

2

Note generali sul sistema

In questa sezione vengono fornite alcune note generali sul sistema MAS stesso e sulle problematiche affrontate per giungere alla sua realizzazione: per quanto riguarda una generica descrizione del sistema si faccia riferimento a quanto riportato nell'abstract.

A livello pratico, la realizzazione del sistema è stata effettuata utilizzando Java, Tucson e Tuprolog.

2.1 Aspetti generali

Lo sviluppo di questo progetto si basa su due tipologie ben definite di astrazioni: quella di agente e quella di MAS.

Per quanto riguarda il concetto di agente, questo progetto vuole mettere in luce, tra le caratteristiche principali di agente quelle di proattività, reattività e abilità sociale.

Ogni agente del MAS realizzato è infatti proattivo:

- Ad esempio un agente rilevatore, ad ogni istante logico prende decisioni riguardanti la prossima cella libera nella quale si muoverà, comunica agli altri agenti quali sono le caselle libere e quali quelle minate, il tutto senza che nessuno gli dica di farlo.

- Allo stesso modo gli agenti sminatori autonomamente controllano se vi siano o meno mine segnalate nelle loro vicinanze (non ancora target di altri eventuali agenti sminatori) e, nel caso siano presenti, provvedono a individuare, se presente, un percorso che li conduca alla mina target (in questo senso è senz'altro importante anche il means-ends reasoning di questi agenti, mediante il quale essi si calcolano un vero e proprio piano).

L'aspetto legato alla reattività invece è più importante per ciò che riguarda gli agenti sminatori: essi infatti sono reattivi ai cambiamenti che avvengono nel loro ambiente (il campo minato nella fattispecie). I cambiamenti possono ad esempio essere indotti da un altro agente oppure da un agente umano, il quale può ad esempio decidere di rimuovere una mina (per aiutare gli agenti nel loro compito di bonificare completamente il campo minato) già target di un agente: l'agente sarà in grado di capire che la mina target non è più presente, reagendo a questo evento tramite una riconsiderazione delle proprie intenzioni, ovvero scegliendo (se presente) una nuova mina come proprio target.

Per quanto riguarda l'abilità sociale, si tratta di un aspetto cruciale per quanto riguarda questo progetto, soprattutto perché esso è focalizzato principalmente sul concetto di MAS e di goal globale.

Nello specifico il goal globale è la bonifica totale (o quantomeno più completa possibile) dell'ambiente rappresentante il campo minato: a tal fine, visto che i 2 tipi di agenti hanno capacità parziali, diventa cruciale la questione relativa all'abilità sociale degli agenti: questo aspetto riveste particolare importanza soprattutto per quanto riguarda gli agenti sminatori, in quanto la loro unica capacità di disinnescare mine, li rende

completamente dipendenti (per quanto riguarda i loro movimenti nel campo minato) dagli agenti rilevatori.

In parole povere, gli agenti sminatori sono ciechi e si possono muovere soltanto grazie alle informazioni provenienti dagli agenti rilevatori: se infatti gli agenti sminatori fossero costretti a muoversi “alla cieca”, rischierebbero di esplodere su di una mina non segnalata.

Gli agenti rilevatori, da questo punto di vista hanno meno problemi perché, data la loro capacità di “vedere” il terreno loro circostante e di rilevare eventuali mine, non corrono il pericolo (a meno di malfunzionamenti teorici degli apparati) di esplodere su di una mina: malgrado questo, la loro presenza come componenti del MAS è fondamentale perché, senza di essi, gli agenti sminatori sarebbero perfettamente inutili e non si sarebbe in grado di completare il goal globale.

Dai temi appena citati emergono anche, quali temi importanti, quelli della cooperazione e coordinazione all’interno del MAS: entrambi gli aspetti (come si sarà compreso da quanto appena detto) sono affrontati in questo progetto. A questo proposito, questo sistema dimostra anche come sia possibile, a partire dai goal dei singoli agenti, giungere al goal globale dell’intero MAS:

- Gli agenti sminatori hanno come goal quello di disinnescare mine;
- Gli agenti rilevatori hanno come goal quello di esplorare casualmente quanto più terreno possibile, comunicando di volta in volta la posizione delle mine individuate.

