Revisiting Coexistence by Seki (雙活) in the Game of Go (圍棋)

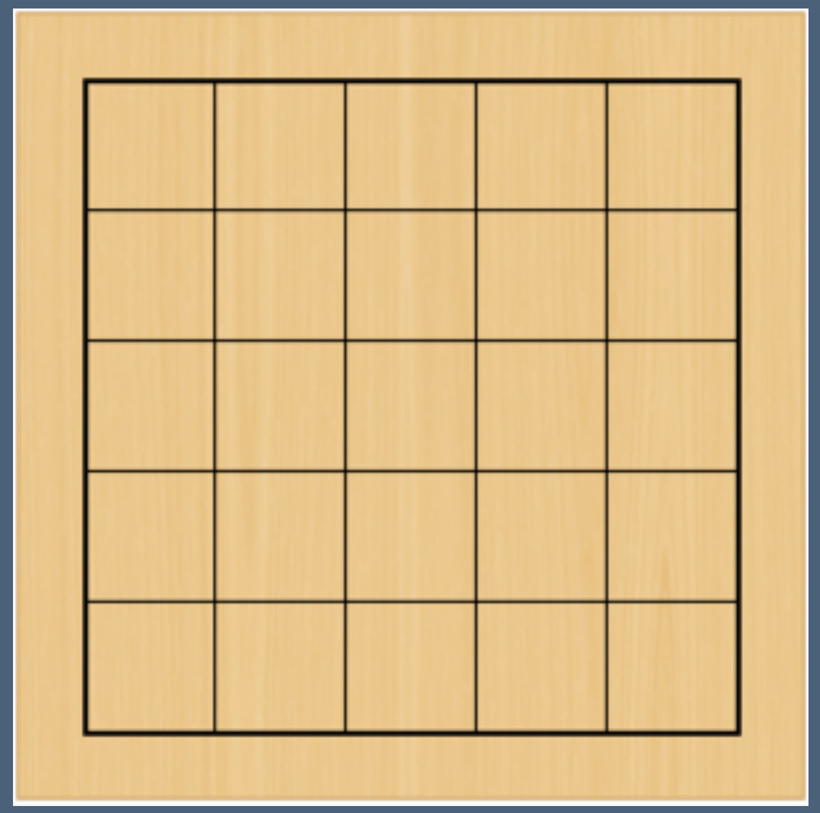
Towards an efficient approach for solving small Go boards

Martin Müller, University of Alberta, May/June 2024

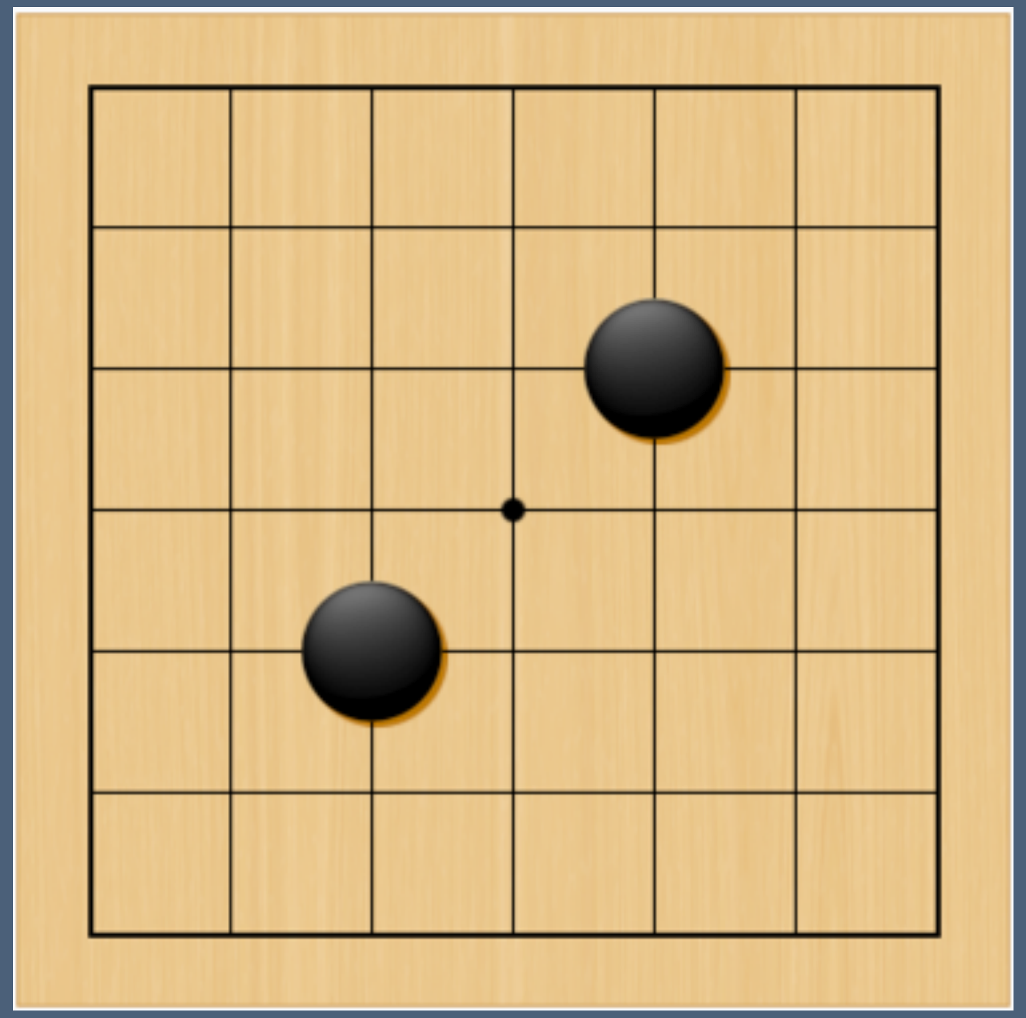
Chinese Go terms from https://en.wikipedia.org/wiki/List_of_Go_terms

Introduction - Timeline

- Work on exact knowledge for programming the game of Go
- Popular research topic up to about 2006
- Then came Monte Carlo Tree Search (MCTS)
- Then came AlphaGo, Alpha Zero etc.
- Other work stopped for many years
- Now we try to solve small board Go
- Need to re-visit exact methods



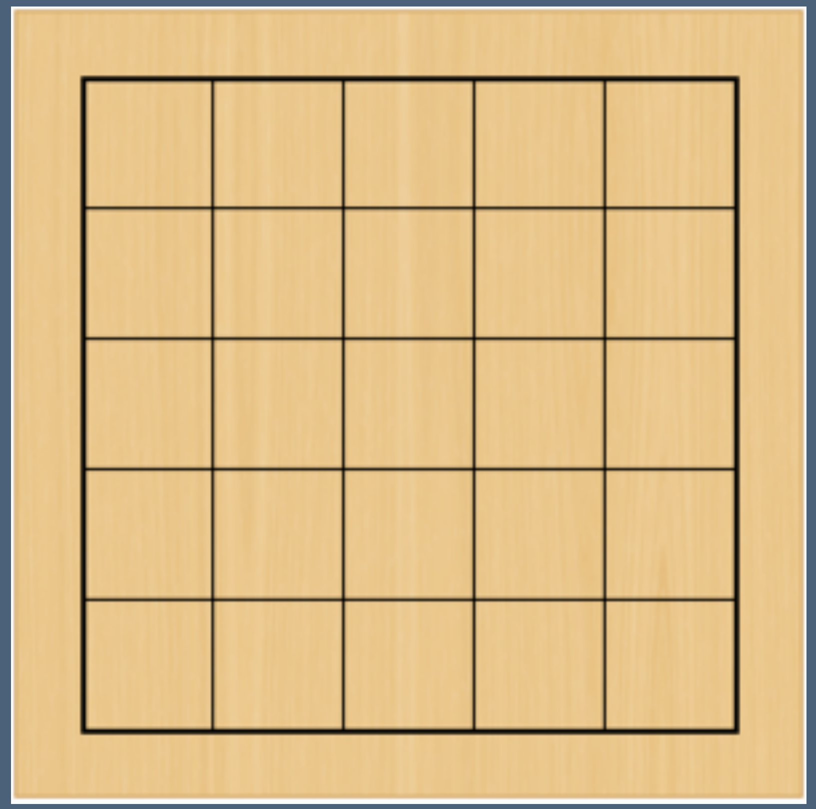
6x6 Go



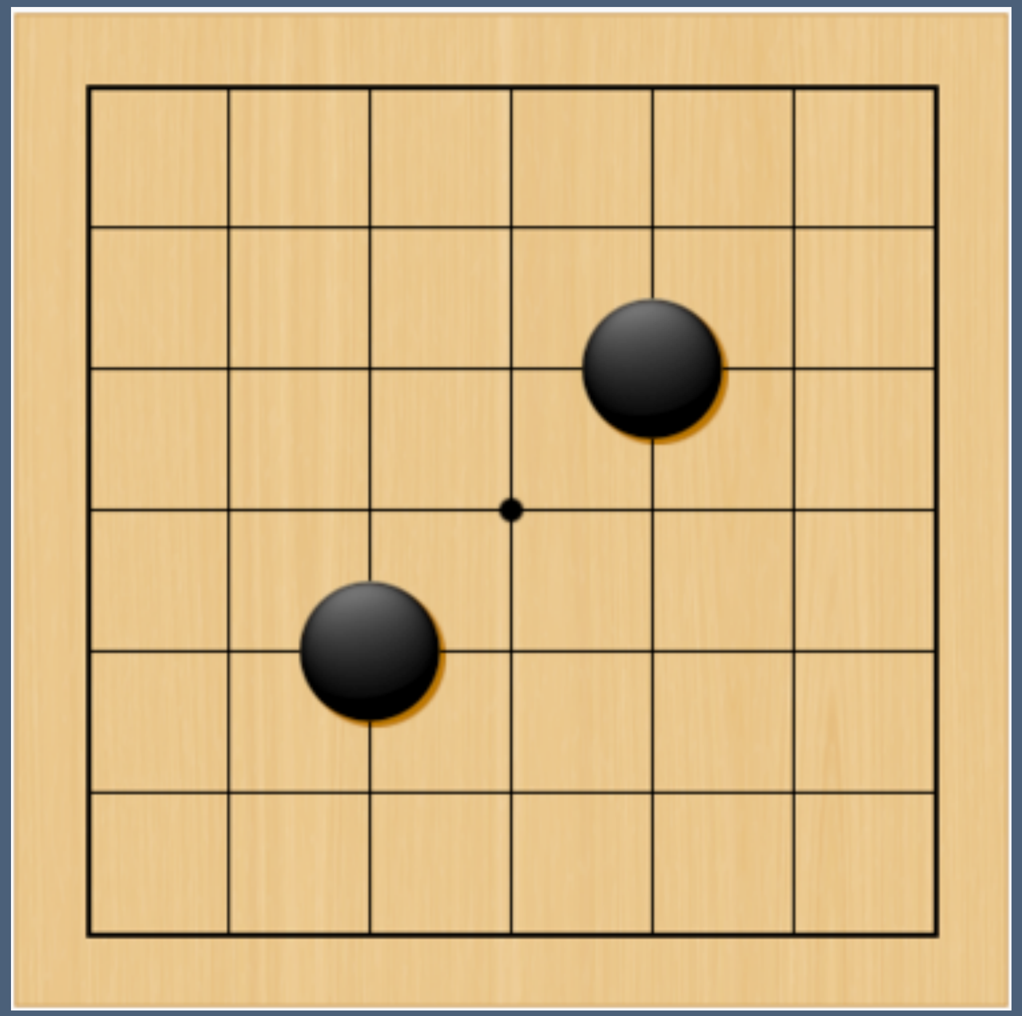
7x7 Killall Go example

Motivation for this Talk

- Goal: solve Go on small boards
  - 6x6 board
    - University of Alberta project
  - 7x7 killall Go
    - White passes at first move
    - Black must kill all white styones
    - Academia Sinica project
- Two very large search problems
- How to solve efficiently?



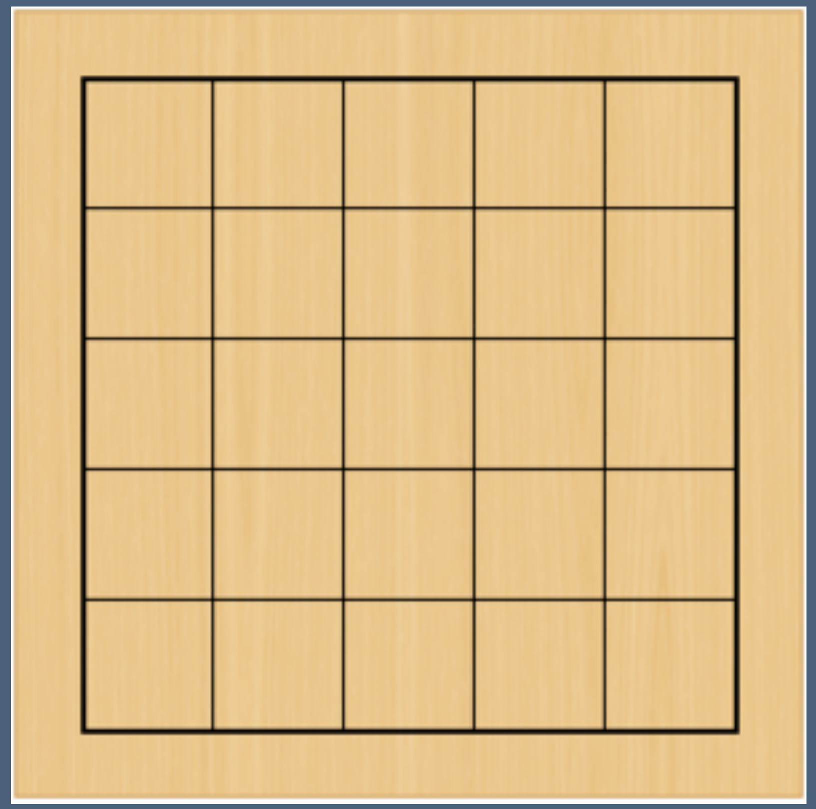
6x6 Go

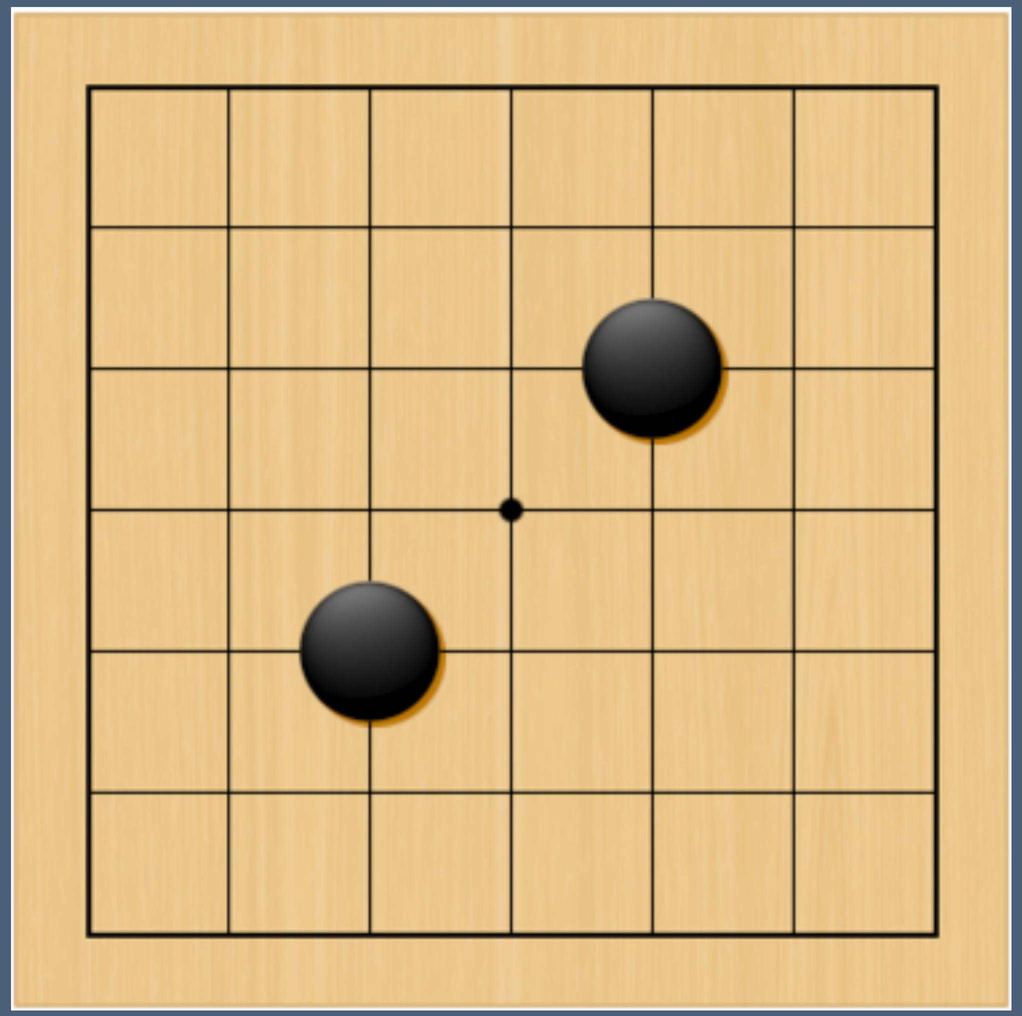


7x7 Killall Go example

Motivation for this Talk (2)

- Goal: solve Go on small boards efficiently
- Two main approaches
- Reduce branching factor - move ordering
- Reduce search depth: solve positions earlier, without search
    - Approach so far: static knowledge for safe points
    - Problem: does not work in seki situations

