

6.5 Analisi comparativa delle performance

In questa sezione verrà presentato uno schema riassuntivo dei risultati ottenuti nella fase sperimentale, al fine di poter effettuare una comparazione tra le performance registrate dalle varie strategie adottate.

Per lo scenario *Illegal Dump* sono state adattate entrambe le strategie di esplorazione introdotte nei capitoli precedenti e sono state confrontate con i risultati di altri studi presenti in letteratura. La tabella 6.20 mostra i risultati degli esperimenti realizzati.

Algorithm	Performance (tick)	Approx. Optimization duration
SFE	121.70 \pm 4.75	90 <i>h</i>
SFE-RR1	107.40 \pm 3.10	6 <i>h</i>
ACO-E	161.34 \pm 6.21	36 <i>h</i>

Tabella 6.20: Strategia di esplorazione dello scenario *Illegal Dump*, analisi comparativa delle performance.

Come si evince da tali risultati sullo scenario *Illegal Dump*, il nuovo approccio supera sensibilmente sia le performance ottenute in [22], sia l'adattamento dell'algoritmo implementato in [17], accuratamente integrato nel simulation testbed. SFE-RR1, infatti, migliora le performance di missione di circa il 12% rispetto a SFE e del 25% rispetto ad ACO-E.

Per quanto riguarda le strategie di reclutamento, invece, i risultati prodotti dalle varie esecuzioni del simulatore sono riassunti nella tabella 6.21.

Algorithm	Performance (tick)
SFE-RR5	220.20 \pm 15.13
ACO-ABC-RR5-E	275.28 \pm 3.87

Tabella 6.21: Strategia di reclutamento nello scenario *Illegal Dump*, analisi comparativa delle performance.

Anche in questo caso, le metaeuristiche utilizzate nell'approccio SFE-RR hanno prodotto risultati interessanti, soprattutto in relazione a quanto ottenuto dall'integrazione, adattamento e simulazione della strategia ACO-ABC-RR5-E. Le performan-

ce della strategia di esplorazione e reclutamento SFE-RR5, infatti, risulta migliore dell'approccio ACO-ABC-RR5-E di circa il 20%.

Per gli scenari *Rural Mine* e *Urban Mine* si è scelto di adattare solo la strategia di esplorazione SFE-RR1, al fine di testarne le performance su scenari di diverse conformazioni. Anche in questo caso si è scelto di valutare le performance del simulatore attraverso un confronto con le performance registrate, sugli stessi scenari, da SFE. Le tabelle 6.22 e 6.23 mostrano i risultati degli esperimenti realizzati.

Algorithm	Performance (tick)	Approx. Optimization duration
SFE	125.96 \pm 8.90	90 <i>h</i>
SFE-RR1	120.32 \pm 4.91	8 <i>h</i>

Tabella 6.22: Strategia di esplorazione dello scenario *Rural Mine*, analisi comparativa delle performance.

Algorithm	Performance (tick)	Approx. Optimization duration
SFE	152.38 \pm 5.25	90 <i>h</i>
SFE-RR1	160.34 \pm 5.42	10 <i>h</i>

Tabella 6.23: Strategia di esplorazione dello scenario *Urban Mine*, analisi comparativa delle performance.

Nei due scenari riguardanti le missioni di sminamento, la strategia di SFE-RR1 è riuscita a raggiungere performance simili a quanto già prodotto in SFE. È fondamentale ricordare, ad ogni modo, che il nuovo meccanismo di ottimizzazione parametrica prevede un'esecuzione parallelizzata dell'algoritmo di Differential Evolution. Questo si traduce in un costo molto minore in termini di tempo necessario al completamento del task, rispetto all'esecuzione dello stesso algoritmo in Matlab, come si evince dalla durata, indicata in tabella, delle ottimizzazioni. Il miglioramento di tali prestazioni, infatti, è pari a circa il 90%, in entrambi gli scenari.

6.6 Conclusioni

Il documento riassume la progettazione di ambiente di simulazione per l'integrazione di approcci bio-ispirati per il coordinamento degli UAV coinvolti nel rilevamento e nel monitoraggio dei target distribuiti. La logica di coordinamento comprende un'autoformazione spaziale e una collaborazione basata su comunicazioni dirette e indirette tra gli attori della missione. Lo sciame, inoltre, adatta i suoi parametri alla missione specifica utilizzando un algoritmo di ottimizzazione basato su evoluzione differenziale.

Gli esperimenti sono incoraggianti, in quanto l'approccio proposto si estende e, in alcuni casi, supera approcci simili in letteratura. La possibilità di parallelizzare i calcoli eseguiti dall'algoritmo di Differential Evolution, infatti, si è dimostrato molto più efficiente in termini di performance temporali. La durata di una completa esecuzione del meccanismo di ottimizzazione, infatti, passa dalle 90 ore circa ad appena 6-10 ore, con un risultato simile o migliore in termini di prestazioni dello sciame.

Un ulteriore miglioramento delle prestazioni si potrebbe ottenere dallo studio degli iperparametri del DE: i parametri utilizzati in fase sperimentale, infatti, vanno ad influenzare il cammino all'interno dell'iperspazio definito dagli intervalli dei parametri da ottimizzare. Andando ad abbassare i parametri F e CR , infatti, potremmo sì rallentare la convergenza, ma anche esplorare in modo migliore l'iperspazio alla ricerca del minimo globale o, comunque, di un punto ad esso vicino. Le performance garantite da questo nuovo approccio parallelizzato, infatti, ci danno la possibilità di allargare gli intervalli in cui spaziare e di avere, in ogni caso, esecuzioni con durata inferiore alle 24h.

Il lavoro futuro potrebbe, dunque, consistere nell'approfondire le funzionalità adattive del simulatore ed integrare nuovi scenari e nuove strategie per fornire nuovi risultati comparativi, al fine di validare le caratteristiche strutturali dell'ambiente di simulazione.