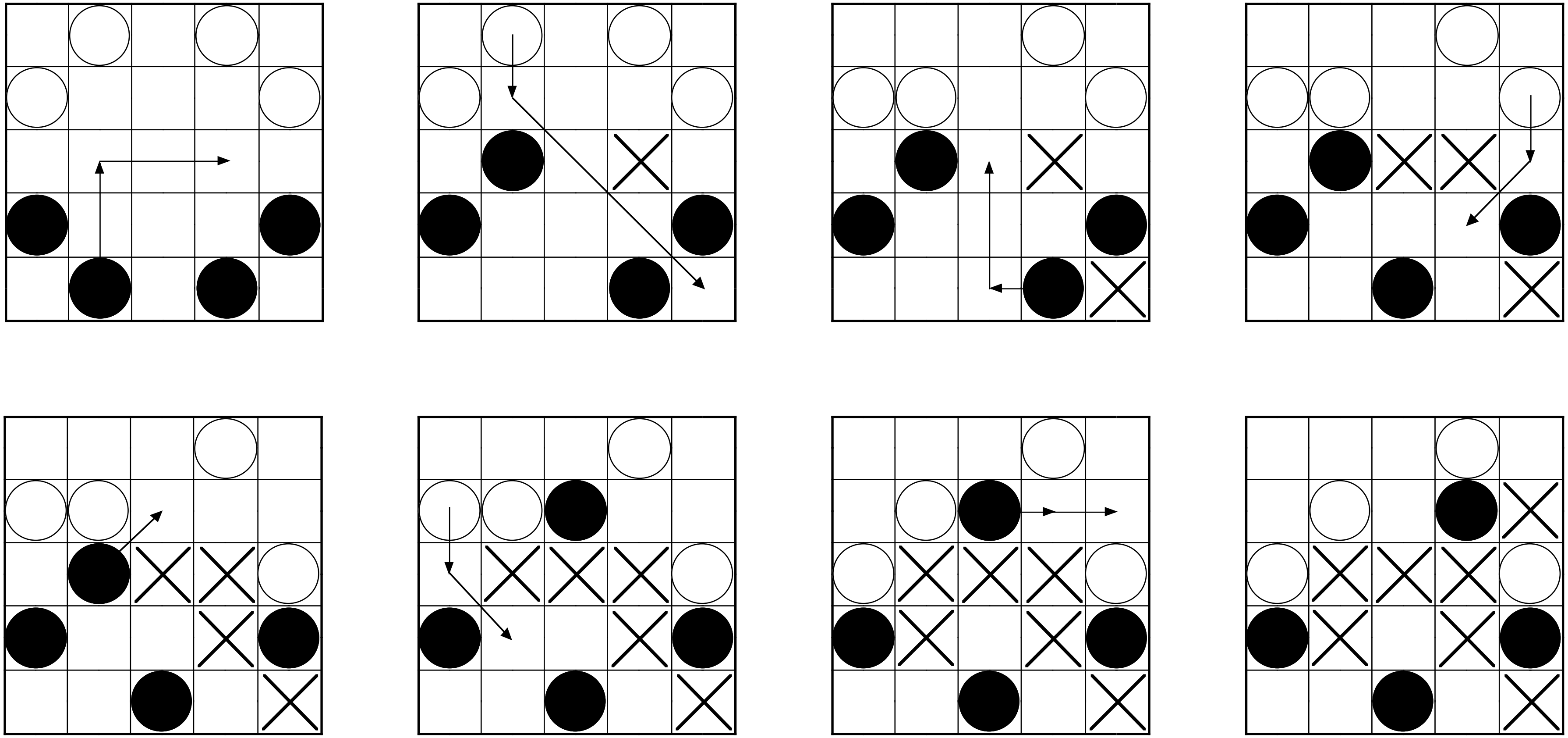
# Useful Observations

- Looking at positions near the end of the PV often suggested improvements to the bounds calculation: if position looks easy for human, but is not recognized by the program, formulate a new rule to cover this case.
- Number of nodes needed in proof is an excellent measure of the quality of the heuristic evaluation function

