Beliefs representation and management

Autonomous Software Agents

A.A. 2022-2023

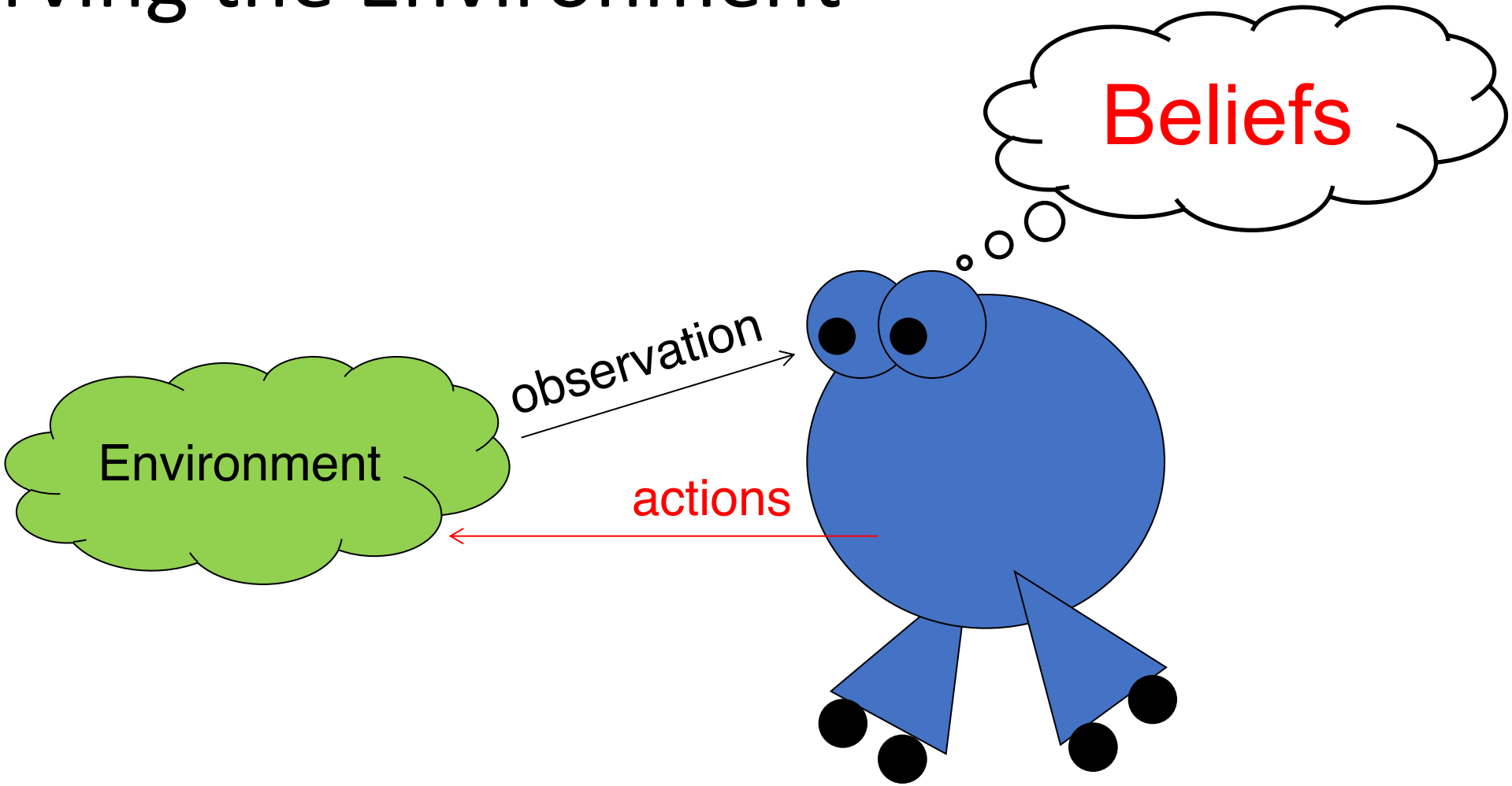
Prof. Paolo Giorgini

Dr. Marco Robol

UNIVERSITY OF TRENTO - Italy

Department of Information and Communication Technology

Observing the Environment



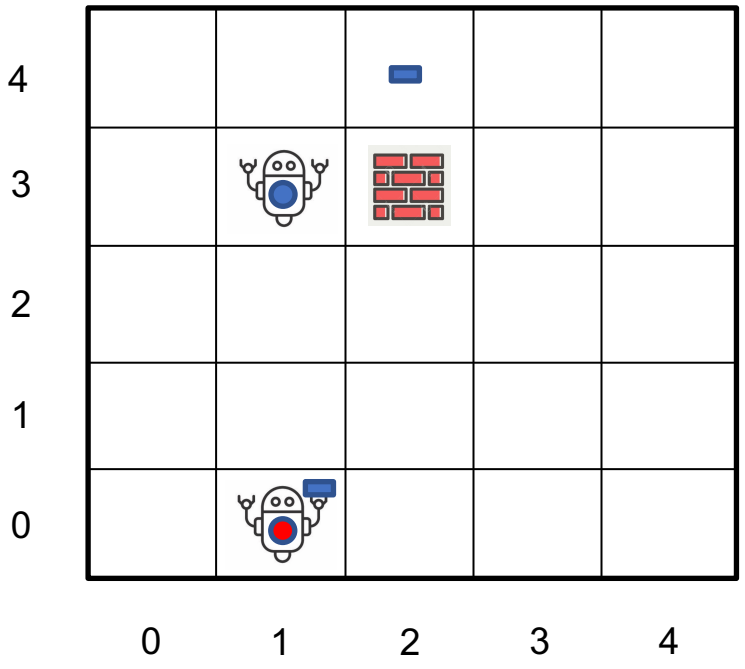
From sensors' data to Belief

- An agent acquires data from its sensors
  - E.g., temperature from the thermostat or the agent's position from GPS

Time	Temperature	X	Y
