

Neizrazito, evolucijsko i neuroračunarstvo

Neizrazito upravljanje.

prof.dr.sc. Bojana Dalbelo Bašić prof.dr.sc. Marin Golub
dr.sc. Marko Čupić

Fakultet elektrotehnike i računarstva
Sveučilište u Zagrebu
Akademska godina 2013./2014.

31. listopada 2013.

Neizrazito AKO-ONDA pravila

Pretpostavimo da imamo vrlo složen sustav čiji analitički oblik prijenosne karakteristike (tj. ovisnost izlaza o ulazu) ne znamo.

- Prijenosnu karakteristiku možemo opisati AKO-ONDA pravilima.
- Pravila mogu biti iskustvena (ekspertno znanje) ili dobivena na temelju eksperimenta.
- Svako AKO-ONDA pravilo možemo interpretirati kao binarnu relaciju $A \times B$.
- Unija takvih relacija približno opisuje ponašanje sustava i naziva se **neizraziti graf**.
- Neizraziti graf predstavlja aproksimaciju prave zavisnosti izlaznih varijabli od ulaznih.

Modeliranje nelinearnih sustava

Primjer 1. Promotrimo sustav koji na temelju ulaznog napona generira izlazni napon. Iskustveno, došli smo do sljedećih spoznaja.

Ulazni napon	1V	2V	3V	4V	5V
Izlazni napon	$\approx 1V$	$\approx 3V$	$\approx 4V$	$\approx 2V$	$\approx 1V$

Takav sustav možemo opisati uporabom 5 AKO-ONDA pravila. Modelirajmo neizrazit skup "oko n " \wedge -funkcijom centriranom na n i simetrične širine ± 3 . Npr.

$$\text{"približno } 3V\text{"} = \frac{0.33}{1V} + \frac{0.67}{2V} + \frac{1.00}{3V} + \frac{0.67}{4V} + \frac{0.33}{5V}.$$

Modeliranje nelinearnih sustava

$R_1 = \text{ulazni napon } 1V \times \text{izlazni napon oko } 1V$

$$= \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \times \begin{bmatrix} 1 & 0.67 & 0.33 & 0 & 0 \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 0.67 & 0.33 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Modeliranje nelinearnih sustava

$R_2 = \text{ulazni napon } 2V \times \text{izlazni napon oko } 3V$

$$= \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} \times [0.33 \quad 0.67 \quad 1 \quad 0.67 \quad 0.33]$$

$$= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0.33 & 0.67 & 1 & 0.67 & 0.33 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Na sličan način izračunamo relacije R_3 , R_4 i R_5 koje odgovaraju preostalim trima pravilima.

Modeliranje nelinearnih sustava

Konačna relacija koja opisuje odnos ulaznog i izlaznog napona dana je unijom tih relacija:

$$R = R_1 \cup R_2 \cup R_3 \cup R_4 \cup R_5$$

$$= \begin{bmatrix} 1.00 & 0.67 & 0.33 & 0.00 & 0.00 \\ 0.33 & 0.67 & 1.00 & 0.67 & 0.33 \\ 0.00 & 0.33 & 0.67 & 1.00 & 0.67 \\ 0.67 & 1.00 & 0.67 & 0.33 & 0.00 \\ 1.00 & 0.67 & 0.33 & 0.00 & 0.00 \end{bmatrix}$$

Modeliranje nelinearnih sustava

Sada možemo provjeriti što ćemo dobiti za ulazni napon 4V:

$$Y = X \circ R$$

$$\begin{aligned}
 &= [0.0 \quad 0.0 \quad 0.0 \quad 1.0 \quad 0.0] \circ \begin{bmatrix} 1.00 & 0.67 & 0.33 & 0.00 & 0.00 \\ 0.33 & 0.67 & 1.00 & 0.67 & 0.33 \\ 0.00 & 0.33 & 0.67 & 1.00 & 0.67 \\ 0.67 & 1.00 & 0.67 & 0.33 & 0.00 \\ 1.00 & 0.67 & 0.33 & 0.00 & 0.00 \end{bmatrix} \\
 &= [0.67 \quad 1.00 \quad 0.67 \quad 0.33 \quad 0.00] \\
 &= \frac{0.67}{1V} + \frac{1.00}{2V} + \frac{0.67}{3V} + \frac{0.33}{4V} + \frac{0.0}{5V} \equiv \text{"približno } 2V\text{"}.
 \end{aligned}$$

Modeliranje nelinearnih sustava

Primjer 2. Neka imamo sustav modeliran sljedećim AKO-ONDA pravilima:

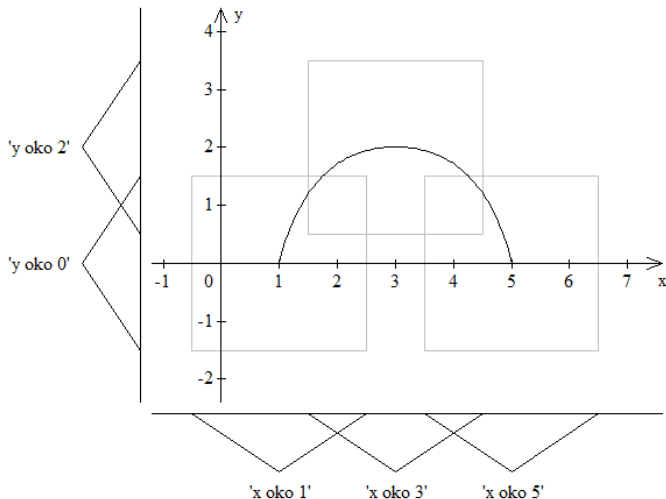
- **AKO je "x oko 1" ONDA je "y oko 0",**
- **AKO je "x oko 3" ONDA je "y oko 2",**
- **AKO je "x oko 5" ONDA je "y oko 0".**

pri čemu su:

- "x oko 1" = FuzzyTrokut(-0.5,1,2.5),
- "x oko 3" = FuzzyTrokut(1.5,3,4.5),
- "x oko 5" = FuzzyTrokut(3.5,5,6.5),
- "y oko 0" = FuzzyTrokut(-1.5,0,1.5),
- "y oko 2" = FuzzyTrokut(0.5,2,3.5).

Modeliranje nelinearnih sustava

Ovaj opis definira **neizraziti graf** prikazan u nastavku.



Izlazak iz neizrazitosti

Već smo u prethodnom primjeru vidjeli da je zaključak koji dobijemo kompozicijom neizrazit skup. Radimo li sustav koji treba generirati neki upravljački izlaz, na taj izlaz možemo dovesti jedino čistu (engl. *crisp*) vrijednost.

- **Dekodiranje neizrazitosti** (engl. *defuzzyfication*) je postupak određivanja klasičnog reprezentanta koji najbolje opisuje "karakter" (odnosno značenje) neizrazitog skupa.
- Postoji niz metoda koje se u praksi koriste:
 - centar površine (engl. *CenterOfArea*, COA)
 - bisektor površine (engl. *BisectorOfArea*, BOA)
 - najmanji od maksimuma (engl. *SmallestOfMax*)
 - najveći od maksimuma (engl. *LargestOfMax*)
 - srednji maksimum (engl. *MeanOfMax*)
 - ...

