The Pass-Through Problem

Multi-Agent Epistemic Planning over Maps the Robots Build, and a Knowledge Precondition Enforced as a Fixed Point

Haniel Ulises

Epistemic Robotics

September 25, 2026

In the preceding demonstrations the occupancy grid and the Kripke model are two views of one fact, and the epistemic layer never asks where a piece of knowledge came from. This report builds a problem in which it must. A racking block crosses a thirty by fifty metre hall; exactly one of three pass-through bays in it is open. Two scouts read bays from their own SLAM maps, a map exchange is fused into the receiver's map and applied to the model as a semi-private announcement, and a carrier may cross only a bay it knows to be open. That precondition is enforced twice: by the executor against the model, and by the carrier's route, which is the least fixed point of a reachability formula whose safe set is itself epistemic.

On one branch of the policy the two scouts come to know the open bay only distributedly, and two exchanges make it the carrier's by elimination. The carrier then drives through a bay no robot surveyed and of which the exchanges carried no cell. All three branches were executed to the goal in simulation. Two further results are negative: a formulation of the exchanges as private announcements admits a policy that exploits a non-serial relation, and a published footprint of the AWS shelf model permutes the axes of its mesh.

Contents

1 Scope

The earlier warehouse reports execute epistemic policies over large floors, and each states that floor area and epistemic difficulty are independent. The multi-site report coupled them through the location of a proposition. This report couples them through the source of knowledge: what an agent knows about the floor is what its own map, or a map it has been sent, contains.

Three claims are advanced; Section 13 sets out what the next iteration takes up from them.

1. Sensing outcomes, communication and the knowledge precondition of an ontic action were each grounded in occupancy grids built by the robots during the run, and the three groundings agreed with the epistemic model at every step of every run.
2. On the third branch of the policy, knowledge held only distributedly by two agents was transferred to a third by two map exchanges, none of which carried a cell of the bay concerned, and the third agent crossed that bay.
3. The carrier's route is the least fixed point of a formula whose safe set is the extension of an epistemic formula, evaluated over the model the executor maintains; the route exists exactly when the carrier knows of an open bay.

Four bounds are stated at the outset. The floor plan of the building is given to every robot as common knowledge, and only the interiors of the three bays are left unknown in it. Localisation is the simulator's odometry, so the maps are records of observation and not an evaluation of SLAM. The radio is assumed available throughout. And the executor traverses the policy in order, so the robots act sequentially.

2 Preliminaries

The vocabulary is that of the companion reports and is restated only where this report depends on it.

Definition 1 (Epistemic model)**.** Let $\mathcal{A}$ be a finite set of agents and P a finite set of atoms. An *epistemic model* is $\mathcal{M} = (W, R, V, W_d)$ with W a finite set of worlds, $R : \mathcal{A} \to 2^{W\times W}$, $V : W \to 2^P$, and $\emptyset \neq W_d \subseteq W$ the designated worlds. A formula holds at $\mathcal{M}$ when it holds at every designated world, and $\mathcal{M}, w \models K_a\varphi$ iff φ holds at every v with $(w, v) \in R(a)$.

Definition 2 (Distributed knowledge)**.** For a group $G \subseteq \mathcal{A}$, $\mathcal{M}, w \models D_G\varphi$ iff φ holds at every v with $(w, v) \in \bigcap_{a\in G} R(a)$. The group knows distributedly what it would know if its members pooled what they know.

Event models, product update and the three action types of the survey domains are as defined in *Contamination with a Location*, Section 2. One consequence is used repeatedly below. In a *semi-private* action the agents outside the audience relate all events of the action to one another, so they learn that the action occurred and not which event did; in a *private* action they relate every event to a *nil* event with precondition $\top$, so they continue to consider possible that nothing occurred.

Proposition 1 (Vacuity)**.** If $(w, v) \notin R(a)$ for every v, then $\mathcal{M}, w \models K_a\varphi$ for every φ, including $\bot$.

Proposition 1 is immediate from the definition of satisfaction, and is the whole of the defect of Section 5. In S5 every relation is reflexive and the antecedent never holds; product update does not preserve seriality in general, and the antecedent can be reached.

3 The domain

Agents $\mathcal{A} = \{west, east, carrier\}$, objects $T = \{t_1, t_2, t_3\}$ of type `tunnel`. The initial model has three worlds, all designated, with $V(w_k) \ni open(t_k)$ only, and every relation total:

$$s \models C_{\mathcal{A}}\big(open(t_1) \oplus open(t_2) \oplus open(t_3)\big), \qquad s \models \neg Kw_i\, open(t_k) \quad \text{for every } i \in \mathcal{A},\ k \in \{1,2,3\}.$$

Three static facts are common knowledge,

$$covers(west, t_1), \qquad covers(east, t_3), \qquad hauls(carrier),$$

and no others of those predicates hold. The bay t_2 opens onto the carrier's own lane and is covered by no agent, so the only way any agent comes to know it is open is by excluding the other two.

Table 1: The actions of the domain, as event models.

Action	**Type**	**Designated events**	**Observability**
$survey(i,t)$	semi-private sensing	$open(t)$; $\neg open(t)$, each with $covers(i,t) \wedge \neg Kw_i\, open(t)$	i fully, others partially
$share\text{-}open(i,j,t)$	semi-private announcement	$K_i\, open(t)$	i, j fully, others partially
$share\text{-}shut(i,j,t)$	semi-private announcement	$K_i \neg open(t)$	i, j fully, others partially
$cross(i,t)$	public ontic	$hauls(i) \wedge K_i\, open(t)$, effect *delivered*	everyone fully

The goal is *delivered*. It is ontic, and its epistemic content is the precondition of *cross*: the goal can only be reached by an agent that knew. `plank` grounds the problem into 16 atoms and 72 ground actions.