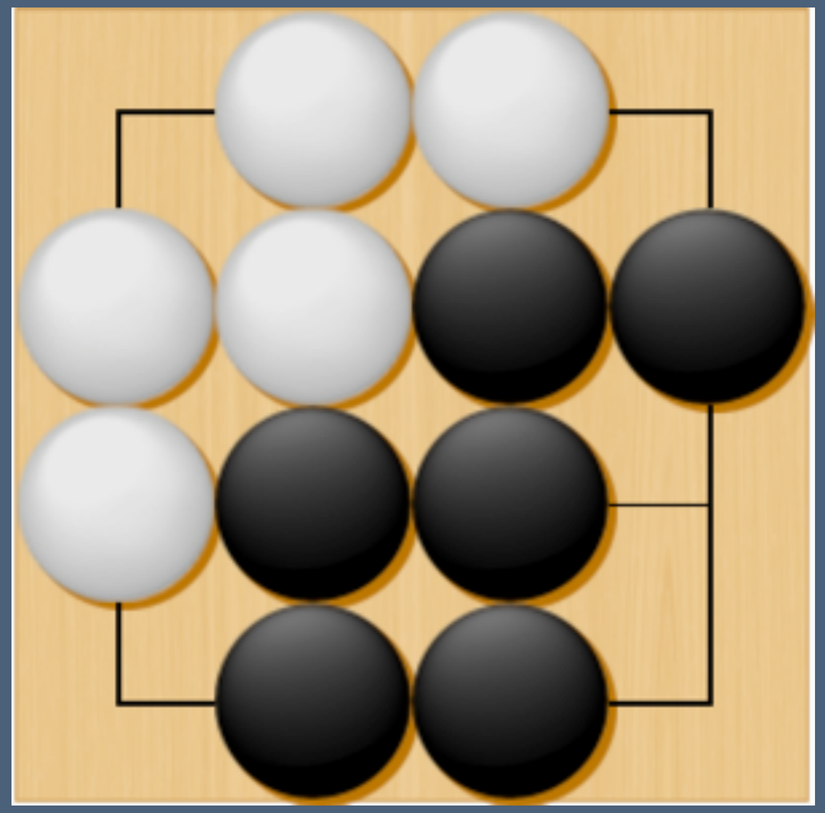


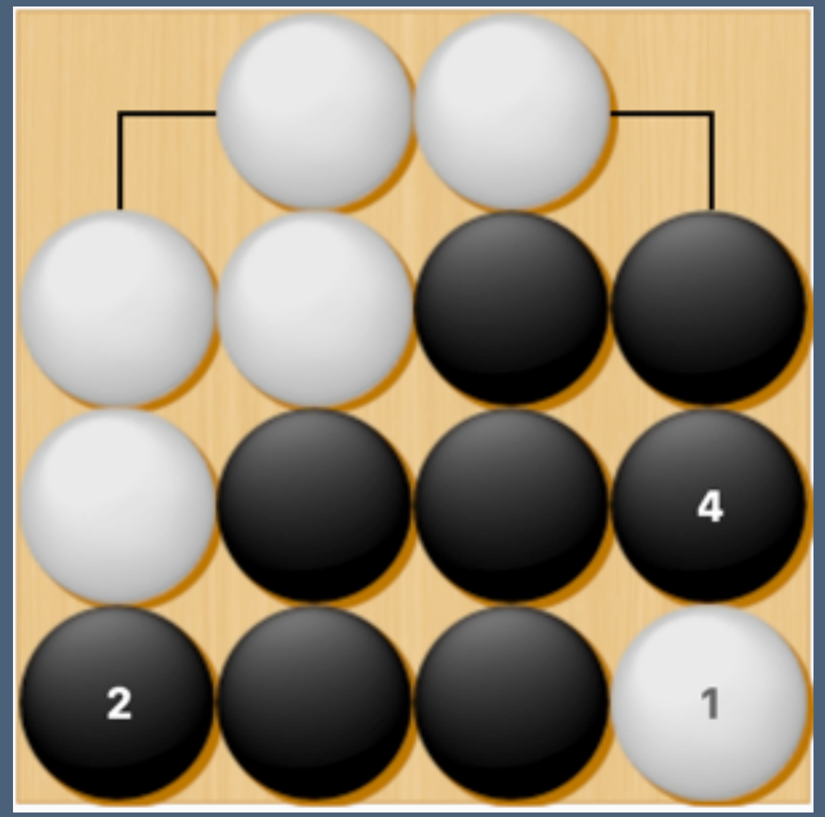


Example: A Seki on the 4x4 Board

- Position from 4x4 Go solution
- Seki - both co-exist,
  no-one can capture the other
- Our current program needs a tree search
  to solve this position
  depth d=6
  438 position evaluations
- Humans can easily see the seki at this point:
  human depth d=0,
  1 human position evaluation

White to play

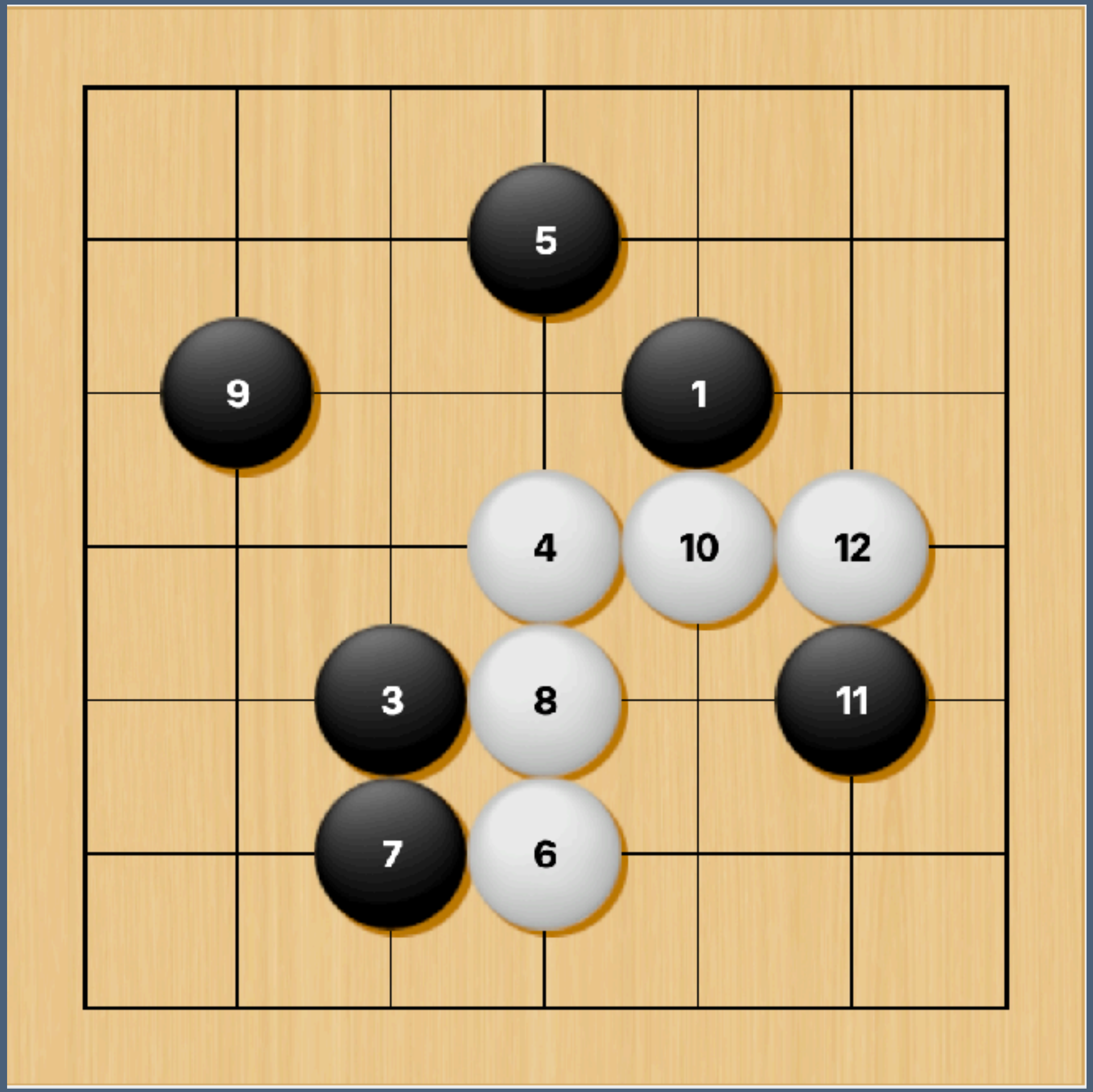




- 1. White D1
- 2. Black A1
- 3. White Pass
- 4. Black D2
- 5. White Pass
- 6. Black Pass

A More Difficult Motivating Example

- Example from a recent talk by Yun-Jui Tsai
- Bottom right is at least seki for white
- Goal: solve many of these positions efficiently
  - Without search if possible?



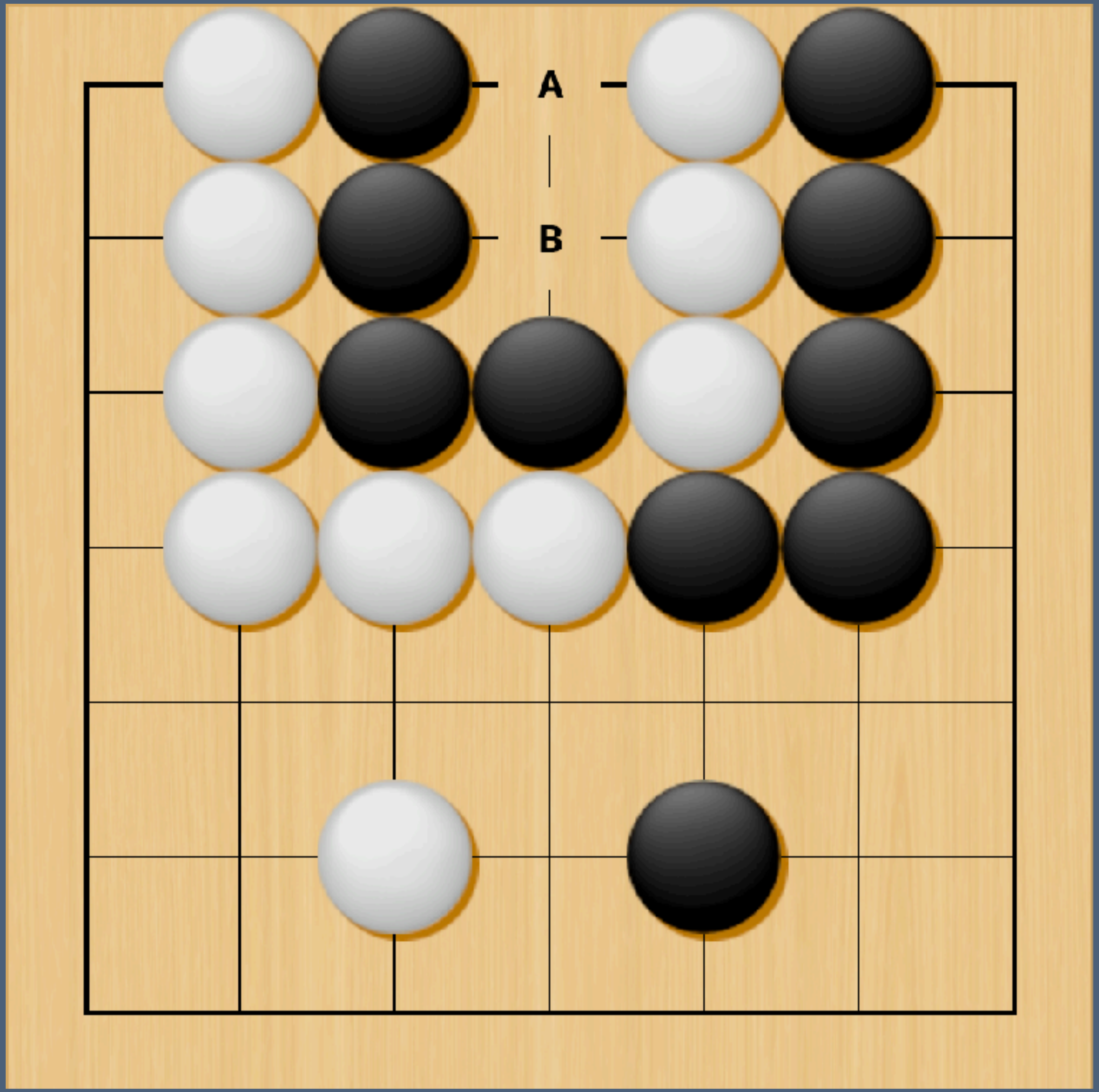
Topics for this Talk

- Coexistence (seki) in Go - examples and definitions
- Review of some old previous work
  - Seki theory
  - Proving seki by search
  - Databases of “large eyes”
- Current and future work: finding seki in small board Go
- Challenge: Can we recognise more seki statically, without search?

Introduction - Seki
Definitions and Examples

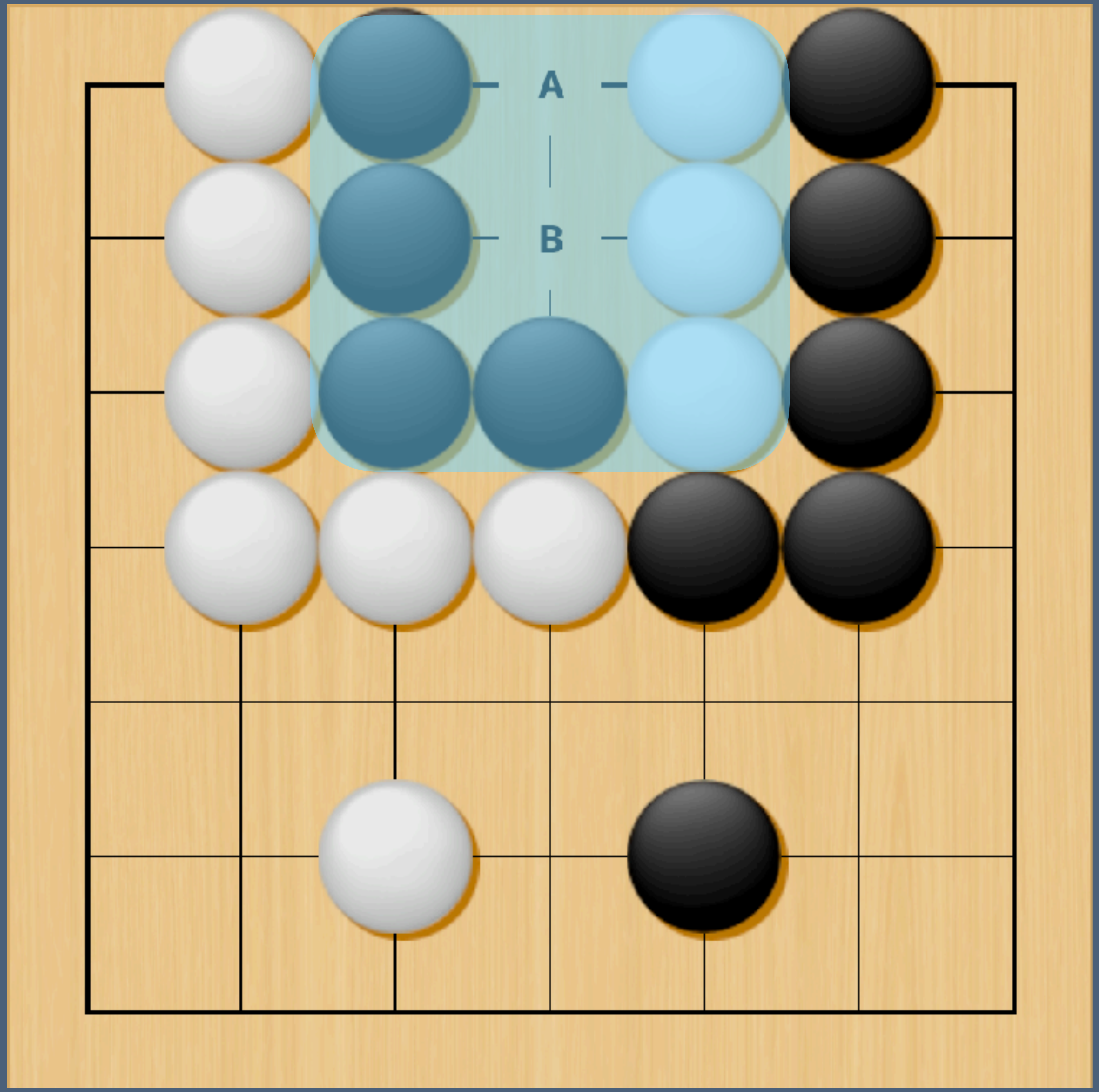
What is Coexistence by Seki?

- Starting point: safety of Go stones
- Standard method: prove that stones of one color have *two eyes*
- *Seki is different!*
- Coexistence (seki): stones of both colors share one or more liberties
  - No side can capture the other
  - Both are alive without two eyes



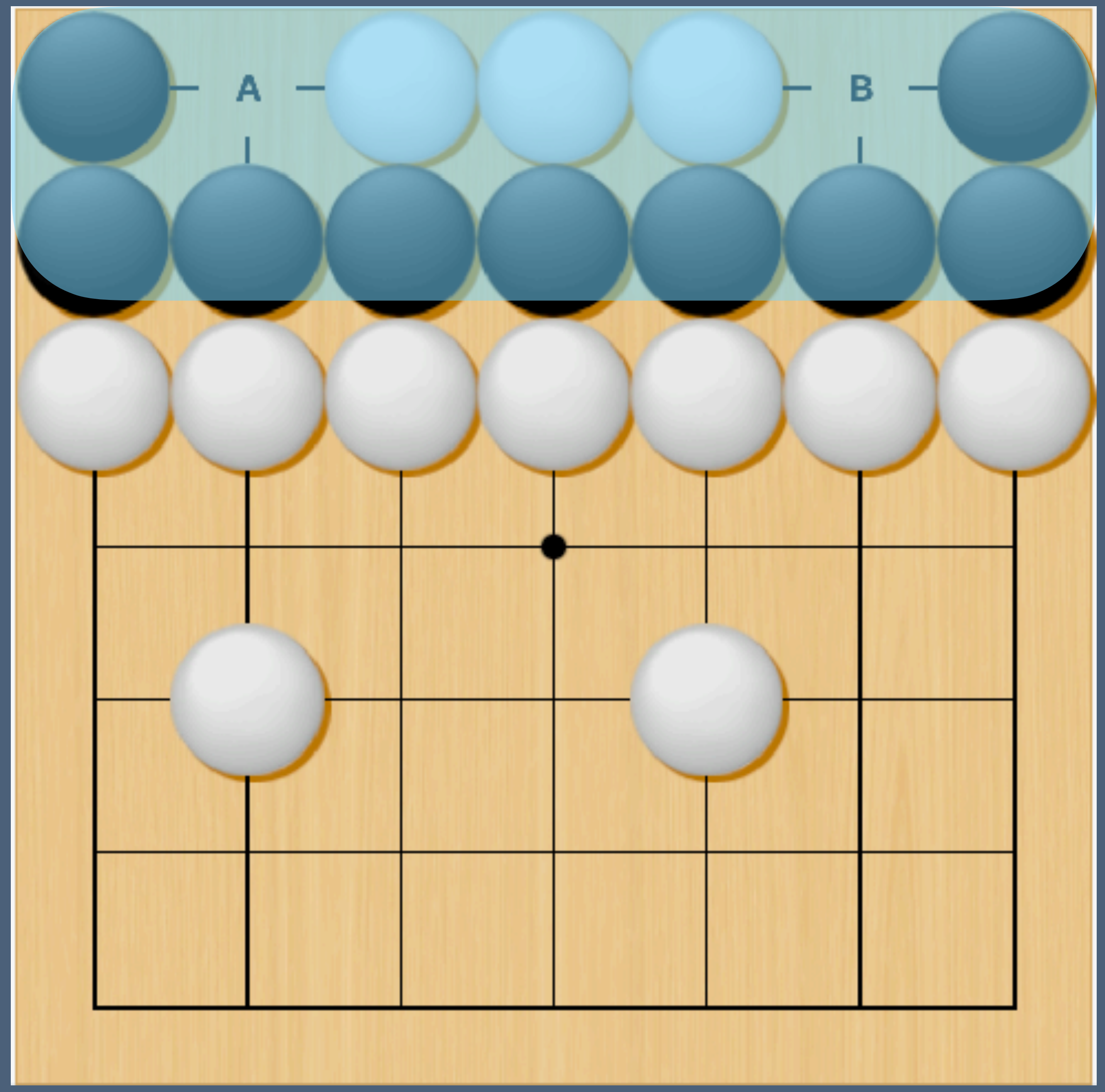
Example Seki, No Eyes

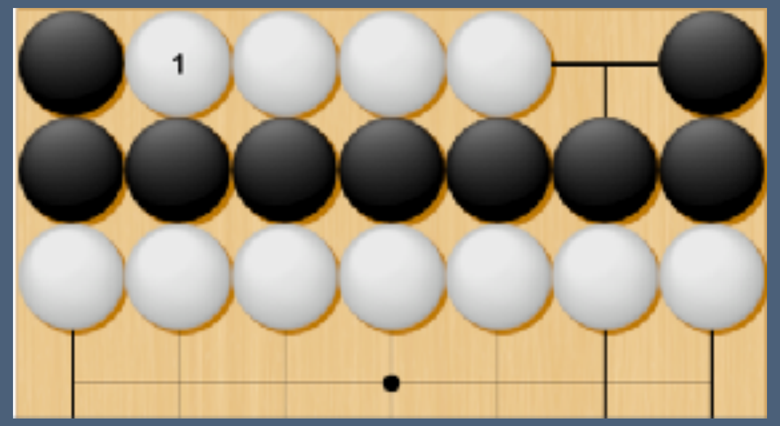
- One black block and one white block
- Share two or more liberties (A and B here)
- Surrounded by safe stones on the outside
- Playing at A or B is very bad for both players
  - The opponent will capture you!

