

Tic-Tac-Toe

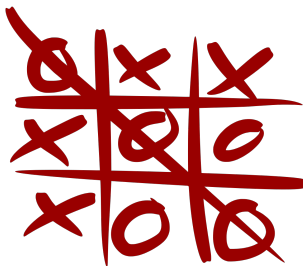
Progetto di Intelligenza Artificiale

Davide Rigoni, Giovanni Mazzocchin
September 21, 2017



UNIVERSITÀ
DEGLI STUDI
DI PADOVA

Lo scopo del progetto era l'implementazione in **Java** del gioco **Tris**, con diversi algoritmi di ricerca e valutazione dei nodi terminali.



L'interfaccia scelta è la riga di comando, in cui l'utente seleziona la configurazione e inserisce le mosse.

```
giovanni@giovanni: ~/Desktop-DV/Notebook-PC: /desktop
----- Initial Config -----
Insert the number of the first player:
 0 - Player0
 1 - Player1
Number: 0
Insert the number of rows and columns: 3
Insert the number of the locked cells: 0
Do you want to play against the PC? 'yes' or 'no': yes
Choose one of these algorithms
 0 - Normal
 1 - Rotate
 2 - Alpha-Beta Pruning
 3 - Pruning with Rotate
Number: 0
Choose one of these EF
 0 - Function number 1
 1 - Function number 2
Number: 0
Insert the max depth: 100
----- End Initial Config -----
----- START -----
Following the field:
| | | |
| | | |
| | | |
| | | |
Make your move player0
Enter the row number: 0
Enter the column number: 0
Following the field:
| x | | |
| x | | |
| | | |
| | | |
```

- Numero del giocatore che esegue la prima mossa;
- Dimensione del tavolo da gioco;
- Numero di celle bloccate;
- Variante di Minimax utilizzata dall'intelligenza;
- Funzione di valutazione utilizzata dall'intelligenza;
- Profondità massima raggiungibile da Minimax.

Sono state implementate quattro versioni dell'algoritmo:

- **Minimax** semplice;
- **Minimax** con controllo delle **rotazioni**;
- **Minimax** con **alpha-beta pruning**;
- **Minimax** con controllo delle **rotazioni** ed **alpha-beta pruning**.

Le tre rotazioni possibili sono state implementate così:

- *90 gradi a destra*: trasposizione + inversione delle colonne;
- *90 gradi a sinistra*: trasposizione + inversione delle righe;
- *180 gradi*: inversione delle righe + inversione delle colonne.

L'albero di un campo 3×3 può essere visitato interamente, ma già con un campo 5×5 questo diventa impossibile su un computer comune.

Sono state dunque realizzate due funzioni euristiche che rendono possibile un gioco pseudo-intelligente.

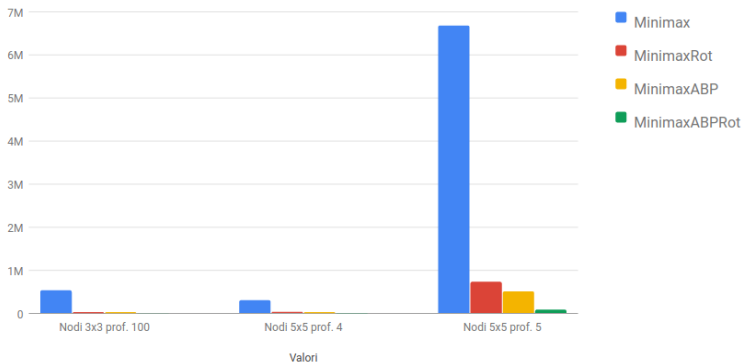
Questa funzione di valutazione dà un punteggio al campo visitato unicamente in base al numero di caratteri allineati del giocatore interessato.

Non si preoccupa minimamente di bloccare le mosse dell'avversario.

Qui viene considerato anche il punteggio dell'avversario:
utilizzando questa strategia è possibile notare come l'intelligenza
sia in grado di difendersi.

- Facendole giocare una contro l'altra il risultato finale è di **patta**;
- Giocando a profondità diverse tra loro il risultato finale è di **vittoria** da parte dell'euristica associata alla profondità maggiore.

Statistiche: nodi esplorati



Statistiche: tempi di esecuzione

