

Projecte de final de carrera – Enginyeria informàtica.

Gorgon: IA per a Tsumegos en el joc del Go.

Autor: Josep Floriach Ventosinos

Director: Ramon Grau Sala

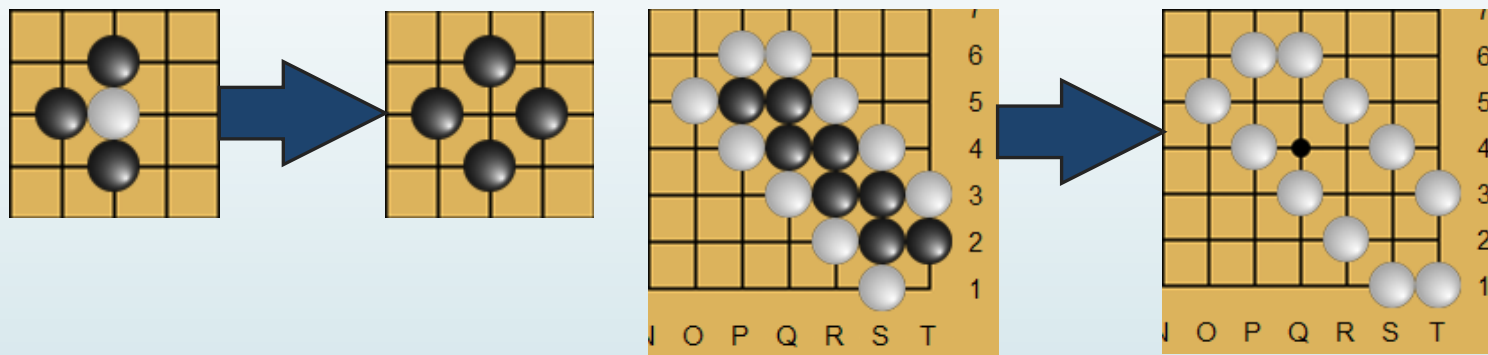


Índex de continguts

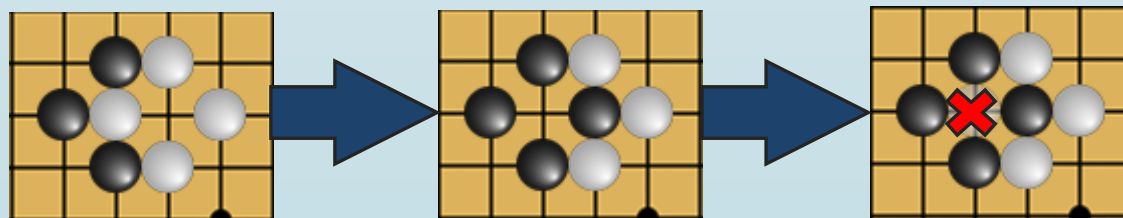
- Introducció al Go: Regles del Go
- Tsumegos
- Estat de l' art
- Objectius del projecte
- Implementació
 - GoGUI
 - Estructures i conceptes bàsics
 - Intel·ligència artificial
- Resultats
- Conclusions
- Treball futur

Introducció: Regles del Go

- Taulell de 19x19 interseccions. En el seu torn, cada jugador col·loca una pedra del seu color a qualsevol intersecció.
- Regla de les llibertats. No es permet el suïcidi.

