

M Coping with Complex Games using Machine Learning M

Pedro d'Aquino

Computer Science & Engineering

Shiva Ghose

Robotics and Autonomous Vehicles

Keegan R. Kinkade

Robotics and Autonomous Vehicles

Motivation

Robocode is an open source, dynamic, deterministic tank battling simulator implemented in Java. Tanks incorporate:

- Differentially Driven Movement
- Mounted Enemy Scanning Radar
- Mounted Rotating Gun Turret

The environment has provided a powerful base upon which artificial intelligence and machine learning methods can be built up, taught and tested.

Problem Statement

The purpose of the project was to incorporate machine learning algorithms in a Robocode agent in order to assist the decision making process while attempting to defeat opponents. Specifically, we focused on incorporating machine learning in two areas:

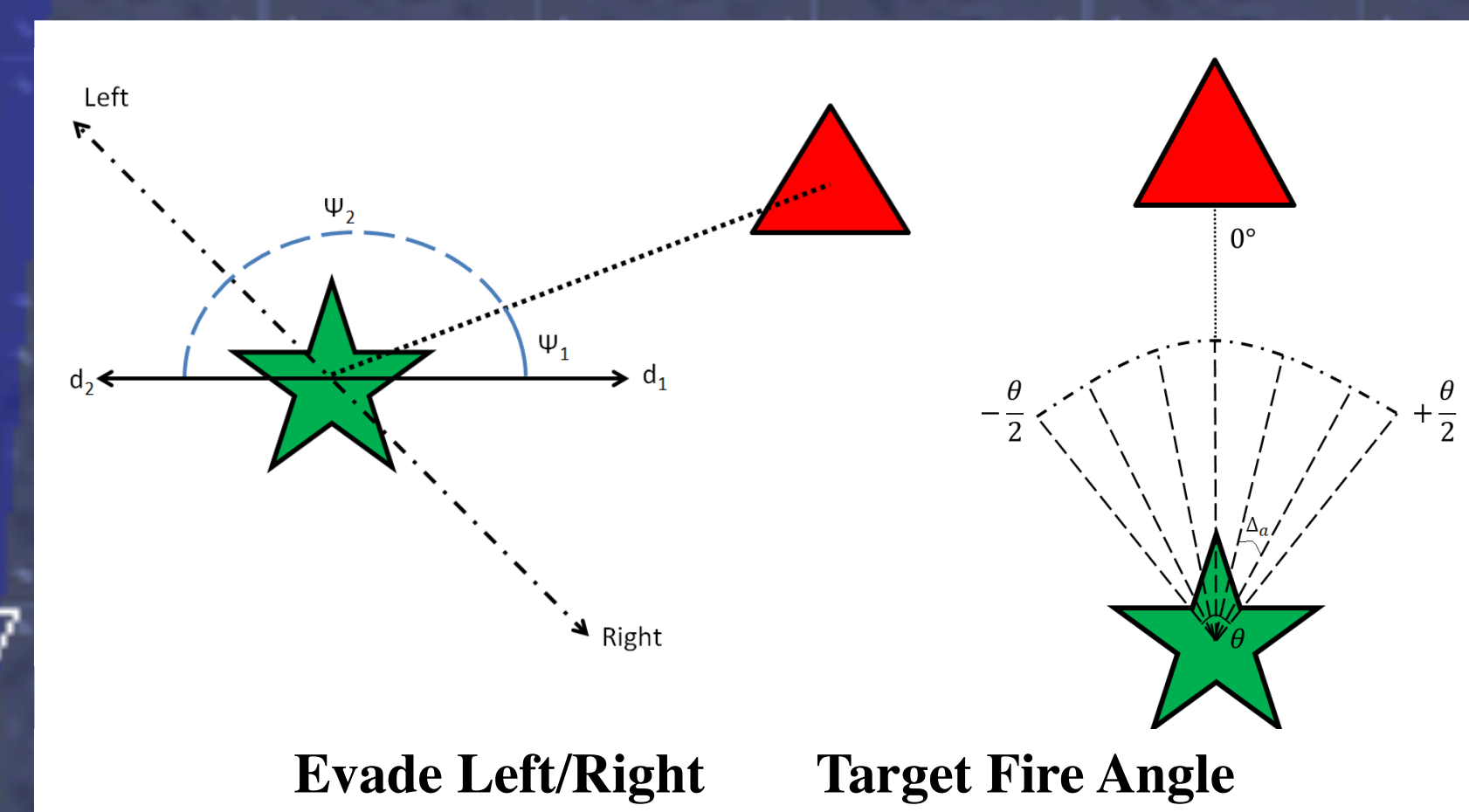
- Evasion
- Targeting

To simplify the models, these areas were considered separately. With respect to evasion, the goal was to evade incoming bullets in order to minimize the damage inflicted by opponents. Conversely, the goal of targeting was to fire bullets in a manner that maximizes the damage inflicted upon opponents.

Contributions

Much research has been done to optimize agents using both neural networks as well as genetic programming. However, little work has been done to incorporate sophisticated machine learning techniques designed to better inform Robocode agents when making decisions. Thus, our work presents novel applications of Support Vector Machines, as well as Temporal Difference Learning, in order to design competitive Robocode agents.

Approach



EVASION: Four evasion movements were created (dodge left/right, feign, halt). Data was collected by randomly applying a strategy. SVMs were trained for each strategy using probability models.

TARGETING: Using a discrete firing angle as an additional state feature, a single probability SVM was trained. In addition, Q-Learning was implemented for targeting.

Results