09:00:00	8	3	3
10:00:00	9	3	4
11:00:00	11	3	5
12:00:00	13	4	6

  - Data can be stored as they are acquired
    - Acquisition time tells us actual values and it draws data evolution

Data completeness and correctness



Agent “Ag_1” (or ”Ag_2”)

Positions

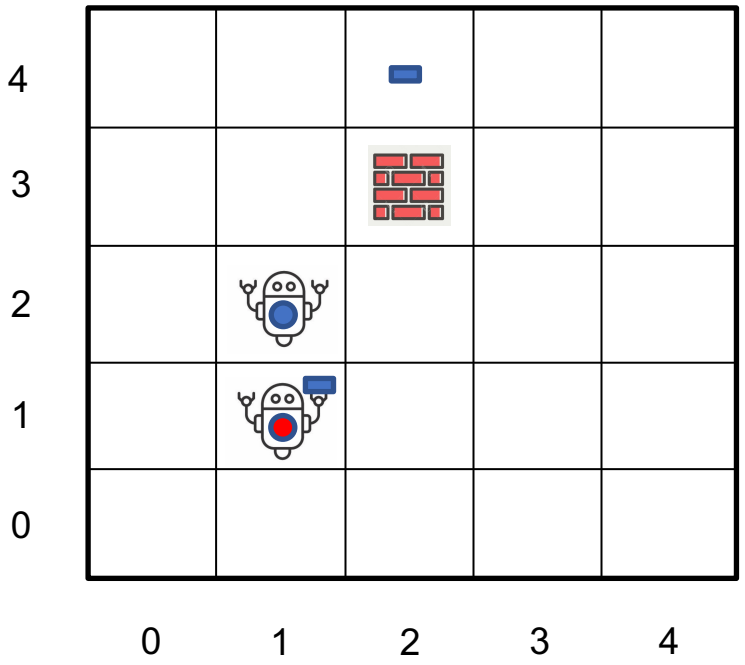
Time	Object	X	Y
1	Ag_1	1	0
1	Ag_2	1	3
1	Obst_1	2	3
1	Pack_1	2	4

Carry

Time	Agent	Pack
1	Ag_1	Pack_2

```
B = { In(Ag_1,1,0), In(Ag_2,1,3), In(Obst_1,2,3),
      In(Pack_1,2,4), carry(Ag_1,Pack_2) }
```