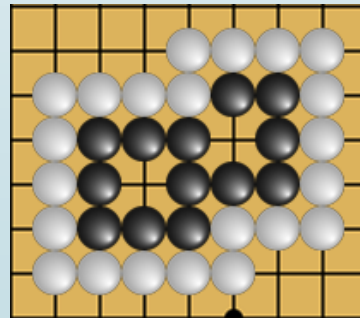


- Regla del KO



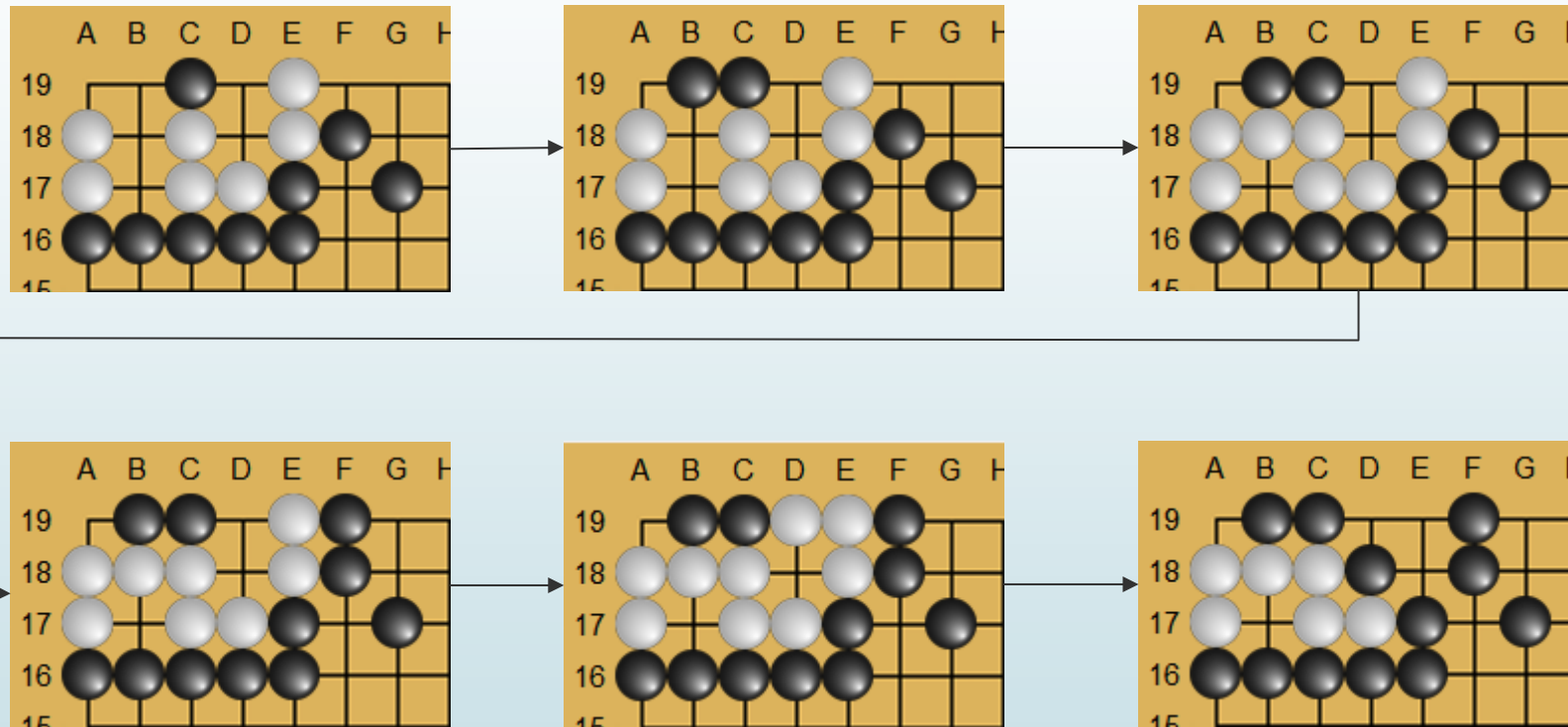
Tsumegos

- **Tsumego**: problema similar als problemes d'escacs del diari, però adaptat al joc del Go.
- L'objectiu és aconseguir o impedir que un grup determinat de pedres visqui en un territori concret.
- Ajuden al desenvolupament del jugador en una partida completa.



Dos ulls

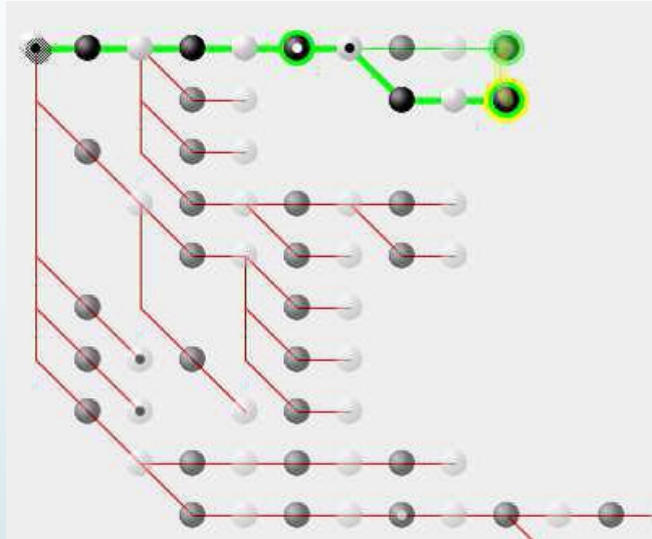
Exemple de Tsumego



Negres guanyen

Estat de l'art

- La majoria d'eines es basen en arbres de jugades.



- **Problema:** no és viable tindre en compte totes les possibilitats, inclús en una zona de joc relativament petita.



