# Stochastic games with neural perception mechanisms

Prof. Marta Kwiatkowska

Joint work with Gethin Norman, Dave Parker, Gabriel Santos and Rui Yan

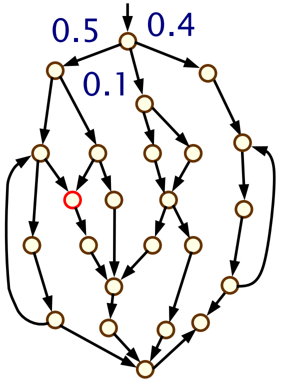
Department of Computer Science
University of Oxford

# Probabilistic model checking

Automatic verification and strategy synthesis from temporal logic properties for probabilistic models

System

Probabilistic model
e.g. Markov chain



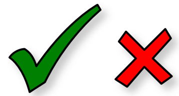

System require–ments

$P_{<0.01}$ [ $F^{\leq t}$ crash ]

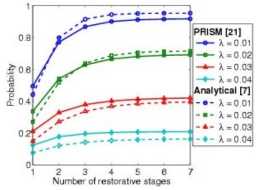
Probabilistic temporal logic specification
e.g. PCTL, CSL, LTL

Probabilistic model checker
e.g. PRISM

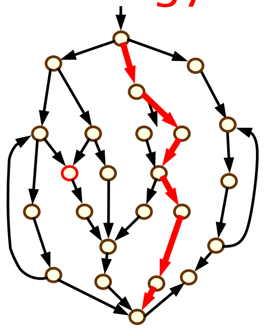
Result



Quantitative results




Strategy

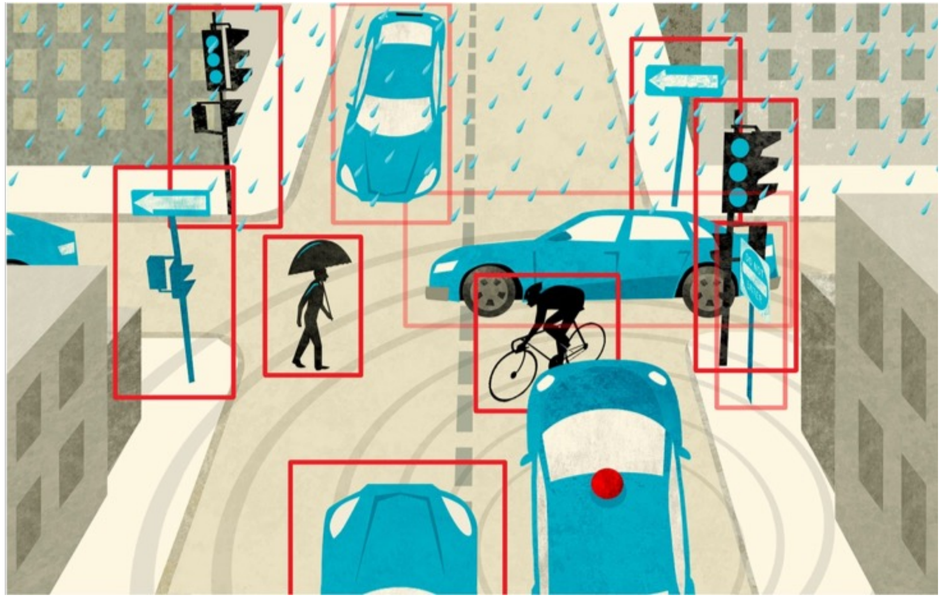


# Timeline and tool support: PRISM

- First algorithms proposed in 1980s
  - algorithms [Vardi, Courcoubetis, Yannakakis, Hansson, Jonsson, de Alfaro…]
  - & first implementations
- 2001: general purpose tools released
  - PRISM, STORM, and more: efficient extensions of conventional model checking
- Now mature area, of industrial relevance, new model checkers, tool competition
  - PRISM successfully used by non-experts in many domains
    - distributed algorithms, communication protocols, security protocols, biological systems, quantum cryptography, planning, robotics, …
  - genuine flaws found and corrected in real-world systems
  - ETAPS 2024 Test-of-Time Tool Award for PRISM
  - www.prismmodelchecker.org

# But which modelling abstraction?

- Complex scenarios!
  - human and artificial agents
  - distinct goals
  - autonomy
  - competitive/collaborative behaviour
  - context



- Natural to adopt a game-theoretic view
  - to account for the uncontrollable behaviour, in addition to controllable events
  - to enable strategic planning, negotiation and incentives
- Relevant for a multitude of autonomous and AI scenarios
  - e.g. decision making in economics, security, energy management, and …

# Driving as a game-theoretic problem

- Merging into traffic difficult for autonomous cars
- Human drivers are not behaving rationally!
- Here dynamic games, and interplay between long-horizon strategic game playing with short-horizon tactical moves, plus incentives

2019 International Conference on Robotics and Automation (ICRA)
Palais des congres de Montreal, Montreal, Canada, May 20-24, 2019

**Hierarchical Game-Theoretic Planning for Autonomous Vehicles**

Jaime F. Fisac*[1] Eli Bronstein*[1] Elis Stefansson[2] Dorsa Sadigh[3] S. Shankar Sastry[1] Anca D. Dragan[1]

***Abstract*— The actions of an autonomous vehicle on the road affect and are affected by those of other drivers, whether overtaking, negotiating a merge, or avoiding an accident. This mutual dependence, best captured by dynamic game theory, creates a strong coupling between the vehicle's planning and its predictions of other drivers' behavior, and constitutes an open problem with direct implications on the safety and viability of autonomous driving technology. Unfortunately, dynamic games are too computationally demanding to meet the real-time constraints of autonomous driving in its continuous state and action space. In this paper, we introduce a novel game-theoretic trajectory planning algorithm for autonomous driving, that enables real-time performance by hierarchically decomposing the underlying dynamic game into a long-horizon "strategic" game with simplified dynamics and full information structure, and a short-horizon "tactical" game with full dynamics and a simplified information structure. The value of the strategic game is used to guide the tactical planning, implicitly extending the planning horizon, pushing the local trajectory optimization closer to global solutions, and, most importantly, quantitatively accounting for the autonomous vehicle and the human driver's ability and incentives to influence each other. In addition, our approach admits non-deterministic models of human decision-making, rather than relying on perfectly rational predictions. Our results showcase richer, safer, and more effective autonomous behavior in comparison to existing techniques.**

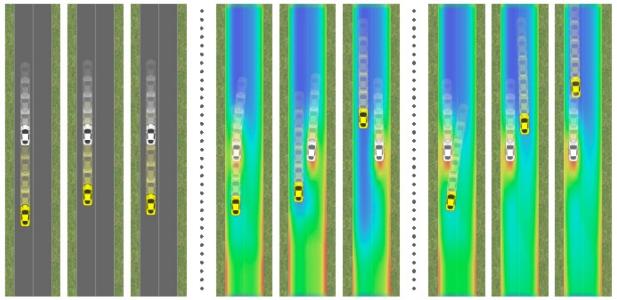


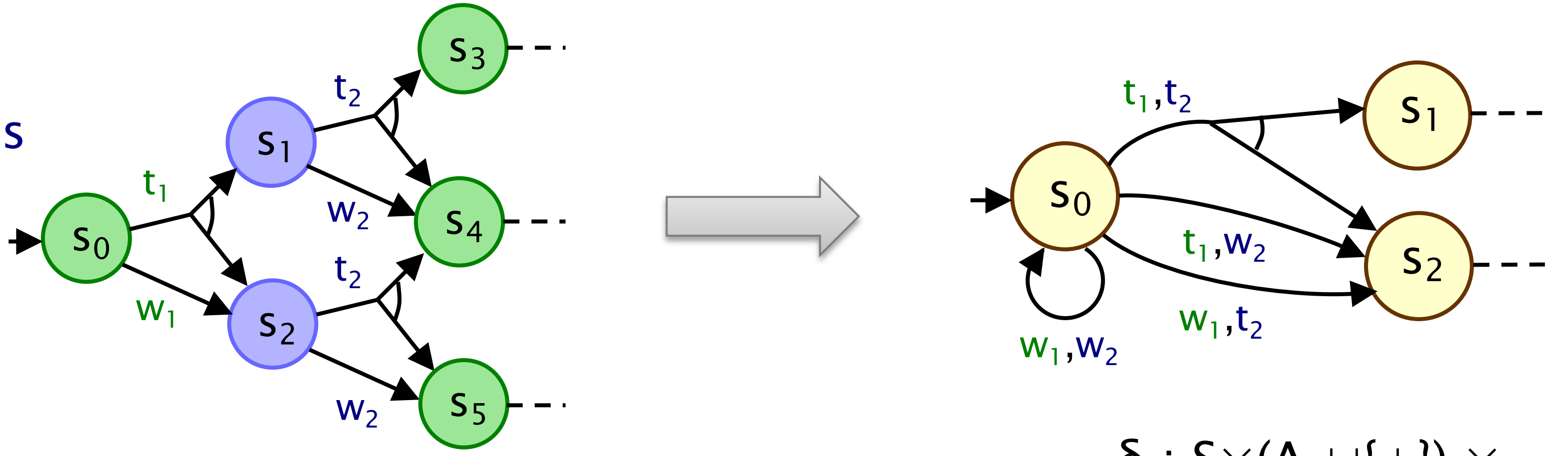
Fig. 1: Demonstration of our hierarchical game-theoretic planning framework on a simulated overtaking scenario. The heatmap displays the hierarchical planner's strategic value, ranging from red (low value) to blue (high value), which accounts for the outcome of possible interactions between the two vehicles. *Left:* Using a short-horizon trajectory planner, the autonomous vehicle slows down and is unable to overtake the human. *Center:* Using the hierarchical game-theoretic planner, the autonomous vehicle approaches the human from behind, incentivizing her to change lanes and let it pass (note the growth of a high-value region directly behind the human in the left lane). *Right:* If the human does not maneuver, the autonomous vehicle executes a lane change and overtakes, following the higher values in the right lane.

autonomous driving. Most approaches in the literature follow
a "pipeline" approach that generates predictions of the trajec-

# Concurrent stochastic games

- Stochastic games are a realistic abstraction
  - multiple agents acting autonomously and concurrently
  - making action choices without knowledge of others
  - stochasticity arises from uncertainty, noise, etc
- Concurrent stochastic games (CSGs)
  - players choose actions concurrently & independently
  - jointly determines (probabilistic) successor state

Turn-based stochastic games (TSGs)

