

**Politechnika Śląska  
Wydział Mechaniczny Technologiczny  
Katedra Podstaw Konstrukcji Maszyn**

**Krzysztof Psiuk**

**Obiektowe modele  
procesu propagacji stanów  
w układach maszyn**

**Gliwice 2001**

*Recenzenci*

Prof. dr hab. inż. Józef Korbicz - Politechnika Zielonogórska

Prof. dr hab. inż. Jan Kosmol - Politechnika Śląska

*Redaktor*

Wojciech Cholewa

*Redaktor językowy*

Ewa Opoka

*Redaktor techniczny*

Marek Wyleżoł

*Projekt okładki*

Wojciech Cholewa, Marek Wyleżoł

**ISBN 83-906533-7-0**

*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://kpkmt.polsl.gliwice.pl>

---

Druk i oprawa (z dostarczonych materiałów)

Drukarnia cyfrowa D&D sp. z o.o. ul. Moniuszki 6, 44-100 Gliwice

## **Od autora**

Niniejszy zeszyt opracowałem na podstawie rozprawy doktorskiej, wykonanej pod kierunkiem prof. dr hab. inż. Wojciecha Cholewy. Publiczna obrona odbyła się 17 lipca 2001 roku przed komisją powołaną przez Radę Wydziału Mechanicznego Technologicznego. W opracowaniu uwzględniłem uwagi recenzentów pracy doktorskiej: prof. dr hab. inż. Józefa Korbicza oraz prof. dr hab. inż. Jana Kosmola.

Bardzo dziękuję szanownym recenzentom za wszystkie cenne uwagi. Pozwoliły mi one na wydanie niniejszego zeszytu w postaci pozbawionej błędów, których nie ustrzegłem się podczas pisania pracy doktorskiej.

Składam serdeczne podziękowania promotorowi prof. Wojciechowi Cholewie za cierpliwość i wyrozumiałość okazaną mi podczas wykonywania mojej pracy.

Dziękuję również Koleżankom i Kolegom z Katedry Podstaw Konstrukcji Maszyn za okazaną mi pomoc i życzliwość podczas wykonywania pracy.

Gliwice, lipiec 2001

Krzysztof Psiuk



# Spis treści

<b>OD AUTORA.....</b>	<b>3</b>
<b>1. WSTĘP.....</b>	<b>11</b>
1.1. CEL PRACY.....	12
1.2. TEZY PRACY.....	13
<b>2. PRZEDMIOT BADAŃ.....</b>	<b>15</b>
2.1. NIEZAWODNOŚĆ OBIEKTÓW TECHNICZNYCH.....	15
2.2. METODY REPREZENTACJI STANÓW I ZMIAN STANÓW.....	18
2.2.1. Zbiory stwierdzeń.....	19
2.2.2. Macierzowa metoda badania stanu diagnostycznego.....	22
2.2.3. Modele matematyczne bazujące na rozkładach zmiennych losowych.....	24
2.2.4. Sieci Petriego.....	25
2.2.5. Sieci przekonań (Bayesa).....	27
2.2.6. Sieci Pearla.....	28
2.2.7. Modele obiektowe.....	30
2.2.8. Automat skończony.....	32
2.2.9. Systemy wieloagentowe.....	36
2.2.10. Drzewa uszkodzeń.....	40
2.2.11. Podsumowanie.....	42
<b>3. MODEL PROCESU PROPAGACJI STANÓW.....</b>	<b>43</b>
3.1. MODEL I MODELOWANIE.....	43
3.2. MODEL UKŁADU.....	46
3.3. STOPNIE SZCZEGÓŁOWOŚCI MODELU.....	48
3.4. WARSTWY MODELU.....	49
3.5. OGÓLNA SYNTEZA MODELU.....	51
3.6. MODUŁ PODSTAWOWY.....	52
3.7. LOKALNA BAZA WIEDZY.....	57
3.8. REPREZENTACJA STANÓW ELEMENTÓW UKŁADU.....	60

<b>4. IMPLEMENTACJA MODELU PROPAGACJI STANÓW .....</b>	<b>65</b>
4.1. MODELOWANIE STANÓW ZA POMOCĄ MATLAB STATEFLOW.....	65
4.2. ZAŁOŻENIA DO IMPLEMENTACJI MODELU .....	65
4.2.1. Elementy podstawowe programu Stateflow .....	65
4.2.2. Założenia .....	67
4.3. TABLICA OGŁOSZEŃ .....	68
4.4. BLOK WIELU AUTONOMICZNYCH MODUŁÓW.....	72
<b>5. MODEL PROPAGACJI STANÓW NA PRZYKŁADZIE WYBRANEGO UKŁADU.....</b>	<b>77</b>
5.1. OPIS MODELOWANEGO UKŁADU .....	77
5.2. ANALIZA UKŁADU.....	81
5.2.1. Opis struktury układu.....	81
5.2.2. Analiza reguł i stwierdzeń.....	82
5.2.3. Zapis reguł.....	89
5.3. MODEL UKŁADU .....	92
5.3.1. Główne elementy modelu .....	92
5.3.2. Prezentacja wyników .....	98
5.3.3. Elementy pomocnicze .....	100
<b>6. UWAGI KOŃCOWE I WNIOSKI .....</b>	<b>105</b>
6.1. PODSUMOWANIE .....	105
6.2. WNIOSKI WYNIKAJĄCE Z PRACY.....	106
6.3. KIERUNKI DALSZYCH BADAŃ.....	107
<b>LITERATURA.....</b>	<b>109</b>
<b>SPIS ILUSTRACJI.....</b>	<b>117</b>

## Wykaz ważniejszych oznaczeń

- $attr$  - atrybut stwierdzenia,
- $attr(x)$  - funkcja zwracająca nazwę cechy,
- $b$  - waga (stopień ważności stwierdzenia),
- $b(x)$  - funkcja zwracająca wartość miary akceptacji wypowiedzi o stwierdzeniu  $x$ ,
- $B$  - symbol pusty,
- $BEL(B)$  - dynamiczna wartość prawdopodobieństwa węzła  $B$ ,
- $C$  - zbiór cech stanu,
- $c$  - cecha stanu,
- $D$**  - zbiór sprawdzeń wymaganych do procesu lokalizacji uszkodzeń,
- $D_I$  - relacja incydencji (określa zbiór łuków skierowanych w sieci Petriego),
- $D_B^-$  - dana opisująca część sieci związaną z rodzicami węzła  $B$ ,
- $D_B^+$  - dana opisująca część sieci związaną z potomkami węzła  $B$ ,
- $E$**  - zbiór elementów podstawowych,
- $e$  - element,
- $F$  - zbiór stanów końcowych automatu skończonego,
- $G$  - funkcja incydencji wejściowych (określa zbiór łuków skierowany od miejsc do przejść w sieci Petriego),
- $H$  - funkcja incydencji wyjściowych (określa zbiór łuków skierowany od przejść do miejsc w sieci Petriego),
- $K$  - funkcja pojemności miejsc (określa maksymalną liczbę znaczników, jakie może posiadać miejsce w sieci Petriego),
- $M_0$  - funkcja znakowania (markowania) początkowego, w sieci Petriego,
- $O$**  - obiekt,
- $obj(x)$  - funkcja zwracająca nazwę obiektu,
- $P_I$  - niepusty i skończony zbiór miejsc (stanów), w sieci Petriego,
- $P(x_6|x_5)$  - prawdopodobieństwo warunkowe,
- $p(x)$  - prawdopodobieństwo zdarzenia  $x$ ,

$Q$	- skończony zbiór stanów automatu skończonego,
$q_0$	- stan początkowy automatu skończonego,
$q[t]$	- stan systemu w chwili $t$ ,
$q[0]$	- znany stan początkowy systemu,
$R$	- zbiór relacji opisanych na elementach układu,
$S$	- zbiór stanów,
$S_e$	- zbiór wszystkich stanów elementarnych,
$s_{ei}$	- stan elementarny,
$S_u$	- zbiór wszystkich możliwych stanów,
$s_u$	- stan układu,
$s_{u,t_i}$	- stan układu w chwili $t_i$ ,
$s_u(t)$	- stan układu w chwili $t$ ,
$T$	- niepusty i skończony zbiór przejść (zdarzeń), w sieci Petriego,
$T(x)$	- stopień prawdziwości stwierdzenia $x$ ,
$t$	- czas makro,
$V$	- zbiór wartości cech stanu,
$v$	- wartość cechy stanu,
$val$	- wartość stwierdzenia,
$val(x)$	- funkcja zwracająca wartość cechy,
$W$	- funkcja pojemności łuków (krotności krawędzi, określa liczbę przenoszonych znaczników w sieci Petriego),
$w$	- ocena stopnia prawdziwości lub stopnia przekonania o prawdziwości wypowiedzi,
$\underline{x}[t]$	- wejście systemu w chwili $t$ ,
$\underline{x}[t_1:t_2)$	- wejście systemu w przedziale czasu od chwili $t_1$ , do chwili bezpośrednio poprzedzającej chwilę $t_2$ ,
$\underline{y}[t]$	- wyjście systemu w chwili $t$ ,
$Z$	- zbiór sygnałów wyjściowych,
$\alpha$	- stała normalizująca,
$\beta$	- funkcja rozkładu kolorów znacznika na wejściowych pozycjach przejść sieci Petriego,



- $\delta$  - funkcja przejść automatu skończonego,
- $\Delta$  - alfabet wyjściowy automatu skończonego,
- $\Gamma$  - skończony zbiór dopuszczalnych symboli taśmowych,
- $\varepsilon(e_j)$  - stan elementu  $e_j$ ,
- $\mu_0$  - początkowe oznakowanie sieci Petriego,
- $\Sigma$  - skończony zbiór akcji automatu skończonego,
- $\tau$  - czas mikro,
- $\psi$  - funkcja rozkładu kolorów znacznika na wyjściowych pozycjach przejść sieci Petriego.



## 1. Wstęp

Jednym z głównych zadań procesu eksploatacji dużych obiektów technicznych jest bieżąca ocena ich stanu technicznego, pozwalająca na podejmowanie decyzji o warunkach ich dalszej eksploatacji oraz zakresie ewentualnych prac remontowych. Ze względu na znaczny stopień skomplikowania tego typu obiektów, jak również coraz większą wiedzę na temat ich projektowania, konstruowania i przede wszystkim eksploatacji, konieczne staje się określenie sposobu zapisu informacji o ich działaniu. W tym celu wykorzystuje się układy monitorowania stanu obiektów. Układy takie zbierają dane dotyczące działania obiektu, wstępnie je przetwarzają i umożliwiają przeprowadzenie analiz pozwalających na podjęcie decyzji dotyczących dalszego eksploataowania obiektu. Ponieważ liczba takich danych jest bardzo duża, coraz częściej czynione są próby wykorzystania systemów doradczych w układach monitorowania stanu maszyn. Systemy doradcze mają pomóc w użytkowaniu maszyny i obniżyć koszty związane z jej eksploatacją. Jednym z głównych elementów systemów doradczych jest odpowiednio opracowana baza wiedzy. Wiedzę w bazach wiedzy systemów doradczych można zapisywać w różny sposób. Wprowadzono kilka technik reprezentacji wiedzy [9] [25] [15] [52] [51] [53]:

- tablice decyzyjne,
- zbiory stwierdzeń,
- zbiory reguł,
- obiekty abstrakcyjne,
- sieci semantyczne,
- ramy,
- scenariusze,
- techniki bazujące na bezpośrednim zastosowaniu logiki (rachunek predykatów),
- sieci neuronalne.

Bardzo często wiedza w diagnostycznych systemach doradczych zapisywana jest w postaci reguł. Podstawą poprawnego działania opracowanego systemu doradczego jest prawidłowe określenie i zdefiniowanie reguł diagnostycznych opisujących zachowanie zbioru maszyn, dla którego system został opracowany. Reguły diagnostyczne mogą być pozyskiwane w różny sposób. Jednym ze sposobów pozyskiwania reguł diagnostycznych, a także sprawdzenia istniejących, znanych reguł, może być modelowanie zmian stanów technicznych, występujących w układzie.

Modelowanie takie pozwala na sprawdzenie poprawności reguł już istniejących, poszukiwanie nowych zależności oraz opracowanie na ich podstawie nowych reguł.

Innym ważnym i ciekawym zagadnieniem jest sposób przedstawienia propagacji stanów w układzie. Obserwując zmiany stanów elementów układu, można zauważyć, że zmiana stanu jednego elementu układu może wpływać na stan innego elementu tego samego układu, pomimo tego, że elementy te nie są ze sobą bezpośrednio połączone.

Badania opisane w tej pracy przeprowadzono częściowo w ramach projektów KBN 7T07B 045 16 oraz PBZ-038-06.

## **1.1. Cel pracy**

Zagadnienia związane z modelowaniem zmian stanów w rzeczywistych układach nie są nowe. Liczne znane rozwiązania opierają się głównie na elementach dwustanowych. Istnieje wiele teorii opisujących zachowanie się takich układów, np. teoria maszyn o skończonej liczbie stanów lub sieci Petriego. W większości wypadków jednak wymienione teorie są stosowane w automatyce, elektronice i informatyce, gdzie stany modelowanych obiektów można rozpatrywać dwuwartościowo.

Celem niniejszej pracy jest zbudowanie modelu pozwalającego na opisanie procesu propagacji stanów elementów układu, ze szczególnym uwzględnieniem propagacji zmian stanów w modelowanym układzie. Model uwzględnia nie tylko zmiany stanu układu, ale również opis struktury układu, sieciowe połączenia między elementami modelowanego układu oraz wielowarstwowy zapis relacji występujących pomiędzy elementami układu. Struktura układu rozumiana jest w niniejszym opracowaniu jako sposób wzajemnego przyporządkowania elementów składowych i połączenia ich w pewną całość.

Poszukiwania modelu opisu stanów układu prowadzone są z uwzględnieniem różnych aspektów przeznaczenia opracowanego modelu. Przeprowadzono identyfikację potrzeby, a także dokonano przeglądu istniejących metod realizacji. Podjęto decyzję o sposobach zapisu danych o modelowanym układzie, a także o sposobie reprezentowania niesprawności, ze szczególnym uwzględnieniem zmian stanów elementów układu oraz propagacji tych stanów w układzie. Przeprowadzone badania zmierzają do opisu stanów w sposób bardziej rozszerzony, nie ograniczając się tylko do stanów dwuwartościowych. W wyniku przeprowadzonych badań opracowano odpowiedni model układu, opisujący zmiany stanów oraz ich propagację (wpływ tych zmian na zmiany stanów innych elementów układu).

## 1.2. Tezy pracy

1. Propagację stanów oraz zmian stanów elementów maszyn można modelować za pomocą wielowarstwowych modeli sieciowych, o zmiennym stopniu szczegółowości, takich, które posiadają aktywne węzły.
2. Wszystkie węzły takiej sieci mogą być rozpatrywane jako autonomiczne moduły, oparte na jednej ogólnej, nadrzędnej klasie modułów (węzłów).

Opracowanie modelu w postaci sieci z aktywnymi węzłami umożliwi rozpatrywanie oddziaływań występujących pomiędzy poszczególnymi elementami obiektu. Wprowadzenie sieci wielowarstwowej pozwoli na uproszczenie opisu tych oddziaływań, każda z warstw sieci może bowiem opisywać oddziaływania określonego typu, co uprości proces szukania określonych zależności pomiędzy elementami. Zmienny stopień szczegółowości modelu umożliwi rozpatrywanie stanów elementów oraz propagacji stanów tych elementów dla modeli o zróżnicowanym stopniu złożoności.

Przyjęcie prawdziwości tezy 1. oznacza, że węzły takiej sieci, będące węzłami aktywnymi, oddziałują na inne węzły w tej samej sieci, a dla każdego z aktywnych węzłów sieci należy zdefiniować zakres wykonywanych działań i sposób wzajemnego oddziaływania na inne węzły.

Duża różnorodność typów węzłów takiej sieci może prowadzić do zbyt wielkiej komplikacji modelu. Dlatego też teza 2. mówi o tym, iż możliwe jest ograniczenie tej różnorodności przez zdefiniowanie jednej nadrzędnej klasy aktywnych węzłów (nazwanych autonomicznymi modułami), takiej, która będzie stanowiła podstawę do opracowania węzłów sieci.

Modelowanie propagacji stanów związane jest bezpośrednio z wiedzą, jaką posiada się na temat działania modelowanego obiektu oraz z dostępną wiedzą diagnostyczną. Proponuje się, aby wiedza ta była reprezentowana w postaci odpowiednich widoków opracowanej sieci, stanowiąc jej wybrane fragmenty. Takie rozwiązanie umożliwia podział całości rozpatrywanych zagadnień, często bardzo złożonych, na mniejsze zagadnienia, związane wyłącznie z wybranymi aspektami naszej wiedzy.



## 2. Przedmiot badań

### 2.1. niezawodność obiektów technicznych

Pojęcie „niezawodność” definiowane jest w różny sposób w różnych dziedzinach nauki. W literaturze technicznej i medycznej pojęcie to rozumiane jest ogólnie jako właściwość rozpatrywanego obiektu określona przez jego zdolność do wykonania stawianych mu zadań [38]. Za miarę niezawodności obiektu, który ma do wykonania określone zadanie, przyjmuje się prawdopodobieństwo wykonania tego zadania.

Podstawowe terminy i pojęcia w teorii niezawodności rozpatrywane są z reguły w odniesieniu do określonych warunków eksploatacji. Poniżej przedstawiono kilka takich pojęć [13] [48] [70] [95]. Wyróżniono podgrupy opisujące właściwości, stany oraz uszkodzenia obiektu.

#### 1. Właściwości obiektu

- a) niezawodność – właściwość obiektu charakteryzująca jego zdolności do działania zgodnie z przeznaczeniem, czyli do wykonywania określonych zadań, w określonych warunkach i w określonym przedziale czasu eksploatacji.

Niezawodność jest właściwością kompleksową, obejmującą, w zależności od przeznaczenia obiektu i warunków jego eksploatacji, takie właściwości, jak: nieuszkodzalność, trwałość, przechowywalność, naprawialność.

#### 2. Stany obiektu

- a) stan niezawodnościowy – stan obiektu opisany na zbiorze jego stanów fizycznych (np. głębokość bieżnika opony koła samochodu osobowego jest większa od 4 [mm] oraz ciśnienie w kole jest większe od 2 [ba], to koło jest w stanie zdatnym do dalszej eksploatacji);
- b) stan zdatności – stan niezawodnościowy, w którym obiekt może działać zgodnie z przeznaczeniem. W celu rozwiązania praktycznych zadań diagnostyki technicznej rozróżnia się dwa pojęcia: stan zdatności chwilowej i stan zdatności zadaniowej;
- c) stan zdatności chwilowej – stan obiektu, w którym jest on zdolny do realizowania wymaganego zadania w sposób zgodny z wymaganiami dokumentacji normatywno-technicznej przy określonym oddziaływaniu otoczenia;

- d) stan zdadności zadaniowej – stan, w którym obiekt jest zdolny do zrealizowania wymaganego zadania w wyznaczonych warunkach eksploatacyjnych, w losowo zmiennym przedziale czasu;
- e) stan niezdatności – stan niezawodnościowy, w którym obiekt nie jest zdolny do działania zgodnie z przeznaczeniem;

### 3. Uszkodzenia obiektu

- a) uszkodzenie – zdarzenie polegające na utracie przez obiekt zdolności do działania zgodnie z przeznaczeniem;
- b) niesprawność – jako zdarzenie – zdarzenie (lub ciąg zdarzeń) polegające na niekorzystnej zmianie stanu obiektu zmniejszającej jego zdolność do działania zgodnie z przeznaczeniem; przyczyną niesprawności może być np. zużycie lub niewłaściwa regulacja obiektu;
- c) niesprawność – jako stan techniczny – stan obiektu, w którym nie spełnia on chociażby jednego z wymagań dokumentacji technicznej, jest jednak zdolny do wykonania, z ograniczeniem, wyznaczonych mu funkcji. W praktyce rozróżniamy niesprawności powodujące uszkodzenia oraz takie, które ich nie powodują.

W teorii niezawodności uszkodzenia obiektu mogą powstawać z różnych przyczyn. Dotyczy to następujących procesów [37]:

#### 1. Proces konstruowania

Błędy konstrukcyjne mogą wynikać z ograniczonej wiedzy konstruktora lub wynikać z tego, iż nie uwzględniono lub nie przewidziano pewnych czynników, które mają wpływ na działanie obiektu.

#### 2. Proces wytwarzania

Wady technologiczne są skutkiem losowości procesu wytwarzania lub występujących w nim błędów. Jeżeli błędy te są „małe” i mają niewielki wpływ na zdolność obiektu do działań zgodnych z przeznaczeniem, to wpływ tego czynnika na niezawodność obiektu można pominąć.

#### 3. Proces magazynowania

W wyniku złego magazynowania obiektów mogą one ulec uszkodzeniu lub przyspieszonemu zużyciu się. Ponadto nawet wtedy, gdy przechowywanie obiektów jest zgodne z zaleceniami, ulegają one procesowi starzenia. Zmieniają się własności tworzyw, z których wykonane są elementy.

#### 4. Proces eksploatacji

Błędy eksploatacji wynikają z nieprzestrzegania zaleceń dotyczących użytkowania obiektu. Nieprzestrzeganie zasad eksploatacji oraz



losowość procesu eksploatacji może prowadzić do przyspieszonego zużywania się obiektu.

#### 5. Starzenie się (zużywanie) obiektu

Niezależnie od jakości wykonania obiektu, jego części i podzespołów, całość podlega procesowi zużywania się. Zarówno w czasie eksploatacji jak i magazynowania w stosowanych tworzywach zachodzą nieodwracalne zmiany prowadzące do pogorszenia wytrzymałości oraz zdolności współdziałania poszczególnych części i podzespołów, co w rezultacie może prowadzić do powstawania niesprawności i uszkodzeń.

Niesprawności mogą być opisywane przez różne modele procesów ich powstawania. Podzielono je na następujące grupy [37] [38]:

- a) niesprawności spowodowane oddziaływaniami skokowymi (udarowymi, pojedynczymi),
- b) niesprawności rozwijające się (stopniowe), spowodowane oddziaływaniami kumulującymi się,
- c) niesprawności spowodowane oddziaływaniami relaksacyjnymi.

Niesprawności spowodowane działaniem bodźców skokowych to niesprawności występujące w sposób niesygnalizowany (nagły). Do tego typu uszkodzeń można zaliczyć np. złamanie zęba w uzębieniu koła przekładni zębatej spowodowane przypadkowym przeciążeniem zewnętrznym. Charakterystyczne dla tego typu niesprawności jest to, że pojawiają się one niejednokrotnie niezależnie od tego, jak długo dany obiekt znajdował się w eksploatacji oraz w jakim stanie znajdował się wcześniej.

Niesprawności rozwijające się (stopniowe) są rezultatem występowania procesów stopniowego zużywania się obiektu. Niesprawności te są zazwyczaj sygnalizowane i mogą być wcześniej rozpoznane. Często ustalane są dopuszczalne warunki eksploatacji, których przekroczenie jest jednoznaczne z wystąpieniem tego typu niesprawności. Przykładem tego typu niesprawności może być utrata sprawności hamulców w samochodzie na skutek starcia oklein na klockach hamulcowych.

Niesprawności spowodowane działaniem bodźców relaksacyjnych powstają wtedy, gdy zmiana stanu obiektu odbywa się w sposób skokowy i wynika z działania bodźców kumulujących się. Jako przykład może posłużyć współdziałanie wodzika z prowadnicą. Ich zużycie prowadzi do powstania nadmiernych luzów, co przy niesprzyjających warunkach zewnętrznych, może prowadzić do zaklinowania. Przy tego typu niesprawnościach stopniowe starzenie się obiektu i jego nagłe przejście w stan niesprawności są z sobą powiązane. Za przykład może posłużyć niesprawność obiektu elektronicznego z rezerwą. Niesprawności poszczególnych elementów

rezerwowych obiektu pociągają za sobą nadmierne obciążenie elementów pozostałych ([37]).

Bardzo często niesprawności nie powstają w wyniku pojedynczego oddziaływania, ale kilku oddziaływań występujących jednocześnie. W takim przypadku należy rozróżnić przyczyny działające niezależnie i przyczyny zależne od siebie. Przykładem przyczyn działających niezależnie może być uszkodzenie opony. Opona może ulec uszkodzeniu na skutek przebicia ostrymi przedmiotami lub wskutek nadmiernego zużycia. Wówczas, gdy mamy do czynienia z wieloma czynnikami wywołującymi niesprawności, często jedna lub dwie są przyczynami dominującymi. Wpływ pozostałych przyczyn jest tak mały, że można je pominąć. Dlatego też dąży się wtedy do wykrycia przyczyn decydujących o powstawaniu niesprawności, a następnie uwzględnia się wpływ pozostałych przyczyn. Inaczej postępuje się wówczas, gdy uwzględnia się przyczyny zależne. Bierze się pod uwagę bardzo dużą liczbę różnych powiązań występujących pomiędzy poszczególnymi przyczynami niesprawności. Uwzględnienie wszystkich powiązań może spowodować, że model opisujący ich występowanie stanie się bardzo skomplikowany. Dlatego czasami, gdy oddziaływania te są silnie powiązane, przyjmuje się, że występuje tylko jedna przyczyna, oraz że tylko ona jest powodem niesprawności.

Nie zawsze jednak takie uproszczenie można przyjąć. Dla pewnych obiektów lub warunków działania, czy też przeznaczenia obiektu badań, przyjęcie takiego założenia może być niemożliwe. Wymaga się w takim wypadku uwzględnienia wielu różnych czynników mających wpływ na wystąpienie danej niesprawności. Takie działania mogą być podyktowane ważnością przeznaczenia obiektu. Oznacza to, że potrzebne jest opracowanie sposobu uwzględniania wielu współwystępujących, różnych czynników, które mogą mieć wpływ na wystąpienie niesprawności obiektu. Szczególną uwagę należy zwrócić na przyczyny wzajemnie zależne.

## **2.2. Metody reprezentacji stanów i zmian stanów**

Sposoby reprezentacji stanów poszczególnych elementów złożonych układów maszyn zostały częściowo opracowane. Należy jednak wyraźnie zaznaczyć, że wiele z opracowanych metod uwzględnia wyłącznie stany występujące w układach rozpatrywanych statycznie, natomiast nie zwraca się uwagi na występujące zmiany stanów. W wielu przypadkach, równie ważna jak informacja o aktualnym stanie danego układu, jest informacja o tym, czy stan ten uległ zmianie, czy też nie zmienia się od pewnego czasu. Przy rozpatrywaniu stanów technicznych układów maszyn zagadnienie określenia stanu nie powinno sprowadzać się jedynie do prostego stwierdzenia typu „element sprawny” – „element niesprawny”, „działa” – „nie działa” itp. W zdecydowanej większości wypadków stan danego elementu,

czy też całego układu, nie może być określany jedynie dwuwartościowo. Pomędzy granicznymi stanami, np. sprawny-niesprawny, istnieje wiele stanów pośrednich, w których może znajdować się dany element lub układ elementów. Nowym zagadnieniem, które należy w takim wypadku rozwiązać, jest odpowiedni zapis zależności, jakie występują pomiędzy poszczególnymi stanami granicznymi i wpływającymi na możliwość przechodzenia obiektu z aktualnego stanu w inny.

Poszukiwanie rozwiązania tak postawionego zadania związane jest z możliwościami wykorzystania tego rozwiązania w różnych dziedzinach. Opracowanie odpowiedniego modelu opisującego zmiany stanów wielowartościowych i przygotowanie na tej podstawie narzędzia, umożliwiającego obserwację zmian stanów pozwoli na zastosowanie go w rzeczywistych systemach monitorujących stan i działanie dużych zespołów maszyn, a zwłaszcza w zastosowanych tam układach wspomaganie diagnostycznego oraz systemach doradczych.

W literaturze omawiane są różne metody i techniki reprezentacji stanów układu. Spośród wielu na uwagę zasługuje kilka:

- zbiory stwierdzeń [18][19],
- macierzowa metoda badania stanu diagnostycznego [59][61][62][60],
- modele matematyczne bazujące na rozkładach zmiennych losowych [38][37][6],
- sieci Petriego [7][36],
- automaty skończone [8][30][33][84],
- sieci przekonań [69],
- modele obiektowe [7][90],
- drzewa uszkodzeń [63][64].

Poniżej przedstawiono krótką analizę tych metod pod kątem zastosowania ich do opisu stanów, zmian stanów oraz wpływu zmian stanu jednego elementu na inne elementy znajdujące się w rozpatrywanym układzie.

### 2.2.1. Zbiory stwierdzeń

Zbiory stwierdzeń bazują na pojęciu stwierdzenia [19][22][14]. *Stwierdzenie* jest informacją o uznaniu wypowiedzi orzekającej o obserwowanych faktach lub reprezentującą określoną opinię [19]. Stwierdzenie może być reprezentowane jako następująca szóstka uporządkowana [19]:

$$\langle obj, attr, val, w, t, b \rangle \quad (1)$$

gdzie:

- $obj, attr, val$  - treść stwierdzenia, czyli wypowiedź o tym, że obiektowi  $obj$  przysługuje atrybut  $attr$  o wartości  $val$ ,
- $t$  - czas, w którym jest rozpatrywany obiekt  $obj$  (punkt lub odcinek czasu),
- $b$  - ocena stopnia prawdziwości lub stopnia przekonania o prawdziwości wypowiedzi  $obj, attr, val$  w czasie  $t$ ,
- $w$  - waga (stopień ważności) stwierdzenia.

Treścią stwierdzenia jest wypowiedź orzekająca o obserwowanych faktach lub reprezentująca określoną opinię. Dla stwierdzenia  $x$  wypowiedź taka jest reprezentowana przez następujące elementy [19]:

$$\langle obj(x), attr(x), val(x) \rangle \quad (2)$$

gdzie:

- $obj(x)$  - funkcja zwracająca nazwę obiektu,
- $attr(x)$  - funkcja zwracająca nazwę cechy,
- $val(x)$  - funkcja zwracająca wartość cechy.

Miarą akceptacji stwierdzenia  $x$  jest funkcja  $b(x)$ . Wartość  $b(x)$  może być definiowana na różne sposoby. Najprostszym sposobem definiowania wartości miary akceptacji stwierdzenia  $x$  jest zastosowanie stałych logicznych  $TAK, NIE$  [19]:

$$b(x) \in \{TAK, NIE\} \quad (3)$$

Innym rozwiązaniem jest zastosowanie rachunku prawdopodobieństwa, a szczególnie modelu Bayesa (prawdopodobieństwa warunkowego). W takim wypadku przyjmuje się założenie, że wartością stwierdzenia  $x$  jest prawdopodobieństwo  $p(x)$ , prawdziwości wypowiedzi odpowiadającej treści tego stwierdzenia [19]:

$$b(x) = p[\langle obj(x), attr(x), val(x) \rangle] \in [0,1] \quad (4)$$

Takie podejście może zostać uogólnione po przyjęciu założenia, że wartością stwierdzenia  $x$  jest jego stopień prawdziwości  $T(x)$ . Stopień prawdziwości  $T(x)$  jest liczbą rzeczywistą z przedziału  $[0,1]$ .

Ciekawym rozwinięciem koncepcji bazującej na stopniu prawdziwości, jest wprowadzenie stopni możliwości  $p(x)$  i konieczności  $n(x)$ , stwierdzenia  $x$ . Wartości tych miar

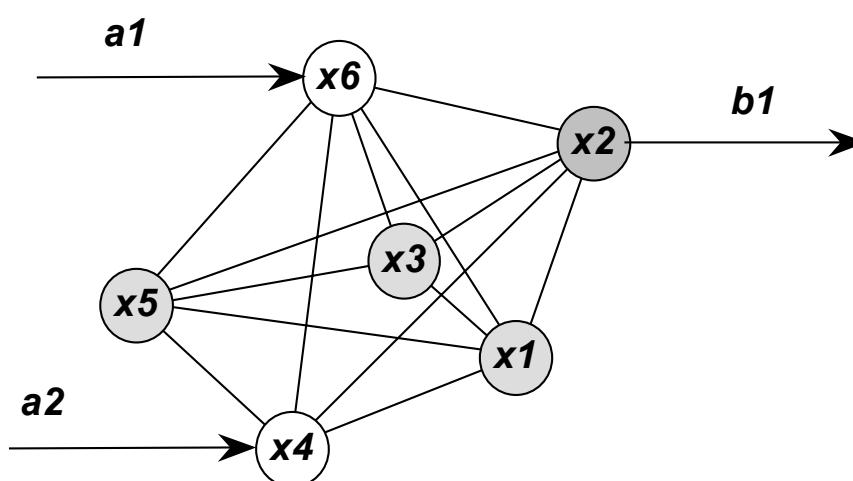
$$[n(x), p(x)] \subset [0,1] \quad (5)$$

wyznaczają hipotetyczny przedział, w którym znajduje się nieznana wartość stopnia prawdziwości  $T(x)$ .

Modele opierające się na pojęciu stwierdzenia znalazły zastosowanie między innymi w dynamicznych systemach doradczych do ciągłego lub cyklicznego interpretowania danych. Ciekawe rozwiązanie, w którym wykorzystano zbiory stwierdzeń, zaproponowano w systemie doradczym DT3D100 [14][17][28], będącym częścią systemu diagnostycznego dla turbozespołów energetycznych DT200-1 [23].

W systemie tym wprowadzono oryginalne rozwiązanie polegające na zastosowaniu stwierdzeń aktywnych reprezentujących jakąś opinię o badanym obiekcie i łączących w sobie koncepcje stwierdzeń, agentów oraz ram.

Stwierdzenia aktywne mogą występować jako powiązane i współdziałające węzły sieci, które ulegają stałym modyfikacjom w związku z dążeniem całej sieci do stanu równowagi. Każdy węzeł takiej sieci może być modyfikowany bezpośrednio przez zadania zewnętrzne (np. procesy realizowane przez układy pomiarowe), jak również może być dostępny dla zewnętrznych obserwatorów (np. użytkowników systemu doradczego). Przyjęto, że zmiana wartości stwierdzenia może pociągnąć za sobą zmianę wartości innych stwierdzeń (rys. 1), zgodnie ze stosowanym zbiorem reguł diagnostycznych. Sposób reprezentowania (zapisywania) takich reguł



Rys. 1. Sieć stwierdzeń dynamicznych:

$a^*$  - oddziaływania otoczenia,  $b^*$  - do użytkownika,  $x^*$  - stwierdzenia dynamiczne [19]

pokazano w [19].

Stwierdzenia aktywne mogą występować jako stwierdzenia pierwotne, których wartości nie zależą jawnie od wartości innych stwierdzeń oraz stwierdzenia wtórne, których wartości zależą w sposób jawny od wartości innych stwierdzeń.

### 2.2.2. Macierzowa metoda badania stanu diagnostycznego

Macierzowa metoda badania stanu diagnostycznego systemów przeznaczona jest dla obiektów technicznych, których sygnały wyjściowe (parametry) można przedstawić w sposób jawny (mierzalny). Metoda ta oparta jest na kilku założeniach, z których najważniejszym w odniesieniu do rozpatrywanego problemu jest to, iż diagnozowanie opiera się na logice dwuwartościowej (opracowano również metodę opartą na logice trójwartościowej). Najczęściej przyjmowane jest założenie, że niezdatności układu są spowodowane uszkodzeniami pojedynczymi, a więc nie uwzględnia się możliwości jednoczesnego wystąpienia uszkodzenia więcej niż jednego elementu.

Macierzowa metoda badania stanu diagnostycznego systemu oparta została na odpowiednio przygotowanym modelu funkcjonalnym tego systemu i odpowiednich założeniach. W metodzie tej zakłada się przeprowadzenie analizy diagnostycznej badanego obiektu  $O$ . W analizie takiej uwzględnia się schemat funkcjonalny obiektu  $O$ , zasadę działania tego obiektu, jego przeznaczenie i specyfikę pracy oraz wymaganą głębokość wnikania w strukturę badanego obiektu w celu lokalizacji uszkodzeń. Wynikiem przeprowadzonej analizy jest:

- zbiór elementów podstawowych  $E=\{e_j\}$ ,
- zbiór sprawdzeń, wymaganych do procesu lokalizacji uszkodzeń  $D=\{d_i\}$ .

Ponadto przyjmuje się następujące założenia [62]:

- a) niezdatności modelowanego obiektu są spowodowane uszkodzeniami pojedynczymi,
- b) wszystkie elementy podstawowe obiektu objęte są bezpośrednią kontrolą,
- c) istnieje pełna wiarygodność sprawdzeń,
- d) nie nakłada się żadnych ograniczeń na liczbę sygnałów wyjściowych  $z_k \in Z$ , elementu podstawowego  $e_j \in E$ ,
- e) diagnozowanie będzie się opierać na logice dwuwartościowej, z której zastosowania wynikają następujące konsekwencje:
  - element  $e_j$  obiektu  $O$  może znajdować się w stanie zdatności  $\varepsilon(e_j)=1$  lub w stanie niezdatności  $\varepsilon(e_j)=0$ ,

- sprawdzenie  $d_i$  może dać wynik pozytywny  $D_i(\varepsilon(e_j)=1)=1$  lub negatywny  $D_i(\varepsilon(e_j)=0)=0$ , w zależności od tego w jakim stanie znajduje się badany obiekt i jego elementy.

Każde sprawdzenie  $d_i$ , przeprowadzane jest na pewnym podziorze elementów  $E_j \in \mathbf{E}$ . W wyniku takiego działania otrzymywany jest tor sprawdzenia [62]:

$$\forall_{\substack{d_i \in D \\ i=1..N}} (d_i \mapsto [E_i]d_i); \quad E_i \in \mathbf{E} \quad (6)$$

gdzie:

- $\mapsto$  - relacja przyporządkowania,
- $[E_i]d_i$  - podzbiór elementów podstawowych  $e_j$ , objętych sprawdzeniem  $d_i$ .

