

**Università degli Studi di Torino**

Dipartimento di Informatica

Anno Accademico 2023/2024

**Relazione progetto CLIPS:  
Master-Mind**

Magnano Samuel, Roman Flavio, Mongitore  
Cristian

# Capitolo 1

## Soluzione Human

Nella versione Human puntiamo a raggiungere nel minor numero possibile di step i 4 colori presenti nel secret-code, indipendentemente dalla loro posizione rispetto alla soluzione. Il modo in cui lo facciamo è il seguente:

- Guess manuale iniziale con ottenimento di right-placed e misplaced
- Caso01 (right-placed + misplaced = 0 oppure 1): cambio tutti e 4 i colori scelti, in modo da ottenere right-placed + misplaced = 3 oppure 4 a seconda di quanti colori avevamo indovinato nel primo tentativo.
- Caso23 (right-placed + misplaced = 2 oppure 3): valuto lo swap di ciascuno dei miei colori con gli altri 4 rimasti fuori dalla scelta finchè non ottengo right-placed + misplaced = 4. Ad ogni swap vado a controllare se la guess appena fatta è migliore della best-guess, ossia la miglior guess fatta finora, asserita appena entro nel caso23 ed aggiornata man mano che le nuove guess hanno right-placed + misplaced maggiore. In base al valore di right-placed + misplaced che viene restituito ottieniamo informazioni riguardo sia alla presenza o meno del colore che abbiamo appena inserito, che quello appena rimosso. In questo modo possiamo passare direttamente al colore successivo in quanto sicuri che quello appena valutato sia presente o rimuovere dalla lista di colori per cui fare lo swap il colore appena inserito in quanto sicuri non sia presente nel secret-code, velocizzando il processo.

Una volta raggiunto il nostro obiettivo ricadremo sempre nel caso4 (right-placed + misplaced = 4) dove andremo a generare, una alla volta, permutazioni diverse dei colori ottenuti andando sempre a controllare che la guess che stiamo per asserire non sia già stata proposta. In tal caso valuteremo

permutazioni alternative fin tanto che non ne troveremo una non precedentemente asserita. Il processo verrà iterato fino a raggiungimento della soluzione o termine dei tentativi permessi.

I fatti principali sui quali lavoriamo sono:

- valori di right-placed e misplaced passati da Game
- informations
  - best-guess
  - best-guess-rp-mp (somma right-placed + misplaced)
  - to-be-evaluated (lista di colori da valutare nei vari swap per il caso<sup>23</sup>)
- new-guess-colors: guess che viene costruita ogni volta da 0 sulla base delle guess precedenti o della best-guess e che viene confrontata con le guess già fatte per evitare di asserire tentativi uguali. Una volta verificate le condizioni per asserire la nuova guess verrà fatto retract di tutte le informazioni accessorie presenti in Working Memory in modo da partire con il tentativo successivo con le sole informazioni essenziali.

# Capitolo 2

## Soluzione Computer

Nella versione Computer cerchiamo sempre di raggiungere i 4 colori corretti nel minor numero di passi andando però a valutare i singoli cambiamenti dei right-placed e misplaced che avvengono nel caso23, i quali ci suggeriscono non solo quali colori siano presenti nel secret-code ma anche in quali posizioni.

Proprio per questo motivo abbiamo definito un nuovo template chiamato "solution-informations" come segue:

```
(deftemplate solution-information
  ;; il simbolo "X" indica che non sono sicuri su cosa ci sia in quella posizione
  (multislot known-positions (allowed-values X blue green red yellow orange white black purple))
)
```

La prima volta che entreremo in Computer.clp andremo ad asserire known-positions pari a "X X X X", in modo tale da definire il fatto che non sappiamo quali colori siano presenti in quale posizione. Nel momento in cui entreremo nel caso23, ad esempio subito dopo la guess iniziale o a partire dal caso01, durante la valutazione degli swap non terremo conto esclusivamente della somma right-placed + misplaced ma anche delle variazioni di right-placed tra la nuova guess e la best-guess come segue:

- right-placed della nuova guess > right-placed della best-guess: il nuovo colore è inserito nella posizione corretta rispetto al secret-code. Andremo quindi a modificare la X nella posizione corrispondente in known-positions con il colore appena inserito.
- right-placed della nuova guess < right-placed della best-guess: il colore della best-guess era inserito nella posizione corretta rispetto al secret-code. Andremo quindi a modificare la X nella posizione corrispondente in known-positions con tale colore.

- right-placed della nuova guess == right-placed della best-guess: non possibile in quanto se faccio swap di un colore che era nella posizione corretta con un altro, right-placed può solo diminuire.

