

Stanisław Płaczek  
Akademia Finansów i Biznesu Vistula – Warszawa

## ZASTOSOWANIE TEORII SYSTEMÓW HIERARCHICZNYCH DO ANALIZY SZTUCZNYCH SIECI NEURONOWYCH

### Streszczenie

Sztuczne Sieci Neuronowe (SSN) okazały się wygodnym narzędziem, przydatnym przy realizacji bardzo wielu różnych praktycznych zadań inżynierskich, ekonomicznych finansowych, medycznych i innych. SSN mogą być zastosowane tam, gdzie pojawiają się problemy z przetwarzaniem i analizą danych, prognozą, klasyfikacją czy sterowaniem. Sukces spowodowany jest tym, że w tych zastosowaniach SSN pełni rolę uniwersalnego aproksymatora nieliniowej, wektorowej funkcji wielu zmiennych. Podstawowym problemem jest efektywne uczenie złożonej konfiguracji sieci, jaką niewątpliwie jest struktura wielowarstwowej sieci neuronowej o wielu wejściach i wyjściach. Uczenie polega na poszukiwaniu minimum globalnej funkcji celu, którą najczęściej definiujemy jako błąd średniokwadratowy wyjścia sieci i wartości zadanej. Zadanie nie jest trywialne i ze względu na wielowymiarowość wektorów wejścia i wyjścia oraz wielowarstwowość sieci. Z tego też względu szuka się rozwiązań w sieciach o strukturze z jedną warstwą ukrytą. W celu wykorzystania możliwości sieci wielowarstwowych, do analizy złożonych struktur zastosowano metody i techniki opracowane dla wielowarstwowych, hierarchicznych struktur technicznych. Systemy hierarchiczne występują nie tylko w przyrodzie, lecz również w organizacjach ludzi. Tego typu struktury są bardzo efektywne z punktu widzenia zarządzania i kierowania organizacjami. Z systemami hierarchicznymi związane są zagadnienia dekompozycji dużego, podstawowego systemu na podsystemy oraz umiejętne skoordynowanie rozwiązań częściowych, w celu otrzymania rozwiązania optymalnego dla całego systemu. W artykule przedstawiono próbę zastosowania dekompozycji oraz koordynacji w stosunku do SSN o złożonej, wielowarstwowej strukturze. Dekomponując strukturę sieci oraz algorytm uczenia na podzadania, analizuje się wymagania, które musi spełnić algorytm w celu efektywnej koordynacji rozwiązań częściowych. Tak więc problem koordynacji jest problemem centralnym w analizie i konstrukcji algorytmu uczenia SSN. Artykuł ma charakter koncepcyjny.

**Słowa kluczowe:** Sztuczne Sieci Neuronowe, hierarchiczne struktury, dekompozycja, koordynacja, systemy złożone.

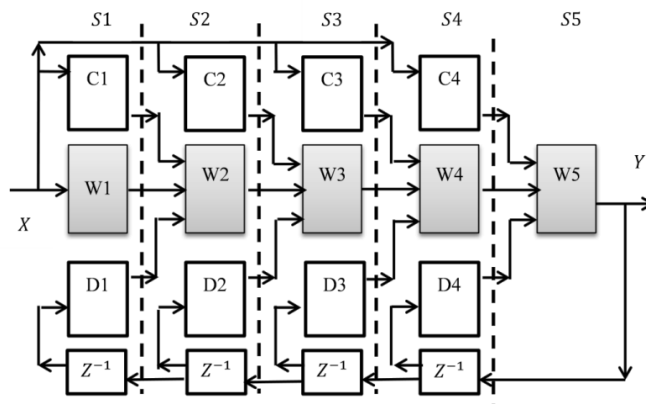
**Kody JEL:** C45

## Struktura SSN

Spotykane w praktycznych rozwiązaniach i w artykułach struktury SSN są bardzo zróżnicowane i trudno podać pełny schemat opisujący wszystkie konfiguracje. Nas interesują struktury Sztucznych Sieci Neuronowych, w których można wyróżnić trzy podstawowe słoje przetwarzania informacji (por. schemat 1):

- Słój centralny zwany również podstawowym (*Forward Connection*). Wektor wejściowy  $X$  jest przetwarzany przez warstwę wejściową, wszystkie warstwy ukryte oraz wyjściową w celu obliczenia wartości wektora wyjściowego  $Y$ . Wszystkie warstwy reprezentowane są przez macierze, dla  $i=1,2,\dots,n$ , gdzie  $n$  – numer warstwy wyjściowej.
- Słój górny zwany wyprzedzającym (*Cross Connection*). Wektor wejściowy  $X$  dostarczany jest przez macierze do wszystkich warstw ukrytych i warstwy wyjściowej. Wprowadzenie do struktury sieci słoja wyprzedzającego pozwala na szybsze przekazywanie zmian wartości wektora wejściowego  $X$  wzdłuż podstawowych warstw SSN. Poprawia się dynamika uczenia sieci. Z drugiej strony, komplikuje się algorytm uczenia sieci np. algorytm wstecznej propagacji błędów (*Back Propagation Algorithm*).
- Słój dolny zwany wstecznym (*Back Connection*). Tym razem wektor wyjściowy  $Y$ , przez takty opóźnienia oraz macierze dostarczany jest do poszczególnych warstw SSN. Wprowadzenie sygnału sprzężenia zwrotnego pozwala na realizację neuronowych nieliniowych obiektów dynamicznych.

