

Projet Dédale

FOSYMA

1. Structure d'un agent
2. Protocoles de communication
3. Détail des comportements
4. Application à des exemples

Encadrants :
Cédric Herpson
Aurélie Beynier
Nicolas Maudet

Structure d'un agent

- Dualité interaction / logique
- Interaction :
 - Les autres : observation, écoute
 - Moi : déplacement, communication
- Logique :
 - Désirs : discuter, découvrir, chasser
 - Connaissances : traitement des connaissances accumulées
 - Décision : prise de décision
 - Exécution d'un comportement

Connaissances AgentKnowledge

- Que sais-je de cet agent ?
 - Accumulation de connaissances, traitement (unpacking) de messages.
- Que faisait-il, où et quand ?
- Que sait-il de la topologie ?
- A quel point veux-je discuter avec lui ?
 - Compromis désir, distance et différence de connaissances

$$U(A) = 1000 \times (1 + wantToMeet) \times \frac{0.5 * \Delta edges + \Delta nodes}{distance^2}$$

LA COMMUNICATION

ListenBehaviour

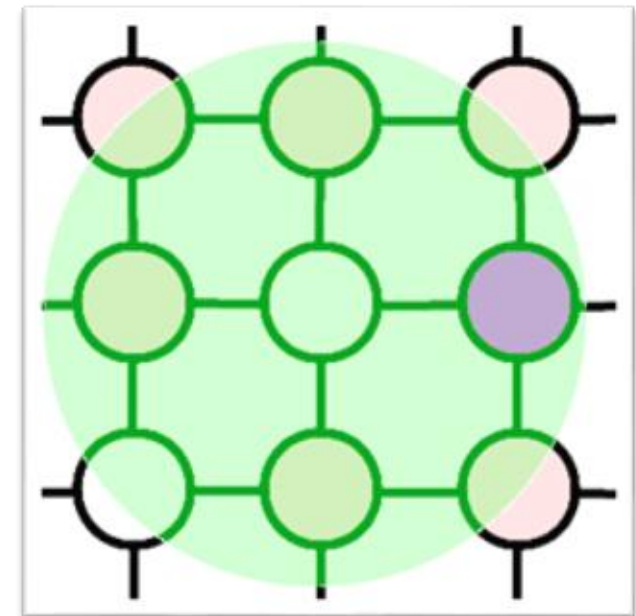
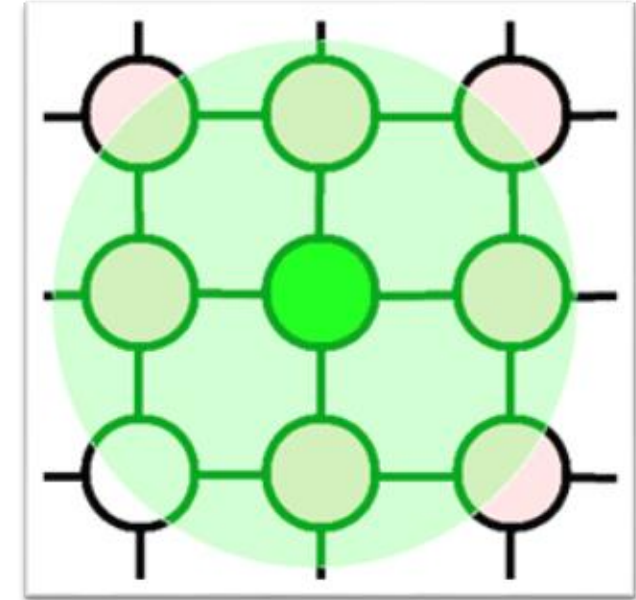
PingPositionBehaviour

- Où suis-je ? Quest-ce que je sens ?
Que fais-je ?

ShareMapBehaviour

SharePathBehaviour

- Où vais-je ?