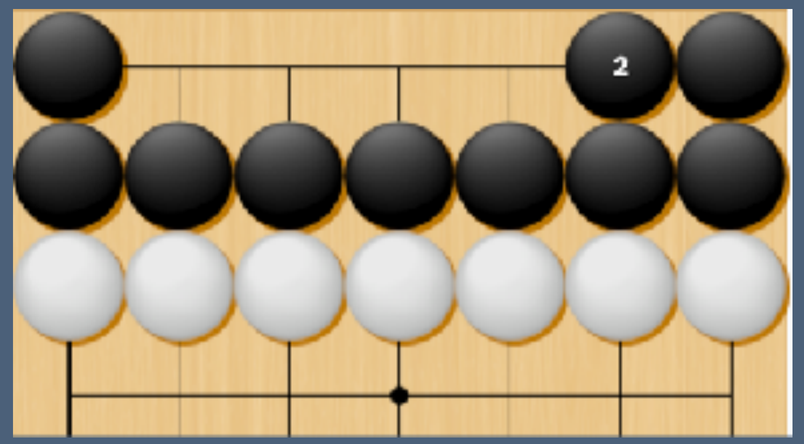


Example - No Eyes, Inside/Outside

- Similar to previous case
- Black stones surround White
- Again, both A and B are bad moves, lose the own stones
- White can try to sacrifice 4 stones...
- ...but Black captures and is alive

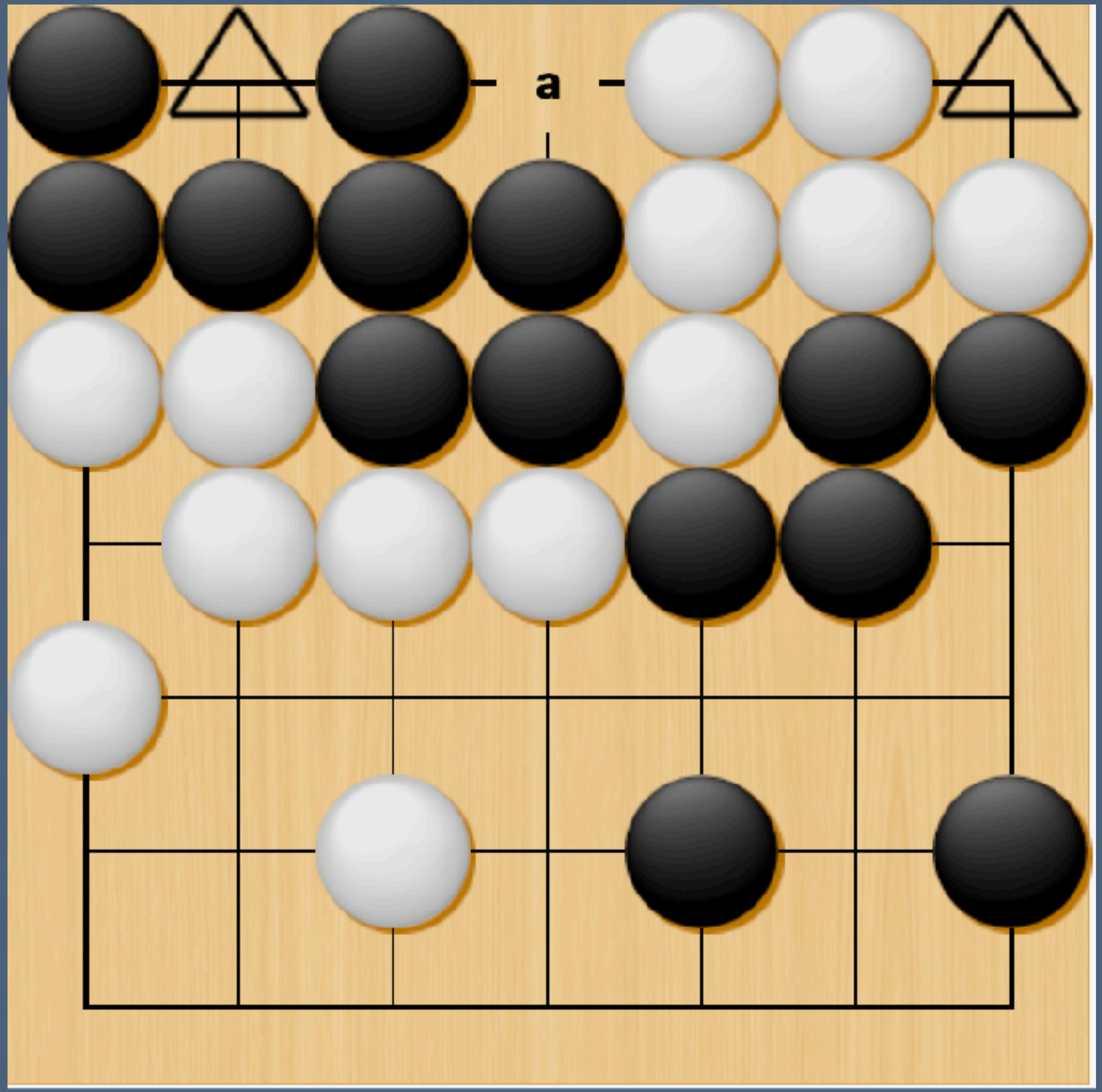






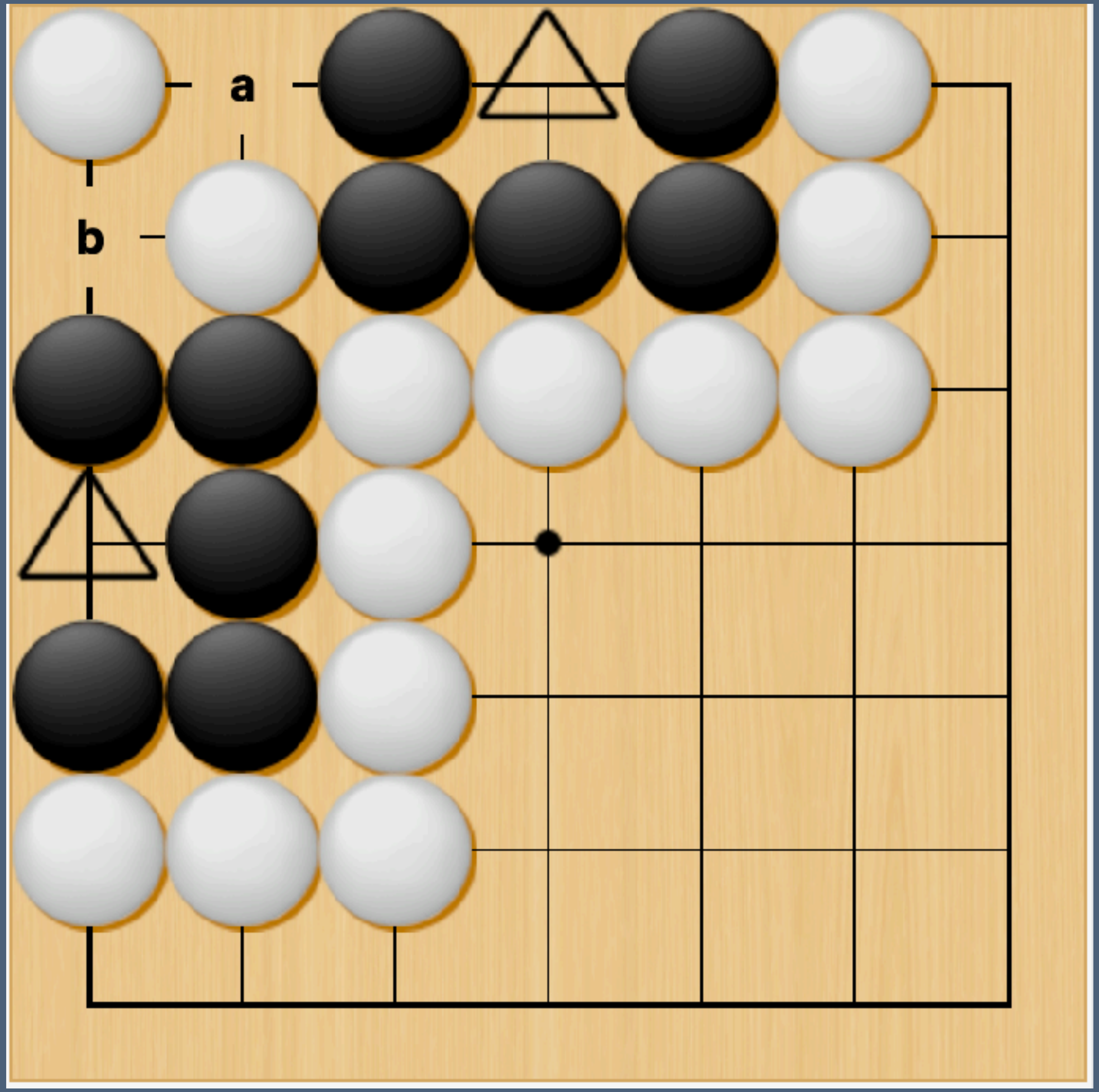
Example - One Eye Each

- Both sides have one eye (△)
- Shared liberty at **a**
- Again, playing **a** is very bad for both sides



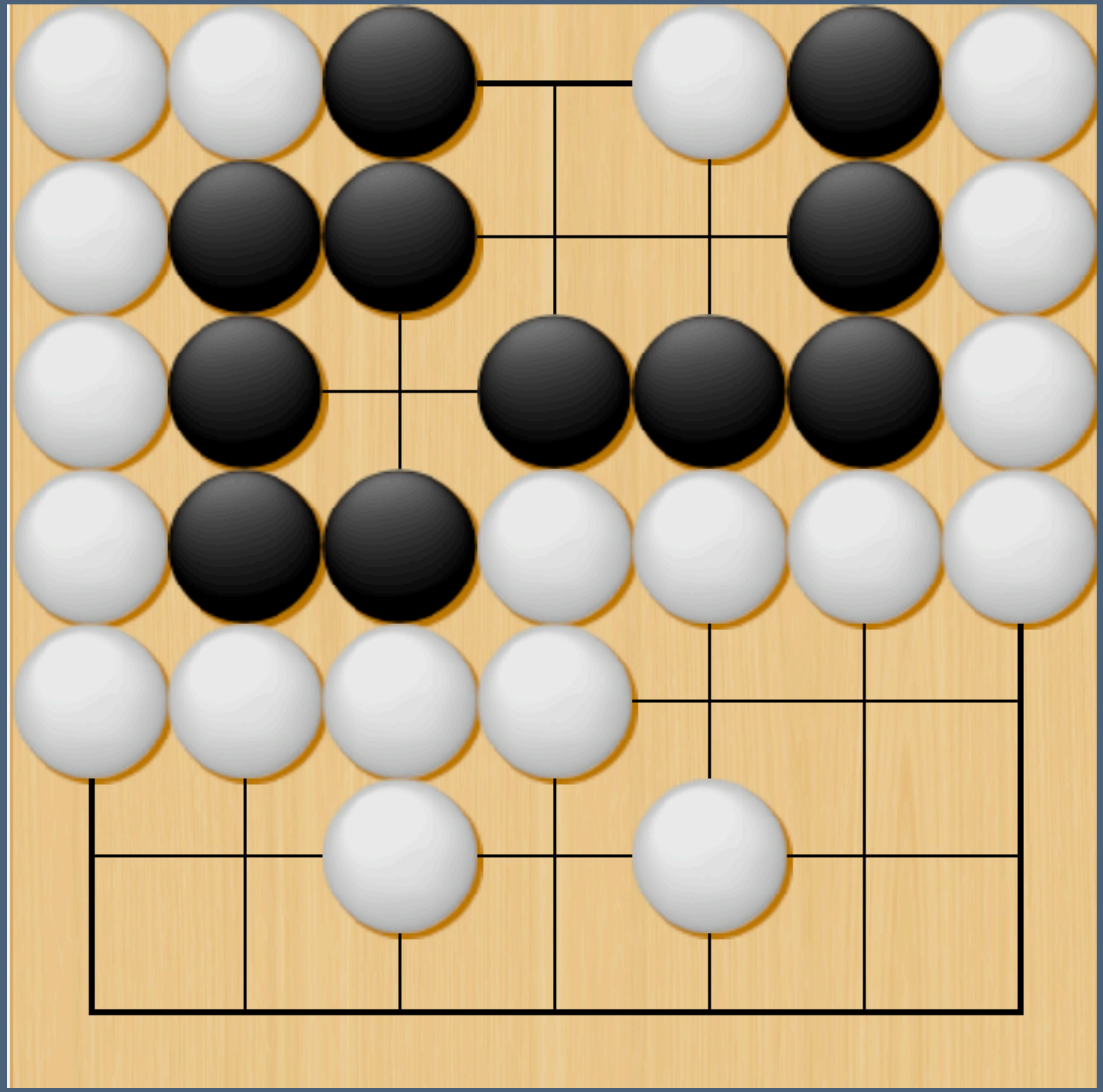
Example - Two Eyes, but Separated

- Black has two eyes, but..
- ...white's stones split them
- Again, no good moves for either player
- All stones are alive in seki



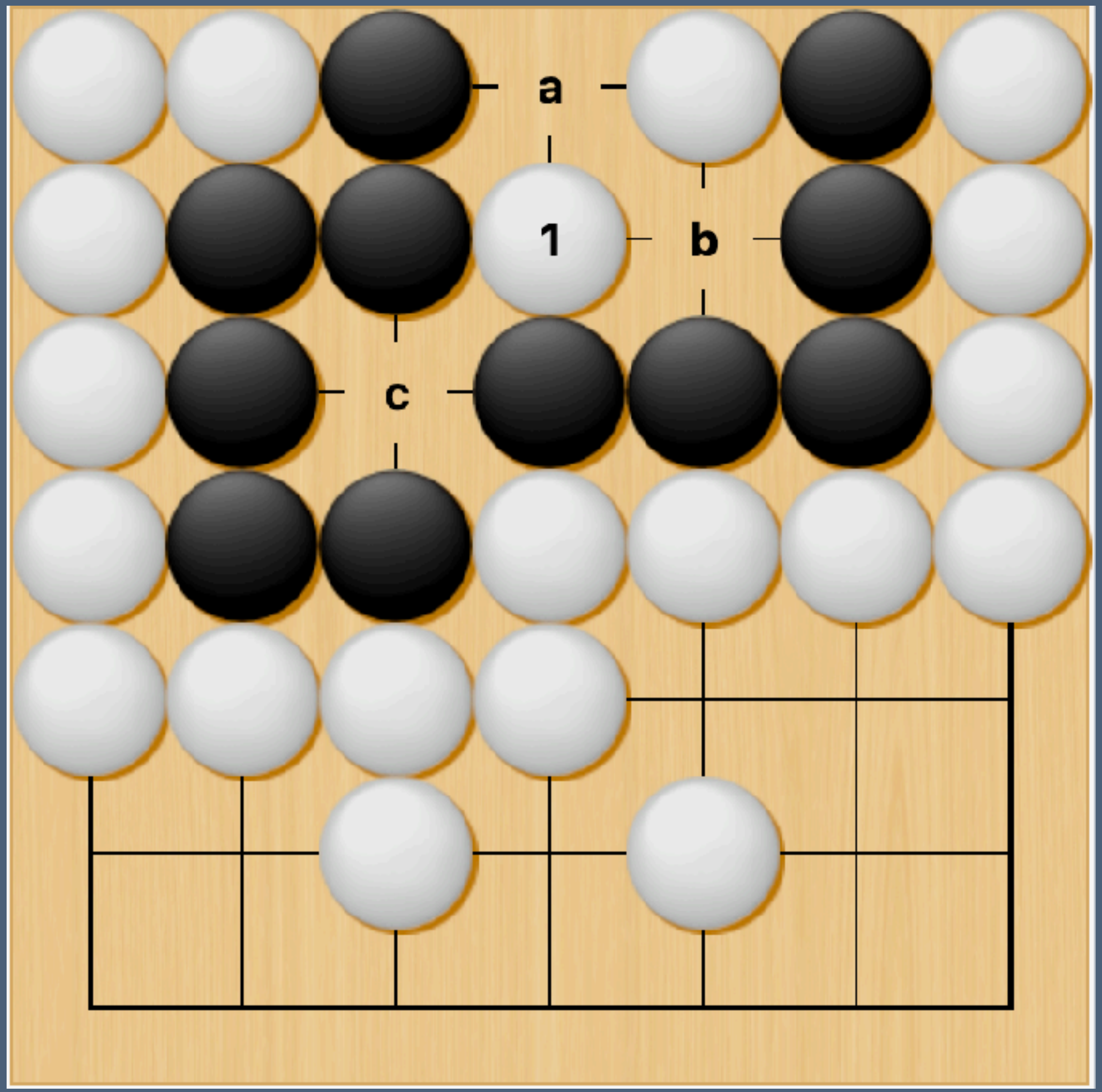
A Seki Surprise...

- The Black area looks safe...
- Many Go programs think so, too...