Un'ultima cosa che si ritiene importante sottolineare per quanto riguarda gli agenti rilevatori, è relativa all'episodicità dell'ambiente: infatti, la loro decisione su quale casella andare ad occupare in un certo istante logico viene presa tenendo in considerazione un numero limitato di caratteristiche dell'ambiente, nella fattispecie viene tenuto in considerazione soltanto ciò che avviene nelle caselle adiacenti al detect agent.

2.2 Analisi del problema

In questa sezione vengono analizzati sinteticamente e in maniera astratta gli aspetti del sistema che si ritengono importanti, analizzando ordinatamente i singoli attori del sistema.

2.2.1 L'ambiente

L'ambiente, relativamente a questo progetto, è il campo minato. Il campo minato è una porzione di territorio limitata rappresentata tramite una mappa in 2 dimensioni, suddivisa in tante caselle: ogni casella è individuata tramite una coppia di coordinate x-y (da considerarsi alla stregua di longitudine e latitudine). Nell'ambiente sono inoltre presenti delle mine che vanno rimosse (disinnescate). A tal proposito interagiscono con l'ambiente più agenti, facenti capo a due distinte tipologie: agenti sminatori (in grado di rimuovere mine) e agenti rilevatori (in grado di vedere le mine, e di conseguenza anche le caselle libere).

2.2.2 Agenti sminatori

Gli agenti sminatori sono in grado solamente di disinnescare le mine, ma sono ciechi e quindi hanno bisogno della guida (delle informazioni) dei detect agents per muoversi senza correre il rischio di esplodere su di una mina. In quale modo è possibile renderli intelligenti? O meglio in che

modo è possibile strutturare i loro ragionamenti, per dirigerli nella direzione voluta?

Di norma possiamo identificare 3 tipi principali di intenzioni per quanto riguarda questi agenti:

- Trovare una mina nelle vicinanze non ancora target di altri agenti sminatori verso la quale muovere (il problema relativo alla ricerca di un percorso dalla posizione attuale dell'agente alla mina è un problema di means-ends reasoning o pianificazione). Il fatto di aver specificato "una mina nelle vicinanze" piuttosto che "una qualsiasi mina già rilevata" è importante: infatti, fare in modo che normalmente l'agente operi su mine rilevate che si trovano nelle proprie vicinanze, fa sì che nell'ottica dell'intero MAS il goal globale sia raggiunto in maniera più efficace.
- Nel caso non vi siano mine nelle vicinanze, l'agente rimarrà fermo nella sua posizione, oppure nel caso vi siano caselle a lui adiacenti dichiarate libere dai detect agents, potrà muovere su una di queste nella speranza di avvicinarsi verso una mina già rilevata (e non ancora target di altri unmine agents).
- E' anche possibile pensare, dopo un certo tempo in cui l'agente si muova senza alcuna mina target, che esso a quel punto recuperi le informazioni relative a tutte le mine rilevate presenti nel campo minato (non soltanto quelle a lui vicine): in questo modo, dopo aver rimosso una di queste mine (anche lontane dalla sua posizione) è sperabile che l'agente abbia altre mine rilevate vicine (che potrà rimuovere in istanti successivi).

2.2.3 Agenti rilevatori

Gli agenti rilevatori hanno il compito di esplorare in maniera casuale il territorio “scoprendo” porzioni di territorio le più ampie possibili. In particolare ad ogni istante logico è ragionevole che il generico detect agent si sposti in una casella libera ad esso adiacente: nel caso l’agente abbia più caselle libere attorno, sceglierà la prossima casella in modo casuale. Questo consente inoltre di pensare ad agenti rilevatori che memorizzino il percorso seguito all’interno del campo minato tramite una sorta di lista a doppio link (tanto per usare un termine informatico!) ovvero tramite un albero o grafo orientato: a tal proposito ad ogni diramazione dell’albero corrispondono più caselle libere adiacenti esplorate dall’agente. L’agente idealmente si sposterà avanti e indietro lungo questo albero, creando nuove diramazioni ogni qualvolta vengano trovate nuove caselle libere non ancora esplorate, che saranno l’inizio del nuovo ramo (il ramo terminerà quando non vi saranno più caselle libere in cui muoversi relativamente al ramo considerato: a questo punto l’agente ripercorrerà a ritroso il ramo alla ricerca di eventuali altre caselle libere prima ignorate e che potranno essere l’inizio di un nuovo sottoramo).