**Schemat 1. Schemat trójsłojowej wielowarstwowej SSN**



Źródło: opracowanie własne

W opisie trójsłojowej SSN używa się pojęcia warstwy jako zbioru macierzy i funkcji aktywacji, którą najczęściej w warstwach ukrytych jest funkcja sigmoidalna unimodalna lub bipolarna. Wydziela się :

- warstwę wejściową, której rozmiar integralnie związany jest z rozmiarem wektora wejściowego  $X$ . Warstwa wejściowa przyjmuje dane od świata zewnętrznego (środowiska), może dokonać transformacji danych, jak normalizowanie, filtrowanie, przekształcenie. Liczba neuronów w warstwie wejściowej jest równa wymiarowi wektora danych wejściowych  $X$ . Warstwa wejściowa przesyła dane do pierwszej warstwy ukrytej.
- warstwy ukryte, które dokonują właściwej operacji nad danymi wejściowymi. W praktycznych zastosowaniach mamy jedną lub więcej warstw ukrytych. Jednym z najtrudniejszych problemów jest optymalizacja struktury SSN polegająca na określeniu liczby warstw ukrytych i dystrybucji neuronów między warstwami w taki sposób, aby rozwiązać problem w sposób optymalny.
- warstwa wyjściowa związana jest z wymiarowością wektora wyjściowego  $Y$ . Warstwa ta przyjmuje sygnały z ostatniej warstwy ukrytej i dokonuje sumowania w celu otrzymania odpowiedniego sygnału wyjściowego. Dla zadań aproksymacji przyjmuje się liniową funkcję aktywacji, natomiast dla zadań klasyfikacji standardową sigmoidalną.

W publikacjach spotyka się różne definicje rozmiaru SSN w aspekcie liczby warstw. Do liczby warstw włącza się lub nie warstwę wejściową. Tak więc te same struktury u różnych autorów mogą mieć różne wymiary. Dla uniknięcia nieporozumienia do rozmiaru sieci zaliczać będziemy wszystkie warstwy ukryte oraz warstwę wejściową.

Użycie pojęcia warstwy może być rozważane jako milczące założenie, że struktura SSN jest hierarchiczna. Opierając się na powyższym, do opisu i obliczeń wprowadzimy kilka pojęć związanych z warstwami, czyli wewnętrzną hierarchią SSN. Termin „warstwa”, jako pierwotny, używać będziemy tylko i wyłącznie w przypadkach nie budzących wątpliwości.

## Dekompozycja SSN

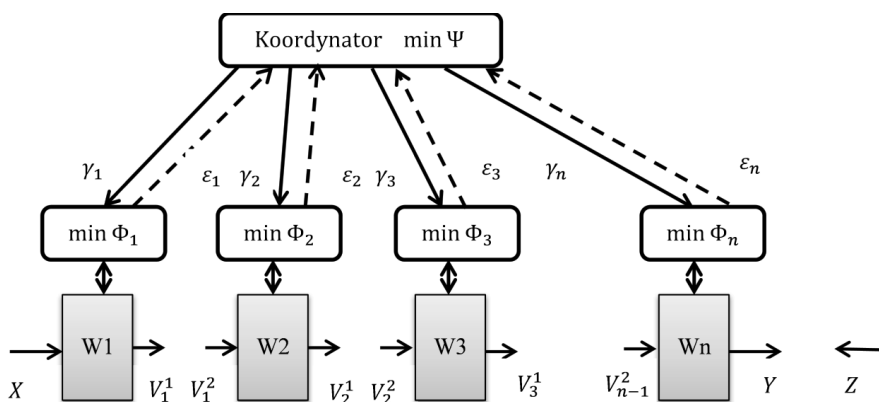
Złożonych systemów technicznych jak i przyrodniczych nie można opisać dokładnie i precyzyjnie używając pojęć i terminologii tylko z jednej dziedziny (Mesarovic i in. 1970). Problemem jest konflikt między prostotą opisu a dokładnością. Poniżej przedstawiono dwa różne sposoby opisy hierarchicznej struktury SSN.

Wprowadzając pojęcie straty lub stratyfikowanego opisu przyjmujemy zbiór modeli z różnych dziedzin nauki i techniki. Każdy model używa swoich zmiennych i terminologii o różnym poziomie abstrakcji. Tak więc dla holistycznego zrozumienia SSN należy zapoznać się z koncepcjami i pojęciami stosowanymi w każdej stracie. Przejrzyste zhierarchizowanie pojęć ułatwia zrozumienie procesu analizy lub syntezy SSN.

Przeprowadzając identyfikację nieliniowych stacjonarnych systemów z użyciem SSN wystarczy tylko raz przeprowadzić proces uczenia sieci na podstawie dostępnych danych wejściowych i wyjściowych. Strojenie parametrów sieci odbywa się w słoju optymalizacji minimalizującym błąd średniokwadratowy. Sytuacja ulega skomplikowaniu w przypadku procesów niestacjonarnych, w których wartości parametrów mogą zmieniać się w czasie. W miarę upływu czasu, SSN z coraz z mniejszą dokładnością odwzorowuje pierwotny niestacjonarny proces. Należy dokonywać ciągłej korekty parametrów przez wprowadzenie kolejnego poziomu podejmowania decyzji – słoju adaptacji. Dla bardziej złożonych zagadnień następny słoju samoorganizacji będzie miał zastosowanie. Wielo-słojowa struktura podejmowania decyzji będzie optymalną konfiguracją w przypadku nieliniowych, niestacjonarnych procesów.

