A Fuzzy Approach For The CEC2007 Simulated Racing Car Competition

Duc Thang Ho and Jon Garibaldi

Intelligent Modelling and Analysis Research Group, School of Computer Science, University of Nottingham, UK

May 5, 2009

Outline

1. Background
2. Experimental setup
3. The controllers
4. Results & conclusion

Outline

1. Background
2. Experimental setup
3. The controllers
4. Results & conclusion

Car racing simulations

The University of Nottingham

Car racing in games

- Developing software to control a real car is very challenging and expensive
- Developing controller for a car racing simulation is also interesting and challenging in its own right
- Examples of simulations: The Open Racing Car Simulator(TORCS), Robot Auto Racing Simulator(RARS), etc.

world-changing research from The University of Nottingham

The controller architectures

The University of Nottingham

The perception-decision-action paradigm

Three main functions are to be found in any control architecture: perception, decision, action. It is possible to differentiate between:

- State-based approaches: complex models of the car and the environment are known or can be built from sensory data.
- Action-based approaches: favor reactivity. Actions follow perception closely, almost like a reflex. Suitable to deal with unexpected events.
- Hybrid approaches: combine both the advantages of these two aforementioned approaches

Our approach

The University of Nottingham

The perception-decision-action paradigm

- Fuzzy approach: allows approximate reasoning and a human-like description of the car's reactive behaviours
- Path planning: only happens in specific circumstances

world-changing research from The University of Nottingham

Outline

1. Background
2. Experimental setup
3. The controllers
4. Results & conclusion

The rules

Rules

- Based on the CEC2007 car racing competion organised by Dr Julian Togelius and Dr. Simon M. Lucas (Essex University,UK) at the CEC2007 conference
- The car competes against an opponent car
- Collecting points at waypoints
- Only the first car to reach the waypoint gets the point
- Points must be collected in strict order
- A car may choose to miss a point

Screenshot

Rules

- Only next two waypoints are shown to cars at each point in time
- Waypoints are generated randomly
- As soon as one is reached, another is generated

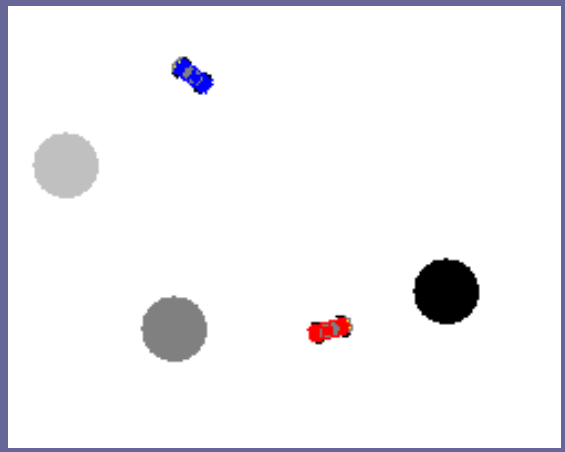

Figure: Screenshot

The Physics

The University of Nottingham

The physics

- The dynamics of the car are reasonably realistic
- Allows skidding and car-to-car collisions
- Noise is added to observations and actions

The Inputs

The inputs

- Velocity, orientation, position of the car and its opponent
- Positions of visible waypoints

The Outputs

The outputs

- back
- back-right
- back-left
- neutral
- left
- right
- forward
- forward-left
- forward-right

Competition

The University of Nottingham

How good is a controller

- All played each other in a round-robin league
- 1000 steps each match in the total of 500 matches
- The winner is the one having the most points