Per finire, gli agenti rilevatori avranno anche il compito di segnalare a tutti gli altri agenti (di tipo unmine in particolare) le caselle esplorate, dichiarandole come libere o come minate.

2.2.4 Il canale radio

In canale radio rappresenta il fulcro della coordinazione e della cooperazione tra gli agenti, il governo della loro interazione: esso sarà utilizzato dagli agenti detect per segnalare le caselle visitate (dichiarandole come libere o minate), e dagli agenti unmine per rilevare quali siano le caselle minate (che diventeranno potenziali target della loro

opera di sminamento) e le caselle libere (potenziali target per i loro movimenti).

Parlare di canale radio è soltanto una astrazione per farne comprendere al lettore il ruolo: infatti è anche possibile pensare al canale radio come facente parte dell'ambiente, ma questo non ha importanza ai fini del progetto in questione.

2.2.5 Il goal globale del MAS MineField

Il goal globale del MAS, come già detto più volte è quello di liberare completamente dalle mine il campo minato o, anche più semplicemente, di rimuoverne il maggior numero possibile.

2.3 Progetto

In questo paragrafo verranno discussi brevemente gli aspetti principali che hanno caratterizzato il progetto del MAS: si cercherà di rimanere quanto più possibile indipendenti dagli aspetti più prettamente tecnologici.

In figura 2.1 viene riportato lo schema generale su cui si baserà tutta la discussione relativa al progetto.

Dalla figura di ricavano subito alcuni aspetti molto importanti del progetto:

- L'ambiente è rappresentato in maniera assolutamente simbolica, per cui saranno simbolici anche i ragionamenti che gli agenti faranno sull'ambiente stesso (sposando quindi la visione più classica dell'A.I.).
- In relazione a quanto detto al punto precedente, si utilizzerà la proprietà di mondo chiuso (Close World Assumption): tutte le

cose non esplicitamente dette saranno considerate false (NaF). Questo significa ad esempio che, per quanto riguarda la rappresentazione del campo minato, tutte le caselle simbolicamente non rappresentate saranno da considerarsi vuote, ovvero sgombre da mine.

