MONOGRAFIE, STUDIA, ROZPRAWY

M56

Grzegorz Słoń

RELACYJNE ROZMYTE MAPY KOGNITYWNE W MODELOWANIU ZŁOŻONYCH SYSTEMÓW

Kielce 2013

MONOGRAFIE, STUDIA, ROZPRAWY NR M56

Redaktor Naukowy serii
NAUKI TECHNICZNE – ELEKTRYKA
prof. dr hab. inż. Roman NADOLSKI

Recenzenci
prof. zw. dr hab. inż. Andrzej LEWIŃSKI
dr hab. inż. Kazimierz WORWA, prof. WAT

Redakcja
Aneta STARZYK

Redakcja techniczna
Aneta STARZYK

Projekt okładki
Tadeusz UBERMAN

PL ISSN 1897-2691

Wydawnictwo Politechniki Świętokrzyskiej
25-314 Kielce, al. Tysiąclecia Państwa Polskiego 7
tel./fax 41 34 24 581
e-mail: wydawca@tu.kielce.pl
www.tu.kielce.pl/organizacja/wydawnictwo

Spis treści

Ważniejsze skróty i oznaczenia

$\bar{A}$ – wartość wyostrzona zbioru rozmytego A

A_α – α-przekrój (zbiór α-poziomy) zbioru rozmytego A

$\neg A$ – dopełnienie (negacja) zbioru rozmytego A

$-A$ – przeciwieństwo liczby rozmytej A (liczba rozmyta przeciwna)

A^{-1} – odwrotność liczby rozmytej A

$|A|$ – wartość bezwzględna liczby rozmytej A

AHL – aktywny algorytm uczenia Hebba (*Active Hebbian Learning*)

BDA – zrównoważony różnicowy algorytm uczenia (*Balanced Differential Algorithm*)

$\mathbf{C}$ – wektor czynników relacyjnej rozmytej mapy kognitywnej

$\mathbf{c}$ – wektor czynników relacyjnej mapy kognitywnej

$\mathrm{core}(A)$ – jądro (*core*) zbioru rozmytego A

DDNHL – nieliniowy algorytm uczenia Hebba sterowany danymi (*Data-driven NHL*)

DHL – różnicowy algorytm uczenia Hebba (*Differential Hebbian Learning*)

$\mathbf{E}$ – macierz sąsiedztwa (*adjacency matrix*) rozmytej mapy kognitywnej

$e(C_i, C_j)$ – rozmyta przyczynowa funkcja krawędzi (*fuzzy causal edge function*) pomiędzy rozmytymi czynnikami C_i i C_j rozmytej mapy kognitywnej

FCM – rozmyta mapa kognitywna (akronim anglojęzycznej nazwy *Fuzzy Cognitive Map*)

$F(2^A)$ – rozmyty zbiór potęgowy zbioru rozmytego A

f_p – funkcja progowa ($f_p : \mathbb{R} \longrightarrow \mathbb{R}$), ograniczająca maksymalną wartość skumulowanych oddziaływań czynników

G – klasa funkcji przynależności zbioru rozmytego zdefiniowana równaniem (4.13)

$\mathrm{h}(A)$ – wysokość zbioru rozmytego A

$J(r)$ – kryterium optymalizacji mocy relacji relacyjnej mapy kognitywnej

k – liczba singletonów rozmytych zbioru rozmytego; również: liczba punktów próbkowania dyskretnego nośnika liczby rozmytej i relacji rozmytej

$l(A, B)$ – znormalizowana odległość liniowa (Hamminga) pomiędzy zbiorami rozmytymi A i B

n – liczba kluczowych czynników mapy kognitywnej

NHL – nieliniowy algorytm uczenia Hebba (*Nonlinear Hebbian Learning*)

$q(A, B)$ – znormalizowana odległość kwadratowa (euklidesowa) pomiędzy zbiorami rozmytymi A i B

$\mathbb{R}$ – zbiór liczb rzeczywistych

$\mathbf{R}$ – macierz relacji relacyjnej rozmytej mapy kognitywnej

$\mathbf{r}$ – macierz relacji relacyjnej mapy kognitywnej

$R_{i,j}$ – relacja rozmyta pomiędzy czynnikami X_i i X_j w relacyjnej rozmytej mapie kognitywnej

$r_{i,j}$ – moc relacji pomiędzy czynnikami x_i i x_j w relacyjnej mapie kognitywnej

$\bar{r}_{i,j}$ – współczynnik kierunkowy funkcji mocy relacji $r_{i,j}(u)$ pomiędzy czynnikami X_i i X_j w relacyjnej rozmytej mapie kognitywnej

RMK – relacyjna mapa kognitywna

RRMK – relacyjna rozmyta mapa kognitywna

$\mathrm{supp}(A)$ – nośnik (*support*) zbioru rozmytego A

U – przestrzeń rozważań (uniwersum, *universum*) zbioru rozmytego

SSN – sztuczna sieć neuronowa

WL – wartość lingwistyczna

$\mathbf{X}$ – wektor wartości czynników relacyjnej rozmytej mapy kognitywnej

$\mathbf{x}$ – wektor wartości czynników relacyjnej mapy kognitywnej

X_i – rozmyta wartość pojedynczego (i-tego) czynnika rozmytej relacyjnej mapy kognitywnej

$\bar{X}_i$ – wartość wyostrzona pojedynczego (i-tego) czynnika rozmytej relacyjnej mapy kognitywnej

$\hat{X}_i$ – zadana (skalarna) znormalizowana wartość pojedynczego (i-tego) czynnika rozmytej relacyjnej mapy kognitywnej

$X_{j,S}$ – skumulowane oddziaływanie pozostałych czynników na czynnik j-ty w relacyjnej mapie kognitywnej

$y_{i,j}$ – funkcja wrażliwości czynnika x_j na zmiany wartości $r_{i,j}$ w relacyjnej mapie kognitywnej

$\mathbb{Z}$ – zbiór liczb całkowitych

Z_i – zadana (wzorcowa), ostra wartość i-tego czynnika w relacyjnej rozmytej mapie kognitywnej

z_i – zadana (wzorcowa), ostra wartość i-tego czynnika w relacyjnej mapie kognitywnej

Δk – krok próbkowania nośnika liczby rozmytej

δ_i – różnica pomiędzy zadaną (z) a modelowaną (z_i) wartością i-tego czynnika w relacyjnej mapie kognitywnej

η – współczynnik korekcji mocy relacji w procesie uczenia adaptacyjnego relacyjnej mapy kognitywnej (współczynnik uczenia)

Λ – klasa funkcji przynależności zbioru rozmytego zdefiniowana równaniem (3.29)

μ_A – funkcja przynależności zbioru rozmytego A

$\frac{\mu_A(u)}{u}$ – singleton rozmyty dla elementu u

Π – klasa funkcji przynależności zbioru rozmytego zdefiniowana równaniem (3.31)

$\sum \mathrm{Count}(A)$ – liczność nierozmyta (*sigma-count*) zbioru rozmytego A

σ_i – współczynnik rozmycia wartości rozmytej (z funkcją przynależności klasy G) pojedynczego (i-tego) czynnika

$\sigma_{i,j}$ – współczynnik rozmycia relacji rozmytej (z funkcją przynależności klasy G) pomiędzy czynnikami rozmytymi X_i i X_j

φ_A – funkcja charakterystyczna zbioru konwencjonalnego A

$\oplus$ – operator dodawania rozmytego

$\ominus$ – operator odejmowania rozmytego

$\odot$ – operator mnożenia rozmytego

$\oslash$ – operator dzielenia rozmytego

$\circ$ – operator kompozycji rozmytej

1 WPROWADZENIE

Modelowanie, zwłaszcza w ujęciu matematycznym, jest ważnym aspektem projektowania i monitorowania pracy wielu systemów. Dotyczy to zarówno systemów o charakterze społecznym, biologicznym, ekonomicznym, geofizycznym, jak i technicznym. Pierwotnie modelowanie matematyczne było metodą opisu teorii tłumaczących rozmaite zjawiska, później zaczęto je stosować w procesach projektowania obiektów technicznych zamiast budowy i testowania kosztownych prototypów. Obecnie, przy zachowaniu ważności wcześniejszych zastosowań, modelowanie matematyczne jest wykorzystywane jako główne narzędzie analizy, syntezy, monitorowania i predykcji działania urządzeń i systemów.

Z punktu widzenia niniejszej pracy, pod pojęciem modelu matematycznego będzie rozumiana reprezentacja istotnych aspektów istniejącego (lub budowanego) systemu, prezentująca w użytkowej formie wiedzę o tym systemie [34], sformalizowana z wykorzystaniem metod właściwych matematyce.

Najdokładniejszą metodą tworzenia modelu matematycznego jest sporządzenie układu równań opisujących zjawiska fizyczne zachodzące w modelowanym obiekcie, a następnie znalezienie rozwiązania w oparciu o rzeczywiste (zmierzone) parametry tego obiektu. Jeśli opis został wykonany poprawnie, z uwzględnieniem wszystkich zjawisk i wszystkich parametrów, to rozwiązanie takiego układu równań powinno być dokładnym odpowiednikiem stanu rzeczywistego. Stosowanie takiej, słusznej co do założeń, metody napotyka wiele trudności przy próbach badania nieco bardziej złożonych systemów, dla których opis zachodzących zjawisk fizycznych jest utrudniony lub wręcz niemożliwy, np. wskutek niepełnej wiedzy na temat ich przebiegu (dotyczy to m.in. zagadnień medycznych, geograficznych, logistycznych, społecznych, militarnych i, w dużej mierze, ekonomicznych). Ponadto, nawet w sytuacji dobrej znajomości charakteru zjawisk, stosowanie modelu w postaci układu równań matematycznych ujmujących jego parametry, nie zawsze jest możliwe, co może wynikać ze złożoności tych równań (i tym samym trudności w ich rozwiązaniu) bądź niemożności pomiaru parametrów kluczowych dla uzyskania rozwiązania. Aspekt ten występuje m.in. w zagadnieniach geograficznych, ekonomicznych oraz, po części, logistycznych i technicznych, ze szczególnym uwzględnieniem diagnostyki i monitorowania. Dochodzą do tego dodatkowe problemy związane z uwzględnieniem złożoności systemów, w których występuje wiele powiązanych i wzajemnie na siebie wpływających czynników, co skutkuje trudnościami z formułowaniem odpowiednich równań opisujących przebiegi zmienności poszczególnych wielkości.

Niezależnie od wyżej wymienionych zagadnień wynikających wprost ze struktury obiektów, pojawiają się dodatkowe problemy związane z pomiarem kluczowych parametrów, a właściwie z niepewnością co do ich faktycznych wartości, które, choć są ujmowane ilościowo, to jednak podlegają subiektywnej ocenie ob-

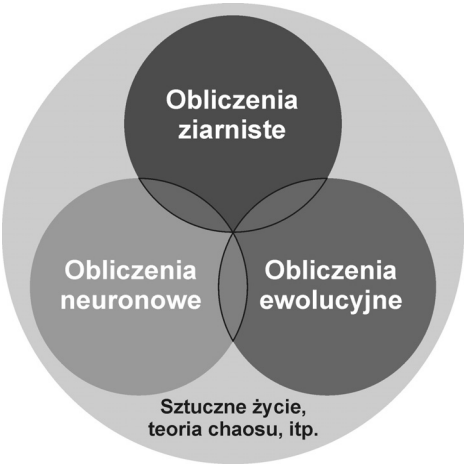
serwatora. Może to dotyczyć wielkości zarówno mierzonych, jak i obserwowanych (np. określenie „jest zimno", nawet poparte pomiarem temperatury, wynika z subiektywnego odczucia, a przecież może zależeć od niego kosztowna decyzja uruchomienia instalacji grzewczej).

W obliczu powyższych problemów od dawna podejmowano prace nad metodami modelowania, które będą mniej zależne od stopnia złożoności, kompletności wiedzy oraz pewności i precyzji informacji na temat struktury modelowanego obiektu, a jednocześnie pozwolą zachować wystarczającą dokładność. Szczególna rola przypada tu badaniom nad sposobami wnioskowania istot inteligentnych. Umysł ludzki nie tworzy układów równań różniczkowych, a jednak jest w stanie planować i przewidywać rozwój, niekiedy bardzo złożonych, procesów, wykorzystując do tego doświadczenie i bieżące obserwacje. Implementacja podobnego sposobu działania w maszynach przeniosłaby rozwój cywilizacyjny na zupełnie inny poziom. Efektem badań w tej dziedzinie są metody i modele tzw. **sztucznej inteligencji** (ang. ***Artificial Intelligence***), której celem jest odwzorowanie ludzkiej inteligencji w maszynach tak, aby mogły „działać" i „myśleć" jak ludzie. Celem tych badań było (i jest nadal) opracowywanie metod modelowania nieprecyzyjnej rzeczywistości w różnych jej aspektach bez odwoływania się do złożonego aparatu matematycznego.

Samo pojęcie **Sztucznej Inteligencji (SI)** wprowadzono w 1956 r. i dotyczy ono, w ogólnym zarysie, tworzenia komputerowych algorytmów naśladujących wnioskowanie zachodzące w ludzkim mózgu. SI jest rozległą dziedziną wiedzy, obejmującą wnioskowanie, uczenie maszynowe, planowanie, inteligentne wyszukiwanie oraz budowanie percepcji. Rozległości tematycznej towarzyszy wielość definicji, tworzonych w odniesieniu do poszczególnych zagadnień rozważanych w ramach SI. Wydaje się, że istotę SI najlepiej oddaje określenie sformułowane przez E. Feigenbauma [160]: „Sztuczna inteligencja stanowi dziedzinę informatyki dotyczącą metod i technik wnioskowania symbolicznego przez komputer oraz symbolicznej reprezentacji wiedzy stosowanej podczas takiego wnioskowania". W tradycyjnym podejściu wnioskowanie w systemach SI polega na dążeniu do osiągnięcia założonego celu z użyciem zbioru dostępnych faktów oraz predefiniowanej bazy wiedzy, która jest reprezentowana przez zbiór reguł typu IF-THEN. Większość klasycznych problemów rozpatrywanych z użyciem SI formułuje się jako problemy przestrzeni stanów [157], gdzie określenie „stan" oznacza pewien szczególny przypadek rozwiązania problemu, a „przestrzeń" jest zbiorem stanów. Jeden lub kilka początkowych stanów dobiera się arbitralnie „zgadując" rozwiązanie, następnie, jeżeli wstępny wynik nie odpowiada założonemu celowi, wykonuje się „ruchy" w przestrzeni stanów, oddziałując na istniejące stany przy użyciu pewnych operatorów. Tak otrzymane nowe stany są porównywane z założonym celem i jeśli nie zostanie on wykryty, to ruchy są kontynuowane. Proces ten trwa dopóty, dopóki cel nie zostanie wykryty wśród nowych stanów lub dopóki nie zostanie przerwany przez użytkownika. W takim ujęciu modelowanie rzeczywistości polega, ogólnie mówiąc, na budowaniu bazy pewnych reguł, a następnie przeszukiwaniu tej bazy, co rzeczywiście do pewnego stopnia odpowiada ludzkiemu sposobowi

rozumowania w niektórych okolicznościach. Dość wcześnie zorientowano się jednak, że taka metoda, chociaż pozwala na rozwiązywanie złożonych problemów bez użycia zaawansowanego aparatu wyższej matematyki, ma niedostatki wynikające z istoty symbolicznej reprezentacji wiedzy. Przejawiają się one m.in. rosnącą złożonością struktury budowanych reguł, niemożnością nadzorowanej adaptacji struktury modelu oraz brakiem zobiektywizowanych metod optymalizacji używanych podczas syntezy modeli. Braki te należało uzupełnić i uczyniono to rozszerzając klasyczne podejście do sztucznej inteligencji o zestaw metod i modeli o wspólnej nazwie **Inteligencji Obliczeniowej (IO).**

Pojęcie **inteligencji obliczeniowej** (ang. ***Computational Intelligence***) zostało wprowadzone we wczesnych latach dziewięćdziesiątych dwudziestego wieku w pewnej opozycji do klasycznie rozumianej sztucznej inteligencji. Jedna z pierwszych definicji [14] określała mianem obliczeniowo inteligentnego system, który: operuje wyłącznie danymi liczbowymi (niskiego poziomu), posiada komponent rozpoznawania wzorców oraz nie używa wiedzy w rozumieniu SI. W miarę powstawania kolejnych metod obliczeniowych, zwłaszcza w zakresie populacyjnych technik adaptacji parametrów systemowych, rozszerzano też pojęcie IO. Obecnie dotyczy ono raczej pewnej kombinacji tzw. miękkich obliczeń (ang. *Soft Computing*), rozumianych jako zespół technik obliczeniowych wykorzystujących teorię zbiorów rozmytych, inspiracje biologiczne (algorytmy ewolucyjne, rojowe, sieci neuronowe) i wnioskowanie probabilistyczne oraz przetwarzania numerycznego łączonych tak, aby uzyskać efekt synergii. Nowoczesne podejście do inteligencji obliczeniowej określa ją jako rodzinę biologicznie inspirowanych modeli inteligencji maszynowej [99], ze szczególnym uwzględnieniem modeli obliczeń ziarnistych, obliczeń neuronowych i obliczeń genetycznych w powiązaniu z ich wzajemnym oddziaływaniem z modelami sztucznego życia, teorią chaosu i innymi teoriami nieprecyzyjnego opisu rzeczywistości. Ogólny zarys tak rozumianej rodziny inteligencji obliczeniowej przedstawia rysunek 1.1.



Rys. 1.1. *Ogólny zarys nowoczesnej definicji rodziny inteligencji obliczeniowej (na podst. [99])*

Przedstawiony na rysunku 1.1 podział na główne podgrupy modeli IO jest umowny, można go jednak uzasadnić powiązaniem z kluczowymi problemami, z którymi klasyczne modele SI sobie nie radzą:

1. Rosnąca złożoność bazy reguł.

W tradycyjnym podejściu do SI stany są reprezentowane przez symbole, a zbiory reguł opisują przejścia pomiędzy stanami. Często osiągnięcie celu wymaga operowania zbiorem bardzo wielu reguł, co jednakże wydłuża czas przeszukiwania i w konsekwencji obniża efektywność systemu wnioskowania. Metodą na neutralizację tego problemu jest stworzenie bazy wiedzy z mniejszą liczbą reguł, ale pozwalającą tylko na częściowe dopasowanie stanów. Takie częściowe dopasowanie jest możliwe z użyciem obliczeń ziarnistych, ze szczególnym uwzględnieniem logiki zbiorów rozmytych [98, 222, 223].

2. Trudności ze stosowaniem nadzorowanej adaptacji parametrów modelu.

W klasycznym podejściu do SI można stosować uczenie przez indukcję [155] i przez analogię [197], jednakże nie jest możliwe uczenie nadzorowane [199], w którym trener dostarcza pewną liczbę danych uczących (wejściowych i wyjściowych), a uczący się system adaptuje swoje wewnętrzne parametry tak, aby na zadane sygnały wejściowe odpowiedzieć odpowiednimi sygnałami wyjściowymi. Takie możliwości mają natomiast modele oparte na obliczeniach neuronowych (ze szczególnym uwzględnieniem sztucznych sieci neuronowych) [53, 160, 161], co jest o tyle istotne, że w większości zastosowań uczenia maszynowego wymagane jest użycie procedur uczenia nadzorowanego.

3. Trudności z optymalizacją parametrów podczas projektowania nowych systemów.

Dla klasycznej SI trudnym obszarem jest optymalizacja parametrów systemowych w takich zagadnieniach, jak projektowanie, synteza, diagnostyka czy planowanie, które są podstawowymi problemami rzeczywistych systemów decyzyjnych [42, 198]. Wśród metod przeszukiwania przestrzeni stanów jedynie algorytm heurystyczny zawiera elementy przydatne w tego typu zadaniach, jednakże heurystyczne eksperymentowanie nie zawsze jest możliwe (z uwagi choćby na niedostatek wiedzy ekspertowej [44] o rozważanym problemie). Rozwiązania mogą być tutaj osiągane dzięki zastosowaniu obliczeń ewolucyjnych, a w szczególności algorytmów genetycznych [129], których główne cechy umożliwiają efektywne poszukiwanie globalnego optimum rozwiązania.

Wymienione powyżej podejścia angażujące obliczenia ziarniste (idea wprowadzona w 1963 r. [15]), szczególnie w tym ich aspekcie, który wykorzystuje logikę rozmytą (wprowadzoną w 1965 r. [221, 222]), sieci neuronowe (rozwijane od 1943 r. [127, 192]), algorytmy genetyczne (wprowadzone w 1975 r. [59]), sztuczne systemy immunologiczne (zarys opracowany w 1974 r. [84]), ale także, np. programowanie ewolucyjne (rozwijane od 1966 r. [36]) należą do kanonu inteligencji

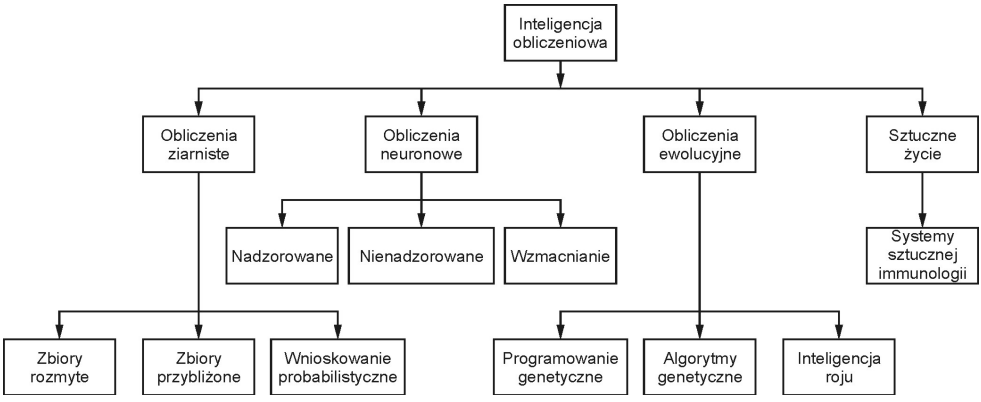
obliczeniowej. W takim ujęciu termin Inteligencja Obliczeniowa może być rozumiany jako rozwiązywanie różnych problemów SI z użyciem komputerów wykorzystujących obliczenia numeryczne [49, 159, 160], z zastosowaniem takich technik, jak: sieci neuronowe, logika rozmyta czy algorytmy ewolucyjne, jak również, niewymienione tu bezpośrednio, zbiory przybliżone, zmienne niepewne i metody probabilistyczne.

Jako stosunkowo nowa dziedzina, IO doczekała się różnych, mniej lub bardziej formalnych, opisów. Można ją scharakteryzować zbiorczą definicją, ujmującą jej główne cechy [99].

DEFINICJA 1.1

Inteligencja Obliczeniowa to obliczeniowe modele i narzędzia inteligencji zdolne do: bezpośredniego wprowadzania surowych danych liczbowych i sensorycznych, przetwarzania ich z wykorzystaniem reprezentatywnego paralelizmu i potokowości problemu, generowania w odpowiednim czasie wiarygodnych odpowiedzi, a ponadto wykazujące wysoką odporność na błędy.

Ogólny zarys rodziny metod IO można rozwinąć, uzupełniając każdy z głównych elementów o podejścia pochodne. Pewien rodzaj takiego rozwinięcia, w postaci drzewa genealogicznego członków rodziny IO, przedstawiono na rysunku 1.2.

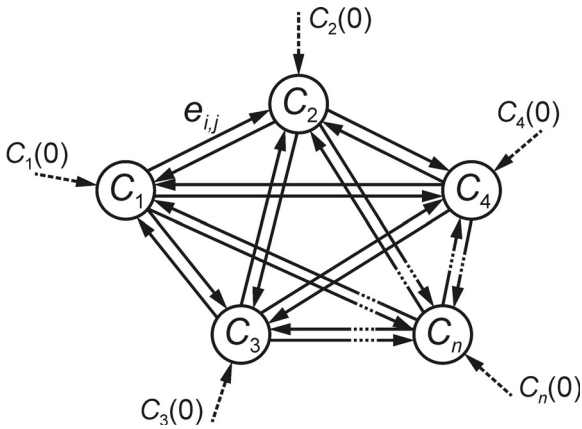


Rys. 1.2. *Fragment drzewa genealogicznego inteligencji obliczeniowej (na podst. [99])*

Przedstawiony na rysunku 1.2 podział ma charakter ogólny i nie wyczerpuje wszystkich opcji, tym niemniej obejmuje on główne stosowane obecnie typy modeli IO. Należy też podkreślić, że poszczególne typy mogą być zestawiane w swego rodzaju hybrydy, łączące w sobie charakterystyczne cechy składników – dotyczy to np. łączenia w jednym modelu cech obliczeń ziarnistych (wyrażonych np. przez stosowanie zbiorów rozmytych), obliczeń neuronowych (poprzez grafową strukturę formułowania problemu) oraz obliczeń ewolucyjnych (zaimplementowanych w algorytmach optymalizacji), i są to modele najczęściej stosowane w prak-

tyce. Warto przy tym nadmienić, że hybrydowe modele łączące w sobie cechy SI oraz IO przeważnie mają charakter synergiczny [49, 159], co stanowi dodatkową zaletę tego rodzaju podejść.

Zasadniczo, wszelkie podejścia do inteligentnego modelowania, wykorzystujące metody SI i IO, opierają się założeniu, że dla prawidłowego odwzorowania pracy danego układu nie jest konieczna dokładna znajomość jego struktury i zachodzących w jego wnętrzu zjawisk. Zamiast tego wystarczy dysponować odpowiednio obszerną bazą danych pomiarowych (obserwacyjnych) oraz względnie prostą, szkieletową, strukturą zawierającą kluczowe dla celów modelowania czynniki połączone odpowiednimi powiązaniami przyczynowymi. Struktura taka może być wyrażana poprzez zestaw reguł przyczynowo-skutkowych (typu IF-THEN) lub w bardziej przystępnej formie wizualnej, w postaci grafu skierowanego, takiego jaki został przedstawiony na rysunku 1.3.

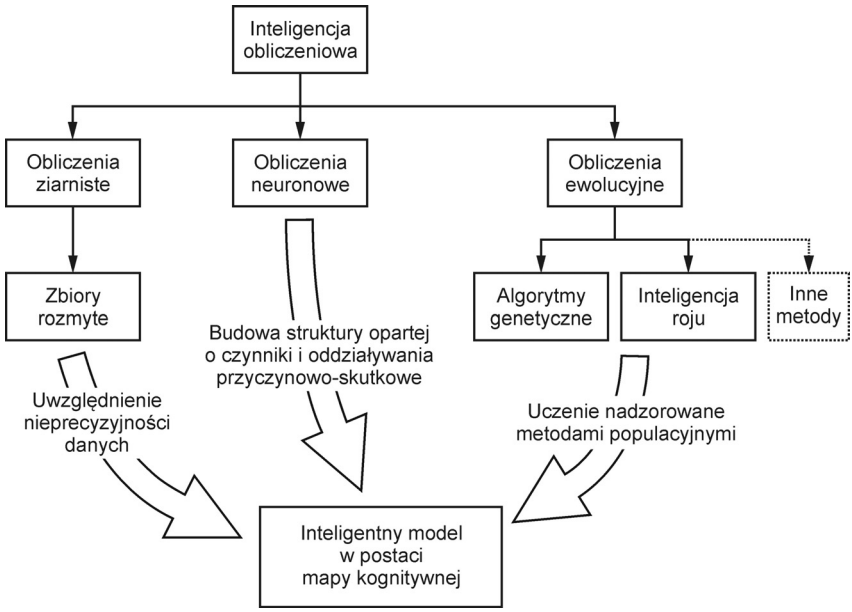


Rys. 1.3. *Ogólna postać modelu tworzonego przy użyciu inteligencji obliczeniowej.* C_1, …, C_n *– kluczowe czynniki,* $C_i(0)$ *– wartość początkowa (pobudzająca)* i*-tego czynnika;* $\{e_{i,j}\}$ *– zbiór powiązań przyczynowo-skutkowych (wzajemnych wpływów) pomiędzy czynnikami* $(i, j = 1, \dots, n)$*;* n *– liczba kluczowych czynników*

Zależnie od przyjętej metody, ogólny model przedstawiony na rysunku 1.3 może przybierać różne formy szczegółowe (przykładowo dla sztucznej sieci neuronowej czynniki, zwane tam neuronami i posiadające swoją architekturę modelową, będą przydzielane do poszczególnych warstw, a powiązania będą miały charakter np. jednokierunkowy – od warstwy wejściowej do wyjściowej; w mapie kognitywnej struktura modelu będzie miała charakter wielokierunkowy, bardziej zbliżony do rys. 1.3), tym niemniej ogólny charakter modelu pozostanie ten sam. Określenie „inteligentny” zarezerwowane jest do modeli, w których zastosowano techniki uczące, w wyniku których struktura wewnętrzna modelu jest modyfikowana tak, aby jak najlepiej odwzorowywała zachowanie rzeczywistego obiektu. Techniki te służą adaptacji powiązań pomiędzy czynnikami modelu, przy czym powiązania te mogą mieć różny, nie tylko liczbowy, charakter (będzie to dokładniej omówione

w dalszej części monografii). Każda z technik inteligencji obliczeniowej ma swoją specyfikę, predestynującą ją do określonych zadań. Sztuczna sieć neuronowa dobrze sprawdzi się w zadaniach klasyfikacji, algorytmy ewolucyjne będą wybierane do rozwiązywania problemu poszukiwania globalnego optimum, sztuczny system immunologiczny znajdzie zastosowanie w rozwiązywaniu zadań związanych z działaniem systemów detekcyjnych, a programowanie ewolucyjne może być najlepszą metodą prognozowania zmian w otoczeniu.

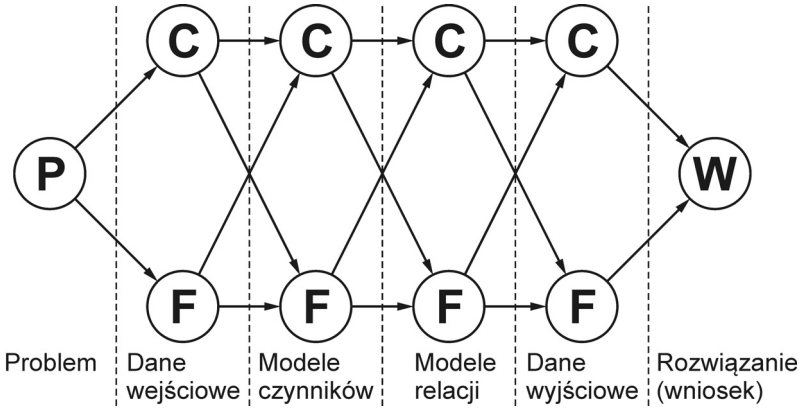
W praktycznych zastosowaniach szczególne znaczenie mają metody hybrydowe, synergicznie łączące w sobie główne zalety składających się na nie elementów. Wyróżnia się tutaj specyficzne połączenie obliczeń ziarnistych, neuronowych i ewolucyjnych, dzięki któremu możliwe jest tworzenie najbardziej uniwersalnych modeli wykorzystujących inteligencję obliczeniową, czyli tzw. map kognitywnych, których podstawy są znane od 1948 r. [194], a początki ich nowoczesnej koncepcji datuje się na rok 1986 [108]. Mapy kognitywne mogą być wykorzystywane zarówno w zadaniach klasyfikacji, jak i tradycyjnie rozumianego modelowania, ponieważ łączą w sobie cechy wielu wcześniej stosowanych podejść do modelowania inteligentnego. Graficzna reprezentacja ogólnego modelu mapy kognitywnej jest właściwie dokładnym odpowiednikiem postaci przedstawionej na rysunku 1.3, przy czym kluczowe dla budowy i funkcjonowania modelu mechanizmy wykorzystują poszczególne, wybrane metody inteligencji obliczeniowej. Główny zarys tych zależności przedstawia rysunek 1.4, na którym uwypuklono te metody IO, które odgrywają największą rolę w tym procesie.



Rys. 1.4. *Ogólny zarys sposobu wykorzystania metod IO podczas budowy mapy kognitywnej*

Przedstawione powyżej podejścia i określenia nie obejmują, oczywiście, pełnego zakresu ani możliwości zastosowań IO, tym bardziej, że w praktyce można napotkać wiele zagadnień, w których IO jest stosowana wybiórczo, do rozwiązywania problemów cząstkowych, lub też stosowane rozwiązania zawierają elementy podobieństwa do klasycznych metod IO. Dotyczy to zarówno badania systemów, do których metody IO są stosowane częściej (np. podejmowanie decyzji w oparciu o zasoby baz wiedzy [204–206]), jak i rzadziej (jak np. systemy logistyczne i transportowe [114–117]). Ta płynność klasyfikacji wynika z mnogości zastosowań modelowania, która wymusza dostosowywanie metod do realnych problemów. Niniejsza praca z konieczności ogranicza się do fragmentu tego obszaru, jakim jest modelowanie z użyciem różnego rodzaju map kognitywnych, ze szczególnym uwzględnieniem relacyjnej rozmytej mapy kognitywnej.

Uniwersalność modelu mapy kognitywnej przejawia się nie tylko w wielokierunkowej strukturze systemu przepływu sygnałów, ale również w elastycznym podejściu do charakteru przetwarzanych danych. Można w nim operować zarówno danymi precyzyjnymi, jak i nieprecyzyjnymi o różnych stopniach pewności co do ich wartości. W sytuacji wysokiej pewności i precyzji posiadanych danych wystarczające jest podejście, w którym dane te (zarówno wartości czynników, jak i powiązań pomiędzy nimi) mają liczbowe reprezentacje z dziedziny liczb rzeczywistych, co eliminuje potrzebę wprowadzania technik rozmywania. W miarę wzrostu poziomu nieprecyzyjności można wprowadzać metody reprezentacji danych z użyciem zbiorów rozmytych, ze szczególnym uwzględnieniem liczb rozmytych, aż do modeli, w których wszystkie dane mają charakter rozmyty. Pomiędzy skrajnymi (w pełni liczbowym „ostrym" oraz w pełni rozmytym) podejściami można znaleźć szereg form pośrednich, które mogą być komponowane na różne sposoby, zależnie od aktualnych potrzeb. Ilustruje to rysunek 1.5 przedstawiający możliwości sposobu organizacji przepływu danych przez model wnioskujący.



Rys. 1.5. *Schemat blokowy modelu struktur powiązań i obliczeń modeli map kognitywnych, gdzie: P – sformułowanie problemu, C – podejście klasyczne (ostre), F – podejście rozmyte, W – uzyskanie rozwiązania (wniosek końcowy)*

Na rysunku 1.5 pokazano, w ogólny sposób, możliwości organizacji pracy modelu mapy kognitywnej, której kolejne etapy mogą mieć charakter ostry lub rozmyty – zależnie od dostępnych danych oraz założonych celów. Dla lepszego zobrazowania tych możliwości poniżej przedstawiono trzy przykładowe zastosowania.

Przykład 1.1. *System diagnostyczny układu technicznego (model ostry)*

W modelu, zwłaszcza zautomatyzowanym, służącym do diagnozowania stanu obiektu technicznego (np. wybranych elementów elektrycznego wyposażenia pojazdu samochodowego) dane wejściowe są zwykle pobierane z odpowiednich czujników pomiarowych, mają więc wymiar liczbowy. Na ogół też (dotyczy to bardziej złożonych obiektów) trudno byłoby zmierzyć wszystkie fizyczne parametry niezbędne do stworzenia dokładnego modelu matematycznego, w związku z czym uzasadnione jest stosowanie innych podejść, np. mapy kognitywnej. Czynnikami takiej mapy mogą więc być wybrane (niektóre) parametry fizyczne obiektu oraz inne wielkości przydatne podczas wnioskowania (jak np. ujęty procentowo poziom poprawności pracy poszczególnych podzespołów czy czas pozostały do kolejnego przeglądu). Wartości tych czynników mogą mieć charakter ostry, ponieważ przemawia za tym ostry typ uzyskanych danych pomiarowych. Powiązania pomiędzy takimi czynnikami również mogą mieć wymiar ostry (czysto liczbowy), co wynika z ostrego charakteru danych odniesienia, których można użyć podczas uczenia modelu, a które również pochodzą z pomiarów. Także dane wyjściowe (są nimi tutaj wartości wybranych czynników, będące podstawą wnioskowania) będą miały charakter ostry, podobnie jak wszystkie pozostałe czynniki. Tak więc cały proces wnioskowania w takim modelu będzie miał charakter ostry.

Przykład 1.2. *System logistyczny przedsiębiorstwa (model mieszany)*

Działanie systemu logistycznego, rozumianego jako zespół powiązanych ze sobą węzłów decyzyjnych i wykonawczych, zależy od wielu, wzajemnie na siebie wpływających parametrów o charakterze zarówno ilościowym, jak i jakościowym. Efektywne zarządzanie takim systemem, sprowadzające się w zasadzie do zapewnienia równomiernej dystrybucji zasobów, musi uwzględniać elementy predykcji, a więc opierać się de facto na pewnym modelu matematycznym, który będzie w stanie z wystarczającą dokładnością odwzorować działanie całego systemu. Dane wejściowe mogą mieć charakter liczbowy, jednakże lepiej jest przedstawiać je w postaci lingwistycznej (np. taki parametr jak „średni czas realizacji dostawy", podany w liczbach bezwzględnych może być mylący, jako że „dostawa" może być realizowana na różnych dystansach, w związku z czym lepiej stosować określenia typu „krótki", „średni" itp.), tym bardziej, że część z nich wprost będzie miała taki charakter (np. „kompetencje kierownictwa"). Tak więc dane wejściowe, a co za tym idzie również wartości czynników, oraz dane wyjściowe mają charakter rozmyty. Z drugiej strony, z uwagi na możliwy mieszany rodzaj danych wejściowych (mogą one przecież zawierać również elementy ściśle liczbowe), możliwe i dogod-

ne jest projektowanie relacji pomiędzy czynnikami w formie liczb (zarówno dodatnich, jak i ujemnych) z pewnego zakresu, co jest możliwe pod warunkiem zastosowania dodatkowych mechanizmów łączenia liczbowych relacji i rozmytych czynników. Taki model ma charakter mieszany.

Przykład 1.3. ***System powiązań społeczno-ekonomicznych w państwie (model rozmyty)***

W systemach tego rodzaju większość dostępnych danych ma charakter względny i opisowy (np. „skala ubóstwa", „poziom edukacji", „stopień akceptacji społecznej" itp.), podobnie jak określanie wzajemnych powiązań pomiędzy takimi czynnikami (wzrost wartości danego czynnika może, np. powodować spadek wartości innego, przy tym spadek ten może być np. „znaczny", „niewielki" itp.). Modele tego rodzaju, z rozmytymi czynnikami i relacjami, są modelami w pełni rozmytymi.

Powyższe przykłady mają za zadanie tylko ogólnie zilustrować pewne zjawiska. Należy przy tym pamiętać, że dany rodzaj systemu nigdy nie ma jednoznacznie zdeterminowanej formy opisu. System techniczny można opisać całkowicie rozmytym modelem mapy kognitywnej, a system społeczny – modelem w pełni ostrym. Zależy to przede wszystkim od charakteru dostępnych danych oraz od ekspertowej wiedzy na temat powiązań pomiędzy czynnikami.

Stosowane obecnie rozwiązania konstrukcyjne modeli map kognitywnych najczęściej odnoszą się do struktur mieszanych, w których dane wejściowe mają charakter rozmyty, jednakże wewnętrzne mechanizmy adaptacji parametrów operują na ich liczbowych odpowiednikach. Modele takie noszą nazwę **rozmytych map kognitywnych** (ang. ***FCMs – Fuzzy Cognitive Maps***) i stosuje się je w analizie zagadnień społecznych, politycznych, ekonomicznych, medycznych, biologicznych, militarnych i, do pewnego stopnia, technicznych. Bliższy opis takiego podejścia, zwłaszcza w aspekcie algorytmów uczenia nienadzorowanego i nadzorowanego, został przedstawiony w podrozdziale 3.2. Z uwagi na często stosowany w literaturze anglojęzyczny akronim (FCM), w dalszej części monografii będzie on stosowany zamiennie z nazwą „rozmyta mapa kognitywna". Charakterystyczną cechą modelowania z użyciem FCM jest zaangażowanie ekspertów do wyłonienia kluczowych czynników analizowanego systemu oraz określenia powiązań pomiędzy tymi czynnikami. Przy tym eksperci determinują zarówno lingwistyczne wartości rozmyte, które mogą przyjmować czynniki, jak też (również przy użyciu wartości lingwistycznych z odpowiedniego zbioru) charakter wzajemnych oddziaływań pomiędzy tymi czynnikami. Użycie na tym etapie rozmytego (czyli wykorzystującego wartości lingwistyczne) opisu rzeczywistości jest korzystne przy modelowaniu złożonych systemów, których parametry podlegają subiektywnemu oszacowaniu obserwatora. Niestety, automatyzacja procesów adaptacji tak zbudowanych modeli jest bardzo trudna, stąd też w większości zastosowań wprowadza się swoiste uproszczenie, polegające na zastąpieniu rozmytych wartości czynników i powiązań przyczynowych pomiędzy czynnikami ich reprezentacjami liczbowymi,

które mogą mieć ciągły lub dyskretny charakter. Dalsze działania adaptacyjne (uczące), kluczowe dla prawidłowego działania modelu, prowadzi się w oparciu o te liczbowe odpowiedniki. Oznacza to próbę stosowania liczbowych (ostrych) metod uczenia w nieliczbowym (rozmytym) środowisku, a to z kolei sprawia, że rozmycie modelu, będące jego głównym atutem, jest tylko deklaracyjne, co de facto niweluje pożytek płynący z rozmywania danych. W tej sytuacji korzystniejsze wydaje się albo bezpośrednie stosowanie opisu w pełni liczbowego, który można nazwać „ostrym" (jak w przykładzie 1.1), bez odwoływania się do teorii zbiorów rozmytych, co pozwala na stosowanie szerszej gamy metod uczenia modelu (dzięki możliwości wykorzystania innych niż populacyjne metod uczenia nadzorowanego), albo, tam, gdzie jest to uzasadnione, posługiwanie się podejściem w pełni rozmytym (jak w przykładzie 1.3), co z kolei pozwoli zachować korzyści płynące z lingwistycznego opisu rzeczywistości. O ile metody zbliżone do ostrej mapy kognitywnej były już stosowane (pewne analogie, chociaż w prostszej formie, występują np. przy konstruowaniu sztucznych sieci neuronowych), to modelowanie w pełni rozmytych map kognitywnych nie występuje w literaturze, co może być spowodowane brakiem odpowiednich metod projektowania rozmytych powiązań pomiędzy rozmytymi czynnikami, brakiem odpowiedniego aparatu matematyczno-algorytmicznego do analizy pracy takich modeli oraz trudnościami w algorytmicznej implementacji tego typu rozwiązań.

Celem badań, których wyniki opisano w niniejszej monografii było opracowanie metody inteligentnego modelowania systemów charakteryzujących się zarówno niepewnością, jak i nieprecyzyjnością informacji. Jako podstawę przyjęto założenie, że modelowanie takie będzie się opierać na rozmytym opisie własności systemu oraz, że taki rozmyty opis będzie wykorzystywany na wszystkich etapach działania, tzn. podczas definiowania kluczowych czynników i ich powiązań, uczenia oraz eksploatacji modelu. Przyjęto ponadto, że tak rozumiane modelowanie będzie realizowane w oparciu o w pełni rozmytą mapę kognitywną.

Monografia przedstawia wyniki badań nad projektowaniem różnych modeli map kognitywnych, ze szczególnym uwzględnieniem modeli w pełni ostrych (ścieżka PCCCCW – rys. 1.5) i w pełni rozmytych (ścieżka PFFFFW – rys. 1.5). Na potrzeby projektowania, syntezy i analizy zarówno w pełni ostrych, jak i w pełni rozmytych map kognitywnych przyjęto, że graficzna reprezentacja takich modeli stanowi odpowiednik grafu – rysunek 1.3, z tym, że w miejsce oddziaływań przyczynowo-skutkowych pomiędzy czynnikami wprowadzono relacje dwuargumentowe, odpowiednio: ostre lub rozmyte (zależnie od charakteru modelu), stąd też prezentowane modele map kognitywnych noszą dodatkową nazwę „relacyjnych" (bliższe wyjaśnienie na temat struktury relacyjnych map kognitywnych zostało zawarte w rozdziale 2, a relacyjnych rozmytych map kognitywnych – w rozdziale 4).

W kolejnych rozdziałach opisano możliwości, atuty i zagrożenia związane z wykorzystaniem relacyjnych map kognitywnych, w ujęciach: ostrym i rozmytym, do modelowania złożonych systemów. Pod pojęciem złożonego systemu będzie rozumiany system o właściwie dowolnej strukturze, której szczegóły (charakter zachodzących wewnątrz systemu zjawisk) nie muszą być dokładnie znane, można nato-

miast wyróżnić w nim pewne czynniki o wartościach kluczowych z punktu widzenia celów modelowania. Scharakteryzowano, najczęściej obecnie stosowaną, metodę wykorzystania map kognitywnych w postaci FCM, jak również, w związku ze wspomnianymi powyżej wadami takiej metody, zaproponowano dwa inne, skrajne podejścia do budowy map kognitywnych. Jedno z nich polega na zastosowaniu modelu całkowicie ostrego, o nazwie **Relacyjna Mapa Kognitywna (RMK)**, w którym zarówno wartości czynników, jak i powiązań przyczynowo-skutkowych (czyli relacji) pomiędzy czynnikami są reprezentowane przez odpowiednie miary liczbowe. Drugie – bardziej złożone – dotyczy całkowicie nowego podejścia do procesu budowy i uczenia rozmytych map kognitywnych. Polega ono na rezygnacji z zastępowania rozmytych wartości czynników i ich powiązań liczbowymi odpowiednikami na rzecz zachowania w pełni rozmytych form w postaci odpowiednio: liczb rozmytych i relacji rozmytych. Model tego rodzaju nazwano **Relacyjną Rozmytą Mapą Kognitywną (RRMK)**. Jego stosowanie jest trudne algorytmicznie, ale pozwala na niemal pełną automatyzację procesu budowy, uczenia i testowania modelu RRMK z jednoczesnym zachowaniem jej rozmytej struktury.

Tworząc nową koncepcję oraz metodę budowy modelu RRMK, autor wprowadził szereg oryginalnych rozwiązań, wśród których do najważniejszych można zaliczyć:

1. Opracowanie nowego sposobu opisu rozmytej wartości czynnika.
2. Określenie sposobu odwzorowywania zależności przyczynowo-skutkowych pomiędzy czynnikami.
3. Określenie charakteru relacji rozmytych.
4. Opracowanie nowego sposobu normalizacji danych.
5. Nowe zdefiniowanie pojęcia nośnika liczby rozmytej.
6. Opracowanie metody uczenia modelu adekwatnej do jego charakteru.

W obydwu opisanych wyżej podejściach, dzięki wysokiej autonomii algorytmów uczenia nadzorowanego, można minimalizować skutki ewentualnych błędów wprowadzanych przez ekspertów projektujących początkową strukturę modelu, co jest istotne szczególnie w dziedzinach, w których takich ekspertów brak lub jest ich niewielu. Zasady budowy RMK przedstawiono w rozdziale 2, natomiast główną część pracy dotyczącą tworzenia rozmytych map kognitywnych, ze szczególnym uwzględnieniem RRMK, zawierają rozdziały 3 i 4 (w rozdziale 3 zamieszczono także charakterystykę obecnie stosowanego „klasycznego" podejścia z użyciem FCM). W ramach tego rozdziału umieszczono również charakterystykę obecnie stosowanego „klasycznego" podejścia z użyciem FCM.

Układ pracy zaplanowano tak, aby czytelnik mógł stopniowo przejść od całkowicie ostrych postaci modelowania z użyciem relacyjnych map kognitywnych, poprzez mieszane formy rozmytych map kognitywnych, których metodologia stanowi pewną kompozycję podejść ostrego i rozmytego, aż po w pełni rozmyty model relacyjnej rozmytej mapy kognitywnej.

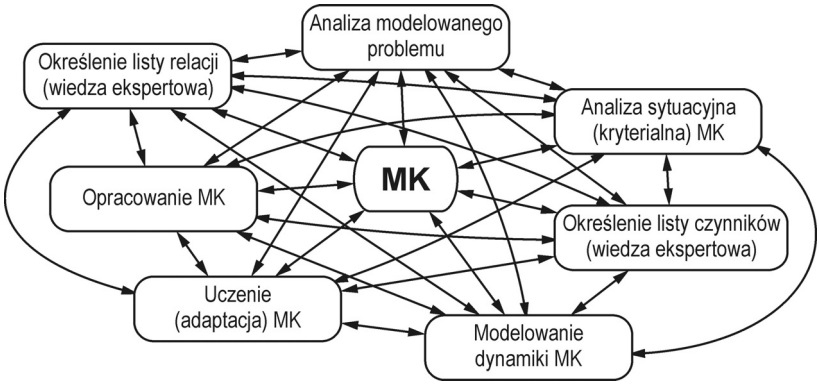
2 RELACYJNE MAPY KOGNITYWNE

Pojęcie mapy kognitywnej, jako pewnego systemu wnioskującego, wprowadzono już w 1948 r. [194], jednakże szersze prace badawcze pod kątem zastosowania tej koncepcji w układach sztucznej inteligencji rozpoczęto w roku 1986 [108] – od wprowadzenia do terminologii naukowej pojęcia rozmytej mapy kognitywnej. Powiększanie mocy obliczeniowych komputerów powodowało stopniowe zwiększanie zainteresowania wykorzystaniem map kognitywnych w coraz to nowych dziedzinach. Ich uniwersalny charakter powoduje, że mogą być wykorzystywane w systemach diagnostycznych [147, 172], w robotyce [90], w systemach kontroli procesów [142, 188, 189], w predykcji szeregów czasowych [57] czy w modelowaniu wirtualnym [1, 2, 30]. Podejmuje się też udane próby implementacji map kognitywnych w inteligentnych systemach wspomagania decyzji (jak np. [91, 163, 226]) oraz identyfikacji zagrożeń [167]. Po odpowiedniej modyfikacji, mapa kognitywna może służyć również do modelowania przebiegów czasowych wybranych parametrów układów technicznych [64, 174, 186].

Modele tworzone w oparciu o mapy kognitywne charakteryzują się następującymi cechami:

- wygodne przedstawienie analizowanego problemu,
- duża elastyczność w doborze czynników i relacji,
- prosta, graficzna interpretacja charakteru modelu.

Mapy kognitywne mogą być z powodzeniem stosowane przy syntezie i analizie modeli obiektów w warunkach pewnej nieprecyzyjności (np. niepełnej, niepewnej lub trudnej do standardowego analizowania wiedzy) dotyczącej samego systemu, jego otoczenia lub metody decyzyjnej (m.in. [108, 141, 151, 164, 166, 169, 176, 218]). Główne elementy takiego modelowania schematycznie przedstawiono na rysunku 2.1.



Rys. 2.1. *Ogólny schemat modelowania opartego na mapach kognitywnych. MK – mapa kognitywna*

Rysunek 2.1 przedstawia pewną ideę działań podejmowanych podczas budowy konkretnej mapy kognitywnej. Proces ten sam w sobie stanowi również swoistą mapę kognitywną.

Przedstawione w niniejszym rozdziale podejście dotyczy ogólnego zarysu modelowania przy użyciu tzw. relacyjnych map kognitywnych o charakterze ostrym, w których kluczowe czynniki powiązane są wzajemnymi relacjami, a wartości zarówno czynników, jak i relacji mają wyłącznie liczbowe reprezentacje podczas całego procesu tworzenia i późniejszej eksploatacji. Relacyjne rozmyte mapy kognitywne, jako wykorzystujące specyficzny aparat arytmetyki liczb rozmytych, zostaną przedstawione odrębnie – w rozdziale 4.

2.1. OGÓLNA CHARAKTERYSTYKA – CZYNNIKI I RELACJE

Pod pojęciem Relacyjnej Mapy Kognitywnej (RMK) rozumie się zestawienie [82]:

$$< \mathbf{c}, \mathbf{x}, \mathbf{r} > \tag{2.1}$$

gdzie:

$\mathbf{c} = [c_1, c_2, \dots, c_n]^T$ – wektor czynników (ang. *concepts*) mapy kognitywnej;
$\mathbf{x} = [x_1, x_2, \dots, x_n]^T$ – wektor wartości czynników mapy kognitywnej;
$\mathbf{r} = \{r_{i,j}\}, (i, j = 1, \dots, n)$ – macierz relacji między czynnikami x_i i x_j (parametry mapy); $r_{i,j} \in [0, 1]$ lub $r_{i,j} \in [-1, 1]$;
n – liczba czynników.

Występujące w (2.1) czynniki mogą być wielkościami fizycznymi występującymi w modelowanym systemie bądź mogą mieć charakter czysto abstrakcyjny (pełnią wtedy rolę pomocniczą jako łączniki pomiędzy czynnikami kluczowymi, których odwzorowanie jest właściwym celem modelowania). W zastosowaniach praktycznych używa się na ogół tylko wartości czynników, w związku z czym w dalszych rozważaniach zależność (2.1) będzie występować w formie (2.2):

$$< \mathbf{x}, \mathbf{r} > \tag{2.2}$$

Z tego samego powodu określenia „czynnik" oraz „wartość czynnika" będą uważane za tożsame i stosowane zamiennie.

Wartości czynników, niezależnie od ich charakteru (może on być zarówno ilościowy, jak i jakościowy), rzadko są wykorzystywane w postaci uzyskanej z bezpośrednich pomiarów lub obserwacji. Przeważnie wymagają one dodatkowego przetworzenia, którego celem jest zrównoważenie ich oddziaływań. Przetworzenie to polega na poddaniu ich procedurze bezwymiarowej normalizacji bądź standaryzacji, przy czym normalizacja, z racji wbudowanych ograniczeń wartości, jest zdecydowanie bardziej przydatna. Zasadniczo, normalizacja polega na zastąpieniu bezwzględnych wartości czynników (mierzonych w określonych jednostkach) ich bezwymiarowymi, względnymi odpowiednikami, mieszczącymi się w granicach $[-1, 1]$ lub $[0, 1]$ (zależnie od charakteru projektowanej RMK). Taka operacja pozwala ujedno-

licić operacje arytmetyczne oraz, co ważniejsze, zachować właściwe proporcje pomiędzy poziomami wzajemnych oddziaływań (relacji) pomiędzy czynnikami.

Normalizacja może mieć charakter funkcyjny lub tabelaryczny. Przykłady normalizacji funkcyjnej przedstawiają równania (2.3) i (2.4):

$$x_i(t) = \frac{x_i^*(t) - \min\,(x_i^*)}{\max\left(x_i^*\right) - \min\,(x_i^*)} \tag{2.3}$$

$$x_i(t) = \mathrm{Sgn}(x_i^*(t))\frac{|x_i^*(t)|}{\max\left(|x_i^*|\right)} \tag{2.4}$$

gdzie:

t – wybrany krok czasu dyskretnego rozumianego jako zbiór kolejnych chwil czasu, próbkowanego ze stałym krokiem (przedziałem próbkowania) Δt; kolejnym krokom czasu dyskretnego nadano numery, począwszy od 0;

$x_i^*(t)$ – wartość rzeczywista i-tego czynnika w chwili t czasu dyskretnego;

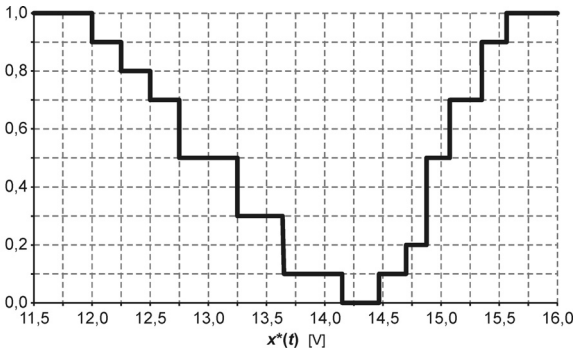
$x_i(t)$ – wartość znormalizowana i-tego czynnika w chwili t czasu dyskretnego.

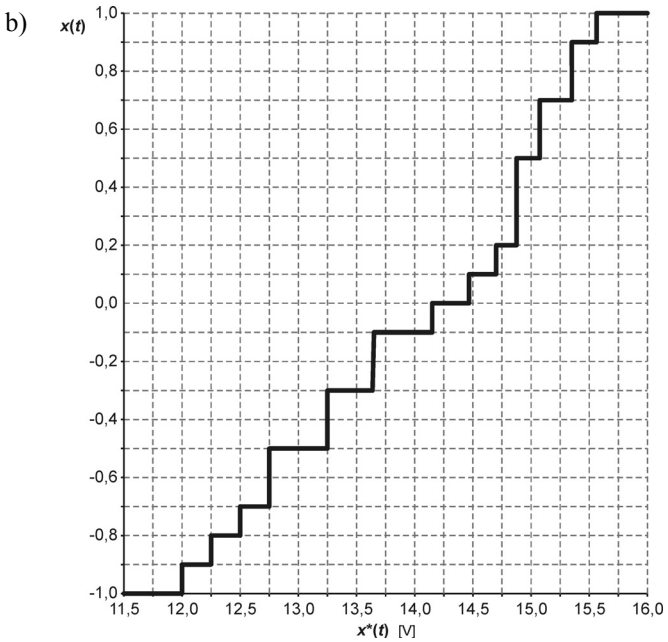
Normalizacja według (2.3) jest często stosowana w modelowaniu elementów sztucznej inteligencji (sztucznych sieci neuronowych, drzew decyzyjnych, niektórych typów map kognitywnych itp.) – np. [141, 142]. Metoda zilustrowana przez (2.4) [74] pozwala dodatkowo zachować rzeczywiste proporcje pomiędzy wartościami dodatnimi i ujemnymi.

Na rysunku 2.2 pokazano główne podejścia do normalizacji tabelarycznej (w ujęciu graficznym). Wybór któregoś z nich zależy od charakteru modelu i rodzaju pracy modelowanego obiektu. W określonych sytuacjach przebiegi pokazane na rysunku 2.2 można aproksymować przedziałami przy użyciu różnych funkcji (z przeprowadzonych testów wynika, że wystarczające do tego celu są funkcje wielomianowe drugiego rzędu).

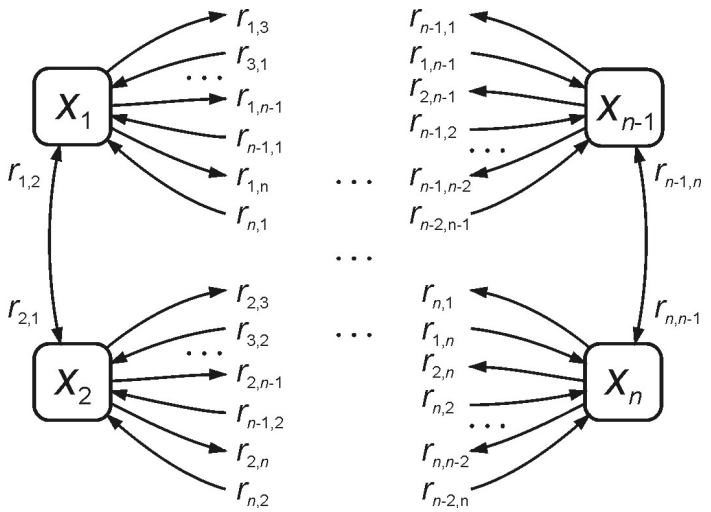
Relacyjna mapa kognitywna może przyjmować różne postaci formalne, jednakże jej reprezentacja graficzna, na poziomie ogólnym, zwykle przyjmuje formę grafu, tak jak na rysunku 2.3.

a)





Rys. 2.2. *Przykłady (w postaci reprezentacji graficznej) normalizacji tabelarycznej: a) przy założeniu zakresu znormalizowanych wartości czynników* $[0, 1]$*; b) przy założeniu zakresu znormalizowanych wartości czynników* $[-1, 1]$*.* $x^*(t)$ *– wartość rzeczywista (chwilowa) czynnika;* $x(t)$ *– wartość znormalizowana (chwilowa) czynnika*



Rys. 2.3. *Ogólna postać relacyjnej mapy kognitywnej;* $x_1 \div x_n$ *– wartości czynników;* $r_{i,j}$ *– relacja pomiędzy czynnikami:* x_i *i* x_j

Inną formą jest opis przy użyciu kwadratowej macierzy **r** zgodnie z (2.5):

$$\mathbf{r} = \begin{array}{c} \\ x_1 \\ x_2 \\ \vdots \\ x_n \end{array} \begin{array}{c} \begin{array}{cccc} x_1 & x_2 & \cdots & x_n \end{array} \\ \begin{bmatrix} 0 & r_{1,2} & \cdots & r_{1,n} \\ r_{2,1} & 0 & \cdots & r_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ r_{n,1} & r_{n,2} & \cdots & 0 \end{bmatrix} \end{array} \tag{2.5}$$

gdzie:
n – liczba czynników.

Należy zauważyć, że w ogólnym przypadku macierz relacji **r** może być tworzona przez ekspertów, z których każdy może przedstawić inne wartości poszczególnych relacji jednostkowych $r_{i,j}$. Powstanie wtedy wiele takich macierzy, a macierz wynikowa będzie efektem ich sumowania według (2.6):

$$\mathbf{r} = \mathbf{r}_1 + \cdots + \mathbf{r}_p \tag{2.6}$$

gdzie:
$\mathbf{r}_i$ – macierz relacji stworzona przez i-tego eksperta;
p – liczba ekspertów.

W takiej sytuacji konieczne są dodatkowe operacje progowe zapewniające utrzymanie elementów macierzy **r** w założonych granicach (najczęściej $[-1, 1]$). Mogą to być operacje normalizacyjne w rodzaju (2.4).

Z drugiej strony, zważywszy szerokie możliwości uczenia (adaptacji) RMK (przedstawione dalej), posiadanie dostępu do odpowiednio licznego zbioru danych odniesienia czyni tego rodzaju działania kumulacyjne mniej potrzebnymi.

Model (2.2) wymaga pewnej konkretyzacji polegającej na wyborze równań dynamiki modelowanego obiektu oraz sposobu adaptacji parametrycznej [63].

2.2. PODSTAWOWE MODELE RELACYJNYCH MAP KOGNITYWNYCH

Relacyjne mapy kognitywne mogą przyjmować różne formy, zależnie od charakteru modelowanego obiektu. Głównym wyznacznikiem wyboru którejś z nich jest odpowiedź na pytanie: czy wewnętrzna struktura modelowanego obiektu ma charakter statyczny (co oznacza, że wartość danego czynnika w kolejnej chwili czasu jest zależna od wartości pozostałych czynników oraz mocy relacji) – zgodny z ogólną postacią (2.7), czy dynamiczny (wtedy wartość czynnika w kolejnej chwili czasu zależy od mocy relacji oraz zmian wartości pozostałych czynników) – o postaci jak w (2.8). W praktycznych zastosowaniach stosuje się komputerowe metody obliczania kolejnych wartości czynników, stąd też pojęcie czasu jest w pewnym sensie umowne i dotyczy raczej czasu dyskretnego. Dlatego też w dalszych rozważaniach oznaczenie t będzie dotyczyło wybranego kroku czasu dyskretnego. Wspomniane powyżej modele przyjmują następujące postaci formalne:

a) model statyczny:

$$\mathbf{x}(t+1) = \omega(\mathbf{r}, \mathbf{x}(t)) \tag{2.7}$$

b) model dynamiczny:

$$\mathbf{x}(t+1) = \omega(\mathbf{r}, \mathbf{x}(t), \Delta\mathbf{x}(t)) \tag{2.8}$$

gdzie:
$\mathbf{x}$ – wektor wartości czynników;
ω – funkcja kumulacji oddziaływań;
$\Delta\mathbf{x}(t) = \mathbf{x}(t) - \mathbf{x}(t-1)$;
$\mathbf{x}(0)$ – początkowy wektor wartości czynników (zadany).

Główne możliwe do wykorzystania typy modeli RMK [63], wynikające z równań (2.7) i (2.8), w odniesieniu do wartości pojedynczego (j-tego) czynnika, przedstawiono poniżej w równaniach (2.9)–(2.13):

a) model liniowy z nieliniowością dodatnią

$$x_j(t+1) = x_j(t) + f_p\left(\sum_{\substack{i=1 \\ i \neq j}}^{n} r_{i,j} x_i(t)\right) \tag{2.9}$$

gdzie:
$t = 0, \ldots, T$; $x_i(0)$ – wartość zadana; $j = 1, \ldots, n$;

b) model nieliniowy

$$x_j(t+1) = f_p\left(x_j(t) + \sum_{\substack{i=1 \\ i \neq j}}^{n} r_{i,j} x_i(t)\right) \tag{2.10}$$

gdzie:
$t = 0, \ldots, T$; $x_i(0)$ – wartość zadana; $j = 1, \ldots, n$;

c) model liniowy z nieliniowością szybkości zmian

$$x_j(t+1) = x_j(t) + f_p\left(\sum_{\substack{i=1 \\ i \neq j}}^{n} r_{i,j} \Delta x_i(t)\right) \tag{2.11}$$

gdzie:
$\Delta x_i(t) = x_i(t) - x_i(t-1)$; $x_i(0)$, $x_i(-1)$ – zadane; $j = 1, \ldots, n$;

d) model nieliniowy z nieliniowością szybkości zmian

$$x_j(t+1) = f_p\left(x_j(t) + \sum_{\substack{i=1 \\ i\neq j}}^{n} r_{i,j}\Delta x_i(t)\right) \tag{2.12}$$

gdzie:
$\Delta x_i(t) = x_i(t) - x_i(t-1)$; $x_i(0)$, $x_i(-1)$ – zadane; $j = 1, \dots, n$;

e) model nieliniowy z nieliniowością dodatnią

$$x_j(t+1) = f_p\left(\sum_{\substack{i=1 \\ i\neq j}}^{n} r_{i,j}\Delta x_i(t)\right) \tag{2.13}$$

gdzie:
$\Delta x_i(t) = x_i(t) - x_i(t-1)$; $x_i(0)$, $x_i(-1)$ – zadane; $j = 1, \dots, n$;

W modelach (2.9)–(2.13) nieliniowa funkcja progowa $f_p(v)\colon \mathbb{R} \longrightarrow \mathbb{R}$ może przybrać jedną z następujących postaci:

a) liniowa z ograniczeniami

$$f_p(v) = \begin{cases} -1 & \text{dla} \quad v \leq -a \\ 0 & \text{dla} \quad v = 0 \\ \dfrac{v}{a} & \text{dla} \quad 0 < |v| < a \\ 1 & \text{dla} \quad v \geq a \end{cases} \tag{2.14}$$

gdzie:
$a > 0$ – parametr;

b) sigmoidalna

$$f_p(v) = \frac{1}{1 + e^{-\beta v}} \tag{2.15}$$

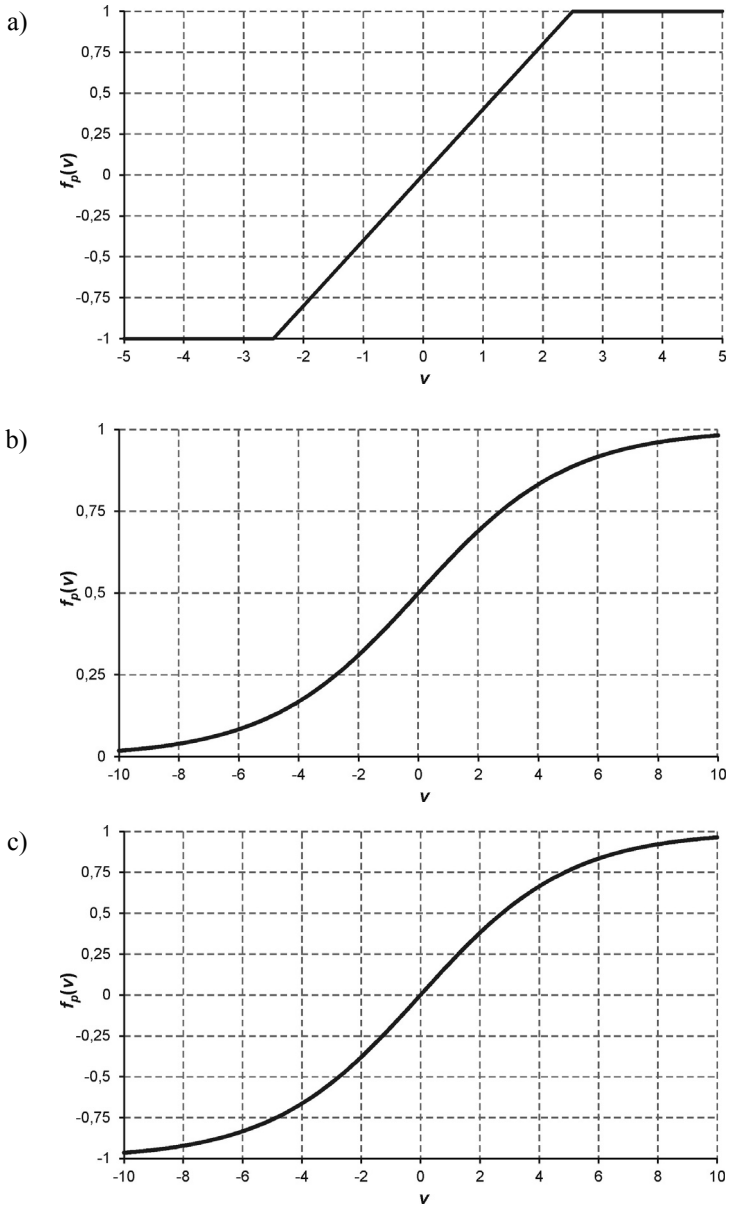
gdzie:
$\beta > 0$ – parametr;

c) tangensoidalna

$$f_p(v) = \frac{1 - e^{-\beta v}}{1 + e^{-\beta v}} \tag{2.16}$$

gdzie:
$\beta > 0$ – parametr.

Na rysunku 2.4 przedstawiono przykładowe graficzne reprezentacje postaci funkcji progowej $f_p(v)$ opisanych równaniami (2.14)–(2.16).



Rys. 2.4. *Graficzne reprezentacje funkcji progowej f_p: a) postać liniowa z ograniczeniami (opisana równaniem (2.14)); b) postać sigmoidalna (opisana równaniem (2.15)); c) postać tangensoidalna (opisana równaniem (2.16))*

Uwaga 2.1

Zadaniem funkcji f_p z (2.14)–(2.16) jest zapewnienie stabilności modelu. W określonych sytuacjach można zrezygnować z jej stosowania. Trzeba też pamiętać, że w modelach (2.9) i (2.11) działanie tej funkcji dotyczy tylko części wyrażenia wynikowego. W tych modelach trzeba więc wprowadzać dodatkowe mechanizmy progowe albo należy ograniczać ich stosowanie do obiektów, dla których nie występuje zagrożenie przekroczenia progowych wartości czynników.

Typ modelu najbardziej odpowiedni dla danego obiektu (systemu) dobiera się na podstawie wiedzy ekspertowej, podobnie, w oparciu o to samo źródło, określa się charakter kluczowych czynników oraz sposoby normalizacji ich wartości. Można też, choć nie jest to konieczne, dokonać wstępnego oszacowania wartości mocy relacji wiążących czynniki. W większości przypadków ekspertowe zdefiniowanie mocy relacji nie prowadzi do stworzenia modelu o optymalnych parametrach. Model taki może powstać dopiero po przeprowadzeniu odpowiedniej adaptacji parametrów (głównie mocy relacji), czyli w procesie uczenia modelu.

