

Maria MRÓWCZYŃSKA<sup>1</sup>

## METODY SZTUCZNEJ INTELIGENCJI W ZASTOSOWANIU DO PRZEKSZTAŁCENIA UKŁADU WSPÓŁRZĘDNYCH

### Streszczenie

W treści artykułu rozpatrywano zagadnienie transformacji w postaci funkcji  $f: R^2 \rightarrow R^2$  jako zależności między współrzędnymi  $[x, y]$  układu pierwotnego a współrzędnymi  $[X, Y]$  układu docelowego. Zadanie transformacji  $[X, Y] = f(x, y)$  technicznie rozumianej jako przekształcenie układu współrzędnych, zrealizowano za pomocą sieci neuronowych jednokierunkowych dwuwarstwowych typu sigmoidalnego, sieci neuronowych radialnych, sieci neuronowych kaskadowych rekurencyjnych i systemów neuronowo-rozmytych z zastosowaniem modelu Takagi-Sugeno-Kanga. Zastosowane procedury numeryczne umożliwiają uzyskanie poziomu dokładności zadania odpowiadającej dokładności kartometrycznej obrazów w Systemach Informacji Przestrzennej.

### 1. Wstęp

Transformacja współrzędnych jest istotnym zagadnieniem z dziedziny geodezji ze względu wprowadzenia jednolitego, zgodnego ze standardem układu współrzędnych w dobie integracji europejskiej. Przekształcenie współrzędnych układów kartograficznych funkcjonujących w różnych systemach odniesienia jest możliwe wówczas, gdy znana jest formuła określająca wzajemne związki między współrzędnymi punktów łącznych o znanych współrzędnych w obu systemach. Metodą najczęściej stosowaną do tego rodzaju obliczeń jest transformacja Helmerta (transformacja afiniczna), której estymacja parametrów jest realizowana przy założeniu minimalnej ważonej długości wektora poprawek współrzędnych punktów dopasowania

$$\mathbf{v}^T \mathbf{P} \mathbf{v} = \min \quad (1)$$

gdzie:  $\mathbf{v}$  - wektor poprawek współrzędnych

$\mathbf{P}$  - macierz wagowa współrzędnych.

Transformacja ta stosowana jest w zadaniach:

- transformacji współrzędnych geodezyjnych i fotogrametrycznych,
- kalibracji afinicznej,
- wyznaczenia parametrów odkształcenia elementu skończonego obiektu inżynierskiego.

W procesie wyznaczania parametrów transformacji metoda Helmerta wykazuje niestety istotną niedoskonałość w postaci braku odporności na błędy grube.

W odniesieniu do wspomnianej wyżej metody, autorka proponuje alternatywne podejście przekształcenia współrzędnych za pomocą sieci neuronowych jednokierunkowych systemów neuronowo-rozmytych. Wyniki realizacji numerycznej zadania transformacji współrzędnych punktów z układu pierwotnego na układ docelowy, porównano w aspekcie poziomu dokładności z wynikami, jakie uzyskano z pomocą algorytmów profesjonalnych.

---

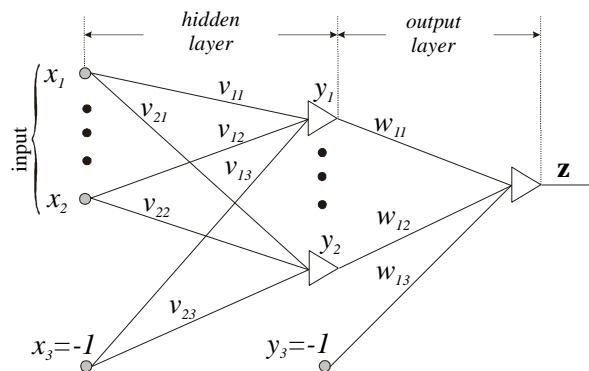
<sup>1</sup>dr inż., Wydział Inżynierii Lądowej i Środowiska, Zakład Geotechniki i Geodezji, Uniwersytet Zielonogórski

## 2. Wybrane metody przekształcania współrzędnych

### 2.1. Algorytmy gradientowe uczenia sieci jednokierunkowych dwuwarstwowych

Sztuczne sieci neuronowe odznaczają się mocnymi podstawami teoretycznymi oraz użytecznością w praktyce. Dowolny problem, który może być rozwiązany za pomocą klasycznego modelowania lub metodami statystycznymi, daje się rozwiązać przy użyciu sieci neuronowych (Osowski 2006). Zdecydowana większość zastosowań sieci neuronowych dotyczy sieci jednokierunkowych wielowarstwowych typu sigmoidalnego. Z matematycznego punktu widzenia pełnią rolę aproksymacji stochastycznej funkcji wielu zmiennych, która odwzorowuje zbiór zmiennych wejściowych  $\mathbf{x} \in R^N$  w zbiór zmiennych wyjściowych  $\mathbf{y} \in R^M$ .

