Application of Fuzzy Sets to Autonomous Robot Vehicle Navigation Using the PIC18F452 Microcontroller

by
Robert Casey and Mike Hensler

MAT362
Summer 2006
DigiPen Institute of Technology

Introduction

In CS105 (Computer Environment II):

- Students programmed RC cars and trucks to obtain autonomous vehicle navigation.
- The intelligent use of ***sensor input*** was a key factor in obstacle avoidance.

One potential driving algorithm:

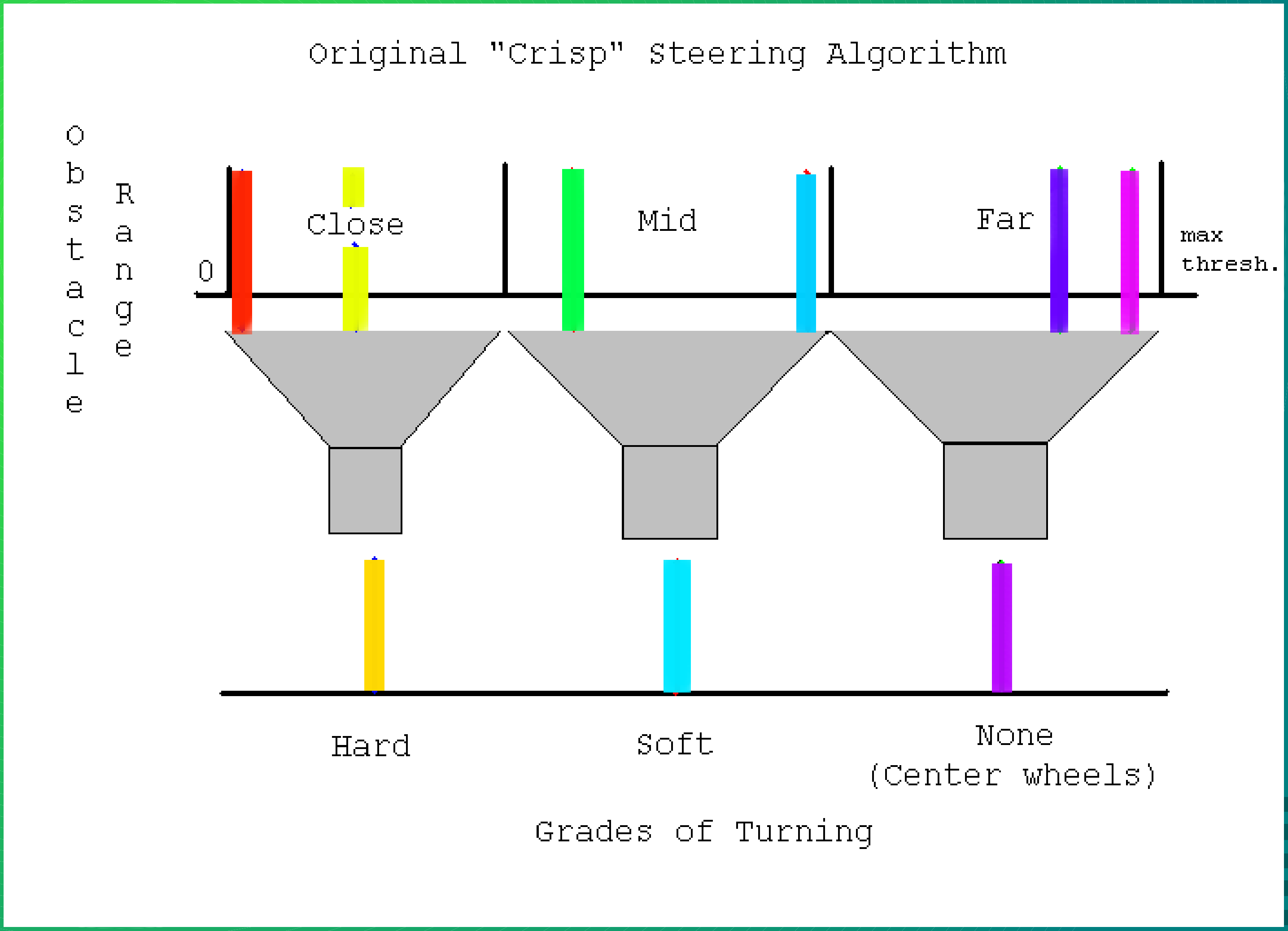
- Set a few discrete thresholds for sensor ranges which then map to steering output voltages (resulting in an angle of turning).

