

# Generierung von Zügen in GO

Seminar:  
Knowledge Engineering und Lernen in Spielen

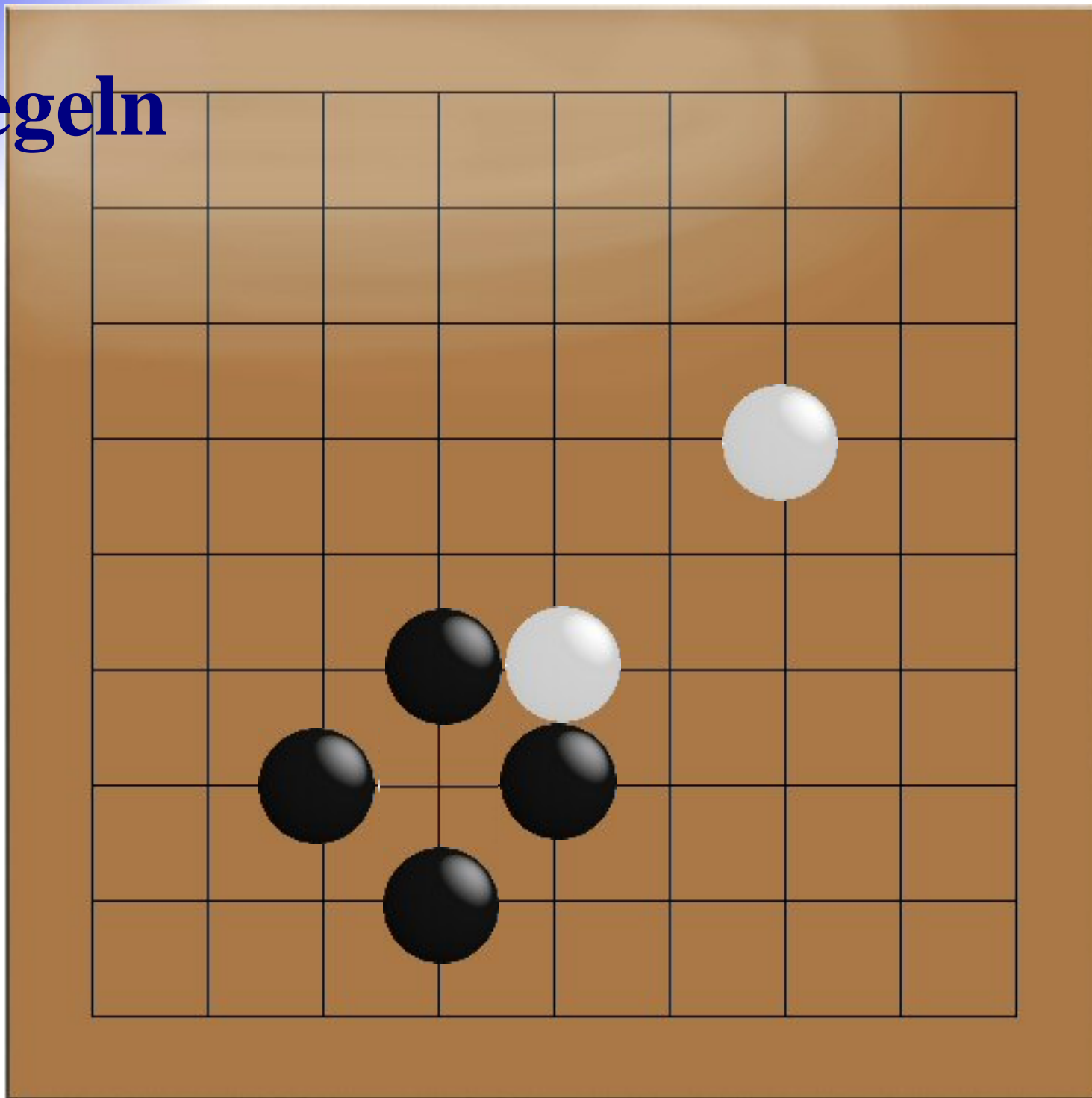
Datum:  
15.06.2004

Vortragender:  
Lars Both

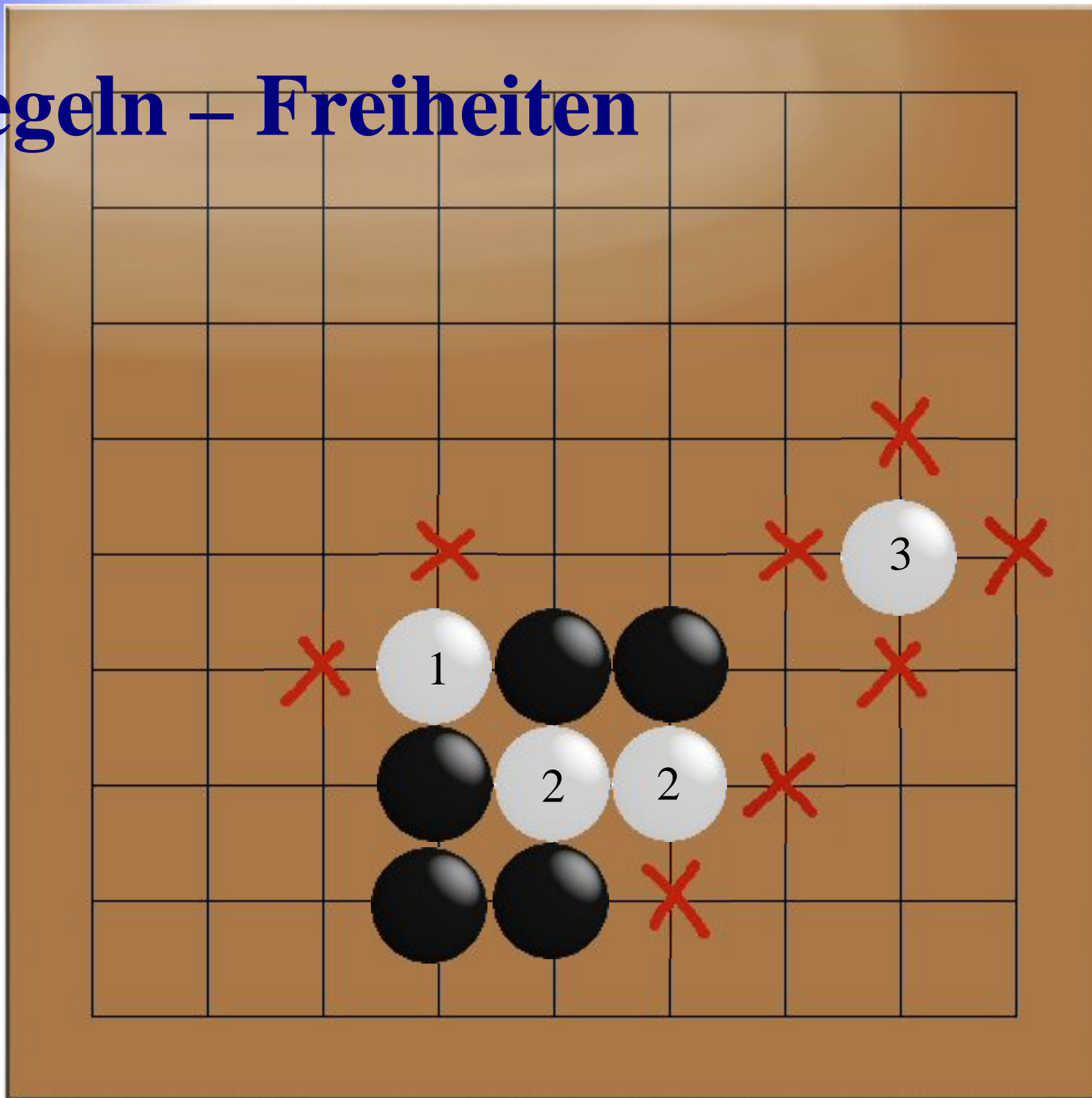
# Übersicht

- **Einführung**
  - GO - Regeln
- **Abstraktion**
  - Steine
  - Blöcke
  - Ketten
  - Gruppen
- **Evaluation**
  - Leben und Tod
  - Augen
- **Zugfindung**
  - Gruppensicherheit
  - Territorien
- **Zuggeneratoren**
  - Zuggeneratoren
  - Globale Suche
  - Zugentscheidung
- **Zusammenfassung**

