

**function** HYBRID-WUMPUS-AGENT(*percept*) **returns** an *action*

**inputs:** *percept*, a list, [*stench*,*breeze*,*glitter*,*bump*,*scream*]

**persistent:** *KB*, a knowledge base, initially the atemporal “wumpus physics”

*t*, a counter, initially 0, indicating time

*plan*, an action sequence, initially empty

TELL(*KB*, MAKE-PERCEPT-SENTENCE(*percept*, *t*))

TELL the *KB* the temporal “physics” sentences for time *t*

$safe \leftarrow \{[x, y] : \text{ASK}(KB, OK_{x,y}^t) = true\}$

**if** ASK(*KB*, *Glitter*<sup>*t*</sup>) = *true* **then**

$plan \leftarrow [Grab] + \text{PLAN-ROUTE}(current, \{[1,1]\}, safe) + [Climb]$

**if** *plan* is empty **then**

$unvisited \leftarrow \{[x, y] : \text{ASK}(KB, L_{x,y}^{t'}) = false \text{ for all } t' \leq t\}$

$plan \leftarrow \text{PLAN-ROUTE}(current, unvisited \cap safe, safe)$

**if** *plan* is empty and ASK(*KB*, *HaveArrow*<sup>*t*</sup>) = *true* **then**

$possible\_wumpus \leftarrow \{[x, y] : \text{ASK}(KB, \neg W_{x,y}) = false\}$

$plan \leftarrow \text{PLAN-SHOT}(current, possible\_wumpus, safe)$

**if** *plan* is empty **then** // no choice but to take a risk

$not\_unsafe \leftarrow \{[x, y] : \text{ASK}(KB, \neg OK_{x,y}^t) = false\}$

$plan \leftarrow \text{PLAN-ROUTE}(current, unvisited \cap not\_unsafe, safe)$

**if** *plan* is empty **then**

$plan \leftarrow \text{PLAN-ROUTE}(current, \{[1, 1]\}, safe) + [Climb]$

*action*  $\leftarrow$  POP(*plan*)

TELL(*KB*, MAKE-ACTION-SENTENCE(*action*, *t*))

*t*  $\leftarrow$  *t* + 1

**return** *action*

**function** HYBRID-WUMPUS-AGENT(*percept*) **returns** an *action*

**inputs:** *percept*, a list, [*stench*,*breeze*,*glitter*,*bump*,*scream*]

**persistent:** *KB*, a knowledge base, initially the atemporal “wumpus physics”

*t*, a counter, initially 0, indicating time

*plan*, an action sequence, initially empty

TELL(*KB*, MAKE-PERCEPT-SENTENCE(*percept*, *t*))

TELL the *KB* the temporal “physics” sentences for time *t*

$safe \leftarrow \{[x, y] : \text{ASK}(KB, OK_{x,y}^t) = true\}$

**if** ASK(*KB*, *Glitter*<sup>*t*</sup>) = *true* **then**

$plan \leftarrow [Grab] + \text{PLAN-ROUTE}(current, \{[1,1]\}, safe) + [Climb]$

**if** *plan* is empty **then**

$unvisited \leftarrow \{[x, y] : \text{ASK}(KB, L_{x,y}^{t'}) = false \text{ for all } t' \leq t\}$

$plan \leftarrow \text{PLAN-ROUTE}(current, unvisited \cap safe, safe)$

**if** *plan* is empty and ASK(*KB*, *HaveArrow*<sup>*t*</sup>) = *true* **then**

$possible\_wumpus \leftarrow \{[x, y] : \text{ASK}(KB, \neg W_{x,y}) = false\}$

$plan \leftarrow \text{PLAN-SHOT}(current, possible\_wumpus, safe)$

**if** *plan* is empty **then** // no choice but to take a risk

$not\_unsafe \leftarrow \{[x, y] : \text{ASK}(KB, \neg OK_{x,y}^t) = false\}$

$plan \leftarrow \text{PLAN-ROUTE}(current, unvisited \cap not\_unsafe, safe)$

**if** *plan* is empty **then**

$plan \leftarrow \text{PLAN-ROUTE}(current, \{[1, 1]\}, safe) + [Climb]$

*action*  $\leftarrow$  POP(*plan*)

TELL(*KB*, MAKE-ACTION-SENTENCE(*action*, *t*))

*t*  $\leftarrow$  *t* + 1

**return** *action*

Assign percept sentence to variable

Add 1<sup>st</sup> 3 clauses of sentence to KB

\* One at a time

# Make\_Percept\_Sentence

- Initialization:
  - Set up index offsets
    - B\_OFFSET = 0;
    - G\_OFFSET = 16;
    - P\_OFFSET = 32;
    - S\_OFFSET = 48;
    - P\_OFFSET = 64;
  - Set up index in cell
    - TABLE = [13,14,15,16; 9,10,11,12; 5,6,7,8; 1,2,3,4];

Logical variable index (e.g., stench):  $\text{index} = \text{S\_OFFSET} + \text{TABLE}(\text{row}, \text{col})$

