



Università di Salerno

Laurea Triennale in Informatica

Corso di Fondamenti di Intelligenza Artificiale

Professore F. Palomba

PokeTeam-OPT

<https://github.com/FrancescoPinto02/PokeTeam-OPT>

Pinto Francesco Alessandro

matricola: 0512113216

Anno Accademico 2023-2024

Indice

1	Introduzione	3
1.1	Introduzione ai Pokemon	3
1.1.1	Statistiche	3
1.1.2	Tipi	4
1.1.3	Evoluzioni	4
1.1.4	Varianti	5
1.1.5	Pokémon rari	5
1.1.6	Pokédex	5
1.2	Scopo del progetto	6
1.3	Specifica PEAS	6
1.3.1	Caratteristiche dell'ambiente	7
2	Analisi del problema	8
2.1	Algoritmi di ricerca Locale	8
2.1.1	Hill Climbing	8
2.1.2	Simulated Annealing	8
2.2	Alogoritmi Genetici	9
2.3	Considerazioni e scelta dell'algoritmo	9
3	Struttura del Progetto	10
3.1	Package ga	10
3.2	Package pokemon	11
3.3	PokeTeamOPTRunner	11
4	Algoritmo Genetico	12
4.1	Codifica degli individui	12
4.2	Popolazione	12
4.3	Funzione di Fitness	13

4.4	Selezione	14
4.5	Crossover	14
4.6	Mutazione	15
4.7	Stopping Condition	15
5	Risultati	16
6	Conclusioni e sviluppi futuri	18

1 Introduzione

I Pokémon hanno catturato l'immaginazione di milioni di persone in tutto il mondo sin dal loro debutto. Originariamente creati per un pubblico giovane, i Pokémon sono rapidamente diventati un fenomeno globale, influenzando la cultura pop, il marketing e persino la tecnologia attraverso videogiochi, serie TV, film, carte collezionabili, applicazioni mobile, ecc. Questo progetto mira ad esplorare e affrontare una delle sfide più complesse del mondo dei Pokemon, mediante l'utilizzo dell'intelligenza artificiale.

Nelle sottosezioni seguenti, verranno introdotti i concetti base relativi ai Pokémon, necessari per comprendere tutte le sfaccettature del progetto. Verrà inoltre descritto lo scopo principale del progetto e verrà fornita la specifica PEAS, includendo le caratteristiche dell'ambiente.

1.1 Introduzione ai Pokemon

Pokémon è un media franchise giapponese di proprietà della "The Pokémon Company", creato nel 1996 da Satoshi Tajiri. Il franchise nasce come una coppia di videogiochi sviluppati da Game Freak e pubblicati da Nintendo per la console portatile Game Boy. Il videogioco si basa su un mondo immaginario popolato da creature chiamate "Pokémon", che possono essere catturate, addestrate e inserite all'interno della propria squadra. I giocatori assumono il ruolo di Allenatori Pokémon e l'obiettivo principale è diventare il più abile allenatore, sconfiggendo sia altri giocatori che personaggi controllati dal computer (NPC). Le lotte Pokémon sono infatti il fulcro del videogioco.

Ogni Pokémon ha delle caratteristiche uniche che permettono di distinguerli dagli altri. Queste caratteristiche si concentrano principalmente su due aspetti fondamentali: le statistiche e i tipi.

1.1.1 Statistiche

Ogni Pokémon ha sei statistiche base: Health Point (HP), Attack (Atk), Defence (Def), Special Attack (Sp. Atk), Special Defence (Sp. Def) e Speed (Speed). La somma di queste statistiche va a formare il Totale (Tot). Le statistiche variano da un Pokémon all'altro e possono essere influenzate dalla loro specie e dalle abilità.

1.1.2 Tipi

I Pokémon sono associati a uno o al più a due tipi, come Fuoco, Acqua, Erba, Elettrico, ecc. Ogni tipo ha forze e debolezze contro altri tipi. Ad esempio, i Pokémon di tipo Acqua sono deboli contro i Pokémon di tipo Elettrico, ma sono efficaci contro quelli di tipo Fuoco.