# Regeln



# Regeln – Freiheiten



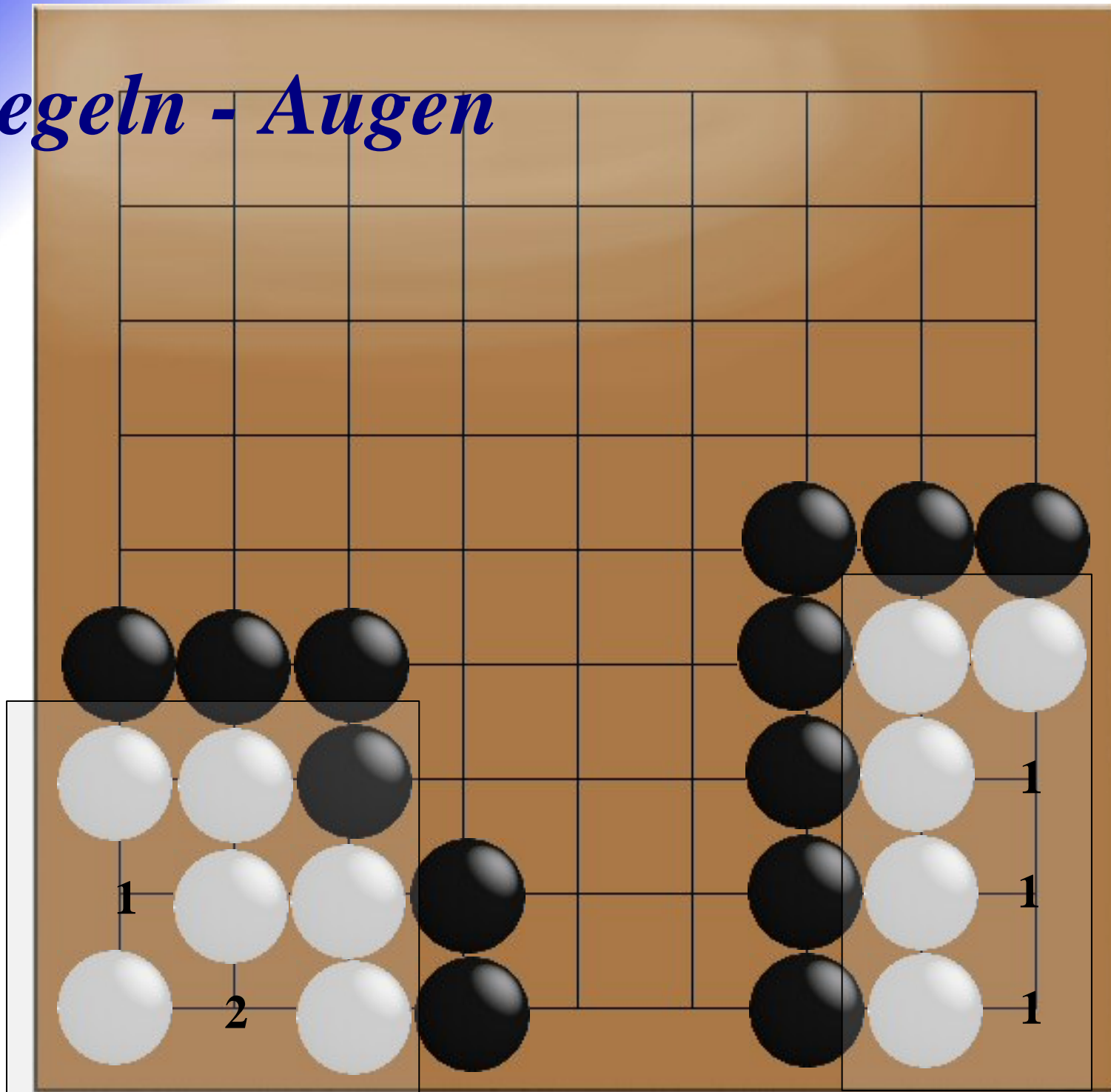
# Regeln – Verbotene Züge I



# Regeln – Verbotene Züge II



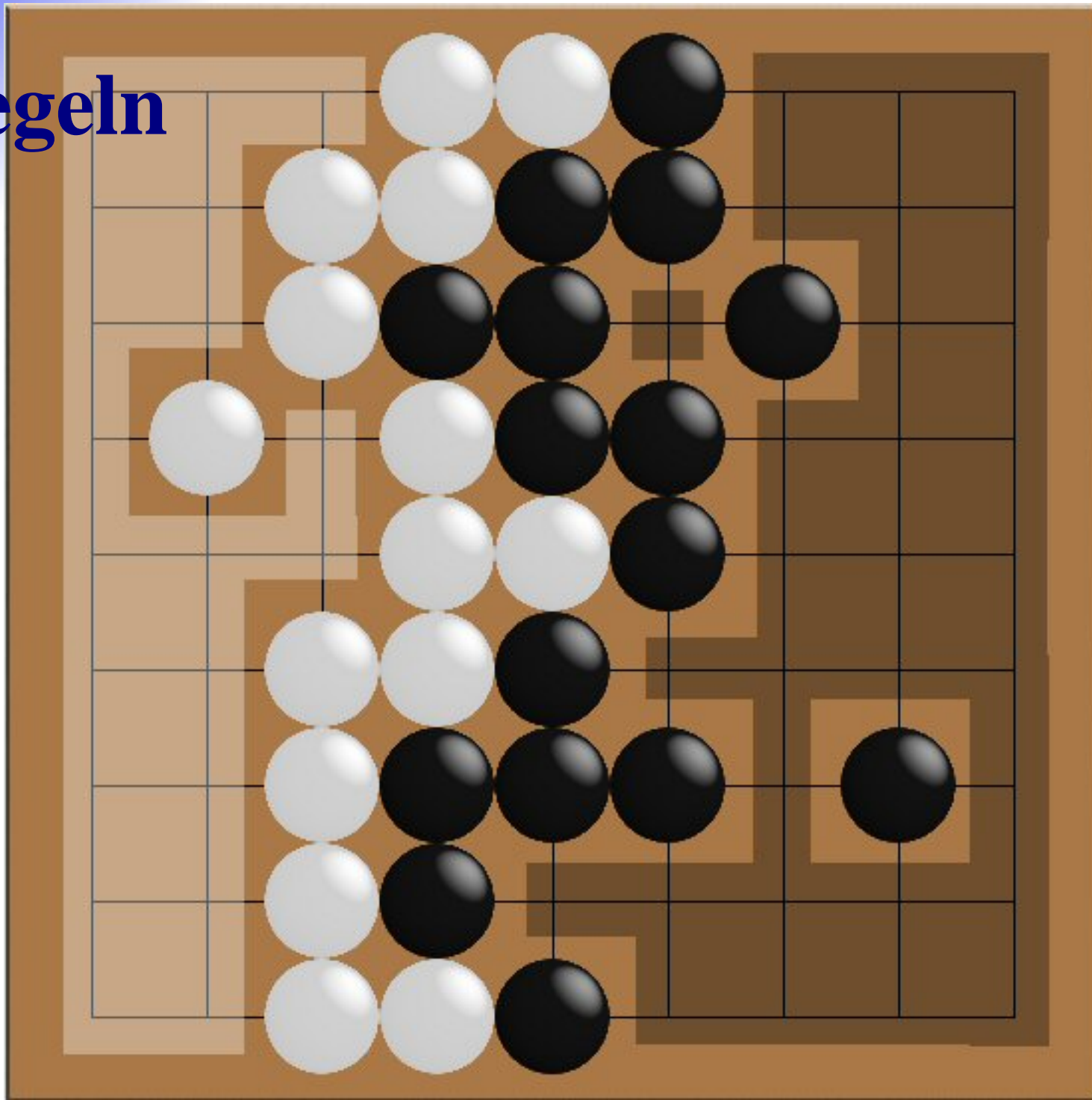
# *Regeln - Augen*







# Regeln



# Regeln – das richtige Brett



# Problematik

- Verzweigungsfaktor  $\sim 200$  pro Halbzug
  - $10^{600}$  Knoten im Suchbaum
  - „brute-force“ Suche unmöglich
- Klaffende Lücke zwischen Brettaufbau und Zugfindung
  - Statische Evaluation unmöglich
  - Hohe Exaktheit schwer zu erreichen

# Abstraktion, Ebene 1 – Steine

- Verteilung der Steine
- Eingabereihenfolge
  - Nächster Spieler
  - Vermeiden von Ko – Situationen