Dotychczas wprowadzone pojęcia straty oraz słoja realizują pionową dekompozycję pojęć i algorytmów uczenia. Nie pokazują w sposób przejrzysty konkretnej struktury lub koncepcji algorytmu uczenia. W tym też celu wprowadzamy pojęcie eszelonu jako opisu dwuwarstwowej koncepcji algorytmu uczenia SSN (por. schemat 2). Na pierwszym poziomie, znajdują się autonomiczne podsystemy z lokalnymi funkcjami celu  $\Phi_1, \Phi_2, \dots, \Phi_n$ . Celem każdego podsystemu jest minimalizacja lokalnej funkcji celu przez iteracyjną modyfikację współczynników macierzy w SSN podzielonej na warstwy. Natomiast na drugim poziomie umieszczamy koordynator z własną funkcją celu  $\Psi$ . Głównym, chociaż nie jedynym celem koordynatora, jest takie skoordynowanie podsystemów pierwszego poziomu, żeby osiągnąć końcowy cel algorytmu uczenia, czyli zminimalizować globalną funkcję celu  $\Phi$ .

**Schemat 2. Eszelonowana struktura algorytmu uczenia SSN**



Źródło: jak w schemacie 1.

Tak więc przez jawną dekompozycję SSN na warstwy, można zaproponować nową efektywną strukturę algorytmu uczenia opartą na koordynacji podsystemów, czyli koordynacji podzadań pierwszego poziomu.

## Zasady koordynacji

Zadanie koordynacji nie jest prostym zadaniem. Powyższe wynika z kilku przesłanek, a mianowicie:

- SSN oraz algorytmy uczenia w swojej podstawowej strukturze są zadaniami nieliniowymi, które rozwiązuje się metodami iteracyjnymi.
- Są to zadania wielowymiarowe, gdzie wymiary wektorów wejściowych ukrytych, jak i wyjściowych mogą być naprawdę duże.
- Dekomponując podstawową strukturę SSN w sposób jawny na warstwy i przypisując lokalnym podsystemom funkcje celu, wprowadzamy podwójną sytuację konfliktową między podzadaniami pierwszego poziomu oraz konflikt między poziomami – pierwszego poziomu i koordynatorem. W pierwszym przypadku mówimy o konflikcie wewnętrznym poziomu pierwszego, natomiast drugi to konflikt między poziomami w wewnętrznej strukturze.

Głównym zadaniem koordynatora jest więc niedopuszczenie do powstania konfliktów, a w przypadku zaistnienia, koordynator musi podjąć decyzje (rozwiązania) usuwające przyczynę konfliktu. W celu znalezienia przyczyn konfliktu, definiujemy:

- Globalną funkcję celu uczenia SSN, która jest zależna nie tylko od wektorów wejściowego  $X$  i uczenia  $Z$ , lecz również od całej struktury sieci wyrażonej zbiorem macierzy  $W = (W_1, W_2, \dots, W_n)$ , czyli  $\Phi(X, Z, W)$ ,
- Funkcją celu koordynatora  $\Psi$ , która zależy od sygnałów sprzężenia zwrotnego  $\varepsilon = (\varepsilon_1, \varepsilon_2, \dots, \varepsilon_n)$ , jak również od wypracowanych przez koordynatora sygnałów koordynujących  $\gamma = (\gamma_1, \gamma_2, \dots, \gamma_n)$ , czyli  $\Psi(\varepsilon, \gamma)$ . Sygnały sprzężenia zwrotnego wypracowane są w każdej iteracji przez podsystemy pierwszego poziomu i przesyłane do koordynatora (por. schemat 2).
- Zbiór funkcji celu podsystemów pierwszego poziomu  $\Phi_1, \Phi_2, \dots, \Phi_n$ . Każda funkcji  $\Phi_i(\gamma_i)$ , dla  $i = 1, 2, \dots, n$ , zależy od swojego wektora wejściowego i wyjściowego, macierzy podsystemu oraz koordynującego parametru  $\gamma_i$ , (por. schemat 2). Koordynator, w każdej iteracji, na podstawie swojej własnej funkcji celu  $\Psi$  oraz sygnału sprzężenia zwrotnego  $\varepsilon$  oblicza nowe, lepsze wartości sygnału koordynującego  $\gamma$ .

Otwarte pozostaje pytanie, jaką strategię powinien zastosować koordynator, wypracowując w iteracyjnym procesie wymiany informacji między podsystemami pierwszego poziomu a koordynatorem, nowe wartości wektora koordynującego  $\gamma(n+1)$  na podstawie wektora błędu  $\varepsilon(n)$ . W teorii systemów hierarchicznych, zaaprobowano trzy zasady koordynacji oraz zdefiniowano warunki jakie muszą spełniać wszystkie podsystemy w celu rozwiązania konfliktów.