Comportements : an FSM to rule them all

Interactions

Communication :

- ListenBehaviour (*CyclicBehaviour*) écoute et traite les messages reçus.
- PingPositionBehaviour (*CyclicBehaviour*) envoie des nouvelles aux autres agents
- ShareMapBehaviour (*OneShotBehaviour*) envoie ma carte de façon pertinente aux autres agents
- SharePathBehaviour (*OneShotBehaviour*) envoie mon chemin aux autres agents

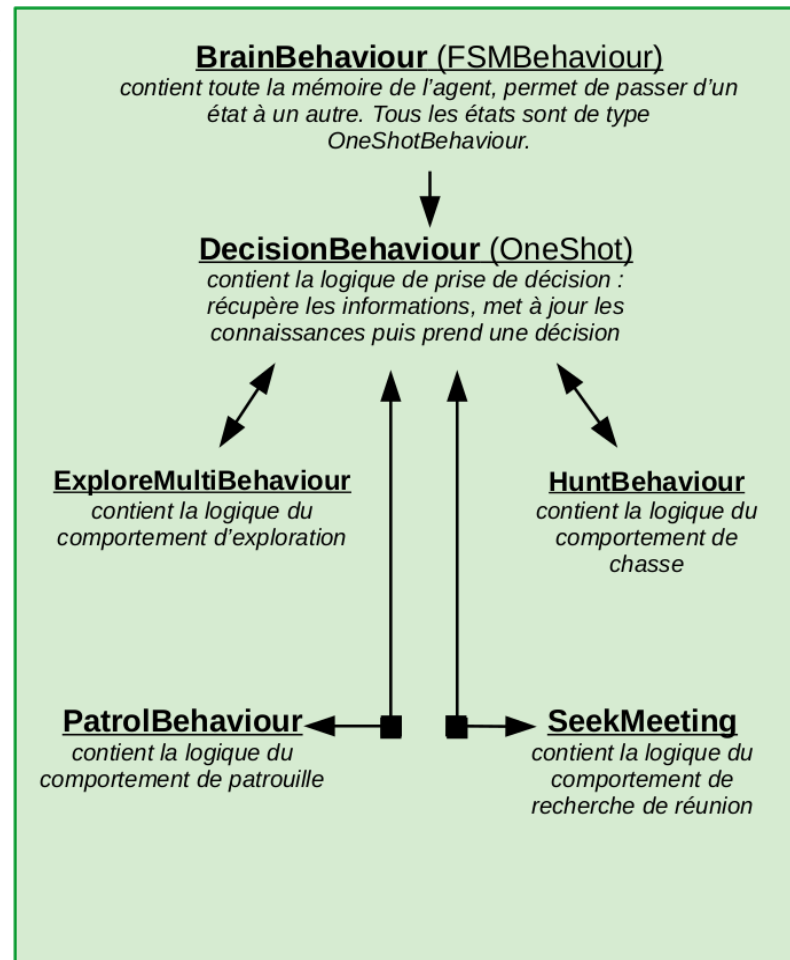
Déplacements :

- ExploMultiAgent.moveToIntention se déplace dans l'environnement
- ExploMultiAgent.discover découvre et observe les noeuds alentours

Détection :

- ExploMultiAgent.getAgentsAround met à jour le repérage des agents alentours
- ExploMultiAgent.getStenchAround met à jour l'odeur des intrus détectée

Logique



- Dualité interaction / logique
- Le cerveau (FSM) contient l'accumulation des connaissances
- Prise de décision cyclique :
 - Décision (Que faire ?)
 - Réléxion (Comment ?)
 - Exécution (J'essaie)
- Cerveau et "Corps" en interaction