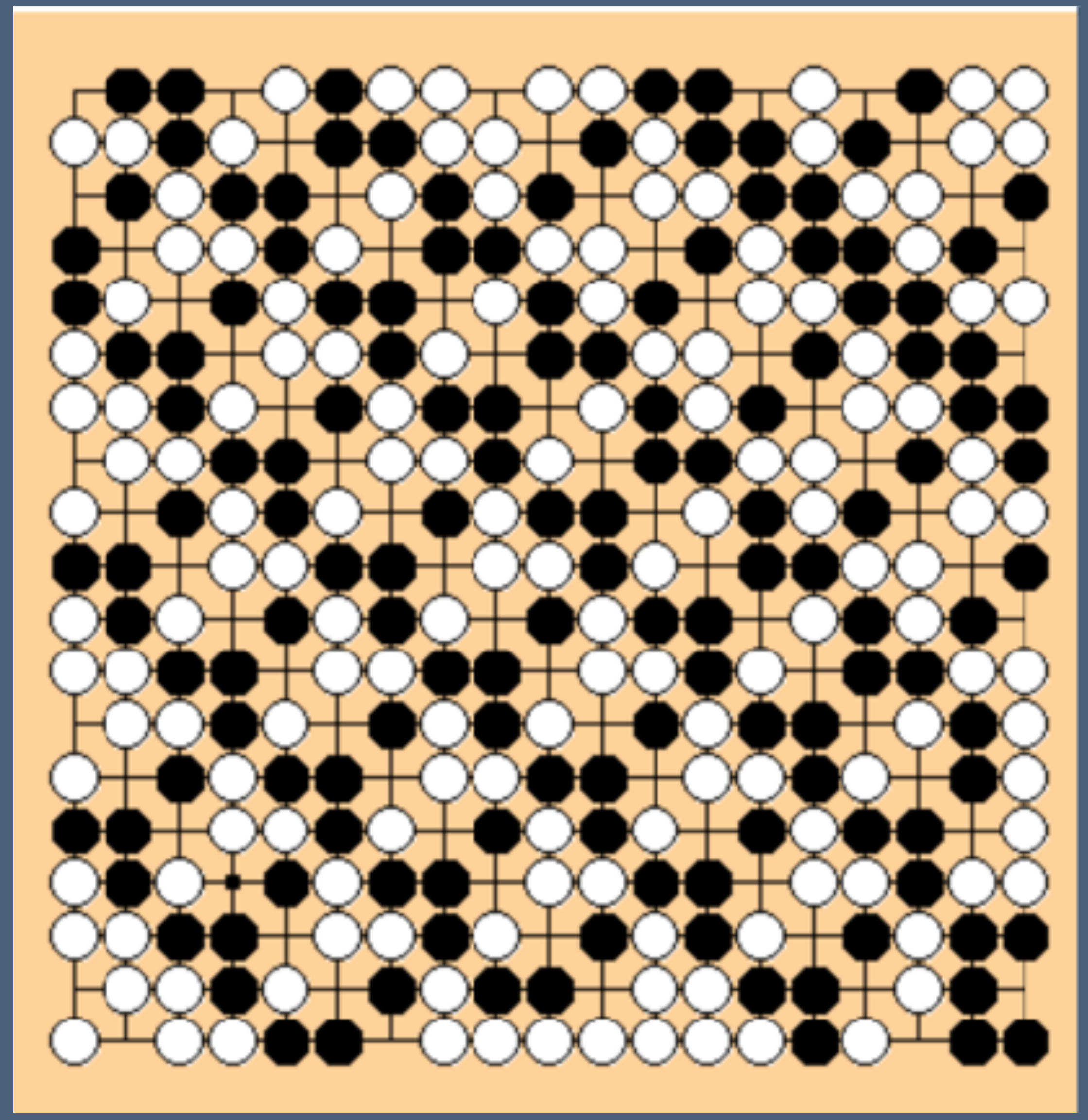
A Seki Surprise…

- …but white can make a seki!
- If Black tries a or b…
- …then White captures at c
- So it is seki, both are alive



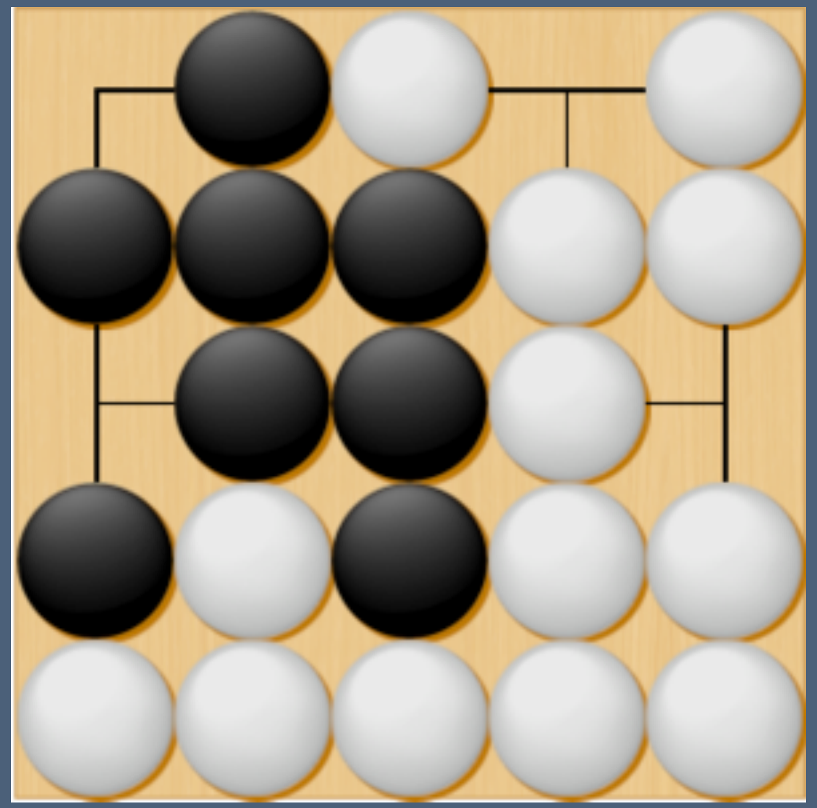
Complicated Seki

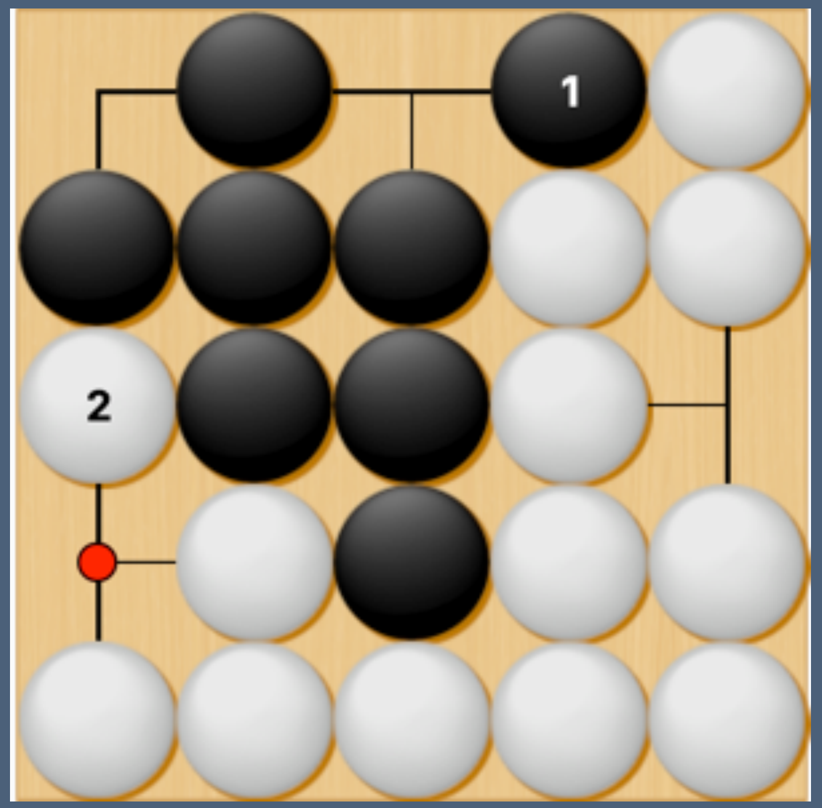
- Seki can be very complex
- Many liberties
- Many blocks of stones mixed up



G. Hungerink, 129 blocks in seki

- Can involve Ko fights (打劫)

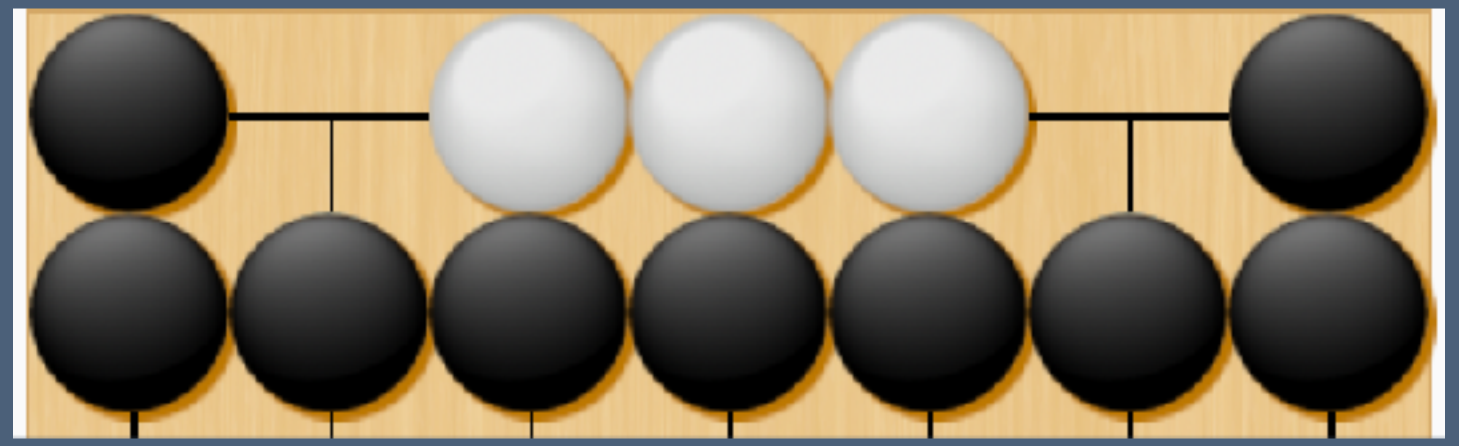




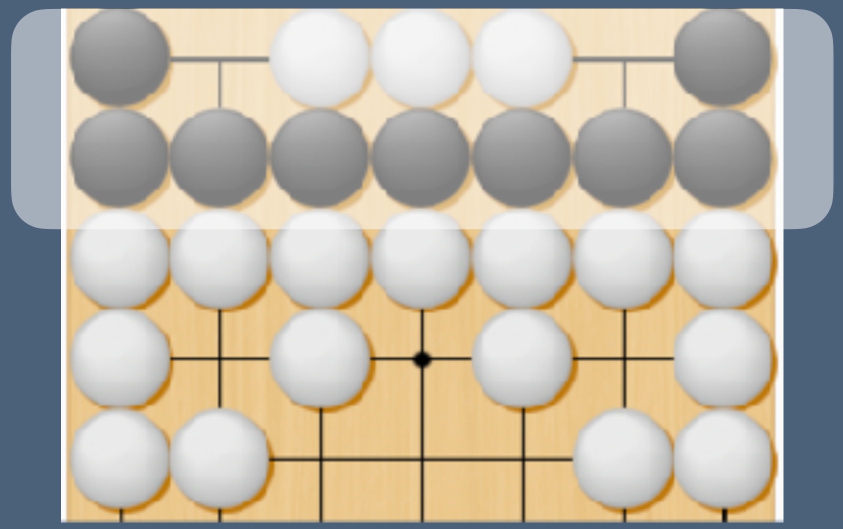
Seki with a double ko

A Simpler Goal - Is One Player Alive?

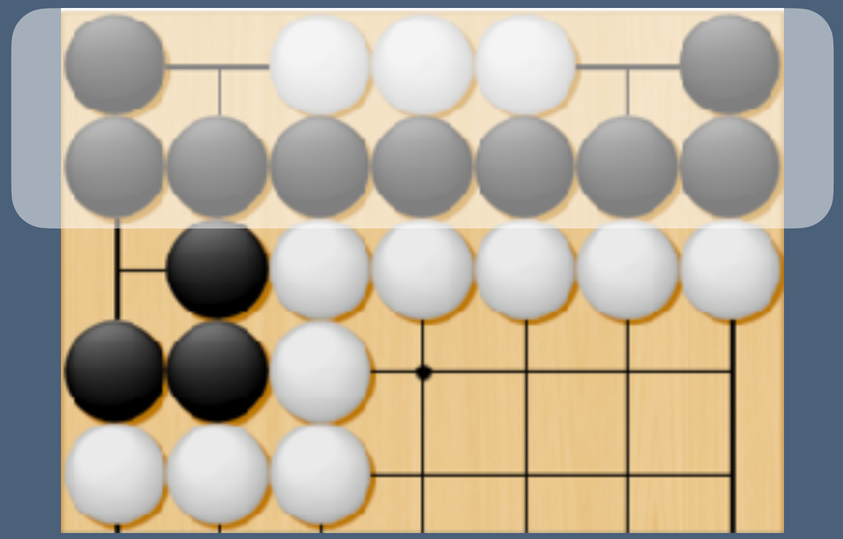
- For seki, both players' stones are alive
- Sometimes, we are only interested in one player
  - At least seki means: seki, or two eyes
  - This is enough to prove that one player's stones are alive
  - Advantage: At least seki is a local property
    - Can be proven by local search, in one region
- Application: Killall Go
  - If white gets at least seki then white wins in killall



At least seki for Black - Alive



Black is Alive - True seki



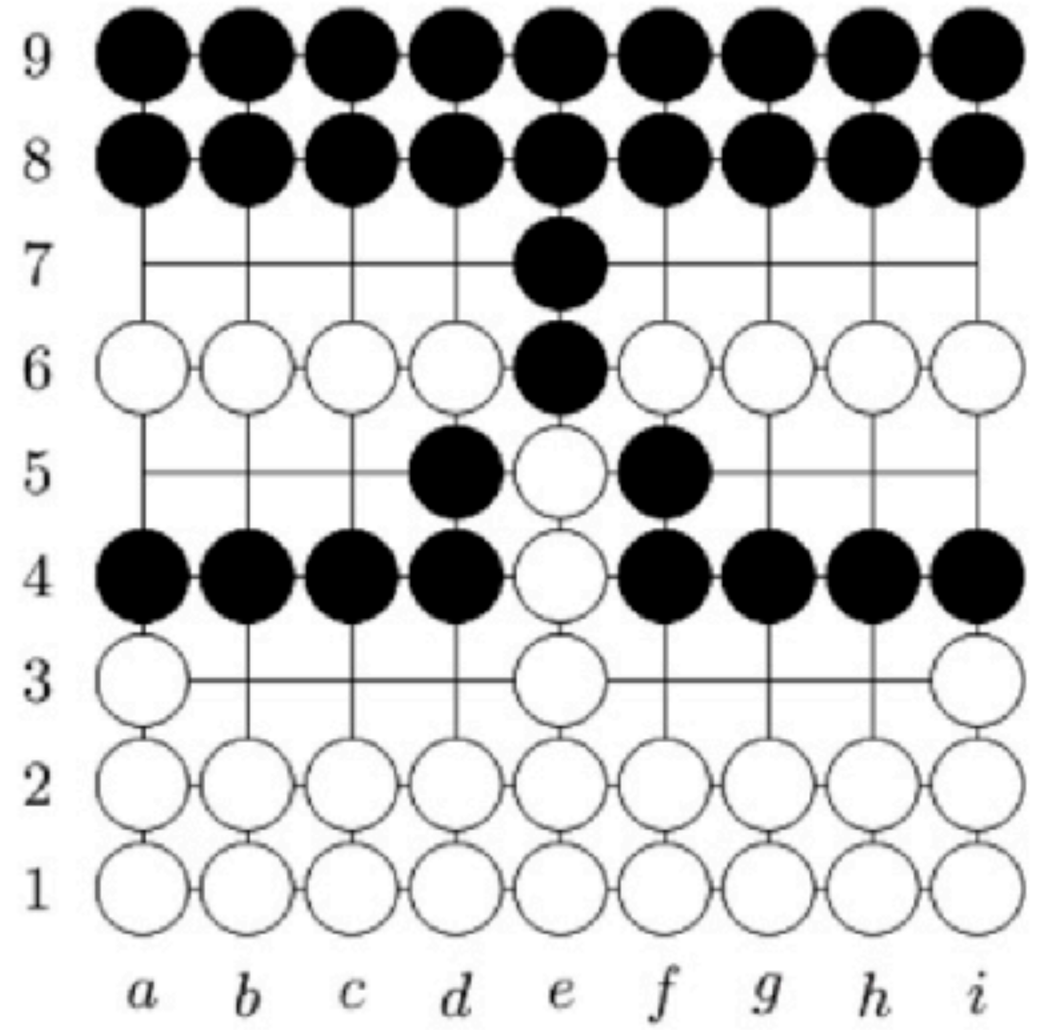
Black is Alive - Two eyes

Previous Work - Theory of Seki

Previous Work - Seki Theory (1)

- Gol'berg, Gurvich et al, Discrete Mathematics (2014)
- Study mathematical abstractions related to seki
  - Model number of shared liberties in a matrix
  - Mathematical conditions for when one side can win
  - Simple shared liberties, many blocks of stones

A. Gol'berg et al. / Discrete Mathematics 329 (2014) 19–32



0	3	3
4	0	3
4	3	0

Fig. 4. A semi-complete seki position in GO and the corresponding semi-complete seki matrix.

Not very useful in practice?

Previous Work - Seki Theory (2)

- T. Wolf, Seki with 2 Liberties per Chain in the game of Go (2016)
- Mathematical characterizations of "equivalent" seki using concepts of common fate graph (CFG) and basic seki graph (BSG)
- Special case where all blocks have 2 liberties

Not very useful in practice?

