Akademia Górniczo-Hutnicza im. Stanisława Staszica w Krakowie

Wydział Elektrotechniki, Automatyki, Informatyki i Inżynierii Biomedycznej

KATEDRA INFORMATYKI STOSOWANEJ



ROZPRAWA DOKTORSKA

MATEUSZ WÓJCIK

HYBRYDOWY SYSTEM NEURONALNY INTELIGENTNEGO MONITORINGU TURBINY WIATROWEJ

PROMOTOR: dr hab. Andrzej Bielecki, prof. AGH PROMOTOR POMOCNICZY: dr inż. Aleksander Dziadecki

Kraków 2016

AGH

University of Science and Technology in Krakow

Faculty of Electrical Engineering, Automatics, Computer Science and Biomedical Engineering

DEPARTMENT OF APPLIED COMPUTER SCIENCE



PHD THESIS

MATEUSZ WÓJCIK

HYBRID NEURAL SYSTEM FOR INTELLIGENT MONITORING OF WIND TURBINE

SUPERVISOR: Andrzej Bielecki, Ph.D., D.Sc. AUXILIARY SUPERVISOR: Aleksander Dziadecki, Ph.D

Krakow 2016

Serdecznie dziękuję Panu Profesorowi Andrzejowi Bieleckiemu za opiekę naukową i cenne wskazówki podczas realizacji niniejszej pracy Żonie za wsparcie, wyrozumiałość i cierpliwość

Streszczenie

Niniejsza dysertacja doktorska dotyczy detekcji potencjalnych usterek i stanów przedawaryjnych dla turbin wiatrowych. Tematyka skutecznego monitoringu i diagnostyki turbin wiatrowych podejmowana jest od wielu lat. Popularnie wykorzystywanymi w tym zakresie metodami przetwarzania danych są metody statystyczne, analizy trendów i analizy spektralne. Ponieważ większość systemów bazujących na tych metodach analizuje poszczególne dane pomiarowe niezależnie, co jest niewystarczające z uwagi na chaotyczny charakter oddziaływania wiatru na turbinę, zaczęto wykorzystywać również inne, inteligentne metody bardziej kompleksowego przetwarzania danych. W niniejszej dysertacji został wprowadzony nowatorski inteligentny system hybrydowy, który charakteryzuje się autonomicznością, kompleksowością przetwarzania wielowymiarowych danych w czasie rzeczywistym oraz faktem, że do wykonywania swoich zadań potrzebuje stosunkowo niewielkiej wiedzy wstępnej o monitorowanym obiekcie.

Zaproponowany w pracy system przetwarza strumień wielowymiarowych danych opisujących działanie poszczególnych komponentów turbiny wiatrowej i jej środowiska. Każdy wejściowy wielowymiarowy punkt danych kierowany jest do podsystemu VBMoG (Variational Bayssian Mixture of Gaussians), który wykorzystywany jest w początkowej fazie działania systemu do detekcji szczególnych punktów danych. Używany jest tu algorytm VBEM (Variational Bayssian Expectation-Maximization). Następnie dane przekazywane są do modułów MoG (Mixture of Gaussians), które pełniąc funkcję klasyfikatora OCC (One Class Classifier) mają na celu zapamiętywanie stabilnych stanów pracy maszyny. Jeden moduł MoG reprezentuje jeden ogólny stan operacyjny w jakim przez dłuższy czas znajdowała się maszyna. Początkowo system nie posiada żadnych modułów MoG, a każdy nowy dodawany jest poprzez użycie algorytmu EM (Expectation-Maximization), który wyznacza obszar obejmujący wszystkie jeszcze niesklasyfikowane, ale zarejestrowane już dane. Moduły MoG sprawdzają każdorazowo czy wejściowy punkt danych pasuje do jednego z wcześniej wyznaczonych obszarów. Jeśli tak, wówczas system przechodzi do następnego punktu danych. W przeciwnym przypadku punkt danych jest kierowany poprzez moduł realizujący dodatkowy preprocesing danych do sieci neuronowej ART-2 (Adaptive Resonance Theory). Sieć ART-2 jest główną składową całego systemu, której aplikowalność dla zadań monitoringu stanów operacyjnych turbiny wiatrowej została wykazana przez autora we wcześniejszych badaniach. Przewagą sieci ART-2 nad większością innych inteligentnych metod monitoringu jest nienadzorowany proces jej nauki. Daje to większą elastyczność systemu, nie wymagając kosztownego czasowo treningu na dużych zbiorach danych przygotowanych przez eksperta. Zadaniem sieci ART-2 w systemie jest klasteryzacja przetwarzanych lokalnych, bo ograniczonych przez moduły MoG, danych. System posiada proces kontrolny, który sprawdza czy liczba wyznaczonych przez sieć ART-2 klastrów zmienia się. Utrzymywanie się przez dłuższy czas jednakowej liczby klastrów oznacza, że maszyna jest w stabilnym stanie operacyjnym. Proces kontrolny wyzwala wówczas procedurę dodania nowego obszaru MoG i usunięcia lokalnych klastrów znalezionych przez sieć ART-2, przygotowując ją do dalszej pracy. Proces kontrolny sprawdza również czy liczba rozpoznanych przez sieć ART-2 klastrów mieści się w określonym przedziale, a jeśli nie - modyfikuje parametr czułości tej sieci, co zapewnia systemowi wrażliwość na zmianę charakteru danych wejściowych. Parametr czułości inicjowany jest przez podsystem VBMoG po określonym czasie działania systemu. Na wyjściu dla każdego punktu danych wejściowych system zwraca identyfikator rozpoznanego obszaru MoG lub identyfikator lokalnie przyporządkowanego przez sieć ART-2 klastra danych.

Przedstawiony w dysertacji system został zaimplementowany w środowisku MATLAB. Wykonano weryfikację jego działania przy użyciu posiadanych danych z monitoringu turbiny wiatrowej. Pokazano, że system może być traktowany jako w pełni samodzielny system monitorujący turbinę, działający jako system wczesnego ostrzegania. Może również być używany przez eksperta jako dodatkowe źródło informacji o bieżącym stanie operacyjnym, a także o archiwalnych ułożeniach stanów w jakich się znajdowała monitorowana turbina. System potrafi przetwarzać dowolne dane liczbowe, a dzięki temu nie musi być traktowany wyłącznie jako alternatywa dla innych systemów monitoringu turbin wiatrowych. Możliwa jest taka konfiguracja systemów monitoringu, w której przedstawiony w niniejszej dysertacji system zarządzałby informacjami przekazywanymi z systemów bazujących na innych metodach przetwarzania danych, co świadczy o uniwersalności stworzonego rozwiązania.

Abstract

This doctoral dissertation refers to a problem of detection of faults and pre-faults states of wind turbines. The subject of how to monitor and diagnose wind turbines has been considered for many years. Data processing methods widely used in this field are statistical methods, trend analysis and spectral analysis. Since most systems based on these methods analyze individual measurement data seperately, which is insufficient due to the chaotic nature of wind, there is a need to use more comprehensive, intelligent methods of data processing. This dissertation introduces an innovative intelligent hybrid system which is autonomic, can comprehensively process multidimensional data in real-time, and requires a relatively little initial knowledge of a monitored object.

The proposed system processes a stream of multi-dimensional data, describing an activity of wind turbine components and environment. After loading, each multi-dimensional point is directed to the VBMoG (Variational Bayssian Mixture of Gaussians) subsystem which is used in an initial phase of the system for detecting specific data points. The Variational Expectation-Maximization Bayssian (VBEM) algorithm is being used here. The data is then transmitted to the MoG (Mixture of Gaussians) modules that as a OCC (One Class Classifier) are aimed to remember some stable operating states of the machine. One MoG module can represent one general operating state in which the machine has been for a long time. Initially, the system does not have any MoG modules, and every new one is added through an use of the Expectation-Maximization (EM) algorithm, which designates an area covering all yet unclassified data points. Modules check whether each data point matches one of the pre-designated areas. If so, then the system takes a next data point. Otherwise, the data point is directed through a preprocessing module to the ART-2 (Adaptive Resonance Theory) neural network. The ART-2 network is a main component of the whole system. Its application to the task of monitoring a wind turbine operational states has been demonstrated by the author in his earlier studies. An advantage of ART-2 network over most other intelligent methods used for monitoring is an unsupervised learning process. It gives more flexibility to the system without requiring any time consuming training on large data sets prepared by an expert. In the system, the ART-2 network clusters local, as limited by the MoG modules, data. The system has a control process that checks whether the number of clusters designated by the ART-2 is changing. A long time maintanance of the same number of clusters indicates that the machine is in a stable operating state. In such a case, the control process triggers a procedure for adding a new MoG area and removes local clusters found by the ART-2 network, preparing it for further work. The control process also checks whether the number of clusters identified by the ART-2 network is within a specified range, and if not modifies a vigilance parameter of the network, which provides the system some receptivity for changes of the input data's nature. The vigilance parameter is initiated by the VBMoG subsystem after a given period of system operation. For each input data point the system returns an identifier of the recognized MoG area or an identifier of cluster locally assigned by the ART-2 network.

The system presented in this dissertation has been implemented in the MATLAB environment. A verification of its operation has been done using data from the wind turbine monitoring. It has been shown that the system can be regarded as a fully independent wind turbine monitoring system, acting as an early warning tool. It can also be used by experts as an additional source of information about

the current operating state, as well as about archival configurations of the states in which the monitored turbine was operating. The system can process any numerical data, and thus can be regarded not only as an alternative to other monitoring systems for wind turbines but also as a manager of information passed by systems based on any other data processing methods, what indicates the versatility of a created solution.

Spis treści

1.	Wpr	prowadzenie				
	1.1.	Motyv	vacja, teza i specyfikacja celów dysertacji	11		
	1.2. Struktura dysertacji					
2.	2. Sieci neuronowe typu ART-2 a problem klastrowania danych w czasie rzeczywistym			13		
	2.1.	Klastr	owanie danych	13		
	2.2.	Przetv	varzanie danych w czasie rzeczywistym	14		
	2.3.	Sieć ART-2				
		2.3.1.	Architektura sieci ART-2	15		
		2.3.2.	Pamięć sieci ART-2	17		
		2.3.3.	Dobór i znaczenie parametrów sieci ART-2	19		
		2.3.4.	Algorytm przetwarzania sygnałów w sieci ART-2	22		
		2.3.5.	Kontrolowanie szybkości adaptacji wag sieci ART-2	23		
		2.3.6.	Własności sieci ART-2	24		
	2.4.	4. Sieć ART-2 jako narzędzie klastrujące dane w czasie rzeczywistym				
		2.4.1.	Normalizacja danych a problem podziału przestrzeni danych przez sieć ART-2	26		
		2.4.2.	Problem podziału przestrzeni danych przez sieć ART-2 przy przetwarzaniu da-			
			nych w czasie rzeczywistym	27		
		2.4.3.	Modyfikacje sieci ART-2	28		
		2.4.4.	Wpływ normalizacji stereograficznej na efektywność klasyfikacji sieci ART-2	28		
3.	. Modele gausowskie jako mechanizm separacji danych					
	3.1.	Miesz	anina Gausowska	32		
3.2. Algorytm EM jako metoda wyznaczania MoG		ytm EM jako metoda wyznaczania MoG	33			
	3.3. MoG jako klasyfikator obszarowy					
4.	Dete	Detekcja szczególnych punktów zbioru danych poprzez klastrowanie za pomocą waria-				
	cyjn	/jnych modeli gausowskich				
	4.1.	Optymalizacja wariacyjna				
	4.2.	Wariacyjny model gaussowski				
	4.3.	Wariacyjny algorytm EM 39				

5.	Hyb	Hybrydowy system neuronalny służący do uniwersalnego klastrowania danych w czasie					
	rzeczywistym						
5.1. Cele i wymagania stawiane nowemu systemowi			wymagania stawiane nowemu systemowi	46			
	5.2.	tektura rozwiązania	46				
5.3. Opis dz			Iziałania	47			
		5.3.1.	Przetwarzanie strumienia danych wejściowych	48			
		5.3.2.	Proces kontrolny	50			
	5.4.	Złożo	ność czasowa algorytmu	51			
6.	Intel	ligentny	monitoring turbin wiatrowych	53			
	6.1.	Zadan	ie monitoringu turbin wiatrowych	53			
		6.1.1.	Znaczenie monitoringu w eksploatacji turbin wiatrowych	53			
		6.1.2.	Techniki monitorowania turbin wiatrowych	53			
		6.1.3.	Wiatr jako źródło zmienności obciążenia turbiny wiatrowej	54			
		6.1.4.	Przetwarzanie danych pochodzących z monitoringu turbiny wiatrowej	55			
	6.2.	Sieć A	ART-2 jako klasyfikator stanów operacyjnych turbiny wiatrowej	56			
		6.2.1.	Diagnoza stanów awaryjnych dla określonego przedziału danych pomiarowych	56			
		6.2.2.	Klasyfikacja on-line danych pomiarowych	56			
	6.3. Autorski system jako inteligentne narzędzie realizujące zadanie monitoringu tu						
		wiatrowej					
		6.3.1.	Dobór parametrów systemu i początkowy etap pracy przy turbinie wiatrowej	57			
		6.3.2.	Algorytm wczesnego rozpoznawania stanu przedawaryjnego przy użyciu mo-				
			dułu OCC	57			
7.	Wer	yfikacja	a działania hybrydowego systemu neuronalnego przy użyciu danych z moni-				
	toringu turbiny wiatrowej						
	7.1.	Weryf	ikacja - część I	59			
		7.1.1.	Dane pomiarowe i parametry systemu hybrydowego	59			
		7.1.2.	Wyniki	59			
		7.1.3.	Dyskusja wyników	65			
	7.2.	Weryfikacja - część II					
		7.2.1.	Dane pomiarowe i parametry systemu hybrydowego	65			
		7.2.2.	Wyniki	67			
		7.2.3.	Dyskusja wyników	70			
	7.3.	Weryf	ikacja - część III	70			
		7.3.1.	Dane pomiarowe i parametry systemu hybrydowego	70			
		7.3.2.	Wyniki	70			
		7.3.3.	Dyskusja wyników	73			

	7.4.	Podsumowanie weryfikacji	73		
8.	Wnio	ski końcowe	74		
9.	Doda	tek	76		
	Imple	mentacja hybrydowego systemu neuronalnego w środowisku MATLAB	76		
Bil	Bibliografia				

1. Wprowadzenie

1.1. Motywacja, teza i specyfikacja celów dysertacji

Energetyka wiatrowa jest jedną z najbardziej rozwijających się gałęzi energetyki nie tylko na świecie, ale także w Polsce [27, 41, 60]. W przypadku dużych turbin wiatrowych ich utrzymanie i konserwacja generują główne koszty eksploatacyjne. Do poprawnego planowania remontów i napraw służą systemy oparte o zdalny monitoring poszczególnych mechanicznych części składowych turbiny. Opierając się na dostępnej literaturze [36, 48, 29, 19, 25, 20, 21, 38, 67, 43, 44] można stwierdzić, że większość aktual-nych systemów monitoringu polega na analizie poszczególnych parametrów pomiarowych wyznaczając dla nich jedynie progi alarmowe. Jako że turbiny wiatrowe są maszynami działającymi pod zmiennym obciążeniem - wiatr oddziałuje na turbinę w sposób chaotyczny [3], to tego typu systemy mogą być w pewnych przypadkach nieskuteczne. Zauważono potrzebę inteligentniejszego monitorowania działania pracy turbiny wiatrowej, co było motywacją do rozpoczęcia badań w ramach niniejszej dysertacji.

Postawiono tezę, że przy pomocy narzędzi sztucznej inteligencji możliwe jest stworzenie w pełni autonomicznego systemu, który w czasie rzeczywistym będzie analizować parametry pracy turbiny wiatrowej, wykrywając nietypowe stany operacyjne lub taką zmianę parametrów pomiarowych, która może świadczyć o usterce lub wystąpieniu stanu przedawaryjnego.

Przedmiotowy system to hybrydowy system neuronalny, którego głównym modułem jest sieć neuronowa ART-2 (Adaptive Resonance Theory). We wcześniejszych badaniach [5, 6, 8, 12, 7] wykazana została aplikowalność tej sieci w zastosowaniach klasyfikacyjnych dotyczących danych pochodzących z monitoringu turbin wiatrowych. W niniejszej dysertacji poprzednie wyniki rozszerzone zostały do wskazania całościowego systemu, który wykorzystuje znalezione zalety sieci ART-2, natomiast zaobserwowane wcześniej niedogodności niweluje wprowadzonymi dodatkowymi metodami i podsystemami. Podsystemy te zbudowane są w oparciu o gaussowskie modele gęstościowe danych: mieszaninę gausowską (MoG) i jej wariacyjną postać (VBMoG) opartą o wariacyjne bayessowskie wnioskowanie (VB). Wykorzystany został algorytm Expectation-Maximization (EM) użyty dla modelu MoG oraz jego wariacyjny odpowiednik - algorytm Variational Bayssian Expectation-Maximization (VBEM), który dotychczas nie został opisany w polskiej literaturze. Ponadto zastosowane zostały wybrane metody preprocesingu danych.

Głównym celem dysertacji jest przedstawienie projektu tego systemu poprzez podanie definicji części składowych jego architektury oraz algorytmu jego szczegółowego działania, a także weryfikacja działania implementacji systemu na danych pochodzących z monitoringu turbiny wiatrowej. Dodatkowym celem dysertacji jest analiza teoretycznych aspektów działania sieci neuronowych typu ART-2. Szczegóły działania, własności i ograniczenia sieci ART-2, poza krótkimi wzmiankami, jak [70, 50], nie były do tej pory opisywane w polskiej literaturze.

1.2. Struktura dysertacji

Dysertacja podzielona jest na dziewięć rozdziałów. Poza Wprowadzeniem (Rozdział 1.), Wnioskami końcowymi (Rozdział 8.) i Dodatkiem (Rozdział 9.), zawiera sześć zasadniczych rozdziałów, które zapoznają czytelnika z modułami autorskiego hybrydowego systemu neuronalnego oraz możliwościami jego zastosowania w inteligentnym monitoringu turbin wiatrowych. W Rozdziale 2. przedstawiona jest sieć ART-2 wraz z omówieniem dostrzeżonych i rozwiązanych problemów występujących przy jej zastosowaniu w przetwarzaniu danych w czasie rzeczywistym. Rozdziały 3. i 4. omawiają gaussowskie modele gestościowe. W Rozdziale 3. skorzystano z modelu MoG jako mechanizmu potrafiacego separować dane, natomiast w Rozdziale 4. pokazano użycie modelu VBMoG jako środka do wyodrębnienia z określonego zbioru pomiarowego jego punktów szczególnych. W Rozdziale 5. omówiona jest teoria dotycząca architektury i działania hybrydowego systemu neuronalnego opertego o opisane wcześniej moduły. Kolejna część pracy, Rozdział 6., przedstawia zagadnienie inteligentnego monitoringu turbin wiatrowych. Opisany jest problem samego monitoringu, a następnie propozycja jego rozwiązania przy użyciu systemów opartych o sieci neuronowe typu ART-2. Wskazane są prace pokazujące skuteczność w wykorzystaniu tej sieci, po czym zaprezentowana jest implementacja hybrydowego systemu neuronalnego opisanego w Rozdziałe 5. Rozdział 7. zawiera funkcjonalną weryfikacje zastosowania hybrydowego systemu neuronalnego w monitoringu turbiny wiatrowej przy użyciu trzech zbiorów posiadanych archiwalnych danych pomiarowych.

2. Sieci neuronowe typu ART-2 a problem klastrowania danych w czasie rzeczywistym

Opisany w niniejszej dysertacji system inteligentnego monitoringu turbiny wiatrowej jest hybrydowym systemem neuronalnym czasu rzeczywistego, w którym istotne znaczenie ma klasteryzacja zbioru parametrów opisujących stan turbiny. W niniejszym rozdziale, jak również w dwóch następnych, opisane są metody i algorytmy, na których bazuje proponowany system.

2.1. Klastrowanie danych

Klastrowanie jest procesem nienadzorowanej rozłącznej klasyfikacji danych [45]. Rysunek 2.1 przedstawia taksonomię metod klasyfikacji danych z wyróżnieniem klasteryzacji. Podział na klastry może być reprezentowany poprzez funkcję, która przypisuje każdej obserwacji ze zbioru danych X liczbę naturalną oznaczającą numer klastra. Dla danego X przez P_X oznaczmy przestrzeń wszystkich podziałów zbioru X. Z algorytmem klasteryzacji można utożsamiać funkcję celu $F_X : P_X \to \mathbb{R}$, która dla danego zbioru obserwacji X, ocenia podział. Algorytmy klasteryzacji znajdują maksimum funkcji F_X [47].

Ze względu na sposób przetwarzania danych w ramach metod klasteryzacji wyróżnione są:

- Hierarchiczne algorytmy klastrowania danych (*Hierarchical Clustering Algorithms*) które mogą być podzielone na *single link/complete link*
- Algorytmy partycjonowania (*Partitional Algorithms*)
 które obejmują zarówno algorytmy oparte na wyliczaniu błędu kwadratowego jak i metody grafowej reprezentacji danych. Dodatkowo wyszczególnione są następujące techniki klastrowania danych [46]:
 - Mixture-Resolving
 - Nieparametryczne analizy przestrzeni cech (Mode-Seeking Algorithms)
 - Klastrowanie metodą najbliższego sąsiada (Nearest Neighbor Clustering)
 - Klastrowanie rozmyte (Fuzzy Clustering)
 - Sztuczne sieci neuronowe do klastrowania danych (ANN Artificial Neural Networks for Clustering)
 - Metody ewolucyjne do klastrowania danych (Evolutionary Approaches for Clustering)



Rysunek 2.1: Podział metod klasyfikacji danych [45]

- Podejścia Search-Based

Zastosowanie sztucznych sieci neuronowych cechuje:

- umiejętność przetwarzania wektorów liczbowych i wzorców, które muszą być reprezentowane przy użyciu wyłącznie cech ilościowych,
- z natury równoległa i rozproszona architektura przetwarzania,
- możliwość ustawiania wewnętrznych połączeń sieci w sposób adaptacyjny.

2.2. Przetwarzanie danych w czasie rzeczywistym

System czasu rzeczywistego (*Real-Time System*) jest to system komputerowy, w którym obliczenia prowadzone są równolegle z przebiegiem zewnętrznego procesu i mają na celu nadzorowanie, sterowanie lub terminowe reagowanie na zachodzące w tym procesie zdarzenia [74].

