Alessandro Tironi, Filippo Vajana

Università degli studi di milano

Agente Stealth

Progetto di Intelligenza Artificiale

Specifiche di progetto

# Introduzione

Accade, in molti videogiochi, che il giocatore debba impersonare un soldato con l’ordine di infiltrarsi tra le fila nemiche per recuperare un obiettivo di valore. Scopo di questo progetto è la realizzazione di un agente intelligente in grado di portare a compimento la missione con un atteggiamento attivo verso l’ambiente circostante.

# Agente

L’agente soggetto delle simulazione ha l’obiettivo primario di raggiungere l’obiettivo minimizzando la distanza, e quindi il tempo, rimanendo però non identificato dalle sentinelle.

Inizialmente l’agente conosce solamente la planimetria della mappa, la posizione delle sentinelle e la posizione del suo obiettivo; durante la simulazione l’agente dovrà fare delle assunzioni sui percorsi di ronda, generare un piano ottimale in base alla propria conoscenza, mettere in pratica il piano e in base allo stato del mondo aggiornare coerentemente la base di conoscenza.

Nel caso l’agente venisse scoperto la simulazione verrà fatta ripartire mantenendo intatta la base di conoscenza dell’agente, in questo modo i fallimenti saranno occasione di apprendimento (“*Sbagliando si impara*”).

# Dominio di azione

L’agente si muove all’interno di aree rettangolari dalla planimetria via via più complessa con l’obiettivo di raggiungere il prigioniero restando celato alle sentinelle.

Le sentinelle si muovono lungo ronde chiuse ed a seconda della mappa potranno avere un comportamento non deterministico. Ogni sentinella ha intorno a sè un’area di influenza all’interno della quale è in grado di avvistare l’agente.

All’infuori di ostacoli fisici e sentinelle il terreno sul quale si svolge la simulazione non presenta elementi nè di vantaggio nè di svantaggio per l’agente.

## Rappresentazione geometrica

Elenco dei principali tipi e predicati utilizzati e dichiarati nel file livello\_spec

*type [size(number,number)]:size.*

Rappresenta le dimensioni della mappa rettangolare

*type [p(number,number)]:punto.*

Rappresenta una posizione 2D all’interno della mappa

*type [o, p, ' ']:terreno.*

Rappresenta i diversi tipi di terreno dove ‘o’ è un ostacolo mentre ‘p’ è l’obiettivo

*type [d(number,number)]: direzione.*

Definisce una direzione mediante lo spostamento unitario (Verticale, Orizzontale)

*type [n,s,e,o]:punto\_cardinale.*

I punti cardinali fondamentali

*type [area(punto,punto)]:area.*

Area rettangolare descritta dal vertice di NE e da quello di SO

*pred direzioni(punto\_cardinale, direzione).*

Fornisce la direzione relativa ad un punto cardinale

*pred adiacenti(punto, list(punto)).*

Fornisce la lista di posizioni adiacenti al punto specificato

*pred distanza\_quadretti(punto, punto, number).*

Calcola la distanza tra due punti nello spazio Manhattan

*pred map(punto,terreno).*

Definisce un punto della mappa caricata con un determinato tipo

*pred position(punto).*

Definisce la posizione dell’agente

*pred goal(punto).*

Definisce la posizione dell’obiettivo

## Area di gioco e sentinelle

Nell’area di gioco i nemici sono rappresentati dalle sentinelle e dalle relative aree di influenza lungo i percorsi di ronda. I percorsi di ronda sono definiti in file separati e possono avere elementi di casualità per aumentare il livello di difficoltà

*pred punto\_area(punto, area).*

Verifica che il punto specificato si trovi all’interno dell’area fornita. Usato per determinare

l’avvistamento dell’agente ad opera di una sentinella

*pred stato\_sentinella stato\_sentinella(?Id,?Punto,?Dir) SEMIDET*

Fornisce lo stato attuale della sentinella specificata che si troverà in una posizione e guarderà in una direzione

*pred soldato\_avvistato(id\_sentinella,punto,tempo).*

Rileva se il soldato è stato avvistato da una sentinella

*pred ronda\_sentinella(id\_sentinella,id\_ronda).*

Definisce la relazione tra una sentinella e la rispettiva ronda

*pred ronda(id\_ronda,punto,direzione,tempo).*

Descrive una ronda mediante la variazione nel tempo di posizione e direzione della sentinella

*pred area\_sentinella(sentinella, punto\_cardinale, area).*

*pred area\_sentinella(id\_sentinella, direzione, area).*

Definiscono l’area rettangolare di influenza della sentinella

## Fluenti e stato del gioco

*gen [st(punto, punto, tempo)]:stato.*

Definisce lo stato del gioco tramite la posizione al tempo attuale di agente e obiettivo

L’agente può prendere le seguenti decisioni

* *vado(punto, punto)* : l’agente cerca un piano per raggiungere la destinazione
* *avanzo(punto, tempo)* : l’agente al tempo specificato avanza al punto
* *attesa* : l’agente attende 3 cicli temporali
* *aspetto* : l’agente attende 1 ciclo temporale
* *termino(evento)* : l’agente termina

*gen [map(punto, terreno)]: assumibile.*

L’agente può effettuare assunzioni circa la natura di un punto della mappa