Defender \ Attacker	Normal	Fire	Water	Grass	Electric	Ice	Fighting	Poison	Ground	Flying	Psychic	Bug	Rock	Ghost	Dragon	Dark	Steel	Fairy
Normal	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Fire	1	2	1/2	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Water	1	1/2	2	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Grass	1	1	1	2	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Electric	1	1	1	1	2	1	1	1	1	1	1	1	1	1	1	1	1	1
Ice	1	1	1	1	1	2	1	1	1	1	1	1	1	1	1	1	1	1
Fighting	1	1	1	1	1	1	2	1	1	1	1	1	1	1	1	1	1	1
Poison	1	1	1	1	1	1	1	2	1	1	1	1	1	1	1	1	1	1
Ground	1	1	1	1	1	1	1	1	2	1	1	1	1	1	1	1	1	1
Flying	1	1	1	1	1	1	1	1	1	2	1	1	1	1	1	1	1	1
Psychic	1	1	1	1	1	1	1	1	1	1	2	1	1	1	1	1	1	1
Bug	1	1	1	1	1	1	1	1	1	1	1	2	1	1	1	1	1	1
Rock	1	1	1	1	1	1	1	1	1	1	1	1	2	1	1	1	1	1
Ghost	1	1	1	1	1	1	1	1	1	1	1	1	1	2	1	1	1	1
Dragon	1	1	1	1	1	1	1	1	1	1	1	1	1	1	2	1	1	1
Dark	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	2	1	1
Steel	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	2	1
Fairy	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	2

Figura 1: Tipi Pokémon e relazioni

1.1.3 Evoluzioni

Molte specie di Pokémon hanno la capacità di evolversi in una o due forme più avanzate e solitamente più forti.

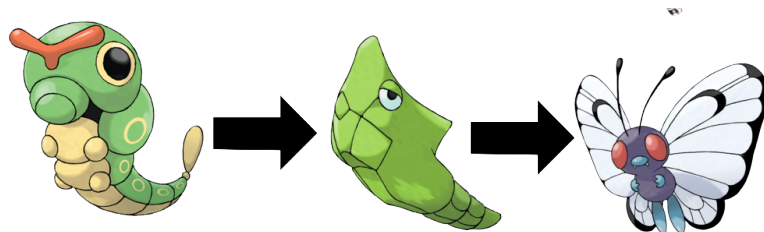


Figura 2: Evoluzione

1.1.4 Varianti

Alcuni Pokémon hanno delle varianti o forme alternative. È bene distinguere questo meccanismo da quello delle evoluzioni, in quanto le varianti sono forme alternative di uno stesso Pokémon, le quali possono avere tipi diversi, statistiche modificate e aspetti estetici unici



Figura 3: Variante

1.1.5 Pokémon rari

Alcuni Pokémon sono estremamente rari e unici nel loro genere da essere catalogati come Pokémon Leggendarî. Questa tipologia di Pokémon tendono ad avere statistiche base molto elevate, rendendoli formidabili in battaglia. La disponibilità dei Pokémon Leggendarî però è spesso limitata a eventi speciali nei giochi, in cui i giocatori hanno l'opportunità di catturarli o riceverli.

1.1.6 Pokédex

Nel gioco dei Pokémon, il Pokédex è un dispositivo elettronico o un'applicazione che funge da enciclopedia virtuale. All'interno del Pokédex vengono registrati i dati e le informazioni di tutti i Pokémon, indicizzati in base ad un numero.

È importante notare che le varie forme evolutive di un Pokémon sono associate a indici diversi (spesso consecutivi), mentre le forme alternative sono associate tutte allo stesso numero.

1.2 Scopo del progetto

La strategia di gioco, particolarmente evidente nelle lotte Pokémon, è diventata con il passare degli anni sempre più complessa e sfumata, evolvendosi in un'attività che richiede calcolo e previsione.

Con il primo videogioco uscito nel 1996, il Pokédex registrava 151 specie diverse di Pokémon. Considerando che i Team di Pokémon possono essere composti da massimo 6 membri, le possibili combinazioni erano $C(151, 6) = \frac{151!}{6!(151-6)!} = 14,888,600,755$. Quindi già all'epoca, costruire un team era un compito abbastanza arduo. Tuttavia, eliminando qualche Pokémon particolarmente debole o le forme evolutive intermedie, il numero di Pokémon che aveva senso considerare calava drasticamente, rendendo il tutto più semplice.