Per quanto riguarda l'ottenimento della soluzione, una volta raggiunto right-placed + misplaced = 4 il ragionamento è analogo a quanto fatto nel modulo Human, con la differenza che qualora dovessimo avere il campo known-positions di solution-information diverso da "X X X X" manteremmo fisso il/i colore/i che sappiamo essere in posizione corretta, andando quindi a permutare randomicamente solo gli altri colori. Anche in questo caso andremo successivamente a valutare per ogni nuovo tentativo di guess se è già stata fatto in precedenza al fine di evitare ripetizioni. La condizione di terminazione rimane sempre l'asserimento della guess soluzione o il raggiungimento del limite dei tentativi.

# Capitolo 3

## Differenze

A seguire abbiamo valutato le differenze tra le performance del modulo Human e del modulo Computer facendo cambiare randomicamente il secret-code e mantenendo la stessa guess iniziale, la quale viene poi modificata automaticamente dal programma per il raggiungimento della soluzione.

In entrambi i casi abbiamo valutato i risultati ottenuti dall'esecuzione del programma 100 volte.

Data from 100 random secret-codes and fixed guess (black white green red) at step 0

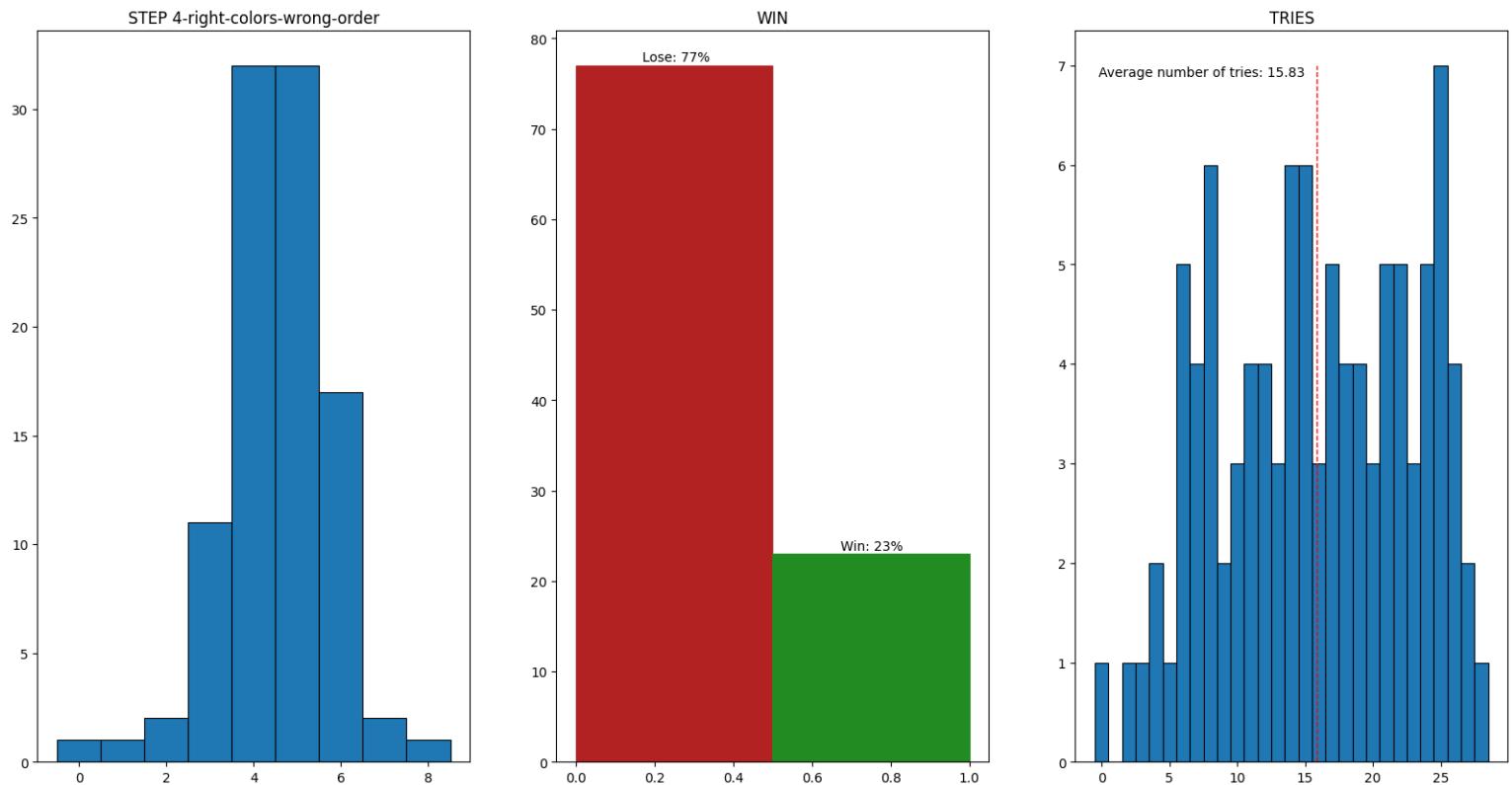


Figura 3.1: Performance modulo Human.