E.g.: if at  $x = 2$  and  $y = 3$ , find row and column in table:

$\text{col} = x; \text{row} = 4 - y + 1 \rightarrow \text{col} = 2, \text{row} = 2$

$\text{index} = 48 + 10 = 58$

		Columns			
		1	2	3	4
Rows	1	61	62	63	64
	2	57	58	59	60
	3	53	54	55	56
	4	49	50	51	52
		1	2	3	4
		X			

**function** HYBRID-WUMPUS-AGENT(*percept*) **returns** an *action*

**inputs:** *percept*, a list, [*stench*,*breeze*,*glitter*,*bump*,*scream*]

**persistent:** *KB*, a knowledge base, initially the atemporal “wumpus physics”

*t*, a counter, initially 0, indicating time

*plan*, an action sequence, initially empty

TELL(*KB*, MAKE-PERCEPT-SENTENCE(*percept*, *t*))

TELL the *KB* the ~~temporal~~ sentences for time *t*

*safe*  $\leftarrow \{[x, y] : \text{ASK}(\text{KB}, \text{OK}_{x,y}^t) = \text{true}\}$

**if** ASK(*KB*, *Glitter*<sup>*t*</sup>) = *true* **then**

*plan*  $\leftarrow$  [*Grab*] + PLAN-ROUTE(*current*, {[1,1]}, *safe*) + [*Climb*]

**if** *plan* is empty **then**

*unvisited*  $\leftarrow \{[x, y] : \text{ASK}(\text{KB}, L_{x,y}^{t'}) = \text{false} \text{ for all } t' \leq t\}$

*plan*  $\leftarrow$  PLAN-ROUTE(*current*, *unvisited*  $\cap$  *safe*, *safe*)

**if** *plan* is empty and ASK(*KB*, *HaveArrow*<sup>*t*</sup>) = *true* **then**

*possible\_wumpus*  $\leftarrow \{[x, y] : \text{ASK}(\text{KB}, \neg W_{x,y}) = \text{false}\}$

*plan*  $\leftarrow$  PLAN-SHOT(*current*, *possible\_wumpus*, *safe*)

**if** *plan* is empty **then** // no choice but to take a risk

*not\_unsafe*  $\leftarrow \{[x, y] : \text{ASK}(\text{KB}, \neg \text{OK}_{x,y}^t) = \text{false}\}$

*plan*  $\leftarrow$  PLAN-ROUTE(*current*, *unvisited*  $\cap$  *not\_unsafe*, *safe*)

**if** *plan* is empty **then**

*plan*  $\leftarrow$  PLAN-ROUTE(*current*, {[1, 1]}, *safe*) + [*Climb*]

*action*  $\leftarrow$  POP(*plan*)

TELL(*KB*, MAKE-ACTION-SENTENCE(*action*, *t*))

*t*  $\leftarrow$  *t* + 1

**return** *action*





**function** HYBRID-WUMPUS-AGENT(*percept*) **returns** an *action*

**inputs:** *percept*, a list, [*stench*,*breeze*,*glitter*,*bump*,*scream*]

**persistent:** *KB*, a knowledge base, initially the atemporal “wumpus physics”

*t*, a counter, initially 0, indicating time

*plan*, an action sequence, initially empty

TELL(*KB*, MAKE-PERCEPT-SENTENCE(*percept*, *t*))

TELL the *KB* the temporal “physics” sentences for time *t*

$safe \leftarrow \{[x, y] : \text{ASK}(KB, OK_{x,y}^t) = true\}$

**if** ASK(*KB*, *Glitter*<sup>*t*</sup>) = *true* **then**

*plan*  $\leftarrow$  [*Grab*] + PLAN-ROUTE(*current*, {[1,1]}, *safe*) + [*Climb*]

**if** *plan* is empty **then**

*unvisited*  $\leftarrow$  {[*x*, *y*] : ASK(*KB*, *L*<sub>*x,y*</sub><sup>*t'*</sup>) = *false* for all *t'*  $\leq$  *t*}

*plan*  $\leftarrow$  PLAN-ROUTE(*current*, *unvisited*  $\cap$  *safe*, *safe*)

**if** *plan* is empty and ASK(*KB*, *HaveArrow*<sup>*t*</sup>) = *true* **then**

*possible\_wumpus*  $\leftarrow$  {[*x*, *y*] : ASK(*KB*,  $\neg W_{x,y}$ ) = *false*}

*plan*  $\leftarrow$  PLAN-SHOT(*current*, *possible\_wumpus*, *safe*)

**if** *plan* is empty **then** //no choice but to take a risk

*not\_unsafe*  $\leftarrow$  {[*x*, *y*] : ASK(*KB*,  $\neg OK_{x,y}^t$ ) = *false*}

*plan*  $\leftarrow$  PLAN-ROUTE(*current*, *unvisited*  $\cap$  *not\_unsafe*, *safe*)

**if** *plan* is empty **then**

*plan*  $\leftarrow$  PLAN-ROUTE(*current*, {[1, 1]}, *safe*) + [*Climb*]