## Koordinacje względem funkcji celu koordynatora

Relacje między koordynatorem a pierwszym poziomem podzadań w dwu-poziomym algorytmie uczenia SSN zdefiniowane są przez wymianę informacji za pomocą sygnału koordynującego  $\gamma = (\gamma_1, \gamma_2, \dots, \gamma_n)$ , a sygnałem sprzężenia zwrotnego  $\varepsilon = (\varepsilon_1, \varepsilon_2, \dots, \varepsilon_n)$ .

Logiczną zależność koordynacji względem funkcji celu koordynatora najlepiej wyrazić za pomocą predykatora (Mesarovic i in. 1970).

$$(\exists \gamma) (\exists W) [\mathbf{P}(W, (\Phi_1(\gamma_1), (\Phi_2(\gamma_2), \dots, (\Phi_n(\gamma_n)))) \text{ i } \mathbf{P}(\gamma, \Psi(\gamma, \varepsilon))] \quad (1)$$

Wyrażenie (1) stwierdza, że koordynacja względem funkcji celu koordynatora osiągnięta jest wtedy, kiedy koordynator osiąga swoje optymalne rozwiązanie i wypracowuje odpowiedni sygnał koordynujący  $\gamma$ , który z kolei jest wykorzystany przez podzadania pierwszego poziomu  $\Phi_1(\gamma_1), (\Phi_2(\gamma_2), \dots, (\Phi_n(\gamma_n)$ . Podzadania te mają swoje optymalne rozwiązania, wyrażone za pomocą zbioru macierzy  $W = (W_1, W_2, \dots, W_n)$ . Jaka jest konkretna konstrukcja funkcji celu koordynatora, zależy od wybranego prawa koordynacji.

## Koordinacja względem globalnej funkcji celu

Domyślnie założyliśmy, że nie istnieje żadna jawna relacja między globalną funkcją celu  $\Phi(X, Z, W)$ , lokalnymi funkcjami celu  $\Phi_1(\gamma_1), (\Phi_2(\gamma_2), \dots, (\Phi_n(\gamma_n)$  a funkcją celu koordynatora  $\Psi(\gamma, \varepsilon)$ . Z drugiej strony oczekujemy, że zarówno koordynator, jak i lokalne podzadania pierwszego poziomu będą działały zgodnie w celu osiągnięcia globalnej funkcji celu. „Działały zgodnie” oznacza, że podzadania pierwszego poziomu znajdą takie wartości współczynników macierzy we wszystkich warstwach  $W = (W_1, W_2, \dots, W_n)$  że globalna funkcja celu  $\Phi(X, Z, W)$  również osiąga swoje minimum. Powyższe stwierdzenia można ująć w następującym predykanie.

$$(\exists \gamma) (\exists W) [\mathbf{P}(W = (W_1, W_2, \dots, W_n), (\Phi_1(\gamma_1), (\Phi_2(\gamma_2), \dots, (\Phi_n(\gamma_n))) \text{ i } \mathbf{P}(W, \Phi(X, Z, W))] \quad (2)$$

Powyższe dwie zasady koordynacji zawsze łączą tylko dwie funkcje celu. W pierwszym przypadku analizujemy relacje między podzadaniami pierwszego poziomu a koordynatorem. W drugim przypadku uwzględniamy globalną funkcję celu i jej relacje z podzadaniami pierwszego poziomu. Intuicyjnie wyczuwamy, że musi istnieć pewnego rodzaju spójność, zgodność między powyższymi funkcjami celu. Warunki te definiowane są przez postulat zgodności.

## Postulat zgodności

W celu osiągnięcia zgodności działania dwupoziomowego algorytmu uczenia SSN, między wyżej zdefiniowanymi funkcjami celu musi istnieć pewna wewnętrzna harmonia, z uwzględnieniem następujących przesłanek:

- Bezpośredni kontakt z macierzami  $W = (W_1, W_2, \dots, W_n)$  realnej SSN mają tylko podsystemy pierwszego poziomu. W celu spełnienia globalnej funkcji celu  $\Phi$ , zasada wyrażona wzorem (2) musi być spełniona.
- Koordynator przez sygnały koordynujące  $\gamma = (\gamma_1, \gamma_2, \dots, \gamma_n)$ , oddziałuje na lokalne podzadania w taki sposób, aby spełniona została funkcja celu koordynatora. Tak więc zasada wyrażona wzorem (1) również musi być spełniona.
- Globalne zadanie wyrażone przez globalną funkcję celu  $\Phi$  jest zdefiniowane na zewnątrz dwupoziomowego algorytmu uczenia SSN. Żadne z podzadań pierwszego oraz drugiego poziomu nie jest w stanie rozwiązać globalnego zadania.

Ten konflikt można rozwiązać tylko przez odpowiednią wewnętrzną relację między zadaniami, wyrażoną wzorem (3), którą nazywamy postulatem zgodności w dwupoziomowym algorytmie uczenia SSN.

$$\begin{aligned}
 (\forall \gamma) (\forall W) [\mathbf{P}(W, (\Phi_1(\gamma_1), (\Phi_2(\gamma_2), \dots, (\Phi_n(\gamma_n)))) \text{ i } \mathbf{P}(\gamma, \Psi(\gamma, \varepsilon))] \Rightarrow \\
 \Rightarrow [\mathbf{P}(W = (W_1, W_2, \dots, W_n), (\Phi_1(\gamma_1), (\Phi_2(\gamma_2), \dots, (\Phi_n(\gamma_n)))) \\
 \text{ i } \mathbf{P}(W, \Phi(X, Z, W))] \quad (3)
 \end{aligned}$$

