Jagunmolu B Oke

[Email address]

Abstract

[Draw your reader in with an engaging abstract. It is typically a short summary of the document.   
When you’re ready to add your content, just click here and start typing.]

Racing simulation (driver behaviour)

Artificial Intelligence for Games

Table of Contents

[Project Overview 2](#_Toc154827844)

[AI-Driven Behaviour Characteristics 2](#_Toc154827845)

[Car Model 2](#_Toc154827846)

[Decision Making 3](#_Toc154827847)

[Finite State Machine 3](#_Toc154827848)

[Awareness 3](#_Toc154827849)

[Fuzzy Logic and Set 3](#_Toc154827850)

[Behaviour Tree 3](#_Toc154827851)

[Obstacle Avoidance 4](#_Toc154827852)

[Fuzzy Logic and Set 5](#_Toc154827853)

[Pathfinding 6](#_Toc154827854)

# Project Overview

This project is about racing cars, where AI embodies these car model and have their own unique behaviours for analysis and understanding the play space and make swift decision in a fun and playful manner. These behaviours include and not limited to Acceleration and Breaking, Environmental Awareness, Turning behaviour, Offensive behaviour, Defensive behaviour, avoidance behaviour and reactive behaviours.

This project aims to create an AI-driven behaviour model for a racing simulation using Unity. These would be achieved using Applied mathematics (majorly Vector calculation, mathematical functions) and AI techniques (revolving around decision making then pathfinding and Autonomous Agents) to model these systems.

## AI-Driven Behaviour Characteristics

Primarily these would include two unique AI-driven behaviours:

1. Aggressive Racer: This is a behaviour considered reckless behaviour where the driver takes more risks. Below is a list of characteristics:
   1. Constantly seeks to overtake.
   2. Aggressive in manoeuvres.
   3. Constant pressure on an opponent (forcing them to error and close proximity to the opponent).
   4. If behind, constantly aiming to catch up by aggressive overtaking and risky moves.
2. Cautious Racer: This is a behaviour considered defensive behaviour where the driver takes less risks. Below is a list of characteristics:
   1. Avoids risky overtaking.
   2. Avoids collision.
   3. Priorities a safe proximity to opponents.
   4. Sacrificing speed (Brakes early to Corners).
   5. Adjust speed based on potential collision and unpredictable opponent behaviour.

## Car Model

The car used in this project is a basic car model based on unity physics wheel collider neglecting other vehicle physics dynamics.

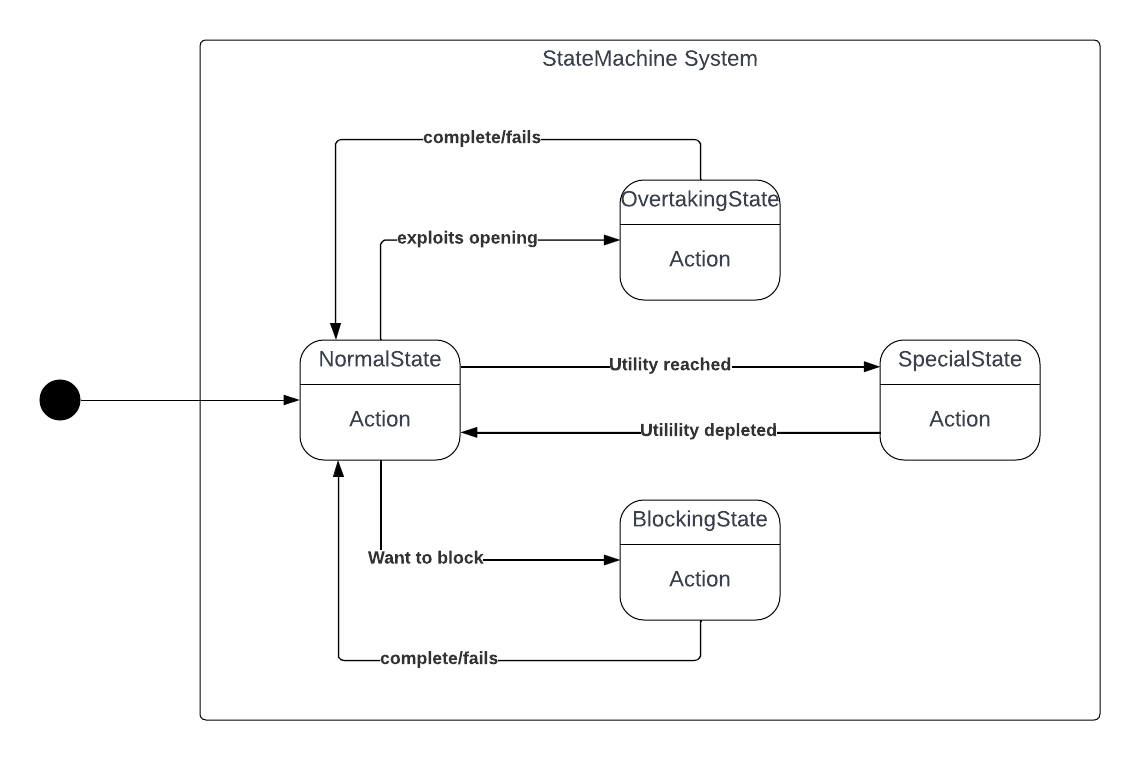
# Decision Making

Decision-making is primarily the backbone of creating a believable AI model behaviour, where the model can act and react based on its internal or external knowledge which the process in turn could affect the model's internal state and external environment state.

## Finite State Machine

The Finite State Machine (FSM) is the abstraction of multiple components of a system to a finite number of states, where only one can exist at a time and transitions are made between those states based on predetermined conditions.

FSM is used to model and manage the behaviours of the drivers; the following is an illustration of the use of FSM for a basic driver model.

 The fig above shown the UML diagram of the state interaction for a basic driver model, when conditions are check using current gameplay situations with Boolean logic to decide a condition is meant to transition between state. But Special State use utility values to decide switch for example the Aggressive Racer special state is Aggressive state ranging from 0 – 10. Which increases or decreases over time based on the game state. If a threshold is meet it switch to the state or out of the state. Below is the table illustration of the model.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Opening (OnStart) | Special Utility < Threshold | Special Utility > Threshold | Potential Block (OnStart) | OnComplete |
| Normal | Overtake | - | Special | Block | - |
| Overtake |  | - | Special |  | Normal |
| Block |  | - | Special | - | Normal |
| Special | Overtake (Possible Ram) | Normal | - | Block (Ram) | - |

Using FSM to model behaviours in games could get a bit complicated if the system being handled gets larger and more complex. Switching from one state to another, having multiple conditions to switch to another state. For a racing AI system, behaviours that are not large FSM will suffice to represent those behaviours (Tomlinson & Melder, 2014).

Overtaking state, a trail for ghost (potential predicted position), Dot product is used to calucte is car is ahead or behind positions

## Awareness

## Fuzzy Logic and Set

## Behaviour Tree

# Obstacle Avoidance

Talk about whiskeys

# Fuzzy Logic and Set

# Pathfinding

This week (18/12 – 24/12) – Complete a refactorize FSM, with some Obstacle avoidance and (Utility AI / fuzzy logic and set).

Next week (25/12 – 31/12)– BT, with some Obstacle avoidance and Utility AI (if possible)

Week After the next week (01/01 – 07/01)– if not (fully integrate FSM together BT), then polish with Utility AI and better pathfinding. Look into PID or Rule Based System (if possible)

(08/01 – 10/01) – Extra stuffs