

# PUAV

Wykład 14

# Układy logiki rozmytej

Co to jest?

# Układy logiki rozmytej

Co to jest?

Logika rozmyta (fuzzy logic) jest to dział matematyki precyzyjnie formalizujący nieprecyzyjne, nieformalne ludzkie rozumowanie.

# Układy logiki rozmytej

Co to jest?

Logika rozmyta (fuzzy logic) jest to dział matematyki precyzyjnie formalizujący nieprecyzyjne, nieformalne ludzkie rozumowanie.

Logika rozmyta wykorzystuje pojęcia zbioru rozmytego i algebry zbiorów rozmytych, które można uznać za uogólnienie tradycyjnej teorii mnogości i algebry Boole'a.

# Układy logiki rozmytej

Co to jest?

Logika rozmyta (fuzzy logic) jest to dział matematyki precyzyjnie formalizujący nieprecyzyjne, nieformalne ludzkie rozumowanie.

Logika rozmyta wykorzystuje pojęcia zbioru rozmytego i algebry zbiorów rozmytych, które można uznać za uogólnienie tradycyjnej teorii mnogości i algebry Boole'a.

Logika rozmyta umożliwia tworzenie skutecznych algorytmów decyzyjnych i metod sterowania dla problemów, dla których tradycyjny model matematyczny nie jest znany lub jest zbyt skomplikowany, aby mógł być użyteczny do celów praktycznych.

# Układy logiki rozmytej

Zalety

# Układy logiki rozmytej

## Zalety

Zastosowania logiki rozmytej wynikają z obserwacji, że człowiek jest w stanie sterować bardzo złożonymi obiektami i procesami posługując się prostymi regułami wynikającymi z praktycznej wiedzy i doświadczenia.

# Układy logiki rozmytej

## Zalety

Zastosowania logiki rozmytej wynikają z obserwacji, że człowiek jest w stanie sterować bardzo złożonymi obiektami i procesami posługując się prostymi regułami wynikającymi z praktycznej wiedzy i doświadczenia.

Algorytmy logiki rozmytej nie wymagają dużej dokładności danych wejściowych ani dużej precyzji obliczeń.



# Układy logiki rozmytej

## Zalety

Zastosowania logiki rozmytej wynikają z obserwacji, że człowiek jest w stanie sterować bardzo złożonymi obiektami i procesami posługując się prostymi regułami wynikającymi z praktycznej wiedzy i doświadczenia.

Algorytmy logiki rozmytej nie wymagają dużej dokładności danych wejściowych ani dużej precyzji obliczeń.

Algorytmy logiki rozmytej prowadzą do prostych implementacji technicznych.

# Układy logiki rozmytej

## Zalety

Zastosowania logiki rozmytej wynikają z obserwacji, że człowiek jest w stanie sterować bardzo złożonymi obiektami i procesami posługując się prostymi regułami wynikającymi z praktycznej wiedzy i doświadczenia.

Algorytmy logiki rozmytej nie wymagają dużej dokładności danych wejściowych ani dużej precyzji obliczeń.

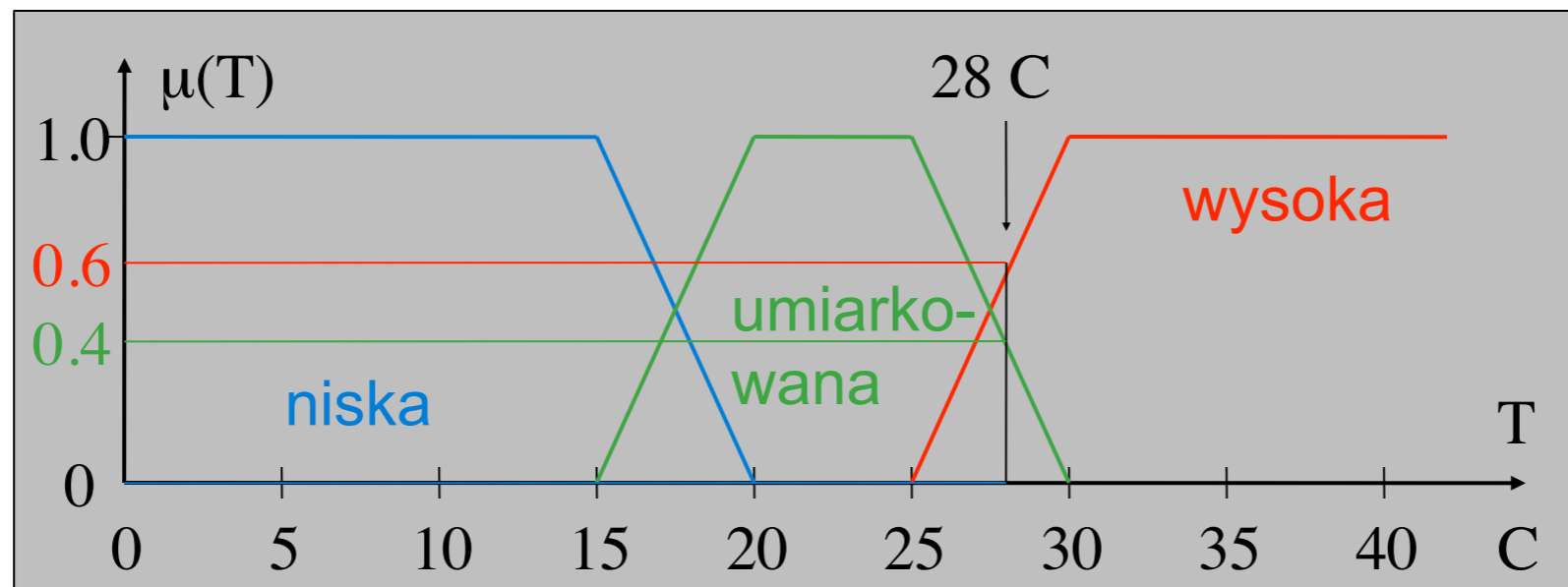
Algorytmy logiki rozmytej prowadzą do prostych implementacji technicznych.

Możliwe są implementacje w postaci układów analogowych.

# Układy logiki rozmytej

## Zbiory rozmyte

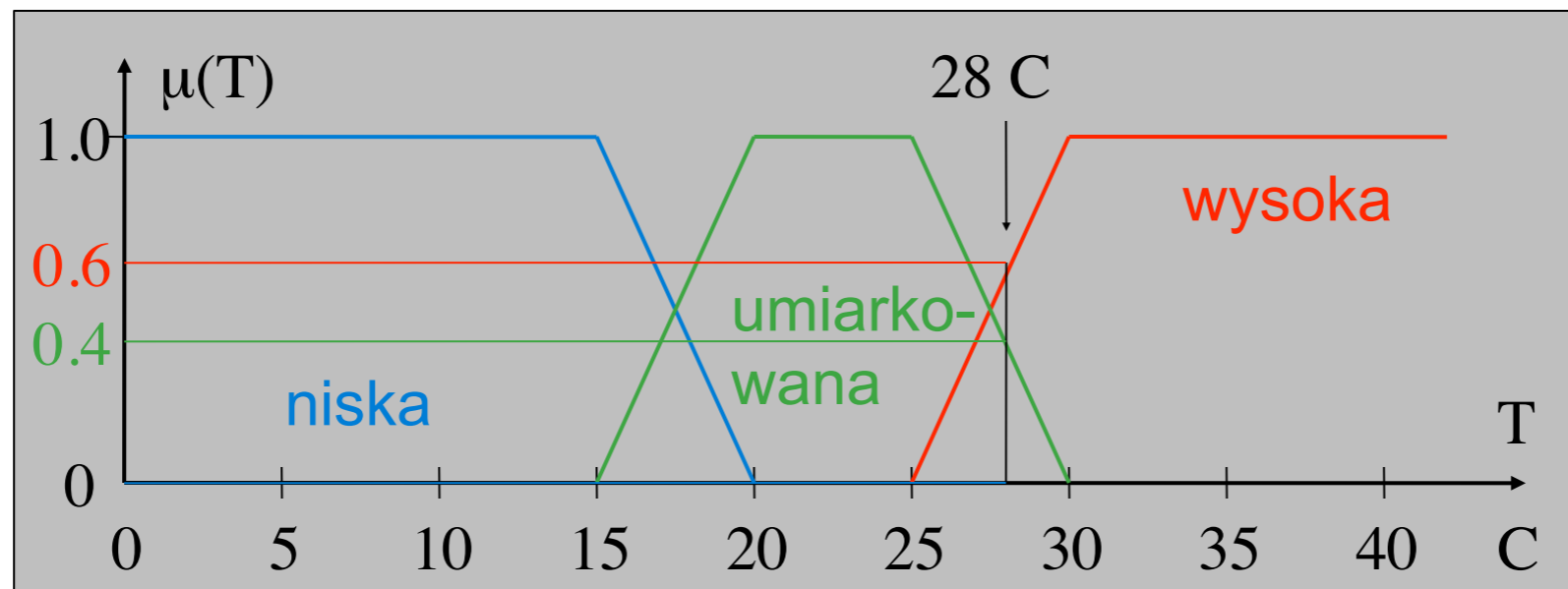
Przykład trzech zbiorów rozmytych dla temperatury



# Układy logiki rozmytej

## Zbiory rozmyte

Przykład trzech zbiorów rozmytych dla temperatury

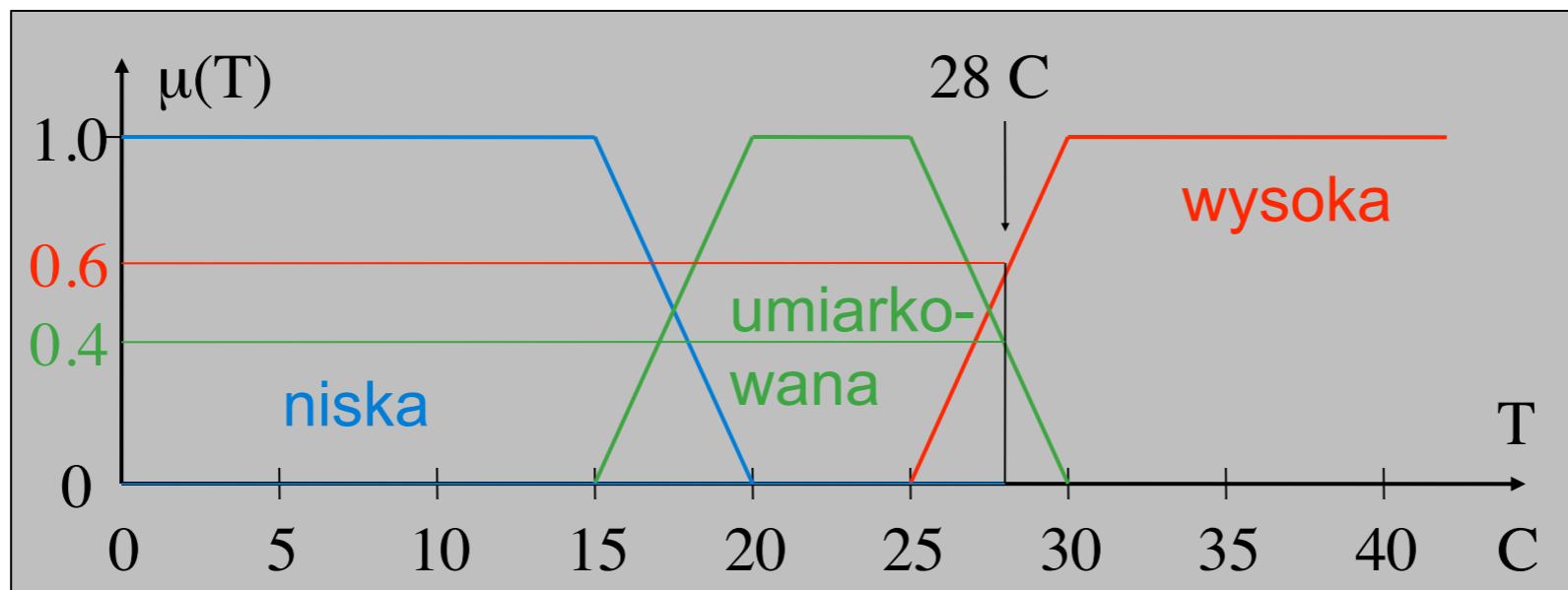


Zbiór rozmyty charakteryzuje się *funkcją przynależności*, która może przybierać wartości od 0 do 1.

# Układy logiki rozmytej

## Zbiory rozmyte

Przykład trzech zbiorów rozmytych dla temperatury

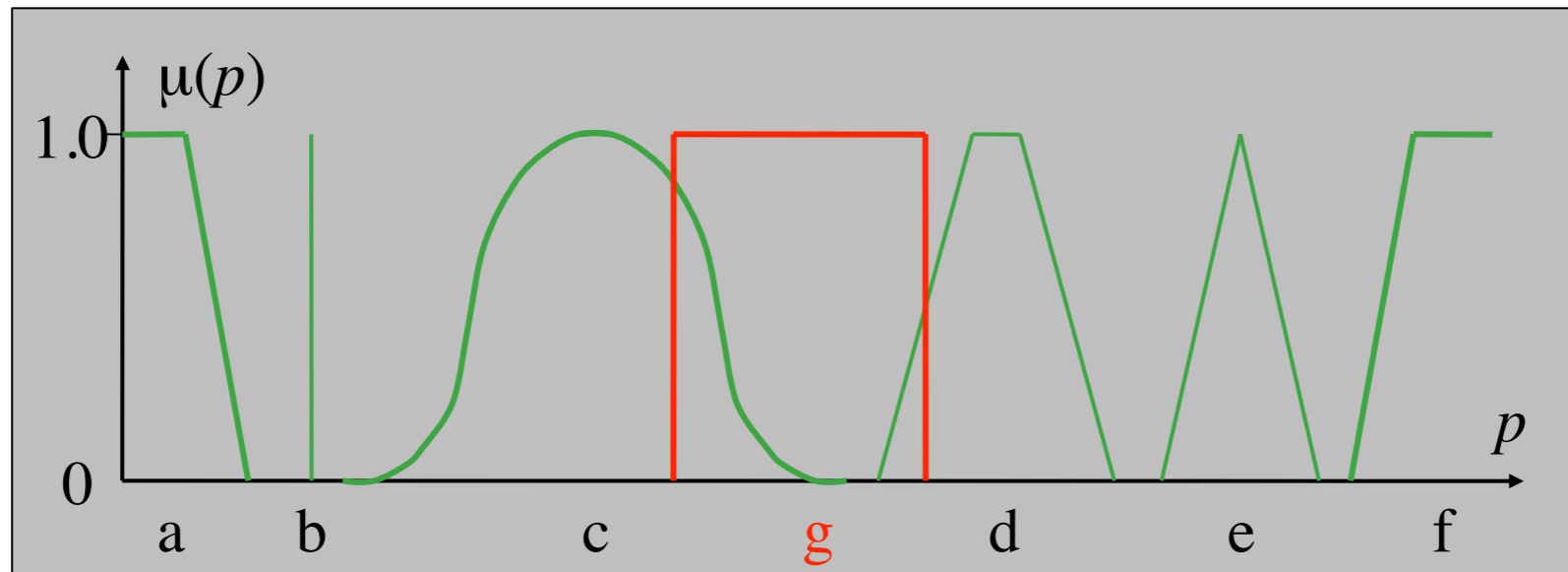


Zbiór rozmyty charakteryzuje się *funkcją przynależności*, która może przybierać wartości od 0 do 1.

Przykładowo, dla zbiorów określonych jak wyżej temperatura 28 C należy do zbioru temperatur umiarkowanych z wartością przynależności 0,4 oraz do zbioru temperatur wysokich z wartością przynależności 0,6.