W celu dokonania obliczeń dotyczących z przekształceniem współrzędnych z układu pierwotnego na układ wtórny wykorzystano sieć neuronową dwuwarstwową (perceptron dwuwarstwowy), której ogólny schemat przedstawiono na rys. 1.



Rys. 1. Sieć neuronowa dwuwarstwowa (Żurada et al. 1996)

Określenie relacji jako związku przyczynowo – skutkowego między wejściem i wyjściem sieci o ustalonej *a priori* topologii jest procesem uczenia sieci, który polega na adaptacji parametrów sieci zwanych wagami.

Technika uczenia sieci neuronowych wykorzystuje gradientowe metody optymalizacji. Podstawę algorytmów uczenia sieci stanowi funkcja celu (funkcja energetyczna), definiowana z wykorzystaniem metryki euklidesowej jako suma kwadratów różnic między wartościami sygnałów wyjściowych sieci a wartościami zadanymi w postaci:

$$E = \frac{1}{2} \sum_{i=1}^p \sum_{j=1}^M \left( z_j^{(i)} - d_j^{(i)} \right)^2, \quad (2)$$

gdzie:

- $p$  - liczba wektorów wejściowych,
- $M$  - liczba współrzędnych wektora wyjściowego,
- $z_j^{(i)}$  - współrzędna wektora wyjściowego ( $i = 1, 2, \dots, p$ ), ( $j = 1, 2, \dots, M$ ),
- $d_j^{(i)}$  - współrzędna wektora zadanego  $(x^{(i)}, y^{(i)})$  ( $i = 1, 2, \dots, p$ ), ( $j = 1, 2, \dots, M$ )

Minimalizacja funkcji celu (2) z uwzględnieniem sigmoidalnej funkcji aktywacji, polega na dokonaniu korekcji wag  $v_{ij}$  oraz  $w_{ij}$  (rys. 1) na podstawie informacji, które zawiera gradient funkcji celu  $\nabla E(\mathbf{w})$ , zgodnie z zależnością

$$\Delta \mathbf{w} = -\eta \nabla E(\mathbf{w}), \quad (3)$$

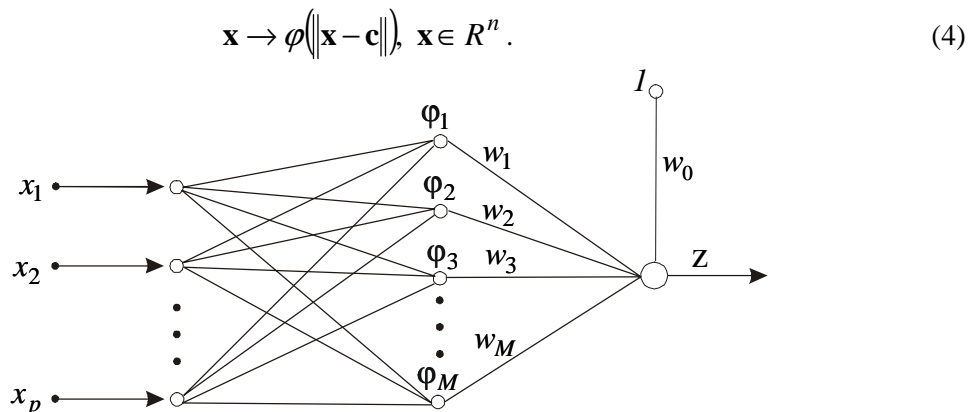
w której  $-\nabla E(\mathbf{w})$  oznacza globalny kierunek minimalizacji, zaś  $\eta$  jest współczynnik uczenia (współczynniki kroku iteracyjnego).

Większość gradientowych metod optymalizacji wykorzystuje model kwadratowy funkcji w sąsiedztwie  $\mathbf{w}(t)$  (rozwiniecie w szereg Taylora), w celu pozyskania informacji na temat krzywizny funkcji zawartej w hesjanie w kolejnym kroku aktywacji  $t$ . W celu uzyskania zbieżności do rozwiązania optymalnego wykorzystano w pracy powszechnie znane z teorii optymalizacji gradientowe metody uczenia sieci, a mianowicie (Bishop 2006):

- największego spadku (liniowe przybliżenie funkcji  $E(\mathbf{w})$ ),
- quasi-newtonowskie: zmiennej metryki, gradientów sprzężonych, Levenberga–Marquardta oraz algorytm *RPROP* (*Resilient backPropagation*).

