UNIVERSITÀ DI TRENTO
Dipartimento di Ingegneria e Scienza dell'Informazione
Corso di Laurea in Informatica

MULTI-AGENT PATH FINDING

Algoritmi allo Stato dell'Arte a Confronto su Scenari Reali

RELATORI:

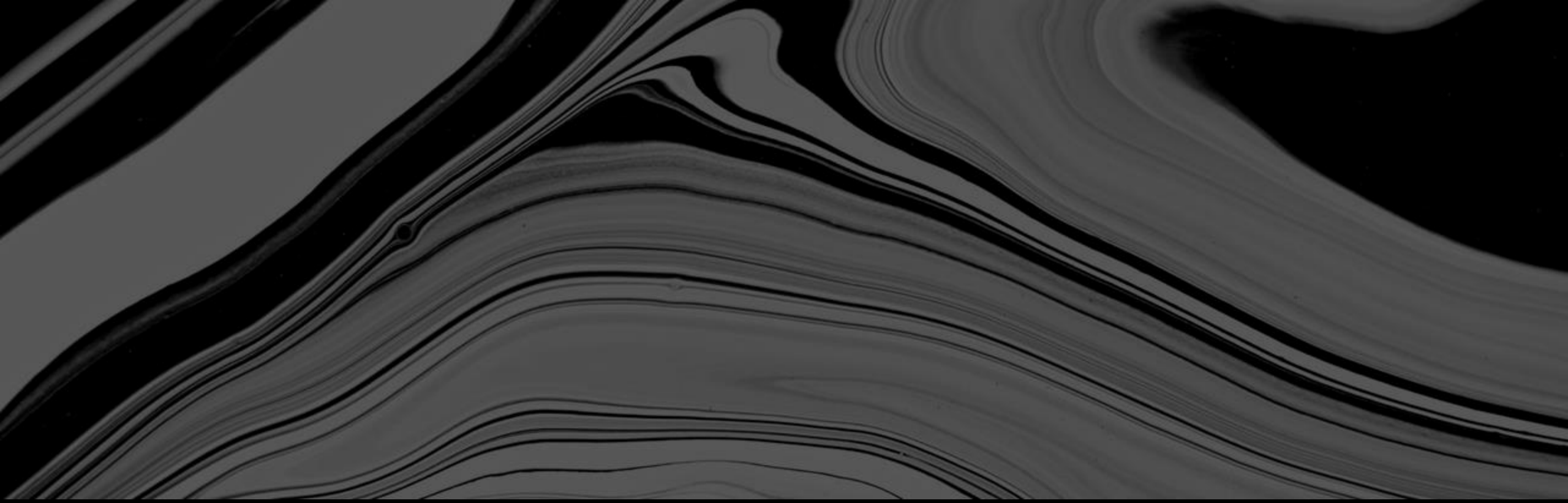
Prof. Marco Roveri

Dr. Enrico Saccon

CANDIDATO:

Alessandro Sartore

INDICE



SEZIONE 1: INTRODUZIONE

INTRODUZIONE AL MULTI AGENT PATH FINDING (MAPF)

DEFINIZIONE

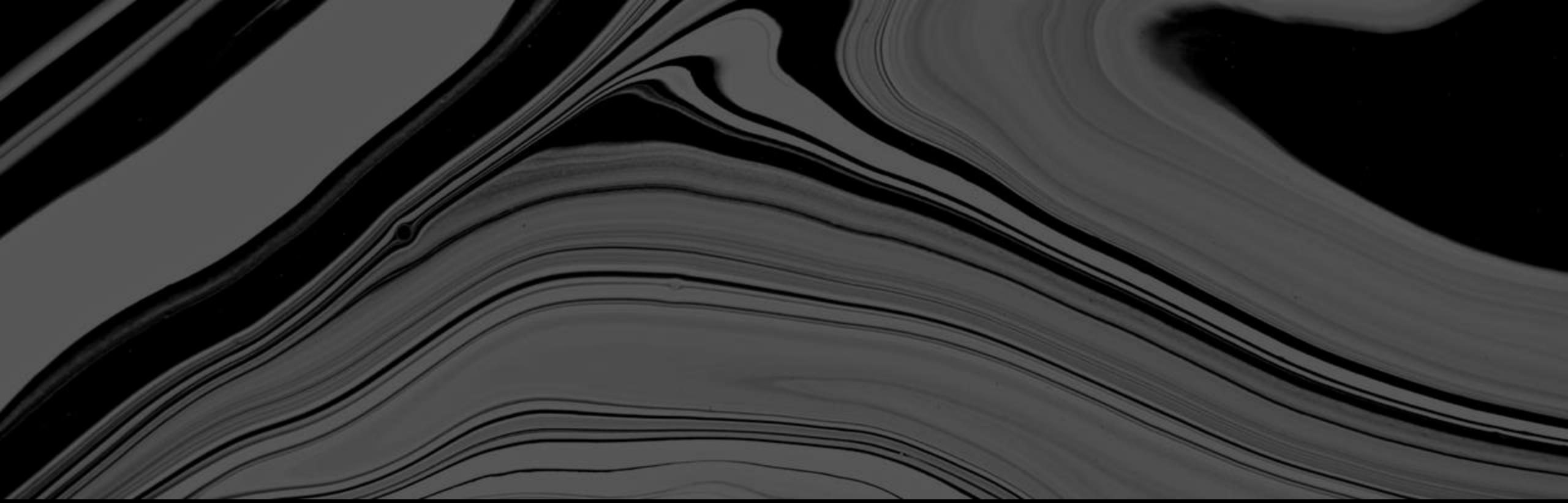
- Trovare percorsi per più agenti verso i loro obiettivi evitando collisioni
- Versione più complessa del Single-Agent Path Finding (SAPF)

COMPLESSITÀ

- Aumenta con il numero di agenti
- Ostacoli e dimensioni rendono il problema più difficile da risolvere

ALGORITMI TESTATI

- CBS
- ICR
- ICTS
- ICTS + ID
- X*



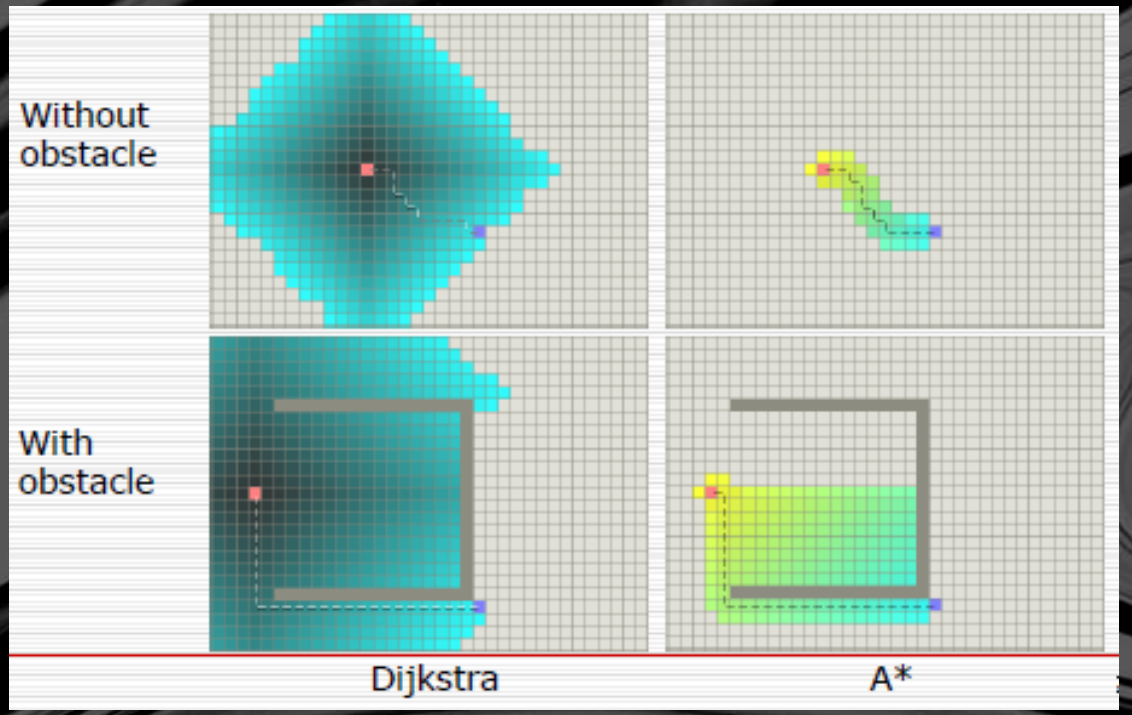
SEZIONE 2: PROBLEMA

DIJKSTRA E A*

- SAPF è alla base dei problemi più complessi di MAPF

DIFFERENZE:

- Tipologia di ricerca
- Efficienza
- Complessità



https://devforum.roblox.com/t/how-does-apathfinding-work/271356

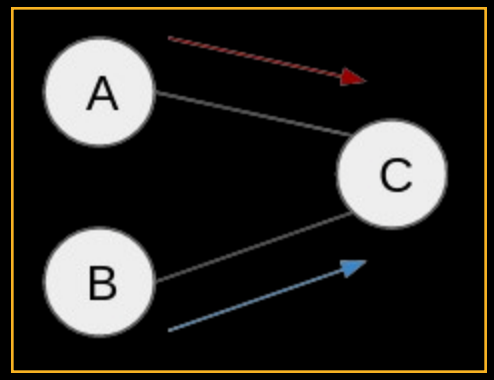
PROBLEMA CLASSICO MAPF

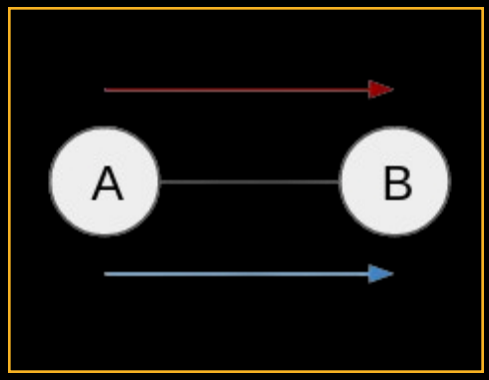
PROBLEMA

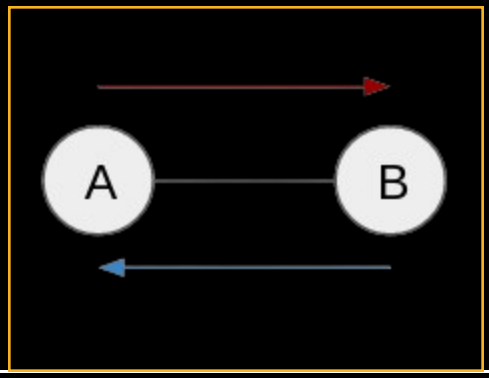
L'input con *k* agenti è una tupla <*G*, *s*, *t*> tale che:

- G = (V, E) è un grafo non orientato
- $s : [1, \ldots, k] \rightarrow V$
- $t : [1, \ldots, k] \rightarrow V$
- Il tempo è discreto
- Azioni possibili: *move* o wait

TIPOLOGIE DI CONFLITTO PIÙ COMUNI

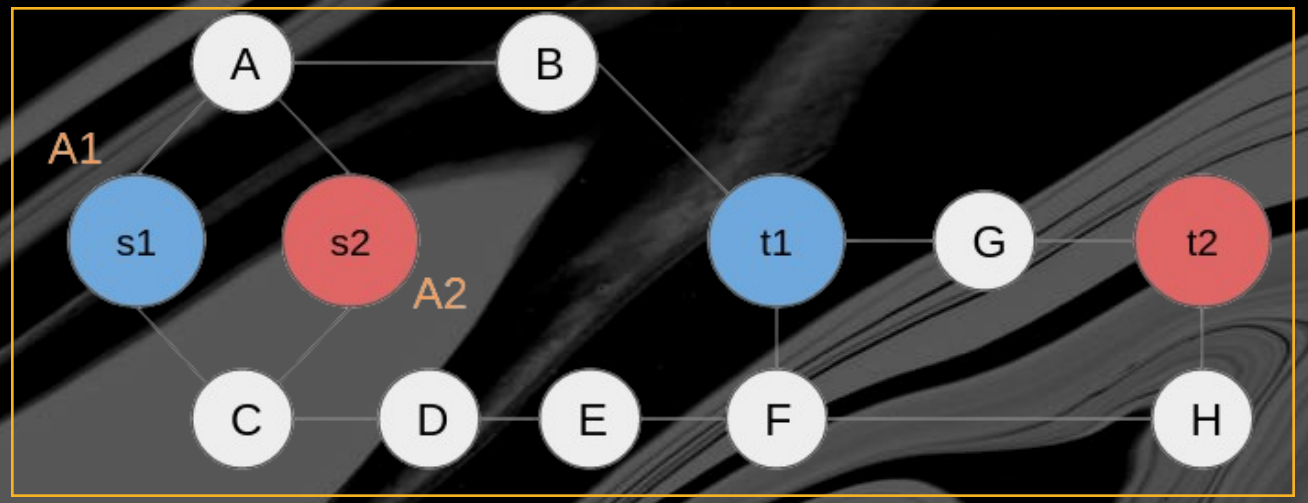






FUNZIONI OBIETTIVO

- **SUM OF INDIVIDUAL COSTS (SIC).** Formalmente $C_{SIC} = \Sigma_i(C(\pi_i))$
- **MAKESPAN (MKS).** Formalmente $C_{MKS} = \max_i(C(\pi_i))$

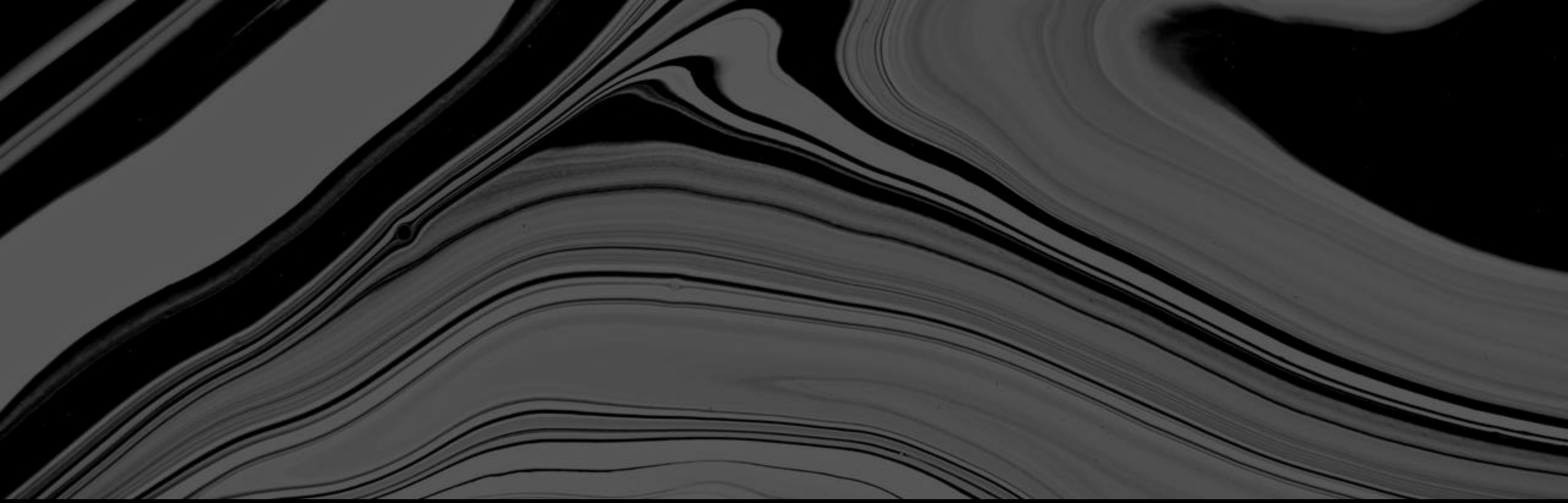


$A_1 = (s_1, A, B, t_1)$
$A_2 = (s_2, C, D, E, F, H, t_2)$

$C_{SIC} = 9$ $C_{MKS} = 6$

$A_1 = (s_1, C, D, E, F, t_1)$
$A_2 = (s_2, A, B, t_1, G, t_2)$

$C_{SIC} = 10$ $C_{MKS} = 5$



SEZIONE 3: STATO DELL'ARTE

CONFLICT BASED SEARCH (CBS)