*action*  $\leftarrow$  POP(*plan*)

TELL(*KB*, MAKE-ACTION-SENTENCE(*action*, *t*))

*t*  $\leftarrow$  *t* + 1

**return** *action*

**This will be several questions about:**

- \* neighbors of current location
- \* whose safety status is unknown
- \* need to ask both  $P_{xy}$  and  $\neg P_{xy}$
- \* need to ask both  $W_{xy}$  and  $\neg W_{xy}$

**Then TELL any proven fact:**

- \* to the KB

**function** HYBRID-WUMPUS-AGENT(*percept*) **returns** an *action*

**inputs:** *percept*, a list, [*stench*,*breeze*,*glitter*,*bump*,*scream*]

**persistent:** *KB*, a knowledge base, initially the atemporal “wumpus physics”

*t*, a counter, initially 0, indicating time

*plan*, an action sequence, initially empty

TELL(*KB*, MAKE-PERCEPT-SENTENCE(*percept*, *t*))

TELL the *KB* the temporal “physics” sentences for time *t*

$safe \leftarrow \{[x, y] : \text{ASK}(KB, OK_{x,y}^t) = true\}$

Decide whether in ad hoc or KB

**if** ASK(*KB*, *Glitter*<sup>*t*</sup>) = *true* **then**

*plan*  $\leftarrow$  [*Grab*] + PLAN-ROUTE(*current*, {[1,1]}, *safe*) + [*Climb*]

**if** *plan* is empty **then**

*unvisited*  $\leftarrow \{[x, y] : \text{ASK}(KB, L_{x,y}^{t'}) = false \text{ for all } t' \leq t\}$

*plan*  $\leftarrow$  PLAN-ROUTE(*current*, *unvisited*  $\cap$  *safe*, *safe*)

**if** *plan* is empty and ASK(*KB*, *HaveArrow*<sup>*t*</sup>) = *true* **then**

*possible\_wumpus*  $\leftarrow \{[x, y] : \text{ASK}(KB, \neg W_{x,y}) = false\}$

*plan*  $\leftarrow$  PLAN-SHOT(*current*, *possible\_wumpus*, *safe*)

**if** *plan* is empty **then** // no choice but to take a risk

*not\_unsafe*  $\leftarrow \{[x, y] : \text{ASK}(KB, \neg OK_{x,y}^t) = false\}$

*plan*  $\leftarrow$  PLAN-ROUTE(*current*, *unvisited*  $\cap$  *not\_unsafe*, *safe*)

**if** *plan* is empty **then**

*plan*  $\leftarrow$  PLAN-ROUTE(*current*, {[1, 1]}, *safe*) + [*Climb*]

*action*  $\leftarrow$  POP(*plan*)

TELL(*KB*, MAKE-ACTION-SENTENCE(*action*, *t*))

*t*  $\leftarrow$  *t* + 1

**return** *action*

**function** HYBRID-WUMPUS-AGENT(*percept*) **returns** an *action*

**inputs:** *percept*, a list, [*stench*,*breeze*,*glitter*,*bump*,*scream*]

**persistent:** *KB*, a knowledge base, initially the atemporal “wumpus physics”

*t*, a counter, initially 0, indicating time

*plan*, an action sequence, initially empty

TELL(*KB*, MAKE-PERCEPT-SENTENCE(*percept*, *t*))

TELL the *KB* the temporal “physics” sentences for time *t*

$safe \leftarrow \{[x, y] : \text{ASK}(KB, OK_{x,y}^t) = true\}$

**if** ASK(*KB*, *Glitter*<sup>*t*</sup>) = *true* **then**

$plan \leftarrow [Grab] + \text{PLAN-ROUTE}(current, \{[1,1]\}, safe) + [Climb]$

**if** *plan* is empty **then**

$unvisited \leftarrow \{[x, y] : \text{ASK}(KB, L_{x,y}^{t'}) = false \text{ for all } t' \leq t\}$

$plan \leftarrow \text{PLAN-ROUTE}(current, unvisited \cap safe, safe)$

**if** *plan* is empty and ASK(*KB*, *HaveArrow*<sup>*t*</sup>) = *true* **then**

$possible\_wumpus \leftarrow \{[x, y] : \text{ASK}(KB, \neg W_{x,y}) = false\}$

$plan \leftarrow \text{PLAN-SHOT}(current, possible\_wumpus, safe)$

**if** *plan* is empty **then** // no choice but to take a risk

$not\_unsafe \leftarrow \{[x, y] : \text{ASK}(KB, \neg OK_{x,y}^t) = false\}$

$plan \leftarrow \text{PLAN-ROUTE}(current, unvisited \cap not\_unsafe, safe)$

**if** *plan* is empty **then**

$plan \leftarrow \text{PLAN-ROUTE}(current, \{[1, 1]\}, safe) + [Climb]$

$action \leftarrow \text{POP}(plan)$

TELL(*KB*, MAKE-ACTION-SENTENCE(*action*, *t*))

$t \leftarrow t + 1$

**return** *action*

**This is in ad hoc knowledge**

**Questions?**