Wynik sprawdzenia  $d_i$  zależy od stanu elementów  $[E_i]d_i$  [62]:

$$\exists_{e_i \in E_j} ((\varepsilon(e_i) = 0) \Leftrightarrow (D_j(\varepsilon(e_i) = 0) = 0)), \quad (7)$$

$$\forall_{e_j \in E_i} ((\varepsilon(e_j) = 1) \Leftrightarrow (D_i(\varepsilon(e_j) = 1) = 1)), \quad (8)$$

gdzie:

- $\varepsilon(e_j)=0$  - stan niezdatności elementu  $e_j$  obiektu  $\mathbf{O}$ ,
- $\varepsilon(e_j)=1$  - stan zdatności elementu  $e_j$  obiektu  $\mathbf{O}$ ,
- $D_j(\varepsilon(e_j)=0)$  - wynik sprawdzenia  $d_i$  przy niezdatnym elemencie  $e_j$ ,
- $D_j(\varepsilon(e_j)=1)$  - wynik sprawdzenia  $d_i$  przy zdatnym elemencie  $e_j$ ,

Opisana metoda prowadzi do ciekawego rozwiązania, służącego do opisu stanu układu. Rozpatrując tę metodę pod kątem zastosowania jej do rozpatrywanego zagadnienia opisu propagacji stanów, należy zwrócić uwagę na szereg ograniczeń. Metoda ta jest przeznaczona do opisu stanów dwu- ([61][62]) lub trójwartościowych ([60]). W przypadku stanów trójwartościowych sposób reprezentacji stanów staje się bardzo skomplikowany. Ponadto w metodzie tej zakłada się, że wszystkie niezdatności spowodowane są przez wystąpienie tylko pojedynczego uszkodzenia. Nie można więc uwzględnić niezdatności, które mogą być spowodowane przez wystąpienie kilku uszkodzeń. Można oczywiście przyjąć założenie upraszczające, że spośród grupy uszkodzeń, powodujących

wystąpienie niezdatności zostanie wybrane jedno uszkodzenie dominujące. Wystąpienie takiego uszkodzenia będzie wtedy jednoznaczne z wystąpieniem danej niesprawności.

### **2.2.3. Modele matematyczne bazujące na rozkładach zmiennych losowych**

Zasada działania modeli matematycznych bazujących na rozkładach zmiennych losowych, opiera się na interpretowaniu wyników pomiarów poprzez weryfikację hipotez statystycznych. W modelach tego typu rozpatruje się utratę przez badany obiekt stanu zdatności i pojawienie się stanu niezdatności. Rozróżnia się tutaj stany zdatności całkowitej, częściowej oraz stany niezdatności. Do najczęściej stosowanych modeli matematycznych można zaliczyć [38][6][34][39]:

a) rozkłady zmiennych losowych:

- dwumianowy (Bernoulliego),
- Poissona,
- hipergeometryczny,
- geometryczny,
- normalny, ucięty w zerze,
- gamma,
- Weibulla,
- wykładniczy,
- logarytmiczny gamma,
- logarytmo-normalny,
- logarytmiczny rozkład Weibulla,

b) procesy losowe:

- Poissona,
- normalny,
- Markowa,
- semi-Markowa.

Poszczególne rozkłady zmiennych losowych oraz rodzaje procesów losowych są stosowane dla różnych obiektów, w zależności od przeznaczenia danego obiektu, jego sposobu działania czy obserwowanych cech stanu. Modele takie wymagają wielu danych potrzebnych do ich weryfikacji i poprawnego działania [37], a co za tym idzie, konieczny jest odpowiedni czas związany z gromadzeniem dostatecznie dużej liczby



danych wejściowych. W większości wypadków modele takie są opracowywane do wyznaczania pewnych miar określających stan elementów. Nie pozwalają one na bezpośrednie modelowanie zmian stanów występujących w układach.

#### 2.2.4. Sieci Petriego

Sieć Petriego [7] [36] jest grafem zbudowanym z dwóch podstawowych elementów - węzłów i gałęzi. Węzły mogą odpowiadać miejscom lub przejściom. Miejsce służy do przechowywania informacji, a przejście opisuje operacje przetwarzania tych informacji. Możliwe są jedynie połączenia pomiędzy węzłami różnych typów. Do opisu dynamicznego służą znaczniki, które mogą znajdować się tylko w miejscach. Obecność znacznika w miejscach określa stan urządzenia. Znaczniki mogą poruszać się z miejsca na miejsce - odbywa się to poprzez przełączanie (zapalanie) odpowiedniego przejścia. Sieć Petriego opisana jest jako szóstka uporządkowana

$$(P_1, T, G, K, W, M_0) \quad (9)$$

gdzie:

- $P_1$  - niepusty i skończony zbiór miejsc (stanów),
- $T$  - niepusty i skończony zbiór przejść (zdarzeń),
- $D_1 \subset G \cup H$  - relacja incydencji (określa zbiór gałęzi skierowanych w sieci),
- $K: P_1 \rightarrow N$  - funkcja pojemności miejsc (określa maksymalną liczbę znaczników, jakie może posiadać miejsce),
- $W: D_1 \rightarrow N$  - funkcja pojemności gałęzi (krotności krawędzi, określa liczbę przenoszonych znaczników),
- $M_0: P_1 \rightarrow N_0$  - funkcja znakowania (markowania) początkowego,
- $G: P_1 \times T \rightarrow N$  - funkcja incydencji wejściowych (określa zbiór gałęzi skierowanych od miejsc do przejść),
- $H: T \times P_1 \rightarrow N$  - funkcja incydencji wyjściowych (określa zbiór gałęzi skierowany od przejść do miejsc),
- $N$  - zbiór liczb naturalnych.

Sieć Petriego inicjowana jest poprzez przypisanie każdemu miejscu określonej liczby znaczników. Dla podstawowego typu sieci Petriego istnieją różne rozszerzenia:

- każdy pojemnik może otrzymać atrybut zwany pojemnością, definiujący, ile ładunków może być maksymalnie odłożonych w danym pojemniku;

- poprzez przyporządkowanie przewodom różnych wag możliwe jest zdefiniowanie liczby ładunków, jakie mogą w jednym kroku “przepływać” przez dany przewód;
- ładunki mogą reprezentować sobą złożone typy danych;
- definiowane są różne elementy będące rozszerzeniami elementów podstawowych, np. moduły służące hierarchizacji sieci.

W praktyce stosowane są sieci Petriego wyższego rzędu, jak np. sieci z predykatami lub sieci kolorowe. Kolorowe sieci Petriego opisane są następującymi parametrami [94]:

$$(P_1, T, \Omega, G, H, \beta, \psi, \mu_0) \quad (10)$$

gdzie:

- $P_1$  - niepusty i skończony zbiór miejsc (stanów),
- $T$  - niepusty i skończony zbiór przejść (zdarzeń),
- $\Omega$  - niepusty i skończony zbiór kolorów znacznika,
- $G: P_1 \times T \rightarrow \{0, 1, 2, \dots\}$  - funkcja incydencji wejściowych,
- $H: T \times P_1 \rightarrow \{0, 1, 2, \dots\}$  - funkcja incydencji wyjściowych,
- $\beta: (P_1 \times \Omega) \times T \rightarrow (0, 1)$  - funkcja rozkładu kolorów znacznika na wejściowych pozycjach przejść sieci,
- $\psi: T \times (P_1 \times \Omega)$  - funkcja rozkładu kolorów znacznika na wyjściowych pozycjach przejść sieci,
- $\mu_0: P_1 \times \Omega \rightarrow (0, 1, 2, \dots)$  - początkowe oznakowanie sieci.

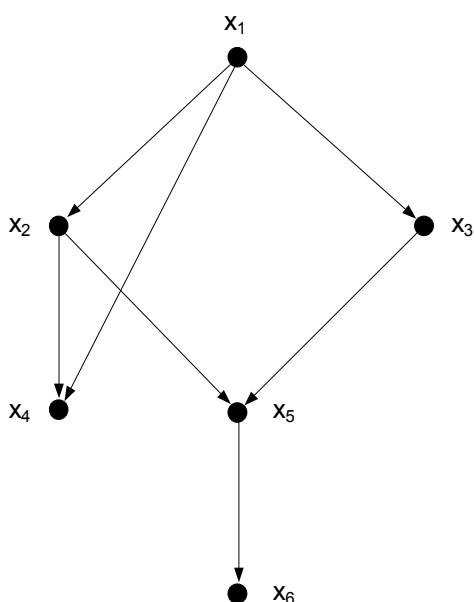
Podczas analizy systemu opisanego siecią Petriego mogą pojawić się następujące pytania:

- Czy sieć jest tak skonstruowana, że istnieje zawsze pewna liczba przejść gotowych do przełączenia, czy też istnieje stan, w którym sieć jest “martwa”?
- Czy każde przejście jest zdolne do przełączenia, tzn. czy istnieją przejścia, które poprzez ciągłe przełączanie blokują możliwość zaznaczenia innych przejść?
- Czy istnieją “zakleszczenia”, tzn. czy dochodzi do sytuacji, w których żadne przejście nie może być przełączone, a których uniknięcie możliwe byłoby poprzez zmianę kolejności przełączania przejść (tam, gdzie jest to możliwe)?

Odpowiedzi na te pytania trudno jest uzyskać drogą analityczną. Dlatego też dla zbadania zachowania się systemu modelowanego za pomocą sieci

Petriego przeprowadza się często symulacje takich systemów i na ich podstawie szuka się odpowiedzi na nie.

Zastosowanie sieci Petriego do bezpośredniego modelowania stanów w układach maszyn wydaje się raczej nieuzasadnione. Sieci te bardziej nadają się do modelowania układów z przepływem informacji i/lub danych. Dlatego też można raczej znaleźć ich zastosowanie do modelowania przepływu informacji np. pomiędzy węzłami w sieci stwierdzeń lub pomiędzy agentami w systemach wieloagentowych.



Rys. 2. Przykład sieci Bayesa [69]

### 2.2.5. Sieci przekonań (Bayesa)

Ciekawym rozwiązaniem umożliwiającym nie tylko reprezentację stanu, ale również, a może przede wszystkim, opisanie przepływu danych pozwalających określić stan poszczególnych elementów modelowanego obiektu, są sieci przekonań, nazywane również sieciami Bayesa lub sieciami oddziaływań ([69],[49]). Sieć przekonań jest skierowanym acyklicznym grafem, w którym poszczególnym węzłom sieci przypisane są zmienne, natomiast łuki łączące poszczególne węzły odpowiadają bezpośrednim oddziaływaniom występującym pomiędzy zmiennymi, a siła tych powiązań reprezentowana jest poprzez prawdopodobieństwo warunkowe. Łączny rozkład prawdopodobieństwa uzyskuje się jako złożenie warunkowych prawdopodobieństw poszczególnych zmiennych względem węzłów nadrzędnych. Przykład typowej sieci Bayesa dla rozkładu:

$$P(x_1, \dots, x_6) = P(x_6|x_5) P(x_5|x_2, x_3) P(x_4|x_1, x_2) P(x_3|x_1) P(x_3|x_1) P(x_2|x_1) P(x_1)$$

przedstawiono na rys. 2. Jedną z zalet tego typu modeli opartych na łącznym prawdopodobieństwie jest ich przejrzystość, co pozwala sprawdzić, czy w sieci nie występują niezgodności.

Sieci przekonań znalazły bardzo szerokie zastosowanie nie tylko w opisie stanów, ale przede wszystkim w sztucznej inteligencji i w takich dziedzinach jak statystyka, filozofia czy analiza decyzji.

W przeciwieństwie do rozwiązania przyjętego w zbiorach stwierdzeń, dla prawidłowego działania sieci przekonań konieczna jest znajomość wartości prawdopodobieństw warunkowych wszystkich węzłów sieci. Określenie wartości tych prawdopodobieństw nie jest zadaniem prostym, ponieważ do ich określenia konieczna jest duża grupa obiektów badań. Wartości prawdopodobieństw można również przyjąć w sposób subiektywny, podobnie jak jest to określane w przypadku stwierdzeń.

Ciekawym rozwiązaniem jest podejście zaproponowane przez Pearla [69]. Zaproponował on zastosowanie sieci Bayesa do reprezentacji wiedzy niepewnej. Pearl zaproponował również, m.in. metodę fuzji, na podstawie której określa się prawdopodobieństwo poszczególnych węzłów sieci. Sieci takie zostały nazwane sieciami Pearla.

#### 2.2.6. Sieci Pearla

W sieciach przekonań podstawową sprawą jest opracowanie metody, która pozwoli określić, w jaki sposób obliczane są prawdopodobieństwa w poszczególnych węzłach sieci. Opracowanych zostało wiele różnych metod pozwalających na wyznaczenie poszukiwanego prawdopodobieństwa.

Pearl w swojej metodzie zaproponował, by każdy z węzłów sieci potraktować jako autonomiczny element. W ramach takiego elementu rozdzielone zostały dane pochodzące od sąsiadów danego węzła na dane pochodzące od węzłów nadrzędnych (rodziców) oraz dane pochodzące z węzłów podrzędnych (potomków). Prawdopodobieństwo określane jest na podstawie danych otrzymanych od bezpośrednich sąsiadów w sposób następujący [69]:

$$BEL(B) = P(B | D_B^+, D_B^-) = \alpha P(D_B^- | B) \cdot P(B | D_B^+) \quad (11)$$

gdzie:

$BEL(B)$  - dynamiczna wartość prawdopodobieństwa węzła B,

$\alpha$  - stała normalizująca,

$D_B^-$  - dane opisujące część sieci związaną z rodzicami węzła B,

$D_B^+$  - dane opisujące część sieci związaną z potomkami węzła B.

Przedstawioną powyżej zależność można uogólnić do poszczególnych węzłów sieci:

$$\lambda(B_i) = P(D_B^- | B_i) \quad (12)$$

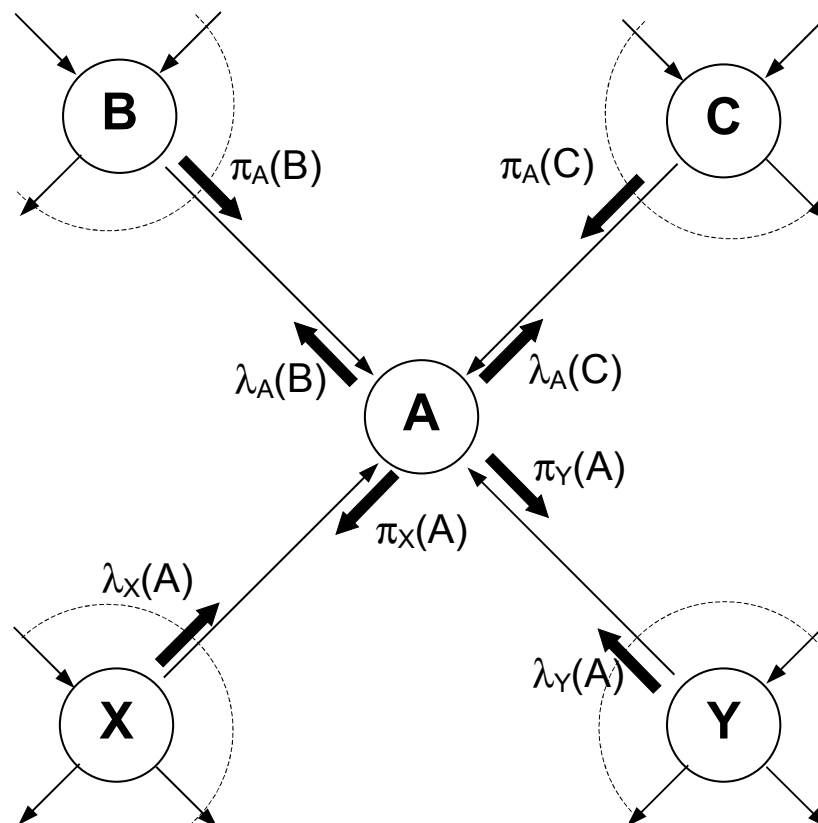
$$\pi(B_i) = P(B_i | D_B^+) \quad (13)$$

co pozwala wyznaczyć całkowitą „siłę” oddziaływań:

$$BEL(B_i) = \alpha \lambda(B_i) \pi(B_i) \quad (14)$$

Na rys. 3 przedstawiono fragment sieci przekonań z wydzielonym węzłem. Oddziaływania na elementy nadrzędne (rodziców) są opisane za pomocą parametru  $\lambda$ , a oddziaływania na elementy podrzędne opisuje parametr  $\pi$ . W rozwiązaniu zaproponowanym przez Pearla parametry  $\lambda$  i  $\pi$  przekazywane są pomiędzy węzłami sieci w postaci odpowiednich komunikatów. Na tej podstawie wyznacza się wartość zmiennej opisującej dany węzeł w sieci.

Wszystkie zmiany w sieci odbywają się przez przekazywanie danych pomiędzy węzłami; zmiana wartości jednego węzła pociąga za sobą zmiany



Rys. 3. Fragment sieci z pojedynczym węzłem [69]

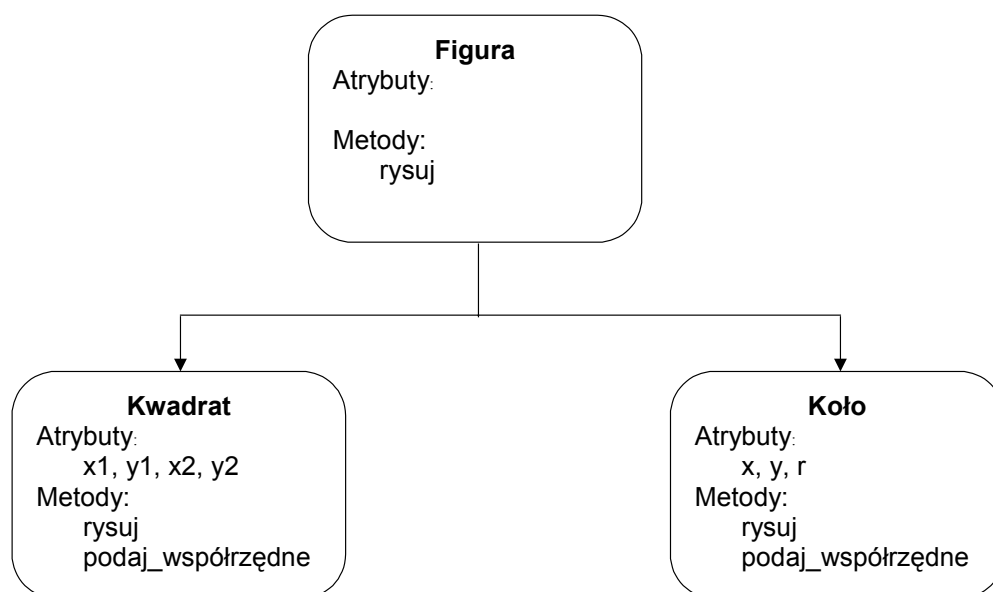
w pozostałych węzłach sieci.

Podejście zaproponowane przez Pearla prowadzi do bardzo ciekawego rozwiązania, które wydaje się szczególnie nadawać do opisu propagacji stanów w układach maszyn. Pozwala mianowicie na modelowanie propagacji stanów pomiędzy węzłami sieci. Możliwe jest dostosowanie takiej sieci do opisu stanów elementów modelowanego układu, co pozwoli opisać propagację stanów pomiędzy elementami. Wadą tej metody wydaje się być sposób określenia wag jako wartości prawdopodobieństw warunkowych, gdyż wiąże się to, w przypadku badań niezawodnościowych, z koniecznością przeprowadzenia takich badań na dużej grupie badanych obiektów. W praktyce taka możliwość nie zawsze istnieje. Rozwiązaniem, które wówczas można zastosować, jest przyjęcie odpowiednich wag w sposób subiektywny i na tej podstawie określenie siły powiązań pomiędzy węzłami sieci.

### 2.2.7. Modele obiektowe

W modelach obiektowych [7] [90] [73] [74] realny świat zostaje odwzorowany na pewien zbiór obiektów. Obiekty o tych samych (lub podobnych) cechach zaliczane są do tej samej klasy. Z modelami obiektowymi związane są cztery podstawowe pojęcia:

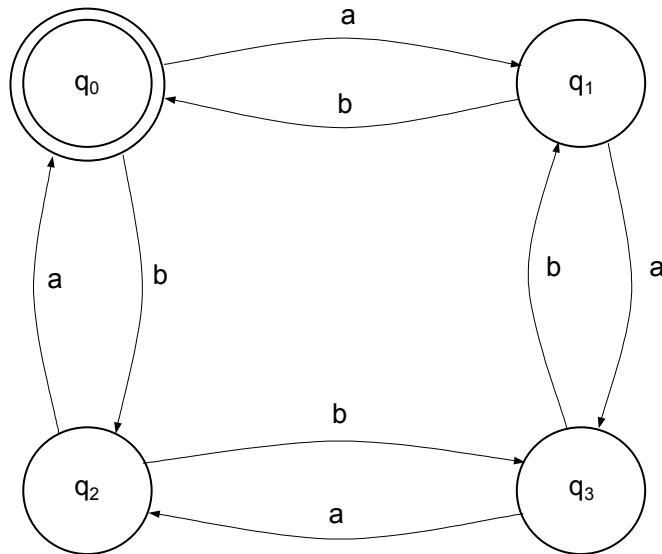
*Abstrakcja* - realne przedmioty, struktury, pojęcia są zastępowane abstrakcyjnymi obiektami. W ich definicji pomijanych jest wiele szczegółów. Pozostawiane są tylko te cechy, które jednoznacznie odróżniają dany obiekt od pozostałych obiektów.



Rys. 4. Przykład struktury klas obiektów

- Hermetyzacja* - obiekt jest "hermetycznym" komponentem, zawierającym dane (atrybuty) oraz opis sposobu manipulacji tymi danymi (metody). Jedynym możliwym sposobem dostępu do danych i operowania nimi jest wykorzystywanie metod dostępu do danych. Obiekt może, w celu implementacji własnych operacji, wykorzystywać metody innych obiektów. Istnieje tylko jedna ścieżka sterowania - oznacza to, że obiekt wywołujący metodę innego obiektu musi czekać na zakończenie odpowiednich operacji w obiekcie wywoływanym.
- Dziedziczenie* - jest to sposób reprezentacji różnych poziomów abstrakcji. Klasy obiektów organizowane są w hierarchiczne struktury. Klasy położone niżej w hierarchii dziedziczą atrybuty oraz metody od klas położonych wyżej.
- Polimorfizm* - jest związany z pojęciem dziedziczenia. Umożliwia on różne implementacje tej samej metody na różnym poziomie hierarchii klas. Niech klasy *Kwadrat* i *Koło* będą podklasami klasy *Figura* (rys. 4). Klasy te dziedziczą z klasy *Figura*, będącej dla nich klasą nadrzędną, pewne wspólne atrybuty, jak np. kolor, grubość linii itp. oraz wspólne metody. Załóżmy, że w klasie *Figura* została zdefiniowana metoda *rysuj*. W klasach podrzędnych *Kwadrat* i *Koło* można tę metodę zdefiniować ponownie, lecz przypisać jej nowe działanie. W klasie *Kwadrat* będzie to rysowanie kwadratu, a w klasie *Koło* rysowanie koła. Dzięki temu, wiedząc tylko, że każdy obiekt, będący realizacją klasy *Figura* (lub jej klas pochodnych), może być narysowany przy pomocy metody *rysuj*, możemy być pewni, że odpowiednie konkretne figury (kwadrat, kółko) zostają narysowane we właściwy sposób.

Podjęcie obiektowe posiada wiele zalet. Umożliwia stosunkowo proste i szybkie przygotowanie odpowiednich obiektów z istniejących klas. Model oparty na podejściu obiektowym można w miarę prosto modyfikować i rozwijać. Dzięki zdefiniowanym klasom i wykorzystaniu dziedziczenia i polimorfizmu modele obiektowe można prosto adaptować w zależności od potrzeb. Wadą tego typu rozwiązania w modelowaniu zmian stanów jest pewna trudność w opracowaniu odpowiedniej biblioteki klas oraz sposobu przekazywania oddziaływań pomiędzy różnymi obiektami. Podejście obiektowe może być natomiast bardzo przydatne w połączeniu z innymi metodami, np. zbiorami stwierżeń.



Rys. 5. Diagram przejść dla automatu skończonego [44]

### 2.2.8. Automat skończony

Automat (skończony) ([8] [30] [33] [84] [41] [50]) charakteryzuje się skończoną liczbą stanów, pozwalających opisać przyszłe zachowanie się systemu w zależności od tego, co wydarzyło się wcześniej.

*Automat skończony* jest modelem systemu o dyskretnych wejściach i wyjściach. System taki może znajdować się w jednej ze skończonej liczby konfiguracji, czyli stanów. Aktualny stan takiego systemu jest wynikiem reakcji automatu na poprzednie wejścia, co ma wpływ na zachowania systemu przy następnych wejściach. Bardzo wiele przykładów systemów o skończonej liczbie stanów można znaleźć w życiu codziennym. Przykładem takiego prostego systemu może być mechanizm sterujący windy. Taki mechanizm nie pamięta wcześniejszych żądań, a jedynie bieżące - piętro, kierunek ruchu (górze czy dół) oraz zbiór żądań, które pozostały do zrealizowania. W celu lepszego wyjaśnienia opisu działania automatów o skończonej liczbie stanów należy zapoznać się z kilkoma definicjami. Definicje te zaczerpnięte zostały z [44].

Automat skończony składa się ze skończonego zbioru stanów i zbioru przejść ze stanu do stanu, które mogą następować dla różnych symboli wejściowych, wybieranych z pewnego alfabetu  $\Sigma$ . Dla każdego symbolu wejściowego istnieje dokładnie jedno przejście z każdego stanu (w szczególnym przypadku może prowadzić do tego samego stanu), odpowiadające temu symbolowi. Jeden ze stanów, oznaczany zazwyczaj symbolem  $q_0$ , jest stanem początkowym. Od niego automat rozpoczyna działanie. Niektóre ze stanów należą do zbioru stanów końcowych.



Z automatem skończonym wiąże się diagram przejść, który jest grafem skierowanym zdefiniowanym w następujący sposób: wierzchołki grafu odpowiadają stanom, a gałęzie – przejściom. Jeżeli w automacie istnieje przejście ze stanu  $q$  do stanu  $p$ , dla zadanego symbolu wejściowego  $a$ , to diagram przejść zawiera gałąź prowadzącą ze stanu  $q$  do stanu  $p$  i opatrzoną etykietą  $a$  (rys. 5). Formalnie automat skończony przedstawiany jest jako piątka uporządkowana:

$$(Q, \Sigma, \delta, q_0, F) \quad (15)$$

gdzie:

- $Q$  – jest skończonym zbiorem stanów,
- $\Sigma$  – jest skończonym zbiorem akcji
- $\delta: Q \times \Sigma \rightarrow Q$  – jest funkcją przejść,
- $q_0 \in Q$  – jest stanem początkowym,
- $F \subset Q$  – jest zbiorem stanów końcowych.

Funkcja przejść  $\delta$  przyporządkowuje każdemu stanowi  $q$  i każdemu symbolowi  $a$  nowy stan  $\delta(q, a)$ . Automat skończony można wyobrazić sobie jako układ czytający ciąg symboli  $\Sigma$  zapisanych na taśmie. W każdym kroku automat znajdujący się w stanie  $q$  odczytuje symbol  $a$  z zadanego ciągu symboli wejściowych, przechodzi w stan  $\delta(q, a)$  i przesuwają układ odczytujący o jeden symbol w prawo.

Przedstawiony powyżej automat nosi nazwę *deterministycznego automatu skończonego*. Deterministyczny automat skończony jest najprostszym przykładem automatu skończonego. Rozwinięciem deterministycznego automatu skończonego jest niedeterministyczny automat skończony. W niedeterministycznym automacie skończonym dopuszcza się istnienie 0, 1 lub wielu przejść ze stanu, przy tym samym symbolu wejściowym. Jeżeli w przypadku deterministycznego automatu skończonego automat może znaleźć się w danej chwili tylko w jednym stanie, to automat niedeterministyczny może znaleźć się w dowolnej liczbie stanów. *Niedeterministyczny automat skończony* definiowany jest, podobnie jak automat deterministyczny, jako piątka uporządkowana (15). Inaczej jednak definiowana jest funkcja przejść  $\delta$ , która odwzorowuje  $Q \times \Sigma$  w  $2^Q$ . Funkcja  $\delta(q, a)$  jest zbiorem wszystkich stanów  $p$ , dla których istnieje przejście z  $q$  do  $p$  o etykiecie  $a$ . Deterministyczny automat skończony jest szczególnym przypadkiem automatu niedeterministycznego.

Kolejne rozszerzenia doprowadziły do utworzenia *automatów dwukierunkowych* i *automatów z wyjściem*. Dwukierunkowy automat skończony to taki automat, którego układ odczytujący symbole z listy symboli

wejściowych może przesuwac się zarówno w przód jak również w tył tej listy. Automaty skończone z wyjściem usuwają ograniczenia poprzednich typów automatów związanych z ograniczeniem wyjścia automatu skończonego do wielkości dwuwartościowych. Istnieją tutaj dwa podejścia: wyjście związane jest ze stanem (*automat Moore'a*) lub z przejściem (*automat Mealy'ego*).

Automat Moore'a to szóstka uporządkowana

$$(Q, \Sigma, \Delta, \delta, q_0, \lambda) \quad (16)$$

gdzie:

- $Q, \Sigma, \delta, q_0$  – takie same jak dla deterministycznego automatu skończonego,
- $\Delta$  – alfabet wyjściowy,
- $\lambda$  – odwzorowanie  $q$  w  $\Delta$  zadające wyjście związane z każdym ze stanów.

Automat Mealy'ego to również szóstka uporządkowana

$$(Q, \Sigma, \Delta, \delta, \lambda, q_0) \quad (17)$$

gdzie:

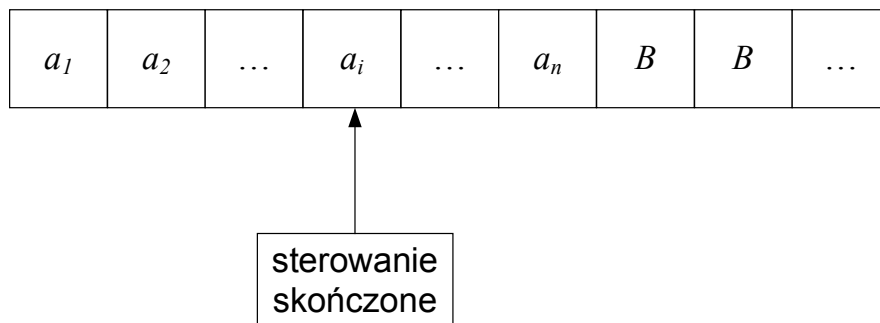
- $Q, \Sigma, \delta, q_0$  – takie same jak dla deterministycznego automatu skończonego,
- $\Delta$  – alfabet wyjściowy,
- $\lambda$  – odwzorowanie  $Q \times \Sigma$  w  $\Delta$ .

Przedstawiony opis automatu skończonego pozwala na stwierdzenie, że taki zapis może być przydatny zarówno do opisu stanów, w jakich może znaleźć się dany układ, bądź jego fragment, jak również do określenia samych zmian stanów.

Ogólnym zapisem maszyny o skończonej liczbie stanów jest *maszyna Turinga*. Opisane wcześniej modele automatów skończonych są szczególnymi przypadkami maszyny Turinga. Podstawowy model takiej maszyny (rys. 6) ma:

- skończone sterowanie,
- taśmę wejściową podzieloną na komórki,
- głowicę taśmy, która może obserwować w danej chwili tylko jedną komórkę.

Taśma w takiej maszynie jest prawostronnie nieskończona. W każdej z komórek taśmy może znajdować się jeden ze skończonej liczby symboli



Rys. 6. Podstawowa maszyna Turinga

wejściowych. Pierwszych  $n$  komórek zawiera wejście, które jest zbiorem symboli wejściowych. Maszyna Turinga w pojedynczym ruchu, w zależności od stanu sterowania skończonego oraz symbolu aktualnie obserwowanego przez głowicę taśmy, wykonuje następujące działania:

- zmienia stan,
- drukuje symbol w określonej komórce taśmy, zastępując symbol poprzednio tam wpisany,
- przesuwa głowicę o jedną komórkę w prawo lub w lewo.

Formalnie maszyną Turinga nazywamy:

$$M=(Q, \Sigma, \Gamma, \delta, q_0, B, F) \quad (18)$$

gdzie:

- $Q$  – skończony zbiór stanów,
- $\Gamma$  – skończony zbiór dopuszczalnych „symboli taśmowych”,
- $B$  – symbol pusty, należący do  $\Gamma$ ,
- $\Sigma$  – podzbiór  $\Gamma$ , nie zawierający  $B$ ,
- $\delta$  – funkcja następnego ruchu, będąca odwzorowaniem  $Q \times \Gamma$  w  $Q \times \Gamma \times \{L, P\}$ ,
- $q_0$  – stan początkowy, ( $q_0 \in Q$ ),
- $F$  – zbiór stanów końcowych ( $F \subseteq Q$ ).

Istnieje wiele modyfikacji maszyny Turinga, np.: z taśmą dwustronnie nieskończoną, wielotaśmowa maszyna Turinga, maszyna wielowymiarowa, maszyna wielogłowicowa itp. Model maszyny Turinga jest ogólnie akceptowanym modelem formalnym efektywnej procedury.

Podsumowując opis maszyn o skończonej liczbie stanów, można powiedzieć, że metoda opisu stanów i zmian stanów za pomocą automatu skończonego jest ciekawym i godnym rozważenia rozwiązaniem. Należy tutaj jednak zwrócić uwagę na pojęcie czasu. Zakładając (zgodnie z zasadą działania maszyny o skończonej liczbie stanów), że zmiana stanu może się odbywać jedynie po odczytaniu kolejnego znaku z taśmy wejściowej, należy w tego typu systemach liczyć się z pewną zwłoką czasową związaną z kolejnymi cyklami pracy automatu skończonego. Za pomocą automatu skończonego można uzyskać opis stanów nie tylko dwuwartościowych, ale również wielowartościowych, co czyni tę metodę szczególnie przydatną w poszukiwaniu rozwiązania zagadnienia modelowania zmian stanów.

### **2.2.9. Systemy wieloagentowe**

Pojęcie *autonomicznego agenta* zostało wprowadzone w sztucznej inteligencji. Obecnie w wyniku rozwoju sieci komputerowych i systemów rozproszonych, pojęcie to stosowane jest nie tylko w tej dziedzinie wiedzy, ale również w innych. Istnieje wiele definicji tego pojęcia. Jednakże wybranie jednej, odpowiedniej definicji nie jest zadaniem ani prostym ani jednoznacznym. Niżej przedstawiono kilka z nich:

#### **Definicja 1**

Agentem można nazwać to wszystko co, postrzega swoje otoczenie za pomocą sensorów i oddziałuje na nie poprzez efekторы [86].

#### **Definicja 2**

Według Maes [54] autonomicznym agentem jest system obliczeniowy, umieszczony w złożonym, dynamicznym otoczeniu. Współdziała samodzielnie z tym otoczeniem i realizuje zbiór celów lub zadań, do których wykonania został utworzony.

#### **Definicja 3**

Inną definicję agenta zaproponowała Hayes-Roth [42]:

Inteligentny agent realizuje w sposób ciągły trzy zadania:

- obserwuje dynamicznie zmieniające się warunki w otoczeniu,
- może wpływać na warunki otoczenia,
- rozwiązuje problemy, wyciąga wnioski i wyznacza kolejne zadania.

#### **Definicja 4**

Bardziej złożoną definicję agenta przedstawili Jennigs i Wooldrige [92]. Według nich agentem jest „(...) system komputerowy (program komputerowy lub urządzenie komputerowe), posiadające następujące własności:

- autonomiczność: działa bez ingerencji człowieka i posiada pewną kontrolę nad wykonywanymi działaniami i swoim stanem,
- komunikatywność: współdziała z innymi agentami (również z ludźmi) za pośrednictwem specjalnie zdefiniowanego języka komunikacji,
- reaktywność: postrzega swoje środowisko, którym może być świat rzeczywisty, użytkownik dostępny przez specjalny interfejs, zbiór innych agentów, INTERNET lub ich kombinacja,
- kreatywność: agent powinien nie tylko biernie komunikować się z otoczeniem, ale również aktywnie oddziaływać na otoczenie, zgodnie z realizowanym celem.”

### **Definicja 5**

Z kolei Franklin i Graesser autonomicznym agentem nazywają system, który umieszczony w pewnym otoczeniu i będący jego częścią, współdziała z tym otoczeniem i oddziałuje na nie, działa według określonego schematu, w celu określenia zadań, które mają być wykonane w przyszłości [35].

Jak widać z powyższych przykładów, zdefiniowanie pojęcia agent nie jest zadaniem prostym. Można jednak we wszystkich tych definicjach zauważyć pewne cechy wspólne dla nich. Otóż za agenta uważany jest pewien system, czy to komputerowy lub inny, „żyjący” w pewnym środowisku i z tym środowiskiem współdziałający. Środowisko to ma wpływ zarówno na działania agenta, jak i jego stan, ale jednocześnie sam agent również może na środowisko w jakiś sposób oddziaływać. Gdyby definicja agenta miała dotyczyć tylko wymienionych cech, to można by powiedzieć, że jest to zwykły program komputerowy. Jednakże agent, to również zawarta w nim wiedza i zdefiniowany sposób jej wykorzystania. Agent sam określa, na podstawie aktualnego stanu środowiska, w którym się znajduje, jakie zadania mają być przez niego wykonywane.

Agent powinien więc charakteryzować się następującymi cechami:

- współdziałać z otoczeniem, w którym się znajduje,
- być aktywny, nie tylko odpowiadać otoczeniu na jego żądanie, ale również w wyniku realizacji założonego celu inicjować własne działania,
- być komunikatywny, współdziałać z innymi agentami lub ludźmi, w celu realizacji zadanego celu lub pomocy innym agentom.

W niniejszej pracy przyjęto, że agentem nazywany będzie system komputerowy, oddziałujący ze środowiskiem, w którym został umieszczony, posiadający pewną wiedzę na temat wykonywanych zadań oraz określony cel działania. Przykładową strukturę agenta przedstawiono na rys. 7. Agent komunikuje się z otoczeniem poprzez wysyłanie i odbieranie komunikatów.