# Układy logiki rozmytej

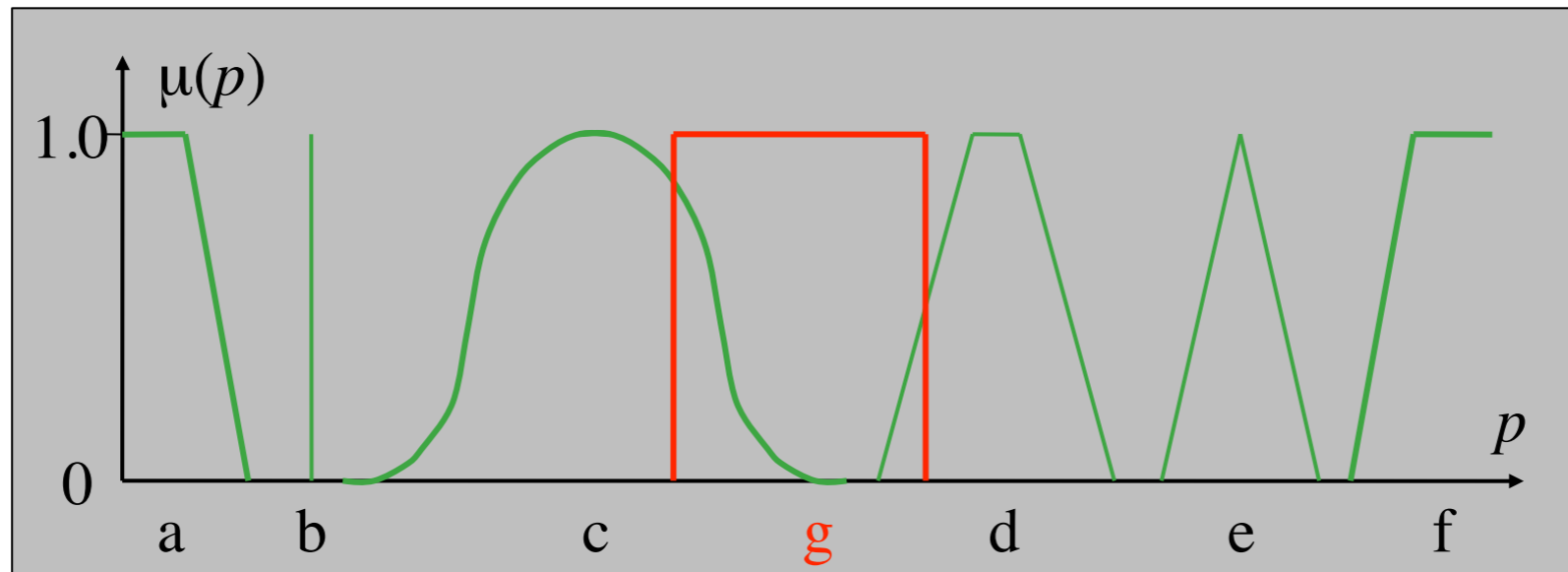
Zbiory rozmyte i funkcje przynależności



Funkcje przynależności mogą mieć różne kształty

# Układy logiki rozmytej

## Zbiory rozmyte i funkcje przynależności

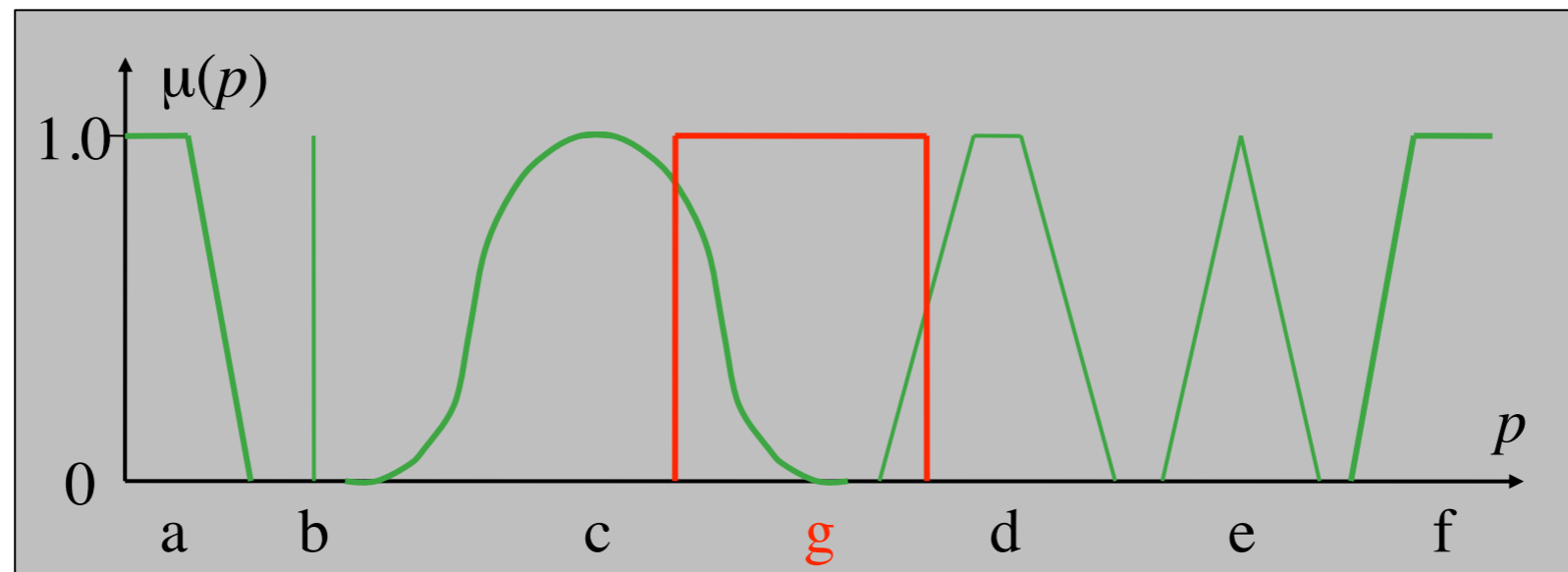


Funkcje przynależności mogą mieć różne kształty

a: funkcja typu Z

# Układy logiki rozmytej

## Zbiory rozmyte i funkcje przynależności



Funkcje przynależności mogą mieć różne kształty

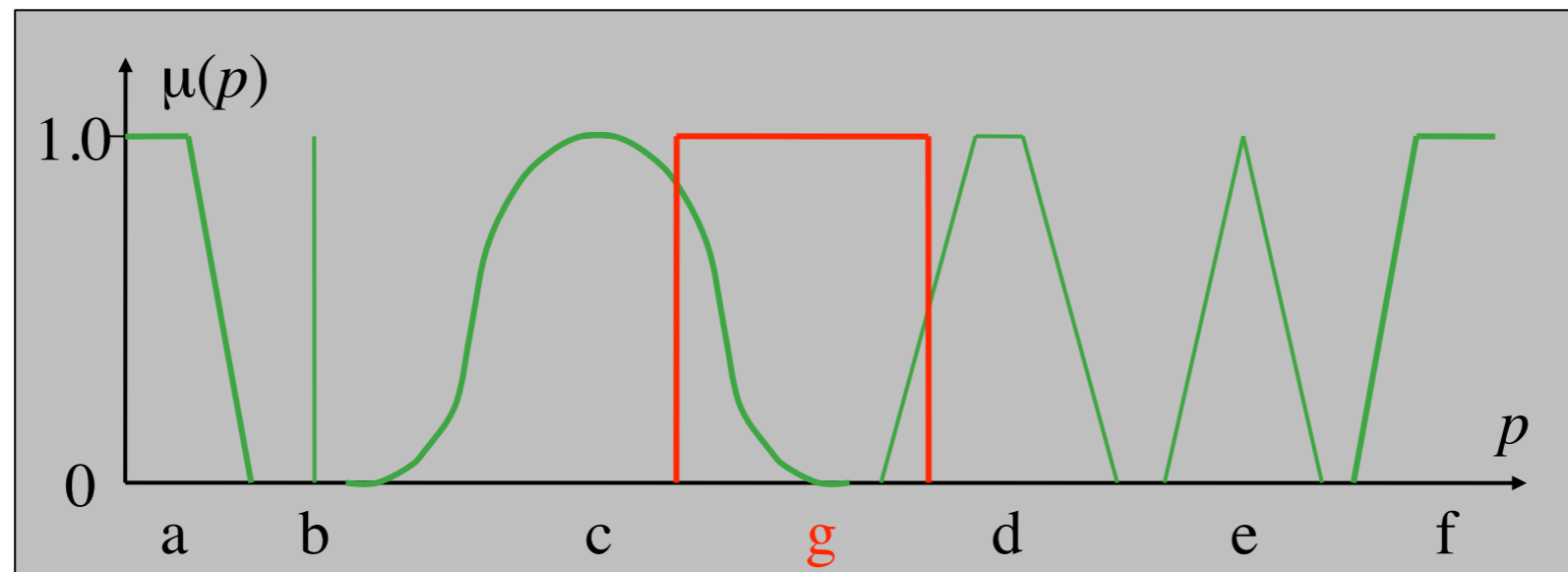
a: funkcja typu Z

b: singleton



# Układy logiki rozmytej

## Zbiory rozmyte i funkcje przynależności



Funkcje przynależności mogą mieć różne kształty

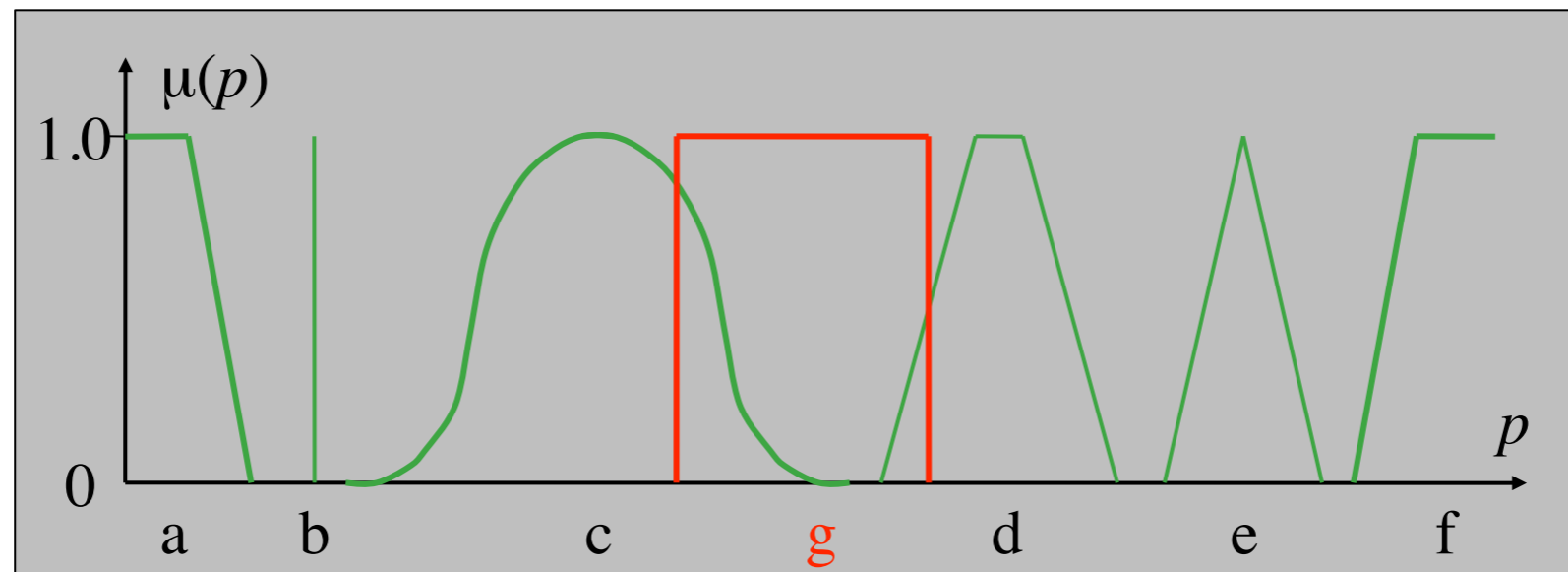
a: funkcja typu Z

b: singleton

c: funkcja Gaussa

# Układy logiki rozmytej

## Zbiory rozmyte i funkcje przynależności



Funkcje przynależności mogą mieć różne kształty

a: funkcja typu Z

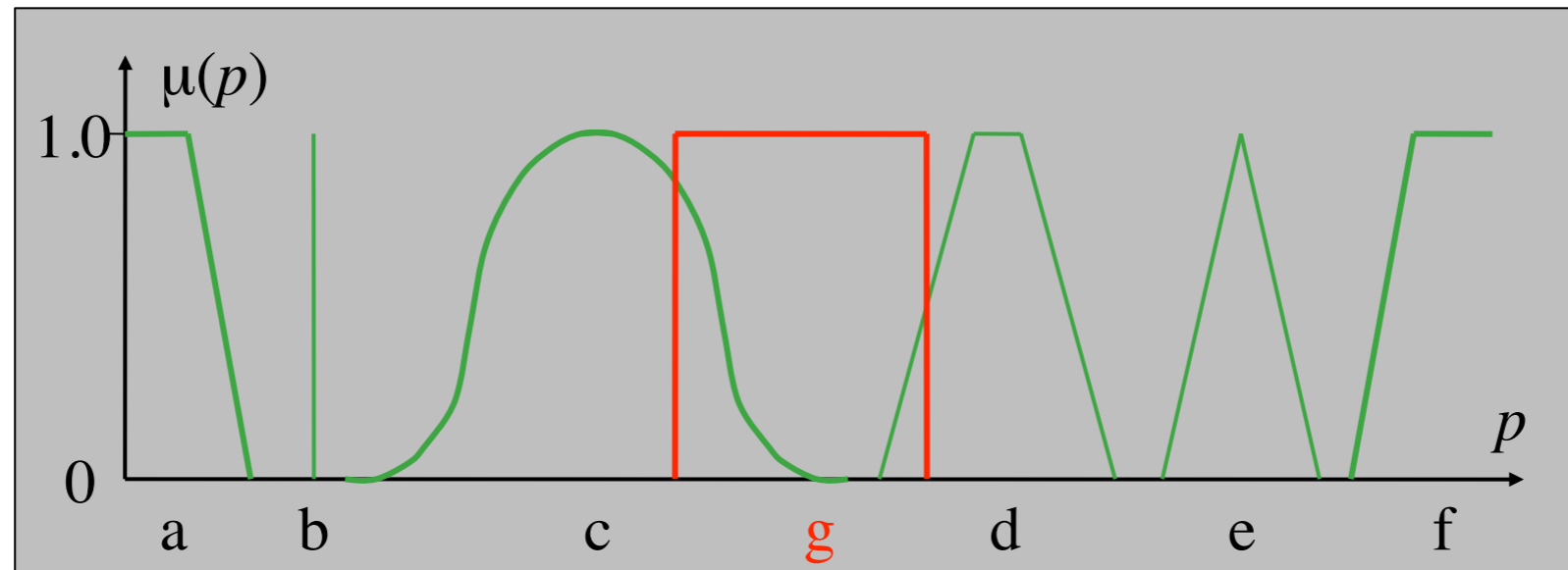
b: singleton

c: funkcja Gaussa

d: funkcja trapezoidalna

# Układy logiki rozmytej

## Zbiory rozmyte i funkcje przynależności



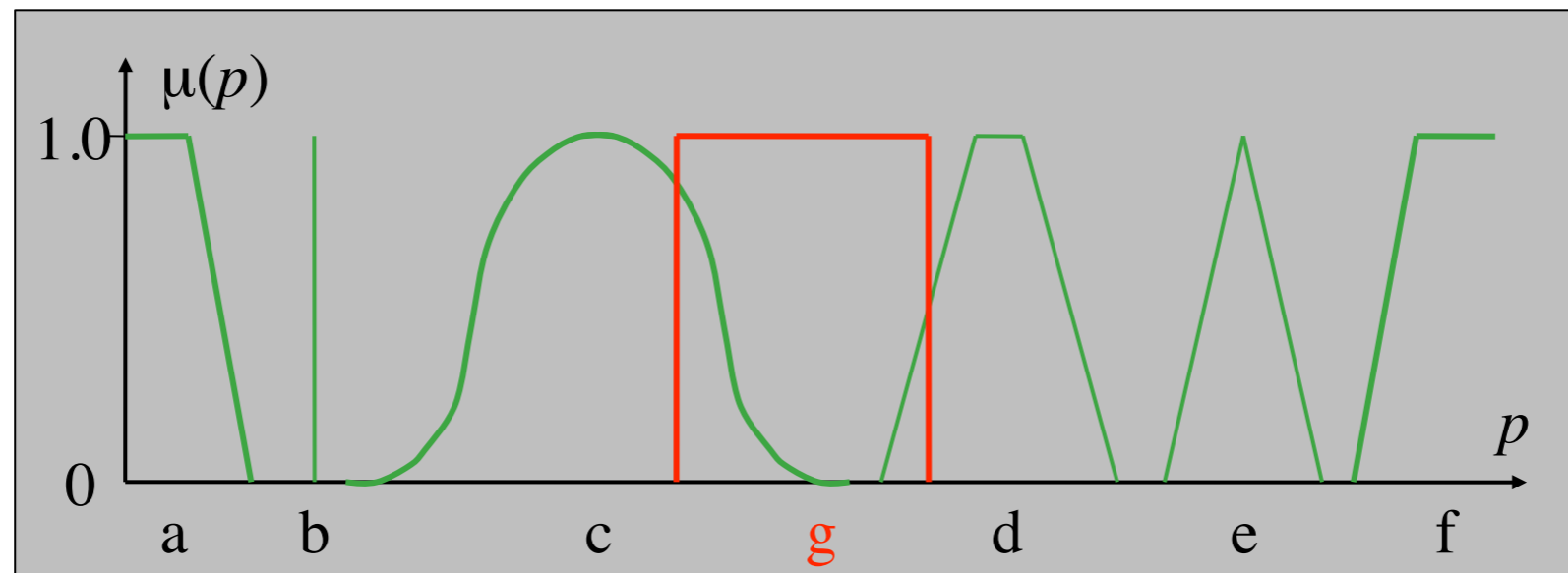
Funkcje przynależności mogą mieć różne kształty

a: funkcja typu Z  
b: singleton  
c: funkcja Gaussa

d: funkcja trapezoidalna  
e: funkcja trójkątna

# Układy logiki rozmytej

## Zbiory rozmyte i funkcje przynależności



Funkcje przynależności mogą mieć różne kształty

a: funkcja typu Z

b: singleton

c: funkcja Gaussa

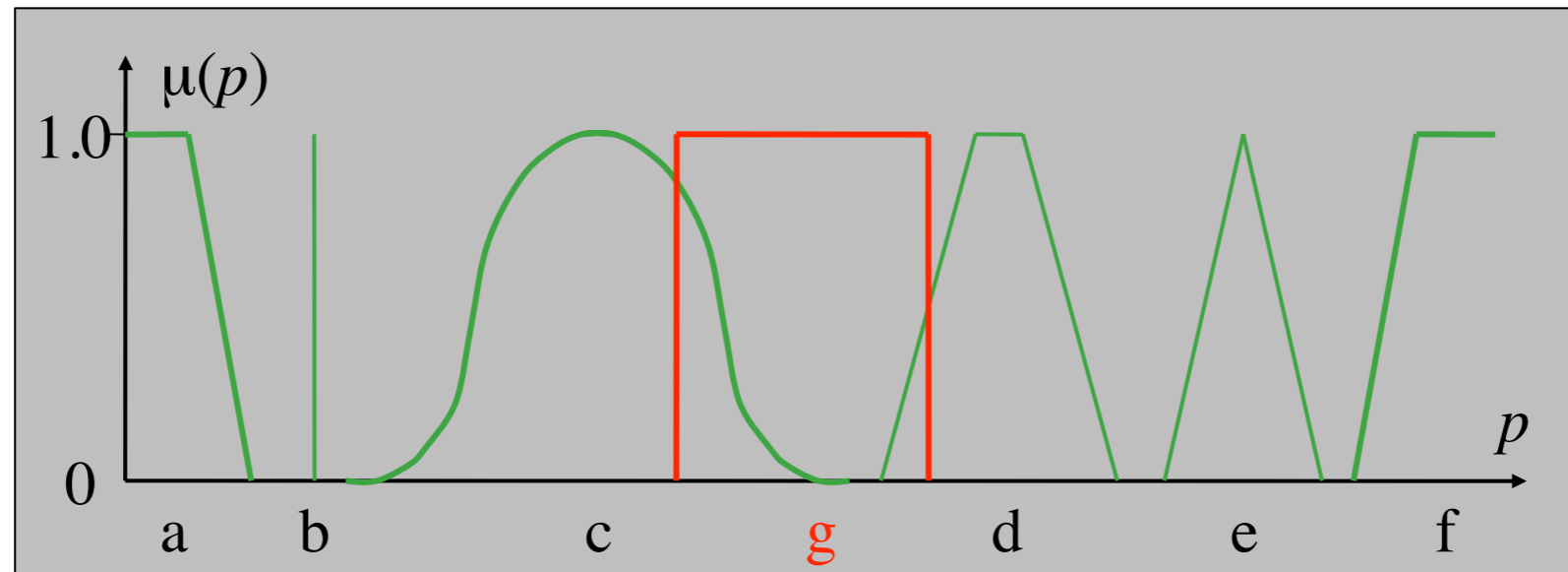
d: funkcja trapezoidalna

e: funkcja trójkątna

f: funkcja typu S

# Układy logiki rozmytej

## Zbiory rozmyte i funkcje przynależności



Funkcje przynależności mogą mieć różne kształty

a: funkcja typu Z

b: singleton

c: funkcja Gaussa

d: funkcja trapezoidalna

e: funkcja trójkątna

f: funkcja typu S

g: zbiór tradycyjny

# Układy logiki rozmytej

Operacje logiczne na zbiorach rozmytych

# Układy logiki rozmytej

Operacje logiczne na zbiorach rozmytych

**B = NIE A**; B to zbiór rozmyty, dla którego  $\mu_B = 1 - \mu_A$

# Układy logiki rozmytej

Operacje logiczne na zbiorach rozmytych

**B = NIE A**; B to zbiór rozmyty, dla którego  $\mu_B = 1 - \mu_A$

**C = A LUB B**; C to zbiór rozmyty, dla którego  $\mu_C = \max(\mu_A, \mu_B)$



# Układy logiki rozmytej

Operacje logiczne na zbiorach rozmytych

**B = NIE A**; B to zbiór rozmyty, dla którego  $\mu_B = 1 - \mu_A$

**C = A LUB B**; C to zbiór rozmyty, dla którego  $\mu_C = \max(\mu_A, \mu_B)$

**C = A I B**; C to zbiór rozmyty, dla którego  $\mu_C = \min(\mu_A, \mu_B)$

# Układy logiki rozmytej

Operacje logiczne na zbiorach rozmytych

**B = NIE A**; B to zbiór rozmyty, dla którego  $\mu_B = 1 - \mu_A$

**C = A LUB B**; C to zbiór rozmyty, dla którego  $\mu_C = \max(\mu_A, \mu_B)$

**C = A I B**; C to zbiór rozmyty, dla którego  $\mu_C = \min(\mu_A, \mu_B)$

Operacje logiczne na zbiorach rozmytych sprowadzają się do operacji *arytmetycznych* na wartościach funkcji przynależności.

# Układy logiki rozmytej

Operacje logiczne na zbiorach rozmytych

**B = NIE A**; B to zbiór rozmyty, dla którego  $\mu_B = 1 - \mu_A$

**C = A LUB B**; C to zbiór rozmyty, dla którego  $\mu_C = \max(\mu_A, \mu_B)$

**C = A I B**; C to zbiór rozmyty, dla którego  $\mu_C = \min(\mu_A, \mu_B)$

Operacje logiczne na zbiorach rozmytych sprowadzają się do operacji *arytmetycznych* na wartościach funkcji przynależności.

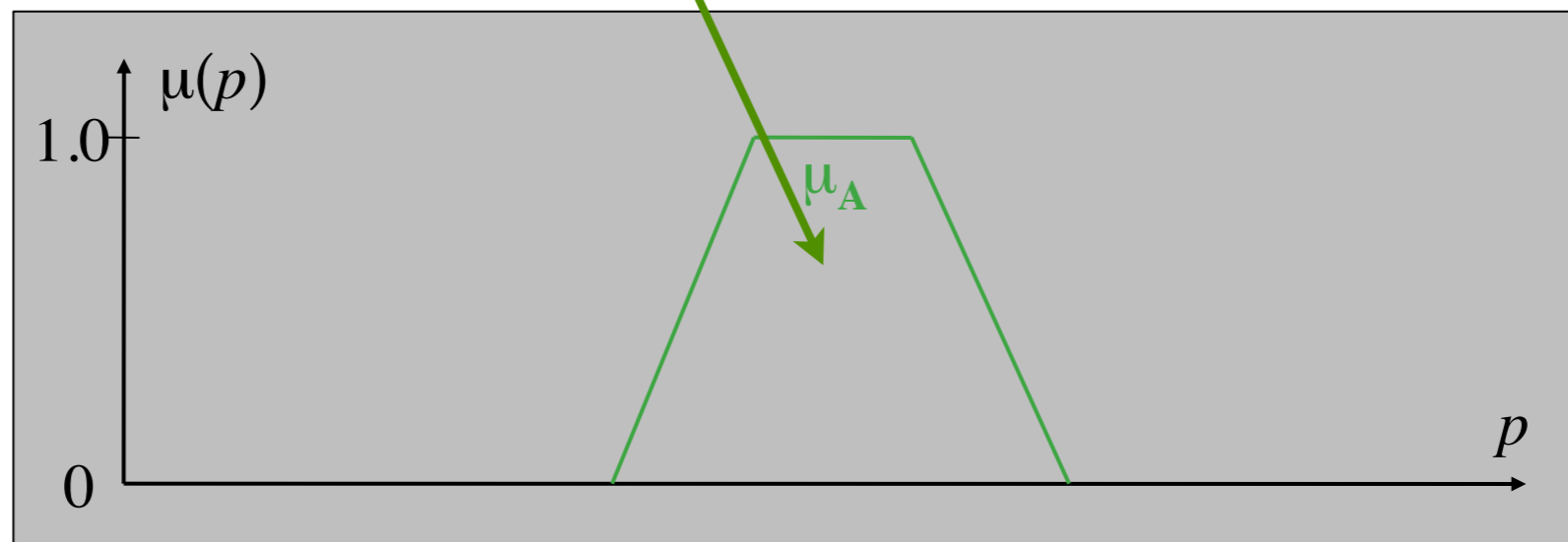
Operacje te mogą być więc wykonywane cyfrowo, a także przez niezbyt skomplikowane układy analogowe.

# Układy logiki rozmytej

Operacje logiczne na zbiorach rozmytych

Funkcja NOT (NIE, negacja)

$$B = \text{NIE } A \text{ gdy } \mu_B = 1 - \mu_A$$

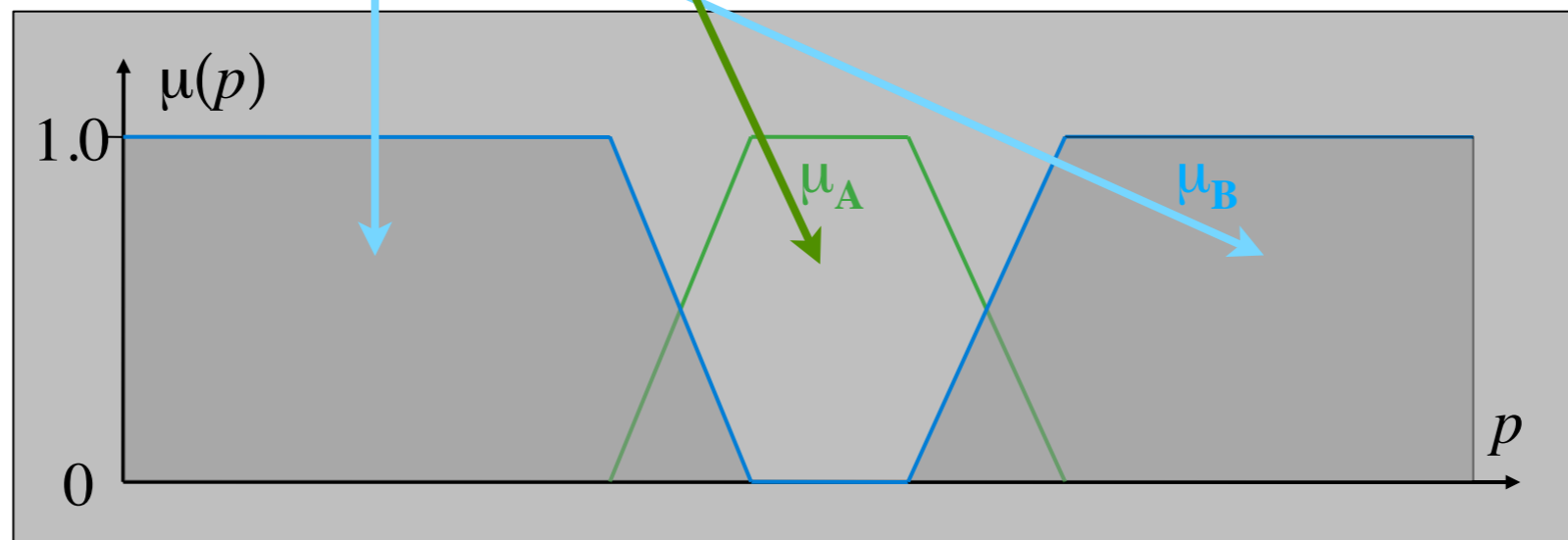


# Układy logiki rozmytej

Operacje logiczne na zbiorach rozmytych

Funkcja NOT (NIE, negacja)

$$B = \text{NIE } A \text{ gdy } \mu_B = 1 - \mu_A$$

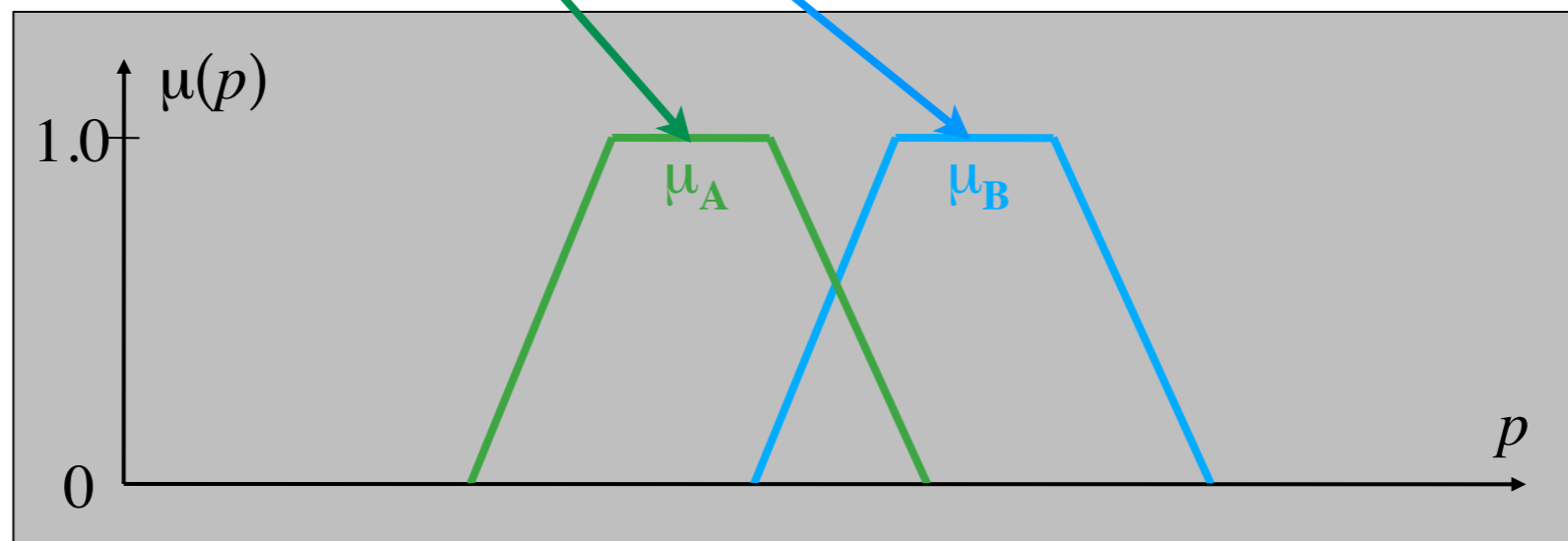


# Układy logiki rozmytej

Operacje logiczne na zbiorach rozmytych

Funkcja OR (LUB)

$$C = A \text{ LUB } B \text{ gdy } \mu_C = \max(\mu_A, \mu_B)$$

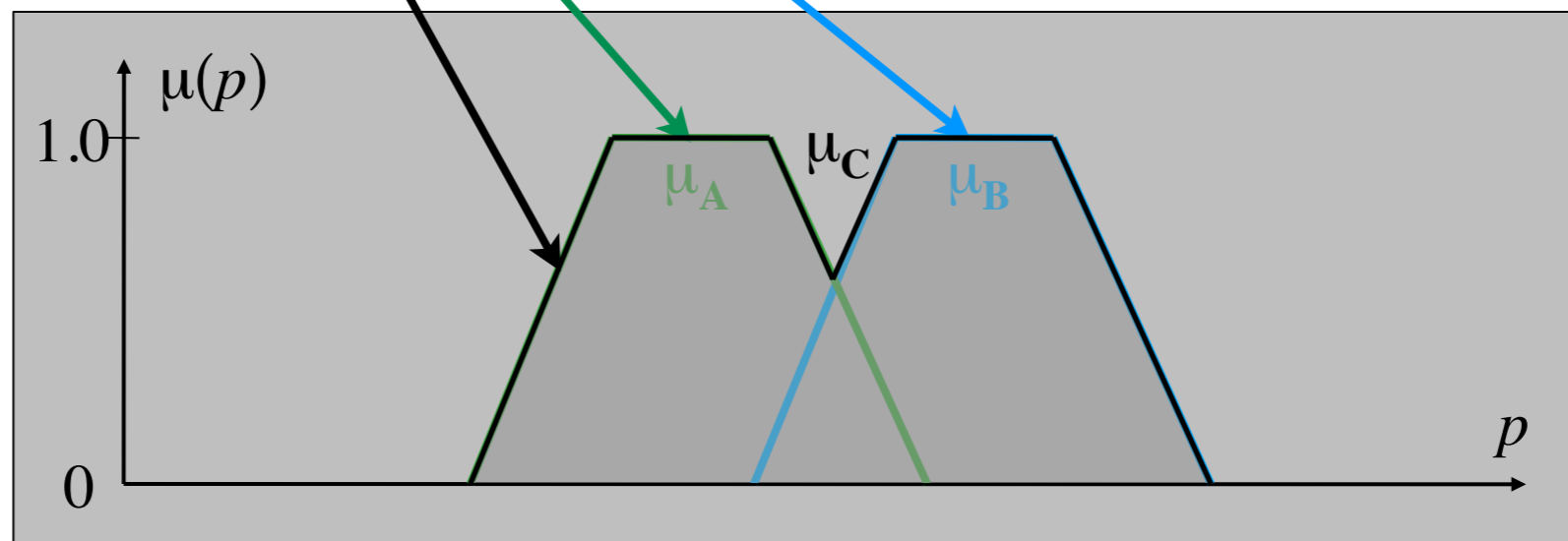


# Układy logiki rozmytej

Operacje logiczne na zbiorach rozmytych

Funkcja OR (LUB)

