Podstawy sztucznej inteligencji AI

część 3

Logika rozmyta

Wnioskowanie przybliżone

Przyjmijmy, że zdania występujące w regułach modus ponens oraz modus tollens zawierają nieprecyzyjne określenia.

Oznacza to, że zdaniom tym odpowiadają pewne zbiory rozmyte.

Rozmyta reguła *modus ponens*

x jest A′	przesłanka
JEŻELI x jest A TO y jest B	implikacja
y jest B′	wniosek

gdzie: $A, A' \subseteq X$ i $B, B' \subseteq Y$ są zbiorami rozmytymi

x, y są zmiennymi lingwistycznymi

Przykład

Prędkość samochodu jest duża — przesłanka

Jeżeli prędkość samochodu jest bardzo duża
to poziom hałasu jest wysoki — implikacja

Poziom hałasu w samochodzie jest średniowysoki — wniosek

Zmienne lingwistyczne

x – prędkość samochodu

Zbiór wartości: T_x={mała, średnia, duża, bardzo duża}

y – poziom hałasu

Zbiór wartości: T_y={mały, średni, średniowysoki, wysoki}

Przykład (cd)

Zbiory rozmyte:

A = "bardzo duża prędkość samochodu"

A′ = "duża prędkość samochodu"

B = "wysoki poziom hałasu"

B′ = "średniowysoki poziom hałasu"

Jaka jest różnica między nierozmytą i rozmytą regułą *modus ponens* ?

A	x jest A′	przesłanka
A → B	JEŻELI x jest A TO y jest B	implikacja
B	y jest B′	wniosek

To samo zdanie

To samo zdanie

Zbiory inne, „zbliżone”

Zbiory inne, „zbliżone”

Zbiór rozmyty B′ jest złożeniem zbioru rozmytego A′ i rozmytej implikacji A → B, która jest równoważna pewnej relacji rozmytej R⊆X×Y czyli

$$B' = A' \circ (A \to B)$$

Zatem

$$\mu_{B'}(y) = \sup_{x \in X}\{\mu_{A'}(x) \overset{T}{*} \mu_{A \to B}(x,y)\}$$

gdzie $\mu_{A \to B}(x,y)$ jest funkcją przynależności relacji R.

Jeżeli T(a,b)=min{a,b} wówczas

$$\mu_{B'}(y) = \sup_{x \in X}\{\min\{\mu_{A'}(x), \mu_{A \to B}(x,y)\}\}$$

Zauważmy, że jeżeli A=A′ i B=B′ wówczas rozmyta reguła *modus ponens* redukuje się do zwykłej reguły *modus ponens*.

Jak modelować implikację?

Model Mamdaniego

$$\mu_{A\to B}(x,y) = T(\mu_A(x), \mu_B(y))$$

gdzie T jest dowolną *t*-normą.

Przykład

Reguła typu minimum

$$\mu_{A\to B}(x,y) = \min\{\mu_A(x), \mu_B(y)\}$$

Reguła typu iloczyn (Larsena)

$$\mu_{A\to B}(x,y) = \mu_A(x) \cdot \mu_B(y)$$

Zauważmy, że

$\mu_A(x)$	$\mu_B(y)$	$min\{\mu_A(x), \mu_B(y)\}$	$\mu_A(x)\mu_B(y)$
0	0	0	0
0	1	0	0
1	0	0	0
1	1	1	1

Wniosek

Reguły typu Mamdaniego nie są implikacjami w sensie logicznym.

Model logiczny

Definicja

Funkcję I: $[0,1]^2 \rightarrow [0,1]$ spełniającą następujące warunki:

1) $a_1 \leq a_3 \;\Rightarrow\; I(a_1, a_2) \geq I(a_3, a_2) \;\; dla \;\; a_1, a_2, a_3 \in [0,1]$
2) $a_2 \leq a_3 \Rightarrow I(a_1, a_2) \leq I(a_1, a_3)$ *dla* $a_1, a_2, a_3 \in [0,1]$
3) $I(0, a_2) = 1 \;\; dla \;\; a_2 \in [0,1]$
4) $I(a_1, 1) = 1 \;\; dla \;\; a_1 \in [0,1]$
5) $I(1,0) = 0$

nazywamy implikacją rozmytą.

Przykład

Implikacja binarna $I(a,b)=\max\{1-a,\ b\}$

Implikacja Łukasiewicza $I(a,b)=\min\{1,1-a+b\}$

Implikacja Reichenbacha $I(a,b)=1-a+a\cdot b$

Implikacja Gödela $I(a,b)=\begin{cases}1 & dla \quad a\leq b\\ b & dla \quad a>b\end{cases}$

Funkcja I pozwala nam definiować funkcję przynależności dla implikacji

$$\mu_{A\to B}(x,y)=I(\mu_A(x),\mu_B(y))$$

(implikacja jest relacją rozmytą $R\subseteq X\times Y$)

Implikacja binarna

$\mu_A(x)$	$\mu_B(y)$	$1-\mu_A(x)$	$max\{1- \mu_A(x),\mu_B(y)\}$
0	0	1	1
0	1	1	1
1	0	0	0
1	1	0	1

Implikacja Gödela

$\mu_A(x)$	$\mu_B(y)$	
0	0	1
0	1	1
1	0	0
1	1	1

Rozmyty system wnioskujący
PRZYKŁAD

Problem

Chcemy zbudować przykładowy sterownik rozmyty, który dla otrzymanej na wejściu odległości od przeszkody (odleglosc) wyznaczy nam prędkość (predkosc) pojazdu.

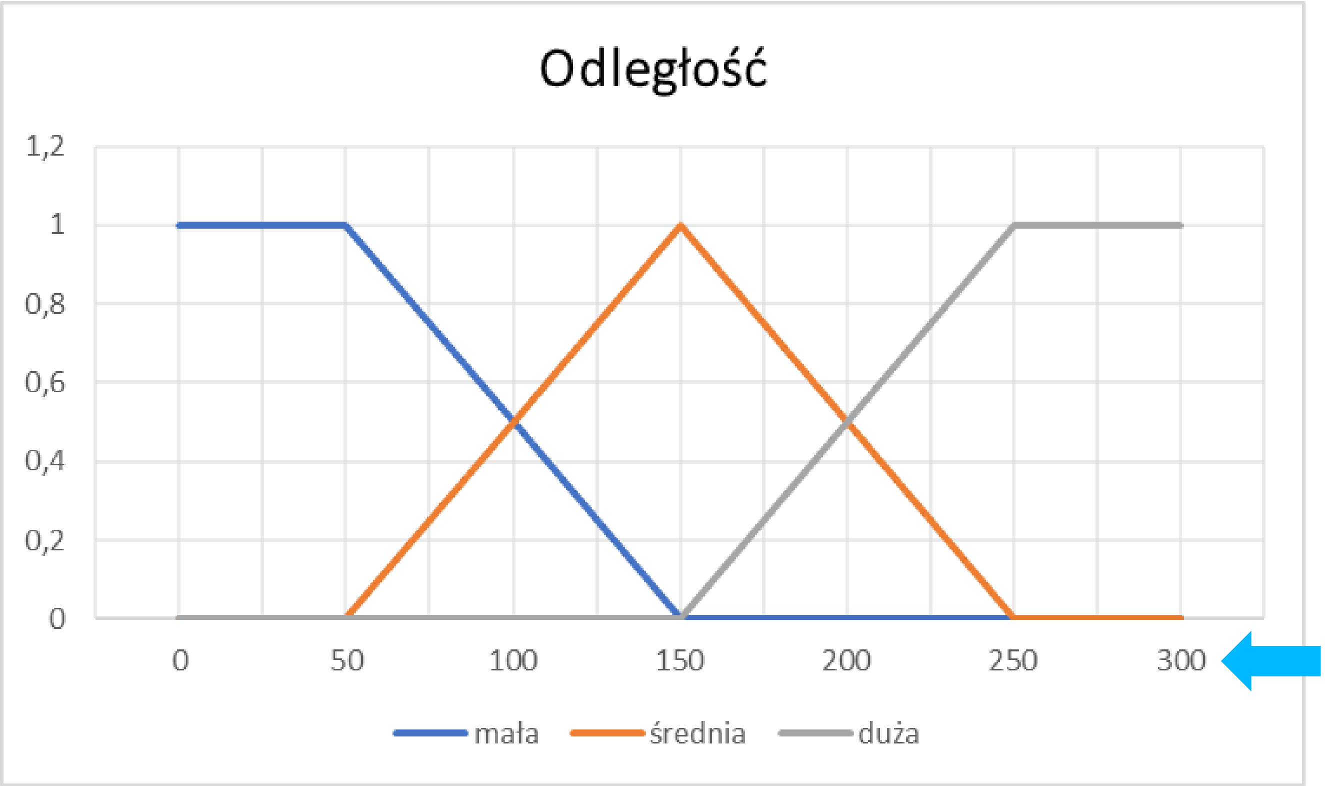
Wykorzystamy dwie zmienne lingwistyczne:

Odległość

Zmienna lingwistyczne:

odległość (d) → przestrzeń rozważań: $D = [0,300]$

Możliwe wartości:



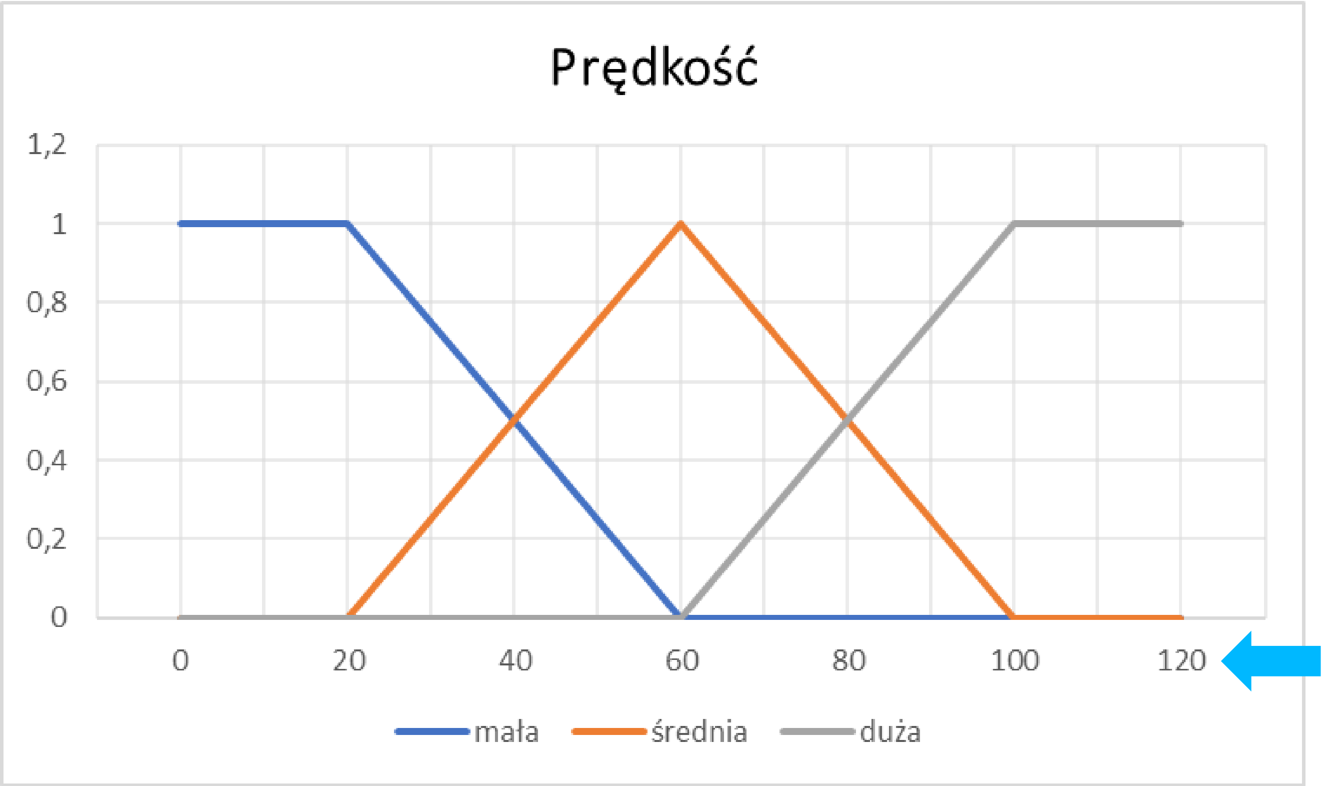
!

Prędkość

Zmienna lingwistyczne:

prędkość (v) → przestrzeń rozważań: $V = [0,120]$

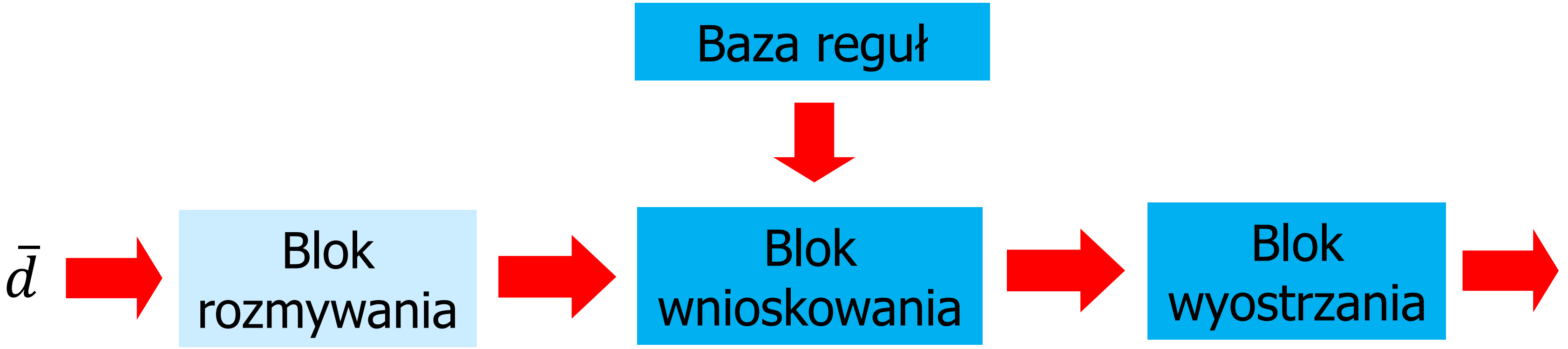
Możliwe wartości:



!

Rozmyty system wnioskujący

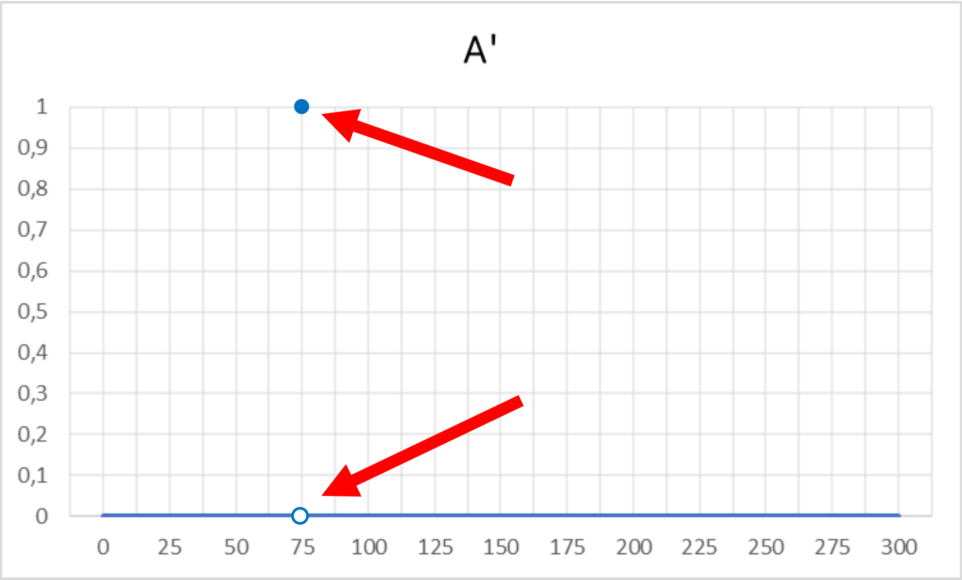
Przyjmijmy, że na wejście sterownika podana jest pewna wartość odległości $\bar{d} = 75m$



Blok rozmywania

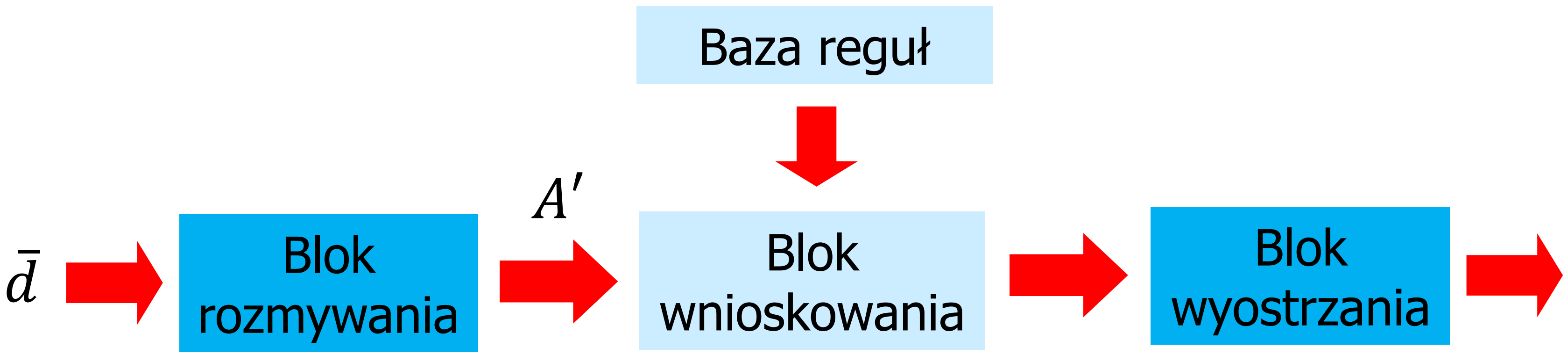
Blok rozmywania wartości otrzymanej na wejściu $\bar{d} = 75m$ przyporządkowuje zbiór rozmyty **A’** zdefiniowany w przestrzeni $\mathrm{D} = [0{,}300]$ o następującej funkcji przynależności:

$$\mu_{A'}(d) = \delta(d - \bar{d}) = \begin{cases} 1, & d = \bar{d} \\ 0, & d \neq \bar{d} \end{cases}$$



Rozmyty system wnioskujący

Przyjmijmy, że na wejście sterownika podana jest pewna wartość odległości $\bar{d} = 75m$



Baza reguł

Rozważymy następujące reguły:

R_1 IF odległość IS **mała** THEN prędkość IS **mała**

R_2 IF odległość IS **średnia** THEN prędkość IS **średnia**

R_3 IF odległość IS **duża** THEN prędkość IS **duża**

Reguły

Każda reguła ma postać implikacji:

R_k IF odległość IS ….. THEN prędkość IS …..

pojęcie rozmyte pojęcie rozmyte

Implikację możemy zapisać symbolicznie jako:

$$R_k: A_k \rightarrow B_k$$

gdzie $A_k \subseteq D = [0,300]$ i $B_k \subseteq V = [0,120]$ są zbiorami rozmytymi zdefiniowanymi wcześniej.