## 2.2. Sieci neuronowe radialne

Aproksymacja stochastyczna funkcji wielu zmiennych realizowana za pomocą sieci neuronowych wielowarstwowych ma charakter globalny, ponieważ odwzorowanie estymowanej funkcji w dowolny punkt przestrzeni następuje w wyniku łącznego pobudzenia wielu neuronów. Komplementarnym sposobem odwzorowania zbioru wejściowego w zbiór wyjściowy jest dopasowanie wielu pojedynczych funkcji aproksymujących do elementów zbioru wartości zadanych w ograniczonym obszarze przestrzeni wielowymiarowej. Odwzorowanie ma charakter lokalny, natomiast odwzorowanie pełnego wektora wejściowego  $\mathbf{x} \in R^n$  w wektor wyjściowy  $y \in R^m$  stanowi rezultat w postaci odwzorowań lokalnych, realizowany przez sieci o radialnych funkcjach bazowych (rys.2), złożone z neuronów, które w warstwie ukrytej realizują odwzorowanie (Osowski 2006):



Rys. 2. Sieć o radialnych funkcjach bazowych

Jeżeli na wejściu podamy  $M$  wektorów wejściowych  $\mathbf{x}(1), \dots, \mathbf{x}(M)$ , które mają zostać odwzorowane w zbiór liczb rzeczywistych  $d(1), \dots, d(M)$  wówczas problem polega na poszukiwaniu estymatora funkcji odwzorowania

$$F(\mathbf{x}_i) = d_i. \quad (5)$$

Wprowadzając liczbę neuronów  $M$  jako znacznie mniejszą od liczby wzorców uczących  $p$  ( $M \ll p$ ), wówczas estymowana funkcja odwzorowania przyjmuje postać

$$F(\mathbf{x}) = \sum_{i=1}^M w_i \varphi(\|\mathbf{x} - \mathbf{c}_i\|). \quad (6)$$

Symbole zawarte we wzorze (6) oznaczają:

$w_0, w_1, \dots, w_M$  - współczynniki skalarne zwane wagami,

$c_1, c_2, \dots, c_M \in R^d$  - wektory nazywane centrami, nad centrami lokowane są funkcje  $\varphi$

$\mathbf{x} \in R^d$  - wektor wejść.

Najczęściej stosowaną funkcją radialną  $\varphi$  (obok wielu innych, niektóre doskonałe) jest funkcja Gaussa (postać uproszczona)

$$\varphi(\mathbf{x}) = \varphi(\|\mathbf{x} - \mathbf{c}_i\|) = \exp\left(-\frac{\|\mathbf{x} - \mathbf{c}_i\|^2}{2\sigma_i^2}\right), \quad (7)$$

która wprowadzie nie ma nic wspólnego z założeniem rozkładu normalnego, lecz spełnia warunki metody Parzena i w praktyce daje dobre wyniki. Na podstawie odpowiedniej procedury minimalizującej funkcję celu

$$E = \sum_{i=1}^p \left[ \sum_{j=1}^M w_j \varphi(\|\mathbf{x}_i - \mathbf{c}_j\| - \mathbf{d}_i) \right]^2 \quad (8)$$

wyznaczamy współrzędne wektora wag  $\mathbf{w}$  oraz wartości parametrów  $\mathbf{c}$  i  $\sigma$  funkcji radialnych. Jeżeli znane są parametry, wówczas rozwiązanie minimalizacji jest realizowane techniką pseudoinwersji macierzy Grena

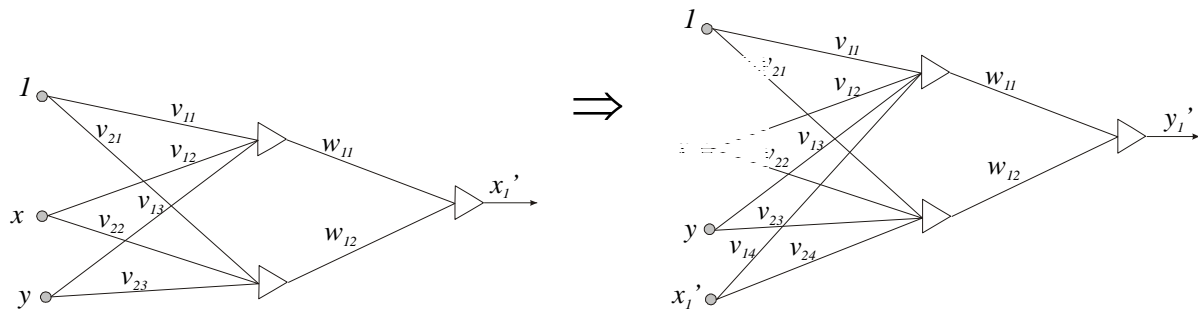
$$\mathbf{w} = \mathbf{G}^+ \mathbf{d}. \quad (9)$$

Uzyskanie zadawalających rezultatów przekształcenia współrzędnych za pomocą sieci neuronowych o radialnych funkcjach bazowych zależy od odpowiedniej architektury sieci, liczby radialnych funkcji bazowych i ich szerokości oraz wartości współczynników uczenia (Barsi 2001; Gil 2006).