Ma więc pewną wiedzę, posiada określony cel, a jego działanie jest zdefiniowane.

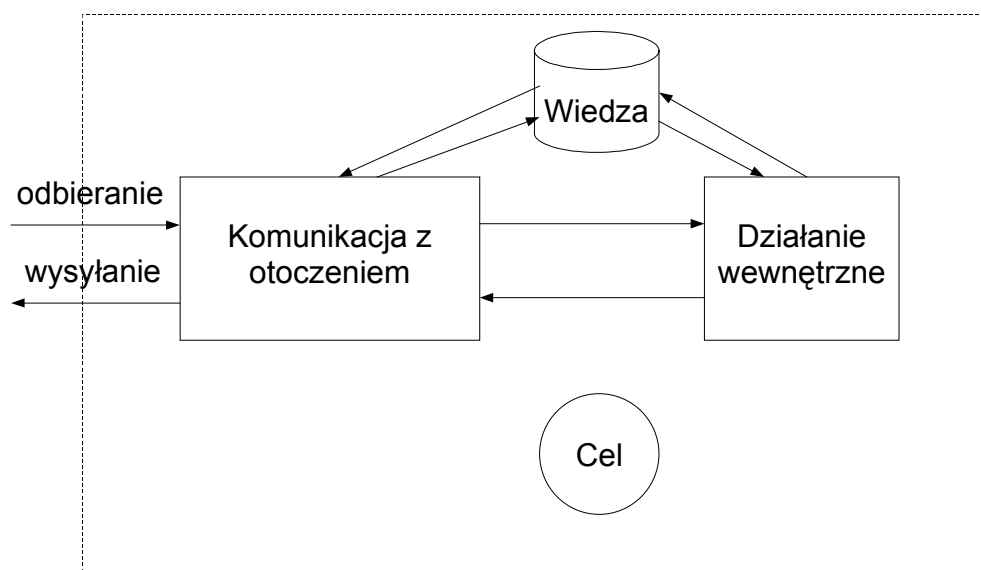
Na szczególną uwagę zasługują rozwiązania bazujące na grupie współpracujących agentów. Mowa tu o *systemach agentowych*. Systemy agentowe zostały wprowadzone jako alternatywa dla dużych scentralizowanych systemów. Miały one na celu rozdzielenie głównego zadania na mniejsze podzadania, realizowane w różnych miejscach i z różnymi zasobami wiedzy.

Przez system agentowy rozumiany jest taki system, w którym elementem kluczowym jest agent. Każdy system agentowy musi zawierać niezerową liczbę agentów. Częściej niż system agentowy wykorzystywane są *systemy wieloagentowe*. Są one bardziej rozpowszechnione i bardziej złożone. Ważnym zagadnieniem rozpatrywanym w systemach z wieloma agentami jest problem komunikacji między nimi. Zaproponowano wiele rozwiązań tego zagadnienia. Pomiedzy agentami mogą istnieć bezpośrednie połączenia, służące komunikacji między nimi. Mogą one współdziałać poprzez tablicę ogłoszeń [19][93], lub specjalnie opracowane języki służące komunikacji między agentami [92][2][12].

Mówiąc o systemach wieloagentowych można wyróżnić różne rodzaje agentów:

- „inteligentne”,
- ruchome,
- adaptacyjne.

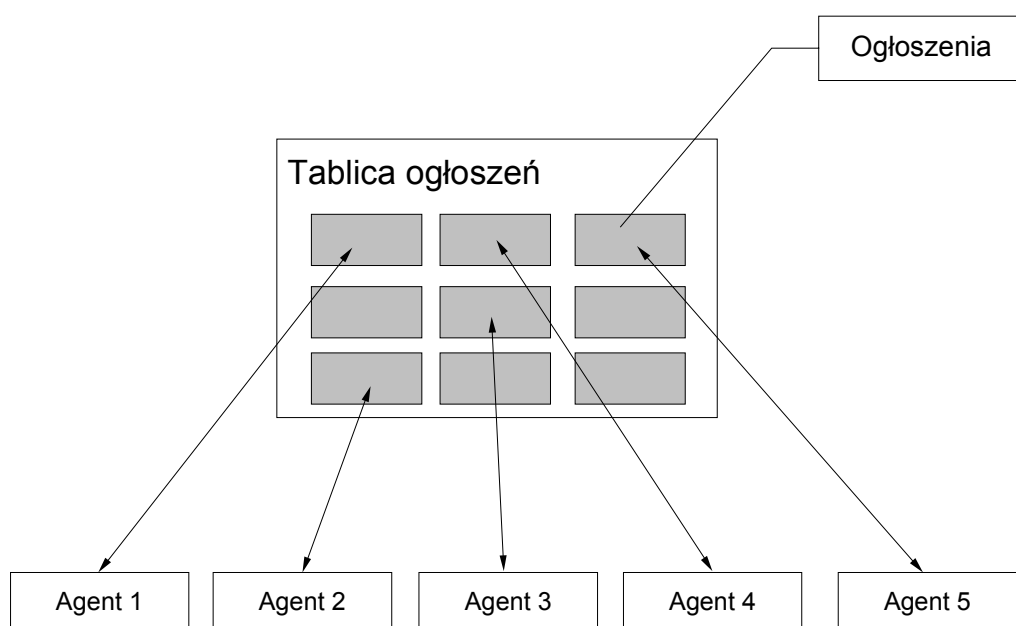
Niektóre typy agentów mogą uwzględniać różne wielkości



Rys. 7. Struktura Agentu [93]

charakterystyczne, zależne od przeznaczenia. Jednakże, aby wydobyć w pełni zalety zastosowania systemu agentowego, system taki powinien zawierać opisane wcześniej cechy charakterystyczne. Jest to ważne również dlatego, by odróżnić je od innych rozwiązań, takich jak systemy zorientowane obiektowo, systemy rozproszone, typu klient-serwer ([91]) czy systemy doradcze.

Jednym z ciekawszych rozwiązań pozwalających na komunikację między agentami w systemach wieloagentowych jest koncepcja tablicy ogłoszeń [19][93]. Ogólna koncepcja tablicy ogłoszeń została przedstawiona na rys. 8. Zasada działania tablicy ogłoszeń jest, ogólnie rzecz biorąc, prosta.



Rys. 8. Koncepcja tablicy ogłoszeń

Przyjmuje się, że istnieje wspólna tablica, na której każdy z agentów może umieszczać swoje ogłoszenia, dotyczące informacji o swoim działaniu, zapytania do innych agentów, poszukuje tym samym danych do wykonania swojego zadania. Odpowiednio przygotowana tablica ogłoszeń może pełnić rolę otoczenia, z którym agent komunikuje się i wymienia dane oraz poprzez które komunikuje się z innymi agentami.

Systemy wieloagentowe mają bardzo wiele zastosowań. Są wykorzystywane w rozwiązaniach aplikacji przemysłowych do kontroli procesów, produkcji i rozwiązywania problemów komunikacyjnych. Znalazły również szerokie zastosowanie w aplikacjach zarządzania informacją (np. w Internecie), handlu elektronicznym i zarządzaniu. Ciekawe zastosowanie systemy wieloagentowe znalazły w medycynie. Przygotowywano rozwiązania, oparte na działaniu systemów wieloagentowych, dotyczące

obsługi pacjentów oraz opieki zdrowotnej [46]. Systemy wieloagentowe nie są, oczywiście, rozwiązaniem uniwersalnym i nie we wszystkich wypadkach należy je stosować. Rozwiązań opartych na systemach wieloagentowych nie należy stosować wówczas, gdy wymaga się zachowanie globalnych ograniczeń lub działania w czasie rzeczywistym. Działanie agenta i jego wiedza ograniczone są do jego bezpośredniego otoczenia, a więc jest to wiedza lokalna. Dlatego też nie należy stosować takich systemów w układach, gdzie trzeba określić wiedzę globalną dla całego systemu.

Do opracowania systemów opartych na pojęciu agenta zostało przygotowanych wiele specjalistycznych narzędzi programistycznych. Aby oprogramowanie mogło być uznane za agenta lub system agentów, nie ma potrzeby opracowywania go przez specjalistyczne narzędzia. Można wykorzystać istniejące języki programowania, za pomocą których można opracować zarówno pojedynczego agenta jak również całe systemy agentowe.

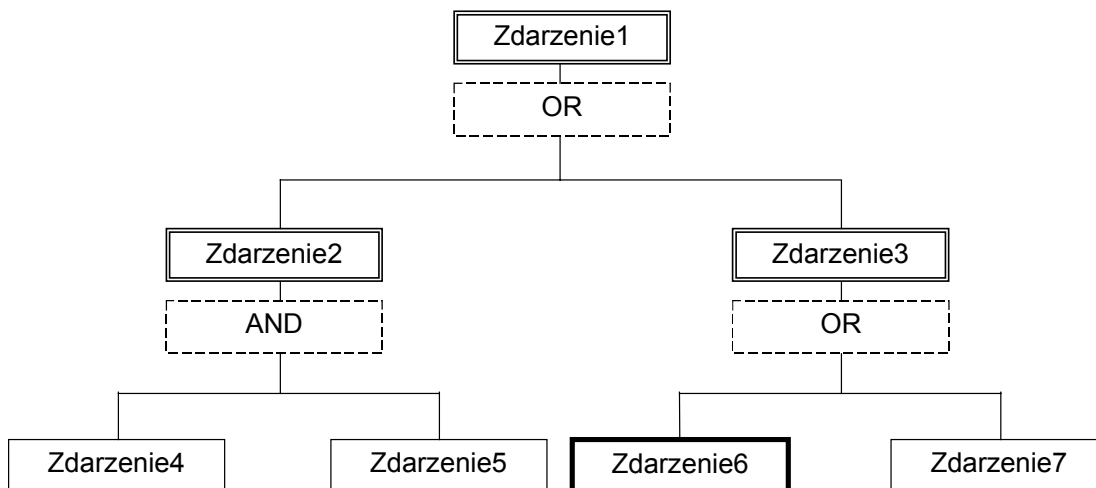
Zastosowanie systemów wieloagentowych do modelowania stanów, a szczególnie zmian stanów jest możliwe i wydaje się uzasadnione. Z podobnym rozwiązaniem mamy do czynienia w opisanym wcześniej podejściu opartym na zbiorach stwierdzeń, gdzie aktywnymi elementami są węzły sieci stwierdzeń. Wykorzystanie tego typu systemów do modelowania stanów wymaga zdefiniowania odpowiednich metod związanych z komunikacją między agentami oraz zdefiniowaniem działań wykonywanych przez same agenty. Podejście takie jest dosyć ciekawym, nowatorskim rozwiązaniem, lecz wymaga prowadzenia wielu prac adaptacyjnych związanych z przystosowaniem go do modelowania stanów.

#### **2.2.10. Drzewa uszkodzeń**

Drzewa uszkodzeń [22][64][67] służą do znalezienia pierwotnego źródła uszkodzeń, przez odtwarzanie wstecz przyczynowo-skutkowej ścieżki propagacji uszkodzeń. Drzewa uszkodzeń są powszechnie stosowane w badaniach dotyczących niezawodności maszyn [1][43][65]. Budowane są dla układów złożonych. Układy takie poddaje się dekompozycji na podukłady, z uwzględnieniem zależności przyczynowo-skutkowych pomiędzy stanami podukładu i stanem układu nadrzędnego. Następnie, w celu zbudowania drzewa uszkodzeń, dla każdego z podukładów oraz dla całego układu określa się [22]:

- listę oczekiwanych działań, które powinny być realizowane poprawnie,
- listę uszkodzeń powodujących, że układ lub podukład nie może realizować poprawnie przyporządkowanych mu działań,
- listę warunków powodujących uszkodzenia lub warunków sprzyjających powstawaniu uszkodzeń,





Rys. 9. Przykładowa postać drzewa uszkodzeń

- listę podukładów, których nieodpowiedni stan techniczny może być przyczyną uszkodzenia. Zarówno uszkodzenia jak również ich przyczyny są w drzewach uszkodzeń rozpatrywane jako zdarzenia. Drzewo uszkodzeń może być reprezentowane w postaci grafu, w którym:

- zdarzenia są przypisane węzłom grafu,
- gałęzie grafu reprezentują istniejące relacje przyczynowo-skutkowe.

Gałęzie takiego grafu mogą być pogrupowane za pomocą operatorów, takich jak np.: AND, OR, XOR. W drzewie uszkodzeń rozróżnia się następujące rodzaje węzłów [22]:

- zdarzenia podstawowe, wskazujące na uszkodzenie określonego podzespołu bez konieczności prowadzenia dalszych badań,
- zdarzenia nierozpoznane, dla których identyfikacji potrzebne są dane uzupełniające (co jest zazwyczaj związane z przeprowadzeniem dodatkowych badań),
- zdarzenia złożone, które są skutkami lub przyczynami innych zdarzeń.

Na rys. 9 przedstawiono przykładową postać drzewa uszkodzeń. Stwierdzenia podstawowe obramowano linią ciągłą, zdarzenia nierozpoznane obramowano linią ciągłą grubą, a zdarzenia złożone są obramowane linią podwójną.

Zarówno węzłom drzewa uszkodzeń, jak również krawędziom można przyporządkować pewne wagi, interpretowane jako prawdopodobieństwa lub prawdopodobieństwa warunkowe wystąpienia określonego zdarzenia. Wagi te mogą być stosowane do wyznaczania wartości miar określających niezawodność obiektu. Takie drzewa nazywane są ważonymi drzewami

uszkodzeń [22]. Wagi te mogą być określane bądź to na drodze odpowiednich badań, bądź w sposób subiektywny, np. przez specjalistę.

### **2.2.11. Podsumowanie**

Z przedstawionego w tym rozdziale opisu metod przeznaczonych do reprezentacji wielowartościowych stanów, zmian stanów oraz propagacji stanów wynikają następujące wnioski:

1. Opisane metody są przeznaczone głównie do opisu stanów dwu- lub trójwartościowych. Do opisu stanów wielowartościowych można zastosować jedynie niektóre z opisanych rozwiązań, jak np. rozwiązanie oparte na zbiorach stwierdzeń, modelach obiektowych czy systemach wieloagentowych.
2. Niektóre z przedstawionych metod opisu stanów wymagają dużej liczby obiektów badań, co wynika z wykorzystywanego w nich aparatu matematycznego (rachunku prawdopodobieństwa). To w znacznym stopniu ogranicza możliwości ich wykorzystania, gdyż w diagnostyce nie zawsze można uzyskać odpowiednio liczne grupy obiektów.
3. Na szczególną uwagę zasługuje rozwiązanie oparte na zbiorach stwierdzeń. Pozwala ono na modelowanie stanów i zmian stanów oraz zależności występujących pomiędzy elementami układu.
4. Położono nacisk na kilka spośród przedstawionych metod. Są to: zbiory stwierdzeń, modele obiektowe oraz systemy wieloagentowe. Posłużyły one do opracowania modelu procesu propagacji stanów, który jest oryginalnym osiągnięciem autora rozprawy.

## 3. Model procesu propagacji stanów

### 3.1. Model i modelowanie

Opracowywanie modeli obiektów (układów) jest spowodowane potrzebą opisanie opisów zjawisk w zależności od aktualnie posiadanej wiedzy. Im większa jest wiedza na temat rozpatrywanego fragmentu rzeczywistości, tym bardziej szczegółowe mogą być budowane modele i tym lepiej oddają one modelowaną rzeczywistość. Abstrakcyjny lub konkretny układ, posiadający własności i właściwości stanowiące cechy umożliwiające jego wyróżnienie lub wyodrębnienie z otoczenia (spośród innych zbiorów [31]) został w pracy nazwany *obiektem*.

Na obiekt, będący przedmiotem modelowania, można spojrzeć w różnoraki sposób. Najczęściej poszukiwana jest zależność pozwalająca możliwie wiernie opisać działanie i/lub stan obiektu. W diagnostyce technicznej maszyn głównym celem jest identyfikacja stanu obiektu. Stan ten może być opisywany za pomocą wielu różnych cech, pozwalających na jego klasyfikację. Na stan obiektu mają wpływ czynniki związane z działaniem obiektu, oddziaływaniami występującymi wewnątrz obiektu oraz pomiędzy rozpatrywanym obiektem i innymi obiektami znajdującymi się w jego otoczeniu. W dalszej części pracy zwrócono szczególną uwagę na oddziaływania występujące wewnątrz obiektu, pomiędzy jego elementami. Przyjęto, że stan złożonego obiektu może być określany, między innymi, na podstawie stanów jego elementów składowych. Stany poszczególnych elementów często nie powinny być rozpatrywane jako stany niezależne, ponieważ stan każdego z nich może zależeć od stanu (jednego lub kilku) pozostałych elementów, jak również jego stan może wpływać na stan innych elementów. Takie oddziaływania stanów określono jako *proces propagacji stanów*.

Proces propagacji stanów można rozpatrywać, stosując podejście indywidualne lub ogólne. Podejście indywidualne polega na oddzielnej (indywidualnej) analizie każdego rozpatrywanego obiektu. W podejściu ogólnym zakłada się możliwość opracowania uniwersalnej (ogólnej) metody postępowania, pozwalającej na podobne rozpatrywanie różnych obiektów należących do w miarę możliwości, licznej grupy. Stosując pierwsze podejście, konieczne jest zastosowanie wielu specjalistycznych metod postępowania, przeznaczonych tylko dla jednego obiektu. W drugim przypadku postuluje się stosowanie jednej ogólnej metody. Wydaje się, że drugie podejście jest postępowaniem bardziej racjonalnym, pozwalającym między innymi na gromadzenie wiedzy. Aby można je było zastosować, celowe jest określenie klasy rozpatrywanych obiektów [64], gdzie klasa

obiektów rozumiana jest jako zbiór wszystkich obiektów, spełniających odpowiednie, przyjęte, warunki przynależności do tej klasy. Takie podejście ułatwia opracowanie modelu opisującego proces propagacji stanów i wyznacza możliwości jego zastosowania.

Pojęciu *model* przypisywane są różne znaczenia. Szczególnie w ostatnim okresie znaczenie tego pojęcia podlegało wielu zmianom, wynikającym z pojawienia się nowych dziedzin, w których znalazło ono zastosowanie. Niżej przedstawiono kilka wybranych definicji znaczenia pojęcia „model”, zaczerpniętych z [88]:

- 1) wzór, według którego coś jest lub ma być wykonywane;
- 2) model ekonomiczny – hipotetyczna konstrukcja myślowa obejmująca układ założeń przyjętych w ekonomii dla uchwycenia najistotniejszych cech i zależności występujących w danym procesie ekonomicznym
- 3) modele matematyczne – zależności opisujące wyidealizowane zjawiska fizyczne lub ekonomiczne; przyrządy matematyczne służące do rozwiązywania albo do ilustracji tych zależności, interpretacje różnych pojęć i teorii, matematycznych,
- 4) układ względnie odosobniony, możliwie mało skomplikowany, działający analogicznie do oryginału, którym może być istota żywa, maszyna, zakład przemysłowy, organizacja społeczna itd.

Model diagnostyczny to relacja wskazująca na współwystępowanie określonych wartości cech stanu obiektu i wartości cech sygnałów diagnostycznych (symptomów). Ograniczony zakres naszej wiedzy oraz obserwacji obiektu nie pozwalają, niestety, na zakładanie, iż relacje takie będą przyczynowo-skutkowe. W zależności od potrzeb tworzone są modele indywidualne, które opisują właściwości pojedynczego obiektu lub grupowe, opisujące wspólne właściwości pewnej klasy obiektów [22]. Ze względu na złożoność modelowanych układów i cel modelowania, czasami konieczne jest przyjęcie uproszczeń, których skutkiem może być ograniczenie zgodności opracowywanego modelu z modelowanym obiektem.

W diagnostyce maszyn szerokie zastosowanie znalazły między innymi następujące klasy modeli diagnostycznych [64]:

- model strukturalny,
- model matematyczny,
- model systemowy,
- model logiczny,
- model jakościowy.

*Model strukturalny* [64][10], to model odwzorowujący dynamiczną strukturę modelowanego układu. Model taki odwzorowuje również wzajemne oddziaływania występujące pomiędzy elementami rozpatrywanego układu. Zastosowanie tego typu modeli pozwala na przeprowadzanie analiz dotyczących działania badanego układu oraz, między innymi, analiz w zakresie [64]:

- określenia zbioru stanów technicznych,
- identyfikacji wielkości fizycznych, których zmiany należy obserwować,
- wyboru cech sygnałów diagnostycznych,
- lokalizacji czujników pomiarowych,
- określenia warunków działania obiektu podczas badań,
- określenia związków przyczynowo skutkowych pomiędzy cechami stanów i cechami sygnałów diagnostycznych.

*Model matematyczny* opisuje związki pomiędzy wartościami szeroko rozumianych cech obiektu i jest budowany na podstawie zidentyfikowanego modelu strukturalnego [64]. Zazwyczaj model taki jest reprezentowany przez zależności liniowe lub nieliniowe, obowiązujące dla określonych zakresów wartości cech obiektu [64][66][45]. Przykładem takiego modelu opracowanego dla potrzeb diagnostyki jest model opisujący działanie silnika elektrycznego prądu stałego [64][4] lub model opisujący ruch czopa w łożysku ślizgowym [64][47].

Ciekawym zastosowaniem modeli matematycznych jest wykorzystanie ich do poszukiwania odwrotnych relacji diagnostycznych [21].

Do określania *modeli systemowych* stosowane są wybrane pojęcia z ogólnej teorii systemów [64][57]. Centralną rolę wśród tych pojęć pełni pojęcie systemu określonego jako relacja na iloczynie kartezjańskim wejść i wyjść tego obiektu. Przykładem zastosowania tego typu modeli w diagnostyce jest model symptomowy [64][11].

*Model logiczny*, budowany w oparciu o aparat logiki matematycznej obejmuje stwierdzenia, orzekające o własnościach i właściwościach modelowanego obiektu oraz wnioski, dotyczące stanu obiektu, które na ich podstawie można wyciągnąć, a dotyczące własności i zachowania się modelowanego układu [64][57].

W przypadku *modeli jakościowych* rozpatrywane są dwa odrębne podejścia, które znalazły zastosowanie w diagnostyce:

- modele jakościowe z zastosowaniem logiki wielowartościowej ([64]),
- modele jakościowe działania układów dynamicznych z zastosowaniem metody QSIM (Qualitative SIMulation) ([64][21])

Modele jakościowe opracowane z zastosowaniem logiki wielowartościowej są identyfikowane w sposób indukcyjny, z wykorzystaniem tzw. „uczenia maszynowego” na podstawie przykładów i kontrprzykładów. Nieco inaczej opracowywane są modele jakościowe opracowane z zastosowaniem metody QSIM. Wywodzą się one z tradycyjnego podejścia do modelowania działania obiektu za pomocą układu równań różniczkowych. Równania te oparte są na wielkościach jakościowych.

W rozpatrywanym dalej procesie budowania modeli diagnostycznych wyróżniono kilka etapów postępowania [9]:

- identyfikacja obiektu (systemu),
- wybór przekroju obiektu (fragmentu obiektu, zawierającego wybrane ze względu na przyjęte kryteria elementy i odpowiednio wyodrębnione relacje odpowiadające w logiczny sposób założonym celom badawczym),
- identyfikacja modelu obiektu,
- weryfikacja opracowanego modelu.

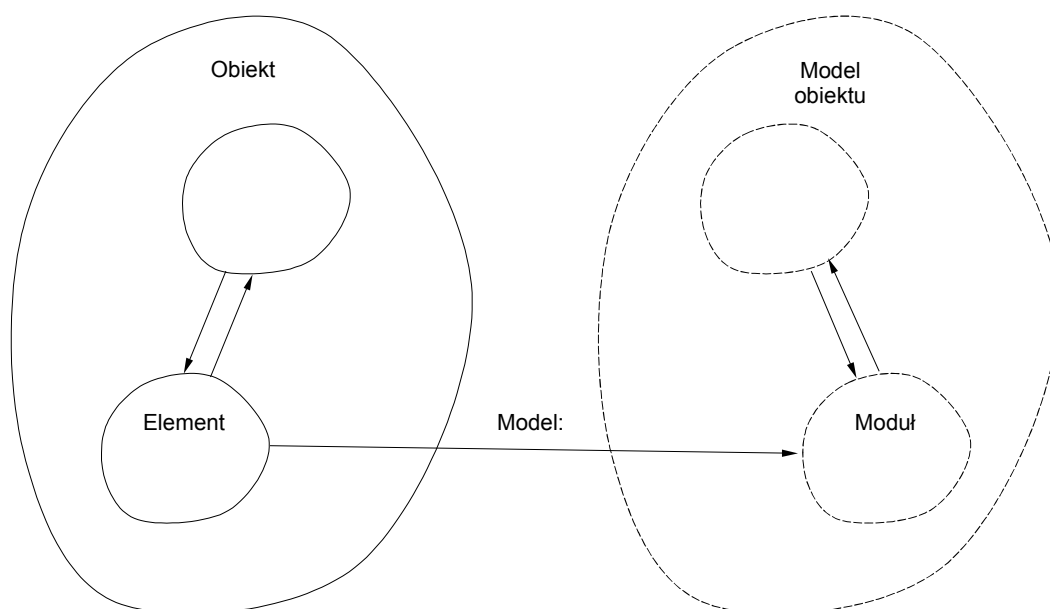
Przed opracowaniem modelu procesu propagacji stanów w układach maszyn należy znaleźć odpowiedzi na kilka pytań:

1. W jaki sposób wyodrębnić rozpatrywane obiekty, czyli jak podzielić modelowany układ na podukłady?
2. Jak reprezentować rozpatrywane obiekty?
3. Jak opisać stan obiektów?
4. W jaki sposób opisać oddziaływania występujące między poszczególnymi obiektami (elementami składowymi układów), których skutkiem mogą być zmiany stanów obiektów?

### **3.2. Model układu**

Współczesne maszyny i urządzenia są układami składającymi się z setek lub tysięcy elementów, podzespołów i zespołów, często bardzo skomplikowanych. Poprawne działanie jednego z podzespołów ma często duże znaczenie dla realizacji zadań wykonywanych przez inne podzespoły. Wiele z oddziaływań występujących pomiędzy poszczególnymi elementami jest znanych i jasno zdefiniowanych. Wynikają one m.in. z praw fizyki, istoty działania maszyny, jej budowy. Prace prowadzone w celu zwiększenia niezawodności działania maszyn prowadzą do odkrywania i definiowania różnych relacji diagnostycznych i niezawodnościowych, stanowiących podstawę do wnioskowania o aktualnym stanie maszyny. Znajomość stanu maszyny pozwala na podjęcie działań zwiększających bezpieczeństwo eksploatacji, a także zmniejszających koszty użytkowania maszyny oraz koszty remontowe.

Maszyna, będąca obiektem badań, jest układem złożonym, tzn. zbudowana jest z odpowiednio wykonanych współdziałających podzespołów i zespołów. Przyjęto, że modele uwzględnianych elementów będą nazywane *modułami*. Zbiór odpowiednio dobranych współdziałających modułów tworzyć będzie model zbioru współdziałających elementów, czyli model obiektu. Nazwę „moduł” wprowadzono tutaj celowo, rezygnując ze stosowania określenia „element modelu”, które mogłoby prowadzić do nieporozumień, przy jednoczesnym stosowaniu określenia „element obiektu”. Obiektem, którego modelem jest moduł, może być część elementu (np. czop wału), element (np. wał), podzespół lub zespół. Poszczególne elementy obiektu połączone są rzeczywistymi więzami, wynikającymi z konstrukcji obiektu. Ważnym założeniem jest to, że moduły mogą być również modelami abstrakcyjnych (umownych) elementów obiektów, występujących na przykład w postaci więzów pomiędzy elementami rzeczywistymi.



Rys. 10. Element obiektu i moduł modelu obiektu

Formułując założenia opisywanego ogólnego modelu obiektu (maszyny), przyjęto, że spełniony ma być postulat minimalnej różnorodności środków niezbędnych do opisanego modelu. Przez analogię do formułowanych w informatyce własności obiektowych języków programowania uznano, że warunek ten pozwoli na minimalizację ograniczeń możliwych zastosowań opracowywanego modelu. Spełnienie postulatu minimalnej różnorodności jest możliwe przy przyjęciu założenia, że rozpatrywany będzie wyłącznie jeden, uniwersalny typ modułów, tworzących model obiektu. Korzystając z rozwiązań stosowanych powszechnie w obiektowych językach programowania, założono, że wprowadzone moduły mają posiadać zdolność do:

- zagnieżdżania,
- hermetyzacji.

Wymaganą zdolność do zagnieżdżania można zilustrować za pomocą struktury katalogów na twardym dysku komputera. Dowolny katalog może zawierać inne katalogi nazywane podkatalogami. Podobnie wygląda to w przypadku zagnieżdżania modułów. Dowolny moduł może zawierać jeden lub wiele innych modułów.

Hermetyzacja umożliwia jednocześnie przechowywanie wewnątrz modułu danych oraz metod (procedur), przy czym dostęp do danych znajdujących się wewnątrz modułu może odbywać się wyłącznie za pomocą odpowiednio przygotowanych metod.

Wynikiem przyjętych założeń jest to, że moduł może zawierać inne moduły, które nie są bezpośrednio dostępne z zewnątrz modułu. Tak uzyskane własności modułów odzwierciedlają w pewnym zakresie własności modelowanego układu, gdzie często obserwować można zespoły (np. kompletną podporę łożyskową wału turbiny) bez możliwości obserwacji i dostępu do elementów tego zespołu (np. panewki łożyska) w czasie normalnej eksploatacji.

Ważnym wnioskiem wynikającym z przyjętych założeń jest stwierdzenie, iż kompletny model obiektu jest również modułem, spełniającym podane wyżej warunki.

### **3.3. Stopnie szczegółowości modelu**

Model obiektu może być rozpatrywany z różnym stopniem szczegółowości [75][80]. Może bardziej lub mniej szczegółowo uwzględniać konstrukcję i działanie samego obiektu. Można zbudować taki model, który będzie zawierał jedynie moduły odpowiadające zespołom należącym do modelowanego obiektu oraz również taki, który będzie uwzględniał najmniejsze nawet elementy modelowanego obiektu. Wiedza dostępna na temat obiektu, jak również wymagania stawiane modelowi ograniczają rozpatrywany stopień szczegółowości modelu. Wydaje się, że najlepszym rozwiązaniem jest rozpatrywanie modelu o różnych stopniach szczegółowości, uwzględniającego zarówno dostępną wiedzę jak i wymagania stawiane modelowi.

Budując model o dużym stopniu szczegółowości, zawierający znaczną liczbę modułów, należy się zastanowić, czy tak duża szczegółowość jest potrzebna oraz czy stopień szczegółowości powinien być taki sam dla całego modelu. Można rozpatrywać przypadki, gdy pewne fragmenty modelu należy przedstawić bardziej szczegółowo, a inne mniej szczegółowo. Jest to zależne nie tylko od oczekiwań dotyczących modelu, ale również od stanu wiedzy diagnostycznej na temat poszczególnych elementów, podzespołów lub



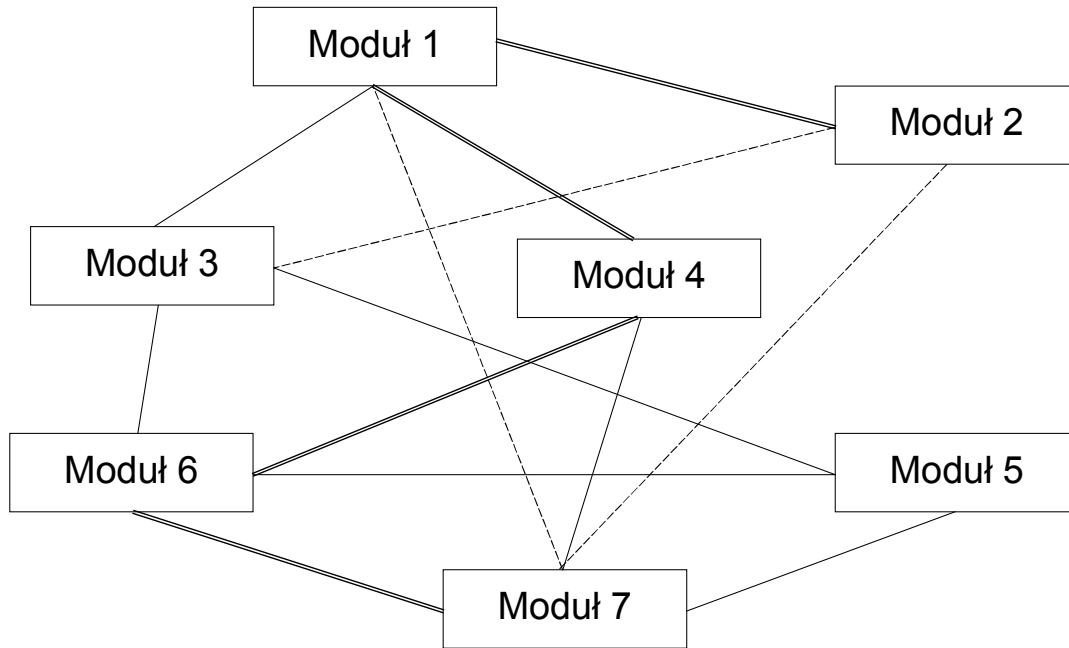
zespołów modelowanego obiektu. Wiedza dotycząca niektórych elementów może być bardzo mała lub może jej w ogóle nie być. Nie można by było wtedy zamodelować poszczególnych modułów. Natomiast w innych przypadkach, mimo tego, że wiedza jest wystarczająca, nie ma potrzeby rozpatrywania obiektu z dużym stopniem szczegółowości, a wystarczy jedynie uwzględnienie podzespołu lub zespołu.

Przyjęta we wcześniejszym rozdziale koncepcja zagnieżdżonych modułów doskonale nadaje się do reprezentacji modelu z różnym stopniem szczegółowości przez odpowiedni dobór liczby modułów zawartych wewnątrz każdego modułu. Takie podejście pozwala na rozpatrywanie modelu zbudowanego ze zmiennym stopniem szczegółowości. W zależności od potrzeb oraz posiadanej wiedzy możliwe jest zagnieżdżanie modułów do odpowiedniego poziomu szczegółowości i to niezależnie dla każdego z nich.

W celu umożliwienia równoczesnego stosowania różnych stopni szczegółowości wprowadzono w pracy pojęcie „widoku modułu”. *Widok modułu* to odpowiednio zdefiniowany sposób patrzenia na moduł. W definicji widoku modułu można określić, które moduły należące do danego modułu mają być uwzględniane i jak „głęboko” będą uwzględniane zagnieżdżone moduły. Zdefiniowanie różnych widoków modułu umożliwi rozpatrywanie modelu układu z różnym stopniem szczegółowości oraz pozwoli na komponowanie widoków złożonych, które będą składały się z wcześniej zdefiniowanych widoków.

### **3.4. Warstwy modelu**

Rozpatrując poszczególne moduły należy zauważyć, że w każdym z modułów można uwzględnić różne aspekty działań wykonywanych w rzeczywistym obiekcie. Na działanie modułu mogą mieć wpływ inne moduły, które również rozpatrują te same aspekty działań. Pomędzy poszczególnymi modułami mogą istnieć oddziaływania wiążące je między sobą (rys. 11). Oddziaływania te mogą wynikać z bezpośrednich oddziaływań łączących elementy układu, które były podstawą utworzenia modułów lub relacji diagnostycznych, jakie można między nimi znaleźć. Dla dużych układów, gdzie oddziaływań występujących pomiędzy poszczególnymi elementami jest bardzo dużo, taki opis może być bardzo nieczytelny i zbyt skomplikowany w zapisie, a jednocześnie rozpatrywanie różnych aspektów działania modułów prawie niemożliwe. Można jednak zauważyć, że wszystkie aspekty działania można podzielić na szereg grup. Dlatego też zaproponowano, by uwzględniane w działaniach modułów różne aspekty ich działań połączyć w pewne grupy, przy czym każda z tych grup zawierałaby aspekty tego samego typu, np. związane z przepływem czynnika roboczego, sterowaniem, smarowaniem itp. Na rys. 11 przedstawiono prosty przykład

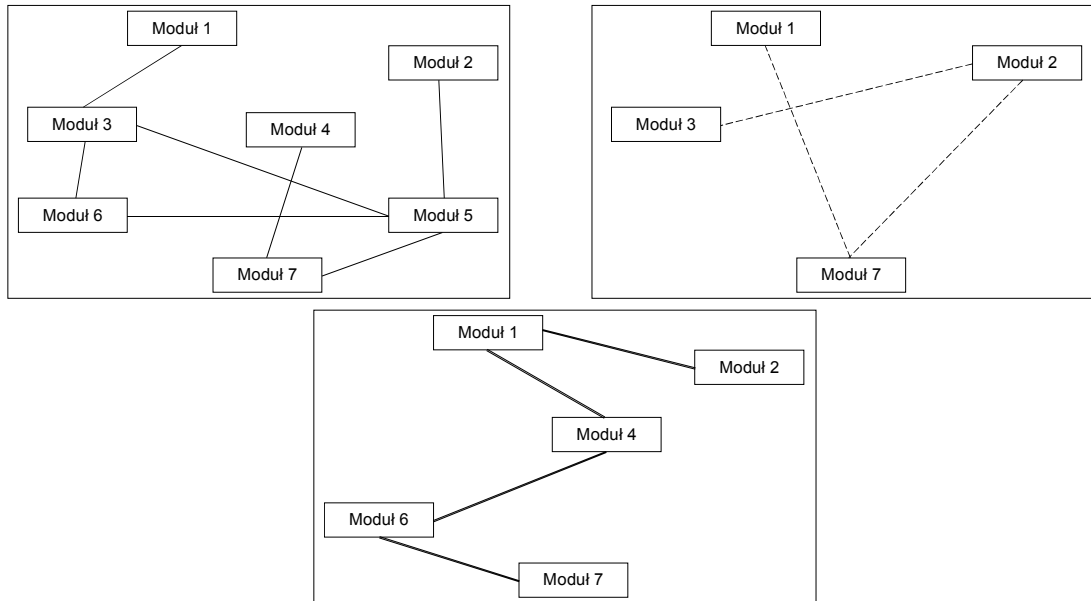


Rys. 11. Wybrane moduły modelowanego układu

modułów, w których uwzględniono jednocześnie różne aspekty ich działań. Z całego przedstawionego na tym rysunku układu można wyróżnić podukłady, z których każdy będzie opisywał jeden z typów oddziaływań między modułami. Przykłady wybranych fragmentów układu przedstawiono na rys. 12. Poszczególne fragmenty układu przedstawione na tym rysunku zawierają różną liczbę modułów, co zależy od aspektów działania rozpatrywanych przez poszczególne moduły w ramach danego, przedstawionego, fragmentu. W wyniku przeprowadzono podziału otrzymano szereg warstw, uwzględniających różne aspekty działania poszczególnych modułów.