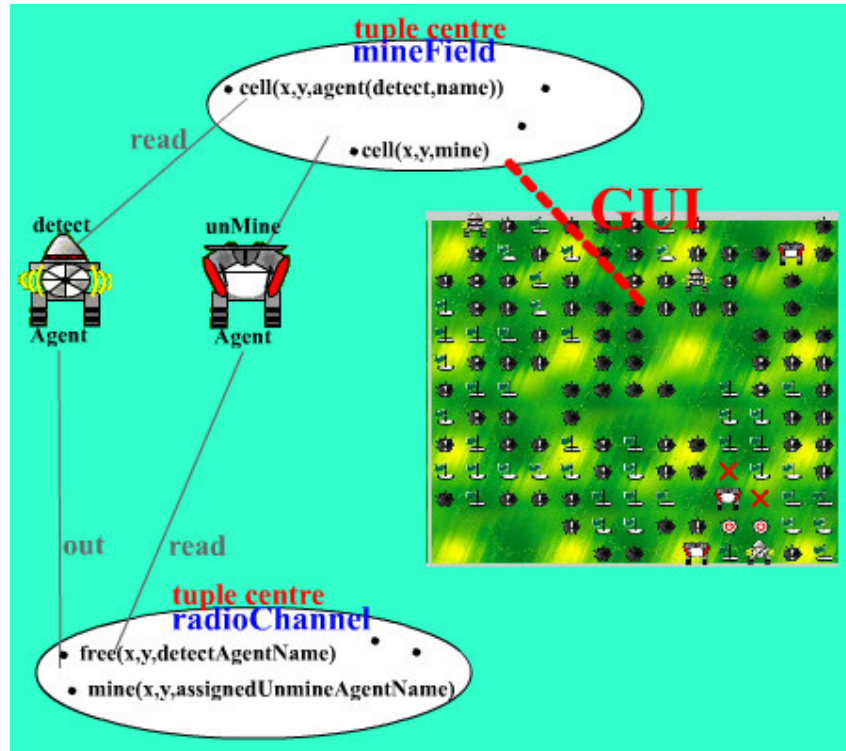


Figura 2.1: Architettura logica del MAS MineField

- Il campo minato sarà realizzato utilizzando un centro di tuple, ovvero uno spazio di tuple più una serie di regole per governare l'accesso degli agenti al campo minato stesso. Lo spazio di tuple consentirà la rappresentazione simbolica del campo minato, mentre le regole imporranno una ben precisa condotta agli agenti unMine e detect che vi accederanno: per la verità, gli agenti unMine non accederanno mai direttamente a questo centro di tuple in quanto, per loro stessa definizione, sono

ciechi e quindi non hanno la capacità di “leggere il contenuto” del campo minato stesso. Gli agenti di tipo detect invece saranno in grado di leggere il contenuto del tuple centre rappresentante il campo minato.

- Anche il progetto del canale radio si basa su di un centro di tuple: a tal proposito vale la pena di sottolineare come si tratti di una sorta di canale radio virtuale e generativo, in cui i messaggi inseriti sono indipendenti dalla vita o dalla morte dell’agente che li ha generati e, cosa ancora più importante, sono permanenti. Per quanto riguarda il canale radio l’ipotesi di CWA non è da considerarsi valida: sono infatti rappresentate esplicitamente sia tutte le caselle rilevate come libere che tutte le caselle rilevate come minate; questa è l’unica strada percorribile se si pensa al fatto che il canale radio è l’unico modo che hanno gli agenti unmine per sapere se una casella è sicura o meno per il movimento. Se non fosse rappresentata esplicitamente alcuna casella dichiarata come libera, il generico agente unMine potrebbe dedurre che tutte le caselle non rilevate come minate (dai vari detect agents) siano libere: come risultato sarebbe garantita prima o poi (è solo questione di fortuna o meno) la sua esplosione su di una qualche mina non ancora rilevata!

Visti rapidamente questi punti apparirà forse meno chiaro il concetto di ambiente: l’ambiente è il solo campo minato, il solo canale radio, oppure entrambi? Per chi scrive l’ambiente è da considerarsi costituito sia dal campo minato che dal canale radio, ovvero questi 2 centri di tuple sono da considerarsi come 2 aspetti di uno stesso ambiente con il quale

interagiscono detect e unmine agents al fine di portare a compimento il goal globale.

2.3.1 Organizzazione di una simulazione del MAS

Ogni simulazione del MAS è organizzata a turni, in maniera molto simile a quanto accade per un gioco: ogni turno rappresenta un istante logico della simulazione all'interno del quale tutti gli agenti effettueranno il loro "turno" di mosse. Per ulteriori dettagli circa le azioni compiute da ogni agente all'interno di un turno, si faccia riferimento al paragrafo dedicato al progetto.

La scelta di organizzare la simulazione secondo un modello a turni è una scelta puramente di convenienza: infatti questa scelta rende solamente più semplice rappresentare graficamente all'utente le azioni virtuali intraprese dai vari agenti.

2.4 Implementazione

Per quanto riguarda l'implementazione vera e propria del sistema, si è fatto riferimento allo schema riportato nella già citata figura 2.1.

I centri di tuple rappresentanti il canale radio e il campo minato sono stati realizzati utilizzando Tucson.

