TECHNICAL REPORT (4/1/2022-5/17/2023)

Project title: Control, Learning and Adaptation in Information-Constrained, Adversarial Environments

Institution: The University of Texas at Austin

Technical point of contact: Ufuk Topcu, utopcu@utexas.edu

Administrative point of contact: James Medina, james@oden.utexas.edu

1 Project description

We propose to develop a theoretical and algorithmic foundation that will help create autonomous robotic agents capable of executing patrol missions in urban environments, possibly in mixed teams of a set of autonomous robotic agents with heterogeneous sensing, perception, computation and actuation capabilities and a smaller number of soldiers (possibly in a supervisory role). To this end, we will formalize a range of problems—some of which are considered for the first time in the proposed effort—in the context of partial-information, stochastic games. While partial-information, stochastic games provide a highly expressive modeling language, synthesis of strategies in such games subject to temporal and logical constraints in their general form is known to be algorithmically impractical. Therefore, we plan to establish trade-offs between the expressivity of the problems and their algorithmic and computational tractability through a hierarchy of abstractions.

We partition the effort into three thrusts:

Thrust I – Synthesis in partial-information, stochastic games: We adopt partial-information, stochastic, two-player games played over finite graphs as the base model for the proposed effort. Additionally, we append this setting with temporal logic specifications in order to capture the constraints on the evolution of the plays in the game as well as of the knowledge of the patroller. Thrust I will develop approaches to suppress the computational complexity in synthesis with this modeling class and to extract strategies that balance the induced risk, ambiguity and randomization:

- Task I.1 – Strategy synthesis via belief set abstractions
- Task I.2 – Strategies with risk and ambiguity budgets
- Task I.3 – Strategies with partial and restricted randomization

Thrust II – Proactive strategies in adversarial environments: While Thrust I takes a passive approach by focusing on the synthesis of strategies that account for the limitations in prior knowledge and run-time information, Thrust II aims at proactively coping with these limitations at run time through *learning* and *active sensing*:

- Task II.1 – Safety-constrained learning in adversarial domains
- Task II.2 – Proactive sensing

Thrust III – Safeguarding against adversary's adaptation and deception: Thrust III will help establish an understanding of *cascading levels of reasoning* between the patroller and the adversary. The methods developed under Thrust III will account for the effects of such cascading levels of reasoning in the decisions of the patroller. The proposed tasks are though also receptive to the unsurmountable complexity of directly modeling such mutual adaptation into a stochastic, partial-information game setting. We will rather pursue two indirect and complementary approaches:

- Task III.1 – Suppressing the adversary's ability to infer

– Task III.2 – Discovering the adversary's deceptive tactics

2 Accomplishments

Task-Aware Verifiable RNN-Based Policies for Partially Observable

Partially observable Markov decision processes (POMDPs) are models for sequential decision-making under uncertainty and incomplete information. Machine learning methods typically train recurrent neural networks (RNN) as effective representations of POMDP policies that can efficiently process sequential data. However, it is hard to verify whether the POMDP driven by such RNN-based policies satisfies safety constraints, for instance, given by temporal logic specifications. We propose a novel method that combines techniques from machine learning with the field of formal methods: training an RNN-based policy and then automatically extracting a so-called finite-state controller (FSC) from the RNN. Such FSCs offer a convenient way to verify temporal logic constraints. Implemented on a POMDP, they induce a Markov chain, and probabilistic verification methods can efficiently check whether this induced Markov chain satisfies a temporal logic specification. Using such methods, if the Markov chain does not satisfy the specification, a byproduct of verification is diagnostic information about the states in the POMDP that are critical for the specification. The method exploits this diagnostic information to either adjust the complexity of the extracted FSC or improve the policy by performing focused retraining of the RNN. The method synthesizes policies that satisfy temporal logic specifications for POMDPs with up to millions of states, which are three orders of magnitude larger than comparable approaches.

Entropy Maximization for Partially Observable Markov Decision Processes

We study the problem of synthesizing a controller that maximizes the entropy of a partially observable Markov decision process (POMDP) subject to a constraint on the expected total reward. Such a controller minimizes the predictability of an agent's trajectories to an outside observer while guaranteeing the completion of a task expressed by a reward function. Focusing on finite-state controllers (FSCs) with deterministic memory transitions, we show that the maximum entropy of a POMDP is lower bounded by the maximum entropy of the parameteric Markov chain (pMC) induced by such FSCs. This relationship allows us to recast the entropy maximization problem as a so-called parameter synthesis problem for the induced pMC. We then present an algorithm to synthesize an FSC that locally maximizes the entropy of a POMDP over FSCs with the same number of memory states. In a numerical example, we highlight the benefit of using an entropy-maximizing FSC compared with an FSC that simply finds a feasible policy for accomplishing a task.

Identity Concealment Games: How I Learned to Stop Revealing and Love the Coincidences

In an adversarial environment, a hostile player performing a task may behave like a non-hostile one in order not to reveal its identity to an opponent. To model such a scenario, we define identity concealment games: zero-sum stochastic reachability games with a zero-sum objective of identity concealment. To measure the identity concealment of the player, we introduce the notion of an average player. The average player's policy represents the expected behavior of a non-hostile player. We show that there exists an equilibrium policy pair for every identity concealment game and give the optimality equations to synthesize an equilibrium policy pair. If the player's opponent follows a non-equilibrium policy, the player can hide its identity better. For this reason, we study how the hostile player may learn the opponent's policy. Since learning via exploration policies would quickly reveal the hostile player's identity to the opponent, we consider the problem of learning a near-optimal policy for the hostile player using the game runs collected under the average player's policy. Consequently, we propose an algorithm that provably learns a near-optimal policy and give an upper bound on the number of sample runs to be collected.

Probabilistic Control of Heterogeneous Swarms Subject to Graph Temporal Logic Specifications: A Decentralized and Scalable Approach

We develop a probabilistic control algorithm, GTLProCo, for swarms of agents with heterogeneous dynamics and objectives, subject to high-level task specifications. The resulting algorithm not only achieves decentralized control of the swarm but also significantly improves scalability over state-of-the-art existing algorithms. Specifically, we study a setting in which the agents move along the nodes of a graph, and the high-level task specifications for the swarm are expressed in a recently-proposed language called graph temporal logic (GTL). By constraining the distribution of the swarm over the nodes of the graph, GTL can specify a wide range of properties, including safety, progress, and response. GTLProCo, with a computational complexity agnostic to the number of agents comprising the swarm, controls the density distribution of the swarm in a decentralized and probabilistic manner. To this end, it synthesizes a time-varying Markov chain modeling the time evolution of the density distribution under the GTL constraints. We first identify a subset of GTL, namely reach-avoid specifications, for which we can reduce the synthesis of such a Markov chain to either linear or semi-definite programs. Then, in the general case, we formulate the synthesis of the Markov chain as a mixed-integer nonlinear program (MINLP).We exploit the structure of the problem to provide an efficient sequential mixed-integer linear programming scheme with trust regions to solve the MINLP. We empirically demonstrate that our sequential scheme is at least three orders of magnitude faster than off-the-shelf MINLP solvers and illustrate the effectiveness of GTLProCo in several swarm scenarios.

Reward Machines for Cooperative Multi-Agent Reinforcement Learning

In cooperative multi-agent reinforcement learning, a collection of agents learns to interact in a shared environment to achieve a common goal. We propose the use of reward machines (RM) – Mealy machines used as structured representations of reward functions – to encode the team's task. The proposed novel interpretation of RMs in the multi-agent setting explicitly encodes required teammate interdependencies, allowing the team-level task to be decomposed into sub-tasks for individual agents. We define such a notion of RM decomposition and present algorithmically verifiable conditions guaranteeing that distributed completion of the sub-tasks leads to team behavior accomplishing the original task. This framework for task decomposition provides a natural approach to decentralized learning: agents may learn to accomplish their sub-tasks while observing only their local state and abstracted representations of their teammates. We accordingly propose a decentralized q-learning algorithm. Furthermore, in the case of undiscounted rewards, we use local value functions to derive lower and upper bounds for the global value function corresponding to the team task. Experimental results in three discrete settings exemplify the effectiveness of the proposed RM decomposition approach, which converges to a successful team policy an order of magnitude faster than a centralized learner and significantly outperforms hierarchical and independent q-learning approaches.

Smooth Convex Optimization Using Sub-Zeroth-Order Oracles

We consider the problem of minimizing a smooth, Lipschitz, convex function over a compact, convex set using sub-zeroth-order oracles: an oracle that outputs the sign of the directional derivative for a given point and a given direction, an oracle that compares the function values for a given pair of points, and an oracle that outputs a noisy function value for a given point. We show that the sample complexity of optimization using these oracles is polynomial in the relevant parameters. The optimization algorithm that we provide for the comparator oracle is the first algorithm with a known rate of convergence that is polynomial in the number of dimensions. We also give an algorithm for the noisy-value oracle that incurs sublinear regret in the number of queries and polynomial regret in the number of dimensions

Decentralized Online Influence Maximization

We consider the problem of finding the maximally influential node in random networks where each node influences every other node with constant yet unknown probability. We develop an online algorithm that learns the relative influences of the nodes. It relaxes the assumption in the existing literature that a central observer can monitor the influence spread globally. The proposed algorithm delegates the online updates to the nodes on the network; hence requires only local observations at the nodes. We show that using an explore-then-commit learning strategy, the cumulative regret accumulated by the algorithm over horizon T approaches O(T2/3) for a network with a large number of nodes. Additionally, we show that, for fixed T , the worst case-regret grows linearly with the number n of nodes in the graph. Numerical experiments illustrate this linear dependence for Chung-Lu models. The experiments also demonstrate that -greedy learning strategies can achieve similar performance to the explore-then-commit strategy on Chung-Lu models.

Exploiting Partial Observability for Optimal Deception

Deception is a useful tool in situations where an agent operates in the presence of its adversaries. We consider a setting where a supervisor provides a reference policy to an agent, expects the agent to operate in an environment by following the reference policy, and partially observes the agent's behavior. The agent instead follows a different, deceptive policy to achieve a different task. We model the environment with a Markov decision process and study the synthesis of optimal deceptive policies under partial observability. We formalize the notion of deception as a hypothesis testing problem and show that the synthesis of optimal deceptive policies is NP-hard. As an approximation, we consider the class of mixture policies, which provides a convex optimization formulation of the deception problem. We give an algorithm that converges to the optimal mixture policy. We also consider a special class of Markov decision processes where the transition and observation functions are deterministic. For this case, we give a randomized algorithm for path planning that generates a path for the agent in polynomial time and achieves the optimal value for the considered objective function.

Memoryless Adversaries in Imperfect Information Games

Given an agent with limited sensing capabilities, we analyze whether it is possible to deploy a new agent in the operational space of the preexisting agent in a safe manner. One approach for modeling the interaction of the introduced agent with its environment, which contains the preexisting agent, is through a two-player game of im- perfect information. However, the computational cost of solving this game is prohibitive. Restricting the preexisting agent's strategy to just memoryless strategies and assuming that the introduced agent has perfect information alleviates the computational cost while still modeling realistic environments. The proposed algorithm for solv- ing the game finds a winning strategy for the introduced agent by solving a quantified Boolean formula (QBF) for the game. We justify this approach by establishing a matching PSPACE lower bound. We also show that this result holds even when the preexisting agent uses bounded history to condition its play.

Convex Optimization for Parameter Synthesis in MDPs

Probabilistic model-checking aims to prove whether a Markov decision process (MDP) satisfies a temporal logic specification. The underlying methods rely on an often unrealistic assumption that the MDP is precisely known. Consequently, parametric MDPs (pMDPs) extend MDPs with transition probabilities that are functions over unspecified parameters. The parameter synthesis problem is to compute an instantiation of these unspecified parameters such that the resulting MDP satisfies the temporal logic specification. We formulate the parameter synthesis problem as a quadratically constrained quadratic program, which is nonconvex and is NP-hard to solve in general. We develop two approaches that iteratively obtain locally optimal solutions. The first approach exploits the so-called convex–concave procedure (CCP), and the second approach utilizes a sequential convex programming (SCP) method. The techniques improve the runtime and scalability by multiple orders of magnitude compared to black-box CCP and SCP by merging ideas from convex optimization and probabilistic model-checking. We demonstrate the approaches on a satellite collision avoidance problem with hundreds of thousands of states and tens of thousands of parameters and their scalability on a wide range of commonly used benchmarks.

Online Learning with Implicit Exploration in Episodic Markov Decision Processes

A wide range of applications require autonomous agents that are capable of learning an a priori unknown task. Additionally, an autonomous agent may be put in the same environment multiple times, each time having to learn a different task. Motivated by these applications, we study the problem of learning an a priori and evolving task in an online manner. In particular, we consider an agent whose behavior is modeled by an episodic Markov decision process. The agent's task, captured by a loss function, is unknown to the agent and, furthermore, may change in an adversarial manner from episode to episode. However, in each episode, the agent receives a bandit feedback corresponding to the loss function at that episode every time it takes an action. Given a limited budget of T episodes, the objective is to learn a policy with minimum regret with respect to the best policy in hindsight. We propose a policy search algorithm that employs online mirror descent using an optimistically biased estimator of the loss function. We prove that the proposed algorithm achieves both on expectation and with high probability a sublinear regret of O (L T— S —— A —), where L is the length of each episode, — S — is the number of states, and — A — is the number of actions.

