Politechnika Śląska
Wydział Mechaniczny Technologiczny
Instytut Podstaw Konstrukcji Maszyn

ROZPRAWA DOKTORSKA

Analiza ryzyka w inżynierii wymagań dla systemów diagnostycznych

Marcin AMAROWICZ

Promotor:
prof. dr hab. inż. Wojciech CHOLEWA

Gliwice, 2015

Nie można zrozumieć procesu przez zatrzymanie go.
Zrozumienie musi podążać z biegiem procesu,
musi przyłączyć się i płynąć razem z nim.

I prawo Mentata, F. Herbert – Diuna

Promotorowi niniejszej pracy, dziękuję
za wskazanie tematu oraz za opiekę naukową.

Autor

W okresie 01.10.2012 – 30.09.2014 doktorant był Stypendystą, w ramach projektu
DoktoRIS – Program stypendialny na rzecz innowacyjnego Śląska,
współfinansowanego przez Unię Europejską w ramach Europejskiego Funduszu Społecznego.

Wybrane fragmenty pracy, realizowane były częściowo w ramach Zadania Badawczego Nr 4,
Strategicznego Programu Badań *Zaawansowane technologie pozyskiwania energii*,
ustanowionego przez Narodowe Centrum Badań i Rozwoju.

Spis treści

Wykaz ważniejszych skrótów i oznaczeń **vi**

1 Wprowadzenie **1**
1.1 Charakterystyka problemu badawczego 1
1.2 Cel i tezy rozprawy 3
1.3 Zakres rozprawy 4
1.4 Podstawowe terminy 4

2 Diagnozowanie stanu maszyn i procesów **6**
2.1 Proces eksploatacji obiektów technicznych 6
2.2 Stan techniczny obiektu 9
2.3 Metody i techniki diagnostyczne 12
2.3.1 Diagnostyka symptomowa 12
2.3.2 Diagnostyka wsparta modelem 12
2.4 Systemy monitorowania i diagnostyki 13
2.5 Podsumowanie 15

3 Ryzyko techniczne w eksploatacji maszyn i procesów **16**
3.1 Ogólna problematyka ryzyka 16
3.2 Ryzyko techniczne 17
3.3 Wybrane metody oceny ryzyka technicznego 21
3.4 Zarządzanie ryzykiem technicznym 22
3.5 Podsumowanie 24

4 Wybrane zagadnienia inżynierii wymagań **25**
4.1 Zarys problematyki 25
4.2 Wymagania oraz ich rodzaje 26
4.3 Definiowanie zbioru wymagań 27
4.4 Zarządzanie procesem definiowania wymagań 31
4.5 Ograniczenia w procesie pozyskiwania wymagań 33
4.6 Podsumowanie 34

5 Metody i techniki sztucznej inteligencji w diagnostyce **35**
5.1 Idea sztucznej inteligencji 35
5.2 Wiedza diagnostyczna oraz metody jej pozyskiwania 36
5.3 Metody reprezentacji wiedzy 38

5.3.1 Reguły . . . 38
5.3.2 Zbiory rozmyte . . . 39
5.3.3 Sieci stwierdzeń . . . 40
5.3.4 Ontologie i sieci semantyczne . . . 42
5.4 Systemy doradcze . . . 44
5.5 Podsumowanie . . . 45

6 Metoda projektowania systemów diagnostycznych z zastosowaniem zbiorów wymagań 47
6.1 Założenia i uwagi wstępne . . . 47
6.2 Charakterystyka proponowanego podejścia projektowego . . . 48
6.3 Zapis zbioru wymagań . . . 50
6.4 Pozyskiwanie zbioru wymagań . . . 53
6.5 Generowanie zbioru potencjalnych rozwiązań systemu diagnostycznego . . . 54
6.6 Ocena zbioru wymagań oraz potencjalnych rozwiązań systemu diagnostycznego . 55
6.7 Podsumowanie . . . 58

7 Implementacja zaproponowanej metody 60
7.1 Struktura i funkcjonalności *Edytora wymagań* . . . 60
7.2 Praca z *Edytorem wymagań* . . . 64
7.3 Podsumowanie . . . 69

8 Weryfikacja opracowanej metody 71
8.1 Cel i zakres badań weryfikacyjnych . . . 71
8.2 Charakterystyka obiektu badań . . . 71
8.3 Proces pozyskiwania wymagań oraz formalizacja potrzeby . . . 73
8.4 Ocena zebranego zbioru wymagań oraz optymalizacja postaci systemu diagnostycznego . . . 80
8.5 Przykłady projektów systemów diagnostycznych . . . 83
8.6 Podsumowanie . . . 88

9 Podsumowanie i wnioski 89
9.1 Podsumowanie . . . 89
9.2 Spostrzeżenia i wnioski . . . 90
9.3 Kierunki dalszych badań . . . 90

Bibliografia 92

Spis rysunków 106

Spis tabel 107

Streszczenie 108

Summary 109

Wykaz ważniejszych skrótów i oznaczeń

O_t	obiekt techniczny
R_t	ryzyko techniczne
P_i	prawdopodobieństwo wystąpienia i-tego uszkodzenia/zdarzenia
c_i	koszt wystąpienia i-tego uszkodzenia/zdarzenia
DS	system diagnostyczny
req_i	wymaganie funkcjonalne opisujące funkcjonalność systemu diagnostycznego
$drule_i$	reguła diagnostyczna realizująca funkcjonalność systemu diagnostycznego
$subs_i$	podsystem diagnostyczny
pf	stopień preferencji
REQ	zbiór wymagań funkcjonalnych
$DRULE$	zbiór reguł diagnostycznych
$SUBS$	zbiór podsystemów diagnostycznych

Rozdział 1

Wprowadzenie

W niniejszym rozdziale przedstawiono genezę badań doktorskich, nakreślono problem badawczy, sformułowano cel i tezy pracy oraz zwięźle przedstawiono zakres niniejszej rozprawy.

1.1. Charakterystyka problemu badawczego

Postęp techniki jest nieuchronnym skutkiem rozwoju każdej cywilizacji[1]. Wraz z nim opracowywane są coraz to nowsze środki techniczne, maszyny oraz urządzenia, których zadaniem jest zaspokajanie pojawiających się oczekiwań i potrzeb poszczególnych jednostek, wchodzących w skład danego społeczeństwa. Rozwój techniczny niesie za sobą możliwość wykonywania tych samych prac i czynności szybciej, dokładniej, efektywniej itd., przy jednoczesnym użyciu mniejszych nakładów finansowych, materialnych itp. W wielu przypadkach nowo opracowane rozwiązania pozwalają na znaczną poprawę jakości ludzkiego życia. Przykładami mogą być na przykład środki techniczne opracowane w ramach kolejnych rewolucji przemysłowych.

Wyróżnia się (Dietrych, 1978; Grieves, 2005) pięć faz/etapów życia każdego produktu, tj. projektowanie, konstruowanie, wytwarzanie, eksploatowanie oraz utylizowanie. W procesie projektowania opracowywany/obmyślany jest oczekiwany sposób działania przyszłego środka technicznego. W fazie konstruowania opracowanemu projektowi przypisywane są odpowiednie cechy konstrukcyjne, rozumiane jako struktura oraz układ wymiarów. Na podstawie opracowanej konstrukcji, w kolejnym etapie wytwarzany jest oczekiwany środek techniczny. Kolejny z etapów to eksploatacja środka technicznego zgodnie z przyjętymi w procesie projektowo-konstrukcyjnym założeniami. Ostatni z etapów to utylizacja środka technicznego, czyli wyłączenie go z eksploatacji, recykling wybranych jego podzespołów oraz unieszkodliwienie substancji niebezpiecznych.

Z uwagi m.in. na duży stopień złożoności współczesnych obiektów technicznych, innowacyjność zastosowanych rozwiązań, jak również możliwe błędy popełnione zarówno na etapie projektowo-konstrukcyjnym jak również podczas wytwarzania oraz z uwagi na nieprzestrzeganie zaleceń eksploatacyjnych, w wielu przypadkach proces użytkowania danego obiektu nie przebiega zgodnie z założonymi oczekiwaniami. Pojawiają się różnego rodzaju zdarzenia, których konsekwencją są m.in. odstępstwa od normy w jakości realizowanych działań, brak możliwości realizacji wybranych działań, uszkodzenia wybranych podzespołów a nawet konieczność całkowitego wyłączenia z użytkowania danego obiektu technicznego, po wystąpieniu określonych awarii.

[1] Cywilizacja wg F. Bacona to zdolność rozwiązywania problemów i zaspokajania potrzeb ludzkich.

We współczesnym szybko rozwijającym się świecie, nie można sobie pozwolić na żadne z wymienionych rodzai zdarzeń. Dlatego też konieczne staje się przeprowadzanie procesu oceny stanu technicznego eksploatowanych obiektów technicznych.

W większości przypadków, kroki zmierzające do określenia stanu technicznego obiektów, sprowadzają się do (Timofiejczuk, 2011):

- rejestracji danych (sygnałów diagnostycznych),
- analizy zgromadzonych danych, celem wyznaczenia cech zarejestrowanych sygnałów,
- konwersji otrzymanych danych na wiedzę opisującą rozpatrywany obiekt/proces,
- wnioskowania diagnostycznego o stanie rozpatrywanego obiektu lub procesu technicznego.

W przypadku prostych obiektów technicznych, dla których istnieje potrzeba/możliwość rejestracji i analizy niewielkiej liczby sygnałów, problem określenia stanu technicznego, poza pewnymi specyficznymi przypadkami, jest zadaniem stosunkowo prostym. Sytuacja niezmiernie jednak komplikuje się, w przypadku bardziej złożonych obiektów, dla których liczba dostępnych sygnałów może być liczona w tysiącach a poszczególne z nich, mogą być wynikiem różnego rodzaju procesów i ich oddziaływań z otoczeniem (Walsh i in., 2011).

Dla potrzeb efektywnego określania stanu technicznego wybranych obiektów, opracowywane mogą być stosowne systemy diagnostyczne. Wymagania stawiane tym systemom mogą być bardzo różne. Pomijając obiekty techniczne wytwarzane w ramach produkcji seryjnej, można stwierdzić że każdy system diagnostyczny jest w pewnym sensie unikalny i *szyty na miarę*.[2] Podejmując próbę opracowania systemu diagnostycznego, konieczne jest znalezienie odpowiedzi na dwa zasadnicze pytania, mianowicie:

- jakie powinny być funkcjonalności systemu diagnostycznego, tj. które stany techniczne rozpatrywanego obiektu, powinny być rozpoznawanie przez system diagnostyczny oraz jakie w tym celu powinny być zastosowane reguły diagnostyczne,
- jaka powinna być struktura systemu diagnostycznego, czyli jakie elementy powinny wchodzić w skład systemów pomiarowych oraz systemów wnioskowania.

Projektowany system diagnostyczny musi być uzasadniony ekonomicznie oraz dostarczać niezbędnych informacji, pozwalających na sformułowanie poprawnej diagnozy na temat stanu technicznego diagnozowanego obiektu. Opracowanie projektu odpowiedniego systemu diagnostycznego, jest w ogólnym przypadku zadaniem trudnym. Konieczne jest uwzględnienie wielu różnych czynników, wśród których można wymienić m.in. (Barszcz, 2006; Czech, 2011; Tarełko, 2011):

- szczegółową wiedzę o rozpatrywanym obiekcie technicznym, w tym informacje o jego strukturze,
- informacje o warunkach działania obiektu technicznego,
- ryzyko wystąpienia poszczególnych uszkodzeń w rozpatrywanym obiekcie,
- metody i techniki diagnostyczne, w tym m.in. metody pomiaru i analizy sygnałów itp.,
- ograniczenia projektowe, np. technologiczne, kosztowe itp.,
- warunki i wymagania biznesowe,
- koszty instalacji oraz przyszłej eksploatacji systemu diagnostycznego.

[2] W inżynierii oprogramowania popularne określenie systemów informatycznych, które są indywidualnie konfigurowane pod określone wymagania klienta.

Niezbędne do opracowania systemu diagnostycznego dane, mogą być pozyskiwane z wielu różnych źródeł, m.in. jako wyniki analiz literaturowych, opinie ekspertów, wyniki czynnych i biernych eksperymentów diagnostycznych, zalecenia producenta, obserwacje itp. Ogrom informacji koniecznych do uwzględnienia w tym procesie oraz fakt, że część z pozyskanych informacji może być niepewna lub niedokładna, sprawia że niezbędne jest zastosowanie odpowiedniego podejścia projektowego do procesu opracowywania projektu systemu diagnostycznego.

W przypadku szeroko rozumianego problemu projektowania maszyn, proces projektowy związany jest zazwyczaj (Pólya, 1957; Dietrych, 1978) z realizacją następujących etapów:

- rozpoznanie (identyfikacja) potrzeby – faza, w czasie której następuje identyfikacja i opis tego co jest nam potrzebne, przygotowany opis potrzeby nie może narzucać możliwych rozwiązań,
- analiza możliwości zaspokojenia potrzeby – faza, w czasie której podejmuje się próbę znalezienia możliwych rozwiązań, które mogłyby zaspokoić zdefiniowaną potrzebę,
- rozpoznanie ewentualnych ograniczeń – określenie zbioru możliwych ograniczeń (czasowych, kosztowych, technologicznych itp.), które mogą wpływać zarówno na sformułowaną potrzebę (np. może być ona niemożliwa do zaspokojenia, przy obecnym poziomie rozwoju technicznego), jak również na opracowane koncepcje rozwiązań (np. część z nich może być zbyt kosztowna w realizacji),
- wybór rozwiązania optymalnego – ze zbioru opracowanych potencjalnych rozwiązań, które mieszczą się w przyjętych ograniczeniach, wybierane jest na podstawie zdefiniowanych kryteriów rozwiązanie optymalne,
- implementacja wybranego rozwiązania celem osiągnięcia zakładanych celów projektu.

Bezpośrednie zastosowanie przedstawionego podejścia projektowego, przy próbie opracowania projektu systemu diagnostycznego napotyka na pewne opory na etapie opracowywania zbioru potencjalnych rozwiązań systemu diagnostycznego. Spowodowane jest to koniecznością uwzględnienia wielu różnych czynników, w tym m.in. rozległej wiedzy o możliwych uszkodzeniach rozpatrywanego obiektu i sposobach ich diagnostyki. Przeprowadzenie tego procesu wprost na podstawie słownego opisu potrzeby nie jest do końca możliwe. Powyższy problem może zostać rozwiązany poprzez zastosowanie dodatkowego etapu projektowego, w czasie którego przeprowadzony zostanie sformalizowany zapis zdefiniowanej potrzeby. Powinien on uwzględniać, wyniki analizy ryzyka oraz informacje o potencjalnych sposobach diagnostyki poszczególnych uszkodzeń.

1.2. Cel i tezy rozprawy

Celem podjętych w niniejszej rozprawie badań naukowych, jest opracowanie nowego podejścia do procesu opracowywania projektu systemu diagnostycznego, który umożliwiałby na etapie formułowania potrzeby, aktywne uwzględnianie wyników analizy ryzyka, dostępnych metod i technik pomiarowych oraz istniejących kryteriów ograniczających. Opracowane podejście projektowe powinno umożliwiać:

- aktywne wspomaganie procesu akwizycji wiedzy,
- integrację wiedzy dotyczącej działania, budowy oraz eksploatacji rozpatrywanego obiektu technicznego,
- analizę potrzeb oraz optymalizację założeń przy racjonalnych ograniczeniach,
- wybór rozwiązania optymalnego na podstawie zdefiniowanych kryteriów oceny.

Przyjmuje się, że większość czynności opracowanego podejścia projektowego, może zostać zrealizowana z poziomu specjalnie opracowanego, dedykowanego środowiska informatycznego implementującego wybrane etapy proponowanego podejścia projektowego.

Na podstawie przeprowadzonej analizy problemu badawczego oraz nakreślonego celu rozprawy, sformułowano następujące tezy pracy:

Teza 1 Sformalizowany zapis potrzeb dla projektowanego systemu diagnostycznego, może zostać zrealizowany poprzez zastosowanie zbioru wymagań.

Teza 2 Zbiór wymagań, może być identyfikowany w procesie wnioskowania, prowadzonym z zastosowaniem systemu doradczego, który reprezentuje kryteria pozwalające na uzyskanie efektywnego systemu diagnostycznego.

1.3. Zakres rozprawy

Rozprawa składa się z 9 rozdziałów, bibliografii, spisu rysunków i tabel oraz streszczeń polsko- i angielskojęzycznego. Rozdział pierwszy zawiera wprowadzenie do pracy, opisano w nim cel badawczy rozprawy oraz zawarto tezy pracy i słownik podstawowych terminów. W rozdziałach 2–5 przedstawiono wybrane zagadnienia związane z obszarami/dziedzinami, których bezpośrednio dotyczy realizowana rozprawa doktorska. I tak, w rozdziale drugim opisano podstawowe zagadnienia związane z eksploatacją i diagnostyką obiektów technicznych. W rozdziale trzecim opisano zagadnienia dotyczące ryzyka technicznego, towarzyszącego eksploatowanym obiektom technicznym. Rozdział czwarty poświęcony jest inżynierii wymagań, dziedzinie która odniosła niebywały sukces w obszarze inżynierii oprogramowania, a której to metody i techniki z powodzeniem stosowane są w innych dziedzinach nauki. Zagadnienia sztucznej inteligencji, w szczególności metody i techniki bezpośrednio związane z tematyką rozprawy doktorskiej zostały opisane w rozdziale piątym. Rozdział szósty poświęcony jest opisowi zaproponowanego podejścia projektowego do problemu opracowywania projektu systemu diagnostycznego z wykorzystaniem zbiorów wymagań. Proces implementacji zaproponowanego podejścia projektowego przedstawiony został w rozdziale siódmym, natomiast w rozdziale ósmym przedstawiono wyniki badań weryfikacyjnych. Rozdział dziewiąty to podsumowanie pracy, w którym zamieszczono stosowne wnioski i nakreślono możliwe kierunki dalszych prac. Na samym końcu pracy umieszczono spis wykorzystanej w tekście literatury oraz spisy rysunków oraz tabel. Rozprawa kończy się streszczeniami pracy w językach polskim i angielskim.

1.4. Podstawowe terminy

W celu uniknięcia niejasności związanych ze stosowanymi w rozprawie pojęciami, poniżej zamieszczono przyjęte definicje najważniejszych ze stosowanych terminów. Definicje pozostałych terminów związanych z diagnostyką techniczną, można znaleźć m.in. w (Cempel i Tomaszewski, 1992; Żółtowski i Ćwik, 1996; Isermann i Balle, 1997; Korbicz i in., 2002).

funkcjonalność – zdolność do dostarczenia funkcji zaspokajających wyznaczone lub zakładane potrzeby, termin ten nie jest tożsamy z użytecznością.

niesprawność – stan obiektu, w którym nie spełnia on chociażby jednego z wymagań dokumentacji technicznej, jest jednak zdolny do wykonywania z ograniczeniem wyznaczanych mu funkcji.

obiekt techniczny – maszyna lub proces techniczny realizujący przewidzianą w procesie projektowo-konstrukcyjnym funkcję celu.

podsystem systemu diagnostycznego – zbiór połączonych i współpracujących ze sobą elementów (czujników, algorytmów itp.), których celem jest generowanie informacji o wystąpieniu lub też nie danego uszkodzenia.

projekt systemu diagnostycznego – lista wyszczególnionych funkcjonalności, które powinny być zaimplementowane w systemie diagnostycznym, wraz z informacją jakie do tego celu powinny zostać zastosowane podsystemy, tj. czujniki, oprogramowanie itp.

reguła diagnostyczna – wskaźnik/relacja będący przesłanką do uznania wystąpienia określonych zdarzeń, np. uszkodzenia obiektu.

stan obiektu – chwilowa wartość parametrów obiektu, pozwalających na opis obiektu w funkcji jego użyteczności.

sygnał diagnostyczny – przebieg dowolnej wielkości fizycznej w czasie, będącej nośnikiem informacji pozwalającej na orzeczenie o stanie technicznym obiektu, w praktyce interpretowany jest jako wynik interakcji obiektu diagnozowanego z otoczeniem.

system diagnostyczny – zbiór połączonych i współpracujących ze sobą elementów (czujników, algorytmów itp.), których celem jest generowanie informacji o stanie technicznym obiektu, z którym to dany system współpracuje.

uszkodzenie – dowolne zdarzenie eksploatacyjne, po wystąpieniu którego, następuje utrata zdolności obiektu do wypełniania wymaganych funkcji.

wymaganie – sformalizowany opis funkcji/cechy, którą powinien realizować lub posiadać projektowany środek techniczny.

Rozdział 2

Diagnozowanie stanu maszyn i procesów

W rozdziale opisano najważniejsze zagadnienia związane z diagnostyką techniczną. W szczególności omówiono proces eksploatacji maszyn i procesów technicznych oraz ich diagnostykę. Ponadto omówiono problematykę związaną ze stosowaniem systemów diagnostycznych.

2.1. Proces eksploatacji obiektów technicznych

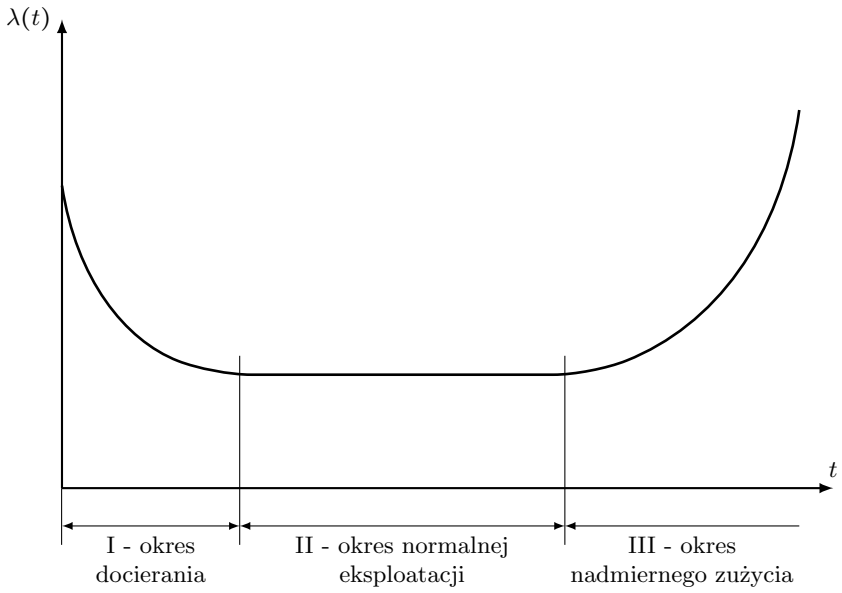
W trakcie eksploatacji obiektów technicznych możliwe jest wystąpienie różnego rodzaju zdarzeń, wpływających na działanie tych obiektów. Wśród nich, wyróżnić można m.in. zaniki zasilania, braki surowców, uszkodzenia, zużycia itp. Uszkodzenia obiektu technicznego, uwidaczniające się w trakcie jego eksploatacji, w ogólnym przypadku mogą być spowodowane różnymi przyczynami. Wyróżnić można (Legutko, 2004; O'Connor i Kleyner, 2011) przede wszystkim przyczyny:

- konstrukcyjne – błędy w projektowaniu i konstruowaniu obiektu,
- produkcyjne – błędy i niedokładności procesów technologicznych lub wady materiałowe występujące w elementach obiektu,
- eksploatacyjne – nieprzestrzeganie zaleconych zasad eksploatacji danego obiektu,
- starzeniowe – naturalne zmiany stanu obiektu, prowadzące do pogorszenia się wytrzymałości i zdolności współdziałania poszczególnych podzespołów i elementów,
- celowe – stosowanie podzespołów o małej trwałości, bądź zniszczenie obiektu przez wrogo nastawione jednostki, np. ataki hakerskie[1].

Natomiast proces zużycia, czyli stopniowe niszczenie elementów obiektu, przebiega (Żółtowski, 1996) pod wpływem czynników fizyko-chemicznych oraz działającego obciążenia w okresie eksploatacji obiektu, przy czym procesy zużywania zachodzą tylko w okresie realizacji przez obiekt techniczny procesów roboczych. Możliwe jest przyjęcie założenia upraszczającego (Korbicz i in., 2002), że zużycie obiektu będzie rozpatrywane jako jeden z rodzai uszkodzenia.

[1]Przykładem może być robak komputerowy Stuxnet, wprowadzony m.in. do systemu sterowania irańskich elektrowni atomowych. Poprzez sterowniki SIMATIC S7-300 oraz S7-400 modyfikował on parametry pracy wybranych sterowników PLC, zainstalowanych w systemie sterowania tych elektrowni (Matrosov i in., 2010; Falliere i in., 2010).

W trakcie eksploatacji obiektu technicznego można zaobserwować (Cholewa i in., 2008; Birolini, 2010) zmienną intensywność uszkodzeń $\lambda(t)$. Przykład typowej krzywej uszkodzeń pokazano na rys. 2.1. Wyróżnić można trzy podstawowe okresy. Pierwszy z nich to tzw. okres docierania. Obiekt techniczny jest tuż po wyprodukowaniu, obserwuje się podwyższoną liczbę uszkodzeń spowodowaną głównie wadami, które nie zostały wychwycone na etapie testowania obiektu oraz naturalnymi procesami, które są wymagane aby wybrane podzespoły zaczęły normalnie ze sobą współpracować. Są to tak zwane wady okresu dziecięcego. Kolejny okres to faza normalnej eksploatacji. Obserwuje się w niej w miarę stałą liczbę uszkodzeń przypadającą na określoną jednostkę czasu. Po tym okresie zauważa się wzrost liczby uszkodzeń, spowodowany starzeniem się obiektu oraz nadmiernym jego zużyciem.



Rys. 2.1: Intensywność uszkodzeń w okresie użytkowania typowego obiektu technicznego

Dla potrzeb przeprowadzania analiz eksploatowanych obiektów technicznych, możliwe jest wprowadzenie terminu niezawodność obiektu $R(t)$, rozumianego jako zdolność obiektu do spełniania stawianych mu wymagań. Przyjmując za miarę niezawodności obiektu technicznego, prawdopodobieństwo jego poprawnego działania w określonym czasie i w określonych warunkach, otrzymuje się wielkość bezwymiarową, przyjmującą wartości z przedziału $< 0; 1 >$. Wartość 1 odpowiada obiektowi całkowicie niezawodnemu, natomiast wartość 0 odpowiada obiektowi całkowicie zawodnemu. Zatem niezawodność obiektu może być wyrażona zależnością:

$$R(t) = p\{s_\tau = s;\ 0 < \tau < t\}, \tag{2.1}$$

gdzie:

- p – prawdopodobieństwo trwania obiektu w stanie s w każdej chwili τ z przedziału od 0 do t,
- s_τ – stan obiektu w dowolnej chwili τ z przedziału od 0 do t,
- s – stan poprawnego działania obiektu.

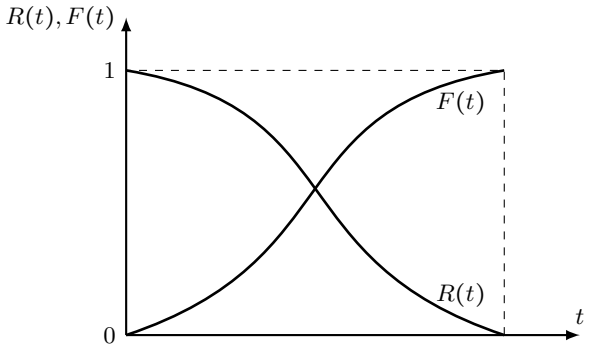
Zapisując funkcję niezawodności jako:

$$R(t) = p\{T \geq t\}, \tag{2.2}$$

gdzie T jest czasem zdatności obiektu, możliwe jest zdefiniowanie funkcji zawodności, wyrażonej jako:

$$F(t) = p\{T < t\} = 1 - R(t), \tag{2.3}$$

która jest dystrybuantą zmiennej losowej T. Zależność pomiędzy funkcjami $R(t)$ oraz $F(t)$ można przedstawić w postaci wykresu pokazanego na rys. 2.2.



Rys. 2.2: Funkcja niezawodności $R(t)$ oraz zawodności $F(t)$ obiektu technicznego

Dla potrzeb opisu eksploatowanych obiektów technicznych możliwe jest zdefiniowanie (Rausand i Høyland, 2004) pewnych wielkości charakterystycznych, którymi są m.in.:

- średni czas pomiędzy uszkodzeniami MTBF (ang. *Mean Time Between Failures*) – średni oczekiwany czas pomiędzy kolejnymi uszkodzeniami danego obiektu, przy założeniu że urządzenie przechodzi naprzemienne okresy uszkodzeń i napraw,
- średni czas naprawy MTTR (ang. *Mean Time To Repair*) – średni czas od momentu wystąpienia uszkodzenia do naprawy uszkodzonego obiektu,
- średni czas do wystąpienia uszkodzenia MTTF (ang. *Mean Time To Failure*) – czas jaki upłynie od momentu naprawy obiektu do wystąpienia kolejnego uszkodzenia.

Projektując maszyny i procesy techniczne powinno dążyć się do maksymalizacji czasu ich użytkowania, skrócenia czasu potrzebnego na ewentualną naprawę, przy jednoczesnym zachowaniu wszystkich funkcji produktu. W ostatnich latach daje się jednak zauważyć (Slade, 2009; Ryś, 2015), pewną tendencję wśród producentów, której celem jest takie projektowanie obiektów technicznych, aby miały one ograniczony czas użytecznego działania. Zazwyczaj po ściśle określonym przedziale czasu (zazwyczaj tuż po upływie gwarancji), obiekty te stają się niesprawne a ich naprawa często nieopłacalna (koszt zbliżony do kupna nowego urządzenia) lub niemożliwa. Takie postępowanie nazywane jest w literaturze planowanym starzeniem lub planowanym postarzaniem produktu (ang. *planned obsolescence*). W głównej mierze działania te podyktowane są przyjętą strategią biznesową, której celem jest maksymalizacja zysków. Jako przykłady produktów poddanych tej strategii wymienić można m.in. żarówki[2] oraz kartridże do drukarek[3]. Więcej przykładów można znaleźć m.in. w (Ryś, 2015).

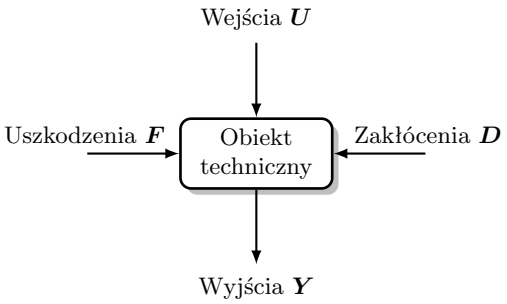
[2]Działający od 23 grudnia 1924 roku aż do roku 1939 kartel Phoebus, w ramach którego firmy m.in. Osram, Philips oraz General Electric podjęły działania, których celem było ograniczenie żywotności typowych żarówek do poziomu ok. 1000h, przy jednoczesnym wzroście ich cen. Działania te określane są często jako tzw. spisek żarówkowy (Krajewski, 2014).

[3]Niektóre firmy np. Epson stosowały czujniki, które zliczały ilość wydrukowanych stron. Po osiągnięciu założonej liczby wydrukowanych stron, programowo blokowana była możliwość dalszego druku i przekazywana informacja o konieczności wymiany kartridża na nowy, niezależnie od ilości tuszu w nim pozostającego (Ryś, 2015).

2.2. Stan techniczny obiektu

Dla potrzeb procesu diagnozowania, każdy obiekt techniczny, może być rozpatrywany tak jak to pokazano na rys. 2.3. Dla tak przyjętego modelu obiektu, możliwe jest wyróżnienie następujących zbiorów sygnałów:

- wejścia **U** – zewnętrzne oddziaływania na obiekt, w postaci znanych sygnałów wymuszających, sterujących itp.,
- wyjścia **Y** – oddziaływania rozpatrywanego obiektu na otoczenie, np. zmienne procesowe,
- zakłócenia **D** – zewnętrzne oddziaływania na obiekt, o nieznanych wartościach,
- uszkodzenia **F** – zewnętrzne i/lub wewnętrzne (od wybranych podzespołów) oddziaływania na obiekt.



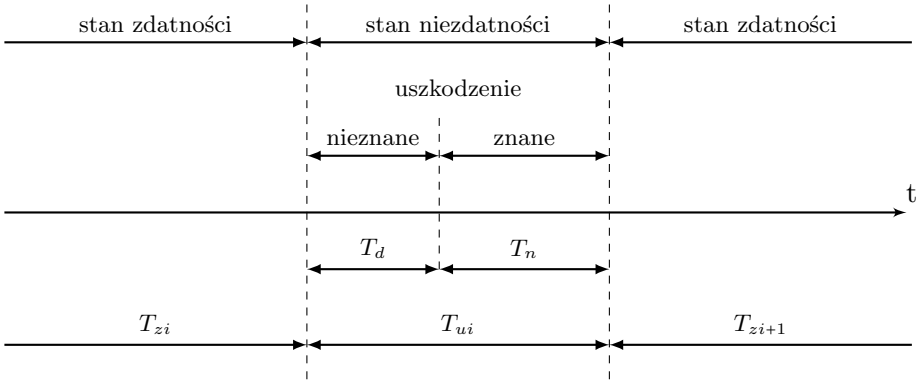
Rys. 2.3: Model obiektu diagnozowania (Korbicz i in., 2002)

Obiekt techniczny poddany działaniu wyżej wymienionych sygnałów, w dowolnej chwili czasu t, znajduje się w jednym z tzw. stanów obiektu. W zależności od dziedziny techniki, w literaturze można znaleźć różne definicje terminu *stan obiektu*. Dla potrzeb diagnostyki maszyn i procesów przyjmuje się (Korbicz i in., 2002), że stan techniczny obiektu opisany jest zbiorem chwilowych wartości pewnych parametrów, wykorzystywanych do jednoznacznego opisu tego obiektu (tzw. współrzędne stanu). W wielu przypadkach (Korbicz i in., 2002) nie istnieje potrzeba (lub jest to technicznie niemożliwe), rozpatrywania dokładnych wartości współrzędnych stanu. Wystarcza jedynie ograniczenie się do klas stanu, np. *sprawność maszyny poniżej wartości nominalnej*. W najprostszym przypadku możliwe jest rozróżnienie jedynie dwóch stanów technicznych, tzw. dwustanowy sens niezawodnościowy, tj. stan zdatności oraz stan niezdatności obiektu technicznego. Stan zdatności rozumiany jest jako stan obiektu, w którym jest on zdolny do działania zgodnie z określoną w procesie projektowo-konstrukcyjnym funkcją celu. Stan obiektu eksploatacji, który wyklucza jego poprawne działanie jest natomiast interpretowany jako stan niezdatności.

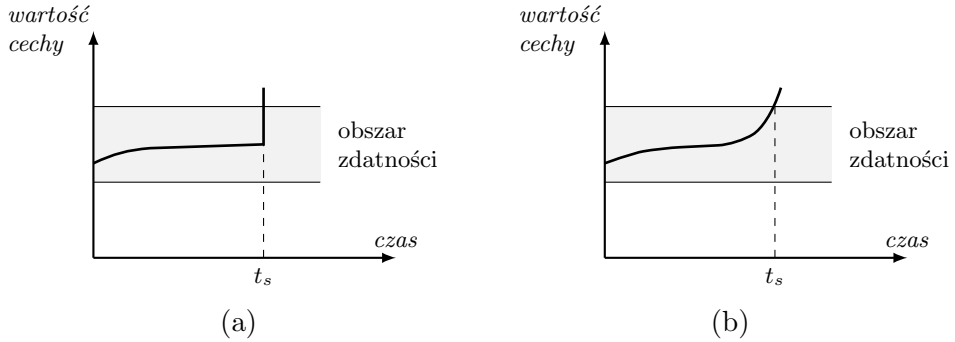
Zmiana stanu technicznego następuje w wyniku wystąpienia określonych zdarzeń eksploatacyjnych (rys. 2.4). Dla potrzeb diagnostyki technicznej przyjmuje się założenie (Korbicz i in., 2002), że wszystkie one są rozpatrywane jako uszkodzenia. Przy czym możliwe jest rozpatrywanie uszkodzeń jako zdarzeń nagłych oraz naturalnych (rys. 2.5). Można zatem stwierdzić, że stan techniczny obiektu $z(t)$ jest funkcją uszkodzeń:

$$z(t) = z[f(t)] \tag{2.4}$$

i jest on zmienny w czasie. Uwzględniając fakt, że dla dowolnego obiektu technicznego możliwe



Rys. 2.4: Schemat wystąpienia uszkodzenia w trakcie eksploatacji obiektu, T_d – czas diagnozowania, T_n – czas naprawy, T_{ui} – czas trwania i-tego uszkodzenia, T_{zi}, T_{zi+1} – czas poprawnej pracy pomiędzy uszkodzeniami



Rys. 2.5: Przebieg uszkodzenia nagłego (a) oraz naturalnego (b), t_s – czas detekcji uszkodzenia

jest dysponowanie pewnym skończonym zbiorem znanych uszkodzeń:

$$F = \{f_k : k = 1, 2, \ldots K\} \tag{2.5}$$

oraz, że możliwe jest jednoczesne występowanie więcej niż jednego uszkodzenia, możliwe jest rozpatrywanie stanu obiektu jako zależności wyrażonej poprzez stany poszczególnych uszkodzeń:

$$Z = \{z(f_1), z(f_2), z(f_k) \ldots z(f_K)\}, \tag{2.6}$$

przy czym stan uszkodzenia $z(f_k)$, definiowany jest jako:

$$z(f_k) = \begin{cases} 0 & \text{bez uszkodzenia } f_k \\ 1 & \text{z uszkodzeniem } f_k. \end{cases} \tag{2.7}$$

Poprzez przyjęcie takiego sposobu postępowania, liczba wszystkich możliwych stanów technicznych rozpatrywanego obiektu jest równa:

$$card(Z) = 2^K. \tag{2.8}$$

Przejście z jednego stanu technicznego w drugi, może odbywać się w sposób płynny z uwagi na występującą w trakcie eksploatacji obiektu propagację uszkodzeń (Gabbar, 2007b; Psiuk, 2001; Gao i in., 2008).