Data updating



Positions

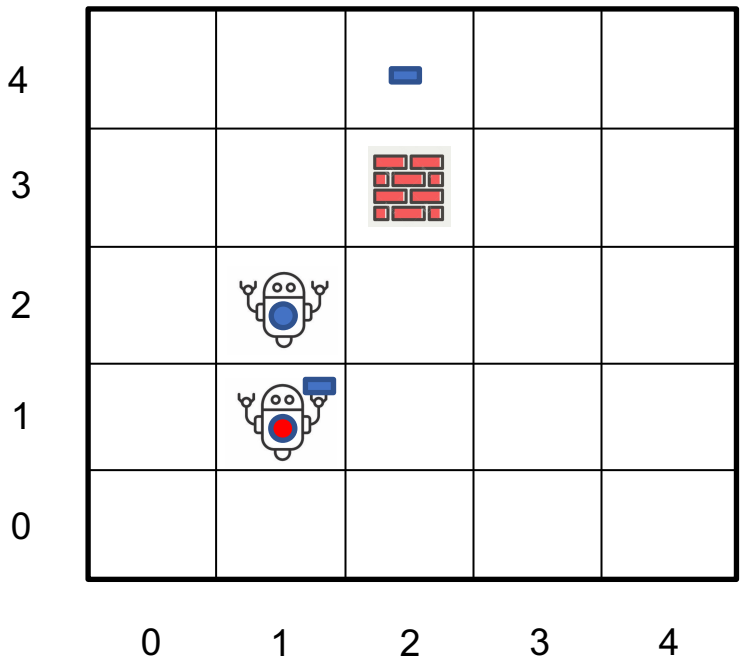
Time	Object	X	Y
~~1~~	~~Ag_1~~	~~1~~	~~0~~
~~1~~	~~Ag_2~~	~~1~~	~~3~~
~~1~~	~~Obst_1~~	~~2~~	~~3~~
~~1~~	~~Pack_1~~	~~2~~	~~4~~
2	Ag_1	1	1
2	Ag_2	1	2
2	Obst_1	2	3
2	Pack_1	2	4

Carry

Time	Agent	Pack
~~1~~	~~Ag_1~~	~~Pack_2~~
2	Ag_1	Pack_2

```
B = { In(Ag_1,1,1), In(Ag_2,1,2), In(Obst_1,2,3),
      In(Pack_1,2,4), carry(Ag_1,Pack_2) }
```

Derived Beliefs



Positions

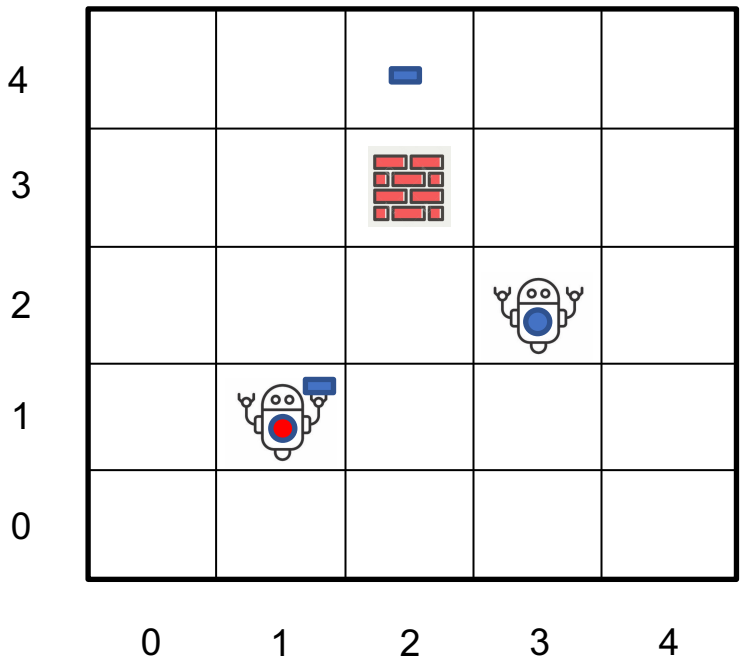
Time	Object	X	Y	Move
1	Ag_1	1	0	No
1	Ag_2	1	3	No
1	Obst_1	2	3	No
1	Pack_1	2	4	No
2	Ag_1	1	1	UP
2	Ag_2	1	2	DOWN
2	Obst_1	2	3	No
2	Pack_1	2	4	No

Carry

Time	Agent	Pack
1	Ag_1	Pack_2
2	Ag_1	Pack_2

```
B = { In(Ag_1,1,1), In(Ag_2,1,2), In(Obst_1,2,3),
      In(Pack_1,2,4), carry(Ag_1,Pack_2),
      move(Ag_1,UP),move(Ag_2,DOWN) }
```

Managing Inconsistencies



Positions

Time	Object	X	Y	Move
1	Ag_1	1	0	No
1	Ag_2	1	3	No
1	Obst_1	2	3	No
1	Pack_1	2	4	No
2	Ag_1	1	1	UP
2	Ag_2	1	2	DOWN
2	Obst_1	2	3	No
2	Pack_1	2	4	No
3	Ag_1	1	1	No
3	Ag_2	3	2	RIGHT
3	Obst_1	2	3	No
3	Pack_1	2	4	No