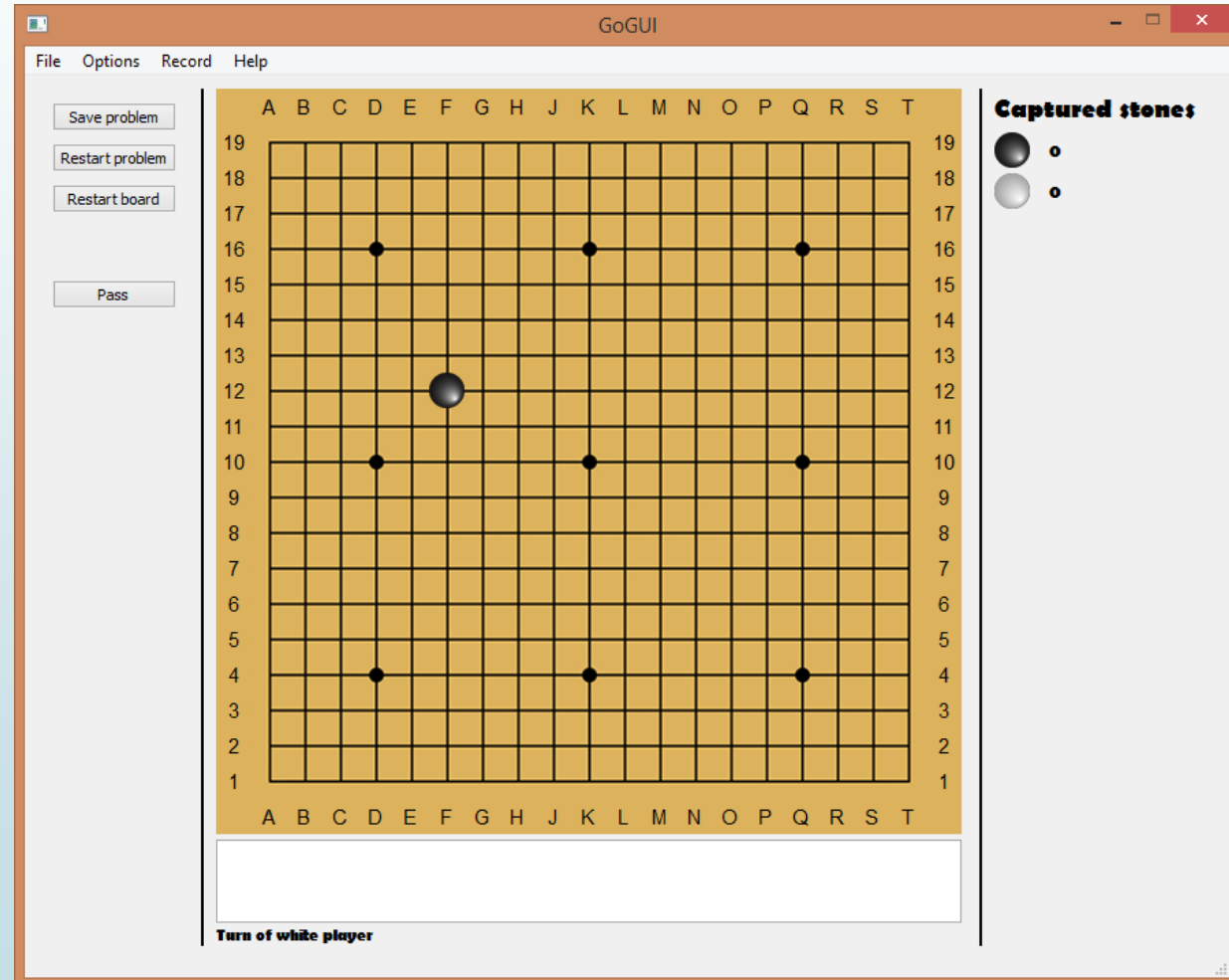
Objectius del projecte

- **Solució:** Implementar una IA que sigui capaç de jugar en aquests Tsumegos sense cap tipus de coneixement previ.
- **Avantatges:**
 - El jugador podrà veure les conseqüències de les seves jugades fins al final.
 - Els usuaris que creïn els problemes només hauran de definir la disposició de les peces (sense cap tipus de feina addicional).
- **Objectius:**
 - Ajudar al jugador novell a entendre aquests problemes, fent que millori les seves capacitats d'anàlisi del joc, i capacitats estratègiques i tàctiques.

Implementació

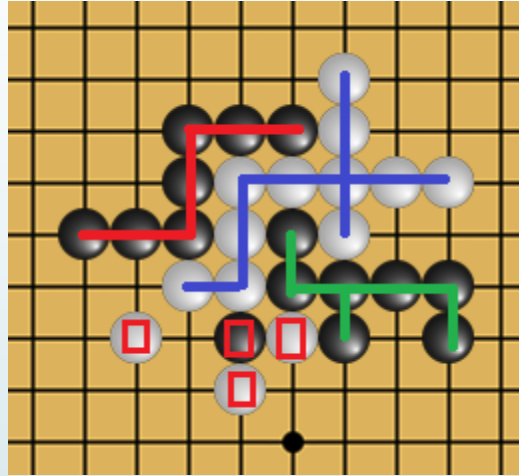
Eines de desenvolupament: C++/Qt framework