Pojašnjenja

- Metoda *centar površine* određuje točku na x-osi koja u fizikalnom smislu odgovara težištu lika koji tvori funkcija pripadnosti neizrazitoga skupa.
- Metoda *bisektora površine* određuje onu točku na x-osi koja predstavlja točku kroz koju prolazi okomiti pravac koji lik što ga tvori funkcija pripadnosti neizrazitoga skupa dijeli na dvije površinom jednake polovice.
- Metoda *najmanjeg maksimuma* određuje najmanji element domene za koji je funkcija pripadnosti neizrazitog skupa maksimalna.
- Metoda *najvećeg maksimuma* određuje najveći element domene za koji je funkcija pripadnosti neizrazitog skupa maksimalna.
- Metoda *srednjeg maksimuma* određuje srednju vrijednosti onih točaka za koje je funkcija pripadnosti neizrazitog skupa maksimalna.

Metoda *centar površine*

Predstavnik se računa za konačni neizraziti skup A prema izrazu:

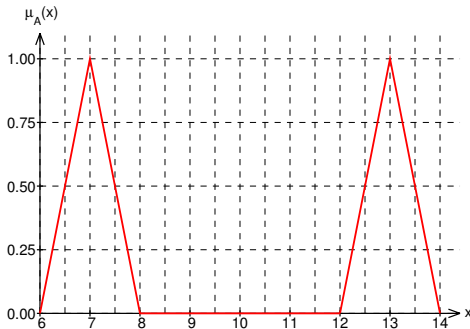
$$X_{CoA}(A) = \frac{\sum_i \mu_A(x_i) \cdot x_i}{\sum_i \mu_A(x_i)} \quad (1)$$

dok se za kontinuirane skupove koristi:

$$X_{CoA}(A) = \frac{\int \mu_A(x) \cdot x \, dx}{\int \mu_A(x) \, dx}. \quad (2)$$

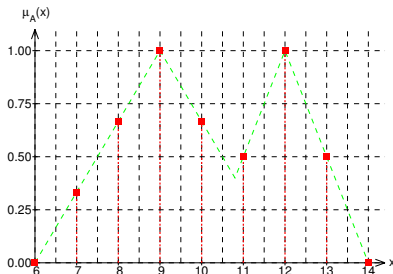
Metoda *bisektor površine*

Metoda *bisektor površine* u općem slučaju ne daje jednoznačno rješenje. Primjer je ilustriran u nastavku. Radimo s neizrazitim skupom definiranim nad univerzalnim skupom relanih brojeva – kao rezultat dekodiranja neizrazitosti ovom metodom može se uzeti bilo koji x iz skupa $[8, 12]$, a takvih je beskonačno.



Primjer

Neizraziti skup A definiran je nad univerzalnim skupom $U = \{6, 7, 8, 9, 10, 11, 12, 13, 14\}$ kao:



- COA=?,
- MOM=?

Primjer

$$A = \left\{ \frac{0}{6} + \frac{1}{7} + \frac{2}{8} + \frac{1}{9} + \frac{2}{10} + \frac{1}{11} + \frac{1}{12} + \frac{1}{13} + \frac{0}{14} \right\}.$$

Dekodiranje metodom *CenterOfArea* daje:

$$\begin{aligned} X_{CoA}(A) &= \frac{\sum_i \mu_A(x_i) \cdot x_i}{\sum_i \mu_A(x_i)} \\ &= \frac{0 \cdot 6 + \frac{1}{3} \cdot 7 + \frac{2}{3} \cdot 8 + 1 \cdot 9 + \frac{2}{3} \cdot 10 + \frac{1}{2} \cdot 11 + 1 \cdot 12 + \frac{1}{2} \cdot 13 + 0 \cdot 14}{0 + \frac{1}{3} + \frac{2}{3} + 1 + \frac{2}{3} + \frac{1}{2} + 1 + \frac{1}{2} + 0} \\ &= \frac{\frac{142}{3}}{\frac{14}{3}} \\ &= \frac{71}{7} \approx 10.143 \end{aligned}$$

Primjer

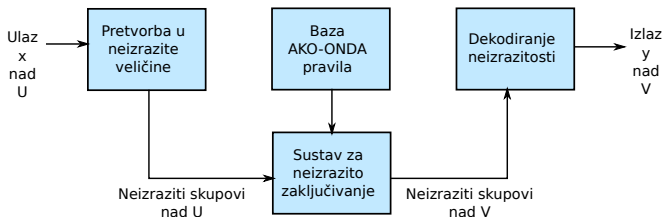
$$A = \left\{ \frac{0}{6} + \frac{\frac{1}{3}}{7} + \frac{\frac{2}{3}}{8} + \frac{1}{9} + \frac{\frac{2}{3}}{10} + \frac{\frac{1}{2}}{11} + \frac{1}{12} + \frac{\frac{1}{2}}{13} + \frac{0}{14} \right\}.$$

Dekodiranje metodom *MeanOfMax* daje:

$$\begin{aligned} X_{MoM}(A) &= \frac{9 + 12}{2} \\ &= \frac{21}{2} \\ &= 10.5 \end{aligned}$$

Građa produkcijskog sustava neizrazitog upravljanja

- Razmatramo *produkcijske sustave*
- Njihova baza znanja sastoji se od niza **AKO-ONDA** pravila.
- Građa jednog takvog sustava prikazana je u nastavku.



Građa produkcijskog sustava neizrazitog upravljanja

- kao ulaz se dobiva r podataka (x_1 do x_r) nad univerzalnim skupovima U_1 do U_n
- baza pravila sastoji od n_R
- A_{ij} neizraziti skup definiran nad univerzalim skupom U_j
- svi neizraziti skupovi B_i definirani su nad istim univerzalnim skupom V

Građa produkcijskog sustava neizrazitog upravljanja

Neka su pravila:

- **AKO** x_1 je A_{11} I \dots I x_r je A_{1r} **ONDA** y je B_1
- ...
- **AKO** x_1 je A_{i1} I \dots I x_r je A_{ir} **ONDA** y je B_i
- ...
- **AKO** x_1 je A_{n_R1} I \dots I x_r je A_{n_Rr} **ONDA** y je B_{n_R}

gdje su x_1 do x_r , općenito govoreći, neizraziti skupovi koji predstavljaju trenutne ulaze sustava.

Načini zaključivanja u produkcijskim sustavima neizrazitog upravljanja

U produkcijskim sustavima neizrazitog upravljanja koriste se dva uobičajena načina zaključivanja:

- 1 zaključivanje temeljeno na kompoziciji te
- 2 zaključivanje temeljeno na pojedinačnim pravilima.