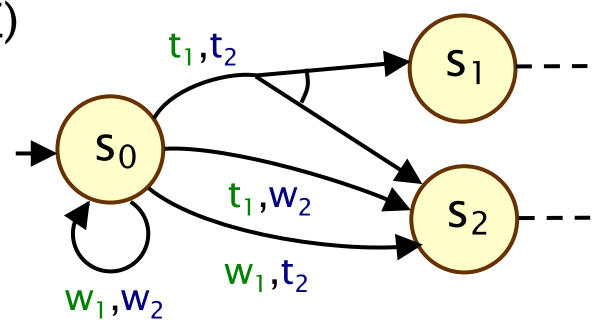



Concurrent stochastic games (CSGs)

$$\delta : S \times (A_1 \cup \{\perp\}) \times \ldots \times (A_n \cup \{\perp\}) \to Dist(S)$$

# Model checking for CSGs

- Overall model checking algorithm for logic rPATL (reward prob. variant of ATL)
  - key ingredient is solving (zero-sum) 2-player CSGs (PSPACE)
  - note that optimal strategies are randomised




- Rely on a value iteration approach
  - e.g. max/min reachability probabilities
  - $\sup_{\sigma_1} \inf_{\sigma_2} \Pr_s^{\sigma_1,\sigma_2}(F\ ✓)$ for all states $s$
  - values $p(s)$ are the least fixed point of:

$$p(s) = \begin{cases} 1 & \text{if } s \models ✓ \\ val(Z) & \text{if } s \not\models ✓ \end{cases}$$

  - where $Z$ is the matrix game with $z_{ij} = \Sigma_{s'}\ \delta(s,(a_i,b_j))(s') \cdot p(s')$

- Implementation
  - matrix games solved as linear programs
    - (LP problem of size |A|)
  - required for every iteration/state
    - which is the main bottleneck
  - but we solve CSGs of ~3 million states

Automatic Verification of Concurrent Stochastic Games, FMSD 2021

# Equilibria-based properties

- Beyond zero-sum games:
  - players/components may have distinct objectives but which are not directly opposing (zero-sum)
- We work with Nash equilibria (NE)
  - no incentive for any player to unilaterally change strategy
  - actually, we use $\epsilon$-NE, which always exist for CSGs

$\sigma=(\sigma_{1,\ldots},\sigma_n)$ is an $\epsilon$-NE for objectives $X_1,\ldots,X_n$ iff:

for all $i$ : $E_s^{\sigma}(X_i) \geq \sup \{ E_s^{\sigma'}(X_i) \mid \sigma'=\sigma_{-i}[\sigma_i'] \text{ and } \sigma_i' \in \Sigma_i \} - \epsilon$

- We extend rPATL model checking for CSGs
  - with social-welfare Nash equilibria (SWNE)
  - i.e., NE which also maximise the joint sum $E_s^{\sigma}(X_1) + \ldots E_s^{\sigma}(X_n)$
  - also correlated equilibria and fairness/cost constraints

Zero-sum properties

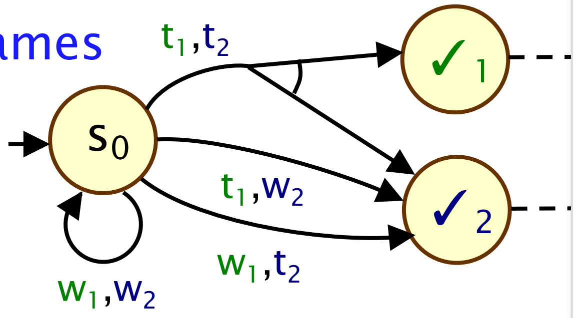
$\langle\langle robot_1 \rangle\rangle_{max=?}\, P\, [\, F^{\leq k}\, goal_1\, ]$

$\langle\langle robot_1{:}robot_2 \rangle\rangle_{max=?}\, (P\, [\, F^{\leq k}\, goal_1\, ]+P\, [F^{\leq k}\, goal_2])$

Equilibria-based properties (SWNE)

# Verification and synthesis for Nash equilibria

- Equilibria verification for CSGs
  - needs solution of bimatrix games
  - (basic problem is EXPTIME)
  - strategies need history and randomisation
- Can also synthesise equilibria
- We extend the value iteration approach:




- Implementation
  - we adapt a known approach using labelled polytopes, and implement it via SMT
  - optimisations: filtering of dominated strategies

$$p(s) = \begin{cases} (1,1) & \text{if } s \models \checkmark_1 \wedge \checkmark_2 \\ (1, p_{max}(s, \checkmark_2)) & \text{if } s \models \checkmark_1 \wedge \neg\checkmark_2 \\ (p_{max}(s, \checkmark_1), 1) & \text{if } s \models \neg\checkmark_1 \wedge \checkmark_2 \\ val(Z_1, Z_2) & \text{if } s \models \neg\checkmark_1 \wedge \neg\checkmark_2 \end{cases}$$

standard MDP analysis

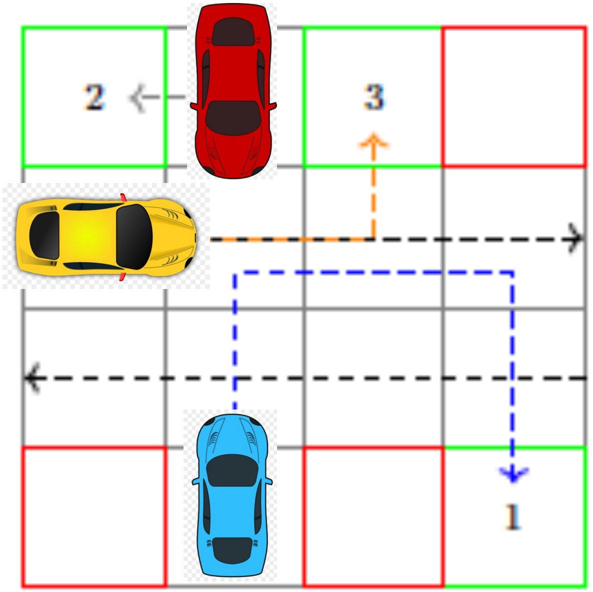
bimatrix game

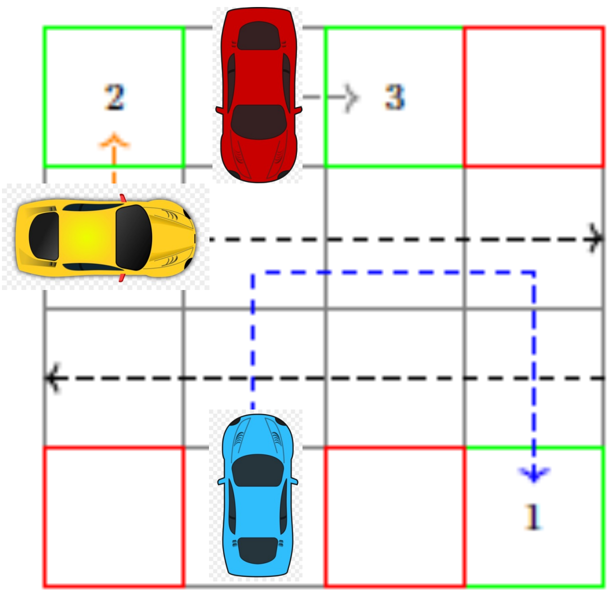
  - where $Z_1$ and $Z_2$ encode matrix games similarly to before

Equilibria-based Model Checking for Concurrent Stochastic Games, In *Proc* FM 2019

# Automated parking example

- Safe parking
  $\langle\langle v_1,v_2,v_3\rangle\rangle(NE,SW)_{max\geq 3}$ $(P[\neg coll_1 U\ park_1] + P[\neg coll_2 U\ park_2] + P[\neg coll_3 U\ park_3])$
- Minimise collective travelled distance
  $\langle\langle v_1,v_2,v_3\rangle\rangle(NE,SC)_{min=?}$ $(R[F\ park_1]+R[F\ park_2]+R[F\ park_3])$



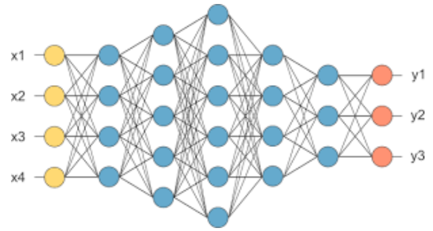



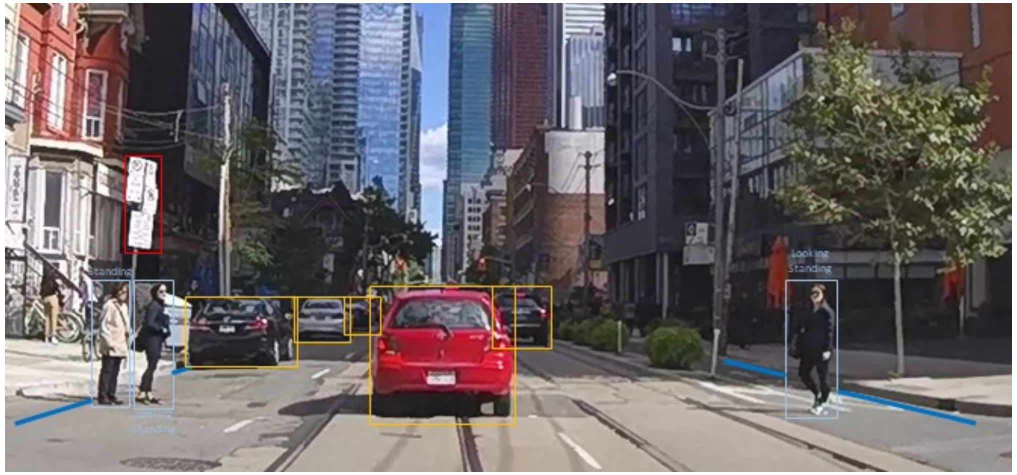

Two equilibria: distance 10 distance 8 (SCNE)

# Challenges beyond

- Advances in algorithmic solutions combining symbolic SAT/SMT and numerical computation, including equilibria
- Complex CSG scenarios: flaws found, automated synthesis of strategies
- Finite, discrete state CSGs, solve up to ~2 million states
- Implemented in PRISM-games, extension of PRISM

- Challenges due to emerging trends
- Neural networks used for perception!