GoGUI:

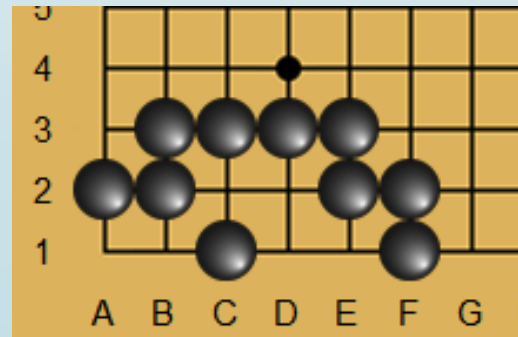
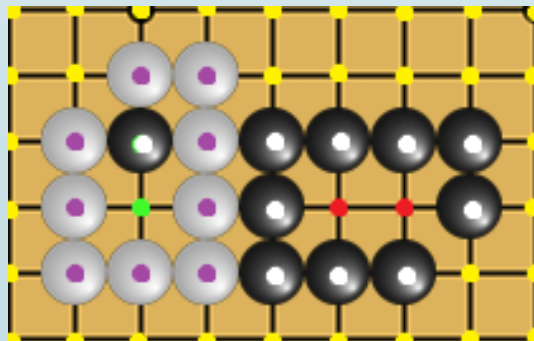


Estructures i conceptes bàsics (I)

Serps: cadena de pedres del mateix color, connectades entre sí.

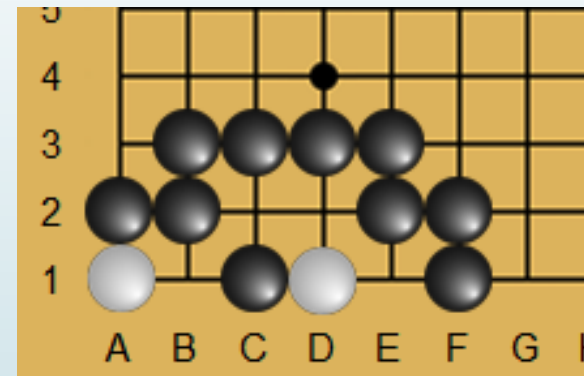
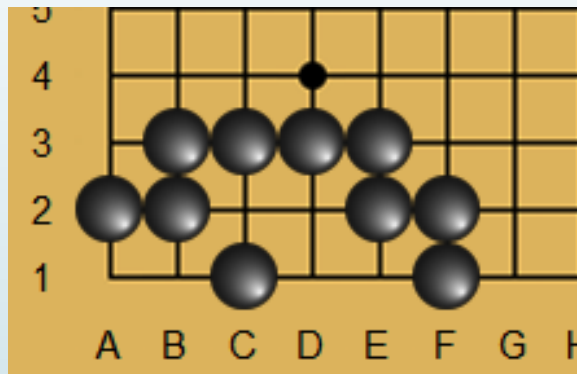


Tàrtars: regions adjacents a una serp o conjunt de serps.



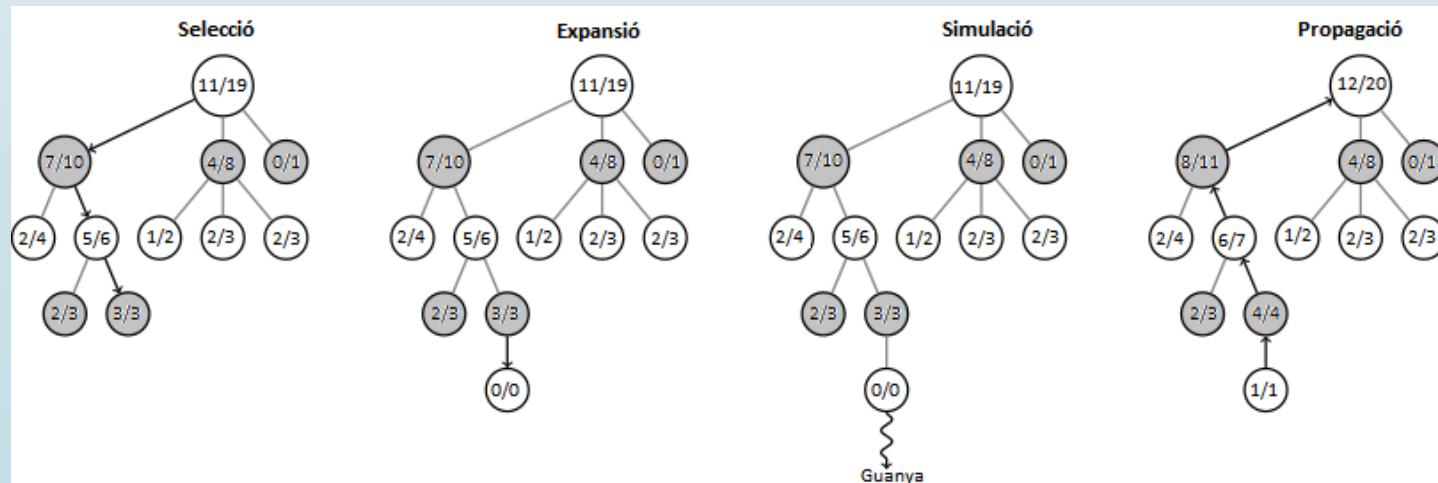
Estructures i conceptes bàsics (II)