4 The plan

Aletheia, through the ePlanSys plan solver under `strategy=aostar`, returns the policy of Figure 1 at depth five, with three leaves, after 445 expansions and 647 generations.

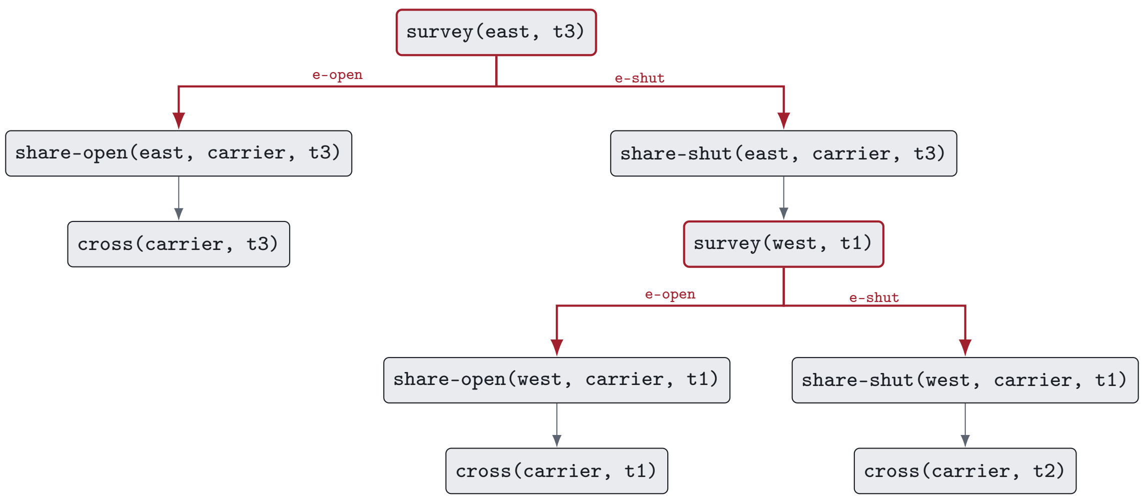

Figure 1: The policy returned: nine nodes, depth five, three leaves. Nodes outlined in red are sensing actions. The recorded run of Section 10 took the rightmost branch.

4.1 Distributed knowledge, made individual

Consider the third leaf. After $\mathit{survey}(\mathit{east}, t_3)$ returns $\neg \mathit{open}(t_3)$ and $\mathit{survey}(\mathit{west}, t_1)$ returns $\neg \mathit{open}(t_1)$, each scout has settled its own bay and not the other's, since sensing is semi-private. Then

$$s \models D_{\{\mathit{west},\mathit{east}\}}\, \mathit{open}(t_2) \;\wedge\; \neg K_{\mathit{west}}\, \mathit{open}(t_2) \;\wedge\; \neg K_{\mathit{east}}\, \mathit{open}(t_2).$$

The intersection of the two scouts' relations at the designated world contains one world, the one at which t_2 is open; each relation alone contains two. The two announcements

$$\mathit{share\text{-}shut}(\mathit{east}, \mathit{carrier}, t_3) \quad \text{and} \quad \mathit{share\text{-}shut}(\mathit{west}, \mathit{carrier}, t_1)$$

restrict the carrier's relation to that world, and $s \models K_{\mathit{carrier}}\, \mathit{open}(t_2)$. Neither announcement mentions t_2.

Table 2: The model after each action of the third leaf, computed by product update from the grounded task. Each of the last four columns is the set of bays open in some world the agent, or the pair, considers possible from the designated worlds: a singleton is knowledge of which bay is open, and a set without t_k is knowledge that t_k is shut. World and designated counts agree with those the epistemic state logged during the recorded run.

After	$\lvert W\rvert$	$\lvert W_d\rvert$	*west*	*east*	*carrier*	$D_{\{\mathit{west},\mathit{east}\}}$
`initial`	3	3	$\{t_1, t_2, t_3\}$	$\{t_1, t_2, t_3\}$	$\{t_1, t_2, t_3\}$	$\{t_1, t_2, t_3\}$
`survey_east_t3` $\to$ `e-shut`	3	2	$\{t_1, t_2, t_3\}$	$\{t_1, t_2\}$	$\{t_1, t_2, t_3\}$	$\{t_1, t_2\}$
`share-shut_east_carrier_t3`	3	2	$\{t_1, t_2, t_3\}$	$\{t_1, t_2\}$	$\{t_1, t_2\}$	$\{t_1, t_2\}$
`survey_west_t1` $\to$ `e-shut`	3	1	$\{t_2, t_3\}$	$\{t_1, t_2\}$	$\{t_1, t_2\}$	$\{t_2\}$
`share-shut_west_carrier_t1`	3	1	$\{t_2, t_3\}$	$\{t_1, t_2\}$	$\{t_2\}$	$\{t_2\}$
`cross_carrier_t2`	1	1	$\{t_2\}$	$\{t_2\}$	$\{t_2\}$	$\{t_2\}$

The size of the model never changes along this branch: three worlds throughout, with only the designated set and the relations narrowing, until the public crossing collapses it to one. The fourth row is the one the domain exists for: the pair's column is a singleton and neither member's is.