Tryb przetwarzania w czasie rzeczywistym jest takim trybem, w którym programy są zawsze gotowe do przetwarzania danych napływających z zewnątrz, a wynik ich działania jest dostępny nie później niż po zadanym czasie. Moment nadejścia kolejnych danych może być losowy (asynchroniczny) lub ściśle określony (synchroniczny) [53].

Można wyróżnić następujące podstawowe cechy charakterystyczne oraz wymagania stawiane systemom czasu rzeczywistego [63].

Ciągłość działania System działa bez przerwy, oczekuje na bodźce z otoczenia,

- Zależność od otoczenia System musi być rozpatrywany w kontekście otoczenia, jego działanie uzależnione jest od bodźców z otoczenia,
- Współbieżność Otoczenie systemu składa się z wielu podsystemów, które działają współbieżnie, generując bodźce wymagające obsługi przez system – wymagana jest ich równoczesna obsługa, stąd narzuca się również współbieżna struktura systemu,



Rysunek 2.2: Schemat architektury sieci ART-2

- **Przewidywalność** Bodźce z otoczenia pojawiają się zazwyczaj w momentach przypadkowych, a współbieżne wystąpienie bodźców wymaga ich jednoczesnej obsługi; mimo współbieżnej struktury wewnętrznej systemu, na zewnątrz system musi zachowywać się deterministycznie (reagować na zdarzenia według założonych wymagań),
- **Punktualność** Odpowiedzi systemu (reakcje na bodźce z otoczenia) powinny być obliczane zgodnie z zaprojektowanymi algorytmami i dostarczane do otoczenia w odpowiednich momentach czasowych.

2.3. Sieć ART-2

Sieć ART-2 należy do nienadzorowanych sieci neuronowych. Bazuje na teorii rezonansu adaptacyjnego (*Adaptive Resonance Theory* - ART). Sieć została opracowana przez Carpentera i Grossberga [17]. Jej model został zaprojektowany do pracy z danymi analogowymi.

2.3.1. Architektura sieci ART-2

Sieć ART-2 posiada dwie warstwy neuronów F_1 i F_2 , podsystem orientujący oraz macierze wag B (o elementach b_{ij}) i T (o elementach t_{ji}). Architektura sieci ART-2 jest przedstawiona na rysunku 2.3.1.

Sygnał wejściowy *s* posiada *n* składowych $(s_1 \dots s_n)$ jako element zbioru \mathbb{R}^n . Kierowany jest on najpierw do warstwy F_1 , która zawiera sześć rodzajów jednostek - komórek: *w*, *x*, *u*, *v*, *p* i *q*. Komórki te

również posiadają n składowych, a ich wartości wyliczane wg wzorów: (2.1), (2.2), (2.3), (2.4), (2.5) i (2.6).

$$w_i = s_i + au_i \tag{2.1}$$

$$x_i = \frac{w_i}{e + \|w\|} \tag{2.2}$$

$$q_i = \frac{p_i}{e + \|p\|} \tag{2.3}$$

$$v_i = f(x_i) + bf(q_i) \tag{2.4}$$

$$u_i = \frac{v_i}{e + \|v\|} \tag{2.5}$$

$$p_i = u_i + \sum g(y_i) t_{ji} \tag{2.6}$$

W tych wzorach, oprócz wartości poszczególnych komórek, występują wagi t_{ji} , funkcje f i g (opisane poniżej), a także parametry a, b i e, które zostały opisane w podrozdziale 2.3.3.

W warstwie F_1 odbywa się proces aktywacji warstwy F_1 , która obejmuje normalizację (2.3), (2.5) oraz redukcję szumów sygnału, która realizowana jest przez funkcję f mogącą mieć postać (2.7) lub (2.8), gdzie parametr θ jest wartością progową opisaną w podrozdziale 2.3.3.

$$f(x) = \begin{cases} 0, & \text{jeżeli } 0 \le x < \theta \\ x, & \text{jeżeli } x \ge \theta \end{cases}$$
(2.7)

$$f(x) = \begin{cases} \frac{2\theta x^2}{x^2 + \theta^2}, & \text{jeżeli } 0 \le x < \theta \\ x, & \text{jeżeli } x \ge \theta \end{cases}$$
(2.8)

Funkcja f wykorzystywana jest przy obliczaniu wartości komórki według wzoru (2.4). Rezultatem aktywacji warstwy F_1 jest wyjściowy sygnał p warstwy F_1 obliczony wg wzoru (2.6). Odpowiednio dla tego sygnału aktywacja przebiega w ramach warstwy F_2 poprzez wagi b_{ij} . Ponieważ warstwa F_2 jest warstwą konkurencyjną z modelem *winner-takes-all*, tylko jeden zachowany wzorzec staje się zwycięzcą. To zachowanie opisuje funkcja g, która zwycięskiemu wzorcowi przypisuje wartość d - wzór (2.12). Reprezentuje on najbardziej pasujący wzorzec dla sygnału wejściowego warstwy F_1 . Ponadto, zwycięski wzorzec warstwy F_2 wysyłany jest z powrotem do warstwy F_1 poprzez wagi t_{ji} . Następnie wykorzystywany jest podsystem orientujący, który używa mechanizmu resetowania r (2.9) i parametru czułości ρ dla sprawdzania podobieństwa pomiędzy wyjściowym wzorcem warstwy F_2 i oryginalnym sygnałem od warstwy F_1 .

$$r_i = \frac{u_i + cp_i}{e + \|u\| + \|cp\|}$$
(2.9)

Jeśli sygnał jest zgodny z wybranym wzorcem, tzn miara zgodności jest mniejsza od zadanej wartości parametru ρ , sieć wpada w stan rezonansu, gdzie następuje adaptacja wag sieci względem przetwarzanego sygnału oraz wybranego zwycięskiego neuronu *J* w warstwie *F*₂ (równania (2.10), (2.11), (2.13), (2.14)).

$$F_2 \to F_1 : \Delta t_{ji} = g(y_j)[p_i - t_{ji}], \text{ gdzie } g(y_j) \text{ zdefiniowana jest jako } (2.12)$$
 (2.10)

$$F_1 \rightarrow F_2$$
: $\Delta b_{ij} = g(y_j)[p_i - b_{ij}]$, gdzie $g(y_j)$ zdefiniowana jest jako (2.12) (2.11)

$$g(y_j) = \begin{cases} d, & \text{jeżeli } j = J \\ 0, & \text{w p.p.} \end{cases}$$
(2.12)

$$\Delta t_{Ji} = d[p_i - t_{Ji}] = d(1 - d) \left[\frac{u_i}{1 - d} - t_{Ji} \right]$$
(2.13)

$$\Delta b_{iJ} = d[p_i - b_{iJ}] = d(1 - d) \left[\frac{u_i}{1 - d} - b_{iJ} \right]$$
(2.14)

Jeśli natomiast sygnał nie jest zgodny z wybranym wzorcem, to sieć resetuje zwycięski neuron warstwy F_2 oznaczając go jako nieaktywny, po czym te same operacje zostaną wykonane dla kolejnego neuronu warstwy F_2 , który posiada największą wartość aktywacji spośród pozostałych aktywnych neuronów. We wzorach (2.9), (2.12), (2.13) i (2.14) występują parametry *c* i *d*, które zostaną omówione w podrozdziale 2.3.3.

2.3.2. Pamięć sieci ART-2

Pamięć sieci ART-2 dzieli się na dwie kategorie:

- 1. STM (*Short Term Memory*), którą stanowią stany neuronów warstwy *F*₁ i *F*₂ sieci w trakcie przetwarzania pojedynczego sygnału wejściowego. Wartości sygnałów wyjściowych neuronów tych warstw potrzebne są po wstępnym inicjowaniu aż do zakończenia nauczania danego sygnału.
- 2. LTM (Long Term Memory), na którą składają się wagi "dół-góra"(b_{ij}) i "góra-dół"(t_{ji}).

Pamięć długotrwała i jej wykorzystywanie stanowi najważniejsze ogniowo dla zapewnienia elastycznego i stabilnego działania sieci. Rozróżnienie pomiędzy wagami b_{ij} a t_{ji} występuje w sposobie inicjowania tych wag.

2.3.2.1. Wagi "góra-dół" - macierz t

Wagi "góra-dół" przechowują szablony wzorców zebranych kategorii. Ich wartości trafiają do systemu orientującego - wzór (2.9). Wartość normy *r* można zapisać w postaci wzoru (2.15), gdzie widoczna jest zależność pomiędzy wartością wagi $T_J = (t_{1J}, ..., t_{nJ})$ a sygnałem *u*, gdzie *J* jest zwycięskim neuronem warstwy F_2 .

$$\|r\| = \frac{\sqrt{(1+c)^2 + 2(1+c)\|cdT_J\|cos(u,T_J) + \|cdT_J\|^2}}{1 + \sqrt{c^2 + 2c\|cdT_J\|cos(u,T_J) + \|cdT_J\|^2}}$$
(2.15)

Zależność ta zobrazowana jest na rys. (2.3). System orientujący pozostawi wybrany J-ty neuron F_2 jako aktywny gdy spełniana jest nierówność $\rho \leq ||r||$. Jeśli ρ jest mniejsze od jeden, to rys. (2.3) pokazuje, że może to nastąpić w przypadku kiedy $cos(u, T_J)$ jest bliskie 1 (warunek równoległości u i T_J) albo $||T_J||$ jest bliskie 0. Wzór (2.13) pokazuje, że T_J staje się równoległe do u podczas procesu nauki, więc uniemożliwiony jest wtedy reset neuronu J. Mechanizm resetowania powinien również nie zadziałać zaraz po dodaniu nowego neuronu do warstwy F_2 . Żeby to osiągnąć wartości wag t_{ji} wystarczy zainicjować jak w równaniu (2.16).

$$t_{ji}(0) = 0 \tag{2.16}$$

Równania (2.13) i (2.16) gwarantują, że T_J pozostanie równoległe do *u* podczas trwania procesu nauki. System orientujący na pewno nie zresetuje nowo wybranej kategorii.

Rys. 2.3 sugeruje też jak ustawić wartości parametrów w systemie orientującym (*c* i *d*), tak żeby sieć z biegiem nauki była coraz bardziej czuła na błędne klasyfikacje. Rysunek wskazuje, że dla stałej wartości $cos(u, T_J) ||r||$ jest funkcją malejącą przy $||cdT_J|| \le 1$. Przy założeniu, że *c* dąży do 0, minimum każdej krzywej ||r|| przechodzi przez linię $||cdT_J|| = 1$. Wzory (2.13) i (2.16) implikują, że $||T_J|| < 1/(1-d)$ i $||T_J||$ dąży do 1/(1-d) poprzez proces nauki. Wymienione wzory implikują, że własność wysokiej czułości na błędne klasyfikacje da się uzyskać wykorzystując nierówność (2.17).

$$\frac{cd}{1-d} \le 1 \tag{2.17}$$

Im bliżej wyrażenie cd/(1-d) jest wartości 1 tym czułość sieci jest wyższa. Dzięki nierówności (2.17) podczas procesu nauki nie jest możliwa sytuacja, w której system orientujący mógłby zresetować neuron J-ty. Dla wykazania tej własności rozważmy wzorzec T_J . Jeśli T_J jest nowym wzorcem, jak pokazano w poprzednim paragrafie, jego wartość jest ustawiana wg wzoru (2.16) dzięki czemu $||r|| \equiv 1$, a więc $||r|| > \rho$. Jeśli natomiast T_J jest już nauczonym wcześniej wzorcem, wówczas wartość $||T_J||$ jest bliska 1/(1-d) przed rozpoczęciem i po zakończeniu procesu nauki. Jednak najczęściej wartość $||T_J||$ maleje, a następnie rośnie w trakcie tego procesu. Dlatego też, jeśli cd/(1-d) > 1, wówczas wartość ||r|| mogłaby zejść poniżej ρ , co spowodowałoby reset wzorca J. Dzięki nierówności (2.17), nawet jeśli wartość $||T_J||$ maleje, spełniona jest nierówność $||r|| > \rho$.

2.3.2.2. Wagi "dół-góra" - macierz b

Wagi "dół-góra" muszą również mieć wyznaczone odpowiednie wartości początkowe tak, by sieć mogła poprawnie wybierać kategorie dla analizowanego sygnału. Rozważmy B_J jako wybrany wzorzec poprzez warstwę F_2 . Równanie (2.14) implikuje, że $||B_J||$ dąży do 1/(1-d) podczas procedury nauczania. Jeśli wartość początkowa $||B_J||$, czyli $||B_J(0)||$, zostałaby wybierana powyżej 1/(1-d) wówczas podczas procedury nauki wybrana kategoria J mogłaby zostać zmieniona w trakcie trwania nauki na inną z uwagi na większą wartość aktywacji w warstwie F_2 . Wymagane jest więc by była spełniona nierówność (2.18).



Rysunek 2.3: Wykres normy ||r|| jako funkcji $||cdT_J||$ [17]

$$|B_J(0)|| \le \frac{1}{1-d} \tag{2.18}$$

Oznacza ona, że w przypadku równego podziału wartości pośród wszystkie wagi b_{ij} , każdy z tych elementów powinien spełniać nierówność (2.19), gdzie M to ilość składowych przetwarzanego sygnału s.

$$b_{ij}(0) \le \frac{1}{(1-d)\sqrt{M}}$$
 (2.19)

Wybierając równość we wzorze (2.19) sieć ART-2 będzie starała się dodawać nieużywany wcześniej neuron, który reprezentuje wzorzec, do warstwy F_2 w przypadku, gdy inne wzorce nie będą dobrze pasować do sygnału p. Sieć w takim wypadku wyznaczy nieużywany neuron, który zostanie poddany nauczaniu. Przypisanie dużej wartości początkowej wagom $||B_J||$ pomaga w stabilizacji działania sieci. Sieć utworzy nową kategorię zamiast zmieniać wzorzec jednej z dotychczas utworzonych kategorii, które nie pasują do analizowanego przykładu w przypadku gdy współczynnik ρ jest ustawiony na zbyt niskim poziomie.

2.3.3. Dobór i znaczenie parametrów sieci ART-2

Sieć ART-2 posiada parametry $a, b, c, d, e, \rho, \theta$. Ich zakres wartości a także początkowe wartości wag sieci mają bezpośredni wpływ na sposób działania i skuteczność sieci. Parametry sieci klasyfikuje i opisuje tabela 2.1

Tablica 2.1:	Wykorzyst	anie i znacz	zenie paran	netrów si	ieci ART-2
--------------	-----------	--------------	-------------	-----------	------------

M. Wójcik Hybrydowy system neuronalny inteligentnego monitoringu turbiny wiatrowej

			Parametry a i b wyznacza się w celu
			zapewnienia odpowiedniej proporcji
			pomiędzy sygnałami wejściowymi a
		Teoretycznie $[0, +\infty)$.	wagami sieci. Jeśli parametr b jest
		Ze wzorów na zmianę	równy lub bliski 0 wówczas w trakcie
		wag sieci wynika, że	nauczania wagi sieci będą zbyt mocno
		$ b_{ij} $ oraz $ t_{ji} $ dążą	modyfikowane, a nawet zastąpione
		do $1/(1-d)$.	przez dany sygnał wejściowy s.
		Możliwa jest więc	Efektem takiego działania jest
a,b	warstwa F_1	strategia doboru	niestabilność sieci. Podobny efekt
		parametrów <i>a</i> i <i>b</i> taka,	występuje po ustawieniu parametru d
		że ich wartość jest nie	bliskiego 0. Obierając podobnie
		większa niż	parametr a bliski lub równy 0, może
		$1/(1-d)\sqrt{M}$, gdzie	również pojawić się zjawisko
		M to ilość składowych	niestabilności także wywołane przez
		sygnału s.	zbyt mocne nadpisywanie wag sieci
			sygnałem wejściowym. Podobny efekt
			otrzymamy dla parametru d równego
			ok. 0,1.
$e \neq 0$	warstwa F_1	$[-oldsymbol{\varepsilon},oldsymbol{\varepsilon}]$ - wartość	Zabezpieczenie przed dzieleniem
	walstwa 1 I	bliska 0, $\varepsilon \neq 0$	przez zero
			Parametr c wymagany jest w systemie
			orientującym do sterowania wielkością
			sygnału p względem u we wzorze
			(2.9). Parametr d , bezpośrednio
			używany we wzorze (2.12) do
c d	podsystem	nierówność (2.17)	obliczenia wartości funkcji g, jest
c,u	orientujący	merownose (2.17)	pośrednio również wykorzystywany
			do wyliczenia wartości r, występując
			we wzorze na wartość p - wzór (2.6).
			Parametr d wyznacza jaki duży jest
			wpływ wzorca t _{ji} względem wartości
			komórki u na wartość komórki p.

d	procedura nauczania	nierówność (2.17)	Parametr <i>d</i> bezpośrednio wpływa na wysokość zmian wag b_{ij} i t_{ji} . Używany jest również do wyliczenia wartości <i>r</i> , występując we wzorze na wartość <i>p</i> - (2.6), gdzie oznacza proporcję wartości wzorca t_{ji} względem sygnału <i>u</i> .
ρ	system orientujący	[0, 1)	Parametr ρ jest parametrem czułości sieci. Ustawienie tego parametru na odpowiednio wysokim poziomie umożliwia działanie mechanizmu wyszukiwania najlepszej kategorii dla sygnału wejściowego. Szablon wzorca kategorii wskazanej przez warstwę F_2 (waga góra-dół) może być niezgodny z przetwarzanym sygnałem. Wówczas dzięki parametrowi ρ może wystąpić zresetowanie warstwy F_2 i wybranie innej kategorii.
θ	warstwa F ₁	$(0, \frac{1}{\sqrt{M}}]$, gdzie <i>M</i> to ilość składowych sygnału <i>s</i>	Parametr θ używany jest w funkcji f we wzorze na wartość v (2.4). Powoduje, że funkcja f jest nieliniowa. Parametr θ oznacza próg szumu, który jest odfiltrowywany dzięki funkcji f . Ustawienie tego progu na zbyt niskiej wartości lub jego wyeliminowanie ($\theta = 0$), w przypadku gdy równocześnie parametr czułości ρ ma zbyt niską wartość, może powodować przypisywanie istotnie różnych sygnałów wejściowych do tej samej kategorii. Powodem błędnego kategoryzowania w takim przypadku może być pokrywanie się szumu w jednym sygnale z dominującą częścią zapamiętanego wcześniej wzorca.

2.3.4. Algorytm przetwarzania sygnałów w sieci ART-2

Opisana w niniejszej dysertacji sieć ART-2 wykorzystana jest do przetwarzania strumienia danych ($S^1, ..., S^N$), gdzie strumień danych z naturalnych przyczyn nie jest ciągłym zapisem danego zjawiska, lecz jego próbkowaniem z daną częstotliwością. Opisana w podrozdziale 2.3.1 oraz w [17] architektura sieci ART-2 określa jedynie umiejscowienie poszczególnych składowych sieci i ich znaczenie w przetwa-rzaniu strumienia danych. Podana jest zależność pomiędzy składowymi sieci, brak natomiast receptury jak w czasie dokonywać zmian w ich stanie. Sugeruje to potrzebę wyspecyfikowania całościowego algorytmu pracy sieci ART-2. Zakładając zachowanie właściwych wartości parametrów sieci opisanych w poprzednich podrozdziałach, taki algorytm działania sieci został przedstawiony poniżej. Wykorzystano w nim wzory (2.20), (2.21), (2.22) służące do opisu niektórych elementarnych czynności w tym algorytmie. Algorytm ten jest tożsamy z opisem działania sieci ART-2 przedstawionym w [28].

$$T_j = \sum_i p_i b_{ij} \tag{2.20}$$

$$T_J = max\{T_j | j \notin R\}$$

$$(2.21)$$

$$\rho > \|r\| + e \tag{2.22}$$

- 1. Ustal parametry sieci $a, b, c, d, e, \rho, \theta$
- 2. Inicjuj wagi sieci wzór (2.16) i (2.19)
- 3. Dla każdego przykładu źródłowego S
 - (a) Inicjuj warstwę F_1
 - i. $u \leftarrow 0$
 - ii. $p \leftarrow 0$
 - iii. $q \leftarrow 0$
 - iv. Wyznacz w wzór (2.1)
 - v. Wyznacz x wzór (2.2)
 - vi. Wyznacz v wzór (2.4)
 - (b) Uaktualnij warstwę F_1
 - i. Wyznacz u wzór (2.5)
 - ii. Wyznacz p wzór (2.6)
 - iii. Wyznacz q wzór (2.3)
 - iv. Wyznacz w wzór (2.1)
 - v. Wyznacz x wzór (2.2)
 - vi. Wyznacz v wzór (2.4)
 - (c) Znajdź neuron J warstwy F_2

i. Oznacz wszystkie neurony warstwy F_2 jako aktywne

R - zbiór nieaktywnych neuronów

 $R := \emptyset$

- ii. Wyznaczenie wartości wyjściowych sygnałów neuronów w warstwie F₂
 wzór (2.20)
- iii. Jeśli brak jest aktywnych neuronów w warstwie F_2 to dodaj w niej aktywny neuron i zwiększ zmienną określającą ilość rozpoznanych wzorców.
- iv. Wybranie neuronu T_J z największą wartością aktywacji wzór (2.21)
- v. Uaktualnij wartość *u* wzór (2.5)
- vi. Uaktualnij wartość p wzór (2.6)
- vii. Wyznaczenie r wzór (2.9)
- viii. Jeśli zachodzi nierówność (2.22)
 - A. Oznacz neuron *i* jako nieaktywny

 $R:=R\cup\{i\}$

- B. Wróć do punktu 3(c)iv
- (d) Przeprowadź nauczanie dla neuronu J

i. Uaktualnij wartość w - wzór (2.1)

- ii. Uaktualnij wartość x wzór (2.2)
- iii. Uaktualnij wartość q wzór (2.3)
- iv. Uaktualnij wartość v wzór (2.4)
- v. Dopóki warunek zakończenia zmiany wagi sieci nie zostanie osiągnięty to:
 - A. uaktualnij wagi b_{ij} wzór (2.13 oraz t_{ji} wzór (2.14)
 - B. Uaktualnij wartość u wzór (2.5)
 - C. Uaktualnij wartość p wzór (2.6)
 - D. Uaktualnij wartość q wzór (2.3)
 - E. Uaktualnij wartość w wzór (2.1)
 - F. Uaktualnij wartość x wzór (2.2)
 - G. Uaktualnij wartość v wzór (2.4)

Warunek występujący w pętli 3(d)v można sformułować w jednej z następujących postaci.