Con il videogioco uscito più di recente nel 2022, il numero di specie diverse registrate dal Pokédex è arrivato a 1010 (senza considerare le varianti), e facendo qualche calcolo, le possibili combinazioni sono diventate $C(1010, 6) = \frac{1010!}{6!(1010-6)!}$. Pur adottando la stessa strategia utilizzata nei primi videogiochi, il numero di Pokémon che ha senso considerare continua ad essere un numero molto elevato.

Per coloro che, come il sottoscritto, aspirano sempre alla perfezione, la ricerca del team perfetto si rivela un compito impegnativo che richiede notevole dedizione. L'obiettivo di PokeTeam-OPT è quello di semplificare questo processo, trovando una soluzione al problema tramite l'intelligenza artificiale. Tuttavia, è importante precisare che lo scopo di PokeTeam-OPT non è quello di creare team ottimizzati per il gioco competitivo (non ancora), poichè ciò implicherebbe considerazioni che renderebbero il problema ancora più complesso. L'obiettivo primario è piuttosto quello di creare dei team potenti e diversificati che offrano un'esperienza di gioco divertente e permettano di completare i vari videogiochi senza troppe difficoltà.

1.3 Specifica PEAS

Di seguito è riportata la descrizione PEAS:

- **Performance:** le performance dell'agente sono valutate in base alle statistiche totali medie, la diversità dei tipi, le resistenze e le debolezze del team.
- **Environment:** l'ambiente in cui opera l'agente è costituito dall'insieme di tutte le possibili squadre che si possono comporre con i Pokémon all'interno del Pokédex.

- **Actuators:** l'agente agisce sull'ambiente selezionando i Pokémon dal Pokédex per comporre e modificare le squadre.
- **Sensors:** l'agente acquisisce dall'ambiente tutti i dati relativi ai Pokémon che compongono le varie squadre.

1.3.1 Caratteristiche dell'ambiente

Di seguito sono riportate le caratteristiche dell'ambiente:

- **Completamente Osservabile:** i sensori dell'agente gli danno accesso allo stato completo dell'ambiente in ogni momento.
- **Singolo Agente:** c'è un solo agente intelligente che opera sull'ambiente.
- **Statico:** l'ambiente non cambia mentre l'agente prende decisioni.
- **Stocastico:** è presente una componente randomica che determina lo stato successivo.
- **Discreto:** ci sono un numero finito e distinto di possibili azioni e stati.
- **Sequenziale:** le decisioni dell'agente (come selezionare o cambiare un Pokémon) influenzano le decisioni successive e le prestazioni complessive del team.

2 Analisi del problema

La seguente sezione si occupa di analizzare la natura del problema posto da PokeTeam-OPT e di valutare diverse strategie di risoluzione. Il problema centrale consiste nel determinare la composizione ottimale di un team Pokémon, tenendo conto di una serie di parametri chiave come le statistiche complessive, le resistenze, la diversità di tipi e le debolezze in comune. Come è facile intuire, si tratta di un problema di ottimizzazione, in cui si deve comporre una squadra utilizzando i Pokémon a disposizione in modo tale da massimizzare una o più funzioni obiettivo. Tra gli algoritmi più adatti a risolvere questa tipologia di problemi ci sono gli Algoritmi di Ricerca Locale e gli Algoritmi Genetici.

2.1 Algoritmi di ricerca Locale

Gli algoritmi di ricerca locale sono algoritmi utilizzati per risolvere problemi di ottimizzazione combinatoria. Questa classe di algoritmi opera attraverso la modifica iterativa di una singola soluzione candidata, con lo scopo di migliorare la qualità della soluzione rispetto ad una funzione obiettivo, tipicamente attraverso l'esplorazione delle soluzioni vicine alla soluzione corrente.

2.1.1 Hill Climbing

L'algoritmo di ricerca Hill Climbing tiene traccia di un solo stato corrente e ad ogni iterazione passa allo stato vicino con valore più alto. Esistono diverse varianti di questo algoritmo, ma nella sua versione classica, esso si fermerà una volta raggiunto un qualsiasi punto di massimo locale.