Zaključivanje temeljeno na kompoziciji

- razmotrimo i -to pravilo
- antecedent je konjunkcija r zahtjeva
- čitav antecedent može se prikazati kao neizrazita relacija koja je definirana nad $U_1 \times \dots \times U_r$ kao kartezijev produkt neizrazitih skupova A_{i1} do A_{ir} :

$$\mu_{A_{i1} \times \dots \times A_{ir}}(x_1, \dots, x_r) = T(\mu_{A_{i1}}(x_1), \dots, \mu_{A_{ir}}(x_r)).$$

- $T(\dots)$ predstavlja bilo koju t -normu a uobičajeno je koristiti minimum
- i -to **AKO-ONDA** pravilo možemo promatrati kao pravilo čiji je antecedent upravo izvedeni neizraziti skup a konsekvent drugi neizraziti skup
- čitavo pravilo predstaviti ćemo kao jednu relaciju definiranu nad univerzalnim skupom $U_1 \times \dots \times U_r \times V$ uporabom bilo kojeg od operatora neizrazite implikacije

Zaključivanje temeljeno na kompoziciji

- temeljem i -tog pravila dobili smo relaciju R_i
- ako imamo n_R pravila, dobit ćemo ukupno n_R relacija:
 R_1, \dots, R_{n_R}

Sljedeći korak je izgradnja konačne relacije koja kombinira relacije R_i .

- ako je korištena implikacija lokalne semantike, konačna relacija je unija svih relacija:

$$R = R_1 \cup \dots \cup R_{n_R}.$$

- ako je korištena implikacija s globalnom semantikom, konačna relacija je presjek svih relacija:

$$R = R_1 \cap \dots \cap R_{n_R}.$$

Zaključivanje temeljeno na kompoziciji

- Nakon generiranja relacije R , zaključivanje se svodi na uporabu *modus ponensa*.
- Neka je korisnik kao ulaz neizrazitog sustava definirao x_1 je C_1 , \dots , x_r je C_r , gdje su C_1 do C_r neizraziti skupovi.
- Gradimo neizraziti skup C kao kartezijev produkt neizrazitih skupova $C_1 \times \dots \times C_r$ (uporabom neke t -norme) i potom zaključak izvodimo kao kompoziciju neizrazitog skupa C i relacije R :

$$\mu_B(y) = \sup_{x_1, \dots, x_r} T(\mu_C(x_1, \dots, x_r), \mu_R(x_1, \dots, x_r, y)). \quad (3)$$

Zaključivanje temeljeno na kompoziciji

- Ako su ulazni neizraziti skupovi *jednočlani skupovi* (engl. *singletons*), izračun zaključka bitno se pojednostavljuje.
- Označimo u svakom od neizrazitih skupova C_i oznakom x_i^* taj element domene za koji je μ_{C_i} jednaka 1:

$$\mu_{C_i}(x_i) = \begin{cases} 1, & x_i = x_i^* \\ 0, & \text{inače} \end{cases}$$

- Tada vrijedi:

$$\mu_B(y) = \mu_R(x_1^*, \dots, x_r^*, y). \quad (4)$$

što se, uz pretpostavku da je R unaprijed izračunat, može vrlo efikasno odrediti (nije potrebno provoditi kompoziciju).

Zaključivanje temeljeno na pojedinačnim pravilima

- Postupak je sličan kao kod zaključivanja temeljenog na kompoziciji:
 - razlika je što se ne gradi konačna relacija već se direktno koriste relacije koje odgovaraju svakom od pravila
 - svako pravilo generira svoj zaključak (neizraziti skup)
 - svi se zaključci kombiniraju u konačni zaključak

Zaključivanje temeljeno na pojedinačnim pravilima

- za i -to pravilo gradimo relaciju $R_i(x_1, \dots, x_r, y)$ na jednak način kako smo to napravili kod zaključivanja temeljenog na kompoziciji
- neka je $C(x_1, \dots, x_r)$ relacija koja je dobivena kao kartezijev produkt neizrazitih skupova C_i (ulaza u sustav), koristeći odabranu t -normu
- za svako pravilo računamo neizrazit skup koji predstavlja lokalni zaključak B'_i kao generalizirani modus ponens pri čemu je prva premisa " x je C " a druga promatrano pravilo
- tako dobiveni zaključak je neizraziti skup sa sljedećom funkcijom pripadnosti:

$$\mu_{B'_i}(y) = \sup_{x_1, \dots, x_r} T(\mu_C(x_1, \dots, x_r), \mu_{R_i}(x_1, \dots, x_r, y)). \quad (5)$$

Zaključivanje temeljeno na pojedinačnim pravilima

- provođenjem prethodnog postupka završavamo s n_R različitih zaključaka (svako pravilo generiralo je jedan; to su neizraziti skupovi B'_1 do B'_{n_R})

Konačan zaključak generira se kombiniranjem prethodno izvedenih n_R zaključaka.

- ako je korištena implikacija koja pravilu daje lokalnu semantiku, kombinacija je unija:

$$\mu_{B'}(y) = \bigcup_{i=1, \dots, n_R} \mu_{B'_i}(y) = S(\mu_{B'_1}(y), \dots, \mu_{B'_{n_R}}(y)) \quad (6)$$

- ako je korištena implikacija koja pravilu daje globalnu semantiku, kombinacija je presjek:

$$\mu_{B'}(y) = \bigcap_{i=1, \dots, n_R} \mu_{B'_i}(y) = T(\mu_{B'_1}(y), \dots, \mu_{B'_{n_R}}(y)). \quad (7)$$

Strojevi za zaključivanje koji se temelji na minimumu

Stroj za zaključivanje koji se temelji na minimumu (engl. *Minimum inference engine*) je implementacija zaključivanja koje:

- se temelji na zaključivanju prema pojedinačnim pravilima,
- kao operator implikacije koristi Mamdanijevu minimum-implikaciju,
- za sve s -norme koristi operator maksimuma,
- za sve t -norme koristi operator minimuma te
- pojedinačne zaključke svakog pravila kombinira unijom u konačan zaključak.

Strojevi za zaključivanje koji se temelji na minimumu

$$\mu_{B'}(y) = \max_{i=1, \dots, n_R} \left(\sup_{x_1, \dots, x_r} \min(\mu_C(x_1, \dots, x_r), \mu_{R_i}(x_1, \dots, x_r, y)) \right) \quad (8)$$

$$= \max_{i=1, \dots, n_R} \left\{ \sup_{x_1, \dots, x_r} \min(\mu_C(x_1, \dots, x_r), \min(\mu_{A_{i1}}(x_1), \dots, \mu_{A_{ir}}(x_r), \mu_{B_i}(y))) \right\} \quad (9)$$

$$= \max_{i=1, \dots, n_R} \left(\sup_{x_1, \dots, x_r} \min(\mu_{C_1}(x_1), \dots, \mu_{C_r}(x_r), \mu_{A_{i1}}(x_1), \dots, \mu_{A_{ir}}(x_r), \mu_{B_i}(y)) \right) \quad (10)$$

Strojevi za zaključivanje koji se temelji na minimumu

U slučaju da se kao ulaz koriste singleton neizraziti skupovi i ako označimo s x_1^* do x_r^* konkretne vrijednosti koje jedine pripadaju ulaznim neizrazitim skupovima C_1 do C_r , tada se prethodni izraz pojednostavljuje u:

$$\mu_{B'}(y) = \max_{i=1, \dots, n_R} \{ \min(\mu_{A_{i1}}(x_1^*), \dots, \mu_{A_{ir}}(x_r^*), \mu_{B_i}(y)) \}. \quad (11)$$

Strojevi za zaključivanje koji se temelji na produktu

Stroj za zaključivanje koji se temelji na produktu

Stroj za zaključivanje koji se temelji na produktu (engl. *Product inference engine*) je implementacija zaključivanja koje:

- se temelji na zaključivanju prema pojedinačnim pravilima,
- kao operator implikacije koristi Mamdanijevu produkt-implikaciju,
- za sve *s*-norme koristi operator maksimuma,
- za sve *t*-norme koristi operator produkt te
- pojedinačne zaključke svakog pravila kombinira unijom u konačan zaključak.

