**PROGETTO CLIPS**

*Amedeo Racanati 928995*

*Angelo Pio Sansonetti 928869*

**Introduzione al problema**

Lo scopo del progetto è stato quello di implementare un sistema esperto che giochi ad una versione semplificata della battaglia navale all’interno di una griglia 10x10. All’interno della griglia l’agente deve cercare di individuare le seguenti navi:

* 1 corazzata da 4 caselle
* 2 incrociatori da 3 caselle ciascuno
* 3 cacciatorpedinieri da 2 caselle ciascuno
* 4 sottomarini da 1 casella ciascuno

La versione di battaglia navale cui l’agente dovrà giocare è semplificata poiché:

* il contenuto di alcune celle sarà noto fin dall’inizio
* in corrispondenza di ciascuna riga e colonna l’agente conosce il numero di celle che contengono navi
* Nella disposizione delle navi deve esserci almeno una cella libera (cioè con dell’acqua) tra due navi. Noi abbiamo assunto che tra due navi vi deve essere almeno una cella libera anche diagonalmente

Il sistema esperto potrà eseguire all’interno della griglia le seguenti azioni:

* Fire [x,y]: permette di vedere il contenuto della cella
* Guess [x,y]: il sistema esperto ipotizza che ci sia una nave in posizione [x, y].
* Unguess [x,y]: il sistema esperto ritratta l’ipotesi sviluppata con la guess
* Solve [x,y]: il sistema esperto ritiene di aver risolto il gioco e attiva il calcolo dello score

Il sistema esperto ha a disposizione massimo 5 fire e in un dato momento il numero di celle marcate come *guessed* è 20. L’obiettivo è quindi marcare tutte le celle contenenti porzioni di navi come guessed o eventualmente colpirle con una azione di fire.

Quando il sistema esperto ritiene di aver concluso e richiama il calcolo dello score, questo viene calcolato con la seguente formula:

(10∗*fok*+10∗*gok*+15∗*sink* )−(25∗*fko*+15∗*gko*+10∗*safe*)

Dove:

* fok è il numero di azioni fire che sono andate a segno
* gok è il numero di celle guessed corrette
* sink è il numero di navi totalmente affondate
* fko è il numero di azioni fire andate in acqua
* gko è il numero di celle guessed errate
* safe è il numero di celle che contengono una porzione di nave e che sono rimaste inviolate (né guessed né fired)

**Il nostro sistema esperto**

*La relazione deve evidenziare:*

*- come avete modellato la conoscenza, in particolare se avete dovuto creare fatti (ordinati o nonordinati)*

*per modellare ipotesi ecc.*

*- come avete modellato le regole di expertise. Sarebbe bene che provaste ad implementare diverse*

*strategie di soluzione. Cioè diversi sistemi esperti più o meno capaci e li metteste a confronto.*

*- dovete fare prove con scenari alternativi, variando, oltre alla posizione delle navi, anche*

*l’osservabilità iniziale (più o meno caselle note all’inizio).*

*- dovreste poter individuare i limiti della vostra soluzione: come si comporta se all’inizio non*

*conosce nulla?*

Il nostro sistema esperto è composto da 3 moduli:

* Agent
* Inference cell base
* Agent decision

QUESTA PARTE LA FA AMEDEO

Il modulo agent ha lo scopo di effettuare delle prime inizializzazioni di fatti,….

Il modulo inference cell base effettua diverse inferenze basate su ….

Ci sono diversi livelli di inferenza. Quelle basate sul mero contenuto di singole celle adiacenti tra loro. Quelle basate sulle navi e quelle basate sulle possibili combinazioni

CAPIRE SE SERVE COMMENTARE OGNI METODO O MENO

Infine il modulo agent decision è quello che effettua le decisioni sulle azioni da intraprendere.

Abbiamo sviluppato due tipi di questo modulo, greedy e standard

….

**Mappe e risultati**

Per sperimentare il comportamento degli agenti sono state create diverse **mappe** che in combinazione con quanto l’agente conoscesse all’inizio sul contenuto di alcune caselle, contribuisce alla definizione di diversi **scenari**.