### 2.3. Sieci neuronowe kaskadowe rekurencyjne

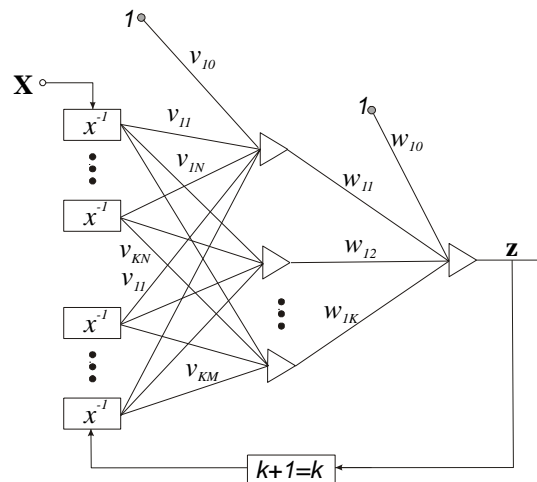
Sieci neuronowe kaskadowe rekurencyjne (*Recurrent Cascade MultiLayer Perceptron*) typu perceptronowego powstają poprzez dodanie odpowiednich sprzężeń zwrotnych do sieci jednokierunkowych. Sprzężenia zwrotne są wyprowadzane z warstwy wyjściowej neuronów i kierowane do neuronów warstwy wejściowej. Jest to zatem układ dynamiczny, który ze względu na sposób tworzenia sygnału wyjściowego działa jak sieć jednokierunkowa wielowarstwowa.

Działanie sieci neuronowej kaskadowej dzieli się na dwa etapy. W etapie pierwszym zgodnie ze strukturą przedstawioną na rys. 3 została wykorzystana sieć neuronowa kaskadowa nierekurencyjna. Architekturę sieci kaskadowej nierekurencyjnej jako sieci jednokierunkowej stanowi jednostopniowy wzrost wymiaru wektora wejściowego i wyjściowego. W fazie początkowej pierwsza warstwa sieci otrzymuje pobudzenie z warstwy wejściowej w postaci wektora  $\mathbf{x}$  o współrzędnych  $(x, y)$  punktu w układzie pierwotnym, zaś pożądanym sygnałem wyjściowym jest współrzędna  $x'_1$  punktu w układzie docelowym. Po dokonaniu procesu uczenia następuje wzrost wymiaru wektora wejściowego, który zawiera zarówno współrzędne  $(x, y)$  w układzie pierwotnym oraz współrzędną  $x'_1$  uzyskaną na wyjściu, czyli  $\mathbf{x}' = [x, y, x'_1]$ . Przyłożenie tego wektora na wejściu uruchamia kolejny cykl uczenia z pożądanym sygnałem wyjściowym w postaci współrzędnej  $y'_1$  w układzie docelowym. W rezultacie takiego postępowania otrzymujemy wektor  $\mathbf{y}' = [x, y, x'_1, y'_1]$ . Liczba cykli uczących odpowiada liczebności zbioru uczącego.



Rys. 3. Sieć neuronowa kaskadowa nierekurencyjna

W ramach etapu drugiego postępowanie koncentruje się na doborze struktury sieci neuronowej kaskadowej rekurencyjnej (rys. 4).



Rys. 4. Sieć neuronowa kaskadowa rekurencyjna (Osowski 2006)

W drugim etapie budujemy sieć neuronową kaskadową rekurencyjną (rys. 4), w której wektor wejściowy tworzą współrzędne punktów w układzie pierwotnym i wtórnym  $\mathbf{X} = (x, y, x', y')$  oraz istnieją połączenia zwrotne pomiędzy warstwą wyjściową i wejściową. Należy zauważyć, że w trakcie procesu uczenia aktualizowany jest wektor wejściowy. Dla iteracji  $k+1$  w wektorze wejściowym występują współrzędne punktów w układzie wtórnym uzyskane w iteracji  $k$ . Algorytmy uczenia sieci rekurencyjnej wykorzystują omówione wcześniej gradientowe metody optymalizacji i tak jak w przypadku sieci neuronowej jednokierunkowej obliczamy gradient funkcji celu (2) względem każdej wagi. Szczegółowa postać wzoru rekurencyjnego umożliwia obliczenie gradientu w dowolnej chwili  $k$  na podstawie jego wartości w chwilach poprzednich został zawarty w pracy (Osowski 2006).

