

ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE
FAKULTA ELEKTROTECHNICKÁ
KATEDRA ŘÍDICÍ TECHNIKY



BAKALÁŘSKÁ PRÁCE

Řízení couvajícího vozíku s využitím fuzzy logiky

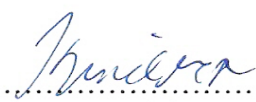
Praha, 2008

Vladimír Kučera

Prohlášení

Prohlašuji, že jsem svou bakalářskou práci vypracoval samostatně a použil jsem pouze podklady (literaturu, projekty, SW atd.) uvedené v příloženém seznamu.

V Praze, dne 25.6 2008


.....
podpis

Poděkování

Děkuji Ing. Petru Hušekovi, Ph.D. za vedení bakalářské práce a cenné rady a podmínky. Dále děkuji všem, kteří mě podporovali při vyhotovení této práce.

Abstrakt

Předkládaná bakalářská práce se dá rozdělit do dvou hlavních částí. V první části se práce zabývá především teoretickým základem lingvistických vyjádření experta ke konstrukci pravidel ve fuzzy systémech a stručným úvodem do fuzzy množin, který je nezbytný pro pochopení dalšího textu. Následují teoretické poznatky z konstrukce fuzzy regulátorů, kde se rozeberou postupy použité při řízení konkrétního systému couvajícího vozíku. Do pomyslné druhé části lze zařadit rozbor metody zabývající se konstrukcí pravidel z naměřených vstupů a výstupů při demonstraci řízení nelineárního systému couvajícího vozíku expertem. Poslední kapitola práce je věnována konečnému návrhu fuzzy regulátoru a realizaci řízení systému couvajícího vozíku v prostředí Matlab.

Abstract

The submitted Bachelor thesis consists of two major parts. The first part mainly focuses on the theoretical basis of the expert's linguistic expression to fuzzy system rule construction. A short introduction to fuzzy sets which is essential to further understanding of the text is also presented in the first part. Thereafter, theoretical findings on fuzzy regulator construction are discussed as well as procedures used when control the particular backer-upper truck. The second part inquires into the analysis of the method dealing with rule construction from measured inputs and outputs while demonstrating the driving of non-linear system of the backer-upper truck by the expert. The closing part of the thesis is devoted to the final proposal of the fuzzy regulator and the implementation of the backer-upper truck conduction system in the Matlab environment.

České vysoké učení technické v Praze
Fakulta elektrotechnická

Katedra řídicí techniky

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

Student: **Vladimír Kučera**

Studijní program: Elektrotechnika a informatika (bakalářský), strukturovaný
Obor: Kybernetika a měření

Název tématu: **Řízení couvajících vozíků s využitím fuzzy logiky**

Pokyny pro vypracování:

1. Na základě vstupních a výstupních dat navrhnete fuzzy regulátor pro systém couvajících vozíků.
2. Porovnejte chování navrženého regulátoru se zpětnovazebním regulátorem implementovaným v prostředí Matlab.

Seznam odborné literatury:

L.X Wang: A Course in Fuzzy Systems and Control, Prentice Hall, 1997

Vedoucí: Ing. Petr Hušek, Ph.D.

Platnost zadání: do konce zimního semestru 2008/2009

prof. Ing. Michael Šebek, DrSc.
vedoucí katedry



B. Šimák
doc. Ing. Boris Šimák, CSc.
děkan

V Praze dne 25. 2. 2008

Obsah

Úvod 1

1 Základy fuzzy přístupu 2

- 1.1 Lingvistická proměnná 2
- 1.2 Lingvistická hodnota 2
- 1.3 Lingvistické pravidlo 2
- 1.4 Fuzzy množiny 3
 - 1.4.1 Zavedení fuzzy množin 3
- 1.5 Operace na fuzzy množinách 4
 - 1.5.1 Fuzzy doplněk 4
 - 1.5.2 Fuzzy průnik 4
 - 1.5.3 Fuzzy sjednocení 5

2 Konstrukce fuzzy regulátoru 6

- 2.1 Inferenční mechanismus 6
 - 2.1.1 Mamdaniho implikace 7
 - 2.1.2 Larsenova implikace 9
- 2.2 Báze znalostí 9
 - 2.2.1 Báze dat 9
 - 2.2.2 Báze pravidel 9
- 2.3 Defuzzifikace 10
 - 2.3.1 Centroid 10
 - 2.3.2 Bisector 11

3 Metoda vstupně-výstupních párů 12

- 3.1 Postup nastavení regulátorů 13

4 Řízení couvajícího vozíku 17

- 4.1 Model vozíku 17
- 4.2 Určení intervalů proměnných 17
- 4.3 Sběr dat 18
- 4.4 Analýza dat 19
- 4.5 Řízení couvajícího vozíku 20

Závěr 25

Literatura 26

A Obsah příloženého CD 27

Úvod

Celá práce bude směřovat k nastavení vhodného regulátoru pro problém řízení systému couvajícího vozíku. Při řízení nelineárního systému couvajícího vozíku můžeme, jestliže známe matematický model, využít konstrukce regulátoru založené na nelineární řídicí teorii (Walsh, Tilbury, Sastry, Murray a Laumond). Dalším přístupem je konstrukce regulátoru emulující lidské rozhodování a těmito regulátory se budeme v práci zabývat. Jedná se o přístup využívající znalostí experta a především zpracování dat z měření vstupů a výstupů soustavy, jestliže expert provádí demonstraci činnosti řízení nelineárního systému couvajícího vozíku, tedy couvání do cílové oblasti. Pravidla pro regulátor emulující lidské rozhodování, konkrétně fuzzy regulátor, nastavíme z pravidel získaných metodou návrhu fuzzy systému ze vstupně-výstupních párů dat.

Než se budeme zabývat řízením couvajícího vozíku a metodou využívající strategii převedení vstupů a výstupů na pravidla, je nezbytné zavést si několik pojmů z teorie fuzzy množin a operací, které se na nich provádějí. Dále se obecně podíváme na konstrukci fuzzy regulátorů, kterou využijeme pro pochopení metody vstupně-výstupních párů dat. Nakonec se budeme zabývat řízením nelineárního systému couvajícího vozíku v prostředí Matlab.

1 Základy fuzzy přístupu

Za zakladatele počtu fuzzy množin je pokládán Lotfi A. Zadeh svou prací [3], kterou publikoval v roce 1965. Přestože myšlenkou více pravdivostních hodnot v logice se zabývala celá řada matematiků včetně rodáka z Brna Gödela a dalších matematiků mnohem dříve, je však velikou zásluhou Lotfi A. Zadeho, že se obor fuzzy množin rozšířil do velikého spektra aplikačních oborů hlavně v řízení. Algoritmy založené na fuzzy jsou dnes používané například v rychlých databázových systémech, které používá Google, autopilotech letadel, v úlohách zpracování obrazu, umělé inteligenci, řízení zejména nelineárních systémů v průmyslových procesech a mnoha dalších aplikací.

Podrobný popis fuzzy množin a počítání s nimi je velice obsáhlá záležitost, která mnohonásobně přesahuje možnosti této práce. Podrobně se problematice věnuje literatura [2]. Zde uvedeme jen základní termíny potřebné pro řešení úlohy couvajícího vozíku.

1.1 Lingvistická proměnná

Ve fuzzy řízení se používá takzvaná Lingvistická proměnná, pro popis vstupů a výstupů systému. Jestliže chceme sestavit pravidla pro bázi pravidel ze znalostí experta, který použije lingvistický popis, zavedeme pro vstupy a výstupy lingvistickou proměnnou \tilde{x}_i popisující vstup x_i a lingvistickou proměnnou \tilde{y}_i popisující výstup y_i , kde $i=1,2,\dots,N$. N je počet vstupů a výstupů. Například pro elektrické systémy můžeme jako lingvistické proměnné uvažovat \tilde{x}_i ="napětí zdroje" a pro výstup \tilde{y}_i ="otáčky motoru".

1.2 Lingvistická hodnota

Hodnoty vstupů x_i a výstupů y_i se pohybují na univerzu X_i a Y_i charakterizující, kterých hodnot můžou proměnné nabývat. Vyjádření experta jakou hodnotu má lingvistická proměnná \tilde{y}_i ="otáčky motoru" vyjádříme lingvistickou hodnotou \tilde{A}_i^j , kde $j=1,2,\dots,P$. P je počet lingvistických hodnot definovaných na Y_i lingvistické proměnné \tilde{y}_i . Například pro \tilde{y}_1 ="otáčky motoru" můžeme definovat lingvistické hodnoty \tilde{A}_1^1 ="pomalé", \tilde{A}_1^2 ="střední", \tilde{A}_1^3 ="rychlé".

1.3 Lingvistické pravidlo

Expert, který řídí nějakou soustavu, se při vyhodnocování stavu a řízení systému chová zhruba následovně: Sleduje stavové proměnné systému a poté upraví akční zásah tak, aby se přiblížil k požadovanému výsledku. K tomu souží pravidlo jestliže-pak, které budeme používat v celém textu v anglické verzi IF-THEN.

$$\text{IF } \tilde{x}_i \text{ is } \tilde{A}_i^j \text{ THEN } \tilde{y}_y \text{ is } \tilde{A}_y^j \quad (1.1)$$

Když bychom použily příklad elektrického obvodu s motorem, tak rovnice (1.1) může říkat něco následujícího: Jestliže napětí zdroje je malé, pak otáčky motoru jsou pomalé.

1.4 Fuzzy množiny

V této části se omezíme jen na základy, které jsou nutné k pochopení fuzzy množin a operací s nimi v řízení. Hlubší pohled je nad rámec této práce a naleznete jej například v literatuře [2].

1.4.1 Zavedení fuzzy množin

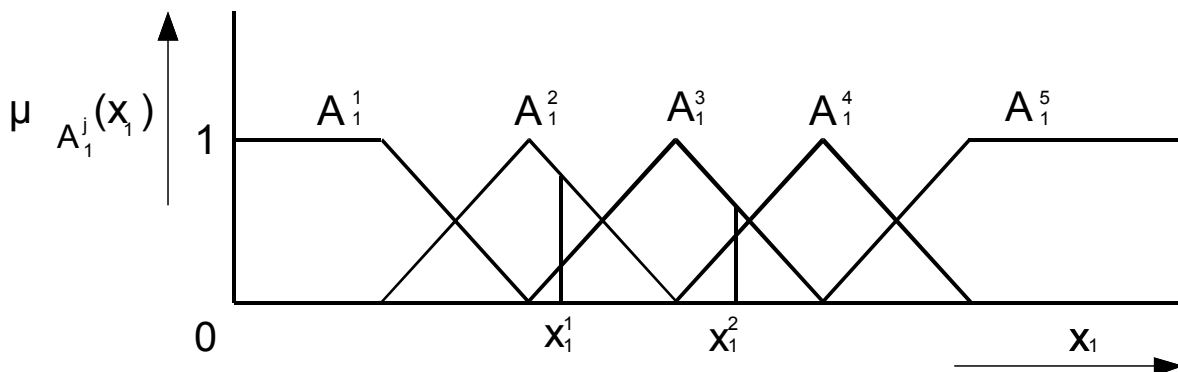
Máme množinu X_i , kterou budeme nazývat univerzum, v literatuře [2] se také nazývá univerzální množina. Fuzzy podmnožina A_i^j univerza X_i je taková podmnožina popsaná funkcí příslušnosti.

$$\mu_{A_i^j} : X_i \rightarrow \langle 0,1 \rangle \quad (1.2)$$

Rovnice (1.2) vyjadřuje do jaké míry prvek x_i , definovaný na univerzu X_i , patří do fuzzy podmnožiny A_i^j tím, že použije zobrazení do intervalu reálných čísel $\langle 0,1 \rangle$. Fuzzy množinu tedy budeme definovat podle rovnice (1.3).

$$A_i^j = \{(x_i, \mu_{A_i^j}(x_i)) : x_i \in X_i\} \quad (1.3)$$

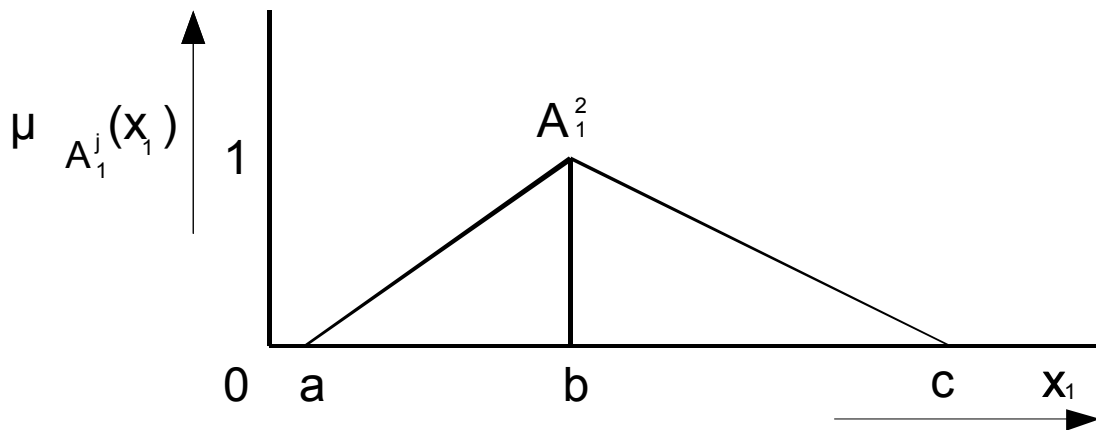
Obrázek 1.1 nám zobrazuje množiny $A_1^j = \{A_1^1, A_1^2, A_1^3, A_1^4, A_1^5\}$ proměnné x_1 na univerzu X_1 . Horní index u proměnné x_1^n pro $n=1,2,\dots,N$, kde N je počet vzorků proměnné x_1 jejichž funkce příslušnosti nabývají hodnot $\mu_{A_1^2}(x_1^1) = 0,7$ a $\mu_{A_1^3}(x_1^2) = 0,6$.