- And hence must consider
  - continuous state spaces
  - partial observability

# This lecture…

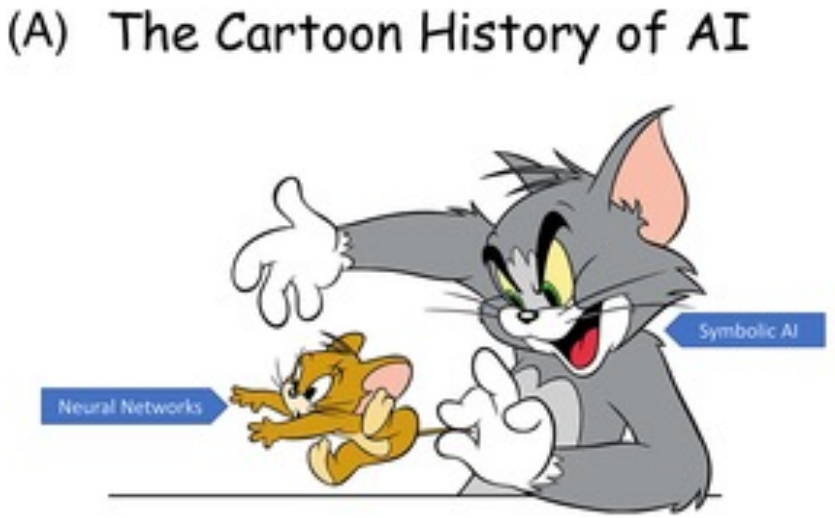
- Progress in probabilistic model checking for concurrent stochastic games
  - modelling competitive/cooperative behaviour, in adversarial environments
  - zerosum and equilibria (Nash, correlated), optimality criteria
  - model checking and strategy synthesis, implemented in PRISM-games
- Ongoing work: neuro-symbolic concurrent stochastic games
  - Strategy synthesis for zero-sum neuro-symbolic concurrent stochastic games, Information and Computation, 2024
  - Point-based value iteration for neuro-symbolic POMDPs, under revision for AI Journal
  - Partially observable stochastic games with neural perception mechanisms, FM 2024
  - Finite-horizon Equilibria for Neuro-symbolic Concurrent Stochastic Games, UAI 2022
  - HSVI-based Online Minimax Strategies for Partially Observable Stochastic Games with Neural Perception Mechanisms, L4DC 2024
- Conclusions and perspectives

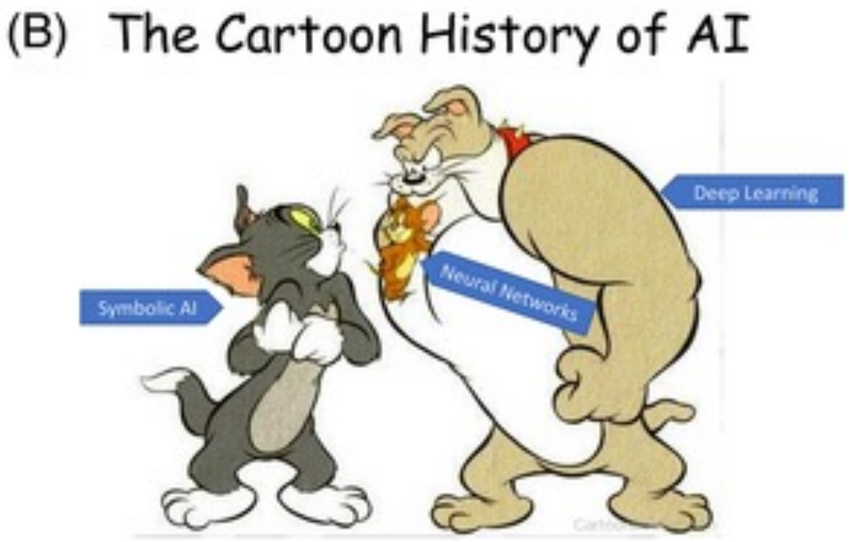
Probabilistic Model Checking for Strategic Equilibria-based Decision Making: Advances and Challenges, In *Proc* MFCS 2022

# But what is neuro-symbolic AI?

- The history of AI

  … from symbolic, logic-based origins… to the rise of neural networks

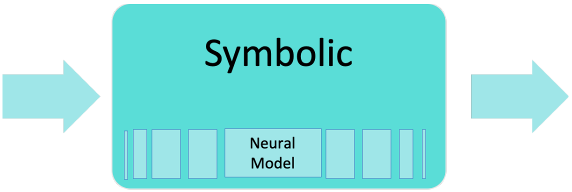


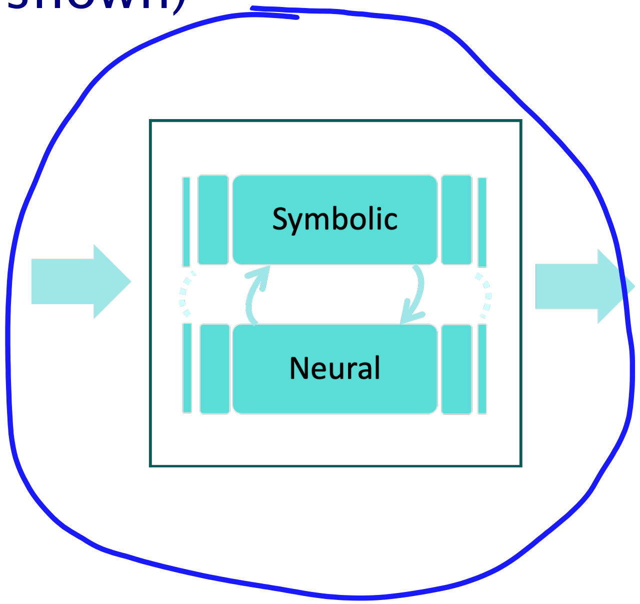

- Neuro-symbolic AI: umbrella term for combining neural and symbolic approaches
- Grand challenge: how to obtain high-performing neuro-symbolic AI

# Neuro-symbolic integration

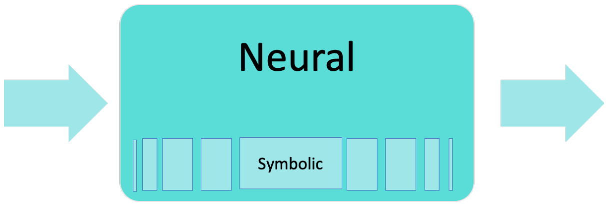
- No agreement on how to integrate symbolic and neural approaches
- 6 designs proposed (only 3 shown)




Symbolic[Neural]




Symbolic;Neural



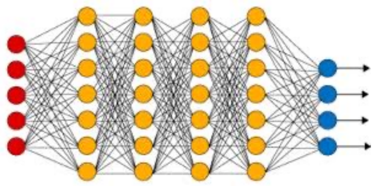

Neural[Symbolic]

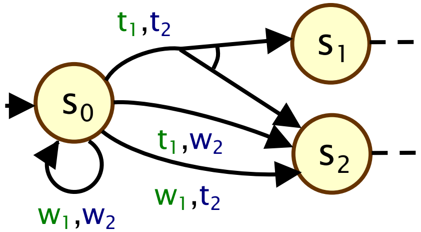
- Focus on safety and rigorous engineering, rather than performance

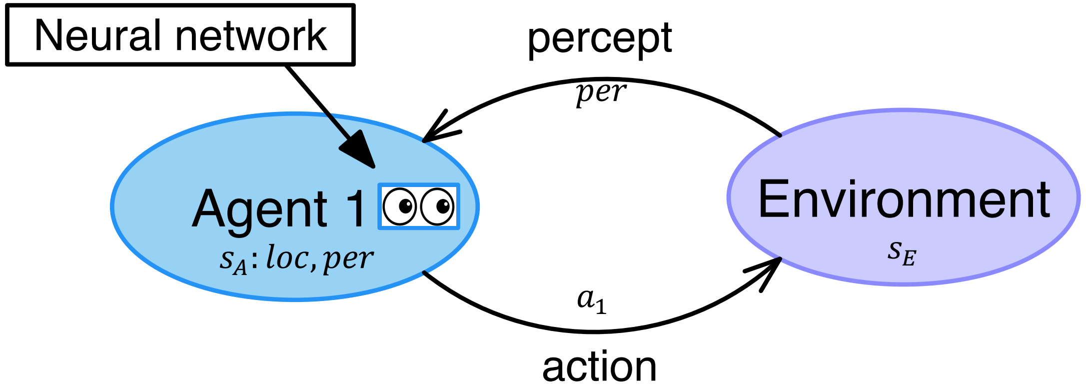
Based on Harsha Kokel's interpretation of The Third AI Summer, Robert S. Engelmore Memorial Lecture at AAAI 2020 by Henry Kautz

# Neural perception mechanisms

- Neural networks (NNs) successful in wide range of perception tasks
  - mapping signals to concepts in vision, speech, behaviour prediction, etc
- Assume each agent
  - uses a neural network classifier to observe continuous environment as symbolic percept
  - percepts are learnt concepts stored locally and used in decision making
  - focus on trained NN classifiers (ReLU or their convex relaxations)
  - conventional, symbolic decision making
- Aim for white box access to NNs
  - polyhedral decompositions of environment via preimage computation
  - different from perception contracts (Mihailovic et al) and perception abstractions (Pasareanu et al)

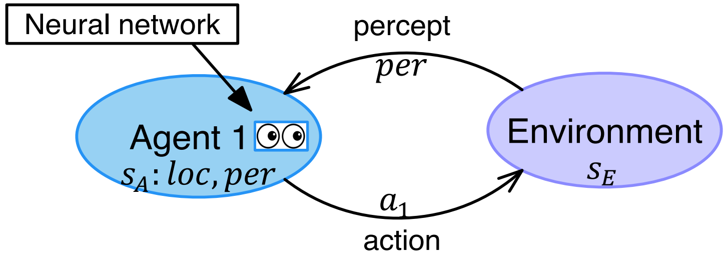




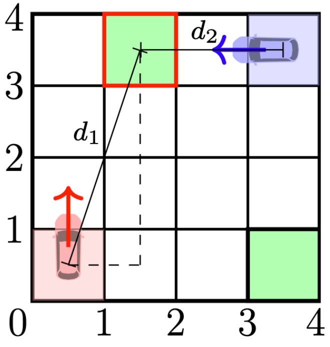
# Neuro-symbolic games

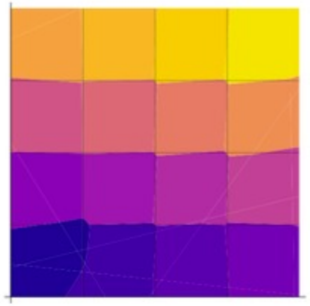
- Agents endowed with neural perception and symbolic decision making
  - here: NN classifiers (or other machine learning) for perception tasks
  - constrained interface: convert inputs such as images to symbolic percepts
  - plus: local strategies for control decisions