- **Tàrtar vital a una serp:** totes les interseccions buides del Tàrtar són llibertats de la serp.



IA: Monte-Carlo Tree Search (I)

- Basat en provar moltes partides aleatòries fins que s'acabi el temps, i escollir com a moviment immediat el que porti a més partides guanyadores.
- Construeix un arbre on va guardant la relació de partides jugades/guanyades a cada node.
- 4 fases:
 - **Selecció:** selecciona els moviments amb millors resultats fins al moment, fins que trobi un node que no ha provat cap moviment.
 - **Expansió:** es crea un node a partir del node seleccionat.
 - **Simulació:** juga moviments aleatoris fins a final de partida.
 - **Propagació:** els resultats de la simulació s'expandeixen fins al node arrel.



IA: Monte-Carlo Tree Search (II)

► Dos problemas bàsics en la fase de selecció.

- El rati de victòries d' un node sense cap partida jugada és 0/0 (NaN).
 - SOLUCIÓ: abans d' aprofundir en l' arbre, sempre seleccionarem un node amb 0 partides jugades.
- Degut a l' aleatorietat del mètode, es podria donar que un node amb un percentatge del 99% de victòries, quedés descartat a la primera simulació, i per tant, no tornar a ser explorat.
 - SOLUCIÓ: buscar un bon equilibri entre exploració i explotació de l' arbre.
 - Upper Confident Bounds - 1 (UCB-1)

$$\frac{W_i}{n_i} + C \sqrt{\frac{\ln(t)}{n_i}}$$

On,

W_i es el nombre de partides guanyades a traves del node.

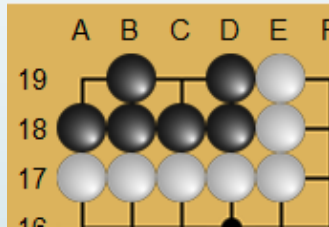
n_i es el nombre de partides jugades a traves del node.

C es una constant escollida empiricament.

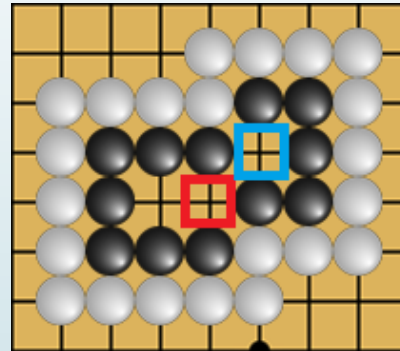
t es el nombre total de partides jugades fins al node.

IA: final de partida en la fase de simulació (I)

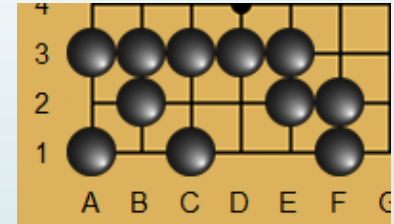
- Si algun dels dos jugadors ha fet una formació de dos ulls, o ha capturat més de 4 pedres enemigues (paràmetre variable), la simulació acaba.
- **Vida incondicional segons Benson:** una pedra o conjunt de pedres no pot ser capturada per moltes jugades seguides que faci l'enemic.



Incondicionalment
viva



No incondicionalment
vives



No incondicionalment
vives

- **Algorisme de Benson:** algorisme per a determinar metòdicament quines serps estan incondicionalment vives.
- Utilitza dos conceptes encara no mencionats.

IA: final de partida en la fase de simulació (II)

► Algorisme de Benson

S = Conjunt de serps.

R = Conjunt de Tàrtars que encerclen a les serps.