Strojevi za zaključivanje koji se temelji na produktu

$$\mu_{B'}(y) = \max_{i=1, \dots, n_R} \left(\sup_{x_1, \dots, x_r} \mu_C(x_1, \dots, x_r) \cdot \mu_{R_i}(x_1, \dots, x_r, y) \right) \quad (12)$$

$$= \max_{i=1, \dots, n_R} \left\{ \sup_{x_1, \dots, x_r} \mu_C(x_1, \dots, x_r) \cdot \left(\prod_{j=1, \dots, r} \mu_{A_{ij}}(x_j) \right) \cdot \mu_{B_i}(y) \right\} \quad (13)$$

$$= \max_{i=1, \dots, n_R} \left\{ \sup_{x_1, \dots, x_r} \left(\prod_{j=1, \dots, r} \mu_{C_j}(x_j) \right) \cdot \left(\prod_{j=1, \dots, r} \mu_{A_{ij}}(x_j) \right) \cdot \mu_{B_i}(y) \right\} \quad (14)$$

Strojevi za zaključivanje koji se temelji na produktu

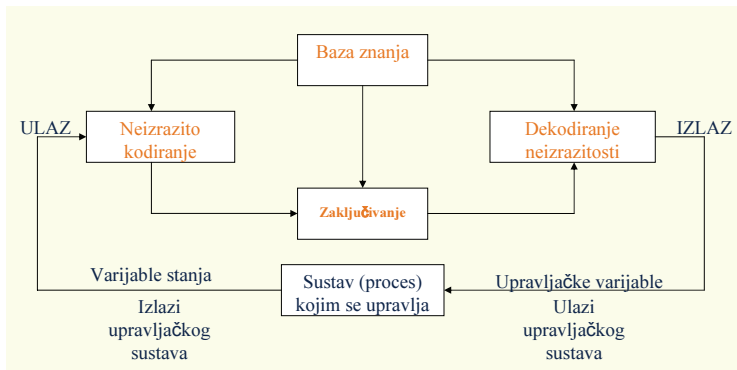
U slučaju da se kao ulaz koriste singleton neizraziti skupovi i ako označimo s x_1^* do x_r^* konkretne vrijednosti koje jedine pripadaju ulaznim neizrazitim skupovima C_1 do C_r , tada se prethodni izraz pojednostavljuje u:

$$\mu_{B'}(y) = \max_{i=1, \dots, n_R} \left\{ \left(\prod_{j=1, \dots, r} \mu_{A_{ij}}(x_j^*) \right) \cdot \mu_{B_i}(y) \right\}. \quad (15)$$

Drugi strojevi za zaključivanje

Još nekoliko drugih strojeva za zaključivanje opisano je u knjizi – pogledati.

Schema upravljačkog sustava



Shema upravljačkog sustava

Četiri su glavne komponente:

- 1 **Sučelje za neizrazito kodiranje:** pretvara ulazne klasične (crisp) vrijednosti u neizrazite vrijednosti
- 2 **Baza znanja:** Sadrži znanje iz domene primjene i ciljeve upravljanja (baza podataka + baza pravila).
- 3 **Logika za zaključivanje:** izvodi zaključivanje za akcije neizrazitog upravljanja
- 4 **Sučelje za dekodiranje neizrazitosti:** prevodi neizrazite zaključke u konkretne vrijednosti koje upravljaju sustavom

Shema upravljačkog sustava

Upravljački sustavi rade s dvije vrste varijabli:

1 Varijable stanja

- ulazne varijable sustava neizrazitog upravljanja
- sadrže informaciju o trenutnom stanju sustava kojim upravljamo
- često se koriste: stanje, pogreška stanja, odstupanje pogreške stanja i sl.

2 Upravljačke varijable

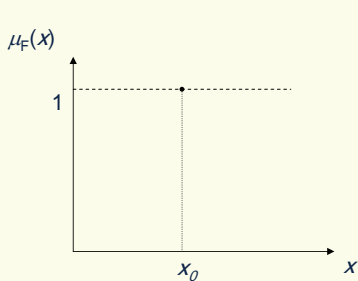
- Izlazne varijable sustava neizrazitog upravljanja

Odabir varijabli – ovisi o znanju eksperta o procesu.

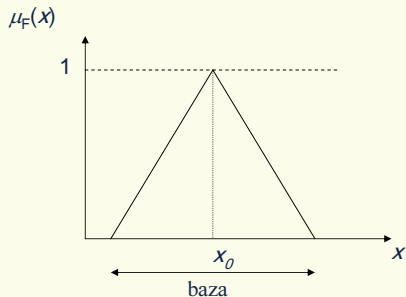
Komponenta: sučelje za neizrazito kodiranje

- ❶ Zadužena za preslikavanje raspona ulaznih podataka
 - pretvara raspon vrijednosti ulaznih podataka u odgovarajući univerzalni skup implementiran u FLC-u (u slučaju normalizacije).
- ❷ Strategija za šum
 - Kada su promatrani podaci poremećeni slučajnim šumom, operator kodiranja neizrazitosti trebao bi pretvoriti vjerojatnosne podatke u neizrazite brojeve.
 - Neizraziti skup s jednim elementom
 - uz pretpostavku da promatrani podatak ne sadrži nesigurnost
 - precizan podatak, nema neizrazitosti (najčešće u FLC)
- ❸ Odabir funkcije za kodiranje neizrazitosti $x = \text{fuzzifier}(x_0)$
 - x_0 je promatrana klasična vrijednost
 - x je neizraziti skup
 - *fuzzifier* predstavlja operator kodiranja neizrazitosti

Komponenta: sučelje za neizrazito kodiranje



Funkcija za kodiranje neizrazitosti za
neizraziti skup s jednim elementom



Funkcija za kodiranje neizrazitosti
za trokutasti neizraziti broj
(baza obično $2x_{st.dev}$)

Komponenta: baza znanja

Čine je baza podataka + baza pravila.