$$C = A \text{ LUB } B \text{ gdy } \mu_C = \max(\mu_A, \mu_B)$$

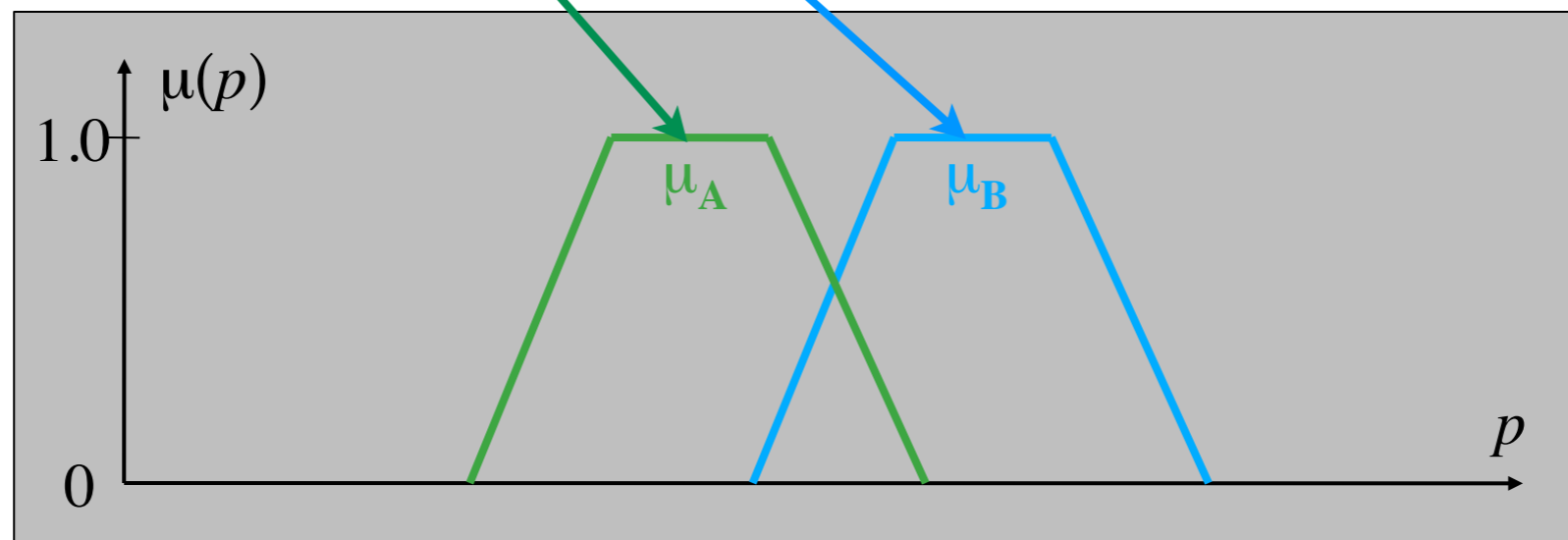


# Układy logiki rozmytej

Operacje logiczne na zbiorach rozmytych

Funkcja AND (I)

$$C = A \text{ I } B \text{ gdy } \mu_C = \min(\mu_A, \mu_B)$$



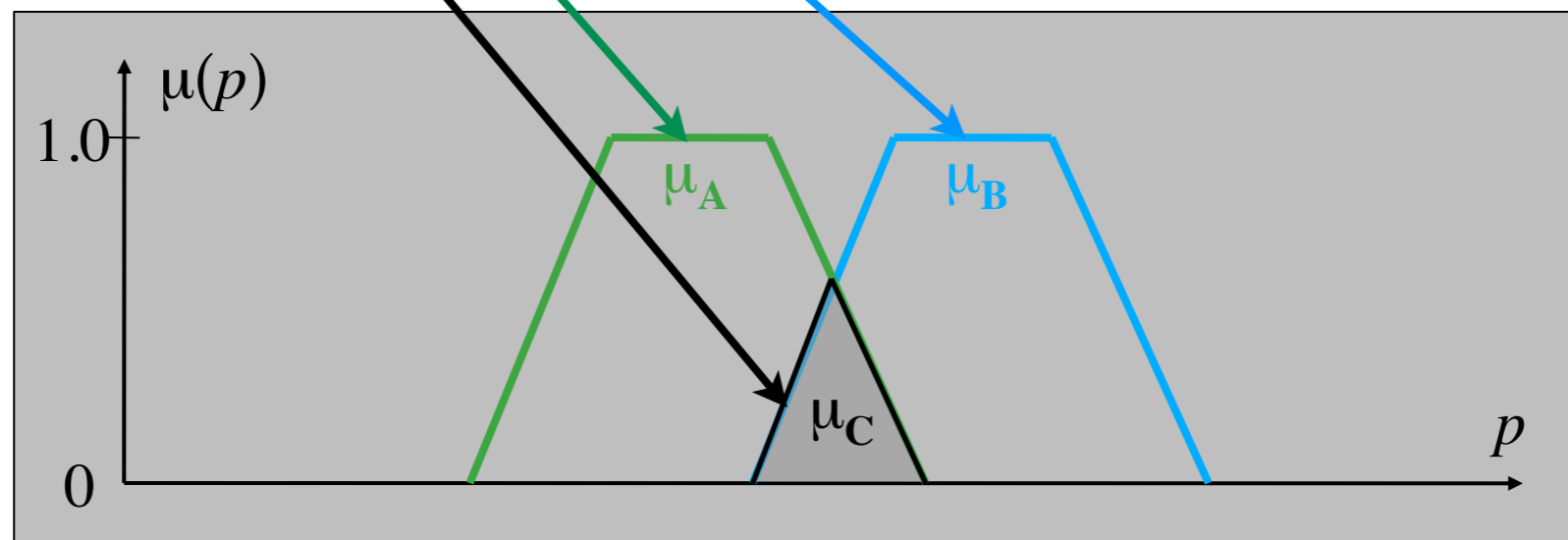


# Układy logiki rozmytej

Operacje logiczne na zbiorach rozmytych

Funkcja AND (I)

$$C = A \text{ I } B \text{ gdy } \mu_C = \min(\mu_A, \mu_B)$$



# Układy logiki rozmytej

Zmienne lingwistyczne

# Układy logiki rozmytej

## Zmienne lingwistyczne

*Zmienną lingwistyczną* nazywamy zmienną, której wartościami są terminy zaczerpnięte z języka naturalnego, a nie liczby

# Układy logiki rozmytej

## Zmienne lingwistyczne

*Zmienną lingwistyczną* nazywamy zmienną, której wartościami są terminy zaczerpnięte z języka naturalnego, a nie liczby

Przykład: jeśli “temperatura” jest zmienną lingwistyczną, to możemy jej nadawać wartości takie, jak “niska”, “wysoka”, “umiarkowana”, “upał”, “zimno” itp.

# Układy logiki rozmytej

## Zmienne lingwistyczne

*Zmienną lingwistyczną* nazywamy zmienną, której wartościami są terminy zaczerpnięte z języka naturalnego, a nie liczby

Przykład: jeśli “temperatura” jest zmienną lingwistyczną, to możemy jej nadawać wartości takie, jak “niska”, “wysoka”, “umiarkowana”, “upał”, “zimno” itp.

Wartościom zmiennych lingwistycznych przyporządkowuje się odpowiednio zdefiniowane zbiory rozmyte.

# Układy logiki rozmytej

## Zmienne lingwistyczne

*Zmienną lingwistyczną* nazywamy zmienną, której wartościami są terminy zaczerpnięte z języka naturalnego, a nie liczby

Przykład: jeśli “temperatura” jest zmienną lingwistyczną, to możemy jej nadawać wartości takie, jak “niska”, “wysoka”, “umiarkowana”, “upał”, “zimno” itp.

Wartościom zmiennych lingwistycznych przyporządkowuje się odpowiednio zdefiniowane zbiory rozmyte.

Szczególne przypadki: zmiennej, której odpowiada ściśle określona wartość liczbowa (np. gdy “temperatura” = “normalna” oznacza 18 C) przyporządkowany jest singleton; zakresowi ściśle określonemu (np. “od 15,0 C do 25,0 C”) odpowiada zbiór tradycyjny

# Układy logiki rozmytej

Reguły wnioskowania

# Układy logiki rozmytej

## Reguły wnioskowania

Zmienne lingwistyczne pozwalają wyrażać zależności oraz reguły wnioskowania lub algorytmy sterowania w języku naturalnym



# Układy logiki rozmytej

## Reguły wnioskowania

Zmienne lingwistyczne pozwalają wyrażać zależności oraz reguły wnioskowania lub algorytmy sterowania w języku naturalnym

Przykład reguły sterowania układem klimatyzacji:

**JĘŚLI** (temperatura **JEST** wysoka) **I** (wilgotność **JEST** niska)  
**TO** (nawiew ciepłego powietrza **JEST** bardzo mały)

# Układy logiki rozmytej

## Reguły wnioskowania

Zmienne lingwistyczne pozwalają wyrażać zależności oraz reguły wnioskowania lub algorytmy sterowania w języku naturalnym

Przykład reguły sterowania układem klimatyzacji:

**JĘŚLI** (temperatura **JEST** wysoka) **I** (wilgotność **JEST** niska)  
**TO** (nawiew ciepłego powietrza **JEST** bardzo mały)

Tutaj “temperatura”, “wilgotność” i “nawiew ciepłego powietrza” są zmiennymi lingwistycznymi, “wysoka”, “niska” i “bardzo mały” to wartości tych zmiennych (odpowiadają im zbiory rozmyte), a całość jest *regułą wnioskowania*.

# Układy logiki rozmytej

## Reguły wnioskowania

Zmienne lingwistyczne pozwalają wyrażać zależności oraz reguły wnioskowania lub algorytmy sterowania w języku naturalnym

Przykład reguły sterowania układem klimatyzacji:

**JĘŚLI** (temperatura **JEST** wysoka) **I** (wilgotność **JEST** niska)  
**TO** (nawiew ciepłego powietrza **JEST** bardzo mały)

Tutaj “temperatura”, “wilgotność” i “nawiew ciepłego powietrza” są zmiennymi lingwistycznymi, “wysoka”, “niska” i “bardzo mały” to wartości tych zmiennych (odpowiadają im zbiory rozmyte), a całość jest *regułą wnioskowania*.

**JEST** oznacza tu przynależność do zbioru rozmytego (w stopniu określonym przez funkcję przynależności).

# Układy logiki rozmytej

Reguły wnioskowania

# Układy logiki rozmytej

## Reguły wnioskowania

Przykład ogólniejszy:

**JEŚLI** (a **JEST** A) **I** (b **JEST** B) **LUB** (c **JEST** C)  
**TO** (d **JEST** D)

# Układy logiki rozmytej

## Reguły wnioskowania

Przykład ogólniejszy:

**JEŚLI** (a **JEST** A) **I** (b **JEST** B) **LUB** (c **JEST** C)  
**TO** (d **JEST** D)

gdzie A, B, C są to zbiory rozmyte dla zmiennych wejściowych, D jest zbiorem rozmytym dla zmiennej wyjściowej, JEST oznacza wartość funkcji przynależności dla zmiennych wejściowych i wyjściowych do odpowiednich zbiorów rozmytych

# Układy logiki rozmytej

## Reguły wnioskowania

Przykład ogólniejszy:

**JEŚLI** (a **JEST** A) **I** (b **JEST** B) **LUB** (c **JEST** C)  
**TO** (d **JEST** D)

gdzie A, B, C są to zbiory rozmyte dla zmiennych wejściowych, D jest zbiorem rozmytym dla zmiennej wyjściowej, JEST oznacza wartość funkcji przynależności dla zmiennych wejściowych i wyjściowych do odpowiednich zbiorów rozmytych

Z tego rodzaju reguł, wyrażonych w języku naturalnym przy użyciu zmiennych lingwistycznych, konstruuje się algorytmy klasyfikacji, sterowania czy też podejmowania decyzji.

# Układy logiki rozmytej

Sterownik “rozmyty” (fuzzy logic controller)



# Układy logiki rozmytej

Sterownik “rozmyty” (fuzzy logic controller)

Sterownik “rozmyty” jest to szczególnego rodzaju mikrokontroler, układ implementujący sprzętowo algorytmy zbudowane przy wykorzystaniu logiki rozmytej. Na wejściu i na wyjściu są konkretne wartości liczbowe, jak na przykład wartość temperatury, wilgotności czy szybkości nawiewu ciepłego powietrza.

# Układy logiki rozmytej

Sterownik “rozmyty” (fuzzy logic controller)

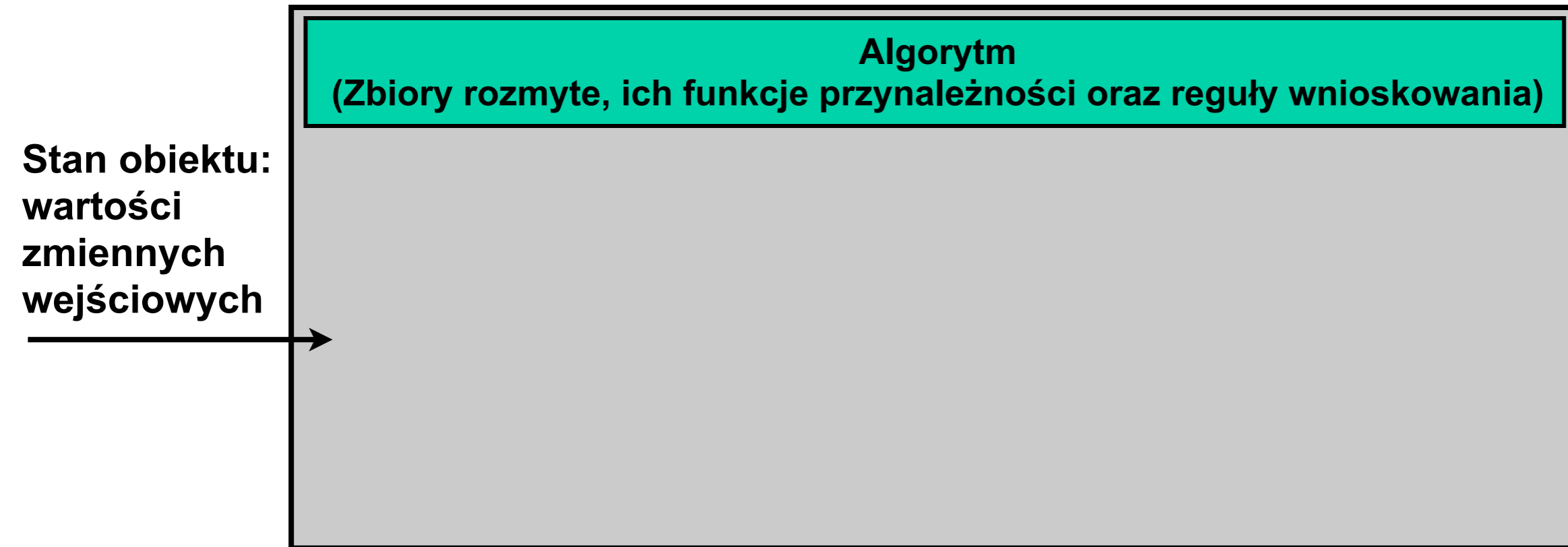
Sterownik “rozmyty” jest to szczególnego rodzaju mikrokontroler, układ implementujący sprzętowo algorytmy zbudowane przy wykorzystaniu logiki rozmytej. Na wejściu i na wyjściu są konkretne wartości liczbowe, jak na przykład wartość temperatury, wilgotności czy szybkości nawiewu ciepłego powietrza.

Algorytmy logiki rozmytej działają w trzech krokach:

- fuzyfikacja (“rozmywanie”): polega na określeniu wartości funkcji przynależności dla wielkości wejściowych
- wnioskowanie: polega na wykonaniu operacji logicznych na zbiorach rozmytych (tj. operacji na wartościach funkcji przynależności) zgodnie z regułami wnioskowania
- defuzyfikacja (“wyostrzanie”): określanie wartości liczbowych zmiennych wyjściowych na podstawie kształtu zbiorów rozmytych otrzymanych w wyniku procedury wnioskowania.

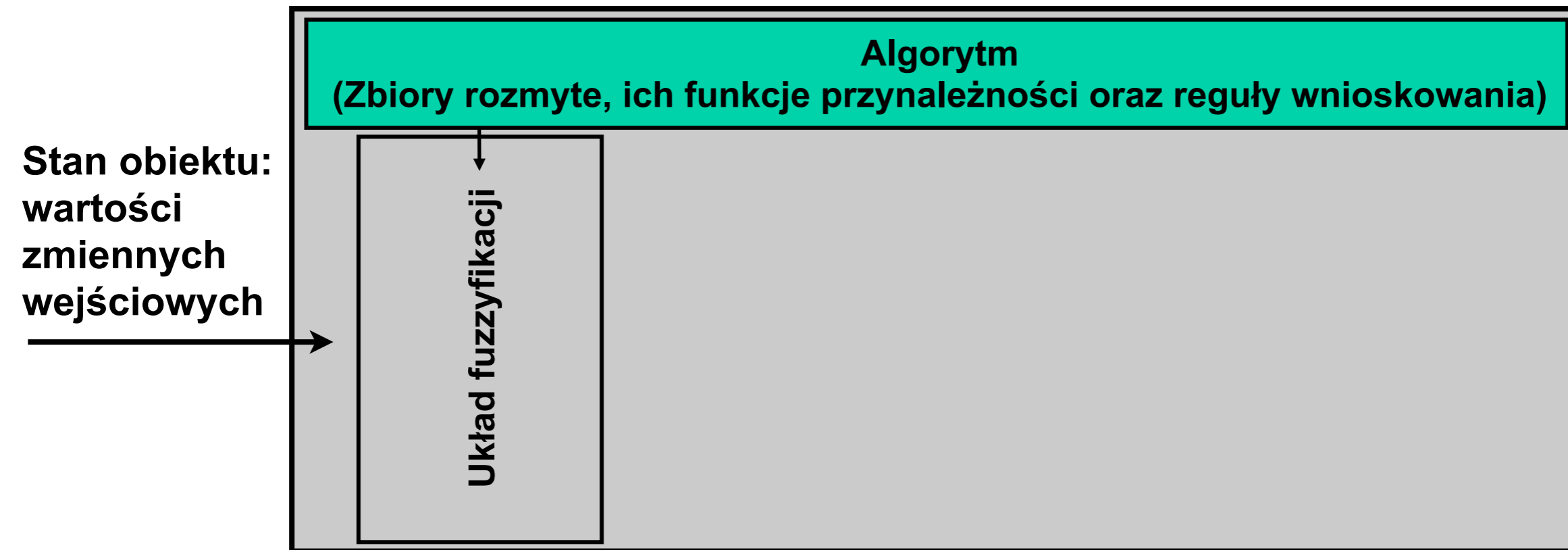
# Układy logiki rozmytej

Sterownik “rozmyty” (fuzzy logic controller)



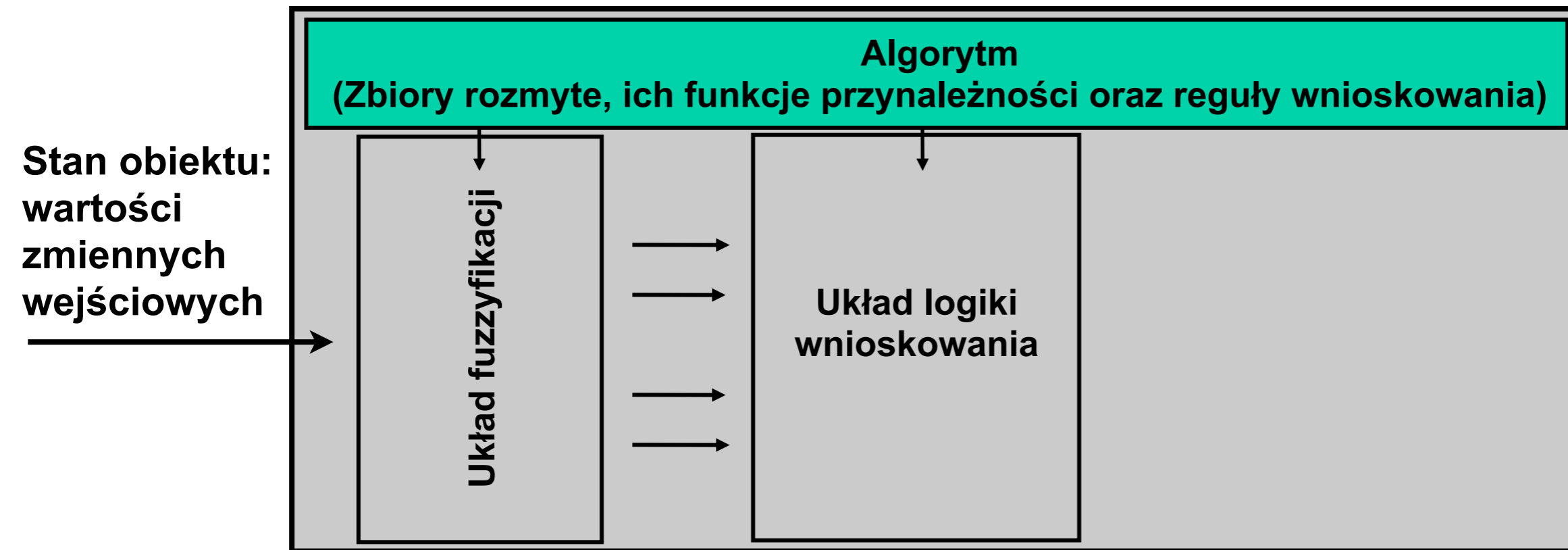
# Układy logiki rozmytej

Sterownik “rozmyty” (fuzzy logic controller)



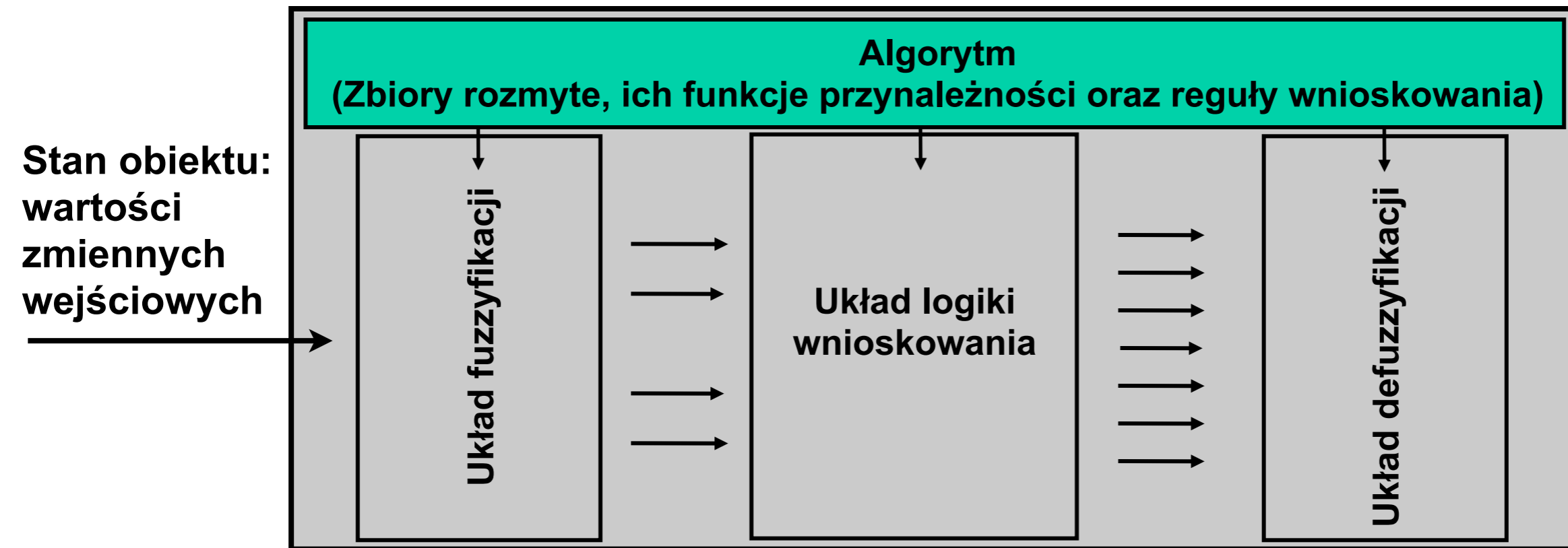
# Układy logiki rozmytej

Sterownik "rozmyty" (fuzzy logic controller)