Implikacja

Implikację: **R$_k$**: $A_k \rightarrow B_k$

traktujemy jako **relację rozmytą** $R_k \subseteq D \times V$.

W modelu Mamdaniego funkcja przynależności dla tej relacji jest zdefiniowana następująco:

$$\mu_{A_k \rightarrow B_k}(d, v) = T(\mu_A(d), \mu_B(v))$$

gdzie T jest operatorem t-normy:

$$T(a, b) = \min\{a, b\}$$

$$T(a, b) = a \cdot b$$

Blok wnioskowania

Wyjściami bloku wnioskowania są zbiory rozmyte $\bar{B}_k$ będące **złożeniami** zbioru $\boldsymbol{A}'$ (wyjście bloku rozmywania) **z każdą regułą** $\boldsymbol{R_k}$ traktowaną jako relacja rozmyta:

$$\mu_{\bar{B}_k}(v) = \sup_{d \in D}\{T(\mu_{A'}(d), \mu_{R_k}(d, v))\}$$

gdzie T jest operatorem t-normy.

W naszym przykładzie przyjmijmy, że:

$$T(a, b) = \min\{a, b\}$$

W ten sam sposób zdefiniujemy t-normę w implikacji.

Blok wnioskowania

Otrzymujemy zatem:

$$\mu_{\bar{B}_k}(v) = \sup_{d \in D}\{T(\mu_{A'}(d), \mu_{R_k}(d, v))\} =$$

$$= \sup_{d \in D}\{\min\{\mu_{A'}(d), \mu_{R_k}(d, v)\}\} =$$

Korzystamy teraz z definicji zbioru A': $\mu_{A'}(d) = \begin{cases} 1, & d = \bar{d} \\ 0, & d \neq \bar{d} \end{cases}$

$$= \min\{1, \mu_{R_k}(\bar{d}, v)\} =$$

Korzystamy z definicji implikacji w modelu Mamdaniego:

$$= \min\{1, \min\{\mu_{A_k}(\bar{d}), \mu_{B_k}(v)\}\} = \min\{\mu_{A_k}(\bar{d}), \mu_{B_k}(v)\}$$

Blok wnioskowania

Czyli zbiór na wyjściu bloku wnioskowania ma następującą funkcję przynależności:

$$\mu_{\bar{B}_k}(v) = \min\{\mu_{A_k}(\bar{d}), \mu_{B_k}(v)\}$$

gdzie $k = 1,2,3$ bo tyle mamy reguł w bazie.

Spróbujmy skorzystać z tej formuły i znaleźć zbiory $\bar{B}_k$.

W tym celu wykorzystamy funkcje przynależności do zbiorów reprezentujących określenia rozmyte z reguł:

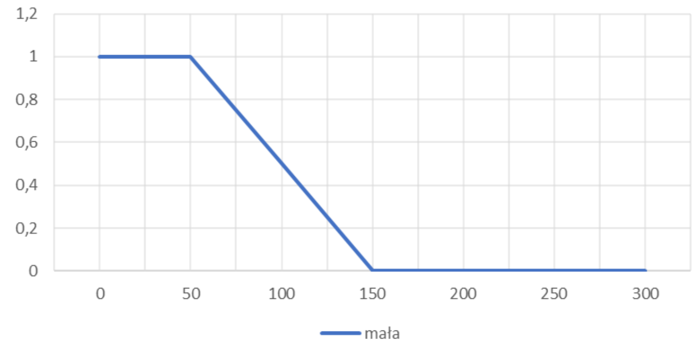
$$\mathbf{R_k}: A_k \rightarrow B_k$$

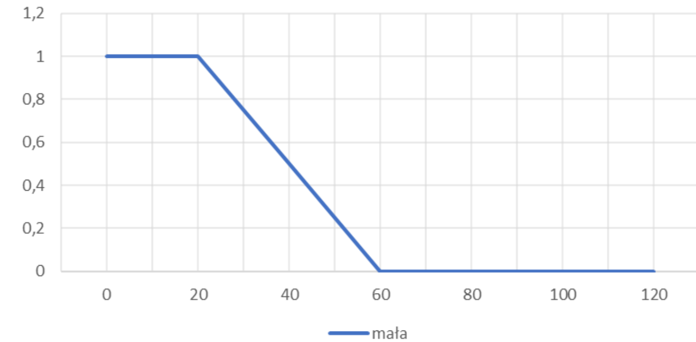
Blok wnioskowania

$$\mu_{\bar{B}_k}(v) = \min\{\mu_{A_k}(\bar{d}), \mu_{B_k}(v)\}$$

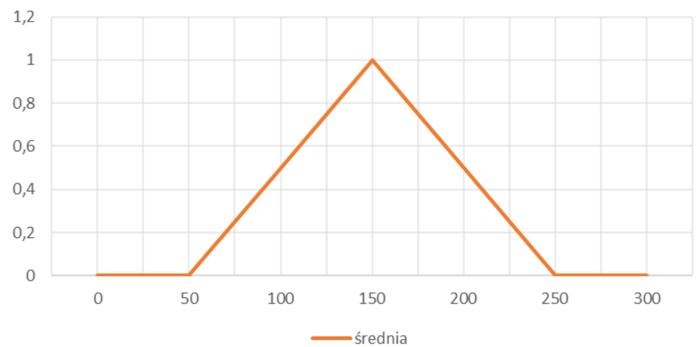
A_k B_k

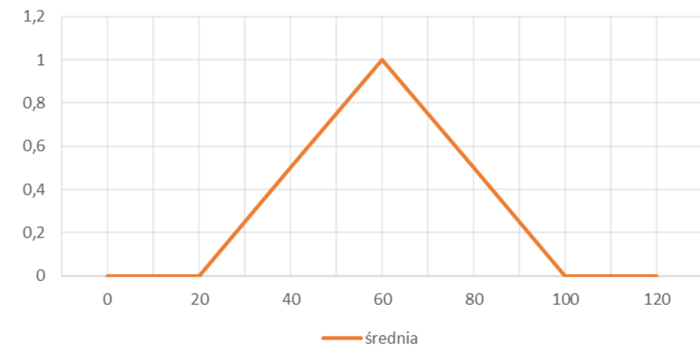
R_1



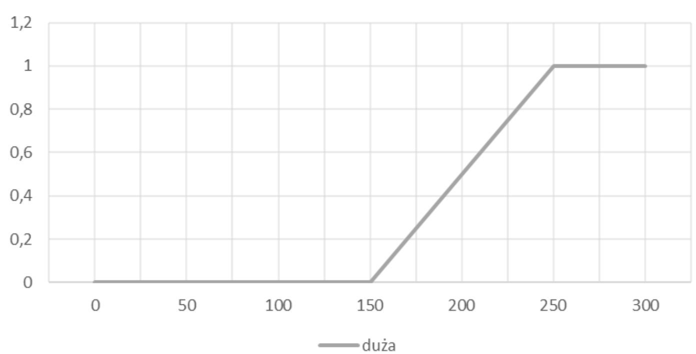


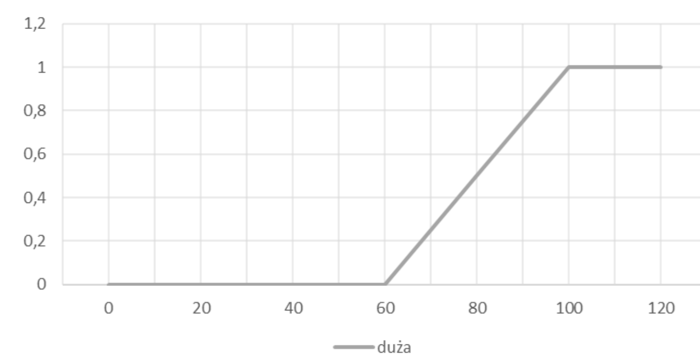
R_2





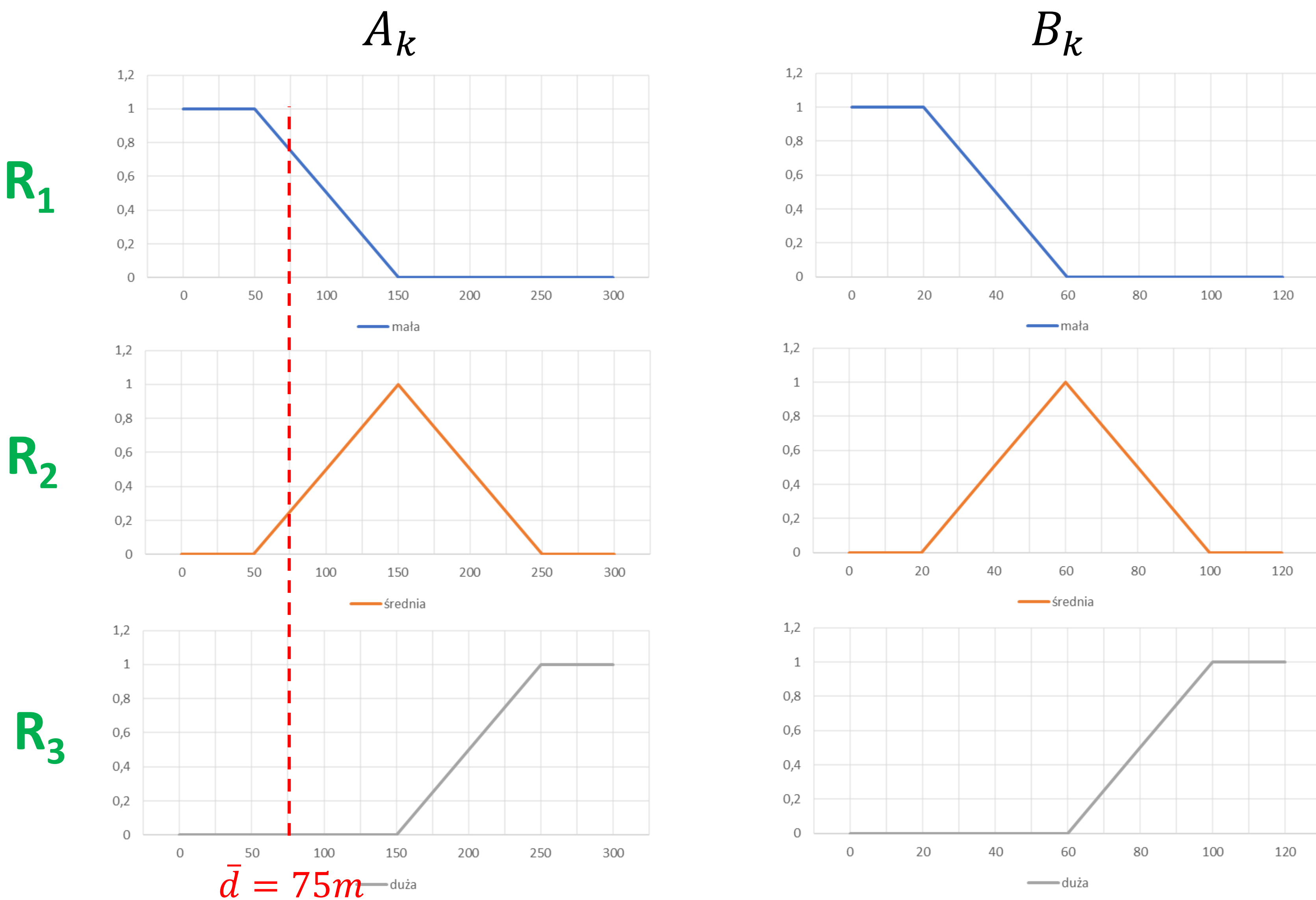
R_3





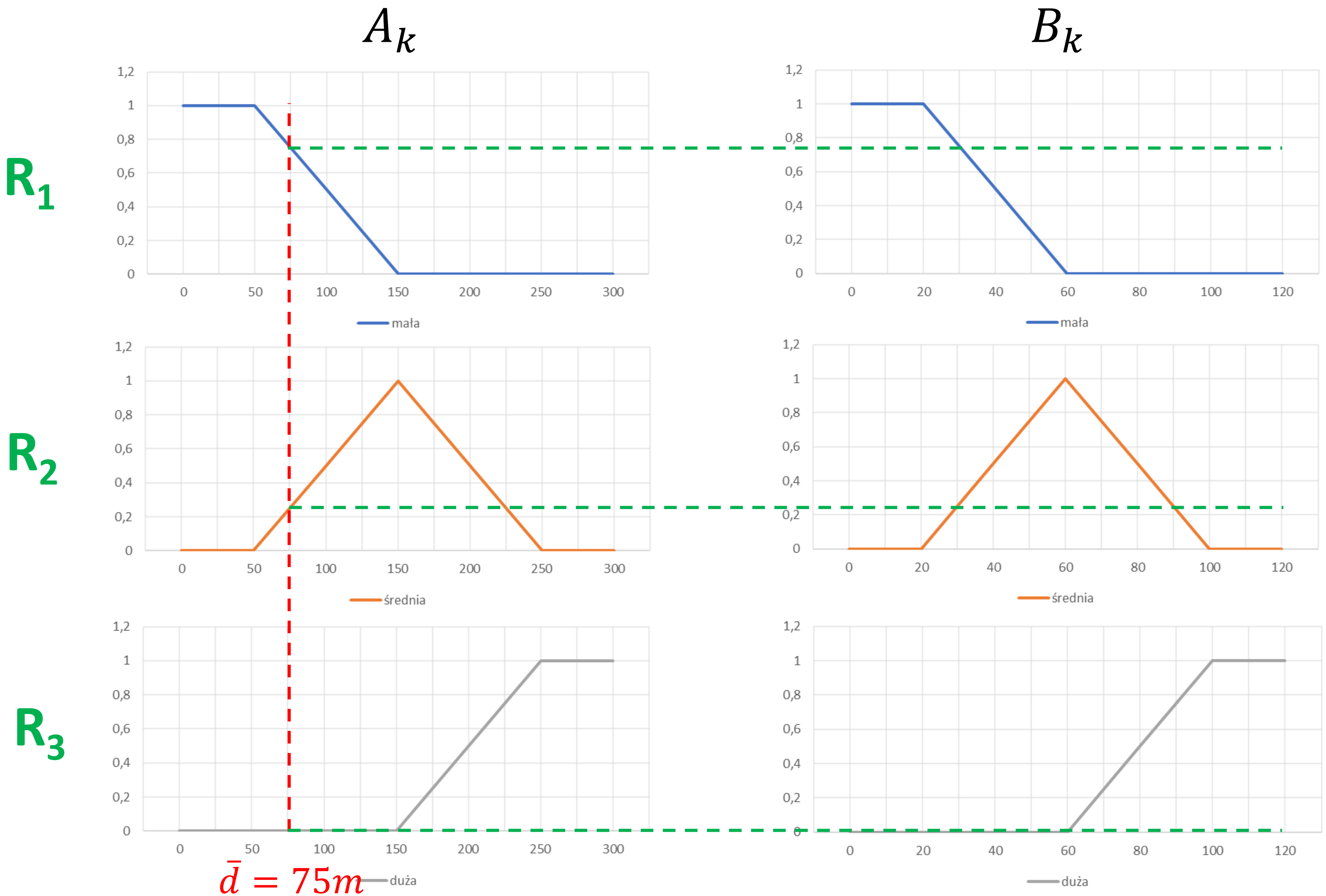
Blok wnioskowania

$$\mu_{\bar{B}_k}(v) = \min\{\mu_{A_k}(\bar{d}), \mu_{B_k}(v)\}$$



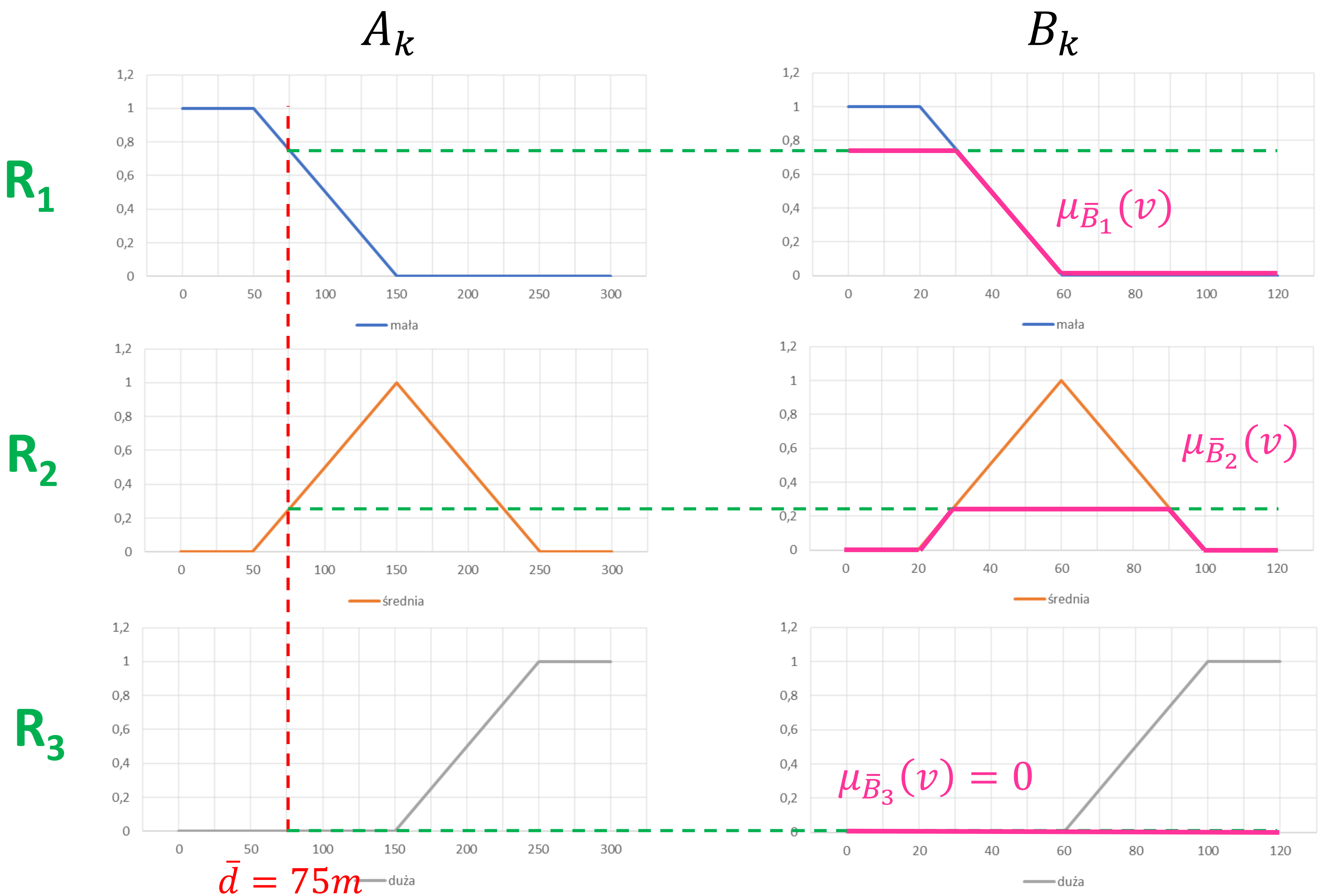
Blok wnioskowania

$$\mu_{\bar{B}_k}(v) = \min\{\mu_{A_k}(\bar{d}), \mu_{B_k}(v)\}$$



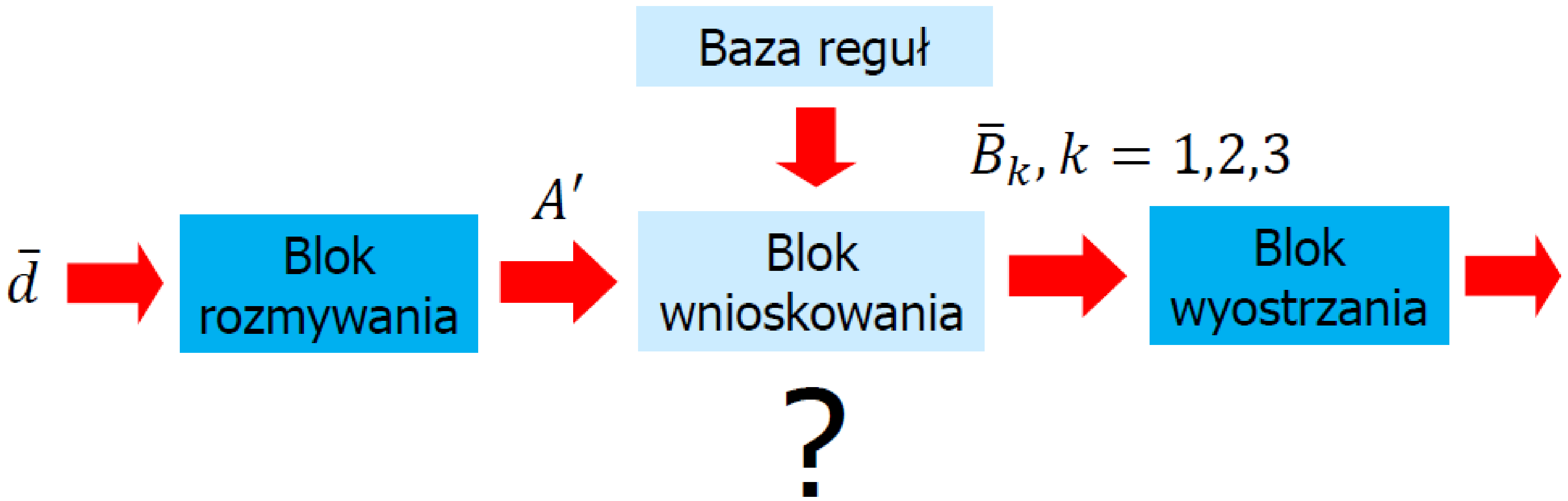
Blok wnioskowania

$$\mu_{\bar{B}_k}(v) = \min\{\boldsymbol{\mu_{A_k}(\bar{d})}, \mu_{B_k}(v)\}$$