- Neuro-symbolic CSGs (two players/coalitions)
  - finite-state agents + continuous-state environment E
    - $S = (Loc_1 \times Per_1) \times (Loc_2 \times Per_2) \times S_E$
  - agents only use a (learnt) perception function to observe E
    - $obs_i : (Loc_1 \times Loc_2) \times S_E \to Per_i$
  - CSG-like joint actions update state probabilistically

- Example: dynamic vehicle parking
  - NN maps exact vehicle position to perceived grid cell
  - stochasticity from, e.g., motion imprecision

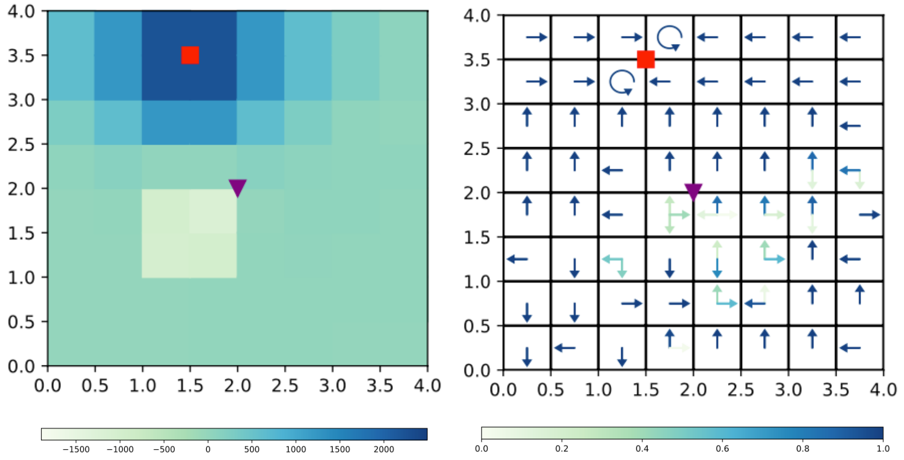


# Strategy synthesis for neuro-symbolic CSGs

- Consider zero-sum (discounted) expected reward over infinite horizon
  - for now, we assume full observability
  - value exists under Borel assumptions, fixed point of minimax
  - but optimal value may not be finitely representable
- Value iteration (VI) approach, exploit structure
  - continuous state-space decomposed into regions with the same percept (and reward)
  - further subdivision at each iteration via Borel decomposition, under assumptions
  - abstraction based on piecewise-continuous value functions, preserved by NNs and VI
- Implementation
  - pre-image computations of NNs
  - polytope representations of regions (ReLU)
  - LPs to solve zero-sum games at each step

Dynamic vehicle parking with larger (8x8) grid and simpler (regression) perception



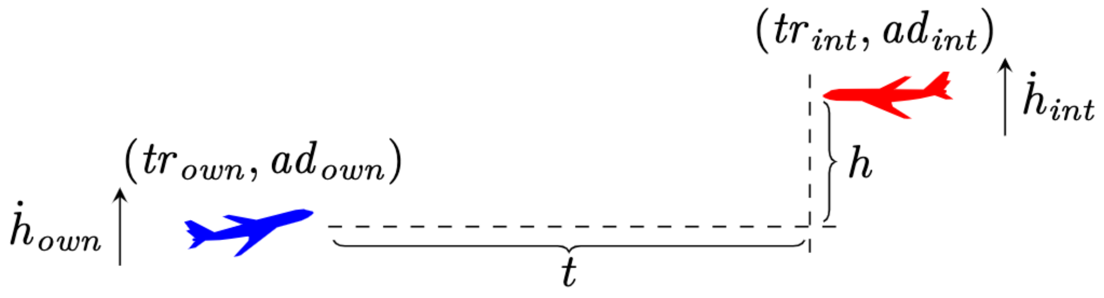

Value function (fragment)

Optimal strategy (fragment)

Strategy synthesis for zero-sum neuro-symbolic concurrent stochastic games, Inf & Comp, 2024

# Equilibria for neuro-symbolic games

- Consider (undiscounted) finite-horizon equilibria (Nash and correlated)
  - uncountable state space, existence of infinite-horizon NE for CSGs open problem
  - subgame perfect equilibria from initial state, social welfare optimal
- Focus on (simpler) finite-horizon game unfolding from an initial state




- Example: Vertical Collision Avoidance System (VCAS)
  - two agents: ownship and intruder
  - ReLU NN monitors relative altitude h, climb rates, and time to horizontal separation t
  - in every time step issues advisory ad (set of accelerations for agent to choose from)
  - added trust tr to monitor compliance, updated probabilistically

# Equilibria synthesized for VCAS

Optimal equilibria over horizon computed for VCAS

Horizon 3 (left)

- optimal to follow advisory

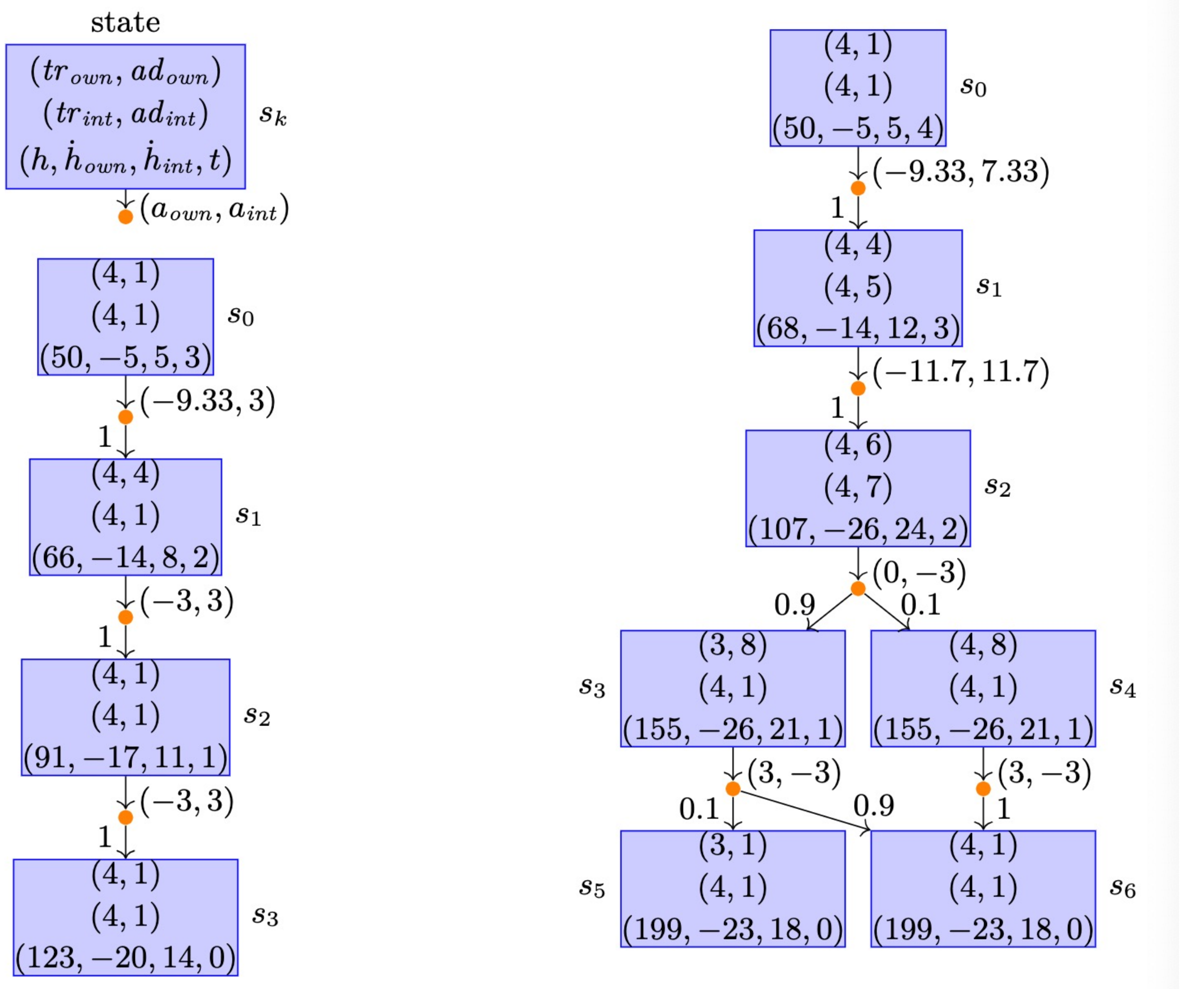
Horizon 4 (right)

- optimal strategy shows deviation from advisory

Example advisories

DNC (do not climb) 9.33

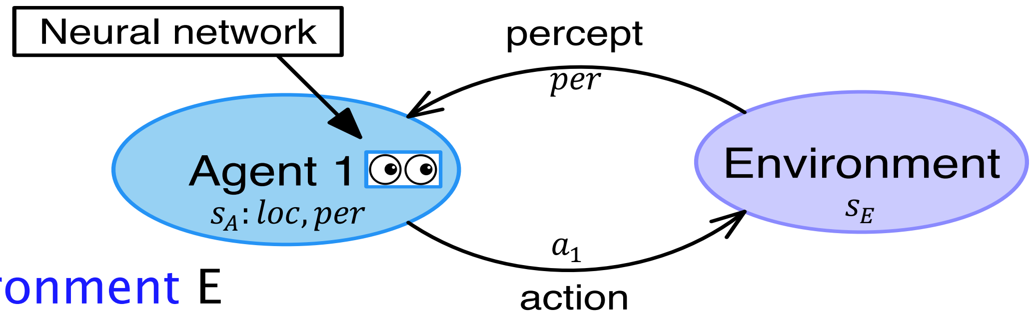
COC (clear of conflict) 3



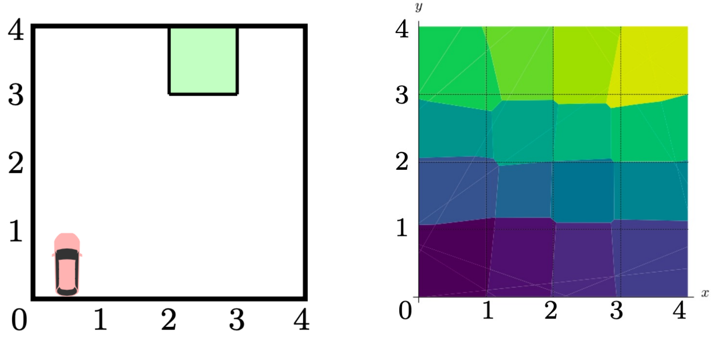

Finite-horizon Equilibria for Neuro-symbolic Concurrent Stochastic Games, In *Proc* UAI 2022

# Neuro-symbolic POMDPs

- Need partial observability for neural perception, not just continuous environment!