Uwzględniając aktualne dane o obiekcie, proces rozpoznawania stanu technicznego może

być (Korbicz i in., 2002) rozpatrywany jako:

- diagnozowanie – celem, którego jest ustalenie aktualnego stanu obiektu, jest to podstawowe zadanie diagnostyczne,
- genezowanie – celem, którego jest ustalenie wcześniejszych stanów obiektu,
- prognozowanie – celem, którego jest ustalenie przyszłych stanów obiektu.

Wyróżnia się (Isermann i Balle, 1997) trzy podstawowe etapy procesu oceny aktualnego stanu technicznego. Są nimi detekcja, lokalizacja oraz identyfikacja uszkodzenia. Detekcja uszkodzenia (ang. *fault detection*) to wykrycie/zaobserwowanie wystąpienia pewnego uszkodzenia w obiekcie, w pewnej chwili czasu. Po detekcji uszkodzenia następuje etap lokalizacji uszkodzenia (ang. *fault isolation*), czyli określenie rodzaju, miejsca i czasu w którym wystąpiło dane uszkodzenie. Ostatnim etapem jest identyfikacja uszkodzenia (ang. *fault identification*), polegająca na określeniu rozmiaru oraz charakteru zmienności w czasie danego uszkodzenia. W wielu przypadkach (Korbicz i in., 2002) nie jest możliwe pełne określenie stanu technicznego, tj. przeprowadzenie wszystkich wymienionych etapów procesu. Zazwyczaj ogranicza się on zatem do detekcji i lokalizacji uszkodzenia, co w większości przypadków jest wystarczające.

Dla potrzeb przeprowadzania procesu wnioskowania o stanie technicznym obiektu, konieczne jest pozyskiwanie sygnałów diagnostycznych. Przyjmuje się (Korbicz i in., 2002), że sygnał diagnostyczny jest przebiegiem dowolnej wielkości fizycznej będącej nośnikiem informacji o stanie technicznym rozpatrywanego obiektu. Zazwyczaj tylko jedynie wybrane cechy danego sygnału (np. wartość skuteczna, wartość maksymalna itp.), pozwalają na prawidłową i skuteczną ocenę stanu technicznego. Jako sygnały diagnostyczne mogą być (Kościelny, 2001; Korbicz i in., 2002) wykorzystywane m.in.:

- zmienne procesowe i procesy resztkowe,
- residua generowane na podstawie modeli obiektów,
- binarne lub wielowartościowe sygnały wyznaczone w wyniku oceny wartości residuów.

Na podstawie oceny wartości sygnałów diagnostycznych, możliwe jest stwierdzenie czy wyznaczona wartość sygnału może być uznana za symptom wystąpienia określonego uszkodzenia. W wielu przypadkach wystąpienie określonego uszkodzenia, połączone jest z pojawieniem się więcej niż jednego symptomu. Zestawienie symptomów związanych z wystąpieniem danego uszkodzenia nazywane jest sygnaturą uszkodzenia. Uszkodzenia oraz ich sygnatury mogą być zestawione w tzw. binarną macierz diagnostyczną (tab. 2.1). Dwa lub więcej uszkodzeń bę-

Tab. 2.1: Przykład binarnej macierzy diagnostycznej

S/F	f_1	f_2	f_3	f_4	f_5
s_1	1	1	1	1	1
s_2		1	1	1	1
s_3	1			1	1
s_4	1	1		1	1

dzie nierozróżnialnych względem siebie, jeżeli będą posiadały identyczną sygnaturę uszkodzenia (np. uszkodzenia f_4 oraz f_5 z tab. 2.1). W takich przypadkach, konieczne jest rozszerzenie zbioru dostępnych sygnałów diagnostycznych o dodatkowe sygnały, które pozwolą na rozróżnienie tych uszkodzeń.

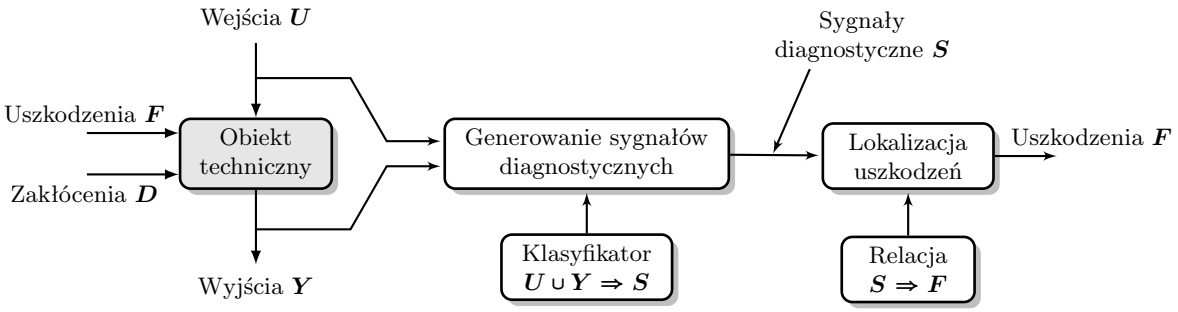
2.3. Metody i techniki diagnostyczne

Proces diagnozowania stanu maszyn i procesów jest zadaniem niezwykle trudnym i skomplikowanym. Wyróżnić można dwa podstawowe podejścia do rozpatrywanego zagadnienia, tj. diagnostykę symptomową oraz wspartą modelem.

2.3.1 Diagnostyka symptomowa

Idea diagnostyki symptomowej (tzw. bezpośredniej) została pokazana na rys. 2.6. W podejściu tym, wnioskowanie o stanie technicznym obiektu odbywa się wyłącznie w oparciu o zarejestrowane sygnały (głównie zmienne procesowe i procesy resztkowe) podczas działania rozpatrywanego obiektu (Cempel, 1982). W takim przypadku detekcja uszkodzeń, może odbywać się (Kościelny, 2001; Korbicz i in., 2002) na podstawie:

- kontroli ograniczeń wartości cech zmiennych procesowych,
- kontroli wartości cech procesów resztkowych,
- analizy zarejestrowanych sygnałów,
- poprzez wykorzystanie prostych związków pomiędzy zmiennymi procesowymi, procesami resztkowymi.



Rys. 2.6: Idea diagnostyki symptomowej (Korbicz i in., 2002)

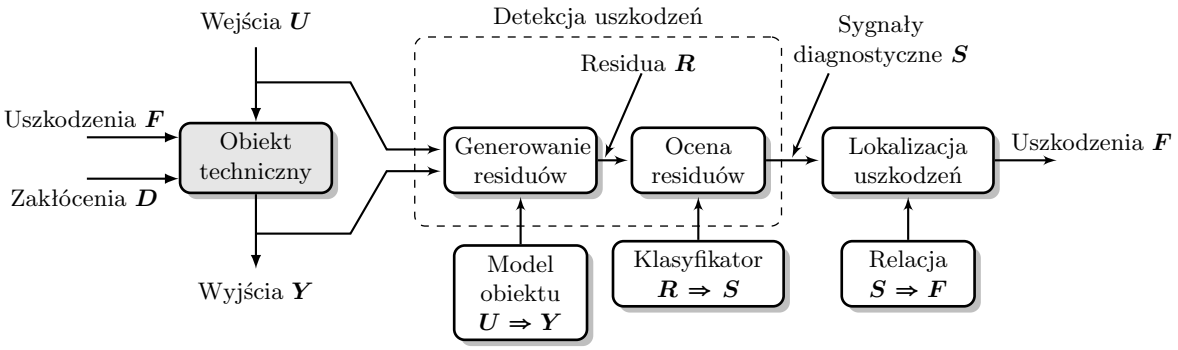
W wyniku przeprowadzenia fazy detekcji uszkodzeń, generowane są sygnały diagnostyczne. Na podstawie znanych relacji pomiędzy sygnałami diagnostycznymi a uszkodzeniami ($S \Rightarrow F$), dokonywany jest proces lokalizacji uszkodzeń a zatem wyznaczany jest stan techniczny rozpatrywanego obiektu.

Proces detekcji uszkodzeń może być przeprowadzany na podstawie rejestracji i analizy procesów resztkowych w tym m.in.: sygnałów drganiowych (Kucharski, 2002; Morel, 1992; Adams, 2010), akustycznych (Cempel, 1989; Lipowczan, 1987), cieplnych (Oliferuk, 2008; Fidali, 2013), produktów zużycia (Gierek, 2005; Lawrowski, 2008) itp.

2.3.2 Diagnostyka wsparta modelem

Ideę diagnostyki wspartej modelem pokazano na rys. 2.7. W podejściu tym, dla celów detekcji uszkodzeń, wykorzystuje się model obiektu odwzorowujący znane związki pomiędzy sygnałami wejściowymi a wyjściowymi $U \Rightarrow Y$. Model ten, opracowywany jest zazwyczaj dla nominalnego (bez uszkodzeń) stanu pracy obiektu. Poprzez porównanie sygnałów wyznaczony przez model, z sygnałami zarejestrowanymi na rzeczywistym obiekcie, generowane są tzw. residua. W przypadku rozpatrywania obiektu, w którym nie wystąpiły żadne uszkodzenia, wartości residuów

powinny oscylować w pobliżu zera. Wystąpienie w obiekcie pewnego uszkodzenia, powinno natomiast skutkować wzrostem bezwzględnej wartości wybranych residuów, które są interpretowane jako symptomy uszkodzeń. Na ich podstawie, z wykorzystaniem modelu odwzorowującego znane relacje pomiędzy wybranymi sygnałami diagnostycznymi a uszkodzeniami ($S \Rightarrow F$), przeprowadzany jest proces lokalizacji uszkodzenia (Vachtsevanos i in., 2006; Ding, 2008; Korbicz i Kościelny, 2010).



Rys. 2.7: Idea diagnostyki wspartej modelem (Korbicz i in., 2002)

W fazie detekcji uszkodzeń, mogą być stosowane następujące typy modeli obiektów (Korbicz i in., 2002):

- analityczne – opracowywane na podstawie znanych zależności fizycznych (np. równania ruchu, bilanse energetyczne itp.), opisujących związki pomiędzy poszczególnymi wielkościami charakteryzującymi rozpatrywany obiekt techniczny.
- parametryczne – opracowywane w wyniku przeprowadzenia procesu identyfikacji obejmującego (Soderstrom i Stoice, 1997): zgromadzenie danych (czynny lub bierny eksperyment) opisujących działanie danego obiektu technicznego, zdefiniowanie struktury modelu, estymację nieznanych parametrów modelu na podstawie zgromadzonych danych oraz weryfikację opracowanego modelu na podstawie przyjętych kryteriów. Powszechnie stosowane są modele autoregresyjne AR, autoregresyjne z zewnętrznym wejściem ARX, autoregresyjne średniej ruchomej i zewnętrznym wejściem ARMAX itp.
- neuronowe – opracowywane są głównie wtedy, gdy istnieje potrzeba odwzorowywania złożonych obiektów, o silnie nieliniowej charakterystyce. Jedynym problemem pojawiającym się w czasie opracowywania takich modeli jest zebranie odpowiedniego zbioru danych wejściowych oraz dobór odpowiedniej struktury sieci neuronowej (Przystałka, 2009).
- rozmyte – opracowywane są one dla obiektów, dla których nie jest posiadana pełna i dokładna wiedza opisująca sposób ich działania. Wśród najczęściej stosowanych modeli rozmytych wymienić można m.in. modele Mamdaniego oraz Takagi-Sugeno-Kanga.
- hybrydowe – opracowywane są jako połączenie innych modeli np. modele neuronowo-rozmyte, przybliżone sieci neuronowe itp., dzięki zastosowaniu takich modeli, możliwe jest uzyskanie dodatkowych zalet modelu wyjściowego, przy jednoczesnej minimalizacji wad poszczególnych modeli składowych.

2.4. Systemy monitorowania i diagnostyki

Dla potrzeb oceny stanu technicznego rozpatrywanych obiektów technicznych, opracowywane są specjalistyczne systemy informatyczne, których celem jest monitorowanie i/lub diagnostyka

tych obiektów. Przez monitorowanie obiektu technicznego (Barszcz, 2006), rozumie się proces w czasie którego, następuje pomiar wartości wybranych sygnałów dostępnych na rozpatrywanym obiekcie, a następnie ich prezentacja operatorowi obiektu. Natomiast jako diagnostyka obiektu, rozumiany jest kolejny etap w tym podejściu, tj. określenie aktualnego stanu technicznego obiektu na podstawie dostępnych danych pomiarowych oraz wiedzy dziedzinowej. Generalnie rzec biorąc, systemy diagnostyki nie mogą istnieć bez części odpowiedzialnej za monitorowanie, z drugiej strony, w wielu przypadkach zastosowanie jedynie systemów monitorowania dla wybranych klas obiektów, może nie być wystarczające. Rodzaj zastosowanych systemów uzależniony jest od wielu czynników w tym m.in. od możliwości technicznych, dostępnych środków finansowych, wymagań narzuconych przez zewnętrzne czynniki w tym przez warunki biznesowe, przepisy, normy itp.

Systemy monitorowania i diagnostyki opracowywane mogą być, jako rozwiązania pracujące w trybie off-line oraz on-line. Systemy off-line realizowane są jako przenośne rejestratory i analizatory sygnałów, z wykorzystaniem których dokonywany jest okresowy pomiar sygnałów w pewnych ustalonych punktach pomiarowych obiektu technicznego. Zaletą takich rozwiązań jest (Barszcz, 2006) ich niski koszt, natomiast wadami duża niepewność pomiaru oraz mała powtarzalność. Systemy on-line przeznaczone są natomiast do pracy ciągłej. Instalowane są bezpośrednio na nadzorowanym obiekcie technicznym a wyniku diagnostyki prezentowane są operatorowi znajdującemu się bezpośrednio przy obiekcie, bądź też przesyłanie do centrum diagnostycznego.

Wśród typowych podzespołów systemów monitorowania i diagnostyki maszyn i procesów wymienić można m.in. (Cholewa i Psiuk, 2005; Barszcz, 2006; Gabbar, 2007a):

- czujniki pomiarowe – zadaniem, których jest przetworzenie wybranych wielkości fizycznych, zazwyczaj nieelektrycznych, np. drgania, temperatura, przepływ, ciśnienie itp. na wielkości elektryczne, które są następnie rejestrowane przez stosowne układy pomiarowe,
- jednostki akwizycji danych – tzw. moduły/karty DAQ (ang. *Data Acquisition Unit/Card*) odpowiedzialne za rejestrację sygnałów, w zależności od poziomu zaawansowania tych układów umożliwiają one m.in. próbkowanie sygnałów, kontrolę toru pomiarowego, wstępne przetwarzanie sygnałów celem wyznaczenia np. prostych cech statystycznych, kontrolę progów alarmowych oraz buforowanie danych,
- serwery danych – przeznaczone do gromadzenia i przechowywania danych pomiarowych oraz danych wyznaczonych w wyniku procesu oceny stanu technicznego obiektu, zgromadzone dane są udostępniane modułowi diagnozowania oraz stacjom operatorskim i centrom diagnostycznym,
- moduł diagnozowania – podzespół, w którym na podstawie wartości cech wybranych sygnałów oraz zgromadzonej wiedzy diagnostycznej dokonywany jest proces oceny stanu technicznego rozpatrywanego obiektu technicznego,
- moduł prezentacji danych – podzespół systemu diagnostycznego przeznaczony do prezentacji danych zarejestrowanych na obiekcie lub wyznaczonych w wyniku procesu diagnozowania, zazwyczaj dane prezentowane są w formie wykresów, tabel oraz ekranów synoptycznych,
- stację operatorską – zazwyczaj zlokalizowana jest tuż przy obiekcie technicznym, za jej pośrednictwem możliwy jest dostęp do systemu monitorowania i diagnostyki rozpatrywanego obiektu technicznego,

- centrum diagnostyczne – podzespół systemu, stosowany przy diagnozowaniu bardzo złożonych lub rozproszonych obiektów technicznych, gromadzi on dane z wielu odrębnych systemów monitorowania stosowanych dla poszczególnych obiektów składowych.

W ostatnich latach można zauważyć wzrost złożoności typowych systemów monitorowania i diagnostyki obiektów technicznych. Bardzo często są one łączone, lub posiadają pewne funkcjonalności systemów odpowiedzialnych za sterowanie obiektami technicznymi. Przykładami mogą być systemy klasy SCADA (ang. *Supervisory Control and Data Acquisition*) oraz DCS (ang. *Distributed Control Systems*). Realizują one szereg zadań, począwszy od monitorowania obiektu i prezentacji wyników za pośrednictwem ekranów synoptycznych, poprzez odpowiednie sterowanie tymi obiektami, alarmowanie o występujących nieprawidłowościach itp.

Jako przykłady systemów diagnostycznych opracowanych w kraju wymienić można m.in. system DT200-1 przeznaczony do monitorowania maszyn wirnikowych (wdrożony w Elektrowni Kozienice), system AMandD przeznaczony do monitorowania i diagnostyki instalacji przemysłowych, system Vibex przeznaczony do monitorowania maszyn wirnikowych oraz system FAMAC VIBRO przeznaczony do monitorowania maszyn stosowanych w przemyśle górniczym. Wśród zagranicznych systemów wymienić należy przede wszystkim, zintegrowane systemy klasy SCADA oraz DCS, takie jak: Ovation (producent: Emerson), System 800xA (ABB), SIPLUS CMS System (Siemens), Experion PKS (Honeywell), AMODIS (Alstom Power).

2.5. Podsumowanie

Z eksploatacją obiektów technicznych wiąże się niebezpieczeństwo wystąpienia różnego rodzaju zdarzeń, zbiorczo nazywanych uszkodzeniami, które mogą mieć niekorzystny wpływ na obiekt techniczny oraz na jego najbliższe otoczenie. Dlatego też, proces oceny ich stanu technicznego, a w konsekwencji podejmowanie odpowiednich decyzji eksploatacyjnych, ma ogromne znaczenie z punktu widzenia prawidłowej pracy tych obiektów. Zastosowanie odpowiednich metod i technik diagnostycznych uwarunkowane jest wieloma różnymi czynnikami, jednakże w większości przypadków sprowadzają się one do ograniczonych możliwości technicznych oraz niewystarczających zasobów finansowych. Proces diagnozowania stanu technicznego, przeprowadzany może być z wykorzystaniem odpowiednio opracowanych systemów monitorowania i diagnostyki. Klasa tych systemów, w dużej mierze, zależy od rodzaju obiektu technicznego dla którego są one stosowane. W ostatnich latach można zauważyć znaczący wzrost rozwiązań opartych na scentralizowanych systemach diagnostycznych, do których podłączonych jest wiele obiektów technicznych, dzięki czemu możliwe jest jednoczesne zarządzanie i monitorowanie wieloma obiektami technicznymi.

Rozdział 3

Ryzyko techniczne w eksploatacji maszyn i procesów

W rozdziale opisano problematykę ryzyka oraz jego miejsce we współczesnym świecie. Nakreślono praktyczne aspekty ryzyka technicznego, metody jego oceny oraz sposoby zarządzania ryzykiem dla potrzeb diagnostyki maszyn i procesów.

3.1. Ogólna problematyka ryzyka

W zależności od dziedziny, w ramach której stosowany jest termin *ryzyko*, może być ono interpretowane m.in. jako:

- niepewność co do osiągnięcia założonych celów (PN-EN ISO 12100:2012),
- niebezpieczeństwo powstania szkody (Rusin, 2008),
- niepewność co do wyniku zamierzonego działania (Dietrych, 1978),
- możliwość osiągnięcia straty (Kaplan i Garrick, 1981),
- niepożądana sytuacja lub okoliczność, która może mieć negatywny wpływ na realizowany projekt (ECSS-M-00-03A:2010).

Uogólniając przedstawione powyżej sformułowania, można przyjąć że ryzyko jest pewną oceną (często subiektywną) zagrożenia wynikającego z możliwych, niekorzystnych konsekwencji po podjęciu (lub ich braku) pewnych decyzji. Należy zauważyć (Adams, 2002; Lupton, 2013), że ryzyko jest nieodłącznie związane z każdą decyzją, z każdym krokiem jaki wykonuje człowiek. W wielu przypadkach jest ono jednak tak małe, że jest pomijalne. Dokładniej rzecz ujmując, to nie zwracamy uwagi na to, że każda nasza decyzja w życiu codziennym jest obarczona jakimś ryzykiem. W większości przypadków nie powoduje to żadnych implikacji, jednakże w niektórych sytuacjach takie postępowanie może doprowadzić do przykrych konsekwencji, które można byłoby w prosty sposób uniknąć. Bardzo ciekawą koncepcją, powiązaną z takimi działaniami, jest tzw. teoria czarnego łabędzia (ang. *black swan theory*) (Taleb, 2007). Opisuje ona zdarzenia z pozoru nieprzewidywalne, jednakże po ich wystąpieniu mające olbrzymie negatywne konsekwencje i które w zasadzie po fakcie, można w bardzo prosty sposób wytłumaczyć i uznać za nieuniknione.

Przyjmuje się, że człowiek w swoim postępowaniu powinien starać się minimalizować ryzyko i postępować zgodnie z zasadą ostrożności (ang. *precautionary principle*), mówiącą (O'Riordan i Cameron, 1994) że lepiej zaniechać pewnych działań obarczonych znacznym ryzykiem, niż później ponosić kosztowne konsekwencje w przypadku niepowodzenia. Jednocześnie pojawiają się także głosy m.in. (Deutsch, 2011), twierdzące że to właśnie ponoszenie ryzyka jest impulsem do rozwoju technologicznego świata. Wydaje się, że podejście takie z powodzeniem (przy zachowaniu pewnych środków ostrożności) może być stosowane w wielu przypadkach. Jednakże tam gdzie ewentualny, niekorzystny wpływ skutków poniesionego ryzyka, będzie niósł ogromne zagrożenia dla środowiska oraz zdrowia i życia ludzkiego, takie postępowanie będzie niewskazane.

Ryzyko w różnym stopniu było, jest i będzie związane z wszystkimi dziedzinami ludzkiego życia. Problematyka ryzyka związanego z eksploatowanymi maszynami i procesami technicznymi, jest szczególnie głośno poruszana, w przypadku wystąpienia zdarzeń katastroficznych, o szerokim spektrum negatywnego oddziaływania. Jako przykłady można wymienić m.in. katastrofy w elektrowniach atomowych w Czarnobylu[1], Three Mile Island[2], wybuch na platformie wiertniczej Piper Alpha[3] itd. Pojawiają sie wtedy pytania, czy aby na pewno eksploatacja takich obiektów, nie niesie ze sobą nadmiernego ryzyka i w jaki sposób można ograniczyć ewentualne zagrożenia. Zasadniczo odpowiedź na takie pytanie nie jest prosta i zależy od wielu czynników. Niemniej jednak, dokładna analiza sytuacji niebezpiecznych i ocena związanego z nimi ryzyka, jest podstawą do formułowania stosownych wniosków, a wielu przypadków do podjęcia decyzji o dopuszczeniu lub nie, danego obiektu do użytkowania.

3.2. Ryzyko techniczne

Proces eksploatacji maszyn i procesów technicznych pociąga za sobą możliwość wystąpienia zdarzeń, polegających na zadziałaniu obiektu niezgodnie z jego podstawową funkcją celu. Zdarzenia te, mogą w różny sposób oddziaływać na rozpatrywany obiekt techniczny jak również na jego najbliższe otoczenie, w tym na operatora danego obiektu technicznego. Dążąc do ograniczenia liczby zdarzeń niepożądanych, możliwe jest rozpatrywanie tego procesu jako zadania polegającego na minimalizacji ryzyka.

Dla dowolnego obiektu technicznego O_t, ryzyko techniczne wystąpienia i-tego zdarzenia, może być opisane zależnością:

$$R_{ti} = P_i c_i, \tag{3.1}$$

gdzie P_i jest prawdopodobieństwem wystąpienia niepożądanego zdarzenia i, natomiast c_i jest możliwym kosztem/stratą związaną z wystąpieniem tego zdarzenia. Uwzględniając, że w analizowanym obiekcie może wystąpić n zdarzeń obarczonych mierzalnym ryzykiem, to ryzyko całkowite wyrażone jest zależnością:

$$R_t = \sum_{i=1}^{n} P_i c_i. \tag{3.2}$$

Z przyczyn czysto praktycznych, prawdopodobieństwo P_i wystąpienia i-tego zdarzenia, może być zastąpione częstością wystąpienia tego zdarzenia w określonym przedziale czasu. Ryzyko

[1]Rok 1986, 68 bezpośrednich ofiar śmiertelnych, ewakuacja ok. 350 000 mieszkańców, skażenie promieniotwórcze środowiska na powierzchni ok. 150 000 km2, straty materialne liczone w setkach miliardów dolarów, koszty katastrofy ponoszone są do dnia dzisiejszego (IAEA, 2006).

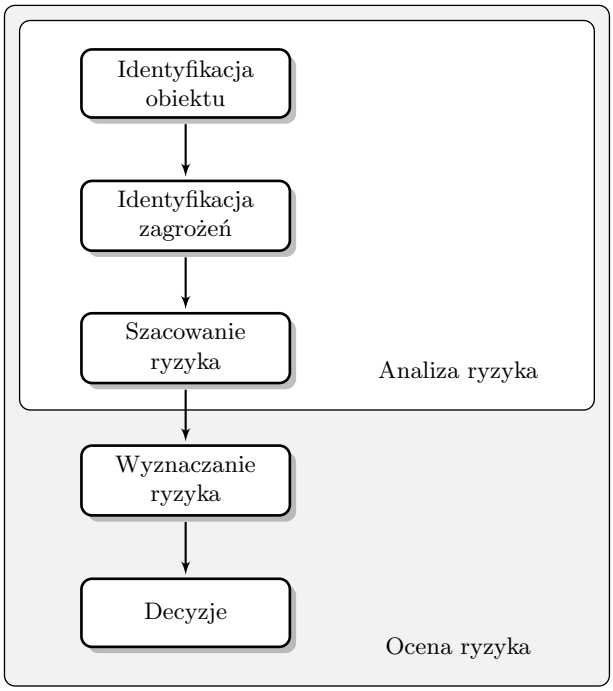
[2]Rok 1979, brak bezpośrednich ofiar śmiertelnych, spadek poparcia społeczeństwa amerykańskiego dla idei użytkowania elektrowni atomowych, wstrzymanie na ok. 30 lat budowy nowych elektrowni atomowych (Fischer, 1981).

[3]Rok 1988, 167 ofiar śmiertelnych, całkowite zniszczenie platformy wiertniczej (Paté-Cornell, 1993).

techniczne wyznaczane jest dla maszyn i procesów technicznych w określonym przedziale czasu. Z uwagi na to, że prawdopodobieństwo P_i wystąpienia i-tego zdarzenia jest uzależnione od aktualnego stanu technicznego rozpatrywanego obiektu oraz przyjmując że proces jego eksploatacji odbywa się w zmiennym otoczeniu, możliwe jest zdefiniowanie (Cholewa i in., 2008) tzw. ryzyka dynamicznego:

$$R_t(t) = \sum_{i=1}^{n} P_i(t) c_i(t). \tag{3.3}$$

Zadaniem wymagających odpowiedniej wiedzy i adekwatnego toku postępowania jest proces oceny istniejącego ryzyka (ang. *risk assessment*). Norma PN-EN ISO 12100:2012 wyróżnia (rys. 3.1) kilka podstawowych etapów tego procesu. Pierwszym z nich jest identyfikacja obiektu technicznego. Celem tego etapu jest dokładne poznanie obiektu badań, warunków jego pracy, sposobu obsługi itp. Następnie przeprowadzana jest identyfikacja zagrożeń (ang. *hazard identification*), w czasie której dokonywane jest rozpoznanie zdarzeń niebezpiecznych, które mogą wystąpić w trakcie eksploatacji obiektu technicznego. W ramach tego etapu przygotowywany jest dokładny opis potencjalnych zdarzeń, a w szczególności identyfikowane są ich przyczyny, skutki oraz możliwe zabezpieczenia. Na podstawie zgromadzonych informacji dokonuje się oszacowania ryzyka (ang. *risk estimation*), najczęściej na poziomie jakościowym. Obejmuje ono wyznaczenie częstości wystąpienia zdarzeń niepożądanych oraz wielkości potencjalnych ich skutków. Te trzy etapy nazywane są łącznie analizą ryzyka (ang. *risk analysis*). Na podstawie wyników przeprowadzonej analizy ryzyka, możliwe jest wyznaczenie wielkości ryzyka (ang. *risk evaluation*) oraz podjęcie stosownych decyzji o akceptowalności lub nie występującego ryzyka.



Rys. 3.1: Etapy procesu oceny ryzyka wg normy PN-EN ISO 12100:2012

Proces szacowania ryzyka może być (Kirchsteiger, 1999) przeprowadzony zarówno na poziomie jakościowym jak i ilościowym. Dla potrzeb oceny ryzyka na poziomie jakościowym przyjmuje się pewne umowne wartości jakościowe, opisujące prawdopodobieństwo wystąpienia poszczególnych zdarzeń oraz potencjalne ich skutki. Ich skala oraz znaczenie, zależy w dużej mierze od typu obiektu technicznego, jego najbliższego otoczenia, warunków pracy itp. Skutki wystąpienia

i-tego zdarzenia, są zazwyczaj oceniane przez pryzmat ich wpływu na uszczerbek na zdrowiu ludzkim, straty materialne, środowiskowe itp. W tabelach 3.1 oraz 3.2, na podstawie przeprowadzonej analizy dostępnych danych literaturowych, m.in. (Morgan i in., 1992; Kirchsteiger, 1999; Petrović i in., 2014; Jovanovic i in., 2014) zestawiono przykładowe wartości jakościowe, stosowane dla potrzeb wyznaczania prawdopodobieństwa oraz skutków wystąpienia i-tego zdarzenia niebezpiecznego. Dla potrzeb dokładniejszych analiz, wyznaczone wartości jakościowe mogą być skorelowane z pewnymi przedziałami wartości np. dokładną liczbą awarii w określonym przedziale czasu, konkretnymi sumami pieniężnymi wyrażającymi poniesione straty itp.

Tab. 3.1: Przykład jakościowego opisu prawdopodobieństwa/częstości wystąpienia i-tego zdarzenia

Kategoria	**Objaśnienie**
bardzo duże	zdarzenia występuje przynajmniej raz w roku
duże	zdarzenia występujące często, częstość zdarzeń na poziomie $1 - 10^{-2}$ w skali roku
średnie	zdarzenia występujące rzadko, częstość zdarzeń na poziomie 10^{-2}–10^{-4} w skali roku
małe	zdarzenia występujące bardzo rzadko, częstość zdarzeń na poziomie 10^{-4} – 10^{-6} w skali roku
znikome	zdarzenia prawie niewystępujące lub pojawiające się w bardzo wyjątkowych i specyficznych okolicznościach, częstość zdarzeń $< 10^{-6}$ w skali roku

Tab. 3.2: Przykład jakościowego opisu skutków wystąpienia i-tego zdarzenia

Kategoria	**Objaśnienie**
katastrofalne	uszkodzenie całego obiektu technicznego, długi przestój do czasu ponownego uruchomienia, duże straty finansowe, znaczne skażenie środowiska, ofiary śmiertelnie wśród personelu oraz ludności cywilnej
duże	uszkodzenie całego obiektu technicznego lub najważniejszych jego podzespołów, znaczny przestój spowodowany koniecznymi naprawami, skażenie środowiska, poszkodowany personel obsługujący obiekt techniczny
średnie	krótkotrwałe wyłączenie obiektu z eksploatacji, spowodowane uszkodzeniem ważniejszych podzespołów, średnie straty finansowe, drobne skażenie środowiska, poszkodowany personel obsługujący obiekt techniczny
małe	niewielkie uszkodzenia obiektu, większość z nich można szybko naprawić, drobne straty finansowe, brak skażenia środowiska, brak poszkodowanego personelu obsługującego obiekt techniczny
nieistotne	drobne skutki niemające żadnego istotnego wpływu na działania obiektu technicznego oraz jego otoczenie

Ryzyko techniczne R_{ti}, opisane zależnością 3.1 szacowane jest subiektywnie, na podstawie wyznaczonych wartości jakościowych prawdopodobieństwa i skutków wystąpienia i-tego zdarzenia. Wyniki oszacowanego ryzyka, można przedstawić w postaci tzw. matrycy ryzyka (rys. 3.2), gdzie poszczególnym elementom P_i, c_i przyporządkowany jest określony poziom ryzyka. W tabeli 3.3 zamieszczono przykład możliwych poziomów ryzyka, opracowany na podstawie analizy dostępnych danych literaturowych m.in. (Morgan i in., 1992; Kirchsteiger, 1999; Jovanovic i in., 2014).

Szacowanie ryzyka na poziomie ilościowym, pozwala na wyznaczenie dokładnych wartości ryzyka (na podstawie zależności 3.1–3.3), zgodnie z przyjętą jego miarą. Najczęściej miarą ryzyka

		Skutki/koszty c_i				
		nieistotne	małe	średnie	duże	katastrofalne
Prawdopodobieństwo P_i	bardzo duże	RT	RN	RN	RNA	RNA
	duże	RT	RT	RN	RN	RNA
	średnie	RT	RT	RT	RN	RN
	małe	RA	RT	RT	RT	RN
	znikome	RA	RA	RT	RT	RT

Rys. 3.2: Przykładowa matryca ryzyka. Oznaczenie ryzyka zgodne z tabelą 3.3

Tab. 3.3: Przykład jakościowego opisu ryzyka związanego z wystąpieniem i-tego zdarzenia

Kategoria	**Objaśnienie**
ryzyko nieakceptowalne (RNA)	ryzyko niedozwolone, nie można go tolerować z uwagi na dużą częstość zdarzeń awaryjnych i ich poważne skutki
ryzyko niepożądane (RN)	ryzyko niepożądane, należy je redukować
ryzyko tolerowane (RT)	ryzyko jest dopuszczalne, jeśli koszt jego redukcji jest nieproporcjonalnie duży względem spodziewanych efektów, to wskazane jest zaniechanie dalszej minimalizacji ryzyka
ryzyko akceptowalne (RA)	poziom ryzyka jest akceptowalny i nie są wymagane żadne dodatkowe działania zapobiegawcze

są straty finansowe oraz liczba ofiar śmiertelnych (ang. *potential loss of life*). Ciekawe przedstawienie możliwych miar ryzyka można znaleźć m.in. w pracy (Jonkman i in., 2003). Rozważając jako miarę ryzyka straty finansowe, możliwe jest posłużenie się terminologią stosowaną przy ocenie ryzyka finansowego, w którym (Christoffersen, 2011; McNeil i in., 2015) wyznacza się tzw. oczekiwaną stratę na pojedynczym zdarzeniu (ang. *Single Loss Expectancy*), opisaną jako:

$$SLE = AV \cdot EF + ELI, \tag{3.4}$$

gdzie: AV – wartość środka technicznego (ang. *Asset Value*), EF – współczynnik możliwych strat (ang. *Exposure Factor*), ELI – spodziewane straty w dochodach (ang. *Expected Loss of Income*). Na jej podstawie możliwe jest wyznaczenie oczekiwanej straty rocznej (ang. *Annual Loss Expectancy*), wyznaczanej jako:

$$ALE = ARO \cdot SLE, \tag{3.5}$$

gdzie: ARO – roczna częstotliwość wystąpienia (ang. *Annual Rate of Occurrence*). Takie podejście pozwala na uwzględnienie w szacowanym ryzyku, zarówno kosztów związanych z naprawą uszkodzonego obiektu (koszty nowych podzespołów oraz koszty robocizny), jak również przychodów finansowych utraconych w wyniku czasowego wyłączenia rozpatrywanego obiektu z procesu eksploatacji.

3.3. Wybrane metody oceny ryzyka technicznego

Proces oceny istniejącego lub potencjalnego ryzyka jest zadaniem niezwykle ważnym. Przyjmuje się (Kirchsteiger, 1999; Khan i Haddara, 2003), że ocena ryzyka związana jest z odpowiedzą na trzy podstawowe pytania, tj.:

- Jakie niebezpieczne zdarzenie może wystąpić?
- Jak bardzo możliwe (jak częste) jest wystąpienie danego zdarzenia?
- Jakie mogą być możliwe konsekwencje wystąpienia danego zdarzenia?

