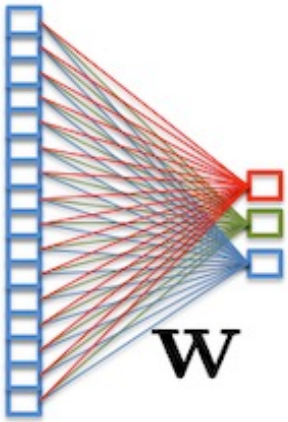
5IF - Deep Learning et Programmation Différentielle

5.3 Robot Navigation



Christian Wolf

Robotics

Perception

Navigation

HCI

Compagnon robots

Visual navigation and spatial reasoning

Office space

Homes

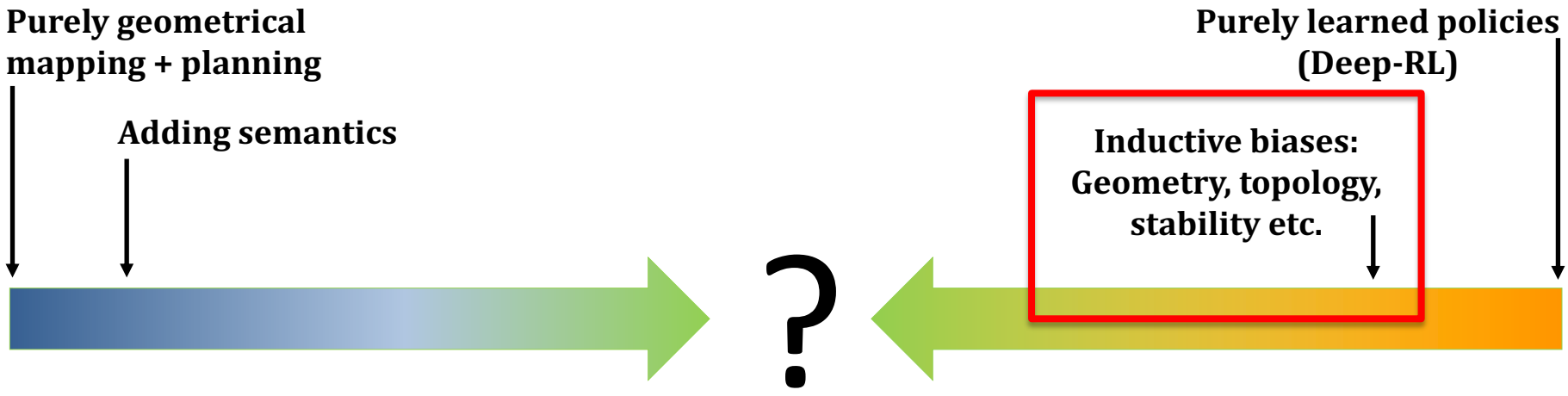
Hospitals

Edward Beeching

Jilles Dibangoye

Olivier Simonin

Christian Wolf

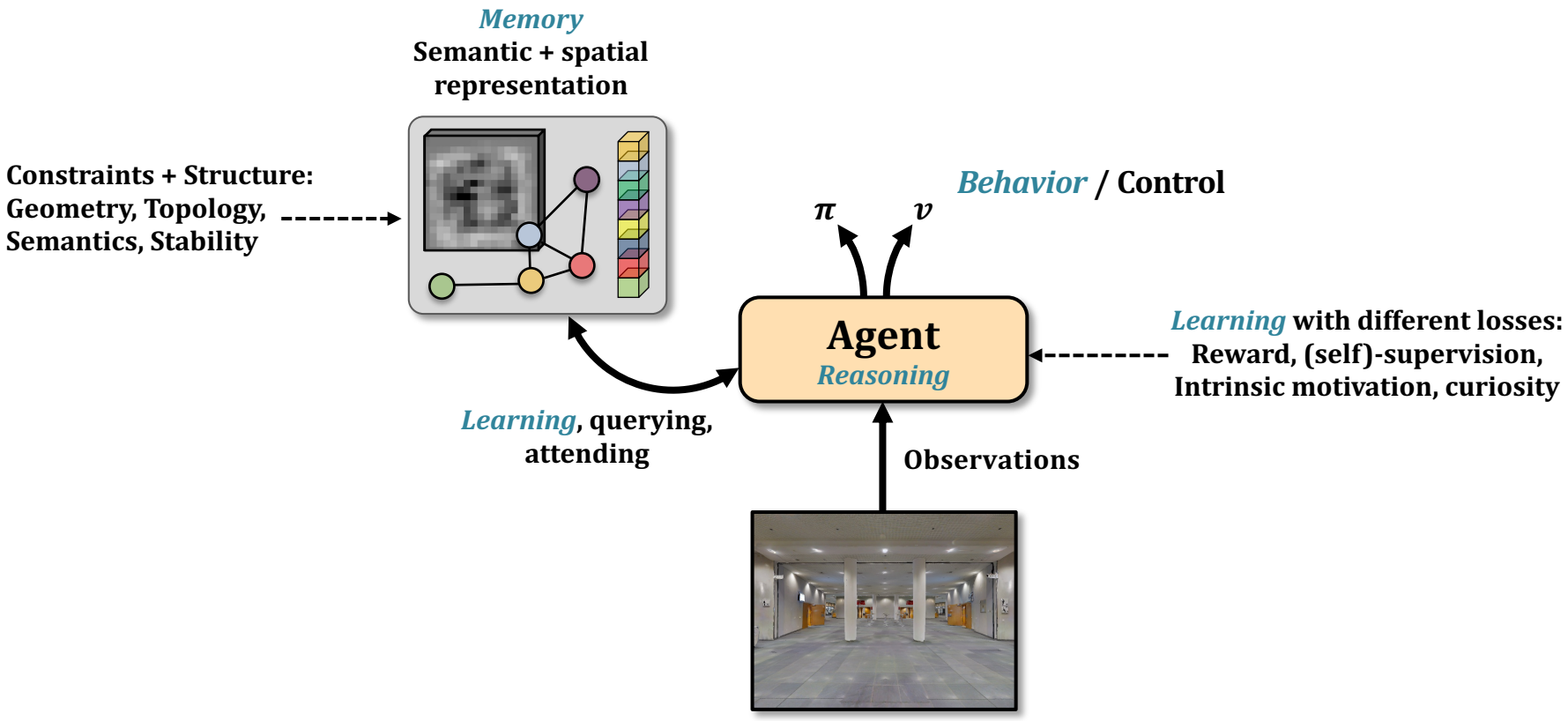


+ Real world solutions
- Simple tasks and reasoning (eg. waypoint navigation)

+ High-level reasoning
+ Discover tasks from reward
- Does not transfer to the real world

Question:

What kind of inductive bias can we create for learning navigation?

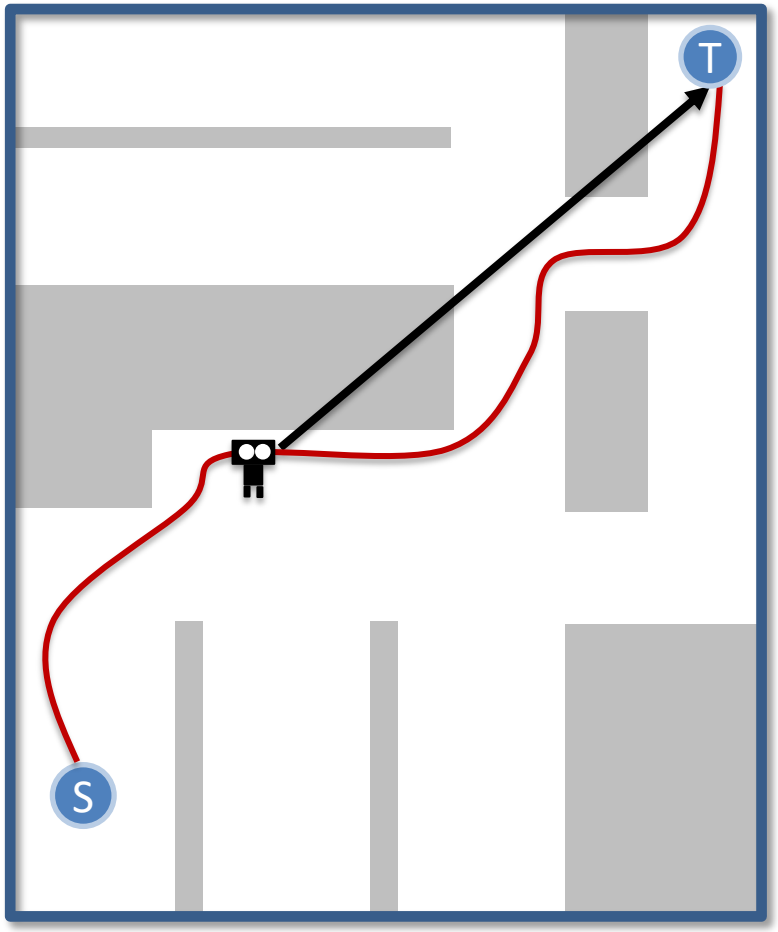

Memory
Semantic + spatial representation
Constraints + Structure:
Geometry, Topology,
Semantics, Stability
Behavior / Control
π
v
Agent
Reasoning
Learning with different losses:
Reward, (self)-supervision,
Intrinsic motivation, curiosity
Learning, querying, attending
Observations

Find my keys!

Can we learn a spatial representation which is richer than navigational space?

Can we learn it from reward alone?

Task: PointGoal (+GPS)



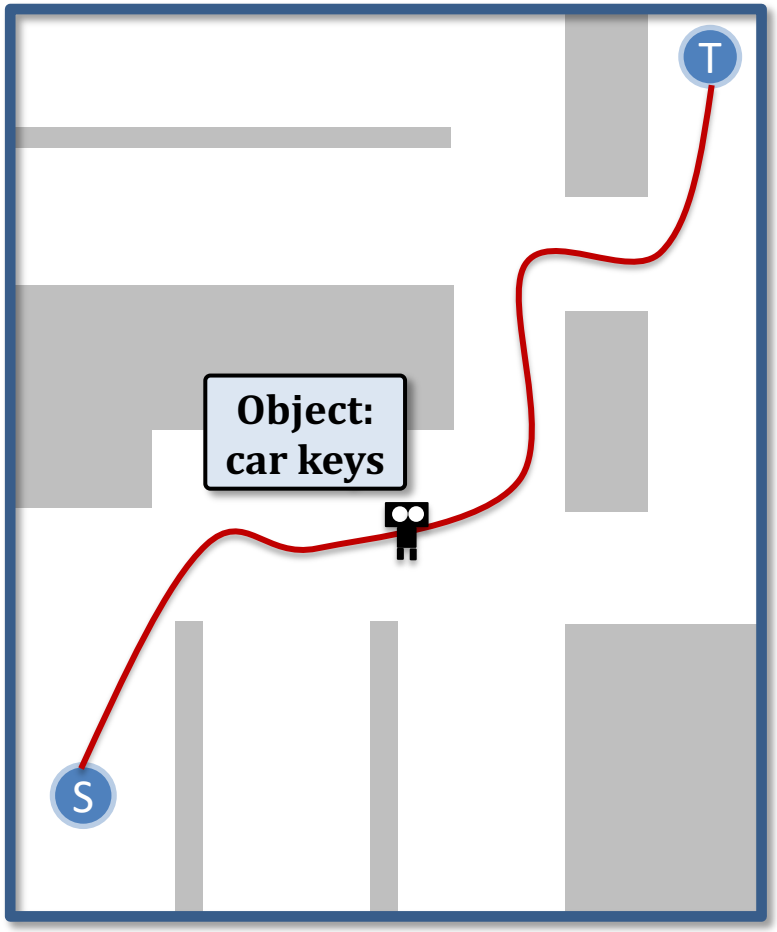
Task:

Find a target given coordinates, receive a direction vector et each step

Required reasoning:

- Recognize free navigation space
- Follow the direction vector
- Learn how to overcome obstacles (difference between Euclidean and geodesic path)

Task: ObjectNav



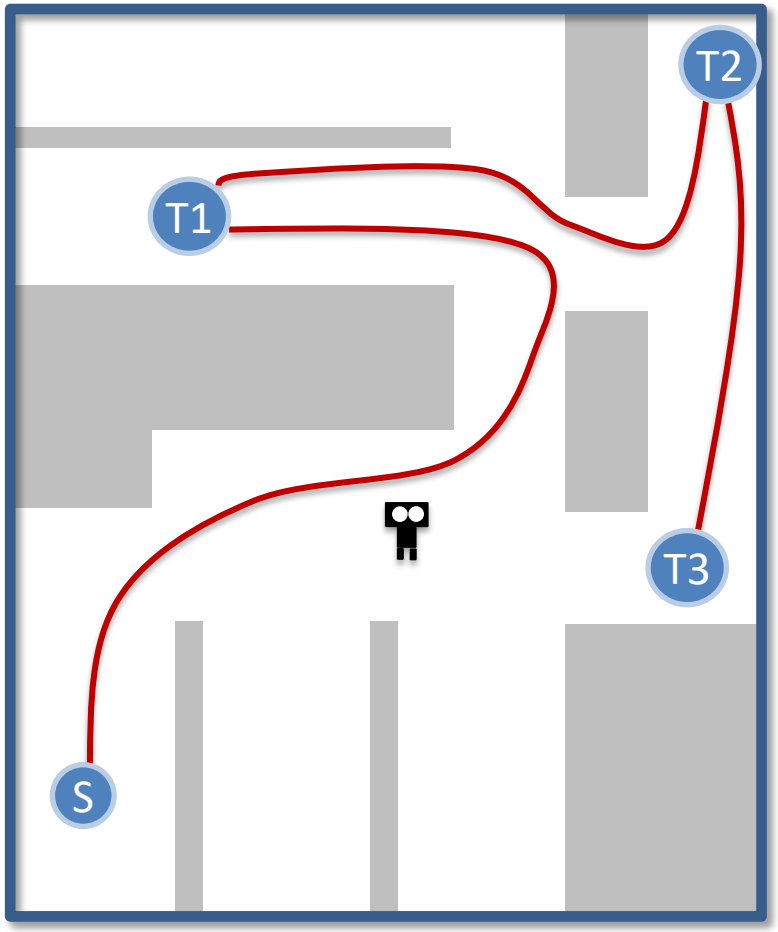
Task:

Find a target object given its object class.

Required reasoning:

- Recognize free navigation space
- *Explore the environment*
- *Recognize the object when seen and move towards it.*

Task: K-item scenario



Task:

Navigate to a list of objects sequentially in the right order.

Required reasoning:

- Recognize free navigation space
- Explore the environment
- *Map an object if I need to find it later (!)*
- Recognize the object when seen and move towards it.

[Beeching, Dibangoye, Simonin, Wolf, ECML-PKDD 2020]

[Beeching, Dibangoye, Simonin, Wolf, ICPR 2020]

What do we want the model to learn?

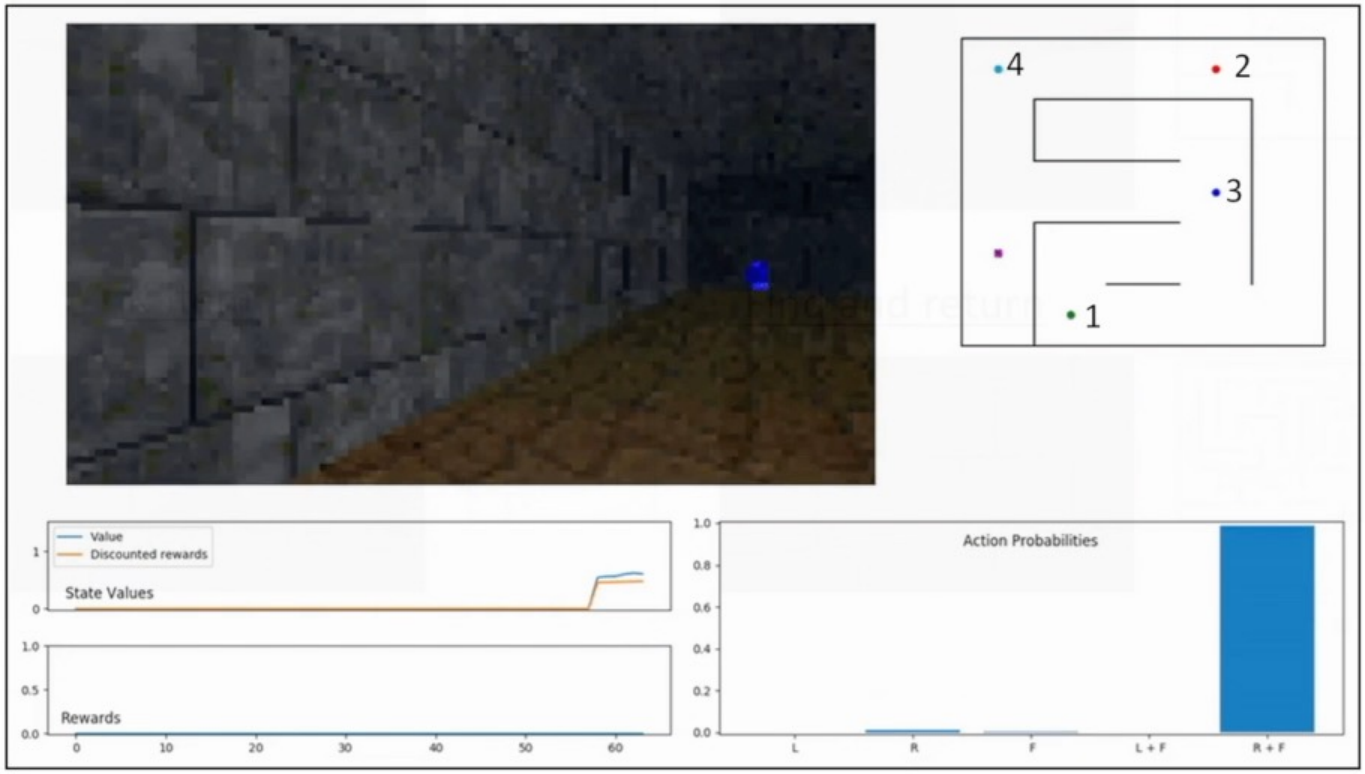
Generalization to new environments requires:

	Agent memory	Network parameters
Information on the (seen) environment, e.g. position of a couch	✓	✗
Positions of objects placed in the environment	✓	✗
Regularities of the environment (eg. bath tubs are in bathrooms; toilets are accessible from an aisle, not the living room)	✗	✓
Object affordances	✓	✓

The task formulation decides how learned information is stored!