2.3. TECHNIKI UCZENIA RELACYJNYCH MAP KOGNITYWNYCH

Parametry RMK (chodzi tu głównie o moce relacji pomiędzy czynnikami) mogą być dobrane w drodze bezpośredniego planowania – na podstawie wiedzy ekspertowej. Mapa taka w zasadzie mogłaby spełniać kryteria poprawności działania – przy założeniu, że eksperci określający parametry dysponują wystarczającą wiedzą. W większości przypadków tak nie jest, stąd też konieczność dodatkowej adaptacji parametrów mapy (uczenia) w oparciu o posiadane dane uzyskane w drodze pomiarów lub obserwacji obiektu rzeczywistego. Przez uczenie będzie rozumiany proces takiej modyfikacji wartości mocy relacji modelu, aby model ten dokładnie odwzorowywał oczekiwane zachowanie obiektu. Zasadniczo, techniki uczenia RMK (podobnie jak w innych typach modeli inteligentnych) można podzielić na dwie grupy: uczenie nadzorowane i uczenie nienadzorowane. Proces uczenia nienadzorowanego (np. w oparciu o metodę Hebba [141, 142]) nie różni się, co do zasady, od metod znanych z literatury. Jest on mniej przydatny w modelowaniu pracy konkretnych układów, ponieważ metody uczenia nienadzorowanego, choć mogą dobrze sprawdzać się w zadaniach grupowania, stwarzają ryzyko pominięcia istotnych dla celów modelowania własności obiektu, zwykle wymagają też zaangażowania większej liczby czynników, co wpływa na czas obliczeń. Dodatkowo, stosowanie metod tego rodzaju uniemożliwia w zasadzie zaplanowanie reakcji modelu dokładnie odpowiadającej reakcji modelowanego systemu. Z wyżej wymienionych względów, w niniejszym rozdziale zostanie szerzej przedstawiona jedynie metoda uczenia nadzorowanego. Metody nadzorowane wykorzystują fakt posiadania dokładnej wiedzy na temat próbki uczącej, tzn. próbka ta jest po pierwsze – odpowiednio liczna, a po drugie – znane są wartości wszystkich czynników (w zbiorze próbki uczącej) w danym kroku czasu dyskretnego. Dzięki temu podstawą modyfikacji mocy poszczególnych relacji

jest porównanie aktualnych wyników pracy modelu z wynikami oczekiwanymi. W opracowaniach dotyczących technik uczenia nadzorowanego map kognitywnych można znaleźć wiele metod, głównie populacyjnych (najważniejsze z nich zostały w skrócie opisane w podrozdziale 3.2.4), które można by wykorzystać również w modelach RMK, jednakże ich wspólną wadą jest stosowanie pewnej przypadkowości podczas zmian wartości adaptowanych parametrów. Wszystkie one posługują się w pewnym sensie wspólną, różnicową metodą zmiany wartości kolejnych parametrów, polegającą na dokonywaniu drobnych zmian mocy relacji ($r_{i,j}$) przy jednoczesnej obserwacji wpływu tych zmian na pracę modelu (można przy tym, zamiast zmian losowych dokonywać zmian zaplanowanych zgodnie z określonym algorytmem [61, 169, 218]). Zmniejszanie różnicy pomiędzy wynikami pracy modelu a wartościami oczekiwanymi powoduje utrwalenie kierunku zmian, podczas gdy jej zwiększanie wymusza zmianę tego kierunku. Opierając się na wyżej wymienionym schemacie opracowano szereg metod, w większości wykorzystujących pewną randomizację wyboru kierunku i wielkości zmian, przy zachowaniu dość dużej automatyki algorytmu. Można do nich zaliczyć podejścia wykorzystujące, np. algorytmy genetyczne [17, 184], optymalizację rojem cząstek [144] i inne techniki ewolucyjne [9]. Jednakże, w pełni skalarny charakter danych RMK pozwala na opracowanie innych, szybszych i bardziej dokładnych metod wykorzystujących znane (m.in. z konstruowania sztucznych sieci neuronowych) reguły propagacji wstecznej [113] i gradientową analizę zmienności wybranych parametrów. Podobne do tego ostatniego podejścia metody były częściowo wykorzystywane przy budowie jednokierunkowych sztucznych sieci neuronowych [8, 16, 32, 92, 103, 113, 122, 202], jednak nie próbowano wcześniej oprzeć na nich techniki uczenia map kognitywnych. Potencjał takiej metody został dowiedziony w badaniach nad uczeniem RMK [63, 71, 72, 172]. Z uwagi na zalety zmodyfikowanego podejścia gradientowego w procesach adaptacji parametrów struktury modelowej RMK, poniżej zostanie ono szerzej przedstawione.

Gradientowa (nadzorowana) metoda uczenia RMK polega na minimalizacji następującego kryterium:

$$J(\mathbf{r}) = \frac{1}{2}\sum_{i=1}^{n} \delta_i(t)^2 \longrightarrow \min_{\mathbf{r}} \tag{2.17}$$

gdzie:
$\delta_i(t)^2 = (z_i(t) - x_i(t))^2$;
$z_i(t)$ – zadana (wzorcowa) funkcja wartości zmian i-tego czynnika;
$x_i(t)$ – modelowana funkcja wartości zmian i-tego czynnika;
n – liczba czynników ($i = 1, \dots, n$);
t – czas dyskretny ($t = 0, 1, \dots, T$).

Należy zauważyć, że użyte w (2.17) kryterium dla chwili czasu dyskretnego t można zastąpić kryterium dla rozpatrywanej liczby kroków tego czasu T, przedstawione równaniem (2.18):

$$J_1(\mathbf{r}) = \frac{1}{2T}\sum_{t=1}^{T}\sum_{i=1}^{n}\delta_i(t)^2 \longrightarrow \min_{\mathbf{r}} \tag{2.18}$$

W proponowanej metodzie gradientowej wystarczające jest zastosowanie kryterium (2.17). Dla jego minimalizacji można przyjąć rekurencyjny algorytm adaptacji relacji (dla $t \longrightarrow t+1$) o następującej postaci:

$$r_{i,j}(t+1) = r_{i,j}(t) + \Delta J(t) \tag{2.19}$$

gdzie:
$\Delta J(t)$ – zmiana wartości funkcji $J(t)$ w zależności od parametrów.

Reprezentatywnym przykładem $\Delta J(t)$ jest kierunek antygradientowy:

$$\Delta J(t) = -\text{grad}\, J(t), \tag{2.20}$$

który dla $r_{i,j}$ można wyznaczyć w następujący sposób:

$$\text{grad}\, J(t) = -\big(z_j(t) - x_j(t)\big)\cdot y_{i,j} \tag{2.21}$$

gdzie:
$y_{i,j}$ – funkcja wrażliwości czynnika x_j na zmiany wartości $r_{i,j}$ $(i = 1, \dots, n;\ i \neq j)$, wyznaczana zgodnie z (2.22):

$$y_{i,j} = \frac{\partial x_j}{\partial r_{i,j}} \tag{2.22}$$

Po uwzględnieniu (2.21) i (2.22) algorytm (2.19) można zapisać w postaci (2.23):

$$r_{i,j}(t+1) = r_{i,j}(t) + \eta\cdot\delta_i(t)\cdot y_{i,j}(t) \tag{2.23}$$

gdzie:
$0 < \eta < 1$ – krok algorytmu.

Uwaga 2.2

Równania (2.22) i (2.23) stanowią rdzeń opracowanej metody i ich prawidłowe rozwiązanie jest kluczowe dla poprawnej pracy algorytmu uczącego. Forma równań wyznaczających wrażliwość $y_{i,j}$ z (2.22) i (2.23) zależy od przyjętej postaci

funkcji progowej f_p (z uwagi na konieczność wyznaczania pochodnych cząstkowych), np. dla modelu z równania (2.9), w którym zastosowano funkcję progową według (2.16), funkcję wrażliwości wyznacza się zgodnie z (2.24):

$$y_{i,j}(t+1) = y_{i,j}(t) + x_i(t) \cdot \frac{2 \cdot \beta \cdot e^{-\beta \cdot \left(\sum_{\substack{i=1 \\ i \neq j}}^{n} r_{i,j} \cdot x_i(t) \right)}}{\left[1 + e^{-\beta \cdot \left(\sum_{\substack{i=1 \\ i \neq j}}^{n} r_{i,j} \cdot x_i(t) \right)} \right]^2} \tag{2.24}$$

gdzie:
$y_{i,j}(t) = \frac{\partial x_j(t)}{\partial r_{i,j}}$; $i, j = 1, \dots, n$.

Biorąc pod uwagę powyższe rozważania, można sformułować układy równań dla adaptacji parametrycznej (uczenia nadzorowanego) poszczególnych typów relacyjnych map kognitywnych:

1. Model symulacyjny adaptacyjny 1 (MSA1) dla mapy opisanej równaniem (2.9):

$$\begin{cases} x_{j,S}(t) = \sum_{\substack{i=1 \\ i \neq j}}^{n} r_{i,j}(t) \cdot x_i(t) \\ x_j(t+1) = x_j(t) + f_d\left(x_{j,S}(t)\right) \\ r_{i,j}(t+1) = r_{i,j}(t) + \eta \cdot \delta_j(t) \cdot y_{i,j}(t) \\ y_{i,j}(t+1) = y_{i,j}(t) + \frac{\partial f_d}{\partial x_{j,S}(t)} \cdot x_i(t) \\ i = 1, \dots, n;\ j = 1, \dots, n;\ j \neq i;\ 0 < \eta < 1;\ t = 0, 1, \dots, T \end{cases} \tag{2.25}$$

gdzie:
$x_{j,S}(t)$ – skumulowane oddziaływanie pozostałych czynników na czynnik j-ty.

Warunki początkowe dla modelu MSA1 z (2.25):

$$\begin{cases} x_j(0) - \text{ zadana (np. } x_j(0) = 0 \text{ przy braku informacji)} \\ y_{i,j}(0) - \text{ zadana (np. } y_{i,j}(0) = 0 \text{ przy braku informacji)} \\ r_{i,j}(0) - \text{ zadana (np. } r_{i,j}(0) = 0 \text{ przy braku informacji)} \\ i = 1, \dots, n;\ j = 1, \dots, n;\ i \neq j \end{cases} \tag{2.26}$$

2. Model symulacyjny adaptacyjny 2 (MSA2) dla mapy opisanej równaniem (2.10)

$$\begin{cases} x_{j,S}(t) = x_i(t) + \sum\limits_{\substack{i=1 \\ i \neq j}}^{n} r_{i,j}(t) \cdot x_i(t) \\ x_j(t+1) = f_d\big(x_{j,S}(t)\big) \\ r_{i,j}(t+1) = r_{i,j}(t) + \eta \cdot \delta_j(t) \cdot y_{i,j}(t) \\ y_{i,j}(t+1) = \dfrac{\partial f_d}{\partial x_{j,S}(t)} \cdot \big(y_{i,j}(t) + X_i(t)\big) \\ i = 1, \dots, n;\ j = 1, \dots, n;\ j \neq i;\ 0 < \eta < 1;\ t = 0, 1, \dots, T \end{cases} \tag{2.27}$$

gdzie:
$x_{j,S}(t)$ – skumulowane oddziaływanie pozostałych czynników na czynnik j-ty.

Warunki początkowe dla MSA2 z (2.27) – jak w (2.26).

3. Model symulacyjny adaptacyjny 3 (MSA3) dla mapy opisanej równaniem (2.11):

$$\begin{cases} x_{j,S}(t) = \sum\limits_{\substack{i=1 \\ i \neq j}}^{n} r_{i,j}(t) \cdot (x_i(t) - x_i(t-1)) \\ x_j(t+1) = x_j(t) + f_d\big(x_{j,S}(t)\big) \\ r_{i,j}(t+1) = r_{i,j}(t) + \eta \cdot \delta_j(t) \cdot y_{i,j}(t) \\ y_{i,j}(t+1) = y_{i,j}(t) + \dfrac{\partial f_d}{\partial x_{j,S}(t)} \cdot (x_i(t) - x_i(t-1)) \\ i = 1, \dots, n;\ j = 1, \dots, n;\ j \neq i;\ 0 < \eta < 1;\ t = 0, 1, \dots, T \end{cases} \tag{2.28}$$

gdzie:
$x_{j,S}(t)$ – skumulowane oddziaływanie pozostałych czynników na czynnik j-ty.

Warunki początkowe dla modelu MSA3 z (2.28):

$$\begin{cases} x_j(0) - \text{ zadana (np. } x_j(0) = 0 \text{ przy braku informacji)} \\ x_j(-1) = 0 \\ y_{i,j}(0) - \text{ zadana (np. } y_{i,j}(0) = 0 \text{ przy braku informacji)} \\ r_{i,j}(0) - \text{ zadana (np. } r_{i,j}(0) = 0 \text{ przy braku informacji)} \\ i = 1, \dots, n;\ j = 1, \dots, n;\ i \neq j \end{cases} \tag{2.29}$$

4. Model symulacyjny adaptacyjny 4 (MSA4) dla mapy opisanej równaniem (2.12):

$$\begin{cases} x_{j,S}(t) = X_i(t) + \sum\limits_{\substack{i=1 \\ i\neq j}}^{n} r_{i,j}(t) \cdot (x_i(t) - x_i(t-1)) \\ x_j(t+1) = f_d\big(x_{j,S}(t)\big) \\ r_{i,j}(t+1) = r_{i,j}(t) + \eta \cdot \delta_j(t) \cdot y_{i,j}(t) \\ y_{i,j}(t+1) = \dfrac{\partial f_d}{\partial x_{j,S}(t)} \cdot \big(y_{i,j}(t) + x_i(t) - x_i(t-1)\big) \\ i = 1, \dots, n;\ j = 1, \dots, n;\ j \neq i;\ 0 < \eta < 1;\ t = 0, 1, \dots, T \end{cases} \tag{2.30}$$

gdzie:
$x_{j,S}(t)$ – skumulowane oddziaływanie pozostałych czynników na czynnik j-ty.

Warunki początkowe dla MSA4 z (2.30) – jak w (2.29).

5. Model symulacyjny adaptacyjny 5 (MSA5) dla mapy opisanej równaniem (2.13):

$$\begin{cases} x_{j,S}(t) = \sum\limits_{\substack{i=1 \\ i\neq j}}^{n} r_{i,j}(t) \cdot (x_i(t) - x_i(t-1)) \\ x_j(t+1) = f_d\big(x_{j,S}(t)\big) \\ r_{i,j}(t+1) = r_{i,j}(t) + \eta \cdot \delta_j(t) \cdot y_{i,j}(t) \\ y_{i,j}(t+1) = \dfrac{\partial f_d}{\partial X_{j,S}(t)} \cdot (x_i(t) - x_i(t-1)) \\ i = 1, \dots, n;\ j = 1, \dots, n;\ j \neq i;\ 0 < \eta < 1;\ t = 0, 1, \dots, T \end{cases} \tag{2.31}$$

gdzie:
$x_{j,S}(t)$ – skumulowane oddziaływanie pozostałych czynników na czynnik j-ty.

Warunki początkowe dla MSA5 z (2.31) – jak w (2.29).

Występujący we wszystkich wyżej wymienionych modelach adaptacyjnych (równania (2.25)–(2.31)) współczynnik korekcji η (zwany też współczynnikiem uczenia) służy do dostosowania szybkości zmian mocy relacji do aktualnego stanu modelu. W ogólnym przypadku jego wartość może być zmienna (np. może maleć w miarę zbliżania się do stanu ustalonego). Założenie wyższej wartości η pozwala skrócić czas adaptacji, jednakże może się to odbić niekorzystnie na wynikowej dokładności modelu.

Uwaga 2.3

W modelach MSA1–MSA5, w równaniu wrażliwości (przy realizacji programowej), w analizie symulacyjnej na potrzeby uczenia (adaptacji) można zastąpić bieżące wartości kluczowych czynników $x_j(t)$ $(j = 1, \dots, n;\ j \neq i$ dla $i = 1, \dots, n)$ wartościami znanymi (np. zmierzonymi) z_j $(j = 1, \dots, n;\ j \neq i)$.

Uwaga 2.4

Funkcja progowa f_p w równaniach (2.9)–(2.13) ma istotne znaczenie zwłaszcza na etapie uczenia modelu. Podczas późniejszej eksploatacji, jeśli dane uczące były wystarczająco reprezentatywne, może się ona okazać zbędna. Decyzję co do jej stosowania podczas normalnej pracy modelu należy podjąć po dłuższej obserwacji wartości przyjmowanych przez poszczególne czynniki.

Praca przedstawionego powyżej typu modelu relacyjnej mapy kognitywnej została pokazana, na przykładzie systemu monitorowania, w podrozdziale 5.1.

3 LOGIKA ROZMYTA W PROJEKTOWANIU MAP KOGNITYWNYCH

W rozdziale 2 przedstawiono koncepcję i sposób konstruowania relacyjnej mapy kognitywnej o charakterze ostrym, tzn. takim, w którym opis matematyczny działania modelu jest dokonywany z użyciem liczb rzeczywistych. Modele tego rodzaju mogą skutecznie zastępować inne tzw. inteligentne podejścia do modelowania złożonych systemów o niepełnej informacji, zwłaszcza gdy zachodzi potrzeba uwzględniania wewnętrznej dynamiki przepływu sygnałów. Podejście takie, chociaż może skompensować niepewność informacji, nie uwzględnia jej nieprecyzyjności, wynikającej z jakościowego opisu zjawisk. Nieprecyzyjność (pojęcie to nie jest tożsame z niepewnością) wynika z jakościowego podejścia do opisu działania układów, a takie podejście legło u podstaw opracowania teorii zbiorów rozmytych i logiki rozmytej, które zostały pokrótce przedstawione w dalszej części pracy, w podrozdziale 3.1. Na tej bazie opracowano również szereg zasad wprowadzania tej teorii do modeli map kognitywnych, co zaowocowało opracowaniem teorii i praktyki tzw. rozmytych map kognitywnych, przedstawionych w podrozdziale 3.2.

Samo pojęcie rozmytej mapy kognitywnej (ang. *Fuzzy Cognitive Map*) zostało wprowadzone w 1986 r. [108] w odniesieniu do modeli wykorzystujących rozwijające się wtedy deskryptywne podejście do logiki rozmytej [193]. W późniejszych latach, w miarę pojawiania się kolejnych modyfikacji tego podejścia (np. [124]) rozwijano nowe metody uczenia i implementowano rozmyte mapy kognitywne do kolejnych dziedzin [22, 30, 167, 188, 189]. Obecnie jest to jedna z ważnych gałęzi inteligencji obliczeniowej, znajdująca zastosowanie w klasyfikacji, predykcji i monitorowaniu pracy złożonych systemów.

Podstawą działania większości współczesnych zastosowań rozmytych map kognitywnych były konstrukcje oparte na zbiorach reguł decyzyjnych typu:

$$\text{IF } X_1 = WL_1 \text{ AND } X_2 = WL_2 \text{ AND } \ldots \text{ AND } X_{j-1} = WL_{j-1} \text{ AND } X_{j+1} = WL_{j+1} \text{ AND } \ldots \text{ AND } X_n = WL_n \text{ THEN } X_j = WL_j \tag{3.1}$$

gdzie:

WL – jedna z wartości lingwistycznych używanych w modelu rozmytej mapy kognitywnej (pojęcie wartości lingwistycznej zostanie wyjaśnione w podrozdziale 3.1).

Regułowa metoda budowy rozmytej mapy kognitywnej, której zasadę pokazuje (3.1), zostanie przybliżona w podrozdziale 3.2. Jest ona skuteczna, jednak jej praktyczne stosowanie wiąże się z licznymi problemami natury praktycznej. Po pierwsze, zbiór reguł, w zasadzie, musi być tworzony bezpośrednio przez eksperta, co utrudnia pełną automatyzację procesu projektowania. Po drugie, uczenie rozmytej mapy kognitywnej jest procesem trudnym, ponieważ również powinno się opierać na konstruk-

cjach typu IF-THEN. W większości współczesnych metod uczenia (przedstawionych w podrozdziale 3.2.4) stosuje się rozmaite „wybiegi" mające na celu zastąpienie konstrukcji stricte rozmytych ich liczbowymi odpowiednikami, lecz takie sposoby legitymizują odejście od podstawowej zalety logiki rozmytej, jaką jest odwzorowywanie wartości czynników za pomocą zmiennych łańcuchowych oddających nieprecyzyjność. Po trzecie, modyfikacja modelu rozmytej mapy kognitywnej jest trudna, gdyż z założenia pociąga za sobą konieczność modyfikacji wszystkich reguł, co implikuje ponowne angażowanie ekspertów do opracowywania właściwie całego modelu od podstaw, a to jest działaniem czasochłonnym i narażonym na błędy.

Zaproponowana w pracy nowa konstrukcja Relacyjnej Rozmytej Mapy Kognitywnej (RRMK) jest odpowiednikiem ostrej Relacyjnej Mapy Kognitywnej (2.2) w przestrzeni zbiorów rozmytych. Model tego rodzaju może mieć zastosowanie w modelowaniu i monitorowaniu układów nieprecyzyjnych, podobnie jak klasyczne modele rozmytych map kognitywnych, z tym, że w procesach projektowania i późniejszego działania wykorzystuje się w nim architekturę opartą na arytmetyce liczb rozmytych, co pozwala przezwyciężyć większość niedogodności występujących w klasycznym podejściu do budowy i eksploatacji rozmytych map kognitywnych.

Proponowana konstrukcja RRMK, w której podstawą działania są liczby rozmyte, relacje rozmyte i operacje arytmetyki liczb i relacji rozmytych, pozwala uzyskać strukturę wynikową w pewnym sensie podobną do rozmytej mapy kognitywnej. Główna różnica polega na zachowaniu rozmytego charakteru wszystkich elementów modelu przez cały czas trwania procesu jego budowy i eksploatacji. Jest to możliwe dzięki zastosowaniu specjalnego aparatu matematycznego, pozwalającego zautomatyzować czynności projektowania i uczenia modelu, dzięki czemu jest on bardziej elastyczny i łatwiejszy do praktycznego stosowania. Rozwiązania takie nie były dotychczas stosowane z uwagi na trudności praktycznego stosowania aparatu matematycznego algebry rozmytej w dyskretnym środowisku algorytmów komputerowych. Nie było także metod zautomatyzowanego opracowywania kształtów relacji rozmytych o pożądanych właściwościach. Przedstawione tu podejście obejmuje rozwiązania obu tych trudności.

W celu ułatwienia prezentacji metody, w niniejszym rozdziale zostaną zaprezentowane najważniejsze pojęcia z dziedziny zbiorów rozmytych, następnie klasyczne podejście do tworzenia rozmytych map kognitywnych. W rozdziale następnym, na tle podstaw arytmetyki rozmytej, zostanie pokazana metoda tworzenia i działanie modelu RRMK.

3.1. LOGIKA ROZMYTA I ZBIORY ROZMYTE – PODSTAWOWE POJĘCIA

Teorię zbiorów rozmytych wprowadził w 1965 r. Lotfi Zadeh [221, 222] jako narzędzie do reprezentacji i przetwarzania informacji o nieprecyzyjnym charakterze. Teoria ta budziła przez lata duże wątpliwości w środowisku naukowym, jednakże dalsze prace badawcze i aplikacyjne stopniowo utwierdzały jej pozycję jako ważnego elementu tzw. inteligencji obliczeniowej. Przełomowe znaczenie miały prace z lat: 1970 [13] i 1972 [220], w których przedstawiono nowe podejścia do procesów decy-

zyjnych w warunkach rozmytości oraz do sterowania rozmytego. Wprowadzenie w 1973 r. [223] pojęcia zmiennej lingwistycznej, rozmytego zdania warunkowego (reguły warunkowej) typu IF-THEN oraz złożeniowych reguł wnioskowania pozwoliło usystematyzować dalsze prace nad tą dziedziną, której rozwój trwa nieprzerwanie do dnia dzisiejszego. W Polsce dość wcześnie podjęto prace nad aplikacyjnymi [26, 89] oraz teoretycznymi [145] aspektami teorii zbiorów rozmytych. Obecnie zajmuje się nimi wiele krajowych i międzynarodowych zespołów badawczych (m.in. [19, 25, 88, 91, 103, 104, 106, 111, 121, 134], jak również [146, 151, 155, 157, 160]).

Pod pojęciem „nieprecyzyjności" można rozumieć opis zjawisk i pojęć przy użyciu określeń jakościowych typu: „ciepło", „zimno", „wysoki", „niski", „szybko", „powoli" itp., które mają w dużej mierze subiektywny charakter, co bardzo utrudnia ich interpretację matematyczną. Na przykład określenie „bardzo ciepło" dla jednego obserwatora może oznaczać 35°C, podczas gdy dla innego będzie to 28°C. Istnieje też grupa zjawisk niepoddająca się pomiarom, jak np. uroda, poczucie humoru, kreatywność, których obiektywne liczbowe przedstawianie nie było w ogóle możliwe. Z drugiej strony ten sposób charakteryzowania zjawisk jest powszechnie stosowany i intuicyjnie rozumiany, więc potrzeba jakiegoś ich teoretycznego usystematyzowania jest oczywista.

3.1.1. Pojęcie zbioru rozmytego

Konwencjonalne podejście do teorii zbiorów zakłada, że zbiór służy prawidłowemu określeniu pewnego pojęcia oraz że istnieje pewien obszar rozważań (używa się też określeń: zbiór odniesienia, uniwersum, zbiór uniwersalny) zawierający wszystkie elementy istotne z punktu widzenia tego pojęcia. W takim ujęciu zbiór będzie tożsamy z funkcją charakterystyczną:

$$\varphi_A : U \longrightarrow \{0, 1\} \tag{3.2}$$

gdzie:
A – rozważany zbiór konwencjonalny,
U – przestrzeń rozważań (uniwersum).

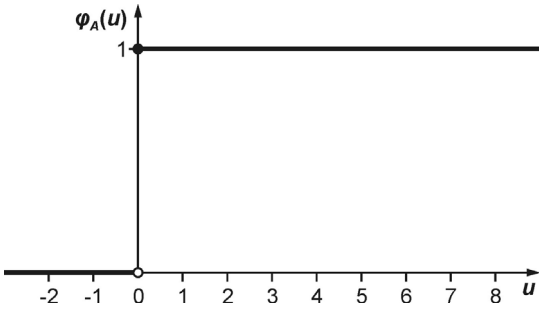
Funkcja charakterystyczna φ_A przypisuje każdemu elementowi u przestrzeni U liczbę $\varphi_A(u) \in \{0, 1\}$ w taki sposób, że jeśli $\varphi_A(u) = 1$, to element u należy do zbioru A, jeśli natomiast $\varphi_A(u) = 0$, to element u nie należy do zbioru A. Przykładem konwencjonalnego zbioru może być ilustracja pojęcia: „liczby naturalne", które można przedstawić w postaci następującego zbioru $\{u \in \mathbb{Z} : u \geq 0\}$ i zilustrować graficznie jak na rysunku 3.1.

W zbiorze konwencjonalnym następuje jednoznaczne, ostre przejście od elementów nienależących do zbioru A do elementów, które do tego zbioru należą. W przypadku pojęć nieprecyzyjnych, w rodzaju „liczby całkowite w przybliżeniu równe 100", nie może być mowy o jednoznacznym przyporządkowaniu elementu do zbioru bądź też o wykluczeniu tego elementu ze zbioru. Czy liczba 99 jest w przybliżeniu równa 100? A jeśli tak, to co z liczbą 95? Z takimi i bardziej złożo-

nymi rodzajami nieprecyzyjności można sobie poradzić, przenosząc rozważania na zaproponowany przez L. Zadeha [222] grunt zbiorów rozmytych, w których funkcja charakterystyczna (3.2) została zastąpiona przez funkcję przynależności (3.3):

$$\mu_A : U \longrightarrow [0,1], \tag{3.3}$$

która obrazuje stopień, w jakim element u należy do zbioru rozmytego A. Brak przynależności elementu u do zbioru rozmytego A występuje w sytuacji, w której $\mu_A(u) = 0$. Wartość $\mu_A(u) = 1$ oznacza całkowitą przynależność elementu u do zbioru rozmytego A. Oprócz wyżej przedstawionych sytuacji skrajnych istnieje możliwość częściowej przynależności i wtedy $0 < \mu_A(u) < 1$.



Rys. 3.1. *Funkcja charakterystyczna konwencjonalnego zbioru „liczby naturalne”*

Formalnie rzecz ujmując, zbiór rozmyty A w przestrzeni rozważań $U = \{u\}$ definiuje się następująco [88]:

$$A = \{(\mu_A(u), u) : u \in U, \mu_A(u) \in [0,1]\} \tag{3.4}$$

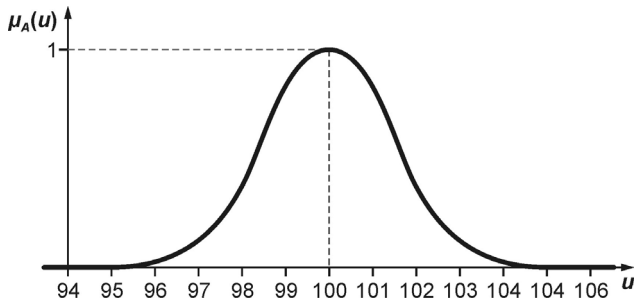
gdzie:
$\mu_A : U \longrightarrow [0,1]$ – funkcja przynależności zbioru rozmytego A,
$\mu_A(u) \in [0,1]$ – stopień przynależności elementu $u \in U$ w zbiorze rozmytym A.

Po zastosowaniu podejścia jak w (3.4) można przedstawić pojęcie „liczby całkowite w przybliżeniu równe 100” poprzez jego graficzną reprezentację pokazaną na rysunku 3.2.

W realizacjach praktycznych przyjmuje się skończone rozmiary obszarów rozważań (uniwersum). Para $\{(\mu_A(u), u)\}$ będzie zapisywana jako $\mu_A(u)/u$ (element taki nosi nazwę „singletonu rozmytego” [88]). Po wprowadzeniu wspomnianych modyfikacji zbiór rozmyty A w U z (3.4) będzie zapisywany w formie (3.5) [88]:

$$A = \left\{\frac{\mu_A(u)}{u}\right\} = \frac{\mu_A(u_1)}{u_1} + \cdots + \frac{\mu_A(u_k)}{u_k} = \sum_{i=1}^{k} \frac{\mu_A(u_i)}{u_i} \tag{3.5}$$

gdzie:
k – liczba singletonów rozmytych zbioru rozmytego A.



Rys. 3.2. *Przykładowa funkcja przynależności rozmytego zbioru „liczby całkowite w przybliżeniu równe 100”*

Stosując zapis (3.5), należy pamiętać, że operatory $+$ i $\sum$ mają tu charakter mnogościowy. Ponadto w zapisie tym pomija się singletony rozmyte, w których $\mu_A(u) = 0$.

3.1.2. Podstawowe własności zbiorów rozmytych

Podstawowe własności zbiorów rozmytych są określane przez charakterystykę operacji na takich zbiorach oraz poprzez główne typy funkcji charakterystycznych służących do definiowania tych zbiorów. Z uwagi na różne definicje spotykane w odniesieniu do różnych zastosowań, poniżej zostaną przytoczone tylko najczęściej spotykane podejścia do niektórych własności zbiorów rozmytych.

1. Nośnik zbioru rozmytego.

Nośnik (ang. *support*) lub baza zbioru rozmytego A w U jest dystrybutywnym zbiorem (nierozmytym) elementów przestrzeni rozważań U, dla których wartość funkcji przynależności jest większa od zera (jak w (3.6)):

$$\text{supp}(A) = \{u \in U : \mu_A(u) > 0\}, \tag{3.6}$$

przy tym: $\emptyset \subseteq \text{supp}(A) \subseteq U$.

2. Jądro zbioru rozmytego.

Jądro (ang. *core*) zbioru rozmytego A w U jest dystrybutywnym zbiorem (nierozmytym) elementów przestrzeni rozważań U, dla których wartość funkcji przynależności jest równa 1 (jak w (3.7)):

$$\text{core}(A) = \{u \in U : \mu_A(u) = 1\}, \tag{3.7}$$

3. Zbiór rozmyty pusty.

Zbiór rozmyty A w U jest pusty, jeśli wartości funkcji przynależności wszystkich elementów przestrzeni rozważań U są zerowe, co zapisuje się jako (3.8):

$$A = \emptyset \Leftrightarrow \underset{u \in U}{\forall} \mu_A(u) = 0 \tag{3.8}$$

4. α-Przekrój zbioru rozmytego.

α-Przekrój (zbiór α-poziomy) zbioru rozmytego A w U jest to zbiór nierozmyty, oznaczany jako A_α, o następujących właściwościach [31]:

$$\underset{\alpha\in(0,1]}{\forall} A_\alpha = \{u \in U : \mu_A(u) \geq \alpha\} \tag{3.9}$$

α-Przekroje pełnią istotną rolę aplikacyjną. Szczególnie istotne (dla teorii i praktyki stosowania zbiorów rozmytych) jest tzw. twierdzenie o reprezentacji [31, 88, 133], które głosi, że każdy zbiór rozmyty A w U można przedstawić jako:

$$A = \sum_{\alpha\in(0,1]} \alpha A_\alpha \tag{3.10}$$

gdzie:

A_α – α-przekrój zbioru rozmytego A, zdefiniowany przez (3.9);

αA_α – zbiór rozmyty, którego stopnie przynależności mają postać (3.11):

$$\mu_{\alpha A_\alpha}(u) = \begin{cases} \alpha & \text{dla } u \in A_\alpha \\ 0 & \text{w przeciwnym przypadku} \end{cases} \tag{3.11}$$

przy tym operator $\sum$ ma charakter mnogościowy.

5. Liczność zbioru rozmytego

Pojęcie „liczność zbioru" lub „moc zbioru" odniesione do zbioru nierozmytego oznacza liczbę elementów tego zbioru. Przeniesienie tego pojęcia do dziedziny zbiorów rozmytych spowodowało zmianę jego interpretacji. Powstało wiele definicji liczności (rozmytej i nierozmytej) zbioru rozmytego [154], jednakże zapewne najczęściej stosowana jest najprostsza z nich, określona równaniem (3.12) [219], opisująca liczność nierozmytą:

$$\sum \text{Count}(A) = \sum_{i=1}^{k} \mu_A(u_i) \tag{3.12}$$

gdzie:

k – liczba singletonów zbioru rozmytego $A = {\mu_A(u_1)}/{u_1} + \cdots + {\mu_A(u_k)}/{u_k}$.

Jak wynika z (3.12), liczność nierozmyta zbioru rozmytego A określa w pewnym sensie uogólniony stopień przynależności całego zbioru rozmytego.

6. Zbiór rozmyty normalny.

Zbiór rozmyty A w U jest normalny, jeśli jego funkcja przynależności przyjmuje wartość 1 przynajmniej dla jednego elementu $u \in U$, czyli wtedy, gdy:

$$\max_{u\in U} \mu_A(u) = 1 \tag{3.13}$$

7. Wysokość zbioru rozmytego.

Wysokością zbioru rozmytego A w U jest najwyższa wartość funkcji przynależności spośród wszystkich elementów $u \in U$. Wyznacza się ją zgodnie z (3.14):

$$\mathrm{h}(A) = \sup\{\mu_A(u) \mid u \in U\} \tag{3.14}$$

8. Równość zbiorów rozmytych.

Zasadniczo, dwa zbiory rozmyte A i B określone w tej samej przestrzeni rozważań U są równe, jeśli:

$$\underset{u \in U}{\forall}\ \mu_A(u) = \mu_B(u), \tag{3.15}$$

co zapisuje się jako:

$$A = B \tag{3.16}$$

Definicja (3.15) jest słuszna zarówno dla zbiorów rozmytych, jak i nierozmytych, jednakże dla uwzględnienia nieprecyzyjności samej materii, wprowadzono także innego rodzaju „rozmyte" określenia równości zbiorów rozmytych [11, 88]. Ostrą definicję (3.16) zastąpiono pojęciem „równości w stopniu $e(A,B)$", która może przyjmować wartości z zakresu $[0,1]$. Taką równość zapisuje się jako:

$$A = {}_eB \tag{3.17}$$

Wartość $e(A,B)$ można wyznaczać na wiele sposobów [11], np. według (3.20). Dla uproszczenia zapisu opisano dodatkowo dwie charakterystyczne sytuacje (3.18) i (3.19):

$$A \neq B \text{ w sensie (3.15) oraz } T = \{u \in U : \mu_A(u) \neq \mu_B(u)\} \tag{3.18}$$

$$A \neq B \text{ w sensie (3.15) oraz } \underset{u \in U}{\exists}(\mu_A(u) = 0 \wedge \mu_{AB}(u) \neq 0) \vee (\mu_A(u) \neq 0 \wedge \mu_{AB}(u) = 0) \tag{3.19}$$

Po uwzględnieniu (3.18) i (3.19) można sformułować jedną z definicji równości w stopniu $e(A,B)$:

$$e(A,B) = \begin{cases} 1 & \text{gdy spełniony jest warunek (3.15)} \\ \min\limits_{u \in T}[\mu_A(u) \wedge \mu_B(u)] & \text{gdy zachodzi sytuacja (3.18)} \\ 0 & \text{gdy zachodzi sytuacja (3.19)} \end{cases} \tag{3.20}$$

9. Zawieranie się zbiorów rozmytych.
Zbiór rozmyty A określony w U jest zawarty w zbiorze rozmytym B określonym w U, gdy spełniony jest warunek (3.21):

$$\underset{u\in U}{\forall}\, \mu_A(u) \leq \mu_B(u), \tag{3.21}$$

co zapisuje się jako:

$$A \subseteq B \tag{3.22}$$

Podobnie jak w przypadku równości zbiorów rozmytych, oprócz definicji (3.21), opracowano wiele dodatkowych określeń [11], w których zamiast ostrego określania istnienia bądź nieistnienia zależności wprowadzono zawieranie się częściowe, determinowane stopniem zawierania się (3.23):

$$c(A,B) \in [0,1], \tag{3.23}$$

którego wartość wyznacza się w oparciu o zasady analogiczne do (3.18)–(3.20), a „zawieranie się w stopniu c" może być zapisane jako (3.24):

$$A \subseteq {}_cB \tag{3.24}$$

10. Odległość pomiędzy dwoma zbiorami rozmytymi.
Podobnie jak większość pojęć dotyczących teorii zbiorów rozmytych, odległość pomiędzy dwoma takimi zbiorami może być określana przy użyciu różnych miar, z których najistotniejsze wydają się być miary znormalizowane [88]. Do najczęściej używanych można zaliczyć znormalizowaną odległość kwadratową (euklidesową) (3.25) oraz znormalizowaną odległość liniową (Hamminga) (3.26).

$$q(A,B) = \sqrt{\frac{1}{k}\sum_{i=1}^{k}[\mu_A(u_i) - \mu_B(u_i)]^2} \tag{3.25}$$

$$l(A,B) = \frac{1}{k}\sum_{i=1}^{k}|\mu_A(u_i) - \mu_B(u_i)| \tag{3.26}$$

gdzie:
A, B – zbiory rozmyte określone w $U = \{u_1, \ldots, u_k\}$;
n – liczba singletonów rozmytych, wspólna dla zbiorów rozmytych A i B.

11. Funkcje charakterystyczne zbioru rozmytego.
Funkcją charakterystyczną zbioru rozmytego jest funkcja przynależności, której ogólną postać opisuje (3.3). Funkcja ta może w zasadzie mieć dowolny kształt,

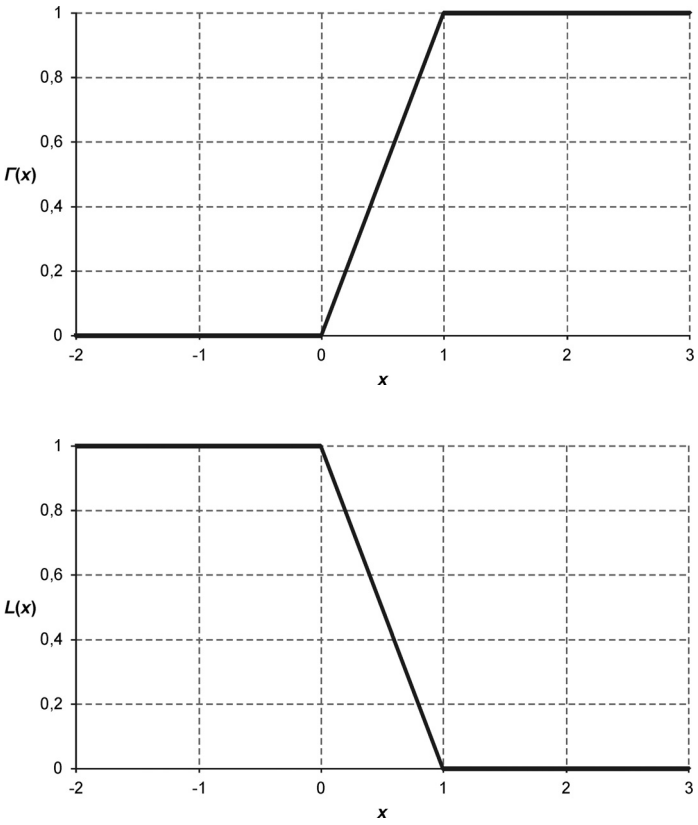
jednakże w praktycznych zastosowaniach spotyka się kilka głównych klas takich funkcji [121, 158]:

a) funkcje klasy Γ i przeciwnej do niej klasy L:

$$\Gamma_{a,b}(x) = \begin{cases} 0 & \text{dla} \quad x \leq a \\ \dfrac{x-a}{b-a} & \text{dla} \quad a < x \leq b \\ 1 & \text{dla} \quad x > b \end{cases} \tag{3.27}$$

$$L_{a,b}(x) = \begin{cases} 1 & \text{dla} \quad x \leq a \\ \dfrac{b-x}{b-a} & \text{dla} \quad a < x \leq b \\ 0 & \text{dla} \quad x > b \end{cases} \tag{3.28}$$

Graficzną reprezentację funkcji z (3.27) i (3.28) przedstawiono na rysunku 3.3.



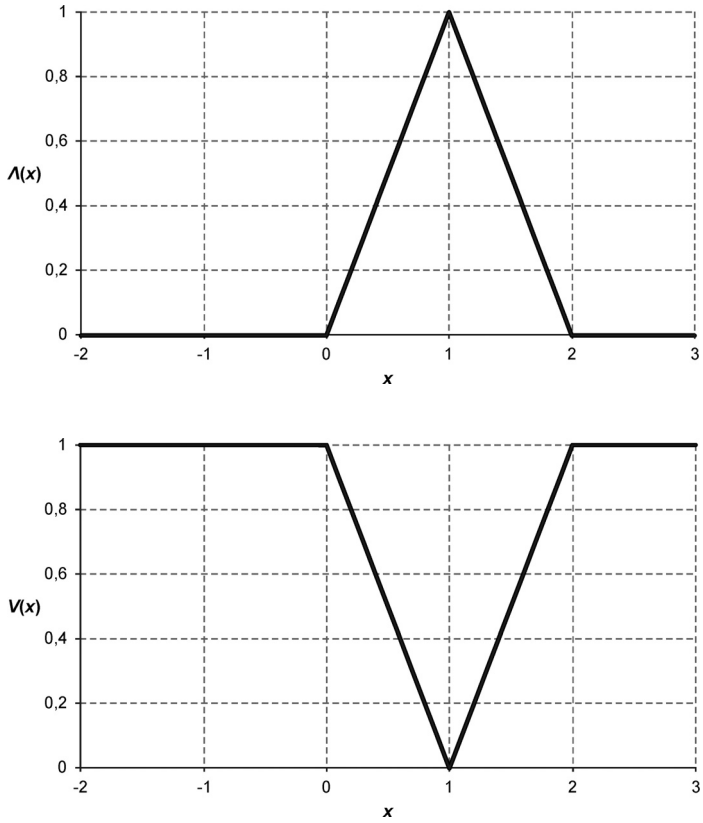
Rys. 3.3. *Reprezentacja graficzna funkcji klas: Γ i L; $a = 0$, $b = 1$*

b) funkcje klasy Λ i przeciwnej do niej klasy V:

$$\Lambda_{a,b,c}(x) = \begin{cases} 0 & \text{dla} \quad x \leq a \text{ lub } x \geq c \\ \dfrac{x-a}{b-a} & \text{dla} \quad a < x \leq b \\ \dfrac{c-x}{c-b} & \text{dla} \quad b < x < c \end{cases} \tag{3.29}$$

$$V_{a,b,c}(x) = \begin{cases} 1 & \text{dla} \quad x \leq a \text{ lub } x \geq c \\ \dfrac{b-x}{b-a} & \text{dla} \quad a < x \leq b \\ \dfrac{x-b}{c-b} & \text{dla} \quad b < x < c \end{cases} \tag{3.30}$$

Graficzną reprezentację funkcji z (3.29) i (3.30) przedstawiono na rysunku 3.4.



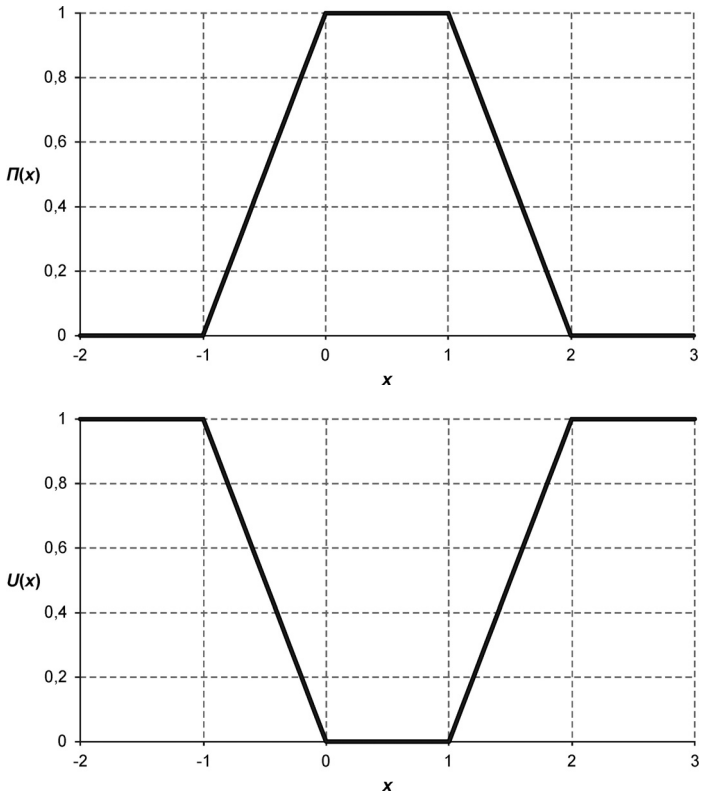
Rys. 3.4. *Reprezentacja graficzna funkcji klas: Λ i V; $a = 0$, $b = 1$, $c = 2$*

c) funkcje klasy Π i przeciwnej do niej klasy U:

$$\Pi_{a,b,c,d}(x) = \begin{cases} 0 & \text{dla} \quad x \leq a \text{ lub } x \geq d \\ \dfrac{x-a}{b-a} & \text{dla} \quad a < x \leq b \\ 1 & \text{dla} \quad b < x \leq c \\ \dfrac{d-x}{d-c} & \text{dla} \quad c < x < d \end{cases} \tag{3.31}$$

$$U_{a,b,c,d}(x) = \begin{cases} 1 & \text{dla} \quad x \leq a \text{ lub } x \geq d \\ \dfrac{b-x}{b-a} & \text{dla} \quad a < x \leq b \\ 0 & \text{dla} \quad b < x \leq c \\ \dfrac{x-c}{d-c} & \text{dla} \quad c < x < d \end{cases} \tag{3.32}$$

Graficzną reprezentację funkcji z (3.31) i (3.32) przedstawiono na rysunku 3.5.



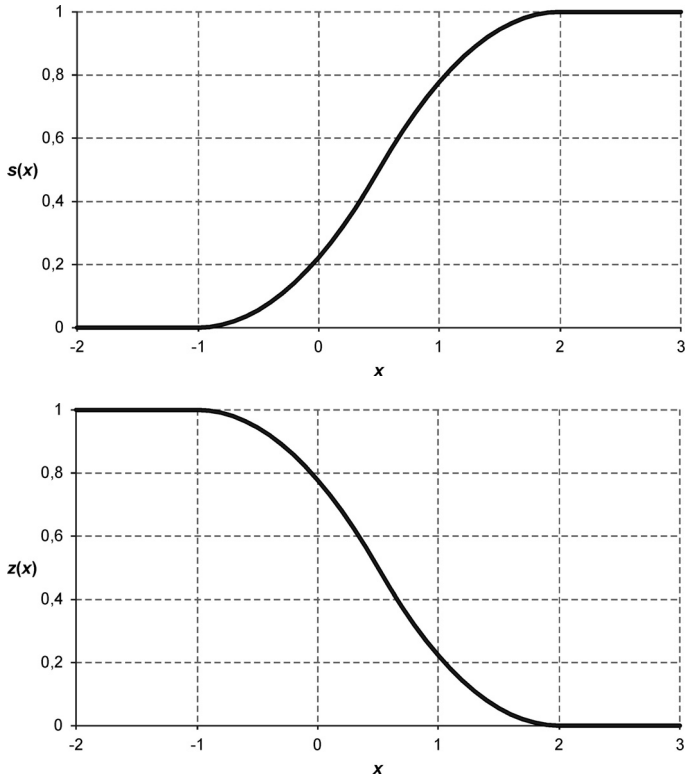
Rys. 3.5 *Reprezentacja graficzna funkcji klas: Π i U;* $a = -1$, $b = 0$, $c = 1$, $d = 2$

d) funkcje klasy *s* i przeciwnej do niej klasy *z*:

$$s_{a,b}(x) = \begin{cases} 0 & \text{dla} \quad x \le a \\ 2 \cdot \left(\frac{x-a}{b-a}\right)^2 & \text{dla} \quad a < x \le \frac{a+b}{2} \\ 1 - 2 \cdot \left(\frac{x-b}{b-a}\right)^2 & \text{dla} \quad \frac{a+b}{2} < x < b \\ 1 & \text{dla} \quad x \ge b \end{cases} \tag{3.33}$$

$$z_{a,b}(x) = \begin{cases} 1 & \text{dla} \quad x \le a \\ 1 - 2 \cdot \left(\frac{x-a}{b-a}\right)^2 & \text{dla} \quad a < x \le \frac{a+b}{2} \\ 2 \cdot \left(\frac{x-b}{b-a}\right)^2 & \text{dla} \quad \frac{a+b}{2} < x < b \\ 0 & \text{dla} \quad x \ge b \end{cases} \tag{3.34}$$

Graficzną reprezentację funkcji z (3.33) i (3.34) przedstawiono na rysunku 3.6.



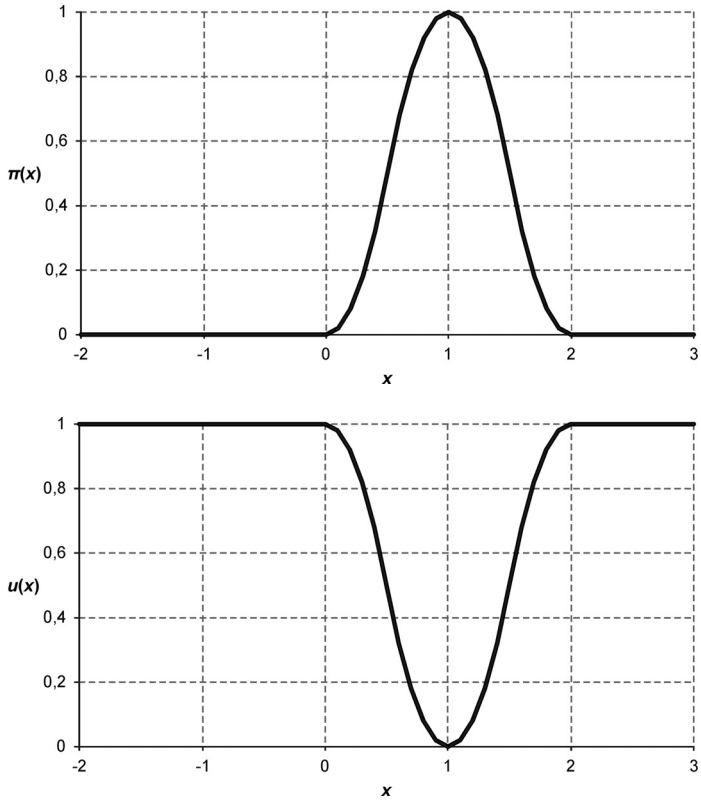
Rys. 3.6. *Reprezentacja graficzna funkcji klas: s i z;* $a = -1$, $b = 2$

e) funkcje klasy π i przeciwnej do niej klasy u:

$$\pi_{a,b}(x) = \begin{cases} s_{b-a,b} & \text{dla} \quad x < b \\ 1 - s_{b,b+a} & \text{dla} \quad x \geq b \end{cases} \tag{3.35}$$

$$u_{a,b}(x) = \begin{cases} 1 - s_{b-a,b} & \text{dla} \quad x < b \\ s_{b,b+a} & \text{dla} \quad x \geq b \end{cases} \tag{3.36}$$

Graficzną reprezentację funkcji z (3.35) i (3.36) przedstawiono na rysunku 3.7.



Rys. 3.7. *Reprezentacja graficzna funkcji klas: π i u; $a = 1$, $b = 1$*

Podane wyżej (3.27) i (3.36) ogólne postaci funkcji przynależności zbiorów rozmytych nie wyczerpują wszystkich możliwości. Podobne kształty można też uzyskać przy użyciu innych zależności funkcyjnych. Dodatkowo, można tworzyć bardziej złożone funkcje przynależności, używając wyżej wymienionych ogólnych prawidłowości.

Na rysunkach 3.3–3.7 przedstawiono pewien sposób prezentacji funkcji przynależności – wykres ciągły. Należy przy tym pamiętać, że możliwe są też inne sposo-

by prezentacji [151], takie jak: wykres dyskretny, równanie matematyczne, tabela, wektor przynależności. Spotyka się też mniej zmatematyzowane podejścia opierające się na wynikach ankiet (po obróbce statystycznej) bądź bezpośrednie definiowanie przez eksperta każdego singletonu rozmytego.

3.1.3. Podstawowe operacje na zbiorach rozmytych

Poniżej zostaną przedstawione wybrane, najczęściej wykonywane operacje na zbiorach rozmytych. Wyboru dokonano, biorąc od uwagę ich przydatność z punktu widzenia powszechnie stosowanych algorytmów logiki rozmytej. W klasycznym podejściu do działań na zbiorach rozmytych wykorzystuje się operatory mnogościowe, tym niemniej można również wykonywać operacje o charakterze algebraicznym, logicznym lub drastycznym [112, 121].

3.1.3.1. Operacje o charakterze mnogościowym

1. Dopełnienie zbioru rozmytego.

Dopełnieniem (negacją) zbioru rozmytego A określonego w U jest zbiór $\neg A$ o funkcji przynależności zdefiniowanej jako:

$$\underset{u\in U}{\forall}\ \mu_{\neg A}(u) = 1 - \mu_A(u) \tag{3.37}$$

Dopełnienie jest odpowiednikiem logicznej negacji. Graficzną reprezentację dopełnienia zbioru rozmytego przedstawia rysunek 3.8a.

2. Suma dwóch zbiorów rozmytych.

Sumą dwóch zbiorów rozmytych A i B określonych w U jest zbiór $A + B$ o funkcji przynależności zdefiniowanej jako:

$$\underset{u\in U}{\forall}\ \mu_{A+B}(u) = \mu_A(u) \vee \mu_B(u) \tag{3.38}$$

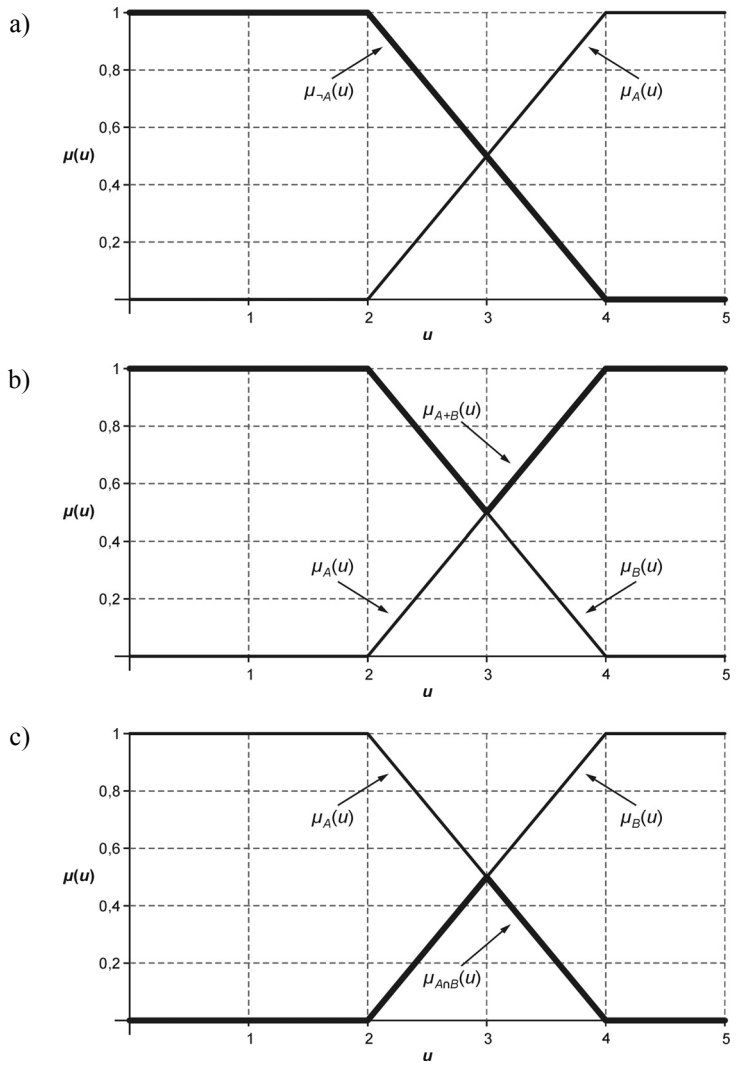
Suma jest odpowiednikiem logicznej alternatywy, przy czym operator $\vee$ oznacza operację: $a \vee b = \max(a, b)$. Graficzną reprezentację sumy dwóch zbiorów rozmytych przedstawia rysunek 3.8b.

3. Przecięcie dwóch zbiorów rozmytych.

Przecięciem (iloczynem) dwóch zbiorów rozmytych A i B określonych w U jest zbiór $A \cap B$ o funkcji przynależności zdefiniowanej jako:

$$\underset{u\in U}{\forall}\ \mu_{A\cap B}(u) = \mu_A(u) \wedge \mu_B(u) \tag{3.39}$$

Przecięcie jest odpowiednikiem logicznej koniunkcji, przy czym operator $\wedge$ oznacza operację: $a \wedge b = \min(a, b)$. Graficzną reprezentację przecięcia dwóch zbiorów rozmytych przedstawia rysunek 3.8c.



Rys. 3.8. *Graficzne reprezentacje wybranych operacji mnogościowych na zbiorach rozmytych. Linią pogrubioną wyróżniono wynik operacji: a) dopełnienie, b) suma, c) przecięcie*

3.1.3.2. Normy trójkątne

Definicje (3.37)–(3.39) mają charakter praktyczny i są powszechnie stosowane z uwagi na ich łatwy do zrozumienia, intuicyjny charakter. Można je również wyrazić bardziej formalnie przy użyciu tzw. norm trójkątnych, znanych pod nazwami: t-norma oraz s-norma (tę ostatnią nazywa się też niekiedy t-konormą). Istnieje wiele odmian t-normy i s-normy, różniących się od siebie poszczególnymi właściwościami [25, 51, 88, 121, 123, 200, 208], poszczególni badacze wprowadzają też, z dobrym skutkiem, własne modyfikacje (jak np. elementy parametryzacji [161, 162]), wszystkie jednak służą do określania własności zbiorów i operatorów rozmytych.

1. *t*-norma:
t-norma jest dwuargumentową funkcją opisaną przez (3.40):

$$t : [0,1] \times [0,1] \longrightarrow [0,1], \tag{3.40}$$

spełniającą następujące warunki:
– monotoniczności: $a \leq b \Longrightarrow t(a,c) \leq t(b,c)$,
– przemienności: $t(a,b) = t(b,a)$,
– łączności: $t(t(a,b),c) = t(a,t(b,c))$,
– brzegowe: $\forall_{a\in[0,1]}\, t(a,1) = a$; $\forall_{a\in[0,1]}\, t(a,0) = 0$.

Najczęściej spotyka się dwa rodzaje zapisu t-normy: $t(a,b)$ lub $a\ t\ b$ [88]. Przykładami zastosowania t-normy mogą być: minimum (3.41) i iloczyn algebraiczny (3.42):

$$t(a,b) = a \wedge b = \min(a,b) \tag{3.41}$$

$$t(a,b) = a \cdot b \tag{3.42}$$

2. *s*-norma:
s-norma jest dwuargumentową funkcją opisaną przez (3.43):

$$s : [0,1] \times [0,1] \longrightarrow [0,1], \tag{3.43}$$

spełniającą następujące warunki:
– monotoniczności: $a \leq b \Longrightarrow s(a,c) \leq s(b,c)$,
– przemienności: $s(a,b) = s(b,a)$,
– łączności: $s(s(a,b),c) = s(a,s(b,c))$,
– brzegowe: $\forall_{a\in[0,1]}\, s(a,0) = a$; $\forall_{a\in[0,1]}\, s(a,1) = 0$.

Podobnie jak dla t-normy, s-normę zapisuje się jako: $s(a,b)$ lub $a\ s\ b$. Niekiedy s-normę określa się mianem t-konormy. Przykładami zastosowania s-normy mogą być: maximum (3.44) i iloczyn probabilistyczny (3.45):

$$s(a,b) = a \vee b = \max(a,b) \tag{3.44}$$

$$s(a,b) = a + b - ab \tag{3.45}$$

3.1.4. Inne rodzaje operacji na zbiorach rozmytych

Spośród wielu innych, niemnogościowych operacji na zbiorach rozmytych na szczególną uwagę zasługują trzy grupy, ściśle powiązane z t-normą i s-normą. Są

to: operacje algebraiczne, operacje logiczne (zwane niekiedy ograniczonymi) oraz operacje drastyczne [121], które zostaną pokazane poniżej na najbardziej charakterystycznych przykładach.

1. Operacje algebraiczne na zbiorach rozmytych

– iloczyn algebraiczny (operator t-normy):

$$\mu_{A\cap B}(u) = \mu_A(u) \cdot \mu_B(u) \tag{3.46}$$

– suma algebraiczna (operator s-normy):

$$\mu_{A\cup B}(u) = \mu_A(u) + \mu_B(u) - \mu_A(u) \cdot \mu_B(u) \tag{3.47}$$

2. Operacje logiczne na zbiorach rozmytych

– iloczyn logiczny (operator t-normy):

$$\mu_{A\cap B}(u) = \max\{0, \mu_A(u) + \mu_B(u) - 1\} \tag{3.48}$$

– suma logiczna (operator s-normy):

$$\mu_{A\cup B}(u) = \min\{\mu_A(u) + \mu_B(u), 1\} \tag{3.49}$$

3. Operacje drastyczne na zbiorach rozmytych

– iloczyn drastyczny (operator t-normy):

$$\mu_{A\cap B}(u) = \begin{cases} \min\{\mu_A(u), \mu_B(u)\} & \text{dla } \max\{\mu_A(u), \mu_B(u)\} = 1 \\ 0 & \text{dla pozostałych przypadków} \end{cases} \tag{3.50}$$

– suma drastyczna (operator s-normy):

$$\mu_{A\cup B}(u) = \begin{cases} \max\{\mu_A(u), \mu_B(u)\} & \text{dla } \min\{\mu_A(u), \mu_B(u)\} = 0 \\ 1 & \text{dla pozostałych przypadków} \end{cases} \tag{3.51}$$

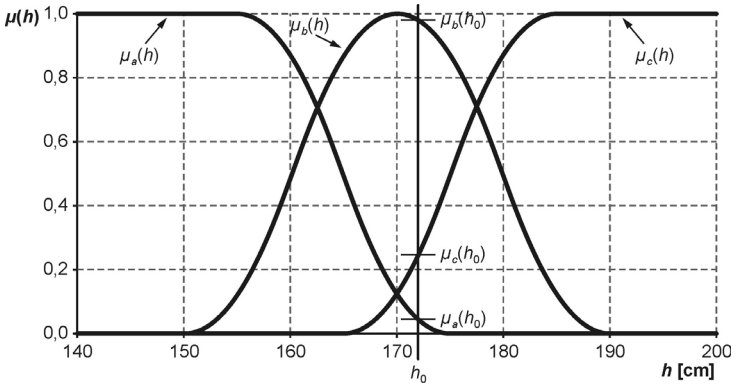
3.1.5. Zmienne lingwistyczne i rozmyte zdania warunkowe

Zmienna lingwistyczna (wprowadzona w pracy [223]) jest zmienną, której wartości nie mają charakteru liczbowego, lecz opisowy. Jej przykładem może być zmienna „wzrost", przyjmująca wartości lingwistyczne: „niski", „wysoki" itp. Wartości takie mogą być utożsamiane z pewnymi zbiorami rozmytymi, można też w oparciu o nie tworzyć bardziej złożone wyrażenia typu: „nie bardzo wysoki" przy użyciu negacji, spójników („i", „lub" itp.) oraz specjalnych modyfikatorów („bardzo", „nieco", „prawie", „około", „mniej", „więcej" itp.) [86]. Takie złożone

określenia tworzy się przy użyciu odpowiednich reguł semantycznych, a następnie wykorzystuje do reprezentacji relacji rozmytych.

Pojęcie zmiennej lingwistycznej można wyjaśnić graficznie za pomocą jej reprezentacji graficznej (rys. 3.9).

Rysunek 3.9 przedstawia zależność wartości wielkości lingwistycznej „wzrost", określonej przez trzy jakościowe wartości lingwistyczne („niski", „średni" i „wysoki") będące zbiorami rozmytymi różnych klas (z, π, s), od ilościowej wartości zmierzonej h. Fakt przyporządkowania konkretnej (podanej w centymetrach) wartości wzrostu h_0 do określonej wartości lingwistycznej zależy od przebiegów funkcji przynależności tych wartości lingwistycznych ($\mu_a(h)$, ($\mu_b(h)$, ($\mu_c(h)$) i ich usytuowania względem siebie. Dana wartość h_0 będzie powiązana z tą wartością lingwistyczną, której funkcja przynależności w punkcie h_0 osiągnie najwyższą wartość (na rys. 3.9 wzrost h_0 jest równoważny lingwistycznej wartości „średni"). W przypadku jednakowych wartości funkcji przynależności kilku wartości lingwistycznych, przyporządkowania do jednej z nich można dokonać na wiele sposobów (np. losowo).



Rys. 3.9. *Przykładowa reprezentacja graficzna wartości zmiennej lingwistycznej „wzrost": μ_a, μ_b, μ_c – funkcje przynależności trzech rozmytych wartości, odpowiednio: a – „niski", b – „średni", c – „wysoki"*

Pokazaną na rysunku 3.9 liczbową wartość h_0 można zapisać w bardziej formalny sposób – jako pewien zbiór singletonów odnoszących się do poszczególnych wartości lingwistycznych:

$$\mu_h(h_0) = \frac{\mu_a(h_0)}{a} + \frac{\mu_b(h_0)}{b} + \frac{\mu_c(h_0)}{c} \tag{3.52}$$

gdzie:
a, b, c – kolejne wartości lingwistyczne w formie zbiorów rozmytych ($a \lesssim b \lesssim c$).

Sposób zapisu wartości wielkości lingwistycznej przedstawiony w (3.52) jest nieprzydatny w klasycznych metodach wnioskowania rozmytego, wykorzystują-

cych reguły typu pokazanego poniżej w (3.53), jednakże okazuje się on bardzo wygodny przy tworzeniu formalnego opisu matematycznego działania modeli relacyjnych rozmytych map kognitywnych (opisanych w rozdziale 4). Zamieszczenie informacji o tej metodzie wynika tutaj z kontekstu, który uczyni późniejsze rozważania na ten temat bardziej przejrzystymi.

Relacje pomiędzy zmiennymi lingwistycznymi wyraża się, formułując rozmyte zdania warunkowe. Przykładem takiego zdania jest (3.53):

$$\text{IF } L = A \text{ THEN } K = B \tag{3.53}$$

gdzie:

L – zmienna lingwistyczna, której wartość jest zbiorem rozmytym A określonym w U;

K – zmienna lingwistyczna, której wartość jest zbiorem rozmytym B określonym w V.

Zamiast (3.53) można użyć równoważnego (skróconego) zapisu według (3.54):

$$\text{IF } A \text{ THEN } B \tag{3.54}$$

Zapis (3.54) jest równoważny definicji następującej relacji rozmytej (3.55):

$$\text{IF } A \text{ THEN } R = A \times B \tag{3.55}$$

Rozmyte zdanie warunkowe (3.55) jest często wykorzystywane w sterowaniu rozmytym i nosi nazwę implikacji Mamdaniego [121, 124, 152]. Definicja występującego w tym zdaniu iloczynu kartezjańskiego dwóch zbiorów rozmytych $A \times B$ została opisana w dalszej części pracy – równaniami (3.138) i (3.139).

3.1.6. Wyostrzanie zbiorów rozmytych

W większości przypadków wynikiem działania rozmytego algorytmu obliczeniowego jest zbiór rozmyty. Z punktu widzenia praktycznych zastosowań (np. w regulatorach lub sterownikach rozmytych) takie rozwiązanie jest nieprzydatne, ponieważ wymaga wprowadzenia ostrej wartości sterującej, która byłaby adekwatną reprezentacją otrzymanego wcześniej rozmytego wyniku obliczeń. Wartość taką uzyskuje się w procesie wyostrzania (defuzyfikacji) zbioru rozmytego.

Istnieje wiele metod wyostrzania [209] zbiorów rozmytych, jednakże najczęściej używa się zaledwie kilku spośród nich, wymienionych poniżej.

1. Metoda pierwszego maksimum.

W metodzie tej przegląda się kolejno elementy przestrzeni rozważań U zbioru rozmytego A w poszukiwaniu maksymalnej wartości funkcji przynależności $\mu_A(u)$:

$$\mu_{A,max}(u) = \max_{u_i}[\mu_A(u)] \tag{3.56}$$

Wynikiem jest ten punkt u_i, dla którego funkcja przynależności po raz pierwszy osiąga swoją maksymalną wartość $\mu_{A,max}(u)$.

2. Metoda ostatniego maksimum

W tej metodzie, podobnie jak w metodzie pierwszego maksimum, poszukuje się maksymalnej wartości funkcji przynależności $\mu_A(u)$ – zgodnie z (3.56), ale wynikiem jest ten punkt u_i, dla którego funkcja przynależności po raz ostatni osiąga swoją maksymalną wartość $\mu_{A,max}(u)$.

3. Metoda środka maksimów

Metoda środka maksimów polega na znalezieniu pierwszego i ostatniego maksimum (jak podano powyżej), po czym wyznaczeniu ich średniej arytmetycznej zgodnie z (3.57):

$$\bar{A} = \frac{\bar{A}_1 + \bar{A}_m}{2} \tag{3.57}$$

gdzie:
$\bar{A}_1$, $\bar{A}_m$ – pierwsze i ostatnie maksimum.

4. Metoda średniej ważonej

Metoda ta znana jest również pod innymi nazwami: środka powierzchni, środka ciężkości. Wartość wyostrzoną zbioru rozmytego A określonego w U wyznacza się według zależności (3.58):

$$\bar{A} = \frac{\sum_{i=1}^{k} u_i \cdot \mu_A(u_i)}{\sum_{i=1}^{k} \mu_A(u_i)} \tag{3.58}$$

gdzie:
$\bar{A}$ – wartość wyostrzona (centrum) zbioru rozmytego A;
k – liczba singletonów rozmytych zbioru rozmytego A.