### 3. System neuronowo rozmyty typu adaptacyjnego

Systemy neuronowo rozmyte są to sieci neuronowe, które charakteryzują się zdolnością przetwarzania zbiorów rozmytych. Systemy neuronowo – rozmyte umożliwiają interpretację wiedzy zgromadzonej w wagach połączeń neuronowych, która stanowi podstawę formułowania zbiorów rozmytych reguł warunkowych „jeżeli – to”. Jedną z podstawowych metod pozyskiwania baz wiedzy złożonej z reguł „jeżeli – to” polega na ekstrakcji reguł na podstawie numerycznych danych na temat wejść i wyjść modelowanego zjawiska.

W tym przypadku zazwyczaj stosowany jest system Takagi – Sugeno – Kanga (TSK), którego zaletą jest mały nakład obliczeniowy związany z wyznaczeniem wartości wyjściowej systemu. Bazę wiedzy systemu TSK stanowi  $M$  reguł wnioskowania „jeżeli – to” w połączeniu z liniową funkcją (wielomian pierwszego stopnia) w konkluzji  $k$ -tej reguły wnioskowania, zapisaną w relacji (Markowska-Kreczmar, 2006)

$$M^{(k)} = \text{jeżeli } \bigwedge_{1 \leq j \leq N} x_j \text{ jest } A_j^{(k)}, \text{ to } y = f_k(\mathbf{x}) \text{ dla } k = 1, 2, \dots, M \quad (10)$$

przy czym funkcja liniowa

$$f_k(\mathbf{x}) = p_{k0} + \sum_{j=1}^N p_{kj} x_j \quad (11)$$

gdzie  $p_k$  oznacza  $(N+1)$  - wymiarowy wektor parametrów. Zbiór prostych funkcji liniowych  $f_k(\mathbf{x})$  stwarza możliwość modelowania skomplikowanych zależności między wejściem a wyjściem systemu.

Zbiory rozmyte jako uogólnienie zbiorów zwykłych charakteryzują się „częściową” przynależnością elementów do danego zbioru. Podobnie jak zbiory klasyczne opisane za pomocą funkcji charakterystycznych, tak zbiory rozmyte opisujemy za pomocą funkcji przynależności o wartościach z zakresu  $[0,1]$  (Zadeh 1965). Jedną z najczęściej stosowanych funkcji przynależności do reprezentacji rozmytej liczb jest funkcja gaussowska, definiowana dla zmiennej  $x$  o centrum w  $c$  i odchyleniu standardowym  $\sigma$  określonej dla zbioru  $A$  w postaci (postać ogólna) (Duch 2001; Osowski 2006)

$$\mu_A(x) = \exp \left[ - \left( \frac{x-c}{\sigma} \right)^2 \right]. \quad (12)$$

W pracy funkcja (12) została zastąpiona funkcją gaussowską wymierną (funkcja dzwonowa) opisaną wzorem

$$\mu_A(x) = \frac{1}{1 + \left( \frac{x-c}{\sigma} \right)^{2b}}, \quad (13)$$

gdzie parametr  $b$  wpływa na kształt uogólnionej funkcji gaussowskiej (funkcja Gaussa, funkcja trójkątna oraz trapezoidalna). Wypadkowa funkcji przynależności  $\mu_A(\mathbf{x})$  stanowi agregacja informacji zawartych w przesłankach implikacji. Operator agregacji reprezentuje odwzorowanie  $\oplus: [0,1]^N$  w celu uzyskania wartości  $x \in [0,1]$  czyli  $\mathbf{x} = \oplus(x_1, x_2, \dots, x_N)$ .

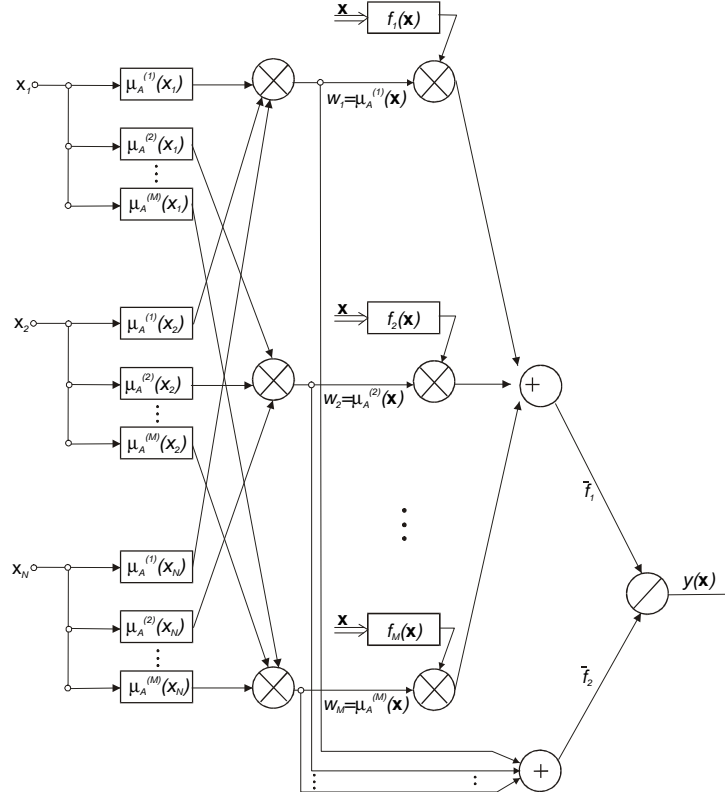
Zgodnie z operacją rozmytą, agregację przesłanek implikacji będziemy interpretować w postaci iloczynu algebraicznego, którą dla  $k$ -tej reguły wnioskowania wyraża zależność