2.1.2 Simulated Annealing

Il nome e il comportamento del Simulated Annealing derivano dall'annealing, ovvero un processo metallurgico che consiste nell'indurire i metalli riscaldandoli ad altissime temperature e raffreddandoli gradualmente. L'algoritmo Simulated Annealing si propone di effettuare una ricerca simile all'hill climbing ma scendendo talvolta anche verso stati con valore inferiore. Inizialmente l'algoritmo esplorerà il panorama degli stati in modo molto casuale, per poi concentrarsi man mano su una regione specifica più "interessante" e cercando una soluzione ottima in quel contesto. In altre parole l'algoritmo tenderà ad accettare con una certa probabilità anche soluzioni peggiori rispetto alla soluzione corrente, e tale probabilità decresce man mano che la ricerca si stabilizza.

2.2 Algoritmi Genetici

Gli algoritmi genetici fanno parte della categoria più ampia degli algoritmi evolutivi, ovvero algoritmi di ricerca/ottimizzazione ispirati dai processi della natura. Con il termine algoritmi genetici ci si riferisce ad una metaeuristica per definire algoritmi di ricerca ispirati alla genetica. Gli algoritmi genetici consistono nel far evolvere una popolazione di individui producendo di volta in volta soluzioni sempre migliori rispetto ad una funzione obiettivo, fino a raggiungere l'ottimo o un'altra condizione di terminazione. La creazione di nuove generazioni avviene applicando gli operatori genetici di selezione, crossover e mutazione.

2.3 Considerazioni e scelta dell'algoritmo

Analizzando più attentamente il problema, si evince facilmente che presenta un numero considerevole di massimi locali, e il principale svantaggio degli algoritmi di ricerca locale è proprio la loro tendenza a rimanere intrappolati in questi punti. Una possibile soluzione a questo problema potrebbe essere rappresentata dall'uso di algoritmi più sofisticati, come il Simulated Annealing, che incorporano meccanismi per evitare tali trappole. Tuttavia, la scelta definitiva è ricaduta sugli algoritmi genetici, noti per la loro efficacia nell'esplorare vasti spazi di ricerca grazie ai meccanismi di crossover e mutazione. La preferenza per gli algoritmi genetici è stata motivata anche da una predilezione personale verso un approccio di progettazione più intuitivo e interessante.

3 Struttura del Progetto

La seguente sezione è dedicata a fornire una panoramica dell'implementazione del progetto PokeTeam-OPT. L'obiettivo è di delineare i vari componenti software utilizzati, spiegando brevemente il ruolo e la funzionalità di ciascun package, nonché le tecnologie e gli strumenti impiegati durante lo sviluppo.

PokeTeam-OPT è stato implementato utilizzando esclusivamente Java (JDK 17), mentre per gestire efficacemente i dati relativi ai Pokémon e ai loro tipi, che sono fondamentali per il funzionamento dell'algoritmo, sono stati utilizzati dei dataset in formato JSON. Per facilitare la lettura e la manipolazione dei file JSON, è stata utilizzata la libreria JSON Simple, la quale è stata scelta per la sua facilità d'uso e la sua efficienza nell'analisi e nella gestione dei dati in formato JSON.

3.1 Package ga

Il package 'ga' costituisce il cuore dell'implementazione dell'algoritmo genetico usato per ottimizzare i team di Pokémon. All'interno di questo package, si trovano vari sub-packages che contengono tutti gli operatori genetici e le classi necessarie per il funzionamento dell'algoritmo. Le classi astratte impiegate sono state adattate dagli esempi forniti sulla piattaforma e-learning del corso.

Lista dei sub-packages:

- **individuals:** contiene le classi per la rappresentazione e la codifica degli individui.
- **population:** contiene le classi per la gestione della popolazione.
- **initializer:** contiene le classi per l'inizializzazione della popolazione.
- **fitness:** contiene le classi relative alla funzione fitness.
- **operators:** contiene le classi che rappresentano gli operatori genetici di selezione, crossover e mutazione.
- **metaheuristics:** contiene le classi che rappresentano le metaeuristiche utilizzate per guidare il processo di ricerca e ottimizzazione.
- **results:** contiene la classe utilizzata per la gestione e l'analisi dei risultati.