DecisionBehaviour

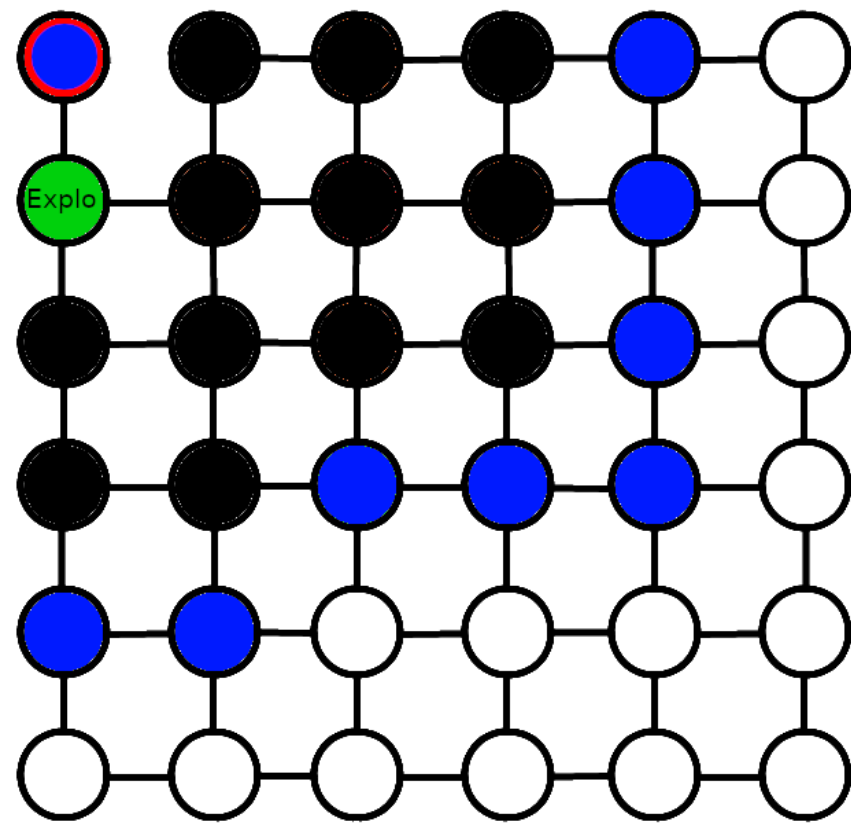
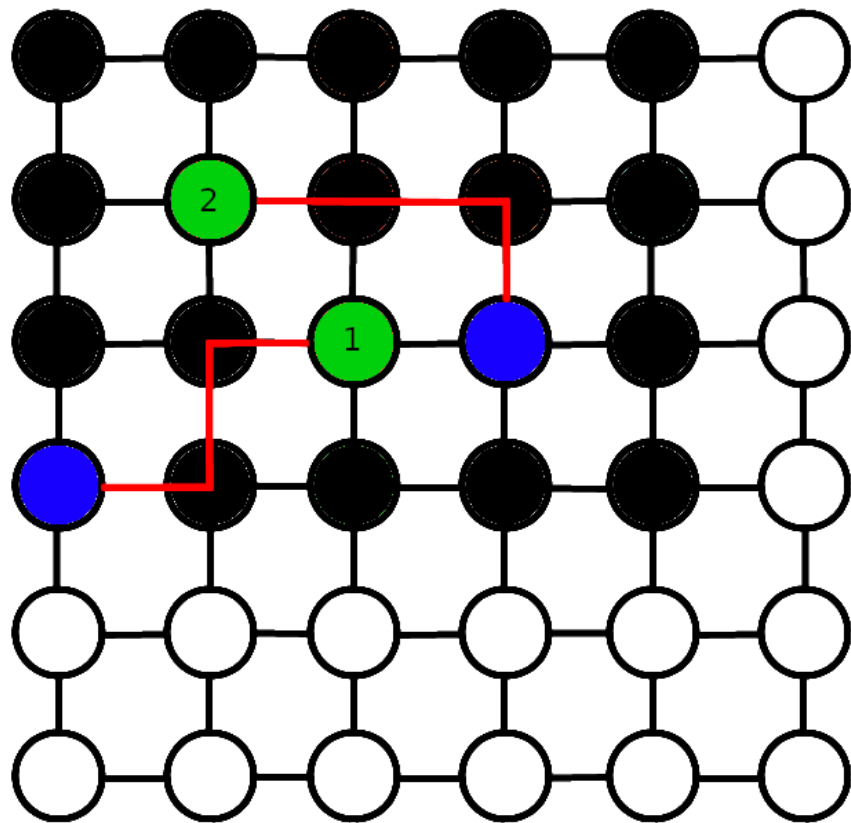
```
@Override
public void action() {
    // Retrieve the necessary information to take a decision
    this.retrieveInformation();
    // Takes a decision based on the retrieved information and the accumulated
    // knowledge
    this.decision = this.takeDecision();

    if (!this.decision.equals(brain.getLastDecision())) {
        ((ExploreMultiAgent) this.myAgent).sayConsole("I took the " + decision + " decision.");
    }
}
```