- 1. Jako z góry określoną, skończona liczbę iteracji adaptacji wag.
- 2. Sprawdzania, czy przyrost wag w ostatniej iteracji jest mniejszy od założonej z góry wartości.

2.3.5. Kontrolowanie szybkości adaptacji wag sieci ART-2

W celu zwiększenia kontroli nad szybkością adaptacji wag sieci ART-2 można wprowadzić dodatkowy parametr sterujący α - współczynnik uczenia sieci. Wzory (2.10) i (2.11), a także ich rozwinięte postacie (2.13) i (2.14) otrzymają czynnik α , który powinien mieścić się w zakresie (0% – 100%]. Po modyfikacji rozwinięte postacie wzorów (2.13) i (2.14) przyjmą postać wzorów: (2.23) i (2.24). Zmiana ta nie wpływa na prezentowane wcześniej cechy sieci ART-2. W szczególności norma wartość wag $||b_{ij}||$ a także $||t_{ji}||$ w dalszym ciągu dąży do 1/(1-d)

$$\Delta t_{ji} = d[p_i - t_{ji}] = \alpha d(1 - d) \left[\frac{u_i}{1 - d} - t_{ji} \right]$$
(2.23)

$$\Delta b_{ij} = d[p_i - b_{ij}] = \alpha d(1 - d) \left[\frac{u_i}{1 - d} - b_{ij} \right]$$
(2.24)

2.3.6. Własności sieci ART-2

Sieć ART-2 ma następujące własności:

- Kompromisowe rozwiązanie problemu stabilności i plastyczności. System ART-2 jest zdolny do zapamiętywania wzorców w postaci stabilnych rozpoznawczych kodów w odpowiedzi na arbitralną sekwencję analogowych wzorców wejściowych. Pomimo, że prezentacje wejściowych danych mogą mieć arbitralny czas trwania, przetwarzanie STM jest zdefiniowane w taki sposób, że dłużej występujący nowy wzorzec wejściowy nie wymazuje wcześniej zgromadzonych informacji. Umożliwia to utrzymanie przez cały czas plastyczności systemu ART.
- 2. Kompromis wyszukiwania i bezpośredniego dostępu. System ART-2 wykonuje równoległe przeszukiwanie w celu dostosowania selekcji odpowiednich kodów rozpoznawczych podczas procesu nauki. Przerwanie procesu wyszukiwania następuje automatycznie jak tylko wejściowy wzorzec stanie się rozpoznawalny. Po czym, po podaniu podobnego wzorca wejściowego, jego rozpoznawcze kody zostają bezpośrednio zlokalizowane niezależnie od tego jak bardzo złożona stała się cała struktura powstała przez dotychczasowe nauczanie.
- 3. Kompromis ciągłego dopasowania zapamiętanych wzorców a powstawania nowych. System sieci ART-2 jest w stanie z jednej strony reagować na arbitralnie małe różnice pomiędzy aktywnym wzorcem STM warstwy F₁ i odczytanym wzorcem z LTM dla ustalonej kategorii. W szczególności, jeśli parametr czułości ρ jest duży, wzorzec STM powstały w warstwie F₁ musi być bliski lub identyczny względem wzorca oczekiwanego góra-dół, aby odpowiadająca jemu kategoria mogła zostać wybrana. Z drugiej strony sieć, w sytuacji gdy nieużywana komórka warstwy F₂ stanie się aktywna, potrafi pozostawić ją aktywną, bez resetowania, dzięki czemu wzorzec dla nowo powstałej kategorii może być poprawnie zachowany.
- 4. Niezmienności wzorca STM przy dopasowanym wzorcu LTM. Warstwa F₁ sieci zawiera 3 poziomy komórek: dolny (komórki w i x), środkowy (komórki u i v) i górny (komórki p i q). Odczytana wartość z warstwy F₂, korzystając z nauczonego wzorca LTM, który znakomicie pasuje do wzorca STM, umieszczana jest wyłącznie w górnej części warstwy F₁. Wzorce w dolnej i środkowej części tej warstwy pozostają niezmienione oraz nie zachodzi zjawisko resetowania. To samo dotyczy odczytania wektora zerowego LTM, przy wykryciu istnienia nowej kategorii. Ta

niezmienność w dolnej i środkowej części warstwy F_1 umożliwia stosowanie nieliniowych transformacji wzorca wejściowego w taki sposób, że pozostaje on stabilny podczas procesu nauczania.

- 5. Współwystępowanie odczytywanych wzorców LTM i normalizacji STM. Istnienie górnej warstwy F_1 umożliwia przeprowadzanie swobodnych kalkulacji przy użyciu znormalizowanego wzorca STM i odczytanego wzorca LTM, przed ich oddziaływaniem na środkową część F_1 . Podobnie występowanie części dolnej umożliwia przeprowadzenie normalizacji wektora wejściowego zanim zostanie uwzględniony w warstwie środkowej, gdzie komórka *u* jest używana przy adaptacji wag sieci. Wyłączenie normalizacji z warstwy F_1 spowodowałoby destabilizację sieci zarówno przy wyszukiwaniu kategorii jak i w procesie nauczania.
- 6. Brak zaburzeń wzorców LTM przez ich nadzbiory jako wzorce wejściowe. Przy pojawieniu się wzorca wejściowego będącego nadzbiorem któregoś wzorca LTM, zostanie wybrany inny wzo-rzec LTM (uaktywniona jest procedura resetu) albo, przy niskim współczynniku czułości, wzorzec będzie skierowany dalej do procesu nauczania, jednak składniki LTM które występowały jedynie we wzorcu wejściowym pozostaną małe.
- 7. **Stabilność procesu wyboru kategorii do momentu resetu**. Wybór kategorii odbywa się zawsze na tych samych zasadach strategia *winner takes all* wzór (2.21). Istnieje tylko jeden sposób na zmianę wybranej kategorii, którą jest procedura resetowania wywoływana przez podsystem orientujący.
- 8. Korekta kontrastu, redukcja szumów i tłumienie niedopasowania dokonywane poprzez funkcje nieliniowe. Kombinacja normalizacji i nieliniowego sprzężenia zwrotnego zastosowana w warstwie F_1 sieci umożliwia oddzielanie sygnału od szumu. W szczególności korekta kontrastu, która stosowana jest dla wzorca STM, powoduje także jej uwzględnienie przy pobieraniu wartości LTM. Funkcje nieliniowe operują na sumie znormalizowanych wzorców dół-góra i góra-dół. Umożliwia to polepszenie tłumienia całkowitej aktywacji warstwy F_1 w przypadku niezgodności wzorca wejściowego dól-góra z oczekiwanym wzorcem LTM góra-dół. W przypadku braku nieliniowości w warstwie F_1 pętla sprzężenia zwrotnego umożliwiałaby, przy niskiej wartości współczynnika czułości, wyznaczenie wszystkim podzbiorom danego wzorca tej samej kategorii.
- 9. Szybka samo-stabilizacja systemu. System nauczania, który jest niestabilny, może stać się bardziej stabilny poprzez obniżenie współczynnika nauczania dla LTM. W takim przypadku potrzeba większej ilości przetwarzanych przykładów, żeby zapamiętać określony zbiór danych wejściowych. Prędkość nauki systemu ART jest relatywnie spowolniona do wartości wzbudzenia STM. System ART-2 jest zdolny do stabilnej nauki nawet w przypadku szybkiego nauczania, gdzie przykłady zmieniają się bardzo często. W każdym przypadku zachowywane są w LTM zrównoważone wartości. Szybkość samo-stabilizacji systemu jest wspomagana przez zastosowanie podsystemu orientującego, który jednak do zachowania samo-stabilności sieci nie jest niezbędny.
- 10. **Przetwarzanie danych lokalnie**. Do obliczeń w ramach STM i LTM używane są informacje dostępne lokalnie w czasie rzeczywistym. Nie występuje w systemie ART-2 zjawisko transportu war-



Rysunek 2.4: Wizualizacja problemu występującego podczas klastrowania danych przy użyciu sieci ART-2. Punkty danych D_1 , D_2 i D_3 są przetwarzane przez sieć ART-2 jako wektory E_1 , E_2 i E_3 . Wektory E_1 , E_2 mają podobną odległość kątową względem zapamiętanego wzorca P_1 , podczas gdy E_3 jest bliski wzorcowi P_2 . Oznacza to, że punkty D_1 i D_2 mogą być klasyfikowane jako ta sama grupa, która jest rozłączna względem D_3 . Przyjmując odległość euklidesową, taka klasyfikacja jest błędna - odległość $|D_1D_2|$ może być nawet kilkakrotnie większa od $|D_1D_3|$, a punkt D_1 będzie zawsze klasyfikowany jako ta sama grupa co punkt D_2 , który jest istotnie różny względem D_3 .

tości wag, takie jak ma miejsce w sieci typu perceptron przy zastosowaniu wstecznej propagacji błędów. Wszystkie obliczenia dokonywane w warstwach F_1 i F_2 oparte są na wzorach o prostych formułach.

2.4. Sieć ART-2 jako narzędzie klastrujące dane w czasie rzeczywistym

2.4.1. Normalizacja danych a problem podziału przestrzeni danych przez sieć ART-2

Przetwarzane przez sieć ART-2 sygnały wejściowe można traktować jako punkty $w = (w_1, ..., w_n)$ należące do przestrzeni $V \subset \mathbb{R}^n$, czyli jako wektory $[w_1, ..., w_n]$. Wzorzec wejściowy w jest normalizowany zgodnie ze wzorem (2.2). Dzieje się tak przy każdym wzorcu wejściowym, co może prowadzić do błędnych klasyfikacji, gdy do oceny przynależności do danej klasy wzorców p będzie brana odległość euklidesowa. Wzorce wejściowe - wektory p_1 i p_2 , będą zaklasyfikowane do tej samej klasy kiedy ich odległość kątowa będzie taka sama lub zbliżona - wzór (2.15). W przypadku odległości kątowych identycznych wzorce p_1 i p_2 będą zaklasyfikowane do tej samej klasy niezależnie od wartości współczynnika czułości sieci ρ . Problem ten może wystąpić w przestrzeni o dowolnym wymiarze. Kolejność wzorców wejściowych nie ma w tym problemie znaczenia. Rys 2.4 przedstawia elementarny przykład błędnej klasyfikacji dwuwymiarowych danych. Punkt D_1 posiada zbliżoną odległość kątową względem D_2 i będzie rozpoznany przez sieć ART-2 jako ten sam klaster danych reprezentowany przez wzorzec P_1 . Punkt D_3 ,



Rysunek 2.5: Wizualizacja problemu przy klastrowaniu danych online

opierając się na metryce euklidesowej, znajduje się dwukrotnie bliżej punktu D_1 niż punkt D_2 . Rozpatrując jednak odległości kątowe, różnica jest istotna, przez co punkt D_3 zostaje sklasyfikowany jako inny klaster. Problem może być trudny w zdiagnozowany przy przetwarzaniu danych wysokowymiarowych.

2.4.2. Problem podziału przestrzeni danych przez sieć ART-2 przy przetwarzaniu danych w czasie rzeczywistym

Sieć ART-2 posiada cechy stabilności i plastyczności, dzięki czemu nadaje się do przetwarzania danych w czasie rzeczywistym. Głównym parametrem sterującym pracą sieci ART-2 jest współczynnik czułości ρ , którego ustawienie determinuje przyszły proces klasteryzacji. Przy przetwarzaniu danych w czasie rzeczywistym zakres wartości danych i ich rozkład mogą być trudne do przewidzenia, przez co wybór współczynnika czułości jest rzeczą skomplikowaną. Rysunek 2.5 przedstawia przykład problemu wyboru współczynnika czułości dla danych dwuwymiarowych. W celu uproszczenia przykładu zakłada się, że dane wejściowe wypełniają kolejne strefy od A do E. Współczynnik czułości zostaje dobrany przy przetwarzaniu danych ze strefy A, tak żeby cała ta strefa była reprezentowana przez jeden klaster danych. Zakres kątowy strefy punktów A jest większa niż kątowa odległość pomiędzy punktami ze stref D i E, co powoduje, że punkty stref D i E mogą zostać rozpoznane jako ten sam klaster. Przykład można uogólnić na dowolną liczbę stref które są wypełniane w kolejnych etapach, jednak zgodnie z zasadą, że jedna ze stref w początkowym etapie działania sieci ma większy zakres kątowy niż odległość między strefami w dalszym etapie działania.

2.4.3. Modyfikacje sieci ART-2

Sieć ART-2 była modyfikowana lub rozszerzana w celu wypełniania różnych specyficznych zadań [18, 51, 1, 16, 55].

Przeprowadzane były prace badawcze nad zbliżonymi do wymienionych w podrozdziałach 2.4.1 i 2.4.2 problemach. Do rozwiązania pierwszego z wymienionych problemów została zaprojektowana sieć PART (Projective Adaptive Resonance Theory) [16]. W ramach tej sieci została wprowadzona dodatkowa funkcja sprawdzająca podobieństwo pomiędzy wektorem wag LTM a wzorcami wejściowymi. Algorytm wykorzystywany w sieci PART posiada jednak wady w związku z przetwarzaniem w czasie rzeczywistym i dlatego została wprowadzona sieć BPART [55]. BPART jest rozszerzeniem sieci PART o obszar buforowy pomiędzy warstwą F_1 a F_2 . Zarządzanie tym obszarem buforowym, a także nowa definicja dodatkowej funkcji wprowadzonej w PART jest istotą działania tej sieci.

Zarówno PART jak i BPART potrzebują określenia dodatkowych względem sieci ART-2 arbitralnie dobieranych parametrów. Najważniejszy z nich jest używany przy wyznaczaniu stopnia podobieństwa wektora wag LTM i wzorca wejściowego. Jest on jednak trudny do wyznaczenia, szczególnie w przypadku przetwarzania danych w czasie rzeczywistym, gdzie prognozowanie przyszłych wartości danych może być niedoskonałe.

2.4.4. Wpływ normalizacji stereograficznej na efektywność klasyfikacji sieci ART-2

Problem sieci ART-2 ujęty w podrozdziale 2.4 może być rozwiązany w prostszy sposób niż wspomniana w poprzednim podrozdziale sieć PART. Tym sposobem jest zastosowanie stereograficznej normalizacji danych jako metody wstępnego przetwarzania sygnałów wejściowych sieci ART-2. Definicja tej normalizacji jest tematem następnego sekcji 2.4.4.1. Natomiast w sekcji 2.4.4.2 pokazano jak efektywnie zaimplementować normalizację stereograficzną dla sieci ART-2.

2.4.4.1. Normalizacja stereograficzna dla sieci neuronowych

Procedura normalizacji sprowadza się do znalezienia odwzorowania

$$F : \mathbb{R}^n \supset A \ni \vec{x} \to \hat{x} \in \mathbb{R}^k$$
, where $\|\hat{x}\| = 1$.

Najbardziej znaną realizacją normalizacji jest użycie wzoru $\hat{x} = \frac{\vec{x}}{\|\vec{x}\|}$. Wzór ten definiuje projekcję

$$\Pi: \mathbb{R}^n \setminus \{\vec{0}\} \to \mathscr{S}^{n-1} \subset \mathbb{R}^n,$$

tak zwaną prostą projekcję zbioru $\mathbb{R}^n \setminus \{\vec{0}\}$ na (n-1)-wymiarowa sferę \mathscr{S}^{n-1} - rys.2.6.

Prosta projekcja posiada istotne wady. Po pierwsze, wymiar przestrzeni danych jest redukowany. Po drugie, projekcja nie jest zdefiniowana na całej przestrzeni - odwzorowanie nie jest zdefiniowane dla wektora $\vec{0}$. Ponadto przestrzeń \mathbb{R}^n posiadająca miarę nieskończoną jest rzutowana na sferę posiadającą miarę skończoną. Dodatkowo, projekcja nie jest odwzorowaniem iniektywnym - jeśli dwa punkty *u* i *w* leżą na tej samej radialnej linii wtedy $\Pi(u) = \Pi(w)$ - rys. 2.6. Odnosząc się do rozpatrywanego problemu, oznacza to, że jeśli dwa klastry są usytuowane wzdłuż tego samego kierunku radialnego, to po normalizacją nie będą mogły być dobrze odseparowane, nawet jeśli były odseparowane przed normalizacją.



Rysunek 2.6: Prosta projekcja zbioru \mathbb{R}^2



Rysunek 2.7: Projekcja stereograficzna dla zbioru \mathbb{R}^2

Dlatego też opisywana do tej pory metoda normalizacji powinna być użyta tylko w takich przypadkach jeśli z góry wiadomo, że klastry w przestrzeni wzorców wejściowych są usytuowane w różnych kierunkach radialnych. Wynika z tego, że w wielu przypadkach sposób normalizacji nie redukujący przestrzeni wzorców wejściowych jest niezbędny. Projekcja stereograficzna

$$\Pi_S:\mathbb{R}^n\to\mathscr{S}^n\subset\mathbb{R}^{n+1}$$

jest przykładem takiego odwzorowania. Została ona stworzona z myślą o przetwarzaniu danych przez sieci neuronowe typu Kohonena - [13]. Geometryczna interpretacja projekcji stereograficznej jest zwizualizowana na rys. 2.7 dla przypadku dwuwymiarowej przestrzeni.

Projekcja stereograficzna transformuje n-wymiarową euklidesową przestrzeń w n-wymiarową sferę, której południowy biegun jest w początku układu współrzędnych. Klasyczna wersja projekcji stereograficznej jest określona jednoznacznie przez wzór algebraiczny dla każdego naturalnego n [30]. Wzory podane poniżej opisują lekko zmodyfikowaną projekcję stereograficzną - przestrzeń \mathbb{R}^n jest transformowana w jednostkową sferę \mathbb{R}^n , której środek leży w początku układu współrzędnych, ponieważ taka forma jest pożądana w przetwarzaniu danych przez sieci neuronowe.

Niech $P = (x_1, ..., x_n)$.. Wtedy $S(P) = \tilde{P} = (\tilde{x}_1, ..., \tilde{x}_{n+1})$ jest wyznaczone poprzez równania (2.25).

$$\tilde{x}_{i} = \frac{2x_{i}}{s+1} \text{ for } i = 1, ..., n;
\tilde{x}_{n+1} = \frac{s-1}{s+1},
\text{gdzie } s := \sum_{i=1}^{n} x_{i}^{2}$$
(2.25)

Jak to zostało powiedziane, projekcja stereograficzna zachowuje wymiar transformowanej przestrzeni i jest definiowana na całym zbiorze \mathbb{R}^n . Ponadto jest to odwzorowanie iniektywne, tzn. jeżeli $u \neq v, u, v \in \mathbb{R}^n$ wtedy $S(u) \neq S(v)$. Transformuje ono przestrzeń o mierze nieskończonej w przestrzeń o mierze skończonej. Implikuje to m.in. fakt, że punkty przestrzeni, które są daleko oddalone od siebie w \mathbb{R}^n mogą być blisko siebie na \mathscr{S}^n , przez co dwa klastry, które są dobrze separowalne w \mathbb{R}^n mogą być ciężko separowalne po zastosowaniu normalizacji. Jednak taki przypadek może wystąpić tylko jeśli klastry są bardzo daleko względem początku układu współrzędnych, są wtedy transformowane blisko północnego bieguna sfery. Ponieważ w praktyce normy transformowanych wektorów są ograniczone, minimalna odległość pomiędzy klastrami po normalizacji może być oszacowana.

2.4.4.2. Adaptacja normalizacji stereograficznej dla sieci ART-2 poprzez skalowanie danych

Opisywana w podrozdziale 2.4.4.1 przez wzory (2.25) klasyczna wersja projekcji stereograficznej może być aplikowana do sieci neuronowych, które potrafią przetwarzać pełny zakres liczb ze zbioru \mathbb{R} . Sieci ART-2 nie potrafią rozpoznawać i zapamiętywać wartości ujemnych. Dane dostarczane do sieci ART-2 przed wykonywaniem normalizacji powinny być więc dodatkowo przygotowane, tak żeby w wyniku normalizacji każdy wzorzec danych zawierał wyłącznie nieujemne wartości. Można w tym celu użyć prostego skalowania danych. Wartości w każdym wymiarze skalowane są do wyznaczonego przedziału [x_a, x_b], gdzie $x_a \ge 0$ wg wzoru (2.26). Wzory (2.25) definiują sferę o środku w początku układu współrzędnych, więc wektory o wartościach bliskich x_a znajdą się przy równiku sfery.

$$V_i(k) = (V_i(k) - \min(V(k))) / \max(V(k))(x_b - x_a) + x_a;$$
(2.26)

Opcjonalnie można przed zastosowaniem wzoru (2.26) użyć transformacji danych $V^N \subset \mathbb{R}^N$ na $V'^{2N} \subset \mathbb{R}^{2N}$ zdefiniowanej w następujący sposób:

if
$$V_i(k) \ge 0$$
 then
 $V'_i(2k-1) := V_i(k)$
 $V'_i(2k) := 0$
else
 $V'_i(2k-1) := 0$
 $V'_i(2k) := V_i(k)$
end if

Powyższe kodowanie zwiększa wymiar przestrzeni dwukrotnie, jednak pozwala na odseparowanie wartości ujemnych, których istnienie może nieść dodatkową informację.