Obrázek 1.1: Funkce příslušnosti

Funkce příslušnosti těchto množin mohou být různého tvaru. Nejčastěji se používá trojúhelníková funkce příslušnosti, jako na obrázku 1.1, nebo Gaussova funkce příslušnosti, jejíž popis naleznete v literatuře[2]. V úloze řízení couvajícího vozíku je použita trojúhelníková funkce. Výpočet hodnoty funkce příslušnosti je pro metodu nastavení regulátorů ze vstupních a výstupních dat klíčovým výpočtem pro určení stupně ke každému pravidlu pro strategii rozhodující, které z pravidel se použije pro konstrukci regulátoru. Hodnotu funkce příslušnosti trojúhelníkové fuzzy množiny na obrázku 1.2 proměnné x_1 vypočítáme dle rovnice (1.4).

$$\mu_{A_1^2}(x_1) = \begin{cases} 0, & x_1 < a \text{ nebo } x_1 > c \\ \frac{x_1 - a}{b - a}, & a \leq x_1 \leq b \\ \frac{c - x_1}{c - b}, & b \leq x_1 \leq c \\ 1, & x_1 = b \end{cases} \quad (1.4)$$



Obrázek 1.2: Trojúhelníková fuzzy množina

1.5 Operace na fuzzy množinách

V této části si ukážeme některé základní operace s fuzzy množinami nutné pro pochopení další problematiky fuzzy regulace. Obsáhlejší výklad je možné nalézt v publikaci [2].

1.5.1 Fuzzy doplněk

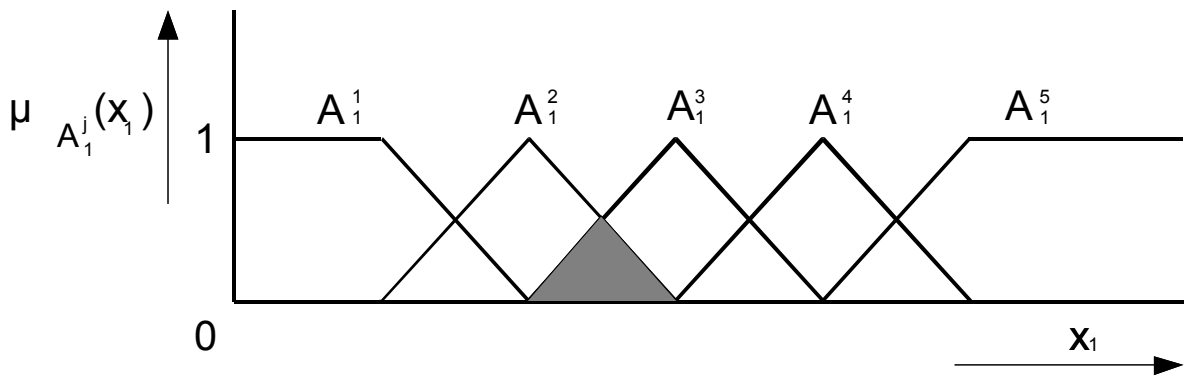
Fuzzy doplněk D množiny A_1^1 proměnné x_1 , která je popsána funkcí příslušnosti $\mu_{A_1^1}(x_1)$ je taková funkce příslušnosti popsána vztahem (1.5).

$$\mu_D(x) = 1 - \mu_{A_1^1}(x_1) \quad (1.5)$$

1.5.2 Fuzzy průnik

Průnik dvou fuzzy množin A_1^2 a A_1^3 popsáných jejich funkcemi příslušnosti $\mu_{A_1^2}(x_1)$, $\mu_{A_1^3}(x_1)$ se vyjádří dle vztahu (1.6). Fuzzy průnik je pak zobrazen na obrázku 1.2.

$$\mu_{A_1^2 \cap A_1^3} = \min \{ \mu_{A_1^2}(x_1), \mu_{A_1^3}(x_1) \} \quad (1.6)$$

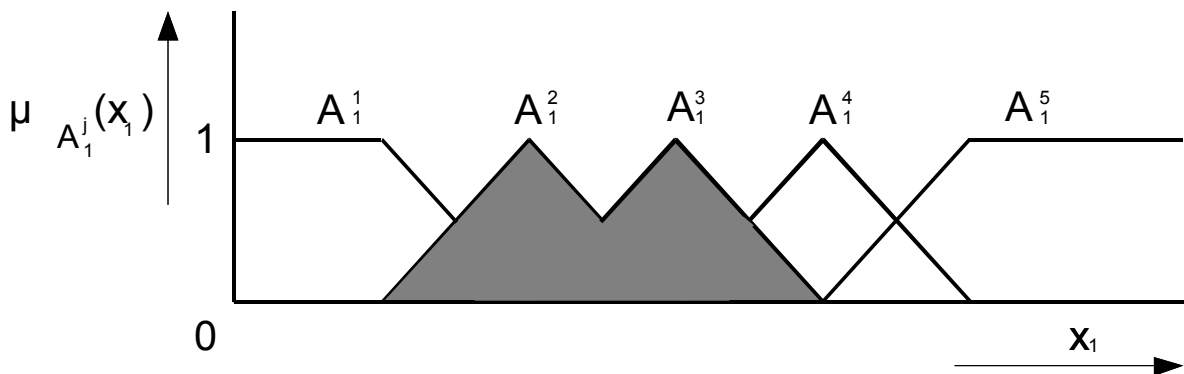


Obrázek 1.2: Průnik dvou fuzzy množin

1.5.3 Fuzzy sjednocení

Sjednocení dvou fuzzy množin A_1^2 a A_1^3 popsaných jejich funkcemi příslušnosti $\mu_{A_1^2}(x_1)$, $\mu_{A_1^3}(x_1)$ se vyjádří dle vztahu (1.7). Fuzzy sjednocení je pak zobrazen na obrázku 1.3.

$$\mu_{A_1^2 \cup A_1^3} = \max \{ \mu_{A_1^2}(x_1), \mu_{A_1^3}(x_1) \} \quad (1.7)$$

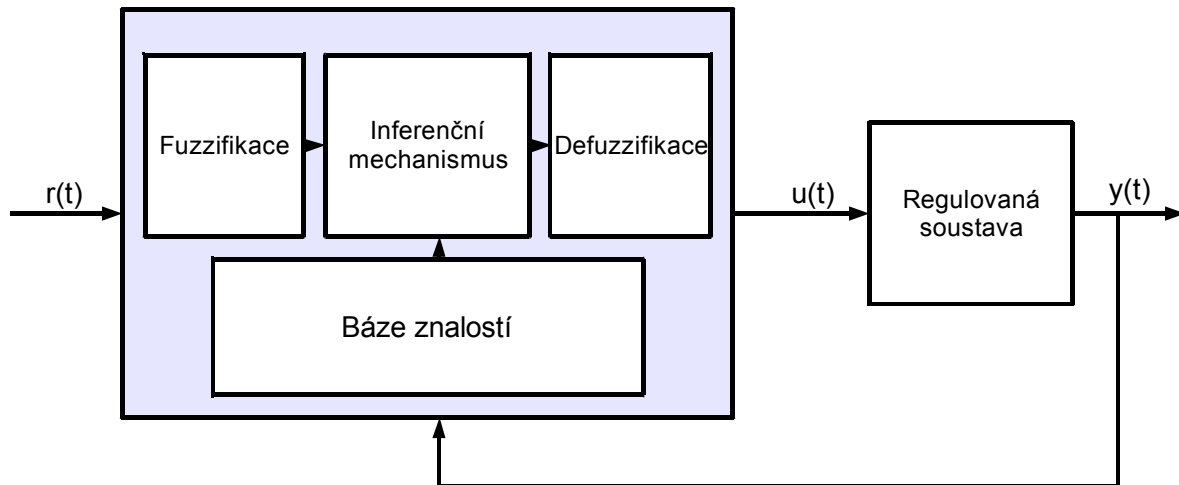


Obrázek 1.3: Sjednocení dvou fuzzy množin

2 Konstrukce fuzzy regulátoru

V této kapitole se podíváme na popis regulátoru, který je možný nastavit pomocí lingvistických vyjádření experta a metod spracování vstupu a výstupu pomocí běžného lingvistického pravidla. Nebudeme se zde zabývat typy fuzzy regulátorů P, PS, PD, PSD, které lze dohledat v literatuře [5].

Fuzzy regulátory se v nejběžnější podobě skládají z bloků fuzzifikace, báze znalostí, inferenčního mechanismu a defuzzifikace, jak je zobrazeno na obrázku 2.1.



Obrázek 2.1: Regulační obvod s fuzzy regulátorem

- a) Inferenční mechanismus
 - Provádí vyvození řídicích fuzzy zásahů regulátoru.
- b) Báze znalostí
 - Báze dat definuje funkce příslušnosti všech vstupních a výstupních fuzzy množin.
 - Báze pravidel obsahuje strategie řízení.
- c) Fuzzifikace
 - Vstupní ostré hodnoty měřené na řízené soustavě převádí do odpovídajících fuzzy množin vstupního fuzzy prostoru odpovídající proměnné.
- d) Defuzzifikace
 - Převádí výstupní fuzzy množinu na ostrou výstupní hodnotu regulátoru.

2.1 Inferenční mechanismus

Inferenční mechanismus pracuje nejčastěji s fuzzy výroky vstupního prostoru X_i a produkuje zase fuzzy výroky nějakého výstupního prostoru Y_i . Proto je nezbytné vstupní číselné hodnoty regulátoru převést do fuzzy množin, což zajišťuje blok fuzzifikace, pokud vstup regulátoru již není ve formě fuzzy. Inferenční mechanismus ve fuzzy řízení je nejčastěji vyjádřen jako implikace vstupních fuzzy výroků na výstupní dle pravidla IF-THEN, které se

také nazývá jako produkční pravidlo. Implikace může pracovat s více vstupními prostory, které odpovídají počtu stavových proměnných použitých k řízení systému, takovéto implikaci říkáme vícerozměrná. Jednotlivé fuzzy výroky vstupního prostoru jsou nejčastěji ve vztahu fuzzy průniku: AND nebo fuzzy sjednocení: OR.

IF < vstupní fuzzy výroky > THEN < výstupní fuzzy výrok >

Zavedeme dva pojmy, které se vážou k implikaci. Části množině vstupních fuzzy výroků, tedy v IF části, budeme říkat antecedent a části výstupní, tedy v THEN části, budeme říkat konsekvent.