Carry

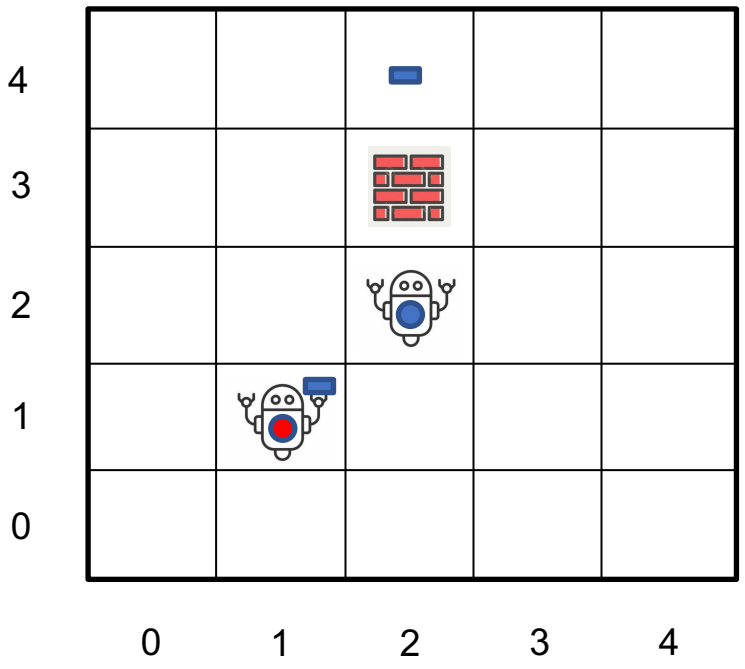
Time	Agent	Pack
1	Ag_1	Pack_2
2	Ag_1	Pack_2

Robot cannot move of two tiles in one step

Inconsistencies

- Inconsistencies can arise for several reasons
  - Sensors can send wrong data
  - Predefined rules are not anymore valid (evolution of the environment)
    - Now the robot can move of two tiles
  - Data are provided by other agents
    - They might lie or they could have wrong beliefs
- How much is it critical to solve the inconsistency?
  - Can we wait a little bit and see what happen?
- How to solve them ?
  - Many different ways can be applied
    - `t=2 :In(Ag_2,1,2), move(Ag_2,DOWN)`
    - `t=3 :In(Ag_2,3,2), move(Ag_2,RIGHT)` ---→ `t=3 :In(Ag_1,2,2), move(Ag_1,RIGHT)`
- Policies/strategies to solve and manage inconsistencies should be part of the design

Managing Inconsistencies



Positions

Time	Object	X	Y	Move
1	Ag_1	1	0	No
1	Ag_2	1	3	No
1	Obst_1	2	3	No
1	Pack_1	2	4	No
2	Ag_1	1	1	UP
2	Ag_2	1	2	DOWN
2	Obst_1	2	3	No
2	Pack_1	2	4	No
3	Ag_1	1	1	No
3	Ag_2	2	2	RIGHT
3	Obst_1	2	3	No
3	Pack_1	2	4	No

Carry

Time	Agent	Pack
1	Ag_1	Pack_2
2	Ag_1	Pack_2

Solving the inconsistency
(most likely position for Ag_2?)

Managing inconsistencies

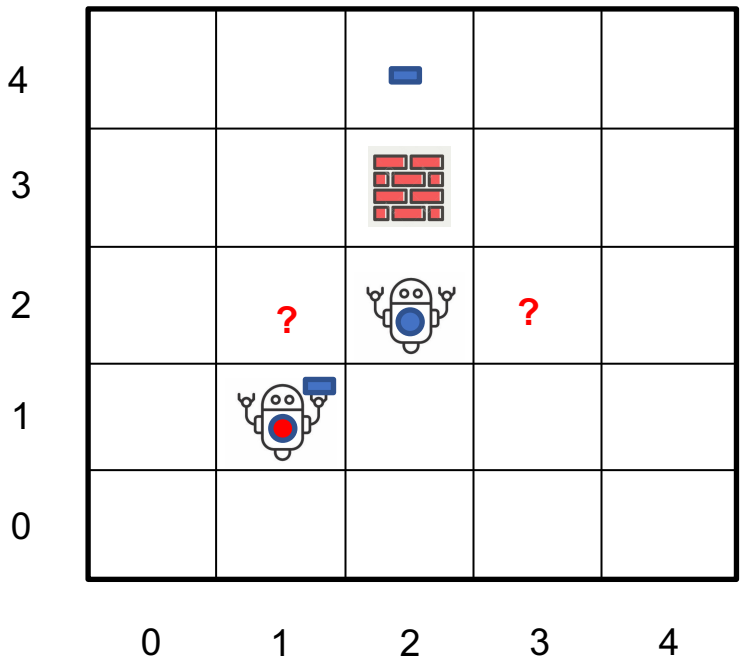
```
t=1: In(Ag_2,1,3)
t=2: In(Ag_2,1,2), move(Ag_2,DOWN)
t=3: In(Ag_2,3,2), move(Ag_2,RIGHT)
```