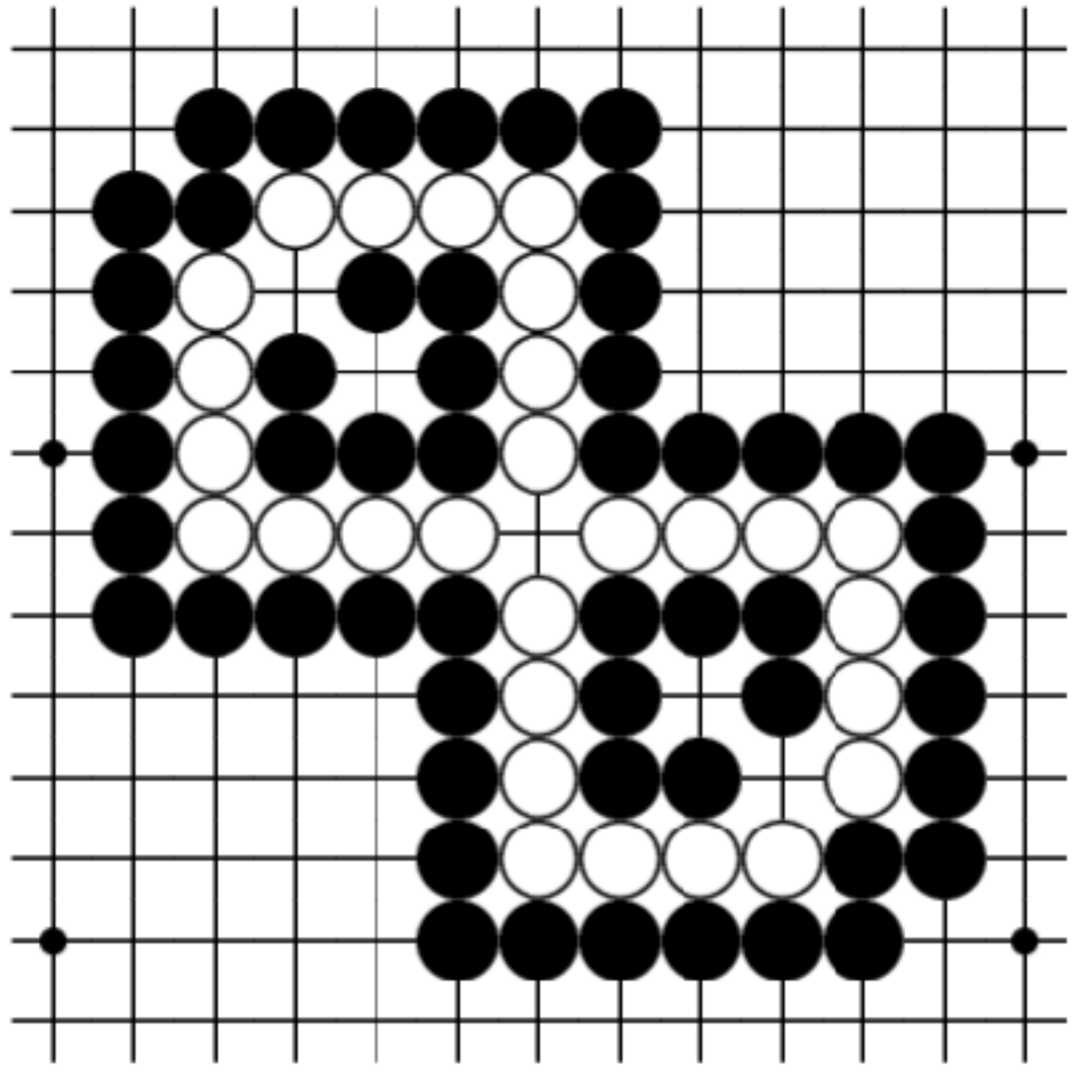


Diagram 22. "No 8"

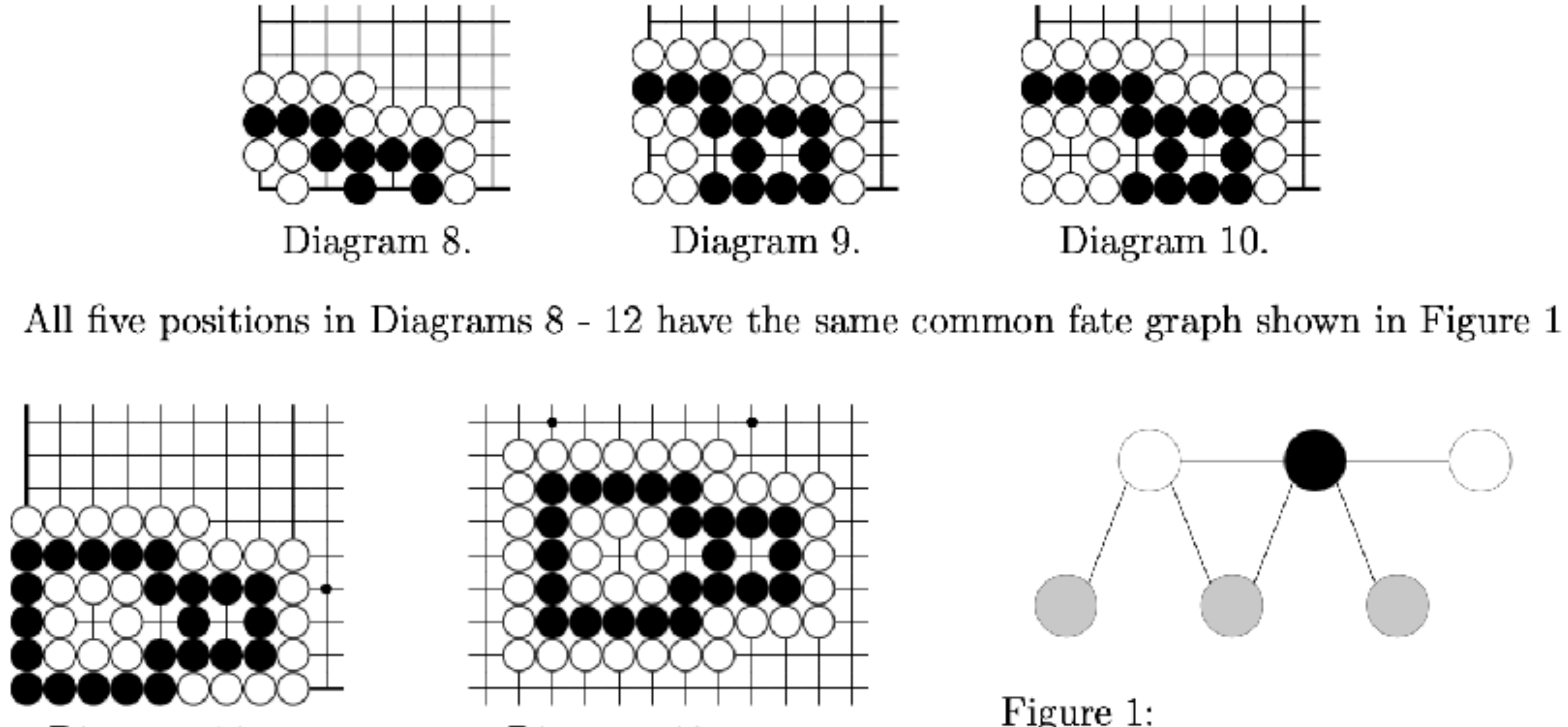


Diagram 8. Diagram 9. Diagram 10.

All five positions in Diagrams 8 - 12 have the same common fate graph shown in Figure 1

Diagram 11. Diagram 12.

Figure 1: The corresponding CFG

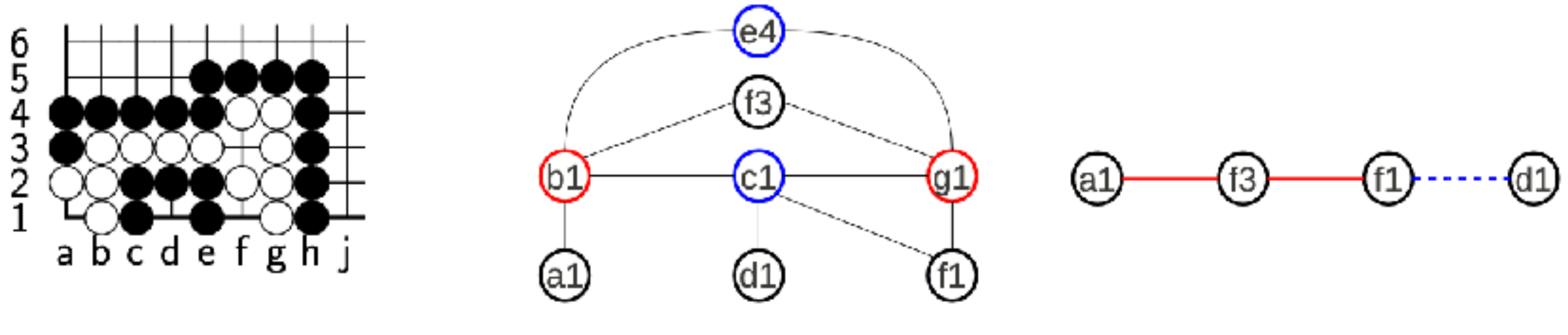


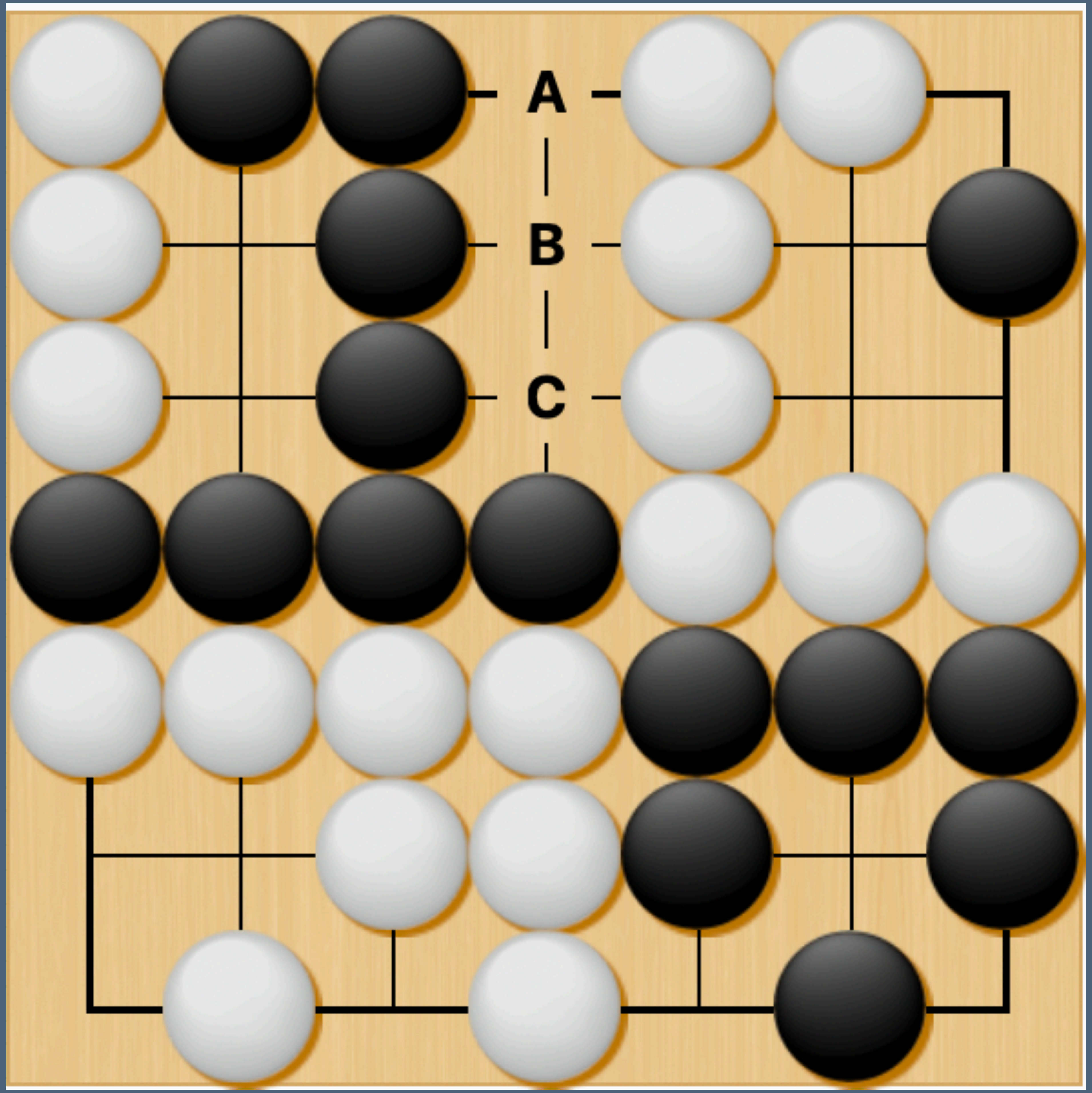
Diagram 20.

Figure 2: The corresponding CFG and BSG

Previous Work - Seki in Capturing Races (对杀)

- M. Müller. Race to capture: Analyzing semeai in Go. Game Programming Workshop in Japan (1999)
- Exact solutions for simple “class 1” capturing races
    - Outside liberties, shared liberties, plain eyes
- Based on traditional semeai formula (Lenz 1982, Hunter 1998)

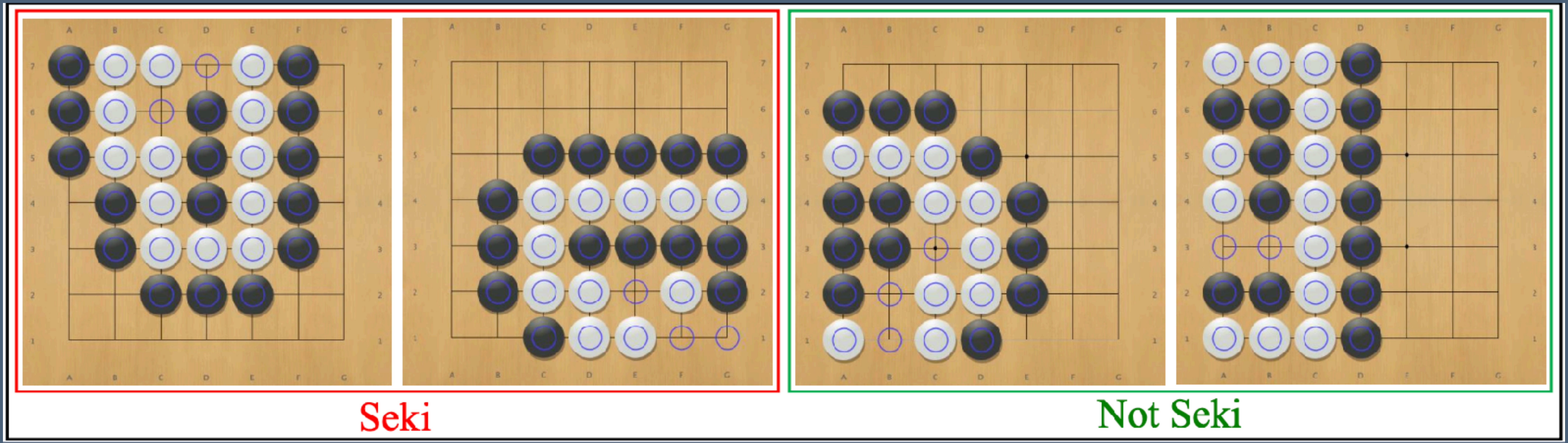
Not very useful in practice?



A seki. White cannot fill all of A, B and C

Seki Table

- Work by Academia Sinica group - T. Wu, C. Chin, Y. Tsai
- Regions of size 5 to 8
- Which ones are at least seki?
- Used in newest version of Killall Go solver - very useful!



Previous Work - Solving Seki by Search

Solving Seki by Search

- Niu, Kishimoto and Müller,
  Recognizing seki in computer Go,
  Advances in Computer Games (2006)

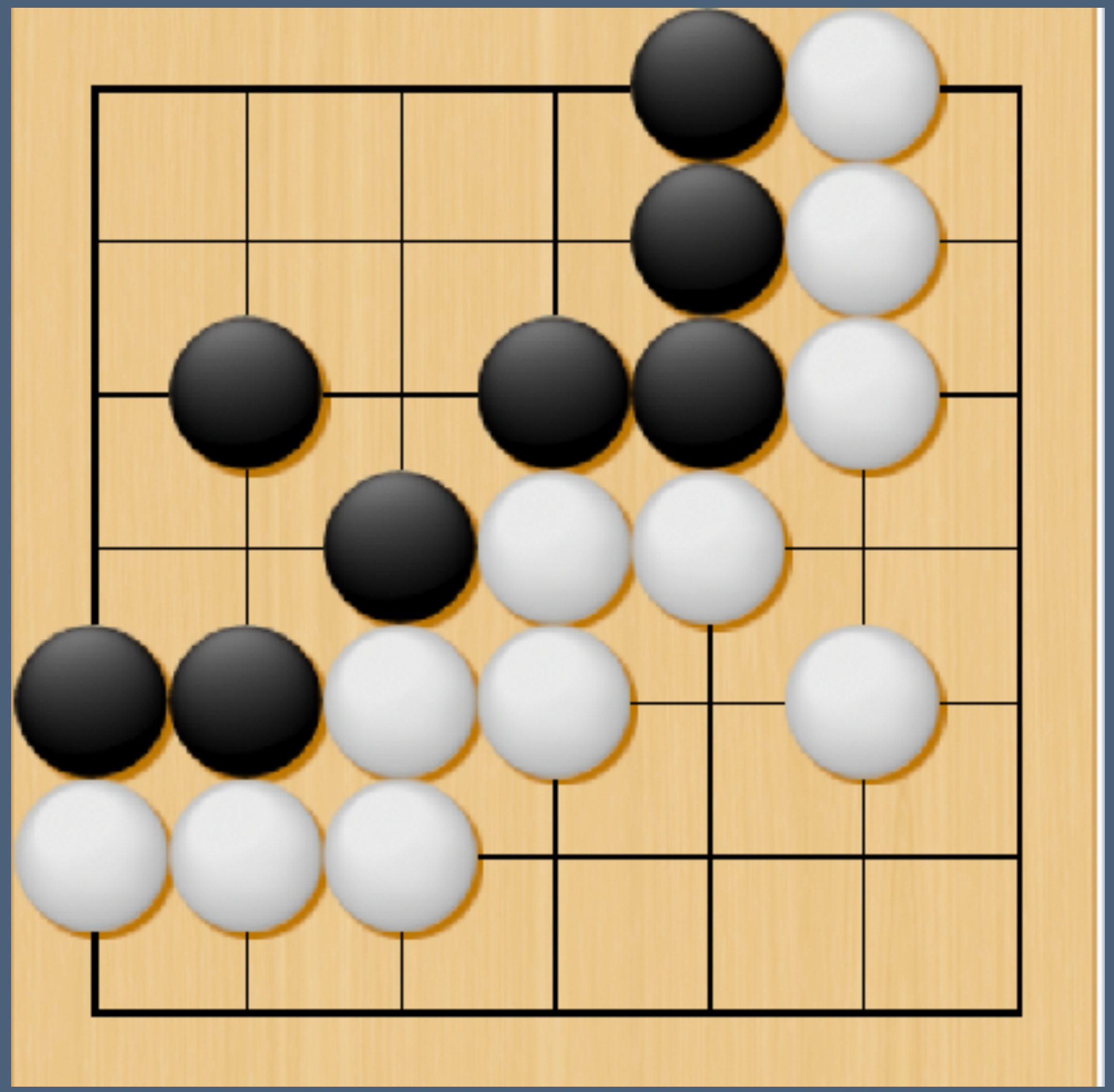
- Board divided into regions,
  surrounded by one player