*Warstwa modelu* to zbiór tych modułów, które rozpatrują zagadnienia tego samego typu [77][79]. Na przykład moduły rozpatrujące zagadnienia związane z analizą dynamiczną obiektu lub przepływem czynnika roboczego są umieszczane na jednej warstwie. W danym obiekcie można wyróżnić wiele różnych zagadnień, które są wspólne dla dwóch lub większej liczby modułów. Dlatego też w wyniku takiego podziału otrzymamy wiele warstw, które mogą zawierać różną liczbę modułów. Otrzyma się w takim przypadku model wielowarstwowy. Wprowadzenie warstw pozwala na „skupienie uwagi” na analizowanym aktualnie zagadnieniu i umożliwia wprowadzenie lokalnych modeli cząstkowych. Takie postępowanie umożliwia podział złożonych zagadnień, związanych z analizą różnych aspektów działania danego modułu, na szereg mniejszych, znacznie mniej skomplikowanych.

Moduły powstają w wyniku przestrzennej lub funkcjonalnej dekompozycji modelowanego układu. Wynikiem przeprowadzonej dekompozycji układu jest



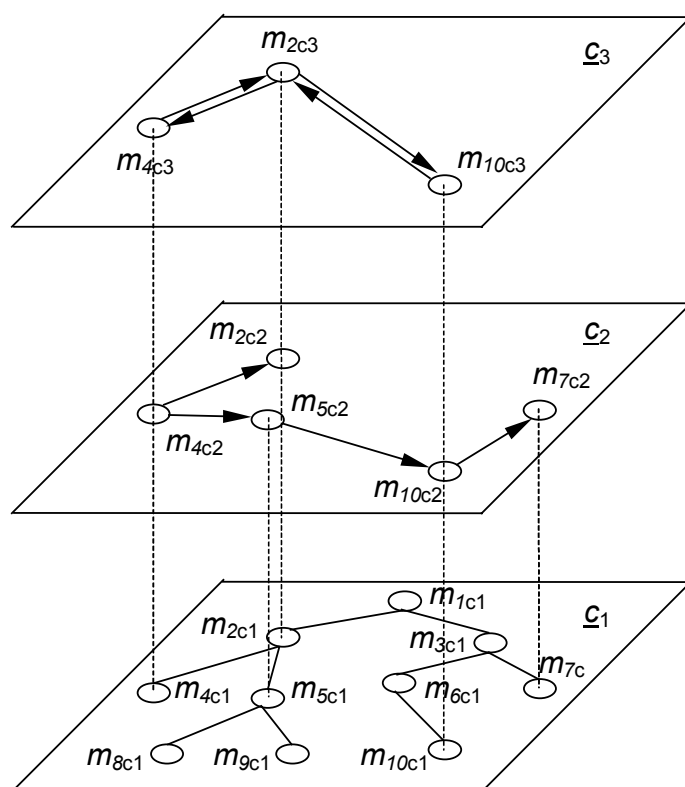
Rys. 12. Podział układu na podukłady

otrzymanie opisu struktury tego układu. Przyjmuje się, że otrzymana struktura układu stanowi podstawową warstwę. W warstwie tej uwzględnione są wszystkie moduły oraz zależności hierarchiczne wynikające z dekompozycji układu. Każda kolejna warstwa jest widokiem na przyjęty w warstwie podstawowej zbiór modułów. W ramach każdego widoku mogą zostać uwzględnione różne moduły w zależności od rozpatrywanych przez nie aspektów działania, które mają wpływ na działanie danego modułu.

Warstwę modelu można traktować jako szczególny przypadek widoku modelu. Podobnie jak w przypadku widoku tak i w przypadku warstwy wybierane są fragmenty modułów, spełniające ponadto dodatkowe kryteria. W przeciwieństwie do widoku, gdzie modułom nie są stawiane żadne wymagania, wszystkie fragmenty modułów należące do danej warstwy łączą wspólny kontekst i wspólne typy oddziaływań. Podobnie jak ma to miejsce w przypadku widoku, do jednej warstwy mogą należeć moduły, znajdujące się na różnych poziomach szczegółowości.

### 3.5. Ogólna synteza modelu

Moduły, z których zbudowany jest model, mogą być elementami wielu warstw [78][81][82][83]. Udział modułu w danej warstwie oznacza, że w swoim działaniu uwzględnia on określony typ oddziaływań, wynikający z przynależności do danej warstwy. Oddziaływania uwzględniane przez moduł w ramach jednej warstwy mogą mieć wpływ na wartości stwierdzeń wyznaczanych przez ten moduł, a wynikających z uwzględniania oddziaływań związanych z inną warstwą. Np. moduł *łożysko ślizgowe nr 2* może należeć do takich warstw jak *obciążenie elementów układu* oraz *układ*



Rys. 13. Moduły należące do różnych warstw

*olejowy*. W takim przypadku istnieje zależność pomiędzy obciążeniem łożyska a własnościami filmu olejowego wytworzonego pomiędzy czopem i panwią łożyska. Dlatego też w ramach danego modułu należy uwzględnić fakt wewnętrznych zależności występujących pomiędzy stwierdzeniami, których wartości są wyznaczane w zależności od przynależności modułu do różnych warstw. Na rys. 13 przedstawiono przykład grupy modułów należących do różnych warstw. W ramach każdej warstwy mamy do czynienia z tym samym modułem, lecz w każdej z nich moduł uwzględnia jedynie oddziaływania związane z daną warstwą. Rozbicie układu na szereg warstw ma celu uproszczenie działań związanych z analizą tych samych aspektów działania kilku modułów. Syntezy modelu dokonuje się wewnątrz modułów, w których zapisana jest wiedza o sposobie ich działania.

### 3.6. Moduł podstawowy

W poprzednim rozdziale wprowadzono pojęcie modułu, który traktowany jest jako model elementu obiektu. Przyjęto, iż każdy moduł posiada możliwość zagnieżdżania i hermetyzacji. Oznacza to, że moduł może zawierać inne moduły, które nie są bezpośrednio dostępne poza modułem je zawierającym. W rozdziale tym rozpatrywany jest wzorcowy moduł będący

przedstawicielem podstawowej klasy modułów. Dla uproszczenia opisu moduł ten będzie nazywany modułem podstawowym.

Przystępując do określenia struktury modułu podstawowego, skupiono uwagę na działaniach realizowanych przez ten moduł. W pierwszym kroku założono, że moduł może być rozpatrywany jako model systemu, dla którego wyróżniono zbiory wartości wejść, wyjść i stanów. Uznano, że dla potrzeb diagnostyki technicznej działania realizowane przez moduł mogą być rozpatrywane w czasie dyskretnym. Działania te opisywane są ogólnie znanymi zależnościami:

$$\begin{aligned} \underline{y}[t] &= f(\underline{x}[t], \underline{q}[t]) \\ \underline{q}[t_2] &= g(\underline{q}[t_1], \underline{x}[t_1:t_2]) \end{aligned} \quad (19)$$

gdzie:

- $\underline{x}[t]$  – wejście systemu w chwili  $t$ ,
- $\underline{x}[t_1:t_2]$  – wejście systemu w przedziale czasu od chwili  $t_1$ , do chwili bezpośrednio poprzedzającej chwilę  $t_2$ ,
- $\underline{y}[t]$  – wyjście systemu w chwili  $t$ ,
- $\underline{q}[t]$  – stan w chwili  $t$ ,
- $\underline{q}[0]$  – znany stan początkowy

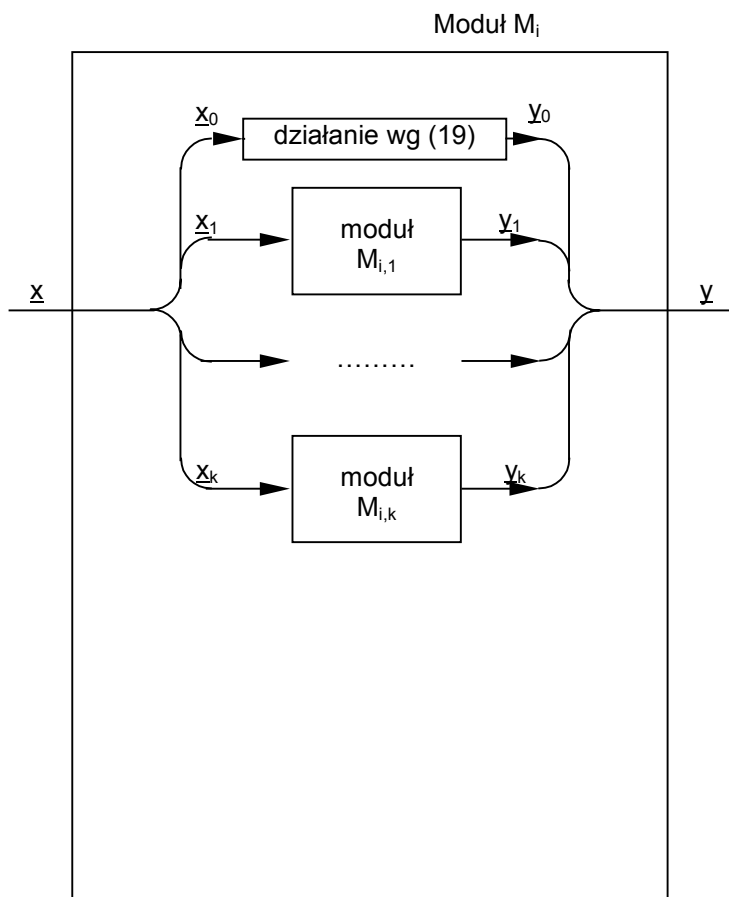
Z przyjętego założenia dotyczącego możliwości zagnieżdżania i hermetyzacji modułów wynika, że działania opisane przez (19) mogą być realizowane przez sieć modułów wewnętrznych, zawartych w rozpatrywanym module, gdzie zarówno rozpatrywany moduł jak i jego moduły wewnętrzne należą do klasy modułów podstawowych (rys. 14).

Głównym przeznaczeniem modułów jest możliwość modelowania stanów i zmian stanów badanych obiektów (maszyn). W wyniku przeprowadzonych badań stwierdzono, że bezpośrednio odwzorowywanie klas stanu obiektu technicznego lub wartości cech jego stanu w wartości zmiennych stanu  $\underline{q}$  (wg (19)) modułów zastosowanych w modelu obiektu jest zadaniem bardzo trudnym. Dla uproszczenia procesu definiowania modułów oraz dla umożliwienia łatwej interpretacji działań wykonywanych przez moduły postanowiono rozszerzyć strukturę modułu pokazaną na rys. 14 przez wprowadzenie dodatkowego bloku wewnętrznych wejść modułu. Założono, że wartości tych wejść (wszystkich lub wybranych) mogą być przekazywane na zewnątrz za pośrednictwem dodatkowych wyjść modułu (rys. 15).

Wprowadzone dodatkowe wewnętrzne wejścia modułu pozwalają na połączenie koncepcji modułu z koncepcją tablicy ogłoszeń w postaci zaproponowanej dla dynamicznych systemów doradczych [19].

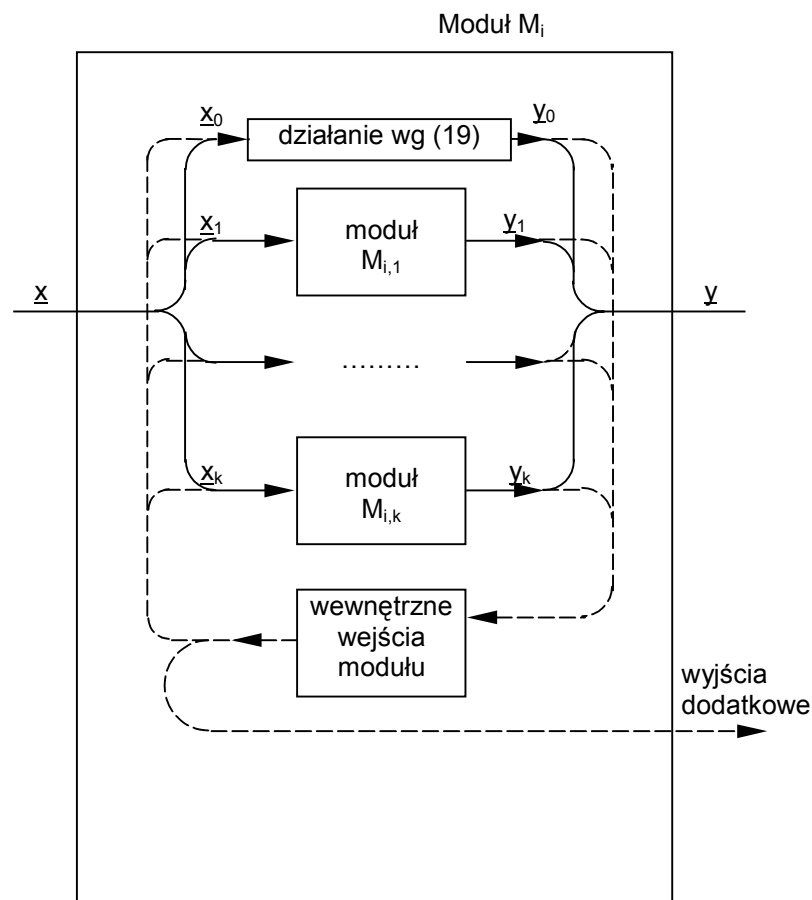
Istotą takiej tablicy jest to, że składa się ona ze stwierdzeń zapisywanych w postaci pary składającej się ze stałej treści stwierdzenia oraz zmiennej jego wartości, gdzie wartość może być np. liczbą rzeczywistą z przedziału  $[0,1]$ . Wartość stwierdzenia równa 1.0 oznacza, że odpowiadające tej wartości stwierdzenie (o znanej treści) może być uznane za prawdziwe. Wartość 0.0, oznacza stwierdzenie, które nie może być uznane za prawdziwe. Wartości pośrednie oznaczają stwierdzenia częściowo uznane za prawdziwe.

Założono, że wartościami wszystkich wewnętrznych wejść modułu są tak interpretowane wartości stwierdzeń. Moduł podstawowy zawierający moduły wewnętrzne oraz wejścia wewnętrzne pozwala na przekształcanie wejść  $\underline{x}$  w wyjścia  $\underline{y}$ . Rozwiązanie takie nie pozwala jednak na bezpośrednie modelowanie procesu propagacji stanu.



Rys. 14. Moduły wewnętrzne zawarte w module podstawowym

W rzeczywistych układach na stan elementu może wpływać wiele czynników. Mogą to być czynniki zewnętrzne, zależne od otoczenia, w którym działa element oraz czynniki wewnętrzne związane z budową samego elementu. Czynniki związane z budową samego elementu i jego działaniem można w pewien sposób przewidzieć i odpowiednio zamodelować wewnątrz modułu. Dużo trudniejszym zadaniem jest określenie wpływu czynników zewnętrznych. W grupie czynników zewnętrznych, mających wpływ na stan elementu można wyróżnić czynniki, których źródłem są inne elementy działające wewnątrz tego samego obiektu. W szczególności źródłami takich czynników są stany innych elementów i to nie tylko elementów znajdujących się w bezpośrednim sąsiedztwie danego elementu, ale również elementów znajdujących się w innych miejscach obiektu. Opisane oddziaływania powodują, że zmiana stanu jednego elementu obiektu może być przyczyną (może pociągać za sobą) zmiany stanu innych elementów. Takie następstwo zdarzeń można rozpatrywać jako propagację stanów w obiekcie. Opis procesu propagacji stanu, występującego wewnątrz obiektu, może dostarczyć wielu ważnych dla użytkownika informacji, dotyczących stanu całego modelowanego obiektu.

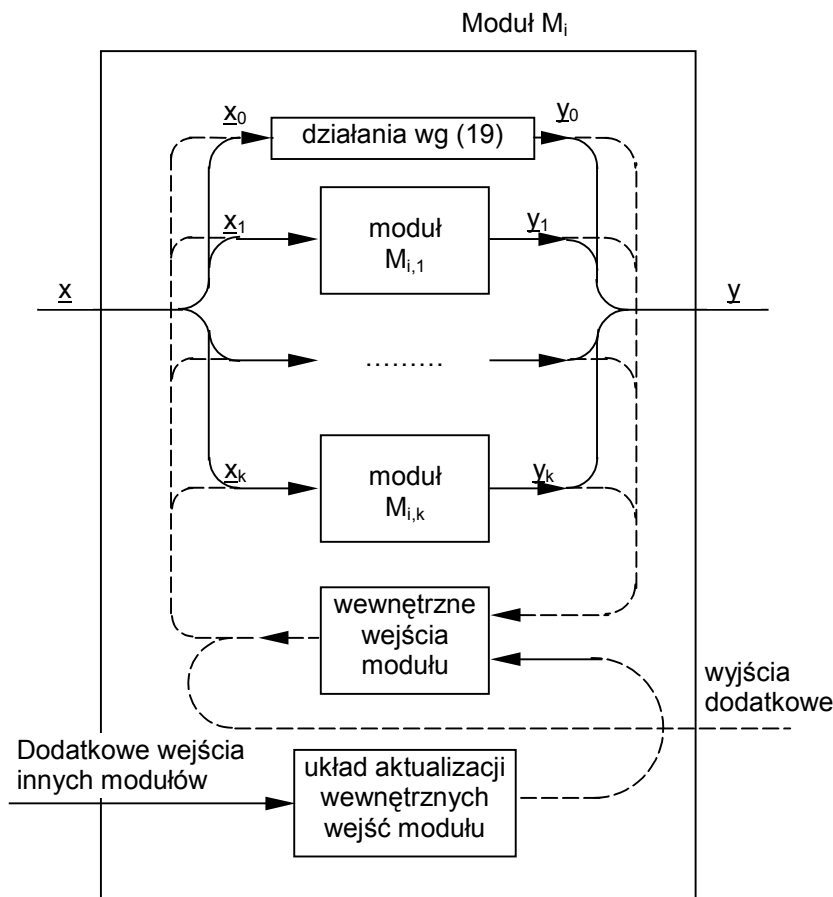


Rys. 15. Wewnętrzne wejścia modułu podstawowego

W celu umożliwienia skutecznego modelowania procesu propagacji stanu za pomocą omawianego modułu podstawowego zaproponowano rozwiązanie polegające na uzgadnianiu przez moduły informacji o wartościach wewnętrznych wejść modułów reprezentujących wartości stwierżeń o cechach stanu obiektu. Przyjęto, że każdy moduł zawierać będzie układ aktualizacji wewnętrznych wejść modułu, który będzie mógł sięgać do wartości wewnętrznych wejść w innych, sąsiadujących z nim, modułach (rys. 16). Moduły sąsiadujące będą wskazywane w fazie konstruowania modelu.

Domyślnie, modułami sąsiadującymi dla rozpatrywanego modułu są następujące trzy grupy modułów:

- moduł nadrzędny (moduł, w którym rozpatrywany moduł zawarty jest jako moduł wewnętrzny),
- moduły posiadające wspólny z rozpatrywanym modułem moduł nadrzędny (moduły, będące modułami wewnętrznymi w tym samym module, w którym rozpatrywany moduł występuje jako moduł wewnętrzny),
- moduły podrzędne (moduły zawarte jako moduły wewnętrzne



Rys. 16. Układ aktualizacji wewnętrznych wejść modułu podstawowego



w rozpatrywanym module).

Wprowadzony układ aktualizacji wewnętrznych wejść modułu zawiera między innymi lokalną bazę wiedzy, ustalającą istotę działania tego układu. Określenie „lokalna” podkreśla, że baza ta jest formułowana odpowiednio do potrzeb modułu. Baza ta określa sposoby modyfikacji wewnętrznych wejść modułu i ustala warunki, w jakich taka modyfikacja ma lub może być dokonywana.

Zaproponowane rozwiązanie prowadzi do modułu hybrydowego, w którym występują zarówno elementy odpowiadające koncepcjom wprowadzonym przez ogólną teorię systemów jak i elementy będące przykładem zastosowania koncepcji wprowadzonych w dziedzinie sztucznej inteligencji.

Moduły podstawowe zawierające elementy pokazane na (rys. 16) pozwalają na pośrednie modelowanie zależności występujących pomiędzy cechami różnych stanów cząstkowych. Ciągi zmian wartości stwierdzeń o wartościach tych cech oraz zmian wywołanych przez wcześniejsze modyfikacje wartości stwierdzeń pozwalają na pośrednie modelowanie procesu propagacji stanu.

### **3.7. Lokalna baza wiedzy**

Lokalna baza wiedzy związana jest bezpośrednio z modułem i określana jest indywidualnie dla każdego modułu. Treścią lokalnej bazy wiedzy są zasoby, określające sposób modyfikowania wewnętrznych wejść modułu oraz warunki, w jakich taka modyfikacja może być dokonywana.

Przyjmując założenia dotyczące lokalnej bazy wiedzy, zastosowano rozwiązanie wprowadzone dla sieci stwierdzeń w dynamicznych systemach doradczych [19]. Istota tego rozwiązania polega na tym, że reguły, wiążące przesłanki z konkluzjami zapisywane są w postaci nierówności pomiędzy wartościami stwierdzeń odpowiadających przesłankom i konkluzjom.

Reguły są najczęściej stosowaną formą reprezentacji wiedzy. Aby możliwe było właściwe formułowanie reguł, które wiążą między sobą przesłanki z konkluzjami, konieczne jest rozróżnianie reguł, określających warunki dostateczne (wystarczające) oraz warunki konieczne. Niech będą dane dwa stwierdzenia  $a$  oraz  $c$ , o wartościach odpowiednio  $b(a)$ ,  $b(c)$ . Możemy przyjąć, że  $a$  jest warunkiem dostatecznym dla  $c$ , jeżeli uznaniu prawdziwości stwierdzenia  $a$  towarzyszy zawsze uznanie prawdziwości stwierdzenia  $c$ , lecz niekoniecznie odwrotnie. Jednocześnie  $c$  jest warunkiem koniecznym dla  $a$ . Jeżeli  $a$  jest jednocześnie warunkiem koniecznym i dostatecznym dla  $c$ , to jednocześnie  $c$  jest warunkiem koniecznym i dostatecznym dla  $a$ . Na podstawie przeprowadzonego rozumowania można zauważyć, że [19]:

- uznanie prawdziwości stwierdzenia  $a$ , będącego warunkiem dostatecznym dla stwierdzenia  $c$  pociąga za sobą uznanie prawdziwości stwierdzenia  $c$ ,
- uznanie fałszywości stwierdzenia  $a$ , będącego warunkiem dostatecznym dla stwierdzenia  $c$  nie pociąga za sobą rozstrzygnięć dotyczących uznania lub odrzucenia stwierdzenia  $c$ ,

oraz

- uznanie fałszywości stwierdzenia  $c$ , będącego warunkiem koniecznym dla stwierdzenia  $a$  pociąga za sobą uznanie fałszywości stwierdzenia  $a$ ,
- uznanie prawdziwości stwierdzenia  $c$ , będącego warunkiem koniecznym dla stwierdzenia  $a$  nie pociąga za sobą rozstrzygnięć dotyczących uznania lub odrzucenia stwierdzenia  $a$ .

Takie rozumowanie prowadzi do spostrzeżenia, że dla stwierdzeń  $a$  oraz  $c$  o wartościach odpowiednio:

$$b(a) \in [0, 1] \quad (20)$$

$$b(c) \in [0, 1] \quad (21)$$

informację o tym, że  $a$  jest warunkiem dostatecznym dla  $c$ , można zapisać w postaci następującej nierówności:

$$b(c) \geq b(a) \quad (22)$$

Poprawność nierówności (22) można uzasadnić tym, że po uwzględnieniu (20)(21), prowadzi ona do następujących spostrzeżeń [19]:

- z uznania prawdziwości stwierdzenia  $a$ , czyli z wartości  $b(a)=1$ , wynika  $b(c)=1$ , czyli uznanie prawdziwości stwierdzenia  $c$ ,
- z uznania fałszywości stwierdzenia  $a$ , czyli  $b(a)=0$  wynika jedynie  $b(a) \geq 0$  czyli nic nie znaczący wniosek trywialny dla stwierdzenia  $c$ ,

Podobnie można zapisać informację o tym, że  $a$  jest warunkiem koniecznym dla  $c$  w następujący sposób:

$$b(c) \leq b(a) \quad (23)$$

Weryfikując (23) z uwzględnieniem (20) i (21), należy zaznaczyć, że [19]:

- z uznania fałszywości stwierdzenia  $a$ , czyli z wartości  $b(a)=0$  wynika  $b(c)=0$ , czyli uznanie fałszywości stwierdzenia  $c$ ,

- z uznania prawdziwości stwierdzenia  $a$ , czyli z wartości  $b(a)=1$  wynika jedynie  $b(c)\leq 1$ , czyli nic nie znaczący wniosek trywialny dla stwierdzenia  $c$ .

W niektórych zastosowaniach może okazać się, że konieczne będzie uwzględnienie warunków przybliżonych. Przybliżoność warunków rozumiana jest tutaj w tym sensie, że są one odpowiednio warunkami koniecznymi lub dostatecznymi z ograniczoną dokładnością. Przybliżoną wersję warunku dostatecznego (22) można zapisać w postaci [19]:

$$b(c) \geq b(a) - \delta \quad (24)$$

Analogicznie przybliżoną wersję warunku koniecznego (23) można zapisać w postaci następującej nierówności [19]:

$$b(c) \leq b(a) + \delta \quad (25)$$

gdzie  $\delta$ , jest przyjmowaną arbitralnie lub wyznaczaną w wyniku optymalizacji wielkością stałą:

$$\delta \in [0, 1] \quad (26)$$

Kryterium optymalizacji jest minimalna wartość  $\delta$ . Warunki dokładne (22)(23) są szczególnym przypadkiem warunków przybliżonych (24)(25) dla:

$$\delta = 0 \quad (27)$$

Dla stwierdzeń można również określić operatory koniunkcji i alternatywy stwierdzeń. W wyniku występowania między stwierdzeniami relacji koniunkcji:

$$d = a \text{ AND } c \quad (28)$$

ograniczenia dla wartości stwierdzenia  $d$ , zastępującego stwierdzenia  $a$  i  $c$ , można zapisać w postaci zależności:

$$b(d) = \min(b(a), b(c)) \quad (29)$$

Podobnie określa się alternatywę stwierżeń. Jeżeli pomiędzy stwierdzeniami występuje relacja alternatywy:

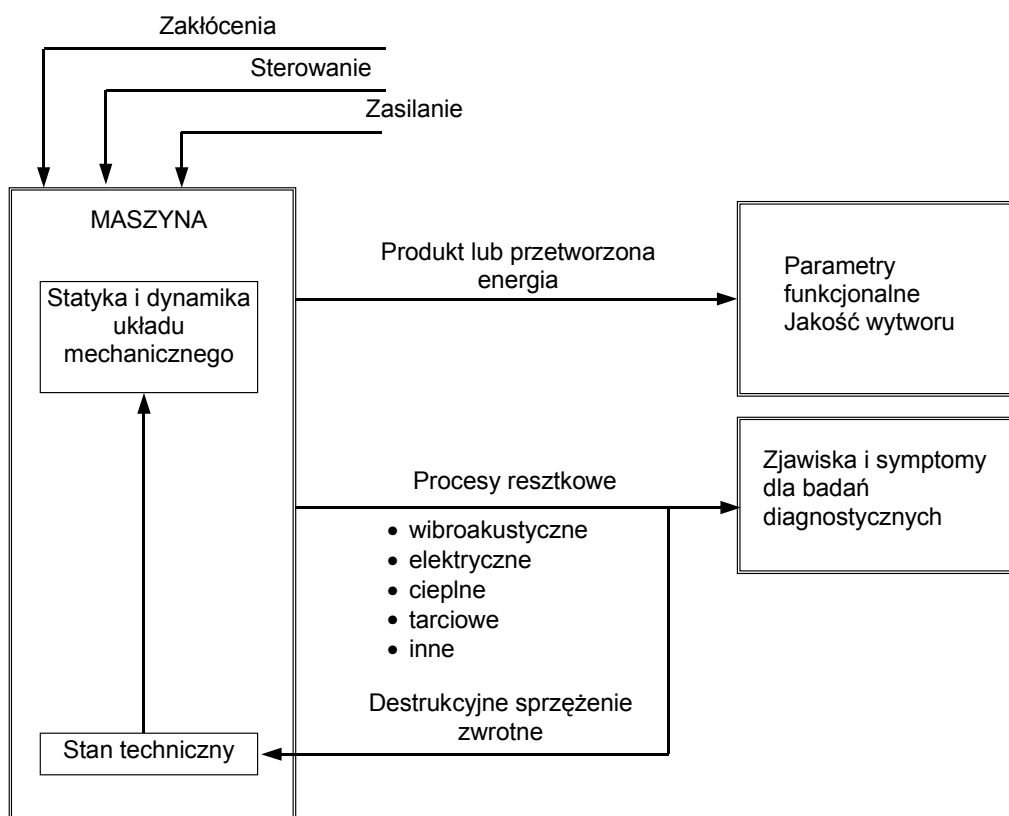
$$d = a \text{ OR } c \quad (30)$$

to ograniczenia dotyczące wartości zastępującego je stwierdzenia  $d$ , można zapisać w postaci zależności:

$$b(d) = \max(b(a), b(c)) \quad (31)$$

### 3.8. Reprezentacja stanów elementów układu

W procesie diagnozowania maszyn przeprowadza się obserwację badanej maszyny. Zazwyczaj obserwuje się chwilowe wartości cech zewnętrznych (względem maszyny). Te chwilowe wartości są odzwierciedleniem obserwowanych zjawisk. Obserwacje te można również odnieść do zjawisk zachodzących wewnątrz obserwowanej maszyny, gdyż stanowią jak gdyby jej obraz. Można zatem powiedzieć, że w badaniach tego typu obserwuje się chwilowy stan układu (określony przez jego cechy wewnętrzne) na podstawie chwilowych wartości cech sygnałów zewnętrznych [22]. W badaniach diagnostycznych działanie maszyny można

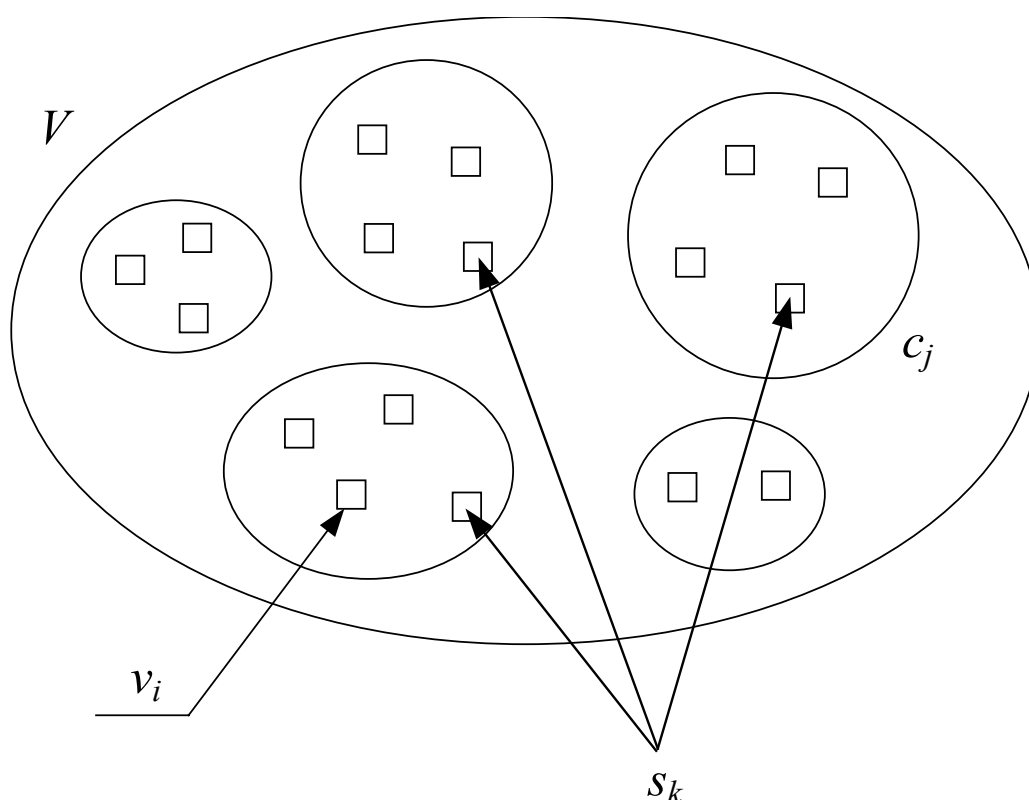


Rys. 17. Model maszyny jako systemu działającego [10]

rozpatrywać jako model: „maszyny jako systemu działającego” [10] [9] [22] [24]. Takie podejście, w kontekście zagadnień związanych z diagnostyką maszyn, pozwala w sposób jawny (w postaci wejść, wyjść przekształceń i sprzężeń zwrotnych) opisać system „mechaniczny” maszyny oraz uwzględnić zakłócenia (odpowiadające czynnikom losowym oraz przyjętym uproszczeniom). Model ten jednakże pomija inne zjawiska, jak np.: elektryczne, cieplne, przepływowe itp., które mogą mieć wpływ na działanie modelowanego układu. Tak więc stan maszyny określany jest nie tyle na podstawie cech stanu, ile na drodze oceny cech oddziaływań pomiędzy maszyną i otoczeniem. W efekcie tego określenie stanu maszyny wiąże się z obserwacją pewnych procesów „równoległych w czasie” do procesu eksploatacji maszyny, takich jak [22]:

- proces reprezentowany przez sekwencję stanów nośnika informacji o badanej maszynie,
- proces reprezentowany przez sekwencję zmian stanów tego nośnika w czasie.

Z obrazu zjawisk obserwowanych na zewnątrz rozpatrywanego układu można odwzorować obraz zjawisk zachodzących wewnątrz tego układu. Te obrazy zjawisk wewnętrznych odpowiadają stanom obserwowanego całego układu lub jego fragmentów. Należy zwrócić uwagę, że istotne jest nie tylko



Rys. 18. Przestrzeń wartości cech stanów

określenie stanu, ale również zmian stanów, jakie w układzie, lub jego fragmentach, mogą wystąpić.

Stan układu może być określany na podstawie stanów modułów, z których układ jest złożony. Powstaje pytanie, w jaki sposób określić stan modułu. Niech będzie dany zbiór wartości cech stanu  $V$ . Cecha stanu  $c_j$  jest definiowana jako podzbiór, zbioru wartości cech stanu  $V$ ,

$$c_j \subset V \quad (32),$$

przy czym, cecha stanu  $c_j$ , jest podzbiorem takich wartości  $v_i$ , które się wzajemnie wykluczają. W ten sposób otrzymuje się zbiór cech stanu  $C$ .

Stan modułu może być opisany poprzez kilka różnych cech stanu, które przyjmują jedną z dopuszczalnych wartości. Niech będzie dany zbiór wartości cech stanu  $V$ :

$$V = \{v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8, v_9, v_{10}\} \quad (33)$$

oraz zbiór cech stanu  $C$ :

$$C = \{c_1, c_2, c_3\} \quad (34)$$

w którym poszczególne cechy stanu zdefiniowano następująco:

$$\begin{aligned} c_1 &= \{v_1, v_2, v_3\} \\ c_2 &= \{v_4, v_5\} \\ c_3 &= \{v_8, v_6, v_9\} \end{aligned} \quad (35)$$

Moduł może znaleźć się w jednym ze stanów, należących do zbioru stanów  $S$ :

$$S = \{s_1, s_2, s_3\} \quad (36)$$

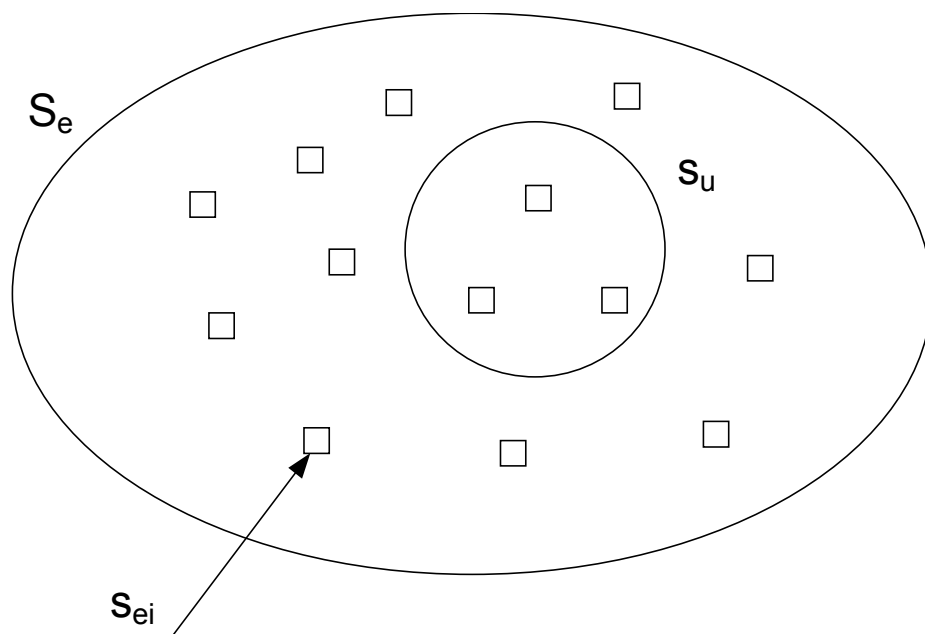
Niech stan  $s_1$  będzie opisany cechami  $c_1$ ,  $c_2$  oraz  $c_3$ , stan  $s_2$  cechami  $c_1$  oraz  $c_2$ , a stan  $s_3$  cechami  $c_1$  i  $c_3$ . Przyjmuje się, że moduł znajduje się w stanie  $s_1$ , jeżeli cecha  $c_1$  ma wartość  $v_1$ , cecha  $c_2$  ma wartość  $v_4$  a cecha  $c_3$  ma wartość  $v_8$ . Podobnie definiuje się pozostałe stany. Ogólnie można powiedzieć, że stan  $s_k$  opisany zbiorem cech  $c_{sk}$  wystąpi wtedy, gdy cechy należące do zbioru  $c_{sk}$  przyjmą określone wartości.