Inferenčních pravidel je celá řada. Ukážeme si zde často používané přístupy, které se používají ve fuzzy logic toolboxu a jsou nejpoužívanější v aplikacích řízení systémů. Jedná se o Mamdaniho a Larsenovu implikaci.

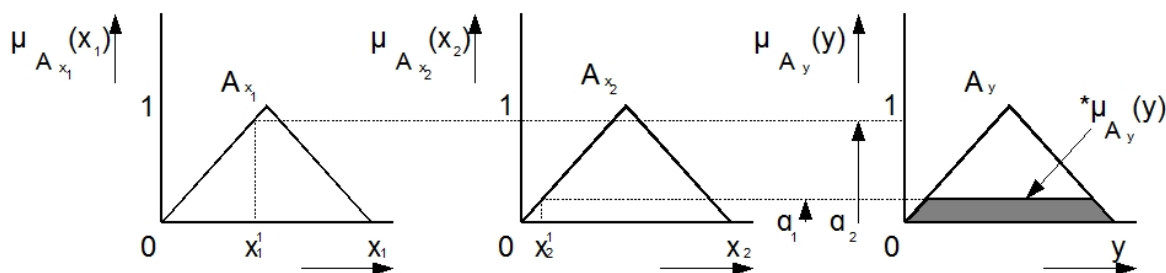
2.1.1 Mamdaniho implikace

Jedná se o jednu z nejrozšířenějších metod používané inferenčním mechanismem v řízení. Rozšířilo se díky své nenáročnosti na výpočetní složitost a dosažených výsledků v řízení.

Pro lepší přehlednost použijí značení množiny, kde je zřetelnější, ke které proměnné se množina vztahuje $A_{x_j}^j$ $j=1,2,\dots,N$, kde N je počet množin na univerzu. Pro nalezení konsekventu využijeme pro jedno pravidlo dvojrozměrné závislosti, kde výroky v antecedentu jsou ve vzájemném vztahu vyjádřeném vazbou AND, rovnice (2.1) a (2.2) zobrazené též na obrázku 2.2. Rovnice (2.1) říká, pomocí funkcí příslušnosti $\mu_{A_{x_1}}(x_1^1)$, $\mu_{A_{x_2}}(x_2^1)$ nalezneme hodnoty stupňů příslušnosti α_1 a α_2 k ostrým hodnotám x_1^1 , x_2^1 z nichž α odpovídá menší z těchto dvou stupňů příslušnosti. Rovnice (2.2) ořízne funkci příslušnosti $\mu_{A_y}(y)$ na hladině α . Výsledkem je nová funkce příslušnosti $^*\mu_{A_y}(y)$, která je na obrázku 2.2 zobrazena jako ohraničení šedé oblasti.

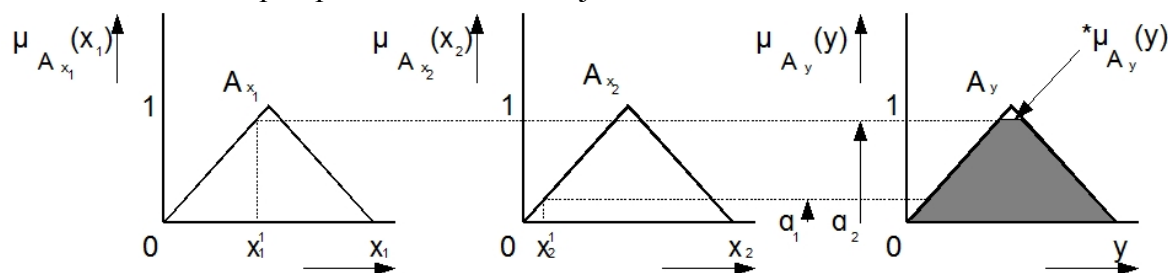
$$\alpha = \min \{ \mu_{A_{x_1}}(x_1^1), \mu_{A_{x_2}}(x_2^1) \} \quad (2.1)$$

$$^*\mu_{A_y}(y) = \min \{ \alpha, \mu_{A_y}(y) \} \quad (2.2)$$



Obrázek 2.2: Mamdaniho pravidlo dvojrozměrné závislosti AND

Pro nalezení konsekventu pro jedno pravidlo dvojrozměrné závislosti, kde výroky v antecedentu jsou ve vzájemném vztahu vyjádřeném vazbou OR, je situace obdobná jen hledáme větší ze stupňů příslušnosti. Situace je zobrazená na obrázku 2.3.



Obrázek 2.3: Nalezení konsekventu pro dvojrozměrnou závislost OR

Pro dvě pravidla dvojrozměrné závislosti, kde výroky v antecedentu jsou ve vzájemném vztahu vyjádřeném vazbou AND, hledáme fuzzy výrok konsekventu podobně jako pro jedno pravidlo. Nalezneme $^* \mu_{A_y^1}$ pomocí rovnic (2.3) a (2.4) a $^* \mu_{A_y^2}$ pomocí rovnic (2.5), (2.6) a výslednou funkci příslušnosti $^* \mu_{A_y}(y)$ hledáme jako fuzzy sjednocení dle rovnice (2.7).

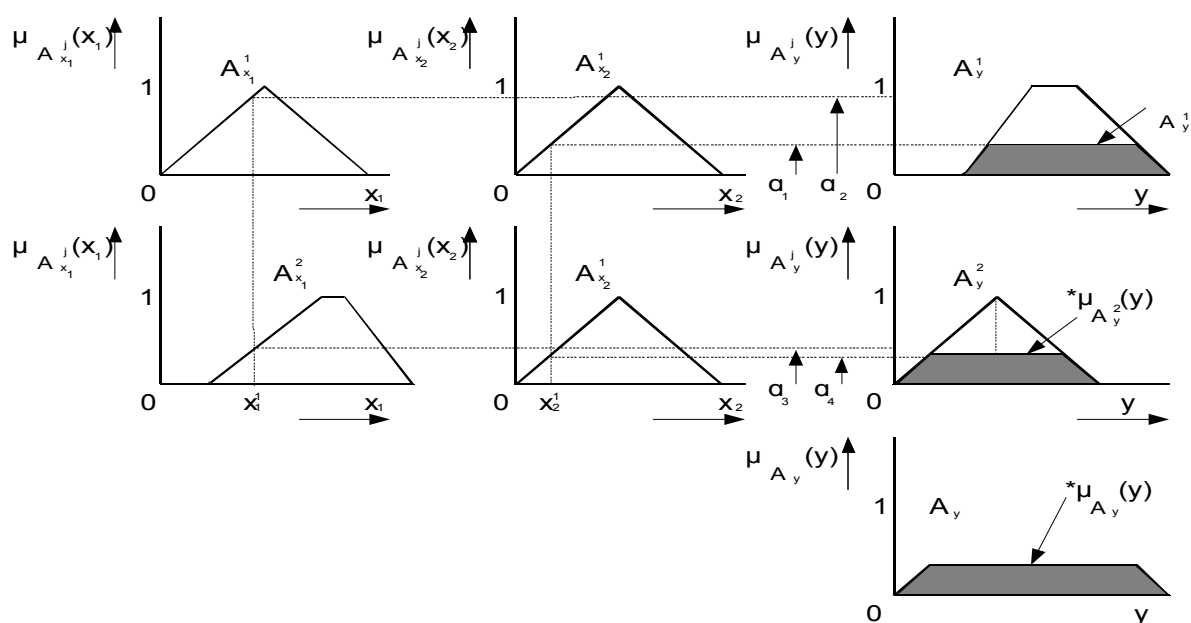
$$\alpha = \min \{ \mu_{A_{x_1}^1}(x_1^1), \mu_{A_{x_2}^1}(x_2^1) \} \quad (2.3)$$

$$^* \mu_{A_y^1}(y) = \min \{ \alpha, \mu_{A_y}(y) \} \quad (2.4)$$

$$\beta = \min \{ \mu_{A_{x_1}^2}(x_1^1), \mu_{A_{x_2}^2}(x_2^1) \} \quad (2.5)$$

$$^* \mu_{A_y^2}(y) = \min \{ \beta, \mu_{A_y}(y) \} \quad (2.6)$$

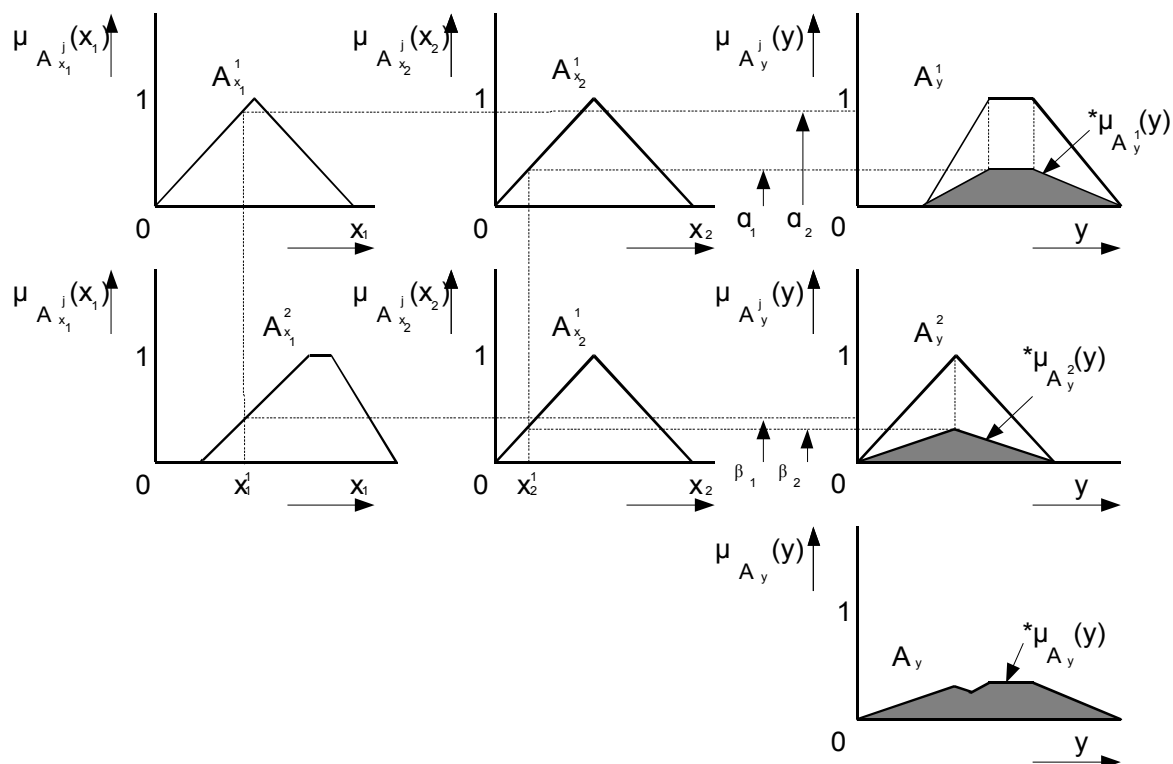
$$^* \mu_{A_y}(y) = \max \{ ^* \mu_{A_y^1}(y), ^* \mu_{A_y^2}(y) \} \quad (2.7)$$



Obrázek 2.4: Mamdaniho implikace ze dvou pravidel dvojrozměrných závislostí

2.1.2 Larsenova implikace

Larsenova implikace se od Mamdaniho liší tak, že nedochází k oříznutí výstupní množiny odpovídajícím stupněm příslušnosti, nýbrž vynásobením stupně příslušnosti a výstupní množiny. Výsledek pro dvě pravidla dvojrozměrné závislosti, kde výroky v antecedentu jsou ve vzájemném vztahu vyjádřeném vazbou AND, je ukázán na obrázku 2.3.



Obrázek 2.3: Larsenova implikace pro dvě pravidla s dvojrozměrnou závislostí.

2.2 Báze znalostí

Báze znalostí se skládá ze dvou částí a to Báze dat a báze pravidel. Podrobněji se konstrukcí báze znalostí věnuje následující kapitola přímo na konkrétní metodě nastavení ze vstupně-výstupních párů dat.

2.2.1 Báze dat

Jednak nese informaci o intervalech, na kterých se pohybují vstupní a výstupní proměnné a také nese definice všech funkcí příslušnosti fuzzy množiny na těchto vstupních a výstupních prostorech.

