

Politechnika Śląska
Wydział Mechaniczny Technologiczny
Katedra Podstaw Konstrukcji Maszyn

Szkieletowy system doradczy MMNET

Wojciech CHOLEWA (red.)



Redaktor zeszytów
Wojciech Cholewa

Redaktorzy techniczni
Paweł Chrzanowski, Marek Wyleźoł

Projekt okładki
Wojciech Cholewa, Marek Wyleźoł

BibTeX:
@BOOK{
title = {Szkieletowy system doradczy {MMNET}},
publisher = {Politechnika Śląska, Katedra Podstaw Konstrukcji Maszyn},
year = {2010},
author = {Cholewa, W. (red.)},
volume = {143},
series = {Zeszyty},
address = {Gliwice},
edition = {1} }

Praca finansowana częściowo ze środków na naukę w latach 2008-2010,
jako fragment projektu badawczego nr N N504 478434

ISBN 978-83-60759-18-9

Wydawca
Katedra Podstaw Konstrukcji Maszyn
Wydział Mechaniczny Technologiczny
Politechnika Śląska
ul. Konarskiego 18a, 44-100 Gliwice
tel. (32) 237-14-67, fax (32) 237-13-60
<http://kpkkm.polsl.pl>

Druk i oprawa (z dostarczonych materiałów):
Drukarnia cyfrowa D&D sp. z o.o. ul. Moniuszki 6, 44-100 Gliwice

Gliwice, 2010

Spis treści

1	Wstęp	1
1.1	Zakres monografii	2
1.2	Oznaczenia i skróty	2
2	Pojęcia podstawowe	3
2.1	Stwierdzenia	3
2.2	Stwierdzenia przybliżone	5
2.3	Działania na stwierdzeniach przybliżonych	6
2.4	Sieci stwierdzeń	6
2.5	Stwierdzenia pierwotne i wtórne	7
	Bibliografia	8
3	Wielomodalne sieci stwierdzeń	9
3.1	Wstęp	9
3.2	Warunki konieczne i dostateczne	10
3.3	Sieci stwierdzeń i ich rozwiązywanie	11
3.3.1	Przykłady	12
3.3.2	Znane dokładne wartości wszystkich stwierdzeń	13
3.3.3	Znane dokładne wartości wybranych stwierdzeń	13
3.3.4	Znane przybliżone wartości wybranych stwierdzeń	13
3.3.5	Ogólny algorytm rozwiązywania sieci	14
3.4	Przybliżone warunki konieczne i dostateczne	15
3.5	Przybliżone sieci stwierdzeń	16
3.5.1	Elementy sprzeczne	17
3.5.2	Maskowanie gałęzi sieci	18
3.6	Rozwiązywanie przybliżonych sieci stwierdzeń	18
3.6.1	Zadanie programowania liniowego	18
3.6.2	Algorytm rozwiązywania przybliżonej sieci stwierdzeń - część I	19
3.6.3	Algorytm rozwiązywania przybliżonej sieci stwierdzeń - część II	22
3.6.4	Elementy warunkowo sprzeczne	24
	Bibliografia	24
4	Wybór środowiska oprogramowania	25
4.1	Wstęp	25
4.2	Zestawienie i weryfikacja dostępnych funkcji i procedur	25
4.2.1	Składowanie sieci	26
4.2.2	Transformacja formatów zapisu sieci stwierdzeń	27
4.2.3	Edycja sieci stwierdzeń	27
4.2.4	Wizualizacja sieci	30
4.2.5	Rozwiązywanie sieci	31

4.2.6	Trenowanie struktury sieci	33
4.2.7	Trenowanie parametrów sieci	34
4.2.8	Narzędzia do analizy sieci	35
4.2.9	Inne: Generowanie przykładów	35
4.2.10	Inne: Dynamiczne sieci przekonañ	36
4.2.11	Podsumowanie	36
4.3	Wybór języka programowania	37
	Bibliografia	38
5	Pakiet MMNET	41
5.1	Opis ogólny	41
5.2	Wymiana danych w formacie XML	42
5.3	Wymagania	44
5.4	Budowanie pakietu	44
5.5	Numerowanie wersji	45
5.6	Instalacja i pierwsze uruchomienie	46
5.7	Aktualizacja pakietu	46
5.8	Stosowanie pakietu MMNET bez GUI	46
	Bibliografia	48
6	Interfejs użytkownika systemu MMNET	49
6.1	Wstęp	49
6.2	Menu oraz zakładka główna	49
6.3	Definiowanie nowego tezaury	51
6.4	Definiowanie stwierdzeń prostych	51
6.5	Przegląd oraz edycja istniejących stwierdzeń prostych	53
6.6	Definiowanie stwierdzeń złożonych	54
6.7	Edycja oraz przegląd istniejących stwierdzeń złożonych	54
6.8	Definiowanie nowego modelu	56
6.9	Edycja modelu	56
6.10	Definiowanie nowych warstw	56
6.11	Edycja istniejących warstw	57
6.12	Przeglądanie warstw	58
6.13	Wykonywanie obliczeń	59
6.14	Zapis i odczyt danych z pliku	62
6.15	Eksport wprowadzonych danych do pliku TXT	63
6.16	Przykład zastosowania interfejsu	64
	Bibliografia	65
7	Biblioteka modeli MMNET	67
7.1	Model z zastosowaniem sieci przekonañ	67
7.1.1	Podsumowanie	68
7.2	Model z zastosowaniem sieci umożliwiających stosowanie analogii mechanicznej	69
7.2.1	Podsumowanie	70
	Bibliografia	70
8	Baza przykładów sieci stwierdzeń	71
8.1	Wstęp	71
8.2	Struktura opracowanej bazy przykładów	71
8.3	Przeszukiwanie bazy przykładów	73
8.4	Ocena przydatności zgromadzonych przykładów	74

8.4.1	Określenie relacji pomiędzy kryteriami	74
8.4.2	Skala ocen	74
8.4.3	Oceny zgromadzonych przykładów	75
8.5	Zastosowanie zgromadzonych przykładów w systemie MMNET	76
8.5.1	Przykład 1	77
8.5.2	Przykład 2	80
	Bibliografia	82
9	Porównanie wybranych modeli sieci stwierżeń	83
9.1	Struktury grafów i warunkowa niezależność pomiędzy zmiennymi	83
9.2	Porównanie wyników dla różnych stanów sieci	84
9.3	Podsumowanie	86
	Bibliografia	86
10	Przykład sieci przekonań wspomagającej proces koncygowania	89
10.1	Założenia dotyczące opracowywanych przykładów	89
10.2	Przyjęta metoda doboru elementów kompletowanego przenośnika	89
10.2.1	Słownik stwierżeń	90
10.2.2	Postaci sieci	93
10.3	Przykłady użycia	95
10.4	Podsumowanie i wnioski	107
	Bibliografia	108
11	Przykład sieci stwierżeń wspomagającej ocenianie postępów studentów	109
11.1	Ocena postępów studenta	109
11.2	Założenia do opracowanego przykładu	110
11.3	Struktura opracowanych sieci	111
11.3.1	Stwierżenia z grupy zagadnień teoretycznych	111
11.3.2	Stwierżenia z grupy zagadnień praktycznych	114
11.3.3	Warstwy sieci stwierżeń	118
11.4	Przykłady zastosowania	120
11.5	Wnioski	127
	Bibliografia	127
12	Przykład sieci stwierżeń wspomagającej projektowanie układów	129
12.1	Wybrane podejścia do projektowania układów napędowych	129
12.1.1	Standardowa metoda doboru elementów napędu	129
12.1.2	Dobór elementów napędu ze względu na precyzyjne sterowanie położeniem kątowym	130
12.1.3	Dobór elementów napędu ze względu na precyzyjne sterowanie prędkością kątową	130
12.1.4	Dobór elementów napędu ze względu na bezpieczeństwo użytkownika	130
12.2	Lista stwierżeń	131
12.3	Postacie opracowanych sieci stwierżeń	134
12.4	Przykłady użycia	137
12.4.1	Standardowa metoda doboru elementów napędu	137
12.4.2	Dobór elementów napędu ze względu na precyzyjne sterowanie położeniem kątowym	139
12.4.3	Dobór elementów napędu ze względu na precyzyjne sterowanie prędkością kątową	143
12.4.4	Dobór elementów napędu ze względu na bezpieczeństwo użytkownika	147

12.5 Podsumowanie	147
Bibliografia	149
13 Zastosowanie wieloskalowej sieci stwierżeń w diagnostycznym systemie doradczym	151
13.1 Przykład wieloskalowego modelu diagnostycznego	151
13.1.1 Model globalny	151
13.1.2 Model cząstkowy - diagnozowanie stanu rozrusznika	153
13.1.3 Model cząstkowy - diagnozowanie stanu akumulatora	154
13.1.4 Domyślny stan sieci przekonań	155
13.1.5 Zadany stan sieci wieloskalowej	156
13.2 Podsumowanie	157
Bibliografia	157
14 Diagnostyczny system doradczy dla symulatora procesu	159
14.1 Rozpatrywany proces	159
14.2 Zbiór rozpatrywanych stanów technicznych obiektu	162
14.3 Zbiór rozpatrywanych sygnałów diagnostycznych	163
14.4 Identyfikacja warstw modelu	164
14.5 Identyfikacja zbioru cech sygnałów diagnostycznych	165
14.6 Przygotowanie zbioru danych	168
14.7 Modele diagnostyczne dla kolejnych warstw sytemu	168
14.7.1 Słownik stwierżeń	168
14.7.2 Określenie struktur sieci stwierżeń	168
14.7.3 Określenie wartości prawdopodobieństw warunkowych	171
14.7.4 Określenie wartości wag modeli	171
14.8 Weryfikacja systemu	173
14.9 Wnioski	177
Bibliografia	177

Rozdział 1

Wstęp

Wojciech CHOLEWA

Monografia omawia wyniki realizacji kolejnego etapu badań związanych z zastosowaniem metod i technik sztucznej inteligencji w diagnostyce technicznej. Badania te dotyczyły możliwości budowania diagnostycznych systemów doradczych z zastosowaniem wieloskalowych sieci stwierdzeń. Procesy wnioskowania o stanach technicznych obiektów i ich zmianach mogą być wspomagane różnymi sformalizowanymi systemami podejmowania decyzji. Szczególne znaczenie mają systemy doradcze, nazywane również systemami ekspertowymi. We wcześniej prowadzonych badaniach rozpatrywano różne sposoby reprezentowania wiedzy w takich systemach. Stwierdzono konieczność stosowania metod umożliwiających wnioskowanie na podstawie wiedzy niedokładnej i niepewnej, a czasami nawet sprzecznej.

Można wymieniać liczne zalety systemów, w których baza wiedzy występuje w postaci sieci przekonań (tzw. sieci bayesowskich). Ciekawym rozwiązaniem jest interpretowanie tych sieci jako sieci stwierdzeń. Elementami tych sieci są stwierdzenia rozpatrywane jako pary składające się z treści stwierdzenia i wartości stwierdzenia. Treść stwierdzenia jest wypowiedzią orzekającą o obserwowanych faktach lub reprezentującą określoną opinię. Wartość stwierdzenia jest definiowana jako stopień prawdziwości lub stopień przekonania o prawdziwości wypowiedzi będącej treścią stwierdzenia. Sieci stwierdzeń pozwalają na wprowadzanie złożonych systemów pomocy, zawierających objaśnienia pojęć, odsyłacze do źródeł i różne komentarze. Główną niedogodnością sieci stwierdzeń rozpatrywanych jako modyfikacja sieci przekonań jest to, że podczas ich budowania, identyfikacji oraz doskonalenia istnieje potrzeba rozpatrywania stale jednej sieci globalnej, określonej dla pełnego zbioru uwzględnianych stwierdzeń. Utrudnia to, a czasami uniemożliwia racjonalne pozyskiwanie wiedzy, zwłaszcza z wielu źródeł. Stwierdzono potrzebę opracowania metody umożliwiającej niezależne rozpatrywanie wybranych fragmentów globalnej sieci stwierdzeń, które mogą różnić się między sobą zakresem, stopniem szczegółowości a nawet rodzajem uwzględnianych relacji (np. relacje dokładne, przybliżone).

Równocześnie, na podstawie wniosków z prac związanych z pozyskiwaniem wiedzy z różnych, niezależnych źródeł, stwierdzono potrzebę rezygnacji z powszechnie akceptowanego wymagania aby baza wiedzy systemu doradczego zawierała wyłącznie elementy niesprzeczne. Złożone systemy doradcze powinny umożliwiać realizowanie procesu wnioskowania również podczas występowania sprzecznych lub warunkowo sprzecznych elementów w ich bazach wiedzy.

Celem omawianych badań było przeprowadzenie analizy możliwości zastosowania ogólnych metod modelowania wieloskalowego w budowie baz wiedzy diagnostycznych systemów doradczych. Określono metodę pozwalającą na budowanie diagnostycznych systemów doradczych,

w których bazy wiedzy występują w postaci sieci stwierdzeń, otrzymywanych w wyniku wieloskalowego złożenia niezależnych fragmentów sieci. Możliwość dekompozycji budowanej sieci pozwala między innymi na uproszczenie procesów pozyskiwania wiedzy, które będą mogły dotyczyć modułów o rozmiarach znacznie mniejszych od rozmiarów sieci globalnej.

1.1. Zakres monografii

W kolejnych rozdziałach monografii omówiono podstawowe pojęcia związane z sieciami stwierdzeń, a w szczególności z sieciami stwierdzeń przybliżonych i z przybliżonymi sieciami stwierdzeń. Pokazano, że sieci reprezentujące warunki konieczne i dostateczne pozwalają na budowanie modeli wielomodalnych w postaci sieci wielowarstwowych. Łączenie takich sieci polega na ich "spłaszczaniu" i nie wymaga stosowania złożonych operatorów agregujących, niezbędnych do stosowania w modelach budowanych z użyciem sieci bayesowskich. Wprowadzone sieci reprezentujące przybliżone warunki konieczne i dostateczne pozwalają na pozyskiwanie wiedzy z częściowo sprzecznych źródeł. Zaproponowano miarę sprzeczności stwierdzeń występujących w takiej sieci. Opisano procedurę rozwiązywania przybliżonych sieci składających się ze stwierdzeń przybliżonych. W celu przeprowadzenia badań proponowanych rozwiązań opracowano system szkieletowy MMNET umożliwiający stosowanie wieloskalowej sieci stwierdzeń. System zbudowano stosując język R. Uzasadniono wybór tego języka. Opisano działanie systemu, interfejs użytkownika oraz sposób jego instalowania na podstawie pakietu dystrybucyjnego. Pokazano wybrane przykłady prostych sieci stwierdzeń, ilustrujące różne sposoby ich pozyskiwania.

Opracowany system szkieletowy jest rozpowszechniany w postaci pakietu MMNET, za pośrednictwem witryny Katedry Podstaw Konstrukcji Maszyn Politechniki Śląskiej, dostępnej pod adresem <http://kpk.m.polsl.pl> (rozdział *MMNET* w dziale *Projekty*).

1.2. Oznaczenia i skróty

Przyjęto następujący system oznaczeń:

\underline{x}	wektor (macierz jednokolumnowa)
\underline{x}^T	wektor transponowany (macierz jednowierszowa)
$\underline{x}_{(n)}$	wektor n-elementowy
$x_j, \underline{x}[j]$	j -ty element wektora \underline{x}
$x_{1:k}, \underline{x}[1 : k]$	wektor (macierz jednokolumnowa) złożony z pierwszych k elementów wektora \underline{x}
$x_{k:m}, \underline{x}[k : m]$	wektor (macierz jednokolumnowa) złożony z elementów od k do m wektora \underline{x}
\underline{X}	macierz
$\underline{X}_{(m \times n)}$	macierz \underline{X} posiadająca m wierszy i n kolumn
$x_{i,j}, \underline{X}[i, j]$	element macierzy \underline{X} (i -ty wiersz, j -ta kolumna)
$x_{1:k,j}, \underline{X}[1 : k, j]$	wektor (macierz jednokolumnowa) złożony z pierwszych k elementów j -tej kolumny macierzy \underline{X}

W tekście stosowany jest również skrót *SPPS* zastępujący grupę wyrazów „stopień przekonania o prawdziwości (słuszności) wypowiedzi będącej treścią stwierdzenia”.

Rozdział 2

Pojęcia podstawowe

Wojciech CHOLEWA

W rozdziale tym omówiono pojęcie *stwierdzenia* oraz wprowadzono podział stwierdzeń na *stwierdzenia proste* i *stwierdzenia złożone*. Zwrócono uwagę na możliwość rozpatrywania *sieci stwierdzeń*. W sieciach tych wyróżniono *stwierdzenia pierwotne*, *stwierdzenia wtórne* oraz *stwierdzenia izolowane*.

2.1. Stwierdzenia

Diagnostyczne systemy doradcze mają wspomagać rozpoznawanie stanu technicznego na podstawie dostępnych informacji o rozpatrywanym obiekcie i jego działaniu. Informacje takie mogą być przekazywane w postaci różnych wypowiedzi opisujących obserwowane zdarzenia, procesy i prawidłowości oraz związane z nimi przekonania i domysły.

Dla potrzeb omawianych systemów informacje te zapisywane są w postaci zdań logicznych, czyli zdań oznajmujących przyjmujących różne wartości logiczne - np. takie jak prawda, fałsz, stopień możliwości, konieczności, przekonania o słuszności.

W systemach bazujących na klasycznym wnioskowaniu logicznym zdania takie reprezentowane są w postaci zmiennych logicznych, którym przypisano odpowiednią interpretację. Wnioskowanie w takich systemach polega na przechodzeniu od znanych przesłanek do wniosków i może być realizowane jako wnioskowanie dedukcyjne lub redukcyjne, w zależności od tego czy między przesłankami i wnioskiem zachodzi stosunek wynikania logicznego. Schematy wnioskowania logicznego mogą być modyfikowane i rozszerzane, umożliwiając np. wnioskowanie niepewne i niedokładne. Większość związanych z tym prac zwraca szczególną uwagę na formalną stronę procesu wnioskowania zakładając, że interpretacja stosowanych zmiennych nie wymaga szczególnych działań.

Z prac związanych z próbami praktycznego zastosowania różnych wersji systemów doradczych wynika jednak, że zagadnienie odpowiedniej interpretacji stosowanych zmiennych, czyli przypisywania im właściwego znaczenia, wymaga szczególnej uwagi. Założenie, że konstruktorzy systemu doradczego, autorzy jego bazy wiedzy oraz przyszli użytkownicy tego systemu posługują się wspólnymi zasobami pojęć oraz takimi samymi zasadami konwersacji, regulującymi komunikację między nimi, jest często bezpodstawne i prowadzić może do niezamierzonych skutków. Informacje przekazywane jako wynik działania systemu mogą być niezrozumiałe lub niejasne dla jego użytkowników.

Omawiając sprawy dotyczące interpretacji należy uwzględnić specyficzną rolę, jaką pełnią omawiane systemy. Są to systemy doradcze, czyli systemy wspomagające podejmowanie decy-

zji, gdzie ostateczną decyzję podejmuje użytkownik systemu. Podejmując tę decyzję korzysta on pośrednio z wiedzy zgromadzonej w bazie wiedzy systemu. Można postawić w tym miejscu pytanie, kto odpowiada za końcowy wynik działań czyli za podjętą decyzję - użytkownik, autorzy bazy wiedzy czy konstruktorzy systemu. Ze zgodnych opinii wielu prawników wynika, że pełną odpowiedzialność ponosi użytkownik systemu. Uznając brak możliwości weryfikacji bazy wiedzy przez użytkownika należy przyjąć, że w celu przejęcia przez niego odpowiedzialności system powinien przekonać użytkownika o słuszności proponowanego rozwiązania. W systemach doradczych stosowane są w tym celu funkcje objaśniające przebieg procesu wnioskowania.

Należy zwrócić uwagę na możliwość stosowania stwierdzeń ułatwiających komunikację z użytkownikiem systemu. Stwierdzenia mogą być interpretowane jako obiekty pełniące rolę zdań logicznych, zawierające jednocześnie opis ich interpretacji, występujący w postaci dowolnie rozbudowanej wypowiedzi lub opinii na określony temat. Stosowanie stwierdzeń jest postępowaniem szczególnie dogodnym dla systemów przeznaczonych do wspomagania diagnostyki technicznej, wymagającej między innymi rozwiązywania zadań związanych z jednoznacznym interpretowaniem wartości danych (np. wartości cech sygnałów) i ich zmian.

Stwierdzenie jest informacją o uznaniu wypowiedzi orzekającej o obserwowanych faktach lub reprezentującej określoną opinię. Stwierdzenie s może być zapisywane w postaci następującej pary

$$s = \langle c, v \rangle \quad (2.1)$$

gdzie c jest treścią stwierdzenia, czyli np. wypowiedzią o tym, że wskazanemu obiektowi przysługuje określony atrybut o ustalonej wartości oraz v jest wartością stwierdzenia, będącą np. jednym z dwóch elementów $\{prawda, fałsz\}$. Treść stwierdzenia jest zdaniem będącym zapisem wypowiedzi orzekającej o zaobserwowanych faktach lub reprezentującej określoną opinię. Treścią stwierdzenia może być wyłącznie zdanie oznajmujące, któremu przypisywana jest jedna z wartości logicznych ($prawda, fałsz$), czyli takie zdanie, które jest prawdziwe lub fałszywe. Zdanie to musi zawierać orzeczenie. Jego podmiot może pozostać domyślny.

Stosowanie stwierdzeń w systemach doradczych pozwala na wprowadzenie złożonego systemu pomocy, zawierającego objaśnienia pojęć, odsyłacze do źródeł i różne komentarze. Zakłada się, że w rozpatrywanych systemach treść stwierdzenia jest stała, a wartość stwierdzenia może się zmieniać. Założenie to umożliwia tworzenie tezaurusów, czyli zbiorów stałych treści stwierdzeń. Tezaurusy ułatwiają zarządzanie objaśnieniami treści stwierdzeń, w tym również sterowanie stopniem szczegółowości objaśnień.

Stosując stwierdzenia należy zwracać uwagę na wyraźne rozróżnianie obiektywnych faktów i wypowiedzi o występujących faktach. Stwierdzenia bardzo często, w sposób błędny, są utożsamiane z faktami.

Definicja (2.1) określa stwierdzenie, zawierające jeden wariant treści, np.

$$c = \text{„jabłko jest czerwone”} \quad (2.2)$$

Stwierdzenie takie nazywane jest *stwierdzeniem prostym*. Definicja stwierdzenia prostego może być rozszerzona do postaci (2.3) określającej *stwierdzenie złożone* (wielowariantowe)

$$s = \langle \underline{c}, \underline{v} \rangle = \langle c_{1:n}, v_{1:n} \rangle \quad (2.3)$$

gdzie treść stwierdzenia złożonego (stwierdzenia n -wariantowego) \underline{c} jest n -elementowym wektorem (listą) wariantów treści stwierdzenia, a wartość stwierdzenia złożonego \underline{v} jest n -elementowym wektorem wartości kolejnych wariantów jego treści. Najczęściej zakłada się, że lista wariantów treści stwierdzenia jest listą wyczerpującą (kompletną) zawierającą warianty wzajemnie wykluczające się, gdzie

- dwa stwierdzenia są stwierdzeniami wzajemnie wykluczającymi się wtedy i tylko wtedy, gdy dla dowolnego stanu rzeczy nie mogą być one równocześnie prawdziwe,

- zbiór stwierdzeń jest zbiorem kompletnym wtedy i tylko wtedy gdy dla dowolnego stanu rzeczy zbiór ten zawiera stwierdzenie prawdziwe.

Przykładem treści stwierdzenia wielowariantowego, o liczbie wariantów $n = 3$, jest lista $c_{1:3}$, gdzie

$$\begin{aligned} c_1 &= \text{„jabłko jest czerwone”} \\ c_2 &= \text{„jabłko jest zielone”} \\ c_3 &= \text{„jabłko ma inny kolor niż czerwony lub zielony”} \end{aligned} \quad (2.4)$$

Stwierdzenia złożone (2.3) mogą być zastępowane odpowiednimi zbiorami stwierdzeń prostych (2.1). Nie jest to jednak postępowanie zalecane ponieważ utrudnia ono reprezentowanie informacji o tym, że tak utworzony zbiór stwierdzeń prostych jest zbiorem wyczerpującym i obejmuje stwierdzenia wzajemnie wykluczające się.

2.2. Stwierdzenia przybliżone

Stwierdzenia zapisywane w postaci (2.1) mogą być rozpatrywane jako stwierdzenia przybliżone wtedy gdy posiadają przybliżoną treść i/lub przybliżoną wartość. W dalszych rozważaniach zakładamy, że treść stwierdzenia jest dokładna a omawiane stwierdzenia są *stwierdzeniami przybliżonymi* wtedy gdy zbiór ich dopuszczalnych wartości nie jest ograniczany do dwóch elementów {prawda, fałsz}.

Przybliżona wartość stwierdzenia może być definiowana np. jako stopień prawdziwości lub stopień przekonania o prawdziwości (słuszności) wypowiedzi będącej treścią stwierdzenia. Dla oznaczenia takiego stopnia przekonania będzie stosowany skrótowiec (akronim) SPPS. Wartość SPPS dla stwierdzenia s będzie zapisywana jako funkcja $b(s)$. Wartość ta pozwala na porównywanie słuszności stwierdzeń i wyznacza liniowy porządek w zbiorze stwierdzeń, w którym stwierdzenia o mniejszych wartościach SPPS poprzedzają stwierdzenia o większych wartościach SPPS.

Formalna definicja wartości SPPS jest trudna do ustalenia. Z określenia „*stopień przekonania*” wynika jedynie subiektywny charakter tej wartości. Wartości SPPS mogą być określane jako wartości punktowe (dokładne), np.

$$b(s) = 0,8 \quad (2.5)$$

lub jako wartości przedziałowe (przybliżone), np.

$$b(s) = [0,4, 0,9]. \quad (2.6)$$

Wartości przedziałowe interpretowane są jako informacja o tym, że o nieznannej dokładnej wartości punktowej wiadomo jedynie to, iż jest zawarta we wskazanym przedziale wartości. Wartości te są zapisywane

$$b(s) = [b_{min}(s), b_{max}(s)] \text{ gdzie } 0 \leq b_{min}(s) \leq b_{max}(s) \leq 1 \quad (2.7)$$

lub

$$b(s) \in [b_{min}(s), b_{max}(s)] \text{ gdzie } 0 \leq b_{min}(s) \leq b_{max}(s) \leq 1 \quad (2.8)$$

lub

$$0 \leq b_{min}(s) \leq b(s) \leq b_{max}(s) \leq 1 \quad (2.9)$$

Przedziały określające przybliżone wartości SPPS rozpatrywane są jako przedziały obustronnie domknięte. Pozwala to na rozpatrywanie wartości punktowej $b(s)$ jako szczególnego przypadku wartości przedziałowej $b(s) \in [b_{min}(s), b_{max}(s)]$, spełniającej warunek $b_{min}(s) = b_{max}(s)$.

Nieokreślona wartość SPPS stwierdzenia s zapisywana jest w postaci następującej wartości przedziałowej

$$b(s) = [0 , 1] \quad (2.10)$$

2.3. Działania na stwierdzeniach przybliżonych

Na stwierdzeniach przybliżonych, podobnie jak na zdaniach w rachunku zdań, można wykonywać następujące podstawowe działania

- suma (alternatywa) stwierdzeń $s = s_i \vee s_j$
- iloczyn (koniunkcja) stwierdzeń $s = s_i \wedge s_j$
- negacja stwierdzenia $s = \neg s_i$

SPPS stwierdzenia będącego wynikiem tych działań można wyznaczyć na podstawie odpowiednich norm trójkątnych. Zakładamy, że SPPS stwierdzeń s_i oraz s_j określane są z dokładnością do przedziałów, jak w (2.13), czyli

$$b(s_i) = [b_{min}(s_i) , b_{max}(s_i)] \quad (2.11)$$

$$b(s_j) = [b_{min}(s_j) , b_{max}(s_j)] \quad (2.12)$$

SPPS wyników działań dla znanych (2.11) i (2.12) wyznaczane są wg następujących zależności

$$b(s_i \vee s_j) = [\max(b_{min}(s_i), b_{min}(s_j)) , \max(b_{max}(s_i), b_{max}(s_j))] \quad (2.13)$$

$$b(s_i \wedge s_j) = [\min(b_{min}(s_i), b_{min}(s_j)) , \min(b_{max}(s_i), b_{max}(s_j))] \quad (2.14)$$

$$b(\neg s_i) = [1 - b_{max}(s_i) , 1 - b_{min}(s_i)] \quad (2.15)$$

Można wykazać, że operatory działań (2.13), (2.14) i (2.15) spełniają prawa De Morgana, tzn.

$$b(\neg(s_i \wedge s_j)) = b(\neg s_i \vee \neg s_j) \quad (2.16)$$

$$b(\neg(s_i \vee s_j)) = b(\neg s_i \wedge \neg s_j) \quad (2.17)$$

2.4. Sieci stwierdzeń

W procesie wnioskowania, prowadzonym w systemie doradczym na podstawie wartości logicznych przesłanek, przyjmuje się lub odrzuca konkluzje. Rozpatrywany zbiór przesłanek i konkluzji jest zbiorem zamkniętym, ustalonym w czasie budowania bazy wiedzy. Podczas działania systemu nie są generowane lub odkrywane nowe konkluzje. Oznacza to, że procesy wnioskowania prowadzone w systemach doradczych są procesami realizowanymi w tzw. zamkniętych światach. W każdym systemie wszystkie przesłanki i konkluzje występujące w takich procesach mogą być zastąpione skończonym zbiorem stwierdzeń. Wartości stwierdzeń mogą być interpretowane np. jako wartości logiczne zastępowanych przez te stwierdzenia przesłanek i konkluzji.

Rozpatrując systemy doradcze bazujące na tablicach ogłoszeń oraz rozwijając wyjściową koncepcję tablicy ogłoszeń [2.4], [2.5] można przyjąć, że występujące na niej ogłoszenia zostaną zastąpione przez stwierdzenia [2.1], [2.2]. Oddziaływania pomiędzy ogłoszeniami, reprezentujące określoną wiedzę, zostaną zastąpione przez oddziaływania występujące w *sieci stwierdzeń*. Zmiany wartości stwierdzeń będą mogły inicjować ciągi działań powodujących zmiany wartości innych stwierdzeń. Umożliwi to realizowanie procesu wnioskowania.

Sposoby wnioskowania w sieci stwierdzeń muszą być dopasowane do rozpatrywanych typów stwierdzeń, czyli do przyjętych definicji wartości stwierdzeń. Np. dla stwierdzeń, których wartości interpretowane są jako stopnie przekonania o prawdziwości stwierdzeń, sieci stwierdzeń przybliżonych mogą być rozpatrywane jako szczególne wersje sieci przekonań, co pozwala na zastosowanie metod wnioskowania opracowanych dla sieci przekonań. Ciekawe wnioski można uzyskać rozpatrując takie sieci jako analogie układów mechanicznych [2.3].

2.5. Stwierdzenia pierwotne i wtórne

Sieć stwierdzeń pozwala na realizowanie procesu wnioskowania, którego istotą jest wyznaczanie nieznanymi wartości stwierdzeń na podstawie pozyskanych z różnych źródeł wartości innych stwierdzeń. W sieci stwierdzeń można wyróżnić następujące podzbiory ogólnego zbioru stwierdzeń S :

- zbiór stwierdzeń o znanych wartościach, zawierający
 - zbiór $S_I \subset S$ *stwierdzeń pierwotnych*, których wartości zostały ustalone bezpośrednio przez procesy zewnętrzne (np. działające układy pomiarowe lub dialog z użytkownikiem systemu),
 - zbiór $S_C \subset S$ *stwierdzeń stałych*, których wartości przyjęto arbitralnie na podstawie wartości domyślnych ustalonych przez konstruktora sieci stwierdzeń (na podstawie różnych źródeł); wartości stwierdzeń stałych pozwalają na ustalanie domyślnych wartości wybranych stwierdzeń wtórnych,
- zbiór stwierdzeń o nieznanymi wartościach, zawierający
 - zbiór $S_{II} \subset S$ *stwierdzeń wtórnych*, których wartości zależą (jawnie) od wartości innych stwierdzeń występujących w sieci i nie są zadawane bezpośrednio przez procesy zewnętrzne,
 - zbiór $S_E \subset S$ *stwierdzeń izolowanych*, których wartość nie jest znana i nie zależy od wartości innych stwierdzeń.

Zakłada się, że znane wartości stwierdzeń pierwotnych są wyznaczone (określone, zadane) poprawnie i nie powinny być modyfikowane w wyniku działania sieci. Możliwość zmieniania tych wartości rozpatrywana jest wyłącznie wtedy gdy wiadomo, że znane wartości stwierdzeń oraz wiedza zapisana w sieci w postaci zależności występujących między stwierdzeniami, pozwala na ograniczanie przedziałów wartości wybranych stwierdzeń przybliżonych. Postępowanie takie pozwala na doskonalenie (uzgadnianie) danych wejściowych w procesie wnioskowania.

Podział zbioru stwierdzeń na stwierdzenia pierwotne i wtórne nie jest podziałem stałym. Zależy on od informacji aktualnie dostępnych ze źródeł zewnętrznych (układy pomiarowe, odpowiedzi użytkownika, itp.).

Nieznanymi wartości stwierdzeń wtórnych mogą być (lub zostały) wyznaczone na podstawie wartości stwierdzeń o znanych wartościach. Należy zwrócić uwagę, że uznanie iż wartości jednych stwierdzeń zależą od wartości innych stwierdzeń nie oznacza, że zależności te są związkami przyczynowo-skutkowymi. W szczególności nie oznacza to, że stwierdzenia pierwotne dotyczą przyczyn, a stwierdzenia wtórne ich skutków. Część stwierdzeń wtórnych może występować w postaci *stwierdzeń ukrytych*, pełniących rolę wniosków pośrednich, niedostępnych (tzn. niewidocznych) dla użytkowników systemu. Stwierdzenia wtórne mogą być stosowane między innymi, jako *stwierdzenia pomocnicze* przeznaczone do reprezentowania wyrażeń złożonych z innych stwierdzeń.

Danymi wejściowymi w procesie wnioskowania w sieciach stwierdzeń są wartości stwierdzeń pierwotnych S_I oraz wartości stwierdzeń stałych S_C . Ze względu na potrzebę uwzględniania

zależności występujących pomiędzy stwierdzeniami, wnioskowanie w sieci stwierdzeń rozpatrywane jest najczęściej jako zadanie poszukiwania równowagi w tej sieci. Celem tego procesu jest wyznaczenie wartości wszystkich lub wybranych stwierdzeń wtórnych S_{II} . Wynik procesu wnioskowania przedstawiany jest w postaci wartości wybranych stwierdzeń należących do zbioru *stwierdzeń wynikowych* S_o , który ustalany jest z uwzględnieniem potrzeb i oczekiwań użytkownika systemu. Może on zawierać zarówno stwierdzenia pierwotne jak i wtórne

$$S_o \subset (S_I \cup S_{II}) \subset S. \quad (2.18)$$

Najczęściej zbiór ten nie zawiera stwierdzeń stałych i stwierdzeń izolowanych

$$S_o \cap (S_C \cup S_E) = \emptyset, \quad (2.19)$$

które nie niosą informacji wynikających z prowadzonego procesu wnioskowania, czyli nie są przydatne dla użytkownika systemu.

Bibliografia

- [2.1] Cholewa W. Sieci stwierdzeń w diagnostyce technicznej. *Diagnostyka*, 38:121–128, 2006.
- [2.2] Cholewa W. Sieci stwierdzeń w diagnostycznych systemach doradczych. Kowalczyk Z., Wiszniewski B., redaktorzy, *Inteligentne wydobywanie informacji w celach diagnostycznych*, strony 187–202. PWNT, Gdańsk, 2007.
- [2.3] Cholewa W. Mechanical analogy of statement networks. *International Journal of Applied Mathematics and Computer Science*, 18(4):477–486, 2008.
- [2.4] Englemore R., Morgan T., redaktorzy. *Blackboard Systems*. Addison-Wesley, 1988.
- [2.5] Hayes-Roth B. An architecture for adaptive intelligent systems. *Artificial Intelligence*, 72:329–365, 1995.

Rozdział 3

Wielomodalne sieci stwierdzeń

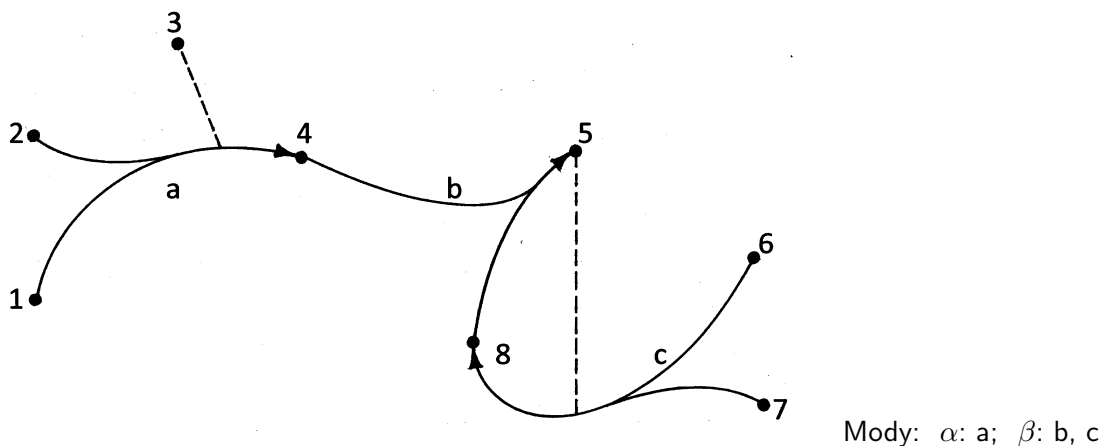
Pojęcia *stwierdzenie* i *stwierdzenie przybliżone* omówiono w rozdziale 2. W niniejszym rozdziale omawiane są zagadnienia dotyczące sieci rozpiętych na takich stwierdzeniach. Pominięto szczegóły dotyczące sieci bayesowskich, które są szeroko omawiane w literaturze, np. [3.6], [3.7]. Uwagę skupiono na sieciach pozwalających na reprezentowanie warunków koniecznych i dostatecznych oraz przybliżonych wersji tych warunków. Pokazano sposoby rozwiązywania omawianych sieci.

3.1. Wstęp

Określenie *wielomodalny* może być stosowane w różnym znaczeniu. Sieci wielomodalne to sieci, w których równocześnie uwzględniane są różne rodzaje oddziaływań występujących pomiędzy elementami sieci. Zależności (relacje) występujące pomiędzy stwierdzeniami w sieci stwierdzeń mogą należeć do różnych klas, reprezentując różne aspekty wiedzy, obowiązujące dla różnych warunków (założeń) i w różnych kontekstach. Przedstawianie takich aspektów, warunków i kontekstów w postaci zwiększanej liczby stwierdzeń (węzłów) sieci może prowadzić do nadmiernego wzrostu rozmiarów sieci (liczby węzłów i gałęzi). Rozwiązaniem alternatywnym jest wykorzystanie koncepcji wielomodalnej sieci stwierdzeń, definiowanej podobnie jak sieć wielomodalna, wprowadzona przez [3.8], [3.5] and [3.4].

Przyjmowane są następujące definicje:

- *Model sieciowy* może być zapisywany jako graf skierowany lub hipergraf skierowany.
- *Graf skierowany* jest parą (V, E) składającą się z niepustego, skończonego zbioru wierzchołków V i skończonego zbioru krawędzi skierowanych E .
- *Krawędź* $e \in E$ skierowana od jej ogona $t \in V$ do jej głowy $h \in V$ jest zapisywana jako $e = \langle t, h \rangle$.
- *Hipergraf skierowany* jest parą (V, E) składającą się z niepustego, skończonego zbioru wierzchołków V i skończonego zbioru hiperkrawędzi skierowanych E .
- *Hiperkrawędź* $e \in E$ skierowana od jej ogona $T \subseteq V$ do jej głowy $H \subseteq V$ jest zapisywana jako $e = \langle T, H \rangle$.
- Poszukując możliwości reprezentowania mod $\gamma \in \Gamma$ i związanych z nimi zbiorów $A \subseteq V$, które nie mogą być włączone bezpośrednio do ogonów lub głów krawędzi, wprowadzono [3.8] pojęcie *hiperkrawędzi modalnej* $e = (T, H, A, \gamma)$.



Rys. 3.1: Przykład sieci wielomodalnej (na podstawie [3.5])

- Sieć wielomodalna jest trójką $MMN = (V, E, \Gamma)$ i może być prezentowana podobnie jak hipergraf.

Rysunek 3.1 pokazuje sieć wielomodalną ze zbiorem wierzchołków $V = \{1, 2, 3, 4, 5, 6, 7, 8\}$, zbiorem mod hiperkrawędzi $\Gamma = \{\alpha, \beta\}$ oraz zbiorem hiperkrawędzi modalnych $E = \{a, b, c\}$ gdzie $a = (\{1, 2\}, \{4\}, \{3\}, \alpha)$, $b = (\{4, 8\}, \{5\}, \emptyset, \beta)$, $c = (\{6, 7\}, \{8\}, \{5\}, \alpha)$.

Istotą rozwiązania przyjętego w omawianym systemie jest stosowanie modeli reprezentowanych przez wielomodalne sieci stwierżeń, występujące w postaci wielowarstwowych sieci stwierżeń (rys.3.2), charakteryzujących się tym, że wybrane węzły niezależnie budowanych sieci składowych mogą należeć równocześnie do kilku warstw [3.3]. Wartość stwierżenia reprezentowanego przez węzeł występujący w kilku warstwach może być wyznaczana w wyniku agregacji wartości tego węzła w kolejnych warstwach [3.1], [3.2]. Postępowanie takie jest konieczne wtedy gdy w co najmniej jednej warstwie występuje sieć bayesowska. Postępowania tego można uniknąć wtedy, gdy we wszystkich warstwach występują sieci reprezentujące warunki konieczne i dostateczne. Wartości stwierżeń mogą być wyznaczane dla takich sieci w wyniku ich „spłaszczenia”, tzn. złożenia w jedną sieć zawierającą wszystkie węzły i gałęzie.

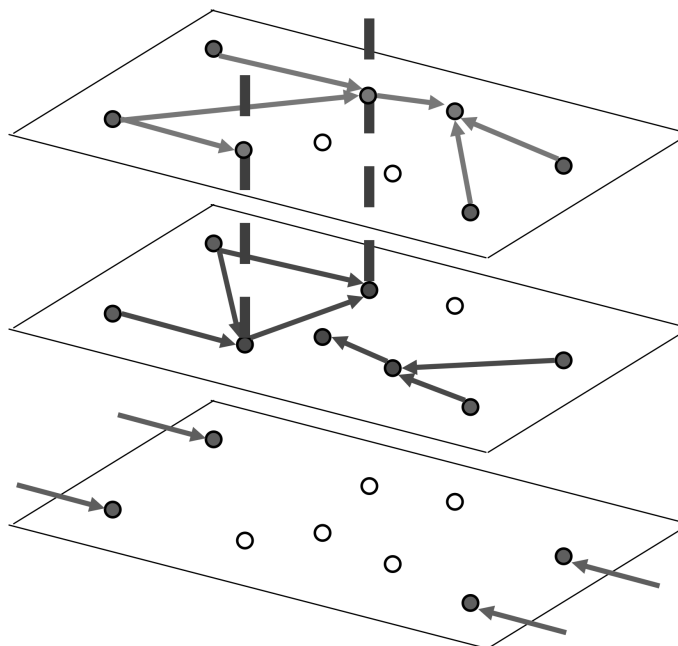
W rozważaniach prowadzonych w tym rozdziale rozpatrywane są wyłącznie stwierżenia proste (2.1). Założono, że stwierżenia złożone (2.3) mogą być zastępowane odpowiednimi zbiorami stwierżeń prostych.

3.2. Warunki konieczne i dostateczne

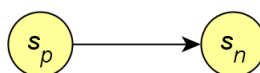
Stosowanie sieci Bayesa związane jest z potrzebą wyznaczania odpowiednich tablic prawdopodobieństw warunkowych. Wartości tych prawdopodobieństw są zazwyczaj trudne do ustalenia. Możliwe jest uproszczenie postępowania polegające na zastosowaniu metod pozwalających na reprezentowanie wiedzy w postaci warunków koniecznych i dostatecznych (wystarczających).

Jeżeli przekonaniu o prawdziwości stwierżenia s_p towarzyszy zawsze przekonanie o prawdziwości stwierżenia s_n , lecz niekoniecznie odwrotnie, to s_p określane jest jako *warunek dostateczny* dla s_n . Jednocześnie s_n jest określane jako *warunek konieczny* dla s_p . Zakładamy, że dla SPSS stwierżeń s_p i s_n przyjmujących wartości

$$b(s_p) \in [0, 1], \quad b(s_n) \in [0, 1] \quad (3.1)$$



Rys. 3.2: Przykład sieci wielowarstwowej

Rys. 3.3: Graf reprezentujący zależność (3.2) czyli pokazujący stwierdzenie s_p jako warunek dostateczny dla s_n oraz stwierdzenie s_n jako warunek konieczny dla s_p

informację o tym, że s_p jest warunkiem dostatecznym dla s_n można zapisać

$$b(s_p) \leq b(s_n) \quad (3.2)$$

gdzie po uwzględnieniu (3.1)

- z pełnego przekonania o prawdziwości stwierdzenia s_p , czyli z $b(s_p) = 1$ wynika $b(s_n) = 1$, czyli pełne przekonanie o prawdziwości stwierdzenia s_n ,
- z braku przekonania o prawdziwości stwierdzenia s_p , czyli z $b(s_p) = 0$ wynika jedynie wniosek trywialny $b(s_n) = [0, 1]$.

Zależność (3.2) można również interpretować jako zapis tego, że s_n jest warunkiem koniecznym dla s_p gdzie po uwzględnieniu (3.1)

- z braku przekonania o prawdziwości stwierdzenia s_n , czyli z $b(s_n) = 0$ wynika $b(s_p) = 0$, czyli brak przekonania o prawdziwości stwierdzenia s_p ,
- z pełnego przekonania o prawdziwości stwierdzenia s_n , czyli z $b(s_n) = 1$ wynika jedynie wniosek trywialny $b(s_p) = [0, 1]$.

Warunki konieczne i dostateczne mogą być reprezentowane w postaci grafów, jak na rys. 3.3, gdzie zwrot gałęzi wskazuje rosnące (lub niemalejące) uporządkowanie wartości SPSS.

3.3. Sieci stwierdzeń i ich rozwiązywanie

Pojęcie *sieć stwierdzeń* wprowadzono ogólnie na str. 6. Przykład bardzo prostej sieci stwierdzeń pokazano na rys. 3.4. Rozpatrujemy stwierdzenie s oraz stwierdzenia s_p i s_n takie, że

- przekonanie o słuszności stwierdzenia s_p pociąga za sobą przekonanie o słuszności stwierdzenia s ,
- przekonanie o braku słuszności stwierdzenia s_n pociąga za sobą przekonanie o braku słuszności stwierdzenia s .

Przykłady treści stwierdzeń spełniających te założenia:

$$\begin{aligned} c(s) &= \text{„Michał ukończył studia”} \\ c(s_p) &= \text{„Michał uzyskał stopień doktora”} \\ c(s_n) &= \text{„Michał zdał maturę”} \end{aligned}$$

Stwierdzenia s_p i s_n są odpowiednio warunkiem dostatecznych i warunkiem koniecznym dla stwierdzenia s (patrz rys. 3.4). Zależności występujące pomiędzy wartościami stwierdzeń można zapisać, uwzględniając (3.2), w postaci układu nierówności

$$b(s_p) \leq b(s) \quad (3.3)$$

$$b(s) \leq b(s_n) \quad (3.4)$$

3.3.1. Przykłady

Poszukiwane są nieznanne wartości stwierdzeń wynikające z dostępnych danych. W celu zilustrowania proponowanego sposobu postępowania rozpatrzono różne stany omawianej sieci.

Znane: $b(s) = 1$

Nierówności (3.3) i (3.4) oraz ograniczenia (3.1) prowadzą do rozwiązania

$$b(s_p) \in [0, 1], \quad (3.5)$$

$$b(s_n) = 1, \quad (3.6)$$

Znane: $b(s) = 0,8$

Nierówności (3.3) i (3.4) oraz ograniczenia (3.1) prowadzą do rozwiązania

$$b(s_p) \in [0, 0,8], \quad (3.7)$$

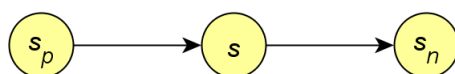
$$b(s_n) \in [0,8, 1]. \quad (3.8)$$

Znane: $b(s) = 0$

Nierówności (3.3) i (3.4) oraz ograniczenia (3.1) prowadzą do rozwiązania

$$b(s_p) = 0, \quad (3.9)$$

$$b(s_n) \in [0, 1]. \quad (3.10)$$



Rys. 3.4: Warunek konieczny i dostateczny dla stwierdzenia s

3.3.2. Znane dokładne wartości wszystkich stwierdzeń

Stosowanie nierówności (3.3) i (3.4) dla dokładnych wartości stwierdzeń nie wymaga dodatkowych objaśnień. W szczególności dla znanych wartości $b(s)$, $b(s_p)$ oraz $b(s_n)$, przez ich wstawienie do (3.3) i (3.4), możliwe jest sprawdzanie zgodności tych wartości, polegające na sprawdzaniu spełnienia warunków określonych układem nierówności (3.3) i (3.4).

3.3.3. Znane dokładne wartości wybranych stwierdzeń

Układ nierówności (3.3) i (3.4) może być stosowany, między innymi, do wyznaczania nieznannej wartości stwierdzenia s na podstawie znanych dokładnych wartości stwierdzeń s_p , s_n . Jeżeli wartości $b(s_p)$ i $b(s_n)$ znane są jako wartości dokładne i spełniony jest warunek

$$b(s_p) \leq b(s_n) \quad (3.11)$$

to rozwiązanie nierówności składowych (3.3) i (3.4), określające wartość stwierdzenia s , mają postać

$$b(s_p) \leq b(s) \implies b(s) \in [b(s_p), 1], \quad (3.12)$$

$$b(s) \leq b(s_n) \implies b(s) \in [0, b(s_n)]. \quad (3.13)$$

Wspólna część rozwiązań (3.12) i (3.13) wyznaczona z uwzględnieniem (3.11)

$$b(s) = [b(s_p), b(s_n)] \quad (3.14)$$

jest rozwiązaniem układu nierówności (3.3) i (3.4), określającym wartość stwierdzenia s .

3.3.4. Znane przybliżone wartości wybranych stwierdzeń

Wartości stwierdzeń s_p i s_n mogą być znane jedynie jako wartości przybliżone

$$b(s_p) = [b_{min}(s_p), b_{max}(s_p)], \quad (3.15)$$

$$b(s_n) = [b_{min}(s_n), b_{max}(s_n)]. \quad (3.16)$$

Przybliżoną wartością stwierdzenia s zawierającą wszystkie możliwe dokładne wartości tego stwierdzenia, czyli wartością przybliżoną zawierającą rozwiązania nierówności (3.3), wyznaczone dla wszystkich możliwych dokładnych wartości $b(s_p)$ spełniających (3.15), jest

$$b(s) \in [b_{min}(s_p), 1]. \quad (3.17)$$

Należy zauważyć, że zgodnie z intuicyjnymi oczekiwaniami, stwierdzenie s_p o wartości nieokreślonej nie wprowadza żadnych ograniczeń dotyczących wartości stwierdzenia s . Na podstawie (3.17) dla nieokreślonej wartości stwierdzenia s_p

$$b(s_p) = [0, 1] \quad (3.18)$$

otrzymujemy rozwiązanie wskazujące nieokreśloną wartość stwierdzenia s

$$b(s) = [0, 1]. \quad (3.19)$$

Postępując analogicznie w sprawie rozwiązań nierówności (3.4) dla danych (3.15) otrzymujemy

$$b(s) \in [0, b_{max}(s_n)]. \quad (3.20)$$

Po złożeniu rozwiązań cząstkowych (3.17) i (3.20) otrzymujemy rozwiązanie określające poszukiwaną wartość stwierdzenia s

$$b(s) = [b_{min}(s_p), b_{max}(s_n)]. \quad (3.21)$$

Oczywiście istnienie rozwiązania (3.21) uzależnione jest od spełnienia warunku

$$b_{min}(s_p) \leq b_{max}(s_n). \quad (3.22)$$

3.3.5. Ogólny algorytm rozwiązywania sieci

W celu przedstawienia ogólnego algorytmu rozwiązywania sieci wprowadza się operatory $\text{pn}(s)$ i $\text{nn}(s)$ zwracające, dla stwierdzenia s w sieci rozpiętej na zbiorze stwierdzeń S , odpowiednio zbiory warunków dostatecznych oraz warunków koniecznych

$$\forall s \in S \forall s_p \in \text{pn}(s) (b(s_p) \leq b(s)), \quad (3.23)$$

$$\forall s \in S \forall s_n \in \text{nn}(s) (b(s) \leq b(s_n)). \quad (3.24)$$

Zbiory zwracane przez operatory $\text{pn}(s)$ i $\text{nn}(s)$ identyfikowane są na podstawie istniejących gałęzi grafu reprezentującego sieć stwierdzeń.

Ogólny algorytm umożliwiający wyznaczanie nieznanymi wartości stwierdzeń występujących w sieci stwierdzeń ma postać następującą:

Wydruk 3.1: Algorytm rozwiązywania sieci dokładnej

```

% ETAP 1                                     1
  S = ...                                     2
  for s ∈ S                                    3
    b(s) = [0, 1]                             4
  end                                          5
% ETAP 2                                     6
  for s ∈ SC                                    7
    b(s) = ...                                8
  end                                          9
  SI = ...                                       10
  for s ∈ SI                                    11
    b(s) = ...                                12
  end                                          13
  SII = ...                                       14
% ETAP 3                                     15
  rep = true                                  16
  while rep                                   17
    rep = false                               18
    for s ∈ SII                                    19
      for sp ∈ pn(s)                               20
        if bmin(s) < bmin(sp)                    21
          bmin(s) = bmin(sp)                    22
          rep = true                               23
        end                                       24
      end                                       25
    end                                       26
  end                                       27
% ETAP 4                                     28
  rep = true                                  29
  while rep                                   30
    rep = false                               31
    for s ∈ SII                                    32
      for sn ∈ nn(s)                               33
        if bmax(s) > bmax(sn)                    34
          bmax(s) = bmax(sn)                    35
          rep = true                               36
        end                                       36
      end                                       36
    end                                       36
  end                                       36

```

```

                                end          37
                            end            38
                        end              39
                    end                40
                % ETAP 5                41
                for s ∈ SII            42
                    if bmin(s) - bmax(s) > eps 43
                        break(" Sprzeczność w sieci stwierdzeń" ) 44
                    end                45
                end                    46

```

W ramach przedstawionego algorytmu realizowane są następujące działania:

- ETAP 1
Inicjalizacja zbioru stwierdzeń S .
- ETAP 2
Podstawianie wartości stwierdzeń pierwotnych $s \in S_I$ oraz podstawianie wartości stwierdzeń stałych $s \in S_C$.
- ETAP 3
Modyfikowanie wartości kolejnych stwierdzeń wtórnych $s \in S_{II}$ na podstawie znanych ich warunków dostatecznych. Każda wprowadzona zmiana wartości tych stwierdzeń wymusza ponowny cykl weryfikacji (sterowany zmienną rep), ponieważ zmieniona wartość może powodować zmianę warunku dostatecznego innego stwierdzenia.
- ETAP 4
Modyfikowanie wartości kolejnych stwierdzeń wtórnych $s \in S_{II}$ na podstawie znanych ich warunków koniecznych. Każda wprowadzona zmiana wartości tych stwierdzeń wymusza ponowny cykl weryfikacji. Uzasadnienie jak dla ETAPu 3.
- ETAP 5
Sprawdzanie czy otrzymano poprawne wartości stwierdzeń, spełniające (2.9). Niepoprawne wartości, czyli takie dla których $b_{min}(s) > b_{max}(s)$ mogą wynikać z błędnej struktury sieci stwierdzeń i/lub z błędnych wartości stwierdzeń pierwotnych. Mała liczbę eps przyjęto dla zabezpieczenia się przed skutkami zakręgleń.

W celu zapewnienia czytelności algorytmu pominięto fragmenty ograniczające liczbę wielokrotnie powtarzanych sprawdzeń.

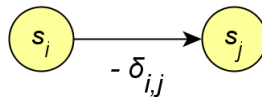
3.4. Przybliżone warunki konieczne i dostateczne

Rozpatrywane systemy powinny dopuszczać realizowanie procesu wnioskowania w warunkach nieprecyzyjnych, niekompletnych a czasami nawet sprzecznych informacji. W celu umożliwienia uwzględniania takich informacji wprowadzono stwierdzenia przybliżone (rozdział 2.2). Potrzeba prowadzenia wnioskowania przybliżonego może występować również w warunkach nieprecyzyjnej, niekompletnej oraz częściowo sprzecznej wiedzy. Taki rodzaj wiedzy może być reprezentowany w postaci przybliżonych warunków koniecznych i dostatecznych.

Przybliżone warunki konieczne i dostateczne definiowane są jako warunki zakładające, że zależność (3.2) może być spełniona z jakąś niewielką, dopuszczalną odchyłką. Wartość dopuszczalnej odchyłki, czyli wartość dopuszczalnego stopnia przybliżenia może być przyjmowana w różny sposób, np. jako

- wartość δ równa dla wszystkich rozpatrywanych warunków (oznaczająca przyjęcie założenia o równej ważności tych warunków)

$$b(s_i) - \delta \leq b(s_j), \quad \text{gdzie } \delta \geq 0, \quad (3.25)$$



Rys. 3.5: Gałąź sieci stwierdzeń reprezentująca warunek $b(s_i) - \delta_{i,j} \leq b(s_j)$

- wartości δ_i i δ_j przypisywane indywidualnie do każdego stwierdzenia s_i i s_j

$$b(s_i) - \delta_i \leq b(s_j) + \delta_j, \quad \text{gdzie } \delta_i \geq 0, \delta_j \geq 0, \quad (3.26)$$

- wartość $\delta_{i,j}$ przypisywana indywidualnie do każdego warunku określonego dla pary stwierdzeń s_i i s_j

$$b(s_i) - \delta_{i,j} \leq b(s_j), \quad \text{gdzie } \delta_{i,j} \geq 0, \quad (3.27)$$

gdzie nie wymaga się aby

$$\delta_{i,j} = \delta_{j,i}, \quad (3.28)$$

- zarówno wartości δ_i i δ_j przypisywane indywidualnie do każdego stwierdzenia s_i i s_j , jak i wartość $\delta_{i,j}$ przypisywana indywidualnie do warunku określonego dla tej pary stwierdzeń

$$b(s_i) - \delta_i - \delta_{i,j} \leq b(s_j) + \delta_j, \quad \text{gdzie } \delta_i \geq 0, \delta_j \geq 0, \delta_{i,j} \geq 0 \quad (3.29)$$

Dopuszczalne odchyłki zapisywane jak w (3.26), (3.27) i (3.29) umożliwiają różnicowanie ważności stwierdzeń warunków. W dalszych rozważaniach szczególna uwaga zostanie zwrócona na dopuszczalne odchyłki zapisywane jak w (3.27).

3.5. Przybliżone sieci stwierdzeń

Układy nierówności (3.25), (3.26), (3.27) i (3.29) mogą być przedstawiane w postaci sieci stwierdzeń (rozdział 2.4), gdzie stwierdzenia reprezentowane są jako węzły a każda nierówność reprezentowana jest przez odpowiadającą jej gałąź sieci (rys. 3.5).

Sieć stwierdzeń jest modelem reprezentującym zależności występujące pomiędzy stwierdzeniami. W ogólnym zbiorze stwierdzeń, na których rozpięta jest sieć, występują stwierdzenia o znanych wartościach, wynikających z informacji pozyskanych z różnych źródeł. Dla części stwierdzeń, nazywanych stwierdzeniami wtórnymi (rozdział 2.5) ich wartości nie są znane. Wartości stwierdzeń wtórnych mogą być rozpatrywane jako niewiadome w układach nierówności (3.25), (3.26), (3.27) lub (3.29). Rozwiązania tych układów mogą być wyznaczone w wyniku poszukiwania stanu równowagi sieci, który jest definiowany jako stan minimalizujący wartość przyjętej funkcji kryterialnej e . Na przykład dla układu nierówności (3.25) funkcję kryterialną można definiować jako

$$e = \delta \rightarrow \min \quad (3.30)$$

i odpowiednio dla układu nierówności (3.27)

$$e = \sum_{i,j} \delta_{i,j} \rightarrow \min, \quad (3.31)$$

gdzie δ i $\delta_{i,j}$ przyjmują wartości pozwalające na wyznaczenie rozwiązań układów nierówności odpowiednio (3.25) i (3.27).

Funkcja kryterialna e może być interpretowana jako *ocena (miara) kosztów* wynikających z *niedoskonałości* rozpatrywanego modelu lub niedoskonałości danych dla których model ten jest

stosowany. Zastosowanie funkcji e wg (3.31) może prowadzić do rozwiązań, w których kilka parametrów $\delta_{i,j}$ będzie osiągać wartości *duże*, podczas gdy pozostałe przyjmą wartości *małe*. Będzie to oznaczać, że warunki (3.27) odpowiadające parametrom $\delta_{i,j}$ o dużych wartościach nie będą wpływały na wyznaczanie stanu równowagi sieci (zostaną pominięte), a pozostałe zostaną uwzględnione jako warunki dokładne. Można uznać, że rozwiązaniem lepszym będzie takie, w którym zamiast kilku *dużych* wartości parametrów $\delta_{i,j}$ wystąpi większa liczba tych parametrów, przyjmujących wartości *średnie*. Będzie to oznaczać, że uwzględniane są wszystkie warunki, jako warunki przybliżone. Rozwiązanie takie można otrzymać przekształcając liniową funkcję kryterialną (3.31) w funkcję nieliniową

$$e = \sum_{i,j} \delta_{i,j}^2 \rightarrow \min, \quad (3.32)$$

Podczas gromadzenia wiedzy reprezentowanej w postaci odpowiednio zapisywanych warunków koniecznych i dostatecznych, występuje potrzeba rozpatrywania warunków obowiązujących zawsze, w większości przypadków, często lub rzadko, czyli charakteryzujących się różnym stopniem ważności. Różnicowanie ważności warunków (3.27) podczas wyznaczania wartości funkcji kryterialnej e wg (3.31) lub (3.32) można uzyskać przez wprowadzenie indywidualnych parametrów (kosztów jednostkowych) $k_{i,j}$ dla ewentualnych odchyłek $\delta_{i,j}$ warunków przybliżonych, odpowiednio

$$e = \sum_{i,j} k_{i,j} \delta_{i,j} \rightarrow \min, \quad \text{gdzie } k_{i,j} \geq 0, \quad (3.33)$$

lub

$$e = \sum_{i,j} k_{i,j}^2 \delta_{i,j}^2 \rightarrow \min. \quad (3.34)$$

Parametry $k_{i,j}$ mogą przyjmować równe wartości (np. $k_{*,*} = 1$), wtedy gdy brak potrzeby różnicowania ważności warunków. Parametry te mogą być rozpatrywane jako przyporządkowane gałęziom (i, j) sieci lub jej węzłom i oraz j . Dla parametrów przyporządkowanych węzłom można zakładać np., że

$$k_{i,j} = k_i k_j. \quad (3.35)$$

gdzie k_i, k_j oznaczają koszty jednostkowe przyporządkowane węzłom.

3.5.1. Elementy sprzeczne

Podczas budowania baz wiedzy zakłada się, że powinny one zawierać wyłącznie elementy niesprzeczne. Formalna poprawność baz wiedzy powinna być przedmiotem weryfikacji w czasie budowania i eksploatacji systemu. Wykrywanie elementów sprzecznych jest zadaniem prostym wtedy gdy rozpatrywane są systemy stosujące dokładne przesłanki i dokładne reguły. Niestety, jest to zadanie trudne dla systemów przybliżonych. W celu jego uproszczenia zakłada się, że ewentualne elementy sprzeczne mogą być wykrywane w czasie działania systemu, jako elementy warunkowo sprzeczne dla ustalonego przypadku.

Dla wprowadzonych przybliżonych warunków koniecznych i dostatecznych można zastosować postępowanie polegające na wyznaczaniu stanu równowagi sieci. Minimalna wartość funkcji kryterialnej e wg (3.30), (3.33) lub (3.34) może być interpretowana jako ocena stopnia sprzeczności uwzględnianego zbioru warunków przybliżonych. Dla zbioru warunków niesprzecznych funkcja ta będzie przyjmowała wartość $e = 0$.

3.5.2. Maskowanie gałęzi sieci

Każda para węzłów może być połączona wieloma gałęziami. Należy zwrócić uwagę, że działanie gałęzi skierowanych w jedną stronę, np. gałęzi od węzła s_i do węzła s_j , jest wyznaczone przez gałąź dominującą, której przyporządkowano dopuszczalną odchyłkę δ o największej wartości. Wynika to z faktu, iż jeżeli nierówność (3.27)

$$b(s_i) - \delta_{i,j} \leq b(s_j)$$

jest spełniona dla

$$\delta_{i,j} = \delta_1, \quad (3.36)$$

to jest również spełniona dla

$$\delta_{i,j} = \delta_2, \quad \text{gdzie: } \delta_2 > \delta_1. \quad (3.37)$$

Należy przyjąć, że dominująca gałąź maskuje pozostałe gałęzie, skierowane w tę samą stronę. Gałąź maskowaną można pominąć.

Dla każdej pary węzłów można określać co najwyżej dwie niezależne gałęzie, skierowane w przeciwne strony.