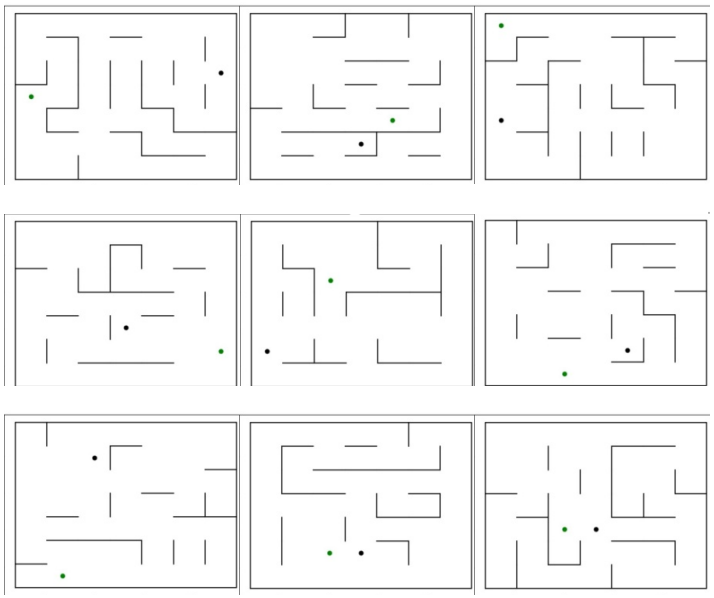
Scenarios: Ordered K-item

- Collect K items in a specific order
- Tests: Vision & memory

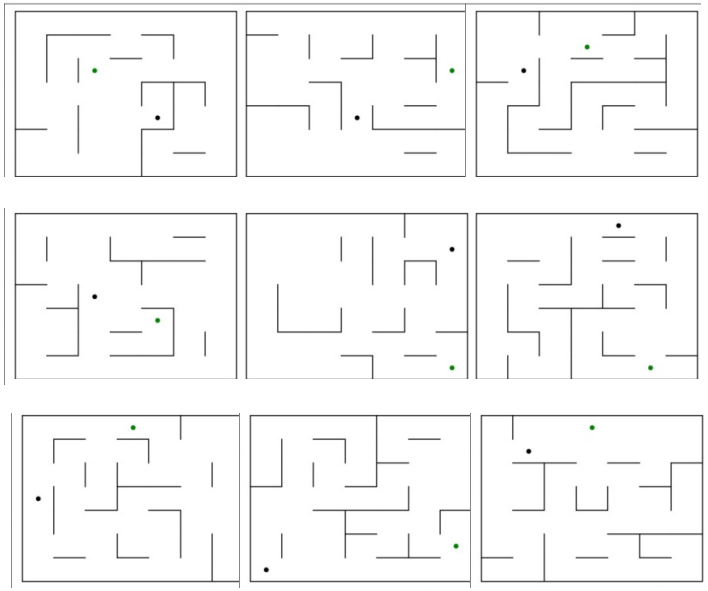


Generalization to unseen mazes

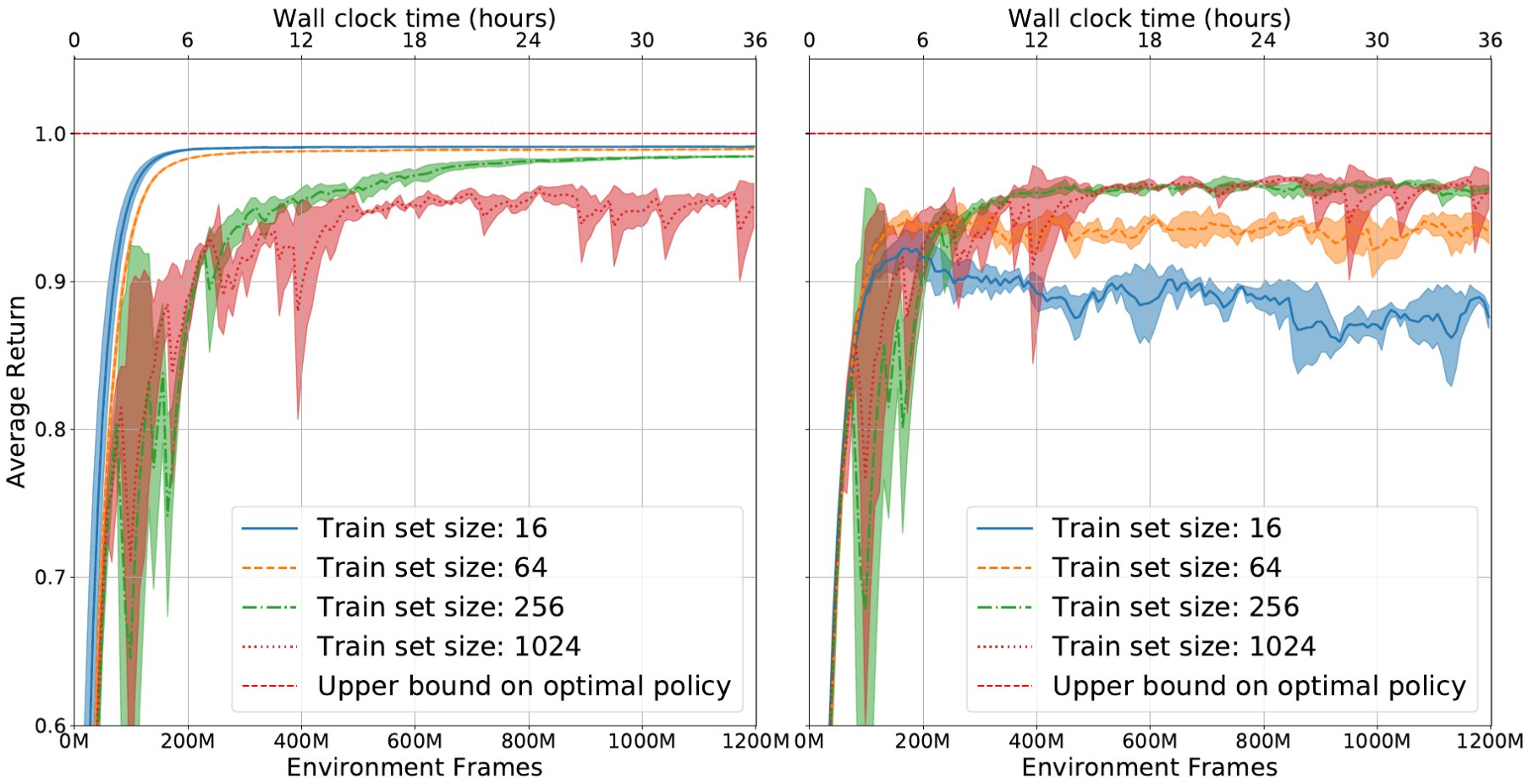
N - Training configurations



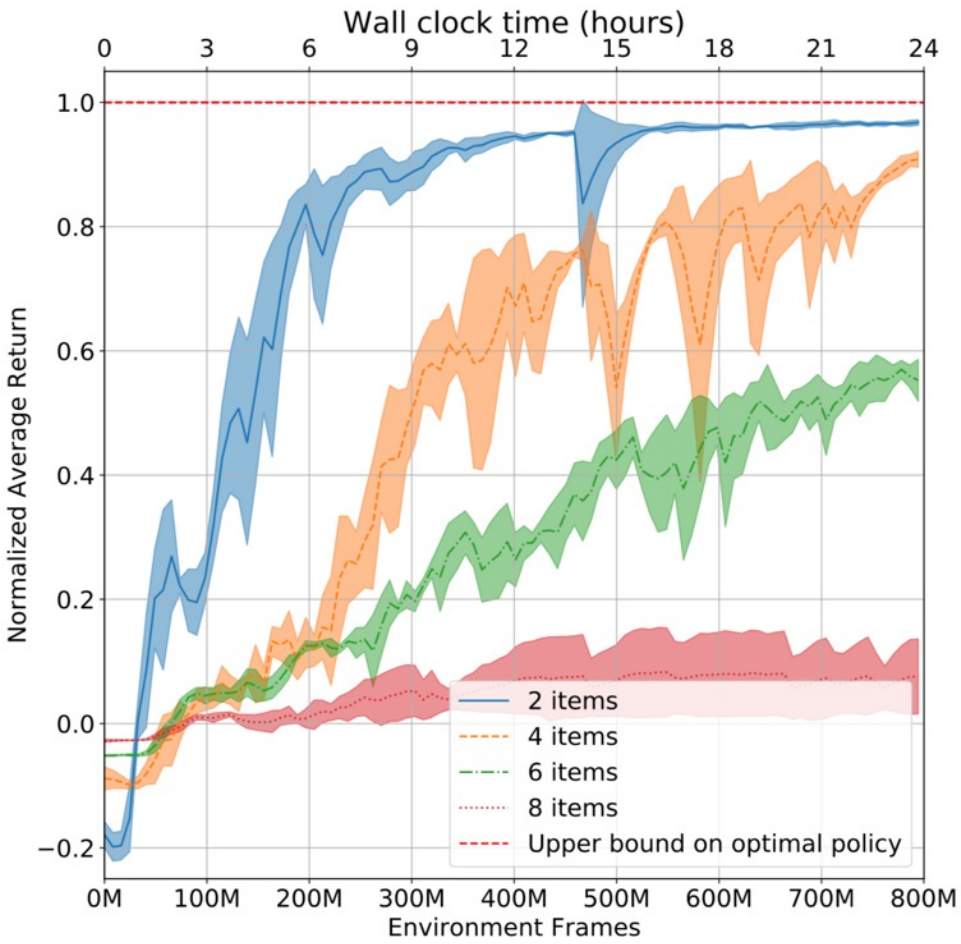
K - Testing configurations



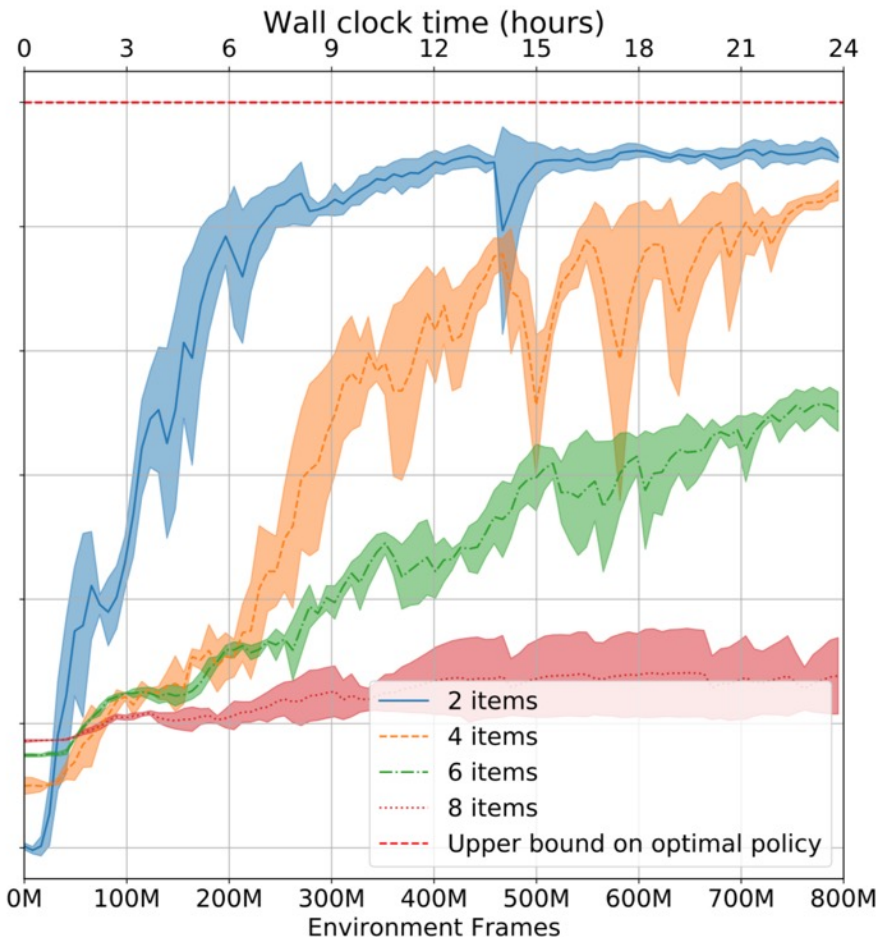
Generalization to unseen mazes



K-item scenario: recurrent agent (A3C)

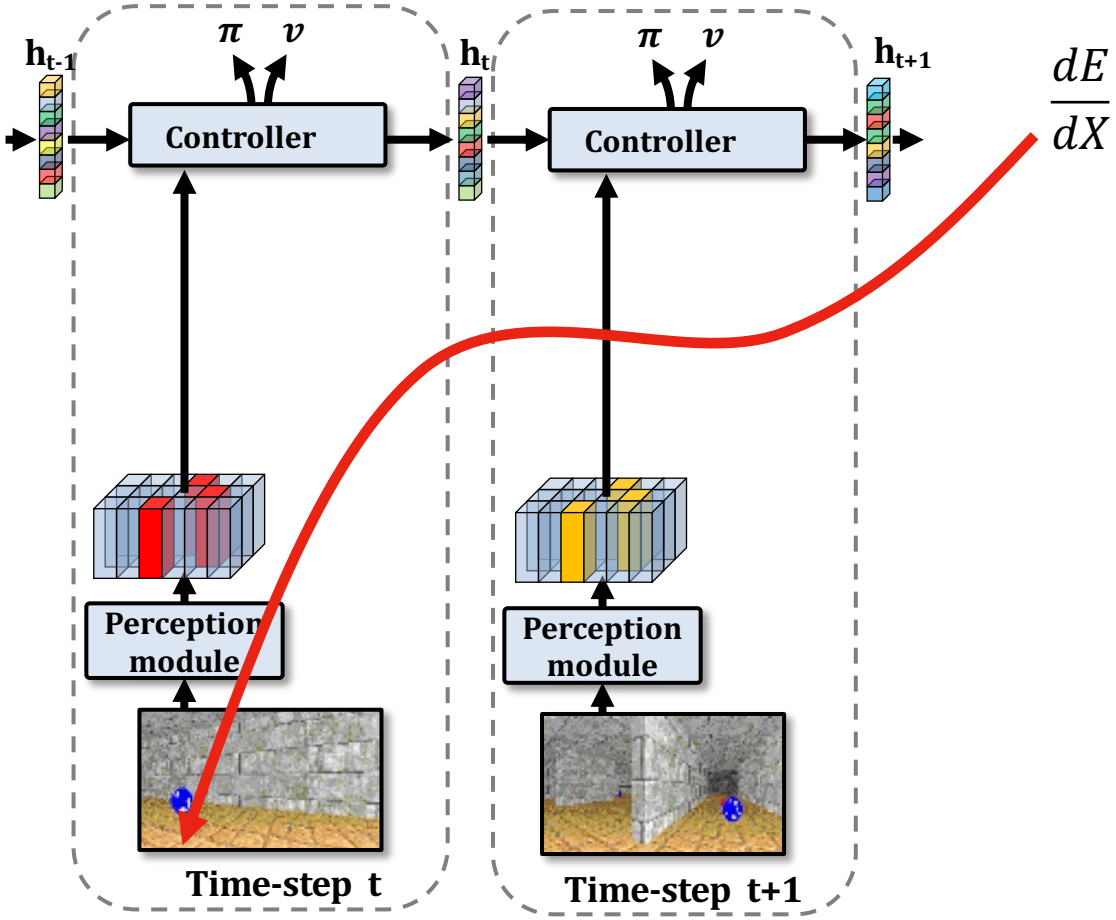


Training scenarios

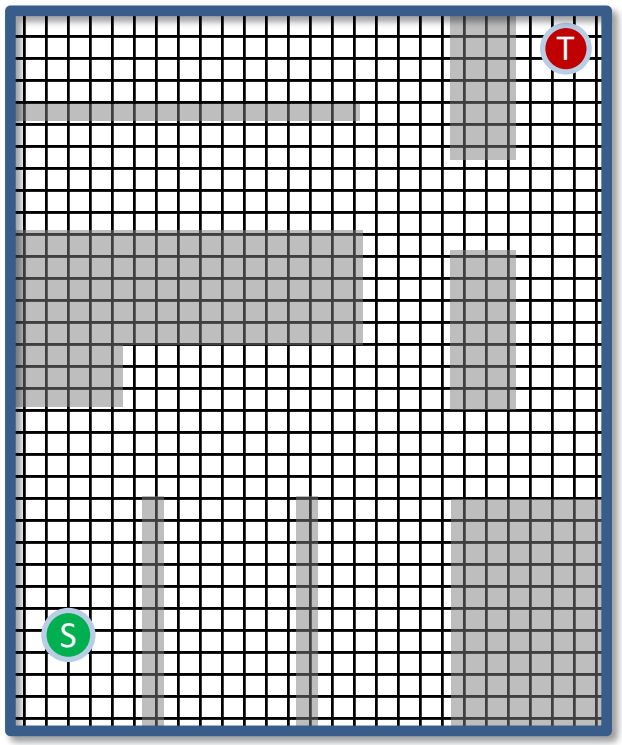


Testing scenarios

The unstructured recurrent baseline

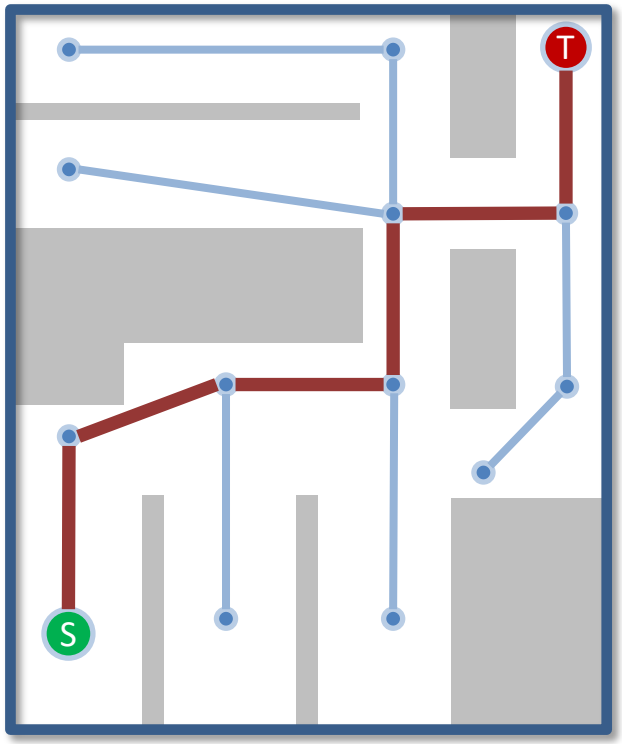


Spatial maps in robotics



Metric map
(=2D or 3D Grid)