RESULT:

- Simple, effective, and somewhat jerky driving.

Introduction

The former algorithm mapped all sensor data to just three singletons...resulting in a loss of **information** and **control**.



An experiment...

Question:

Could fuzzy sets and/or logic be used to improve the results of this "crisp" driving algorithm?

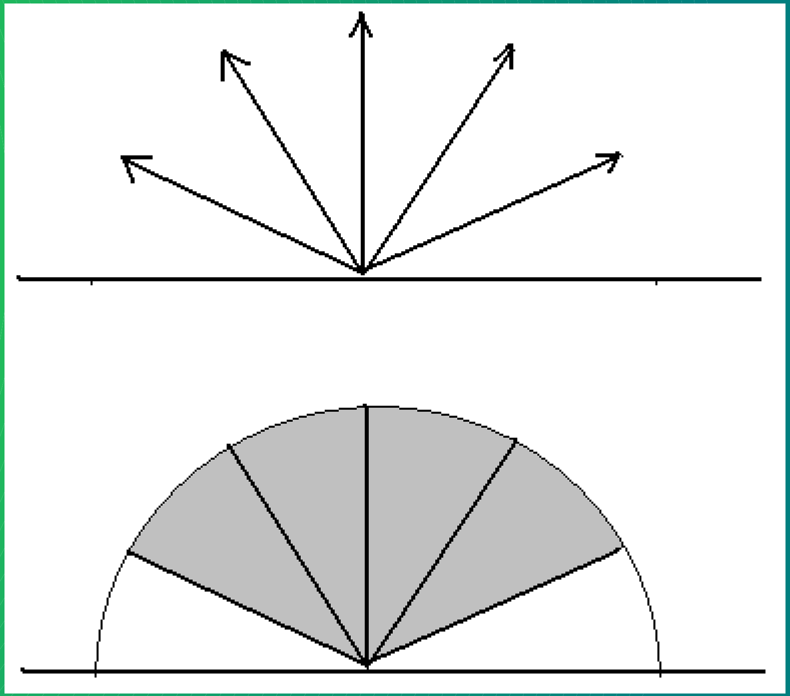
...after some work...

Project Goals

Using the linguistic expressiveness of fuzzy sets:

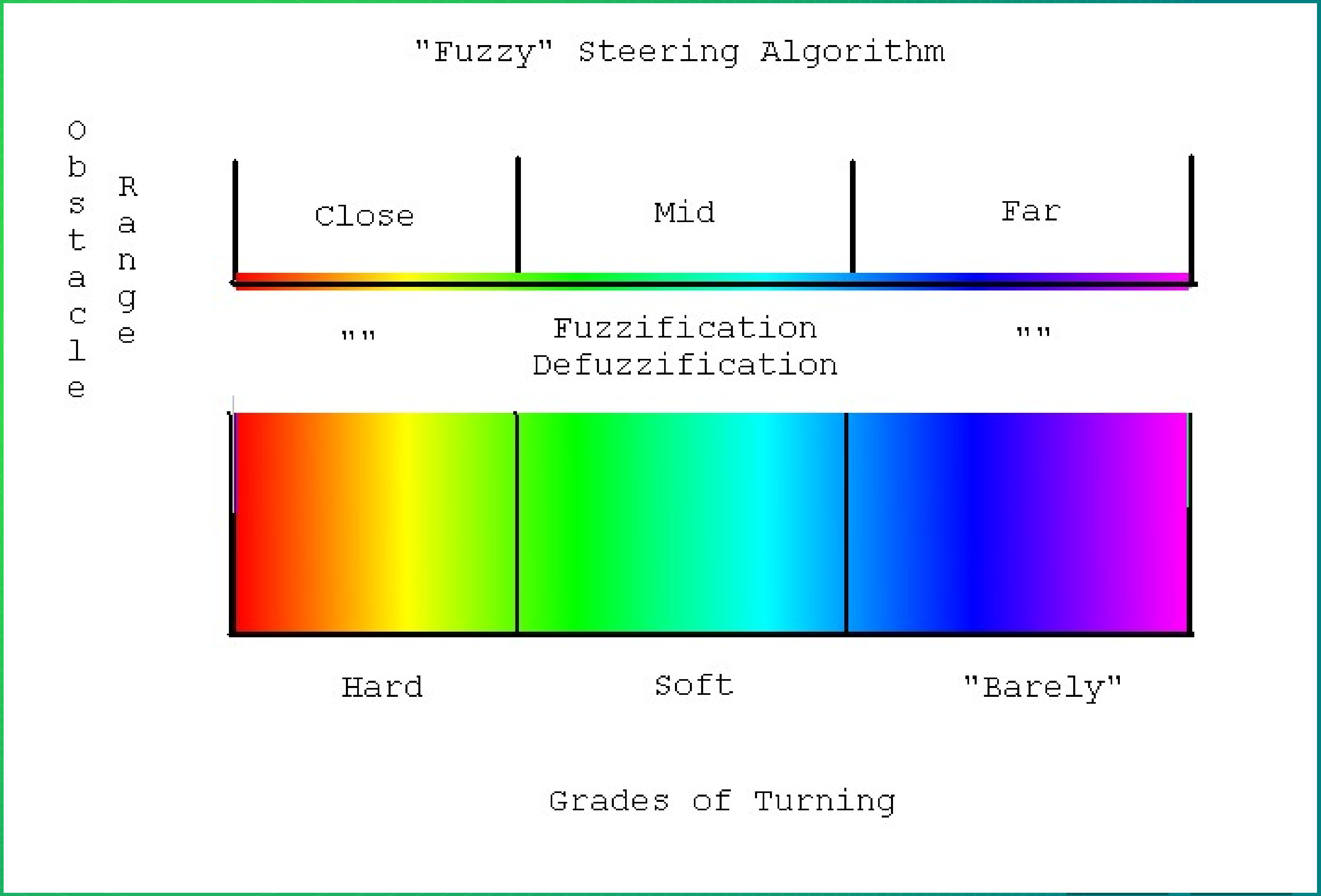
- **Preserve distinctness of sensor data**
- **Achieve smoother turning and a finer level of granularity of control over the vehicle's motion, filling in all the "shades of gray" between the discrete values of steering.**