Le mappa sviluppate sono state di tre tipi:

1. Mappa con disposizione delle navi che non segue un particolare pattern
2. Mappa con disposizione delle navi disposte vicine tra loro
3. Mappa con disposizione delle navi disposte lontane tra loro

Per ognuna di queste mappe è stato testato il comportamento dell’agente in base a quanti indizi gli sono stati forniti inizialmente:

1. Nessun indizio
2. Poco indizi: l’agente conosce il contenuto di 4 caselle della mappa
3. Molti indizi: l’agente conosce il contenuto di 7 caselle della mappa

Per poter avere un riscontro immediato delle azioni dell’agente e dei risultati da esse prodotti oltre a raccogliere lo score, è stata realizzata una piccola interfaccia grafica.

Legenda risultati grafici:

* Bandierina verde: guess corretta
* Bandierina rossa: guess non corretta
* Fuoco: fire corretta
* Acqua: fire non corretta

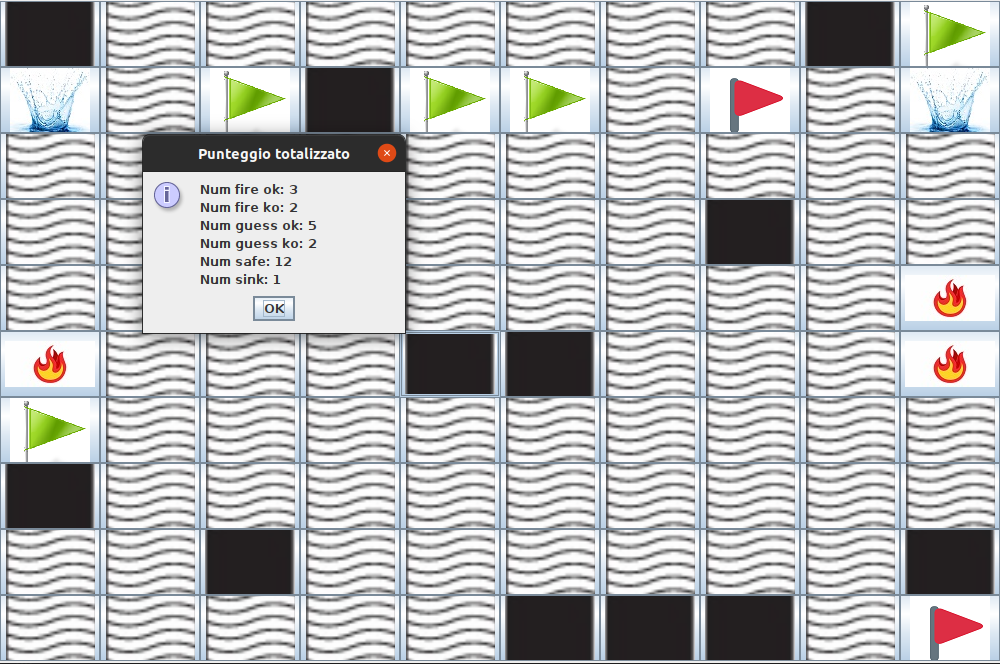
Inoltre quando una casella è evidenziata con dei bordi rossi, essa indica che l’agente era a conoscenza del suo contenuto all’inizio della partita