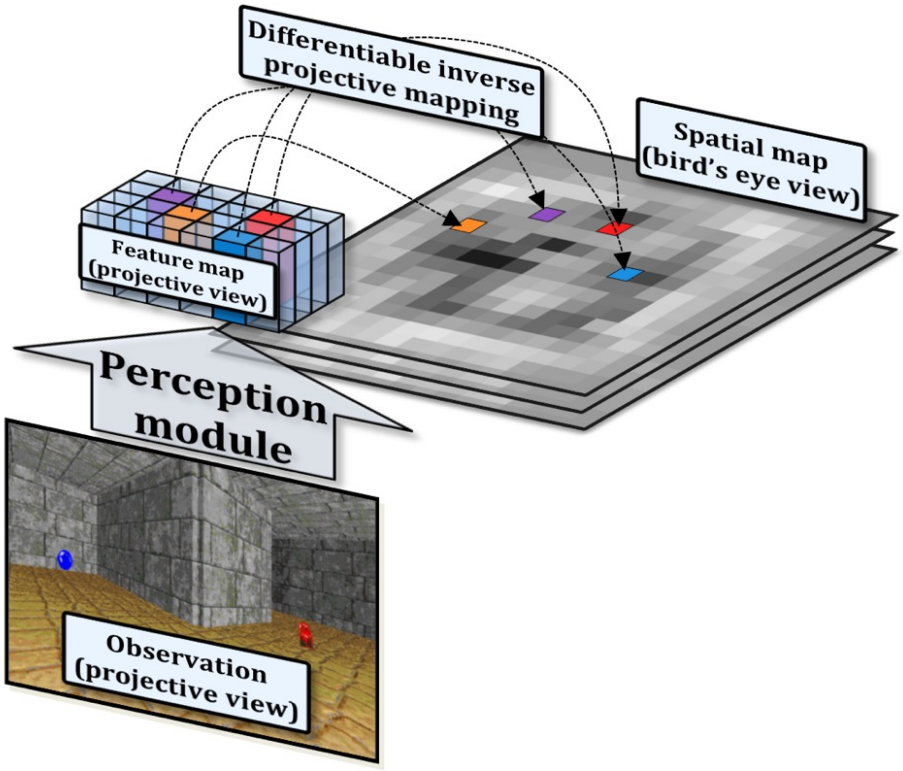
Beeching, Dibangoye, Simonin, Wolf,
EgoMap: Projective mapping and structured egocentric memory for Deep RL,
ECML-PKKD 2020



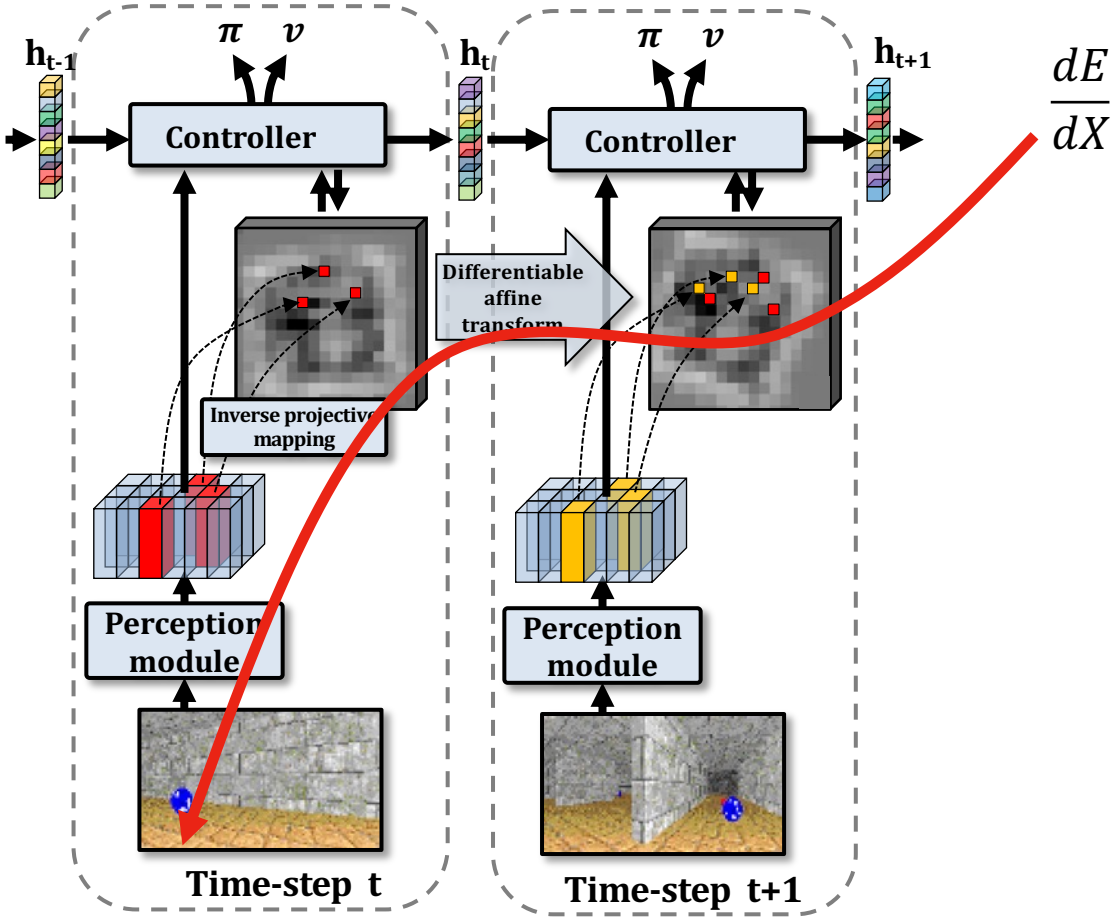
Topological map
(=Graph)

Beeching, Dibangoye, Simonin, Wolf,
Learning to plan with uncertain topological maps,
ECCV 2020

Projective mapping



Learning agent control/behavior



A single forward pass

1. Transform the map to the agent's egocentric frame of reference

$$\hat{M}_t = \text{Affine}(M_{t-1}, dx_t, dy_t, d\phi_t)$$

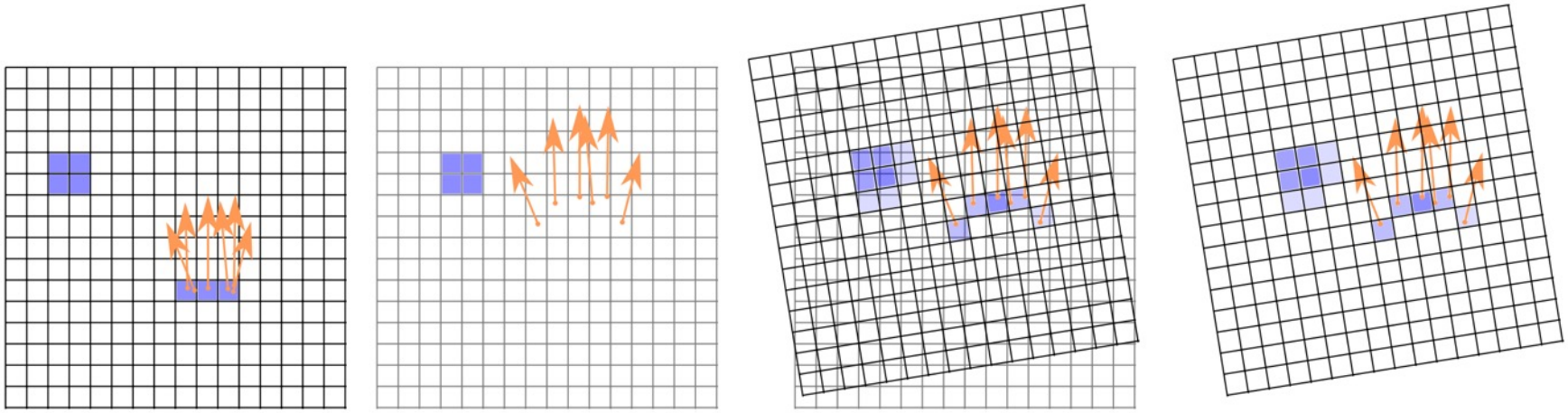
2. Update the map to include new observations

$$\tilde{M}_t = \text{InverseProject}(s_t, D_t) \qquad M'_t = \text{Combine}(\hat{M}_t, \tilde{M}_t)$$

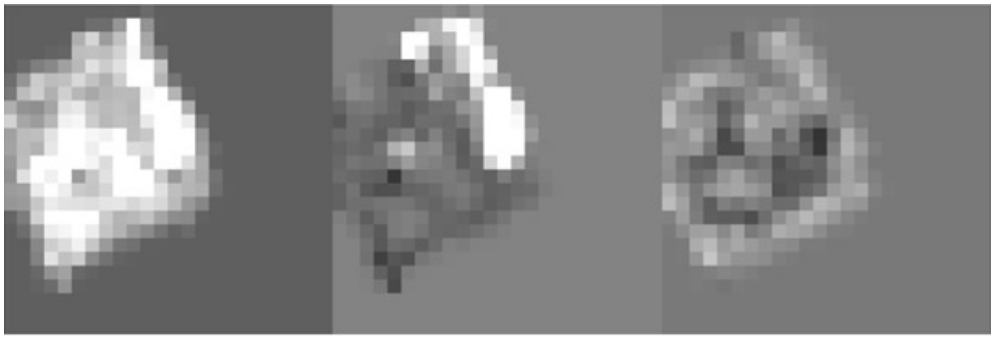
3. Perform a global read and attention based read,

$$r_t = \text{Read}(M'_t) \qquad c_t = \text{Context}(M'_t, s_t, r_t)$$

Occupancy Grid vs. Egomap



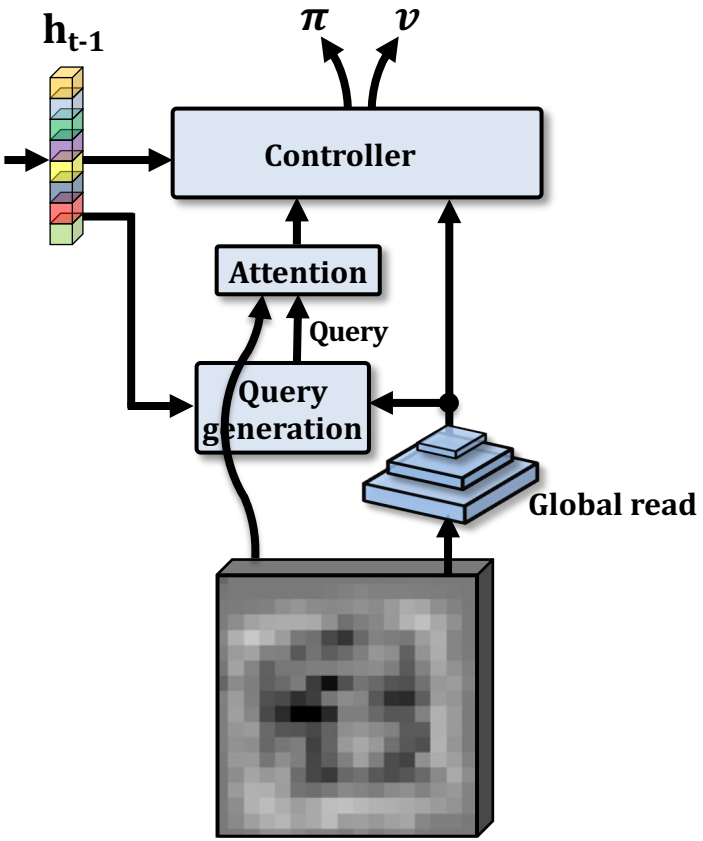
[Rummelhard, Negre, Laugier, 2015]

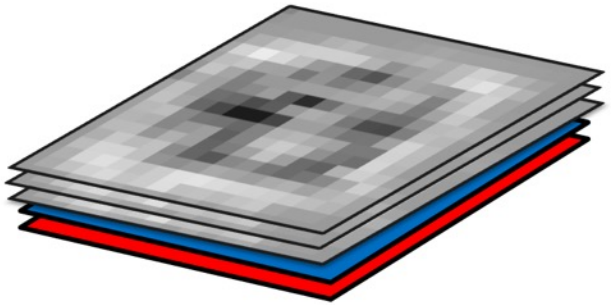


[Beeching, Dibangoye, Simonin, Wolf, ECML-PKDD 2020]