1 Baza podataka

- Glavni dizajn parametara za sustav neizrazitog upravljanja:
 - Diskretizacija i normalizacija univerzalnih skupova
 - Neizrazita podjela (particija) ulaznih i izlaznih prostora
 - Funkcije pripadnosti osnovnih neizrazitih skupova

2 Baza pravila

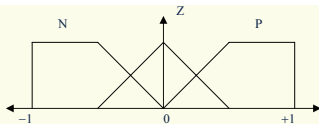
- Izvor pravila neizrazitog upravljanja, koja mogu nastati
 - Heurističkim metodama
 - Determinističkim metodama

Komponenta: baza podataka

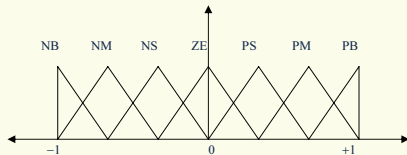
- definira kako se obavlja diskretizacija podataka (rasponi, preslikavanja, normalizacija)
- neizrazita podjela (particija) ulaznih i izlaznih prostora
 - jezična varijabla
 - u antecedent-dijelu pravila tvori neizraziti ulazni prostor
 - u konsekvant-dijelu pravila tvori neizraziti izlazni prostor
 - općenito, povezano sa skupom izraza (engl. *term-set*)
 - neizrazita podjela prostora
 - određuje koliko izraza (vrijednosti jezične varijable) treba postojati u skupu izraza (engl. *term-set*)
 - potrebno za određivanje broja osnovnih neizrazitih skupova (jezičnih izraza)

Komponenta: baza podataka

Primjer neizrazite podjele s 3 odnosno 7 jezičnih vrijednosti.



N: negative, Z: zero, P: positive



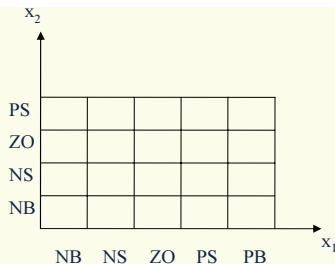
NB, NM, NS, ZE, PS, PM, PB

7 jezičnih izraza koji se često koriste

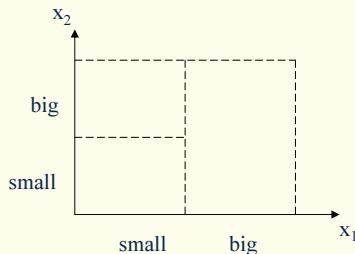
- NB: negative big
- NM: negative medium
- NS: negative small
- ZE: zero
- PS: positive small
- PM: positive medium
- PB: positive big

Komponenta: baza podataka

Različite vrste podjele ulaznog prostora.



Neizrazita podjela
u 2-dimenzionalnom ulaznom prostoru
maksimalni broj upravljačkih pravila = 20 (5x4)



Neizrazita podjela na temelju tri pravila

Komponenta: baza znanja

Četiri načina izvođenja pravila neizrazitog upravljanja:

- 1 Iskustvo eksperta i znanje o metodama upravljanja: priručnici za rad s procesom i upitnici
- 2 Temeljeno na akcijama operatera: promatranjem akcija koje obavlja operater (osoba) u u pogledu ulaznih i izlaznih podataka procesa
- 3 Temeljeno na neizrazitom modelu procesa: jezični opis dinamičkih karakteristika procesa
- 4 Temeljeno na učenju: sposobnost izmjene pravila upravljanja poput samo-organizirajućih sustava upravljanja

Komponenta: baza znanja

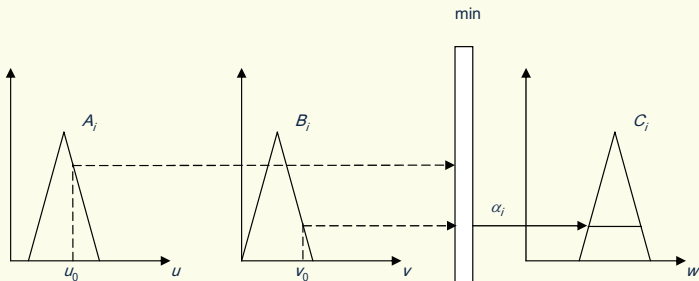
Pravila možemo podijeliti u nekoliko vrsta:

- ➊ MISO (multiple-input-single-output): uobičajena
- ➋ MIMO (multiple-input-multiple-output): možemo ih rastaviti na više MISO pravila

Konsekvent u pravilima ne mora biti neizraziti skup: primjer su pravila koja izlaz definiraju kao funkciju od ulaznih varijabli.

Obrada pravila

Ako je ulaz singleton neizraziti skup:

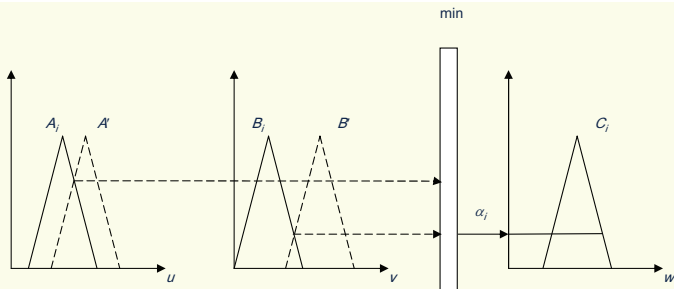


α_i je minimum stupnja slaganja od $A_i(u_0)$ i $B_i(v_0)$.

(stroj za zaključivanje koji se temelji na minimumu!)

Obrada pravila

Ako je ulaz "običan" neizraziti skup:



α_i je minimum stupnja slaganja od (A' and A_i) and (B' and B_i).)

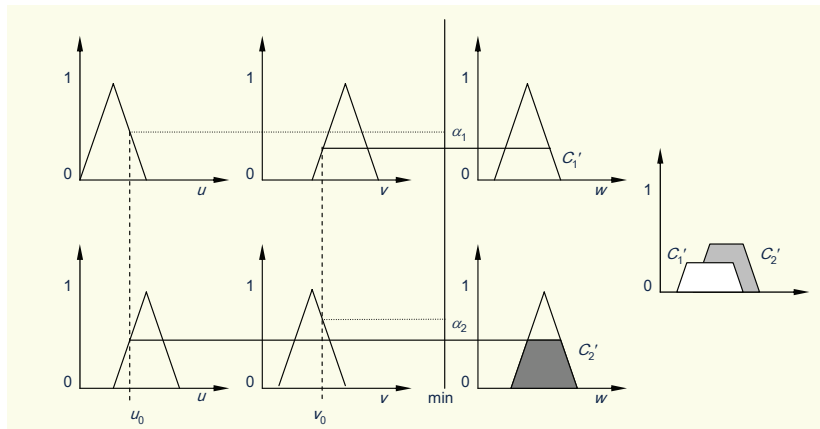
(stroj za zaključivanje koji se temelji na minimumu!)