- NS-POMDP (partially observable MDP)
  - finite-state agent + continuous-state environment E
    - S = (Loc×Per) × $S_E$
  - agent uses a (trained) perception function to observe E
    - obs : (Loc×$S_E$) → Per
  - and transitions based on local state and percept
    - δ : (Loc×Per×A) → Dist(Loc)
- Work in belief space over S (probability distributions)

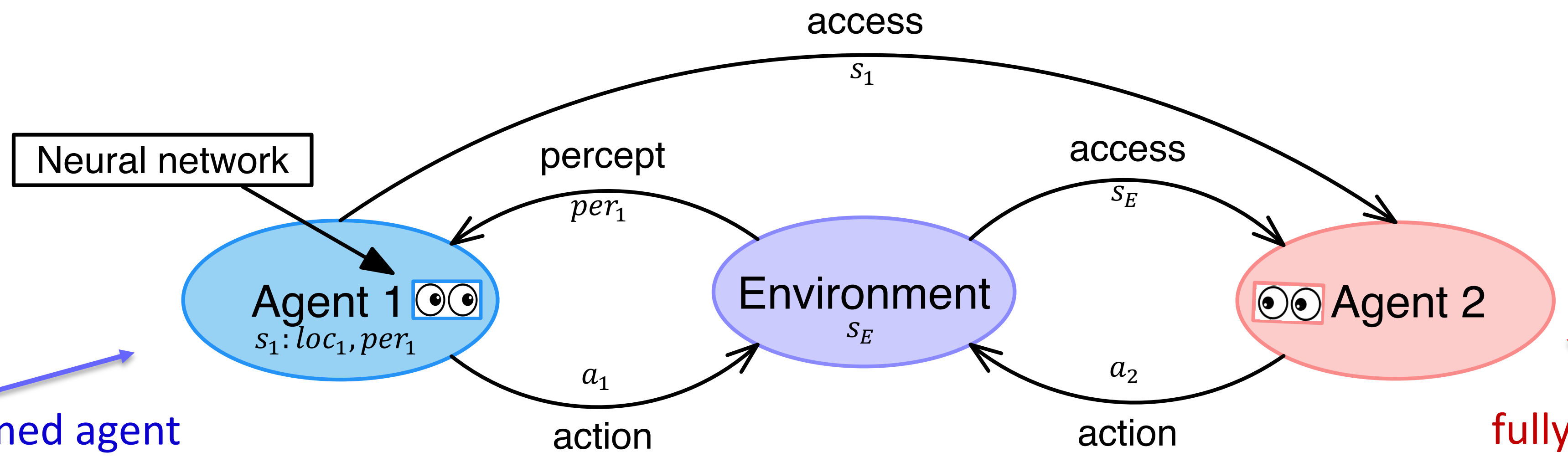


Car parking (4x4)
Localisation NN

Point-based value iteration for neuro-symbolic POMDPs, under revision for AIJ

# Neuro-symbolic POSGs

- Restrict to one-sided variant, new subclass of hybrid-state POSGs




Partially informed agent
maximize

fully informed agent
minimize

NS-POSGs (same syntax as NS-CSGs)

- finite-state agents + continuous-state environment E
- Agent 1 uses a (learnt) perception function to observe E
  - $obs_1 : (Loc_1 \times S_E) \rightarrow Per_1$
- and transitions on percepts and local states $\delta : (Loc_1 \times Per_1 \times A) \rightarrow Dist(Loc)$
- Agent 2 fully informed

discounted reward:

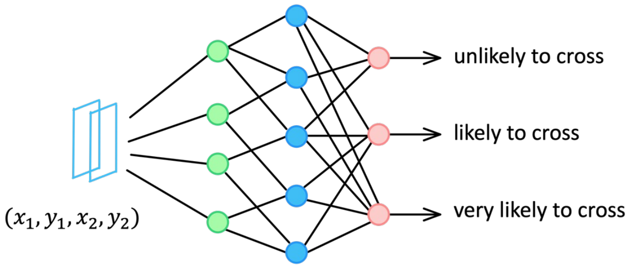
$$Y(\pi) = \sum_{k=0}^{\infty} \beta^k r(\pi(k), \pi[k])$$

zero-sum

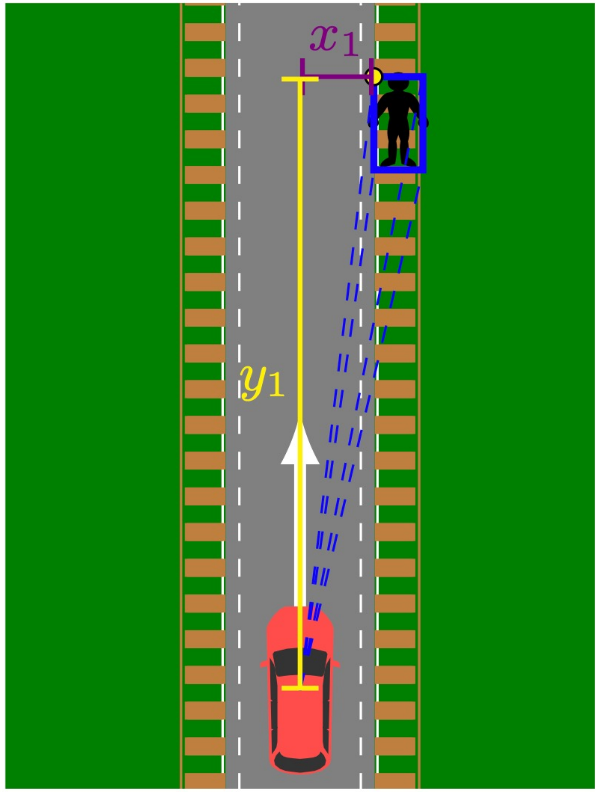
Partially observable stochastic games with neural perception mechanisms, In *Proc* FM 2024

# Example: pedestrian–vehicle interaction

- Autonomous vehicle
  - partially informed
  - aims to predict pedestrian's intention
  - using NN trained from video data



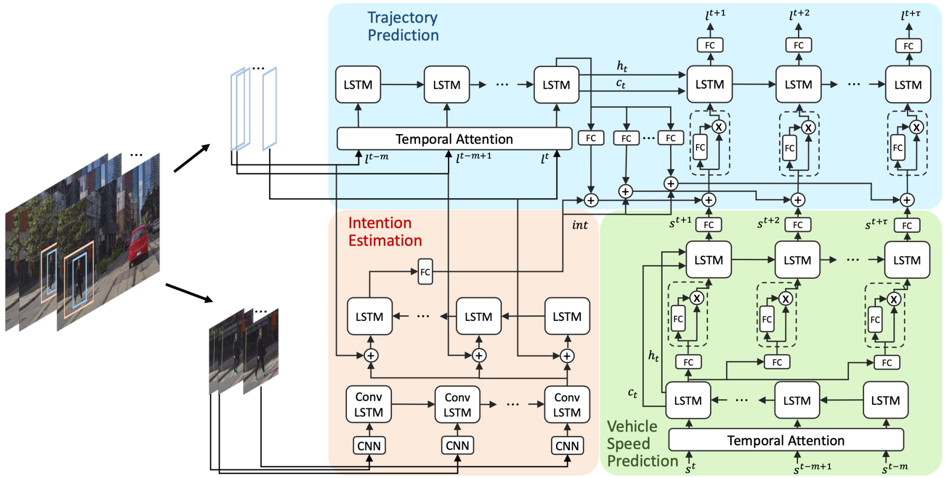

- Pedestrian
  - fully informed for worst-case analysis
  - decides whether to cross or return to sidewalk
- Goal: synthesise strategy for vehicle to minimize likelihood of crash (opposite for pedestrian)

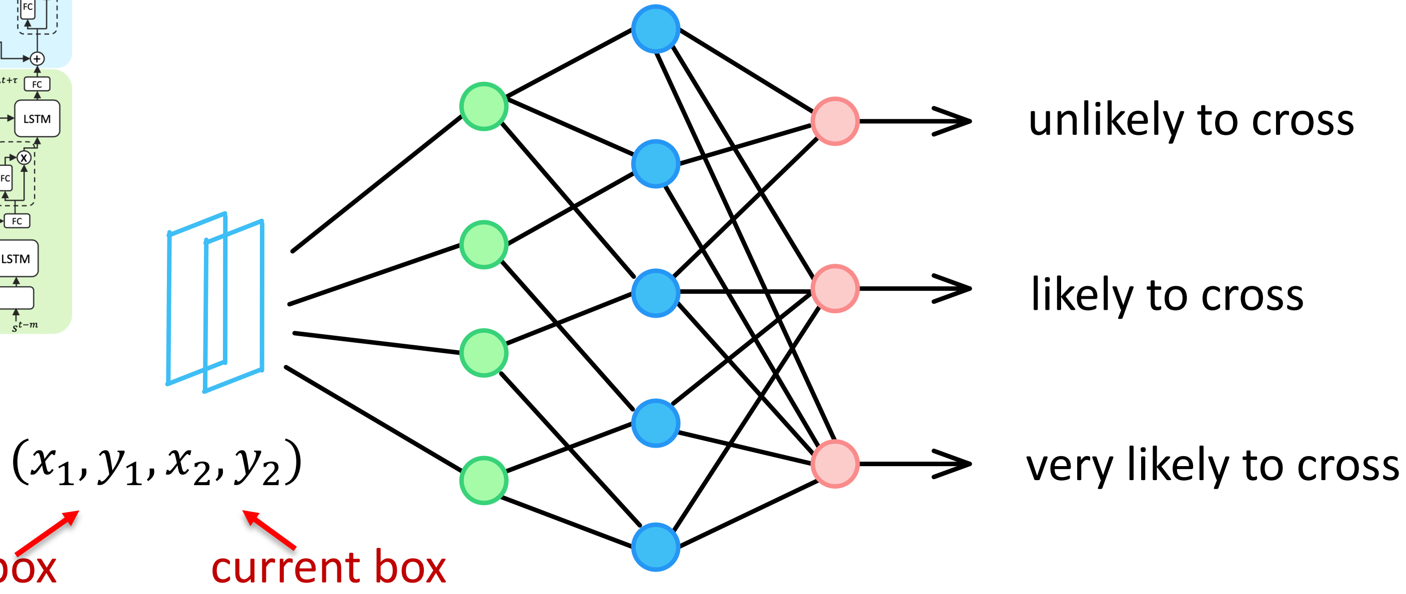



https://data.nvision2.eecs.yorku.ca/PIE_dataset/

# Intention prediction: the neural network

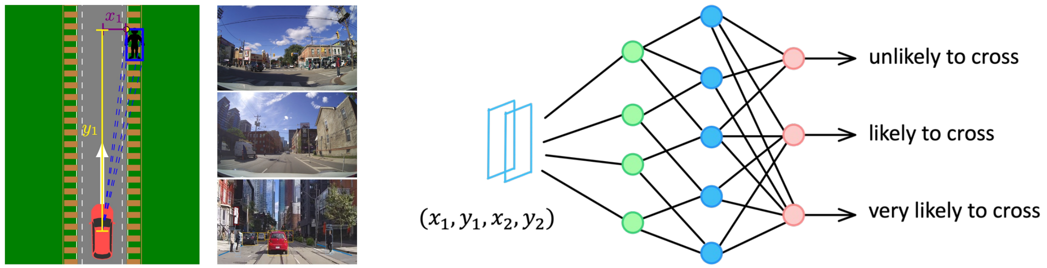
- RNN models trained on video data (PIE dataset)
- Here simplify NN to two fixed size frames (relative positions of pedestrian to vehicle )