5. Metoda maksymalnej przynależności

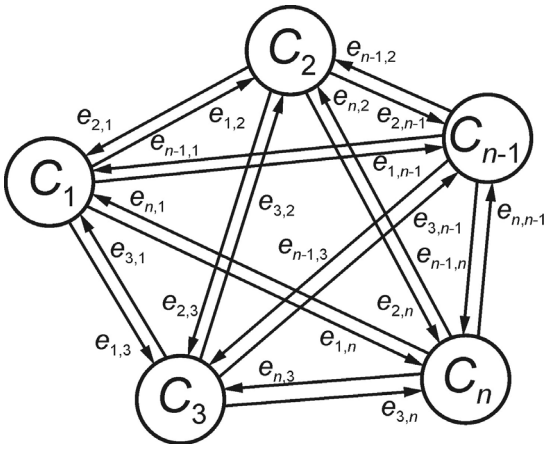
Metoda ta może być efektywnie stosowana jedynie w odniesieniu do tych zbiorów rozmytych, których jądro (3.7) zawiera tylko jeden element. Jest zbliżona do metody pierwszego maksimum, przy czym wynikiem jest po prostu ten punkt u_i, dla którego funkcja przynależności osiąga swoją maksymalną wartość (3.59):

$$\bar{A} = u_i : \mu_A(u_i) = \max_{u_i}[\mu_A(u)] \tag{3.59}$$

3.2. ROZMYTE MAPY KOGNITYWNE W UJĘCIU KLASYCZNYM

Pojęcie i podstawowa struktura rozmytej mapy kognitywnej zostały wprowadzone jako odpowiedź na potrzebę znalezienia formalnej reprezentacji zjawisk, w któ-

rych występuje wnioskowanie przyczynowo-skutkowe. Ogólnie rzecz biorąc, rozmyta mapa kognitywna [90, 225] jest grafem skierowanym, w którego węzłach umieszczono czynniki (ang. *concepts*), takie jak, np. zasady działania, zdarzenia, a gałęzie odzwierciedlają związki przyczynowo-skutkowe. Struktura taka, podobnie jak opisana w rozdziale 2 relacyjna mapa kognitywna, modeluje przyczynowo-skutkowe związki pomiędzy czynnikami, ale zamiast opisu z użyciem liczb rzeczywistych, używa opisu rozmytego. Zwykle jest reprezentowana graficznie w sposób, jaki został przedstawiony na rysunku 3.10. W literaturze często używa się akronimu angloję-zycznej nazwy tzn. FCM, dlatego też w dalszych rozważaniach akronim ten będzie używany wymiennie z określeniem rozmyta mapa kognitywna.

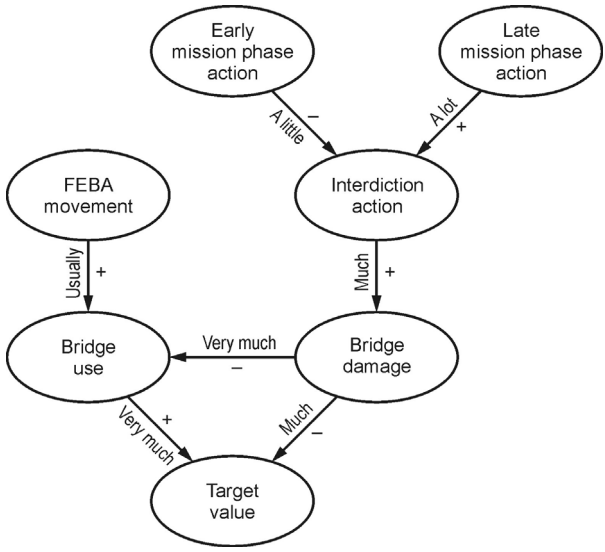


Rys. 3.10. *Ogólna reprezentacja graficzna FCM.* C_1, ..., C_n *– czynniki (concepts),* $e_{i,j}$ *– reprezentacja związku przyczynowo-skutkowego pomiędzy czynnikami:* i*-tym i* j*-tym*

W swojej bazowej postaci FCM przechowuje informacje określające, po prostu, czy wzrost wartości danego czynnika powoduje wzrost, czy też spadek wartości innych czynników. W formach bardziej zaawansowanych towarzyszy temu informacja na temat intensywności tego oddziaływania. Znaki oraz intensywności powiązań są „kodowane" w opisach gałęzi (ang. *edges*) grafu reprezentującego FCM ($e_{i,j}$ na rys. 3.10).

FCM w obrazowy sposób modelują świat jako kolekcję klas i związków przyczynowych pomiędzy nimi [90, 201]. Reprezentowanie złożonych procesów przy użyciu stosunkowo prostych grafów (jak na rys. 3.10) może wydawać się mało skuteczne, okazuje się jednak bardzo dobrą metodą modelowania w sytuacji nieprecyzyjności opisu niektórych zjawisk. Uzasadnieniem określenia **rozmyta** mapa kognitywna jest właśnie nieprecyzyjność opisu wartości czynników uwzględniona przez często stosowany zapis tych wartości w postaci zbiorów rozmytych. Każdy czynnik reprezentuje pewną wielkość lingwistyczną, która z kolei może przybierać jedną z przypisanych do niej wartości lingwistycznych (zgodnie z opisem z podrozdziału 3.1.5), natomiast powiązania przyczynowe oddają po pierwsze, kierunek oddziaływań (tzn.

określają czy wzrost wartości danego czynnika powoduje wzrost, czy też spadek wartości innego czynnika), a po drugie – natężenie oddziaływania (co może być wyrażane w postaci rozmytych modyfikatorów typu: „trochę", „bardzo" itp.). Taki zamysł przyświecał idei tworzenia FCM. W swojej podstawowej postaci graf FCM zawierał węzły (reprezentujące zjawiska bądź wydarzenia) i łuki odzwierciedlające klasyczne operatory zbiorów rozmytych. Na rysunku 3.11 pokazano jeden z pierwszych przykładów opisu pewnej strategii przy użyciu FCM sporządzony przez B. Kosko [108] (zachowano oryginalne opisy).



Rys. 3.11. *Przykład klasycznej FCM [108] przedstawiającej cele strategiczne, możliwe taktyki oraz fakty pola walki powiązane poprzez rozmyte związki przyczynowe*

Na potrzeby formalnej obsługi konstrukcji FCM, takich jak te przedstawione na rysunku 3.11, opracowano zasady wnioskowania przyczynowego, które, choć w pewnym sensie odpowiada implikacji logicznej, jest jednak operacją bardziej od niej złożoną. Dla przykładu [108]: gdyby wyrażenie „*A* powoduje *B*" było równoważne wyrażeniu „*A* implikuje *B*", to, na zasadzie kontrapozycji, wyrażenie „*A* powoduje *B*" można by było stosować zamiennie z „not-*B* powoduje not-*A*", co jednak nie jest prawdą. Chociaż palenie powoduje raka płuc, to brak raka płuc nie powoduje niepalenia. Bliższe prawdy byłoby stwierdzenie, że niepalenie powoduje brak raka płuc. Na tego typu rozważaniach oparto początkowo cały aparat wnioskowania przyczynowego. Powstały dwa główne typy powiązań [108]:

a) korelacja dodatnia:

$$\begin{aligned} &\text{If „}A\text{ powoduje }B\text{"} \\ &\text{then} \quad \text{„zwiększanie }A\text{ powoduje zwiększanie }B\text{" oraz} \\ &\qquad\quad \text{„zmniejszanie }A\text{ powoduje zmniejszanie }B\text{"} \end{aligned} \tag{3.60}$$

b) korelacja ujemna:

$$\begin{aligned} &\text{If „}A\text{ przyczynowo zmniejsza }B\text{”} \\ &\text{then} \quad \text{„zwiększanie }A\text{ powoduje zmniejszanie }B\text{” oraz} \\ &\qquad\quad \text{„zmniejszanie }A\text{ powoduje zwiększanie }B\text{”} \end{aligned} \tag{3.61}$$

Taki nieimplikacyjny charakter wnioskowania przyczynowego może być reprezentowany w strukturze zbioru rozmytego. Jeśli czynniki FCM będą traktowane jako obiekty przyczynowe (ang. *Causal Objects*), to będą one mogły być reprezentowane jako podzbiory rozmyte pewnej przestrzeni rozważań, gdzie zmiana stopnia przynależności w zbiorze rozmytym oznacza zmianę czynnika. W takim układzie pojedynczy czynnik C_i może być zdefiniowany jako dysjunkcja pewnego rozmytego zbioru Q_i i powiązanego z nim zbioru $\neg Q_i$:

$$C_i = Q_i \cup \neg Q_i \tag{3.62}$$

gdzie:
$\neg Q_i$ – dopełnienie zbioru rozmytego Q_i (zgodne z (3.37)).

Biorąc pod uwagę powyższe rozważania można zdefiniować rozmytą przyczynowość na bazie teoretycznych rozmytych powiązań pomiędzy rozmytymi czynnikami.

DEFINICJA 3.1

Niech: $C_i = Q_i \cup \neg Q_i$ *oraz* $C_j = Q_j \cup \neg Q_j$. *Wtedy:*

$$\begin{aligned} &\text{„}C_i \textit{ powoduje } C_j\text{” iff } Q_i \subseteq Q_j \text{ and } \neg Q_i \subseteq \neg Q_j; \\ &\text{„}C_i \textit{ przyczynowo zmniejsza } C_j\text{” iff } Q_i \subseteq \neg Q_j \text{ and } \neg Q_i \subseteq Q_j \end{aligned} \tag{3.63}$$

gdzie:
iff – *operator będący skrótem wyrażenia* „if and only if”;
$\subseteq$ – *operator zawierania się zbiorów rozmytych (wg (3.21)).*

Po wprowadzeniu definicji 3.1 można opisywać przyczynowość ujemną i dodatnią z zastosowaniem tych samych rozmytych wielkości i powiązań. Na mocy tej definicji można też zrezygnować ze stosowania ujemnej korelacji na rzecz odpowiedniego przeformułowania wartości czynników. Jest to podejście często stosowane, ale nie zawsze wygodne. Jeśli jedna przyczyna zwiększa wartość danego czynnika, a inna ją zmniejsza, to zastąpienie wartości czynnika jej przeciwieństwem niczego nie zmieni.

W zasadzie celem wnioskowania przyczynowego w FCM jest sprawdzenie, jak pobudzenie któregoś z czynników wpłynie na wartości pozostałych czynników. To zadanie jest wykonywane z użyciem rozmytej algebry przyczynowej.

3.2.1. Rozmyta algebra przyczynowa

Rozmyta algebra przyczynowa (ang. *Fuzzy Causal Algebra*) stanowi podstawowy mechanizm wnioskowania przyczynowego. Jej opis zostanie poprzedzony wprowadzeniem kilku dodatkowych wielkości istotnych dla abstrakcyjnego opisu FCM [11, 108, 134].

DEFINICJA 3.2

Rozmyty zbiór potęgowy *zbioru rozmytego X jest zbiorem wszystkich podzbiorów rozmytych zbioru X. Oznacza się go jako*: $F(2^X)$.

DEFINICJA 3.3

Stopień zawierania się *zbioru rozmytego A w zbiorze rozmytym B* $(A, B \in F(2^X))$ *określa w jakim stopniu zbiór A należy do rozmytego zbioru potęgowego zbioru B i jest oznaczany jako*: $m_{F(2^B)}(A)$. *Można go wyznaczyć różnymi metodami. Jedną z nich jest metoda stosowana dla skończonych zbiorów rozmytych określonych na tym samym nośniku. Wtedy stopień zawierania się można wyznaczyć (na podst. [134]) według równania (3.64):*

$$m_{F(2^B)}(A) = \frac{\sum_{i=1}^{k} \min\left(\mu_A(u_i), \mu_B(u_i)\right)}{\sum_{i=1}^{k} \mu_A(u_i)} \tag{3.64}$$

gdzie:
μ_A, μ_B – *funkcje przynależności zbiorów rozmytych A i B;*
u – *nośnik.*

DEFINICJA 3.4

Klasa podzbiorów rozmytych $\mathcal{Z} \subseteq F(2^X)$ *jest* ***przestrzenią wielkości*** *zbioru* X, *jeżeli każdy zbiór rozmyty* $A \in \mathcal{Z}$ *może być przedstawiony jako*: $A = Q \cup \neg Q$ *dla jakichś* $Q, \neg Q \in F(2^X)$.

DEFINICJA 3.5

Klasa podzbiorów rozmytych $\mathcal{M} \subseteq F(2^X)$ *jest* ***przestrzenią modyfikatorów*** *zbioru* X, *jeżeli* $X \in \mathcal{M}$.

DEFINICJA 3.6

Klasa podzbiorów rozmytych $\mathcal{C} \subseteq F(2^X)$ *jest* ***przestrzenią czynników*** *zbioru* X, *jeżeli* $\mathcal{C} = \mathcal{Z} \cap \mathcal{M}$.

DEFINICJA 3.7

*Przestrzeń czynników $\mathcal{C}$ jest **przyczynowa**, jeżeli dla wszystkich $C_i, C_j \in \mathcal{C}$:*

$$\begin{cases} Q_i \cap M_i \subseteq Q_j \cap M_j \Longrightarrow \neg Q_i \cap M_i \subseteq \neg Q_j \cap M_j \\ Q_i \cap M_i \subseteq \neg Q_j \cap M_j \Longrightarrow \neg Q_i \cap M_i \subseteq Q_j \cap M_j \end{cases} \tag{3.65}$$

gdzie:
$Q_i \cup \neg Q_i, Q_j \cup \neg Q_j \in \mathcal{Z}$;
$M_i, M_j \in \mathcal{M}$.

DEFINICJA 3.8

*Rozmytą przyczynową **funkcją krawędzi** (ang. Fuzzy Causal Edge Function) pomiędzy czynnikami C_i i C_j jest funkcja $e : \mathcal{C} \times \mathcal{C} \to P$ taka, że*:

$$e(C_i, C_j) = e_{i,j} = m_{F\left(2^{C_j}\right)}(C_i) \tag{3.66}$$

gdzie:
P – częściowo uporządkowany zbiór (zakres) wartości funkcji e;
$e(C_i, C_i) \precsim p$ dla każdego $p \in P$.

Na podstawie definicji 3.1–3.8 można zaprojektować dwa proste abstrakcyjne operatory [108], służące do obliczania skumulowanego wpływu (przyczynowości) jednego czynnika na inny. Są to:

a) **współczynnik efektu pośredniego** I, zdefiniowany zależnością (3.67):

$$I_l(C_i, C_j) = \min\{e\left(C_p, C_{p+1}\right) : (p, p+1) \in (i, k_1^l, \dots, k_{n_l}^l, j)\} \tag{3.67}$$

gdzie:
m – liczba przyczynowych ścieżek pomiędzy C_i a C_j;
l – numer ścieżki, dla której obliczany jest współczynnik efektu pośredniego ($1 \le l \le m$);
$p, p+1$ – kolejne (narastające) wskaźniki ścieżki;
$k_1^l, \dots, k_{n_l}^l$ – numery czynników, przez które prowadzi l-ta ścieżka;

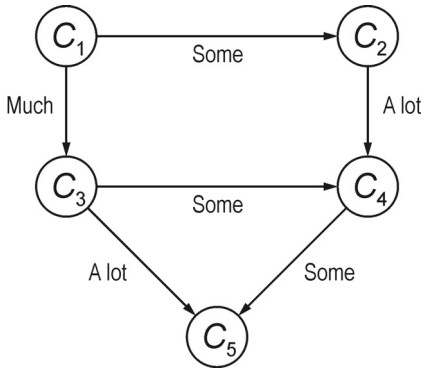
b) **współczynnik efektu łącznego** T, zdefiniowany zależnością (3.68):

$$T(C_i, C_j) = \max_{1 \le l \le m} I_l(C_i, C_j) \tag{3.68}$$

Posługując się operatorami (3.67) i (3.68), można obliczać skumulowane wzajemne przyczynowe oddziaływania pomiędzy czynnikami. Metodę ilustruje przykładowy proces wnioskowania przedstawiony przez B. Kosko [108] (zachowano oznaczenia oryginału), pokazany w przykładzie 3.1.

Przykład 3.1

Działanie pewnego nieprecyzyjnego systemu opisuje FCM przedstawiona na rysunku 3.12.



Rys. 3.12. *Przykładowa mapa kognitywna służąca ilustracji wnioskowania przyczynowego*

Dla FCM z rysunku 3.12 określono następujący zbiór wartości funkcji krawędzi e:

$$P = \{none \lesssim some \lesssim much \lesssim a\ lot\}$$

Zadanie polega na znalezieniu skumulowanego powiązania przyczynowego pomiędzy czynnikami C_1 i C_5.
W przykładowej FCM można wyróżnić trzy ścieżki przyczynowe prowadzące od C_1 do C_5: (C_1, C_3, C_5), (C_1, C_3, C_4, C_5), (C_1, C_2, C_4, C_5). Na tej podstawie oblicza się trzy współczynniki efektu pośredniego według (3.67):

$$I_1(C_1, C_5) = \min\{e_{1,3}, e_{3,5}\} = \min\{much, a\ lot\} = much$$
$$I_2(C_1, C_5) = \min\{e_{1,3}, e_{3,4}, e_{4,5}\} = \min\{much, some, some\} = some$$
$$I_3(C_1, C_5) = \min\{e_{1,2}, e_{2,4}, e_{4,5}\} = \min\{some, a\ lot, some\} = some$$

Następnie, na podstawie (3.68), można znaleźć współczynnik efektu łącznego:

$$T(C_1, C_5) = \max\{I_1(C_1, C_5), I_2(C_1, C_5), I_3(C_1, C_5)\} = \max\{much, some, some\} = much$$

Wynika stąd, że istnieje znaczne ($much$) powiązanie przyczynowo pomiędzy czynnikami C_1 i C_5.

W rozwiązaniach praktycznych używanie na szerszą skalę metody pokazanej w przykładzie 3.1 bywa kłopotliwe z uwagi na trudności w automatyzowaniu operacji na wartościach lingwistycznych. W związku z powyższym często stosowanym podejściem jest zastąpienie zbioru wartości lingwistycznych funkcji krawędzi zbiorem wartości liczbowych (często stosuje się $P = \{-1,0,1\}$) lub pewnym przedziałem liczbowym (np. $P = [0, 1]$ lub $P = [-1, 1]$). Wtedy zbiór wartości funkcji krawędzi można zastąpić odpowiadającym mu zbiorem wag powiązań pomiędzy czynnikami takich, że:

$$e_{i,j} = w_{i,j} \in \mathbb{R} \tag{3.69}$$

Takie podejście, chociaż prostsze rachunkowo, stwarza dodatkowe problemy związane z akwizycją danych ekspertowych (zwłaszcza przy kumulowaniu danych od różnych ekspertów) oraz może nadawać nadmierną rangę danym mało istotnym z punktu widzenia procesów decyzyjnych. Ponadto używanie liczbowych wag zamiast rozmytych modyfikatorów częściowo znosi uzasadnienie dla nazywania takich konstrukcji rozmytymi. Pomimo to metoda taka jest szeroko stosowana (m.in. [1, 12, 30, 37, 50, 61, 96, 100, 136, 141]), dlatego też zostanie ona bliżej zaprezentowana w kolejnych podrozdziałach.

3.2.2. Podstawowe pojęcia

Zarówno węzły, jak i gałęzie grafu FCM mogą (zależnie od potrzeb) mieć różne charaktery, z czego wynika istnienie różnych typów takich map, co prowadzi do sformułowania podstawowych definicji (część z nich zaczerpnięto z [90]).

DEFINICJA 3.9

Czynniki (węzły) FCM będące zbiorami rozmytymi nazywa się czynnikami (węzłami) rozmytymi.

DEFINICJA 3.10

FCM, w której związki przyczynowe pomiędzy czynnikami przyjmują wartości ze zbioru $\{-1, 0, 1\}$ *nosi nazwę prostej FCM (niekiedy używa się też nazwy „trójwartościowa FCM").*

DEFINICJA 3.11

FCM, w której związki przyczynowe pomiędzy czynnikami przyjmują wartości z przedziału $[-1, 1]$*, nosi nazwę złożonej FCM.*

DEFINICJA 3.12

Jeżeli gałęzie grafu FCM są reprezentowane przez $e_{i,j} \in \{-1, 0, 1\}$*, gdzie* $e_{i,j}$ *oznacza powiązanie przyczynowe pomiędzy czynnikami* C_i *i* C_j*, to gałęzie te mogą być zapisane w postaci macierzy* $\boldsymbol{E}$ *(3.70), nazywanej macierzą sąsiedztwa (ang. Adjacency Matrix) lub macierzą powiązań. Przy tym macierz* $\boldsymbol{E}$ *jest macierzą kwadratową i ma zera na głównej przekątnej.*

$$\mathbf{E} = \begin{array}{c} \\ C_1 \\ C_2 \\ \vdots \\ C_n \end{array} \begin{array}{c} \begin{array}{cccc} C_1 & C_2 & \cdots & C_n \end{array} \\ \begin{bmatrix} 0 & e_{1,2} & \cdots & e_{1,n} \\ e_{2,1} & 0 & \cdots & e_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ e_{n,1} & e_{n,2} & \cdots & 0 \end{bmatrix} \end{array} \tag{3.70}$$

DEFINICJA 3.13

Niech C_1, ..., C_n będą czynnikami FCM, a $A = (a_1, \dots, a_n)$ niech będzie wektorem takim, że $a_i \in \{0, 1\}$. A nazywa się chwilowym binarnym wektorem stanu i opisuje wartości (typu on-off) czynników w danej chwili (3.71):

$$a_i = \begin{cases} 0 & \textit{jeżeli } C_i \textit{ jest w pozycji off} \\ 1 & \textit{jeżeli } C_i \textit{ jest w pozycji on} \end{cases} \tag{3.71}$$

gdzie:
$i = 1, \dots, n$ (n – liczba czynników mapy);
stan „on" odpowiada stanowi aktywności (włączenia) czynnika, stan „off" – stanowi nieaktywności (wyłączenia).

Uwaga 3.1

W układach dynamicznych (wg definicji 3.16) wynik przejścia chwilowego binarnego wektora stanu *A* przez macierz sąsiedztwa **E** (wykonania jednego kroku przepływu sygnałów) musi być uzupełniony o operację progowania i aktualizacji, która ma na celu dostosowanie wynikowych współrzędnych wektora *A* do założonych ograniczeń. Operacja przejścia *A* przez **E** jest oznaczana jako $A\mathbf{E}$, a jej wynik (przed progowaniem) jako $(a_1', \dots, a_n')$. Jeśli wektor *A* ma mieć końcową postać $(a_1'', \dots, a_n'')$, to proces przejścia, progowania i aktualizacji oznacza się następująco:

$$A\mathbf{E} = (a_1', \dots, a_n') \longrightarrow (a_1'', \dots, a_n'') \tag{3.72}$$

DEFINICJA 3.14

Niech C_1, ..., C_n będą węzłami grafu FCM, a $\overrightarrow{C_1C_2}$, $\overrightarrow{C_2C_3}$, ..., $\overrightarrow{C_{k-1}C_k}$ – łukami tego grafu ($k \leq n$). Wtedy łuki te tworzą cykl skierowany. FCM jest nazywana cykliczną, jeśli zawiera cykle skierowane. FCM jest nazywana acykliczną, jeśli nie zawiera żadnego cyklu skierowanego.

DEFINICJA 3.15

Cykliczna FCM jest FCM ze sprzężeniem zwrotnym.

DEFINICJA 3.16

Jeśli w FCM występuje sprzężenie zwrotne, to taka FCM jest zwana układem dynamicznym.

DEFINICJA 3.17

Niech $\overrightarrow{C_1\,C_2}$, $\overrightarrow{C_2\,C_3}$, ..., $\overrightarrow{C_{n-1}\,C_n}$ będzie cyklem. Kiedy C_i jest w stanie on i jeżeli związki przyczynowe przepływają przez łuki cyklu tak, że ponownie skutkuje to pobudzeniem C_i, mówi się, że układ zatacza koło. Stan równowagi takiego układu dynamicznego nosi nazwę ukrytego wzorca (ang. hidden pattern).

DEFINICJA 3.18

Jeżeli stan równowagi układu dynamicznego jest unikatowym wektorem stanu, to nazywa się go ustalonym punktem (ang. fixed point) równowagi.

DEFINICJA 3.19

Jeżeli FCM ustala się przy wektorze stanu powtarzającym formę: $A_1 \to A_2 \to \cdots \to A_i \to A_1$, to taki stan równowagi nazywa się cyklem krańcowym (ang. limit cycle).

DEFINICJA 3.20

Jeśli na tym samym zestawie czynników C_1, ..., C_n określono więcej niż jedną FCM, to skończoną liczbę takich FCM można połączyć w celu uzyskania skumulowanego efektu działania ich wszystkich. Niech $\boldsymbol{E}_1, \ldots, \boldsymbol{E}_p$ będą macierzami sąsiedztwa określonymi na czynnikach C_1, ..., C_n. Wtedy skumulowane działanie FCM opisywanych przez te macierze uzyska się, dodając je zgodnie z (3.73):

$$\mathbf{E} = \mathbf{E}_1 + \mathbf{E}_2 + \cdots + \mathbf{E}_p \tag{3.73}$$

Łączenie macierzy sąsiedztwa opisane w definicji 3.20 dotyczy sytuacji, gdy np. zbiór powiązań przyczynowo-skutkowych pomiędzy czynnikami jest definiowany przez wielu ekspertów. Wtedy każdy z nich tworzy własną macierz sąsiedztwa ($\mathbf{E}_i$), a do dalszych rozważań bierze się macierz skumulowaną **E** (zgodnie z (3.73)), pamiętając przy tym o zastosowaniu dodatkowych mechanizmów progowych zapewniających nieprzekraczanie maksymalnych zadanych wartości wag łuków grafu FCM.

3.2.3. Podstawowe własności FCM

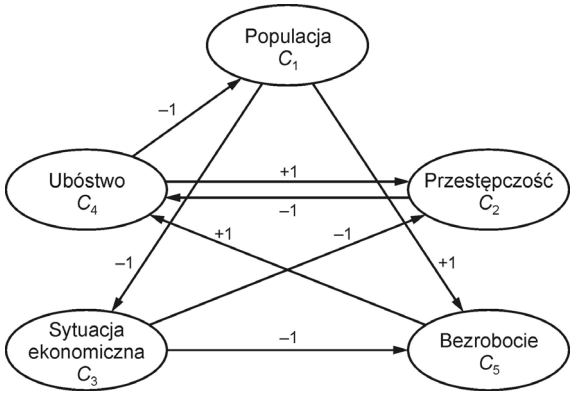
Przeważnie FCM są rozpatrywane jako układy dynamiczne, zgodne z definicją 3.16, w których zmienne w czasie funkcje czynników $C_i(t)$ mierzą nieujemne wystąpienia pewnych zdarzeń rozmytych. Zwykle stosuje się dwa główne podejścia do liczbowego opisu zależności przyczynowych pomiędzy czynnikami: $e_{i,j} \in \{-1, 0, 1\}$ (w prostych FCM – zgodnie z definicją 3.10) lub $e_{i,j} \in [-1, 1]$ (w złożonych FCM – zgodnie z definicją 3.11). W obu tych podejściach wartość $e_{i,j} > 0$ oznacza korelację dodatnią czynników C_i i C_j (w rozumieniu (3.60)), wartość $e_{i,j} < 0$ oznacza ko-

relację ujemną czynników C_i i C_j (w rozumieniu (3.61)), natomiast $e_{i,j} = 0$ oznacza brak powiązania przyczynowego pomiędzy czynnikami C_i i C_j.

W aplikacyjnych modelach decyzyjnych stosuje się raczej złożone FCM. Proste FCM mają zastosowanie przy tworzeniu szybkich wstępnych aproksymacji, ponieważ ich działanie polega głównie na stwierdzeniu występowania przyczynowości (wtedy oddziałuje ona z maksymalnym dodatnim lub ujemnym stopniem) lub jej braku pomiędzy wskazanymi czynnikami. W przykładzie 3.2 pokazano prostą FCM [90] oraz sposób poszukiwania cyklu krańcowego (zgodnie z definicją 3.19) takiego modelu.

Przykład 3.2

Rysunek 3.13 przedstawia model prostej FCM opisującej sytuację społeczno-ekonomiczną pewnej społeczności. Zakłada się, że czynniki (C_1, ..., C_5) mogą przyjmować dwa stany: aktywny i nieaktywny (*on* i *off* z definicji 3.13). Zadaniem jest zbadanie wpływu stałego wzrostu populacji na pozostałe czynniki.



Rys. 3.13. *Przykładowa prosta FCM opisująca sytuację społeczno-ekonomiczną pewnej społeczności [90]*

FCM z rysunku 3.13 może być reprezentowana przez macierz sąsiedztwa (3.74).

$$\mathbf{E} = \begin{bmatrix} 0 & 0 & -1 & 0 & 1 \\ 0 & 0 & 0 & -1 & 0 \\ 0 & -1 & 0 & 0 & -1 \\ -1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix} \tag{3.74}$$

Analizowana FCM ma dynamiczny charakter, w związku z czym poszukiwanie rozwiązania jest równoznaczne ze znalezieniem cyklu krańcowego (zgodnie z de-

finicją 3.19). Zostanie utworzony chwilowy binarny wektor stanu $A = (a_1, \dots, a_5)$ odwzorowujący chwilowe stany czynników. Ponieważ przedmiotem badania jest wpływ przyrostu czynnika populacja (C_1), to współrzędna a_1 będzie utrzymywana na stałym poziomie równym 1. Kolejne indeksy oznaczeń wektora A będą reprezentować jego wartość w następnych krokach przepływu sygnału przez FCM (pojedynczy krok oznacza przepływ sygnału pomiędzy dwoma następującymi po sobie czynnikami). Wartość początkowa chwilowego wektora stanu wynosi:

$$A_1 = (1 \quad 0 \quad 0 \quad 0 \quad 0)$$

W kolejnych krokach przepływu sygnału dokonywane są następujące operacje przejścia, progowania i aktualizacji (zgodnie z uwagą 3.1):

$$\begin{aligned}
A_1\mathbf{E} &= (0 \quad 0 \quad -1 \quad 0 \quad 1) \longrightarrow (1 \quad 0 \quad 0 \quad 0 \quad 1) = A_2 \\
A_2\mathbf{E} &= (0 \quad 0 \quad -1 \quad 1 \quad 1) \longrightarrow (1 \quad 0 \quad 0 \quad 1 \quad 1) = A_3 \\
A_3\mathbf{E} &= (-1 \quad 1 \quad -1 \quad 1 \quad 1) \longrightarrow (1 \quad 1 \quad 0 \quad 1 \quad 1) = A_4 \\
A_4\mathbf{E} &= (-1 \quad 1 \quad -1 \quad 0 \quad 1) \longrightarrow (1 \quad 1 \quad 0 \quad 0 \quad 1) = A_5 \\
A_5\mathbf{E} &= (0 \quad 0 \quad -1 \quad 0 \quad 1) \longrightarrow (1 \quad 0 \quad 0 \quad 0 \quad 1) = A_6 = A_2
\end{aligned}$$

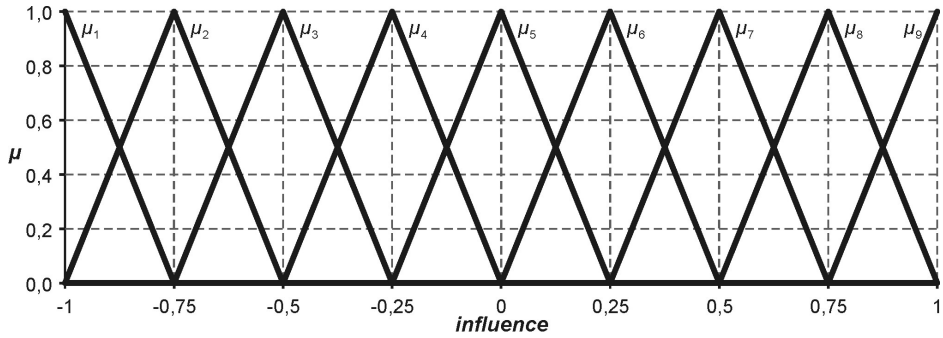
Znaleziono cykl krańcowy. Rozwiązaniem jest wektor A_2. Wynika z niego, że stały wzrost populacji (C_1) skutkuje wzrostem bezrobocia (C_5).

W analizie bardziej złożonych systemów trójwartościowy opis powiązań przyczynowych pomiędzy czynnikami jest niewystarczający. Do takich zastosowań wykorzystuje się złożone FCM, w których powiązania przyczynowe są reprezentowane przez ich wagi liczbowe z zakresu $[-1, 1]$ (zgodnie z definicją 3.11). Każdą z wag $(w_{i,j})$ można wyznaczyć, np. według następującego algorytmu [136, 189, 190]:

1. Zdefiniować zmienną lingwistyczną *influence* opisującą wpływ wartości czynnika C_i na wartość czynnika C_j oraz zbiór wartości lingwistycznych (ang. *term set*) tej zmiennej $T(influence)$. Przykładowa postać takiego zbioru mogłaby być następująca:

$$\begin{aligned}
T(influence) = \{&ujemne\ bardzo\ mocne, ujemne\ mocne, ujemne\ średnie, \\
&ujemne\ słabe, zerowe, dodatnie\ słabe, dodatnie\ średnie, \\
&dodatnie\ mocne, dodatnie\ bardzo\ mocne\}
\end{aligned} \tag{3.75}$$

Każda z wartości w zbiorze (3.75) jest zbiorem rozmytym, określonym na przestrzeni rozważań $U = [-1, 1]$, o pewnej funkcji przynależności. Przykładowe reprezentacje graficzne takich funkcji przynależności pokazano na rysunku 3.14.



Rys. 3.14. *Graficzna reprezentacja przykładowych wartości lingwistycznych zmiennej influence*

Funkcje μ_1, ..., μ_9 pokazane na rysunku 3.14 są funkcjami przynależności kolejnych wartości lingwistycznych z (3.75) – numerację zastosowano dla uproszczenia zapisu. Przy takim podejściu można sformułować regułę semantyczną $M(influence)$ [189] o następującej postaci:

$M(ujemne\ bardzo\ mocne)$ = zbiór rozmyty dla „oddziaływania poniżej $-0{,}75$” z funkcją przynależności μ_1;
$M(ujemne\ mocne)$ = zbiór rozmyty dla „oddziaływania bliskiego $-0{,}75$” z funkcją przynależności μ_2;
$M(ujemne\ średnie)$ = zbiór rozmyty dla „oddziaływania bliskiego $-0{,}5$” z funkcją przynależności μ_3;
$M(ujemne\ słabe)$ = zbiór rozmyty dla „oddziaływania bliskiego $-0{,}25$” z funkcją przynależności μ_4;
$M(zerowe)$ = zbiór rozmyty dla „oddziaływania bliskiego 0” z funkcją przynależności μ_5;
$M(dodatnie\ słabe)$ = zbiór rozmyty dla „oddziaływania bliskiego $0{,}25$” z funkcją przynależności μ_6;
$M(dodatnie\ średnie)$ = zbiór rozmyty dla „oddziaływania bliskiego $0{,}5$” z funkcją przynależności μ_7;
$M(dodatnie\ mocne)$ = zbiór rozmyty dla „oddziaływania bliskiego $0{,}75$” z funkcją przynależności μ_8;
$M(dodatnie\ bardzo\ mocne)$ = zbiór rozmyty dla „oddziaływania powyżej $0{,}75$” z funkcją przynależności μ_9.

2. Zebrać opinie ekspertów na temat sposobu, w jaki czynnik C_i wpływa na czynnik C_j. Każdy ekspert powinien określić kierunek oddziaływania („dodatni” bądź „ujemny”) oraz jego stopień z użyciem wartości lingwistycznych takich jak: „słabe”, „mocne” itp. – zgodnie z (3.75).

3. Dla każdego oddziaływania przyczynowego pomiędzy C_i a C_j dokonać agregacji sugestii wszystkich ekspertów (zgodnie z (3.38)). Należy przy tym pamiętać o uwzględnieniu „wiarygodności” poszczególnych ekspertów, którą

można ocenić na podstawie stopnia odbiegania ich opinii od średniej [190] lub innymi metodami (np. [191]).

4. Otrzymany w ten sposób zbiór rozmyty wyostrzyć metodą średniej ważonej zgodnie z (3.58). Uzyskana liczba $w_{i,j}$ jest wagą oddziaływania czynnika C_i na czynnik C_j.

Dodatkowo pojęcie chwilowego binarnego wektora stanu z definicji 3.13 można rozszerzyć tak, aby mógł on reprezentować więcej niż dwa stany wartości rozmytych czynników $C_1, \dots, C_n$.

DEFINICJA 3.21

Niech C_1, ..., C_n będą czynnikami FCM, a $A = (a_1, \dots, a_n)$ niech będzie wektorem takim, że $a_i \in [0,1]$. A nazywa się chwilowym wektorem stanu i opisuje wartości rozmyte czynników w danej chwili (3.76):

$$a_i = \rho(C_i) \tag{3.76}$$

gdzie:

$\rho_i : C_i \to [0,1]$ – *opracowana na podstawie wiedzy ekspertowej funkcja odwzorowująca rozmytą wartość czynnika C_i w przedziale liczb rzeczywistych o zakresie $[0,1]$;*

$i = 1, \dots, n$ *(n – liczba czynników mapy).*

Wtedy macierz sąsiedztwa **E** z (3.70) można zastąpić macierzą wag **W**, jak w (3.77), i dalsze rozważania prowadzić w oparciu o macierz wag i zmodyfikowany chwilowy wektor stanu;

$$\mathbf{W} = \begin{array}{c} \\ a_1 \\ a_2 \\ \vdots \\ a_n \end{array} \begin{array}{c} \begin{array}{cccc} a_1 & a_2 & \cdots & a_n \end{array} \\ \begin{bmatrix} 0 & w_{1,2} & \cdots & w_{1,n} \\ w_{2,1} & 0 & \cdots & w_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ w_{n,1} & w_{n,2} & \cdots & 0 \end{bmatrix} \end{array} \tag{3.77}$$

Składniki a_i wektora stanu A z definicji 3.21 mogą występować w formie ciągłej lub dyskretnej. W formie dyskretnej pojedynczy element przyjmuje wartości z przedziału $[0,1]$ zgodnie z (3.78):

$$\begin{gathered} a_i = \{a_i^{(0)}, a_i^{(1)}, \dots, a_i^{(m)}, a_i^{(m+1)}\} \\ (0 = a_i^{(0)} < a_i^{(1)} < \cdots < a_i^{(m)} < a_i^{(m+1)} = 1) \end{gathered} \tag{3.78}$$

gdzie:

$m + 2$ – liczba możliwych poziomów wartości a_i.

W ujęciu (3.78) binarny wektor stanu opisany przez (3.71) jest jedynie szczególnym przypadkiem ogólnej postaci (3.78) dla $m = 0$.

FCM reprezentowana przez macierz wag (3.77) jest niekiedy nazywana rozmytą siecią przyczynową (ang. *Fuzzy Causal Network*) [227] i traktowana jak rozwinięcie klasycznej FCM.

Większa rozpiętość możliwych abstrakcyjnych wartości w złożonych FCM powoduje, że wartości czynników w kolejnych krokach przepływu sygnału przez FCM nie mogą być wyznaczane metodą (pokazaną w przykładzie 3.2 ze str. 65). Zamiast niej stosuje się podejście, w którym wektor stanu w kolejnym kroku jest pewną funkcją tego wektora z kroku wcześniejszego. W komputerowych systemach obliczeniowych krok przepływu sygnału można utożsamić z krokiem czasu dyskretnego t i wtedy kolejne wartości wektora stanu będą wyznaczane zgodnie z (3.79):

$$A(t+1) = \omega(\mathbf{W}, A(t)) \tag{3.79}$$

gdzie:
ω – funkcja kumulacji oddziaływań przyczynowych;
t – czas dyskretny.

W klasycznym podejściu do modelowania z użyciem złożonych FCM [50, 128, 131, 181, 188, 210] przyjmuje się, że nowa wartość danego czynnika zależy wyłącznie od aktualnych wartości czynników wpływających. Wtedy następną wartość pojedynczego czynnika wyznacza się zgodnie z zależnością (3.80):

$$a_i(t+1) = f_p\left(\sum_{\substack{j=1 \\ j\neq i}}^{n} w_{i,j} a_i(t)\right) \tag{3.80}$$

gdzie:
f_p – funkcja progowa.

W modelu (3.80) zakłada się, że jeżeli skumulowane oddziaływanie pozostałych czynników jest zerowe (np. na skutek wzajemnego znoszenia się poszczególnych wpływów), to wynikowa wartość danego czynnika też będzie zerowa, niezależnie od jego wcześniejszego stanu. Takie podejście, chociaż często stosowane, nie wszędzie się sprawdza, dlatego też wprowadzono również jego rozwiniętą formę (3.81) [189]:

$$a_i(t+1) = f_p\left(p_1 \cdot a_i(t) + p_2 \cdot \sum_{\substack{j=1 \\ j\neq i}}^{n} w_{i,j} a_i(t)\right) \tag{3.81}$$

gdzie:

p_1, p_2 – współczynniki proporcji, dobierane na podstawie wiedzy ekspertowej (w modelach, budowanych z wykorzystaniem (3.81) [136, 139, 187, 189] przeważnie zakłada się, że $p_1 = 1$ oraz $p_2 = 1$).

Jeżeli wektor stanu A ma charakter dyskretny (jak w (3.78)), to po każdym kroku obliczeń wartości otrzymane przy użyciu (3.80) lub (3.81) muszą być dodatkowo aktualizowane tak, aby przyjmowały tylko dopuszczalne poziomy. Najprostszą metodą jest posłużenie się współczynnikiem bliskości opartym na odległości euklidesowej.

Występująca w równaniach (3.80) i (3.81) funkcja progowa f_p może mieć postać jak w (2.15), wykorzystuje się też inne, prostsze formy, takie jak (3.82) [227] lub, w zmodyfikowanej formie z przełącznikiem, (3.83):

$$f_p(v) = \begin{cases} 0 & \text{dla } v < 1 \\ 1 & \text{dla } v \geq 1 \end{cases} \tag{3.82}$$

$$f_p(v) = \begin{cases} 0 & \text{dla } v < 0{,}5 \\ 1 & \text{dla } v \geq 0{,}5 \end{cases} \tag{3.83}$$

Podobnie jak w prostych FCM, analiza złożonych FCM polega na poszukiwaniu pewnego stanu równowagi, która wytworzy się po pobudzeniu któregoś czynnika – można w ten sposób ocenić pośrednie oddziaływanie przyczynowe danego czynnika na dowolny inny. Dla dyskretnego wektora stanu A można w zasadzie próbować poszukiwania cyklu krańcowego według definicji 3.19 i, jak w przykładzie 3.2, znaleźć rozwiązanie na tej podstawie. Metoda ta nie będzie jednak przydatna przy ciągłych wartościach elementów wektora A. W takiej sytuacji powtarza się cyklicznie działania opisane równaniem (3.81) do chwili osiągnięcia (z założoną dokładnością) punktu ustalonego, czyli stanu, w którym zmiany wektora stanu A w kolejnych cyklach będą wystarczająco nieznaczne.

Z powyższego wynika, że model FCM powinien symulować cykle przepływu sygnałów przez czynniki dopóty, dopóki nie zostanie znaleziony cykl równowagi lub ustalony punkt równowagi. Możliwy jest też trzeci wynik: chaotyczne zmiany wartości czynników, co oznacza złe zaprojektowanie FCM.

Otrzymany wynik może być interpretowany bezpośrednio, w formie liczbowej, lub też, po odpowiednim przetworzeniu (np. metod z podrozdz. 3.1.5), zamieniony na odpowiednią wartość lingwistyczną.

Wiedza ekspertowa, będąca podstawą budowy FCM, rzadko jest w pełni wystarczająca do stworzenia funkcjonalnego modelu. Z tego powodu konieczne było opracowanie metod adaptacji parametrów FCM w celu lepszego dostosowania ich do rzeczywistych wskaźników pracy modelowanego systemu. Ta konieczność była zresztą jednym z powodów wprowadzenia miar liczbowych do rozmytych modeli

wnioskowania takich FCM – operowanie liczbami jest łatwiejsze przy tworzeniu zautomatyzowanych algorytmów adaptacyjnych. FCM, której aparat wnioskujący może się zmieniać (w wyniku uczenia się na danych odniesienia) nazywa się adaptacyjną FCM. W podrozdziale 3.2.4 zostaną pokrótce przedstawione główne metody adaptacji (uczenia) FCM.

3.2.4. Wybrane metody uczenia FCM

Jak wspomniano wcześniej, w większości zastosowań FCM wymaga dodatkowych zabiegów adaptacyjnych, które dostosują jej parametry (chodzi zwłaszcza o własności połączeń przyczynowych pomiędzy czynnikami) do rzeczywistych własności modelowanego obiektu. Proces takiego dostosowania nazywa się adaptacją lub uczeniem FCM. Istnieje wiele metod [136, 179] i wciąż powstają nowe, jednakże generalnie można je podzielić na dwie zasadnicze grupy: metody uczenia nienadzorowanego bazujące na prawie Hebba [55] oraz metody populacyjne wykorzystujące różnego typu algorytmy ewolucyjne [183]. Występują też podejścia mieszane (hybrydowe), łączące w sobie wybrane cechy obydwu wyżej wymienionych grup.

3.2.4.1. Metody wykorzystujące prawo Hebba

Prawo Hebba, ogłoszone w 1949 r., dotyczyło procesów uczenia zachodzących w mózgu i głosiło, że „połączenia neuronalne są wzmacniane i remodelowane w wyniku tego, czego doświadczamy". Dość szybko okazało się, że można tę zasadę, w zmodyfikowanej formie, zastosować również w systemach sztucznej inteligencji. Ogólnie zasadę Hebba, w odniesieniu do dwóch oddziałujących elementów (np. neuronów), można zapisać w następującej postaci:

$$w_{i,j}(k+1) = w_{i,j}(k) + \gamma \cdot x_i \cdot y_j \tag{3.84}$$

gdzie:
x_i – wartość i-tej wielkości wejściowej;
y_j – wartość j-tej wielkości wyjściowej;
$w_{i,j}$ – miara oddziaływania x_i na y_j;
γ – sygnał uczący (przeważnie zależny od $w_{i,j}$);
k – numer kolejnego kroku w cyklu uczenia.

W metodach wykorzystujących ogólną regułę (3.84) na potrzeby uczenia FCM chodzi zasadniczo o to, aby w kolejnych cyklach obiegu sygnałów modyfikować wagi dynamicznej, złożonej FCM dopóty, dopóki nie osiągnie ona stanu stabilnego, w którym wartości czynników przestaną się zmieniać lub też zmiany te będą mniejsze od założonych wartości granicznych. Oznacza to, że po każdej zmianie wartości powiązanych ze sobą czynników waga powiązania pomiędzy nimi również ulegnie zmianie w sposób zależny od wielkości zmian powiązanych czynni-

ków. Po zakończeniu każdego cyklu uczenia sprawdza się wartość odpowiedniego kryterium dokładności uczenia (można stosować różne kryteria, zależnie od charakteru badanego problemu – niektóre z nich zostaną przytoczone dalej przy opisie kolejnych metod uczenia). Osiągnięcie wartości założonej przez ekspertów oznacza zakończenie procedury.

Chronologicznie jako pierwszy pojawił się **różnicowy algorytm uczenia Hebba** (ang. ***DHL – Differential Hebbian Learning***) [30]. Pozostałe metody są w zasadzie modyfikacjami metody DHL. W skrócie rzecz ujmując, w metodzie tej waga $w_{i,j}$ powiązania pomiędzy czynnikami C_i i C_j zmienia się w każdym kroku czasu dyskretnego zgodnie z zależnością (3.85):

$$w_{i,j}(t+1) = \begin{cases} w_{i,j}(t) + \gamma(t)[\Delta a_i(t) \cdot \Delta a_j(t) - w_{i,j}(t)] & \text{dla} \quad \Delta a_i(t) \neq 0 \\ w_{i,j}(t) & \text{dla} \quad \Delta a_i(t) = 0 \end{cases} \tag{3.85}$$

gdzie:

a_i, a_j – elementy chwilowego wektora stanu A (wg definicji 3.21), reprezentujące wartości czynników C_i i C_j powiązanych oddziaływaniem o wadze $w_{i,j}$;

$\Delta a_i(t) = a_i(t) - a_i(t-1)$; $\Delta a_j(t) = a_j(t) - a_j(t-1)$;

$\gamma(t) = 0{,}1\left[1 - \frac{t}{1{,}1q}\right]$ – współczynnik uczenia (malejący z upływem czasu uczenia t);

q – parametr zabezpieczający przez nadmiernym wzrostem wartości wagi (często jest on równy planowanej docelowej liczbie kroków uczących).

Przy tym, w każdym kroku czasu dyskretnego obliczane są również nowe wartości wektora stanu A – zgodnie z (3.80) lub (3.81). Wadą metody DHL jest uwzględnianie jedynie bezpośrednich połączeń pomiędzy czynnikami z pominięciem pośrednich wpływów pozostałych czynników.

Wady tej nie ma **metoda zrównoważonego różnicowego algorytmu uczenia** (ang. **BDA – *Balanced Differential Algorithm***) [61], która uwzględnia również powiązania z innymi czynnikami, a jej działanie opisuje zależność (3.86). Metoda BDA ma ograniczone zastosowanie. Po pierwsze, można jej używać wyłącznie w odniesieniu do FCM z binarnym wektorem stanu. Po drugie, wprowadza (w oparciu o pewne reguły prawdopodobieństwa) wartości niezerowe do głównej przekątnej macierzy sąsiedztwa, czego zwykle się nie robi, ponieważ na ogół digrafy reprezentujące FCM nie posiadają pętli (w realnych systemach rzadko się zdarza, aby dany czynnik wpływał sam na siebie).

$$w_{i,j}(t+1) = \begin{cases} w_{i,j}(t) + \gamma(t)\left[\dfrac{\dfrac{\Delta a_i(t)}{\Delta a_j(t)}}{\sum_{\substack{k=1 \\ \Delta a_i(t)\Delta a_k(t)>0}}^{n} \dfrac{\Delta a_i(t)}{\Delta a_k(t)}} - w_{i,j}(t)\right] & \text{dla } \Delta a_i(t)\Delta a_j(t) > 0; i \neq j \\ w_{i,j}(t) + \gamma(t)\left[\dfrac{-\dfrac{\Delta a_i(t)}{\Delta a_j(t)}}{\sum_{\substack{k=1 \\ \Delta a_i(t)\Delta a_k(t)<0}}^{n} \dfrac{\Delta a_i(t)}{\Delta a_k(t)}} - w_{i,j}(t)\right] & \text{dla } \Delta a_i(t)\Delta a_j(t) < 0; i \neq j \\ w_{i,j}(t) + \dfrac{a_i(t)}{q} & \text{dla } i = j \end{cases} \quad (3.86)$$

Metoda aktywnego algorytmu uczenia Hebba (ang. ***AHL – Active Hebbian Learning***) [142] wykorzystuje pewne cykle w procesie uczenia FCM. W każdym cyklu wykonuje się pewną liczbę kroków symulacji, a w każdym takim kroku wybrane czynniki pełnią rolę czynników aktywujących (ang. *activation concepts*), wyzwalających czynniki z nimi powiązane, które z kolei stają się czynnikami aktywującymi w następnym kroku symulacji. Pojedynczy cykl uczenia kończy się po stwierdzeniu, że każdy czynnik był już czynnikiem aktywującym. Poszczególne kroki symulacji nazywa się krokami aktywacji. W każdym kroku aktywacji dokonuje się modyfikacji wagi oddziaływania pomiędzy czynnikiem aktywującym C_i (reprezentowanym przez odpowiadający mu element wektora stanu a_i) a czynnikiem aktywowanym C_j (reprezentowanym przez a_j). Dodatkowo w metodzie tej wyróżnia się czynniki wejściowe (które mogą być pobudzane zarówno sygnałami zewnętrznymi, jak i pochodzącymi od innych czynników), pośrednie oraz wyjściowe (których wartości mają zasadnicze znaczenie dla monitorowania decyzyjnego). Podział zależy od potrzeb modelowania i jest dokonywany na podstawie wiedzy ekspertowej. Ogólną postać matematyczną algorytmu AHL przedstawia równanie (3.87):

$$w_{i,j}(k+1) = (1-\gamma)\cdot w_{i,j}(k) + \eta \cdot a_i^{act}(k) \cdot \left[a_j(k) - w_{i,j}(k)\cdot a_i^{act}(k)\right] \quad (3.87)$$

gdzie:

k – numer kroku aktywacji;
a_i^{act} – wartość czynnika aktywującego;
a_j – wartość czynnika aktywowanego;
$w_{i,j}$ – waga powiązania pomiędzy czynnikami i-tym i j-tym;
γ, η – dodatnie współczynniki uczenia.

Metoda AHL zawiera pewne uproszczenia, jednakże (przy założeniu niewielkich wartości współczynników uczenia γ i η) pozwala na uzyskanie rezultatów szybciej niż w podstawowym podejściu DHL.

Kolejna metoda uczenia FCM opiera się na algorytmie [139, 143], w którym wykorzystano nieliniowe rozwinięcie [135] ogólnej reguły Hebba (3.84). Rozwinięcie to pierwotnie opracowano na potrzeby uczenia sztucznych sieci neuronowych, jednakże zostało ono dostosowane do specyfiki środowiska FCM. Nosi on nazwę **nieliniowego algorytmu uczenia Hebba** (ang. ***NHL – Nonlinear Hebbian Learning***). W metodzie tej przyjmuje się, że w każdym kolejnym kroku iteracji (czasu dyskretnego) wszystkie czynniki FCM są pobudzane, co prowadzi do zmiany ich wartości (zgodnie z (3.81)), jak również zmieniane są wartości wag połączeń pomiędzy poszczególnymi czynnikami. Dodatkowo przyjmuje się, że niektóre czynniki (wskazane przez ekspertów) pełnią rolę tzw. pożądanych czynników wyjściowych (ang. *DOCs – Desired Output Concepts*) i czynniki te powinny utrzymywać swoje wartości w stałych, wcześniej zaplanowanych przez ekspertów, granicach, które reprezentują docelowy przedział wartości wektora stanu A. Ponadto, również w drodze interwencji ekspertów, określa się znaki wszystkich wag i znaki te także pozostają stałe. Proponowaną w metodzie NHL postać algorytmu nieliniowej adaptacji wag (stosowanego z uwzględnieniem wyżej wymienionego ograniczenia) obrazuje równanie (3.88):

$$w_{i,j}(k+1) = \gamma \cdot w_{i,j}(k) + \eta(k) \cdot a_j(k) \cdot \left[a_i(k) - w_{i,j}(k) \cdot a_i(k)\right] \tag{3.88}$$

gdzie:

a_i, a_j – elementy wektora stanu reprezentujące wartości czynników połączonych przyczynowo z wagą $w_{i,j}$;

k – krok iteracji (często utożsamiany z czasem dyskretnym);

η – współczynnik uczenia (w postaci niewielkiej liczby dodatniej);

γ – współczynnik zaniku wagi (ma zadanie chronić wagę przed nadmiernym wzrostem i poprawiać zbieżność procesu uczenia).

Algorytm NHL wykonuje kolejne cykle adaptacji aż do spełnienia warunku stopu. Na potrzeby tego algorytmu opracowano dwa takie warunki. Pierwszy wynika z osiągnięcia określonej wartości funkcji dopasowania F_1, określonej równaniem (3.89):

$$F_1 = \sqrt{\sum_{i=1}^{n} \left(a_i^{DOC} - T_i\right)^2} \tag{3.89}$$

gdzie:

a_i^{DOC} – wartość i-tego pożądanego czynnika wyjściowego obliczona przez model FCM;

T_i – wartość docelowa i-tego pożądanego czynnika wyjściowego ustalona na podstawie wiedzy ekspertowej, obliczona zgodnie z (3.90):

$$T_i = \frac{T_i^{max} - T_i^{min}}{2} \tag{3.90}$$

gdzie:
T_i^{min}, T_i^{max} – dolna i górna granica docelowej wartości i-tego czynnika.

Drugi warunek określa wpływ zmienności wartości czynników na kontynuowanie algorytmu. Algorytm powinien zaprzestać działania, jeśli zmiany wartości czynników w kolejnych krokach staną się pomijalnie małe, czyli po spełnieniu dla wszystkich DOC zależności (3.91):

$$F_{2,i} = \left| a_{i,k+1}^{DOC} - a_{i,k}^{DOC} \right| < \varepsilon \tag{3.91}$$

gdzie:
ε – graniczna zmiana wartości pojedynczego czynnika;
i – numer kolejnego pożądanego czynnika wyjściowego;
k, $k+1$ – numery kolejnych cykli uczenia.

W metodzie NHL proces uczenia nie jest całkowicie nienadzorowany, ponieważ wybrane (pożądane) czynniki FCM mają zdeterminowane wartości, a wszystkie wagi połączeń – zdeterminowane znaki, jednakże jej zaletą jest szybkie uzyskiwanie rezultatów końcowych.

Pewną modyfikacją algorytmu NHL jest **metoda nieliniowego algorytmu uczenia Hebba sterowanego danymi** (ang. ***DDNHL – Data-driven NHL***) [180], wykorzystująca te same zasady uczenia co metoda NHL, jednakże oprócz wartości DOCs zadanych przez ekspertów, wyznaczane są w niej także (po każdym cyklu uczenia) wartości ustalone DOCs symulowane przez uczoną FCM. Na podstawie porównania tych dwóch grup wartości można szybciej zdecydować o osiągnięciu pożądanego efektu uczenia. W niektórych przypadkach (przedstawionych w [180]) takie podejście pozwala przeprowadzić uczenie FCM szybciej niż z bezpośrednim użyciem metody NHL.

Przedstawione do tej pory podejścia oparte o regułę Hebba operują bezpośrednio na grafie reprezentującym FCM. Rozpatrywano też uzupełnienie analizy takiego grafu o inne mechanizmy. Jedną z tego rodzaju metod jest analiza układu: czynnik modyfikowany + czynniki na niego wpływające jako specyficznego modułu reprezentowanego przez **rozmytą sieć Petriego** [101]. W metodzie tej klasyczna reguła Hebba jest wykorzystywana do modyfikacji wagi połączenia pomiędzy tranzycją, wzbudzaną przez czynniki wpływające, a czynnikiem wynikowym. Z pewnymi uproszczeniami, procedurę adaptacji wagi połączenia w pojedynczym kroku czasu dyskretnego t (pojedynczy cykl uczenia) można realizować zgodnie z równaniem (3.92):

$$w_{i,j}(t+1) = \left(w_{i,j}(t) + \frac{t_j^* n_i^*}{\alpha} \right) (1-\alpha)^{(t+1)} + \frac{t_j^* n_i^*}{\alpha} \tag{3.92}$$

gdzie:
t_j – j-ta tranzycja (pobudzana przez czynniki wpływające);
n_i – i-te miejsce sieci Petriego (reprezentujące czynnik wynikowy);
t_j^* – zakładana wartość $t_j(t)$ dla $t \to \infty$;
n_i^* – zakładana wartość $n_i(t)$ dla $t \to \infty$;
α – współczynnik zaniku wagi.

Metoda wykorzystująca sieć Petriego może być stosowana do badania dynamicznych zachowań modeli FCM warunkowo stabilnych w odpowiednim zakresie współczynnika α. Z powodu warunkowej stabilności algorytmu podejście takie nadaje się do implementacji w złożonych procesach podejmowania decyzji i uczenia (m.in. w procesach uczenia maszynowego [100, 102]).

Jako ostatnia w tej grupie zostanie wymieniona metoda uczenia FCM oparta na własnościach uczenia i wnioskowania **rozmytych sieci logicznych** (ang. ***FBNs – Fuzzy Boolean Nets***) [19–22]. Ma ona zastosowanie wyłącznie do FCM opartych na zbiorach reguł (ang. *RB-FCMs – Rule Based Fuzzy Cognitive Maps*). W metodzie tej każdemu czynnikowi towarzyszy pewna liczba sztucznych neuronów zgrupowanych w tzw. obszar, w którym neurony są połączone w siatkę przy pomocy nieważkich, losowych połączeń. Siatki te są używane do określania połączeń przyczynowych (typu: IF-THEN) pomiędzy obszarami. Każdy neuron posiada m wejść dla każdego z N poprzedzających obszarów oraz do $(m+1)^N$ wewnętrznych pamięci jednostkowych. Wartość każdego czynnika jest określana poprzez współczynnik aktywacji stowarzyszonego z nim obszaru (ten z kolei jest uzyskiwany w drodze analizy powiązań pomiędzy aktywnymi neuronami a pozostałymi neuronami układu). Struktura FBN jest używania do modyfikacji zbioru reguł przyczynowych badanej FCM. Uczenie FCM ma charakter pośredni, ponieważ de facto uczeniu podlega FBN. Operacje FBN bazują na losowych próbkach wejściowych, wobec czego uczenie FBN jest procesem probabilistycznym. Wewnętrzna pamięć binarna każdego następnika neuronowego jest modyfikowana w zależności od aktywacji wejść (przez poprzedniki neuronowe) oraz od stanu następnika. Taki rodzaj uczenia może być uznany za nawiązujący do reguły Hebba tylko w pewnych warunkach (jeśli działania wewnątrz sieci są definiowane z użyciem współczynników wagowych).

3.2.4.2. Metody populacyjne

Metody, które można określić ogólną nazwą populacyjnych posługują się dostępnymi zbiorami danych wejściowych i opierają się na modelach poszukiwawczych, które imitują dane wejściowe. Celem uczenia jest w nich doprowadzenie FCM do oczekiwanych wartości elementów wektora stanu dla wybranych czynników, a osiąga się to, poszukując optymalnej postaci macierzy wag (sąsiedztwa) przy użyciu różnych technik ewolucyjnych.

W **metodzie strategii ewolucyjnych** (ang. ***ES – Evolution Strategies***) [110] stosowane jest standardowe podejście do algorytmu ewolucyjnego oparte na cyklicznym dokonywaniu rekombinacji, mutacji, ewaluacji i selekcji aż do osiągnięcia kryterium stopu, a rozwiązanie problemu optymalizacji polega na znalezieniu minimum funkcji celu $f(W)$, gdzie W jest m-elementowym wektorem stanu. W przypadku adaptacji FCM wektor W jest macierzą sąsiedztwa zawierającą wagi powiązań przyczynowych pomiędzy czynnikami, a $m = n(n-1)$, gdzie n – liczba czynników (można pominąć elementy diagonalne macierzy W). Metoda strategii ewolucyjnych wymaga jednoczesnego istnienia wielu jednostek w postaci m-elementowych wektorów o elementach rzeczywistych (kolejnych wartości wektora stanu), i elementy te (w postaci $x_k \in \mathbb{R},\ 1 \le k \le m$) są zmiennymi obiektowymi. Wektor stanu jest zbudowany z elementów kolejnych wierszy macierzy sąsiedztwa w następujący sposób:

$$\mathbf{x} = (x_1, \dots, x_m) = \left(\underbrace{w_{1,2}, \dots, w_{1,n}}_{1 \text{ wiersz}}, \underbrace{w_{2,1}, \dots, w_{2,n}}_{2 \text{ wiersz}}, \dots, \underbrace{w_{n,1}, \dots, w_{n,n-1}}_{n \text{ wiersz}} \right)^T \tag{3.93}$$

przy czym elementy diagonalne macierzy sąsiedztwa ($w_{1,1}$, $w_{2,2}$, ..., $w_{n,n}$) są wykluczone z wektora **x**.

W metodzie zaproponowano rozszerzone podejście do mutacji, w którym każdy element wektora stanu podlega, jak zwykle w algorytmach ewolucyjnych, zmianie poprzez dodanie do niego losowej wartości o rozkładzie normalnym, o wartości średniej również zero i odchyleniu standardowym σ (będącym parametrem strategii), ale dodatkowo parametr σ również podlega zmianom, ponadto jest wyznaczany oddzielnie dla każdego elementu. Ilustruje to równanie (3.94):

$$\begin{aligned} x_k' &= x_k + \sigma_k{}' \cdot N_k(0,1) \\ \sigma_i' &= \sigma_i \cdot e^{(\tau' \cdot N(0,1) + \tau \cdot N_k(0,1))} \end{aligned} \tag{3.94}$$

gdzie:

x_k – waga kolejnego powiązania przyczynowego pomiędzy czynnikami ($k = 1, 2, \dots, n(n-1)$; n – liczba czynników FCM);

σ_k – parametr mutacji k-tego elementu;

$N(0,1)$ – liczba losowa o rozkładzie normalnym i odchyleniu standardowym równym 1 – parametr wspólny;

$N_k(0,1)$ – liczba losowa o rozkładzie normalnym i odchyleniu standardowym równym 1 – parametr wyznaczany oddzielnie dla każdego elementu;

τ – parametr, którego zadaniem jest zapewnienie uzyskania różnych zmian różnych elementów wektora stanu;

τ' – parametr, którego zadaniem jest zapewnienie zróżnicowania na poziomie całej populacji pomiędzy kolejnymi generacjami.

Na bazie ogólnego schematu (3.94) opracowano kilka form rozwojowych algorytmu adaptacji parametrów strategii [110], których stosowanie zależy od analizowanego przypadku. Proces uczenia FCM w metodzie ES opiera się na zbiorze par wejścia-wyjścia. Algorytm oblicza parametry struktury FCM, która jest w stanie generować sekwencje wektora stanu, co prowadzi do przekształcania wektorów wejściowych w wyjściowe. Główną wadą tej metody jest wymóg istnienia wielu sekwencji wektora stanu, co może być trudne do uzyskania w niektórych zastosowaniach praktycznych.

Metoda **optymalizacji rojem cząstek** (ang. ***PSO – Particle Swarm Optimization***) [140, 144] wykorzystuje znaną wcześniej tzw. inteligencję roju (inna nazwa to „inteligencja stadna") [95], bazującą na wiedzy o zachowaniach w zdecentralizowanym, samoorganizującym się systemie. W metodzie tej zakłada się istnienie roju (stada) cząstek poruszających się w m-wymiarowej przestrzeni poszukiwań $S \subset \mathbb{R}^m$. Możliwe położenia cząstek tego roju stanowią populację składającą się z potencjalnych rozwiązań problemu. Każda z cząstek roju porusza się z własną prędkością oraz jest wyposażona w pamięć swojego najlepszego położenia w przestrzeni poszukiwań. Pojedyncza i-ta cząstka może być reprezentowana przez m-wymiarowy wektor:

$$\mathbf{x}_i = (x_{i1}, x_{i2}, \dots, x_{im})^T, \tag{3.95}$$

a jej prędkość przez podobnego rodzaju wektor:

$$\mathbf{v}_i = (v_{i1}, v_{i2}, \dots, v_{im})^T. \tag{3.96}$$

Najlepsze położenie i-tej cząstki w przestrzeni S opisują współrzędne punktu:

$$\mathbf{p}_i = (p_{i1}, p_{i2}, \dots, p_{im})^T \in S, \tag{3.97}$$

a problem optymalizacji sprowadza się do znalezienia globalnego minimum funkcji celu $f : S \to \mathbb{R}$, czyli do znalezienia takiego punktu $\mathbf{x}^* \in S$, dla którego spełniona jest zależność:

$$\underset{\mathbf{x} \in S}{\forall} f(\mathbf{x}^*) \leq f(\mathbf{x}) \tag{3.98}$$

gdzie: $S \subset \mathbb{R}^m$ jest zbiorem niepustym.

Przenosząc wyżej wymienione podejście na grunt FCM, można uznać (modyfikując nieco oznaczenia), że pojedyncza, j-ta cząstka roju odpowiada wektorowi $\mathbf{x}$ zbudowanemu w oparciu o chwilowe wagi macierzy sąsiedztwa, identycznemu z (3.93), i że cząstka ta przemieszcza się z prędkością reprezentowaną przez wektor $\mathbf{v} = (v_1, \dots, v_m)^T$. Położenie najlepszej cząstki roju można oznaczyć jako $\mathbf{p}_g = \left(p_{g1}, \dots, p_{gm}\right)^T$, a najlepszą odwiedzoną pozycję j-tej cząstki jako $\mathbf{p} = (p_1, \dots, p_m)^T$. Wtedy ruchem takiej pojedynczej cząstki steruje zależność (3.99):

$$x_i(t+1) = x_i(t) + v_i(t+1) \tag{3.99}$$

gdzie:
$i = 1, \dots, m$ – numer pojedynczej współrzędnej wektora **x** (pojedynczej cząstki);
t – czas dyskretny (utożsamiany z krokiem iteracji).

Prędkość, z jaką zmienia się współrzędna pojedynczej cząstki może być wyliczana według (3.100) lub (3.101):

$$v_i(t+1) = \chi\left[v_i(t) + c_1 r_1(p_i(t) - x_i(t)) + c_2 r_2\left(p_{gi}(t) - x_i(t)\right)\right] \tag{3.100}$$

$$v_i(t+1) = \omega v_i(t) + c_1 r_1(p_i(t) - x_i(t)) + c_2 r_2\left(p_{gi}(t) - x_i(t)\right) \tag{3.101}$$

gdzie:
$i = 1, \dots, m$;
χ – współczynnik ścisku (ma na celu ograniczenie szybkości zmian);
ω – współczynnik wagi inercji (pełni rolę podobną do χ);
c_1, c_2 – dodatnie parametry nazywane odpowiednio parametrem poznawczym (kognitywnym) i zbiorowym (socjalnym);
r_1, r_2 – liczby losowe z rozkładu jednorodnego w przedziale $[0, 1]$;
p_i – optimum lokalne;
p_{gi} – optimum globalne.

W oparciu o wyżej wymienione zależności buduje się algorytm uczenia, w którym zakłada się, że wybrane czynniki FCM są czynnikami wyjściowymi:

$$C_{out_1}, \dots, C_{out_M}, \quad 1 \le M \le n \tag{3.102}$$

gdzie:
n – liczba czynników,
których wartości mają być utrzymywane w granicach określonych przez ekspertów. Wartości tych czynników są reprezentowane przez odpowiednie elementy wektora stanu A:

$$a_{out_i}^{min} \le a_{out_i} \le a_{out_i}^{max}, \quad i = 1, \dots, M \tag{3.103}$$

Głównym celem procesu uczenia jest znalezienie takiej macierzy wag FCM (tożsamej z (3.93)), przy której FCM osiągnie stan ustalony z jednoczesnym utrzymaniem wartości czynników wyjściowych (3.102) w ustalonych granicach (3.103). Dla osiągnięcia tego celu minimalizuje się funkcję celu o postaci (3.104):

$$F(\mathbf{x}) = \sum_{i=1}^{M} H\left(a_{out_i}^{min} - a_{out_i}\right)\left|a_{out_i}^{min} - a_{out_i}\right| + \sum_{i=1}^{M} H\left(a_{out_i} - a_{out_i}^{max}\right)\left|a_{out_i}^{max} - a_{out_i}\right| \tag{3.104}$$

gdzie:
H – funkcja Heaviside'a:

$$H(x) = \begin{cases} 0 & \text{dla} \quad x < 0 \\ 1 & \text{dla} \quad x \geq 0 \end{cases} \tag{3.105}$$

Kolejna metoda uczenia FCM opiera się o **algorytm ewolucji różnicowej** (ang. ***DE – Differential Evolution***) [27, 185]. Jest to heurystyczna iteracyjna metoda optymalizacji wykorzystująca mechanizmy ewolucyjne, takie jak krzyżowanie, mutacja oraz selekcja najlepszych osobników. Jej cechą charakterystyczną jest brak wymagań odnośnie istnienia gradientu lub ciągłości optymalizowanych funkcji. Algorytm operuje na populacji L osobników: x_1, ..., x_M. W każdym kroku iteracji dla każdego osobnika x_i tworzony jest osobnik próbny u_i, który powstaje poprzez zastosowanie mutacji i krzyżowania. Następnie sprawdzane jest dopasowanie rodzica x_i oraz osobnika próbnego u_i. Jeśli dopasowanie potomka u_i jest lepsze niż rodzica x_i, to osobnik x_i jest zastępowany osobnikiem u_i. Powyższy schemat implementuje się, tworząc dwie populacje (główną i próbną), z których każda składa się z L wektorów posiadających po M elementów. Tworząc nowy i-ty wektor, w pierwszym etapie dokonuje się mutacji, której wynikiem jest osobnik „mutant" otrzymany zgodnie z (3.106):

$$v_i = x_{r_1} + F \cdot (x_{r_2} - x_{r_3}) \tag{3.106}$$

gdzie:
F – stały współczynnik amplifikacji ($0 \leq F \leq 1$);
r_1, r_2, r_3 – losowo wygenerowane numery osobników ze zbioru $\{1, \ldots, L\}$, przy czym $r_1 \neq r_2 \neq r_3 \neq i$.