Possibile consistent sets:

```
t=1: In(Ag_2,1,3)
t=2: In(Ag_2,1,2), move(Ag_2,DOWN)
```

```
t=3: In(Ag_2,3,2), move(Ag_2,RIGHT)
```

Another example



t=1: In(Ag_2,**2**,**2**)
t=2: In(Ag_2,1,2), move(Ag_2,LEFT)
t=3: In(Ag_2,3,2), move(Ag_2,RIGHT)

Possibile consistent sets:

S_1

t=1: In(Ag_2,2,2)
t=2: In(Ag_2,1,2), move(Ag_2,LEFT)

S_2

t=1: In(Ag_2,2,2)
t=3: In(Ag_2,3,2), move(Ag_2,RIGHT)

More on the example

S_1

```
t=1: In(Ag_2,2,2)
t=2: In(Ag_2,1,2), move(Ag_2,LEFT)
```

what about t=3 ?

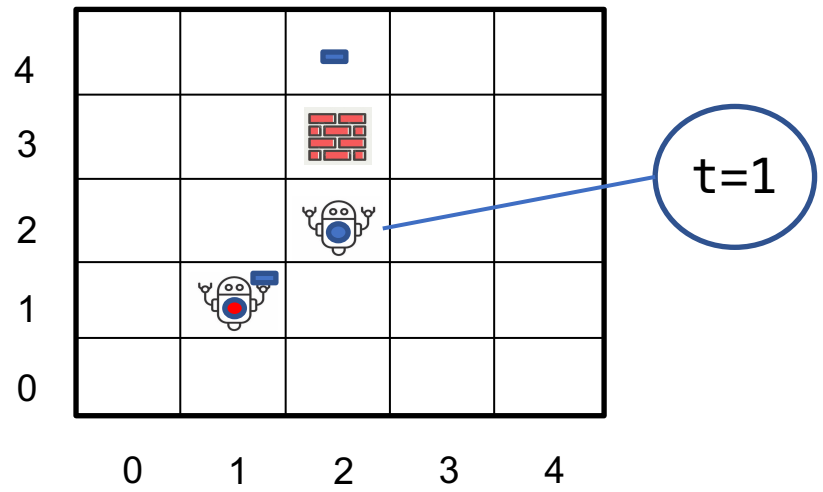
t=3: In(Ag_2,1,2) v In(Ag_2,0,2) v
In(Ag_2,1,3) v In(Ag_2,2,2)

S_2

```
t=1: In(Ag_2,2,2)
t=3: In(Ag_2,3,2), move(Ag_2,RIGHT)
```

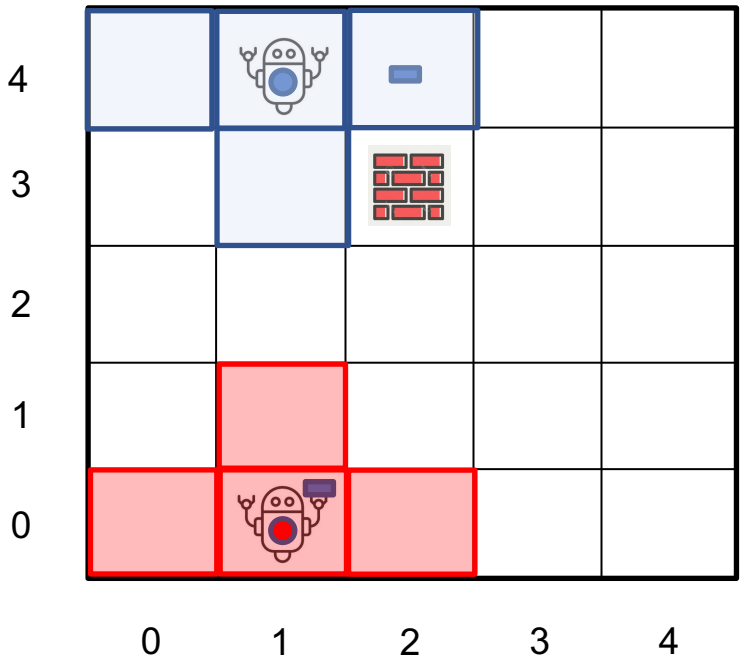
what about t=2 ?

t=2: In(Ag_2,2,2) v In(Ag_2,3,2)



After we had choosen beween S1 and S2, should we update beliefs for t=3 and t=2, respectivly?