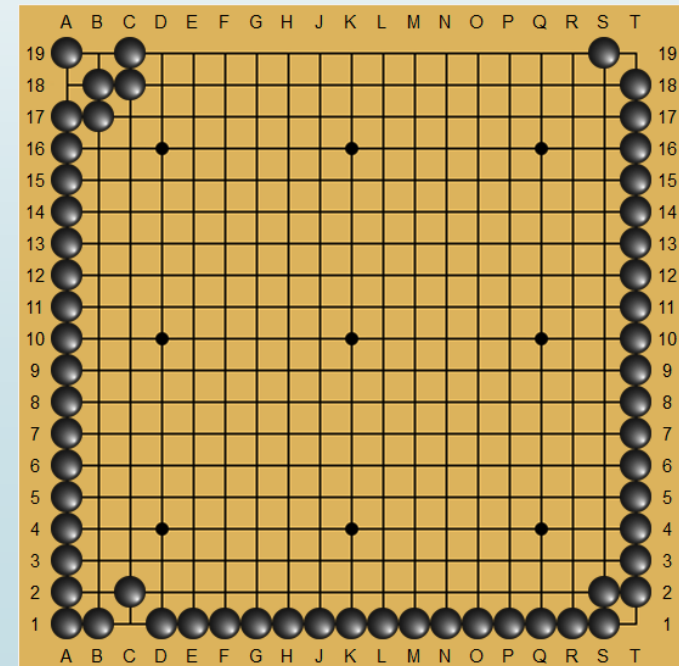
Repetir

Borrar de S totes les serps que tinguin menys de dos Tàrtars vitals a R.

Borrar de R tots els Tàrtars rodejats per alguna serp que no està a S.

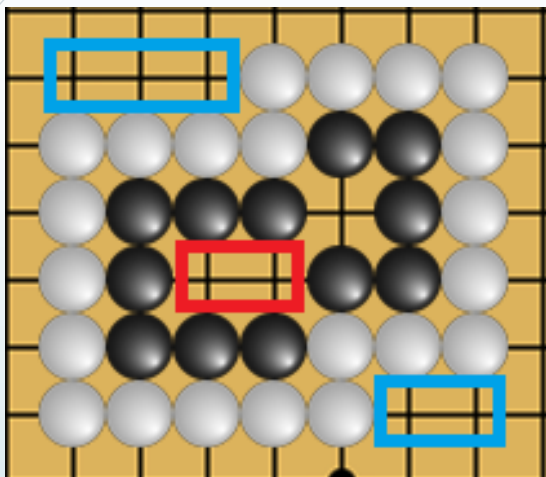
Mentre s'hagi esborrat algun element de S o R.

S = conjunt de serps incondicionalment vives.



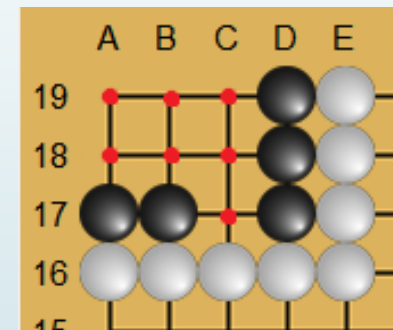
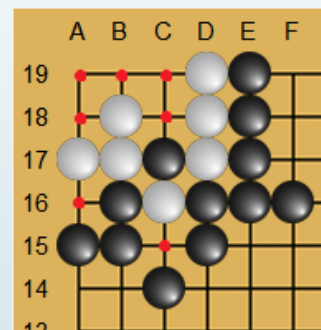
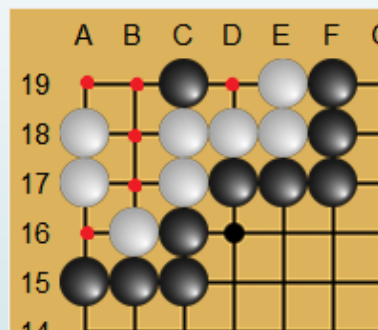
IA: millores al MCTS (I)

- MCTS no sap quina és l'àrea del taulell que forma part del Tsumego en sí.

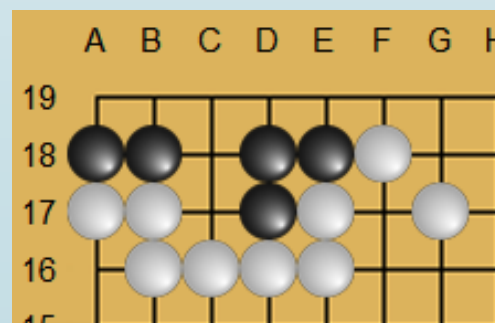
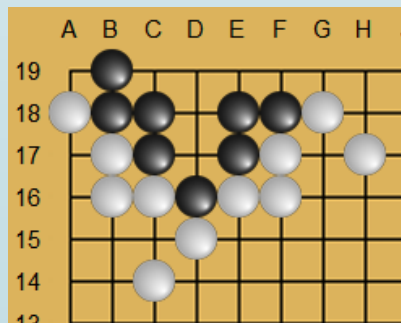


- Generació de moviments sense cap sentit.

- **SOLUCIÓ:** simular només sobre les interseccions buides que NO formen part del Tàrtar exterior.