Mamdanijeva metoda zaključivanja

- Odgovara stroju za zaključivanje koji se temelji na minimumu
- za implikaciju i t -norme koristi se operator *min*
- za uniju se koristi se operator *max*
- vizualno, za svako pravilo utvrđuje se "jakost" slaganja antecedenta i ulaznih podataka te se na tu razinu neizrazit skup konsekventa odsiječe
- pojedini zaključci kombiniraju se u uniju

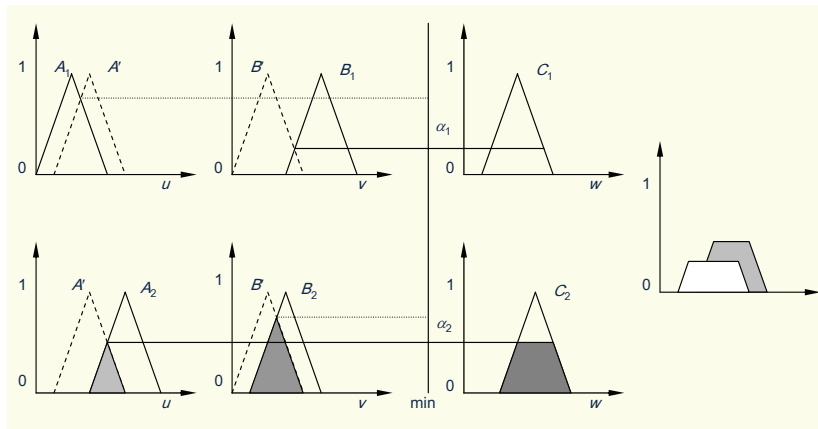
Mamdanijeva metoda zaključivanja

Ako je ulaz singleton neizraziti skup:



Mamdanijeva metoda zaključivanja

Ako je ulaz "običan" neizraziti skup:

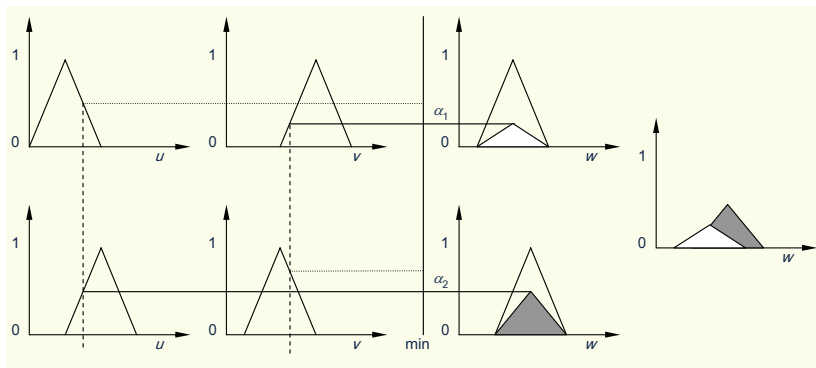


Larsenova metoda zaključivanja

- Odgovara stroju za zaključivanje koji se temelji na produktu
- za implikaciju i t -norme koristi se operator *produkt*
- za uniju se koristi se operator *max*
- vizualno, za svako pravilo utvrđuje se "jakost" slaganja antecedenta i ulaznih podataka te se s tom razinom neizrazit skup konsekventa skalira (čuva oblik ali mijenja visinu)
- pojedini zaključci kombiniraju se u uniju

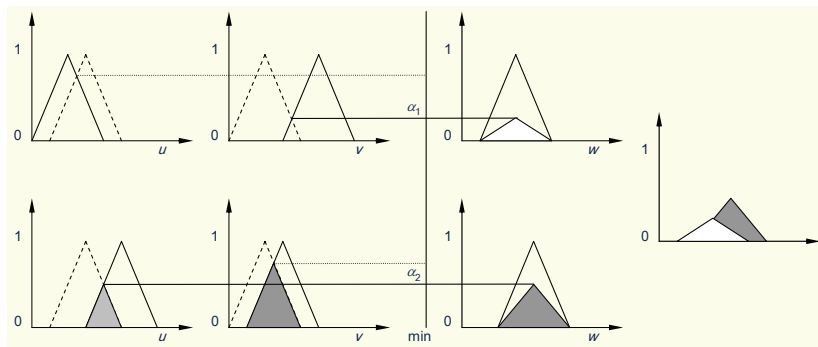
Larsenova metoda zaključivanja

Ako je ulaz singleton neizraziti skup (slika je približno OK – α nije dobar):



Larsenova metoda zaključivanja

Ako je ulaz "običan" neizraziti skup (slika je približno OK – α nije dobar):



Tsukamotova metoda zaključivanja

Tsukamotova metoda zahtjeva da je konsekvent pravila neizraziti skup posebnog oblika: njegova funkcija pripadnosti mora biti monotona (bilo padajuća, bilo rastuća).

- zahvaljujući takvom zahtjevu, postupak zaključivanja može se izvesti računski efikasno
- za svaki antecedent pravila određuje se mjera u kojoj su ulazni podatci u skladu s antecendentom: neka je za pravilo i ta mjera α_i
- neka je konsekvens pravila i neizraziti skup s monotonom funkcijom pripadnosti $\mu_i(w_i)$ nad $w \in \mathcal{W}$
- tada možemo pronaći za koji se w_i^* postiže izračunati α_i , tj. tražimo $\alpha_i = \mu_i(w_i^*)$ (inverznom funkcijom)

Tsukamotova metoda zaključivanja

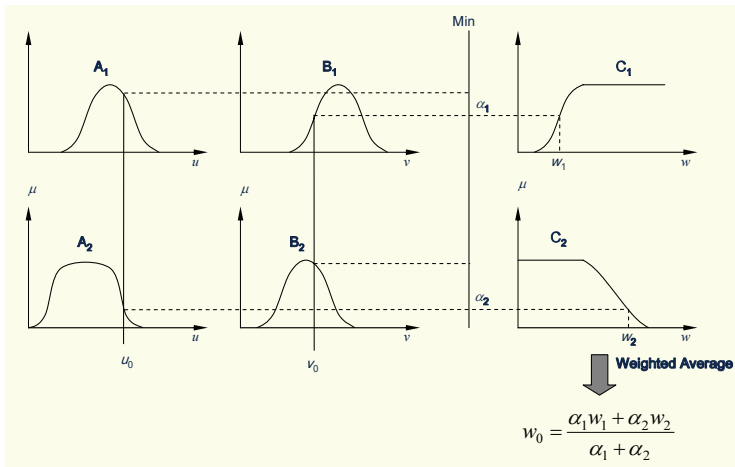
Tsukamotova metoda zahtjeva da je konsekvent pravila neizraziti skup posebnog oblika: njegova funkcija pripadnosti mora biti monotona (bilo padajuća, bilo rastuća).

- ...
- time smo utvrdili da je rezultat izvođenja i -tog pravila jasan zaključak w_i^* koji međutim vrijedi s mjerom α_i
- ukupni zaključak tada je težinska sredina svih zaključaka:

$$w = \frac{\sum_i \alpha_i \cdot w_i^*}{\sum_i \alpha_i}$$

Tsukamotova metoda zaključivanja

Primjer:



TSK metoda zaključivanja

TSK (Takagi-Sugeno-Kang) metoda također koristi različit oblik pravila: konsekvens se daje kao funkcijska ovisnost o ulaznim varijablama. Ideja ovog oblika pravila ilustrirana je sljedećim primjerom.