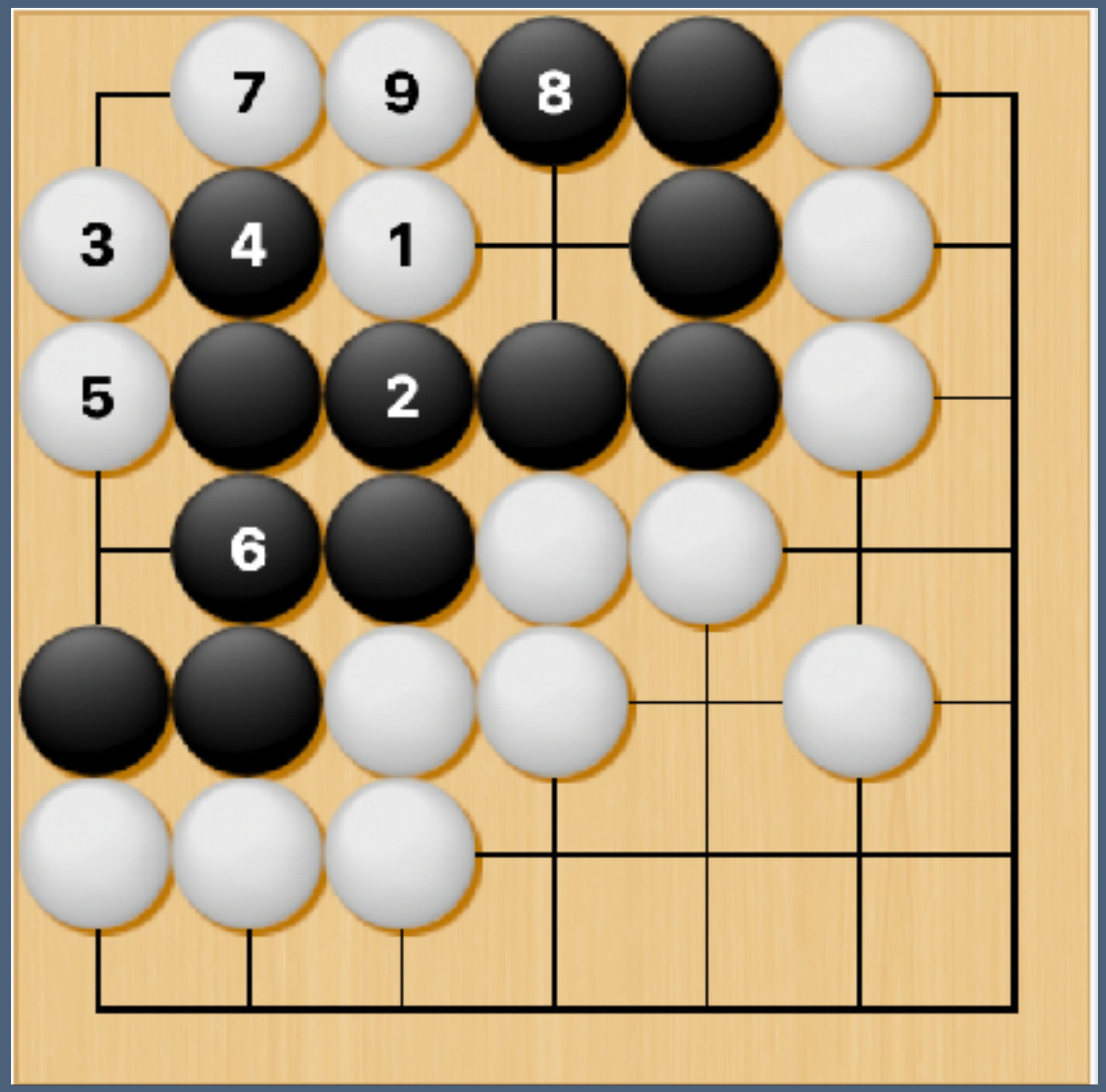
- Two true/false searches:

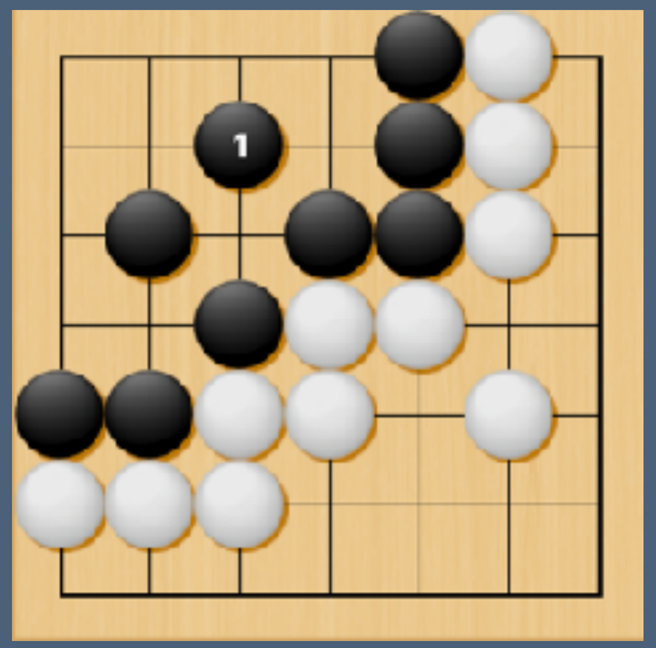
  - Black is seki winner: can Black stones live?

  - White is seki winner: can White stones live?

- If both searches return true, then it is a seki

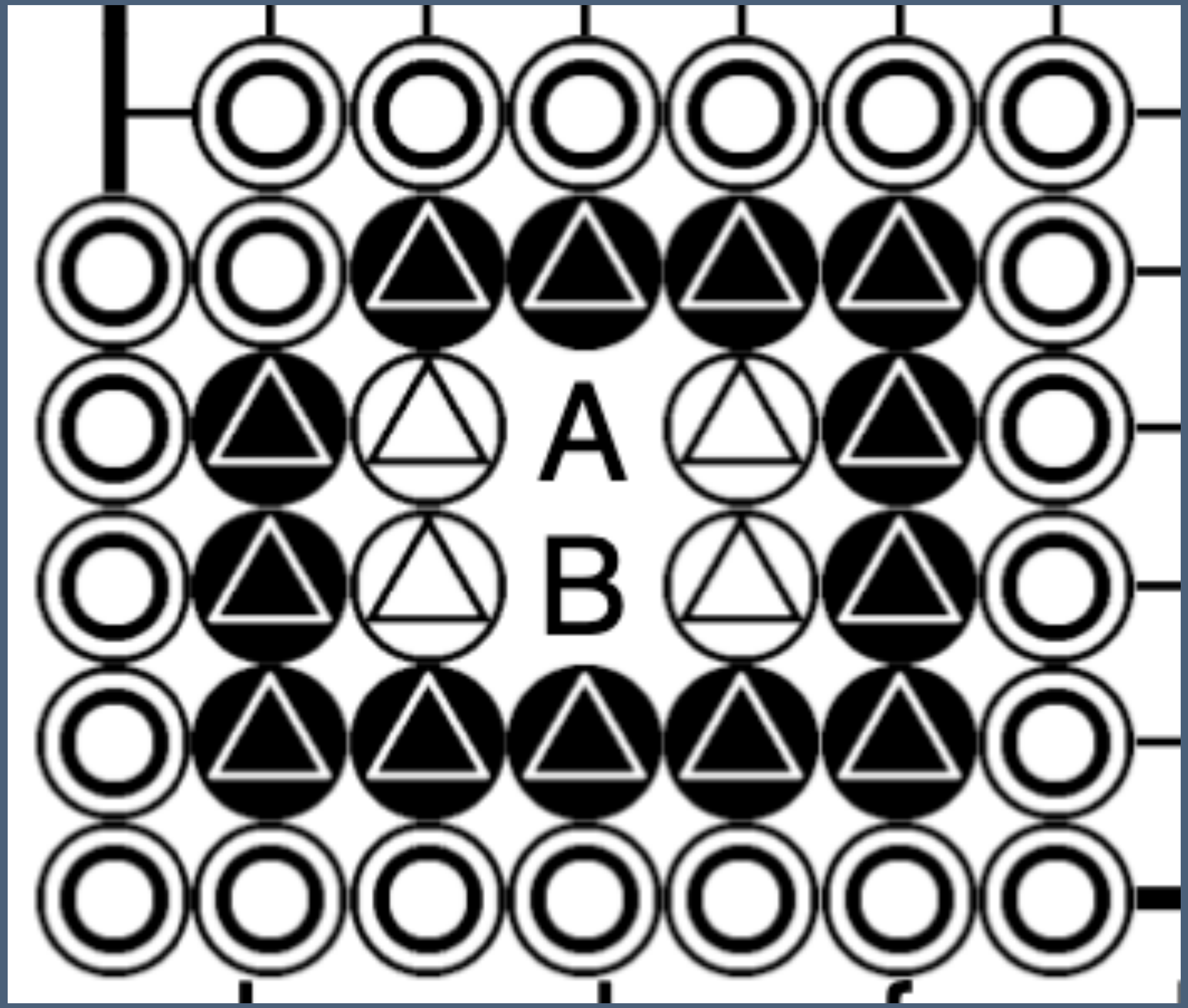






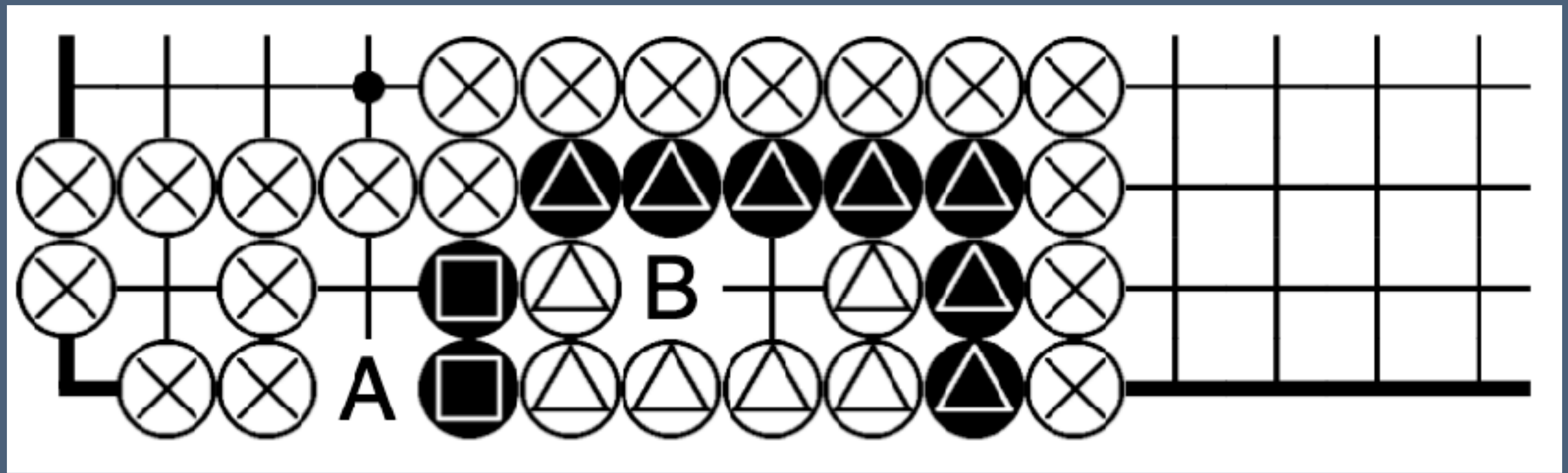
Solving Seki by Search - Local Search

- Local result of seki search, in one region
  - At least seki, at most seki
  - Uses only local information inside a region
  - Example
    - At least seki for Black
    - At most seki for White
    - Final result depends on outside white stones
    - If captured, then the seki collapses



Solving Seki by Search - Global View

- Combine local searches globally
- What is the final result?
- Input: local status of all blocks
  - At least seki, at most seki, and alive
  - Global combination can determine true seki, or alive, or dead



Example: Region B: local seki

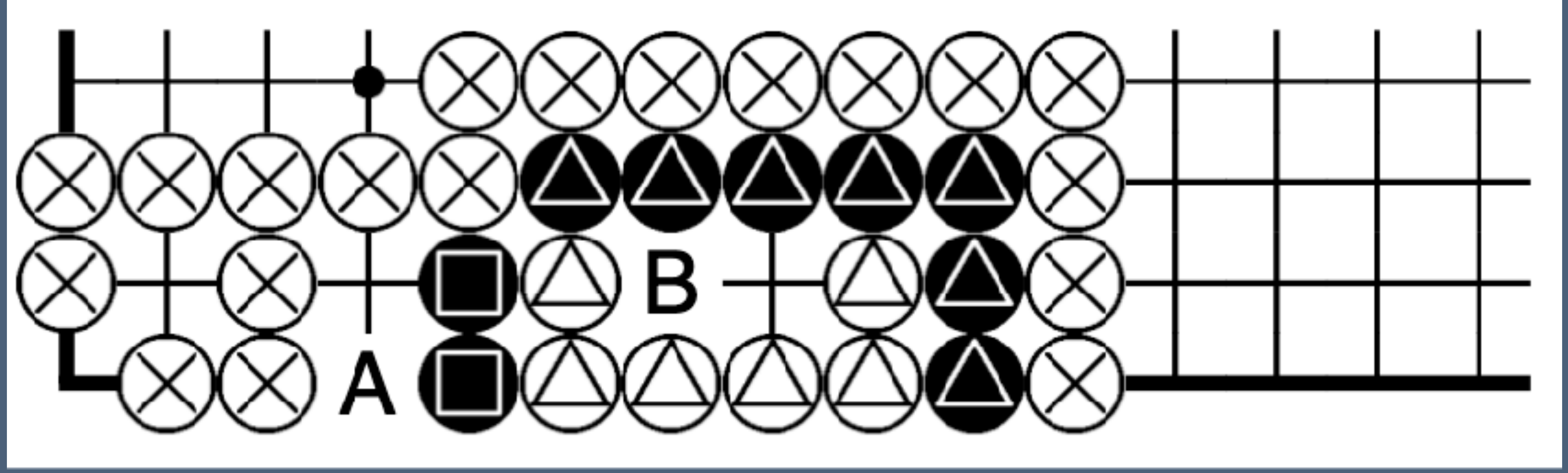
White outside stones X are safe

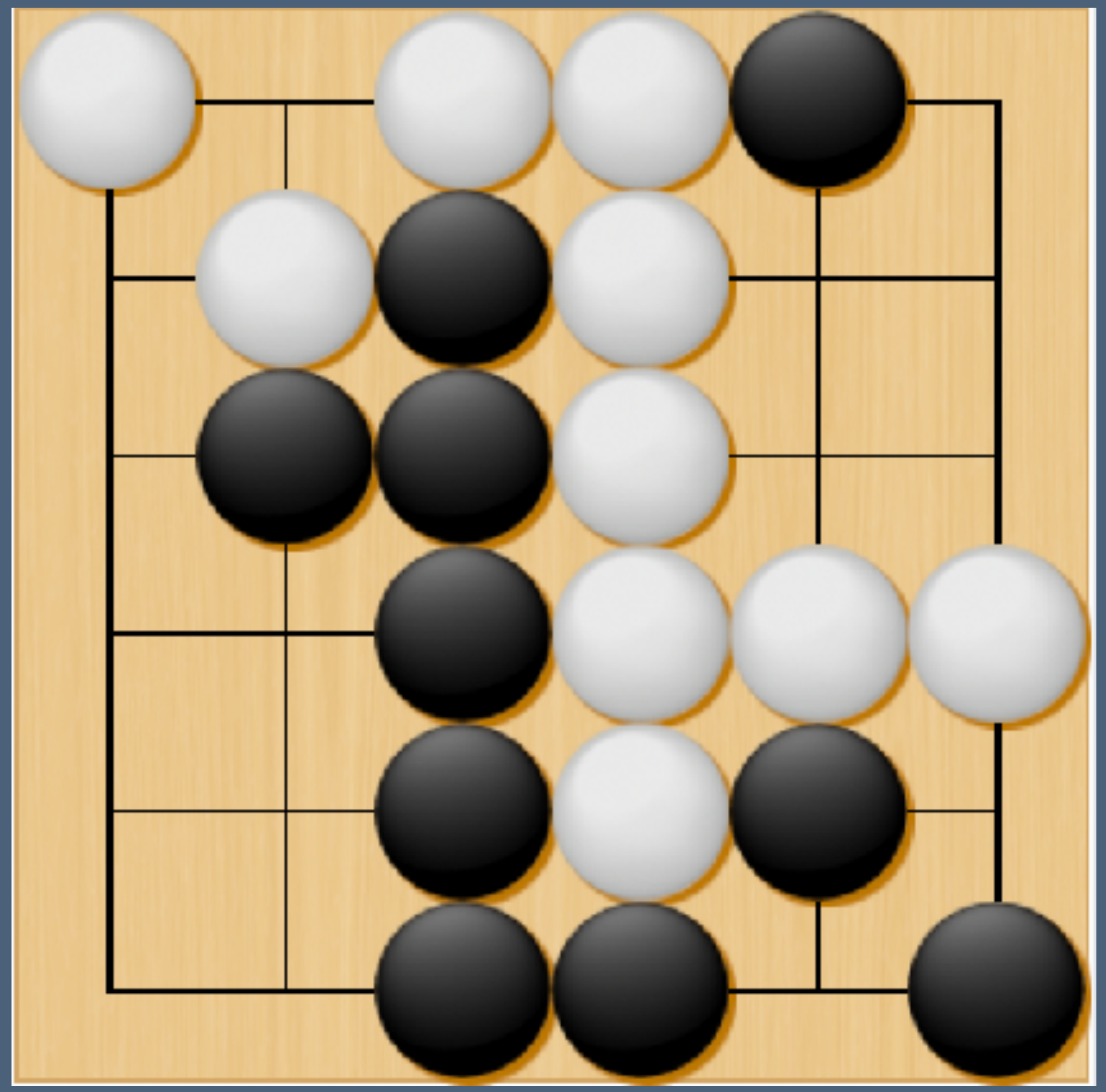
Black outside stones in A have no eyes

Result: no seki, Black is dead

Solving Seki by Search - Limitations

- Local search in each region
- Global combination
- Powerful if regions are well-separated
- Not so useful for solving small boards
- Separation happens too late





Previous Work -
eye Classification and Databases

Static Rules for Regions

- R. Vila, T. Cazenave, When one eye is sufficient: a static classification.
  Advances in Computer Games 10 (2003),
  https://www.lamsade.dauphine.fr/~cazenave/papers/eyeLabelling.pdf
- "Eye" here means fully enclosed region
- Which regions are large enough so they cannot be killed?
  - Includes both 2-eye regions and at-least-seki
  - Some rules based only on shape of region, not on stones inside

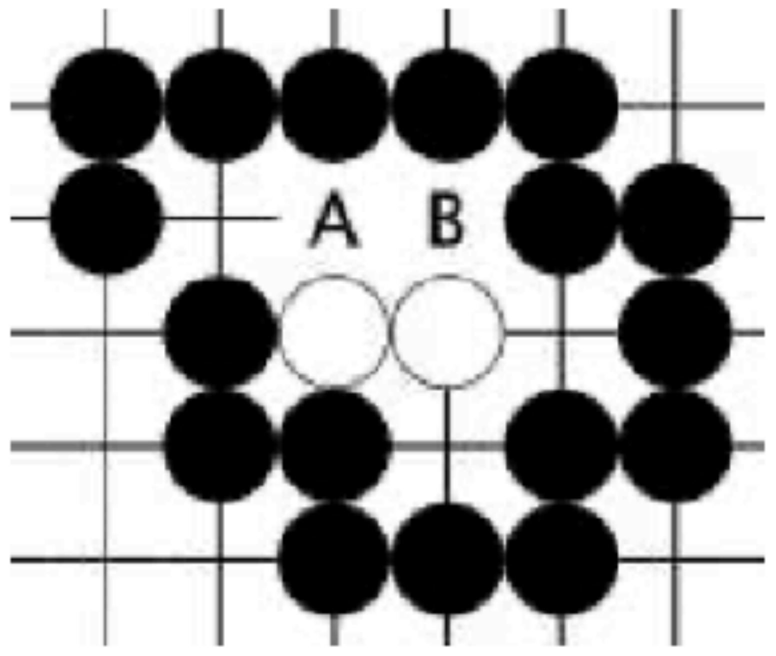


Figure 6a. Alive status for a [1112234]-β eye. Even though these shape can be filled with a rabitty six, A and B belong to the set of *vital points* so we have a miai of life.