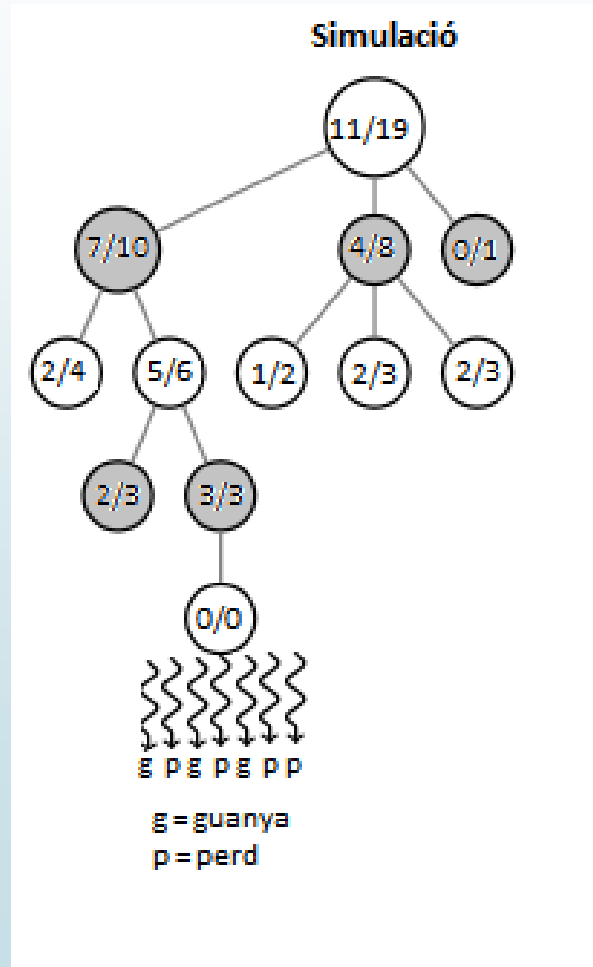


- **LIMITACIÓ:** el Tsumego ha de tindre com a mínim dos Tàrtars: l'àrea del Tsumego, i el Tàrtar exterior.



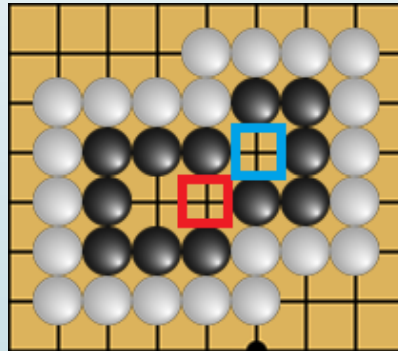
IA: millores al MCTS (II)

- MCTS paral·lelitzat mitjançant OpenMP.



IA: millores al MCTS (III)

- ▶ No fer simulacions sobre interseccions que són ulls potencials per al propi jugador.
 - ▶ Interseccions completament rodejades de pedres aliades que tenen més d'una llibertat.
 - ▶ Algorisme poc precís, però molt poc costós computacionalment.



Resultats

IAN = Juga IA amb color negre.

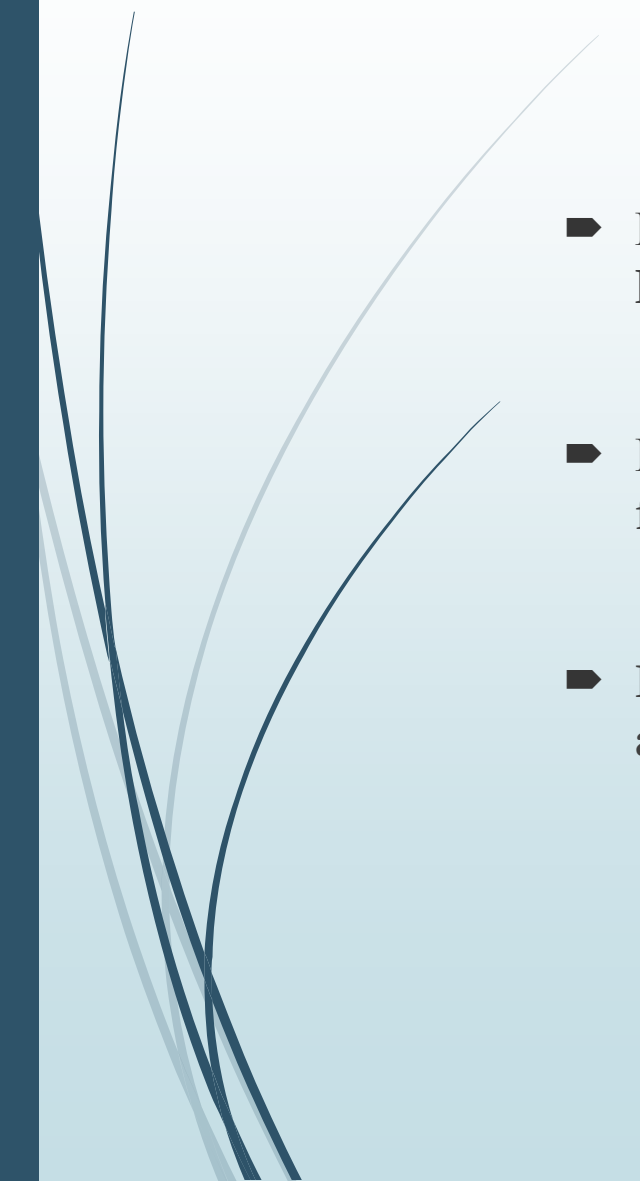
IAB = Juga IA amb color blanc.