Postulat ten prosto stwierdza za pomocą bardzo ogólnych predykatów, że globalna funkcja celu zostanie zawsze zrealizowana przez dobór, we wszystkich warstwach SSN, współczynników macierzy  $W$  [druga część wzoru (3)], wtedy i tylko wtedy, kiedy koordynator wypracuje takie koordynujące sygnały, że rozwiązania lokalnych podzadań znajdą wartości swoich macierzy  $W = (W_1, W_2, \dots, W_n)$ , które okażą się optymalne z punktu widzenia globalnej funkcji celu [pierwsza część wzoru(3)]. Doświadczenie pokazuje, że teoretyczne spełnienie postulatów zgodności, szczególnie dla zadań z nieliniowymi funkcjami celu, nie zawsze jest możliwe. Tym bardziej zastosowanie dwupoziomowej optymalizacji jest możliwe i osiąga się celowe rozwiązanie.

## Prawa koordynacji

Dla wielowarstwowych, hierarchicznych systemów (Mesarovic i in. 1970), definiuje się trzy prawa koordynacji:

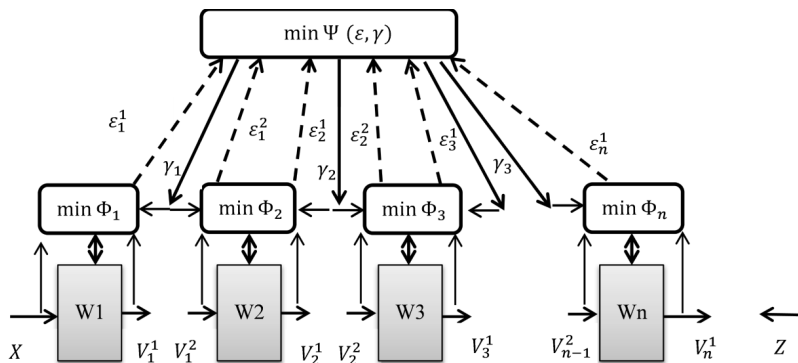
- Predykcja (prognoza) wektorów powiązań (interfejsów) między warstwami SSN. Tak więc jednym z zadań koordynatora jest takie określenie wartości wektorów koordynacji  $\gamma = (\gamma_1, \gamma_2, \dots, \gamma_n)$ , aby rzeczywiste wartości sygnałów międzywarstwach  $(V_1^1, V_1^2, V_2^1, V_2^2, \dots, V_{n-2}^2, V_{n-1}^2)$  były równe wartościom

- prognozowanym (por. schemat 3). Koordynator, prognozując wartości powiązań, oddziałuje na podzadania pierwszego poziomu w małej skali.
- Rozwiązywanie (uwolnienie) wektorów powiązań między warstwami. Przyjmuje się, że podzadania pierwszego poziomu są maksymalnie niezależne przez pełne uwolnienie interfejsów. Podzadania pierwszego poziomu muszą optymalizować swoje funkcje celu przez dobór nie tylko współczynników macierzy, lecz również wartości interfejsów. W tym miejscu warto podkreślić, że koordynator może oddziaływać na podzadania pierwszego poziomu tylko przez wartości lokalnych funkcji celu. Ten sposób koordynacji nazywany jest również koordynacją w dużej skali.
  - Estymacja powiązań wartości wektorów między warstwami. To prawo koordynacji jest rozszerzeniem prawa pierwszego, w którym koordynator zadaje dokładne wartości prognozowanych powiązań. Tym razem koordynator zwiększa swobodę wyboru wartości interfejsów przez określenie przedziałów, w których podzadania pierwszego poziomu wybierają wartości powiązań.

### Implementacja prawa prognozowania powiązań w SSN

Realizując dekompozycję SSN, a następnie proponując eszelonową strukturę algorytmu koordynacji (por. schemat 2), przeprowadzono podział SSN na podsieci rozdzielając wektory wyjściowe i wejściowe każdej podsieci. Przyjęto, że wektory nie są powiązane i każdy z nich może przyjmować wartości zgodnie z określoną strategią koordynacji. Najbardziej naturalnym prawem wydaje się wybranie pierwszego prawa – prawo predykcji wartości interfejsów między warstwami (por. schemat 3).

**Schemat 3. Struktura algorytmu koordynacji na podstawie prawa predykcji wartości interfejsów**



Źródło: jak w schemacie 1.



Sygnal koordynacji zadawany przez koordynator drugiego poziomu, prognozując wartości powiązań interfejsów między warstwami, spełnia jednocześnie dwie funkcje:

- Dla lewej podsieci (czyli dla wektora wyjściowego podsieci wejściowej i każdej ukrytej) przekazuje do funkcji celu wartość zadaną wektora wyjściowego. Tak więc sygnał koordynujący precyzuje, jakie wartości powinny osiągnąć wektory wyjściowe danej podsieci.

Dla podsieci wejściowej, definiujemy lokalną funkcję celu:

$$\Phi_1(W1, X, \gamma) = \frac{1}{2} \sum_{i=1}^{N_1} \sum_{p=1}^{N_p} (v1_i^p - \gamma_i^p)^2 = \frac{1}{2} \sum_{i=1}^{N_1} \sum_{p=1}^{N_p} (f(\sum_{j=0}^{N_0} W1_{ij} \cdot x_j) - \gamma_i^p)^2 \quad (4)$$

gdzie:

$\gamma[1:N_1, 1:N_p]$  – Sygnał koordynatora dla całej paczki,

$N_1$  – Ilość nefronów wyjściowych pierwszej podsieci,

$N_0$  – Ilość neuronów wejściowych pierwszej podsieci.