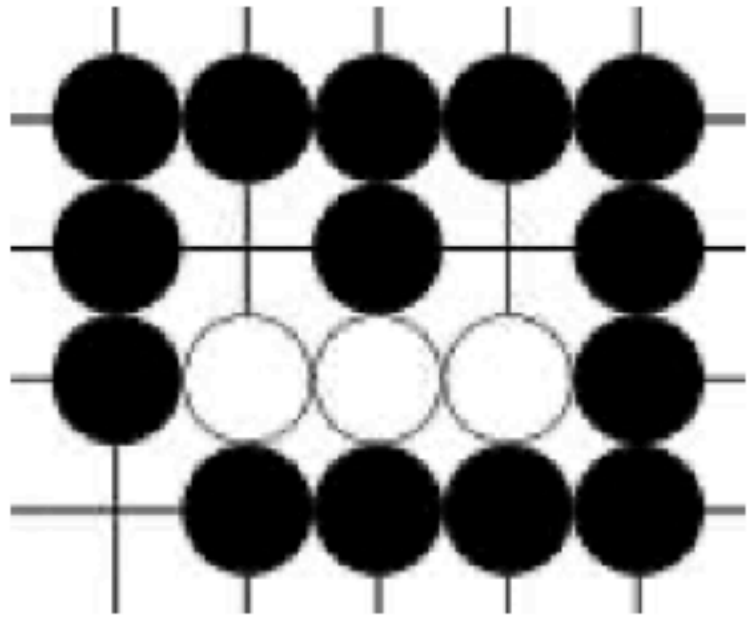


Figure 6b. Alive status for a [11222] eye. No matter how many stones plays White inside, Black is unconditionally alive. The opponent cannot fill the eye space with a *nakade shape* of size four.

Future Work - Solving Small Board Go

How to use a Seki Solver for Small Boards

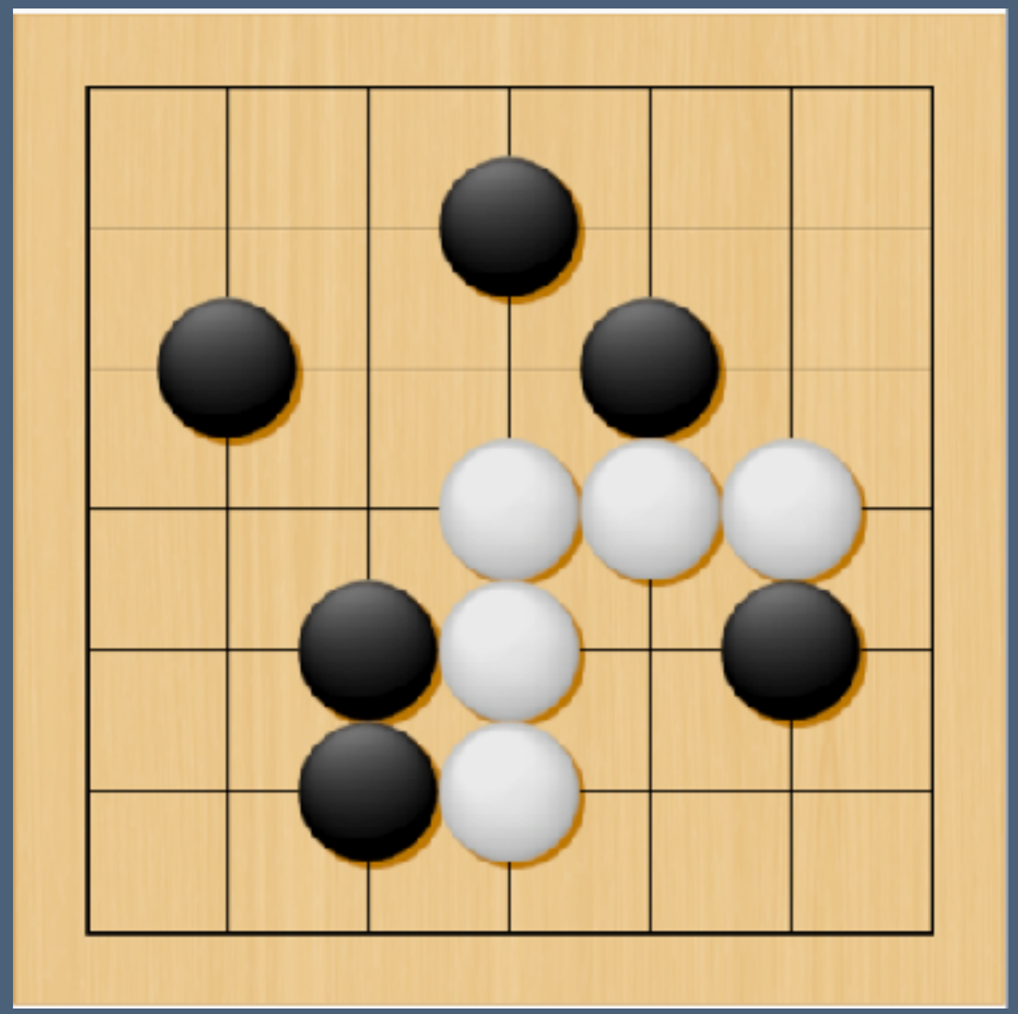
- Killall Go:
  - At-least-seki for white is enough (White wins)
  - Many types of seki-or-ko are also enough (no ko threats, White wins)
- 6x6 Go:
  - At-least-seki is sometimes enough (lower bound)
  - For finding an exact score, need to fully solve seki

Challenges for Small Board Go

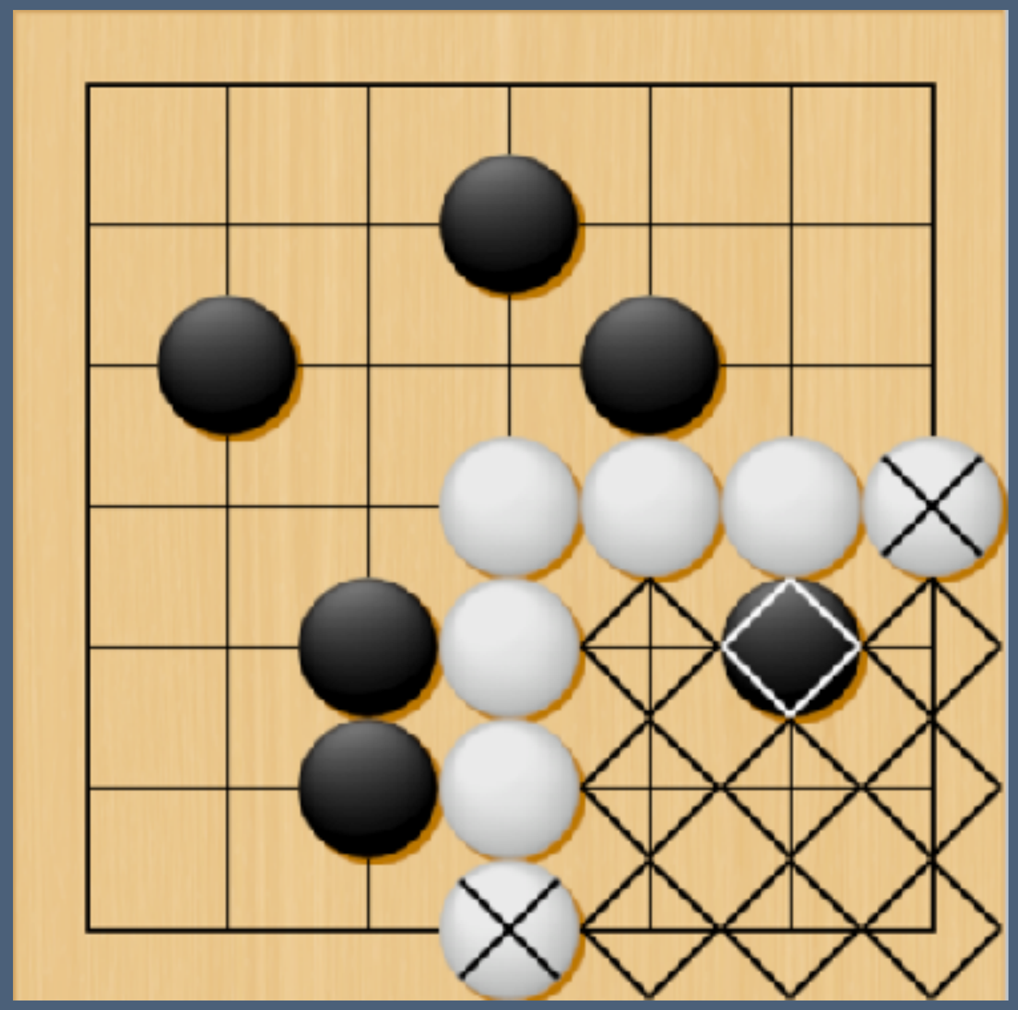
- Methods discussed so far work well if regions are already surrounded
- It may take too long to completely surround a region in real games
- In the search, we will eventually see those states
- How can we back-up seki knowledge to (much) earlier states?
- Humans can do this...

What we want to solve...

...at least seki for white



What we can solve...



Research Ideas

- Open board seki solver
- Large database of eyes, alive shapes, territories
- Open-boundary region database

Towards an Open Board Seki Solver

- Previous work on proving eyes and safety
  - With external conditions
  - With open boundaries
- Can we use these ideas to build a better seki solver?
  - Just ideas, NOT DONE YET...
  - Proving at least seki on an open board
  - Conditions - when should we call such a solver?

Open Boundary Regions (1)

- T. Cazenave, Generation of Patterns With External Conditions for the Game of Go. Advances in Computer Games 9 (2001)
- Abstract count, or bounds, of external liberties
- Can be generated backwards from final position, similar to RZone (but more complicated)

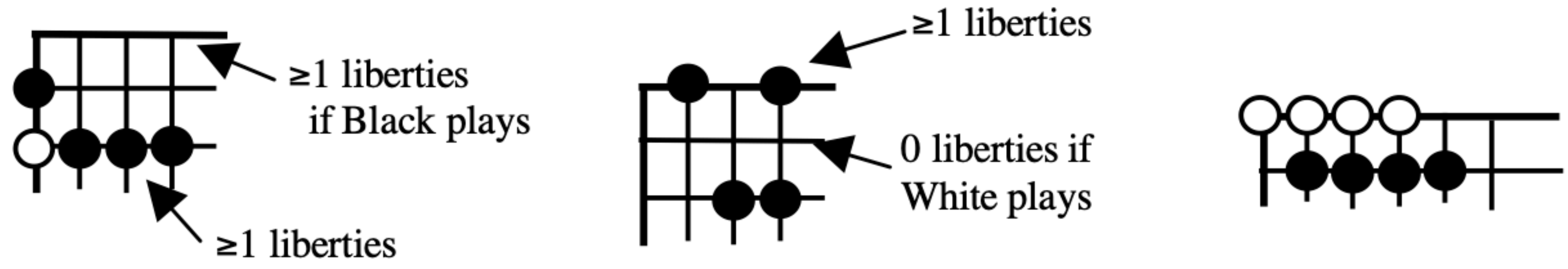
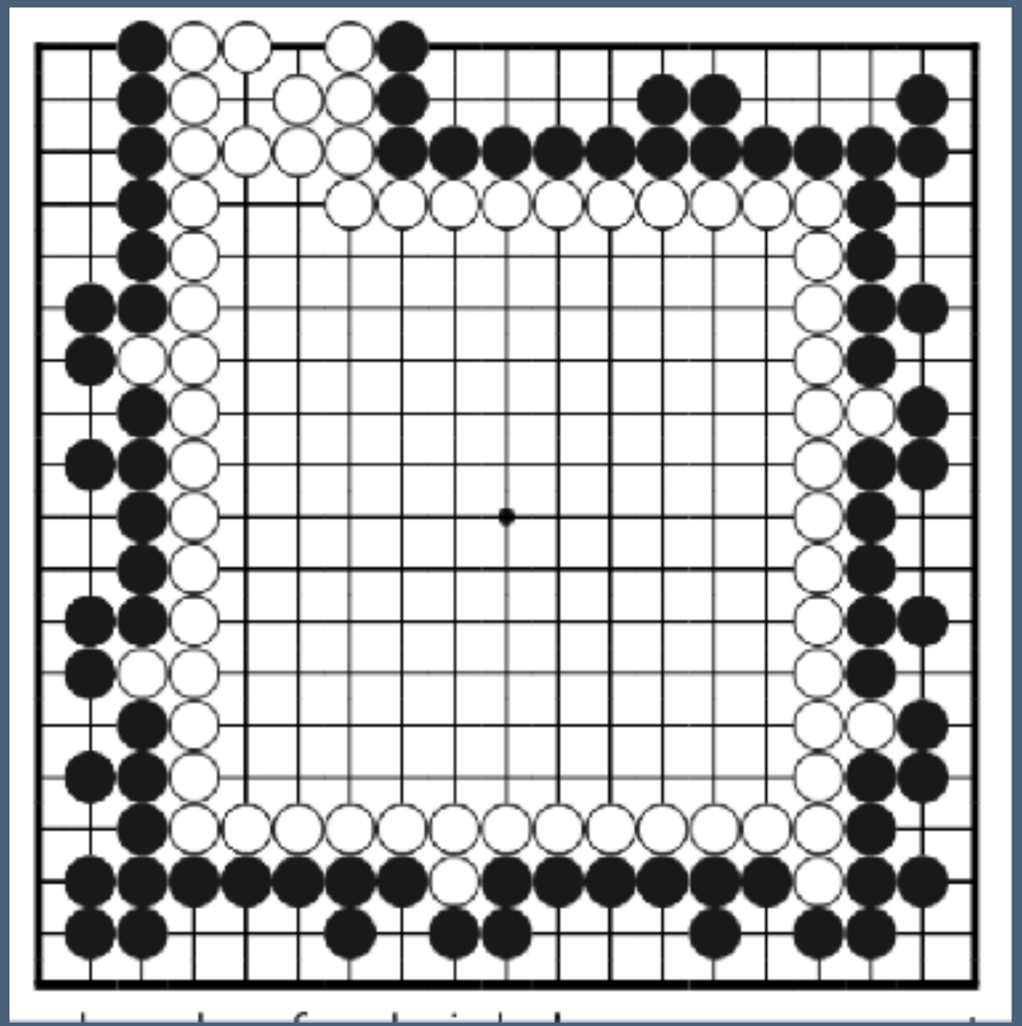


Figure 19: Rules where Black can live in one move.

Open Boundary Regions (2)

- X. Niu and M. Müller. An open boundary safety-of-territory solver for the game of Go. Computer and Games (2007)
- Use heuristic zones to divide up board
- X. Niu and M. Müller. An Improved Safety Solver in Go Using Partial Regions. Computers and Games (2008)
- Prove safety of a part of a (very) large region
- Use “miai pairs” to split off a part
- Strategy for making eyes on inside, and defend on miai pairs against outside



All of Black area proven safe, using 10 miai pairs

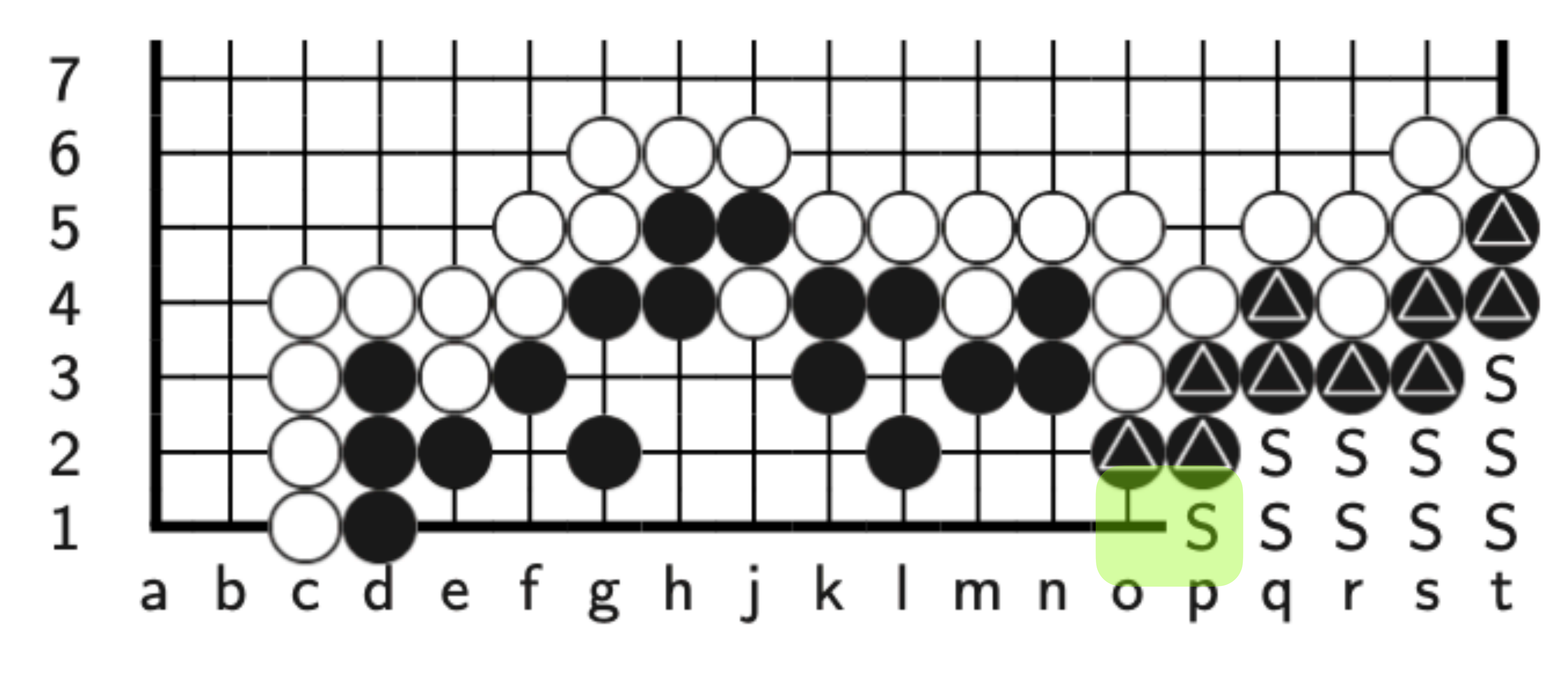
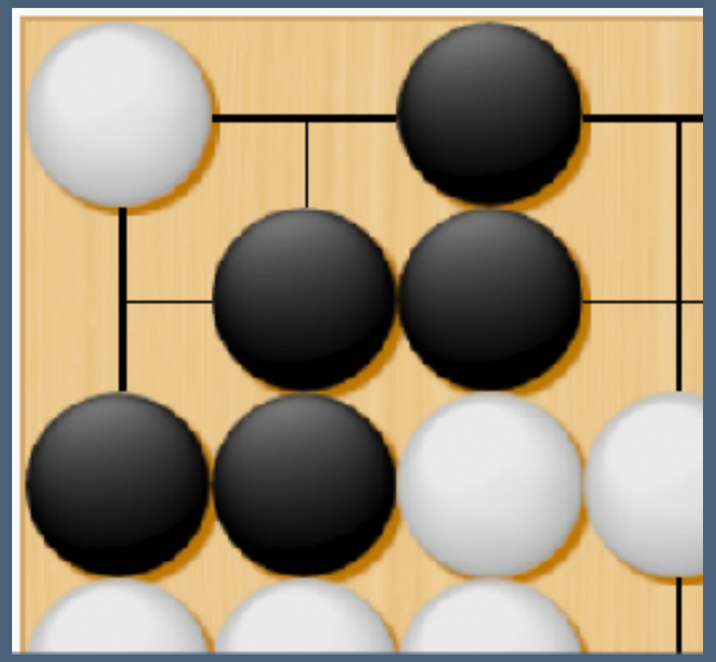


