Safe & Adaptable Vehicle Navigation using Reinforcement Learning and Optimal Control

Jatin Sikka

Advisor: Dr. Donggun Lee

[1]Department of Mechanical and Aerospace Engineering, North Carolina State University

Objectives

The aim of this research is to implement and analyze safe and adaptive navigation with a focus on applications in robotics and AI. We explore two distinct methodologies, with future plans to combine them to solve the optimal control problem efficiently and in real time.

1. The **Hopf-Lax Formula**, a sophisticated tool for open-loop control adaption, enabling swift response to dynamic environmental conditions.

2. The **Trust Region Policy Optimization (TRPO)**, a machine learning-based algorithm tailored for crafting safe navigation policies.

Introduction

Background: The real world presents unpredictability and complexity in solving *vehicle navigation problems*. This semester, we are pivoting our research to tackle these issues by integrating advanced machine learning techniques with real-time open-loop control methods.

ML-based control method: Trust Region Policy Optimization (*TRPO*) represents a significant advancement in reinforcement learning, targeting robust policy optimization. Introduces a novel approach for ensuring monotonic policy improvement, distinguishing it from conventional methods [3].

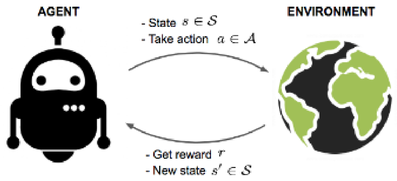

Fig. 1: Reinforcement learning visualization [4]

Limitation of TRPO:

- TRPO's inflexibility to unexpected obstacle behavior requries computationally demanding retraining, which may not be feasible in dynamic environments.
- Real-time policy retraining with TRPO is impractical due to its computational complexity.

Our real-time control solution: *Hopf-Lax formula*, is a sophisticated numerical approach for solving real-time open loop control problems. For real-time computation, Hopf-Lax theory can incorporate various computationally efficient convex-programming methods and approximation techniques, including receding horizon [1].

Applications: Manufacturing automation, underwater vehicles, spacecrafts, satellites, aerial vehicles, air taxis, safety controls, walking robots and more.

References

1. *Donggun Lee, "Safety-Guaranteed Autonomy under Uncertainty", University of Californica, Berkeley, 2022*
2. *J. Darbon and S. Osher, "Algorithms for overcoming the curse of dimensionality for certain hamilton–jacobi equations arising in control theory and elsewhere," Research in the Mathematical Sciences, vol. 3, no. 1, pp. 1–26, 2016.*
3. *Schulman, J., Levine, S., Moritz, P., Jordan, M., & Abbeel, P. (2017). Trust Region Policy Optimization. arXiv preprint arXiv:1502.05477v5 [cs.LG]. University of California, Berkeley, Department of Electrical Engineering and Computer Sciences.*
4. *Gerand Maggiolino, "Creating OpenAI Gym Environments with PyBullet", 2020*
5. *Lilian Weng, "A (Long) Peek into Reinforcement Learning", 2018*

Vehicle Navigation using TRPO: Approach

Agent/Algorithm Framework: Trust Region Policy Optimization (TRPO) agent, leverages a neural network policy model that interprets the simulation state to decide on the best actions. To create a better policy TRPO safely updates the policy based on the experiences from past interactions with the environment.

$$\text{Policy update} \qquad \Pi_{i+1} = \arg\max_{\pi}\left[L_{\pi_i}(\pi) - CD_{\mathrm{KL}}^{\max}(\pi_i, \pi)\right]$$

Environment Setup:

- Custom environment (car, goal, obstacles) in Pybullet with OpenAI Gym
- 2-degree action (acceleration & steering)
- 8-degree observation space

Positive Reinforcement (Rewarding):

- Reaching the goal
- Proximity to the goal
- Optimal path selection (shortest path)

Negative Reinforcement (Penalizing):