3.6. Rozwiązywanie przybliżonych sieci stwierdzeń

Dalsze rozważania ograniczamy do przybliżonej sieci stwierdzeń, której rozwiązanie wyznaczone jest z zastosowaniem funkcji kryterialnej w postaci (3.33). Poszukiwanie stanu równowagi tej sieci może być rozpatrywane jako zadanie programowania liniowego (zadanie optymalizacji liniowej). Przed przystąpieniem do rozwiązywania sieci przybliżonej należy sprawdzić, czy istnieje rozwiązanie dla tej sieci rozpatrywanej jako sieć dokładna. Po uzyskaniu negatywnego wyniku należy kontynuować badanie sieci przybliżonej.

3.6.1. Zadanie programowania liniowego

Podstawowe zadanie programowania liniowego dotyczy wyznaczenia zmiennych (niewiadomych) występujących w postaci wektora $\underline{\mathbf{x}}_{(n)}$ zapewniającego maksymalną wartość liniowej funkcji celu (funkcji kryterialnej)

$$f(\underline{\mathbf{x}}_{(n)}) = \underline{\mathbf{c}}_{(n)}^T \underline{\mathbf{x}}_{(n)} \rightarrow \max, \quad (3.38)$$

dla znanych liniowych warunków ograniczających

$$\underline{\mathbf{A}}_{(m \times n)} \underline{\mathbf{x}}_{(n)} = \underline{\mathbf{d}}_{(m)}, \quad (3.39)$$

oraz

$$\underline{\mathbf{x}}_{(n)} \geq \underline{\mathbf{0}}_{(n)}, \quad (3.40)$$

gdzie macierz $\underline{\mathbf{A}}_{(m \times n)}$ oraz wektory $\underline{\mathbf{d}}_{(m)}$, $\underline{\mathbf{c}}_{(n)}$ są znane. W zadaniach unika się sprzecznego układu warunków ograniczających, co można zapewnić dla

$$n > m. \quad (3.41)$$

W opisywanym zadaniu można alternatywnie zamiast (3.38) rozpatrywać minimalizację funkcji celu zapisywaną

$$-f(\underline{\mathbf{x}}_{(n)}) = -\underline{\mathbf{c}}_{(n)}^T \underline{\mathbf{x}}_{(n)} \rightarrow \max \iff f(\underline{\mathbf{x}}_{(n)}) \rightarrow \min. \quad (3.42)$$

Warunki ograniczające, występujące w postaci układu równań (3.39), można przekształcić w odpowiednie układy nierówności. W tym celu rozpatrywane są dodatkowe nieujemne zmienne występujące w postaci wektora $\underline{\varepsilon}_{(m)}$, dla którego

$$\underline{\varepsilon}_{(m)} \geq \mathbf{0}_{(m)}. \quad (3.43)$$

Nie istnieje potrzeba wyznaczania dokładnych wartości wektora $\underline{\varepsilon}_{(m)}$. Konieczne jest jedynie założenie, że wartości te spełniają nierówność (3.43). Przekształcając (3.39) do postaci (3.43) otrzymujemy odpowiednio

$$\underline{\mathbf{A}}_{(m \times n)} \underline{\mathbf{x}}_{(n)} + \underline{\varepsilon}_{(m)} = \underline{\mathbf{d}}_{(m)}, \quad (3.44)$$

i

$$\underline{\mathbf{A}}_{(m \times n)} \underline{\mathbf{x}}_{(n)} - \underline{\varepsilon}_{(m)} = \underline{\mathbf{d}}_{(m)}, \quad (3.45)$$

otrzymujemy warunki ograniczające, które po uwzględnieniu (3.43) są równoważne warunkom

$$\underline{\mathbf{A}}_{(m \times n)} \underline{\mathbf{x}}_{(n)} \leq \underline{\mathbf{d}}_{(m)}, \quad (3.46)$$

i

$$\underline{\mathbf{A}}_{(m \times n)} \underline{\mathbf{x}}_{(n)} \geq \underline{\mathbf{d}}_{(m)}, \quad (3.47)$$

W celu uzyskania (3.44) i (3.45) w postaci podobnej do (3.39) można połączyć wektory $\underline{\mathbf{x}}_{(n)}$ i $\underline{\varepsilon}_{(m)}$, uzyskując wektor $\underline{\mathbf{x}}_{(n+m)}$, który po rozszerzeniu macierzy $\underline{\mathbf{A}}_{(m \times n)}$ prowadzi, np dla (3.44), do

$$\underline{\mathbf{A}}_{(m \times (n+m))} \underline{\mathbf{x}}_{(n+m)} = \underline{\mathbf{d}}_{(m)}. \quad (3.48)$$

Literatura dotycząca programowania liniowego, a w szczególności literatura dotycząca stosowanych algorytmów jest bardzo obszerna. Algorytmy te są stosowane w procedurach występujących w większości bibliotek numerycznych. Uznano, że ich opisywanie w tym miejscu nie jest konieczne.

3.6.2. Algorytm rozwiązywania przybliżonej sieci stwierdzeń - część I

Rozpatrujemy przybliżoną sieć stwierdzeń obejmującą p węzłów reprezentujących stwierdzenia od s_1 do s_p o nieznanach wartościach i r węzłów reprezentujących stwierdzenia od s_{p+1} do s_{p+r} o znanych wartościach. Na zbiorze $p+r$ węzłów rozpięto q_0 gałęzi skierowanych. Gałęzie reprezentują warunki konieczne i dostateczne zapisywane w postaci (3.27)

$$b(s_i) - b(s_j) - \delta_{i,j} \leq 0,$$

gdzie wartości stwierdzeń są wartościami przybliżonymi (2.11) i (2.12). W zbiorze q_0 gałęzi skierowanych występują:

- q_1 gałęzi łączących stwierdzenia s_i o nieznanach wartościach ze stwierdzeniami s_j o nieznanach wartościach,
- q_2 gałęzi łączących stwierdzenia s_i o nieznanach wartościach ze stwierdzeniami s_j o znanych wartościach,
- q_3 gałęzi łączących stwierdzenia s_i o znanych wartościach ze stwierdzeniami s_j o nieznanach wartościach,
- q_4 gałęzi łączących stwierdzenia s_i o znanych wartościach ze stwierdzeniami s_j o znanych wartościach,

gdzie

$$q_0 = q_1 + q_2 + q_3 + q_4. \quad (3.49)$$

W związku z założeniem o poprawności wartości stwierżeń pierwotnych S_I i stałych S_C , nie przewiduje się weryfikacji ich wartości. Upoważnia to do pominięcia w dalszych działaniach zbioru q_4 gałęzi łączących te stwierżenia. Zbiory q_2 i q_3 gałęzi łączących stwierżenia o znanej wartości ze stwierżeniami przybliżonymi o nieznannej wartości wymagają jednej nierówności do reprezentowania każdej gałęzi. Zbiór q_1 gałęzi łączących stwierżenia o nieznanach wartościach wymagają dwóch nierówności do reprezentowania każdej gałęzi. Do reprezentowania wszystkich gałęzi należy wobec tego zapisać q nierówności, gdzie

$$q = 2 * q_1 + 1 * q_2 + 1 * q_3 + 0 * q_4. \quad (3.50)$$

Stan równowagi sieci, identyfikowany jest z zastosowaniem funkcji kryterialnej (3.33), jako rozwiązanie zadania programowania liniowego (3.39). Dla potrzeb (3.39) przyjmujemy

$$\underline{\mathbf{x}}^T[1 : p] = [\dots, b_{min}(s_i), \dots, b_{min}(s_j), \dots], \quad (3.51)$$

$$\underline{\mathbf{x}}^T[p + 1 : p + p] = [\dots, b_{max}(s_i), \dots, b_{max}(s_j), \dots], \quad (3.52)$$

$$\underline{\mathbf{x}}^T[p + p + 1 : p + p + q] = [\dots, \delta_{i,j}, \dots], \quad (3.53)$$

oraz odpowiadający tym zmiennym fragment macierzy $\underline{\mathbf{A}}$

$$\underline{\mathbf{A}}[1 : q, 1 : p + p + q_1 + q_2 + q_3] \quad (3.54)$$

posiada elementy przyjmujące wartości 0, -1, +1.

Wprowadzamy uporządkowanie w zbiorze rozpatrywanych gałęzi (3.27), przypisując każdej gałęzi (i, j) , łączącej stwierżenie s_i określające warunek dostateczny i stwierżenie s_j określające warunek konieczny, jej numer porządkowy w odpowiednim zbiorze q_1 , q_2 lub q_3 gałęzi

$$m_1 \in \{1, 2, \dots, q_1\}, \quad (3.55)$$

$$m_2 \in \{1, 2, \dots, q_2\}, \quad (3.56)$$

$$m_3 \in \{1, 2, \dots, q_3\}, \quad (3.57)$$

wynikający z sortowania (i, j) wg i a następnie wg j dla każdego i .

Dla każdej m -tej gałęzi obrazującej

$$\text{gałąź } m : \quad b(s_i) - b(s_j) - \delta_{i,j} \leq 0, \quad (3.58)$$

możliwe są różne stany wartości stwierżeń s_i i s_j . Warunkują one różne wartości elementów macierzy $\underline{\mathbf{A}}$ i wektora $\underline{\mathbf{d}}$, w (3.44) gdzie

- dla nieznanego $b(s_i)$ oraz dla nieznanego $b(s_j)$, czyli dla

$$i \leq p, \quad j \leq p, \quad (3.59)$$

rozpatrujemy warunki

$$\text{warunek } 2m_1 - 1 : \quad b_{min}(s_i) - b_{min}(s_j) - \delta_{i,j} \leq 0 \quad (3.60)$$

$$\text{warunek } 2m_1 : \quad b_{max}(s_i) - b_{max}(s_j) - \delta_{i,j} \leq 0 \quad (3.61)$$

i podstawiamy

$$\underline{\mathbf{A}}[2m_1 - 1, i] = +1, \quad (3.62)$$

$$\underline{\mathbf{A}}[2m_1 - 1, j] = -1, \quad (3.63)$$

$$\underline{\mathbf{A}}[2m_1 - 1, p + p + m_1] = -1, \quad (3.64)$$

$$\underline{\mathbf{d}}[2m_1 - 1] = 0, \quad (3.65)$$

$$\underline{\mathbf{A}}[2m_1, p + i] = +1, \quad (3.66)$$

$$\underline{\mathbf{A}}[2m_1, p + j] = -1, \quad (3.67)$$

$$\underline{\mathbf{A}}[2m_1, p + p + m_1] = -1, \quad (3.68)$$

$$\underline{\mathbf{d}}[2m_1] = 0, \quad (3.69)$$

- dla nieznanego $b(s_i)$ oraz dla znanego $b(s_j)$, czyli dla

$$i \leq p, \quad j > p, \quad (3.70)$$

rozpatrujemy warunek

$$\text{warunek } m_2 : b_{max}(s_i) - \delta_{i,j} \leq b_{max}(s_j) \quad (3.71)$$

i podstawiamy

$$\underline{\mathbf{A}}[2q_1 + m_2, p + i] = +1, \quad (3.72)$$

$$\underline{\mathbf{A}}[2q_1 + m_2, p + p + q_1 + m_2] = -1, \quad (3.73)$$

$$\underline{\mathbf{d}}[2q_1 + m_2] = b_{max}(s_j), \quad (3.74)$$

- dla znanego $b(s_i)$ oraz dla nieznanego $b(s_j)$, czyli dla

$$i > p, \quad j \leq p, \quad (3.75)$$

rozpatrujemy warunek

$$\text{warunek } m_2 : -b_{min}(s_j) - \delta_{i,j} \leq -b_{min}(s_i) \quad (3.76)$$

i podstawiamy

$$\underline{\mathbf{A}}[2q_1 + q_2 + m_3, j] = -1, \quad (3.77)$$

$$\underline{\mathbf{A}}[2q_1 + q_2 + m_3, p + p + q_1 + q_2 + m_3] = -1, \quad (3.78)$$

$$\underline{\mathbf{d}}[2q_1 + q_2 + m_3] = -b_{min}(s_i), \quad (3.79)$$

Dla funkcji kryterialnej (3.33) zapisywanej w postaci (3.38) przyjmujemy

$$\underline{\mathbf{c}}[1 : p + p] = \mathbf{0}_{(p+p)} \quad (3.80)$$

$$\underline{\mathbf{c}}[p + p + 1 : p + p + q_1 + q_2 + q_3] = [\dots, -k_{i,j}, \dots]. \quad (3.81)$$

Wartości zmiennych reprezentujących wartości stwierdzeń $b(s)$ muszą spełniać warunek (3.1). Z ograniczenia (3.40) wynika, że są to wartości nieujemne. Dla spełnienia warunku (3.1) wymagane jest spełnienie dodatkowego ograniczenia

$$b_{min}(s) \leq b_{max}(s) \leq 1. \quad (3.82)$$

Prowadzi to do warunków

$$b_{max}(s_i) \leq 1, \quad (3.83)$$

$$b_{min}(s_i) - b_{max}(s_i) \leq 0. \quad (3.84)$$

Ich uwzględnienie wymaga podstawienia dla $i = 1, \dots, p$

$$\underline{\mathbf{A}}[q + i, p + i] = 1, \quad (3.85)$$

$$\underline{\mathbf{d}}[q + i] = 1, \quad (3.86)$$

$$\underline{\mathbf{A}}[q + p + i, i] = 1, \quad (3.87)$$

$$\underline{\mathbf{A}}[q + p + i, p + i] = -1, \quad (3.88)$$

$$\underline{\mathbf{d}}[q + p + i] = 0. \quad (3.89)$$

Otrzymane rozwiązanie jest rozwiązaniem optymalnym w świetle kryterium (3.33), a nie kryterium (3.34). Oznacza to, że rozwiązaniem mogą być duże wartości jednej lub kilku przypadkowych odchyłek $\delta_{i,j}$ i zerowe wartości pozostałych odchyłek. Niedogodność tą można ograniczyć, stosując nadal kryterium liniowe uzupełnione jednak o dodatkowe warunki ograniczające

$$\delta_{i,j} < g \text{ gdzie } g > 0, g \rightarrow \min. \quad (3.90)$$

Wartość g w (3.90) jest zmniejszana iteracyjnie tak długo, jak długo zadanie programowania liniowego posiada rozwiązanie.

Rozpatrywane zadanie programowania liniowego może być rozwiązywane jednym z dostępnych algorytmów. Wybierając algorytm należy zwracać uwagę na ewentualne założenia dotyczące np. nieujemnych wartości stałych $\underline{\mathbf{d}}$, które mogą wymagać dodatkowych modyfikacji danych. Zwracać należy również uwagę, że rozpatrywano warunki w postaci nierówności (3.46), co dla niektórych algorytmów może wymagać zastosowania dodatkowych zmiennych. Na podstawie przeprowadzonych obliczeń uzyskany zostanie wynik w postaci przybliżonych wartości stwierdzeń oraz wartości odchyłek δ . Stosując algorytmy programowania liniowego należy zwracać uwagę na to, że omawiane zadanie może zawierać warunki sprzeczne dla których nie istnieje rozwiązanie.

Opisane zadanie zawiera ograniczony zbiór warunków. Skutkiem tego może być wyznaczenie rozwiązań cząstkowych nie obejmujących wszystkich rozwiązań. Uzyskanie pełnego rozwiązania możliwe jest po zastosowaniu algorytmu opisanego w rozdziale 3.6.3 lub po uzupełnieniu warunków polegających na zastąpieniu zależności (3.80) zależnościami

$$\underline{\mathbf{c}}[1 : p] = -eps, \quad (3.91)$$

$$\underline{\mathbf{c}}[p + 1 : p + p] = +eps, \quad (3.92)$$

gdzie eps jest liczbą dodatnią, znacznie mniejszą od wartości parametrów $k_{i,j}$ w (3.33), np.

$$eps = 0,0001 \quad (3.93)$$

3.6.3. Algorytm rozwiązywania przybliżonej sieci stwierdzeń - część II

Dobrze (kompletnie) sformułowane zadanie programowania liniowego pozwala na wyznaczenie rozwiązania pełnego zawierającego wszystkie możliwe rozwiązania cząstkowe. Stosując niekompletny opis zadania, uproszczoną wersję algorytmu programowania liniowego lub inny algorytm możemy otrzymać rozwiązanie cząstkowe. Zakładamy, że rozwiązanie to określa nieznanne wartości odchyłek $\underline{\delta}_{(q)}$. Wartości tych odchyłek mogą być również ustalone arbitralnie na podstawie opinii specjalisty.

W celu uzyskania rozwiązania obejmującego wszystkie możliwe zestawy wartości stwierdzeń należy przyjąć wyznaczone wartości $\underline{d}_{(q)}$ jako wartości znane i zastosować zmodyfikowaną wersję ogólnego algorytmu rozwiązywania sieci, pokazanego na wydruku 3.1. Modyfikacja dotyczy pętli **for** $s_p \in pn(s)$ i **for** $s_n \in nn(s)$ oraz sygnalizacji ewentualnego błędu wykonania. Zmodyfikowaną wersję algorytmu pokazano na wydruku 11.10. W opisie algorytmu, występującą w (3.27), odchyłkę $\delta_{i,j}$ dla gałęzi łączącej węzeł s_i z węzłem s_j , oznaczono jako $\delta(s_i, s_j)$.

Wydruk 3.2: Algorytm rozwiązywania sieci przybliżonej

```

% ETAP 1                                     1
  S = ...                                     2
  for s ∈ S                                   3
    b(s) = [0, 1]                             4
  end                                         5
% ETAP 2                                     6
  for s ∈ SC                                 7
    b(s) = ...                                8
  end                                         9
  SI = ...                                   10
  for s ∈ SI                                 11
    b(s) = ...                                12
  end                                         13
  SII = ...                                  14
% ETAP 3                                     15
  rep = true                                  16
  while rep                                   17
    rep = false                               18
    for s ∈ SII                               19
      for sp ∈ pn(s)                          20
        if bmin(s) < bmin(sp) - δ(sp, s)  21
          bmin(s) = bmin(sp) - δ(sp, s)  22
          rep = true                           23
        end                                    24
      end                                    25
    end                                    26
  end                                    27
% ETAP 4                                     28
  rep = true                                  29
  while rep                                   30
    rep = false                               31
    for s ∈ SII                               32
      for sn ∈ nn(s)                          33
        if bmax(s) > bmax(sn) + δ(s, sn)  34
          bmax(s) = bmax(sn) + δ(s, sn)  35
          rep = true                           36
        end                                    37
      end                                    38
    end                                    39
  end                                    40
% ETAP 5                                     41
  for s ∈ SII                                 42

```

```

    if  $b_{min}(s) - b_{max}(s) > eps$                                 43
        break(" Błąd obliczeń")                                  44
    end                                                            45
end                                                                46

```

3.6.4. Elementy warunkowo sprzeczne

Należy zwrócić uwagę, że wprowadzenie omawianych modeli pozwala na współwystępowanie w jednej bazie wiedzy elementów warunkowo sprzecznych. Określenie ewentualnej sprzeczności jako sprzeczności warunkowej oznacza, że może się ona objawiać wyłącznie dla wybranych stanów sieci stwierdzeń.

Bibliografia

- [3.1] Beliakov G., Pradera A., Calco T. *Aggregation Functions: A Guide for Practitioners*. Springer Verlag, Heidelberg, 2007.
- [3.2] Cholewa W. Aggregation of fuzzy opinions – an axiomatic approach. *Fuzzy Sets and Systems*, 17:249–258, 1985.
- [3.3] Cholewa W. Multimodal statement networks for diagnostic applications. Sas P., Bergen B., redaktorzy, *Proceedings of the International Conference on Noise and Vibration Engineering ISMA 2010*, strony 817–830, Leuven, Belgium, September 20–22 2010. Katholieke Universiteit Leuven.
- [3.4] Heath L. S., Sioson A. A. Multimodal networks: Structure and operations. *IEEE/ACM Trans. on Computational Biology and Bioinformatics*, 6(2):321–332, 2009.
- [3.5] Heath L. S., Sioson A. A. Semantics of multimodal networks. *IEEE/ACM Trans. on Computational Biology and Bioinformatics*, 6(2):271–280, 2009.
- [3.6] Jensen V. J. *Bayesian Networks and Decision Graphs*. Springer, New York, 2002.
- [3.7] Koski T., Noble John M. *Bayesian Networks. An Introduction*. John Wiley and Sons, Ltd., 2009.
- [3.8] Sioson A.A. *Multimodal networks in biology*. Praca doktorska, Virginia Polytechnic Institute and State University, Blacksburg, Virginia, 2005.

Rozdział 4

Wybór środowiska oprogramowania

Tomasz ROGALA, Paweł CHRZANOWSKI

4.1. Wstęp

Niniejszy rozdział zawiera informacje na temat wyboru środowiska oprogramowania do budowy i rozwoju szkieletowego systemu doradczego MMNET. Szczególną uwagę poświęcono językom wyższego rzędu opartym na programowaniu funkcjonalnym. Wśród nich wytypowano język Matlab oraz język R. Posiadają one zbliżoną składnię oraz filozofię tworzenia oprogramowania opartą na wykorzystaniu macierzowego zapisu danych. Z uwagi na przeznaczenie tych środowisk szczegółowo przeszukano zasoby funkcji tych języków umożliwiające budowanie i stosowanie sieci stwierdzeń. W kolejnym kroku dla przydatnych funkcji umożliwiających budowanie i stosowanie sieci stwierdzeń przeprowadzono odpowiednie testy oraz weryfikację. Po przeprowadzonych badaniach wybrano środowisko oprogramowania. Szczegółowe wnioski związane z wyborem środowiska zawarto w podrozdziale 4.3.

4.2. Zestawienie i weryfikacja dostępnych funkcji i procedur umożliwiających budowanie i stosowanie sieci stwierdzeń

Wybór środowiska oprogramowania poprzedzono szczegółowym przeglądem ogólnodostępnych zasobów umożliwiających budowanie i stosowanie sieci probabilistycznych w środowisku Matlab i R. Niniejszy podrozdział zawiera informacje na temat weryfikacji działania wybranych bibliotek zebranych w trakcie realizacji zadania, których szczegółowy opis zawarto w raporcie [4.19], [4.20]. Podstawowym celem weryfikacji była ocena poprawności działania bibliotek oraz ocena ewentualnych ograniczeń ich stosowania w określonym kontekście np. ocena możliwości zastosowania odpowiedniego algorytmu wnioskowania dla sieci, w której zmienne reprezentowane są w postaci ciągłej i dyskretnej. Ze względu na różnorodne zagadnienia dotyczące m.in. składowania sieci, wizualizacji, rozwiązywania, testowania sieci itp., sprawdzanie poszczególnych funkcji i procedur realizowano osobno w ramach danego zagadnienia. W celu zapewnienia możliwości porównania sprawdzanych funkcji i procedur, w ramach tego samego zadania, starano się, aby było ono realizowane w oparciu o jednakowo przygotowane środowisko uruchomieniowe. Z uwagi na ograniczone możliwości wczytywania sieci zapisanych w powszechnie znanych formatach w różnych pakietach, w których stosowane są indywidualnie przyjęte formaty zapisu sieci, środowiska uruchomieniowe w ramach tego samego zagadnienia różniły się. W takim przypadku procedury i funkcje były najczęściej weryfikowane w oparciu o wybraną sieć zaimplementowaną w danej bibliotece.

Tam, gdzie było to możliwe działanie poszczególnych funkcji i procedur sprawdzano w oparciu o zbiór przykładów różnych typów sieci:

- małych (do kilku węzłów) dyskretnych sieci dwustanowych np.: `sprinkler.net`, `rats.net` [4.4],
- sieci dyskretnych o stosunkowo większej liczbie węzłów np.: `asia.net` (`chest_clinic`), `barley.net` [4.1],
- sieci hybrydowych z węzłami dyskretnymi i ciągłymi [4.1].

Szczegółowe uwagi dotyczące weryfikowanych funkcji i procedur zostały zamieszczone jako podsumowania poszczególnych zadań przez nie realizowanych. Uwagi ogólne, jak również informacje na temat licencji pakietów zamieszczono w końcowej części niniejszego raportu.

Budowanie i stosowanie tradycyjnych sieci stwierdzeń związane jest z koniecznością realizacji wielu zadań. Z uwagi na rozbudowany zakres zadania polegającego na wytypowaniu potencjalnych funkcji i procedur umożliwiających budowanie i stosowanie sieci stwierdzeń, zostało ono podzielone na szereg podzadań:

- składowanie sieci,
- transformacja formatów zapisu sieci stwierdzeń,
- edycja sieci stwierdzeń,
- wizualizacja sieci,
- rozwiązywanie sieci (w tym wyznaczaniem stanu węzłów swobodnych),
- trenowanie parametrów sieci,
- trenowanie struktury sieci,
- narzędzia analizy sieci,
- łączenie mniejszych sieci w większą sieć,
- zarządzanie stwierdzeniami w sieci stwierdzeń,
- inne procedury związane z budowaniem i stosowaniem sieci jak np.: generowanie przykładów i dynamiczne sieci przekonaniań.

4.2.1. Składowanie sieci

Weryfikacja dostępnych funkcji i procedur języka R i języka Matlab dotyczyła głównie możliwości zapisu, odczytu oraz analizy dostępnych formatów dla tych operacji powszechnie stosowanych dla sieci probabilistycznych. Weryfikowano kolejno procedury i funkcje języka R, a następnie Matlab. Dla każdego środowiska zestawiono osobne wnioski.

Język R:

readnet, savenet

Procedura umożliwia zapis i odczyt w formacie `*.net` (format Hugin).

Bibliografia: Pakiet `deal` [4.4].

saveXML(toGXL())

Procedura umożliwia zapis i odczyt *tylko grafów* w formacie `*.GXL`. Plik ten może być bezpośrednio transformowany do formatu XML.

Bibliografia: Pakiet `graph` [4.9].

loadHuginNet, saveHuginNet

Procedura umożliwia zapis i odczyt sieci w formacie `net`, który może być bezpośrednio zapisywany w formacie XML.

Bibliografia: Pakiet `graph` [4.8].

Podczas weryfikacji funkcji i procedur dotyczących składowania sieci sprawdzano czy możliwy jest zapis i odczyt sieci w tym również sieci hybrydowych.

Wnioski

- Podane funkcje **readnet**, **savenet** mogą być wykorzystane do zapisu i odczytu sieci utworzonych tylko w ramach pakietu [4.4] ze względu na własny format zapisu danych.
- Funkcje **readnet**, **savenet** nie działają prawidłowo z sieciami o formacie **.net (HUGIN)*. Zastosowanie funkcji **readnet**, **savenet** dla formatu **.net (HUGIN)* prawdopodobnie wymaga niewielkiej modyfikacji źródeł pakietu [4.4].
- Funkcja **saveXML(toGXL(gR))** umożliwia zapis struktury grafu w formacie GXL.
- Funkcje **loadHuginNet**, **saveHuginNet** prawidłowo realizują zapis sieci przekonań w formacie **.net (Hugin'a)* dla węzłów dyskretnych.
- Zapis i odczyt sieci hybrydowych przy użyciu funkcji **loadHuginNet**, **saveHuginNet** kończy się niepowodzeniem wynikającym z niezgodności rozmiarów tablic prawdopodobieństw warunkowych. W programie *Hugin* zapis węzłów dyskretnych odbywa się na dwa sposoby. Pierwszy z nich to opis prawdopodobieństw warunkowych za pomocą zbioru gaussowskich funkcji gęstości prawdopodobieństwa. Drugi sposób polega na dyskretyzacji węzła ciągłego i zapisania wartości CPT w węzle "interval node". W trakcie weryfikacji zbadano możliwość odczytu i zapisu sieci hybrydowych zapisanych zgodnie z pierwszym sposobem.

Matlab:

bif2bnt

Konwersja z formatu BIF do formatu BNT. Program umożliwia konwertowanie sieci zapisanych w formacie BIF do formatu stosowanego w pakiecie BNTtoolbox.

Bibliografia: Program bif2bnt [4.12]

Wnioski

- Konwerter bif2bnt działa prawidłowo. Konwerter ten udostępniany jest jedynie w postaci aplikacji on-lineowej na stronie [4.12].

4.2.2. Transformacja formatów zapisu sieci stwierdzeń

Stwierdzono brak gotowych funkcji i procedur do transformacji formatów zapisu sieci stwierdzeń w środowiskach Matlab oraz R. Możliwe jest opracowanie własnych procedur w oparciu o funkcje do składowania sieci opisanych powyżej w sekcji 4.2.1.

4.2.3. Edycja sieci stwierdzeń

Wyniki poszukiwań funkcji i procedur umożliwiających edycję sieci stwierdzeń zamieszczono poniżej.

Język R:

DAG

Funkcja do definiowania grafu skierowanego.

Bibliografia: Pakiet ggm [4.18].

adjMatrix

Funkcja transformacji macierzy krawędzi w macierz sąsiedztwa.

Bibliografia: Pakiet ggm [4.18].

allEdges

Funkcja wyszukuje zbiór krawędzi grafu.

Bibliografia: Pakiet ggm [4.18].

cliques

Funkcja znajduje kliki w grafie nieskierowanym.

Bibliografia: Pakiet ggm [4.18].

cmpGraph

Funkcja znajduje wzajemnie się uzupełniający graf w grafie nieskierowanym.

Bibliografia: Pakiet ggm [4.18].

insert

Wstawia połączenia pomiędzy węzłami.

Bibliografia: Pakiet deal [4.4].

network

Definiuje sieć Bayesowską.

Bibliografia: Pakiet deal [4.4].

node

Definiuje węzeł.

Bibliografia: Pakiet deal [4.4].

numbermixed

Zwraca liczbę możliwych sieci dla podanego zbioru węzłów.

Bibliografia: Pakiet deal [4.4].

prob

Definiuje prawdopodobieństwa warunkowe w węźle.

Bibliografia: Pakiet deal [4.4].

localprob

Zwraca wartość tablic CPT w węźle.

Bibliografia: Pakiet deal [4.4].

unique.networkfamily

Usuwa sieć już istniejącą ze zbioru.

Bibliografia: Pakiet deal [4.4].

query, tab, or

Funkcje służące do sieci przy bezpośrednim definiowaniu Tablic Prawdopodobieństw Warunkowych (CPT).

Bibliografia: Pakiet Grappa [4.17].

makeadj

Definiuje macierz sąsiedztwa określającą strukturę sieci.

Bibliografia: Pakiet Grappa [4.17].

vs

Określa zbiór wartości zmiennej.

Bibliografia: Pakiet Grappa [4.17].

fq

Pobiera obliczone wartości prawdopodobieństw dla węzła.

Bibliografia: Pakiet Grappa [4.17].

joint

Pobiera obliczone wartości prawdopodobieństw dla dwóch lub więcej zmiennych w węźle.
Bibliografia: Pakiet Grappa [4.17].

prop.evid

Wprowadza wartości do określonego węzła (jako *evidence*).
Bibliografia: Pakiet Grappa [4.17].

andtable

Generuje tablice prawdopodobieństw warunkowych na podstawie wyrażenia logicznego AND.
Bibliografia: Pakiet gRain [4.8].

ortable

Generuje tablice prawdopodobieństw warunkowych na podstawie wyrażenia logicznego OR.
Bibliografia: Pakiet gRain [4.8].

cptable

Definiuje tablice prawdopodobieństw warunkowych wg. $p(v|pa(v))$.
Bibliografia: Pakiet gRain [4.8].

compile

Weryfikuje niezależność sieci Bayesowskiej.
Bibliografia: Pakiet gRain [4.8].

compileCPT

Tworzy tablicę prawdopodobieństw warunkowych.
Bibliografia: Pakiet gRain [4.8].

querygrain

Zapytanie o niezależność sieci.
Bibliografia: Pakiet gRain [4.8].

grain

Definiuje sieć.
Bibliografia: Pakiet gRain [4.8].

Wnioski

- Pakiet ggn:[4.18]
 - brak możliwości definiowania Tablic Prawdopodobieństw Warunkowych (CPT),
 - pakiet ogranicza się jedynie do operacji na grafach reprezentowanych w postaci macierzy sąsiedztwa;
- Pakiet Grappa [4.17] umożliwia definiowanie i edycję sieci stwierżeń o węzłach dyskretnych.
- Pakiet gRain [4.8] umożliwia definiowanie i edycję sieci stwierżeń o węzłach dyskretnych.
- Pakiet deal [4.4] umożliwia definiowanie i edycję sieci stwierżeń o węzłach ciągłych i węzłach dyskretnych.
- Pakiety Grappa, gRain oraz deal umożliwiają edycję sieci stwierżeń, wybór jednego z nich zależy będzie od pozostałych funkcjonalności weryfikowanych i opisanych w ramach tego rozdziału.

Matlab:**dag**

Łączy węzły.

Bibliografia: Pakiet BNTtoolbox [4.3].

mk_bnet

Definiuje sieć.

Bibliografia: Pakiet BNTtoolbox [4.3].

tabular_CPD

Definiuje tablice gęstości prawdopodobieństw warunkowych (CPD) - dla węzłów ciągłych.

Bibliografia: Pakiet BNTtoolbox [4.3].

CPT

Definiuje tablice prawdopodobieństw warunkowych (CPT) - dla węzłów dyskretnych.

Bibliografia: Pakiet BNTtoolbox [4.3].

parents

Definiuje rodziców dla węzła.

Bibliografia: Pakiet BNTtoolbox [4.3].

Wnioski Biblioteka BNT umożliwia definiowanie i edycję sieci stwierdzeń o węzłach ciągłych i węzłach dyskretnych.

4.2.4. Wizualizacja sieci

Możliwość odpowiedniej wizualizacji grafu jest przydatnym elementem wspomagającym proces budowy sieci probabilistycznych, umożliwia łatwiejszą identyfikację relacji pomiędzy rozpatrywanymi zmiennymi. W związku z tym przeprowadzono analizę dostępnych funkcji w języku R i Matlab umożliwiających wizualizację grafów.

Język R:**drawnetwork**

Umożliwia użytkownikowi budowanie sieci przekonania na podstawie prostego interfejsu typu wskaż i kliknij. Nadaje się do wizualizacji mało złożonych sieci (od aut.).

Bibliografia: Pakiet deal [4.4].

genpicfile

Umożliwia wizualizację sieci z zastosowaniem grafiki wektorowej w formacie *pictex – graphs*.

Bibliografia: Pakiet deal [4.4].

plotmat

Wizualizacja prostych grafów na podstawie macierzy połączeń.

Bibliografia: Pakiet diagram [4.5].

giRaph

Wizualizacja grafów oraz ich edycja.

Bibliografia: Pakiet giRaph [4.7].

graph z zastosowaniem dodatkowego pakietu graphviz

Wizualizacja grafów oraz szereg funkcji związanych z budowaniem i edycją grafów.

Bibliografia: Pakiet graph [4.7].

plot.mathgraph

Wizualizacja prostych grafów w oknie programu R.

Bibliografia: Pakiet `mathgraph` [4.11].

plot.bn

Wizualizacja prostych grafów w oknie programu R.

Bibliografia: Pakiet `bnlearn` [4.2].

W trakcie weryfikacji działania funkcji i procedur dla potrzeb wizualizacji sieci sprawdzano możliwość wizualizacji struktury sieci zarówno dyskretnych jak i ciągłych.

Wnioski

- Pakiet *deal* umożliwia wyświetlanie jedynie prostych sieci bez wyróżnienia typów węzłów.
- Pakiet *diagram* posiada wiele opcji umożliwiających uzyskanie wizualnie atrakcyjnej postaci grafu, możliwa jest również wizualizacja grafów hybrydowych ale wymaga ona rozszerzenia polecenia `plotmat`.
- Pakiet *bnlearn* pozwala na prostą i czytelną wizualizację grafów (bez wyróżnienia typów węzłów).
- Pozostałe pakiety posiadają ubogie możliwości reprezentacji graficznej grafu.
- Procedura *genpicfile* nie nadaje się do bezpośredniego wykorzystania w celu opracowania oprogramowania MMNET, ponieważ wymaga użycia kompilatora TeX. Umożliwia wizualizację sieci z zastosowaniem grafiki wektorowej w formacie *pictex – graphs*. Zapis graficzny sieci przypomina składnię język LISP.

Matlab:**draw graph**

Wizualizacja niewielkich grafów.

Bibliografia: Pakiet `BNTtoolbox` [4.3].

adj2pajek2

Wizualizacja dużych struktur sieci w programie *pajek*.

Bibliografia: Program *pajek* [4.13].

W trakcie weryfikacji działania funkcji i procedur dla potrzeb wizualizacji sieci sprawdzano możliwość wizualizacji struktury sieci zarówno dyskretnych jak i ciągłych.

Wnioski

- Pakiet *BNT* pozwala na wizualizację tylko mało złożonych struktur grafów (bez wyróżnienia typów węzłów).
- Zastosowanie funkcji *adj2pajek* wymaga zastosowania aplikacji zewnętrznej w celu zilustrowania sieci.

4.2.5. Rozwiązywanie sieci

W trakcie weryfikacji funkcji i procedur obejmujących zagadnienia rozwiązywania (wnioskowania) sieci nie prowadzono szczegółowych testów dotyczących porównania wydajności oraz dokładności wyznaczania prawdopodobieństw posteriori dla algorytmów przybliżonych. Działania te nie zostały przeprowadzone dla języka *R* ze względu na to, że większość pakietów wykorzystuje ten sam algorytm wnioskowania (ang. minimum clique weight heuristic) znajdujący się w pakiecie *gRbase* oraz to, że niektóre z pakietów zawierają algorytmy wnioskowania w postaci

zewnątrznie dołączanych bibliotek, co może mieć istotne znaczenie przy ocenie wydajności tych algorytmów. Testowanie w języku *Matlab* wydajności jak również dokładności wyznaczania prawdopodobieństw posteriori dla algorytmów przybliżonych zostało szczegółowo opisane w pozycji [4.16]. Biorąc powyższe pod uwagę sprawdzenie funkcji i procedur obejmujących zagadnienie rozwiązywania sieci ograniczono do weryfikacji ich działania oraz przedstawienia przykładów środowisk uruchomieniowych. Szczegółowe wyniki przedstawiono w raporcie [4.19], [4.20].

Język R:

compile.grain

Utworzenie drzewa złączeń oraz potencjałów dla poszczególnych węzłów (oraz klik) metodą triangulacji.

Bibliografia: Pakiet gRain [4.8].

propagate.grain

Poszukiwanie równowagi sieci dla drzewa utworzonego poleceniem `compile.grain`.

Bibliografia: Pakiet gRain [4.8].

compile

Utworzenie drzewa złączeń.

Bibliografia: Pakiet gRappa [4.10].

equil

Poszukiwanie równowagi sieci dla drzewa złączeń.

Bibliografia: Pakiet gRappa [4.10].

Wnioski

- Podane funkcje w języku *R* realizujące wnioskowanie działają prawidłowo.
- Większość funkcji dotyczących zasadniczego algorytmu triangulacji opartych jest na zastosowaniu algorytmu *mcwh* (ang. minimum clique weight heuristic).
- W przypadku pakietu *gRappa* algorytm wnioskowania dostępny jest w postaci biblioteki *gRappa.dll*.

Matlab:

jtree_inf_engine

Implementacja algorytmu drzewa złączeń.

Bibliografia: Pakiet BNTtoolbox [4.3].

var_elim_inf_engine

Implementacja algorytmu eliminacji zmiennych.

Bibliografia: Pakiet BNTtoolbox [4.3].

enumerative_inf_engine

Algorytm bazujący na przeszukiwaniu wyczerpującym - dokładny dla sieci dyskretnych.

Bibliografia: Pakiet BNTtoolbox [4.3].

pearl_inf_engine

Algorytm *Perla* - dla polidrzew.

Bibliografia: Pakiet BNTtoolbox [4.3].

quickscore_inf_engine

Algorytm *quickscore* dla sieci o strukturze QMR.

Bibliografia: Pakiet BNTtoolbox [4.3].

likelihood_weighting_inf_engine

Algorytm ważonej wiarygodności - algorytm przybliżony.

Bibliografia: Pakiet BNToolbox [4.3].

gibbs_sampling_inf_engine

Algorytm oparty na próbkowaniu Gibbs'a - algorytm przybliżony.

Bibliografia: Pakiet BNToolbox [4.3].

belprop_inf_engine

Loopy belief propagation - algorytm przybliżony.

Bibliografia: Pakiet BNToolbox [4.3].

Wnioski

- Podane funkcje w języku *Matlab* działają prawidłowo.
- Biblioteka dotycząca funkcji rozwiązywania sieci w pakiecie *BNT* jest obszerna i bogata oraz pozwala na zastosowanie różnych algorytmów, które powstawały na przestrzeni lat, gdzie rozwijano sieci przekonań, można znaleźć algorytmy dedykowane dla określonych typów sieci, jak również szereg algorytmów dokładnych i przybliżonych dedykowanych dla różnych typów sieci.
- Badania porównawcze funkcji wnioskowania w sieci przekonań zaimplementowanych w języku *Matlab* zostały przedstawione w [4.16]. Algorytmy testowano zarówno pod kątem wydajności jak również dokładności uzyskanych wyników.
- W przypadku pakietu *gRappa* algorytm wnioskowania dostępny jest w postaci biblioteki *gRappa.dll*.

4.2.6. Trenowanie struktury sieci

Poszukiwano funkcji i procedur umożliwiających trenowanie struktury sieci.

Język R:**gs**

Uczenie struktury z ograniczeniami algorytmem Grow-Shrink.

Bibliografia: Pakiet bnlearn [4.2].

iamb

Uczenie struktury z ograniczeniami algorytmem Incremental Association.

Bibliografia: Pakiet bnlearn [4.2].

fast.iamb

Uczenie struktury z ograniczeniami algorytmem Fast Incremental Association.

Bibliografia: Pakiet bnlearn [4.2].

hc

Uczenie struktury z uwzględnieniem dodatkowych algorytmów strojenia Hill-Climbing.

Bibliografia: Pakiet bnlearn [4.2].

tabu

Uczenie struktury. Zmodyfikowany algorytm Hill-Climbing.

Bibliografia: Pakiet bnlearn [4.2].

compare

Porównuje dwie sieci Bayesowskie.

Bibliografia: Pakiet bnlearn [4.2].

Matlab:**learn_struct_mcmc**

Uczenie struktury sieci Bayesowskich z użyciem metody MCMC (Markov Chain Monte Carlo) lub lokalnego przeszukiwania.

Bibliografia: Pakiet BNTtoolbox [4.3].

learn_struct_K2, learn_struct_pdag_pc), learn_struct_pdag_ic_star

Uczenie z ograniczeniami struktury sieci (IC/PC oraz IC*/FCI).

Bibliografia: Pakiet BNTtoolbox [4.3].

Wnioski

- Biblioteka *BNT* środowiska *Matlab* zawiera dużą grupę specjalistycznych funkcji przeznaczonych do uczenia struktury sieci.
- Biblioteka *bnlearn* środowiska *R* jest przeznaczona do uczenia struktury sieci bayesowskich.
- Każda z powyżej przedstawionych bibliotek pozwala na realizację zadania uczenia struktury sieci bayesowskich w oparciu o różnorodne algorytmy, a wybór środowiska obliczeniowego decyduje o zastosowaniu jednej z nich.
- Wybór procedury uczenia jest zależny od typu rozpatrywanej sieci oraz jej stopnia złożoności.

4.2.7. Trenowanie parametrów sieci

Poszukiwano funkcji i procedur pozwalających na uczenie parametrów już zdefiniowanych struktur sieci zarówno o węzłach dyskretnych, o węzłach ciągłych jak i struktur mieszanych.

Język R:**learn**

Uczenie parametrów sieci o węzłach ciągłych i/lub węzłach dyskretnych na podstawie danych wejściowych.

Bibliografia: Pakiet deal [4.4].

jointprior

Obliczenie łącznego prawdopodobieństwa w węźle.

Bibliografia: Pakiet deal [4.4].

Matlab:**learn_params_em**

Uczenie MLE/MAP parametrów sieci metodą EM.

Bibliografia: Pakiet BNTtoolbox [4.3].

bayes_update_params

Sekwencyjne uczenie parametrów sieci Bayesowskich.

Bibliografia: Pakiet BNTtoolbox [4.3].

Wnioski

- Biblioteka *BNT* środowiska *Matlab* zawiera dużą grupę specjalistycznych funkcji przeznaczonych do uczenia parametrów sieci bayesowskich.

- Biblioteka *deal* środowiska *R* jest przeznaczona do uczenia parametrów sieci bayesowskich zarówno o węzłach ciągłych i/lub dyskretnych.
- Każda z powyżej przedstawionych bibliotek pozwala na realizację zadania uczenia parametrów sieci bayesowskich w oparciu o różnorodne algorytmy, a wybór środowiska obliczeniowego decyduje o zastosowaniu jednej z nich.
- Wybór procedury uczenia jest zależny od typu rozpatrywanej sieci oraz jej stopnia złożoności.

4.2.8. Narzędzia do analizy sieci

W ramach niniejszego podzadania rozpatrywano tylko specyficzne narzędzia do analizy sieci przekonanych. Pominięto typowe algorytmy z teorii grafów dotyczących ich opisu np. algorytm wyznaczania najdłuższej ścieżki w grafie.

Matlab:

calc_mpe, calc_mpe_bucket

Podaje największą wiarygodność węzła.

Bibliografia: Pakiet BNToolbox [4.3].

Wnioski

- Nie znaleziono wyspecjalizowanych narzędzi umożliwiających analizowanie sieci w środowisku *R*.
- Przedstawione powyżej narzędzia działają prawidłowo.
- Nie obejmują one zaawansowanych narzędzi dotyczących analizy kolizji, badania wrażliwości, oceny wartości (znaczenia) zawartych w sieci informacji ponieważ udostępniane nieodpłatnie pakiety nie zawierają takich narzędzi.

4.2.9. Inne: Generowanie przykładów

Język *R*:

rnetwork

Umożliwia generowanie przykładów.

Bibliografia: Pakiet *deal* [4.4].

simulate.grain

Umożliwia generowanie przykładów.

Bibliografia: Pakiet *gRain* [4.8].

simulate

Umożliwia generowanie przykładów.

Bibliografia: Pakiet *gRappa* [4.10].

Wnioski

Wymienione funkcje działają prawidłowo. Podczas weryfikacji nie sprawdzano możliwości generowania danych niepełnych.

4.2.10. Inne: Dynamiczne sieci przekonań

W tym podrozdziale wskazano i zweryfikowano wstępnie działanie pakietów umożliwiających budowanie dynamicznych sieci przekonań. Z uwagi na rozbudowany zakres tego zagadnienia weryfikację ograniczono jedynie do przykładów środowisk uruchomieniowych pozwalających na budowę sieci dynamicznych.

Język R:

pakiet G1DBN

Pakiet pozwalający na budowę i stosowanie dynamicznych sieci bayesowskich.
Bibliografia: Pakiet G1DBN [4.6].

Matlab:

wybrane funkcje pakietu BNT

Umożliwiają budowanie, uczenie i stosowanie dynamicznych sieci bayesowskich.
Bibliografia: Pakiet BNTtoolbox [4.3].

Wnioski

Przedstawione powyżej narzędzia dla środowiska R oraz Matlab działają prawidłowo.

4.2.11. Podsumowanie

W niniejszym podrozdziale przedstawiono jedynie te funkcje i procedury, które zostały szczegółowo opisane w dokumentacji poszczególnych pakietów. Opisano funkcje i procedury, które są stosowane dla potrzeb sieci przekonań, jak również te, które są używane w teorii grafów z ograniczeniem do tych, które zdaniem autorów mogą okazać się przydatne podczas opracowywania oprogramowania. Szczegółowe wyniki dotyczące weryfikacji poszczególnych funkcji i procedur zamieszczono w [4.19], [4.20]. Raporty te obejmują również przykłady zastosowania powyższych funkcji.

Na podstawie analizy dostępnych narzędzi pod kątem rozpatrywanych w niniejszym raporcie zagadnień dla potrzeb stosowania sieci probabilistycznych stwierdzono, że:

- najwięcej funkcji i procedur jest dostępnych dla środowiska *R*,
- spora część dostępnych funkcji i procedur w języku *R* jest w fazie ciągłego rozwoju i wymaga dalszej szczegółowej analizy,
- nie znaleziono nowych pakietów w języku *Matlab*, które w znacznym stopniu rozszerzałyby możliwości powszechnie stosowanego pakietu *BNTtoolbox*,
- przebadane funkcje i procedury w większości działają poprawnie i wymagają ewentualnie niewielkich modyfikacji. Te, które nie działają zgodnie z opisem przedstawionym w instrukcji użytkownika pakietu zostały opatrzone stosownym komentarzem. Dotyczy to głównie funkcji i procedur rozwijanych w ramach języka *R*,
- zadania związane z łączeniem sieci w większą sieć nie są realizowane w weryfikowanych środowiskach,
- zarządzanie stwierdzeniami w sieci stwierdzeń, może być jedynie realizowane na poziomie edycji treści stwierdzeń w postaci nazw węzłów,

- dynamiczny rozwój środowiska *R* i jego pakietów powoduje, że funkcjonalność oraz jakość działania kolejnych wersji przedstawionych pakietów staje się coraz lepsza.

Poniżej zamieszczono informacje na temat licencji wykorzystywanych przez poszczególne pakiety:

- Pakiety *bnlearn*, *deal*, *diagram*, *g1dbn*, *gRain*, *igraph* mogą być wykorzystywane pod warunkiem zachowania licencji GNU GPL => 2.
- Pakiet *graph* może być wykorzystany pod warunkiem zachowania licencji *Artistic 2.0* co oznacza, że udostępniany pakiet może być kopiowany i rozpowszechniany w wersji pierwotnej, natomiast rozpowszechnianie zmodyfikowanej wersji jest tylko możliwe w przypadku jej wersji źródłowej wraz z wyraźnym wskazaniem tych elementów które zostały zmodyfikowane. Szczegóły licencji można znaleźć m.in. na <http://www.opensource.org/licenses/artistic-license-2.0.php>.
- *mathgraph* może być swobodnie rozpowszechniany i stosowany dla celów niekomercyjnych.
- *BNT* toolbox może być stosowany pod warunkiem zachowania warunków licencji GNU GPL.
- *gRappa* jest wolny od opłat dla edukacyjnych i niekomercyjnych celów badawczych.

4.3. Wybór języka programowania

Oprogramowanie Matlab oraz język *R* posiadają wiele wspólnych zalet umożliwiających rozwijanie oprogramowania MMNET w ramach tzw. programowania funkcjonalnego przy pomocy przygotowanych, ogólnie udostępnionych pakietów. Obydwa środowiska umożliwiają szybkie tworzenie oraz dużą łatwość odczytu i debugowania kodu tworzonego oprogramowania, w tym również możliwość opracowywania wydajnych funkcji i procedur napisanych w języku *C*.

Na podstawie przeglądu dostępnych funkcji i procedur umożliwiających budowanie i stosowanie sieci probabilistycznych stwierdzono, że zarówno Matlab jak i *R* umożliwiają budowanie modeli opartych na grafach skierowanych. W przypadku języka Matlab dominującym pakietem narzędzi, które mogą być w tym celu stosowany jest pakiet *BNT* (Bayesian Network Toolbox). W przypadku języka *R*, to zbiór pakietów *gRain*, *bnlearn*, *igraph* *gRBase* oraz inne. Rozbudowana funkcjonalność pakietu *BNT* języka Matlab dotyczy głównie grafów kierunkowych i nie posiada podobnych możliwości dla grafów nieskierowanych oraz łańcuchów. Ograniczone są również możliwości związane z interfejsem dotyczącym budowania grafów. Wśród pakietów języka *R* funkcjonalność dotycząca modeli opartych na grafach jest bardziej rozbudowana; począwszy od algorytmów z teorii grafów, po implementację grafów skierowanych, nieskierowanych i łańcuchów, jednakże i tutaj brak jest odpowiednio rozbudowanych reprezentacji graficznych grafów i ich zawartości.

Na podstawie analizy dostępności oraz weryfikacji działania funkcji i procedur umożliwiających budowanie i stosowanie sieci stwierdzono podjęto decyzję o wyborze języka programowania. Podczas wyboru wzięto pod uwagę również istotne różnice pomiędzy środowiskiem *R* i Matlab. Oprócz pozytywnych cech wspólnych dla obu środowisk występują istotne różnice. Do najważniejszych można zaliczyć:

- Język *R* w przeciwieństwie do języka Matlab jest nieodpłatnie rozpowszechnianym oprogramowaniem, jest rozbudowywany i rozpowszechniany na prawach licencji GPL-GNU, co umożliwia prowadzenie dalszych prac na zasadzie otwartego projektu.
- Język *R* jest skierowany głównie do społeczności zajmującej się modelami statystycznymi, do których należą również probabilistyczne modele graficzne.

Ostatecznie zdecydowano o wyborze języka R jako środowiska do budowy i rozwoju oprogramowania MMNET.

Należy podkreślić, że z uwagi na to, iż język R jest oprogramowaniem wolnym, ma on bardzo aktywną grupę użytkowników i twórców pakietów wykorzystujących to oprogramowanie zarówno dla celów zawodowych jak i naukowych. W obecnym kształcie język R posiada bardzo rozbudowaną funkcjonalność dotyczącą obliczeń statystycznych oraz w mniejszym stopniu wielu innych zagadnień z innych dziedzin takich jak: przetwarzanie sygnałów, bioinformatyka, analiza systemów, itd. Ponadto język R, podobnie jak Matlab, posiada również wiele pakietów umożliwiających integrację oprogramowania z innym oprogramowaniem (np. Excel), jak również wymiany danych z innymi systemami za pomocą formatu danych XML oraz przechowywania i manipulowania danymi zawartymi w bazach danych za pomocą języka SQL. Język R nie traci wiele na integralności poszczególnych pakietów z uwagi na to, że jest oprogramowaniem wolnym. Posiada wytyczone standardy tworzenia nowych pakietów oraz opracowywania dokumentacji, które rozpowszechniane są z pomocą repozytorium CRAN [4.14].

Można również zauważyć, że główne ośrodki naukowe zajmujące się rozwijaniem modeli graficznych są obecnie coraz bardziej zainteresowane integracją działań wielu ośrodków na rzecz utworzenia wspólnego pakietu umożliwiającego budowanie i stosowanie modeli graficznych. Można wymienić tutaj inicjatywę pakietu *gR* [4.15] w ramach języka R. Motywacją do wspólnych działań było w tym przypadku wiele opracowań wolnego oprogramowania dla modeli sieci, przy czym większość z nich posiadała ograniczoną funkcjonalność. Należy podkreślić, że biorąc pod uwagę rachunek zysków i strat również inne zespoły podjęły jak się wydaje słuszną decyzję dotyczącą rozwijania własnych zainteresowań naukowych w tej dziedzinie z zastosowaniem języka R.

Bibliografia

- [4.1] Strona internetowa firmy Hugin. Internet: <http://developer.hugin.com/documentation/samples/>, 03.2010.
- [4.2] Package bnlearn. Internet: <http://www.bnlearn.com/>, 11.2009.
- [4.3] Package bntoolbox. Internet: <http://people.cs.ubc.ca/~murphyk/Software/BNT>, 11.2009.
- [4.4] Package deal. Internet: <http://cran.r-project.org/web/packages/deal/index.html>, 11.2009.
- [4.5] Package diagram: Functions for visualising simple graphs (networks), plotting flow diagrams. Internet: <http://www.ci.tuwien.ac.at/gR/>, 11.2009.
- [4.6] Package gldbn. Internet: <http://cran.r-project.org/web/packages/G1DBN/index.html>, 11.2009.
- [4.7] Package giraph. Internet: <http://www.math.aau.dk/~dethlef/giRaph>, 11.2009.
- [4.8] Package grain. Internet: <http://genetics.agrsci.dk/~sorenh/public/R/gRainweb/>, 11.2009.
- [4.9] Package graph. Internet: <http://www.ci.tuwien.ac.at/gR/>, 11.2009.
- [4.10] Package grappa. Internet: <http://www.stats.bris.ac.uk/~peter/Grappa/>, 11.2009.
- [4.11] Package mathgraph. Internet: <http://www.ci.tuwien.ac.at/gR/>, 11.2009.
- [4.12] Program bif2bnt. Internet: <http://www.digitas.harvard.edu/~ken/bif2bnt>, 11.2009.

-
- [4.13] Program pajek. Internet: <http://vlado.fmf.uni-lj.si/pub/networks/pajek/>, 11.2009.
- [4.14] Comprehensive repository network. Internet: <http://cran.r-project.org>, 11.2010.
- [4.15] Package for graphical models. Internet: <http://cran.r-project.org/gR>, 11.2010.
- [4.16] Chrzanowski P., Rogala T. Comparison of belief network inference algorithms. Burczyński T., Cholewa W., Moczulski W., redaktorzy, *Recent developments in artificial intelligence methods*, strony 35–42, Gliwice, 2007. Silesian University of Technology.
- [4.17] Green Peter J. *Grappa: R functions for probability propagation*, 2005.
- [4.18] Marchetti Giovanni M., Drton Mathias. *ggm: Graphical Gaussian Models*, 2006.
- [4.19] Rogala T. Biblioteka funkcji i procedur umożliwiających budowanie i stosowanie sieci stwierdzeń-czesc II. Raport wewnętrzny MMN-04-012, Politechnika Śląska, Katedra Podstaw Konstrukcji Maszyn, Gliwice, 2010. (wersja 1.10).
- [4.20] Rogala T. Weryfikacja i testowanie funkcji i procedur umożliwiających budowanie i stosowanie sieci stwierdzeń-czesc II. Raport wewnętrzny MMN-04-022, Politechnika Śląska, Katedra Podstaw Konstrukcji Maszyn, Gliwice, 2010. (wersja 1.10).

Rozdział 5

Pakiet MMNET

Paweł CHRZANOWSKI

Realizując przyjętą koncepcję szkieletowego systemu doradczego, opracowano w środowisku R pakiet MMNET. Uznano, że technika programowania obiektowego w języku R będzie najlepiej nadawała się do opracowania kodów źródłowych tego systemu. Ponadto technika obiektowa umożliwia łatwy i elastyczny rozwój oprogramowania pod względem funkcjonalności jak też przez wielu niezależnych programistów.

5.1. Opis ogólny

Przyjęto, iż pakiet MMNET będzie zawierał zbiór klas, metod oraz procedur wraz z dokumentacją i przykładami. Wyodrębniono następujące funkcjonalności pakietu MMNET:

- definiowanie i edycja stwierdzeń prostych i stwierdzeń złożonych,
- definiowanie, edycja i zarządzanie zbiorem stwierdzeń (tezauresem),
- definiowanie, edycja i zarządzanie modelem wieloskalowej sieci stwierdzeń,
- definiowanie i edycja warstwy modelu,
- łączenie warstw poprzez definiowanie węzłów wspólnych,
- obliczanie sieci w zależności od jej typu (np. Bayesa, Markova, itp.),
- agregacja wyników pomiędzy warstwami modelu w oparciu o przyjęte algorytmy,
- wnioskowanie z zastosowaniem danych zewnętrznych,
- interfejs użytkownika.

Dla potrzeb realizacji niektórych funkcjonalności wykorzystano istniejące i szeroko dostępne pakiety środowiska R, które służą między innymi do wymiany danych, realizacji procesu wnioskowania oraz do działania opracowanego graficznego interfejsu użytkownika (GUI).

Przyjęto, że podstawą formą zapisu wiedzy są stwierdzenia proste, które mogą być elementami stwierdzeń złożonych. Zbiór stwierdzeń prostych oraz stwierdzeń złożonych stanowi słownik stwierdzeń tzw. tezaurus. Zarządzanie stwierdzeniami realizowane jest poprzez opracowany przez autora plik wymiany danych w formacie XML. Format ten znacznie ułatwia możliwość podglądu jak i edycję tezaury w dowolnym programie do obsługi plików tekstowych.

W budowanym modelu tezaurus jest elementem bazowym. Model, podobnie jak tezaurus jest zapisywany do formatu XML, gdzie w modelu zawarty jest jedynie stosowany fragment tezaury bazowego oraz opis struktury modelu. Struktura modelu zawiera definicję warstw,

gdzie każda warstwa to odrębna sieć zbudowana na zbiorze stwierdzeń tezaury bazowego. Szczególnym przypadkiem modelu wielowarstwowego jest model jednowarstwowo.

Pakiet MMNET umożliwia wykonywanie obliczeń zdefiniowanego modelu dla każdej warstwy osobno oraz agregację uzyskanych wyników w poszczególnych warstwach. Wyniki agregacji zamieszczane są w tzw. warstwie zerowej.

Proces wnioskowania może być realizowany poprzez wprowadzanie danych z użyciem graficznego interfejsu użytkownika bądź poprzez zbiór danych zewnętrznych zapisanych w pliku tekstowym.

5.2. Wymiana danych w formacie XML

Dla potrzeb archiwizacji oraz wymiany danych zdefiniowano format pliku oparty na standardzie XML (<http://www.w3.org/XML/>). Powodem zastosowania standardu XML była jego prostota implementacji, przejrzystość pliku źródłowego, dzięki czemu możliwy jest jego podgląd oraz edytowanie z użyciem dowolnego edytora tekstu. Szczególną zaletą przyjęcia standardu XML jest możliwość rozszerzania formatu o nowe elementy wraz z rozwojem systemu MMNET.

Jako podstawy element struktury formatu pliku przyjęto tezaurus (klucz: *THESAURUS*) składający się z definicji stwierdzeń (klucz: *STATEMENT*). Każde stwierdzenie zawiera między innymi typ stwierdzenia (klucz: *STMTYPE*), identyfikator stwierdzenia (klucz: *ID*), treść stwierdzenia (klucz: *CONTENT*) oraz definicję nazw wartości (klucz: *VALNAMES*). Fragment pliku zawierającego przykładową definicję tezaury przedstawiono poniżej.

```
<?xml version="1.0"?>
<MMNET VERSION="1.0">
  <STATICPROPERTIES>
    <source value="Core ver. 0.5, Pawel Chrzanowski (KPKM)"/>
    <date value="Mon Dec 6 13:36:44 2010"/>
  </STATICPROPERTIES>
  <THESAURUS stmts="5" notice="Write notice, please" title="Thesaurus ..." idThesa="0">
    <STATEMENT>
      <STMTYPE>CStmtValMBelief_TR</STMTYPE>
      <ID>10</ID>
      <MODE>1</MODE>
      <DESCRIPTION>belief value</DESCRIPTION>
      <LAYERS>0</LAYERS>
      <SHORTNAME>none</SHORTNAME>
      <CONTENT>Stwierdzenie nr 1</CONTENT>
      <HELP>Pomoc dla stwierdzenia nr 1</HELP>
      <VSTMTID></VSTMTID>
      <VALNAMES>
        <VALNAME>yes</VALNAME>
        <VALNAME>no</VALNAME>
      </VALNAMES>
    </STATEMENT>
    <STATEMENT>
      <STMTYPE>CStmtValMBelief_TR</STMTYPE>
      <ID>20</ID>
      <MODE>1</MODE>
      <DESCRIPTION>belief value</DESCRIPTION>
      <LAYERS>0</LAYERS>
      <SHORTNAME>none</SHORTNAME>
      <CONTENT>Stwierdzenie nr 2</CONTENT>
      <HELP>Pomoc dla stwierdzenia nr 2</HELP>
      <VSTMTID></VSTMTID>
      <VALNAMES>
        <VALNAME>yes</VALNAME>
        <VALNAME>no</VALNAME>
      </VALNAMES>
    </STATEMENT>
    .....
  </THESAURUS>
</MMNET>
```


Bazując na definicji tezaurusu, opracowano format zapisu modelu MMNET w standardzie XML. Przyjęto, że format ten będzie zawierał tezaurus o strukturze opisanej powyżej oraz definicję modelu. W strukturze pliku, definicja modelu (klucz: *NETWORK*) zawiera między innymi: identyfikator sieci (klucz: *IDNET*), nazwę modelu (klucz: *NAME*) oraz definicje warstw (klucz: *LAYERS*). W definicji warstwy (klucz: *LAYER*) zamieszczono listę węzłów (klucz: *NODES*) w postaci identyfikatorów stwierdzeń (patrz: identyfikator stwierdzenia), definicje krawędzi (klucz: *EDGES*) wraz z parametrami (klucz: *ARC*) zawierającymi identyfikatory węzłów, zwrot krawędzi (1- od N1 do N2, 2- od N2 do N1, 3-zwrot nieokreślony) oraz przypisane do węzłów tablice wartości (klucz: *NODETABLES*). Fragment pliku zawierającego przykładową definicję modelu zamieszczono poniżej.

```
<?xml version="1.0"?>
<MMNET VERSION="1.0">
  <STATICPROPERTIES>
    <source value="Core ver. 0.5, Pawel Chrzanowski (KPKM)"/>
    <date value="Mon Dec 6 13:40:04 2010"/>
  </STATICPROPERTIES>
  <THESAURUS stmts="5" notice="Write notice, please" title="Thesaurus ..." idThesa="0">
    <STATEMENT>
      .....
    </STATEMENT>
  </THESAURUS>
  <NETWORK>
    <IDNET>1</IDNET>
    <NAME>CNetBayes</NAME>
    <NOTICE>none</NOTICE>
    <LAYERS nrLayers="2">
      <LAYER>
        <IDLAYER>1</IDLAYER>
        <NOTICE>none</NOTICE>
        <WEIGHT>1</WEIGHT>
        <NODES>10 20 70 40</NODES>
        <EDGEVAL></EDGEVAL>
        <EDGES nrEdges="4">
          <ARC N1="10" N2="20" DIRECTION="1"/>
          <ARC N1="10" N2="70" DIRECTION="1"/>
          <ARC N1="20" N2="40" DIRECTION="1"/>
          <ARC N1="70" N2="40" DIRECTION="1"/>
        </EDGES>
        <NODETABLES nrnodeTables="4">
          <NODETABLE node="10" nrows="1">
            <ROW>50 50</ROW>
          </NODETABLE>
          <NODETABLE node="20" nrows="2">
            <ROW>10 90</ROW>
            <ROW>50 50</ROW>
          </NODETABLE>
          <NODETABLE node="70" nrows="2">
            <ROW>80 20</ROW>
            <ROW>70 30</ROW>
          </NODETABLE>
          <NODETABLE node="40" nrows="4">
            <ROW>99 1</ROW>
            <ROW>90 10</ROW>
            <ROW>90 10</ROW>
            <ROW>0 100</ROW>
          </NODETABLE>
        </NODETABLES>
      </LAYER>
      <LAYER>
        .....
      </LAYER>
    </LAYERS>
  </NETWORK>
</MMNET>
```

5.3. Wymagania

Ze względu na dynamiczny rozwój środowiska R i nie zawsze aktualnych pakietów dodatkowych stosowanych przez pakiet MMNET podjęto decyzję o użyciu wyłącznie wersji środowiska R 2.10.1 (<http://www.r-project.org/>). Zastosowane pakiety dodatkowe niezbędne do działania systemu MMNET to:

- biblioteka GTK+ (<http://www.gtk.org/>).
- pakiety środowiska R:
 - gWidgets [5.11],
 - gWidgetsRGtk2 [5.7],
 - igraph [5.4],
 - linprog [5.2],
 - matlab [5.3],
 - gRain [5.5],
 - bnlearn [5.8],
 - BMN [5.6],
 - XML [5.10].