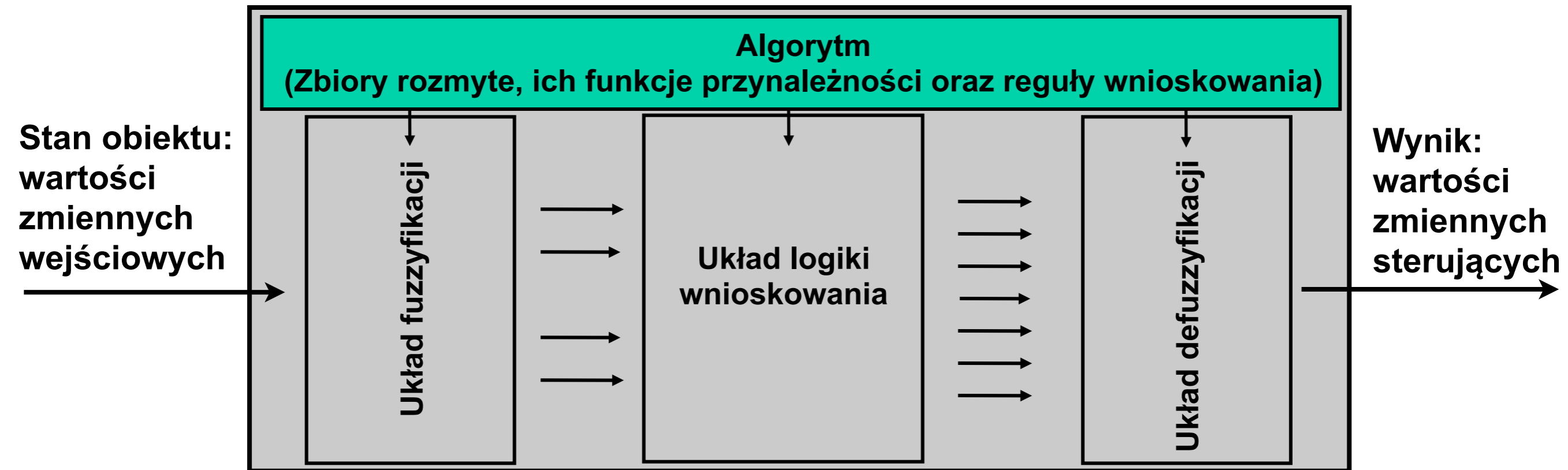
# Układy logiki rozmytej

Sterownik “rozmyty” (fuzzy logic controller)



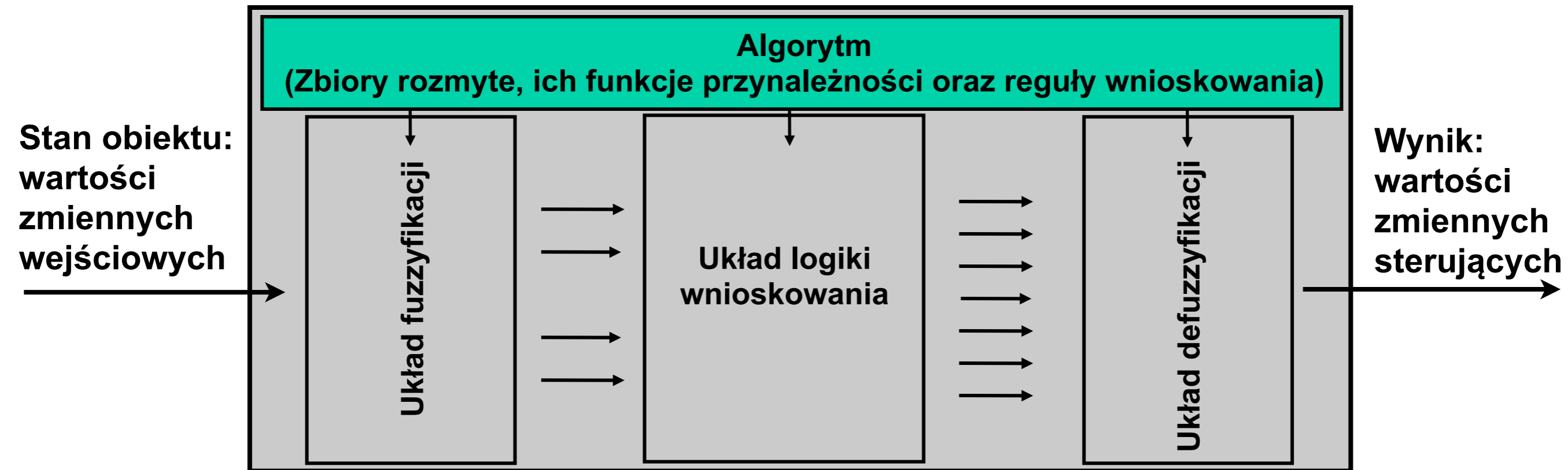
# Układy logiki rozmytej

Sterownik "rozmyty" (fuzzy logic controller)



# Układy logiki rozmytej

Sterownik “rozmyty” (fuzzy logic controller)

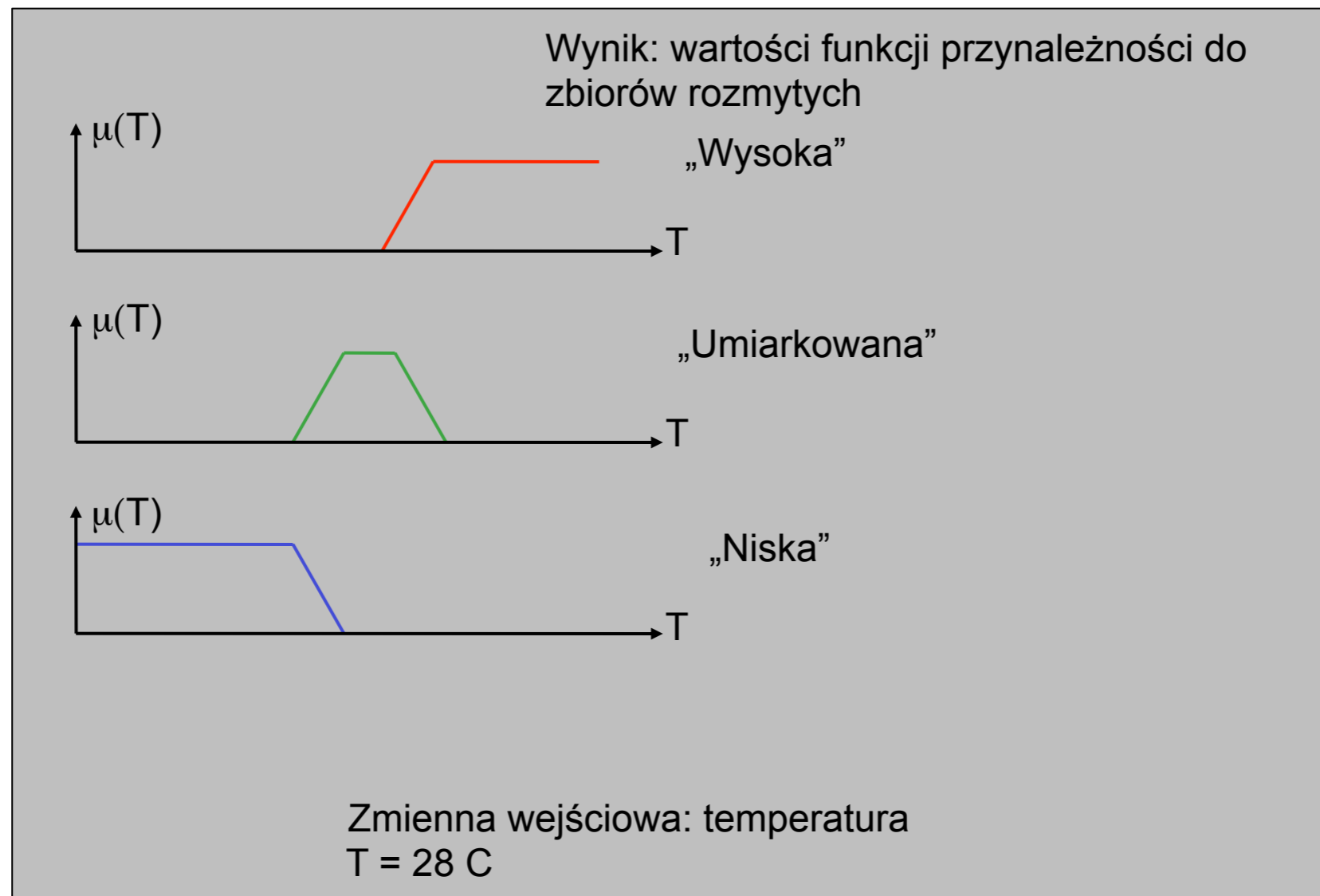


Sterownik “rozmyty” może implementować jeden konkretny algorytm lub być układem programowalnym



# Układy logiki rozmytej

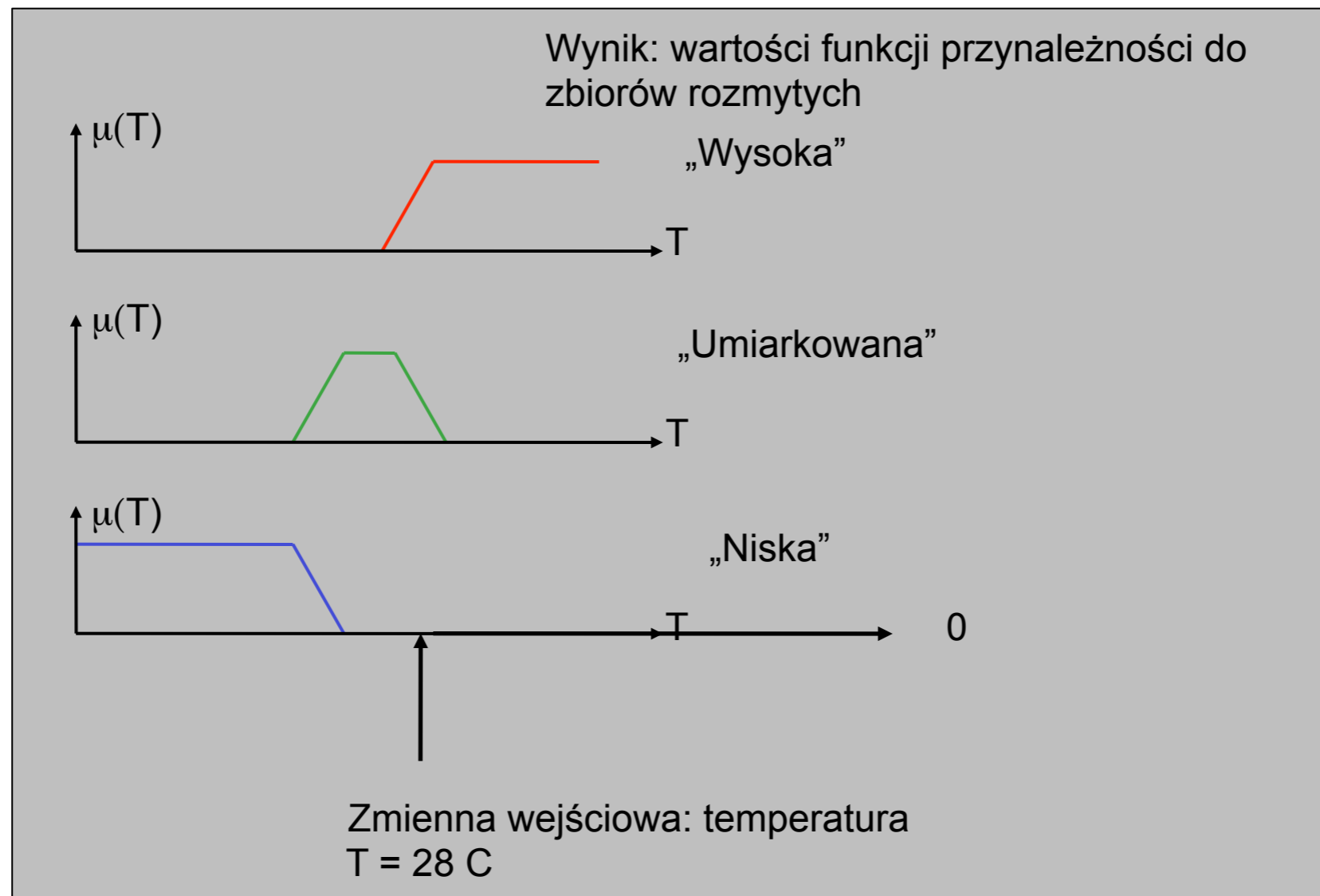
## Jak to działa? Fuzzyfikacja



Dla  $n$  zmiennych wejściowych i  $m$  zbiorów rozmytych zdefiniowanych dla każdej zmiennej otrzymujemy  $n \times m$  wartości funkcji przynależności do tych zbiorów.

# Układy logiki rozmytej

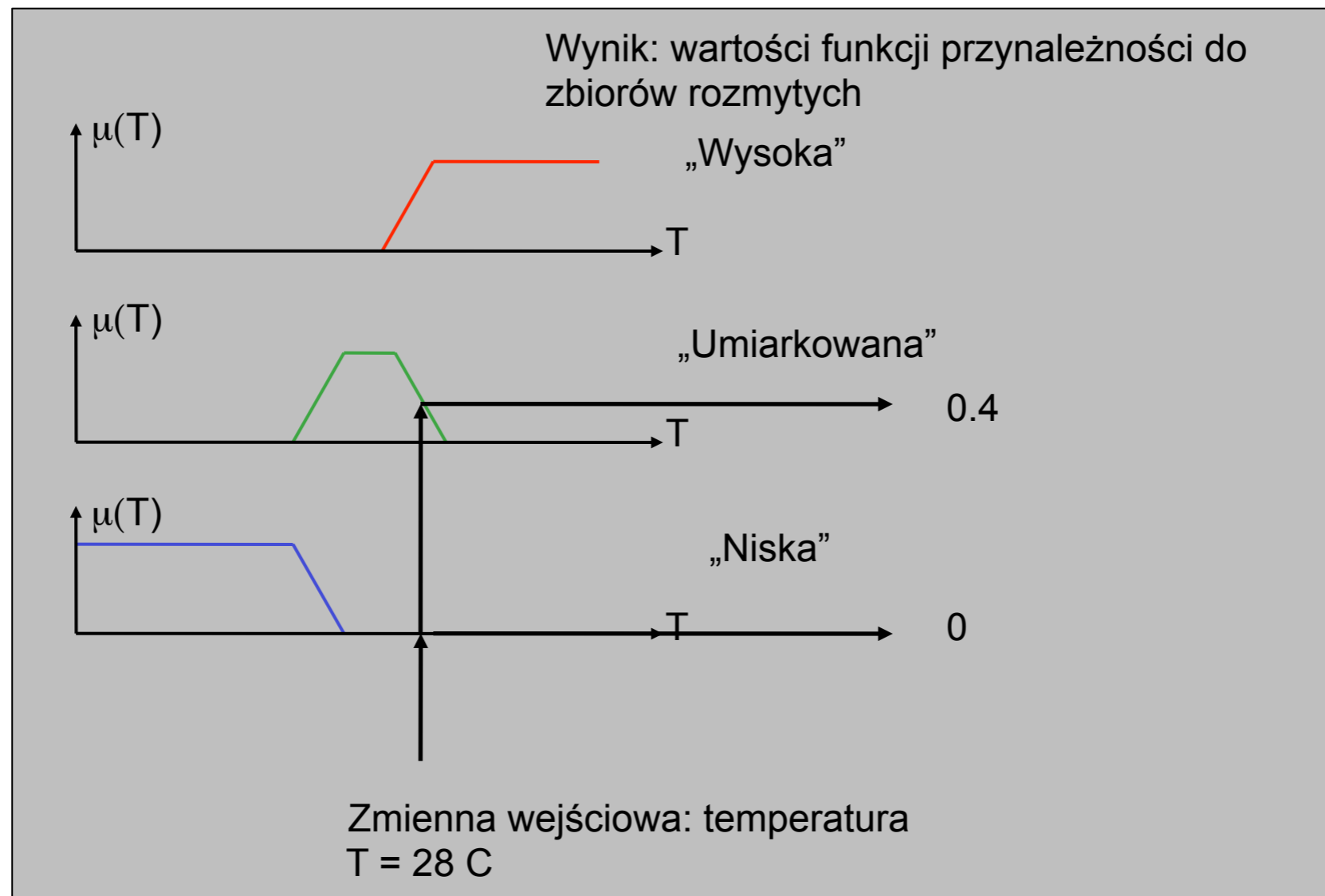
## Jak to działa? Fuzzyfikacja



Dla  $n$  zmiennych wejściowych i  $m$  zbiorów rozmytych zdefiniowanych dla każdej zmiennej otrzymujemy  $n \times m$  wartości funkcji przynależności do tych zbiorów.

# Układy logiki rozmytej

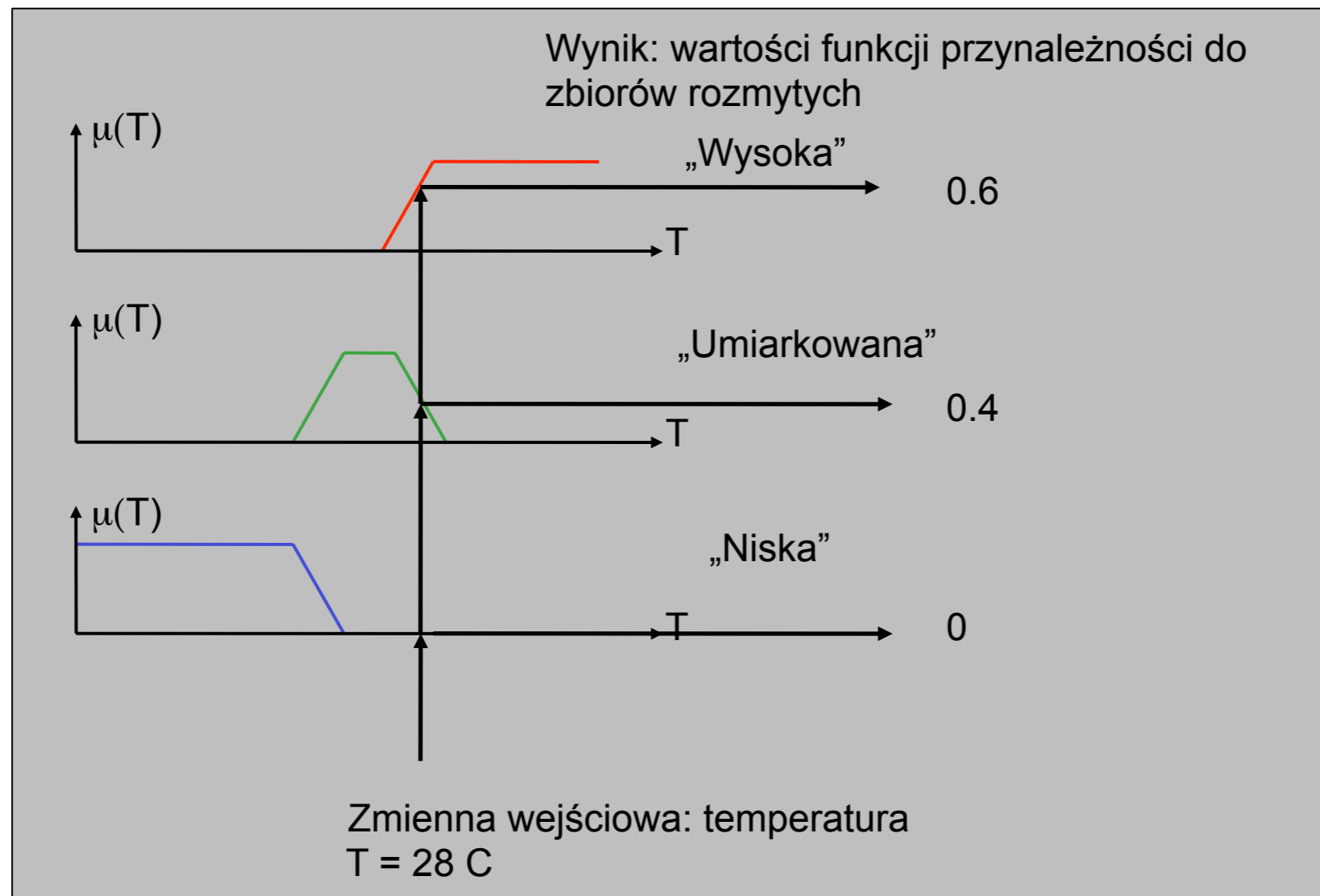
## Jak to działa? Fuzzyfikacja



Dla  $n$  zmiennych wejściowych i  $m$  zbiorów rozmytych zdefiniowanych dla każdej zmiennej otrzymujemy  $n \times m$  wartości funkcji przynależności do tych zbiorów.

# Układy logiki rozmytej

## Jak to działa? Fuzzyfikacja



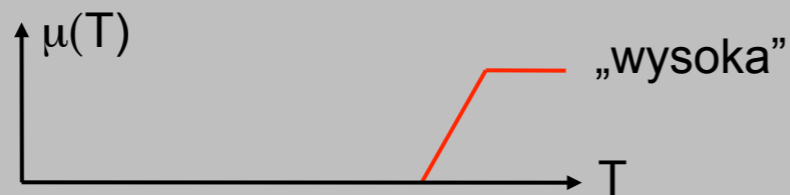
Dla  $n$  zmiennych wejściowych i  $m$  zbiorów rozmytych zdefiniowanych dla każdej zmiennej otrzymujemy  $n \times m$  wartości funkcji przynależności do tych zbiorów.