# Abstraktion, Ebene 2 – Blöcke

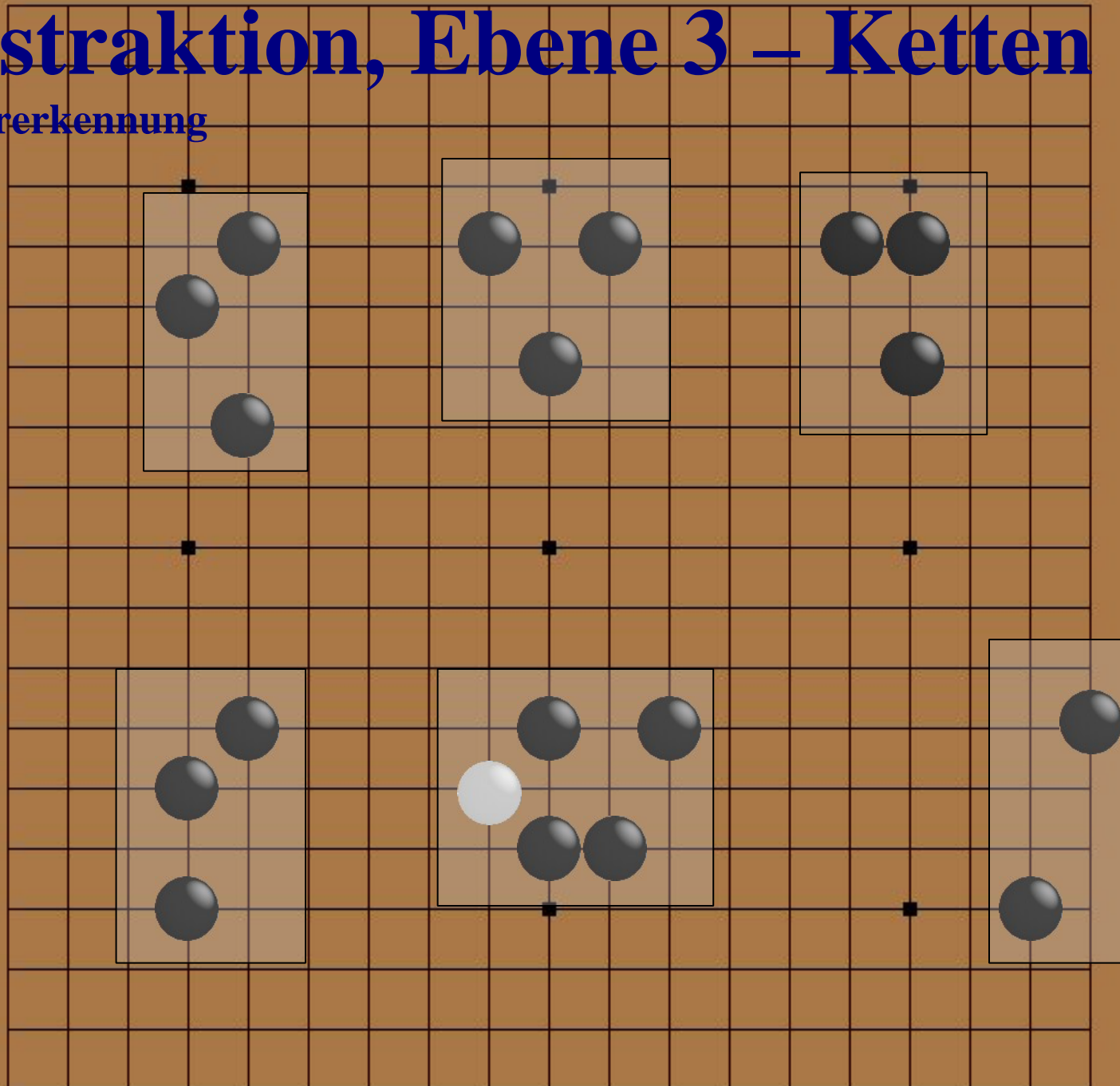
- Benachbarte Steine bilden Blöcke
- Identifikation von Blöcken
  - Einfache Graphensuche.

# Abstraktion, Ebene 3 – Ketten

- Ketten sind nicht *trennbare* Blöcke gleicher Farbe
- Identifikation von Ketten
  - Heuristik
    - Besitzen 2 Blöcke zwei oder mehr gemeinsame Freiheiten, dann gehören sie der selben Kette an.
  - Mustererkennung
    - nächste Folie.
  - Suche
    - Ziel-orientierte lokale Suche.

# Abstraktion, Ebene 3 – Ketten

Mustererkennung



# Abstraktion, Ebene 4 – Gruppen

- strategische Einheiten von Steinen
  - Ketten der eigenen Farbe
  - Tote Blöcke der gegnerischen Farbe
  - Freie Felder

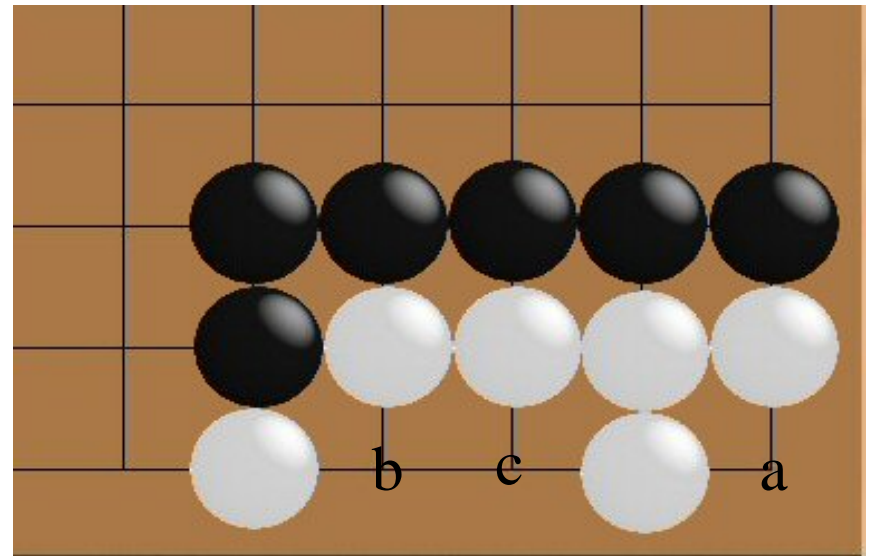


# Evaluation

- Blöcke, Gruppen - Leben und Tod
  - Blöcke
    - Zahl der Freiheiten ermitteln
    - Status wird durch lokale Suche ermittelt.
  - Gruppen
    - Augenzahl
      - Bibliotheken
      - Heuristik
    - Vorausschau
- Territorien
  - Einfluß