2.2.2 Báze pravidel

Obsahuje strategii řízení získanou nejčastěji popisem lingvistických pravidel dodaných expertem nebo učením báze pravidel z množiny příkladů získanou měření vstupů a výstupů a jejich převedením na pravidla za použití například metody zpracování vstupně-výstupních párů dat.

2.3 Defuzzifikace

Jak již bylo řečeno, defuzzifikace je strategie, převedení fuzzy výstupu z inferenčního mechanismu na ostrou hodnotu výstupu regulátoru. Uvedeme zde dvě metody defuzzifikace s anglickým pojmenováním centroid (těžiště) a bisector. Anglické pojmenování zvoleno z důvodu použití fuzzy logic toolboxu v prostředí Matlab při realizaci řízení couvajícího vozíku. Metody centroid a bisector jsou použity dle doporučení literatury [1], [4]. V literatuře můžeme najít mnoho strategií defuzzifikace jako jsou Metody nejvýznamnějšího maxima jako je MOM, LOM a ROM jejich popis lze najít v literatuře [5].

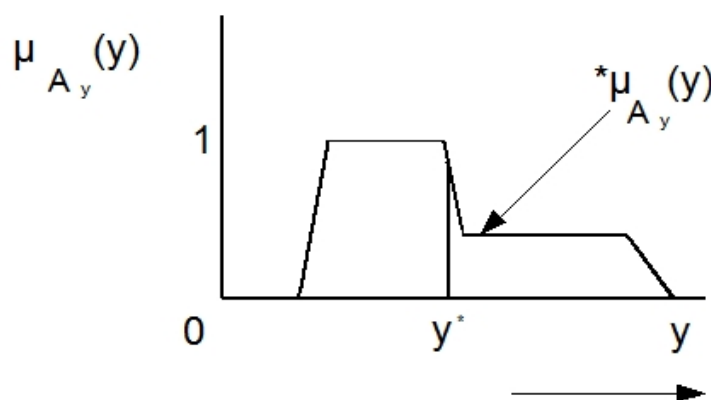
Každá ze strategií dává odlišné výsledky ostré výstupní hodnoty, proto strategii volíme převážně podle druhu aplikace. Formulování požadavků na defuzzifikaci se dá shrnout do následujících bodů.

- Spojitost
- Jednoznačnost obdrženého výsledku
- Výpočetní nároky dané metody
- Výsledná hodnota je požadována zhruba uprostřed množiny konsekventu s co největší hodnotou stupně příslušnosti

2.3.1 Centroid

Centroid oblasti je dnes jednou z nepoužívanějších strategií defuzzifikace v řízení, avšak je zde důvod použití například zmiňovaného bisectoru z důvodu výpočetní složitosti metody. Ostrou výstupní hodnotu y^* dostaneme použitím rovnice (2.8). Výstupní hodnota y^* je vlastně souřadnice těžiště plochy ohraničené funkcí příslušnosti ${}^*\mu_{A_y}(y)$ jak vidíme z obrázku 2.5.

$$y^* = \frac{\int {}^*\mu_{A_y}(y) \cdot y \, dy}{\int {}^*\mu_{A_y}(y) \, dy} \quad (2.8)$$

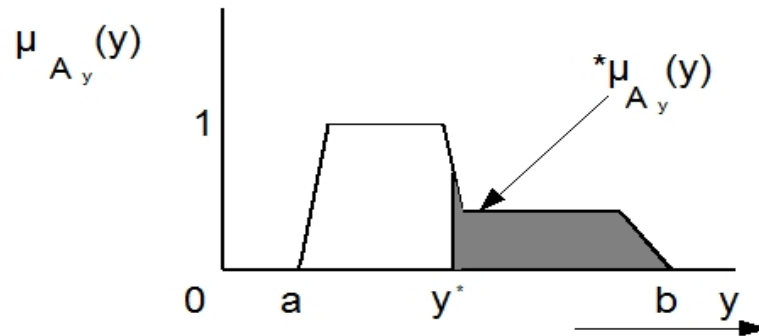


Obrázek 2.5: Metoda centroid

2.3.2 Bisector

Je strategie defuzzifikace vedoucí k podobným výsledkům jako metoda centroid, avšak s mnohem menší výpočetní složitostí. Jedná se o rozdělení do oblastí tak aby platila rovnice (2.9) tedy rovnost dvou ploch rozdělených hodnotou y^* jak je znázorněno na obrázku 2.6.

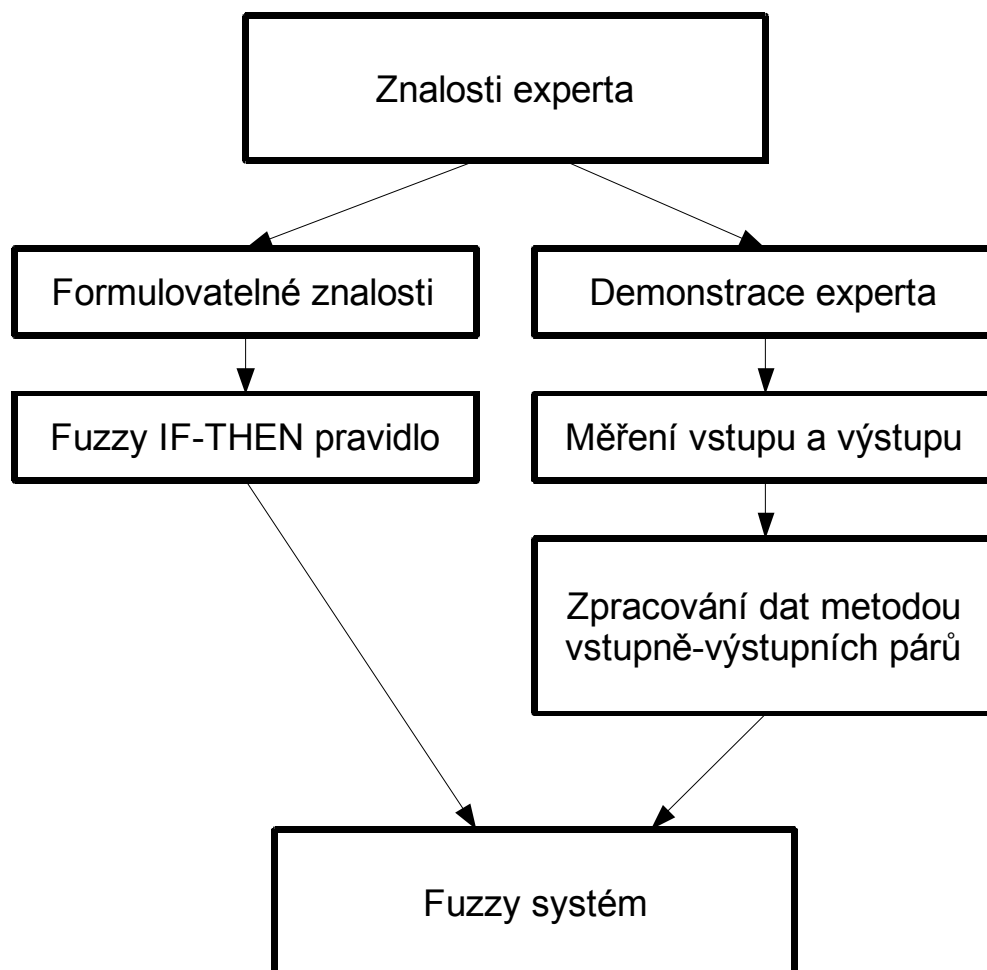
$$\int_a^{y^*} \mu_{A_y}(y) dy = \int_{y^*}^b \mu_{A_y}(y) dy \quad (2.9)$$



Obrázek 2.5: Metoda bisector

3 Metoda vstupně-výstupních párů

Jednou z mnoha výhod fuzzy přístupu je postup nastavení regulátoru s využitím formulovatelných znalostí expertů, kteří vědí jaký zásah do soustavy zavést, aby se soustava přiblížila požadovanému následujícímu stavu. Jedná se o přístup využívající lingvistických popisů. Expert se však může dostat do situace, kdy ví jaký zásah provést, ale přesný popis lze jen těžko formulovat do lingvistických pravidel. Z tohoto důvodu se používá metoda měření numerických dat vstupu a výstupu, kdy si experta můžeme představit za volantem couvajícího vozíku. Expert demonstruje řešení různých situací z různých počátečních podmínek a tím poskytuje množinu dat, které použijeme ke konstrukci fuzzy regulátoru. Jelikož oba přístupy návrhu, jak formulovatelnými, tak neformulovatelnými znalostmi, jsou neúplné, nabízí se jejich propojení při konstrukci fuzzy regulátoru, jak je znázorněno na obrázku 3.1. Neúplnost neformulovatelných znalostí a jejich zpracování v metodě vstupně-výstupních párů, spočívá ve špatné dosažitelnosti pokrytí veškerých situací, které mohou nastat při řízení systému. Neúplnost formulovatelných znalostí spočívá právě v případech, kdy situaci expert nemůže vhodně popsat, aby na základě jeho lingvistického popisu byl sestaven vyhovující fuzzy regulátor.



Obrázek 3.1: Postup konstrukce fuzzy systému.

3.1 Postup nastavení regulátorů

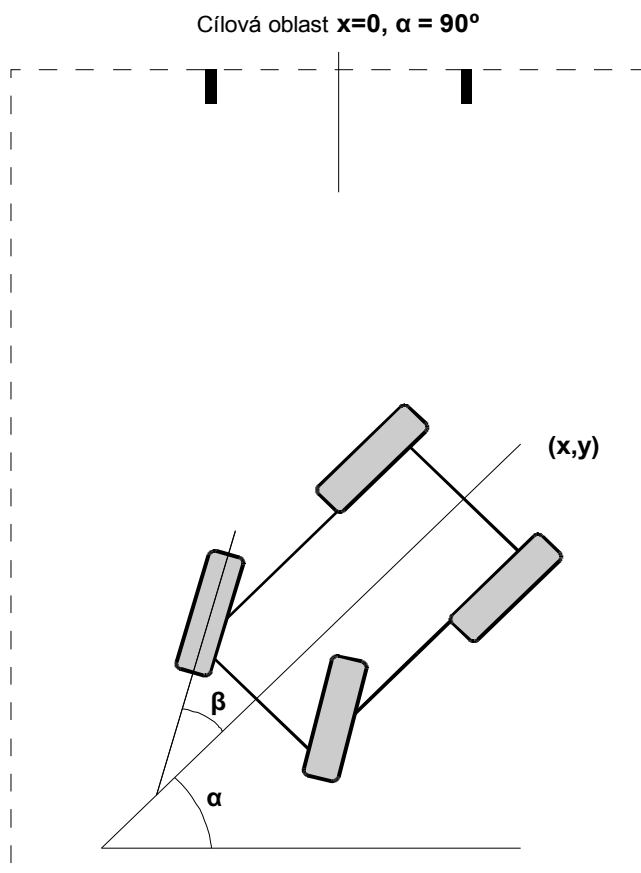
Naměřená data vstupů a výstupů lze reprezentovat vstupně-výstupními páry tvaru, znázorněném ve (3.1).

$$(x_1^1, x_2^1; y_1^1), (x_1^2, x_2^2; y_1^2), \dots, (x_1^n, x_2^n; y_1^n) \quad (3.1)$$

Pro přiblížení metody v konkrétních příkladech využijeme systému couvajícího vozíku. Couvající vozík popisuje vzdálenost od cílové oblasti reprezentované složkami vzdáleností na ose x a na ose y . Souřadnice na ose x udává, jestli se vozík pohybuje vpravo nebo vlevo, vzhledem k poloze osy cílové oblasti. Pro zjednodušení, vzdálenost na ose y nepovažujeme za stavovou proměnnou, vstupního prostoru fuzzy regulátoru, nutnou k řízení couvajícího vozíku a využívá se pouze jako dvojstavový vypínač k zastavení regulace při dosažení požadované vzdálenosti na ose y . Druhou vstupní proměnnou fuzzy regulátoru je natočení osy vozíku vůči přímce kolmé na osu cílové oblasti na obrázku 3.2 se jedná o úhel α . Výstupem fuzzy regulátoru je natočení předních kol, úhel β couvajícího vozíku na obrázku 3.2. Konkrétní vstupně-výstupní páry couvajícího vozíku jsou tedy pro obecné vyjádření (3.1) následující.

$$(x^1, \alpha^1; \beta^1), (x^2, \alpha^2; \beta^2), \dots, (x^n, \alpha^n; \beta^n) \quad (3.2)$$

Pro konstrukci fuzzy regulátoru ze Vstupně-výstupních párů použijeme následujících pět pravidel.