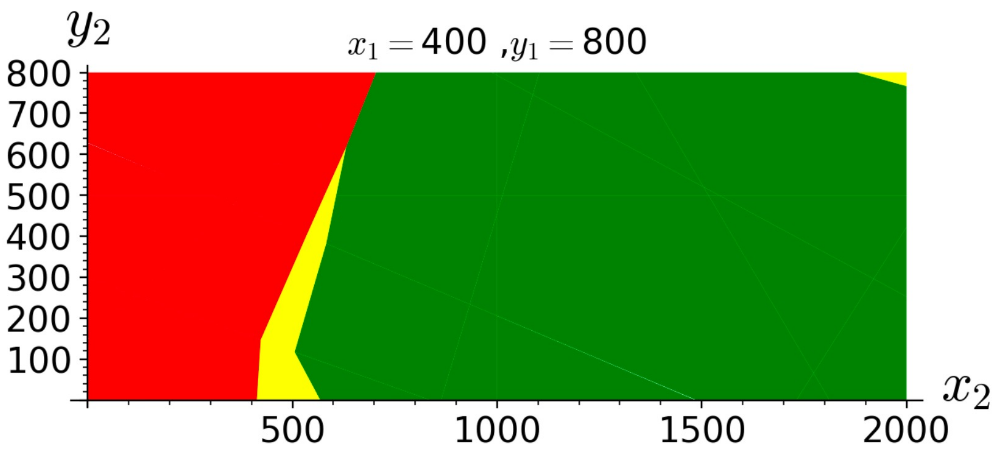


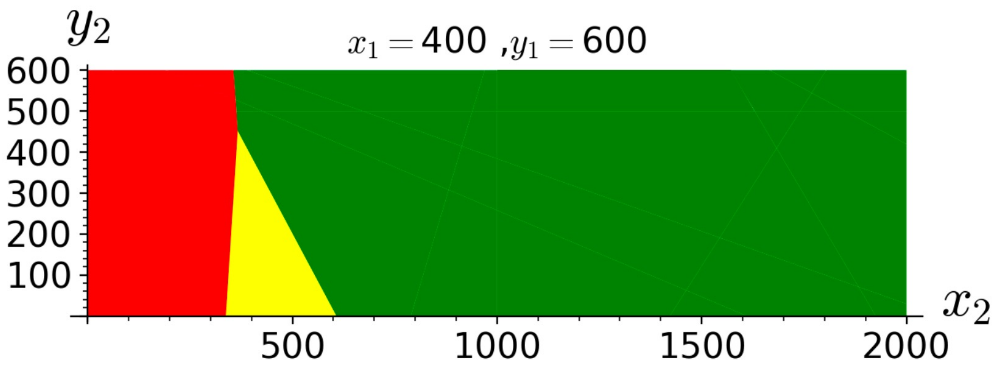




https://data.nvision2.eecs.yorku.ca/PIE_dataset/

# Intention prediction: decomposition via preimage



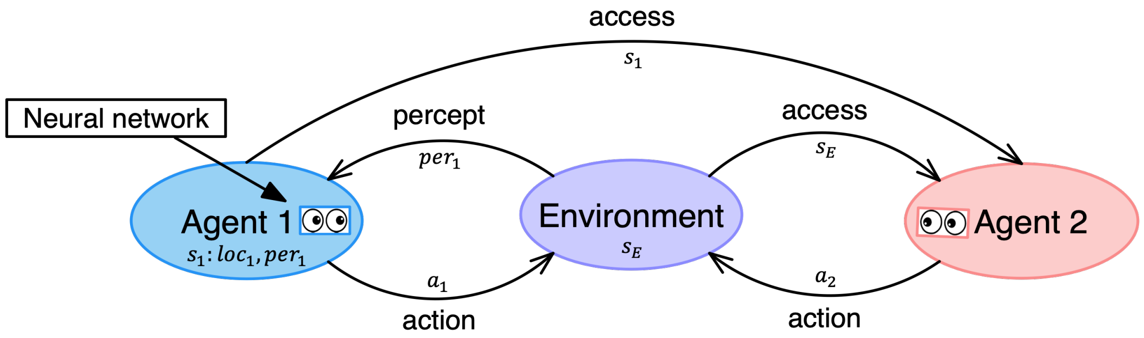

Perception FCP (decomposition of E via preimage)

- NN-based perception divides the environment into 3 classes: green: unlikely, yellow: likely, red: very likely
- Compute the preimage of NN

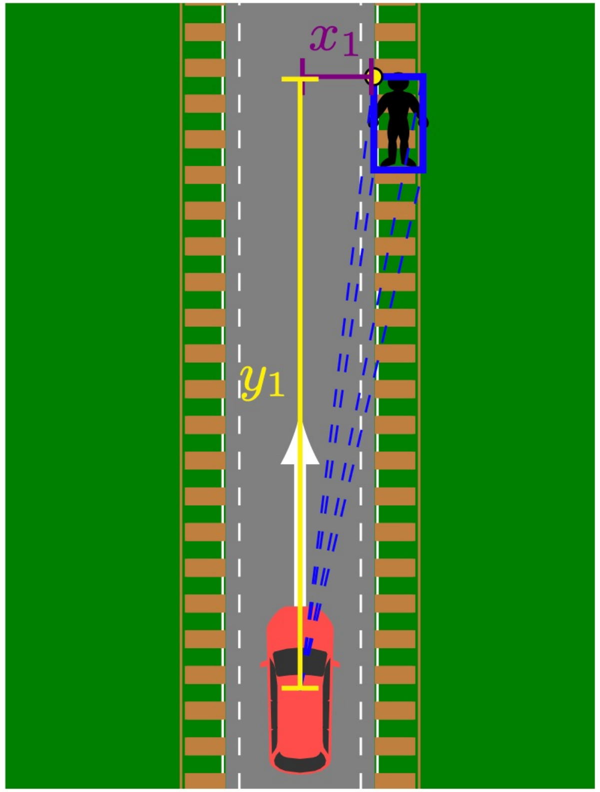
$$0 \le x_1, x_2 \le 2000$$
$$0 \le y_1, y_2 \le 1000$$

- Distances in meters (crash occurs for small values of x, y)

# Pedestrian-vehicle interaction as NS-POSG

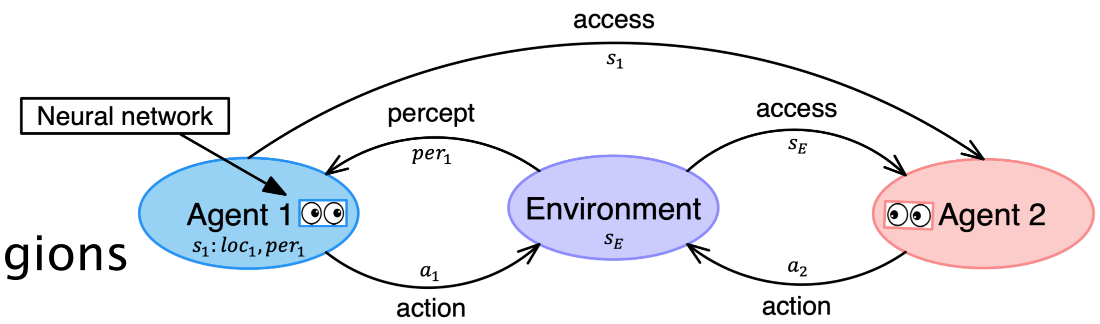



- Agent 1: vehicle
  - $loc_1$: speed
    $per_1$: pedestrian intention
  - $a_1$: acceleration (e.g. +3,-3)
- Agent 2: pedestrian
  - $a_2$: cross, back
- Environment E
  - two successive pedestrian positions ($x_1$,$y_1$,$x_2$,$y_2$)



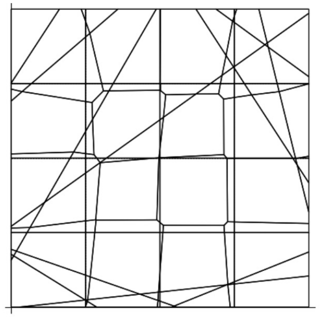

# Strategy synthesis for neuro-symbolic POSGs

- Consider zero-sum (discounted) expected reward over infinite horizon
  - one sided, so Agent 2 can recover beliefs of Agent 1
  - assume determined, as value may not exist




- HSVI approach (extend Horak *et al* 2023)
  - continuous state-space decomposed into regions
  - further subdivision at each iteration
  - work with a class of piecewise-continuous $\alpha$-functions, + closure properties
  - anytime