Partial view of the environment



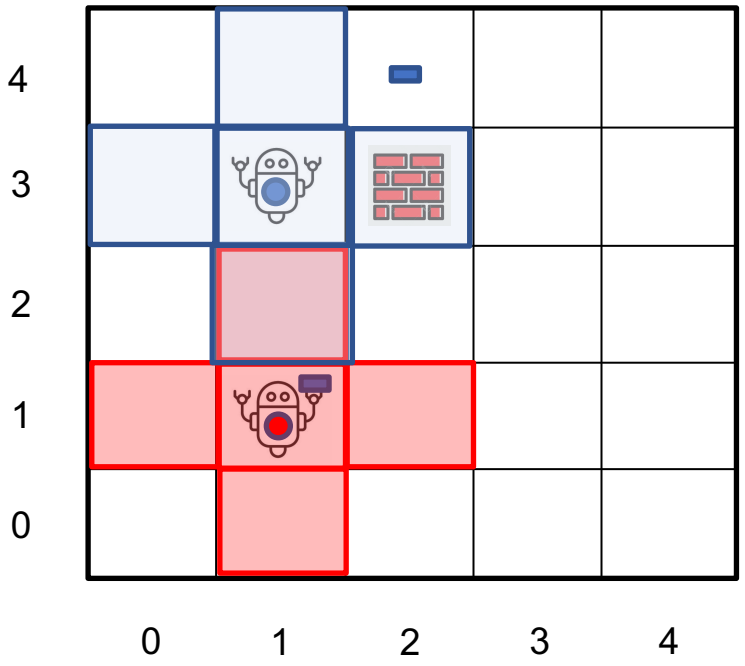
Agent 2

Time	Object	X	Y
1	Ag_2	1	4
1	Pack_1	2	4

Agent 1

Time	Object	X	Y
1	Ag_1	1	0

Partial view of the environment



Agent 2

Time	Object	X	Y
1	Ag_2	1	4
1	Pack_1	2	4
2	Obst_1	2	3
2	Ag_2	1	3

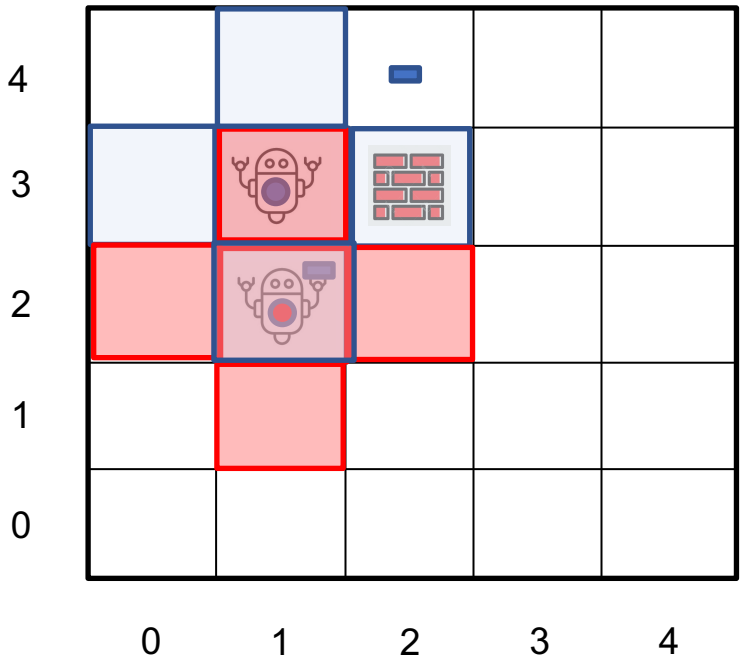
Agent 1

Time	Object	X	Y
1	Ag_1	1	0
2	Ag_1	1	1

At time t=2, is Pack_1 in (2,4)?

- YES ? NO? MAYBE? LIKELY?