2.4.1 "Intelligenza" degli agents

Lintelligenza degli agenti, ovvero tutti gli aspetti riguardanti la deliberazione e il means-ends reasoning sono stati realizzati utilizzando tuProlog come Prolog engine: questo fatto, così come la scelta di Tucson, appare chiara se si pensa che per la realizzazione dell'intero sistema e degli agenti si è utilizzato Java come linguaggio di programmazione.

2.4.2 Architettura “Object Oriented” del sistema

In figura 2.2 è riportato il diagramma statico delle classi costituenti il sistema realizzato.

Come si può notare, il sistema è diviso in vari package facenti capo al package principale denominato `sid`.

2.4.2.1 Package `sid.agent`

In questo package sono contenute tutte le classi che, a run-time, rappresenteranno i due tipi di agenti: `detect` e `unMine` agent.

La classe `RobotAgent` è una estensione della classe astratta `alice.tucson.api.Agent` ed è astratta essa stessa: rappresenta una sorta di modello comune dal quale partire per costruire le classi `DetectAgent` e `UnMineAgent` che rappresentano un modello effettivamente utilizzabile dei due tipi di agenti che realizzeranno il MAS.

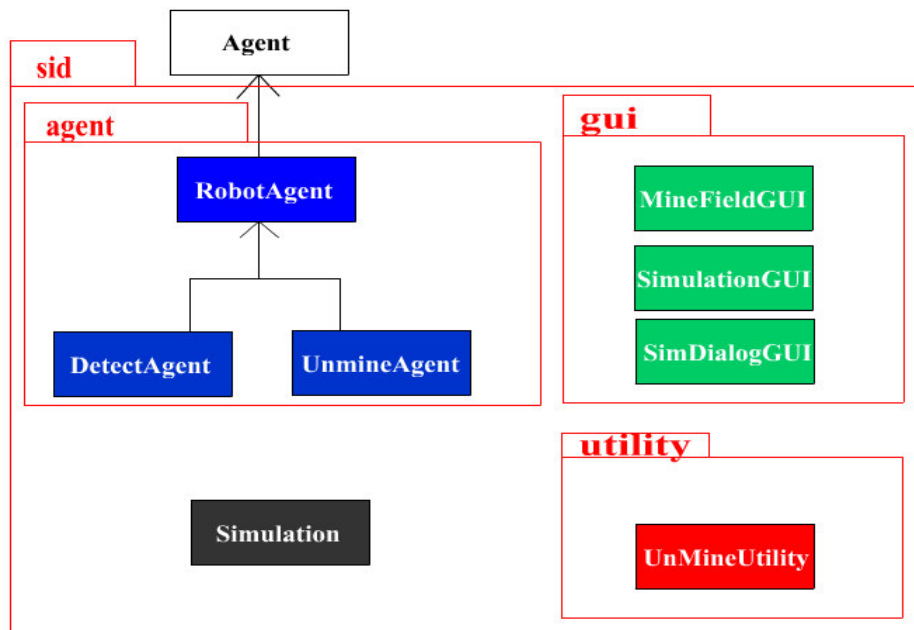


Figura 2.2: diagramma statico delle classi

Nello specifico, gli agenti utilizzati in questo sistema incapsuleranno un unico flusso di controllo e, ad ogni modo, rispetteranno le specifiche definite per gli “agenti Java” forniti dall’infrastruttura Tucson.

A run-time, entrambi i tipi di agenti faranno utilizzo di un motore tuProlog per quanto riguarda l’attività deliberativa e di pianificazione (caricato con una teoria specifica e diversa per unMine agents e detect agents).

2.4.2.2 Package sid.utility

In questo package è contenuta la sola classe `UnMineUtility` che funge da libreria per il motore TuProlog utilizzato all’interno degli agenti di tipo unMine: in particolare è provvista di un metodo che consente in maniera molto semplice (per TuProlog) il calcolo della distanza in linea d’aria tra due caselle della mappa date le rispettive coppie di coordinate.