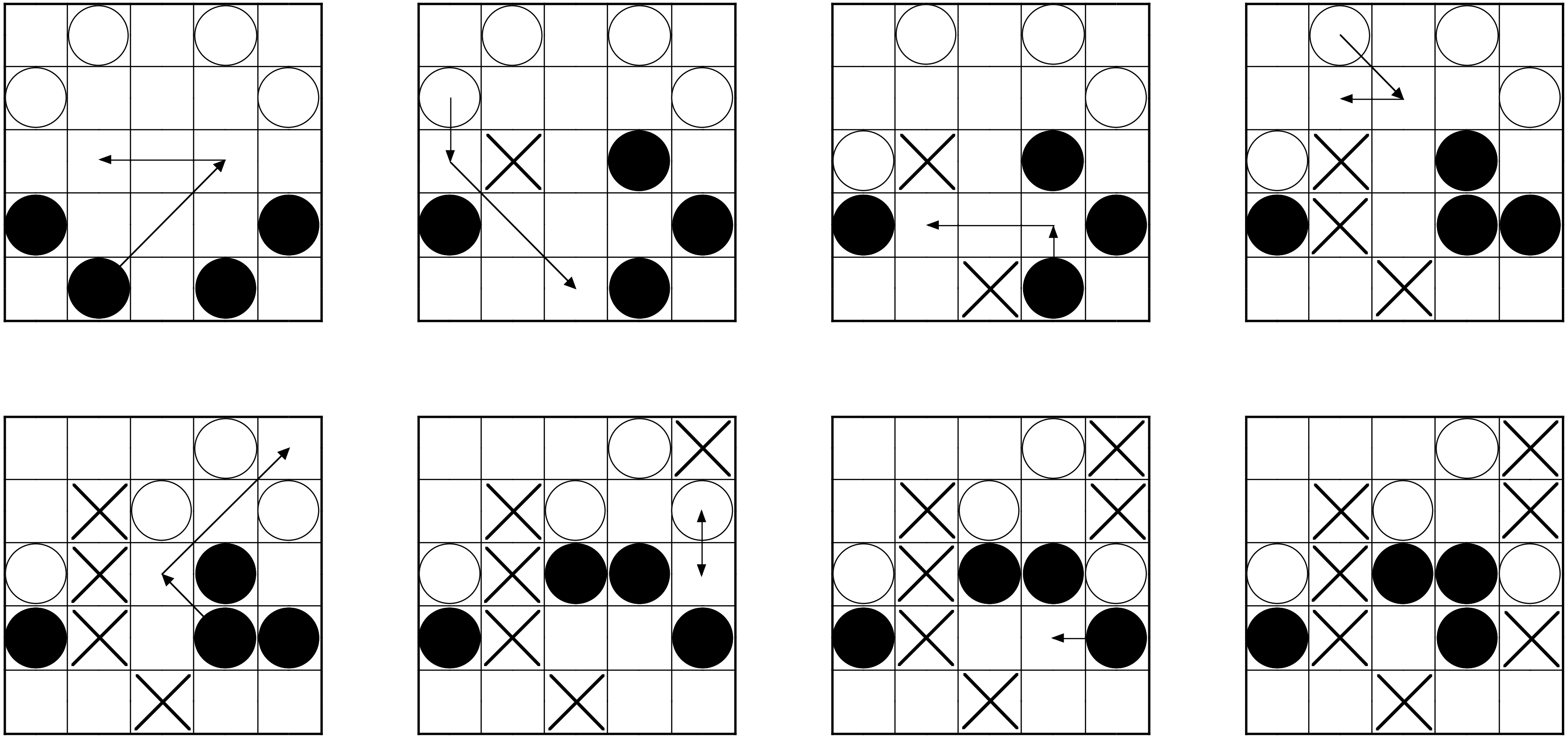
# Effects

- Success: average depth reduction 6-7 ply
- Many bad moves disproven *very* quickly, even on 6x6 board
- Proven first-player win on 5x5 by a 7 ply search from the root
- 21.7M moves, 1.7M node expansions
- Lucky: simple greedy strategy enough to win
- Some other 5x5 positions are much harder, need 10 ply (or maybe more)

# Sample Sequence (1)
# PV for 1.B1-B3xD3

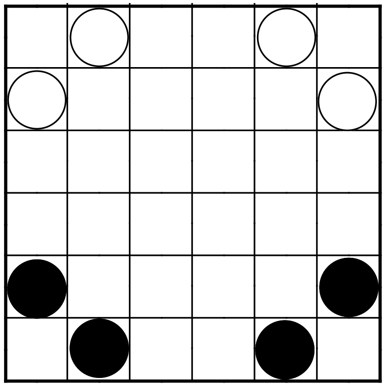


# Sample Sequence (2)
# PV for 1.B1-D3xB3

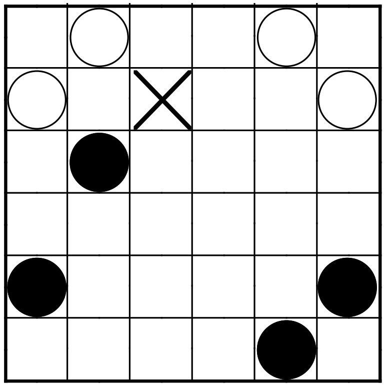


# Ongoing and Future Work

- *Opening book* for 5x5 with Tom Lincke
  - Classify all openings as wins or losses