Partial view of the environment



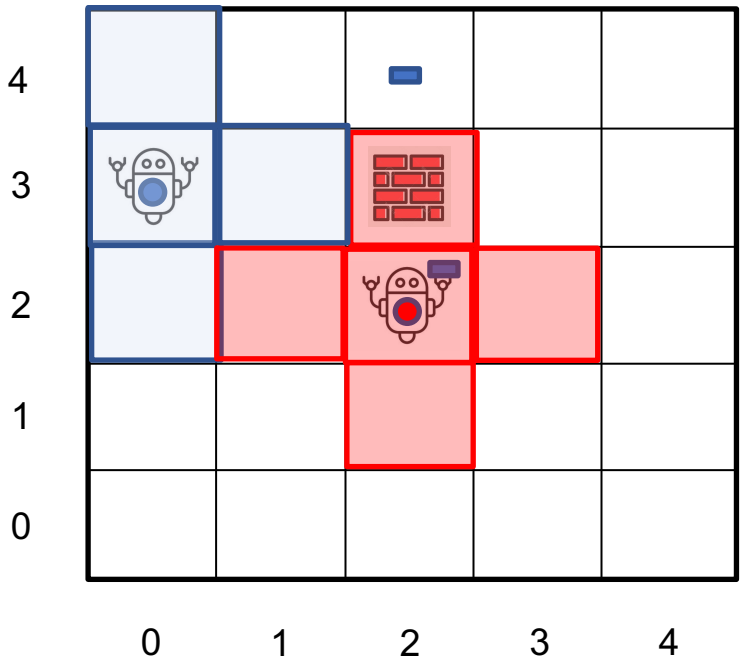
Agent 2

Time	Object	X	Y
1	Ag_2	1	4
1	Pack_1	2	4
2	Obst_1	2	3
2	Ag_2	1	3
3	Obst_1	2	3
3	Ag_2	1	3
3	Ag_1	1	2

Agent 1

Time	Object	X	Y
1	Ag_1	1	0
2	Ag_1	1	1
3	Ag_1	1	2
3	Ag_2	1	3

Partial view of the environment



Agent 2

Time	Object	X	Y
1	Ag_2	1	4
1	Pack_1	2	4
2	Obst_1	2	3
1	Ag_2	1	3
3	Obst_1	2	3
3	Ag_2	1	3
3	Ag_1	1	2
4	Ag_2	0	3

Agent 1

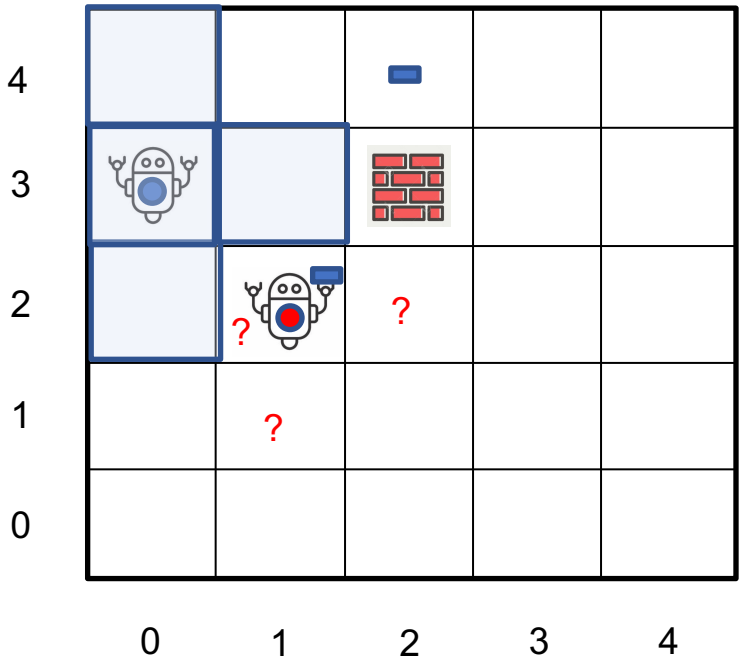
Time	Object	X	Y
1	Ag_1	1	0
2	Ag_1	1	1
3	Ag_1	1	2
3	Ag_2	1	3
4	Ag_1	2	2
4	Obst_1	2	3

They are not there anymore

As in the previous case

Agent 2

Time	Object	X	Y
1	Ag_2	1	4
1	Pack_1	2	4
2	Obst_1	2	3
2	Ag_2	1	3
3	Obst_1	2	3
3	Ag_2	1	3
3	Ag_1	1	2
4	Ag_2	0	3

