

Trabalho sobre sistemas fuzzy

Estacionamento de um caminhão em doca

Mauro Roisenberg

Professor do Departamento de Informática e Estatística da Universidade Federal de Santa Catarina

Fábio Nunes Garcia

Gustavo Figueira Olegário

Alunos da Disciplina INE5430 (2018/1) - Inteligência Artificial
Ciências da Computação do Departamento de Informática e Estatística da
Universidade Federal de Santa Catarina (UFSC)

Florianópolis, 8 de junho de 2018.

1 Introdução

Desenvolvemos um controlador em linguagem FCL a partir do JFuzzyLogic para estacionar de marcha ré um caminhão em uma vaga numa doca. Na controladora apresentamos os métodos de fuzzificação das variáveis x , y e $angle$ que são padrão da plataforma já desenvolvida, enquanto definimos o conjunto de regras utilizadas que instruem o caminhão.

O método de defuzzificação consiste em utilizar as atribuições das regras para obter um valor na faixa $[-1, 1]$ que então será convertido em um valor entre $[-30^\circ, 30^\circ]$ para representar o movimento do volante. Utilizamos a função COG (Center of Gravity) para a defuzzificação (que é o centróide) e o conjunto `{steeringTotallyTurnedToLeft, steeringTurnedToLeft, steeringTendingToTurnLeft, steeringCentered, steeringTendingToTurnRight, steeringTurnedToRight, steeringTotallyTurnedToRight}` representando o quanto o volante deve ser virado.

2 As entradas utilizadas por seu motorista

O cálculo do quanto o volante deve ser virado é feito a partir das variáveis x , y e $angle$ do próprio programa fornecido, já citadas anteriormente. A passagem dos parâmetros é feita através do `setVariable` na classe `RemoteDriver`, passando uma string com o nome o qual chamamos essas variáveis.

`RemoteDriver.java`:

```
fis.setVariable("positionX", x);  
fis.setVariable("positionY", y);  
fis.setVariable("angle", angle);
```

3 Como são os conjuntos fuzzy utilizados

Na variável x usamos um conjunto fuzzy com faixas da esquerda para direita definidas como: `left`, `moderateLeft`, `middle`, `moderateRight`, `right`.

O conjunto fuzzy de y separa, de cima para baixo:

`top`, `moderateTop`, `topBottom`, `moderateBottom`, `bottom`.

O conjunto fuzzy de `angle` utiliza uma separação de 90° em 90°, indo de 0 a 360, sobrepostos um sobre os outros, definido como: `turnedToRight`,

`turnedToRightTop`, `turnedToTop`, `turnedToLeftTop`, `turnedToLeft`, `turnedToLeftBottom`, `turnedToBottom`, `turnedToRightBottom`.

O código que representa esses conjuntos é apresentado abaixo:

```
FUZZIFY positionX
    TERM left := (0, 1) (0.35, 0);
    TERM moderateLeft := (0.3, 0) (0.4, 1) (0.5, 0);
    TERM middle := (0.4, 0) (0.5, 1) (0.6, 0);
    TERM moderateRight := (0.5, 0) (0.6, 1) (0.7, 0);
    TERM right := (0.65, 0) (1, 1);
END_FUZZIFY

FUZZIFY positionY
    TERM top := (0, 1) (0.3, 0);
    TERM moderateTop := (0.2, 0) (0.325, 1) (0.45, 0);
    TERM topBottom := (0.4, 0) (0.5, 1) (0.6, 0);
    TERM moderateBottom := (0.55, 0) (0.675, 1) (0.8, 0);
    TERM bottom := (0.7, 0) (1, 1);
END_FUZZIFY

FUZZIFY angle
    TERM turnedToRight := (0, 1) (45, 0) (315, 0) (360, 1);
    TERM turnedToRightTop := (0, 0) (45, 1) (90, 0);
    TERM turnedToTop := (45, 0) (90, 1) (135, 0);
    TERM turnedToLeftTop := (90, 0) (135, 1) (180, 0);
    TERM turnedToLeft := (135, 0) (180, 1) (225, 0);
    TERM turnedToLeftBottom := (180, 0) (225, 1) (270, 0);
    TERM turnedToBottom := (225, 0) (270, 1) (315, 0);
    TERM turnedToRightBottom := (270, 0) (315, 1) (360, 0);
END_FUZZIFY
```

4 Regras utilizadas

A formação do conjunto de regras se dá considerando apenas a posição em x e o $angle$ do caminhão pela facilidade, conforme orientação do professor.