The planner could equally have routed the knowledge through a scout,

$$\mathit{share\text{-}shut}(\mathit{west}, \mathit{east}, t_1) \;\text{ followed by }\; \mathit{share\text{-}open}(\mathit{east}, \mathit{carrier}, t_2),$$

the second an announcement by a speaker who knows t_2 is open without having observed it. That plan has the same depth. The one returned sends both findings to the carrier directly.

4.2 The crossing is an announcement

cross is public, and its precondition contains $K_{\mathit{carrier}}\, \mathit{open}(t)$. Every agent observes the event and therefore learns that its precondition held at the actual world. After it the model contains one world, and $s \models C_{\mathcal{A}}\, \mathit{open}(t_2)$: the scouts learn which bay was open from seeing the carrier go through.

5 A private announcement makes the carrier know everything

The exchanges were first written as private announcements, with the third agent oblivious. The planner returned a policy of depth four with two leaves, after 212 expansions; on its second leaf the carrier crosses t_1 after being told only that t_3 is shut, and the plan validator accepted it.

The trace of the product updates locates the fault. Let $\mathit{share\text{-}shut}(\mathit{east}, \mathit{west}, t_3)$ be executed privately. The carrier is oblivious to it, and its relation at the designated worlds leads to the copies produced by the *nil* event, at which *west* does not know $\neg \mathit{open}(t_3)$. Let $\mathit{share\text{-}shut}(\mathit{west}, \mathit{carrier}, t_3)$ follow, with precondition $K_{\mathit{west}} \neg \mathit{open}(t_3)$. The carrier observes it fully, so its relation is restricted to worlds at which that precondition holds, and there are none among the worlds it considered possible. Its relation at the designated worlds is empty, and by Proposition 1 $K_{carrier}\,open(t_1)$ holds.

Table 3: The private variant along its second leaf, in the notation of Table 2. After the second exchange the carrier's relation at both designated worlds is empty, so the set of bays it considers possible is empty, and $K_{carrier}\varphi$ holds for every φ.

After	$\|W\|$	$\|W_d\|$	*west*	*east*	*carrier*	$D_{\{west,east\}}$
`initial`	3	3	$\{t_1, t_2, t_3\}$	$\{t_1, t_2, t_3\}$	$\{t_1, t_2, t_3\}$	$\{t_1, t_2, t_3\}$
`survey_east_t3` $\rightarrow$ `e-shut`	3	2	$\{t_1, t_2, t_3\}$	$\{t_1, t_2\}$	$\{t_1, t_2, t_3\}$	$\{t_1, t_2\}$
`share-shut_east_west_t3`	5	2	$\{t_1, t_2\}$	$\{t_1, t_2\}$	$\{t_1, t_2, t_3\}$	$\{t_1, t_2\}$
`share-shut_west_carrier_t3`	7	2	$\{t_1, t_2\}$	$\{t_1, t_2\}$	$\emptyset$	$\emptyset$
`cross_carrier_t1`	2	2	$\{t_1, t_2\}$	$\emptyset$	$\emptyset$	$\emptyset$

Informally: an agent that believed nothing had been said, and is then told something that could only have been learned from what was said, has no world left that agrees with what it has heard. A planner searching over the model's semantics will find that agent and use it. The repair is not a restriction on the planner. Letting the third agent observe that an exchange took place, without observing its content, makes the exchanges semi-private, which keeps every relation an equivalence and the model in S5. The policy of Figure 1 is what the planner returns for that domain.

6 Knowledge from maps

Each robot runs its own `slam_toolbox`, remapped from the absolute topic `/map` to its own namespace. Were the three mappers to publish one map, every observation would be public, and the semi-private sensing the domain declares would not describe the system executing it.

A knowledge node per robot keeps two occupancy grids on the geometry of the floor plan: the robot's own SLAM map, resampled, and its *knowledge map*, the own map fused with every map the robot has been sent.

Definition 3 (Reading of a region)**.** Let G be a grid and B a set of cells. With the thresholds of `epistemic_slam`, B is *blocked* in G if some cell of B is occupied, *clear* if every cell of B is observed free, and *undecided* otherwise.

The quantifier is that of `plansys2_epistemic_perception`. Occupied dominates unobserved, since no further observation removes an obstacle, and unobserved dominates free, since a region is only clear when all of it is. The region read for a bay is its *corridor*, the interior less the navigation's inflation of 0.35 m from each wall. A cell within the inflation band is used by no route, so whether it has been observed says nothing about whether the bay can be crossed; including it made a scout's verdict depend on cells a laser at the mouth sees only at grazing incidence (Section 11).

Sensing. $survey(i, t)$ drives scout i to a standoff of 1.5 m from the southern mouth of t, faces it into the bay, and reports the event its *own* map decides: $open(t)$ if the corridor is clear, $\neg open(t)$ if blocked. A verdict counts when three consecutive map updates agree. If the corridor stays undecided, the scout moves in along the bay's axis over floor its own map shows free, then backs out and parks two metres along the block, off the carrier's approach.

Communication. $share(i, j, t)$ fuses i's knowledge map into j's through `epistemic_slam::fuse`. The reply counts the cells j newly holds over the grid and within each bay. The performer then checks j's knowledge map against the announcement: after *share-shut* the corridor must be blocked, after *share-open* clear. A third outcome is admitted and logged: the corridor undecided in j's map, which is the case exactly when i knew by elimination.