Rozmyty system wnioskujący

Przyjmijmy, że na wejście sterownika podana jest pewna wartość odległości $\bar{d} = 75m$

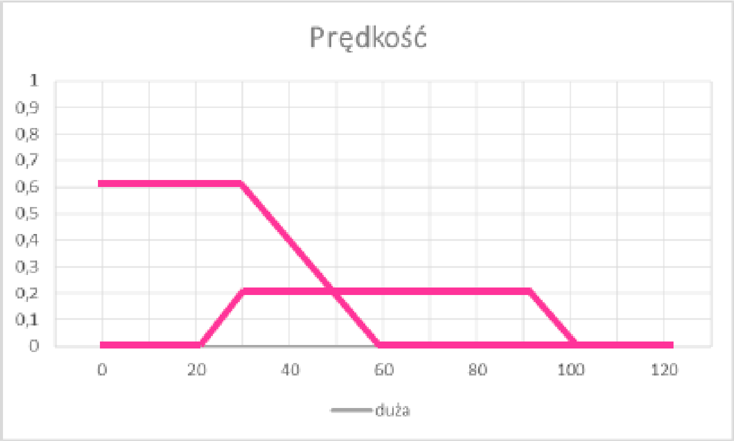


Blok wnioskowania

Zbiory $\bar{B}_k$ sumujemy i wynikowy zbiór $\bar{B}$ jest wejście bloku wyostrzania:

$$\bar{B} = \bigcup_{k=1}^{3} \bar{B}_k = \bar{B}_1 \cup \bar{B}_2 \cup \bar{B}_3$$

W naszym przykładzie otrzymamy:

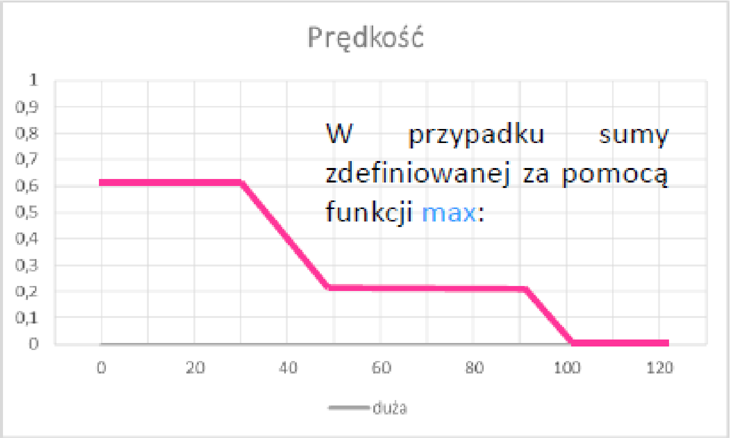


Blok wnioskowania

Zbiory $\bar{B}_k$ sumujemy i wynikowy zbiór $\bar{B}$ jest wejście bloku wyostrzania:

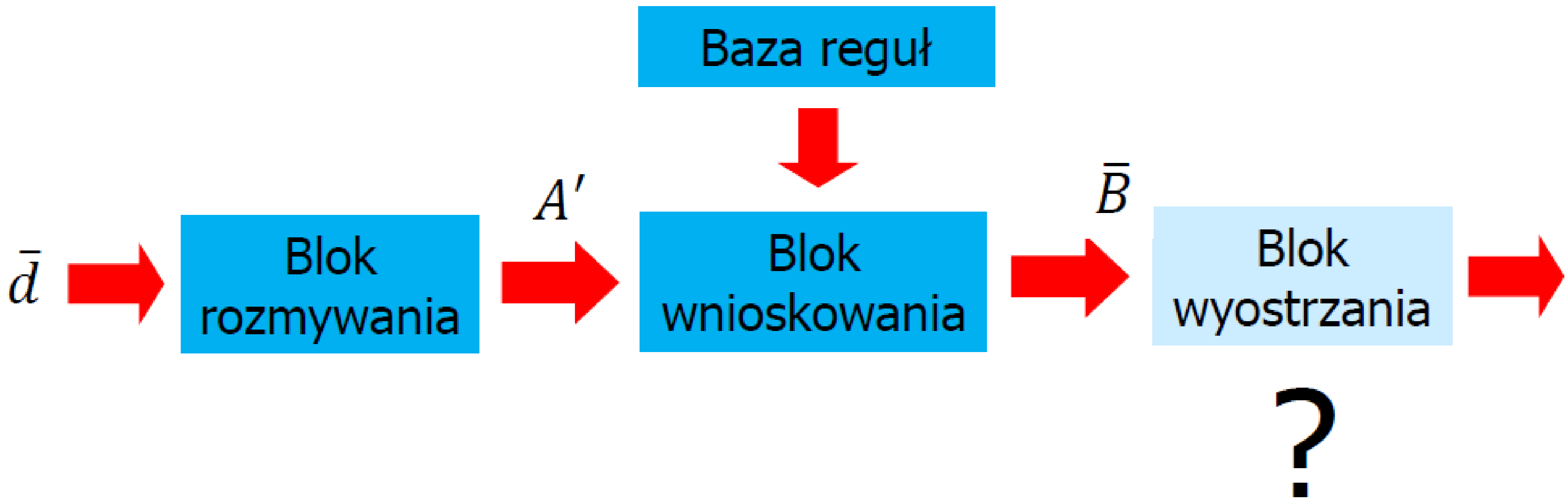
$$\bar{B} = \bigcup_{k=1}^{3} \bar{B}_k = \bar{B}_1 \cup \bar{B}_2 \cup \bar{B}_3$$

W naszym przykładzie otrzymamy:

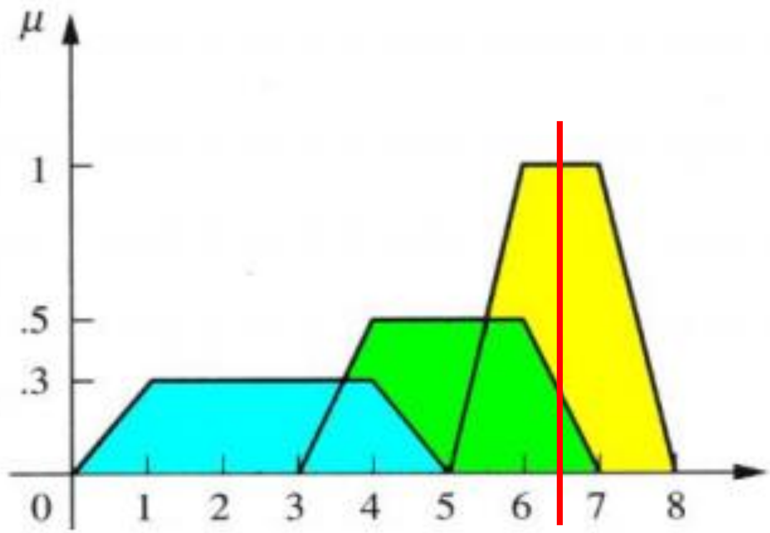


Rozmyty system wnioskujący

Przyjmijmy, że na wejście sterownika podana jest pewna wartość odległości $\bar{d} = 75m$