- Crisp



- Fuzzy

Project Goals



The Math

Using Fuzzy Sets

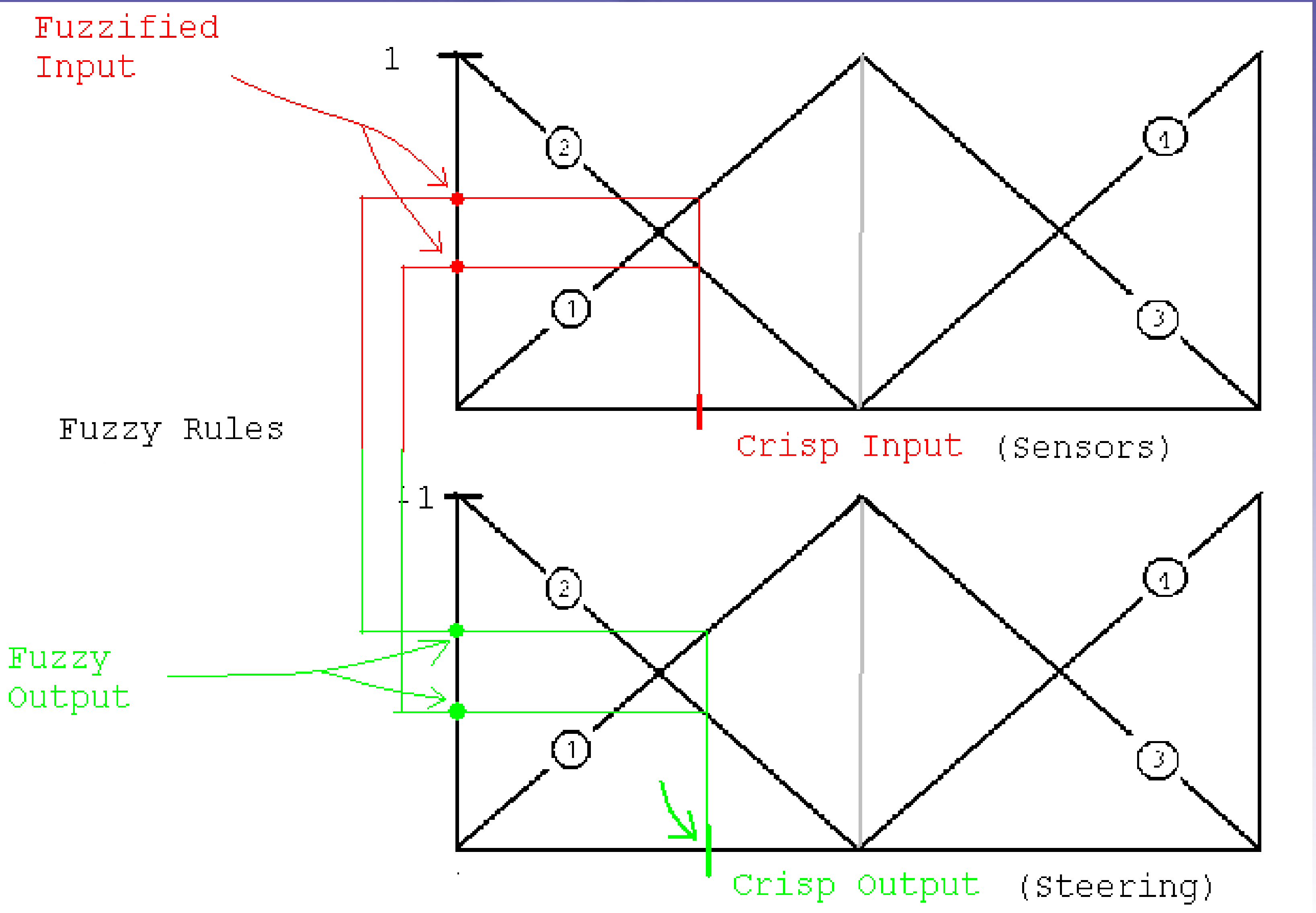
In order to achieve our project goals, we will:

(1) Fuzzify the input values received via the sensors, and

(2) Defuzzify these degrees of membership according to the fuzzy inference rules in the steering algorithm:

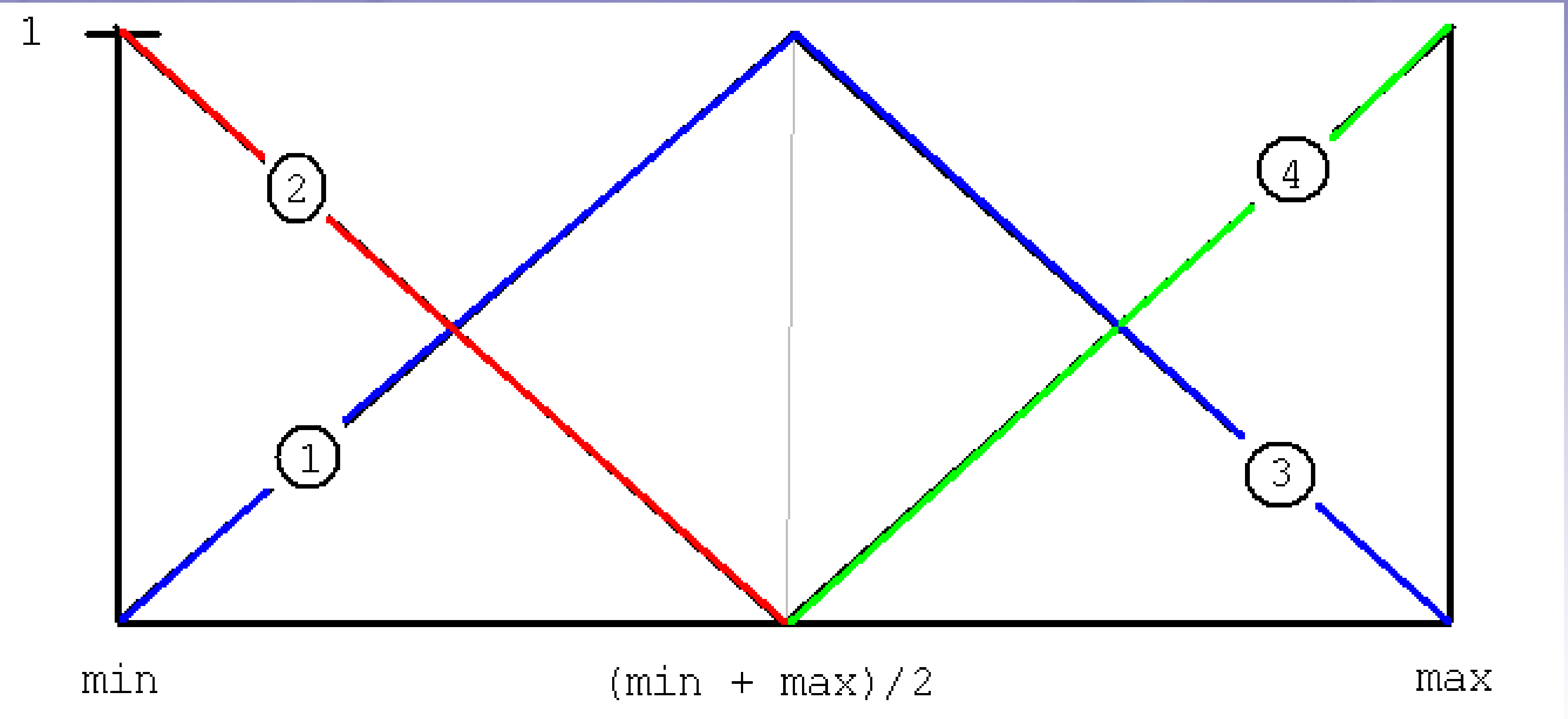
(The ***further away*** *an object is from the truck,*
the ***less*** *the truck will turn.*
The ***closer*** *an object is to the truck,*
the ***more*** *the truck will turn.)*