# Układy logiki rozmytej

Jak to działa? Wnioskowanie

## Reguła typu "I"

**JEŚLI** (temperatura **JEST** wysoka) **I** (wilgotność **JEST** niska) **TO** (nawiew ciepłego powietrza **JEST** bardzo mały)



temperatura:  
 $T = 28 \text{ C}$



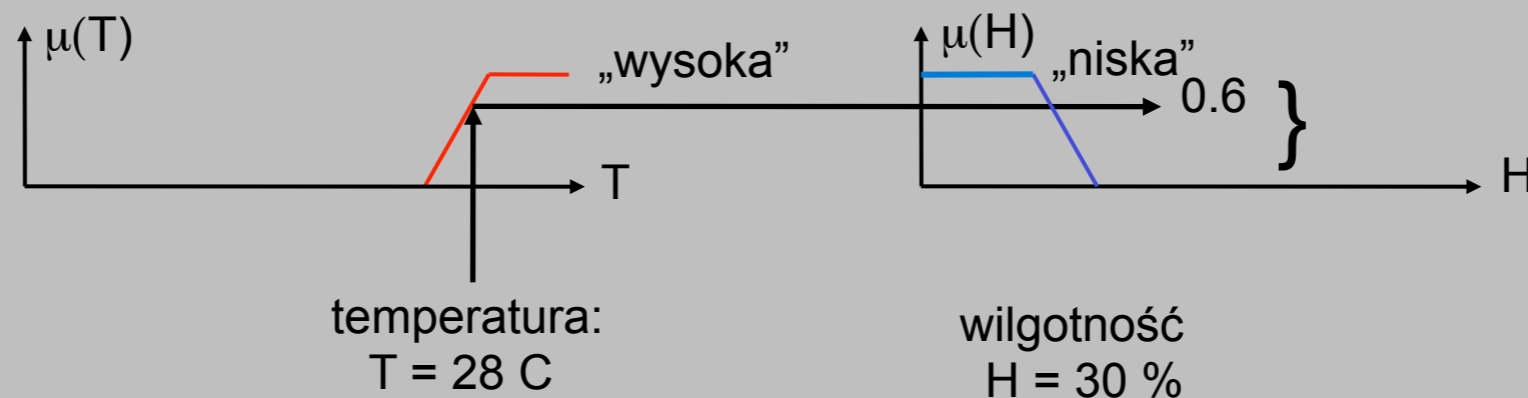
wilgotność  
 $H = 30 \%$

# Układy logiki rozmytej

Jak to działa? Wnioskowanie

## Reguła typu "I"

**JEŚLI** (temperatura **JEST** wysoka) **I** (wilgotność **JEST** niska) **TO** (nawiew ciepłego powietrza **JEST** bardzo mały)

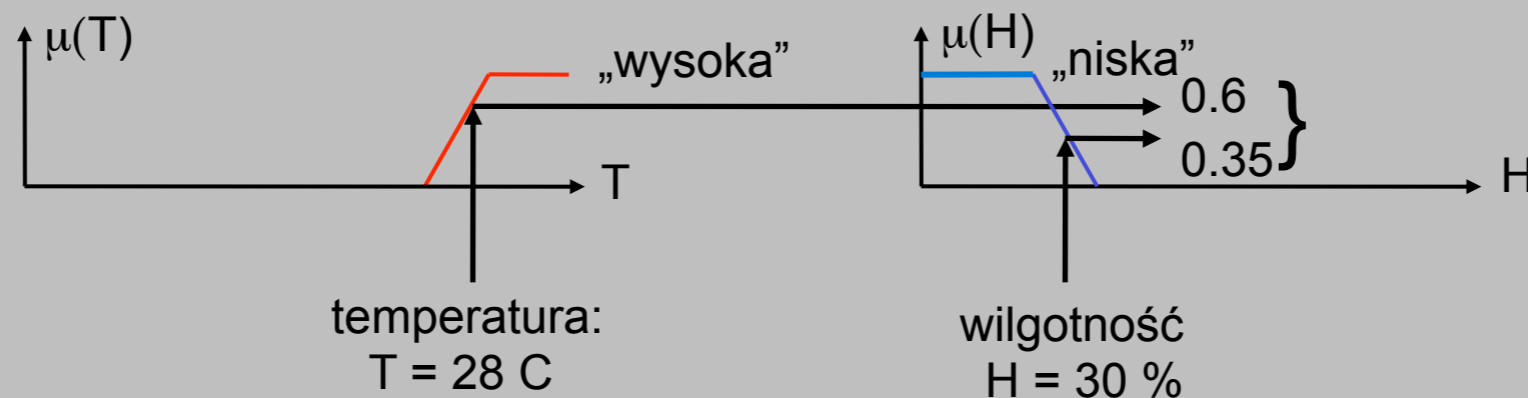


# Układy logiki rozmytej

Jak to działa? Wnioskowanie

## Reguła typu "I"

**JEŚLI** (temperatura **JEST** wysoka) **I** (wilgotność **JEST** niska) **TO** (nawiew ciepłego powietrza **JEST** bardzo mały)



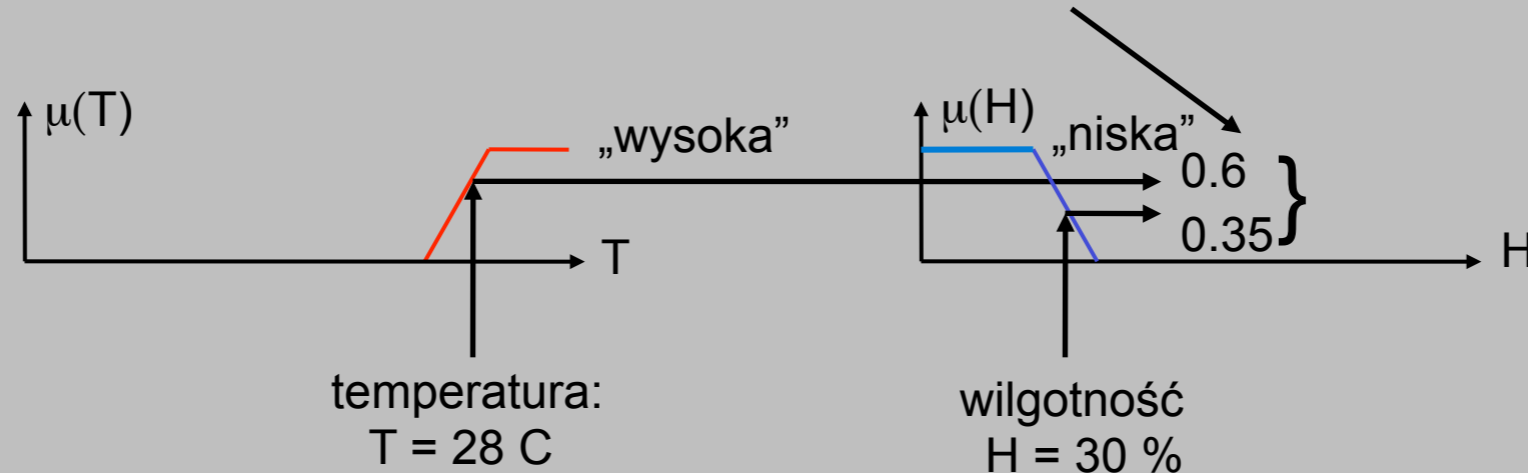
# Układy logiki rozmytej

Jak to działa? Wnioskowanie

## Reguła typu "I"

**JEŚLI** (temperatura **JEST** wysoka) **I** (wilgotność **JEST** niska) **TO** (nawiew ciepłego powietrza **JEST** bardzo mały)

**Reguła "I"** oznacza: wybieramy **mniejszą** wartość





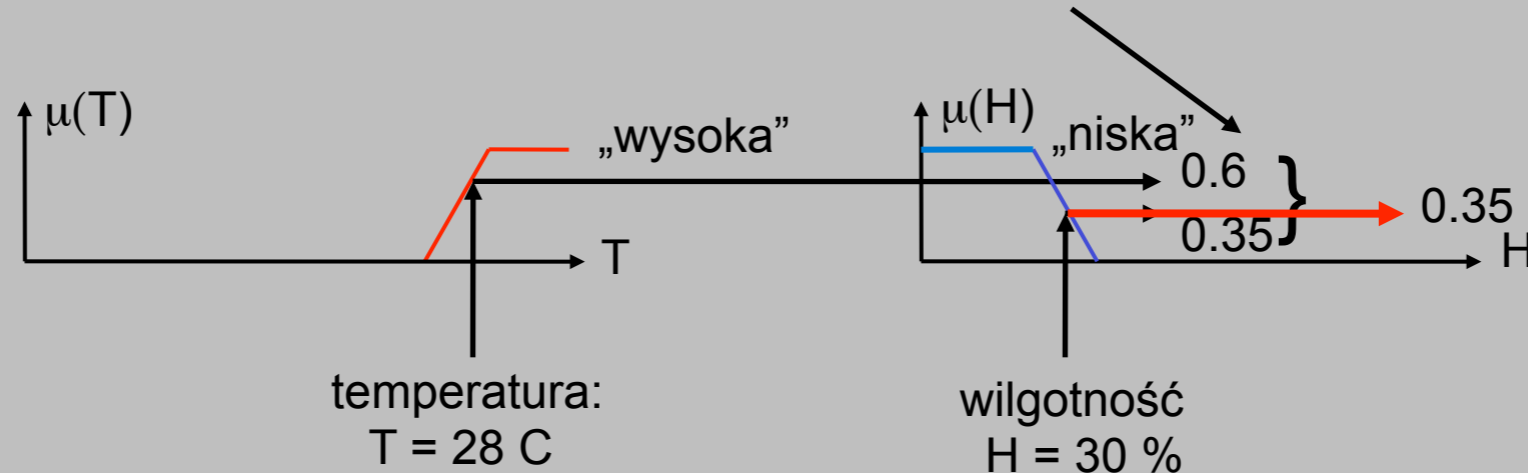
# Układy logiki rozmytej

Jak to działa? Wnioskowanie

## Reguła typu "I"

**JEŚLI** (temperatura **JEST** wysoka) **I** (wilgotność **JEST** niska) **TO** (nawiew ciepłego powietrza **JEST** bardzo mały)

**Reguła "I"** oznacza: wybieramy **mniejszą** wartość



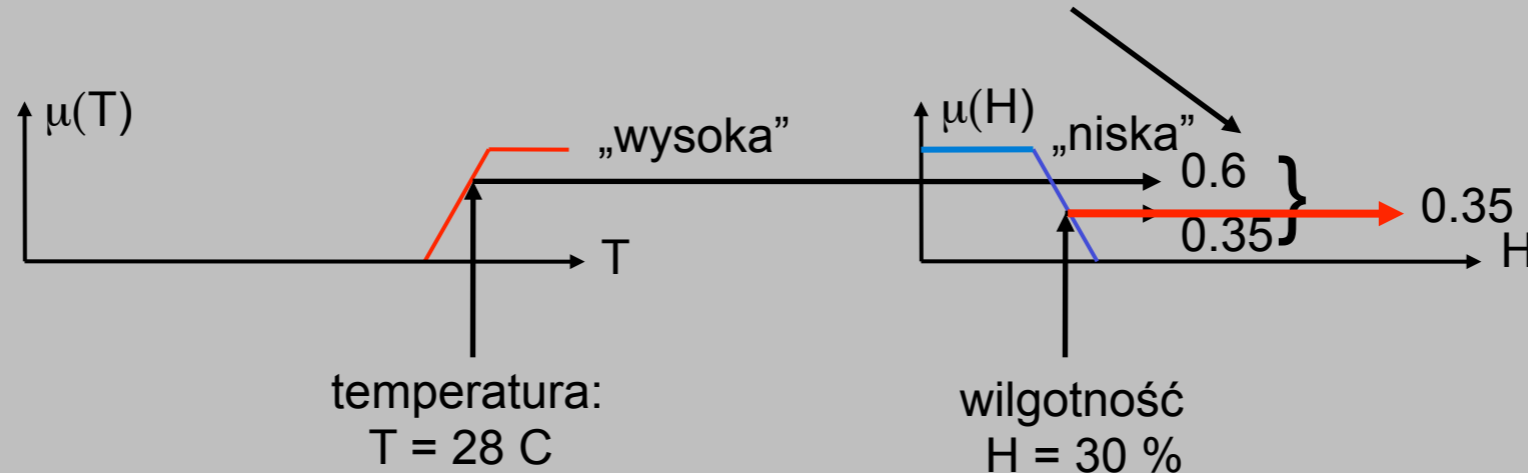
# Układy logiki rozmytej

Jak to działa? Wnioskowanie

## Reguła typu "I"

**JEŚLI** (temperatura **JEST** wysoka) **I** (wilgotność **JEST** niska) **TO** (nawiew ciepłego powietrza **JEST** bardzo mały)

**Reguła "I"** oznacza: wybieramy **mniejszą** wartość



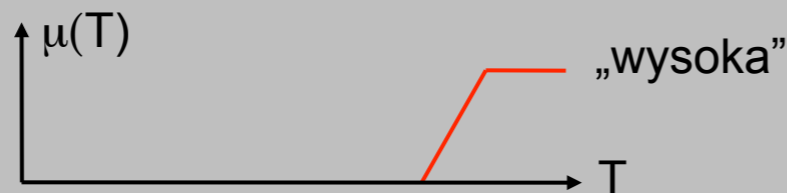
**Wynik:** zgodnie z tą regułą nawiew ciepłego powietrza powinien należeć do zbioru nawiewów bardzo małych, a waga tego wyniku wynosi 0,35.

# Układy logiki rozmytej

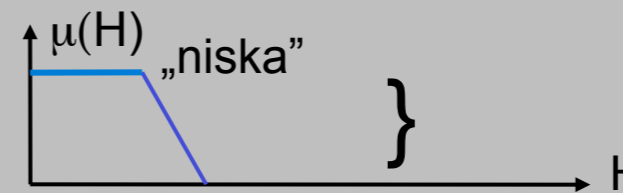
Jak to działa? Wnioskowanie

Reguła typu „LUB”

**JEŚLI** (temperatura **JEST** wysoka) **LUB** (wilgotność **JEST** niska) **TO**  
(nawiew ciepłego powietrza **JEST** bardzo mały)



temperatura:  
 $T = 28 \text{ C}$



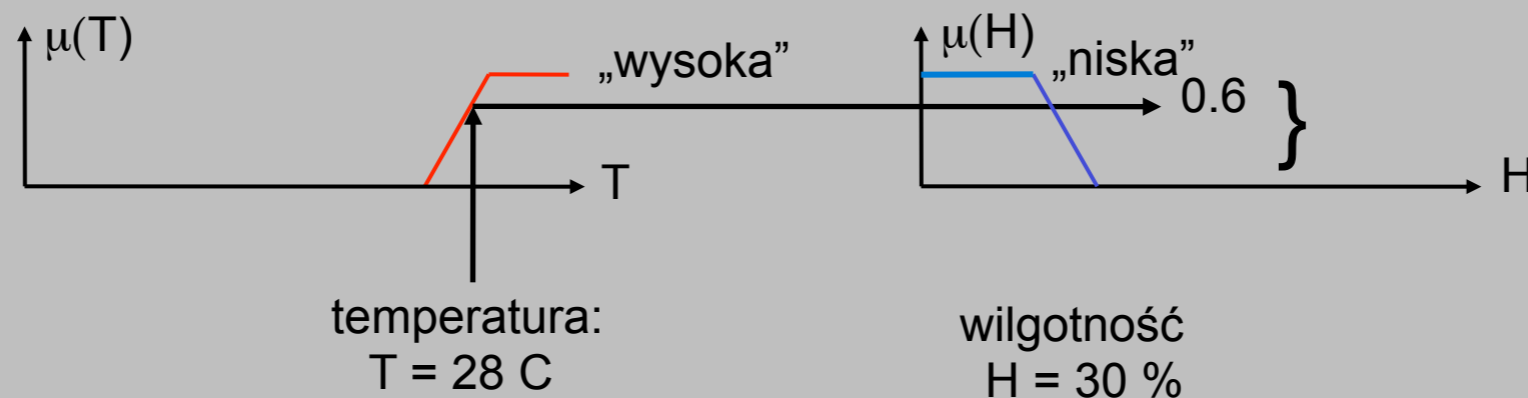
wilgotność  
 $H = 30 \%$

# Układy logiki rozmytej

Jak to działa? Wnioskowanie

Reguła typu „LUB”

**JEŚLI** (temperatura **JEST** wysoka) **LUB** (wilgotność **JEST** niska) **TO**  
(nawiew ciepłego powietrza **JEST** bardzo mały)

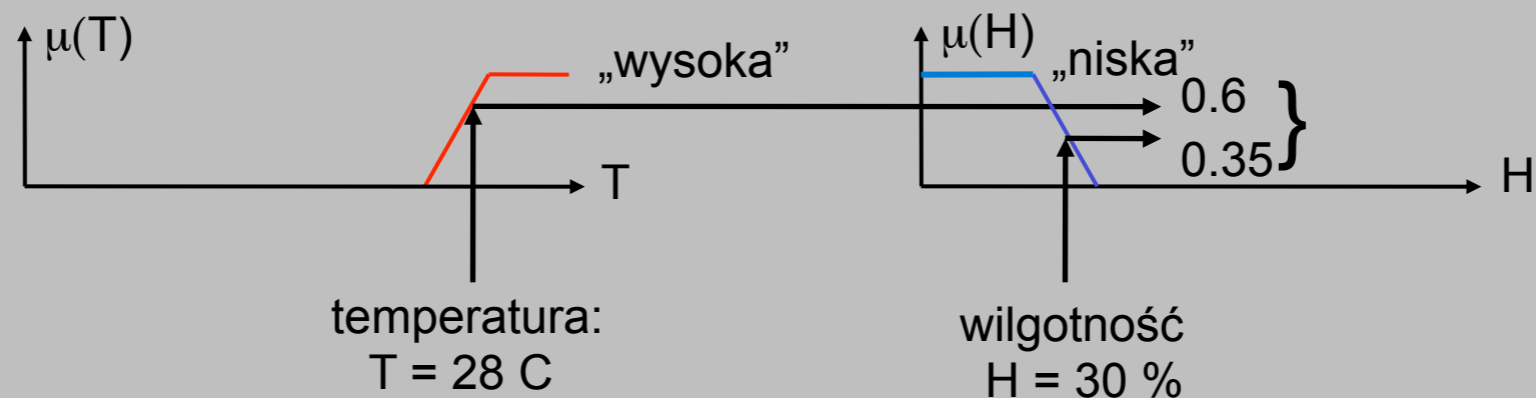


# Układy logiki rozmytej

Jak to działa? Wnioskowanie

Reguła typu „LUB”

**JEŚLI** (temperatura **JEST** wysoka) **LUB** (wilgotność **JEST** niska) **TO**  
(nawiew ciepłego powietrza **JEST** bardzo mały)



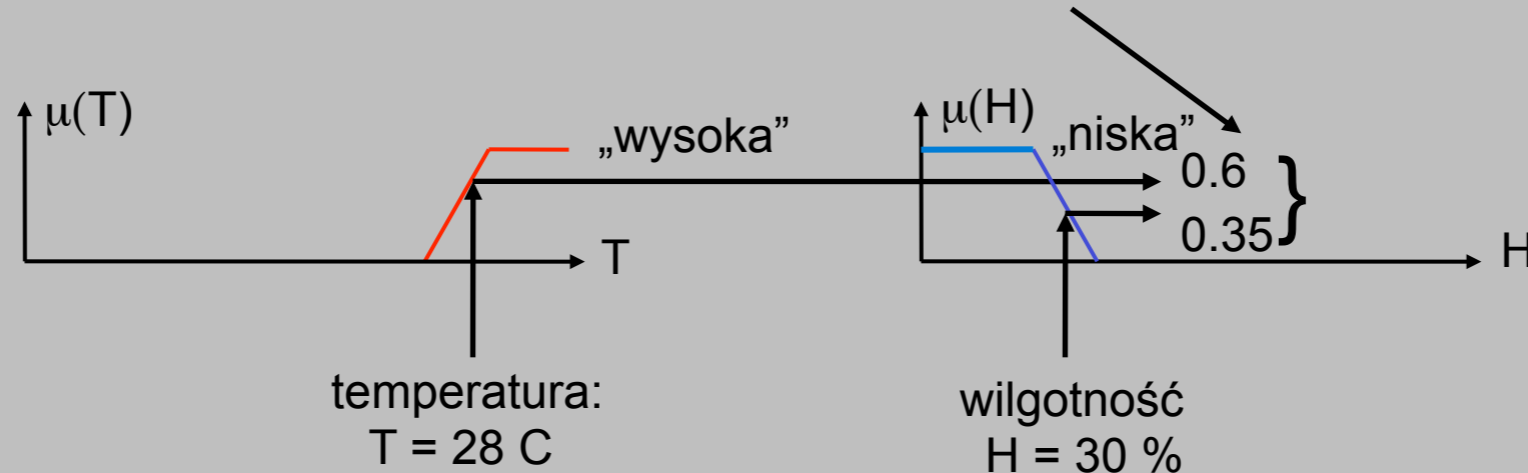
# Układy logiki rozmytej

Jak to działa? Wnioskowanie

## Reguła typu "LUB"

**JEŚLI** (temperatura **JEST** wysoka) **LUB** (wilgotność **JEST** niska) **TO**  
(nawiew ciepłego powietrza **JEST** bardzo mały)

**Reguła "LUB"** oznacza: wybieramy **większą** wartość



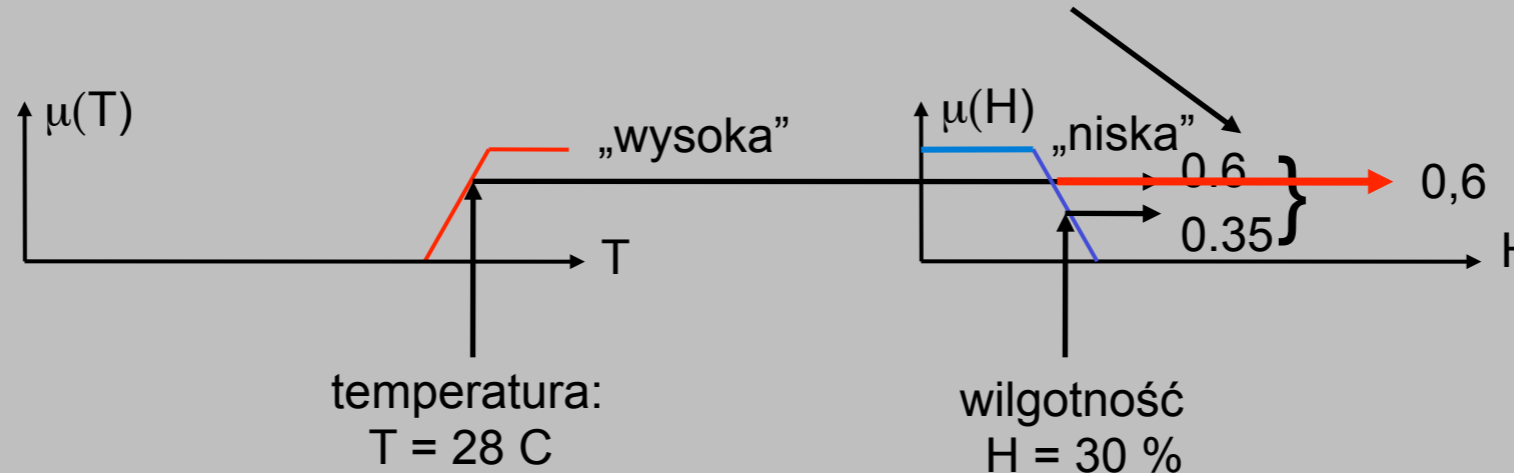
# Układy logiki rozmytej

Jak to działa? Wnioskowanie

Reguła typu „LUB”

**JEŚLI** (temperatura **JEST** wysoka) **LUB** (wilgotność **JEST** niska) **TO**  
(nawiew ciepłego powietrza **JEST** bardzo mały)

Reguła „LUB” oznacza: wybieramy **większą** wartość



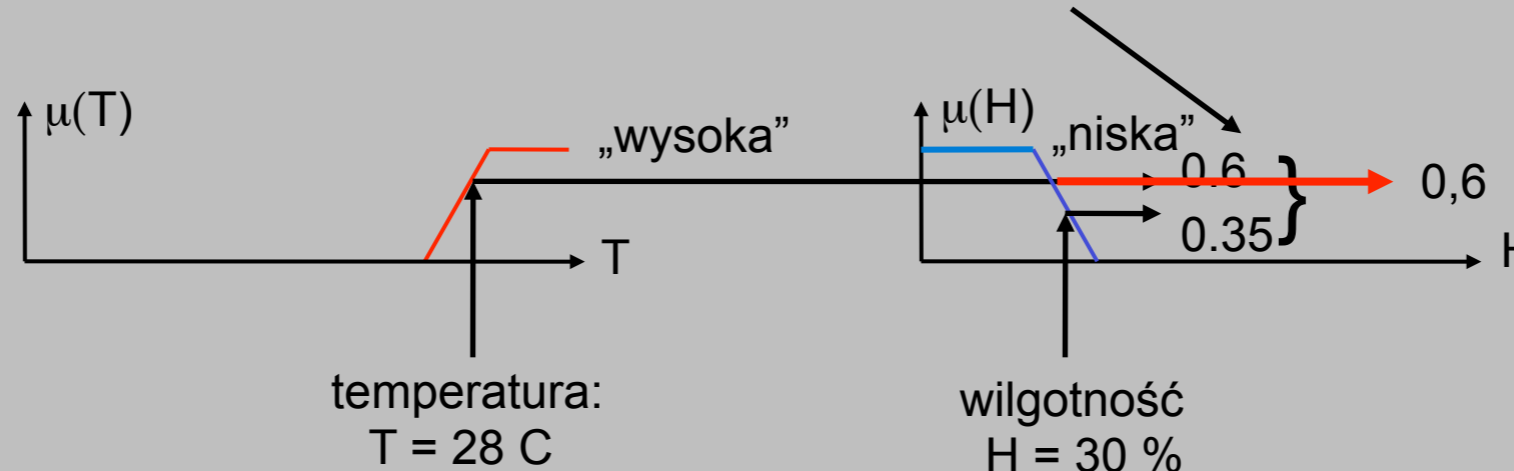
# Układy logiki rozmytej

Jak to działa? Wnioskowanie

## Reguła typu "LUB"

**JEŚLI** (temperatura **JEST** wysoka) **LUB** (wilgotność **JEST** niska) **TO**  
(nawiew ciepłego powietrza **JEST** bardzo mały)

**Reguła "LUB"** oznacza: wybieramy **większą** wartość



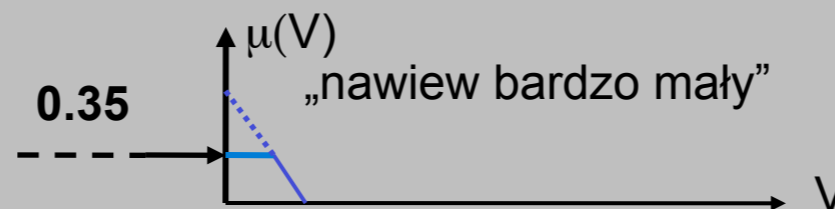
**Wynik: zgodnie z tą regułą nawiew ciepłego powietrza powinien należeć do zbioru nawiewów bardzo małych, a waga tego wyniku wynosi 0,6.**



# Układy logiki rozmytej

Jak to działa? Łączenie reguł

**Interpretacja wyniku działania reguły: funkcja przynależności do wyjściowego zbioru rozmytego jest „ucięta”**

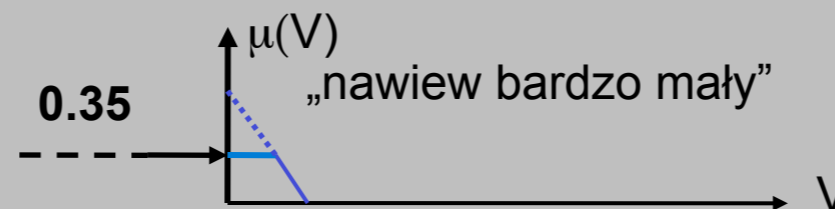


**Jeśli kilka reguł oddziałuje na tę samą zmienną wyjściową, odpowiednie zbiory rozmyte łączy się przy pomocy operatora LUB.**

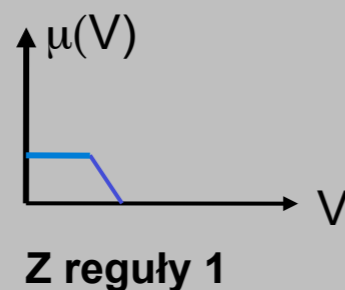
# Układy logiki rozmytej

Jak to działa? Łączenie reguł

Interpretacja wyniku działania reguły: funkcja przynależności do wyjściowego zbioru rozmytego jest „ucięta”



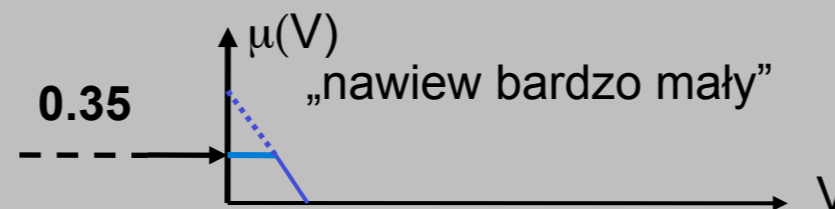
Jeśli kilka reguł oddziałuje na tę samą zmienną wyjściową, odpowiednie zbiory rozmyte łączy się przy pomocy operatora LUB.