Data from 100 random secret-codes and fixed guess (black white green red) at step 0

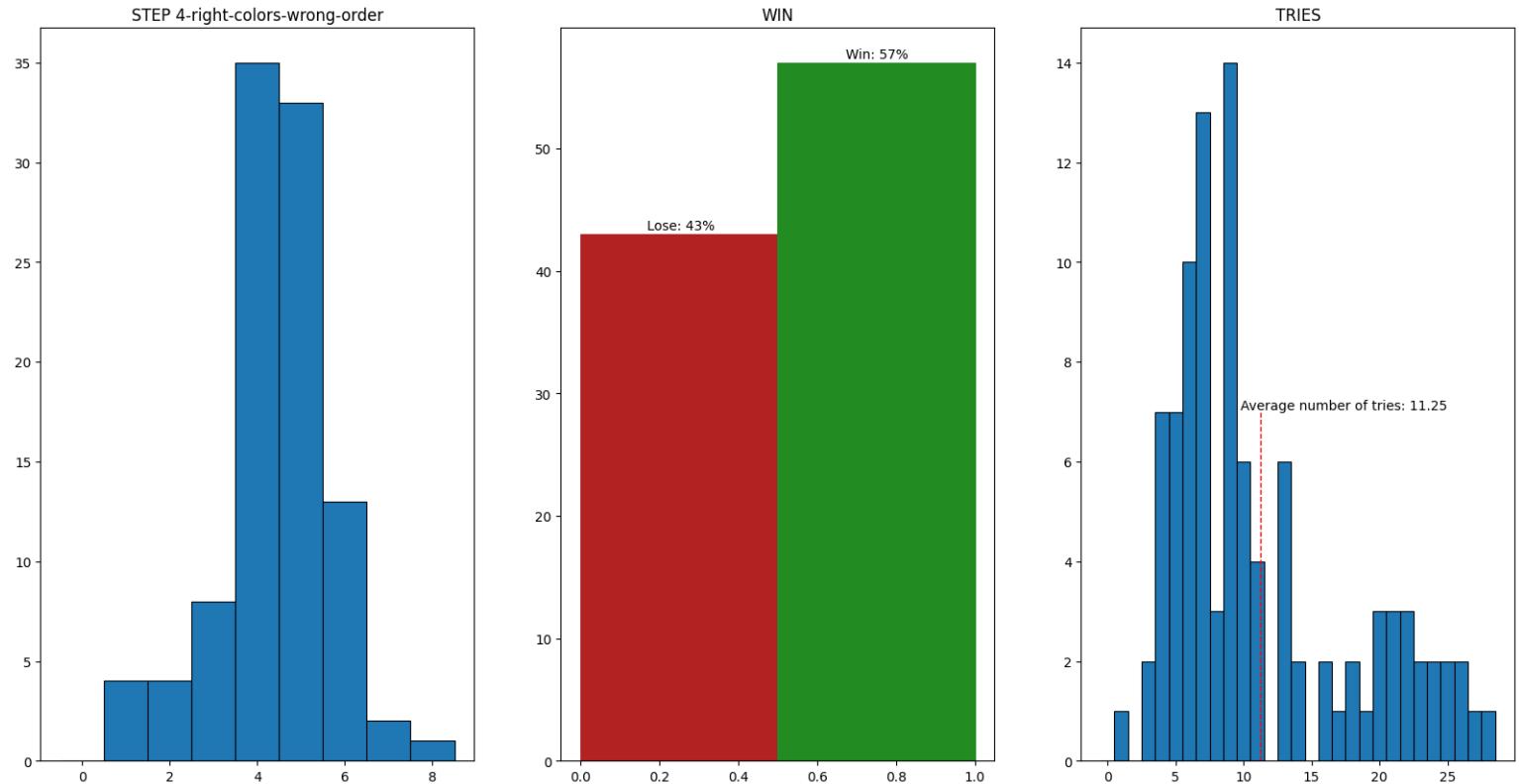


Figura 3.2: Performance modulo Computer.

Il vantaggio principale tra la versione Human e Computer sta nella definizione ed utilizzo del template solution-information e del suo multislot known-positions. In particolare a parità di step per raggiungere right-placed + misplaced = 4 abbiamo i seguenti cambiamenti nelle permutazioni randomiche:

- Caso Human (senza known-positions):  $4!-1 = 23$  permutazioni da valutare
- Caso Computer (known-positions con 1 colore):  $3!-1 = 5$  permutazioni da valutare
- Caso Computer (known-positions con 2 colori):  $2!-1 = 1$  permutazioni da valutare

I miglioramenti sono visibili confrontando i grafici relativi al win/lose rate e numero medio di tentativi per raggiungere la soluzione, notevolmente influenzati dall'aggiunta ed utilizzo del template solution-informations.