Aby proces oceny ryzyka przeprowadzony był z najwyższą starannością i bez pominięcia ważnych aspektów ryzyka, wskazane jest użycie jednej z wielu metod przeznaczonych do oceny ryzyka. Ciekawe porównanie możliwych metod przedstawiono m.in. w (Raspotnig i Opdahl, 2013), natomiast w pozycji (Tixier i in., 2002) przedstawiono przegląd 62 metod przeznaczonych do analizy, oceny oraz hierarchizacji ryzyka. Poniżej zamieszczono, krótką charakterystykę najważniejszych metod przeznaczonych do analizy i oceny ryzyka.

- PHA – wstępna analiza zagrożeń (ang. *Preliminary Hazard Analysis*), metoda umożliwiająca wstępne zidentyfikowanie ryzyka, związanego z eksploatowanym obiektem technicznym (Hyatt, 2010). Obejmuje ona identyfikację zdarzeń niebezpiecznych, dokonanie oceny możliwych ich skutków oraz prawdopodobieństwa ich wystąpienia. Na ich podstawie szacowane jest ryzyko wystąpienia poszczególnych zidentyfikowanych zdarzeń. Analiza PHA przeprowadzana jest zazwyczaj we wstępnych etapach projektu, kiedy nie są do końca znane pełne założenia projektu, a konieczne jest przeprowadzenia szybkiej analizy możliwego ryzyka.
- FMEA – analiza przyczyn i skutków występowania uszkodzeń (ang. *Failure Mode and Effects Analysis*), metoda oceny ryzyka nakierowana na potencjalne uszkodzenia poszczególnych podzespołów obiektu technicznego (Stamatis, 2003; Carlson, 2012). Obejmuje ona dekompozycję obiektu technicznego na funkcjonalne podzespoły, analizę możliwych uszkodzeń, ich przyczyn oraz skutków. Następnie wyznaczane są wskaźniki opisujące ryzyko, tj. prawdopodobieństwo wystąpienia uszkodzenia P, ważność uszkodzenia Z oraz możliwość jego wykrycia W. Na ich podstawie wyliczana jest priorytetowa liczba ryzyka LPR jako iloczyn wskaźników P, Z i W. Ostatnim etapem analizy jest wyselekcjonowanie uszkodzeń krytycznych oraz zaplanowanie stosownych działań zapobiegawczych.
- HAZOP – analiza zagrożeń i zdolności operacyjnych (ang. *Hazard and Operability Analysis*), metoda oceny ryzyka nakierowana na potencjalne odchylenia od normy podstawowych parametrów związanych z działaniem rozpatrywanego obiektu technicznego (Crawley i in., 2008; Hyatt, 2010). Dokonując oceny ryzyka za pośrednictwem tej metody, należy w pierwszej kolejności zidentyfikować wszystkie ważne parametry, związane z analizowanym obiektem. Następnie dla każdego z nich, ustala się możliwe jego odchylenia od stanu nominalnego. W przypadku przepływu może to być np.: brak przepływu, spadek lub też wzrost natężenia przepływu. Dla tak zdefiniowanych par (parametr – odchylenie), ustala się możliwe przyczyny oraz potencjalne skutki wystąpienia poszczególnych zdarzeń, a następnie próbuje się opracować możliwe środki zaradcze.
- FTA – analiza drzewa uszkodzeń (ang. *Fault Tree Analysis*), metoda analizy ryzyka, bazująca na poszukiwaniu możliwych przyczyn wybranych zdarzeń/uszkodzeń niepożądanych (Lee i in., 1985; Rausand i Høyland, 2004; Ericson, 2005). W metodzie tej, w pierwszej kolejności identyfikowane jest tzw. zdarzenie główne, uznane za szczególnie niebezpieczne

dla rozpatrywanego obiektu technicznego. Następnie rozpoznawane są wszystkie możliwe zdarzenia, mogące powodować wystąpienie zdarzenia głównego. Łączone są one ze sobą, za pomocą funktorów logicznych *AND* oraz *OR*. Proces ten powtarzany jest następnie dla wszystkich nowo identyfikowanych zdarzeń, aż do identyfikacji zdarzeń pierwotnych. W ten sposób powstaje drzewo uszkodzeń, obrazujące możliwe przyczyny wystąpienia uszkodzenia głównego.

- ETA – analiza drzewa zdarzeń (ang. *Event Tree Analysis*), metoda oceny ryzyka, oparta na analizie propagacji zdarzeń niebezpiecznych, mogących wystąpić w rozpatrywanym obiekcie (Greenberg i Cramer, 1991; Rausand i Høyland, 2004). Ocena ryzyka rozpoczyna się od identyfikacji tzw. zdarzenia inicjującego. Następnie identyfikowane są kolejne zdarzenia, które mogą wystąpić po zaistnieniu zdarzenia początkowego. Każde z nich rozpatrywane jest w dwóch kategoriach tj. wystąpienie zdarzenia oraz jego brak. Na podstawie zbioru zdarzeń możliwe jest zbudowanie drzewa opisującego pełną propagację zdarzeń, począwszy od zdarzenia inicjującego aż do zdarzenia końcowego. Każdemu zdarzeniu można przypisać prawdopodobieństwo jego wystąpienia. Bazując na nim, możliwe jest wyznaczanie ścieżek o potencjalnie największym ryzyku.
- LOPA – analiza warstw zabezpieczeń (ang. *Layer of Protection Analysis*), metoda oceny ryzyka bazująca na założeniu, że istnieje pewna zależność pomiędzy mogącymi wystąpić zagrożeniami a zastosowanymi systemami bezpieczeństwa (Kacprzak, 2011; CCPS, 2011). Analiza opiera sie na wyznaczeniu możliwych zagrożeń a następnie na opracowaniu możliwego scenariusza awaryjnego. W kolejnym kroku identyfikuje się możliwe do zastosowania systemy bezpieczeństwa oraz określa się, prawdopodobieństwo wystąpienia poszczególnych scenariuszy awaryjnych, przy założeniu zastosowania wybranych systemów bezpieczeństwa.

Wymienione powyżej metody są intensywnie rozwijane i modyfikowane, tak aby z ich wykorzystaniem jeszcze skuteczniej można było przeprowadzać proces oceny ryzyka technicznego.

3.4. Zarządzanie ryzykiem technicznym

Zarządzanie ryzykiem to wg norm (ECSS-M-00-03A:2010; PN ISO 31000:2012) skoordynowane działania, których celem jest ciągła kontrola i ograniczanie poziomu ryzyka w trakcie całego procesu eksploatacji obiektów technicznych. Można zatem powiedzieć, że zarządzanie ryzykiem obejmuje takie czynności jak (Greenberg i Cramer, 1991):

- analiza i ocena zdarzeń niebezpiecznych,
- sterowanie ryzykiem,
- sterowanie eksploatacją (obsługa, remonty),
- kontrola podjętych działań,
- wizualizacja ryzyka.

Dla potrzeb efektywnego zarządzania ryzykiem przyjmuje się, że ryzyko związane z eksploatacją określonego obiektu technicznego, może być (Berg, 2010):

- redukowane – wprowadzane są układy zabezpieczające oraz kontrolujące, których zadaniem jest wykrywanie zdarzeń obarczonych ryzykiem,
- akceptowane – nie są wprowadzane żadne środki zaradcze, akceptuje się możliwość wystąpienia wybranych zdarzeń, które mogą spowodować określone szkody, zazwyczaj koszt

układów zabezpieczających jest dużo większy niż potencjalne straty po wystąpieniu danego zdarzenia,

- transferowane – przeniesienie ryzyka na inny podmiot (niezależny od podmiotu eksploatującego obiekt techniczny), zazwyczaj do firmy ubezpieczeniowej,
- uniknięte – poprzez rezygnację z rozwiązań obarczonych ryzykiem, np. rezygnacja z wybranych modułów obiektu.

Wyróżnić można dwa podstawowe podejścia do zagadnienia minimalizacji ryzyka. Może być ono redukowane w sposób aktywny, czyli poprzez działania polegające na oddziaływaniu na bezpośrednie przyczyny i czynniki ryzyka oraz w sposób pasywny, czyli poprzez działania polegające na zabezpieczaniu przed możliwymi stratami, po wystąpieniu danego zdarzenia obarczonego ryzykiem. Podejmując wysiłki na rzecz minimalizacji ryzyka niepożądanego, konieczne jest zwrócenie uwagi (Skelton, 1997) na zasadę ALARP (ang. *As Low As Reasonably Practicable*), w wolnym tłumaczeniu *tak niskie jak to jest racjonalnie uzasadnione*. Mówi ona, że działania związane z ograniczaniem ryzyka, powinny być kontynuowane jedynie do punktu, kiedy poziom wydatków koniecznych na kolejną redukcję ryzyka, będzie powodował nieproporcjonalnie mały spadek ryzyka. Ponadto ważne jest też spostrzeżenie, że łączenie razem pojedynczych maszyn uznanych za bezpieczne (obarczonych niskim ryzykiem), może powodować wystąpienie różnego rodzaju nowych sytuacji niebezpiecznych, czasami o znacznie większym spektrum oddziaływania niż w przypadku pojedynczej maszyny. Dlatego też dla takich złożonych obiektów technicznych, konieczne jest przeprowadzenie dodatkowych analiz ryzyka.

Przeprowadzanie oceny ryzyka na poziomie jakościowym, może być w wielu przypadkach znacznie utrudnione. Powodem są trudności w dokładnym opisie prawdopodobieństwa wystąpienia zdarzenia oraz potencjalnych jego skutków na poziomie jakościowym. Możliwym rozwiązaniem jest m.in. wykorzystanie (Bowles i Peláez, 1995; Petrović i in., 2014) zbiorów rozmytych do przybliżonej reprezentacji wartości jakościowych.

Wyniki analizy ryzyka mogą być bezpośrednio zastosowane w procesie eksploatacji obiektów technicznych. Dla takich potrzeb, wyróżnić można m.in. metodę planowania inspekcji RBI (ang. *Risk Based Inspection*) oraz metodę planowania eksploatacji RBM (ang. *Risk Based Maintenance*). W metodach tych (Jovanovic, 2003; Khan i Haddara, 2003) inspekcje, kontrole oraz ciągła eksploatacja obiektu technicznego, odbywają się na podstawie opracowanego rankingu zdarzeń obarczonych największym ryzykiem. Na jego podstawie, podejmowane są decyzje odnośnie tego, które elementy obiektu mają być w pierwszej kolejności poddane kontroli oraz z jaką częstotliwością. Na podstawie tych oraz kilku innych metod opracowana została w ramach projektu RIMAP (ang. *Risk Based Inspection and Maintenance Procedures for European Industry*) europejska procedura minimalizacji ryzyka (CWA 15740:2008; Kauer i in., 2004). W projekcie tym, brało udział 16 europejskich przedsiębiorstw m.in. z branży chemicznej, petrochemicznej oraz energetycznej. Celem projektu było opracowanie strategii eksploatacji/inspekcji obiektów technicznych, dzięki której znacznej redukcji ulegną koszty związane z obsługą obiektu lub potencjalnymi stratami po wystąpieniu zdarzenia niebezpiecznego. Pierwszym etapem opracowanej procedury jest analiza wstępna, obejmująca m.in. zidentyfikowanie przedmiotu badań, w tym hierarchicznej struktury obiektu, określenie oczekiwanych rezultatów itd. W kolejnym etapie dokonywana jest wielopoziomowa analiza ryzyka, obejmująca określenie możliwych zdarzeń niebezpiecznych, wyznaczenie prawdopodobieństwa wystąpienia danego zdarzenia oraz możliwych jego skutków. Na ich podstawie dokonywane jest oszacowanie ryzyka oraz podejmowana jest stosowna decyzja odnośnie zaplanowania koniecznych remontów, inspekcji itp. rozpatrywanego obiektu technicznego. Po wprowadzeniu stosownych działań, dokonywana jest ocena ich efek-

towności i cała procedura powtarzana jest ponownie.

W wyniku przeprowadzonej oceny ryzyka, wdrażane są różnego rodzaju środki zapobiegawcze, zazwyczaj w postaci układów zabezpieczających bądź systemów nadzorujących oraz diagnostycznych. Zabezpieczony w taki sposób obiekt techniczny, powinien spełniać określone normy bezpieczeństwa, w tym tzw. poziom nienaruszalności bezpieczeństwa SIL (ang. *Safety Integrity Level*). Wyróżnia się cztery poziomy SIL, każdy z nich opisuje prawdopodobieństwo niezadziałania układu bezpieczeństwa przy założeniu pracy obiektu w dwóch trybach, tj. praca na żądanie oraz praca w trybie ciągłym. Wartości prawdopodobieństw, opowiadające poszczególnym poziomom SIL wyszczególniono w tabelach 3.4 i 3.5.

Tab. 3.4: Poziom nienaruszalności bezpieczeństwa (wg IEC EN 61508) dla systemów pracujących w trybie na żądanie, PFD – prawdopodobieństwo uszkodzenia na żądanie (ang. *Probability of Failure on Demand*)

Poziom SIL	**Wartość PFD**
SIL 1	$10^{-2} - 10^{-1}$
SIL 2	$10^{-3} - 10^{-2}$
SIL 3	$10^{-4} - 10^{-3}$
SIL 4	$10^{-5} - 10^{-4}$

Tab. 3.5: Poziom nienaruszalności bezpieczeństwa (wg IEC EN 61508) dla systemów pracujących w trybie ciągłym, PFH – prawdopodobieństwo uszkodzenia na godzinę (ang. *Probability of Failure per Hour*)

Poziom SIL	**Wartość PFH**
SIL 1	$10^{-6} - 10^{-5}$
SIL 2	$10^{-7} - 10^{-6}$
SIL 3	$10^{-8} - 10^{-7}$
SIL 4	$10^{-9} - 10^{-8}$

3.5. Podsumowanie

Minimalizacja ryzyka technicznego, związanego z eksploatowanymi obiektami technicznymi, jest zagadnieniem niezwykle ważnym i jednocześnie w wielu przypadkach bardzo trudnym. Trudności te, związane są przede wszystkim z potrzebą dokładnego wskazania potencjalnych zdarzeń niebezpiecznych oraz określenia ewentualnych niekorzystnych skutków takich zdarzeń. W wielu przypadkach nie jest możliwe dokładne oszacowanie tych dwóch parametrów z powodu braku wystarczającej wiedzy dziedzinowej, w szczególności braku informacji o wpływie pewnych specyficznych czynników otoczenia, odmiennych niż standardowe, na działanie rozpatrywanych obiektów technicznych.

Jednakże stosowanie wybranych metod oceny ryzyka oraz uwzględniane wyników przeprowadzonych analiz, w procesie eksploatacji obiektów technicznych, pozwala na zmniejszenie prawdopodobieństwa wystąpienia określonych zdarzeń, co zatem prowadzi do zwiększenia bezpieczeństwa użytkowanych maszyn.

Rozdział 4

Wybrane zagadnienia inżynierii wymagań

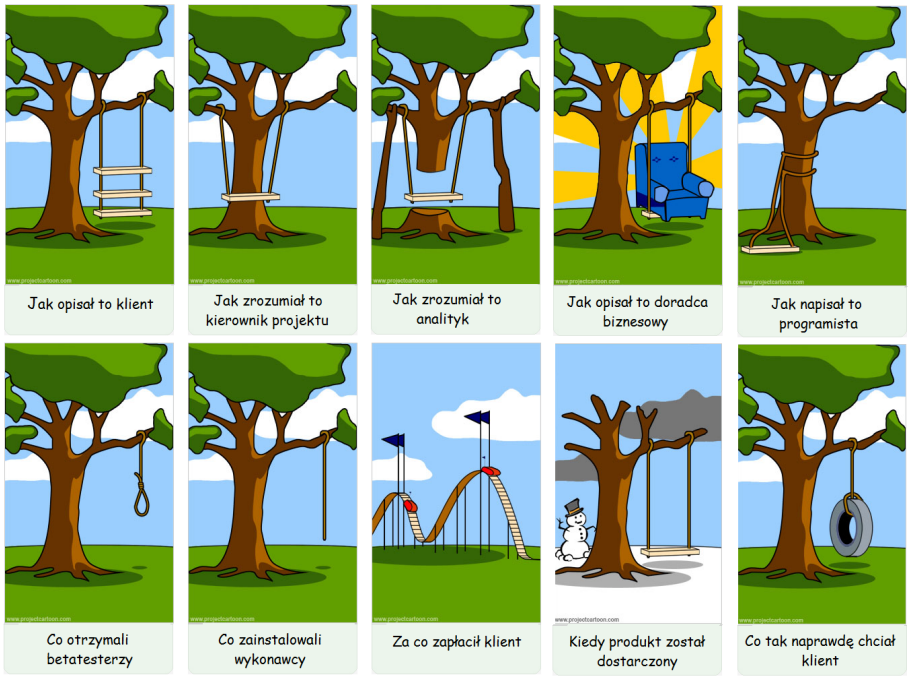
W niniejszym rozdziale opisano problematykę inżynierii wymagań. Nakreślono tło historyczne związane z początkami rozwoju tej dziedziny. Omówiono termin wymaganie, przedstawiono możliwe źródła wymagań, sposoby ich pozyskiwania oraz metody zarządzania wymaganiami.

4.1. Zarys problematyki

Na przestrzeni lat, wraz ze wzrostem poziomu skomplikowania opracowywanych środków technicznych, pojawiały się różne problemy w prawidłowym uchwyceniu idei i oczekiwań stawianych opracowywanym rozwiązaniom. Szczególnie szybko problemy zaczęły pojawiać się w latach 70–80 XX wieku, w obszarze informatyki, która przeżywała wtedy olbrzymi rozwój. Rosnący poziom skomplikowania opracowywanych systemów, liczne oczekiwania klientów i potencjalnych użytkowników, własne wizje programistów co do właściwego wyglądu i sposobu działania opracowywanych systemów itp., sprawiały że w wielu przypadkach nie udawało się osiągnąć zamierzonych celów, jakimi było opracowanie systemu zgodnie z potrzebami i oczekiwaniami klienta. W wielu przypadkach pojawiała się sytuacja pokazana, w sposób humorystyczny, na rys. 4.1.

Konieczne zatem stało się opracowanie, jakiegoś formalnego sposobu zapisu oczekiwań klienta w stosunku do projektowanego systemu. Przyjęto, że zapis powinien być zrozumiały zarówno dla klienta zlecającego wykonanie systemu jak również dla programisty, który miał go zrealizować. Ponadto musi opisywać oczekiwane funkcjonalności oraz cechy projektowanego systemu oraz na podstawie opracowanego zapisu musi istnieć możliwość sprawdzenia, czy zaimplementowany system jest zgodny z wyartykułowanymi oczekiwaniami klienta.

W latach 80 XX wieku, pojawiły się (Alfor i Lawson, 1979) pierwsze zalążki nowej metodologi, która z czasem przekształciła się w inżynierię wymagań (ang. *requirements engineering*), w ramach której wykorzystuje się zbiory wymagań do opisu oczekiwanych funkcjonalności projektowanych systemów informatycznych.



Rys. 4.1: Wizja wyniku końcowego realizowanego projektu z punktu widzenia różnych jego uczestników, (opracowanie własne na podstawie `www.projectcartoon.com`[1])

4.2. Wymagania oraz ich rodzaje

W literaturze przedmiotu zawartych jest bardzo wiele rożnych definicji terminu *wymaganie*. Zazwyczaj są one związane z konkretną dziedziną nauki, np. z inżynierią oprogramowania, w której pojęcie to jest powszechnie stosowane. Spośród licznych definicji rozpatrywanego terminu warto odnotować następujące:

- *1. Warunek lub możliwość, która jest potrzebna użytkownikowi do rozwiązania lub osiągnięcia konkretnego celu.*
  2. Warunek lub możliwość, którą musi mieć system lub komponent systemu aby spełnić warunki kontraktu, standardu, specyfikacji lub innego narzuconego dokumentu.
  3. Udokumentowana reprezentacja warunku lub możliwości opisanych w punktach 1 i 2 (IEEE Std 610).
- *Specyfikacja tego, co powinno być zaimplementowane w systemie. Wymagania są opisem tego jak system powinien się zachować lub opisem właściwości systemu. Mogą one być ograniczeniem podczas procesu wytwarzania systemu* (Sommerville i Sawyer, 1997).
- *1. Możliwość rozwiązania problemu i osiągnięcia celu, wymagana przez użytkownika.*
  2. Możliwość spełnienia umowy, normy, specyfikacji lub innej narzuconej dokumentacji, którą musi mieć system lub komponent systemu (Leffingwell i Widrig, 2003).
- *Stwierdzenie opisujące niezbędne atrybuty, możliwości, charakterystykę lub wskaźniki jakościowe projektowanego systemu, tak aby mógł on być użyteczny dla klienta lub użytkownika końcowego* (Young, 2004).

Możliwe jest podjęcie próby uogólnienia tych definicji poprzez uznanie, że wymaganie to *sformalizowany opis funkcji/cechy, którą powinien realizować lub posiadać projektowany środek techniczny.*

[1]Dostępne online: 5 września 2015

Niezależnie od przyjętych definicji, wymagania stosowane są do opisu projektowanych środków technicznych. Przykładami wymagań dla potrzeb opracowywania projektu systemu diagnostycznego mogą być zatem następujące sformułowania:

- do prawidłowego określenia stanu technicznego obiektu konieczne jest wykrywanie zużycia bieżni łożyska,
- konieczny jest pomiar ciśnienia w rurociągu doprowadzającym czynnik roboczy do zaworu regulacyjnego ZR03,
- pomiar drgań względnych w węźle łożyskowym nr 5, ma odbywać się z zastosowaniem czujnika drgań o zakresie pomiarowym od X_{min} do X_{max} z dokładnością do X_{dop}.

Należy zauważyć, że definiowane wymagania nie mogą zawierać opisu rozwiązania danego problemu a jedynie opis potrzeby. Równocześnie należy mieć na uwadze fakt, że pojęcie wymaganie nie jest tożsame z terminem potrzeba. W ogólnym przypadku pojedyncza potrzeba może być opisana licznym zbiorem wymagań. Jak również różne potrzeby mogą determinować konieczność istnienia tych samych lub podobnych wymagań.

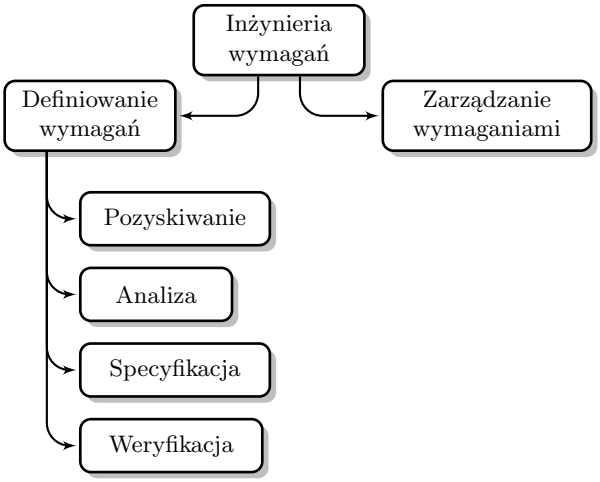
Analizując literaturę przedmiotu m.in. (Sommerville i Sawyer, 1997; Sommerville, 2003; Wiegers i Beatty, 2014) można znaleźć różne klasyfikacje i podziały pozyskiwanych wymagań. Zasadniczo jednak wyróżnia się dwie grupy wymagań tj. wymagania funkcjonalne oraz wymagania pozafunkcjonalne, nazywane też wymaganiami niefunkcjonalnymi. Wymagania funkcjonalne opisują funkcjonalności (operacje, czynności i usługi) jakie projektowane rozwiązanie powinno dostarczać. O tym w jaki sposób są one spełniane (cechy projektowanego środka technicznego) decydują wymagania pozafunkcjonalne (Chung i do Prado Leite, 2009), czasami nazywane również wymaganiami jakościowymi. Poszczególne wymagania można jednocześnie grupować w pewne tematyczne podzbiory, zawierające wymagania związane z wybranymi aspektami lub dziedzinami. Na tej podstawie można wyróżnić m.in. (Abran i in., 2004; Young, 2004) wymagania biznesowe, organizacyjne, użyteczności, wydajnościowe, eksploatacyjne, bezpieczeństwa, prawne, wyglądu itp. Dla potrzeb projektów informatycznych, bardzo często wykorzystuje się model klasyfikacji wymagań FURPS (Grady i Caswell, 1987). Umożliwia on przypisanie wymagań do następujących klas: funkcjonalność (ang. *Functionality*), użyteczność (ang. *Usability*), niezawodność (ang. *Reliability*), wydajność (ang. *Performance*), wsparcie (ang. *Supportability*)

Dziedziną kompleksowo zajmującą się wymaganiami oraz wszelkimi aspektami z nimi związanymi, jest inżynieria wymagań. Jej podstawowe komponenty pokazano na rys. 4.2. Wśród nich wyróżnić można dwa podstawowe procesy, tj. definiowanie oraz zarządzanie wymaganiami.

4.3. Definiowanie zbioru wymagań

Definiowanie wymagań jest złożonym procesem, którego wynikiem końcowym powinien być zbiór przyjętych wymagań, opisujących projektowany środek techniczny. W skład tego procesu wchodzą cztery etapy, tj. pozyskiwanie, analiza, specyfikacja oraz weryfikacja wymagań.

Celem etapu pozyskiwania wymagań jest zgromadzenie wymagań stawianych projektowanemu środkowi technicznemu. W literaturze angielskojęzycznej znaleźć można wiele nazw określających ten etap definiowania wymagań. Oprócz wspomnianego już pozyskiwania (ang. *elicitation*), występuje również gromadzenie (ang. *gathering, acquisition*), formułowanie (ang. *formulating*), identyfikacja (ang. *identifying*), określanie (ang. *determining*) oraz ujawnianie (ang. *exposing, extracting*). Każda z tych nazw oddaje pewną specyfikę etapu pozyskiwania wymagań, która w dużej mierze zależy od własności danego projektu, obszaru w jakim jest on wykonywany



Rys. 4.2: Elementy inżynierii wymagań

oraz od postaci wiedzy o projektowanym środku technicznym jaką dysponują osoby pozyskujące wymagania. Należy podkreślić, że na etapie pozyskiwania wymagań mogą zostać zebrane wymagania, które będą sprzeczne z innymi wymaganiami bądź też z przyjętymi lub narzuconymi ograniczeniami. Są one eliminowane na kolejnych etapach procesu pozyskiwania wymagań, w tym na etapie analizy. Celem tego etapu jest ocena zebranych wymagań, usunięcie wymagań sprzecznych oraz zbędnych z punktu widzenia celu projektu. Zapis wymagań do postaci odpowiedniej dla projektantów danego systemu następuje na etapie specyfikacji. Wśród możliwych formatów zapisu możemy wyróżnić m.in. język naturalny, język formalny, zapis symboliczny czy też zapis graficzny. Ostatnim etapem procesu definiowania wymagań jest weryfikacja, podczas której sprawdza się, np. wraz ze zleceniodawcą lub użytkownikiem końcowym projektowanego środka technicznego, poprawność, spójność, kompletność oraz ważność wszystkich zebranych wymagań (Leffingwell i Widrig, 2003).

Należy zauważyć, że wymienione etapy są częściowo zależne i bardzo często realizowane są (lub muszą być) jednocześnie a ich wyniki bezpośrednio wpływają na siebie. W celu poprawnego (i/lub łatwiejszego) przeprowadzenia procesu definiowania wymagań, konieczne jest zastosowanie wybranych metod zarządzania wymaganiami.

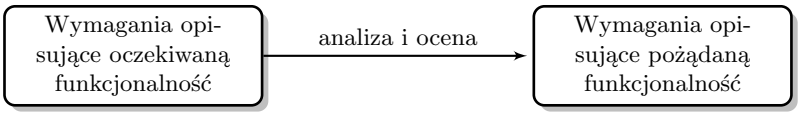
Proces pozyskiwania wymagań nie jest zadaniem łatwym. Stopień jego trudności zależy od wielu czynników, w tym m.in. od rodzaju projektu, wiedzy i doświadczenia osób realizujących projekt, przekonania osób zlecających wykonanie projektu, stopnia innowacyjności rozwiązań, które mają być zastosowane w projektowanym środku technicznym, dostępnej wiedzy itp. W ogólnym przypadku źródłem wymagań mogą być m.in. (Sommerville i Sawyer, 1997; Abran i in., 2004; Chrabski i Zmitrowicz, 2014; Wiegers i Beatty, 2014):

- zleceniodawca projektu,
- użytkownik końcowy,
- wiedza i doświadczenie wykonawców projektu,
- istniejące rozwiązania oraz prototypy opracowywanego środka technicznego,
- standardy, normy i zalecenia, przepisy prawa,
- wiedza zawarta w pozycjach literaturowych, katalogach itp.,
- eksperci dziedzinowi,
- ograniczenia i cele biznesowe.

W zależności od specyfiki projektu, dostęp do poszczególnych źródeł wymagań może być znacznie utrudniony a wielu przypadkach niemożliwy. Zasadniczo wskazane jest pozyskiwanie wymagań z jak największej liczby źródeł, celem uchwycenia różnych punktów widzenia projektowanego środka technicznego. Należy jednak przy tym pamiętać, że wymagania pozyskiwane z wielu różnych źródeł mogą okazać się niespójne lub sprzeczne. Liczba wymagań koniecznych do pozyskania zależy od specyfiki realizowanego projektu i może wahać się od kilkuset dla małych projektów do kilkuset tysięcy dla rozbudowanych i złożonych projektów, przykładowo 300 tyś. przy projektowaniu samolotów pasażerskich (Leffingwell i Widrig, 2003).

Dla potrzeb pozyskiwania wymagań możliwe jest zastosowanie różnych technik pozyskiwania wymagań. Najważniejsze z nich to rozmowy, kwestionariusze, obserwacje, analizy danych literaturowych, prototypowanie, analiza przypadków użycia, analiza zdarzeń i reakcji itp. Dokładny opis poszczególnych technik można znaleźć w licznych pozycjach literaturowych m.in. w (Raghavan i in., 1994; Nuseibeh i Easterbrook, 2000; Robertson, 2001; Young, 2002; Abran i in., 2004; Hickey i Davis, 2004; Zowghi i Coulin, 2005; Wiegers i Beatty, 2014). Wybór odpowiednich technik pozyskiwania wymagań nie jest zadaniem łatwym i zależy on od wielu czynników, w tym m.in.: od rodzaju projektu, wiedzy i umiejętności osób definiujących wymagania, lokalizacji osób definiujących wymagania itp. W wielu przypadkach wskazane jest, lub też jest to niezbędne, aby wymagania były pozyskiwane jednocześnie z wykorzystaniem kilku różnych technik (Wiegers i Beatty, 2014). Proces wyboru odpowiedniej techniki pozyskiwania wymagań, może być dokonywany m.in. (Tang i Feng, 2009) poprzez zastosowanie systemu doradczego opracowanego na podstawie ankiet wypełnionych przez specjalistów z obszaru inżynierii wymagań.

W procesie pozyskiwania wymagań gromadzony jest bardzo liczby zbiór wymagań. W zbiorze tym mogą znaleźć się wymagania, które będą błędne, sprzeczne z innymi wymaganiami lub ograniczeniami projektowymi, opisujące rozwiązania nierealizowalne technicznie itp. Uznaje się, że wszystkie te wymagania opisują (przy uwzględnieniu różnych aspektów i punktów widzenia) oczekiwane funkcjonalności projektowanego środka technicznego. Dokonując analizy poszczególnych wymagań pod kątem ich poprawności, przydatności, zgodności z ograniczeniami itp., otrzymuje się zbiór wymagań opisujących pożądane funkcjonalności projektowanego rozwiązania (rys. 4.3). Należy zauważyć, że pominięcie ważnych wymagań lub też zdefiniowanie wymagań niedokładnych lub błędnie opisujących pożądane funkcje/cechy produktu, skutkuje wzrostem nakładów finansowych ponoszonych w kolejnych etapach realizacji projektu, potrzebnych do jego ukończenia. Wszystkie zdefiniowane i przyjęte (po dokonaniu ich oceny) wymagania, zapisywa-



Rys. 4.3: Dwa skrajne zbiory wymagań

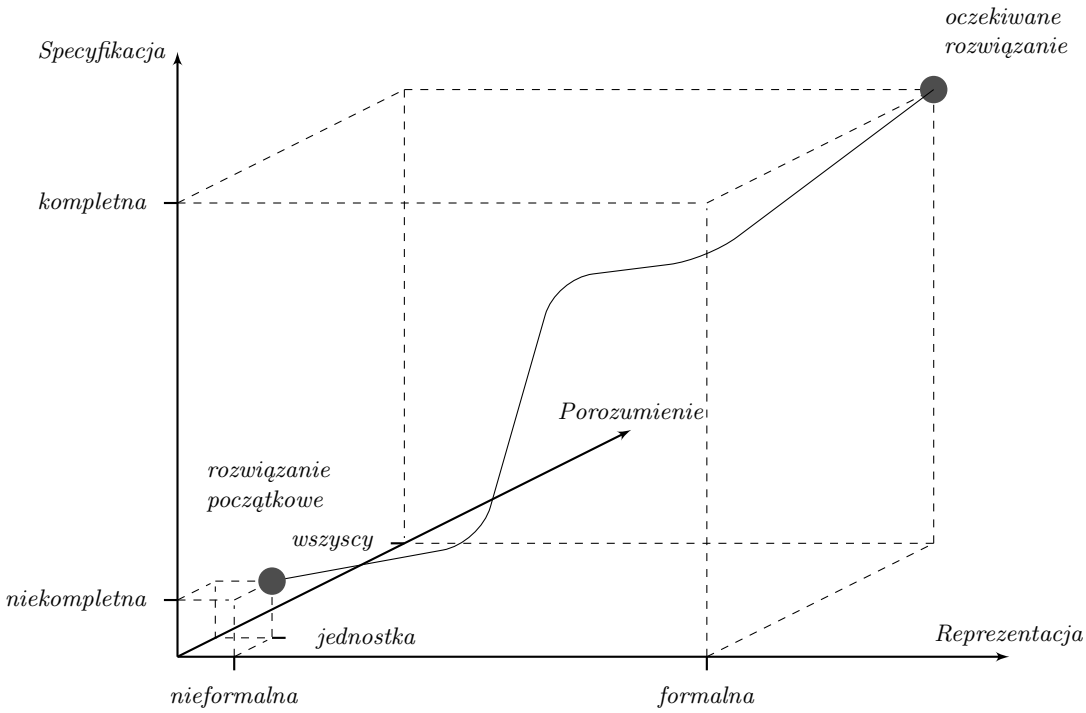
ne są w dokumencie o nazwie specyfikacja wymagań. Dokument ten opisuje ostateczną postać projektowanego środka technicznego. Poprawnie przygotowana specyfikacja wymagań dla projektów z obszaru inżynierii oprogramowania, powinna spełniać wytyczne zawarte w standardzie IEEE 830. Zatem powinna być ona (IEEE Std 830):

- poprawna (ang. *correct*), każde z zebranych wymagań jest wymaganiem, które projektowany system musi spełnić, ponadto musi być ono zgodne z obowiązującymi standardami i regulacjami (nie można w specyfikacji wymagań zawrzeć wymagania sprzecznego np. z polskim prawem),

- jednoznaczna (ang. *unambiguous*), każde ze zgromadzonych wymagań musi być jednoznacznie zdefiniowane, tzn. może być tylko jedna możliwość interpretacji poszczególnych wymagań, niezwykle trudne jest to w przypadku zapisu wymagań z wykorzystaniem języka naturalnego, ponieważ poszczególne osoby różnie interpretują zapisany tekst,
- kompletna (ang. *complete*), specyfikacja wymagań powinna zawierać zbiór wszystkich możliwych i istotnych z określonego punktu widzenia wymagań związanych z danym projektem,
- spójna (ang. *consistent*), pomiędzy zebrany wymaganiami nie mogą występować konflikty i sprzeczności, konfliktem może być np. różna nazwa dla tych samych obiektów, bądź też różne definicje tych samych funkcji systemu w kilku wymaganiach,
- uporządkowana wg ważności/stabilności (ang. *ranked for importance and/or stability*), każde ze zgromadzonych wymagań powinno mieć określony priorytet (stopień ważności), dzięki czemu możliwe jest łatwiejsze zarządzanie zbiorem wymagań, porządkowanie ich według stopnia ważności,
- weryfikowalna (ang. *verifiable*), wymaganie jest weryfikowalne, gdy istnieje proces (o pewnym dopuszczalnym, maksymalnym koszcie i skończonym czasie), za pomocą którego możliwe jest sprawdzenie czy dane wymaganie zostanie spełnione,
- modyfikowalna (ang. *modifiable*), przygotowana specyfikacja wymagań jest modyfikowalna wtedy gdy jej struktura i styl pozwalają na dokonywanie zmian w wymaganiach w prosty, kompletny i spójny sposób,
- umożliwiająca śledzenie powiązań (ang. *traceable*), możliwe jest określenie pochodzenia każdego wymagania (jego źródła), tworzenie powiązań pomiędzy zebranymi wymaganiami a także ustalenie referencji pomiędzy konkretnym wymaganiem a innymi artefaktami np.: projektem technicznym, kodem programu, itp.