Querying spatial memory

The network learns to query for specific content

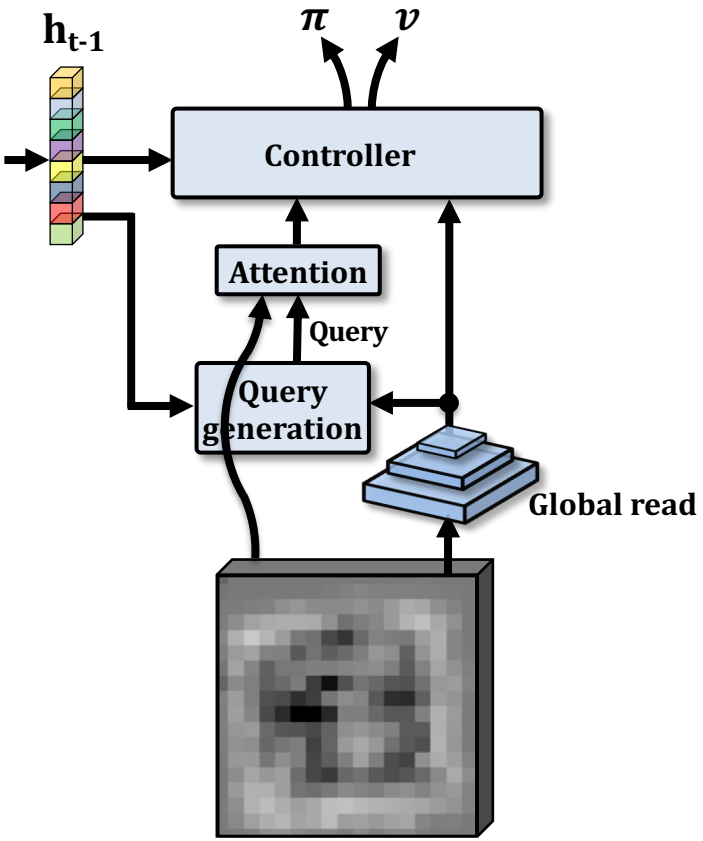


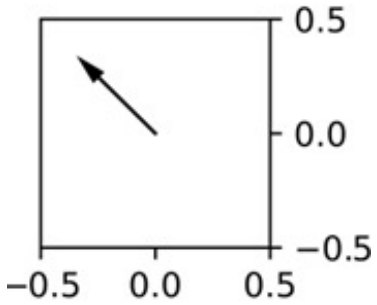


Ease localization with coordinate planes

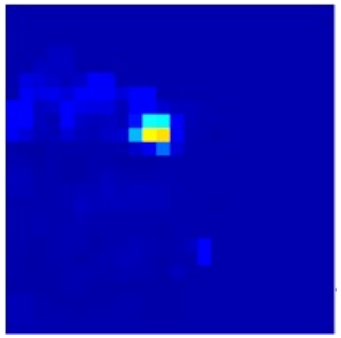
Querying spatial memory

The network learns to query for specific content



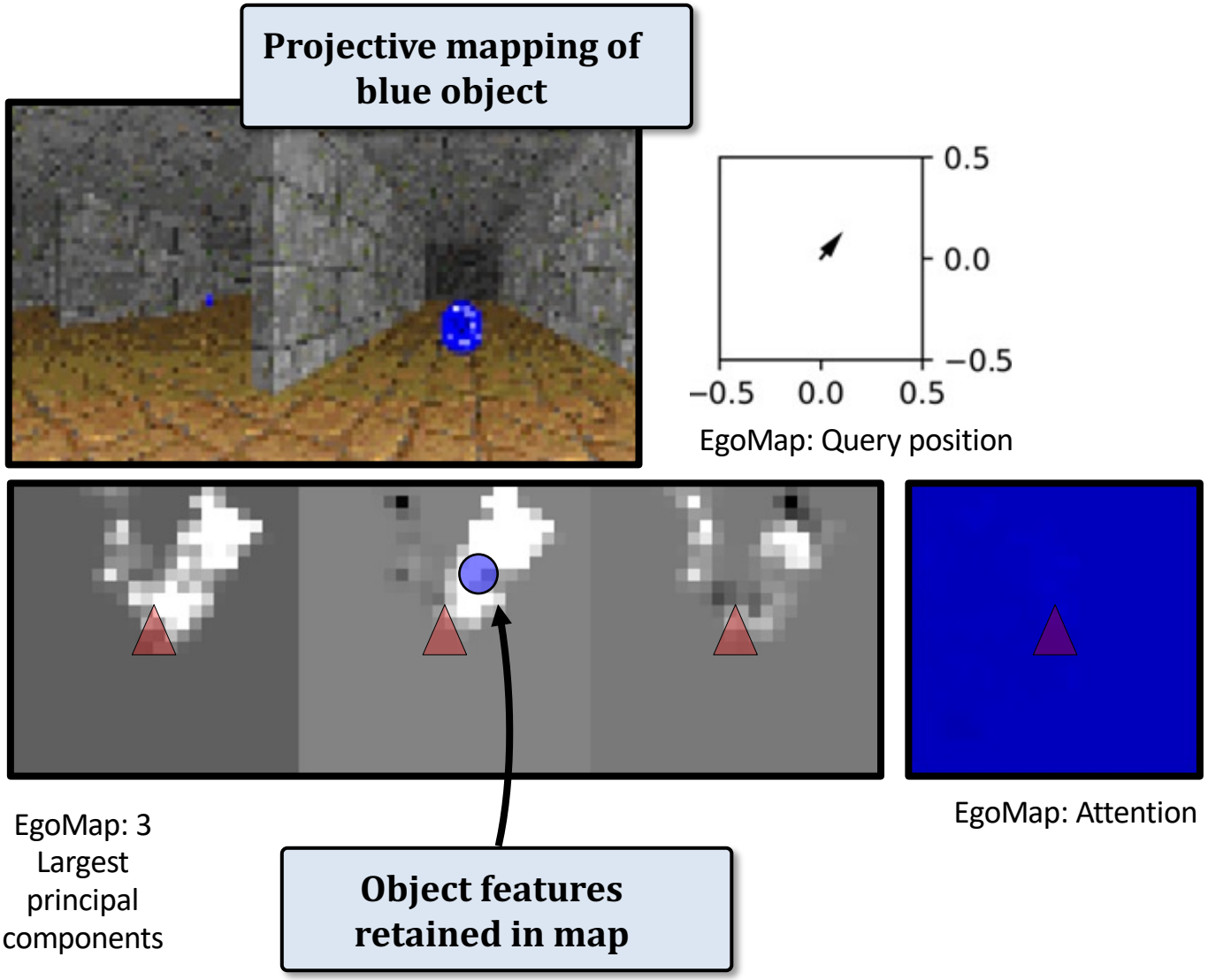


Attention: Spatial direction

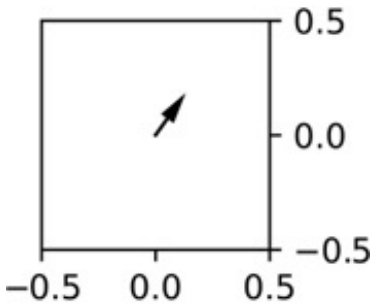


Attention distribution

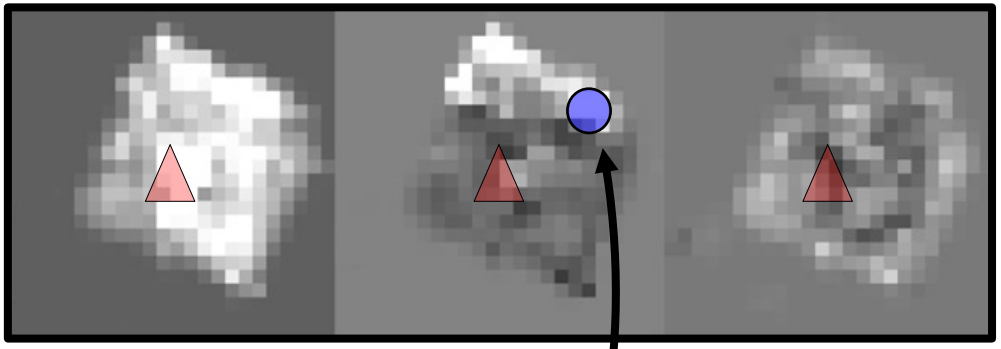
6 item scenario: time-step 005



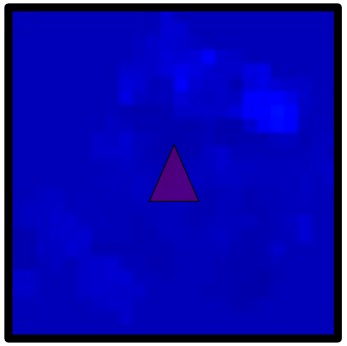
6 item scenario: time-step 105



EgoMap: Query position



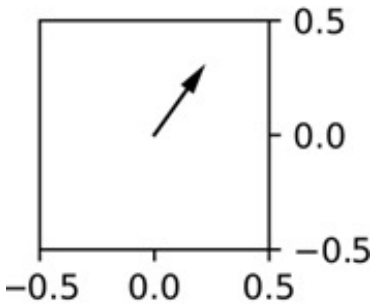
EgoMap: 3 Largest principal components



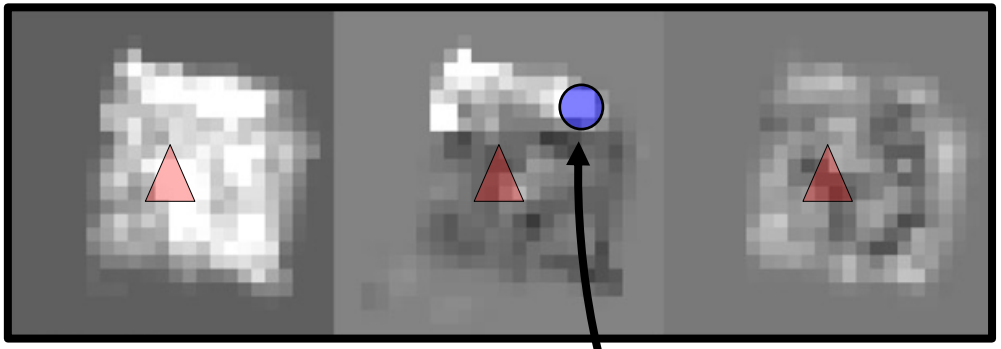
EgoMap: Attention

Object features retained in map

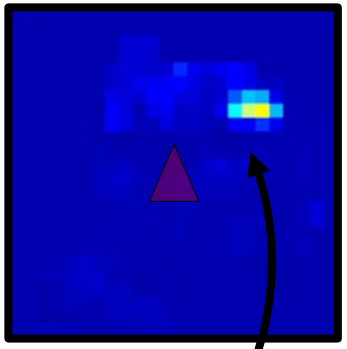
6 item scenario: time-step 108



EgoMap: Query position



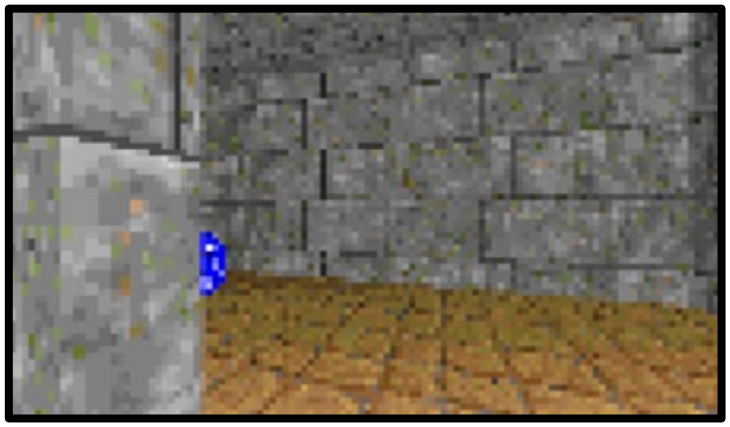
EgoMap: 3 Largest principal components

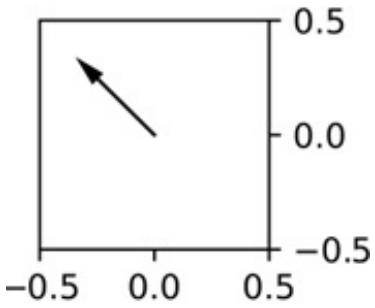


EgoMap: Attention

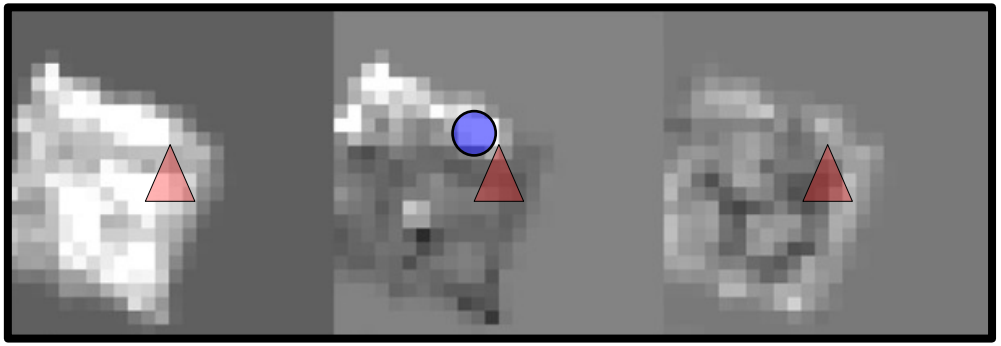
Collection of object n-1 triggers attention to object n

6 item scenario: time-step 134

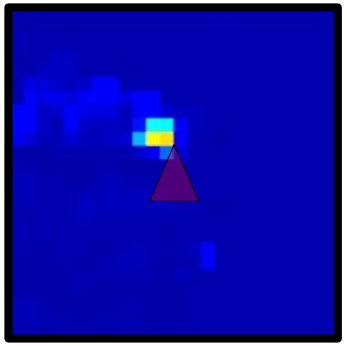




EgoMap: Query position

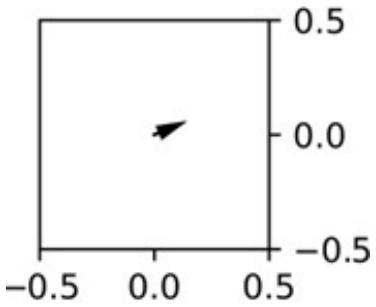


EgoMap: 3 Largest principal components

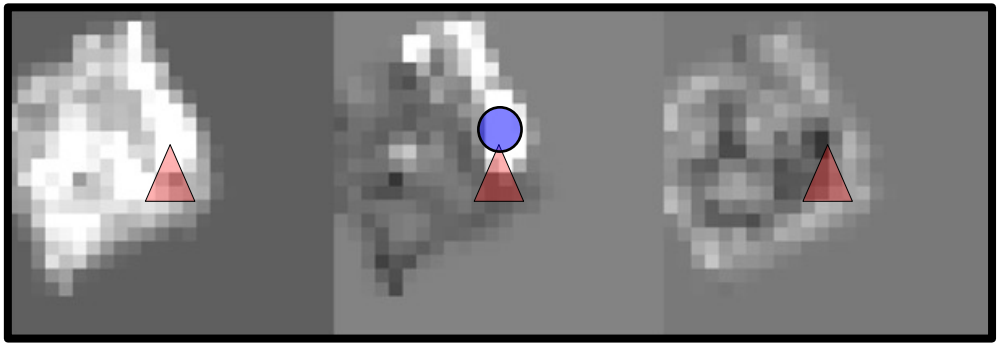


EgoMap: Attention

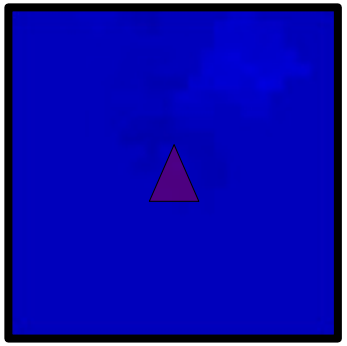
6 item scenario: time-step 140



EgoMap: Query position



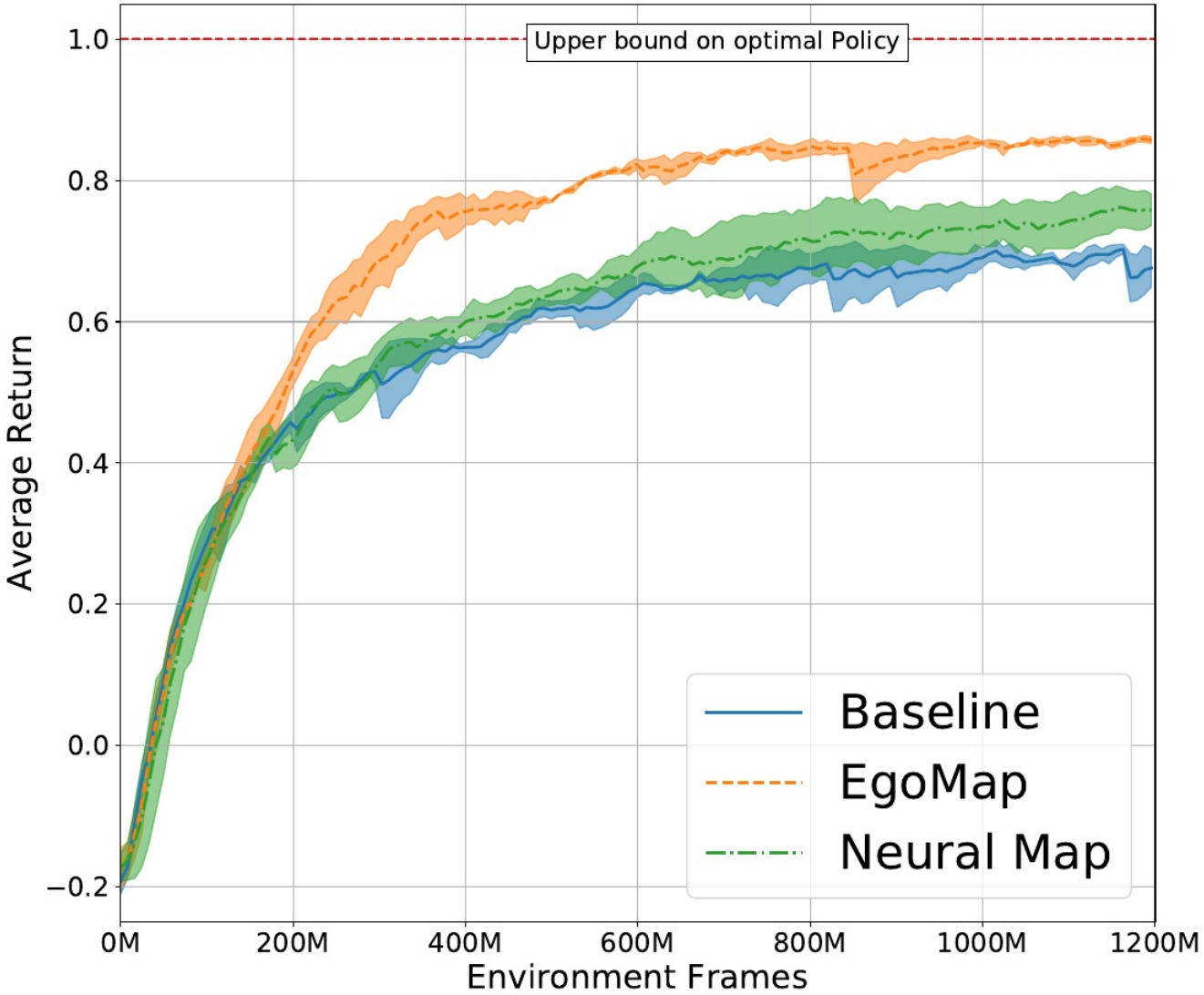
EgoMap: 3 Largest principal components



EgoMap: Attention

When the object is not occluded, the agent does not attend to it

Results


Upper bound on optimal Policy
1.0
0.8
0.6
0.4
0.2
0.0
−0.2
Average Return
0M
200M
400M
600M
800M
1000M
1200M
Environment Frames
Baseline
EgoMap
Neural Map

Quantitative results

	Scenario							
	4 item		6 item		Find and Return		Labyrinth	
Agent	**Train**	**Test**	**Train**	**Test**	**Train**	**Test**	**Train**	**Test**
Random	-0.179	-0.206	-0.21	-0.21	-0.21	-0.21	-0.115	-0.086
Baseline	2.341 ± 0.026	2.266 ± 0.035	2.855 ± 0.164	2.545 ± 0.226	0.661 ± 0.003	0.633 ± 0.027	0.73 ± 0.02	0.694 ± 0.009
Neural Map	2.339 ± 0.038	2.223 ± 0.040	2.750 ± 0.062	2.465 ± 0.034	0.825 ± 0.070	0.723 ± 0.026	**0.769 ± 0.042**	0.706 ± 0.018
EgoMap	**2.398 ± 0.014**	**2.291 ± 0.021**	**3.214 ± 0.007**	**2.801 ± 0.048**	**0.893 ± 0.007**	**0.848 ± 0.017**	0.753 ± 0.002	**0.732 ± 0.016**
Optimum	2.5	2.5	3.5	3.5	1	1	1	1

Mapping objects required

Not required

Results: ablation study

Ablation	Train	Test
Baseline	0.668 ± 0.028	0.662 ± 0.036
No global read	0.787 ± 0.007	0.771 ± 0.029
No query	0.838 ± 0.003	0.811 ± 0.013
No query temperature	0.845 ± 0.014	0.815 ± 0.019
No query position	0.839 ± 0.007	0.814 ± 0.008
Cosine query	0.847 ± 0.011	0.814 ± 0.017
L1 query	$\mathbf{0.851 \pm 0.014}$	$\mathbf{0.828 \pm 0.011}$

Quantitative analysis: Ablations

Ablation	Train	Test
Baseline	0.668 ± 0.028	0.662 ± 0.036
No global read	0.787 ± 0.007	0.771 ± 0.029
No query	0.838 ± 0.003	0.811 ± 0.013
No query temperature	0.845 ± 0.014	0.815 ± 0.019
No query position	0.839 ± 0.007	0.814 ± 0.008
EgoMap	0.847 ± 0.011	0.814 ± 0.017
EgoMap (L1 query)	**0.851 ± 0.014**	**0.828 ± 0.011**

Noise std.	Average return
0.00	0.702 ± 0.026
0.02	0.686 ± 0.027
0.04	0.669 ± 0.031
0.06	0.623 ± 0.018
0.08	0.585 ± 0.017
0.10	0.568 ± 0.023
0.12	0.575 ± 0.020
0.14	0.578 ± 0.023
0.16	0.546 ± 0.017
0.20	0.537 ± 0.031
Baseline	0.527 ± 0.047

Self-supervision for Deep-RL

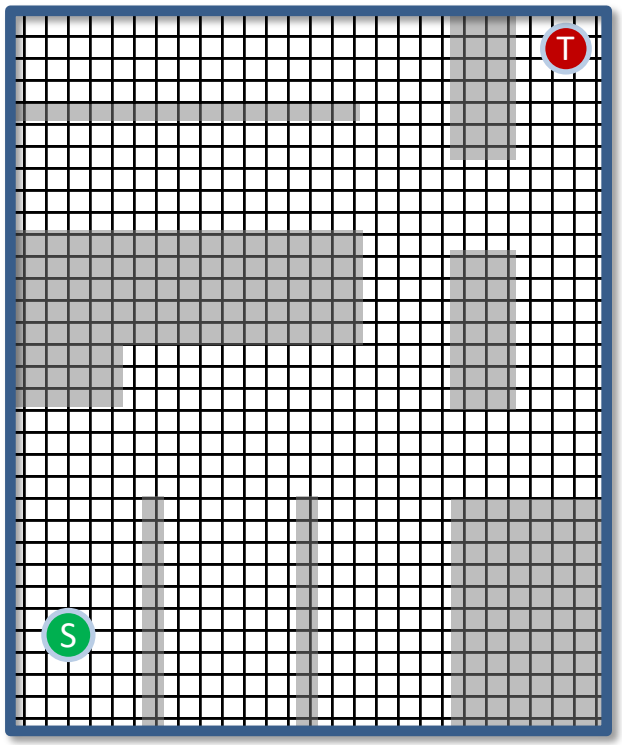
1st ranked at Multi-ON Challenge (CVPR 2021)

Rank	Team	Progress	PPL	Success	SPL
1	Lyon	67	44	55	35
2	SGoLAM	64	38	52	32
3	VIMP	57	36	41	26

1st place: Team Lyon, *Pierre Marza, Laetitia Matignon, Olivier Simonin, Christian Wolf*, Learning mapping and spatial reasoning with auxiliary tasks.

