Carnegie Mellon University

Combining Reinforcement Learning and a Safe Controller towards Optimization and Safety

Samar Rahmouni
Prof. Giselle Reis

Senior Thesis 2021-22

Presentation Overview

- Problem Statement
  - Significance
- Proposed Solution - Different Approaches to Reinforcement Learning
- Hybrid Architecture for Shielding
- Scenario
  - Car Platooning
  - Model in Reinforcement Learning vs. the Safe Controller

Carnegie Mellon University

Carnegie Mellon University

Optimization vs Safety in Reinforcement Learning

Problem Statement

Reinforcement Learning in a Nutshell

- Trial-and-Error
- Exploration vs Exploitation
- Pretty successful overall, think AlphaGo, Deepmind AlphaStar and OpenAI.

Just not really deployable in real-life for now...

Carnegie Mellon University

A Real-life Problem: Vehicle Platooning

- Multiple vehicles following each other, i.e. leader and followers.
- Aims to reduce the distance between them
- Taking less space on the road
- Allowing more vehicles to occupy highways
- Let's solve traffic.

Optimization vs Safety

Safe Controller (SC) [1]

- In Maude
- Formally verified
- Does not optimize the final answer, though is able to return a safe range of velocities
- Guarantees a 100% no crashes

Reinforcement Learning (RL)

- In Python
- Shown to converge to an optimal solution
- Guarantees a 100% a crash (maybe multiple)

Mix and match?

1. Yuri Gil Dantas, Vivek Nigam, and Carolyn Talcott. A formal security assessment framework for cooperative adaptive cruise control. In *2020 IEEE Vehicular Networking Conference (VNC)*

Carnegie Mellon University

Carnegie Mellon University

Three Different Approaches to RL+SC

Proposed Solution

Proposed Approaches

Tabular Q-learning

- No safety guarantees
- Computationally faster
- Known to converge towards optimal policy

Shielding in RL using a SC

- Incorporates a safe controller into a RL architecture
- SC computes a set of safe actions given current state
- RL does not need to learn how to continue “living”, just how to optimize

Logic-Based Inference RL

- Only investigated in deterministic environments
- Learning inference rules
- Uses an inductive reasoning approach to incorporate rule knowledge into decision-making

These different approaches are to be tested on

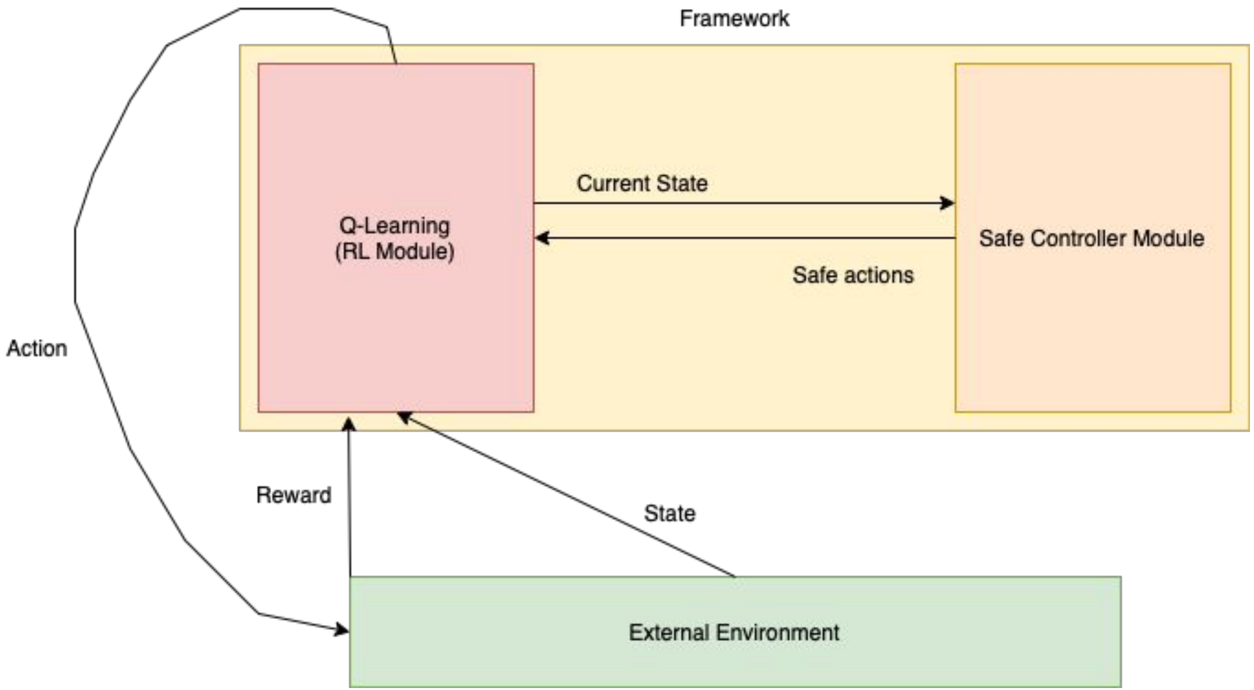
Safety vs. Optimization [includes speed]

Carnegie Mellon University

Hybrid Architecture [Shielding]

Proposed Architecture

Hybrid Architecture for Shielding



- Independent components
- Focus on the framework that translates between both implementations
- First step: formalize scenario

Carnegie Mellon University

Car Platooning Model in RL vs the SC

Scenario

Car Platooning Scenario

- Two vehicles for simplicity, one leader and one follower.
- They are both moving on one line (x-axis)
- Each have the choice to accelerate or decelerate.