Considerando apenas as possibilidades singulares de cada variável, chegamos a 7 possibilidades de $angle$ e 8 de x , com apenas 2 regras usando y , num total de 17 regras:

```
RULE 1 : IF (angle IS turnedToTop) AND (positionX IS middle) THEN steering IS steeringCentered;
RULE 2 : IF (angle IS turnedToRight) AND positionX IS moderateLeft THEN steering IS steeringTurnedToRight;
RULE 3 : IF (angle IS turnedToLeft) AND positionX IS moderateRight THEN steering IS steeringTurnedToLeft;
RULE 4 : IF (angle IS turnedToBottom) AND (positionX IS left) THEN steering IS steeringTotallyTurnedToLeft;
RULE 5 : IF (angle IS turnedToBottom) AND (positionX IS right) THEN steering IS steeringTotallyTurnedToRight;
RULE 6 : IF (angle IS turnedToLeftBottom) THEN steering IS steeringTotallyTurnedToLeft;
RULE 7 : IF (angle IS turnedToRightBottom) THEN steering IS steeringTotallyTurnedToRight;
RULE 8 : IF (positionX IS right) THEN steering IS steeringTotallyTurnedToLeft;
RULE 9 : IF (positionX IS left) THEN steering IS steeringTotallyTurnedToRight;
RULE 10 : IF (positionX IS middle) AND (angle IS turnedToLeftTop) THEN steering IS steeringTendingToTurnLeft;
RULE 11 : IF (positionX IS middle) AND (angle IS turnedToRightTop) THEN steering IS steeringTendingToTurnRight;
RULE 12 : IF (positionX IS moderateLeft) AND (angle IS turnedToTop) THEN steering IS steeringTendingToTurnRight;
RULE 13 : IF (positionX IS moderateRight) AND (angle IS turnedToTop) THEN steering IS steeringTendingToTurnLeft;
RULE 14 : IF (positionY IS topBottom) OR (positionY IS moderateBottom) THEN steering IS steeringCentered;
RULE 15 : IF positionX IS moderateRight OR positionX IS right AND positionY IS bottom AND angle IS turnedToRightBottom THEN steering IS steeringCentered;
RULE 16 : IF (angle IS turnedToLeft) AND positionX IS middle THEN steering IS steeringTurnedToLeft;
RULE 17 : IF (angle IS turnedToRight) AND positionX IS middle THEN steering IS steeringTurnedToRight;
```

5 Método de defuzzificação utilizado

Baseado nos conjuntos de fuzzificação e nas possibilidades de se virar o volante, criamos um conjunto de defuzzificação com 7 faixas como definido na introdução:

```
DEFUZZIFY steering
    TERM steeringTotallyTurnedToLeft := (-30, 1) (-15, 0);
    TERM steeringTurnedToLeft := (-20,0) (-15, 1) (-10, 0);
    TERM steeringTendingToTurnLeft := (-11, 0) (-7.5, 1) (-4, 0);
    TERM steeringCentered := (-7, 0) (0, 1) (7, 0);
    TERM steeringTendingToTurnRight := (4, 0) (7.5, 1) (11, 0);
    TERM steeringTurnedToRight := (10, 0) (15, 1) (20, 0);
    TERM steeringTotallyTurnedToRight := (15, 0) (30,1);
    METHOD : COG;
    DEFAULT := 0;
END_DEFUZZIFY
```

Como método de defuzzificação, foi usado COG (center of gravity), que veio por padrão no código e também é o método do centróide visto em sala.

6 Conclusão

O motorista consegue estacionar na maioria dos casos, embora apresente problemas enquanto está muito próximo da doca. Nos casos onde y é maior ou igual a 0.9, o motorista passa por $y=1$ e encerra a simulação, não conseguindo estacionar.

Os principais obstáculos encontrados giraram em torno da dificuldade de definir as regras, uma vez que dependeu muito de tentativa e erro até encontrarmos um padrão aceitável. O conjunto de regras foi reformulado várias vezes fazendo ajustes às distribuições na fuzzificação e defuzzificação, principalmente no `steering` do volante. Apesar disso, tínhamos um controle sobre os casos de borda de extrema direita e extrema esquerda, onde definimos regras dependendo do ângulo do caminhão.

7 Código fonte

O código fonte `.fcl` usado para a lógica fuzzy pode ser encontrado no `.zip` do entregável, junto com todo o programa principal.