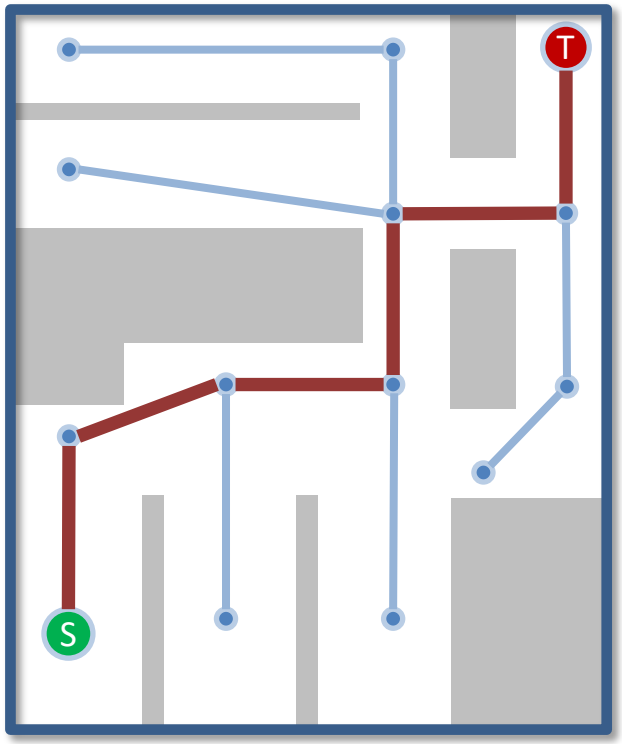
[Marza, Matignon, Simonin, Wolf, Submitted to BMVC 2021]

Spatial maps in robotics



Metric map
(=2D or 3D Grid)

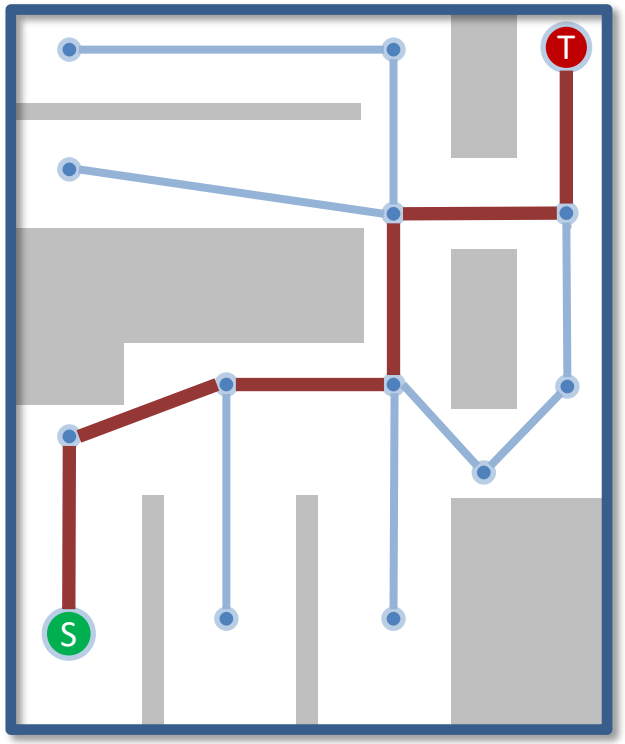
Beeching, Dibangoye, Simonin, Wolf,
EgoMap: Projective mapping and structured egocentric memory for Deep RL,
ECML-PKKD 2020



Topological map
(=Graph)

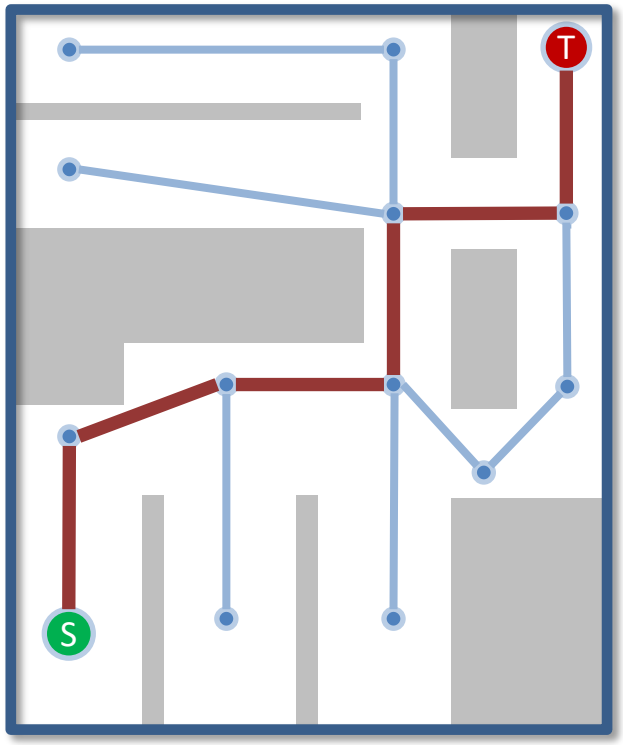
Beeching, Dibangoye, Simonin, Wolf,
Learning to plan with uncertain topological maps,
ECCV 2020

Uncertain topological maps

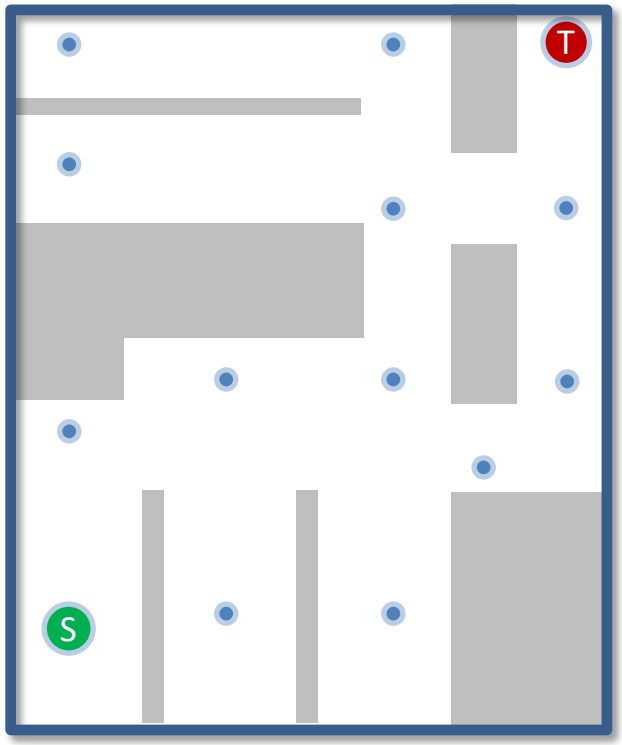


Ground truth

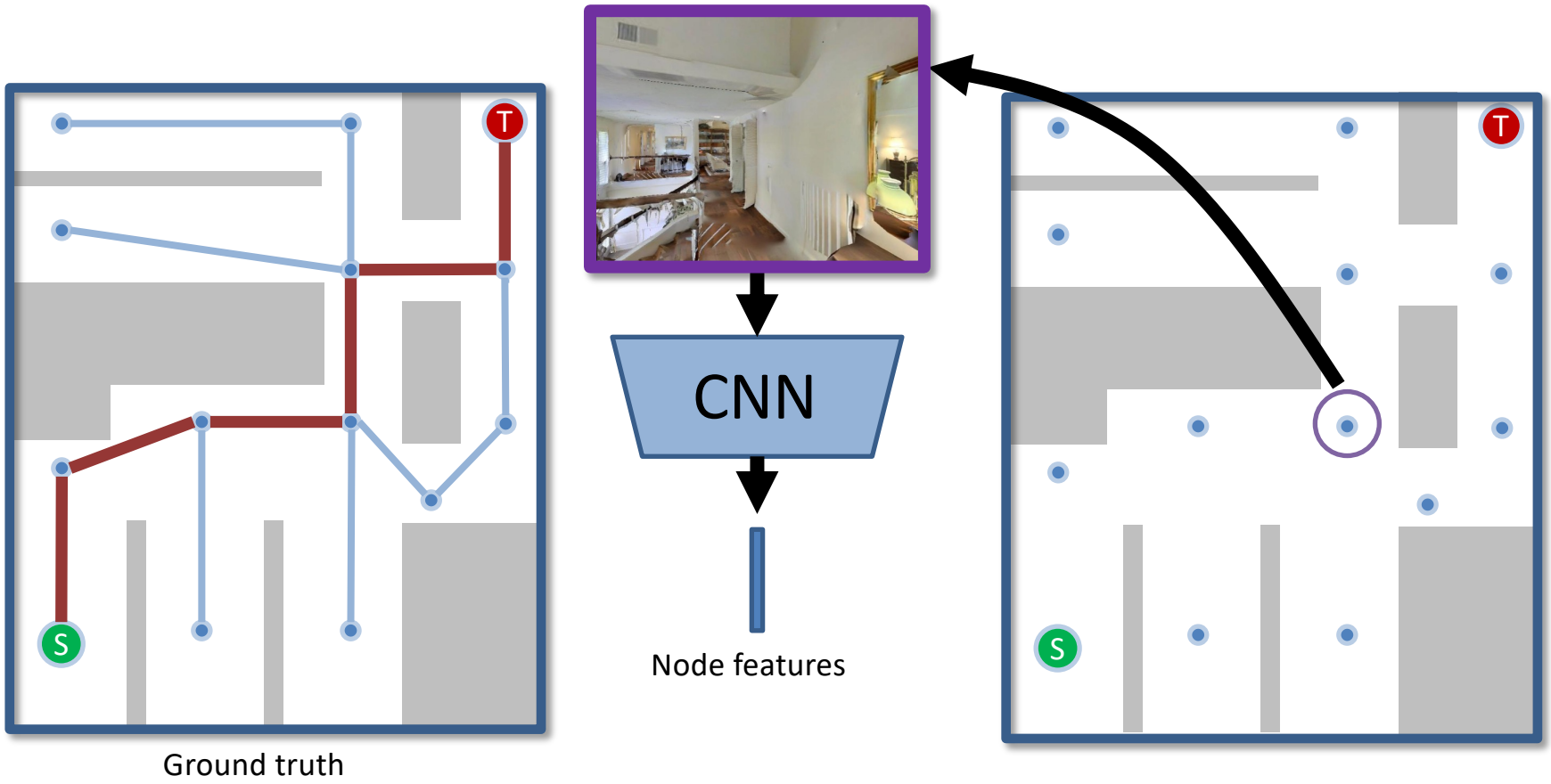
Uncertain topological maps



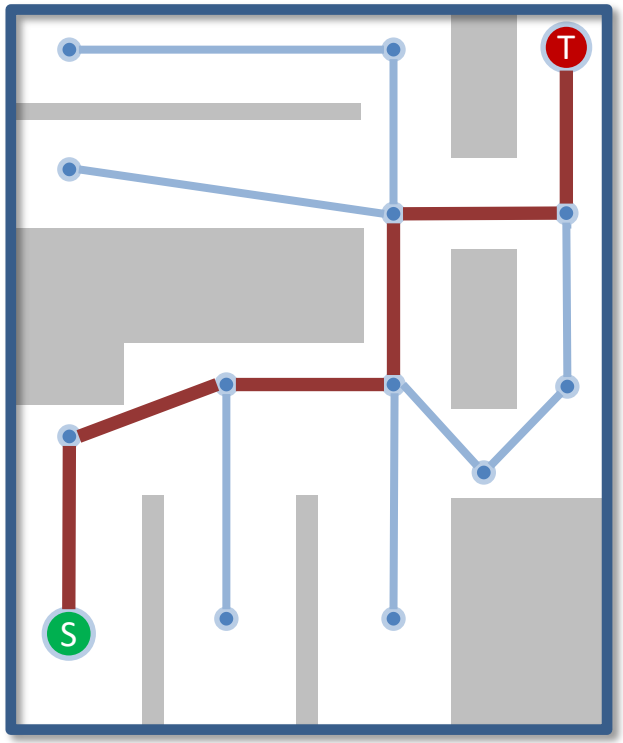
Ground truth



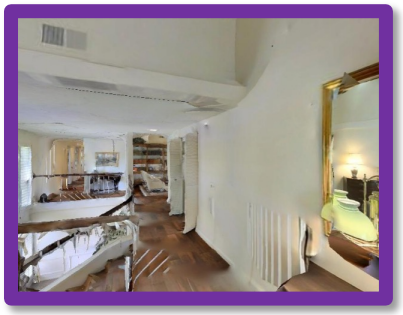
Uncertain topological maps

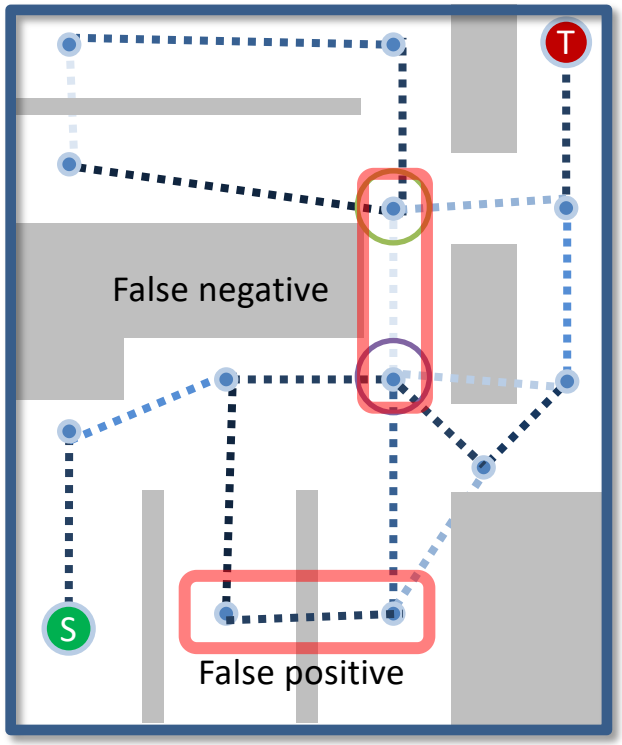


Uncertain topological maps



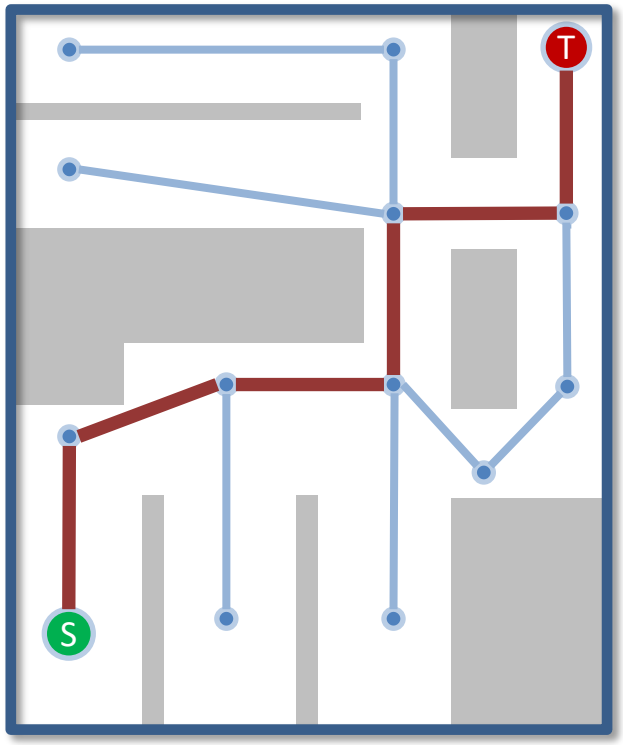
Ground truth



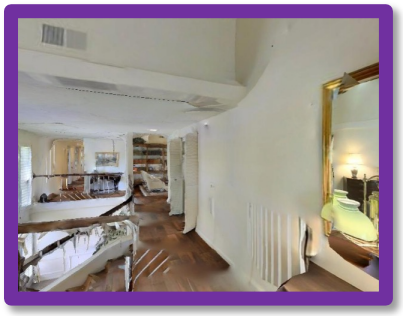


Uncertain graph

Uncertain topological maps

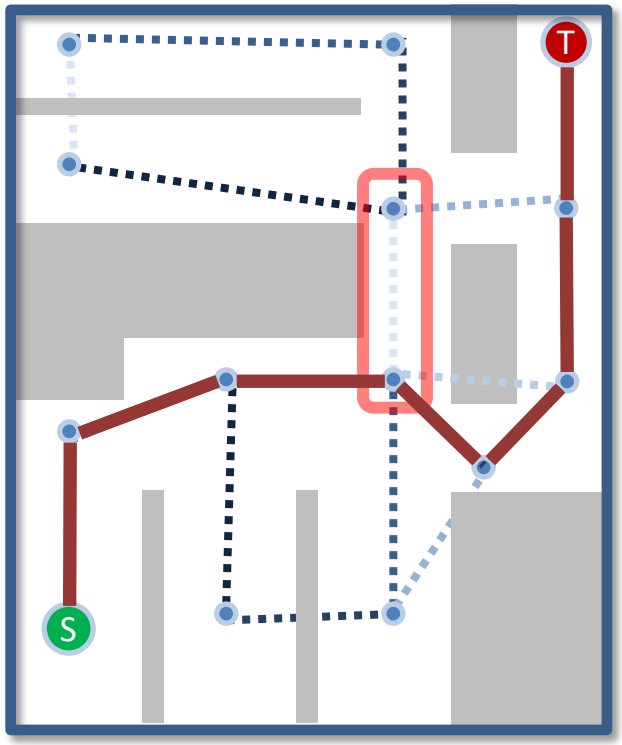


Ground truth



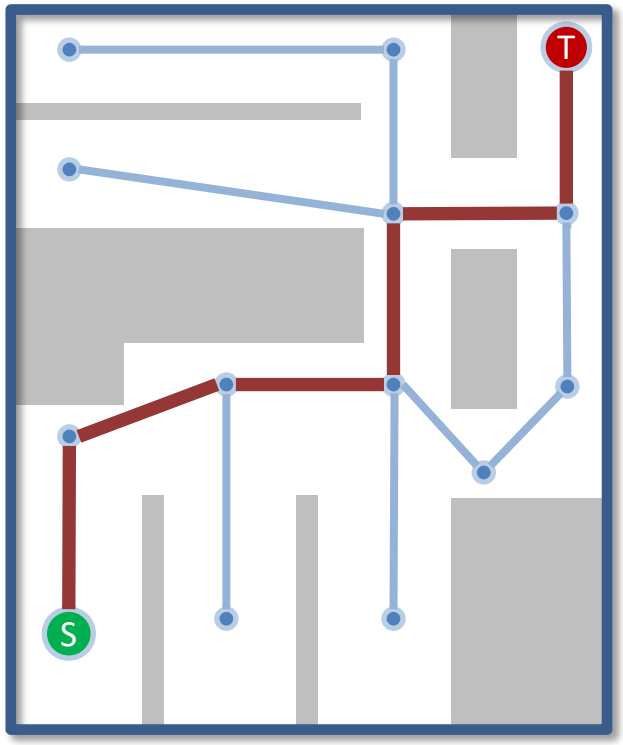
Classical planning:

- **Binary connectivity**
- **Cannot exploit visual information**

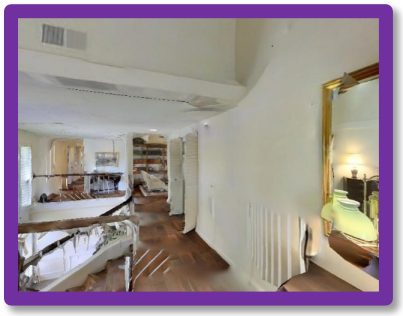


Uncertain graph

Uncertain topological maps



Ground truth

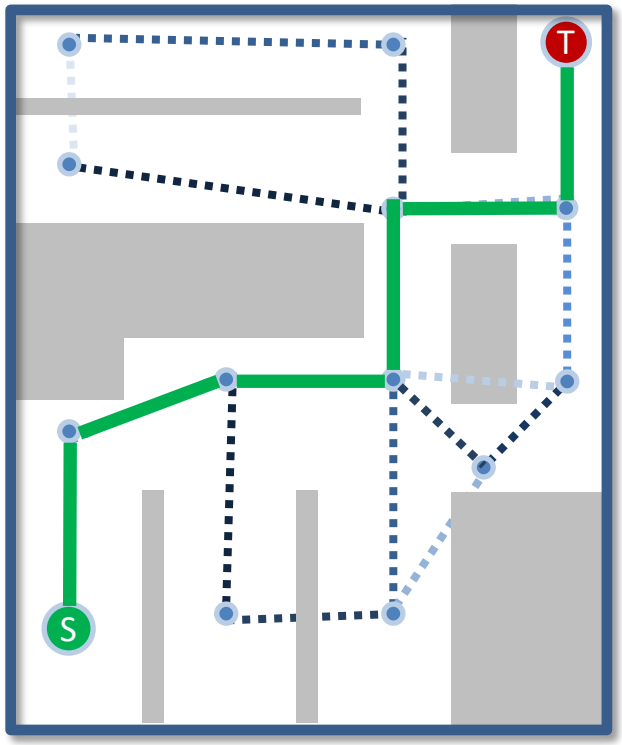


Classical planning:

- **Binary connectivity**
- **Cannot exploit visual information**

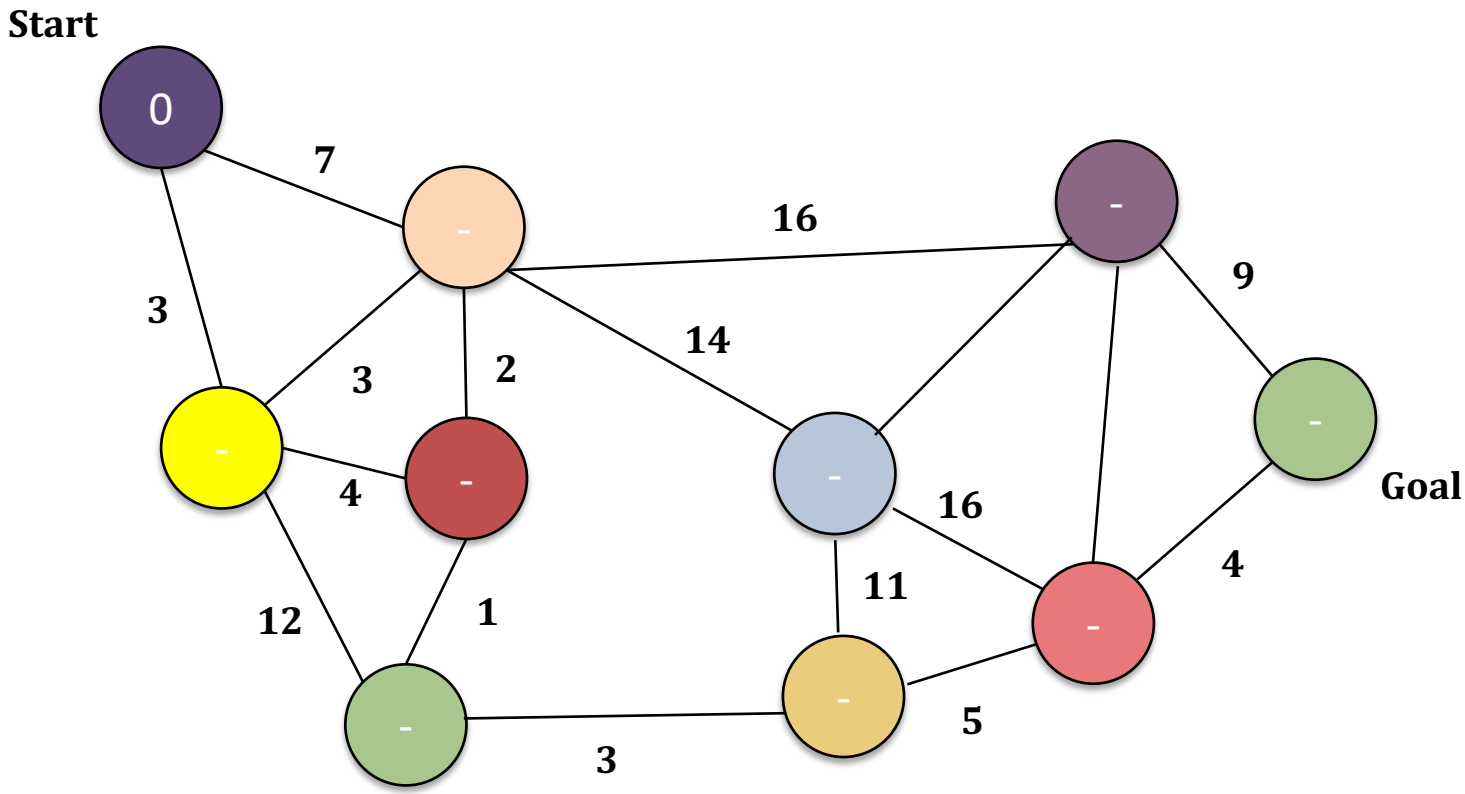
Neural planning:

- **Exploits uncertain connectivity**
- **Uses visual features**
- **Uses neighbor information**

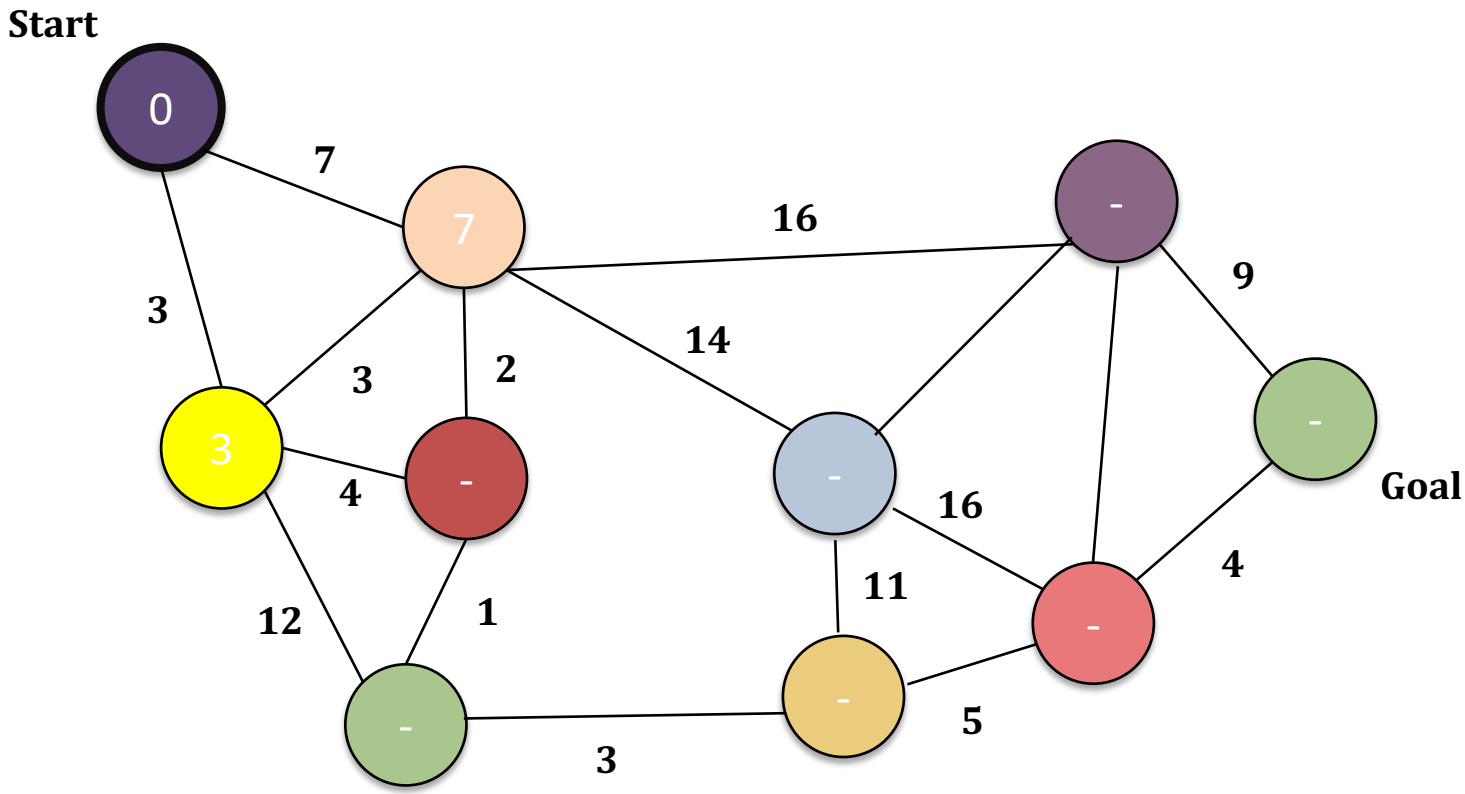


Uncertain graph

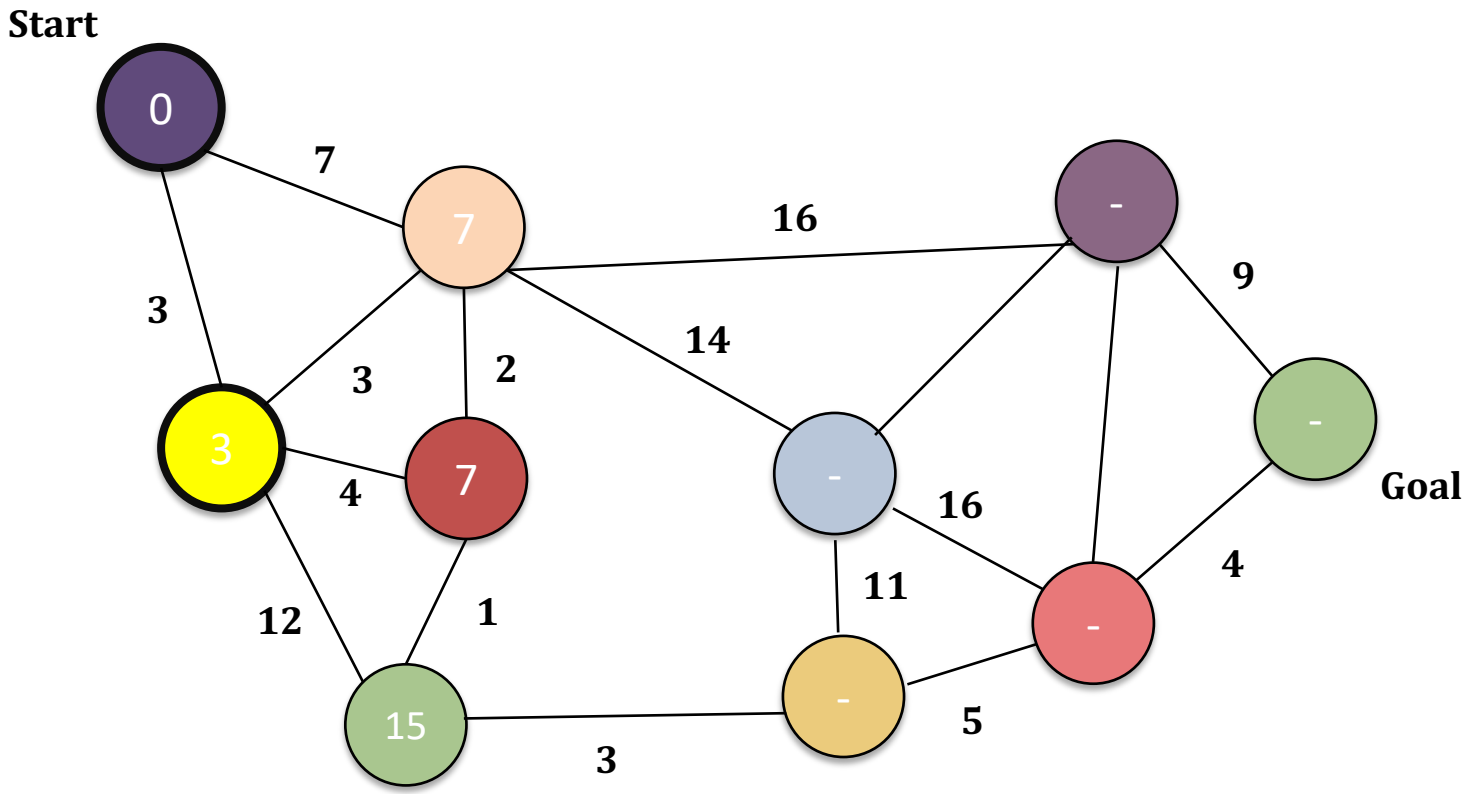
Shortest path problems: Dijkstra's algorithm



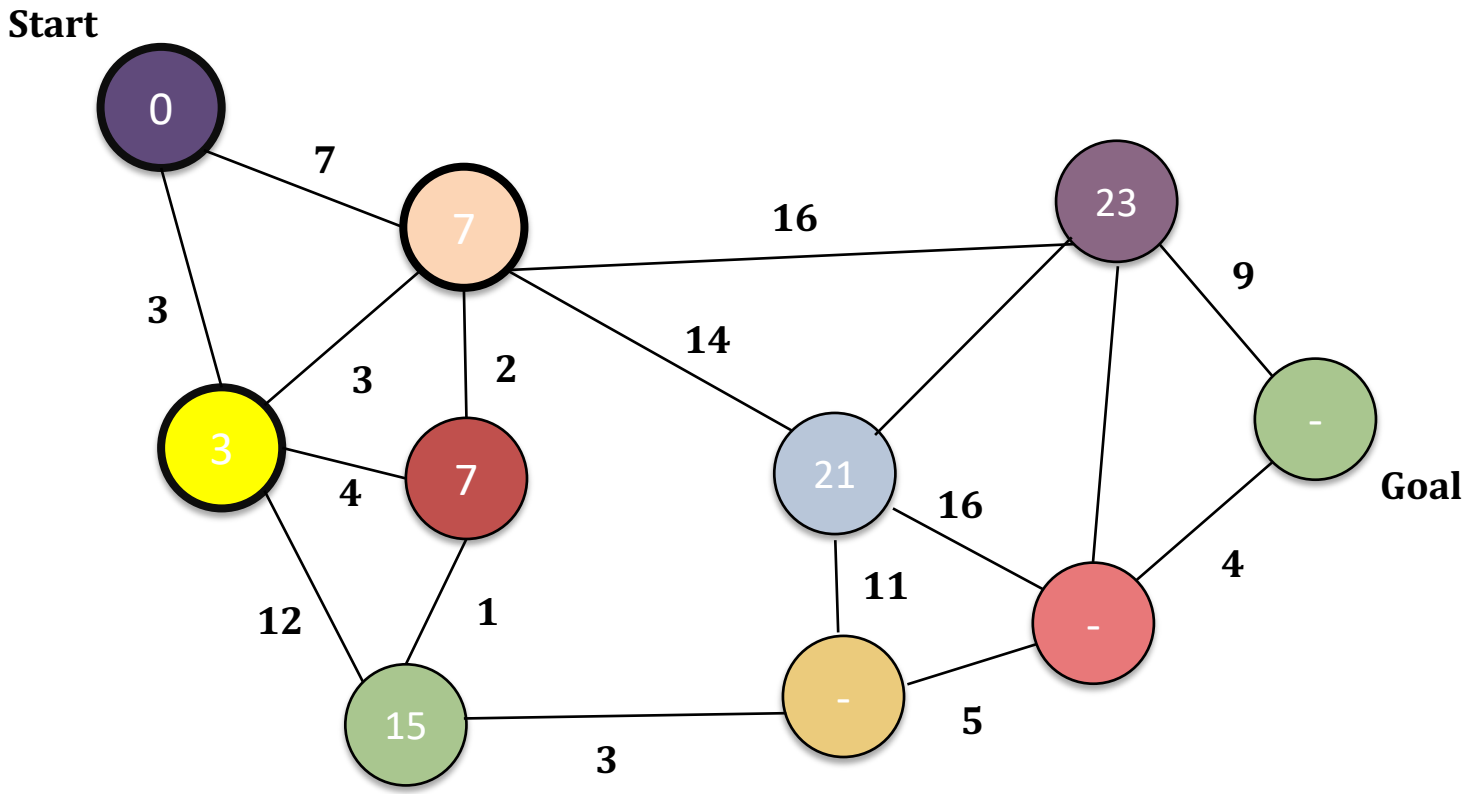
Shortest path problems: Dijkstra's algorithm



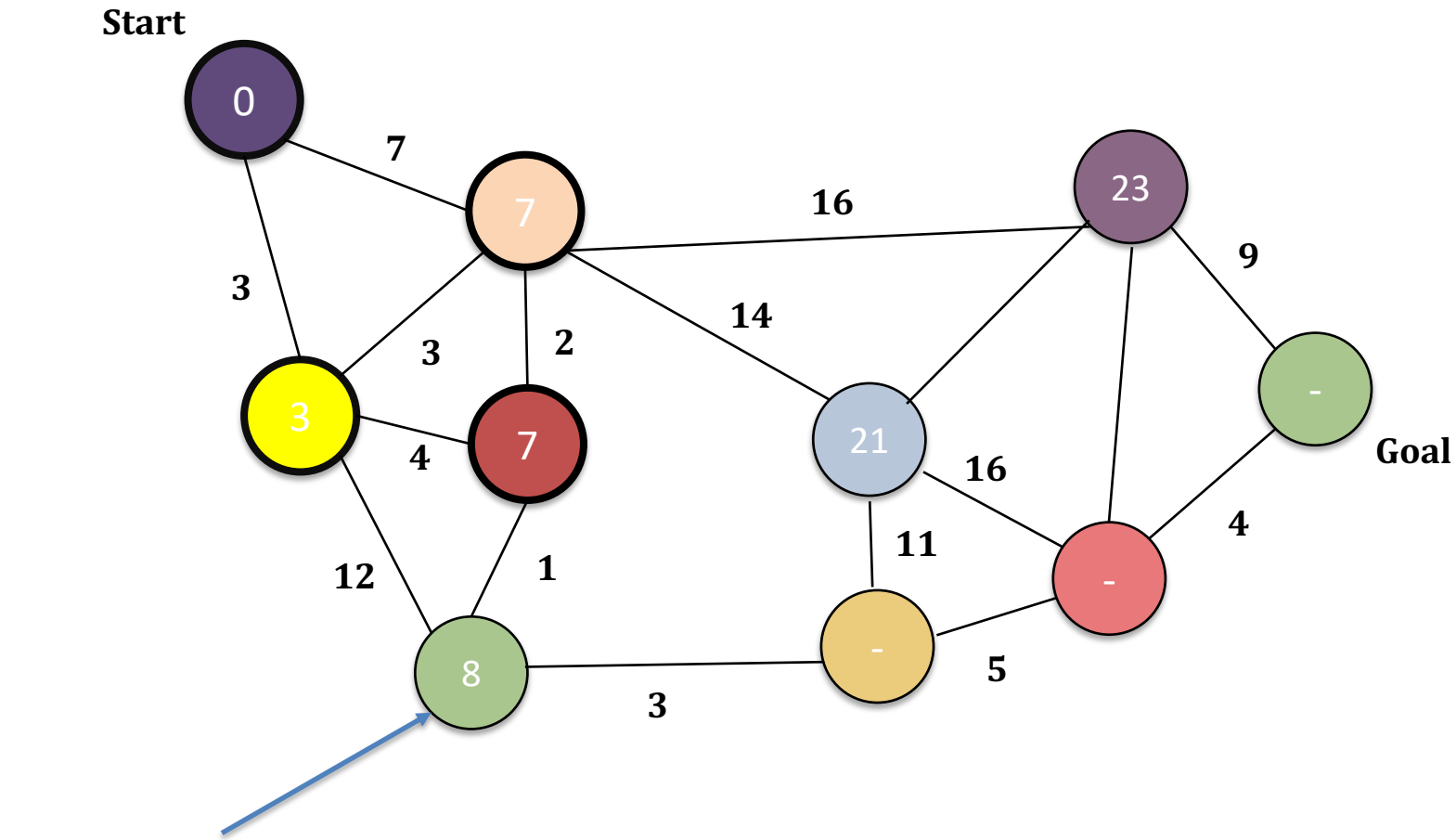
Shortest path problems: Dijkstra's algorithm