2.4.2.3 Package sid.gui

Contiene tutte le classi relative all’interfaccia utente del sistema:

- `SimulationGUI`: questa classe rappresenta il pannello principale dell’applicazione contenente tutti i comandi per interagire con una simulazione in esecuzione (pulsanti Next e Stop) e per inizializzare nuove simulazioni.
- `MineFieldGUI`: gli oggetti di questa classe sono pannelli che consentono di rappresentare graficamente all’utente un campo minato relativamente ad una simulazione in esecuzione, mostrando le mine, gli agenti e i loro movimenti.

- `SimDialogGUI`: rappresenta la finestra di dialogo mediante la quale è possibile impostare tutti i parametri di una nuova simulazione in esecuzione.

2.4.2.4 Package sid: la classe Simulation

L'unica classe contenuta in questo package è la classe `Simulation`: si tratta di una classe molto importante.

Gli oggetti di questa classe rappresentano le simulazioni del MAS relativo a questo progetto: ogni oggetto di tale classe è quindi responsabile della creazione del campo minato e del lancio degli agenti, scegliendone nel contempo casualmente la posizione iniziale nel minefield; compiute queste operazioni gli agenti saranno chiaramente indipendenti e l'unico modo che l'oggetto di classe `Simulation` avrà per comunicare con essi, sarà esclusivamente tramite il loro `environment` ovvero tramite il centro di tuple di `Tucson` rappresentante il campo minato.

E' importante sottolineare come tutte le operazioni di interazione dell'utente con la simulazione (avanzamento del tempo, interruzione di una simulazione e eliminazione di mine dal campo minato) passeranno per tale oggetto che, a livello pratico, interagirà con gli agenti da lui lanciati in esecuzione utilizzando un `DefaultContext` per accedere al nodo `Tucson` locale in cui si trovano i centri di tuple oggetto di questo sistema.

2.4.3 Centri di tuple Tucson: rappresentazione del campo minato e del canale radio

Tramite Tucson sono stati realizzati i centri di tuple rappresentanti il campo minato e il canale radio utilizzato dagli agenti per la loro coordinazione e cooperazione.

2.4.3.1 Il centro di tuple MineField

Rappresenta la mappa del campo minato nel quale si muoveranno virtualmente gli agenti. Gli oggetti (mine e agenti) presenti nel campo minato sono rappresentati mediante tuple (fatti Prolog) del tipo riportato di seguito:

- `cell(1,2,mine)`. Indica che la casella di coordinate (1,2) è minata.
- `cell(3,7,agent(detect,d1))`. Indica che nella casella di coordinate (3,7) è presente un agente rilevatore di mine (di tipo detect quindi) il cui identificativo è d1.
- `cell(9,7,agent(unMine,u3))`. Indica che nella casella di coordinate (9,7) è presente un agente sminatore (di tipo unmine quindi) il cui identificativo è u3.

Come già spiegato in fase di progetto, per tale centro di tuple vale la CWA, per cui non vengono esplicitamente rappresentati i fatti relativi a caselle non minate o non occupate da agenti.

2.4.3.1.1 RespecT del centro di tuple

Come già spiegato in precedenza, ogni centro di tuple è responsabile anche della regolazione dell'interazione con gli agenti: questo in Tucson è realizzato tramite direttive RespecT, tramite le quali è possibile stabilire

le regole con le quali gli agenti potranno accedere al centro di tuple. Le direttive Respect per il tuple centre mineField sono riportate di seguito.

```
reaction(out (incUnminedMines),
(
    in_r (incUnminedMines),
    in_r (unminedMines (N)),
    N1 is N+1,
    out_r (unminedMines (N1))
)).

reaction(inp (mine (X, Y, AgName)),
(
    pre,
    in_r (cell (X, Y, mine)),
    out_r (mine (X, Y, AgName)),
    out_tc (radioChannel, free_tmp (X, Y, AgName))
)).

reaction(out (endSim),
(
    out_r (turn (-1, -1))
)).

reaction(out (agentEnded (AgentId)),
(
    out_r (turn (-1, -1)),
    in_r (no_agentEnded (N)),
    N1 is N+1,
    out_r (no_agentEnded (N1))
)).

reaction(out_r (no_agentEnded (N1)),
(
    rd_r (totAgentNumber (N)),
    N is N1,
    out_r (simulationEnded)
)).

reaction(out (maxDim (Coord, Value)),
(
    out_tc (radioChannel, maxDim (Coord, Value))
)).

reaction(out (agent (X, Y, Type, Name)),
(
    in_r (agent (X, Y, Type, Name)),
    in_r (cell (_, _, agent (Type, Name))),
    out_r (cell (X, Y, agent (Type, Name))),
    out_tc (radioChannel, agent_tmp (X, Y, Type, Name))
))
```

```
)).  
  
reaction(out(turn(No,NoAgent)),  
(  
    rd_r(totAgentNumber(NoAgent)),  
    in_r(turn(No,NoAgent)),  
    out_r(turn_end(No))  
)).
```