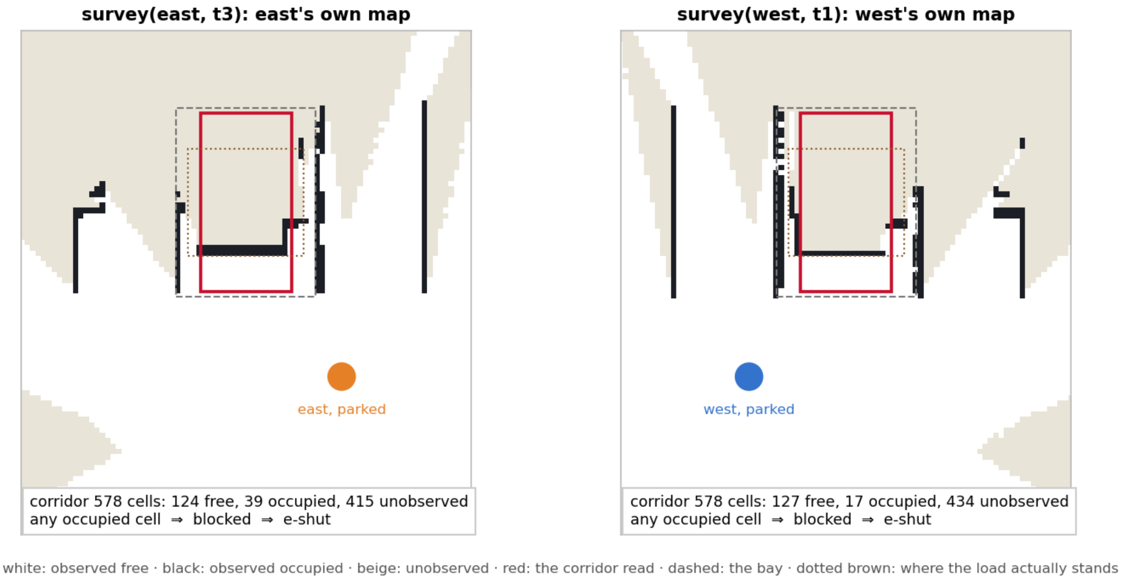

Figure 2: Each scout's own map at the moment its survey was reported, around the bay it read, from a second run of the `open:=t2` world recorded at every change of the model. The red box is the corridor. From the standoff most of it is shadowed by the load; one occupied cell decides *blocked*, and the load's south face supplies dozens.

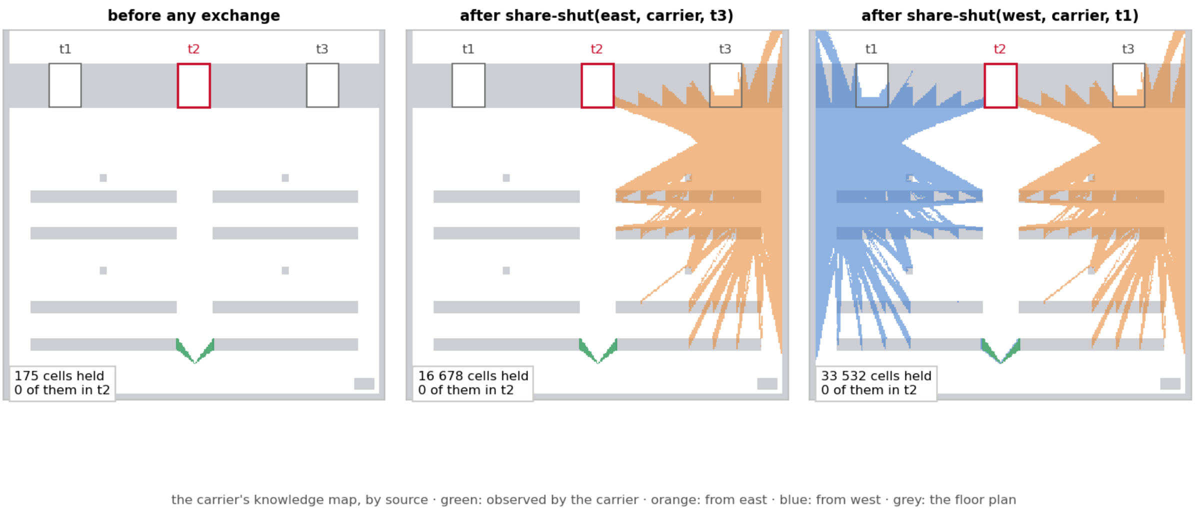

Figure 3: The carrier's knowledge map before any exchange and after each, coloured by the robot that observed each cell: 175 cells, then 16 678, then 33 532. The corridor of t_2 is empty in all three.

Soundness of the correspondence. A fusion carries every cell the sender holds, while an announcement names one bay. The model therefore under-approximates what an exchange conveys. For preconditions this is the safe direction: the model never credits an agent with knowledge its map does not support.

7 The route is the precondition

$cross(carrier, t)$ drives the carrier along the winning region of

$$W \;=\; \mu Z.\, \big(dock \vee (Safe \wedge \Diamond Z)\big), \qquad Safe \;=\; \Big[\!\!\Big[\, free \;\vee\; \bigvee_{t \in T} \big(t \wedge K_{carrier}\, open(t)\big) \Big]\!\!\Big],$$

where $\Diamond$ is one step of four-connectivity, t is true at the cells of bay t in every world, and *free* reads the floor plan outside the bays and the carrier's knowledge map inside them. The extension is taken at every pair of a cell and a world of the model the epistemic state publishes, by the formula evaluator of `mu_path_planner`, and restricted to the designated worlds. Occupied cells are grown by the inflation before the fixed point is taken.