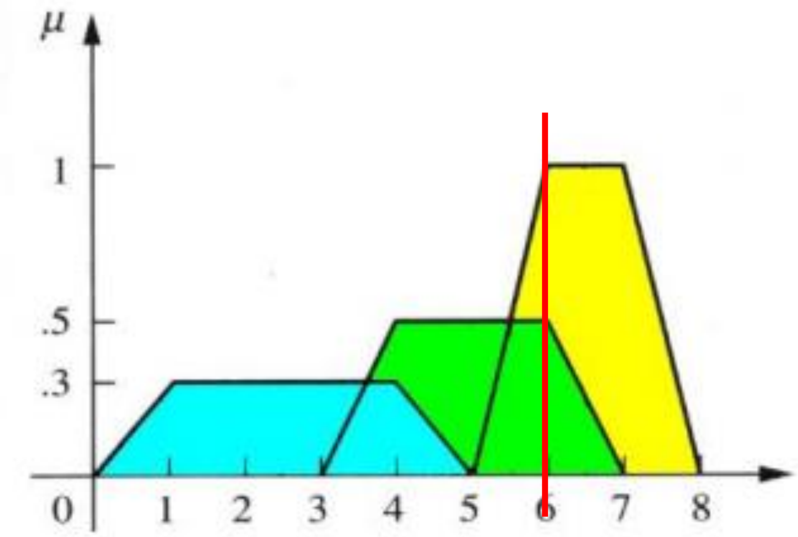
Metody wyostrzania



Metoda Środka Maksimum (SM) (Middle of Max (MOM))

Ostry reprezentant x^* wynikowego zbioru rozmytego konkluzji B' zdefiniowanego funkcją przynależności μ przyjmuje współrzędną x będącą wartością średnią wyjść dla których funkcja przynależności μ osiąga maksimum.

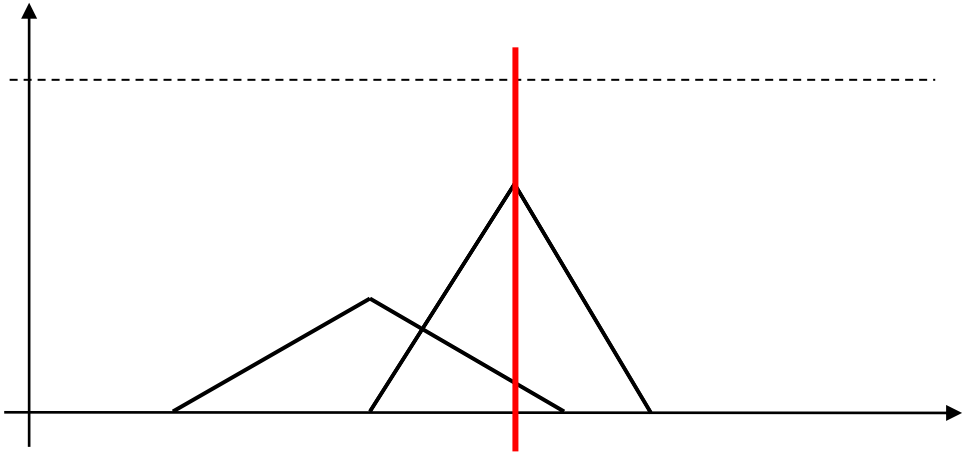
Metody wyostrzania



Metoda Pierwszego Maksimum (PM) (Smallest of Max (SOM))

Ostry reprezentant x^* wynikowego zbioru rozmytego konkluzji B' zdefiniowanego funkcją przynależności μ przyjmuje współrzędną x będącą najmniejszą wartością x dla której funkcja przynależności μ osiąga maksimum.

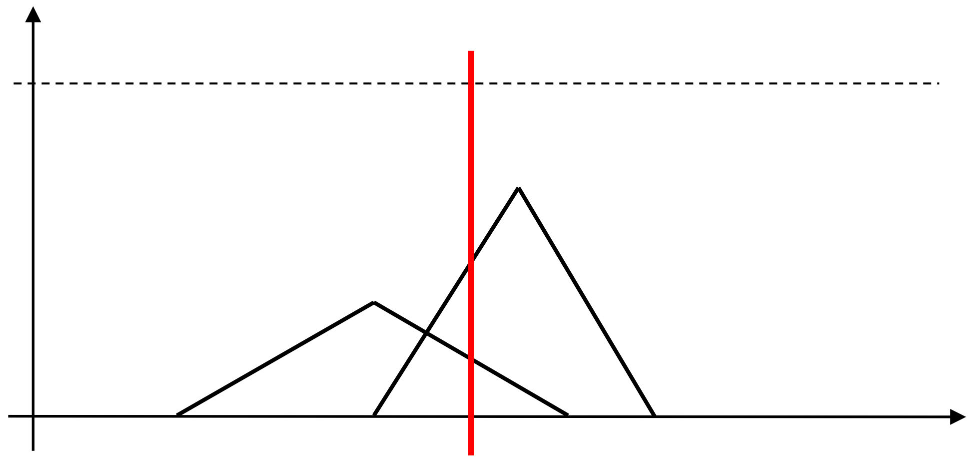
Metody wyostrzania



Metoda Maksymalnej Przynależności (MP) (Max Membership (MM))

Ostry reprezentant x^* wynikowego zbioru rozmytego konkluzji B' zdefiniowanego funkcją przynależności μ przyjmuje współrzędną x będącą <u>wartością x dla której funkcja przynależności μ osiąga maksimum</u>.

Metody wyostrzania



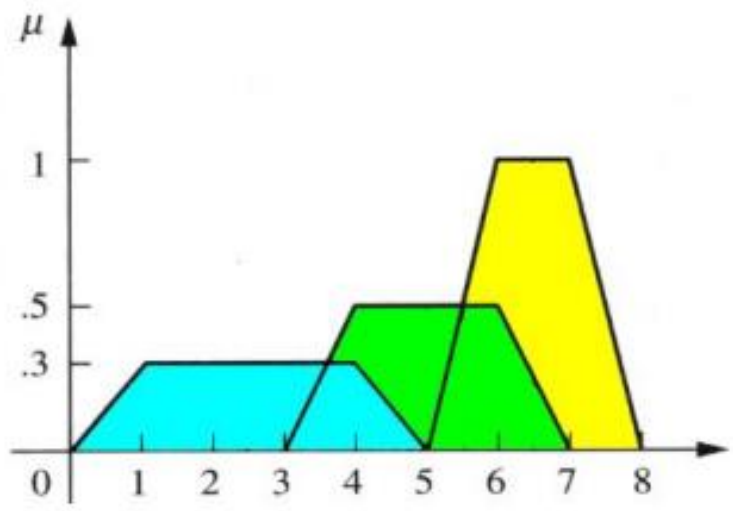
Metoda Średniej Ważonej (ŚW) (Weighted Average (WA))

Ostry reprezentant *x** wynikowego zbioru rozmytego B' wyliczany jest ze wzoru:

$$x^* = \frac{\sum_{k=1}^{N} \mu_{\bar{B}_k}(\bar{x}^k)\bar{x}^k}{\sum_{k=1}^{N} \mu_{\bar{B}_k}(\bar{x}^k)}$$

gdzie $\bar{x}^k$ to punkt w którym funkcja przynależności $\mu_{\bar{B}_k}$ osiąga maksimum.

Metody wyostrzania



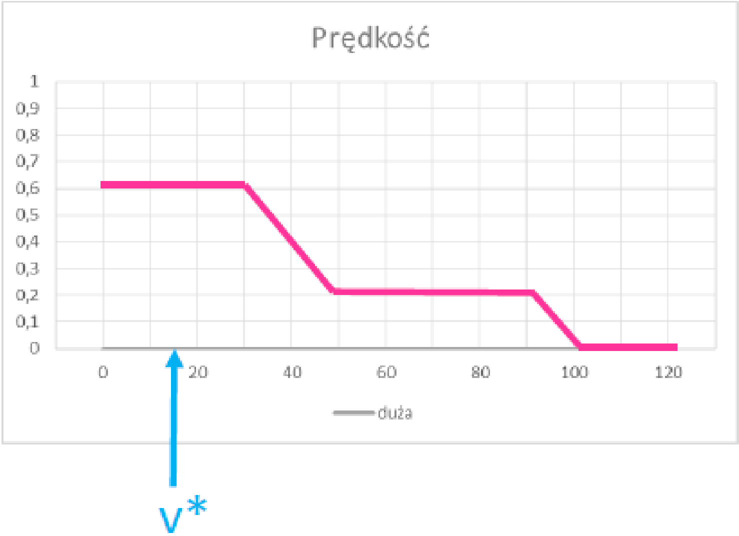
Metoda Środka Ciężkości (SC) (Center of Gravity (COG), Center of Area (COA))