# Evaluation - Augen

- Klassifikation - Heuristik
  - 'full eye-point'
  - 'partial eye-point'
  - 'false eye-point'
  - Herabstufungsregel



# Evaluation - Augen

- Ermitteln der Augenzahl der Augenregionen
  - Ablesbar aus Tabelle:

Länge äussere Grenze

Augen

<= 6

1

7

1.5

8 (square-four)

1

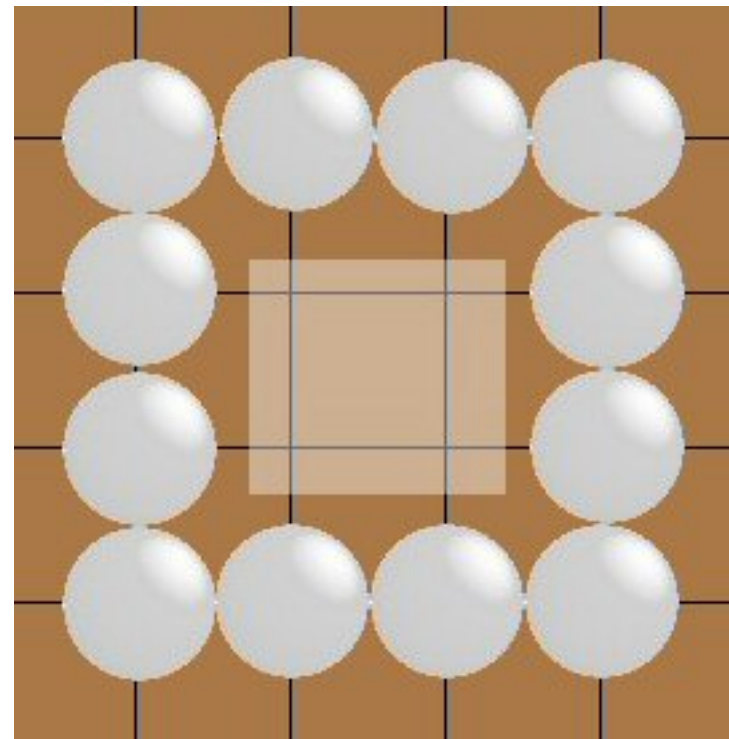
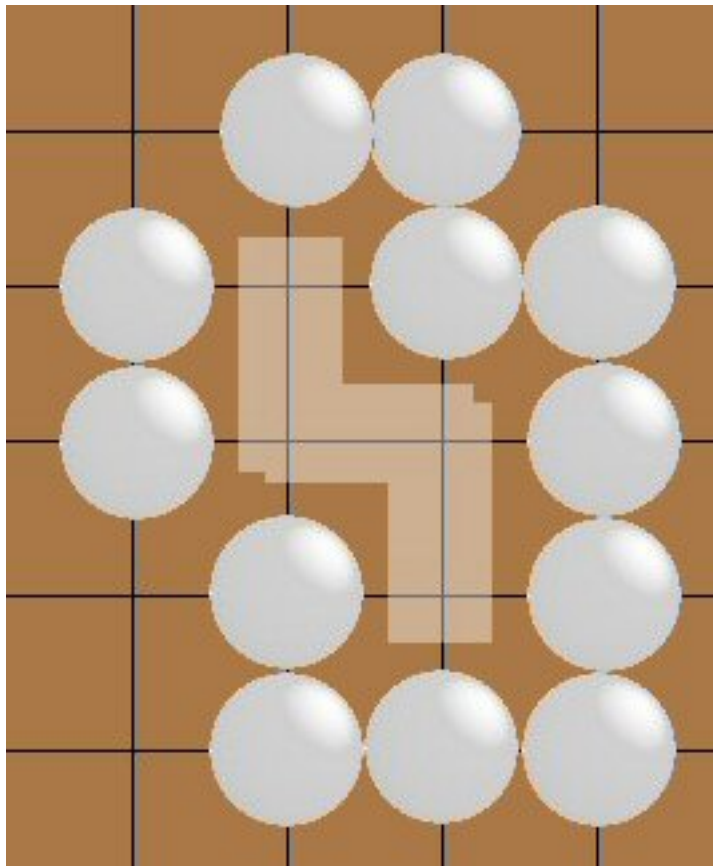
8 (curved-four)

2

8 (andere Form)

1.5

# Evaluation – Augen, Beispiel



# Evaluation - Gruppensicherheit

- Augenzahl, Potential
- Fähigkeit zur Expansion
- Fähigkeit zur Flucht
- Mit Nachbarn zu verbinden
- Sicherheit angrenzender feindlicher Gruppen