Tak więc podsieć przyjmuje tylko jedną wartość prognozy dla sygnału wyjściowego.

Dla podsieci ukrytej, lokalna funkcja celu przyjmuje postać:

$$\Phi_u(Wu, \gamma_{u-1}, \gamma_u) = \frac{1}{2} \sum_{i=1}^{N_1} \sum_{p=1}^{N_p} (v1_i^p - \gamma_i^p)^2 = \frac{1}{2} \sum_{i=1}^{N_1} \sum_{p=1}^{N_p} (f(\sum_{k=0}^{N_{n-1}} W2_{ki} \cdot \gamma_k^p) - \gamma_i^p)^2 \quad (5)$$

gdzie:

oznaczenia jak wyżej.

W tym przypadku, podsieć otrzymuje dwie wartości prognozy. Sygnał wejściowy  $\gamma_{u-1}$  oraz wyjściowy  $\gamma_u$ .

- Dla prawej podsieci (dla każdej warstwy ukrytej oraz warstwy wyjściowej) przekazuje do funkcji celu, a zarazem na wejście danej podsieci, wartości sygnału wejściowego. Zadana wartość wektora wejściowego pozwala na obliczenie wartości wektora wyjściowego i porównane go z zadaną wartością wyjścia.

Dla podsieci wyjściowej, lokalna funkcja celu przyjmuje postać:

$$\Phi_n(W2, Z, \gamma) = \frac{1}{2} \sum_{k=1}^{N_n} \sum_{p=1}^{N_p} (v2_k^p - z_k^p)^2 = \frac{1}{2} \sum_{k=1}^{N_n} \sum_{p=1}^{N_p} (f(\sum_{i=0}^{N_{n-1}} W2_{ki} \cdot \gamma_i^p) - z_k^p)^2 \quad (6)$$

gdzie:

oznaczenia jak wyżej.

We wszystkich wzorach powyżej przez  $f(\cdot)$  oznaczono stosowną funkcję aktywacji, nieliniową typu sigmoidalnego lub liniową dla podsieci wyjściowej.

Oczywiście, koordynator w każdej iteracji prognozując wartości powiązań  $\gamma(n)$  może w sposób nieprecyzyjny zadawać, dla poszczególnych podsieci, wartości wektorów powiązań. Podsieci pierwszego poziomu, realizując swoje własne funkcje celu, wykazują określone różnice występujące między prognozą koordynatora a realnymi wartościami wektorów wejściowych i wyjściowych otrzymanych w każdej podsieci –  $\epsilon(n)$ . Wartości te przekazywane do koordynatora umożliwiają wypracowanie nowych, lepszych wartości sygnałów prognozy.

### Algorytm koordynatora

Jak już podkreślono, koordynator nie ma bezpośredniego kontaktu z SSN. Może oddziaływać na wartości wewnętrznych parametrów SSN przez podsieci pierwszego poziomu. Do tego celu, wykorzystując prawo prognozowania powiązań oraz prawo priorytetu oddziaływania, narzuca w sposób arbitralny podsieciom pierwszego poziomu wartości wejść i wyjść. Niestety, jego prognozy nie zawsze są trafne, podsieci wykorzystując sygnały sprzężenia zwrotnego informują koordynator o rozbieżnościach między prognozą i realnymi wartościami powiązań.

Jako jedną z funkcji celu koordynatora, wybiera się sumaryczny błąd średniokwadratowy dla wszystkich podsieci i całej epoki.

$$\Psi = \frac{1}{2} \sum_{p=1}^{N_p} \sum_{i=1}^{N_i} (\epsilon 1_i^p - \gamma_i^p)^2 + \frac{1}{2} \sum_{p=1}^{N_p} \sum_{i=1}^{N_i} (\gamma_i^p - \epsilon 2_i^p)^2 \quad (7)$$

Stosując metodę gradientową minimalizacji funkcji celu koordynatora, otrzymujemy:

$$\frac{\partial \Psi}{\partial \gamma_i^p} = 2 \cdot \gamma_i^p - (\epsilon 1_i^p - \epsilon 2_i^p) \quad (8)$$

Wartość sygnału koordynatora w następnej iteracji

$$\gamma_i^p(n+1) = \gamma_i^p(n) - \lambda_1 \cdot \frac{\partial \Psi}{\partial \gamma_i^p} \quad (9)$$

gdzie:

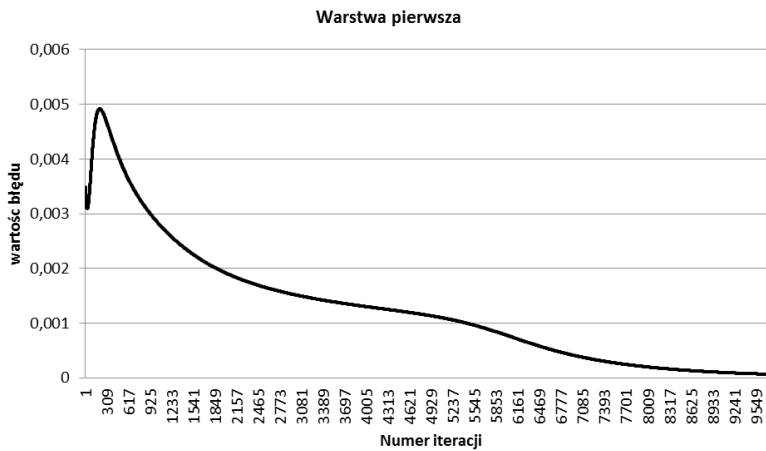
$\lambda_1$  – współczynnik uczenia dla koordynatora.