Conjunt de test amb 20 Tsumegos.

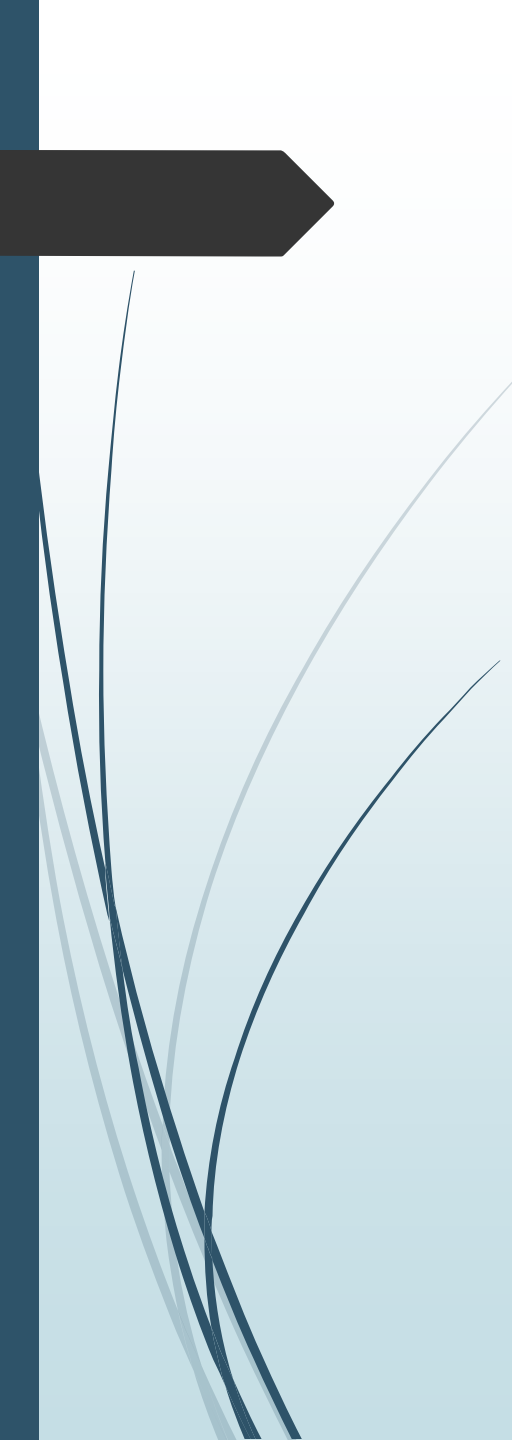
Nº partides simulades	IAN	IAB	Càlcul seqüencial	Càlcul amb OpenMP
50	10%	20%	3 segons	< 1 segon
500	75%	80%	8 segons	< 1 segon
1000	65%	70%	18 segons	1 segon
2000	60%	70%	39 segons	2 segons
5000	55%	65%	1.5 minuts	4 segons
10000	55%	60%	3 minuts	7 segons
20000	45%	55%	> 5 minuts	10 segons



Conclusions



- Per a uns millors resultats, hem de trobar un equilibri entre la precisió de la cerca d'ulls en la fase de simulació, i el seu cost computacional.
- La forma més bàsica de MCTS és molt limitada. Per a cada projecte en concret s'haurà de fer diverses ampliacions per optimitzar els resultats.
- En el cas del Go, les heurístiques pròpies del joc juguen un paper extremadament important a l'hora d'obtenir bons resultats.



Treball futur

► GoGUI

- Poder modificar els parametres de l' IA.
- Grabar partides i reproduir-les
- Mode d' edició de Tsumegos.
- Desfer jugades.

► Gorgon

- Estudiar diverses heurístiques pròpies del joc.
 - Mètode alternatiu per trobar els punts vitals.
 - Mètode alternatiu per a la detecció d' ulls.
- Implementar RAVE (Rapid Action Value Estimation).
- IA a partides complertes.

Projecte de final de carrera – Enginyeria informàtica.

Gorgon: IA per a Tsumegos en el joc del Go.

Autor: Josep Floriach Ventosinos

Director: Ramon Grau Sala