Ostry reprezentant x^* wynikowego zbioru rozmytego B' wyliczany jest ze wzoru:

$$x^* = \frac{\int \mu(x) x dx}{\int \mu(x) dx}$$

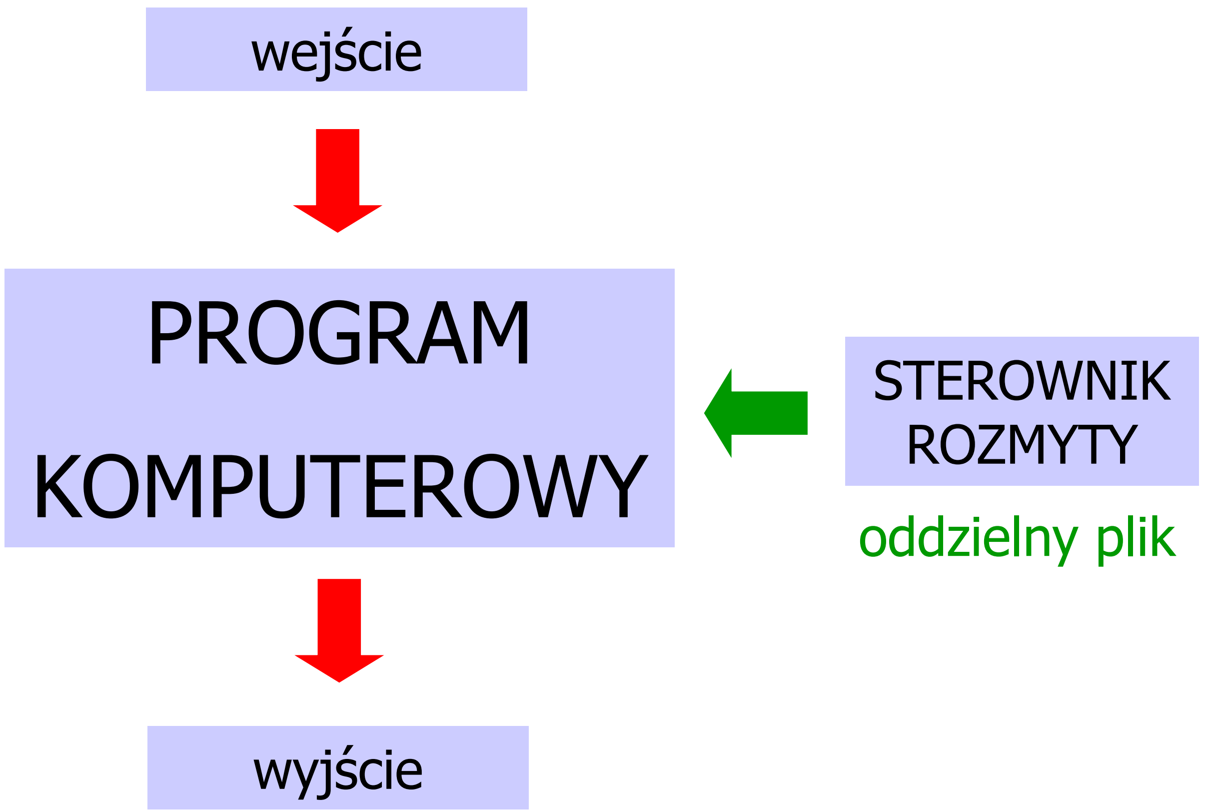
Blok wyostrzania

W naszym przykładzie zbiór $\bar{B}$ ma następującą funkcję przynależności:



v*=15 jest wynikiem uzyskanym po zastosowaniu metody środka maksimum.

Sterowniki rozmyte i programowanie



Fuzzy Control Language

Fuzzy Control Language (FCL) to język pozwalający budować (definiować) sterowniki rozmyte.

Definicja sterownika rozmytego zapisana jest w pliku tekstowym z rozszerzeniem fcl.

Plik fcl zawiera instrukcje określające parametry sterownika. Instrukcje te zawarte są w następującym elemencie:

```
FUNCTION_BLOCK

//instrukcje

END_FUNCTION_BLOCK
```

Fuzzy Control Language

Chcemy zbudować przykładowy sterownik rozmyty, który dla otrzymanej na wejściu odległości od przeszkody (odleglosc) wyznaczy nam prędkość (predkosc) pojazdu.

Wykorzystamy dwie zmienne lingwistyczne:

odleglosc (wejście sterownika)

predkosc (wyjście steronika)

W języku FCL zapisujemy to następująco:

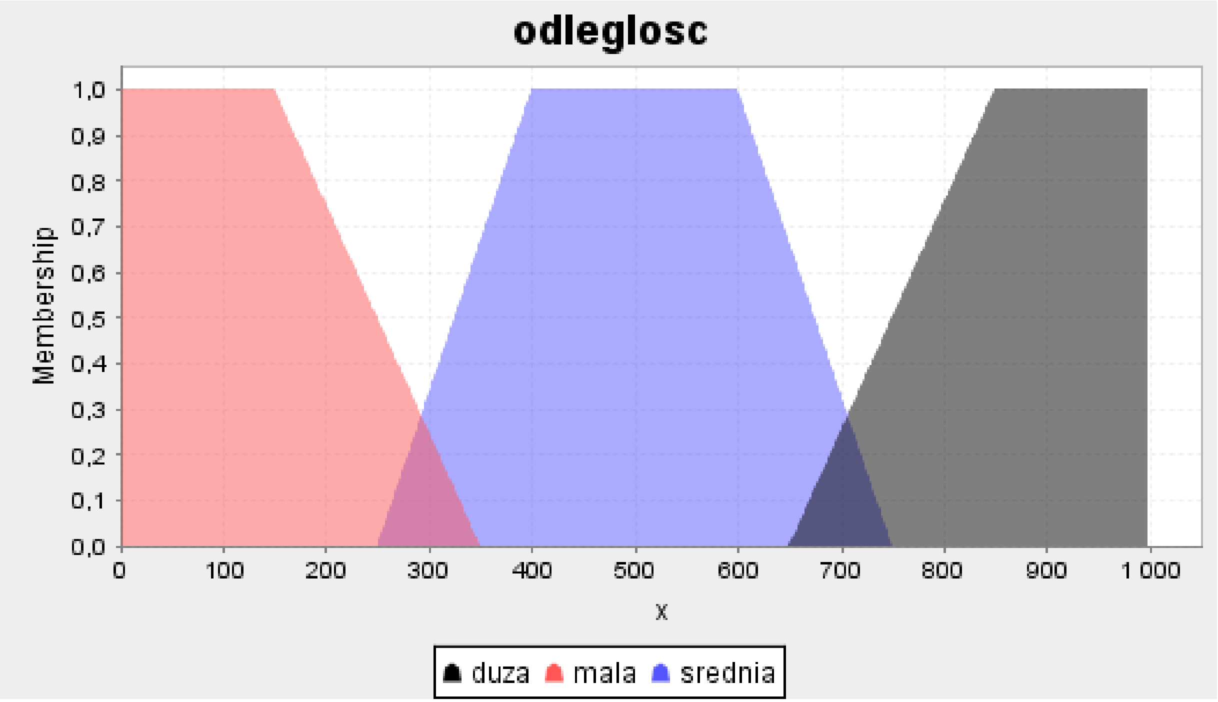
```
VAR_INPUT
odleglosc : REAL;
END_VAR
VAR_OUTPUT
predkosc : REAL;
END_VAR
```

FCL - wejście

Przyjmijmy, że interesuje nas odleglosc w przedziale [0,1000] (m).

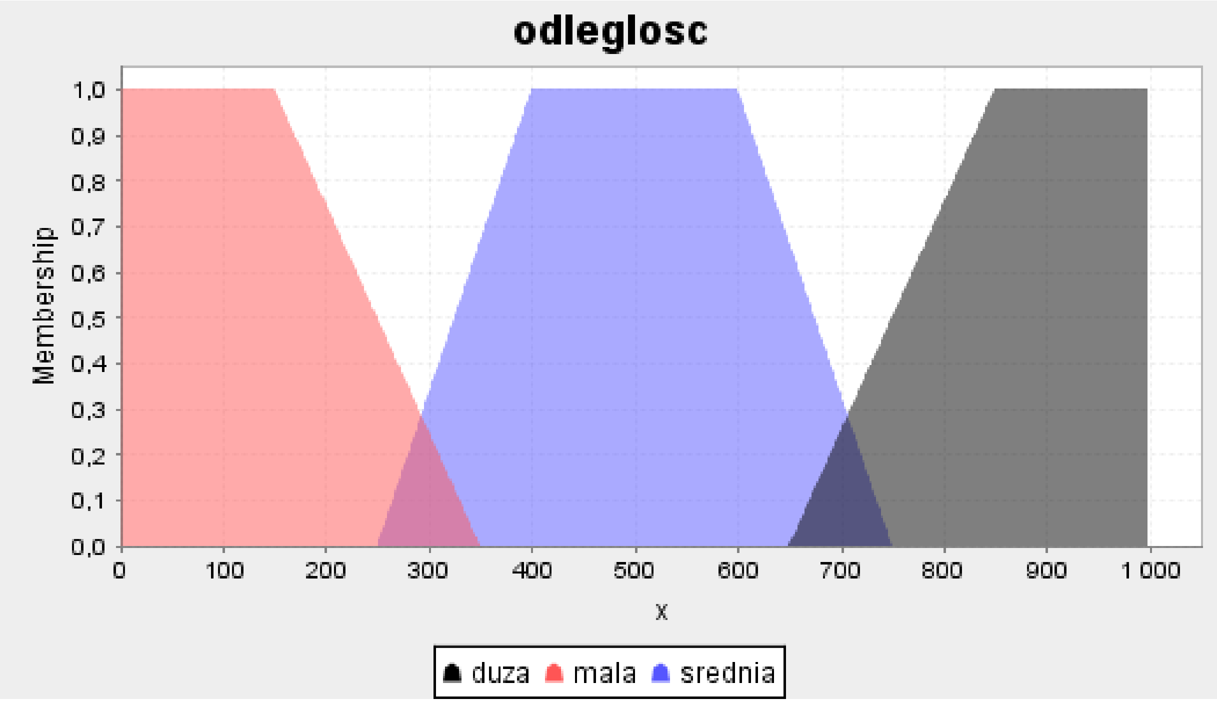
Konkretna wartość zmiennej odleglosc będzie podana na wejściu naszego sterownika. Wartość ta będzie następnie rozmyta.