Wartości cech stanu mogą zmieniać się w sposób dyskretny lub ciągły. W pracy przyjęto, że rozpatrywane będą takie wartości, które zmieniają się

w sposób dyskretny. Nie ogranicza to jednak przedstawionego opisu, gdyż wartości ciągłe można poddać dyskretyzacji.

Stan całego układu można rozpatrywać jako zbiór rozłącznych i różnych stanów elementarnych. Niech  $S_e$  oznacza zbiór wszystkich stanów elementarnych  $s_{ei}$ . Oznaczmy również stan układu jako  $s_u$ . Można zatem zapisać, że  $s_u \subset S_e$ . Stan układu  $s_u$  jest więc podzbiorem zbioru wszystkich stanów elementarnych  $S_e$ . Przestrzenią stanów za [22] jest zbiór zawierający wszystkie możliwe stany  $S_u$ . W szczególnym przypadku podzbiór  $s_u$  może być zbiorem jednoelementowym, co oznacza, że stan elementarny  $s_{ei}$  może być traktowany jako stan układu.

Bardzo ważnym czynnikiem, który należy uwzględnić w tego typu zagadnieniach, jest czas. Pojęcie stanu układu jest ściśle związane z pojęciem czasu. Opisane powyżej rozważania dotyczą określenia stanu w pewnej określonej chwili czasu. W badaniach diagnostycznych naturalne pojęcie czasu jest utożsamiane z pojęciem czasu rzeczywistego. Jednakże ze względu na dwoisty charakter realizowanych zadań (chwilowe i czasowe przetwarzanie informacji diagnostycznej) przy określaniu stanu układu wprowadzono pojęcia czasu mikro ( $\tau$ ) i czasu makro ( $t$ ) [16][22]. Zaproponowano taki podział, ponieważ inna jest dziedzina czasu, w której szacowane są „chwilowe” wartości cech sygnałów, a w innej dziedzinie obserwowane są zmiany stanu rozpatrywanego układu. Wprowadza się tutaj również pojęcie stanu chwilowego określanego w dziedzinie czasu makro -  $t$ .



Rys. 19. Przestrzeń stanów [22],  $S_e$  – zbiór stanów elementarnych  $s_{ei}$ ,  $S_u$  – stan maszyny

Poszczególne stany układu  $s_u$  wyznaczone są w pewnych chwilach czasu  $t$ . Stany powinny być rozpatrywane z uwzględnieniem kolejności ich występowania. Dlatego też zarówno stany jak i zmiany stanów można uporządkować wg czasu ich występowania, a stan układu  $s_u$ , w danej chwili  $t$ , można przedstawić w przestrzeni stanów  $S_u$  w postaci [22]:

$$s_u(t) = \{s_{u,t_1}, s_{u,t_2}, \dots, s_{u,t_i} \in S_u, t_1 < t_2 < \dots\} \quad (37)$$

gdzie:

$S_{u,t_i}$  - stan układu w chwili  $t_i$

Całą przestrzeń stanów można podzielić na klasy stanów, grupujące te cechy stanów, które są wspólne dla całej grupy. Poszczególne klasy będą się więc różniły pomiędzy sobą w zakresie tych cech, które nie były podstawą ich grupowania. W najprostszym przypadku, badając stan układu, rozróżnia się dwie klasy stanów. Są to klasy określane mianem: *sprawny* lub *niesprawny*. Jednakże w przypadkach bardziej złożonych mamy do czynienia z większą liczbą klas stanów. Każdej klasie stanów można przypisać nazwę stanu, która będzie identyfikowała daną klasę. Można na przykład wyróżnić następujące klasy stanów:

- stan niedopuszczalny,
- stan tolerowany,
- stan dopuszczalny,
- stan dobry.



## **4. Implementacja modelu propagacji stanów**

### **4.1. Modelowanie stanów za pomocą Matlab Stateflow**

Do zaimplementowania opracowanego modelu propagacji stanów wybrano środowisko MATLAB [56] oraz Simulink [87] wraz z modułem Stateflow [89], służącym do modelowania układów o skończonej liczbie stanów. Program Stateflow pozwala na modelowanie układów z zastosowaniem notacji schematów przepływowych oraz za pomocą zmian stanów. Program ten umożliwia realizację założeń, dotyczących modelowania zarówno stanów, zmian stanów jak również modułów i oddziaływań występujących między nimi. Model każdego modułu ma za zadanie wykonanie działań związanych z wyznaczeniem wartości stwierdzeń na podstawie danych otrzymanych z układu zbierania sygnałów diagnostycznych. Zadaniem modułów jest również wyznaczenie wartości stwierdzeń, zależnych od wartości innych stwierdzeń z tego samego lub innych modułów. Stwierdzenia takie służą do opisu relacji łączących moduły modelowanego układu.

### **4.2. Założenia do implementacji modelu**

#### **4.2.1. Elementy podstawowe programu Stateflow**

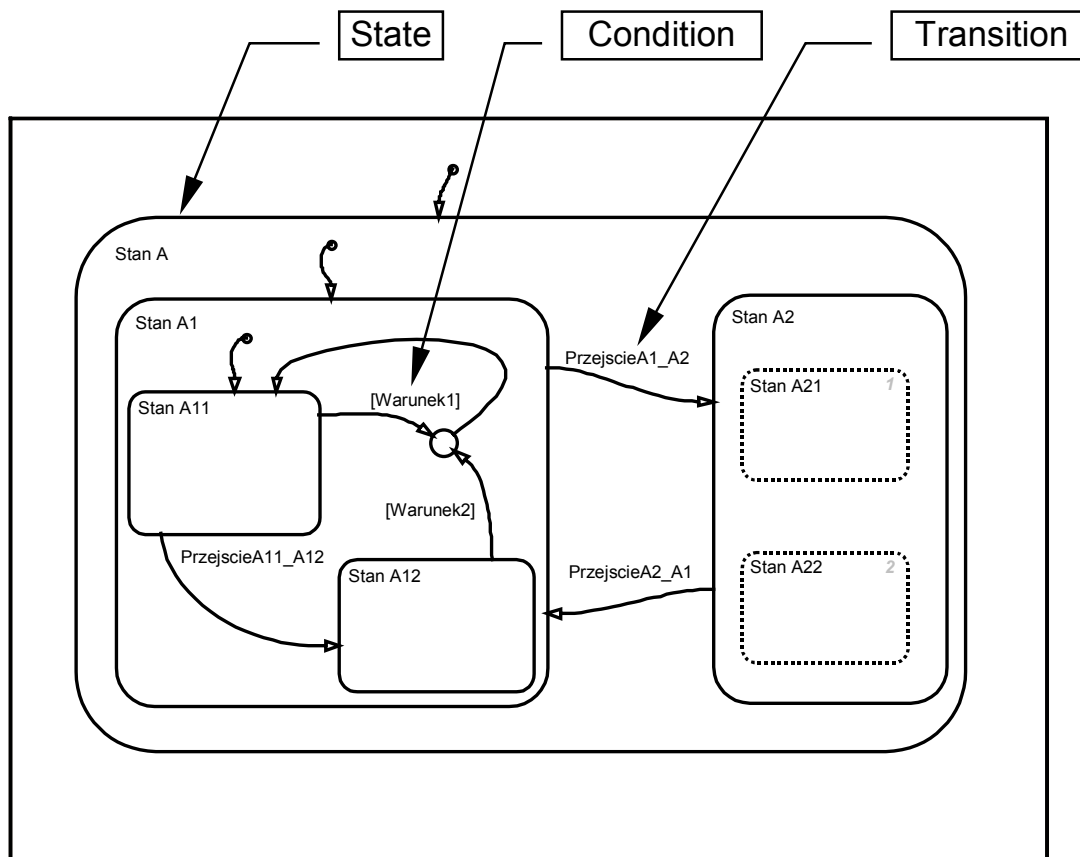
Program Stateflow jest narzędziem służącym do projektowania i rozwijania zadań związanych ze sterowaniem i nadzorem. Projektowanie i rozwijanie oprogramowania odbywa się w tym programie za pomocą odpowiednio przygotowanych graficznych symboli. Pozwala on między innymi na [89]:

- wizualizację modelu oraz symulację złożonych, biernych systemów bazujących na teorii maszyn o skończonej liczbie stanów,
- projektowanie i rozwijanie zdeterminowanych systemów nadzoru zdalnego,
- jednoczesne zastosowanie notacji schematów przepływowych i zmian stanów na jednym schemacie,
- modyfikowanie projektu, ocenę wyników i weryfikację opracowywanego systemu na każdym etapie jego projektowania.

W tradycyjnym podejściu relacje pomiędzy wejściem, wyjściem i stanem automatu skończonego są reprezentowane w postaci tablicy. Tablica wynikowa opisuje układ logiczny, niezbędny do kontrolowania zachowania się modelowanego systemu. Innym podejściem do projektowania systemów sterowanych zdarzeniami jest opisanie zachowania się modelowanego

systemu poprzez zdefiniowanie stanów i przejść między nimi. Aktywny stan jest określany na podstawie występowania określonych zdarzeń przy jednoczesnym spełnieniu określonych warunków. W programie Stateflow zastosowano opis działania automatu skończonego, jaki zaproponował w swoim opracowaniu Harel [40][32]. Za pomocą programu Stateflow przygotowywane są odpowiednie schematy. Schemat taki stanowi graficzną reprezentację automatu skończonego. Na takim schemacie podstawowymi elementami są stany i przejścia.

Stateflow pozwala również na zapisanie struktury hierarchicznej systemu, działania równoległego oraz historii. Zapisanie struktury hierarchicznej pozwala na modelowanie złożonych systemów poprzez zdefiniowanie powiązań typu „element nadrzędny – element podrzędny”. Działanie równoległe daje możliwość równoległego występowania kilku stanów w tym samym czasie.



Rys. 20. Podstawowe elementy programu Stateflow

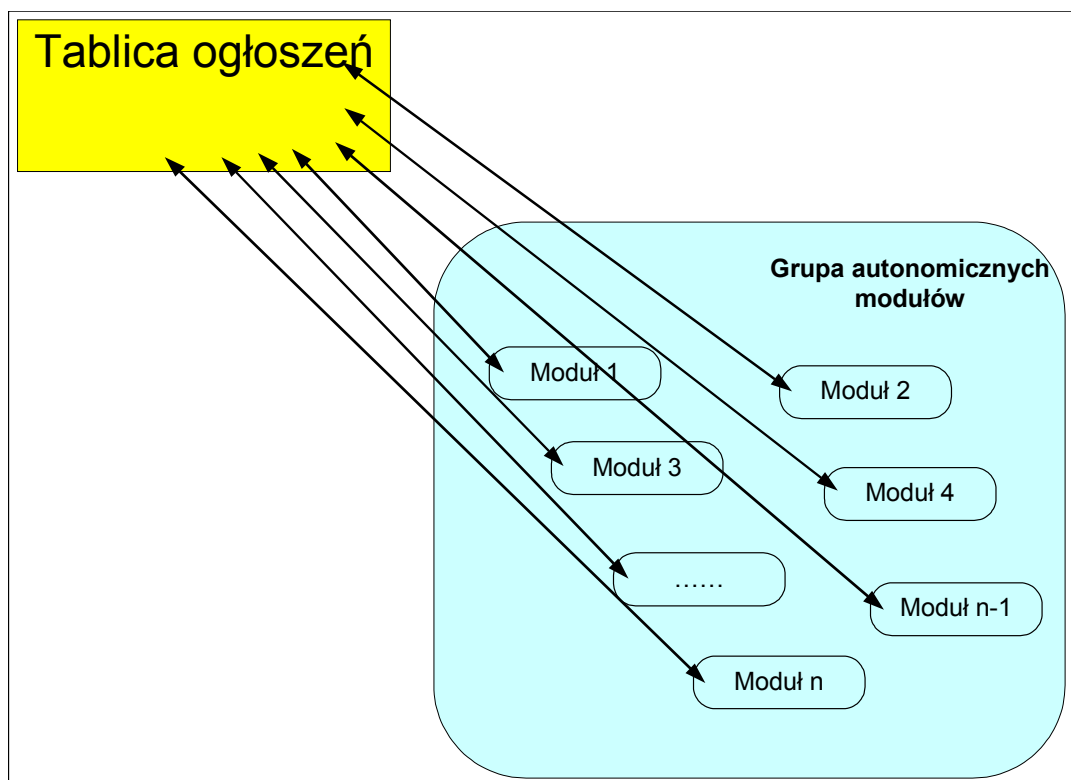
W programie Matlab Stateflow dostępne są następujące elementy podstawowe (rys. 20) służące do opracowania układów modelujących działanie automatu skończonego:

- state (stan) - element (pokazany na schemacie w postaci prostokąta), służący do zapisu obiektów będących stanami lub elementami modelowanego układu,
- transition (przejście) - element pozwalający na zapisanie relacji łączących dwa obiekty,
- junction (węzeł) - element umożliwiający zapisanie relacji złożonej lub warunkowej,
- event (zdarzenie) - pozwala na zdefiniowanie zdarzeń sterujących działaniem modelu,
- data (dane) - umożliwia zapisanie danych wykorzystywanych w modelu.

Elementy te posłużą do zdefiniowania poszczególnych fragmentów przygotowanego modelu.

#### 4.2.2. Założenia

Opracowany model propagacji stanów opiera się na autonomicznym działaniu współdziałających modułów. Każdy z modułów jest opracowany na podstawie założeń przedstawionych w rozdziale 3.6. Zadaniem każdego z modułów jest wyznaczenie wartości zbioru stwierdzeń, które zostały mu



Rys. 21. Ogólna struktura modelu.

przypisane. Moduły zostały tak zaprojektowane, że wszystkie stwierdzenia przypisane danemu modułowi są zapisywane na tablicy ogłoszeń, a działanie modułu opiera się na aktualizacji wartości stwierdzeń na tablicy ogłoszeń.

W modelu całego rozpatrywanego układu wyróżniono dwa współdziałające ze sobą bloki (rys. 21). Pierwszy z bloków związany jest z tablicą ogłoszeń, na której umieszczane są wartości stwierdzeń. Drugi blok reprezentuje działanie wielu autonomicznych modułów. Tablica ogłoszeń została przygotowana w postaci odpowiednio zaprojektowanej bazy danych, w której przechowywane są wszystkie stwierdzenia, zdefiniowane dla potrzeb opracowanego modelu. Ponadto zawarto w niej dane dotyczące działania modelowanego obiektu, dane dotyczące autonomicznych modułów, z których zbudowany jest model oraz zależności występujące między modułami. Szczegółowy opis bazy danych przedstawiono w dalszej części pracy.

Drugi blok – grupa wielu autonomicznych modułów, zawiera definicje funkcji i procedur związanych z przygotowaniem modułów do przeprowadzenia określonych zadań. Wszystkie dane, które są potrzebne w działaniu takiego modułu zostały zapisane w bazie danych.

W opracowanym rozwiązaniu czas jest traktowany w sposób dyskretny, a działania wszystkich modułów są wykonywane w kolejnych krokach czasu.

### **4.3. Tablica ogłoszeń**

W celu przygotowania tablicy ogłoszeń, która miałaby współpracować z grupą wielu autonomicznych modułów, przeznaczonej do opisu propagacji stanów, zaprojektowano bazę danych MPNU (Model Propagacji Niesprawności Układu). Struktura bazy danych przedstawiona została na rys. 22. Tablice zdefiniowane w bazie można podzielić na dwie grupy. W pierwszej grupie znajdują się tablice opisujące układ, moduły i stwierdzenia. Druga grupa tablic to tablice związane bezpośrednio z opisami modułów. A oto szczegółowy spis tablic:

- KlasaModulu
- Moduly
- TUkład,
- TrescStwierdzen,
- TOgloszen,
- Reguly.

Poniżej przedstawiono krótką charakterystykę poszczególnych tablic.

## Tablica KlasaModulu

Tablica ta służy do definiowania klas modułów, rozpatrywanych w modelowanym układzie. Klasy te posłużą następnie do utworzenia konkretnych obiektów, związanych z poszczególnymi modułami z rzeczywistego obiektu. Tablica zawiera trzy pola:

- id\_klasy – identyfikator klasy modułu,
- nazwa – nazwa klasy modułu,
- opis – opis klasy.

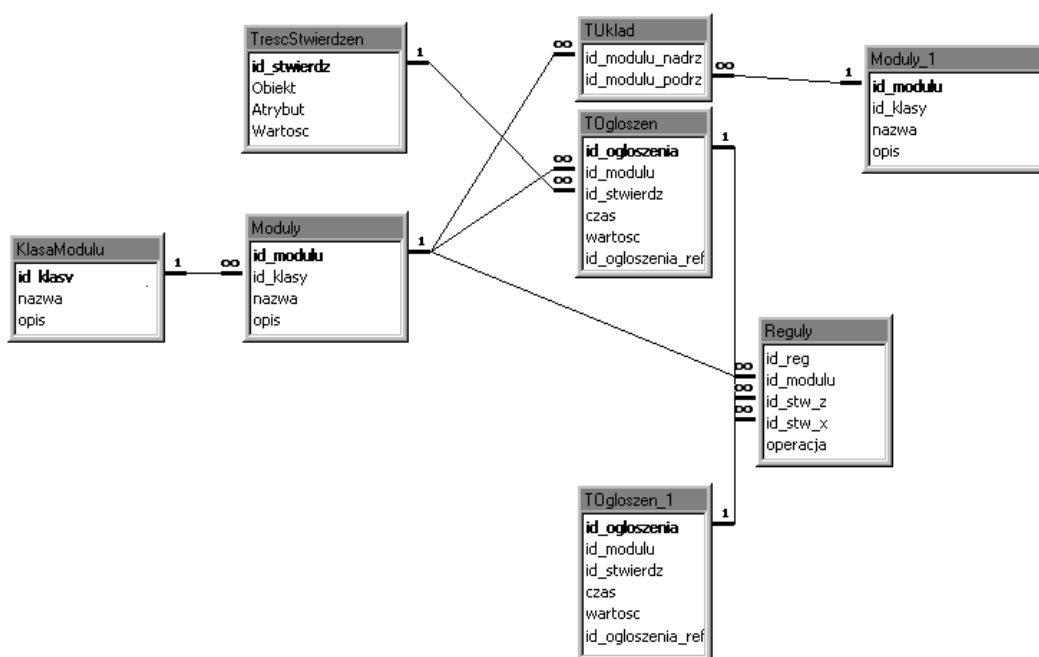
Przykładem klasy zapisanej w tej tablicy jest klasa „Łożysko toczne”. Opisuje ona wszystkie moduły będące odpowiednikami łożysk tocznych w modelowanym układzie.

## Tablica Moduły

W tej tablicy są przechowywane informacje o modułach zdefiniowanych w modelu. Tabela składa się z następujących pól:

- id\_modulu – identyfikator modułu,
- id\_klasy – identyfikator klasy modułu,
- nazwa – nazwa modułu,
- opis – opis modułu.

W tablicy tej zapisywane są konkretne moduły, będące fragmentami



Rys. 22. Struktura bazy MPNU

opracowanego modelu. Przykładem może być moduł *Łożysko Nr 1*, który odpowiada konkretnemu łożysku w modelowanym obiekcie.

### **Tablica TUklad**

Służy do definiowania relacji podrzędności (i nadrzędności) poszczególnych modułów. Tablica zawiera dwa pola:

- `id_modulu_nadrz` – identyfikator modułu nadrzędnego dla modułu `Id_modulu_podrz` (0 jeżeli moduł nie posiada modułu nadrzędnego),
- `id_modulu_podrz` – identyfikator modułu.

Kojarząc odpowiednie identyfikatory zdefiniowanych w tabeli **Moduly** modułów, zapisuje się w bazie danych strukturę rozpatrywanego układu, zgodnie z przyjętymi wcześniej założeniami. Założono, że poszczególne moduły zawierają się wewnątrz siebie, tworząc strukturę hierarchiczną.

### **Tablica TrescStwierden**

Tablica `TrescStwierden` została utworzona w celu zapisywania treści stwierdzeń definiowanych w postaci trójki uporządkowanej *<obiekt, atrybut, wartość>*. Zapisane treści stwierdzeń dotyczą wszystkich stwierdzeń, jakie mogą wystąpić w modelowanym układzie.

Tablica zawiera następujące pola:

- `id_stwierdz` – identyfikator stwierdzenia,
- `obiekt` – nazwa obiektu któremu przypisany jest atrybut,
- `atrybut` – atrybut opisujący dany obiekt,
- `wartosc` – wartość atrybutu,

### **Tablica TOgloszen**

W tabeli `TOgloszen` umieszczane są wartości stwierdzeń. Wartość stwierdzenia zapisywana jest jako para liczb opisująca stopień możliwości oraz stopień konieczności. Tablica zawiera następujące pola:

- `id_ogloszenia` – identyfikator ogłoszenia,
- `id_modulu` – identyfikator modułu, który dokonuje aktualizacji wartości danego stwierdzenia,
- `id_stwierdz` – identyfikator treści stwierdzenia,
- `czas` – czas aktualizacji wartości stwierdzenia,

- wartosc – wyznaczony w wyniku aktualizacji stopień prawdziwości stwierdzenia,
- id\_ogloszenia\_ref – identyfikator stwierdzenia, z którego ma zostać pobrana wartość lub 0 jeżeli ma być uwzględniona wartość danego stwierdzenia.

Czas jest zapisywany jako liczba sekund, które upłynęły od daty 01-01-2000.

Ponieważ niektóre wartości stwierdzeń mają być dostępne z innych modułów, a przyjęto założenie, że dany moduł może sięgać do danych jedynie z modułów bezpośrednio z nim sąsiadujących (rodzica, potomków i sąsiadów), przyjęto dodatkowe założenie pozwalające przekazywać wartości stwierdzeń do innych modułów. Jeżeli będzie to konieczne, to w pole id\_ogloszenia\_ref należy wpisać identyfikator stwierdzenia, z którego ma zostać pobrana wartość. Takie założenie pozwala na przekazywanie wyznaczanych w różnych modułach wartości stwierdzeń bez zwłoki czasowej, która pojawiłaby się, gdyby dane były kopiowane pomiędzy modułami. Domyślnie wartość tego identyfikatora jest równa zero.

### Tablica Reguly

W tej tablicy zapisywane są reguły, które opisują zależności pomiędzy stwierdzeniami. Tablica zawiera następujące pola:

- id\_reguly – identyfikator reguły,
- id\_modulu – identyfikator modułu, którego zadaniem jest wykonanie działań objętych daną regułą,
- id\_stw\_z – identyfikator stwierdzenia  $z$ ,
- id\_stw\_x – identyfikator stwierdzenia  $x$ ,
- operacja – typ relacji ustalonej przez daną regułę:
  - WK - stwierdzenie  $x$  jest warunkiem koniecznym stwierdzenia  $z$ ,
  - WD - stwierdzenie  $x$  jest warunkiem dostatecznym stwierdzenia  $z$ ,

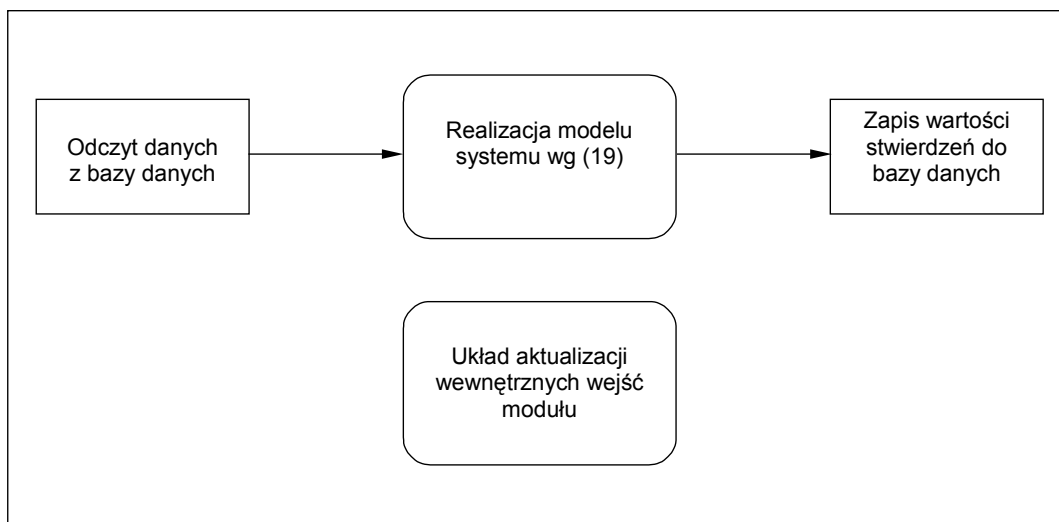
Baza MPNU pełni rolę tablicy ogłoszeń, z którą komunikuje się każdy z działających w modelu autonomicznych modułów.

Przedstawioną bazę danych można zaimplementować w dowolnym systemie relacyjnych baz danych. Ważne jest tylko to, aby były dostępne odpowiednie funkcje, pozwalające komunikować się z bazą danych za pośrednictwem standardu ODBC. W przedstawionym rozwiązaniu baza

danych MPNU została zaimplementowana w dwóch środowiskach baz danych: MS Access oraz MS SQL Server.

#### 4.4. Blok wielu autonomicznych modułów

Blok wielu autonomicznych modułów został zaimplementowany za pomocą narzędzi dostępnych w pakiecie Simulink oraz środowisku obliczeń numerycznych Matlab. Przed opracowaniem całego bloku wielomodułowego zdefiniowano pojedynczy moduł, który stał się podstawą dla definicji całego systemu. Definicja i zadania modułu podstawowego zostały opisane wcześniej. Zgodnie z przyjętą definicją modułu podstawowego opracowana została implementacja takiego modułu. W przedstawionym na rys. 23 rozwiązaniu, w module przygotowano szereg podukładów, pozwalających na



Rys. 23. Schematyczny zapis modułu podstawowego w systemie Simulink

poprawne wykonanie zadań, zgodnych z przyjętą wcześniej definicją modułu podstawowego. I tak można tutaj wyróżnić następujące podukłady:

- układ aktualizacji wewnętrznych wejść modułu,
- układ modelujący działanie modelu systemu.

##### Układ aktualizacji wewnętrznych wejść modułu

Układ aktualizacji wewnętrznych wejść modułu ma za zadanie wyznaczenie wartości stwierżeń przypisanych danemu modułowi na podstawie zdefiniowanych w bazie reguł. W bazie danych zapisane są reguły, według których mają być wyznaczone wartości stwierżeń. Każda z reguł przypisana jest modułowi, który będzie daną regułą wykonywał. Układ aktualizacji wewnętrznych wejść modułu odczytuje z bazy danych wszystkie reguły



przypisane danemu modułowi i zgodnie z nimi wykonuje określone działania. Ponieważ wyznaczone wartości stwierdzeń mogą mieć wpływ na wartości stwierdzeń wyznaczanych w krokach późniejszych, układ aktualizacji wewnętrznych wejść wyznacza wartości stwierdzeń w kilku krokach.

Pomiędzy przesłankami i konkluzjami mogą wystąpić dwa rodzaje relacji:

- WK - przesłanka jest warunkiem koniecznym dla uznania słuszności konkluzji,
- WD - przesłanka jest warunkiem dostatecznym dla uznania słuszności konkluzji.

Przesłanki występujące w regułach mogą być przesłankami prostymi lub złożonymi. Przesłanki proste zawierają tylko jedno stwierdzenie. Przesłanka złożona składa się ze stwierdzeń połączonych operatorami:

- AND - operator koniunkcji,
- OR - operator alternatywy.

Przesłanka złożona może być rozpatrywana jako nowe stwierdzenie. Dla wartości stwierdzeń zapisanych w postaci pary liczb określających stopień konieczności i stopień możliwości stwierdzenia, relacje WK i WD będą zapisywane zgodnie z wzorami przedstawionymi poniżej.

Niech będą dane stwierdzenia  $u$  oraz  $v$  o wartościach odpowiednio  $b(u)$  i  $b(v)$ , równych:

$$\begin{aligned} b(u) &= [u1; u2] \\ b(v) &= [v1; v2] \end{aligned} \tag{38}$$

Jeżeli stwierdzenie  $u$  jest warunkiem koniecznym dla stwierdzenia  $v$ , to wartość stwierdzenia  $v$  (po uwzględnieniu tego warunku) będzie wyznaczana, zgodnie z (25), z zależności:

$$b(v) = [v1; \min(u1 + \delta, v2)] \tag{39}$$

Jeżeli stwierdzenie  $u$  jest warunkiem dostatecznym dla stwierdzenia  $v$ , to wartość stwierdzenia  $v$  (po uwzględnieniu tego warunku) będzie wyznaczana, zgodnie z (24), z zależności:

$$b(v) = [\max(u2 - \delta, v1); v2] \tag{40}$$

Jeżeli wartość stwierdzenia,

$$b(v) = [v1; v2] \tag{41}$$

będzie taka, że  $v_1 > v_2$ , to będzie to oznaczało, że w zbiorze reguł istnieje sprzeczność i dla jej eliminacji należy zwiększyć wartość  $\delta$  lub sprawdzić poprawność reguł.

**Przykład:**

Niech wartości stwierżeń  $u$  i  $v$  wynoszą odpowiednio:

$$\begin{aligned} b(u) &= [0,3; 0,5] \\ b(v) &= [0; 1] \end{aligned}$$

wówczas zgodnie z (41) i (42), po uwzględnieniu warunku, wartość stwierżenia  $v$  (dla  $\delta = 0,02$ ) będzie równa:

$$\begin{aligned} b(v) &= [0; 0,32] && \text{– jeżeli } u \text{ jest warunkiem koniecznym dla } v, \\ b(v) &= [0,48; 1] && \text{– jeżeli } u \text{ jest warunkiem dostatecznym dla } v. \end{aligned}$$

Zdefiniowano również działania dla koniunkcji i alternatywy stwierżeń.

Niech będą dane stwierżenia  $u, v, w$  o wartościach  $b(u), b(v), b(w)$  równych:

$$\begin{aligned} b(u) &= [u_1; u_2] \\ b(v) &= [v_1; v_2] \\ b(w) &= [w_1; w_2] \end{aligned} \tag{42}$$

Jeżeli wartość stwierżenia  $w$  jest koniunkcją stwierżeń  $u$  oraz  $v$ , to wartość stwierżenia  $w$  będzie wyznaczana z zależności:

$$b(w) = b(u \text{ AND } v) = [\min(u_1, v_1); \min(u_2, v_2)] \tag{43}$$

Jeżeli wartość stwierżenia  $w$  jest alternatywą stwierżeń  $u$  oraz  $v$ , to wartość stwierżenia  $w$  będzie wyznaczana z zależności:

$$b(w) = b(u \text{ OR } v) = [\max(u_1, v_1); \max(u_2, v_2)] \tag{44}$$

**Przykład:**

Niech wartości stwierżeń  $u$  i  $v$  wynoszą odpowiednio:

$$\begin{aligned} b(u) &= [0,3; 0,5] \\ b(v) &= [0; 1] \end{aligned}$$

wówczas zgodnie w podanymi wyżej wzorami, wartość stwierżenia  $w$  będzie równa:

- $b(w) = [0; 0,5]$  – jeżeli wartość stwierdzenia  $w$  jest koniunkcją stwierdzeń  $u$  oraz  $v$  ( $b(w) = b(u \text{ AND } v)$ ),
- $b(w) = [0,3; 1]$  – jeżeli wartość stwierdzenia  $w$  jest alternatywą stwierdzeń  $u$  oraz  $v$  ( $b(w) = b(u \text{ OR } v)$ ),.

### Realizacja działania modelu systemu

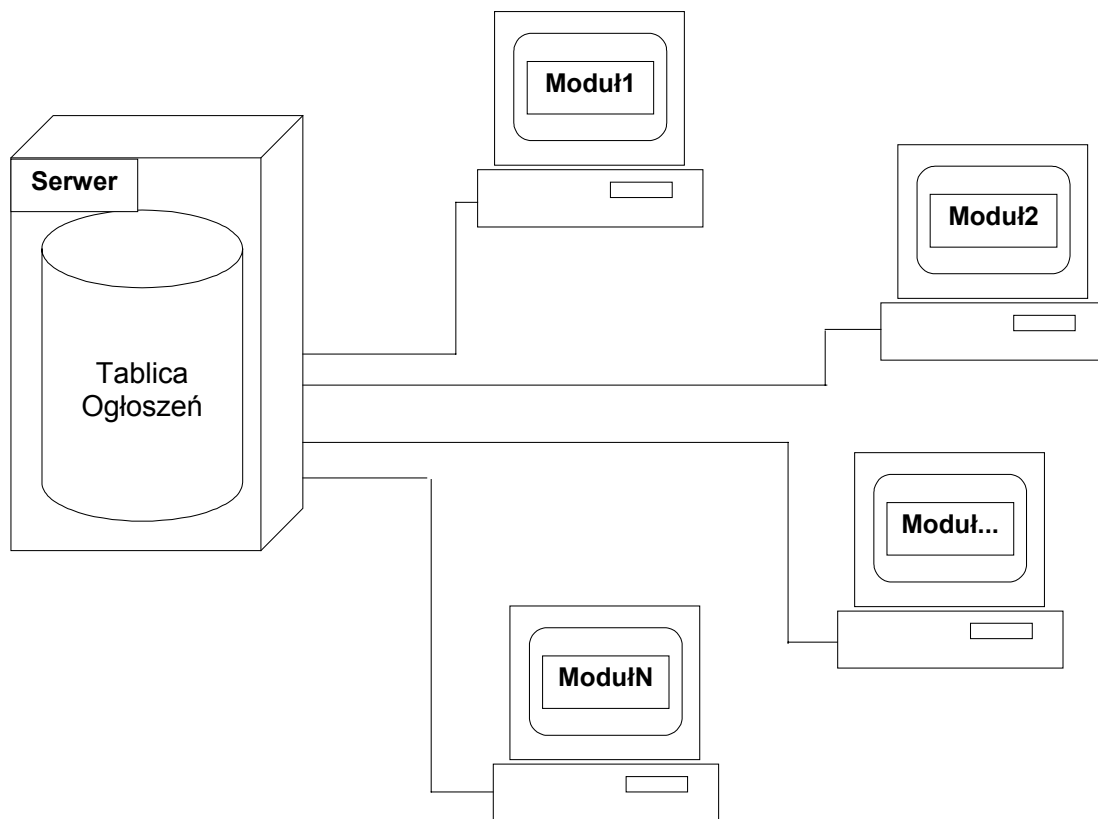
Zadaniem układu realizującego działanie modelu systemu jest wykonanie zadań zapisanych w postaci układu równań (19). W skład tego układu wchodzi podukład komunikacji z bazą danych MPNU, które pozwalają na odczytanie koniecznych do działania systemu danych oraz zapisanie wyznaczonych przez ten podukład wartości stwierdzeń.

Podukład ten ma za zadanie wyznaczenie określonych dla danego modułu wartości stwierdzeń. Został on zaimplementowany w postaci odpowiednio przygotowanego schematu opracowanego w programie Stateflow. Danymi wejściowymi do tego schematu są pobrane z bazy danych aktualne dane dotyczące działania modelowanego układu oraz wartości stwierdzeń, z których moduł ma prawo korzystać. Są to stwierdzenia, których wartości są wyznaczane przez dany moduł oraz stwierdzenia z modułów nadrzędnych, podrzędnych oraz modułów sąsiadujących z danym modułem. Na podstawie potrzebnych danych podukład ten wyznacza wybrane wartości stwierdzeń wewnętrznych modułu, które następnie są wykorzystywane przez układ aktualizacji wewnętrznych wejść modułu.

W przedstawionym rozwiązaniu komunikacja z bazą danych odbywa się z wykorzystaniem protokołu ODBC. W rozwiązaniu opartym na jednym konkretnym serwerze baz danych można zastosować do komunikacji z bazą danych inne oprogramowanie, udostępnianie przez producenta serwera. Dla każdego modułu procesy te sięgają do różnych danych, co jest związane z różnym przeznaczeniem poszczególnych modułów. Jednakże dla modułów należących do tej samej klasy, procesy te wykonują takie same zadania.

Przedstawione powyżej układy opisują działanie jednego modułu. Za pomocą modułów zdefiniowanych w ten sposób tworzy się system wielu niezależnie działających modułów, który jest implementacją modelu propagacji stanów. Działanie systemu wielu modułów można zrealizować dwoma sposobami. Pierwszy sposób to uruchomienie zespołu modułów na jednym komputerze, łącznie z bazą danych MPNU, pełniącą między innymi rolę tablicy ogłoszeń. Takie rozwiązanie jednak nie wykorzystuje w pełni modułowej budowy systemu. Spowoduje nadmierne obciążenie jednostki obliczeniowej, na której zostaną uruchomione wszystkie procesy, szczególnie przy dużej liczbie działających modułów. Dodatkowym obciążeniem tego komputera będzie działający również na nim system zarządzania bazą danych.

