Shield Synthesis for AI

Bettina Könighofer
Roderick Bloem

Ufuk Topcu
Scott Niekum
Mohammed Alshiek
Suda Bharadwaj

Rüdiger Ehlers
Universität Bremen

Sebastian Junges

Rayna Dimitrova

Thomas Henzinger
Guy Avni
Krishnendu Chatterjee

Nils Jansen

IST Austria

Radboud Universiteit Nijmegen

The University of Texas at Austin

University of Leicester

Secure & Correct Systems

www.iaik.tugraz.at
Reinforcement Learning

C \models \phi

Synthesis via Game Solving

Your Controller

Your Specification

Infinitely often, visit R and S. If S is blocked, go to C. Resume visiting R and S once S is unblocked.

\[ G(\neg \text{blocked} \rightarrow FR) \land G(\neg \text{blocked} \rightarrow FS) \land G(\text{blocked} \rightarrow X(C \cup \neg \text{blocked})) \]
Your Controller
- Large
- Complicated
- Highly optimized
- Many sensors
- ...

Your Specification
- Large
- Hard to write
- Greyscale
Reinforcement Learning

Environment \[\xrightarrow{\text{state}}\] Reward \[\xrightarrow{\text{reward}}\] Learning Agent

\[\xrightarrow{\text{action}}\]
Reinforcement Learning

Correctness Guarantees

Optimality

How?
Your Controller
- Large
- Complicated
- Highly optimized
- Many sensors
- …

Your Specification
- Large
- Hard to write
- Greyscale
Reinforcement Learning

• Critical aspects only
• Small & sweet

Correctness Guarantees

• Large
• Complicated
• Highly optimized
• Many sensors
• …

Optimality

Your Controller

Shielding

Critical Spec

• Critical aspects only
• Small & sweet
Preemptive Shielding

Environment → Learning Agent

Shield

Set of safe actions

state

reward

safe action

Minimal Interference
Post-Posed Shielding

Policy Update:

- for \texttt{safe\_action} using \texttt{reward}
- for \texttt{action} if \texttt{action} \neq \texttt{safe\_action}:
  1. Assign a punishment to \texttt{action}
  2. Assign \texttt{reward} to \texttt{action}

Shield can be added in execution phase
A Shield for PAC-MAN

M. Alshiekh, R. Bloem, R. Ehlers, B. Könighofer, S. Niekum, U. Topcu:
Safe Reinforcement Learning via Shielding. AAAI 2018
Outline

- Safety Shields
  - AAAI-18
- Optimal Shields
  - submission
- Safety Shields for Multi-Agent Systems
  - ACC-19
- Probabilistic Safety Shields
  - arXiv
Optimal Shields

Problems of learned controllers

- (Safety problems)

1. Difficult to add new features
2. Poor performance on un-trained behavior
3. No local fairness

Solution: Optimal Shield
Shields for Traffic Light Controllers

Learned Controller: “minimize total waiting time”

1. Difficult to add **new features**
   - priority to public transport, changes due to an accident

2. **Poor performance** on un-trained behavior
   - Uniform traffic congestion meets rush-hour traffic

3. No local fairness
   - Farm road never gets green
Optimal Shields Synthesis

- Lightweight shields → Two cost functions
  - $c_{BEH}$: Cost for behavior
  - $c_{INT}$: Cost for interference

\[ \lambda \cdot c_{BEH} + (1 - \lambda) \cdot c_{INT} \]

- Mean-Payoff Game with 2 Objectives
- Mean-Payoff Game

\[ \lambda: \text{tradeoff between objective of controller vs shield} \]
Dealing with rush-hour traffic

Controller

- Deep Convolutional Q-Network
  - 16 dim input vector
  - num approaching cars, waiting time
  - 4 layers (16, 604, 604, 4 nodes),
  - Q-learning: $\alpha = 0.001, \gamma = 0.95$

- “Minimize waiting time of two junctions”

Shield

- $c_{BEH}$: size of maximal queue
- $c_{INT}$: 1 for interference, 0 otherwise

abstract state

$(1, 8, 1, 2)$
Dealing with \textit{rush-hour traffic}

Outline

- Safety Shields

- Optimal Shields

- Safety Shields for Multi-Agent Systems

- Probabilistic Safety Shields
Safety Shields for Multi-Agent Systems

- Task: Enforce global safety property

1. Quantitative interference costs $c_{INT}$:
   - Counting cost function
   - Different costs for interferences with different agents

2. Fair Shielding
   - Do not always interfere with the same agent repeatedly
Case Study: Warehouse

- Package drop off
- Package pick up locations
- Charging station
Case Study: Warehouse

Outline

- Safety Shields [AAAI-18]
- Optimal Shields [submission]
- Safety Shields for Multi-Agent Systems [ACC-19]
- Probabilistic Safety Shields [arXiv]
Shielding original Pacman?

- State space is huge!
- Not realizable!
Learning the Adversary Model

- Each ghost has its individual behaviour
  - Observe it, model the behaviour
  - Data augmentation techniques
    - Is PAC-MAN north, south, east, or west?

- Results in MDP of environment
- Guaranteed safety w.r.t. \textit{probabilistic} temporal logic spec
MDP is huge! Scalability

- Finite Horizon
  - safety for finite number of steps
  - infinite horizon may cause large errors anyways

- Piecewise Construction
  - compute shield for each state independently
MDP is huge! Scalability

- Independent Agents
  - crashing probabilities for different agents are stochastically independent
  - compute individually, compose shields

- Abstractions
  - adversaries may be far away
  - neglect adversary positions that are not relevant
Probabilistic Safety Shield for Pacman

N. Jansen, B. Könighofer, S. Junges, and R. Bloem:
Shielded Decision-Making in MDPs, arXiv
Future Work

- Safety Shields
  Shields for CPS, Deal with wrong models

- Optimal Shields
  Performance in autonomous systems

- Safety Shields for Multi-Agent Systems
  Distributed Shield Synthesis

- Probabilistic Safety Shields
  Partially observable MDPs
Bettina Könighofer  
Shield Synthesis for AI  

THANK YOU!

LiVe @ ETAPS, Prague  
April 6, 2019