- pretpostavimo da imamo sustav s dva ulaza x i y te jedinim izlazom w
- pretpostavimo da, kada su x i y *mali* brojevi, izlaz je određen izrazom $w(x, y) = f_1(x, y) = x \cdot \sin(y) + e^x$
- pretpostavimo da, kada je x *velik* broj a y *mali* broj, izlaz je određen izrazom $w(x, y) = f_2(x, y) = (x + y)^2$
- pretpostavimo da, kada su x i y *veliki* brojevi, izlaz je određen izrazom $w(x, y) = f_3(x, y) = (x - y)^2$

TSK metoda zaključivanja

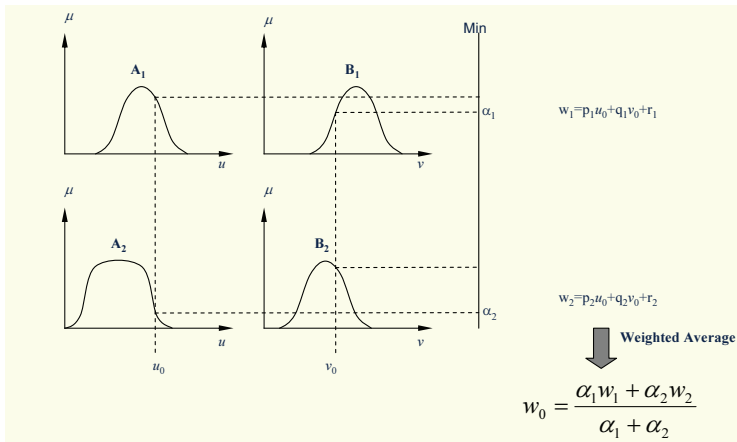
Za neki konkretan ulaz (x, y) , ovisno o načinu kako je napravljena particija ulaznog prostora, moguće je da se antecedenti više od jednog pravila slažu s tim podacima u mjeri koja je veća od 0: stoga će za te podatke svako pravilo generirati i zaključak w_i koji se računa formulom koja je dana u konsekvencu nad trenutnim ulaznim podacima.

- kao i kod Tsukamotove metode, svaki od tih zaključaka w_i vrijedit će određenom mjerom α_i
- konačni zaključak tada će biti težinska sredina svih zaključaka:

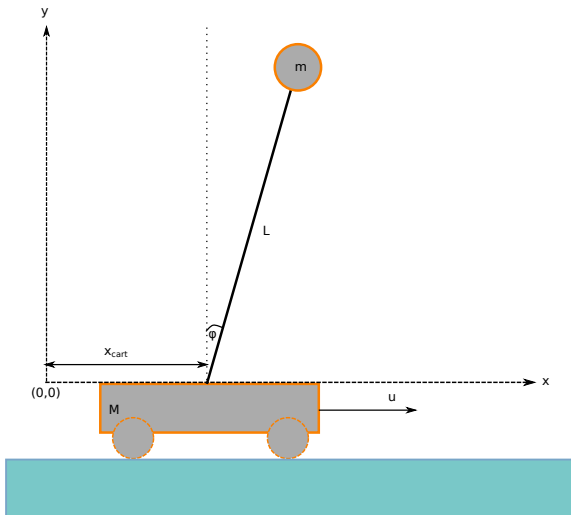
$$w = \frac{\sum_i \alpha_i \cdot f_i(x, y)}{\sum_i \alpha_i} = \frac{\sum_i \alpha_i \cdot w_i^*}{\sum_i \alpha_i}$$

TSK metoda zaključivanja

Primjer:



Stabilizacija okrenutog njihala



Stabilizacija okrenutog njihala

U početnom položaju, njihalo se nalazi u nekom gornjem položaju, i sila teže ga prirodno okreće prema dolje.

- potrebno je djelovati na kolica silom \vec{u} kako bi se njihalo stabiliziralo u okomitom položaju
- fizikalni model koji u obzir uzima mase i inercije svodi se na sustav dvije diferencijalne jednačbe drugog reda:

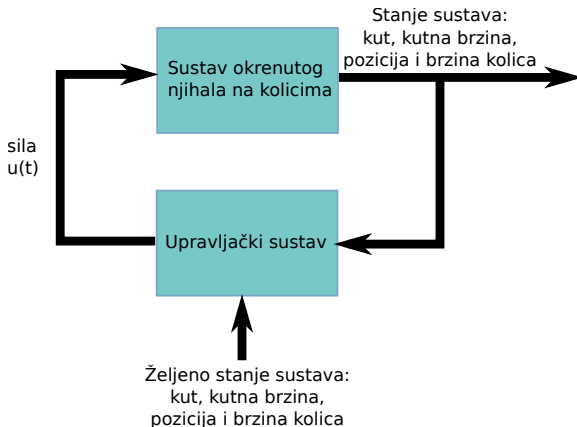
$$(M + m)\ddot{x} - mL \sin(\phi)\dot{\phi}^2 + mL \cos(\phi)\ddot{\phi} = u \quad (16)$$

$$m\ddot{x} \cos(\phi) + mL\ddot{\phi} = mg \sin(\phi). \quad (17)$$

- jedino čime upravljamo je sila \vec{u} ; kakav je njezin utjecaj na ponašanje sustava?

Stabilizacija okrenutog njihala

Stanje ovog sustava čini uređena četvorka $(\phi, \dot{\phi}, x, \dot{x})$, tj. kut, kutna brzina, vodoravni pomak te vodoravna brzina. Želimo razviti upravljački sustav.

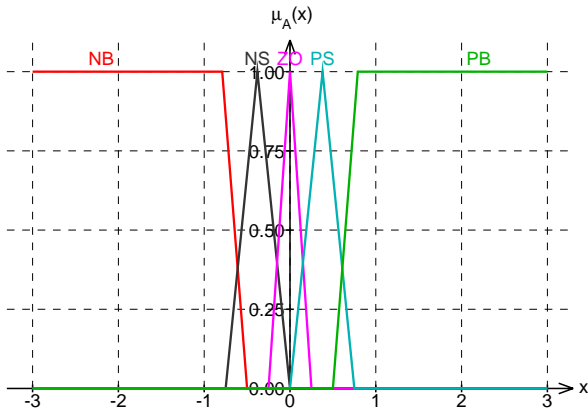


Stabilizacija okrenutog njihala

Pokušat ćemo razviti upravljački sustav koji razmatra samo dvije varijable stanja: kut (u radijanima) te kutnu brzinu (rad/s).

- 1 Definiramo ulaznu jezičnu varijablu KUT čiji je osnovni skup jezičnih termina {NB, NS, ZO, PS, PB}.
- 2 Definiramo ulaznu jezičnu varijablu KUTNA BRZINA čiji je osnovni skup jezičnih termina {NB, NS, ZO, PS, PB}.
- 3 Definiramo izlaznu jezičnu varijablu SILA čiji je osnovni skup jezičnih termina {NB, NS, ZO, PS, PB}.