Multiple Plans are Better than One: Diverse Stochastic Planning

In planning problems, it is often challenging to fully model the desired specifications. In particular, in human-robot interaction, such difficulty may arise due to human's preferences that are either private or complex to model. Consequently, the resulting objective function can only partially capture the specifications and optimizing that may lead to poor performance with respect to the true specifications. Motivated by this challenge, we formulate a problem, called diverse stochastic planning, that aims to generate a set of representative — small and diverse — behaviors that are near-optimal with respect to the known objective. In particular, the problem aims to compute a set of diverse and near-optimal policies for systems modeled by a Markov decision process. We cast the problem as a constrained nonlinear optimization for which we propose a solution relying on the Frank-Wolfe method. We then prove that the proposed solution converges to a stationary point and demonstrate its efficacy in several planning problems.

Robust Policy Synthesis for Uncertain POMDPs via Convex Optimization

We study the problem of policy synthesis for uncertain partially observable Markov decision processes (uPOMDPs). The transition probability function of uPOMDPs is only known to belong to a so-called uncertainty set, for instance in the form of probability intervals. Such a model arises when, for example, an agent operates under information limitation due to imperfect knowledge about the accuracy of its sensors. The goal is to compute a policy for the agent that is robust against all possible probability distributions within the uncertainty set. In particular, we are interested in a policy that robustly ensures the satisfaction of temporal logic and expected reward specifications. We state the underlying optimization problem as a semi-infinite quadratically-constrained quadratic program (QCQP), which has finitely many variables and infinitely many constraints. Since QCQPs are non-convex in general and practically infeasible to solve, we resort to the so-called convex-concave procedure to convexify the QCQP. Even though convex, the resulting optimization problem still has infinitely many constraints and is NP-hard. For uncertainty sets that form convex polytopes, we provide a transformation of the problem to a convex QCQP with finitely many constraints. We demonstrate the feasibility of our approach by means of several case studies that highlight typical bottlenecks for our problem. In particular, we show that we are able to solve benchmarks with hundreds of thousands of states, hundreds of different observations, and we investigate the effect of different levels of uncertainty in the models.

Blending Controllers via Multi-Objective Bandits

Safety and performance are often two competing objectives in sequential decision-making problems. Existing performant controllers, such as controllers derived from reinforcement learning algorithms, often fall short of safety guarantees. On the contrary, controllers that guarantee safety, such as those derived from classical control theory, require restrictive assumptions and are often conservative in performance. Our goal is to blend a performant and a safe controller to generate a single controller that is safer than the performant and accumulates higher rewards than the safe controller. To this end, we propose a blending algorithm using the framework of contextual multi-armed multi-objective bandits. At each stage, the algorithm observes the environment's current context alongside an immediate reward and cost, which is the underlying safety measure. The algorithm then decides which controller to employ based on its observations. We demonstrate that the algorithm achieves sublinear Pareto regret, a performance measure that models coherence with an expert that always avoids picking the controller with both inferior safety and performance. We derive an upper bound on the loss in individual objectives, which imposes no additional computational complexity. We empirically demonstrate the algorithm's success in blending a safe and a performant controller in a safety-focused testbed, the Safety Gym environment. A statistical analysis of the blended controller's total reward and cost reflects two key takeaways: The blended controller shows a strict improvement in performance compared to the safe controller, and it is safer than the performant controller.

3 Outcomes

1. Carr, Steven, Nils Jansen, and Ufuk Topcu. ”Task-aware verifiable RNN-based policies for partially observable Markov decision processes.” Journal of Artificial Intelligence Research 72 (2021): 819-847.

2. Savas, Yagiz, Michael Hibbard, Bo Wu, Takashi Tanaka, and Ufuk Topcu. ”Entropy Maximization for Partially Observable Markov Decision Processes.” arXiv preprint arXiv:2105.07490 (2021).

3. Karabag, Mustafa O., Melkior Ornik, and Ufuk Topcu. ”Identity Concealment Games: How I Learned to Stop Revealing and Love the Coincidences.” arXiv preprint arXiv:2105.05377 (2021).

4. Djeumou, Franck, Zhe Xu, Murat Cubuktepe, and Ufuk Topcu. ”Probabilistic control of heterogeneous swarms subject to graph temporal logic specifications: A decentralized and scalable approach.” IEEE Transactions on Automatic Control (2022).

5. Neary, Cyrus, Zhe Xu, Bo Wu, and Ufuk Topcu. ”Reward machines for cooperative multi-agent reinforcement learning.” arXiv preprint arXiv:2007.01962 (2020).

6. Karabag, Mustafa O., Cyrus Neary, and Ufuk Topcu. ”Smooth Convex Optimization using Sub-Zeroth-Order Oracles.” In Proceedings of the AAAI Conference on Artificial Intelligence, vol. 35, no. 5, pp. 3815-3822. 2021.

7. Bayiz, Yigit E., and Ufuk Topcu. ”Decentralized Online Influence Maximization.” In 2022 58th Annual Allerton Conference on Communication, Control, and Computing (Allerton), pp. 1-8. IEEE, 2022.

8. Karabag, Mustafa O., Melkior Ornik, and Ufuk Topcu. ”Exploiting Partial Observability for Optimal Deception.” IEEE Transactions on Automatic Control (2022).

9. Dhananjay Raju, Georgios Bakirtzis, and Ufuk Topcu. 2023. Memoryless Adversaries in Imperfect Information Games: Extended Abstract. In Proc. of the 22nd International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2023), London, United Kingdom, May 29 – June 2, 2023, IFAAMAS, 3 pages.

10. Cubuktepe, Murat, Nils Jansen, Sebastian Junges, Joost-Pieter Katoen, and Ufuk Topcu. ”Convex Optimization for Parameter Synthesis in MDPs.” IEEE Transactions on Automatic Control (2021).

11. Ghasemi, Mahsa, Abolfazl Hashemi, Haris Vikalo, and Ufuk Topcu. ”Online learning with implicit exploration in episodic Markov decision processes.” In 2021 American Control Conference (ACC), pp. 1953-1958. IEEE, 2021.

12. Ghasemi, Mahsa, Evan Scope Crafts, Bo Zhao, and Ufuk Topcu. ”Multiple Plans are Better than One: Diverse Stochastic Planning.” In Proceedings of the International Conference on Automated Planning and Scheduling, vol. 31, pp. 140-148. 2021.

13. Suilen, Marnix, Nils Jansen, Murat Cubuktepe, and Ufuk Topcu. ”Robust policy synthesis for uncertain POMDPs via convex optimization.” arXiv preprint arXiv:2001.08174 (2020).

14. Gohari, Parham, Franck Djeumou, Abraham P. Vinod, and Ufuk Topcu. ”Blending controllers via multi-objective bandits.” arXiv preprint arXiv:2007.15755 (2020).

4 Conclusions and Lessons

The project has made significant progress in all three directions covered in its objectives. It has achieved 3-to-4-orders of magnitude improvement in scalability in synthesis of policies in partial-information, stochastic environments with limited modeling knowledge. It has developed new methods for learning in uncertain, dynamic environments and improved their data efficiency and generalization by incorporating contextual knowledge. It also contributed to the synthesis of policies that leak minimal critical information to their environment and actively aim to mislead potentially adversarial observers.

The results, put together, point to the fact that algorithmic capabilities for perception and decision-making by autonomous systems are maturing to a level at which it is meaningful to think about how these systems can become strategic agents that manipulate information flows in their interactions with the world around them and create an advantage for their users.

The attached slide deck, presented to the program manager, summarizes the activities and progress in this reporting period.

Control, Learning and Adaptation in Information-Constrained, Adversarial Environments: Updates

Steven Carr

aUTonomous
SYSTEMS GROUP

Outline

Summary of Observations from DSPT-3

Recap

Classification Highlights

Network Descriptions using Graphs

Observations from DSPT-3

Blue Force Command

- Many things happening at once that the blue force commander needs to track
- Was not relying not the system to classify but rather kept their own inferences based on individual responses
- When the system displayed a large change in belief, it was hard to give attribution for how it occurred.

Observations from DSPT-3

Blue Force Command

- Many things happening at once that the blue force commander needs to track
- Was not relying not the system to classify but rather kept their own inferences based on individual responses
- When the system displayed a large change in belief, it was hard to give attribution for how it occurred.

Network Graph Impact

- System operated on the inference that "alike actors congregate together"
- Harder to classify those actors that did not interact with others (Lone Wolf)
- Network effects were misleading: Two police officers (green) arresting a threat caused the system to misclassify the threat (red) as a non-threat (green)

Recap - What is New?

Where we were?

- Sandbox environment that had multiple categories of actors
- Operator can activate a series of triggers to tell actors apart
- The fastest way to converge is to select highly disruptive events
- However, events may change the behavior of the agents — which makes it harder to classify them

Recap - What is New?

Where we were?

- Sandbox environment that had multiple categories of actors
- Operator can activate a series of triggers to tell actors apart
- The fastest way to converge is to select highly disruptive events
- However, events may change the behavior of the agents — which makes it harder to classify them

What's New? (Start from slide 11)

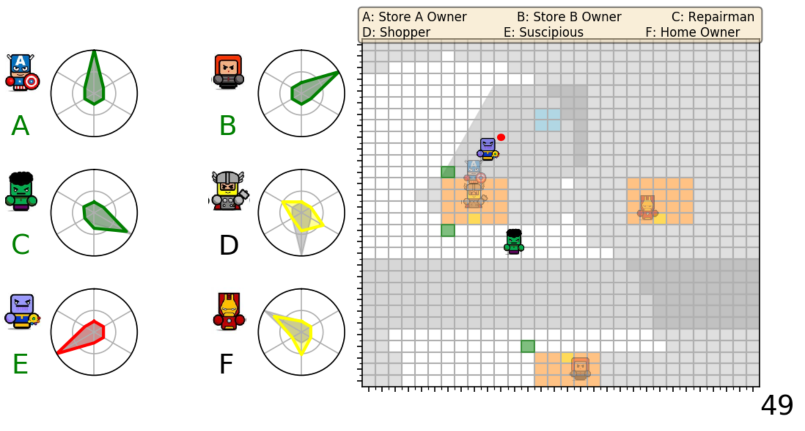
- Provide information for the operator about the most consequential decisions that lead to the system's classification
- Networking graphs to better describe actors behavior relative to each other

Recap - Problem Outline

Goal: Classify each agent in the environment as one of six agent types

Assumptions on each agent

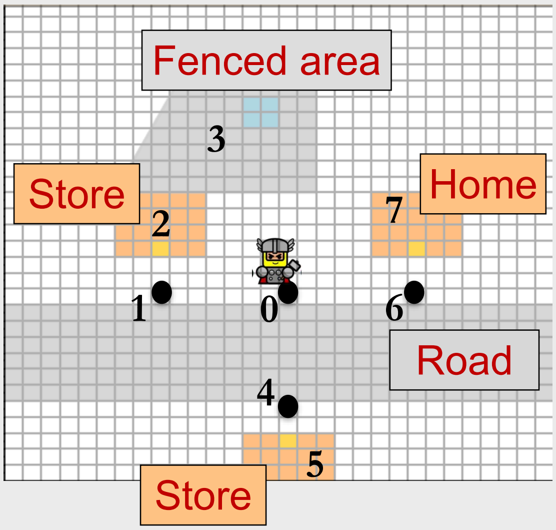
- One of the six types (can be multiples of the same one)
- Fixed agent type
- They operate independently
- Fixed movement models, which are known *a priori*

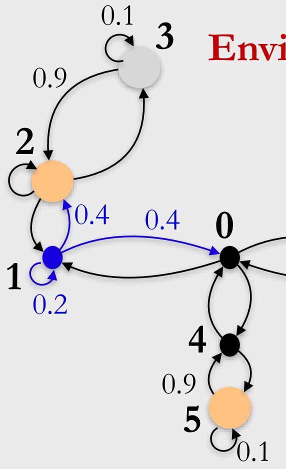


Assumptions on the system

- Fixed number of agents
- The agent types are distinguishable

Recap - Operating Environment



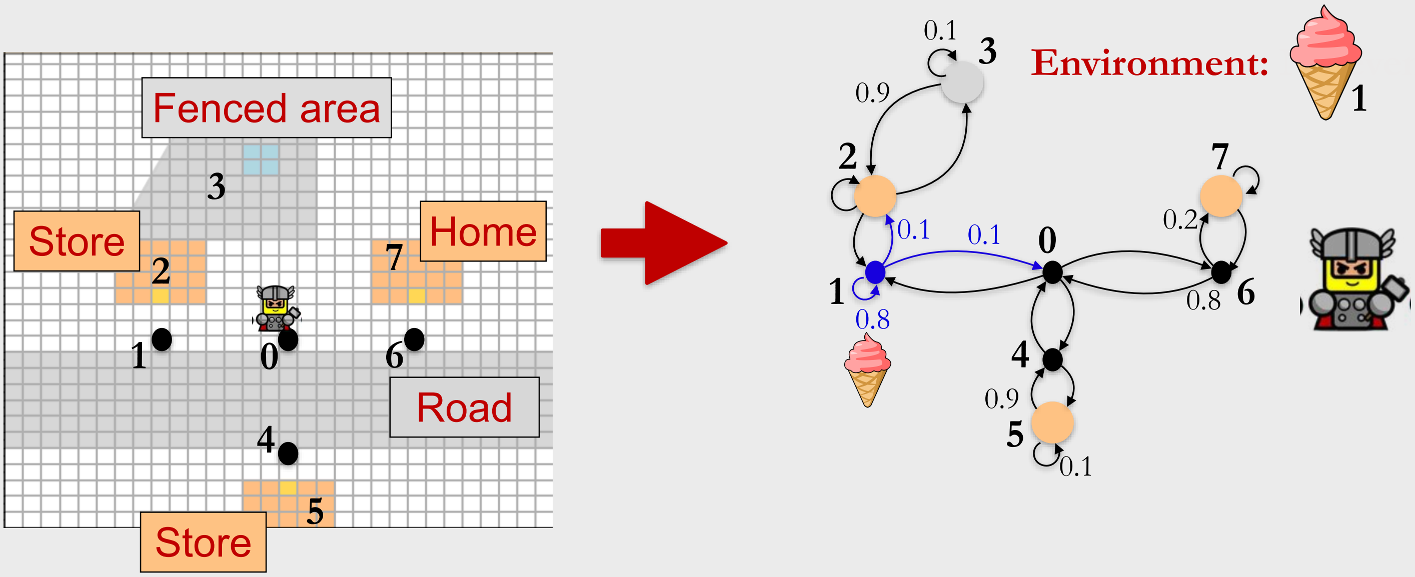


Environment: No event

Each time-step, agents move stochastically between numbers depending on what is happening in the environment

