Università degli Studi di Perugia

Computational Intelligence
Progetto:
Utilizzo degli algoritmi genetici per la creazione di mazzi di carte

Mattia Polticchia
Matricola: 332040

Indice

Prefazione

Le carte del gioco e le rispettive regole sono basate su una semplificazione dei componenti degli attuali giochi di carte, non prevedono una mano e dei costi di giocata delle carte per non aumentare il numero di vincoli da imporre all'algoritmo.
L'idea è comunque ampliabile in varie maniere e i vincoli non limitanti nelle future implementazioni.
I deck creati dall'algoritmo sono da vedersi come dei deck da usare, e l'algoritmo non è da intendersi come un simulatore di battaglie Le carte dei deck potrebbero differire dalle carte della collezione e questa, ai fini del gioco, è una cosa valida.

La struttura del progetto

Per codificare il problema si sono realizzate delle classi che rappresentavano tutti gli elementi utilizzati

- La classe Card:
  Rappresenta l'oggetto rispettivo delle carte e sono gli alleli del nostro genoma.
- La classe Deck:
  Rappresenta l'oggetto rispettivo del deck che nel nostro caso rappresenta il cromosoma.
- La classe CollectionManager:
  Rappresenta l'oggetto con cui viene mantenuta la collezione di carte e che fornisce supportoin fase di mutazione.
- La classe Game:
  Questa classe, dati due deck, effettua una partita del gioco restituendo i punteggi e il numero di turni.
- La classe Problema:
  Contiene tutte le funzioni utili per interfacciare l'algoritmo col problema e con le carte.
- La classe Algoritmo Genetico:
  Rappresenta l'algoritmo nella sua interezza, necessita di problema per funzionare.

Capitolo 1

Il Gioco

1.1 Carte

Le carte sono composte da tre componenti principali:

- Attacco .
  Può variare da 0 a 10
- Difesa:
  Può variare da 2 a 13
- Abilità:
  Alcune carte potrebbero non avere abilità. Quelle che la possiedono attivano la propria in alcuni momenti della partita.
  Tipi di abilità:
  - Attaccante: +1 all'attacco della carta
  - Difensore: +1 alla difesa della carta
  - Esperto: un punto aggiuntivo se la carta sopravvive alla battaglia

1.2 Regole

Ogni giocatore ha un mazzo di 10 carte e un contatore di punti vittoria. Ogni turno si pescano 3 carte e si mettono in campo. Si fanno combattere valutando rispettivamente attacco e difesa. Ogni propria carta che rimane in campo genera un punto vittoria. Quando le carte del mazzo finiscono, esso si rimescola e si continua la partita.
Il primo dei due giocatori che arriva ad un determinato numero di punti vittoria vince la partita.

1.3 La Creazione delle carte

Le carte utilizzate nel gioco sono state create casualmente. Tutte prive di abilità tranne lo 0.05% a cui sono state date delle abilità casuali. Attacco e Difesa sono stati generati tramite la distribuzione "discrete uniform" sui valori menzionati nella descrizione delle carte. I mazzi, invece, vengono generati pescando 10 carte uniche dalla collezione delle carte.

Capitolo 2

L'algoritmo Genetico

2.1 I Parametri

All'algoritmo sono stati impostati dei parametri sceglibili dall'utente ed alcuni necessari per il funzionamento.

- Prob è la rappresentazione del problema
- seed è il seme del generatore casuale
- n_elem è il numero di elementi della popolazione
- p_cross è la probabilità di crossover
- p_mut è la probabilità di mutazione
- selection "none" per utilizzare la selection tramite la fit, mentre "tournament" per utlizzare la selezione a scontri tra le carte
- elit è un flag booleano che ci dice se applicare l'elitismo o no.

2.2 Inizializzazione della popolazione

In questa fase dell'algoritmo andremo a creare la popolazione per l'algoritmo. La popolazione viene inizializzata creando dei deck prendendo le carte dalla collezione generale di carte senza prendere carte duplicate per ogni singolo deck. Ogni deck avrà carte differenti al suo interno, non è assicurato che tutti i deck abbiano le carte diverse fra di loro.

2.3 Selezione

Per la selezione sono stati implementati due metodi:

- Un metodo utilizza la fitness facendo 1 / fo(x) e poi tramite il metodo della roulette wheel seleziona gli elementi per il crossover.
- Il secondo metodo fa scontrare i mazzi uno contro l'altro alla meglio di tre partite e utilizza un rateo vittorie / numero popolazione per impostare la probabilità per la roulette wheel.