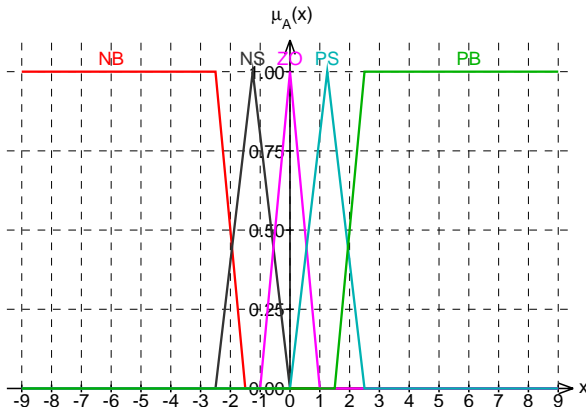
Jezična varijabla KUT



Jezična varijabla KUT

$$\begin{aligned}\text{NB} \quad \mu_{NB}(x) &= L(x; -0.79, -0.50) \\ \text{NS} \quad \mu_{NS}(x) &= \Lambda(x; -0.75, -0.38, 0.00) \\ \text{ZO} \quad \mu_{ZO}(x) &= \Lambda(x; -0.25, 0.00, 0.25) \\ \text{PS} \quad \mu_{PS}(x) &= \Lambda(x; 0.00, 0.38, 0.75) \\ \text{PB} \quad \mu_{PB}(x) &= \Gamma(x; 0.50, 0.79)\end{aligned}$$

Jezična varijabla KUTNA BRZINA



Jezična varijabla KUTNA BRZINA

$$\text{NB} \quad \mu_{\text{NB}}(x) = L(x; -2.5, -1.5)$$

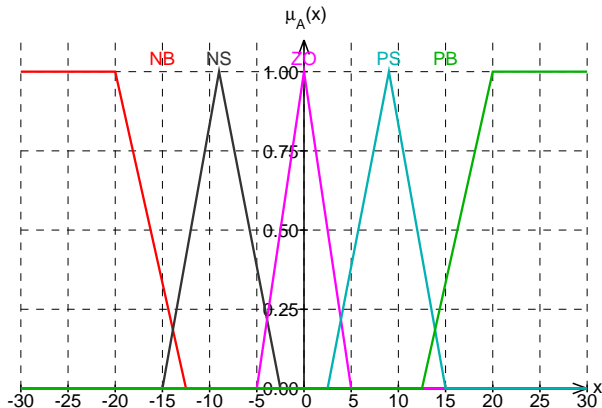
$$\text{NS} \quad \mu_{\text{NS}}(x) = \Lambda(x; -2.5, -1.25, 0.0)$$

$$\text{ZO} \quad \mu_{\text{ZO}}(x) = \Lambda(x; -1.0, 0.0, 1.0)$$

$$\text{PS} \quad \mu_{\text{PS}}(x) = \Lambda(x; 0.0, 1.25, 2.5)$$

$$\text{PB} \quad \mu_{\text{PB}}(x) = \Gamma(x; 1.5, 2.5)$$

Jezična varijabla SILA



Jezična varijabla SILA

$$\text{NB} \quad \mu_{\text{NB}}(x) = L(x; -20.0, -12.5)$$

$$\text{NS} \quad \mu_{\text{NS}}(x) = \Lambda(x; -15.0, -9, -2.5)$$

$$\text{ZO} \quad \mu_{\text{ZO}}(x) = \Lambda(x; -5.0, 0.0, 5.0)$$

$$\text{PS} \quad \mu_{\text{PS}}(x) = \Lambda(x; 2.5, 9, 15.0)$$

$$\text{PB} \quad \mu_{\text{PB}}(x) = \Gamma(x; 12.5, 20.0)$$

Pravila

Sustav neizrazitog upravljanja koristit će pravila oblika:

AKO *kutna brzina* je A_{i1} **I** *kut* je A_{i2} **TADA** *sila* je B_i

- neizraziti skupovi A_{i1} definirani su nad univerzalnim skupom $[-9, +9]$
- neizraziti skupovi A_{i2} definirani su nad univerzalnim skupom $[-3, +3]$
- neizraziti skupovi B_i definirani su nad univerzalnim skupom $[-35, +35]$
- ulazne vrijednosti pretvarat će se u jednoelementne neizrazite skupove (singletone)
- stroj za zaključivanje koji ćemo koristiti je stroj koji se temelji na produktu.

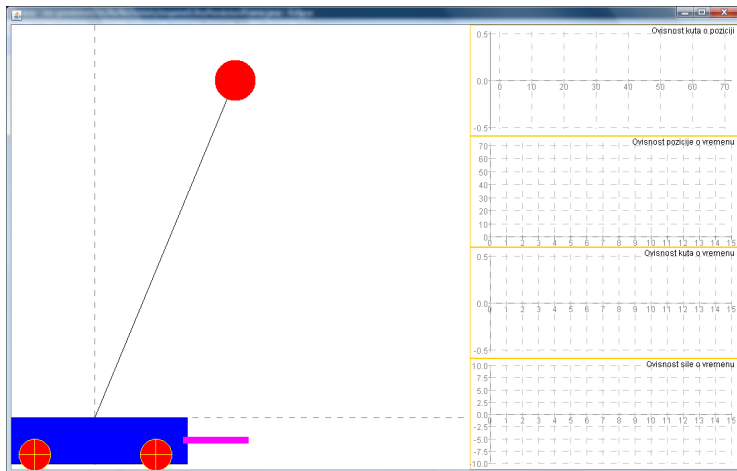
Pravila

Pravila ćemo prikazati u matričnom obliku:

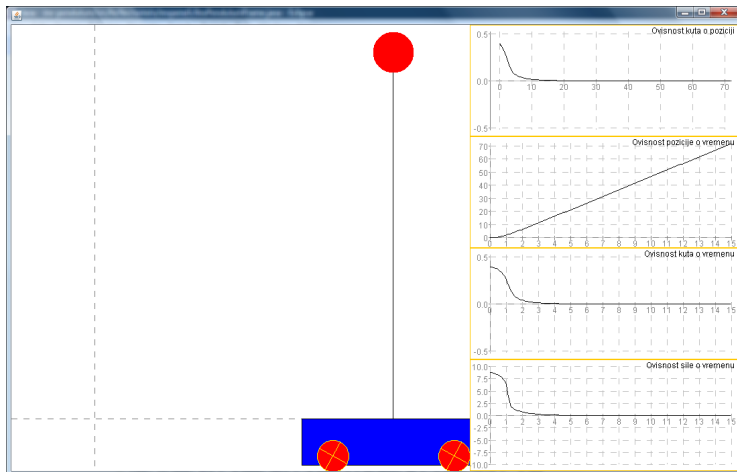
		Kut				
		NB	NS	ZO	PS	PB
Kutna brzina	NB	NB	NB	NB	NS	ZO
	NS	NB	NB	NS	ZO	PS
	ZO	NB	NS	ZO	PS	PB
	PS	NS	ZO	PS	PB	PB
	PB	ZO	PS	PB	PB	PB

Dekodiranje neizrazitosti provodit ćemo metodom *centra površine*.

Simulacija: početno stanje



Simulacija: nakon 15 sekundi



Gradivo

Proučiti u knjizi cjelokupno poglavlje 5.

- Podpoglavlje 5.5.2 dano je samo informativno za one koji žele znati više.