# Evaluation – Vorausschau

- Ermitteln des Status einer Gruppe
  - 2 Vorausschauen
  - Ergebnisse:  
 $\{W|W\}, \{W|?\}, \{W|L\}, \{?|W\}, \{?|?\}, \{?|L\}, \{L|W\}, \{L|?\}, \{L|L\}$ .
  - Verwertbar:  $\{L|L\}, \{W|W\}, \{W|L\}, \{L|W\}$ 
    - Gehen direkt ein in Evaluation
  - Nicht:  $\{?|?\}, \{W|?\}, \{?|L\}$ 
    - Rechenzeit verschwendet
- Teil der langsamen Evaluation von Indigo

# Evaluation - Territorien

- Territorium
  - Innere Punkte und Gefangene
  - Berücksichtigung der Gruppensicherheit
- Potentielles Territorium (Moyo)
  - „Lebende“ Steine üben Einfluß
  - Abschätzung
  - Akkurate Evaluation hat Komplexität vom Spiel

# Zuggeneratoren

- Nur plausible Züge
- GO Intellect hat ~ 20 Generatoren
  - Zielorientiert
    - *ProtectGroup*
    - *EdgeExtension*
    - ..
  - Zugfindung durch..
    - .. Mustererkennung
    - .. lokale Suche



# Zuggeneratoren – lokale Suchen

- Auch zum Analysieren der Brettkonfiguration
  - „capturing search“
  - „semeai search“
  - „connection search“
  - „life-and-death search“
  - „territory-surrounding search“
- Heuristik, Evaluation jeweils unterschiedlich

# Globale Suche

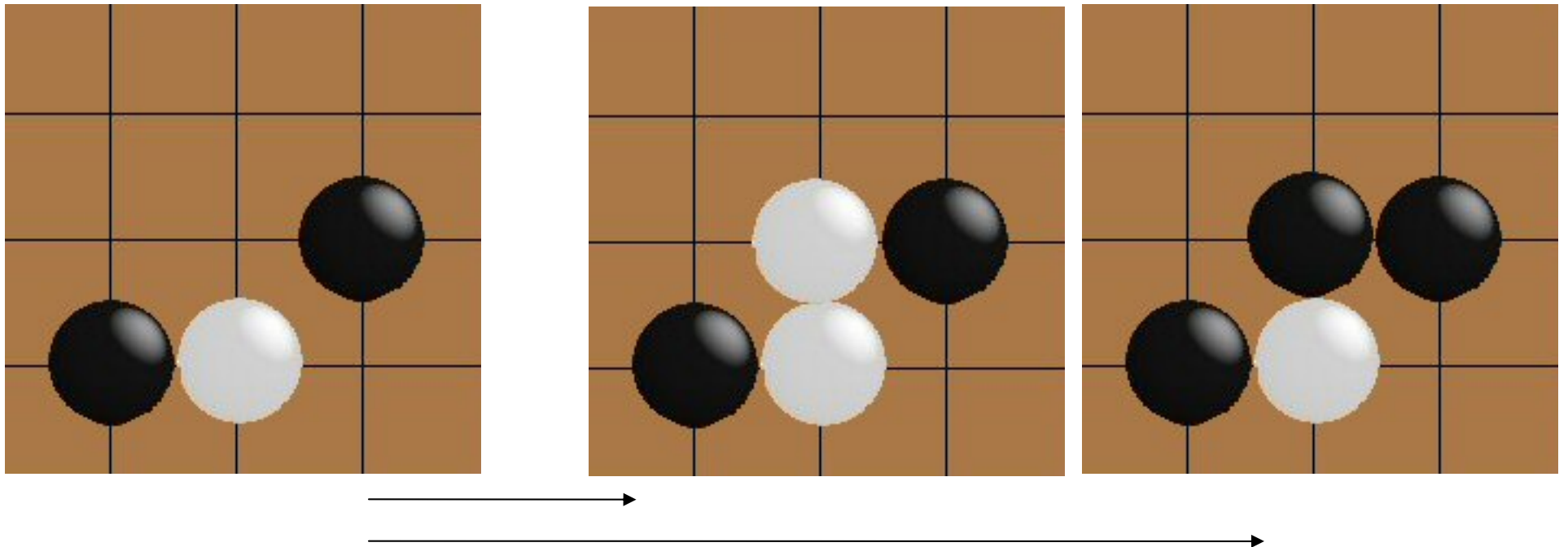
- Zugkandidaten werden gewichtet
  - GO Intellect: Zugwert, Dringlichkeit
  - GNU GO: Zugbegründungen
- Nur Top-Züge werden ausprobiert
  - < 10 Züge
  - Nur Züge mit Wert größer als Durchschnitt (GO Intellect)

# Globale Suche – GO Intellect

- Topzüge ausprobieren
  - Minimal 2 Halbzüge
  - Nur Evaluation stabiler Knoten
    - Stabil
    - Sonst weiter expandieren
    - Bis zu maximaler Suchtiefe
  - Evaluation + Dringlichkeit legen Zug fest