2.4 Crossover

La riproduzione avviene eseguendo un crossover a due tagli. Uno dei tagli è nella prima metà del mazzo mentre l'altro taglio è nella seconda metà.

2.5 Mutazione

La mutazione effettua una scelta casuale tra due possibili:

- La sostituzione di una carta
- Il cambiamento dell'abilità di una carta

Questa scelta avviene qualora la mutazione si verifica. Nel caso venga pescata la stessa carta che si sta sostituendo, si muta.

2.6 Valutazione

La valutazione effettua una stima della bontà del mazzo.
La stima viene fatta con la seguente formula

$$\Sigma_{i=1}^{len(Deck)} \frac{card[i].attack + card[i].defense}{2} + card[i].skill \tag{2.1}$$

Dove la somma e divisione per 2 di attacco e difesa sono detti corpo della carta mentre il punteggio della skill viene sommato per aggiungere anche il suo valore.

2.7 Update

Dato che si effettua il crossover su tutta la popolazione ma non tutte le coppie effettuano il crossover, l'aggiornamento della popolazione viene fatto utilizzando tutti i figli ottenuti. Valutando e sostituendo il miglior elemento con quello peggiore della popolazione corrente se l'elitismo è stato selezionato.

Capitolo 3

Conclusioni

3.1 Parametrizzazione

Prima di passare alle conclusioni si sono sperimentati i valori da passare all'algorimto genetico tramite una parametrizzazione. La parametrizzazione è stata fatta utilizzando il metodo della selezione tramite fitness per velocità d'esecuzione. Non è stata usata una serie di combinazioni di parametri ma sono stati usati dei valori soglia in quanto i parametri accettati dall'algoritmo sono continui.
I parametri sono stati modificati nel seguente modo:

Elementi popolazione	Prob di cross	Prob di Mutazione	Elitismo
10	0.2	0.05	False
10	0.2	0.05	True
20	0.2	0.05	False
20	0.5	0.25	False
30	0.2	0.05	False
30	0.5	0.25	False
30	0.9	0.5	False

3.1.1 Primo Caso

Con pochi membri della popolazione si è cercato di verificare se l'elitismo influenzasse la ricerca del miglior mazzo pur tenendo la probabilità di mutazione e di crossover molto bassa.

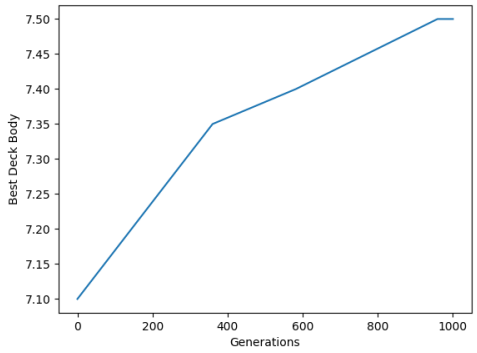

Figura 3.1: Elitismo non utilizzato punteggio 7.5

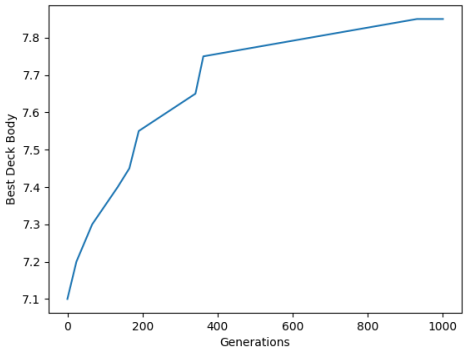

Figura 3.2: Elitismo attivo punteggio 7.85

3.1.2 Secondo Caso

In questo caso si è cercato di evidenziare un effettivo aumento dovuto ai parametri e alla popolazione.

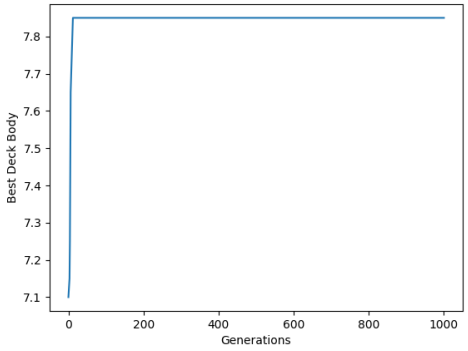

Figura 3.3: Parametri ai valori minimi

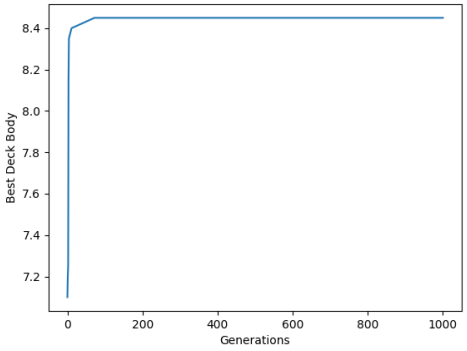

Figura 3.4: Parametri nei valori medi

3.1.3 Terzo Caso

Si è provato tutto il range parametri con una popolazione numerosa per vedere come i parametri influenzavano la generazione dei deck.