Z uwagi na fakt, że są to bardzo uniwersalne wytyczne, można przyjąć że mogą być one zastosowane także przy realizacji projektów w obrębie innych dziedzin.

Proces pozyskiwania właściwych wymagań jest w wielu przypadkach zadaniem trudnym. Niezależnie jednak od przyjętych metodologii oraz zastosowanych metod i technik może być on rozumiany (Pohl, 1993), jako poszukiwanie rozwiązania w trójwymiarowej przestrzeni (rys. 4.4) opisanej przez specyfikację, reprezentację oraz porozumienie. W przestrzeni tej wyróżnić można dwa skrajne rozwiązania, tj. rozwiązanie początkowe oraz rozwiązanie oczekiwane (rozwiązania te mogą być utożsamiane ze zbiorami wymagań pokazanymi na rys. 4.3). Rozwiązanie początkowe jest niekompletnym, zapisanym w sposób nieformalny zbiorem wymagań zdefiniowanych przez pojedynczego uczestnika procesu definiowania wymagań. Z kolei oczekiwane rozwiązanie zawiera pełny zbiór wymagań, zapisanych w sposób formalny oraz uzgodnionych pomiędzy wszystkimi osobami definiującymi wymagania. Proces dochodzenia od rozwiązania początkowego (zazwyczaj wielu rozwiązań początkowych) do rozwiązania końcowego jest w wielu przypadkach niezwykle trudny. Zazwyczaj przybiera on formę negocjacji pomiędzy osobą zlecającą projekt (klient) a potencjalnym jego wykonawcą. Podczas tych negocjacji, klient wykorzystując wybrane techniki, artykułuje wymagania opisujące oczekiwane funkcjonalności projektowanego środka technicznego. Każde z pojawiających się wymagań może być zatwierdzone, zmodyfikowane lub odrzucone przez potencjalnego wykonawcę (Grünbacher i Seyff, 2005). Negocjacje mogą być realizowane przy wykorzystaniu różnych metodologii. Jedną z nich jest metodologia Easy-WinWin (Grünbacher i Boehm, 2001). Wywodzi się ona z modelu negocjacji WinWin, który zakłada, że każdy uczestnik negocjacji powinien je zakończyć w poczuciu zwycięstwa i osiągnięcia zamierzonego celu. Jeżeli w procesie negocjacji pojawią się wymagania, które nie będą



Rys. 4.4: Trzy wymiary inżynierii wymagań (Pohl, 1993)

mogły zostać zaakceptowane przez wykonawcę, to należy zidentyfikować przyczyny tego stanu oraz wspólnie z klientem znaleźć takie rozwiązanie, które usatysfakcjonuje obie strony.

4.4. Zarządzanie procesem definiowania wymagań

Zarządzanie wymaganiami to (Wiegers i Beatty, 2014) kompleksowe podejście do procesu definiowania wymagań, którego nadrzędnym celem jest utrzymanie integralności i kompletności definiowanego zbioru wymagań. Obejmuje ono m.in. takie czynności jak:

- aktualizacja zbioru wymagań,
- śledzenie zmian wprowadzanych w wymaganiach,
- ocena wpływu wprowadzanych zmian w wymaganiach na realizowany projekt,
- definiowanie zależności pomiędzy poszczególnymi wymaganiami a fragmentami realizowanego projektu, kodu programu itp.,
- śledzenie zmian statusów wymagań.

Problem prawidłowego zarządzania w procesie pozyskiwania wymagań jest szczególnie ważny w przypadku złożonych projektów, dla których konieczne jest pozyskanie dużego zbioru wymagań. W takim przypadku, jedną z podstawowych trudności jest przeglądanie posiadanego zbioru wymagań. Proces ten może być znaczenie ułatwiony, poprzez przyporządkowywanie definiowanym wymaganiom stosownych atrybutów. Atrybutami tymi (Young, 2004; Wiegers i Beatty, 2014) mogą być m.in. identyfikator, data utworzenia, numer wersji, autor wymagania, źródło wymagania, priorytet, status itp. Na podstawie tych atrybutów, możliwe jest filtrowanie zbioru zgromadzonych wymagań, celem wyodrębnienia pewnego podzbioru interesujących wymagań.

Kolejnym ważnym elementem w zarządzaniu zbiorem wymagań, jest nadawanie odpowiednich priorytetów poszczególnym wymaganiom. Nadawanie priorytetów jest potrzebne szczególnie

w przypadku projektów, które mają bardzo ograniczone ramy czasowe lub mają być realizowane w kilku etapach (np. dla projektów zwinnych[2]). W takich przypadkach wskazane jest wyodrębnienie tych wymagań, które są najbardziej ważne z punktu widzenia działania całego projektowanego obiektu i zrealizowanie ich w pierwszej kolejności. Kolejne etapy procesu określania priorytetów opisano m.in. w (Firesmith, 2004). Priorytety mogą być nadawane (Karlsson i Ryan, 1997) na podstawie szacowanego wkładu poszczególnych wymagań, w końcową wartość projektowanego obiektu oraz kosztu jaki trzeba ponieść aby je zaimplementować. Szacowanie odbywa się z wykorzystaniem metody AHP[3]. Wymagania o najwyższym priorytecie, będą miały zatem największy wpływ na wartość produktu oraz będą charakteryzowały się niskim kosztem. W przypadku dużych zespołów projektowych ciekawym rozwiązaniem stosowanym do nadawania priorytetów poszczególnym wymaganiom jest zastosowanie rozwiązań (Lim, 2010; Lim i Finkelstein, 2012) opartych na sieciach społecznych (ang. *social network*) oraz systemach rekomendacji bazujących na tzw. *collaborative filtering*[4]. Podejście to zakłada, że poszczególni uczestnicy procesu definiowania wymagań, na podstawie posiadanych przez siebie informacji wystawiają rekomendacje innym członkom zespołu projektowego. Każda przyznawana rekomendacja zawiera subiektywną ocenę wpływu danej osoby na końcowy kształt projektu oraz jej rolę w tym projekcie. Tworzona jest w ten sposób sieć zależności pomiędzy członkami zespołu projektowego. Następnie wybrane osoby przypisują poszczególnym wymaganiom ich stopnie ważności. Końcowe priorytety przyporządkowane poszczególnym wymaganiom, wyznaczane są na podstawie wag przypisanych poszczególnym wymaganiom, przy jednoczesnym uwzględnieniu istniejących zależności pomiędzy osobami definiującymi wymagania.

W niektórych przypadkach (Wiegers i Beatty, 2014) możliwe jest nadawanie priorytetów zgodnie z metodologią *MoSCoW*. Jej nazwa pochodzi od angielskich nazw czterem priorytetów, które można nadać poszczególnym wymaganiom. Są nimi:

- musi (ang. *must*) – rozpatrywane wymaganie musi zostać zaimplementowane w projektowanym środku technicznym. Brak uwzględnienia danego wymagania, jest podstawą do uznania, że dany projekt nie zakończył się sukcesem.
- powinno (ang. *should*) – przypisanie danego priorytetu do wybranego wymagania oznacza, że wymaganie jest ważne i powinno być ono zaimplementowane, jednakże nie jest ono konieczne aby uznać cały projekt za zakończony sukcesem.
- może (ang. *could*) – dane wymaganie jest wskazane, jednakże można go pominąć i wrócić do niego jeżeli pozwoli na to czas oraz zasoby ludzkie i finansowe.
- nie będzie (ang. *won't*) – dane wymaganie nie będzie realizowane w obecnej wersji projektowanego środka technicznego.

[2]Projekty zwinne (ang. *agile projects*) to jeden z nowych trendów w obszarze inżynierii oprogramowania, charakteryzują się one częstymi weryfikacjami wymagań oraz etapowym dostarczaniem produktu wraz z dodawaniem do niego kolejnych funkcjonalności, celem weryfikacji ich poprawności przez klienta lub użytkownika końcowego (Chow i Cao, 2008).

[3]Analityczny proces hierarchiczny (ang. *Analytic Hierarchy Process*) – opracowana przez Thomasa L. Saatiego metoda stosowana do hierarchicznej analizy procesów decyzyjnych. Zakłada ona dekompozycję złożonego problemu decyzyjnego na skończona liczbę jego wariantów, badanie preferencji poszczególnych wariantów w zależności od przyjętych kryteriów a następnie utworzenie końcowego rankingu dla tych wariantów (Saaty, 1980).

[4]Technika prognozowania preferencji (decyzji) bazująca na założeniu, że przyszłe preferencje (wybory) dokonane przez użytkownika można przewidywać na podstawie dokonanych wcześniej przez niego preferencji oraz preferencji innych użytkowników, z którymi to dany użytkownik pozostaje w pewnej relacji podobieństwa na podstawie podjętych preferencji, częściowo lub całkowicie ignorując jednocześnie informacje o cechach charakteryzujących tych użytkowników. Technika ta jest powszechnie stosowana m.in. w systemach rekomendacji filmów/książek w wypożyczalniach/sklepach internetowych oraz do wspomagania zarządzania profilami w serwisach społecznościowych (Ekstrand i in., 2010).

W ostatnich latach można również zauważyć, wykorzystanie rozwiązań opartych na analizie tekstu, do wspomagania procesu pozyskiwania wymagań. Metody te, stosowane są głównie do wyszukiwania podobieństwa pomiędzy istniejącymi a dodawanymi wymaganiami, wyszukiwania powiązań pomiędzy zdefiniowanymi wymaganiami (Natt och Dag i in., 2005), analizy treści zdefiniowanych wymagań celem tworzenia modeli ontologicznych lub słowników terminów (Kof, 2004; Ilieva i Ormandjieva, 2005) itp.

Proces pozyskiwania wymagań może być wspomagany poprzez zastosowanie różnego rodzaju środowisk informatycznych, zarówno komercyjnych jak i open source. Przegląd dostępnych rozwiązań znajduje się m.in. w zestawieniach opracowanych przez organizacje Martinig & Associates[5] oraz Volere[6]. W pozycji (de Gea i in., 2011) zawarto porównanie wybranych aplikacji wykorzystywanych w obrębie inżynierii wymagań oraz podjęto próbę przyporządkowania im wybranych scenariuszy procesu pozyskiwania wymagań, natomiast w pozycji (Hoffmann i in., 2004) wyszczególniono najważniejsze wymagania stawiane takim aplikacjom. Na podstawie analizy informacji zawartych we wspomnianych publikacjach można stwierdzić, że dostępne rozwiązania wspomagają przede wszystkim proces dokumentowania definiowanych wymagań, a zatem pozwalają na zapis wymagań oraz na przypisywanie definiowanym wymaganiom różnego rodzaju atrybutów. Ponadto pozwalają na skuteczne filtrowanie oraz wyszukiwanie określonych wymagań. Zapewniają możliwość śledzenia powiązań pomiędzy wybranymi wymaganiami, umożliwiają współpracę z zewnętrznymi aplikacjami np. bazami danych SQL, dokumentami pakietu MS Office oraz zapewniają możliwość koordynacji pracy grupowej podczas procesu pozyskiwania wymagań. Należy podkreślić, że obecnie trwają intensywne prace związane z rozwojem metod zarządzania wymaganiami, których celem jest (Cheng i Atlee, 2007) m.in. opracowywanie nowych metodologii pozyskiwania wymagań, lepszych formatów zapisu wymagań itp.

4.5. Ograniczenia w procesie pozyskiwania wymagań

Proces pozyskiwania wymagań wiąże się z koniecznością realizacji wielu różnych zadań. Wykonanie części z nich może być utrudnione z powodu wystąpienia różnego rodzaju problemów i ograniczeń. Literatura przedmiotu m.in. (Borenstein, 1994; Saiedian i Dale, 2000; Leffingwell i Widrig, 2003; Wiegers i Beatty, 2014) podaje szereg możliwych niedogodności, wśród których najważniejsze to:

- słaba komunikacja pomiędzy osobami uzgadniającymi wymagania,
- różny punkt widzenia poszczególnych członków zespołu projektowego (głównie klient i wykonawca) na końcową postać opracowywanego środka technicznego,
- opór przed realizacją nowatorskich pomysłów,
- niezgodność i niejednoznaczność stosowanych przez poszczególne osoby terminów i pojęć,
- różne formaty zapisu wymagań zgłaszanych przez różne osoby,
- różne cele przyświecające poszczególnym członkom zespołu projektowego.

W zależności od rodzaju projektu, dziedziny w której jest on realizowany, wiedzy i umiejętności osób definiujących wymagania itp., zbiór potencjalnych ograniczeń może być różny. Jednakże w większości przypadków, bardzo wiele zależy od indywidualnego podejścia do rozpatrywanego

[5]Software Requirements Management Tools: User Specifications, UML Use Case, User Stories, Business Analysis, dostępne online: `http://www.requirementsmanagementtools.com/`, 5 września 2015

[6]Volere Requirements Resources – Requirements Tools, dostępne online: `http://www.volere.co.uk/tools.htm`, 5 września 2015

problemu, przez osoby uczestniczące w procesie pozyskiwania wymagań, czyli od klienta oraz potencjalnego wykonawcy projektu. Bardzo często mówi się m.in. (Bartyzel, 2012), że klient to osoba która wie co chce, ale nie wie co jest mu potrzebne. Słowa te powinno się interpretować jako opis sytuacji, w której klient nie potrafi dokładnie i precyzyjnie określić własności jakie oczekiwanie przez niego rozwiązanie powinno posiadać. W konsekwencji dostarczany jest Mu produkt posiadający oczekiwane funkcjonalności, jednakże nie są one do końca realizowane tak jak by tego oczekiwał klient (niezrozumienie na poziomie wymagań pozafunkcjonalnych). Z drugiej strony, wykonawca projektu bardzo często uważa, że wie lepiej niż osoba zlecająca jego wykonanie, jak ma wyglądać końcowa postać projektowanego rozwiązania i próbuje narzucać swoją wizję rozwiązania. Konieczne jest zatem minimalizowanie wpływu tego rozdźwięku na proces pozyskiwania wymagań. Saiedian i Dale (2000) jako możliwe rozwiązanie wymieniają m.in. efektywną komunikację, prototypowanie oraz graficzną reprezentację rozpatrywanego projektu.

Bardzo często przy pozyskiwaniu wymagań zdarza się sytuacja, w której po zdefiniowaniu pewnego wymagania pojawia się myśl, którą można wyrazić za pomocą następujących słów *ok, fajnie to wygląda ale czy na pewno da się to tak zrobić, czy może łatwiej nie byłoby abyśmy....* W tym momencie pojawia się wątpliwość czy opracowane wymaganie jest naprawdę konieczne i czy nie można by go zastąpić jakimś innym, które będzie prostsze w realizacji. Skutkiem takiego podejścia jest odrzucenie czasem cennego wymagania i zastąpienie go innym, który według osób definiujących wymagania jest łatwiejszy w realizacji ale niekoniecznie oddaje to czego oczekuje się od systemu. Taka sytuacja nazywana jest tzw. *syndromem tak, ale* (Sommerville, 2003).

Definiując kolejne wymagania pojawia się zasadnicze pytanie, kiedy zbiór posiadanych w danej chwili wymagań, można uznać za wystarczający i zakończyć proces pozyskiwania wymagań. Nie ma jednoznacznej odpowiedzi na to pytanie a definiowaniu kolejnych wymagań często towarzyszy przekonanie, które można wyrazić słowami: *Im więcej znajdziesz, tym wiesz, że więcej pozostaje do znalezienia.* Sytuacja ta w żargonie informatyków określana jest problemem *nieodkrytych ruin* (Sommerville, 2003).

4.6. Podsumowanie

Przedsięwzięcia informatyczne bardzo często doświadczają tzw. syndromu LOOP (ang. *Late, Over budget, Overtime, Poor quality*). Oprogramowanie jest zwykle oddawane po terminie, bardzo często budżet jest znacznie przekroczony, programiści pracują w nadgodzinach i dodatkowo finalne oprogramowanie, jest kiepskiej jakości, z wieloma błędami bądź nie do końca spełnia oczekiwania klienta. Jedną z przyczyn syndromu LOOP jest problem w komunikacji pomiędzy klientem a wykonawcą, wynikiem czego jest niedokładny opis tego, co ma być wynikiem realizacji danego projektu. Wygodnym i pomocnym rozwiązaniem są zbiory wymagań służące do opisu projektowanych rozwiązań.

Chociaż początki inżynierii wymagań są ściśle związane z inżynierią oprogramowania, to obecnie jej elementy są szeroko wykorzystywane w wielu innych dziedzinach, w ramach realizacji wielu różnych projektów. Prawidłowe przeprowadzenie procesu pozyskiwania wymagań, jest jednym z wymogów potrzebnych do poprawnego wykonania danego projektu. W wielu przypadkach, pozwalają one na rozstrzygnięcie sporów pomiędzy zleceniodawcą a wykonawcą projektu odnośnie tego, co miało być zrealizowane w ramach danego projektu. Na ich podstawie możliwe jest w wielu przypadkach ustalenie, która ze stron musi ponieść dodatkowe koszty związane z koniecznością przeprowadzenia dodatkowych działań związanych np. z wprowadzeniem niezbędnych poprawek, dodaniem dodatkowych funkcjonalności itp.

Rozdział 5

Metody i techniki sztucznej inteligencji w diagnostyce

W rozdziale umieszczono przegląd wybranych zagadnień związanych ze sztuczną inteligencją, w szczególności metody i techniki sztucznej inteligencji ściśle związane z tematyką rozprawy doktorskiej oraz z opracowanym podejściem projektowym do zagadnienia opracowywania projektu systemu diagnostycznego. Omówiono m.in. metody pozyskiwania oraz reprezentacji wiedzy diagnostycznej, systemy doradcze itp.

5.1. Idea sztucznej inteligencji

Fenomen ludzkiej inteligencji analizowany jest od wielu stuleci. Mimo ogromnego postępu nauki, dokładne stwierdzenie czym tak naprawdę jest inteligencja jest bardzo trudne. Według Słownika języka polskiego[1], jest to *zdolność rozumienia otaczających sytuacji i znajdowania na nie właściwych, celowych reakcji.* W psychologii uznaję się, że inteligencja to m.in. *zdolność rozwiązywania problemów* (J. Piaget[2]), *umiejętność dostrzegania zależności, relacji i wyciągania wniosków* (Ch. Spearman[3]) oraz *zdolność przystosowania się do okoliczności, dzięki dostrzeganiu abstrakcyjnych relacji, korzystaniu z uprzednich doświadczeń* (E. Nęcka[4]).

W 1955 roku J. McCarthy[5] sformułował pojęcie tzw. *sztucznej inteligencji*, jako procesu w wyniku którego, *mogą zostać opracowane maszyny zdolne symulować swoim działaniem ludzką inteligencję* (McCarthy i in., 1955). Wydarzenie to można uznać za początek rozwoju nowej dziedziny nauki, której przedmiotem prowadzonych badań, jest poznanie reguł rządzących inteligentnymi zachowaniami człowieka oraz innych organizmów żywych (roje owadów, ławice ryb itp.) a w konsekwencji opracowywanie odpowiednich algorytmów oraz maszyn naśladujących te zachowania. Abstrahując od różnego rodzaju krytycznych spostrzeżeń; m.in. (Hawking i in., 2014), mówiących o potencjalnych zagrożeniach związanych z rozwojem sztucznej inteligencji,

[1] Red. prof. W. Doroszewski, `http://sjpd.pwn.pl/haslo/inteligencja/`, dostępne online: 5 września 2015.

[2] Jean Piaget (1896-1980) – szwajcarski psycholog, biolog i epistemolog, autor teorii rozwoju poznawczego.

[3] Charles Spearman (1863–1945) – psycholog angielski, pionier analizy czynnikowej oraz autor korelacji rangowej Spearmana.

[4] Edward Nęcka (ur. 1953) – polski psycholog, twórca tzw. formalnej koncepcji inteligencji.

[5] John McCarthy (1927–2011) – informatyk amerykański, laureat Nagrody Turinga w 1971 r., autor języka programowania LISP.

należy podkreślić że ma ona i w dalszym ciągu będzie miała (wydaje się, że nawet jeszcze większy), ogromny wpływ na różne aspekty ludzkiego życia.

Wśród najważniejszych działów sztucznej inteligencji, wymienić obecnie można m.in.: sztuczne sieci neuronowe, algorytmy ewolucyjne, inteligencję rojową, logikę rozmytą, systemy doradcze itp. Problematyka sztucznej inteligencji, poszczególnych metod i technik oraz możliwych zastosowań jest szeroko poruszana w licznych, przekrojowych publikacjach, wśród którym jako najważniejsze można wymienić m.in. (Jain i Martin, 1998; Negnevitsky, 2005; Russell i Norvig, 2009; Rutkowski, 2012).

W obszarze diagnostyki technicznej, elementy sztucznej inteligencji wykorzystywane są przede wszystkim do:

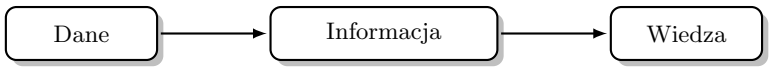
- pozyskiwania wiedzy diagnostycznej,
- zapisu wiedzy diagnostycznej,
- opracowywania modeli diagnostycznych,
- analizy zarejestrowanych sygnałów,
- wnioskowania o stanie technicznym obiektów.

W kolejnych podrozdziałach opisano wybrane metody sztucznej inteligencji związane z pozyskiwaniem i zapisem wiedzy diagnostycznej, systemami doradczymi oraz metodami wnioskowania.

5.2. Wiedza diagnostyczna oraz metody jej pozyskiwania

Termin wiedza jest pojęciem trudnym do zdefiniowania. Zgodnie z klasyczną definicją podaną przez Platona w dialogu *Teajtet* jest to *prawdzie, uzasadnione przekonanie*. W psychologi poznawczej przyjmuje się, że wiedza jest *formą trwałej reprezentacji rzeczywistości, mającą postać uporządkowanej i wzajemnie powiązanej struktury informacji, kodowanej w pamięci długotrwałej* (Nęcka i in., 2007). Natomiast Nowa Encyklopedia Powszechna PWN definiuje wiedzę jako *ogół wiarygodnych informacji o rzeczywistości wraz z umiejętnością ich wykorzystywania.*

Można zauważyć (Turban i in., 2005), że istnieje ścisła zależność (rys. 5.1) pomiędzy danymi, informacją oraz wiedzą. Najogólniej rzecz ujmując, dane są surowymi faktami (przykładowo dane liczbowe) dotyczącymi pewnych obiektów lub zdarzeń. W wyniku ich analizy generowane są informacje, czyli spostrzeżenia co do charakteru zgromadzonych danych (np. wzrost temperatury czynnika roboczego). Natomiast wiedza to zbiór przyjętych informacji pozwalających na generowanie pewnych wniosków i spostrzeżeń o rzeczywistości będącej przedmiotem analizy (np. przy wzroście temperatury czynnika roboczego powyżej wartości dopuszczalnej, istnieje zagrożenie uszkodzenia instalacji).



Rys. 5.1: Schemat zależności pomiędzy danymi, informacją oraz wiedzą

W obszarze diagnostyki technicznej wykorzystywana jest wiedza diagnostyczna, która rozumiana jest jako zbiór informacji o rozpatrywanych obiektach np. maszynach, procesach technologicznych itp. Pozwala ona na przeprowadzenie procesu wnioskowania o stanie technicznym rozpatrywanych obiektów. Zasadniczo wyróżnia się (Moczulski, 1997) wiedzę diagnostyczną dwóch typów:

- wiedza deklaratywna – opisuje znane fakty i cechy rozpatrywanych obiektów oraz relacje zachodzące pomiędzy poszczególnymi obiektami,
- wiedza proceduralna – opisuje zbiór działań niezbędnych do wykonania w celu realizacji określonych zadań, np. kolejne kroki eksperymentu diagnostycznego.

Pozyskiwanie wiedzy diagnostycznej w wielu przypadkach nie jest zadaniem łatwym i wiąże się z koniecznością realizacji szeregu uporządkowanych działań, których efektem końcowym powinna być kompletna i spójna baza wiedzy. Wyróżnić można (Moczulski, 1997), dwie główne grupy metod pozyskiwania wiedzy diagnostycznej:

- metody manualne (wywiady, obserwacje, dyskusje z ekspertem, kwestionariusze itp.),
- metody wspomagane komputerowo (wybrane algorytmy sztucznej inteligencji, data mining).

Metody manualne stosowane są do pozyskiwania zarówno wiedzy deklaratywnej jak i proceduralnej. W dużej mierze opierają się one, na wiedzy posiadanej przez specjalistów (ekspertów) dziedzinowych. Dzięki wieloletniej pracy zawodowej, posiadają oni odpowiedni zasób informacji pozwalający na formułowanie stosownych zbiorów wiedzy dotyczących analizowanych obiektów. Typowy przebieg procesu pozyskiwania wiedzy od ekspertów, może przebiegać w następujący sposób (Pawletko, 2007):

- opracowanie kwestionariusza,
- dobór grupy ekspertów,
- pozyskanie wiedzy od ekspertów poprzez wypełnienie przez nich kwestionariusza,
- agregacja otrzymanych wyników.

Wiedza deklaratywna może być również pozyskiwana z baz danych, zawierających wyniki przeprowadzonych pomiarów, obserwacji, symulacji działania obiektów, czy też zawierających informacje o uszkodzeniach i niesprawnościach wybranych obiektów zgłaszane przez klientów itp. Proces ten, nazywany również jako *data mining*, może być realizowany (Pieczyński, 2003) z wykorzystaniem metod uczenia maszynowego lub jako odkrywanie zależności w bazach danych. Metody uczenia maszynowego stosowane są dla danych wstępnie sklasyfikowanych. Można wyróżnić m.in. (Moczulski, 2001) selektywną indukcję reguł poprzez generowanie pokryć, indukcję reguł z zastosowaniem zbiorów przybliżonych oraz indukcję drzew decyzyjnych. Odkrywanie wiedzy w bazach danych (ang. *Knowledge Discovery in Databases*) (Frawley i in., 1992) opisuje się natomiast jako *nietrywialny proces identyfikacji ważnych, wcześniej nieznanych i potencjalnie użytecznych informacji ze zbioru danych.* Polega on na wyszukiwaniu pewnych regularności w posiadanym zbiorze danych. Regularność (Żytkow i Zembowicz, 1993), definiowana jest jako pewien wzorzec występujący w zbiorze danych. Przykładami wzorców mogą być tablice kontyngencji oraz równania i równoważności logiczne. Podczas pozyskiwania wiedzy z baz danych, konieczne jest wykonanie następujących czynności (Moczulski, 2002):

- gromadzenie zbioru danych wejściowych poprzez przeprowadzenie eksperymentu diagnostycznego na rzeczywistym obiekcie technicznym lub poprzez wykorzystanie symulatora,
- wyznaczenie relewantnych cech sygnałów,
- opracowanie klasyfikatora i jego weryfikacja,
- zastosowanie klasyfikatora do pozyskiwania wiedzy diagnostycznej np. w postaci reguł diagnostycznych.

Wiedza diagnostyczna może być, a w wielu przypadkach powinna być, pozyskiwana z wielu źródeł, celem uchwycenia wszelkich jej niuansów. Niezmiernie ważnym zagadnieniem jest integracja pozyskiwanej wiedzy, z uwagi na fakt że wiedza pozyskana z wielu źródeł, może być niedokładna, niepewna a w wielu przypadkach także sprzeczna. Koniecznie jest zatem zastosowanie odpowiednich technik weryfikacji i walidacji pozyskiwanej wiedzy. Ich zastosowanie ma na celu (Botten i in., 1989; Antoniou i in., 1998; Preece, 2001) m.in.: usunięcie sprzeczności pomiędzy wybranymi fragmentami wiedzy, doprecyzowanie wybranych obszarów wiedzy, usunięcie wiedzy nadmiarowej oraz powtarzającej się, ujednolicenie stosowanych pojęć itp.

5.3. Metody reprezentacji wiedzy

Pozyskiwana wiedza diagnostyczna celem, dalszego jej stosowania, musi być odpowiednio zapisywana. Sposób zapisu powinien być prosty, kompletny, zwięzły, zrozumiały i jednoznaczny, zarówno dla inżynierów wiedzy dokonujących jej zapisu jak i dla późniejszych użytkowników korzystających ze zgromadzonej wiedzy. Użytkownikami mogą być zarówno ludzie jak i programy komputerowe, stosowane w procesach automatycznego wnioskowania. Wybór sposobu reprezentacji wiedzy, zależy głównie od rodzaju wiedzy, dyscypliny której dotyczy wiedza, wymaganej wielkości bazy wiedzy oraz końcowego użytkownika. Wyróżnić można m.in. zapis naturalny, symboliczny, graficzny.

W kolejnych podrozdziałach opisano wybrane metody reprezentacji wiedzy deklaratywnej, która to w głównej mierze będzie wykorzystywana w niniejszej rozprawie. Wiedza proceduralna może być natomiast zapisywana np. w postaci wielowarstwowych schematów blokowych (Wyleżoł, 2000).

5.3.1 Reguły

Fakty i spostrzeżenia na temat rozpatrywanych obiektów technicznych oraz rozpatrywanej dziedziny można zapisać w postaci zdań logicznych, którym na podstawie zewnętrznych procesów np. obserwacji, eksperymentów, pomiarów itp., można przyporządkować wartość logiczną 1 lub 0, mówiącą o ich prawdziwości. Wybrane zdania logiczne mogą być łączone razem w reguły o postaci:

Jeżeli *przesłanka* **to** *konkluzja*, (5.1)

gdzie *przesłanka* jest zdaniem logicznym o znanej wartości, na podstawie którego wnioskuje się o prawdziwości zdania będącego *konkluzją*. Przykładowa reguła pozwalająca na wnioskowanie o stanie technicznym łożyska, może mieć następującą postać:

Jeżeli przekroczone są dopuszczalne wartości drgań względnych w węźle łożyskowym (5.2)
to uszkodzona jest bieżnia łożyska.

Przesłanka reguły może być rozszerzona do wyrażenia złożonego ze zdań prostych połączonych funktorami logicznymi **i/oraz**, **lub**, przykładowo:

Jeżeli przekroczone są dopuszczalne wartości drgań względnych w węźle łożyskowym (5.3)
oraz w widmie pojawia się składowa 1X
to uszkodzona jest bieżnia łożyska.

Reguły pozwalają na zapis wiedzy przyczynowej. Powszechnie stosowane są do opracowywania baz wiedzy tzw. systemów opartych na wiedzy (ang. *rule-based system*) (Ligęza, 2006).

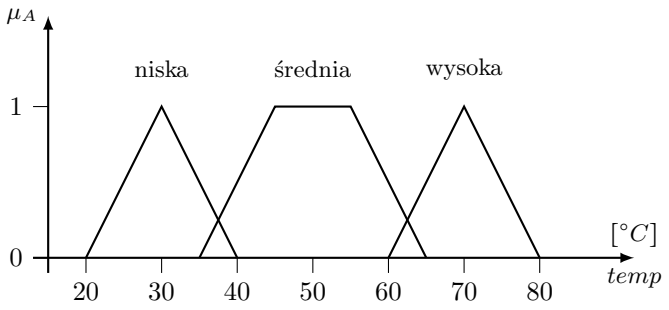
5.3.2 Zbiory rozmyte

Zbiorem rozmytym (Zadeh, 1965) A w pewnej przestrzeni X jest zbiór uporządkowanych par o postaci:

$$A = \{(x, \mu_A(x)); x \in X\} \tag{5.4}$$

gdzie $\mu_A(x)$ jest funkcją przynależność elementu x do zbioru rozmytego A. W odróżnieniu od klasycznej logiki dwuwartościowej, funkcja przynależności przyjmuje dowolne wartości z przedziału $\langle 0, 1\rangle$. W praktyce możliwe jest stosowanie dowolnych funkcji przynależności, natomiast zazwyczaj stosuje się funkcje trapezowe, trójkątne oraz s-funkcje.

Ze zbiorami rozmytymi związane są pojęcia zmiennej lingwistycznej oraz wartości zmiennej lingwistycznej. Zmienna lingwistyczna to wyrażona w języku naturalnym lub sztucznym pewna wielkość, dla której przyporządkowywane są wartości jakościowe (najczęściej zapisywane w języku naturalnym np. małe, duże itp.) będące wartościami zmiennej lingwistycznej. Przyjmując, że zmienna lingwistyczna ma postać *temperatura w węźle łożyskowym*, możliwe jest przyporządkowanie jej wartości lingwistycznych *niska, średnia, wysoka*. Przykładowe zbiory rozmyte dla takiego przypadku pokazano na rys. 5.2.



Rys. 5.2: Temperatura w węźle łożyskowym wyrażona za pomocą zbiorów rozmytych

W oparciu o zbiory rozmyte możliwe jest opracowywanie reguł rozmytych o postaci:

Jeżeli *a* **jest równe** *A1* **oraz** *b* **jest równe** *B2* **to** *c* **jest równe** *C1*, (5.5)

gdzie *a, b, c* są zmiennymi rozmytymi, natomiast *A1, B2, C1* są wartościami lingwistycznymi (jakościowymi) tych zmiennych. Przykładowa reguła rozmyta może mieć zatem postać:

Jeżeli *temperatura* **jest** *wysoka* **oraz** *drgania* **są** *podwyższone* (5.6)
to *łożysko* **jest** *uszkodzone*,

Reguły rozmyte z powodzeniem mogą być stosowane do reprezentacji wiedzy niepewnej i niedokładnej, np. dla potrzeb baz wiedzy systemów doradczych (Siler i Buckley, 2005).

5.3.3 Sieci stwierdzeń

Informacje o rozpatrywanym obiekcie technicznym mogą być zapisywane z wykorzystaniem stwierdzeń, interpretowanych jako para:

$$s = \langle c(s), b(s) \rangle, \tag{5.7}$$

gdzie $c(s)$ jest treścią stwierdzenia a $b(s)$ jest przyporządkowaną jej wartością. Treść stwierdzenia jest wypowiedzą orzekającą o zaobserwowanych faktach, bądź też wyrażającą pewną opinię. Wypowiedzi tej można przyporządkować pewną wartość, najczęściej jest to jeden z elementów zbioru $\{Prawda, Fałsz\}$, mówiącą o jej prawdziwości. Przykładem stwierdzenia może być sformułowanie o treści: *Złamany ząb, czynnego koła zębatego przekładni PZ02*, której to można przyporządkować wartość logiczną *Prawda*.

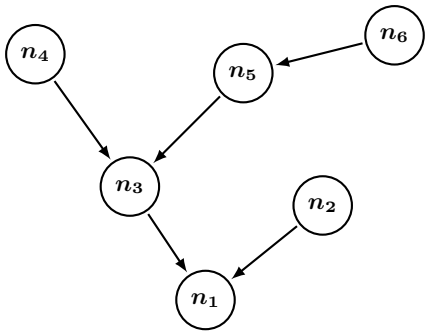
Wykorzystując pewien zbiór zdefiniowanych stwierdzeń, będących względem siebie w pewnym stosunku przyczynowej zależności, można zbudować graf G opisany jako:

$$G = \langle V, E \rangle, \tag{5.8}$$

w którym V jest niepustym i skończonym zbiorem wierzchołków tego grafu, a E jest skończonym zbiorem krawędzi skierowanych. Poszczególne wierzchołki grafu G są stwierdzeniami, natomiast każda krawędź $e \in E$ zapisana w postaci:

$$e = \langle t, h \rangle, \tag{5.9}$$

gdzie $t \in V$ jest wierzchołkiem początkowym krawędzi, a $h \in V$ wierzchołkiem końcowym krawędzi, odwzorowuje istniejące relacje pomiędzy wybranymi stwierdzeniami. Zdefiniowany w ten sposób graf G nazywany jest siecią stwierdzeń i jest jednym z rodzai modeli sieciowym (ang. *graphical models*). Wierzchołki grafu G, bardzo często nazywane są węzłami sieci stwierdzeń. Sieci stwierdzeń obrazują posiadaną wiedzę o rozpatrywanym problemie. Przykładem może być sieć stwierdzeń, prezentująca fragment wiedzy związanej z diagnostyką pompy wirnikowej pokazana na rys. 5.3.



Rys. 5.3: Przykład sieci stwierdzeń obrazującej fragment posiadanej wiedzy, związanej z diagnostyką pompy wirnikowej; n_1 – pompa działa poprawnie, n_2 – drgania podczas pracy pompy, n_3 – zbyt mała wydajność pompy, n_4 – silnie zapowietrzone medium, n_5 – zbyt mała prędkość obrotowa, n_6 – zbyt gęste medium

Dla potrzeb budowania sieci stwierdzeń możliwe jest zrezygnowanie z rozpatrywania, klasycznego, dwuelementowego zbioru możliwych wartości każdego stwierdzenia, na rzecz pewnego

zbioru, dopuszczalnych wartości:

$$b(s) \in \langle 0, 1 \rangle. \tag{5.10}$$

Podzbiór {*Prawda, Fałsz*} jest szczególnym przypadkiem takiego podejścia, w którym stwierdzeniu przyporządkowane są wartości odpowiednio 1 i 0. Stwierdzenia przyjmujące wartość różną od 0 (stwierdzenie nieprawdziwe) lub 1 (stwierdzenie prawdziwe) nazywane są stwierdzeniami przybliżonymi. Dopuszcza się przyjęcie jako wartość stwierdzenia, pewnego zbioru opisanego przedziałem:

$$\langle b_{min},\ b_{max} \rangle. \tag{5.11}$$

Przedział ten można interpretować jako informację o posiadaniu niedokładnej wartości punktowej danego stwierdzenia, ale ograniczonej tym przedziałem. Należy zauważyć, że można wyróżnić dwa skrajne przypadki, w pierwszym z nich:

$$b_{min} = 0,\ b_{max} = 1, \tag{5.12}$$

czyli nie jest posiadana żadna informacja o prawdziwość stwierdzenia, oraz drugi przypadek gdy:

$$b_{min} = b_{max}, \tag{5.13}$$

czyli posiadana jest przybliżona (ale ograniczona do pojedynczej wartości), informacja o wartości danego stwierdzenia.