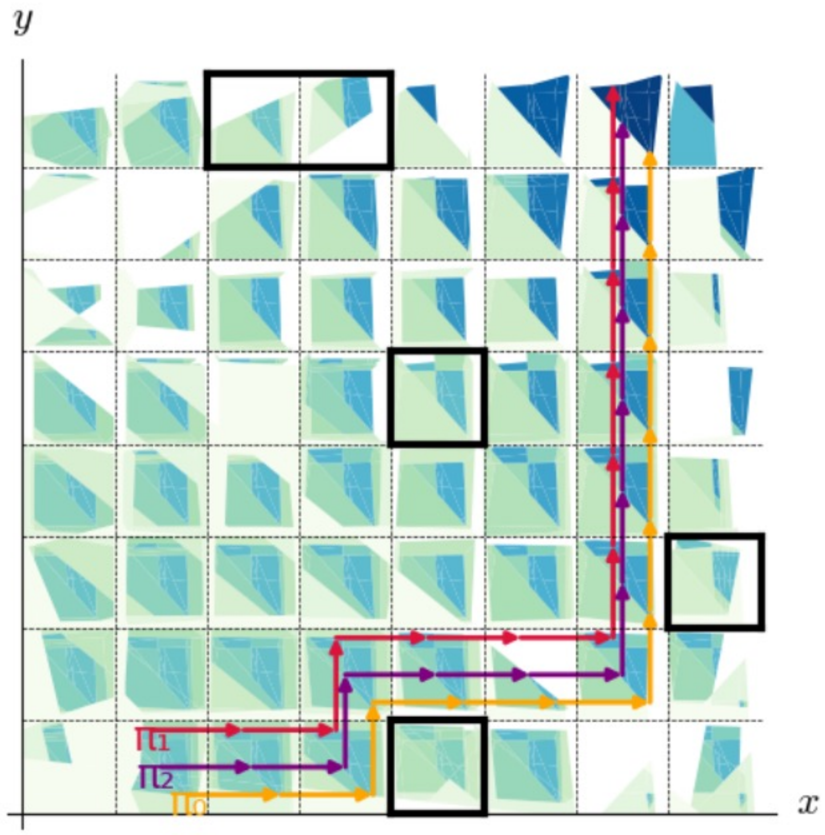
PWC $\alpha$-function
polyhedra + value vector

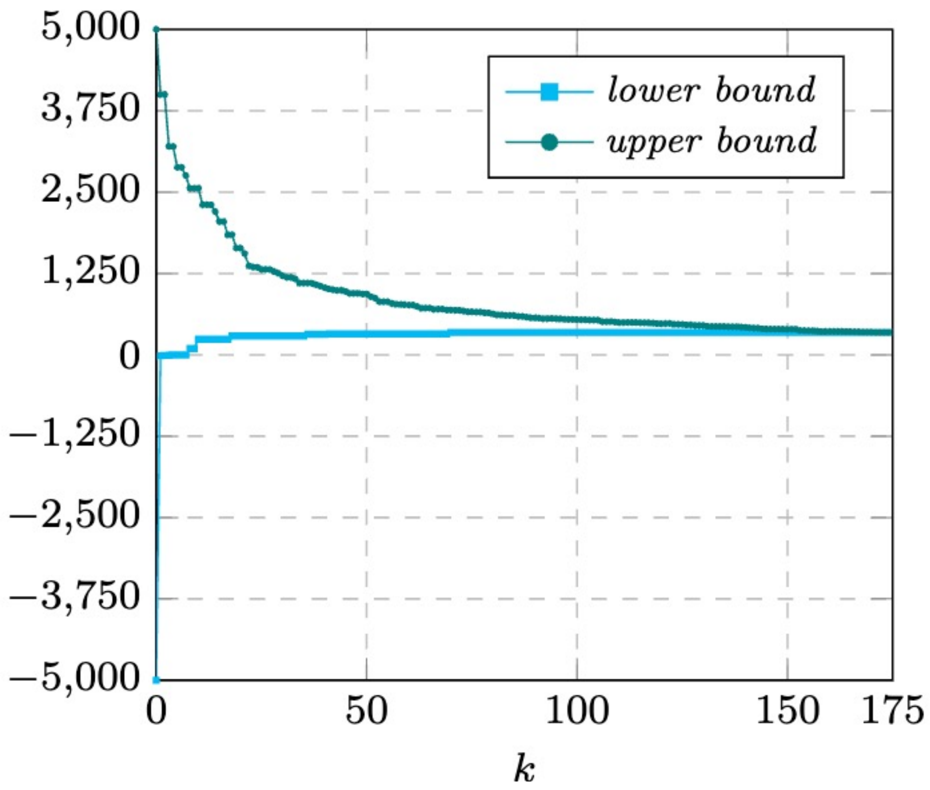


- Implementation
  - polyhedral pre-image computations of NNs
  - LPs to compute lower/upper bound and minimax values
  - expensive to compute!

Partially observable stochastic games with neural perception mechanisms, In *Proc* FM 2024

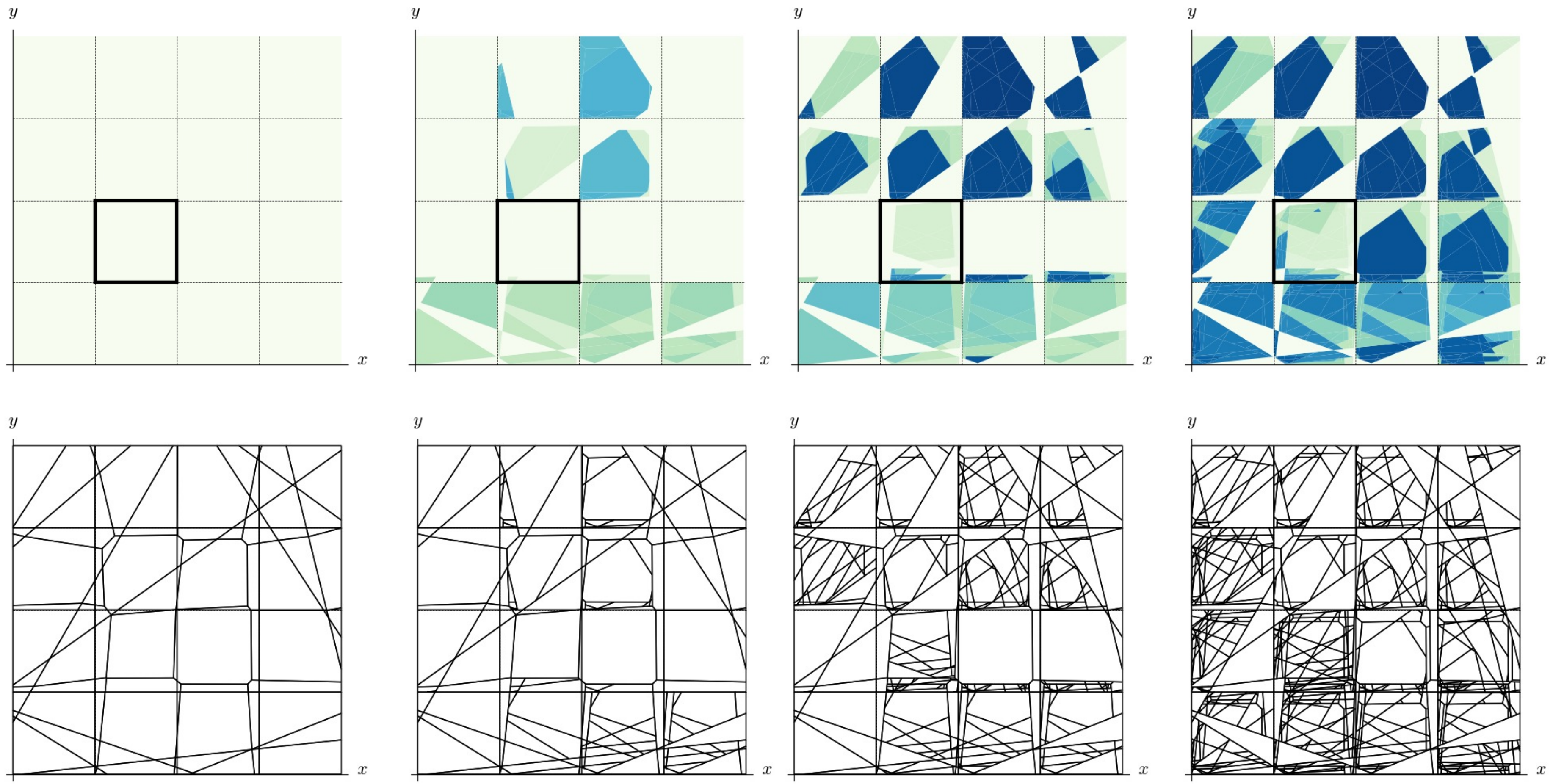
# Strategy synthesis (8x8 car parking w obstacles)




- Paths (from three particles from initial belief) and values (lower bounds) showing parking spot reached while avoiding obstacles
- NS-POMDP variant of HSVI

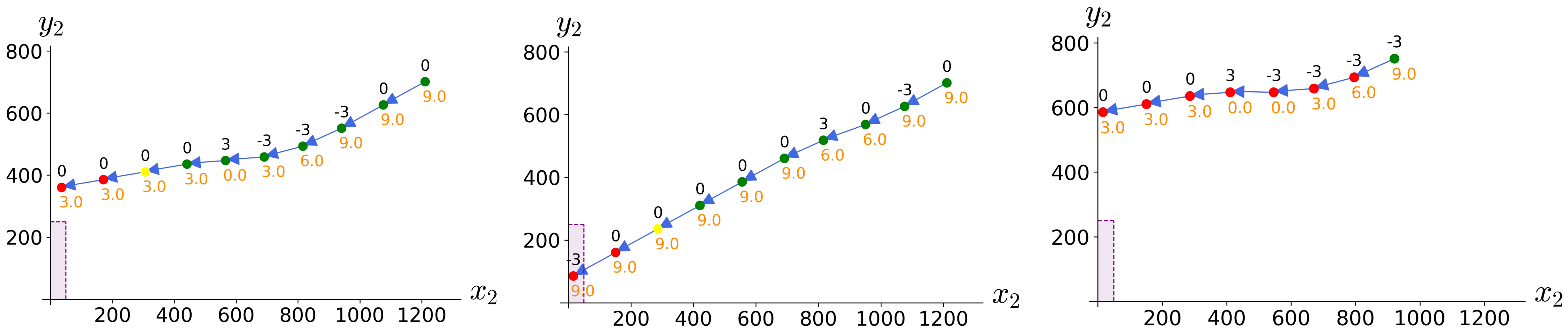
# HSVI value computation (4x4 car parking)




Values (top) and region outlines (bottom) showing refinement for the initial and maximum (all states) the first 5, 25 and all polyhedral decompositions