Proposition 2. Let the floor plan be such that no path from the carrier to the dock avoids every bay, and let no bay be observed occupied in the carrier's knowledge map. Then the carrier lies in W if and only if, for some bay t traversable in the plan, either every cell of its corridor is observed free in the carrier's knowledge map, or $s \models K_{carrier}\, open(t)$.

The first hypothesis is checked on the floor itself before any file is written (Section 9). The proposition is the geometric form of the precondition: a bay the carrier knows open without having seen is safe by the second disjunct, a bay it has seen free by the first, and a bay it knows nothing of by neither. The knowledge precondition of *cross* is therefore enforced twice and independently: by the executor, which checks $K_{carrier}\, open(t)$ against the model before dispatching, and by the route, which does not exist otherwise.

Evaluation. `mu_path_planner::mu_reach` computes W by Kleene iteration, recomputing the backward image of all of Z_k at every step; on the 310×510 grid of this floor that is tens of seconds per query. The package evaluates the same fixed point semi-naively: since the iteration is monotone, only the cells that entered at step k can admit new cells at step $k+1$, and propagating from those visits each cell once. The layer at which a cell enters is the step of the Kleene iteration at which it would have entered, so the region and the iteration count are the same quantities; a unit test checks both against `mu_reach` on sixty random floors.

Measured. In the recorded run, before the last exchange W has 51 428 cells after 327 iterations and stops at the racking; after it, 98 818 cells after 582 iterations, reaching the carrier through t_2, and the formula lifted t_2 and no other bay.

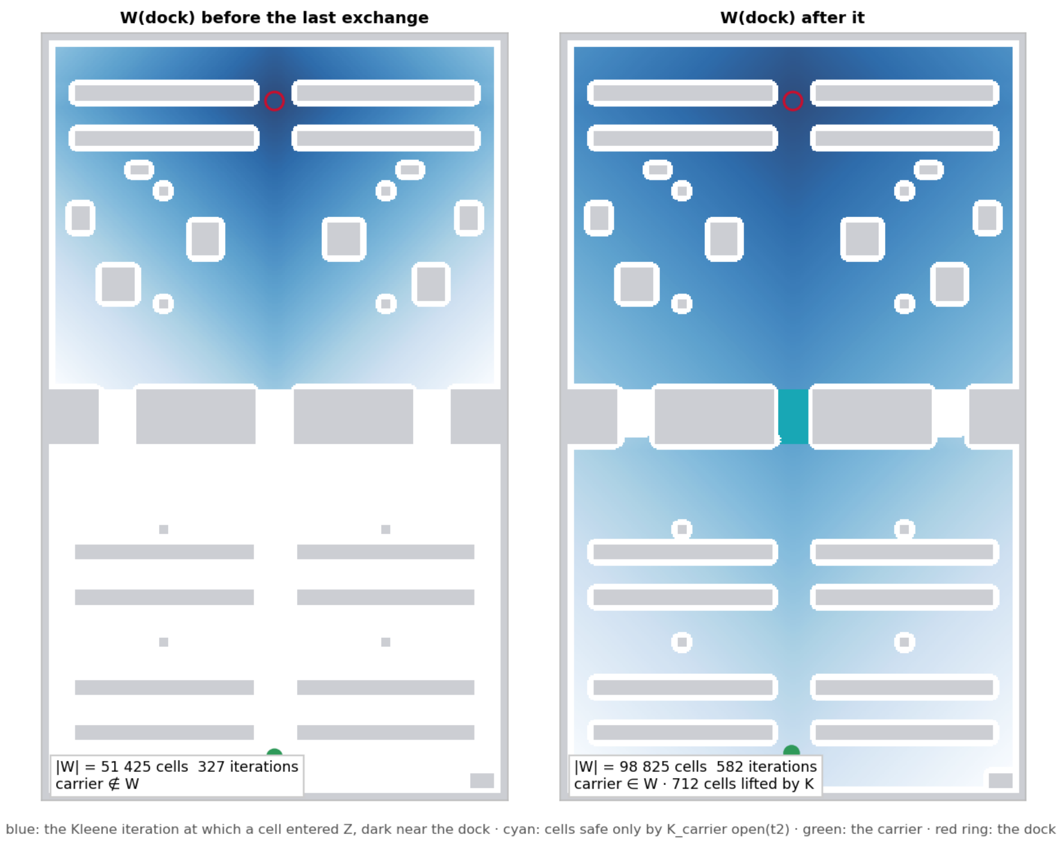

Figure 4: The winning region before and after the last exchange, coloured by the Kleene iteration at which each cell entered it. Cyan: cells of the bay the formula lifts. The iteration counts recomputed here from the recorded regions, 327 and 582, equal those the carrier logged.

8 The carrier's own map

The carrier maps as it drives, and its own map enters its knowledge map like any other. Along the lane it is correct at the south end and then slides west: the rack ends at $x = \pm 1.5\,\mathrm{m}$ are drawn in place at $y = -18$, $0.3\,\mathrm{m}$ west at $y = -12$, $0.45\,\mathrm{m}$ at $y = -9$, and by the bay about a metre. A $3.5\,\mathrm{m}$ scanner in a three-metre lane between repeating racks gives the scan matcher of `slam_toolbox` little to hold across the lane, and its correction accumulates there.