Relacje pomiędzy wybranymi węzłami sieci stwierdzeń, reprezentowane przez stosowne krawędzie sieci, mogą należeć do różnych klas. Powszechnie stosowane są sieci Bayesa, nazywane również sieciami przekonań (ang. *belief networks*), w których zależności wyrażone są w postaci tablic prawdopodobieństw warunkowych CPT (ang. *Conditional Probability Table*). Opisują one brzegowy rozkład prawdopodobieństwa dowolnego zdarzenia X_i (np. węzeł n_1 na rys. 5.3), w relacji do zbioru zdarzeń Y_i (węzły n_3 i n_2 na rys. 5.3), od których to zdarzenie X_i jest zależne (Jensen, 1996; Koski i Noble, 2009).

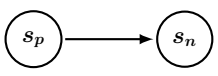
Drugą z możliwość jest wykorzystanie warunków koniecznych i dostatecznych (Cholewa, 2008). Stwierdzenie s_p jest uznawane za warunek dostateczny dla stwierdzenia s_n jeżeli przekonaniu o prawdziwości stwierdzenia s_p towarzyszy przekonanie o prawdziwości stwierdzenia s_n. Równocześnie stwierdzenia s_n jest uznawane za warunek konieczny dla stwierdzenia s_p. Zakładając, że:

$$b(s_p) \in \langle 0, 1 \rangle,\ b(s_n) \in \langle 0, 1 \rangle, \tag{5.14}$$

to informację o tym że stwierdzenie s_p jest warunkiem dostatecznym dla stwierdzenia s_n, można zapisać w postaci:

$$b(s_p) \leq b(s_n). \tag{5.15}$$

Warunki konieczne i dostateczne mogą być reprezentowane w postaci grafów skierowanych (rys. 5.4), w których zwrot gałęzi wskazuje rosnące wartości stopnia przekonania.



Rys. 5.4: Graf reprezentujący warunek konieczny i dostateczny

Relacje pomiędzy poszczególnymi węzłami rozpatrywanych sieci stwierdzeń, mogą wyrażać różne aspekty wiedzy (Skupnik, 2009) np. aspekt struktury obiektu, stanów funkcjonalnych, harmonogramu działania elementów obiektu itp., obowiązujące dla różnych warunków i zało-

żeń, przy jednoczesnym występowaniu różnych kontekstów (Timofiejczuk, 2011, 2013). Próba uwzględnienia wszystkich możliwych zależności na jednej, globalnej sieci stwierdzeń może prowadzić do zbytniego jej rozrostu. Konsekwencją takiego podejścia może być znaczna trudność w trakcie analizy opracowanej w ten sposób sieci. Rozwiązaniem alternatywnym jest wykorzystanie koncepcji sieci wielomodalnej (Heath i Sioson, 2009), na podstawie której możliwe jest (Cholewa, 2010a) zdefiniowanie wielomodalnej sieci stwierdzeń MSN (ang. *Multimodal Statement Network*) o postaci:

$$MSN = \langle V, E, \Gamma \rangle, \tag{5.16}$$

gdzie V i E są zbiorami odpowiednio węzłów i krawędzi takiej sieci, natomiast Γ jest zbiorem mod tej sieci. Mody sieci interpretowane są jako pojedyncze sieci stwierdzeń zapisywane jako:

$$\Gamma_i = \langle V_i, E_i \rangle, \tag{5.17}$$

przy czym $V_i \subset V$, $E_i \subset E$. Cechą charakterystyczną wielomodalnych sieci stwierdzeń jest występowanie węzłów wspólnych pomiędzy wybranymi modami/sieciami składowymi. Sieci wielomodalne mogą być reprezentowane w postaci wielowarstwowych sieci stwierdzeń (rys. 5.5). Niewątpliwą zaletą stosowania wielomodalnych sieci stwierdzeń jest możliwość definiowania poszczególnych sieci składowych przez niezależne osoby, np. ekspertów dziedzinowych, na podstawie wspólnego zbioru stwierdzeń.

Sieci stwierdzeń umożliwiają przeprowadzanie procesu wnioskowania, w którym na podstawie przesłanek (stwierdzeń/węzłów o znanej wartości) wyznacza się konkluzje (nieznane wartości pozostałych stwierdzeń/węzłów). Ważnym zagadnieniem jest dobór odpowiedniej metody wyznaczania wartości węzłów wspólnych. Powszechnie stosowanymi rozwiązaniami są metody agregacji (poszczególne sieci rozwiązywane są oddzielnie a końcowy wynik wyliczany jest za pomocą przyjętych statystyk matematycznych) oraz uzgadnianie (sieć ulega tzw. spłaszczeniu i wszystkie sieci składowe rozwiązywane są jednocześnie).

Wielomodalne sieci stwierdzeń, są wygodnym sposobem do reprezentacji wiedzy niepewnej i niedokładnej, dla potrzeb baz wiedzy systemów doradczych. Umożliwiają one dokładne uwzględnienie wszelkich aspektów wiedzy, pochodzącej z wielu źródeł (Cholewa i in., 2013) oraz jednocześnie mogą być rozpatrywane jako sieci dynamiczne (Cholewa, 2014). Sieci stwierdzeń mogą być wykorzystane do reprezentacji wiedzy dla potrzeb diagnostyki technicznej, przykłady zastosowań można znaleźć m.in. w (Cholewa i in., 2009; Rzydzik i in., 2014; Rogala i Amarowicz, 2015).

Wielomodalne sieci stwierdzeń mogą być budowane z wykorzystaniem oprogramowania opracowanego przez pracowników Instytutu Konstrukcji Maszyn Politechniki Śląskiej o nazwie *MMNET*[6] (Cholewa, 2010b; Cholewa i in., 2011a) oraz REx[7] (Cholewa i in., 2011b). Autor niniejszej rozprawy doktorskiej jest autorem wybranych fragmentów tych systemów m.in. (Amarowicz, 2010).

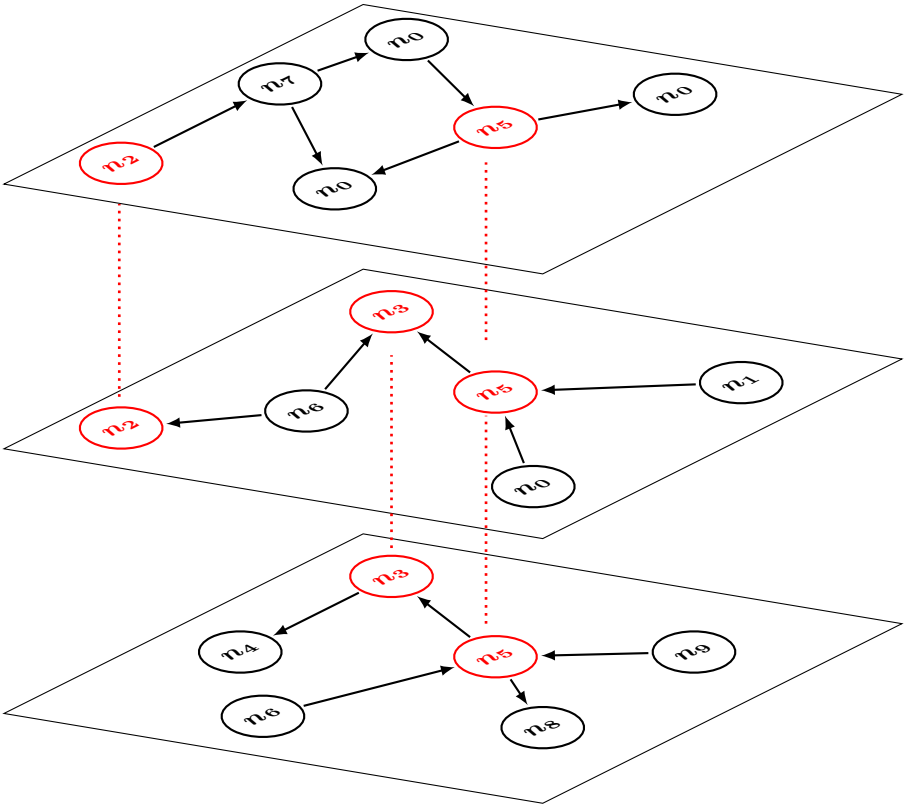
5.3.4 Ontologie i sieci semantyczne

Ontologia w sensie informatycznym[8] to wg (Gruber, 1993) *formalna i jawna specyfikacja wspólnej konceptualizacji*. Jest ona zatem formalną reprezentacją pewnej dziedziny wiedzy. Składa

[6]Szkieletowy system doradczy *MMNET*, strona domowa: `http://ipkm.polsl.pl/index.php?n=ProjMMNET.MMNETAbout`, dostępne online: 5 września 2015

[7]Szkieletowy system doradczy *REx*, strona domowa: `http://ipkm.polsl.pl/index.php?n=Projekty.Rex`, dostępne online: 5 września 2015

[8]W ramach filozofii ontologia rozpatrywana jest jako nauka o bycie.



Rys. 5.5: Przykład wielomodalnej sieci stwierdzeń

się na nią zarówno (Gómez-Pérez i in., 2007) zapis zbioru pojęć jak i relacji pomiędzy nimi. Wg (Maedche i Staab, 2001) każda ontologia może być definiowana jako:

$$O = \{C, R, Hc, Rel, A\}, \tag{5.18}$$

gdzie:

- C – zbiór wszystkich wykorzystanych pojęć w ramach rozpatrywanej ontologii,
- R – zbiór nietaksonomicznych relacji (zwanych właściwościami, slotami, rolami),
- Hc – zbiór taksonomicznych relacji pomiędzy pojęciami,
- Rel – zbiór nietaksonomicznych relacji pomiędzy pojęciami,
- A – zbiór aksjomatów, opisujących dodatkowe ograniczenia.

Interpretacje poszczególnych pojęć i relacji pomiędzy nimi, zapisywane są w leksykonie o postaci:

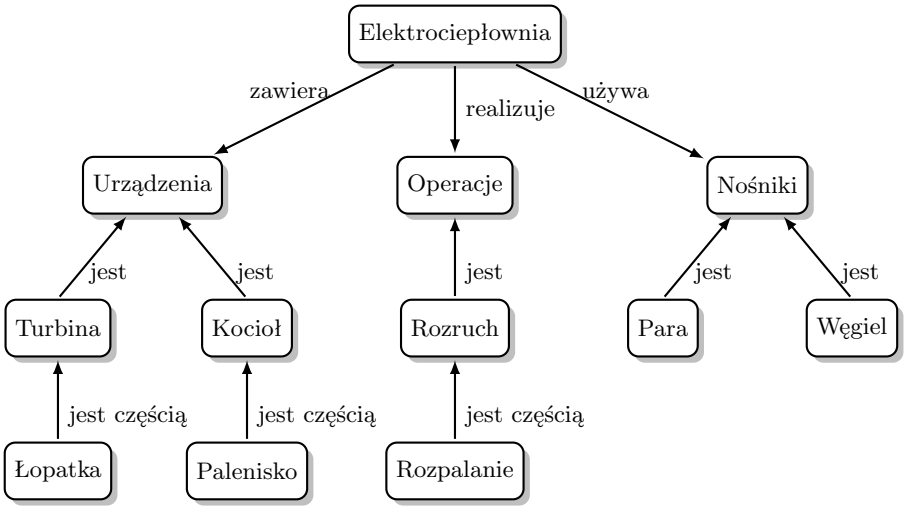
$$L = \{Lc, Lr, F, G\}, \tag{5.19}$$

gdzie:

- Lc – definicje zbioru pojęć,
- Lr – definicje zbioru relacji,
- F – referencje pojęć,
- G – referencje relacji.

Ontologie stosowane są do reprezentacji wiedzy związanej z wybranymi dziedzinami nauki. Na ich podstawie możliwe jest automatyczne wnioskowanie w ramach zgromadzonej wiedzy.

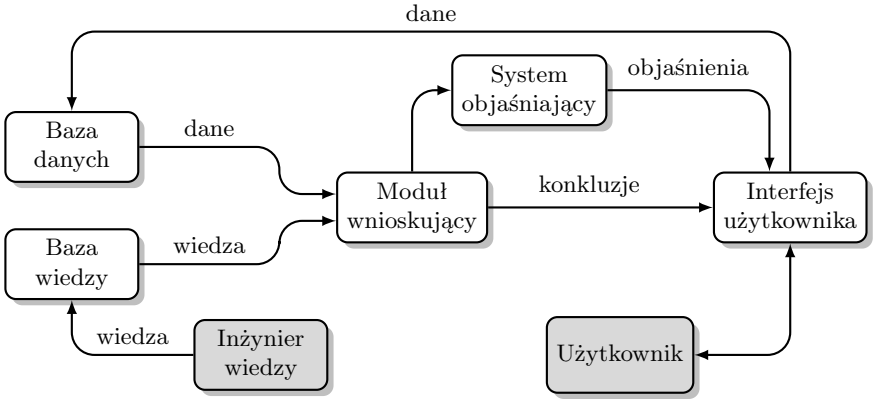
Wiedza reprezentowana poprzez ontologie może być przedstawiana za pomocą sieci semantycznych. W sensie matematycznym sieć semantyczna jest grafem, którego wierzchołkami jest pewien zbiór obiektów/pojęć związanych z określoną dziedziną. Natomiast istniejące relacje pomiędzy tymi elementami wyrażone są za pośrednictwem zdefiniowanych krawędzi. Przykładową sieć semantyczną prezentującą fragment posiadanej wiedzy, związanej z elektrociepłownią pokazano na rys. 5.6.



Rys. 5.6: Fragment wiedzy o elektrociepłowni zapisany w postaci sieci semantycznej (Pieczyński, 2003)

5.4. Systemy doradcze

Systemy doradcze (ang. *expert systems*), nazywane również systemami ekspertowymi lub eksperckimi, wprowadzone zostały w latach siedemdziesiątych XX wieku. Ich zadaniem jest jak najlepsze odwzorowanie procesu podejmowania decyzji, związanego z pewnym zagadnieniem, wykonywanego przez eksperta. Znajdują one szczególne zastosowanie, w przypadku złożonych problemów, gdy potrzebna jest analiza dużych zbiorów danych oraz konieczne jest uwzględnienie rozległej wiedzy (Cholewa i Pedrycz, 1987). Elementy składowe systemu doradczego pokazano na rys. 5.7.



Rys. 5.7: Elementy składowe systemu doradczego oraz przepływ informacji pomiędzy nimi

Podstawowym elementem każdego systemu doradczego jest baza wiedzy, zawierająca wiedzę związaną z rozpatrywanym zagadnieniem. Zazwyczaj jest ona zapisywana m.in. w postaci zbiorów reguł, sieci stwierdzeń oraz procedur. Wielkość bazy wiedzy zależy od rozpatrywanego problemu oraz dostępności wiedzy i np. dla bazy reprezentowanej w postaci reguł, waha się od kilkudziesięciu, poprzez kilkaset aż do kilku tysięcy reguł, np. w przypadku systemu *XCON*, przeznaczonego do kompletowania sprzętu komputerowego (Barker i O'Connor, 1989). Proces kompletowania i zapisu wiedzy dla potrzeb systemu doradczego jest zazwyczaj długotrwały oraz realizowany przez tzw. inżynierów wiedzy. Ważnym zagadnieniem jest integracja oraz weryfikacja pozyskiwanej wiedzy (Smith i Kandel, 1993). Aktualne informacje/fakty o rozpatrywanym problemie gromadzone są w bazie danych. W oparciu o zgromadzoną wiedzę oraz aktualnie posiadane dane moduł wnioskujący stara się wyznaczyć możliwe konkluzje. Wyróżnić można dwa podstawowe schematy wnioskowania:

- wnioskowanie w przód *(modus ponens)* – wnioskowanie na podstawie znanych danych, w czasie którego stara się udowodnić prawdziwość stawianych hipotez, tzw. wnioskowanie o konkluzjach na podstawie przesłanek,
- wnioskowanie wstecz *(modus tollens)* – polega na wykazaniu prawdziwości hipotezy (weryfikacja) na podstawie prawdziwości konkluzji, tzw. wnioskowanie o przesłankach na podstawie konkluzji.

Informacje o przebiegu procesu wnioskowania, w tym m.in. informacje o zastosowanych fragmentach wiedzy (tzw. uzasadnienie decyzji), dostarczane są poprzez system objaśniający. Dostęp do systemu doradczego (wprowadzanie danych, wyniki działania systemu) zapewniony jest za pośrednictwem interfejsu użytkownika (Mulawka, 1996; Liebowitz, 1997).

Dla potrzeb opracowania systemu doradczego dla wybranego problemu, możliwe jest wykorzystanie tzw. szkieletowych systemów doradczych (ang. *expert system shell*). Z technicznego punktu widzenia jest to system doradczy pozbawiony wiedzy. Po uzupełnieniu go wymaganym zbiorem wiedzy otrzymuje się w pełni funkcjonalny system doradczy. Przykładem szkieletowego systemu doradczego jest m.in. system CLIPS[9].

Pierwsze systemy doradcze (opracowane w latach 70 XX wieku) stosowane były m.in. do diagnozowania chorób (Mycin, (Buchanan i Shortliffe, 1984)), analizy struktury molekularnych nieznanych związków chemicznych (Dendral, (Lindsay i in., 1993)) oraz w poszukiwaniu złóż minerałów (Prospector, (Duda i in., 1979)). Obecnie systemy doradcze znajdują szerokie zastosowanie m.in. w medycynie do diagnozowania chorób (Smith, 1992), do sterowania pojazdami (Khalajzadeh i in., 2011), w projektowaniu obiektów technicznych (Pokojski, 2005; Tarełko, 2011), do analizy danych np. notowań giełdowych (Lee i Jo, 1999; Yunusoglu i Selim, 2013) itp.

W obszarze szeroko rozumianej diagnostyki technicznej, systemy doradcze stosowane są dla potrzeb diagnozowania różnego typu obiektów, np. maszyn wirnikowych (Cholewa i Psiuk, 2005), hydrogeneratorów (Amaya i Alvares, 2010), mikro-turbin parowych (Nan i in., 2008) itp. Historię rozwoju diagnostycznych systemów doradczych przedstawiono m.in. w (Angeli, 2010).

5.5. Podsumowanie

Metody i techniki sztucznej inteligencji stosowane są obecnie, w wielu różnych dziedzinach nauki. Pozwalają one na rozwiązywanie szeregu problemów, które to do tej pory, z racji swojej złożoności, były albo trudne do realizacji albo wręcz niemożliwe. Wybrane elementy sztucznej

[9]CLIPS – A Tool for Building Expert Systems, `http://clipsrules.sourceforge.net/`, dostępne online: wrzesień 2015

inteligencji z powodzeniem znalazły także zastosowanie w obszarze szeroko rozumianej diagnostyki technicznej. Z racji swej złożoności, szczególnie w kwestiach związanych z wnioskowaniem o stanie technicznym rozpatrywanych obiektów, dziedzina ta jest szczególnie adekwatna. Wybrane algorytmy sztucznej inteligencji pozwalają na skuteczniejsze wykrywanie wybranych uszkodzeń, odkrywanie nieznanych zależności pomiędzy wybranymi danymi oraz umożliwiają integrację wiedzy diagnostycznej pozyskiwanej z wielu źródeł.

Rozdział 6

Metoda projektowania systemów diagnostycznych z zastosowaniem zbiorów wymagań

W niniejszym rozdziale zawarto opis opracowanego przez autora podejścia projektowego, do problemu projektowania systemów diagnostycznych z wykorzystaniem zbiorów wymagań. Omówiono kolejne etapy proponowanego podejścia, sposób zapisu niezbędnych wymagań oraz proces ich pozyskiwania oraz oceny, celem wyznaczenia optymalnego rozwiązania projektu systemu diagnostycznego.

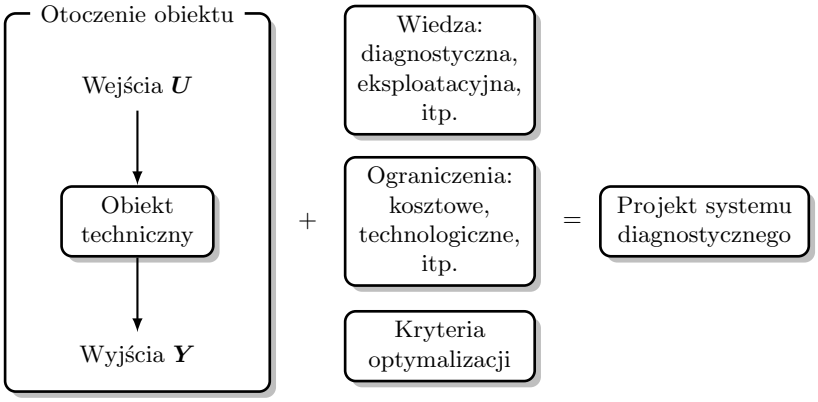
6.1. Założenia i uwagi wstępne

Dla potrzeb przedstawienia proponowanego podejścia projektowego do rozpatrywanego problemu, konieczne jest ustalenie pewnych warunków początkowych, definiujących analizowany problem badawczy.

Rozpatrywany jest (rys. 6.1) złożony obiekt techniczny O_t, realizujący pewien z góry określony w procesie projektowo-konstrukcyjnym zbiór operacji K, przekształcających szeroko rozumiane wejścia do obiektu U w wyjścia Y. W strukturze tego obiektu można wyróżnić szereg współpracujących ze sobą podzespołów oraz elementów. Obiekt ten, działa w pewnym otoczeniu opisanym przez wybrane parametry, np.: temperaturę, ciśnienie, wilgotność itp., definiującym jego warunki działania. Rozpatrywany obiekt techniczny, może znajdować się w jednym ze stanów technicznych s_i z ogólnego zbioru wszystkich możliwych stanów technicznych tego obiektu. Jednocześnie każdy ze stanów technicznych obiektu, zgodnie z (Korbicz i in., 2002) jest funkcją możliwych uszkodzeń tego obiektu, czyli uszkodzeń poszczególnych jego podzespołów i elementów składowych. Z eksploatacją tego obiektu związane jest pewne ryzyko techniczne:

$$R_t(O_t) = \sum_i c_i P_i, \tag{6.1}$$

które to powinno być minimalizowane podczas użytkowania obiektu technicznego. Proces ten może zostać zrealizowany poprzez odpowiednie monitorowanie stanu technicznego obiektu i niedopuszczanie do sytuacji obarczonych wysokim ryzykiem.



Rys. 6.1: Ilustracja rozpatrywanego problemu

Dla tak opisanego obiektu technicznego, możliwe jest sformułowanie potrzeby N zapisanej jako wyrażenie o treści: *Diagnozowanie aktualnego stanu technicznego rozpatrywanego obiektu, celem oceny potencjalnego ryzyka technicznego*. Potrzeba ta może zostać zrealizowana, poprzez zastosowanie pewnego systemu diagnostycznego *DS*, współpracującego z rozpatrywanym obiektem technicznym. Dla potrzeb niniejszych rozważań przyjmuje się, że jako projekt/rozwiązanie systemu diagnostycznego rozumiany będzie:

- zbiór funkcjonalności, opisujących konieczność rozpoznawania określonych stanów technicznych rozpatrywanego obiektu,
- zbiór podsystemów/elementów stosowanych do realizacji poszczególnych funkcjonalności.

Trudności, wymienione w rozdz. 1, związane z opracowaniem potencjalnych rozwiązań systemu diagnostycznego sprawiają, że istnieje konieczność zastosowania odpowiedniego podejścia projektowego, którego szczegóły przedstawiono w kolejnym podrozdziale.

6.2. Charakterystyka proponowanego podejścia projektowego

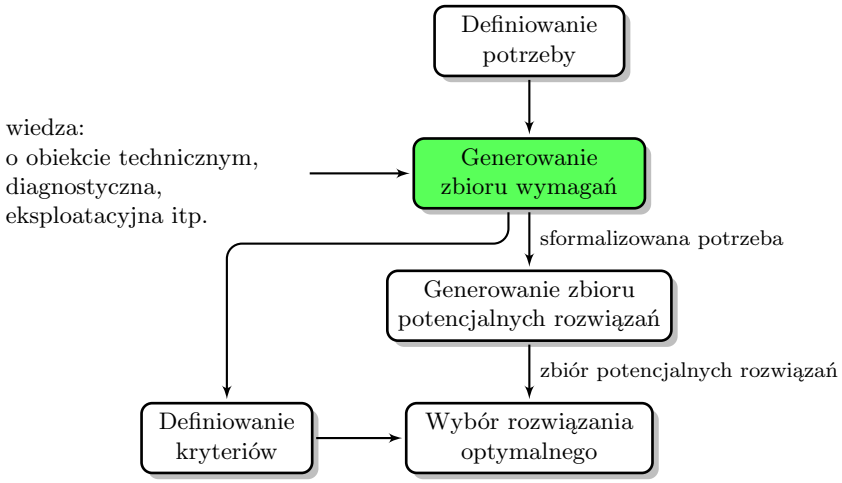
Jak wspomniano we wprowadzeniu do rozprawy, proces projektowania systemów diagnostycznych jest bardzo trudnym i złożonym zagadnieniem. Przede wszystkim z uwagi na fakt, że konieczne jest uwzględnienie w nim wielu różnych czynników, w tym informacji o możliwych uszkodzeniach oraz ich potencjalnych skutkach (analiza ryzyka) a także informacji o dostępnych metodach i technikach diagnostycznych, które mogą zostać zastosowane do wykrywania poszczególnych z uszkodzeń. Dużą rolę odgrywają także istniejące ograniczenia, przede wszystkim finansowe i technologiczne, które niejednokrotnie mają ogromny wpływ na postać projektowanego systemu diagnostycznego.

W tradycyjnym podejściu projektowym, np. podczas projektowania maszyn, na postawie zdefiniowanej potrzeby, opracowywany jest (z wykorzystaniem różnych technik np.: burza mózgów, praca grupowa, tablica morfologiczna itp.) zbiór możliwych rozwiązań projektowanego środka technicznego. Można zauważyć, że działanie takie nie do końca jest możliwe do zastosowania, w przypadku próby opracowania projektu systemu diagnostycznego. Spowodowane jest to przede wszystkim trudnością, w opracowaniu zbioru potencjalnych rozwiązań systemu diagnostycznego wprost na podstawie zdefiniowanej potrzeby, zapisanej w sposób czysto słowny. Próby rozwiązania tego problemu poprzez zastosowanie metod opartych na tablicy morfologicznej (Zwicky, 1969), w których rozpatrywane zagadnienie rozbijane jest na mniejsze problemy

i dla nich opracowywane są zbioru możliwych rozwiązań, po czym następuje integracja wszystkich rozwiązań cząstkowych, też nie mogą być bezpośrednio zastosowane, przede wszystkim z uwagi na problem w wyodrębnieniu zagadnień składowych oraz trudności w ich ocenie.

Dlatego też konieczne jest wprowadzenie pewnych modyfikacji do standardowego procesu projektowego, które umożliwią z jednej strony stosunkowo proste i łatwe opracowanie zbioru potencjalnych rozwiązań systemu diagnostycznego, przy zachowaniu jednoczesnej możliwości oceny tych rozwiązań przez pryzmat zdefiniowanych kryteriów. Zadanie to może zostać osiągnięte poprzez zapisanie potrzeby w pewien sformalizowany sposób. Zapis ten powinien zawierać w sobie, zbiór oczekiwanych funkcjonalności projektowanego systemu diagnostycznego oraz informacje o możliwych elementach struktury tego systemu.

W celu realizacji przedstawionych działań, możliwe jest użycie powszechnie stosowanych w obszarze inżynierii oprogramowania zbiorów wymagań. Jak to już zostało opisane w rozdz. 4, są one powszechnie stosowane do zapisu oczekiwanych funkcjonalności projektowanych systemów informatycznych. W rozpatrywanym przypadku mogą one (po wprowadzeniu pewnych modyfikacji opisanych w rozdz. 6.3) zostać użyte, do zapisu oczekiwanych funkcjonalności projektowanego systemu diagnostycznego, a tym samym do wyrażenia potrzeby w sformalizowany sposób. Do standardowego procesu projektowego, należy zatem wprowadzić dodatkowy etap (rys. 6.2), polegający na generowaniu zbioru wymagań opisujących oczekiwane funkcjonalności projektowanego systemu diagnostycznego. W ogólnym przypadku kolejne etapy, które są konieczne do



Rys. 6.2: Proces projektowania systemu diagnostycznego

wykonania w ramach proponowanego podejścia projektowego, obejmują:

- zdefiniowanie potrzeby,
- definiowanie zbioru wymagań, opisującego oczekiwane funkcjonalności projektowanego systemu diagnostycznego, celem formalizacji potrzeby,
- generowanie zbioru potencjalnych rozwiązań systemu diagnostycznego,
- proces oceny zbioru wymagań oraz zbioru potencjalnych rozwiązań, celem wyodrębnienia rozwiązania optymalnego.

Poszczególne wymagania mogą być definiowane na podstawie posiadanej wiedzy o obiekcie technicznym, wiedzy diagnostycznej, eksploatacyjnej itd. Niewątpliwą zaletą wykorzystania w procesie projektowym wymagań, jest możliwość definiowania poszczególnych wymagań przez niezależne osoby, tzw. ekspertów dziedzinowych. Ma to szczególne znaczenie w obszarze diagnostyki technicznej, z uwagi na to że współczesne obiekty techniczne są bardzo skomplikowane

i zaprojektowanie systemu diagnostycznego przez wąską grupę osób może być bardzo trudne. Dodatkowo definiowane wymagania, mogą posłużyć do wygenerowania zbioru kryteriów, które będą użyte w procesie oceny potencjalnych rozwiązań systemu diagnostycznego, celem wyodrębnienia rozwiązania optymalnego.

W kolejnych podrozdziałach omówiono szczegółowe aspekty związane z zastosowaniem zbiorów wymagań dla potrzeb projektowania systemów diagnostycznych, w tym sposób formalnego zapisu wymagań, metodę generowania zbioru potencjalnych rozwiązań systemu diagnostycznego a także sposób oceny zbiorów wymagań oraz potencjalnych rozwiązań systemu diagnostycznego.

6.3. Zapis zbioru wymagań

Dla potrzeb wykorzystania zbiorów wymagań do opisu oczekiwanych funkcjonalności systemu diagnostycznego, konieczne jest wprowadzenie pewnych dodatkowych założeń, które nie są spotykane w obszarze inżynierii oprogramowania. Założenia te (Amarowicz, 2014b, 2015), związane są z koniecznością przyjęcia pewnego specyficznego sposobu zapisu definiowanych wymagań, tak aby możliwe było uwzględnienie wszystkich niezbędnych informacji, potrzebnych do opracowania zbioru potencjalnych projektów systemu diagnostycznego.

Biorąc pod uwagę wymienione wcześniej uwagi i spostrzeżenia, każde wymaganie funkcjonalne req_i może być zapisane jako:

$$req_i =< e_{ot}, c_r, attr_r, pf_r, drule >, \tag{6.2}$$

gdzie:

- e_{ot} – element obiektu technicznego,
- c_r – treść wymagania,
- $attr_r$ – atrybuty wymagania,
- pf_r – stopień preferencji wymagania,
- $drule$ – zbiór reguł diagnostycznych.

Treść wymagania jest słownym opisem funkcjonalności systemu diagnostycznego, związanej z potrzebą rozpoznawania określonego stanu technicznego obiektu[1], np. *konieczność wykrywania złamania zęba koła zębatego*. Do każdego wymagania, możliwe jest przypisanie pewnego zbioru atrybutów:

$$attr_r = \{attr_{r1}, attr_{r2}, ...\}, \tag{6.3}$$

które będą stosowane do dokładnego opisu danego wymagania oraz będą przydatne dla potrzeb procesu oceny zgromadzonych wymagań. Każdy atrybut $attr_i$ może być zapisany jako:

$$attr_i =< c_i, v_i, imp_i >, \tag{6.4}$$

gdzie:

- c_i – treść/nazwa atrybutu,
- v_i – wartość atrybutu,
- imp_i – stopień ważności atrybutu.

[1]Zgodnie z zależnością 2.4, stan techniczny obiektu jest funkcją uszkodzeń. Można zatem rozpatrywać treść wymagania, jako konieczność wykrywania określonych uszkodzeń lub zużyć obiektu.

Przykładami atrybutów mogą być np. *poziom redukcji/ograniczenia prawdopodobieństwa wystąpienia uszkodzenia opisanego przez wymaganie*, *poziom redukcji kosztów związanych z wystąpieniem danego zdarzenia* itp. Każdemu wymaganiu funkcjonalnemu, można przyporządkować pewną wartość z przedziału $< 0; 1 >$, nazywaną stopniem preferencji pf. Wyraża on konieczność uwzględnienia funkcjonalności opisanej przez dane wymaganie, w końcowym projekcie systemu diagnostycznego. Stopień preferencji może być traktowany jako subiektywna ocena przydatności funkcjonalności opisanej przez dane wymaganie, do osiągnięcia przez system diagnostyczny założonej funkcji celu, czyli minimalizacji ryzyka technicznego. Każde wymaganie funkcjonalne jest przypisywane do wybranego podzespołu/elementu obiektu technicznego e_{ot}, celem uporządkowania definiowanych wymagań i utworzenia hierarchicznego zbioru wymagań.

Do każdego wymagania funkcjonalnego, można przypisać pewien zbiór potencjalnych reguł diagnostycznych:

$$drule = \{drule_1, drule_2, ...\}, \tag{6.5}$$

które mogą być zastosowane, do rozpoznawania stanu technicznego opisanego przez dane wymaganie. Każda reguła diagnostyczna zapisywana jest jako:

$$drule_i =< c_{dr}, attr_{dr}, pf_{dr}, subs, ko >, \tag{6.6}$$

gdzie:

- c_{dr} – treść reguły,
- $attr_{dr}$ – atrybuty reguły,
- pf_{dr} – stopień preferencji reguły diagnostycznej,
- $subs$ – zbiór podsystemów diagnostycznych,
- ko – kryteria ograniczające dla podsystemów.

Analogicznie jak w przypadku opisu wymagania, c_{dr} jest opisem słownym reguły, np. *przekroczona wartość skuteczna drgań*, $attr_{dr}$ to zbiór możliwych atrybutów, za pomocą których można dokładnie opisać każdą z reguł np. *poziom fałszywych alarmów*, natomiast pf_{dr} to stopień preferencji przypisany do reguły. W przypadku reguł diagnostycznych, stopnień preferencji niesie informację o tym czy reguła diagnostyczna jest przydatna do wykrywania danego uszkodzenia, natomiast każdy z atrybutów zapisywany jest identycznie jak w przypadku atrybutów wymagania funkcjonalnego.

Każda z reguł diagnostycznych może być zrealizowana, poprzez zastosowanie tzw. podsystemu diagnostycznego, w skład którego wchodzą określone układy pomiarowe oraz algorytmy analizy i wnioskowania. Każdy podsystem $subs$ może być zapisany jako:

$$subs_i =< etype, v_{fk} >, \tag{6.7}$$

gdzie:

- $etype$ – zbiór typów elementów podsystemu,
- v_{fk} – wartość funkcji kryterialnej.