Obrázek 3.2: Schéma simulovaného vozíku

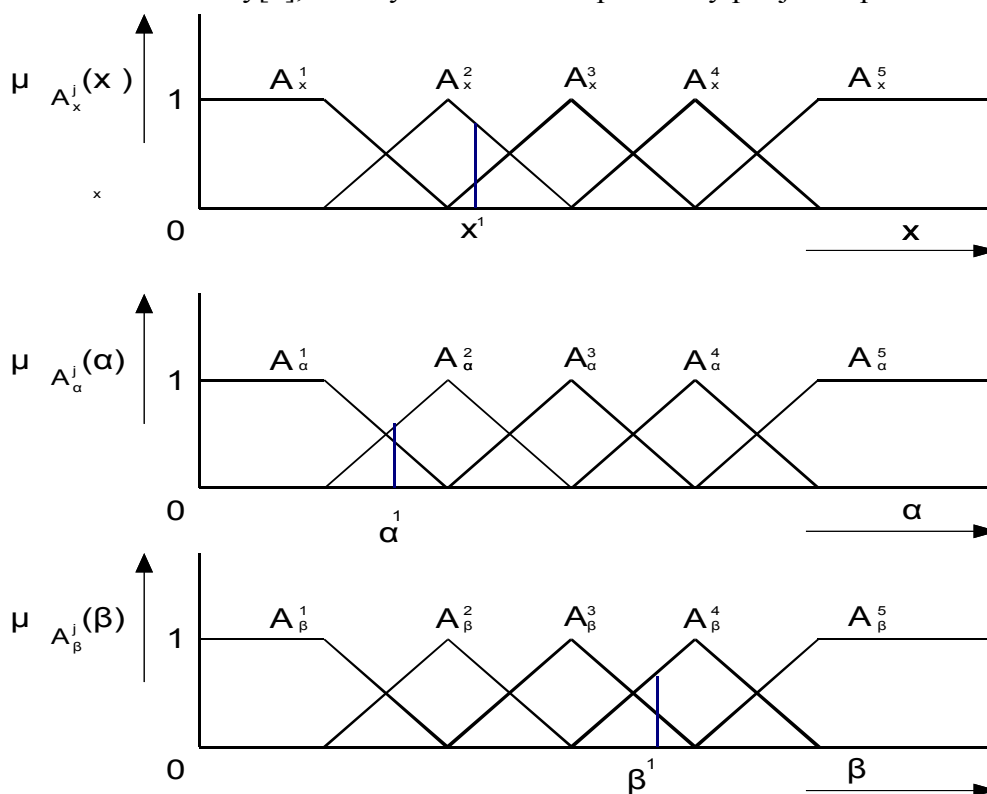
a) Definujeme fuzzy množiny tak abychom pokryli celý vstupní a výstupní prostor.

Pro každou proměnnou vstupně-výstupního páru $(x, \alpha; \beta)$ musíme definovat takové fuzzy množiny A_x^j, A_α^j a A_β^j pro $j=1,2,\dots,k_{x,\alpha,\beta}$, kde $k_{x,\alpha,\beta}$ je konečný počet množin potřebný pro pokrytí celého intervalu, v kterých se mohou proměnné pohybovat tak, aby pro každou hodnotu, ve které se můžou proměnné nacházet, platily rovnice (3.3).

$$\begin{aligned} \mu_{A_x^j}(x) &\neq 0 \\ \mu_{A_\alpha^j}(\alpha) &\neq 0 \\ \mu_{A_\beta^j}(\beta) &\neq 0 \end{aligned} \tag{3.3}$$

b) Pro každý vstupně-výstupní pár generujeme jedno pravidlo.

Máme pokrytý vstupní a výstupní prostor hodnot proměnných, teď potřebujeme sestavit pravidla ze vstupně-výstupních párů. Pro každý vstupně-výstupní pár $(x^n, \alpha^n; \beta^n)$ si určíme hodnotu funkce příslušnosti. Například první pár $(x^1, \alpha^1; \beta^1)$ má na obrázku 3.2 hodnotu x^1 v množině A_x^2 0,8 a v množině A_x^3 má hodnotu 0,2. Stejně se určí hodnoty pro α^1 , jejichž hodnota funkce příslušnosti je 0,45 v množině A_α^1 a 0,55 v množině A_α^2 , pro β^1 je hodnota funkce příslušnosti 0,3 v množině A_β^3 a 0,6 v A_β^4 . Pro tvorbu pravidla vybereme oblast, která má větší hodnotu funkce příslušnosti. Pravidla se pak sestaví dle schéma (3.4), převzatého z literatury[4], tak abychom obdrželi pro každý pár jedno pravidlo.



Obrázek 3.2: Určení hodnoty funkce příslušnosti proměnné

$$(x^1, \alpha^1, \beta^1) \Rightarrow (x^1(0,8 \vee A_x^2, \max), \alpha^1(0,55 \vee A_\alpha^2, \max); \beta^1(0,6 \vee A_\beta^4, \max)) \Rightarrow \text{Pravidlo 1} \quad (3.4)$$

Pravidlo pro schéma (3.4) tedy bude IF x^1 is A_x^2 AND α^1 is A_α^2 THEN β^1 is A_β^4 .

Pro sestavení báze pravidel je nutné mít více naměřených průběhů tedy jejich vstupně-výstupních párů, tak aby báze pravidel byla co neúplnější a reagovala na pohyb couvajícího vozíku ve všech případech, které mohou nastat. To vyžaduje určit velké množství funkcí příslušnosti a pro sestavení pravidla je nutné určit, do jaké fuzzy množiny spadají hodnoty proměnných. Jedná se o velké množství dat, pro jejichž analýzu používám vlastní jednoduchý program v jazyce Java. Do programu se zadají fuzzy množiny definované pro každou ze vstupních a výstupních proměnných. Program podle rovnice (1.4) pro vstupní hodnotu určí funkci příslušnosti a fuzzy množiny, do kterých hodnota zasahuje a zapamatuje si tuto množinu, kde má hodnota funkce příslušnosti větší hodnotu a tedy tato množina bude sloužit pro tvorbu pravidla. Tento postup se provede pro vzdálenost na ose x , pro natočení vozíku vzhledem k počátku a pro výstupní hodnotu natočení kol. Program dále sestaví pravidlo a spočítá hodnotu stupně pravidla používanou v následujícím kroku nastavení regulátoru.

c) Vybereme vhodné pravidlo spočítáním jeho stupně.

Ve velkém množství pravidel se obvykle stane, že je mnoho stejných pravidel a nastává situace, kdy je třeba určit, které z pravidel je vhodné pro konstrukci báze pravidel. U stejných pravidel není žádný problém, použije se jako by bylo jedno, jak je znázorněno na schématu (3.5).

$$\left. \begin{array}{l} \text{IF } A_x^2 \text{ AND } A_\alpha^2 \text{ THEN } A_\beta^4 \\ \text{IF } A_x^2 \text{ AND } A_\alpha^2 \text{ THEN } A_\beta^4 \end{array} \right\} \text{IF } A_x^2 \text{ AND } A_\alpha^2 \text{ THEN } A_\beta^4 \quad (3.5)$$

Problém nastává, jestliže je stejná IF část pravidla a jiná THEN část pravidla v takovém případě nastává problém, které z pravidel je vhodnější. K tomuto účelu se používá stupně pravidla, které rozhodne jaké pravidlo použít. Pro případ hodnot ve schématu (3.4) by stupeň pravidla vypadal dle rovnice (3.6).

$$D(\text{Pravidlo}) = \mu_{A_x^2}(x^1) \times \mu_{A_\alpha^2}(\alpha^1) \times \mu_{A_\beta^4}(\beta^1) \quad (3.6)$$

$$D(\text{Pravidlo}) = 0,8 \times 0,55 \times 0,6 = 0,264$$

Pro konstrukci báze pravidel se následně použije to pravidlo, které má větší hodnotu stupně pravidla D . Tato strategie pro výběr pravidla, se nápadně podobá praxi. Představme si situaci, kdy řidič couvá s naším vozíkem a jeho vstupní informace je vzdálenost od garáže a natočení směrem ke garáži a výstup je natočení kol volantem vozíku. Řidič zhodnotí situaci a zvolí ideální natočení kol, tak aby se přiblížil poloze, kterou požaduje. V případě, kdy o síle pravidla nerozhoduje expert, se používá výpočtu stupně pravidla, aby pro konstrukci báze pravidel bylo vybráno to silnější.

d) Vytvoříme bázi pravidel fuzzy regulátoru.

Pro lepší čitelnost pravidel je výhodné si fuzzy množiny pojmenovat tak aby jasně a čitelně vyjadřovaly stav couvajícího vozíku. Pro umístění ve směru osy x bylo v realizaci zavedeno pojmenování L, P a C kdy L znamená umístění vozíku vlevo na ose x , P znamená umístění vpravo na ose x a C je střed. Pro natočení vozíku bylo zvoleno taktéž označení L, P a C kdy L je natočení vlevo od úhlu k přímce kolmé na osu cílové oblasti, P je natočení vpravo na přímku kolmou k ose cílové oblasti a C znamená, že je vozík rovnoběžně s přímkou kolmou na cílovou oblast. Pro kola bylo v realizaci zavedeno L, P a C , kdy L je natočení kola vlevo P je natočení kola vpravo a C jsou kola rovnoběžně s osou vozíku.

Báze pravidel se sestaví z pravidel, které nekolidují s jinými, které jsme získali v bodě b). Dále pak z pravidel obdržných v bodu c), tedy z těch, co kolidují s jiným pravidlem, vybereme to, které má největší hodnotu stupně pravidla. Naposledy můžeme dodat lingvistická pravidla dodaná expertem tam, kde vstupně-výstupní páry nepokryly prostor báze pravidel. Na obrázku 3.3 vidíme příklad pravidel sestavených do tabulky, která znázorňuje příklad pro pravidla IF x is $P1$ AND α is $L1$ THEN β is $L1$ a IF x is C AND α is C THEN β is C . Tedy slovně řečeno: jestliže je vozík natočený vpravo k přímce kolmé na osu cílové oblasti a nachází se nalevo od cílové oblasti, nastav kola nalevo vzhledem k ose vozíku. Druhé pravidlo: jestliže je vozík rovnoběžně s přímkou kolmou k ose cílové oblasti a jeho poloha na ose x je shodná s polohou cílové oblasti, pak kola jsou nastavená rovnoběžně s osou couvajícího vozíku.

L3					
L2					
L1					
C			C		
P1		L1			
P2					
P3					
	L2	L1	C	P1	P2
	x				

Obrázek 3.3: Sestavení báze pravidel

e) Konečná konstrukce fuzzy regulátoru.

Pro fuzzy regulátor určíme vhodné metody fuzzifikace, defuzzifikace a inferenčního mechanismu z kapitoly 2. Například v řízení couvajícího vozíku je použito Mamdaniho implikace v inferenčním mechanismu a bisector defuzzifikace což vychází z literatury [1] a [4].

4 Řízení couvajícího vozíku

Poznatky shrnuté v předchozích kapitolách především pak v kapitole 3, kterou budeme více konkretizovat na daný problém, nyní uplatníme na řízení couvajícího vozíku.

Simulace řízení couvajícího vozíku do cílové oblasti je realizována v prostředí programu Matlab a jeho částí převážně Simulinku a fuzzy logic toolboxu. Při zpracování naměřených vstupně výstupních párů, jsem pro jejich analýzu používal program, napsaný v jazyce Java z, důvodu jednodušší implementace potřebného algoritmu na zpracování dat.