Kolejny etap to krzyżowanie rodzica x_i i mutanta v_i dokonywane zgodnie z (3.107):

$$u_{i_j} = \begin{cases} v_{i_j} & \text{jeżeli } rnd_j < CR \text{ lub } j = d \\ x_{i_j} & \text{w przeciwnym wypadku} \end{cases} \tag{3.107}$$

gdzie:
u_i – wynikowy wektor próbny (j oznacza numer kolejnego elementu wektora);
rnd_j – liczba losowa z przedziału $[0, 1)$, losowana niezależnie dla każdego j;
CR – prawdopodobieństwo przejścia z wektora mutanta v_j do wektora próbnego u_j ($0 \leq CR \leq 1$) – stały parametr algorytmu;
d – losowy numer elementu wektora losowany ze zbioru $\{1, \ldots, \mathrm{M}\}$.

Metoda DE występuje w różnych wariantach, w których stosuje się różne sposoby tworzenia populacji próbnej. W odniesieniu do uczenia FCM technikę DE wykorzystuje się jako niezależną metodę adaptacji macierzy wag powiązań pomię-

dzy czynnikami [85], pozwalającą unikać „utknięcia" algorytmu uczenia w optimum lokalnym, bądź też (częściej) łącznie z innymi metodami [138] w algorytmach dwustopniowego uczenia FCM. W obu przypadkach pojedynczym wektorem populacji x_i jest wektor złożony z kolejnych elementów macierzy wag, natomiast dopasowanie ocenia się, obliczając wartość odpowiedniej funkcji dopasowania (lub błędu) z wykorzystaniem narzuconych docelowych wartości wektora stanu FCM. Może to być np. funkcja dopasowania o postaci (3.108):

$$F(\mathbf{W}) = \sum_{i=1}^{n} \left(\left| a_i^{\min} - a_i \right| + \left| a_i - a_i^{\max} \right| \right) \tag{3.108}$$

gdzie:

$\mathbf{W}$ – macierz wag połączeń pomiędzy czynnikami FCM;

$a_i^{\min}$, $a_i^{\max}$ – dolna i górna granica wartości i-tego elementu wektora stanu A – zaplanowane na podstawie wiedzy ekspertowej;

a_i – wartość i-tego elementu wektora stanu obliczona przy użyciu modelu FCM (np. wg (3.81)).

Działanie **genetycznie ewoluowanych rozmytych map kognitywnych** (ang. ***GEFCMs – Genetically Evolved Fuzzy Cognitive Maps***) [7, 109] opiera się na połączeniu właściwości FCM z elementami algorytmów genetycznych (ang. *GAs – Genetic Algorithms*) [48, 96, 129, 158, 184]. Podstawą do tworzenia modeli tego rodzaju była chęć wyeliminowania niedogodności związanych z wcześniejszymi propozycjami tworzenia złożonych map kognitywnych. Aby je wyeliminować połączono kilka technik. Najpierw zastosowano neuronową technikę determinacji poziomów aktywacji czynników (ang. *ALs – Activation Levels*) przy pomocy tzw. neuronów pewności (ang. *CN – Certainty Neurons*), tworząc rozmyte mapy kognitywne oparte na neuronach pewności (ang. *CNFCMs – Certainty Neuron Fuzzy Cognitive Maps*) [195]. Następnie uzupełniono tę metodę o technikę adaptacji wag powiązań pomiędzy czynnikami, wykorzystującą algorytmy genetyczne, tworząc w ten sposób **genetycznie ewoluowane rozmyte mapy kognitywne oparte na neuronach pewności** (ang. ***GECNFCMs – Genetically Evolved Certainty Neuron Fuzzy Cognitive Maps***) [7]. Technika GECNFCM, po uzupełnieniu jej o możliwość multiobiektowej optymalizacji wag [125] (uwzględniającej scenariusze dla dwóch lub więcej AL), była adaptowana do implementacji w różnych dziedzinach związanych ze wspomaganiem procesów decyzyjnych [125, 126, 137, 138]. Złożoność pełnej nazwy powoduje, że modele takie niekiedy określa się również prościej – jako ewolucyjne FCM (ang. *EFCM – Evolutionary Fuzzy Cognitive Maps*), przy tym pamiętać należy, że nie wszystkie modele EFCM są w pełni identyczne z GECNFCM. Zadaniem algorytmu uczenia w GECNFCM jest doprowadzenie poszczególnych wag do wartości pozwalających na uzyskanie docelowych poziomów aktywacji (ang. *ALs – Activation Levels*) wybranych czynników w skończonej, założonej z góry, liczbie kroków (poziom aktywacji jest odpowiednikiem ele-

mentu wektora stanu z (3.76)). Algorytm ten operuje populacją chromosomów, z których każdy reprezentuje pojedynczą macierz wag połączeń pomiędzy czynnikami. Chromosomy podlegają modyfikacjom, a ocena jakości uzyskanej macierzy wag jest dokonywana w oparciu o funkcję dopasowania, która w przypadku śledzenia jednego AL przybiera postać (3.109) [125]:

$$F(\mathbf{W}_k) = \frac{1}{1 - \left|AL_{d,i} - mean_{50}(AL_{a,i})\right|} \tag{3.109}$$

gdzie:

$\mathbf{W}_k$ – badany k-ty chromosom (macierz wag);
$AL_{d,i}$ – docelowa wartość poziomu aktywacji i-tego czynnika;
$mean_{50}(AL_{a,i})$ – średnia arytmetyczna z ostatnich pięćdziesięciu wartości poziomu aktywacji i-tego czynnika (wyznaczonych przez CNFCM).

W zmodyfikowanej formie tej metody uczenia można śledzić dwa lub więcej AL i poszukiwać takiej macierzy wag, przy której FCM osiągnie pożądane wartości AL w zakładanej liczbie kroków iteracji. Wtedy funkcja dopasowania będzie miała postać (3.110):

$$F(\mathbf{W}_k) = \frac{1}{\sum_{j=1}^{M}\left|AL_{d,j} - AL_{a,j}\right|} \tag{3.110}$$

gdzie:

M – liczba AL, dla których poszukuje się rozwiązania;
j – numer kolejnego czynnika ze zbioru czynników, dla których określono docelowe wartości AL;
$AL_{d,j}$ – docelowa wartość AL j-tego czynnika;
$AL_{a,j}$ – aktualna wartość AL j-tego czynnika (wyznaczona przez CNFCM).

W zależnościach (3.109) i (3.110) wyższa wartość funkcji F oznacza lepsze dopasowanie macierzy wag $\mathbf{W}_k$. Sprawdzenie tej wartości jest dokonywane po wykonaniu przez CNFCM określonej liczby iteracji. W wyniku szeregu takich sprawdzeń wyłania się pary najlepiej dopasowanych chromosomów (macierzy wag) i dla każdej pary część GA (algorytm genetyczny) algorytmu uczenia przeprowadza rekombinację i stwarza dwa chromosomy potomne, których wartości są generowane metodą zgodną z (3.111):

$$w_{x,y}^{new} = w_{x,y}^{(1)} \cdot a + w_{x,y}^{(2)} \cdot (1 - a) \tag{3.111}$$

gdzie:

$w_{x,y}$ – element macierzy wag leżący na przecięciu wiersza o numerze x i kolumny o numerze y;

new, (1), (2) – indeksy górne oznaczające elementy odpowiednio: nowej (potomnej) macierzy wag oraz pierwszej i drugiej najlepiej dopasowanych macierzy wag;

a – losowa wartość z zakresu $[-0.25, 1.25]$ oddająca procentowy udział każdego chromosomu rodzicielskiego w chromosomie potomnym.

Stworzony w ten sposób zbiór chromosomów pochodnych poddawany jest sprawdzeniu pod kątem dopasowania jego elementów (z użyciem (3.109) i (3.110)), po czym tworzona jest nowa generacja, na którą składają się w równych proporcjach najlepiej dopasowane chromosomy rodzicielskie i potomne. Na tak powstałym zbiorze nowych chromosomów wykonuje się kolejny cykl mutacji, a cykle te powtarza się do osiągnięcia warunku stopu. W swojej podstawowej formie metoda ta może (z uwagi na losowy aspekt mutacji) prowadzić do pogarszania dopasowania, w związku z czym w jej bardziej rozwiniętych formach stosuje się rodzaj pamięci i zwielokrotnia się liczbę mutacji dla każdego chromosomu, co jednak komplikuje algorytm. Ponadto w metodzie tej wymagana jest znajomość danych „historycznych", które nie przy każdym zadaniu są dostępne.

Metoda uczenia oparta na **kodowaniu algorytmu genetycznego przy użyciu liczb rzeczywistych** (ang. ***RCGA – Real-Coded Genetic Algorithm***) [58, 179, 184] powstała na bazie poszukiwań metody uczenia FCM, która byłaby wolna od głównych niedogodności spotykanych w innych metodach, a które wynikały z dyskretnego charakteru danych, konieczności udziału ekspertów w określaniu celów uczenia oraz (w niektórych metodach) dużej ilości danych historycznych używanych przez algorytm. Jej zalety to niemal pełna automatyzacja procesu uczenia oraz wymóg istnienia tylko jednej sekwencji wektora stanu. W pewnym sensie zaletą jest też ciągły charakter parametrów FCM (w pewnym sensie, ponieważ nie zawsze jest to korzystne, a ponadto oddala model od intuicyjnego rozumienia pojęcia rozmycia danych). W metodzie tej operuje się populacją chromosomów określoną jak w (3.93). Zakłada się też, że jeśli istnieje pewien cykl krańcowy (zgodny z definicją 3.19), to jeżeli zbiór danych wejściowych zawiera K sekwencji wektora stanu A, a cykl krańcowy składa się z $L < K$ iteracji, to zbiór danych wejściowych użytych do uczenia FCM redukuje się do pierwszych L iteracji. W dalszych rozważaniach oznaczenie K będzie używane w odniesieniu do zbioru sekwencji pojedynczego cyklu krańcowego (warto zauważyć, że im wyższa wartość K, tym lepsze walory uczące ma zbiór danych wejściowych). Wprowadza się też określenia: „wektor początkowy" (ang. *initial vector*) oraz „odpowiedź systemu" (ang. *system response*), których znaczenie wyjaśnia zależność (3.112). Działanie złożonej FCM polega na wykonywaniu kolejnych iteracji (cykli obiegu sygnałów), w wyniku których czynniki przyjmują wartości reprezentowane przez kolejne wektory stanu. Pary następujących po sobie wektorów stanu mogą być określone następująco:

$$\underset{t=0,1,\ldots,K-1}{\forall} A(t) \to A(t+1) \tag{3.112}$$

gdzie:
K – liczba sekwencji wektora stanu w cyklu krańcowym;
$A(t)$ – wektor początkowy;
$A(t+1)$ – odpowiedź systemu.

Warunkiem działania algorytmu uczącego jest dostęp do zbioru danych odniesienia (wartości oczekiwanych czynników – uzyskanych np. drogą pomiarów w rzeczywistym obiekcie). Funkcja dopasowania (3.113) jest obliczana dla każdego chromosomu w oparciu o analizę różnic pomiędzy odpowiedziami systemu wygenerowanymi przez uczoną FCM, a odpowiedziami systemu odniesienia dla wszystkich $K-1$ par z (3.112):

$$F(\mathbf{W}_j) = h(Error(\mathbf{W}_j)) \tag{3.113}$$

gdzie:
h – funkcja pomocnicza (3.114), której zadaniem jest zapewnienie powiązania lepszego chromosomu z większą wartością funkcji dopasowania oraz dodanie nieliniowości „nagradzającej" chromosomy bliższe oczekiwanemu rozwiązaniu:

$$h(x) = \frac{1}{ax+1} \tag{3.114}$$

gdzie:
a – ustalany eksperymentalnie parametr o dużej wartości (w testach stosowano $a = 10000$);
$Error(\mathbf{W}_j)$ – miara sumarycznego błędu działania FCM z macierzą wag (chromosomem) $\mathbf{W}_j$, określona równaniem (3.115) (jest to jedna z możliwych form):

$$Error(\mathbf{W}_j) = \frac{1}{n(K-1)} \sum_{t=1}^{K-1} \sum_{i=1}^{n} |\hat{a}_i(t) - a_i(t)| \tag{3.115}$$

gdzie:
n – liczba czynników;
$\hat{a}_i(t)$ – i-ty element odpowiedzi systemu odniesienia;
$a_i(t)$ – i-ty element odpowiedzi systemu analizowanej FCM.

Podobnie jak w technice GECNFCM, po dokonaniu oceny dopasowania wszystkich chromosomów należy, przy użyciu metod ewolucyjnych, dokonać rekombinacji genetycznej chromosomów i stworzyć nową populację do kolejnego cyklu obliczania funkcji dopasowania. Możliwości na tym polu są wielorakie, można na przykład zastosować następujące operatory ewolucyjne [228]: funkcja rekombinacji: krzyżowanie jednopunktowe, funkcja mutacji: mutacja niejednorodna, metoda selekcji: koło ruletki, prawdopodobieństwo rekombinacji: 0,9, prawdopodobieństwo mutacji: 0,5.

Zastosowanie techniki RCGA do uczenia FCM daje lepsze rezultaty niż inne metody [37], wiąże się jednak z ograniczeniami wynikającymi z dużych rozmiarów przestrzeni poszukiwań, co przy dużych FCM stanowi poważny problem. Jedną z metod jego rozwiązania jest zastosowanie technik obliczeń równoległych, które można stosować przy złożonych problemach dających się podzielić na mniejsze integralne części. Wtedy każdą z części można rozwiązywać oddzielnie (w tym samym czasie co pozostałe – wykorzystując możliwości procesora lub procesorów oraz pamięci komputera wykonującego obliczenia), a następnie zespolić rozwiązania cząstkowe w jeden wynik końcowy. Podobną techniką posłużono się, tworząc **metodę PRCGA** (ang. ***Parallel Real-Coded Genetic Algorithm***) [182]. W przypadku metod inteligentnych problem stanowi nie tyle techniczny aspekt dokonywania obliczeń, co samo rozdzielenie podzadań do równoległego wykonania. Próby takie, w odniesieniu do RCGA, zostały przeprowadzone [182] i dowiodły swojej skuteczności (przejawiającej się skróceniem czasu uczenia FCM).

Inne podejście do problemu ewolucyjnego uczenia dużych FCM oparto na wprowadzeniu pewnych modyfikacji do algorytmu genetycznego, który z natury rzeczy jest powolny z uwagi na swoją złożoność obliczeniową. Poprawę efektywności czasowej można uzyskać, dzieląc zbiór danych (populację) na podzbiory (subpopulacje) i ucząc model FCM oddzielnie dla każdego podzbioru. Uzyskuje się w ten sposób pewną liczbę modeli wynikowych, które następnie należy scalić dla otrzymania rozwiązania końcowego. Kluczowym problemem jest dobór kryteriów dokonywania podziału populacji wejściowej. Spośród różnych podejść [178] najbardziej popularne są dwa [18]:

a) typu "master – slave" z globalną pojedynczą populacją – przetwarzaniu podlega cała populacja, ale określanie dopasowania odbywa się przy użyciu pracujących równolegle wielu procesorów (podobnie jak w PRCGA);
b) typu „gruboziarnistego” z wieloma populacjami: istnieje wiele populacji (które tylko okazjonalnie mogą wymieniać się elementami) i każda z nich jest przetwarzana oddzielnie, a wyniki są scalane w końcowym etapie.

Metoda typu „gruboziarnistego” opiera się na algorytmie typu „dziel i zwyciężaj” (ang. *divide and conquer*). Jest to w zasadzie rozbudowane podejście wykorzystujące RCGA oraz pewne elementy PRCGA. Zostało ono opisane w [178] i można je określić jako **kodowany przy użyciu liczb rzeczywistych algorytm genetyczny wykorzystujący metodę „dziel i zwyciężaj”** (ang. ***DCRCGA – Divide and Conquer Real-Coded Genetic Algorithm***). Populację wejściową dzieli się na S wzajemnie rozłącznych subpopulacji (S – liczba dostępnych procesorów). Następnie, przy równoległym użyciu S procesorów, techniką RCGA generuje się S odrębnych modeli FCM (odbywa się to w tym samym czasie). W tym podejściu, inaczej niż we wcześniejszych wykorzystujących RCGA, każdy chromosom zawiera n^2 (a nie $n(n-1)$ składników. Wykorzystywana w metodzie funkcja dopasowania ma postać (3.116) [178]:

$$F(\mathbf{W}_k) = \alpha \frac{1}{\beta \cdot \sum_{t=1}^{K-1} \sum_{i=1}^{n} \left(\hat{a}_i(t) - a_i(t)\right)^2 + 1} \tag{3.116}$$

gdzie:
$\hat{a}_i(t)$ – i-ty element odpowiedzi systemu odniesienia;
$a_i(t)$ – i-ty element odpowiedzi systemu analizowanej FCM;
K – liczba obserwacji (punktów danych wejściowych);
α, β – dodatnie współczynniki skalowania.

Scalanie cząstkowych submodeli w model końcowy może być przeprowadzane na różne sposoby, jednak najprostszym jest uśrednianie elementów macierzy wag poszczególnych submodeli FCM, przy czym należy uwzględnić dopasowanie poszczególnych submodeli (wyrażone błędami dopasowania). Wtedy wynikowa macierz wag FCM może być scalana według następującej zależności:

$$w_{i,j} = \frac{\sum_{s=1}^{S} cr_s \cdot w_{i,j}^s}{\sum_{s=1}^{S} cr_s} \tag{3.117}$$

gdzie:
S – liczba submodeli;
$w_{i,j}^s$ – waga połączenia pomiędzy czynnikami C_i i C_j w submodelu o numerze s;
cr_s – współczynnik wiarygodności submodelu o numerze s reprezentujący jego dopasowanie.

Metaheurystyczna metoda uczenia FCM z wykorzystaniem **symulowanego wyżarzania** (ang. ***SA – Simulated Annealing***) [4, 43] opiera się na opisanej w 1983 r. [97] technice poszukiwania globalnego minimum w modelach silnie liniowo niezależnych, z wieloma nieuporządkowanymi zależnościami [97, 150]. Technika ta odwzorowuje w pewnym sensie fizyczny proces wyżarzania metalowego obiektu, podczas którego metal jest rozgrzewany do wysokiej temperatury, a następnie stopniowo schładzany aż do osiągnięcia stanu krystalizacji charakteryzującego się najniższym stanem energetycznym obiektu. Stosuje się też nazewnicze analogie, takie jak: „temperatura", „energia", „wyżarzanie", „schładzanie". W skrócie schemat działania metody jest następujący. Zakłada się, że cały układ składa się z m cząstek. Ustala się temperaturę początkową T oraz (losowo) położenia początkowe wszystkich cząstek, a także oblicza się początkową energię E układu, która staje się energią odniesienia. Następnie wykonuje się kolejne cykle następujących kroków:
– losowanie próbnego przemieszczenia losowo wybranej cząstki;
– sprawdzenie energii układu po takim przemieszczeniu. Jeżeli jest ona mniejsza od energii odniesienia, to krok jest akceptowany, a energia układu staje się nową energią odniesienia. Jeżeli nie, to oblicza się prawdopodobieństwo Boltzmanna:

$$P = e^{-\frac{\Delta E}{k_B T}} \tag{3.118}$$

gdzie:
ΔE – przyrost energii układu;
k_B – stała Boltzmanna;
T – temperatura układu,

po czym, jeśli P wyliczone równaniem (3.118) jest większe od pewnej losowo wybranej liczby R z przedziału $[0, 1]$, to krok ten również jest akceptowany. Takie działanie chroni (do pewnego stopnia) algorytm przed „utknięciem" w minimum lokalnym.

Po wykonaniu $100 \cdot m$ powyższych cykli (lub po uzyskaniu $10 \cdot m$ zaakceptowanych zmian położenia cząstek) obniża się temperaturę zgodnie z zależnością (3.119):

$$T_{l+1} = \alpha \cdot T_l \tag{3.119}$$

gdzie:
T_l, T_{l+1} – temperatury w kolejnych sekwencjach cykli chłodzenia;
α – współczynnik chłodzenia ($0 < \alpha < 1$), który może być wyznaczany na wiele sposobów.

Algorytm zatrzymuje się (warunek stopu) po wystąpieniu jednej z następujących okoliczności: zostanie wykonana założona liczba zmian temperatury, temperatura spadnie poniżej założonego poziomu lub energia układu nie zmieni się pomimo obniżenia temperatury.

W środowisku FCM powyższa metoda wymaga jedynie niewielkich modyfikacji interpretacyjnych [4, 43]. Zbiorem cząstek jest zbiór, którego elementami są wagi połączeń pomiędzy czynnikami. Przemieszczeniu cząstki odpowiada losowa zmiana wartości wagi. Ekwiwalentem energii układu jest błąd dopasowania wyznaczany zgodnie z (3.120):

$$error(\mathbf{W}_k) = \frac{1}{n(K-1)} \sum_{t=1}^{K} \sum_{i=1}^{n} \left(a_i(t) - \hat{a}_i(t)\right)^2 \tag{3.120}$$

gdzie:
$\hat{a}_i(t)$ – i-ty element odpowiedzi systemu odniesienia;
$a_i(t)$ – i-ty element odpowiedzi systemu analizowanej FCM;
K – liczba obserwacji (punktów danych wejściowych);
n – liczba czynników FCM,

a używany przy zmianie temperatury współczynnik α z (3.119) wyznacza się następująco:

$$\alpha = \frac{1}{1 + e^{-\frac{\Delta}{T}}} \tag{3.121}$$

gdzie:

Δ – różnica pomiędzy aktualną a najlepszą wartością energii układu, wyznaczana zgodnie z (3.122):

$$\Delta = error(\mathbf{W}_T) - error(\mathbf{W}_{best}) \tag{3.122}$$

gdzie:

$\mathbf{W}_T$ – aktualna macierz wag uzyskana dla temperatury T;

$\mathbf{W}_{best}$ – najlepiej dopasowana spośród dotychczas uzyskanych macierzy wag.

Warto podkreślić, że rozwiązanie gorsze od istniejącego optimum nie zawsze jest odrzucane. Jeśli spełni ono pewne kryterium aspiracji (określone w (3.118)), to może ono zostać zaakceptowane. Takie rozwiązanie pozwala uchronić algorytm przed zatrzymaniem się w minimum lokalnym. Dodatkowo można zastosować uzupełniające rozszerzenie polegające na zastąpieniu losowania nowych wartości wag przez mechanizmy przejściowo chaotycznej dynamiki, co ma miejsce w **metodzie chaotycznego symulacyjnego wyżarzania** (ang. ***CSA – Chaotic Simulated Annealing***) [4]. Modyfikacja taka, co prawda, komplikuje algorytm, ale pozwala na zwiększenie efektywności metody.

Metoda SA uznawana jest za jeden z nielicznych algorytmów umożliwiających praktyczne uzyskanie minimum globalnego funkcji wielu zmiennych, ma jednak słabe punkty związane z wrażliwością na dobór temperatury początkowej, algorytmu przemieszczania cząstki oraz metody schładzania. Niewłaściwe określenie któregokolwiek z tych parametrów znacząco pogarsza jakość uczenia FCM.

Metoda uczenia oparta na **algorytmie „wielkiej powodzi”** (ang. ***GDA – Great Deluge Algorithm***) [33] symuluje ucieczkę na wzniesienie przed wzbierającą wodą. Ciągłe podnoszenie się poziomu wody grozi utknięciem algorytmu w optimum lokalnym (czyli na najbliższym wzniesieniu, które wcale nie musi być najwyższe w okolicy), w związku z czym w rozszerzonych wersjach tej metody (np. w odmianie nazwanej *record-to-record*) zastosowano dodatkowy mechanizm, który może zaakceptować każde rozwiązanie pod warunkiem, że nie jest ono „znacząco gorsze” od poprzedniego. Metoda ta charakteryzuje się łatwością implementacyjną wynikającą z niewielkiej liczby parametrów wymagających regulacji. Zasadniczo stanowi pewną odmianę metody SA. Do uczenia FCM zastosowano ją [12] pod nazwą *EGDA* (*Extended Great Deluge Algorithm*). Główna idea polega na znalezieniu prawidłowej macierzy wag połączeń pomiędzy czynnikami poprzez minimalizację pewnej funkcji celu. Podobnie jak w metodzie SA rozwiązanie gorsze od bieżącego może zostać zaakceptowane, jeśli spełnione zostaną określone warunki, przy czym tutaj warunkiem takim jest utrzymanie wartości dopasowania na poziomie niższym lub co najwyżej równym pewnej górnej wartości granicznej B, która jest sukcesywnie obniżana o tzw. współczynnik zaniku ΔB. Wraz z obniżaniem się górnej granicy akceptacji B zawęża się przestrzeń poszukiwań rozwiązania i zmniejsza się prawdopodobieństwo akceptacji rozwiązania gorszego. Nowe wartości wag (sąsiedztwo) FCM są generowane w oparciu o równanie (3.123):

$$w_{i,j}^{*} = w_{i,j} + (2 \cdot random() - 1) \cdot step_{C_{i,j}} \tag{3.123}$$

gdzie:

$w_{i,j}$ – waga połączenia pomiędzy czynnikami C_i i C_j przed wykonaniem ruchu w sąsiedztwie;

$w_{i,j}^{*}$ – waga połączenia pomiędzy czynnikami C_i i C_j po wykonaniu ruchu w sąsiedztwie;

$random()$ – funkcja generująca liczby pseudolosowe z zakresu $(0, 1)$;

$step_{C_{i,j}}$ – rozmiar (skalarny) kroku dla ruchu w sąsiedztwie.

Macierz wag jest wstępnie projektowana z użyciem wiedzy ekspertowej, a modyfikuje się ją zgodnie z (3.123), przy czym po każdej modyfikacji sprawdza się wartość funkcji celu (w [12] użyto funkcji o postaci (3.104)). Podobnie jak w metodzie SA, jeśli obliczona wartość funkcji celu jest mniejsza od bieżącego minimum, to zmiana jest akceptowana. Jeśli nie, to taka zmiana też może zostać zaakceptowana pod warunkiem, że wartość funkcji celu nie przekracza ustalonej wcześniej górnej granicy B. Po wykonaniu pewnej liczby cykli stan macierzy wag się stabilizuje. Wtedy obniża się wartość B o współczynnik zaniku ΔB oraz zmniejsza się parametr kroku $step_{C_{i,j}}$, po czym rozpoczyna się kolejną serię cykli. Dla zwiększenia dokładności zmiany parametrów mogą być przeprowadzane kilkuetapowo (np. w [12] parametr $step_{C_{i,j}}$ zmieniano od 0,9 do 0,1 z krokiem 0,1, po czym zastąpiono to zmianami z krokiem $0{,}001$ do osiągnięcia wartości $0{,}001$, natomiast współczynnik zaniku ΔB miał stałą wartość $0{,}001$). Algorytm kończy działanie kiedy funkcja celu osiągnie wartość niższą od założonej granicy lub zostanie przekroczona maksymalna liczba odrzuconych rozwiązań, lub zostanie przekroczona maksymalna liczba iteracji.

Metaheurystyczna metoda uczenia FCM z wykorzystaniem techniki **przeszukiwania tabu** (ang. ***TS – Tabu Search***) [45–47, 150] jest wariantem algorytmu samotnego poszukiwacza stanowiącego niemonotoniczne rozwinięcie deterministycznej heurystyki lokalnych ulepszeń. Podobnie jak w metodzie SA operuje się na populacji (sąsiedztwie) rozwiązań, z których każde jest pewnym układem cząstek, które mogą wykonywać określone ruchy. Po wykonaniu każdego ruchu zmienia się stan układu, więc należy sprawdzić jego dopasowanie zgodnie z określonym kryterium. Jeśli to dopasowanie jest lepsze od bieżącego, to staje się ono nowym rozwiązaniem bieżącym. Ponieważ cząstki wykonują ruchy zgodnie z pewnymi zasadami, to łatwo może dojść do sytuacji, w której uzyskane rozwiązania będą się powtarzać, co może doprowadzić do oscylacji algorytmu wokół minimum lokalnego. Aby temu zapobiec tworzy się listę rozwiązań niedozwolonych (ang. *tabu*), na której umieszcza się rozwiązania, które już wcześniej zostały uznane za bieżące. W praktyce stosuje się pewne odstępstwa od tej reguły:

– zakazem ponownego wyboru obejmuje się, zamiast pełnych rozwiązań, transformacje zwane ruchami (takim ruchem może być np. zmiana wartości którejś cząstki lub zamiana miejscami dwóch cząstek);
– zakaz taki ma charakter czasowy (tzn. że obejmuje tylko ruchy wykonane w ostatnich L iteracjach); osiąga się to, nadając liście tabu formę struktury typu FIFO o długości L.

Dodatkowo, aby nie zablokować akceptacji poprawnych rozwiązań, które jeszcze się nie pojawiły, stosuje się uzupełniającą regułę selekcji opartą na tzw. kryteriach aspiracji, które określają sytuacje wyjątkowe. Ruch spełniający kryteria aspiracji jest ruchem dozwolonym, nawet jeśli znajduje się na liście tabu. Podstawowym kryterium aspiracji jest wytworzenie w wyniku zastosowania danego ruchu rozwiązania lepszego niż bieżące.

Technika TS została z powodzeniem wykorzystana do uczenia FCM [6]. Podobnie jak w prezentowanej wcześniej technice SA, cząstkami analizowanego zbioru były elementy macierzy wag FCM, której wartości początkowe zostały wylosowane. Poszukiwanym rozwiązaniem jest macierz wag o takim układzie elementów, który zapewnia najlepsze dopasowanie FCM do zakładanego wyniku. Miarą dopasowania wyniku jest zależność tożsama z (3.120) – poszukuje się jej minimalnej wartości. Algorytm kończy działanie w jednej z dwóch następujących sytuacji: znalezienie najlepszego rozwiązania, co poznaje się po ustabilizowaniu się błędu dopasowania (3.120); upłynięcie czasu pracy algorytmu (założonego limitu liczby cykli uczenia). Jak pokazały wyniki badań [6], dokładność algorytmu TS jest o rząd wielkości lepsza niż techniki oparte na standardowym algorytmie genetycznym.

Uczenie FCM z użyciem **algorytmu immunologicznego** (ang. ***IA – Immune Algorithm***) wykorzystuje metody i techniki sztucznego systemu immunologicznego [3, 28, 84, 203], którego działanie odwzorowuje (w pewnym sensie) pracę systemu immunologicznego w żywym organizmie. Działanie naturalnego systemu immunologicznego sprowadza się, w ogólnym zarysie, do rozpoznawania obcych struktur (antygenów) przez specjalne komórki zwane limfocytami oraz produkcji (również przez limfocyty) przeciwciał najbardziej odpowiednich do zwalczania antygenów. Istotnymi elementami działania takiego systemu są: mechanizm selekcji klonalnej oraz mechanizmy selekcji pozytywnej i negatywnej. Działanie mechanizmu selekcji klonalnej ma na celu wyróżnienie i namnożenie przeciwciał najlepiej zwalczających określony antygen. W tym celu uaktywniony (w wyniku rozpoznania antygenu) limfocyt jest wielokrotnie powielany poprzez klonowanie (proces ten nosi nazwę proliferacji), a następnie poszczególne klony są poddawane tzw. hipermutacji, w trakcie której poprawia się dopasowanie poszczególnych przeciwciał. Powstała w ten sposób populacja zmutowanych klonów jest sprawdzana pod kątem dopasowania do antygenu, po czym klony słabo dopasowane są usuwane. Mechanizmy selekcji mają za zadanie usunięcie z populacji limfocytów produkujących przeciwciała reagujące na komórki własnego organizmu. Zasada

działania sztucznego systemu immunologicznego jest podobna i opiera się na trzech głównych elementach: reprezentacji komponentów systemu, mechanizmie przeliczania interakcji osobnika z otoczeniem (najczęściej wyrażanym w postaci jakiegoś rodzaju funkcji dopasowania) oraz procedurze adaptacji parametrów systemu. Osią działania systemu są algorytmy immunologiczne, które realizują procesy adaptacji i dywersyfikacji populacji. Opisany powyżej mechanizm może być przydatny w uczeniu FCM [54, 56, 118, 119], przy tym wykorzystuje się różne podejścia. Jedno z nich [54, 56] zakłada, że celem uczenia jest otrzymanie takich parametrów wektora wag, przy których czynniki FCM (po pobudzeniu układu określoną sekwencją sygnałów) osiągną zakładane wartości. Na bazie tych zakładanych wartości tworzy się docelowy wektor stanu $\mathbf{g} = [g_1, \dots, g_n]$ (n – liczba czynników). W trakcie uczenia czynniki będą przyjmować pewne wartości bieżące odwzorowywane przez atraktor wektora stanu $\mathbf{a} = [a_1, \dots, a_n]$. Zadaniem procesu uczenia jest minimalizacja funkcji celu, danej równaniem (3.124):

$$E(\mathbf{a}, \mathbf{g}) = (\mathbf{a} - \mathbf{g})\mathbf{D}(\mathbf{a} - \mathbf{g})^T \tag{3.124}$$

gdzie:

D – macierz diagonalna (jednostkowa) o wymiarach $n \times n$ (n – liczba czynników FCM).

Rolę pojedynczego przeciwciała pełni macierz wag FCM, a pojedyncza waga jest odpowiednikiem genu. Inicjacja następuje poprzez losowy dobór poszczególnych wag. Następnie oblicza się początkową wartość funkcji celu, po czym dokonuje się klonowania, uzyskując K jednakowych obiektów. Obiekty te (klony) poddaje się mutacji i krzyżowaniu z użyciem standardowych operatorów genetycznych, po czym wybiera spośród nich N najlepiej dopasowanych. Każdy z nich, drogą klonowania, powiela się, otrzymując populację K przeciwciał każdego z N rodzajów. Dalsze działania stanowią powielenie wyżej wymienionych kroków, które powtarza się aż do spełnienia warunku stopu, którym jest uzyskanie odpowiedniej wartości funkcji celu lub przekroczenie założonej liczby iteracji. Wynikiem końcowym jest przeciwciało o najlepszym dopasowaniu do antygenu (czyli macierz wag, przy której działanie modelu jest najbliższe założeniom). Dla zagwarantowania dostatecznej różnorodności przeciwciał, selekcja najlepszych odbywa się z uwzględnieniem pewnego prawdopodobieństwa, zgodnie z (3.125):

$$p_i = \alpha p_{f_i} + (1 - \alpha) p_{d_i} = \alpha \frac{f(i)}{\sum_{j=1}^{N} f(j)} + (1 - \alpha) \frac{1}{N} e^{-\beta C_i} \tag{3.125}$$

gdzie:

i – numer badanego przeciwciała;
α, β – stałe współczynniki dopasowujące;
$f(i)$ – dopasowanie i-tego przeciwciała;
C_i – współczynnik gęstości (zwartości) i-tego przeciwciała, obliczany zgodnie z (3.126):

$$C_i = \frac{1}{N}\sum_{i=1}^{N} c_{i,j} \tag{3.126}$$

gdzie:

$$c_{i,j} = \begin{cases} 1 & \text{dla } ac_{i,j} > T_{ac} \\ 0 & \text{w pozostałych przypadkach} \end{cases} \tag{3.127}$$

przy czym:
T_{ac} – współczynnik progowy;
$ac_{i,j}$ – stopień podobieństwa pomiędzy przeciwciałami i-tym i j-tym, wyznaczany według (3.128):

$$ac_{i,j} = \frac{1}{1 + H(2)} \tag{3.128}$$

gdzie:
$H(2)$ – entropia zbioru alleli wszystkich genów przeciwciała [56] (*allel* jest jedną z możliwych wersji genu utożsamianą często z wartością genu).

3.2.4.3. Metody hybrydowe

Hybrydowe metody uczenia FCM zwykle zawierają w sobie elementy obu wyżej wymienionych podejść (tzn. opartego o regułę Hebba oraz populacyjnego). Przeważnie kombinacja taka przybiera formę uczenia dwuetapowego, w którym łączy się efektywność metod Hebba (zwłaszcza NHL) z możliwościami metod populacyjnych w zakresie przeszukiwania globalnego.

Przykładem hybrydowej metody uczenia FCM może być **hybrydowy algorytm ewolucyjny** (ang. ***HEA – Hybrid Evolutionary Algorithm***) [138], w którym zastosowano dwa etapy uczenia FCM. W etapie pierwszym FCM (o czynnikach i wagach wstępnie zaprojektowanych w oparciu o wiedzę ekspertową) poddaje się działaniu algorytmu NHL w celu uzyskania zbieżnego modelu. Następnie, w etapie drugim, tak przygotowany model uczy się z użyciem populacyjnej metody DE, co umożliwia znalezienie optimum globalnego konfiguracji macierzy wag FCM. Podejście takie, wypróbowane na przykładach o różnej skali złożoności, pozwala uzyskać FCM działającą dokładniej niż uczona metodą NHL, a z drugiej strony czas poszukiwania rozwiązania jest krótszy niż gdyby stosować wyłącznie metodę DE. Tak więc, takie połączenie poprawia skuteczność procesu uczenia FCM.

W kolejnej metodzie [228] (którą można skrótowo nazwać ***RCGA-NHL***) zastosowano połączenie populacyjnej metody RCGA oraz algorytmu typu NHL, przy czym każdemu z tych algorytmów cząstkowych postawiono inne cele niż w podejściu prezentowanym przez HEA. Uczenie FCM odbywa się dwuetapowo. W etapie

pierwszym konstruuje się ogólną strukturę FCM w oparciu o opinie ekspertów, którzy określają liczbę i charakter czynników, następnie, na podstawie danych historycznych (sekwencji danych wejściowych uzyskanych z pomiarów lub z przekazanych informacji ekspertów), model poddaje się procedurze automatycznego uczenia techniką RCGA. Otrzymana tą drogą zgrubna postać modelu jest następnie douczana z użyciem nienadzorowanej metody NHL w celu poprawy zbieżności i lepszego dopasowania do potrzeb i założeń procesu modelowania. Podczas obu etapów uczenia wykorzystuje się standardowe techniki i operatory obu zaangażowanych metod uczenia. Podobnie jak w metodzie HEA takie połączenie technik pozwala uzyskać synergiczny efekt lepszy niż użycie każdej z technik oddzielnie.

Technika **uczenia w oparciu o gry** (ang. ***GBL – Game-based Learning***) [153] ma za zadanie podnieść poziom wiedzy „ucznia" poprzez jego uczestnictwo w specjalnie opracowanej grze. Motywacją do nauki jest chęć osiągnięcia sukcesu w rozwiązywaniu kolejnych zadań (pokonywaniu kolejnych poziomów) gry, co jest możliwe tylko na drodze dokładnego poznania zawartych w grze treści. Techniki tego rodzaju są coraz częściej wprowadzane do metodyk nauczania różnych treści. Podjęto też próby wykorzystania do tego celu modeli FCM [120]. Zasadniczo w metodzie tej potrzebny jest „nauczyciel", który będzie przewodnikiem „ucznia" w procesie uczenia. Ich właściwości odwzorowują dwa submodele FCM: nauczyciela i ucznia. Model nauczyciela jest rozszerzoną FCM, którą uczy się dwuetapowo. Na etapie uczenia lokalnego projektuje się czynniki i powiązania przyczynowe pomiędzy nimi, a także nadaje się wartości początkowe tym powiązaniom w oparciu o wiedzę ekspertową, po czym poddaje się submodel procesowi uczenia nienadzorowanego z użyciem algorytmu Hebba (np. DHL). Następnie uczenie submodelu nauczyciela przechodzi etap globalny, polegający na douczeniu na podstawie danych oczekiwanych (eksperymentalnych) z użyciem którejś z metod populacyjnych. Tak nauczony submodel FCM nauczyciela stanowi następnie bazę odniesienia dla submodelu FCM ucznia, który posiada takie same czynniki, jednakże wszystkie oddziaływania między tymi czynnikami są zerowe. Oba te modele są nakładane na scenariusz gry (w [120] gra polega na uczeniu się zasad prowadzenia samochodu w ruchu miejskim). Scenariusz ten generuje losowo różne sytuacje, na które uczeń powinien zareagować, wybierając określone działania z predefiniowanego zestawu. Wygenerowane sytuacje oraz działania wybierane przez ucznia są przetwarzane zarówno przez submodel FCM nauczyciela, jak i submodel FCM ucznia. Wyliczone przez system gry różnice w wartościach odpowiadających sobie czynników obu modeli służą do adaptacji parametrów submodelu FCM ucznia, a zarazem są podstawą do kierowania pod adresem ucznia nagród (w przypadku reakcji prawidłowych – zbliżających do siebie oba submodele) lub kar (w przypadku reakcji nieprawidłowych). Gra kończy się, gdy sumaryczny błąd wyznaczany na podstawie różnic wartości pomiędzy czynnikami submodeli spadnie poniżej wartości progowej lub gdy zostanie przekroczony czas gry. Metoda ta nie jest właściwie metodą uczenia FCM samą w sobie, stanowi natomiast przykład aplikacyjnego wykorzystania technik FCM w połączeniu z innymi technikami w celu stworzenia spójnego systemu kontrolnego.

Stosowanie podczas uczenia FCM mieszanych technik zawierających różne metody analizy danych nie zawsze musi mieć na celu poprawę efektywności czy też skrócenie czasu uczenia FCM. W niektórych sytuacjach problemem samym w sobie może być rozległość grafu FCM, rozumiana jako duża liczba węzłów (czynników) i wynikające z tego trudności w operowaniu takim modelem. Wtedy można podjąć próbę uproszczenia modelu poprzez redukcję liczby czynników. Jedno z podejść polega na zastosowaniu klasteryzacji czynników FCM [5]. Klasteryzacja jest procesem redukcji wymiaru danych poprzez podział licznego zbioru obiektów na grupy o podobnych cechach, zwane klastrami [52]. Klaster jest zbiorem obiektów, które są podobne do innych obiektów własnego klastra i jednocześnie różnią się od obiektów należących do innych klastrów. Pojedynczy klaster może być traktowany jak jedna grupa danych i dlatego klasteryzacja jest formą kompresji danych. Z punktu widzenia uczenia maszynowego klasteryzacja jest jedną z technik uczenia nienadzorowanego. Procedura przydziału danego elementu do danego klastra wykorzystuje pewną miarę odległości pomiędzy węzłami (podobna miara była stosowana w jednej z wczesnych metod automatycznego uczenia FCM [165], gdzie badano m.in. stopień podobieństwa pomiędzy czynnikami). Punktem wyjścia jest wstępnie zdefiniowana FCM, w której określono czynniki i ich wzajemne powiązania oraz przeprowadzono uczenie nadzorowane dowolną metodą populacyjną. Następnie przeprowadza się klasteryzację dzieląc czynniki na grupy elementów o podobnym zachowaniu. Można w tym celu posłużyć się szeroko stosowaną **metodą DEMATEL** (ang. ***Decision-making Trial and Evolution Laboratory***) [38]. Powstałe w ten sposób klastry są mniej liczne niż pierwotne czynniki. Centra klastrów stają się nowymi, hipotetycznymi czynnikami nowego, prostszego modelu FCM.

Hybrydyzacja polega nie tylko na łączeniu technik samouczenia (Hebba) i populacyjnych. Można spotkać również podejścia, które łączą w sobie techniki należące do jednej klasy. Należy do nich np. **hybrydowa memetyczna metoda optymalizacji rojem cząstek** (ang. ***MPSO – Memetic Particle Swarm Optimization***) [148, 149]. Stanowi ona pewną modyfikację metody PSO dokonaną poprzez jej uzupełnienie o lokalny algorytm memetyczny (ang. *MA – Memetic Algorithm*), który zastosowano do rozwiązania problemu optymalizacji (3.104). Algorytm memetyczny może być zaliczony do grupy algorytmów ewolucyjnych, przy czym, zamiast genów, którymi operują algorytmy genetyczne, występują w nim tzw. memy (zwane również jednostkami imitacji), które są nośnikami tzw. informacji kulturowej i rzeczywiście pierwotnie były wykorzystywane przez badaczy kultury jako jednostki idei, przekonań czy wzorców zachowań [29, 132]. Później jednak metodę tę zaczęto wykorzystywać do rozwiązywania problemów optymalizacji lokalnej w innych dziedzinach [111], w szczególności w połączeniu z innymi, globalnymi technikami optymalizacji. Zasadnicza różnica pomiędzy algorytmami genetycznymi a memetycznymi polega na tym, że o ile geny są dziedziczone przez potomków (zmiany pojawiają się w kolejnych populacjach), o tyle memy mogą być modyfikowane w obrębie tej samej populacji, czyli algorytm memetyczny

może działać „pomiędzy" generowaniem nowych populacji, a więc może na przykład lokalnie poprawiać dopasowanie populacji przed podjęciem kolejnych kroków przez nadrzędny algorytm optymalizacji globalnej. Zależnie od charakteru problemu może on działać przed generowaniem kolejnej populacji, w fazie oceny dopasowania (przed selekcją), po zakończeniu mutacji i krzyżowania lub po zakończeniu przeszukiwania ewolucyjnego w celu poprawy wyniku końcowego. Istnieją różne podejścia do samej techniki adaptacji parametrów populacji i dwa z nich [60, 177] wykorzystano [148, 149] (po zakończeniu każdego ewolucyjnego cyklu PSO) do zwiększenia efektywności uczenia FCM metodą PSO.

Przedstawione w tym podrozdziale zestawienie obejmuje szereg metod adaptacji rozmytych map kognitywnych, które, przy całej ich różnorodności, łączy jedna wspólna cecha: wszystkie one operują na wektorze stanu reprezentującym (liczbowo) wartości czynników oraz na macierzy wag reprezentującej (też liczbowo) charakter powiązań pomiędzy czynnikami. Podejmowano, co prawda, próby częściowego wprowadzenia liczb rozmytych do modelu FCM [24], jak i implementacji pewnego rodzaju relacji (w miejsce powiązań przyczynowych) [23], jednakże nie odniosły one większego sukcesu. Tak więc, w rzeczywistości operuje się na modelu, który nie jest modelem rozmytym, a przypomina raczej model relacyjnej mapy kognitywnej, omówiony w rozdziale 2. Również działanie większości modeli FCM uczonych tymi metodami opiera się nie na zbiorze reguł typu chociażby (3.1), a na zależnościach liczbowych, takich jak np. (3.80). Oczywiście, po zakończeniu procesu uczenia wartości liczbowe oddziaływań przyczynowych mogą zostać przekonwertowane na łańcuchowe wartości lingwistyczne, co przywróci im walor rozmycia, jednakże, po pierwsze, taka konwersja pogorszy dokładność (zwłaszcza blisko granic zakresów poszczególnych wartości lingwistycznych), a po drugie, przekształcenie modelu typu (3.80) w model z funkcjami typu (3.1) może zmienić jego charakter w stopniu poddającym w wątpliwość sens wcześniejszego uczenia. Z tego powodu tym większego znaczenia nabiera potrzeba poszukiwania takich podejść do budowy rozmytych map kognitywnych, w których możliwe będzie rachunkowe znalezienie rozwiązania i które jednocześnie pozwolą zachować zalety wynikające z rozmywania wartości parametrów. Obie te cechy posiada zaproponowana w dalszej części pracy metoda budowy modeli tzw. Relacyjnych Rozmytych Map Kognitywnych.

3.3. RELACJE ROZMYTE

Relacja jest niezwykle istotnym składnikiem matematyki dyskretnej, który umożliwia zestawianie ze sobą elementów różnych zbiorów, przez co ułatwia definiowanie wzajemnych powiązań pomiędzy zbiorami. Relacje są powszechnie używane w teorii zbiorów [156], a ponieważ teoria zbiorów rozmytych jest jej integralną częścią, to również relacje rozmyte [26, 88, 121, 158] odgrywają istotną rolę w operacjach logiki rozmytej.

W ogólnym ujęciu relacją dwuargumentową R dwóch zbiorów A i B nazywa się podzbiór produktu kartezjańskiego $A \times B$, którego elementami są pary uporządkowane (a, b) takie, że $a \in A$ i $b \in B$ [156]. Jeśli R jest relacją w zbiorze $A \times A$, to mówi się, że R jest relacją w zbiorze A. Fakt przynależności pary elementów do relacji oznacza się jak w (3.129):

$$(a, b) \in R \tag{3.129}$$

Relacja rozmyta $R_{A,B}$ pomiędzy dwoma nierozmytymi zbiorami $A = \{a\}$ oraz $B = \{b\}$ może być definiowana jako zbiór rozmyty określony w iloczynie kartezjańskim $A \times B$ zgodnie z (3.130) [88]:

$$R_{A,B} = \{\mu_R(a, b), (a, b)\} = \left\{\frac{\mu_R(a,b)}{(a,b)}\right\} \text{ dla każdego } (a, b) \in A \times B \tag{3.130}$$

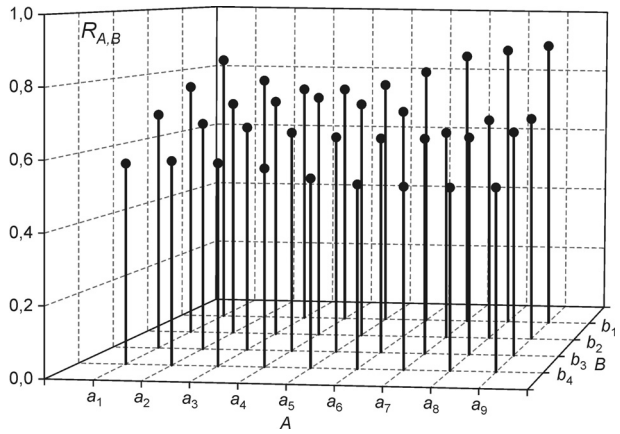
gdzie:
$\mu_R(a, b)\colon A \times B \longrightarrow [0, 1]$ – funkcja przynależności relacji rozmytej $R_{A,B}$;
$\mu_R(a, b) \in [0, 1]$ – stopień, w jakim elementy $a \in A$ i $b \in B$ są ze sobą w relacji $R_{A,B}$.

Relacja zdefiniowana przez (3.130) jest określona w iloczynie kartezjańskim dwóch zbiorów, stąd ten typ relacji nazywa się binarnym. Możliwe są też relacje rozmyte określone w iloczynie kartezjańskim większej liczby zbiorów.

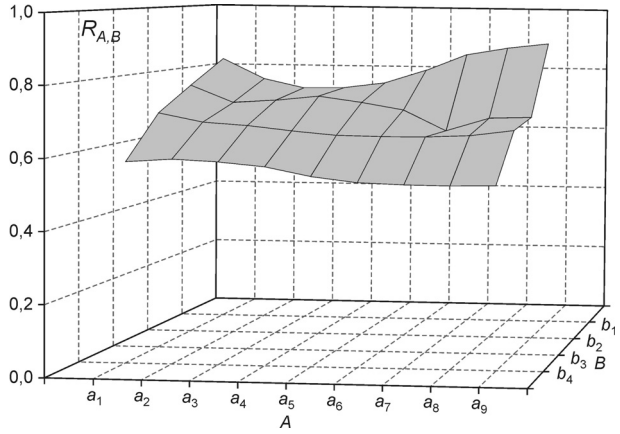
Relacja rozmyta przyporządkowuje każdej parze elementów ze zbiorów A i B stopień, w jakim elementy te są ze sobą powiązane, przy czym wartość 0 oznacza brak powiązania, a wartość 1 – pełną zgodność. Dla zbiorów skończonych relację rozmytą można przedstawić w postaci macierzowej (3.131) [158]:

$$R_{A,B} = \begin{array}{c} \\ (a_1) \\ (a_2) \\ \vdots \\ (a_k) \end{array} \begin{array}{c} \begin{array}{cccc} (b_1) & (b_2) & \cdots & (b_l) \end{array} \\ \begin{bmatrix} \mu_R(a_1, b_1) & \mu_R(a_1, b_2) & \cdots & \mu_R(a_1, b_l) \\ \mu_R(a_2, b_1) & \mu_R(a_2, b_2) & \cdots & \mu_R(a_2, b_l) \\ \vdots & \vdots & \ddots & \vdots \\ \mu_R(a_k, b_1) & \mu_R(a_k, b_2) & \cdots & \mu_R(a_k, b_l) \end{bmatrix} \end{array} \tag{3.131}$$

Działanie relacji rozmytej można lepiej zrozumieć poprzez jej reprezentację graficzną, przy czym możliwe są różne sposoby prezentacji, np. w postaci słupków bądź punktów (rys. 3.15) lub w postaci wykresu powierzchniowego (rys. 3.16).



Rys. 3.15. *Graficzna reprezentacja relacji rozmytej $R_{A,B}$ między dwoma zbiorami: $A = \{a_1,\ a_2, \dots, a_9\}$ i $B = \{b_1, b_2,\ b_3, b_4\}$ – w postaci punktów określających stopnie relacji*



Rys. 3.16. *Graficzna reprezentacja relacji rozmytej $R_{A,B}$ między dwoma zbiorami: $A = \{a_1,\ a_2, \dots, a_9\}$ i $B = \{b_1, b_2,\ b_3, b_4\}$ (analogicznej do przedstawionej na rys. 3.15) – w postaci wykresu powierzchniowego*

Jak wspomniano wcześniej, przy pomocy relacji (w tym również rozmytych), określa się wzajemne powiązania pomiędzy zbiorami (również zbiorami rozmytymi). Niezależnie od tego, relacji rozmytej dotyczą własności i operacje odnoszące się do zbiorów rozmytych (ponieważ relacja rozmyta jest zbiorem rozmytym). Oprócz nich istnieją operacje (m.in. [87, 93]), które są specyficzne dla relacji rozmytych i przy tym są kluczowe dla praktycznego wykorzystania takich relacji. Do najbardziej reprezentatywnych można zaliczyć następujące:

1. Złożenie maksyminowe dwóch relacji rozmytych.

Złożenie maksyminowe dwóch relacji rozmytych: R w $A \times B$ i S w $B \times C$ oznacza się jako $R \circ_{max-min} S$. Definiuje ono następującą relację rozmytą (3.132) [87, 88, 223]:

$$\underset{a \in A}{\forall} \underset{c \in C}{\forall} \mu_{R \circ_{max-min} S}(a, c) = \max_{b \in B}[\mu_R(a, b) \wedge \mu_S(b, c)] \tag{3.132}$$

gdzie:
R, S – relacje rozmyte;
A, B, C – zbiory objęte relacjami R i S; $A = \{a\}$, $B = \{b\}$, $C = \{c\}$.

Warto zauważyć, że operacja złożenia maksyminowego relacji rozmytych charakteryzuje się [35] takimi własnościami, jak:
– łączność:

$$R_1 \circ (R_2 \circ R_3) = (R_1 \circ R_2) \circ R_3 \tag{3.133}$$

– monotoniczność:

$$\text{jeżeli } R_1 \subseteq R_1' \text{ to } R_1 \circ R_2 \subseteq R_1' \circ R_2 \tag{3.134}$$

– rozdzielność względem sumy:

$$\mathrm{R}_1 \circ (\mathrm{R}_2 \cup \mathrm{R}_3) = (\mathrm{R}_1 \circ \mathrm{R}_2) \cup (\mathrm{R}_1 \circ \mathrm{R}_3) \tag{3.135}$$

– słaba rozdzielność względem przekroju:

$$R_1 \circ (R_2 \cap R_3) \subseteq (R_1 \circ R_2) \cap (R_1 \circ R_3) \tag{3.136}$$

gdzie:
R_1, R_1', R_2, R_3 – relacje rozmyte.

2. Złożenie maksyiloczynowe dwóch relacji rozmytych.

Złożenie maksyiloczynowe dwóch relacji rozmytych: R w $A \times B$ i S w $B \times C$ oznacza się jako $R \circ_{max-prod} S$. Definiuje ono następującą relację rozmytą (3.137) [87, 88, 93]:

$$\underset{a \in A}{\forall} \underset{c \in C}{\forall} \mu_{R \circ_{max-prod} S}(a, c) = \max_{b \in B}[\mu_R(a, b) \cdot \mu_S(b, c)] \tag{3.137}$$

gdzie:
R, S – relacje rozmyte;
A, B, C – zbiory objęte relacjami R i S; $A = \{a\}$, $B = \{b\}$, $C = \{c\}$.

3. Iloczyn kartezjański dwóch zbiorów rozmytych.

Iloczyn kartezjański dwóch zbiorów rozmytych: A w U i B w V jest relacją rozmytą oznaczaną jako $A \times B$ i definiowaną zgodnie z (3.138) [88]:

$$\underset{a\in A}{\forall}\ \underset{b\in B}{\forall}\ \mu_{A\times B}(a,b) = [\mu_A(u) \wedge \mu_B(v)] \tag{3.138}$$

lub według tożsamego zapisu (3.139):

$$\underset{a\in A}{\forall}\ \underset{b\in B}{\forall}\ \mu_{A\times B}(a,b) = min[\mu_A(u), \mu_B(v)] \tag{3.139}$$

gdzie: $A = \{a\}$, $B = \{b\}$.

4. Kompozycja zbioru rozmytego i relacji rozmytej.

Kompozycją zbioru rozmytego A określonego na przestrzeni rozważań U oraz binarnej relacji rozmytej R określonej na iloczynie kartezjańskim dwóch przestrzeni rozważań $U \times V$ jest zbiór rozmyty $B = A \circ R$ określony na przestrzeni rozważań V, którego funkcję przynależności definiuje (3.140) [121]. Definicja ta wykorzystuje t-normę, co nadaje jej ogólny charakter. Konkretyzacja funkcji przynależności (3.140) następuje dopiero na etapie realizacji konkretnego projektu (zostanie to pokazane w rozdziale 4 – przy omawianiu działania relacyjnych rozmytych map kognitywnych).

$$\mu_B(v) = \sup_{u\in U}[\mu_A(u)\ t\ \mu_R(u,v)] \tag{3.140}$$

3.4. ARYTMETYKA LICZB ROZMYTYCH

Liczby rozmyte odgrywają w modelach rozmytych rolę podobną, jak liczby nierozmyte w modelach konwencjonalnych. Wraz z nimi powstał aparat matematyczny służący do ich obsługi, który charakteryzuje się specyficznymi cechami, właściwymi dla zbiorów rozmytych. Liczba rozmyta jest zbiorem rozmytym, którego przestrzenią rozważań najczęściej jest dziedzina liczb rzeczywistych i który posiada pewne cechy szczególne.

W ogólności, aby zbiór rozmyty A określony w zbiorze liczb rzeczywistych ($A \subseteq \mathbb{R}$), z funkcją przynależności (3.141):

$$\mu_A : \mathbb{R} \longrightarrow [0,1] \tag{3.141}$$

mógł być uznany za liczbę rozmytą, musi on spełniać następujące warunki [88, 104–107, 121]:

– zbiór A jest normalny – zgodnie z (3.13),

– zbiór A jest wypukły, czyli (3.142):

$$\underset{u,v\in\mathbb{R}}{\forall}\ \underset{\lambda\in[0,1]}{\forall}\ \mu_A(\lambda\cdot u + (1-\lambda)\cdot v) \geq \min\left(\mu_A(u), \mu_A(v)\right), \tag{3.142}$$

– μ_A jest funkcją przedziałami ciągłą, co oznacza, że funkcja ta ma skończoną liczbę skoków jednostkowych, czyli takich punktów u_0, w których:

$$\lim_{u \to u_0^-} \mu_A(u) \neq \lim_{u \to u_0^+} \mu_A(u) \tag{3.143}$$

Liczby rozmyte można podzielić na pewne klasy funkcjonalne. Podstawą tego podziału jest liczebność jądra (3.7) liczby rozmytej A:
– właściwe liczby rozmyte, dla których $|core(A)| = 1$;
– przedziały rozmyte, dla których $|core(A)|$ jest przedziałem ograniczonym;
– niewłaściwe liczby rozmyte, dla których $|core(A)|$ jest przedziałem nieograniczonym.

W większości zastosowań występują liczby rozmyte właściwe, których ogólną postać można zdefiniować przy pomocy następującej funkcji przynależności [121]:

$$\mu_A(u) = \begin{cases} f_L\left(\dfrac{c-u}{\alpha}\right) & \text{dla } u < c \\ 1 & \text{dla } u = c \\ f_P\left(\dfrac{u-c}{\beta}\right) & \text{dla } u > c \end{cases} \tag{3.144}$$

gdzie:
A – liczba rozmyta;
α, β – ustalone rozrzuty lewo- i prawostronny;
c – ustalona wartość średnia;
f_L, f_P – ustalone funkcje bazowe (funkcje odniesienia).

Spotyka się również określenie „liczba rozmyta" odnoszące się do zbioru rozmytego o wysokości mniejszej niż 1, ale takie podejście nie będzie stosowane w niniejszej pracy. Wypada nadmienić, że tego typu zbiór rozmyty może być poddany normalizacji (np. metodą zgodną z (3.145)), co przekształci go w liczbę rozmytą jednej z wyżej wymienionych klas.

$$\underset{u \in U}{\forall} \mu_A(u) = \frac{\mu_{\hat{A}}(u)}{\mathrm{h}(\hat{A})} \tag{3.145}$$

gdzie:
$\mu_{\hat{A}}(u)$ – funkcja przynależności zbioru rozmytego $\hat{A}$ określonego w U;
$\mu_A(u)$ – funkcja przynależności wynikowego zbioru A określonego w U, będącego wynikową liczbą rozmytą.

W dalszych rozważaniach będą brane pod uwagę jedynie właściwe liczby rozmyte.

Liczby rozmyte można grupować również według innych kryteriów, np. według znaku [158] na:
- liczby dodatnie, w których: $\mu_A(u) = 0$ dla $u \leq 0$;
- liczby ujemne, w których: $\mu_A(u) = 0$ dla $u \geq 0$;
- liczby mieszane, którymi są liczby niebędące ani dodatnimi, ani ujemnymi.

W dalszych rozważaniach będą brane pod uwagę jedynie dodatnie liczby rozmyte.

Teoretyczne modele algebry rozmytej [104, 121, 158, 224] wprowadzają, oprócz definicji liczb rozmytych, również specyficzne operacje, będące odpowiednikami operacji na liczbach rzeczywistych. Ich wykonywanie jest oparte na tzw. zasadzie rozszerzania [88, 121, 158, 223].

3.4.1. Zasada rozszerzania

Zasada rozszerzania została wprowadzona przez L. Zadeha w 1973 r. [223]. Jest to jedno z najważniejszych narzędzi obróbki zbiorów rozmytych. Jej poniższe krótkie omówienie zamieszczono w rozdziale dotyczącym liczb rozmytych z uwagi na charakter późniejszych rozważań, tym niemniej należy pamiętać, że zastosowanie zasady rozszerzania nie ogranicza się jedynie do liczb rozmytych. Zasada rozszerzania określa sposób definiowania zależności pomiędzy wielkościami rozmytymi tak, aby zależności te były równoważne swoim nierozmytym odpowiednikom.

Definicja zasady rozszerzania opiera się na założeniu istnienia pewnego ostrego odwzorowania f przestrzeni U w przestrzeń V:

$$f : U \longrightarrow V, \tag{3.146}$$

które może oddziaływać również na zbiory rozmyte.

Jeśli U będzie iloczynem kartezjańskim zbiorów dystrybutywnych $U_1, \ldots, U_k$, to (3.146) można zapisać jako:

$$f : U_1 \times \cdots \times U_k \longrightarrow V \tag{3.147}$$

gdzie:
$U_1 = \{u_1\}$, ..., $U_k = \{u_k\}$;
f – funkcja (nierozmyta) taka, że $v = f(u_1, \ldots, u_k)$.

Jeżeli dodatkowo przyjmie się, że $A_1, \ldots, A_k$ są zbiorami rozmytymi, określonymi w odpowiednio $U_1, \ldots, U_k$, to zasada rozszerzania definiuje zbiór rozmyty B określony w $V = \{v\}$, indukowany przez zbiory $A_1, \ldots, A_k$ za pomocą funkcji f, poprzez następującą definicję (3.148) [88, 158]:

$$\mu_B(v) = \sup_{(u_1,\ldots,u_k) \in U_1 \times \cdots \times U_k : v = f(u_1,\ldots,u_k)} \min_{i=1}^{k} \mu_{A_i}(u_i) \tag{3.148}$$

Zależność (3.148) stanowi podstawę do wykonywania operacji arytmetycznych na liczbach rozmytych (niektóre z nich pokazano dalej w równaniach (3.151)–(3.154)) oraz wielu innych operacji [94].

W szczególnym przypadku równania (3.146), kiedy A jest zbiorem rozmytym określonym na przestrzeni rozważań $U = \{u_1, \dots, u_k\}$, złożonym z k singletonów rozmytych, o postaci (3.5), oraz jeśli odwzorowanie f jest wzajemnie jednoznaczne, to, zgodnie z zasadą rozszerzania, zbiór rozmyty B określony w $V = \{v\}$, indukowany przez to odwzorowanie ma następującą postać:

$$B = f(A) = \frac{\mu_A(u_1)}{f(u_1)} + \dots + \frac{\mu_A(u_k)}{f(u_k)}. \tag{3.149}$$

Przy tym, jeśli więcej niż jeden element ze zbioru U jest odwzorowywany w ten sam element $v \in V$, to stopień przynależności nowego elementu do zbioru $B = f(A)$ jest równy maksymalnemu ze stopni przynależności rozważanych elementów ze zbioru U.

Zależność typu (3.149) można przedstawić przy pomocy równoważnego zapisu α-przekrojów o postaci (3.9). Wtedy zbiór rozmyty B w V, indukowany przez A za pomocą f przybierze postać (3.150):

$$B = f(A) = f\left(\sum_{\alpha \in (0,1]} \alpha \cdot A_\alpha\right) = \sum_{\alpha} \alpha \cdot f(A_\alpha) \tag{3.150}$$

gdzie:
A_α – α-przekroje zbioru rozmytego A w U.

3.4.2. Podstawowe operacje arytmetyczne na liczbach rozmytych

Liczby rozmyte podlegają regułom arytmetycznym podobnym jak liczby rzeczywiste. Przy tym, poszczególne operacje arytmetyczne na tych liczbach są definiowane w oparciu o regułę rozszerzenia (3.148). Poniżej przedstawiono sformułowania głównych operacji arytmetycznych na liczbach rozmytych:

a) dodawanie rozmyte (operator $\oplus$):

$$\mu_{A \oplus B}(u) = \sup_{\substack{u_1, u_2 \\ u = u_1 + u_2}} \min\{\mu_A(u_1), \mu_B(u_2)\} \tag{3.151}$$

b) odejmowanie rozmyte (operator $\ominus$):

$$\mu_{A \ominus B}(u) = \sup_{\substack{u_1, u_2 \\ u = u_1 - u_2}} \min\{\mu_A(u_1), \mu_B(u_2)\} \tag{3.152}$$

c) mnożenie rozmyte (operator $\odot$):

$$\mu_{A\odot B}(u) = \sup_{\substack{u_1,u_2 \\ u=u_1\cdot u_2}} \min\{\mu_A(u_1), \mu_B(u_2)\} \tag{3.153}$$

d) dzielenie rozmyte (operator $\oslash$):

$$\mu_{A\oslash B}(u) = \sup_{\substack{u_1,u_2 \\ u=u_1/u_2}} \min\{\mu_A(u_1), \mu_B(u_2)\} \tag{3.154}$$

gdzie:

A, B – liczby rozmyte;

$\mu_A(u)$ – wartość funkcji przynależności liczby rozmytej A w punkcie u przestrzeni rozważań $\mathbb{R}$;

$\mu_B(u)$ – wartość funkcji przynależności liczby rozmytej B w punkcie u przestrzeni rozważań $\mathbb{R}$.

Warto zauważyć, że zgodnie z twierdzeniem Duboisa-Prade'a [31], jeśli A i B są ciągłymi liczbami rozmytymi, których funkcje przynależności są surjekcjami, wyniki działań (3.151)–(3.154) również są ciągłymi liczbami rozmytymi.

Operacje (3.151)–(3.154) dają wyniki (po wyostrzeniu) dokładnie odpowiadające podobnym operacjom skalarnym pod warunkiem zachowania pełnej symetrii liczby wokół jej centrum oraz pełnej ciągłości funkcji przynależności (nieskończenie dużej liczby wartości lingwistycznych). W praktyce działania te są wykonywane technikami numerycznymi, co wymusza dyskretyzację liczb i ograniczenie bazowego uniwersum. Należy również pamiętać, że liczby rozmyte mają w pewnym sensie dualny charakter i mogą być traktowane jako składniki arytmetyki rozmytej lub też jako zbiory singletonów rozmytych. W pewnych sytuacjach wykorzystuje się oba te podejścia równocześnie. W konsekwencji pojawia się szereg utrudnień i niedokładności, które należy uwzględniać przy budowie modeli rozmytych.

Oprócz opisanych powyżej operacji dwuargumentowych, definiuje się również operacje jednoargumentowe na liczbach rozmytych [26, 121, 158]:

a) przeciwieństwo liczby rozmytej A, oznaczane jako $-A$, definiowane równaniem (3.155):

$$\mu_{-A}(u) = \mu_A(-u) \tag{3.155}$$

b) odwrotność liczby rozmytej A, oznaczana jako A^{-1}, definiowana równaniem (3.156):

$$\mu_{A^{-1}}(u) = \mu_A(u^{-1}) \tag{3.156}$$

Operacja odwrotności jest definiowana jedynie dla liczb rozmytych dodatnich i ujemnych (gdyby liczba rozmyta A była mieszana, to zbiór wynikowy A^{-1} nie byłby zbiorem wypukłym, czyli liczbą rozmytą);

c) wartość bezwzględna liczby rozmytej A, oznaczana jako $|A|$, definiowana równaniem (3.157):

$$\mu_{|A|}(u) = \begin{cases} \max(\mu_A(u), \mu_A(-u)) & \text{dla } u \geq 0 \\ 0 & \text{dla } u < 0 \end{cases} \tag{3.157}$$

d) eksponent liczby rozmytej A, oznaczany jako e^A, definiowany równaniem (3.158):

$$\mu_{e^A}(u) = \begin{cases} \mu_A(\ln(u)) & \text{dla } u > 0 \\ 0 & \text{dla } u < 0 \end{cases} \tag{3.158}$$

e) skalowanie liczby rozmytej A, oznaczane jako λA, definiowane równaniem (3.159):

$$\mu_{\lambda A}(u) = \mu_A(u\lambda^{-1}) \tag{3.159}$$

Projektując system wykorzystujący arytmetykę liczb rozmytych, należy pamiętać, że operacje na liczbach rozmytych są, co prawda, odpowiednikami operacji na liczbach rzeczywistych, jednak nie odwzorowują wszystkich ich własności. Dodawanie i mnożenie rozmyte są łączne i przemienne, jednakże liczba rozmyta A nie posiada liczby przeciwnej względem dodawania (czyli: $A \oplus (-A) \neq 0$) ani odwrotnej względem mnożenia (czyli: $A \odot (A^{-1}) \neq 1$).