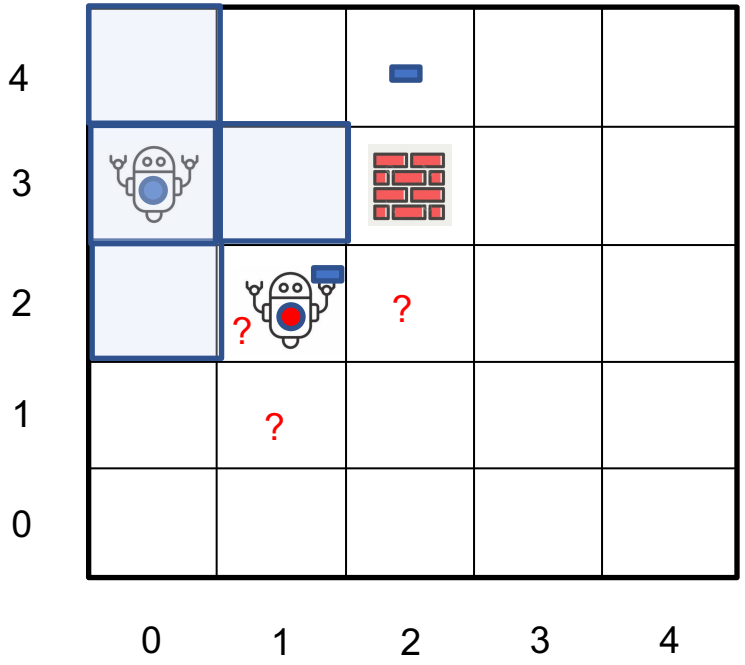


→ `t=4: In(Ag_1,1,2) v In(Ag_1,1,1) v In(Ag_1,2,2)`

Beliefs models

- Several beliefs models can be used
  - No memory: only beliefs based on current data
  - With memory: beliefs based on current data and keeping true not updated beliefs
  - With uncertainty
    - “the probability pack_1 that I saw long time ago is still in position (x,y) is very low”
    - ”the probability obst_1 is in position (x,y) is 1” (obst_1 is a wall a nobody can move it)
    - “I saw Ag_1 going in the direction of Pack_1, the probability Pack_1 is in position (x,y) is very low”
    - “Ag_1 was in position (x,y) and it was moving, the probability it is still there is very low”

In our example



Agent 2

Time	Object	X	Y
1	Ag_2	1	4
1	Pack_1	2	4
2	Obst_1	2	3
2	Ag_2	1	3
3	Obst_1	2	3
3	Ag_2	1	3
3	Ag_1	1	2
4	Ag_2	0	3

→ `t=4: In(Ag_1,1,2) v In(Ag_1,1,1) v In(Ag_1,2,2)`

Time	Object	X	Y	Probability
4	Ag_2	0	3	1
4	Ag_1	1	2	0.33
4	Ag_1	1	1	0.33
4	Ag_1	2	2	0.33

Beliefs and Introspective abilities

- What about beliefs concerning my intentions, desires, plans, actions? (Introspection)
  - If “I intend G, do I believe I intend to achieve G?”
  - Necessary to reasoning about intentions
    - If there is the opportunity to `pick_up(Pack_2)`
    - What about my current intensions? `B:{Intend(pick_up(Pack_1))}`
    - Are `Intend(pick_up(Pack_1))` and `Intend(pick_up(Pack_2))` consistent?
  - Not easy to implement
  - Synchronization between Beliefs – Intentions – Plans
  - Easy to get into self-contradictory reasoning

Beliefs about other agents' mental states

- Beliefs about other agents' beliefs
  - I belief you belief - `B:{belief(Ag_2,In(Pack_1,2,4))}`
  - Important for coordination, negotiation, and competition
  - We will se more on agents' communication and the speech act theory
- Beliefs about other agents' Intentions
  - I belief you Intend - `B:{intend(Ag_2,pick_up(Pack_1))}`
  - Reasoning about the others' behaviours (coordination, negotiation, and competition)
  - Prediction of others' actions – need to explore possible plans
- Beliefs about other agents' plans
  - I belief you have the plan of `B:{plan(Ag_2,[move(UP),move(RIGHT),pick_up(Pack_1)]}`
  - Usually related to intentions but not always intensions are known
    - I know it will follow that path, but I don't know why
  - Prediction of others' actions

Lets try on Deliveroojs

- No memory
- With memory
- Derived Beliefs
  - Elaborating rewards
- Managing uncertainty
- Not need to manage Inconsistencies