world-changing research from The University of Nottingham

SAMPLE RUN

Outline

1. Background
2. Experimental setup
3. The controllers
4. Results & conclusion

Overall design

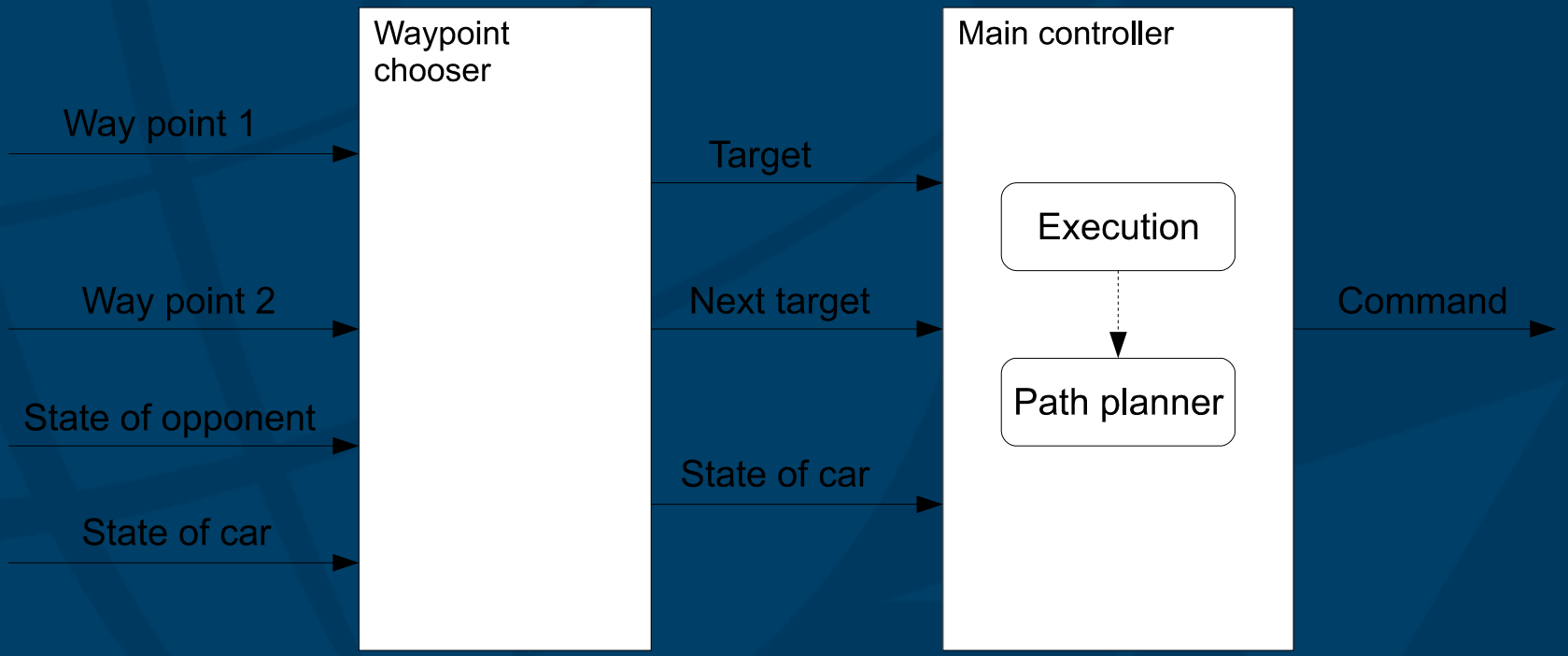

Figure: Control Architecture

EXECUTION CONTROL

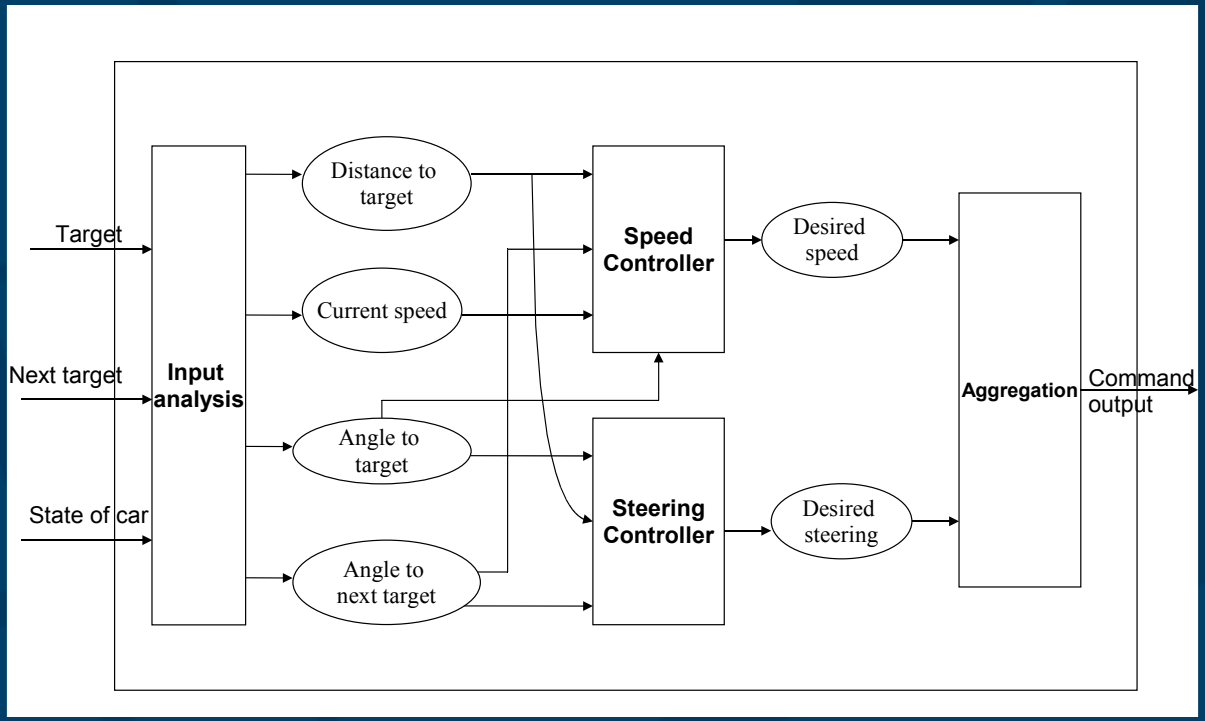

FIGURE: Fuzzy controllers

FUZZY SETS

- Fuzzy sets have been introduced by Lotfi A. Zadeh (1965) as an extension of the classical notion of set
- Fuzzy sets: are sets whose elements have degrees of membership.
- Fuzzy variables: are variables which associate with a set of fuzzy sets