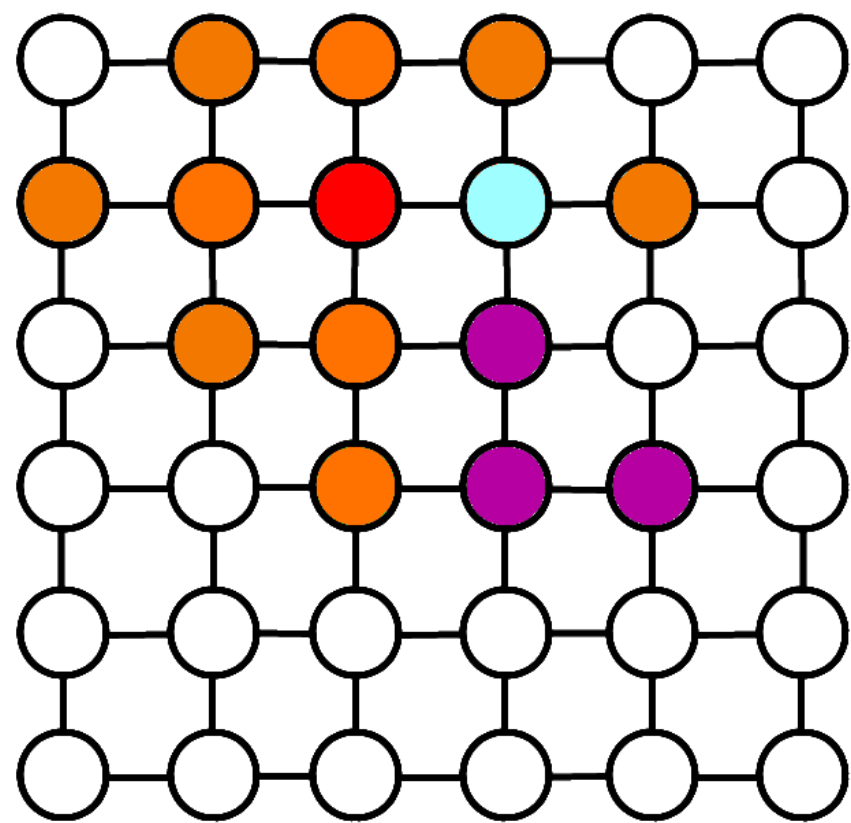
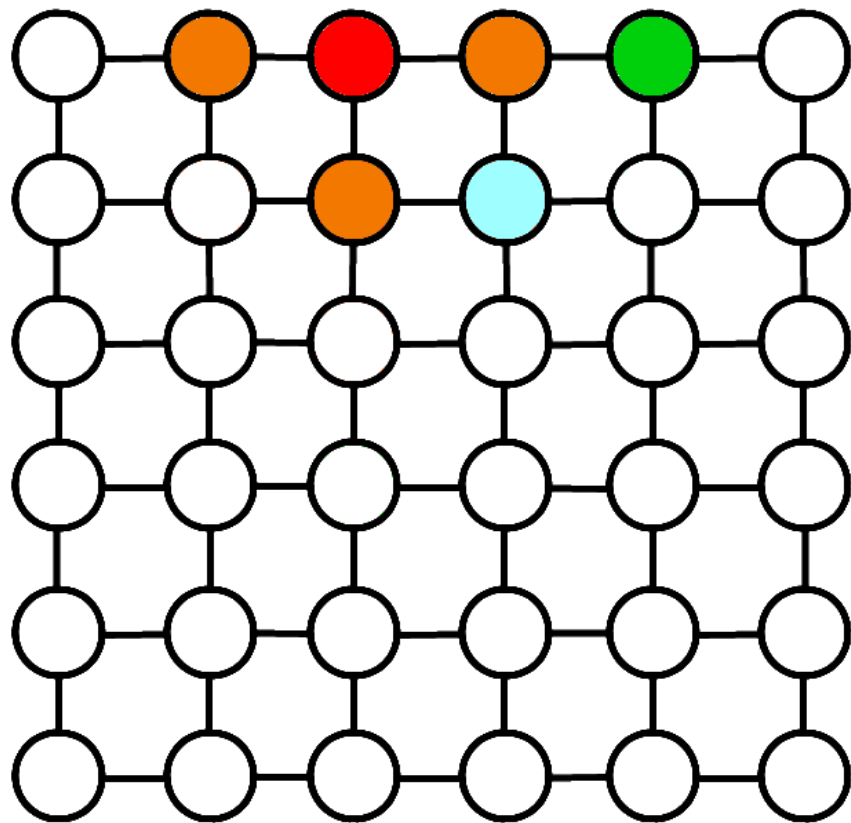
- Que veux-je :
 - Connaissances accumulées
- Que sais-je (DecisionBehaviour.retrieveInformation):
 - J'interroge l'agent (observations), je traite les connaissances
- Je prends ma décision (DecisionBehaviour.takeDecision).