Overall View of the Translation Process



Generalized Membership Functions and Inference Rules Used

	Sensors	Rule	Steering
(2):	Far	1-to-1	Barely
(1),(3):	Mid-Range	1-to-1	Soft
(4):	Near	1-to-1	Hard



Analytic Form of Membership Functions Used

$$A(x) = \begin{cases} (1)\quad 2\dfrac{(x - min)}{(max - min)}, & \{x \mid \dfrac{(min + max)}{2} \le x \le max\} \\ (2)\quad \dfrac{(max + min - 2x)}{(max - min)}, & \\ (3)\quad 2\dfrac{(max - x)}{(max - min)}, & \{x \mid min \le x \le \dfrac{(min + max)}{2}\} \\ (4)\quad \dfrac{(2x - (max + min))}{(max - min)}, & \end{cases}$$

$$A : X \rightarrow [0,1]$$

Degrees of Membership Generated for Sensor Input Values

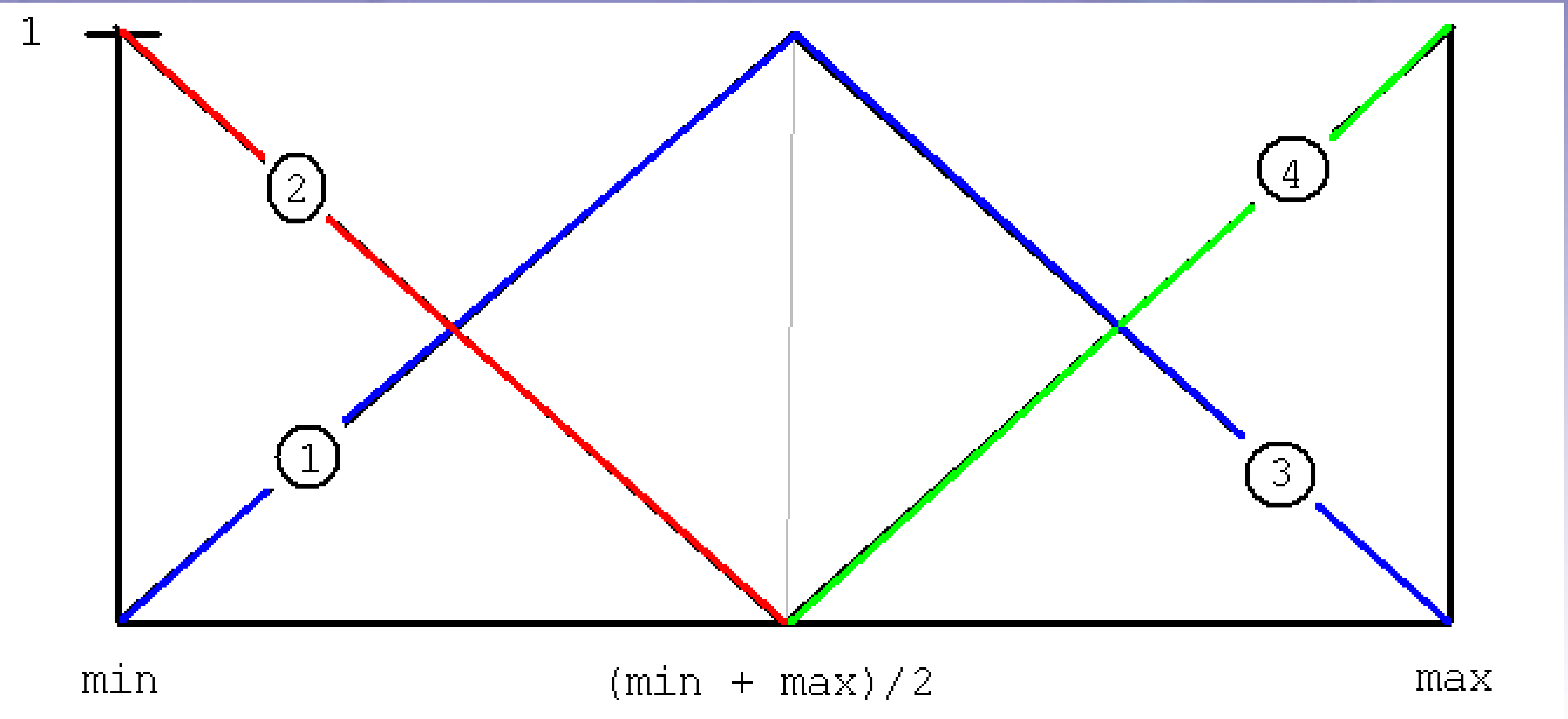
X	(1)DOM	(2)DOM
F	0.00	1.00
10	0.02	0.98
11	0.03	0.97
12	0.05	0.95
13	0.06	0.94
14	0.08	0.92
...		