Przyjmijmy, że zmienna odleglosc będzie przyjmowała następujące 3 wartości:



FCL - wejście

```
FUZZIFY odleglosc
TERM mala := (0, 1) (150, 1) (350, 0) ;
TERM srednia := (250, 0) (400,1) (600,1) (750,0);
TERM duza := (650, 0) (850, 1) (1000, 1);
END_FUZZIFY
```

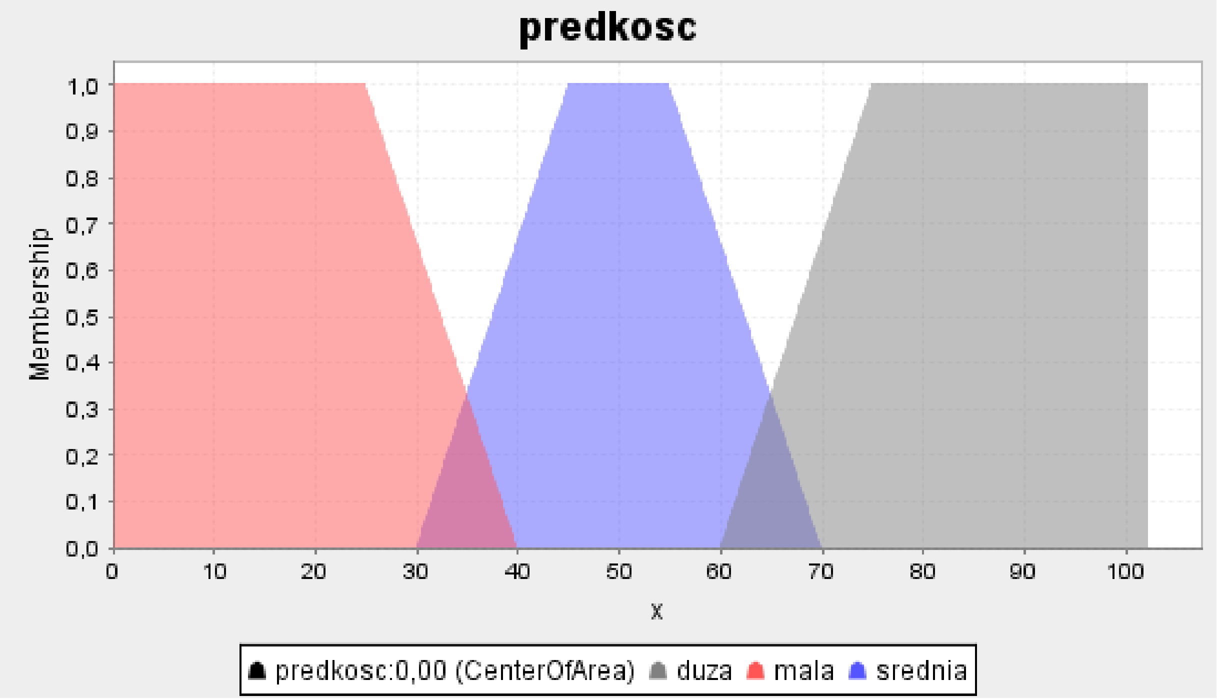


FCL - wyjście

Przyjmijmy, że interesuje nas predkosc w przedziale [0,100] (km/h).

Konkretna wartość zmiennej predkosc będzie zwrócona na wyjściu naszego sterownika. Wartość ta będzie efektem wyostrzania.

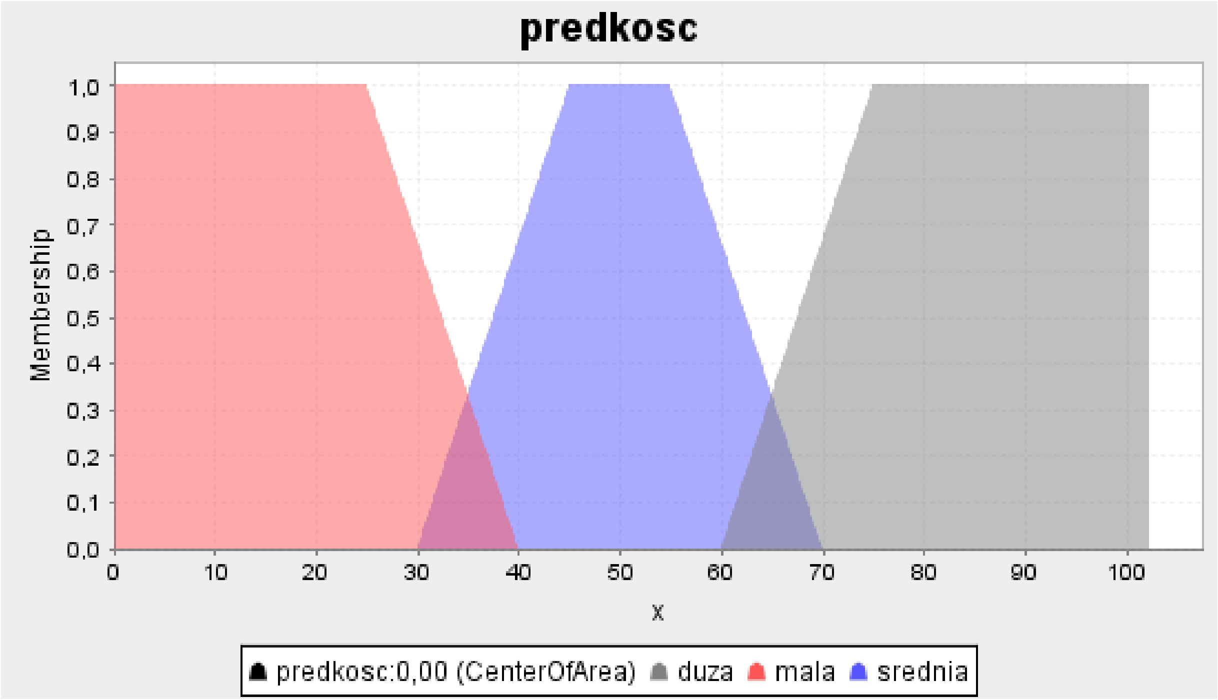
Przyjmijmy, że zmienna predkosc będzie przyjmowała następujące 3 wartości:



FCL - wyjście

```
DEFUZZIFY  predkosc
TERM mala := (0, 1) (25, 1) (40,0);
TERM srednia := (30,0) (45,1) (55, 1) (70, 0);
TERM duza := (60, 0) (75, 1) (100, 1);
METHOD : COA;
END_DEFUZZIFY
```

METODA WYOSTRZANIA



FCL - wyostrzanie

Metody wyostrzania:

COG - Centre of Gravity
COGS - Centre of Gravity for Singletons
COA - Centre of Area
LM - Left Most Maximum
RM - Right Most Maximum

Możemy teraz przystąpić do zdefiniowania bazy reguł.

FCL – baza reguł

Przyjmijmy następującą bazę reguł:

JEŻELI odleglosc jest mala TO predkosc jest mala

JEŻELI odleglosc jest srednia TO predkosc jest srednia

JEŻELI odleglosc jest duza TO predkosc jest duza

W języku FCL zapisujemy to następująco:

```
RULE 1 : IF odleglosc IS mala THEN predkosc IS mala;
RULE 2 : IF odleglosc IS srednia THEN predkosc IS srednia;
RULE 3 : IF odleglosc IS duza THEN predkosc IS duza;
```

FCL – AND i OR

Ponadto musimy określić jeszcze:

- Metodę AND i OR (do wykorzystania po lewej stronie implikacji)

Mamy do wyboru:

operator **OR**		operator **AND**	
keyword for **Algorithm**	**Algorithm**	keyword for **Algorithm**	**Algorithm**
MAX	Max $(\mu_1(x), \mu_2(x))$	MIN	Min$(\mu_1(x), \mu_2(x))$
ASUM	$\mu_1(x)+\mu_2(x) - \mu_1(x)\ \mu_2(x)$	PROD	$\mu_1(x)\ \mu_2(x)$
BSUM	Min$(1, \mu_1(x) + \mu_2(x))$	BDIF	Max $(0, \mu_1(x) + \mu_2(x) - 1)$

W języku FCL zapisujemy to następująco:

```
AND : MIN;
```

Wystarczy, że określimy jeden operator!

FCL - aktywacja

Ponadto musimy określić jeszcze:

- Metodę aktywacji (implikacja!)

Mamy do wyboru:

Name	Keyword	Algorithm
Product	**PROD**	$\mu_1(x)\ \ \mu_2(x)$
Minimum	**MIN**	Min($\mu_1(x)$, $\mu_2(x)$)

W języku FCL zapisujemy to następująco:

```
ACT : MIN;
```

FCL - akumulacja

Ponadto musimy określić jeszcze:

- Metodę akumulacji (suma zbiorów!)

Mamy do wyboru:

Name	Keyword	Formula
Maximum	MAX	$Max(\mu_1(x), \mu_2(x))$
Bounded Sum	BSUM	$Min(1, \mu_1(x) + \mu_2(x))$
Normalised Sum	NSUM	$\frac{\mu_1(x) + \mu_2(x)}{Max(1, MAX(\mu_1(x') + \mu_2(x')))}$

W języku FCL zapisujemy to następująco:

```
ACCU : MAX;
```

FCL – blok reguł

Ostatecznie blok reguł w FCL wygląda następująco:

```
RULEBLOCK No1
AND : MIN;
ACT : MIN;
ACCU : MAX;

RULE 1: IF odleglosc IS mala THEN predkosc IS mala;

RULE 2: IF odleglosc IS srednia THEN predkosc IS srednia;

RULE 3 : IF odleglosc IS duza THEN predkosc IS duza;
END_RULEBLOCK
```

Jak zobaczymy bloków takich może być kilka!