3.2 Package pokemon

Il package 'pokemon' contiene tutte le classi necessarie a gestire le funzionalità strettamente legate ai Pokémon stessi. È divisa nei seguenti sub-packages:

- **type:** contiene le classi e le enum necessarie a rappresentare i tipi pokemon e le relazioni tra essi (debolezze e resistenze). Questo package contiene anche il dataset JSON con tutte le informazioni necessarie per il types pool.
- **core:** contiene le classi e le enum necessarie a rappresentare i pokemon e la loro rarità. Inoltre contiene la classe per generare pokemon casuali.
- **pokedex:** contiene la classe per rappresentare il pokedex e il dataset JSON da cui vengono caricati i dati relativi ai vari pokemon. È importante notare che il pokedex è stato implementato utilizzando un hash map, così da garantire una maggiore velocità d'accesso.
- **team:** contiene la classe per generare delle squadre di Pokémon casuali.

3.3 PokeTeamOPTRunner

PokeTeamOPTRunner.java rappresenta il file da eseguire per avviare l'algoritmo. All'interno del metodo main potranno essere impostati i vari parametri dell'algoritmo, quali il numero di individui della popolazione, la probabilità di mutazione, il numero massimo di iterazioni e il numero massimo di iterazioni senza miglioramenti. Inoltre si potranno scegliere anche vari algoritmi di selezione e/o crossover. I risultati di ogni iterazione dell'algoritmo e il risultato finale verranno mostrati nella console.

NB: Per l'algoritmo di selezione K-Tournament si potrà impostare la dimensione dei tornei e si potrà scegliere di decretare il vincitore utilizzando un criterio proporzionale (simile alla Roulette Wheel).

4 Algoritmo Genetico

L'Algoritmo Genetico costituisce la colonna portante di PokeTeam-OPT, fornendo il meccanismo essenziale per l'ottimizzazione dei team di Pokémon. In questa sezione, verranno spiegate nel dettaglio le strategie adottate in termini di operatori genetici, come sono stati configurati i parametri ed altri aspetti rilevanti dell'algoritmo genetico.

4.1 Codifica degli individui

La codifica degli individui è un aspetto cruciale in ogni algoritmo genetico. Nel contesto di PokeTeam-OPT, un individuo è rappresentato da un team composto da sei Pokémon. Questi individui sono codificati tramite un array di oggetti appartenenti alla classe Pokémon, che rappresentano i geni dell'individuo. Questo approccio, comunemente noto come 'codifica basata su oggetti', sfrutta efficacemente le caratteristiche di un linguaggio orientato agli oggetti come Java. Tutto questo, rende la rappresentazione degli individui più naturale e intuitiva, facilitandone la manipolazione e l'accesso alle loro caratteristiche.

4.2 Popolazione

La creazione e la gestione della popolazione iniziale sono aspetti fondamentali in un algoritmo genetico. In PokeTeam-OPT, viene utilizzata una popolazione di dimensione fissa, ciò significa che il numero di individui all'interno della popolazione rimane costante in ogni iterazione. Per determinare la dimensione ottimale della popolazione, sono stati condotti esperimenti con 30, 50, 80 e 100 individui, riscontrando i migliori risultati con una popolazione di 100 individui. Sebbene 100 possa sembrare un numero elevato di individui, ciò non incide significativamente sulla memoria del sistema. Questo è dovuto al fatto che ogni Pokémon esistente viene istanziato una sola volta durante la fase di inizializzazione del Pokédex, pertanto i geni di ogni individuo sono semplicemente riferimenti a tali Pokémon.

La popolazione iniziale viene generata casualmente attraverso la classe PokemonTeamGenerator, la quale compone le squadre selezionando in modo casuale Pokémon dal Pokédex. Questa strategia di generazione assicura un grado di diversità adeguato tra gli individui della popolazione.