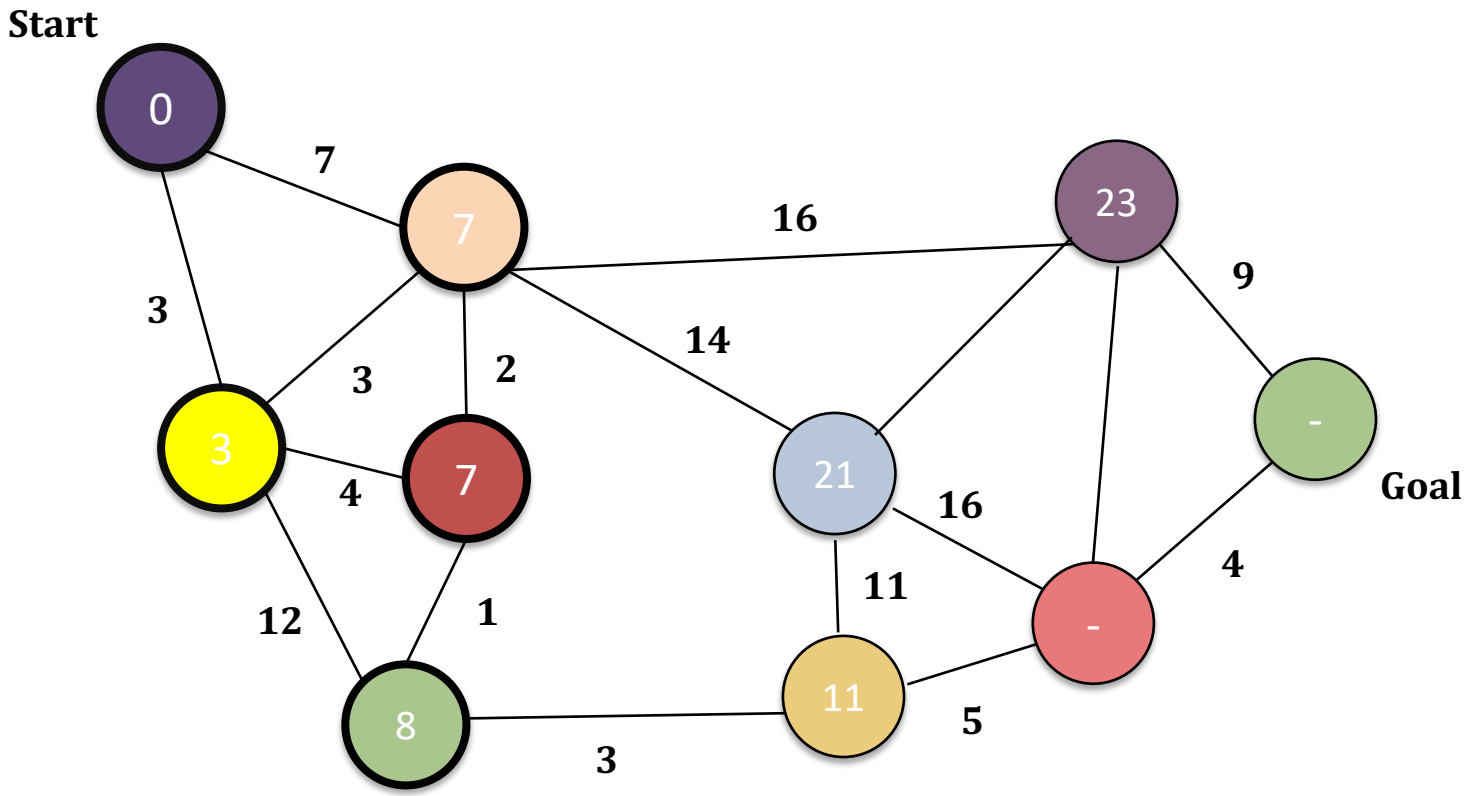
Shortest path problems: Dijkstra's algorithm



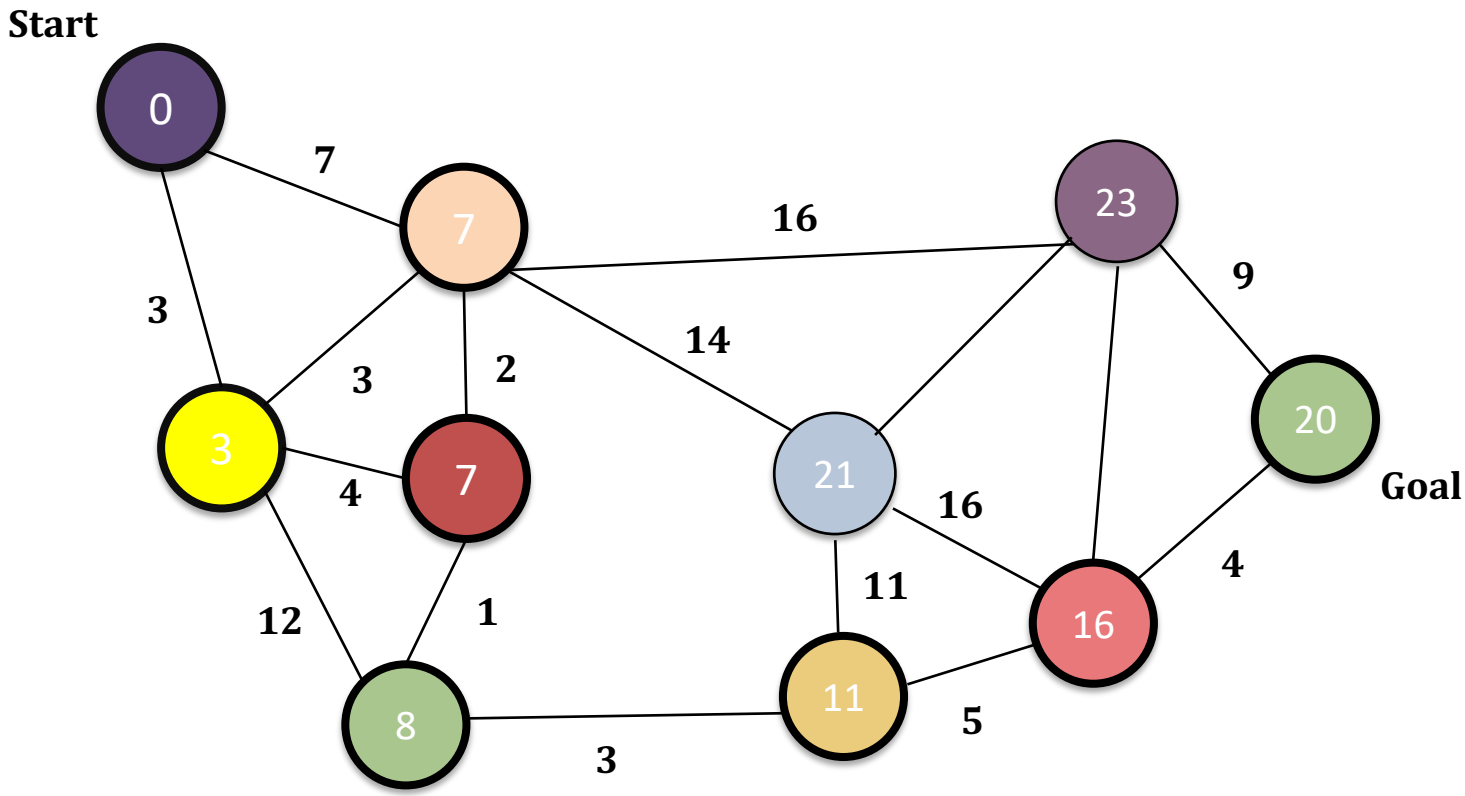
Shortest path problems: Dijkstra's algorithm



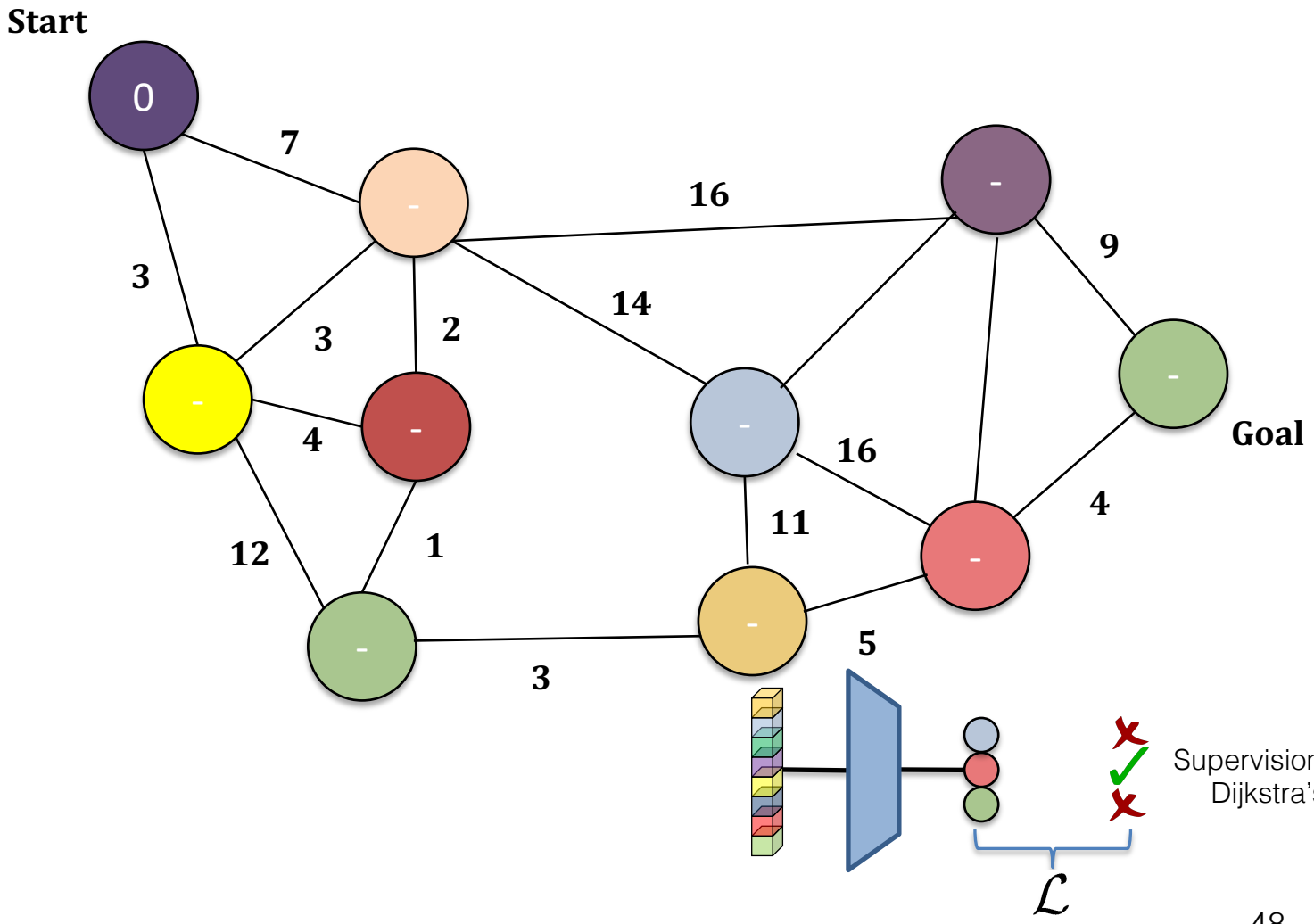
Shortest path problems: Dijkstra's algorithm

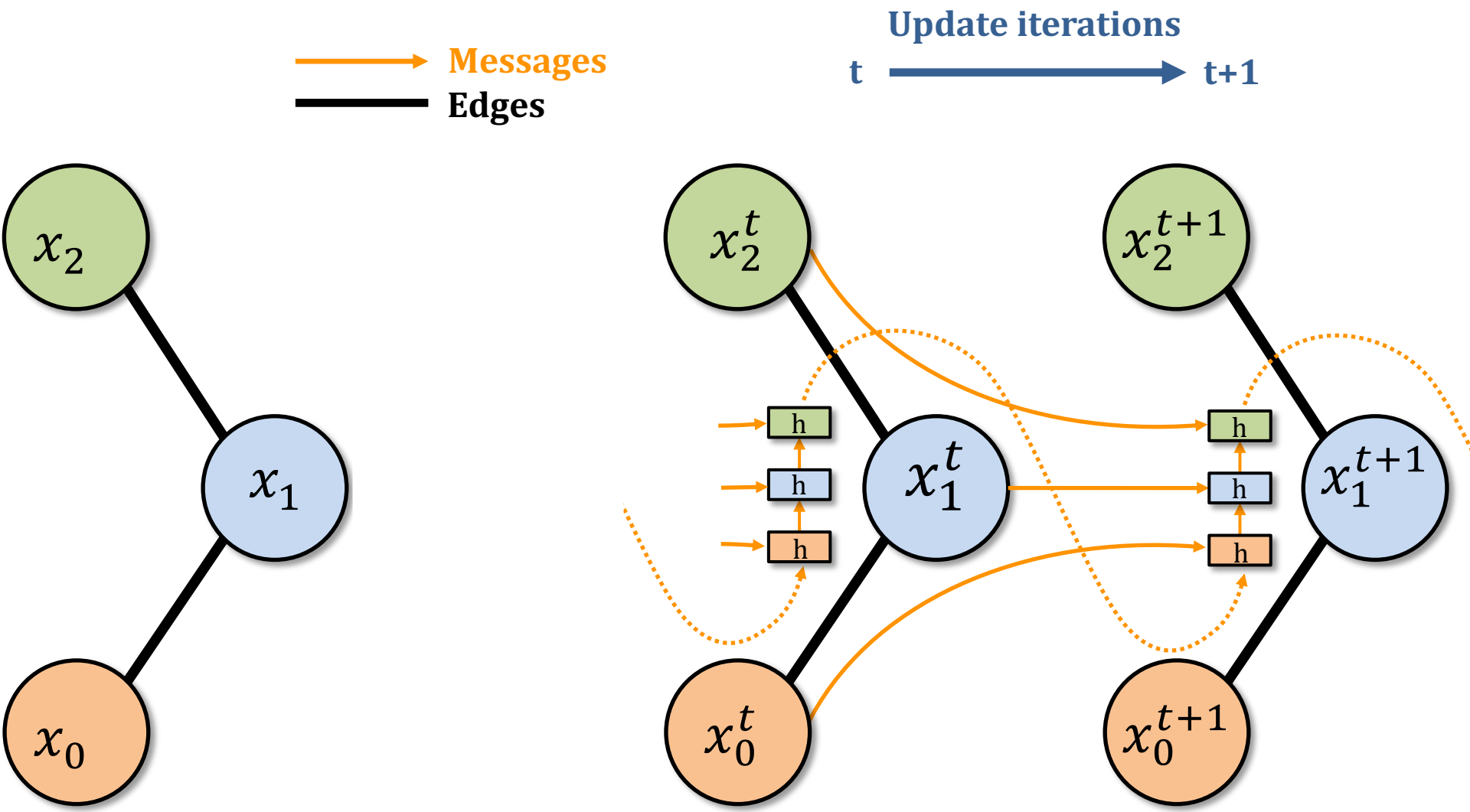


Shortest path problems: Dijkstra's algorithm



Planning as classification
with Graph Neural Networks




Messages
Edges
Update iterations
t
t+1
x_2
x_1
x_0
x_2^t
x_2^{t+1}
x_1^t
x_1^{t+1}
x_0^t
x_0^{t+1}
h
h
h
h
h
h

Results: Neural planner

(a) Uncertain graphs

Method	Acc	H-SPL
Symbolic (threshold)	0.114	0.184
Symbolic (custom cost)	0.115	0.269
Neural (w/o visual)	0.251	0.468
Neural (w visual)	**0.262**	**0.501**

(b) Ground truth graphs

Method	Acc	H-SPL
Symbolic (GT)	**1.00**	**1.00**
Neural planner (GT)	0.921	0.983

We can beat optimal algorithms ...

... by cheating ... data is King!

Results: Neural planner

(a) Uncertain graphs

Method	Acc	H-SPL
Symbolic (threshold)	0.114	0.184
Symbolic (custom cost)	0.115	0.269
Neural (w/o visual)	0.251	0.468
Neural (w visual)	**0.262**	**0.501**

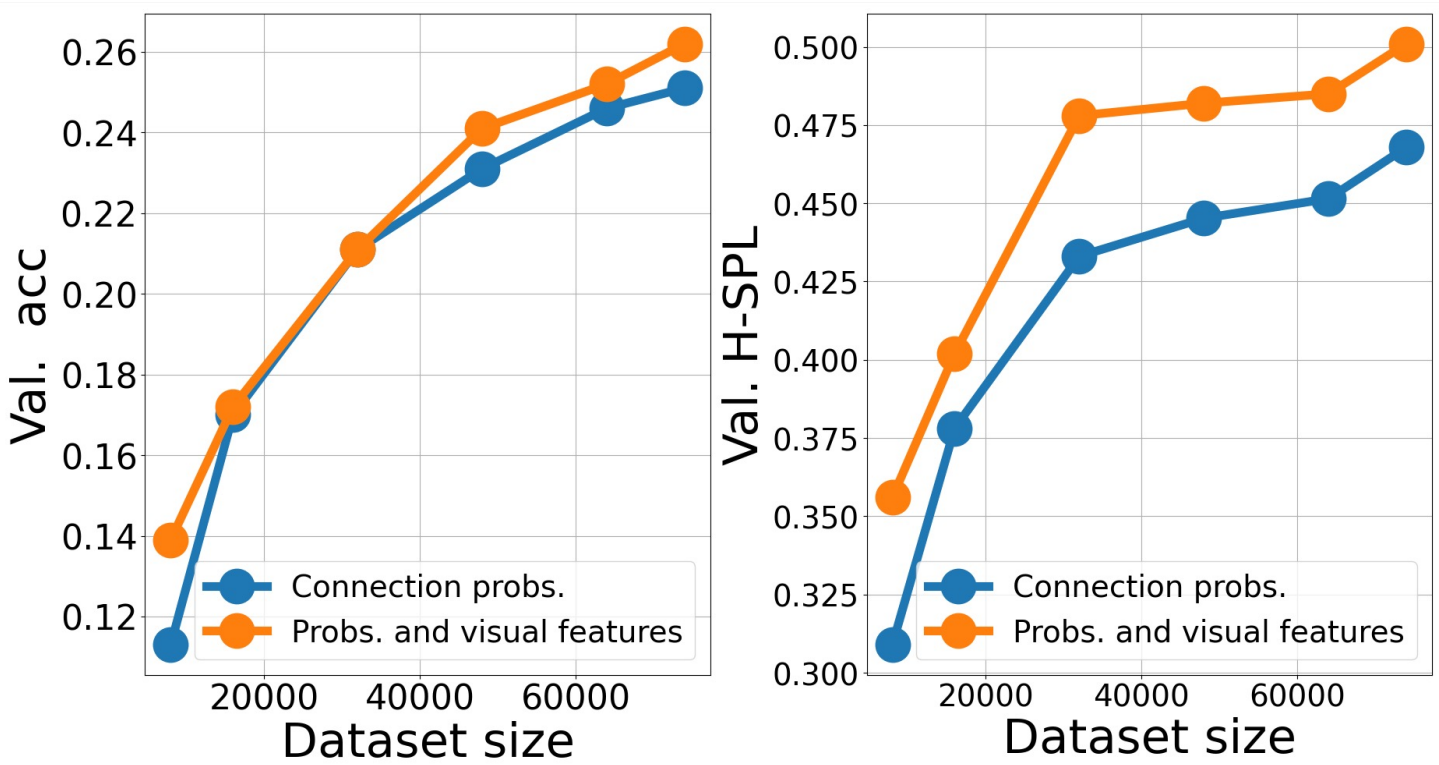
We can beat optimal algorithms ...