Znacznie lepszym rozwiązaniem jest rozproszenie działających modułów i umieszczenie ich na różnych komputerach. Na rys. 24 przedstawiono takie przykładowe rozwiązanie, które opracowano opierając się na sieci komputerowej. Na jednym komputerze (serwerze) umieszczono bazę danych pełniącą funkcję tablicy ogłoszeń. Na innych komputerach umieszczonych w tej samej sieci, w której działa serwer, umieszczone zostało oprogramowanie realizujące działania poszczególnych modułów. Każdy z komputerów podłączony do sieci umożliwia połączenie z bazą danych (komunikację modułu z tablicą ogłoszeń). Wykorzystując możliwości dostępnego sprzętu komputerowego, można na jednostkach lepiej



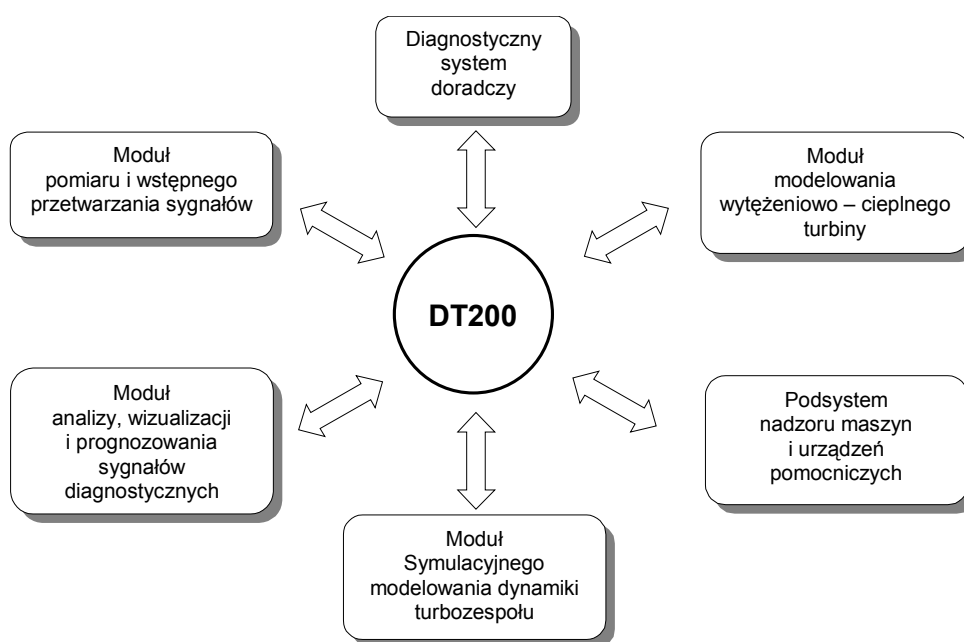
Rys. 24. Implementacja modelu w postaci systemu wielu modułów

wyposażonych uruchomić nie jeden moduł, ale grupę modułów. Zdaniem autora pracy takie rozwiązanie lepiej oddaje istotę działania układu wielu autonomicznych modułów.

## 5. Model propagacji stanów na przykładzie wybranego układu

### 5.1. Opis modelowanego układu

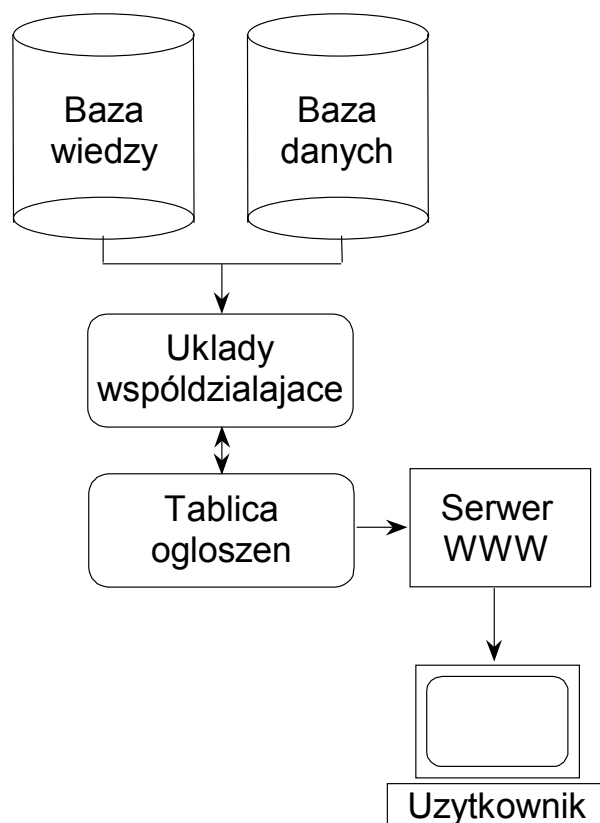
Do przetestowania poprawności działania opracowanego modelu propagacji stanów wybrano rzeczywisty obiekt badań. Jest nim turbina o mocy 200 MW zainstalowana w Elektrowni „Kozienice” S.A. Obiekt badań wybrano ze względu na dostęp do rzeczywistych danych pomiarowych i parametrów procesowych. W wyniku kilkuletnich prac, prowadzonych przez



Rys. 25. Struktura systemu DT200-1 [23]

Katedrę Podstaw Konstrukcji Maszyn Politechniki Śląskiej wraz z kilkoma innymi ośrodkami naukowymi w kraju, opracowano system diagnostyczny DT200-1 [23] w ramach projektu zamawianego PBZ-038-06. Ogólną strukturę systemu przedstawiono na rys. 25. W celu odróżnienia od pojęcia modułu wprowadzonego w rozdziale 3, elementy oznaczone na rys. 25 nazwą "moduł" będą dalej nazywane blokami. System DT200-1 składa się z kilku podstawowych bloków. Głównymi elementami systemu DT200-1 są:

- blok pomiaru, przetwarzania, analizy i prognozowania sygnałów diagnostycznych związanych z działaniem turbiny,
- blok symulacyjnego modelowania dynamiki turbiny,



Rys. 26. Uproszczony schemat układu przedstawiającego treść tablicy ogłoszeń [27]

- diagnostyczny system doradczy, współdziałający z modułami systemu i stanowiący wyposażenie stanowiska diagnostycznego,
- autonomiczny podsystem nadzoru maszyn i urządzeń pomocniczych.

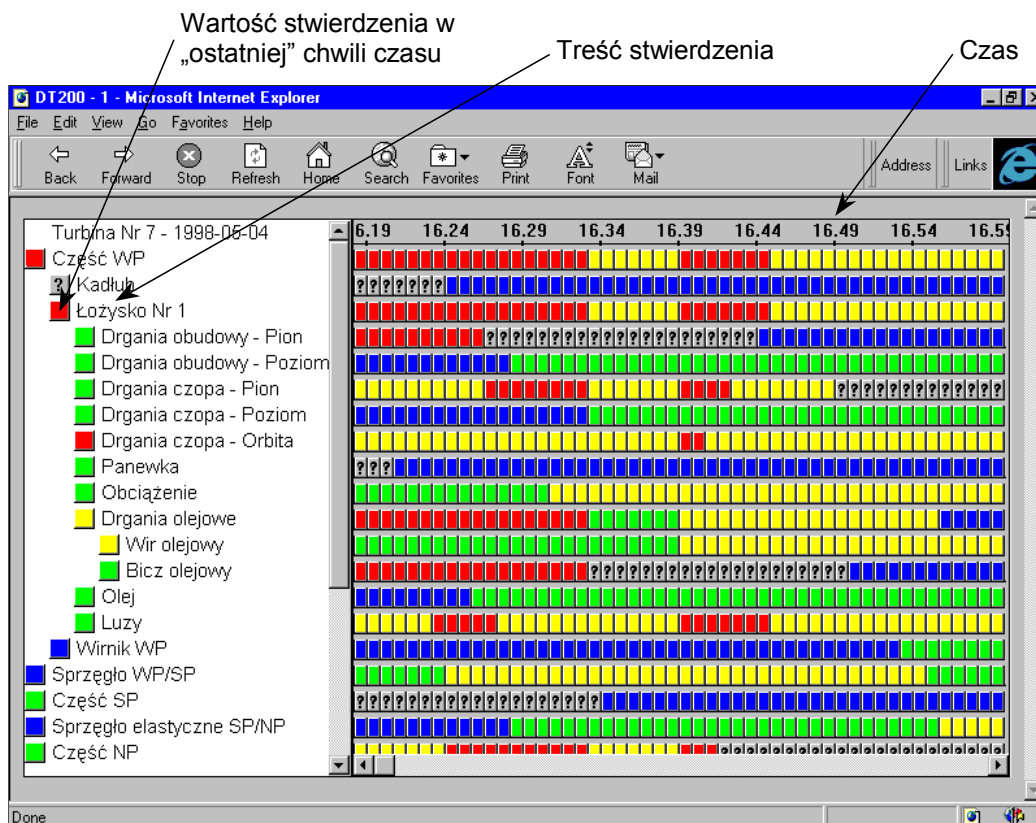
W systemie DT200-1 wyróżniono sieciowy układ akwizycji sygnałów, przeznaczony między innymi do odbierania z czujników sygnałów dynamicznych oraz ich wstępnego przetwarzania. Układ ten działa w trybie asynchronicznym, co nie pozwala na bezpośredni dostęp do opracowywanych przez ten układ danych, dlatego też do przechowywania tych danych zastosowano bazę danych. Gromadzone są w niej dane dostarczane przez układ akwizycji sygnałów, jak również dane procesowe pochodzące z układu WDPF. Zgromadzone dane posłużyły do symulowania środowiska, w którym działa model.

Jednym z elementów systemu diagnostycznego DT200-1 jest system doradczy. Opracowany system doradczy jest systemem dynamicznym, przeznaczonym do realizacji zadań w ograniczonym czasie i przy ograniczonych zasobach. Konstruując ten system, dla diagnostyki technicznej, uwzględniono uproszczenia wynikające z ograniczonej dynamiki zmian stanu technicznego obiektu rzeczywistego. System doradczy systemu DT200-1 oparty jest na pojęciu stwierdzenia [18][27][17]. Stwierdzenie jest wypowiedzią orzekającą o obserwowanych faktach lub reprezentującą

określoną opinię. W systemie tym wprowadzono oryginalne rozwiązanie polegające na zastosowaniu stwierdzeń aktywnych reprezentujących jakąś opinię o badanym obiekcie i łączących w sobie koncepcje stwierdzeń, agentów oraz ram. Stwierdzenia aktywne mogą występować jako powiązane i współdziałające węzły sieci, które ulegają stałym modyfikacjom w związku z dążeniem całej sieci do stanu równowagi. Każdy węzeł takiej sieci może być modyfikowany bezpośrednio przez zadania zewnętrzne (np. procesy realizowane przez układy pomiarowe), jak również może być dostępny dla zewnętrznych obserwatorów (np. użytkowników systemu doradczego). Przyjęto, że zmiana wartości stwierdzenia może pociągnąć za sobą zmianę wartości innych stwierdzeń, zgodnie ze stosowanym zbiorem reguł diagnostycznych. Rozwiązanie zastosowane w systemie DT200-1 opiera się na spostrzeżeniu, że tablica ogłoszeń może być modelowana jako sieć stwierdzeń oddziałujących na siebie, zgodnie ze znanym zbiorem reguł, czyli jako układ przekształcający stwierdzenia. Proces wnioskowania w takim układzie polega na aktualizacji ogłoszeń umieszczonych na tablicy.

Aplikacja przedstawiająca stan turbozespołu monitorowanego przez system diagnostyczny DT200 umożliwia między innymi:

- przedstawianie użytkownikowi wartości stwierdzeń, znajdujących się na



Rys. 27. Widok tablicy ogłoszeń przedstawiający kolorowe wartości stwierdzeń [27]

tablicy ogłoszeń,

- rozpatrywanie wartości stwierdzeń w pewnym określonym przedziale czasu,
- aktualizowanie zawartości okna głównego aplikacji zadaną, określoną częstotliwością,
- przedstawienie układu w postaci hierarchicznej, umożliwiającej użytkownikowi wybór stopnia szczegółowości przeglądania treści tablicy ogłoszeń,

Aplikacja została opracowana jako komponent ActiveX, który, umieszczony na odpowiednio opracowanych stronach WWW, jest dostępny w sieci Internet/Intranet,

Na rys. 26 przedstawiono ogólną strukturę rozpatrywanego systemu monitorującego. Dane zbierane przez układ monitorowania stanu maszyny zapisywane są w bazie danych. Opracowane układy współdziałające przetwarzają dane zgromadzone w bazie danych i, korzystając z wiedzy zapisanej w bazie wiedzy systemu doradczego ([29]), zmieniają treść tablicy ogłoszeń, na której umieszczane są wartości stwierdzeń. Stwierdzenia znajdujące się na tablicy ogłoszeń mogą być następnie prezentowane użytkownikowi.

Na serwerze WWW umieszczono odpowiednio przygotowane strony, zawierające komponenty ActiveX oraz aktywne elementy poszerzający zakres usług serwera WWW. Dzięki takiemu rozwiązaniu użytkownik dynamicznie sięga do zawartości tablicy ogłoszeń. Użytkownicy mogą sięgać do stron WWW ze swoich stacji za pomocą przeglądarki WWW.

Na rys. 27 przedstawiono postać okna przeglądarki internetowej pokazującej zawartość tablicy ogłoszeń. Okno to zostało podzielone na dwie części. Po lewej stronie znajduje się część, w której przedstawiono monitorowany układ, zapisany w postaci hierarchicznej. Elementami tej struktury są stwierdzenia. Obok każdego stwierdzenia umieszczono kolorowy kwadrat, z aktualną wartością stwierdzenia. Wartości stwierdzeń reprezentowane są za pomocą odpowiednich kolorów. Przedstawiona struktura jest aktywna i użytkownik może sam zdecydować, z jakim stopniem szczegółowości będzie prezentowana. Warto również zwrócić uwagę na fakt, że wartości stwierdzeń na określonym poziomie wpływają na wartości innych stwierdzeń znajdujących się nadrzędnie do danego stwierdzenia. Jako przykład niech posłuży stwierdzenie „Drgania czopa - Orbita”, znajdujące się na poziomie bardziej szczegółowym niż stwierdzenie „Łożysko Nr 1”. Wartością stwierdzenia „Drgania czopa - Orbita”, jest kolor „czerwony” (informujący o pewnym stopniu zagrożenia). Pozostałe stwierdzenia znajdujące się na tym samym poziomie mają wartości o niższym priorytecie. Dlatego też wartością stwierdzenia nadrzędnego dla



całej tej grupy jest również kolor „czerwony”. Takie rozwiązanie umożliwia przedstawianie stanu układu z różnym stopniem szczegółowości oraz zapewnia, że nie wystąpi zasłanianie informacji o stanie wybranego elementu wynikające z ograniczenia stopnia szczegółowości. W prawej części okna prezentowane są wyłącznie wartości stwierdzeń, odpowiadające opisom z lewej części okna. Wartości stwierdzeń przedstawiane w tej części okna są prezentowane w pewnym określonym przedziale czasu. W górnej części tego fragmentu okna znajduje się opis osi czasu.

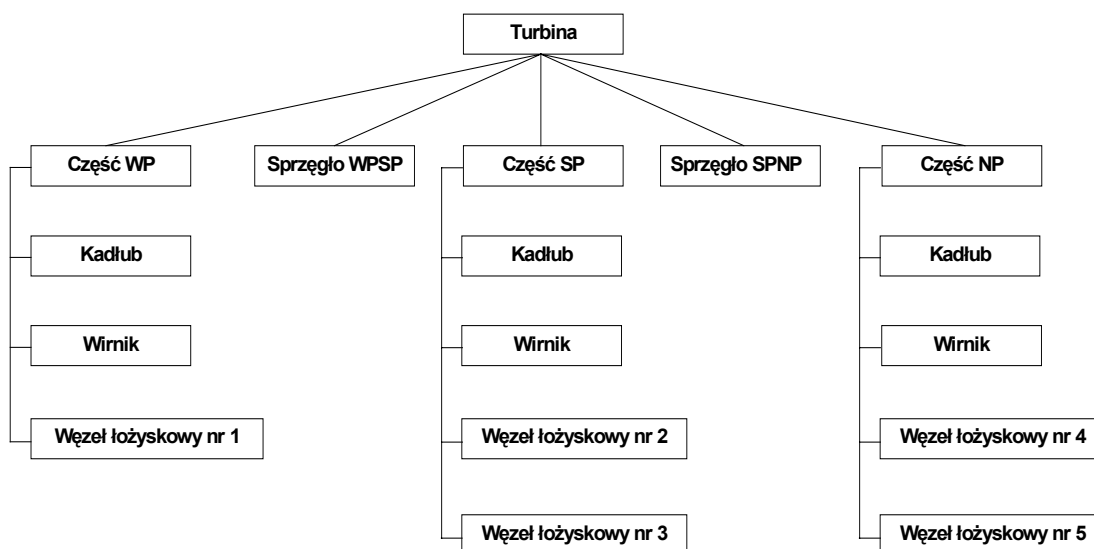
## 5.2. Analiza układu

### 5.2.1. Opis struktury układu

Rozpatrywany jest duży i bardzo złożony obiekt, jakim jest turbina 13K215 o mocy 200MW. Aby nie zaciemniać rozwiązania opartego na modelu propagacji stanów dla tego złożonego obiektu, poczyniono pewne uproszczenia. Ograniczona została liczba elementów, które wchodzi w skład opracowanego modelu.

Po przeprowadzeniu analizy rozpatrywanego układu wyróżniono następujące klasy modułów, które posłużą do opracowania struktury turbiny:

- Turbina – klasa dla wszystkich obiektów typu turbina,
- Stopień – klasa opisująca obiekty, takie jak część wysokoprężna, średnioprężna czy niskoprężna turbiny,
- Sprzęgło – opisuje obiekty typu sprzęgło,
- Kadłub – opisuje obiekty związane z kadłubami poszczególnych części turbiny,



Rys. 28. Podział turbiny na moduły

- Wirnik – reprezentuje wirniki turbiny,
- Węzeł Łożyskowy – opisuje podpory łożyskowe turbiny.

Całą turbinę podzielono na moduły odpowiadające rzeczywistym podzespołom i elementom turbiny. Schemat podziału całej turbiny na moduły przedstawiono na rys. 28. W turbinie wyróżniono pięć głównych zespołów: część wysokoprężną, część średnioprężną, część niskoprężną, sprzęgło sztywne łączące część wysokoprężną z częścią średnioprężną oraz sprzęgło elastyczne łączące część średnioprężną z częścią niskoprężną. Dodatkowo poszczególne stopnie turbiny podzielono na podzespoły. I tak w części wysokoprężnej wyróżniono takie moduły, jak: kadłub, wirnik oraz węzeł łożyskowy podpory numer 1. W części średnioprężnej wydzielono następujące moduły: kadłub, wirnik, węzeł łożyskowy podpory numer 2 oraz węzeł łożyskowy podpory numer 3. Część niskoprężna składa się z następujących modułów: kadłub, wirnik, węzeł łożyskowy podpory numer 4 oraz węzeł łożyskowy podpory numer 5. Przedstawiona na rys. 28 struktura modelu turbiny przedstawia relacje podrzędności i nadrzędności poszczególnych modułów, z których model został opracowany.

### 5.2.2. Analiza reguł i stwierdzeń

Dla rozpatrywanego obiektu badań liczba reguł, które można zbudować na podstawie dostępnej wiedzy literaturowej i eksploatacyjnej jest bardzo duża. Do przetestowania opracowanego modelu wybrano kilkanaście reguł związanych z różnymi stanami, jakie mogą wystąpić w czasie działania turbiny. Wszystkie reguły zostały zaczerpnięte z opracowania [20], które jest katalogiem relacji diagnostycznych z zakresu eksploatacji turbin parowych o mocy 200 MW. Poniżej przedstawiono wybrane reguły, które zostały zaimplementowane w omawianym przykładzie. Reguły podzielono na grupy w zależności od treści konkluzji. Poniżej przedstawiono wybrane reguły rozpatrywane w opracowanym przykładzie.

#### Reguła 01

##### Jeżeli:

Nadmierny poziom drgań względnych wału

##### To:

Możliwe wystąpienie przycierania

## **Reguła 02**

### **Jeżeli:**

Wystąpił wzrost szumów w widmie drgań

### **To:**

Występuje przycieranie uszczelnień

## **Reguła 03**

### **Jeżeli:**

Wystąpiło wygięcie termiczne wału

### **To:**

Występuje przycieranie

## **Reguła 04**

### **Jeżeli:**

Wzrosła składowa podstawowa drgań względnych wału  
i występują podwyższone amplitudy składowych  
harmonicznych 0,5X, 1,5X, 2,5X

### **To:**

Występuje przycieranie wału

## **Reguła 05**

### **Jeżeli:**

W widmie drgań względnych występuje podwyższona składowa  
(0,35-0,49)X

### **To:**

Mogą występować drgania olejowe (wir olejowy) w łożysku

## **Reguła 06**

### **Jeżeli:**

Wartość sumarycznych drgań wału jest wysoka  
i wartość podharmonicznej 0,49X jest wysoka  
i wartość składowej 1X jest podwyższona

i występuje dodatkowa składowa  $1X + 0,49X$

i trajektoria ruchu czopa w łożysku dla częstotliwości subharmonicznej ma kształt spłaszczonej elipsy rozciągniętej w kierunku poziomym

**To:**

Występuje niestabilność olejowa w łożysku

### **Reguła 07**

**Jeżeli:**

Dominuje częstotliwość równa częstotliwości krytycznej i częstotliwość ta jest niezależna od prędkości obrotowej i występują dwa znaczniki fazowe w jednym cyklu trajektorii

**To:**

Występuje bicie olejowe

### **Reguła 08**

**Jeżeli:**

Występuje podwyższona składowa  $1X$

**To:**

Może występować niewywaga wirnika

### **Reguła 09**

**Jeżeli:**

Występuje podwyższona składowa  $2X$

**To:**

Wirnik może być niewyrównowazony

### **Reguła 10**

**Jeżeli:**

Kształt trajektorii środka czopa to koło lub elipsa i w widmie drgań w kierunku promieniowym dominuje tylko składowa o częstotliwości  $1X$

i liczba znaczników na trajektorii wynosi 1  
i dla określonej prędkości obrotowej wału znacznik nie zmienia swego położenia na trajektorii  
i kierunek obrotu trajektorii jest zgodny z kierunkiem obrotów wału

**To:**

Występuje niewyrównoważenie wirnika

### **Reguła 11**

**Jeżeli:**

Składowa 1X jest podwyższona

**To:**

Może występować rozosiowanie linii łożysk

### **Reguła 12**

**Jeżeli:**

Składowa 2X jest podwyższona

**To:**

Może występować rozosiowanie linii łożysk

### **Reguła 13**

**Jeżeli:**

Poziom drgań łożysk po obu stronach sprzęgła jest podwyższony

i różnica fazy drgań promieniowych na łożyskach po obu stronach sprzęgła wynosi  $0^\circ$

i poziom drgań poosiowych jest większy od poziomu drgań promieniowych

i różnica faz drgań poosiowych na łożyskach po obu stronach sprzęgła wynosi  $180^\circ$

**To:**

Występuje niewspółosiowość wałów (załamanie kątowe)

## Reguła 14

### Jeżeli:

Poziom drgań promieniowych łożysk po obu stronach sprzęgła jest podwyższony

i różnica faz drgań promieniowych na łożyskach po obu stronach sprzęgła wynosi  $180^\circ$

i poziom drgań osiowych nie jest podwyższony

### To:

Występuje niewspółosiowość wałów (przesunięcie promieniowe)

Przygotowano odpowiednie stwierdzenia, na podstawie których będzie możliwe zapisanie przedstawionych reguł w bazie MPNU. Przygotowanie stwierdzeń wymagało opracowania najpierw treści stwierdzeń i wpisania ich do odpowiedniej tablicy w bazie danych MPNU. Kolejnym krokiem było przygotowanie tablicy ogłoszeń i wypełnienie jej odpowiednimi stwierdzeniami, z zaznaczeniem modułów, które będą miały za zadanie aktualizowanie wartości tych stwierdzeń. W Tabeli 1 przedstawiono fragment zapytania do tablicy ogłoszeń. W zapytaniu uwzględniono wyłącznie stwierdzenia, których wartości są wyznaczone przez moduł WęzełŁozyskowy1. W Tabeli 1 przedstawiono treść stwierdzenia (w postaci trójki obiekt, atrybut, wartość) oraz nazwę modułu, który ma wyznaczyć wartość stwierdzenia. Podobnie zapisane zostały wartości stwierdzeń dla pozostałych modułów. Treść stwierdzenia, w którym w polu *Wartość*, tekst został poprzedzony znakiem #, oznacza stwierdzenie pomocnicze, którego wartość jest koniunkcją lub alternatywą innych stwierdzeń.

Tab. 1. Fragment treści tablicy TOgloszen

Obiekt	Atrybut	Wartość	Nazwa modułu
WirnikWP	Niewyrownowazenie	Wystepuje	WezelLozyskowy1
Lozysko1	Przycieranie	#Wystepuje	WezelLozyskowy1
Lozysko1	Przycieranie	#Może wystepowac	WezelLozyskowy1
Lozysko1	Wir olejowy	#Wystepuje	WezelLozyskowy1
Lozysko1	Wir olejowy	#Może wystepowac	WezelLozyskowy1
Lozysko1	Wir olejowy	#Może wystepowac	WezelLozyskowy1
Lozysko1	Bicie olejowe	#Wystepuje	WezelLozyskowy1
Lozysko1	Bicie olejowe	#Może wystepowac	WezelLozyskowy1
WirnikWP	Niewyrownowazenie	#Wystepuje	WezelLozyskowy1
Lozysko1	Poziom drgan względných	Obniżony	WezelLozyskowy1
Lozysko1	Poziom drgan względných	Podwyższony	WezelLozyskowy1
Lozysko1	Poziom drgan względných	Normalny	WezelLozyskowy1
Lozysko1	Poziom drgan względných	Normalny	WezelLozyskowy1
Lozysko1	Poziom szumow	Normalny	WezelLozyskowy1

Obiekt	Atrybut	Wartość	Nazwa modułu
Lozysko1	Poziom szumow	Podwyższony	WezelLozyskowy1
Lozysko1	Skladowa 0,5X drgan	Wystepuje	WezelLozyskowy1
Lozysko1	Skladowa 0,5X drgan	Nie wystepuje	WezelLozyskowy1
Lozysko1	Skladowa 0,5X drgan	Wzrosła	WezelLozyskowy1
Lozysko1	Skladowa 0,5X drgan	Zmalala	WezelLozyskowy1
Lozysko1	Skladowa 0,5X drgan	Normalna	WezelLozyskowy1
Lozysko1	Skladowa 0,5X drgan	Podwyższona	WezelLozyskowy1
Lozysko1	Skladowa 1X drgan	Wystepuje	WezelLozyskowy1
Lozysko1	Skladowa 1X drgan	Nie wystepuje	WezelLozyskowy1
Lozysko1	Skladowa 1X drgan	Wzrosła	WezelLozyskowy1
Lozysko1	Skladowa 1X drgan	Zmalala	WezelLozyskowy1
Lozysko1	Skladowa 1X drgan	Normalna	WezelLozyskowy1
Lozysko1	Skladowa 1X drgan	Podwyższona	WezelLozyskowy1
Lozysko1	Skladowa 1,5X drgan	Wystepuje	WezelLozyskowy1
Lozysko1	Skladowa 1,5X drgan	Nie wystepuje	WezelLozyskowy1
Lozysko1	Skladowa 1,5X drgan	Wzrosła	WezelLozyskowy1
Lozysko1	Skladowa 1,5X drgan	Zmalala	WezelLozyskowy1
Lozysko1	Skladowa 1,5X drgan	Normalna	WezelLozyskowy1
Lozysko1	Skladowa 1,5X drgan	Podwyższona	WezelLozyskowy1
Lozysko1	Skladowa 2X drgan	Wystepuje	WezelLozyskowy1
Lozysko1	Skladowa 2X drgan	Nie wystepuje	WezelLozyskowy1
Lozysko1	Skladowa 2X drgan	Wzrosła	WezelLozyskowy1
Lozysko1	Skladowa 2X drgan	Zmalala	WezelLozyskowy1
Lozysko1	Skladowa 2X drgan	Normalna	WezelLozyskowy1
Lozysko1	Skladowa 2X drgan	Podwyższona	WezelLozyskowy1
Lozysko1	Skladowa 2,5X drgan	Wystepuje	WezelLozyskowy1
Lozysko1	Skladowa 2,5X drgan	Nie wystepuje	WezelLozyskowy1
Lozysko1	Skladowa 2,5X drgan	Wzrosła	WezelLozyskowy1
Lozysko1	Skladowa 2,5X drgan	Zmalala	WezelLozyskowy1
Lozysko1	Skladowa 2,5X drgan	Normalna	WezelLozyskowy1
Lozysko1	Skladowa 2,5X drgan	Podwyższona	WezelLozyskowy1
Lozysko1	Skladowa 3X drgan	Wystepuje	WezelLozyskowy1
Lozysko1	Skladowa 3X drgan	Nie wystepuje	WezelLozyskowy1
Lozysko1	Skladowa 3X drgan	Wzrosła	WezelLozyskowy1
Lozysko1	Skladowa 3X drgan	Zmalala	WezelLozyskowy1
Lozysko1	Skladowa 3X drgan	Normalna	WezelLozyskowy1
Lozysko1	Skladowa 3X drgan	Podwyższona	WezelLozyskowy1
Lozysko1	Skladowa 4X drgan	Wystepuje	WezelLozyskowy1
Lozysko1	Skladowa 4X drgan	Nie wystepuje	WezelLozyskowy1
Lozysko1	Skladowa 4X drgan	Wzrosła	WezelLozyskowy1
Lozysko1	Skladowa 4X drgan	Zmalala	WezelLozyskowy1
Lozysko1	Skladowa 4X drgan	Normalna	WezelLozyskowy1
Lozysko1	Skladowa 4X drgan	Podwyższona	WezelLozyskowy1
Lozysko1	Skladowa 0,5X drgan	Wystepuje	WezelLozyskowy1
Lozysko1	Skladowa 0,5X drgan	Nie wystepuje	WezelLozyskowy1
Lozysko1	Skladowa 0,5X drgan	Wzrosła	WezelLozyskowy1
Lozysko1	Skladowa 0,5X drgan	Zmalala	WezelLozyskowy1
Lozysko1	Skladowa 0,5X drgan	Normalna	WezelLozyskowy1
Lozysko1	Skladowa 0,5X drgan	Podwyższona	WezelLozyskowy1
Lozysko1	Skladowa 1X drgan	Wystepuje	WezelLozyskowy1

Obiekt	Atrybut	Wartość	Nazwa modułu
Lozysko1	Skladowa 1X drgan	Nie wystepuje	WezelLozyskowy1
Lozysko1	Skladowa 1X drgan	Wzrosła	WezelLozyskowy1
Lozysko1	Skladowa 1X drgan	Zmalala	WezelLozyskowy1
Lozysko1	Skladowa 1X drgan	Normalna	WezelLozyskowy1
Lozysko1	Skladowa 1X drgan	Podwyższona	WezelLozyskowy1
Lozysko1	Skladowa 1,5X drgan	Wystepuje	WezelLozyskowy1
Lozysko1	Skladowa 1,5X drgan	Nie wystepuje	WezelLozyskowy1
Lozysko1	Skladowa 1,5X drgan	Wzrosła	WezelLozyskowy1
Lozysko1	Skladowa 1,5X drgan	Zmalala	WezelLozyskowy1
Lozysko1	Skladowa 1,5X drgan	Normalna	WezelLozyskowy1
Lozysko1	Skladowa 1,5X drgan	Podwyższona	WezelLozyskowy1
Lozysko1	Skladowa 2X drgan	Wystepuje	WezelLozyskowy1
Lozysko1	Skladowa 2X drgan	Nie wystepuje	WezelLozyskowy1
Lozysko1	Skladowa 2X drgan	Wzrosła	WezelLozyskowy1
Lozysko1	Skladowa 2X drgan	Zmalala	WezelLozyskowy1
Lozysko1	Skladowa 2X drgan	Normalna	WezelLozyskowy1
Lozysko1	Skladowa 2X drgan	Podwyższona	WezelLozyskowy1
Lozysko1	Skladowa 2,5X drgan	Wystepuje	WezelLozyskowy1
Lozysko1	Skladowa 2,5X drgan	Nie wystepuje	WezelLozyskowy1
Lozysko1	Skladowa 2,5X drgan	Wzrosła	WezelLozyskowy1
Lozysko1	Skladowa 2,5X drgan	Zmalala	WezelLozyskowy1
Lozysko1	Skladowa 2,5X drgan	Normalna	WezelLozyskowy1
Lozysko1	Skladowa 2,5X drgan	Podwyższona	WezelLozyskowy1
Lozysko1	Skladowa 3X drgan	Wystepuje	WezelLozyskowy1
Lozysko1	Skladowa 3X drgan	Nie wystepuje	WezelLozyskowy1
Lozysko1	Skladowa 3X drgan	Wzrosła	WezelLozyskowy1
Lozysko1	Skladowa 3X drgan	Zmalala	WezelLozyskowy1
Lozysko1	Skladowa 3X drgan	Normalna	WezelLozyskowy1
Lozysko1	Skladowa 3X drgan	Podwyższona	WezelLozyskowy1
Lozysko1	Skladowa 4X drgan	Wystepuje	WezelLozyskowy1
Lozysko1	Skladowa 4X drgan	Nie wystepuje	WezelLozyskowy1
Lozysko1	Skladowa 4X drgan	Wzrosła	WezelLozyskowy1
Lozysko1	Skladowa 4X drgan	Zmalala	WezelLozyskowy1
Lozysko1	Skladowa 4X drgan	Normalna	WezelLozyskowy1
Lozysko1	Skladowa 4X drgan	Podwyższona	WezelLozyskowy1
Lozysko1	Drgania wzgledne	Normalne	WezelLozyskowy1
Lozysko1	Drgania wzgledne	Wysokie	WezelLozyskowy1
Lozysko1	Drgania bezwzgledne	Normalne	WezelLozyskowy1
Lozysko1	Drgania bezwzgledne	Wysokie	WezelLozyskowy1
Lozysko1	Kształt trajektorii	Elipsa	WezelLozyskowy1
Lozysko1	Kształt trajektorii	Splaszczona elipsa	WezelLozyskowy1
Lozysko1	Kierunek rozciagniecia elipsy	Poziomy	WezelLozyskowy1
Lozysko1	Kierunek rozciagniecia elipsy	Pionowy	WezelLozyskowy1
Lozysko1	Kształt trajektorii	Kolo	WezelLozyskowy1
Lozysko1	Czestotliowosc rowna	Dominuje	WezelLozyskowy1
Lozysko1	Czestotliowosc rowna	Zalezy	WezelLozyskowy1
Lozysko1	Czestotliowosc rowna	Nie zalezy	WezelLozyskowy1
Lozysko1	Znacznik fazowy	Jeden	WezelLozyskowy1
Lozysko1	Znacznik fazowy	Dwa	WezelLozyskowy1
Lozysko1	Znacznik fazowy	Wiele	WezelLozyskowy1



Obiekt	Atrybut	Wartość	Nazwa modułu
Lozysko1	Polozenie znacznika	Nie zmienia się	WezelLozyskowy1
Lozysko1	Polozenie znacznika	Zmienia się	WezelLozyskowy1
Lozysko1	Poziom drgan promieniowych	Podwyszony	WezelLozyskowy1
Lozysko1	Poziom drgan promieniowych	Normalny	WezelLozyskowy1
Lozysko1	Poziom drgan poosiowych	Podwyszony	WezelLozyskowy1
Lozysko1	Poziom drgan poosiowych	Normalny	WezelLozyskowy1
Lozysko1	Przycieranie	Występuje	WezelLozyskowy1
Lozysko1	Przycieranie	Może występować	WezelLozyskowy1
Lozysko1	Wir olejowy	Występuje	WezelLozyskowy1
Lozysko1	Wir olejowy	Może występować	WezelLozyskowy1
Lozysko1	Bicie olejowe	Występuje	WezelLozyskowy1
Lozysko1	Bicie olejowe	Może występować	WezelLozyskowy1

Podane w tabeli stwierdzenia są jedynie przykładami i nie wyczerpują wszystkich stwierdzeń, które można zdefiniować w stosunku do rozpatrywanego obiektu badań.

### 5.2.3. Zapis reguł

Po przeprowadzeniu analizy reguł i zdefiniowaniu wymaganego zbioru stwierdzeń kolejnym etapem było zapisanie reguł w odpowiedniej postaci w bazie danych MPNU. Przedstawione w poprzednim rozdziale reguły zostały zapisane w tablicy Reguły. Zapis reguł dla potrzeb modułów różni się nieco od innych metod zapisu reguł. Każda reguła zapisywana jest w postaci reguły prostej. Reguły złożone zamieniane są na szereg reguł prostych. Operatory WK oraz WD, stosowane w zapisie reguł prostych, odpowiadają odpowiednio: warunkom koniecznym oraz warunkom dostatecznym. Sposób zapisywania reguł podanych wcześniej, przedstawiony zostanie na przykładach.

#### Przykład 1

Dana jest następująca reguła:

Jeżeli:

Nadmierny poziom drgań względnych wału

To:

Możliwe wystąpienie przycierania

W celu zapisania takiej reguły w opracowanym modelu konieczne jest przygotowanie dwóch stwierdzeń odpowiadających przesłance i konkluzji w powyższej regule:

1) Stwierdzenie  $x$  o treści zapisanej zgodnie z (2):

<"Wał części wysokoprężnej", "Poziom drgań względnych wału", "Nadmierny">

i wartości zapisanej w postaci pary liczb określających stopień konieczności i możliwości, np.: [0,8; 1]

2) Stwierdzenie  $y$  o treści zapisanej zgodnie z (2):

<"Wał części wysokoprężnej", "Przycieranie", "Możliwe">

i wartości początkowej (przed uwzględnieniem przesłanek) zapisanej w postaci pary liczb określających stopień konieczności i możliwości, np.: [0; 1]

Dla tak przygotowanych stwierdzeń można zapisać regułę, przyjmując, że stwierdzenie  $x$  jest warunkiem dostatecznym dla uznania stwierdzenia  $y$ , w postaci:

$$b(y) \geq b(x) - \delta \quad (45)$$

Zgodnie z (40) wartość stwierdzenia  $y$  (po uwzględnieniu przesłanki) będzie w takim przypadku równa (dla  $\delta=0,02$ ):

$$b(y) = [0,78; 1]$$

O takim stwierdzeniu można powiedzieć, że stopień prawdziwości tego stwierdzenia jest dosyć duży. A więc stwierdzenie takie jest w wysokim stopniu prawdziwe.