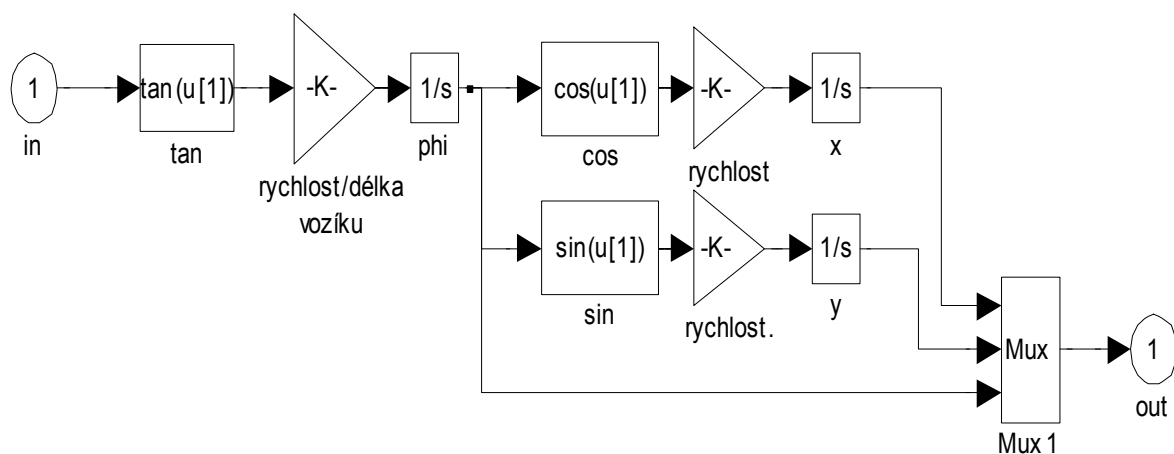
4.1 Model vozíku

Matematický popis nelineárního systému couvajícího vozíku je znázorněn trojicí rovnic (4.1), (4.2), (4,3). Kde x, y jsou souřadnice couvajícího vozíku. Konstanta v je rychlost jakou se pohybuje vozík. Konstanta l vyjadřuje vzdálenost mezi předními a zadními koly. Zopakujme, že úhel α [°] znázorňuje natočení osy vozíku vůči přímce kolmé na cílovou oblast. Úhel β [°] znázorňuje natočení kol vůči ose couvajícího vozíku. Model na obrázku (4.1) používaný v Simulinku je převzatý z příkladu Simulinku, který lze spustit příkazem sltbu ze zdroje[8], pro fuzzy logic toolbox.

$$\dot{x} = -v \cos(\alpha) \quad (4.1)$$

$$\dot{y} = -v \sin(\alpha) \quad (4.2)$$

$$\dot{\alpha} = -\frac{v}{l} \tan(\beta) \quad (4.3)$$

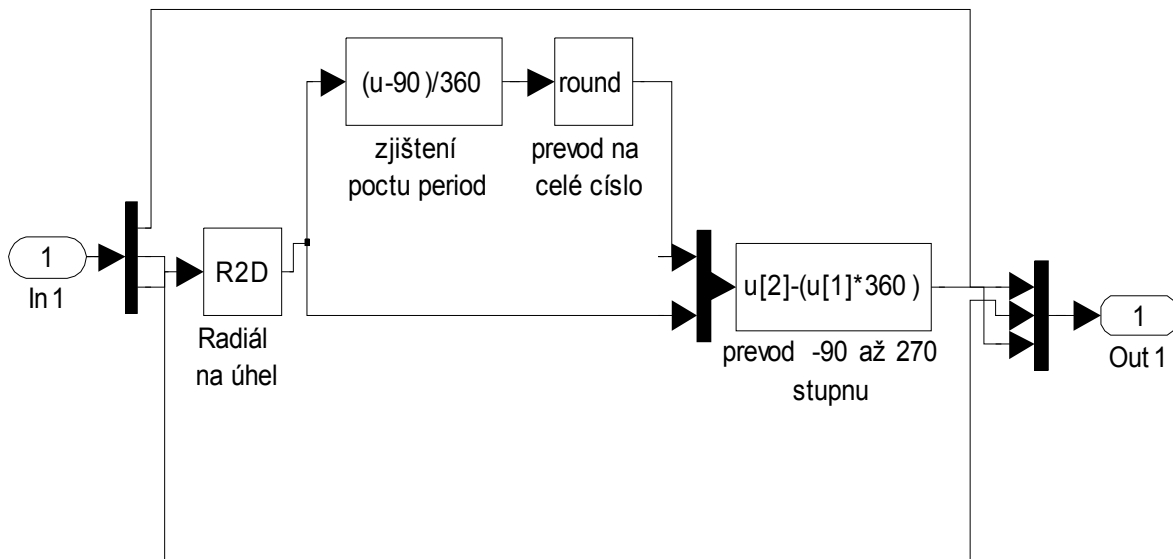


Obrázek 4.1: Model vozíku

4.2 Určení intervalů proměnných

Jako základ realizace řízení couvajícího vozíku je použit příklad Simulinku sltbu, který používá odlišný regulátor. Na tomto příkladě byly zjišťovány některé potřebné hodnoty. Příklad sltbu pracuje s úhly v radiánech a při každém přetočení vozíku se úhel zvětšil o periodu a je nezbytné obnovení počátečních stavů. Pro větší přehlednost a odstranění sečítání

úhlu je realizován blok, na obrázku 4.2, pro převod na stupně v rozsahu -90° až 270° . Blok pracuje tak, že spočítá počet period a následně je odečte, tím se docílí požadovaného intervalu.



Obrázek 4.2: Převod na úhel v rozsahu -90° , 270°

Jak víme z kapitoly 3, bodu nastavení regulátoru b), musíme vstupní a výstupní prostor báze dat regulátoru pokrýt v celém jeho rozsahu fuzzy množinami. Pro tento úkol je nutné určit některé rozsahy a hodnoty.

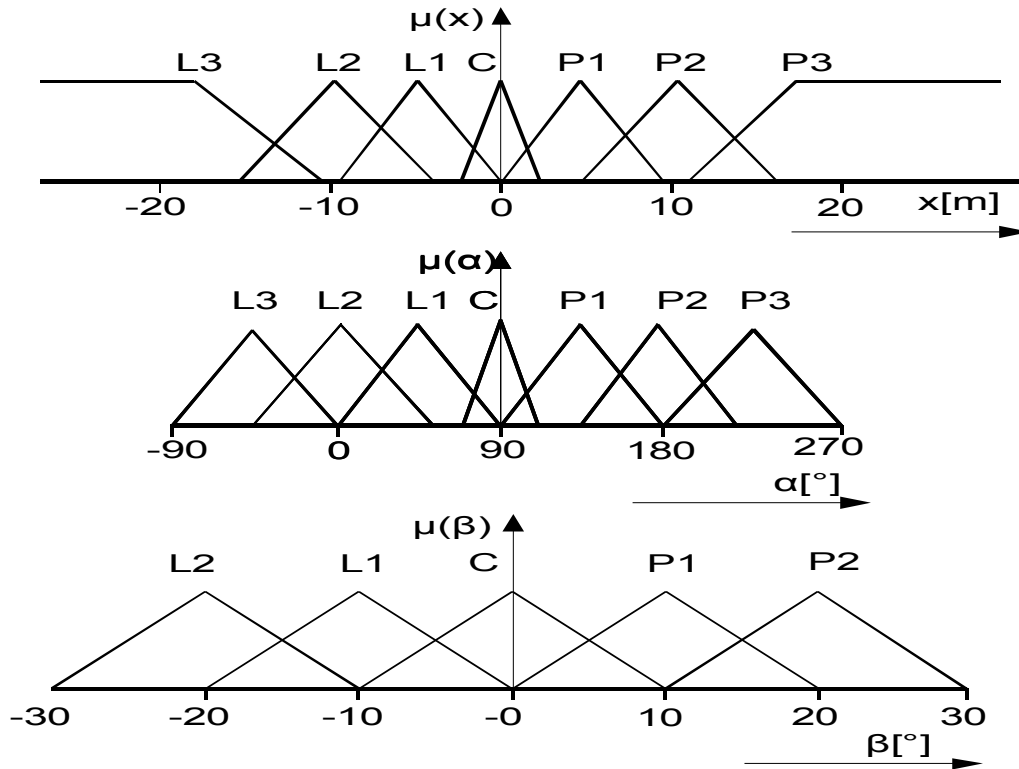
Měřením hodnot vstupů a výstupů soustavy se odhalily hodnoty cílové oblasti, které chceme dosáhnout, jedná se o nulovou hodnotu souřadnice x a y a polohy osy vozíku, tak aby byla kolmá na osu cílové oblasti. Hodnota úhlu α pro tuto oblast je 90° . Vozík se může nyní otáčet v obou směrech o 360° posunutých o -90° z důvodu, aby žádaná hodnota 90° byla uprostřed vstupního prostoru. Hodnoty souřadnice x jsou omezeny hodnotou -50 až 50 . Souřadnice y se v řízení couvajících vozíku této metody nepoužívá a slouží jen pro vypnutí celé simulace. Hodnota úhlu β , tedy natočení kol, je omezena v rozsahu -30° a 30° .

Pokrytí vstupních a výstupních prostorů fuzzy regulátoru funkcemi příslušnosti se pro problém couvajících vozíku vychází z literatury [1] a [4]. Tvary funkcí příslušnosti i jejich počet se upravovaly na základě zkušeností získaných každým dalším pokusem během testování. Pro dosažení větší přesnosti v určitých stavech regulované soustavy byla upravována základna trojúhelníkových funkcí příslušnosti. Konečný výsledek je vidět na obrázku 4.2. Zde vidíme také počty množin pro jednotlivé vstupní a výstupní prostory.

4.3 Sběr dat

Sběr dat pro tvorbu báze pravidel regulátoru ze vstupně-výstupních párů je realizován pomocí bloku převodu na úhel a ukládáním hodnot na vstupu a výstupu do externího souboru z příkladu řízení couvajících vozíku Simulinku sltbu. Tento příklad byl použit pro získání vstupně-výstupních párů dat, jelikož ruční řízení couvajících vozíku je velice obtížná záležitost a bylo by potřeba volantu a dalších úprav, aby byl člověk schopný s modelem vozíku zacouvat do cílové oblasti.

Pro úspěšné sestavení báze pravidel je třeba naměřit hodnoty z různých počátečních podmínek, které je důležité volit tak, aby bylo pokryto co největší množství stavů, kterých může vozík dosáhnout. Pro nastavení regulátoru je použito 20 měření z vhodně zvolených poloh.



Obrázek 4.3: Báze dat regulátoru

4.4 Analýza dat

Data se analyzují v externím programu napsaném v jazyce Java. V jednoduchém programu se nadefinují funkce příslušnosti vstupního a výstupního prostoru stejně jako v prostředí Matlab. Program poté pro ostré vstupní hodnoty vstupu a výstupu určí potřebná pravidla a jejich stupně, jak je řečeno v bodu b) a c) nastavení regulátoru v kapitole 3. Dále je na expertovi aby sestavil z pravidel bázi pravidel a upravil některé pravidla, aby dosáhl požadovaného chování vozíku. Báze sestavená dle návodu v kapitole 3 v bodě d) je znázorněna na obrázku 4.4. Při sestavování báze musíme zajistit její úplnost, tedy aby každá kombinace vstupní a výstupní množiny měla výstup.