- Collision with an obstacle
- Unsafe proximity to obstacle

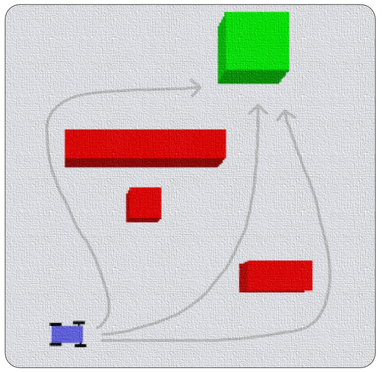

Fig. 2: Visualization of the problem to optimize, along with the 3 basic approaches the agent might take

Vehicle Navigation using TRPO: Results

Agent Training:

- Epoch (# of iterations): 2,000
- Batch size: 12,000

Training Outcome:

We saw the rewards boosting from low 1000's at the start of the training to mid 5000's at the end of the training underscoring the effectiveness of our TRPO agent.

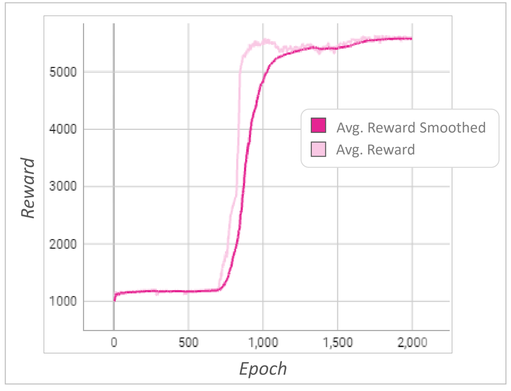

Fig. 3: Average reward over 2000 iterations

Simulation:

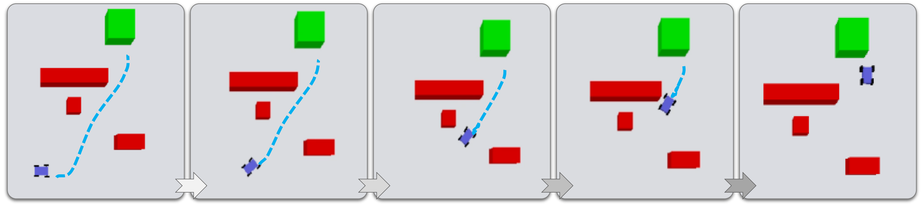

Fig. 4: Simulation results following the training

Real-Time Control using Hopf Formula: Approach

Hopf formula for linear system:

$$\max_{p}\left[-g^*(p) + \int_0^{T-t} H(s,p)\mathrm{d}s + x \cdot p\right]$$

is the optimal value to be maximized, $g^*: R^n \to R\ U\ \{+\infty\}$ is the Fenchel-Legendre transform of a convex function and T is the terminal time.

Real-time computation: Hopf-Lax incorporate various computationally efficient convex programming methods and approximation techniques, including receding horizon [1].

Baseline method: Direct optimization method

Real-Time Control using Hopf Formula: Results

Significant Computational Speed Advantage: Demonstrated a significant advantage in computation speed of approx. 18 times when employing the Hopf formula compared to traditional direct optimization.

	Initial State	Direct Method	Hopf-Lax
1.	1	0.00427	0.00075
2.	142	0.03056	0.00099
3.	4895	0.02011	0.00089

Table 1: Time taken for varying initial state

Well-Suited for Complex Problems: Not only does the Hopf-Lax consistently outpace the direct method, but it also exhibits a growing advantage as initial conditions become more complex.

Conclusions

In the research we have explored 2 different methods for solving optimal control problems, Hopf-Lax Formula and a machine learning based algorithm TRPO.

Contributions:

- Designed a simulation environment to analyze the safe navigation problem
- Extended TRPO (ML-based method) to update a safe navigation policy.
- Implemented the Hopf formula to get an open-loop control adaption to the environment in real-time
- Validated significant reduction in computational time using the Hopf formula and showed its efficiency, compared to the baseline method.

Broad Impact: TRPO and Hopf formula are not just theoretical robustness but also in practical computational applications. The significant reduction underscores its potential utility in fields demanding quick, real-time solutions.

Future work:

- Test Adaptability of Hopf formula by adding uncertainties in the environment.
- Extend Hopf formula for nonlinear systems
- Explore and utilize machine learning algorithms to a wider range of optimal control problems, and practical testing of these methods.