4.3 Funzione di Fitness

La funzione di fitness è il criterio secondo cui viene valutata la bontà dei diversi individui. In PokeTeam-OPT si utilizza una singola funzione di fitness che è effettivamente una combinazione lineare di diverse sotto-funzioni di fitness. Il valore di fitness complessivo viene calcolato usando una formula di questo tipo:

$$Fitness = w1 * f1 + w2 * f2 + ... + wi * fi$$

Di seguito vengono riportati i dettagli delle varie sotto-funzioni di fitness:

- **Average Team Stats:** Vengono calcolate le statistiche totali medie dell'intero team. Ai team con una media più elevata vengono assegnati valori di fitness superiori.
- **Types Diversity:** Viene calcolato il numero di tipi distinti presenti all'interno del team. Ai team che utilizzano un maggior numero di Pokémon con tipi diversi tra loro, vengono attribuiti valori di fitness superiori.
- **Team Resistances:** Si calcola il numero di resistenze del team rispetto ai diversi tipi di Pokémon. Ai team con una maggiore copertura (numero di resistenze) viene assegnato un valore di fitness superiore.
- **Common Weaknesses:** Si calcola il numero di debolezze in comune tra i Pokémon all'interno del team. Ai team composti da Pokémon con meno debolezze in comune viene assegnato un valore di fitness superiore.
- **Legendary Count:** Viene calcolato il numero di Pokémon particolarmente rari (legendari, mistici, sub-legendari e paradosso) presenti nel team. Ai team composti da un minor numero di Pokémon rari viene assegnato un valore di fitness superiore. È importante sottolineare che i Pokémon rari rappresentano degli outlier, poiché possiedono statistiche significativamente superiori alla media. Questa sotto-funzione di fitness è stata concepita per gestire questi casi particolari, infatti impedisce che l'algoritmo tenda a convergere verso individui formati esclusivamente da tali Pokémon pur consentendo al contempo il loro utilizzo nella costruzione dei team.

Poiché le varie sotto-funzioni di fitness restituiscono valori che variano notevolmente in termini di range, per prevenire che l'algoritmo attribuisca maggior importanza ad alcune di esse rispetto ad altre, i valori restituiti vengono normalizzati prima di essere combinati nella combinazione lineare. La normalizzazione viene effettuata utilizzando una min-max normalization:

$$y = a + \frac{(x - \min(x))(b - a)}{\max(x) - \min(x)}$$

4.4 Selezione

L'operatore genetico di Selezione rappresenta il meccanismo secondo il quale gli individui più promettenti della popolazione vengono selezionati per formare il mating pool. In PokeTeam-OPT ad ogni iterazione vengono selezionati esattamente 100 individui (un individuo può essere anche selezionato più volte). Sono state provate varie strategie di selezione:

- **Roulette Wheel:** Ogni individuo può essere selezionato con una probabilità proporzionale al suo punteggio di fitness relativo a l'intera popolazione.
- **K-Tournament:** Vengono selezionati K individui casualmente per formare un torneo e il migliore di questi K viene decretato vincitore. Si ripete il processo finché non si arriva ad un certo numero M di vincitori. Nel caso specifico di PokeTeam-OPT, M è pari a 100.
- **K-Tournament Proporzionale:** Uguale al K-Tournament classico ma il vincitore di ogni torneo viene selezionato con un criterio simile alla Roulette Wheel Selection.

Tra i diversi algoritmi di selezione testati, i migliori risultati sono stati ottenuti con la K-Tournament selection classica, impostando $K=5$. Questo metodo ha dimostrato prestazioni nettamente superiori rispetto agli altri approcci testati.

4.5 Crossover

Il crossover è il meccanismo mediante il quale i geni di due team di Pokémon vengono combinati per generare nuove soluzioni. In PokeTeam-OPT, vengono formate casualmente delle coppie selezionando gli individui presenti nel mating pool, e ogni coppia produce due nuovi individui. Anche in questo contesto, sono state testate diverse strategie di crossover:

- **Single Point:** viene scelto un punto casuale lungo l'array dei geni dei genitori, e i segmenti di array prima di questo punto vengono scambiati tra i due genitori per creare due nuovi individui.
- **Double Point:** si selezionano due punti casuali sull'array dei geni dei genitori. I segmenti di array tra questi due punti vengono scambiati tra i due genitori per formare due nuovi individui,
- **Uniform:** ogni gene dei nuovi individui viene selezionato casualmente da uno dei due genitori.