# Mittelspiel - Indigo

- Baumsuche benötigt akkurate Evaluation
  - schwierig zu finden

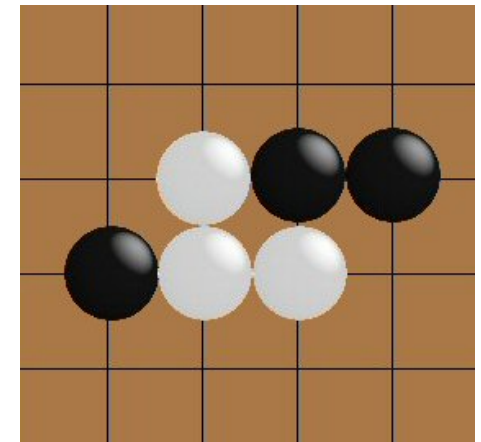
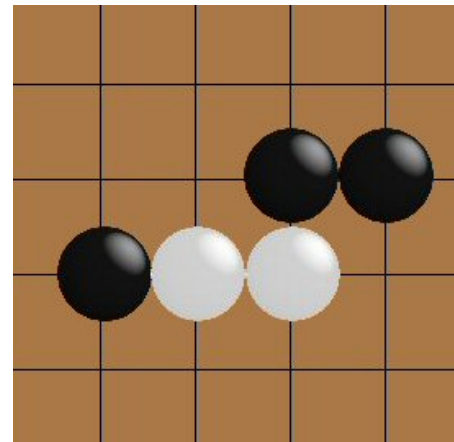
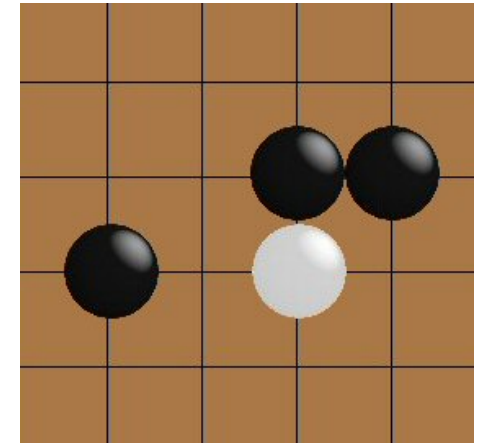
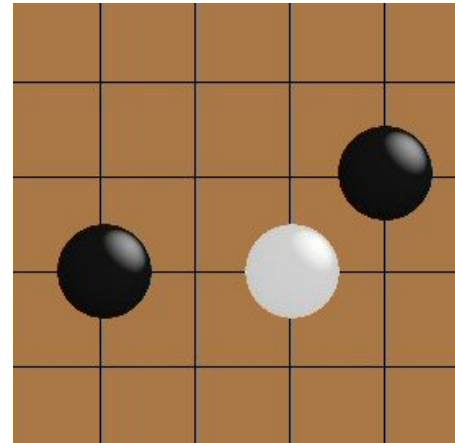


# Mittelspiel - Indigo

- Indigo benutzt Schablonen um..
  - .. gegnerische Steine zu trennen.
  - .. eigene Steine zu verbinden.
- Adäquate Modellierung mit „8-Connections“
- Schablonen haben assoziierte Dringlichkeiten

# Zielstabilität

- Ziel: Verbinden der 2 Steine
- Konter von Weiß
- Schwarz spielt auf neues Ziel
- Weiß trennt die schwarzen Blöcke
- Ansatz in INDIGO  
verdoppeln der Dringlichkeit



# Zugentscheidung - Indigo

- 3 Methoden zur Zuggenerierung
  - „urgent“
    - *8-Connections*
    - Verifikation mit Suche und schneller Evaluation
  - „calm“
    - selektive Suche mit schneller Evaluation
  - „life-and-death“
    - Langsame Evaluation (Suche)
- Darüberliegende Ebene entscheidet

# Zusammenfassung

- Evaluation eines GO-Spiels
- Zuggeneratoren
  - Expertenwissen
  - Zielorientiert
- Zugfindung beispielhaft für:
  - Indigo
  - GO Intellect



# Quellen

- K.-H. Chen: Some Practical Techniques for Global Search in Go. International Computer Games Journal 23(2), 2000.
- K.-H. Chen: Computer Go: Knowledge, Search, and Move Decision. International Computer Games Journal 24(4), 2001.
  - <http://www.epp.infonomics.nl/FdAW/chen/go.htm>
- B. Bouzy: The Move-Decision Strategy of INDIGO. International Computer Games Journal 26(1):14-27, 2003.
  - <http://www.math-info.univ-paris5.fr/~bouzy/publications/MyBouzy-ICGAJournal.pdf>
- GNU GO
  - <http://www.gnu.org/software/gnugo/devel.html>