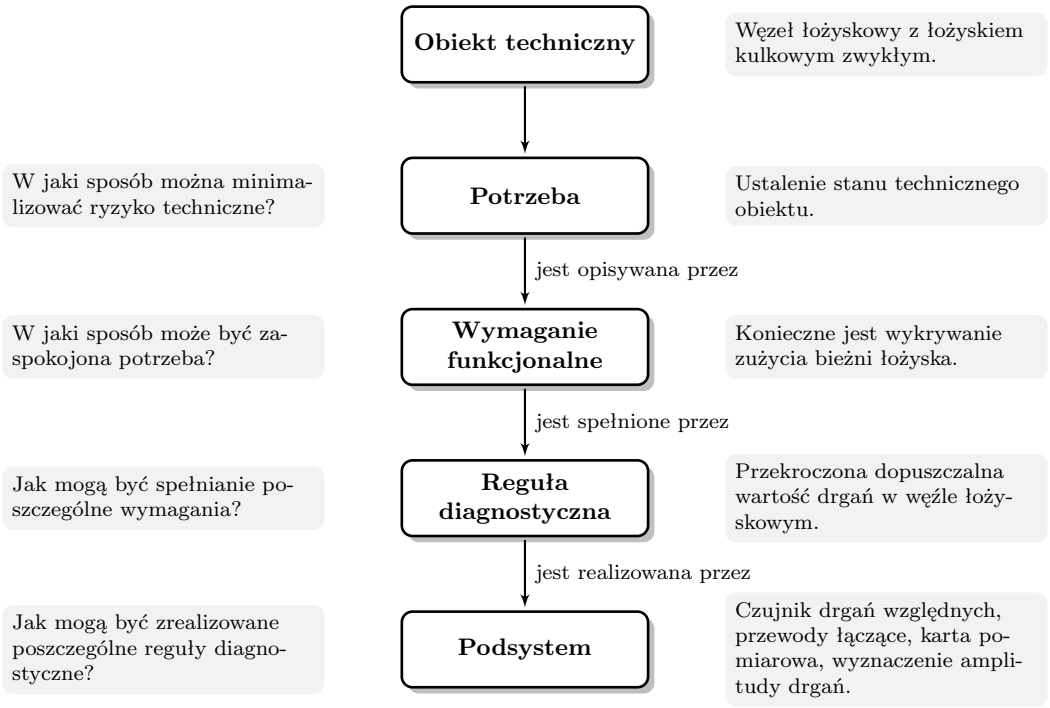
Jako typy elementów podsystemu rozumiane są możliwe do zastosowania kategorie czujników, kart pomiarowych, operacje analizy sygnałów itd. Przyjęcie takiej formy zapisu podsystemu diagnostycznego, pozwala na etapie formalizowania potrzeby projektowej, na opracowanie zbioru potencjalnych podsystemów diagnostycznych bez wnikania w dokładne ich szczegóły, tj. w kon-

kretne egzemplarze poszczególnych elementów. Dla potrzeb dalszych rozważań, każdy typ elementu podsystemu może być zapisany jako:

$$etype_i =< elem, attr_e >, \tag{6.8}$$

gdzie *elem* to zbiór możliwych egzemplarzy elementów danego typu, czyli np. konkretne czujniki z katalogów a $attr_e$ to zbiór atrybutów opisujących dany typ elementu, np. zakres pomiarowy, czułość, cena itd. Atrybuty te stosowane są w procesie optymalizacji projektu systemu diagnostycznego. Dla potrzeb ograniczania możliwości zastosowania wybranych elementów wchodzących w skład podsystemów, możliwe jest zdefiniowanie zbioru kryteriów ograniczających *ko*, przypisanych do poszczególnych reguł diagnostycznych $drule_i$. Poszczególnym podsystemom w procesie optymalizacji, przypisywana jest pewna wartość v_{fk}, będąca wynikiem zastosowania funkcji kryterialnej. Przykładowo poszukując rozwiązania cechującego się minimalną ceną, będzie to po prostu koszt danego podsystemu itd. W wielu przypadkach możliwe lub konieczne jest zastosowanie optymalizacji wielokryteriowej.

Opisany w sposób formalny, schemat zależności pomiędzy wymaganiami funkcjonalnymi, regułami diagnostycznymi oraz podsystemami, wraz z przykładowymi ich egzemplarzami, zilustrowany został w sposób graficzny na rys. 6.3. Dla potrzeb rozważań w ramach inżynierii wymagań, reguły diagnostyczne oraz przypisane do nich podsystemy mogą być traktowane jako wymagania pozafunkcjonalne przypisane do wybranych wymagań funkcjonalnych, opisujących oczekiwane funkcjonalności projektowanego systemu diagnostycznego. Opisują one potencjalne możliwe ścieżki, w ramach których mogą być zrealizowane poszczególne z wymagań funkcjonalnych.



Rys. 6.3: Schemat zależności pomiędzy elementami proponowanego formatu zapisu wymagań

6.4. Pozyskiwanie zbioru wymagań

W ogólnym przypadku proces pozyskiwania wymagań niezbędnych do zapisu potrzeby, w przedstawiony sformalizowany sposób, nie jest zadaniem łatwym i dodatkowo w wielu przypadkach jest jednocześnie niezwykle czasochłonny. Identycznie tak jak i w obszarze inżynierii oprogramowania, uzależniony jest on od wielu różnych czynników. Przede wszystkim konieczne jest uwzględnienie szczegółowej wiedzy o obiekcie technicznym, w tym informacji o jego strukturze, warunkach działania oraz możliwych jego uszkodzeniach. Kolejnym ważnym czynnikiem jest dostępna wiedza diagnostyczna na temat możliwych symptomów, a tym samym metod i technik diagnostycznych, które to mogą być zastosowane do wykrycia poszczególnych uszkodzeń. W zależności od klasy obiektu technicznego, w tym m.in. od poziomu innowacyjności rozwiązań w nim zastosowanych, dostępność poszczególnych fragmentów wiedzy będzie różna.

Z uwagi na wymienione trudności, proces pozyskiwania wymagań powinien być adaptacyjne wspomagany, poprzez zastosowanie różnego rodzaju dostępnych banków wiedzy, katalogów elementów itd. W ogólnym przypadku dla każdego obiektu technicznego, możliwe jest opracowanie tzw. katalogu zdarzeń niebezpiecznych. Obejmuje on listę możliwych uszkodzeń poszczególnych podzespołów rozpatrywanego obiektu. Katalog ten, może być opracowywany na podstawie przeprowadzonej analizy ryzyka, która to w większości przypadków musi być dokonana dla potrzeb nowo opracowywanych obiektów technicznych. Poszczególnym uszkodzeniom zapisanym w takim katalogu, możliwe jest przypisanie wartości oszacowanego ryzyka, tj. wartości prawdopodobieństwa/częstości wystąpienia uszkodzenia oraz wielkości potencjalnych skutków wystąpienia tego uszkodzenia. Dodatkowo poszczególnym uszkodzeniom, można przypisać potencjalne ich symptomy. Na ich podstawie możliwe jest opracowanie stosownych reguł diagnostycznych. W ogólnym przypadku niezbędna wiedza diagnostyczna może być pozyskiwana na podstawie (Cholewa i Amarowicz, 2012; Amarowicz, 2014a) dostępnych danych literaturowych, wiedzy i doświadczenia ekspertów, jako wynik przeprowadzonych czynnych i biernych eksperymentów diagnostycznych, w wyniku procesu stosowania symulatorów obiektów technicznych itp. Osobnym zagadnieniem są katalogi typowych elementów wchodzących w skład torów pomiarowych i algorytmów analizy i wnioskowania, udostępniane przez producentów tych rozwiązań.

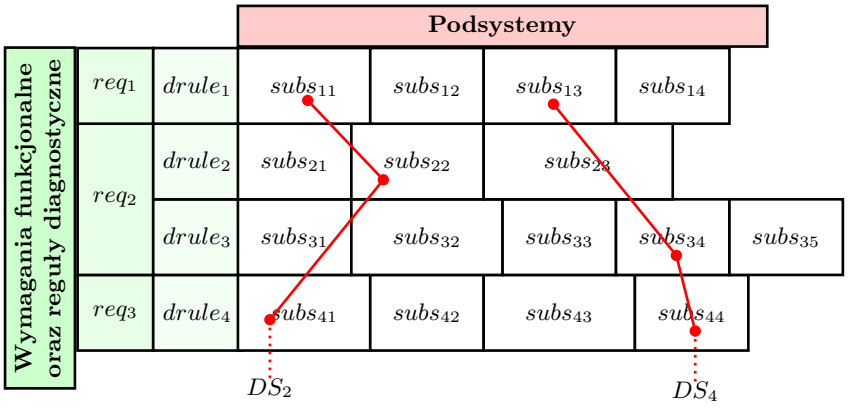
Na podstawie tak opracowanych banków danych możliwe jest przeprowadzenie procesu zapisu potrzeby w sformalizowany sposób. W zasadzie sprowadza się on do sekwencyjnego definiowania kolejnych elementów sformalizowanej potrzeby tj. wymagania funkcjonalnego, reguły diagnostycznej oraz podsystemu, wg schematu pokazanego na rys. 6.3. W proponowanym podejściu projektowym zakłada się, że pozyskiwane są wszelkie możliwe wymagania, opisujące potencjalne funkcjonalności, bez wnikania w celowość zastosowania tych funkcjonalności w końcowym projekcie systemu diagnostycznego. Proces oceny celowości zastosowania poszczególnych funkcjonalności przeprowadzony zostanie w kolejnym etapie procesu projektowego. Duża zaletą opisanego podejścia do procesu definiowania wymagań, jest możliwość osobnego budowania banków danych oraz pozyskiwania wymagań. Oba te etapy mogą być przeprowadzane nie tylko niezależnie, ale również przez różne osoby np. specjalistów dziedzinowych. Cały proces pozyskiwania wymagań, może być dodatkowo uproszczony poprzez dokonanie dekompozycji obiektu technicznego na podzespoły i definiowanie wymagań dla tak wyodrębnionych elementów składowych.

W procesie pozyskiwania wymagań, ważną rolę odgrywa także wiedza i doświadczenie osób projektujących system diagnostyczny. Niejednokrotnie mogą oni swoim doświadczeniem, formułować adekwatne wymagania oraz poprawne i niezbędne kryteria ograniczające, które mają być spełnione przez podsystemy wykorzystywane do realizacji wybranych reguł diagnostycznych.

6.5. Generowanie zbioru potencjalnych rozwiązań systemu diagnostycznego

Zaproponowany sposób zapisu potrzeby umożliwia rozpatrywanie projektu systemu diagnostycznego na trzech, zhierarchizowanych poziomach. Pierwszy z nich wyrażony za pomocą wymagań funkcjonalnych jest zbiorem możliwych funkcjonalności systemu diagnostycznego, przy czym są one rozumiane jako konieczność rozpoznawania potencjalnych stanów technicznych rozpatrywanego obiektu technicznego. Drugi z poziomów, wyrażony za pomocą reguł diagnostycznych, jest opisem możliwych do zastosowania metod i technik pomiarowych oraz niezbędnych metod analizy zarejestrowanych sygnałów, dzięki czemu możliwe jest wnioskowanie o wystąpieniu określonych uszkodzeń, a tym samym określonych stanów rozpatrywanego obiektu. Ostatni z poziomów, zapisany pod postacią podsystemów, związany jest z konkretnymi rozwiązaniami technicznymi, które umożliwiają przeprowadzenie działań opisanych przez reguły diagnostyczne.

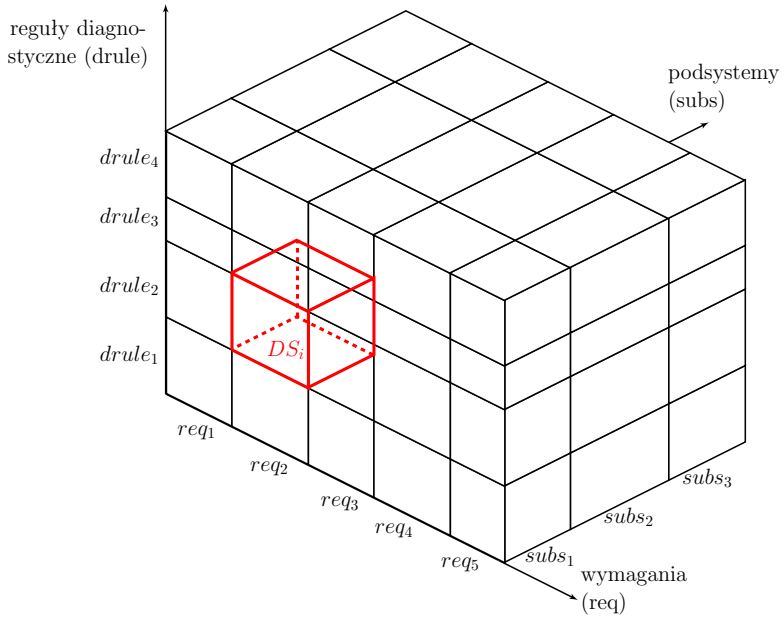
Zgodnie z założeniami, przedstawionymi we wstępie do rozdziału, jako projekt systemu diagnostycznego rozumiany jest zbiór funkcjonalności oraz zbiór elementów wchodzących w skład jego struktury. Jak można zauważyć, wymienione trzy poziomy sformalizowanej potrzeby, mogą jednoznacznie definiować potencjalne rozwiązania systemu diagnostycznego. Mianowicie, zbiór wymagań funkcjonalnych wprost opisuje oczekiwane funkcjonalności systemu diagnostycznego, natomiast podsystemy w ramach danej reguły diagnostycznej, opisują potencjalne elementy składowe struktury tego systemu. Potencjalne rozwiązania systemu diagnostycznego są zatem kombinacją wymagań funkcjonalnych, zdefiniowanych w ramach nich reguł diagnostycznych oraz przypisanych do nich podsystemów. W celu łatwiejszej prezentacji zbioru możliwych rozwiązań można posłużyć się tzw. tablicą morfologiczną lub też prezentacją w przestrzeni trójwymiarowej. Sposoby tych prezentacji wraz z przykładowymi rozwiązaniami systemu diagnostycznego pokazano stosownie na rys. 6.4 i 6.5.



Rys. 6.4: Tablica morfologiczna wraz z zaznaczonymi przykładowymi projektami systemu diagnostycznego

W procesie generowania zbioru potencjalnych rozwiązań systemu diagnostycznego, możliwe jest uwzględnianie następujących założeń:

- dla potrzeb generowania zbioru rozwiązań mogą zostać uwzględniane tylko te wymagania i reguły diagnostyczne, którym w procesie oceny przyporządkowano wartość współczynnika preferencji większą od pewnej ustalonej wartości minimalnej, dzięki takiemu podejściu eliminowane są rozwiązania mało przydatne dla potrzeb oceny stanu technicznego analizowanego obiektu,



Rys. 6.5: Trójwymiarowa przestrzeń rozwiązań systemu diagnostycznego, wraz z zaznaczony przykładowym rozwiązaniem DS_i

- dla potrzeb realizacji funkcjonalności opisanej przez dane wymaganie, wystarczające jest zastosowanie jednej reguły diagnostycznej,
- w procesie definiowania podsystemów uwzględnianie są tylko te elementy, które spełniają kryteria ograniczające zdefiniowane w ramach danej reguły,
- liczba możliwych podsystemów, zdefiniowanych w ramach danej reguły, zależna jest od liczby typów elementów definiujących podsystem oraz od liczebności zbioru elementów w ramach każdego z typów.

W ogólnym przypadku, uwzględniając powyższe informacje, możliwe jest wygenerowanie zbioru potencjalnych rozwiązań systemu diagnostycznego o liczebności:

$$\prod_i \prod_j card(etype_{j,i}), \tag{6.9}$$

gdzie $card(etype_{j,i})$ jest liczbą elementów, dostępnych w ramach j-tego typu elementu wchodzącego skład definicji podsystemu, realizującego regułę diagnostyczną dla i-tego wymagania. Ze zbioru tego, na podstawie zdefiniowanych kryteriów optymalizacji, konieczne jest wyodrębnienie optymalnego projektu systemu diagnostycznego.

6.6. Ocena zbioru wymagań oraz potencjalnych rozwiązań systemu diagnostycznego

W procesie pozyskiwania wymagań gromadzony jest bardzo liczby zbiór wymagań funkcjonalnych:

$$REQ = \{req_1, req_2, \ldots req_n\}, \tag{6.10}$$

opisujących możliwe funkcjonalności projektowanego systemu diagnostycznego. Zbiór ten, zawiera zarówno wymagania niezbędne z punktu widzenia projektowanego systemu diagnostycznego,

jak również wymagania które są mało przydatne dla potrzeb przyjętej funkcji celu, polegającej na minimalizacji ryzyka technicznego. Dlatego też konieczne jest przeprowadzenie procesu oceny zebranego zbioru wymagań, celem wyodrębnienia pewnego podzbioru wymagań:

$$subREQ = \{req_1, req_2, \ldots req_m\}, \tag{6.11}$$

niezbędnych z punktu widzenia projektowanego systemu, przy czym prawdziwa jest zależność:

$$card(REQ) \geq card(subREQ). \tag{6.12}$$

Poszczególnym z wymagań, możliwe jest przypisanie pewnego zbioru reguł diagnostycznych:

$$DRULE = \{drule_1, drule_2, \ldots drule_r\}, \tag{6.13}$$

opisujących sposób diagnozowania poszczególnych uszkodzeń. Ze zbioru tego, konieczne jest wyodrębnienie zazwyczaj jednej reguły, pozwalającej na efektywne wnioskowanie o wystąpieniu określonego stanu technicznego.

Na podstawie zbioru wymagań funkcjonalnych i reguł diagnostycznych oraz zdefiniowanych katalogów elementów, możliwe jest wygenerowanie zbioru:

$$DS = \{ds_1, ds_2, \ldots ds_p\}, \tag{6.14}$$

potencjalnych rozwiązań systemu diagnostycznego. Ze zbioru tego, konieczne jest wyodrębnienie rozwiązania optymalnego, najlepiej spełniającego przyjęte kryteria.

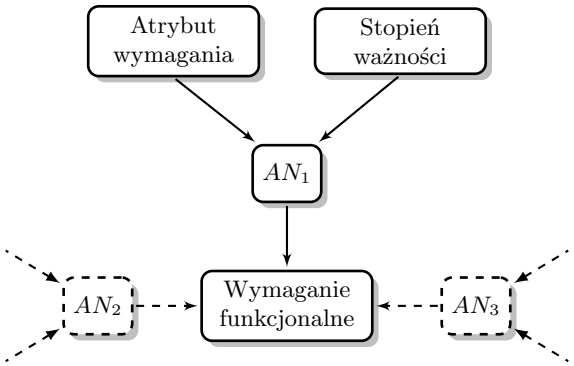
W wyniku przeprowadzenia wymienionych powyżej operacji, otrzymuje się odpowiedź na dwa zasadnicze pytania, związane z procesem projektowania systemów diagnostycznych tj.:

- jakie powinny być funkcjonalności systemu diagnostycznego, tj. które stany techniczne i jakimi metodami powinny być rozpoznawane poprzez zastosowanie systemu diagnostycznego,
- jaka powinna być struktura systemu diagnostycznego, czyli jakie elementy układów pomiarowo-analizujących powinny być częściami składowymi projektowanego systemu.

W obszarze inżynierii oprogramowania, proces oceny zebranych wymagań przybiera różne formy. W większości przypadków sprowadza się on, do procesu negocjacji wymagań pomiędzy klientem zlecającym projekt a jego potencjalnym wykonawcą. W ramach przeprowadzanych negocjacji, każdy z uczestników artykułuje swoje oczekiwania i potrzeby. Po serii rozmów wspólnie dochodzą oni do porozumienia w kwestii ostatecznego zbioru przyjętych wymagań. Z powodu kilku czynników, bezpośrednie zastosowanie takiego podejścia nie może być w rozpytywanym przypadku możliwe. Pierwszym z nich jest brak jawnego klienta dla potrzeb procesu negocjacji wymagań. Osoba zlecająca/wyrażająca zapotrzebowanie na wykonanie systemu diagnostycznego, w wielu przypadkach nie posiada całkowitej wiedzy o rozpatrywanym obiekcie technicznym oraz zazwyczaj nie jest ona zaznajomiona z zagadnieniami diagnostycznymi. Z drugiej strony, potencjalni wykonawcy systemu diagnostycznego (np. informatycy), również mogą nie posiadać wymaganej wiedzy z obszaru diagnostyki technicznej oraz wiedzy o strukturze obiektu. Dodatkowo, w wielu przypadkach wiedza o obiekcie technicznym a w szczególności wiedza na temat sposobów jego diagnozowania może nie być w pełni dostępna i jakościowo dobra. W związku z tym potencjalni uczestnicy procesu negocjacji nie będą zdolni do prawidłowego artykułowania wszystkich wymagań. Dlatego też konieczne jest zastosowanie pewnego systemu informatyczne-

go, wspomagającego proces oceny wymagań na podstawie zgromadzonych danych. System ten może być traktowany jako tzw. wirtualny klient, reprezentujący obiekt techniczny i artykułujący niezbędne wymagania, biorąc pod uwagę określone dane wejście wprowadzane do systemu, np. wiedza o obiekcie, ograniczenia itp.

Dla potrzeb wymienionych działań możliwe jest opracowanie stosownego systemu doradczego. Za jego pomocą, możliwe będzie przeprowadzenie procesu wnioskowania i oceny zebranego zbioru wymagań funkcjonalnych oraz reguł diagnostycznych, na podstawie zgromadzonej wiedzy i danych. Baza wiedzy takiego systemu, może zostać opracowana za pomocą stosownych modeli sieciowych. W rozpatrywanym przypadku przyjęto sieci przekonań. Struktura wymaganego modelu sieciowego, wynika wprost z formatu zapisu wymagań i pokazana została (dla wymagania funkcjonalnego) na rys. 6.6.



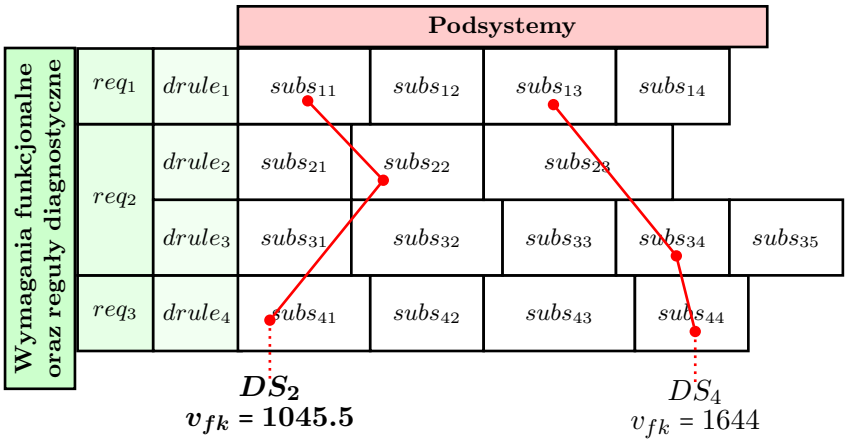
Rys. 6.6: Struktura modelu sieciowego dla wymagania funkcjonalnego, AN_i – węzły dodatkowe

Wyróżnić w niej można węzły czterech klas, tj. *wymaganie funkcjonalne*, *atrybut wymagania*, *stopień ważności* oraz *węzeł dodatkowy*. W ogólnym przypadku węzeł *wymaganie* opisuje oczekiwaną funkcjonalność projektowanego systemu diagnostycznego. Wartość poszczególnych atrybutów wymagania, przechowywana jest w węzłach *atrybut wymagania*. W węzłach *stopień ważności*, przechowywane są informacje o stopniu ważności danego atrybutu, z punktu widzenia procesu oceny rozpatrywanego wymagania. *Węzły dodatkowe* wynikają z przyjętej struktury sieci i nie niosą wprost żadnej informacji. Węzłami wejściowymi są węzły klasy: *atrybut wymagania* oraz *stopień ważności*, natomiast węzłem wyjściowym jest węzeł *wymaganie*. Przyjęta struktura sieci pozwala na rozpatrywanie wymagań, dla których przypisana jest różna liczba atrybutów. Zadaniem wymagającym indywidualnego rozwiązania, jest opracowanie tablic prawdopodobieństw warunkowych, przypisanych do węzła *wymaganie* oraz *węzłów dodatkowych*. Wartości poszczególnych prawdopodobieństw, zależą od rodzaju atrybutu oraz jego wartości. W identyczny sposób definiowany jest model sieciowy dla potrzeb oceny reguł diagnostycznych.

Stosowne modele sieciowe można generować w sposób automatyczny, dla każdego wymagania funkcjonalnego oraz reguły diagnostycznej. W wyniku procesu wnioskowania, na podstawie wartości węzłów wejściowych wyznaczane są wartości stopnia preferencji, przypisanego stosownie do węzłów *wymaganie* oraz *reguła diagnostyczna*. Wartość tego współczynnika, niesie ze sobą informację o tym, jak bardzo dane wymaganie/reguła diagnostyczna jest pożądana z punktu widzenia projektowanego systemu diagnostycznego. Im większa wartość tego współczynnika tym istnieje większe przekonanie, co do potrzeby implementacji rozwiązania opisanego przez dane wymaganie/regułę diagnostyczną.

Proces oceny potencjalnych rozwiązań systemu diagnostycznego, sprowadza się do procesu oceny poszczególnych podsystemów i może być on rozumiany jako proces optymalizacji pod-

systemów przypisanych do poszczególnych reguł diagnostycznych. Przebiega on na podstawie kryteriów optymalizacji, które są skorelowane z wybranymi atrybutami przypisanymi do poszczególnych elementów podsystemów. W najprostszym przypadku, można zatem mówić o procesie maksymalizacji lub minimalizacji wartości wybranych atrybutów np. minimalizacja ceny. Dopuszcza się możliwość optymalizacji lokalnej oraz globalnej, jak również optymalizacji wielokryterialnej. Optymalizacja lokalna realizowana jest w obrębie pojedynczego podsystemu, natomiast globalna dla wszystkich podsystemów. Z uwagi na to, że projekt systemu diagnostycznego jest złożony z licznych podsystemów, możliwe jest zsumowanie wyznaczonych wartości funkcji kryterialnej dla poszczególnych podsystemów (pod warunkiem, że stosowane są globalne kryteria optymalizacji), dzięki czemu otrzymuje się stosowną wartość funkcji kryterialnej dla poszczególnych rozwiązań systemu diagnostycznego (rys. 6.7).



Rys. 6.7: Tablica morfologiczna ze zbiorem możliwych rozwiązań systemu diagnostycznego, v_{fk} – wartość funkcji kryterialnej

Uporządkowując, zgodnie z wyznaczonymi wartościami stopnia preferencji lub funkcji kryterialnej, wymagania funkcjonalne, reguły diagnostyczne oraz podsystemy, możliwe jest otrzymanie list tych elementów, wg stopnia przydatności rozwiązań przez nie opisywanych, do realizacji podstawowej potrzeby N, to jest diagnozowania aktualnego stanu technicznego analizowanego obiektu, celem minimalizacji ryzyka technicznego. Otrzymane listy wymagań funkcjonalnych, reguł oraz podsystemów opisują projekt optymalnego systemu diagnostycznego.

6.7. Podsumowanie

W niniejszym rozdziale przedstawiono opis proponowanego podejścia do problemu projektowania systemów diagnostycznych z wykorzystaniem zbiorów wymagań. Jego podstawowym elementem jest założenie, że konieczne jest wprowadzenie sformalizowanego zapisu potrzeby projektowo-konstrukcyjnej, celem dokładniejszego wyrażenia potrzeb związanych z koniecznością opracowania systemu diagnostycznego. Do sformalizowanego zapisu potrzeby, wykorzystane zostały zbiory wymagań powszechnie stosowane podczas realizacji różnego rodzaju projektów informatycznych. Po zapisaniu potrzeby w sformalizowany sposób, możliwe jest wygenerowanie zbioru potencjalnych projektów systemu diagnostycznego. Niezwykle ważnym zagadnieniem jest proces pozyskiwania a następnie oceny zebranego zbioru wymagań. Proces oceny wymagań może być rozpatrywany jako proces wnioskowania w modelach sieciowych, opisujących zgromadzoną wiedzę związaną z rozpatrywanym zagadnieniem. W wyniku przeprowadzenia procesu wniosko-

wania, poszczególnym wymaganiom funkcjonalnych oraz regułom diagnostycznym przyporządkowywana jest stosowna wartość stopnia preferencji. Niesie ona informację o tym, jak bardzo rozwiązanie opisane przez dane wymaganie/regułę diagnostyczną jest potrzebne w końcowym projekcie systemu diagnostycznego. Na podstawie wyodrębnionego zbioru wymagań oraz zdefiniowanego katalogu elementów możliwe jest wygenerowanie zbioru potencjalnych rozwiązań systemu diagnostycznego. Wybór rozwiązania optymalnego odbywa się na podstawie zdefiniowanych kryteriów optymalizacji, które są skorelowane z atrybutami przypisanymi do poszczególnych elementów podsystemów.

Zaproponowane podejście projektowe umożliwia ominięcie ważnego problemu, który pojawia się podczas próby projektowania systemów diagnostycznych z użyciem standardowych metod. Zapisana w zaproponowany, sformalizowany sposób potrzeba, pozwala na uwzględnienie na etapie generowania zbioru potencjalnych rozwiązań, wyników analizy ryzyka oraz dostępnych elementów typowych systemów pomiarowych. Dzięki takiemu podejściu możliwe jest opracowywanie projektów efektywnych systemów diagnostycznych.

Rozdział 7

Implementacja zaproponowanej metody

W niniejszym rozdziale opisano wyniki procesu implementacji zaproponowanej metody wspomagania projektowania systemów diagnostycznych. Omówiono strukturę opracowanego Edytora wymagań, wyszczególniono najważniejsze funkcjonalności edytora oraz opisano przykładowy tok postępowania z zastosowaniem opracowanego oprogramowania.

7.1. Struktura i funkcjonalności *Edytora wymagań*

Dla potrzeb praktycznej weryfikacji zaproponowanego podejścia do zagadnienia projektowania systemów diagnostycznych, opracowane zostało dedykowane środowisko informatyczne, zwane dalej *Edytorem wymagań*, z zastosowaniem którego możliwe jest przeprowadzenie większości działań przedstawionych w proponowanym podejściu projektowym. W procesie implementacji, uwzględniono następujące założenia upraszczające proponowaną metodę:

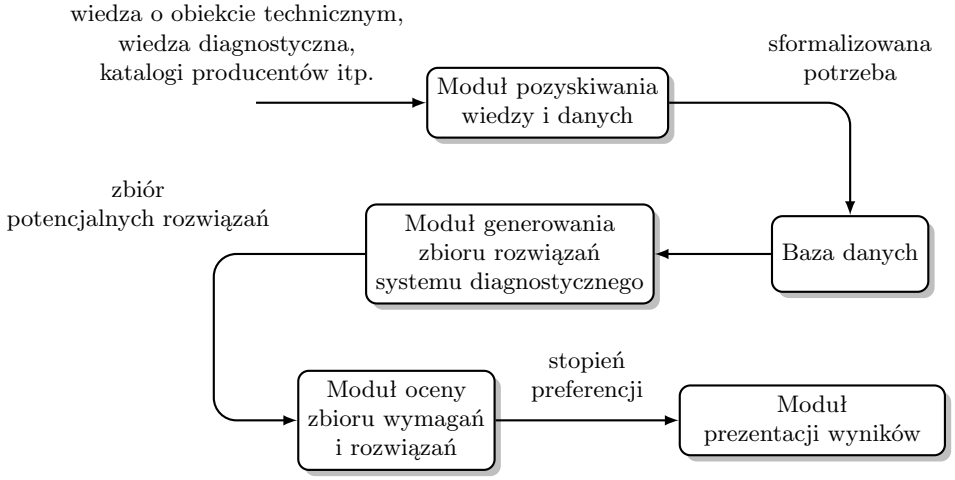
- stała liczba, z góry określonych atrybutów przypisanych do poszczególnych wymagań funkcjonalnych oraz reguł diagnostycznych, dla wymagań są to: poziom redukcji prawdopodobieństwa wystąpienia zdarzenia niebezpiecznego, poziom redukcji kosztów wygenerowanych po wystąpieniu danego zdarzenia, poziom redukcji czasu przestoju po wystąpieniu uszkodzenia, stosowalność wymagania oraz informacja o tym czy wymaganie spełnia ograniczenia projektowe, natomiast dla reguł diagnostycznych są to: poziom fałszywych alarmów, stosowalność reguły oraz informacja o spełnieniu ograniczeń projektowych,
- rozpatrywanie stopnia ważności atrybutów przypisanych do wymagań funkcjonalnych oraz reguł diagnostycznych, na dwóch poziomach: *tak*, *nie*, przy czym przyjęcie wartości *nie* oznacza, że wartość danego atrybutu nie będzie brana pod uwagę w procesie oceny,
- ocena atrybutów przypisanych do wymagań oraz reguł diagnostycznych na poziomie jakościowym,
- stałe tablice prawdopodobieństw warunkowych przypisanych do poszczególnych węzłów generowanych modeli sieciowych,
- rezygnacja z jawnej prezentacji wszystkich rozwiązań projektu systemu diagnostycznego,

na rzecz prezentacji rozwiązania optymalnego w formie raportu będącego zbiorem funkcjonalności systemu diagnostycznego oraz odpowiadającym im podsystemom,

- optymalizacja postaci podsystemów z zastosowaniem jedynie kryteriów globalnych oraz przy założeniu optymalizacji jednokryterialnej.

Założenia upraszczające wprowadzane zostały z uwagi na ograniczone ramy czasowe realizowanych badań doktorskich, w ramach których problematyczne byłoby opracowanie odpowiedniego środowiska informatycznego, realizującego w sposób elastyczny wszystkie aspekty proponowanego podejścia projektowego.

Edytor wymagań opracowany został w języku programowania *Java*. Do opracowania bazy danych współpracującej z edytorem zastosowano środowisko *MySQL*, natomiast proces wnioskowania w modelach sieciowych, przeprowadzany jest z wykorzystaniem funkcji zawartych w darmowym pakiecie *REx*[1], opracowanym w ramach języka *R*.

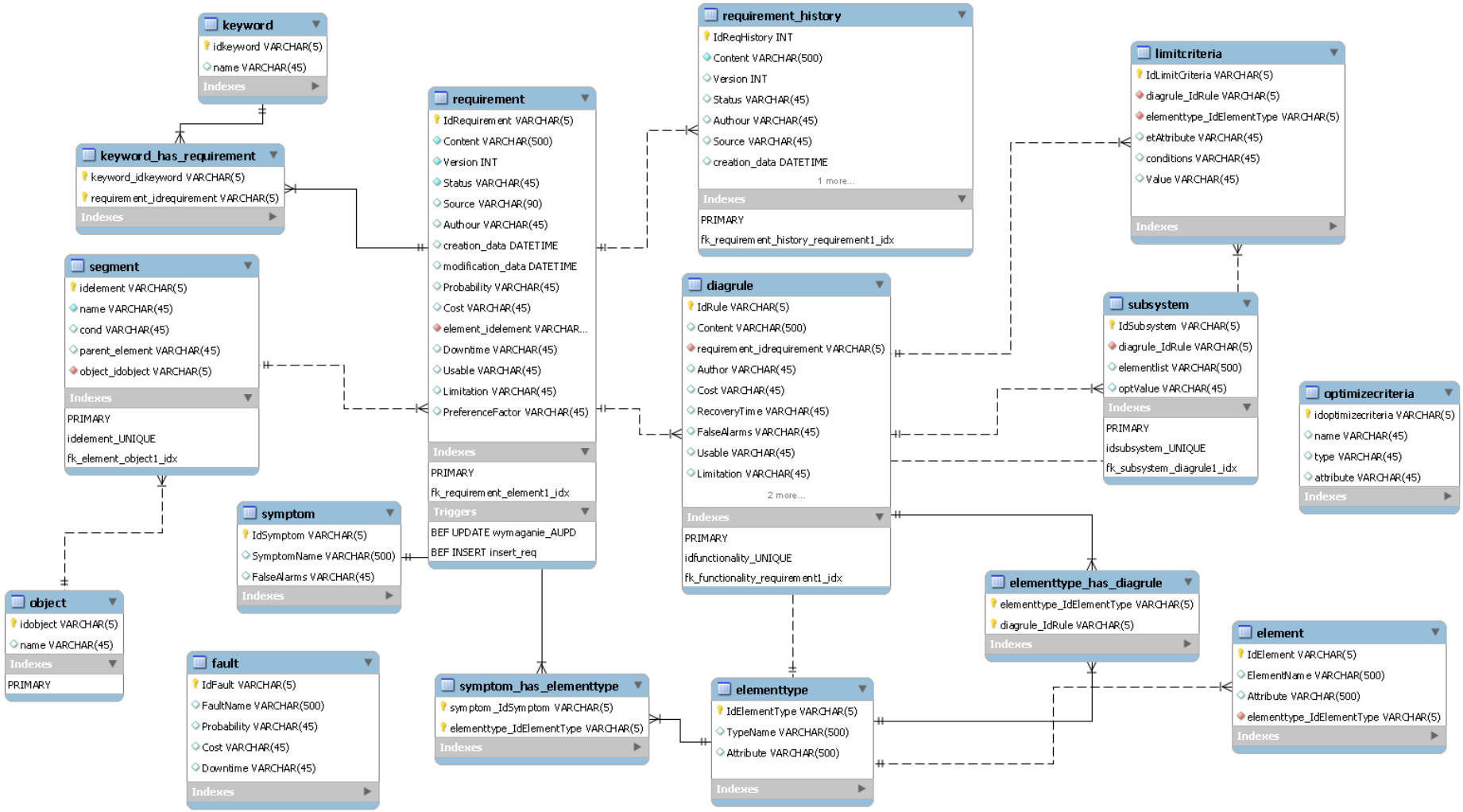


Rys. 7.1: Struktura opracowanego *Edytora wymagań*

Strukturę opracowanego edytora pokazano na rys. 7.1. Wyróżnić w niej można, następujące główne moduły:

- moduł pozyskiwania wiedzy oraz danych – fragment aplikacji zrealizowany w postaci zestawu formularzy, za pośrednictwem których możliwe jest definiowanie, edycja oraz przeglądanie poszczególnych składników sformalizowanego zapisu potrzeby tj. wymagań, reguł diagnostycznych, podsystemów, jak również katalogów uszkodzeń, symptomów oraz elementów podsystemów,
- relacyjna baza danych – stosowana jest ona dla potrzeb przechowywania wszystkich niezbędnych elementów, koniecznych do opracowania zapisu potrzeby w przyjęty sformalizowany sposób, struktura bazy danych, w postaci diagramu *EER* (ang. *Enhanced Entity–Relationship*) pokazana została na rys. 7.2,
- moduł generowania zbioru rozwiązań – odpowiedzialny jest on za wygenerowanie, na postawie zgromadzonych danych oraz przyjętych kryteriów ograniczających, zbioru potencjalnych rozwiązań projektu systemu diagnostycznego,
- moduł oceny - za pośrednictwem tego modułu realizowany jest proces wnioskowania w modelach sieciowych oraz wyszukiwane jest optymalne rozwiązanie projektu systemu diagnostycznego, proces wnioskowania opiera się na funkcjach zawartych w pakiecie *REx*, wy-

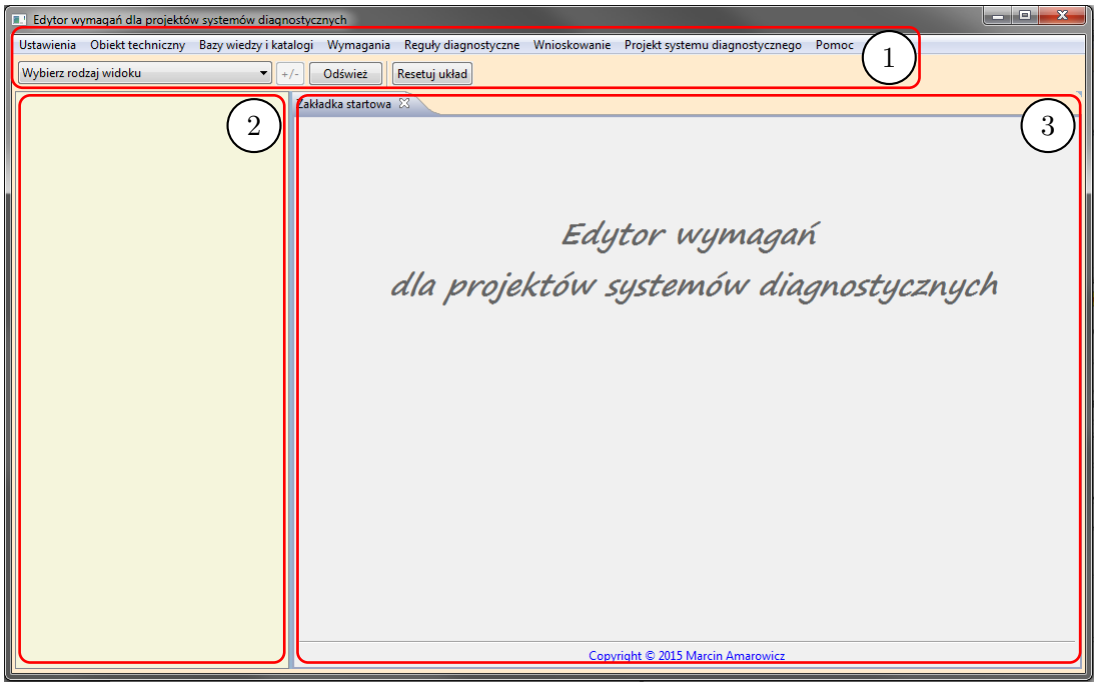
[1]Autor rozprawy jest autorem wybranych fragmentów tego pakietu.