**Agente standard**

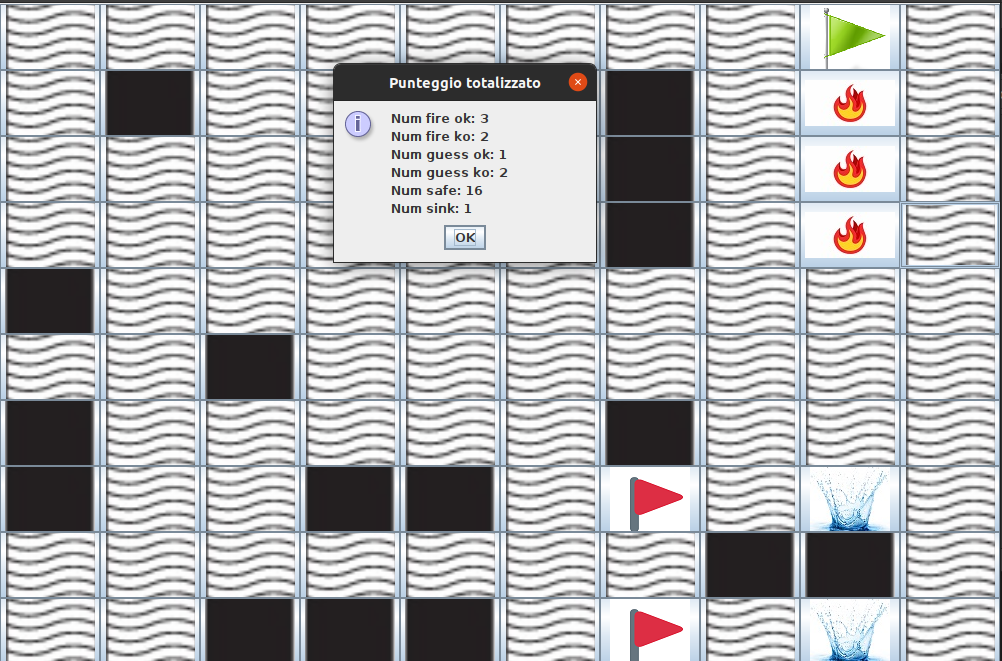
|  |  |  |  |
| --- | --- | --- | --- |
| **Mappa/Indizi** | **Nessuno** | **Pochi** | **Molti** |
| **Disp. sparsa** | **-105** | 105 | 215 |
| **Disp. vicine** | 160 | 235 | **280** |
| **Disp. Normale** | **-185** | 195 | 205 |

Di seguito sono mostrati graficamente i risultati grafici dei due scenari peggiori e del migliore.

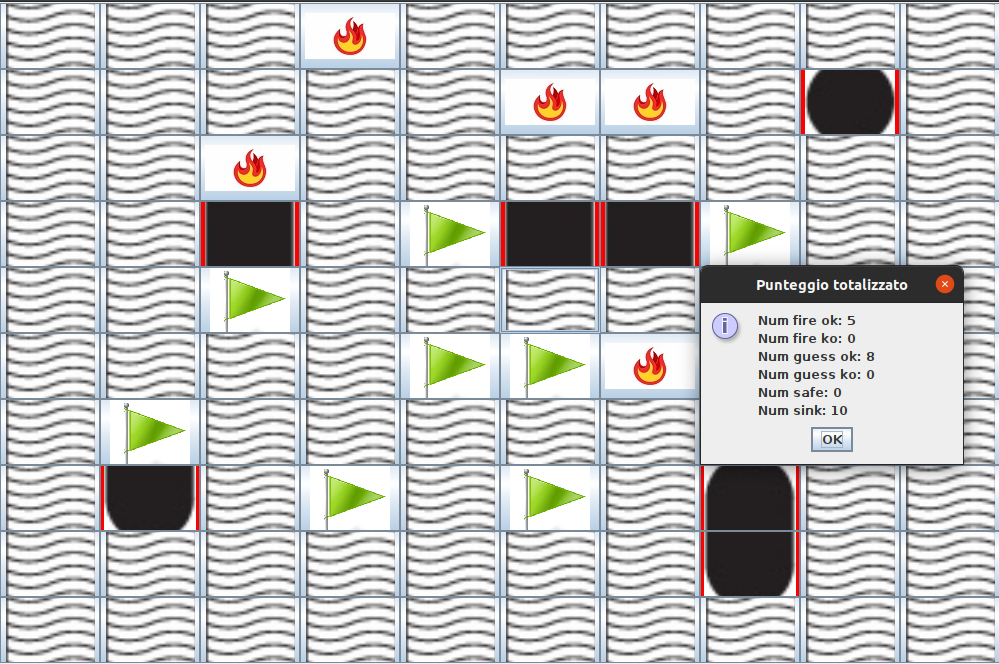
Disposizione navi sparsa e nessun indizio:



Disposizione navi normale e nessun indizio:



Disposizione navi vicine e molti indizi:

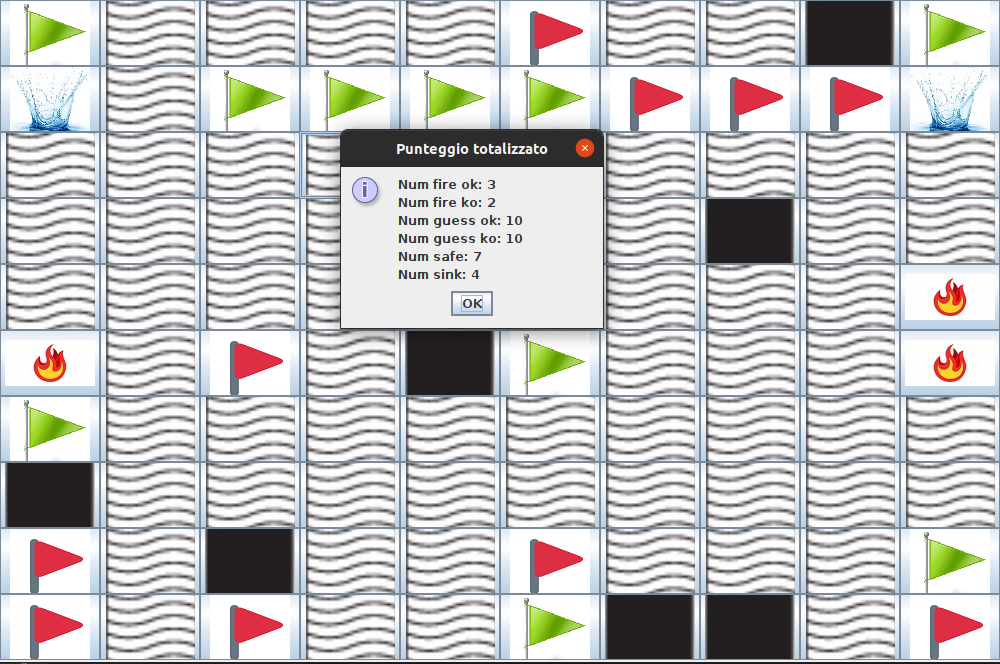


**Agente greedy**

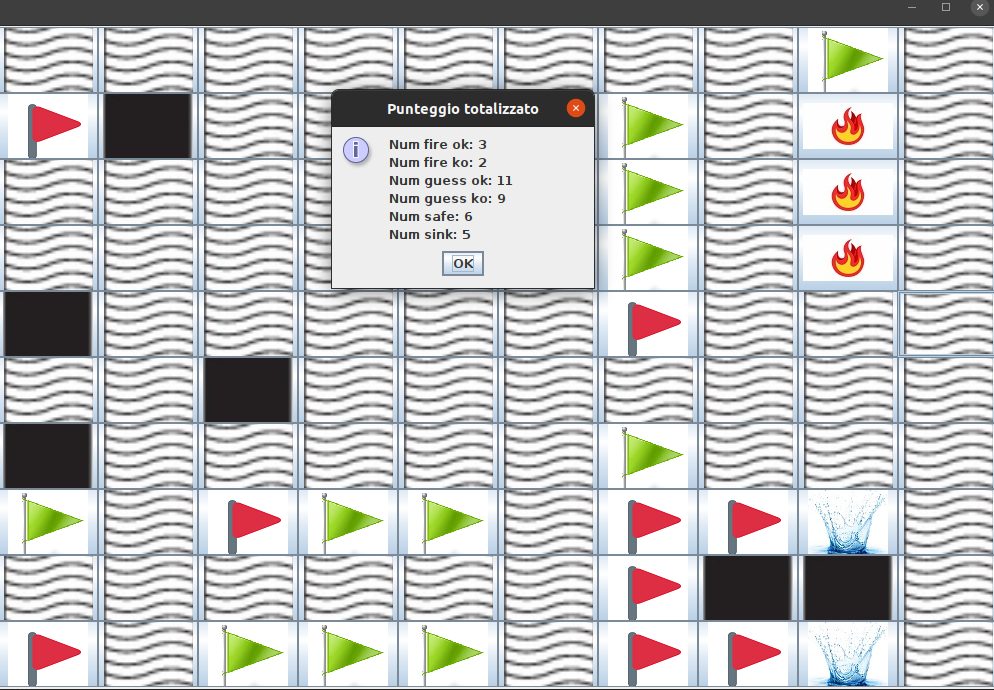
|  |  |  |  |
| --- | --- | --- | --- |
| **Mappa/Indizi** | **Nessuno** | **Pochi** | **Molti** |
| **Disp. sparsa** | **-80** | 25 | 205 |
| **Disp. vicine** | 25 | 205 | **280** |
| **Disp. normale** | **-30** | 175 | 185 |