2.4.3.2 Il centro di tuple Tucson RadioChannel

E' la realizzazione dell'astrazione del canale radio, utilizzato dagli agenti per cooperare al fine di raggiungere il goal globale del MAS.

Come già precedentemente chiarito, per questo centro di tuple non è da ritenersi valida la CWA, per cui ogni cosa viene riportata esplicitamente. Di seguito vengono elencati alcune tuple "tipo" che possono trovarsi in questo centro di tuple durante una simulazione.

- `free(2,3,d1)`. Indica che la casella di coordinate (2,3) è stata esaminata e dichiarata libera da mine dall'agente d1.
- `mine(6,3,d2,none)`. Indica che la casella di coordinate (6,3) è stata esaminata e dichiarata minata dall'agente d2; inoltre questa mina non è ancora target di alcun agente sminatore e quindi può ancora essere presa potenzialmente in consegna da un agente sminatore.
- `mine(8,3,d4,u2)`. Indica che la casella di coordinate (8,3) è stata esaminata e dichiarata minata dall'agente d4; inoltre questa mina è stata presa in consegna dall'agente sminatore u2, che ha già calcolato un percorso per raggiungere la posizione.

- `agent(4,5,detect,d2)`. Indica che la casella di coordinate (4,5) è occupata dall'agente d2 di tipo detect.

Si sarà certamente notato che ogni tupla riporta anche l'identificativo dell'agente di tipo detect che ha ispezionato la relativa casella, dichiarandola come libera oppure come minata: questo è stato fatto pensando a sviluppi futuri del MAS, nei quali sarà ad esempio possibile simulare malfunzionamenti dell'apparato di rilevazione di un qualche agente di tipo detect; nel caso in cui venga accertato (in qualche modo) un malfunzionamento agli apparati di un agente rilevatore di mine, sarà possibile informare gli altri agenti (in special modo quelli di tipo unmine) di non tenere in considerazione le rilevazioni fatte dall'agente detect malfunzionante.

2.4.3.2.1 *RespecT*

Di seguito vengono riportate le direttive RespecT relative al tuple centre `radioChannel`.

```
reaction(out(agent_tmp(X,Y,Type,Name)),
(
    in_r(agent_tmp(X,Y,Type,Name)),
    in_r(agent(_,_,Type,Name)),
    out_r(agent(X,Y,Type,Name))
)).

reaction(in_r(free_tmp(X,Y,AgentName)), (
(
    rd_r(maxDim(x,Max_X)), rd_r(maxDim(y,Max_Y)),
    X>=0 , Y>=0 ,
    X<Max_X ,
    Y<Max_Y ,
    no_r(free(X,Y,_)),
    out_r(free(X,Y,AgentName))
)).

reaction(out_r(free(X,Y,AgentName)),
(
    in_r(mine(X,Y,_))
)).
```

```
reaction(out (free_tmp (X, Y, AgentName)),  
(  
    in_r (free_tmp (X, Y, AgentName))  
)).  
  
reaction(out (mine_tmp (X, Y, AssignedUnmineAgent)),  
(  
    in_r (mine_tmp (X, Y, AssignedUnmineAgent))  
)).  
  
reaction(in_r (mine_tmp (X, Y, AssignedUnmineAgent)),  
(  
    no_r (mine (X, Y, _)),  
    out_r (mine (X, Y, AssignedUnmineAgent))  
)).  
  
reaction(inp (rd_all (T, L)),  
(  
    pre,  
    out_r (rdall_collect (T, []))  
)).  
  
reaction(out_r (rdall_collect (T, L)),  
(  
    current_tuple (rdall_collect (T1, _)),  
    in_r (T1), in_r (rdall_collect (T, L)),  
    out_r (rdall_collect (T, [T1|L]))  
)).  
  
reaction(out_r (rdall_collect (T, _)),  
(  
    current_tuple (rdall_collect (T1, _)),  
    no_r (T1), in_r (rdall_collect (T, L)),  
    out_r (rd_all (T, L)),  
    out_r (rdall_restore (L))  
)).  
  
reaction(out_r (rdall_restore ([])),  
(  
    in_r (rdall_restore ([]))  
)).  
  
reaction(out_r (rdall_restore ([H|T])),  
(  
    in_r (rdall_restore ([H|T])),  
    out_r (H),  
    out_r (rdall_restore (T))  
)).
```