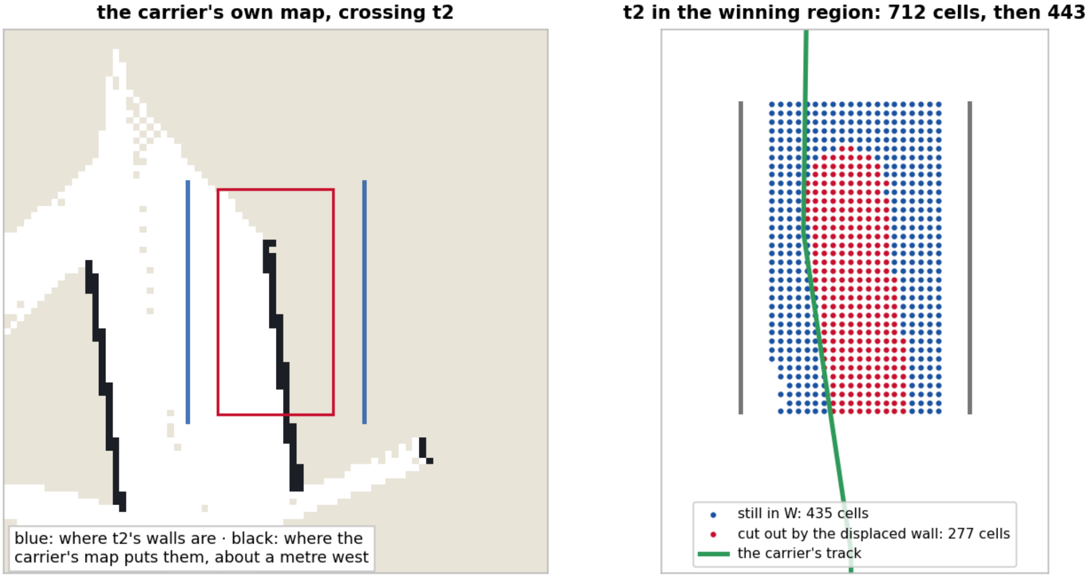

Figure 5: Left: the carrier's own map while it crosses t_2; blue marks where the bay's walls are, black where the map puts them. Right: the cells of t_2 in the winning region before the carrier sees the bay (712) and while it crosses (435 remain); the displaced wall and its inflation cut out 277, and the carrier's track runs west of them.

Inside t_2 the displaced east wall runs down the middle of the bay, and with the inflation around it removes a strip from the safe set. The carrier drove the western strip, at $x \approx -0.5\,\text{m}$, and reached the dock. The epistemic reading is exact. The carrier knew t_2 was open by elimination, from two maps that were right; its own observation of the bay, when it arrived, was wrong, and the knowledge map merged it without question, since *free* is read from that map inside the bays. The route survived because the bay is wider than the robot, and not because anything reconciled what the carrier had been told with what it saw.

9 The floor

The hall is the $30 \times 50\,\text{m}$ `OpenRobotics/Warehouse` model; the racking and the loads are AWS RoboMaker models. The world, the floor plan, the regions, the robots' poses and the camera shots are written from one layout file. The block is four rows of shelving from wall to wall with three bays 2.6 m wide on $x \in \{-10.44, 0, 10.44\}$; the storage and dispatch floors carry rows either side of a 3 m lane.

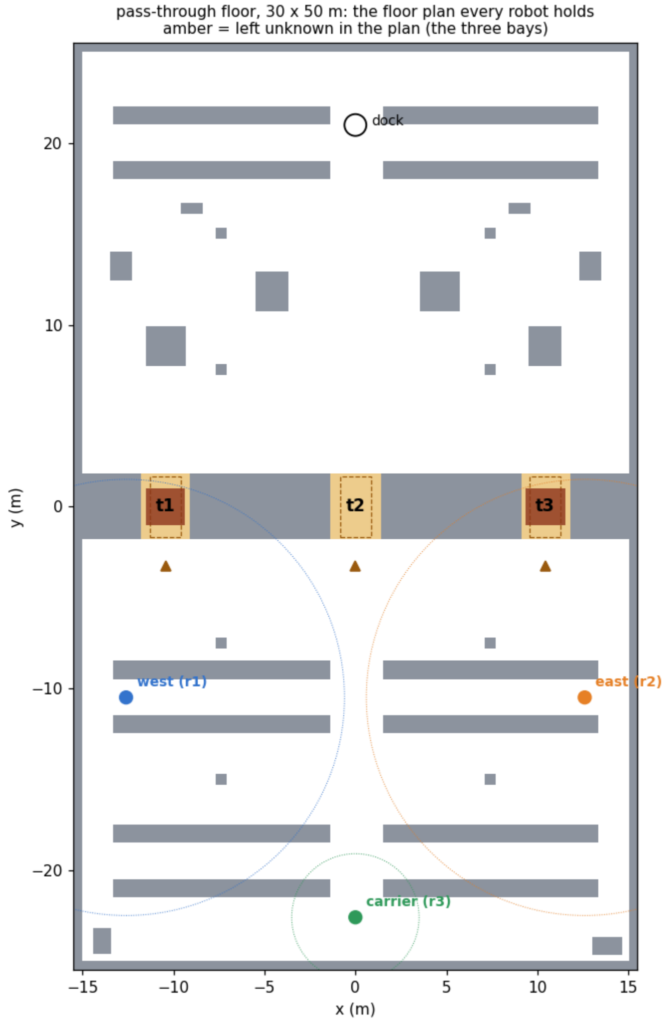

Figure 6: The floor plan every robot holds. Amber is what it leaves unknown, the three bays; the loads shown are those of the `open:=t2` world and are in no robot's plan.

Before the plan is written the floor is checked for the properties the mission depends on: that the carrier cannot reach the dock while no bay is known, that it can through each bay once that bay is, that a load closes its bay to a robot with the inflation applied, and that each scout reaches the mouth of its bay and its parking place. Each placed model's collision mesh is then sliced at laser height, with its node transform and unit applied, turned and placed as the world places it, and required to lie inside the box the plan draws for it to within one cell. The 73 placed models agree to 0.015 m.