W przypadku reguł prostych zapis takiej reguły w bazie jest łatwy. Trochę bardziej skomplikowany jest zapis reguł zawierających złożone przesłanki.

## Przykład 2

W tym przykładzie przedstawiony zostanie sposób zapisu reguły zawierającej złożoną przesłankę będącą koniunkcją kilku stwierdzeń.

Niech będzie dana reguła:

### Reguła 14

#### Jeżeli:

Poziom drgań promieniowych łożysk po obu stronach sprzęgła jest podwyższony

i

różnica faz drgań promieniowych na łożyskach po obu stronach sprzęgła wynosi  $180^\circ$

i

poziom drgań osiowych nie jest podwyższony

**To:**

Występuje niewspółosiowość wałów (przesunięcie promieniowe)

Rozpatrywany przypadek będzie dotyczył sprzęgła łączącego wał części średnioprężnej i niskoprężnej turbiny, a znajdującego się pomiędzy łożyskami nr 3 i 4. Tak jak poprzednio, w pierwszym kroku zostaną określone stwierdzenia, na podstawie których będzie można zapisać regułę:

1) Stwierdzenie  $x_1$  o treści zapisanej zgodnie z (2):

<"Łożysko Nr 3", "Poziom drgań promieniowych", "Podwyższony">

i wartości zapisanej w postaci pary liczb określających stopień konieczności i możliwości ( $[x_1-1; x_1-2]$ ), np.: [0,8; 1]

2) Stwierdzenie  $x_2$  o treści zapisanej zgodnie z (2):

<"Łożysko Nr 4", "Poziom drgań promieniowych", "Podwyższony">

i wartości zapisanej w postaci pary liczb określających stopień konieczności i możliwości ( $[x_2-1; x_2-2]$ ), np.: [0,75; 1]

3) Stwierdzenie  $x_3$  o treści zapisanej zgodnie z (2):

<"Sprzęgło SP/NP", "Różnica faz", "Równa 180°">

i wartości zapisanej w postaci pary liczb określających stopień konieczności i możliwości ( $[x_3-1; x_3-2]$ ), np.: [0,75; 1]

4) Stwierdzenie  $x_4$  o treści zapisanej zgodnie z (2):

<"Łożysko Nr 3", "Poziom drgań poosiowych", "Normalny">

i wartości zapisanej w postaci pary liczb określających stopień konieczności i możliwości ( $[x_4-1; x_4-2]$ ), np.: [1; 1]

5) Stwierdzenie  $x_5$  o treści zapisanej zgodnie z (2):

<"Łożysko Nr 4", "Poziom drgań poosiowych", "Normalny">

i wartości zapisanej w postaci pary liczb określających stopień konieczności i możliwości ( $[x_5-1; x_5-2]$ ), np.: [1; 1]

W celu zapisania rozpatrywanej reguły w bazie MPNU konieczne jest zastąpienie koniunkcji stwierdzeń równoważnym im stwierdzeniem pomocniczym:

$$y_1 = x_1 \text{ AND } x_2 \text{ AND } x_3 \text{ AND } x_4 \text{ AND } x_5 \quad (46)$$

o treści zapisanej zgodnie z (2):

<"Sprzęgło SP/NP", "Niewspółosiowość wałów", "Występuje">

Konkluzja  $y$  posiada treść zapisaną zgodnie z (2):

<"Sprzęgło SP/NP", "Niewspółosiowość wałów", "Występuje">

i wartość zapisaną w postaci pary liczb określających stopień konieczności i możliwości, o wartości początkowej [0; 1]

Do wyznaczenia stopnia prawdziwości stwierdzenia  $y$ , stosowane jest stwierdzenie pomocnicze  $y_1$ , będące warunkiem wystarczającym dla stwierdzenia  $y$ . Wartość stwierdzenia  $y_1$  (46) wyznaczana jest zgodnie z (43).

$$b(y_1) = [\min(x_1 - 1, x_2 - 1, x_3 - 1, x_4 - 1, x_5 - 1); \min(x_1 - 2, x_2 - 2, x_3 - 2, x_4 - 2, x_5 - 2)] \quad (47)$$

Do wyznaczania wartości konkluzji  $y$  stosowana jest zależność (40) reprezentująca warunek dostateczny. W bazie MPNU zapisywane są wszystkie podane nierówności poprzez skojarzenie par stwierdzeń i określenie, czy ma być sprawdzany warunek konieczny, dostateczny, czy też ma być wyznaczona koniunkcja lub alternatywa stwierdzeń.

Dla podanych przykładowych wartości stwierdzeń, wartość stwierdzenia pomocniczego  $y_1$  będzie równa:

$$b(y_1) = [0,75; 1]$$

natomiast wartość stwierdzenia  $y$  (dla  $\delta = 0,02$ ) będzie równa:

$$b(y) = [0,73; 1]$$

Zaletą takiego toku postępowania w celu wyznaczenia wartości stwierdzenia  $y$  jest to, że do jej wyznaczenia nie jest konieczna znajomość wszystkich wartości stwierdzeń  $x_1-x_5$ . Pozwala to na przeprowadzenie procesu wnioskowania nawet przy niekompletnych danych. Na przykład, jeżeli nie będzie znana wartość stwierdzenia  $x_3$ , tzn.  $b(x_3)=[0; 1]$ , to otrzymamy wartość stwierdzenia pomocniczego  $y_1=[0; 1]$  oraz wartość konkluzji  $y=[0; 1]$ .

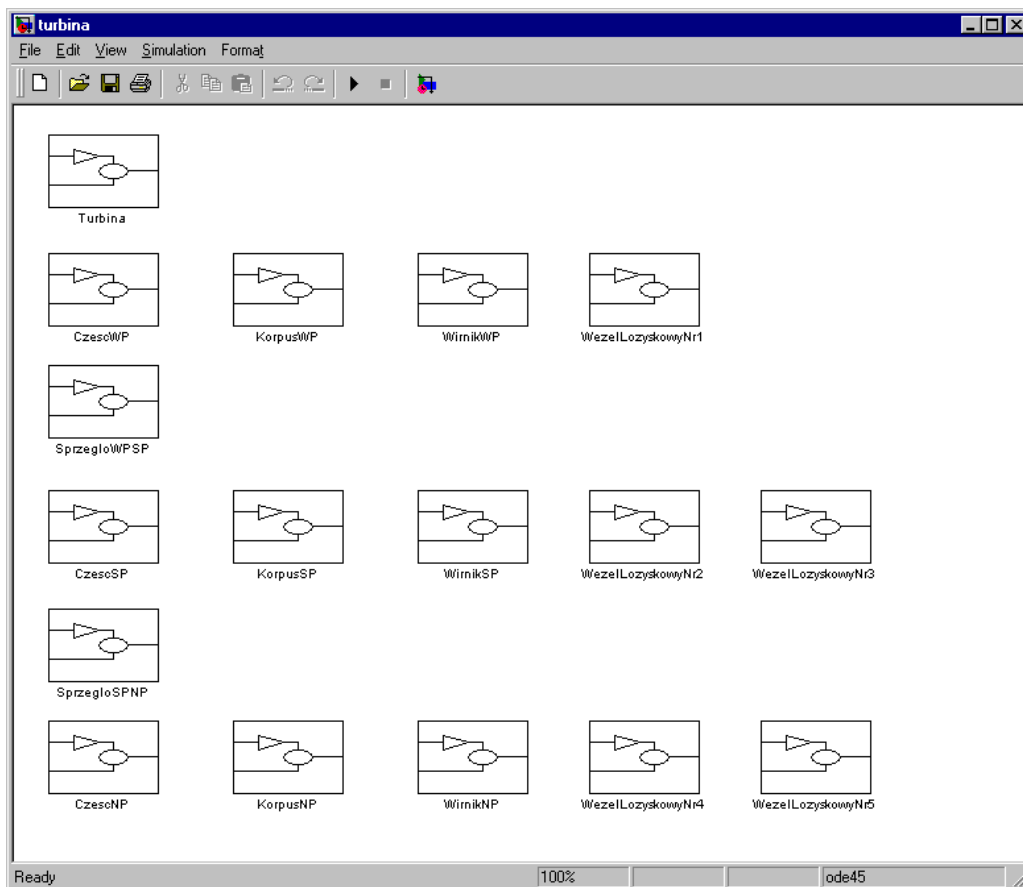
## 5.3. Model układu

### 5.3.1. Główne elementy modelu

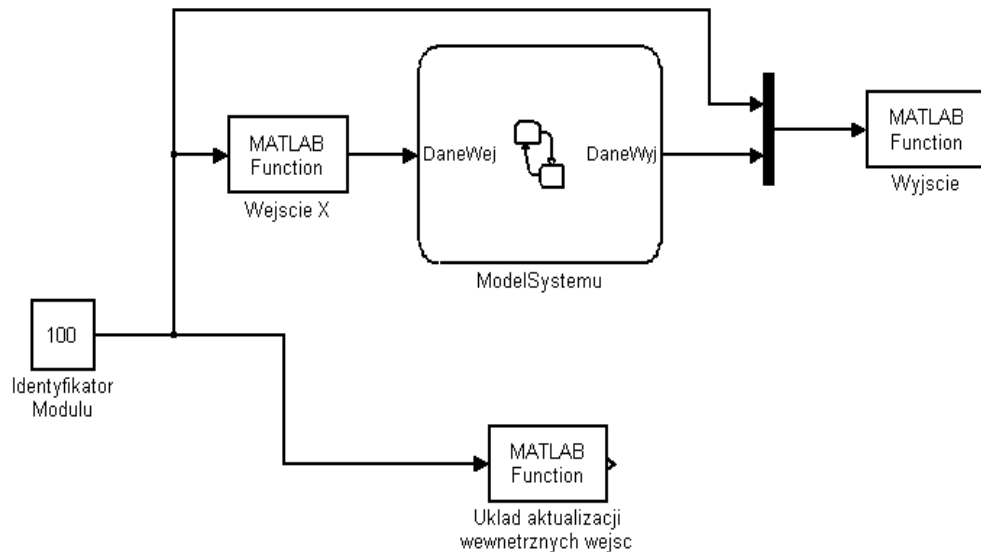
W rozdziale opisano działania związane z opracowaniem modelu złożonego obiektu, jakim jest turbina o mocy 200 MW. Pierwszym krokiem jest przygotowanie grupy modułów reprezentujących poszczególne elementy turbiny, które zostały wybrane w wyniku analizy struktury układu.

Opracowano grupę następujących modułów:

- Turbina,
- CzescWP,
- SprzegloWPSP,
- CzescSP,
- SprzegloSPNP,
- CzescNP,
- KorpusWP,
- KorpusSP,
- KorpusNP,
- WirnikWP,
- WirnikSP,
- WirnikNP,
- WezeLozyskowyNr1,



Rys. 29. Grupa modułów w modelu turbiny



Rys. 30. Moduł Turbina zapisany w systemie Simulink

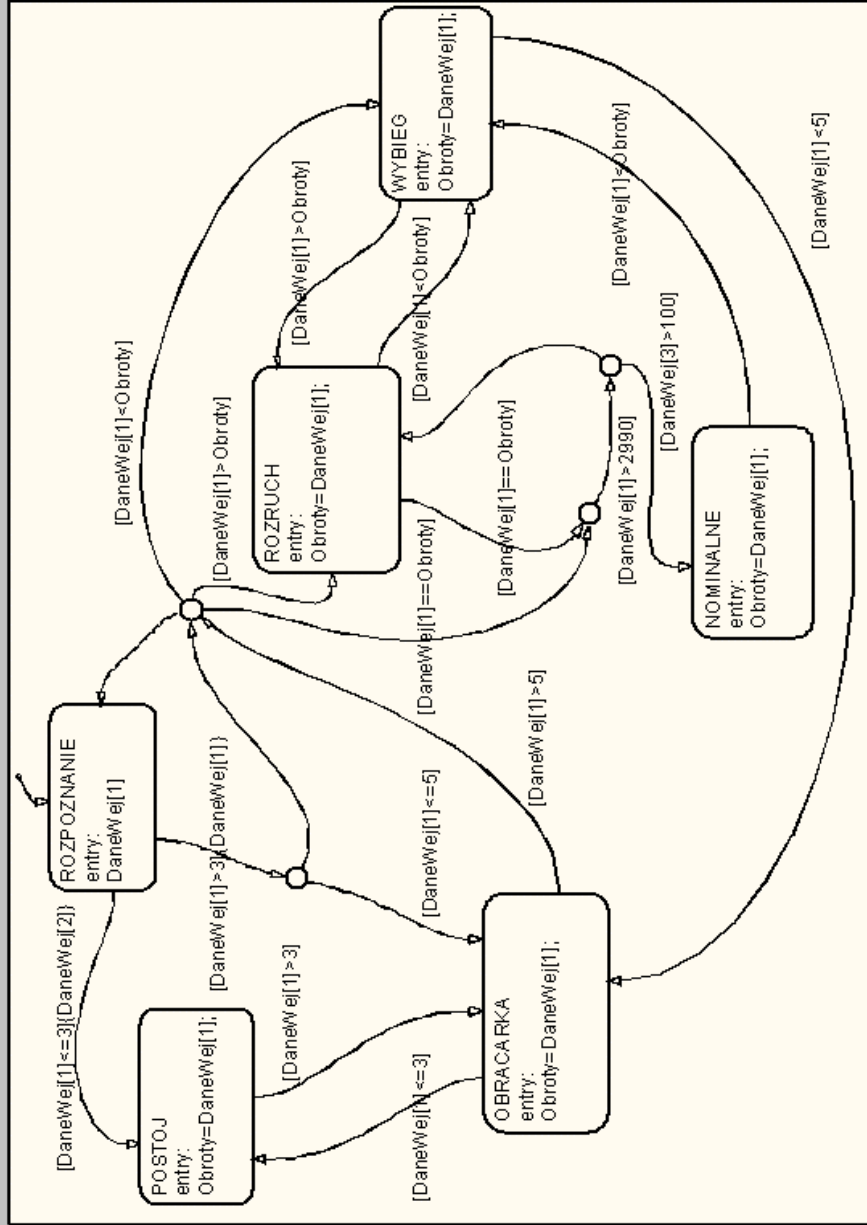
- WezelLozyskowyNr2,
- WezelLozyskowyNr3,
- WezelLozyskowyNr4,
- WezelLozyskowyNr5.

Na rys.29 przedstawiono całą, utworzoną, grupę modułów. Dla uproszczenia, prezentowana grupa znajduje się w jednym oknie i jest uruchamiana jako całość. Każdy moduł ma również odpowiednio przygotowane wnętrze, pozwalające na wykonywanie swoich zadań. Poniżej przedstawiono kilka przykładów.

### Moduł – Turbina

Pierwszym z rozpatrywanych przykładów jest moduł Turbina. Główne elementy tego modułu przedstawiono na rys. 30. Przedstawiona realizacja modułu zawiera kilka elementów. Wejście X to fragment odpowiedzialny za pobieranie z bazy danych wartości aktualnych wielkości fizycznych mierzonych na rzeczywistym układzie. Wyjście Y umożliwia zapisywanie na tablicy ogłoszeń zmodyfikowanych wartości stwierdzeń. Wartości te są modyfikowane w bloku ModelSystemu. Wejściem do tego bloku są rzeczywiste dane uzyskane z pomiarów. Blok ModelSystemu to odpowiednio przygotowany blok programu Stateflow. Postać tego bloku i realizowane działania przedstawiono na rys. 31. W module Turbina za pomocą bloku Stateflow przygotowano procedurę rozpoznawania stanu pracy turbiny. Informacja o stanie pracy jest następnie wykorzystywana w pozostałych modułach.

Turbina/System



Rys. 31. Podukład reprezentujący model systemu w module Turbina

## Moduł – WezelLozyskowyNr4

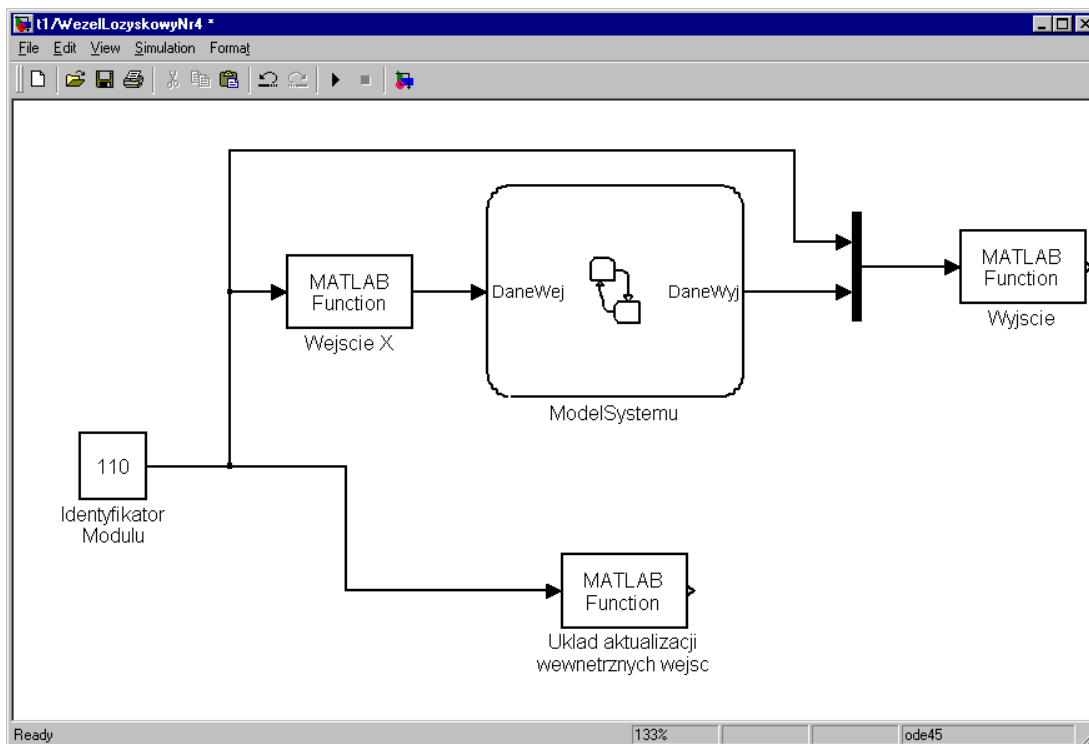
Kolejnym prezentowanym modułem jest moduł modelujący węzeł łożyskowy nr 4 turbiny. Moduł ten wyznacza wartości stwierdzeń dotyczących łożyska nr 4 turbiny, zgodnie z regułami dotyczącymi obiektów klasy Łożysko. Ponieważ podstawowe elementy modułu są takie same we wszystkich modułach, główna część modułu wygląda tak samo jak w module Turbina. Na rys. 32 przedstawiono postać modułu WezelLozyskowyNr4 zaimplementowany w systemie Simulink. W przedstawionej implementacji można wyróżnić kilka podukładów. Są to:

WejscieX - odczytanie danych z rzeczywistego obiektu,

WyjscieY - zapisanie na tablicy ogłoszeń wyznaczonych wartości stwierdzeń,

Model Systemu - implementacja układu równań (19) w postaci modułu Stateflow,

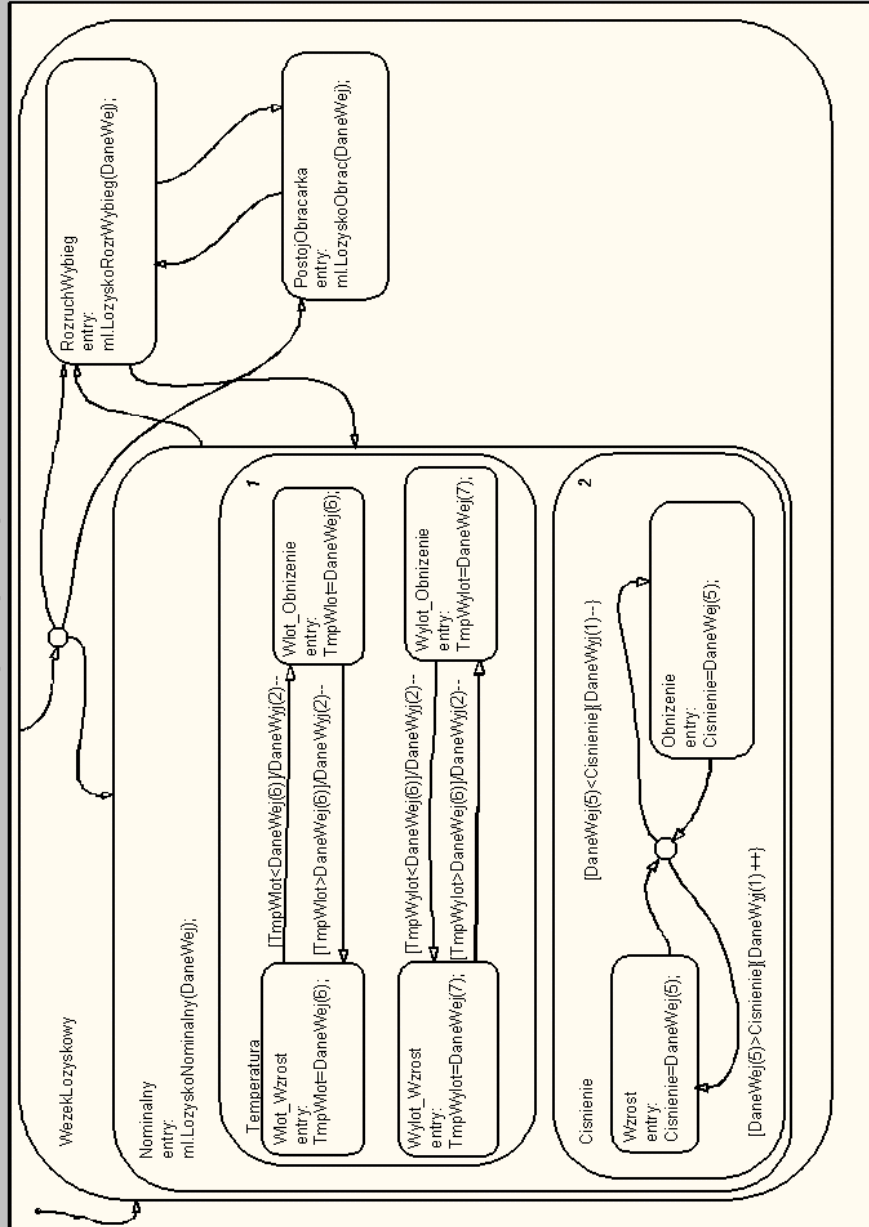
Układ aktualizacji wewnętrznych wejść modułu - podukład wyznaczający wartości stwierdzeń na podstawie zdefiniowanych reguł, przeznaczonych dla tego modułu.



Rys. 32. Moduł WezelLozyskowyNr4 zapisany w systemie Simulink



## Wezeł LozyskowyNr4/System

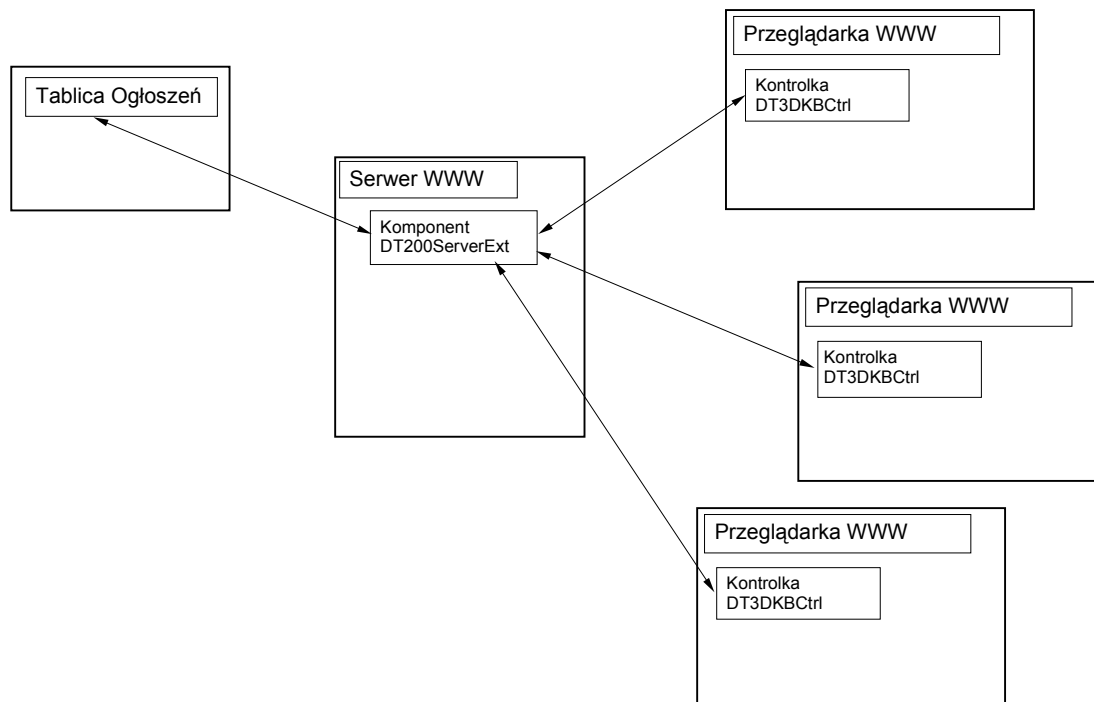


Rys. 33. Podukład reprezentujący model systemu w module WezełLozyskowyNr4

Moduł `WezełLozyskowyNr4` pobiera z rzeczywistego obiektu takie dane jak: wartości amplitud drgań poszczególnych składowych, wartości faz, stan pracy turbozespołu (wyznaczony przez moduł `Turbina`) oraz dane dotyczące parametrów pracy łożyska jak ciśnienie i temperatura oleju na wlocie i wylocie z łożyska. Dane te są następnie przekazywane do bloku `ModelSystemu`, w którym zamieniane są na odpowiednie wartości stwierżeń, zapisywanych na tablicy ogłoszeń. Na rys. 33 przedstawiono postać bloku `ModelSystemu` zdefiniowanego w tym module. Blok `ModelSystemu` jest zapisywany w postaci schematu Stateflow. Przedstawiony blok wyznacza zadane wartości stwierżeń w zależności od stanu pracy turbiny. Umożliwia on również określenie wartości parametrów dotyczących temperatury oleju (na wlocie i na wylocie z łożyska) oraz ciśnienia. Układ rozpoznaje, czy temperatura oleju i ciśnienia wzrosła, czy zmalała.

### 5.3.2. Prezentacja wyników

Wynikiem działania modelu są wartości stwierżeń zapisanych na tablicy ogłoszeń. Do prezentacji wyników działania modelu przygotowano oprogramowanie przygotowane do działania w sieci Internet/Intranet, wykorzystywane również w systemie DT200 (rys. 34). Opracowane oprogramowanie składa się z dwóch komponentów.



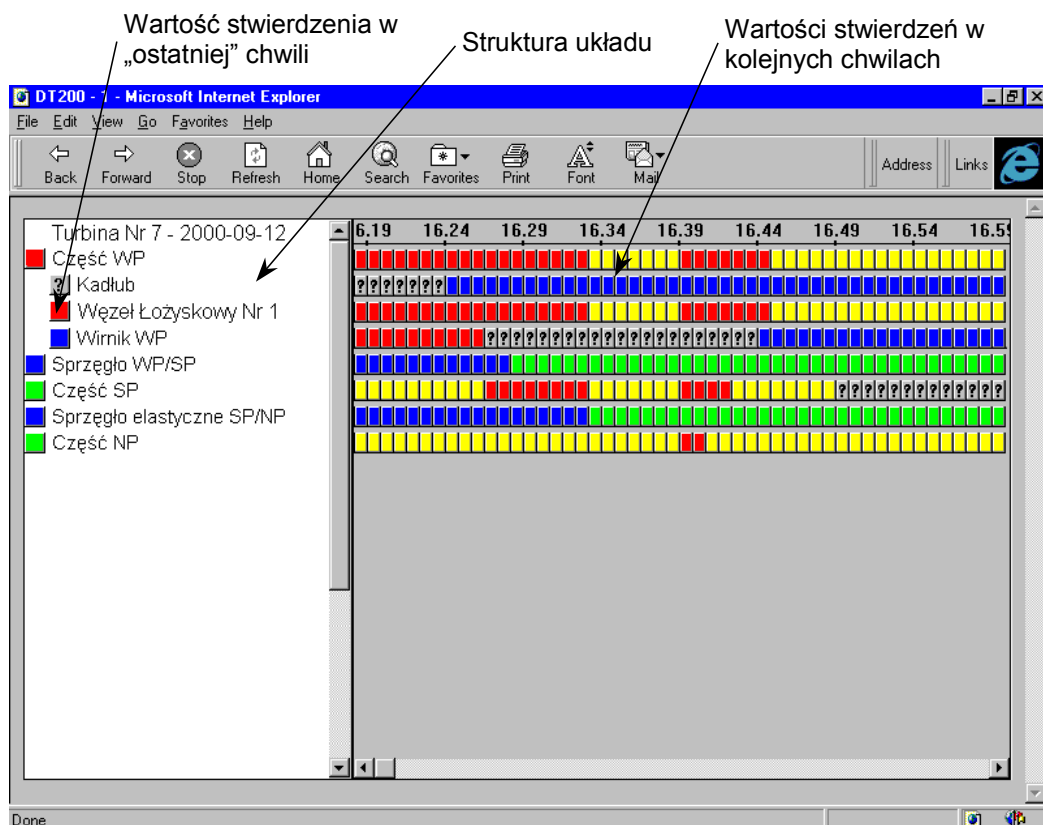
Rys. 34. Schemat dostępu do tablicy ogłoszeń przez przeglądarkę internetową

Pierwszy z komponentów *DT200ServerExt*, umożliwia rozszerzenie możliwości serwera WWW o dostęp do wartości stwierdzeń zapisanych na tablicy ogłoszeń. Umożliwia on także określenie struktury modelowanego układu oraz prezentowanych stwierdzeń. Rozszerzenie możliwości serwera WWW polegało na opracowaniu grupy funkcji umożliwiających sięgnięcie do:

- struktury modelowanego układu zapisanego w tablicy **TUkład**,
- wartości stwierdzeń z tablicy ogłoszeń.

Oprogramowanie zostało przygotowane dla serwera WWW MS Internet Information Server w postaci odpowiedniej biblioteki DLL zgodnie z wymaganiami ISAPI.

Drugi z komponentów przygotowanego oprogramowania to kontrolka ActiveX *DT3DKBCtrl*, umieszczona na przygotowanej stronie WWW. Kontrolka ta umożliwia prezentację wartości stwierdzeń umieszczonych na tablicy ogłoszeń (rys. 35). Kontrolka umieszczona jest na odpowiednio przygotowanej stronie WWW, do której można sięgać z przeglądarki internetowej działającej w środowisku MS Windows i obsługującej komponenty ActiveX. Okno główne kontrolki składa się z dwóch części. Lewa część okna przedstawia strukturę rozpatrywanego układu, natomiast w prawej części okna przedstawiono kolorowe wartości stwierdzeń w kolejnych chwilach czasu. Kontrolka *DT3DKBCtrl* jest elementem



Rys. 35. Okno przeglądarki z widokiem tablicy ogłoszeń

interaktywnym, tzn. użytkownik może za pomocą myszy określać stopień szczegółowości, z jakim chce oglądać rozpatrywany układ. Należy w lewym okienku kontrolki kliknąć w ikonkę określającą wartość stwierdzenia w „bieżącej” chwili. W ten sposób można ustalać wartości stwierdzeń, modułów i ich poziomów szczegółowości na jakim mają być prezentowane.

### 5.3.3. Elementy pomocnicze

Dla zastosowania w praktyce omawianych modeli niezbędne było odpowiednie oprogramowanie pomocnicze, umożliwiające między innymi dostęp do cech sygnałów rzeczywistego obiektu. Jednym z elementów pomocniczych jest aplikacja użytkownika systemu DT200 [26][76]. Okno główne aplikacji przedstawiono na rys. 36. Aplikacja DT3A101 [23] jest aplikacją typu MDI (Multiple Document Interface), działającą w środowisku Windows. Umożliwia ona jednoczesne otwarcie w osobnych okienkach wielu dokumentów - opcji analizy danych. Aplikacja umożliwia przeglądanie zarówno danych bieżących, jak i danych historycznych. Aplikacja DT3A101 została opracowana przy współudziale zespołu z Instytutu Maszyn Ciepłych Politechniki Częstochowskiej. Zespół ten był jednym ze współwykonawców projektu badawczego PBZ-038-06, mającego na celu opracowanie systemu diagnostycznego dla turbozespołów energetycznych – DT200-1. Okna związane z prezentacją danych (okna poszczególnych dokumentów) zostały opracowane przez pracowników Instytutu Maszyn Ciepłych Politechniki Częstochowskiej. Natomiast fragmenty oprogramowania związane z głównym oknem aplikacji i komunikacji z bazą danych zostały opracowane przez autora niniejszej pracy.

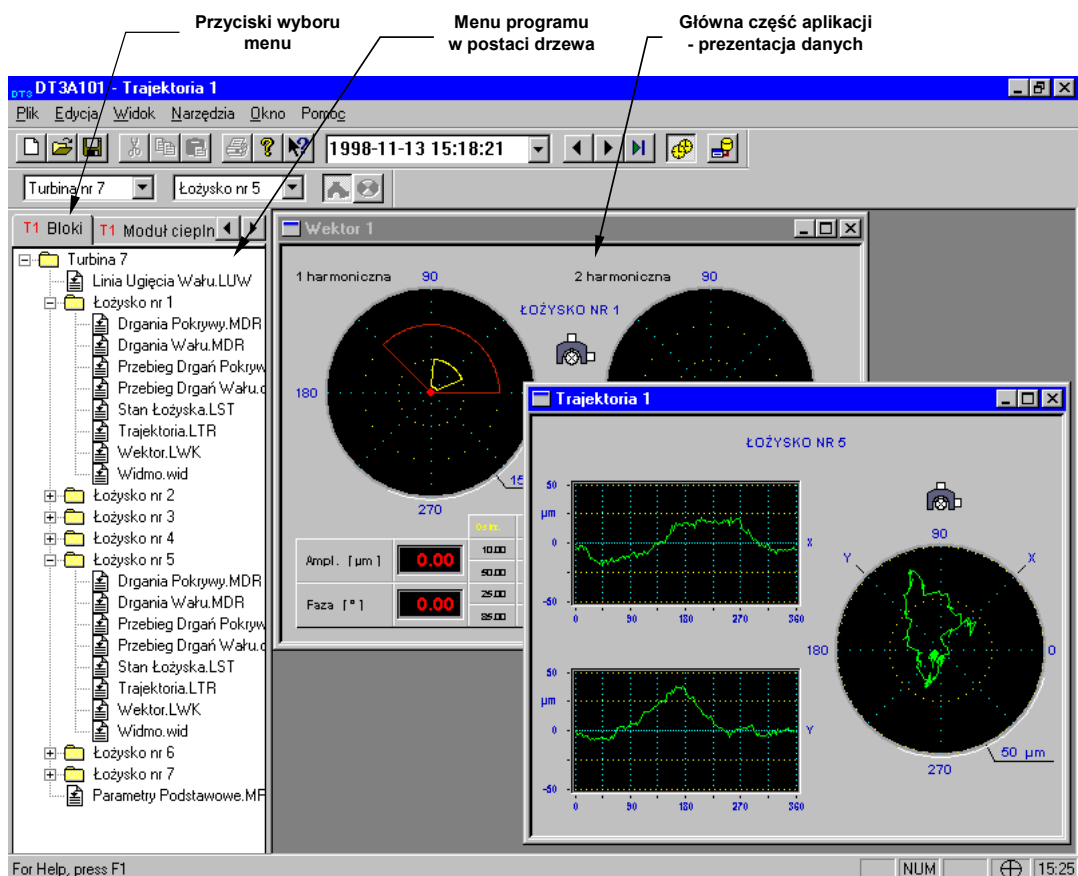
Dla aplikacji DT3A101 opracowano szereg bibliotek klas umożliwiających opracowanie tej aplikacji. Do podstawowych bibliotek należą następujące:

- DT3DMainApp - główna biblioteka aplikacji użytkownika systemu DT200 [72],
- DT3DDB - biblioteka komunikacji z bazą danych systemu DT200 [71],
- DT3G - biblioteka nadzoru drganiowego,
- DT3F - biblioteka nadzoru ciepno-wytężeniowego.

Struktura klas biblioteki DT3DMainApp.dll opiera się na bibliotece Microsoft Foundation Class (MFC) [58]. Strukturę klas biblioteki DT3DMainApp przedstawiono na rys. 37. Na rysunku tym zaznaczono utworzone klasy, dziedziczące z odpowiednich klas biblioteki MFC. Poniżej przedstawiono krótką charakterystykę wybranych nowo utworzonych klas:

- CDT3App - główna klasa aplikacji, odpowiedzialna za zarządzanie wszystkimi oknami,
- CdtMainFrame - klasa odpowiedzialna za utworzenie szkieletu okna głównego aplikacji,
- CdtChildFrame - klasa odpowiedzialna za utworzenie szkieletu okna dokumentu; pozwala na asynchroniczne pobieranie danych z bazy,
- CTreeWnd - klasa przeznaczona do obsługi okienka z drzewkiem (menu programu w postaci drzewa rys. 36); w klasie tej znajdują się informacje dotyczące poszczególnych węzłów w drzewku, struktury drzewka jak również postaci wyświetlanego drzewka.

