

# Relazione progetto IALAB

Andrea Aloi, Maurizio Dominici, Alessandro Serra

10 aprile 2014

## **Sommario**

Relazione del progetto Intelligenza Artificiale e Laboratorio anno scolastico 2012-2013. Illustrazione delle diverse strategie proposte, con una disamina dettagliata dei pregi e difetti di ognuna di esse, approfondendo anche i motivi delle scelte fatte sia in fase di progettazione che di implementazione. Vengono illustrate X strategie: ...

# Indice

<b>1</b>	<b>Introduzione</b>	<b>1</b>
1.1	Agent . . . . .	1
1.2	Moduli . . . . .	1
1.3	Controllore . . . . .	2
<b>2</b>	<b>Strategia di base</b>	<b>3</b>

# Capitolo 1

## Introduzione

Tutte le strategie illustrate in questo documento hanno in comune la struttura. Il codice è strutturato in maniera gerarchica, in modo tale che ogni modulo si occupi di un compito particolare da svolgere, il tutto coordinato da una serie di controllori inseriti nel file `agent.clp`, fornito nello scheletro iniziale del progetto.

### 1.1 Agent

Partiamo illustrando il modulo `agent`: questo modulo, concettualmente, si occupa solo di gestire e coordinare i vari moduli a cui è assegnato un compito specifico. `Agent` controlla che la condizione di entrata per un modulo sia soddisfatta, quindi dà il focus al modulo in questione attendendo che il lavoro sia svolto. Quando questo modulo rilascerà il focus attraverso una `pop`, `agent` avrà a questo punto la possibilità di dare il focus a un altro modulo che svolgerà un altro sottocompito.

### 1.2 Moduli

Ogni modulo, come detto, ha un compito particolare (verranno illustrati in seguito i vari moduli con i rispettivi compiti). La struttura di ogni modulo è abbastanza semplice: riceve il focus dal modulo `agent`, il quale controlla che la condizione affinché sia svolto il compito assegnato al modulo specifico sia soddisfatta, svolge le operazioni delle quali ha responsabilità di esecuzione, asserisce una condizione per cui sia comprensibile al modulo `agent` che il compito richiesto è stato eseguito con successo e quindi rilascia il focus mediante una `pop`. Ci sono poi, a seconda delle varie strategie, delle varianti a questo comportamento, ma generalmente il flusso di esecuzione è quello

illustrato. Alcune eccezioni possono essere la necessità di spezzare la routine di ogni turno, ad esempio quando il modulo del tempo inferisce che il tempo rimanente per dirigersi verso un gate scarseggia, ma entreremo nel dettaglio durante l'illustrazione specifica delle varie strategie.

## 1.3 Controllore

Entriamo ora più nello specifico nel comportamento del controllore. Il controllore si occupa, come detto, di dare il focus al modulo corretto. Quest'ultimo viene fatto identificando una determinata routine da svolgere per ogni passo (identificato dallo slot step del template status fornito nello scheletro iniziale del progetto):

- controllo del tempo rimanente
- inform delle celle visibili
- individuazione della cella verso cui spostarsi
- controllo che la cella individuata permetta di raggiungere un'uscita (questo ai fini di non finire in vicoli ciechi)
- calcolo del percorso per raggiungere la cella individuata mediante l'algoritmo  $A^*$
- computazione effettiva dei passi calcolati e memorizzati dall'algoritmo  $A^*$

## Capitolo 2

### Strategia di base

La prima strategia illustrata è quella più semplice, che useremo come metro di paragone.