Przeskalowanie danych do przedziału $[x_a, x_b]$ i zastosowanie wprost normalizacji wg wzorów (2.25) wprowadza problem przy porównywaniu punktów danych w postaci wektorowej. Jeśli wartość x_b jest duża, to wektory dużej długości i dużych odległościach kątowych pomiędzy sobą, zostaną po normalizacji przekształcone w wektory wskazujące okolicę bieguna północnego z małą wartością kątową pomiędzy sobą. Prowadzi to do błędów klasyfikacji przez sieć ART-2. Dla potrzeb zastosowania normalizacji stereograficznej do sieci ART-2, oprócz skalowania danych, potrzebna jest więc modyfikacja definicji samej normalizacji znanej z (2.25). Taką modyfikacją może być odwrócenie biegunów (kierunku rzutowania danych na sferę) - wzory (2.27). W okolicę bieguna północnego przekształcane są wówczas wektory początkowo bliskie zera. Natomiast jeśli sfera ma promień równy 1 wówczas największe wartości będą rzutowane na równik jeśli przedziałem skalowania będzie $\left[0, \frac{1}{\sqrt{2}}\right]$.

$$\tilde{x}_{i} = \frac{2x_{i}}{1+s} \text{ for } i = 1, ..., n;
\tilde{x}_{n+1} = \frac{1-s}{1+s},
\text{gdzie } s := \sum_{i=1}^{n} x_{i}^{2}$$
(2.27)

3. Modele gausowskie jako mechanizm separacji danych

W tym rozdziale podjęty zostanie problem separacji danych jako klasyfikacji jednej klasy danych (*One Class Classification*) - OCC. Celem takiej klasyfikacji jest poprawna identyfikacja obiektów jednej klasy danych względem wszystkich możliwych obiektów, przy użyciu zbioru treningowego zawierającego tylko obiekty tej jednej klasy. Przyjęte dalej zostaje, że rozpatrywanymi obiektami są punkty danych z *R*ⁿ. Klasyfikatory OCC potrafią więc zamodelować granicę obszaru określonych punktów zbioru danych tak, żeby odróżnić je od innych punktów wyróżnianych jako przynależne do innej klasy. W [72] został szeroko przedstawiony problem OCC wraz z omówieniem kilku różnych klasyfikatorów tego typu. Jednym z nich jest gęstościowa metoda analizy danych, tzw. Mieszanina Gausowska (*Mixture of Gaussian*) - MoG, zaproponowana przez Duda i Hart[24]. W poniższym rozdziale przedstawiona zostanie definicja MoG oraz algorytmu wyznaczania MoG, a następnie będą one zaimplementowane jako część hybrydowego systemu separująca przetwarzane dane. Metoda ta została wybrana spośród innych z uwagi na swą prostotę i małą ilość parametrów modelu (porównanie różnych metod OCC - patrz [66]).

3.1. Mieszanina Gausowska

Mieszaninę Gausowską można zdefiniować jako kombinację liniową rozkładów normalnych podaną w równaniu (3.1), gdzie π_k to współczynniki mieszające, spełniające warunki (3.2), a funkcje $\mathcal{N}(x \mid \mu_k, \Sigma_k)$ są gaussowskimi rozkładami prawdopodobieństwa z centrami μ_k i macierzami kowariancji Σ_k .

$$p(x) = \sum_{k=1}^{K} \pi_k \mathcal{N}(x \mid \mu_k, \Sigma_k)$$
(3.1)

$$0 < \pi_k < 1, \sum_{k=1}^K \pi_k = 1 \tag{3.2}$$

Sygnał wejściowy x jest zmienną wielowymiarową w R^n . Liczba rozkładów Gaussa określona jest arbitralnie jako K. Każdy z nich może być wyrażony jako równanie (3.3).

$$\mathcal{N}(x \mid \mu, \Sigma) = \frac{1}{(2\pi)^{d/2} |\Sigma|^{1/2}} \exp\left(-\frac{1}{2} (x - \mu)^T \Sigma^{-1} (x - \mu)\right)$$
(3.3)

Macierz kowariancji Σ_k może być dowolną pełną macierzą nieosobliwą (odwracalną).

3.2. Algorytm EM jako metoda wyznaczania MoG

Wyznaczanie MoG polega określeniu parametrów π_k , μ_k i Σ_k równania (3.1) poprzez maksymalizację prawdopodobieństwa p(x) dla każdego punktu x ze zbioru danych. Współczynniki mieszające równania (3.1) mogą być interpretowane jako prawdopodobieństwo a priori. p(x) może być więc określone jako prawdopodobieństwo całkowite - wzór (3.4).

$$p(x) = \sum_{k=1}^{K} p(k)p(x \mid k)$$
(3.4)

Dla danej wartości *x* odpowiadające prawdopodobieństwo a posteriori może być obliczone ze wzoru (3.5). Te prawdopodobieństwo jest nazywane odpowiedzialnością i bazuje na twierdzeniu Bayesa.

$$p(k \mid x) = \frac{\pi_k \mathcal{N}(x \mid \mu_k, \Sigma_k)}{p(x)}$$
(3.5)

Funkcja logarytmicznego prawdopodobieństwa dla zbioru danych X przyjmuje postać wzoru (3.6).

$$\ln p(X \mid \pi, \mu, \Sigma) = \sum_{i=1}^{N} ln \sum_{k=1}^{K} \pi_k \mathcal{N}(x_i \mid \mu_k, \Sigma_k)$$
(3.6)

Nie ma jawnego wzoru dla znajdowania maksimum tego prawdopodobieństwa. Możliwe jest jednak użycie iteracyjnej metody EM (*Expectation - Maximization*). Jest ona logicznie podzielona na dwa kroki, które występują na przemian: krok E (*expectation*) oraz M (*maximization*). Krok E ma na celu wyznaczenie wartości nazwanych odpowiedzialnością (p(k | x)) dla każdego wejściowego wzorca, natomiast w kroku M wyszukiwane są wartości μ_k , Σ_k i π_k . W kroku E wartości obliczane są na podstawie zmiennych wyliczonych w poprzednim M-kroku i na odwrót.

W kroku E wartości $p(k \mid x)$ wyliczane są wg wzoru (3.5).

W kroku M wartości μ_k , Σ_k i π_k wyliczane są niezależnie. W tym celu funkcja (3.6) jest różniczkowana częściowo względem osobno μ_k oraz Σ_k , a następnie przyrównywana do zera wyznaczając argumenty, dla których uzyskuje się maksymalną wartość. Tak uzyskiwane są nowe wartości μ_k i Σ_k określone wzorami (3.7) i (3.8).

$$\mu_{k} = \frac{\sum_{i=1}^{N} p(k \mid x_{i}) x_{i}}{\sum_{i=1}^{N} p(k \mid x_{i})}$$
(3.7)

$$\Sigma_{k} = \sum_{i=1}^{N} \frac{p(k \mid x_{i})(x_{i} - \mu_{k})(x_{i} - \mu_{k})^{T}}{\sum_{i=1}^{N} p(k \mid x_{i})}$$
(3.8)

W celu obliczenia nowej wartości π_k rozszerza się funkcję logarytmu prawdopodobieństwa - wzór (3.6) o składnik dotyczący współczynników π_k - wzór (3.2) uzyskując postać wzoru (3.9).

$$\ln p(X \mid \pi, \mu, \Sigma) + \lambda (\sum_{k=1}^{K} \pi_k - 1)$$
(3.9)

Różniczkowanie częściowe równania (3.9) po zmiennej π_k i przyrównanie wartości do zera w celu wyszukania jej maksimum daje rezultat w postaci równania (3.10).

$$\pi_k = \frac{1}{N} \sum_{i=1}^{N} p(k \mid x_i)$$
(3.10)

Metoda EM przed rozpoczęciem właściwych kroków E i M inicjalizuje zmienne dla każdego $k \in [1, K]$ następująco:

- μ_k jest wybierana losowo jako wielowymiarowy punkt spośród danych,
- Σ_k jest obliczana zakładając, że μ_k jest centrum rozkładu gaussa. Punkty danych są rozdystrybuowane pomiędzy wszystkimi centrami. Niech X_k oznacza macierz punktów należących do centrum k. Macierz Σ_k wyznaczana jest jako $\frac{(X_k - \mu_k)^T (X_k - \mu_k)}{N_k}$, gdzie N_k jest ilością punktów przyporządkowanych do centrum μ_k ,
- $-\pi_k=\frac{N_k}{N}.$

3.3. MoG jako klasyfikator obszarowy

W celu skorzystania z modelu MoG, wyznaczonego algorytmem EM, jako klasyfikatora obszarowego należy określić wartość progową p_x . Punkty x, których wartości p(x) wyznaczone poprzez równanie (3.1) są większe od wartości progowej p_x , uznawane są jako należące do definiowanego obszaru. Przyjmuje się, że punkty x, o wartościach poniżej wartości progowej p_x , do definiowanego obszaru nie należą. Wartość p_x można ustalić przy pomocy zbioru danych X poprzez wartość procentową $p_{\%}$, taką że $p_{\%} = \frac{\sharp X_{p_x}}{\sharp X}$, gdzie $X_{p_x} = \{x \in X | p(x) > p_x\}$, a operator \sharp oznacza moc wskazywanego zbioru. Rysunki 3.1 oraz 3.2 przedstawiają przykłady działania klasyfikatorów obszarowych wyznaczonych przez użycie Mieszaniny Gaussowej. Oba klasyfikatory zostały stworzone z tej samej Mieszaniny Gaussa wyznaczonej algorytmem EM, jednak przy użyciu innych wartości $p_{\%}$.



Rysunek 3.1: Przykład działania klasyfikatora obszarowego uzyskanego z Mieszaniny Gaussowej o K = 7 i $p_{\%} = 90\%$ dla danych 2D pochodzących z monitoringu turbiny wiatrowej.



Rysunek 3.2: Przykład działania klasyfikatora obszarowego uzyskanego z Mieszaniny Gaussowej o K = 7 i $p_{\%} = 99\%$ dla danych 2D pochodzących z monitoringu turbiny wiatrowej.

4. Detekcja szczególnych punktów zbioru danych poprzez klastrowanie za pomocą wariacyjnych modeli gausowskich

Tematem tego rozdziału jest określenie metody wyszukiwania cech szczególnych w zbiorze danych. Metoda ta tworzy jeden z modułów hybrydowego systemu opisywanego w rozdziale 5. Jej celem jest wyróżnienie podzbioru szczególnych punktów ze zbioru danych $X \subset \mathbb{R}^n$, które reprezentują różniące się znacznie obszary zbioru X. Wewnątrz systemu hybrydowego każdy element tego podzbioru będzie przekształcony w osobny wzorzec pamięci LTM sieci ART-2. Przy użyciu zbioru wynikowego określona zostanie początkowa dokładność sieci ART-2, poprzez wyznaczenie współczynnika czułości sieci ρ .

Do rozwiązania przedstawionego problemu posłuży metoda zbliżona do EM MoG prezentowanej w rozdziale 3. Stosując wprost metodę EM MoG zbiór danych zostałby przybliżony poprzez MoG (równanie 3.1) z zadaną ilością składników mieszających *K*. Liczba ta jest stała i określałaby bezpośrednio ilość wynikowych punktów. Przy małej liczbie *K* liczba użytych rozkładów Gaussa jest mała i prawdopodobnie mogą nie interferować one w sposób znaczący, co generowałoby niską liczbę klas ART-2 nie nakładających się na siebie. W przypadku jednak dużej wartości parametru *K*, klasy sieci ART-2 mogłyby dzielić przestrzeń nierównomiernie, logicznie nakładając się na siebie, co stwarzałoby problem. Należy zauważyć również, że zbiory wynikowe o dużej ilości elementów mogą powodować powstanie zbyt dużej ilości wzorców sieci ART-2, a co za tym idzie, ustawienie współczynnika czułości sieci na zbyt wysoki poziom, powodujący jej niestabilność.

W tym rozdziale zostanie przedstawiony algorytm VBEM (Variational Bayess EM) MoG, który działając podobnie jak EM MoG, dla zadanej dużej wielkości *K* wyznacza MoG wg wzoru (3.1). Ma jednak tą wykorzystywaną w tej pracy zaletę, że potrafi w trakcie działania odrzucać niepotrzebne składowe MoG, redukując końcową ilość znalezionych punktów, tak aby pozostawić tylko te punkty, które znacząco się od siebie różnią. Docelowa ilości punktów szczególnych nie jest więc definiowana, a automatyczny ich dobór przez algorytm VBEM powoduje tworzenie wzorców w sieci ART-2, które mało są ze sobą powiązane.

W poniższym podrozdziale 4.1 przedstawiony jest alternatywny względem podrozdziału 3.2 sposób na maksymalizowanie funkcji logarytmu prawdopodobieństwa danych. Algorytm VBEM bazuje na tym sposobie, posługując się wariacyjną wersją modelu Gaussa, która jest opisana w podrozdziale 4.2. Rozdział kończy szczegółowy opis działania samego algorytmu VBEM podany w podrozdziale 4.3. Wymienione podrozdziały zostały spisane bazując na [14].
4.1. Optymalizacja wariacyjna

Wiele problemów optymalizacji może być sformułowanych w sposób równoważny jako znajdowanie ekstremów pewnego funkcjonału. Rozwiązanie jest uzyskiwane przez przeszukiwanie wszystkich możliwych zadanych funkcji, tak aby znaleźć taką, która minimalizuje lub maksymalizuje funkcjonał. Metody wariacyjne ograniczają zakres funkcji, dla których optymalizacja jest wykonywana. Rozważmy pełny model Bayesowski, gdzie wszystkie parametry posiadają rozkład prawdopodobieństwa, w którym zakodowana jest informacja o określonej zmiennej. Wśród zmiennych możemy wyróżnić dane (zmienne obserwowalne X) oraz zmienne ukryte i parametry Z, które zdefiniowane są w ramach konkretnego modelu. Załóżmy, że mamy K niezależnych zmiennych $X = x_1, ..., x_K$, dla których istnieją w modelu zmienne ukryte $Z = z_1, ..., z_K$. Model probabilistyczny specyfikuje łączny rozkład prawdopodobieństwa p(X,Z), gdzie naszym celem jest znalezienie rozkładu prawdopodobieństwa a posteriori p(Z|X), a także określenie p(X). Możemy rozłożyć funkcję logarytmu prawdopodobieństwa danych lnp(X) (obliczaną też w algorytmie EM - podrozdział 3.2) wg poniższego wzoru:

$$\ln p(X) = L(q) + KL(q||p)$$
(4.1)

gdzie składowe są zdefiniowane jako:

$$L(q) = \sum_{Z} q(Z) ln \frac{p(X,Z)}{q(Z)}$$
(4.2)

$$KL(q||p) = -\sum_{Z} q(Z) ln \frac{p(Z|X)}{q(Z)}.$$
(4.3)

L jest dolną granicą logarytmu prawdopodobieństwa danych. Możemy maksymalizować L przez optymalizację rozkładu q(Z), co jest równoważne minimalizacji KL. Jeśli wybierzemy q(Z), tak że dolna granica L osiągnie wartość maksymalną, wówczas KL zaniknie. Ma to miejsce kiedy rozkład q(Z) jest równy rozkładowi a posteriori p(Z|X). Dopasowywanie wszystkich możliwych rozkładów q(Z) jest niemożliwe, więc ograniczamy zakres poszukiwanych rozkładów q(Z) i szukamy takiego rozkładu, który minimalizuje *KL*. Rozdzielamy elementy Z na rozłączne grupy Z_i gdzie i = 1...M. Przyjmujemy, że rozkład q faktoryzuje się na czynniki względem poszczególnych grup, tj:

$$q(Z) = \prod_{i=1}^{M} q_i(Z_i)$$
(4.4)

Nie ma ograniczeń odnośnie rozkładu q(Z), jak również jego czynników $q_i(Z_i)$. Celem jest maksymalizacja dolnej granicy L. Wstawiając równianie (4.4) do (4.2) i wyodrębniając jeden czynnik $q_j(Z_j)$ (oznaczając go jako q_j) otrzymujemy

$$L(q) = \sum_{Z} (\prod_{i} q_{i}) (\ln p(X, Z) - \sum_{i} \ln q_{i}) =$$

=
$$\sum_{Z_{j}} (q_{j}) (\sum_{Z_{i}} \ln p(X, Z) \prod_{i \neq j} q_{i}) - \sum_{Z_{j}} q_{j} \ln q_{j} + const =$$

=
$$\sum_{Z_{j}} q_{j} \ln p'(X, Z_{j}) - \sum_{Z_{j}} q_{j} \ln q_{j} + const$$

(4.5)



Rysunek 4.1: Schemat wariacyjnego modelu Gaussa przy użyciu *Plate notation*. Kwadratami oznaczono stałe parametry, okręgami zmienne losowe. Wypełnione figury oznaczają wartości definiowalne. Wskazanie [K] oznacza wektor o rozmiarze K, natomiast [D,D] macierz o wymiarach DxD.

gdzie został zdefiniowany nowy rozkład $p'(X, Z_i)$ zgodnie z relacją

$$\ln p'(X,Z_j) = E_{i \neq j}(\ln p(X,Z)) + const$$

w której notacja $E_{i\neq j}(\ln p(X,Z))$ oznacza $\sum_{Z_i}(\ln p(X,Z)\prod_{i\neq j}q_i)$.

Optymalizacja wariacyjna polega na ustawieniu sztywno rozkładów $q_{i\neq j}$ i maksymalizacji L(q) w (4.5) uwzględniając wszystkie możliwe formy rozkładów $q_j(Z_j)$. Zauważając, że (4.5) jest przeciwna do dywergencji *KL* pomiędzy $q_j(Z_j)$ a $p'(X,Z_j)$, maksymalizacja (4.5) jest więc równoważna minimalizacji *KL*, gdzie minimum występuje gdy $q_j(Z_j) = p'(X,Z_j)$. Daje to całościowe wyrażenie optymalnego rozwiązania $q_j^*(Z_j)$:

$$\ln q_{i}^{*}(Z_{j}) = E_{i \neq j}(\ln p(X, Y)) + const.$$
(4.6)

Oznacza to, że logarytm z optymalnego rozwiązania dla czynnika q_j jest osiągane przez proste wyliczenie logarytmu ze wspólnego rozkładu wszystkich zmiennych ukrytych i obserwowanych, a następnie wyliczenie wartości oczekiwanej z uwzględnieniem czynników q_i dla $i \neq j$. Układ równań zadany wzorem (4.6) stanowi zbiór spójnych warunków dla osiągnięcia maksymalizacji dolnej granicy L, przy założeniu ograniczenia w postaci faktoryzacji. Jednakże równania te nie stanowią dosłownego rozwiązania, ponieważ wyrażenie w prawej stronie równania (4.6) dla zachowania optimum $q_j^*(Z_j)$ zależy od wartości oczekiwanych wyliczanych dla innych czynników $q_i(Z_i)$ dla $i \neq j$. Dlatego też, aby uzyskać spójne rozwiązanie, najpierw inicjalizowane są odpowiednio wszystkie czynniki $q_i(Z_i)$, a następnie cyklicznie przechodzi się po nich wszystkich, gdzie w każdym kroku wartość danego czynnika ponownie estymuje się za pomocą prawej strony równania (4.6) przy użyciu aktualnie znanych wartości innych czynników. Zbieżność jest gwarantowana, ponieważ granica jest wypukła dla każdego czynnika $q_i(Z_i)$ [15].

4.2. Wariacyjny model gaussowski

Zdefiniowana w poprzednim rozdziale kombinacja liniowa rozkładów Gaussa (3.1) może być wyrażona poprzez model widoczny na rys. 4.1. Dla każdej obserwacji x_n mamy stowarzyszoną zmienną ukrytą z_n zawierającą binarne wektory mające K składowych z elementami z_{nk} dla k = 1,...,K. Zbiór analizowanych danych oznaczmy jako $X = x_1,...,x_N$, a zbiór zmiennych ukrytych jako $Z = z_1,...,z_N$. Prawdopodobieństwo warunkowe zbioru zmiennych ukrytych pod warunkiem posiadania współczynnika mieszającego π wyraża wzór:

$$p(Z|\pi) = \prod_{n=1}^{N} \prod_{k=1}^{K} \pi_k^{z_{nk}}.$$
(4.7)

Posiadając zbiór zmiennych ukrytych oraz parametry Gaussa μ oraz $\Delta = \Sigma^{-1}$, prawdopodobieństwo otrzymania zbioru danych wynosi:

$$p(X|Z,\mu,\Delta) = \prod_{n=1}^{N} \prod_{k=1}^{K} \mathscr{N}(x \mid \mu_k, \Sigma_k)^{z_{n,k}}, \text{ gdzie } \mu = \{\mu_k\} \text{ i } \Delta = \{\Delta_k\}.$$

$$(4.8)$$

Definicje prawdopodobieństw a priori dla parametrów μ , Δ oraz π są następujące:

$$p(\pi) = Dir(\pi | \alpha_0) = C(\alpha_0) \prod_{k=1}^K \pi_k^{\alpha_0 - 1}.$$
(4.9)

$$p(\boldsymbol{\mu}, \boldsymbol{\Delta}) = p(\boldsymbol{\mu} | \boldsymbol{\Delta}) p(\boldsymbol{\Delta}) = \prod_{k=1}^{K} \mathscr{N}(\boldsymbol{\mu}_k | \boldsymbol{m}_0, (\boldsymbol{\beta}_0 \boldsymbol{\Delta}_k)^{-1}) \mathscr{W}(\boldsymbol{\Delta}_k | \boldsymbol{W}_0, \boldsymbol{v}_0).$$
(4.10)

Dla współczynników mieszających π użyty zostaje rozkład Dirichleta (4.9), gdzie symetrycznie zostaje wybrany ten sam parametr α_0 dla każdego gausowskiego komponentu, natomiast $C(\alpha_0)$ jest znormalizowaną stałą rozkładu Dirichleta. Parametr α_0 może być interpretowany jako skuteczna, wstępna ilość obserwacji, która jest powiązana z każdym komponentem mieszaniny gausowskiej. Jeśli wartość α_0 jest mała, to na rozkład a posteriori będą miały wpływ głównie dane. Podobnie wprowadzony zostaje rozkład Gaussa-Wisharta zarządzający parametrami μ i Δ (4.10).