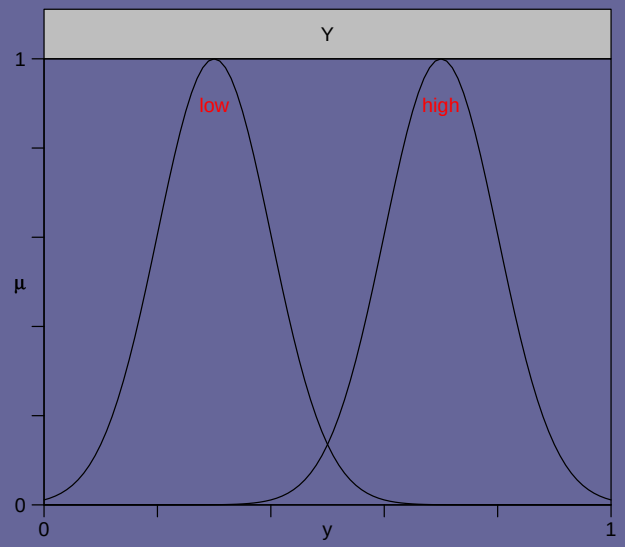

FIGURE: Fuzzy sets

Fuzzy controller

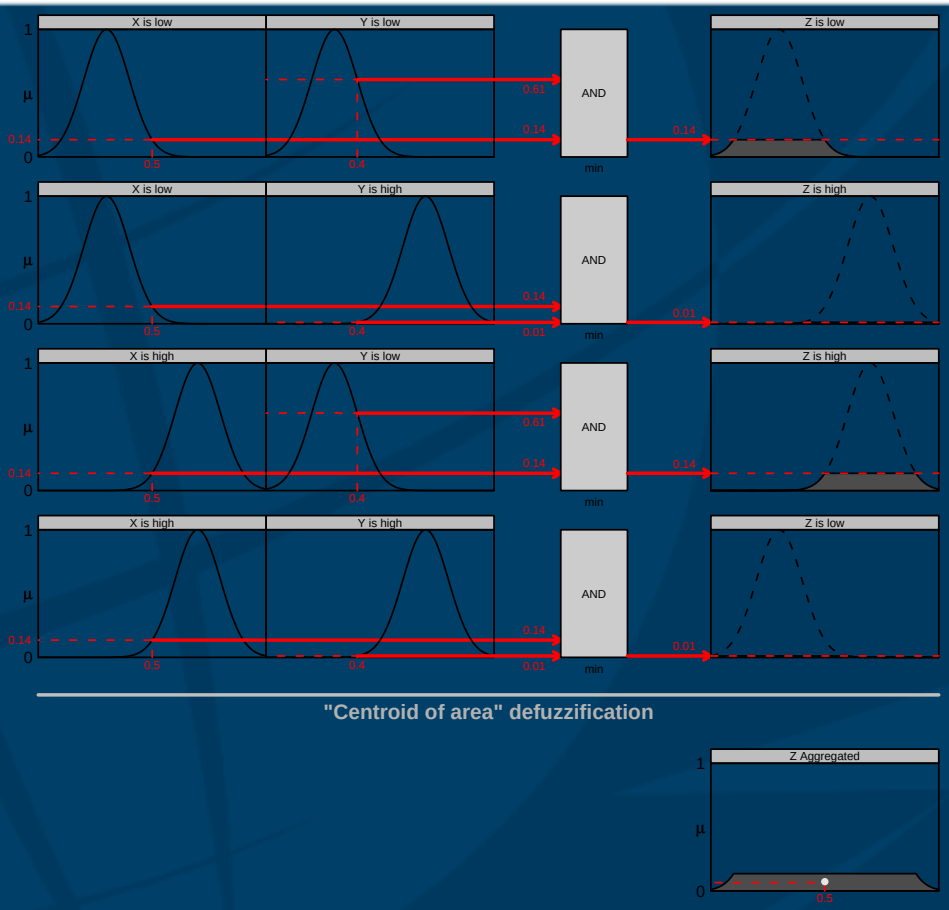

Figure: Fuzzy controllers

The University of Nottingham

THE SPEED CONTROLLER

- Four input variables: distance, speed, heading angle and next angle and one output variable: expectedspeed and 10 rules
- The car goes forward most of the time (positive speed). It should only try to reverse in the circumstance that the car is moving at a slow speed away from the target and the distance to the target is not far away.
- Once going backward (negative speed), the car will maintain its direction until the target is in front of the car and at a reasonably far distance.
- The speed of the car is always adjusted to be proportional to the distance to the target.

THE STEER CONTROLLER

- 9 rules, 3 inputs, one ouput
- The steering controller always turns the car towards the target until the angle towards the target is within a small tolerance
  - If the distance to the target is too near or the car is at a small angle away from the next target, keep neutral steering.
  - Otherwise, the car will deflect itself from the next target by a small angle.

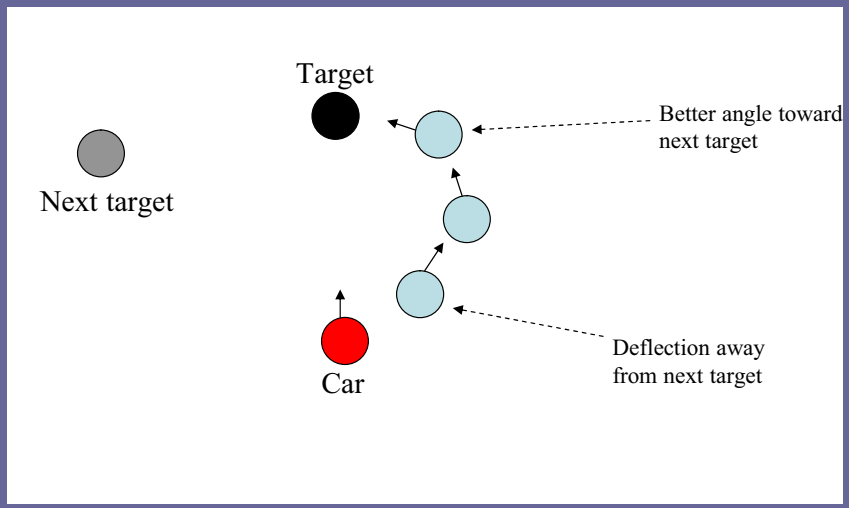

FIGURE: Steering controller

Sample solo run

Outline

1. Background
2. Experimental setup
3. The controllers