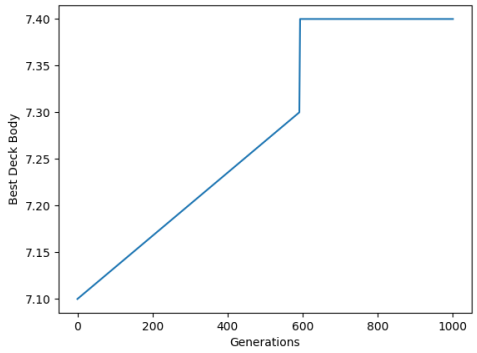

Figura 3.5: Parametri ai valori minimi

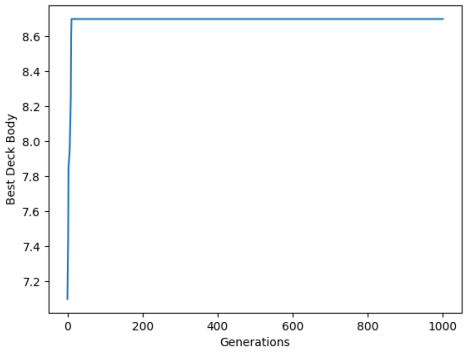

Figura 3.6: Parametri nei valori medi

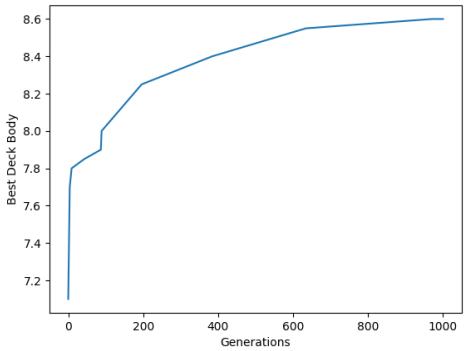

Figura 3.7: Parametri ai valori massimi

3.2 conclusioni

Dopo aver assestato i valori alla soglia massima, l'algoritmo ha generato dei mazzi generalmente buoni. Per trarre le conclusioni finali si sono utilizzati i parametri finali anche sul metodo di selection con lo winrate.
I due metodi di selezione usati hanno prodotto dei risultati simili tra loro ma con alcune particolarità:

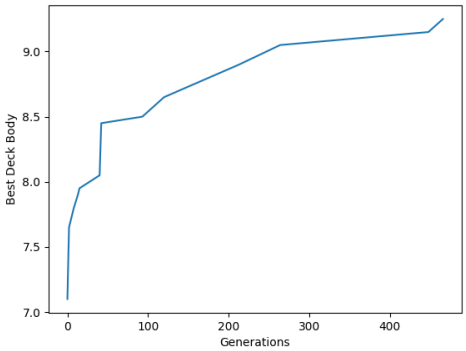

Figura 3.8: 500 Generations 166 secondi per completare

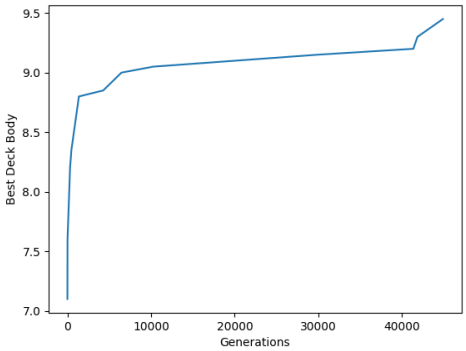

Figura 3.9: 50000 generations 55 secondi per completare

Come possiamo vedere dai due grafici il metodo che simula le partite facendo affrontare i mazzi alla meglio di tre ha portato un aumento del corpo del mazzo significativo in meno generazioni ma ha aumentato anche i tempi d'esecuzione. Il metodo della fitness sul corpo, dalla sua, ha una velocità d'esecuzione molto più alta e ha permesso di ottenere un corpo alto ma in un numero altissimo di generazioni.
I due tipi di mazzo generato differiscono molto.Il primo ha delle carte con un alto corpo ma non molte carte hanno abilità mentre nel secondo abbiamo delle carte con dei body leggermente più bassi ma il punteggio viene bilanciato dalla presenza dell' abilità.