Coût algorithmique
lourd

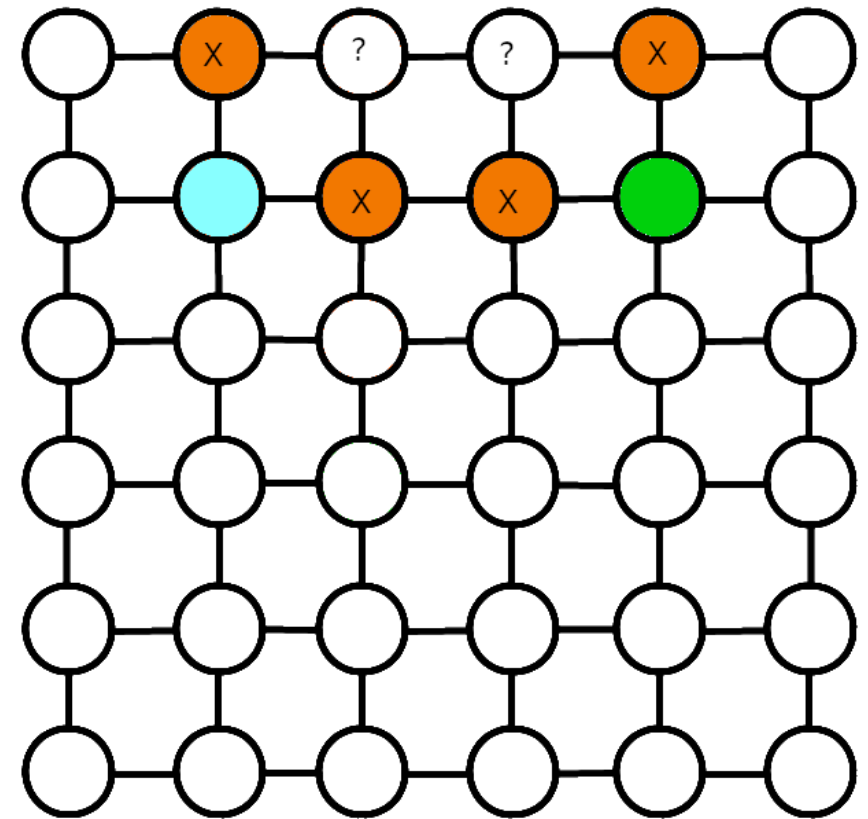
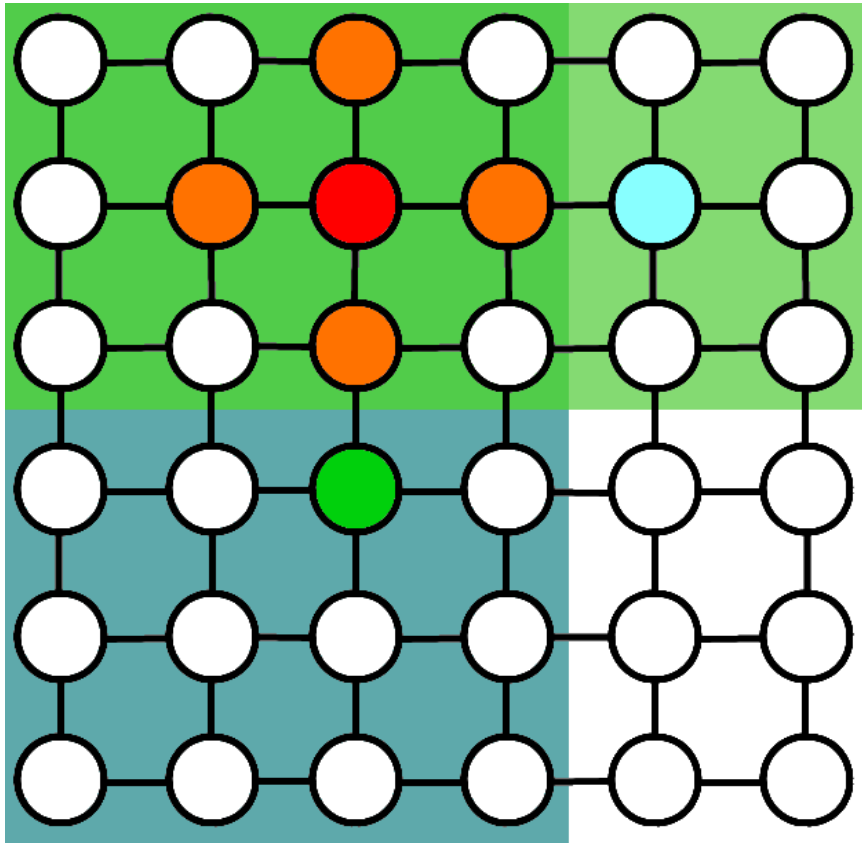
L'exploration



La Patrouille et la chasse



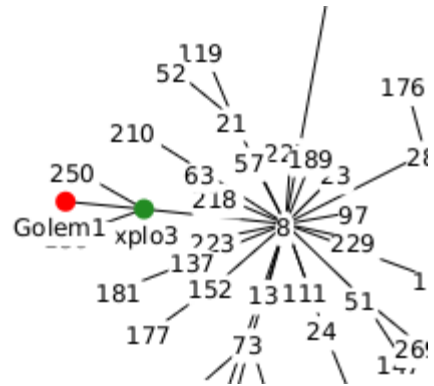
Les améliorations



SeekMeeting

- Pas d'accusé de réception et envoi des messages lourds aux agents alentours.
- Idée :
 - Rencontrer un agent pour qu'il me transmette les données lourdes.
 - Je m'en vais lorsque je n'ai plus besoin de lui.
- Problèmes :
 - Surcharge des messages lorsque deux agents se suivent.
 - Transmission de mes connaissances uniquement -> amélioration : transmettre l'horloge matricielle.

Examples



|Explo3: I finished the game! The intruders have been blocked.