4 MODELOWANIE ZŁOŻONYCH SYSTEMÓW Z UŻYCIEM RELACYJNEJ ROZMYTEJ MAPY KOGNITYWNEJ

Celem rozpoczętych w 2009 r. [79, 80, 81] prac nad teorią relacyjnych rozmytych map kognitywnych była potrzeba uzupełnienia istniejącej metodologii rozmytych map kognitywnych o mechanizmy pozwalające w maksymalnym stopniu zautomatyzować procesy projektowania i funkcjonowania modelu. Główna metoda tworzenia relacji pomiędzy czynnikami, opierająca się na regułach typu IF-THEN (bardziej szczegółowo została ona omówiona w podrozdziale 3.2), chociaż stosunkowo prosta implementacyjnie, stwarza wiele problemów w sytuacjach, w których konieczne jest dokonanie jakiejś modyfikacji w systemie (np. zmiana liczby czynników lub zmiana układu wartości lingwistycznych któregoś z nich), jak również na etapie uczenia modelu. Reguły tego typu buduje się w oparciu niemal wyłącznie o wiedzę ekspertową, czyli ich zestawy są tworzone odrębnie dla każdego czynnika na podstawie obserwacji, co jest czynnością bardzo pracochłonną i narażoną na błędy. Praktycznie każda zmiana w konfiguracji czynników wymusza powtarzanie prawie całej procedury tworzenia modelu od początku. Równie uciążliwy jest proces uczenia mapy, czyli dostosowywania zestawów relacji do pracy rzeczywistego systemu, będącego przedmiotem modelowania. Z drugiej strony wprowadzone później podejścia do działania modeli FCM, polegające na operowaniu liczbowymi odpowiednikami zarówno wartości czynników, jak i powiązań, zaprzeczają idei rozmywania, a późniejsze transformowanie takich liczbowych rozwiązań w przestrzeń wielkości rozmytych musi pogarszać dokładność modeli. Niezależnie od tego, angażowanie wiedzy ekspertowej (czyli grupy ekspertów) na etapie projektowania stwarza ryzyko wprowadzenia dodatkowych błędów. W istniejących metodach zbieranie wiedzy ekspertowej polega po prostu na przeprowadzeniu ankiety wśród pewnej liczby osób. Co prawda dość wcześnie przewidziano mechanizmy zmniejszające znaczenie lub wręcz eliminujące opinie szczególnie odbiegające od średniej [30], ale zagrożenie pozostało. O ile można założyć, że eksperci prawidłowo wyodrębnią czynniki kluczowe dla celów modelowania, o tyle brak pewności, że (zwłaszcza, jeśli będzie ich niewielu) poprawnie oszacują liczbę wartości lingwistycznych opisujących stany czynników i powiązań przyczynowych.

Z powyższych powodów podjęto prace nad wprowadzeniem nowego podejścia do tworzenia i eksploatacji modeli rozmytych map kognitywnych. W założeniach podejście to miało umożliwić opracowanie ogólnych postaci arytmetycznych głównych elementów modelu oraz uogólnionych algorytmów budowy i uczenia takiego modelu, które pozwoliłyby zachować rozmyty charakter środowiska i byłyby w minimalnym tylko stopniu narażone na błędy interpretacyjne eksperta. Rozwiązaniem okazało się zaimplementowanie reguł arytmetyki rozmytej, co spowodowało kolejne trudności związane m.in. z projektowaniem odpowiednich teoretycznych formuł oddających związki pomiędzy czynnikami. W wyniku kolejnych badań [64, 73, 75, 76, 78, 168, 169, 171, 174, 176] znale-

ziono szereg rozwiązań ułatwiających tworzenie modeli rozmytych map kognitywnych wykorzystujących liczby rozmyte i ich algebrę. Głównym celem było opracowanie uogólnionego sposobu odwzorowywania powiązań pomiędzy czynnikami za pomocą relacji rozmytych (stąd nazwa typu modelu: **relacyjna** rozmyta mapa kognitywna) oraz odpowiedniego wykorzystania tych relacji do modelowania dynamiki pracy modelu. Opracowano także podejście do nadzorowanego uczenia takich modeli, co umożliwia ich stosowanie do monitorowania wielu rodzajów systemów o nieprecyzyjnej informacji. W dalszych podrozdziałach przedstawiono zasady budowy, działania i uczenia modelu relacyjnej rozmytej mapy kognitywnej oraz działanie komputerowej aplikacji stworzonej na potrzeby projektowania i testowania takiego modelu.

4.1. OGÓLNA POSTAĆ RELACYJNEJ ROZMYTEJ MAPY KOGNITYWNEJ

Ogólna postać rozmytej relacyjnej mapy kognitywnej jest tożsama z ogólną postacią relacyjnej (ostrej) mapy kognitywnej, opisanej przez (2.2), czyli:

$$< \mathbf{X}, \mathbf{R} > \tag{4.1}$$

z tym, że w (4.1):

$\mathbf{X} = [X_1, X_2, \dots, X_n]^T$ – wektor rozmytych wartości czynników, określonych jako liczby rozmyte;

$\mathbf{R} = \{R_{i,j}\}, (i, j = 1, \dots, n)$ – macierz relacji rozmytych między rozmytymi czynnikami X_i i X_j (parametry mapy).

Graficzna reprezentacja relacyjnej rozmytej mapy kognitywnej ma postać jak na rysunku 2.3, przy czym występujące na rysunku 2.3 relacje $r_{i,j}$ są zastąpione przez relacje rozmyte $R_{i,j}$.

Podobnie podstawowe modele relacyjnych rozmytych map kognitywnych odpowiadają podstawowym modelom relacyjnych (ostrych) map kognitywnych (2.9)–(2.13). Przedstawione dalej szczegółowe rozważania na temat budowy i działania relacyjnych rozmytych map kognitywnych zostaną ograniczone tylko do jednego z tych modeli: nieliniowego z nieliniowością szybkości zmian (postać ostrą analogicznego modelu relacyjnej mapy kognitywnej opisuje równanie (2.12)). Wybór taki został podyktowany przydatnością tej postaci relacyjnej mapy kognitywnej do modelowania systemów charakteryzujących się wewnętrzną dynamiką, które są często spotykane w praktyce.

Nieliniowy model relacyjnej rozmytej mapy kognitywnej, uwzględniający nieliniową szybkość zmian wartości czynników, działa w oparciu o zasadę zilustrowaną przez (4.2):

$$X_j(t+1) = f_p\left(X_j(t) \oplus \bigoplus_{\substack{i=1 \\ i \neq j}}^{n} \left[\left(X_i(t) \ominus X_i(t-1) \right) \circ R_{i,j} \right] \right) \tag{4.2}$$

gdzie:

$X_j(t)$ – wartość rozmyta rozważanego (j-tego) czynnika w chwili t czasu dyskretnego;

$R_{i,j}$ – relacja rozmyta pomiędzy czynnikami o numerach i oraz j;

f_p – funkcja progowa, analogiczna do funkcji opisanych w (2.14)–(2.16);

n – liczba czynników; $i, j = 1, \ldots, n$.

Problemy związane z rozmywaniem czynników i relacji oraz operowaniem ich rozmytymi wartościami zostaną poruszone w dalszej części rozdziału. Na tym etapie należy zaznaczyć, że bezpośrednie stosowanie funkcji progowej f_p w odniesieniu do modeli wykorzystujących liczby rozmyte (m.in. typu (4.2)), chociaż formalnie możliwe (np. w oparciu o (3.149)), wydaje się niecelowe. Jak zostanie to pokazane później, w modelach tego rodzaju kluczowe znaczenie ma zachowanie niezmiennego zbioru punktów przestrzeni rozważań U, w której opisuje się rozmyte wartości czynników i relacje rozmyte. Odwzorowania funkcyjne oparte na zasadzie rozszerzenia (takie jak (3.149)) deformują zbiór tych punktów, co skutkuje koniecznością wprowadzania dodatkowych procedur stabilizujących, a to z kolei może mieć wpływ na dokładność modelowania. Zamiast wprowadzania funkcji progowej można stosować prostsze zabezpieczenia, związane np. z odpowiednią metodą normalizacji wartości czynników. Dzięki temu model (4.2) przybierze prostszą postać (4.3) i w takiej postaci będzie występował dalej:

$$X_j(t+1) = X_j(t) \oplus \bigoplus_{\substack{i=1 \\ i \neq j}}^{n} \left[\left(X_i(t) \ominus X_i(t-1) \right) \circ R_{i,j} \right] \tag{4.3}$$

Dodatkowym uzasadnieniem dla podejścia (4.3) jest fakt, że algorytm uczenia, w którym wykorzysta się odpowiednio liczny zbiór uczący (a wartości czynników w takim zbiorze zawsze mieszczą się w zakresie $[-1, 1]$) samoistnie dobierze parametry relacji rozmytych, przy których wyostrzone wartości czynników w zasadzie nie będą przekraczały dopuszczalnego zakresu.

4.2. NOŚNIK RELACYJNEJ ROZMYTEJ MAPY KOGNITYWNEJ

W przedstawianych dotychczas definicjach związanych z teorią zbiorów rozmytych używano pojęcia przestrzeni rozważań (uniwersum) U, w której określano zbiór rozmyty (lub liczbę rozmytą) A. W ogólnym przypadku przestrzeń U ma charakter ciągły i może być stosowana w rozważaniach teoretycznych, jednakże w zastosowa-

niach praktycznych większe znaczenie ma nośnik (baza) liczby rozmytej supp(A) (zdefiniowany równaniem (3.6)), który jest dystrybutywnym zbiorem tych elementów przestrzeni U, dla których wartość funkcji przynależności liczby rozmytej A jest większa od zera. Dla niektórych klas funkcji przynależności (jak np. klasa *s* zdefiniowana równaniem (3.33)) zakres nośnika sięga nieskończoności, co oczywiście utrudnia jego praktyczne stosowanie. Nawet jeśli ten zakres jest ograniczony formalnie (jak dla klasy π określonej przez (3.35)), to i tak nośnik określony na ciągłej przestrzeni ma ciągły charakter, czyli zawiera nieskończoną liczbę elementów. Z tego powodu w praktycznych zastosowaniach (do których używa się komputerowych algorytmów obliczeniowych) wygodnie jest przyjąć, że liczby rozmyte są definiowane jako zbiory singletonów rozmytych (zgodnie z (3.5)). Jeśli przyjmie się, że w takiej notacji zwykle pomija się elementy, dla których funkcja przynależności ma zerową wartość, to nośnik liczby rozmytej można uznać za zbiór (nierozmyty) elementów odpowiadających singletonom rozmytym, czyli:

$$\text{supp}(A) = \{u_1, \dots, u_k\} \tag{4.4}$$

gdzie:
u_1, ..., u_k – punkty przestrzeni rozważań U występujące w równaniu (3.5).

Elementy nośnika liczby rozmytej są liczbami rzeczywistymi. Z punktu widzenia numerycznych algorytmów obliczeniowych dogodnie jest założyć, że odległości pomiędzy tymi elementami są jednakowe. Wtedy, idąc dalej tym tokiem rozumowania, można stwierdzić, że nośnikiem liczby rozmytej A jest pewien przedział:

$$\text{supp}(A) = [u_1, u_k], \tag{4.5}$$

próbkowany z określonym krokiem Δk:

$$\Delta k = \frac{u_k - u_1}{k - 1} \tag{4.6}$$

gdzie:
k – liczba punktów próbkowania nośnika odpowiadająca liczbie singletonów rozmytych liczby rozmytej;
$u_1, \dots, u_k$ – kolejne punkty nośnika oddalone od siebie o stałą odległość Δk.

W dalszej kolejności można rozszerzyć definicję nośnika (rozszerzenie to będzie aktualne jedynie w odniesieniu do liczb i relacji rozmytych) w taki sposób, aby objąć nim również te punkty przestrzeni rozważań, dla których wartość funkcji przynależności wynosi co prawda 0, ale należą one do pewnego bazowego przedziału wartości przestrzeni rozważań wybranego do singletonowej reprezentacji rozmytych liczb i relacji w danym modelu. Wtedy definicja nośnika z (4.5) przybierze zmodyfikowaną postać (4.7):

$$\overset{*}{\text{supp}}(A) = \{u_1, \dots, u_k\} \tag{4.7}$$

gdzie:

$A = \frac{\mu_A(u_1)}{u_1} + \cdots + \frac{\mu_A(u_k)}{u_k}$ – liczba rozmyta określona na zmodyfikowanym nośniku;

$u_1 < u_2 < \cdots < u_k$ – kolejne punkty próbkowania nośnika zmodyfikowanego;

$\forall_{i=1,\ldots,k-1}\, u_{i+1} - u_i = \Delta k$;

$\mu_A(u_1), \ldots, \mu_A(u_k) = [0, 1]$ – wartości funkcji przynależności liczby rozmytej A w kolejnych punktach próbkowania nośnika zmodyfikowanego.

W dalszych rozważaniach sformułowanie „nośnik" będzie oznaczało nośnik zmodyfikowany w sensie (4.7).

Relacyjna mapa kognitywna (również w formie rozmytej) operuje czynnikami, których wartości ostre (niezależnie od tego czy mają charakter ilościowy, czy jakościowy) poddaje się normalizacji według którejś z metod przedstawionych w rozdziale 2.1. Oznacza to, że znormalizowane (ostre) wartości czynników powinny mieścić się w zakresie $[0, 1]$ lub $[-1, 1]$ (zależnie od założonych celów modelowania). Narzuca to w pewnym sensie również planowanie zakresu nośnika liczb i relacji rozmytych, używanych w modelu relacyjnej rozmytej mapy kognitywnej. Zarówno wartości czynników, jak i relacje powinny zostać rozmyte na tym nośniku tak, aby późniejsze operacje arytmetyczne pozwoliły uzyskać możliwie dokładne wyniki. Jak zostanie pokazane w dalszej części pracy, zakres nośnika powinien umożliwić zachowanie maksymalnej symetrii przetwarzanych liczb rozmytych, ponieważ zależy od niej dokładność wyostrzania. Wynika stąd, że dla większości zastosowań zakres nośnika powinien być znacząco szerszy niż podane wyżej granice normalizacji. Z przeprowadzonych prób [171] wynika, że dla normalizacji do zakresu $[0, 1]$ wystarczającym zakresem nośnika jest $[-1, 2]$, natomiast w modelu wykorzystującym normalizację do zakresu $[-1, 1]$ wystarczy, aby nośnik miał zakres $[-2, 2]$. Dalsze rozszerzanie zakresu nośnika (potwierdzają to badania, ale jest to również zgodne z intuicyjnym rozumieniem tego pojęcia) pozwala na zwiększenie dokładności modelu, ale wzrost nakładu czasu komputerowej realizacji algorytmów obliczeniowych nie jest wystarczająco rekompensowany przez poprawę dokładności, w związku z czym można uznać takie działanie za bezcelowe.

Dla zachowania spójności modelu, wszystkie występujące w nim wartości czynników powinny być rozmywane na tym samym nośniku (4.7). Oznacza to, że wiążące je relacje rozmyte (w formie np. (3.131)) powinny mieć kwadratową podstawę, której bok zawiera k elementów (tyle samo ile wynosi liczność punktów próbkowania nośnika czynników rozmytych). Wynika stąd, że wszystkie wielkości rozmyte występujące w modelu relacyjnej rozmytej mapy kognitywnej (czynniki i relacje) powinny być rozmywane na tym samym nośniku. Jest to podstawowy warunek prawidłowego działania modelu.

4.3. OPERACJE ALGEBRY ROZMYTEJ W RELACYJNEJ ROZMYTEJ MAPIE KOGNITYWNEJ

Jak wspomniano wcześniej, praktyczne zastosowania relacyjnej rozmytej mapy kognitywnej wykorzystują środowisko dyskretne z uwagi na komputerowe algorytmy stosowane przy obliczeniach. W związku z tym zasada rozszerzania, określona równaniem (3.148), ulegnie pewnej modyfikacji mającej na celu dostosowanie jej do dyskretnego środowiska. Zmodyfikowana zasada rozszerzania przybierze następujący kształt (4.8):

$$\mu_B(v) = \max_{(u_1,\ldots,u_k)\in U_1\times\cdots\times U_k : v=f(u_1,\ldots,u_k)} \min_{i=1}^{k} \mu_{A_i}(u_i) \tag{4.8}$$

Modyfikacja zasady rozszerzania pociąga za sobą konieczność podobnego dostosowania definicji operacji na liczbach rozmytych ((3.151)–(3.154)). W kolejnych rozważaniach będą stosowane dwie takie operacje: dodawanie i odejmowanie, których zmodyfikowane definicje przedstawiają równania (4.9) i (4.10):

– dodawanie rozmyte (operator $\oplus$):

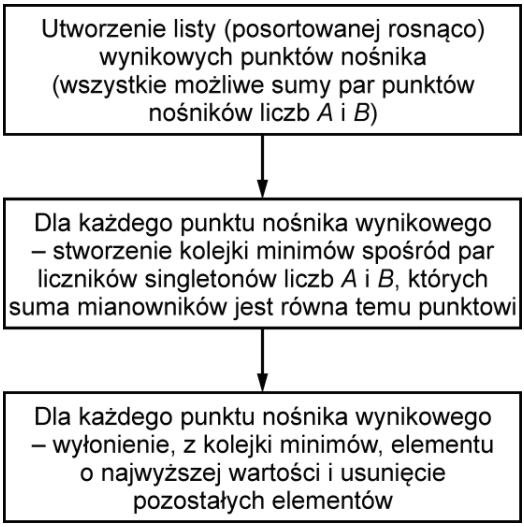
$$\mu_{A\oplus B}(u) = \max_{\substack{u_1,u_2 \\ u=u_1+u_2}} \min\{\mu_A(u_1), \mu_B(u_2)\} \tag{4.9}$$

– odejmowanie rozmyte (operator $\ominus$):

$$\mu_{A\ominus B}(u) = \max_{\substack{u_1,u_2 \\ u=u_1-u_2}} \min\{\mu_A(u_1), \mu_B(u_2)\} \tag{4.10}$$

Uwaga 4.1

Algorytmiczna realizacja operacji arytmetycznych na liczbach rozmytych jest dość kłopotliwa, ponieważ wymaga operowania strukturami danych, których rozmiary są trudne do przewidzenia (a to dlatego, że są zależne od założonej liczby punktów próbkowania nośnika, która z kolei może się zmieniać w procesie budowy modelu). Wynika stąd konieczność szerokiego stosowania dynamicznych struktur danych (takich jak kolejki i listy), których obsługa może być czasochłonna. Sama idea wykonywania takich operacji nie jest szczególnie skomplikowana (pokazano ją na rys. 4.1 – na przykładzie operacji dodawania rozmytego dwóch liczb rozmytych), jednak jej sprawna realizacja może napotykać trudności.



Rys. 4.1. *Ogólna postać algorytmu realizującego dodawanie dwóch liczb rozmytych A i B*

Dla lepszego zrozumienia sensu operatorów dodawania i odejmowania rozmytego przedstawiono poniżej wyniki ich działania na prostym przykładzie liczbowym.

Przykład 4.1

Dodawanie i odejmowanie dwóch liczb rozmytych A i B o tym samym nośniku o zakresie $[0, 1]$ (operatory + i − mają charakter mnogościowy).

$$A = \frac{0{,}68}{0{,}00} + \frac{1{,}00}{0{,}25} + \frac{0{,}68}{0{,}50} + \frac{0{,}21}{0{,}75} + \frac{0{,}03}{1{,}00}$$

$$B = \frac{0{,}03}{0{,}00} + \frac{0{,}21}{0{,}25} + \frac{0{,}68}{0{,}50} + \frac{1{,}00}{0{,}75} + \frac{0{,}68}{1{,}00}$$

$$A \oplus B = \frac{0{,}03}{0{,}00} + \frac{0{,}21}{0{,}25} + \frac{0{,}68}{0{,}50} + \frac{0{,}68}{0{,}75} + \frac{1{,}00}{1{,}00} + \frac{0{,}68}{1{,}25} + \frac{0{,}68}{1{,}50} + \frac{0{,}21}{1{,}75} + \frac{0{,}03}{2{,}00}$$

$$A \ominus B = \frac{0{,}68}{-1{,}00} + \frac{0{,}68}{-0{,}75} + \frac{1{,}00}{-0{,}50} + \frac{0{,}68}{-0{,}25} + \frac{0{,}68}{0{,}00} + \frac{0{,}21}{0{,}25} + \frac{0{,}21}{0{,}50} + \frac{0{,}03}{0{,}75} + \frac{0{,}03}{1{,}00}$$

Sposób tworzenia powyższych wyników zostanie zilustrowany (w oparciu o algorytm z rys. 4.1) na przykładzie pojedynczego singletonu rozmytego będącego elementem wyniku dodawania. Będzie nim singleton o mianowniku $0{,}75$. Poniżej przedstawiono kolejne etapy jego tworzenia:

a) do budowy singletonu wynikowego użyto par tych singletonów liczb A i B, których mianowniki dają sumę równą $0{,}75$:

$$\left(\frac{0{,}68}{0{,}00}\right)_A i \left(\frac{1{,}00}{0{,}75}\right)_B;\ \left(\frac{1{,}00}{0{,}25}\right)_A i \left(\frac{0{,}68}{0{,}50}\right)_B;\ \left(\frac{0{,}68}{0{,}50}\right)_A i \left(\frac{0{,}21}{0{,}25}\right)_B;\ \left(\frac{0{,}21}{0{,}75}\right)_A i \left(\frac{0{,}03}{0{,}00}\right)_B$$

b) w każdej parze wybrano licznik o niższej wartości, w wyniku czego w tymczasowym liczniku singletonu wynikowego powstała kolejka następujących elementów:

$$\frac{0{,}68\ ;\ 0{,}68\ ;\ 0{,}21\ ;\ 0{,}03}{0{,}75}$$

c) z elementów tymczasowego licznika wyłoniono jeden, o najwyższej wartości, a pozostałe usunięto i uzyskano docelową wartość singletonu rozmytego:

$$\frac{0{,}68}{0{,}75}$$

Uwaga 4.2

Należy zauważyć, że wykonanie któregoś z działań arytmetycznych (4.9), (4.10) skutkuje zmianą rozpiętości oraz liczności punktów próbkowania nośnika. Lewy kraniec nośnika wynikowego jest równy sumie lub różnicy (zależnie od działania) lewych krańców nośników liczb bazowych; podobnie wygląda kwestia prawego krańca nośnika – jest on równy sumie lub różnicy (zależnie od działania) prawych krańców nośników liczb bazowych. Zjawisko to jest niekorzystne i musi być uwzględniane przy tworzeniu algorytmów obsługi modeli wykorzystujących takie działania.

Na etapie projektowania modelu rozmytego należy również podjąć decyzję o sposobie definiowania powiązań pomiędzy poszczególnymi czynnikami rozmytymi. W relacyjnej rozmytej mapie kognitywnej czynniki rozmyte są ze sobą powiązane poprzez relacje rozmyte, których definicję zawiera równanie (3.130). Równanie to przedstawia relację jako pewien zbiór rozmyty charakteryzujący interakcję pomiędzy dwoma zbiorami nierozmytymi. Przeniesienie tego pojęcia w dziedzinę liczb rozmytych wymaga użycia jednej z operacji na relacjach rozmytych. Nasuwające się intuicyjnie rozwiązanie polegające na zastosowaniu mnożenia rozmytego, nie może mieć tu zastosowania, ponieważ argumentami tego „mnożenia" mają być nie dwie liczby rozmyte, a liczba rozmyta i relacja rozmyta. W tej sytuacji należy znaleźć operator będący swoistym odpowiednikiem mnożenia rozmytego, pasujący przy tym do takiego typu argumentów. Warunki takie spełnia kompozycja rozmyta, opisana równaniem (3.140). Definicja (3.140) ma charakter ogólny (wykorzystujący t-normę), należy więc dokonać jej konkretyzacji. W przypadku relacyjnej rozmytej mapy kognitywnej działającej w środowisku dyskretnym, wspomniana t-norma będzie miała formę operacji maksyminowej kompozycji liczby rozmytej i relacji rozmytej, a operacja przybierze postać (4.11):

$$\mu_B(y) = \mu_{A \circ R}(y) = \max_{x \in A}\{\min[\mu_A(x), \mu_R(x, y)]\} \tag{4.11}$$

gdzie:

A, B – liczby rozmyte powiązane relacją rozmytą R;

$\circ$ – operator maksyminowej kompozycji rozmytej;

x – punkty nośnika liczby rozmytej A;
y – punkty nośnika liczby rozmytej B;
(x, y) – punkty nośnika relacji rozmytej R.

Wszystkie liczby rozmyte w omawianym modelu są określone na tym samym nośniku U (zgodnym z (4.7)), w związku z czym w (4.11): $x \in U$, $y \in U$.

Poniższy prosty przykład ilustruje działanie dyskretnej operacji kompozycji rozmytej.

Przykład 4.2

Kompozycja rozmyta liczby rozmytej A i relacji rozmytej R na nośniku o zakresie $[0, 1]$:

$$A = \frac{0{,}030}{0{,}00} + \frac{0{,}210}{0{,}25} + \frac{0{,}677}{0{,}50} + \frac{1{,}000}{0{,}75} + \frac{0{,}677}{1{,}00}$$

$$R = \begin{bmatrix} 1{,}000 & 0{,}677 & 0{,}210 & 0{,}030 & 0{,}002 \\ 0{,}958 & 0{,}841 & 0{,}338 & 0{,}062 & 0{,}005 \\ 0{,}841 & 0{,}958 & 0{,}499 & 0{,}119 & 0{,}013 \\ 0{,}677 & 1{,}000 & 0{,}677 & 0{,}210 & 0{,}030 \\ 0{,}499 & 0{,}958 & 0{,}841 & 0{,}338 & 0{,}062 \end{bmatrix}$$

$$B = A \circ R = \frac{0{,}677}{0{,}00} + \frac{1{,}000}{0{,}25} + \frac{0{,}677}{0{,}50} + \frac{0{,}338}{0{,}75} + \frac{0{,}062}{1{,}00}$$

Sposób powstawania powyższego wyniku zostanie zilustrowany na przykładzie pojedynczego singletonu rozmytego – o mianowniku 0,75. Operacja kompozycji rozmytej nie zmienia nośnika wynikowego, w związku z czym pozycja takiego singletonu w wynikowej liczbie rozmytej B będzie taka sama jak w liczbie A, czyli znajdzie się on na miejscu czwartym. W celu wyznaczenia jego licznika należy wykonać następujące kroki:

a) wyodrębnić pary powstałe z liczników singletonów liczby A oraz liczb czwartej kolumny relacji rozmytej R:

$$(0{,}030\ ;\ 0{,}030), (0{,}210\ ;\ 0{,}062), (0{,}677\ ;\ 0{,}119), (1{,}000\ ;\ 0{,}210), (0{,}677\ ;\ 0{,}338)$$

b) z każdej pary pozostawić liczbę o niższej wartości i dodać ją do kolejki powiązanej z tymczasowym licznikiem singletonu wynikowego:

$$\frac{0{,}030\ ;\ 0{,}0{,}62\ ;\ 0{,}119\ ;\ 0{,}210\ ;\ 0{,}338}{0{,}75}$$

c) z elementów tymczasowego licznika wyłonić jeden, o najwyższej wartości, a pozostałe usunąć uzyskując w ten sposób docelową wartość singletonu rozmytego:

$$\frac{0{,}338}{0{,}75}$$

Uwaga 4.3

Operacja maksyminowej kompozycji rozmytej zbioru rozmytego i relacji rozmytej, taka jak w (4.11), nie zmienia nośnika wynikowego (w przeciwieństwie do innych operacji na liczbach rozmytych), co stanowi jej dodatkową przewagę w systemach modelowania komputerowego. Z drugiej strony, wprowadzenie do systemu operacji kompozycji rozmytej wiąże się z bardzo poważnym problemem definiowania i późniejszych modyfikacji kształtu relacji rozmytych – jest to zagadnienie o kluczowym znaczeniu.

4.4. ROZMYWANIE WARTOŚCI CZYNNIKÓW

W klasycznych modelach rozmytych map kognitywnych rozmywanie wartości każdego czynnika dokonywane jest w oparciu o metody wykorzystujące pewną liczbę wartości lingwistycznych do opisu pewnych, wybranych poziomów wartości [86, 223], gdzie klasyfikacja polega na porównywaniu wartości poszczególnych funkcji przynależności w tym samym punkcie nośnika. Przy tym wspomniane wyżej funkcje przynależności mogą przybierać różne formy, zgodne z klasami opisanymi przez równania (3.27)–(3.36) lub inne, i formy te mogą być różne dla różnych wartości lingwistycznych. Rozmytą wartość czynnika można przedstawić w postaci zbioru rozmytego operującego na pewnej skończonej liczbie wartości lingwistycznych, tak jak to pokazano w równaniu (3.52). Bezpośrednie stosowanie takiej metody jest trudne algorytmicznie, ponieważ wiąże się z analizą funkcji przynależności wszystkich tych wartości lingwistycznych i badaniem ich związków z rozmywaną wartością liczbową. Dlatego też zostanie zaproponowane zmodyfikowane podejście wykorzystujące do tego samego celu tylko jedną, odpowiednio zaprojektowaną rozmytą wartość lingwistyczną.

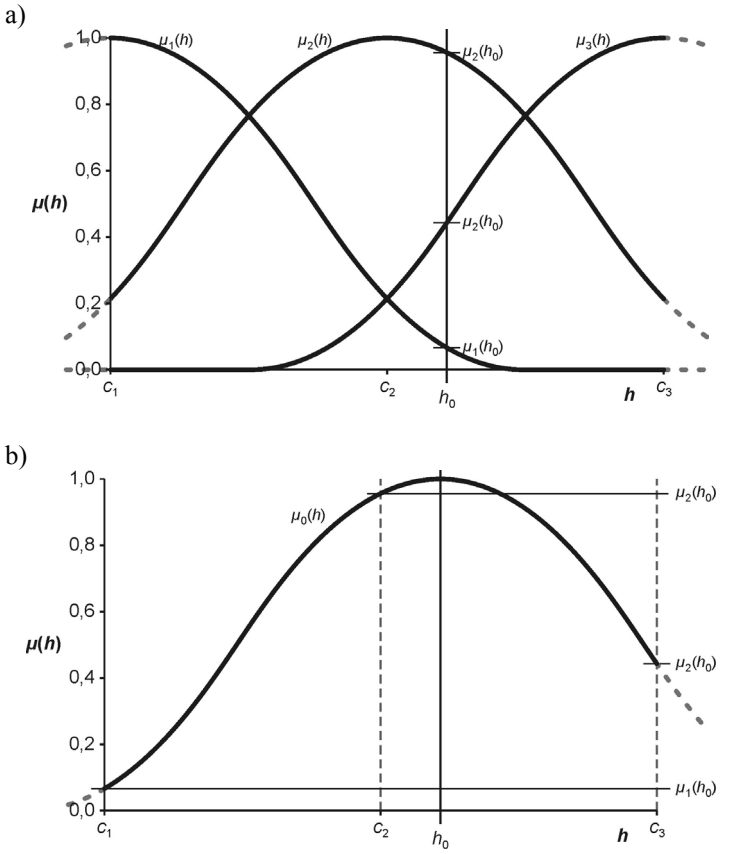
4.4.1. Zmodyfikowane podejście do reprezentacji rozmytych wartości czynników

Wprowadzenie nowego podejścia do sposobu rozmywania wartości czynnika należy rozpocząć od poczynienia kilku prostych założeń:

- wszystkie wartości lingwistyczne danego czynnika rozmytego są normalne (w rozumieniu (3.13)),
- funkcje przynależności tych wartości lingwistycznych mają jednakowy charakter,
- funkcje przynależności są symetryczne i równomiernie rozmieszczone na nośniku.

Przyjęcie takich założeń pozwala zastosować prostsze podejście, w którym dla scharakteryzowania rozmywanej wartości liczbowej stosuje się tylko jedną funkcję

przynależności, której centrum (oś symetrii) jest równe rozmywanej liczbie. Dodatkowo podejście takie pozwala swobodnie zmieniać liczbę używanych wartości lingwistycznych (np. w procesie adaptacji modelu [169, 176, 218]) bez konieczności zmiany algorytmów obliczeniowych. Taka metoda reprezentacji zestawu wartości lingwistycznych przy użyciu zaledwie jednego zbioru rozmytego pozwala operować dokładnie takim samym zbiorem singletonów rozmytych jak metoda tradycyjna. Ilustruje to rysunek 4.2, na którym pokazano sposób odwzorowania wartości liczbowej h_0 w wielkość lingwistyczną reprezentowaną przez trzy rozmyte wartości lingwistyczne o funkcjach przynależności odpowiednio: μ_1, μ_2, μ_3.



Rys. 4.2. *Ilustracja klasycznej (a) i zmodyfikowanej (b) metody odwzorowania wartości* h_0 *wielkości liczbowej* h *przez trzy rozmyte wartości lingwistyczne pewnej wielkości lingwistycznej*

Zamiast wyznaczać wartości wszystkich funkcji przynależności dla argumentu h_0 – jak na rysunku 4.2a (co w środowisku dyskretnym może być kłopotliwe, ponieważ punkt h_0 przeważnie nie będzie się pokrywał z żadnym z punktów próbkowania nośnika), można – jak na rysunku 4.2b – użyć jednej funkcji przynależności (z centrum w punkcie h_0) i znaleźć jej wartości w punktach próbkowania no-

śnika c_1, c_2, c_3 (punkty te pokrywają się z jądrami trzech rozmytych wartości lingwistycznych z rys. 4.2a). Uzyskany zestaw singletonów rozmytych (4.12) (w formie zgodnej z (3.52)) będzie identyczny w obu wyżej wymienionych przypadkach, co przemawia za wyborem metody przedstawionej na rysunku 4.2b.

$$\mu_h(h_0) = \frac{\mu_1(h_0)}{c_1} + \frac{\mu_2(h_0)}{c_2} + \frac{\mu_3(h_0)}{c_3} \tag{4.12}$$

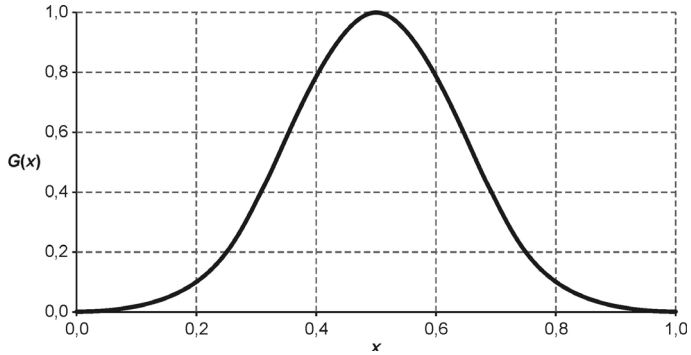
4.4.2. Funkcje przynależności wartości lingwistycznych reprezentujących rozmyte wartości czynników

Dalsze rozważania zostaną poprzedzone wprowadzeniem dodatkowej klasy funkcji charakterystycznej zbioru rozmytego (w uzupełnieniu do klas pokazanych w podrozdziale 3.1.2). Będzie to klasa G (funkcjonalnie zbliżona do klasy π z (3.35)), której podstawą jest kształt funkcji typu Gaussa, o następującym przebiegu:

$$G_{a,b}(x) = e^{-\left(\frac{x-a}{b}\right)^2} \tag{4.13}$$

gdzie:
a, b – kluczowe parametry decydujące o własnościach funkcji.

Graficzną reprezentację funkcji klasy G (gaussoidy) pokazano na rysunku 4.3.



Rys. 4.3. *Reprezentacja graficzna funkcji klasy G.* $a = 0{,}5$*;* $b = 0{,}2$

Funkcja charakterystyczna klasy G, pokazana w (4.13), nie przyjmuje wartości zerowych, co sprawia, że nośnik zbioru rozmytego A określonego w przestrzeni rozważań U, o funkcji przynależności $G_A(u)$ zgodnej z (4.13), może być utożsamiany z przestrzenią rozważań U bez żadnych dodatkowych modyfikacji.

Działanie modelu relacyjnej rozmytej mapy kognitywnej opiera się na wykorzystaniu liczb rozmytych i algebry tych liczb, w związku z tym wartości czynników w takim modelu będą miały formę liczb rozmytych, czyli zbiorów rozmytych

spełniających warunki (3.13), (3.142) i (3.143). Przy tym nawiązując do powyższych rozważań na temat sposobu wyznaczania singletonów rozmytych, rozmywanie wartości czynników mapy będzie się odbywało przy zachowaniu następujących ważnych założeń:

ZAŁOŻENIE 4.1

Wszystkie wartości lingwistyczne rozmytej wartości danego czynnika mają funkcje przynależności tej samej klasy, różniące się jedynie usytuowaniem centrum (osi rozmywania).

ZAŁOŻENIE 4.2

Funkcje przynależności wszystkich wartości lingwistycznych rozmytej wartości danego czynnika należą do jednej z następujących klas: Λ, π lub G.

ZAŁOŻENIE 4.3

Centra wartości lingwistycznych rozmytej wartości danego czynnika są równomiernie rozmieszczone na przestrzeni nośnika z krokiem Δk zgodnym z (4.6) – określa się je mianem punktów próbkowania nośnika.

ZAŁOŻENIE 4.4

Wartości lingwistyczne rozmytej wartości danego czynnika określa się na zmodyfikowanym nośniku zgodnym z (4.7).

4.4.3. Wyznaczanie singletonów rozmytych wartości czynników

Dla ujednolicenia późniejszych rozważań, wprowadza się następujące oznaczenia:

X_i – rozmyta wartość i-tego czynnika;

$\bar{X}_i$ – wyostrzona wartość i-tego czynnika;

$\hat{X}_i$ – zadana (skalarna), znormalizowana wartość i-tego czynnika.

Ponadto wprowadza się jednolite dla wszystkich czynników modelu zasady:

1. Wszystkie dalsze rozważania i przykłady będą operowały na wartościach czynników poddanych uprzedniej normalizacji zgodnie z (2.4).
2. Zgodnie z rozważaniami przedstawionymi w podrozdziale 4.2, wartości czynników będą rozmywane na nośniku zgodnym z (4.7).
3. Liczba i rozmieszczenie punktów próbowania nośnika będą jednakowe dla wszystkich czynników.

Rozmywanie wartości czynników powinno być poprzedzone analizą charakteru każdego czynnika i wykonaniem następujących kroków:

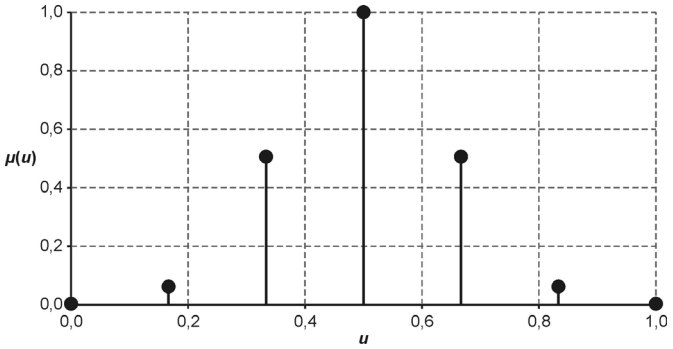
a) wybór funkcji przynależności liczb rozmytych, które będą reprezentowały wartości czynników – zgodnie z założeniem 4.2 mogą to być funkcje klasy: Λ, π lub G;
b) ustalenie przybliżonych wartości współczynników rozmycia czynników;
c) wybór zakresu nośnika i liczby punktów próbkowania nośnika (więcej informacji na ten temat zawarto w podrozdziale 4.8).

Z wielu względów (głównie technicznych, związanych z algorytmami późniejszej adaptacji modelu) dogodnie jest używać możliwie najbardziej uniwersalnej klasy funkcji przynależności – jest nią klasa G zdefiniowana równaniem (4.13). Jeśli to ona zostanie wybrana, to indywidualna funkcja przynależności liczby rozmytej reprezentującej wartość rozmytą pojedynczego czynnika przybierze następującą postać:

$$\mu_{X_i}(u) = e^{-\left(\frac{u-\bar{X}_i}{\sigma_i}\right)^2} \tag{4.14}$$

gdzie:
X_i – rozmyta wartość i-tego czynnika;
u – nośnik zgodny z (4.7);
$\bar{X}_i$ – centrum (znormalizowana wartość rzeczywista) i-tego czynnika;
σ_i – współczynnik rozmycia i-tego czynnika;
$i = 1, \ldots, n$;
n – liczba czynników.

Funkcja przynależności (4.14) ma przebieg zgodny z przedstawionym na rysunku 4.3, pamiętać jednak należy, że w rzeczywistych realizacjach algorytmicznych liczba rozmyta reprezentująca wartość czynnika ma charakter dyskretnego zestawu singletonów, w związku z czym jej reprezentacja graficzna przypomina raczej tę przedstawioną na rysunku 4.4.



Rys. 4.4. *Reprezentacja graficzna rozmytej wartości czynnika o funkcji przynależności zgodnej z (4.14).* $\bar{X} = 0{,}5$*;* $\sigma = 0{,}2$*;* $k = 7$*; nośnik o zakresie* $[0, 1]$

Zaznaczone na rysunku 4.4 punkty obrazują kolejne singletony rozmyte liczby rozmytej reprezentującej rozmytą wartość X pewnego czynnika, przy czym na nośniku o zakresie $[0, 1]$ równomiernie rozłożono 7 punktów próbkowania.

Sposób rozmywania wartości czynników zgodny z (4.14) będzie podstawową metodą wykorzystywaną w dalszych rozważaniach i przykładach dotyczących modelu relacyjnej rozmytej mapy kognitywnej.

4.5. BUDOWA RELACJI ROZMYTYCH

Relacje rozmyte są elementami o kluczowym znaczeniu dla poprawnej pracy modelu relacyjnej rozmytej mapy kognitywnej. Przy tym główna idea budowy takiego modelu (tzn. automatyzacja procesu tworzenia i uczenia mapy) nie dopuszcza „ręcznego" definiowania poszczególnych singletonów takiej relacji. Byłoby to zresztą bardzo trudne na poziomie teoretycznych algorytmów, które muszą być poprawne dla dowolnej liczby punktów próbkowania nośnika oraz dla dowolnej rozpiętości jego zakresu. Z tego powodu tak ważne jest znalezienie sposobu zautomatyzowanego (w pewnym sensie) tworzenia relacji rozmytych.

Wzajemne zależności pomiędzy czynnikami mogą mieć zróżnicowany charakter, dlatego też niezwykle istotny jest dobór odpowiedniego kształtu funkcji przynależności μ_R każdej relacji rozmytej. Projektując ją, należy uwzględnić zarówno moc relacji (będącą wyrazem siły wzajemnych oddziaływań czynników), jak też jej charakter, wyrażony na przykład koncentracją tej mocy w określonych punktach. Ponadto należy uwzględnić czysto techniczne aspekty numerycznego algorytmu dokonywania operacji matematycznych na modelu. Jeśli, jak wykazano w podrozdziale 4.2, wartości wszystkich czynników powinny być rozmywane na tym samym nośniku, to i relacje rozmyte powinny korzystać z tego nośnika (dotyczy to zwłaszcza liczby singletonów rozmytych).

Definiowanie typu relacji rozmytych musi być dokonywane na pewnym poziomie ogólności, a z tego wynika, że należy mu nadać formę zależności algebraicznych, które będą przy tym stosunkowo łatwe do komputerowej implementacji. Znalezienie metody budowy takich relacji oraz kształtu ich postaci algebraicznych było jednym z głównych problemów rozwiązanych podczas opracowywania proponowanego podejścia. Liczne badania i próby [73, 76, 79, 82, 168, 171] doprowadziły do opracowania pewnej ogólnej postaci funkcyjnej (4.15), w oparciu o którą mogą być projektowane wszystkie rozmyte relacje modelu:

$$\mu_R(a, b) = f_R\left(\frac{p_1 \cdot b - p_2 \cdot r(a)}{p_3 \cdot \sigma}\right) \tag{4.15}$$

gdzie:

a, b – nośniki liczb rozmytych połączonych relacją R (w modelach relacyjnych rozmytych map kognitywnych $a = b$);

f_R – funkcja bazowa zależna od klasy wybranej funkcji przynależności;

σ – współczynnik rozmycia (rozrzut);

$r(a)$ – funkcyjny współczynnik mocy relacji;
p_1, p_2, p_3 – współczynniki zależne od klasy wybranej funkcji przynależności.

W takim podejściu każda relacja rozmyta $R_{i,j}$ (pomiędzy dwoma czynnikami o numerach i oraz j) jest determinowana przez pewną funkcję przynależności $\mu_{R_{i,j}}$. Oczywiście charakter funkcji μ_R można różnicować dla różnych relacji pomiędzy różnymi czynnikami, jednakże, z punktu widzenia automatyzacji pewnych procesów pracy z modelem (np. automatycznego uczenia), dogodnie jest bazować na pewnym ujednoliconym typie kształtu, który może być określony, podobnie jak przy rozmywaniu wartości czynników, przez funkcję klasy Λ, π lub G. Ponadto, ponieważ relacje rozmyte nie podlegają takim rygorom jak liczby rozmyte, można też wykorzystać do tego celu inne funkcje, np. klasy Π. Przy takim ujęciu można automatyzować budowę relacji rozmytych poprzez narzucenie im pewnej ogólnej wspólnej formy funkcji przynależności. Równanie (4.16) przedstawia jedną z możliwych postaci takiej funkcji (klasy G):

$$\mu_{R_{i,j}}(u_i, u_j) = e^{-\left(\frac{u_j - r_{i,j}(u_i)}{\sigma_{i,j}}\right)^2}, \tag{4.16}$$

gdzie:
i, j – numery czynników powiązanych relacją;
u_i, u_j – nośniki czynników powiązanych relacją;
$r_{i,j}(u_i)$ – współczynnik mocy relacji rozmytej pomiędzy czynnikami i-tym i j-tym (w postaci funkcyjnej);
$\sigma_{i,j}$ – współczynnik rozmycia relacji rozmytej pomiędzy czynnikami i-tym i j-tym.

Uwaga 4.4

Warto zauważyć, że występujący w równaniu (4.16) współczynnik mocy relacji $r_{i,j}(u_i)$ jest funkcją nośnika oraz, że nośniki czynników u_i i u_j są zbiorami tych samych k punktów, czyli: $u_i = \{u_{i(1)}, u_{i(2)}, \dots, u_{i(k)}\} = u_j = \{u_{j(1)}, u_{j(2)}, \dots, u_{j(k)}\}$.

Na rysunku 4.5 pokazano reprezentację graficzną relacji rozmytej, w której $r_{i,j}(u_i)$ ma charakter liniowy (jak w (4.17)):

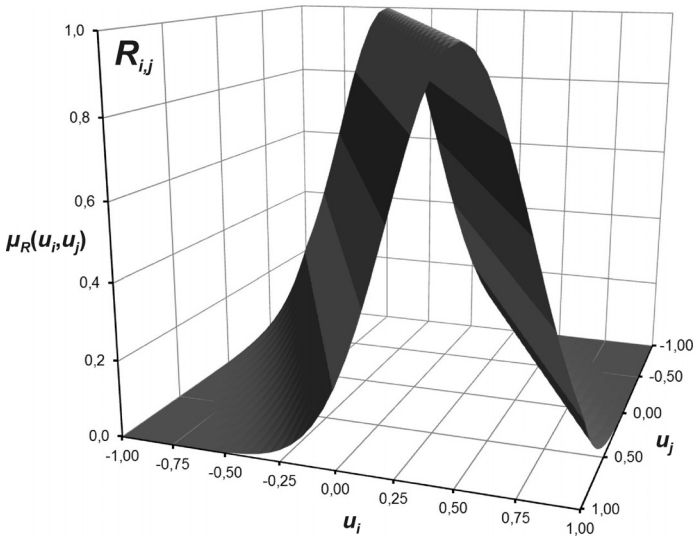
$$r_{i,j}(u_i) = \bar{r}_{i,j} \cdot u_i \tag{4.17}$$

gdzie:
$\bar{r}_{i,j}$ – współczynnik kierunkowy liniowej funkcji mocy relacji pomiędzy czynnikami i-tym i j-tym.

Czyli funkcja przynależności relacji rozmytej klasy G przybiera tu postać:

$$\mu_{R_{i,j}}(u_i, u_j) = e^{-\left(\frac{u_j - \bar{r}_{i,j} \cdot u_i}{\sigma_{i,j}}\right)^2}, \tag{4.18}$$

która, jak zostanie pokazane dalej, może być z powodzeniem stosowana do budowy automatycznego algorytmu uczącego.



Rys. 4.5. *Graficzna reprezentacja relacji rozmytej o funkcji przynależności typu (4.16), w której:* $\bar{r}_{i,j} = 0{,}5$, $\sigma_{i,j} = 0{,}4$, $k = 41$, *rozmytej na nośniku o zakresie* $[-1, 1]$

Przez analogię równanie (4.19) i rysunek 4.6 pokazują postać oraz graficzną reprezentację relacji rozmytej zbudowanej na bazie zmodyfikowanej funkcji klasy Λ, natomiast równanie (4.20) i rysunek 4.7 podobną relację zbudowaną na bazie zmodyfikowanej funkcji klasy Π.

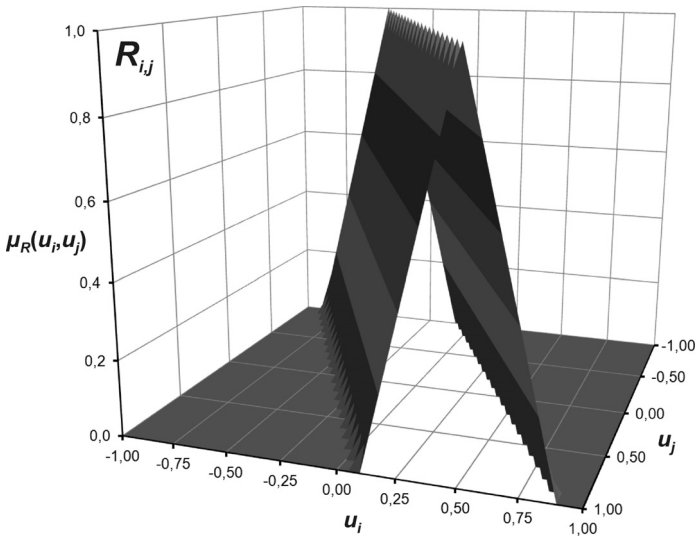
$$\mu_{R_{i,j}}(u_i, u_j) = \begin{cases} 0 & \text{dla} \quad \begin{matrix} u_j \le r_{i,j}(u_i) - \Delta L \text{ lub} \\ u_j \ge r_{i,j}(u_i) + \Delta P \end{matrix} \\ 1 + \dfrac{u_j}{\Delta L} - \dfrac{r_{i,j}(u_i)}{\Delta L} & \text{dla} \quad r_{i,j}(u_i) - \Delta L < u_j \le r_{i,j}(u_i) \\ 1 - \dfrac{u_j}{\Delta P} + \dfrac{r_{i,j}(u_i)}{\Delta P} & \text{dla} \quad r_{i,j}(u_i) < u_j < r_{i,j}(u_i) + \Delta P \end{cases} \tag{4.19}$$

gdzie:

$\bar{r}_{i,j}$ – współczynnik kierunkowy liniowej funkcji mocy relacji pomiędzy czynnikami i-tym i j-tym (jak w (4.17));

ΔL – rozrzut lewostronny;

ΔP – rozrzut prawostronny.



Rys. 4.6. *Graficzna reprezentacja relacji rozmytej o funkcji przynależności zmodyfikowanego typu Λ, w której:* $\bar{r}_{i,j} = 0{,}5$, $\Delta L = 0{,}4$, $\Delta P = 0{,}4$, $k = 41$, *rozmytej na nośniku o zakresie* $[-1, 1]$

$$\mu_{R_{i,j}}(u_i, u_j) = \begin{cases} 0 & \text{dla} \quad u_j \le r_{i,j}(u_i) - \dfrac{\Delta W}{2} - \Delta L \text{ lub} \\ & \quad u_j \ge r_{i,j}(u_i) + \dfrac{\Delta W}{2} - \Delta P \\ \dfrac{u_j}{\Delta L} + \dfrac{\Delta L - r_{i,j}(u_i) + \dfrac{\Delta W}{2}}{\Delta L} & \text{dla} \quad u_j > r_{i,j}(u_i) - \dfrac{\Delta W}{2} - \Delta L \text{ oraz} \\ & \quad u_j \le r_{i,j}(u_i) - \dfrac{\Delta W}{2} \\ 1 & \text{dla} \quad u_j > r_{i,j}(u_i) - \dfrac{\Delta W}{2} \text{ oraz} \\ & \quad u_j \le r_{i,j}(u_i) + \dfrac{\Delta W}{2} \\ 1 - \dfrac{u_j}{\Delta P} + \dfrac{r_{i,j}(u_i) + \dfrac{\Delta W}{2}}{\Delta P} & \text{dla} \quad u_j > r_{i,j}(u_i) + \dfrac{\Delta W}{2} \text{ oraz} \\ & \quad u_j < r_{i,j}(u_i) + \dfrac{\Delta W}{2} + \Delta P \end{cases} \tag{4.20}$$

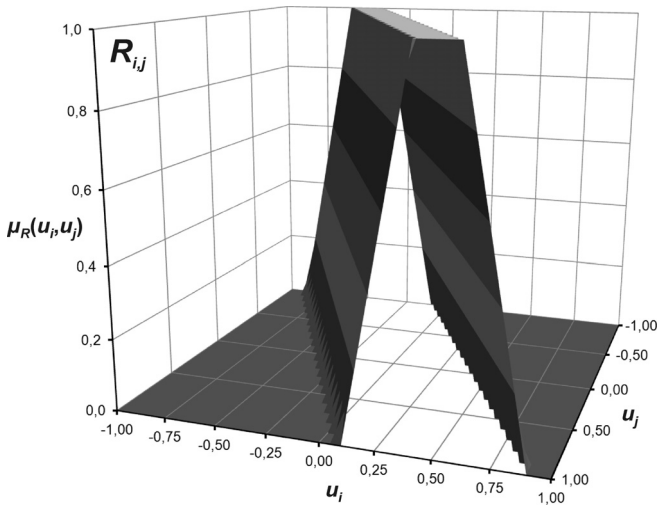
gdzie:

$\bar{r}_{i,j}$ – współczynnik kierunkowy liniowej funkcji mocy relacji pomiędzy czynnikami i-tym i j-tym (jak w (4.17));

ΔW – szerokość jądra bazowej funkcji przynależności typu Π;

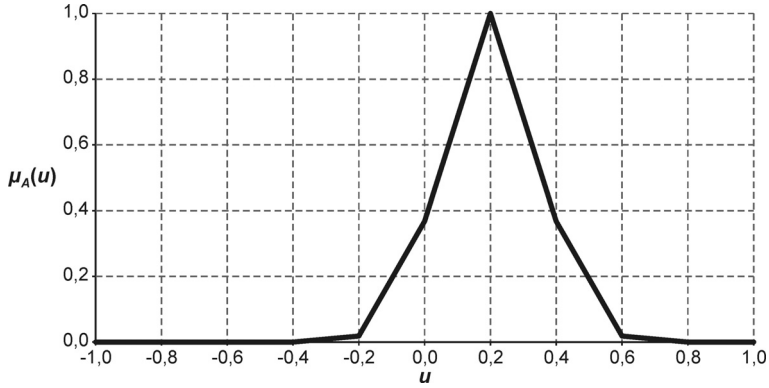
ΔL – rozrzut lewostronny;

ΔP – rozrzut prawostronny.



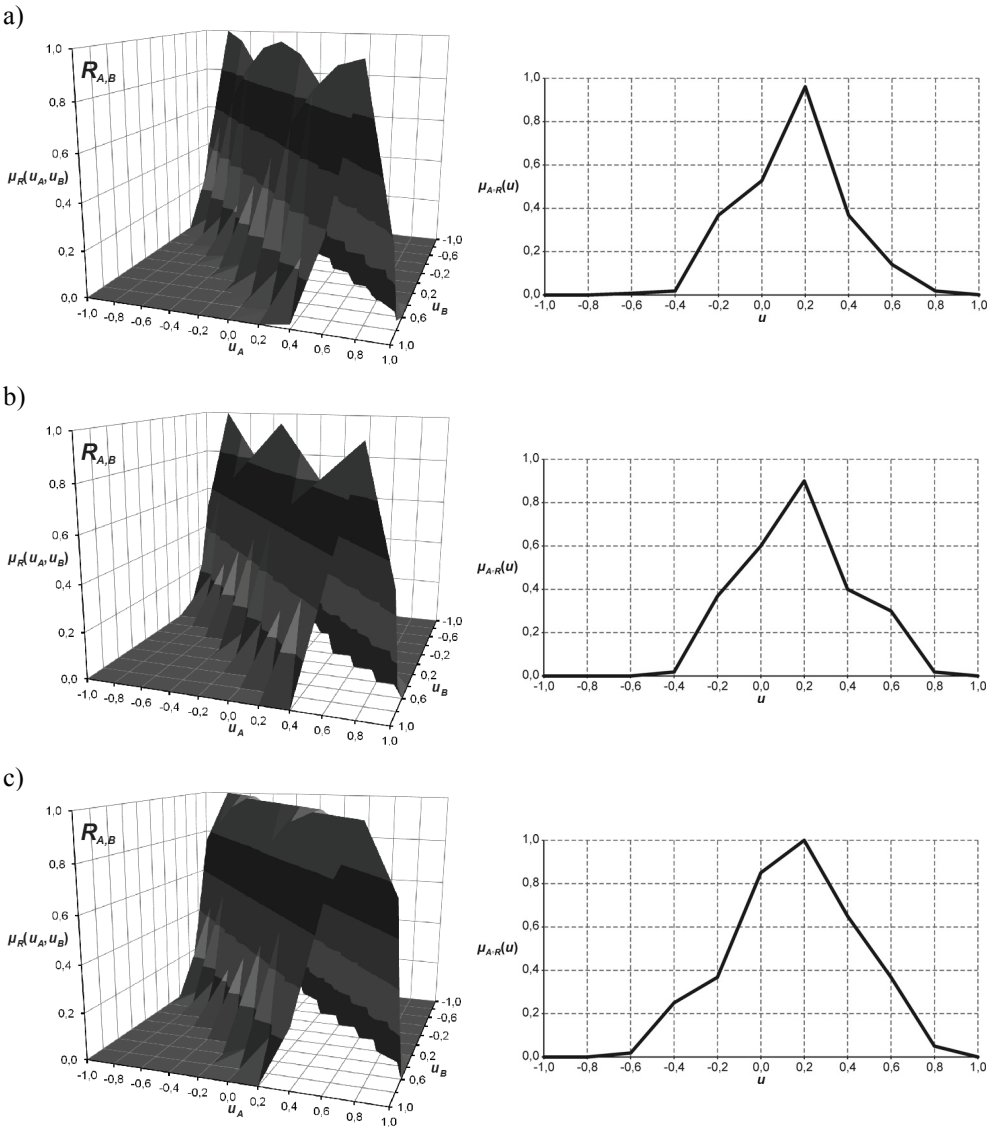
Rys. 4.7. *Graficzna reprezentacja relacji rozmytej o funkcji przynależności zmodyfikowanego typu Π, w której:* $\bar{r}_{i,j} = 0{,}5$, $\Delta W = 0{,}2$, $\Delta L = 0{,}3$, $\Delta P = 0{,}3$, $k = 41$, *rozmytej na nośniku o zakresie* $[-1, 1]$

Każdy z wyżej przedstawionych typów relacji rozmytej charakteryzuje się pewnymi kluczowymi parametrami, od których zależy sposób, w jaki relacja odwzorowuje wzajemne oddziaływania czynników rozmytych. Czynniki są związane z relacjami za pomocą działań maksyminowej kompozycji rozmytej zdefiniowanej przez (4.11). Należy przy tym pamiętać, że dokonując kompozycji danego czynnika z różnymi relacjami (o tych samych mocach), można uzyskać różne wynikowe kształty funkcji przynależności, co pokazują rysunki 4.8 i 4.9 [170].



Rys. 4.8. *Przykładowa wartość rozmyta czynnika A – funkcja przynależności klasy G, centrum* $\bar{A} = 0{,}2$, *współczynnik rozmycia* $\sigma_A = 0{,}2$, *zakres nośnika* $[-1, 1]$, *liczba punktów próbkowania nośnika* $k = 11$

Relacje rozmyte przedstawione na rysunku 4.9 mają jednakowe współczynniki mocy i podobne współczynniki rozmycia (rozrzutu), wszystkie są też symetryczne, a jednak ich kompozycje z tą samą liczbą rozmytą dają wyniki różniące się zarówno kształtem funkcji przynależności, jak i wartością wyostrzoną. Daje to pojęcie o wielorakich możliwościach manipulowania parametrami relacyjnej rozmytej mapy kognitywnej.



Rys. 4.9. *Wyniki kompozycji rozmytej liczby A z rysunku 4.8 z relacjami rozmytymi charakteryzującymi się tymi samymi mocami relacji ($\bar{r} = 0{,}8$) i różnymi typami bazowej funkcji przynależności: a) relacja typu G, b) relacja typu Λ, c) relacja typu Π*

W procesie adaptacji (uczenia) modelu parametry poszczególnych relacji rozmytych muszą być zmieniane tak, aby wynikowy model możliwie dokładnie odzwierciedlał pracę modelowanego systemu. W relacji typu G (4.16) występują dwa takie parametry, w relacji typu Λ (4.19) – trzy, natomiast w relacji typu Π (4.20) – aż cztery. To z kolei, do pewnego stopnia, warunkuje wybór typu relacji do praktycznych zastosowań. Proces uczenia jest dość złożony i czasochłonny – tym bardziej, im więcej składników ma wpływ na ostateczny wynik. Z tego powodu, w rozwiązaniach aplikacyjnych, najbardziej dogodne wydaje się stosowanie relacji typu G.

4.6. WYOSTRZANIE WARTOŚCI CZYNNIKÓW

W podrozdziale 3.1.6 opisano najczęściej stosowane metody wyostrzania (defuzyfikacji) zbiorów rozmytych. Liczby rozmyte, jako specyficzne zbiory rozmyte, podlegają tym samym zasadom, w związku z czym można do nich stosować każdą z podanych tam metod. Jednakże, mnogość możliwych typów połączeń pomiędzy czynnikami a relacjami powoduje, że nie wszystkie metody wyostrzania są jednakowo przydatne. Gdyby założyć, że wszystkie występujące w systemie liczby rozmyte, łącznie z wynikami kompozycji czynników i relacji, będą liczbami właściwymi (zgodnie z definicją z podrozdziału 3.4), to najlepiej byłoby stosować którąś z metod wyostrzania odwołujących się do maksimum zbioru rozmytego (np. metodę pierwszego maksimum). Zarówno duża rozpiętość możliwych do użycia typów relacji, jak i charakter dyskretnego środowiska obliczeń komputerowych powodują, że wyniki działań mogą być zbiorami rozmytymi o więcej niż jednym singletonie z maksymalnym stopniem przynależności i dlatego należy poddawać je dodatkowej obróbce. W tej sytuacji najbardziej uniwersalna wydaje się metoda średniej ważonej, której definicję zawiera równanie (3.58). Jej stosowanie jest o tyle korzystne, że nie odnosi się ona bezpośrednio do maksymalnych wartości stopnia przynależności, a raczej do przynależności w sensie uogólnionym. Metoda średniej ważonej jest jednak wrażliwa na asymetrię zbioru rozmytego względem nośnika (skutki tej asymetrii pokazano na rys. 4.13), co może skutkować niedokładnością pracy modelu. Można temu dość łatwo zaradzić, stosując nośnik o większej rozpiętości, tak jak to opisano w podrozdziale 4.8.

Z wyżej wymienionych względów, w modelu relacyjnej rozmytej mapy kognitywnej proponuje się stosowanie wyostrzania metodą średniej ważonej.

4.7. STABILIZACJA NOŚNIKA

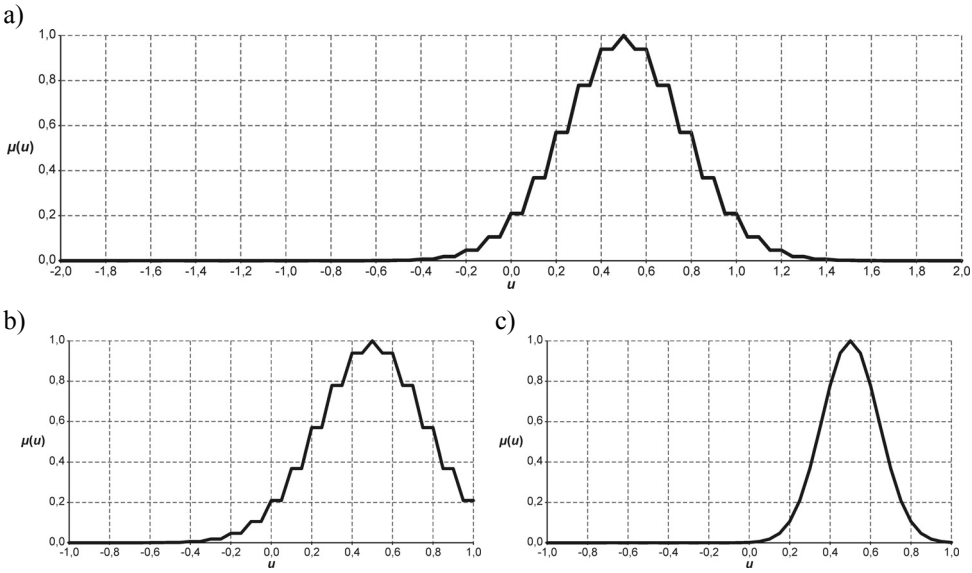
Działanie modelu relacyjnej rozmytej mapy kognitywnej opiera się na operacjach kompozycji maksyminowej pomiędzy rozmytymi wartościami czynników a relacjami rozmytymi. Każdorazowo wynik takiej kompozycji musi być liczbą rozmytą określoną na nośniku bazowym, co oznacza, że również jej argumenty muszą być określone na takim nośniku. Wynika stąd konieczność zachowania stałego charakteru nośnika, na którym określa się wszystkie komponenty modelu. Z drugiej strony istota definicji operacji algebraicznych na liczbach rozmytych, przedstawionych w (3.151)–(3.154) oraz (4.9) i (4.10) sprawia, że samorzutne za-

chowanie stabilności nośnika nie jest możliwe. Każda taka operacja powoduje zniekształcenie nośnika, ponieważ bazowe punkty próbkowania zostają zastąpione przez punkty, których wartości są, odpowiednio, sumami, różnicami, iloczynami lub ilorazami (zależnie od wykonywanej operacji) wszystkich możliwych par punktów próbkowania nośnika. Prowadzi to do jednoczesnego wystąpienia dwóch problemów. Po pierwsze, następuje zmiana (najczęściej rozszerzenie) zakresu nośnika, co zwiększa stopień rozmycia liczby wynikowej. Po drugie, zmianie ulegają zarówno liczba, jak i położenie poszczególnych punktów próbkowania nośnika, co jest niedopuszczalne z punktu widzenia niektórych komputerowych algorytmów modelu rozmytego. Należy dążyć do utrzymywania stałego charakteru nośnika przez cały czas pracy modelu. Istnieją metody takiego wykonywania rozmytych operacji arytmetycznych, które nie powodują zmiany zakresu nośnika, wymagają one jednak przedefiniowania pojęcia liczby rozmytej. Można np. zastosować tzw. skierowane liczby rozmyte [104–107], jednakże łatwiejsza jest numeryczna implementacja liczb rozmytych o klasycznym charakterze. Pozostanie przy nich wymaga wprowadzenia pewnych mechanizmów zapewniających niezmienność nośnika. Jedna z metod polega na „obcięciu" nośnika do wymaganych rozmiarów z jednoczesnym dopasowaniem punktów charakterystycznych liczby rozmytej do bazowych punktów próbkowania nośnika. Można to uczynić, np. na drodze interpolacji. Drugi sposób polega na wyostrzeniu liczby w celu uzyskania jej centrum, a następnie ponownym rozmyciu wokół tego centrum na bazowym nośniku z użyciem funkcji przynależności przypisanej w modelu do danego czynnika. Rysunek 4.10 ilustruje obie te metody.

Rysunek 4.10a przedstawia wynik dodawania dwóch liczb rozmytych (z centrami: $0{,}1$ i $0{,}4$) o nośniku bazowym $[-1, 1]$ i 41 punktach próbkowania. Na rysunku 4.10b i 4.10c pokazano dwie metody dopasowania tego wyniku do bazowego nośnika. Wybór metody dopasowania zależy od konkretnej sytuacji oraz sposobu wykorzystania wyniku końcowego. W opracowanym modelu, którego pracę charakteryzuje równanie (4.3), wykorzystuje się obie metody przedstawione na rysunku 4.10b i 4.10c.

Równanie (4.3) pokazuje pojedynczy cykl obiegu sygnałów wewnątrz modelu. W modelach o dynamicznej strukturze wewnętrznej przeważnie potrzeba większej liczby takich cykli do ustabilizowania stanu modelu. Każdy cykl obliczeń zaczyna się od rozmycia wartości czynników. Następnie wyznacza się różnice wartości każdego czynnika w dwóch kolejnych krokach czasu dyskretnego: bieżącym (t) i poprzednim $(t-1)$. Wynikiem takiego odejmowania jest zestaw liczb rozmytych o rozszerzonym nośniku. Nośnik ten należy zawęzić do pierwotnych rozmiarów i w tym celu stosuje się metodę przedstawioną na rysunku 4.10b. W kolejnym etapie każda różnica jest poddawana kompozycji maksyminowej z odpowiednimi relacjami rozmytymi, a wyniki tych operacji są sumowane dla każdego badanego czynnika. Do tak uzyskanej sumy (odwzorowującej wpływ zmian czynników na czynnik badany) dodaje się aktualną wartość tego czynnika, co pozwala na wyznaczenie jego wartości dla następnego kroku czasu dyskretnego $(t+1)$. Rozmyte wartości wynikowe poszczególnych czynników są rozłożone na bardzo szerokich

nośnikach, w związku z czym, przed wprowadzeniem ich (poprzez sprzężenia) na początek kolejnego cyklu czasowego obliczeń, należy je wyostrzyć, co dodatkowo pozwala na uzyskanie cząstkowych wniosków o stanie systemu. Wyostrzania końcowego dokonuje się metodą przedstawioną na rysunku 4.10c.



Rys. 4.10. *Graficzna ilustracja metod dopasowywania liczby rozmytej do nośnika bazowego o zakresie* $[-1,1]$ *z wyróżnionymi* 41 *punktami próbkowania: a) pierwotna funkcja przynależności, b) funkcja przynależności po dopasowaniu do nośnika bazowego, c) funkcja przynależności po wyostrzeniu i ponownym rozmyciu na nośniku bazowym*

4.8. DOBÓR PARAMETRÓW ROZMYWANIA

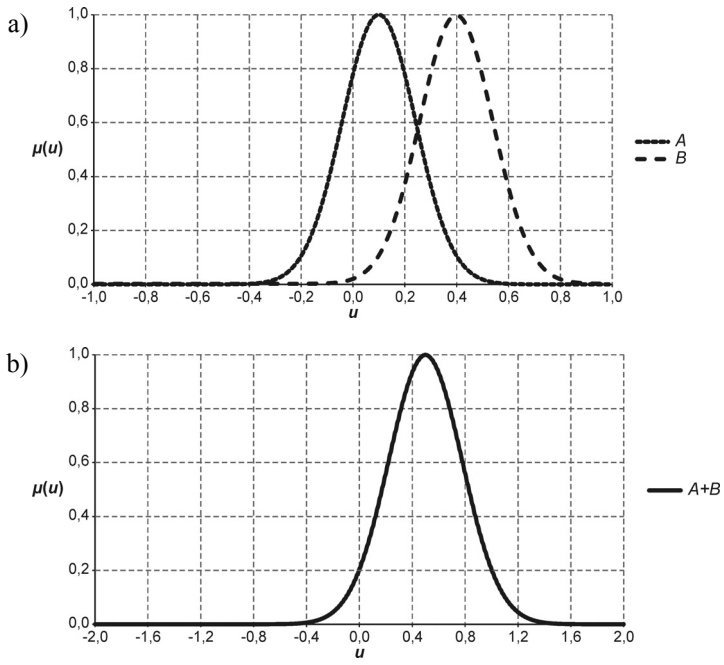
Właściwy dobór parametrów rozmywania czynników i relacji jest bardzo istotnym problemem, którego niewłaściwe rozwiązanie może znacząco pogorszyć dokładność działania całego modelu. Za kluczowe parametry można uznać: zakres nośnika oraz liczbę punktów próbkowania nośnika. Inne, również istotne, parametry rozmywania są związane ze stopniem rozmywania i, w odróżnieniu od wyżej wymienionych, mają charakter indywidualny (inny dla każdego czynnika i każdej relacji rozmytej), w związku z czym mogą być dobierane w procesie adaptacji modelu.