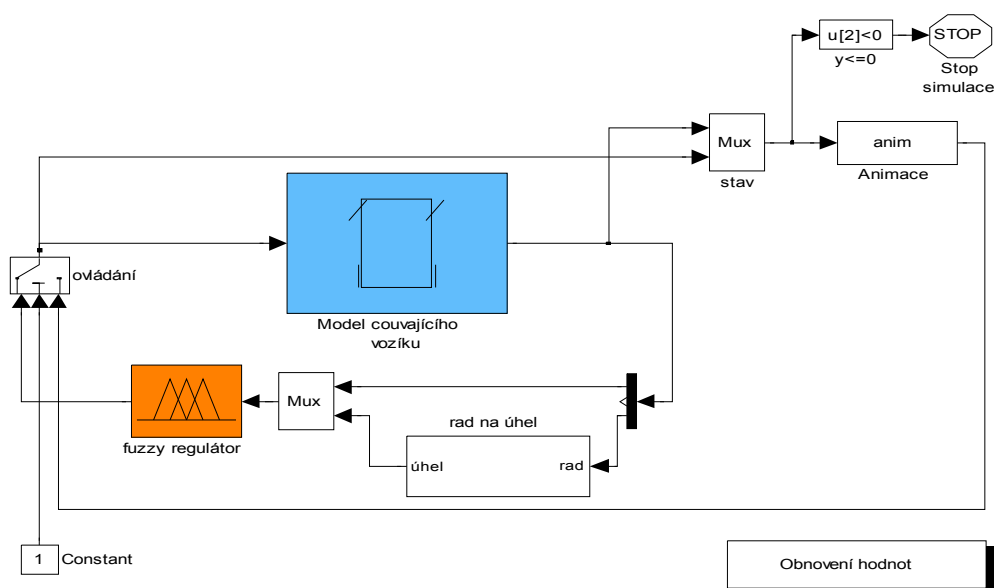
α	L3	L2	L2	L2	L2	L2	L1
	L2	L1	L2	L2	L2	L1	P2
	L1	L2	L1	L1	P2	P2	P2
	C	L2	L1	C	P1	P2	P2
	P1	L2	L2	P1	P1	P2	P2
	P2	L2	L1	L1	P2	P2	P2
	P3	P2	P2	P2	P1	P2	P2
	L3	L2	L1	C	P1	P2	P3
				x			

Obrázek 4.4: Báze pravidel

4.5 Řízení couvajícího vozíku

Model řízení couvajícího vozíku v Simulinku na obrázku 4.5 také vychází z příkladu sltbu. V modelu je jiný regulátor navíc je zde zmiňovaný převodník radiánu na úhel a pozměňená funkce anim, která zajišťuje výstup, jak je vidět třeba na obrázku 4.7. Báze znalostí fuzzy regulátoru se vytváří ve fuzzy logic toolboxu v programu fis editor. Zavedení popsaných poznatků je intuitivní záležitost, stejně jako export báze znalostí, které fuzzy regulátor využívá. Více informací je obsaženo v nápovědě fuzzy logic toolboxu v prostředí programu Matlab.

Výstupní okno umožňuje uchopit couvající vozík myší a umístit jej kdekoli na ploše okna a spustit simulaci. Také je zde možnost změny počátečního stavu natočení vozíku, kdy snadno myší chytíme zadní kolo vozíku a otočíme pohybem myši.

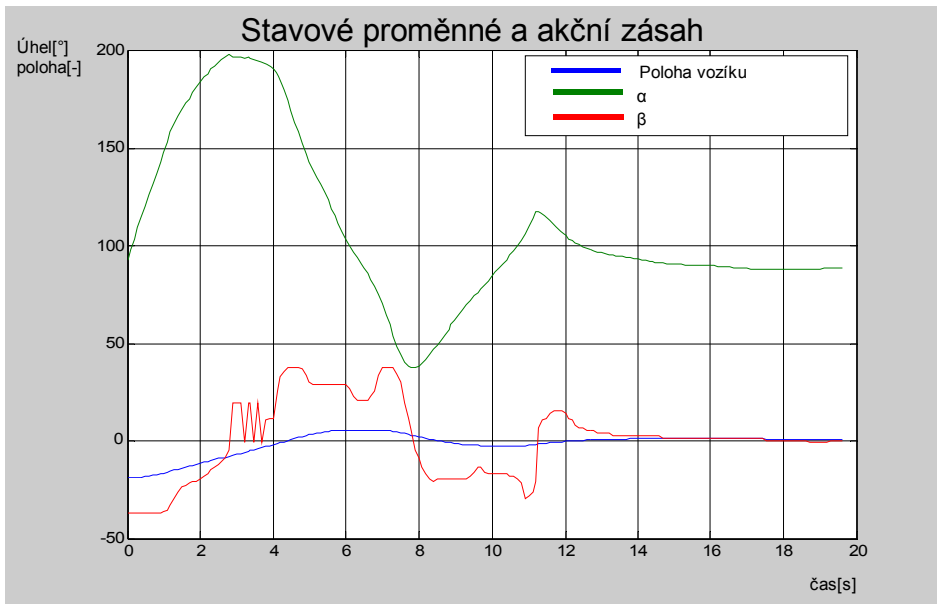


Obrázek 4.5: Model řízení v Simulinku

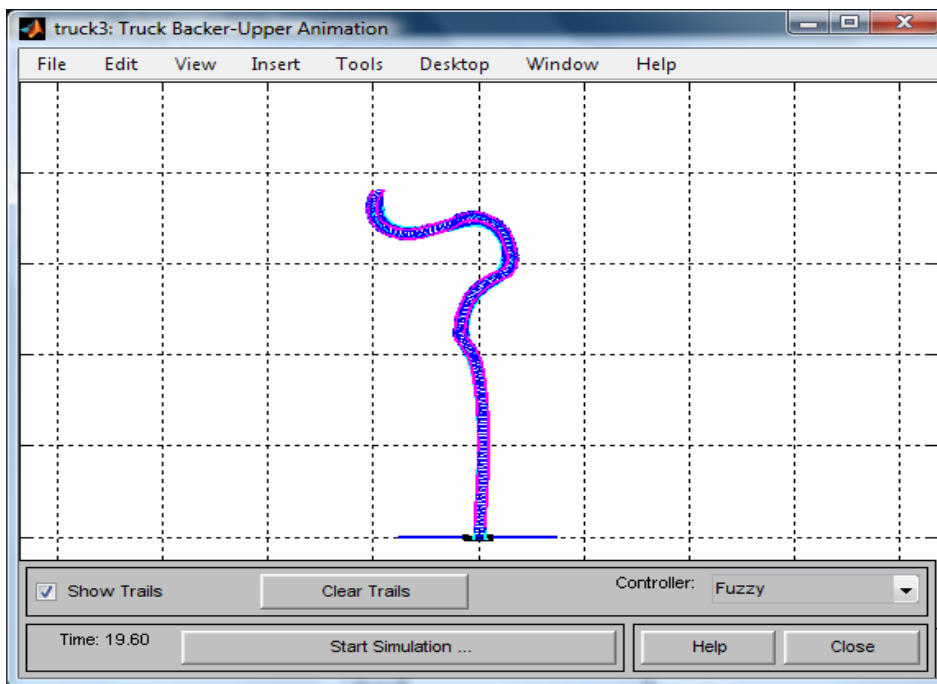
Na obrázku 4.6 vidíme měřené stavové veličiny, použité pro řízení fuzzy regulátorem a výstup natočení předních kol při couvání do cílové oblasti. Dráha vozíku je pak znázorněna na obrázku 4.7. Můžeme si všimnout například vztahu natáčení kol, tedy červená čára obrázku 4.6, k natočení vozíku k přímce kolmé na cílovou oblast, tedy zelené čáry obrázku 4.6.

Jelikož typ řízení neuvažuje souřadnici y a couvající vozík by v určitých polohách na souřadnici x jednoduše narazil do bariéry, tvořenou nulovou hodnotou osy y , proto jsem jednoduchým zásahem do pravidel zvolil strategii zobrazenou na obrázku 4.8 a 4.9, kdy se vozík pootočí tak, aby poodjel dál od osy y .

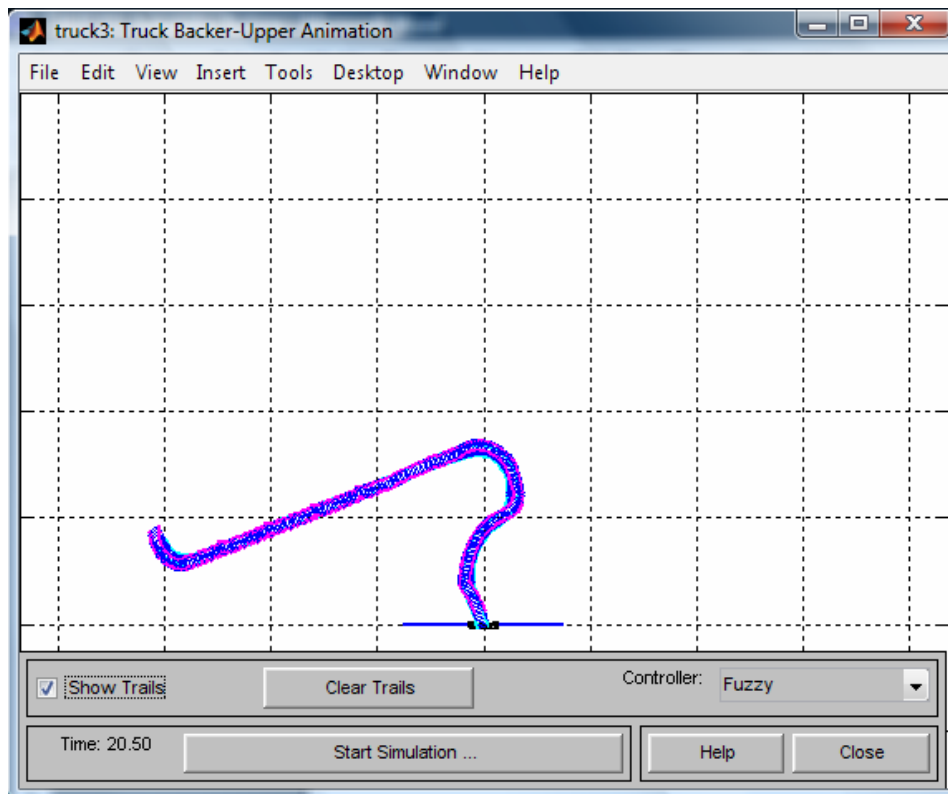
Na obrázcích 4.10 a 4.11 můžeme vidět řízení vozíku z různých poloh, kdy jednou je otočen předem a jednou zády k cílové oblasti. Na obrázcích si můžeme také všimnout překmitu na ose x což má negativní vliv pouze v případě, když vozík couvá z poloh příliš blízko ose y , kdy omezení úhlu natočení kol nedovoluje větší zásah do řízení soustavy. Tento případ je vidět na obrázku 4.13.