Model the agents in the environment using a **Markov decision process (MDP)**

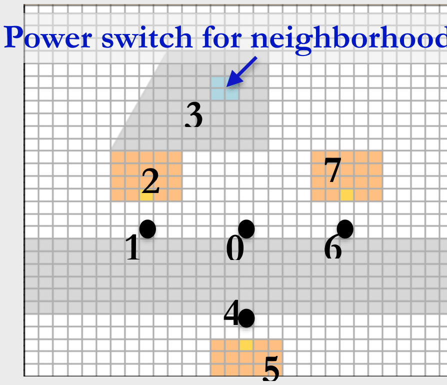
Recap - Operating Environment



Each time-step, agents move stochastically between numbers depending on what is happening in the environment

Model the agents in the environment using a **Markov decision process (MDP)**

Recap - Agent Types and Behaviors

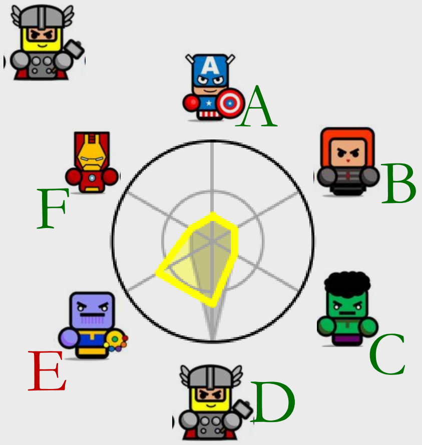


6 different types	Corresponding behavior
A - Store Owner at 2 (NW)	Tends to stay at 2
B - Store Owner at 5 (S)	Tends to stay at 5
C - Repairman	Goes between fenced area (3) and 5
D - Shopper	Mostly goes between 2 and 5
E - Suspicious	Copies shopper then takes advantage of event to access neighborhood switch at 3
F - Home Owner at 7 (NE)	Tends to stay at 7 but will go to both 2 and 5

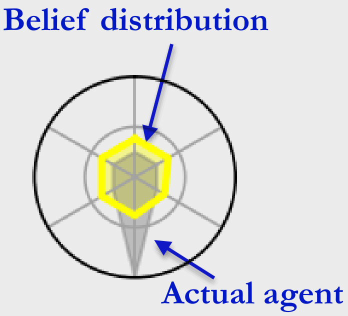
Recap - Agent Types and Behaviors

Track the belief of each agent's type using a wheel plot

Distance to the edge indicates the confidence a given agent corresponds to a type



Unsure if D or E



Belief distribution

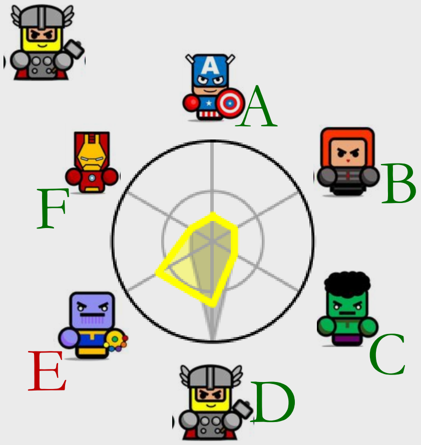
Actual agent

Uniform belief

Recap - Agent Types and Behaviors

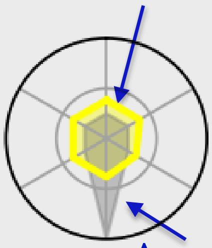
Track the belief of each agent's type using a wheel plot

Distance to the edge indicates the confidence a given agent corresponds to a type



Unsure if D or E

Belief distribution

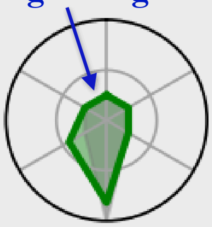


Actual agent

Uniform belief

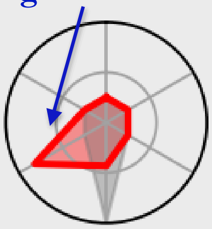
Indicator thresholds

Green for confident agent is good



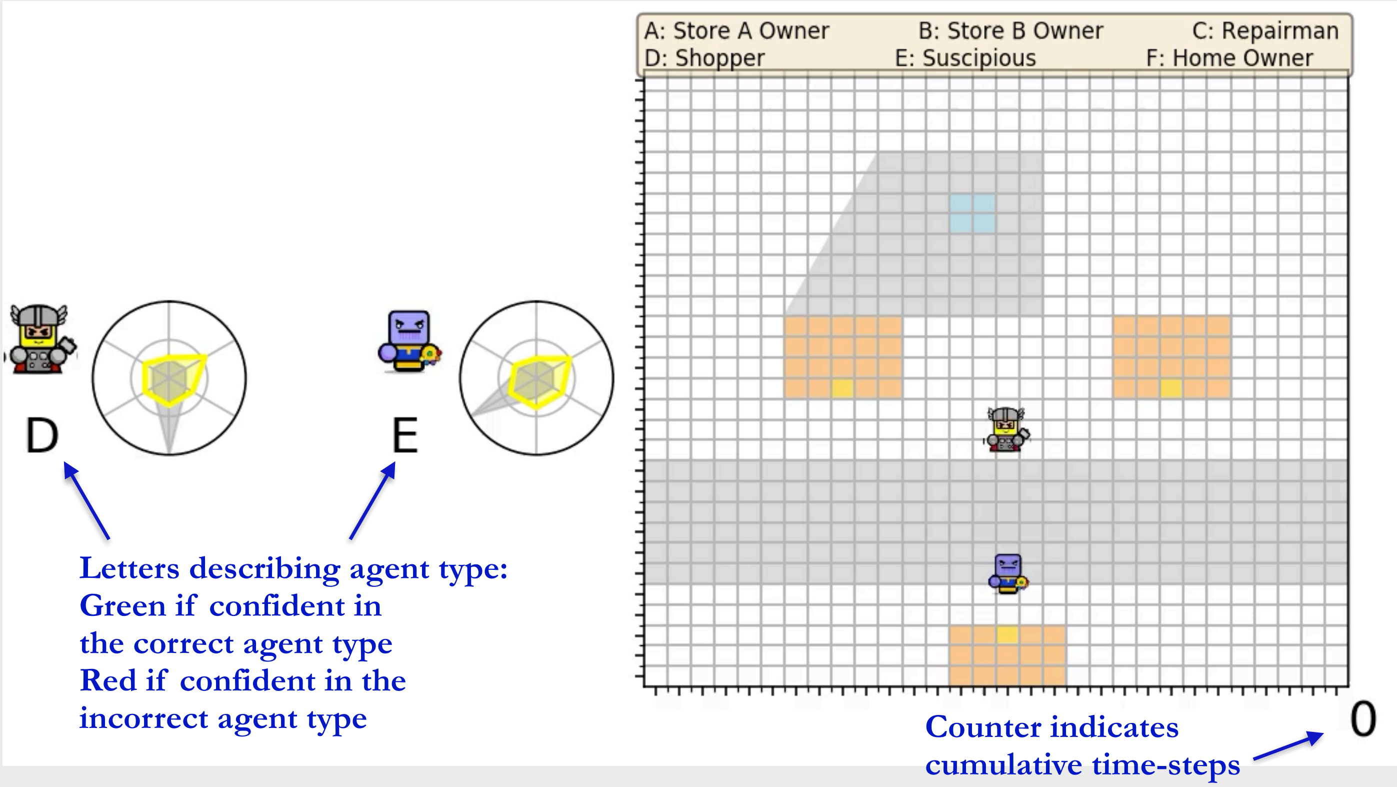
75% chance of D

Red for confident agent is threat



75% chance of E (threat)

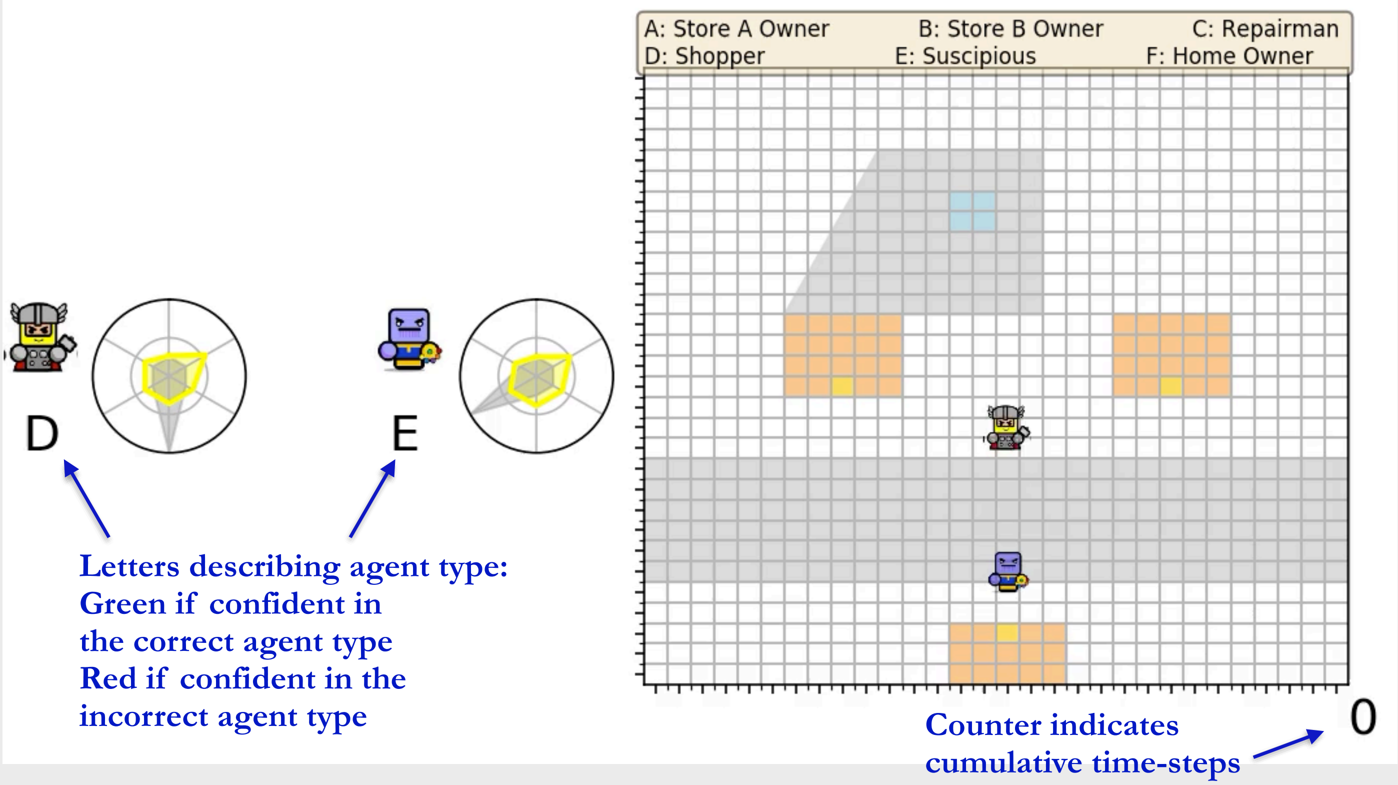
Recap - Agent Types and Behaviors



Letters describing agent type:
Green if confident in
the correct agent type
Red if confident in the
incorrect agent type