10 Runs

10.1 The recorded run

The `open:=t2` world was recorded on a 3840×1080 Xvfb display with Gazebo on the left half and RViz on the right, and composed into a film of 162 s whose captions are lines of the run's own log, placed at the instant they were written. Table 4 gives the sequence.

Table 4: The recorded run, as the run reported it.

Action	**Reported**	**Model after**
plan	9 nodes, 3 leaves, 0.2 s; carrier $\notin W$	3 worlds, 3 designated
$survey(east, t_3)$	own map: 168 of 578 corridor cells observed, 37 occupied; $\neg open$	3 worlds, 2 designated
$share\text{-}shut(east, carrier, t_3)$	16 749 cells newly known to the carrier, 163 in t_3	3 worlds, 2 designated
$survey(west, t_1)$	own map: 144 of 578 observed, 17 occupied; $\neg open$	3 worlds, 1 designated
$share\text{-}shut(west, carrier, t_1)$	16 707 cells newly known, 144 in t_1, 0 in t_2	3 worlds, 1 designated
$cross(carrier, t_2)$	t_2 lifted by $K_{carrier}$; 43.1 m; dock after 90 s	1 world

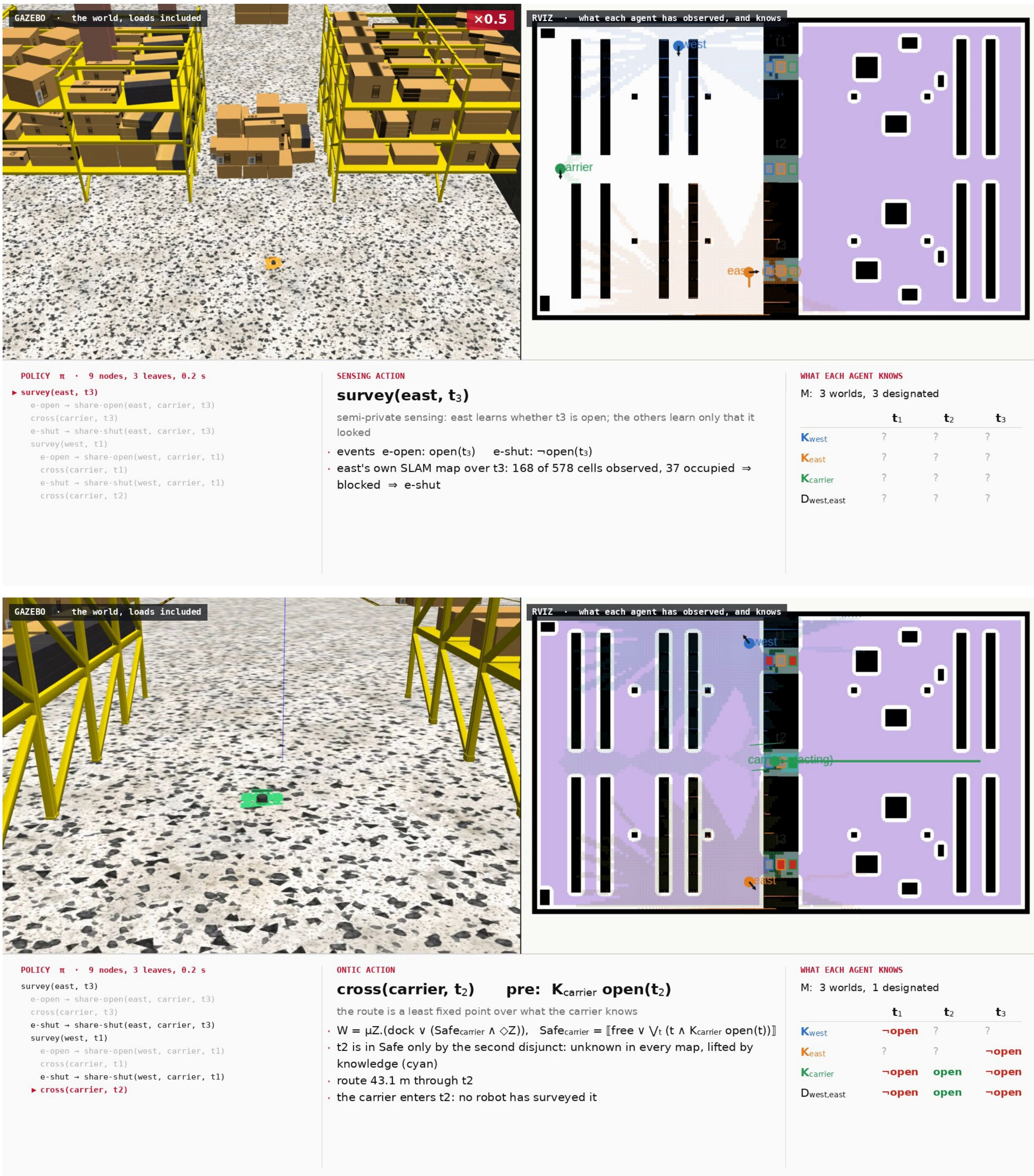

Figure 7: Two frames of the film. Above: *east* at the mouth of t_3, with the load in the bay and the reading of its own map in the panel. Below: the carrier inside t_2; the knowledge table shows $K_{carrier}\,open(t_2)$ beside $D_{\{west,east\}}\,open(t_2)$, while *west* and *east* still do not know it individually.