$$\mu_A^{(k)}(\mathbf{x}) = \prod_{j=1}^N \left[ \frac{1}{1 + \left( \frac{x_j - c_j^{(k)}}{\sigma_j^{(k)}} \right)^{2b_j^{(k)}}} \right]. \quad (14)$$

Wyjściową wartość systemu uzyskujemy jako średnią ważoną wartości wyjściowych poszczególnych reguł

$$y(\mathbf{x}) = \frac{\sum_{k=1}^M \mu_A^{(k)}(\mathbf{x}) f_k(\mathbf{x})}{\sum_{k=1}^M \mu_A^{(k)}(\mathbf{x})}. \quad (15)$$

Architektura sieci neuronowo rozmytej została przedstawiona na rys. 4.



Rys. 5. Struktura sieci neuronowo rozmytej TSK

W sieci wyróżniamy pięć warstw. W warstwie pierwszej wartości wejściowe  $x_j$  podlegają rozmyciu oraz wyznaczane są parametry  $c_j^{(k)}, \sigma_j^{(k)}, b_j^{(k)}$  (centra, szerokości i kształty) z zastosowaniem gradientowych metod uczenia. W warstwie drugiej następuje agregacja przesłanek implikacji. W warstwie trzeciej generowane są wartości funkcji  $f_k(\mathbf{x})$  (funkcja TSK) na podstawie adaptacji parametrów  $p_{kj}$  ( $k=1,2,\dots, M, j=1,2,\dots, N$ ). Dwa neurony sumacyjne tworzą warstwę czwartą, z których jeden oblicza sumę ważoną sygnałów  $y_k(\mathbf{x})$ , a drugi sumę wag (wagi mają interpretację wartości agregowanej  $\mu_A^{(k)}(\mathbf{x})$ ). Ostatnia warstwa wartości agregowanej  $\mu_A^{(k)}(\mathbf{x})$ . Ostatnia warstwa czyli piąta zawiera tylko jeden neuron, który generuje sygnał wyjściowy sieci  $y(\mathbf{x})$  według zależności (15).

System rozmyty Takagi – Sugeno –Kanga i normalizowane radialne sieci neuronowe przy spełnieniu określonych warunków są równoważne przy spełnieniu określonych warunków (Łęski, 2008).

#### 4. Przykład liczbowy

Transformację współrzędnych z układu pierwotnego na układ docelowy zrealizowano z zastosowaniem odpowiedniej struktury sieci neuronowych dopasowanych do danego zadania. W pracy podjęto próbę doboru struktury i parametrów sieci tak, aby aproksymować wartości zadane w

sposób optymalny w sensie statystycznym, czyli uzyskać akceptowalnie mały poziom błędu na danych testowych.

Rozwiązanie zadania transformacji przeprowadzono na podstawie zbioru uczącego i testowego, przy czym liczebność każdego zbioru wynosiła 1600 punktów. Punkty zbioru uczącego reprezentowały punkty dostosowania, natomiast wynik przekształcenia współrzędnych punktów zbioru testowego z układu pierwotnego na układ docelowy stanowi rezultat działania sieci na danych nie uczestniczących w procesie uczenia. Dla każdej z wymienionych i zaimplementowanych struktur sieci wyszczególnionych w tablicach 1÷5 została określona dla danej metody minimalizacji optymalna wartość funkcji celu, której miarę wyraża wartość pierwiastka błędu średniokwadratowego (*RMSE*). Jako kryterium zatrzymania procesu iteracyjnego przyjęto zmianę wartości gradientu w dwóch kolejnych iteracjach na poziomie  $1e-10$ .

W uzupełnieniu informacji dotyczącej procesu uczenia sieci należy dodać, że dla sieci perceptronowych wielowarstwowych zastosowano zmienną liczbę warstw ukrytych oraz zmienną liczbę neuronów w poszczególnych warstwach. Trening sieci radialnych uwzględniał zmienną liczbę radialnych funkcji bazowych oraz zmiennych wartości parametru  $\sigma$  w doborze ich kształtu.