# Experiments for the pedestrian–vehicle POSG

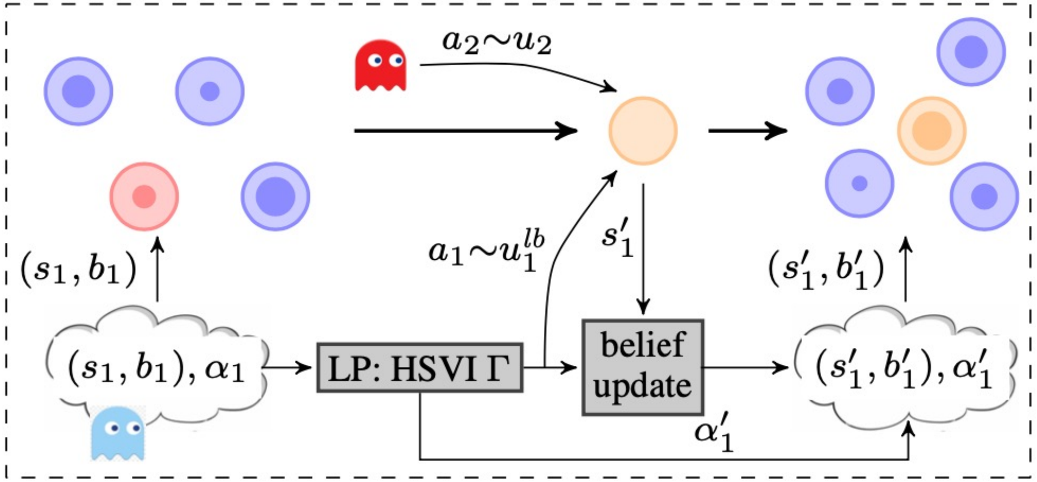
- Note, working with learnt percepts!



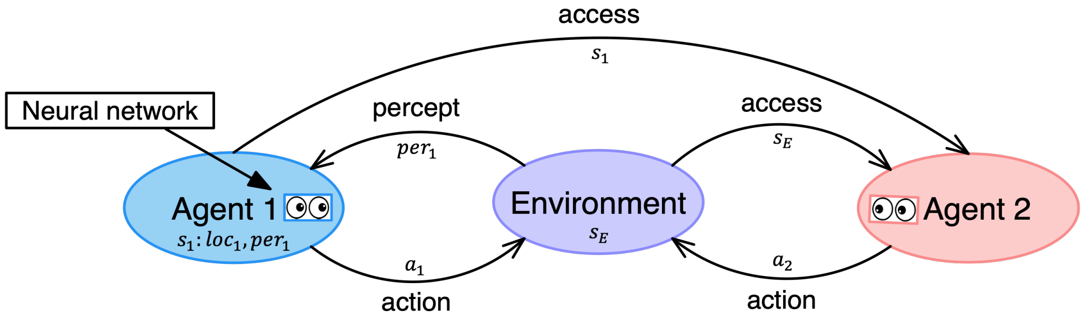

- Game setup
  - assume the pedestrian is crossing
  - showing intention
    - unlikely, likely, very likely
  - acceleration, speed, crash region
- Results
  - most paths are safe
  - intention prediction looks reasonable
  - crash may not be avoided (when intention detected too late)

# Efficient online minimax strategies

- How to synthesize strategies based on the lower and upper bound functions




NS-HSVI continual re-solving for $Ag_1$




Online continual resolving

- keeps track of belief and counterfactual values
- builds and solves a game without storing complete strategy

Our variant

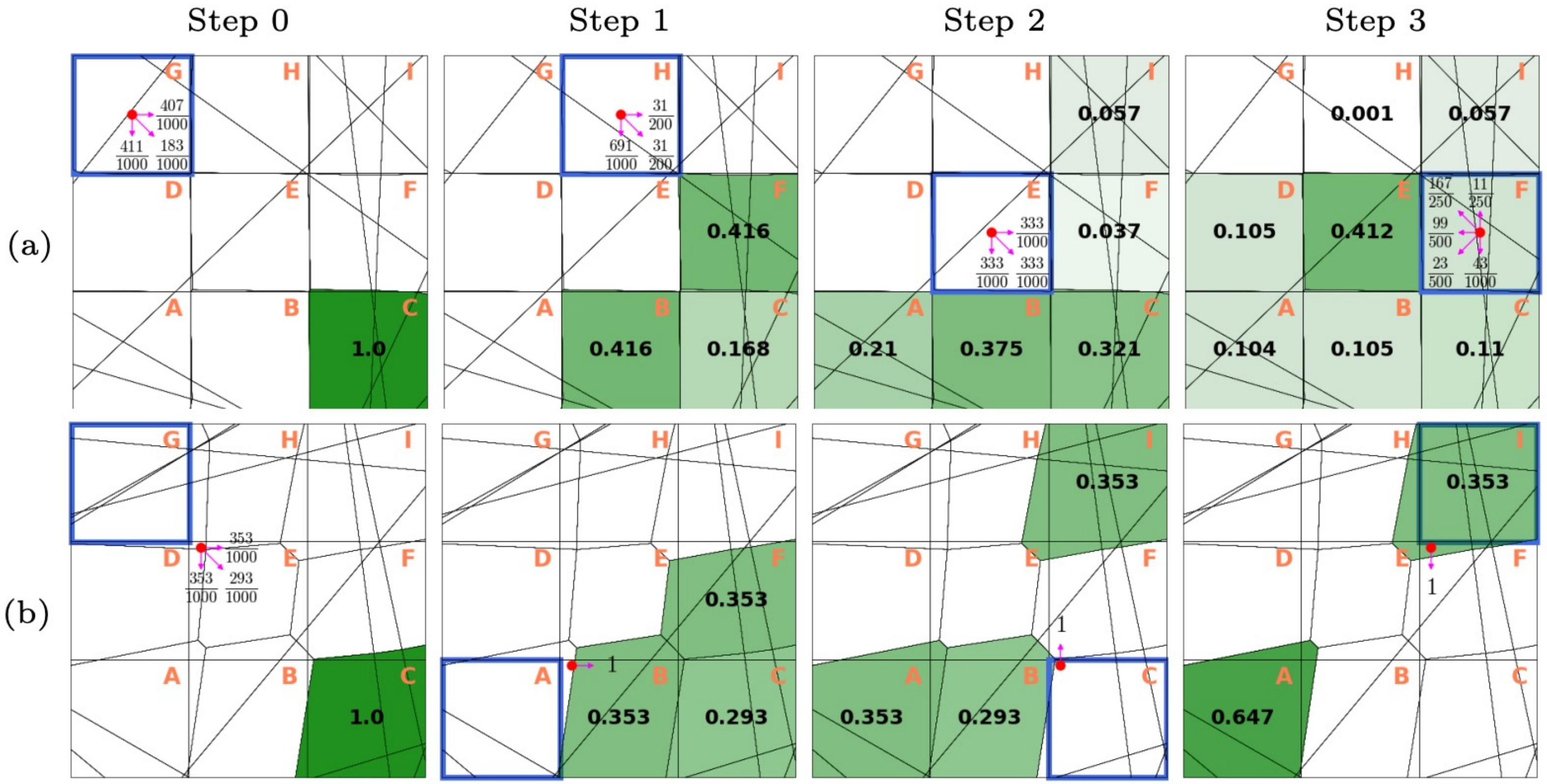
- precomputes HSVI lower bound
- keeps track of belief and PWC function $\alpha_1$
- solves a single LP at each stage

HSVI-based Online Minimax Strategies for Partially Observable Stochastic Games with Neural Perception Mechanisms, In *Proc* L4DC 2024

# Pursuer–evader with neural perception

Precise NN

Coarse NN




Strategies for pursuer (red dot) shown as pink arrows

Pursuer's percept is blue border, pursuer's belief of evader's position in green

(Modified variant of Horak 2023, with added NNs)

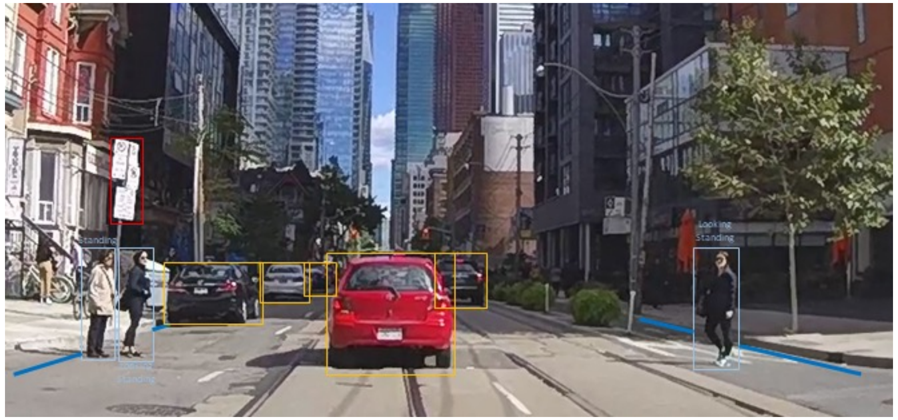
# Conclusions

- Presented advances in formulating neuro-symbolic games
  - neural and symbolic components loosely coupled, constrained interface
  - … but able to work with and evaluate exactly learnt percepts
  - high computational complexity, but compare to NN training…
  - restricted game variants
  - … but realistic case studies

- Challenges & directions
  - beyond rationality, towards psychological games
  - characterisation of the polyhedral abstraction
  - scalability, e.g., abstraction, pre-image approximation
  - smarter algorithms, combine with sampling
  - probabilistic perception, e.g. via Bayesian neural networks
  - further applications and case studies

# Perspective on challenges for neuro-symbolic AI

- Despite the challenges, need rigorous verification methodologies for AI
- Greater emphasis on safety, not just performance



- To make progress:
- Neuro-symbolic modelling: rigorous verification methodologies for a broader class of systems integrating neural and symbolic components
- Compositionality: formulate a compositional framework based on assume-guarantee interfaces
- Reasoning vs statistics: neural network learn statistical features of input data and lack (deductive) reasoning ability, need smart integration of the two

# Acknowledgements


- My group and collaborators in this work

- Project funding
  - ERC Advanced Grant fun2model
  - EPSRC project FAIR: Framework for responsible adoption of artificial intelligence in the financial services industry, https://www.turing.ac.uk/research/research-projects/project-fair-framework-responsible-adoption-artificial-intelligence
  - ELSA European Lighthouse on Secure and Safe AI, https://www.elsa-ai.eu/
  - UKRI AI Hub on Mathematical foundations of intelligence: an ‘Erlangen Programme’ for AI


- See also
  - PRISM www.prismmodelchecker.org