Łączny rozkład prawdopodobieństwa wszystkich zmiennych w modelu na rys. 4.1 zadany jest przez wzór (4.11), gdzie każdy z wariacyjnych czynników zdefiniowany jest powyżej.

$$p(X, Z, \pi, \mu, \Delta) = p(X|Z, \mu, \Delta)p(Z|\pi)p(\pi)p(\mu|\Delta)p(\Delta)$$
(4.11)

4.3. Wariacyjny algorytm EM

Zakłada się, że wariacyjny rozkład q z równania (4.1) można wyrazić przez faktoryzację zmiennych ukrytych i parametrów w równaniu (4.12).

$$q(Z, \pi, \mu, \Delta) = q(Z)q(\pi, \mu, \Delta) \tag{4.12}$$

Równania przybliżające rozkłady q(Z) i $q(\pi, \mu, \Delta)$ można wyrazić za pomocą ogólnej zasady przedstawionej w równaniu (4.6). Dla czynnika q(Z) wyraża je równanie (4.13).

$$\ln q^*(Z) = E_{\pi,\mu,\Delta}[\ln p(X,Z,\pi,\mu,\Delta)] + const$$
(4.13)

Biorąc pod uwagę łączny rozkład prawdopodobieństw w równaniu (4.11) można zastosować dekompozycję - wszystkie wyrażenia niezależne od Z mogą być ujęte jako dodatkowa stała normalizacji uzyskując równanie (4.14).

$$\ln q^*(Z) = E_{\pi}[\ln p(Z|\pi)] + E_{\mu,\Delta}[\ln p(X|Z,\mu,\Delta)] + const$$
(4.14)

Rozwijając dwa warunkowe rozkłady prawej strony równania i ponownie ujmując wszystkie wyrażenia niezależne od Z w dodatkowej stałej, otrzymuje się równanie (4.15), gdzie $\ln \rho_{n,k}$ jest zdefiniowane poprzez równanie (4.16).

$$\ln q^*(Z) = \sum_{n=1}^N \sum_{k=1}^K z_{n,k} \ln(\rho_{n,k}) + const$$
(4.15)

$$\ln \rho_{n,k} = E[\ln \pi_k] + \frac{1}{2}E[\ln |\Delta_k|] - \frac{D}{2}\ln(2\pi) - \frac{1}{2}E_{\mu_k,\Delta_k}[(x_n - \mu_k)^T \Delta_k(x_n - \mu_k)]$$
(4.16)

W równaniu (4.16) D oznacza wymiar przestrzeni, do której należy zmienna x. Wyliczając wartość $q^*(Z)$ za pomocą (4.15) otrzymujemy równanie (4.17). Wymagając aby ten rozkład był znormalizowany i zauważając, że dla każdej wartości n wartości z_{nk} są binarne i sumują się do jedności dla wszystkich k, otrzymujemy równanie (4.18), gdzie r_{nk} jest wyrażone poprzez równanie (4.19).

$$q^*(Z) \propto \prod_{n=1}^N \prod_{k=1}^K \rho_{nk}^{z_{nk}}$$
 (4.17)

$$q^*(Z) = \prod_{n=1}^N \prod_{k=1}^K r_{nk}^{z_{nk}}$$
(4.18)

$$r_{nk} = \frac{\rho_{nk}}{\sum_{j=1}^{K} \rho_{nj}} \tag{4.19}$$

Dla dyskretnych rozkładów $q^*(Z)$ można określić, że dla $z_{nk} \in Z$ zachodzi (4.20).

$$E[z_{nk}] = r_{nk} \tag{4.20}$$

Za pomocą zmiennych r_{nk} można wyznaczyć analogicznie do metody EM trzy parametry statystyczne MoG:

$$N_k = \sum_{n=1}^{N} r_{n,k}$$
(4.21)

$$\bar{x_k} = \frac{1}{N_k} \sum_{n=1}^N r_{n,k} x_n$$
(4.22)

$$S_k = \frac{1}{N_k} \sum_{n=1}^N r_{n,k} (x_n - \bar{x_k}) (x_n - \bar{x_k})^T$$
(4.23)

Przechodząc do wyliczenia drugiego czynnika z równania (4.12) uzyskujemy

$$\ln q^*(\pi,\mu,\Delta) = \ln p(\pi) + \sum_{k=1}^K \ln p(\mu_k,\Delta_k) + E_z[\ln p(Z|\pi)] + \sum_{k=1}^K \sum_{n=1}^N E[z_{nk}] \ln \mathcal{N}(x_n|u_n,\Delta_k^{-1}) + const$$
(4.24)

Można zaobserwować, że prawa strona równania może być rozłożona jako suma wyrażeń obejmujących tylko zmienną π oraz wyrażeń dotyczących tylko zmiennych μ i Δ . Implikuje to możliwość faktoryzacji rozkładu $q(\pi, \mu, \Delta)$ na $q(\pi)q(\mu, \Delta)$. Wyrażenia z udziałem zmiennych μ i Δ zawierają sumowanie po wszystkich wartościach k, stąd całość faktoryzacji $q(\pi, \mu, \Delta)$ można zapisać wg wzoru (4.25).

$$q(\pi,\mu,\Delta) = q(\pi) \prod_{k=1}^{K} q(\mu_k,\Delta_k)$$
(4.25)

Rozpoznając wyrażenia po prawej stronie równania (4.24), które zależą od π , można odczytać następującą wartość optymalnego rozkładu $q^*(\pi)$:

$$\ln q^*(\pi) = (\alpha_0 - 1) \sum_{k=1}^K \ln \pi_k + \sum_{k=1}^K \sum_{n=1}^N r_{nk} \ln \pi_k + const$$
(4.26)

Następnie, nakładając na obie strony równania funkcję *exp*, można zauważyć, że $q^*(\pi)$ wyraża się poprzez rozkład (4.27), gdzie α posiada komponenty α_k dane jako równanie (4.28).

$$q^*(\pi) = Dir(\pi|\alpha) \tag{4.27}$$

$$\alpha_k = \alpha_0 + N_k \tag{4.28}$$

Rozkład $q^*(\mu_k, \Delta_k)$ nie jest faktoryzowany na składowe. Korzystając jednak z prawdopodobieństwa warunkowego można zawsze otrzymać $q^*(\mu_k, \Delta_k) = q^*(\mu_k | \Delta_k) q^*(\Delta_k)$, gdzie dwa wynikowe czynniki można uzyskać analizując równanie (4.24) pod kątem udziału zmiennych μ_k i Δ_k . Rezultatem jest rozkład Gaussa-Wisharta, zadany przez równanie (4.29), gdzie składowe β_k , m_k , W_k^{-1} oraz v_k zdefiniowane są przez równania (4.30-4.33).

$$q^*(\mu_k, \Delta_k) = \mathscr{N}(\mu_k | m_k, (\beta_k \Delta_k)^{-1}) \mathscr{W}(\Delta_k | W_{k, \nu_k})$$
(4.29)

$$\beta_k = \beta_0 + N_k \tag{4.30}$$

$$m_k = \frac{1}{\beta_k} (\beta_0 m_0 + N_k \bar{x}_k)$$
(4.31)

$$W_k^{-1} = W_0^{-1} + N_k S_k + \frac{\beta_0 N_k}{\beta_0 + N_k} (\bar{x_k} - m_0) (\bar{x_k} - m_0)^T$$
(4.32)

$$v_k = v_0 + N_k \tag{4.33}$$

Równania (4.28) oraz (4.30-4.33) są analogiczne do kroku M algorytmu EM i są określane jako wariacyjny M-krok. W celu wyliczenia prawych stron tych równań potrzebna jest informacja o wartości zmiennych r_{nk} (reprezentujących odpowiedzialności). Są one uzyskiwane przez normalizowanie zmiennej $\rho_{n,k}$ w równaniu (4.16), gdzie potrzebnymi składowymi są wartości oczekiwane $E_{\mu_k,\Delta_k}[(x_n - \mu_k)^T \Delta_k (x_n - \mu_k)]$, $E[ln|\Delta_k|]$ oraz $E[ln\pi_k]$. Te wartości oczekiwane można oszacować za pomocą zdefiniowanych wariacyjnych rozkładów uzyskując równania (4.34-4.36), gdzie wprowadzone zostały oznaczenia ψ (funkcja digamma) oraz $\hat{\alpha}$ (równa $\sum_k \alpha_k$). Równania (4.35) i (4.36) wynikają ze standardowych własności rozkładów Wisharta i Dirichleta.

$$E_{\mu_k,\Delta_k}[(x_n - \mu_k)^T \Delta_k (x_n - \mu_k)] = D\beta_k^{-1} + \nu_k (x_n - m_k)^T W_k (x_n - m_k)$$
(4.34)

$$E[\ln|\Delta_k|] = \sum_{i=1}^{D} \psi(\frac{v_k + 1 - i}{2}) + Dln2 + \ln|W_k|$$
(4.35)

$$E[\ln |\pi_k|] = \psi(\alpha_k) - \psi(\hat{\alpha}) \tag{4.36}$$

Optymalizacja rozkładów wariacyjnych pociąga za sobą naprzemienne wykonywanie dwóch etapów analogicznych do kroków E i M algorytmu wyznaczania maksymalnego prawdopodobieństwa EM. W wariacyjnej odmianie kroku E używane są aktualnie znane rozkłady, oparte na przybliżanych modelowych parametrach, które służą do wyliczenia równań (4.34-4.36), a następnie wartości współczynnika r_{nk} równania (4.20). Wariacyjny krok M bazuje na aktualnie wyliczonych wartościach współczynnika r_{nk} , używając ich do ponownego wyliczenia wariacyjnego rozkładu przy użyciu (4.27) i (4.29). W każdym przypadku wariacyjny rozkład a posteriori posiada tą samą funkcyjną postać co odpowiadający czynnik łącznego rozkładu prawdopodobieństwa zadanego równaniem (4.11). Ten rezultat jest konsekwencją wyboru sprzężonych rozkładów prawdopodobieństwa.

W wariacyjnym algorytmie EM istnieje kompromis w wyznaczaniu Bayesowskiego modelu pomiędzy dopasowaniem się modelu do danych a jego złożonością, gdzie złożoność wzrasta dla komponentów o parametrach, które zostały zmienione względem swoich początkowych wartości. Komponenty, które są od początku niezmienione lub przestają być odpowiedzialne za modelowanie punktów danych otrzymują wartości $r_{nk} \simeq 0$, a więc też $N_k \simeq 0$. Z równań (4.28) oraz (4.30-4.33) widzimy, że dla takich komponentów *k* ich parametry wracają do ustawień początkowych. Zasadniczo takie komponenty są dopasowane w jakimś stopniu do punktów danych, ale z powodu ustawienia wartości początkowych ten stopień jest numerycznie niezauważalny. Wartość oczekiwana zmiennych mieszających w modelu VBMoG prezentuje równanie (4.37).

$$E[\pi_k] = \frac{\alpha_k + N_k}{k_{\alpha_0} + N} \tag{4.37}$$

Można rozważyć przypadki brzegowe wartości a prori α_0 dla komponentu *k*, dla którego $N_k \simeq 0$. Jeśli wartość $\alpha_0 \to 0$, wówczas $E[\pi_k] \to 0$ i komponent nie odgrywa żadnej roli w modelu. Natomiast w przypadku wartości $\alpha_0 \to \infty$ zachodzi $E[\pi_k] \to 1/K$. Można zauważyć bliskie podobieństwo pomiędzy algorytmami VBEM i EM. W przypadku kiedy $N \rightarrow \infty$ wówczas przytoczony model Bayesowski jest zbieżny do wyliczanego maksymalnego prawdopodobieństwa klasycznego algorytmu EM.

Oba algorytmy dzieli różnica w koszcie obliczeniowym, na niekorzyść algorytmu VBEM, gdzie przez bardziej skomplikowany model m.in. wyliczanie współczynnika r_{nk} jest bardziej kosztowne.

Algorytm VBEM posiada jednak ważne, wykorzystywane w tej pracy zalety. Po pierwsze, dla algorytmu VBEM komponenty Gaussa nie dopasowują się do osobliwości w zbiorze danych, jak może to mieć miejsce w klasycznym algorytmie EM. Dodatkowo, zjawisko przeuczenia nie występuje nawet pomimo wybrania dużej ilości komponentów *K*. Algorytm VBEM pozwala więc na wyznaczenie optymalnej ilości komponentów w sumie gausowskiej bez potrzeby korzystania z technik takich jak walidacja krzyżowa.

Dla wariacyjnego algorytmu EM zastosowanego dla Mieszaniny Gaussa można oszacować dolną granicę L, która zdefiniowana jest wzorem (4.2) - patrz [14] rozdział 10.2.2. Wyliczona wartość L może pomóc w monitorowaniu pracy algorytmu, a także wyznaczeniu warunku jego zatrzymania, jako braku zmienności wartości L w czasie.

Rysunki 4.2-4.5 ilustrują przykład działania algorytmu VBEM dla danych 2D złożonych z 1000 kolejnych punktów wyjętych z zapisu monitoringu jednej z turbin wiatrowych. W przedstawionym przykładzie użyto 50 składowych Gaussa. Na rysunkach 4.2-4.4 zostało pokazane rozłożenie składowych Gaussa po pierwszej iteracji algorytmu, w środkowej fazie działania i po ostatniej (191-wszej) iteracji. Natomiast rysunek 4.5 prezentuje jak w trakcie działania algorytmu zmieniała się maksymalizowana dolna granica *L*. W iteracji 191 uzyskana wartość *L* nie różniła się o więcej niż zadany wcześniej próg od wartości z wcześniejszej iteracji, dlatego algorytm zakończył wtedy pracę. Ostatecznie zostało wyznaczonych 5 istotnych składowych Gaussa, których środki są szukanymi szczególnymi punktami dla zadanego zbioru danych.



Rysunek 4.2: Przykład działania algorytmu VBEM dla danych 2D - stan po pierwszej iteracji. Dane zawierają 1000 kolejnych punktów zapisu z monitoringu turbiny wiatrowej.







Rysunek 4.4: Przykład działania algorytmu VBEM dla danych 2D - stan po 191-wszej (ostatniej) iteracji. Dane zawierają 1000 kolejnych punktów zapisu z monitoringu turbiny wiatrowej (niebieskie kropki). Algorytm VBEM wyznaczył centra (czerwone kropki) składowych Gaussa (czerwone elipsy).



Rysunek 4.5: Wykres dolnej granicy L maksymalizowanej w trakcie działania algorytmu VBEM.

5. Hybrydowy system neuronalny służący do uniwersalnego klastrowania danych w czasie rzeczywistym

5.1. Cele i wymagania stawiane nowemu systemowi

Proponowany system posiada zalety sieci ART-2, a w głównej mierze jej stabilność i plastyczność (podrozdział 2.3.6). Równocześnie, system nie jest podatny na zagrożenia ujęte w podrozdziale 2.4.2, które wynikają z podstawowych założeń działania sieci ART-2. System może być uruchomiony dla dowolnego strumienia danych liczbowych o skończonym wymiarze. Celem działania systemu jest zgłaszanie informacji o wystąpieniu szczególnych zdarzeń, polegających na pojawieniu się nowych danych, które zostały odmiennie zaklasyfikowane względem danych wcześniej przetworzonych. Zakładając, że wcześniej przetworzone dane opisują stany pracującego prawidłowo monitorowanego systemu, dane odmiennie sklasyfikowane będą informowały o stanach alarmowych.

5.2. Architektura rozwiązania

Wysokopoziomową architekturę systemu prezentuje rys. 5.1. Składowymi hybrydowego systemu są:

- sieć neuronowa ART-2 (opisana w podrozdziale 2.3 z uwzględnieniem podrozdziału 2.3.5)
- moduł do projekcji stereograficznej jako realizacja podrozdziału 2.4.4.1
- K-jednostek MoG (opisanych w rozdziale 3)
- VBMoG (opisany w rozdziale 4)

Strumień danych przechodzi kolejno podsystemy VBMoG, MoG, projekcję stereograficzną, aby końcowo trafić do sieci neuronowej ART-2. Moduł VBMoG wpływa na stan sieci ART-2, moduły MoG fil-



Rysunek 5.1: Schemat wysokopoziomowej architektury hybrydowego systemu neuronalnego.



Rysunek 5.2: Schemat architektury hybrydowego systemu neuronalnego.

trują ruch przechodzący do sieci ART-2, natomiast ilość i zawartość modułów MoG zależy od działania sieci ART-2.

Patrząc na taksonomię metod klastrowania danych zaprezentowanej na rys. 2.1, prezentowany system należy do obu gałęzi nienadzorowanych metod klasyfikacji danych. Sieć ART-2 charakteryzuje typ partycjonowania. Jednak w opisywanym systemie wyznaczane są dwa logiczne poziomy klasyfikacji danych, wstępny poziom modułu MoG, a następnie poziom niższy sieci ART-2. Występuje, więc w przypadku nowego systemu dwupoziomowa hierarchiczna klasyfikacja.

Szczegółową architekturę systemu prezentuje rys. 5.2.

Do zainicjowania systemu potrzebne są głównie parametry: θ_{StartT} oraz θ_{DeltaT} , ale można także zdefiniować $\theta_{MinIloscKlastrow}$ oraz $\theta_{MaxProporcjallosciKlastrow}$. Na wejściu zainicjalizowany system otrzymuje dane wejściowe reprezentowane jako punkty ze zbioru \mathbb{R}^N . Dla każdego wejścia generowana jest informacja zwrotna czy dany punkt wchodzi w skład nowego rozpoznanego zdarzenia. Dodatkowo można odczytywać z systemu historię rozpoznanych stanów pracy maszyny.

5.3. Opis działania

System hybrydowy zaprojektowany jest do ciągłego przetwarzania strumienia danych. Głównym modułem, na którym oparte jest to przetwarzanie, jest sieć ART-2. Informowanie o wystąpieniu szczególnego zdarzenia jest tożsame z wystąpieniem nowego wzorca w tej sieci. Dodatkowe moduły wspierają tą sieć na różny sposób. Projekcja stereograficzna pozwala przesyłać do sieci ART-2 dowolne dane liczbowe oraz zapobiega wystąpieniu problemu opisanego w podrozdziale 2.4.1. Moduł MoG został zaprojektowany do odciążenia sieci ART-2 w procesowaniu wszystkich danych. Dzięki temu modułowi, sieć ART-2 zajmuje się tylko najświeższymi danymi, przez co może dostosowywać się do różnego ich charakteru. Dostosowywanie polega na możliwości zmiany współczynnika czułości sieci. Moduł MoG pełni też funkcję opisującą historyczne dane - wyznacza obszary, w których praca maszyny była stabilna, co wnioskowane jest z niezmiennej ilości rozpoznanych przez sieć ART-2 klastrów. Poprawne wyznaczenie początkowej ilości klastrów umożliwia moduł VBMoG, który wykorzystywany jest w pierwszym etapie pracy systemu.

Logicznie można wyróżnić dwa procesy biorące udział w nauce takiego systemu. Należy do nich podstawowe przetwarzanie strumienia danych otrzymywane na wejściu oraz proces kontrolny.

5.3.1. Przetwarzanie strumienia danych wejściowych

Schemat pracy opisywanego systemu przedstawiony jest na rysunku 5.3. System po zainicjalizowaniu wszystkich parametrów analizuje strumień danych punkt po punkcie. Przez czas $[0, \theta_{StartT}]$ używany jest wyłącznie moduł VBMoG. Sieć ART-2 nie może być używana od samego początku z uwagi na nieznajomość jej parametru czułości dla nowych danych. Po przekroczeniu progu czasowego θ_{StartT} , dla zgromadzonych danych uruchamiany jest algorytm VBEM, który wyznacza *K* struktur gaussowych, ustalając położenie centrów μ_k i macierzy kowariancji Σ_k . Wyjściowe *K* struktur redukowane jest do samego zbioru *K* punktów - centrów μ_k . Następnie biorą one udział we wstępnym nauczaniu sieci ART-2. Polega ono na nauczeniu sieci kolejno wszystkich *K* punktów w taki sposób, aby każdy nowy punkt został rozpoznany jako nowa klasa. W tym celu współczynnik czułości ρ ustawiany jest na najmniejszą możliwą wartość jaką spełnia to założenie (nierówność 2.22). Po zakończeniu nauki w sieci ART-2 powstaje *K* nowych klas gotowych na rozpoznawanie dotychczasowych wzorców danych i wyróżnianiu nowych od nich odmiennych. Poniżej zaprezentowane są szczegółowe kroki algorytmu inicjalizacji sieci ART-2:

- 1. Uruchom algorytm VBEM na określonym zbiorze danych otrzymując K punktów.
- 2. Zainicjalizuj sieć ART-2 ustawiając parametr czułości $\rho = 0$.
- 3. Parametrowi α , zdefiniowanemu w podrozdziale 2.3.5, ustal wartość 1
- 4. Oznacz wszystkie neurony warstwy F_2 jako nieużywane.
- 5. Dla każdego punktu danych $p \ge K$
 - (a) Dla każdego neuronu n warstwy F_2 , który jest oznaczony jako używany
 - i. Wyznacz wartość czułości v dla punktu p i neuronu n, jak to ma miejsce w czasie standardowego przetwarzania punktu przez sieć ART-2 przy ustalaniu prawej strony nierówności 2.22.
 - ii. Jeżeli v > p to $p \leftarrow v$.
 - (b) Oznacz jeden z nieużywanych dotąd neuronów jako używany i wykonaj dla niego standardowy proces nauczania.
- 6. Przywróć poprzednią wartość parametru α .



Rysunek 5.3: Schemat blokowy algorytmu sterującego działaniem hybrydowego systemu neuronalnego.



Rysunek 5.4: Schemat blokowy algorytmu używanego w procesie kontrolnym.

Po chwili θ_{StartT} przetwarzanie danych realizowane jest w pętli analizującej każdy kolejny punkt danych. Najpierw sprawdzana jest możliwa przynależność nowych punktów do któregoś z zapamiętanych obszarów MoG. Jeśli taka przynależność została stwierdzona, wówczas przetwarzanie punktu zostaje zakończone z informacją zwrotną, stwierdzającą występowanie znanego już wcześniej stanu maszyny. W przypadku kiedy punkt danych nie został rozpoznany przez moduł MoG, zostaje poddany projekcji stereograficznej i jest przekazany na wejście sieci ART-2. W przypadku wyznaczenia przez sieć ART-2 nowej klasy/wzorca dla danego punktu danych, informuje ona, że przetworzony punkt wprowadził maszynę w nowy nieznany wcześniej stan.

5.3.2. Proces kontrolny

Schemat przebiegu procesu kontrolnego prezentuje rysunek 5.4. Proces kontrolny wyzwalany jest co chwilę θ_{DeltaT} . Jego zadaniem jest kontrolowanie ilości klas powstających w sieci ART-2. Głównym jego sprawdzeniem jest określenie czy ilość klas K nie zmieniła się od ostatniego wywołania lub wielu ostatnich wywołań (# $K_{\theta_{deltaT}}$). Jeśli ilość klas jest niezmieniona, wówczas stwierdza się, że maszyna jest w stabilnym stanie pracy. Wyznacza się wówczas nowy obszar MoG dla wszystkich zgromadzonych danych zbieranych od wyznaczenia poprzedniego obszaru MoG. Używany jest algorytm EM opisany w podrozdziale 3.2, gdzie zadawana ilość rozkładów Gaussa jest proporcjonalna do ilości klas *K* sieci ART-2. Po czym wszystkie dotychczas rozpoznane klasy w sieci ART-2 są usuwane, poprzez sprowadzenie wszystkich wag sieci do wartości początkowych.

Proces kontrolny nadzoruje również czy ilość klastrów nie przekroczyła wyznaczonych wcześniej granic.