Skuteczny sposób pozyskania wyników transformacji o wysokiej dokładności polega na wykorzystaniu wyników odwzorowania  $[X, Y] = f(x, y)$  za pomocą sieci neuronowych, które stanowią zmienne wyjściowe sieci neuronowo – rozmytej (*ang. neuro-fuzzy networks*) Takagi – Sugeno – Kanga (*TSK*). Istotnym zagadnieniem budowy struktury sieci neuronowo – rozmytej *TSK* jest określenie liczby reguł wnioskowania rozmytego, którą wyznaczono na podstawie minimalnej wartości globalnej miary statystycznej

$$\alpha = a_1 V_h - a_2 D_A - a_3 D_w + a_4 t_A. \quad (16)$$

Minimum lokalne tej funkcji opisanej w pracy (Babuska and Verbruggen 1998) umożliwia wyznaczenie suboptymalnej liczby klastrów dla danego zbioru danych. Współczynniki skali  $a_i$  ( $i = 1, 2, 3, 4$ ) wyznaczono za pomocą algorytmu genetycznego z zastosowaniem turniejowej selekcji chromosomów, dla prawdopodobieństwa krzyżowania jednopunktowego równego 0,77 i prawdopodobieństwa mutacji równego 0,0077.

Skuteczność działania poszczególnych algorytmów do rozwiązania zadania transformacji współrzędnych ilustrują wyniki zawarte w tablicach 1÷5, w postaci błędu średniokwadratowego *RMSE* obliczanego jako

$$RMSE = \sqrt{\frac{1}{P} \sum_{p=1}^P (d_p - z_p)^2}. \quad (17)$$

Tablica 1. Charakterystyka dokładności odwzorowania  $[X, Y] = f(x, y)$  za pomocą sieci neuronowych

Gradientowe metody optymalizacji	Architektura sieci	Błąd transformacji <i>RMSE</i> [m]	
		zbiór uczący	zbiór testowy
Levenberga – Marquardta	2_5_2_1	0.011	0.012
algorytm <i>RPROP</i>		0.012	0.013
zmiennej metryki		0.012	0.013
gradienty sprzężone		0.132	0.158
największego spadku		0.794	0.975



Tablica 2. Charakterystyka dokładności odwzorowania  $[X, Y] = f(x, y)$  za pomocą sieci radialnych w zależności od liczby funkcji radialnych.

Liczba radialnych funkcji bazowych	Błąd transformacji <i>RMSE</i> [m]	
	zbiór uczący	zbiór testowy
60	0.034	0.029
90	0.013	0.014
100	0.057	0.092

Tablica 3. Charakterystyka dokładności odwzorowania  $[X, Y] = f(x, y)$  za pomocą sieci radialnych w zależności od zastosowanych radialnych funkcji bazowych

Rodzaj radialnej funkcji bazowej	Błąd transformacji <i>RMSE</i> [m]	
	zbiór uczący	zbiór testowy
funkcja Gaussa	0.013	0.014
funkcja sklejana czwartego stopnia	0.015	0.022
funkcja sklejana trzeciego stopnia	0.013	0.024
funkcja bicentralna	0.008	1.124
funkcja potęgowa Hardy'ego	0.045	189.271
funkcja liniowa	0.099	220.404
funkcja potęgowa	0.020	227.510

Tablica 4. Charakterystyka dokładności odwzorowania  $[X, Y] = f(x, y)$  za pomocą sieci neuronowych rekurencyjnych o strukturze kaskadowej

Gradientowe metody optymalizacji	Architektura sieci (w pierwszej iteracji)	Błąd transformacji <i>RMSE</i> [m]	
		zbiór uczący	zbiór testowy
Levenberga – Marquardta	2_10_1	0.007	0.008
gradienty sprzężone	2_15_1	0.009	0.011
zmiennej metryki	2_10_1	0.022	0.034
największego spadku	2_15_1	0.594	0.684

Tablica 5. Charakterystyka dokładności odwzorowania  $[X, Y] = f(x', y')$  uzyskana w wyniku zastosowania systemu TSK na bazie wyników określonych za pomocą sieci neuronowych

Gradientowa metoda optymalizacji	Błąd transformacji <i>RMSE</i> [m] (zbiór testowy)	
	Sieć neuronowa	System TSK
Levenberga – Marquardta	0.012	0.008
algorytm <i>RPROP</i>	0.013	0.008
zmiennej metryki	0.013	0.010
gradienty sprzężone	0.158	0.092
największego spadku	0.975	0.729

## 5. Podsumowanie

Zastosowane metody przekształcania współrzędnych punktów z systemu pierwotnego do systemu wtórnego umożliwiają pełniejsze wykorzystanie potencjału obliczeniowego sztucznej inteligencji. Optymalizowaną miarą jakości sieci neuronowych i zastosowanych algorytmów do przekształcania współrzędnych z jednego systemu do drugiego jest powszechnie preferowany błąd średniokwadratowy na wyjściu, traktowany jako podstawowa miara błędów o czysto matematycznej konstrukcji.