10.2 Three worlds, three leaves

The worlds differ by which two bays hold a load and by nothing else. Each was run headless, and each reached the goal on its own leaf.

Table 5: One policy in three worlds.

Open	Scouts read	Route	Safe by	To dock
t_3	*east*: t_3 clear, 578/578	60.6 m via t_3	observed free	112 s
t_1	*east*: t_3 blocked; *west*: t_1 clear, 578/578	60.7 m via t_1	observed free	115 s
t_2	*east*: t_3 blocked; *west*: t_1 blocked	43.1 m via t_2	$K_{carrier}\, open(t_2)$	83 s

From the standoff the scouts observed 471 and 453 of an open bay's 578 corridor cells, and in both cases moved in until every cell had been observed. In the first two worlds the bay crossed is in the carrier's knowledge map, observed free, and the formula lifts nothing. The third is the only world in which a robot drives through a bay that no map contains.

11 Defects

The shelves stood a quarter turn from the plan. The footprint of the AWS ShelfD model was first taken from the table used by the larger warehouse demonstrations, which gives it as $0.88 \times 2.613 \times 3.918$ m. Its collision mesh, read with its node transform and its centimetre unit, is $3.918 \times 0.88 \times 2.64$ m: the table permutes the axes. Every unit of the first floor therefore stood a quarter turn from where its plan put it, the block was a row of north-south columns with 1.7 m gaps, and the lasers saw between them. It was found in the t_1 world, where the carrier stopped 0.35 m short of the south end of a column the plan did not contain. The mesh check of Section 9 rejects that layout and names the same extent. The table has not been corrected in the package that ships it.

Every cell of a wider region. With a corridor margin of 0.30 m the scouts at the mouths of open bays observed 678 and 608 of 714 cells and never decided. Moving in resolved all but one, in a far corner inside the inflation band; the corridor of Section 6 excludes that band by construction.

A scout pinned at an aisle end. On the first floor with the corrected shelves the aisle ends left 1.2 m to the wall. A scout leaving its aisle there cut the corner of the row, its laser held it 0.3 m from the rack, and it did not move again. The lane was narrowed to widen those gaps to 1.7 m, shortcuts are taken against the safe set less one cell, and a robot held by its laser for two seconds backs off and plans again.

Unknown written as free. The floor plan writes unknown as grey 205, an occupancy of 0.196. Against a free threshold of 0.25 it reads back as free, and the three bays arrive open in every robot's plan. The threshold is 0.196, as map savers write it.

A carrier map that read an open bay as blocked. In the t_1 world and in the recorded run the carrier's own map read the open bay it was crossing as blocked. Section 8 traces it to the drift of that map along the lane, measured against the rack ends.

A float32 resolution. A grid's resolution crosses a message as a float32, and a geometry comparison at 10^{-9} refused every exchange as not being on the receiver's grid.

The camera stayed on a shot. Driven at 25 poses per second, the Gazebo view held two fixed shots and then stayed on the second for the rest of the run while chase poses were being published. The same switch worked in isolation. The client renders four or five frames per second under the software renderer, and a queue of poses is the likeliest reading; it was not established. At 5 Hz every cut landed.

12 What the runs establish

That a multi-agent epistemic policy can be executed in which every sensing outcome is decided from the performing agent's own map, every announcement is carried by a map fusion whose effect on the receiver's map is checked against the announcement, and the knowledge precondition of the ontic action is enforced by the reachability of its route as well as by the executor.

That on the branch where it is needed, knowledge held only distributedly by two agents became individual knowledge of a third through exchanges that carried none of the cells it concerns, and that the third agent acted on it.

And that one policy reached its goal in each of the three worlds its uncertainty ranges over.

13 Directions for the next iteration

Reconciling what an agent sees with what it was told. The knowledge map merges every source with one rule, the more confident reading. When the carrier's own drifted observation of t_2 contradicted the bay it knew to be open, nothing registered the contradiction. A knowledge map that kept its sources apart, as the epistemic state keeps its worlds, would let the carrier detect that its own map disagrees with its knowledge, and treat that as evidence about its localisation.

Localisation from the maps. Poses here are the simulator's odometry, so the maps are records of observation. With localisation taken from the same SLAM maps the drift of Section 8 would move the robot as well as the walls, and whether a bay is where the model says it is becomes an epistemic question in its own right.

Exchanges that carry what the map carries. A fusion carries every cell the sender holds, while the announcement names one bay. An announcement generated from the fused cells, one per region the fusion decides, would close that gap and remove the under-approximation of Section 6.

Communication that can fail. The link is assumed available and the exchange instantaneous. The link-outage apparatus of *A Link That Actually Falls* gates exactly the topics an exchange uses, and composing the two would make the timing of an exchange a planning decision.

A harder epistemic problem. Three worlds and 445 expansions. More bays, more carriers, or a goal that constrains who else may come to know would make the search itself the object of measurement.

Reproduction

```
ros2 launch pass_through_demo pass_through_launch.py # open:=t2
ros2 launch pass_through_demo pass_through_launch.py open:=t3
bash tools/validate.sh # no simulator
bash tools/record_demo.sh t2 /tmp/raw.mkv /tmp/run.log # Xvfb recording
python3 tools/make_video.py --raw /tmp/raw.mkv --log /tmp/run.log \
       --policy /tmp/raw_policy.json --out /tmp/pass_through.mp4
python3 tools/make_floorplan.py --out maps/floorplan \
       --models <aws models>/ # floor and mesh checks
```