Druga z opracowanych bibliotek umożliwia dostęp do bazy danych systemu DT200-1. Opracowana biblioteka DT3DDB stanowi integralną część opracowywanego modułu 'interface' użytkownika DT3A101. Biblioteka ta spełnia następujące wymagania:



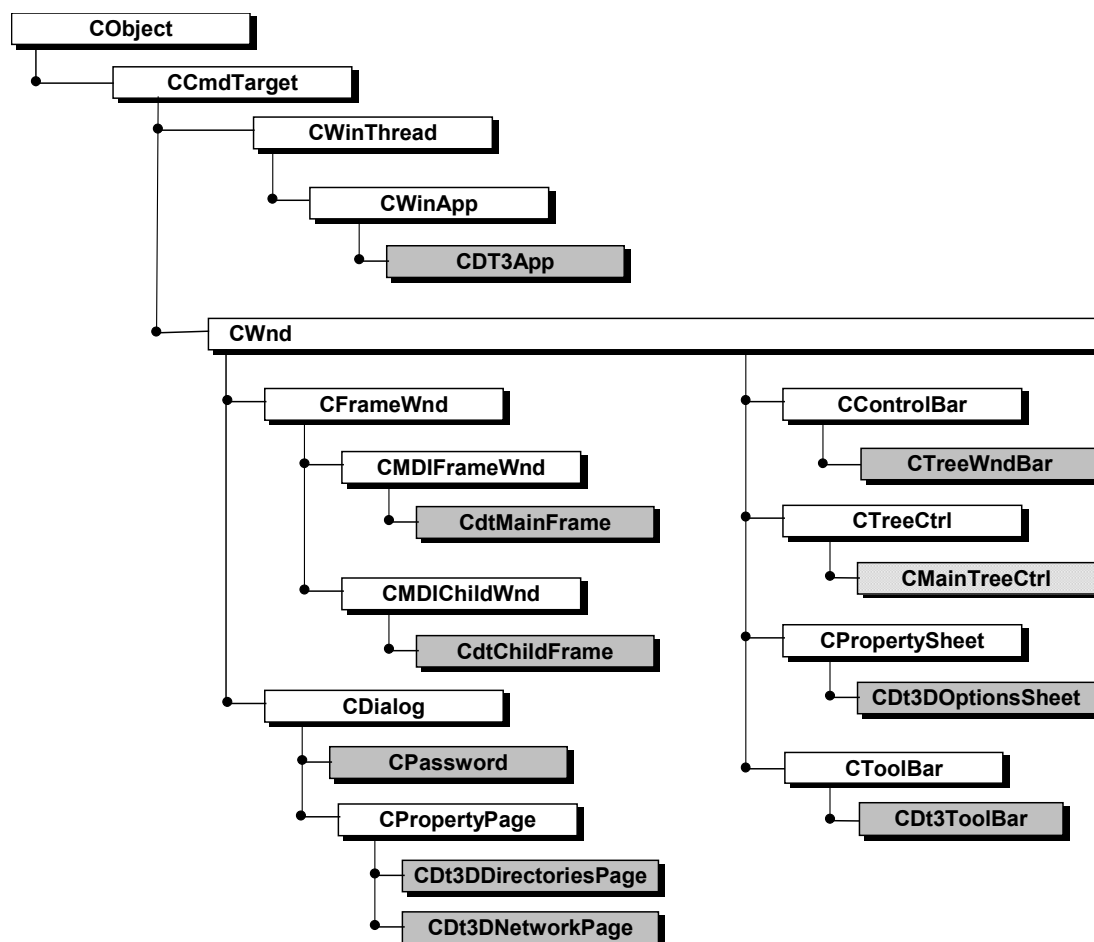
Rys. 36. Okno główne aplikacji użytkownika DT3A101 w systemie DT200 [23] (okna „Wektor” i „Trajektoria” zostały opracowane przez pracowników IMC Politechniki Częstochowskiej we współpracy z KPKM Politechniki Śląskiej)

- umożliwia pobieranie danych zapisanych w bazie danych systemu DT200, zgodnie ze specyfikacją tej bazy,
- dostęp do części tablic umożliwia nie tylko odczyt danych, ale również ich modyfikację i dodawanie nowych danych,
- dostęp do danych zapisanych w bazie jest realizowany poprzez sterowniki ODBC i poprzez bibliotekę DB-Library.

Biblioteka DT3DDB, umożliwia pobieranie i wstawianie danych do tablic zdefiniowanych w bazie danych systemu DT200 zarówno poprzez połączenie ODBC, jak również poprzez funkcje zdefiniowane w bibliotece DB-Library dla Microsoft SQL Server 6.5. Biblioteka DT3DDB składa się z dwóch głównych części. Pierwsza część odpowiada za połączenie z bazą danych poprzez łącza ODBC, druga pozwala na bezpośrednią komunikację z bazą danych systemu DT200, za pośrednictwem biblioteki DB-Library. Ogólna struktura biblioteki klas została przedstawiona na rys. 38. Poniżej krótko opisano wybrane klasy biblioteki DT3DDB.

### Klasa CDBExt

Klasa **CDBExt** została zdefiniowana dla potrzeb biblioteki **DT3DDB**.

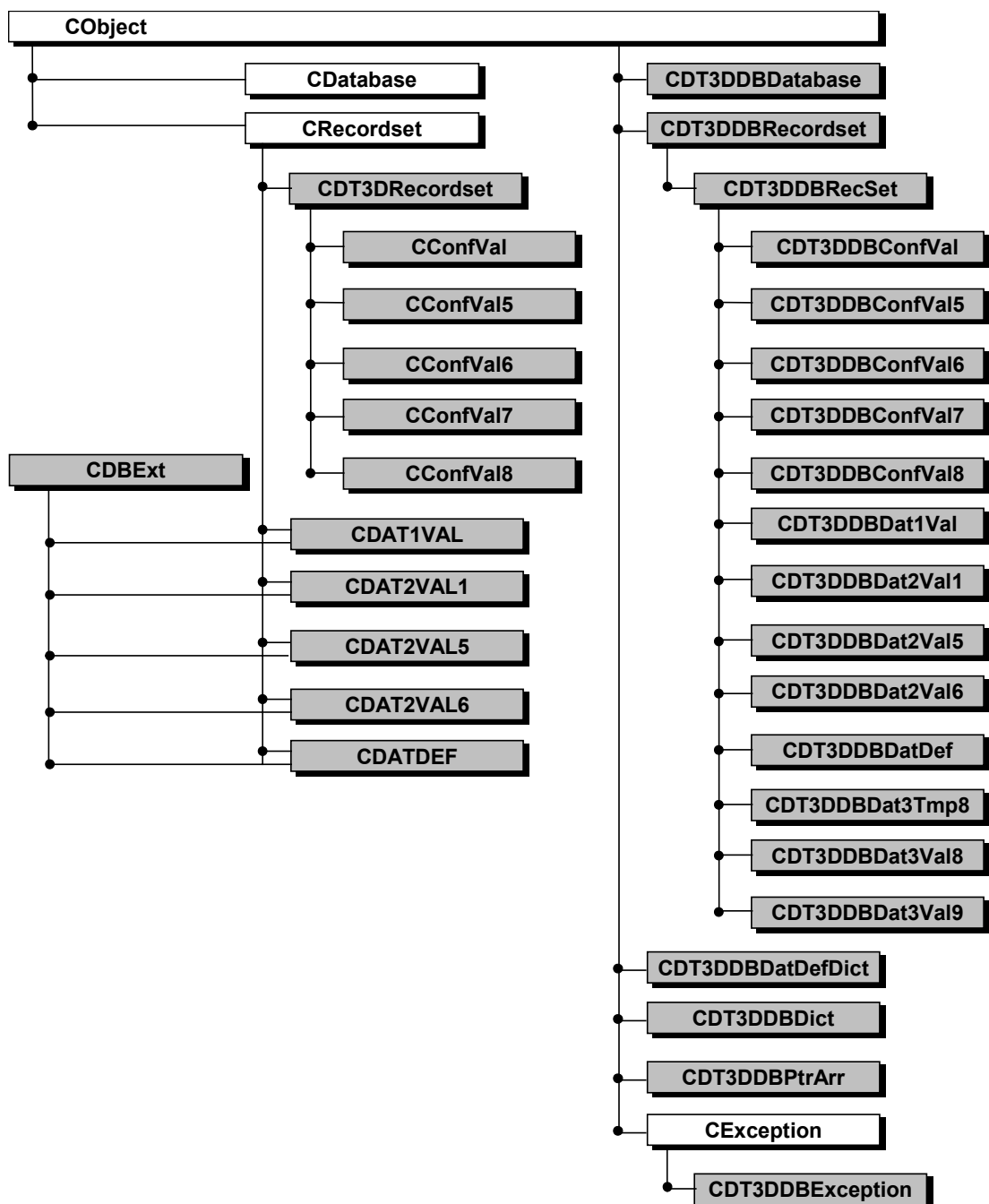


Rys. 37. Struktura klas DT3DMainApp i fragment struktury klas MFC

W klasie tej zdefiniowano zmienne i funkcje wspólne dla kilku tablic, a związane z zapisaniem podstawowych informacji o wybranych tablicach w bazie danych, sposobem odczytywania danych z tych tablic, a także z określeniem przedziału czasu z jakiego dane mają zostać odczytane z bazy.

### Klasa CDT3DRecordset

Klasa CDT3DRecordset jest klasą bazową dla wszystkich klas umożliwiających dostęp do tablic w bazie danych systemu DT200. Umożliwia



Rys. 38. Struktura klas biblioteki DT3DDB

pobieranie, modyfikowanie i dodawanie nowych danych do tych tablic.

### **Klasa CDT3DDBRecordset**

Klasa CDT3DDBRecordset jest klasą bazową dla wszystkich klas umożliwiających dostęp do tablic w bazie danych systemu DT200, poprzez funkcje biblioteki DB-Library. Klasa CDT3DDBRecordset jest odpowiednikiem klasy CRecordset zdefiniowanej w bibliotece MFC.

### **Klasa CDT3DDBRecSet**

Klasa CDT3DDBRecSet jest klasą pośrednią, zdefiniowaną ze względu na wspólne działania wykonywane na wszystkich tablicach bazy danych systemu DT200.

### **Klasa CDT3DDBException**

Klasa **CDT3DDBException** została zdefiniowana dla potrzeb biblioteki **DT3DDB**, aby można było obsługiwać błędy w działaniu biblioteki. Klasa dziedziczy z klasy CException.



## 6. Uwagi końcowe i wnioski

### 6.1. Podsumowanie

Zagadnienia związane z modelowaniem zmian stanów w układach nie są zagadnieniami nowymi. W pracy zamieszczono przegląd dotychczas stosowanych metod opisu stanów, takich jak:

- macierzowa metoda badania stanu diagnostycznego,
- modele matematyczne bazujące na rozkładach zmiennych losowych,
- sieci Petriego,
- automaty skończone,
- sieci przekonań,
- modele obiektowe,
- zbiory stwierdzeń,
- drzewa uszkodzeń.

Przedstawiono ich zalety i wady. Stwierdzono, że stosowane często rozwiązania, opierające się głównie na elementach dwustanowych, są niewystarczające.

Celem rozprawy było opracowanie sposobu reprezentacji propagacji stanów w układach maszyn. W wyniku analizy stosowanych obecnie rozwiązań dotyczących opisu stanów i przeglądu technik stosowanych w rozwoju oprogramowania, zaproponowano koncepcję opracowania wielowarstwowego modelu procesu propagacji stanów. Koncepcja modelu opiera się na pojęciu modułu. Moduły są modelami rzeczywistych oraz abstrakcyjnych elementów rozpatrywanego obiektu badań. Przygotowano definicję modułu podstawowego, która może być wykorzystana do modelowania różnych klas elementów obiektu. Poprawność koncepcji została zweryfikowana poprzez opracowanie implementacji modelu oraz jego przykładowego zastosowania. Moduł został zdefiniowany w ten sposób, że może zawierać wewnątrz inne moduły. Takie rozwiązanie znane jest z podejścia obiektowego i nazywane zagnieżdżaniem. Umożliwia to opracowanie modelu ze zmiennym stopniem szczegółowości. W celu uproszczenia zadań związanych z opracowaniem przygotowanego modelu wprowadzono pojęcie warstwy. Zastosowanie warstw w opracowanym modelu pozwala skupić uwagę na wybranych aspektach działania modelowanego obiektu badań. Różne aspekty działania układu są następnie uwzględniane w ramach kolejnych warstw ustalających łącznie działanie modułu.

## 6.2. Wnioski wynikające z pracy

W wyniku przeprowadzonych badań można sformułować następujące wnioski dotyczące tej pracy:

Słuszność tezy 1. wykazano przez opracowanie modelu w postaci systemu wielu modułów. Pokazano, że system taki może być zapisany w postaci sieci, w której poszczególne węzły odpowiadają uniwersalnym modułom, wykonującym określone działania. Działania wykonywane przez moduły umieszczone w poszczególnych węzłach takiej sieci zależą od wielu czynników. Pierwszą grupą czynników, wpływających na rodzaj wykonywanych działań, są wyniki obserwacji obiektu, czyli dane uzyskane z rzeczywistego obiektu. Mogą to być np. wartości cech sygnałów diagnostycznych, dotyczących tego fragmentu obiektu, który został zamodelowany przez moduł. Druga grupa czynników, które mogą mieć wpływ na działanie modułu, to wartości stwierdzeń (między innymi orzekających o stanach technicznych) wyznaczanych przez inne moduły. Wykazanie słuszności tezy 1. było możliwe po przyjęciu założenia, że bezpośredni wpływ na działanie modułu mają tylko moduły sąsiadujące z nim (moduły podrzędne, sąsiadujące oraz moduł nadrzędny). Pośrednio natomiast mogą wpływać pozostałe moduły, poprzez oddziaływanie na moduły sąsiadujące z danym modułem. Równocześnie każdy z modułów może wpływać na inne moduły poprzez wyznaczone przez niego wartości stwierdzeń. Zastosowano pojęcie węzła aktywnego, tzn. oddziałującego na inne węzły. Wykazano również, że poprzez opracowanie sieci propagacji stanów w postaci wielowarstwowych widoków, z których każdy uwzględnia jedynie pewną grupę oddziaływań występujących pomiędzy węzłami sieci, upraszcza się proces budowania takiej sieci. Pokazano, że grupowanie oddziaływań według ich typu upraszcza sposób opisu oddziaływań występujących pomiędzy poszczególnymi elementami rzeczywistego obiektu badań.

Słuszność tezy 2. wykazano przez podanie w rozdziale 3.6 definicji podstawowej klasy modułu, wspólnej dla wszystkich modułów stosowanych w opracowanym modelu. Przyjęto, że działanie takiego modułu będzie się opierać na pojęciu stwierdzenia. W module wyróżniono układ wewnętrznych wejść modułu, poprzez który następuje oddziaływanie z innymi modułami działającymi w tej samej sieci. Dla potrzeb opisu oddziaływań między modułami założono, że uwzględniane są jedynie bezpośrednie oddziaływania z modułu nadrzędnego, modułów podrzędnych oraz sąsiadujących. Pokazano, że przyjęcie jednej podstawowej klasy modułu znacznie upraszcza modelowanie procesu propagacji stanów i budowę całego modelu.

Przedstawione w pracy rozwiązanie pozwalające na opis stanów wielowartościowych jest rozwiązaniem nowatorskim i oryginalnym. Zdaniem

autora, stanowi rozwinięcie zagadnień związanych z opisem stanów i ciekawe uzupełnienie dotychczas stosowanych metod. Na szczególną uwagę zasługuje podjęta przez autora próba opisu zmian stanów w układach maszyn. Myślowy podział modułu, przez zdefiniowanie odpowiednich widoków, na fragmenty opisujące różne typy oddziaływań, jakie mogą wystąpić między modułami, pozwala na modelowanie nawet bardzo złożonych układów, w których na stan jednego modułu wpływ ma wiele zróżnicowanych czynników związanych z innymi modułami.

### **6.3. Kierunki dalszych badań**

Autor zdaje sobie sprawę, iż zaproponowane rozwiązanie wymaga jeszcze przeprowadzenia wielu dalszych prac związanych z przetestowaniem modelu na rzeczywistych obiektach. Szczególnie ciekawe mogą być prace związane z określaniem sposobów wyznaczania wartości stwierdzeń i poszukiwaniem optymalnych wartości stopni konieczności i możliwości. Warto również podjąć dalsze badania w kierunku rozwinięcia metod związanych z odkrywaniem nowych reguł [83]. Badania takie można prowadzić w trybie adaptacyjnym na podstawie wyników otrzymanych w czasie działania modelu. Będą one szczególnie przydatne wtedy, gdy będą oparte na danych pochodzących z rzeczywistego obiektu. Można przypuszczać, że dzięki współpracy Katedry Podstaw Konstrukcji Maszyn Politechniki Śląskiej, z Elektrownią „Kozienice” S.A., takie badania będą możliwe do zrealizowania.



## Literatura

- [1] Adamkiewicz W., Hempel L., Podsiadło A., Śliwiński R.: *Badania i ocena niezawodności maszyny w systemie transportowym*. WKiŁ, Warszawa 1983.
- [2] Ambroszkiewicz S., Penczek W.: *Modelling Rational BDI-Agents Within the Framework of Asynchronous Automata*. ICS Report 822, 1997
- [3] Barlow R.E., Proschan F.: *The Mathematical Theory of Reliability*. New York-London-Sidney, Willey 1965.
- [4] Barschdorff D., Moczulski W., Pfläging G.: *Motordiagnose unter Anlaufbedingungen mit Hilfe von Mustereckennungsverfahren*. Mustereckennung 1986, 8 DAGM-Symposium, Informatik-Fachberichte 125, s. 149-153, Springer-Verlag, Berlin Heidelberg.
- [5] Będkowski L.: *Elementy diagnostyki technicznej (rozdział 8)*. WAT, Warszawa 1991.
- [6] Bobrowski D.: *Modele i metody matematyczne teorii niezawodności w przykładach i zadaniach*. WNT Warszawa 1985
- [7] Bogdanowicz L.: *Podstawowe techniki rozwoju oprogramowania*. Wirtualna Akademia <http://www.ue.eti.pg.gda.pl/WA/KI/a-p-k/dok02-00.htm>, 1995.
- [8] Bromirski J.: *Teoria automatów*. WNT Warszawa 1969.
- [9] Cempel Cz., Tomaszewski F. (red.): *Diagnostyka maszyn. Zasady ogólne. Przykłady zastosowań*. Międzyresortowe Centrum Naukowe Eksploatacji Majałtku Trwałego, Radom 1992
- [10] Cempel Cz.: *Diagnostyka wibroakustyczna maszyn*. PWN, Warszawa 1989.
- [11] Cempel Cz.: *Ewolucyjne modele symptomowe w diagnostyce maszyn*. Kongres Diagnostyki Technicznej KDT'96, Gdańsk, 1996, t. 1, s. 31-38.
- [12] Cetnarowicz K., Nawarecki E., Żabińska M.: *M-Agent Architecture and Its Application to the Agent Oriented Technology of Decentralized System*. Proc. Of the DAIMAS'97. International workshop Distributed Artificial Intelligence and Multi-Agent Systems, St. Petersburg, Russia 1997.
- [13] Chmurawa M. Bińkowski W.: *Podstawy niezawodności i eksploatacji maszyn roboczych*. Skrypt Politechniki Śląskiej Nr 936, Gliwice 1980.

- [14] Cholewa W.: *Diagnostyczny system doradczy DT3D100. Organizacja procesu wnioskowania*. Raport częściowy nr DT6D131 z realizacji projektu PBZ-038-06, KPKM Politechniki Śląskiej, Gliwice, 1997.
- [15] Cholewa W.: *Dynamiczne systemy doradcze w diagnostyce technicznej*. Kongres Diagnostyki Technicznej. Materiały. Gdańsk 1996, tom I, s. 57-84.
- [16] Cholewa W.: *Metoda diagnozowania maszyn z zastosowaniem zbiorów rozmytych*. Zeszyty Naukowe Politechniki Śląskiej „Mechanika”, Zeszyt 79, Gliwice, 1983.
- [17] Cholewa W.: *Proces wnioskowania w systemie diagnostycznym DT200*. Diagnostyka Procesów Przemysłowych, Jurata k/Gdańska, 1998, s.157-162.
- [18] Cholewa W.: *Tablice ogłoszeń w diagnostyce technicznej*. Diagnostyka Procesów Przemysłowych, Podkowa Leśna k/Warszawy, 1996, s.42-47.
- [19] Cholewa W.: *Tablice ogłoszeń w diagnostycznych systemach doradczych*. PAK, Nr 4, 1998.
- [20] Cholewa W., Drobnia S., Elsner W., Moczulski W., Piątkowski J., Psiuk K., Skarka W., Wyczółkowski W., Wysogład B.: *Katalog relacji diagnostycznych*. Zeszyt Katedry Podstaw Konstrukcji Maszyn, RMT6514, 1999.
- [21] Cholewa W. Kiciński J.(red.): *Diagnostyka techniczna. Odwrotne modele diagnostyczne*. Monografie – Budowa i Eksploatacja Maszyn, Wydawnictwo Politechniki Śląskiej, 1997.
- [22] Cholewa W., Kaźmierczak J.: *Diagnostyka techniczna maszyn. Przetwarzanie cech sygnałów*. Skrypt Politechniki Śląskiej Nr 1693, Gliwice 1992.
- [23] Cholewa W., Kiciński J. (red.): *DT200-1. System diagnostyczny dla turbozespołów energetycznych o mocy 200 MW*. Wydawnictwo IMP PAN, Gdańsk 1998.
- [24] Cholewa W., Moczulski J.: *Diagnostyka techniczna maszyn. Pomiary i analiza sygnałów*. Skrypt Politechniki Śląskiej Nr 1758, Gliwice 1993.
- [25] Cholewa W., Pedrycz W.: *Systemy doradcze*. Skrypt Politechniki Śląskiej Nr 1447, Gliwice 1987.
- [26] Cholewa W., Psiuk K.: *'Interface' użytkownika systemu DT200. Proces rozwoju oprogramowania*. Raport częściowy nr DT6D186 z realizacji projektu PBZ-038-06, KPKM Politechniki Śląskiej, Gliwice, 1997.

- [27] Cholewa W., Psiuk K.: *Tablica ogłoszeń w systemie DT200. Interface użytkownika*. Raport RMT6443, KPKM, Politechnika Śląska, Gliwice, 1998.
- [28] Cholewa W., Psiuk K.: *Wizualizacja stanu złożonego obiektu technicznego*. Diagnostyka Procesów Przemysłowych, Kazimierz Dolny, 1999, s.173-178.
- [29] Cholewa W., Wysogład B., Psiuk K.: *Tablica ogłoszeń w systemie DT200. Wyjściowa baza wiedzy*. Raport RMT6444, KPKM, Politechnika Śląska, Gliwice, 1998.
- [30] Daciuk J.: *Finite-State Automata in Document Retrieval*. - Intelligent Information Systems, Malbork, 1998, s. 199-202.
- [31] Dietrych J.: *System i konstrukcja*. WNT, Warszawa, 1985.
- [32] Drusinski D., Harel D.: *Using Statecharts for Hardware Description and Synthesis*. IEEE Transaction on Computer-Aided Design of Integrated Circuits & Systems, vol. 8, no. 7, 1989, s. 798-807.
- [33] Findeisen W. (red.): *Poradnik inżyniera. Automatyka*. WNT, Warszawa 1973.
- [34] Firkowicz S.: *Statystyczna ocena jakości i niezawodności lamp elektronowych*. WNT, Warszawa, 1963.
- [35] Franklin S., Greasser A.: *Is it an Agent or just a Program? A Taxonomy for Autonomous Agents*. Third International Workshop on Agent Theories, Architectures, and Languages, Springer-Verlag, 1996.
- [36] Friedman A.D., Menon P.R.: *Wykrywanie uszkodzeń w układach cyfrowych*. WNT, Warszawa 1978.
- [37] Gerbach I.B., Kordoński Ch.B.: *Modele niezawodnościowe obiektów technicznych*. WNT Warszawa 1968.
- [38] Girtler J.: *Modele matematyczne w badaniach niezawodności silników o zapłonie samoczynnym*. Explo-Diesel '98
- [39] Gniedenko B.W., Bielajew J.K., Sołowiew A.D.: *Metody matematyczne w teorii niezawodności*. WNT, Warszawa, 1968.
- [40] Harel D.: *A Visual Formalism for Complex Systems*. Science of Computer Programming 8, 1987, s. 231-274
- [41] Harrison M.A.: *Wstęp do teorii sieci przełączających i teorii automatów*. PWN Warszawa 1973.
- [42] Hayes-Roth B.: *An Architecture for Adaptive Intelligent Systems*. Artificial Intelligence: Special Issue on Agents and Interactivity, 72, s. 329-365.

- [43] Hebda M.: *Elementy teorii eksploatacji systemów technicznych*. MCNEMT, Radom 1990.
- [44] Hopcroft J.E., Ullman J.D.: *Wprowadzenie do teorii automatów, języków i obliczeń*. WNT Warszawa 1994.
- [45] Iserman R.: *Supervision, Fault-Detection and Fault-Diagnosis Methods – An Introduction*. Control Engineering Practice, Vol. 5, No. 5, s. 639-652.
- [46] Jennings N.R., Wooldridge M. (red.): *Agent Technology: Foundations, Applications, and Markets. Applications of Intelligent Agents*. Springer, 1998, pp. 3-28
- [47] Kiciński J.: *Hydrodynamiczne poprzeczne łożyska ślizgowe*. Wydawnictwo IMP PAN, Gdańsk 1996.
- [48] Kiliński A.: *Definicje opisowe i normatywne pojęcia niezawodności*. Problemy Jakości Nr 3, Warszawa 1972, s.5-13.
- [49] Kłopotek M.A.: *Jakościowe oraz ilościowe aspekty wnioskowania w sieciach bayesowskich*. URL:  
<http://alpha.mini.pw.edu.pl/~klopotek/suiwon/siecibayesa.html>
- [50] Kohavi Z.: *Switching and Finite Automata Theory*. McGraw-Hill 1978.
- [51] Korbicz J., Pieczyńska A., Marczevska G.: *Ekspertowy system diagnostyczny dla symulatora elektrowni cieplnej*. Mat. III Krajowej Konferencji Diagnostyka Techniczna Urządzeń i Systemów, Szczyrk, 10-13 październik 1995, tom I, s. 145-153.
- [52] Korbicz J.: *Metody sztucznej inteligencji w diagnostyce technicznej. Przegląd struktur i technik*. Kongres Diagnostyki Technicznej. Materiały. Gdańsk 1996, tom II, s. 373-380.
- [53] Kosmol J.: *Monitorowanie stanu ostrza skrawającego: Metody konwencjonalne i sieci neuronowe*. WNT, Warszawa 1996.
- [54] Maes P.: *Artificial Life Meets Entertainment: Life like Autonomous Agents*. Communications of the ACM, 38, 11, str. 108-114.
- [55] Mannila H., Ronkainen P.: *Similarity of Event Sequences*. IEEE Transactions on knowledge and data engineering, 1997, s. 136-139.
- [56] Matlab. Users guide - The MathWorks Inc, 1999.
- [57] Mesarovic M.D., Takahara Y.: *General Systems Theory: Mathematical Foundations*. Academic Press, New York 1975.
- [58] *Microsoft Foundation Class Reference*. Microsoft, 1997.
- [59] Młokosiewicz J.R., Tomkiewicz K.: *Indukcyjny system informacyjny w diagnozowaniu złożonych obiektów technicznych*. Zagadnienia Eksploatacji Maszyn Zeszyt 1 1995, s.137-148.



- [60] Młokosiewicz J.R., Więcek A.: *Kryterium wyznaczania optymalnych programów diagnostycznych w przypadku trójwartościowej klasyfikacji stanów obiektu*. Biuletyn Wojskowej Akademii Technicznej im. J. Dąbrowskiego, nr 12, 1986, s.115-125.
- [61] Młokosiewicz J.R.: *Badanie stanu diagnostycznego zintegrowanego systemu sterowania*. Postępy cybernetyki Nr 4, 1983, s.213-228.
- [62] Młokosiewicz J.R.: *Macierzowa metoda badania stanu diagnostycznego systemów*. Postępy cybernetyki Nr 4 1983, s.199-212.
- [63] Moczulski W.: *Indukcyjne pozyskiwanie wiedzy diagnostycznej dla złożonej struktury drzewa stanów*. Diagnostyka Procesów Przemysłowych, Jurata k/Gdańska, 1998, s.145-156
- [64] Moczulski W.: *Metody pozyskiwania wiedzy dla potrzeb diagnostyki maszyn*. Zeszyty Naukowe Politechniki Śląskiej „Mechanika”, z. 130, Gliwice 1997.
- [65] Oprzędkiewicz J.: *Wspomaganie komputerowe w niezawodności maszyn*. WNT, Warszawa, 1993.
- [66] Patton R., Frank P., Clark R. (Eds.): *Fault Diagnosis in Dynamic Systems – Theory and Applications*. Prentice Hall, Englewood Cliffs, NJ 1989.
- [67] Pau L.F.: *Failure Diagnosis and Performance Monitoring*. Marcel Dekker, New York, 1981.
- [68] PBZ-038-06 pt. ”Opracowanie nowoczesnych systemów diagnostycznych turbozespołów krajowych bloków energetycznych dużej mocy.”, 1997.
- [69] Pearl J.: *Fusion, Propagation, and Structuring in Belief Networks*. Artificial Intelligence 29, 1986
- [70] PN-80/N-04000 Niezawodność w technice. Terminologia.
- [71] Psiuk K.: *Biblioteka klas DT3DDB do współdziałania z bazą danych. Dokumentacja programisty*. Raport częściowy nr DT6D161 z realizacji projektu PBZ-038-06, KPKM Politechniki Śląskiej, Gliwice, 1998.
- [72] Psiuk K.: *Biblioteka klas DT3DMainApp do współdziałania z bazą danych. Dokumentacja programisty*. Raport częściowy nr DT6D184 z realizacji projektu PBZ-038-06, KPKM Politechniki Śląskiej, Gliwice, 1997.
- [73] Psiuk K.: *Diagnostyczne modele relacyjne o zmiennym stopniu szczegółowości*. - Kongres Diagnostyki Technicznej KDT’96, Gdańsk, 1996, s.181-184.

- [74] Psiuk K.: *Diagnostyczny, obiektowy model złożonego układu na przykładzie turbozespołu*. Diagnostyka Procesów Przemysłowych, Łągów k/Zielonej Góry, 1997.
- [75] Psiuk K.: *Modelling of Reliability and/or Damages Propagation in Large Machinery Sets*. Intelligent Information Systems, Zakopane, 1997, s. 36-40.
- [76] Psiuk K.: *Stanowisko diagnostyczne systemu DT200. Założenia dotyczące rozwoju oprogramowania*. Raport częściowy nr DT6D182 z realizacji projektu PBZ-038-06, KPKM Politechniki Śląskiej, Gliwice, 1997.
- [77] Psiuk K.: *Zastosowanie wielowarstwowej sieci do reprezentowania niesprawności układów*. Diagnostyka Procesów Przemysłowych, Podkowa Leśna k/Warszawy, 1996, s.76-79.
- [78] Psiuk K.: *Modelowanie propagacji stanów w układach maszyn*. Methods of Artificial Intelligence in Mechanics and Mechanical Engineering, Gliwice, 2000, s. 277-284.
- [79] Psiuk K.: *Monitorowanie propagacji zmian stanów elementów w układach maszyn*. II Międzynarodowy Kongres Diagnostyki Technicznej, Warszawa 2000.
- [80] Psiuk K.: *Modelowanie zmian stanów obiektów technicznych na przykładzie turbozespołu*. Problemy Badawcze Energetyki Ciepłej, Warszawa 1999, s. 169-176.
- [81] Psiuk K. Skarka W.: *New Idea of Employing User's Technical Documentation Connected with Expert System*. Computer Integrated Manufacturing Volume II, WNT Warszawa 1999, s. 162-171.
- [82] Psiuk K. Skarka W.: *Nowe funkcje dokumentacji techniczno-ruchowej zastosowanej w systemie doradczym*. III Szkoła komputerowego wspomaganie projektowania, wytwarzania i eksploatacji, Helion, Gliwice 1999, s. 553-562.
- [83] Psiuk K.: *Zastosowanie oprogramowania MATLAB/STATEFLOW do modelowania i badania reguł diagnostycznych*. Diagnostyka Procesów Przemysłowych, Jurata k/Gdańska, 1998, s. 61-66.
- [84] Rivest, R.L., Schapire, R.E.: *Inference of Finite Automata Using Homing Sequences*. Information and Computation, 103:299-347, 1993
- [85] Rozwadowski T.: *Diagnostyka techniczna obiektów złożonych*. WAT, Warszawa 1974.
- [86] Russell S. J., Norvig P.: *Artificial Intelligence: A Modern Approach*. Englewood Cliffs, NJ Prentice Hall 1995.

- [87] Simulink. Users guide - The MathWorks Inc, 1999.
- [88] *Słownik Języka Polskiego* PWN Warszawa 1979.
- [89] Stateflow. Users guide - The MathWorks Inc, 1997.
- [90] Taylor D.A. *Technika obiektowa*. Helion, Gliwice, 1994.
- [91] Vaskevitch D.: *Strategie Klient-Serwer. Systemy informatyczne w procesie przekształcania przedsiębiorstwa*. IDG Poland S.A. Warszawa 1995.
- [92] Wooldridge M., Jennings N. R.: *Intelligent Agents: Theory and Practice*. The Knowledge Engineering Review, 10(2) ss. 115-152, 1995.
- [93] Yufeng L., Shuzhen Y.: *Research on the Multi-Agent Model of Autonomous Distributed Control Systems*. IEEE, 1999.
- [94] Zdanowicz R.: *Dyskretne procesy produkcyjne. Laboratorium z modelowania i symulacji*. Skrypt Politechniki Śląskiej Nr 2125, Gliwice 1998.
- [95] Żółtowski B., Ćwik Z.: *Leksykon diagnostyki technicznej*. Wydawnictwo Uczelniane Akademii Techniczno-Rolniczej, - Bydgoszcz 1996.
- [96] Żółtowski B.: *Podstawy diagnostyki maszyn*. Wydawnictwo Uczelniane Akademii Techniczno-Rolniczej, Bydgoszcz 1996.



# Obiektowe modele procesu propagacji stanów w układach maszyn

## Streszczenie

Monitorowanie stanu maszyn, szczególnie dużych i skomplikowanych, należy do jednego z ważniejszych zadań. Jest to związane z obniżeniem kosztów ich eksploatacji. Opracowywane są różnego typu systemy nadzoru, służące nadzorowaniu działania maszyny, jak również służące do oceny jej aktualnego stanu, czy stopnia zużycia maszyny. Obecnie w układach monitorowania stanu maszyn coraz częściej stosowane są systemy doradcze. Stosowanie systemów doradczych ma na celu przeprowadzenie wstępnej analizy danych, a na tej podstawie podejmują decyzje dotyczące dalszych analiz lub podejmowane są próby oceny stanu maszyny. Jednym z głównych elementów systemów doradczych jest baza wiedzy. Bardzo często wiedza w bazie wiedzy jest zapisywana w postaci reguł. Jedną z metod weryfikacji istniejących reguł jest modelowanie propagacji stanów. Modelowanie propagacji stanów umożliwia nie tylko weryfikację reguł istniejących, ale również odkrywanie nowych.

W pracy przedstawiono metody i techniki, które wykorzystywane są do opisu stanów. Zaproponowano nową metodę opisu propagacji stanów. Metoda ta oparta jest na pojęciu stwierdzenia i autonomicznego modułu. W zaproponowanym rozwiązaniu maszyna jest traktowana jako moduł złożony, zawierający inne moduły. Moduł jest fragmentem maszyny, takim jak np. element, podzespół, zespół lub dowolnym innym fragmentem maszyny, składającym się z wymienionych wcześniej. Zdefiniowano moduł podstawowy i wykonywane przez niego działania, oparte na stwierdzeniach, a odwzorowujące zewnętrzne i wewnętrzne oddziaływania działające na niego. Oddziaływania te zostały podzielone, w zależności od ich typu. Każdy z typów oddziaływań między modułami można przedstawić w postaci warstwy. Łącząc wszystkie warstwy, otrzymuje się system wielowarstwowy. Jedną z warstw ma specjalne przeznaczenie. Warstwa ta opisuje strukturę modelowanej maszyny. Wszystkie pozostałe warstwy są jedynie odpowiednimi widokami modułów znajdujących się na tej warstwie.

Stan całej maszyny jest określany jako funkcja stanów poszczególnych modułów. Na stan każdego modułu składają się stany określone w poszczególnych warstwach. Do opisu propagacji stanów wykorzystano samodzielne moduły wyznaczające wartości stwierdzeń, zapisywane następnie na tablicy ogłoszeń.

W pracy przedstawiono również implementację zaproponowanej metody. Rozwiązanie zostało zaimplementowane z wykorzystaniem środowiska Matlab, pakietów Stateflow oraz Simulink, a także środowiska baz danych MS Access i MS SQL Serwer.

# Object Models of State Propagation in Machine Sets

## Summary

Monitoring of state in machines, especially large and complicated ones is a very important task. This is connected with exploitation and reducing its costs. Various kinds of monitoring systems are prepared to increase reliability of a machine and also to estimate its current state or wearing degree. Nowadays in monitoring systems more and more often attempts on using expert systems are take on. The expert systems are used to carry out preliminary data analysis and on the basis of these, the systems try to estimate the state of the machine. One of the most important elements of the expert systems is knowledge database. Very often the knowledge in expert systems is represented as rules. One of the methods of rules verification is state propagation modelling. Using this method, it is possible to verify existing rules and discover the new ones.

In this dissertation several methods and techniques that are used for state description are presented, paying special attention to: finite state automata, belief networks and object models. A new method used for state propagation is proposed. This method is based on statements and autonomous modules. In the proposed solution investigated machine is described as a complex modules, that contains other modules. The module is a subsystem of the machine such as a part, component or complex or some other subsystem that is composed of these subsystems. A base module and its action are defined. These actions are based on statements and describe internal and external interactions that influence it. It is possible to define some kinds of interactions between the modules. These interactions come e.g. from physical actions or machine construction. All interactions are divided according to interaction type. Each type of interactions can be presented as a layer. When joining these layers we get a multilayer system. One layer has a special assignment. It describes the structure of a modelled machine. All other layers are a special view of modules contained on this one.

The state of whole machine is determined as function of all modules state. The state of each module consists of states that are estimated on each layer. It is defined a based module which functioning is grounded on statements. Autonomous modules are used for state propagation between modules. Each module evaluates statement values and save this values on blackboard.

In this dissertation an implementation of this method is presented. This solution is implemented with the help of Matlab computational environment with Stateflow toolbox and Simulink as well as with MS Access and MS SQL Server database systems.