Di seguito sono mostrati graficamente i risultati grafici dei due scenari peggiori e del migliore.

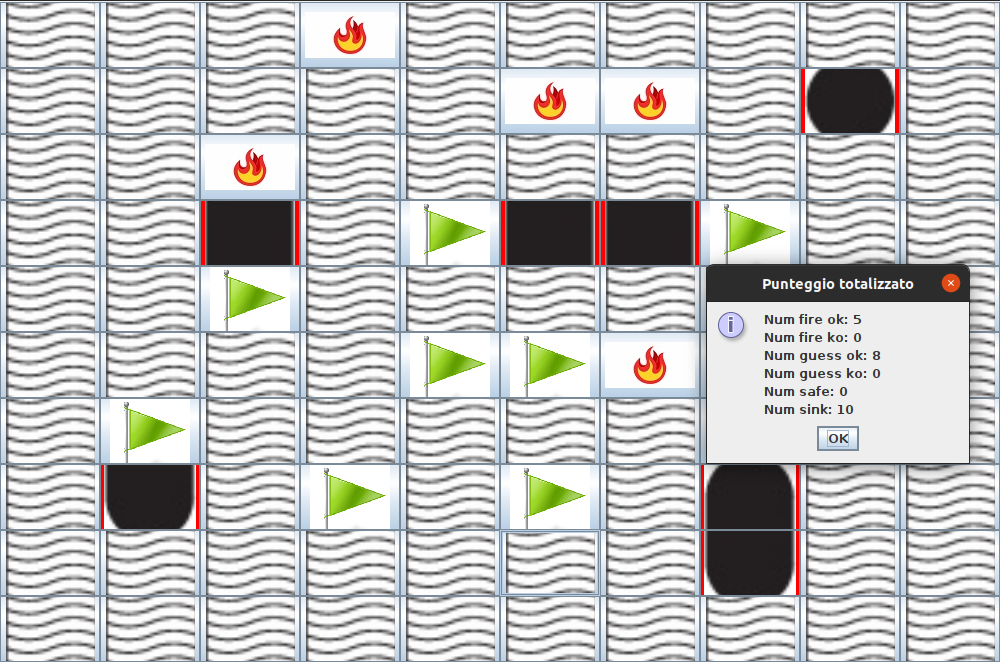
Disposizione navi sparsa e nessun indizio:



Disposizione navi normale e nessun indizio:



Disposizione navi vicine e molti indizi:

****

**Considerazioni**

Come era aspettabile entrambi gli agenti riscontrano performance migliori quando ricevono più indizi, viceversa sono peggiori quando gli agenti non ricevono alcun indizio. In particolare si nota che quando l’agente non conosce nulla all’inizio i punteggi per entrambi gli agenti sono sempre negativi tranne che per la mappa in cui le navi sono disposte vicine tra loro questo perché conoscendo per ogni riga e colonna il numero di celle contenenti porzioni di navi, essendo quest’ultime concentrate gli agenti riescono ad inferire ed escludere molte celle che contengono acqua nonostante non ricevano alcun indizio. Confrontando invece il comportamento dei due agenti tra le diverse mappe, per lo stesso motivo descritto precedentemente, entrambi gli agenti registrano le performance peggiori in corrispondenza di disposizioni delle navi sparse e migliori quando quest’ultime sono tra loro vicine. Confrontando ora i due agenti si nota come in corrispondenza di scenari con nessun indizio, l’agente greedy riesce ad arginare di più gli effetti di questo svantaggio cumulando uno score migliore rispetto l’agente standard.