Counter indicates
cumulative time-steps

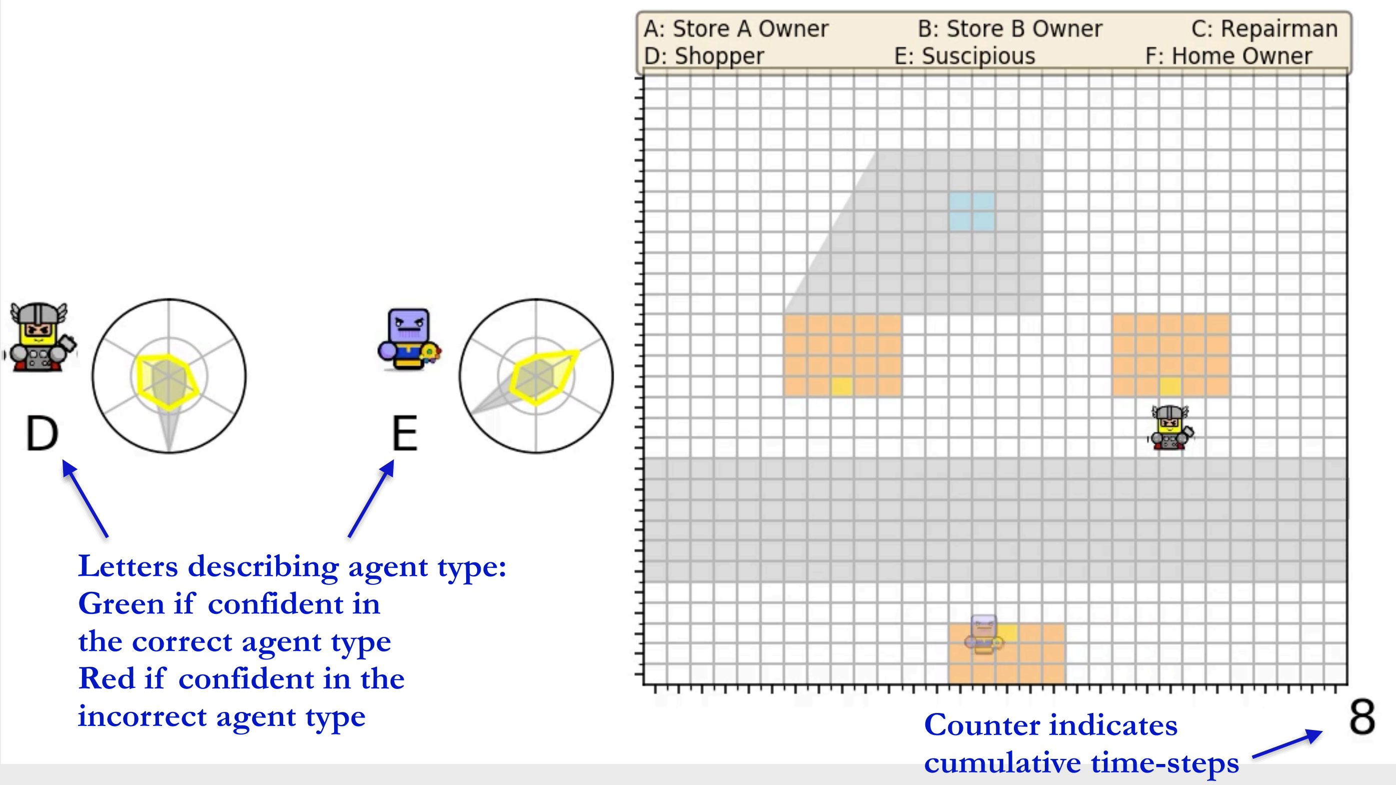
Recap - Agent Types and Behaviors



Letters describing agent type:
Green if confident in
the correct agent type
Red if confident in the
incorrect agent type

Counter indicates
cumulative time-steps

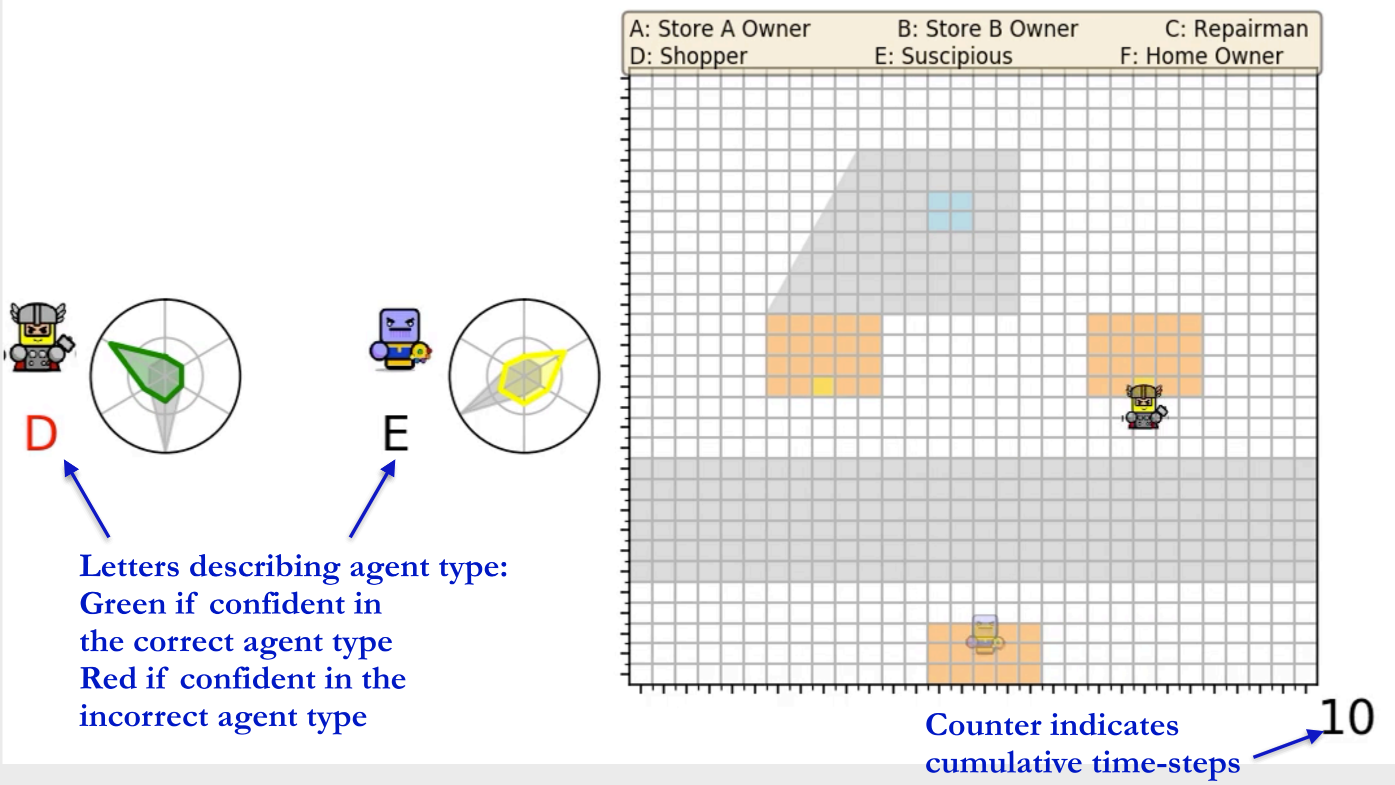
Recap - Agent Types and Behaviors



Letters describing agent type:
Green if confident in
the correct agent type
Red if confident in the
incorrect agent type

Counter indicates
cumulative time-steps

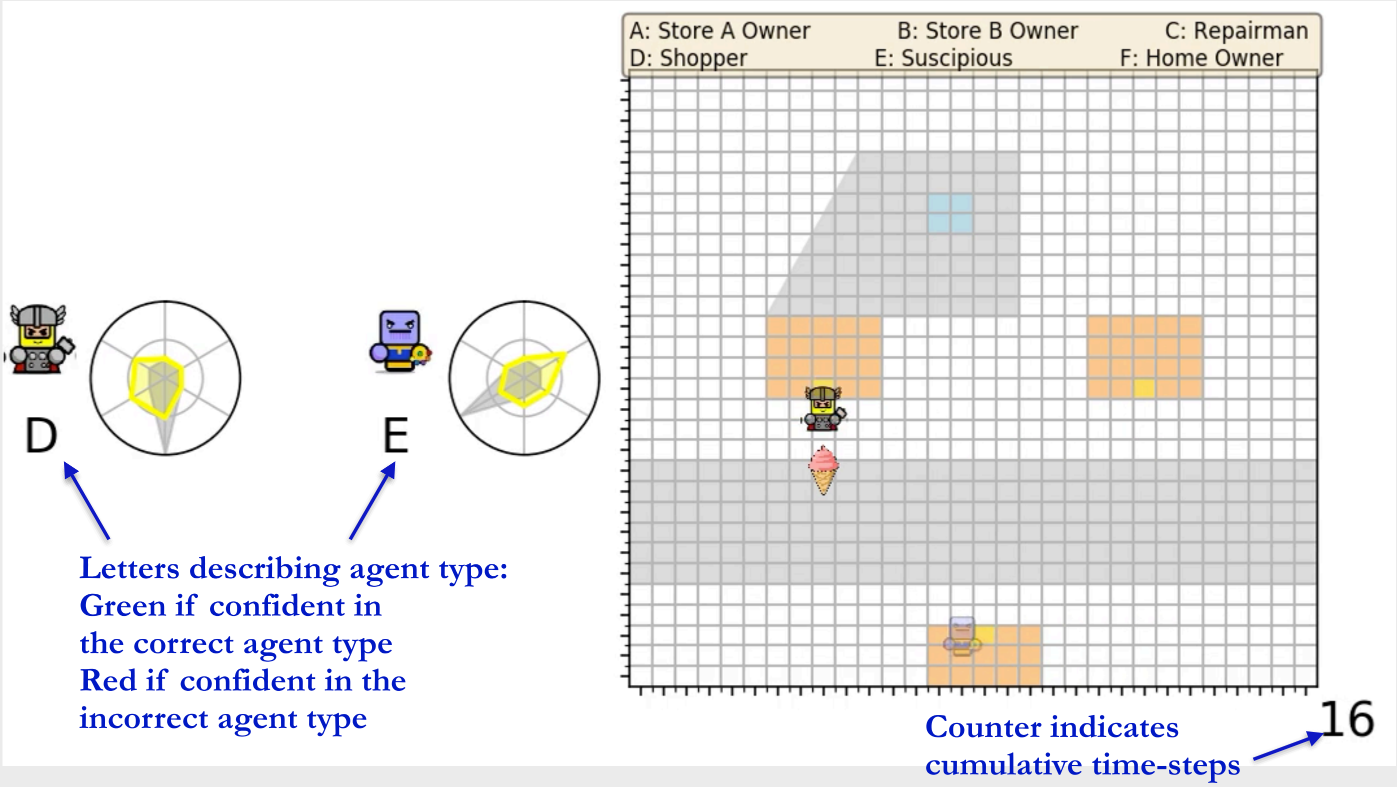
Recap - Agent Types and Behaviors



Letters describing agent type:
Green if confident in
the correct agent type
Red if confident in the
incorrect agent type

Counter indicates
cumulative time-steps

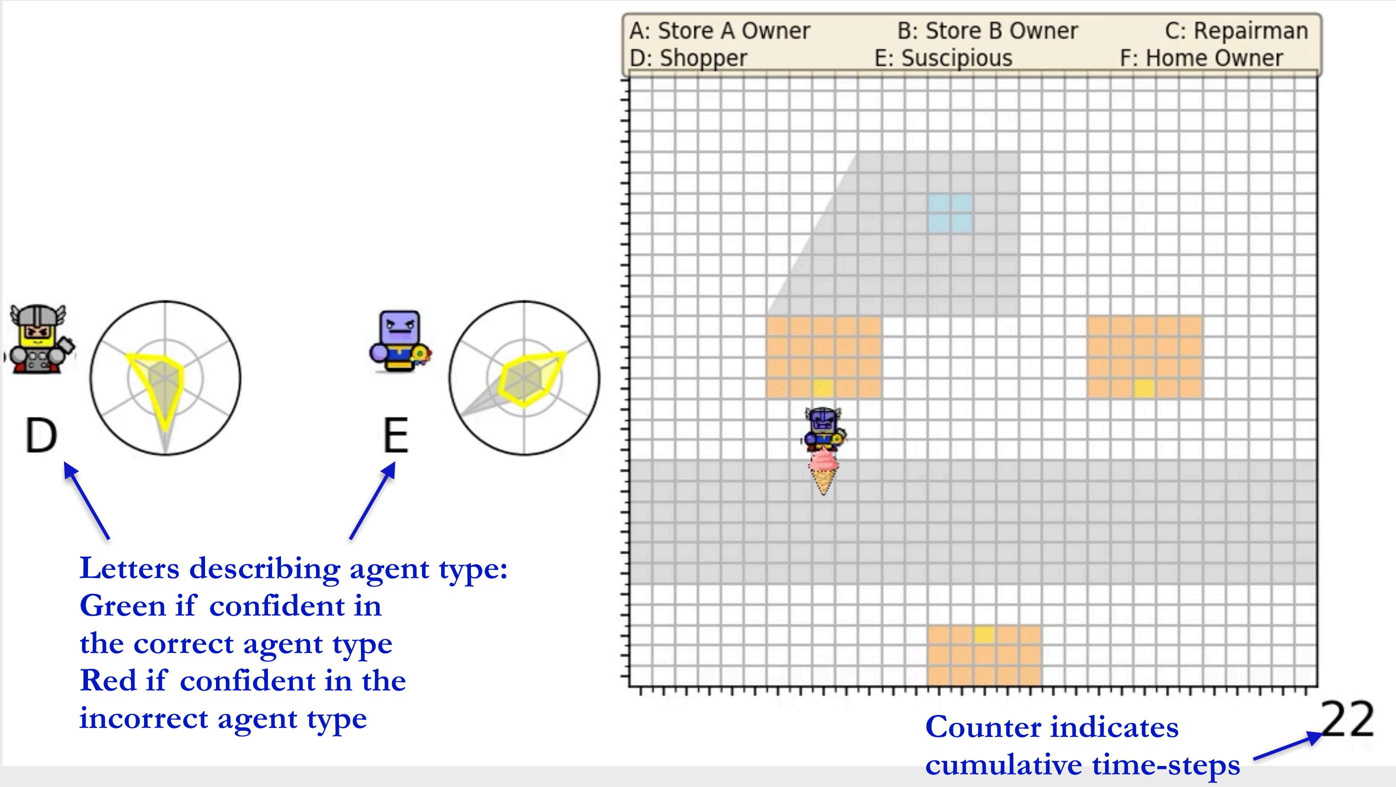
Recap - Agent Types and Behaviors



Letters describing agent type:
Green if confident in
the correct agent type
Red if confident in the
incorrect agent type

