

FUZZY LOGIC

CORSO DI GAME PROGRAMMING
1° ANNO

Docente **Davide Caio**



LOGICA

La logica è uno strumento usato da centinaia di anni per rappresentare la conoscenza. Esistono diversi tipi di logica:

- Logica proposizionale: valore di verità per le proposizioni
Es: (L'erba è verde) è “true”.
- Logica del primo ordine: valore di verità per i predicati
Ci sono variabili e quantificatori
Es: $\exists X : (\text{Verde } X)$ is “true”
- Logica del secondo ordine: predicati di predicati
Es: $\exists X, Y : (\text{Believe } Y ((\text{Green } X) \text{ is true}))$ is “false”



LOGICA

Nonostante il potere espressivo dei vari tipi di logica cambia molto, tutti questi tipi di logica si basano su un fondamento comune: cioè che il risultato, il valore che possono assumere predici o proposizione è solamente “true” oppure “false”.

Esiste anche una logica a più valori:

- Logica a tre valori: true (1), false (0), undefined ($\frac{1}{2}$)
- oppure logica a infiniti valori: un valore continuo di verità T con $T \in [0..1]$



FUZZY LOGIC

La fuzzy logic è una logica a infiniti valori, con la verità che può essere espressa con un valore da 0 a 1.

Le proposizioni sono espresse nella forma

A is L

dove:

- A è una variabile linguistica
- L è una etichetta che denota un insieme fuzzy (fuzzy set).

Esempio: Temperature is HIGH



FUZZY LOGIC

Spesso il **grado di verità** viene confuso con la **probabilità**.

Sbagliatissimo.

Il grado di verità indica quando è vera (su una scala da 0 a 1) una certa sentenza.

La probabilità indica quanto probabilmente un evento accadrà (e quindi diventerà vero).



FUZZY LOGIC - ESEMPIO

Il player è vicino

Logica tradizionale:

→ Nella logica tradizionale questa proposizione può essere solamente vera oppure falsa. Quindi per implementare una logica di questo tipo bisogna impostare un valore di distanza, e se il giocatore è più vicino di questo valore di distanza allora la sentenza ottiene il valore di True, se è più lontano ottiene il valore di False.

Fuzzy Logic:

→ Nella fuzzy logic invece possiamo ottenere un valore tra 0 e 1 che indica quando il giocatore è vicino. Possiamo inserire una soglia massima che porta il valore di vicinanza a 0 e fare una interpolazione di questo tipo:

```
playerClose = Math.Clamp (1 - (playerDistance / maxDistance),0,1);
```



FUZZY LOGIC - FUZZY SET

Un fuzzy set quindi è un insieme dove gli elementi che lo compongono non appartengono solo a un sottoinsieme ma possono appartenere a diversi sottoinsiemi con un certo grado di appartenenza (anche esso tra 0 e 1).

Insieme Altezze

Sottoinsieme:

- alti
- medi
- bassi

Una persona alta 1.40 m può appartenere al sottoinsieme Bassi con un valore di appartenenza più elevato rispetto al valore di appartenenza del sottoinsieme medi, e può appartenere con un valore di appartenenza prossimo allo 0 all'insieme dei alti.



FUZZY LOGIC - AI

Come questo tipo di logica può venirci in aiuto nella costruzione di intelligenze artificiali?

Semplicemente perché alcuni comportamenti non possono essere modellati con la logica tradizionale. O meglio, possono essere modellati ma non con un risultato soddisfacente.

Esempio:

Aggro in WoW (not ufficial):

- Danno subito dai vari player (più è alto più pesa nel valore totale dell'aggo)
 - I tank hanno un moltiplicatore molto alto
- Distanza del player dal nemico
- Classe del player (gli healer hanno un modificatore più alto rispetto a tutti gli altri).
- Vita del giocatore

Il valore di aggressività di un nemico verso un giocatore è la somma di tutti questi valori pesati in base ad alcuni parametri. Chi ha la somma pesata più elevata sarà il player attaccato dal nemico.



HEADS - PROGETTO DA CONSEGNARE

In heads abbiamo due situazioni in cui possiamo utilizzare la fuzzy logic:

- Quando rileva più di un player nel suo cono visivo e deve decidere chi attaccare
- Quando sta attaccando e deve decidere cosa fare
 - Continuare ad attaccare lui
 - Passare a un altro player visibile
 - Andare a recuperare la vita



HEADS - GET PLAYER TO ATTACK

Per decidere quale giocatore visibile attaccare i classificatori da usare sono:

- Distanza dal giocatore (più è vicino più il valore di verità fuzzy di distanza dal giocatore sarà alto)
- Percentuale di energia del player (più la sua energia è bassa rispetto alla sua energia totale, più il valore di verità fuzzy è alto)
- Direzione del giocatore (più il giocatore è diretto verso la direzione del nemico, quindi si sta avvicinando, e più il valore di verità fuzzy è alto).



HEADS - RECHARGE

La decisione se deve stare nello stato attuale del nemico oppure passare allo stato di recharge.:

- Energia rimasta: più la mia percentuale di energia è bassa e più questo parametro si avvicina a 1.
- DifferenzaConEnergiaPlayer: più la mia energia è alta rispetto a quella del player e più questo parametro si avvicina a 0.
- Distanza dal power up più vicino: più il power up è vicino più questo parametro si avvicina a 1.

Se la somma di questi pesi è maggiore di una soglia che impostate allora passa dallo stato di attack a quello di recharge.

(Opzionale) Il power up da andare a prendere non è quello semplicemente più vicino, ma, in base a dei parametri di fuzzyness che descrivete nel video, sarà quello che si avvicina di più alla somma dei pesi.



HEADS - COMPORTAMENTO NEMICO

- Patrol → va in giro in modo casuale nella finestra
 - (Plus) Entro una certa distanza dalla posizione iniziale che aveva quando è entrato in questo stato.
- ChasePlayer → Quando uno o più player entrano nel campo visivo, seguendo le regole di fuzzyness descritte nelle slide precedenti, deve iniziare a inseguire il giocatore migliore
 - Ogni 0.5 secondi ricalcola se il player maggiore è ancora quello che sta inseguendo in questo momento. In caso positivo continua a inseguire quello, altrimenti inizia ad inseguire un altro player.
- AttackPlayer → Quando è abbastanza vicino al player inizia ad attaccarlo
 - Ogni 0.5 secondi ricalcola sempre se il player che sta attaccando è il miglior player da targettare secondo le regole di fuzzyness descritte prima.
- Recharge: Quando entra nello stato di recharge si avvicina al power up più vicino.
 - (Plus) se il power up viene preso da un giocatore trovo quello più vicino e mi avvicino a lui. Se sono finiti allora mi rimetto in Patrol

IN OGNI STATO, OGNI 0.5 S, CONTROLLA SE NON DOVREBBE PASSARE NELLO STATO DI RECHARGE.



HEADS - DEADLINE

Il progetto heads descritto nelle slide precedenti dovrà essere consegnato entro il **1 Maggio ore 12.00.**

Come al solito dovrete fornire progetto, eseguibile e video di spiegazione delle vostre meccaniche.

Se implementate tutto quello che c'è scritto tranne i Plus potrete ottenere un voto tra 6 e 7.

Se aggiungete i Plus otterrete un valore tra 8 e 9.

Se aggiunge altre nuove feature (che andranno dettagliate in un file di testo lasciato nello zip della consegna) potrete prendere un voto tra 10 e 11.