5.4. Budowanie pakietu

Środowisko R umożliwia budowanie pakietów w oparciu o zbiór własnych kodów źródłowych, dzięki czemu możliwe jest łatwe dystrybuowanie pakietów. W celu wygenerowania pakietu instalacyjnego dla środowiska R (w tym przypadku) w systemie Windows, należy uprzednio zainstalować odpowiednie oprogramowanie oraz przygotować system Windows. Wszelkie szczegółowe informacje dotyczące budowania i weryfikacji pakietu dla środowiska R zostały opisane w <http://win-builder.r-project.org>.

Przygotowanie systemu Windows wymaga starannej realizacji działań. W tym celu należy:

1. Zapoznać się z dokumentacją opisującą narzędzie dla Windows (<http://cran.r-project.org/doc/manuals/R-admin.html#The-Windows-toolset>) w podręczniku administratora R.
2. Przejść na stronę (<http://www.murdoch-sutherland.com/Rtools/>)
3. Pobrać i zainstalować na komputerze pakiet **Rtools.exe**, odpowiedni do wersji środowiska R. Pakiet ten zawiera:
 - zbiór narzędzi uruchamianych z linii poleceń,
 - Perl,
 - kompilator MinGW,
4. Zainstalować dodatkowo:
 - InnoSetup installer (<http://jrsoftware.org/>)
 - MikTeX (<http://www.miktex.org/>),
 - Microsoft HTML Help Workshop (<http://msdn.microsoft.com/en-us/library/ms669985.aspx>) (nie jest wymagane),

5. Zweryfikować czy w zmiennej systemowej **PATH** istnieje wpis do:
`c:/Rtools/bin;c:/Rtools/perl/bin;c:/Rtools/MinGW/bin;c:/R/bin`
 (Uwaga: ścieżki należy podać zgodnie z lokalizacją oprogramowania).
6. Dokonać wpisu w **PATH** (patrz: Panel Sterowania – > System – > Zaawansowane ustawienia systemu – > Zmienne środowiskowe)
7. Niepoprawne wpisy w **PATH** są przyczyną nieprawidłowego działania procedury generowania i weryfikacji pakietu.
8. zrestartować komputer.

Następnie można przystąpić do przygotowania źródłowej wersji pakietu. W tym celu należy zdefiniować bazową strukturę folderów (tab.5.1) oraz plik *DESCRIPTION* w formacie tekstowym [5.9] zawierający podstawowe parametry użyte podczas generowania pakietu.

Tab. 5.1: Struktura folderów wersji źródłowej pakietu MMNET

Folder	Zawartość	Uwagi
główny	plik DESCRIPTION	podstawowy plik, zawiera parametry kompilacji
R	pliki źródłowe pakietu	*.R
man	pliki źródłowe dokumentacji	*.rd
data	pliki z przykładowymi danymi	np. tezaurus, sieć, *.XML

Budowanie pakietu w wersji binarnej należy przeprowadzić wg następującej kolejności:

- przygotować wersję źródłową w folderze np. `c:/Moje_Pakiety/MMNET`,
- uruchomić wiersz poleceń systemu Windows,
- przejść do katalogu `c:/Moje_Pakiety/`
- wpisać polecenie: `RCMD build --binary MMNET`
- zostanie wygenerowany pakiet o nazwie ustalonej w pliku DESCRIPTION np. `MMNET_0.5.2`

Weryfikację poprawności zbudowanego pakietu należy:

- przejść do katalogu `c:/Moj_Pakiety/`
- wpisać polecenie: `RCMD check MMNET`
- zweryfikować treść pliku log zawierającego wyniki weryfikacji,
- w przypadku powstania ostrzeżeń, starać się je wyeliminować wprowadzając w wersji źródłowej ewentualne poprawki i ponownie wykonać budowanie i weryfikację pakietu.

5.5. Numerowanie wersji

Przyjęto, że numeracja wersji oprogramowania będzie składała się z trzech poziomów, gdzie każdy zapisywany jest liczbą całkowitą od zera. Poziomy te oddzielone są kropką wg formatu `< poziom1.poziom2.poziom3 >` np. 0.5.1. Pierwszy poziom oznacza numer główny wersji i dotyczy będzie znacznych zmian w oprogramowaniu (wstępna oznaczona jest cyfrą zero). Drugi poziom oznacza podwersję zawierającą zmiany, będące rozszerzeniem wersji poprzedniej. Natomiast poziom trzeci dotyczy niewielkich (kosmetycznych) zmian wersji poprzedniej np. poprawiona stabilność działania pakietu.

5.6. Instalacja i pierwsze uruchomienie

Instalację pakietu MMNET należy przeprowadzić z poziomu środowiska R wg następującej procedury:

1. Uruchomić środowisko R (w wersji GUI).
2. Zainstalować pakiet MMNET z lokalnych zasobów (*Packages – > Install package(s) from local zip files ...*) - nie wolno zmieniać nazwy pakietu.
3. Załadować pakiet MMNET z menu *Packages– >Load package ...*
4. W linii poleceń środowiska R wpisać polecenie uruchamiające interfejs graficzny pakietu: **MMNET_gui()**. Jeżeli dodatkowe pakiety wskazane powyżej oraz biblioteka GTK+ nie są zainstalowane, wówczas rozpocznie się automatyczna instalacja tych pakietów ze wskazanego repozytorium środowiska R. Po instalacji GTK+ należy zrestartować komputer, a następnie ponownie załadować (aktywować) bibliotekę MMNET oraz uruchomić interfejs poleceniem **MMNET_gui()**.
5. W przypadku problemów z automatycznym zainstalowaniem dodatkowych pakietów, można przeprowadzić ich ręczną instalację z dowolnego repozytorium (*Packages – > Install package(s)*).

5.7. Aktualizacja pakietu

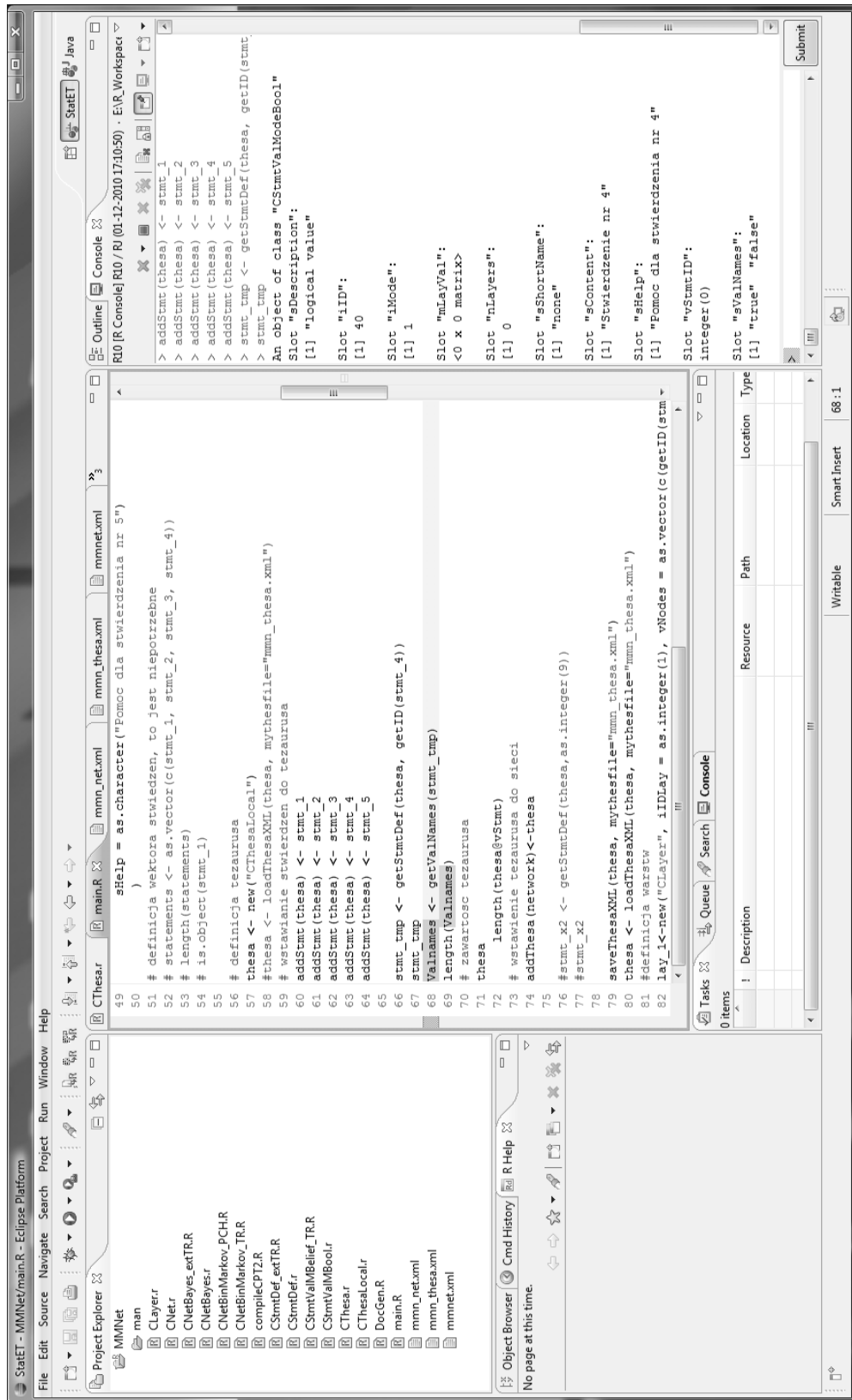
W przypadku aktualizacji pakietu MMNET, w pierwszej kolejności należy:

- Przeprowadzić deinstalację poprzedniej wersji pakietu MMNET (polecenie: *remove.packages("MMNET",lib=.libPaths())*).
- Zrestartować R (przy zamykaniu środowiska R, nie zapisywać zmiennych środowiskowych).
- Uruchomić środowisko R i zainstalować nową wersję pakietu MMNET z lokalnych zasobów (*Packages – > Install package(s) from local zip files ...*).
- Załadować pakiet MMNET z menu *Packages– >Load package ...*

5.8. Stosowanie pakietu MMNET bez GUI

Pakiet MMNET może być stosowany bez użycia własnego interfejsu graficznego. Opracowany interfejs graficzny jest jedynie narzędziem ułatwiającym konstruowanie modeli wielowarstwowej sieci stwierdzeń MMNET. Po instalacji pakietu nie ma konieczności uruchamiania interfejsu graficznego. Dostępne są wszystkie metody i klasy umożliwiające budowanie zbioru stwierdzeń, modelu sieci, obliczanie modelu, wnioskowanie oraz odczyt/zapis modelu czy słownika stwierdzeń z/do pliku formatu XML. Można również rozszerzać funkcjonalność pakietu MMNET czy też stosować go we własnych pakietach.

Wygodnym narzędziem znacznie ułatwiającym rozwój oprogramowania w środowisku R jest platforma Eclipse (<http://www.eclipse.org/platform>) [5.1], która umożliwia integrację ze środowiskiem R. Zaletą tej platformy jest łatwość pełnej integracji ze środowiskiem R i oferowanie typowych narzędzi programistycznych jak edycja kodu, debugowanie, system pomocy środowiska R, budowanie dokumentacji, budowanie pakietów i wiele innych. Przykład użycia platformy Eclipse zintegrowanej ze środowiskiem R przedstawiono na rysunku 5.1.



Rys. 5.1: Okno platformy Eclipse zintegrowanej ze środowiskiem R

Bibliografia

- [5.1] Eclipse platform. Dostępny w Internecie: <http://www.eclipse.org/platform>.
- [5.2] Arne Henningsen A. Linprog: Linear programming // optimization. Dostępny w Internecie: cran.r-project.org/web/packages/linprog/linprog.pdf.
- [5.3] Bengtsson H., Riedy J. R.matlab: Read and write of mat files together with r-to-matlab connectivity. Dostępny w Internecie: cran.r-project.org/web/packages/R.matlab/R.matlab.pdf.
- [5.4] Csardi G. Network analysis and visualization. Dostępny w Internecie: cran.r-project.org/web/packages/igraph/igraph.pdf.
- [5.5] Hřjsgaard Sřren. gRain: Graphical independence networks. Dostępny w Internecie: genetics.agrsci.dk/~sorenh/public/R/gRainweb/, 2010.
- [5.6] Hoefling Holger. BMN: The pseudo-likelihood method for pairwise binary markov networks, 2010.
- [5.7] Lawrence Michael, Verzani John. gWidgetsRGtk2: Toolkit implementation of gWidgets for RGtk2. Dostępny w Internecie: <http://cran.r-project.org/web/packages/gWidgetsRGtk2/index.html>, 2010.
- [5.8] Scutari Marco. Learning bayesian networks with the bnlearn R package, 2010.
- [5.9] Team R Development Core. Writing R Extensions. Dostępny w Internecie: cran.r-project.org/doc/manuals/R-exts.pdf, Version 2.12.0 (2010-10-15).
- [5.10] Temple Lang Duncan. XML: Tools for parsing and generating XML within R and S-Plus. Dostępny w Internecie: cran.r-project.org/web/packages/XML/index.html, Version 3.2-0 (2010-10-05).
- [5.11] Verzani John. gWidgets: gWidgets API for building toolkit-independent, interactive GUIs. Dostępny w Internecie: cran.r-project.org/web/packages/gWidgets/index.html, 2010.

Rozdział 6

Interfejs użytkownika systemu MMNET

Marcin AMAROWICZ

6.1. Wstęp

W celu ułatwienia pracy z pakietem MMNET dołączono do niego interfejs graficzny. Umożliwia on definiowanie różnego typu stwierdzeń, łączenie ich w słowniki stwierdzeń, budowę oraz modyfikację wielowarstwowych modeli diagnostycznych a także wykonywanie obliczeń na zdefiniowanych wcześniej modelach bez konieczności posługiwania się komendami języka R, w którym to został napisany pakiet MMNET. Przyjęto, że każda z operacji wykonywana za pomocą interfejsu będzie odbywała się z wykorzystaniem osobnej zakładki związanej tylko z jednym zagadnieniem. Dzięki takiemu rozwiązaniu możliwa jest stosunkowo łatwa rozbudowa interfejsu o nowe funkcje. Do wykonania interfejsu graficznego wykorzystano pakiet *gWidgets* [6.6]. Dodatkowe informacje związane z tworzeniem interfejsów graficznych z wykorzystaniem tego pakietu można znaleźć w [6.4, 6.5]. Do wizualizacji warstw wykorzystano pakiet *igraph*, jego opis można znaleźć w [6.3], a dodatkowe informacje na [6.1].

6.2. Menu oraz zakładka główna

Na rys. 6.1 pokazano widok interfejsu użytkownika pakietu MMNET w wersji 0.5.8. Główna zakładka (nazywana również zakładką startową) składa się z dwóch ramek. W pierwszej z nich znajdują się ogólne informacje o pakiecie. Natomiast druga z ramek jest zbiorem przycisków, za pomocą których obsługujemy interfejs. Dostępność poszczególnych przycisków uwarunkowana jest możliwościami do wykonania operacjami.

Opis przycisków znajdujących się na zakładce startowej:

- *Tezaurus*
 - *Nowy* - definiowanie nowego tezaurusa,
 - *Otwórz* - otwarcie tezaurusa zapisanego w pliku w postaci binarnej,
 - *Edycja* - edycja aktualnie używanego tezaurusa,
 - *Przeglądanie* - przeglądanie bez możliwości edycji aktualnie używanego tezaurusa,
 - *Zapisz* - zapis w postaci binarnej tezaurusa do pliku,
 - *Zapisz jako* - zapis w postaci binarnej tezaurusa do pliku, przy kolejnym uruchomieniu zapis do nowego pliku,
 - *Zapisz XML* - zapis tezaurusa do pliku w formacie XML,



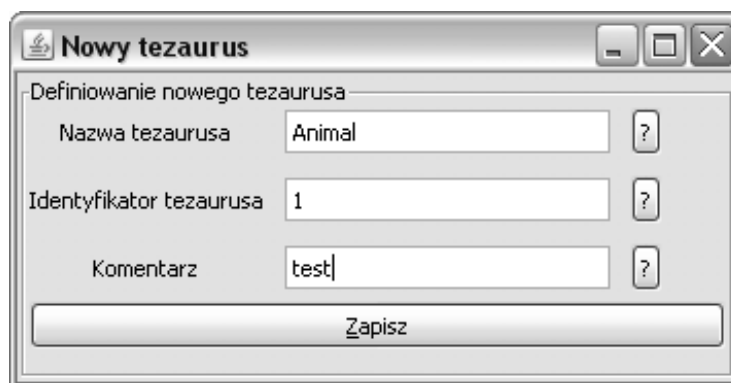
Rys. 6.1: Zakładka startowa interfejsu MMNET wersja 0.5.8, tuż po uruchomieniu

- *Zapisz XML jako* - zapis tezaury do pliku w formacie XML, przy kolejnym uruchomieniu zapis do nowego pliku,
- *Otwórz XML* - otwarcie tezaury, zapisanego w formacie XML na dysku.
- *Stwierdzenia proste*
 - *Nowy* - definiowanie nowego stwierdzenia prostego w otwartym tezaurysie,
 - *Edycja* - edycja stwierdzeń prostych dostępnych w otwartym tezaurysie,
 - *Przeglądanie* - przeglądanie bez możliwości edycji stwierdzeń prostych dostępnych w otwartym tezaurysie.
- *Stwierdzenia złożone*
 - *Nowy* - definiowanie nowych stwierdzeń złożonych na podstawie wcześniej zdefiniowanych stwierdzeń prostych dostępnych w otwartym tezaurysie,
 - *Edycja* - edycja stwierdzeń złożonych dostępnych w otwartym tezaurysie,
 - *Przeglądanie* - przeglądanie stwierdzeń złożonych dostępnych w otwartym tezaurysie.
- *Model*
 - *Nowy* - definiowanie nowego modelu,
 - *Otwórz* - otwarcie modelu zapisanego w pliku w postaci binarnej,
 - *Edycja* - edycja aktualnie używanego modelu,
 - *Przeglądanie* - przeglądanie bez możliwości edycji aktualnie używanego modelu,
 - *Zapisz* - zapis w postaci binarnej modelu do pliku,

- *Zapisz jako* - zapis w postaci binarnej modelu do pliku, przy kolejnym uruchomieniu zapis do nowego pliku,
 - *Zapisz XML* - zapis modelu do pliku w formacie XML,
 - *Zapisz XML jako* - zapis modelu do pliku w formacie XML, przy kolejnym uruchomieniu zapis do nowego pliku,
 - *Otwórz XML* - otwarcie modelu zapisanego w formacie XML na dysku.
- *Warstwy*
 - *Nowy* - definiowanie nowej warstwy modelu,
 - *Edycja* - edycja istniejących warstw modelu,
 - *Przeglądanie* - przeglądanie istniejących warstw modelu.
 - *Zadania*
 - *Obliczenia* - przeprowadzanie obliczeń,
 - *Katalog roboczy* - zmiana katalogu roboczego,
 - *Zapis do pliku txt* - zapis danych do pliku *.txt.

6.3. Definiowanie nowego tezaurusa

Nowy tezaurus definiuje się z wykorzystaniem okna pokazanego na rys. 6.2. Dla każdego tezaurusa możemy ustalić jego nazwę (ciąg znaków), identyfikator (wartość liczbowa) oraz komentarz (ciąg znaków).



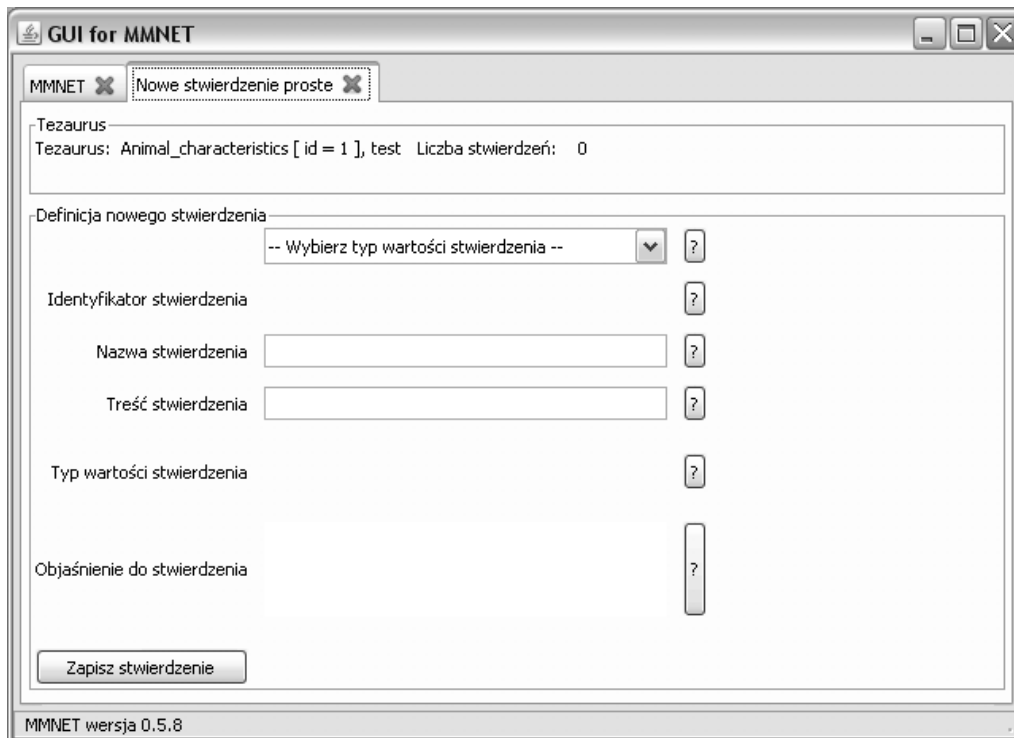
Rys. 6.2: Okno umożliwiające utworzenie nowego tezaurusa

6.4. Definiowanie stwierdzeń prostych

Definiowanie nowych stwierdzeń prostych odbywa się z wykorzystaniem zakładki pokazanej na rys. 6.3. Obecnie istnieje możliwość definiowania stwierdzeń, dla których typem wartości jest wartość logiczna, stopień przekonania o słuszności stwierdzenia lub przybliżony stopień przekonania o słuszności stwierdzenia. W celu zdefiniowania nowego stwierdzenia należy z listy rozwijanej wybrać odpowiedni typ wartości stwierdzenia. Automatycznie wygenerowany zostanie wtedy identyfikator nowego stwierdzenia. Po uzupełnieniu formularza o pozostałe dane można zapisać stwierdzenie i przejść do definiowania kolejnego. Opis dostępnych parametrów:

- parametry generowane automatycznie, które nie podlegają edycji,

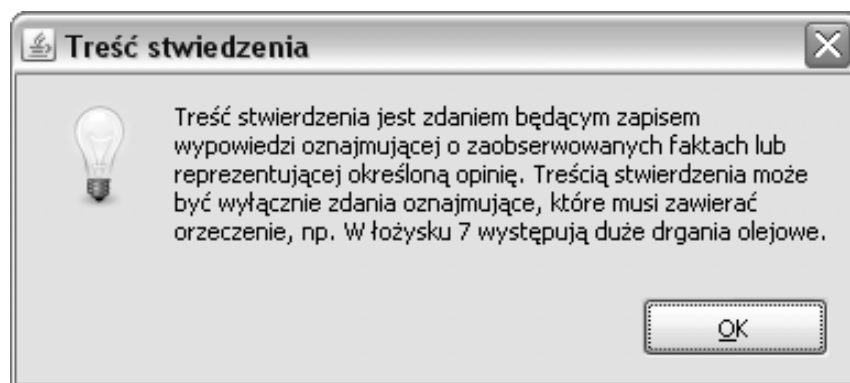
- *identyfikator stwierdzenia* - unikalna liczba (docelowo łańcuch znakowy) stosowana w celu identyfikacji stwierdzenia, generowany automatycznie z krokiem 10,
- *typ wartości stwierdzenia* - określa rodzaj danych reprezentujących wartości stwierdzeń np. wartość logiczna, stopień przekonania, przybliżony stopień przekonania.



Rys. 6.3: Zakładka umożliwiająca definiowanie nowego stwierdzenia prostego

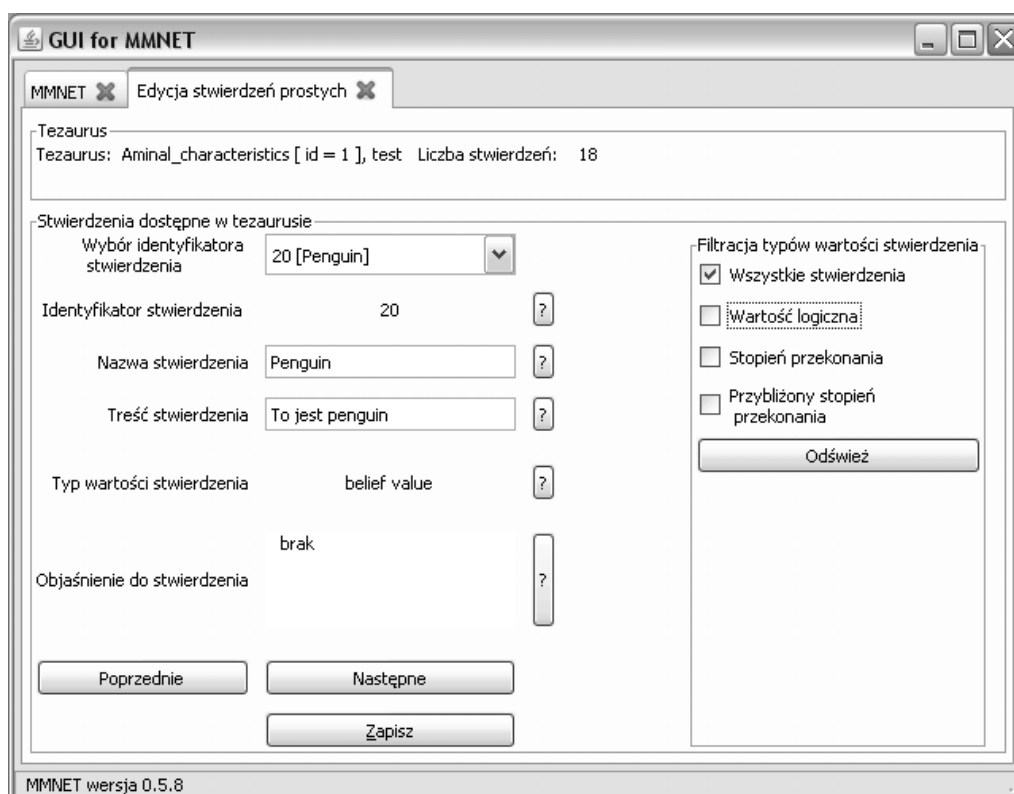
- parametry określone przez użytkownika:
 - *nazwa stwierdzenia* - stosowana jest w celu łatwiejszej identyfikacji stwierdzenia i jego treści, stosowana w tablicach, grafach i tagach, nazwa stwierdzenia prostego wykorzystywana jest przy budowie stwierdzeń złożonych do definiowania parametru o nazwie *wektor wariantów stwierdzenia*, w nazwie stwierdzenia nie mogą pojawić się spacje,
 - *treść stwierdzenia* - treść stwierdzenia zawiera pełny zapis treści stwierdzenia np. duże drgania olejowe w łożysku 7,
 - *objaśnienie do stwierdzenia* - krótki i zwięzły opis znaczenia treści zawartych w polu *treść stwierdzenia*, np.: wartość ciśnienia sprężania w komorze spalania jest prawidłowa jeżeli różnica pomiędzy wartością nominalną podaną przez producenta a wartością odczytaną nie jest większa niż 10 procent.

Pod przyciskami ze znakiem zapytania umieszczono zwięzłą pomoc podręczną. Po kliknięciu na dany przycisk pojawia się okno wraz z opisem dotyczącym konkretnego pola. Na rys. 6.4 pokazano okno z informacją związaną z polem *Treść stwierdzenia*.

Rys. 6.4: Dodatkowa informacja związana z polem *Treść stwierdzenia*

6.5. Przegląd oraz edycja istniejących stwierdzeń prostych

Przegląd oraz edycja istniejących stwierdzeń prostych odbywa się z wykorzystaniem prawie identycznych zakładek. W przypadku przeglądania stwierdzeń nie występuje możliwość edycji żadnego z pól (brak przycisku *Zapisz*), natomiast w przypadku edycji możemy zmienić tylko te parametry, które można było ręcznie ustawić przy definiowaniu nowego stwierdzenia (rozdział 6.4). Zakładkę umożliwiającą edytowanie stwierdzeń pokazano na rys. 6.5. W celu dokonania edycji stwierdzenia, lub jedynie jego przeglądnięcia należy z listy rozwijanej wybrać odpowiednie stwierdzenie. Ewentualne zmiany zapisujemy przyciskiem *Zapisz*.



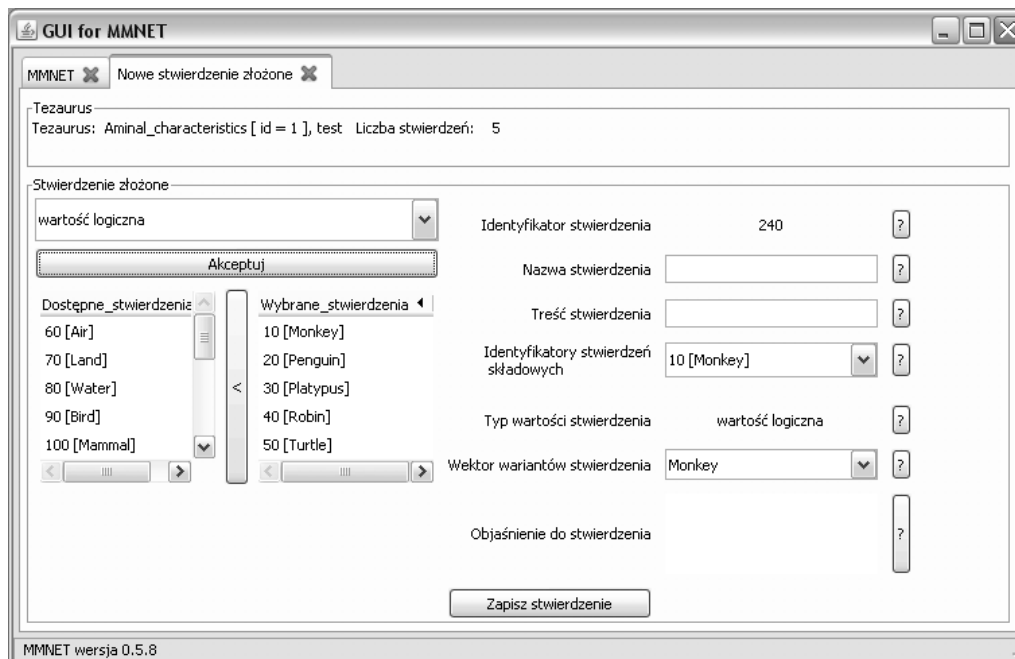
Rys. 6.5: Zakładka umożliwiająca edycję istniejących stwierdzeń prostych

W celu łatwiejszego dostępu do poszczególnych stwierdzeń zastosowano filtrację typów stwierdzeń, która umożliwia zawężenie dostępnych identyfikatorów stwierdzeń do przedziału

odpowiadającym stwierdzeniom, których typem wartości jest na przykład stopień przekonania lub wartość logiczna.

6.6. Definiowanie stwierdzeń złożonych

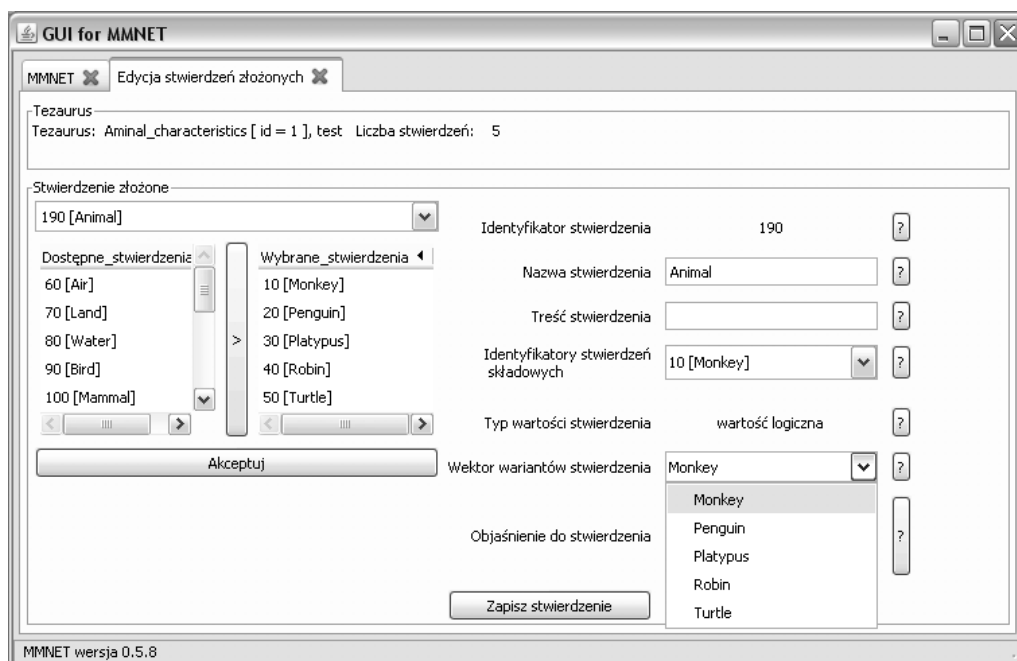
W celu zdefiniowania stwierdzenia złożonego należy najpierw zdefiniować odpowiednie stwierdzenia proste, które wchodzi w skład stwierdzenia złożonego. Należy przy tym pamiętać aby nadać tym stwierdzeniom prostym odpowiednie typy wartości (np. stopień przekonania lub wartość logiczna). Następnie z wykorzystaniem zakładki pokazanej na rys. 6.6 można przystąpić do definiowania nowego stwierdzenia złożonego. Z listy rozwijanej należy wybrać odpowiedni typ wartości stwierdzeń składowych. Do tabelki zostaną wczytane identyfikatory dostępnych stwierdzeń prostych wybranego typu. Po wybraniu identyfikatorów stwierdzeń, które mają wejść w skład stwierdzenia złożonego i zatwierdzeniu wyboru przyciskiem *Akceptuj* utworzone zostanie nowe stwierdzenie złożone. Automatycznie wygenerowany zostanie identyfikator stwierdzenia złożonego, pobrane zostaną identyfikatory stwierdzeń składowych oraz nazwy stwierdzeń składowych. Po uzupełnieniu pozostałych danych można zapisać stwierdzenie i przystąpić do definiowania następnego.



Rys. 6.6: Zakładka umożliwiająca definiowanie nowego stwierdzenia złożonego, stan kliknięcia przycisku *Akceptuj*

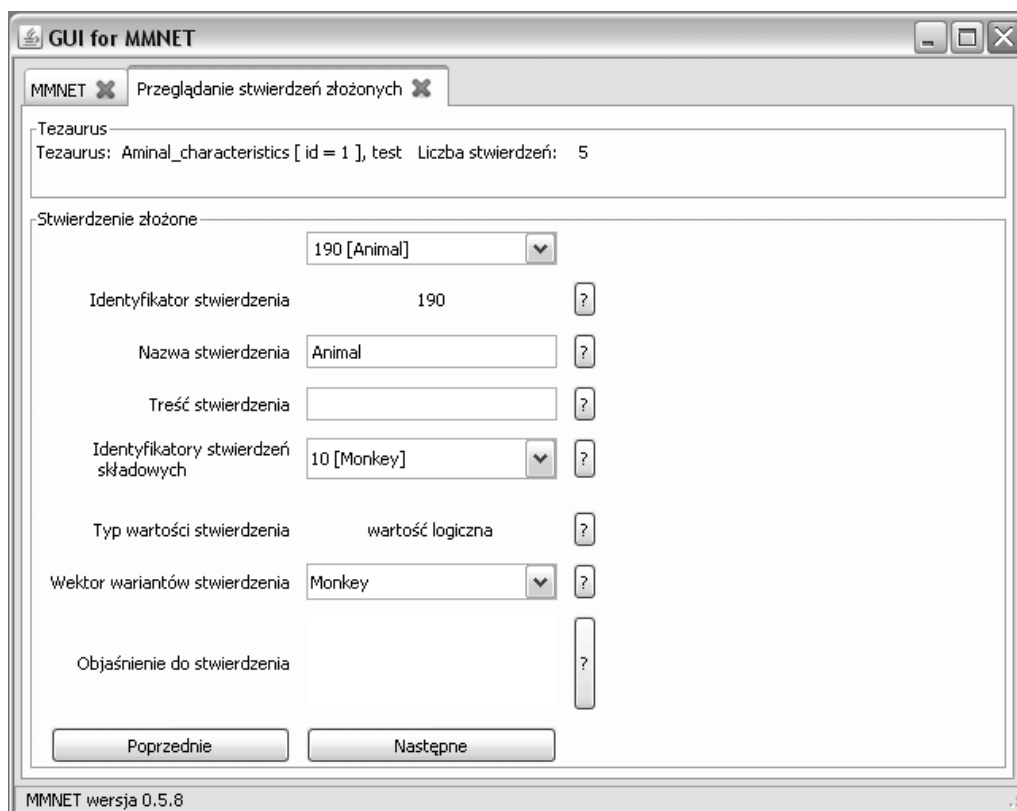
6.7. Edycja oraz przegląd istniejących stwierdzeń złożonych

Zdefiniowane stwierdzenia złożone można edytować z wykorzystaniem zakładki pokazanej na rys. 6.7. Po wybraniu z listy rozwijanej identyfikatora stwierdzenia złożonego wczytane zostaną identyfikatory stwierdzeń składowych oraz identyfikatory stwierdzeń dostępnych w tezaurysie. Po dokonaniu zmian klikamy na przycisk *Akceptuj*, zaktualizowane zostaną wtedy identyfikatory stwierdzeń składowych oraz nazwy stwierdzeń składowych. W razie potrzeby możemy również dokonać modyfikacji innych dostępnych parametrów stwierdzenia złożonego. Wszystkie zmiany zatwierdzamy przyciskiem *Zapisz*.



Rys. 6.7: Zakładka umożliwiająca edycję stwierdzeń złożonych

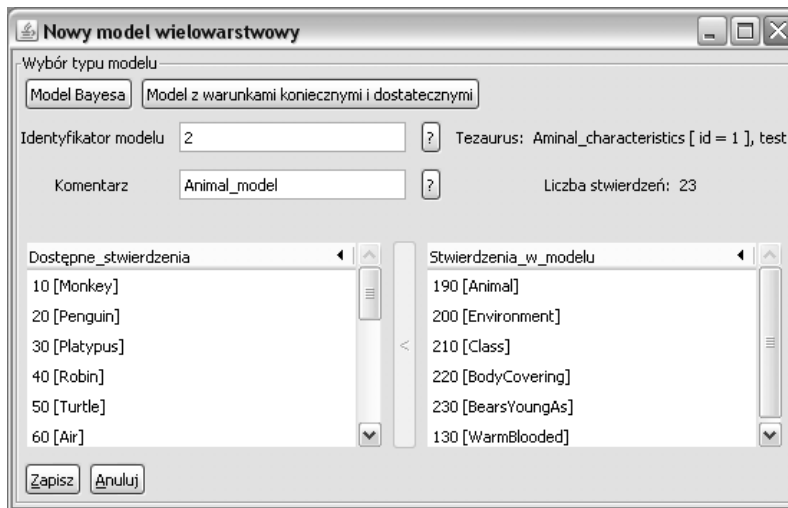
W celu przeglądu istniejących w tezaursie stwierdzeń złożonych należy skorzystać z zakładki pokazanej na rys. 6.8. Odpowiednie stwierdzenia wybieramy z listy rozwijanej z dostępnymi identyfikatorami i nazwami stwierdzeń.



Rys. 6.8: Zakładka umożliwiająca przeglądanie stwierdzeń złożonych

6.8. Definiowanie nowego modelu

Nowy model definiuje się z wykorzystaniem okna pokazanego na rys. 6.9. Po wybraniu typu modelu (Model Bayesa lub Model z warunkami koniecznymi i dostatecznymi) można ustalić identyfikator modelu (ciąg liczbowy) oraz opcjonalny komentarz (ciąg znaków). Następnie z listy dostępnych stwierdzeń w teaurusie należy wybrać te, które będą wykorzystane w modelu do budowy warstw (wybrane stwierdzenia przeliczamy z kolumny lewej do prawej) i zatwierdzić wszystko przyciskiem *Zapisz*.



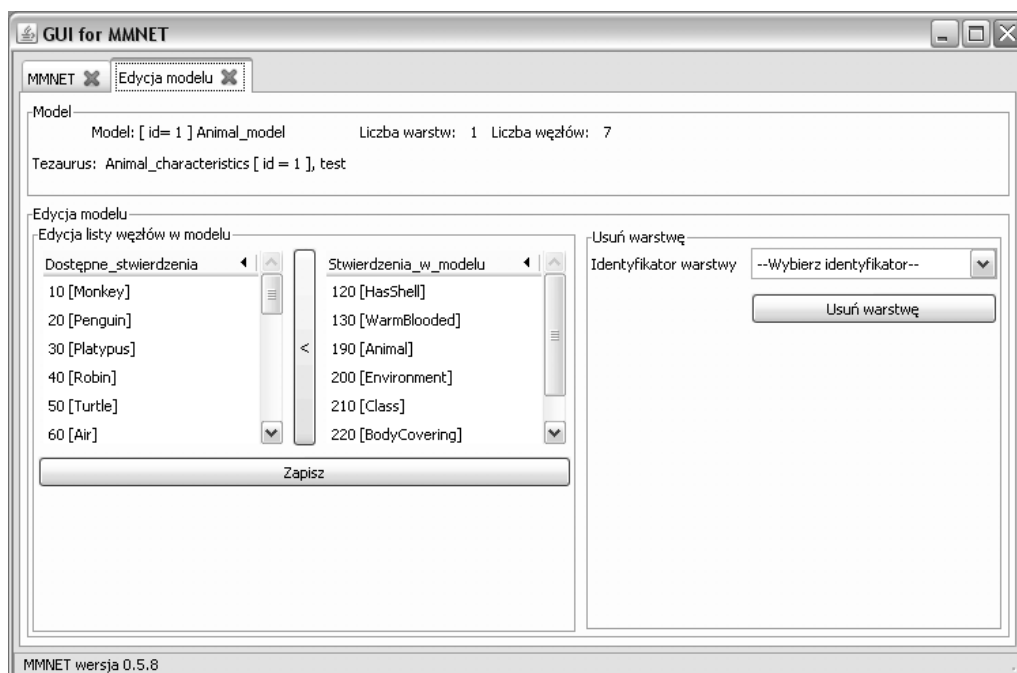
Rys. 6.9: Okno umożliwiające utworzenie nowego modelu

6.9. Edycja modelu

Każdy utworzony model możemy modyfikować. Zakres zmian jest zależny od struktury modelu w chwili jego edycji. Możemy dodać nowe stwierdzenia do modelu jak również usunąć te, które są już w modelu jednakże pod warunkiem, że nie są one jeszcze wykorzystywane w warstwach. Można również usuwać z modelu zdefiniowane wcześniej warstwy. Zakładkę, która umożliwia edycję modelu pokazano na rys. 6.10.

6.10. Definiowanie nowych warstw

Do każdego zdefiniowanego modelu możemy dodawać warstwy. Proces ten odbywa się z wykorzystaniem zakładki pokazanej na rys. 6.11. Po kliknięciu na przycisk *Dodaj nową warstwę* automatycznie wygenerowany zostanie identyfikator warstwy i wczytane zostaną dostępne stwierdzenia. Dla modelu Bayesa stwierdzenia proste i złożone, natomiast dla modelu z warunkami koniecznymi i dostatecznymi tylko stwierdzenia proste. Do tworzonej warstwy można dodawać komentarz oraz ustalać wagę warstwy (dowolna liczba dodatnia, domyślnie 1). Następnie należy zdefiniować, które stwierdzenia dostępne w modelu będą występowały w warstwie i staną się jej węzłami. Proces ten jest podobny do wyboru stwierdzeń, które mają być obecne w modelu. Po skończeniu tej czynności należy przejść do definiowania krawędzi występujących w danej warstwie. W ramce *Definicja krawędzi* z listy rozwijanej *Węzeł 1* wybieramy pierwszy węzeł, następnie z listy *Węzeł 2* drugi węzeł. Następnie należy zdefiniować kierunek krawędzi pomiędzy dwoma rozpatrywanymi węzłami. W przypadku modelu z warunkami koniecznymi i dostatecznymi należy również określić wagę krawędzi. Wszystkie ustawienia zatwierdzamy przyciskiem



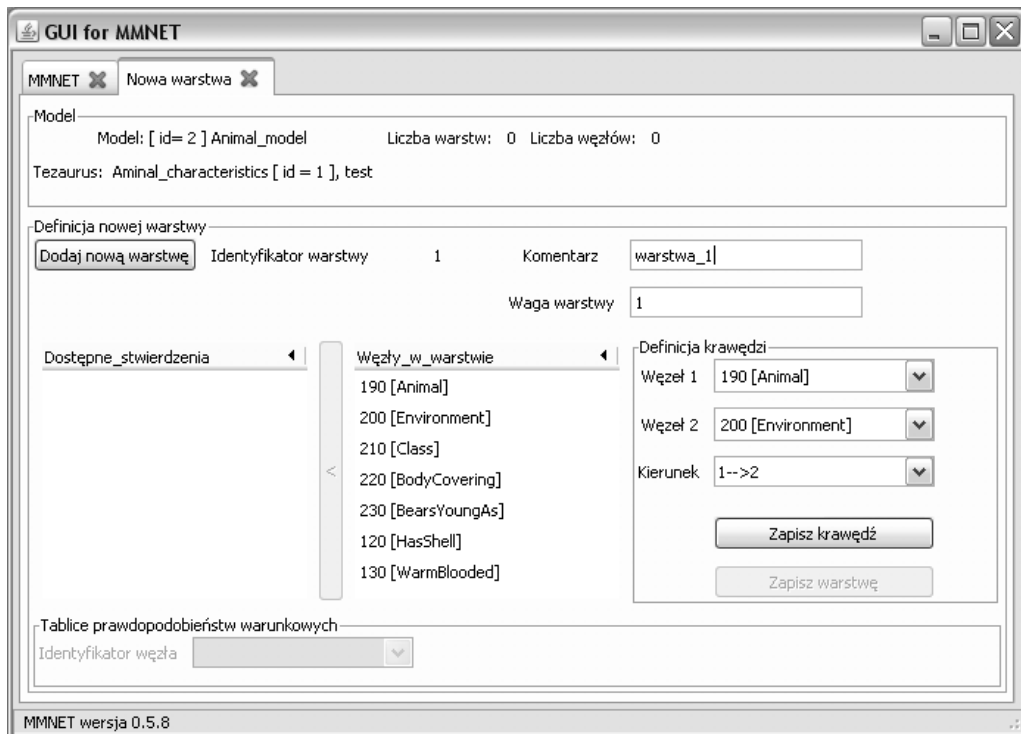
Rys. 6.10: Zakładka umożliwiająca edycję istniejącego modelu

Zapisz krawędź. Powyższą operację należy przeprowadzić tyle razy ile jest krawędzi w danej warstwie, za każdym razem zapisując krawędź. Przy próbie zapisu krawędzi, która została już wcześniej zdefiniowana pojawi się stosowny komunikat informujący o niemożliwości zapisania danej krawędzi. Należy zwrócić uwagę na to, aby węzeł o niższym identyfikatorze był wybrany zawsze z listy *Węzeł 1*. Po zdefiniowaniu wszystkich krawędzi naciskamy przycisk *Zapisz warstwę* w celu zapisania warstwy w modelu. W przypadku definiowania warstwy w modelu Bayesa należy następnie przejść do zdefiniowania tablic prawdopodobieństw warunkowych dla poszczególnych węzłów (tablice CPT nie występują w drugim rodzaju modelu).

Po wybraniu z listy rozwijanej *Identyfikator węzła* dostępnej w ramce *Tablice prawdopodobieństw warunkowych* identyfikatora węzła otwiera się okno pokazane na rys. 6.12, w którym mamy możliwość zdefiniowania wartości w tablicy prawdopodobieństw warunkowych dla wybranego węzła. Początkowe wartości tablic ustawiane są domyślnie na zero. Opis kombinacji stanów węzłów nadrzędnych jest zgodny z kolejnością podaną w pozycji *Identyfikatory węzłów nadrzędnych*. Po zdefiniowaniu nowych wartości prawdopodobieństw zapisujemy zmiany przyciskiem *Zapisz tablicę CPT*.

6.11. Edycja istniejących warstw

Istniejące warstwy możemy edytować używając w tym celu zakładkę pokazaną na rys. 6.13. Po wybraniu identyfikatora warstwy można przystąpić do edycji poszczególnych elementów wybranej warstwy, klikając na odpowiedni przycisk. Uaktywniona zostanie wtedy odpowiednia ramka z możliwością dokonania edycji. Obsługa jest taka sama jak w przypadku identycznych obiektów na innych zakładkach. Zmiana wagi warstwy oraz komentarza do warstwy zatwierdzana jest przyciskiem *Zapisz* znajdującym się poniżej pól z tymi zmiennymi. Dokonanie zmian (dodanie nowych węzłów, dodanie/usunięcie krawędzi) wiąże się z wyzerowaniem tablic prawdopodobieństw warunkowych, dla węzłów dla których przeprowadzono wspomniane wcześniej operacje. Dlatego też konieczne jest ponowne ustalenie wartości odpowiednich tablic CPT.



Rys. 6.11: Zakładka umożliwiająca dodawanie nowych warstw do modelu

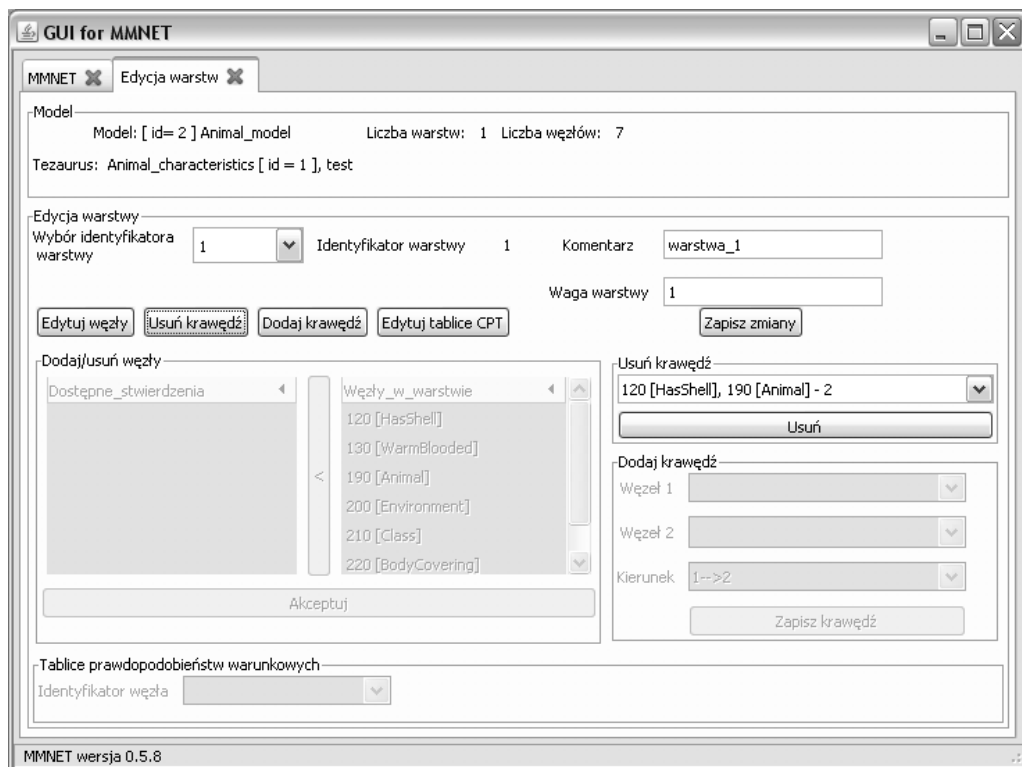


Rys. 6.12: Okno umożliwiająca definicję tablic CPT

6.12. Przeglądanie warstw

Warstwy zdefiniowane w modelu możemy przeglądać z wykorzystaniem zakładki pokazanej na rys. 6.14. Po wybraniu z listy rozwijanej identyfikatora warstwy możemy przeglądać aktualne wartości następujących parametrów:

- *komentarz* - opcjonalny komentarz do warstwy,
- *krawędzie* - lista wszystkich dostępnych krawędzi w warstwie, zapisana w postaci $x_1, x_2 - x_3$, gdzie: x_1 to identyfikator i nazwa pierwszego węzła, x_2 to identyfikator i nazwa drugiego węzła, natomiast x_3 to kierunek pomiędzy węzłami, 1 w przypadku krawędzi



Rys. 6.13: Zakładka umożliwiająca edycję istniejących warstw modelu

od węzła pierwszego do drugiego, 2 dla przypadku od węzła drugiego do pierwszego,

- *identyfikatory węzłów* - lista identyfikatorów wszystkich węzłów występujących w danej warstwie.

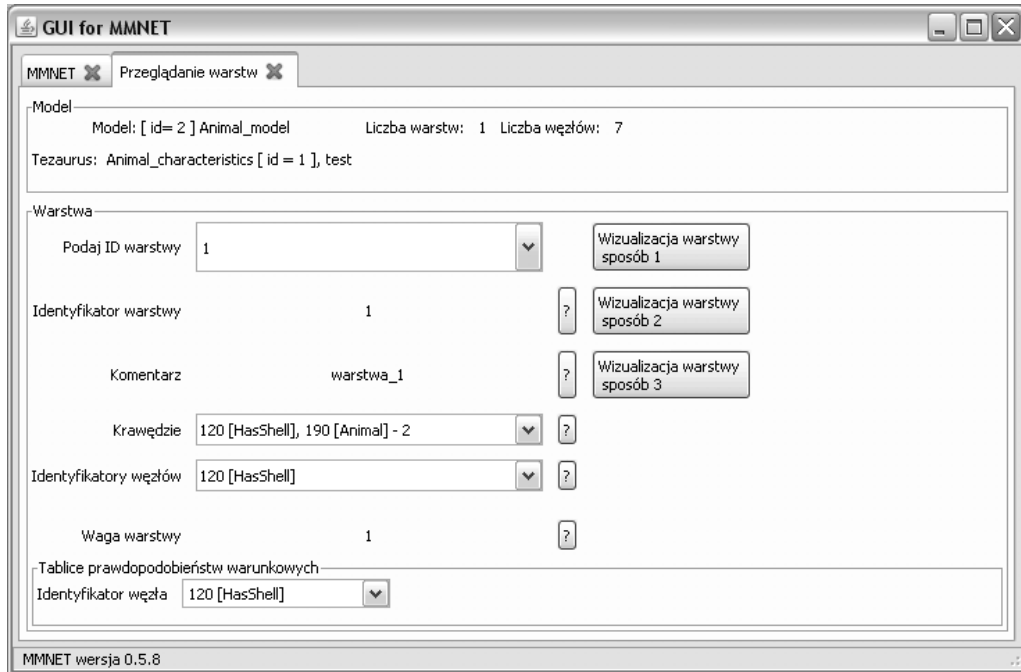
Tablice prawdopodobieństw warunkowych możemy przeglądać wybierając odpowiedni identyfikator węzła z listy rozwijanej w ramce *Tablice prawdopodobieństw warunkowych*. Istnieje również możliwość zobaczenia wizualizacji warstwy w postaci grafu, wykorzystując jeden z trzech dostępnych sposobów. Pierwszy z nich umożliwia wyświetlenie grafu w postaci statycznej. W przypadku drugiego sposobu użytkownik ma możliwość zmiany położenia poszczególnych węzłów warstwy. Ostatni ze sposobów umożliwia użytkownikowi obracanie grafu w przestrzeni trójwymiarowej. Przykład wizualizacji z wykorzystaniem pierwszego sposobu pokazano na rys. 6.15.

6.13. Wykonywanie obliczeń

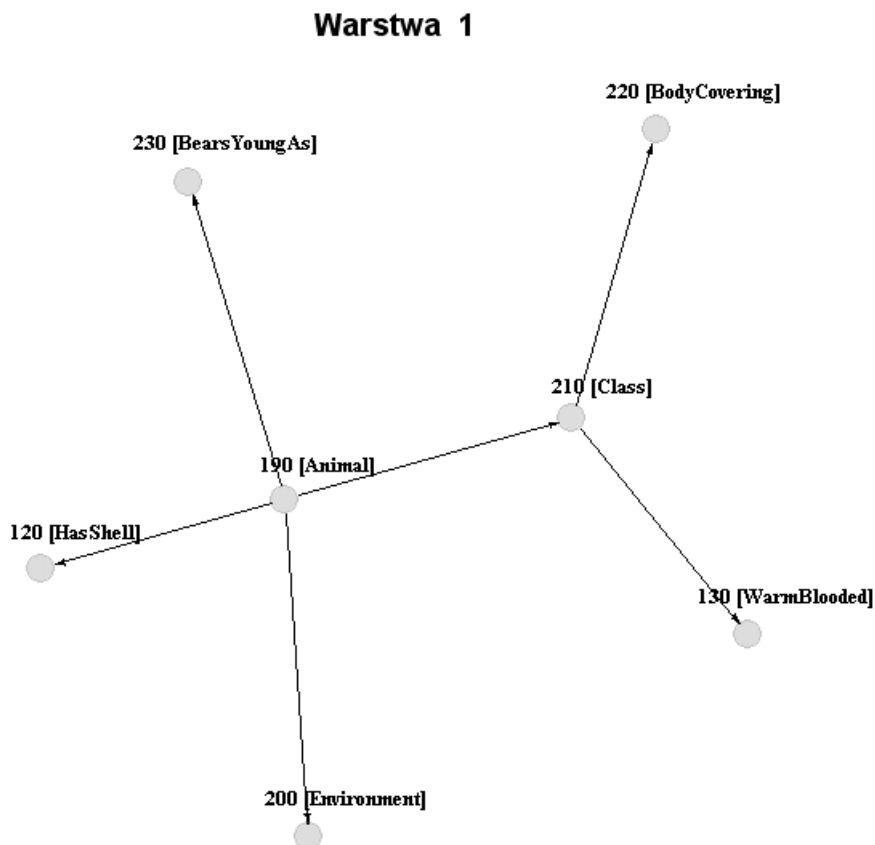
Zakładkę umożliwiającą przeprowadzenie obliczeń pokazaną na rys. 6.16 uruchamia się za pomocą przycisku *Oblicz* znajdującego się na zakładce startowej interfejsu. Użytkownik ma możliwość przeprowadzenia obliczeń z wykorzystaniem wartości zapisanych w modelu (przycisk *Oblicz model dla domyślnych parametrów*), lub dla nowych danych wprowadzonych do konkretnych węzłów (przycisk *Wprowadź dane do węzłów*). Wagę warstwy można zmienić wybierając odpowiedni identyfikator warstwy i wpisując dowolną dodatnią liczbę.

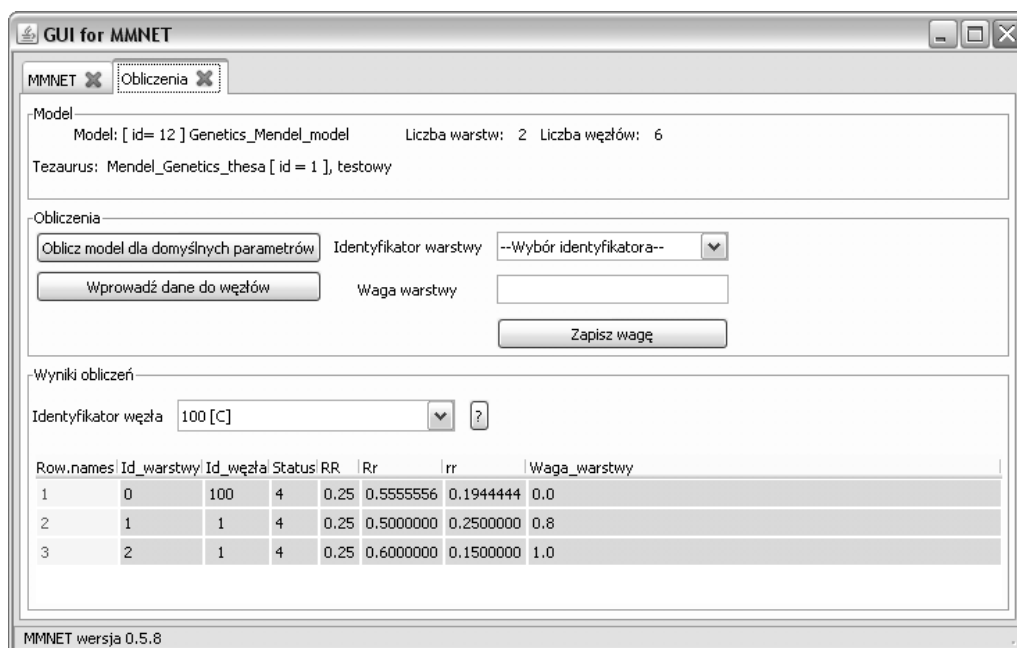
Wyniki obliczeń wyświetlane są w tabeli, w której znajdują się następujące kolumny:

- *Id_warstwy* - identyfikator warstwy, w której rozpatrywany węzeł występuje, dla modeli wielowarstwowych występuje dodatkowa warstwa o identyfikatorze równym zero (tzw.



Rys. 6.14: Zakładka umożliwiająca przeglądanie warstw w modelu Bayesa

Rys. 6.15: Przykładowa wizualizacja warstwy dla modelu *Animal characteristics*

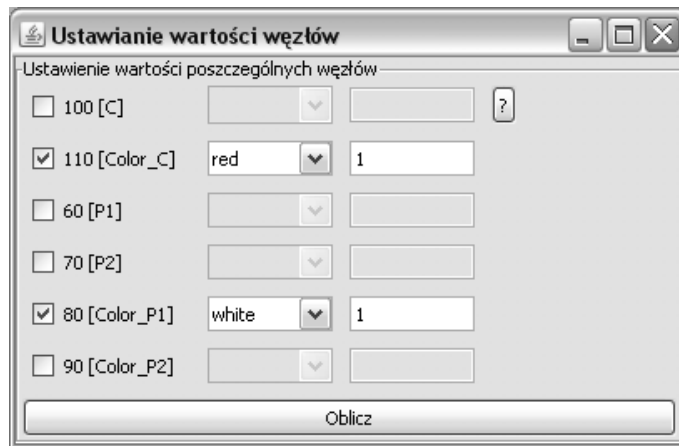


Rys. 6.16: Zakładka umożliwiająca przeprowadzenie obliczeń

warstwa zerowa), w której gromadzone są zagregowane wartości węzłów występujących w warstwach modelu,

- *Id_węzła* - numer porządkowy węzła w wektorze węzłów warstwy, dla warstwy zerowej jest to identyfikator węzła,
- *Status* - informuje o sposobie uzyskania wartości danego węzła, 0 - nieznan, 1 - ustalony, 2 - wyliczony, 3 - zagregowany, 4 - domyślny,
- *wartości węzła* - w przypadku modelu Bayesa n kolumn o nazwach odpowiadających nazwom wartości jakie przyjmuje węzeł, dla przykładu z rys. 6.16 jest to *RR*, *Rr*, *rr*, dla modelu z warunkami koniecznymi i dostatecznymi dwie kolumny *bmin(Si)* i *bmax(Si)*, określające odpowiednio dolną i górną granicę przedziału, w którym znajduje się wartość stopnia przekonania,
- *Waga_warstwy* - waga poszczególnych warstw modelu.

W celu ustalenia nowych wartości w poszczególnych węzłach należy kliknąć w przycisk *Wprowadź dane do węzłów*. W zależności od typu modelu pojawi się okno pokazane na rys. 6.17 dla modelu Bayesa, bądź okno pokazane na rys. 6.18 dla modelu z warunkami koniecznymi i dostatecznymi. Aby wprowadzić dane należy dla pierwszego przypadku wybrać odpowiedni węzeł, a następnie z listy rozwijanej dla danego węzła odpowiedni stan węzła. W polu obok należy wpisać wartość w węzle dla wybranego stanu, domyślnie należy wpisać wartość jeden. W drugim przypadku po wybraniu odpowiedniego węzła należy wpisać wartości stopnia przekonania w postaci przedziału, wpisując w kolumnie *bmin(Si)* dolną granicę przedziału a w kolumnie *bmax(Si)* górną granicę przedziału. Jeżeli dla któregoś z węzłów przedział stopnia przekonania zawęży się do punktu należy w obu kolumnach wpisać te same wartości, odpowiadające danemu stopniowi przekonania.