Counter indicates
cumulative time-steps

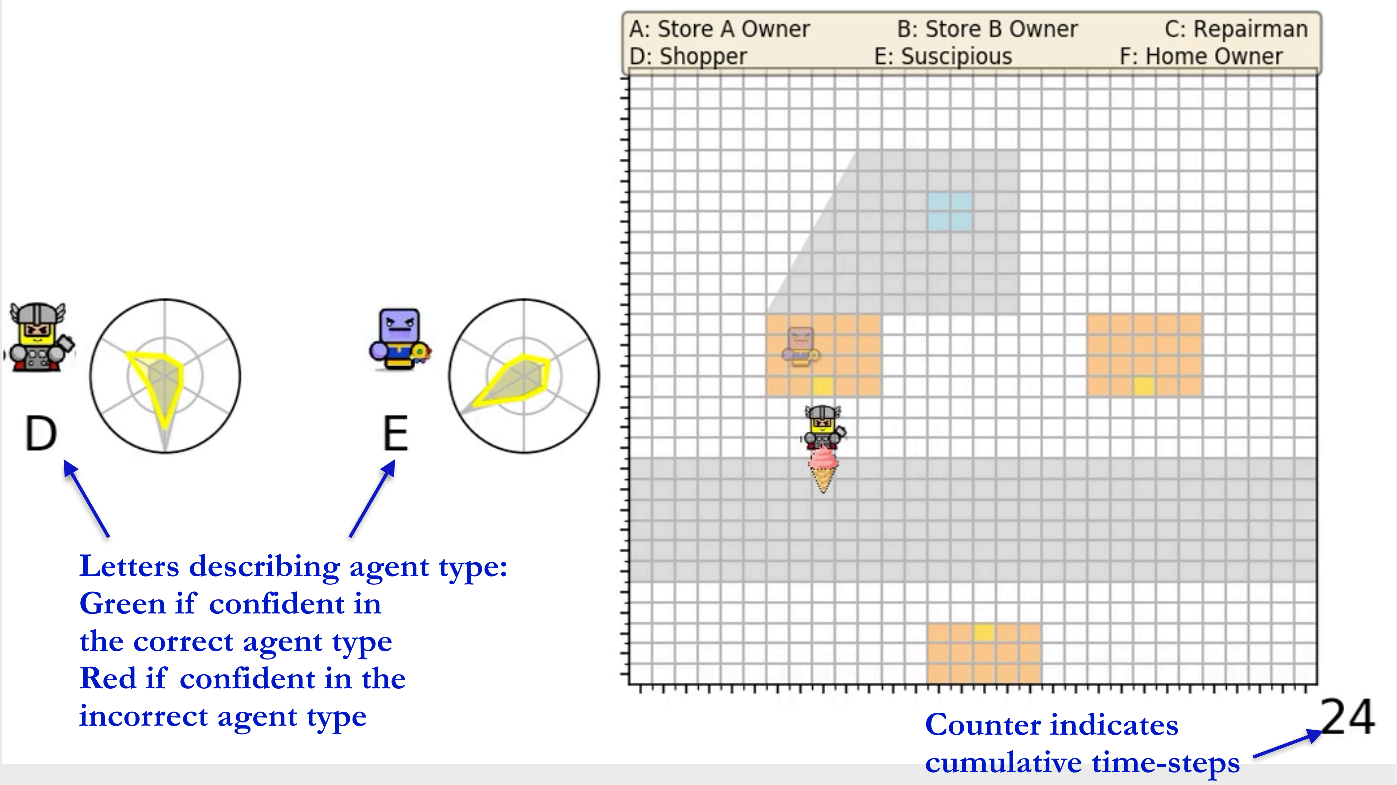
Recap - Agent Types and Behaviors



Letters describing agent type:
Green if confident in
the correct agent type
Red if confident in the
incorrect agent type

Counter indicates
cumulative time-steps

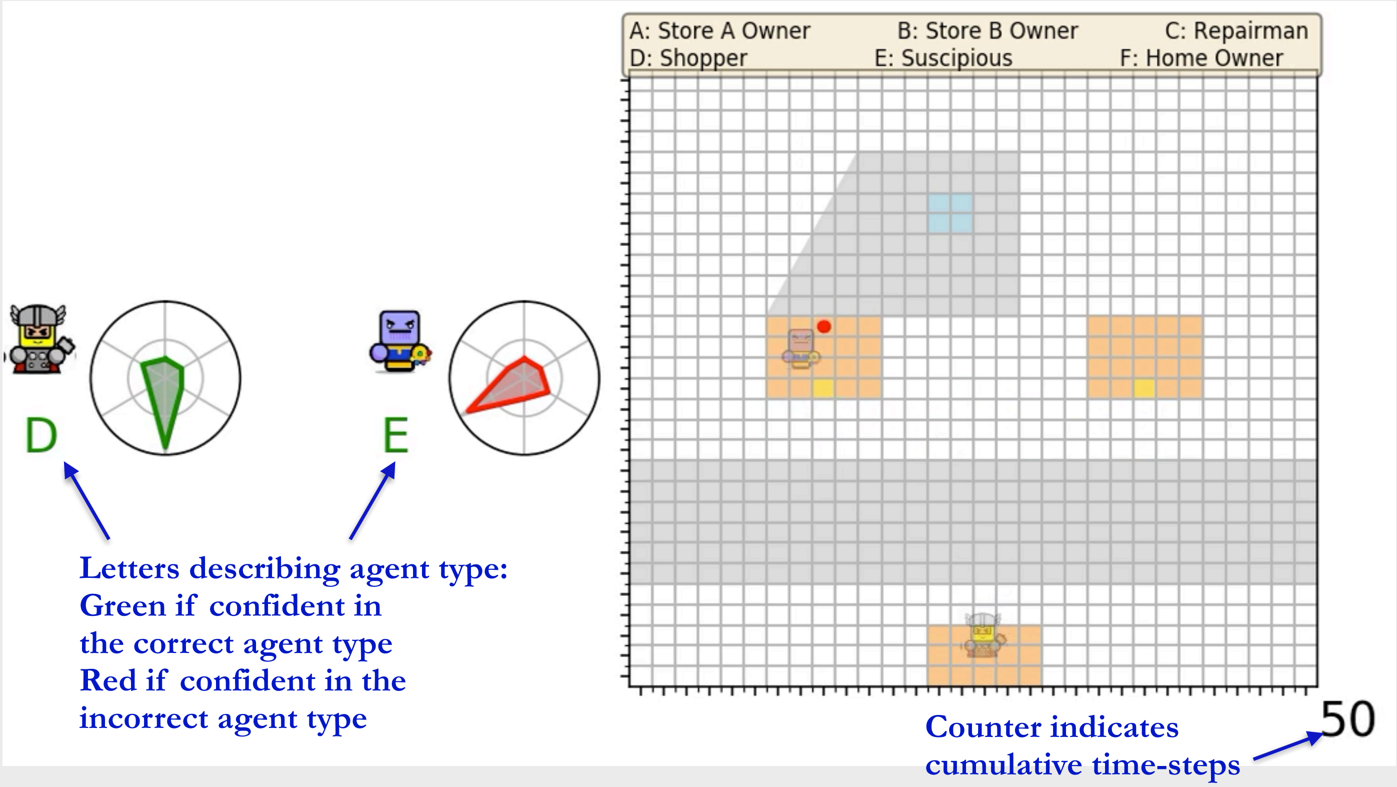
Recap - Agent Types and Behaviors



Letters describing agent type:
Green if confident in
the correct agent type
Red if confident in the
incorrect agent type

Counter indicates
cumulative time-steps

Recap - Agent Types and Behaviors



Letters describing agent type:
Green if confident in
the correct agent type
Red if confident in the
incorrect agent type

Counter indicates
cumulative time-steps

Recap: Belief Updates

Track the agent's movement and compare the likelihoods for each agent type

States **Agent type** θ

$$l(s_{t+1} \mid s_t, a_t, \theta)$$

Event trigger a_t

Recap: Belief Updates

Track the agent's movement and compare the likelihoods for each agent type

States Agent type θ

$$l(s_{t+1} \,|\, s_t, a_t, \theta)$$

Event trigger a_t

Use the sequence of likelihoods to form a **belief**, i.e. the probability that a given agent is of agent type $\theta \in \{A, B, C, D, E, F\}$

Likelihood update Previous belief

Belief update

$$b_{t+1}(\theta) = \frac{1}{c} l(s_{t+1} \,|\, s_t, a_t, \theta) b_t(\theta)$$

Normalization constant to ensure beliefs sum to 1

Recap: Belief Updates

Track the agent's movement and compare the likelihoods for each agent type

States Agent type θ

$$l(s_{t+1} \,|\, s_t, a_t, \theta)$$

Event trigger a_t

Use the sequence of likelihoods to form a **belief**, i.e. the probability that a given agent is of agent type $\theta \in \{A, B, C, D, E, F\}$

Likelihood update Previous belief

Belief update $$b_{t+1}(\theta) = \frac{1}{c} l(s_{t+1} \,|\, s_t, a_t, \theta) b_t(\theta)$$

Normalization constant to ensure beliefs sum to 1

Belief analysis operates independently for each agent

Highlight Reel

What are the most impactful moments for that lead the system to its conclusion?

How much does belief change with each decision?