FUNZIONAMENTO

1. **Inizializzazione**: creazione di un nodo radice
2. **Ricerca in ampiezza**: esplorazione dell'albero dei vincoli
3. **Elaborazione di un nodo**
4. **Risoluzione di un conflitto**
5. **Soluzione:** identificazione del nodo obiettivo. Ritorna l'insieme dei percorsi validi

ALBERO DEI VINCOLI

INTERSECTION CONFLCIT RESOLUTION (ICR)

FUNZIONAMENTO

1. Risolve i percorsi in modo ottimale per ogni agente (SAPF)
2. Gestisce i conflitti tra gli agenti:
    a. Estrae la porzione di grafo in conflitto
    b. Considera solo gli agenti del sottografo
    c. Risolve il problema tramite *constraint programming*
3. Unisce la nuova soluzione con quella globale
4. Se non trova soluzioni, espande il sottografo

INCREASING COST TREE SEARCH (ICTS)

ALTO LIVELLO

- Cerca la soluzione in un albero chiamato Increasing Cost Tree (ICT).
- Ogni nodo rappresenta una combinazione di costi di percorso per gli agenti.
- La ricerca avanza aumentando i costi individuali fino a trovare una soluzione senza conflitti.

BASSO LIVELLO

- Testa le combinazioni di costi del nodo corrente per trovare percorsi senza conflitti.
- Utilizza una struttura chiamata Multi-value Decision Diagram (MDD) per memorizzare i percorsi.
- Se trova una soluzione senza conflitti, conferma il nodo come soluzione; altrimenti, il processo continua.

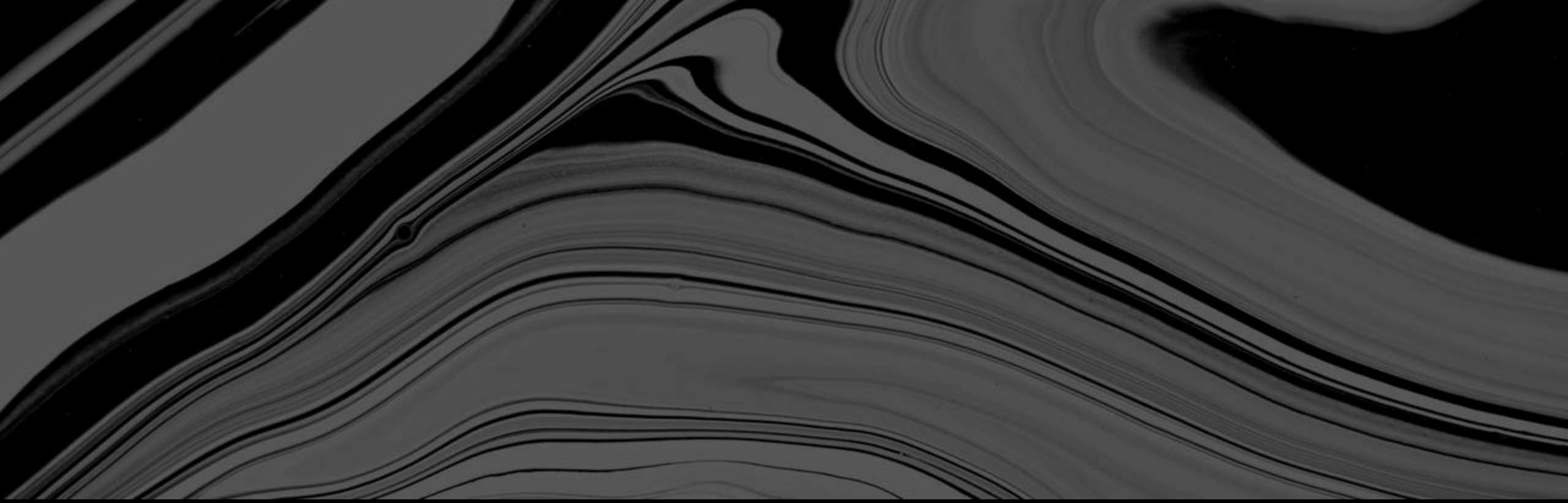
ICTS + INDEPENDENCE DETECTION (ICTS + ID)

INDIPENDENZA DEI GRUPPI

- Divide gli agenti in gruppi indipendenti che non si confliggono
- Utilizza A* per trovare soluzioni ottimali per ogni gruppo

GESTIONE DEI CONFLITTI

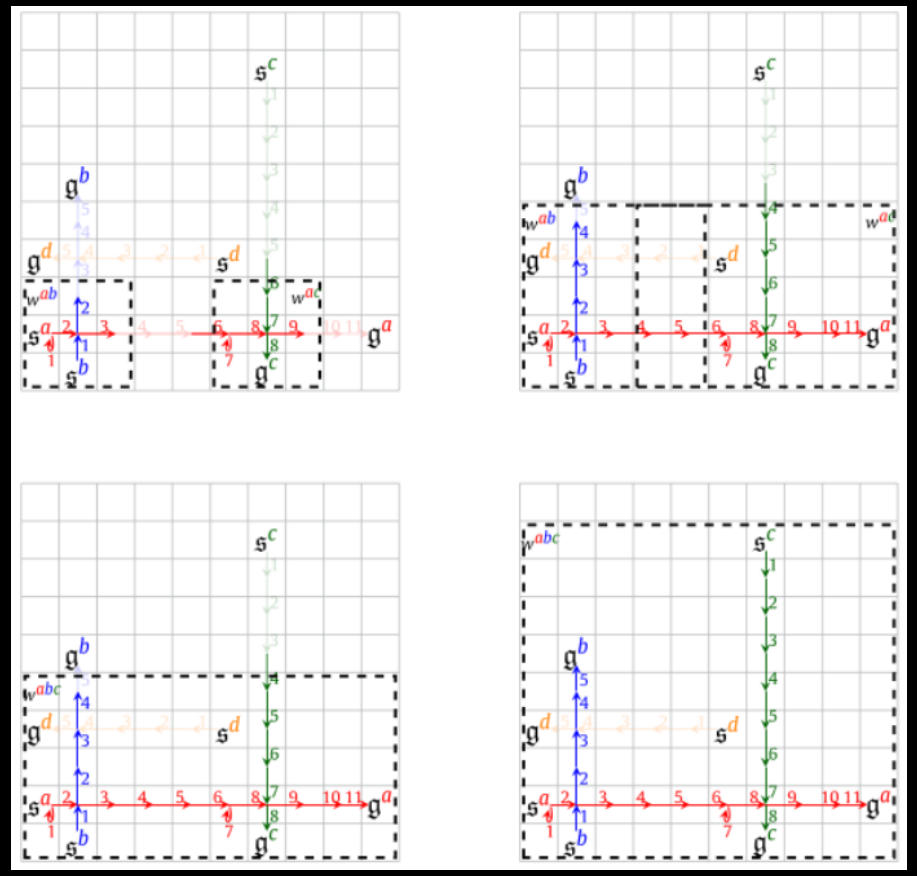
- Esegue le soluzioni dei gruppi in parallelo
- In caso di conflitto, ripianifica i percorsi per evitare fusioni
- Se il conflitto persiste, i gruppi conflittuali vengono fusi e risolti nuovamente con A*