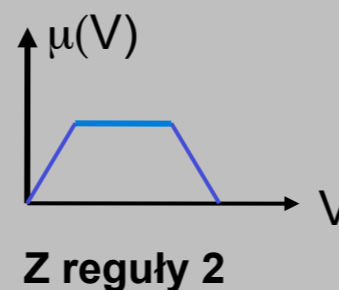
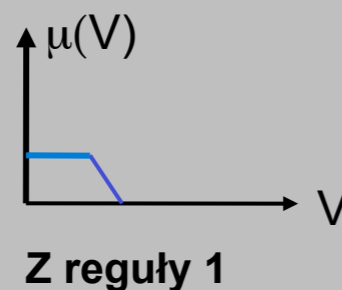
# Układy logiki rozmytej

## Jak to działa? Łączenie reguł

Interpretacja wyniku działania reguły: funkcja przynależności do wyjściowego zbioru rozmytego jest „ucięta”



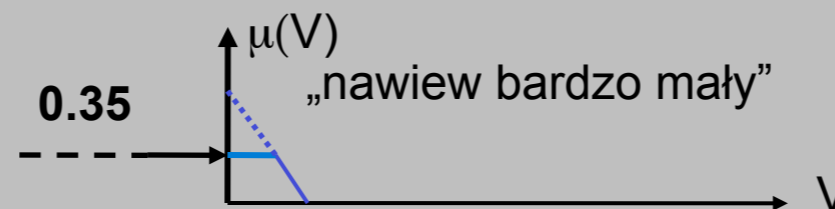
Jeśli kilka reguł oddziałuje na tę samą zmienną wyjściową, odpowiednie zbiory rozmyte łączą się przy pomocy operatora LUB.



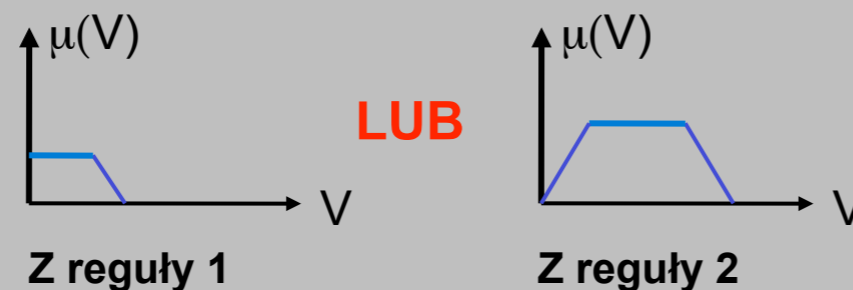
# Układy logiki rozmytej

Jak to działa? Łączenie reguł

Interpretacja wyniku działania reguły: funkcja przynależności do wyjściowego zbioru rozmytego jest „ucięta”



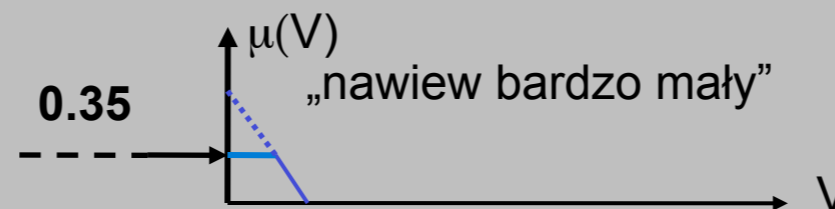
Jeśli kilka reguł oddziałuje na tę samą zmienną wyjściową, odpowiednie zbiory rozmyte łączy się przy pomocy operatora LUB.



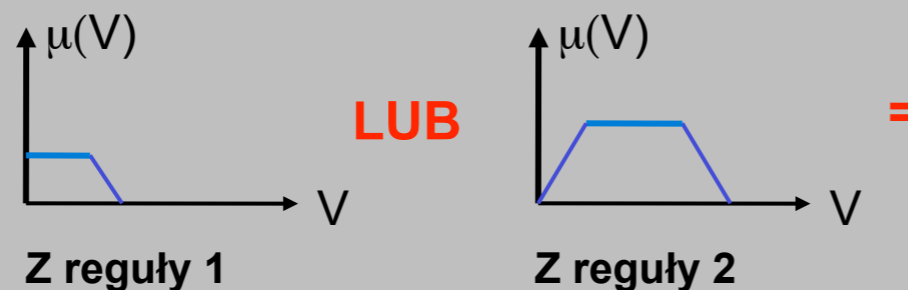
# Układy logiki rozmytej

Jak to działa? Łączenie reguł

Interpretacja wyniku działania reguły: funkcja przynależności do wyjściowego zbioru rozmytego jest „ucięta”



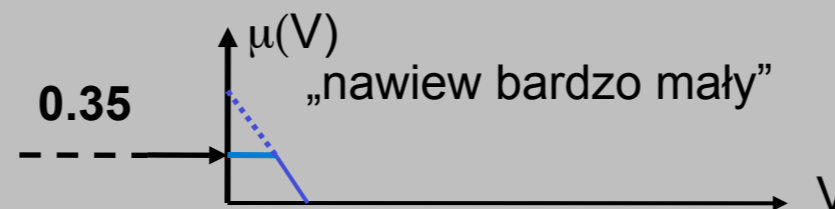
Jeśli kilka reguł oddziałuje na tę samą zmienną wyjściową, odpowiednie zbiory rozmyte łączy się przy pomocy operatora LUB.



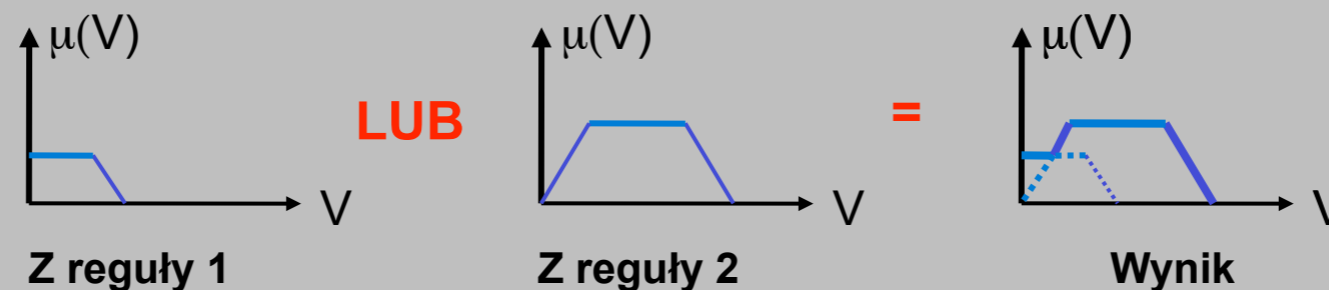
# Układy logiki rozmytej

Jak to działa? Łączenie reguł

Interpretacja wyniku działania reguły: funkcja przynależności do wyjściowego zbioru rozmytego jest „ucięta”



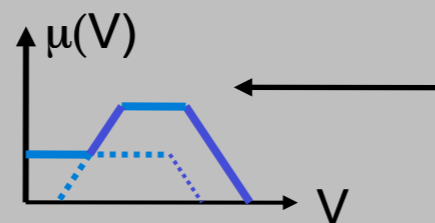
Jeśli kilka reguł oddziałuje na tę samą zmienną wyjściową, odpowiednie zbiory rozmyte łączy się przy pomocy operatora LUB.



# Układy logiki rozmytej

## Jak to działa? Defuzyfikacja

Defuzyfikacja oznacza określenie wartości zmiennej wyjściowej na podstawie kształtu wyjściowego zbioru rozmytego. Można to wykonać na kilka sposobów, jednym z często stosowanych jest wyznaczenie “środka ciężkości” funkcji przynależności.



**Wyjściowy zbiór rozmyty dla zmiennej  
“nawiew ciepłego powietrza”**

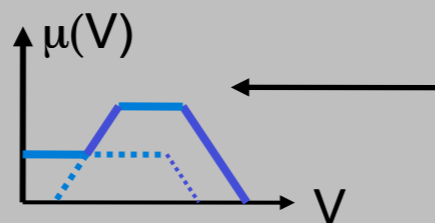
# Układy logiki rozmytej

## Jak to działa? Defuzyfikacja

Defuzyfikacja oznacza określenie wartości zmiennej wyjściowej na podstawie kształtu wyjściowego zbioru rozmytego. Można to wykonać na kilka sposobów, jednym z często stosowanych jest wyznaczenie “środka ciężkości” funkcji przynależności.

Dla zmiennej  $V$ , której funkcja przynależności do zbioru wyjściowego jest  $\mu(V)$ , wyjściowa wartość jest wyznaczana ze wzoru

$$V_{wy} = \frac{\int_0^{\infty} V \mu(V) dV}{\int_0^{\infty} \mu(V) dV}$$



Wyjściowy zbiór rozmyty dla zmiennej “nawiew ciepłego powietrza”



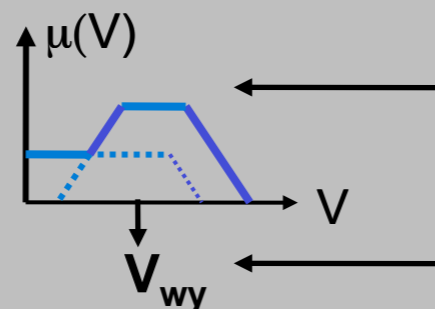
# Układy logiki rozmytej

## Jak to działa? Defuzyfikacja

Defuzyfikacja oznacza określenie wartości zmiennej wyjściowej na podstawie kształtu wyjściowego zbioru rozmytego. Można to wykonać na kilka sposobów, jednym z często stosowanych jest wyznaczenie “środka ciężkości” funkcji przynależności.

Dla zmiennej  $V$ , której funkcja przynależności do zbioru wyjściowego jest  $\mu(V)$ , wyjściowa wartość jest wyznaczana ze wzoru

$$V_{wy} = \frac{\int_0^{\infty} V \mu(V) dV}{\int_0^{\infty} \mu(V) dV}$$



**Wyjściowy zbiór rozmyty dla zmiennej  
“nawiew ciepłego powietrza”**

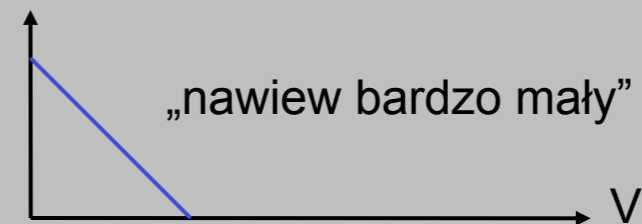
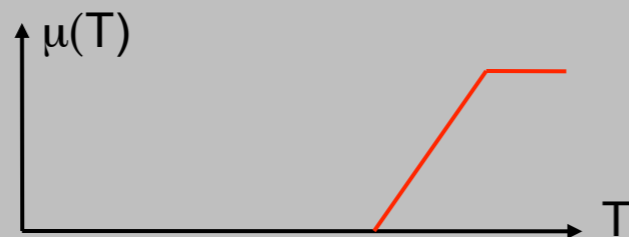
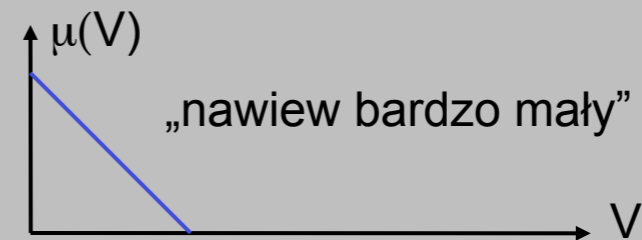
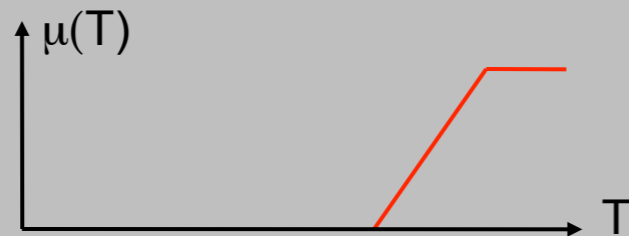
**Wyjściowa wartość zmiennej  
“nawiew ciepłego powietrza”**

# Układy logiki rozmytej

Jak to działa? Defuzyfikacja

Uproszczony przykład: jedna zmienna wejściowa, jedna zmienna wyjściowa i jedna reguła

**JEŚLI** (temperatura **JEST** wysoka) **TO**  
(nawiew ciepłego powietrza **JEST** bardzo mały)

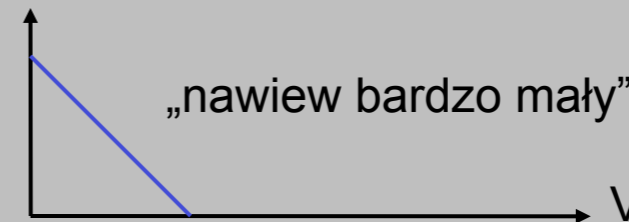
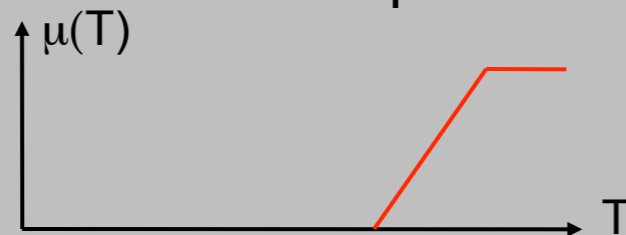
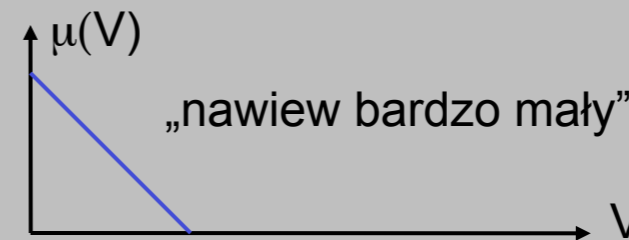
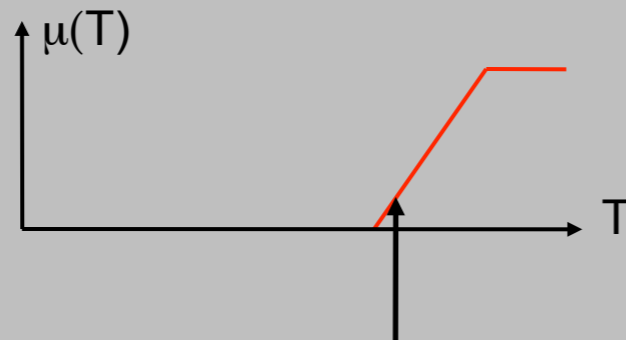


# Układy logiki rozmytej

Jak to działa? Defuzyfikacja

Uproszczony przykład: jedna zmienna wejściowa, jedna zmienna wyjściowa i jedna reguła

**JEŚLI** (temperatura **JEST** wysoka) **TO**  
(nawiew ciepłego powietrza **JEST** bardzo mały)

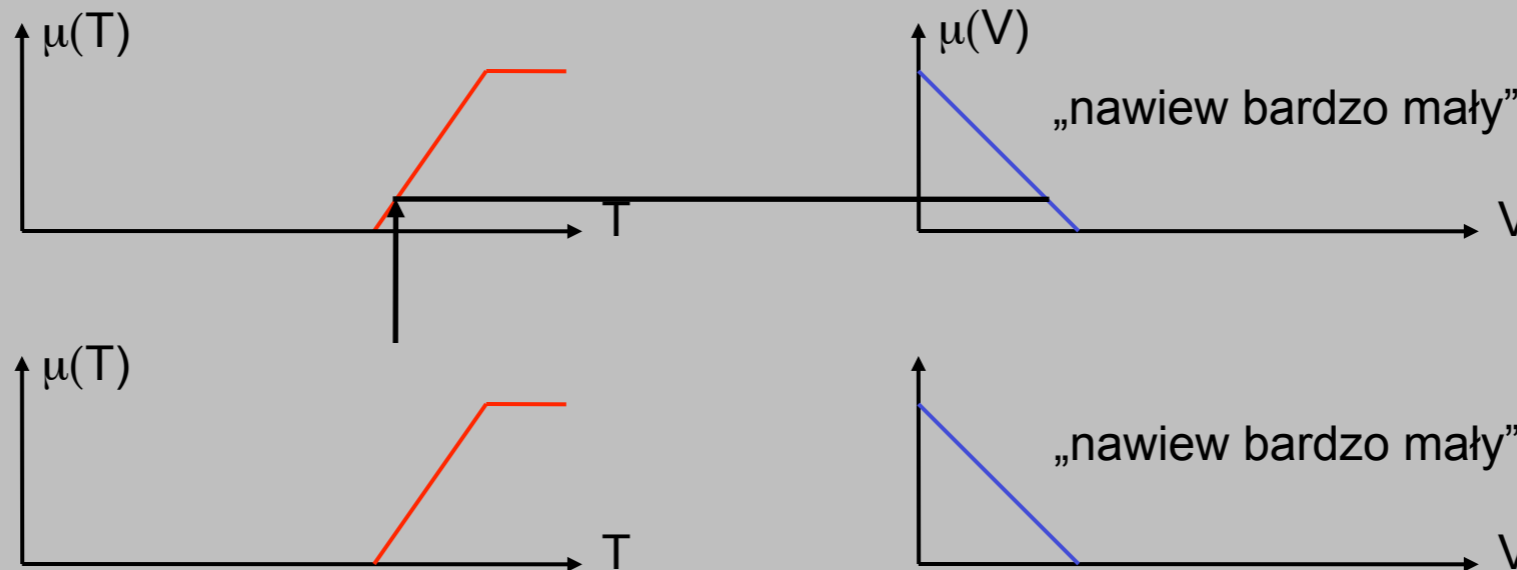


# Układy logiki rozmytej

Jak to działa? Defuzyfikacja

Uproszczony przykład: jedna zmienna wejściowa, jedna zmienna wyjściowa i jedna reguła

**JEŚLI** (temperatura **JEST** wysoka) **TO**  
(nawiew ciepłego powietrza **JEST** bardzo mały)

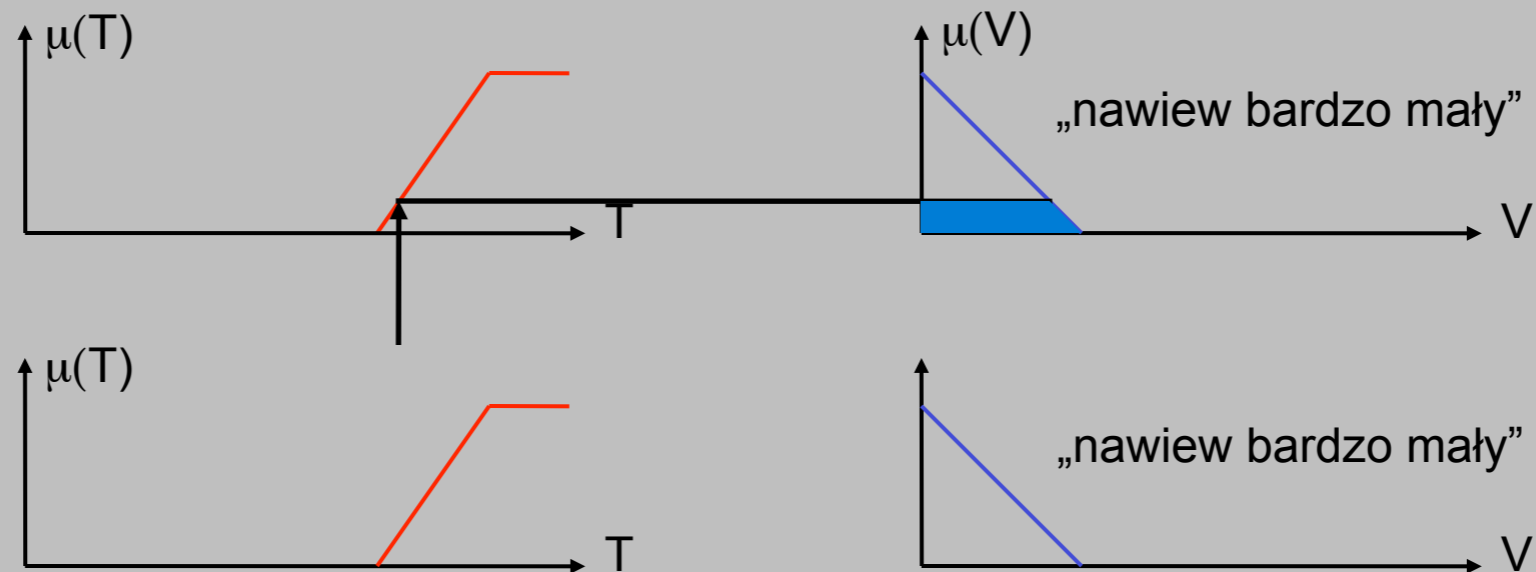


# Układy logiki rozmytej

Jak to działa? Defuzyfikacja

Uproszczony przykład: jedna zmienna wejściowa, jedna zmienna wyjściowa i jedna reguła

**JEŚLI** (temperatura **JEST** wysoka) **TO**  
(nawiew ciepłego powietrza **JEST** bardzo mały)

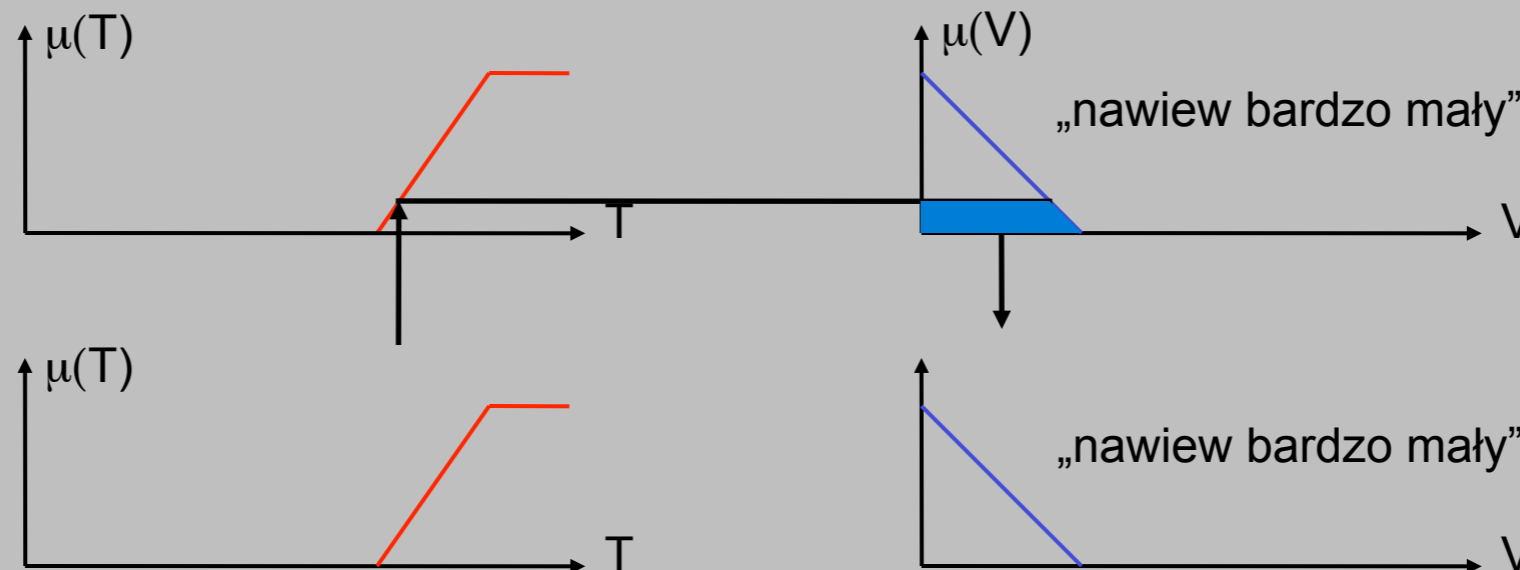


# Układy logiki rozmytej

Jak to działa? Defuzyfikacja

Uproszczony przykład: jedna zmienna wejściowa, jedna zmienna wyjściowa i jedna reguła