Liczba punktów próbkowania nośnika jest w pewnym sensie tożsama z liczbą wartości lingwistycznych przedstawionych w podrozdziale 3.1.5. Zarówno intuicja, jak i badania [73, 168, 171] wskazują na związek tego parametru z dokładnością modelu. Zwiększanie liczby punktów próbkowania zwiększa, co prawda, tę dokładność, jednakże charakterystyka operacji algebraicznych na liczbach rozmytych (np. (4.7)) powoduje, że zwiększanie liczby punktów próbkowania nośnika bardzo wydłuża czas obliczeń komputera realizującego algorytm. Korzystne jest zatem

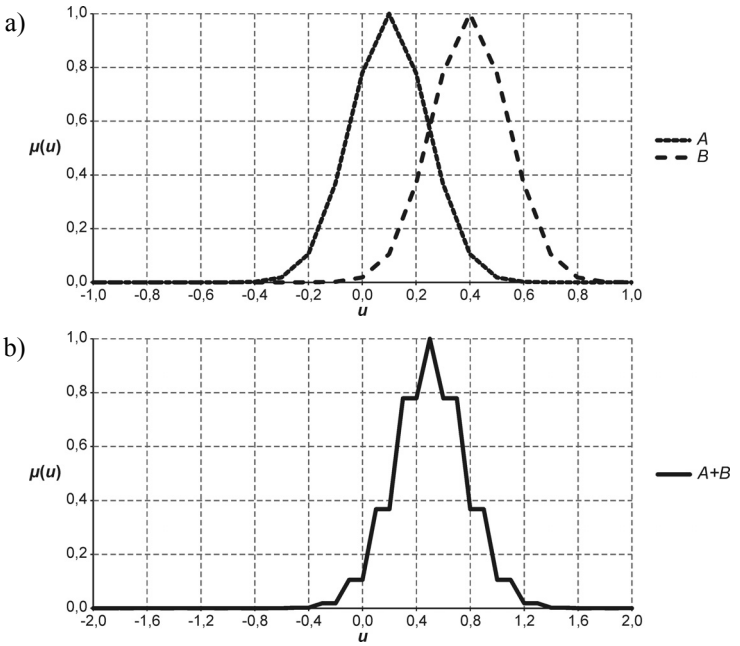
założenie możliwie niskiej wartości tego parametru. Z pewnych badań psychologicznych [130] wynika, że w większości zastosowań wystarczające walory informacyjne posiada liczba rozmyta odwzorowana przez 7 ± 2 wartości lingwistyczne, jednakże nie zawsze jest to wystarczające. Warto pamiętać, że niska liczba punktów próbkowania oznacza na ogół niższą dokładność pracy modelu. Zależność pomiędzy gęstością próbkowania nośnika a dokładnością przebiegów wynikowych funkcji przynależności pokazano na rysunkach 4.11 i 4.12 na przykładzie operacji dodawania rozmytego dwóch liczb rozmytych o centrach: 0,1 i 0,4 i funkcjach przynależności klasy G.

Jak widać na rysunkach 4.11 i 4.12 zmniejszanie liczby punktów próbkowania nośnika wprowadza znaczące zniekształcenia funkcji przynależności liczby rozmytej będącej wynikiem operacji.

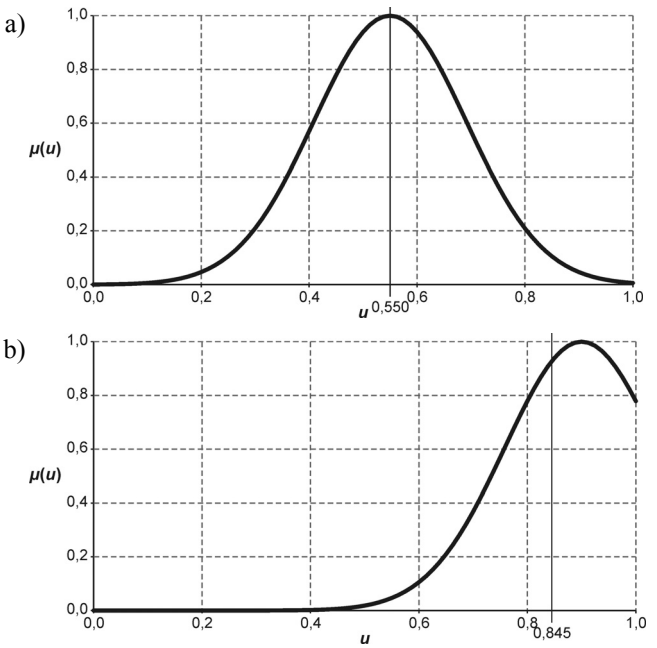
Kolejna trudność jest związana z ograniczeniem zakresu nośnika, koniecznym przy praktycznych realizacjach. Niektóre metody wyostrzania (takie jak metoda średniej ważonej) są wrażliwe na niesymetrię funkcji przynależności względem centrum liczby rozmytej. Zbliżanie tego centrum do granicy nośnika powoduje wzrost błędu wyostrzania. Rysunek 4.13 przedstawia wynik rozmycia (na nośniku $[0, 1]$) i ponownego wyostrzenia (metodą średniej ważonej) dwóch liczb o centrach: 0,55 i 0,90 i funkcjach przynależności klasy G.



Rys. 4.11. *Wynik (b) dodawania dwóch liczb rozmytych (a) określonych na nośniku o zakresie* $[-1, 1]$ *z liczbą punktów próbkowania nośnika* = 201



Rys. 4.12. *Wynik (b) dodawania dwóch liczb rozmytych (a) określonych na nośniku o zakresie* $[-1, 1]$ *z liczbą punktów próbkowania nośnika* $= 21$



Rys. 4.13. *Zależność wyniku wyostrzania liczby rozmytej od jej niesymetrii: a) liczba rozmyta wokół centrum* $0{,}55$*; b) liczba rozmyta wokół centrum* $0{,}9$*. Pionową ciągłą linią zaznaczono wyniki wyostrzania metodą średniej ważonej*

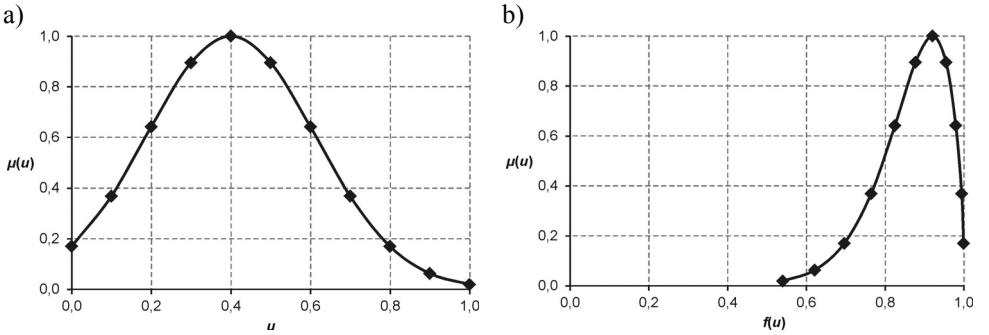
Jak widać na rysunku 4.13, przesunięcie jądra (centrum) liczby rozmytej w stronę granicy zakresu nośnika może doprowadzić do przekłamań w systemie decyzyjnym (ponieważ w systemie takim używa się przeważnie wartości wyostrzonych). Dla wyeliminowania problemu zilustrowanego na rysunku 4.13 można zastosować inne metody wyostrzania [88, 104, 121, 158, 224] (niektóre spośród nich wymieniono w podrozdziale 3.1.6), jednakże metoda średniej ważonej ma zaletę polegającą na pewnym odwzorowywaniu uogólnionego stopnia przynależności, co (niezależnie od prostoty aplikacyjnej) sprawia, że jest ona dogodna do stosowania. W tej sytuacji pozostaje poszerzenie zakresu nośnika do takich granic, przy których niesymetria będzie miała marginalne znaczenie. Z doświadczeń wynika, że w modelach operujących wartościami znormalizowanymi wystarczające jest poszerzenie bazowego zakresu nośnika do zakresu $[-2, 2]$. Konsekwencją takiego podejścia jest zwiększenie liczby punktów próbkowania nośnika (jeśli chce się zachować niezmienioną gęstość tego próbkowania), jednak metoda ta jest prosta implementacyjnie, dlatego też została wybrana do zastosowań praktycznych.

4.9. UCZENIE RELACYJNEJ ROZMYTEJ MAPY KOGTNITYWNEJ

Uczenie modelu ma kluczowe znaczenie dla jego prawidłowego działania. W modelach inteligentnych jest to właściwie jedyny sposób adaptacji parametrów modelu do realnych warunków pracy obiektu, przy tym, jeśli model ma służyć odwzorowaniu pracy rzeczywistego obiektu, to jego uczenie powinno mieć charakter nadzorowany – z wykorzystaniem historycznych danych odniesienia. W literaturze (np. w [10, 108, 136, 178, 184, 194, 207]) opisano różne podejścia do uczenia rozmytych map kognitywnych (FCM), jednakże większość z nich operuje relacjami, które są określone poprzez zbiory reguł albo poprzez moce powiązań reprezentowane przez liczby rzeczywiste. Operacje arytmetyczne w takich modelach (jeśli w ogóle są wykonywane) mają charakter działań na liczbach rzeczywistych, co warunkuje techniczne aspekty nie tylko funkcjonowania, ale także uczenia modelu. Operując wartościami liczbowymi, można implementować do map kognitywnych metody uczenia znane z innych zastosowań, np. algorytmy genetyczne [178], ewolucyjne [207] lub wielokrokowe [137], a w niektórych zastosowaniach – gradientowe metody optymalizacji [72]. Większość tego rodzaju metod uczenia opiera się na funkcyjnym przetwarzaniu wartości wyrażanych liczbowo, co w przypadku systemu opartego na liczbach rozmytych jest trudne i może wprowadzać dodatkowe błędy. Trzeba pamiętać, że funkcja, której argumentem jest liczba rozmyta, oddziałuje de facto na nośnik tej liczby (jak pokazują (3.149) i (3.150)), zmieniając przeważnie zarówno zakres, jak i parametry punktów próbkowania tego nośnika. Obrazuje to dokładniej rysunek 4.14, na którym pokazano efekt oddziaływania prostej funkcji (cosinus) na przykładową liczbę rozmytą klasy G.

Jednocześnie, z punktu widzenia poprawności pracy modelu, konieczne jest zachowanie niezmiennych parametrów nośnika. Zachodzi tu więc pewna sprzeczność, którą można przezwyciężyć, wprowadzając dodatkowe mechanizmy korygujące nośnik po każdej jego zmianie (tak jak opisano w podrozdziale 4.7). Mechani-

zmy takie w zasadzie już w modelu funkcjonują (przy interpretacji wyników dodawania i odejmowania rozmytego), ale ich powielanie jest niewskazane, ponieważ każde ich zadziałanie deformuje wynikowy kształt funkcji przynależności przetwarzanej liczby rozmytej. Wspomniane wyżej problemy obliczeniowe zawężają możliwości wyboru metody uczenia RRMK do grupy metod populacyjnych, przy czym bezpośrednie zastosowanie istniejących metod tego typu jest niemożliwe (z uwagi na charakter przetwarzanych danych), a to powoduje konieczność zaproponowania nowej metody, która tę specyfikę uwzględni.



Rys. 4.14. *Wynik działania funkcji na przykładową liczbę rozmytą A: a) liczba rozmyta A z funkcją przynależności klasy G; b)* $f(A) = cos\,(A)$. *Punktami oznaczono położenie singletonów rozmytych*

Niezależnie od założonego podejścia, należy przyjąć jakiś sposób oceny stopnia dopasowania modelu do odpowiadającego mu obiektu rzeczywistego. Można w tym celu zastosować znane z metod populacyjnych kryterium wykorzystujące tzw. funkcję celu czy współczynnik dopasowania (można też spotkać inne określenia). Na potrzeby stosowania takiego kryterium należy stworzyć wektor parametrów Q, którego elementami będą parametry modelu podlegające modyfikacji w procesie uczenia. Ogólna postać takiego wektora może przybrać następujący kształt:

$$Q = [\{q_1, \dots, q_m\}, k]^T \tag{4.21}$$

gdzie:

k – liczba punktów próbkowania nośnika;

m – liczba parametrów modelu modyfikowanych w procesie uczenia;

$\{q_1, \dots, q_m\}$ – zbiór parametrów modelu modyfikowanych w procesie uczenia (parametry te zależą od typu przyjętego modelu, a w szczególności od rodzaju funkcji przynależności relacji rozmytych).

Kryterium dla analizy dokładności, oparte o wykorzystanie wektora parametrów Q, może przybrać następującą postać:

$$J(Q) = \Phi(\|\bar{X}_i(t) - Z_i(t)\|) \Longrightarrow \min_{Q} \tag{4.22}$$

gdzie:
$\Phi()$ – wybrana funkcja optymalizacji (np. kwadratowa);
$\bar{X}_i(t), Z_i(t)$ – wyostrzona i ostra (zadana) trajektorie zmian wartości i-tego czynnika;
$\| \ \|$ – wybrana norma;
t – czas dyskretny.

W modelach dynamicznych system zwykle potrzebuje pewnego czasu (kilku lub kilkunastu kroków czasu dyskretnego) na wewnętrzną stabilizację po pobudzeniu zewnętrznym. Zależnie od celów modelowania procedura ucząca może dostosować parametry mapy do wskazywania prawidłowych wartości w jednym, wskazanym kroku takiego cyklu, bez przywiązywania wagi do kroków poprzednich (jest to metoda szybsza) lub w całym cyklu stabilizacji. Kryterium dla pojedynczego kroku czasu dyskretnego może przybrać postać współczynnika bliskości (4.23):

$$J_i(t) = \sqrt{\left(\bar{X}_i(t) - Z_i(t)\right)^2} \tag{4.23}$$

gdzie:
$J_i(t)$ – współczynnik bliskości i-tego czynnika w t-tym kroku czasu dyskretnego;
t – czas dyskretny;
$\bar{X}_i(t)$ – wyostrzona wartość i-tego czynnika relacyjnej rozmytej mapy kognitywnej w t-tym kroku czasu dyskretnego;
$Z_i(t)$ – ostra (zadana) wartość i-tego czynnika w t-tym kroku czasu dyskretnego,

natomiast dla opcji dopasowania przebiegu – współczynnika bliskości określonego równaniem (4.24):

$$J_i = \sqrt{\frac{1}{T}\sum_{t=1}^{T}\left(\bar{X}_i(t) - Z_i(t)\right)^2} \tag{4.24}$$

gdzie:
J_i – współczynnik bliskości i-tego czynnika;
T – czas pracy modelu dynamicznego (wyrażony w liczbie kroków czasu dyskretnego).

Równania (4.23) i (4.24) dotyczą badania pojedynczego czynnika. W ogólności procedurom minimalizacji można poddać współczynnik zawierający wartości wszystkich czynników. Przykładową formę takiego współczynnika bliskości dla wybranego kroku czasu dyskretnego przedstawia równanie (4.25), a dla przebiegu – (4.26):

$$J_t(Q) = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(\bar{X}_i(t) - Z_i(t)\right)^2} \tag{4.25}$$

$$J(Q) = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\frac{1}{T}\sum_{t=1}^{T}\left(\bar{X}_i(t) - Z_i(t)\right)^2} \tag{4.26}$$

gdzie:
n – liczba czynników relacyjnej rozmytej mapy kognitywnej.

W razie potrzeby zależności (4.25) i (4.26) można zmodyfikować tak, aby współczynnik bliskości był wyznaczany w oparciu jedynie o wybrane czynniki (zamiast wszystkich). Wtedy równania (4.25) i (4.26) przybierają postać (4.27) i (4.28):

$$J_t(Q) = \sqrt{\frac{1}{p}\sum_{i=1}^{n} y_i \cdot \left(\bar{X}_i(t) - Z_i(t)\right)^2} \tag{4.27}$$

$$J(Q) = \sqrt{\frac{1}{p}\sum_{i=1}^{n}\frac{1}{T}\sum_{t=1}^{T} y_i \cdot \left(\bar{X}_i(t) - Z_i(t)\right)^2} \tag{4.28}$$

gdzie:
$y_i = \begin{cases} 1 & \text{gdy } X_i \text{ jest uwzględniany przy obliczaniu współczynnika bliskości} \\ 0 & \text{gdy } X_i \text{ nie jest uwzględniany przy obliczaniu współczynnika bliskości} \end{cases}$
$p = \sum_{i=1}^{n} y_i$ – liczba czynników uwzględnianych przy obliczaniu współczynnika bliskości.

Metoda wykorzystująca którąś z zależności (4.23)–(4.30) polega, ogólnie rzecz ujmując, na sukcesywnym wykonywaniu następujących kroków: modyfikacji parametrów modelu, wyznaczeniu wyostrzonych wartości czynników modelu, obliczaniu wartości funkcji dopasowania J według (4.23–4.29) lub (4.30) i powtarzaniu tego cyklu do momentu uzyskania satysfakcjonującej (tzn. wystarczająco niskiej) wartości J. W ogólnym podejściu modyfikowanymi parametrami mogą być zarówno właściwości czynników, jak i relacji, jednakże w praktyce dogodnie jest ograniczyć działanie algorytmów uczących do właściwości relacji rozmytych, dobierając stałe parametry rozmywania czynników w oparciu o wiedzę ekspertową.

Dla modelu, w którym relacje rozmyte budowane są z użyciem funkcji przynależności klasy G wektor parametrów Q przybiera następującą postać:

$$Q = \left[\{\bar{r}_{i,j}\}, \{\sigma_{i,j}\}, k\right]^T \tag{4.29}$$

gdzie:

k – liczba punktów próbkowania nośnika;

$\{\bar{r}_{i,j}\}$ – współczynniki kierunkowe funkcyjnych współczynników mocy relacji rozmytych $R_{i,j}$ $(i, j = 1, \ldots, n;\ i \neq j)$ – jak np. w (4.16);

$\{\sigma_{i,j}\}$ – współczynniki rozmycia relacji rozmytych $R_{i,j}$ $(i, j = 1, \ldots, n;\ i \neq j)$ – jak np. w (4.16);

n – liczba czynników modelu.

Przy relacjach rozmytych innych klas wektor Q zawiera po prostu inne parametry, jednak ogólna idea pozostaje niezmieniona.

Kluczowym elementem procesu uczenia modelu jest sposób modyfikacji parametrów, który w efekcie doprowadzi do minimalizacji funkcji dopasowania. Jak już wspomniano, stosowanie istniejących metod populacyjnych w odniesieniu do modeli RRMK nie jest możliwe, jeśli chce się zachować rozmyty charakter modelu na wszystkich etapach jego budowy. Biorąc to pod uwagę, dla modeli relacyjnej rozmytej mapy kognitywnej, opracowano metodę uczenia nadzorowanego wykorzystującą algorytm **kolejnych przybliżeń ze zmiennym krokiem zmian parametrów** [176], którego podstawą jest badanie różnic pomiędzy wyostrzonymi wartościami czynników uczonej relacyjnej rozmytej mapy kognitywnej, a odpowiadającymi im ostrymi wartościami czynników systemu odniesienia. Jest to metoda, którą typologicznie można zaliczyć do grupy metod populacyjnych, a jej istota opiera się na dokonywaniu drobnych zmian wartości kluczowych parametrów kolejnych relacji rozmytych i obserwacji skutków tych zmian poprzez badanie wartości odpowiednio skonstruowanej funkcji celu lub funkcji bliskości. W metodzie tej celem analizy obliczeniowej modelu dynamicznego jest rozważenie związku pomiędzy dokładnością działania a głównymi parametrami rozmytego systemu obliczeniowego, do których należą parametry rozmywania (centra i współczynniki rozmycia) oraz liczba wybranych wartości lingwistycznych (punktów próbkowania nośnika) relacji rozmytych. Jak wspomniano w podrozdziale 4.5, różne typy relacji rozmytych charakteryzują się różnymi liczbami kluczowych parametrów, co ma duże znaczenie w metodzie kolejnych przybliżeń, ponieważ analizuje się zmiany wyników pracy modelu w odpowiedzi na zmiany każdego z kluczowych parametrów. Najmniejszą ich liczbę posiada relacja rozmyta typu G, w związku z czym przedstawiony poniżej opis metody uczenia będzie się posługiwał własnościami tego właśnie typu relacji. Adaptację relacji innych typów przeprowadza się w analogiczny sposób.

Z formalnego punktu widzenia, algorytm kolejnych przybliżeń ze zmiennym krokiem zmian parametrów, wykorzystywany w procesie uczenia (adaptacji) relacyjnej rozmytej mapy kognitywnej, buduje się w oparciu o trzy elementy [176]:

– dynamiczne związki pomiędzy $X_i(t+1)$ oraz $X_i(t)$ w analizowanym obiekcie i modelu typu (4.3),

– kryterium adaptacji (4.22),
– algorytm adaptacji relacji rozmytych, którego formalną postać odzwierciedla (4.30):

$$R_{i,j}(t+1) = R_{i,j}(t) + \Delta R_{i,j}(t) \tag{4.30}$$

gdzie:
$\Delta R_{i,j}(t)$ – „przyrost" relacji rozmytej $R_{i,j}$ ($R_{i,j}(0)$ – relacja rozmyta o początkowych wartościach parametrów).

Wyrażenie „$+\ \Delta R_{i,j}(t)$" w równaniu (4.30) ma charakter do pewnego stopnia zastępczy – symbolizuje zmiany wszystkich parametrów relacji rozmytej wprowadzone w pojedynczym kroku adaptacji (powiązanym z wybranym punktem czasu dyskretnego t). Poszukiwanie wartości relacji rozmytej $R_{i,j}$ dla kolejnych kroków czasu dyskretnego sprowadza się do poszukiwania kształtów funkcji przynależności tej relacji takich, dla których wybrany współczynnik bliskości modelu (np. wg (4.26)) będzie osiągać minimalne wartości. Tak więc, ogólną zależność (4.30) można doprecyzowywać poprzez podanie nowego przebiegu funkcji przynależności zależnej od założonego typu modelu. Na przykład dla funkcji przynależności opisanej równaniem (4.18), formalna postać algorytmu adaptacji przybierze następujący kształt:

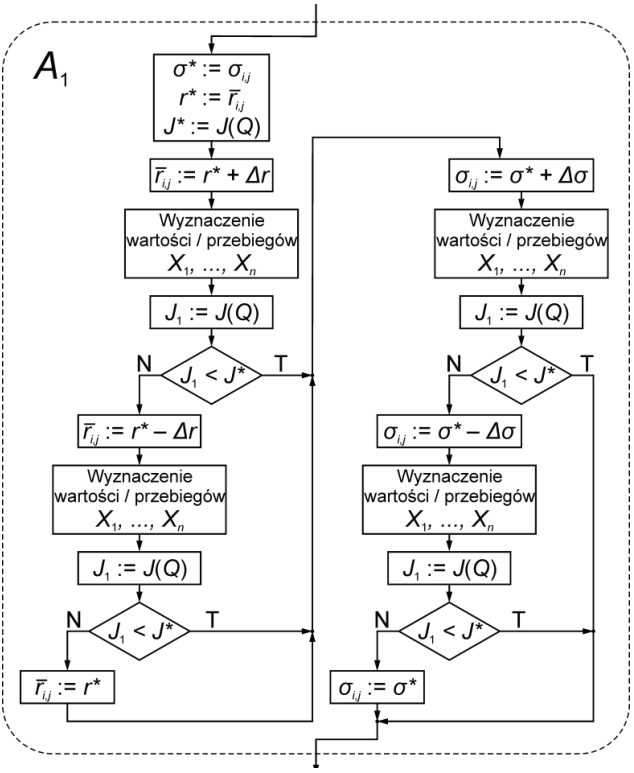
$$\mu_{R_{i,j}}(t+1) = e^{-\left(\frac{u_j - \bar{r}_{i,j}(t+1)\cdot u_i}{\sigma_{i,j}(t+1)}\right)^2} \tag{4.31}$$

gdzie:
$\bar{r}_{i,j}(t+1) = \bar{r}_{i,j}(t) + \Delta\bar{r}_{i,j}$;
$\Delta\bar{r}_{i,j}$ – zmiana wartości współczynnika kierunkowego liniowej funkcji mocy relacji pomiędzy czynnikami i-tym i j-tym;
$\sigma_{i,j}(t+1) = \sigma_{i,j}(t) + \Delta\sigma_{i,j}$;
$\Delta\sigma_{i,j}$ – zmiana wartości współczynnika rozmycia relacji rozmytej pomiędzy czynnikami i-tym i j-tym.

Analityczne wyznaczenie wartości $\Delta R_{i,j}(t)$ z równania (4.30), co odpowiada wyznaczeniu, np. wartości $\Delta\bar{r}_{i,j}$ i $\Delta\sigma_{i,j}$ z równania (4.31), nie jest możliwe, a to z powodu charakteru modelu relacyjnej rozmytej mapy kognitywnej (np. (4.3)). Sprawia on, że funkcja współczynnika bliskości (np. (4.26)), która przecież posługuje się wyostrzonymi wartościami czynników, nie poddaje się działaniu klasycznych metod różniczkowania (dotyczy to zresztą także podobnych form funkcji dopasowujących stosowanych w innych modelach inteligentnych – np. w FCM). Jest to główny powód, dla którego należało opracować nową metodę uczenia modelu.

Głównym celem prac nad modelem relacyjnej rozmytej mapy kognitywnej było znalezienie metody pozwalającej budować modele rozmytych map kognitywnych, które będą z jednej strony posługiwały się możliwie niską liczbą wartości lingwistycznych, a z drugiej – pozwolą na uzyskanie dobrej dokładności. Z tego powodu działanie

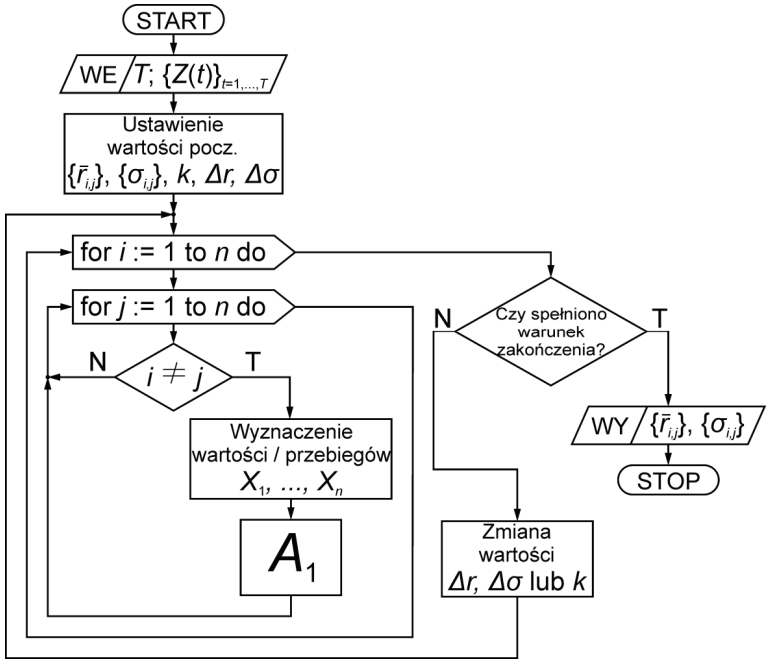
algorytmu uczącego rozpoczyna się od zadeklarowania możliwie niskiej liczby punktów próbkowania nośnika k (np. 5), po czym aktywuje się właściwy algorytm adaptacyjny. Jeśli założona wartość współczynnika nie zostanie osiągnięta, to zwiększa się wartość k i ponownie aktywuje algorytm adaptacyjny. Ogólnie rzecz ujmując, adaptacja polega na dokonywaniu sukcesywnych zmian (z określonymi skokami) parametrów $\bar{r}_{i,j}$ oraz $\sigma_{i,j}$ każdej relacji rozmytej $R_{i,j}$ z jednoczesnym śledzeniem wartości współczynnika bliskości. Działanie wspomnianych wyżej metod populacyjnych opiera się na podobnej zasadzie, jednakże przeważnie zmiany parametrów są w nich generowane losowo (tzn. stosuje się losowy wybór wartości zmienianego parametru oraz, niekiedy, losowo określa się wielkość tej zmiany). W modelu RRMK takie podejście byłoby utrudnione, ponieważ pojedyncza relacja rozmyta jest opisana więcej niż jednym parametrem. Ponadto oparcie się na losowym doborze modyfikowanych parametrów wprowadza do systemu uczenia dodatkowy czynnik niepewności, który, w pewnych sytuacjach, może doprowadzić do pominięcia rozwiązania optymalnego. Z tego powodu zdecydowano o użyciu zautomatyzowanego podejścia deterministycznego, w którym zarówno kolejność modyfikowanych elementów (relacji rozmytych), jak i kierunek tych modyfikacji są ściśle określone. Ogólny algorytm pojedynczego kroku adaptacji pojedynczej relacji rozmytej przedstawiono, w postaci pseudokodu, na rysunku 4.15.



Rys. 4.15. *Ogólny schemat działań algorytmu uczącego w pojedynczym kroku adaptacji pojedynczej relacji rozmytej* $R_{i,j}$

Na rysunku 4.15 widać powtarzającą się sekwencję działań (dla zmian $\bar{r}_{i,j}$ oraz $\sigma_{i,j}$). W modelach posługujących się relacjami rozmytymi zbudowanymi w oparciu o inne (niż klasy G) funkcje przynależności wykorzystuje się podobny algorytm, przy czym liczba wspomnianych sekwencji odpowiada liczbie modyfikowanych parametrów. Dla modeli, w których stosowane są relacje różnych typów sytuacja jest o tyle bardziej złożona, że każdy typ relacji rozmytej jest modyfikowany w oparciu o własny typ modułu adaptacyjnego A_1 (rys. 4.15) – poszczególne moduły różnią się głównie liczbą wyżej wymienionych sekwencji obliczeniowych. Funkcja $J(Q)$, do której odwołuje się algorytm przedstawiony na rysunku 4.15, jest wybraną dla danego modelu funkcją współczynnika bliskości – może ona być wyznaczana np. według (4.23)–(4.26) lub inną metodą dogodną z punktu widzenia celów modelowania. Blok „wyznaczania wartości/przebiegów X_1, ..., X_n" służy do obliczenia rozmytych wartości czynników zgodnie z przyjętym modelem (np. wg (4.3)).

Rysunek 4.15 przedstawia działania dla pojedynczej, wybranej relacji rozmytej wiążącej czynniki i-ty i j-ty. Pełny model zawiera wiele takich relacji (ich łączna liczba dla n czynników wynosi $\binom{n}{2}$), w związku z czym kompletny algorytm uczenia jest oczywiście bardziej złożony. Jego ogólny schemat blokowy pokazano na rysunku 4.16.



Rys. 4.16. *Ogólny algorytm uczenia (metodą kolejnych przybliżeń ze zmiennym krokiem zmian parametrów) relacyjnej rozmytej mapy kognitywnej. A_1 – moduł algorytmu dla pojedynczej relacji rozmytej (jak na rys. 4.15); n – liczba czynników; T – zakres czasu dyskretnego będący przedmiotem uczenia; $\{Z\}$ – zbiór zadanych wartości czynników (historyczne dane odniesienia)*

Algorytm przedstawiony na rysunku 4.16 sprawdza „warunek zakończenia". Warunek ten musi być spełniony, aby algorytm zakończył pracę. Formułuje się go jako alternatywę dwóch warunków cząstkowych: uzyskania określonej wartości współczynnika bliskości lub wykonania określonej, granicznej liczby obiegów. Dwa wyżej wymienione warunki cząstkowe można uzupełniać dodatkowymi, mającymi na celu uniknięcie niepotrzebnych cykli obliczeniowych. Mogą to być: brak zmian (lub osiągnięcie granicznej niskiej wartości zmian) współczynnika bliskości w kolejnych cyklach adaptacji lub przekroczenie zaplanowanego czasu trwania obliczeń.

Pewnym problemem, który zawsze występuje w modelach tego rodzaju, jest określenie początkowych wartości parametrów. W modelach opisywanych liczbami rzeczywistymi wartości te są przeważnie losowane. W modelu RRMK o relacjach rozmytych metoda losowania wartości również może być wykorzystana, ale w ograniczonym zakresie – losowaniu można poddawać początkowe wartości współczynników mocy relacji ($\bar{r}_{i,j}$). Współczynniki rozmywania ($\sigma_{i,j}$) są dodatkowo powiązane z liczbą kroków próbkowania nośnika (k). Przeprowadzone badania [73, 77, 83] wykazały, że zbyt niski współczynnik rozmywania relacji niekorzystnie wpływa na dokładność. Dla zadanej wartości k istnieje pewna wartość σ, dla której dokładność modelu jest najwyższa. W związku z tym, początkowe wartości $\sigma_{i,j}$ lepiej jest określać w oparciu o wiedzę ekspertową, zależnie od założonej początkowej wartości k – można wtedy ustalić pewną początkową wartość σ wspólną dla wszystkich relacji, a następnie później, w procesie uczenia, określić zindywidualizowany współczynnik rozmycia dla każdej relacji. Jeśli chodzi o wartość k, to im jest ona wyższa, tym wyższa jest dokładność modelu, jednakże wzrost wartości k pociąga za sobą znaczące wydłużenie czasu obliczeń i zwiększenie obciążenia pamięci komputera, co wynika z charakteru maksyminowych operacji arytmetyki liczb rozmytych. Stąd też bardzo istotne w procesie uczenia modelu jest określenie „akceptowalnej" dokładności, która powinna być osiągnięta przy możliwie najniższej wartości k.

Opracowana metoda ma za zadanie znalezienie jednego parametru wspólnego dla całego modelu (tzn. k) oraz zestawu indywidualnych parametrów każdej relacji rozmytej (tzn. $\bar{r}_{i,j}$ oraz $\sigma_{i,j}$), takich, przy których zostanie osiągnięta akceptowalna wartość współczynnika bliskości (dopasowania) dla wskazanego czynnika ((4.23) lub (4.24)) lub jego sumarycznego odpowiednika dla całego systemu ((4.25) lub (4.26)). Określenie „akceptowalna" zostało użyte także dlatego, że z charakteru metody wynika, iż globalnie rozumiana minimalna wartość współczynnika bliskości może pozostać niewykryta, a algorytm zatrzyma się w jakimś minimum lokalnym. Wtedy do eksperta będzie należała decyzja co do dalszego postępowania – czy kontynuować poszukiwania, czy też uznać, że uzyskany wynik jest wystarczający. Czynnikiem pomocnym w unikaniu lokalnych minimów jest stosowanie zmiennego kroku zmian parametrów – od stosunkowo dużych (rzędu $0{,}1$) poprzez stopniowo malejące (do poziomu $0{,}005$) wartości zarówno Δr (krok zmian współczynnika mocy relacji), jak i $\Delta\sigma$ (krok zmian współczynnika rozmycia relacji).

Warto zauważyć, że wśród kluczowych parametrów podlegających adaptacji umieszczono współczynniki kierunkowe funkcyjnych współczynników mocy relacji rozmytych $\bar{r}_{i,j}$, a nie pełne postaci funkcyjne tych mocy (czyli $r_{i,j}(u)$). Uczyniono tak, ponieważ przedstawiony na rysunku 4.16 ogólny algorytm uczenia działa w oparciu o założenie, że wszystkie funkcje $r_{i,j}(u)$ są liniowe. Takie założenie upraszcza algorytm uczenia i skraca czas jego działania, co oczywiście nie oznacza, że musi być stosowane. Należy jednak pamiętać, że wprowadzanie współczynników mocy relacji w postaci funkcji innych niż liniowe, choć nie zmienia w zasadzie sposobu pracy gotowego modelu, ma zasadniczy wpływ na algorytm uczenia. Po pierwsze, zachodzi problem doboru odpowiedniej funkcji dla każdej relacji rozmytej $r_{i,j}(u)$. Po drugie, funkcja taka jest opisana przez pewną liczbę parametrów, z których każdy również powinien być przedmiotem działania algorytmu uczącego, co wydłuża czas jego działania. Po trzecie, jeśli każda relacja rozmyta ma własną funkcję mocy (różną od pozostałych relacji), to trudno jest stworzyć jednolity algorytm uczący. Trwają obecnie prace nad jednolitym rozwiązaniem tego problemu. Najbardziej obiecujące wydaje się podejście parametryczne, w którym kształty funkcji $r_{i,j}(u)$ są determinowane przy użyciu krzywych Béziera, co ma tę zaletę, że każda taka funkcja (nawet o stosunkowo złożonym kształcie) może być opisana zaledwie czterema parametrami, a to z kolei ułatwia automatyzację pracy algorytmu.

Działanie opisanego wyżej algorytmu uczenia RRMK zostało przedstawione, na przykładach, w podrozdziale 5.2.

5 ANALIZA DZIAŁANIA WYBRANYCH METOD MODELOWANIA ZŁOŻONYCH SYSTEMÓW

Zastosowania zaproponowanych w poprzednich rozdziałach metod modelowania przy użyciu zarówno RMK, jak i RRMK mogą być różnorakie, począwszy od rozwiązywania zadań klasyfikacji i identyfikacji, poprzez monitorowanie pracy z jednoczesnym działaniem procedur decyzyjnych, aż po odwzorowywanie przebiegów czasowych wybranych wielkości. Warto zwrócić uwagę, że, o ile pewne techniki działania RMK (poza niezwykle istotnymi technikami uczenia) były już wcześniej stosowanie w modelach FCM, to podejście wykorzystujące RRMK jest nowe, zarówno pod względem pracy modelu, jak i technik jego uczenia.

Poniżej zaprezentowano wybrane przykłady aplikacyjnych zastosowań RMK oraz RRMK. Do ich prezentacji posłużono się aplikacjami stworzonymi specjalnie do celów badania wyżej wymienionych modeli.

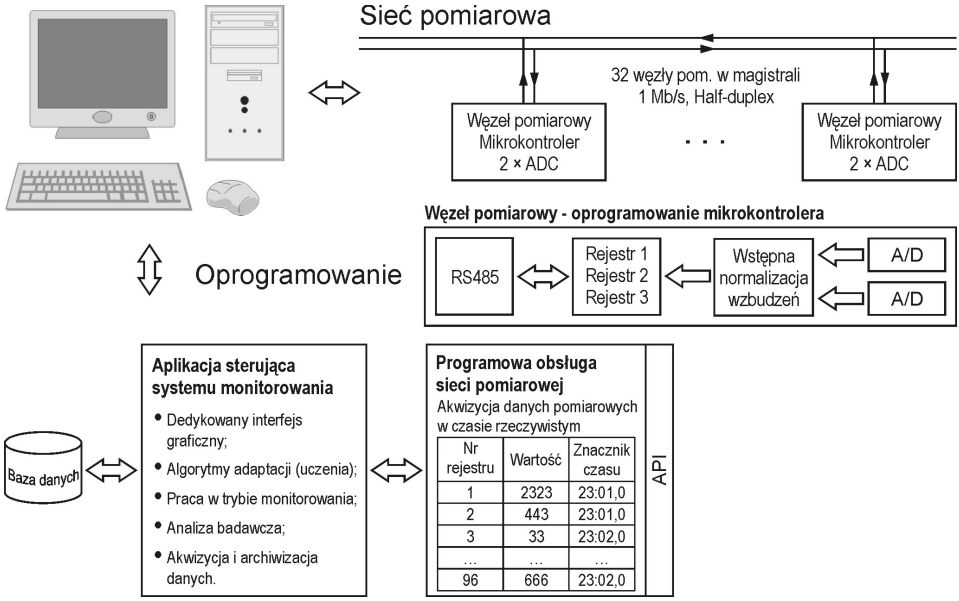
5.1. PRZYKŁADOWA REALIZACJA APLIKACYJNA RELACYJNEJ MAPY KOGNITYWNEJ

Metoda monitorowania pracy złożonego systemu wykorzystującego Relacyjną Mapę Kognitywną (RMK) zostanie przedstawiona na przykładzie systemu monitorowania zaprojektowanego w ramach projektu badawczego nr N N510 468136 (autor był jednym z wykonawców ww. projektu) [172]. W metodzie tej wykorzystuje się RMK o charakterze ostrym, w której wszystkie parametry mają wymiar liczbowy na wszystkich etapach modelowania (projektowanie, uczenie i eksploatacja modelu). Charakterystykę struktury takiego typu RMK oraz algorytmów jej uczenia i pracy przedstawiono w rozdziale 2.

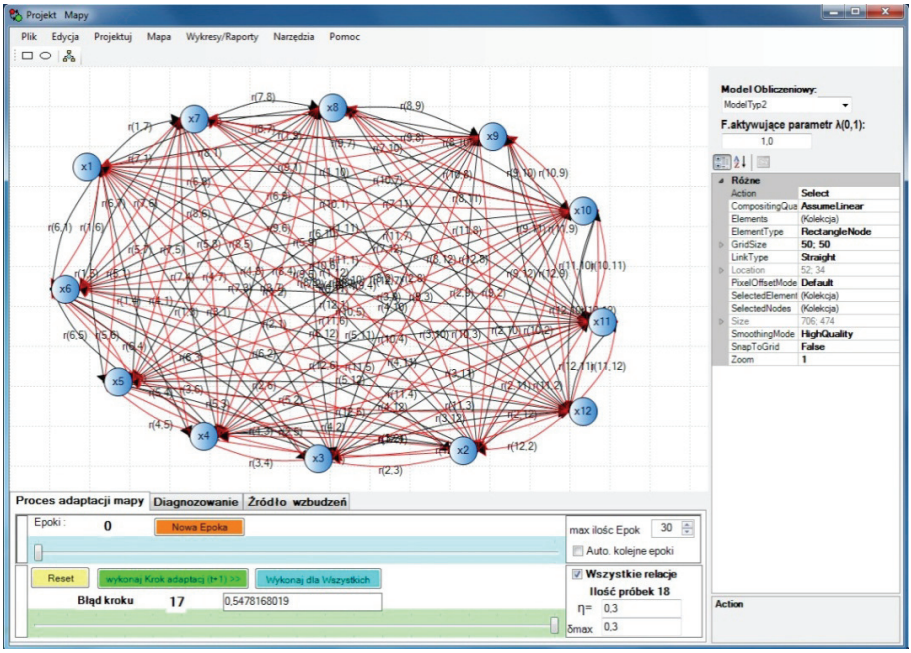
Proces projektowania i uczenia, a następnie praktycznej eksploatacji RMK, jest wspomagany przez specjalnie w tym celu stworzony system komputerowy, którego ogólną strukturę pokazano na rysunku 5.1. System ten składa się z dwóch powiązanych ze sobą części: aplikacji komputerowej, opracowanej z wykorzystaniem platformy Microsoft.Net 4.0 (panel sterujący tej aplikacji przedstawia rys. 5.2) oraz sieci pomiarowej obejmującej dziesięć niezależnych węzłów pomiarowych zbudowanych na bazie mikrokontrolerów ATMEGA8L (widok pojedynczego węzła przedstawiono na rys. 5.3).

Zadaniem aplikacji jest wsparcie procesu tworzenia oraz eksploatacji RMK. W ramach modułów konstrukcyjnych system pozwala zaprojektować strukturę RMK, dokonywać w niej zmian w oparciu o wiedzę ekspertową oraz prowadzić uczenie (metodą opisaną w rozdziale 2) z wykorzystaniem danych generowanych na bieżąco (przez wzorcową mapę odniesienia) lub pobieranych z przygotowanego wcześniej pliku dyskowego. Zbudowana w ten sposób RMK może następnie zostać wykorzystana do badań symulacyjnych (w których źródłem danych są sygnały

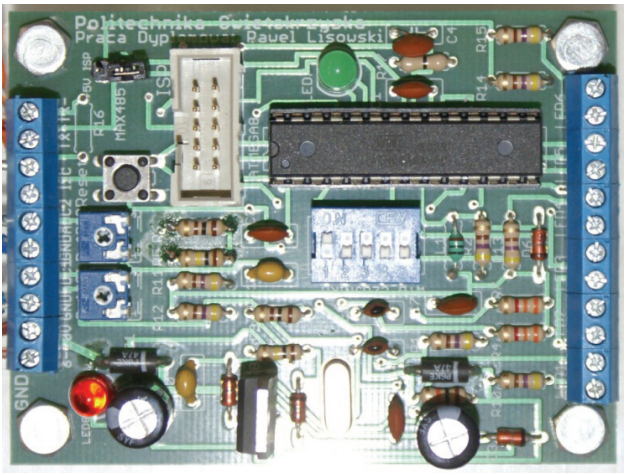
generowane lub pobierane z pliku) lub do monitorowania rzeczywistego obiektu, podczas którego system otrzymuje dane z sieci pomiarowej.



Rys. 5.1. *Ogólna struktura systemu monitorującego wykorzystującego RMK*



Rys. 5.2. *Interfejs graficzny aplikacji sterującej systemem wsparcia monitorowania z użyciem RMK*



***Rys. 5.3.** Widok pojedynczego węzła pomiarowego zbudowanego w oparciu o mikrokontroler ATMEGA8L*

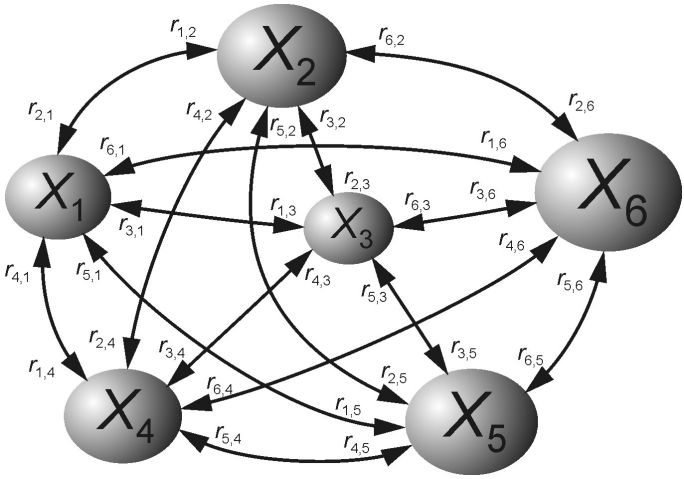
Każdy węzeł sieci pomiarowej umożliwia pomiar oraz adaptację dwóch sygnałów napięciowych z dedykowanych czujników. Wspólna magistrala komunikacyjna oparta na standardzie RS485 zapewnia komunikację z komputerem z szybkością 1 Mb/s. W zależności od wymagań sieć pomiarowa może, w ramach jednej magistrali, obsłużyć do 32 węzłów (64 sygnały pomiarowe). Cyfrowa transmisja danych zapewnia większą odporność na zakłócenia oraz zwiększa dopuszczalną odległość węzłów pomiarowych od komputera sterującego. Sterowanie i pobieranie danych z węzłów pomiarowych odbywa się za pomocą komunikatów alfanumerycznych sformatowanych w siedmiobajtowe ramki. Komunikacja inicjowana jest po stronie komputera sterującego poprzez cykliczne przepytywanie węzłów, z których każdy posiada unikatowy adres, dzięki czemu w danym momencie pobierane są informacje tylko z tego węzła, który został zadeklarowany w ramce komunikacyjnej. Komunikacja może być obsługiwana poprzez dowolny terminal tekstowy wyposażony w port szeregowy. Poszczególne mikrokontrolery zostały oprogramowane przy użyciu dedykowanego środowiska programistycznego AVR Studio oraz programatora typu STK 500.

W opracowanym systemie zastosowano dynamiczny model RMK zgodny z (2.8) o strukturze modelu nieliniowego z nieliniowością szybkości zmian zdefiniowanego równaniem (2.12). Metoda uczenia modelu została w związku z tym zdeterminowana układem równań (2.30), przy tym poszczególne jego elementy mogły przybierać różne formy zależnie od zakładanej postaci funkcji progowej f_d.

Przykład 5.1. *Zastosowanie RMK do modelowania pracy układu elektrotechnicznego*

Przykładem działania systemu monitorowania wykorzystującego RMK jest monitorowanie pracy jednego z obwodów elektrycznych pojazdu samochodowego – obwodu zasilania energią elektryczną. Wydzielono w nim sześć kluczowych czynników,

oznaczonych następująco: X_1 – napięcie na zaciskach akumulatora, X_2 – napięcie regulowane (napięcie wyjściowe alternatora z regulatorem napięcia), X_3 – prąd wzbudzenia alternatora, X_4 – sygnał napięciowy ze stacyjki pojazdu (para styków nr 1), X_5 – sygnał zapłonowy ze stacyjki pojazdu, X_6 – stan alternatora (odpowiadający ocenie poprawności pracy alternatora). Celem testu jest zbudowanie RMK, która posłużyłaby do rozpoznawania stanu alternatora w oparciu o dostępne informacje pomiarowe z czujników umieszczonych w obiekcie badawczym (był nim pojazd typu Daewoo Lanos). Dane uczące uzyskano drogą eksperymentalną na stanowisku pomiarowym wyposażonym w wyżej wymieniony pojazd. Przyjęto zerowe wartości początkowe współczynników mocy wszystkich relacji RMK. Wstępnie zaprojektowana struktura RMK miała postać jak na rysunku 5.4.



Rys. 5.4. *Graficzna reprezentacja struktury RMK zaprojektowanej do modelowania pracy obwodu zasilania energią elektryczną w pojeździe samochodowym.* $X_1 \div X_6$ *– kluczowe czynniki;* $\{r_{i,j}\}$ *– współczynniki mocy relacji pomiędzy czynnikami;* $i, j = 1, \ldots, 6$*;* $i \neq j$

Algorytm uczenia wykorzystywał metodę zgodną z (2.30), przy czym oceny dopasowania dokonywano w oparciu o kryterium (2.18). Z uwagi na specyfikę pracy badanego systemu zastosowano tabelaryczną metodę normalizacji wartości czynników (jak na rys. 2.2a).

Czynniki podzielono umownie na dwie grupy: czynniki o charakterze symptomowym, których wartości mogą być odczytane na podstawie pomiarów ($X_1 \div X_5$) oraz czynnik o charakterze decyzyjnym, którego wartość może być podstawą wnioskowania (X_6). Wynika stąd, że z punktu widzenia celów modelowania istotne jest prawidłowe odwzorowanie jedynie wartości czynnika X_6.

Uwaga 5.1

Warto podkreślić, że zgodnie z ogólną definicją struktury mapy kognitywnej, niektóre czynniki mają charakter ilościowy (odzwierciedlają mierzalne wartości parametrów

systemu), a inne (X_6) są czynnikami o charakterze jakościowym, których liczbowe reprezentacje mogą być ustalane arbitralnie w trakcie projektowania modelu.

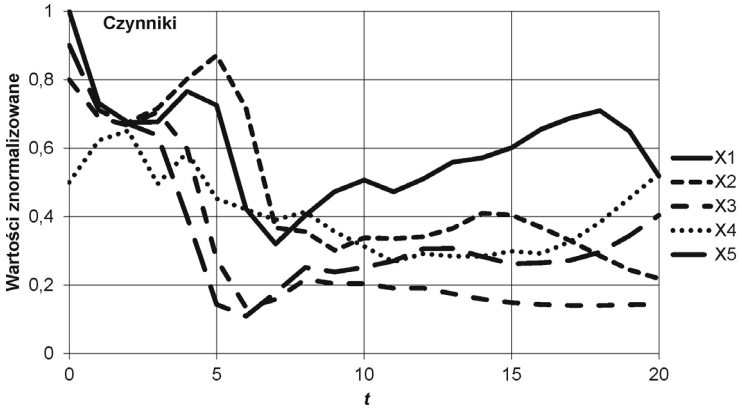
W kolejnych krokach czasu dyskretnego system otrzymywał dane uczące (kompletne), dostosowywał do nich strukturę RMK (poprzez modyfikacje współczynników mocy relacji), po czym przy użyciu tak zmodyfikowanej mapy, dokonywał symulacji działania układu. Wyniki tej symulacji były następnie porównywane z danymi wzorcowymi (wg kryterium (2.18)), co stanowiło bazę do kolejnych modyfikacji w kolejnych krokach czasu dyskretnego.

Efektem uczenia było uzyskanie finalnych wartości współczynników mocy poszczególnych relacji RMK – zestawiono je w tabeli 5.1.

Tabela 5.1. *Wynikowe wartości współczynników mocy relacji RMK zbudowanej na potrzeby modelowania pracy obwodu zasilania energią elektryczną w pojeździe samochodowym*

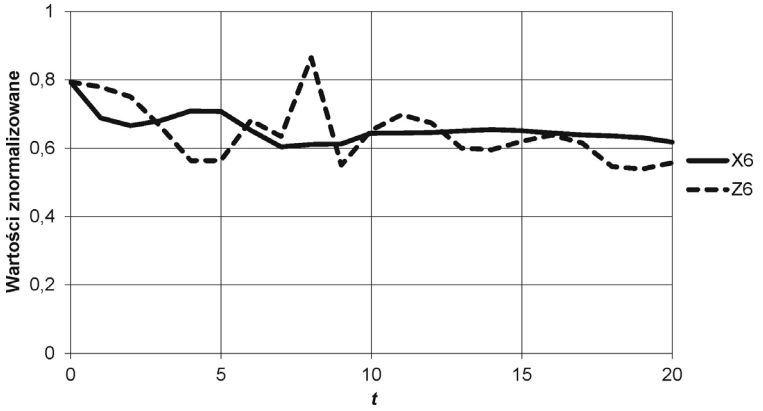
r	X_1	X_2	X_3	X_4	X_5	X_6
X_1	0,00	−0,84	−0,92	−0,03	−0,29	−0,17
X_2	−0,38	0,00	−0,91	−0,48	−0,54	−0,05
X_3	−0,47	−0,64	0,00	−0,35	−0,60	−0,10
X_4	−0,43	−0,52	−0,66	0,00	−0,29	−0,07
X_5	−0,57	−0,58	−0,81	−0,22	0,00	−0,11
X_6	−0,68	−0,76	−0,88	0,03	−0,11	0,00

Badaniu poddano stan niepoprawnej pracy alternatora. Na rysunku 5.5 przedstawiono przebiegi wartości czynników symptomowych (znormalizowanych) w kolejnych krokach czasu dyskretnego.



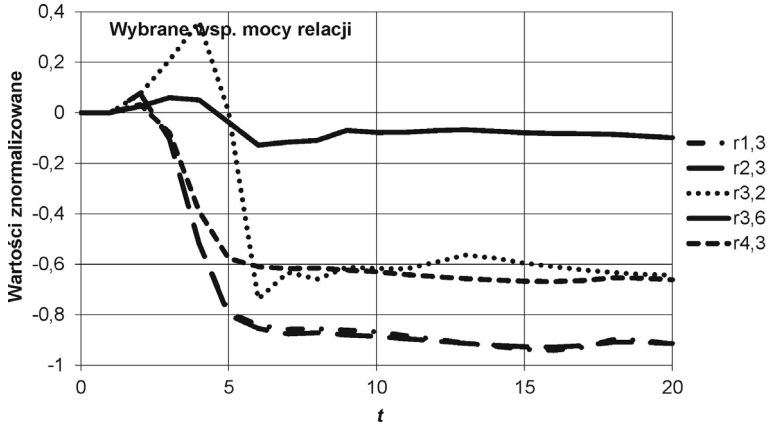
Rys. 5.5. *Znormalizowane wartości czynników symptomowych w kolejnych krokach czasu dyskretnego, uzyskane przez system monitorujący;* t *– czas dyskretny*

Rysunek 5.6 przedstawia przebieg wartości czynnika decyzyjnego X_6 (wygenerowany z użyciem RMK) zestawiony z wartościami odpowiedniego sygnału uczącego (Z_6).



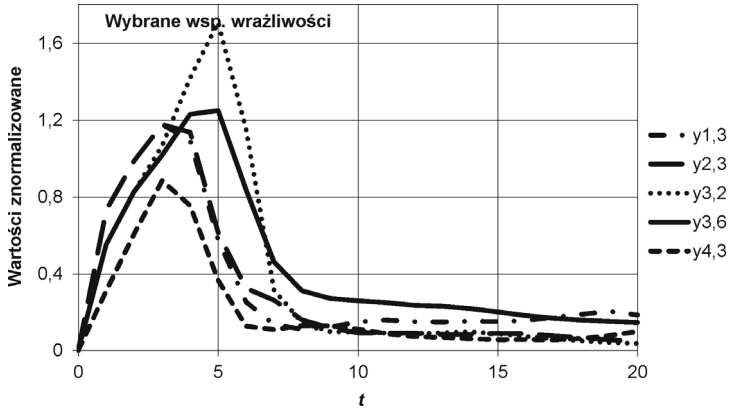
Rys. 5.6. *Znormalizowane wartości czynnika decyzyjnego w kolejnych krokach czasu dyskretnego. X_6 – wartość wyznaczona symulacyjnie przez system monitorujący; Z_6 – wartość wzorcowa (ze zbioru uczącego); t – czas dyskretny*

Z przebiegów zestawionych na rysunku 5.6 wynika, że w procesie stopniowej adaptacji system monitorujący uzyskuje stan, w którym z dużą dokładnością odwzorowuje wartości rzeczywiste czynnika decyzyjnego, wykorzystując do tego celu wprowadzane wartości czynników symptomowych. Proces dochodzenia do końcowych wartości poszczególnych relacji zilustrowano na rysunku 5.7, na którym pokazano wartości wybranych współczynników mocy relacji, uzyskiwane w kolejnych krokach procesu adaptacji. RMK zawiera 30 takich relacji. Na rysunku 5.7 uwzględniono tylko kilka spośród nich, a to w celu zwiększenia przejrzystości przekazu.



Rys. 5.7. *Wartości wybranych współczynników mocy relacji; t – czas dyskretny*

Pewnym wyznacznikiem tempa i jakości procesu adaptacji może być analiza przebiegów poszczególnych współczynników wrażliwości ($y_{i,j}$). Powinny one dążyć do ustabilizowania się na relatywnie niskich poziomach. Z rysunku 5.8, na którym przedstawiono przebiegi wybranych współczynników wrażliwości, jasno wynika, że tak właśnie jest w prezentowanym przykładzie.



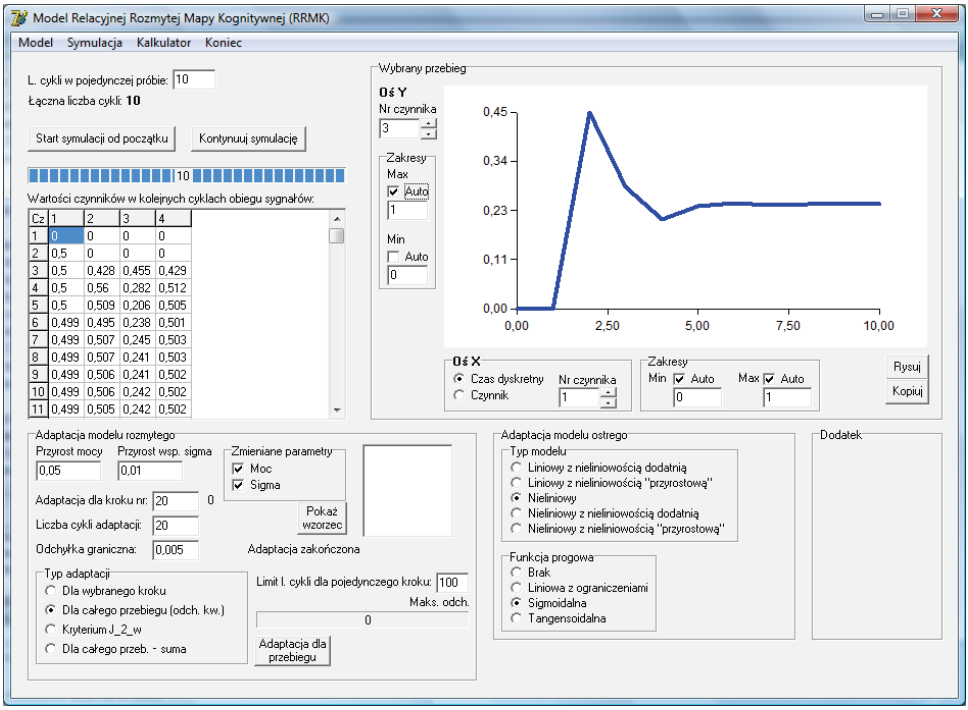
Rys. 5.8. *Wartości wybranych współczynników wrażliwości;* t *– czas dyskretny*

Przedstawione wyżej dane ilustrują (na przykładzie złożonego systemu elektrotechnicznego) działanie modelu RMK w zastosowaniu do odwzorowywania wartości wybranych parametrów systemu. Specyfika środowiska pracy oraz charakter czynników sprawiają, że klasyczne modelowanie takiego systemu byłoby trudne, podczas gdy użycie RMK czyni ten proces stosunkowo łatwym.

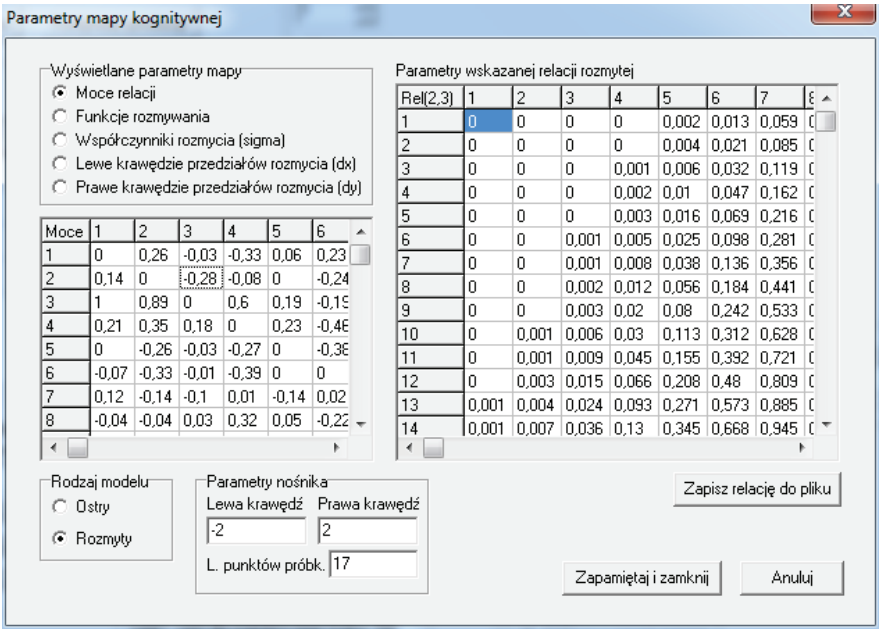
5.2. PRZYKŁADOWA REALIZACJA APLIKACYJNA RELACYJNEJ ROZMYTEJ MAPY KOGNITYWNEJ

Na potrzeby projektowania i analizy działania różnych typów modeli RRMK zbudowano komputerowy system projektowo-testujący, wykorzystujący mechanizmy arytmetyki liczb i relacji rozmytych, opisane w podrozdziałach 3.3, 3.4 i rozdziale 4, oraz mechanizm uczenia (adaptacji), opisany w podrozdziale 4.9. Aplikację stworzono w środowisku programistycznym Borland Delphi, a jej poszczególne funkcje realizowano poprzez specjalnie zaprojektowane moduły obliczeniowe. Rysunek 5.9 przedstawia ogólny widok panelu sterującego tej aplikacji.

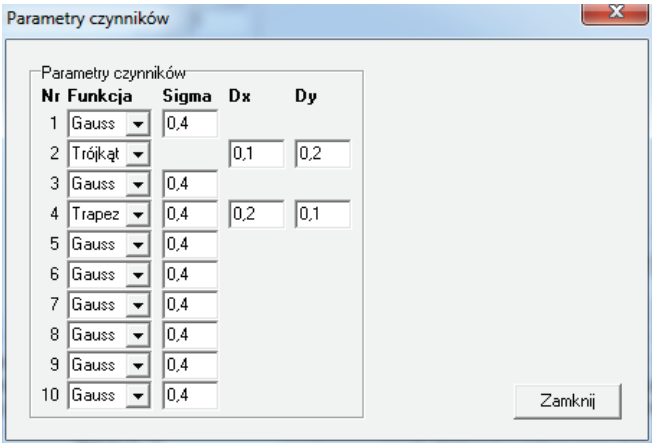
Aplikacja umożliwia projektowanie i modyfikacje relacji rozmytych i nośnika (przykładowy widok okna głównych parametrów mapy kognitywnej przedstawia rys. 5.10), modyfikację podstawowych parametrów czynników rozmytych (okno własności czynników zostało pokazane na rys. 5.11) oraz testowanie różnych typów map kognitywnych zarówno ostrych, jak i rozmytych – ze szczególnym uwzględnieniem różnych algorytmów uczenia RRMK.



Rys. 5.9. *Widok panelu sterującego aplikacji zarządzającej pracą modelu relacyjnej rozmytej mapy kognitywnej*

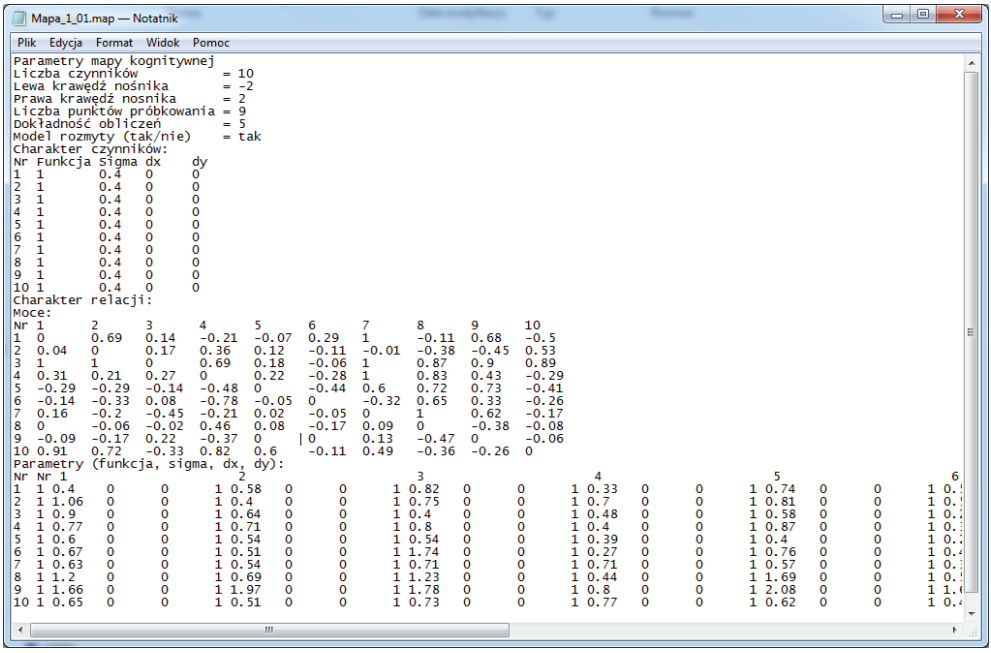


Rys. 5.10. *Widok okna głównych parametrów przykładowej mapy kognitywnej*



Rys. 5.11. *Widok okna parametrów przykładowych czynników rozmytych*

Model RRMK może być tworzony przez aplikację lub wprowadzany z zewnątrz, z pliku tekstowego (przykładowy wygląd takiego pliku pokazano na rys. 5.12). Wyniki pracy algorytmu uczącego (model wynikowy) również mogą być zapisane do podobnego pliku tekstowego.



Rys. 5.12. *Widok pliku tekstowego z przykładowym modelem RRMK*

Celem, dla którego stworzono opisywaną w niniejszym rozdziale aplikację było umożliwienie testowania opracowywanych modeli RRMK oraz metod projektowa-

nia i uczenia tych modeli. Zasadniczo nie optymalizowano pracy aplikacji pod kątem jej praktycznego wykorzystania w systemach czasu rzeczywistego. Jest to odrębny problem wymagający dalszych prac, ponieważ aparat arytmetyki rozmytej, wykorzystywany w modelu RRMK, cechuje się wysoką złożonością obliczeniową, co przekłada się na wysoką czasochłonność. Zagadnienie to nie jest rozważane w niniejszej monografii, ponieważ jej przedmiotem jest prezentacja pewnego nowego, teoretycznego, podejścia do modelowania pracy złożonych systemów. Tym niemniej problem był brany pod uwagę przy projektowaniu oprogramowania. Chcąc umożliwić korzystanie z aplikacji na standardowym komputerze typu PC, nie przystosowano jej do pracy wieloprocesorowej, jednakże zastosowano podział zadań na moduły funkcjonalne (głównym jest wykonanie obliczeń pojedynczego cyklu obiegu sygnałów), które są wykonywane techniką wielowątkową, co poprawia elastyczność pracy, skraca czas obliczeń oraz ułatwia sterowanie aplikacją.

Proces budowy modelu rozpoczyna się od zaprojektowania (w oparciu o wiedzę ekspertową) ogólnej postaci RRMK. W ramach tego etapu określa się liczbę czynników, typy i parametry funkcji przynależności rozmytych wartości czynników oraz typy funkcji przynależności relacji rozmytych łączących poszczególne czynniki. Można też wstępnie określić moce i współczynniki rozmycia poszczególnych relacji, jak również liczbę punktów próbkowania nośnika. W kolejnym kroku determinuje się sposób normalizacji ostrych (lub wyostrzonych) wartości czynników. Dokonuje się tego na podstawie posiadanych danych pomiarowych oraz wiedzy na temat przewidywanych lub możliwych do osiągnięcia wartości poszczególnych czynników. Zależnie od założonego charakteru modelu można w tym celu posłużyć się zależnością (2.3) lub (2.4) – normalizację według (2.4) wybiera się w sytuacjach, w których istotne jest odwzorowanie znaku danego czynnika. Na tym etapie wybiera się również sposób wyostrzania rozmytych wartości czynników, przy tym w aplikacji zaimplementowano jedynie metodę średniej ważonej (wg równania (3.58)) ponieważ najlepiej oddaje ona „energię" liczby rozmytej charakteryzującej rozmytą wartość czynnika. Taki wybór powoduje, że system obliczeniowy staje się wrażliwy na symetrię wartości funkcji przynależności, a to z kolei skutkuje koniecznością stosowania poszerzonego zakresu nośnika. Z doświadczeń wynika, że wystarczającą (z tego punktu widzenia) dokładność zapewnia rozszerzenie nośnika do zakresu $[-2, 2]$, tym niemniej aplikacja umożliwia dowolne ustawianie jego granicznych wartości.

Wstępnie dobrane parametry systemu zapisuje się w pliku tekstowym (takim jak na rys. 5.12) i wprowadza się je do systemu (można też wpisać je bezpośrednio w aplikacji z pominięciem pliku parametrów). Następnie wprowadza się do systemu dane uczące (czyli znormalizowane wartości czynników wzorcowych). Dane te uzyskuje się drogą badania działania rzeczywistego układu będącego przedmiotem modelowania i zapisuje się w pliku tekstowym, z którego następnie pobiera je aplikacja. Dodatkowo (z kolejnego pliku tekstowego) wprowadza się znormalizowane wartości początkowe czynników (dogodnie jest do tego pobrać wartości kilku pierwszych rekordów zbioru uczącego, ale zasadniczo mogą to być dowolne, np. losowe wartości z zakresu $[-1, 1]$).

Po wykonaniu wyżej wymienionych operacji aplikacja jest gotowa do uczenia modelu RRMK. Wybiera się metodę uczenia (dostępne metody wyszczególniono na głównym panelu sterującym) oraz parametry uczenia, takie jak: startowy krok zmian wartości współczynników mocy i rozmycia relacji oraz liczbę cykli zmian, które system wykona dla pojedynczego kroku adaptacji. Dodatkowo można określić liczbę rekordów uczących branych pod uwagę (nie muszą to być wszystkie dostępne w zbiorze uczącym) – zależnie od przyjętej metody uczenia wybrana liczba określa długość (mierzoną liczbą kroków czasu dyskretnego) przebiegów do odwzorowania lub numer kroku czasu dyskretnego, w którym model ma uzyskiwać najdokładniejsze wartości. Następnie uruchamia się proces uczenia, który dalej przebiega już automatycznie. Proces ten kończy się po spełnieniu warunku stopu, którym jest osiągnięcie zakładanej dokładności modelu lub osiągnięcie stanu, w którym dokładność w kolejnych cyklach się nie poprawia, lub wykonanie granicznej liczby cykli. Po jego zakończeniu model RRMK jest gotowy do pracy, a jego parametry mogą być zapisane do pliku tekstowego o postaci przedstawionej, tak jak na rysunku 5.12.