SEZIONE 4: ALGORITMO X*

WINDOWED ANYTIME MULTIAGENT PLANNING FRAMEWORK (WAMPF)

- È fondamentale usare sottospazi per accelerare la ricerca
- Il WAMPF introduce il concetto di "finestra", un sottoinsieme di stati e agenti
- La "finestra" è posizionata intorno a una collisione per riparare il percorso



PROPRIETÀ DELLA FINESTRA

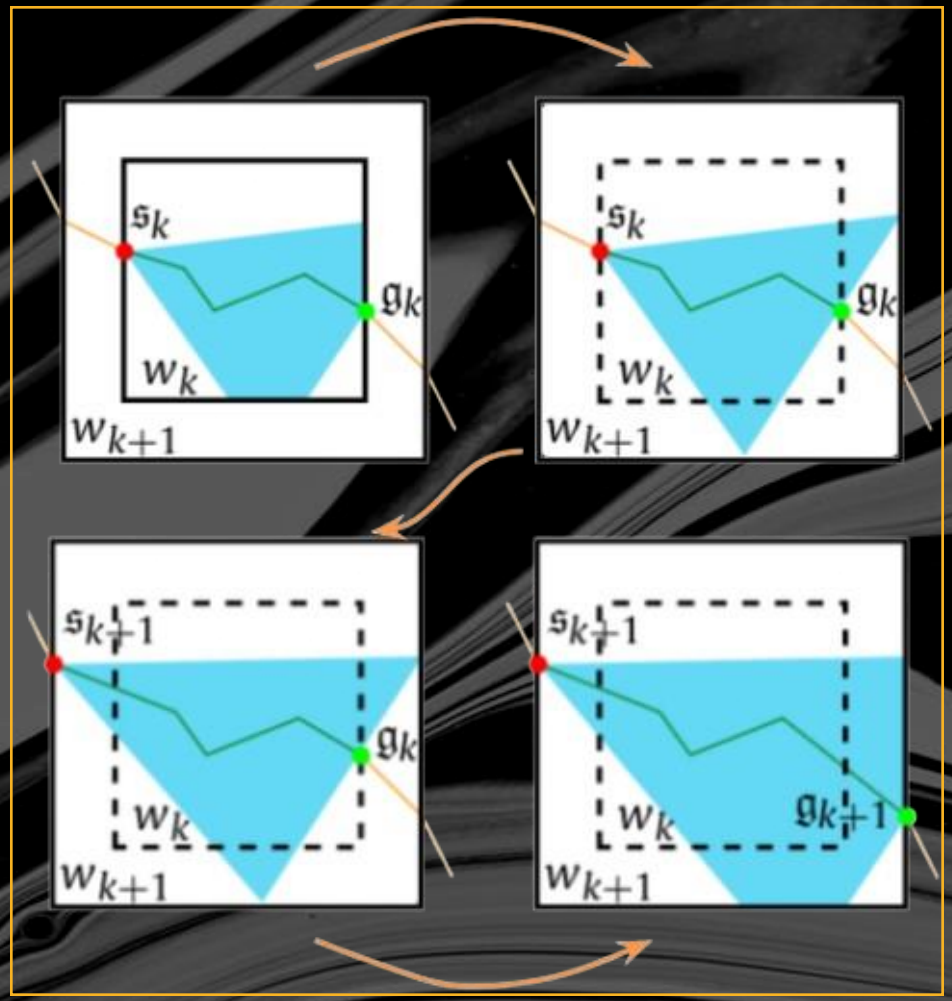
- Include un sottoinsieme connesso di stati per un gruppo di agenti
- Possiede un punto di partenza e uno di arrivo sul percorso globale
- Avere una finestra successiva con più stati e lo stesso gruppo di agenti

OPERAZIONI SULLE FINESTRE

- Unione (∪). Combina due finestre in una più grande
- Intersezione (∩). Sovrappone due finestre

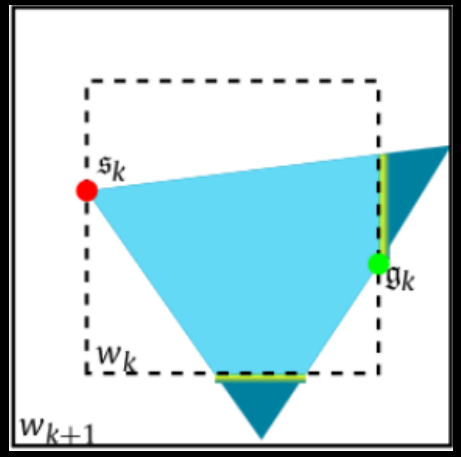
EXPANDING A*: X*

- Si basa su WAMPF
- Fa parte degli Anytime Path Planners
- Tre fasi di trasformazione:
  - **ESPANSIONE DELLA FINESTRA**
  - **SPOSTAMENTO DELL'INIZIO**
  - **SPOSTAMENTO DELL'OBIETTIVO**



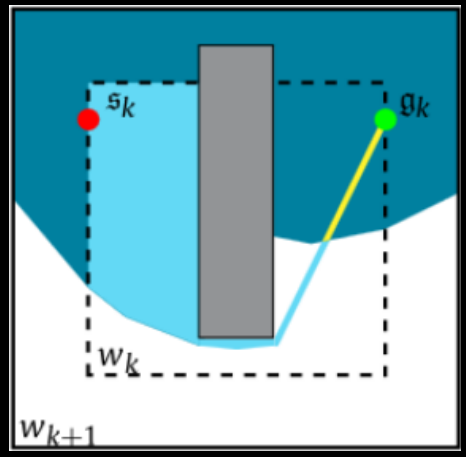
BOOKKEEPING IN X*

INSIEME FUORI DALLA FINESTRA

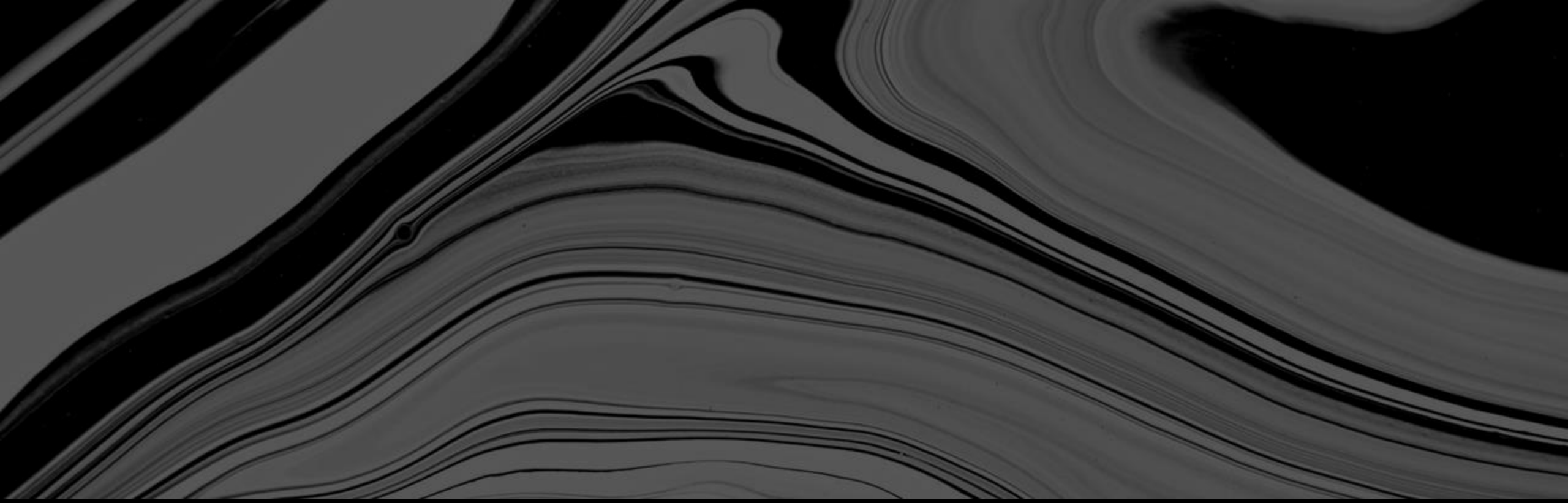


- Gli stati nella regione blu scuro non vengono considerati immediatamente nel processo di ricerca
- Vengono archiviati per un eventuale esame futuro, se la finestra di ricerca si espande