Tra le varie tecniche di crossover testate, si è scelto di utilizzare lo Uniform Crossover per introdurre una maggiore diversità nella popolazione e per esplorare uno spazio di ricerca più ampio.

4.6 Mutazione

La mutazione introduce variazioni casuali nei team di Pokémon, elemento essenziale per esplorare nuove soluzioni e mantenere la diversità genetica. In PokeTeam-OPT, è stato implementato un algoritmo di Mutazione Uniforme, in cui un Pokémon all'interno del team viene scelto casualmente e sostituito con un altro Pokémon selezionato a caso dal Pokédex. Dopo varie prove, la probabilità di mutazione è stata fissata al 30%, quindi ogni offspring (individuo della nuova generazione) avrà una probabilità del 30% di subire una mutazione.

4.7 Stopping Condition

Per determinare il criterio di arresto più efficace per l'algoritmo, sono state condotte diverse esecuzioni complete e i risultati sono stati analizzati. Dall'analisi è emerso che si verifica un picco di miglioramento nelle prime 15 iterazioni, il quale diventa meno significativo dopo aver raggiunto le 20 iterazioni. Oltre le 30 iterazioni, il miglioramento diventa quasi impercettibile. Pertanto, come criterio di arresto è stato stabilito un limite massimo di 35 iterazioni.

5 Risultati

Per valutare i risultati dell'algoritmo, sono state avviate una serie di esecuzioni, e gli individui generati al termine di queste sono stati analizzati tramite un'applicazione esterna. Di seguito, viene presentato un esempio di questi risultati con la relativa analisi.

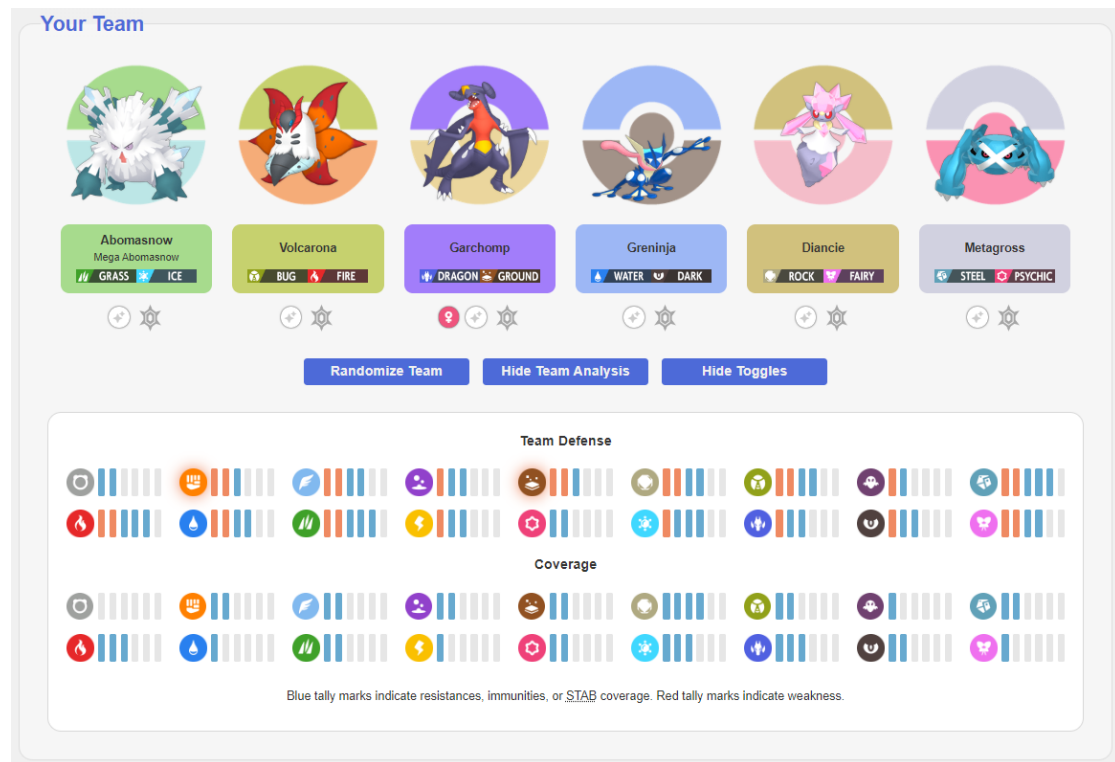


Figura 4: risultato

Analisi del team:

- **Average Stats:** il punteggio totale medio delle statistiche è pari a 585 su un limite di 600.
- **Types Diversity:** Il team è composto esclusivamente da Pokémon di tipo doppio, e ogni tipo è diverso dagli altri. Il numero totale di tipi differenti all'interno del team è 12, il che rappresenta il massimo possibile.
- **Team Resistances:** Come si può osservare dall'immagine, il team è in grado di resistere a ogni tipo di Pokémon esistente, come indicato dai rettangoli azzurri che rappresentano le resistenze.

- **Common Weaknesses:** Come si può notare dall'immagine, il numero di debolezze in comune tra i Pokémon del team è molto contenuto. Infatti, non ci sono più di due Pokémon vulnerabili allo stesso tipo, come indicato dai rettangolini arancioni che rappresentano le debolezze.
- **Legendary Count:** nel team è presente un solo pokemon raro (Diancie).

Dall'analisi risulta che l'individuo generato dall'algoritmo ha un punteggio di fitness molto prossimo al limite teorico. Sebbene il numero di debolezze in comune potrebbe teoricamente essere ridotto, è importante considerare che alcuni obiettivi sono in una certa misura contraddittori. Per ottenere una maggiore copertura, è infatti necessario includere Pokémon con tipi diversi, il che potrebbe aumentare anche il numero di debolezze. Tenendo conto di questo aspetto, i risultati raggiunti dall'algoritmo sono eccellenti. Inoltre, il team è stato composto utilizzando un solo Pokémon raro, il che è in perfetta sintonia con gli obiettivi di PokeTeam-OPT.

6 Conclusioni e sviluppi futuri

In conclusione, il progetto PokeTeam-OPT ha dimostrato l'efficacia dell'uso degli algoritmi genetici nell'ottimizzazione dei team di Pokémon. Grazie a una configurazione accurata dei parametri e degli operatori genetici, PokeTeam-OPT non solo riesce a creare team potenti e diversificati, ma lo fa in un modo che riflette le complessità e le sfide strategiche del gioco Pokémon.

Nonostante i successi, esistono ancora alcune limitazioni legate all'efficacia e all'usabilità pratica dell'algoritmo. Ad esempio, mantenere l'equilibrio tra i diversi obiettivi di fitness rimane una sfida, e c'è margine per ulteriori perfezionamenti nell'algoritmo. Inoltre, il problema affrontato da PokeTeam-OPT rappresenta solo una semplificazione della realtà, in quanto molti altri aspetti legati ai Pokémon non vengono considerati, tra cui:

- **Mosse:** Ogni Pokemon può avere 4 mosse, alle quali è associato un tipo che non necessariamente corrisponde al tipo del Pokemon stesso.
- **Natura:** Ogni Pokémon ha una natura che aumenta di un 10% una statistica e ne diminuisce di un 10% un'altra.
- **Abilità:** Ogni Pokémon ha un'abilità che può influenzare le sue resistenze e debolezze.
- **Strumenti:** Ogni Pokémon può portare uno strumento che ha diversi effetti in battaglia.

Considerando tutti questi aspetti, l'algoritmo diventerebbe molto più applicabile nella realtà, ma ciò richiederebbe ulteriori dataset contenenti tutte le informazioni necessarie per raggiungere questi obiettivi aggiuntivi.

Creare un algoritmo in grado di generare team di Pokémon perfetti, anche per il gioco competitivo, rappresenta una sfida notevole, anche per chi ha una maggiore esperienza del sottoscritto. Pertanto, questo obiettivo rimane un ambito di ricerca aperto, ricco di potenziali sviluppi futuri. PokeTeam-OPT rappresenta un passo significativo in questa direzione, offrendo una base su cui costruire e migliorare per avvicinarsi sempre di più a questo ambizioso traguardo.

Riferimenti bibliografici

- [1] S. Russel, P. Norvig *Intelligenza artificiale - un approccio moderno (Vol.1)*.
- [2] [Pokemondb](#)
- [3] [Bulbapedia](#)