Rys. 6.17: Okno aplikacji umożliwiające wprowadzanie danych do węzłów w przypadku modelu Bayesa, wybrano węzeł 110[Color_C], stan: red, wartość: 1, węzeł 80[Color_P1], stan: white, wartość: 1



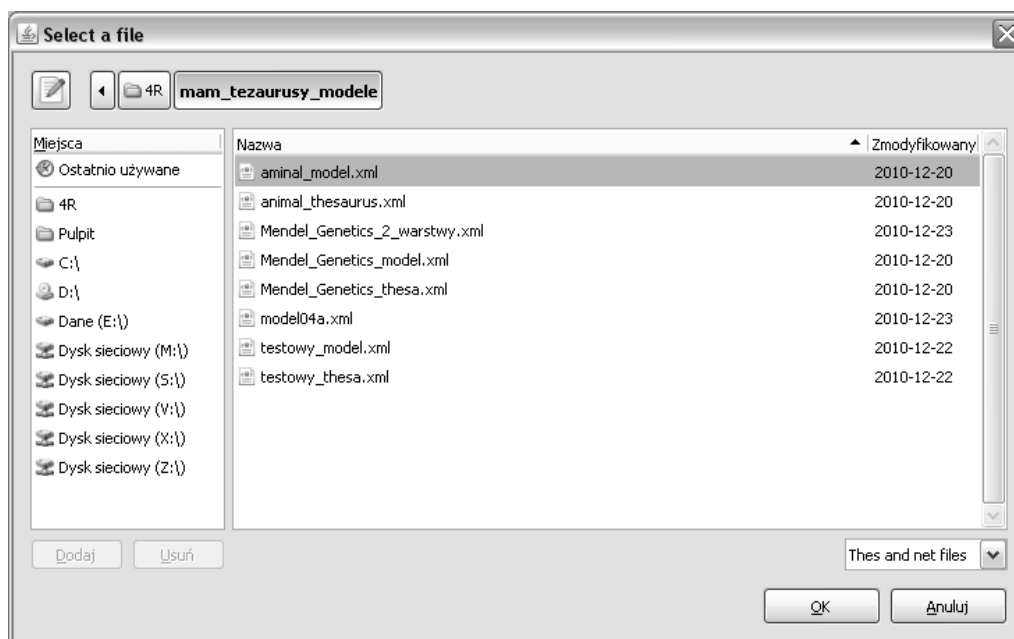
Rys. 6.18: Okno aplikacji umożliwiające wprowadzanie danych do węzłów w przypadku modelu z warunkami koniecznymi i dostatecznymi, wybrano węzeł 20[Penguin] stopień przekonania w przedziale od 0.2 do 1, węzeł 40[Robin] stopień przekonania równy 0.6

6.14. Zapis i odczyt danych z pliku

Baza stwierdzeń oraz zdefiniowane modele mogą być przechowywane na dysku lokalnym w postaci plików binarnych lub w postaci plików XML. W przypadku plików binarnych słowniki stwierdzeń zapisywane są z rozszerzeniem **.thes*, natomiast pliki zawierające zdefiniowane modele mają rozszerzenie **.net*. Aby zapisać tezaursus bądź model do pliku należy kliknąć w odpowiedni przycisk, a następnie w pojawiającym się okienku wpisać nazwę pod jaką chcemy zapisać na dysku model bądź słownik stwierdzeń. Podajemy nazwę bez polskich znaków oraz bez rozszerzenia. Pliki domyślnie zapisywane są w katalogu roboczym środowiska R. Aby dokonać zmiany katalogu należy z poziomu środowiska R wybrać *File* – *> Change dir...*, lub skorzystać z przycisku *Katalog roboczy* umieszczonego na zakładce głównej interfejsu pakietu MMNET.

W celu odczytania danych z pliku należy kliknąć w odpowiedni przycisk i w pojawiającym się oknie wybrać odpowiedni plik i zatwierdzić wybór przyciskiem *OK*. Stosowny komunikat

informuje o trwaniu procesu wczytywania danych. Nie wskazane jest ręczne modyfikowanie (np. za pomocą notatnika) plików binarnych zawierających zapisane słowniki lub modele, grozi ono utratą danych.



Rys. 6.19: Okno odczytu słownika stwierdzeń z pliku

Przy odczytywaniu danych z pliku należy pamiętać, że w danej chwili mamy dostęp wyłącznie do jednego egzemplarza, zarówno słownika stwierdzeń jak i modelu. Odczyt danych z pliku spowoduje zatem skasowanie bieżącej zawartości obiektu i zapisanie na jego miejscu danych z pliku. Dlatego też, aby zachować bieżącą zawartość słownika stwierdzeń bądź modelu należy je przed odczytem nowych danych zapisać do pliku.

6.15. Eksport wprowadzonych danych do pliku TXT

Wprowadzone za pomocą interfejsu dane można zapisać do pliku *.txt za pomocą przycisku *Zapis do pliku txt* umieszczonego na zakładce startowej interfejsu. Poniżej przedstawiono przykład fragmentu pliku z wyeksportowanymi danymi:

Wieloskalowa sieć stwierdzeń

```
Identyfikator modelu: 1
Nazwa modelu: animal
Liczba warstw w modelu: 1
Typ warstw modelu: CNetBayes
```

```
Warstwa 1 :
Identyfikator warstwy: 1
Komentarz do warstwy: warstwa_1
Węzły warstwy:
    120 [HasShell]
    130 [WarmBlooded]
    190 [Animal]
```

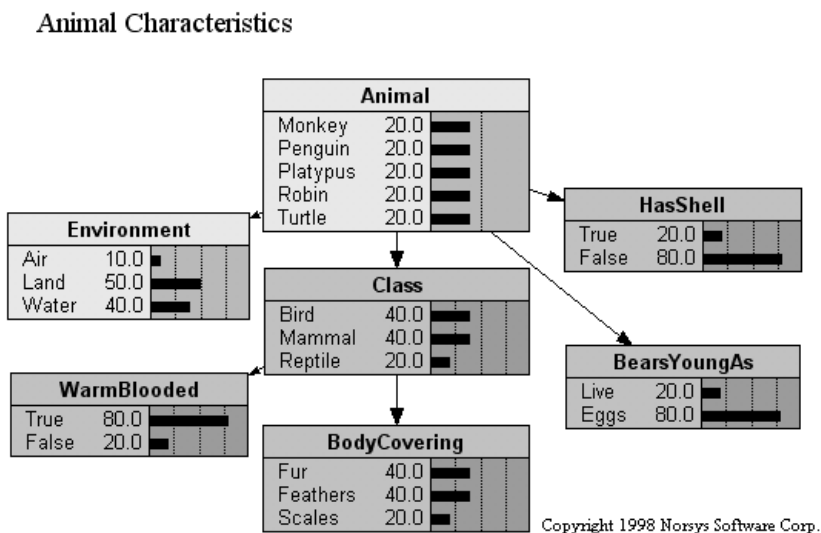
200 [Environment]
 210 [Class]
 220 [BodyCovering]
 230 [BearsYoungAs]

Krawędzie występujące w warstwie:

120 [HasShell] , 190 [Animal] - 2
 130 [WarmBlooded] , 210 [Class] - 2
 190 [Animal] , 200 [Environment] - 1
 190 [Animal] , 210 [Class] - 1
 190 [Animal] , 230 [BearsYoungAs] - 1
 210 [Class] , 220 [BodyCovering] - 1

6.16. Przykład zastosowania interfejsu

Zastosowanie niniejszego interfejsu do tworzenia modeli diagnostycznych pokazano na przykładzie modelu *Animal characteristics*. Model ten dostępny jest w programie Netica a jego strukturę pokazano na rys. 6.20.



Rys. 6.20: Struktura modelu *Animal characteristics* [6.2]

Z punktu widzenia pakietu MMNET model ten zawiera:

- 18 stwierdzeń prostych: Monkey, Penguin, Platypus, Robin, Turtle, Air, Land, Water, Bird, Mammal, Reptile, Live, Eggs, Fur, Feathers, Scales, HasShell, WarmBlooded,
- 5 stwierdzeń złożonych: Animal (Monkey, Penguin, Platypus, Robin, Turtle), Environment (Air, Land, Water), Class (Bird, Mammal, Reptile), BearsYoungAs (Live, Eggs), BodyCovering (Fur, Feathers, Scales),
- jedną warstwę, w której znajdują się węzły: Animal, Environment, HasShell, Class, WarmBlooded, BearsYoungAs, BodyCovering.
- 6 krawędzi: Animal - Environment, Animal - Class, Animal - HasShell, Animal - BearsYoungAs, Class - WarmBlooded, Class - BodyCovering.

Proces budowy tego modelu z wykorzystaniem interfejsu graficznego pakietu MMNET przebiega w następującej kolejności:

- utworzenie nowego tezaury, nadanie mu nazwy, identyfikatora, komentarza,
- zdefiniowanie 18 stwierdzeń prostych,
- zdefiniowanie 5 stwierdzeń złożonych wykorzystując odpowiednie stwierdzenia proste,
- utworzenie nowego modelu (identyfikator, komentarz) i wybranie stwierdzeń, które w nim występują (Animal, Environment, HasShell, Class, WarmBlooded, BearsYoungAs, BodyCovering),
- utworzenie nowej warstwy, zdefiniowanie węzłów, które w niej występują (Animal, Environment, HasShell, Class, WarmBlooded, BearsYoungAs, BodyCovering),
- zdefiniowanie krawędzi pomiędzy węzłami występującymi w danej warstwie, zdefiniowanie tablic CPT dla poszczególnych węzłów warstwy,
- przeprowadzenie obliczeń,
- przeglądanie lub modyfikacja istniejących stwierdzeń, modeli, warstw.

Bibliografia

- [6.1] The igraph library. Dostępny w Internecie: igraph.sourceforge.net. grudzień 2010.
- [6.2] Corp. Norsys Software. <http://www.norsys.com>. grudzień 2010.
- [6.3] Csardi Gabor. Network analysis and visualization. Dostępny w Internecie: cran.r-project.org/web/packages/igraph/igraph.pdf. grudzień 2010.
- [6.4] Verzani John. Examples for gwidgets. Dostępny w Internecie: cran.r-project.org/web/packages/gWidgets/vignettes/gWidgets.pdf. grudzień 2010.
- [6.5] Verzani John. gwidgets: a toolkit-independent api for building guis in r. Dostępny w Internecie: user2007.org/program/presentations/verzani-1.pdf. grudzień 2010.
- [6.6] Verzani John. gwidgets api for building toolkit-independent, interactive guis. Dostępny w Internecie: cran.r-project.org/web/packages/gWidgets/gWidgets.pdf. grudzień 2010.

Rozdział 7

Biblioteka modeli MMNET

Tomasz ROGALA

Pakiet MMNET umożliwia definiowanie modeli służących do opisywania i analizowania złożonych dziedzin wiedzy, często niepewnych, niedokładnych i niepełnych za pomocą grafów reprezentujących dyskretną uproszczoną postać rozpatrywanego zagadnienia. Zgodnie z przyjętymi założeniami systemu MMNET modele te mogą być budowane zarówno jako jedno i wielowarstwowe, a ich poszczególne warstwy mogą być reprezentowane jednym z wybranych typów sieci:

- sieci przekonań, w której metody wnioskowania oparte są na teorii prawdopodobieństwa,
- ogólnej sieci stwierdzeń, gdzie wnioskowanie realizowane jest na podstawie ogólnych praw logiki.

Poszczególne typy sieci wykorzystywane w warstwach modeli są podstawowym elementem, który różnicuje budowane modele pod względem sposobu ich definiowania oraz stosowania. Niniejszy rozdział jest opisem zbioru takich modeli. Szczegóły teoretyczne dotyczące własności poszczególnych rodzajów modeli wielowarstwowych oraz sieci wykorzystywanych w warstwach tych modeli omówiono w raporcie [7.4]. W niniejszym rozdziale opisano natomiast sposób definiowania oraz stosowania różnych modeli, głównie z punktu widzenia użytkownika. Wytyczne poprzedzono opisem poszczególnych typów sieci.

7.1. Model z zastosowaniem sieci przekonań

Zgodnie z przyjętymi założeniami, pakiet MMNET umożliwia budowanie sieci wielowarstwowych składających się z tego samego typu warstw reprezentowanego wybranym typem sieci. Jednym z podstawowych typów modeli jest model, w którym poszczególne warstwy reprezentowane są w postaci sieci przekonań. Sieć przekonań jest dyskretną reprezentacją łącznego rozkładu prawdopodobieństwa reprezentowanego w postaci acyklicznego grafu nieskierowanego, w którym poszczególne wierzchołki reprezentują wybrane zmienne losowe natomiast krawędzie reprezentują asocjacje pomiędzy tymi zmiennymi. Podstawą zwięzłego opisu rozkładu łącznego jest warunkowa niezależność poszczególnych zmiennych losowych od pewnego podzbioru innych zmiennych. Niezależność ta opisana jest strukturą grafu natomiast wartości poszczególnych zmiennych losowych wyznaczane są na podstawie tablic prawdopodobieństw warunkowych przyporządkowanych poszczególnym wierzchołkom. Jak opisano w raporcie [7.4] poszczególne zmienne losowe mogą być reprezentowane w postaci stwierdzeń prostych jak i złożonych.

Analogicznie jak w przypadku innych sieci, definiowanie modelu rozpoczyna się od wybrania tych stwierdzeń prostych i złożonych, które stanowią dziedzinę opisywanego przez model zagadnienia. Dziedzinę opisuje zdefiniowany przez użytkownika tezaursus. Graficznie tę dziedzinę reprezentują węzły poszczególnych warstw. W przypadku definiowania węzłów zawierających stwierdzenia złożone, należy zwrócić szczególną uwagę na konieczność definiowania stanów stwierdzenia złożonego w taki sposób, aby stany stwierdzenia złożonego były wartościami wzajemnie wykluczającymi się, a ich zbiór był zupełny. W przypadku trudności ze zdefiniowaniem zupełnego zbioru stanów stwierdzenia złożonego zbiór ten należy uzupełnić o stan który obejmuje wszystkie pozostałe niewymienione wartości stwierdzenia np. Inne. Po zdefiniowaniu zbioru wszystkich węzłów, należy zdefiniować asocjacje występujące pomiędzy stwierdzeniami.

Po zdefiniowaniu poprawnej struktury sieci użytkownik rozpoczyna definiowanie parametrów w postaci tablic prawdopodobieństw warunkowych, w których definiowane są prawdopodobieństwa warunkowe lub wartości odwzorowujące jego subiektywne stopnie przekonania na temat wybranych zdarzeń. Po zdefiniowaniu kompletu parametrów tablic prawdopodobieństw warunkowych możliwe jest uruchomienie procedury wnioskowania w sieci. Algorytm rozwiązywania sieci przekonań został oparty na algorytmie wyznaczania drzewa złączeń w którym fragment dotyczący wyznaczania triangulacji grafu został oparty na eksperymentalnym algorytmie heurystycznym polegającym na wyznaczeniu jak najmniejszych klik. Algorytm ten został opracowany w ramach pakietu *gRBase* [7.2] natomiast jego pozostałe elementy zostały opracowane również w ramach pakietu *gRain* [7.1]. Szczegóły dotyczące algorytmu budowania drzewa złączeń oraz propagacji stopni przekonania zostały szeroko opisane w bogatej literaturze np. [7.6] i nie będą tutaj przytaczane.

Dodatkowe zadania połączone z rozwiązywaniem sieci przekonań to m.in. zadania związane z:

- inicjacją sieci i zapisem stanu sieci dla domyślnych wartości węzłów w celu ograniczenia konieczności ponownego budowania drzewa złączeń w przypadku, gdy nie została zmieniona struktura sieci,
- wprowadzaniem i usuwaniem wartości określonych stwierdzeń, których wartości zostały przyjęte jako znane,
- uruchamianiem operacji związanych z agregacją wartości stwierdzeń w modelu.

Implementacja sieci przekonań w pakiecie MMNET nie obejmuje rozwiązywania sieci za pomocą algorytmu maksymalizacji wartości oczekiwanej (ang. expectation maximization). W związku z powyższym brak jest możliwości wprowadzania zadanych, przybliżonych wartości stwierdzeń, z przedziału od zera do jeden, a jedynie jako wartości dokładnych zero lub jeden.

Dla potrzeb stosowania sieci wielomodalnych zaimplementowano odpowiednie metody agregacji wartości stwierdzeń. W pakiecie MMNET zagregowane wartości stwierdzeń mogą być wynikiem uśredniania arytmetycznego lub też wynikiem uśredniania ważonego, gdzie poszczególne wagi odpowiadają przyporządkowanym wagom poszczególnych warstw modelu.

7.1.1. Podsumowanie

Pakiet MMNET dostarcza pełną funkcjonalność budowania oraz stosowania sieci stwierdzeń jaka jest niezbędna dla potrzeb jej stosowania w ramach diagnostycznych systemów doradczych. Jedynym ważnym ograniczeniem jej stosowania jest brak dodatkowych funkcji i procedur umożliwiających strojenie parametrów sieci oraz brak odpowiednich narzędzi do jej analizy m.in. analizy wrażliwości oraz brak możliwości definiowania wartości punktowych dla wybranych stanów stwierdzeń w postaci liczby z przedziału od zera do jeden. Ograniczenia te będą usuwane wraz z dalszym rozwojem oprogramowania pakietu MMNET.

7.2. Model z zastosowaniem sieci umożliwiających stosowanie analogii mechanicznej

Biblioteka modeli MMNET obejmuje również możliwość stosowania ogólnych sieci stwierdzeń, w których proces wnioskowania został oparty na podstawie ogólnych praw logiki, a dokładniej implikacji z wykorzystaniem warunków wystarczających i koniecznych. Szczegółowy opis dotyczących tego typu sieci, został przedstawiony w raporcie [7.4].

W przypadku sieci stwierdzeń opracowany został algorytm wnioskowania według opisu przedstawionego w [7.4]. Obejmuje on:

- ogólny algorytm rozwiązywania sieci stwierdzeń,
- zadanie programowania liniowego w celu uzyskania wartości szczegółowych.

Zgodnie z opisem, zadanie programowania liniowego oraz zmodyfikowany algorytm rozwiązywania sieci stwierdzeń są uruchamiane w przypadku wystąpienia sprzeczności w sieci. W celu poszukiwania rozwiązania zadania programowania liniowego wykorzystano implementację algorytmu pakietu *linprog* [7.5] oraz *solveLp* [7.3], które umożliwiają sterowanie procesem poszukiwania rozwiązania za pomocą następujących parametrów:

- liczba iteracji,
- pole tolerancji rozwiązania.

W przypadku modeli ogólnych sieci stwierdzeń definiowanie sieci rozpoczyna się od dokładnego opisu rozpatrywanej dziedziny na podstawie jedynie stwierdzeń prostych. W przypadku tych stwierdzeń, które dotychczas definiowane były jako złożone, istnieje konieczność wykorzystania jedynie stwierdzeń prostych, na podstawie których było ono zbudowane. Również w tym przypadku należy zwrócić szczególną uwagę na konieczność definiowania wartości stwierdzeń prostych w taki sposób, aby ich wartości z pozostałymi stwierdzeniami prostymi były wartościami wzajemnie wykluczającymi się, a zbiór tych stwierdzeń był zbiorem zupełnym. Po zdefiniowaniu zbioru wszystkich węzłów, należy zdefiniować zależności występujące pomiędzy stwierdzeniami zgodnie z kierunkowością odpowiadającą warunkom wystarczającym (dostatecznym). Nieprawidłowe zdefiniowanie kierunkowości może prowadzić do uzyskania błędnych wyników i niepoprawnej struktury grafu. Dla każdej gałęzi kierunkowej możliwe jest również zdefiniowanie wag krawędzi, które są uwzględniane podczas obliczeń wymagających rozwiązanie sprzeczności w sieci. Należy podkreślić, że w zależności od przyjętych wag rozpatrywany jest inny układ sieci, w którym określone asocjacje pomiędzy węzłami są mniej lub bardziej ważne od pozostałych i wiąże się z uzyskaniem innych wartości funkcji kryterialnej. W przeciwieństwie do sieci przekonania, ogólne sieci stwierdzeń nie wymagają definiowania innych parametrów. Z drugiej strony wynik wnioskowania dla domyślnie przyjętych wartości stwierdzeń nie prowadzi do uzyskania istotnych wyników do momentu, gdy jakiegokolwiek z wartości stwierdzeń nie zostaną zdefiniowane przez użytkownika lub inny zewnętrzny proces będący obserwatorem wybranej zmiennej.

Dodatkowe zadania związane z rozwiązywaniem ogólnych sieci stwierdzeń to m.in. zadania związane z:

- inicjacją sieci, która obejmuje wstępną inicjację wartości poszczególnych stwierdzeń i krawędzi je łączących oraz ich kasowanie w przypadku ponownego uruchomienia procesu wnioskowania,
- wprowadzaniem i usuwaniem wartości określonych stwierdzeń, których wartości zostały przyjęte jako znane,
- uruchamianiem operacji związanych z operacją łączenia wartości stwierdzeń występujących w wielu warstwach modelu wielomodalnego.

W przypadku modeli wielowarstwowych sieci stwierdzeń proces uzgadniania wartości stwierdzeń polega na poszukiwaniu iloczynu poszczególnych przedziałów.

7.2.1. Podsumowanie

Ogólny model sieci stwierdzeń zawarty w bibliotece modeli pakietu MMNET zawiera pełną funkcjonalność umożliwiającą budowanie i stosowanie tego rodzaju sieci dla potrzeb diagnostycznych systemów doradczych.

Bibliografia

- [7.1] Package grain. Internet: <http://genetics.agrsci.dk/~sorenh/public/R/gRainweb/>, 11.2009.
- [7.2] Package grbase. Internet: <http://genetics.agrsci.dk/~sorenh/public/R/gRbaseweb/>, 12.2010.
- [7.3] Berkelaar M. *lpSolve: Interface to Lpsolve to solve linear or integer programs*, 2010.
- [7.4] Cholewa W. Ogólny model sieci stwierdzeń. Raport wewnętrzny MMN-03-031, Politechnika Śląska, Katedra Podstaw Konstrukcji Maszyn, Gliwice, 2010.
- [7.5] Henningsen A. *Linprog package: Linear Programming and Optimization*, 2010.
- [7.6] Jensen F., Nielsen T. *Bayesian Networks and Decision Graphs*. Springer, New York, USA, 2001.

Rozdział 8

Baza przykładów sieci stwierdzeń

Damian SKUPNIK

8.1. Wstęp

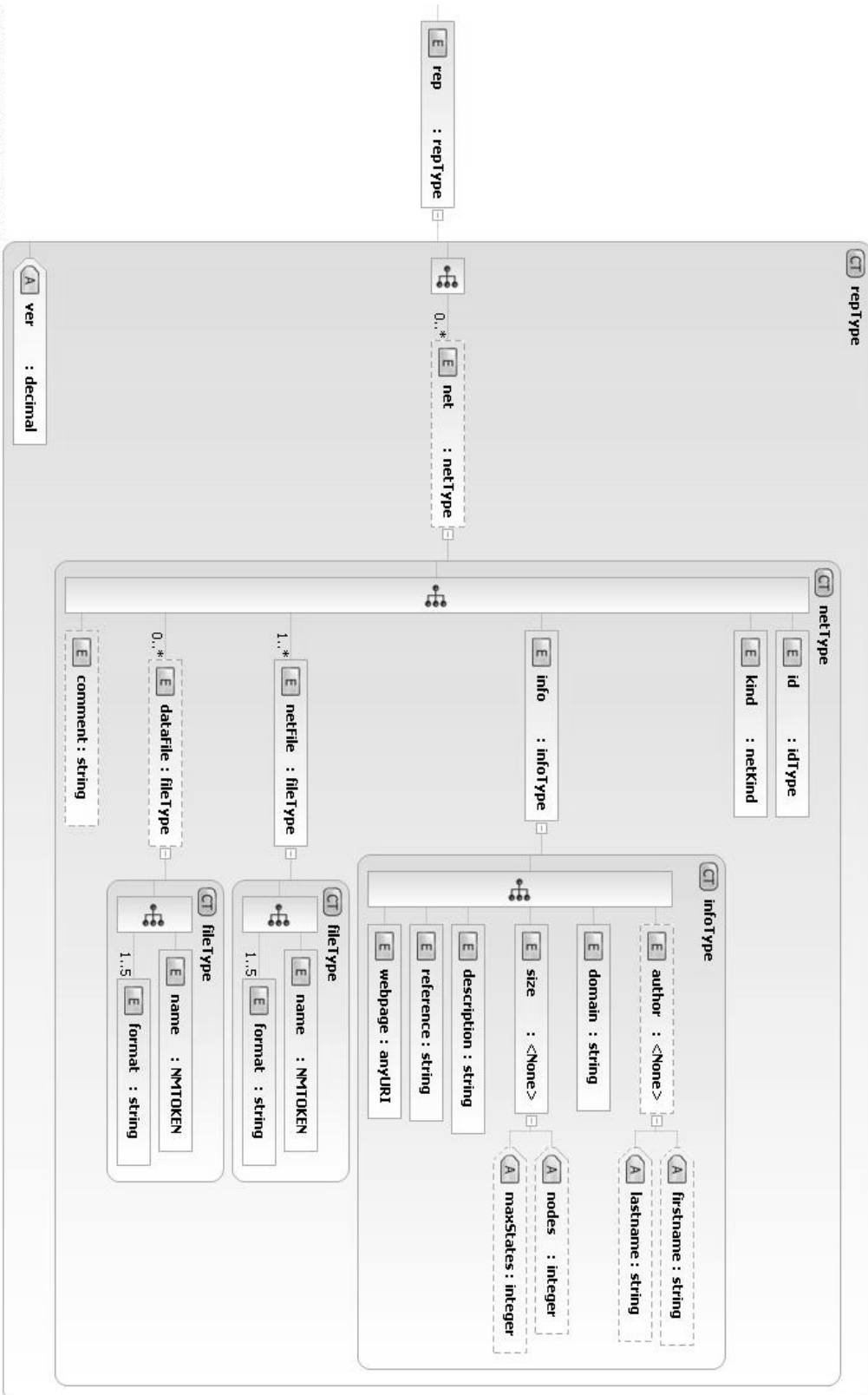
Uzyskanie zadowalających wyników działania diagnostycznych systemów doradczych opartych na wieloskalowych sieciach stwierdzeń wymaga opracowania specjalnych układów wnioskujących. W celu testowania ich skuteczności zgromadzony został zbiór 54 przykładów różnego rodzaju sieci (przede wszystkim bayesowskich). Informacje o zgromadzonych przykładach zawarto w bazie danych, dla której przyjęto hierarchiczną strukturę drzewiastą zapisaną w formacie XML [8.9]. Jest to elastyczne podejście, ponieważ jak zauważono w [8.3] można założyć, że w takim wypadku nie jest wymagana pełna specyfikacja struktury bazy przed jej zastosowaniem, a więc w zależności od potrzeb możliwe jest dodawanie nowych elementów do struktury oraz dopuszczalne jest stosowanie struktur niekompletnych.

Zapisanie bazy przykładów sieci stwierdzeń w formacie XML ma wiele zalet. Możliwa jest, np. za pomocą standardu XML Schema [8.10], weryfikacja (tzw. walidacja) bazy pod względem spójności i poprawności zawartej w niej treści. Zapytania pozwalające przeszukiwać bazę można formułować zgodnie ze standardem XPath [8.12], który stanowi rozszerzenie sposobu wybierania katalogów znanego z systemów operacyjnych DOS lub UNIX.

8.2. Struktura opracowanej bazy przykładów

Strukturę opracowanej bazy przykładów sieci stwierdzeń przedstawiono na rys. 8.1. Wersję dokumentu określa liczba dziesiętna (atrybut *ver*). Dokument może być „pusty” lub zawierać informacje o nieograniczonej liczbie sieci (element o nazwie *net*). Każdą sieć opisuje zbiór 6 elementów, przy czym występowanie elementów *dataFile* (dotyczy informacji o dostępnych plikach z danymi trenującymi) oraz *comment* nie jest konieczne (oznaczone za pomocą linii przerywanych).

Identyfikator każdej sieci (klucz) reprezentowany jest przez element *id*. Na postać identyfikatora nałożono ograniczenie, tzn. musi składać się on kolejno z 3 cyfr i 3 dużych liter, np. 015CAR. Niespełnienie tego warunku, brak identyfikatora lub jego powtórzenie spowoduje wyświetlenie komunikatu o błędzie. Nie dotyczy to jednak każdego walidatora (np. niektóre wersje XML Spy [8.11] nie zapewniają walidacji kluczy). Ograniczenie nałożono również na element *kind*, który musi przyjąć jedną z następujących wartości (wielkość liter ma znaczenie): Bayesian network, Markov network, Dynamic decision network.



Liquid XML Studio - FREE Community Edition 7.1.8.1440

Rys. 8.1: Definicja struktury bazy przykładów sieci stwierżeń

Informacja o pliku z daną siecią przechowywana jest w elemencie *netFile*, w którym zapisana jest nazwa pliku (podelement *name*) oraz nazwy co najwyżej 5 dostępnych formatów tego pliku (podelement *format*). Nazwa pliku nie może zawierać spacji.

Podstawowe dane o sieci przechowywane są w elemencie *info*, który składa się z 6 podelementów. Występowanie podelementu *author* nie jest konieczne. Podelement *domain* może przyjąć jedną z następujących wartości: biology, economics, education, sociology, technology, medicine, meteorology.

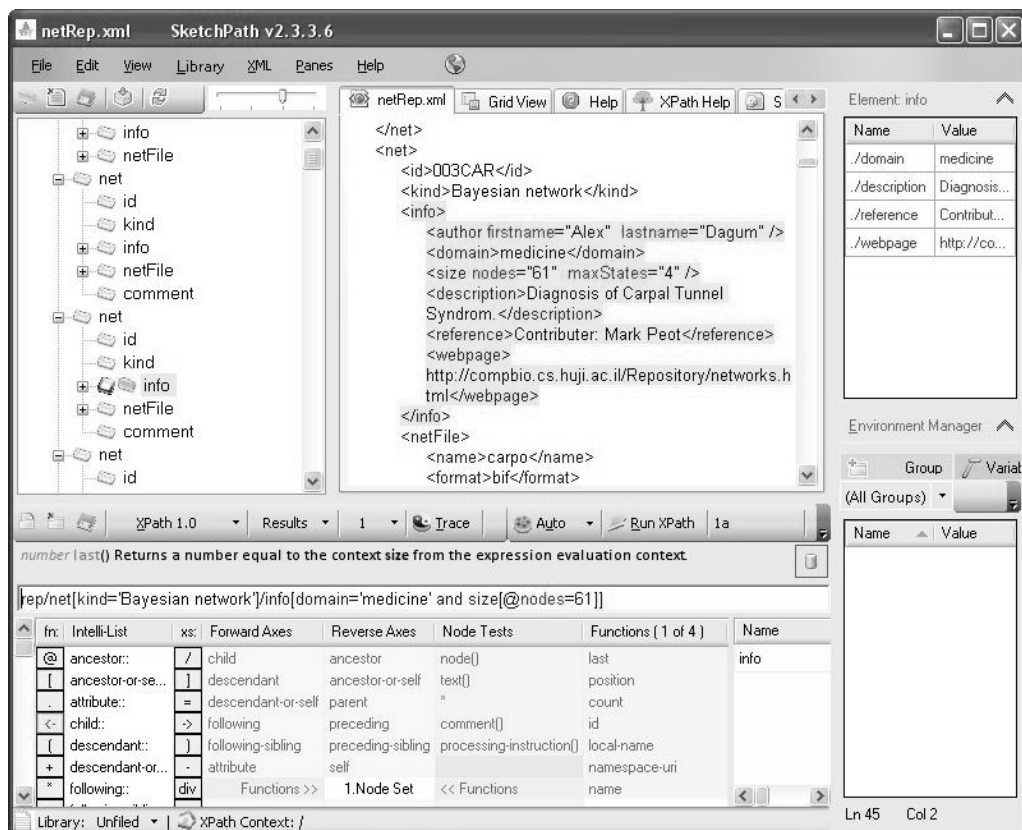
Podelement *size* zawiera ogólne dane o rozmiarze sieci, tzn. liczbę węzłów oraz maksymalną liczbę stanów w węźle. W podelemencie *description* zawarty jest opis zastosowania sieci, natomiast podelement *reference* przechowuje informację zapisaną w formacie BibTEX o najważniejszych publikacjach, które związane są z daną siecią. Ostatni podelement o nazwie *webpage* zawiera adres strony internetowej, z której ściągnięto pliki z daną siecią.

8.3. Przeszukiwanie bazy przykładów

Jak wspomniano wcześniej do przeszukiwania dokumentów XML można użyć zapytań sformułowanych zgodnie ze standardem XPath. Przykładowe zapytanie o sieć bayesowską z dziedziny „medycyna”, o liczbie węzłów równej 61 ma następującą postać:

```
rep/net[kind='Bayesian network']/info[domain='medicine' and size[@nodes=61]]
```

Wynik przeszukania opracowanego dokumentu XML dla powyższego przykładu przedstawia rys. 8.2.



Rys. 8.2: Wynik przeszukania opracowanej bazy (za pomocą programu SketchPath [8.8])

8.4. Ocena przydatności zgromadzonych przykładów

Spośród wszystkich zgromadzonych przykładów tylko część okazała się użyteczna podczas testowania układów wnioskujących opartych na wieloskalowych sieciach stwierdzeń. Dokonano wyboru najlepszych na podstawie zbioru 8 następujących kryteriów (w nawiasach podano oznaczenie kodowe rozpatrywanych kryteriów):

1. Rozpatrywana dziedzina wiedzy (K1; sieci dotyczące zagadnień ogólnych lub technicznych traktowano jako uprzywilejowane).
2. Liczba węzłów sieci (K2).
3. Maksymalna liczba stanów w węźle (K3).
4. Liczba gałęzi łączących węzły (K4).
5. Liczba „korzeni” (K5).
6. Najdłuższa ścieżka w sieci (K6).
7. Liczba dostępnych formatów sieci (K7).
8. Stopień wiarygodności sieci (K8; określany subiektywnie na podstawie publikacji, w której dana sieć była rozpatrywana).

8.4.1. Określenie relacji pomiędzy kryteriami

Wymienione kryteria oceny zgromadzonych przykładów nie powinny być traktowane jako jednakowo ważne (np. kryterium liczby dostępnych formatów sieci nie jest tak istotne, ponieważ w wielu wypadkach można dokonać konwersji sieci zapisanej w jednym formacie na inny format; przykładowy konwerter to [8.2]). Ponadto należy wyraźnie zaznaczyć, że ocena zgromadzonych przykładów ze względu na kryteria K1 oraz K8 jest subiektywna, a więc należy ostrożnie formułować wnioski na podstawie tak określonej noty.

W tabeli 8.1 przedstawiono jakościową relację pomiędzy rozpatrywanymi kryteriami. Cyfra 1 oznacza, że kryterium X (z pierwszej kolumny) jest mniej ważne niż kryterium Y (z pierwszego wiersza), 2 oznacza, że kryteria są równoważne, natomiast 3 określa relację odwrotną do relacji oznaczonej cyfrą 1.

Tab. 8.1: Relacje jakościowe pomiędzy rozpatrywanymi kryteriami

	K1	K2	K3	K4	K5	K6	K7	K8	$\Sigma = waga$
K1	x	2	1	1	1	1	3	2	11
K2	2	x	1	1	1	1	3	3	12
K3	3	3	x	1	2	2	3	3	17
K4	3	3	3	x	2	1	3	3	18
K5	3	3	2	2	x	1	3	3	17
K6	3	3	2	3	3	x	3	3	20
K7	1	1	1	1	1	1	x	2	8
K8	1	1	1	1	1	1	2	x	8

8.4.2. Skala ocen

Każdy zgromadzony w bazie przykład sieci oceniany jest wg kolejnych kryteriów. Biorąc pod uwagę fakt, że rozpatrywane kryteria dotyczą zarówno wartości jakościowych (K1 i K8) jak i ilościowych (pozostałe) konieczne jest określenie odpowiedniej skali ocen (tab. 8.2).

Tab. 8.2: Skala ocen dla rozpatrywanych kryteriów (DN — dziedzina wiedzy nieuprzywilejowana; DU — dziedzina wiedzy uprzywilejowana; LW — liczba węzłów; MLS — maks. liczba stanów; LG — liczba gałęzi; LK — liczba korzeni; DS — długość ścieżki; LF — liczba formatów; SMW — sieć mało wiarygodna; SW — sieć wiarygodna)

Kryterium	Ocena		
K1	1 (DN)		2 (DU)
K2	1 ($LW \leq 10$)	2 ($10 < LW \leq 100$)	3 ($LW > 100$)
K3	1 ($MLS = 2$)	2 ($2 < MLS \leq 4$)	3 ($MLS > 4$)
K4	1 ($LG \leq 10$)	2 ($10 < LG \leq 100$)	3 ($LG > 100$)
K5	1 ($LK \leq 2$)	2 ($2 < LK \leq 4$)	3 ($LK > 4$)
K6	1 ($DS \leq 2$)	2 ($2 < DS \leq 4$)	3 ($DS > 4$)
K7	1 ($LF \leq 2$)	2 ($2 < LF \leq 4$)	3 ($LF > 4$)
K8	1 (SMW)		2 (SW)

8.4.3. Oceny zgromadzonych przykładów

W tab. 8.3 przedstawiono posortowaną malejąco ocenę wszystkich przykładów sieci zgromadzonych w bazie. Kolumny zatytułowane K1, K2, . . . K8 zawierają ocenę rozpatrywanego przykładu ze względu na dane kryterium. Przedostatnia kolumna przedstawia ocenę końcową przykładu uzyskaną jako suma ważonych ocen cząstkowych (wartości wag — patrz tab. 8.1), natomiast ostatnia kolumna opisuje jaki procent maksymalnej możliwej oceny stanowi ocena danego przykładu.

Tab. 8.3: Ocena przydatności zgromadzonych przykładów sieci

ID przykładu	K1	K2	K3	K4	K5	K6	K7	K8	OCENA	PROCENT
015AND	2	3	3	3	3	3	1	2	298	95
016DTT	2	3	3	3	3	3	1	2	298	95
009MUN	1	3	3	3	3	3	2	2	295	94
004DIA	1	3	3	2	3	3	2	2	277	88
007LIN	1	3	2	3	3	3	1	2	270	86
005HAI	1	2	3	2	3	3	2	2	265	84
017HEP	1	2	2	3	3	3	1	2	258	82
002BAR	1	2	3	2	3	3	2	1	257	82
008MIL	1	2	3	2	3	3	2	1	257	82
001ALA	1	2	2	2	3	3	3	2	256	82
010PAT	1	3	3	3	1	3	1	2	253	81
013WIN	2	2	1	3	3	3	2	1	252	80
014POW	2	2	2	2	3	3	1	2	251	80
054RES	2	2	2	2	3	3	1	2	251	80
011PIG	1	3	2	3	3	2	2	1	250	80
055RIS	2	2	3	2	3	2	1	2	248	79
057SYS	2	2	3	2	3	2	1	2	248	79
062CAR	2	2	2	2	3	3	1	1	243	77
064OIL	2	2	2	2	2	3	1	2	234	75
006INS	1	2	3	2	1	3	1	2	223	71
012WAT	1	2	2	2	3	2	1	2	220	70
045AGR	1	2	2	2	3	2	1	2	220	70

Tab. 8.3: (ciąg dalszy)

ID przykładu	K1	K2	K3	K4	K5	K6	K7	K8	OCENA	PROCENT
042SPA	2	2	1	2	3	2	1	2	214	68
003CAR	1	2	2	2	3	2	1	1	212	68
053PRO	1	2	3	2	3	1	1	1	209	67
059WIL	1	2	2	2	1	3	1	2	206	66
048BUS	1	2	3	2	1	2	1	2	203	65
066SPI	1	2	3	2	1	2	1	2	203	65
069POK	2	1	3	2	1	2	1	2	202	64
060BOE	2	2	1	2	1	3	1	2	200	64
049COR	1	1	3	2	3	1	1	1	197	63
051CRE	2	2	2	2	1	2	1	2	197	63
052MID	2	2	2	2	1	2	1	2	197	63
063CHE	1	2	1	2	2	2	1	2	186	59
038CAR	2	1	2	2	1	2	1	2	185	59
071RIN	2	1	2	2	1	2	1	2	185	59
046FUN	1	1	2	1	3	2	1	1	182	58
047BAT	1	1	3	1	3	1	1	1	179	57
033LUC	1	2	1	2	2	2	1	1	178	57
067ZHA	2	1	1	2	1	2	1	2	168	54
068ZHA	2	1	1	2	1	2	1	2	168	54
065OIL	2	1	2	1	1	2	1	2	167	53
050DIA	1	1	2	2	2	1	1	1	163	52
056SUG	1	1	3	1	1	1	1	2	153	49
036CHE	1	2	1	1	1	2	1	2	151	48
039FIR	2	1	1	1	1	2	1	2	150	48
040NEA	2	1	1	1	1	2	1	2	150	48
041OIL	2	1	2	1	1	1	1	2	147	47
043UMB	2	1	2	1	1	1	1	1	139	44
058WHI	1	1	2	1	1	1	1	2	136	43
061BOO	2	1	1	1	1	1	1	2	130	41
044WET	2	1	1	1	1	1	1	1	122	39
037CAN	1	1	1	1	1	1	1	2	119	38
070AID	1	1	1	1	1	1	1	2	119	38

8.5. Zastosowanie zgromadzonych przykładów w systemie MMNET

Weryfikację działania układu wnioskującego, który został zaimplementowany w systemie MMNET, można przeprowadzić z zastosowaniem dowolnego przykładu zgromadzonego w bazie. Niestety wymaga to albo dosyć żmudnego wprowadzania danych za pośrednictwem GUI systemu MMNET, albo przetworzenia danych wejściowych na format obsługiwany przez system MMNET, przy czym preferowany jest format XML. Drugie z wymienionych podejść jest szczególnie uzasadnione w wypadku sieci stwierżeń o dużej liczbie węzłów.

Konwersję wejściowego formatu danych na format XML o wymaganej strukturze i treści można dokonać na wiele sposobów, np.:

- a) jeżeli formatem danych wejściowych jest również XML, to można opracować odpowiedni

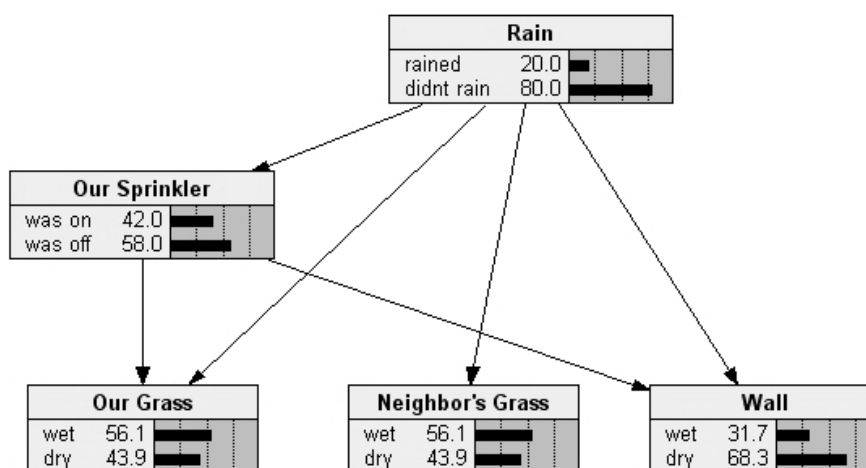
arkusz ze stylem dokumentu i użyć do konwersji dowolnego procesora XSLT (np. XSLT Processor [8.7]);

- b) stosując GUI systemu MMNET zapisać w formacie XML prosty tezaurus oraz model sieci stwierżeń a następnie zmodyfikować, powielić i uzupełnić o wymaganą treść odpowiednie fragmenty tych plików;
- c) jeżeli dane wejściowe zapisane są w formacie tekstowym, to do modyfikacji treści można użyć mechanizmu „Znajdź i zamień”, który dostępny jest w wielu edytorach tekstu;
- d) ...

Warto zwrócić uwagę, że wszystkie przykłady zgromadzone w bazie to sieci jednowarstwowe.

8.5.1. Przykład 1

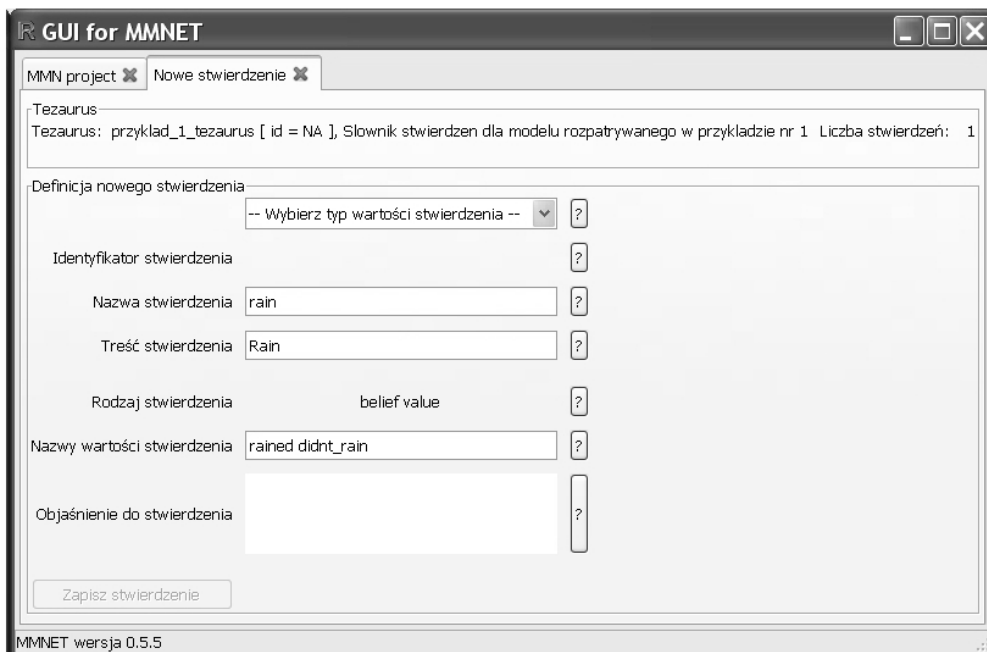
W celu ilustracji zastosowania GUI systemu MMNET wybrano z bazy sieć o identyfikatorze 044WET, która składa się z 5 węzłów (rys. 8.3). Przykład został przygotowany przez Norsys Software Corp. [8.4] na podstawie sieci opracowanej przez Judeę Perl'a [8.6]. Dotyczy sytuacji, w której na podstawie określonych przesłanek poszukiwana jest odpowiedź na pytanie czy padał deszcz.



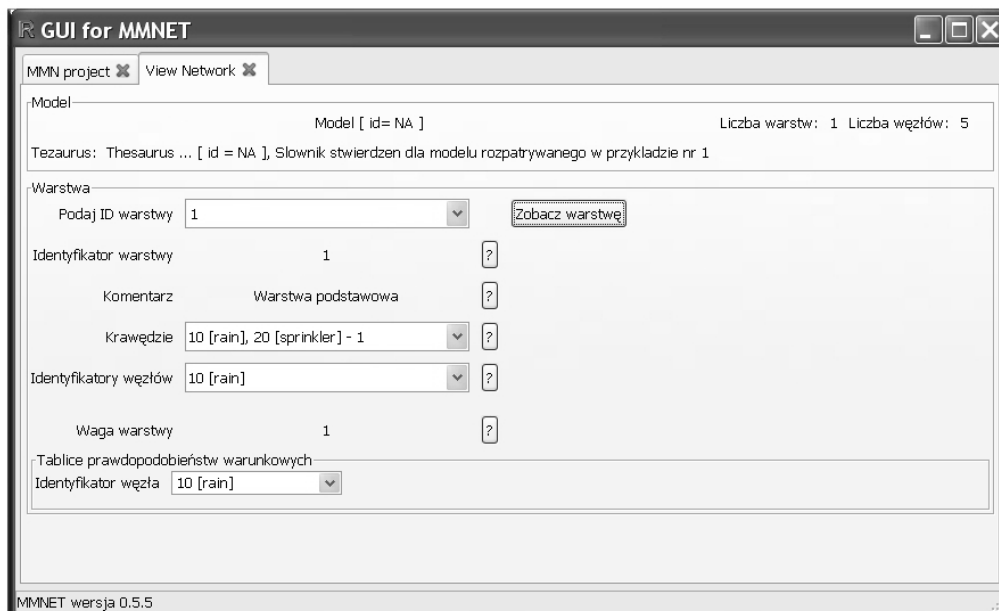
Rys. 8.3: Przykładowa sieć bayesowska [8.4]

Procedura tworzenia modelu reprezentowanego w postaci sieci wielowarstwowej w systemie MMNET opisana jest w instrukcji użytkownika. W skrócie można powiedzieć, że w pierwszej kolejności należy zdefiniować słownik stwierżeń (na rys. 8.4 przedstawiono zrzut ekranu po zdefiniowaniu stwierżenia o treści „Rain”) a następnie opracować model jednowarstwowej sieci (rys. 8.5) z zastosowaniem stwierżeń zawartych w tym słowniku.

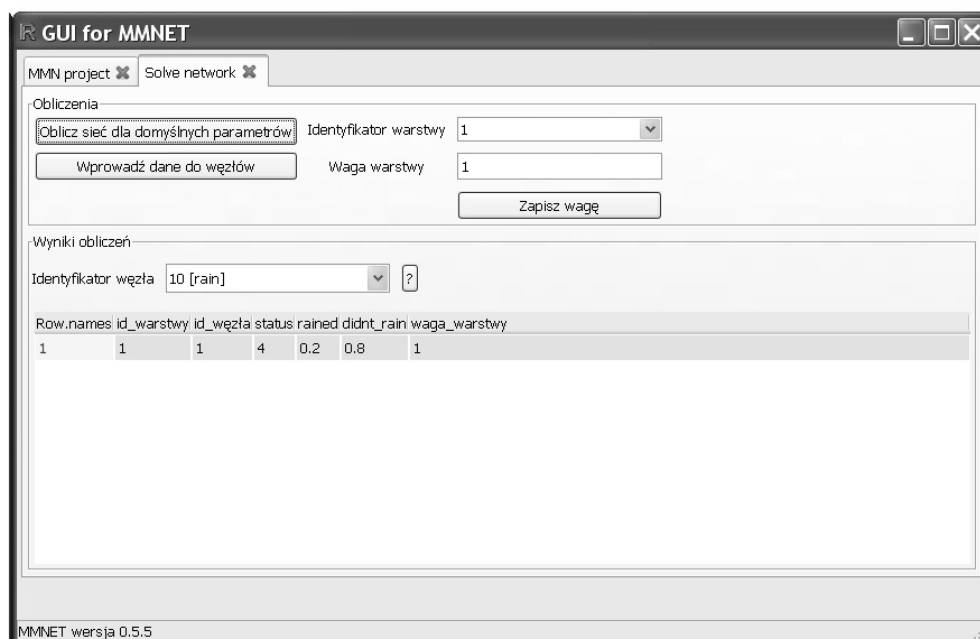
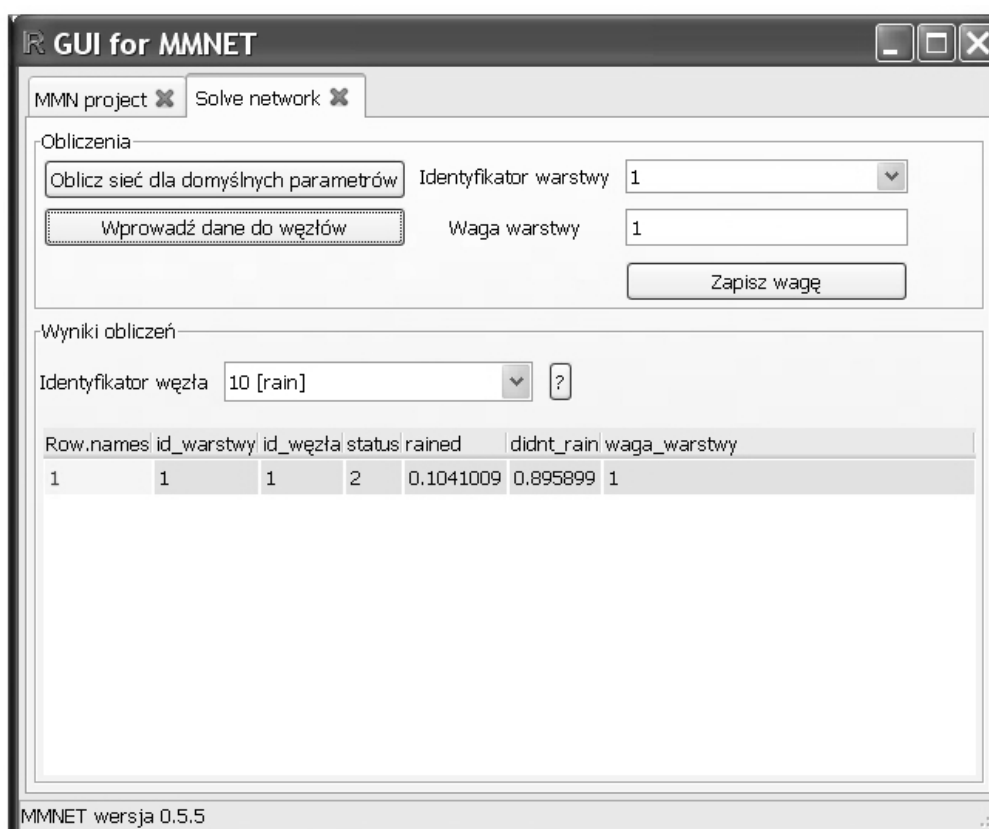
Na rys. 8.6 przedstawiono domyślny stan węzła *Rain* (zgodny ze stanem widocznym na rys. 8.3), natomiast rys. 8.7 przedstawia stan tego węzła po wprowadzeniu informacji, że ściana jest mokra.



Rys. 8.4: Sformułowanie twierdzenia o treści „Rain”



Rys. 8.5: Model jednowarstwowej sieci stwierdzeń dla przykładu zapisanego w bazie pod identyfikatorem 044WET

Rys. 8.6: Stan węzła *Rain* dla domyślnych wartości stwierżeńRys. 8.7: Stan węzła *Rain* w sytuacji gdy stwierdzono, że ściana jest mokra

8.5.2. Przykład 2

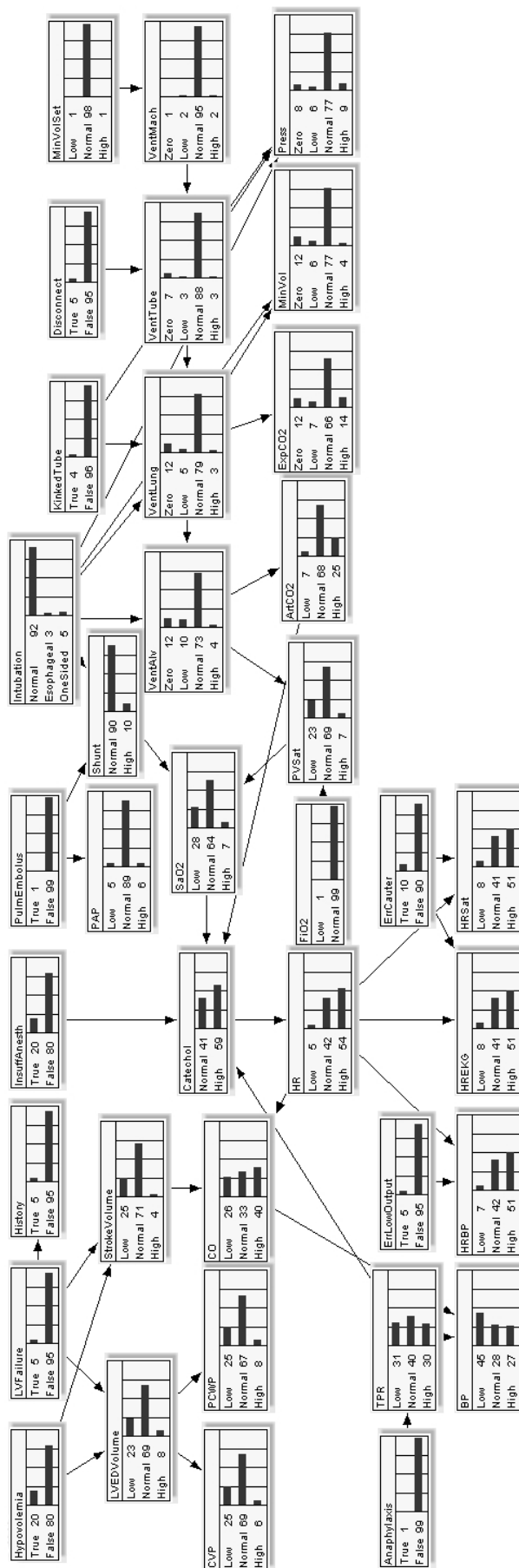
Drugi przykład został zaczerpnięty z [8.1] i dotyczy sieci składającej się z 37 węzłów, zapisanej w bazie pod identyfikatorem 001ALA (rys. 8.8). Tym razem zastosowano podejście b) opisane w punkcie 8.5. Do edycji pliku XML użyto programu Liquid XML Studio 2009 [8.5]. Warto zauważyć, że najprostszy słownik stwierdzeń, na podstawie którego tworzony jest model sieci stwierdzeń, musi zawierać co najmniej 3 stwierdzenia.

Poniżej przedstawiono fragment pliku XML (wygenerowany za pośrednictwem GUI systemu MMNET) z szablonem słownika.

```
<?xml version="1.0"?> <!-- Created with Liquid XML Studio - FREE
Community Edition 7.0.2.746 (http://www.liquid-technologies.com) -->
<MMNET VERSION="1.0">
  <STATICPROPERTIES>
    <source value="Core ver. 0.5, Pawel Chrzanowski (KPKM)" />
    <date value="Tue Dec 28 17:24:56 2010" />
  </STATICPROPERTIES>
  <THESAURUS stmts="3" notice="" title="szablon_stwierdzenia" idThesa="NA">
    <STATEMENT>
      <STMTYPE>CStmtValMBelief_TR</STMTYPE>
      <ID>10</ID>
      <MODE>2</MODE>
      <DESCRIPTION>belief value</DESCRIPTION>
      <LAYERS>0</LAYERS>
      <SHORTNAME>nazwa_stwierdzenia</SHORTNAME>
      <CONTENT>Tresc stwierdzenia</CONTENT>
      <HELP></HELP>
      <VSTMTID></VSTMTID>
      <VALNAMES>
        <VALNAME>Yes</VALNAME>
        <VALNAME>No</VALNAME>
      </VALNAMES>
    </STATEMENT>
    ...
  </THESAURUS>
</MMNET>
```

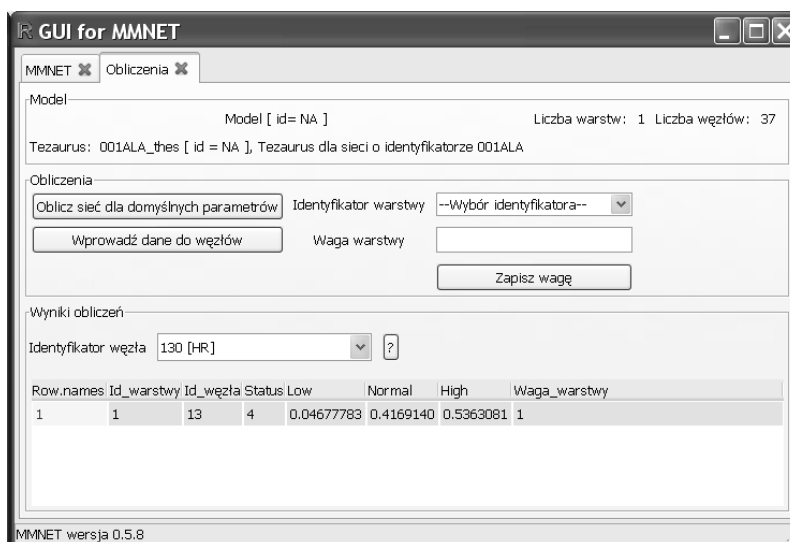
Po modyfikacji (wprowadzeniu odpowiednich danych) wygląda on następująco:

```
<?xml version="1.0"?> <!-- Created with Liquid XML Studio - FREE
Community Edition 7.0.2.746 (http://www.liquid-technologies.com) -->
<MMNET VERSION="1.0">
  <STATICPROPERTIES>
    <source value="Core ver. 0.5, Pawel Chrzanowski (KPKM)" />
    <date value="Tue Dec 28 17:27:57 2010" />
  </STATICPROPERTIES>
  <THESAURUS stmts="37" notice="Tezaurus dla sieci o identyfikatorze 001ALA"
title="001ALA_thes" idThesa="NA">
    <STATEMENT>
      <STMTYPE>CStmtValMBelief_TR</STMTYPE>
      <ID>10</ID>
      <MODE>2</MODE>
      <DESCRIPTION>belief value</DESCRIPTION>
      <LAYERS>0</LAYERS>
      <SHORTNAME>Anaphylaxis</SHORTNAME>
      <CONTENT>Anaphylaxis</CONTENT>
      <HELP/>
      <VSTMTID></VSTMTID>
      <VALNAMES>
        <VALNAME>True</VALNAME>
        <VALNAME>False</VALNAME>
      </VALNAMES>
    </STATEMENT>
    ...
  </THESAURUS>
</MMNET>
```



Rys. 8.8: Sieć bayesowska umożliwiająca redukcję fałszywych alarmów na oddziałach intensywnej opieki medycznej [8.1]

Analogiczne działania przeprowadzono na pliku z modelem i w rezultacie otrzymano plik XML z modelem jednowarstwowej sieci stwierdzeń. Po wczytaniu pliku przeprowadzono testowanie układu wnioskującego. Na rys. 8.9 przedstawiono stopnie przekonania o prawdziwości stwierdzenia, że pacjent ma podwyższone tętno, w sytuacji, gdy nie są dostępne żadne informacje na temat tego pacjenta (równy w przybliżeniu 0.54).



Rys. 8.9: Stan węzła *HR* w sytuacji braku dodatkowych danych

Bibliografia

- [8.1] Beinlich I., Suermondt G., Chavez R., Cooper G. The ALARM monitoring system: A case study with two probabilistic inference techniques for belief networks. *Proceedings of the 2nd European Conference on AI and Medicine*, 1989.
- [8.2] BIF-BNT converter. <http://www.digitas.harvard.edu/~ken/bif2bnt/>, grudzień 2010.
- [8.3] Cholewa Wojciech. Zbiory danych. Kiciński Jan, redaktor, *Modelowanie i diagnostyka oddziaływań mechanicznych, aerodynamicznych i magnetycznych w trubozespołach energetycznych*, rozdział 25, strony 1227–1260. Instytut Maszyn Przepływowych Polskiej Akademii Nauk, Gdańsk, 2005.
- [8.4] Corp. Norsys Software. <http://www.norsys.com>, grudzień 2010.
- [8.5] Liquid XML Studio. <http://www.liquid-technologies.com/XmlStudio/XmlStudio.aspx>, grudzień 2010.
- [8.6] Pearl J. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann, San Mateo, CA, 1988.
- [8.7] Processor XSLT. http://www.download3000.com/download_24505.html, grudzień 2010.
- [8.8] SketchPath. <http://qutoric.com/sketchpath/>, grudzień 2010.
- [8.9] XML. <http://www.w3.org/XML/>, grudzień 2010.
- [8.10] XML Schema. <http://www.w3.org/XML/Schema>, grudzień 2010.
- [8.11] XML Spy. <http://www.altova.com/xml-editor/>, grudzień 2010.
- [8.12] XPath. <http://www.w3.org/TR/xpath20/>, grudzień 2010.

Rozdział 9

Porównanie wybranych modeli sieci stwierdzeń

Tomasz ROGALA

Reprezentacją tej samej rozpatrywanej dziedziny wiedzy mogą być różne rodzaje modeli graficznych. Postać grafu modeli, sposób ich rozwiązywania oraz sposób ich konstruowania różnią się i wymagają osobnego opisu. Na przykład dla grafów nieskierowanych warunkową niezależność pomiędzy poszczególnymi stwierdzeniami można odczytać bezpośrednio ze struktury grafu i wynika ona bezpośrednio ze ścieżki pomiędzy stwierdzeniami w grafie. W przypadku sieci kierunkowych np. sieci przekonań, określenie warunkowej niezależności pomiędzy rozpatrywanymi stwierdzeniami wymaga przeprowadzenia wnikliwej analizy [9.2], [9.5], [9.3], a w przypadku rozległych sieci zastosowania odpowiednich algorytmów. Przykład ten pokazuje, jak bardzo różne może być znaczenie m.in. struktury grafu w zależności od rodzaju modelu graficznego.

9.1. Struktury grafów i warunkowa niezależność pomiędzy zmiennymi

Modele graficzne oferowane w ramach biblioteki MMNET: sieci przekonań i ogólne sieci stwierdzeń [9.1] wymagają odrębnych sposobów konstruowania modeli, specyficznych dla każdego rodzaju modelu. Na rysunku 9.1a) pokazano powszechnie znany przykład sieci będącej modelem zraszacza (ang. sprinkler) [9.4], która jest przykładem sieci opisującej prawdopodobieństwo wystąpienia mokrej trawy w zależności od warunków pogodowych oraz działania zraszacza. Poszczególne zmienne występujące na rysunku 9.1 reprezentują stwierdzenia o następującej treści:

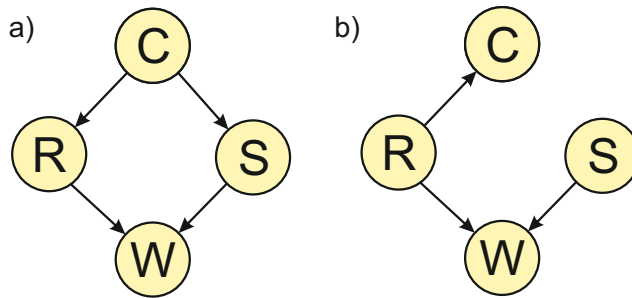
C Niebo jest zachmurzone,

R Pada deszcz,

S Działa zraszacz,

W Trawa jest mokra.