VALORE CHIUSO



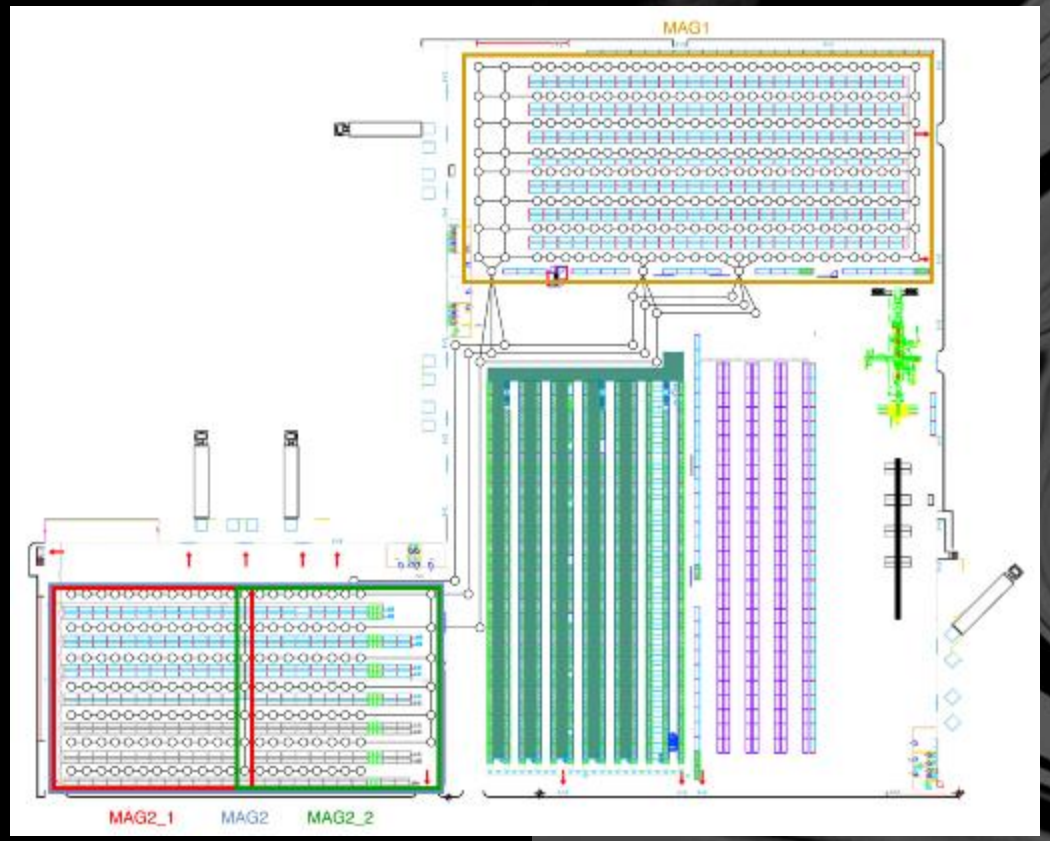
- I "valori chiusi" rappresentano i costi finali assegnati agli stati dopo che sono stati esaminati
- Questi valori sono utilizzati per evitare di riesaminare stati già processati, garantendo efficienza nel calcolo



SEZIONE 5: ANALISI SPERIMENTALE

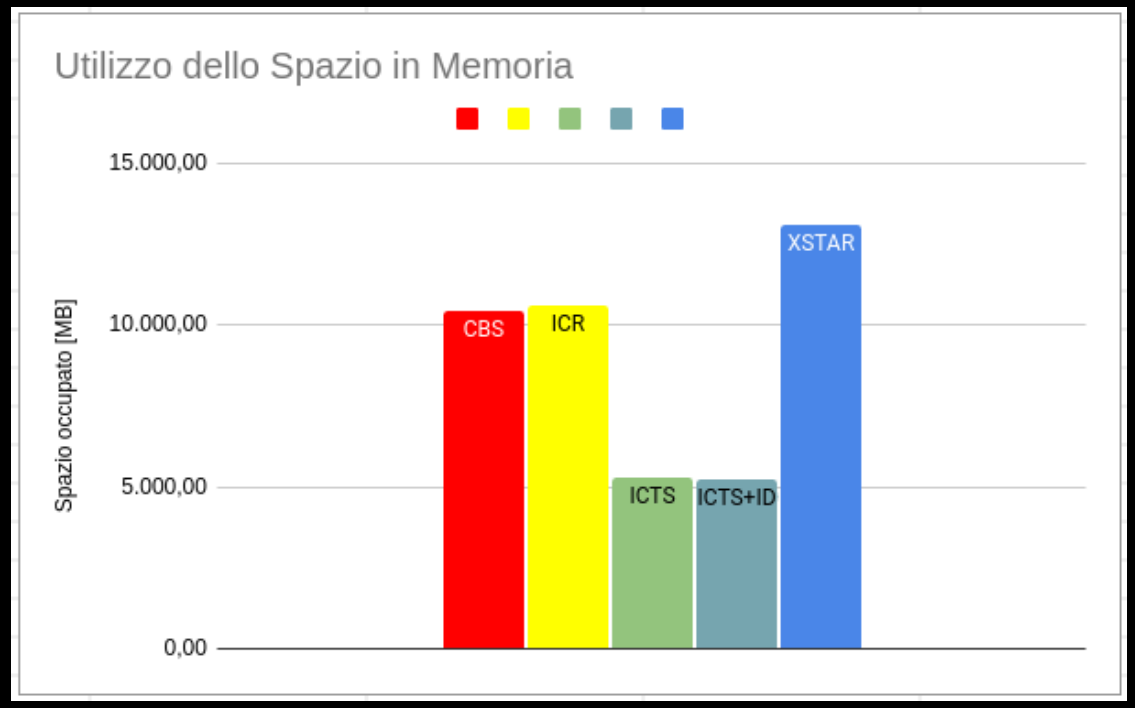
SCENARIO REALE

- Mappa topologica: 406 nodi, archi non orientati
- Suddivisione in sottoproblemi (WH1, WH2, ecc.)
- 12 test per ogni scenario, per un totale di 108 test
- Agenti: {2, 4, 8}
- Obiettivi: {2, 4, 8, 10}
- Obiettivi tipici delle attività logistiche