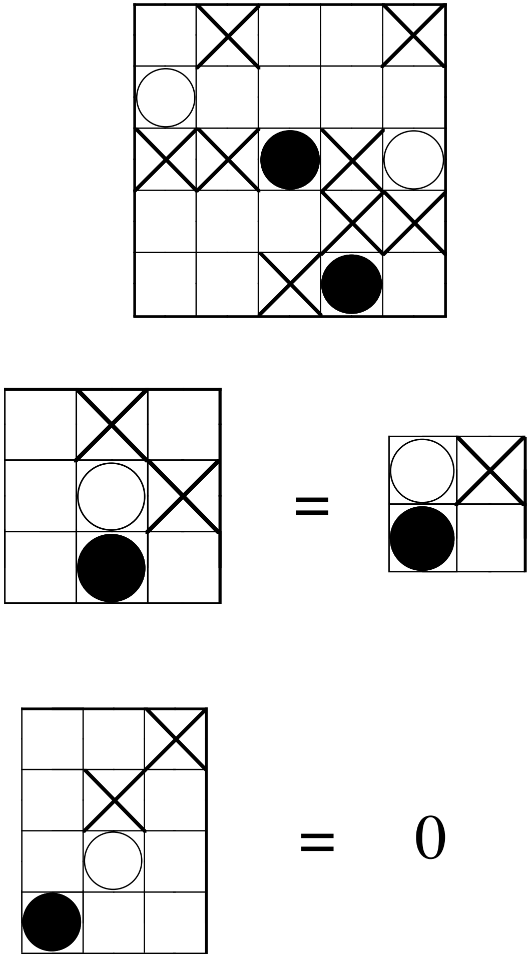


- Interface to *databases* for defective territories and combinatorial games
- Tom also started solving *6x6*:
  - After 1. B1-B4xC5,
    about 5% of replies are proven losses…
  - …but these are the *really* bad moves!
  - My guess: >100.000 times harder



# Future Work (1) - Improve Bounds

- Use blocker in the *outside* area
- Use pairs of local minimax searches
- Use combinatorial game databases



# Future Work (2) - Prune Moves

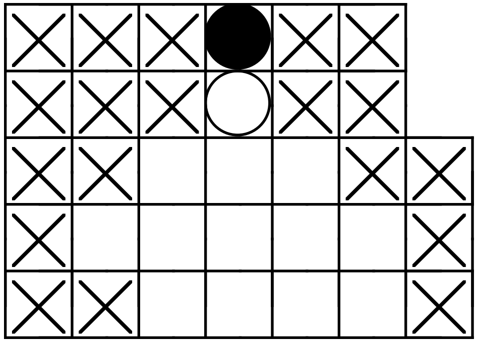
- Use combinatorial game databases to find dominated moves
- Play equivalent *abstract* combinatorial game instead of game on Amazons board
- Better rules for identifying locally *forced* moves

# Future Work (3) - Heuristics or Exact Rules?

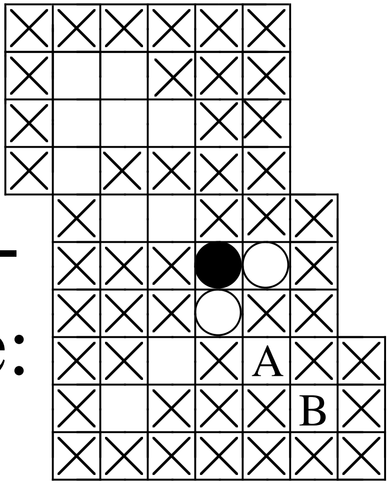
- Plausible rules
- Almost always true
- Used in heuristic program, but not in prover
- Challenge:
  can you prove that they are correct in special cases?

# Heuristic 1: A Blocker Should Block

- Heuristic rule:
  if a blocker moves towards the inside, it must shoot back to its starting square (or to the outside). It should not shoot to the inside.
- Not always true.
- When is it true?

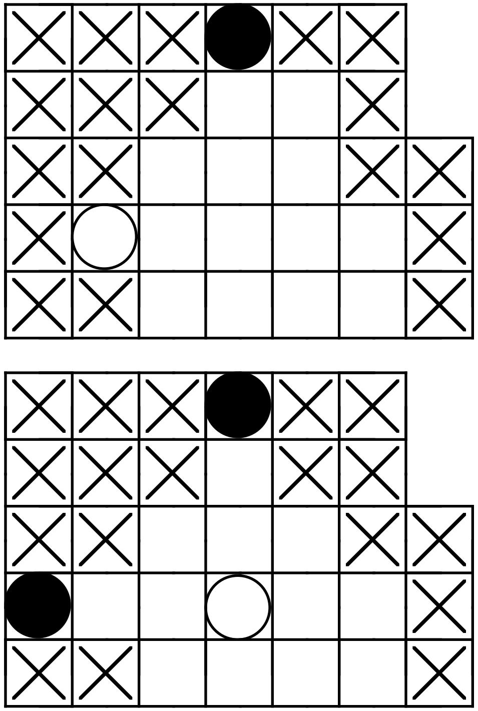


Counter-example:



# Heuristic 2: Take it all - if You Can

- Heuristic rule:
  if you can block off an area from the opponent completely, then that's the best move locally.
- Potential problems:
  - If opponent moves away, we might get more if we don't block
  - Block might make area defective



# Final Words…

- 5x5 Amazons proven a first player win
- Main idea: reduce search depth and width by computing bounds on value of subgames
- Lucky - on 5x5 a simple conservative strategy is good enough. This simplifies proof a lot.
- 6x6 is the real challenge!