... by cheating ... data is King!

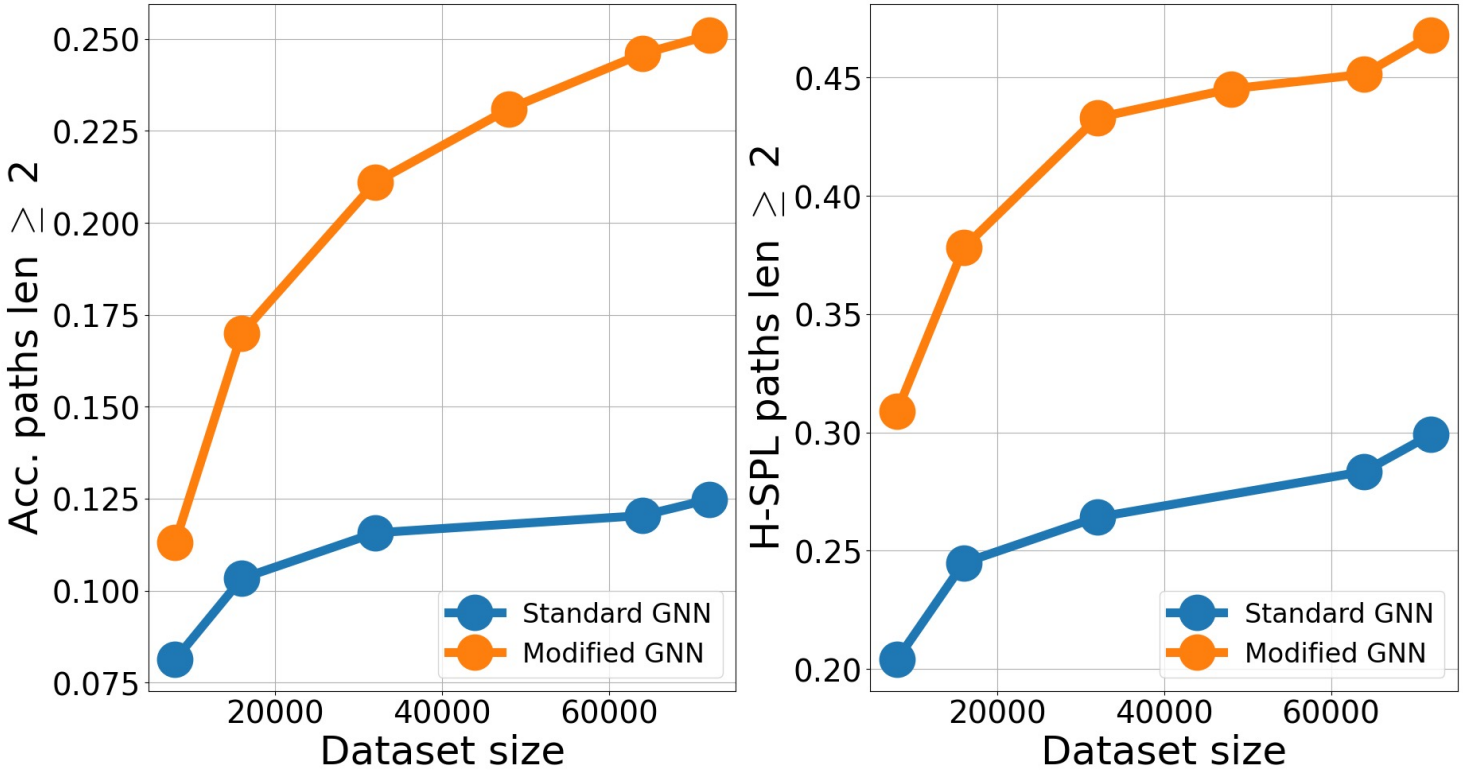
(b) Ground truth graphs

Method	Acc	H-SPL
Symbolic (GT)	**1.00**	**1.00**
Neural planner (GT)	0.921	0.983

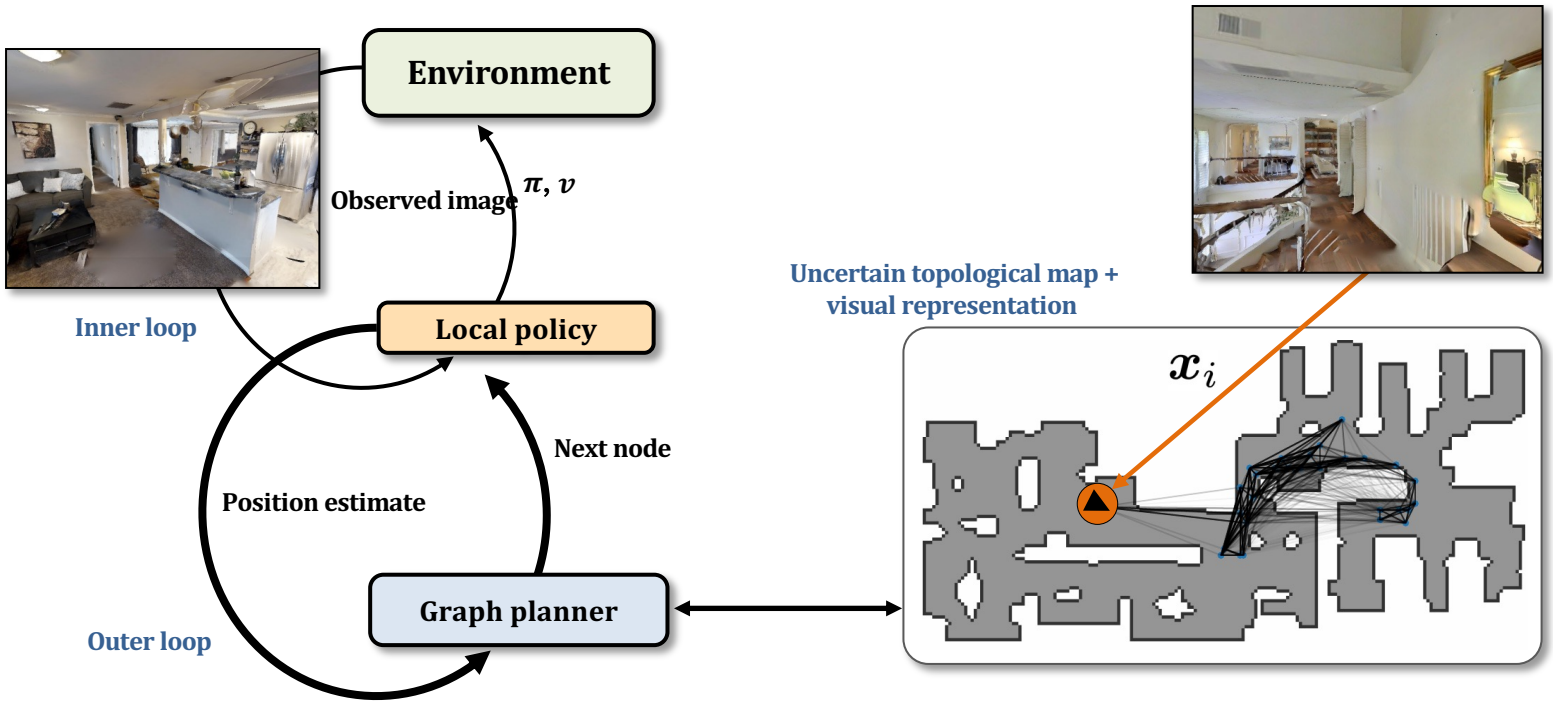
Ablation: Visual features



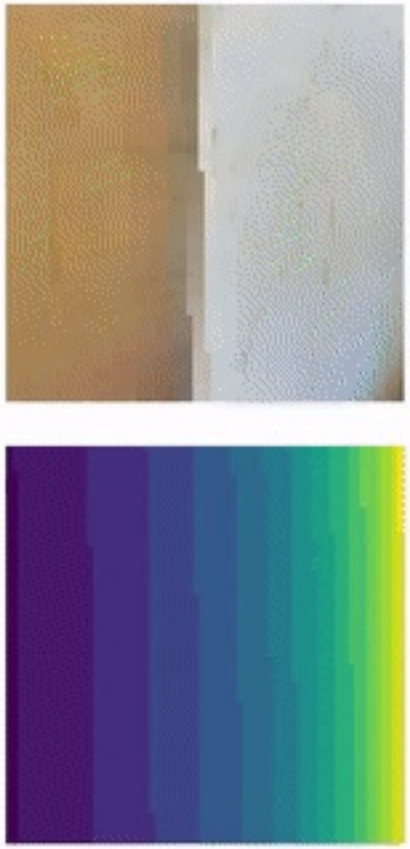
Ablation: GRU for min operation



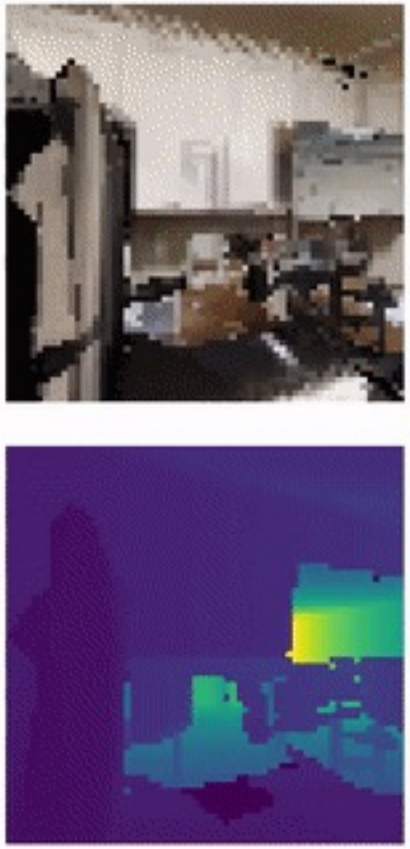
Hierarchical planning

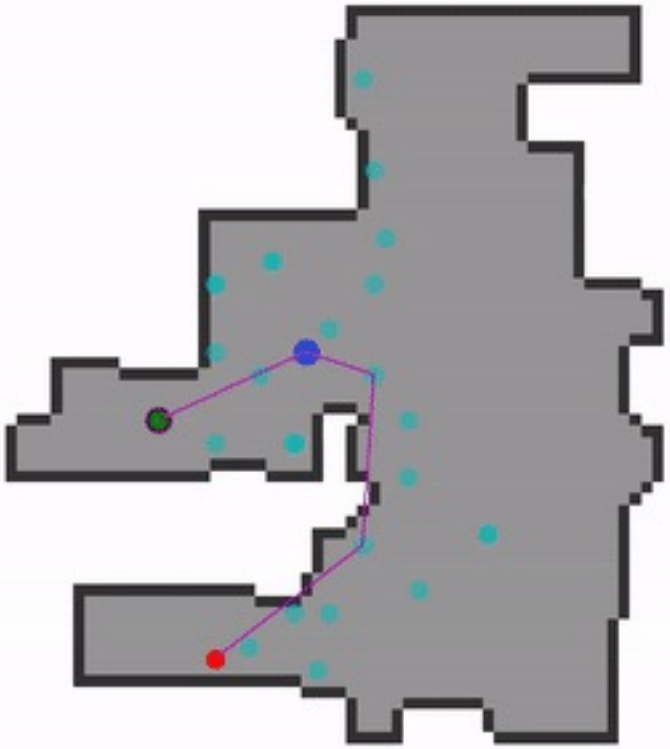


Hierarchical planning and control



Hierarchical planning and control

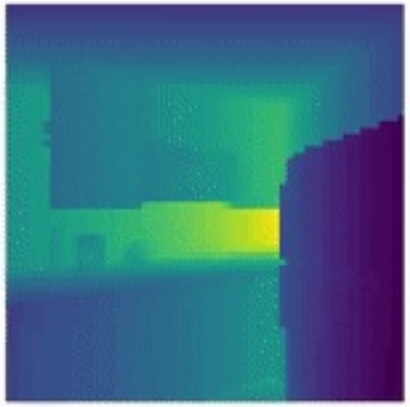


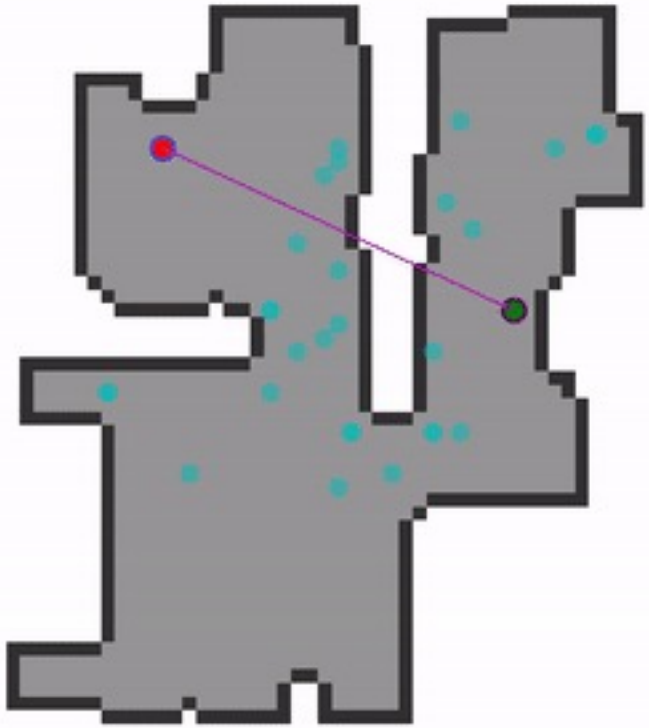


Results: Hierarchical planning and control

Method: Planner + Local policy	**Success rate**	**SPL**
Graph oracle (optimal point-goals, not comparable)	0.963	0.882
Random	0.152	0.111
Recurrent Image-goal agent	0.548	0.248
Symbolic (threshold)	0.621	0.527
Symbolic (custom cost)	0.707	0.585
Neural planner (sampling)	0.966	0.796
Neural planner (deterministic)	**0.983**	**0.877**

Failure Case





Navigation in real environments

NAVER LABS EUROPE

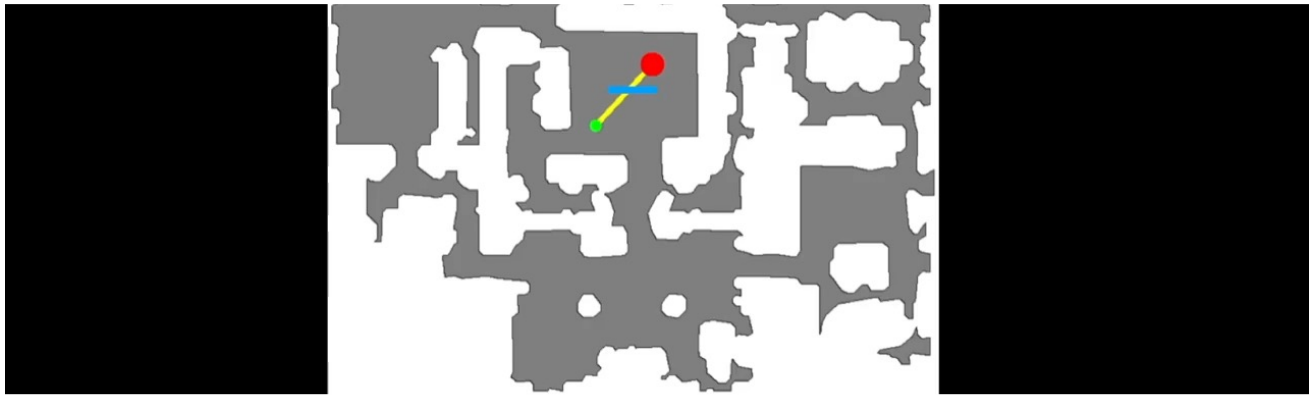

Assem Sadek


Guillaume Bono

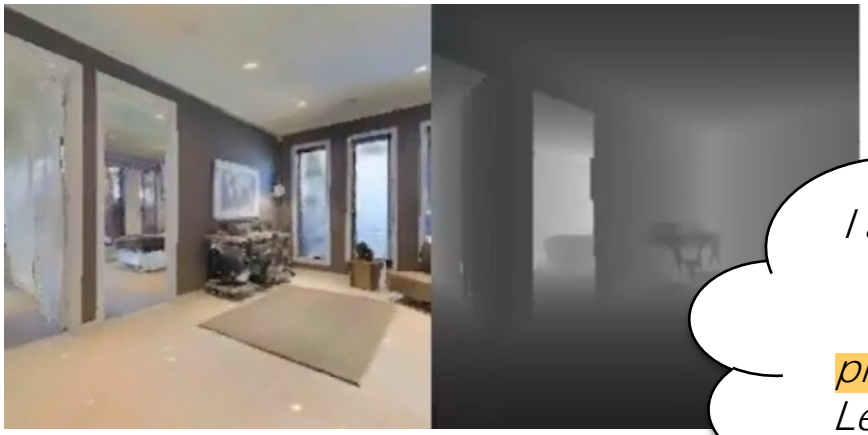

Boris Chidlovskii


Christian Wolf

Navigation in real environments



Language as Universal knowledge source



I am looking for a pillow and I a see a glimpse of a bedroom there, where pillows are frequently found. Let's check at this place first

Pierre Marza

Laetitia Matignon

Olivier Simonin

Christian Wolf