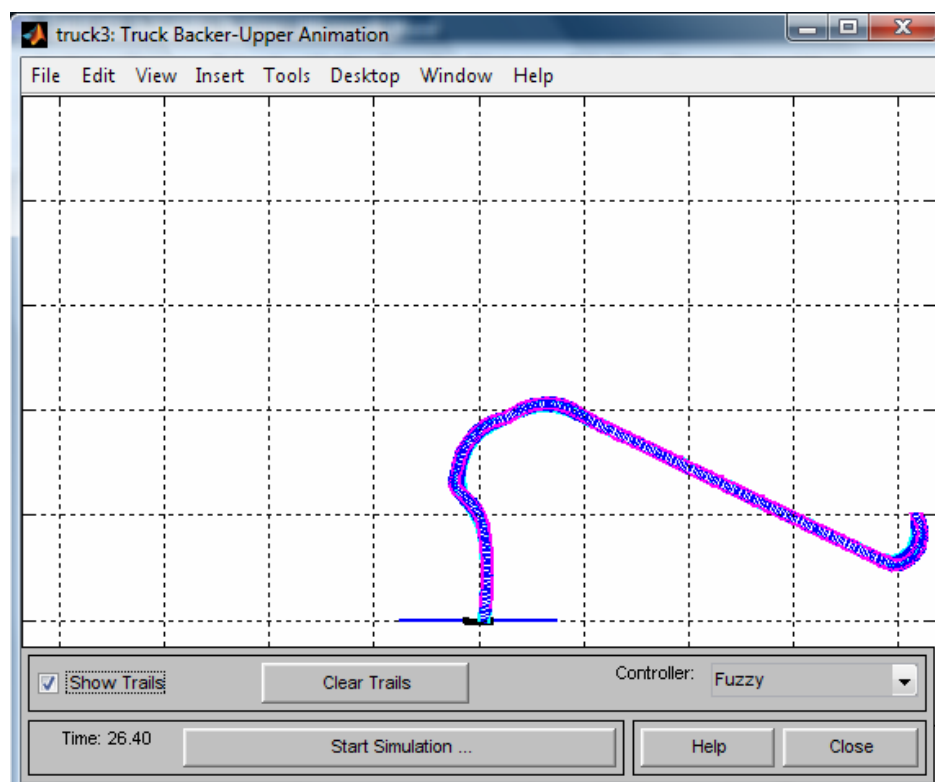
Obrázek 4.6: Výsledky realizace



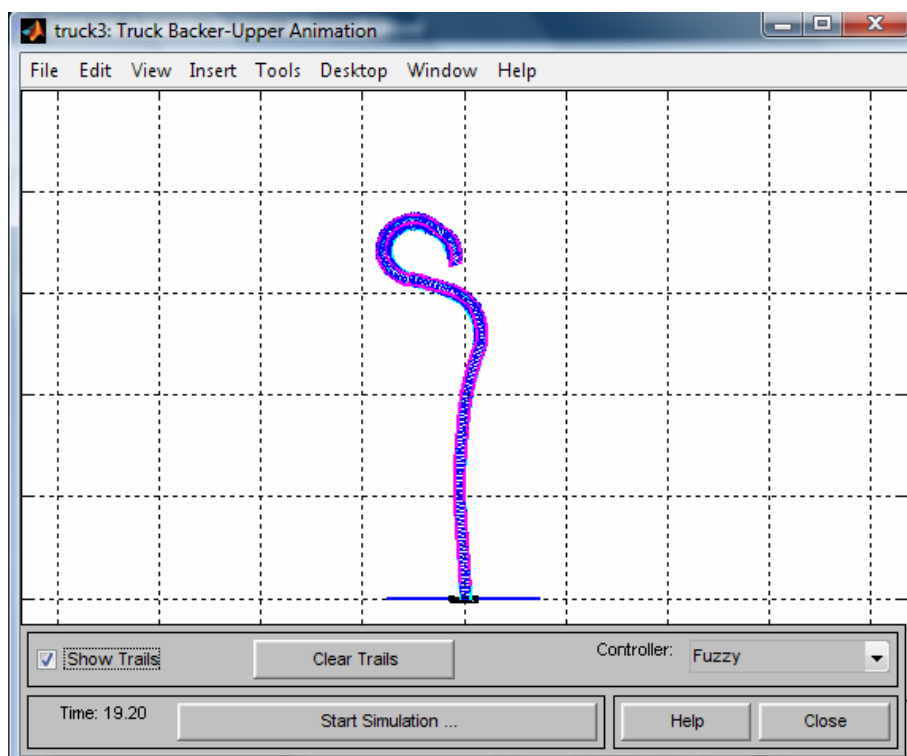
Obrázek 4.7: Výsledky realizace



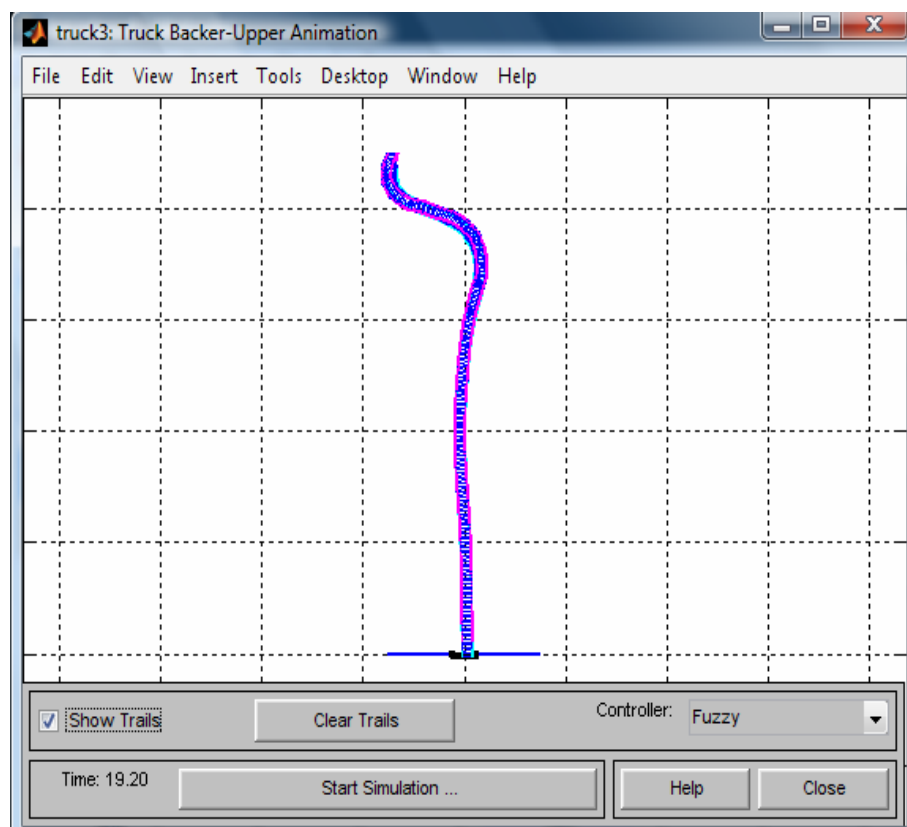
Obrázek 4.8: Výsledky realizace



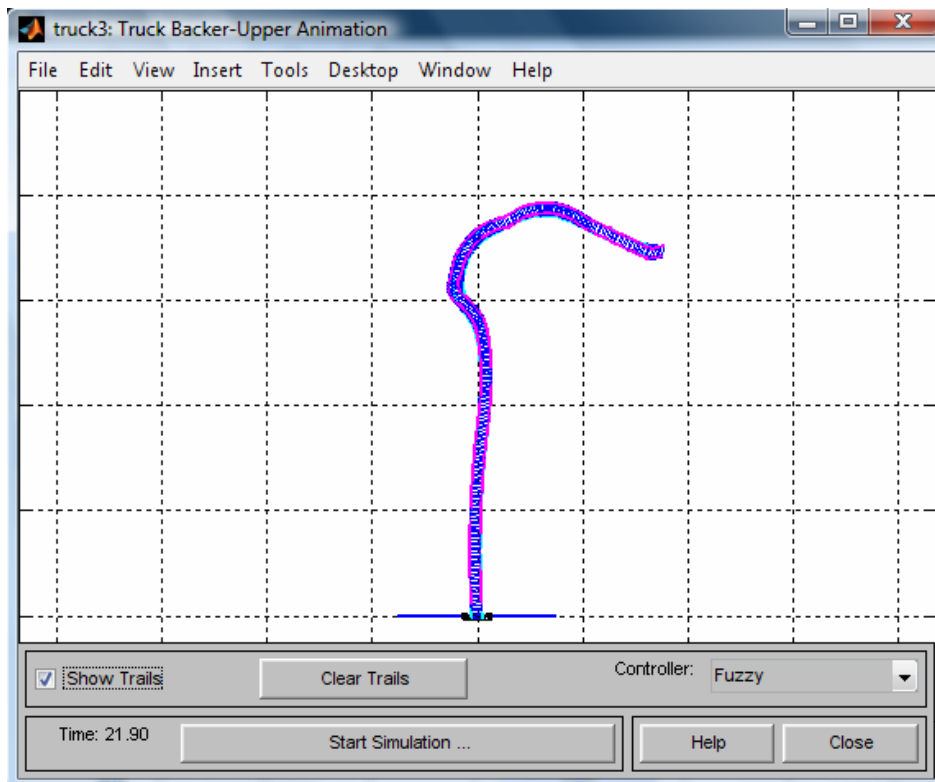
Obrázek 4.9: Výsledky realizace



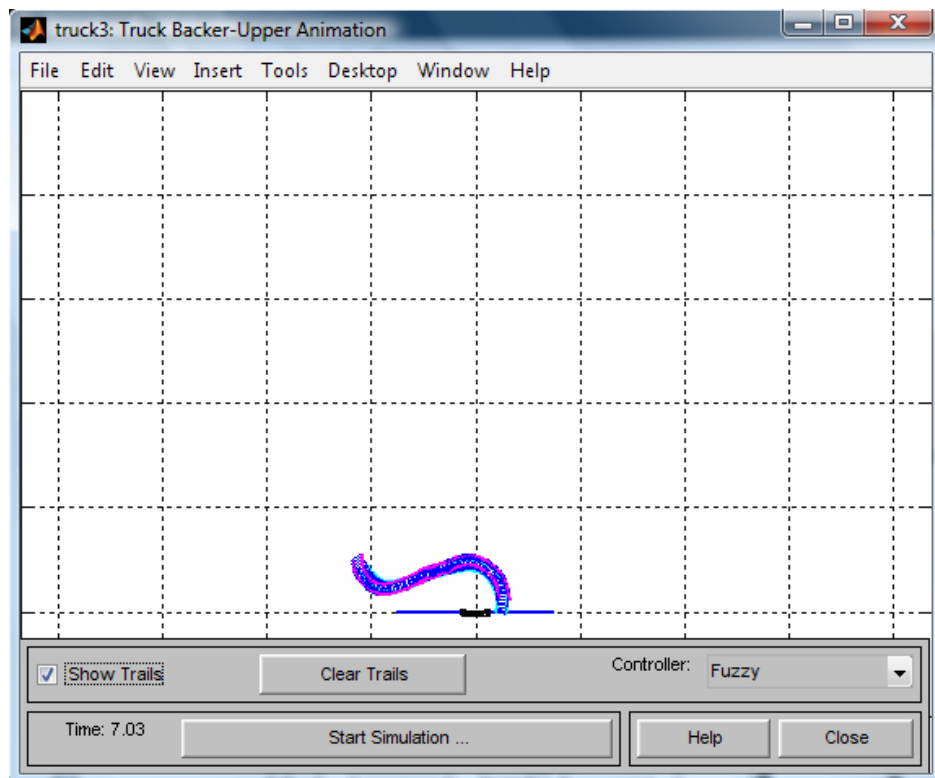
Obrázek 4.10: Výsledky realizace



Obrázek 4.11: Výsledky realizace



Obrázek 4.12: Výsledky realizace



Obrázek 4.13: Výsledky realizace

Závěr

Fuzzy řízení v aplikaci nelineárního systému couvajícího vozíku se ukázalo jako velice silná metoda řízení, což je zřetelné v kapitole 4. Dnes je v mnohých provedení fuzzy přístup pro podobné aplikace, jako je couvající vozík, preferován a to především v řízení nelineárních systémů

Výhody fuzzy přístupu oproti klasickému řízení jsou zřetelné z práce. Jedná se především o snadný návrh a ladění regulátoru, kde pro nelineární systém couvajícího vozíku není nutná složitá nelineární teorie řízení. Fuzzy regulace je jednoduchým a rychlým řešením problému řízení převážně nelineárních soustav. Výhodou je i dosažitelná doba regulace, při dobrém návrhu regulátoru, oproti klasickému řízení. Řízení je snadno čitelné před i po ladění fuzzy regulátoru z důvodu formulovatelných znalostí experta. Fuzzy přístup má také důležitou aproximační vlastnost spojitě funkce na uzavřeném univerzu.

Nevýhodou fuzzy přístupu je obtížné zaručení vlastností. Především se jedná o otázku spojenou s pojmem stability fuzzy systému. Za nevýhodu se také považuje určení správného počtu pravidel znalostní báze.

Literatura

- [1] Wang, L.X.: A Course in Fuzzy Systems and Control, Prentice Hall, 1997.
- [2] Navara, M., Olšák, P.: Základy fuzzy množin ,ČVUT Praha, 2007.
- [3] Zadeh, L.A.:Fuzzy sets ,Department of Electrical Engineering and Electronics ResearchLaboratory,University of California, Berkeley,1965.
- [4] Wang, L.X., Mendet, J.:Generating Fuzzy Rules by Learning from Examples,IEEE Transaction on Systems and Cybernetics, vol. 22, No. 6 November/December 1992 .
- [5] Passino, K.M., Yurkovich, S.:Fuzzy Control , Addison Wesley Longman, 1998
- [6] Kasabov, N.K.: Foundations of Neural Networks, Fuzzy Systems, and Knowledge Engineering, The MIT Press, 1996
- [7] Jang, R.: animtbu.m, The MathWorks, 2004
- [8] Jang, R.: sltbu.fig , The MathWorks, 2005

Příloha A

Obsah přiloženého CD

Adresář text práce:	Text bakalářské práce v pdf
Adresář java program:	Jednoduchý program na určení pravidel
Adresář sim:	Model a kód realizace řízení vozíku
Adresář sim:	pdf návod k řízení vozíku
Adresář sim:	funkce start.m pro snadné spuštění
Adresář sim:	naměřené hodnoty