Poszczególne węzły reprezentują stwierdzenia proste o stanach *tak*, *nie*. Na rysunku 9.1b) pokazano strukturę ogólnej sieci stwierdzeń opisującej ten sam obiekt. Struktura grafu wynika m.in. z tego, że stwierdzenie *R-pada deszcz* jest warunkiem wystarczającym dla stwierdzenia *C-niebo jest zachmurzone* i niekoniecznie odwrotnie. Podobnie można stwierdzić, że stwierdzenie *R-pada deszcz* jest warunkiem wystarczającym dla stwierdzenia *W-trawa jest mokra* ale nie odwrotnie ponieważ takie wnioskowanie może być zawodne tzn. stwierdzenie *W-trawa jest*



Rys. 9.1: Sieć dotycząca przykładu pt. Zraszacz. a) Struktura grafu dla sieci przekonań, b) Struktura grafu dla ogólnej sieci stwierdzeń.

mokra może być wynikiem, że *S-dziąta spryskiwacz*. W przypadku ogólnej sieci stwierdzeń nie zdefiniowano asocjacji pomiędzy stwierdzeniem *S-dziąta spryskiwacz* i *C-niebo jest zachmurzone* ponieważ trudno w takim przypadku zdefiniować warunek wystarczający pomiędzy nimi. W przypadku sieci stwierdzeń możliwe jest zdefiniowanie związku pomiędzy stwierdzeniami *C-niebo jest zachmurzone* i *S-dziąta zraszacz* ponieważ stopień przekonania o tym, że *S-dziąta zraszacz* pod warunkiem że *C-niebo jest zachmurzone* może być mniejsze od przypadku gdy *S-dziąta zraszacz* pod warunkiem, że nieprawda że, *C-niebo jest zachmurzone*.

Analizując warunkową niezależność pomiędzy rozpatrywanymi węzłami obu grafów, w sposób łatwy można ją odczytać w przypadku sieci stwierdzeń, ale już niekoniecznie w przypadku sieci przekonań zwłaszcza, gdy te ostatnie są bardzo rozległymi sieciami. Biorąc pod uwagę warunkową niezależność pomiędzy stwierdzeniami *W* i *C* w analizowanych grafach można wykazać, że w przypadku sieci przekonań:

$$P(W \perp C | R, S), \quad (9.1)$$

a w przypadku sieci stwierdzeń

$$P(W \perp C | R). \quad (9.2)$$

Pokazany przykład wyraźnie wskazuje na różnice w interpretacji struktur grafów dla ogólnej sieci stwierdzeń oraz sieci przekonań.

9.2. Porównanie wyników dla różnych stanów sieci

Na rysunku 9.2 pokazano prosty, ale nieco bardziej rozbudowany przykład sieci pt. Zameć. W rozpatrywanym przykładzie celowo przyjęto, że rozpatrywana struktura sieci przekonań jest identyczna ze strukturą ogólnej sieci stwierdzeń. Jest to możliwe ze względu na to, że w przypadku sieci przekonań, prawdopodobieństwo łączne może być wyrażone jako [9.3], [9.5]:

$$P(A \cap B) = P(A|B)P(B) = P(B|A)P(A), \quad (9.3)$$

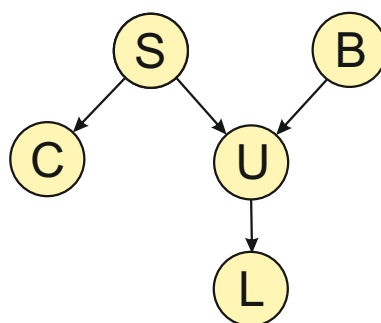
co oznacza, że łączny rozkład prawdopodobieństwa może być wyrażony za pomocą grafów o różnej strukturze.

W rozpatrywanej sieci, węzły reprezentują stwierdzenia proste o stanach *tak, nie*. Poszczególne zmienne występujące na rysunku 9.2 oznaczają:

C Niebo jest zachmurzone,

S Pada śnieg,

U Ulica jest pokryta śniegiem,



Rys. 9.2: Sieć dotycząca przykładu pt. Zamieć

B Jest zamieć, tumany śniegu są unoszone przez podmuch silnego wiatru,

L Na ulicy jest ślisko.

W tabeli 9.1 pokazano domyślny stan węzłów sieci. Jest to stan, w którym wszystkie wartości stwierdzeń są przyjęte jako niewiadome. W przypadku sieci przekonań wyznaczany jest stan równowagi, a wartości stwierdzeń przyjmują wartości wynikające z przyjętych parametrów w tablicach prawdopodobieństw warunkowych. Dla sieci stwierdzeń stan ten odpowiada przyjęciu wartości początkowych jako przedział $[0, 1]$.

Tab. 9.1: Wyniki dla domyślnych wartości stwierdzeń. Dla sieci przekonań podano wartości odpowiadające wystąpieniu stanu *tak*, w przypadku sieci stwierdzeń podano przedział wartości

Stwierdzenie	S	C	B	U	L
Sieć przekonań <i>tak</i>	0.2	0.36	0.1	0.25	0.32
Sieć stwierdzeń	$[0,1]$	$[0,1]$	$[0,1]$	$[0,1]$	$[0,1]$

W tabeli 9.2 pokazano wartości dla zadanego stanu sieci w którym stwierdzenie *C- niebo jest zachmurzone tak=1*, oraz *B- jest zamieć... tak=1* są wartościami znanymi. Zaznaczono je wytłuszczoną czcionką. W wyniku poszukiwania równowagi sieci pozostałe stwierdzenia przyjmują wartości pokazane w tabeli 9.2. Należy zwrócić uwagę, że stwierdzeniu *S-pada śnieg*, które dla sieci stwierdzeń przyjmuje wartość 0.56 , odpowiada wartość z przedziału $[0,1]$. W pozostałych przypadkach wartości przedziałów dla stwierdzeń w ogólnej sieci stwierdzeń obejmują wartość wyznaczoną za pomocą sieci stwierdzeń i nie są szerokie. Na przykład dla stwierdzenia *U-ulica jest pokryta śniegiem* wartość stwierdzenia dla sieci przekonań jest równa $tak=0.86$, a dla sieci stwierdzeń jest równa $[0.7-1]$.

Tab. 9.2: Wyniki dla sieci z zadanymi wybranymi wartościami stwierdzeń. Dla sieci przekonań podano wartości odpowiadające wystąpieniu stanu *tak*, w przypadku sieci stwierdzeń podano przedział wartości. Wytłuszczone wartości odpowiadają wartościom zadanym.

Stwierdzenie	S	C	B	U	L
Sieć przekonań <i>tak</i>	0.56	1	1	0.86	0.86
Sieć stwierdzeń	$[0,1]$	$[0.9,1]$	$[0.7,0.9]$	$[0.7,1]$	$[0.7,1]$

Kolejnym zadanym stanem sieci jest stan, który w przypadku sieci stwierdzeń prowadzi do sprzeczności i wymusza uruchomienie zadania programowania liniowego dla potrzeb poszukiwania rozwiązania przybliżonego. W tabeli 9.3 pokazano wartości dla zadanego stanu sieci, w którym stwierdzenie *C- niebo jest zachmurzone tak=1*, oraz *B- jest zamieć... tak=1* oraz *L-*

na ulicy jest ślisko... $tak=0$ są wartościami znanymi. Zaznaczono je wytłuszczoną czcionką. Wartości zadane dla sieci stwierdzeń wyrażono w postaci przedziałów, które w mniejszym lub większym stopniu są zgodne z wartościami punktowymi zadanymi w sieci przekonań.

Tab. 9.3: Wyniki dla sieci z zadanymi wybranymi wartościami wymuszającymi sprzeczność. Dla sieci przekonań podano wartości odpowiadające stanu tak , w przypadku sieci stwierdzeń podano przedział wartości. Wytłuszczone wartości odpowiadają wartościom zadanym.

Stwierdzenie	S	C	B	U	L
Sieć przekonań tak	0.11	1	1	0.12	0
Sieć stwierdzeń	[0,1]	[0.9,1]	[0.7,0.9]	[0.2,0.4]	[0.2,0.4]

Uzyskane wyniki pokazują dużą zgodność w przypadku stwierdzenia wtórnego U - *ulica jest pokryta śniegiem...* Wartość wyznaczona przez sieć przekonań jest równa $tak=0.12$, natomiast w przypadku sieci stwierdzeń znajduje się w przedziale $[0.2,0.4]$. W przypadku stwierdzenia wtórnego S - *pada śnieg...* przedział wartości nie został zawężony. Na przedstawione wyniki można również wpłynąć poprzez odpowiednie różnicowanie ważności poszczególnych asocjacji pomiędzy stwierdzeniami, poprzez nadawanie odpowiednich wag poszczególnym krawędziom grafu sieci stwierdzeń.

9.3. Podsumowanie

W niniejszym rozdziale pokazano przykłady sieci przekonań oraz sieci stwierdzeń przeznaczonych do opisu takiego samego zadania. W wyniku przeprowadzonego porównania pokazano, że struktury grafów dla obu rodzajów sieci posiadają różne znaczenie, a ze względu na własności opisu prawdopodobieństwa łącznego w sieciach przekonań, możliwe jest w przypadku sieci przekonań budowanie różnych postaci grafów reprezentujących tę samą dziedzinę wiedzy¹ czego nie da się stwierdzić w przypadku ogólnych sieci stwierdzeń. W przypadku tych ostatnich w łatwy sposób na podstawie ścieżek pomiędzy rozpatrywanymi stwierdzeniami można określić ich warunkową niezależność.

Z analizy uzyskanych wyników wnioskowania na rozpatrywanych przykładach wynika, że sieci stwierdzeń dają podobne wyniki do wyników uzyskanych za pomocą sieci przekonań. Chociaż w pierwszym przypadku są wyrażone za pomocą przedziałów wartości, to najczęściej wyniki punktowe dla sieci stwierdzeń znajdują się wewnątrz tych przedziałów. Przypadki takie mają miejsce dla dokładnego rozwiązania sieci stwierdzeń oraz rzadziej dla przybliżonego rozwiązania sieci stwierdzeń, w których występuje warunkowa sprzeczność. Należy podkreślić, że w przypadku sieci stwierdzeń nie wymagają one konieczności definiowania wielu parametrów tak jak ma to miejsce podczas definiowania tablic prawdopodobieństw warunkowych, gdzie wartości te są najczęściej trudne do pozyskania.

Bibliografia

- [9.1] Cholewa W. Ogólny model sieci stwierdzeń. Raport wewnętrzny MMN-03-031, Politechnika Śląska, Katedra Podstaw Konstrukcji Maszyn, Gliwice, 2010.
- [9.2] Jensen F., Nielsen T. *Bayesian Networks and Decision Graphs*. Springer, New York, USA, 2001.

¹ Należy zauważyć, że nadrzędnym celem konstruowania sieci stwierdzeń jest uzyskiwanie możliwie najprostszego opisu rozpatrywanej dziedziny, co w omawianym zagadnieniu powinno ograniczyć liczbę możliwych do rozparzenia struktur grafów sieci przekonań.

-
- [9.3] Korb K., Nicholson A. *Bayesian artificial intelligence*. Chapman & Hall, Boca Raton, USA, 2004.
- [9.4] Murphy K. A brief introduction to graphical models and bayesian networks. Internet: <http://www.arne-henningsen.nam>, 12.2010.
- [9.5] Pearl J. *Causality: Models, Reasoning and Inference*. Cambridge University Press, Cambridge, UK, 2000.

Rozdział 10

Przykłady sieci przekonań przeznaczonych do wspomagania procesu koncyptowania w zakresie kompletowania zespołów

Adam CHOLEWA

Opracowywane przykłady sieci przekonań dotyczą wspomagania procesu koncyptowania w zakresie kompletowania zespołów. W trakcie opracowywania założeń dotyczących postaci sieci wybrano grupę maszyn i urządzeń, których będzie dotyczyła opracowywana sieć. Zdecydowano się na wybór procesu koncyptowania podczas kompletowania przenośników traktowanych jako urządzenia służące do transportu bliskiego i przeznaczone do przemieszczania materiałów sypkich lub określonych ładunków (o dużych i małych gabarytach) po określonym torze.

10.1. Założenia dotyczące opracowywanych przykładów

W procesie projektowania przenośników a w szczególności rozpatrując konieczność doboru odpowiednich elementów, z których te urządzenia się składają, należy wziąć pod uwagę wiele czynników.

Wybrano dwie dziedziny przemysłu, w których bardzo powszechnie są stosowane tego typu grupy urządzeń. Opracowano dwie sieci stwierdzeń. Jedna z sieci pozwala na dobór przenośnika w przemyśle spożywczym, druga natomiast pozwala na przeprowadzenie procesu doboru przenośnika dla przemysłu górniczego.

Dla wybranych grup przemysłu opracowano wymagania odnośnie stosowania poszczególnych elementów maszyn w trakcie procesu projektowego. Pod uwagę wzięto różne typy przenośników.

10.2. Przyjęta metoda doboru elementów kompletowanego przenośnika

Dla rozpatrywanych dziedzin przemysłu uznano, że dobór elementów przenośnika jest możliwy na podstawie znajomości następujących parametrów jego pracy:

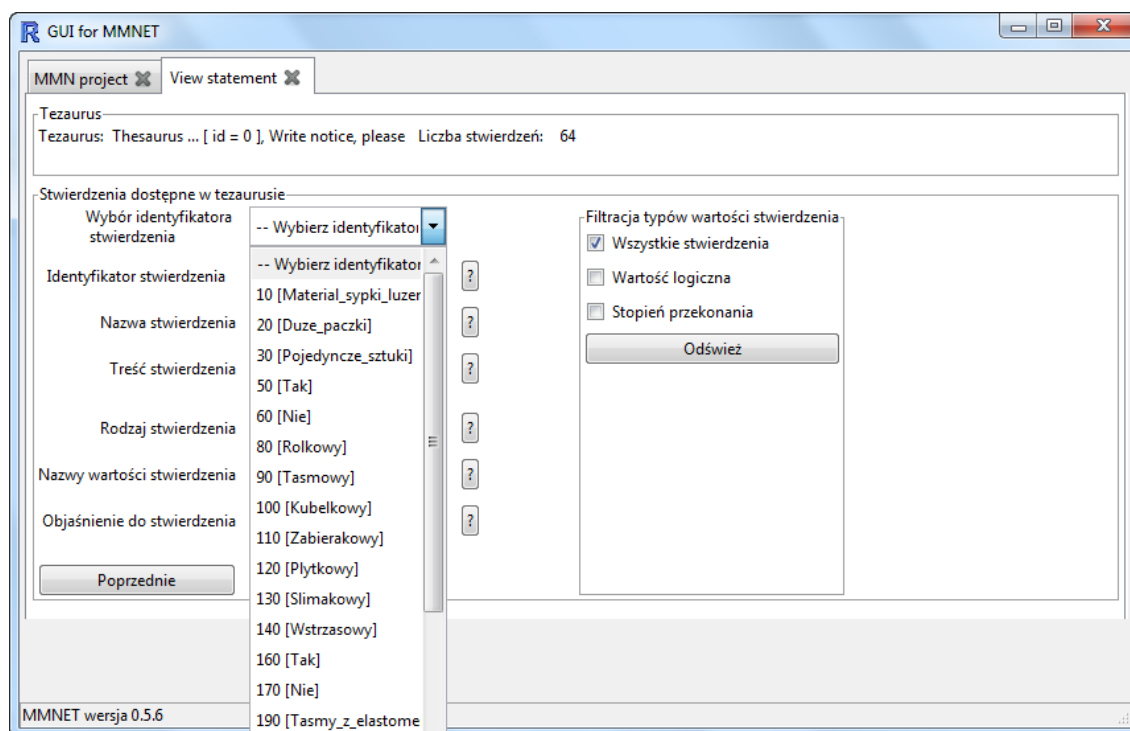
- rodzaju transportowanego ładunku,
- trybu pracy urządzenia,
- gabarytów transportowanego materiału,

- masy transportowanego materiału,
- wilgotności transportowanego materiału.

Węzły projektowanej sieci, zawierające przedstawione informacje wyróżniono jako węzły wejściowe. Do opracowania przykładu zastosowano pakiet MMNET v.0.5.6.

10.2.1. Słownik stwierdzeń

Zdefiniowano łącznie 49 stwierdzeń prostych (tzw. słownik - tezaursus), które stanowiły podstawę do budowy sieci przekonań do doboru elementów przenośnika. W oparciu o tak zdefiniowany słownik utworzono 15 stwierdzeń złożonych.

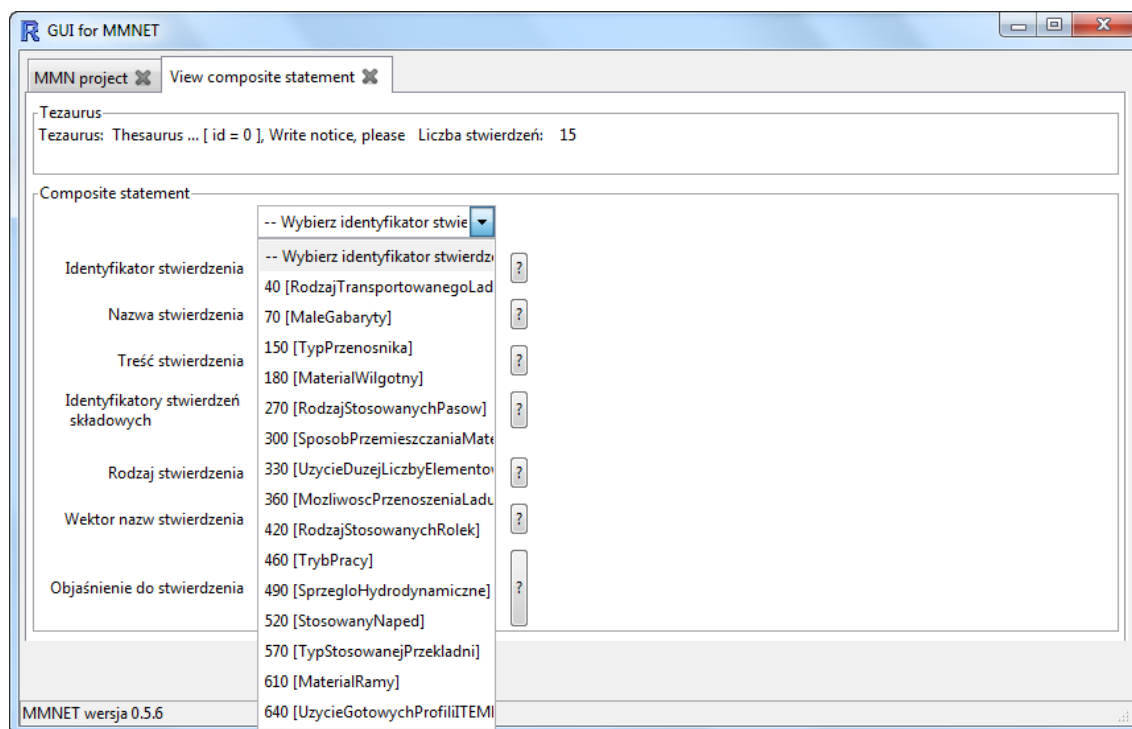


Rys. 10.1: MMNET. Stwierdzenia proste

Każde stwierdzenie złożone zostało opisane za pomocą unikalnego identyfikatora ID, skróconej nazwy stwierdzenia oraz treści stwierdzenia w następującym formacie ID [skrótowa nazwa stwierdzenia]. Treści stwierdzeń złożonych odpowiadają ich nazwie skróconej (np. nazwie skróconej **RodzajTransportowanegoŁadunku** odpowiada treść „*Rodzaj transportowanego ładunku*”). Ponadto, na każde stwierdzenie złożone składa się różna liczba stwierdzeń prostych.

Węzeł:	40 [RodzajTransportowanegoŁadunku]
Wartości:	10 [Materiał sypki luzem] 20 [Duże paczki] 30 [Pojedyncze sztuki]

Węzeł:	330 [UzycieDuzejLiczbyElementowZnormalizowanych]
Wartości:	310 [Tak] 320 [Nie]



Rys. 10.2: MMNET. Stwierdzenia złożone

Węzeł:	70 [MaleGabaryty]
Wartości:	50 [Tak] 60 [Nie]

Węzeł:	460 [TrybPracy]
Wartości:	430 [Ciągła] 440 [Częste cykle] 450 [Okresowa]

Węzeł:	360 [MozliwoscPrzenoszeniaLadunkuODuzejMasie]
Wartości:	340 [Tak] 350 [Nie]

Węzeł:	640 [UzycieGotowychProfiliiITEMlubBoschRexroth]
Wartości:	620 [Tak] 630 [Nie]

Węzeł:	610 [MaterialRamy]
Wartości:	580 [Stal] 590 [Aluminium] 600 [Stal nierdzewna]

Węzeł:	520 [StosowanyNapęd]
Wartości:	500 [Silnik asynchroniczny] 510 [Serwonapęd]

Węzeł:	570 [TypStosowanejPrzekładni]
Wartości:	530 [Ślimakowa] 540 [Brak przekładni] 550 [Walcowa] 560 [Planetarna]

Węzeł:	490 [SprzęgłoHydrodynamiczne]
Wartości:	470 [Jest] 480 [Brak]

Węzeł:	180 [MaterialWilgotny]
Wartości:	160 [Tak] 170 [Nie]

Węzeł:	270 [RodzajStosowanychPasow]
Wartości:	190 [Taśmy z elastomerów] 200 [Brak pasów] 210 [Taśmy z PTFE] 220 [Taśmy modularne] 230 [Taśmy tkaninowo gumowe] 240 [Taśmy siatkowo metalowe] 250 [Taśmy siatkowe z poliestru] 260 [Taśmy specjalne]

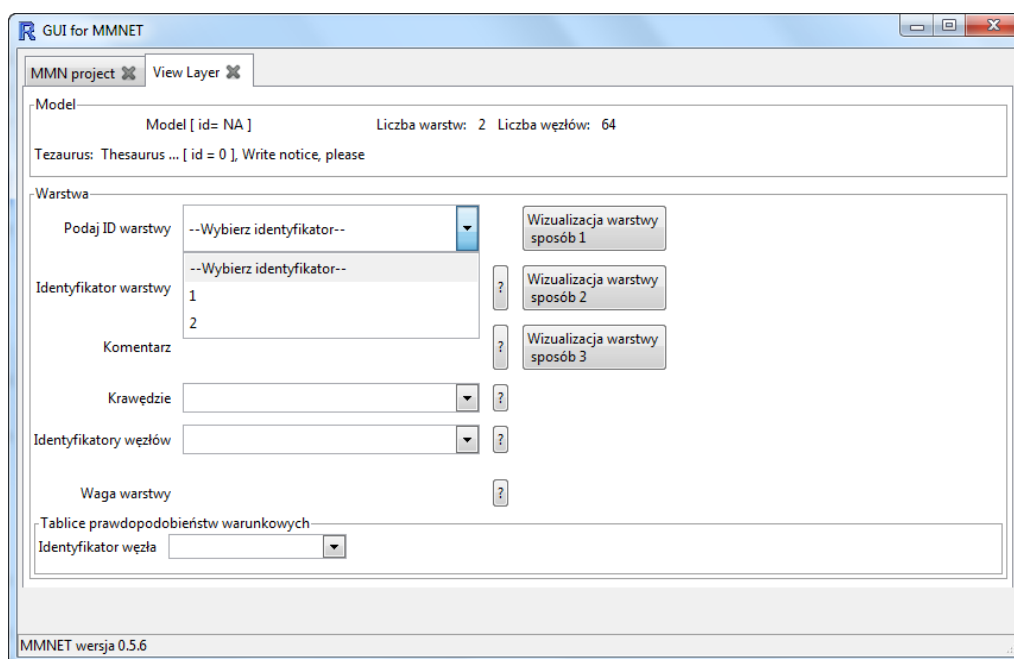
Węzeł:	300 [SposobPrzemieszczaniaMaterialuNaPrzenosniku]
Wartości:	280 [Grawitacyjnie] 290 [Napęd elektrorolkami]

Węzeł:	420 [RodzajStosowanychRolek]
Wartości:	370 [Krążnik stalowy gładki] 380 [Brak rolek] 390 [Krążnik tarczowy gumowy] 400 [Krążnik z tworzywa sztucznego] 410 [Krążnik pierścieniowy gumowy]

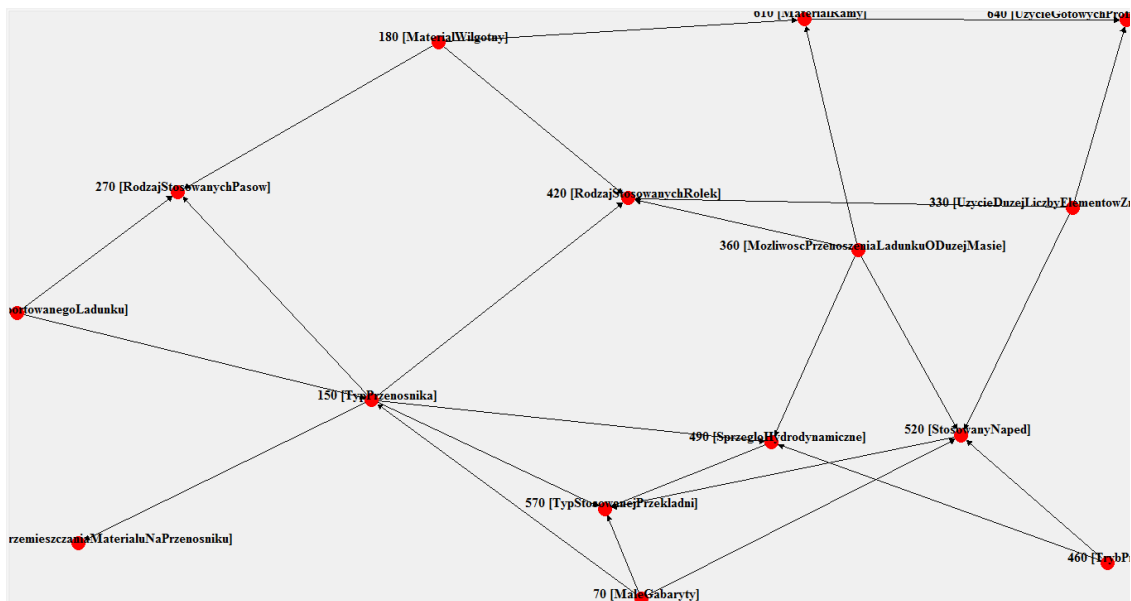
Węzeł:	150 [TypPrzenośnika]
Wartości:	80 [Rolkowy] 90 [Taśmowy] 100 [Kubelkowy] 110 [Zabierakowy] 120 [Płytkowy] 130 [Ślimakowy] 140 [Wstrząsowy]

10.2.2. Postaci sieci

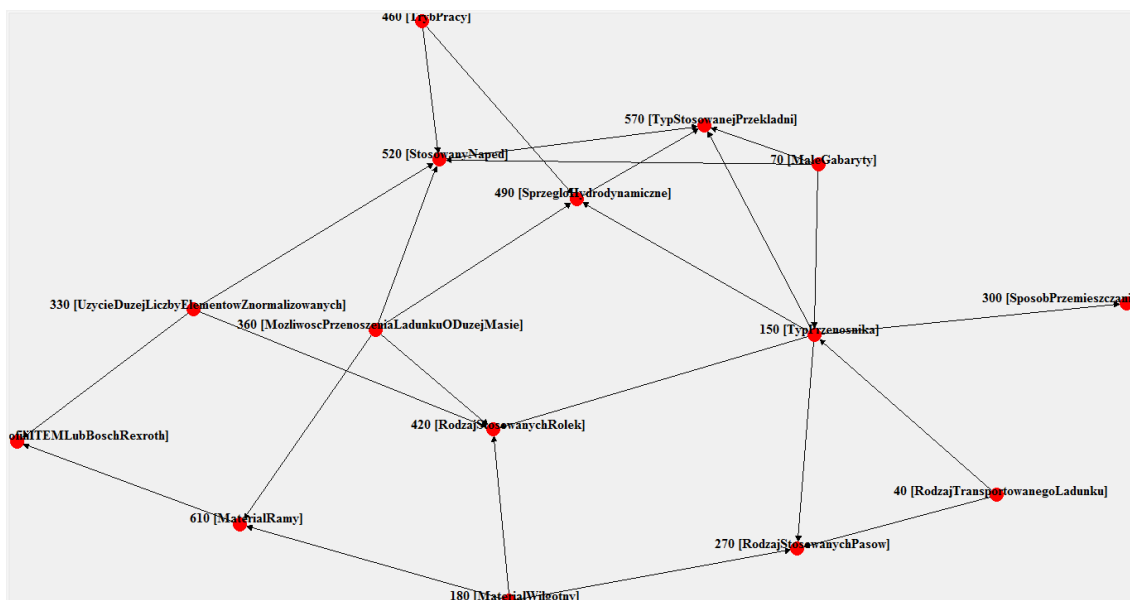
W oparciu o opisany powyżej słownik stwierdzeń opracowano dwie sieci przekonań, za pomocą których można dobierać elementy przenośników w zależności od dziedziny przemysłu, w jakiej mają one być stosowane. Zdefiniowano model składający się z dwóch warstw, gdzie każda warstwa modelu odpowiada jednej sieci przekonań (rys. 10.3). Schemat połączeń sieci dla przypadku przemysłu górniczego pokazany na rys. 10.4 i schemat połączeń sieci dla przypadku przemysłu spożywczego pokazany na rys. 10.5 nie różnią się ilością węzłów oraz sposobem ich połączenia. Sieci te różnicują tablice prawdopodobieństw warunkowych, które były budowane zgodnie z przyjętymi metodami konstruowania przenośników oraz zaleceniami dotyczącymi doboru ich elementów.



Rys. 10.3: MMNET. Warstwy modelu



Rys. 10.4: Schemat sieci przekonań do doboru elementów przenośnika dla przemysłu górniczego



Rys. 10.5: Schemat sieci przekonań do doboru elementów przenośnika dla przemysłu spożywczego

10.3. Przykłady użycia

Przykład 1: Dobór elementów przenośnika w przemyśle górniczym

W pierwszym teście przyjęto następujące wartości w węzłach wejściowych, których wykaz umieszczono w rozdziale 10.2:

Węzeł:	40 [RodzajTransportowanegoLadunku]	
Wartości:	10 [Material sypki luzem]	100
	20 [Duże paczki]	0
	30 [Pojedyncze sztuki]	0

Węzeł:	330 [UzycieDuzejLiczbyElementowZnormalizowanych]	
Wartości:	310 [Tak]	0
	320 [Nie]	100

Węzeł:	70 [MaleGabaryty]	
Wartości:	50 [Tak]	0
	60 [Nie]	100

Węzeł:	460 [TrybPracy]	
Wartości:	430 [Ciągła]	0
	440 [Częste cykle]	100
	450 [Okresowa]	0

Węzeł:	360 [MozliwoscPrzenoszeniaLadunkuODuzejMasie]	
Wartości:	340 [Tak]	100
	350 [Nie]	0

Węzeł:	180 [MaterialWilgotny]	
Wartości:	160 [Tak]	100
	170 [Nie]	0

Węzły wyjściowe (obserwowane) uzyskały następujące wartości:

Węzeł:	150 [TypPrzenośnika]	
Wartości:	80 [Rolkowy]	0
	90 [Taśmowy]	30
	100 [Kubelkowy]	30
	110 [Zabierakowy]	30
	120 [Płytkowy]	0
	130 [Ślimakowy]	0
	140 [Wstrząsowy]	10

Węzeł:	420 [RodzajStosowanychRolek]	
Wartości:	370 [Krążnik stalowy gładki]	0
	380 [Brak rolek]	10
	390 [Krążnik tarczowy gumowy]	29.7
	400 [Krążnik z tworzywa sztucznego]	29.7
	410 [Krążnik pierścieniowy gumowy]	30.6

Węzeł:	270 [RodzajStosowanychPasow]	
Wartości:	190 [Taśmy z elastomerów]	0.6
	200 [Brak pasów]	10
	210 [Taśmy z PTFE]	2.10
	220 [Taśmy modularne]	0
	230 [Taśmy tkaninowo gumowe]	0
	240 [Taśmy siatkowo metalowe]	0
	250 [Taśmy siatkowe z poliestru]	1.05
	260 [Taśmy specjalne]	85.8

Węzeł:	490 [SprzegloHydrodynamiczne]	
Wartości:	470 [Jest]	63
	480 [Brak]	37

Węzeł:	520 [StosowanyNapęd]	
Wartości:	500 [Silnik asynchroniczny]	80
	510 [Serwonapęd]	20

Drugi test polegał na zmianie wartości w węźle [RodzajTransportowanegoLadunku] natomiast wartości w pozostałych węzłach wejściowych nie uległy zmianie:

Węzeł:	40 [RodzajTransportowanegoLadunku]	
Wartości:	10 [Material sypki luzem]	0
	20 [Duże paczki]	100
	30 [Pojedyncze sztuki]	0

Węzeł:	330 [UzycieDuzejLiczbyElementowZnormalizowanych]	
Wartości:	310 [Tak]	0
	320 [Nie]	100

Węzeł:	70 [MaleGabaryty]	
Wartości:	50 [Tak]	0
	60 [Nie]	100

Węzeł:	460 [TrybPracy]	
Wartości:	430 [Ciągła]	0
	440 [Częste cykle]	100
	450 [Okresowa]	0

Węzeł:	360 [MozliwoscPrzenoszeniaLadunkuODuzejMasie]	
Wartości:	340 [Tak]	100
	350 [Nie]	0

Węzeł:	180 [MaterialWilgotny]	
Wartości:	160 [Tak]	100
	170 [Nie]	0

W tym przypadku węzły wyjściowe (obserwowane) uzyskały następujące wartości:

Węzeł:	150 [TypPrzenosnika]	
Wartości:	80 [Rolkowy]	0
	90 [Taśmowy]	50
	100 [Kubekowy]	0
	110 [Zabierakowy]	10
	120 [Płytkowy]	30
	130 [Ślimakowy]	0
	140 [Wstrząsowy]	10

Węzeł:	420 [RodzajStosowanychRolek]	
Wartości:	370 [Krążnik stalowy gładki]	0
	380 [Brak rolek]	40
	390 [Krążnik tarczowy gumowy]	19.8
	400 [Krążnik z tworzywa sztucznego]	19.8
	410 [Krążnik pierścieniowy gumowy]	20.4

Węzeł:	270 [RodzajStosowanychPasow]	
Wartości:	190 [Taśmy z elastomerów]	1.0
	200 [Brak pasów]	40
	210 [Taśmy z PTFE]	2.50
	220 [Taśmy modularne]	0
	230 [Taśmy tkaninowo gumowe]	0
	240 [Taśmy siatkowo metalowe]	0
	250 [Taśmy siatkowe z poliestru]	2.50
	260 [Taśmy specjalne]	54.0

Węzeł:	490 [SprzegloHydrodynamiczne]	
Wartości:	470 [Jest]	66
	480 [Brak]	34

Węzeł:	520 [StosowanyNapęd]	
Wartości:	500 [Silnik asynchroniczny]	80
	510 [Serwonapęd]	20

Kolejny test polegał na ponownej zmianie wartości w węźle [RodzajTransportowanegoLadunku] przy czym podobnie jak poprzednio wartości w pozostałych węzłach wejściowych nie uległy zmianie:

Węzeł:	40 [RodzajTransportowanegoLadunku]	
Wartości:	10 [Material sypki luzem]	0
	20 [Duże paczki]	0
	30 [Pojedyncze sztuki]	100

Węzeł:	330 [UzycieDuzejLiczbyElementowZnormalizowanych]	
Wartości:	310 [Tak]	0
	320 [Nie]	100

Węzeł:	70 [MaleGabaryty]	
Wartości:	50 [Tak]	0
	60 [Nie]	100

Węzeł:	460 [TrybPracy]	
Wartości:	430 [Ciągła]	0
	440 [Częste cykle]	100
	450 [Okresowa]	0

Węzeł:	360 [MozliwoscPrzenoszeniaLadunkuODuzejMasie]	
Wartości:	340 [Tak]	100
	350 [Nie]	0

Węzeł:	180 [MaterialWilgotny]	
Wartości:	160 [Tak]	100
	170 [Nie]	0

Węzły wyjściowe (obserwowane) uzyskały następujące wartości:

Węzeł:	150 [TypPrzenosnika]	
Wartości:	80 [Rolkowy]	0
	90 [Taśmowy]	40
	100 [Kubekowy]	20
	110 [Zabierakowy]	20
	120 [Płytkowy]	0
	130 [Ślimakowy]	0
	140 [Wstrząsowy]	20

Węzeł:	420 [RodzajStosowanychRolek]	
Wartości:	370 [Krążnik stalowy gładki]	0
	380 [Brak rolek]	20
	390 [Krążnik tarczowy gumowy]	26.4
	400 [Krążnik z tworzywa sztucznego]	26.4
	410 [Krążnik pierścieniowy gumowy]	27.2

Węzeł:	270 [RodzajStosowanychPasow]	
Wartości:	190 [Taśmy z elastomerów]	0.80
	200 [Brak pasów]	20
	210 [Taśmy z PTFE]	2.00
	220 [Taśmy modularne]	0
	230 [Taśmy tkaninowo gumowe]	0
	240 [Taśmy siatkowo metalowe]	0
	250 [Taśmy siatkowe z poliestru]	2.00
	260 [Taśmy specjalne]	75.2

Węzeł:	490 [SprzegloHydrodynamiczne]	
Wartości:	470 [Jest]	56
	480 [Brak]	44

Węzeł:	520 [StosowanyNapęd]	
Wartości:	500 [Silnik asynchroniczny]	80
	510 [Serwonapęd]	20

W następnym kroku zmieniono wartość w węźle [TrybPracy] ustawiając tryb pracy ciągłej przy czym wartości w pozostałych węzłach wejściowych nie uległy zmianie:

Węzeł:	40 [RodzajTransportowanegoLadunku]	
Wartości:	10 [Material sypki luzem]	0
	20 [Duże paczki]	0
	30 [Pojedyncze sztuki]	100

Węzeł:	330 [UzycieDuzejLiczbyElementowZnormalizowanych]	
Wartości:	310 [Tak]	0
	320 [Nie]	100

Węzeł:	70 [MaleGabaryty]	
Wartości:	50 [Tak]	0
	60 [Nie]	100

Węzeł:	460 [TrybPracy]	
Wartości:	430 [Ciągła]	100
	440 [Częste cykle]	0
	450 [Okresowa]	0

Węzeł:	360 [MozliwoscPrzenoszeniaLadunkuODuzejMasie]	
Wartości:	340 [Tak]	100
	350 [Nie]	0

Węzeł:	180 [MaterialWilgotny]	
Wartości:	160 [Tak]	100
	170 [Nie]	0

Dla przyjętych w ten sposób wartości, w węzłach wyjściowych (obserwowane) uzyskano następujące wyniki:

Węzeł:	150 [TypPrzenosnika]	
Wartości:	80 [Rolkowy]	0
	90 [Taśmowy]	40
	100 [Kubełkowy]	20
	110 [Zabierakowy]	20
	120 [Płytkowy]	0
	130 [Ślimakowy]	0
	140 [Wstrząsowy]	20

Węzeł:	420 [RodzajStosowanychRolek]	
Wartości:	370 [Krążnik stalowy gładki]	0
	380 [Brak rolek]	20
	390 [Krążnik tarczowy gumowy]	26.4
	400 [Krążnik z tworzywa sztucznego]	26.4
	410 [Krążnik pierścieniowy gumowy]	27.2

Węzeł:	270 [RodzajStosowanychPasow]	
Wartości:	190 [Taśmy z elastomerów]	0.80
	200 [Brak pasów]	20
	210 [Taśmy z PTFE]	2.00
	220 [Taśmy modularne]	0
	230 [Taśmy tkaninowo gumowe]	0
	240 [Taśmy siatkowo metalowe]	0
	250 [Taśmy siatkowe z poliestru]	2.00
	260 [Taśmy specjalne]	75.2

Węzeł:	490 [SprzęgłoHydrodynamiczne]	
Wartości:	470 [Jest]	24
	480 [Brak]	76

Węzeł:	520 [StosowanyNapęd]	
Wartości:	500 [Silnik asynchroniczny]	100
	510 [Serwonapęd]	0

Rodzaj transportowanego ładunku ma silny wpływ na typ stosowanego przenośnika a także na rodzaj użytych rolek i pasów. W zależności od wskazanego ładunku sieć prawidłowo sugeruje dobór odpowiednich elementów przenośnika. Próba zmiany trybu pracy pokazuje wyraźnie, że ten parametr ma wpływ jedynie na konieczność stosowania sprzęgła hydrodynamicznego oraz rodzaju stosowanego silnika. Otrzymane wyniki wskazują na prawidłową konstrukcję i działanie sieci.

Przykład 2: Dobór elementów przenośnika w przemyśle spożywczym

W przykładzie drugim zastosowano dokładnie takie same podejście jak w przypadku przenośnika dla przemysłu górniczego. Dokonano takich samych modyfikacji w węzłach wejściowych i obserwowano kolejno zmiany zachodzące w wybranych węzłach wyjściowych:

Węzeł:	40 [RodzajTransportowanegoLadunku]	
Wartości:	10 [Material sypki luzem]	100
	20 [Duże paczki]	0
	30 [Pojedyncze sztuki]	0

Węzeł:	330 [UzycieDuzejLiczbyElementowZnormalizowanych]	
Wartości:	310 [Tak]	0
	320 [Nie]	100

Węzeł:	70 [MaleGabaryty]	
Wartości:	50 [Tak]	0
	60 [Nie]	100

Węzeł:	460 [TrybPracy]	
Wartości:	430 [Ciągła]	0
	440 [Częste cykle]	100
	450 [Okresowa]	0

Węzeł:	360 [MozliwoscPrzenoszeniaLadunkuODuzejMasie]	
Wartości:	340 [Tak]	100
	350 [Nie]	0

Węzeł:	180 [MaterialWilgotny]	
Wartości:	160 [Tak]	100
	170 [Nie]	0

Węzły wyjściowe (obserwowane) uzyskały następujące wartości:

Węzeł:	150 [TypPrzenosnika]	
Wartości:	80 [Rolkowy]	0
	90 [Taśmowy]	0
	100 [Kubekowy]	40
	110 [Zabierakowy]	40
	120 [Płytkowy]	0
	130 [Ślimakowy]	0
	140 [Wstrząsowy]	20

Węzeł:	420 [RodzajStosowanychRolek]	
Wartości:	370 [Krążnik stalowy gładki]	0
	380 [Brak rolek]	20
	390 [Krążnik tarczowy gumowy]	26.4
	400 [Krążnik z tworzywa sztucznego]	26.4
	410 [Krążnik pierścieniowy gumowy]	27.2

Węzeł:	270 [RodzajStosowanychPasow]	
Wartości:	190 [Taśmy z elastomerów]	40
	200 [Brak pasów]	20
	210 [Taśmy z PTFE]	40
	220 [Taśmy modularne]	0
	230 [Taśmy tkaninowo gumowe]	0
	240 [Taśmy siatkowo metalowe]	0
	250 [Taśmy siatkowe z poliestru]	0
	260 [Taśmy specjalne]	0

Węzeł:	490 [SprzęgłoHydrodynamiczne]	
Wartości:	470 [Jest]	56
	480 [Brak]	44

Węzeł:	520 [StosowanyNapęd]	
Wartości:	500 [Silnik asynchroniczny]	40
	510 [Serwonapęd]	60

Drugi test polegał na zmianie wartości w węźle [RodzajTransportowanegoLadunku] przy czym wartości w pozostałych węzłach wejściowych nie uległy zmianie:

Węzeł:	40 [RodzajTransportowanegoLadunku]	
Wartości:	10 [Materiał sypki luzem]	0
	20 [Duże paczki]	100
	30 [Pojedyncze sztuki]	0

Węzeł:	330 [UzycieDuzejLiczbyElementowZnormalizowanych]	
Wartości:	310 [Tak]	0
	320 [Nie]	100

Węzeł:	70 [MaleGabaryty]	
Wartości:	50 [Tak]	0
	60 [Nie]	100

Węzeł:	460 [TrybPracy]	
Wartości:	430 [Ciągła]	0
	440 [Częste cykle]	100
	450 [Okresowa]	0

Węzeł:	360 [MozliwoscPrzenoszeniaLadunkuODuzejMasie]	
Wartości:	340 [Tak]	100
	350 [Nie]	0

Węzeł:	180 [MaterialWilgotny]	
Wartości:	160 [Tak]	100
	170 [Nie]	0

Węzły wyjściowe (obserwowane) uzyskały następujące wartości:

Węzeł:	150 [TypPrzenosnika]	
Wartości:	80 [Rolkowy]	30
	90 [Taśmowy]	30
	100 [Kubekowy]	0
	110 [Zabierakowy]	5
	120 [Płytkowy]	30
	130 [Ślimakowy]	0
	140 [Wstrząsowy]	5

Węzeł:	420 [RodzajStosowanychRolek]	
Wartości:	370 [Krążnik stalowy gładki]	0
	380 [Brak rolek]	35
	390 [Krążnik tarczowy gumowy]	20.6
	400 [Krążnik z tworzywa sztucznego]	20.6
	410 [Krążnik pierścieniowy gumowy]	23.9

Węzeł:	270 [RodzajStosowanychPasow]	
Wartości:	190 [Taśmy z elastomerów]	11.5
	200 [Brak pasów]	65
	210 [Taśmy z PTFE]	11.50
	220 [Taśmy modularne]	3
	230 [Taśmy tkaninowo gumowe]	0
	240 [Taśmy siatkowo metalowe]	0
	250 [Taśmy siatkowe z poliestru]	9
	260 [Taśmy specjalne]	0

Węzeł:	490 [SprzęgłoHydrodynamiczne]	
Wartości:	470 [Jest]	69.5
	480 [Brak]	30.5

Węzeł:	520 [StosowanyNapęd]	
Wartości:	500 [Silnik asynchroniczny]	40
	510 [Serwonapęd]	60

Kolejny test polegał na ponownej zmianie wartości w węźle [RodzajTransportowanegoLadunku] przy czym wartości w pozostałych węzłach wejściowych nie uległy zmianie:

Węzeł:	40 [RodzajTransportowanegoLadunku]	
Wartości:	10 [Materiał sypki luzem]	0
	20 [Duże paczki]	0
	30 [Pojedyncze sztuki]	100

Węzeł:	330 [UzycieDuzejLiczbyElementowZnormalizowanych]	
Wartości:	310 [Tak]	0
	320 [Nie]	100

Węzeł:	70 [MaleGabaryty]	
Wartości:	50 [Tak]	0
	60 [Nie]	100

Węzeł:	460 [TrybPracy]	
Wartości:	430 [Ciągła]	0
	440 [Częste cykle]	100
	450 [Okresowa]	0

Węzeł:	360 [MozliwoscPrzenoszeniaLadunkuODuzejMasie]	
Wartości:	340 [Tak]	100
	350 [Nie]	0

Węzeł:	180 [MaterialWilgotny]	
Wartości:	160 [Tak]	100
	170 [Nie]	0

Węzły wyjściowe (obserwowane) uzyskały następujące wartości:

Węzeł:	150 [TypPrzenosnika]	
Wartości:	80 [Rolkowy]	20
	90 [Taśmowy]	20
	100 [Kubekowy]	0
	110 [Zabierakowy]	20
	120 [Płytkowy]	20
	130 [Ślimakowy]	0
	140 [Wstrząsowy]	20

Węzeł:	420 [RodzajStosowanychRolek]	
Wartości:	370 [Krążnik stalowy gładki]	0
	380 [Brak rolek]	40
	390 [Krążnik tarczowy gumowy]	19.2
	400 [Krążnik z tworzywa sztucznego]	19.2
	410 [Krążnik pierścieniowy gumowy]	21.6

Węzeł:	270 [RodzajStosowanychPasow]	
Wartości:	190 [Taśmy z elastomerów]	16
	200 [Brak pasów]	60
	210 [Taśmy z PTFE]	16.00
	220 [Taśmy modularne]	2
	230 [Taśmy tkaninowo gumowe]	0
	240 [Taśmy siatkowo metalowe]	0
	250 [Taśmy siatkowe z poliestru]	2.00
	260 [Taśmy specjalne]	0

Węzeł:	490 [SprzegloHydrodynamiczne]	
Wartości:	470 [Jest]	58
	480 [Brak]	42

Węzeł:	520 [StosowanyNapęd]	
Wartości:	500 [Silnik asynchroniczny]	80
	510 [Serwonapęd]	20

W następnym kroku zmieniono wartość w węźle [TrybPracy] ustawiając tryb pracy ciągłej przy czym wartości w pozostałych węzłach wejściowych nie uległy zmianie:

Węzeł:	40 [RodzajTransportowanegoLadunku]	
Wartości:	10 [Material sypki luzem]	0
	20 [Duże paczki]	0
	30 [Pojedyncze sztuki]	100

Węzeł:	330 [UzycieDuzejLiczbyElementowZnormalizowanych]	
Wartości:	310 [Tak]	0
	320 [Nie]	100

Węzeł:	70 [MaleGabaryty]	
Wartości:	50 [Tak]	0
	60 [Nie]	100

Węzeł:	460 [TrybPracy]	
Wartości:	430 [Ciągła]	100
	440 [Częste cykle]	0
	450 [Okresowa]	0

Węzeł:	360 [MozliwoscPrzenoszeniaLadunkuODuzejMasie]	
Wartości:	340 [Tak]	100
	350 [Nie]	0

Węzeł:	180 [MaterialWilgotny]	
Wartości:	160 [Tak]	100
	170 [Nie]	0

Węzły wyjściowe (obserwowane) uzyskały następujące wartości:

Węzeł:	150 [TypPrzenosnika]	
Wartości:	80 [Rolkowy]	0
	90 [Taśmowy]	40
	100 [Kubełkowy]	20
	110 [Zabierakowy]	20
	120 [Płytkowy]	0
	130 [Ślimakowy]	0
	140 [Wstrząsowy]	20

Węzeł:	420 [RodzajStosowanychRolek]	
Wartości:	370 [Krążnik stalowy gładki]	0
	380 [Brak rolek]	20
	390 [Krążnik tarczowy gumowy]	26.4
	400 [Krążnik z tworzywa sztucznego]	26.4
	410 [Krążnik pierścieniowy gumowy]	27.2

Węzeł:	270 [RodzajStosowanychPasow]	
Wartości:	190 [Taśmy z elastomerów]	0.80
	200 [Brak pasów]	20
	210 [Taśmy z PTFE]	2.00
	220 [Taśmy modularne]	0
	230 [Taśmy tkaninowo gumowe]	0
	240 [Taśmy siatkowo metalowe]	0
	250 [Taśmy siatkowe z poliestru]	2.00
	260 [Taśmy specjalne]	75.2

Węzeł:	490 [SprzegloHydrodynamiczne]	
Wartości:	470 [Jest]	22
	480 [Brak]	78

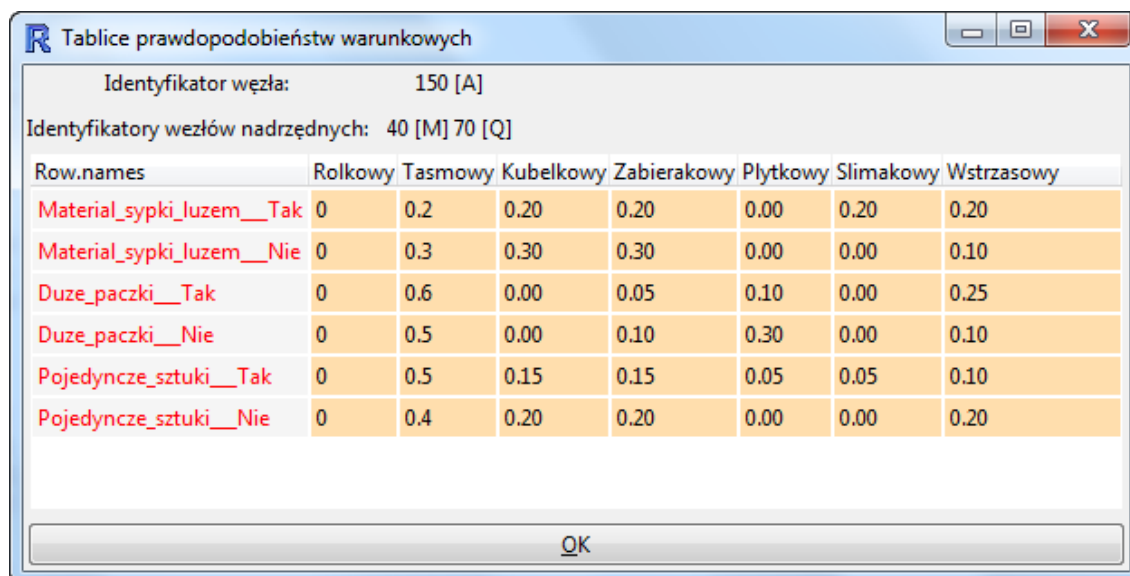
Węzeł:	520 [StosowanyNaped]	
Wartości:	500 [Silnik asynchroniczny]	80
	510 [Serwonaped]	20

Podobnie jak w przypadku doboru elementów przenośnika dla przemysłu górniczego tak i w przypadku przemysłu spożywczego rodzaj transportowanego ładunku wpływa na typ przenośnika, i odgrywa istotną rolę przy doborze rodzaju materiału stosowanego na pasy oraz jest bezpośrednio związany z rodzajem użytych rolek. W zależności od wskazanego ładunku sieć prawidłowo sugeruje dobór odpowiednich elementów przenośnika. Zmiana trybu pracy pokazuje wyraźnie, że ten parametr ma wpływ jedynie na konieczność stosowania sprzęgła hydrodynamicznego oraz rodzaju stosowanego silnika podczas gdy pozostałe wartości ustalone w sieci nie ulegają zmianie. Otrzymane wyniki wskazują na prawidłową konstrukcję i działanie sieci.

10.4. Podsumowanie i wnioski

W obydwu przedstawionych przypadkach struktura logiczna sieci pozostała dokładnie taka sama. Połączenia pomiędzy poszczególnymi węzłami nie uległy zmianie. Prawidłowe działanie sieci w obydwu przypadkach zapewniły inne postacie tablic prawdopodobieństw warunkowych w każdym z węzłów sieci. Wartości prawdopodobieństw w tych tablicach zostały dobrane na podstawie przeprowadzonych badań literaturowych oraz praktyki własnej. Przedstawione przykłady pokazały, że zmiany takie pozwalają na przeprowadzenie prawidłowej klasyfikacji w obydwu opisanych przypadkach.

Na rys. 10.6 przedstawiono tablicę prawdopodobieństw warunkowych, która umożliwiła podjęcie decyzji o typie przenośnika w zależności od rodzaju transportowanego ładunku. Można zauważyć, że w przypadku transportowania materiału sypkiego luzem wartości prawdopodobieństw warunkowych wskazują na możliwość zastosowania przenośników taśmowego, kubełkowego, zabierakowego i ślimakowego. W przypadku materiałów o dużych gabarytach zdecydowanie zalecany jest przenośnik taśmowy dla którego prawdopodobieństwo warunkowe w tym przypadku określono na 60%.



Identyfikator węzła: 150 [A]
 Identyfikatory węzłów nadrzędnych: 40 [M] 70 [Q]

Row.names	Rolkowy	Tasmowy	Kubelkowy	Zabierakowy	Plytkowy	Slimakowy	Wstrzasowy
Material_sypki_luzem__Tak	0	0.2	0.20	0.20	0.00	0.20	0.20
Material_sypki_luzem__Nie	0	0.3	0.30	0.30	0.00	0.00	0.10
Duze_paczki__Tak	0	0.6	0.00	0.05	0.10	0.00	0.25
Duze_paczki__Nie	0	0.5	0.00	0.10	0.30	0.00	0.10
Pojedyncze_sztuki__Tak	0	0.5	0.15	0.15	0.05	0.05	0.10
Pojedyncze_sztuki__Nie	0	0.4	0.20	0.20	0.00	0.00	0.20

OK

Rys. 10.6: Przykład tablicy prawdopodobieństw warunkowych

Bibliografia

- [10.1] Enitra. <http://www.enitra.pl>, grudzień 2010.
- [10.2] F.W. Przystupa. *Diagnozer w systemie technicznym*. Oficyna Wydawnicza Politechniki Wrocławskiej, 2010.
- [10.3] Honczarenko J. *Elastyczna automatyzacja wytwarzania*. Wydawnictwa Naukowo-Techniczne, 2000.
- [10.4] STROMAG. SPRZĘGŁA HYDROKINETYCZNE - systemy i komponenty dla techniki napędowej. <http://www.stromag.pl>, grudzień 2010.

Rozdział 11

Przykłady sieci stwierdzeń przeznaczonych do wspomagania procesu podejmowania decyzji dotyczących oceniania postępów studentów

Krzysztof PSIUK

W rozdziale tym został przedstawiony przykład sieci stwierdzeń, pokazujący możliwość oceniania postępów w przyswajaniu wiedzy przez studentów w procesie nauczania. Przykład dotyczy sprawdzania stanu wiedzy z takich zagadnień jak prowadzenie procesu koncyptowania, znajomość metod wspomagających koncyptowanie oraz umiejętność zastosowania poznanych metod w praktycznym ich wykorzystaniu. Opisano założenia do opracowanych przykładów sieci stwierdzeń, przedstawiono strukturę opracowanych sieci, wyniki ich działania i wnioski.

11.1. Ocena postępów studenta

Jednym z elementów procesu nauczania jest ocena postępów studenta. Ocena taka powinna ułatwić zarówno prowadzącemu zajęcia jak również samemu studentowi zebranie informacji pozwalających na określenie jakie zagadnienia zostały już opanowane, a z jakimi należy się jeszcze zapoznać lub poprawić ich znajomość. W rozpatrywanym przypadku nie uwzględnia się sposobu weryfikacji oceny postępów studentów, a jedynie sam fakt określenia znajomości lub nieznanności określonych zagadnień. W ramach wybranego przedmiotu student ma zazwyczaj za zadanie zapoznać się z pewnymi zagadnieniami teoretycznymi oraz umiejętnością zastosowania poznanych zagadnień w praktyce. Dlatego też wydaje się, że stosowane zazwyczaj podejście sprawdzające stan wiedzy z jednej i drugiej części omawianych zagadnień jest poprawne. Oceniając postępy pracy studenta w poznawaniu poszczególnych zagadnień omawianych w ramach danego przedmiotu, należy uwzględnić między innymi wymagania związane z procesem kształcenia dla wybranego kierunku studiów. W szczególnym przypadku, wymagania dla tego samego przedmiotu mogą być różne dla różnych kierunków studiów. Można również przedstawić przykłady zagadnień, które mogą być omawiane nie tylko na różnych kierunkach studiów ale również w ramach różnych przedmiotów. Dlatego omawiane zagadnienie, oceny postępów pracy studenta, może być rozpatrywane w zależności od kierunku studiów oraz przedmiotu, w ramach którego jest ono rozpatrywane.

11.2. Założenia do opracowanego przykładu

Przygotowując założenia do przykładów sieci stwierdzeń, dotyczących oceny postępów pracy studenta należało wybrać zagadnienia, dla których ten proces będzie opracowany. Ze względu na różnorodność zajęć z jakimi spotykają się studenci na naszym wydziale (Wydział Mechaniczny-Technologiczny Politechniki Śląskiej), zdecydowano się na wybór jednego przedmiotu, w ramach którego będzie prowadzona ocena. Jako przykładowy przedmiot wybrano Podstawy Konstrukcji Maszyn (PKM). Przedmiot ten jest prowadzony na wielu różnych kierunkach i w różnym zakresie, zarówno tematycznym jak i wymiaru liczby godzin zajęć. Ponieważ przedmiot ten jest stosunkowo obszerny dla potrzeb niniejszego opracowania ograniczono się do zagadnień związanych z koncyplinowaniem, metodami wspomagającymi ten proces oraz umiejętnością stosowania tego typu metod w rozwiązywaniu rzeczywistych problemów.

W ramach zadania utworzenia sieci oceniającej postępy pracy studentów wydzielono dwie grupy podzadań. Pierwsza grupa podzadań dotyczyła określenia grup studentów, dla których będzie prowadzona ocena, druga grupa zadań dotyczyła określenia ocenianej tematyki. W ramach pierwszego zadania zdecydowano się na przygotowanie sieci dla czterech grup studentów, obejmujących następujące zagadnienia:

1. Pierwsza grupa studentów to studenci kierunków mechanicznych, będący na początku zajęć z PKM. Studenci Ci posiadają ogólną wiedzę z zakresu materiałoznawstwa, mechaniki czy wytrzymałości materiałów i rozpoczynają kurs z podstaw konstrukcji maszyn.
2. Druga grupa to studenci kierunków mechanicznych, którzy kończą zajęcia z podstaw konstrukcji maszyn i posiadli wiedzę przekazywaną im w zakresie określonym wymogami dla tego przedmiotu. Ich wiedza i doświadczenie wynikłe z wykonanych zadań projektowych pozwoli im na inne spojrzenie na zadania dotyczące koncyplinowania.
3. Trzecią grupę stanowią studenci kierunków związanych z zarządzaniem i organizacją produkcji, dla których prowadzone są zajęcia z PKM. Wymagania dotyczące zakresu omawianych problemów na tych kierunkach, są nieco inne od tych prezentowanych na kierunkach mechanicznych. Stąd ich wiedza i doświadczenie są inne niż poprzednio rozpatrywanych studentów.
4. Ostatnia grupa dotyczy również studentów kierunków związanych z zarządzaniem, którzy w swoim zakresie poznali zagadnienia związane z wymaganiami urbanistycznymi i architektonicznymi. Dzięki tej wiedzy będą mogli uwzględnić w swoich rozwiązaniach uwarunkowania terenu i otoczenia.

Dla tych wybranych grup docelowych studentów zostały opracowane wymagania dotyczące zagadnień, z którymi powinni się zapoznać. Całość postanowiono podzielić na dwie części. Pierwsza część dotyczy zagadnień teoretycznych, z którymi każda z grup studentów w określonym zakresie powinna się zapoznać. Druga część dotyczy rozwiązania zadania projektowego, w zakresie posiadanej wiedzy. Założono, że nie jest dany szczegółowy temat rozpatrywanego zadania projektowego, a jedynie zakres zadań, które powinno ono obejmować.

W ramach części teoretycznej studenci wszystkich wymienionych grup powinni zapoznać się z zagadnieniami dotyczącymi ogólnych zagadnień związanych z procesem projektowo konstrukcyjnym i jego etapami [11.2]. Założono również, że znane im powinny być metody wspomagania koncyplinowania oraz metody oceny opracowanych koncepcji i wyboru rozwiązania optymalnego. Założono, że studenci będą musieli poznać metody wspomagające proces koncyplinowania takie jak [11.8]:

- zagadnienia związane z formułowaniem zadania projektowego,
- określanie wymagań projektowych,
- metody wspomagające określanie wymagań projektowych,

- metody ściśle heurystyczne (np. metoda pytań, burza mózgów, zasady Altszullera),
- operatory metod heurystycznych (np. analogie, superpozycja, gra słowami),
- metody systematyczno-heurystyczne (np. metoda morfologiczna, metoda drzewa rozwiązań).

Dla tak określonych zagadnień opracowano przykładową wielowarstwową sieć stwierdzeń, umożliwiającą określenie, na jakim etapie wiedzy teoretycznej i praktycznej znajdują się aktualnie studenci. Pozwolą one na przekazanie informacji zarówno studentowi jak i prowadzącemu, które z wymaganych zagadnień wymagają jeszcze poprawy.

11.3. Struktura opracowanych sieci

W zadaniu zostały opracowane cztery typy sieci prezentujące różne aspekty oceny dla tej samej grupy zagadnień. Tak jak opisano to wcześniej, podstawą do opracowania sieci była ocena postępów pracy studenta w czasie jego poznawania zagadnień związanych z metodami wspomagającymi koncyrowanie. Przy opracowywaniu sieci przyjęto założenie, że stwierdzenia, z których będzie się składać sieć zostaną podzielone na dwie główne grupy. Pierwsza grupa będzie obejmowała zagadnienia związane ze znajomością metod wspomagających proces koncyrowania, ich sposobem działania i przeznaczeniem. Druga grupa będzie obejmowała stwierdzenia, na podstawie których można podjąć próbę oceny postępów pracy studenta związanych z umiejętnością praktycznego zastosowania poznanych metod. Powstał w ten sposób zbiór stwierdzeń, które można połączyć ze sobą na kilka sposobów uwzględniających różne aspekty oceny studentów. Stwierdzenia zostały podzielone na dwie główne grupy. W pierwszej grupie umieszczono stwierdzenia dotyczące zagadnień teoretycznych, w drugiej grupie znajdują się stwierdzenia dotyczące oceny postępów pracy studenta w ramach zajęć praktycznych. Ze względu na skomplikowaną strukturę omawianych zagadnień, konieczne było opracowanie sieci zawierającej stwierdzenia złożone, których wartości wyznaczone są na podstawie wartości stwierdzeń prostych. Najpierw opracowano więc zestaw stwierdzeń prostych, a następnie korzystając z przygotowanych stwierdzeń prostych przygotowano stwierdzenia złożone. Przygotowane stwierdzenia złożone zostały wykorzystane do opracowania sieci stwierdzeń. Poniżej przedstawiono zestawy poszczególnych stwierdzeń z podziałem na grupy. Poszczególne węzły opatrzone krótkim komentarzem.