Belief delta

$$\Delta b_{t+1}(\theta) = \max(|b_{t+1}(\theta) - b_t(\theta)|)$$

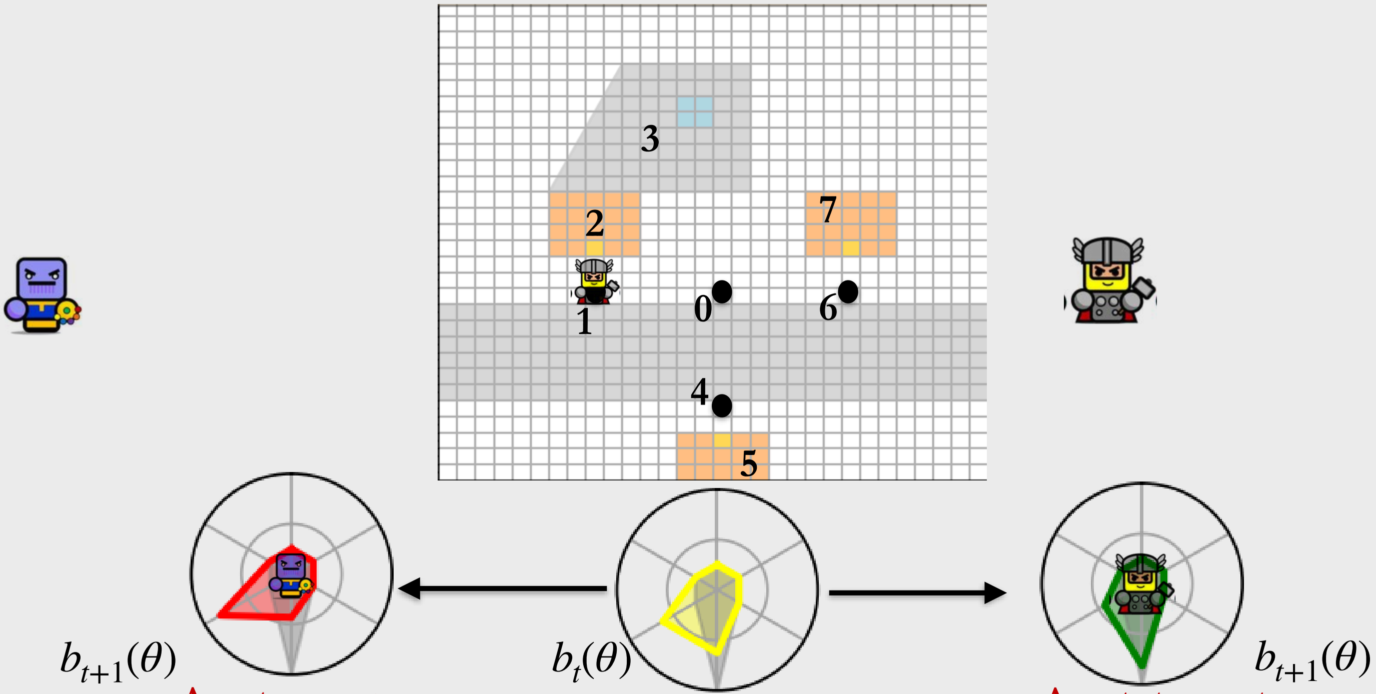
Current belief **Previous belief**

Metric for threat/
non-threat condition

Find the decision point t from run T with the highest change in belief

$$\underset{t\in T}{\operatorname{argmax}}\left(\Delta b_t(\theta)\right)$$

Expected Decisions



$b_{t+1}(\theta)$

Agent goes up

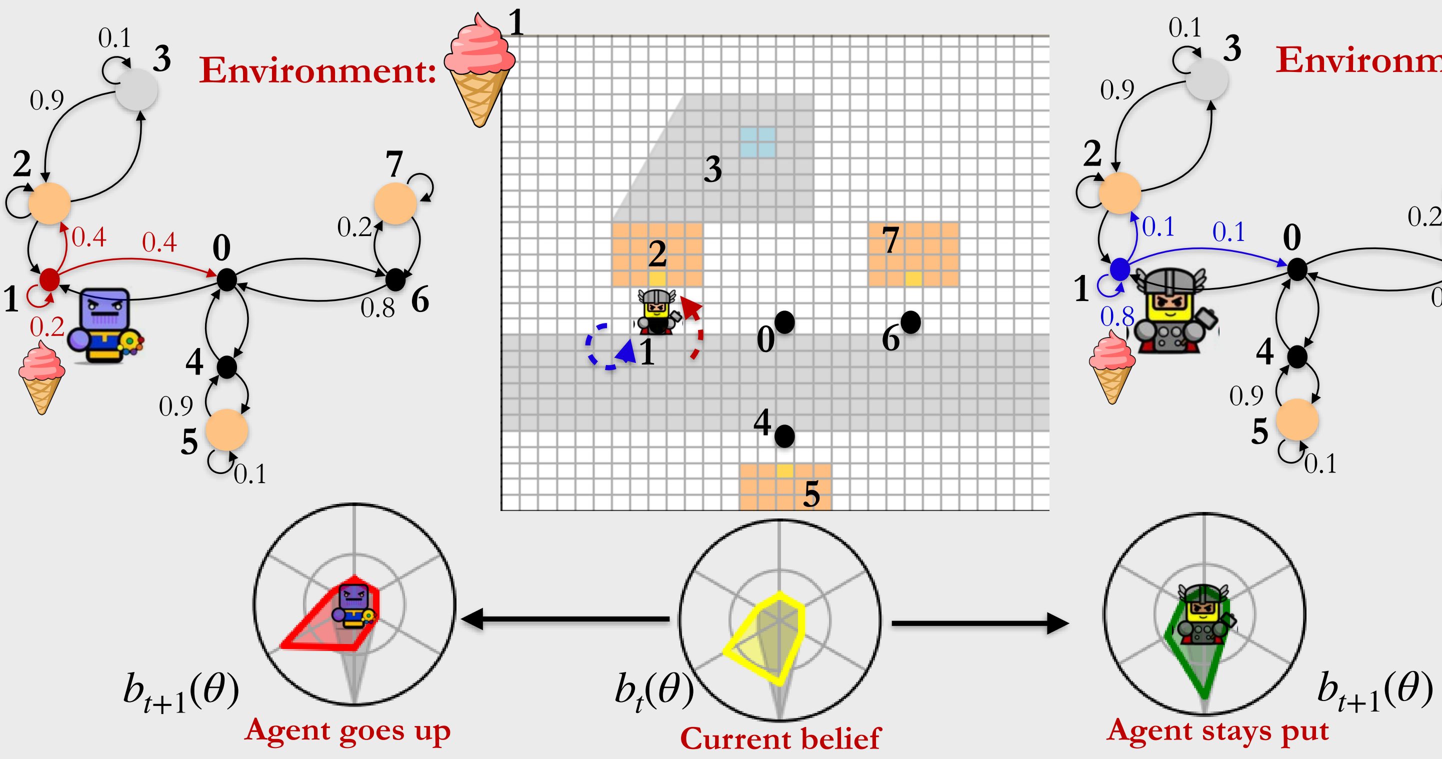
$b_t(\theta)$

Current belief

$b_{t+1}(\theta)$

Agent stays put

Expected Decisions



Environment:

Environment:

$b_{t+1}(\theta)$

Agent goes up

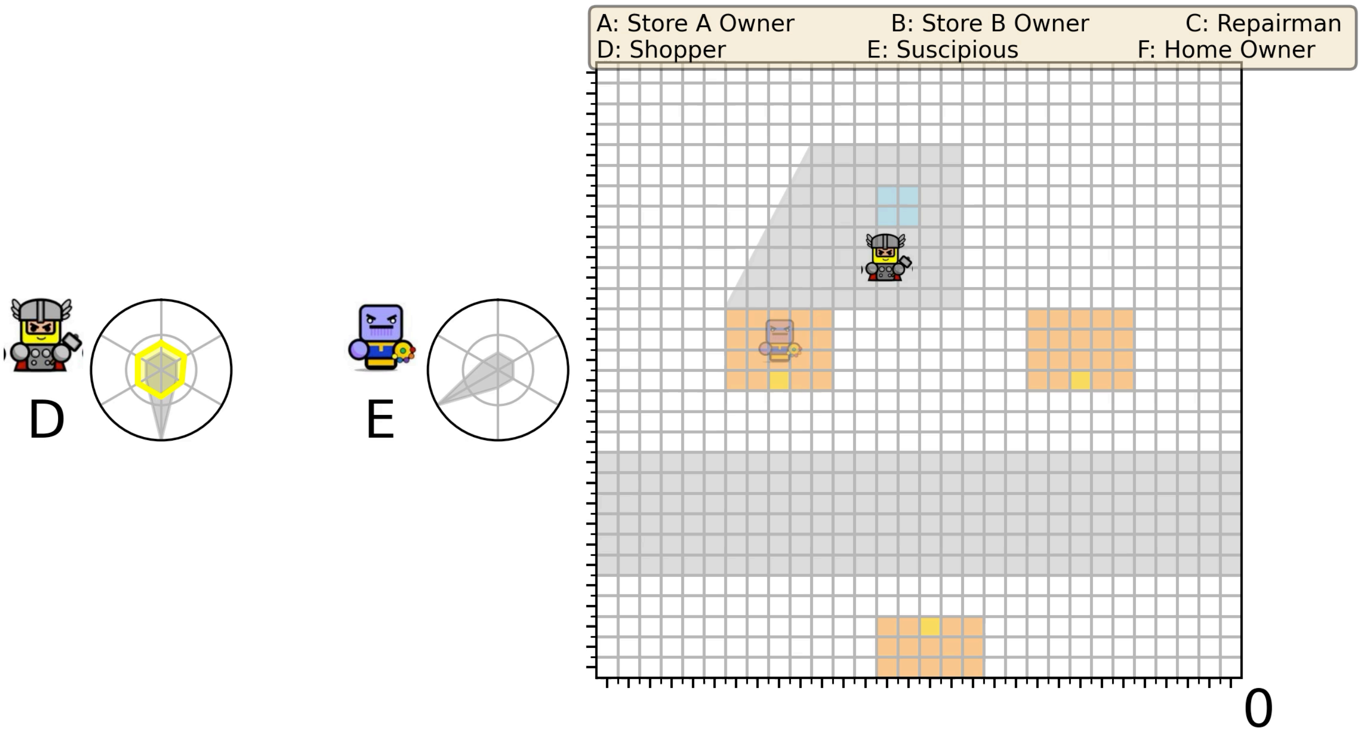
$b_t(\theta)$

Current belief

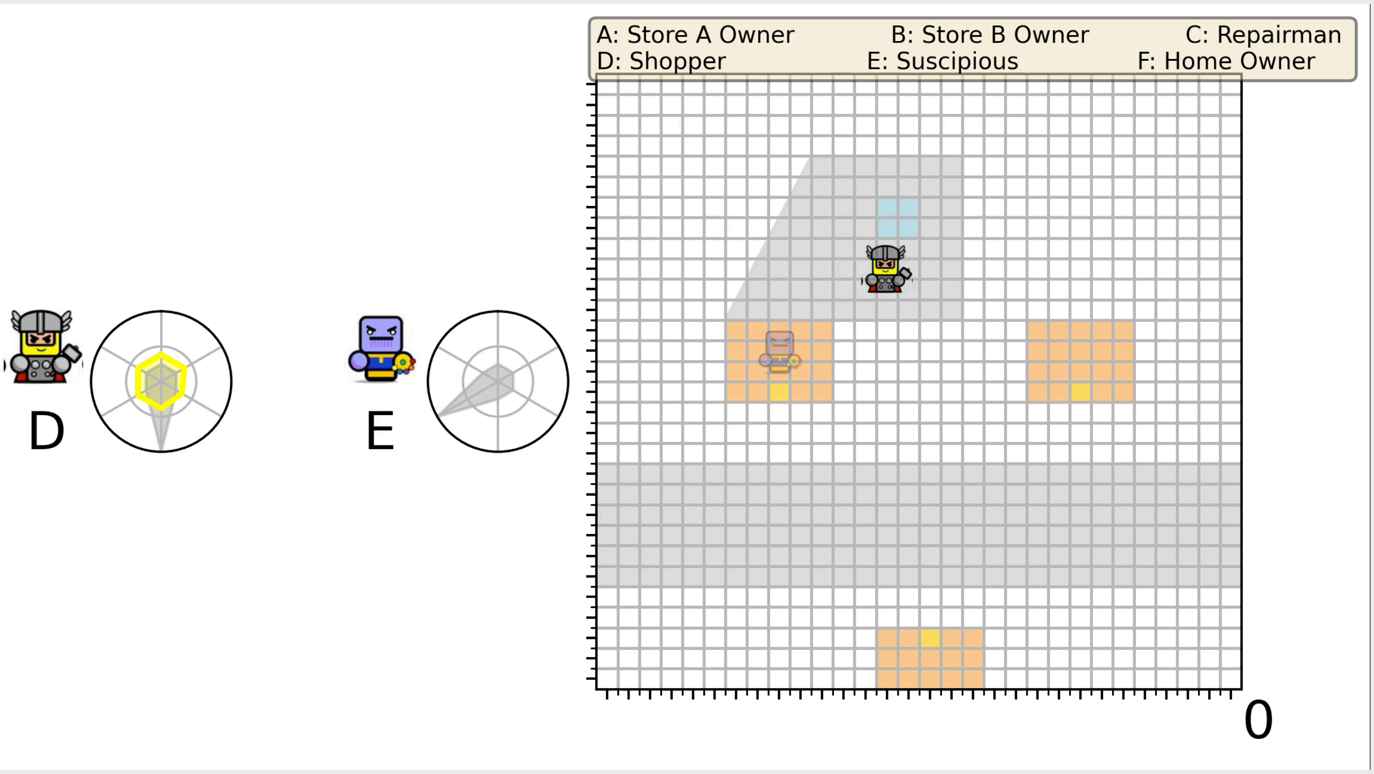
$b_{t+1}(\theta)$

Agent stays put

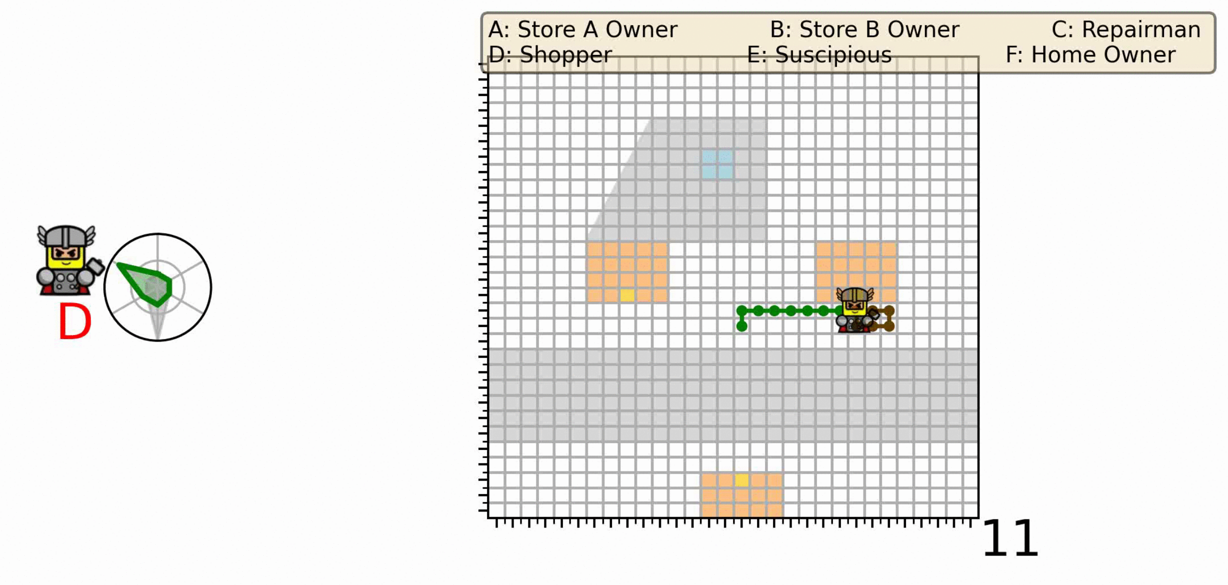
Scenario 1


A: Store A Owner
B: Store B Owner
C: Repairman
D: Shopper
E: Suscipious
F: Home Owner
D
E
0