Important goals:

1. **Don't crash.**
2. Minimize the distance between the vehicles.

There's two ways the vehicles can crash, either by (1) hitting each other or (2) running a red light/stop sign.

Let's look at how this model is formalized in RL vs. a SC.

Carnegie Mellon University

Formalization of the Scenario

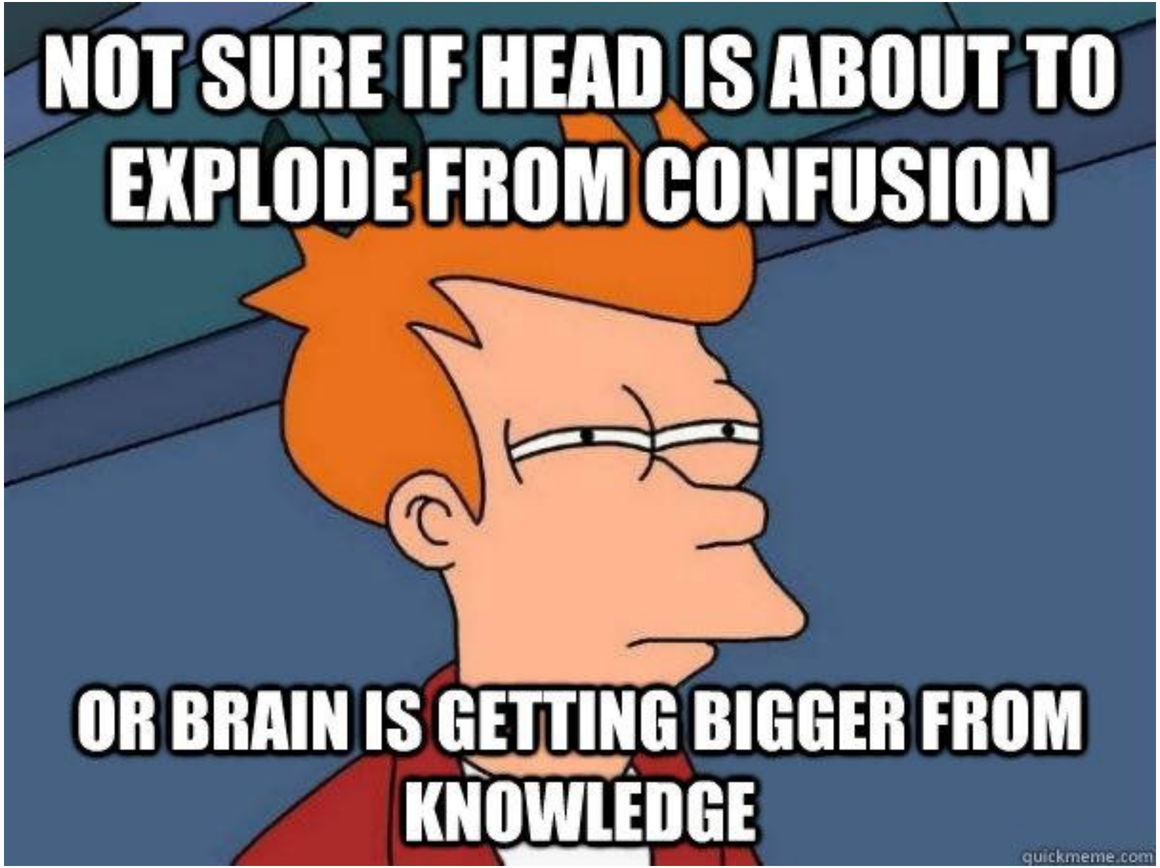
Model in Reinforcement Learning

- S - A state vector to denote *time, position, velocities.*
- A - An action vector for possible accelerations.
- R - A reward function (i.e. given the distance between the vehicles.
- φ - A transition function s.t. $\varphi(s_t,a_t) = s_{t+1}$
- γ - A discounting factor in [0,1]

Model in the Safe Controller

- Local Knowledge Base
  - Set of grounded facts p@t
- Events = ev@t
  - Equivalent to task@0
- Executable semantics
- System configuration search to ensure that no path given those accelerations result in a crash.

Next-time: Translation between model in RL into/out model in SC



Le moi for the past three months