4. Results & conclusion

RESULTS

TABLE: Top four competitors and their methods

Competitors	Methods	Training methods
Ho&Garibaldi	Fuzzy controller	-
Tomoharu	Neural networks	Reinforcement learning
Chin Hong	Magnetic force field	GA
Binatix, Inc.	Feed-forward neural net	Td-learning

RESULTS(2)

TABLE: Solo Score

Controller	Solo Score
Ho & Garibaldi	24.4
Binatix, Inc.	21.03
Chin Hiong	20.07
Tomoharu	19.2

RESULTS(3)

TABLE: Head-to-head competition results

	Tomoharu	Chin Hiong	Binatix, Inc
Ho&Garibaldi vs	17.6/11.2	19.2/19.1	18.3/17.5
Tomoharu vs	-	13.0/18.8	13.7/18.5
Chin Hiong vs		-	19.4/19.4
Binatix, Inc vs			-

The University of Nottingham

Conclusion & Future work

- The proposed approach does not follow the classical approach of planning the optimal path first and then follow the path as closely as possible.
- It was built upon underlying fuzzy controllers that are able to react in real-time based on the current states of the car without referring to any pre-computed path.
- Only in some particular situation where path planning could be quickly performed do we pre-compute the optimal trajectory and follow it as in the classical approach
- It is hoped that the methods presented in this paper is applicable in a more complex environment such as the Robot Auto-Racing Simulator, The Open Racing Car Simulator (TORCS), etc.