Scenario 1

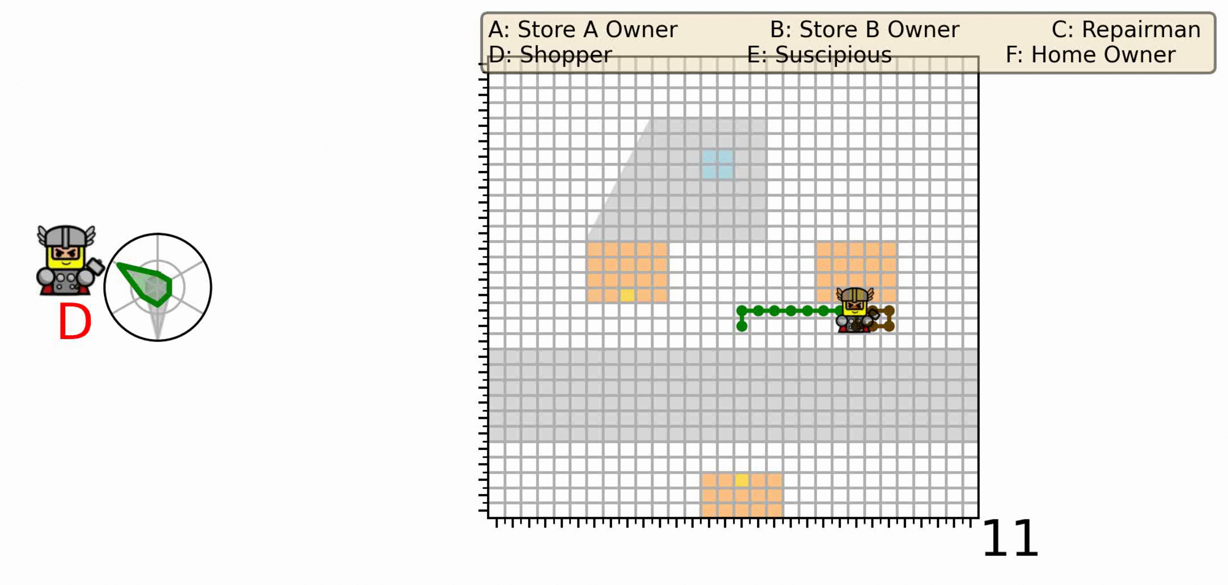

A: Store A Owner
B: Store B Owner
C: Repairman
D: Shopper
E: Suscipious
F: Home Owner
D
E
0