**JEŚLI** (temperatura **JEST** wysoka) **TO**  
(nawiew ciepłego powietrza **JEST** bardzo mały)

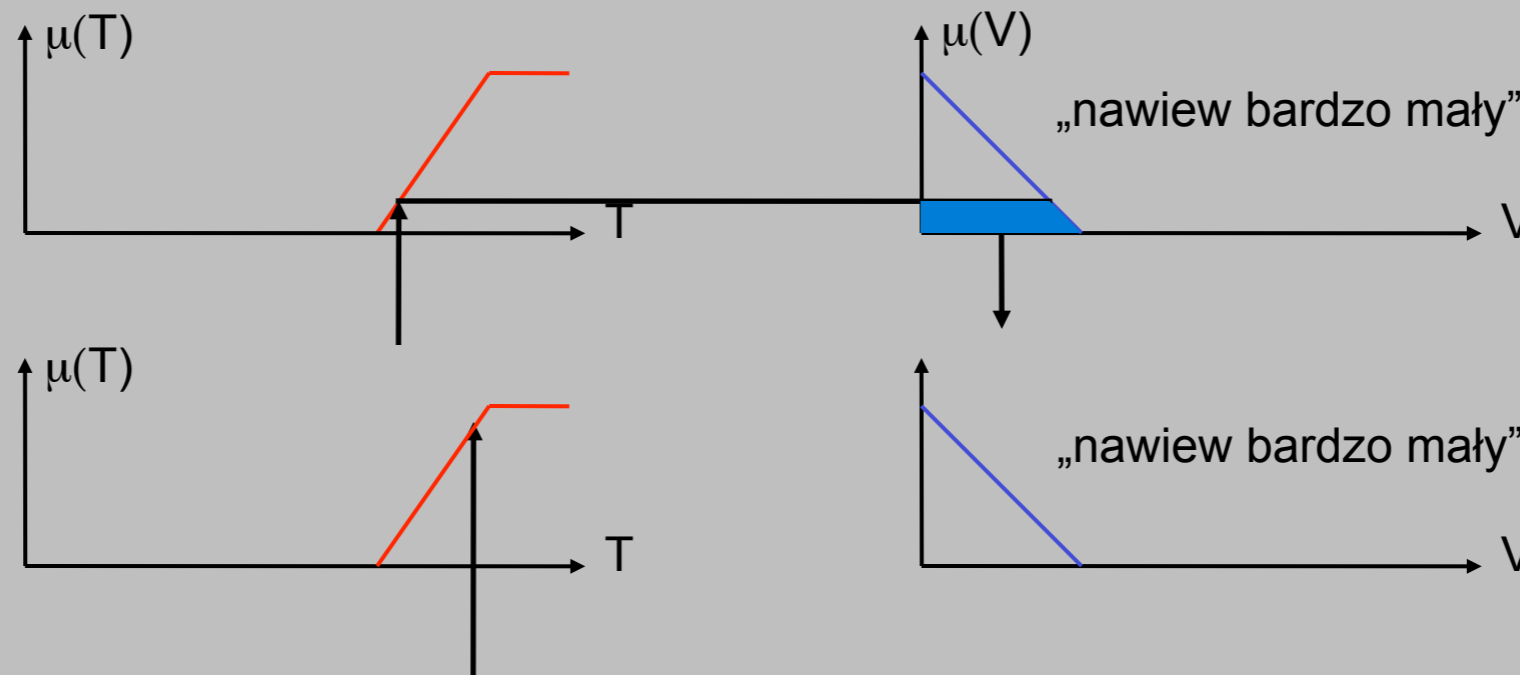


# Układy logiki rozmytej

Jak to działa? Defuzyfikacja

Uproszczony przykład: jedna zmienna wejściowa, jedna zmienna wyjściowa i jedna reguła

**JEŚLI** (temperatura **JEST** wysoka) **TO**  
(nawiew ciepłego powietrza **JEST** bardzo mały)

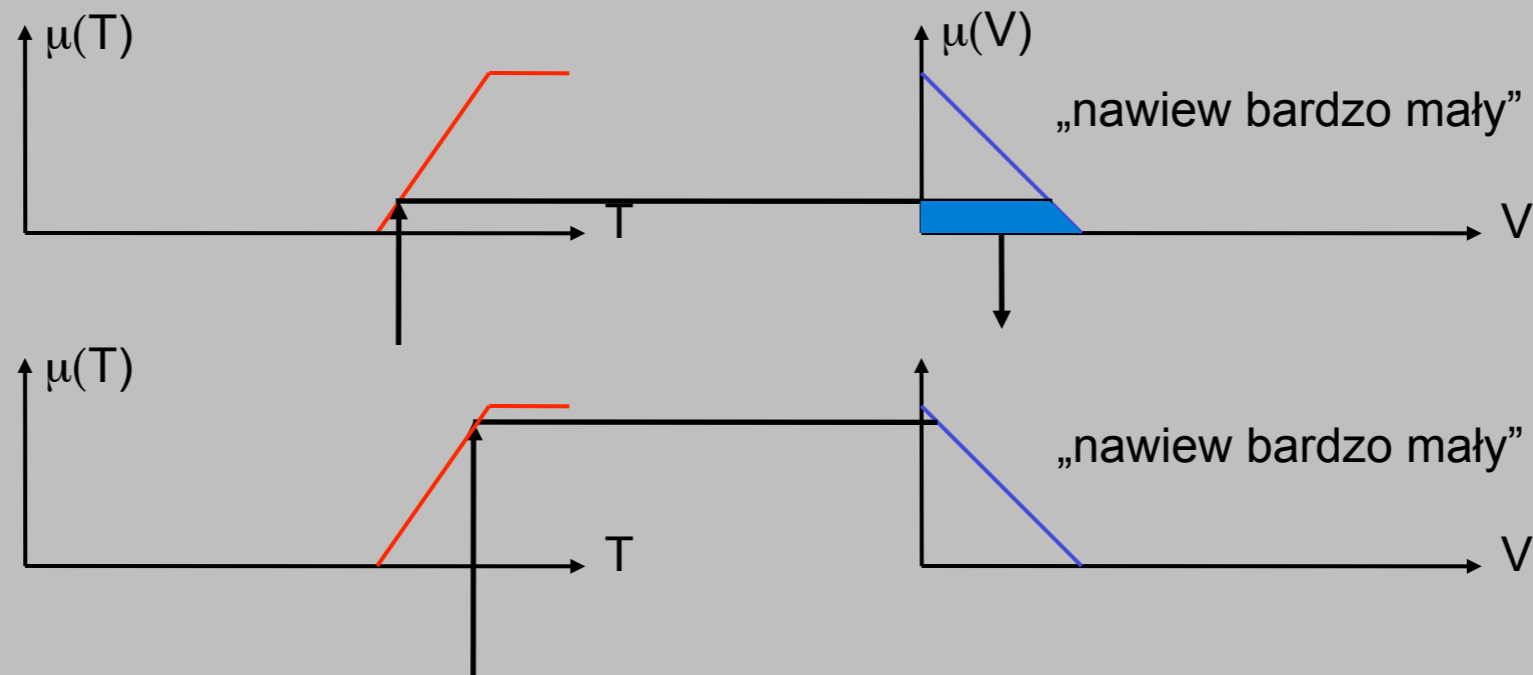


# Układy logiki rozmytej

Jak to działa? Defuzyfikacja

Uproszczony przykład: jedna zmienna wejściowa, jedna zmienna wyjściowa i jedna reguła

**JEŚLI** (temperatura **JEST** wysoka) **TO**  
(nawiew ciepłego powietrza **JEST** bardzo mały)



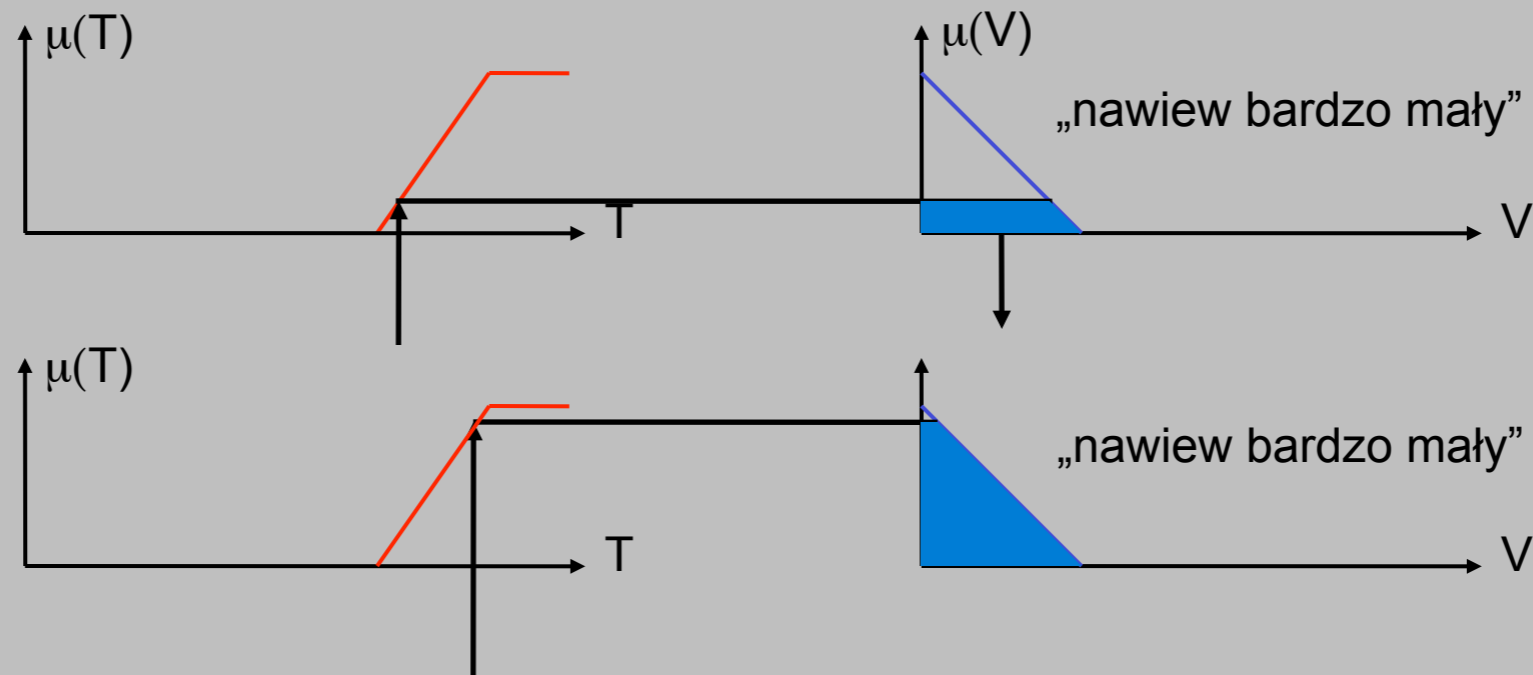


# Układy logiki rozmytej

Jak to działa? Defuzyfikacja

Uproszczony przykład: jedna zmienna wejściowa, jedna zmienna wyjściowa i jedna reguła

**JEŚLI** (temperatura **JEST** wysoka) **TO**  
(nawiew ciepłego powietrza **JEST** bardzo mały)

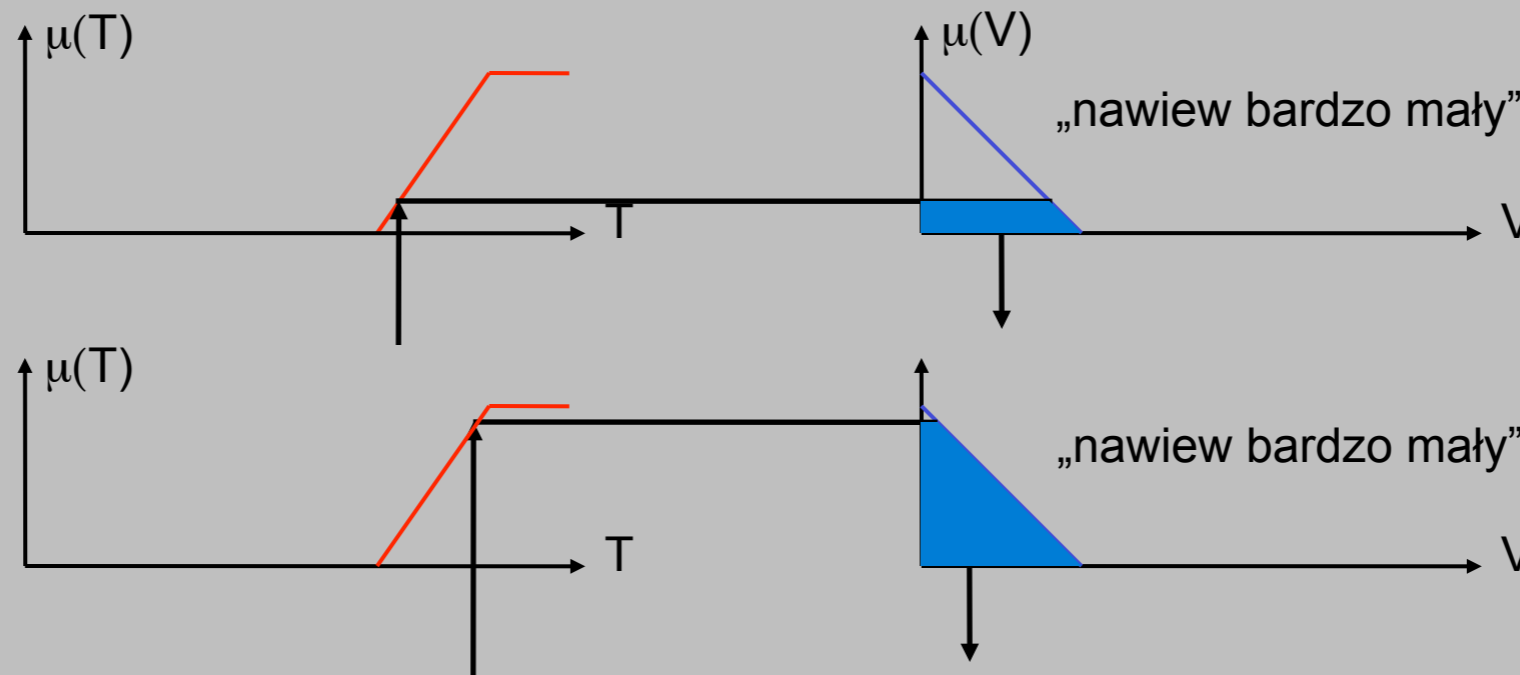


# Układy logiki rozmytej

Jak to działa? Defuzyfikacja

Uproszczony przykład: jedna zmienna wejściowa, jedna zmienna wyjściowa i jedna reguła

**JEŚLI** (temperatura **JEST** wysoka) **TO**  
(nawiew ciepłego powietrza **JEST** bardzo mały)



Wyższa temperatura -> słabszy nawiew

# Układy logiki rozmytej

## Uogólnienia

Operatory logiczne oraz wzory określające sposób defuzyfikacji mogą być definiowane na wiele sposobów.

# Układy logiki rozmytej

## Uogólnienia

Operatory logiczne oraz wzory określające sposób defuzyfikacji mogą być definiowane na wiele sposobów.

Operator logiczny LUB, reprezentowany w przykładach przez funkcję *max*, jest w ogólności określany jako operator S-normy, musi on spełniać następujące warunki:

$$S(1, 1) = 1$$

$$S(x, 0) = S(0, x) = x$$

$$S(x, y) \leq S(u, v) \text{ if } x < u \text{ and } y < v$$

$$S(x, y) = S(y, x)$$

$$S(x, S(u, v)) = (S(S(x, u), v)$$

# Układy logiki rozmytej

Uogólnienia

# Układy logiki rozmytej

## Uogólnienia

Operator logiczny  $I$ , reprezentowany w przykładach przez funkcję *min*, jest w ogólności określany jako operator T-normy, musi on spełniać następujące warunki:

$$T(0, 0) = 0$$

$$T(x, 1) = T(1, x) = x$$

$$T(x, y) \leq T(u, v) \text{ if } x < u \text{ and } y < v$$

$$T(x, y) = T(y, x)$$

$$T(x, T(u, v)) = (T(T(x, u), v))$$

# Układy logiki rozmytej

## Uogólnienia

We wnioskowaniu można użyć *dowolnego* sposobu zdefiniowania operatorów S-normy i T-normy, jeśli spełniają one podane wcześniej warunki.

# Układy logiki rozmytej

## Uogólnienia

We wnioskowaniu można użyć *dowolnego* sposobu zdefiniowania operatorów S-normy i T-normy, jeśli spełniają one podane wcześniej warunki.

Przykład S-normy i T-normy:



# Układy logiki rozmytej

## Uogólnienia

We wnioskowaniu można użyć *dowolnego* sposobu zdefiniowania operatorów S-normy i T-normy, jeśli spełniają one podane wcześniej warunki.

Przykład S-normy i T-normy:

(oba argumenty ograniczone do przedziału  $[0, 1]$ ):

T-norma:  $T(x, y) = xy$  (zwykłe mnożenie)

S-norma:  $S(x, y) = (x + y - xy)$

(S-norma *nie może być* zdefiniowana przez zwykłe dodawanie, bo  $1 + 1 \neq 1$  !)

# Układy logiki rozmytej

## Uogólnienia

We wnioskowaniu można użyć *dowolnego* sposobu zdefiniowania operatorów S-normy i T-normy, jeśli spełniają one podane wcześniej warunki.

Przykład S-normy i T-normy:

(oba argumenty ograniczone do przedziału  $[0, 1]$ ):

T-norma:  $T(x, y) = xy$  (zwykłe mnożenie)

S-norma:  $S(x, y) = (x + y - xy)$

(S-norma *nie może być* zdefiniowana przez zwykłe dodawanie, bo  $1 + 1 \neq 1$  !)

Uwaga: chociaż zbiory rozmyte są uogólnieniem zbiorów teorii mnogości, to reguły algebry Boole'a w ogólności *nie są* słuszne w teorii zbiorów rozmytych!

# Układy logiki rozmytej

Jak tego użyć w praktyce?

# Układy logiki rozmytej

Jak tego użyć w praktyce?

Wybór liczby i kształtu zbiorów rozmytych dla zmiennych wejściowych, oraz reguł wnioskowania zależy od problemu, jaki rozwiązujemy. Trafność tych wyborów decyduje o skuteczności algorytmu.

# Układy logiki rozmytej

Jak tego użyć w praktyce?

Wybór liczby i kształtu zbiorów rozmytych dla zmiennych wejściowych, oraz reguł wnioskowania zależy od problemu, jaki rozwiązujemy. Trafność tych wyborów decyduje o skuteczności algorytmu.

Wybór reprezentacji operatorów logicznych oraz metody defuzyfikacji zależy od wybranej techniki implementacji i ma niewielki wpływ na wyniki działania algorytmu.

# Układy logiki rozmytej

## Implementacje

Implementacja programowa: w postaci oprogramowania komputerowego, definicje funkcji przynależności, operatory logiczne i sposób defuzyfikacji są realizowane jako odpowiednie procedury lub funkcje

Implementacja sprzętowa cyfrowa: specjalizowany układ cyfrowy, wszystkie wielkości reprezentowane jako liczby, bloki fuzyfikacji, wnioskowania i defuzyfikacji to bloki wykonujące odpowiednie operacje w technice cyfrowej

Implementacja sprzętowa analogowa: specjalizowany układ analogowy, wszystkie wielkości reprezentowane jako wartości napięć lub prądów, bloki fuzyfikacji, wnioskowania i defuzyfikacji to bloki wykonujące odpowiednie operacje w technice analogowej

# Układy logiki rozmytej

## Implementacje

Implementacja programowa: program komputerowy (często dla mikrokontrolera); działa najwolniej, największe zużycie energii, obliczenia dokładne (ale wystarczą zwykle liczby 8-bitowe, a nawet 4-bitowe)

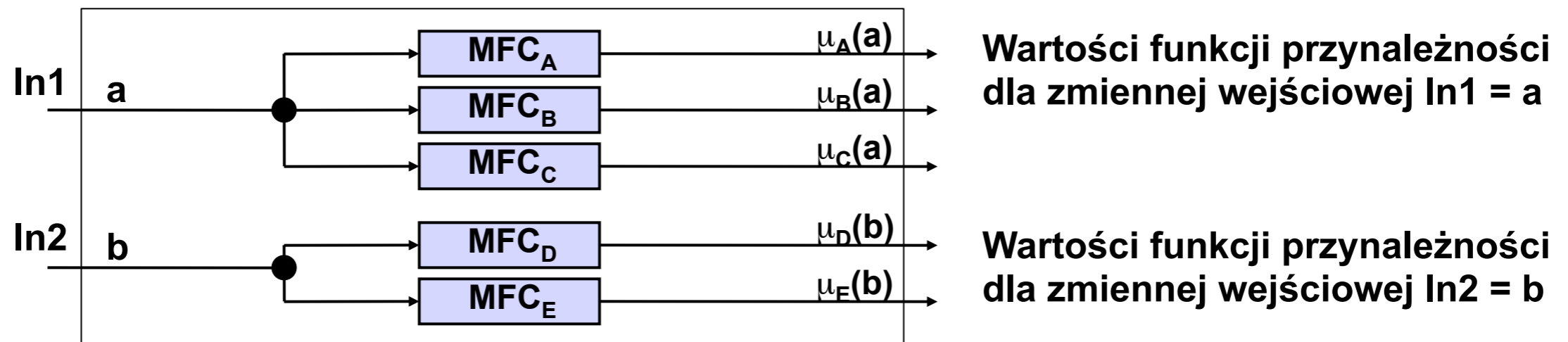
Implementacja sprzętowa cyfrowa: działa szybciej, układ może być programowalny, umiarkowane zużycie energii, obliczenia dokładne (ale wystarczą zwykle liczby 8-bitowe, a nawet 4-bitowe)

Implementacja sprzętowa analogowa: działa najszybciej, programowalność bardzo ograniczona, obliczenia realizowane analogowo mało dokładne (co nie jest problemem!)

# Układy logiki rozmytej

## Implementacje sprzętowe: blok fuzyfikacji

Jest to zbiór układów, z których każdy przyporządkowuje zmiennej wejściowej wartość funkcji przynależności do jednego zbioru rozmytego (MFC). Liczba tych układów jest równa liczbie wszystkich zbiorów rozmytych.





# Układy logiki rozmytej

Implementacje sprzętowe: blok wnioskowania

# Układy logiki rozmytej

Implementacje sprzętowe: blok wnioskowania

Ten blok jest zbiorem “bramek logicznych” logiki rozmytej, czyli układów wykonujących operacje negacji, I, LUB. Bramki te połączone są w sposób wynikający z reguł wnioskowania. Jest to odpowiednik tradycyjnego kombinacyjnego układu logicznego.

# Układy logiki rozmytej

Implementacje sprzętowe: blok wnioskowania

Ten blok jest zbiorem “bramek logicznych” logiki rozmytej, czyli układów wykonujących operacje negacji, I, LUB. Bramki te połączone są w sposób wynikający z reguł wnioskowania. Jest to odpowiednik tradycyjnego kombinacyjnego układu logicznego.

Schemat bloku wnioskowania wynika z przyjętych reguł wnioskowania. Istnieją metody minimalizacji logicznej takiego układu, ale są one odmienne od metod znanych z teorii układów logicznych, bo algebra Boole’a nie jest prawdziwa dla układów logiki rozmytej.