11.3.1. Stwierdzenia z grupy zagadnień teoretycznych

Operatory metod heurystycznych

Stwierdzenia określające znajomość operatorów metod heurystycznych oraz zbiorcze stwierdzenie oceniające.

Wydruk 11.1: Algorytm rozwiązywania sieci przybliżonej

```

Nazwa stwierdzenia – Analogie
{
    Stwierdzenia proste = (Nie Zna Metod ,
                          Zna Typy Analogii ,
                          Zna i Rozumie Metody ,
                          Zna Metody i Rozwiązuje Zadania);
    Treść stwierdzenia = "Znajomość Analogii";
};

```

1
2
3
4
5
6
7
8
9
10

```

Nazwa stwierdzenia – Superpozycja                               11
{                                                                 12
    Wartości = (Nie Zna Metody,                                  13
                Zna Ogólną Ideę Metody,                        14
                Zna i Rozumie Metodę,                          15
                Zna Metodę i Rozwiązuje Zadanie);              16
    Tytuł = "Znajomość Superpozycji";                           17
};                                                                 18
                                                                 19
Nazwa stwierdzenia – MetodaRozwiazaniaIdealnego                20
{                                                                 21
    Stwierdzenia proste = (Nie Zna Metody,                      22
                            Zna Ogólną Ideę Metody,            23
                            Zna i Rozumie Metodę,              24
                            Zna Metodę i Rozwiązuje Zadanie);  25
    Treść stwierdzenia = "Znajomość Metody rozwiązania        26
                        idealnego";
};                                                                 27
                                                                 28
Nazwa stwierdzenia – KPytaniaPK4                               29
{                                                                 30
    Stwierdzenia proste = (Nie Zna Metod,                       31
                            Znajomość Nazw Metod,              32
                            Posiada Ogólną Znajomość Istoty Metod, 33
                            Posiada Szczegółowa Znajomość Metod); 34
    Treść stwierdzenia = "Znajomość Operatorów Metod          35
                        Heurystycznych";
};                                                                 36

```

Metody ściśle heurystyczne

Stwierdzenia określające znajomość metod ściśle heurystycznych oraz zbiorcze stwierdzenie oceniające.

Wydruk 11.2: Algorytm rozwiązywania sieci przybliżonej

```

                                                                 1
Nazwa stwierdzenia – BurzaMozgow                               2
{                                                                 3
    Stwierdzenia proste = (Nie Zna Metody,                      4
                            Zna Ogólny Opis Metody,            5
                            Zna Szczegóły Metody,               6
                            Zna Szczegóły I Metodę 635,         7
                            Zna Metodę i Rozwiązuje Zadanie);  8
    Treść stwierdzenia = "Znajomość Metody Burza Mózgów";      9
};                                                                 10
                                                                 11
Węzeł – MetodaPytan                                           12
{                                                                 13
    Stwierdzenia proste = (Nie Zna Metody,                      14
                            Zna Ogólny Opis Metody,            15
                            Zna Metodę Zestaw Pytań,           16
                            Zna Metodę i Rozwiązuje Zadanie);  17

```

```

        Treść stwierdzenia = "Znajomość Metody Pytań";           18
    };                                                             19
                                                                    20
Nazwa stwierdzenia – WMetodyScisleHeurystyczne                 21
{                                                                 22
    Stwierdzenia proste = (Nie Zna,                             23
        Orientuje Sie,                                         24
        Zna Bardzo Dobrze Metodę);                             25
    Treść stwierdzenia = "Znajomość Metod Ściśle               26
        Heurystycznych";
};                                                                 27

```

Metody systematyczno-heurystyczne

Stwierdzenia określające znajomość metod systematyczno-heurystycznych oraz zbiorcze stwierdzenie oceniające.

Wydruk 11.3: Algorytm rozwiązywania sieci przybliżonej

```

                                                                    1
Nazwa stwierdzenia – MetodaMorfologiczna                       2
{                                                                 3
    Stwierdzenia proste = (Nie Zna Metody,                     4
        Zna Ogólny Opis Metody,                               5
        Zna Procedurę Metody,                                 6
        Zna Dobrze Metodę,                                    7
        Zna Metodę i Rozwiązuje Zadanie);                     8
    Treść stwierdzenia = "Znajomość Metod Morfologicznych";   9
};                                                             10
                                                                    11
Nazwa stwierdzenia – MetodaDrzewaRozwiazan                    12
{                                                                 13
    Stwierdzenia proste = (Nie Zna Metody,                     14
        Zna Ogólny Opis Metody,                               15
        Zna Szczegóły Metody,                                 16
        Zna Metodę i Rozwiązuje Zadanie);                     17
    Treść stwierdzenia = "Znajomość Metody Drzewa             18
        Rozwiązań";
};                                                             19
                                                                    20
Nazwa stwierdzenia – MetodaSystemowa                           21
{                                                                 22
    Stwierdzenia proste = (Nie Zna Metody,                     23
        Zna Rozumie Pojęcie Systemu,                          24
        Zna Procedure Postępowania,                            25
        Zna Metodę i Rozwiązuje Zadanie);                     26
    Treść stwierdzenia = "Znajomość Metody systemowej";      27
};                                                             28
                                                                    29
Nazwa stwierdzenia – WMetodySystHeurystyczne                 30
{                                                                 31
    Stwierdzenia proste = (Nie Zna,                             32
        Orientuje Sie,                                         33

```

```

        Zna Bardzo Dobrze);
    Treść stwierdzenia = "Znajomość Metody systematyczno
        heurystycznej";
};

```

Ocena części teoretycznej

Stwierdzenie zbiorcze dotyczące ogólnej oceny znajomości zagadnień teoretycznych.

Wydruk 11.4: Algorytm rozwiązywania sieci przybliżonej

```

Nazwa stwierdzenia – WWspomaganieKoncypowania
{
    Stwierdzenia proste = (Brak znajomości metody,
        Słaba znajomość metody,
        Dobra znajomość metody,
        Bardzo Dobra znajomość metody);
    Treść stwierdzenia = "Znajomość metod wspomagających
        koncypowanie";
};

```

Ocena znajomości zagadnień koncypowania

Stwierdzenie zbiorcze określające znajomość zagadnień dotyczących koncypowania. Wartości stwierdzenia opisane są za pomocą wartości stwierdzeń prostych i określają ogólną znajomość zagadnień uwzględnianych w procesie oceny zarówno części teoretycznej jak również praktycznej. Poszczególne stwierdzenia proste przedstawiają się następująco:

- *OcenaBDB* - określa bardzo dobrą znajomość zarówno zagadnień teoretycznych jak również praktycznych dotyczących koncypowania i wspomaganie koncypowania,
- *OcenaDB* - określa ogólną znajomość metod koncypowania lub dobrą umiejętność zastosowania poznanych metod do rozwiązywania zadań,
- *OcenaDST* - określa słabą znajomość metod wspomaganie koncypowania,
- *OcenaNDST* - określa bardzo słabą znajomość metod koncypowania zarówno części teoretycznej jak i praktycznego wykorzystania tych metod.

Wydruk 11.5: Algorytm rozwiązywania sieci przybliżonej

```

Nazwa stwierdzenia – WspomaganieKoncypowania
{
    Stwierdzenia proste = (OcenaBDB,
        OcenaDB,
        OcenaDST,
        OcenaNDST);
    Treść stwierdzenia = "Znajomość zagadnień wspomaganie
        koncypowania";
};

```

11.3.2. Stwierdzenia z grupy zagadnień praktycznych

Wymagania projektowe

Stwierdzenia określające uwzględnienie w przedstawionym rozwiązaniu zadania różnych

wymagań. Oprócz poszczególnych wymagań ustalono stwierdzenie zbiorcze określające uwzględnienie poszczególnych typów wymagań projektowych. Przedstawione stwierdzenia złożone dotyczą:

- wymagań ekonomicznych,
- wymagań ergonomicznych i estetycznych,
- wymagań niezawodnościowych,
- wymagań dynamicznych,
- wpływu układu urbanistycznego,
- wpływu ogólnej struktury.

Wydruk 11.6: Algorytm rozwiązywania sieci przybliżonej

```

1
Nazwa stwierdzenia – UkładUrbanistyczny 2
{ 3
    Wartości: (Nie Opracowano Układu, 4
                Opracowano Założenia, 5
                Opracowano W Pełni Układ Urbanistyczny); 6
    Treść stwierdzenia = "Opracowano Układ urbanistyczny"; 7
}; 8
9
Nazwa stwierdzenia – UkładZarządzania 10
{ 11
    Stwierdzenia proste = (Nie Uwzględniono W Zadaniu, 12
                Uwzględniono Ogólne Rozwiązania, 13
                Opracowano Rozwiązania Szczegółowe); 14
    Treść stwierdzenia = "Opracowano Strukturę ogólną"; 15
}; 16
17
Nazwa stwierdzenia – WymaganiaDynamiczne 18
{ 19
    Stwierdzenia proste = (Nie Uwzględniono Wymagań, 20
                Uwzględniono Prędkość Obiektu, 21
                Uwzględniono Inne Właściwości Dynamiczne); 22
    Treść stwierdzenia = "Uwzględniono Wymagania 23
                dynamiczne";
}; 24
25
Nazwa stwierdzenia – WymaganiaNiezawodnościowe 26
{ 27
    Stwierdzenia proste = (Nie Uwzględniono Wymagań, 28
                Uwzględniono Wpływ Czynników Zewnętrznych, 29
                Uwzględniono Oddziaływania Na Otoczenie, 30
                Uwzględniono Wpływ Czynników Zewnętrznych i 31
                Wpływ Na Otoczenie);
    Treść stwierdzenia = "Uwzględniono Wymagania 32
                Niezawodnościowe";
}; 33
34

```

Nazwa stwierdzenia – WymaganiaErgonomiczne	35
{	36
Stwierdzenia proste = (Nie Uwzględniono Wymagań,	37
Uwzględniono Wymagania Ergonomiczne ,	38
Uwzględniono Wymagania Estetyczne ,	39
Uwzględniono Wymagania Ergonomiczne i	40
Estetyczne);	
Treść stwierdzenia = "Uwzględniono Wymagania	41
ergonomiczne i estetyczne";	
};	42
	43
Nazwa stwierdzenia – WymaganiaEkonomiczne	44
{	45
Stwierdzenia proste = (Nie Uwzględniono ,	46
Oszacowano Główne Wskaźniki ,	47
Oszacowano Zużycie Energii , Sprawność i Inne	48
Nakłady);	
Treść stwierdzenia = "Uwzględniono Wymagania	49
ekonomiczne";	
};	50
	51
Nazwa stwierdzenia – ZP1Wymagania	52
{	53
Stwierdzenia proste = (Niepoprawne ,	54
Poprawne ,	55
Bardzo Dobre);	56
Treść stwierdzenia = "Uwzględniono Wymagania	57
Projektowe";	
};	58

Ocena wyboru rozwiązania optymalnego

Stwierdzenie uwzględniające przeprowadzoną ocenę opracowanych rozwiązań zadania projektowego.

Wydruk 11.7: Algorytm rozwiązywania sieci przybliżonej

	1
Nazwa stwierdzenia – ZZadanie4	2
{	3
Stwierdzenia proste = (Niepoprawna ,	4
Błędy W Tablicy Ocen ,	5
Poprawna);	6
Treść stwierdzenia = "Ocena Przeprowadzonej	7
Optymalizacji";	
};	8

Ocena opracowanych koncepcji

Zbiór stwierdzeń uwzględniających poprawność przedstawionych rozwiązań konstrukcyjnych. Uwzględniają one takie czynniki jak poprawność przygotowania koncepcji i wybór metod koncygowania.

Wydruk 11.8: Algorytm rozwiązywania sieci przybliżonej

Nazwa stwierdzenia – ZP1Koncypowanie2 {	2
Stwierdzenia proste = (Zbyt Ogólne ,	3
Zbyt Szczegółowe ,	4
Nie Związane Z Tematem ,	5
Poprawne);	6
Treść stwierdzenia = "Ocena opracowanych koncepcji";	7
};	8
	9
Nazwa stwierdzenia – ZP1Koncypowanie1	10
{	11
Stwierdzenia proste = (Poprawny ,	12
Niepoprawny);	13
Treść stwierdzenia = "Wybrane metody koncypowania";	14
};	15
	16
Nazwa stwierdzenia – ZP1Koncepcje	17
{	18
Stwierdzenia proste = (Niepoprawne ,	19
Poprawne ,	20
Bardzo Dobre);	21
Treść stwierdzenia = "Zaproponowane Koncepcje";	22
};	23

Ocena części praktycznej

Stwierdzenie zbiorcze dotyczące ogólnej oceny umiejętności zastosowania poznanych metod do rozwiązywania zadań.

Wydruk 11.9: Algorytm rozwiązywania sieci przybliżonej

	1
Nazwa stwierdzenia – ZP1Koncypowanie	2
{	3
Stwierdzenia proste = (Błędnie Wykonane ,	4
Wykonane Poprawnie ,	5
Wykonane Bardzo Dobrze);	6
Treść stwierdzenia = "Zadanie Dotyczące Koncypowania";	7
};	8

Proces projektowo konstrukcyjny

Stwierdzenia dotyczące znajomości ogólnych zagadnień dotyczących procesu projektowo konstrukcyjnego.

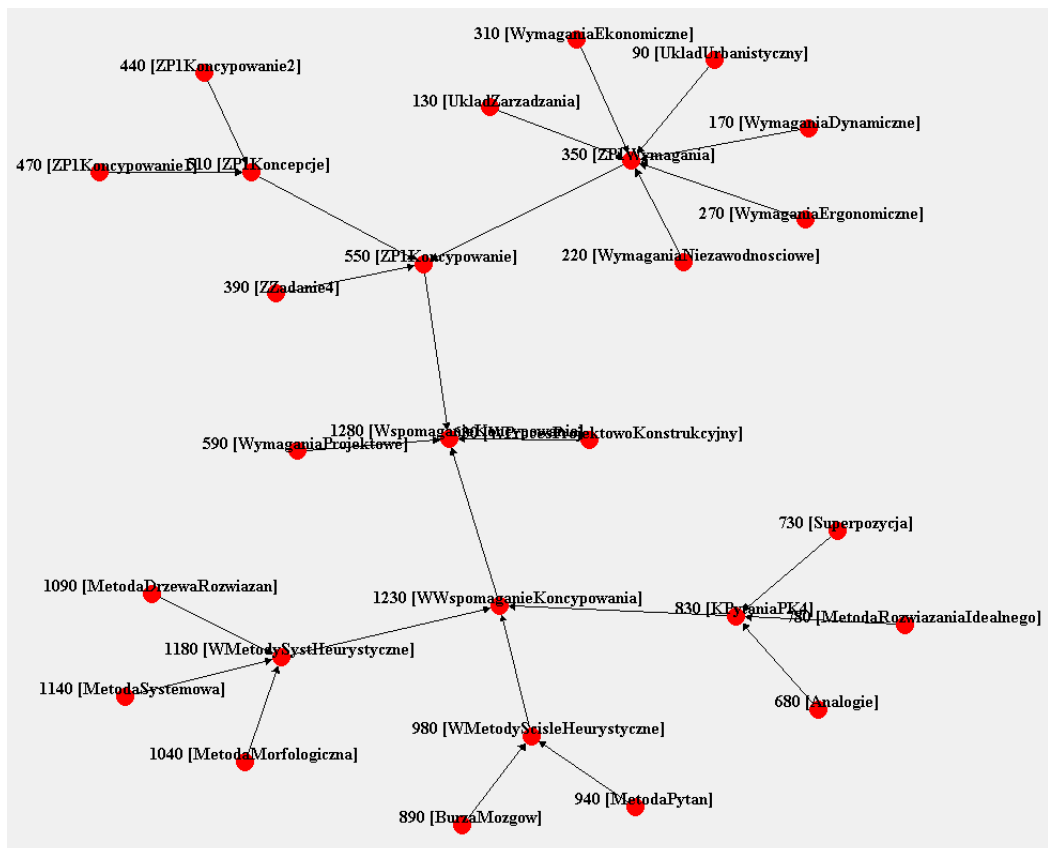
Wydruk 11.10: Algorytm rozwiązywania sieci przybliżonej

	1
Nazwa stwierdzenia – WymaganiaProjektowe	2
{	3
Stwierdzenia proste = (Brak Umiejętności ,	4
Uwzględniono Ogólne Wymagania Projektowe ,	5
Opracowano Wymagania Szczegółowe);	6
Treść stwierdzenia = "Umiejętność opracowywania wymagań projektowych";	7
};	8

	9
Nazwa stwierdzenia – WProcesProjektowoKonstrukcyjny	10
{	11
Stwierdzenia proste = (Całkowity Brak Znajomości	12
zagadnień ,	
Znajomość Ogólnego Schematu Procesu ,	13
Znajomość Zagadnień Szczegółowych);	14
Treść stwierdzenia = "Znajomość procesu	15
projektowo–konstrukcyjnego";	
};	16

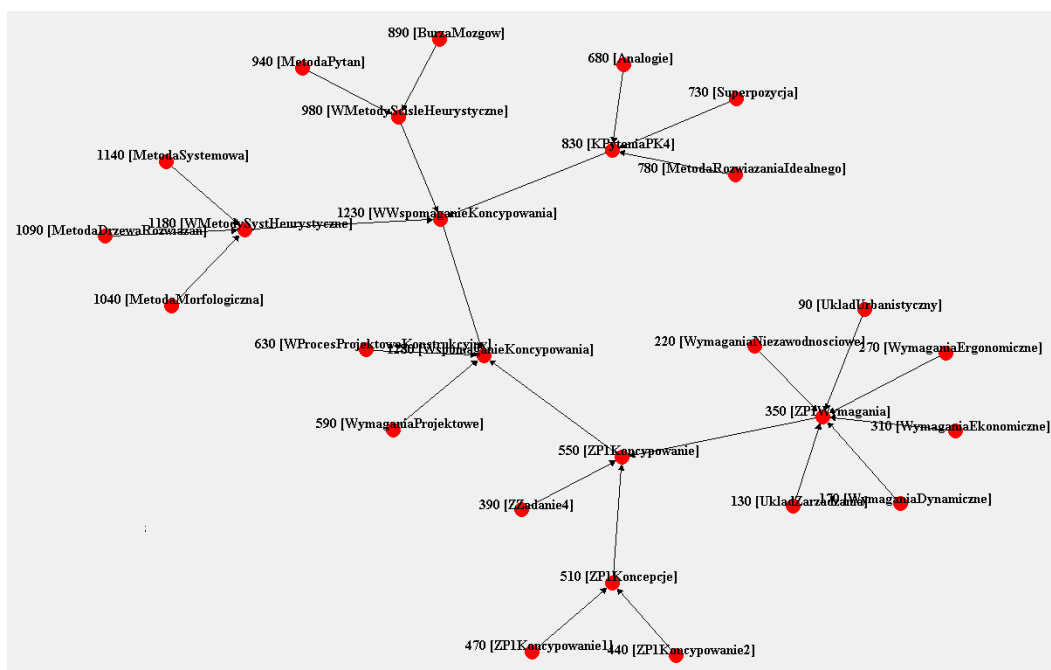
11.3.3. Warstwy sieci stwierdzeń

Pomiędzy przygotowanymi stwierdzeniami wykonano połączenia określające relacje nadrzędności i podrzędności poszczególnych stwierdzeń. Relacje te tworzą strukturę opracowanej sieci stwierdzeń. Przyjęcie różnych relacji pomiędzy stwierdzeniami odpowiada różnym oddziaływaniom opisującym różne aspekty oceny pracy studenta. Tak jak opisano to w rozdziale 11.2, przygotowano różne sieci, uwzględniające różne aspekty oceny studentów, w zależności od aspektu w jakim rozpatrywany jest proces nauczania. Każda z przygotowanych sieci została zapisana w systemie MMNET w postaci pojedynczej warstwy.



Rys. 11.1: Schemat połączeń stwierdzeń w sieci - Przykład 1.

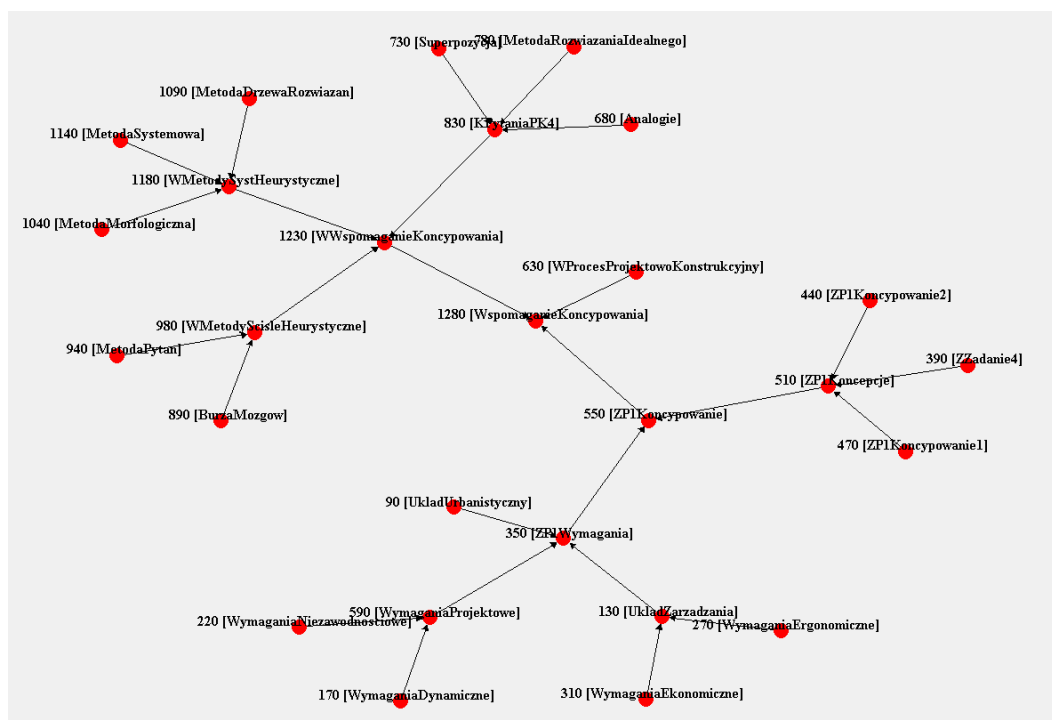
Na rys. 11.1 oraz rys. 11.2 przedstawiono układ połączeń dla sieci, które mogą zostać wykorzystane do oceniania postępów prac studentów dla kierunków mechanicznych. Są to warstwy 1 i 2. W Katedrze Podstaw Konstrukcji Maszyn, prowadzone są zajęcia projektowe z podstaw



Rys. 11.2: Schemat połączeń stwierdzeń w sieci - Przykład 2.

konstrukcji maszyn (PKM), na których studenci wykonują zadanie projektowe z wykorzystaniem metod wspomagających koncypowanie. Zadanie takie może być wykonywane na początku nauczania PKM, kiedy studenci nie posiadają jeszcze pełnej wiedzy z tego przedmiotu oraz pod koniec tego typu zajęć, kiedy studenci posiadają już pełną wiedzę dotyczącą zagadnień omawianych w ramach zajęć z PKM. Inne więc będą aspekty oceny takiego zadania. Inne będzie również podejście studentów do rozwiązania postawionego problemu. Będzie ono uwzględniało poznane w czasie nauki PKM zagadnienia. Dlatego też przedstawione na rysunkach postacie sieci różnią się. Główna różnica pomiędzy tymi sieciami dotyczy nie struktury sieci, a wartości prawdopodobieństw warunkowych, ustalonych pomiędzy poszczególnymi stwierdzeniami. W przypadku sieci przedstawionej na rys. 11.1, większy wpływ na ocenę końcową postępów pracy studentów ma znajomość zagadnień teoretycznych oraz procesu projektowo konstrukcyjnego. Wpływ na ocenę znajomości wiedzy z tego zakresu ma również umiejętność poprawnego zastosowania zagadnień teoretycznych w rozwiązaniu zadania. W przypadku pierwszej sieci na ocenę znajomości zagadnień mają wpływ zagadnienia teoretyczne uwzględniane we wszystkich węzłach wymienionych w grupie zagadnień teoretycznych. Ta część zagadnień jest uwzględniana w całości. Ocena postępów, uwzględniona w grupie stwierdzeń dotyczących zagadnień praktycznych jest bardziej zróżnicowana. Część stwierdzeń nie jest uwzględniana w ocenie końcowej lub ma mniejszy wpływ na tę ocenę. Można do nich zaliczyć np. stwierdzenia określające "Uwzględniono Wymagania Niezawodnościowe", uwzględniające "Opracowano Układ Urbanistyczny", czy "Uwzględniono Strukturę ogólną". W mniejszym zakresie uwzględniana jest również szczegółowość opracowanych koncepcji (węzeł "Ocena opracowanych koncepcji") oraz "Wybrane metody koncypowania".

W przypadku sieci przedstawionej na rys. 11.2, większą wagę mają stwierdzenia pominięte w poprzednim przypadku. Ocena postępów pracy studenta uwzględnia oprócz zagadnień teoretycznych również różne aspekty konieczne do rozważenia w ramach wykonywanego zadania projektowego. Ponieważ student, który wykonuje takie zadanie, powinien mieć znacznie większą wiedzę, wynikającą z zapoznania się z całością zagadnień omawianych w ramach kursu PKM, więc powinien on uwzględnić posiadaną wiedzę w rozwiązaniu postawionego przed nim



Rys. 11.4: Schemat połączeń stwierdzeń w sieci - Przykład 4.

że dla dobrej znajomości zagadnień związanych metodami wspomagającymi koncypowanie, konieczna jest:

- ogólna znajomość istoty metod związanych z operatorami metod heurystycznych,
- bardzo dobra znajomość metod ściśle heurystycznych,
- orientowanie się w metodach systematyczno-heurystycznych.

Nieco inaczej zdefiniowano tablice prawdopodobieństw warunkowych stwierdzeń w warstwach 3 i 4. Z tego względu wyliczone wartości różnią się od tych wyliczonych dla warstw 1 i 2.

Sposób przedstawiony powyżej jest bardziej przydatny w przypadku określenia wymagań jakie muszą być spełnione, aby można było mówić o określonym poziomie wiedzy uzyskanym przez studenta. Innym podejściem, jest podanie znajomości poszczególnych zagadnień przez studenta. Tym sposobem można określić, czy wiedza którą posiadał jest wystarczająca do uzyskania zaliczenia, czy też należy ją poszerzyć.

Osobnym zagadnieniem jest druga część sieci, która pozwala na określenie umiejętności zastosowania poznanych metod w rozwiązywaniu rzeczywistego zadania. W tym przypadku należałoby porównać uzyskane wyniki dla warstw 1 i 2 oraz 3 i 4. W sieci zapisanej w warstwie 1, główny nacisk położono na rozwiązanie zadania projektowego pod kątem przedstawionych koncepcji oraz ich oceny. Na rys. 11.6 i 11.7 przedstawiono wpływ węzła *ZP1 Koncypowanie Zadanie* na oczekiwane wartości węzłów określających stan prac nad koncepcjami rozwiązania zadania projektowego oraz jego oceny. Na rys. 11.6 przedstawiono oczekiwane wartości węzłów dla przypadku zadania wykonanego poprawnie, a na rys. 11.7 dla zadania wykonanego bardzo dobrze. Z przedstawionego przykładu wynika, że na ocenę końcową tego zadania niewielki wpływ mają elementy związane z wymaganiami projektowymi. Co jest zgodne z przyjętymi wcześniej założeniami.

Podobnie można przedstawić zadanie odwrotne, w którym oceniane będą poszczególne elementy zadania, a oczekiwanym wynikiem będzie ocena określająca, czy aktualny postęp prac jest zadowalający czy też wymaga jeszcze poprawy.

Wyniki obliczeń

Identyfikator węzła 1230 [WWspomaganieKoncypow] ?

Row.names	id_warstwy	id_węzła	status	Brak	Słaba	Dobra	BardzoDobra	waga_warstwy
2	1	27	1	0	0	1	0	1
3	2	27	1	0	0	1	0	1
4	3	26	1	0	0	1	0	1
5	4	26	1	0	0	1	0	1

Wyniki obliczeń

Identyfikator węzła 830 [KPytaniaPK4] ?

Row.names	id_warstwy	id_węzła	status	NieZnaMetod	ZnajomoscNazwMetod	OgolnaZnajomoscIstotyMetod	SzczegolowaZnajomoscMetod	waga_warstwy
2	1	19	2	0.04917764	0.2017739	0.6276863	0.1213621	1
3	2	19	2	0.01231406	0.1087451	0.6729918	0.2059490	1
4	3	18	2	0.01231406	0.1087451	0.6729918	0.2059490	1
5	4	18	2	0.01231406	0.1087451	0.6729918	0.2059490	1

Wyniki obliczeń

Identyfikator węzła 980 [WMetodyScisleHeurystyczne] ?

Row.names	id_warstwy	id_węzła	status	NieZna	OrientujeSie	ZnaBardzoDobrze	waga_warstwy
2	1	22	2	0.25644077	0.3476426	0.3959167	1
3	2	22	2	0.05638105	0.3158852	0.6277337	1
4	3	21	2	0.05638105	0.3158852	0.6277337	1
5	4	21	2	0.05638105	0.3158852	0.6277337	1

Wyniki obliczeń

Identyfikator węzła 1180 [WMetodySystHeurystyczne] ?

Row.names	id_warstwy	id_węzła	status	NieZna	OrientujeSie	ZnaBardzoDobrze	Metode	waga_warstwy
2	1	26	2	0.3590091	0.4841877	0.1568032		1
3	2	26	2	0.1477125	0.5327050	0.3195825		1
4	3	25	2	0.1477125	0.5327050	0.3195825		1
5	4	25	2	0.1477125	0.5327050	0.3195825		1

Rys. 11.5: Ocena znajomości zagadnień teoretycznych dla warstw 1, 2, 3 i 4.

Trochę inaczej zdefiniowane zostały zależności w warstwie 2. W tym przypadku, ponieważ studenci posiadają już większą wiedzę z zakresu projektowania, inaczej ustalono wpływ poszczególnych węzłów na ocenę końcową. Przykładowe wyniki działania tej warstwy przedstawiono na rys. 11.8. W tej warstwie zmienione zostały wartości w tablicach prawdopodobieństw warunkowych w taki sposób, aby zwiększyć wpływ wymagań projektowych i ich spełnienia na ocenę pracy studenta. Na rys. 11.8, przedstawiono przykładowe wyniki działania sieci, w przypadku, gdy ustalane są wartości węzłów elementów ocenianych. Ustalono zaawansowanie prac nad poszczególnymi elementami projektu (opracowanie koncepcji, ocena koncepcji) oraz spełnienie wymagań projektowych. Na tej podstawie określony jest stopień zaawansowania prac nad całym zadaniem projektowym. Wynikiem ostatecznym jest uwzględnienie również w ocenie końcowej wyników prowadzonej wcześniej weryfikacji wiedzy teoretycznej. Inne podejście do zadania przedstawiono na rys. 11.9. W tym przypadku ustalono, że konieczne jest uzyskanie poprawnego rozwiązania zadania projektowego. Na tej podstawie można określić, jakie kryteria muszą być spełnione aby taki stopień zaawansowania prac na zadaniem projektowym uzyskać. Jak widać na przedstawionym rysunku, dla zatwierdzenia zadania na tym poziomie konieczne jest co najmniej poprawne opracowanie koncepcji rozwiązania zadania, ocena opracowanych koncepcji, która może zawierać drobne błędy oraz poprawne uwzględnienie wymagań projektowych. Ponieważ w tym przykładzie nie uwzględniono wymagań dotyczących urbanistyki oraz struktury ogólnej, nie mają one wpływu na końcową ocenę. Pozostałe kryteria muszą być spełnione co najmniej w stopniu dostatecznym.

Wyniki obliczeń

Identyfikator węzła: 550 [ZP1Koncypowanie]

Row.names	id_warstwy	id_węzła	status	BlednieWykonane	WykonanePoprawnie	WykonaneBardzoDobrze	waga_warstwy
2	1	13	1	0	1	0	1
3	2	13	1	0	1	0	1
4	3	14	1	0	1	0	1
5	4	14	1	0	1	0	1

Wyniki obliczeń

Identyfikator węzła: 350 [ZP1Wymagania]

Row.names	id_warstwy	id_węzła	status	Niepoprawne	Poprawne	BardzoDobre	waga_warstwy
2	1	8	2	0.2890994	0.5990182	0.11188240	1
3	2	8	2	0.3217062	0.5189240	0.15936980	1
4	3	9	2	0.3430088	0.5441536	0.11283763	1
5	4	9	2	0.1585568	0.7692684	0.07217471	1

Wyniki obliczeń

Identyfikator węzła: 510 [ZP1Koncepcje]

Row.names	id_warstwy	id_węzła	status	Niepoprawne	Poprawne	BardzoDobre	waga_warstwy
2	1	12	2	0.2641234	0.5474309	0.1884457	1
3	2	12	2	0.2271059	0.5263938	0.2465003	1
4	3	13	2	0.3842526	0.4355907	0.1801567	1
5	4	13	2	0.4477049	0.3943930	0.1579021	1

Wyniki obliczeń

Identyfikator węzła: 390 [ZZadanie4]

Row.names	id_warstwy	id_węzła	status	Niepoprawna	BledyWTablicyOcen	Poprawna	waga_warstwy
1	0	390	2	0.2593281	0.3334252	0.4072467	0
2	1	9	2	0.1904512	0.3901802	0.4193686	1
3	2	9	2	0.2636660	0.3052109	0.4311231	1
4	3	12	2	0.2852364	0.3167766	0.3979870	1

Rys. 11.6: Wyniki obliczeń sieci dla przykładu 1.

Inne wymagania zostały postawione przy określaniu połączeń pomiędzy stwierdzeniami tworzącymi węzły sieci stwierdzeń znajdujących się na warstwie 3 i 4. W przypadku tych sieci, inaczej również zostały położone akcenty przy określaniu prawdopodobieństw warunkowych poszczególnych stwierdzeń. Inna struktura tych sieci została narzucona innymi wymaganiami i ich przeznaczeniem. Z przeprowadzonych analiz tych sieci wynika, że w przypadku sieci znajdującej się na warstwie 3, większy nacisk położono na opracowanie i spełnienie wymagań projektowych, a mniejszy na samo wykonanie zadania projektowego. Różnice nie są duże, jednak na tyle istotne, że skutkują zmianami oceny poprawności wykonania zadania projektowego. Zupełnie inaczej w stosunku do pozostałych przykładów ustalone zostały relacje pomiędzy węzłami. Dotyczy to zarówno wymagań projektowych jak również samej oceny realizacji zadania projektowego. Takie podejście wynika z innej grupy docelowej studentów, których postępy prac będą oceniane. Inne wobec tych studentów są stawiane wymagania odnośnie zakresu poznawanej wiedzy i tych jej elementów, na które należy zwrócić uwagę.

W przypadku sieci znajdującej się na warstwie 4, zmianie uległy głównie relacje pomiędzy węzłami dotyczącymi wymagań projektowych oraz tablice wartości prawdopodobieństw

Wyniki obliczeń

Identyfikator węzła: 550 [ZP1Koncypowanie]

Row.names	id_warstwy	id_węzła	status	BlednieWykonane	WykonanePoprawnie	WykonaneBardzoDobrze	waga_warstwy
2	1	13	1	0	0	1	1
3	2	13	1	0	0	1	1
4	3	14	1	0	0	1	1
5	4	14	1	0	0	1	1

Wyniki obliczeń

Identyfikator węzła: 350 [ZP1Wymagania]

Row.names	id_warstwy	id_węzła	status	Niepoprawne	Poprawne	BardzoDobre	waga_warstwy
2	1	8	2	0.2423638	0.4517733	0.3058629	1
3	2	8	2	0.1868516	0.3109163	0.5022322	1
4	3	9	2	0.1764391	0.3381640	0.4853969	1
5	4	9	2	0.0000000	0.6987957	0.3012043	1

Wyniki obliczeń

Identyfikator węzła: 510 [ZP1Koncepcje]

Row.names	id_warstwy	id_węzła	status	Niepoprawne	Poprawne	BardzoDobre	waga_warstwy
2	1	12	2	0.00000000	0.1944345	0.8055655	1
3	2	12	2	0.00000000	0.3307863	0.6692137	1
4	3	13	2	0.12453114	0.2156286	0.6598403	1
5	4	13	2	0.00000000	0.3208154	0.6791846	1

Wyniki obliczeń

Identyfikator węzła: 390 [ZZadanie4]

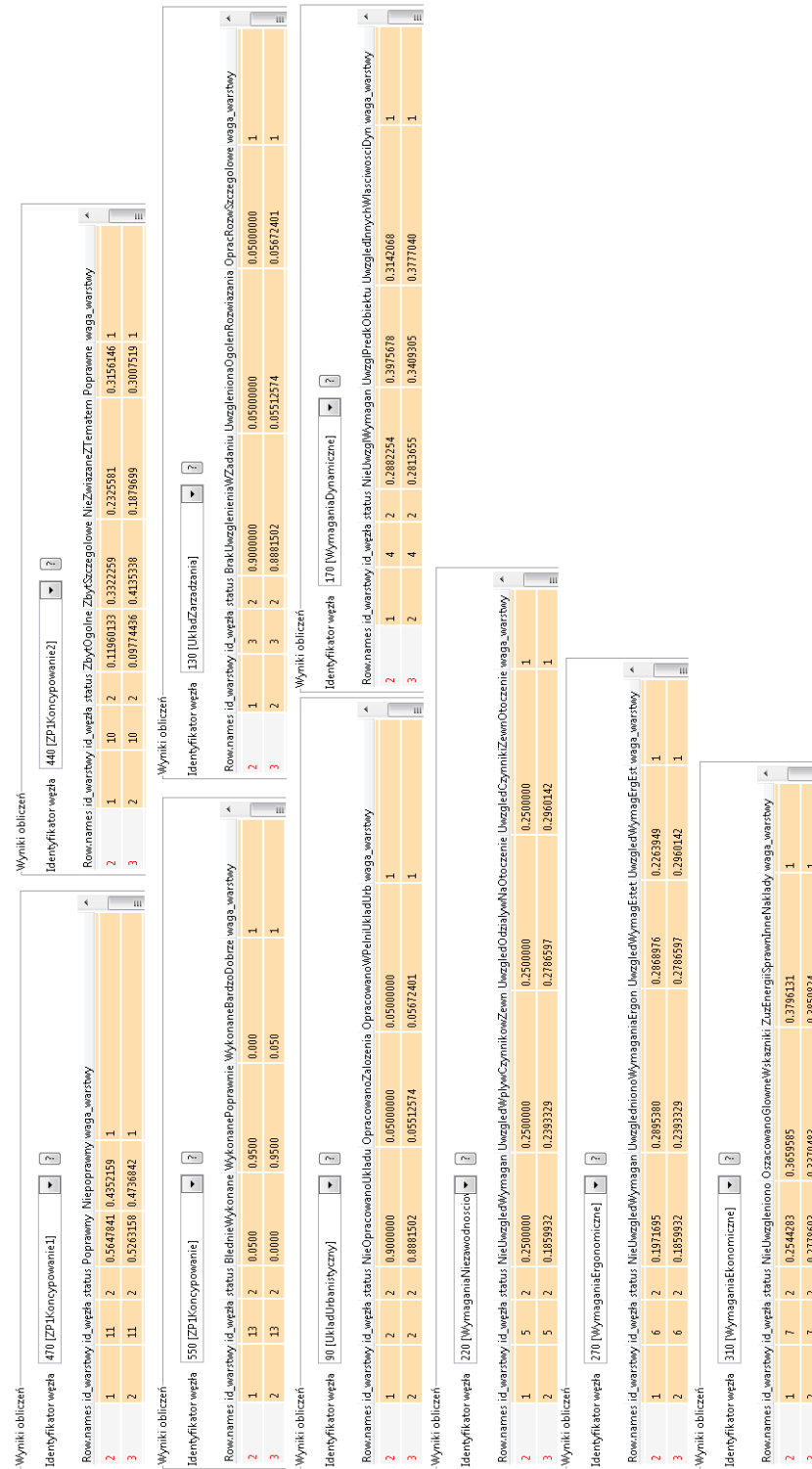
Row.names	id_warstwy	id_węzła	status	Niepoprawna	BledyWTablicyOcen	Poprawna	waga_warstwy
2	1	9	2	0.06833141	0.06833141	0.8633372	1
3	2	9	2	0.05543826	0.12708279	0.8174789	1
4	3	12	2	0.23448521	0.27328066	0.4922341	1
5	4	12	2	0.20943343	0.26544984	0.5251167	1

Rys. 11.7: Wyniki obliczeń sieci dla przykładu 1.

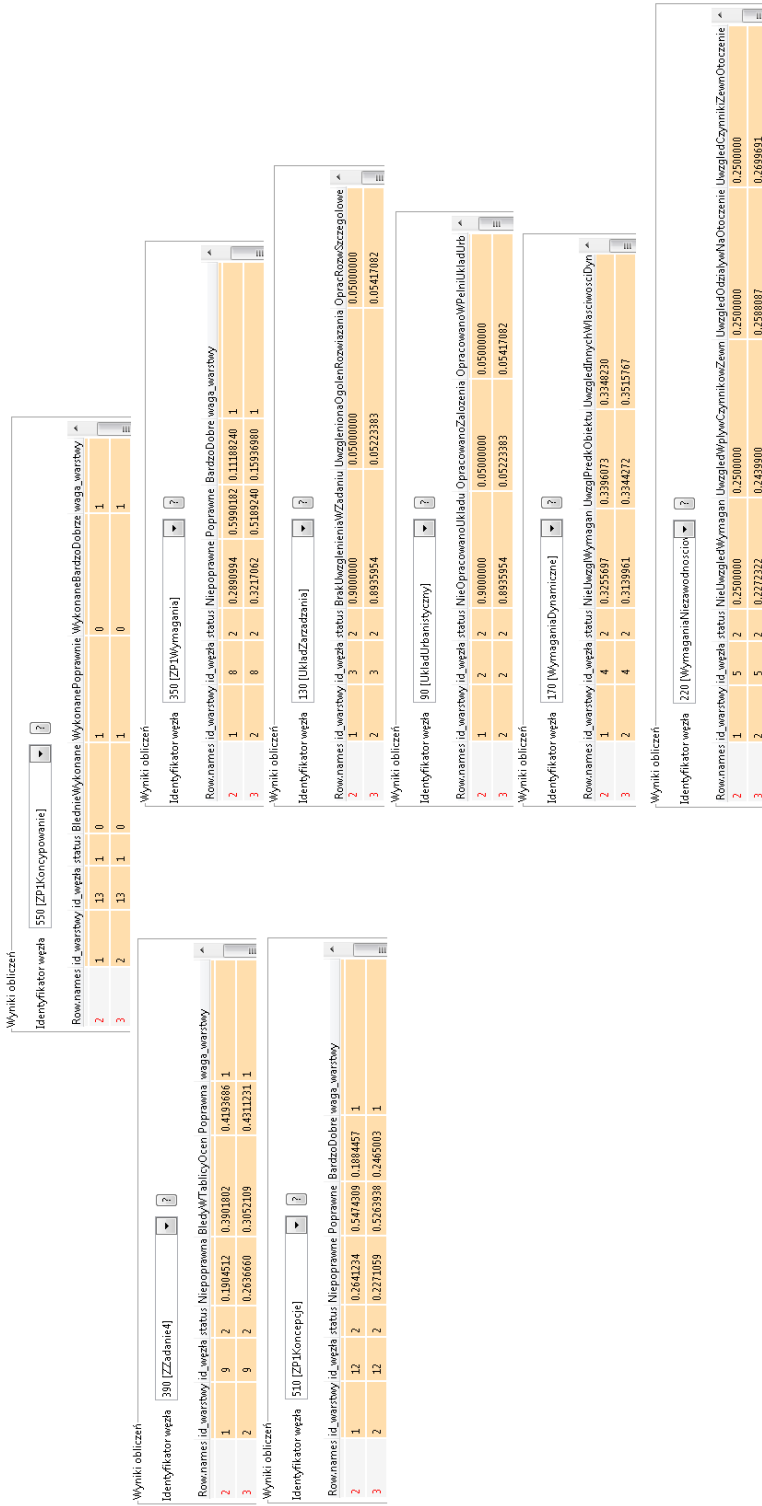
warunkowych stwierżeń złożonych. Z przeprowadzonych obliczeń sieci dla różnych wartości stwierżeń, wynika, że duży wpływ na ocenę postępu prac nad zadaniem projektowym, mają takie elementy jak:

- opracowanie koncepcji rozwiązania zadania,
- wybór metody koncypowania,
- opracowanie wymagań projektowych,
- opracowanie struktury ogólnej,
- opracowanie układu urbanistycznego.

Poprawne wykonanie tych elementów zadania, jest konieczne do jego bardzo dobrej oceny.



Rys. 11.8: Wyniki obliczeń sieci dla przykładu 2.



Rys. 11.9: Wyniki obliczeń sieci dla przykładu 2.

11.5. Wnioski

Celem prac było sprawdzenie możliwości oceniania postępu nabywania wiedzy przez studentów w zakresie nauczania wybranego zagadnienia. W tym celu opracowano przykładowe sieci stwierdzeń umożliwiające wspomaganie procesu podejmowania decyzji dotyczących oceny postępu prac studentów. Z przeprowadzonych analiz działania samych sieci wynika, że możliwe jest wspomaganie takiego procesu. Można opracować sieć stwierdzeń, o określonej strukturze, która pozwoli ocenić, w jaki sposób wiedza przekazywana studentowi jest przez niego przyswajana i jak potrafi on z tej wiedzy korzystać. Tak przygotowane sieci stwierdzeń mogą zostać wykorzystane np. w systemach wspomagających proces nauczania na odległość. Koncepcję takiego systemu przedstawiono np. w [11.3][11.7]. Tego typu systemy mają przygotowane moduły z określonymi zagadnieniami, które studenci przyswajają sobie w procesie uczenia. Dzięki zastosowaniu takich sieci, można poprzez okresowe sprawdzanie wiedzy studentów, wskazywać im z jakiego zakresu wiedzę powinni jeszcze poszerzyć lub poznać. Dzięki temu proces przekazywania wiedzy można zindywidualizować do potrzeb konkretnego studenta. Daje to również możliwość automatycznego dopasowania zagadnień, z którymi student powinien się jeszcze zapoznać bardziej szczegółowo oraz takich, które student powinien sobie tylko powtórzyć.

Oczywiście opracowane sieci są prostymi przykładami ilustrującymi jedynie możliwości zastosowania sieci stwierdzeń w tym zakresie. Do rzeczywistych rozwiązań konieczne byłoby opracowanie sieci dużo bardziej złożonych, które obejmowałyby dużo szerszy zakres wiedzy. W takim przypadku najważniejszym elementem takiej sieci byłaby nie tylko jej struktura, ale również wagi przypisane poszczególnym stwierdzeniom. To wymagałoby przeprowadzenia wielu testów związanych z dostosowaniem wag stwierdzeń. Taką sieć można również wzbogacić o cały system podpowiedzi i pomocy, tłumaczący działanie sieci stwierdzeń. Przykładem takiego systemu jest system DIADYN [11.1]. W systemie tym do zapisu wiedzy również wykorzystywane są sieci stwierdzeń. W systemie DIADYN, oprócz samych sieci można przygotować bibliotekę haseł, która umożliwia lepsze zrozumienie treści stwierdzeń z których zbudowana jest sieć. Przykłady budowy sieci stwierdzeń, ich konstruowania i zastosowania można znaleźć np. w [11.5], [11.4], [11.6].

Bibliografia

- [11.1] Cholewa W., redaktor. *Szkieletowy system doradczy DIADYN*. Politechnika Śląska, Katedra Podstaw Konstrukcji Maszyn, 2008.
- [11.2] Dietrych Janusz. *System i konstrukcja*. WNT, Warszawa, 1985.
- [11.3] Psiuk K. A concept of e-learning environment. *Artificial Intelligence Methods*, 2005.
- [11.4] Psiuk K. Konstruowanie sieci stwierdzeń. Cholewa W., redaktor, *Szkieletowy system doradczy DIADYN*, strony 55–65. Politechnika Śląska, Katedra Podstaw Konstrukcji Maszyn, 2008.
- [11.5] Psiuk K. Redagowanie sieci stwierdzeń. Cholewa W., redaktor, *Szkieletowy system doradczy DIADYN*, strony 45–54. Politechnika Śląska, Katedra Podstaw Konstrukcji Maszyn, 2008.
- [11.6] Psiuk K. Stosowanie sieci stwierdzeń. Cholewa W., redaktor, *Szkieletowy system doradczy DIADYN*, strony 83–88. Politechnika Śląska, Katedra Podstaw Konstrukcji Maszyn, 2008.
- [11.7] Psiuk K. Koncepcja systemu do automatyzacji doboru materiałów szkoleniowych w nauczaniu pkm. *Materiały Konferencji Wirtualny Uniwersytet 2010*, 2010.

[11.8] Tarnowski Wojciech. *Podstawy projektowania technicznego*. WNT, Warszawa, 1997.

Rozdział 12

Przykłady sieci stwierdzeń przeznaczonych do wspomagania początkowej fazy procesu projektowania układów napędowych

Sebastian RZYDZIK

W rozdziale przedstawiono zastosowanie sieci stwierdzeń do wspomagania projektowania układów napędowych. Przez fazę początkową rozumie się wstępny projekt układu napędowego, tj. silnik elektryczny (wraz z odpowiednim sterowaniem), sprzęgło oraz przekładnię mechaniczną, w którym nie wymaga się zastosowania dokładnych typów i modeli podzespołów składowych, tylko ich rodzaj oraz ich podstawowe własności opisane wartościami jakościowymi.

12.1. Wybrane podejścia do projektowania układów napędowych

Przed przystąpieniem do budowy sieci stwierdzeń, wyróżniono cztery podejścia do projektowania układów napędowych. Każde podejście pozwala rozpatrywać problem wspomagania początkowej fazy procesu projektowania układów napędowych w różnym kontekście, w zależności od zadanej liczby i rodzajów parametrów wejściowych.

12.1.1. Standardowa metoda doboru elementów napędu

Pierwsze podejście nazwano „Standardowa metoda doboru elementów napędu” ze względu na rodzaj danych wejściowych, które używa się w przypadku standardowego doboru elementów napędu. Jako stwierdzenia wejściowe do sieci stwierdzeń wyróżniono:

moc MOC {L,M,H}

obrotyPWy PRĘDKOŚĆ KĄTOWA NA WYJŚCIU PRZEKŁADNI {L,M,H}

odl ODLEGŁOŚĆ POMIĘDZY OSIĄ WAŁKA SILNIKA A OSIĄ WAŁKA WYJŚCIOWEGO PRZEKŁADNI {L,M,H}

Wartości ujęte w nawiasach klamrowych oznaczają:

L niska (Low)

M średnia (Medium)

H wysoka (Height)

12.1.2. Dobór elementów napędu ze względu na precyzyjne sterowanie położeniem kątowym

Drugie podejście rozpatruje problem doboru elementów układu napędowego ze względu na precyzyjne sterowanie położeniem kątowym. Jako stwierdzenia wejściowe do sieci stwierdzeń wyróżniono:

obrotyPWy PRĘDKOŚĆ KĄTOWA NA WYJŚCIU PRZEKŁADNI {L,M,H}

rpol REGULACJA POŁOŻENIA {Y,N}

odl ODLEGŁOŚĆ POMIĘDZY OSIĄ WAŁKA SILNIKA A OSIĄ WAŁKA WYJŚCIOWEGO PRZEKŁADNI {L,M,H}

Wartości ujęte w nawiasach klamrowych oznaczają:

Y tak (Yes)

N nie (No)

L niska (Low)

M średnia (Medium)

H wysoka (Height)

12.1.3. Dobór elementów napędu ze względu na precyzyjne sterowanie prędkością kątową

Trzecie podejście rozpatruje problem doboru elementów układu napędowego ze względu na precyzyjne sterowanie prędkością kątową. Jako stwierdzenia wejściowe do sieci stwierdzeń wyróżniono:

obrotyPWy PRĘDKOŚĆ KĄTOWA NA WYJŚCIU PRZEKŁADNI {L,M,H}

rpred REGULACJA PRĘDKOŚCI KĄTOWEJ {Y,N}

odl ODLEGŁOŚĆ POMIĘDZY OSIĄ WAŁKA SILNIKA A OSIĄ WAŁKA WYJŚCIOWEGO PRZEKŁADNI {L,M,H}

Wartości ujęte w nawiasach klamrowych oznaczają:

Y tak (Yes)

N nie (No)

L niska (Low)

M średnia (Medium)

H wysoka (Height)

12.1.4. Dobór elementów napędu ze względu na bezpieczeństwo użytkownika

Czwarte podejście rozpatruje problem doboru elementów układu napędowego ze względu na bezpieczeństwo jego użytkownika. Jako stwierdzenia wejściowe do sieci stwierdzeń wyróżniono:

bezp BEZPIECZEŃSTWO UŻYTKOWANIA {Y,N}

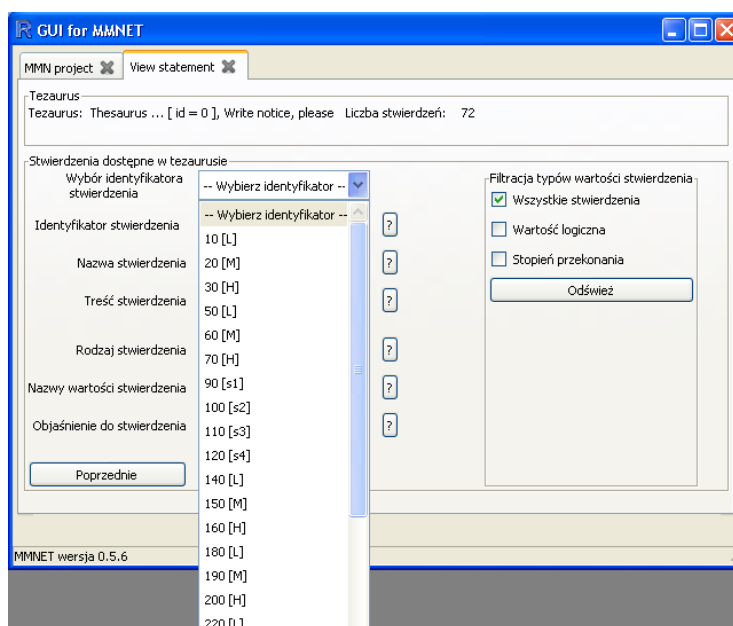
Wartości ujęte w nawiasach klamrowych oznaczają:

Y tak (Yes)

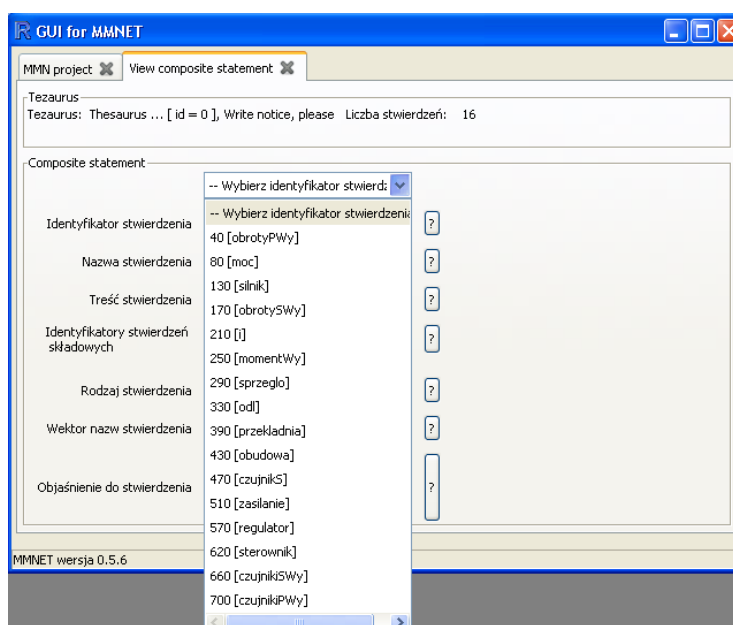
N nie (No)

12.2. Lista stwierdzeń

W tym punkcie przedstawiono listę stwierdzeń, z których budowano poszczególne sieci stwierdzeń dla metod doboru elementów układu napędowego przedstawionych we wcześniejszych punktach. Na podstawie listy stwierdzeń utworzono słownik stwierdzeń (tezaurus) z użyciem pakietu MMNET. Zdefiniowano 72 stwierdzenia proste (rys. 12.1), z czego część z tych stwierdzeń utworzyła 16 stwierdzeń złożonych (rys. 12.2).



Rys. 12.1: Pakiet MMNET. Stwierdzenia proste



Rys. 12.2: Pakiet MMNET. Stwierdzenia złożone

Stwierdzenia przyjęto oznaczać w następujący sposób:

id:NAZWA - identyfikator stwierdzenia nadany przez pakiet MMNET [nazwa skrócona]:NAZWA STWIERDZENIA

x{.}:wartość_x - identyfikator wartości stwierdzenia (identyfikator składa się z identyfikatora stwierdzenia i kolejnej wartości liczby porządkowej):nazwa wartości stwierdzenia o identyfikatorze x

80 [moc]: MOC

P1: L

P2: M

P3: H

250 [momentWy]: MOMENT NA WYJŚCIU PRZEKŁADNI

M1: L

M2: M

M3: H

40 [obrotyPWy]: PRĘDKOŚĆ KĄTOWA NA WYJŚCIU PRZEKŁADNI

wp1: L

wp2: M

wp3: H

170 [obrotySWy]: PRĘDKOŚĆ KĄTOWA NA WYJŚCIU SILNIKA

ws1: L

ws2: M

ws3: H

210 [i]: PRZEŁOŻENIE

i1: L

i2: M

i3: H

330 [odl]: ODLEGŁOŚĆ POMIĘDZY OSIĄ WAŁKA SILNIKA A OSIĄ WAŁKA WYJŚCIOWEGO PRZEKŁADNI

l1: L

l2: M

l3: H

710 [rpol]: REGULACJA POŁOŻENIA

rpol1: Y

rpol2: N

720 [rpred]: REGULACJA PRĘDKOŚCI KĄTOWEJ

rpred1: Y

rpred2: N

510 [zasilanie]: ZASILANIE

z1: napięcie stałe

z2: napięcie zmienne niskie

z3: napięcie zmienne wysokie

620 [sterownik]: STEROWNIK

st1: sterownik silnika prądu stałego

st2: falownik

st3: sterownik silnika krokowego

st4: brak sterownika

570 [regulator]: REGULATOR

reg1: położenia

reg2: prędkości obrotowej

reg3: brak regulatora

130 [silnik]: SILNIK

s1: 1-fazowy

s2: prądu stałego

s3: asynchroniczny 3-fazowy

s4: krokowy

470 [czujnikS]: CZUJNIKI SILNIKA

cs1: przeciążeniowy

cs2: przeciwprzepięciowy

cs3: brak czujnika

660 [czujnikiSWy]: CZUJNIKI NA WAŁKU WYJŚCIOWYM SILNIKA

csw1: czujnik położenia

csw2: czujnik prędkości kątowej

csw3: brak czujnika

290 [sprzęgło]: SPRZĘGŁO (pomiędzy silnikiem a przekładnią)

sp1: niepodatne skrętne

sp2: podatne skrętne

sp3: przeciążeniowe

390 [przekładnia]: PRZEKŁADNIA

p1: przekładnia cięgnowa z pasami klinowymi

p2: przekładnia cięgnowa z pasami synchronicznymi

p3: przekładnia zębata

p4: przekładnia planetarna

p5: przekładnia ślimakowa

700 [czujnikiPWy]: CZUJNIKI NA WAŁKU WYJŚCIOWYM PRZEKŁADNI

cpw1: czujnik położenia

cpw2: czujnik prędkości kątowej

cpw3: brak czujnika

430 [obudowa]: OBUDOWA

o1: osłona

o2: korpus uszczelniony

o3: brak obudowy

530 [smarowanie]: SMAROWANIE OLEJEM

smar1: Y

smar2: N

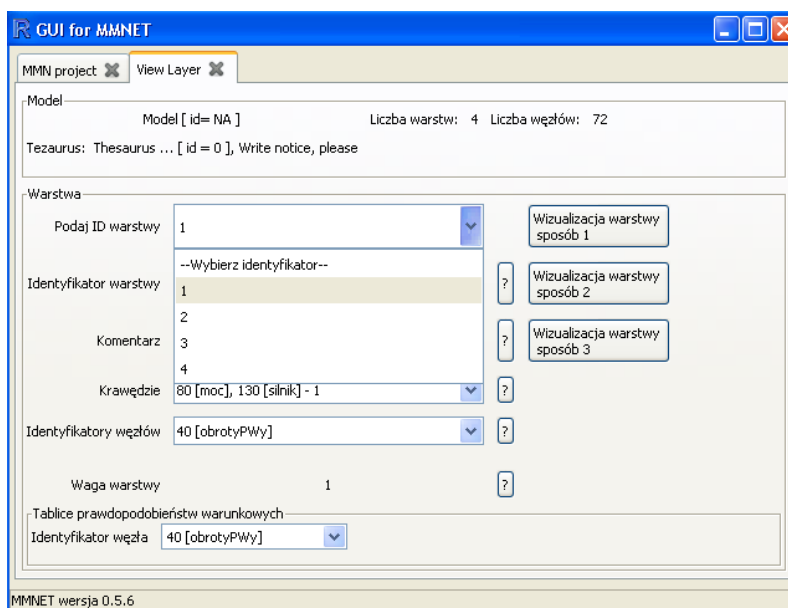
520 [bezp]: BEZPIECZEŃSTWO UŻYTKOWANIA

bezp1: Y

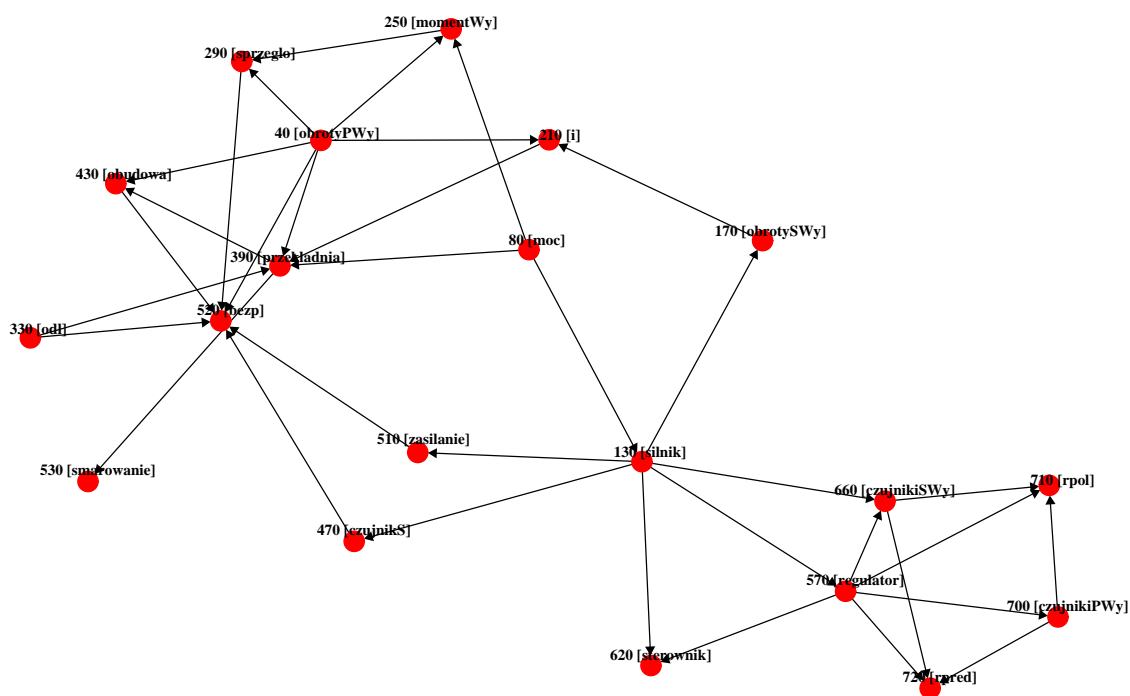
bezp2: N

12.3. Postacie opracowanych sieci stwierdzeń

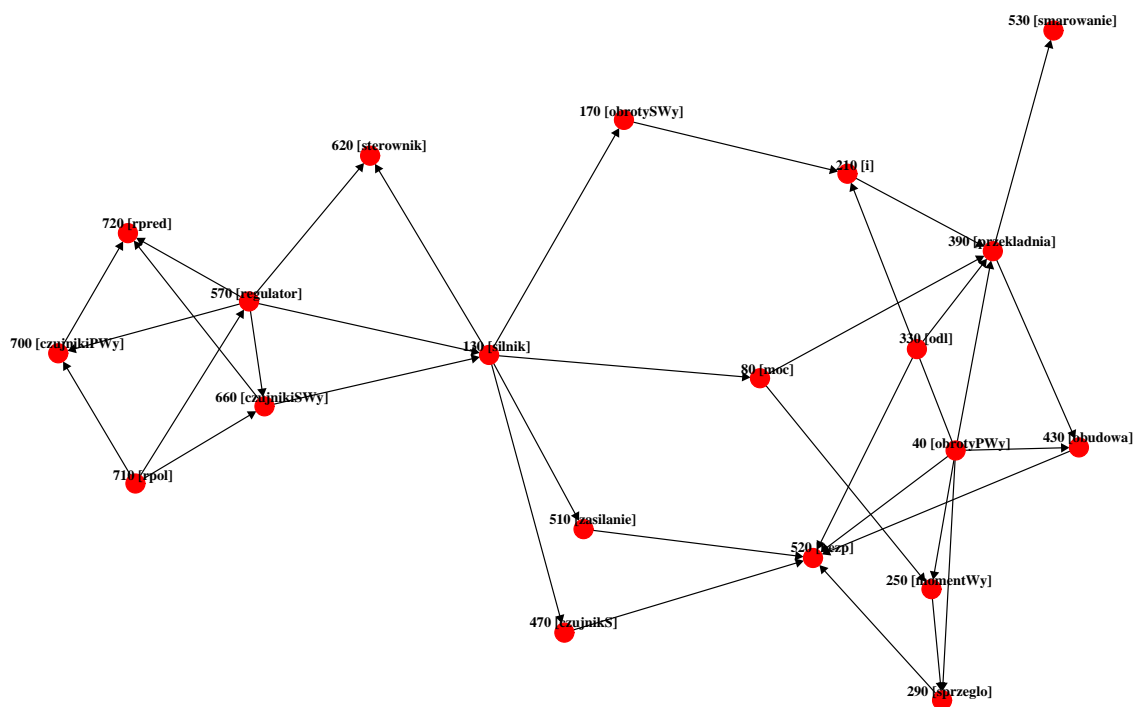
Opracowany słownik stwierdzeń pozwolił na utworzenie czterech sieci stwierdzeń, z których każda pokazuje odmienne podejście do procesu projektowania układów napędowych. Wszystkie sieci utworzono za pomocą pakietu MMNET jako jeden model, w którym każda warstwa odpowiada jednej sieci stwierdzeń (rys. 12.3). Na rys. 12.4 pokazano sieć stwierdzeń w przypadku standardowej metody doboru elementów napędu (warstwa nr 1), na rys. 12.5 pokazano przykład sieci do doboru elementów napędu ze względu na precyzyjne sterowanie położeniem kątowym (warstwa nr 2), na rys. 12.6 pokazano przykład sieci do doboru elementów napędu ze względu na precyzyjne sterowanie prędkością kątową (warstwa nr 3), a na rys. 12.7 pokazano postać sieci, która pozwala dobrać elementy układu napędowego ze względu na bezpieczeństwo jego użytkowania (warstwa nr 4). Wartości prawdopodobieństw warunkowych przypisane do stwierdzeń były dobierane zgodnie z ogólnie przyjętymi metodami projektowania i konstruowania układów napędowych [12.1] [12.2] [12.3].