Dolna granica ($\theta_{MinIloscKlastrow}$) ustalana jest przy inicjalizacji systemu i powinna być większa od 1. W przypadku określenia tylko jednej klasy jako minimum i zmienienia się charakteru danych, współczynnik czułości sieci ART-2 może być znacznie za niski i jego odchylenie od pożądanej wartości może się w niekontrolowany sposób zwiększać. W przypadku minimalnej ilości klas większej od 1, istnieje bezpieczny margines, po którego przekroczeniu można podjąć odpowiednią akcję. W przypadku zbyt małej ilości klastrów jest to zwiększenie parametru czułości o zdefiniowaną z góry wartość procentową. Jednak zbyt mała ilość klastrów może wystąpić tylko po wyznaczeniu nowego obszaru MoG, gdyż wówczas sieć ART-2 kontynuuje pracę bez żadnych wzorców LTM. Z definicji sieci ART-2 ilość rozpoznanych wzorców nie może się zmniejszać. Jeśli jednak sieć rozpozna wiele klas, a następnie charakter danych się zmieni, co pociągnie za sobą wymaganie zmniejszenia współczynnika czułości, wówczas system poczeka na określenie stabilnej ilości klas (a taka będzie miała miejsce skoro współczynnika czułości jest za niski) i po wyznaczeniu nowego obszaru podejmie korektę wartości współczynnika czułości.

Górna granica ($\theta_{MaxIloscKlastrow}$) jest wyznaczana po zakończeniu pracy przez moduł VBMoG, gdzie sieć ART-2 otrzymuje N początkowych klas. Wartość $\theta_{MaxIloscKlastrow}$ wyznacza się poprzez zdefiniowany przy inicjalizacji systemu parametr $\theta_{MaxProporc jaIlosciKlastrow}$ - równanie (5.1).

$$\theta_{MaxIloscKlastrow} = max(\theta_{MinIloscKlastrow}, N) * \theta_{MaxProporcjallosciKlastrow}$$
(5.1)

Po przekroczeniu $\theta_{MaxIloscKlastrow}$ współczynnik czułości jest zmniejszany o zdefiniowaną wcześniej wartość procentową.

Implementacja wyzwalania czynności prowadzącej do stworzenia nowego obszaru MoG może się różnić od implementacji wyzwalania czynności nadzorowania ilości klas względem ustalonych granic - wartość θ_{DeltaT} może być ustalana odmiennie dla tych dwóch czynności. Dodatkowo, dla pierwszej z nich, jednostki czasu mogą być liczone względem punktów przetwarzanych przez system hybrydowy, natomiast dla drugiej czynności, względem punktów przetwarzanych przez sieć ART-2.

W czasie procesu kontrolnego w przypadku stwierdzenia stabilności pracy maszyny można wstrzymać tworzenie nowego obszaru MoG jeśli ilość punktów danych zebranych do tego celu jest mniejsza od ustalonej wcześniej wartości.

5.4. Złożoność czasowa algorytmu

Złożoność czasowa wykorzystania modułu VBMoG jest związana ze zdefiniowanym warunkiem stopu dla algorytmu VBEM. Jeśli założymy ograniczenie na maksymalną ilość iteracji tego algorytmu, to czas wykonania jest liniowy względem ilości punktów danych.

Przetwarzanie pojedynczych punktów danych wejściowych przez sieć ART-2 ma złożoność czasową zależną od ilości klas w sieci. Ta jednak z góry jest ograniczana.

Złożoność modułu MoG, gdzie w sposób zrównoleglony można sprawdzać przynależność punktu do wszystkich wyznaczonych obszarów, ma złożoność stałą.

Zastosowanie projekcji stereograficznej to również stały koszt czasu.

Tak więc po zakończeniu działania modułu VBMoG, proces przetwarzania jednego punktu danych ma w całości stałą złożoność czasową, co pozwala przetwarzać strumień danych z dowolnie wysokim ich próbkowaniem.

Proces kontrolny występujący co chwilę θ_{DeltaT} ma pesymistyczną złożoność zbliżoną do algorytmu VBEM. Ten pesymistyczny wariant występuje tylko wówczas gdy algorytm EM wyznacza nowy obszar MoG. Czas wykonania algorytmu EM dla modelu MoG jest niższy od VBMoG z uwagi na operowanie mniej skomplikowanym modelem danych.

6. Inteligentny monitoring turbin wiatrowych

6.1. Zadanie monitoringu turbin wiatrowych

6.1.1. Znaczenie monitoringu w eksploatacji turbin wiatrowych

Energetyka wiatrowa jest w ostatnich latach najszybciej rozwijającą się gałęzią energetyki, nie tylko na świecie, ale także na terenie Unii Europejskiej, w tym w Polsce [27, 41, 60]. Udział źródeł odnawialnych w produkcji energii elektrycznej na świecie jest szacowany powyżej 25% w 2035r, gdzie jedna czwarta cześć ma przypaść na energię wiatrową [34]. Struktura kosztów w czasie eksploatacji turbiny wiatrowej jest znacząco różna w stosunku do energetyki konwencjonalnej. Koszty inwestycji są relatywnie wysokie, podczas gdy w jednostkach konwencjonalnych znaczącą rolę odgrywa koszt paliwa. Po uruchomieniu największe koszta turbin wiatrowych stanowi ich utrzymanie. Potrzebne jest zatem właściwe planowanie przeglądów i remontów obecnych i przyszłych instalacji oraz skuteczne wykrywanie usterek i awarii na możliwie wczesnym etapie. Dzięki poprawnej polityce utrzymania i konserwacji turbiny wiatrowej może być ona wykorzystywana prawie przez cały czas, osiągając dostępność nawet do 98%. Podstawą poprawnego utrzymania/konserwacji jest ciągły monitoring pracy turbiny wiatrowej.

6.1.2. Techniki monitorowania turbin wiatrowych

Rysunek 6.1 [2, 33] prezentuje wybrany układ turbiny wiatrowej. Jest on relatywnie często spotykany, a przy tym dobrze ilustruje podstawowe elementy mechaniczne turbiny wiatrowej z poziomą osią





obrotu. W praktyce spotyka się jeszcze wiele innych rozwiązań, różniących się rodzajem i konstrukcją samej turbiny, rodzajem i konstrukcją przekładni głównej, rodzajem użytego generatora. Główny rotor z trzema łopatami jest wspierany przez główne łożysko i przekazuje moment obrotowy do przekładni planetarnej [32]. Wejście planetarnej przekładni jest koszykiem, z którym złączony jest główny wirnik. Przekładania planetarna posiada trzy planety z wałami przyłączonymi do koszyka. Planety toczą się po kole pierścieniowym i przekazują moment obrotowy do koła słonecznego. Wał słoneczny jest wyjściem przekładni. Napędza on dwustopniową równoległą przekładnię. Równoległa przekładnia ma trzy wały: wał wolny połączony do wału słonecznego, wał średni i wał szybkoobrotowy, który napędza generator. Generator przekształca energię mechaniczną turbiny wiatrowej na energię elektryczną. Parametry tej energii (napięcie, częstotliwość) zwykle uniemożliwiają bezpośrednie dołączenie takiego generatora do sieci. Niezbędny jest więc odpowiedni przekształtnik, dla przekształcenia tych parametrów do wymaga-nych wartości. Istnieje wiele różnych rozwiązań zarówno samych generatorów, jak i przekształtników.

W turbinie wiatrowej monitoringiem objęty jest układ napędowy, w tym główne łożysko, przekładnie i generator. Wszystkie z wymienionych komponentów zawierają kilka łożysk tocznych. Przekładnie i łożyska są najbardziej wrażliwymi częściami całej struktury i przez to monitoring jest przede wszystkim skierowany na nie. W celu wymuszenia najwyższych standardów w zakresie czynności konserwujących, od 2003 roku firmy ubezpieczeniowe wprowadziły na rynku europejskim certyfikację systemów monitorujących dla segmentu energetyki wiatrowej.

Od wielu lat coraz więcej turbin wiatrowych posiada system monitorujący stan jej pracy. Większość tego typu systemów wykorzystuje techniki analizy drgań składowych turbiny i analizy kluczowych zmiennych procesowych [2, 22]. Liczba czujników wibracyjnych zależy w ogólności od projektu turbiny. Istnieje jednak kilka powszechnie znanych ich konfiguracji, jednak najbardziej popularna zawiera 8 czujników wibracyjnych (zgodnie z rysunkiem 6.1). Czujniki G1 i G2 są używane do monitorowania struktury wibracji gondoli i wieży, na której turbina jest osadzona, natomiast czujniki T1-T6 mierzą wibracje układu przeniesienia napędu. Na tych samych instalacjach turbin jest możliwe połączenie czujników G1 i T1 oraz G2 i T2, wówczas tylko 6 czujników jest wystarczających dla zadania monitoringu. Wszystkie czujniki wibracyjne są akcelerometrami, w większości przypadków z wyjściem ICP (*Integrated Circuit Piezoelectric*). Monitorowanymi zmiennymi procesowymi są m.in. moc wyjściowa, prędkość wiatru oraz prędkość obrotowa generatora. Dodatkowo monitorowane mogą być również: temperatura otoczenia oraz łożysk, a także temperatura oleju i zwojów generatora.

6.1.3. Wiatr jako źródło zmienności obciążenia turbiny wiatrowej

Wiatr wpływa bezpośrednio na obciążenie każdej turbiny wiatrowej. Jego zmienność oddziaływuje więc wprost na stan pracy turbiny. W pracy [3] opisany został model analityczny do opisu prędkości wiatru. Model ten polega na przedstawieniu serii danych prędkości wiatru w postaci odcinków, w których każdy jest przybliżany równaniem 6.1. W równaniu tym występują dwie składowe: funkcja liniowa at + b oraz reszta równania, którą stanowi funkcja Weierstrassa.

$$u(t) = at + b + \sum_{k=1}^{K} \lambda^{(s-2)k} \sin\left(\lambda^{k} x(t)\right) + C.$$
(6.1)

W pracy [11] jest zaprezentowany sposób użycia tego modelu do generacji serii danych prędkości wiatru o zadanych parametrach. Wskazana praca umożliwia także symulowanie obciążenia zadanej turbiny wiatrowej, dzięki czemu możliwa jest również analiza pracy turbiny na podstawie jej całościowego modelu.

Model z równania (6.1), ze względu na obecność fraktalnej składowej, potwierdza chaotyczny charakter zmian prędkości wiatru.

6.1.4. Przetwarzanie danych pochodzących z monitoringu turbiny wiatrowej

Obciążenie każdej turbiny ma charakter chaotyczny z uwagi na własności wiatru (podrozdział 6.1.3). Wynika z tego potrzeba używania inteligentnych systemów monitorujących pracę turbiny wiatrowej.

Bazując na obecnie dostępnej literaturze [36, 48, 29, 19, 25, 20, 21, 38, 67, 43, 44] można stwierdzić, że większość dotychczasowych systemów monitoringu turbin wiatrowych nie uwzględnia w pełni ich zmiennego obciążenia. Większość działających obecnie lub proponowanych automatycznych systemów monitoringu polega najczęściej na wyznaczeniu stałych progów alarmowych w oparciu o następujące metody przetwarzania danych pomiarowych:

- analiza statystyczna,
- analiza trendów,
- analiza spektralna, analiza rzędów.

W kolejnym podrozdziale 6.2 omówione zostaną metody użycia sieci neuronowej typu ART-2 dla potrzeb inteligentnego monitoringu turbin wiatrowych. Autor prowadził badania w tym zakresie [5, 6, 8, 12, 7]. Następnie w podrozdziale 6.3 zostanie przedstawiony nowy inteligentny system monitorujący pracę turbiny wiatrowej jako implementacja systemu hybrydowego opisanego w rozdziale 5 na potrzeby specyfiki pracy turbin wiatrowych.

Rozwiązania zaprezentowane w podrozdziałach 6.2 i 6.3 można zaklasyfikować do grupy nowatorskich technik (*novel technics*) monitoringu turbin wiatrowych, wśród których w ostatnich latach pojawiło się wiele różnych realizacji [61, 40, 76, 75, 78, 35, 56, 71, 59, 26, 57, 31]. Tą grupę nowatorskich technik można skategoryzować następująco na [58, 77]:

- Fault detection and diagnosis systems (FDD),
- Systemy sztucznej inteligencji (AI Artificial intelligence),
- Systemy ekspertowe.

Przedstawione w tej dysertacji rozwiązania wzbogacają wymienioną kategorię systemów sztucznej inteligencji. Wyróżniają się jednak od innych realizacji w tej kategorii, które głównie cechują się trenowaniem danego systemu, a następnie predykcją za jego pomocą poprawnego stanu turbiny wiatrowej, w jakim powinna się w określonym czasie znajdować [77]. Choć takie systemy mogą uwzględniać zmienne środowisko pracy turbiny, a co za tym idzie zmienne jej obciążenie, to jednak wymagają czasochłonnego procesu nauczania nadzorowanego, które jest utrudnieniem przy przetwarzaniu danych w czasie rzeczywistym trwającym dłuższy okres czasu. W takim przypadku, nieaktualne mogą stać się dane treningowe - założenia dotyczące zarówno warunków środowiskowych w jakich pracuje turbina wiatrowa, jak i parametrów związanych ze stanem jej komponentów, które podlegają ciągłej eksploatacji. Wówczas możliwe jest, że wykonanie ponownie procesu nauczania takiego systemu jest konieczne dla jego poprawnego działania. Rozwiązania zaprezentowane w podrozdziałach 6.2 i 6.3 wykorzystują sieć neuronową ART-2, której nienadzorowane nauczanie jest zawarte w czasie przetwarzania sygnałów wejściowych przez tą sieć. System hybrydowy z podrozdziału 6.3 nie prognozuje jedynego poprawnego stanu turbiny wiatrowej, ale rejestruje różne, już zaistniałe jej stany, zgłaszając znalezione nowe położenia stanów jako ostrzegawcze sygnały wyjściowe.

6.2. Sieć ART-2 jako klasyfikator stanów operacyjnych turbiny wiatrowej

6.2.1. Diagnoza stanów awaryjnych dla określonego przedziału danych pomiarowych

W pracy [5] pokazano, że sieć ART-2 klasyfikuje poprawnie stany turbiny wiatrowej rozpatrując jej parametry podstawowe: prędkość wiatru, prędkość obrotową oraz moc wyjściową turbiny, przy użyciu skalowania danych opisanego w podrozdziale 2.4.4.2. W publikacji [8] przedstawiono symulację wykorzystania sieci ART-2 jako narzędzia diagnostycznego, dla danych pomiarowych obejmujących oprócz parametrów podstawowych jeden z dostępnych kanałów wibracyjnych. Wynika z tych badań, że sieć ART-2 klasyfikuje stany operacyjne turbiny w sposób stabilny, a ilość rozpoznanych stanów rośnie wraz ze wzrostem różnych zaburzeń w danych. Pozwala na prowadzenie poprawnej diagnostyki przy użyciu sieci ART-2.

6.2.2. Klasyfikacja on-line danych pomiarowych

Wśród niektórych badanych danych pomiarowych pochodzących z turbin wiatrowych zauważono, że dochodzi do przypadku występowania problemu podziału przestrzeni danych przez sieć ART-2 (patrz podrozdział 2.4.1). Wówczas do systemów opartych o sieć ART-2 potrzebne jest dodanie modułu projekcji stereograficznej opisanej w podrozdziałe 2.4.4.1. Taki moduł został skutecznie zintegrowany z siecią ART-2 w publikacji [7].

W pracy [12] wprowadzony został system hybrydowy złożony z sieci ART-2 oraz modułu obszarów OCC opisanych w rozdziale 3. Wprowadzony został tam uproszczony algorytm znany z rozdziału 5.4. Pozbawiony jest jednak m.in. początkowego modułu VBEM oraz możliwości adaptacji współczynnika czułości sieci ART-2. Pomimo tego, taki uproszczony system był w stanie przeprowadzić poprawną klasyfikację on-line stanu turbiny wiatrowej analizując wybrane dane złożone z trzech parametrów operacyjnych i jednego kanału wibracyjnego.

6.3. Autorski system jako inteligentne narzędzie realizujące zadanie monitoringu turbiny wiatrowej

W niniejszej pracy na bazie doświadczeń [5, 6, 8, 12, 7] zostaje wprowadzony hybrydowy system opisany w rozdziałe 5, a korzystający z rozwiązań omówionych w rozdziałach 2-4. Prace [5, 6, 8, 12, 7] pokazały, że dobrze skonfigurowana sieć ART-2 z użyciem odpowiedniego preprocesingu danych potrafi klasteryzować stany operacyjne turbiny wiatrowej. Celem niniejszej pracy jest zaproponowanie systemu automatyzującego cały proces monitoringu. Dotychczas sieć ART-2 była wstępnie konfigurowana przy udziale eksperta, gdzie konfiguracja polegała na wybraniu odpowiedniego parametru czułości sieci ρ . Wcześniej, raz ustawiona sieć ART-2, nie ulegała już modyfikacjom. Rozważając jednak system w pełni automatyczny, który będzie w stanie monitorować pracę turbiny przez długi czas, można założyć, że charakter danych pomiarowych będzie się zmieniał, przez co wartość parametru czułości sieci ART-2 również powinna być zmienna. Poniżej opisany zostanie sposób automatyzacji tych zagadnień jako implementacja zdefiniowanego w niniejszej pracy systemu hybrydowego. Tak utworzony system monitoringu został z powodzeniem przetestowany na dostępnych autorowi danych pomiarowych dotyczących turbin wiatrowych. Wyniki testów na trzech różnych zbiorach danych znajdują się w rozdziale 7.

6.3.1. Dobór parametrów systemu i początkowy etap pracy przy turbinie wiatrowej

Inicjalizacja systemu hybrydowego wprowadzonego w niniejszej pracy może być dokonywana automatycznie, bez nadzoru eksperta. Służy ku temu moduł VBEM opisany w rozdziale 4. Parametrem początkowym jest θ_{StartT} , który określa czas rozruchu turbiny, w którym dane pomiarowe są jedynie gromadzone i po którym rozpoczyna się monitorowanie. Dodatkowo mogą zostać zdefiniowane parametry $\theta_{MinIloscKlastrow}$ oraz $\theta_{MaxProporcjallosciKlastrow}$, które służą do określenia przedziału $[\theta_{MinIloscKlastrow}, \theta_{MaxIloscKlastrow}]$. Przedział ten jest stały i wyznacza stopień wrażliwości systemu.

6.3.2. Algorytm wczesnego rozpoznawania stanu przedawaryjnego przy użyciu modułu OCC

Logicznie przedział [$\theta_{MinIloscKlastrow}$, $\theta_{MaxIloscKlastrow}$] odpowiada za określenie minimalnej i maksymalnej ilości stanów operacyjnych turbiny w danym przedziale czasowym. Jeśli system zaobserwuje poprzez sieć ART-2, że w turbinie wiatrowej od pewnego już czasu (θ_{DeltaT}) stany operacyjne pozostają w tej samej konfiguracji (ich ilość się nie zmieniła), wówczas określa to jako sytuację stabilną i zapamiętuje za pomocą modułu OCC opisanego w rozdziale 3. Następnie sieć ART-2, porzucając poprzednio zapamiętane wzorce, analizuje tylko nowe stany i poprzez przedział [$\theta_{MinIloscKlastrow}$, $\theta_{MaxIloscKlastrow}$] dobiera nową wartość współczynnika czułości sieci ART-2. System dostosowuje się więc do zmieniających się warunków pracy turbiny, co ma naturalnie miejsce w czasie jej eksploatacji. Ilość analizowanych stanów zależy od wymiarowości danych pomiarowych. Początkowemu parametrowi $\theta_{MaxProporcjaIlosciKlastrow}$ należy więc ustawić tym wyższą wartość im dane pomiarowe mają więcej składowych, które powodują większą ilość możliwych teoretycznie stanów pracy.

7. Weryfikacja działania hybrydowego systemu neuronalnego przy użyciu danych z monitoringu turbiny wiatrowej

W tym rozdziale zostanie zaprezentowana weryfikacja działania hybrydowego systemu neuronalnego monitoringu turbiny wiatrowej zaproponowanego w podrozdziale 6.3 przy użyciu trzech różnych zbiorów danych pomiarowych. Weryfikacja podzielona została na trzy części względem użytego zbioru danych. Dla każdej z części przedstawiona zostanie charakterystyka zbioru danych, konfiguracja systemu hybrydowego oraz rezultaty otrzymanych wyników. Na koniec nastąpi podsumowanie wszystkich części weryfikacji.

Posiadane dane pomiarowe, które posłużyły do weryfikacji, obejmują historyczne dane z farmy wiatrowej firmy SeaCom GmbH z miejscowości Herne w Niemczech:

- 1. Zbiór I dane pomiarowe obejmujące parametry podstawowe oraz zapisy z kanałów wibracyjnych,
- 2. Zbiór II dane pomiarowe obejmujące rozszerzone parametry podstawowe oraz odczyty czujników temperatury,
- 3. Zbiór III dane pomiarowe obejmujące podstawowe parametry pomiarowe.

Zbiór I i II są rejestrem przebiegu pracy turbin wiatrowych, w których wystąpiły usterki. Zaprezentowany w tej dysertacji system ma na celu zgłoszenie nietypowych zdarzeń w pracy turbin wiatrowych, jako zapowiedź powstania tych usterek. Natomiast Zbiór III ma na celu zbadanie zachowania systemu dla danych pozbawionych awarii, dla którego stany operacyjne turbiny wiatrowej były poprawne klasyfikowane przez sieć ART-2 [5].

W każdej z części weryfikacja funkcjonalna hybrydowego systemu polegała na sprawdzeniu jego działania na przedstawionych danych pomiarowych z danego zbioru danych. Każdorazowo przeprowadzono symulację, która polegała na odtworzeniu kolejno wszystkich danych pomiarowych, w taki sposób, że z chwilą początkową uruchomiony został system hybrydowy. Każdy punkt danych odtworzony był tylko raz zgodnie z czasem jego zarejestrowania. Każde zgłoszenie nowego zdarzenia przez system było rejestrowane. Wyróżniono dwa typy zdarzeń: powstanie nowego wzorca sieci ART-2 oraz wyznaczenie nowego obszaru poprzez OCC.