Oprócz danych w postaci rekordów (przebiegów) uczących aplikacja może posługiwać się danymi odniesienia, które sama wygeneruje, posługując się wbudowanym ostrym modelem RMK. W takiej sytuacji budowę modelu RRMK poprzedza się wprowadzeniem do aplikacji modelu RMK (może on być także poddany modyfikacjom), a następnie przebiegi wartości czynników RMK mogą służyć jako wzorzec do uczenia RRMK. Metoda taka może być stosowana w niektórych szczególnych sytuacjach, ale w zasadzie jej zastosowania ograniczają się jedynie do potrzeb szybkiego testowania nowych procedur uczących.

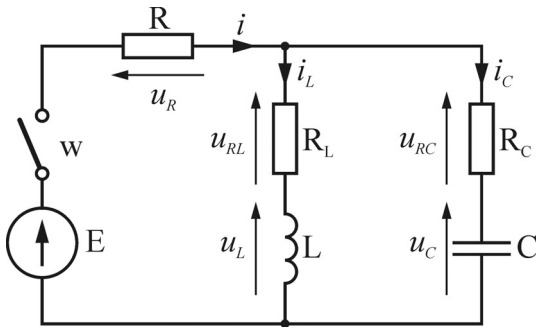
Modelowanie z użyciem RRMK może służyć zarówno do rozwiązywania typowych zadań klasyfikacji (takie zastosowanie jest dostępne także w innych, prostszych metodach – jak np. modele sztucznych sieci neuronowych), jak również bardziej złożonych problemów odwzorowywania pracy badanego systemu w określonych warunkach. Dotyczy to wszelkiego rodzaju systemów, w których występują wielokierunkowe wzajemne oddziaływania kluczowych czynników (są to np. systemy ekonomiczne, medyczne, logistyczne, transportowe, militarne, socjologiczne), w tym również, choć jest to mniej oczywiste, systemów technicznych. Przy tym metody tej można używać zarówno do predykcji, jak i identyfikacji.

Proces tworzenia i wykorzystywania modelu RRMK zilustrowano na poniższych przykładach, w których wykorzystano zbiory danych o różnym charakterze przewidywalności.

Przykład 5.2. *Model układu technicznego na przykładzie obwodu RLC*

Na wstępie należy zaznaczyć, że do celów analizy prostych obwodów RLC, które są dobrze opisane matematycznie, bardziej efektywna będzie droga analityczna. Wybór takiego przykładu został podyktowany właśnie tą cechą, która sprawia, że stosunkowo łatwo można generować zbiory danych uczących dla różnych stanów pracy systemu, dzięki czemu można dokonywać pogłębionej analizy zarówno poprawności doboru głównych parametrów, jak i skuteczności poszczególnych metod uczenia.

Badany przykładowy obwód RLC został przedstawiony na rysunku 5.13.



Rys. 5.13. *Badany obwód RLC. R = 10 Ω; R_L = 0,5 Ω; R_C = 0,5 Ω; L = 0,08 H; C = 0,03 F; E = 10 V*

Obwód przedstawiony na rysunku 5.13 może być opisany analitycznie za pomocą układu równań (5.1):

$$\begin{cases} \dfrac{di_L(t)}{dt} = \dfrac{1}{L}[E(t) - R \cdot i(t) - R_L \cdot i(t)] \\ \dfrac{du_C(t)}{dt} = \dfrac{1}{C \cdot R_C}[E(t) - R \cdot i(t) - u_C(t)] \\ i_C(t) = \dfrac{E(t) - R \cdot i_L(t) - u_C(t)}{R + R_C} \\ i(t) = i_L(t) + i_C(t) \\ u_R(t) = R \cdot i(t) \\ u_L(t) = E(t) - u_R(t) - R_L \cdot i_L(t) \end{cases} \tag{5.1}$$

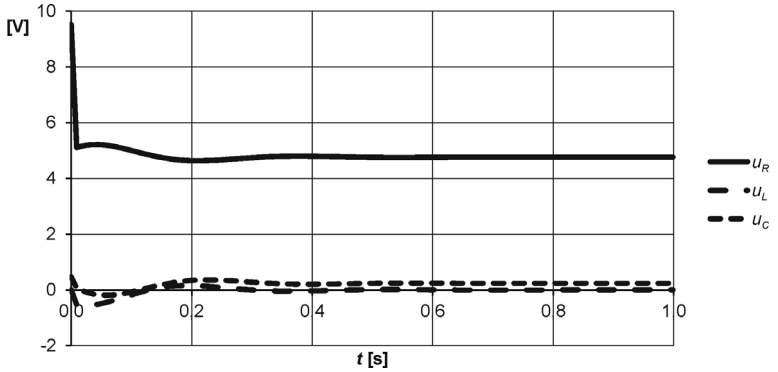
Rozwiązanie układu równań (5.1) pozwala wygenerować wzorcowe przebiegi czasowe poszczególnych wielkości obwodu w różnych stanach pracy. Przebiegi te mogą być następnie użyte jako przebiegi wzorcowe dla celów uczenia modelu RRMK.

Przed wprowadzeniem przebiegów wzorcowych do systemu obliczeniowego należy spełnić dodatkowe warunki wstępne:

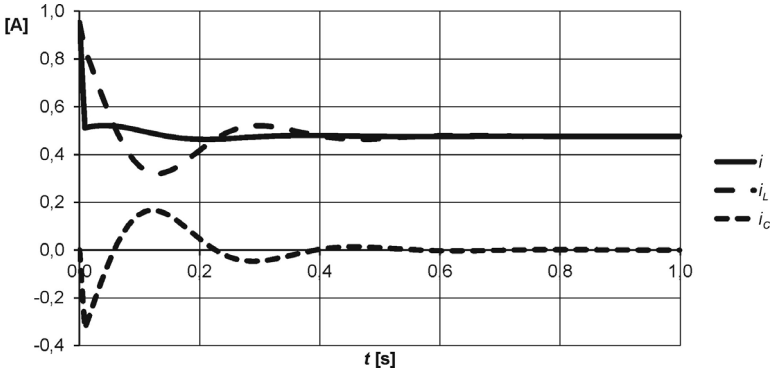
– zastosować normalizację wartości wielkości wzorcowych,
– dobrać czasowy odpowiednik długości pojedynczego kroku czasu dyskretnego, z którym będzie pracować model RRMK.

Na potrzeby niniejszego przykładu założono, że celem modelowania będzie odwzorowanie stanu przejściowego, który wystąpi, kiedy w uprzednio ustabilizowanym obwodzie nastąpi skokowy spadek napięcia zasilania E z 10 do 5 V. Przedmiotem modelowania będą przebiegi czasowe wartości wybranych wielkości

w czasie 1 s. W celu uzyskania wzorcowych przebiegów odniesienia przeprowadzono komputerową symulację systemu opisanego równaniami (5.1). Wybrane wyniki ilustrują rysunki 5.14 i 5.15.

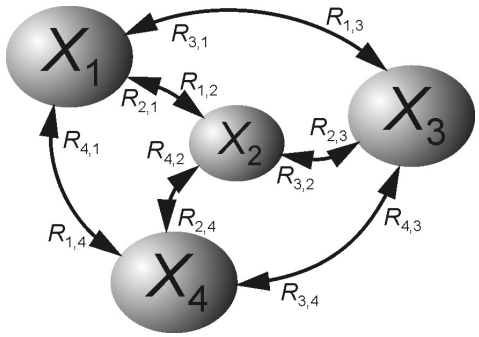


Rys. 5.14. *Przebiegi czasowe napięć w badanym obwodzie RLC po spadku napięcia zasilania E z* 10 *do* 5 *V*



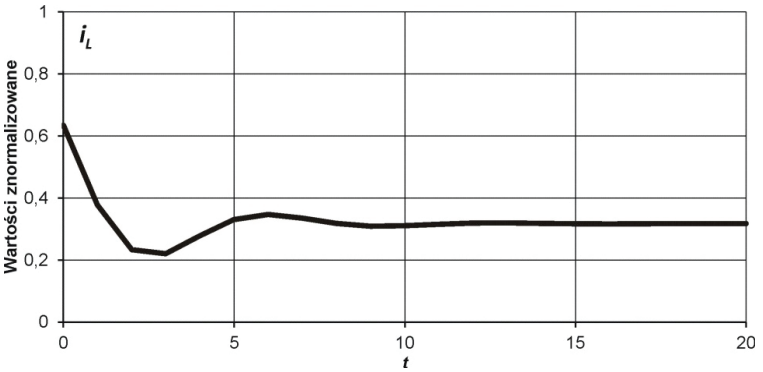
Rys. 5.15. *Przebiegi czasowe prądów w badanym obwodzie RLC po spadku napięcia zasilania E z* 10 *do* 5 *V*

Z (5.1) wynika, że do analitycznego opisu obiektu potrzeba sześciu zmiennych, z których każda odpowiada określonej wielkości fizycznej. Należy do nich doliczyć jeszcze wartość E, ponieważ badaniu podlegają skutki jej zmiany. Do celów modelowania z użyciem RRMK wybrano cztery czynniki, zakładając, że taka liczba zapewni wymaganą dokładność (w przypadku nieosiągnięcia celu liczbę tę można zmienić). Wybranymi czynnikami są: E, i_L, u_C oraz i. Rozmyte wartości tych czynników oznaczono odpowiednio jako: X_1, X_2, X_3 oraz X_4. W oparciu o wyżej wymieniony wybór zaprojektowano RRMK, której graficzną reprezentację przedstawia rysunek 5.16.



Rys. 5.16. *Graficzna reprezentacja przykładowej RRMK zbudowanej na potrzeby modelowania obwodu RLC*

Przebiegi pokazane z rysunkach 5.14 i 5.15 nie mogą jeszcze być użyte jako wzorcowe dla uczenia RRMK. Należy je odpowiednio wstępnie przygotować, czyli poddać normalizacji. Na potrzeby tej normalizacji zastosowano metodę zgodną z (2.4), przy czym graniczne wartości poszczególnych czynników dobrano, biorąc pod uwagę nie tylko zakresy danych wzorcowych, ale także wartości potencjalnie możliwe do wystąpienia w systemie. Ponadto rozważono dobór odpowiedniej długości kroku czasu dyskretnego, gdyż zbytnie jego skrócenie nadmiernie wydłuża czas adaptacji przy marginalnej poprawie dokładności modelu. Wybrano krok czasu dyskretnego będący odpowiednikiem 50 ms czasu rzeczywistego. Takie założenie powoduje, co prawda, pewne pogorszenie dokładności modelu, ale wydatnie skraca czas adaptacji parametrów. Następnie przebiegi zmodyfikowano zgodnie z wyżej wymienionymi zasadami, uzyskując w ten sposób zbiór danych uczących. Cząstkowy efekt takiej modyfikacji ilustruje rysunek 5.17, który pokazuje wzorcowy przebieg prądu i_L (podobnie przekształcono pozostałe czynniki).



Rys. 5.17. *Przebieg czasowy prądu $i_L(t)$ po transformacji do postaci znormalizowanej, użytej na potrzeby uczenia modelu RRMK; t – czas dyskretny*

Dla zachowania maksymalnej zgodności z danymi wzorcowymi założono, że w dwóch pierwszych krokach czasu dyskretnego ($t = 0$ oraz $t = 1$) wartości czyn-

ników będą rozmywane wokół wartości dwóch pierwszych rekordów danych wzorcowych, natomiast w kolejnych krokach czasu dyskretnego model będzie sam generował wartości poszczególnych czynników. Przyjęto, że wszystkie czynniki będą rozmywane zgodnie z funkcją przynależności (4.14), natomiast relacje rozmyte – zgodnie z (4.16). Wartości początkowe współczynników mocy ($\bar{r}_{i,j}$) wszystkich relacji wynosiły 0, a ich początkowe współczynniki rozmycia ($\sigma_{i,j}$) były równe 0,4. Na potrzeby uczenia, jak również późniejszej pracy modelu, przyjęto jednakowe współczynniki rozmycia wszystkich czynników ($\sigma_i = 0{,}6$). Model operował na nośniku o zakresie $[-2, 2]$ z liczbą punktów próbkowania nośnika $k = 17$. Pracę modelu determinowano według zależności (4.3). Uczenie przeprowadzono metodą kolejnych przybliżeń ze zmiennym krokiem zmian parametrów, opisaną w podrozdziale 4.9. W efekcie uzyskano relacyjną rozmytą mapę kognitywną (jak na rys. 5.16), w której główne parametry relacji rozmytych mają wartości jak w tabeli 5.2.

Tabela 5.2. *Współczynniki mocy (a) oraz rozmycia (b) modelu RRMK otrzymane w wyniku uczenia*

a)

$\bar{r}$	X_1	X_2	X_3	X_4
X_1	0,00	0,01	–0,08	0,09
X_2	0,00	0,00	–0,25	0,02
X_3	0,00	0,38	0,00	-0,07
X_4	0,00	0,08	–0,22	0,00

b)

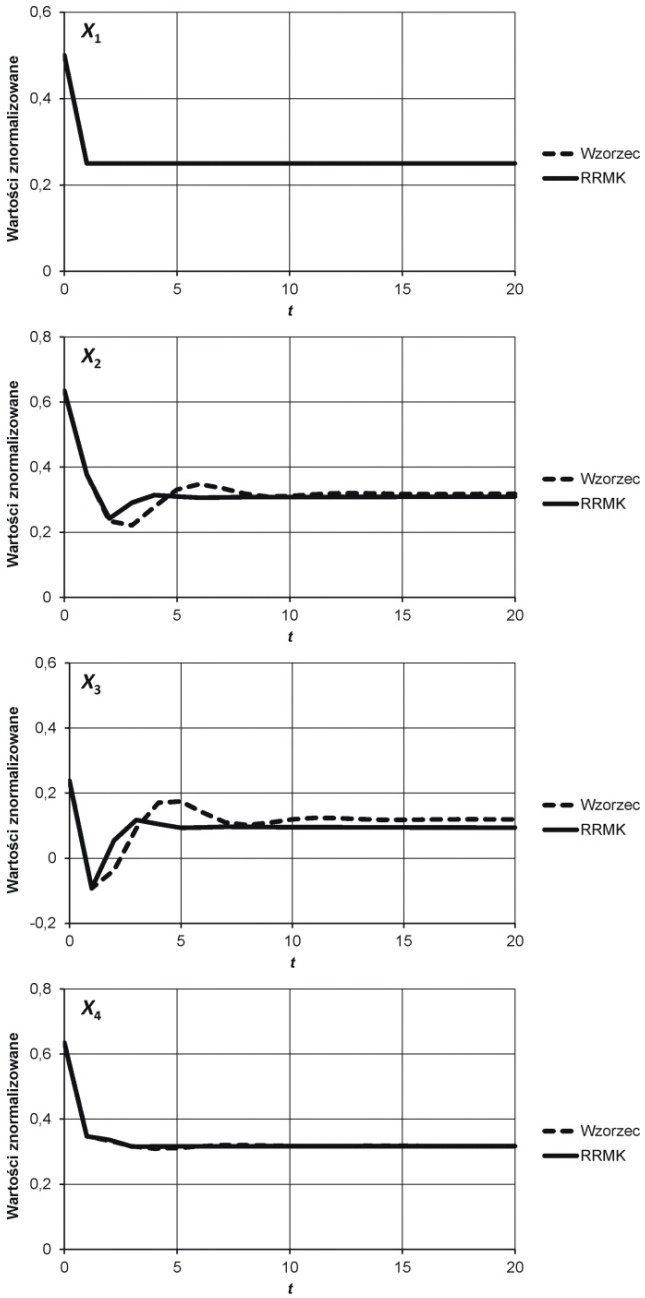
σ	X_1	X_2	X_3	X_4
X_1	0,40	0,42	0,29	0,38
X_2	1,63	0,40	0,30	1,08
X_3	1,63	0,55	0,40	0,54
X_4	1,63	0,38	0,43	0,40

Po każdej modyfikacji każdego parametru obliczano wartości współczynników bliskości poszczególnych czynników zgodnie z (4.24) i następne działania podejmowano w oparciu o analizę ich wartości.

Po zakończeniu uczenia przeprowadzono symulację pracy modelu. Uzyskane tą drogą przebiegi wyostrzonych wartości czynników, wraz z wartościami wzorcowymi, przedstawiono na rysunku 5.18.

Uwaga 5.2

Należy pamiętać, że przebiegi, które zostały przedstawione na rysunku 5.18, reprezentują znormalizowane wartości czynników. Przekształcenie ich do postaci bezwzględnych (zwymiarowanych w odpowiednich jednostkach fizycznych) jest prostą operacją arytmetyczną wykonywaną z użyciem tych samych współczynników, co zabieg normalizacji.



Rys. 5.18. *Przebiegi czasowe wyostrzonych wartości czynników (w znormalizowanej formie) uzyskane podczas modelowania z użyciem RRMK w odniesieniu do przebiegów wzorcowych. t – czas dyskretny*

Przedstawiony w niniejszym przykładzie model RRMK zbudowano na bazie jedynie czterech czynników. Dlatego też, chociaż wszystkie czynniki mają sens fi-

zyczny, charakter relacji rozmytych jest czysto abstrakcyjny. Co więcej, liczba użytych wartości lingwistycznych (17 na nośniku o zakresie $[-2, 2]$, co oznacza 5 wartości lingwistycznych na standardowym zakresie nośnika $[-1, 1]$) jest relatywnie niska. Pomimo tych ograniczeń, odpowiednia adaptacja parametrów relacji rozmytych pozwoliła osiągnąć zadowalającą reprezentację modelowanych przebiegów.

Przykład 5.3. *Model systemu transportowego na przykładzie systemu kolejowych przewozów pasażerskich*

W przykładzie 5.2 przedstawiono wybrane wyniki modelowania pracy systemu, którego charakter sprawiał, że zmiany wartości czynników były do pewnego stopnia przewidywalne [175]. W niniejszym przykładzie zostanie zaprezentowana próba wykorzystania RRMK do zamodelowania pracy systemu bardziej złożonego, którego czynniki zmieniają się w sposób mniej przewidywalny. Jest nim wybrany fragment rzeczywistego systemu transportowego – systemu kolejowych przewozów pasażerskich w Polsce. Dane odniesienia zostały w większości zaczerpnięte z oficjalnego raportu [196] obejmującego dane statystyczne przewozów kolejowych z lat 2003–2011 (oprócz jednego parametru – dynamiki PKB, który pochodzi z danych opublikowanych przez GUS). Na potrzeby budowy modelu wybrano następujące czynniki kluczowe: X_1 – liczba pasażerów przewiezionych w ciągu roku, X_2 – praca przewozowa (wyrażana w pasażerokilometrach), X_3 – średnia odległość przewozu pasażera, X_4 – średnia liczba pociągów pasażerskich uruchamianych w ciągu doby, X_5 – średnia liczba pasażerów w pociągach, X_6 – liczba wagonów pasażerskich, X_7 – zatrudnienie, X_8 – przychody z działalności operacyjnej, X_9 – zyski (liczone jako różnica pomiędzy przychodami a kosztami) z działalności operacyjnej, X_{10} – dynamika PKB. Wartości bezwzględne wyżej wymienionych parametrów w ujęciu czasowym zestawiono w tabeli 5.3.

Tabela 5.3. *Wartości bezwzględne parametrów systemu kolejowego transportu pasażerskiego w Polsce (na podst. raportu [196]) – dane uczące*

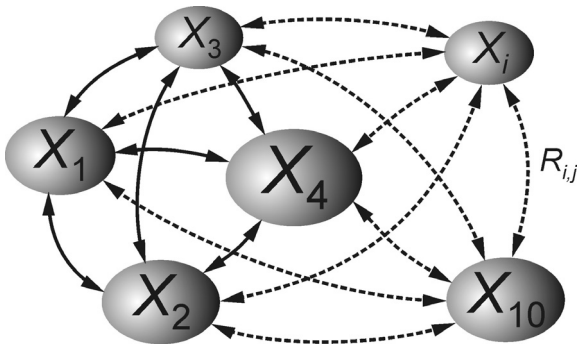
Rok / X	2003	2004	2005	2006	2007	2008	2009	2010	2011
X_1 [mln]	282,5	271,2	257,6	262,6	278,75	292,74	284,05	262,33	264,54
X_2 [mln]	19382,5	18305,3	17814,8	18298,9	19495,2	20263,1	18691,7	17917,9	18169,3
X_3 [km]	68,62	67,5	69,16	69,67	69,94	69,22	67,16	68,3	68,68
X_4 [szt.]	3778	3529	3555	3746	4471	3825	3970	4113	4062
X_5 [os/poc.]	120,78	127,16	132,4	128,92	138,87	140,96	130,84	121,98	125,81
X_6 [szt.]	9010	8829	8487	8353	8247	8060	7921	7900	8079
X_7 [os.]	23856	22881	22477	22098	22488	28541	28233	26681	25915
X_8 [mld zł]	3,03	4,1	3,5	3,97	3,32	4,54	4,21	4,43	4,77
X_9 [mld zł]	−1,18	−0,03	−0,6	−0,18	−0,76	−0,05	−0,25	−0,15	−0,02
X_{10} [%]	3,9	5,3	3,7	6,2	6,6	4,9	1,7	3,8	4,2

Wartości bezwzględne z tabeli 5.3 poddane normalizacji zgodnie z (2.4) (zachowano pewien dodatkowy margines bezpieczeństwa pomiędzy przyjętymi wartościami skrajnymi a zakresami danych statystycznych), w wyniku czego otrzymano wartości znormalizowane zestawione w tabeli 5.4.

Tabela 5.4. *Znormalizowane wzorcowe wartości czynników z tabeli 5.3*

Rok / X	2003	2004	2005	2006	2007	2008	2009	2010	2011
X_1	0,550	0,475	0,384	0,414	0,532	0,609	0,575	0,408	0,418
X_2	0,438	0,331	0,285	0,349	0,438	0,512	0,390	0,282	0,317
X_3	0,621	0,583	0,638	0,658	0,662	0,640	0,579	0,603	0,629
X_4	0,389	0,265	0,424	0,369	0,572	0,452	0,493	0,449	0,590
X_5	0,208	0,272	0,324	0,300	0,393	0,411	0,313	0,215	0,251
X_6	0,670	0,610	0,492	0,455	0,418	0,371	0,326	0,311	0,370
X_7	0,386	0,288	0,202	0,226	0,344	0,733	0,820	0,715	0,511
X_8	0,015	0,550	0,126	0,493	0,259	0,684	0,721	0,776	0,670
X_9	−0,787	−0,020	−0,430	−0,127	−0,414	−0,072	−0,103	−0,021	−0,178
X_{10}	0,593	0,687	0,563	0,750	0,717	0,675	0,474	0,531	0,645

W oparciu o dziesięć wyżej wymienionych czynników kluczowych zaprojektowano strukturę RRMK służącej do modelowania działania systemu. Jej reprezentację graficzną przedstawia rysunek 5.19.



Rys. 5.19. *Graficzna reprezentacja przykładowej RRMK zbudowanej na potrzeby modelowania systemu kolejowego transportu pasażerskiego.* $X_1 \div X_{10}$ *– rozmyte wartości kolejnych czynników (*X_i *– reprezentuje czynniki* $X_5 \div X_9$ *pominięte na rysunku dla zwiększenia jego przejrzystości);* $R_{i,j}$ *– ogólne oznaczenie relacji rozmytych pomiędzy poszczególnymi czynnikami*

Na potrzeby modelowania założono, że, podobnie jak w przykładzie 5.2, wszystkie wartości czynników będą rozmywane z użyciem funkcji przynależności typu (4.14), a wszystkie relacje rozmyte – z użyciem funkcji przynależności typu (4.16). Założono ponadto wspólny dla wszystkich czynników rozmytych współczynnik rozmycia ($\sigma_i = 0{,}4$). Przyjęto następujące parametry RRMK: zakres nośnika $[-2, 2]$, liczba punktów próbkowania nośnika $k = 17$, wartości początkowe współczynników mocy wszystkich relacji rozmytych $\bar{r}_{i,j} = 0$, wartości początkowe współczynników rozmycia wszystkich relacji rozmytych $\sigma_{i,j} = 0{,}4$. Przyjęto także, że jeden krok czasu dyskretnego odpowiada jednemu rokowi czasu rzeczywistego. Badaniu poddano model RRMK zgodny z (4.3). Zaimplementowany w systemie komputerowym algorytm uczenia metodą kolejnych przybliżeń ze zmiennym krokiem zmian parametrów (przedstawiony w podrozdziale 4.9) zmodyfikował parametry początkowe RRMK do postaci pozwalającej modelować badany system. Wynikowe współczynniki mocy relacji rozmytych zestawiono w tabeli 5.5, a wynikowe współczynniki rozmycia relacji rozmytych – w tabeli 5.6.

Uwaga 5.3

Warto podkreślić, że procedura uczenia modyfikuje parametry modelu wyłącznie w oparciu o obserwowane zachowanie rzeczywistego systemu w danej sytuacji, bez żadnej wiedzy na temat wewnętrznej struktury tego systemu.

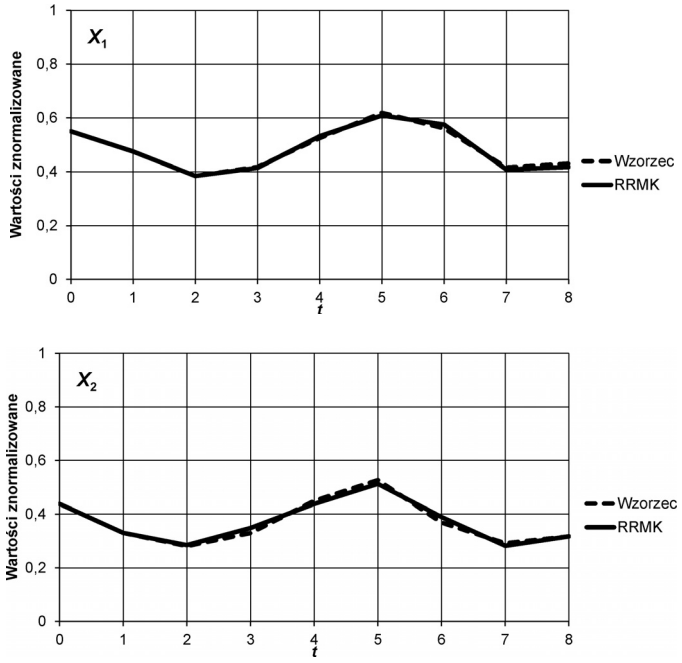
Tabela 5.5. *Wynikowe wartości współczynników mocy relacji rozmytych RRMK zbudowanej na potrzeby modelowania systemu kolejowych przewozów pasażerskich*

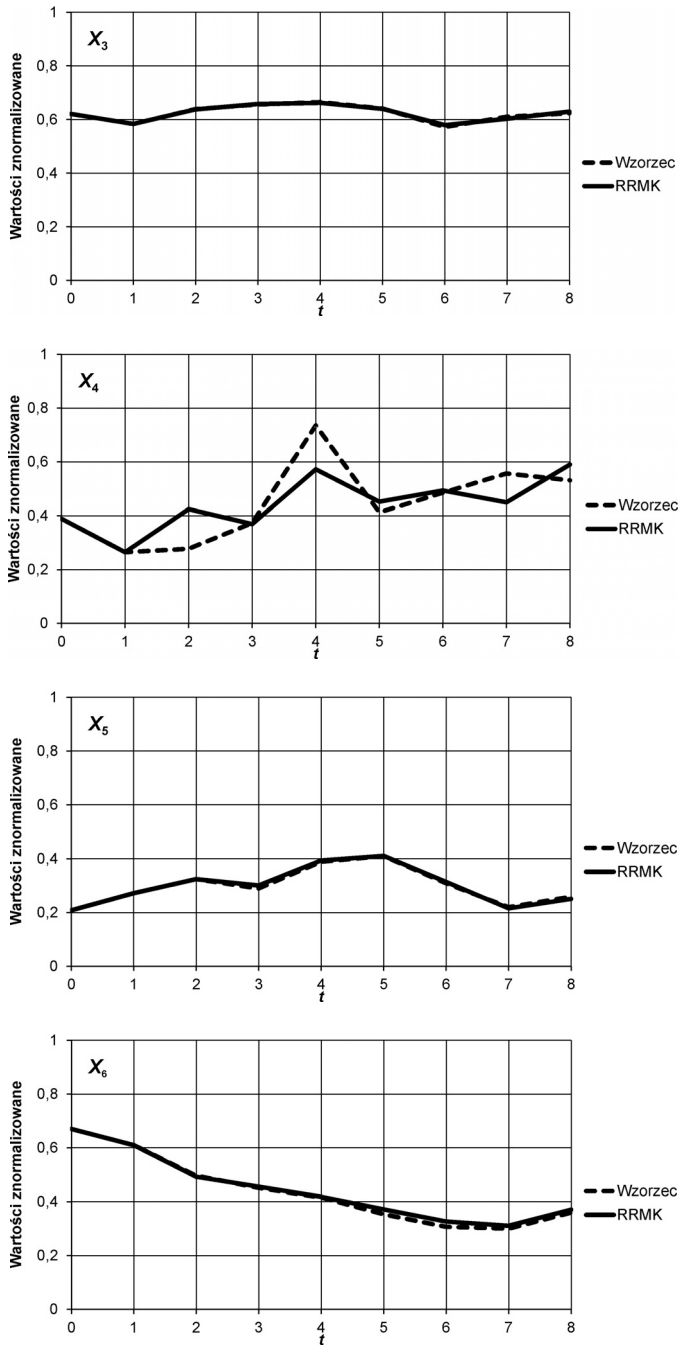
$\bar{r}$	X_1	X_2	X_3	X_4	X_5	X_6	X_7	X_8	X_9	X_{10}
X_1	0,00	0,26	– 0,03	– 0,33	0,06	0,23	0,57	– 0,39	0,16	– 0,40
X_2	0,14	0,00	0,28	– 0,08	0,00	– 0,24	0,48	0,05	– 0,24	0,05
X_3	1,00	0,89	0,00	0,60	0,19	– 0,19	0,14	0,21	0,42	0,57
X_4	0,21	0,35	0,18	0,00	0,23	– 0,46	0,99	0,68	0,43	0,16
X_5	0,00	– 0,26	– 0,03	– 0,27	0,00	– 0,38	052	0,86	0,40	– 0,46
X_6	– 0,07	– 0,33	– 0,01	– 0,39	0,00	0,00	– 0,67	– 0,25	0,15	– 0,39
X_7	0,12	– 0,14	– 0,10	0,01	– 0,14	0,02	0,00	0,61	0,61	– 0,11
X_8	– 0,04	– 0,04	0,03	0,32	0,05	– 0,22	0,36	0,00	– 0,57	– 0,08
X_9	– 0,06	– 0,03	0,02	– 0,11	0,00	– 0,05	– 0,29	– 0,42	0,00	– 0,09
X_{10}	0,69	0,55	0,01	0,58	0,43	0,27	0,51	– 0,30	– 0,49	0,00

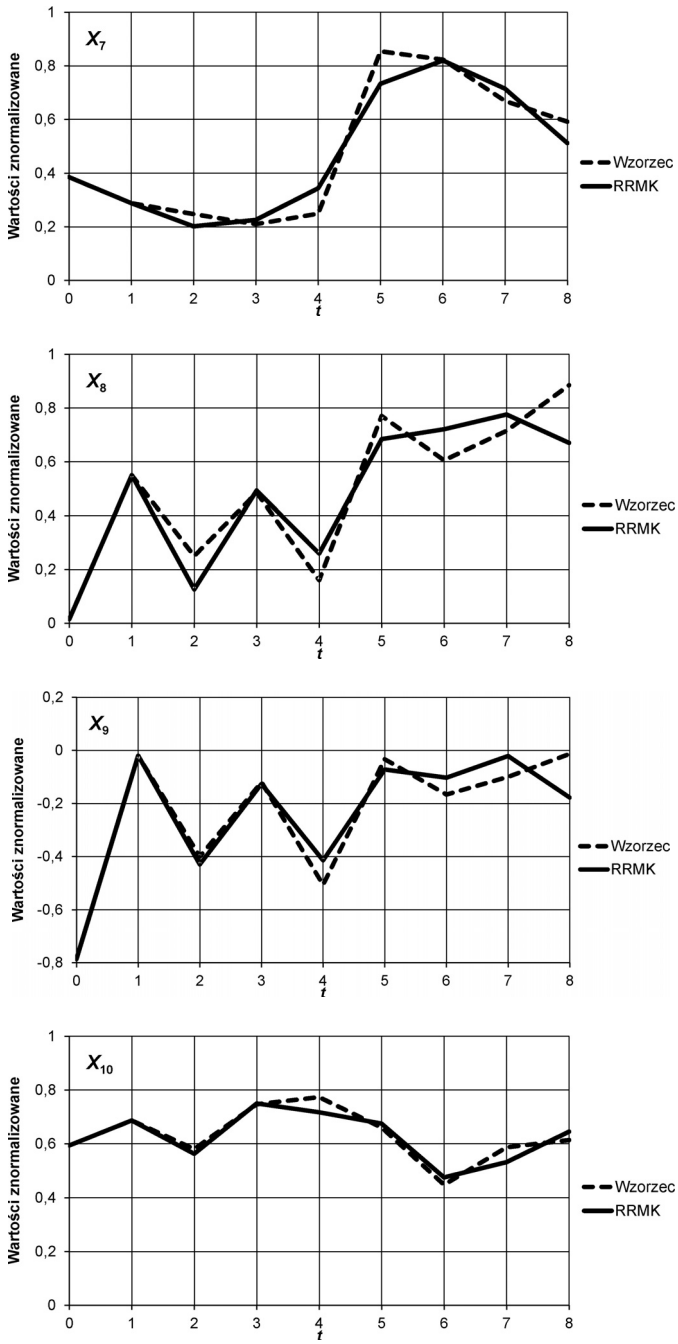
Tabela 5.6. *Wynikowe wartości współczynników rozmycia relacji rozmytych RRMK zbudowanej na potrzeby modelowania systemu kolejowych przewozów pasażerskich*

σ	X_1	X_2	X_3	X_4	X_5	X_6	X_7	X_8	X_9	X_{10}
X_1	0,40	0,43	0,59	0,29	0,91	0,54	0,36	0,56	0,42	0,44
X_2	0,60	0,40	0,63	0,75	0,57	0,52	0,30	0,54	0,32	0,64
X_3	0,63	0,58	0,40	0,44	0,56	0,43	0,66	0,61	0,45	0,45
X_4	0,63	0,59	0,53	0,40	0,66	0,30	0,37	0,59	0,38	0,68
X_5	0,63	0,43	0,52	0,34	0,40	0,31	0,22	0,38	0,40	0,41
X_6	0,63	0,38	1,19	0,30	0,71	0,40	0,61	0,55	0,46	0,18
X_7	0,62	0,55	0,80	0,75	0,49	0,47	0,40	0,58	0,53	0,44
X_8	1,62	0,65	0,89	0,61	0,40	0,44	0,49	0,40	0,38	0,56
X_9	0,55	1,68	1,79	1,30	1,83	0,42	1,11	0,36	0,40	0,58
X_{10}	0,65	0,47	0,49	0,43	0,64	0,47	0,69	0,34	0,28	0,40

Po zakończeniu uczenia sprawdzono działanie modelu (używając do tego celu zależności (4.3)) i porównano je z danymi wzorcowymi. Wyniki przedstawiono na rysunku 5.20.

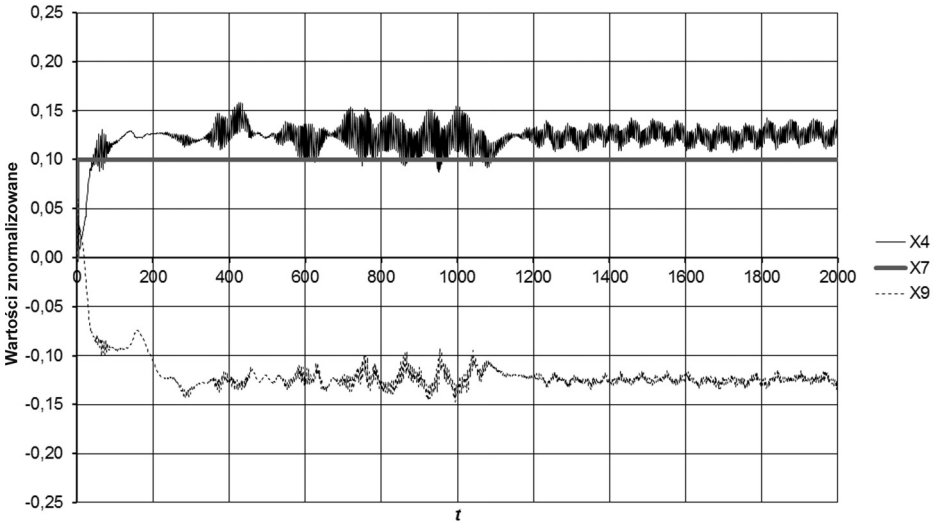



X3
X4
X5
X6
Wartości znormalizowane
t
Wzorzec
RRMK



Rys. 5.20. *Wyniki pracy RRMK modelującej działanie systemu kolejowych przewozów pasażerskich zestawione z przebiegami wzorcowymi. Wzorzec – znormalizowane wzorcowe (rzeczywiste) wartości czynników, RRMK – znormalizowane wyostrzone wartości czynników modelu, t – czas dyskretny*

Głównym zadaniem modeli budowanych na bazie RRMK jest badanie wpływu, jaki w złożonym systemie wywoła zmiana wartości któregoś z czynników na pozostałe. Jest to zagadnienie trudne do analizy wyłącznie na podstawie obserwacji zmienności przebiegów czasowych, a to z uwagi na wielorakie powiązania wzajemne pomiędzy czynnikami. Wartości wszystkich czynników zmieniają się jednocześnie, więc wyodrębnienie wpływu tylko jednego z nich rzadko jest możliwe. Mapa kognitywna (w tym również RRMK) jest narzędziem, które umożliwia dokonywanie tego typu analiz. W następnym teście opracowany w tym przykładzie model RRMK został zaprogramowany do symulowania sytuacji, w której wartości początkowe wszystkich czynników (po wyostrzeniu) są zerowe i następuje zmiana wartości jednego z nich (X_7) z zera do $0{,}1$, po czym wartość jest utrzymywana. Częściowe wyniki pracy modelu pokazano na rysunku 5.21.



Rys. 5.21. *Wybrane wyniki badania wpływu wzrostu wartości X_7 na pozostałe czynniki*

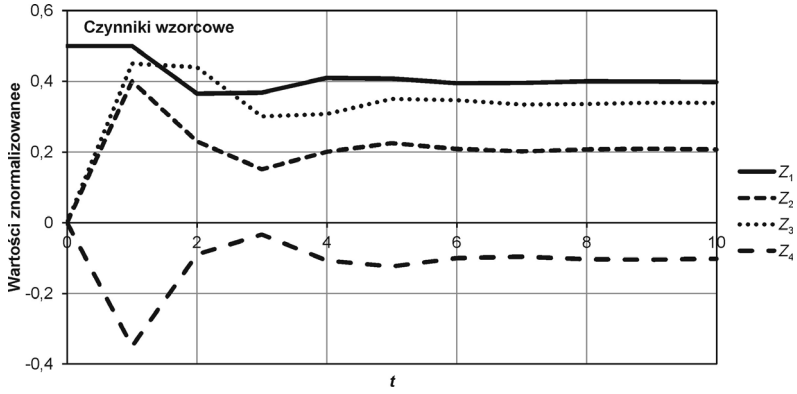
Jak pokazano na rysunku 5.21 wzrost X_7 powoduje wzrost X_4 i spadek X_9. Podobnego rodzaju wyniki można uzyskać dla innych czynników.

Przykład 5.4. *Porównanie dokładności modelu w zależności od przyjętego celu modelowania*

Dokładność modelu jest jednym z głównych wyznaczników oceny jego przydatności. Problem polega na sposobie pomiaru tej dokładności, ponieważ jest on ściśle uzależniony od głównego celu modelowania. Każdy model (w tym również RRMK) powinien zapewniać możliwie najlepsze odwzorowanie pracy rzeczywistego obiektu, jednakże samo pojęcie odwzorowania może być różnie rozumiane w różnych sytuacjach. Jak wynika z rozważań przedstawionych w podpunkcie 4.9, można wyróżnić dwa podstawowe typy celów modelowania. Po pierwsze, celem modelowania może być znalezienie wartości wybranych czynników po zakończeniu pewnego procesu

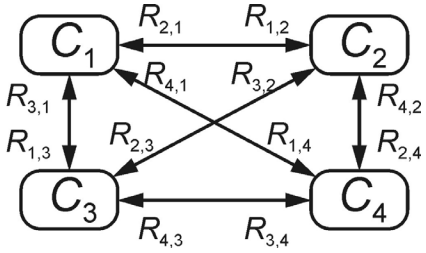
stabilizacji wewnętrznej modelu (model jest dynamiczny, co powoduje, że po pobudzeniu któregoś czynnika dąży oscylacyjnie do nowego stanu ustalonego), czyli w określonym punkcie czasu dyskretnego – uczenie modelu może być wtedy prowadzone z zastosowaniem kryterium (4.23) lub (4.25). Po drugie, celem modelowania może być maksymalnie dokładne odwzorowanie całego przebiegu czasowego wartości jednego bądź większej liczby czynników – do uczenia modelu można tu stosować kryterium określone wyrażeniem (4.24) lub (4.26). Wynika stąd, że oceny dokładności modelu nie można dokonywać w oderwaniu od przyjętego celu modelowania. Zostanie to wykazane na poniższym przykładzie.

Przedmiotem badania będzie hipotetyczny obiekt, w którym wyłoniono 4 czynniki o kluczowym charakterze. W stanie początkowym wartości wszystkich czynników były zerowe, następnie czynnik nr 1 został pobudzony zewnętrznym sygnałem o wartości 0,5, skutkiem czego było przejście obiektu do nowego stanu ustalonego. Proces ten został zilustrowany na rysunku 5.22.



Rys. 5.22. *Przebiegi czasowe wartości czynników obiektu (Z_1, ...,Z_4) po pobudzeniu czynnika nr* 1 *sygnałem o wartości* 0,5

Na potrzeby modelowania wyżej wymienionego obiektu zaprojektowano RRMK o czterech czynnikach, której reprezentację graficzną pokazuje rysunek 5.23.



Rys. 5.23. *Reprezentacja graficzna hipotetycznej RRMK o czterech czynnikach, będącej przedmiotem analizy dokładności w zależności od przyjętego celu modelowania. $C_1 \div C_4$ – czynniki RRMK (o wartościach rozmytych odpowiednio $X_1 \div X_4$); $\{R_{i,j} : i,j = 1,\dots,4; i \neq j\}$ – rozmyte relacje pomiędzy czynnikami RRMK*

Model przedstawiony na rysunku 5.23 poddano uczeniu metodą kolejnych przybliżeń ze zmiennym krokiem zmian parametrów (przedstawioną w podrozdziale 4.9) – malejącym od 0,05 do 0,01, biorąc pod uwagę różne cele modelowania: uzyskanie najdokładniejszego odwzorowania wartości czynników w pojedynczym kroku czasu dyskretnego oraz uzyskanie najlepszego odwzorowania przebiegów czasowych wszystkich czynników. Przyjęto, że zarówno wartości czynników, jak i relacje będą rozmywane w oparciu o funkcje przynależności typu G. Na potrzeby uczenia przyjęto następujące, wspólne początkowe wartości parametrów relacji rozmytych: $\bar{r}_{i,j} = 0$, $\sigma_{i,j} = 0{,}4$. Przyjęto także, że wartości wszystkich czynników będą rozmywane ze współczynnikiem rozmycia $\sigma_i = 0{,}5$ oraz, że nośnik modelu będzie miał zakres $[-1, 1]$ i $k = 17$.

1. Cel modelowania: uzyskanie założonych wartości wszystkich czynników w wybranym pojedynczym kroku czasu dyskretnego.

Przedmiotem zainteresowania były wartości wszystkich czynników w 10. kroku czasu dyskretnego. Uczenie przeprowadzono z zastosowaniem kryterium, którym był współczynnik bliskości zgodny z (4.25) liczony dla $t = 10$. Otrzymano model o parametrach jak w tabeli 5.7.

Tabela 5.7. *Współczynniki mocy (a) oraz rozmycia (b) modelu RRMK otrzymane w wyniku uczenia dla celu polegającego na odwzorowaniu wartości czynników w pojedynczym kroku czasu dyskretnego*

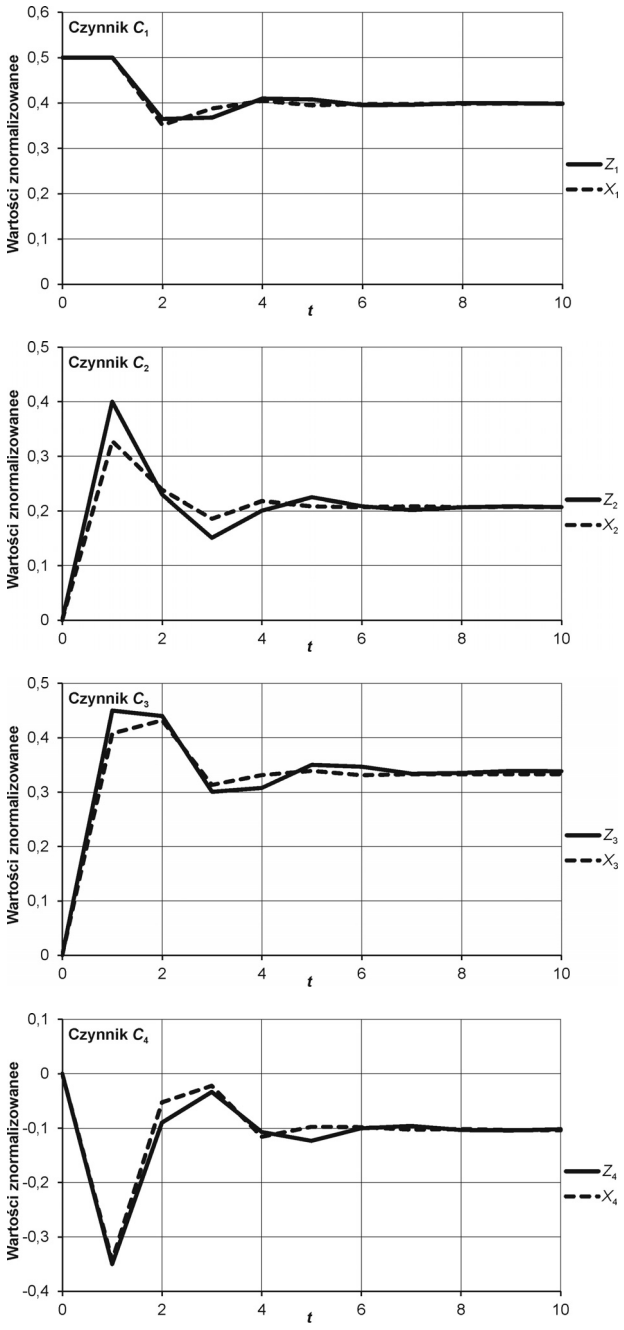
a)

$\bar{r}$	X_1	X_2	X_3	X_4
X_1	0,00	0,70	1,00	−0,71
X_2	−0,24	0,00	0,15	0,54
X_3	−0,17	0,00	0,00	0,32
X_4	0,12	0,25	0,07	0,00

b)

σ	X_1	X_2	X_3	X_4
X_1	0,40	0,38	0,50	0,42
X_2	1,32	0,40	0,36	0,51
X_3	1,27	0,49	0,40	0,46
X_4	1,33	0,44	0,89	0,40

Następnie sprawdzono działanie modelu w warunkach odpowiadających działaniu obiektu wzorcowego. Uzyskane wyniki (porównane z wzorcowymi) zestawiono na rysunku 5.24.



Rys. 5.24. *Przebiegi czasowe wyostrzonych wartości czynników modelu (X_1, ..., X_4) w porównaniu z wartościami czynników obiektu (Z_1, ..., Z_4) po pobudzeniu czynnika nr 1 sygnałem o wartości* 0,5 *– po zakończeniu uczenia, którego celem było dopasowanie modelu do dokładnego odwzorowywania wartości czynników w 10. kroku czasu dyskretnego (liczonym od momentu pobudzenia systemu)*

W wyniku uczenia uzyskano model, dla którego wartość funkcji bliskości (4.25) dla $t = 10$ wyniosła $J_{10} = 0{,}000009$.

2. Cel modelowania: uzyskanie zgodnych przebiegów wartości wszystkich czynników w badanym zakresie czasu dyskretnego.

Analizie poddano przebiegi w zakresie od $t = 0$ do $t = 10$. Uczenie przeprowadzono z zastosowaniem kryterium, którym był współczynnik bliskości zgodny z (4.26) liczony dla $T = 10$. Otrzymano model o parametrach jak w tabeli 5.8.

Tabela 5.8. *Współczynniki mocy (a) oraz rozmycia (b) modelu RRMK otrzymane w wyniku uczenia dla celu polegającego na odwzorowaniu przebiegów czasowych wartości czynników*

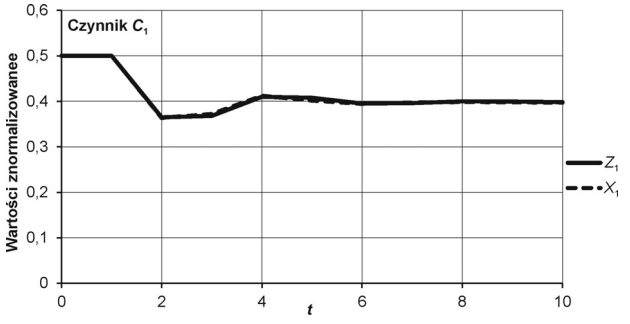
a)

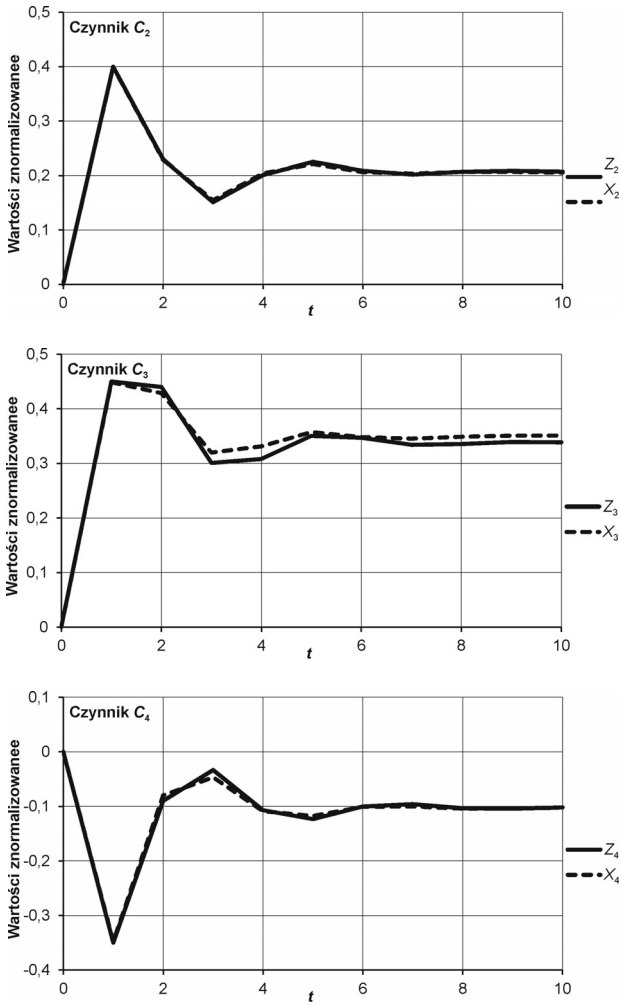
$\bar{r}$	X_1	X_2	X_3	X_4
X_1	0,00	0,95	1,00	−0,78
X_2	−0,28	0,00	0,12	0,37
X_3	−0,18	−0,27	0,00	0,30
X_4	−0,14	0,16	0,18	0,00

b)

σ	X_1	X_2	X_3	X_4
X_1	0,40	0,46	0,25	0,45
X_2	0,87	0,40	0,42	0,45
X_3	0,69	0,58	0,40	0,48
X_4	0,72	0,48	0,44	0,40

Następnie sprawdzono działanie modelu w warunkach odpowiadających działaniu obiektu wzorcowego. Uzyskane wyniki (porównane z wzorcowymi) zestawiono na rysunku 5.25.





Rys. 5.25. *Przebiegi czasowe wyostrzonych wartości czynników modelu (X_1, ..., X_4) w porównaniu z wartościami czynników obiektu (Z_1, ..., Z_4) po pobudzeniu czynnika nr 1 sygnałem o wartości* 0,5 *– po zakończeniu uczenia, którego celem było dopasowanie modelu do dokładnego odwzorowywania przebiegów czasowych wartości czynników podczas 10 kroków czasu dyskretnego (licząc od momentu pobudzenia systemu)*

W wyniku uczenia uzyskano model, dla którego wartość funkcji bliskości (4.26) dla $T = 10$ wyniosła $J = 0{,}00019$.

3. Porównanie wyników.

Model *a* charakteryzuje się wysoką dokładnością w wybranym punkcie czasu dyskretnego (sumaryczny błąd równy 0,000009), przy relatywnie niższej dokładności odwzorowywania pełnych przebiegów czasowych (sumaryczny błąd równy 0,00121). Model *b* natomiast wykazuje przeszło czterokrotnie niższą dokładność

w tym samym wybranym kroku czasu (sumaryczny błąd równy 0,000038) przy jednocześnie znacznie wyższej (przeszło sześciokrotnie) dokładności odwzorowywania przebiegów (sumaryczny błąd równy 0,00019). Dane te nie pozwalają na jednoznaczne wskazanie modelu, który byłby lepszy. Wybór taki zależy głównie od spełniania przez dany model wymagań stawianych w danym momencie. Wynika stąd, że, niezależnie od rodzaju inteligentnego podejścia zastosowanego do modelowania złożonego nieprecyzyjnego systemu, najlepszym rozwiązaniem jest tworzenie banku modeli i bieżące posługiwanie się tym spośród nich, który będzie najlepiej dostosowany do aktualnego celu modelowania.

6 PODSUMOWANIE

Modelowanie realnie istniejącego systemu zawsze jest trudnym przedsięwzięciem, a staje się jeszcze trudniejsze przy braku wystarczającej informacji o jego strukturze. W niniejszej pracy użyto określenia „system złożony", przy tym określenie to niekoniecznie oznacza system zbudowany z wielu elementów. Odnosi się ono raczej do systemów, których struktura jest trudna lub niemożliwa do opisu z wykorzystaniem klasycznych metod modelowania matematycznego. Nawet system zawierający niewiele elementów będzie w tym sensie złożony, jeżeli powiązania pomiędzy jego elementami nie będą wystarczająco jasno zdefiniowane. Dodatkową, bardzo ważną, przesłanką tak rozumianej złożoności jest często występujące zjawisko subiektywnego wartościowania poszczególnych elementów oraz ich wzajemnych oddziaływań, co wynika z braku odpowiednio skalowanych (w sensie fizycznym) wzorców. Istnieje szeroka gama systemów charakteryzujących się zarówno niepewnością (dotyczącą nieznajomości wewnętrznych struktur przyczynowo-skutkowych), jak i nieprecyzyjnością (przejawiającą się w braku dokładnych obiektywnych miar do określania wartości), których modelowanie jest trudne z wyżej wymienionych przyczyn. Przede wszystkim są to systemy społeczne, polityczne i ekonomiczne, ale można wśród nich znaleźć również przykłady systemów medycznych, logistycznych, geograficznych, a nawet technicznych. Prezentowane w pracy podejście dotyczy tworzenia modeli takich właśnie systemów.

Od lat trwają badania nad wykorzystaniem tzw. sztucznej inteligencji do modelowania złożonych systemów, jednakże stosowane dotychczas podejścia nie zawsze są wystarczające. Najczęściej można spotkać się ze stosowaniem sztucznych sieci neuronowych różnych typów, które dobrze sprawdzają się w statycznych zadaniach klasyfikacji, ale w zasadzie nie nadają się do modelowania dynamicznych stanów przejściowych w systemach o wielokierunkowych powiązaniach. Dla takich systemów dotychczas brakowało alternatywnych, a przy tym relatywnie prostych, propozycji metod modelowania. Taką metodą jest zaproponowane w rozdziale 2 podejście oparte na wykorzystaniu modelu Relacyjnej Mapy Kognitywnej (RMK). Zastosowane w nim unikatowe rozwiązania konstruowania i uczenia (adaptacji) modelu pozwalają na modelowanie zarówno statycznych, jak i dynamicznych stanów badanych systemów. Dzięki digraficznej budowie model RMK, a właściwie sposób jego działania, można łatwo dostosować do struktury modelowanego systemu, co czyni takie podejście bardzo elastycznym. Modele RMK sprawdzają się dobrze w modelowaniu systemów z niepewnością, nie oddają natomiast nieprecyzyjności.

Nieprecyzyjność informacji, przejawiającą się subiektywnym wartościowaniem danych, uwzględnia się, wprowadzając do modelu techniki rozmywania i oparte o te techniki działania na zbiorach rozmytych. Modele wykorzystujące takie działania są badane od lat osiemdziesiątych XX wieku. Noszą one nazwę rozmytych

map kognitywnych (ang. *Fuzzy Cognitive Maps*). Ich ogólną charakterystykę, sposoby tworzenia i techniki uczenia, wraz z podstawowymi operacjami na zbiorach rozmytych, przedstawiono w podrozdziałach 3.1 i 3.2. Z ich wykorzystaniem wiążą się jednak liczne, dotąd nierozwiązane, problemy. Problemy te mają dwojaki charakter. Po pierwsze, jak we wszystkich metodach tzw. inteligentnych, bazuje się w nich na wiedzy ekspertów, którą wykorzystuje się do: wyłonienia tzw. kluczowych czynników, decydujących o zachowaniu się modelowanego systemu, określenia charakteru relacji pomiędzy czynnikami, zdefiniowania sposobów rozmywania wartości każdego czynnika (czyli określenia, dla każdego czynnika oddzielnie, liczby i charakteru funkcji przynależności kolejnych wartości rozmytych) oraz zdefiniowania sposobów rozmywania powiązań przyczynowo-skutkowych pomiędzy czynnikami (podobnie, jak dla wartości poszczególnych czynników). Po drugie, większość proponowanych metod uczenia (adaptacji) modeli FCM polega na wstępnym przekształceniu takiego modelu do postaci, w której zarówno czynniki, jak i powiązania pomiędzy nimi są wyrażane liczbami rzeczywistymi, po czym dopiero następuje właściwy proces adaptacji. Po jego zakończeniu, model powinien być ponownie przekształcony do postaci z wielkościami rozmytymi, ale nie zawsze się to robi. W tej sytuacji po pierwsze, traci się walor rozmywania jako narzędzia do kompensacji nieprecyzyjności, a po drugie, pojawiają się wątpliwości co do zasadności stosowania szeregu populacyjnych metod uczenia nadzorowanego, prezentowanych w literaturze przedmiotu – skoro dla modeli ostrych można zastosować podejście w rodzaju prezentowanego w podrozdziale 2.3.

W pracy zaproponowano nowe, niestosowane dotąd, podejście do uwzględniania zarówno niepewności, jak i nieprecyzyjności danych modelowanego systemu. Opiera się ono na wykorzystaniu modelu, który nazwano Relacyjną Rozmytą Mapą Kognitywną (RRMK). Jest to strukturalne rozwinięcie modeli zarówno RMK, jak i FCM. Czynniki RRMK reprezentują rozmyte wielkości, a formalnie są prezentowane w postaci liczb rozmytych. Powiązania pomiędzy czynnikami mają również charakter rozmyty, przy czym przyjmują one formę relacji rozmytych. W takim środowisku osadzono algorytmy pracy modelu, w których wykorzystano działania arytmetyki liczb rozmytych, co powoduje, że model może pozostać rozmyty na wszystkich etapach działania, dzięki czemu nie traci się zalet wynikających ze stosowania technik rozmytych. Dodatkowo, podejście arytmetyczne ułatwia automatyzację procesów tworzenia, modyfikacji, uczenia i eksploatacji modelu, co ogranicza możliwość popełniania błędów. Tak pojęta metoda modelowania ma charakter bardziej uniwersalny od opisanej wcześniej metody z użyciem RMK, zwłaszcza przy modelowaniu systemów o znacznej nieprecyzyjności (czyli takich, w których subiektywne wartościowanie czynników i ich powiązań przyczynowych odgrywa znaczącą rolę). Szczególnie dotyczy to systemów społecznych, politycznych, ekonomicznych, logistycznych i im podobnych. Przy ich modelowaniu (właśnie z uwagi na pewną automatyzację w uwzględnianiu nieprecyzyjności) metoda z użyciem RRMK przyniesie bardziej miarodajne rezultaty. Stworzenie efektywnej metody budowy modeli RRMK wymagało rozwiązania szeregu problemów oraz wprowadzenia nowych rozwiązań i koncepcji, wśród których najważniejszymi były:

1. Opracowanie nowego sposobu opisu rozmytej wartości czynnika.

W rozmytych mapach kognitywnych stosowano dotychczas dwa sposoby przedstawiania wartości czynników. Pierwszy polegał na określeniu (w oparciu o wiedzę ekspertową), zestawów wartości rozmytych wraz z reprezentującymi je funkcjami przynależności – dla każdego czynnika osobno, po czym wyrażano wartość czynnika poprzez wskazanie wybranej wartości rozmytej. Drugi sposób stanowiący rozwinięcie pierwszego polegał na zastąpieniu określonej wcześniej wartości rozmytej wartością liczbową (w postaci liczby rzeczywistej – najczęściej z zakresu $[0, 1]$). We wcześniejszych rozwiązaniach stosowano też podejście, w którym czynnik przybierał wartości ze zbioru $\{0, 1\}$. Każde z wyżej wymienionych rozwiązań wiązało się z niedogodnościami, ponieważ posługiwanie się wartościami rozmytymi w ich klasycznej postaci utrudnia proces tworzenia, uczenia i modyfikacji modelu, natomiast wprowadzenie liczb rzeczywistych niweluje zalety rozmywania.

Zaproponowano sposób reprezentacji rozmytych wartości czynników z zastosowaniem liczb rozmytych, które są zbiorami rozmytymi o szczególnych cechach. Opracowano także nową metodę reprezentacji wartości rozmytej przy użyciu tylko jednej liczby rozmytej (zamiast operowania stopniami przynależności do wielu możliwych wartości rozmytych). Dodatkowo, wprowadzono zasadę, że centra (jądra) liczb rozmytych reprezentujących rozmyte wartości czynników mogą przyjmować wartości z zakresu $[-1, 1]$ (nawiązuje to do nowego sposobu normalizacji wartości wejściowych, który opisano dalej). Dzięki takiemu, niestosowanemu wcześniej, podejściu można wprowadzić do modelu zautomatyzowane procedury obliczeniowe oparte na arytmetyce liczb rozmytych, a to z kolei umożliwia zachowanie rozmytego charakteru danych na każdym etapie uczenia i pracy modelu. Wymagało to opracowania sposobów projektowania i przetwarzania takich liczb w modelu RRMK. Między innymi zaproponowano nową klasę funkcji przynależności – typ G. Stworzono też algorytmy implementacji takich liczb i operacji arytmetycznych w komputerowym systemie obliczeniowym.

2. Określenie sposobu odwzorowywania zależności przyczynowo-skutkowych pomiędzy czynnikami.

Zależności przyczynowo-skutkowe pomiędzy czynnikami rozmytych map kognitywnych określano wcześniej jako wagi (w postaci liczb rzeczywistych) lub jako wielkości rozmyte definiowane przy użyciu zestawu wartości rozmytych. Pierwsze rozwiązanie nie pozwala w pełni wykorzystać walorów rozmywania, natomiast drugie występowało w formie zbiorów rozmytych, co utrudnia zarówno uczenie, jak i późniejszą pracę modelu.

Zaproponowano wykorzystanie relacji rozmytej jako łącznika, który odwzorowuje oddziaływanie jednego czynnika na drugi. Realizacja tej propozycji wymagała zastosowania nowego sposobu matematycznego opisu powiązań pomiędzy czynnikami. Zaproponowano użycie w tym celu operacji kompozycji maksyminowej pomiędzy liczbą rozmytą opisującą wartość czynnika a wybraną relacją rozmytą.

3. Określenie charakteru relacji rozmytych.
Wprowadzenie relacji rozmytych do modelu RRMK wymaga rozwiązania podstawowego problemu – kształtu takich relacji. Problem ten nie był dotychczas rozwiązany ani nawet poruszany w literaturze, a ma on kluczowe znaczenie.
Zaproponowano (w podrozdziale 4.5) nowy sposób projektowania funkcji przynależności relacji rozmytych pomiędzy czynnikami. Reprezentacja graficzna takiej relacji ma kształt trójwymiarowego tunelu o wysokości równej 1, którego grzbiet determinuje funkcja jednej zmiennej (nośnika). Połączenie kompozycją rozmytą wartości czynnika (w postaci liczby rozmytej) oraz tak zaprojektowanej relacji rozmytej zapewnia dobre odwzorowanie zależności pomiędzy czynnikami, jest też dobrym ekwiwalentem adekwatnych działań na liczbach rzeczywistych. Jednoczesne zastosowanie liczb rozmytych do reprezentacji wartości czynników oraz relacji rozmytych do reprezentacji powiązań przyczynowo-skutkowych umożliwia daleko idącą automatyzację działań mapy (zwłaszcza w zakresie uczenia) bez utraty zalet rozmywania.

4. Opracowanie nowego sposobu normalizacji danych.
Metody normalizacji, stosowane powszechnie w systemach inteligentnych, powodują sprowadzenie wartości wejściowych wielkości do wspólnego zakresu $[0, 1]$, niezależnie od znaków tych wartości wejściowych, co może powodować przekłamania w modelowaniu systemów o dynamicznej strukturze wewnętrznej, w których znaki sygnałów mają znaczenie.
Zaproponowano zmodyfikowaną metodę normalizacji, w której zakresem docelowym jest $[-1, 1]$, a wartości po normalizacji zachowują swoje znaki i odpowiednie proporcje. Takie podejście pozwala lepiej odwzorowywać fizyczne własności modelowanych systemów.

5. Nowe zdefiniowanie pojęcia nośnika liczby rozmytej.
W ujęciu klasycznym nośnikiem zbioru rozmytego są te punkty przestrzeni rozważań (universum), dla których stopnie przynależności zbioru są większe od zera.
Na potrzeby modeli RRMK wprowadzono (w podrozdziale 4.2) zmodyfikowane pojęcie nośnika liczby rozmytej. Modyfikacja polega na określeniu nośnika jako przedziału przestrzeni rozważań, będącego podstawą budowy modelu RRMK, w którym wyodrębniono k równomiernie rozłożonych punktów. Ponieważ wartość k jest jednakowa dla wszystkich rozmytych elementów modelu, a jednocześnie może być ona zmieniana w procesie uczenia modelu, założono, że nośnik stanowią wszystkie punkty podziału niezależnie od odpowiadających im stopni przynależności. Takie rozwiązanie umożliwia efektywne operowanie działaniami arytmetyki liczb rozmytych.

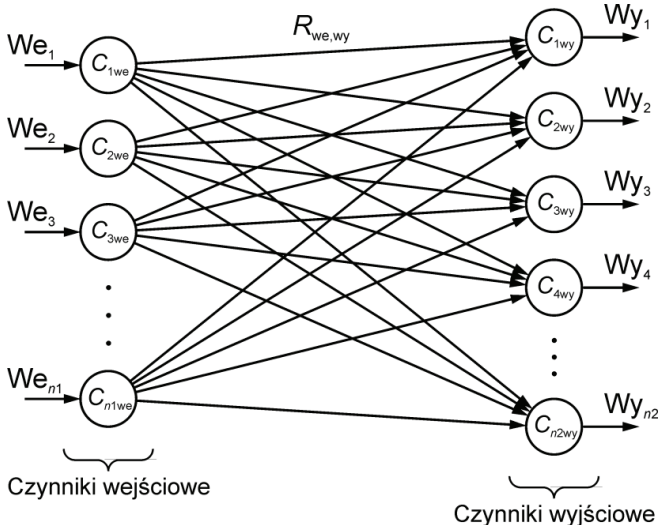
6. Opracowanie metody uczenia modelu adekwatnej do jego charakteru.
W klasycznych modelach FCM stosuje się różne metody uczenia modelu, jednakże operują one liczbami rzeczywistymi reprezentującymi wartości czynników i relacji,

co sprawia, że są nieprzydatne dla modelu RRMK, w którym zakłada się utrzymywanie postaci rozmytych.
Zaproponowano metodę uczenia wykorzystującą opracowany algorytm kolejnych przybliżeń ze zmiennym krokiem zmian parametrów (bliższy opis zawiera podrozdział 4.9), w której pojedynczy element może być opisywany wieloma parametrami i która nie wymaga wyostrzania modyfikowanych wartości.

Opracowane i opisane w pracy rozwiązania przetestowano na różnych przykładach i wykazano ich przydatność. Podjęto ponadto próby implementacji tych metod w konkretnych systemach monitorujących. Jedną z takich prób przedstawiono w podrozdziale 5.1, gdzie opisano działanie systemu relacyjnej mapy kognitywnej sprzężonego z monitorowanym pojazdem (na specjalnym stanowisku laboratoryjnym). Kolejną próbę (na poziomie aplikacji operującej danymi w formie zbioru zmierzonych parametrów historycznych) pokazano w podrozdziale 5.2, w którym opisano system służący do budowy i eksploatacji relacyjnej rozmytej mapy kognitywnej. System ten stanowi podstawę do dalszych prac implementacyjnych. Pokazano także kierunki dalszych badań i ścieżki rozwoju zaproponowanych metod.

Dodatek
ZASTOSOWANIE STRUKTUR NEUROPODOBNYCH W PROJEKTOWANIU RELACYJNYCH MAP KOGNITYWNYCH

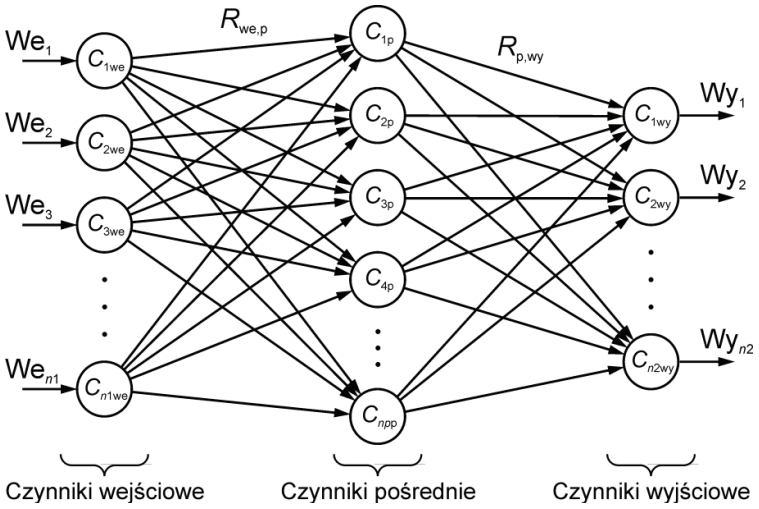
Ogólny model relacyjnej mapy kognitywnej (zarówno w postaci ostrej, jak i rozmytej) przyjmuje postać digrafu (jak na rys. 2.3 czy na rys. 3.10). Przeważnie przyjmuje się przy tym, że wszystkie czynniki mają podobny status, tzn. nie wyróżnia się wśród nich czynników wejściowych ani wyjściowych, co jest spowodowane wielokierunkowością relacji przyczynowo-skutkowych. Tym niemniej, w określonych sytuacjach (np. w diagnostyce symptomowej [65]), model taki może zostać uproszczony do postaci, w której część relacji zostanie usunięta. Wtedy możliwe będzie wskazanie w modelu czynników wejściowych (czyli jedynie wpływających na inne) oraz wyjściowych (czyli takich, których wartości zależą od innych czynników, same jednakże wpływu na inne nie wywierają). Tego rodzaju model można graficznie przedstawić jak na rysunku D.1.



