

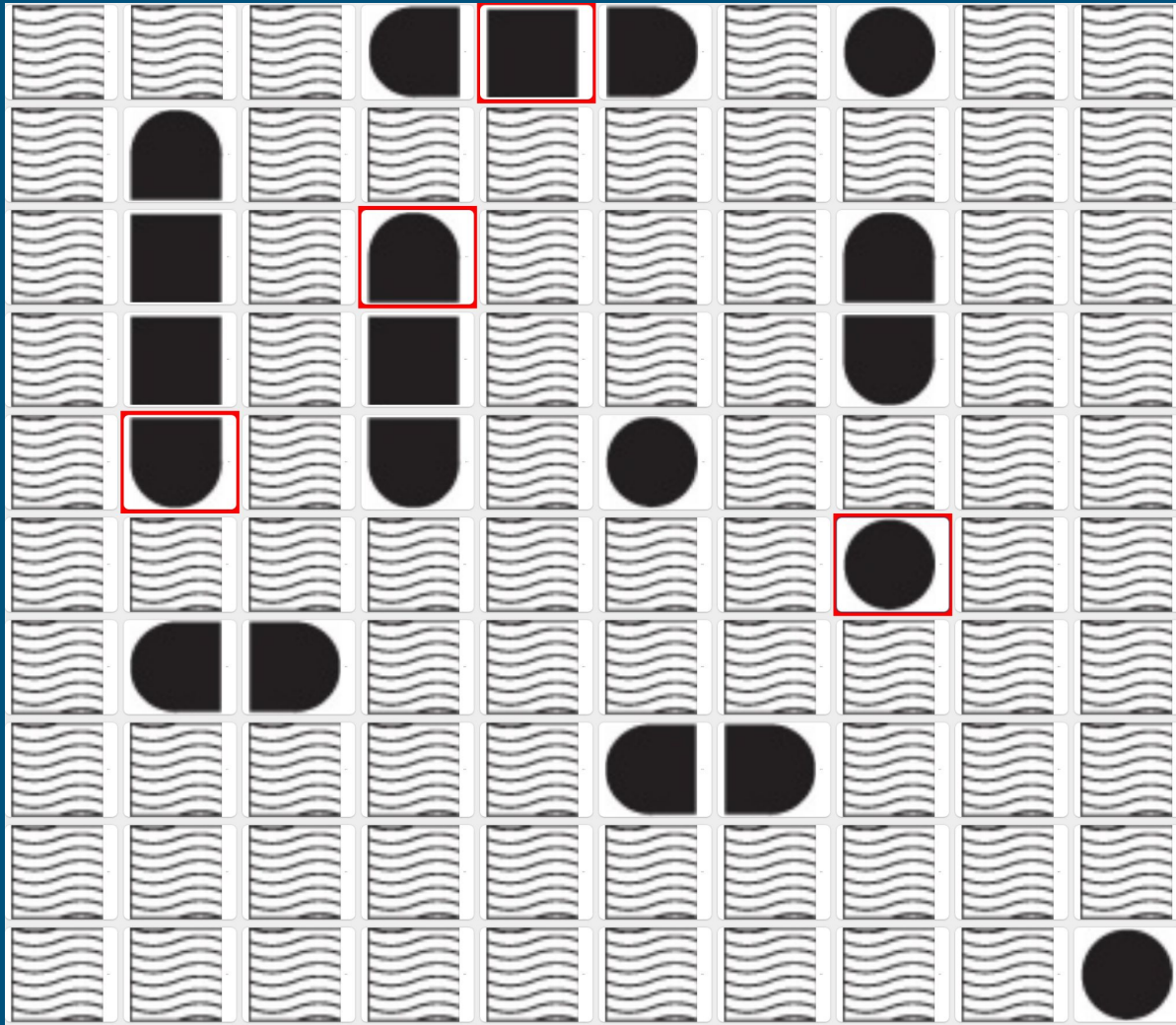
Progetto Clips – Battaglia Navale

Università degli studi di Torino
2024/25

Lavoro a cura di

Paolo Bonicco m. 833708
Giada Pedone m. 1013452

Introduzione



Una delle mappe
utilizzate per testare
i nostri due agenti

—

Le celle evidenziate
in rosso
rappresentano la
conoscenza iniziale
dell'agente

Modellazione della conoscenza

Tipi di fatti utilizzati

- **Fatti ordinati** — liste di valori semplici
Esempio: (fire 2 3), (guess 5 7), (known-water 3 4)
- **Fatti non ordinati (deftemplate)** — strutture con slot nominali
Esempio: (exec (step 5) (action fire) (x 2) (y 3))
- **Template principali:**
 - exec: azioni da eseguire
 - k-cell: conoscenza celle (water/top/bot/left/right/middle/sub)
 - k-per-row / k-per-col: contatori navi per riga/colonna