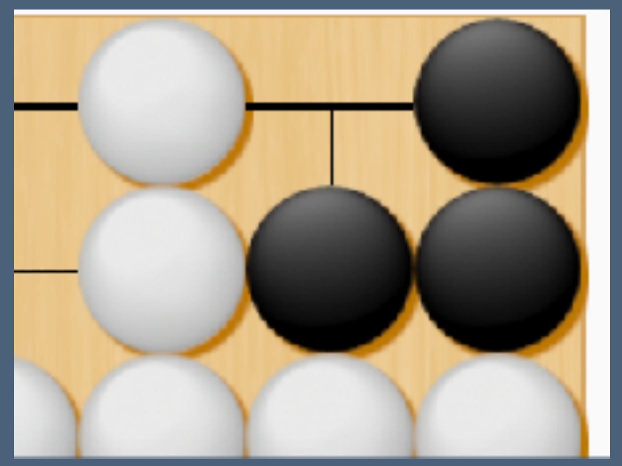
Fig. 3. Part of region is proven safe

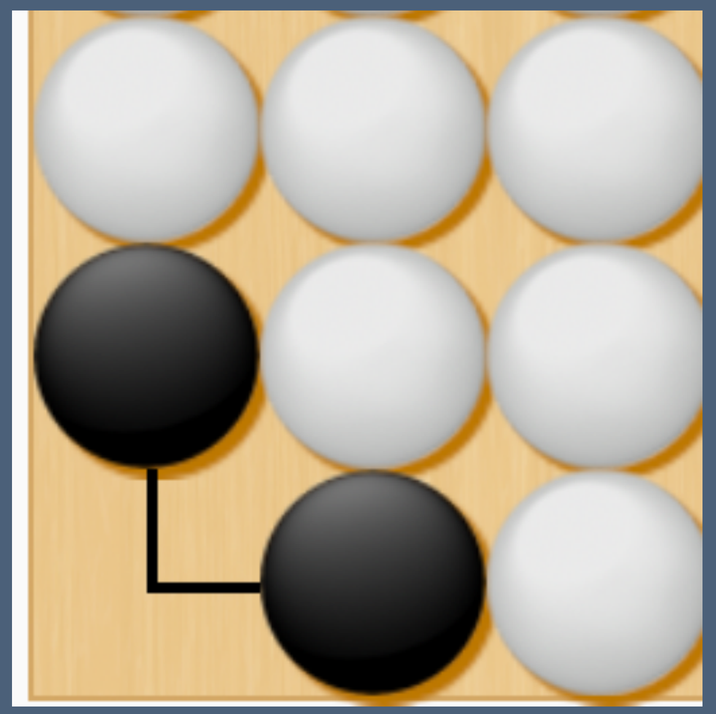
Corner S proven safe
Using miai pair {o1, p1}

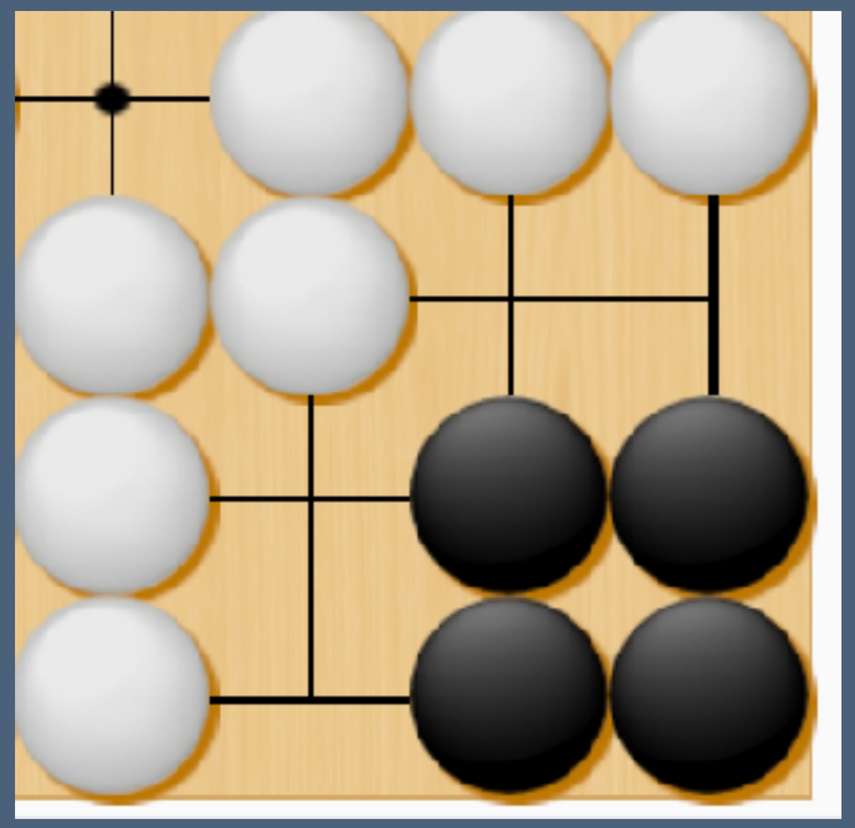
Large Database of Eyes, Alive shapes, Territories

- Generalize work on seki table, and by Vila/ Cazenave
- Include other information - one eye, two eyes, safe territory
- Learn new table entries while solving
- Challenge: extend to earlier positions by retrograde analysis?



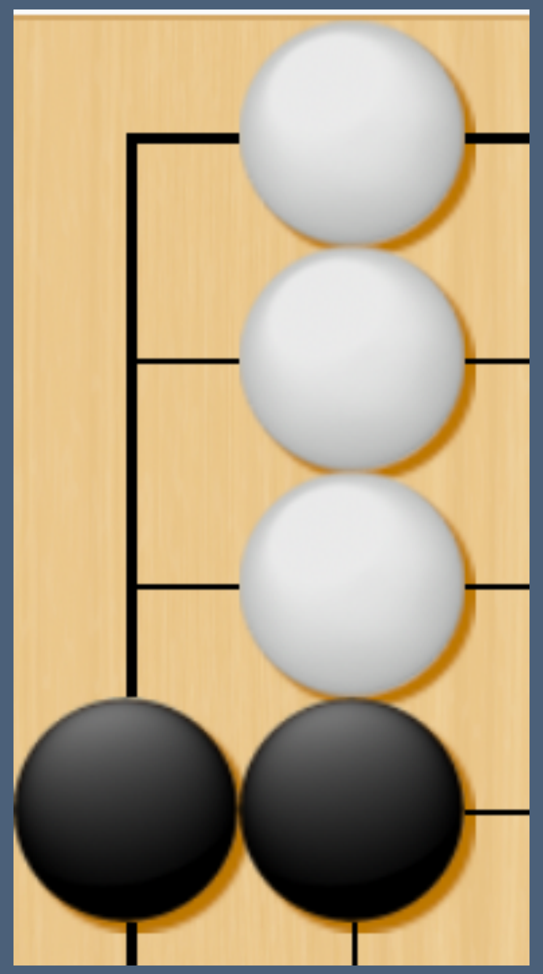


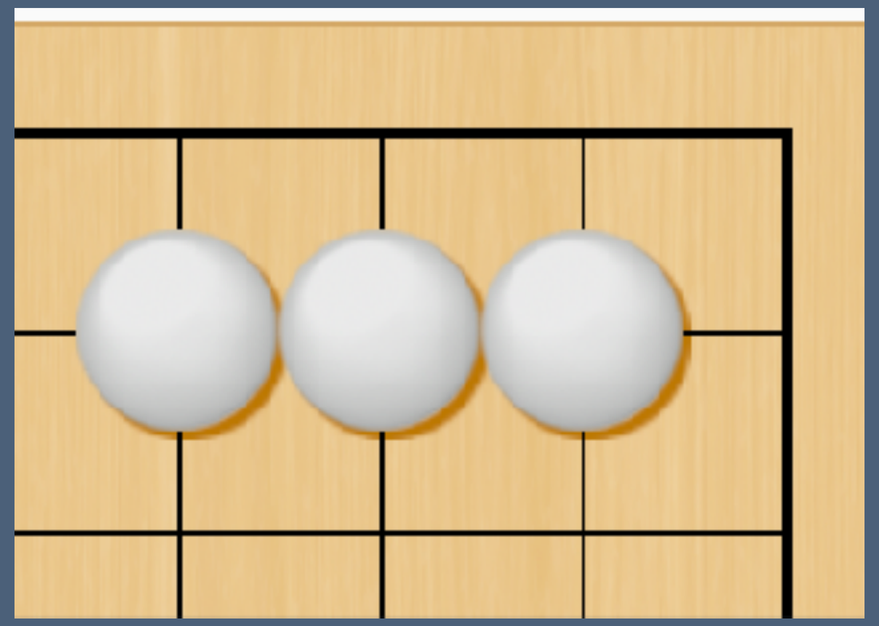


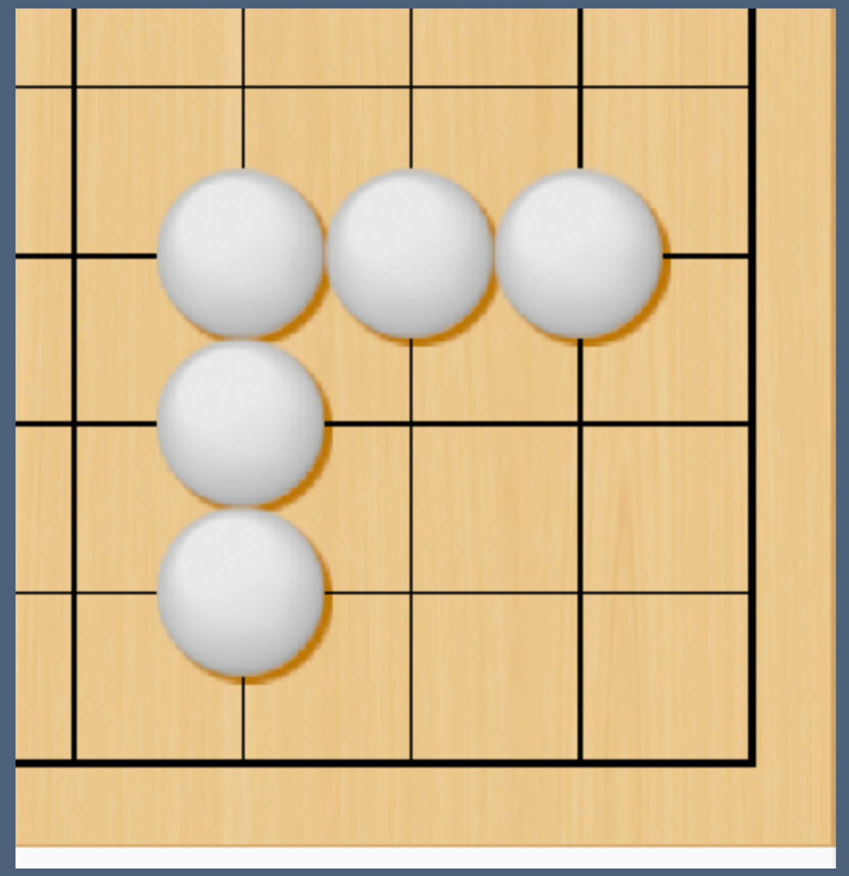


Open-boundary Region Database

- Extend database to regions that are not fully enclosed
- Store strategy that will fully enclose them
- Derive correct rules for combining such local results

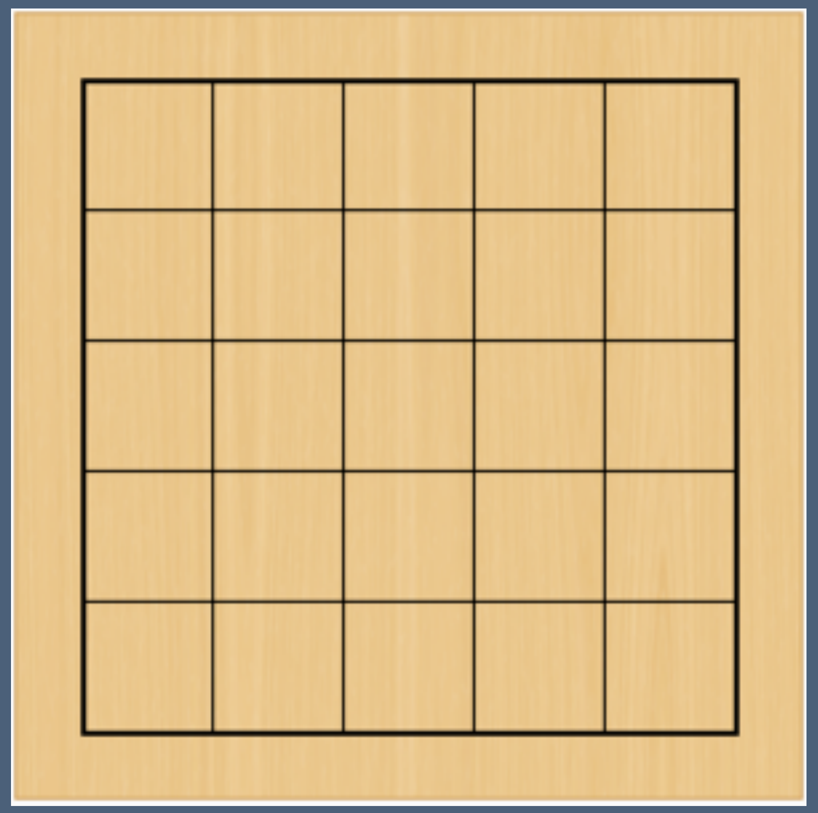




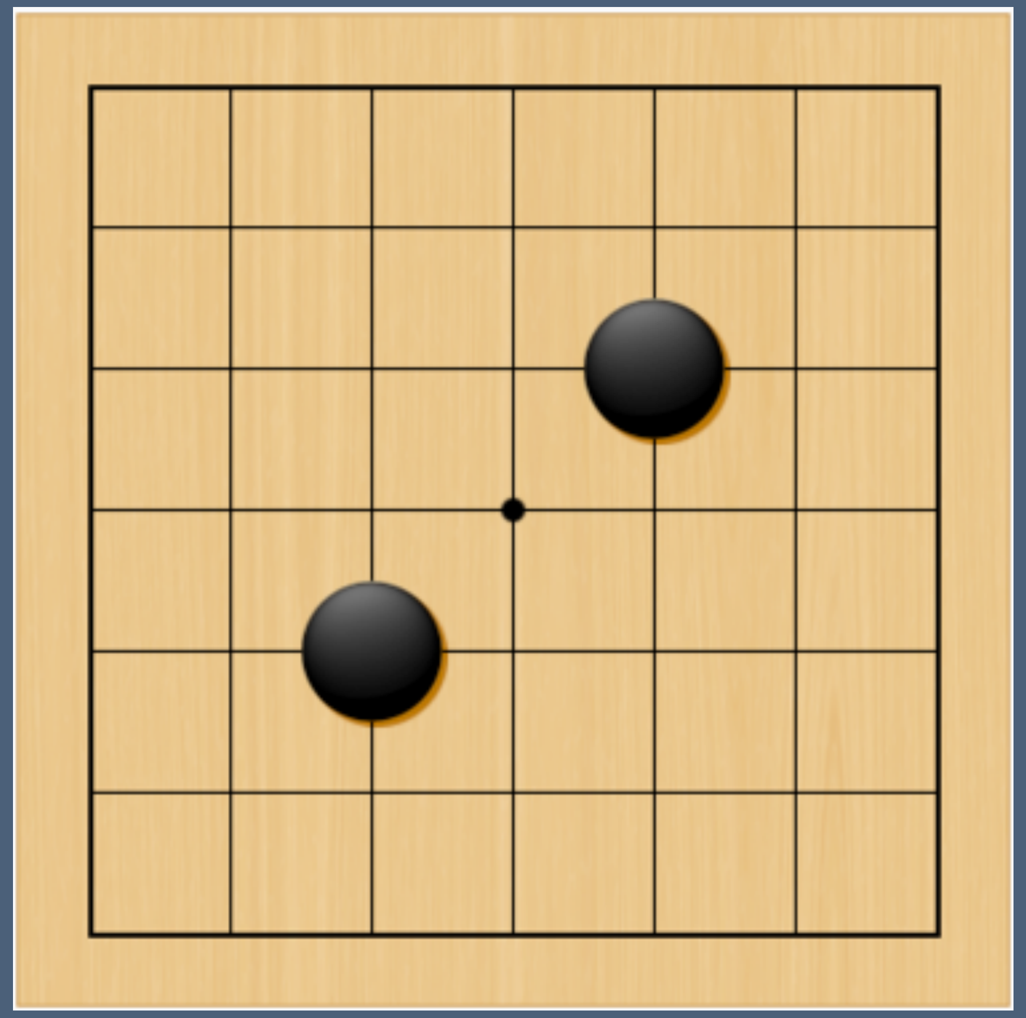


Summary

- Reviewed previous work on seki (coexistence) in Go
- Much work, mostly theoretical
- How to apply to solving small board Go?
- Developed some proposals based on recent and old work
- Challenge: seki is just one part of using exact knowledge about local shapes, patterns in Go



6x6 Go



7x7 Killall Go example