Alma Mater Studiorum – Università di Bologna

Tablut Challenge

Gruppo MAC
Mattia Moffa, Luca Andreetti, Valentino Cavallotti

Macblut

- Macblut è il giocatore che abbiamo implementato per questa challenge.
- Si basa parzialmente sull'implementazione Tablut in Java fornita dal Prof. Galassi, di cui utilizza, per esempio, le classi `State` e `TablutClient`.
- Il game engine è stato reimplementato da zero in modo che contenesse solo le funzionalità necessarie al nostro giocatore.

Strategia di ricerca

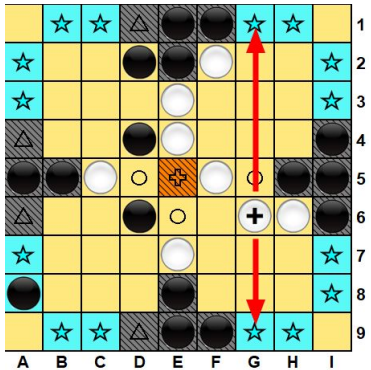
- La soluzione usa una ricerca nello spazio degli stati, in particolare l'implementazione dell'algoritmo minimax fornita da AIMA.
- Opportune modifiche sono state apportate alla strategia di ricerca ereditando dalla classe `IterativeDeepeningAlphaBetaSearch`.
- La profondità massima è determinata dal tempo a disposizione del giocatore (dinamico).

Euristiche

- Due euristiche, una per i turni del giocatore bianco ed una per i turni del giocatore nero.
- Le euristiche implementate per gli stati di cutoff si basano sui seguenti parametri:
  - numero di pezzi di ogni giocatore,
  - numero di pezzi vulnerabili,
  - posizione del re in relazione alle fughe e al castello,
  - posizione strategica delle pedine del nero (*rombo speciale*),
  - vicinanza media delle pedine nemiche al re.

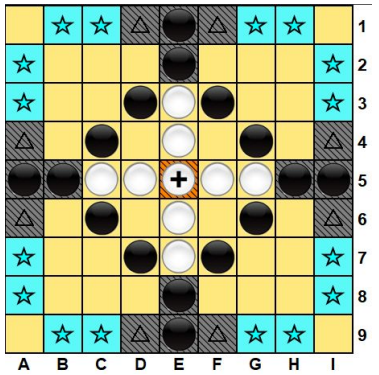
Euristica del giocatore bianco

- Percentuale di pedine bianche non ancora catturate (peso = 50)
- Percentuale di pedine nere catturate:
  - Nel caso sia il turno del bianco e almeno una pedina nera può essere mangiata: peso = 49
  - Altrimenti: peso = 50
- Possibilità per il re di vincere:
  - Se ci sono più vie di fuga: peso = 500
  - Se ce n'è una sola:
    - Se è il turno bianco oppure il nero non può bloccarla: peso = 500
    - Altrimenti peso = 0
  - Altrimenti, peso = 0
- Distanza delle pedine nere dal re (peso = 2.5)
- Il re si trova nel castello (peso = 1.5, conta solo a inizio gioco)
- Il re è in pericolo (peso = $-\infty$, conta solo se è il turno del nero)



Euristica del giocatore nero

- Percentuale di pedine nere non ancora catturate (peso = 50)
- Percentuale di pedine bianche catturate:
  - Nel caso sia il turno del nero e almeno una pedina bianca può essere mangiata: peso = 49
  - Altrimenti: peso = 50
- Formazione a rombo (peso = 3, solo in early game)
- Distanza delle pedine nere dal re (peso = 3)
- Il re è in pericolo (peso = 500, conta solo se è il turno del nero)



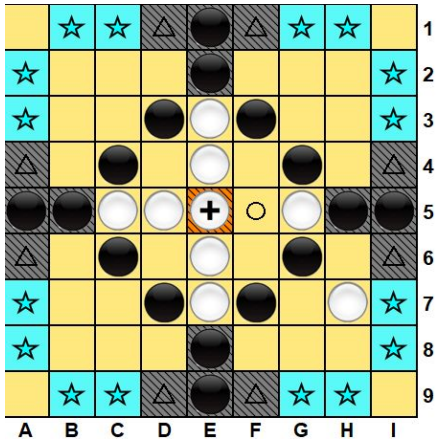
Testing e collaudo

- Macblut è stato collaudato contro tre implementazioni di challenge passate.
  - *WebTablut*, implementazione Web del giocatore *O(sarracino)*, arrivato al secondo posto nella challenge del 2019.
  - *Tavoletta*, partecipante alla challenge del 2022.
  - *Pickle Ricks*, arrivato al primo posto nella challenge del 2020.
- Complessivamente, Macblut vince spesso come giocatore nero, mentre tende più spesso a pareggiare come giocatore bianco. Ha performance particolarmente buone contro WebTablut e Tavoletta.

Altre euristiche considerate

Le seguenti euristiche per il bianco sono state considerate, ma non sono stati trovati pesi tali che portassero a migliori abilità di gioco:

- Euristica della mobilità del re:
  - Dà maggiore valore a stati in cui il re ha più azioni disponibili; può tendere a muovere il re in posizioni potenzialmente pericolose.
- Euristica del re protetto:
  - Dà maggiore valore a stati in cui il re ha pedine amiche vicine, che lo proteggono dall'essere mangiato; può portare a stati in cui pedine amiche impediscono al re di muoversi.
- Euristica dello "sparare alcuni bianchi lontano":
  - All'inizio del gioco, dà alto valore a stati con pedine bianche poste in periferia del tavolo da gioco, in modo che le pedine bianche non siano complessivamente "circondate".



Grazie per l'attenzione

GitHub: https://github.com/MAC-Projects/Macblut