Rys. D.1. *Uproszczony model relacyjnej mapy kognitywnej (z ograniczoną liczbą relacji), w którym wyróżniono czynniki wejściowe i wyjściowe:* n_1 *– liczba czynników wejściowych;* n_2 *– liczba czynników wyjściowych;* $R_{we,wy}$ *– uogólnione oznaczenie relacji pomiędzy czynnikami wejściowymi a wyjściowymi*

Model przedstawiony na rysunku D.1 jest neuropodobną strukturą przypominającą najprostszą postać sztucznej sieci neuronowej (SSN) – jednokierunkowy perceptron neuronowy (bez warstw ukrytych) i może być rozpatrywany podobnie jak tego typu perceptron. Takie konstrukcje były wykorzystywane, z pozytywnym rezultatem, w inteligentnym diagnozowaniu wybranych układów technicznych (np.

w [62, 66, 173, 215]). Podobne analogie można wskazać pomiędzy relacyjną mapą kognitywną a uogólnioną postacią perceptronu, taką jaka została przedstawiona na rysunku D.2. Zachodzą one, kiedy relacje pomiędzy czynnikami również nie są kompletne, mają jednak bardziej zróżnicowany charakter niż w modelu pokazanym na rysunku D.1, tzn. niektóre czynniki wywierają wpływ na inne, chociaż same nie są od innych zależne, część czynników jedynie kumuluje wpływy pozostałych, a część czynników pełni obie wyżej wymienione role.



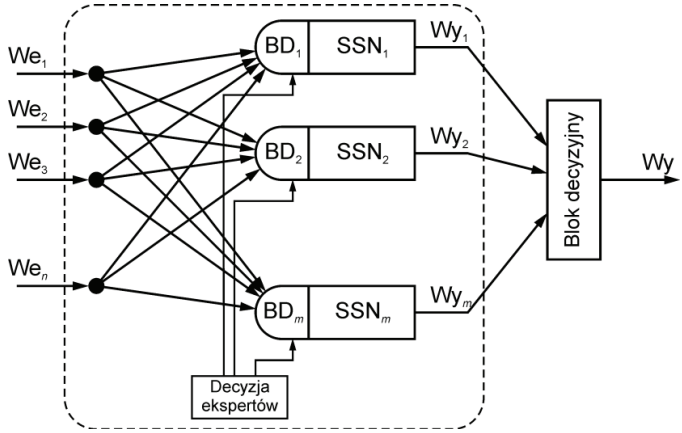
Rys. D.2. *Uproszczony model relacyjnej mapy kognitywnej wykazujący podobieństwo do wielowarstwowego perceptronu: n_1 – liczba czynników wejściowych; n_2 – liczba czynników wyjściowych; n_p – liczba czynników pośrednich; $R_{we,p}$ – uogólnione oznaczenie relacji pomiędzy czynnikami wejściowymi a pośrednimi; $R_{p,wy}$ – uogólnione oznaczenie relacji pomiędzy czynnikami pośrednimi a wyjściowymi*

W modelu, przedstawionym na rysunku D.2 również można wydzielić czynniki wejściowe i wyjściowe, a sam model wykazuje pewne podobieństwo do modeli wielowarstwowych perceptronów neuronowych. W tym przypadku zachodzi jednak istotna różnica koncepcyjna, która wynika z odmiennego sposobu traktowania neuronów i czynników mapy. W modelu relacyjnej mapy kognitywnej występuje swobodny dostęp do wartości wszystkich czynników (również tych, które na rys. D.2 określono mianem „pośrednich"), natomiast w modelu perceptronu (którego wizualizacja graficzna jest w zasadzie identyczna z rys. D.2) neurony warstw ukrytych nie mają bezpośredniego związku z wielkościami modelowanego obiektu i ich wartości wyjściowe nie są bezpośrednio dostępne, co rzutuje także na sposób uczenia modelu. Tym niemniej, w określonych sytuacjach, modele wielowarstwowych perceptronów neuronowych mogą być stosowane równie skutecznie jak relacyjne mapy kognitywne. Z ich użyciem implementowano inteligentne metody diagnozowania nieco bardziej złożonych układów technicznych (m.in. w [39, 41, 67, 68, 69, 70, 211, 213, 217]).

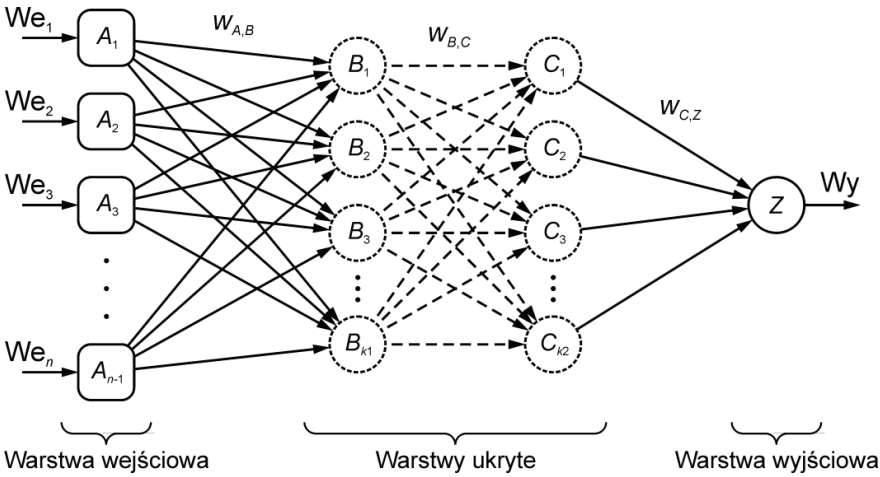
Różne typy sztucznych sieci neuronowych (zarówno ostrych, jak i rozmytych), podobnych do tych jakie pokazano na rysunkach D.1 i D.2, są powszechnie wykorzystywane, zwłaszcza przy rozwiązywaniu zadań klasyfikacji. Podejmowano także udane próby wykorzystania tego rodzaju struktur w procesach diagnozowania i monitoringu niektórych rodzajów systemów. Cechą charakterystyczną metod wykorzystujących SSN jest (podobnie jak w mapach kognitywnych) konieczność wyodrębnienia wybranych wielkości badanego systemu i przedstawienia ich w formie dogodnej do przetwarzania. W sieciach ostrych stosuje się różne metody normalizacji (np. adekwatne do (2.3)) wartości poszczególnych sygnałów i wtedy wzajemne oddziaływania neuronów mają formę połączeń z wagami liczbowymi. W sieciach rozmytych stosuje się dwa podejścia. W pierwszym model SSN jest w pełni rozmyty, czyli neurony mają budowę rozmytą, a ich kolejne wartości wyznacza się w oparciu o zbiory reguł (metodą zbliżoną do IF-THEN z równania (3.1)). W drugim, rozmyte wartości neuronów są transponowane do postaci liczb rzeczywistych, po czym model pracuje jak wspomniany wyżej model ostry.

Stosowanie modeli takich jak te przedstawione na rysunku D.1 lub D.2 opiera się na założeniu, że do analizy pracy całego badanego systemu wystarcza jeden rozbudowany model, którego wyjścia będą sygnalizatorami (znacznikami) określonych stanów systemu. W bardziej złożonych zastosowaniach pojedyncza SSN nie jest w stanie wystarczająco precyzyjnie oddać działania modelowanego obiektu. Sytuacja taka zachodzi, jeśli badaniu poddaje się wiele czynników lub jeśli czynniki te są w jakiś sposób pogrupowane, przy czym powiązania wewnątrzgrupowe są silniejsze niż powiązania pomiędzy grupami. Taki układ wzajemnych, nie zawsze jasnych, powiązań bardzo utrudnia działanie systemów, których zadaniem jest identyfikacja problemów i podejmowanie jednoznacznych decyzji. SSN tworzona na potrzeby modelowania takiego obiektu musiałaby być bardzo złożona, a jej obsługa – uciążliwa (zwłaszcza w aspekcie uczenia i wnioskowania). Zamiast tego można zastosować strukturę, nazwaną bankiem sztucznych sieci neuronowych, w której, zamiast jednej rozbudowanej SSN, wprowadzono zestaw prostszych SSN dzielących pomiędzy siebie różne zadania. Podejście takie zostało opracowane i przetestowane na potrzeby monitorowania pracy wybranych systemów technicznych pod kątem lokalizacji uszkodzeń i decydowania o dalszych działaniach [40, 67, 211–214, 216], a zastosowana struktura banku miała postać jak na rysunku D.3.

Cechą charakterystyczną sieci stanowiących składniki banku jest ich struktura wewnętrzna, w której występuje tylko jedno wyjście – jak na rysunku D.4.



Rys. D.3. *Ogólny schemat systemu decyzyjnego zbudowanego w oparciu o bank sztucznych sieci neuronowych:* n *– liczba czynników informacyjnych (symptomów);* m *– liczba rozważanych problemów;* BD_i *–* i*-ty blok dyskryminacyjny;* SSN_i *–* i*-ta sztuczna sieć neuronowa*



Rys. D.4. *Przykładowa reprezentacja graficzna sztucznej sieci neuronowej (SSN) o* n *wejściach i* 1 *wyjściu;* k_1*,* k_2*, ... – liczby neuronów kolejnych warstw ukrytych*

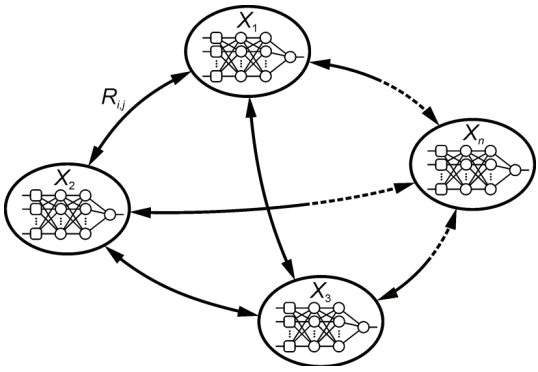
SSN stosowane w budowie konstrukcji (rys. D.3) podlegają procesowi uczenia nadzorowanego zgodnie z ogólnymi regułami przyjętymi dla jednokierunkowych perceptronów (np. z użyciem algorytmu wstecznej propagacji sygnałów) tak, aby każda z nich była w stanie, w oparciu o sygnały wejściowe, zlokalizować pewien wyodrębniony stan badanego systemu.

Struktury takie, jak te pokazane na rysunku D.3, dobrze sprawdzają się w badaniach i modelowaniu systemów statycznych, zwłaszcza opisywanych zmiennymi rozmytymi, tym niemniej posiadają ograniczenia związane ze sposobem działania SSN. Sposób ten opiera się na wyraźnym wyodrębnieniu elementów (neuronów, czynników) wejściowych i wyjściowych, ponieważ analizie poddaje się jednokie-

runkowy przepływ sygnałów – od wejścia do wyjścia. W systemach o dynamicznej strukturze wewnętrznej, w których przepływ sygnałów jest wielokierunkowy, bezpośrednie stosowanie SSN jest bardzo trudne. Jednakże zarówno same SSN, jak i ich banki posiadają zalety (głównie elastyczność działania), które byłyby przydatne przy modelowaniu systemów dynamicznych.

Prezentowane w rozdziałach 2 i 4 modele różnych typów map kognitywnych działają w oparciu o pewną, wspólną zasadę – kolejną wartość danego czynnika (zarówno w systemach ostrych, jak i rozmytych) wyznacza się jako funkcję sumy bieżącej wartości tego czynnika oraz skumulowanego oddziaływania czynników wpływających (przykładowe rozwiązania pokazują równania (2.10), (3.80) oraz (4.3)). Tymczasem stosowanie prostego sumatora nie zawsze daje zadowalające efekty. Z drugiej strony wypracowywanie złożonych konstrukcyjnie funkcji agregujących (a każdy czynnik powinien mieć właściwie własną funkcję tego rodzaju), chociaż możliwe, byłoby nieefektywne, zwłaszcza z punktu widzenia procesów adaptacji (uczenia) modelu. Z tego powodu, poszukując metod tworzenia bardziej elastycznych rozwiązań konstrukcyjnych czynników, można rozważyć inne niż arytmetyczne podejście do agregacji wielkości wejściowych. Pojedynczy czynnik można budować w oparciu o strukturę inteligentną, taką jak np. jednokierunkowa sztuczna sieć neuronowa z wieloma wejściami i pojedynczym wyjściem (w rodzaju wielowarstwowego perceptronu, takiego jak na rys. D.4).

Biorąc pod uwagę wyżej wymienioną koncepcję, rozpoczęto badania nad włączeniem wybranych cech SSN do struktur relacyjnych map kognitywnych (zarówno ostrych, jak i rozmytych). Przyjęto założenie, że działanie głównego, dynamicznego algorytmu pracy mapy kognitywnej pozostanie niezmienione, jednakże modyfikacji ulegnie wewnętrzna budowa wybranych czynników. Zostaną one zaprojektowane jako odrębne sztuczne sieci neuronowe lub banki sztucznych sieci neuronowych (zależnie od specyfiki problemu). Istotne jest przy tym zachowanie głównej formy czynnika, tzn.: wiele wejść, jedno wyjście. Ogólny wygląd tak zmodyfikowanej mapy kognitywnej przedstawia rysunek D.5.



Rys. D.5. *Ogólny schemat zmodyfikowanej relacyjnej mapy kognitywnej, której czynniki zawierają sztuczne sieci neuronowe.* X_1-X_n *– wartości czynników;* n *– liczba czynników;* $R_{i,j}$ *– uogólniona reprezentacja relacji pomiędzy czynnikami*

Podobnie jak przy opisanych wcześniej RMK oraz RRMK, do wyznaczania kolejnych wartości czynników mapy przedstawionej na rysunku D.5 można posłużyć się różnymi modelami podstawowymi, jednakże ich charakter będzie się różnił od stosowanego dotychczas sumowania oddziaływań. Na przykład dla modelu ostrego – nieliniowego z nieliniowością szybkości zmian, opisanego równaniem (2.12), wartość czynnika w kolejnym punkcie czasu dyskretnego będzie opisana następująco:

$$x_j(t+1) = f_p\big(x_j(t) + \Delta x_j(t)\big) \tag{D.1}$$

gdzie:

$\Delta x_j(t) = f_{SSN_j}\left(\left\{r_{i,j} \cdot \big(x_i(t) - x_i(t-1)\big)\right\}_{i=1,..n;i\neq j}\right)$ – przyrost wartości x_j pod wpływem działania pozostałych czynników;

f_p – funkcja progowa;

n – liczba czynników;

f_{SSN_j} – ogólna reprezentacja działania SSN należącej do struktury j-tego czynnika.

Podobnej modyfikacji ulegnie również wyrażenie (4.3) opisujące model RRMK.

O ile sam sposób działania tak zmodyfikowanego modelu relacyjnej mapy kognitywnej nie jest bardziej złożony od dotychczasowego, o tyle algorytmy uczenia muszą być, z konieczności, całkowicie przebudowane. Zaproponowane w podrozdziale 2.3 techniki uczenia RMK oparte na metodzie gradientowej nie mają zastosowania do sztucznych sieci neuronowych. W związku z powyższym dla RMK, których czynniki zawierają SSN należy sięgać po metody populacyjne (niektóre z nich przedstawiono w podrozdziale 3.2.4). Nieco inaczej wygląda problem uczenia RRMK z czynnikami w postaci SSN. Po pierwsze, stosowane modele SSN powinny być przystosowane do przetwarzania liczb rozmytych, co wymaga opracowania nowych technik uczenia, ponieważ problem ten nie został jeszcze rozwiązany. Po drugie, zaproponowana w podrozdziale 4.9 metoda uczenia RRMK wykorzystująca algorytm kolejnych przybliżeń ze zmiennym krokiem zmian parametrów jest metodą populacyjną, co czyni ją łatwiejszą do przystosowania do nowego charakteru czynników.

Wprowadzenie sztucznych sieci neuronowych lub banków sztucznych sieci neuronowych do struktur relacyjnych map kognitywnych może docelowo znacząco poprawić jakość pracy takich modeli. Wiąże się to jednak z koniecznością rozwiązania szeregu problemów natury teoretycznej i praktycznej. Obecnie trwają prace nad stworzeniem odpowiednich metod uczenia zarówno dla RMK, jak i RRMK, które pozwoliłyby na zastosowanie SSN do budowy modeli poszczególnych czynników.

Literatura

[1] Aguilar J., *Adaptive random fuzzy cognitive maps* [in:] IBERAMIA 2002, Lecture "Notes in Artificial Intelligence", eds. Garijio F.J., Riquelme J.C., Toro M., Vol. 2527, Springer-Verlag, Berlin-Heidelberg (Germany) 2002, pp. 402–410.

[2] Aguilar J., *Dynamic Random Fuzzy Cognitive Maps*, "*Computación y Sistemas*" 2004, Vol. 7, No. 4, pp. 260–270.

[3] Aickelin U., Dasgupta D., *Artificial Immune Systems* [in:] *Search Methodologies: Introductory Tutorials in Optimization and Decision Support Techniques*, eds. Burke E.K., Kendall G., Springer Science+Business Media, LLC, ch. 13, New York (USA) 2005, pp. 375–399.

[4] Alizadeh S., Ghazanfari M., *Learning FCM by chaotic simulated annealing*, "Chaos, Solitons & Fractals" 2009, Vol. 41, No. 3, pp. 1182–1190.

[5] Alizadeh S., Ghazanfari M., Fathian M., *Using Data Mining for Learning and Clustering FCM*, "International Journal of Information and Mathematical Sciences" 2008, Vol. 4, No. 2, pp. 118–125.

[6] Alizadeh S., Ghazanfari M., Jafari M., Hooshmand S., *Learning FCM by Tabu Search*, "International Journal of Electrical and Computer Engineering" 2007, Vol. 2, No. 2, pp. 142–149.

[7] Andreou A.S., Mateou N.H., Zombanakis G.A., *Evolutionary Fuzzy Cognitive Maps: A Hybrid System for Crisis Management and Political Decision-Making* [in:] Proc. of the Computational Intelligent for Modeling, Control & Automation CIMCA, Vienna (Austria) 2003, pp. 732–743.

[8] Anthony M., Bartlett P.L., *Neural Network Learning: Theoretical Foundations*, Cambridge University Press, New York (USA) 2009.

[9] Arabas J., *Wykłady z algorytmów ewolucyjnych*, WNT, Warszawa 2001.

[10] Axelrod R., *Structure of Decision: the cognitive maps of political elites*, Princeton Univ. Press, New York (USA) 1976.

[11] Bandler W., Kohout L.J., *Fuzzy power sets and fuzzy implication operators*, "Fuzzy Sets and Systems" 1980, Vol. 4, pp. 13–30.

[12] Baykasoglu A., Durmusoglu Z.D.U., Kaplanoglu V., *Training Fuzzy Cognitive Maps via Extended Great Deluge Algorithm with Applications*, "Computers in Industry" 2011, Vol. 62, No. 2, pp. 187–195.

[13] Bellman R.E., Zadeh L.A., *Decision making in a fuzzy environment*, "Management Science" 1970, Vol. 17, pp. 141–164.

[14] Bezdek J.C., *What is computational intelligence?* [in:] *Computational Intelligence: Imitating Life*, eds. Zuranda M.J., Marks II R.J., Robinson C.J., IEEE Press, New York (USA) 1994, pp. 1–12.

[15] Boche R.E., *An operational interval arithmetic* [in:] Proc. of Illinois National Electronics Conference, Paper No. CP-1431 CD-1, Chicago 1963.

[16] Borisov W.W., Kruglov W.W., Fedulov A.S., *Rozmyte modele i sieci*, Gorąca Linia – Telekom, Moskwa (Rosja) 2007 (w jęz. rosyjskim).

[17] Cantú-Paz E., *A Survey of Parallel Genetic Algorithms*, "Calculateurs Parallèles, Réseaux et Systèmes Répartis" 1998, Vol. 10, No. 2, pp. 141–171.

[18] Cantú-Paz E., *Parameter setting in parallel genetic algorithms* [in:] *Parameter Setting in Evolutionary Algorithms. Studies in Computational Intelligence* (*SCI*), eds. Lobo F., Lima C., Michalewicz Z., Vol. 54, Springer-Verlag, Berlin-Heidelberg (Germany) 2007, pp. 259–276.

[19] Carvalho J.P., Tomé J.A., *Fuzzy Mechanisms for Qualitative Causal Relations* [in:] *Studies in Fuzziness and Soft Computing. Views on Fuzzy Sets and Systems from Different Perspectives*, ed. by Seising R., Vol. 243/2009, Springer-Verlag, Berlin-Heidelberg 2009, pp. 393–415.

[20] Carvalho J.P., Tomé J., *Qualitative Optimization of Fuzzy Causal Rule Bases using Fuzzy Boolean Nets*, "Fuzzy Sets and Systems" 2007, Vol. 158, No. 17, pp. 1931–1946.

[21] Carvalho J.P., Tomé J.A., *Rule Based Fuzzy Cognitive Maps – Qualitative Systems Dynamics*, Proc. of the 19th International Conference of the North American Fuzzy Information Processing Society, Atlanta (USA) 2000, pp. 407–411.

[22] Carvalho J.P., Tomé J.A., *Rule-based fuzzy cognitive maps – Expressing Time in Qualitative System Dynamics* [in:] Proc. of the FUZZ-IEEE'2001, Melbourne (Australia) 2001, pp. 280–283.

[23] Chaib-draa B., Decharnais J., *A relational model of cognitive maps*, Vol. 49, 1988, pp. 137–148.

[24] Cordero O.X., Peláez E., *Fuzzy Numbers for the Improvement of Causal Knowledge Representation in Fuzzy Cognitive Maps*, Proc. of the 18th National Conference on Artificial Intelligence and 14th Conference on Innovative Applications of Artificial Intelligence, Edmonton (Canada) 2002, pp. 949–955.

[25] Cpałka K., *Zagadnienie interpretowalności wiedzy i dokładności działania systemów decyzyjnych*, Akademicka Oficyna Wydawnicza EXIT, Warszawa 2009.

[26] Czogała E., Pedrycz W., *On identification in fuzzy systems and its applications in control problems*, "Fuzzy Sets and Systems" 1981, Vol. 6, pp. 73–83.

[27] Das S., Abraham A., Konar A., *Differential Evolution Algorithm: Foundations and Perspectives* [in:] *Metaheuristic Clustering. Studies in Computational Intelligence*, Vol. 178, Springer-Verlag, Berlin-Heidelberg (Germany) 2009, ch. 2, pp. 63–110.

[28] Dasgupta D. (ed)., *Artificial immune systems and their applications*, Springer, Berlin (Germany) 1999.

[29] Dawkins R., *The Selfish Gene*, Oxford Univ. Press, New York (USA) 1976.

[30] Dickerson J.A., Kosko B., *Virtual worlds as fuzzy cognitive maps*, "Presence" 1994, Vol. 3, No. 2, pp. 173–189.

[31] Dubois D., Prade H., *Fuzzy Sets and Systems: Theory and Applications*, Academic Press Inc., San Diego–New York–Boston (USA) 1980.

[32] Duch W., Korbicz J., Rutkowski L., Tadeusiewicz R., *Biocybernetyka i inżynieria medyczna. Sieci neuronowe*, Vol. 6, Akademicka Oficyna Wydawnicza Exit, Warszawa 2000.

[33] Dueck G., *New Optimization Heuristics: The Great Deluge Algorithm and the Record-to-Record Travel*, "Journal of Computational Physics" 1993, Vol. 104, No. 1, pp. 86–92.

[34] Eykhoff P., *System Identification: Parameter and State Estimation*, John Wiley & Sons, New York (USA) 1974.

[35] Fodor J.C., Roubens M., *Fuzzy preference modelling and multicriteria decision support*, Kluwer Academic Publishers, Dordrecht (Netherlands) 1994.

[36] Fogel L.J., Owens A.J., Walsh M.J., *Artificial Intelligence through Simulated Evolution*, John Wiley & Sons, New York (USA) 1966.

[37] Froelich W., Juszczuk P., *Predictive Capabilities of Adaptive and Evolutionary Fuzzy Cognitive Maps – A Comparative Study* [in:] *Intelligent Systems for Knowledge Management, SCI 252*, eds. Nguyen N.T., Szczerbicki E., Springer-Verlag, Berlin-Heidelberg (Germany) 2009, pp. 153–174.

[38] Gabus A., Fontela E., *World problems, an invitation to further thought within the framework of DEMATEL*, Battelle Geneva Research Centre, Geneva (Switzerland) 1972.

[39] Gad S., Łaskawski M., Słoń G., Yastrebov A., Zawadzki A., *Fuzzy-Neural Networks in the Diagnosis of Motor-Car's Current Supply Circuit*, "Artificial Intelligence and Soft Computing, Lecture Notes in Artificial Intelligence" 2004, Vol. 3070, pp. 296–301.

[40] Gad S., Łaskawski M., Słoń G., Yastrebov A., Zawadzki A., *Symptom models of diagnostic of motor-car electrical equipment*, Proc. of First IFAC Symposium on "Advances in Automotive Control" IFAC AAC 04, Salerno 2004, pp. 422–427.

[41] Gad S., Słoń G., Jastriebow A., *Analiza komputerowa diagnozowania defektów akumulatora za pośrednictwem sztucznych sieci neuronowych*, Materiały VII Konferencji Naukowo-Technicznej Zastosowania Komputerów w Elektrotechnice, ZKwE'2002, Poznań-Kiekrz 2002, s. 671–674.

[42] Gawroński R., *Rozpoznanie i decyzja*, Państwowe Wydawnictwo Naukowe, Warszawa 1970.

[43] Ghazanfari M., Alizadeh S., Fathian M., Koulouriotis D.E., *Comparing simulated annealing and genetic algorithm in learning FCM*, "Applied Mathematics and Computation" 2007, Vol. 192, pp. 56–68.

[44] Giarratano J.C., Riley G.D., *Expert systems. Principles and Programming. Course Technology*, Thomson Learning Inc., 2005.

[45] Glover F., *Tabu Search-Part I*, "ORSA Journal on Computing" 1989, Vol. 1, No. 3, pp. 190–206.

[46] Glover F., *Tabu Search-Part II*, "ORSA Journal on Computing" 1990, Vol. 2, No. 1, pp. 4–32.

[47] Glover F., Laguna M., *Tabu Search*, Kluwer Academic Publishers, Norwell-MA (USA) 1997.

[48] Goldberg D.E., *Algorytmy genetyczne i ich zastosowania*, wyd. 3, Wydawnictwo Naukowo-Techniczne, Warszawa 2003.

[49] Gorzałczany M.B., *Computational Intelligence Systems and Applications: Neuro-Fuzzy and Fuzzy Neural Synergisms*, A Springer-Verlag Physica-Verlag Company, Heidelberg (Germany) 2002.

[50] Hagiwara M., *Extended fuzzy cognitive maps*, IEEE International Conference on Fuzzy Systems, San Diego-CA 1992, pp. 795–801.

[51] Hamacher H., *Über logische Verknüpfungen unscharfer Aussagen und deren zugehörige Bewertungsfunktionen* [in:] *Progress in Cybernetics and Systems*

Research, eds. Trappl W.R., Klir G.J., Ricciardi L., Vol. 3, Hemisphere Publ. Comp., Washington-DC (USA) 1978, pp. 276–288.

[52] Han J., Kamber M., *Data Mining: Concepts and Techniques. Solution Manual*, 2nd ed., Morgan Kaufmann, San Francisco-CA (USA) 2006.

[53] Haykin S., *Neural Networks: A Comprehensive Foundation*, 29th ed., Pearson Education Inc., Delhi (India) 2005.

[54] He Y., *Application Study in Decision Support with Fuzzy Cognitive Map* [in:] *KES 2008,* P. II, *Lecture Notes on Artificial Intelligence*, eds. Lovrek I., Howlett R.J., Jain L.C., Vol. 5178, Springer-Verlag, Berlin-Heidelberg (Germany) 2008, pp. 324–331.

[55] Hebb D.O., *The Organization of Behavior: A Neuropsychological Theory*, Taylor & Francis, Psychology Press, 2002 (reprint wydania z 1949 r.).

[56] He Y., Lin Y., Guo Y., *Goal-Oriented Decision Support Based on Fuzzy Cognitive Map*, Proc. of the 7th WSEAS International Conference on Simulation, "Modelling and Optimization", Beijing (China) 2007, pp. 374–379.

[57] Hengjie Song, Chunyan Miao, Roel W., Zhiqi Shen, *Implementation of Fuzzy Cognitive Maps Based on Fuzzy Neural Network and Application in Prediction of Time Series*, "IEEE Trans. Fuzzy Systems" 2010, Vol. 18, No. 2, pp. 233–250.

[58] Herrera F., Lozano M., Verdegay J.L., *Tackling Real-Coded Genetic Algorithms: Operators and Tools for Behavioural Analysis*, "Artificial Intelligence Review" 1998, Vol. 12, pp. 265–319.

[59] Holland J.H., *Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence*, University of Michigan Press, USA 1975.

[60] Hooke R., Jeeves T.A., *Direct search solution of numerical and statistical problems*, "Journal of the Association for Computing Machinery" 1961, Vol. 8, pp. 212–229.

[61] Huerga A., *A balanced differential learning algorithm in fuzzy cognitive maps*, Proc. of the 16th International Workshop on Qualitative Reasoning, QR'2002, Spain 2002, pp. 210–214.

[62] Jastriebow A., Gad S., Słoń G., *Analiza komputerowa diagnozowania defektów wyposażenia elektrycznego samochodów z wykorzystaniem sztucznych sieci neuronowych*, XIV Krajowa Konferencja Automatyki KKA'2002, Vol. I, Zielona Góra 2002, s. 605–610.

[63] Jastriebow A., Gad S., Słoń G., *Mapy kognitywne w monitorowaniu decyzyjnym systemów*, Studia i Materiały Polskiego Stowarzyszenia Zarządzania Wiedzą, 2011, nr 47, s. 64–77.

[64] Jastriebow A., Gad S., Słoń G., *Rozmyte mapy kognitywne w monitorowaniu decyzyjnym obiektów technicznych*, Biuletyn Wojskowej Akademii Technicznej, 2010, Vol. LIX, nr 4, s. 209–219.

[65] Jastriebow A., Gad S., Słoń G., *Synteza i analiza symptomowych metod diagnozowania*, „Diagnostyka" 2008, nr 1(45), s. 131–134.

[66] Jastriebow A., Gad S., Słoń G., *Sztuczne sieci neuronowe w inteligentnych decyzyjnych systemach diagnostycznego monitorowania układów technicznych* [w:] *Sterowanie i automatyzacja: aktualne problemy i ich rozwiązania*, pod red. Malinowski K., Rutkowski L., Akademicka Oficyna Wydawnicza EXIT, Warszawa 2008, s. 660–669.

[67] Jastriebow A., Gad S., Słoń G., Kałwa D., Łaskawski M., *Metody sztucznej inteligencji w diagnostyce wyposażenia elektrycznego samochodów*, t. 3, WNT, Warszawa; Warszawa 20", XV Krajowa Konferencja Automatyki, Vol. 3, Warszawa 2005, s. 285–290.

[68] Jastriebow A., Gad S., Słoń G., Łaskawski M., *Inteligentne metody diagnozowania uszkodzeń elektrycznego wyposażenia samochodu*, Zeszyty Naukowe Politechniki Świętokrzyskiej „Elektryka" 2005, nr 43, pod red. Nadolski R., Wyd. Politechniki Świętokrzyskiej, Kielce 2005, s. 85–90.

[69] Jastriebow A., Gad S., Słoń G., Łaskawski M, *Neuronowo-rozmyte metody lokalizacji uszkodzeń elektrycznego wyposażenia samochodu*, „Pomiary, Automatyka, Kontrola" 2005, nr 9bis, s. 118–120.

[70] Jastriebow A., Gad S., Słoń G., Zawadzki A., Pawlak M., *Sygnały diagnostyczne w diagnozowaniu wyposażenia elektrycznego samochodów*, „Diagnostyka" 2004, Vol. 32, s. 139–142.

[71] Jastriebow A., Piotrowska K., *Modelowanie ekspertowych systemów logistycznych opartych na relacyjnych mapach kognitywnych*, „Logistyka" 2012, nr 3, s. 871–878.

[72] Jastriebow A., Piotrowska K., Słoń G., *Algorithm of Building the Intelligent Models Based on Relational Cognitive Maps* [in:] Computer Technologies in Science, Technology and Education, eds. Jastriebow A., Kuźmińska-Sołśnia B., Raczyńska M., Institute for Sustainable Technologies – National Research Institute, Radom 2012, pp. 138–151.

[73] Jastriebow A., Słoń G., *Accuracy of the intelligent dynamic models of relational fuzzy cognitive maps* [in:] *Computer Applications in Electrical Engineering*, ed. by Nawrowski R., Vol. 8, Poznan University of Technology, Poznań 2010, pp. 150–160.

[74] Jastriebow A., Słoń G., *Logistyczne zastosowania modelu rozmytej relacyjnej mapy kognitywnej*, „Logistyka" 2012, nr 3, s. 879–886.

[75] Jastriebow A., Słoń G., *Modelowanie kognitywne systemów i maszyn*, Studia i Materiały Polskiego Stowarzyszenia Zarządzania Wiedzą, 2011, nr 47, s. 78–91.

[76] Jastriebow A., Słoń G., *Modelowanie słabostrukturalnych systemów logistycznych oparte na rozmytych relacyjnych mapach kognitywnych*, „Logistyka" 2010, Vol. 2/2012, s. 143–152.

[77] Jastriebow A., Słoń G., *Obliczenia ziarniste w modelowaniu nieprecyzyjnych obiektów przy użyciu relacyjnych rozmytych map kognitywnych*, „Pomiary, Automatyka, Kontrola" 2010, Vol. 56, nr 12/2010, s. 1449–1452.

[78] Jastriebow A., Słoń G., *Parametric identification of dynamic imprecise models based on relational fuzzy cognitive maps* [in:] *Computer Applications in Electrical Engineering*, ed. by Nawrowski R., Vol. 9, Poznań: Poznan University of Technology, 2011, pp. 234–244.

[79] Jastriebow A., Słoń G., *Relational Fuzzy Cognitive Maps in Modelling Low-Structural Closed Systems*, Proc. of 13th International Conference System Modelling and Control SMC 2009, Zakopane 2009 (wersja elektroniczna, s. 8).

[80] Jastriebow A., Słoń G., *Relational Modelling in Monitoring Technical States of Objects*, „Logistyka" 2009, Vol. 6 (wersja elektroniczna, s. 8).

[81] Jastriebow A., Słoń G., *Rozmyte mapy kognitywne w relacyjnym modelowaniu systemów monitorowania* [w:] *Systemy wykrywające, analizujące i tolerujące usterki*, pod red. Kowalczuk Z., PWNT, Gdańsk 2009, s. 217–227.

[82] Jastriebow A., Słoń G., *Synteza i analiza obliczeniowa modeli inteligentnych opartych na mapach kognitywnych*, cz. I, *Synteza* [w:] *Informatyka w dobie XXI wieku. Technologie informatyczne i ich zastosowania*, pod red. Jastriebowa A., Wyd. Naukowe Instytutu Technologii Eksploatacyjnej – Państwowego Instytutu Badawczego, Radom 2010, s. 67–76.

[83] Jastriebow A., Słoń G., *Synteza i analiza obliczeniowa modeli inteligentnych opartych na mapach kognitywnych*, cz. II, *Analiza* [w:] *Informatyka w dobie XXI wieku. Technologie informatyczne i ich zastosowania*, pod red. Jastriebow A., Wyd. Naukowe Instytutu Technologii Eksploatacyjnej – Państwowego Instytutu Badawczego, Radom 2010, s. 77–86.

[84] Jerne N.K., *Towards a network theory of the immune system*, "*Ann. Immunol.*" (Inst Pasteur), 1974, Vol. 125C, No. 1/2, pp. 373–389.

[85] Juszczuk P., Froelich W., *Learning Fuzzy Cognitive Maps Using a Differential Evolution Algorithm*, "Polish Journal of Environmental Studies" 2009, Vol. 18, No. 3B, pp. 108–112.

[86] Kacprzyk J., *Fuzzy dispositions as a tool for more human-perception-consistent models in systems analysis* [in:] *Fuzzy Sets: Applications, Methodological Approaches and Results*, ed. by Bocklisch W.S. et al., Vol. 30, Mathematical Research, Akademie-Verlag, Berlin (Germany) 1986, pp. 64–73.

[87] Kacprzyk J., *Towards human-consistent multistage decision making and control models via fuzzy sets and fuzzy logic*, "Fuzzy Sets and Systems – Special issue: Dedicated to the memory of Richard E. Bellman" 1986, Vol. 18, No. 3, pp. 299–314.

[88] Kacprzyk J., *Wieloetapowe sterowanie rozmyte*, WNT, Warszawa 2001.

[89] Kacprzyk J., *Zastosowanie teorii zbiorów rozmytych do optymalnego przydziału stanowisk pracy*, „Archiwum Automatyki i Telemechaniki" 1975, Vol. XX, s. 247–260.

[90] Kandasamy W.B.V., Smarandache F., *Fuzzy Cognitive Maps and Neutrosophic Cognitive Maps*, Xiquan-Phoenix (USA) 2003.

[91] Kandasamy W.B.V., Smarandache F., Ilanthenral K., *Elementary Fuzzy Matrix Theory and Fuzzy Models for Social Scientists*, Automaton, American Research Press, Los Angeles (USA) 2007.

[92] Kasabov N.K., *Foundations of Neural Networks, Fuzzy Systems and Knowledge Engineering*, 2nd ed., A Bradford Book The MIT Press Cambridge, Masschusetts, London (England) 1998.

[93] Kaufmann A., *Introduction to the Theory of Fuzzy Subsets*, Vol. 1, Academic Press, New York (USA) 1975.

[94] Kaufmann A., Gupta M.M., *Introduction to Fuzzy Mathematics – Theory and Applications*, Van Nostrand Reinhold, New York (USA) 1985.

[95] Kennedy J., Eberhart R.C., Shi Y., *Swarm Intelligence* (*The Morgan Kaufmann Series in Evolutionary Computation*), Morgan Kaufmann Publishers, San Francisco-CA (USA) 2001.

[96] Khan M.S., Chong A., *Fuzzy Cognitive Map Analysis with Genetic Algorithm* [in:] Proc. of the 1st Indian International Conference on Artificial Intelligence, IICAI 2003, Hyderabad (India) 2003, pp. 1196–1205.

[97] Kirkpatrick S., Gelatt C.D., Vecchi, *Optimization by Simulated Annealing*, "*Science, New Series*", Vol. 220, No. 4598, pp. 671–680, 1983.

[98] Klir G.J., Yuan B., *Fuzzy Sets and Fuzzy Logic: Theory and Applications*, Upper Saddle River, Prentice Hall PTR, New Jersey (USA) 1995.

[99] Konar A., *Computational Intelligence. Principles, Techniques and Applications*. Berlin-Heidelberg, Springer, New York 2005.

[100] Konar A., *Distributed Machine Learning Using Fuzzy Cognitive Maps* [in:] *Computational Intelligence: Principles, Techniques and Applications*, ch. 18, Springer-Verlag, Berlin-Heidelberg (Germany) 2005, pp. 497–520.

[101] Konar A., Chakraborty U.K., *Reasoning and unsupervised learning in a fuzzy cognitive map*, "Information Sciences" 2005, Vol. 170, No. 2–4, pp. 419–441.

[102] Konar A., Jain L., *Distributed Learning Using Fuzzy Cognitive Maps*,[in:] *Cognitive Engineering: A Distributed Approach to Machine Intelligence*, ch. 6, Springer-Verlag, London (England) 2005, pp. 181–204.

[103] Korbicz J., Kościelny J.M., Kowalczuk Z., Cholewy H., *Diagnostyka procesów. Modele. Metody sztucznej inteligencji*, WNT, Warszawa 2002.

[104] Kosiński W., Prokopowicz P., *Algebra liczb rozmytych*, „Matematyka Stosowana", Vol. 5, 2004, s. 37–63.

[105] Kosiński W., Prokopowicz P., Ślęzak D., *Drawback of fuzzy arithmetics – new intuitions and propositions* [in:] Proc. of AI METH, Methods of Artificial Intelligence, Gliwice 2002, pp. 231–237.

[106] Kosiński W., Prokopowicz P., Ślęzak D., *On algebraic operations on fuzzy numbers* [in:] *Intelligent Information Processing and Web Mining*, Proc. of the Internationl IIS: IIPWM'03, Zakopane 2003, Kłopotek M., Wierzchoń S.T., Trojanowski K., Heidelberg: Physica-Verlag, 2003, pp. 353–362.

[107] Kosiński W., Prokopowicz P., Ślęzak D., *Ordered fuzzy numbers*, "Bulletin of the Polish Academy of Sciences" 2003, Vol. 51, No. 3, pp. 327–338.

[108] Kosko B., *Fuzzy cognitive maps*, "Int. Journal of Man-Machine Studies" 1986, Vol. 24, pp. 65–75,.

[109] Koulouriotis D.E., Diakoulakis I.E., Emiri D.M., Antonidakis E.N., Kaliakatsos I.A., *Efficiently Modeling and Controlling Complex Dynamic Systems using Evolutionary Fuzzy Cognitive Maps* (*Invited Paper*), "International Journal of Computational Cognition" 2003, Vol. 1, No. 2, pp. 41–65.

[110] Koulouriotis D.E., Diakoulakis I.E., Emiris D.M., *Learning Fuzzy Cognitive Maps using Evolution Strategies: a Novel Schema for Modeling and Simulating High-Level Behavior*, Proc. of the Congress on Evolutionary Computation, Vol. 1, Seoul (Korea) 2001, pp. 364–371.

[111] Krasnogor N., Smith J., *Tutorial for Competent Memetic Algorithms: Model, Taxonomy, and Design Issues*, "IEEE Transactions on Evolutionary Computation" 2005, Vol. 9, No. 5, pp. 474–488.

[112] Kuniszyk-Jóźkowiak W., *Algorytmy logiki rozmytej*, Uniwersytet Marii Curie-Skłodowskiej, praca współfinansowana ze środków Unii Europejskiej w ramach Europejskiego Funduszu Społecznego, Lublin (Polska) 2012.

[113] Larose D.T., *Odkrywanie wiedzy z danych. Wprowadzenie do eksploracji danych*, Wydawnictwo Naukowe PWN, Warszawa 2006.

[114] Lewiński A., Bester L., *Additional warning system for cross level* [in:] *Communications in Computer and Information Science (104)*, Springer-Verlag, Berlin-Heidelberg 2010, pp. 226–231.

[115] Lewiński A., Perzyński T., *The modeling of multicomputer structure in railway control applications* [in:] *Advances in Transport Systems Telematics*, Silesian University of Technology, Katowice 2006, pp. 253–258.

[116] Lewiński A., Perzyński T., *The modelling of computer networks for railway control and management*, Prace konferencji Instytutu Transportu Politechniki Śląskiej TRANSPORT SYSTEMS TELEMATICS – 2004, Katowice–Ustroń 2004.

[117] Lewiński A., Perzyński T., *The Reliability and Safety of Railway Control Systems Based on New Information Technologies* [in:] *Communications in Computer and Information Science (104)*, Springer-Verlag, Berlin-Heidelberg 2010, pp. 427–433.

[118] Lin C., *An immune algorithm for complex fuzzy cognitive map partitioning* [in] Proc. of the first ACM/SIGEVO Summit on Genetic and Evolutionary Computation, Shanghai (China) 2009, pp. 315–320.

[119] Lin C., Chen K., He Y., *Learning Fuzzy Cognitive Map Based on Immune Algorithm*, "WSEAS Transactions on Systems" 2007, Vol. 6, No. 3, pp. 582–588.

[120] Luo X., Wei X., Zhang V., *Game-based Learning Model Using Fuzzy Cognitive Map*, [in:] Proc. of the First ACM International Workshop on Multimedia Technologies for Distance Learning MTDL '09, Beijing (China) 2009, pp. 67–76.

[121] Łachwa A., *Rozmyty świat zbiorów, liczb, relacji, faktów, reguł i decyzji*, Akademicka Oficyna Wydawnicza EXIT, Warszawa 2001.

[122] Łęski J., *Systemy neuronowo-rozmyte*, WNT, Warszawa 2008.

[123] Łukasiewicz J., *Elementy logiki matematycznej*, 2 wyd., PWN, Warszawa 1958.

[124] Mamdani E.H., *Application of fuzzy algoritms for the control of a simple dynamic plant*, "IEE Proceedings" 1974, Vol. 121, No. 12, pp. 1585–1588.

[125] Mateou N.H., Moiseos M., Andreou A.S., *Multi-objective evolutionary fuzzy cognitive maps for decision support* [in:] Proc. of the IEEE Congress on Evolutionary Computation, Edinburgh (Scotland) 2005, pp. 824–830.

[126] Mateou N.H., Stylianou C., Andreou A.S., *Hybrid Fuzzy Cognitive Map Modeller: A Novel Software Tool for Decision Making* [in:] Proc. of the Fourth IEEE International Workshop on Soft Computing as Transdisciplinary Science and Technology WSTST '05, Muroran (Japan) 2005, pp. 851–862.

[127] McCulloch W.S., Pitts W., *A logical calculus of the ideas immanent in neurons activity*, "Bull. Math. Biophys." 1943, Vol. 5, pp. 115–133.

[128] Miao Y., Liu Z.-Q., *On causal inference in fuzzy cognitive maps*, "IEEE Transactions on Fuzzy Systems" 2000, Vol. 8, No. 1, pp. 107–119.

[129] Michalewicz Z., *Algorytmy genetyczne + struktury danych = programy ewolucyjne*, WNT, Warszawa 1999.

[130] Miller G.A., *The magical number seven, plus or minus two: Some limits on our capacity for processing information*, "Psychological Review" 1956, Vol. 63, No. 2, pp. 343–355.

[131] Montazemi A.R., Conrath D.W., *The use of cognitive mapping for information requirement analysis*, "MIS Quarterly" 1986, Vol. 10, pp. 45–55.

[132] Moscato P., *On evolution, Search, Optimization, Genetic Algorithms and Martial Arts: Toward Memetic Algorithms*, "California Institute of Technology", Caltech Concurrent Computation Program, Technical Report 826, Pasadena-CA (USA) 1989.

[133] Negoita C.V., Ralescu D.A., *Application of Fuzzy Sets to System Analysis*, Birkhaüser, Basel and Halstead Press, New York (USA) 1975.

[134] Nguyen H.T., Kreinovich V., *Computing Degrees of Subsethood and Similarity for Interval-Valued Fuzzy Sets: Fast Algorithms* [in:] Proceedings of the 9th International Conference on Intelligent Technologies InTech'08, Samui (Thailand) 2008, pp. 47–55.

[135] Oja E., Ogawa H., Wangviwattana J., *Learning in Nonlinear Constrained Hebbian Networks* [in:] Proc. of the International Conference on Artificial Neural Networks ICANN'91, Vol. 1, Espoo (Finland) 1991, pp. 385–390.

[136] Papageorgiou E.I., *Learning Algorithms for Fuzzy Cognitive Maps – A Review Study*, "IEEE Transactions on Systems, Man and Cybernetics, Part C: Applications and Reviews" 2012, Vol. 42, No. 2, pp. 150–163.

[137] Papageorgiou E.I., Froelich W., *Multi-step prediction of pulmonary infection with the use of evolutionary fuzzy cognitive maps*, "Neurocomputing" 2012, No. 92/2012, pp. 28–35.

[138] Papageorgiou E.I., Groumpos P.P., *A new hybrid method using evolutionary algorithms to train Fuzzy Cognitive Maps*, "Applied Soft Computing" 2005, Vol. 5, No. 4, pp. 409–431.

[139] Papageorgiou E.I., Groumpos P.P., *A weight adaptation method for fuzzy cognitive map learning*, "Soft Computing" 2005, Vol. 9, No. 11, pp. 846–857.

[140] Papageorgiou E.I., Parsopoulos K.E., Stylios C.S., Groumpos P.P., Vrahatis M.N., *Fuzzy Cognitive Maps Learning Using Particle Swarm Optimization*, "Journal of Intelligent Information Systems" 2005, Vol. 25, No. 1, pp. 95–121.

[141] Papageorgiou E.I., Stylios C.D., *Fuzzy Cognitive Maps* [in:] *Handbook of Granular Computing*, Pedrycz W., Skowron A., Kreinovich V., ch. 34, John Wiley & Son Ltd, Publication Atrium, Chichester (England) 2008, pp. 755–774.

[142] Papageorgiou E.I., Stylios C.D., Groumpos P.P., *Active Hebbian learning algorithm to train fuzzy cognitive maps*, "International Journal of Approximate Reasoning", No. 37, 2004, pp. 219–249.

[143] Papageorgiou E., Stylios C.D., Groumpos P.P., *Fuzzy Cognitive Map Learning Based on Nonlinear Hebbian Rule* [in:] "*Lecture Notes in Artificial Intelligence – AI 2003*", Gedeon T.D., Fung L.C.C., Vol. 2903, Springer-Verlag, Berlin-Heidelberg (Germany) 2003, pp. 254–268.

[144] Parsopoulos K.E., Papageorgiou E.I., Groumpos P.P., Vrahatis M.N., *A First Study of Fuzzy Cognitive Maps Learning Using Particle Swarm Optimization* [in:] IEEE Congress on Evolutionary Computation, CEC 2003, Vol. 2, Canberra (Australia) 2003, pp. 1440–1447.

[145] Pedrycz W., *An approach to the analysis of fuzzy systems,* "International Journal of Control" 1981, Vol. 34, pp. 403–421.

[146] Pedrycz W., *Fuzzy Control and Fuzzy Systems*, John Wiley and Sons, New York (USA) 1993.

[147] Peláez C.A., Bowles J.B., *Using fuzzy cognitive maps as a system model for failure modes and effects analysis*, "Information Sciences" 1996, Vol. 88, pp. 177–199.

[148] Petalas Y.G., Papageorgiou E.I., Parsopoulos K.E., Groumpos P.P., Vrahatis M.N., *Fuzzy Cognitive Maps Learning using Memetic Algorithms* [in:] Proc. of the International Conference of "Computational Methods in Sciences and Engineering" (ICCMSE 2005), Loutraki (Greece) 2005, pp. 1420–1423.

[149] Petalas Y.G., Parsopoulos K.E., Vrahatis M.N., *Improving fuzzy cognitive maps learning through memetic particle swarm optimization*, "Soft Computing", Vol. 13, pp. 77–94, 2009.

[150] Pham D.T., Karaboga D., *Intelligent Optimisation Techniques: Genetic Algorithms, Tabu Search, Simulated Annealing and Neural Networks*, Springer, New York (USA) 2000.

[151] Piegat A., *Fuzzy Modelling and Control*, Physica-Verlag, Springer-Verlag Company, Heidelberg (Germany) 2001.

[152] Piegat A., *Modelowanie i sterowanie rozmyte*, Akademicka Oficyna Wydawnicza EXIT, Warszawa 1999.

[153] Prensky M., *Digital Game-Based Learning*, Paragon House Publishers, St. Paul-MN (USA) 2007.

[154] Ralescu D.A., *Cardinality, quantifiers, and the aggregation of fuzzy criteria*, "Fuzzy Sets and Systems" 1995, Vol. 69, pp. 355–365.

[155] Ross T.J., *Fuzzy Logic with Engineering Applications*, 3rd ed., John Wiley & Sons Ltd., Chichester (UK) 2010.

[156] Ross K.A., Writh C.R.B., *Matematyka dyskretna*, Wydawnictwo Naukowe PWN, Warszawa 2008.

[157] Russell S.J., Norvig P., *Artificial Intelligence: A Modern Approach*, 3rd ed. Upper Saddle River, Prentice Hall, Pearson Education Inc., New Jersey (USA) 2010.

[158] Rutkowska D., Piliński M., Rutkowski L., *Sieci neuronowe, algorytmy genetyczne i systemy rozmyte*, PWN, Warszawa 1997.

[159] Rutkowski L., *Computational Intelligence. Methods and Techniques*, Springer-Verlag, Berlin-Heidelberg 2008.

[160] Rutkowski L., *Metody i techniki sztucznej inteligencji*, PWN, Warszawa 2005.

[161] Rutkowski L., Cpalka K., *Designing and learning of adjustable quasi-triangular norms with applications to neuro-fuzzy systems*, "IEEE Transactions on Fuzzy Systems" 2005,Vol. 13, No. 1, pp. 140–151.

[162] Rutkowski L., Cpałka K., *Flexible neuro-fuzzy systems*, "IEEE Transactions on Neural Networks" 2003, Vol. 14, No. 3, pp. 554–574.

[163] Salmeron J.L., *Supporting Decision Makers with Fuzzy Cognitive Maps*, "Research Technology Management" 2009, Vol. 52, No. 3, pp. 53–59.

[164] Satur R., Liu Z.Q., *A contextual fuzzy cognitive map framework for geographic information systems*, "IEEE Trans. Fuzzy Syst". 1999, Vol. 7, pp. 481–494.

[165] Schneider M., Shnaider E., Kandel A., Chew G., *Automatic construction of FCMs*, "Fuzzy Sets and Systems" 1998, Vol. 93, No. 2, pp. 161–172.

[166] Silov V.B., *Podejmowanie strategicznych decyzji w rozmytym otoczeniu: w makroekonomii, polityce, socjologii, zarządzaniu, ekologii, medycynie*, INPRO-RES, Moskwa (Rosja) 1995 (w języku rosyjskim).

[167] Siraj A., Bridges S.M., Vaughn R.B., *Fuzzy Cognitive Maps for Decision Support in an Intelligent Intrusion Detection System*, [in:] IFSA World Congress and 20th NAFIPS International Conference, Vancouver (Canada) 2001, pp. 2165–2170.

[168] Słoń G., *Adaptacja relacji w dynamicznych rozmytych relacyjnych mapach kognitywnych*, „Logistyka" 2010, Vol. 6/2012, s. 3061–3069.

[169] Słoń G., *Analiza wybranych algorytmów adaptacji relacji w rozmytych mapach kognitywnych*, „Pomiary, Automatyka, Kontrola" 2010, Vol. 56, nr 12, s. 1445–1448.

[170] Słoń G., *The Use of Fuzzy Numbers in the Process of Designing Relational Fuzzy Cognitive Maps* [in:] *Artificial Intelligence and Soft Computing - Lecture Notes in Artificial Intelligence LNAI 7894,* 12th International Conference ICAISC 2013, Zakopane 2013, Rutkowski L. et al., 7894/Part 1, Springer-Verlag, Heidelberg–Dordrecht–London–New York 2013, Vol., pp. 376–387.

[171] Słoń G., *Wybrane problemy projektowania rozmytych relacyjnych map kognitywnych* [w:] *Technologie komputerowe w rozwoju nauki, techniki i edukacji*, pod red. Jastriebow A., Kuźmińska-Sołśnia B., Raczyńska M., Wydawnictwo Naukowe Instytutu Technologii Eksploatacji – Państwowego Instytutu Badawczego, Radom 2012, s. 131–144.

[172] Słoń G., Gad S., *System monitorowania decyzyjnego stanu pojazdu oparty na relacyjnej mapie kognitywnej* [w:] *Technologie komputerowe w rozwoju nauki, techniki i edukacji*, pod red. Jastriebow A., Kuźmińska-Sołśnia B., Raczyńska M., Wydawnictwo Naukowe Instytutu Technologii Eksploatacji – Państwowego Instytutu Badawczego, Radom 2012, s. 145–156.

[173] Słoń G., Gad S., Jastriebow A., *Wykorzystanie sztucznych sieci neuronowych do diagnozowania elementów wyposażenia pojazdów samochodowych*, Zeszyty Naukowe Politechniki Świętokrzyskiej „Elektryka" 2004, nr 41, pod red. Nadolski R., Wyd. Politechniki Świętokrzyskiej, Kielce 2004, s. 125–135.

[174] Słoń G., Yastrebov A., *Application of Fuzzy Relational Cognitive Maps in Intelligent Modelling the Technical Systems*, "Poznan University of Technology Academic Journals. Electrical Engineering" 2012, No. 69, pp. 175–182.

[175] Słoń G., Yastrebov A., *Designing and Training Relational Fuzzy Cognitive Maps* [in:] *Intelligent Systems Reference Library – Fuzzy Cognitive Maps for Applied Sciences and Engineering*, ed. Papageorgiou E., Vol. 54, Springer, 2013, 13 p.

[176] Słoń G., Yastrebov A., *Optimization and Adaptation of Dynamic Models of Fuzzy Relational Cognitive Maps* [in:] *Lecture Notes in Artificial Intelligence 6743*, RSFDGrC 2011, ed. Kuznetsov S.O., Springer-Verlag, Berlin-Heidelberg (Germany) 2011, pp. 95–102.

[177] Solis F.T., Wets R.J.-B., *Minimization by Random Search Techniques*, "*Mathematics of Operations Research*" 1981, Vol. 6, No. 1, pp. 19–30.

[178] Stach W., Kurgan L., Pedrycz W., *A divide and conquer method for learning large Fuzzy Cognitive Maps*, "Fuzzy Sets and Systems" 2010, No. 161/2010, pp. 2515–2532.

[179] Stach W., Kurgan L.A., Pedrycz W., *A Survey of Fuzzy Cognitive Map Learning Methods* [in:] *Issues in Soft Computing: Theory and Applications*, ed. Grzegorzewski P., Krawczak M., Zadrożny S., Akademicka Oficyna Wydawnicza EXIT, Warszawa 2005, pp. 71–84.

[180] Stach W., Kurgan L., Pedrycz W., *Data-driven Nonlinear Hebbian Learning method for Fuzzy Cognitive Maps* [in:] IEEE World Congress on Computational Intelligence FUZZ-IEEE 2008, Hong Kong (China) 2008, pp. 1975–1981.

[181] Stach W., Kurgan L.A., Pedrycz W., *Numerical and Linguistic Prediction of Time Series With the Use of Fuzzy Cognitive Maps*, "IEEE Transactions on Fuzzy Systems" 2008, Vol. 16, No. 1, pp. 61–72.

[182] Stach W., Kurgan L.A., Pedrycz W., *Parallel Learning of Large Fuzzy Cognitive Maps*, [in:] Proc. of International Joint Conference on Neural Networks, Orlando-Florida (USA) 2007, pp. 1584–1889.

[183] Stach W., Kurgan L., Pedrycz W., Reformat M., *Evolutionary Development of Fuzzy Cognitive Maps* [in:] Proc. of he 14th IEEE International Conference on Fuzzy Systems FUZZ '05, Reno (NV) USA 2005, pp. 619–624.

[184] Stach W., Kurgan L., Pedrycz W., Reformat M., *Genetic Learning of Fuzzy Cognitive Maps*, "Fuzzy Sets and Systems" 2005, Vol. 153, pp. 371–401.

[185] Storn R., Price K., *Differential Evolution – A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces*, "Journal of Global Optimization" 1997, Vol. 11, pp. 341–359.

[186] Styblinski M.A., Meyer B.D., *Fuzzy cognitive maps signal flow graphs and qualitative circuit analysis* [in:] Proc. of the 2nd IEEE International Conference on Neural Network (ICNN) 1987, San Diego-California (USA) 1988, pp. 549–556.

[187] Stylios C.D., Georgopoulos V., Groumpos P.P., *Fuzzy cognitive map approach to process control systems*, "Journal of Advanced Computational Intelligence" 1999, Vol. 3, No. 5, pp. 409–417.

[188] Stylios C.D., Georgopoulos V.C., Groumpos P.P., *The Use of Fuzzy Cognitive Maps in Modeling Systems* [in:] Proc. of 5th IEEE Mediterranean Conference on Control and Systems, Paphos, 1997, Paper No. 67.

[189] Stylios C.D., Groumpos P.P., *Fuzzy cognitive maps in modeling supervisory control systems*, "Journal of Intelligent & Fuzzy Systems" 2000, Vol. 8, No. 2, pp. 83–98.

[190] Stylios C.D., Groumpos P.P., *The challenge of modeling supervisory systems using fuzzy cognitive maps*, "Journal of Intelligent Manufacturing" 1998, Vol. 9, No. 4, pp. 339–345.

[191] Taber W.R., Siegel M.A., *Estimation of Expert Weights Using Fuzzy Cognitive Maps*, [in:] Proc. of the First International Conference on Neural Networks, Vol. 2, San Diego-CA (USA) 1987, pp. 319–325.

[192] Tadeusiewicz R., *Sieci neuronowe*, 2 wyd., Akademicka Oficyna Wydawnicza RM, Warszawa 1993.

[193] Takagi H., Sugeno M., *Fuzzy Identification of Systems and Its Application to Modeling and Control*, "IEEE Transactions on Systems, Man and Cybernetics SMC-15" 1985, No. 1, pp. 116–132.

[194] Tolman E.C., *Cognitive maps in rats and men*, "Psychological Review", Vol. 42, No. 55, 1948, pp. 189–208.

[195] Tsadiras A.K., Margaritis K.G., *Cognitive Mapping and Certainty Neuron Fuzzy Cognitive Maps*, "Information Sciences" 1997, Vol. 101, No. 1–2, pp. 109-130.

[196] Urząd Transportu Kolejowego, *Funkcjonowanie rynku transportu kolejowego w Polsce w 2011 roku*, raport wrzesień, Warszawa 2012.

[197] Veloso M.M., *Planning and Learning by Analogical Reasoning*, Springer-Verlag, Berlin-Heidelberg (Germany) 1994.

[198] Wang K., *Intelligent Condition Monitoring and Diagnosis Systems: A Computational Intelligence Approach*, IOS Press, Amsterdam (Holandia) 2003.

[199] Wasserman P.D., *Neural Computing: Theory and Practice*, Van Nostrand Reinhold, New York (USA) 1989.

[200] Weber S., *A generat concept of fuzzy connectives, negations and implications based on t-norms and t-conorms*, "Fuzzy Sets and Systems" 1983, Vol. 11, No. 1–3, pp. 103–113.

[201] Wellman M.P., *Inference in cognitive maps*, "Mathematics and Computers in Simulation" 1994, Vol. 36, pp. 137–148,.

[202] White H., *Artificial Neural Networks: Approximation and Learning Theory*, Blackwell Publishers Inc., Cambridge-MA (USA) 1992.

[203] Wierzchoń S.T., *Sztuczne systemy immunologiczne. Teoria i zastosowania*, Akademicka Oficyna Wydawnicza Exit, Warszawa 2001.

[204] Worwa K., *Using an optimization methods to selection the best software developer*, „Symulacja w Badaniach i Rozwoju" 2011, Vol. 2, No. 2, pp. 113–123.

[205] Worwa K., Stanik J., *Quality modelling for Web-based information systems*, Biuletyn Instytutu Systemów Informatycznych, No. 7, 2011, pp. 77–86.

[206] Worwa K., Stanik J., *Quality of Web-Based Information Systems*, "Journal of Internet Banking and Commerce" 2010, Vol. 15, No. 3, pp. 169–185.

[207] Xiao Z., Chen W., Li L., *An integrated FCM and fuzzy soft set for supplier selection problem based on risk evaluation*, "Applied Mathematical Modelling" 2012, No. 36/2012, pp. 1444–1454,.

[208] Yager R.R., *On a general class of fuzzy connectives*, "Fuzzy Sets and Systems" 1980, Vol. 4, No. 3, pp. 235–242.

[209] Yager R.R., Filev D.P., *Podstawy modelowania i sterowania rozmytego*, WNT, Warszawa 1995.

[210] Yaman D., Polat S., *A fuzzy cognitive map approach for effect-based operations: An illustrative case*, "Information Sciences" 2009, Vol. 179, No. 4, pp. 382–403.

[211] Yastrebov A., Gad S., Słoń G., *Artificial Neural Networks in Diagnostic of Motorcar Vehicles' Electrical Equipment* [in:] Proc. of International conference on signals and electronic systems ICSES 2006, Vol. 2, Łódź 2006, pp. 781–784.

[212] Yastrebov A., Gad S., Słoń G., *Bank of Artificial Neural Networks MLP Type in Symptom System of Technical Diagnostics*, "Polish Journal of Environmental Studies" 2008, Vol. 17, No. 2A, pp. 118–123.

[213] Yastrebov A., Gad S., Słoń G., Gad R., *Symptom Computer Diagnostics of Automotive Vehicles' Electrical Equipment* [in:] System Modelling Control 2005. Ed., AO EXIT Warszawa 2005, Zakopane 2005, pp. 327–332.

[214] Yastrebov A., Gad S., Słoń G., Kułakowski A., Vinogradowa E., Artificial Neural Networks in Rule-based Decision Systems [in:] Proc. of International Conference on Computer as tool EUROCON'2007, IEEE Xplore 1-4244-0813-X/07, Warszawa 2007, pp. 686–691.

[215] Yastrebov A., Gad S., Słoń G., Łaskawski M., *Computer Analysis of Electrical Vehicle Equipment Diagnosing with Artificial Neural Networks* [in:] Proc. of the II

International Conference "System Identification and Control Problems", Moskwa 2003, pp. 1331–1348.

[216] Yastrebov A., Gad S., Słoń G., Łaskawski M., *Intelligent methods of ANN type in symptom diagnostic of motorcar vehicles' electrical equipment*, „Diagnostyka", No. 1(37)/2006, pp. 69–76.

[217] Yastrebov A., Gad S., Słoń G., Pietrasik L., *Analysis of computer intelligent diagnostic models in automotive vehicle's electrical equipment* [in:] Proc. of the 15th International Conference on Systems Science, Systems Science XV, Vol. III, Wrocław 2004, pp. 373–380.

[218] Yastrebov A., Słoń G., *Optimization of models of fuzzy relational cognitive maps* [in:] *Computers in scientific and educational activity*, pod red. Jastriebow A., Raczyńska M., Institute for Sustainable Technologies – National Research Institute, Radom 2011, pp. 60–71.

[219] Zadeh L.A., *A computational approach to fuzzy quantifiers in natural languages*, "Computers and Mathematics with Applications" 1983, Vol. 9, pp. 149–184.

[220] Zadeh L.A., *A rationale for fuzzy control*, "Journal of Dynamic Systems. Measurement and Control" 1972, Vol. 34, pp. 3–4.

[221] Zadeh L.A., *Fuzzy sets and systems* [in:] *System Theory. Microwave Research Institute Symposia Series XV*, Fox J., Polytechnic Press, Brooklyn-NY (USA) 1965, pp. 29–37.

[222] Zadeh L.A., *Fuzzy Sets*, "Information and Control" 1965, Vol. 8, pp. 338–353.

[223] Zadeh L.A., *Outline of a New Approach to the Analysis of Complex Systems and Decision Processes*, "IEEE Transactions on Systems. Man and Cybernetics" 1973, Vol. SMC-3, No. 1, pp. 28–44.

[224] Zadeh L.A., *The role of fuzzy logic in the management of uncertainty in expert systems*, "Fuzzy Sets and Systems" 1983, Vol. 11, No. 1–3, pp. 199–228.

[225] Zhang W.R., Chen S.S., *A logical architecture for cognitive maps* [in:] Proc. of the 2nd IEEE International Conference on Neural Networks (ICNN-88), Vol. 1, 1988, pp. 231–238.

[226] Zhang W.R., Chen S.S., Wang W., King R.S., *A cognitive-map-based approach to the coordination of distributed cooperative agents*, "IEEE Trans. Systems Man Cybernet"., Vol. 22, 1992, pp. 103–114.

[227] Zhou S., Liu Z.-Q., Zhang J.Y., *Fuzzy causal networks: General model, inference, and convergence*, "IEEE Transactions on Fuzzy Systems" 2006, Vol. 14, No. 3, pp. 412–420.

[228] Zhu Y., Zhang W., *An Integrated Framework for Learning Fuzzy Cognitive Map using RCGA and NHL Algorithm* [in:] Proc. of the 4th International Conference on Wireless Communications, Networking and Mobile Computing WiCOM'08, Dalian (China) 2008, pp. 1–5.

RELACYJNE ROZMYTE MAPY KOGNITYWNE W MODELOWANIU ZŁOŻONYCH SYSTEMÓW

Streszczenie

W monografii przedstawiono rozważania i wyniki badań nad nową metodą inteligentnego modelowania systemów charakteryzujących się niepewnością i nieprecyzyjnością informacji. Systemy takie trudno modelować w klasyczny sposób z powodu niedostatku danych o ich strukturze i zachowaniu, trzeba więc stosować podejścia, które te niedostatki skompensują. Należy do nich m.in. podejście wykorzystujące model tzw. rozmytej mapy kognitywnej, gdzie system rozpatruje się jako graf zorientowany, którego węzłami są kluczowe, wybrane przez eksperta parametry (zwane czynnikami), a łuki odwzorowują powiązania przyczynowo-skutkowe pomiędzy tymi wielkościami. Dodatkowo nieprecyzyjność danych uwzględnia się, przedstawiając wartości poszczególnych wielkości przy użyciu zbiorów rozmytych. Podejście to cechuje szereg niedogodności związanych głównie z procesami uczenia, do którego konieczne jest odejście od rozmytego opisu zarówno samych czynników, jak i powiązań pomiędzy nimi, co z kolei poddaje w wątpliwość sens wcześniejszego rozmywania danych. W monografii przedstawiono dwa alternatywne sposoby pokonania tych trudności. Pierwszy, z wykorzystaniem tzw. relacyjnej mapy kognitywnej, polega na całkowitym odejściu od rozmywania na rzecz posługiwania się wyłącznie liczbowymi wartościami czynników i powiązań pomiędzy nimi (zwanych tu relacjami). Drugi, główny, zaproponowany w monografii sposób modelowania, oparty na tzw. relacyjnej rozmytej mapie kognitywnej, stanowi rozwinięcie tego podejścia o rozmywanie parametrów. W nowym modelu wartości czynników przedstawia się za pomocą liczb rozmytych, a powiązania pomiędzy nimi – przy użyciu relacji rozmytych. Takie potraktowanie problemu pozwala na zachowanie rozmycia parametrów na każdym etapie projektowania i eksploatacji modelu, wyzwalając jednakże szereg, nierozwiązanych wcześniej, problemów związanych z metodami rozmywania, projektowaniem kształtów relacji, normalizacją przetwarzanych wartości, wykorzystaniem arytmetyki liczb rozmytych oraz uczeniem tak zdefiniowanego modelu. W monografii przedstawiono rozwiązania tych problemów oraz przykłady ilustrujące praktyczne działanie opracowanego podejścia.

RELATIONAL FUZZY COGNITIVE MAPS IN COMPLEX SYSTEMS MODELING

Summary

The monograph presents the considerations and results of the research into new method of modeling intelligent systems characterized by uncertainty and imprecision of the information. Such systems are difficult to classical modeling due to a lack of information about their structure and behavior, thus it is needed using an approach that can compensate these weaknesses. These include, among others, an approach using the model of so-called fuzzy cognitive map, where the system is considered as a directed graph, nodes of which are crucial, selected by an expert, parameters (called concepts), and arcs reproduce causal relationships between these quantities. In addition, imprecision of the data is taken into account by presenting the individual values using fuzzy sets. This approach has a number of disadvantages associated primarily with the processes of learning to which it is necessary to move away from the fuzzy description of concepts as well as connections between them, which in turn calls into question the meaning of prior fuzzyfication of the data. The monograph presents two alternative ways to overcome these difficulties. The first, with the use of the so-called relational cognitive maps, depends on a complete departure from fuzzifyication to the use of only numerical values of the concepts and the connections between them (called here relations). The second and main, proposed in the monograph modeling method, based on so-called relational fuzzy cognitive map, is a development of this approach by fuzzyfying parameters. In the new model, the concepts are presented with the use of fuzzy numbers, and the connections between them – using fuzzy relations. Such a treatment of the problem allows to keep the fuzzy parameters at every stage of the design and operation of the model, however, it triggers a number, not solved earlier, problems connected with methods of fuzzyfication, designing shapes of relations, normalization of processed values, using fuzzy arithmetic and learning such defined model. The monograph presents solutions to these problems and examples illustrating practical action of developed approach