### Przykład numeryczny

W przykładzie numerycznym nacisk będzie położony na charakterystyki lokalnych funkcji celu pierwszego poziomu oraz koordynatora. Z przedstawionej teorii wynika, że to właśnie koordynator musi zapewnić pełną syn-

chronizację poszukiwania minimum błędów średniokwadratowych, które zostały zdefiniowane w równaniach (4), (5), (6) oraz dla koordynatora (7). Na wejście i wyjście wielowarstwowej SSN, o liczbie warstw ukrytych większej niż jedna, podawane są odpowiednio wektory wejściowe oraz wyjściowe (wektor nauczyciela). Koordynator rozpoczyna proces uczenia SSN zadając początkowe wartości powiązaniom międzywarstwowym (interfejsom) przesyłając je do podsieci pierwszego poziomu. Podsieci poszukują optymalnych wartości swoich współczynników macierzy  $W$  oraz błędów sprzężenia zwrotnego  $\varepsilon(n)$ , które przesyłają do koordynatora. W kroku następnym koordynator oblicza nowe wartości powiązań (9) (lepsze od poprzednich!) i cały proces powtarza się tak długo, aż podsieci pierwszego poziomu oraz koordynator osiągną zadane minimalne wartości błędów. Na wykresie 1 pokazano, jak zmienia się błąd średniokwadratowy w pierwszej podsieci. Można zauważyć pewne opóźnienie w stosunku do błędu przedstawionego na wykresie 2.

**Wykres 1. Wartości lokalnej funkcji celu dla pierwszej warstwy w funkcji numeru iteracji**



Źródło: opracowanie własne.

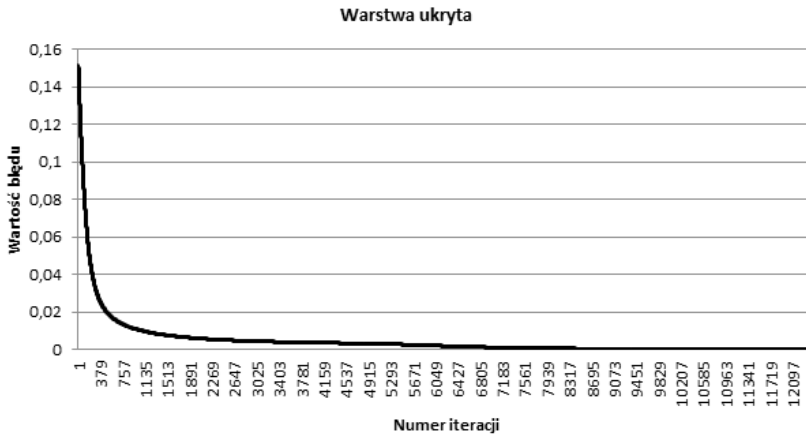
Wszystkie warstwy ukryte szybciej osiągają małe wartości funkcji celu. W granicach 5000 iteracji wartość błędu uczenia danej warstwy nie jest większa niż 1%.

Jakościowe i ilościowe różnice w zachowaniu się lokalnych funkcji celu można wytłumaczyć na dwa sposoby:

- Wartość błędu w warstwy wyjściowej musi zostać przekazana przez koordynator do warstwy pierwszej, przechodząc przez wszystkie warstwy ukryte. Występuje więc pewne opóźnienie czasowe, co ma wpływ na szybkość ustalania się wartości współczynników macierzy  $W1$ .

- Zaproponowany algorytm dla koordynatora jest prosty i nie wyróżnia w swojej strukturze żadnej podsieci. Należy przeprowadzić badania nad zmienną strukturą algorytmu uczenia w aspekcie każdej podsieci.

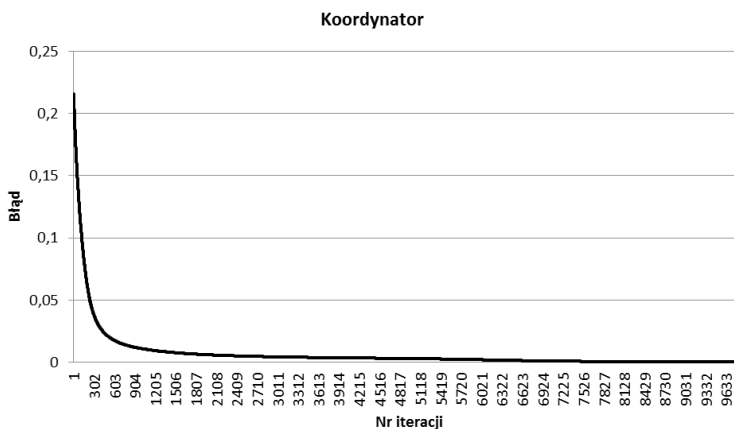
**Wykres 2. Wartość lokalnej funkcji celu dla warstwy ukrytej w funkcji numeru iteracji**



Źródło: jak w wykresie 1.