Rysunek 7.1: Wykres danych z jednego z kanałów wibracyjnych monitoringu turbiny wiatrowej

7.1. Weryfikacja - część I

7.1.1. Dane pomiarowe i parametry systemu hybrydowego

Zbiór danych I obejmuje kolejne 27060 punktów pomiarowych w czasie, które odpowiadają okresowi od 1 października 2009r. do 6 kwietnia 2010r. Częstotliwość próbkowania wynosi 1 na 10 minut. Każdy punkt danych posiada 3 składowe danych podstawowych (prędkość obrotowa, moc i prędkość wiatru), a także 6 składowych wibracyjnych (zarejestrowanych przez czujniki umiejscowione jako T1-T6 wg opisu z podrozdziału 6.1.2). W czasie trwania zapisu wystąpiła usterka jednego z łożysk w turbinie. Usterkę widać szczególnie w jednym z kanałów wibracyjnych, który pokazany jest na rysunku 7.1. W okolicy chwili czasowej 15000-go punktu pomiarowego wartość sygnału wibracyjnego drastycznie wzrosła co świadczy o usterce łożyska. Użyte dane pomiarowe można więc podzielić na trzy kategorie:

- Dane A zgromadzone w czasie bezawaryjnego działania turbiny wiatrowej,
- Dane B przedawaryjne, które sygnalizują powstanie usterki,
- Dane C po wystąpieniu usterki, reprezentujące działanie maszyny z uszkodzonym jednym łożyskiem.

System hybrydowy został zainicjalizowany z następującymi głównymi parametrami: $\theta_{StartT} = 1000$, $\theta_{DeltaT} = 1000 \ \theta_{MinIloscKlastrow} = 3$, $\theta_{MaxProporcjallosciKlastrow} = 2$.

7.1.2. Wyniki

System przez pierwsze 1000 jednostek czasu gromadził dane, aby następnie uruchomić moduł VBEM, który znalazł wśród zagregowanych danych 3 szczególne jej punkty. Znalezione punkty obrazuje



Rysunek 7.2: Wynik działania modułu VBEM dla 1000 początkowych punktów pomiarowych. Kolorem czerwonym oznaczono znalezione szczególne punkty danych. Wykres przedstawia wybrane 3 z 9-ciu analizowanych składowych danych: moc, prędkość wiatru i wartości 6-tego kanału wibracyjnego.

rysunek 7.2. Po zakończeniu pracy modułu VBEM, system zainicjował sieć ART-2 parametrem czułości ρ wynoszącym 0.949. Od tego momentu system był gotowy do zgłaszania znalezionych nowych wzorców wśród danych oraz do wyznaczania stabilnych obszarów pracy turbiny. Wszystkie zaobserwowane zdarzenia wymienione są w tabeli 7.1.

Sporządzono również poglądowe rysunki 7.3-7.7 przedstawiające graficznie rezultaty klastrowania dla wybranych 3-ech z 9-ciu analizowanych składowych danych w wybranych jednostkach czasu.



Rysunek 7.3: Rezultat klastrowania danych pomiarowych 9-cio wymiarowych po 1583 jednostkach czasu. Wykres przedstawia wybrane 3 z 9-ciu analizowanych składowych danych: moc, prędkość wiatru i wartości 6-tego kanału wibracyjnego. Punkty oznaczono kolorami względem wzorców sieci ART-2, do których zostały one zaklasyfikowane.



Rysunek 7.4: Rezultat klastrowania danych pomiarowych 9-cio wymiarowych po 2242 jednostkach czasu. Wykres przedstawia wybrane 3 z 9-ciu analizowanych składowych danych: moc, prędkość wiatru i wartości 6-tego kanału wibracyjnego. Punkty oznaczono kolorami względem wzorców sieci ART-2, do których zostały one zaklasyfikowane.



Rysunek 7.5: Rezultat klastrowania danych pomiarowych 9-cio wymiarowych po 13100 jednostkach czasu. Wykres przedstawia wybrane 3 z 9-ciu analizowanych składowych danych: moc, prędkość wiatru i wartości 6-tego kanału wibracyjnego. Punkty oznaczono kształtami względem obszaru OCC, do którego zostały zaklasyfikowane. Kółka obrazują wyznaczony obszar 1-szy, natomiast gwiazdki obszar 2-gi, który został wyznaczony w chwili 13100.



Rysunek 7.6: Rezultat klastrowania danych pomiarowych 9-cio wymiarowych po 14921 jednostkach czasu. Wykres przedstawia wybrane 3 z 9-ciu analizowanych składowych danych: moc, prędkość wiatru i wartości 6-tego kanału wibracyjnego. Punkty oznaczono kształtami i kolorami. Kółka obrazują wyznaczony 1-szy obszar OCC, a gwiazdki obszar 2-gi. Resztę punktów oznaczono różnymi kolorami względem wzorców sieci ART-2, do których zostały one zaklasyfikowane.



Rysunek 7.7: Rezultat klastrowania danych pomiarowych 9-cio wymiarowych po wszystkich 27060 jednostkach czasu. Wykres przedstawia wybrane 3 z 9-ciu analizowanych składowych danych: moc, prędkość wiatru i wartości 6-tego kanału wibracyjnego. Punkty oznaczono kształtami i kolorami. Kółka obrazują wyznaczony 1-szy obszar OCC, gwiazdki obszar 2-gi, a krzyżyki obszar 3-ci. Resztę punktów oznaczono różnymi kolorami względem wzorców sieci ART-2, do których zostały one zaklasyfikowane. Z uwagi na małą liczebność punktów nie należących do obszarów 1-3, widoczne są głównie punkty koloru brązowego, oznaczającego wzorzec najczęściej używany przez sieć ART-2 w ostatniej fazie symulacji.

j. czasu	OCC	wzorce	notyfikacje systemu	
1000	0	3	3 wzorce ART-2 stworzone przez moduł VBEM	
1009	0	4	4-ty wzorzec został dodany do sieci ART-2	
1037	0	5	5-ty wzorzec został dodany do sieci ART-2	
1038	0	6	6-ty wzorzec został dodany do sieci ART-2	
1583	0	7	7-ty wzorzec został dodany do sieci ART-2	
1601	0	7	obniżenie współczynnika czułości ART-2 do wartości 0.944	
2240	0	8	8-ty wzorzec został dodany do sieci ART-2	
2242	0	9	9-ty wzorzec został dodany do sieci ART-2	
2251	0	9	obniżenie współczynnika czułości ART-2 do wartości 0.939	
4100	1	0	ustanowiony nowy obszar OCC z 4099 punktów	
4100	1	1	1-szy wzorzec został dodany do sieci ART-2	
4541	1	2	2-gi wzorzec został dodany do sieci ART-2	
4977	1	3	3-ci wzorzec został dodany do sieci ART-2	
6137	1	4	4-ty wzorzec został dodany do sieci ART-2	
8182	1	5	5-ty wzorzec został dodany do sieci ART-2	
8281	1	6	6-ty wzorzec został dodany do sieci ART-2	
13100	2	0	ustanowiony nowy obszar OCC z 500 punktów	
13100	2	1	1-szy wzorzec został dodany do sieci ART-2	
13109	2	2	2-gi wzorzec został dodany do sieci ART-2	
13577	2	2	podwyższenie współczynnika czułości ART-2 do wartości 0.945	
13835	2	2	podwyższenie współczynnika czułości ART-2 do wartości 0.950	
14295	2	3	3-ci wzorzec został dodany do sieci ART-2	
14921	2	4	4-ty wzorzec został dodany do sieci ART-2	
16200	3	0	ustanowiony nowy obszar OCC z 1512 punktów	
16200	3	1	1-szy wzorzec został dodany do sieci ART-2	
18968	3	1	podwyższenie współczynnika czułości sieci ART-2 do wartości 0.955	
19706	3	1	podwyższenie współczynnika czułości sieci ART-2 do wartości 0.960	
21688	3	1	podwyższenie współczynnika czułości sieci ART-2 do wartości 0.964	
21755	3	1	podwyższenie współczynnika czułości sieci ART-2 do wartości 0.967	
22130	3	1	podwyższenie współczynnika czułości sieci ART-2 do wartości 0.971	
22513	3	1	podwyższenie współczynnika czułości sieci ART-2 do wartości 0.974	
23599	3	1	podwyższenie współczynnika czułości sieci ART-2 do wartości 0.976	
23858	3	1	podwyższenie współczynnika czułości sieci ART-2 do wartości 0.979	
24397	3	2	2-gi wzorzec został dodany do sieci ART-2	
24408	3	3	3-ci wzorzec został dodany do sieci ART-2	
25345	3	4	4-ty wzorzec został dodany do sieci ART-2	
26286	3	5	5-ty wzorzec został dodany do sieci ART-2	

Tablica 7.1: Zapis przebiegu pracy hybrydowego systemu neuronalnego dla rzeczywistych danych pomiarowych ze zbioru I.

7.1.3. Dyskusja wyników

Znając czas wystąpienia usterki (ok. 15000 jednostki czasu), od systemu nadzorującego stan pracy danej maszyny wymaga się przede wszystkim, żeby stan przed i po wystąpieniu problemu był jednoznacznie wyróżniony. Oceniając przebieg opisanej symulacji widać, że ten warunek jest w pełni spełniony. Co więcej, przedstawiony w tej pracy system hybrydowy, daje podczas symulacji wiele sygnałów, mówiących o zmianie charakteru napływających danych przed wystąpieniem awarii. Do tych sygnałów zaliczają się informacje o powstaniu nowych wzorców w sieci ART-2 oraz wyznaczeniu obszarów OCC. System jednoznacznie wydzielił obszar 1-szy i 2-gi, gdzie obszar 1-szy oznacza niezmienną pracę turbiny trwającą parę miesięcy - Dane A. Obszar 2-gi zapowiada mające nastąpić zmiany w stanach operacyjnych turbiny - Dane B. Następowało tutaj zużycie uszkodzonego łożyska. Obszar 3-ci składa się wyłącznie z Danych C. Oprócz poprawnego wyznaczenia obszarów OCC widać, że system we właściwy sposób używał funkcjonalności zmniejszania i zwiększania wartości współczynnika czułości sieci ART-2. Zwiększenie czułości w j. czasu 13577 oraz 13835 umożliwiło dodatkowe zaalarmowanie w j. czasu 14295 i 14921, czyli tuż przed i w czasie występowania usterki.

7.2. Weryfikacja - część II

7.2.1. Dane pomiarowe i parametry systemu hybrydowego

Zbiór danych II obejmuje kolejne 4320 punktów pomiarowych w czasie, które odpowiadają okresowi 1-30 listopada 2012r. Źródłowe odczyty wykonywane były co sekundę, jednak częstotliwość ich rejestracji wynosiła 1 na 10 minut. Co 10 minut dla ostatnich 600 sekund zostały wyliczone wartości minimalne, maksymalne, średnie i odchylenia standardowe dla następujących parametrów operacyjnych: moc czynna, moc bierna, prędkość wiatru, prędkość wirnika, prędkość generatora, napięcie elektryczne, natężenie prądu, temperatura generatora (dwa osobne pomiary), temperatura skrzyni biegów. Każdy punkt danych posiada oprócz tych 4-ech wartości dla wymienionych 10 parametrów ma także następujące wartości składowe: dostarczona moc czynna, odebrana moc czynna, dostarczona moc bierna oraz otrzymana moc bierna. Daje to 44 składowe dla każdego punktu danych.

W czasie trwania zapisu wystąpiła usterka, która była powodem zatrzymania turbiny pomiędzy 1231wszą a 1265-tą jednostką czasu. Po 849-tej jednostce czasu oba czujniki temperatury generatora odnotowywały wysoko podwyższone wartości. Obrazuje to rysunek 7.8, przedstawiający wykres średniej wartości temperatury mierzonej przez jeden z czujników generatora. W tym czasie nastąpiła także podwyższona wartość prędkości wiatru - patrz rysunek 7.9. Patrząc jednak na cały dostępny zapis temperatury i prędkości wiatru, wzrost temperatury jest wyraźnie większy od wzrostu prędkości wiatru po 849-tej jednostce czasu.

System hybrydowy został zainicjalizowany z następującymi głównymi parametrami: $\theta_{StartT} = 500$, $\theta_{DeltaT} = 300 \ \theta_{MinIloscKlastrow} = 2$, $\theta_{MaxProporcjallosciKlastrow} = 2$.



Rysunek 7.8: Wykres danych z jednego z czujników temperatury umieszczonego na generatorze. Na wykresie pionowa niebieska linia oznacza 849-tą jednostkę czasu, natomiast linia czerwona jednostkę 1231-wszą, w której turbina została wyłączona.





7.2.2. Wyniki

System przez pierwsze 500 jednostek czasu gromadził dane, aby następnie uruchomić moduł VBEM, który znalazł wśród zagregowanych danych 2 szczególne jej punkty. Po zakończeniu pracy modułu VBEM, system zainicjował sieć ART-2 parametrem czułości ρ wynoszącym 0.998. Od tego momentu system był gotowy do zgłaszania znalezionych nowych wzorców wśród danych oraz do wyznaczania stabilnych obszarów pracy turbiny. Wszystkie zaobserwowane zdarzenia wymienione są w tabeli 7.2.

j. czasu	OCC	wzorce	notyfikacje systemu
500	0	2	2 wzorce ART-2 stworzone przez moduł VBEM
516	0	3	3-ci wzorzec został dodany do sieci ART-2
650	0	4	4-ty wzorzec został dodany do sieci ART-2
797	0	5	5-ty wzorzec został dodany do sieci ART-2
820	0	5	obniżenie współczynnika czułości ART-2 do wartości 0.996
850	0	6	6-ty wzorzec został dodany do sieci ART-2
860	0	6	obniżenie współczynnika czułości ART-2 do wartości 0.995
1231	0	7	7-my wzorzec został dodany do sieci ART-2
1246	0	8	8-my wzorzec został dodany do sieci ART-2
1260	0	8	obniżenie współczynnika czułości ART-2 do wartości 0.992
1820	1	0	ustanowiony nowy obszar OCC z 1820 punktów
1821	1	1	1-szy wzorzec został dodany do sieci ART-2
2182	1	2	2-gi wzorzec został dodany do sieci ART-2
2655	1	3	3-ci wzorzec został dodany do sieci ART-2
2998	1	4	4-ty wzorzec został dodany do sieci ART-2
3430	2	0	ustanowiony nowy obszar OCC z 129 punktów
3436	2	1	1-szy wzorzec został dodany do sieci ART-2
3603	2	2	2-gi wzorzec został dodany do sieci ART-2
4044	3	0	ustanowiony nowy obszar OCC z 133 punktów
4074	3	1	1-szy wzorzec został dodany do sieci ART-2
4255	3	2	2-gi wzorzec został dodany do sieci ART-2

Tablica 7.2: Zapis przebiegu pracy hybrydowego systemu neuronalnego dla rzeczywistych danych pomiarowych ze zbioru II.

Sporządzono poglądowe rysunki 7.10-7.13 przedstawiające graficznie rezultaty klastrowania dla wybranych 3-ech z 44-ech analizowanych składowych danych w wybranych jednostkach czasu.



Rysunek 7.10: Rezultat klastrowania danych pomiarowych 44-ro wymiarowych po 850 jednostkach czasu. Wykres przedstawia wybrane 3 z 44-ech analizowanych składowych danych: średnia moc czynna, średnia prędkość wiatru i średnia temperatura generatora. Punkty oznaczono kolorami względem wzorców sieci ART-2, do których zostały one zaklasyfikowane.



Rysunek 7.11: Rezultat klastrowania danych pomiarowych 44-ro wymiarowych po 1246 jednostkach czasu. Wykres przedstawia wybrane 3 z 44-ech analizowanych składowych danych: średnia moc czynna, średnia prędkość wiatru i średnia temperatura generatora. Punkty oznaczono kolorami względem wzorców sieci ART-2, do których zostały one zaklasyfikowane. Punkty szare powstały po 1231 jednostce czasu.



Rysunek 7.12: Rezultat klastrowania danych pomiarowych 44-ro wymiarowych po 1821 jednostkach czasu. Wykres przedstawia wybrane 3 z 44-ech analizowanych składowych danych: średnia moc czynna, średnia prędkość wiatru i średnia temperatura generatora. Kółka oznaczają punkty należące do 1-wszego wyznaczonego obszaru OCC.



Rysunek 7.13: Rezultat klastrowania danych pomiarowych 44-ro wymiarowych po wszystkich 4320 jednostkach czasu. Wykres przedstawia wybrane 3 z 44-ech analizowanych składowych danych: średnia moc czynna, średnia prędkość wiatru i średnia temperatura generatora. Punkty oznaczono kształtami i kolorami. Kółka obrazują wyznaczony 1-szy obszar OCC, gwiazdki obszar 2-gi, a iksy obszar 3-ci. Resztę punktów oznaczono różnymi kolorami względem wzorców sieci ART-2, do których zostały one zaklasyfikowane.

7.2.3. Dyskusja wyników

Usterka i zatrzymanie turbiny objawiło się niedługo po rozpoczęciu rejestrowania danych pomiarowych, stąd system nie miał czasu na wyodrębnienie stabilnego stanu pracy turbiny wiatrowej za pomocą obszaru OCC. Pomimo tego system poprawnie rozpoznał stan pracy turbiny związany ze wzrostem temperatury na generatorze - notyfikacja sieci ART-2 w 850 jednostce czasu (kolor żółty na rysunkach 7.10-7.11). Stan zatrzymania został także odnotowany poprzez pojawienie się wzorca sieci ART-2 w 1231-wszej jednostce czasu (kolor szary na rysunku 7.11). System wyznaczając pierwszy obszar OCC w 1820 jednostce czasu, objął wszystkie do tamtej pory występujące stany pracy turbiny, włącznie z wystąpieniem usterki. Później, wyróżniających się, nowych stanów pracy turbiny nie odnotowano, dlatego prawie cała reszta punktów pomiarowych została ostatecznie zaklasyfikowana właśnie przez ten pierwszy obszar.

7.3. Weryfikacja - część III

7.3.1. Dane pomiarowe i parametry systemu hybrydowego

Zbiór danych III obejmuje kolejne 2869 punktów pomiarowych w czasie, które odpowiadają okresowi 11-30 września 2009r. Częstotliwość próbkowania wynosi około 1 na 10 minut. Każdy punkt danych posiada 3 składowe danych podstawowych: prędkość obrotową, moc i prędkość wiatru. W czasie trwania zapisu nie wystąpiła żadna usterka.

System hybrydowy został zainicjalizowany z następującymi głównymi parametrami: $\theta_{StartT} = 500$, $\theta_{DeltaT} = 300 \ \theta_{MinIloscKlastrow} = 2$, $\theta_{MaxProporcjallosciKlastrow} = 2$.

7.3.2. Wyniki

System przez pierwsze 500 jednostek czasu gromadził dane, aby następnie uruchomić moduł VBEM, który znalazł wśród zagregowanych danych 2 szczególne jej punkty. Znalezione punkty obrazuje rysunek 7.14. Po zakończeniu pracy modułu VBEM, system zainicjował sieć ART-2 parametrem czułości ρ wynoszącym 0.992. Od tego momentu system był gotowy do zgłaszania znalezionych nowych wzorców wśród przetwarzanych danych oraz do wyznaczania stabilnych obszarów pracy turbiny. Wszystkie zaobserwowane zdarzenia wymienione są w tabeli 7.3.

Sporządzono rysunki 7.15-7.16 przedstawiające graficznie rezultaty klastrowania w wybranych jednostkach czasu.



Rysunek 7.14: Wynik działania modułu VBEM dla 400 początkowych punktów pomiarowych. Kolorem czerwonym oznaczono znalezione szczególne punkty danych. Wykres przedstawia analizowane składowe danych: moc, prędkość wiatru i prędkość obrotowa.



Rysunek 7.15: Rezultat klastrowania danych pomiarowych 3 wymiarowych po 799 jednostkach czasu. Wykres przedstawia analizowane składowe danych: moc, prędkość wiatru i prędkość obrotowa. Punkty oznaczono kolorami względem wzorców sieci ART-2, do których zostały one zaklasyfikowane.

j. czasu	OCC	wzorce	notyfikacje systemu
400	0	2	2 wzorce ART-2 stworzone przez moduł VBEM
426	0	3	3-ci wzorzec został dodany do sieci ART-2
523	0	4	4-ty wzorzec został dodany do sieci ART-2
600	1	0	ustanowiony nowy obszar OCC z 800 punktów
881	1	1	1-szy wzorzec został dodany do sieci ART-2
884	1	2	2-gi wzorzec został dodany do sieci ART-2

Tablica 7.3: Zapis przebiegu pracy hybrydowego systemu neuronalnego dla rzeczywistych danych pomiarowych ze zbioru III.



Rysunek 7.16: Rezultat klastrowania danych pomiarowych 3 wymiarowych po wszystkich 2869 jednostkach czasu. Wykres przedstawia analizowane składowe danych: moc, prędkość wiatru i prędkość obrotowa. Punkty oznaczono kształtami i kolorami. Kółka obrazują wyznaczony 1-szy obszar OCC. Resztę punktów oznaczono różnymi kolorami względem wzorców sieci ART-2, do których zostały one zaklasyfikowane.
7.3.3. Dyskusja wyników

System poprawnie rozpoznał stany operacyjne turbiny wiatrowej. Wyznaczone stany operacyjne w 799 jednostce czasu są zbliżone do stanów znalezionych przy użyciu samej sieci ART-2 dla tych samych danych w [5]. Jednak w przypadku systemu hybrydowego współczynnik czułości sieci ART-2 nie musiał zostać wybrany arbitralnie, został poprawnie ustalony poprzez moduł VBMoG. W danych pomiarowych nie nastąpiło żadne niespodziewane zjawisko, a system poprawnie to zinterpretował wyznaczając tylko jeden obszar OCC, do którego zaklasyfikowane zostały prawie wszystkie punkty danych.

7.4. Podsumowanie weryfikacji

W weryfikacji I system miał wystarczającą ilość czasu, żeby rozpoznać stabilne stany operacyjne turbiny wiatrowej w postaci jednego obszaru OCC, a następnie stany prowadzące do awarii. Wypełnił poprawnie zadanie. Dla zbioru II, czasu na rozpoznanie stabilizacji pracy turbiny brakowało, jednak pomimo to system umiał poprawnie zgłosić wystąpienie podejrzanego zdarzenia. Natomiast na zbiorze III system pokazał, że pomimo rozbudowanej swojej architektury potrafi proste, pozbawione cech szczególnych dane, klasteryzować poprawnie, ograniczając ilość zgłaszanych powiadomień.

Oprócz wysyłania poprawnych powiadomień o podejrzanych zdarzeniach w pracy turbiny wiatrowej, system pokazał, że klasyfikuje stany operacyjne turbiny wiatrowej poprawnie. Wyznaczane klastry za pomocą sieci ART-2 są spójne i funkcjonalnie poprawne. Natomiast obszary OCC, wyznaczane przy ustabilizowaniu się sieci ART-2, opisując stany pracy turbiny wiatrowej na wyższym poziomie, dają użytkownikowi mapę zdefiniowanych ogólnych stanów pracy maszyny. Stany OCC rozdzielają dane pomiarowe jednoznacznie na rozłączne grupy. Testy weryfikacyjne pokazały, że zaproponowana architektura systemu hybrydowego jest niewrażliwa na ilość przetwarzanych punktów danych oraz na wymiar przestrzeni, w jakiej te punkty się znajdują. System wykonywał klastrowanie przez sieci ART-2 z wykorzystywaniem modułu projekcji stereograficznej, jak i wyznaczał obszary OCC przy użyciu jednostek MoG przetwarzając poprawnie dane 3, 9-cio i 44-ro wymiarowe.