- Notice how the sum of both columns in the left and right table add up to 1.0
- **This symmetry simplified calculations for fuzzification and defuzzification**
- **A previous graph illustrated such data redundancy**

X	(3)DOM	(4)DOM
7A	0.33	0.67
7B	0.31	0.69
7C	0.30	0.70
7D	0.28	0.72
7E	0.27	0.73
7F	0.25	0.75
...		

Generalized Membership Functions and Inference Rules Used

	Sensors	Rule	Steering
(2):	Far	1-to-1	Barely
(1), (3):	Mid-Range	1-to-1	Soft
(4):	Near	1-to-1	Hard



Final Algebraic Translation

Now that we have a degree of membership, A(x), for the sensor input value, let's find the steering output value.

(1) The domain of x will highlight two potential membership functions for y – pick either (symmetry).

(2) Set A(x) = A(y)

(3) Solve for y to obtain crisp steering output.

Note: Using x -> sensor input, y -> steering output

Engineering Considerations & Implementation Details

- **Accuracy** and **validity** of sensor data very important

SOLUTIONS

(1) ***Take the average of two readings***
- Adding two 8-bit values could overflow
- No built-in division operation
- **Fix:** ad hoc division and averaging functions created with microcontroller-supported op's (STATUS carry bit, rrcf)

(2) ***Bounds checking:***
- Input > Max ? . . . Input = Max
- Input < Min ? . . . Input = Min

Engineering Considerations & Implementation Details

OTHER CHALLENGES

(1) No floating point operations available

(2) MCU is slow (even in ASM...*up to* 10 MIPS)

Run-time calc.'s of membership fcts?

SOLUTION: LOOKUP TABLE

(1) Precompute all possible input/output values

(2) Load a table with these values into the chip at build-time

(3) Access the appropriate table value at run-time through the stabilized sensor reading

Lookup Table Generation & Other Membership Functions

The previous triangular membership functions were just a few of the candidates explored during this project:

Link to Install FuzzyGUI on your machine

steer left table   steer right table

Results

Corridor Comparisons

Crisp

Results

Corridor Comparisons

Fuzzy

Results

Stress Test 1

Crisp

Results

Stress Test 1

Fuzzy

Results

Stress Test 2

Crisp

Results

Stress Test 2

Fuzzy

Results

"Turn Hugging" Comparison

Crisp

Results

"Turn Hugging" Comparison

Fuzzy

Final Commentary

Even with the limited
computational resources provided
by an 8-bit microcontroller,
the concepts behind fuzzy sets
offer a simple yet effective way
to improve significantly
the quality of control
over robotic vehicle navigation.

Acknowledgments

Thank you for your time!

Special thanks to Tajina Casey for help with slide editing and video footage.