Highlights from Scenario 1



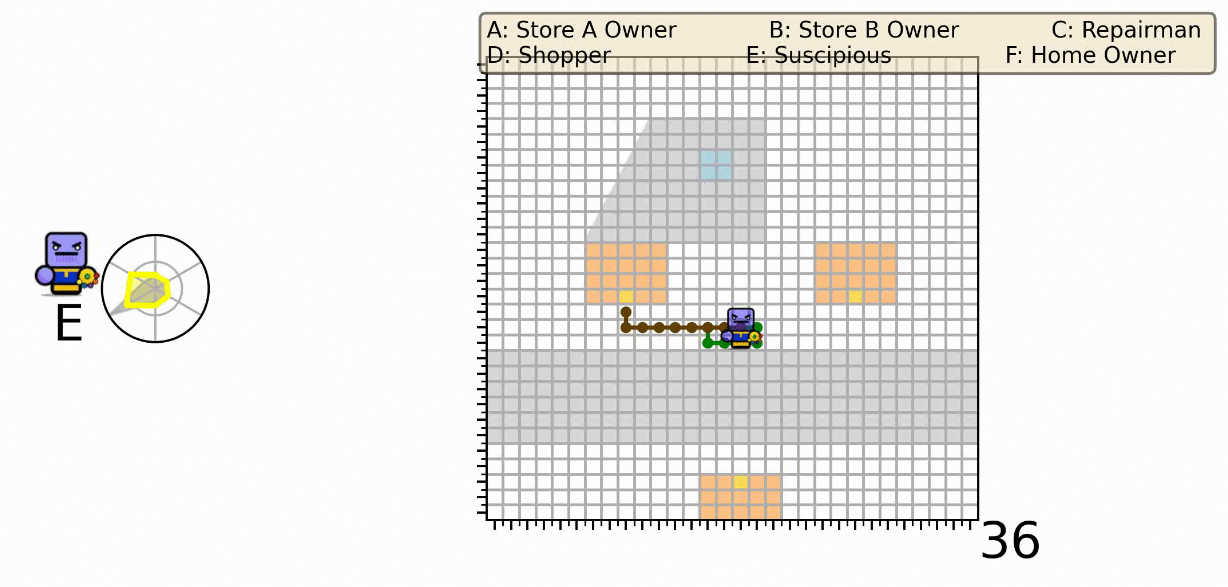
$$\Delta b_{t+1}(\theta) = 0.517$$

Highlights from Scenario 1



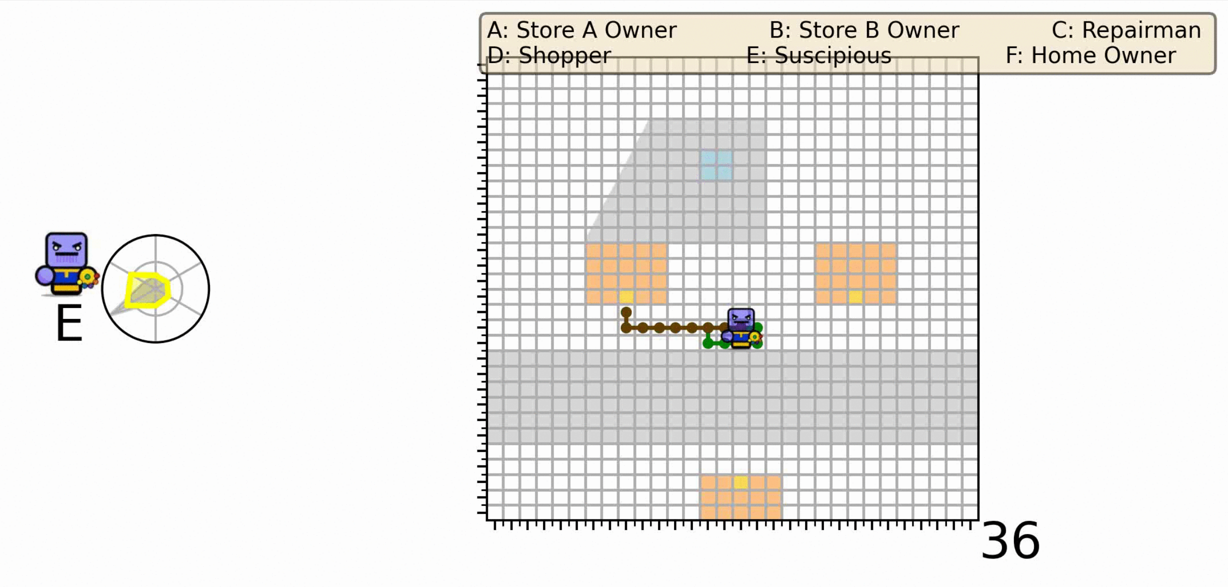
$$\Delta b_{t+1}(\theta) = 0.517$$

Highlights from Scenario 1



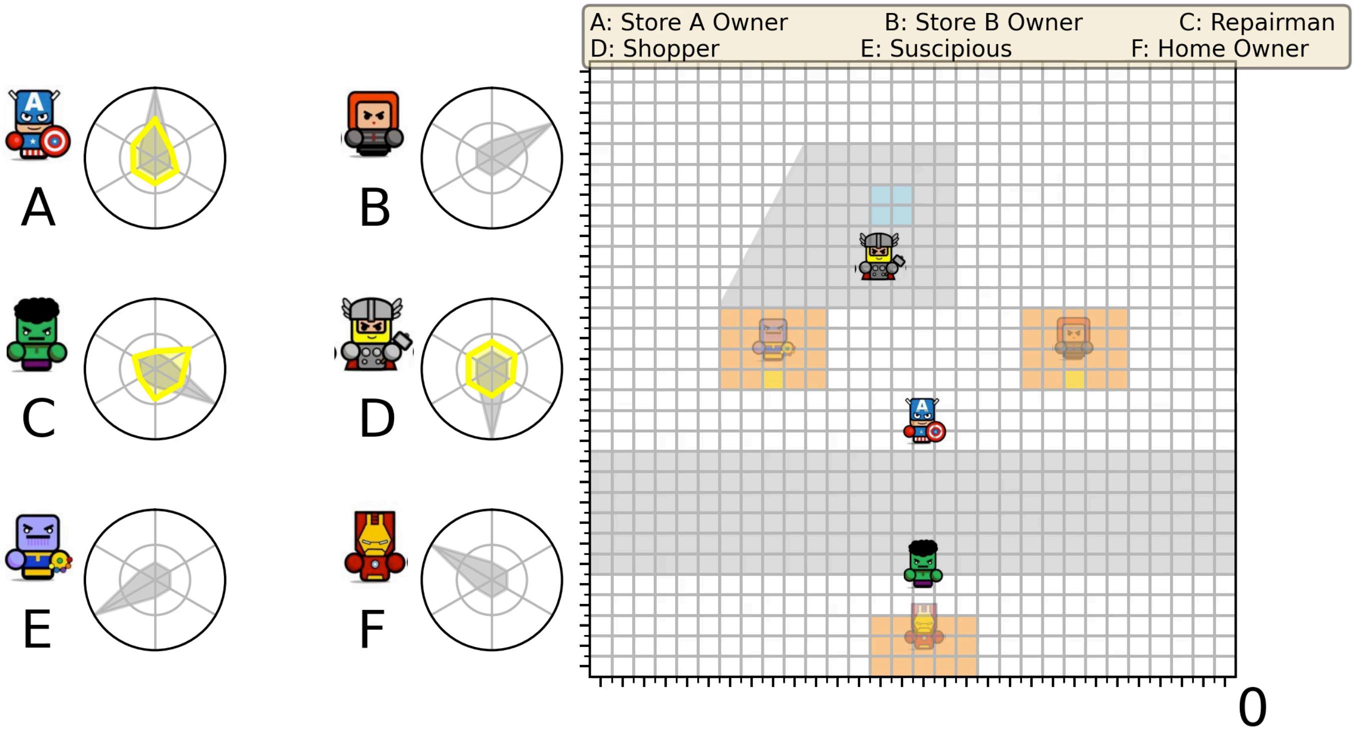
$$\Delta b_{t+1}(\theta) = 0.505$$

Highlights from Scenario 1

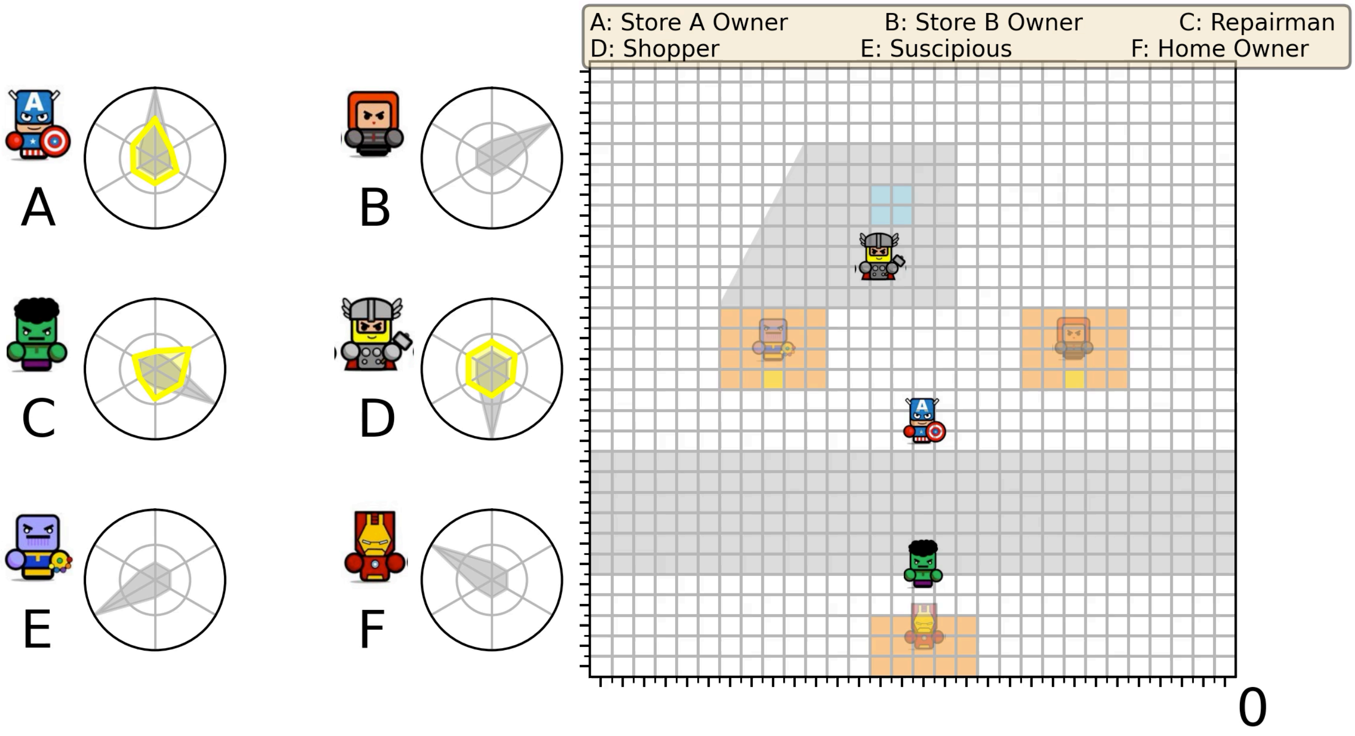


$$\Delta b_{t+1}(\theta) = 0.505$$

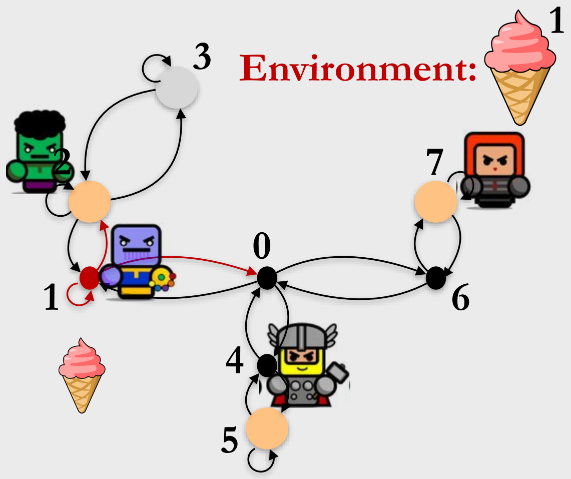
Simulation 2 - Network Demonstration

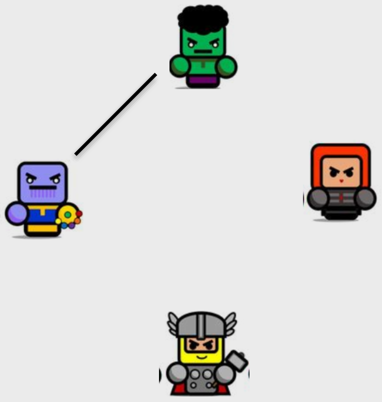


Simulation 2 - Network Demonstration



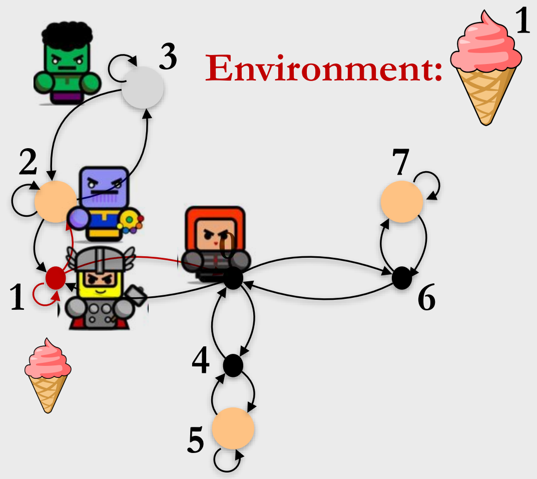
Network Graph Construction

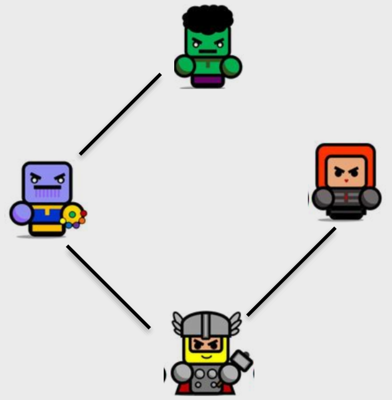




Is agent within one successor state of current location? Yes — add to neighbors

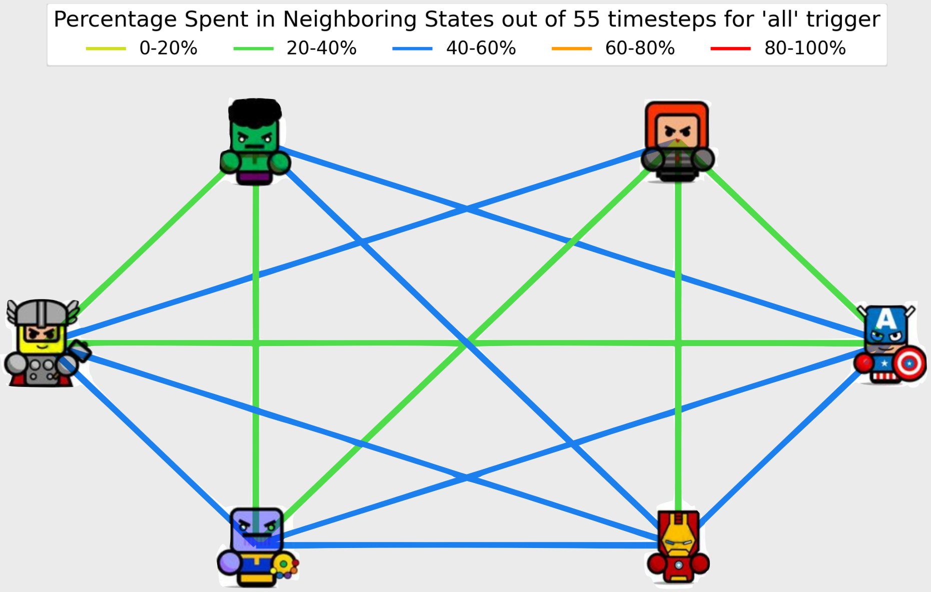
Network Graph Construction





Is agent within one successor state of current location? Yes — add to neighbors

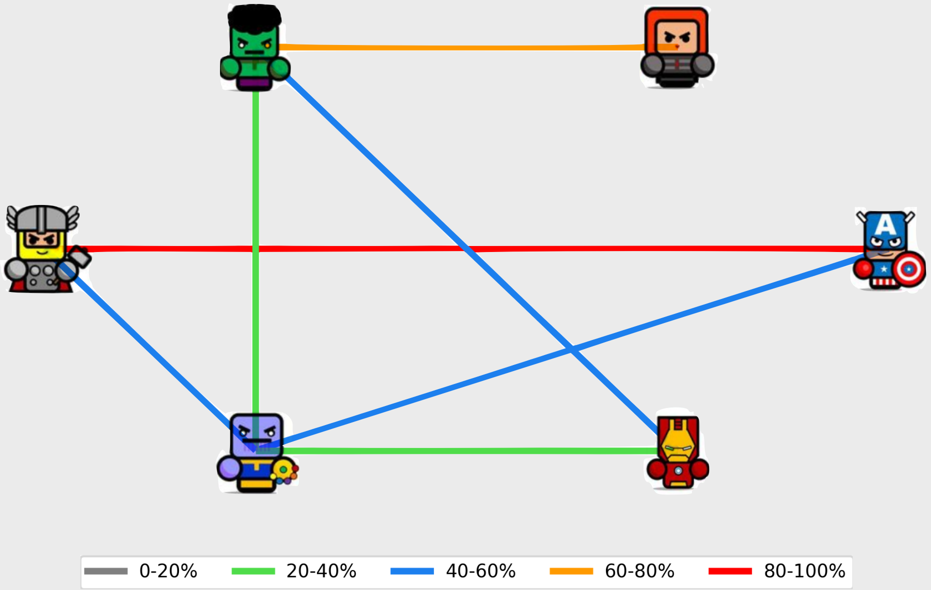
Network Graph History


Percentage Spent in Neighboring States out of 55 timesteps for 'all' trigger
0-20%
20-40%
40-60%
60-80%
80-100%

Remove rare connections
< 20%

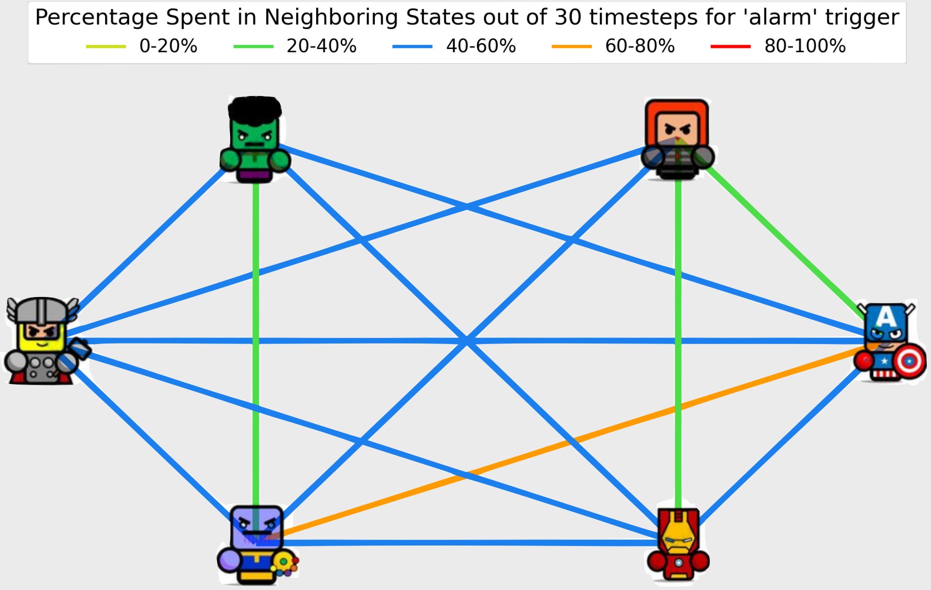
Graph for the entire run

Network Graph History



Graph for when the "nominal" condition is active

Network Graph History


Percentage Spent in Neighboring States out of 30 timesteps for 'alarm' trigger
0-20%
20-40%
40-60%
60-80%
80-100%

Graph for when alarm is active

Next Steps

Feed network information into belief update rule

Incorporate agents entering/leaving the scene

Account for expected information (update belief even when we don’t directly observe an agent)

REPORT DOCUMENTATION PAGE	*Form Approved* *OMB No. 0704-0188*

1. REPORT DATE *(DD-MM-YYYY)* 03/14/2023	**2. REPORT TYPE** Final Technical Report	**3. DATES COVERED** *(From - To)* 04/1/2022-05/17/2023

4. TITLE AND SUBTITLE Control, Learning and Adaptation in Information-Constrained, Adversarial Envi-ronments	**5a. CONTRACT NUMBER** **5b. GRANT NUMBER** D19AP00004-04 **5c. PROGRAM ELEMENT NUMBER**
6. AUTHOR(S) Bayiz, Yigit E.; Carr, Steven; Crafts, Evan Scope; Cubuktepe, Murat; Djeumou, Franck; Ghasemi, Mahsa; Hashemi, Abolfazl; Hibbard, Michael; Jansen, Nils; Junges, Sebastian; Katoen, Joost-Pieter; Karabag, Mustafa O.; Neary, Cyrus; Ornik, Melkior; Raju, Dhananjay; Suilen, Marnix; Tanaka, Takashi; Topcu, Ufuk; Vinod, Abraham P.; Vikalo, Haris; Wu, Bo; Xu, Zhe; and Zhao, Bo.	**5d. PROJECT NUMBER** **5e. TASK NUMBER** **5f. WORK UNIT NUMBER**
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) The University of Texas at Austin Office of Sponsored Projects 3925 West Braker Lane Building 156, Suite 3.340, MC: A9000	**8. PERFORMING ORGANIZATION REPORT NUMBER**
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) Defense Advanced Research Projects Agency (DARPA) Defense Sciences (DSO) 675 North Randolph Street Arlington, VA 2223-1714	**10. SPONSOR/MONITOR'S ACRONYM(S)** **11. SPONSOR/MONITOR'S REPORT NUMBER(S)**

12. DISTRIBUTION/AVAILABILITY STATEMENT

Distribution statement A, approved for public release.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

We developed theory and algorithms that will help autonomous systems execute patrol missions in urban environments, possibly in mixed teams of autonomous robotic agents with heterogeneous sensing, perception, computation, and actuation capabilities and a smaller number of soldiers (possibly in a supervisory role). We formalized a range of problems in the context of partial-information, stochastic games. While partial-information, stochastic games provided a highly expressive modeling language, synthesis of strategies in such games subject to temporal and logical constraints in their general form was known to be algorithmically impractical. Therefore, we established trade-offs between the expressivity of the problems and their algorithmic and computational tractability through a hierarchy of abstractions. The effort had three thrusts. Thrust I focused on synthesis in partial-information, stochastic games with temporal logic specifications. It developed approaches to suppress computational complexity and extract strategies that balanced the induced risk, ambiguity, and randomization. Thrust II took a proactive approach to cope with limitations at runtime through learning and active sensing in adversarial environments, while Thrust III established an understanding of cascading

15. SUBJECT TERMS

Autonomous robotic agents, patrol missions, heterogeneous sensing, perception, computation, adversarial environments, deceptive tactics

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON**
a. REPORT	**b. ABSTRACT**	**c. THIS PAGE**	S	48	Ufuk Topcu
					19b. TELEPHONE NUMBER *(Include area code)* 512-232-4195

Reset