Rys. 7.2: Edytor wymagań; diagram *EER* przedstawiający strukturę opracowanej bazy danych

magane modele sieciowe są generowane w sposób automatyczny na podstawie pozyskanej wcześniej wiedzy i danych,

- moduł prezentacji wyników – moduł aplikacji, przeznaczony do prezentacji wyników procesu oceny, w tym przedstawienia struktury optymalnego rozwiązania systemu diagnostycznego, pod postacią raportu będącego listą oczekiwanych funkcjonalności oraz odpowiadającymi im podsystemami.



Rys. 7.3: Edytor wymagań; okno główne aplikacji

Z punktu widzenia graficznego interfejsu użytkownika (okno główne aplikacji pokazano na rys. 7.3), wyróżnić można trzy główne części aplikacji, tj.:

1. menu główne – obszar aplikacji, z poziomu którego uruchamiane są m.in. poszczególne z formularzy modułu pozyskiwania wiedzy i danych, proces wnioskowania itd.,
2. drzewo struktury – obszar aplikacji, w którym wyświetlana jest w trybie drzewiastym zdefiniowania struktura obiektu wraz z przypisanymi do niej wymaganiami, regułami itd.
3. obszar roboczy – obszar aplikacji, w którym wyświetlane są poszczególne z formularzy, przeznaczone m.in. do definiowania, edycji oraz przeglądania poszczególnych fragmentów bazy wiedzy.

Opracowany *Edytor wymagań* charakteryzuje się następującym zbiorem funkcjonalności:

- definiowanie oraz edycja struktury obiektu technicznego,
- definiowanie oraz edycja katalogów uszkodzeń, ich symptomów oraz katalogów typowych elementów wchodzących w skład podsystemów diagnostycznych (rys. 7.5),
- definiowanie oraz edycja wymagań funkcjonalnych, przypisywanie ich do wybranych elementów struktury obiektu technicznego (rys. 7.6),
- definiowanie oraz edycja reguł diagnostycznych (rys. 7.7),
- definiowanie oraz edycja kryteriów ograniczających możliwe do zastosowania elementy podsystemów,

- przechowywanie gromadzonej wiedzy i danych w zewnętrznej bazie danych,
- kontrola poprawności formatu gromadzonych danych w bazie danych,
- przeglądanie zgromadzonych danych, stosowanie wybranych metod filtracji danych,
- wizualizacja hierarchicznej struktury danych,
- automatyczne generowanie, przy uwzględnieniu zdefiniowanych kryteriów, zbioru potencjalnych rozwiązań projektu systemu diagnostycznego,
- automatyczne generowanie, na podstawie zgromadzonej wiedzy i danych, niezbędnych modeli sieciowych,
- ocena zbioru wymagań funkcjonalnych oraz reguł diagnostycznych jako proces wnioskowania w modelach sieciowych,
- wyznaczanie rozwiązania optymalnego, na podstawie zdefiniowanych kryteriów optymalizacji,
- prezentacja wyników procesu wnioskowania (rys. 7.8),
- eksport pozyskanych zbiorów wymagań funkcjonalnych, reguł diagnostycznych itp., do tabel w środowisku LaTeX(rys. 7.9),
- generowanie raportu podsumowującego optymalną postać projektu systemu diagnostycznego.

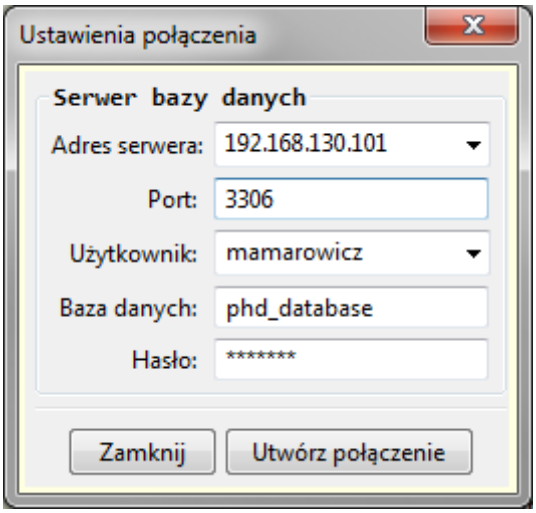
7.2. Praca z *Edytorem wymagań*

Poprzez zastosowanie opracowanego *Edytora wymagań* możliwe jest przeprowadzenie większości czynności, związanych z zaproponowanym podejściem projektowym do procesu opracowywania projektu systemu diagnostycznego. Przykładowy tok postępowania z wykorzystaniem niniejszego edytora jest następujący:

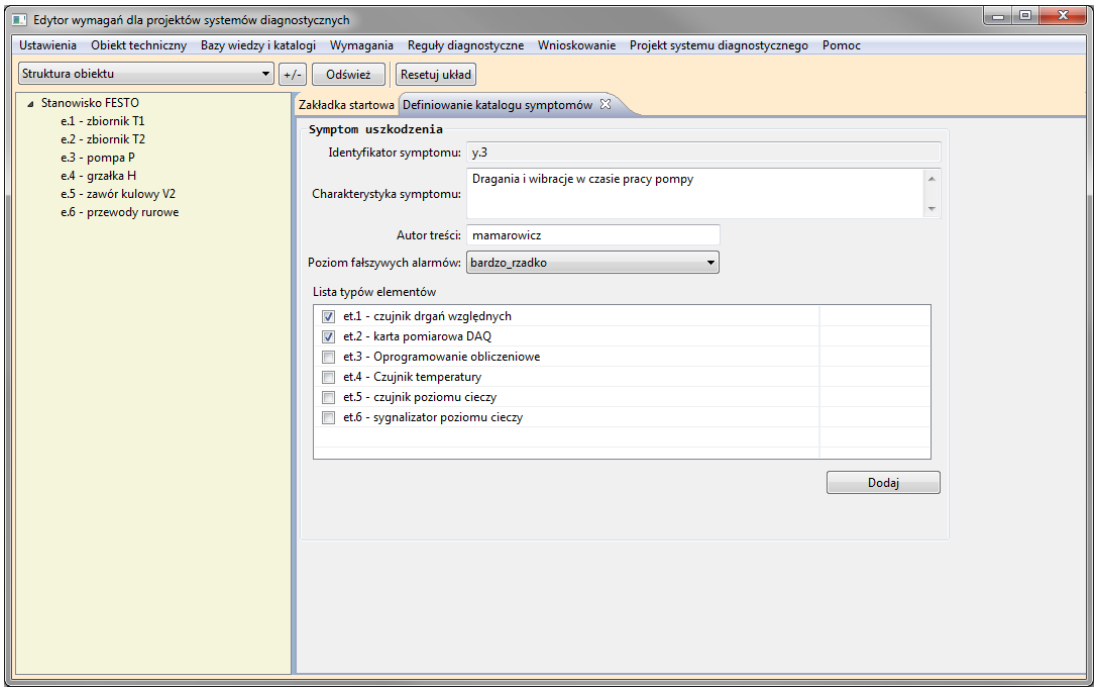
1. Uruchomienie *Edytora wymagań*.
2. Nawiązanie połączenia z bazą danych, w której gromadzone są poszczególne elementy sformalizowanej potrzeby (rys. 7.4).
3. Zdefiniowanie obiektu technicznego i jego elementów składowych.
4. Zdefiniowanie katalogu uszkodzeń, symptomów oraz katalogów elementów typowych podzespołów systemów diagnostycznych.
5. Zdefiniowanie zbioru wymagań funkcjonalnych opisujących konieczne do wykrywania uszkodzenia.
6. Przypisanie do poszczególnych wymagań funkcjonalnych reguł diagnostycznych, opisujących możliwe do zastosowania metody i techniki diagnostyczne przeznaczone do wykrywania poszczególnych uszkodzeń.
7. Zdefiniowanie podsystemów realizujących poszczególne reguły diagnostyczne.
8. Zdefiniowanie kryteriów ograniczających dla poszczególnych podsystemów.
9. Export danych, wymagań funkcjonalnych, reguł diagnostycznych do modeli sieciowych.
10. Proces wnioskowania w modelach sieciowych przeprowadzany z wykorzystaniem funkcji pakietu *REx* (wydruk 7.1).
11. Import wyników wnioskowania do Edytora wymagań.

12. Wygenerowanie zbioru potencjalnych rozwiązań projektu systemu diagnostycznego, jako kompilacji podsystemów dla poszczególnych reguł diagnostycznych.
13. Ocena podsystemów i wybór rozwiązania optymalnego.
14. Wygenerowanie raportu zawierającego optymalną postać projektu systemu diagnostycznego. Przykład raportu pokazano na wydruku 7.2.

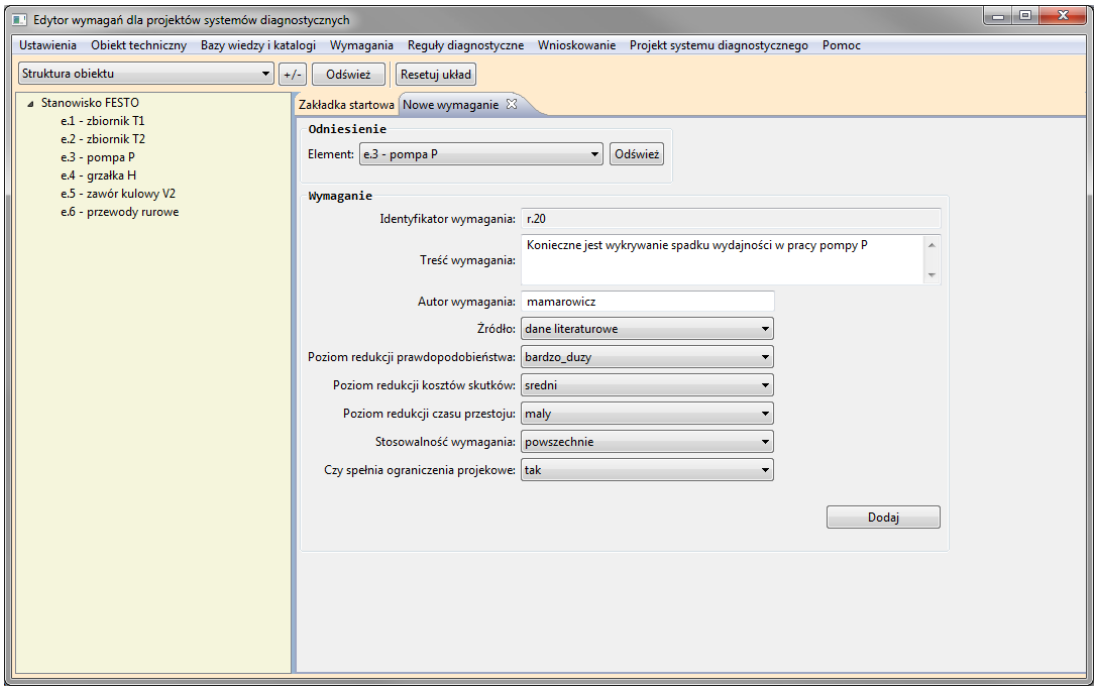
Na każdym z wyżej wymienionych etapów, możliwe jest przeprowadzenie procesu edycji oraz przeglądania zdefiniowanych elementów bazy wiedzy i danych. Zbiorcze zestawienie większości formularzy i okien opracowanego *Edytora wymagań* pokazano na rys. 7.10.



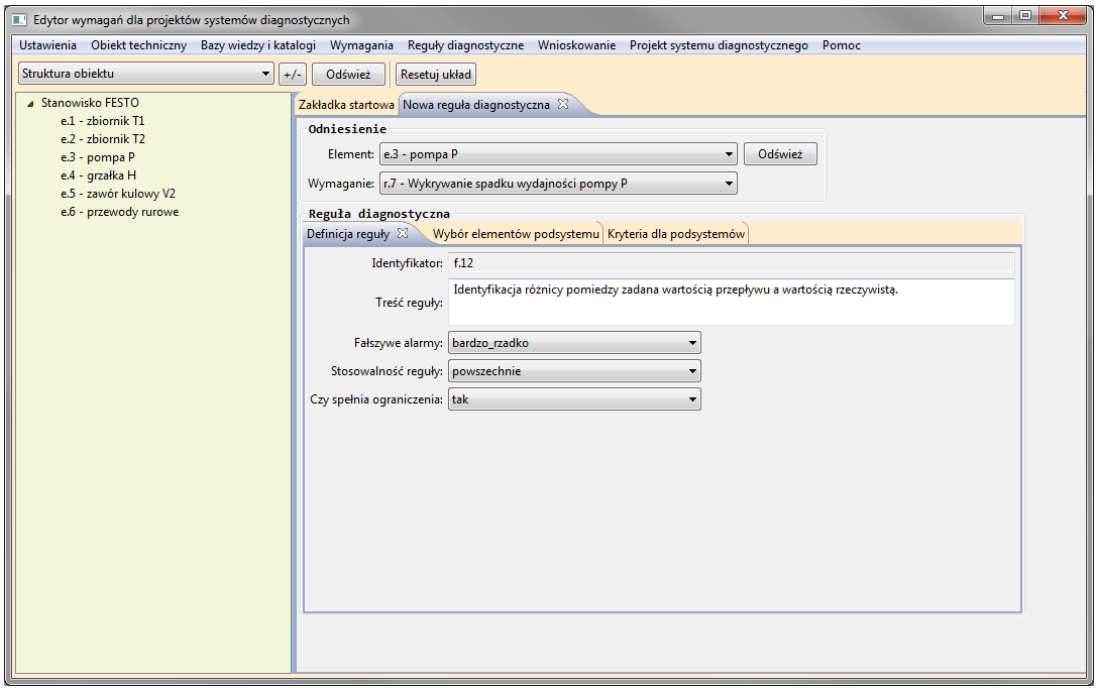
Rys. 7.4: Edytor wymagań; okno aplikacji umożliwiające nawiązanie połączenia z bazą danych



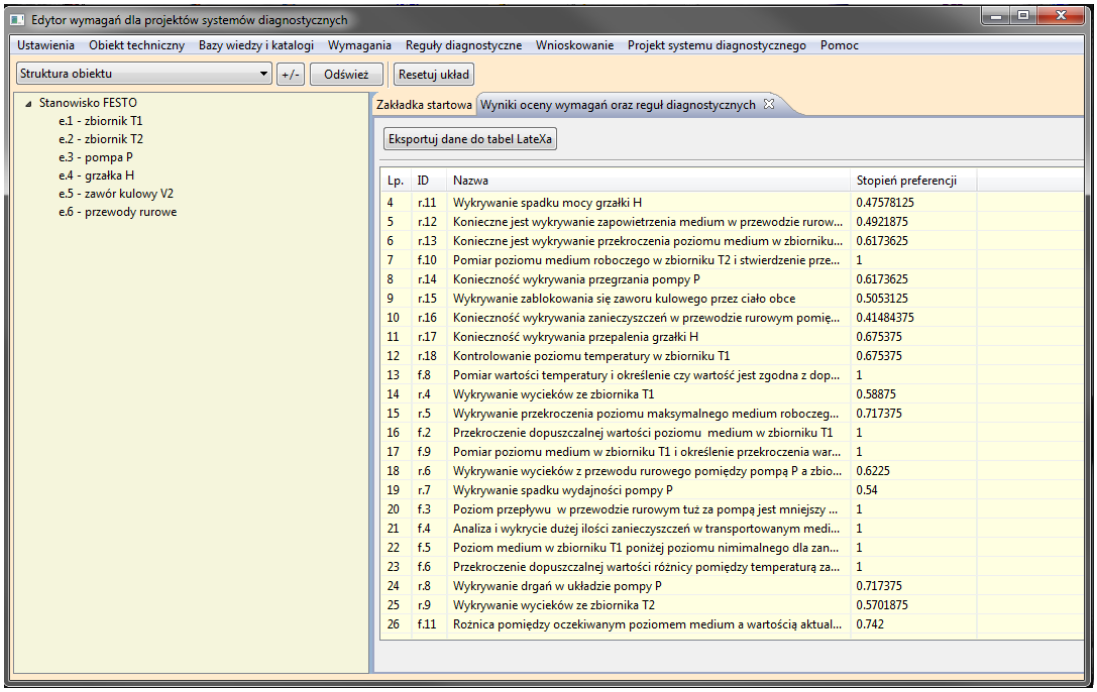
Rys. 7.5: Edytor wymagań; formularz przeznaczony do definiowania katalogu symptomów



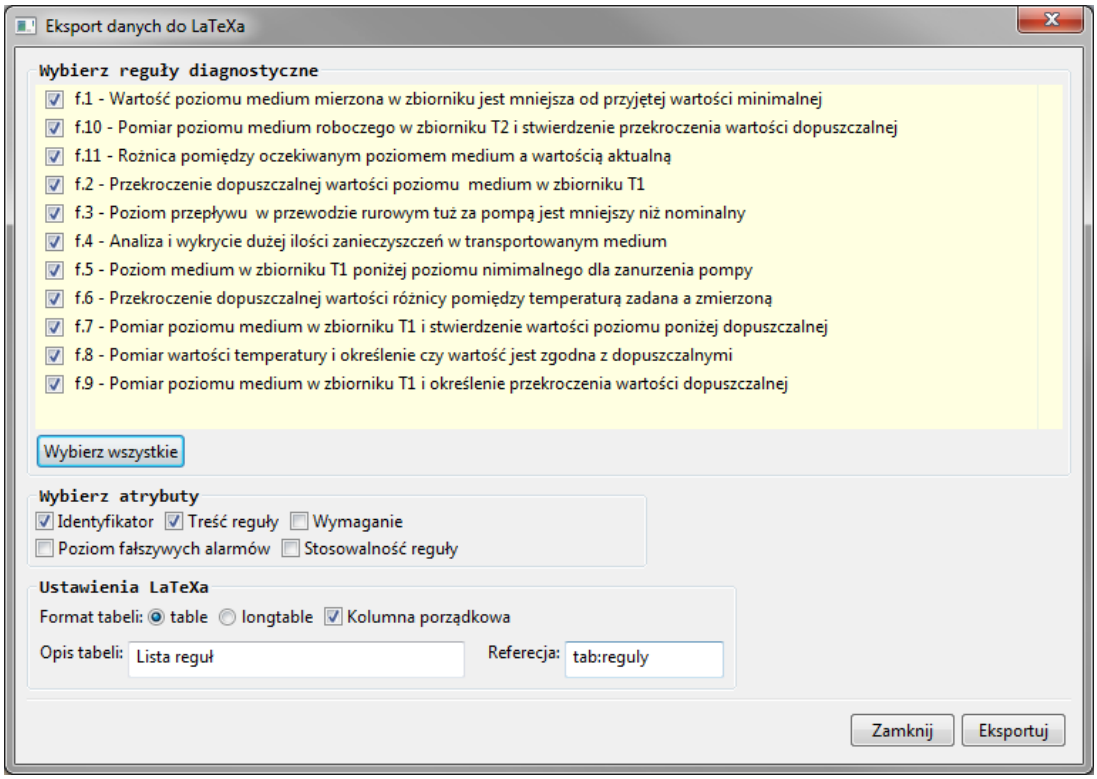
Rys. 7.6: Edytor wymagań; formularz przeznaczony do definiowania wymagania



Rys. 7.7: Edytor wymagań; formularz przeznaczony do definiowania reguły diagnostycznej



Rys. 7.8: Edytor wymagań; formularz prezentujący przykładowe wyniki procesu wnioskowania



Rys. 7.9: Edytor wymagań; okno aplikacji umożliwiające eksport zdefiniowanych reguł diagnostycznych do tabel środowiska LaTeX

Wydruk 7.1: Skrypt umożliwiający przeprowadzenie procesu wnioskowania z użyciem funkcji pakietu *REx*

```
1  ------------------------------------------------------------------
2  ------------------------- Edytor wymagań -------------------------
3  ------------ Copyright © 2015 Marcin Amarowicz IPKM PoISI----------
4  ------------------------------------------------------------------
5  --------------- Wnioskowanie w modelach sieciowych ---------------
6  ------------------------------------------------------------------
7
8  ## wczytanie pakietu REx oraz załadowanie niezbędnych funkcji
9  library(REx)
10 rex_gui();
11
12 ## wczytywanie danych z pliku
13 network <- new("CModel")
14 network <- loadModelXML(network, "dane.xml")
15
16 ## inicjacja zmiennych
17 nodes = vector(mode = "character")  # węzły
18 states = vector(mode = "character") # wartości
19 preferenceFactor = vector(mode = "character")   # stopień preferencji
20
21 ## kopia sieci do obliczeń
22 network2<-network
23
24 ## wprowadzenie danych do kopii
25 for (il in 1:network@iLayers){
26 if(strsplit(network@vLayers[[il]]@sName, "[.]")[[1]][1]=="r"){
27              nodelist = c('prob', 'cost', 'time', 'use', 'lim', ↙
                   ↳ paste(network@vLayers[[il]]@sName,"prob", sep=""),
28                            paste(network@vLayers[[il]]@sName,"cost", sep=""),
29              paste(network@vLayers[[il]]@sName,"time", sep=""),
30                            paste(network@vLayers[[il]]@sName,"use", sep=""),
31              paste(network@vLayers[[il]]@sName,"lim", sep=""))
32     }else{
33              nodelist = c('alarm', 'use', 'lim',
34                            paste(network@vLayers[[il]]@sName,"alarm", ↙
                                  ↳ sep=""),
35                            paste(network@vLayers[[il]]@sName,"use", sep=""),
36                            paste(network@vLayers[[il]]@sName,"lim", sep=""))
37     }
38
39     for (i in 1:length(nodelist)){
40              nodes[i]=nodelist[i]
41              for (j in 1: ↙
                   ↳ length(network@vLayers[[il]]@vNodes[[nodes[i]]]@sValNames)){
42                      if(network@vLayers[[il]]
43              @vNodes[[nodes[i]]]@sValNames[j]==1){
44              states[i]=names(network@vLayers[[il]]
45              @vNodes[[nodes[i]]]@sValNames[j])
46                            break
47                      }
48              }
49     }
50
51     ## wnioskowanie
52     network2@vLayers[[il]]<-solveBayesianLay(network2@vLayers[[il]])
53     network2@vLayers[[il]]<-setPrimary(network2@vLayers[[il]],
54     vid_nodes=nodes, vid_states=states)
55
56
```

```
57      ## pobieranie wyników
58      preferenceFactor[il]<-paste(network2@vLayers[[il]]@sName,
59  network2@vLayers[[il]]@vNodes[[network2@vLayers[[il]]@sName]]@sValNames[1])
60  }
61
62  ## wyświetlenie wyników
63  print(preferenceFactor)
64
65  ## zapis wyników do pliku
66  write(preferenceFactor, file="wyniki.txt")
67
68  ------------------------------------------------------------------------
69  ---------------------------- Koniec ------------------------------------
70  ------------------------------------------------------------------------
```

Wydruk 7.2: Przykład fragmentu raportu, opisującego projekt systemu diagnostycznego

```
  ------------------------------------------------------------------------
  ---------------------------- Edytor wymagań ----------------------------
  ------------ Copyright © 2015 Marcin Amarowicz IPKM PolSl ---------
  ------------------------------------------------------------------------
  ----------------- Projekt systemu diagnostycznego ----------------------
  ------------------------------------------------------------------------

wymaganie funkcjonalne:
 r.10  Konieczność wykrywania obniżenia poziomu medium w zbiorniku T1 ↙
    ↳ poniżej wartości minimalnej pf = 0.9
reguła diagnostyczna:
 f.21  Przekroczenie dopuszczalnej wartości medium w zbiorniku T1 poniżej ↙
    ↳ wartości minimalnej pf = 0.49
podsystem: v_fk = 325,85
- czujnik poziomu cieczy: e.1 - SKC-100.95
- sygnalizator poziomu cieczy: e.2 - ELCLUWO 111S
-----------------------------------------
wymaganie funkcjonalne:
 ...

 ------------------------------------------------------------------------
 ------------------------------------------------------------------------
 ------------------------------------------------------------------------
```

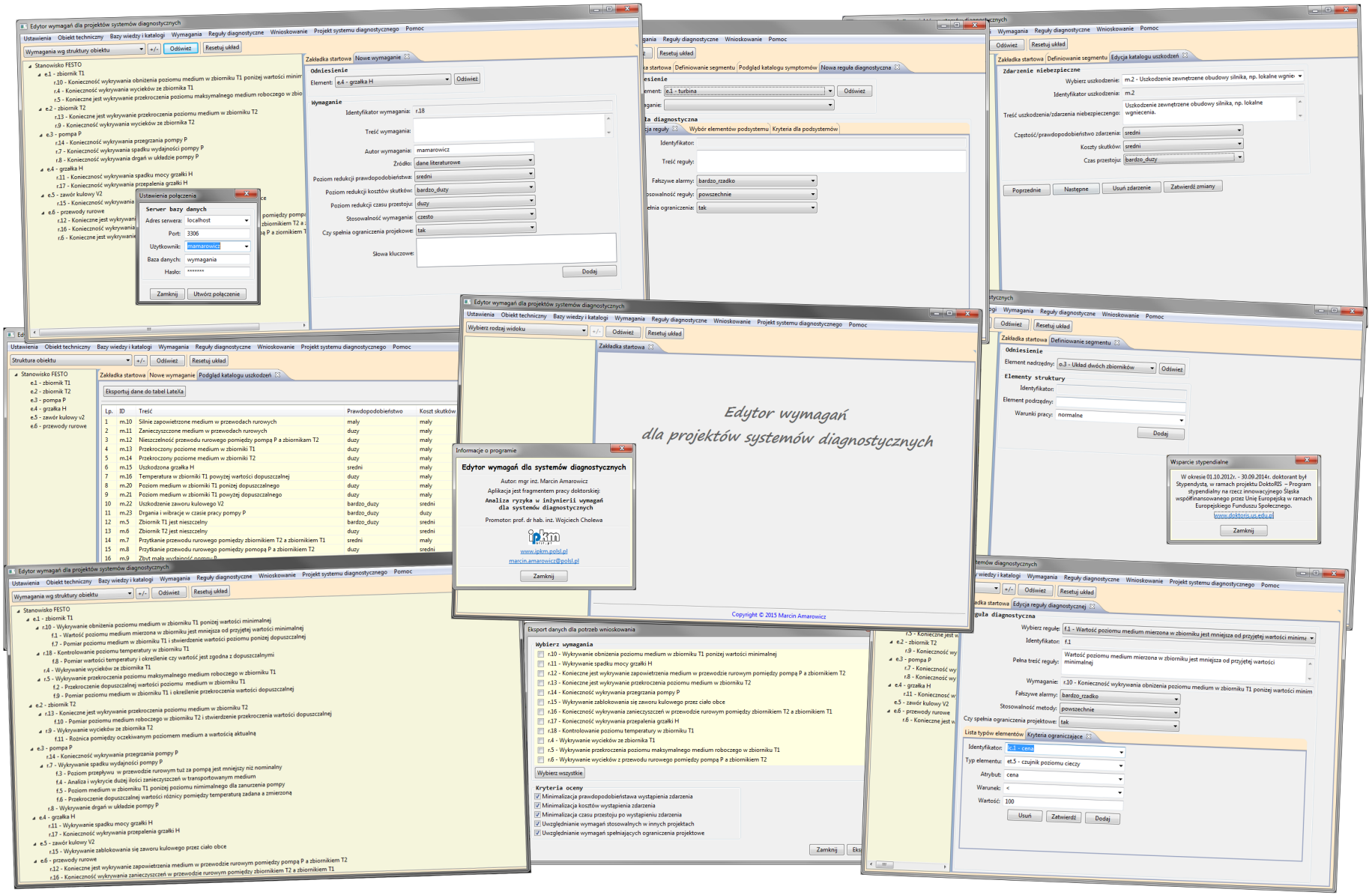
kolejno: wartość funkcji kryterialnej oraz dobrane elementy z katalogu

kolejno: identyfikator reguły, treść oraz wyznaczona wartość stopnia preferencji

kolejno: identyfikator wymagania, treść oraz wyznaczona wartość stopnia preferencji

7.3. Podsumowanie

Z wykorzystaniem opracowanego *Edytora wymagań*, możliwe jest przeprowadzenie większości etapów związanych z procesem opracowywania projektu systemu diagnostycznego, zgodnie z zaproponowanym podejściem projektowym. Niniejsza aplikacja, umożliwia pozyskiwanie niezbędnej wiedzy i danych na podstawie, których możliwe jest opracowanie sformalizowanego zapisu potrzeby projektowo-konstrukcyjnej. Dodatkowo z poziomu opracowanej aplikacji możliwe jest wygenerowanie niezbędnych modeli sieciowych, dla potrzeb procesu wnioskowania oraz przeprowadzanie procesu optymalizacji, celem wyodrębnienia najlepszego projektu sytemu diagnostycznego.



Rys. 7.10: Edytor wymagań; zestawienie wybranych formularzy i okien aplikacji

Rozdział 8

Weryfikacja opracowanej metody

W niniejszym rozdziale zawarto opis procesu weryfikacji zaproponowanego podejścia projektowego, do problemu opracowywania projektu systemu diagnostycznego. Opisano przykładowy obiekt techniczny, wyszczególniono fragmenty niezbędnych zbiorów wymagań, reguł diagnostycznych oraz podsystemów, będące podstawą opracowania sformalizowanego zapisu potrzeby. Przedstawiono sposób generowania zbioru potencjalnych rozwiązań projektu systemu diagnostycznego oraz wybór przykładowych rozwiązań optymalnych dla różnych danych wejściowych. Na zakończenie przedstawiono wnioski płynące z przeprowadzonego procesu weryfikacji.

8.1. Cel i zakres badań weryfikacyjnych

Celem przeprowadzonych badań weryfikacyjnych, było wykazanie przydatności zaproponowanego podejścia projektowego oraz opracowanego *Edytora wymagań*, dla potrzeb procesu projektowania systemu diagnostycznego, a tym samym praktyczne wykazanie słuszności przyjętych tez pracy.

Badania weryfikacyjne obejmowały takie czynności jak:

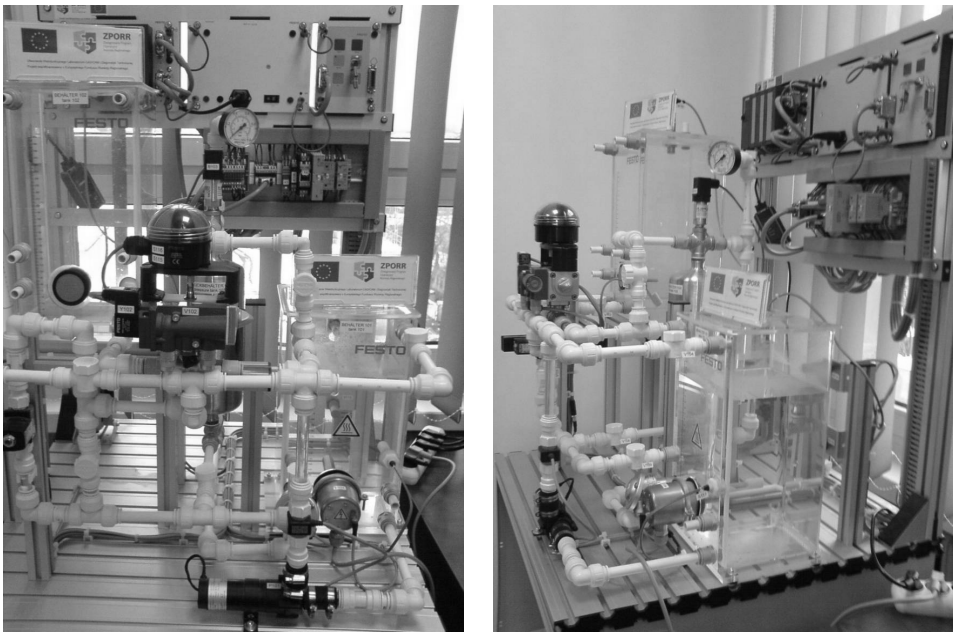
- identyfikację przyjętego obiektu technicznego, w tym jego struktury, możliwych uszkodzeń oraz katalogu typowych elementów systemu diagnostycznego,
- definiowanie wymagań funkcjonalnych, reguł diagnostycznych oraz podsystemów,
- ocenę zbioru wymagań oraz reguł diagnostycznych jako proces wnioskowania w modelach sieciowych,
- wybór optymalnego rozwiązania projektu systemu diagnostycznego.

Opisane w niniejszym rozdziale badania weryfikacyjne, przeprowadzone zostały z zastosowaniem opracowanego *Edytora wymagań*, opisanego w rozdz. 7.

8.2. Charakterystyka obiektu badań

Dla potrzeb przeprowadzenia badań weryfikacyjnych wybrano stanowisko badawcze FESTO S7 EduTrainer Compact Siemens S7-300 CPU313C, pokazane na rys. 8.1. Obiekt ten (Skupnik, 2009), jest fizycznym modelem instalacji przemysłowych, powszechnie stosowanych w przemyśle

chemicznym, spożywczym itp. Umożliwia on przeprowadzanie procesu diagnozowania symulowanych procesów przemysłowych. Schemat rozpatrywanego obiektu (za wyjątkiem: panelu sterowania, programowalnego sterownika logicznego PLC (ang. *Programmable Logic Controller*), przyłączy wejść/wyjść cyfrowych oraz analogowych, przetworników sygnałów prąd–napięcie, częstotliwość–napięcie, termopara–napięcie) pokazany został na rys. 8.2. Z punktu widzenia sterowania rozpatrywanego obiektu, możliwa jest regulacja takimi zmiennymi procesowymi jak: temperatura cieczy w zbiorniku $T1$, poziom cieczy w zbiorniku $T2$, ciśnienie powietrza w zbiorniku $T3$ oraz natężenie przepływu cieczy mierzone przez czujnik FS.