Modellazione delle regole di expertise

Gestione della conoscenza incerta

- Separazione tra fatti certi e ipotesi
- Meccanismo di retraction (unguess)
- Preservazione integrità contatori
- Evoluzione: da regole locali (Naive) a reasoning globale (Smart)

Strategia a livelli progressivi

1. Azioni certe
 - a. guess su celle note (k-cell)
2. Estensioni logiche
 - a. celle adiacenti a navi note
3. Esplorazione euristica
 - a. fire/guess su celle probabili
4. Terminazione

Strategie di soluzione implementate

AGENTE NAIVE

- **Guess celle note**
 - Decrementa contatori
- **Estensione navi**
 - 4 regole direzionali (left/right/up/down)
 - Non verifica orientamento middle
- **Fire strategico**
 - Max riga e max colonna (no somma)
 - Se occupata non trova un'altra cella
- **Fire fallback**
 - Prima cella promettente disponibile (prima cella che ha $k\text{-per-row}/k\text{-per-col} > 0$)
- **Guess speculativo**
 - Come fire sopra
 - Decrementa contatori (anche se non certezza di colpire)

Le regole sono in ordine di priorità (salience)

Strategie di soluzione implementate

AGENTE SMART

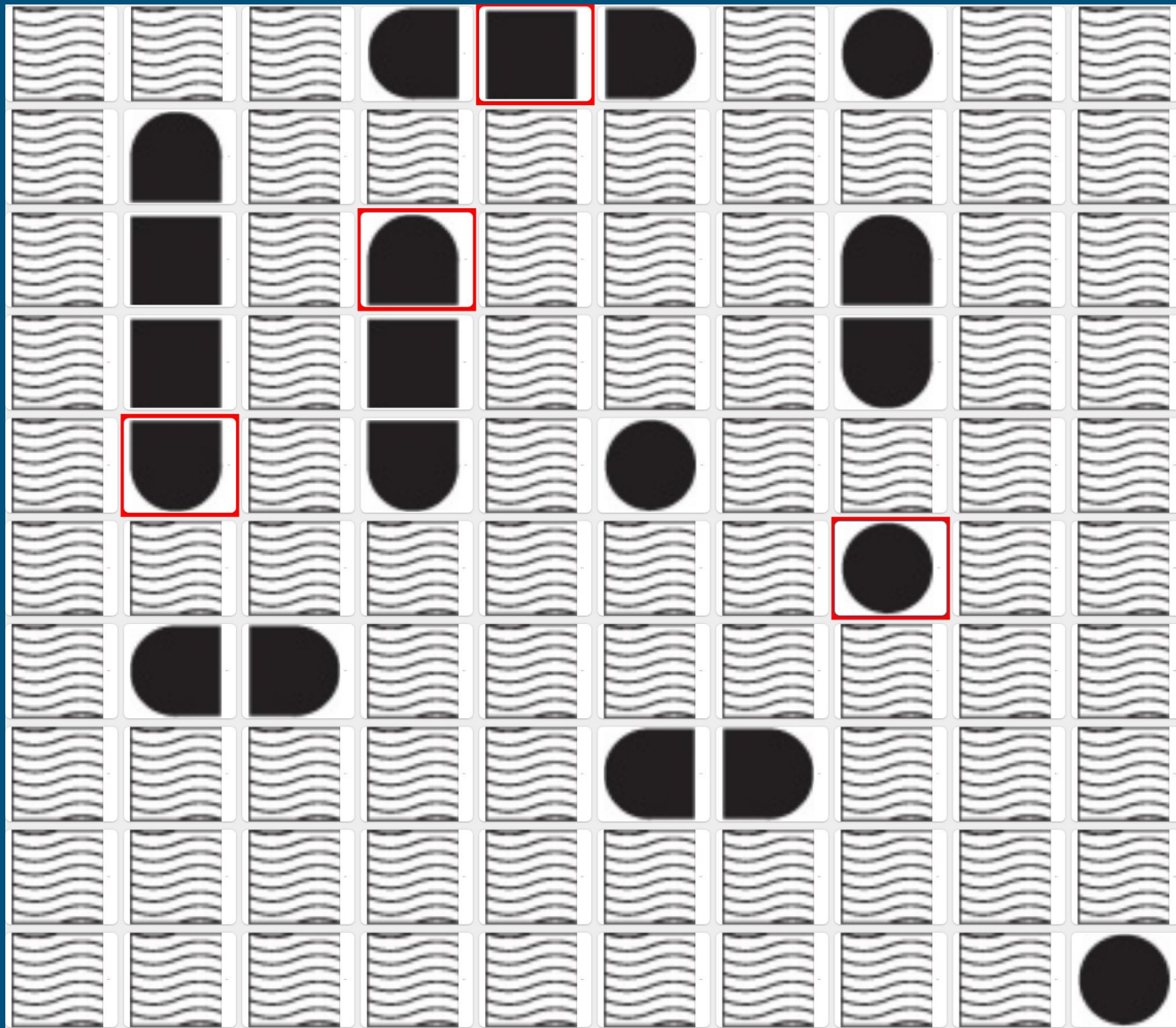
- **Deduzione acqua**
 - In base alle regole di gioco che prevedono navi non adiacenti
 - se troviamo un pezzo di nave, attorno c'è acqua
 - controlla orientamento nave per middle
- **Un-guess su celle water**
 - ripuliamo guess errati
- **Guess celle note**
 - decrementa contatori
- **Estensioni intelligenti**
 - Verifica orientamento e contatori
- **Fire con score nr+nc**
 - Se cella già guessed/fire ne cerca un'altra
- **Guess speculativo**
 - Non decrementa contatori riga/colonna