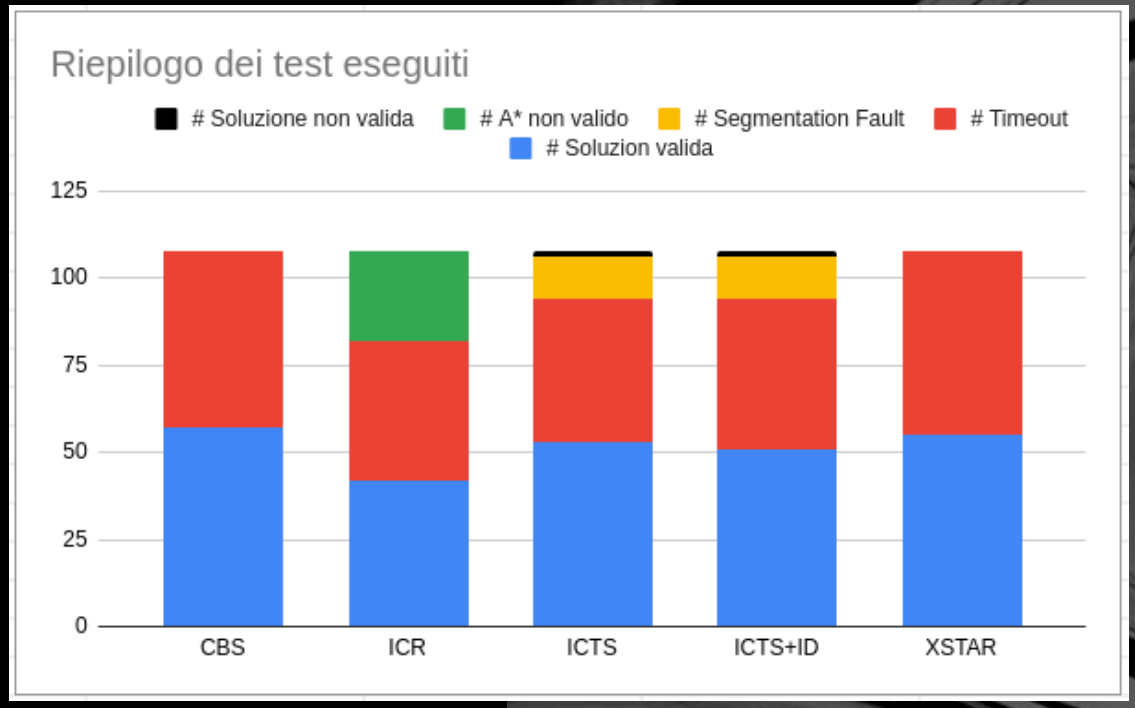


TESTING

SPAZIO OCCUPATO IN MEMORIA

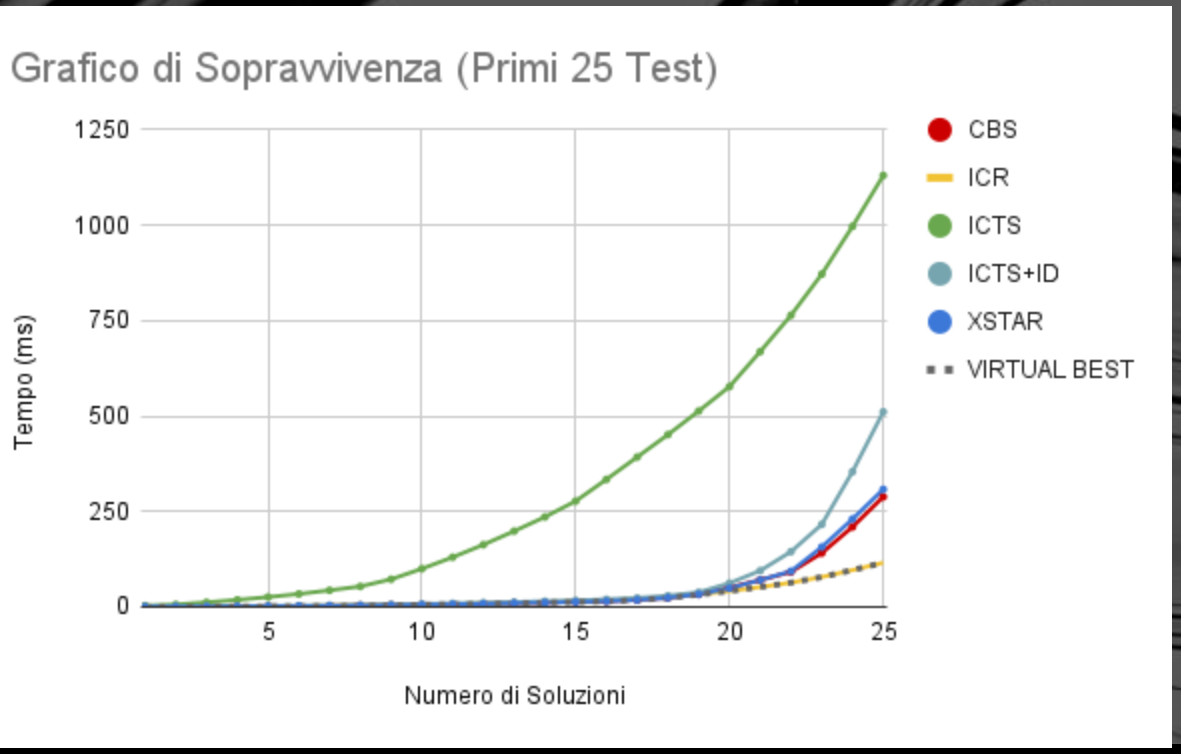


RIEPILOGO DEL TESTING

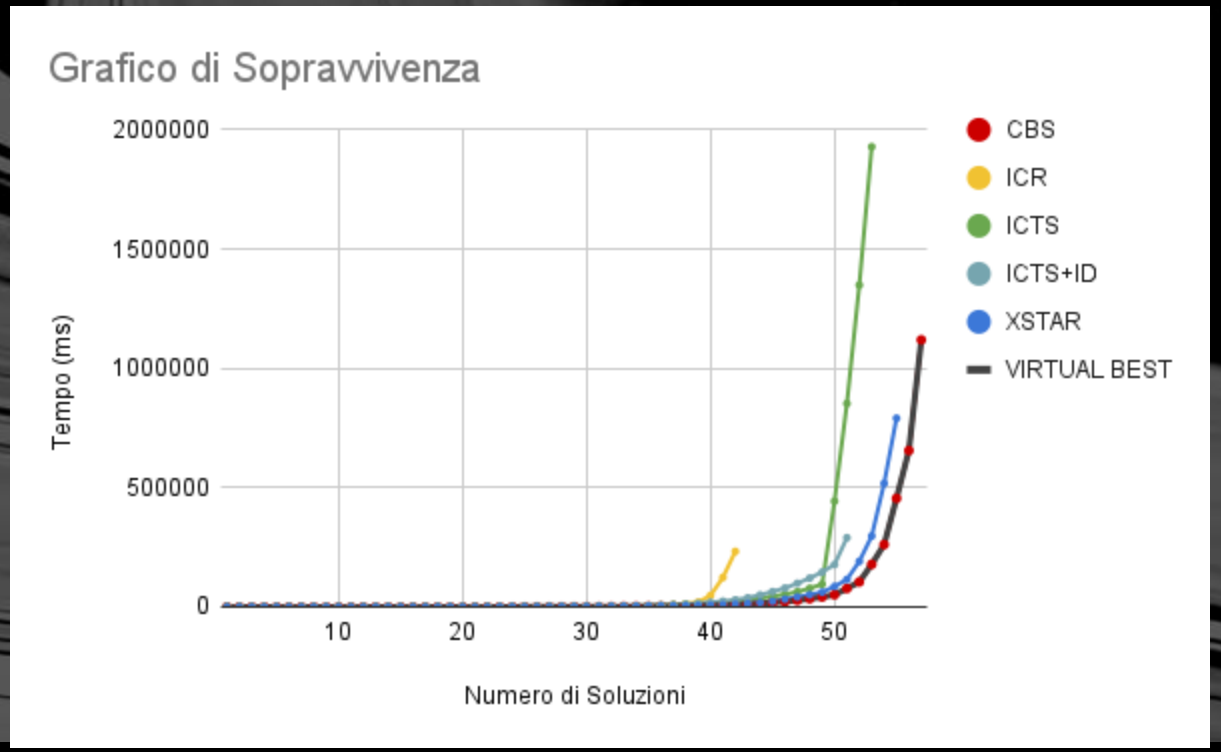


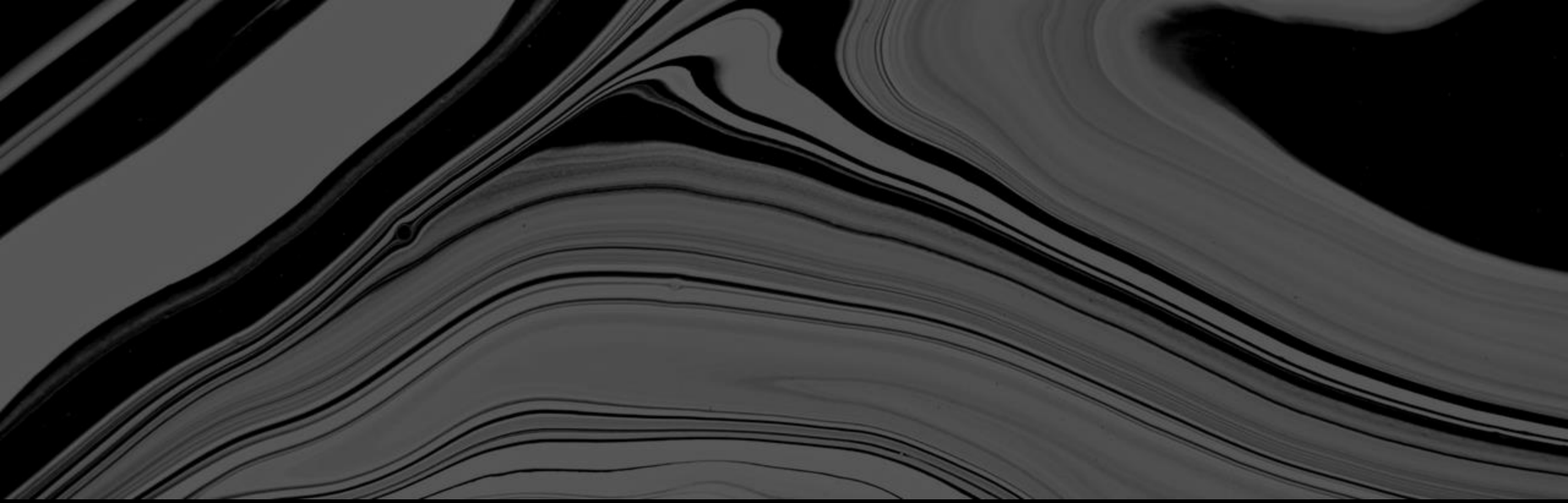
TESTING

SURVIVAL PLOT (PRIMI 25 TEST CON SOLUZIONE)



SURVIVAL PLOT COMPLESSIVO





SEZIONE 6: CONCLUSIONE

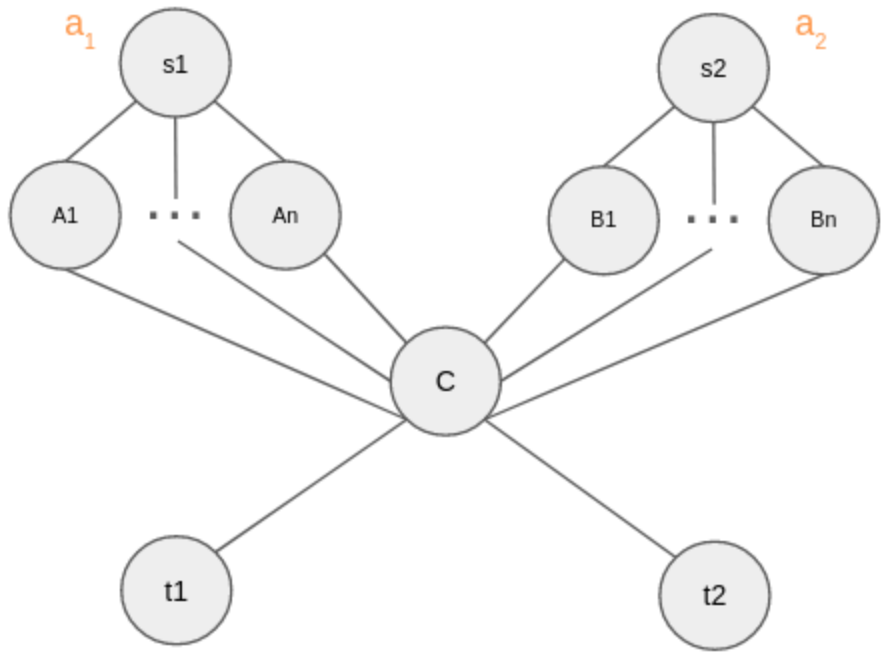
CONCLUSIONE

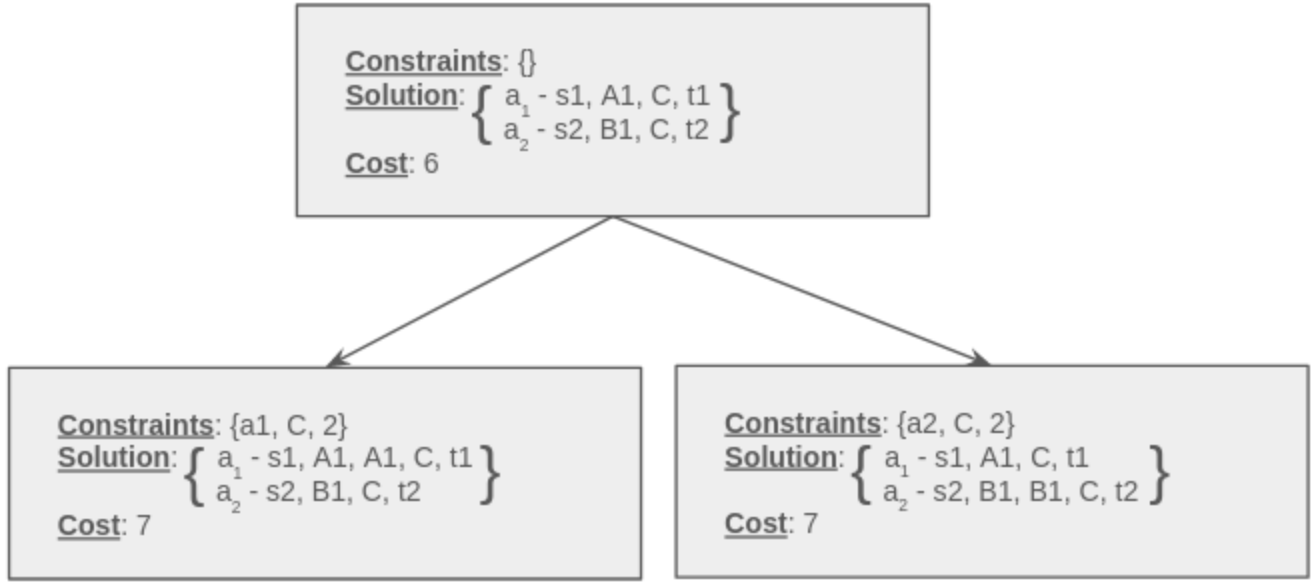
- Alcuni test troppo semplici per evidenziare differenze significative
- **X*** e **CBS** sono risultati i migliori, spesso con esecuzioni quasi identiche
- Ciascun algoritmo ha punti di forza specifici

- Sviluppo di nuove euristiche per scenari complessi
- Progettazione di nuovi algoritmi combinando i punti di forza di ciascun approccio
- Soluzioni più efficaci per la gestione delle risorse e l'allocazione dei compiti in ambienti complessi e dinamici