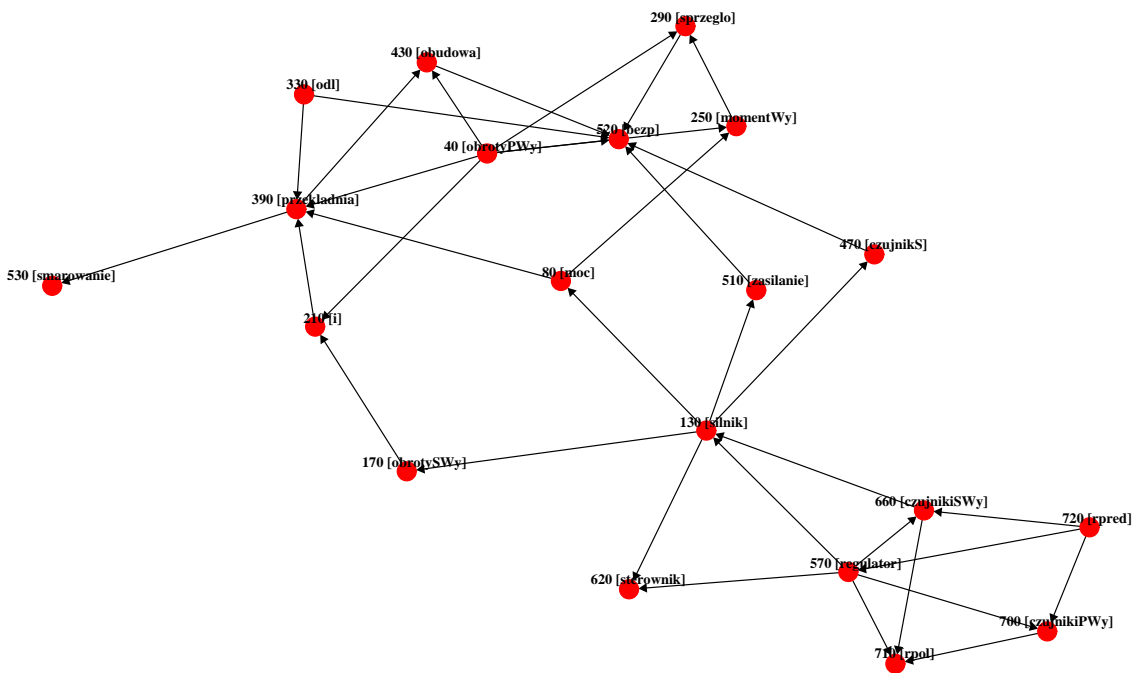
Rys. 12.3: Pakiet MMNET. Model wielowarstwowy



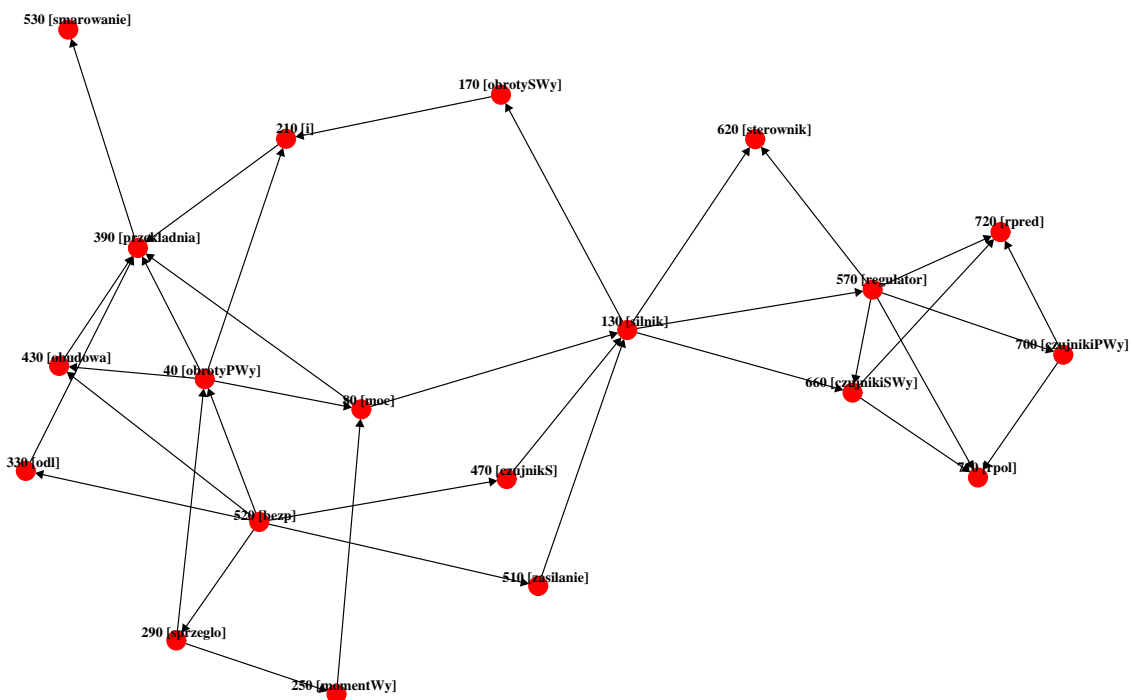
Rys. 12.4: Podejście 1. Standardowa metoda doboru elementów napędu



Rys. 12.5: Podejście 2. Dobór elementów napędu ze względu na precyzyjne sterowanie położeniem kątowym



Rys. 12.6: Podejście 3. Dobór elementów napędu ze względu na precyzyjne sterowanie prędkością kątową



Rys. 12.7: Podejście 4. Dobór elementów napędu ze względu na bezpieczeństwo użytkownika

12.4. Przykłady użycia

12.4.1. Standardowa metoda doboru elementów napędu

W tab. 12.1 i w tab. 12.2 pokazano wyniki testów jakie przeprowadzono dla sieci opracowanej dla podejścia 1.

Tab. 12.1: Lista wartości wybranych stwierdzeń dla sieci utworzonej dla podejścia 1. (opis w tekście)

Lp.	Stwierdzenia wejściowe (niezależne)			Wybrane stwierdzenia wyjściowe (obserwowane)				
	moc	obrotyPWy	odl	momentWy	obrotySWy	i (przełożenie)	przekładnia	silnik
1	L:100	L:100	L:100	L:100	L:46.7 M:43.3 H:10.0	L:15.6 M:58.9 H:25.6	p1:22.3 p2:22.3 p3:34.9 p4:10.2 p5:10.2	s1:33.3 s2:33.3 s3:0.0 s4:33.3
2	L:100	M:100	L:100	L:30 M:70	L:46.7 M:43.3 H:10.0	L:14.4 M:71.1 H:14.4	p1:17.8 p2:17.8 p3:61.4 p4:1.44 p5:1.44	s1:33.3 s2:33.3 s3:0.0 s4:33.3
3	L:100	H:100	L:100	M:80 H:20	L:46.7 M:43.3 H:10.0	L:3.33 M:46.7 H:50	p1:5.33 p2:5.33 p3:89.3 p4:0.0 p5:0.0	s1:33.3 s2:33.3 s3:0.0 s4:33.3
4	M:100	L:100	L:100	L:30 M:70	L:36 M:40.5 H:23.5	L:12 M:52.5 H:35.5	p1:19.4 p2:19.4 p3:32.9 p4:14.2 p5:14.2	s1:25 s2:35 s3:20 s4:20
5	M:100	M:100	L:100	M:100	L:36 M:40.5 H:23.5	L:13.5 M:73 H:13.5	p1:18 p2:18 p3:61.3 p4:1.35 p5:1.35	s1:25 s2:35 s3:20 s4:20
6	M:100	H:100	L:100	M:60 H:40	L:36 M:40.5 H:23.5	L:7.83 M:48.3 H:43.8	p1:6.4 p2:6.4 p3:87.2 p4:0 p5:0	s1:25 s2:35 s3:20 s4:20
7	H:100	L:100	L:100	M:80 H:20	L:12 M:26 H:62	L:4 M:30 H:66	p1:1.7 p2:5.1 p3:33.8 p4:29.7 p5:29.7	s1:0 s2:20 s3:80 s4:0
8	H:100	M:100	L:100	M:60 H:40	L:12 M:26 H:62	L:8.67 M:82.7 H:8.67	p1:2.74 p2:6.39 p3:84.8 p4:3.03 p5:3.03	s1:0 s2:20 s3:80 s4:0
9	H:100	H:100	L:100	H:100	L:12 M:26 H:62	L:20.7 M:46.7 H:32.7	p1:0 p2:0 p3:100 p4:0 p5:0	s1:0 s2:20 s3:80 s4:0
10	M:100	L:100	M:100	L:30 M:70	L:36 M:40.5 H:23.5	L:12 M:52.5 H:35.5	p1:25.2 p2:25.2 p3:46.1 p4:1.78 p5:1.78	s1:25 s2:35 s3:20 s4:20

Tab. 12.2: Lista wartości wybranych stwierdzeń dla sieci utworzonej dla podejścia 1. (opis w tekście, ciąg dalszy tab. 12.1)

Lp.	Stwierdzenia wejściowe (niezależne)			Wybrane stwierdzenia wyjściowe (obserwowane)				
	moc	obrotyPWy	odl	momentWy	obrotySWy	i (przełożenie)	przekładnia	silnik
11	M:100	M:100	M:100	M:100	L:36 M:40.5 H:23.5	L:13.5 M:73 H:13.5	p1:29.6 p2:29.6 p3:40.3 p4:0.27 p5:0.27	s1:25 s2:35 s3:20 s4:20
12	M:100	H:100	M:100	M:60 H:40	L:36 M:40.5 H:23.5	L:7.83 M:48.3 H:43.8	p1:14 p2:14 p3:71.9 p4:0 p5:0	s1:25 s2:35 s3:20 s4:20
13	H:100	L:100	H:100	M:80 H:20	L:12 M:26 H:62	L:4 M:30 H:66	p1:30.5 p2:47.5 p3:22 p4:0 p5:0	s1:0 s2:20 s3:80 s4:0
14	H:100	M:100	H:100	M:60 H:40	L:12 M:26 H:62	L:8.67 M:82.7 H:8.67	p1:21.2 p2:76 p3:2.89 p4:0 p5:0	s1:0 s2:20 s3:80 s4:0
15	H:100	H:100	H:100	H:100	L:12 M:26 H:62	L:20.7 M:46.7 H:32.7	p1:17.6 p2:71.5 p3:10.9 p4:0 p5:0	s1:0 s2:20 s3:80 s4:0

Rozpatrzmy dwa przykłady.

Przykład 1

Przykład dotyczy wiersza nr 1 w tab. 12.1. Ustalono wartości stwierdzeń wejściowych jako:

moc MOC {L=100,M=0,H=0}

obrotyPWy PRĘDKOŚĆ KĄTOWA NA WYJŚCIU PRZEKŁADNI {L=100,M=0,H=0}

odl ODLEGŁOŚĆ POMIĘDZY OSIĄ WAŁKA SILNIKA A OSIĄ WAŁKA WYJŚCIOWEGO PRZEKŁADNI {L=100,M=0,H=0}

Stwierdzenia wyjściowe (obserwowane) uzyskały następujące wartości:

momentWy MOMENT NA WYJŚCIU PRZEKŁADNI {L=100,M=0,H=0}

obrotySWy PRĘDKOŚĆ KĄTOWA NA WYJŚCIU SILNIKA {L=46.7,M=43.3,H=10}

i PRZEŁOŻENIE {L=15.6,M=58.9,H=25.6}

przekładnia PRZEKŁADNIA {p1:22.3, p2:22.3, p3:34.9, p4:10.2, p5:10.2}

silnik SILNIK {s1:33.3, s2:33.3, s3:0.0, s4:33.3}

Moc przekładni wiąże się z wyborem rodzaju silnika. W tym przykładzie sieć wskazała trzy rodzaje silników jakie mogą zostać użyte, czyli s1: 1-fazowy, s2: prądu stałego, s4: krokowy. Rodzaj silnika ma wpływ na stwierdzenie „obrotySWy: PRĘDKOŚĆ KĄTOWA NA WYJŚCIU SILNIKA”. Uzyskane wyniki wskazują, że prędkość kątowa na wyjściu silnika jest niska (L:46.7) lub wysoka (H:43.3). Prędkość kątowa na wyjściu przekładni i prędkość kątowa na wejściu przekładni (prędkość kątowa na wyjściu silnika) mają wpływ na przełożenie „i: PRZEŁOŻENIE”.

Uzyskane wyniki wskazują, że przełożenie jest na poziomie średnim (M:58.9) lub, w mniejszym stopniu, na poziomie wysokim (H=25.6). Ostatecznie, otrzymane wartości stwierdzeń, a w tym ustalona wartość stwierdzenia „odl: ODLEGŁOŚĆ POMIĘDZY OSIĄ WAŁKA SILNIKA A OSIĄ WAŁKA WYJŚCIOWEGO PRZEKŁADNI”, wskazują, że najlepszą przekładnią będzie przekładnia p3: przekładnia zębata (wartość 34.9) lub p1: przekładnia cięgnowa z pasami klinowymi (wartość 22.3), lub p2: przekładnia cięgnowa z pasami synchronicznymi (wartość 22.3).

Otrzymane wyniki wskazują, że zbudowana sieć działa prawidłowo.

Przykład 2

Przykład dotyczy wiersza nr 14 w tab. 12.2. Ustalono wartości stwierdzeń wejściowych jako:

moc MOC {L=0,M=0,H=100}

obrotyPWy PRĘDKOŚĆ KĄTOWA NA WYJŚCIU PRZEKŁADNI {L=0,M=100,H=0}

odl ODLEGŁOŚĆ POMIĘDZY OSIĄ WAŁKA SILNIKA A OSIĄ WAŁKA WYJŚCIOWEGO PRZEKŁADNI {L=0,M=0,H=100}

Stwierdzenia wyjściowe (obserwowane) uzyskały następujące wartości:

momentWy MOMENT NA WYJŚCIU PRZEKŁADNI {L=0,M=60,H=40}

obrotySWy PRĘDKOŚĆ KĄTOWA NA WYJŚCIU SILNIKA {L=12,M=26,H=62}

i PRZEŁOŻENIE {L=8.67,M=82.7,H=8.67}

przekładnia PRZEKŁADNIA {p1:21.2, p2:76, p3:2.89, p4:0, p5:0}

silnik SILNIK {s1:0, s2:20, s3:80.0, s4:0}

Dla ustalonych wartości stwierdzeń wejściowych, sieć wskazała dwa rodzaje silników jakie mogą zostać użyte, czyli s2: prądu stałego lub s3: asynchroniczny 3-fazowy. Rodzaj silnika ma wpływ na stwierdzenie „obrotySWy: PRĘDKOŚĆ KĄTOWA NA WYJŚCIU SILNIKA”. Uzyskane wyniki określają, że prędkość kątowa na wyjściu silnika jest wysoka (H:62) lub, w mniejszym stopniu, średnia (M:26). Prędkość kątowa na wyjściu przekładni i prędkość kątowa na wejściu przekładni (prędkość kątowa na wyjściu silnika) mają wpływ na przełożenie „i: PRZEŁOŻENIE”. Uzyskane wyniki wskazują, że przełożenie jest na poziomie zdecydowanie średnim (M:82.7). Ostatecznie, otrzymane wartości stwierdzeń, a w tym ustalona wartość stwierdzenia „odl: ODLEGŁOŚĆ POMIĘDZY OSIĄ WAŁKA SILNIKA A OSIĄ WAŁKA WYJŚCIOWEGO PRZEKŁADNI”, wskazują, że najlepszą przekładnią będzie przekładnia p2: przekładnia cięgnowa z pasami synchronicznymi (wartość 76) lub, w mniejszym stopniu, przekładnia p1: przekładnia cięgnowa z pasami klinowymi (wartość 21.2).

Otrzymane wyniki wskazują, że zbudowana sieć działa prawidłowo.

12.4.2. Dobór elementów napędu ze względu na precyzyjne sterowanie położeniem kątowym

W tab. 12.3 i w tab. 12.4 pokazano wyniki testów jakie przeprowadzono dla sieci opracowanej dla podejścia 2. Rozpatrzmy cztery przykłady.

Tab. 12.3: Lista wartości wybranych stwierdzeń dla sieci utworzonej dla podejścia 2. (opis w tekście)

Lp.	Stwierdzenia wejściowe (niezależne)			Wybrane stwierdzenia wyjściowe (obserwowane)				
	rpol (regulacja położenia)	obrotyPWy	odl	momentWy	przekładnia	moc	silnik	sterownik
1	Y:100	L:100	L:100	L:26.7 M:65.5 H:7.8	p1:13.6 p2:15.3 p3:36.4 p4:17.4 p5:17.4	L:12 M:49 H:39	s1:30 s2:30 s3:30 s4:10	st1:30 st2:60 st3:10 st4:0
2	Y:100	L:100	M:100	L:26.7 M:65.5 H:7.8	p1:19.7 p2:21.4 p3:54.9 p4:2.03 p5:2.03	L:12 M:49 H:39	s1:30 s2:30 s3:30 s4:10	st1:30 st2:60 st3:10 st4:0
3	Y:100	L:100	H:100	L:26.7 M:65.5 H:7.8	p1:37.2 p2:45.6 p3:17.3 p4:0 p5:0	L:12 M:49 H:39	s1:30 s2:30 s3:30 s4:10	st1:30 st2:60 st3:10 st4:0
4	Y:100	M:100	L:100	L:3.6 M:80.8 H:15.6	p1:11.8 p2:13.2 p3:70.0 p4:2.47 p5:2.47	L:12 M:49 H:39	s1:30 s2:30 s3:30 s4:10	st1:30 st2:60 st3:10 st4:0
5	Y:100	M:100	M:100	L:3.6 M:80.8 H:15.6	p1:21.9 p2:25.4 p3:52.1 p4:0.27 p5:0.27	L:12 M:49 H:39	s1:30 s2:30 s3:30 s4:10	st1:30 st2:60 st3:10 st4:0
6	Y:100	M:100	H:100	L:3.6 M:80.8 H:15.6	p1:33.8 p2:54.5 p3:11.7 p4:0 p5:0	L:12 M:49 H:39	s1:30 s2:30 s3:30 s4:10	st1:30 st2:60 st3:10 st4:0
7	Y:100	H:100	L:100	L:0 M:39.0 H:61.0	p1:3.89 p2:3.89 p3:92.2 p4:0 p5:0	L:12 M:49 H:39	s1:30 s2:30 s3:30 s4:10	st1:30 st2:60 st3:10 st4:0
8	Y:100	H:100	M:100	L:0 M:39.0 H:61.0	p1:8.56 p2:8.56 p3:82.9 p4:0 p5:0	L:12 M:49 H:39	s1:30 s2:30 s3:30 s4:10	st1:30 st2:60 st3:10 st4:0
9	Y:100	H:100	H:100	L:0 M:39.0 H:61.0	p1:29.5 p2:51.0 p3:19.5 p4:0 p5:0	L:12 M:49 H:39	s1:30 s2:30 s3:30 s4:10	st1:30 st2:60 st3:10 st4:0

Tab. 12.4: Lista wartości wybranych stwierdzeń dla sieci utworzonej dla podejścia 2. (opis w tekście, ciąg dalszy tab. 12.3)

Lp.	Stwierdzenia wejściowe (niezależne)			Wybrane stwierdzenia wyjściowe (obserwowane)				
	rpol (regulacja położenia)	obrotyPWy	odl	momentWy	przekładnia	moc	silnik	sterownik
10	N:100	L:100	L:100	L:34.3 M:59.2 H:6.5	p1:15.0 p2:16.4 p3:36.1 p4:16.3 p5:16.3	L:20.0 M:47.5 H:32.5	s1:25 s2:25 s3:25 s4:25	st1:25 st2:40 st3:25 st4:10
11	N:100	M:100	L:100	L:6 M:81 H:13	p1:13.0 p2:14.2 p3:68.4 p4:2.17 p5:2.17	L:20.0 M:47.5 H:32.5	s1:25 s2:25 s3:25 s4:25	st1:25 st2:40 st3:25 st4:10
12	N:100	H:100	L:100	L:34.3 M:59.2 H:6.5	p1:3.58 p2:3.58 p3:92.8 p4:0 p5:0	L:20.0 M:47.5 H:32.5	s1:25 s2:25 s3:25 s4:25	st1:25 st2:40 st3:25 st4:10

Przykład 1

Rozpatrując zmianę wartości stwierdzenia „rpol: REGULACJA POŁOŻENIA” z Y:100 na N:100, niezależnie od wartości stwierdzeń „obrotyPWy: PRĘDKOŚĆ KĄTOWA NA WYJŚCIU PRZEKŁADNI” i „odl: ODLEGŁOŚĆ POMIĘDZY OSIĄ WAŁKA SILNIKA A OSIĄ WAŁKA WYJŚCIOWEGO PRZEKŁADNI”, widać że zmianie ulegają wartości stwierdzeń:

silnik SILNIK

sterownik STEROWNIK (który jest stwierdzeniem zależnym m.in. od stwierdzenia SILNIK)

moc MOC (który jest stwierdzeniem zależnym od stwierdzenia SILNIK)

Powodem jest to, że wprowadzenie regulacji położenia kąтового wałka silnika, wymusza dobór odpowiedniego rodzaju silnika oraz dodatkowego osprzętu w postaci dodatkowych czujników, regulatora i sterownika. Dany rodzaj silnika wpływa na zakres dostępnych mocy.

Przykład 2

Rozpatrując zmianę wartości stwierdzenia „obrotyPWy: PRĘDKOŚĆ KĄTOWA NA WYJŚCIU PRZEKŁADNI” na jedną z wartości L:100, M:100 lub H:100, niezależnie od wartości stwierdzeń „rpol: REGULACJA POŁOŻENIA” i „odl: ODLEGŁOŚĆ POMIĘDZY OSIĄ WAŁKA SILNIKA A OSIĄ WAŁKA WYJŚCIOWEGO PRZEKŁADNI”, widać że zmianie ulega wartość stwierdzenia „momentWy: MOMENT NA WYJŚCIU PRZEKŁADNI”, które jest stwierdzeniem zależnym m.in. od stwierdzenia „obrotyPWy: PRĘDKOŚĆ KĄTOWA NA WYJŚCIU PRZEKŁADNI”. Powodem jest to, że zmiana wartości momentu obrotowego na wyjściu przekładni jest wprost proporcjonalna do zmiany prędkości kątovej przekładni na wyjściu przy założeniu stałej wartości mocy.

Przykład 3

Dla danych przedstawionych w tabelach 12.3 i 12.4, widać że wybór rodzaju przekładni jest determinowany przede wszystkim wartościami stwierdzenia „odl: ODLEGŁOŚĆ POMIĘDZY OSIĄ WAŁKA SILNIKA A OSIĄ WAŁKA WYJŚCIOWEGO PRZEKŁADNI”. Inne stwierdzenia, które mają bezpośredni wpływ na wybór rodzaju przekładni, to:

momentWy MOMENT NA WYJŚCIU PRZEKŁADNI

obrotyPWy PRĘDKOŚĆ KĄTOWA NA WYJŚCIU PRZEKŁADNI

moc MOC

Przy czym stwierdzenie „moc: MOC” jest zależne od stwierdzeń „silnik: SILNIK”, „sterownik: STEROWNIK”, które z kolei są zależne od stwierdzenia „rpol: REGULACJA POŁOŻENIA”.

Przykład 4

W tabeli 12.5 pokazano zmiany wartości stwierdzeń sieci w przypadku zmian wartości stwierdzenia „rpol: REGULACJA POŁOŻENIA” z Y:100 na N:100.

Tab. 12.5: Lista wartości wybranych stwierdzeń dla sieci utworzonej dla podejścia 2. w przypadku zmian wartości stwierdzenia „rpol: REGULACJA POŁOŻENIA” z Y:100 na N:100

Lp.	Stwierdzenia wejściowe (niezależne)	Wybrane stwierdzenia wyjściowe (obserwowane)				
	rpol (regulacja położenia)	czujnikiPWy	czujnikiSWy	regulator	silnik	sterownik
1	Y:100	cpw1:100 cpw2:0 cpw3:0	csw1:100 csw2:0 csw3:0	reg1:100 reg2:0 reg3:0	s1:30 s2:30 s3:30 s4:10	st1:30 st2:60 st3:10 st4:0
2	N:100	cpw1:0 cpw2:50 cpw3:50	csw1:0 csw2:50 csw3:50	reg1:0 reg2:50 reg3:50	s1:25 s2:25 s3:25 s4:25	st1:25 st2:40 st3:25 st4:10

Decyzja o wyborze możliwości regulacji położenia kąтового wałka wyjściowego silnika powoduje, że jako:

- czujnik na wałku wyjściowym przekładni (stwierdzenie „czujnikiPWy: CZUJNIKI NA WAŁKU WYJŚCIOWYM PRZEKŁADNI”) wybrany zostaje czujnik położenia (cpw1:100);
- czujnik na wałku wyjściowym silnika (stwierdzenie „czujnikiSWy: CZUJNIKI NA WAŁKU WYJŚCIOWYM SILNIKA”) wybrany zostaje czujnik położenia (csw1:100);
- regulator (stwierdzenie „regulator: REGULATOR”) wybrany zostaje regulator położenia (reg1:100));

- silniki (stwierdzenie „silnik: SILNIK”), które można zastosować w takim układzie wybrane zostają przede wszystkim s1: 1-fazowy (s1:30), s2: prądu stałego (s2:30), s3: asynchroniczny 3-fazowy (s3:30);
- sterowniki (stwierdzenie „sterownik: STEROWNIK”), zostają wybrane st2: falownik (st2:60) i st1: sterownik silnika prądu stałego (st1:30).

Należy wyjaśnić dlaczego w tym zestawieniu nie został wybrany silnik krokowy (s4:10), a co za tym idzie również nie został wybrany sterownik silnika krokowego (st3:10). Powodem jest to, że wybrane zostały czujniki położenia, które nie są wymagane w przypadku stosowania silników krokowych.

Otrzymane wyniki wskazują, że zbudowana sieć działa prawidłowo.

12.4.3. Dobór elementów napędu ze względu na precyzyjne sterowanie prędkością kątową

W tab. 12.6 i w tab. 12.7 pokazano wyniki testów jakie przeprowadzono dla sieci opracowanej dla podejścia 3. Rozpatrzmy cztery przykłady. Należy zaznaczyć, że sytuacja jest podobna do opisanej w przykładach w punkcie dotyczącym podejścia 2.

Przykład 1

Rozpatrując zmianę wartości stwierdzenia „rpred: REGULACJA PRĘDKOŚCI KĄTOWEJ” z Y:100 na N:100, niezależnie od wartości stwierdzeń „obrotyPWy: PRĘDKOŚĆ KĄTOWA NA WYJŚCIU PRZEKŁADNI” i „odl: ODLEGŁOŚĆ POMIĘDZY OSIĄ WAŁKA SILNIKA A OSIĄ WAŁKA WYJŚCIOWEGO PRZEKŁADNI”, widać że zmianie ulegają wartości stwierdzeń:

silnik SILNIK

sterownik STEROWNIK (który jest stwierdzeniem zależnym m.in. od stwierdzenia SILNIK)

moc MOC (który jest stwierdzeniem zależnym od stwierdzenia SILNIK)

Powodem jest to, że wprowadzenie regulacji prędkości kątowej, wymusza dobór odpowiedniego rodzaju silnika oraz dodatkowego osprzętu w postaci dodatkowych czujników, regulatora i sterownika. Dany rodzaj silnika wpływa na zakres dostępnych mocy.

Przykład 2

Rozpatrując zmianę wartości stwierdzenia „obrotyPWy: PRĘDKOŚĆ KĄTOWA NA WYJŚCIU PRZEKŁADNI” na jedną z wartości L:100, M:100 lub H:100, niezależnie od wartości stwierdzeń „rpred: REGULACJA PRĘDKOŚCI KĄTOWEJ” i „odl: ODLEGŁOŚĆ POMIĘDZY OSIĄ WAŁKA SILNIKA A OSIĄ WAŁKA WYJŚCIOWEGO PRZEKŁADNI”, widać że zmianie ulega wartość stwierdzenia „momentWy: MOMENT NA WYJŚCIU PRZEKŁADNI”, które jest stwierdzeniem zależnym m.in. od stwierdzenia „obrotyPWy: PRĘDKOŚĆ KĄTOWA NA WYJŚCIU PRZEKŁADNI”. Powodem jest to, że zmiana wartości momentu obrotowego na wyjściu przekładni jest wprost proporcjonalna do zmiany prędkości kątowej przekładni na wyjściu przy założeniu stałej wartości mocy.

Tab. 12.6: Lista wartości wybranych stwierżeń dla sieci utworzonej dla podejścia 3. (opis w tekście)

Lp.	Stwierzenia wejściowe (niezależne)			Wybrane stwierzenia wyjściowe (obserwowane)				
	rpmred (regulacja prędkości)	obrotWy	odl	momentWy	przekładnia	moc	silnik	sterownik
1	Y:100	L:100	L:100	L:26.7 M:65.5 H:7.8	p1:13.6 p2:15.3 p3:36.4 p4:17.4 p5:17.4	L:12 M:49 H:39	s1:30 s2:30 s3:30 s4:10	st1:30 st2:60 st3:10 st4:0
2	Y:100	L:100	M:100	L:26.7 M:65.5 H:7.8	p1:19.7 p2:21.4 p3:54.9 p4:2.03 p5:2.03	L:12 M:49 H:39	s1:30 s2:30 s3:30 s4:10	st1:30 st2:60 st3:10 st4:0
3	Y:100	L:100	H:100	L:26.7 M:65.5 H:7.8	p1:37.2 p2:45.6 p3:17.3 p4:0 p5:0	L:12 M:49 H:39	s1:30 s2:30 s3:30 s4:10	st1:30 st2:60 st3:10 st4:0
4	Y:100	M:100	L:100	L:3.6 M:80.8 H:15.6	p1:11.8 p2:13.2 p3:70.0 p4:2.47 p5:2.47	L:12 M:49 H:39	s1:30 s2:30 s3:30 s4:10	st1:30 st2:60 st3:10 st4:0
5	Y:100	M:100	M:100	L:3.6 M:80.8 H:15.6	p1:21.9 p2:25.4 p3:52.1 p4:0.27 p5:0.27	L:12 M:49 H:39	s1:30 s2:30 s3:30 s4:10	st1:30 st2:60 st3:10 st4:0
6	Y:100	M:100	H:100	L:3.6 M:80.8 H:15.6	p1:33.8 p2:54.5 p3:11.7 p4:0 p5:0	L:12 M:49 H:39	s1:30 s2:30 s3:30 s4:10	st1:30 st2:60 st3:10 st4:0
7	Y:100	H:100	L:100	L:0 M:39.0 H:61.0	p1:3.89 p2:3.89 p3:92.2 p4:0 p5:0	L:12 M:49 H:39	s1:30 s2:30 s3:30 s4:10	st1:30 st2:60 st3:10 st4:0
8	Y:100	H:100	M:100	L:0 M:39.0 H:61.0	p1:8.56 p2:8.56 p3:82.9 p4:0 p5:0	L:12 M:49 H:39	s1:30 s2:30 s3:30 s4:10	st1:30 st2:60 st3:10 st4:0
9	Y:100	H:100	H:100	L:0 M:39.0 H:61.0	p1:29.5 p2:51.0 p3:19.5 p4:0 p5:0	L:12 M:49 H:39	s1:30 s2:30 s3:30 s4:10	st1:30 st2:60 st3:10 st4:0

Tab. 12.7: Lista wartości wybranych stwierdzeń dla sieci utworzonej dla podejścia 3. (opis w tekście, ciąg dalszy tab. 12.6)

Lp.	Stwierdzenia wejściowe (niezależne)			Wybrane stwierdzenia wyjściowe (obserwowane)				
	rpred (regulacja prędkości)	obrotyPWy	odl	momentWy	przekładnia	moc	silnik	sterownik
10	N:100	L:100	L:100	L:34.3 M:59.2 H:6.5	p1:15.0 p2:16.4 p3:36.1 p4:16.3 p5:16.3	L:20.0 M:47.5 H:32.5	s1:25 s2:25 s3:25 s4:25	st1:25 st2:40 st3:25 st4:10
11	N:100	M:100	L:100	L:6 M:81 H:13	p1:13.0 p2:14.2 p3:68.4 p4:2.17 p5:2.17	L:20.0 M:47.5 H:32.5	s1:25 s2:25 s3:25 s4:25	st1:25 st2:40 st3:25 st4:10
12	N:100	H:100	L:100	L:34.3 M:59.2 H:6.5	p1:3.58 p2:3.58 p3:92.8 p4:0 p5:0	L:20.0 M:47.5 H:32.5	s1:25 s2:25 s3:25 s4:25	st1:25 st2:40 st3:25 st4:10

Przykład 3

Dla danych przedstawionych w tabelach 12.6 i 12.7, widać że wybór rodzaju przekładni jest determinowany przede wszystkim wartościami stwierdzenia „odl: ODLEGŁOŚĆ POMIĘDZY OSIĄ WAŁKA SILNIKA A OSIĄ WAŁKA WYJŚCIOWEGO PRZEKŁADNI”. Inne stwierdzenia, które mają bezpośredni wpływ na wybór rodzaju przekładni, to:

momentWy MOMENT NA WYJŚCIU PRZEKŁADNI

obrotyPWy PRĘDKOŚĆ KĄTOWA NA WYJŚCIU PRZEKŁADNI

moc MOC

Przy czym stwierdzenie „moc: MOC” jest zależne od stwierdzeń „silnik: SILNIK”, „sterownik: STEROWNIK”, które z kolei są zależne od stwierdzenia „rpred: REGULACJA PRĘDKOŚCI KĄTOWEJ”.

Przykład 4

W tabeli 12.8 pokazano zmiany wartości stwierdzeń sieci w przypadku zmian wartości stwierdzenia „rpred: REGULACJA PRĘDKOŚCI KĄTOWEJ” z Y:100 na N:100.

Tab. 12.8: Lista wartości wybranych stwierdzeń dla sieci utworzonej dla podejścia 3. w przypadku zmian wartości stwierdzenia „rpred: REGULACJA PRĘDKOŚCI KĄTOWEJ” z Y:100 na N:100

	Stwierdzenia wejściowe (niezależne)	Wybrane stwierdzenia wyjściowe (obserwowane)				
Lp.	rpred (regulacja prędkości)	czujnikPWy	czujnikSWy	regulator	silnik	sterownik
1	Y:100	cpw1:0 cpw2:100 cpw3:0	csw1:0 csw2:100 csw3:0	reg1:0 reg2:100 reg3:0	s1:30 s2:30 s3:30 s4:10	st1:30 st2:60 st3:10 st4:0
2	N:100	cpw1:50 cpw2:0 cpw3:50	csw1:50 csw2:0 csw3:50	reg1:50 reg2:0 reg3:50	s1:25 s2:25 s3:25 s4:25	st1:25 st2:40 st3:25 st4:10

Decyzja o wyborze możliwości regulacji prędkości kątowej powoduje, że jako:

- czujnik na wałku wyjściowym przekładni (stwierdzenie „czujnikiPWy: CZUJNIKI NA WAŁKU WYJŚCIOWYM PRZEKŁADNI”) wybrany zostaje czujnik prędkości kątowej (cpw2:100);
- czujnik na wałku wyjściowym silnika (stwierdzenie „czujnikiSWy: CZUJNIKI NA WAŁKU WYJŚCIOWYM SILNIKA”) wybrany zostaje czujnik prędkości kątowej (csw2:100);
- regulator (stwierdzenie „regulator: REGULATOR”) wybrany zostaje regulator prędkości obrotowej (reg2:100);
- silniki (stwierdzenie „silnik: SILNIK”), które można zastosować w takim układzie wybrane zostają przede wszystkim s1: 1-fazowy (s1:30), s2: prądu stałego (s2:30), s3: asynchroniczny 3-fazowy (s3:30);
- sterowniki (stwierdzenie „sterownik: STEROWNIK”), zostają wybrane st2: falownik (st2:60) i st1: sterownik silnika prądu stałego (st1:30).

Podobnie jak w przykładzie 4. podejścia 2. także i tutaj nie został wybrany silnik krokowy (s4:10), a co za tym idzie również nie został wybrany sterownik silnika krokowego (st3:10). Powodem jest to, że wybrane zostały czujniki prędkości kątowej, które nie są wymagane w przypadku stosowania silników krokowych.

Otrzymane wyniki wskazują, że zbudowana sieć działa prawidłowo.

12.4.4. Dobór elementów napędu ze względu na bezpieczeństwo użytkownika

W tab. 12.9 pokazano wyniki testów jakie przeprowadzono dla sieci opracowanej dla podejścia 4. Zmiany wartości wybranych stwierdzeń w przypadku zmian wartości stwierdzenia „bezp: BEZPIECZEŃSTWO UŻYTKOWANIA” z Y:100 na N:100 pokazują, że priorytet bezpieczeństwa ma wpływ na wiele elementów układu napędowego. Zmniejszona zostaje moc, a co za tym idzie prędkość kątowa i moment obrotowy na wyjściu przekładni. Zaleca się stosowanie przekładni zabudowanych o zwartej budowie (o małych odległościach pomiędzy osią wałka silnika a osią wałka wyjściowego przekładni mechanicznej). Dodatkowo zaleca się użycie sprzęgieł zabezpieczających (np. przeciążeniowych) oraz dodatkowych czujników przeciążeniowych i przeciwprzepięciowych wbudowanych w silnik elektryczny.

Otrzymane wyniki wskazują, że zbudowana sieć działa prawidłowo. W dalszej części rozdziału pokazano przykładowe wyniki wybranych stwierdzeń (rys. 12.8) oraz jak wprowadzono wartości dla stwierdzenia „bezp: BEZPIECZEŃSTWO UŻYTKOWANIA” (rys. 12.9).

Tab. 12.9: Lista wartości wybranych stwierdzeń dla sieci utworzonej dla podejścia 4. w przypadku zmian wartości stwierdzenia „bezp: BEZPIECZEŃSTWO UŻYTKOWANIA” z Y:100 na N:100

Stwierdzenia wejściowe (niezależne)		Wybrane stwierdzenia wyjściowe (obserwowane)										
Lp.	bezp (bezpieczeństwo)	moc	momentWy	obrotyPWy	obrotySWy	odl (odległość)	czujnikS	sterownik	obudowa	przekładnia	silnik	sprzeglo
1	Y:100	L:54.3 M:44.0 H:1.75	L:50 M:40 H:10	L:65 M:35 H:0	L:37.0 M:49.7 H:13.2	L:80 M:20 H:0	cs1:50 cs2:50 cs3:0	st1:32.4 st2:41.7 st3:17.6 st4:8.33	o1:50 o2:50 o3:0	p1:23.0 p2:23.1 p3:43.8 p4:4.03 p5:6.02	s1:50 s2:32.4 s3:0 s4:17.6	sp1:0 sp2:0 sp3:100
2	N:100	L:8.4 M:57.2 H:34.4	L:36.7 M:34.7 H:28.7	L:2 M:36 H:62	L:15.6 M:36.7 H:47.7	L:0 M:20 H:80	cs1:0 cs2:0 cs3:100	st1:6.18 st2:75.0 st3:3.82 st4:15.0	o1:18 o2:18 o3:64	p1:31.9 p2:47.0 p3:21.0 p4:0.03 p5:0.03	s1:37.2 s2:6.18 s3:52.8 s4:3.82	sp1:40 sp2:40 sp3:20

12.5. Podsumowanie

Podzespoły i elementy składowe układu napędowego można dobrać na różne sposoby. W rozdziale pokazano cztery sposoby uwzględniające różne parametry wejściowe. Rozpatrywano podstawową metodę, gdzie na wejściu określano wartość mocy układu napędowego, wartość prędkości kątowej na wyjściu przekładni oraz odległość pomiędzy osią wałka silnika a osią wałka wyjściowego przekładni mechanicznej, oraz metody użyteczne podczas projektowania układu napędowego przeznaczonego do sterowania położeniem kątowym wałka wyjściowego przekładni oraz precyzyjnym sterowaniem prędkością kątową na wyjściu przekładni. Dodatkowo uwzględniono metodę pozwalającą dobrać podzespoły i elementy układu napędowego ze względu na bezpieczeństwo użytkownika całego układu.

Każda z sieci stwierdzeń dysponowała takim samym zestawem stwierdzeń. Różnice pojawiły się w strukturze połączeń pomiędzy stwierdzeniami, co spowodowało różnice w postaci tablic prawdopodobieństw warunkowych. Jedna sieć odpowiadała jednej metodzie doboru podzespołów i elementów składowych układu napędowego. Uzyskane wyniki pozwalają wywnioskować, że sieci stwierdzeń zostały zbudowane prawidłowo, a otrzymywane wartości były zgodne z oczekiwanymi.

The image shows three overlapping windows of the MMNET GUI, each displaying calculation results for a different node. The windows are titled 'MMNET wersja 0.5.6'.

Window 1: Node 40 [obrotowy]

Model: Model [id= NA] Liczba warstw: 4 Liczba węzłów: 72
Tezaurus: Thesaurus ... [id = 0], Write notice, please

Obliczenia: Oblicz sieć dla domyślnych parametrów, Wprowadź dane do węzłów, Identyfikator warstwy: --Wybór identyfikatora--, Waga warstwy, Zapisz wagę

Wyniki obliczeń: Identyfikator węzła: 40 [obrotowy]

Row.names	id_warstwy	id_węzła	status	L	M	H	waga_warstwy
2	1	1	2	0.3795711	0.3363899	0.2840390	1
3	2	1	2	0.3804935	0.3360166	0.2834899	1
4	3	1	2	0.3804935	0.3360166	0.2834899	1
5	4	5	2	0.0200000	0.3600000	0.6200000	1

Window 2: Node 330 [od]

Wyniki obliczeń: Identyfikator węzła: 330 [od]

Row.names	id_warstwy	id_węzła	status	L	M	H	waga_warstwy
2	1	8	2	0.2583190	0.3154377	0.4262433	1
3	2	11	2	0.2566026	0.3162772	0.4271202	1
4	3	11	2	0.2566026	0.3162772	0.4271202	1
5	4	7	2	0.0000000	0.2000000	0.8000000	1

Window 3: Node 130 [silnik]

Wyniki obliczeń: Identyfikator węzła: 130 [silnik]

Row.names	id_warstwy	id_węzła	status	s1	s2	s3	s4	waga_warstwy
2	1	3	2	0.1982632	0.2902999	0.3307096	0.1807273	1
3	2	5	2	0.2770865	0.2729145	0.2723312	0.1776679	1
4	3	5	2	0.2770865	0.2729145	0.2723312	0.1776679	1
5	4	10	2	0.3716800	0.0617600	0.5283200	0.0382400	1

Rys. 12.8: Pakiet MMNET. Wyniki obliczeń dla wybranych stwierdzeń dla przypadku „bezp: BEZPIECZEŃSTWO UŻYTKOWNIA” N:100.

Ustawianie wartości węzłów

Ustawienie wartości poszczególnych węzłów

<input type="checkbox"/>	40 [obrotyPWy]			?
<input type="checkbox"/>	80 [moc]			
<input type="checkbox"/>	130 [silnik]			
<input type="checkbox"/>	170 [obrotySWy]			
<input type="checkbox"/>	210 [i]			
<input type="checkbox"/>	250 [momentWy]			
<input type="checkbox"/>	290 [sprzęgło]			
<input type="checkbox"/>	330 [odl]			
<input type="checkbox"/>	390 [przekładnia]			
<input type="checkbox"/>	430 [obudowa]			
<input type="checkbox"/>	470 [czujnikS]			
<input type="checkbox"/>	510 [zasilanie]			
<input checked="" type="checkbox"/>	520 [bezp]	No	1	
<input type="checkbox"/>	530 [smarowanie]			
<input type="checkbox"/>	570 [regulator]			
<input type="checkbox"/>	620 [sterownik]			
<input type="checkbox"/>	660 [czujnikSWy]			
<input type="checkbox"/>	700 [czujnikPWy]			
<input type="checkbox"/>	710 [rpol]			
<input type="checkbox"/>	720 [rpred]			

Oblicz

Rys. 12.9: Pakiet MMNET. Ustalanie wartości stwierdzenia „bezp: BEZPIECZEŃSTWO UŻYTKOWANIA”.

Bibliografia

- [12.1] B. Branowski (red.). *Podstawy konstrukcji napędów maszyn*. Wydawnictwo Politechniki Poznańskiej, Poznań 2007.
- [12.2] Kosmol J. *Serwonapędy obrabiarek sterowanych numerycznie*. Wydawnictwa Naukowo-Techniczne WNT, Warszawa 1998.
- [12.3] (red.) E. Mazanek. *Przykłady obliczeń z podstaw konstrukcji maszyn Tom 2. Łoży-ska, sprzęgła i hamulce, przekładnie mechaniczne*. Wydawnictwa Naukowo-Techniczne WNT, Warszawa 2008.

Rozdział 13

Zastosowanie wieloskalowej sieci stwierdzeń w diagnostycznym systemie doradczym

Tomasz ROGALA

W niniejszym rozdziale opisano przykład sieci stwierdzeń wykonanej w dwóch wariantach jako sieć przekonań oraz ogólnej sieci stwierdzeń które są przykładem fragmentu modelu diagnostycznego wybranego podsystemu pojazdu samochodowego. Z uwagi na obszerny zakres takiego zagadnienia pokazano jedynie fragment modelu dotyczący zagadnień związanych z układem rozruchowym silnika. Rozpatrywany przykład sieci stwierdzeń dotyczy szczególnego przypadku sieci wielomodalnej jakim jest sieć wieloskalowa. Definicja sieci i jej szczegółowy opis został podany w [13.3].

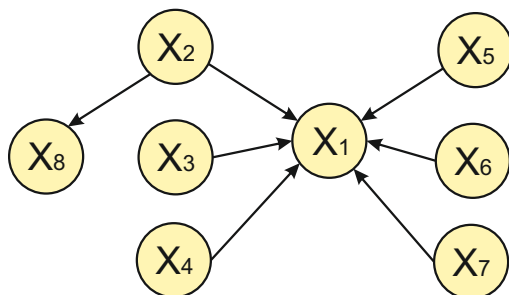
13.1. Przykład wieloskalowego modelu diagnostycznego

Zgodnie z przyjętą w raporcie [13.3] definicją, model wieloskalowy to model, w którym w poszczególne warstwy zawierają sieci opisujące rozpatrywaną dziedzinę wiedzy w różnym stopniu szczegółowości, przy czym co najmniej jedna z warstw powinna obejmować w stopniu wystarczająco ogólnym inne warstwy modelu. Warstwa ogólna ujmuje w całość rozpatrywaną dziedzinę, natomiast pozostałe warstwy najczęściej opisują wybrane fragmenty lub podsystemy rozpatrywanej dziedziny. Należy zwrócić uwagę, iż modele takie mają istotne znaczenie w opisywaniu złożonych środków technicznych m.in. dla potrzeb konstruowania diagnostycznych systemów doradczych [13.1], w szczególności tych, które dotychczas były opisywane modelami globalnymi, cząstkowymi, hierarchicznymi itp. [13.7], [13.5], [13.4], [13.2], [13.6], [13.8].

13.1.1. Model globalny

Przedmiotem dalszych rozważań jest przykład sieci wieloskalowej przeznaczonej dla potrzeb diagnozowania układu rozruchowego i jego oprzyrządowania. Informacje na temat relacji diagnostycznych pozyskano z literatury dotyczącej budowy pojazdów samochodowych m.in. [13.9] W tym celu opracowany został ogólny model diagnostyczny zapisany w postaci sieci przekonań oraz ogólnej sieci stwierdzeń pierwszej warstwy modelu. Warstwę tą przedstawiono na rysunku 13.1. Model ten służy do diagnozowania stanu objawiającego się brakiem możliwości uruchomienia silnika mimo ustawienia włącznika zapłonu na pozycję *START*. Stwierdzenie to zapisane zostało w treści stwierdzenia oznaczonego symbolem X_1 . Poniżej przedstawiono znaczenie symboli wykorzystanych na rysunku 13.1. Ze względu na większą przejrzystość

dalszych rozważań, w opisach poszczególnych stwierdzeń opuszczono treść objaśnień dla tych stwierdzeń.



Rys. 13.1: Warstwa ogólna przykładu wieloskalowej sieci stwierdzeń

W rozpatrywanej sieci, węzły reprezentują stwierdzenia proste. Poszczególne krawędzie reprezentują, w przypadku sieci przekonań, asocjacje pomiędzy tymi stwierdzeniami, a w przypadku ogólnych sieci stwierdzeń, warunki dostateczne występujące pomiędzy nimi [13.3]. Poszczególne oznaczenia przyjęte na rysunku 13.1 reprezentują następujące stwierdzenia:

X_1 -START

- Treść stwierdzenia = "Nie można uruchomić silnika";

X_2 -Rozrusznik

- Treść stwierdzenia = "Rozrusznik jest uszkodzony lub niesprawny";

X_3 -Akumulator

- Treść stwierdzenia = "Akumulator jest rozładowany";

X_4 -Układ paliwowy

- Treść stwierdzenia = "Niesprawny lub zanieczyszczony układ doprowadzania paliwa";

X_5 -Rozrząd

- Treść stwierdzenia = "Niewłaściwe kąty ustawienia rozrządu";

X_6 -Włącznik zapłonu

- Treść stwierdzenia = "Uszkodzony włącznik zapłonu";

X_7 -Ciśnienie sprężania

- Treść stwierdzenia = "Za niskie ciśnienie sprężania";

X_8 -Przygasające światła

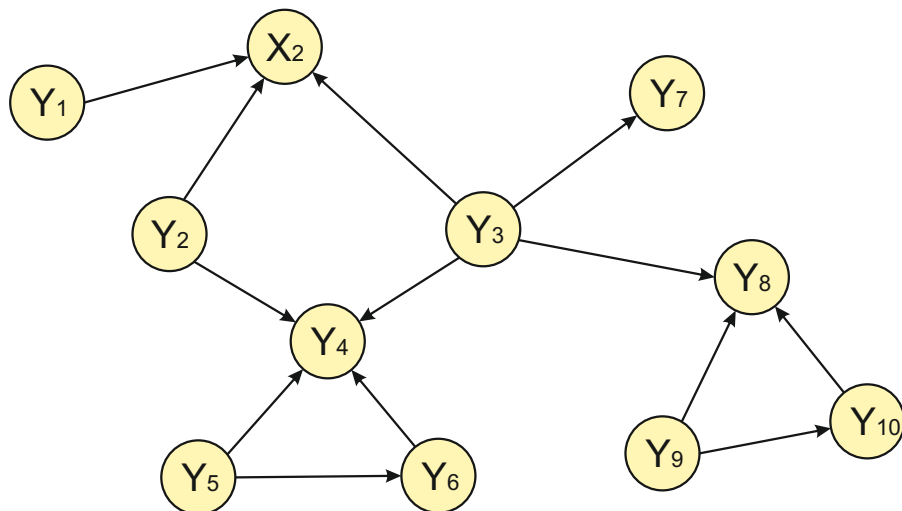
- Treść stwierdzenia = "Światła kontrolki tablicy rozdzielczej oraz światła mijania znacznie przygasają podczas próby uruchamiania silnika";

Dla każdego ze stwierdzeń oznaczonych symbolami $X_2, X_3, X_4, X_5, X_6, X_7$ możliwe jest opracowanie cząstkowego modelu diagnostycznego w postaci sieci przekonań lub ogólnej sieci stwierdzeń zapisanej w kolejnych warstwach modelu wieloskalowego. Ze względu na rozbudowany zakres rozpatrywanego zagadnienia i w celu zapewnienia większej przejrzystości dalszych rozważań, w kolejnych warstwach opisano jedynie wybrane sieci będące modelami diagnostycznymi stanu technicznego rozrusznika oraz akumulatora. Ich ogólny stan techniczny został

opisany stwierdzeniami oznaczonymi symbolami X_2 i X_3 . Należy zauważyć, że diagnozowanie stanu akumulatora X_3 może być realizowane na podstawie ogólnodostępnych symptomów takich jak ten, występujący w ogólnej strukturze sieci, wyrażony stwierdzeniem X_3 *światła kontrolki tablicy rozdzielczej oraz światła mijania znacznie przygasają podczas próby uruchamiania silnika*.

13.1.2. Model cząstkowy - diagnozowanie stanu rozrusznika

Na rysunku 13.2 pokazano strukturę sieci przekonań przedstawiającą model diagnozowania stanu rozrusznika. Dla celów wykonania drugiego wariantu sieci jako ogólnej sieci stwierżeń opracowano podobną strukturę sieci jak tą pokazaną na rysunku 13.2. Dla potrzeb wariantu ogólnej sieci stwierżeń usunięto z niej krawędzie pomiędzy węzłami Y_5 i Y_6 oraz Y_9 i Y_{10} ponieważ pomiędzy wskazanymi stwierdzeniami nie występuje warunek implikacji odpowiadający warunkowi dostatecznemu. W obu wariantach model opisuje m.in. możliwe przyczyny oraz symptomy związane z określonym stanem technicznym rozrusznika. Poszczególne oznaczenia przyjęte na rysunku 13.2 reprezentują następujące stwierdzenia:



Rys. 13.2: Warstwa szczegółowa przykładu wieloskalowej sieci stwierżeń - model diagnostyczny rozrusznika

Y_1 -Przerwa w przewodzie

- Treść stwierdzenia = "Przerwa w przewodzie lub w połączeniu z masą";

Y_2 -Ograniczony przepływ prądu

- Treść stwierdzenia = "Ograniczony przepływ prądu z powodu uszkodzonych lub skorodowanych połączeń";

Y_3 -Włacznik elektromagnetyczny

- Treść stwierdzenia = "Uszkodzony włącznik elektromagnetyczny";

Y_4 -Niskie obroty rozrusznika

- Treść stwierdzenia = "Rozrusznik obraca się za wolno i nie uruchamia silnika";

Y_5 -Szczotki

- Treść stwierdzenia = "Uszkodzone, zanieczyszczone lub zużyte szczotki";

Y_6 -Komutator

- Treść stwierdzenia = "Komutator jest zanieczyszczony, nadpalony, lub ma wyłobienia.";

Y_7 -Ciągła praca rozrusznika

- Treść stwierdzenia = "Rozrusznik pracuje po zwolnieniu wyłącznika zapłonu";

Y_8 -Rozprzęganie

- Treść stwierdzenia = "Mechanizm rozprzęgania zębika nie rozłącza się.";

Y_9 -Sprężyna zwrotna

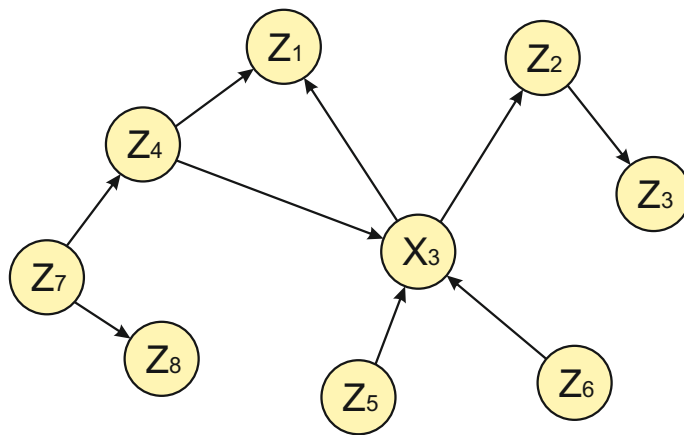
- Treść stwierdzenia = "Pęknięta sprężyna zwrotna";

Y_{10} -Wielowypust śrubowy

- Treść stwierdzenia = "Uszkodzony wielowypust śrubowy";

13.1.3. Model cząstkowy - diagnozowanie stanu akumulatora

Kolejna warstwa zawiera sieć reprezentującą model diagnozowania stanu akumulatora. Przedstawiona na rysunku 13.3 warstwa wykonana jako sieć przekonań zawiera stwierdzenia, których symbole i treść przedstawiono poniżej. Podobnie jak poprzednio, dla wariantu drugiego, reprezentowanego ogólną siecią stwierdzeń, opracowano strukturę sieci podobną do tej, która wykonana została dla sieci przekonań. Usunięto z niej krawędzie pomiędzy stwierdzeniami Z_4 i Z_1 ponieważ pomiędzy wymienionymi stwierdzeniami nie występuje warunek dostateczny.



Rys. 13.3: Warstwa szczegółowa wieloskalowej sieci stwierdzeń - model diagnostyczny akumulatora

Z_1 -Spadek napięcia

- Treść stwierdzenia = "Za duży spadek napięcia podczas uruchamiania";

Z_2 -Gęstość elektrolitu

- Treść stwierdzenia = "Za mała gęstość elektrolitu";

Z_3 -Wysoka impedancja

- Treść stwierdzenia = "Za wysoka impedancja w stosunku do sprawnego akumulatora";

Z_4 -Napięcie ładowania

- Treść stwierdzenia = "Za niskie napięcie ładowania";

Z_5 -Poziom elektrolitu

- Treść stwierdzenia = "Za niski poziom elektrolitu";

Z_6 -Zwarcie

- Treść stwierdzenia = "Zwarcie w instalacji elektrycznej";

Z_7 -Zużyty akumulator

- Treść stwierdzenia = "Akumulator jest zużyty";

Z_8 -Niska impedancja

- Treść stwierdzenia = "Za niska impedancja w stosunku do sprawnego akumulatora";

Przedstawiony model wieloskalowy może zawierać również dodatkowe warstwy reprezentujące pozostałe zagadnienia występujące w modelu globalnym, np. model diagnozowania stanu układu paliwowego.

Należy zwrócić uwagę, że w poszczególnych warstwach modelu zawierających zbiory stwierdzeń S_1, S_2, S_3 występują takie stwierdzenia, dla których

$$S_1 \cap S_2 = \{X_2\} \quad (13.1)$$

oraz

$$S_1 \cap S_3 = \{X_3\} \quad (13.2)$$

gdzie X_2, X_3 to symbole stwierdzeń, które występują równocześnie w różnych warstwach modelu.

13.1.4. Domyślny stan sieci przekonań

Dla opracowanej struktury sieci przekonań zdefiniowano wartości tablic prawdopodobieństw warunkowych. Wartości te dobrano na podstawie subiektywnych opinii autora modelu. Następnie dla rozpatrywanej sieci wieloskalowej wyznaczono wartości stwierdzeń dla domyślnego stanu sieci. Wyniki zostały przedstawione w tabeli 13.1. Zagregowane wartości stwierdzeń X_2 i X_3 wyznaczone zostały na podstawie wartości średniej ważonej. Poszczególne wagi średniej zostały ustalone na podstawie własnej subiektywnej opinii o przydatności poszczególnych warstw do diagnozowania poszczególnych elementów rozpatrywanego zadania. Przyjęto, że wagi dla warstw opisujących stan techniczny akumulatora oraz rozrusznika są jednakowe i równe 0.45, natomiast waga warstwy ogólnej wynosi 0.1. Zagregowane wartości stwierdzeń podano w ostatnim wierszu tabeli 13.1. W przypadku drugiego wariantu odpowiadającego ogólnej sieci stwierdzeń wyniki dla wszystkich występujących w sieci stwierdzeń odpowiadają przedziałom wartości $[0; 1]$.

Tab. 13.1: Wyniki dla domyślnych wartości stwierdzeń. Dla sieci przekonań (*SP*) podano wartości odpowiadające wystąpieniu stanu *tak*

Warstwa 1	SP	Warstwa 2	SP	Warstwa 3	SP
X_1	0.07	Y_1	0.1	Z_1	0.35
X_2	0.01	Y_2	0.05	Z_2	0.12
X_3	0.1	Y_3	0.1	Z_3	0.15
X_4	0.01	Y_4	0.13	Z_4	0.26
X_5	0.01	Y_5	0.1	Z_5	0.05
X_6	0.04	Y_6	0.02	Z_6	0.1
X_7	0.02	Y_7	0.19	Z_7	0.31
X_8	0.19	Y_8	0.12	Z_8	0.19
		Y_9	0.01		
		Y_{10}	0.04		
		X_2	0.15	X_3	0.31
Agg		X_2	0.12	X_3	0.27

13.1.5. Zadany stan sieci wieloskalowej

Rozpatrywany jest ustalony stan sieci wieloskalowej odpowiadający następującemu scenariuszowi zdarzeń.

Silnik nie uruchamia się pomimo ustawienia włącznika zapłonu na pozycję START $X_1 = tak$. Dodatkowym objawem podczas próby uruchomienia silnika są przygasające światła mijania w pojeździe $X_8 = tak$. Jednocześnie założono, że nie są one objawem zużytych szczotek rozrusznika, które w ostatnim czasie zostały wymienione $Y_5 = nie$. Drugim niepokojącym kierowcę objawem jest wolnoobracający się rozrusznik $Y_6 = tak$. Ponieważ ogólne objawy wskazały na rozładowanie akumulatora przeprowadzono test związany z pomiarem spadku napięcia podczas ponownego uruchamiania silnika Z_1 , pomiar dał wynik negatywny $Z_1 = nie$.

Wprowadzając tak ustalone wartości poszczególnych stwierdzeń uzyskano wyniki wartości stwierdzeń pokazane w tabeli 13.2. W przypadku ogólnych sieci stwierdzeń wartości przybliżone stwierdzeń dobrano w taki sposób, aby możliwe było obserwowanie sposobu propagacji tych wartości w sieci. Wartości uzyskane dla sieci stwierdzeń pokazano w kolumnach oznaczonych symbolem *SP*, natomiast wartości dla ogólnych sieci stwierdzeń pokazano w kolumnach oznaczonych symbolem *OSS*. Wartości zadane zaznaczono wytłuszczonym drukiem. Zagregowane wartości dla sieci stwierdzeń oraz w przypadku ogólnych sieci stwierdzeń, wartości stwierdzeń, których wartości przedziałów ustalono na podstawie ich wszystkich składowych w poszczególnych warstwach, podano w ostatnim wierszu tabeli.

Analiza wyników uzyskanych dla zadanego stanu wieloskalowej sieci stwierdzeń w obu wariantach wskazują, że nie ma odpowiednich informacji, które pozwoliłyby stwierdzić, że rozrusznik jest sprawny lub że nie jest sprawny. Wyniki dla tego stwierdzenia wynoszą: w przypadku sieci przekonań wartość zagregowana dla $X_2 = 0.4$, a w przypadku sieci stwierdzeń wartość iloczynu poszczególnych przedziałów dla stwierdzenia $X_2 = [0; 0.9]$. Dla obu wariantów sieci, wyniki dotyczące stanu akumulatora wynoszą dla sieci przekonań $X_3 = 0.2$, a dla ogólnych sieci stwierdzeń $X_3 = [0; 0.15]$ i wskazują na poprawny stan akumulatora. Szczegółowa analiza wyników dla sieci przekonań i jej warstwy dotyczącej rozrusznika wskazuje na konieczność sprawdzenia stanu przewodów łączących rozrusznik z układem zasilającym. W celu osiągnięcia bardziej zawężonych wartości przedziałów dla ogólnych sieci stwierdzeń konieczne jest pozyskanie dodatkowych informacji i wprowadzenie ich jako wartości zadane.

Tab. 13.2: Wyniki dla zadanych wartości wybranych stwierdzeń. Dla sieci przekonań SP podano wartości odpowiadające wystąpieniu stanu tak . OSS - wartości stwierdzeń dla ogólnych sieci stwierdzeń. Wartości zadane zaznaczono wytłuszczonym drukiem.