Le regole sono in ordine di priorità (salience)

Esempi di Regole Significative

Regola	Precondizioni	Effetto
guess-known-boat (Naive/Smart, salience 10)	Esiste k-cell con nave (top/bot/left/right/middle/sub)	Guess sulla cella + decrementa contatori riga/colonna
deduce-water-row-zero (Smart, salience 20)	k-per-row ha num=0, cella non nota	Asserisce (known-water x y)
guess-extend-left (Smart, salience 8)	Esiste right o middle con vicino confermato, contatori riga/colonna > 0	Guess sulla cella a sinistra
fire-best-cell (Smart, salience 5)	Trova cella con max(nr+nc), non nota, non acqua dedotta	Fire sulla cella con score massimo globale

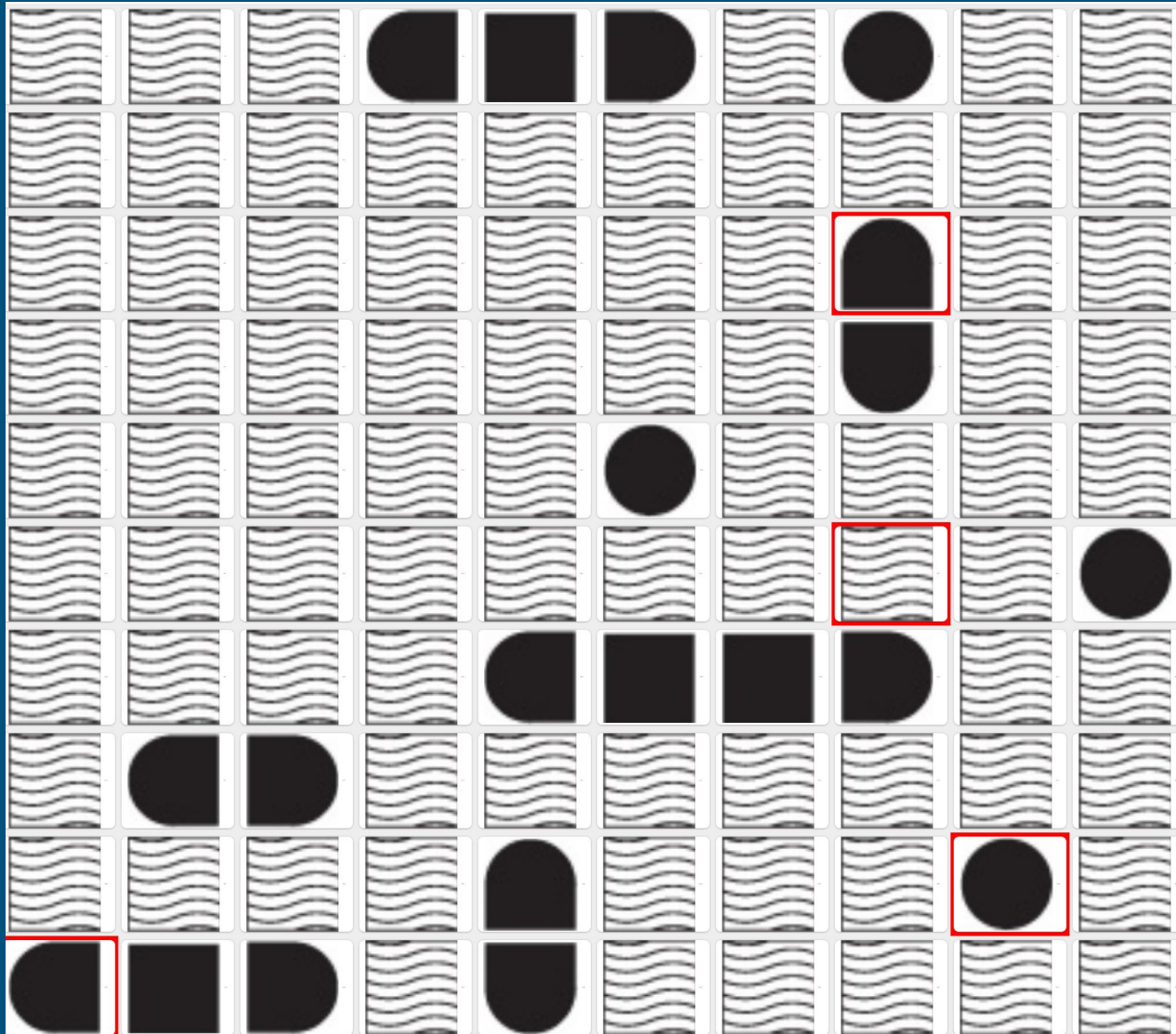
MAPPA 1



Score mappa 1 con celle conosciute diverse

Mappe	4 k-cell	2 k-cell	0 k-cell
Agente Naive	25	40	-85
Agente Smart	175	135	130

MAPPA 2



Score mappa 2 con celle conosciute diverse

Mappe	4 k-cell	2 k-cell	0 k-cell
Agente Naive	65	25	-85
Agente Smart	140	100	25

Limiti Agenti Naive

L'agente naive presenta tutte le limitazioni presenti nell'agente smart più le seguenti

1. Estensioni limitate

- a. Regole funzionano solo su celle già note
- b. Non controllano orientamento nave

2. Fire/guess rudimentali

- a. Fire-best-cell valuta solo riga e colonna separatamente, senza combinare le informazioni
- b. Guess e fire fallback scelgono la prima cella promettente trovata

Limiti Agente Smart

1. **Estensioni** non sempre ottimali
 - a. Non vanno a controllare se è stata colpita tutta la nave
 - b. Non verificano lunghezza nave prima di estendere
2. Manca backtracking su estensioni errate (**unguess**)
3. **Guess speculativi** non ottimizzati (vengono attivati troppo spesso)
4. Scelgono prima cella con $\max(nr+nc)$ trovata (euristica non ottimale)
5. Non apprendono da guess speculativi precedenti
6. Nessun controllo di consistenza tra contatori e celle scoperte
7. Decisioni greedy locali, non valuta conseguenze future
8. Non ottimizza sequenza fire per massimizzare informazione

Grazie