Przeglądając dane zawarte w tablicach 1÷5 można porównać efektywność algorytmów uczących sieci neuronowych o określonej strukturze na podstawie przeprowadzonych testów. Stosując sieci neuronowe z zastosowaniem gradientowych algorytmów uczenia, najbardziej korzystne wyniki optymalizacji uzyskano z zastosowaniem metody Levenberga – Marquardta, metody zmiennej metryki oraz algorytmu *RPROP*. Pozostałe metody gradientowe okazały się nieskuteczne do rozwiązania postawionego zadania (tabl. 1). Kolejną procedurą wykorzystaną do transformacji współrzędnych sieci typu radialnego jako naturalne uzupełnienie sieci sigmoidalnych. Zadawające rezultaty przekształcenia współrzędnych można uzyskać przy odpowiednim doborze architektury sieci, liczby i rodzaju radialnych funkcji bazowych i ich szerokości oraz jak zawsze wartości współczynników uczenia. (tabl. 2 i 3).

Zdecydowaną poprawę jakości dopasowania układów współrzędnych uzyskano w wyniku zastosowania sieci neuronowych o strukturze kaskadowej oraz systemu neuronowo rozmytego *TSK* działającego na bazie wyników uczenia sieci neuronowych metodami gradientowymi. Wyniki optymalnej aktywizacji neuronów wyjściowych zawarte w tabl. 4 i 5 dowodzą, że silnie nieliniowe układy złożone z dużej liczby zmiennych należy optymalizować przez wytworzenie wyników pośrednich, które następnie przetworzone dają ostatecznie bardziej dokładne rezultaty przeprowadzonych obliczeń.

### Literatura

- [1] BABUSKA R., VERBRUGGEN H.B., *Constructing fuzzy models by product space clustering*, in "Fuzzy model identification", H. Hellendoorn, D. Driankov, Eds., Springer, Berlin, 1998,
- [2] BARSİ A., *Performing coordinate transformation by artificial neural network*, Allgemeine Vermessungs-Nachrichten 4/2001,
- [3] BISHOP CH.M., *Pattern Recognition and Machine Learning*, Springer, New York, 2006,
- [4] DUCH W., KORBICZ J., RUTKOWSKI L., TADEUSIEWICZ R., *Sieci neuronowe*, Akademicka Oficyna Wydawnicza Exit, Warszawa 2000,
- [5] GIL J., *Przykłady zastosowań sieci neuronowych w geodezji*, Oficyna Wydawnicza Uniwersytetu Zielonogórskiego, Zielona Góra 2006,
- [6] ŁĘSKI J., *Systemy neuronowo – rozmyte*, Wydawnictwo Naukowo-Techniczne, Warszawa 2008,
- [7] MARKOWSKA-KACZMAR U., *Ekstrakcja reguł z sieci neuronowych. Podejście ewolucyjne*, Oficyna Wydawnicza Politechniki Wrocławskiej. Wrocław 2006,
- [8] OSOWSKI S., *Sieci neuronowe do przetwarzania informacji*, Oficyna Wydawnicza Politechniki Warszawskiej, Warszawa 2006,
- [9] ZADEH L.A., *Fuzzy sets*. Information and Control, 1965,
- [10] ŻURADA J., BARSKI M., JĘDRUCH W.: *Sztuczne sieci neuronowe. Podstawy teorii i zastosowania*, Wydawnictwo Naukowe PWN, Warszawa 1996,

## METHODS OF ARTIFICIAL INTELLIGENCE USED FOR TRANSFORMING A SYSTEM OF COORDINATES

### Summary

The article discusses the problem of transformation in the form of the function  $f: R^2 \rightarrow R^2$  as a dependence between the coordinates  $[x, y]$  of the original system and the coordinates  $[X, Y]$  of the secondary system. The task of the transformation of  $[X, Y] = f(x, y)$  technically understood as the transformation of a system of coordinates has been solved by means of one direction two layer neural networks of the sigmoidal type, radial neural networks, recurrent cascade neural networks, and neuro-fuzzy systems with the use of the Takagi-Sugeno-Kang model. The numerical procedures applied make it possible to obtain a level of accuracy of the task equivalent to the cartographic accuracy of pictures in the Land Information Systems.

**Keywords:** neural networks, fuzzy networks, coordinate transformation