Warst. 1	SP	OSS	Warst. 2	SP	OSS	Warst. 3	SP	OSS
X_1	1	[0.8;0.95]	Y_1	0.1	[0; 1]	Z_1	0	[0.05;0.15]
X_2	0.07	[0; 0.9]	Y_2	0.78	[0; 0.9]	Z_2	0.1	[0; 1]
X_3	0.52	[0; 0.95]	Y_3	0.25	[0; 0.9]	Z_3	0.14	[0; 1]
X_4	0.1	[0; 0.95]	Y_4	1	[0.8;0.9]	Z_4	0.01	[0; 0.15]
X_5	0.13	[0; 0.95]	Y_5	0	[0.1;0.2]	Z_5	0.02	[0; 0.15]
X_6	0.55	[0; 0.95]	Y_6	0.06	[0; 0.9]	Z_6	0.01	[0; 0.15]
X_7	0.15	[0; 0.95]	Y_7	0.33	[0; 1]	Z_7	0.03	[0; 0.15]
X_8	1	[0.7;0.9]	Y_8	0.27	[0; 1]	Z_8	0.13	[0; 1]
			Y_9	0.01	[0; 1]			
			Y_{10}	0.04	[0; 1]			
			X_2	0.48	[0; 1]	X_3	0.13	[0; 0.15]
			X_2	0.4	[0; 0.9]	X_3	0.2	[0.15; 0.15]

13.2. Podsumowanie

W rozdziale pokazano prosty przykład sieci wieloskalowej dotyczącej diagnozowania stanu układu rozruchowego silnika pojazdu samochodowego. Przykład pokazano w dwóch wariantach sieci: jako sieć przekonań oraz ogólną sieć stwierdzeń. Przedstawiono działanie sieci oraz wyniki procesu wnioskowania dla domyślnych jak i ustalonych wartości stwierdzeń. Należy podkreślić, że opracowany przykład stanowi jedynie ilustrację budowy oraz stosowania wieloskalowych sieci stwierdzeń i może być znacznie rozszerzony. Zaletą opisywanego przykładu jest możliwość jego opracowywania przez wielu specjalistów, znających i opisujących wybraną dziedzinę np. wybrane komponenty instalacji technologicznej, jak i integrację tej wiedzy na poziomie ogólnym np. w warstwie ogólnej. W odróżnieniu od tradycyjnego modelowania diagnostycznego, modelowanie z zastosowaniem modeli graficznych w tym przede wszystkim modeli wielowarstwowych w systemie *MMNET*, stanowi pomocne środowisko dla potrzeb budowania diagnostycznych systemów doradczych, w których formalny zapis wiedzy jest usystematyzowany i transparentny.

Bibliografia

- [13.1] Cholewa W. *Fault Diagnosis. Models, Artificial Intelligence, Applications*, rozdział/Expert systems in technical diagnostics., strony 591–674. Springer, 2004.
- [13.2] Cholewa W., redaktor. *Szkieletowy system doradczy DIADYN*, wolumen 137 serii *Zeszyty*. Politechnika Śląska, Katedra Podstaw Konstrukcji Maszyn, Gliwice, 2008.
- [13.3] Cholewa W. Ogólny model sieci stwierdzeń. Raport wewnętrzny MMN-03-031, Politechnika Śląska, Katedra Podstaw Konstrukcji Maszyn, Gliwice, 2010.
- [13.4] Cholewa W., Kiciński J. *Diagnostyka techniczna. Odwrotne modele diagnostyczne*. Wydawnictwo Politechniki Śląskiej, Gliwice, 1997.
- [13.5] Cholewa W., Kiciński J. (red). *Diagnostyka techniczna, Metody dwracania nieliniowych modeli odwrotnych*. Wydawnictwo Politechniki Śląskiej, Gliwice, 2001.
- [13.6] Cholewa W., Korbicz J., Kościelny J., Chrzanowski P., Patan K., Rogala T., Syfert M., Witczak M. *Modelowanie, diagnostyka i sterowanie nadrzędne procesami - Implementa-*

cja w systemie DiaSter, rozdział I Metody diagnostyki., strony 211–316. WNT, W-wa, 2004.

- [13.7] Cholewa W., Korbicz J., Kościelny J.M., Kowalczyk Z. *Diagnostyka procesów. Modele. Metody sztucznej inteligencji. Zastosowania*. WNT, Warszawa, 2002.
- [13.8] Cholewa W., Rogala T. *Modele odwrotne i modelowanie diagnostyczne*, wolumen 136 serii *Zeszyty*. Politechnika Śląska, Katedra Podstaw Konstrukcji Maszyn, Gliwice, wydanie 2, 2008.
- [13.9] Zembowicz J. *Fiat Seicento, sam naprawiam*. Wydawnictwo Komunikacji i Łączności, wydanie 6, W-wa, 2010.

Rozdział 14

Diagnostyczny system doradczy dla symulatora procesu

Damian SKUPNIK

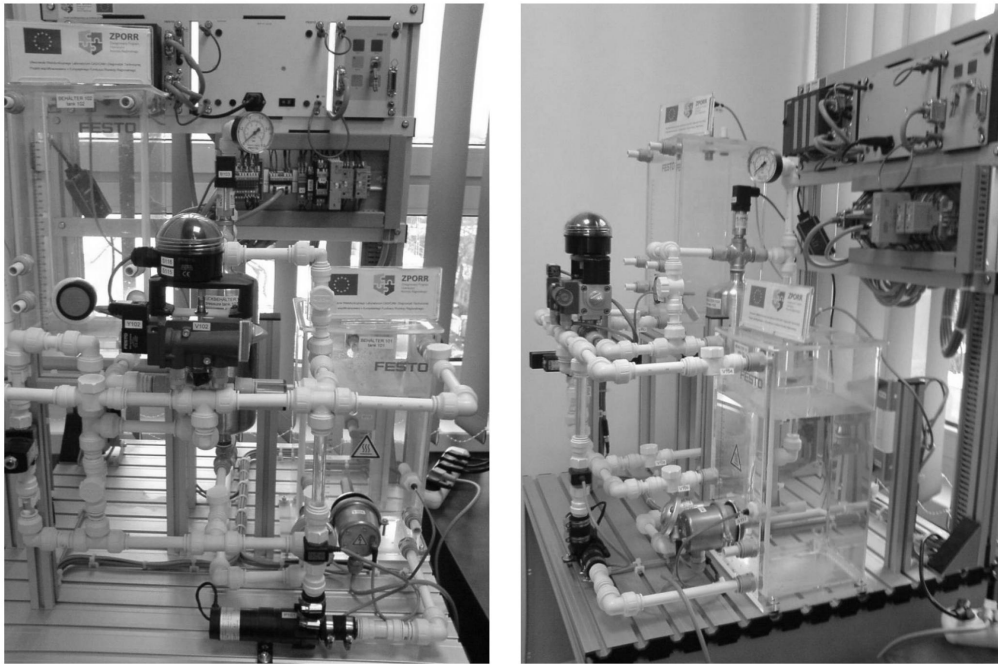
W niniejszym rozdziale przedstawiono, opracowany za pomocą pakietu MMNET i na podstawie [14.10], przykład diagnostycznego systemu doradczego dla symulatora procesu realizowanego przez obiekt techniczny, który stanowi fizyczną miniaturyzację instalacji przemysłowej stosowanej, np. w przemyśle chemicznym, spożywczym itp. Zidentyfikowano zbiór możliwych do rozpoznania stanów technicznych obiektu, przy czym uwzględniano zarówno proste jak i złożone stany techniczne. Przedstawiono sygnały, które powinny stanowić wystarczający nośnik informacji o stanie technicznym obiektu, a więc również o stanie procesu realizowanego przez ten obiekt.

W wyniku przeprowadzenia czynnych eksperymentów diagnostycznych pozyskano zbiory danych uczących, które umożliwiły skonstruowanie systemu doradczego składającego się z modeli umieszczonych w 3 warstwach. Wszystkie modele diagnostyczne reprezentowano w postaci sieci stwierdzeń, przy czym ich strukturę i parametry określono z zastosowaniem odpowiednio algorytmu *K2* oraz *Junction Tree*. Na podstawie zgromadzonego zbioru danych testujących dokonano weryfikacji działania systemu i ustalono wartości wag dla opracowanych modeli diagnostycznych.

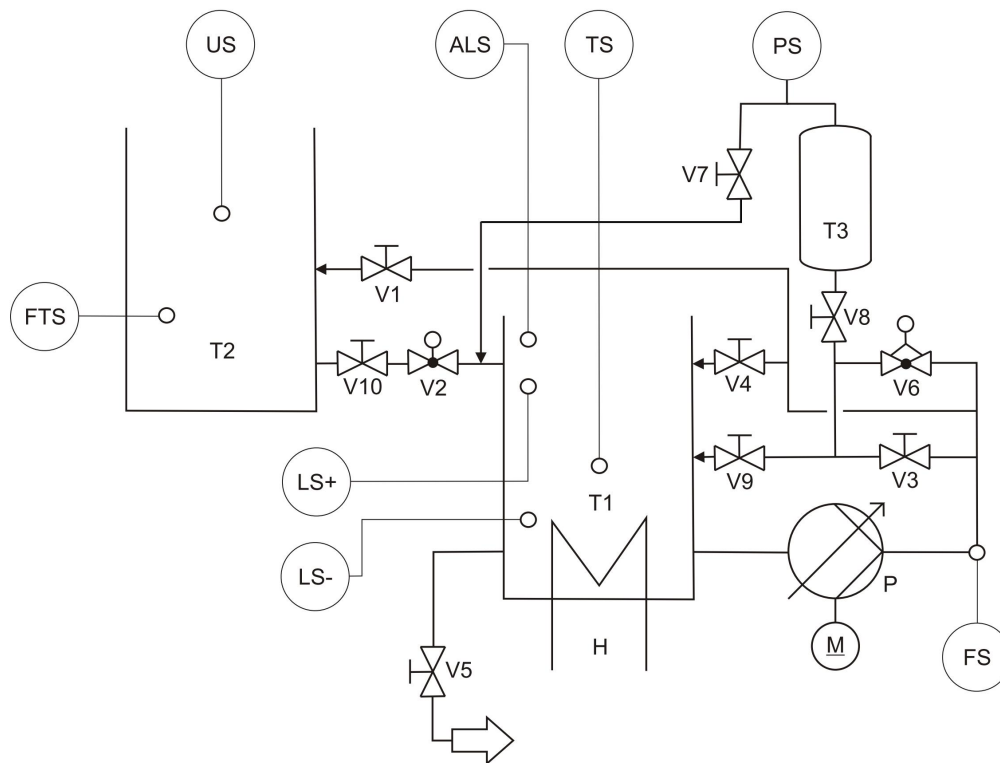
14.1. Rozpatrywany proces

Opracowany system doradczy umożliwia diagnostykę procesu realizowanego przez obiekt techniczny przedstawiony na rys. 14.1. Budowę obiektu w sposób schematyczny ilustruje rys. 14.2, przy czym nie pokazano na nim panelu sterowania, sterownika PLC, przyłączy wejść/wyjść cyfrowych i analogowych, przetworników sygnałów prąd–napięcie, częstotliwość–napięcie oraz termopara–napięcie.

Wymienione elementy, stanowiące układ sterowania działaniem obiektu, w ogólnym wypadku umożliwiają regulację jednej z czterech zmiennych procesowych: temperatury cieczy w zbiorniku T1, poziomu cieczy w zbiorniku T2, ciśnienia powietrza w zbiorniku T3 oraz natężenia przepływu cieczy w miejscu zamontowania czujnika FS. Strukturę układu, który umożliwia regulację danej zmiennej procesowej uzyskuje się poprzez odpowiednią kombinację otwarcia/zamknięcia zaworów ręcznych *V_i*. W związku z tym w trakcie działania obiektu czynności te musi wykonywać operator.



Rys. 14.1: FESTO S7 EduTrainer Compact Siemens S7-300 CPU313C

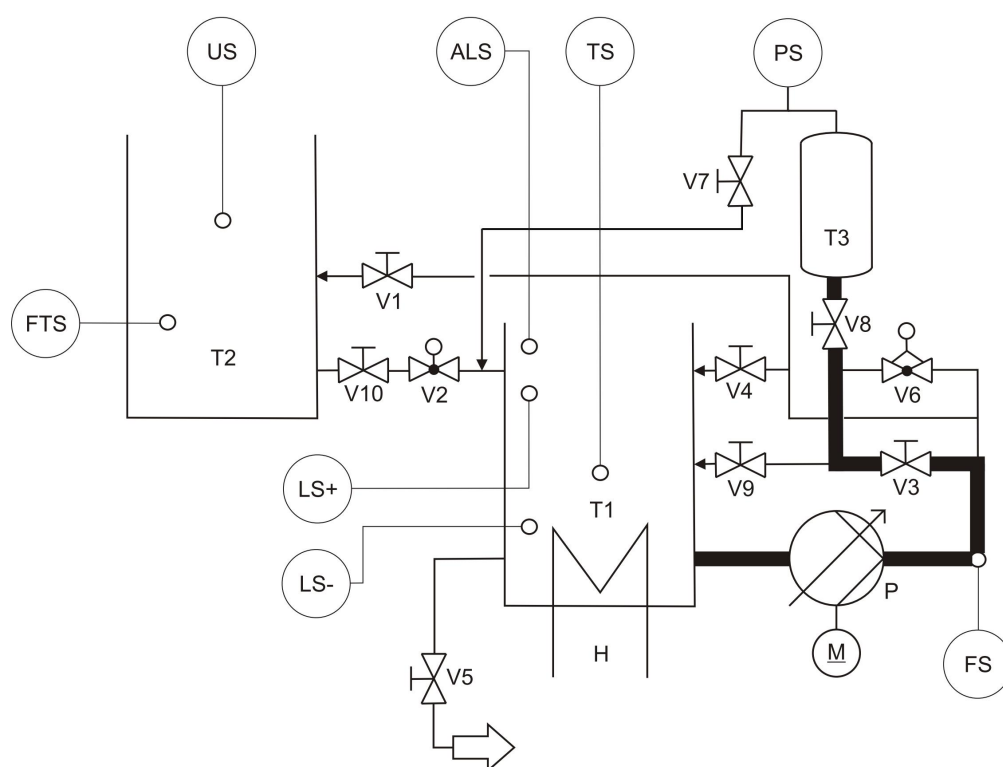


Rys. 14.2: Schemat rozpatrywanego obiektu technicznego (na podstawie [14.3])

gdzie:

- ALS — przełącznik pływakowy poziomu alarmowego;
- FS — czujnik natężenia przepływu cieczy;
- FTS — przełącznik pływakowy;
- H — grzałka;
- LS- — czujnik zbliżeniowy dolny;
- LS+ — czujnik zbliżeniowy górny;
- M — silnik;
- P — pompa odśrodkkowa;
- PS — czujnik ciśnienia;
- T1, T2 — odpowiednio zbiornik dolny i górny;
- T3 — zbiornik ciśnieniowy;
- TS — czujnik temperatury;
- US — ultradźwiękowy czujnik poziomu;
- V2 — zawór kulowy;
- V6 — zawór proporcjonalny;
- V_i — zawory ręczne ($i = 1, 3, 4, 5, 7, 8, 9, 10$; V5 — zawór spustowy).

Opracowany symulator procesu dotyczy regulacji ciśnienia powietrza w zbiorniku T3. Schemat przepływu cieczy podczas sprężania powietrza w zbiorniku T3 ilustruje rys. 14.3.



Rys. 14.3: Schemat przepływu cieczy podczas sprężania powietrza w zbiorniku T3

Działanie obiektu sterowane jest opracowanym przez autora programem, który wykonywany jest przez sterownik PLC. Po sprawdzeniu, że obiekt znajduje się w wymaganym stanie początkowym, tzn. wtedy, gdy:

- w zbiorniku dolnym T1 znajduje się minimalna ilość cieczy konieczna do zmiany stanu czujnika zbliżeniowego dolnego LS- z 0 na 1,
- w zbiorniku górnym T2 wysokość słupa cieczy wynosi 155 ± 5 mm,
- w zbiorniku ciśnieniowym T3 nie ma cieczy, a ciśnienie zawartego w nim powietrza jest równe ciśnieniu atmosferycznemu,
- temperatura cieczy w zbiorniku T1 wynosi około 20 °C,
- zawory ręczne V_j ($j = 1, 3, 4, 5, 7, 8, 9$) są zamknięte,
- zawór kulowy V2 jest zamknięty,
- zawór proporcjonalny V6, pompa P oraz grzałka H są wyłączone,

operator obiektu otwiera zawory ręczne V3 i V8 (rys. 14.3) oraz naciska przycisk „Start” umieszczony na panelu sterowania. Następuje uruchomienie, na czas 20. sekund, pompy P i zwiększanie względnego ciśnienia powietrza w zbiorniku T3 do wartości 200 mbar (poziomym odniesienia jest ciśnienie powietrza w otoczeniu obiektu). W celu osiągnięcia i ustabilizowania wymaganej wartości ciśnienia, zastosowano regulator PID.

W wyniku powyższej operacji do zbiornika T3 zostaje wtłoczona ze zbiornika dolnego T1 ściśle określona ilość cieczy. Tuż przed upływem 20. sekundy działania obiektu operator zamyka zawór ręczny V8. Po 20. sekundzie następuje wyłączenie pompy P i zatrzymanie działania obiektu. Powietrze w zbiorniku T3 zostało sprężone do wymaganego ciśnienia.

14.2. Zbiór rozpatrywanych stanów technicznych obiektu

Na podstawie analizy działania rozpatrywanego obiektu zidentyfikowano i przedstawiono w tab. 14.1 zbiór 4 prostych stanów technicznych (oprócz stanu bazowego wszystkie pozostałe stany techniczne dotyczą tylko jednego elementu obiektu). Zrezygnowano z rozpatrywania uszkodzeń toru pomiarowego, zarówno w odniesieniu do czujnika ciśnienia PS, natężenia przepływu FS jak i sterownika PLC, z uwagi na brak części ruchomych, które mogłyby ulec przyspieszonemu zużyciu.

Możliwe jest jednak wystąpienie złożonego stanu technicznego, tzn. przypadku gdy w tym samym czasie uszkodzone są co najmniej dwa elementy obiektu. W związku z tym liczba możliwych do detekcji prostych i złożonych stanów technicznych w rozpatrywanym przykładzie wynosi 12 (tab. 14.2).

Tab. 14.1: Lista możliwych do detekcji prostych stanów technicznych obiektu

ID	OPIS PROSTEGO STANU TECHNICZNEGO
1.	Obiekt techniczny sprawny (stan bazowy)
2.	Pompa P nie pompuje (symulowane poprzez modyfikację programu sterującego działaniem obiektu)
3.	Przytkanie kanału pomiędzy zbiornikiem T1 i zbiornikiem ciśnieniowym T3 (dwa warianty, symulowane poprzez częściowe lub całkowite zamknięcie zaworu ręcznego V8)
4.	Nieszczelność zbiornika T3 (symulowana poprzez otwarcie zaworu ręcznego V7)

Tab. 14.2: Zbiór wszystkich rozpatrywanych stanów technicznych obiektu

ID	OPIS STANU TECHNICZNEGO
z ₁	Stan bazowy (obiekt sprawny)
z ₂	Pompa P pompuje, kanał pomiędzy zbiornikami T1 i T3 jest prawidłowy, zbiornik T3 jest nieszczelny
z ₃	Pompa P pompuje, kanał pomiędzy zbiornikami T1 i T3 jest częściowo przytkany, zbiornik T3 jest szczelny
z ₄	Pompa P pompuje, kanał pomiędzy zbiornikami T1 i T3 jest częściowo przytkany, zbiornik T3 jest nieszczelny
z ₅	Pompa P pompuje, kanał pomiędzy zbiornikami T1 i T3 jest całkowicie przytkany, zbiornik T3 jest szczelny
z ₆	Pompa P pompuje, kanał pomiędzy zbiornikami T1 i T3 jest całkowicie przytkany, zbiornik T3 jest nieszczelny
z ₇	Pompa P nie pompuje, kanał pomiędzy zbiornikami T1 i T3 jest prawidłowy, zbiornik T3 jest szczelny
z ₈	Pompa P nie pompuje, kanał pomiędzy zbiornikami T1 i T3 jest prawidłowy, zbiornik T3 jest nieszczelny
z ₉	Pompa P nie pompuje, kanał pomiędzy zbiornikami T1 i T3 jest częściowo przytkany, zbiornik T3 jest szczelny
z ₁₀	Pompa P nie pompuje, kanał pomiędzy zbiornikami T1 i T3 jest częściowo przytkany, zbiornik T3 jest nieszczelny
z ₁₁	Pompa P nie pompuje, kanał pomiędzy zbiornikami T1 i T3 jest całkowicie przytkany, zbiornik T3 jest szczelny
z ₁₂	Pompa P nie pompuje, kanał pomiędzy zbiornikami T1 i T3 jest całkowicie przytkany, zbiornik T3 jest nieszczelny

14.3. Zbiór rozpatrywanych sygnałów diagnostycznych

Pozyskanie informacji o rozpatrywanych stanach technicznych obiektu jest możliwe m.in. poprzez pomiar przebiegu określonych wielkości fizycznych. Przedstawione na rys. 14.2 czujniki pomiarowe umożliwiają rejestrację wielu sygnałów. Na podstawie analizy działania rozpatrywanego obiektu można stwierdzić, że nie ma potrzeby obserwowania zmian wszystkich dostępnych do pomiaru wielkości fizycznych. W tab. 14.3 przedstawiono te sygnały, które powinny stanowić wystarczający nośnik informacji o rozważanych stanach technicznych badanego obiektu.

Tab. 14.3: Zbiór rozpatrywanych sygnałów

L.P.	RODZAJ SYGNAŁU
1.	Sygnał analogowy zmian wielkości sterowanej (prędkość obrotowa wirnika pompy P)
2.	Sygnał analogowy zmian ciśnienia powietrza w zbiorniku T3 (uzyskiwany z czujnika ciśnienia PS)
3.	Sygnał analogowy zmian natężenia przepływu cieczy (uzyskiwany z czujnika natężenia przepływu cieczy FS)
4.	Sygnał cyfrowy upływu czasu (uzyskiwany z wewnętrznego zegara sterownika PLC)

14.4. Identyfikacja warstw modelu

Zagadnienie identyfikacji warstw modelu wielowarstwowego na ogół nie jest zadaniem prostym. W prezentowanym przykładzie zostało ono rozwiązane za pomocą języka UML, który posłużył do precyzyjnego i jednoznacznego opisu procesu realizowanego przez rozpatrywany obiekt techniczny.

Istnieje wiele pozycji literaturowych dotyczących języka UML, np. do najnowszych należą [14.2, 14.8, 14.11, 14.12]. W skrócie można powiedzieć, że jest to język formalny umożliwiający tworzenie zestawu diagramów, za pomocą których określana jest z różnych punktów widzenia struktura i/lub dynamika działania modeli reprezentujących realne obiekty z wybranego fragmentu otoczenia (np. osoby, przedmioty, jednostki organizacyjne, zdarzenia itp.).

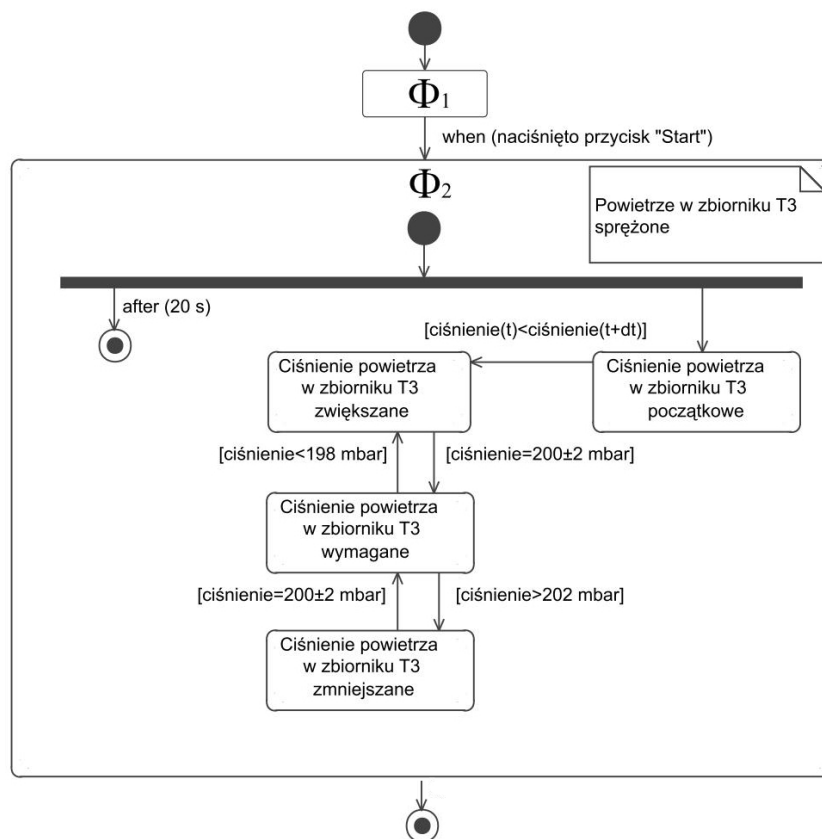
W wersji 2.0 języka UML wyróżniono trzynaście diagramów. Sześć z nich przeznaczonych jest do opisu struktury, natomiast pozostałe pozwalają opisać dynamikę (działanie) systemu. Jak pokazuje praktyka w wielu wypadkach stosowanie wszystkich diagramów lub nadmierna dokładność modelowania może prowadzić do zatracenia głównego celu w wyniku tak zwanej „gorączki” modelowania [14.1]. Poniżej przedstawiono pełny wykaz diagramów UML [14.11, 14.12]:

- diagramy opisujące strukturę systemu:
 - *diagram klas,*
 - *diagram obiektów,*
 - *diagram pakietów,*
 - *diagram komponentów,*
 - *diagram rozlokowania,*
 - *diagram struktur połączonych.*
- diagramy opisujące dynamikę systemu:
 - *diagram przypadków użycia,*
 - *diagram maszyny stanowej,*
 - *diagram czynności,*
 - *diagram sekwencji,*
 - *diagram komunikacji,*
 - *diagram harmonogramu,*
 - *diagram sterowania interakcją.*

Na rys. 14.4 do 14.6 przedstawiono opracowane diagramy UML opisujące przebieg rozpatrywanego procesu. W wyniku ich analizy przyjęto, że tworzony w systemie MMNET model diagnostyczny będzie składał się z następujących 3 warstw:

1. **Warstwa dotycząca stanu funkcjonalnego obiektu** — opisuje ją diagram maszyny stanowej (rys. 14.4). Rozróżnić można 3 podstawowe stany funkcjonalne obiektu:
 - (a) postój,
 - (b) sprężanie powietrza w zbiorniku T3,
 - (c) niesprężanie powietrza w zbiorniku T3 (pomimo działania obiektu).
2. **Warstwa dotycząca działania wybranych elementów obiektu** — opisuje ją diagram czynności (rys. 14.5). Jak wspomniano wcześniej uszkodzenia czujników oraz sterownika PLC nie są rozpatrywane podczas formułowania diagnozy o stanie technicznym obiektu.

3. **Warstwa dotycząca harmonogramu działania wybranych elementów obiektu** — opisuje ją diagram harmonogramu (rys. 14.6), przedstawiający zmianę stanu funkcjonalnego wybranych elementów obiektu w funkcji czasu.



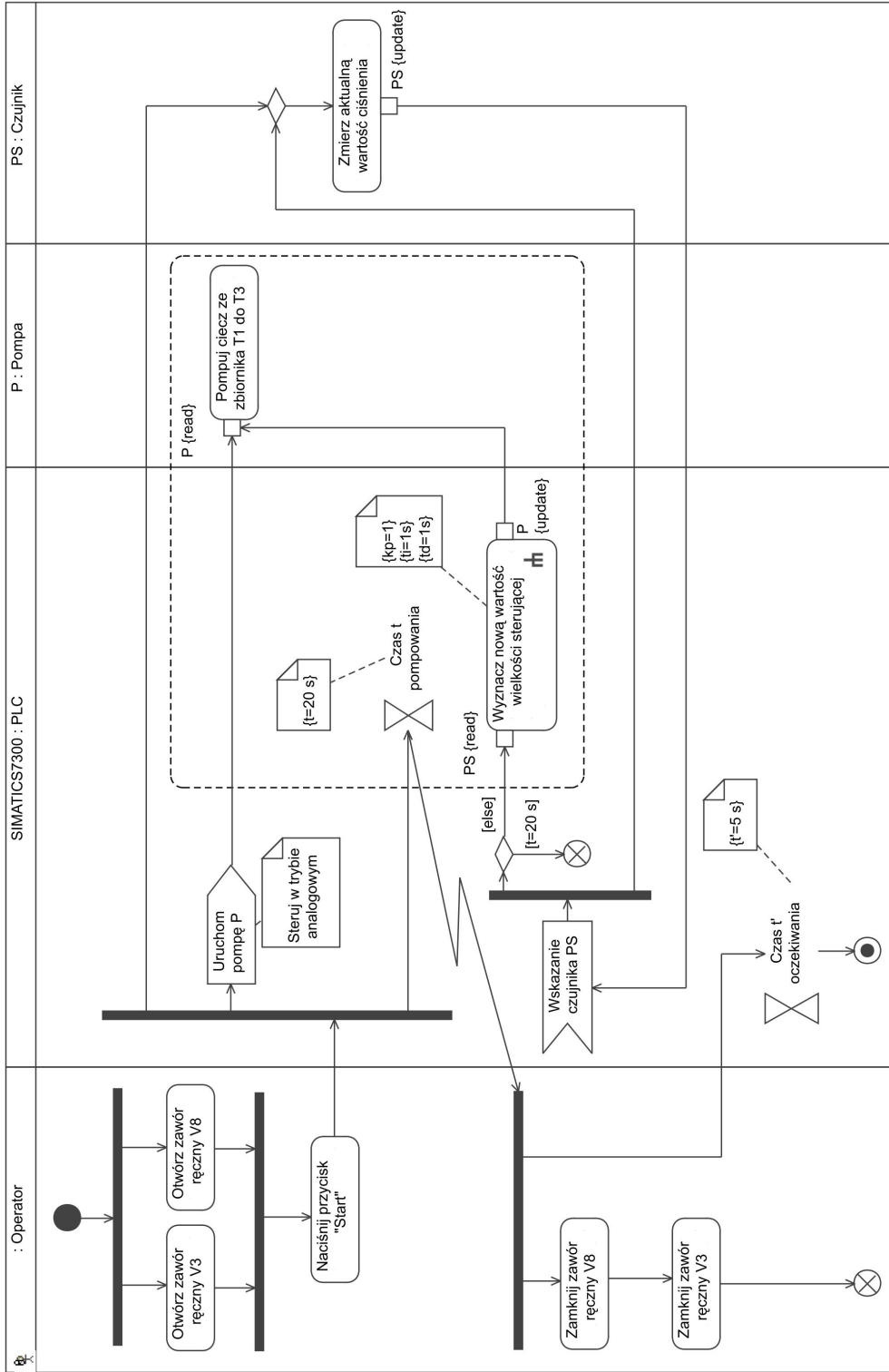
Rys. 14.4: Diagram maszyny stanowej

14.5. Identyfikacja zbioru cech sygnałów diagnostycznych

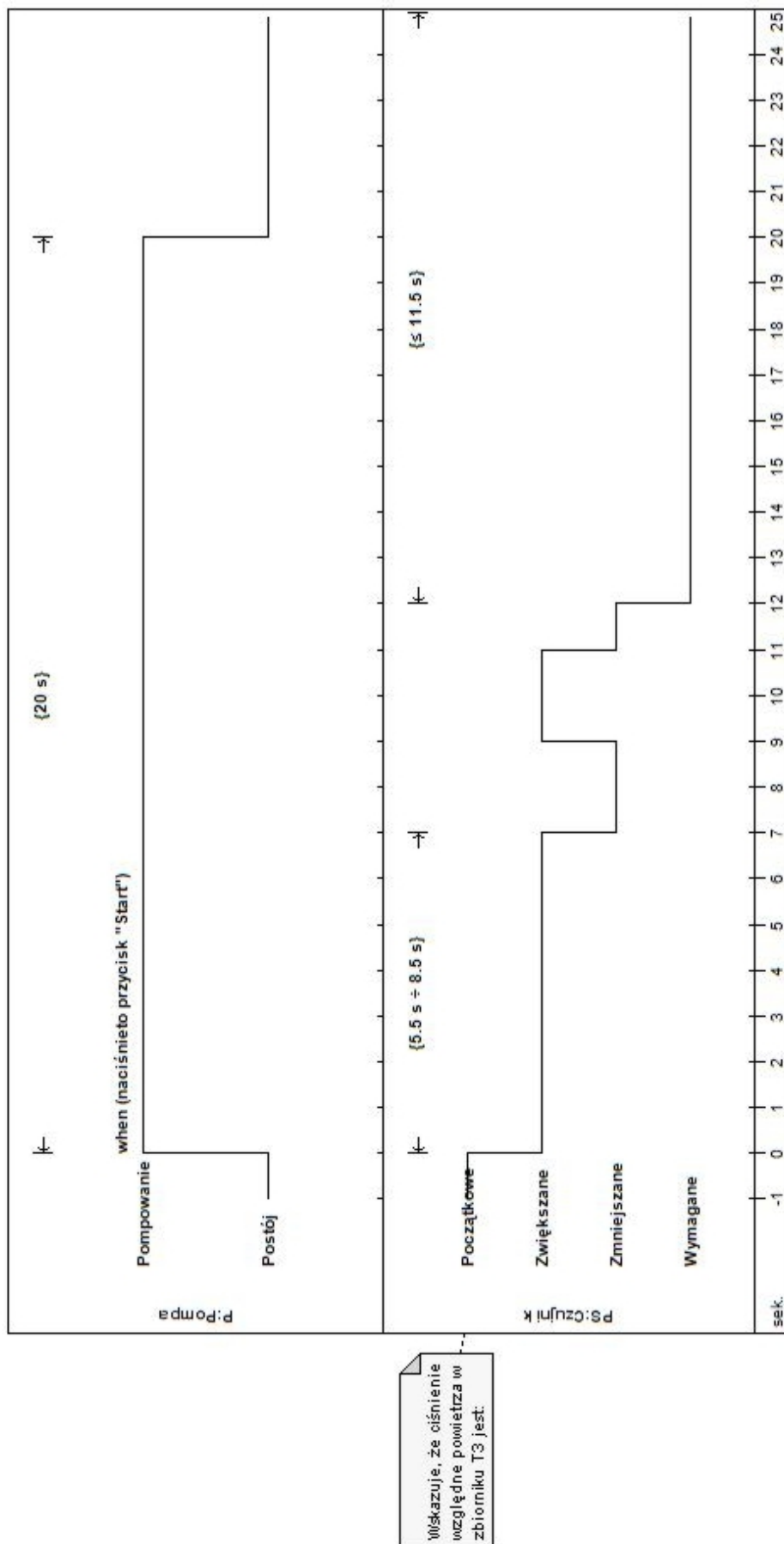
Warto zauważyć, że na podstawie opracowanych diagramów UML ułatwiona jest identyfikacja zbioru cech rozpatrywanych sygnałów diagnostycznych. Wyróżnione cechy sygnałów zostały przedstawione w tab. 14.4. Pierwszy indeks oznaczenia cechy sygnału dotyczy numeru warstwy, natomiast kolumna o nagłówku „RYS.” wskazuje numery rysunków (diagramów), na podstawie których określono rozpatrywane cechy.

Tab. 14.4: Zbiór rozpatrywanych cech sygnałów diagnostycznych

OZN.	CECHA SYGNAŁU DIAGNOSTYCZNEGO	RYS.
$s_{1,1}$	Wzrastająca wartość ciśnienia powietrza w zbiorniku T3	14.4
$s_{1,2}$	Malejąca wartość ciśnienia powietrza w zbiorniku T3	14.4
$s_{1,3}$	Wartość ciśnienia powietrza w zbiorniku T3	14.4
$s_{2,1}$	Wartość natężenia przepływu cieczy	14.5
$s_{2,2}$	Wartość wielkości sterującej	14.5
$s_{3,1}$	Czas zwiększania ciśnienia w zbiorniku T3	14.6



Rys. 14.5: Diagram czynności elementów obiektu



Rys. 14.6: Diagram harmonogramu działania elementów obiektu

14.6. Przygotowanie zbioru danych

Do niedawna pozyskiwanie zbioru danych dla celów diagnostycznych w przeważającej większości realizowane było na podstawie obserwacji obiektu bez modyfikowania w sposób zamierzony jego stanu technicznego (tzw. bierny eksperyment diagnostyczny). Jest to związane z potrzebą wzięcia pod uwagę kryteriów bezpieczeństwa oraz kosztów. Wynikający z tego długi czas badań można skrócić obserwując dużą liczbę podobnych obiektów.

Obecnie coraz częściej istnieje możliwość opracowywania dokładnych numerycznych modeli obiektów technicznych, które pozwalają wiernie symulować działanie danego obiektu. W związku z tym możliwe jest bezpieczne i niedrogo prowadzenie tzw. czynnych eksperymentów diagnostycznych. Polegają one na celowym wprowadzaniu uszkodzeń w badanym obiekcie, po to aby wykryć zewnętrzne objawy ich występowania.

W prezentowanym przykładzie rozpatrywany jest fizyczny model instalacji przemysłowej (rys. 14.1), a więc pozyskanie wiarygodnych danych jest wyjątkowo ułatwione. Przeprowadzone czynne eksperymenty diagnostyczne polegały na zarejestrowaniu sygnałów przedstawionych w tab. 14.3 dla wszystkich rozpatrywanych stanów technicznych obiektu (tab. 14.2), przy czym dla stanu bazowego zgromadzono 8 przykładów (dla innych stanów po 1 przykładzie).

Poszczególne przykłady zostały powielone proporcjonalnie do określonego w sposób subiektywny prawdopodobieństwa a priori zaistnienia danego stanu technicznego. Tak przygotowany zbiór danych stanowił podstawę do konstruowania w sposób automatyczny modeli diagnostycznych dla wybranych warstw systemu.

14.7. Modele diagnostyczne dla kolejnych warstw systemu

Zdecydowano, że modele diagnostyczne dla kolejnych warstw systemu będą reprezentowane w postaci sieci bayesowskich (przekonań). Tworzenie modelu reprezentowanego w tej postaci polega na zdefiniowaniu zmiennych i relacji między nimi (określenie struktury sieci) oraz przyjęciu odpowiednich prawdopodobieństw a priori i prawdopodobieństw warunkowych.

W systemie MMNET zmienne reprezentowane są w postaci stwierdzeń, a więc zdań oznajmujących, którym można przypisać wartość logiczną („Prawda”, „Fałsz”) lub stopień przekonania o prawdziwości ich treści. W związku z powyższym opracowywanie modelu wielowarstwowego w systemie MMNET należy rozpocząć od przygotowania odpowiedniego słownika stwierdzeń, tzw. tezauryusa.

14.7.1. Słownik stwierdzeń

Na podstawie analizy rozpatrywanych stanów technicznych obiektu oraz cech sygnałów diagnostycznych opracowano zbiór 9 stwierdzeń. W tab. 14.5 przedstawiono ich treść oraz możliwe do przyjęcia wartości.

14.7.2. Określenie struktur sieci stwierdzeń

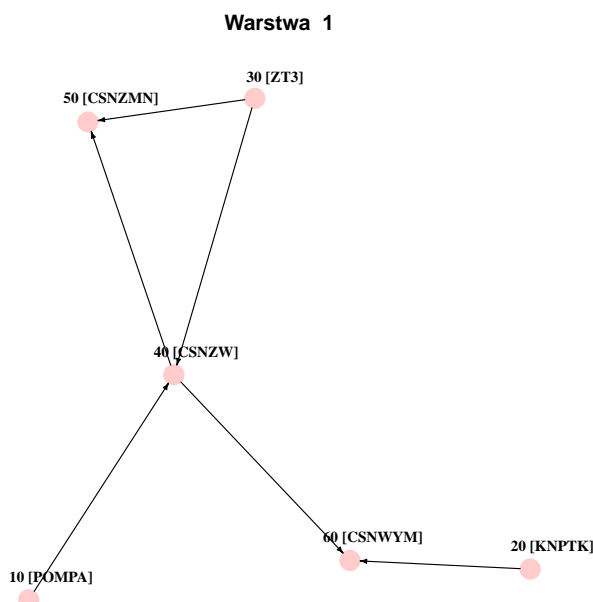
Identyfikację struktur sieci przekonań przeprowadzono w środowisku MATLAB z zastosowaniem dostępnej bezpłatnie biblioteki BNT [14.9]. Sformułowane stwierdzenia stanowią węzły sieci. W celu identyfikacji połączeń węzłów zastosowano algorytm K2 dla przygotowanego wcześniej zbioru danych. Aby móc skorzystać z tego algorytmu należy podać m.in. wybraną kolejność węzłów. Z uwagi na fakt niewielkiej liczby węzłów sieci w danej warstwie rozpatrywano wszystkie możliwe permutacje kolejności. W rezultacie, w wielu wypadkach, otrzymano kilka różnych struktur równoważnych sieci dla danej warstwy. Wybrano te, dla których pozyskanie wartości prawdopodobieństw warunkowych wydaje się najłatwiejsze (np. poprzez przyjęcie założenia, że

Tab. 14.5: Tezaurus dla rozpatrywanego przykładu

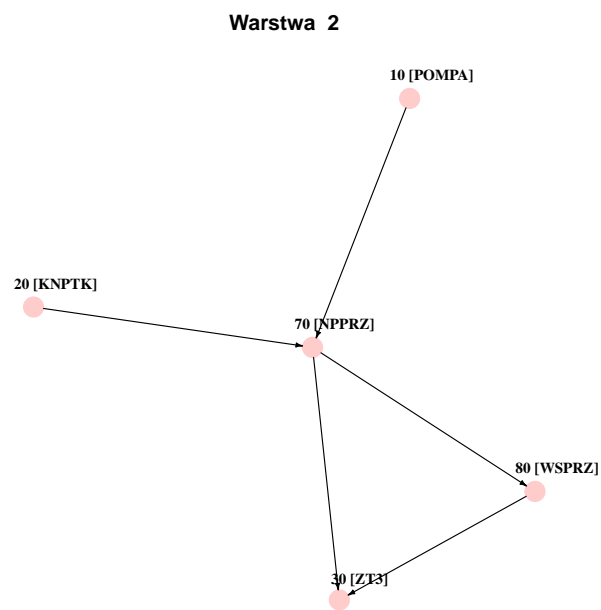
L.P.	TREŚĆ STWIERDZENIA	SKRÓCONA NAZWA	MOŻLIWE WARTOŚCI
1.	Pompa działa prawidłowo	POMPA	TAK/NIE
2.	Kanał pomiędzy zbiornikami T1 i T3 jest przytkany	KNPTK	NIE/CZĘŚCIOWO/CAŁKOWICIE
3.	Zbiornik T3 jest szczelny	ZT3	TAK/NIE
4.	Ciśnienie powietrza w zbiorniku T3 jest zwiększane	CSNZW	TAK/NIE
5.	Ciśnienie powietrza w zbiorniku T3 jest zmniejszane	CSNZMN	TAK/NIE
6.	Ciśnienie powietrza w zbiorniku T3 jest sprężone do wymaganej wartości	CSNWYM	TAK/NIE
7.	Wartość natężenia przepływu ma oczekiwany przebieg	NPPRZ	TAK/NIE
8.	Wartość wielkości sterowanej ma oczekiwany przebieg	WSPRZ	TAK/NIE
9.	Czas zwiększania ciśnienia w zbiorniku T3 jest dopuszczalny	CZWCSN	TAK/WARUNKOWO/NIE

węzeł dotyczący stwierdzenia odnoszącego się do wybranego elementu obiektu nie może być potomkiem węzła dotyczącego stwierdzenia związanego z cechą sygnału diagnostycznego).

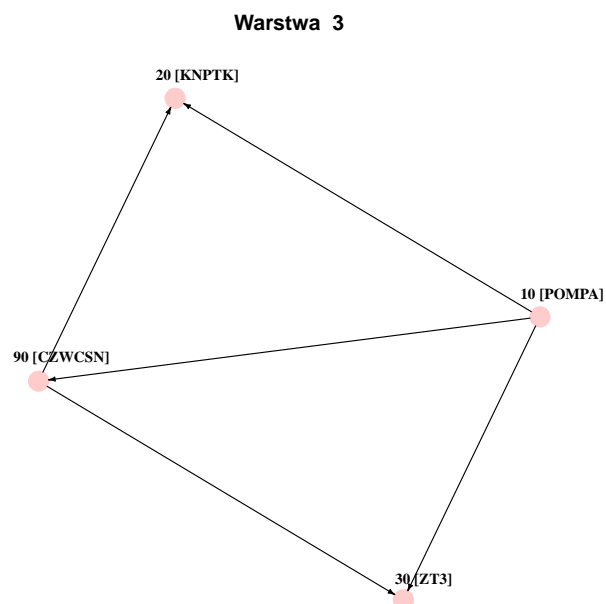
Na rys. 14.7 do 14.9 przedstawiono wybrane i zidentyfikowane w sposób automatyczny struktury sieci stwierdzeń.



Rys. 14.7: Struktura sieci stwierdzeń w warstwie 1



Rys. 14.8: Struktura sieci stwierżeń w warstwie 2



Rys. 14.9: Struktura sieci stwierżeń w warstwie 3

14.7.3. Określenie wartości prawdopodobieństw warunkowych

Wartości prawdopodobieństw warunkowych zostały zidentyfikowane w środowisku MATLAB. Po określeniu struktur sieci we wszystkich warstwach oraz mając do dyspozycji odpowiedni zbiór danych, dokonano wstępnego oszacowania poszukiwanych wartości prawdopodobieństw warunkowych w oparciu o rozkład Dirichleta (zakładając brak jakiegokolwiek wiedzy), a następnie zaktualizowano te wartości dzięki zastosowaniu algorytmu z grupy Junction Tree (informacje na temat tej klasy algorytmów można znaleźć np. w [14.5, 14.6, 14.7]).

W prezentowanym przykładzie nie ma potrzeby wstępnego oszacowania wartości prawdopodobieństw warunkowych, ponieważ przygotowany zbiór danych zawierał przykłady dotyczące wszystkich rozpatrywanych stanów technicznych obiektu. Gdyby przygotowany zbiór danych nie zawierał przykładów dotyczących pewnego stanu technicznego obiektu, co mogłoby wynikać, np. z potrzeby ograniczenia rozmiaru zbioru, to prawdopodobieństwo wystąpienia takiego stanu w tworzonej sieci, byłoby równe zero (zgodnie ze sposobem określania prawdopodobieństwa na podstawie częstości występowania zdarzeń). Chcąc uniknąć takiego błędu dobrym zwyczajem jest wstępne oszacowanie prawdopodobieństw warunkowych na podstawie rozkładu Dirichleta lub rozkładu beta dla zmiennych binarnych [14.4].

14.7.4. Określenie wartości wag modeli

W systemie MMNET tworzona jest w sposób automatyczny, tzn. bez udziału użytkownika, warstwa specjalna (tzw. warstwa zerowa), w której ulokowane są stwierdzenia z wszystkich sieci znajdujących się w zdefiniowanych wcześniej warstwach. Stwierdzeniom zawartym w warstwie zerowej przypisane są zagregowane wartości stopni przekonań o prawdziwości tych stwierdzeń, przy czym wybranym operatorem agregacji jest średnia ważona. Oznacza to, że użytkownik systemu może wpływać na wynik końcowy poprzez modyfikację wartości wag przypisanych do modeli ulokowanych w danej warstwie.

Wartości wag dla warstw, w których znajdują się opracowane sieci stwierdzeń zostały określone na podstawie analizy wyników uzyskiwanych z tych sieci. Utworzono dwa podzbiory danych:

- podzbiór danych wzorcowych, który zawierał znane wartości stopni przekonań o prawdziwości stwierdzeń dotyczących pompy P, kanału pomiędzy zbiornikami T1 i T3 oraz zbiornika T3 w odniesieniu do kolejnych przykładów zawartych w zbiorze danych wejściowych,
- podzbiór wyników uzyskiwanych z sieci stwierdzeń znajdujących się w opracowanych warstwach modelu dla kolejnych przykładów zawartych w zbiorze danych wejściowych.

Zbiór danych wejściowych stanowił zbiór pojedynczych przykładów dotyczących rozpatrywanych stanów technicznych obiektu (punkt 14.6).

W tab. 14.6 i 14.7 przedstawiono przykład ukazujący wartości stopni przekonań o prawdziwości stwierdzenia, że zbiornik T3 jest szczelny odpowiednio dla podzbioru danych wzorcowych i podzbioru uzyskanych wyników.

Tab. 14.6: Wzorcowe wartości stopni przekonań o prawdziwości stwierdzenia, *Zbiornik T3 jest szczelny* wtedy, gdy zbiornik T3 był nieszczelny

WARTOŚĆ STWIERDZENIA	WARTOŚCI STOPNI PRZEKONAŃ DLA SIECI STWIERDZEŃ		
	Warstwa W1	Warstwa W2	Warstwa W3
TAK	0	0	0
NIE	1	1	1

Tab. 14.7: Uzyskane wartości stopni przekonań o prawdziwości stwierdzenia, *Zbiornik T3 jest szczelny* wtedy, gdy zbiornik T3 był nieszczelny (wartości zaokrąglono)

WARTOŚĆ STWIERDZENIA	WARTOŚCI STOPNI PRZEKONAŃ DLA SIECI STWIERDZEŃ		
	Warstwa W1	Warstwa W2	Warstwa W3
TAK	0.04	0.01	0.73
NIE	0.96	0.99	0.27

Chcąc określić odchyłki uzyskanych wartości stopni przekonań od wartości wzorcowych przyjęto, że wartości stopni przekonań stanowią współrzędne punktów w przestrzeni wielowymiarowej. Następnie obliczono odległości (metryka euklidesowa) pomiędzy tymi punktami. Dla prezentowanego przykładu uzyskane odchyłki wynoszą [0.057, 0.014, 1.032]. Dokonując analogicznych działań dla kolejnych przykładów znajdujących się w zbiorze danych wejściowych oraz dodając odchyłki dotyczące stwierdzeń odnoszących się do pompy P, kanału pomiędzy zbiornikami T1 i T3 oraz zbiornika T3, otrzymano sumaryczny błąd (tab. 14.8) generowany przez sieci stwierdzeń z kolejnych warstw.

Tab. 14.8: Sumaryczny błąd generowany przez opracowane sieci dla rozpatrywanych stwierdzeń

TREŚĆ STWIERDZENIA	SUMARYCZNY BŁĄD DLA SIECI STWIERDZEŃ		
	Warstwa W1	Warstwa W2	Warstwa W3
Pompa działa prawidłowo	9.2059	8.0652	9.9084
Kanał pomiędzy zbiornikami T1 i T3 jest przytkany	9.2866	9.5333	8.4639
Zbiornik T3 jest szczelny	6.9629	5.6851	7.7255
Σ	25.4554	23.2836	26.0978

Jasne jest, że sieć stwierdzeń, która generuje najmniejszy błąd powinna mieć największą wagę. W związku z tym zdecydowano, że wartości wag dla opracowanych sieci będą wyznaczone na podstawie następującej zależności:

$$W = \frac{C}{\Sigma}$$

gdzie C to dowolna stała. Po przyjęciu $C = 100$ i zaokrągleniu wyników otrzymano wektor następujących wag dla kolejnych sieci stwierdzeń [3.93, 4.30, 3.83].

14.8. Weryfikacja systemu

Weryfikację opracowanego w systemie MMNET diagnostycznego systemu doradczego przeprowadzono z zastosowaniem metody resubstytucji. W większości wypadków prowadzi to do zawyżania wyników oceny jakości modelu, jednak dla rozpatrywanego obiektu wpływ czynników zakłócających jego działanie jest nieistotny a wprowadzenie do danych odpowiedniego szumu nie jest proste. Oznacza to, że dane testujące pozyskane w wyniku powtórzenia czynnych eksperymentów diagnostycznych nie będą różniły się w sposób istotny statystycznie od danych, na podstawie których opracowywano modele. Ponadto wystąpienie stanu technicznego, który nie został uwzględniony podczas identyfikacji modeli sieci stwierdzeń jest mało prawdopodobne.

Diagnoza klasyfikowana jest jako trafna, jeżeli uzyskany stopień przekonania o prawdziwości danego stwierdzenia różni się od wartości wzorcowej nie więcej niż o 0.4. Biorąc pod uwagę przykładowe wyniki przedstawione w tab. 14.7 należy stwierdzić, że dla opracowanych sieci stwierdzeń błędy wynoszą odpowiednio 0.04, 0.01, 0.73, a więc diagnoza jest trafna tylko dla dwóch pierwszych sieci.

Diagnoza klasyfikowana jest jako nietrafna, jeżeli popełniony zostanie jeden z następujących błędów:

- **B1** — niewskazanie istniejącego uszkodzenia (np. model z warstwy nr 3 w przykładzie przedstawionym w tab. 14.7),
- **B2** — wskazanie nieistniejącego uszkodzenia.

W tab. 14.9 przedstawiono procentową trafność określania stanu technicznego pompy P, kanału pomiędzy zbiornikami T1 i T3 oraz zbiornika T3 (wyniki zostały zaokrąglone „w dół”) a także procent popełnionych błędów typu **B1** i **B2** (zaokrąglone „w górę”). Kolumna **N** określa liczbę niesprawnych elementów obiektu.

Tab. 14.9: Trafność diagnoz formułowanych przez opracowany system diagnostyczny dla rozpatrywanego symulatora procesu

N	TRAFNOŚĆ DIAGNOZY, %			B1, %	B2, %
	Pompa P	Kanał T1T3	Zbiornik T3		
0	100	100	100	0	0
1	50	50	100	25	9
2	20	0	40	74	7
3	0	0	0	100	0

Analizując wyniki uzyskane z opracowanego modelu należy stwierdzić, że nie działa on w sposób zadowalający zwłaszcza w odniesieniu do złożonych stanów technicznych, tzn. wtedy, gdy uszkodzone są co najmniej 2 elementy obiektu.

Z tab. 14.8 wynika, że nie można jednoznacznie stwierdzić, który model daje najlepsze wyniki, np. do orzekania o stanie technicznym pompy najlepszy wydaje się model z warstwy 2 ale w odniesieniu do kanału pomiędzy zbiornikami najlepsze wyniki otrzymuje się najczęściej z modelu ulokowanego w warstwie 3. W tab. 14.10 przedstawiono kolejność modeli pod względem trafności diagnoz dotyczących stanu technicznego pompy P w odniesieniu do wszystkich przykładów zawartych w zbiorze danych wejściowych.

Tab. 14.10: Lokaty modeli pod względem trafności diagnoz dotyczących stanu technicznego pompy P

L.P.	LOKATA SIECI STWIERDZEŃ		
	Warstwa W1	Warstwa W2	Warstwa W3
1	1	1	2
2	2	1	3
3	1	1	2
4	3	1	2
5	2	3	1
6	2	3	1
7	2	1	3
8	2	1	3
9	2	1	3
10	2	1	3
11	2	1	3
12	2	1	3

Na tej podstawie możliwe jest określenie prawdopodobieństw zajęcia określonego miejsca przez opracowane modele (tab. 14.11), a następnie obliczenie wartości oczekiwanych dla tak zdefiniowanych zmiennych dyskretnych, które w prezentowanym przykładzie wynoszą odpowiednio $\left[1\frac{11}{12}, 1\frac{1}{3}, 2\frac{5}{12}\right]$.

Tab. 14.11: Prawdopodobieństwa zajęcia określonej lokaty przez opracowane sieci

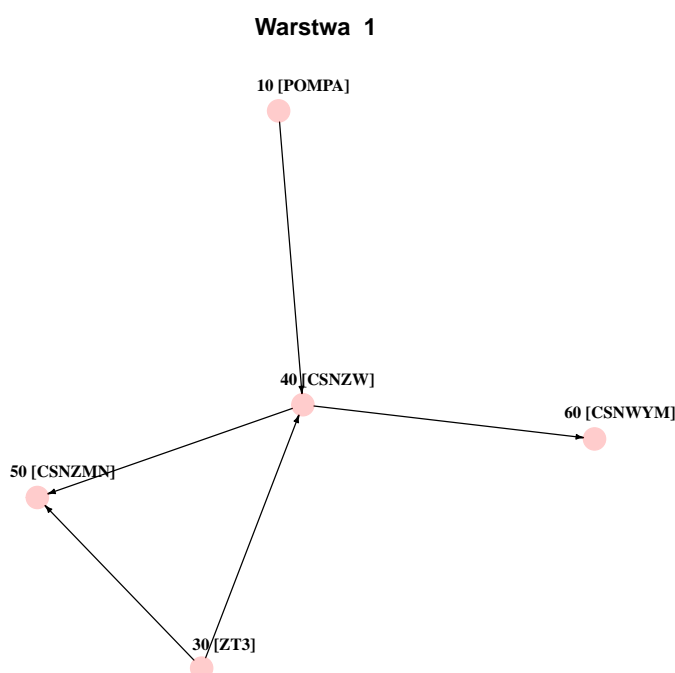
MIEJSCE	PRAWDOPODOBIENSTWA DLA SIECI STWIERDZEŃ		
	Warstwa W1	Warstwa W2	Warstwa W3
1	$\frac{1}{6}$	$\frac{5}{6}$	$\frac{1}{6}$
2	$\frac{3}{4}$	0	$\frac{1}{4}$
3	$\frac{1}{12}$	$\frac{1}{6}$	$\frac{7}{12}$

W tab. 14.12 przedstawiono obliczone wartości oczekiwane zajęcia przez sieci z kolejnych warstw określonego miejsca po przeprowadzeniu analogicznych działań w odniesieniu do kanału pomiędzy zbiornikami T1 i T3 oraz zbiornika T3.

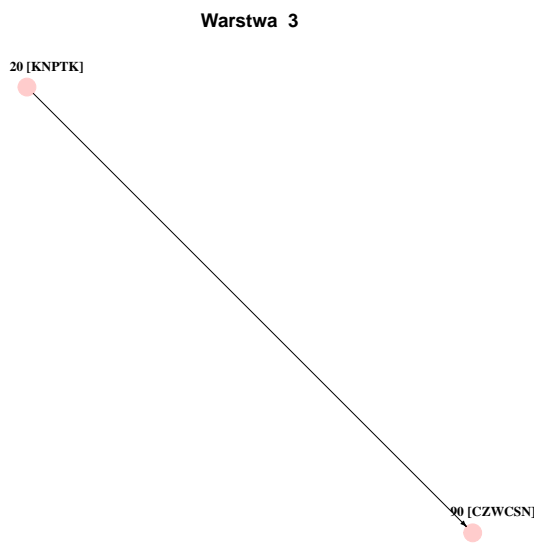
Tab. 14.12: Wartości oczekiwane zajęcia danego miejsca przez opracowane sieci ze względu na trafność określania wartości rozpatrywanych stwierdzeń

TREŚĆ STWIERDZENIA	WARTOŚĆ OCZEKIWANA MIEJSCA SIECI STWIERDZEŃ		
	Warstwa W1	Warstwa W2	Warstwa W3
Pompa działa prawidłowo	$1\frac{11}{12}$	$1\frac{1}{3}$	$2\frac{5}{12}$
Kanał pomiędzy zbiornikami T1 i T3 jest przytkany	$2\frac{1}{3}$	2	$1\frac{2}{3}$
Zbiornik T3 jest szczelny	$2\frac{1}{12}$	$1\frac{2}{3}$	$2\frac{1}{4}$

Na podstawie analizy otrzymanych wyników zdecydowano, że z sieci stwierdzeń znajdującej się w warstwie 1 usunięty zostanie węzeł dotyczący kanału pomiędzy zbiornikami T1 i T3 a z sieci z warstwy 3 stwierdzenia dotyczące zarówno pompy P jak i zbiornika T3. Na rys. 14.10 i 14.11 przedstawiono nowe struktury sieci stwierdzeń w tych warstwach, które uzyskano dzięki zastosowaniu algorytmu K2. Tak samo jak poprzednio wartości prawdopodobieństw warunkowych w tych sieciach określono za pomocą algorytmu z grupy Junction Tree.



Rys. 14.10: Zmodyfikowana struktura sieci stwierdzeń w warstwie 1



Rys. 14.11: Zmodyfikowana struktura sieci stwierżeń w warstwie 3

Biorąc pod uwagę fakt, że sieć z warstwy 3 umożliwia wyłącznie diagnozę stanu technicznego kanału pomiędzy zbiornikami, przy czym jak wynika z tabeli 14.12 diagnoza ta jest statystycznie bardziej pewna niż dla sieci z warstwy 2, zatem sieć z warstwy 3 musi mieć największą wagę.

Waga sieci z warstwy 2 może zostać określona procentowo w odniesieniu do wagi sieci z warstwy 3 jako stosunek odpowiednich wartości oczekiwanych, tzn. $1\frac{2}{3} : 2 = \frac{5}{6}$. Podobnie waga sieci z warstwy 1 może zostać określona procentowo w odniesieniu do wagi sieci z warstwy 2 jako średnia arytmetyczna stosunku odpowiednich wartości oczekiwanych, tzn. $(1\frac{1}{3} : 1\frac{11}{12} + 1\frac{2}{3} : 2\frac{1}{12}) / 2$

Przyjmując dla sieci z warstwy 3 wagę równą 1 otrzymuje się wagi dla sieci z warstw 2 i 1 równe odpowiednio $\frac{5}{6}$ oraz $\frac{43}{69}$. Po przeprowadzeniu weryfikacji zmodyfikowanego diagnostycznego systemu doradczego otrzymano znacznie lepsze wyniki niż poprzednio. Zostały one przedstawione w tab. 14.13. Podobnie jako poprzednio wyniki zostały zaokrąglone „w dół” a błędy „w górę”.

Tab. 14.13: Trafność diagnoz formułowanych przez zmodyfikowany system diagnostyczny dla rozpatrywanego symulatora procesu

N	TRAFNOŚĆ DIAGNOZY, %			B1, %	B2, %
	Pompa P	Kanał T1T3	Zbiornik T3		
0	100	100	100	0	0
1	75	50	100	17	9
2	80	0	60	47	7
3	100	0	0	67	0

14.9. Wnioski

Wyniki uzyskiwane z opracowanego diagnostycznego systemu doradczego dla prezentowanego symulatora procesu są zadowalające. Skonstruowany model dobrze rozpoznaje proste stany techniczne i niektóre stany złożone. Wraz ze wzrostem liczby niesprawnych elementów obiektu trafność diagnozy maleje, szczególnie w odniesieniu do kanału pomiędzy zbiornikami T1 i T3. Zmniejszona trafność diagnozy dotycząca złożonych stanów technicznych obiektu wynika z faktu występowania identycznych symptomów jak dla określonych stanów prostych. Wykrycie przytkania kanału lub nieszczelności zbiornika przy niesprawnej pompie jest praktycznie niemożliwe na podstawie dostępnych sygnałów diagnostycznych. Oznacza to istotny wzrost wartości błędu **B1**, tzn. niewykrycia istniejącego uszkodzenia.

Obecnie nie jest znana ogólna metoda, która umożliwiłaby rozróżnianie takich stanów. Najczęściej formułowana diagnoza powinna wskazywać wystąpienie najbardziej prawdopodobnego stanu technicznego, a więc stanu, w którym niesprawny lub uszkodzony jest tylko 1 element obiektu.

Przedstawiony przykład konstruowania sieci wielowarstwowej za pomocą pakietu MMNET ukazuje w wyraźny sposób jak ważne jest poprawne zrozumienie zasady działania tego typu sieci. Konstruowanie modelu w sposób rutynowy może prowadzić do wyników, które będą znacznie odbiegały od oczekiwań. Chcąc uniknąć wystąpienia takiej sytuacji należy bardzo starannie rozważyć, które stwierdzenia powinny znaleźć się w danej warstwie. Jak dowodzi prezentowany przykład nie zawsze korzystne jest uwzględnianie wszystkich stwierdzeń dotyczących elementów obiektu w każdej warstwie.

Bibliografia

- [14.1] Bell Alex. E. Death by uml fever. *Queue*, 2(1):72–80, March 2004.
- [14.2] Bruegge Bernd, Dotoit Allen H. *Object-oriented software engineering. Using UML, patterns, and Java (3rd Edition)*. Pearson Education, Inc., NJ, 2010.
- [14.3] FESTO. Stanowisko PCS. Dokumentacja techniczna.
- [14.4] Gelman Andrew, Carlin John B., Stern Hal S., Rubin Donald B. *Bayesian Data Analysis*. Chapman & Hall/CRC, Boca Raton, Florida, 2004.
- [14.5] Huang Cecil, Darwiche Adnan. Inference in belief networks: A procedural guide. *International Journal of Approximate Reasoning*, 15(3):225–263, 1996.
- [14.6] Lauritzen Steffen L., Spiegelhalter David J. Local computations with probabilities on graphical structures and their application to expert systems. *Journal of the Royal Statistical Society*, 50(2):157–224, 1988.
- [14.7] Madsen A. L., Nilsson D. Solving influence diagrams using hugin, shafer–shenoy and lazy propagation. *Proceedings of the Seventeenth Conference on Uncertainty in Artificial Intelligence*, strony 337–345, 2001.
- [14.8] Miles Russ, Hamilton Kim. *Learning UML 2.0*. O'Reilly Media, Inc., CA, 2006.
- [14.9] Murphy Kevin. <http://code.google.com/p/bnt/>, luty 2010.
- [14.10] Skupnik Damian. *Wieloaspektowe modele diagnostyczne*. Zeszyt Katedry Podstaw Konstrukcji Maszyn nr 140. Wydawnictwo Politechniki Śląskiej, Gliwice, 2009.
- [14.11] Śmiałek Michał. *Zrozumieć UML 2.0. Metody modelowania obiektowego*. Wydawnictwo HELION, Gliwice, 2005.
- [14.12] Wrycza Stanisław, Marcinkowski Bartosz, Wyrzykowski Krzysztof. *Język UML 2.0 w modelowaniu systemów informatycznych*. Wydawnictwo HELION, Gliwice, 2005.