Na wykresie 3 przedstawiono charakterystykę błędów uczenia dla koordynatora. Jest to funkcja monotonicznie malejąca która w sposób asymptotyczny dąży do zera. Jest zgodna z postulatem zgodności i pewnej monotonicznej zależności między wszystkimi podzadaniami w SSN.

**Wykres 3. Wartość funkcji celu koordynatora w funkcji numeru iteracji**



Źródło: jak w wykresie 1.

W rozpatrywanym przykładzie struktura SSN jest bardzo prosta, zawiera tylko warstwę *Forward Connection* (środkową ze schematu 1.). Chcąc przyspieszyć algorytm uczenia należy zastosować bardziej skomplikowane struktury SSN przekazujące wartość wektora wejściowego  $X$  do warstw ukrytych i wyjściowej.

## Podsumowanie

Przedstawiona koncepcja zdekomponowania struktury SSN i algorytmu uczenia na podstawie teorii systemów hierarchicznych i wypracowanych praw koordynacji jest bardzo elastyczna i może być stosowana dla wielu struktur SSN. Umożliwia jednocześnie wykorzystanie współczesnych narzędzi programowania wielowątkowego, a tym samym przyspieszenia procesu uczenia przez pełne zrównoleglenie algorytmu. W dalszych opracowaniach nacisk należy położyć na poszukiwanie efektywnych algorytmów koordynacji. Należy wykorzystać doświadczenia teorii regulacji przez zastosowanie PID regulatora.

## Bibliografia

- Bishop Ch.M. (2006), *Pattern Recognition and Machine Learning*, Springer Science + Business Media, LLC.
- Findeisen W., Szymanowski J., Wierzbicki A. (1977), *Teoria i metody obliczeniowe optymalizacji*, PWN, Warszawa.
- Hou Zeng-Guang, Gupta M.M., P.N. Nikiforuk, Tan M., Cheng L. (2007), *A Recurrent Neural network for Hierarchical Control of Inyterconnected Dynamic Systems*, "IEEE Transactions on Neural Networks", Vol. 18, No. 2, March.
- Marciniak A., Korbicz J., Kus J. (2000), *Wstępne przetwarzanie danych*, „Sieci Neuronowe”, tom 6, Akademicka Oficyna Wydawnicza EXIT.
- Mesarovic M.D., Macko D., Takahara Y. (1970), *Theory of hierarchical, multilevel systems.*, Academic Press, New York and London.
- Osowski S. (2006), *Sieci Neuronowe do Przetwarzania Informacji*, Oficyna Wydawnicza Politechniki Warszawskiej, Warszawa.
- Płaczek S., Adhicari B. (2013), *Analysis of Multilayer Neural Network with Direct and Cross – Forward Connection*, CS7P Conference in the University of Warsaw, Warsaw.
- Płaczek S. (2014), *Hierarchiczne struktury w sztucznych sieciach neuronowych*, „Zeszyty Naukowe Uczelni Vistula”, nr 38.

# Application of the Theory of Hierarchical Systems to Analyse Artificial Neural Networks

## Summary

Artificial neural networks (ANN) have appeared to be a convenient tool, useful for implementation of very many practical engineering, economic, financial, medical, and other tasks. ANN may be applied where the problems with data processing and analysis, forecast, classification or steering appear. The success is caused by the fact that in these applications ANN plays the role of universal approximator of the non-linear, vectored function of many variables. The basic problem is an effective teaching of the complex configuration of the network which, no doubt, the structure of multilayer neural network with many inputs and outputs is. Teaching consists in seeking for the minimum global function of the purpose, which is most often defined as a mean squared error of the network input and the set-point. The task is not trivial also due to the multidimensionality of vectors of input and output as well as due to the multilayer nature of the network. Also having this in mind, there are attempts to find solutions in networks with the structure with one hidden layer. In order to make use of the possibilities of multilayer networks, the author applied for the analysis of complex structures the methods and techniques developed for multilayer, hierarchical technical structures. Hierarchical systems take place not only in the nature but also in human organisations. Such structures are very effective from the point of view of organisation management and direction. The hierarchical systems are combined with the issues of decomposition of a big, basic system into subsystems and a skilful coordination of partial solutions in order to obtain a solution optimal for the entire system. In his article, the author presented an attempt to apply decomposition and coordination in relation to ANN with a complex, multilayer structure. Decomposing the network structure and the algorithm of teaching into subtasks, he analyses the requirements to be met by the algorithm for the purpose of effective coordination of partial solutions. Thus, the problem of coordination is the central problem in the analysis and construction of the ANN algorithm of teaching. The article is of the conceptual nature.

**Key words:** artificial neural networks, hierarchical structures, decomposition, coordination, complex systems.

**JEL codes:** C45

Artykuł nadesłany do redakcji w maju 2015 roku.

© All rights reserved

Afiliacja:  
dr inż. Stanisław Płaczek  
Akademia Finansów i Biznesu Vistula  
ul. Stokłosy 3  
02-787 Warszawa  
tel.: 22 457 23 00  
e-mail: [s.placzek@vistula.edu.pl](mailto:s.placzek@vistula.edu.pl)