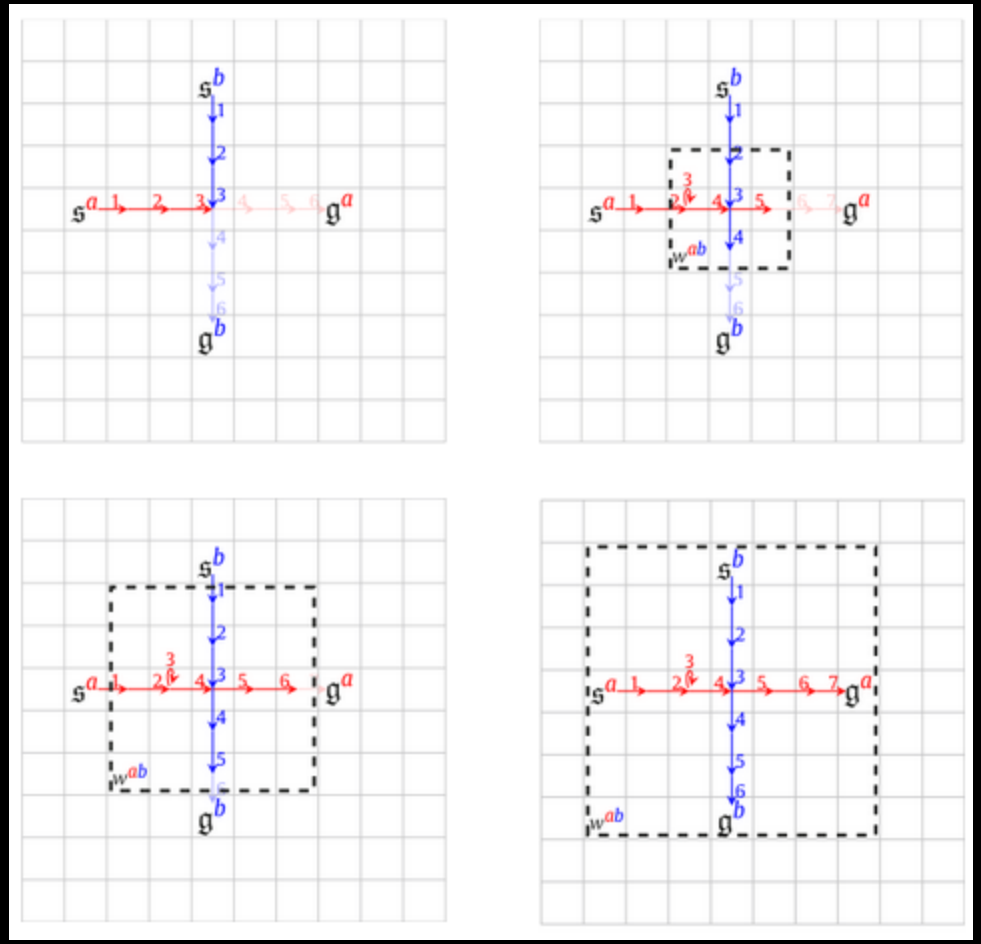
GRAZIE PER LA VOSTRA ATTENZIONE

CONFLICT BASED SEACRH (CBS)

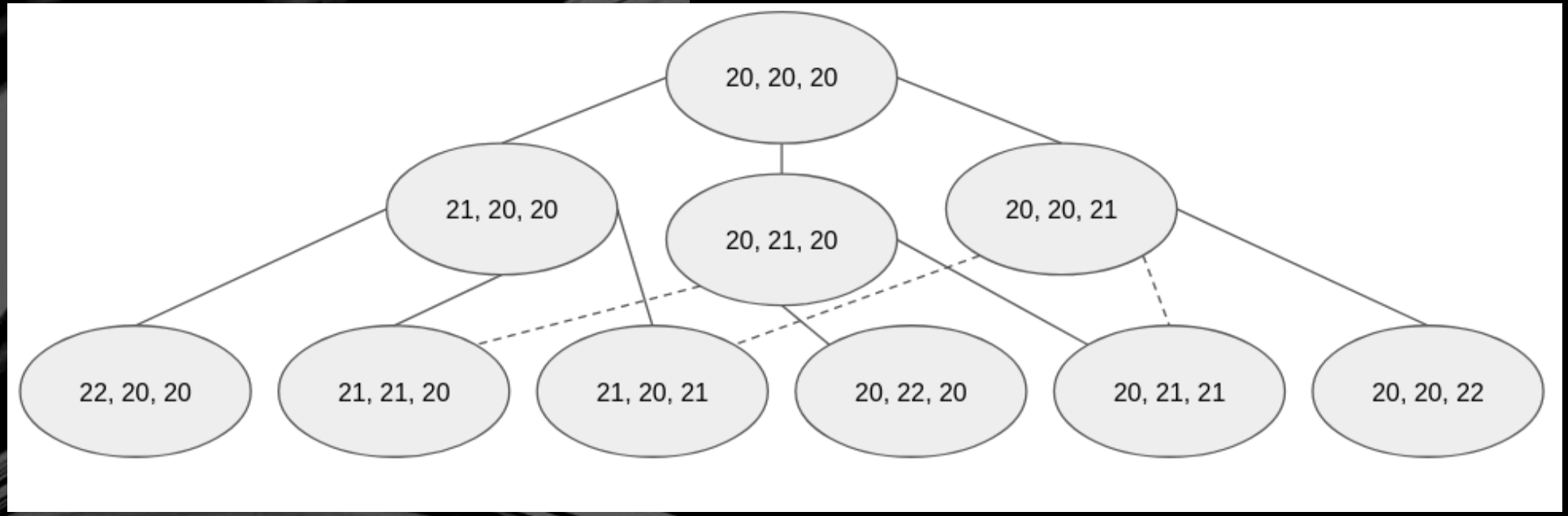




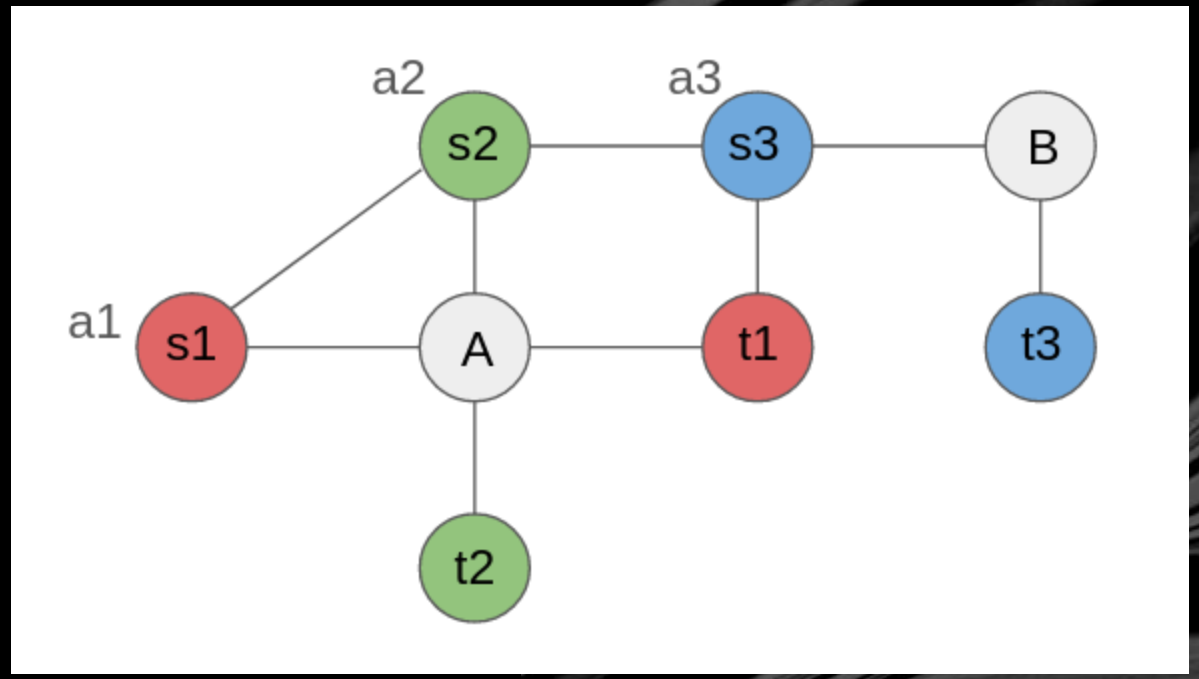
INTERSECTION CONFLICT RESOLUTION (ICR)



INCREASING COST TREE SEARCH (ICTS)



ICTS + INDEPENDENCE DETECTION (ICTS + ID)



ANYTIME PATH PLANNER (APP)

DEFINIZIONE

- Sviluppano una soluzione iniziale in poco tempo
- Con più tempo di calcolo, migliorano iterativamente la qualità della soluzione

PROBLEMA INIZIALE

- Nessun riutilizzo delle informazioni
- Iterazioni successive sempre più lente

VANTAGGI

- Adatta le soluzioni a diverse esigenze di tempo e risorse
- Flessibile per ambienti dinamici