# Układy logiki rozmytej

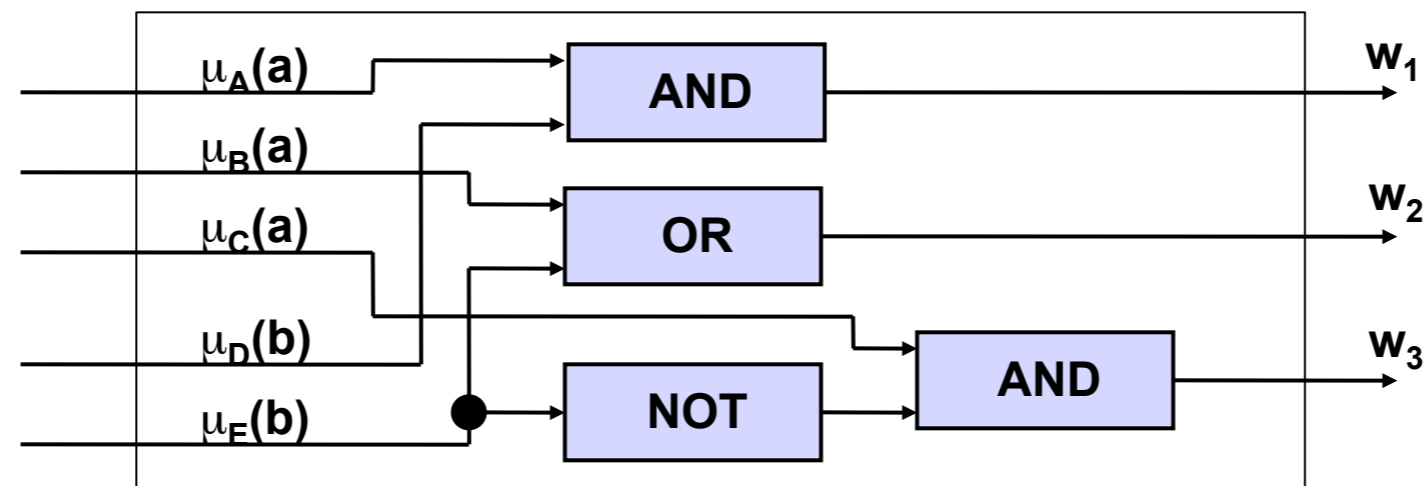
Implementacje sprzętowe: blok wnioskowania

**JEŚLI (a JEST A) I (b JEST D) TO ... (reguła 1)**

**JEŚLI (a JEST B) LUB (b JEST E) TO ... (reguła 2)**

**JEŚLI (a JEST C) I NIE (b JEST E) TO ... (reguła 3)**

Z bloku fuzyfikacji



Wyniki

# Układy logiki rozmytej

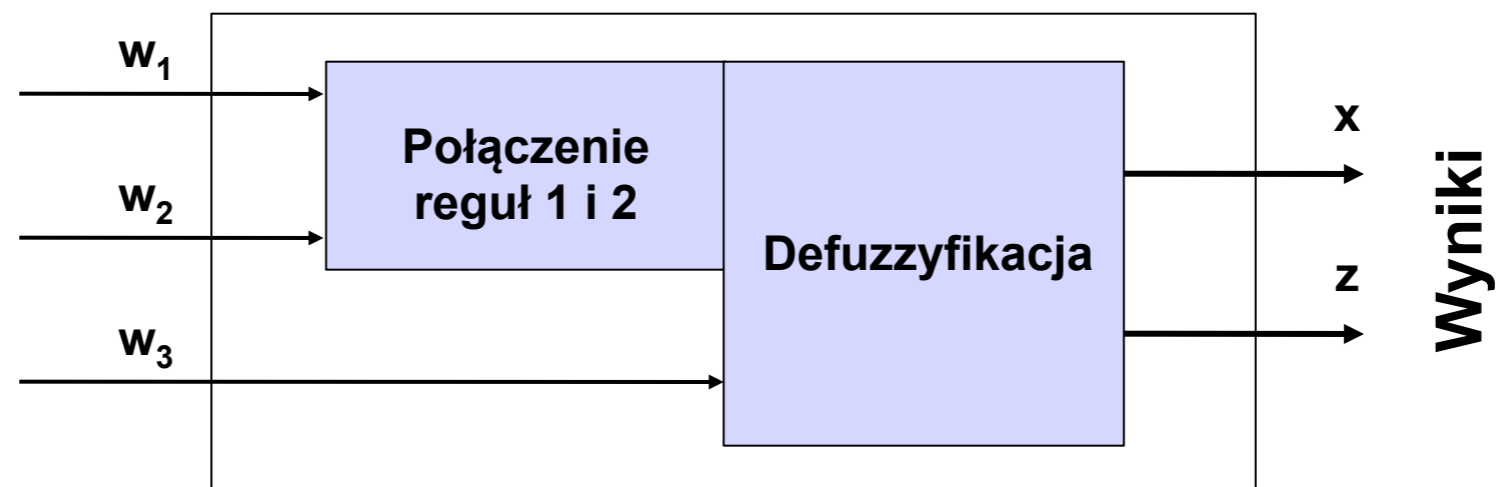
Implementacje sprzętowe: blok defuzyfikacji

**JEŚLI (a JEST A) I (b JEST D) TO (x JEST X) (reguła 1)**

**JEŚLI (a JEST B) LUB (b JEST E) TO (x JEST Y) (reguła 2)**

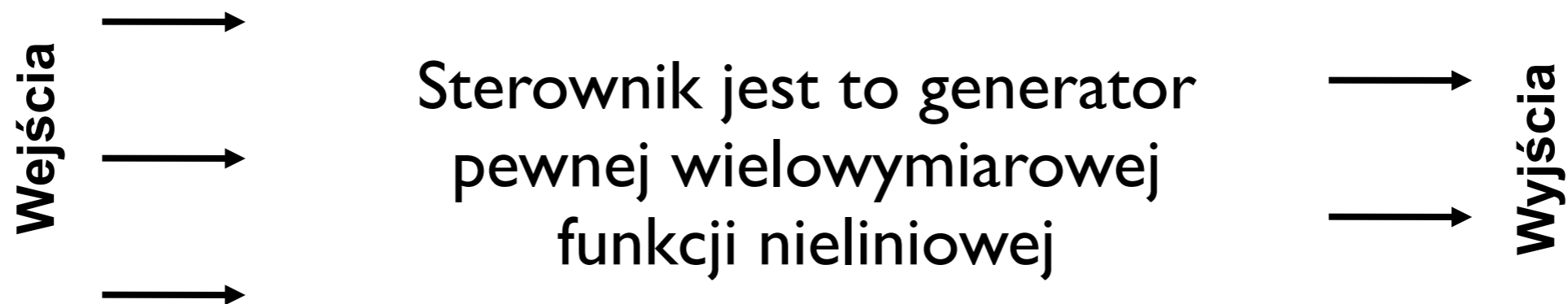
**JEŚLI (a JEST C) I NIE (b JEST E) TO (z JEST Z) (reguła 3)**

Z bloku wnioskowania



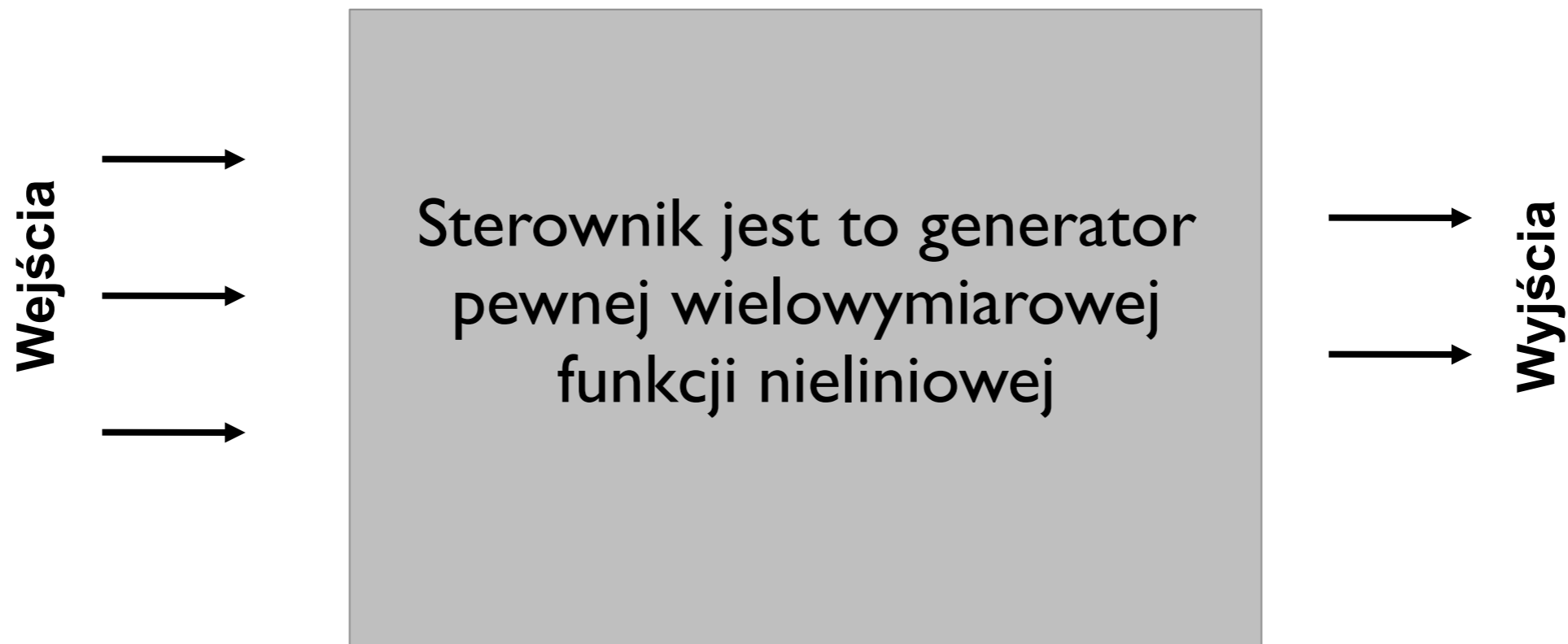
# Układy logiki rozmytej

Sterownik jako “czarna skrzynka”



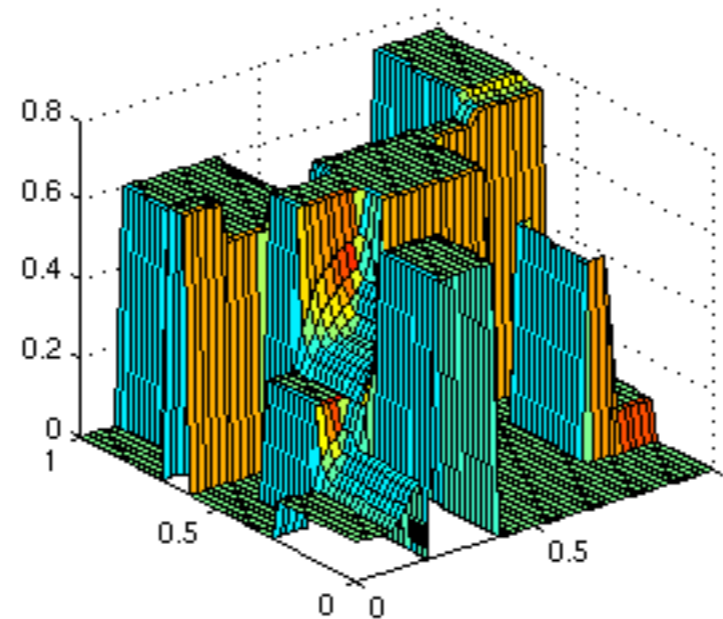
# Układy logiki rozmytej

Sterownik jako “czarna skrzynka”



# Układy logiki rozmytej

Sterownik jako “czarna skrzynka”

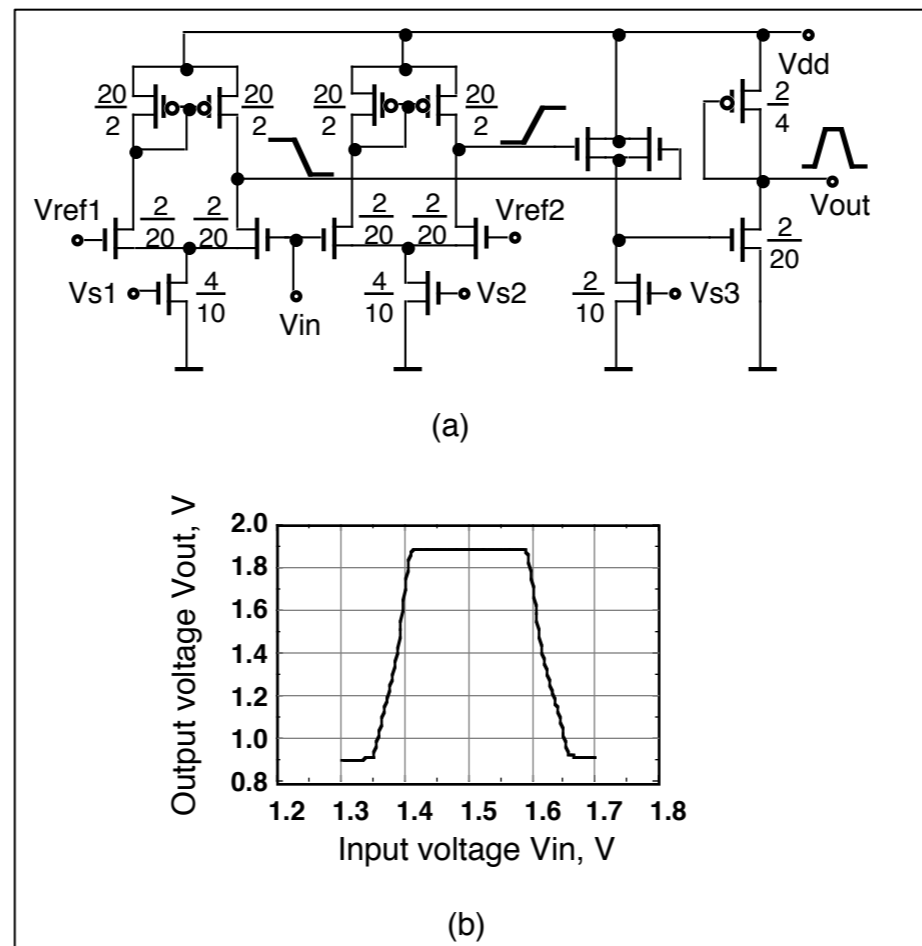


Przykład funkcji - dwie zmienne wejściowe (5 zbiorów rozmytych dla jednej, 4 dla drugiej), jedna zmienna wyjściowa, 6 reguł.



# Układy logiki rozmytej

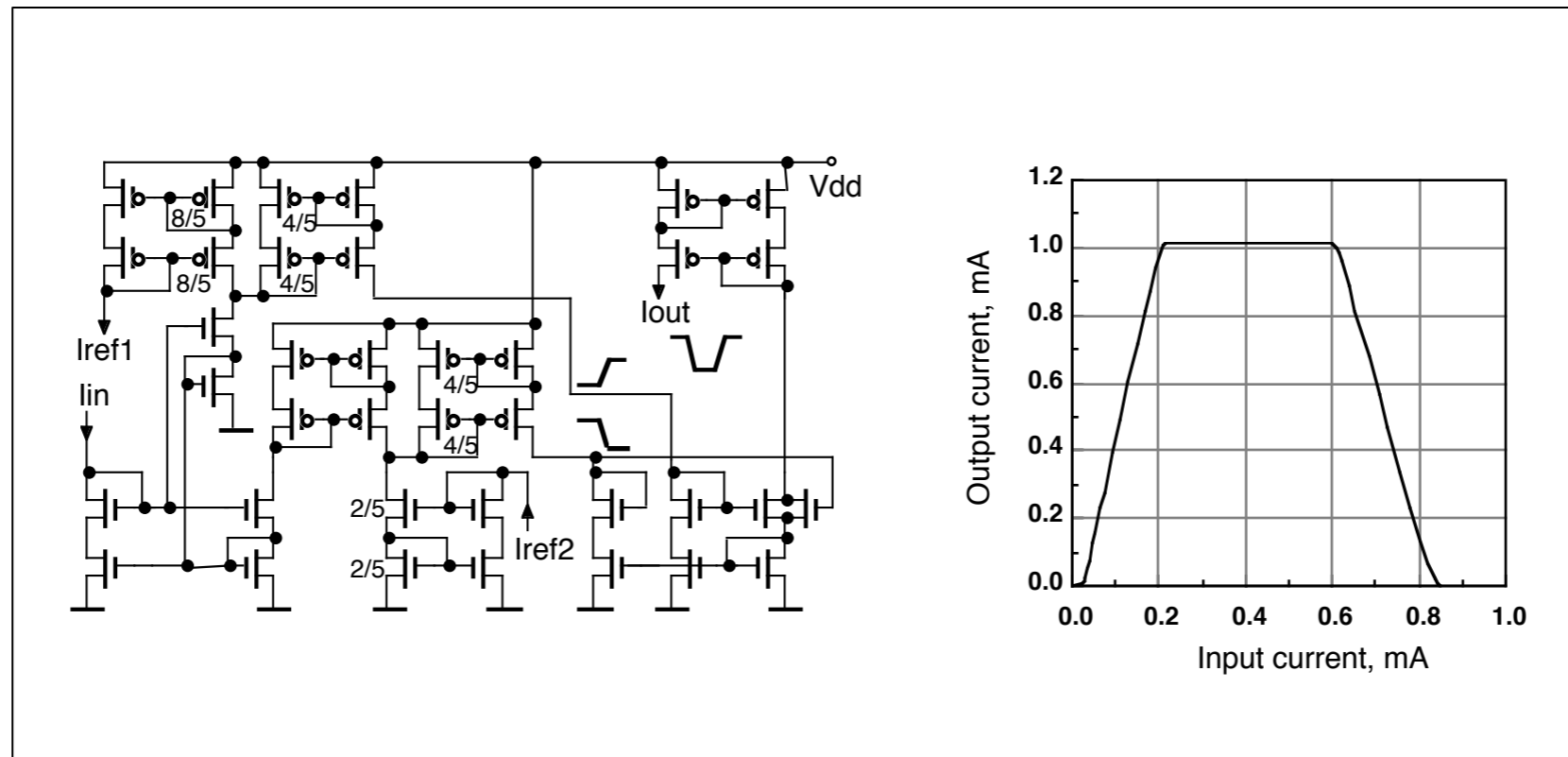
## Implementacje analogowe



Przykład układu CMOS generującego trapezoidalną funkcję przynależności

# Układy logiki rozmytej

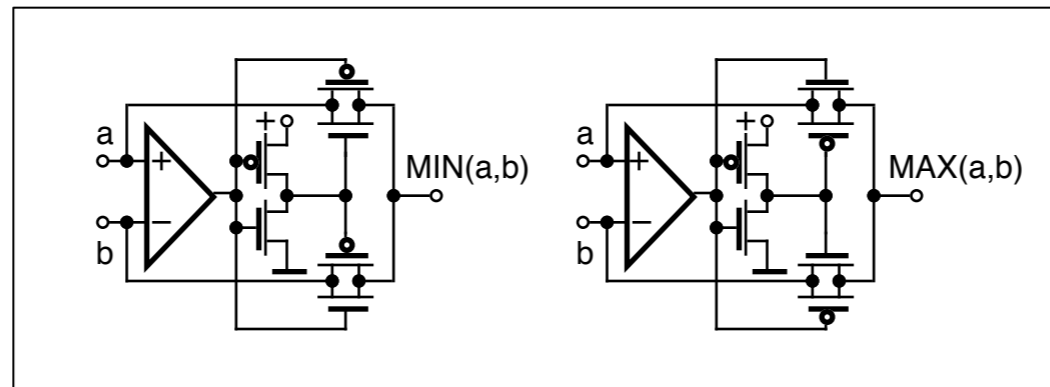
## Implementacje analogowe



Przykład układu CMOS generującego trapezoidalną funkcję przynależności (układ pracujący w trybie prądowym)

# Układy logiki rozmytej

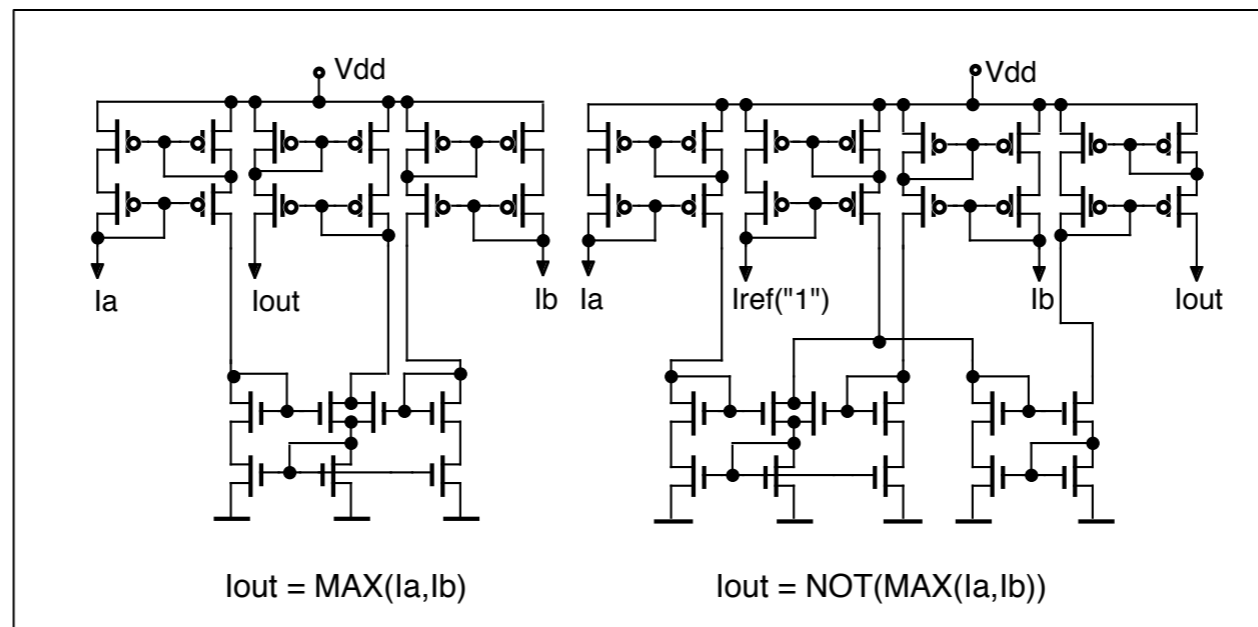
## Implementacje analogowe



Przykład układów CMOS realizujących operacje *min* i *max*

# Układy logiki rozmytej

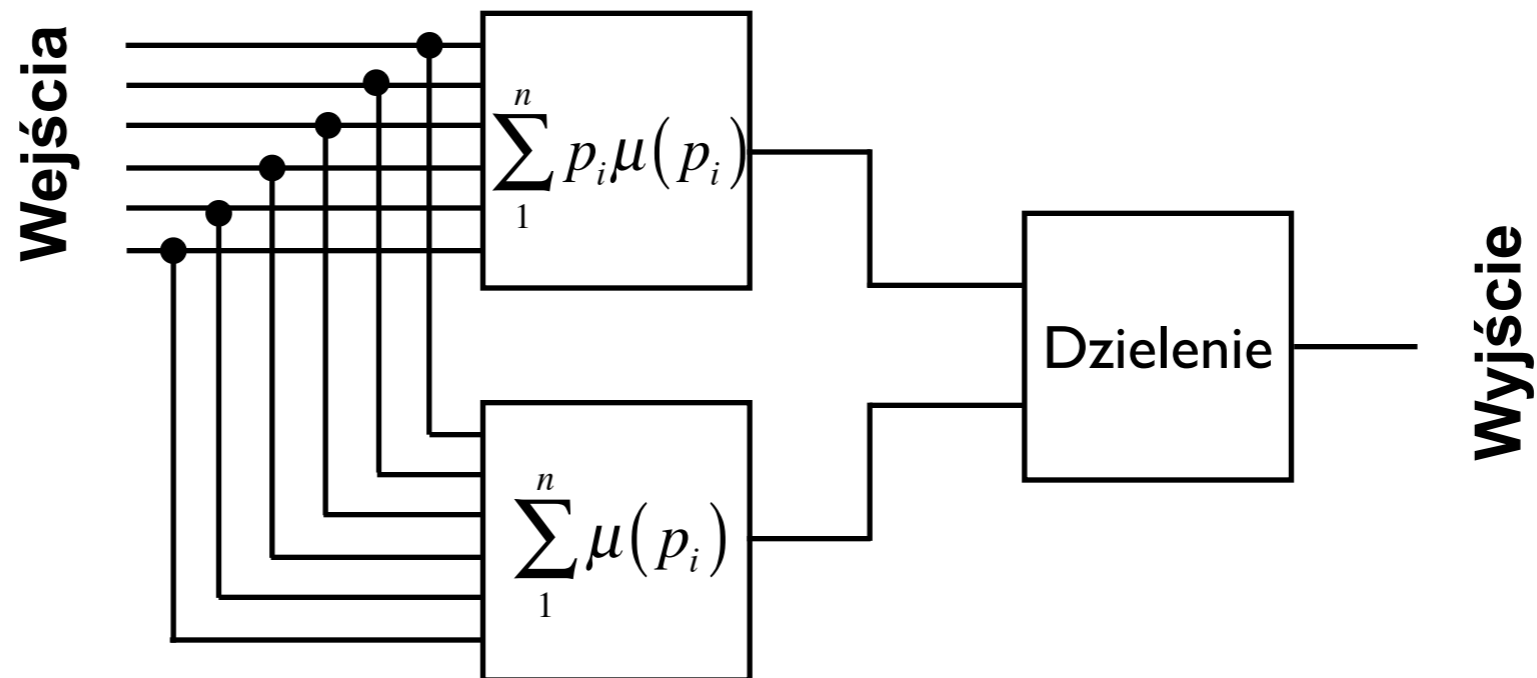
## Implementacje analogowe



Przykład układów CMOS realizujących operacje *min* i *max*  
(układy pracujące w trybie prądowym)

# Układy logiki rozmytej

## Implementacje analogowe



Idea układu realizującego defuzyfikację  
(całkowanie przybliżone sumowaniem)

# Układy logiki rozmytej

Podsumowanie

# Układy logiki rozmytej

## Podsumowanie

Układy logiki rozmytej pozwalają budować skuteczne algorytmy decyzyjne, klasyfikacji, sterowania bez tradycyjnych modeli matematycznych, a przy wykorzystaniu ludzkiego doświadczenia.

Implementacje mogą wykorzystywać technikę analogową