8. Wnioski końcowe

Niniejsza praca omawia zagadnienia sieci neuronowych ART-2, a także gaussowskich modeli gęstościowych MoG i VBMoG. Odpowiednie połączenie tych zagadnień tworzy system hybrydowy, stanowiący nowy teoretyczny system sztucznej inteligencji, którego celem jest klastrowanie wielowymiarowych danych w czasie rzeczywistym. Głównym modułem gwarantującym przetwarzanie danych jest sieć ART-2. W pracy wyjaśniono, że sama sieć ART-2, ze względu na zadawaną stałą arbitralną wartość współczynnika czułości ρ , nie może być traktowana jako niezależny system realizujący zadanie klastrowania występujące w warunkach analizy danych pomiarowych z turbin wiatrowych. Niezbędne są dodatkowe moduły odpowiedzialne za wstępne przetwarzanie danych (projekcja stereograficzna - rozdział 1.4), a także za kontrolowanie i manipulowanie współczynnikiem ρ (moduły MoG - rozdział 3, użyte w algorytmie omówionym w podrozdziale 5.4). System posiada również udogodnienie w postaci automatycznego nadawania wstępnej wartości współczynnikowi ρ (moduł VBEM - rozdział 4).

Przedstawiony w pracy hybrydowy system neuronalny został skutecznie zaaplikowany do problemu klastrowania danych pochodzących z monitoringu turbiny wiatrowej w celu wyznaczania nietypowych stanów operacyjnych turbiny, które mogą świadczyć o usterce lub wystąpieniu stanu przedawaryjnego. W ramach niniejszej dysertacji wykonano weryfikację funkcjonalną stworzonego systemu przy użyciu trzech zbiorów posiadanych danych. Pokazano, że może on być traktowany jako w pełni samodzielny system analizujący w czasie rzeczywistym parametry pracy turbiny wiatrowej, działający jako system wczesnego ostrzegania. Tym samym teza dysertacji została pozytywnie zweryfikowana. Należy podkreślić, że opracowany i zaimplementowany system może być również używany przez eksperta jako dodatkowe źródło informacji o bieżącym stanie, a także o archiwalnych ułożeniach stanów operacyjnych w jakich się znajdowała monitorowana maszyna. Hybrydowy system neuronalny potrafi przetwarzać dowolne dane liczbowe, a dzięki temu nie musi być traktowany wyłącznie jako alternatywa dla innych systemów monitoringu turbin wiatrowych. Możliwa jest taka konfiguracja systemów monitoringu, w której przedstawiony w tej pracy system zarządzałby informacjami przekazywanymi z innych systemów, które bazują na innych technikach i metodach przetwarzania danych. Wówczas hybrydowy system neuronalny analizowałby przetworzone już dane pomiarowe lub inne sygnały pochodzące z podpiętych do niego systemów. Takimi sygnałami mogą być np. informacje o przekroczonych progach ostrzegawczych dla konkretnego komponentu turbiny.

Opisany w niniejszej dysertacji hybrydowy system neuronalny inteligentnego monitoringu może być również użyty w gałęziach gospodarki innych niż energetyka wiatrowa, do monitorowania maszyn innego typu. Dobrym kandydatem wydaje się być, na przykład, górnictwo odkrywkowe i maszyny tam wykorzystywane, które podobnie jak turbiny wiatrowe pracują pod zmiennym obciążeniem. W związku z tym narażone są na częste awarie, dlatego też możliwość ich automatycznego, inteligentnego monitoringu jest intensywnie badana [68, 10, 79, 80, 69].

9. Dodatek

Implementacja hybrydowego systemu neuronalnego

System hybrydowy opisany w niniejszej dysertacji został zaimplementowany, włącznie ze wszystkimi jego składowymi wymienionymi w rozdziale 5.2, w postaci biblioteki języka programistycznego MATLAB. Implementacja pozwala na skorzystanie z systemu hybrydowego w całości lub niezależnie z jego poszczególnych składowych. Biblioteka została umieszczona jako projekt programistyczny "ART-2 Neural Network & Machine Monitoring Hybrid System" w serwisie internetowym GitHub. Zawartość można pobrać ze strony https://github.com/ mjwojcik/Art2MonitoringHybridSystem. Z uwagi na wykorzystany język programowania projekt został podłączony do serwisu internetowego Matlab File Exchange, gdzie dostępny jest pod adresem http://www.mathworks.com/matlabcentral/fileexchange/ 54377-art-2-neural-network---machine-monitoring-hybrid-system.

Do biblioteki dołączony jest jej krótki opis, który pełni rolę instrukcji użytkownika. Podane zostały przykłady użycia sieci ART-2 jak i całości systemu hybrydowego. Wraz z kodem źródłowym zostały opublikowane również testy automatyczne systemu hybrydowego w postaci testów end-to-end, jak również testy jednostkowe poszczególnych składowych tego systemu.

Bibliografia

- Aradhye, H. B., Davis, J. F. & Bakshi, B. R. (2002). Art-2 and multiscale art-2 for on-line process fault detection - Validation via industrial case studies and Monte Carlo simulation. Annual Reviews in Control 26 (1), 113-127
- [2] Barszcz, T. (2009). Application Of Diagnostic Algorithms For Wind Turbines. Diagnostyka 2(50), 7-11
- [3] Barszcz, T., Bielecka, M., Bielecki, A. & Wójcik, M. (2012). Wind speed modelling using Weierstrass function fitted by a genetic algorithm. Journal of Wind Engineering and Industrial Aerodynamics, vol. 109, 68–78
- [4] Barszcz, T., Bielecki, A. & Romaniuk, T. (2009). Application of probabilistic neural networks for detection of mechanical faults in electric motors. Electrical Review 8/2009, 37-41
- [5] Barszcz, T., Bielecki, A. & Wójcik, M. (2010). ART-Type Artificial Neural Networks Applications for Classification of Operational States in Wind Turbines. Lecture Notes in Artificial Intelligence, vol. 6114, 11-18.
- [6] Barszcz, T., Bielecki, A. & Wójcik, M. (2011). Wind Turbines States Classification by a Fuzzy-ART Neural Network with a Stereographic Projection as a Signal Normalization. Lecture Notes in Computer Science, vol. 6594, 225-234
- [7] Barszcz, T., Bielecki, A., Bielecka, M., Wójcik, M. & Włuka, M. (2016). Vertical axis wind turbine states classification by an ART-2 neural network with a stereographic projection as a signal normalization. In Advances in Condition Monitoring of Machinery in Non-Stationary Operations, 265-275
- [8] Barszcz, T., Bielecki, A., Wójcik, M. & Bielecka, M. (2013). Art-2 artificial neural networks applications for classification of vibration signals and operational states of wind turbines for intelligent monitoring. Diagnostyka, vol. 14, no 4, 21-26.
- [9] Barszcz, T. & Randall, R.B. (2009). *Application of spectral kurtosis for detection of a tooth crack in the planetary gear of a wind turbine*. Mechanical Systems and Signal Processing 23, 1352-1365
- [10] Bartelmus, W., Chaari, F., Zimroz, R. & Haddar M. (2010). Modelling of gearbox dynamics under time-varying nonstationary load for distributed fault detection and diagnosis. European Journal of Mechanics – A/Solids, 29/4, s. 637-646.

- [11] Bielecki, A., Barszcz, T. & Wójcik, M. (2015). Modelling of a chaotic load of wind turbines drivetrain. Mechanical Systems and Signal Processing, 54-55, 491-505
- [12] Bielecki, A., Barszcz, T., Wójcik, M. & Bielecka, M. (2014). Hybrid System of ART and RBF Neural Networks for Classification of Vibration Signals and Operational States of Wind Turbines. Lecture Notes in Computer Science, vol. 8467, 2014, 3-11
- [13] Bielecki, A., Bielecka, M. & Chmielowiec A. (2008). Input signals normalization in Kohonen neural networks. Lecture Notes in Artificial Intelligence, 5097, 3-10
- [14] Bishop, C.M. (2006). Pattern Recognition and Machine Learning (Information Science and Statistics). Springer-Verlag New York, Inc., Secaucus, NJ, USA.
- [15] Boyd, S. & Vandenberghe, L. (2004). Convex optimization. Cambridge university press.
- [16] Cao, Y. & Wu, J. (2002). Projective ART for clustering data sets in high dimensional spaces. Neural Networks 15 (1), 105-120
- [17] Carpenter, G. A. & Grossberg, S. (1987). ART2: self-organization of stable category recognition codes for analog input pattern. Applied Optics 26, 4919-4930
- [18] Carpenter, G. A., Grossberg, S. & Rosen, D. B. (1991). ART 2-A: An adaptive resonance algorithm for rapid category learning and recognition. Neural Networks 4 (4), 493-504
- [19] Caselitz, P. & Giebhardt, J. (2002), Advance maintenance and repair for offshore wind farms using fault prediction techniques. Institut fur Solare Energieversorgungstechnik (ISET), Division of Energy Conversion and Control Engineering Konigstor 59, D-34119 Kassel, Germany.
- [20] Cempel, C. (1987). Simple condition forecasting techniques in vibroacoustical diagnostics. Mechanical Systems and Signal Processing 1/1
- [21] Cempel, C. (1990). Limit value in practice of vibration diagnosis. Mechanical Systems and Signal Processing 4/6
- [22] Coronado, D., Fischer, K. (2015). *Condition monitoring of wind turbines: State of the art, user experience and recommendations*. Bremerhaven: Fraunhofer IWES, 2015, III, 83 pp.
- [23] Du, K. L. (2010). Clustering: A neural network approach. Neural Networks 23, 1, 89-107
- [24] Duda, R. & Hart, P. (1973). Pattern Classification and Scene Analysis. John Wiley & Sons, New York.
- [25] Duda, R.O., Hart, P.E. & Stork, D.G. (2000). Pattern Classification. John Wiley & Sons, ISBN: 0-471-05669-3.
- [26] Entezami, M., Hillmansen, S., Weston, P. & Papaelias M.Ph. (2012). Fault detection and diagnosis within a wind turbine mechanical braking system using condition monitoring. Renewable Energy, Volume 47, November 2012, Pages 175-182

- [27] Ezio, S. & Claudio, C. (1998). Exploitation of wind as an energy source to meet the world's electricity demand. Wind Engineering vol.74–76, 375–387
- [28] Fausett, L. (1994). Fundamentals of Neural Networks: Architectures, Algorithms, and Applications. Prentice-Hall, Inc., Upper Saddle River, NJ, USA.
- [29] Fung, K.T., Scheffler, R.L. & Stolpe, J. (1981). Wind energy—a utility perspective, IEEE Trans Power Apparatus. System, 100, 1176–1182
- [30] Gancarzewicz, J. (1987). Differential Geometry, . PWN, Warszawa (1987) (in Polish)
- [31] Garcia, M.C., Sanz-Bobi, M.A. & Javier del Pico. (2006). SIMAP: Intelligent System for Predictive Maintenance: Application to the health condition monitoring of a windturbine gearbox. Computers in Industry, Volume 57, Issue 6, Pages 552-568
- [32] Gasch, R. & Twele J. (2002). *Wind Power Plants Fundamentals, Design, Construction and Operation.* James & James.
- [33] Gellermann, T. (2003). Requirements for Condition Monitoring Systems for Wind Turbines. AZT Expertentage, 10-11.11.2003, Allianz 2003.
- [34] Global Wind Energy Council (2014). *Global Wind Report: Annual Market Update 2013*. Global Wind Energy Council.
- [35] Guolian, H., Pan, J., Zhentao W. & Jianhua Z. (2010). Research on fault diagnosis of wind turbine control system based on Artificial Neural Network. in Intelligent Control and Automation (WCICA), 2010 8th World Congress on , vol., no., pp.4875-4879, 7-9 July 2010
- [36] Hameeda, Z., Honga, Y.S., Choa, T.M., Ahnb, S.H. & Son, C.K. (2009). Condition monitoring and fault detection of wind turbines and related algorithms: A review. Renewable and Sustainable Energy Reviews 13, 1–39
- [37] Han, J., Kamber, M. & Pei, J. (2011). Data Mining: Concepts and Techniques (3rd ed.). Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.
- [38] Heng, A., Zhang, S., Tan, A.C.C. & Mathew, J. (2009). Rotating machinery prognostics: state of the art, challenges and opportunities. Mechanical Systems and Signal Processing 23/3
- [39] Holtzer, G. & Holtzer, M. (2012). Ustawa o odnawialnych źródłach energii. Przegląd Odlewnictwa, vol.62, 446-451
- [40] Huang, Q., Jiang, D., Hong, L. & Ding, Y. (2008). Application of wavelet neural networks on vibration fault diagnosis for wind turbine gearbox. in Proc. 5th Int. Symp. Neural Netw.: Advances in Neural Netw., Part II, Sept. 2008, pp. 313-320.
- [41] Igliński, B., Kujawski, W., Buczkowski, R. & Cichosz, M. (2010). Renewable energy in the Kujawsko-Pomorskie voivodeship. Renewable and Sustainable Energy Reviews vol.14, 1336–1341

- [42] Jabłoński, A. & Barszcz, T. (2012). Procedure for data acquisition for machinery working under non-stationary operational conditions,. The Ninth International Conference on Condition Monitoring and Machinery Failure Prevention Technologies, 12-14 June 2012, London
- [43] Jabłoński, A., Barszcz, T. & Bielecka, M. (2011). Automatic validation of vibration signals in wind farm distributed monitoring systems. Measurement, vol.44, 1954-1967
- [44] Jabłoński, A., Barszcz, T., Bielecka, M. & Brehaus, P. (2013). Modeling of probability distribution functions for automatic threshold calculation in condition monitoring systems. Measurement, vol.46, 727-738
- [45] Jain, A. K. & Dubes, R.C. (1988). Algorithms for Clustering Data. Prentice-Hall, Inc., Upper Saddle River, NJ, USA.
- [46] Jain, A. K., Murty, M. N. & Flynn, P. J. (1999). *Data clustering: a review*. ACM Comput. Surv. 31, 3, 264-323
- [47] Jain, A. K., Topchy, A., Law, M.H.C. & Buhmann, J. M. (2004). Landscape of clustering algorithms. Pattern Recognition, ICPR 2004. Proceedings of the 17th International Conference on , vol.1, no., pp.260,263 Vol.1, 23-26 doi: 10.1109/ICPR.2004.1334073
- [48] Joselin Herbert, G.M., Iniyan, S., Sreevalsan, E. & Rajapandian, S. (2007). A review of wind energy technologies. Renew Sustain Energy Rev, 11 (6), 1117–1145
- [49] Kim, Y.S. (2010). Performance evaluation for classification methods: A comparative simulation study. Expert Systems with Applications, vol.37 (3), 2292-2306
- [50] Korbicz, J., Obuchowicz, A. & Uciński, D. (1994). Sztuczne sieci neuronowe. Podstawy i Zastosowania. Akademicka Oficyna Wydawnicza PLJ, Warszawa
- [51] Kuo, R. J., Liao, J. L. & Tu, C. (2005). Integration of ART2 neural network and genetic K-means algorithm for analyzing Web browsing paths in electronic commerce. Decision Support Systems 40 (2), 355-374
- [52] Kusiak, A. & Li, W. (2011). The prediction and diagnosis of wind turbine faults. Renewable Energy, vol.36, 2011, 16-23.
- [53] Lal, K., Rak, T. & Orkisz, K. (2003). RTLinux system czasu rzeczywistego. HELION.
- [54] Lloyd, G. (2007). Richtlinien f
 ür die Zertifizierung von Condition Monitoring Systemen f
 ür Windenergieanlagen. Vorschriften und Richtlinien, Selbstverlag Germanischer Lloyd 2007.
- [55] Liu, L., Huang L., Lai, M. & Ma, C. (2009). Projective ART with buffers for the high dimensional space clustering and an application to discover stock associations. Neurocomputing 72 (4-6), 1283-1295

- [56] Lu, D. & Qiao W. (2013). Adaptive feature extraction and SVM classification for real-time fault diagnosis of drivetrain gearboxes. in Energy Conversion Congress and Exposition (ECCE), 2013 IEEE, vol., no., pp.3934-3940, 15-19 Sept. 2013
- [57] Manrique, R.F., Giraldo, F.A. & Esmeral, J.S. (2012). Fault detection and diagnosis for wind turbines using data-driven approach. in Computing Congress (CCC), 2012 7th Colombian , vol., no., pp.1-6, 1-5 Oct. 2012
- [58] Márquez, F.P.G., Tobias, A.M., Pérez, J.M.P. & Papaelias, M. (2012). Condition monitoring of wind turbines: Techniques and methods. Renewable Energy, Volume 46, 169-178
- [59] Martinez-Rego, D., Fontenla-Romero, O. & Alonso-Betanzos, A. (2011). Power wind mill fault detection via one-class v-SVM vibration signal analysis. in Neural Networks (IJCNN), The 2011 International Joint Conference on , vol., no., pp.511-518, July 31 2011-Aug. 5 2011
- [60] Paska, J., Sałek, M. & Surma, T. (2009). Current status and perspectives of renewable energy sources in Poland. Renewable and Sustainable Energy Reviews vol.13, 142–154.
- [61] Qiao, W. & Lu, D. (2015). A Survey on Wind Turbine Condition Monitoring and Fault Diagnosis—Part II: Signals and Signal Processing Methods. in Industrial Electronics, IEEE Transactions on, vol.62, no.10, pp.6546-6557, Oct. 2015
- [62] Rutkowski, L. (1996). Sieci neuronowe i neurokomputery. Wydawnictwo Politechniki Częstochowskiej, Częstochowa.
- [63] Rutkowski, T. (2015). Systemy Czasu Rzeczywistego, Rysy historyczny oraz podstawowe pojęcia i definicje. wykład, Politechnika Gdańska, Wydział Elektrotechniki i Automatyki. Pozyskano z http://eia.pg.edu.pl/documents/1111711/29851529/W02_SCR_ historia_podstawowe_pojecia_definicje.pdf
- [64] Shieh, M.D., Yan, W. & Chen, C.H. (2008). Soliciting customer requirements for product redesign based on picture sorts and ART2 neural network. Expert Systems with Applications 34, 194-204
- [65] Shuhui, L., Wunsch, D.C., O'Hair, E. & Giesselmann, M.G. (2001). Comparative analysis of regression and artificial neural network models for wind turbine power curve estimation,. Journal of Solar Energy Engineering 123, 327-332
- [66] Spurek, P., Tabor, J. & Wójcik. M. (2015). Cross-Entropy Clustering Approach to One-Class Classification. Lecture Notes in Computer Science, vol. 9119 (2015), 481-490
- [67] Strączkiewicz, M., Barszcz, T. & Jabłoński, A. (2015). Detection and classification of alarm threshold violations in condition monitoring systems working in highly varying operational conditions. Journal of Physics: Conference Series. Vol. 628. No. 1. IOP Publishing, 2015.
- [68] Stefaniak, P., Obuchowski, J., Sawicki, M., Zimroz, R., Żak, G., Wyłomańska, A. & Bartelmus, W. (2015). Wybrane problemy i wyzwania automatycznej diagnostyki elementów maszyn górniczych. Cuprum : czasopismo naukowo-techniczne górnictwa rud, nr 3, 189-198.

- [69] Stefaniak, P., Zimroz, R., Sliwinski, P., Andrzejewski, M. & Wyłomanska W. (2016). Multidimensional signal analysis for technical condition, operation and performance understanding of heavy duty mining machines. Advances in Condition Monitoring of Machinery in Non-Stationary Operations, serie: Applied Condition Monitoring, Vol. 4, 2016, str. 197-210
- [70] Tadeusiewicz, R. (1993). Sieci neuronowe. Akademicka Oficyna Wydawnicza RM, Warszawa
- [71] Tang, B., Song, T., Li, F. & Deng, L. (2014). Fault diagnosis for a wind turbine transmission system based on manifold learning and Shannon wavelet support vector machine. Renewable Energy, Volume 62, February 2014, Pages 1-9
- [72] Tax, D. (2001). One-class classification: Concept-learning in the absence of counter-examples. Doctoral Dissertation, University of Delft, The Netherlands.
- [73] Tax, D.M.J. (2013). DDtools, the Data Description Toolbox for Matlab
- [74] Ułasiewicz, J. (2007). Systemy czasu rzeczywistego. QNX6 Neutrino, BTC, Warszawa.
- [75] Wang, Y. & Infield, D.G. (2012). Neural network modelling with autoregressive inputs for wind turbine condition monitoring. in Sustainable Power Generation and Supply (SUPERGEN 2012), International Conference on , vol., no., pp.1-6, 8-9 Sept. 2012
- [76] Yang, S., Li, W. & Wang, C. (2008). The intelligent fault diagnosis of wind turbine gearbox based on artificial neural network. in Condition Monitoring and Diagnosis, CMD 2008. International Conference on , vol., no., pp.1327-1330, 21-24 April 2008.
- [77] Zaher, A.S. & McArthur, S.D.J. (2007). A Multi-Agent Fault Detection System for Wind Turbine Defect Recognition and Diagnosis. in Power Tech, 2007 IEEE Lausanne, vol., no., pp.22-27, 1-5 July 2007
- [78] Zaher, A.S., McArthur, S.D.J., Infield, D.G. & Patel, Y. (2009). Online wind turbine fault detection through automated SCADA data analysis. Wind Energy, 12, 574–593.
- [79] Zimroz, R. & Bartelmus, W. (2011). Application of Adaptive Filtering for Weak Impulsive Signal Recovery for Bearings Local Damage Detection in Complex Mining Mechanical Systems Working under Condition of Varying Load. Solid State Phenomena, vol. 180, s. 250-257.
- [80] Zimroz, R., Wodecki, J., Król, R., Andrzejewski, M., Śliwiński, P. & Stefaniak, P.K. (2014). Selfpropelled mining machine monitoring system – data validation, processing and analysis. Mine planning and equipment selection: proceedings of the 22nd MPES Conference, Dresden, Germany, 14th-19th October 2013, vol. 2, Carsten Drebenstedt, Raj Singhal (eds.). Springer, s. 1285-1294.