2.4.4 Turno tipo degli agenti

Si descrive brevemente il turno tipo di ogni agente.

2.4.4.1 Agenti di tipo detect

Il turno di un agente di tipo detect si articola nel seguente modo:

- L'agente ispeziona le celle adiacenti, e ne comunica lo stato agli altri agenti utilizzando il tuple centre radioChannel.
- L'agente, utilizzando un engine Tuprolog, decide in quale delle celle libere attorno muoversi. La nuova casella nella quale l'agente andrà a muoversi verrà "accodata" alla teoria Prolog dell'engine, andando ad aggiungersi all'albero rappresentante il cammino seguito dall'agente fino a quel momento.
- Inoltre, al fine di evitare collisioni, è importante che prima di effettuare ogni movimento, l'agente controlli se la casella target del movimento sia o meno occupata da un altro agente: questo per evitare spiacevoli collisioni tra agenti.

2.4.5 Agenti di tipo unMine

Il turno di un agente di tipo unmine è articolato come segue:

- L'agente controlla se nel proprio Prolog engine sia presente un piano per raggiungere una mina target: per piano qui si intende una successione di coppie di coordinate di caselle nelle quali muoversi in sequenza al fine di giungere alla mina target.
- Nel caso sia presente un piano l'agente si muoverà nella successiva casella prevista dal piano: nel caso questa casella sia quella contenente la mina, l'agente provvederà al disinnescamento.

- Nel caso l'agente non abbia alcun piano provvederà al recupero di tutte le mine segnalate a lui vicine non ancora assegnate ad altri agenti (e delle relative caselle a lui vicine dichiarate libere): per ogni mina provvederà, tramite il proprio engine Prolog a verificare l'esistenza di un percorso verso la mina in esame. Nel caso in cui tale percorso esista, la mina diventerà il target dell'agente e il piano verrà eseguito a partire dal turno successivo; l'agente provvederà altresì ad avvisare gli altri unmine agents (tramite il tuple centre radioChannel) del fatto che da ora quella mina sarà assegnata a lui e quindi esso stesso sarà l'unico responsabile per il suo disinnesco.

2.5 Aspetti “tecnici” e implementativi della simulazione

A livello pratico il simulatore realizzato per il MAS in questione, è soggetto ad alcune limitazioni.

Per quanto riguarda il numero di agenti presenti nella simulazione è stato previsto un limite massimo di 4 agenti di tipo unmine e 4 di tipo detect (8 agenti in tutto quindi): questo perché, visto che gli agenti risiederanno tipicamente su di un'unica macchina (di solito monoprocessore), non si vuole degradare troppo la qualità della simulazione stessa; è bene sottolineare che si tratta comunque solamente di un limite di carattere pratico e non assolutamente di tipo teorico.

Per ciò che riguarda il calcolo di un piano verso una mina distante da parte degli unmine agents, è possibile avere qualche rallentamento del simulatore, da ascrivere sempre ai motivi espliciti al sottoparagrafo precedente.