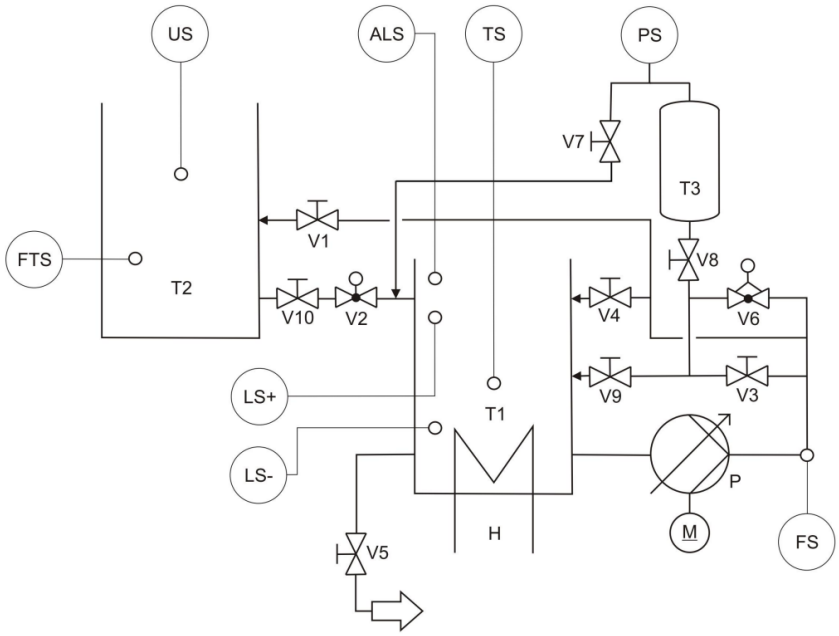


Rys. 8.1: Stanowisko FESTO S7 EduTrainer Compact Siemens S7-300 CPU313C (Skupnik, 2009)

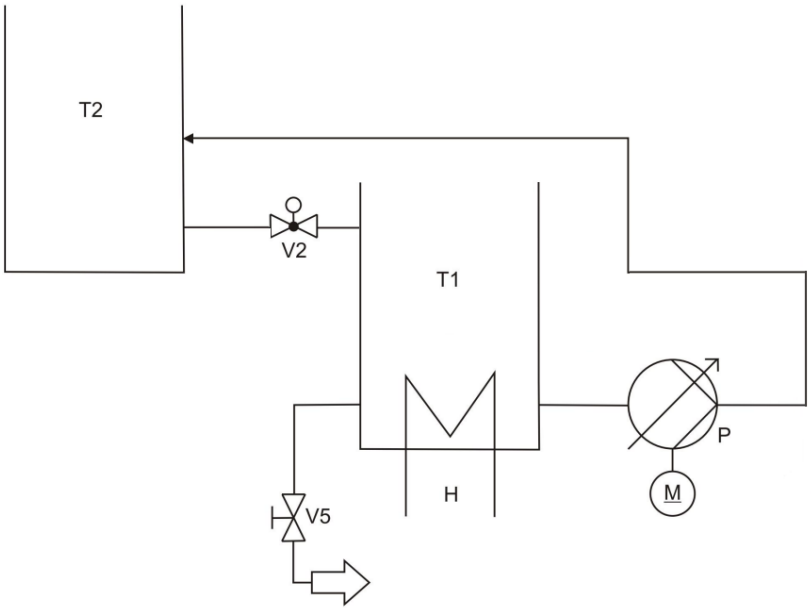
Odpowiednie stany rozpatrywanego obiektu, mogą być zadawane poprzez zamknięcie/otwarcie zaworów ręcznych v_i. Przykładowo Skupnik (2009), wyróżnił 9 stanów funkcjonalnych działania tego obiektu, poczynając od stanu postoju, poprzez przepompowywanie cieczy ze zbiornika $T1$ do zbiorników $T2$ lub $T3$, upuszczenie cieczy ze zbiorników $T2$ lub $T3$ do zbiornika $T1$ itp.

Dla potrzeb weryfikacji opracowanego podejścia projektowego, przyjęto że rozpatrywany będzie tylko jeden stan funkcjonalny (rys. 8.3), polegający na przepompowywaniu cieczy ze zbiornika $T1$ do zbiornika $T2$. Ciecz w zbiorniku $T1$ podgrzewana jest za pomocą grzałki H, natomiast po przekroczeniu ustalonego poziomu w zbiorniku $T2$ jest ona upuszczana do zbiornika $T1$. Z punktu widzenia rozpatrywanego zagadnienia, nie są istotne zastosowane w układzie czujniki pomiarowe.

Dla tak przedstawionego obiektu technicznego, możliwe jest zdefiniowanie potrzeby mówiącej o konieczności diagnozowania aktualnego stanu technicznego rozpatrywanego obiektu, celem oceny ryzyka technicznego pojawiającego się w trakcie jego eksploatacji. Potrzeba ta, może zostać zaspokojona poprzez zastosowanie odpowiednio opracowanego systemu diagnostycznego. Etapy realizacji tego zadania, z wykorzystaniem zaproponowanego podejścia projektowego, w tym opracowanego *Edytora wymagań*, zostały pokazane w kolejnych podrozdziałach.



Rys. 8.2: Schemat stanowiska FESTO S7 EduTrainer Compact Siemens S7-300 CPU313C (FESTO); ALS — przełącznik pływakowy poziomu alarmowego; FS — czujnik natężenia przepływu cieczy, FTS — przełącznik pływakowy, H — grzałka, LS- — czujnik zbliżeniowy dolny, LS+ — czujnik zbliżeniowy górny, M — silnik, P — pompa odśrodkowa, PS — czujnik ciśnienia, T1, T2 — odpowiednio zbiornik dolny i górny, T3 — zbiornik ciśnieniowy, TS — czujnik temperatury, US — ultradźwiękowy czujnik poziomu, V2 — zawór kulowy, V6 — zawór proporcjonalny, Vi — zawory ręczne (i = 1, 3, 4, 5, 7, 8, 9, 10; V5 — zawór spustowy)



Rys. 8.3: Uproszczony schemat stanowiska FESTO S7 EduTrainer Compact, obejmujący rozpatrywany przypadek; H — grzałka, M — silnik, P — pompa odśrodkowa, T1, T2 — odpowiednio zbiornik dolny i górny, V2 — zawór kulowy, V5 – zawór spustowy ręczny

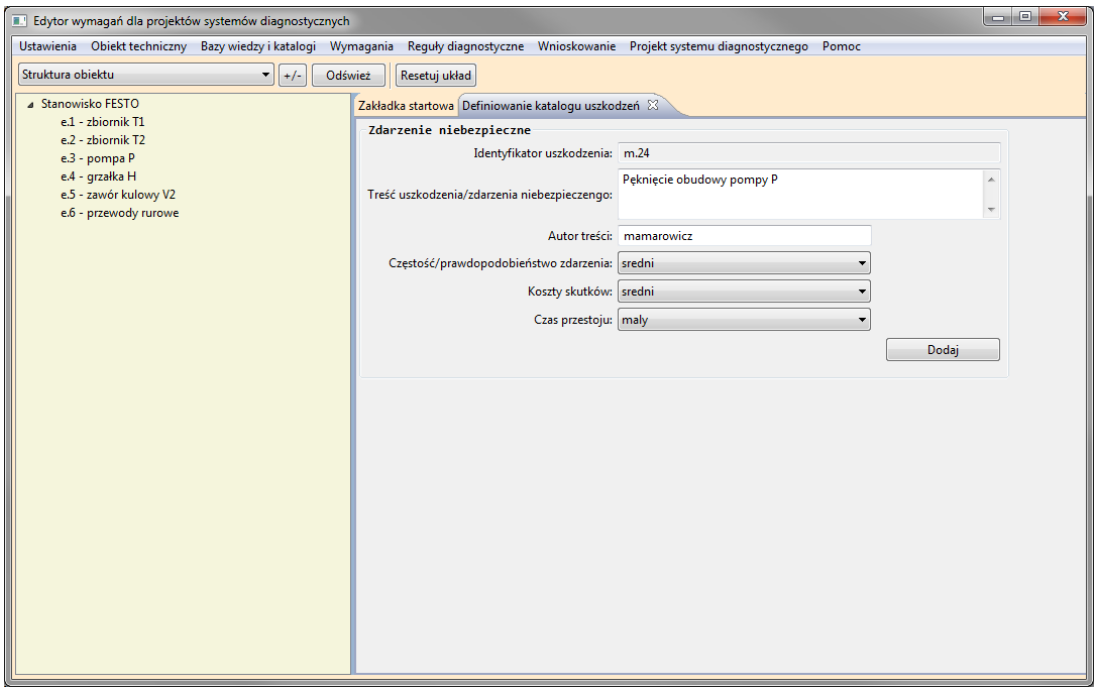
8.3. Proces pozyskiwania wymagań oraz formalizacja potrzeby

Proces projektowania systemu diagnostycznego, z zastosowaniem zaproponowanego podejścia projektowego, rozpoczyna się od dekompozycji obiektu technicznego na funkcjonalne podze-

społy i elementy. W rozpatrywanym przypadku możliwe jest wyodrębnienia takich elementów struktury obiektu jak: zbiornik $T1$, zbiornik $T2$, pompa P, grzałka H, zawór kulowy $V2$ oraz przewody rurowe.

W kolejnym etapie, należy zdefiniować potencjalne banki wiedzy, obejmujące zbiory uszkodzeń oraz symptomów a także katalogi potencjalnych elementów (czujników, kart pomiarowych itp.) przyszłego systemu diagnostycznego. W ogólnym przypadku, wiedza ta gromadzona może być na podstawie istniejących danych literaturowych, danych producenta oraz ekspertów dziedzinowych. W rozpatrywanym przypadku wykorzystane zostały dostępne dane literaturowe m. in. (Skupnik, 2009; FESTO) oraz wybrane katalogi producentów. Proces opracowywania stosownych banków wiedzy, zrealizowany został za pośrednictwem opracowanego *Edytora wymagań*. Na rys. 8.4 pokazano ekran aplikacji wraz z formularzem przeznaczonym do definiowania uszkodzeń obiektu, natomiast w tab. 8.1 zamieszczono fragment zdefiniowanego katalogu. Z kolei na rys. 8.5 pokazano ekran aplikacji wraz ze zdefiniowanym katalogiem sygnalizatorów poziomu cieczy. Ogólnie, dla potrzeb badań weryfikacyjnych zdefiniowano bank wiedzy obejmujący:

- katalog uszkodzeń, zawierający 20 najważniejszych potencjalnych uszkodzeń i niesprawności rozpatrywanego obiektu,
- katalog symptomów, zawierający 25 symptomów,
- katalog elementów, 115 elementów pogrupowanych w 17 typów.

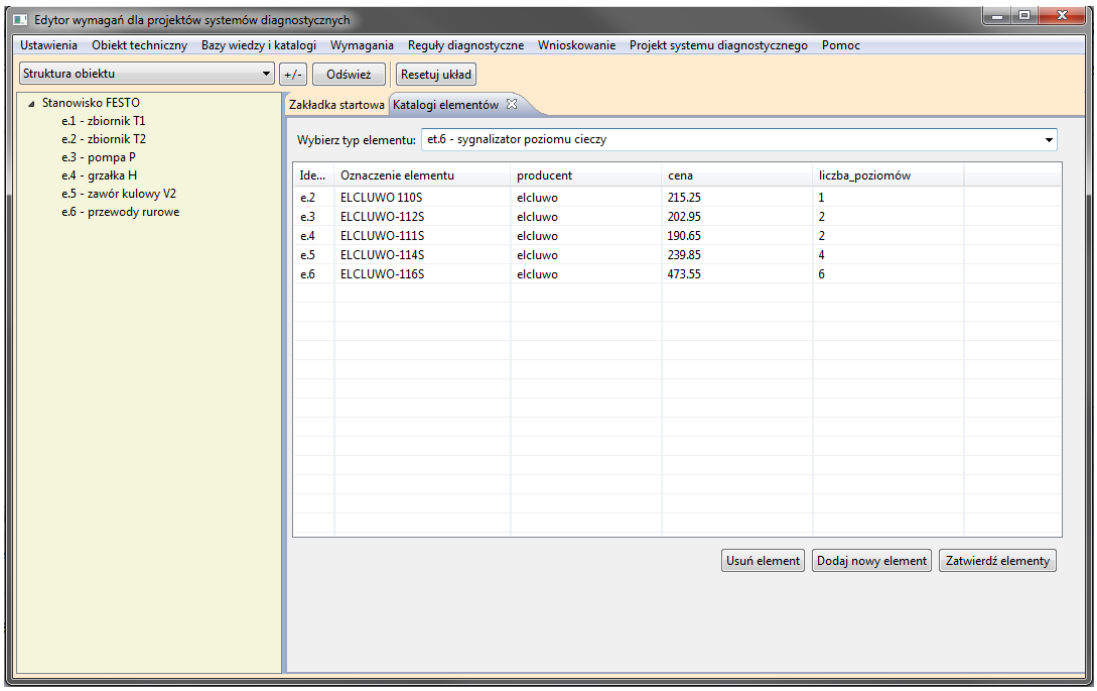


Rys. 8.4: Edytor wymagań; formularz przeznaczony do definiowania zbioru uszkodzeń

Tab. 8.1: Fragment zdefiniowanego zbioru uszkodzeń rozpatrywanego obiektu[1]

Lp.	ID	Treść	Prawdopodobieństwo	Koszty zdarzenia	Czas przestoju
1	m.10	Silnie zapowietrzone medium w przewodach rurowych	maly	maly	sredni
2	m.11	Zanieczyszczone medium w przewodach rurowych	duzy	maly	sredni
3	m.12	Nieszczelność przewodu rurowego pomiędzy pompą P a zbiornikiem T2	duzy	maly	bardzo_duzy
4	m.13	Przekroczony poziom medium roboczego w zbiorniku T1	duzy	maly	sredni
5	m.14	Przekroczony poziom medium roboczego w zbiorniku T2	duzy	maly	sredni
6	m.15	Uszkodzona grzałka H	sredni	maly	sredni
7	m.16	Temperatura w zbiorniku T1 powyżej wartości dopuszczalnej	duzy	maly	sredni
8	m.20	Poziom medium w zbiorniku T1 poniżej dopuszczalnego	duzy	maly	sredni
9	m.22	Uszkodzenie zaworu kulowego V2	bardzo_duzy	sredni	sredni
10	m.24	Pęknięcie obudowy pompy P	duzy	maly	sredni
11	m.5	Zbiornik T1 jest nieszczelny	bardzo_duzy	sredni	maly
12	m.6	Zbiornik T2 jest nieszczelny	duzy	sredni	maly
13	m.7	Przytkanie przewodu rurowego pomiędzy zbiornikiem T2 a zbiornikiem T1	sredni	maly	sredni
14	m.8	Przytkanie przewodu rurowego pomiędzy pomopą P a zbiornikiem T2	duzy	sredni	maly
15	m.9	Zbyt mała wydajność pompy P	sredni	sredni	maly

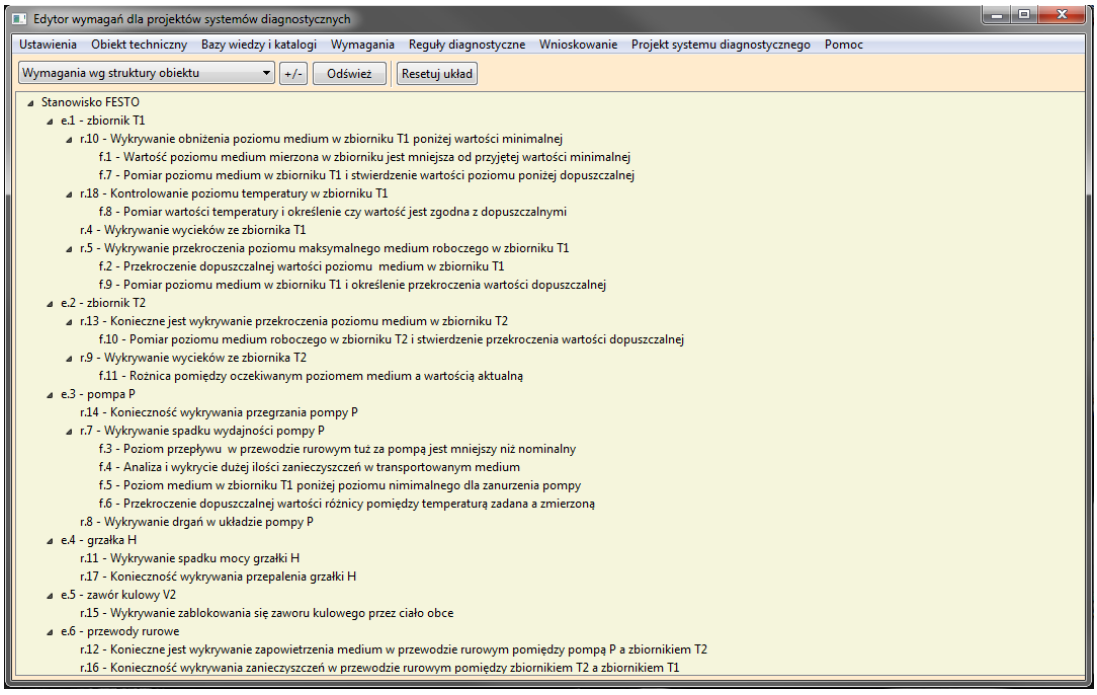
[1]Prezentowane tabele generowane są w sposób automatyczny przez opracowany *Edytor wymagań*. Z przyczyn praktycznych w opracowanej aplikacji wartości atrybutów zapisywane są bez polskich znaków diakrytycznych oraz spacji. W prezentowanych w rozprawie tabelach, utrzymano oryginalną formę zapisu wartości poszczególnych atrybutów.



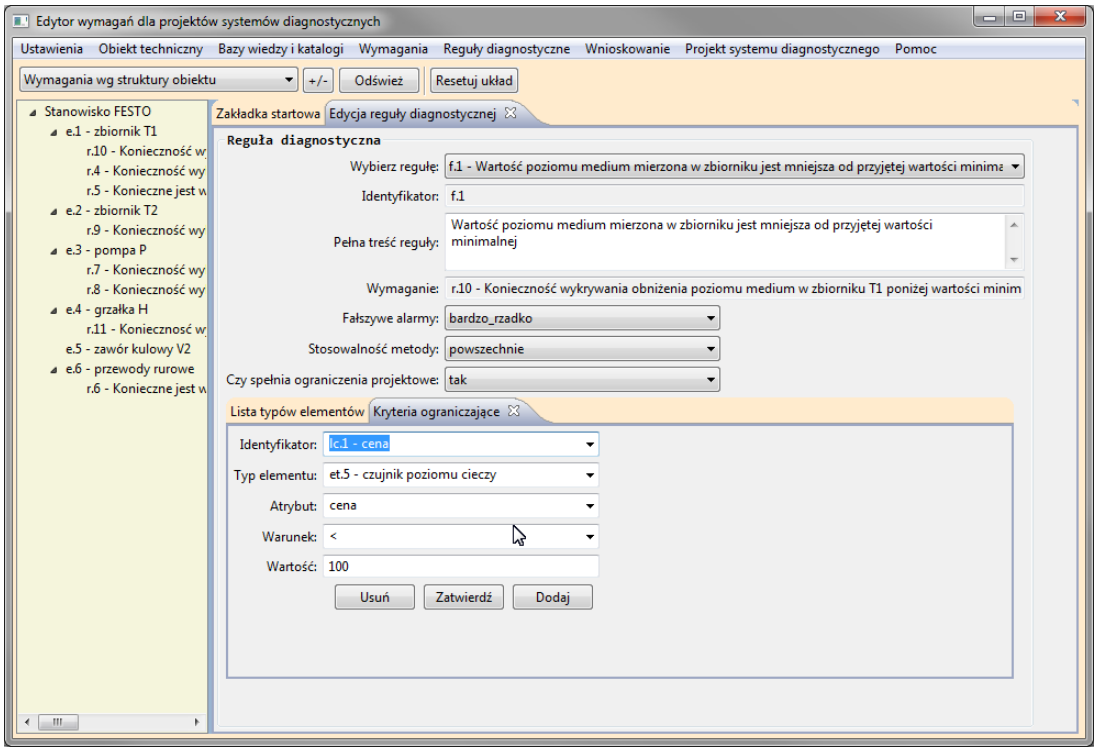
Rys. 8.5: Edytor wymagań; zdefiniowany katalog sygnalizatorów poziomu cieczy

Po zdefiniowaniu banków wiedzy przystąpiono do definiowania zbioru wymagań oraz reguł diagnostycznych, a tym samym do zapisu potrzeby w sformalizowany sposób. Dla rozpatrywanego obiektu zdefiniowano 15 wymagań funkcjonalnych (fragment opracowanego katalogu zawarto w tab. 8.2), które zostały przyporządkowane do wybranych podzespołów obiektu (rys. 8.6), wyodrębnionych w procesie dekompozycji. Poszczególnym wymaganiom, na podstawie dostępnych danych literaturowych, przypisano wartości poszczególnych atrybutów. W kolejnym kroku, każdemu z wymagań funkcjonalnych zostały przyporządkowane potencjalne reguły diagnostyczne, które mogą zostać użyte do wykrywania wybranych uszkodzeń. Fragment opracowanego zbioru reguł diagnostycznych (obejmującego łącznie 20 reguł) pokazano w tab. 8.3. Jednocześnie podczas definiowania reguł diagnostycznych, zdefiniowane zostały typy elementów wchodzące w skład potencjalnych podsystemów mogących realizować poszczególne z reguł diagnostycznych oraz kryteria ograniczające dla wybranych typów elementów wchodzących w skład tych podsystemów. Przykład jednego ze zdefiniowanych kryteriów ograniczających pokazano na rys. 8.7.

Poprzez wykonanie wymienionych czynności, zgromadzone zostały odpowiednie zbiory wymagań funkcjonalnych, reguł diagnostycznych oraz podsystemów opisujących w sformalizowany sposób potrzebę projektową. Kolejnym etapem proponowanego podejścia projektowego jest ocena zebranego zbioru wymagań funkcjonalnych i reguł diagnostycznych, z punktu widzenia ich przydatności do osiągnięcia przez projektowany system diagnostyczny założonej funkcji celu. W wyniku przeprowadzenia tego procesu wyodrębniony zostanie podzbiór wymagań, najlepiej opisujących oczekiwane funkcjonalności projektowanego systemu diagnostycznego.



Rys. 8.6: Edytor wymagań; przyporządkowanie wymagań funkcjonalnych oraz reguł diagnostycznych do podzespołów obiektu technicznego



Rys. 8.7: Edytor wymagań; przykład zdefiniowanego kryterium ograniczającego dla wybranej reguły diagnostycznej

Tab. 8.2: Fragment zdefiniowanego zbioru wymagań funkcjonalnych dla rozpatrywanego obiektu technicznego

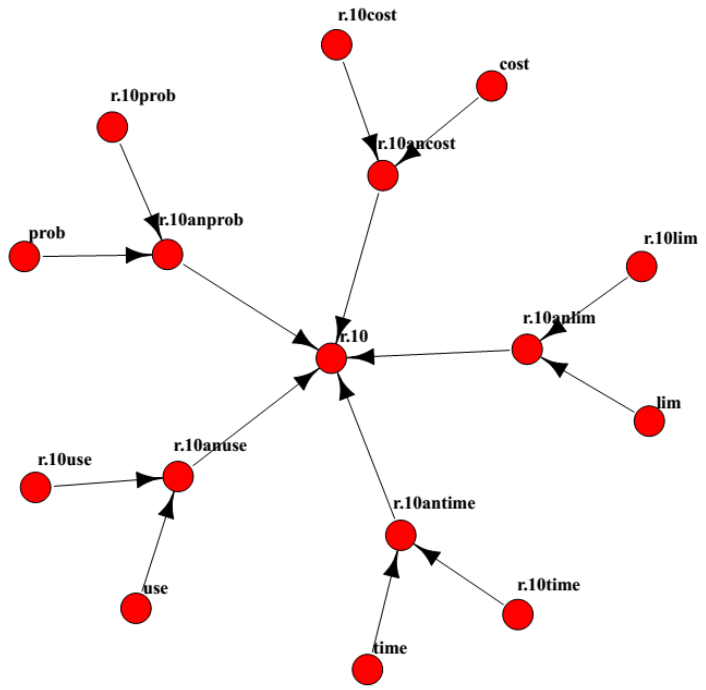
Lp.	ID	Treść wymagania	Stopień redukcji prawdopodobieństwa	Redukcja skutków	Redukcja czasu przestoju	Stosowalność
1	r.10	Wykrywanie obniżenia poziomu medium w zbiorniku T2 poniżej wartości minimalnej	maly	maly	maly	rzadko
2	r.11	Wykrywanie spadku mocy grzałki H	bardzo_duzy	sredni	maly	powszechnie
3	r.12	Wykrywanie zapowietrzenia medium w przewodzie rurowym pomiędzy pompą P a zbiornikiem T2	sredni	sredni	sredni	czesto
4	r.13	Wykrywanie przekroczenia poziomu medium w zbiorniku T2	duzy	duzy	sredni	czesto
5	r.14	Wykrywanie przegrzania pompy P	bardzo_duzy	duzy	sredni	powszechnie
6	r.15	Wykrywanie zablokowania zaworu kulowego przez ciało obce	sredni	sredni	duzy	czesto
7	r.16	Wykrywanie zanieczyszczeń w przewodzie rurowym pomiędzy zbiornikiem T2 a zbiornikiem T1	sredni	maly	sredni	czesto
8	r.17	Wykrywanie przepalenia grzałki H	sredni	bardzo_duzy	duzy	czesto
9	r.18	Kontrolowanie poziomu temperatury w zbiorniku T1	sredni	bardzo_duzy	duzy	czesto
10	r.4	Wykrywanie wycieków ze zbiornika T1	duzy	sredni	sredni	powszechnie
11	r.5	Wykrywanie przekroczenia poziomu maksymalnego medium roboczego w zbiorniku T1	duzy	bardzo_duzy	duzy	powszechnie
12	r.6	Wykrywanie wycieków z przewodu rurowego pomiędzy pompą P a zbiornikiem T2	duzy	sredni	bardzo_duzy	powszechnie
13	r.7	Wykrywanie spadku wydajności pompy P	sredni	sredni	duzy	powszechnie
14	r.8	Wykrywanie drgań w układzie pompy P	bardzo_duzy	duzy	duzy	powszechnie
15	r.9	Wykrywanie wycieków ze zbiornika T2	duzy	sredni	sredni	czesto

Tab. 8.3: Fragment zdefiniowanego zbioru reguł diagnostycznych dla rozpatrywanego obiektu technicznego

Lp.	ID	Treść reguły	Wymaganie	Fałszywe alarmy	Stosowalność
1	f.1	Wartość poziomu medium mierzona w zbiorniku jest mniejsza od przyjętej wartości minimalnej	Wykrywanie obniżenia poziomu medium w zbiorniku T1 poniżej wartości minimalnej	bardzo_rzadko	powszechnie
2	f.10	Pomiar poziomu medium roboczego w zbiorniku T2 i stwierdzenie przekroczenia wartości dopuszczalnej	Konieczne jest wykrywanie przekroczenia poziomu medium w zbiorniku T2	bardzo_rzadko	powszechnie
3	f.11	Różnica pomiędzy oczekiwanym poziomem medium a wartością aktualną jest większa niż dopuszczalna	Wykrywanie wycieków ze zbiornika T2	rzadko	czesto
4	f.2	Przekroczenie dopuszczalnej wartości poziomu medium w zbiorniku T1	Wykrywanie przekroczenia poziomu maksymalnego medium roboczego w zbiorniku T1	bardzo_rzadko	powszechnie
5	f.3	Poziom przepływu w przewodzie rurowym tuż za pompą jest mniejszy niż nominalny	Wykrywanie spadku wydajności pompy P	bardzo_rzadko	powszechnie
6	f.4	Analiza i wykrycie dużej ilości zanieczyszczeń w transportowanym medium	Wykrywanie spadku wydajności pompy P	bardzo_rzadko	czesto
7	f.5	Poziom medium w zbiorniku T1 poniżej poziomu nominalnego dla zanurzenia pompy	Wykrywanie spadku wydajności pompy P	bardzo_rzadko	czesto
8	f.6	Przekroczenie dopuszczalnej wartości różnicy pomiędzy temperaturą zadaną a zmierzoną	Wykrywanie spadku mocy grzałki H	bardzo_rzadko	powszechnie
9	f.7	Pomiar poziomu medium w zbiorniku T1 i stwierdzenie wartości poziomu poniżej dopuszczalnego	Wykrywanie obniżenia poziomu medium w zbiorniku T1 poniżej wartości minimalnej	bardzo_rzadko	powszechnie
10	f.8	Pomiar wartości temperatury i określenie czy wartość jest zgodna z dopuszczalną	Kontrolowanie poziomu temperatury w zbiorniku T1	bardzo_rzadko	powszechnie
11	f.9	Pomiar poziomu medium w zbiorniku T1 i określenie przekroczenia wartości dopuszczalnej	Wykrywanie przekroczenia poziomu maksymalnego medium roboczego w zbiorniku T1	bardzo_rzadko	powszechnie

8.4. Ocena zebranego zbioru wymagań oraz optymalizacja postaci systemu diagnostycznego

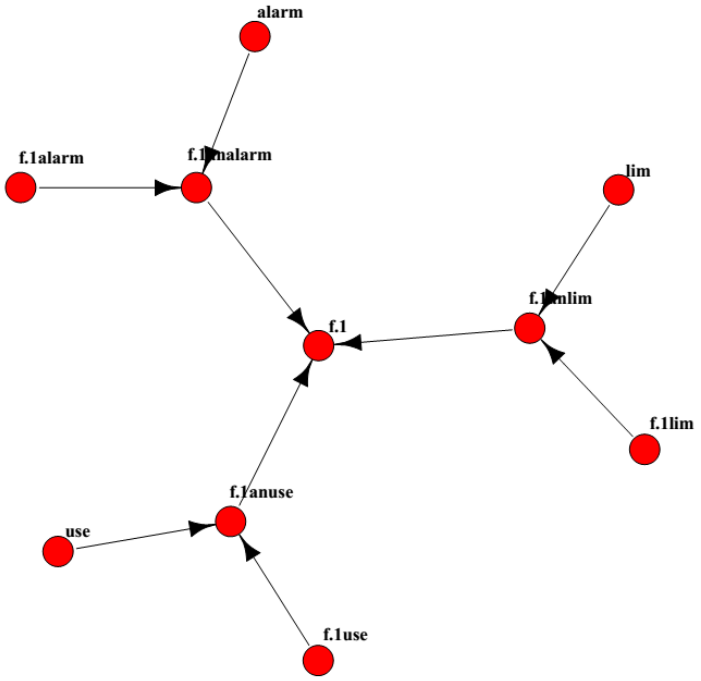
Proces oceny zebranych zbiorów wymagań funkcjonalnych oraz reguł diagnostycznych, realizowany jest jako proces wnioskowania w modelach sieciowych reprezentujących posiadaną wiedzę. Proces ten przeprowadzany jest, za pomocą funkcji dostępnych w pakiecie *REx*, opracowanym w ramach środowiska *R*. W tym celu z poziomu *Edytora wymagań* wyeksportowano opracowane zbiory wymagań funkcjonalnych oraz reguł diagnostycznych do stosownych modeli sieciowych, wymaganych przez pakiet *REx*. Przykład wizualizacji modelu sieciowego (z wykorzystaniem funkcji pakietu *REx*) dla wymagania funkcjonalnego oraz reguły diagnostycznej pokazano odpowiednio na rys. 8.8 oraz rys. 8.9. Struktura modelu jest zgodna z założeniami opisanymi w rozdz. 6.6. Właściwy proces wnioskowania, odbywał się poprzez zastosowanie skryptu pokazanego na wydruku 7.1. Otrzymane wyniki, czyli stopnie preferencji wyznaczone dla poszczególnych wymagań funkcjonalnych oraz reguł diagnostycznych, zostały zapisane do pliku tekstowego a następnie zaimportowane do *Edytora wymagań* celem dalszych działań. Przykład uzyskanych wyników pokazano na rys. 8.10.



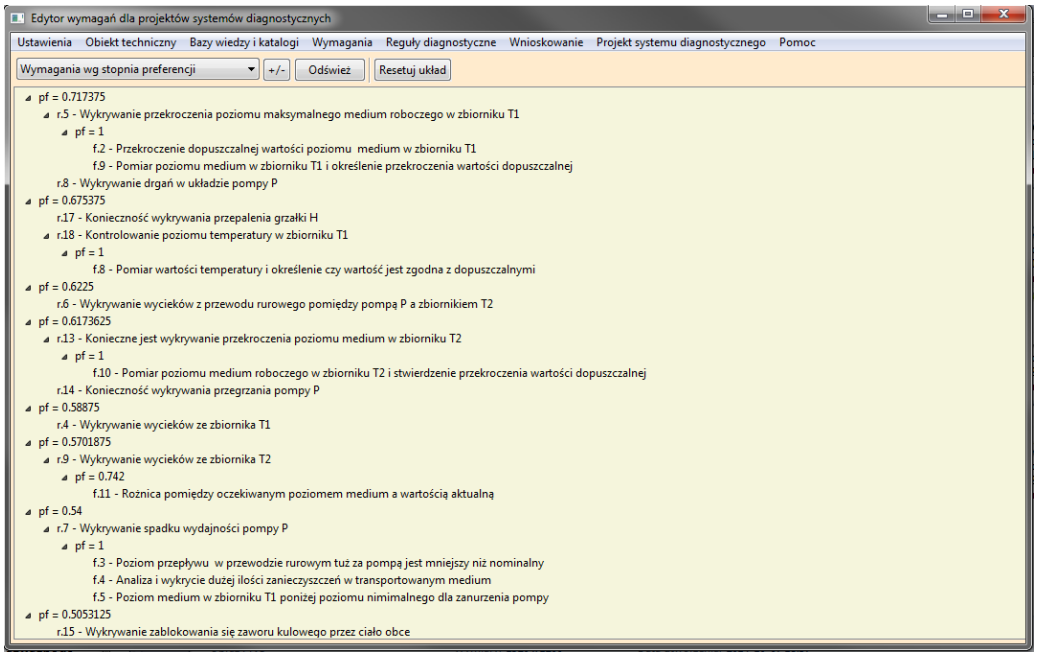
Rys. 8.8: Pakiet *REx*; wizualizacja modelu sieciowego dla wymagania funkcjonalnego, opis węzłów; *r.10* – wymaganie funkcjonalne, *r.1use*, *r.10prob*... – węzły przechowujące informację o wartościach wybranych atrybutów, *use*, *prob*,.... - węzły przechowujące informację o stopniu ważności danego atrybutu, *r.10anuse*, *r.10anprob*... – węzły dodatkowe

Proces wnioskowania przebiegał z uwzględnieniem wartości atrybutów przypisanych do poszczególnych wymagań funkcjonalnych i reguł diagnostycznych oraz przy uwzględnieniu stosownych wartości stopnia ważności poszczególnych atrybutów. Przeprowadzanie wielokrotnego procesu wnioskowania dla tych samych modeli sieciowych, ale z uwzględnieniem różnych danych wejściowych pozwala na rozpatrzenie różnych wariantów projektowanego systemu diagnostycznego.

Jako przykład możliwe jest rozpatrzenie wymagania *r.8* o treści *Wykrywanie drgań w ukła-*



Rys. 8.9: Pakiet *REx*; wizualizacja modelu sieciowego dla reguły diagnostycznej, opis węzłów; *f.1* – reguła diagnostyczna, *f.1use*, *r.1lim...* – węzły przechowujące informację o wartościach wybranych atrybutów, *use*, *lim*,.... - węzły przechowujące informację o stopniu ważności danego atrybutu, *f.1anuse*, *f.1anlim...* – węzły dodatkowe



Rys. 8.10: Edytor wymagań; lista wymagań funkcjonalnych oraz przypisanych do nich reguł diagnostycznych wraz z wyznaczonymi w procesie wnioskowania wartościami stopnia preferencji, uszeregowanie wg malejącej wartości stopnia preferencji

dzie pompy P. W tab. 8.4 zaprezentowano przykład wpływu wartości wybranych atrybutów oraz informacji o stopniu ważności tych atrybutów, na wartość współczynnika preferencji wyznacza-

nego w procesie wnioskowania. Jak można zauważyć, w zależności od wprowadzonych danych zmienia się preferencja co do uwzględniania danego wymagania w końcowym projekcie systemu diagnostycznego.

Tab. 8.4: Wpływ wartości wybranych atrybutów na wartość stopnia preferencji pf dla wymagania o treści: *Wykrywanie drgań w układzie pompy P*, wejściowe wartości atrybutów: Poziom redukcji prawdopodobieństwa: bardzo_duży, Poziom redukcji kosztów: duzy, Poziom redukcji czasu przestoju: duzy, Stosowalność wymagania: powszechnie, Czy spełnia ograniczenia projektowe: tak, w procesie wnioskowania uwzględniane są wszystkie atrybuty (stopień ważności = tak)

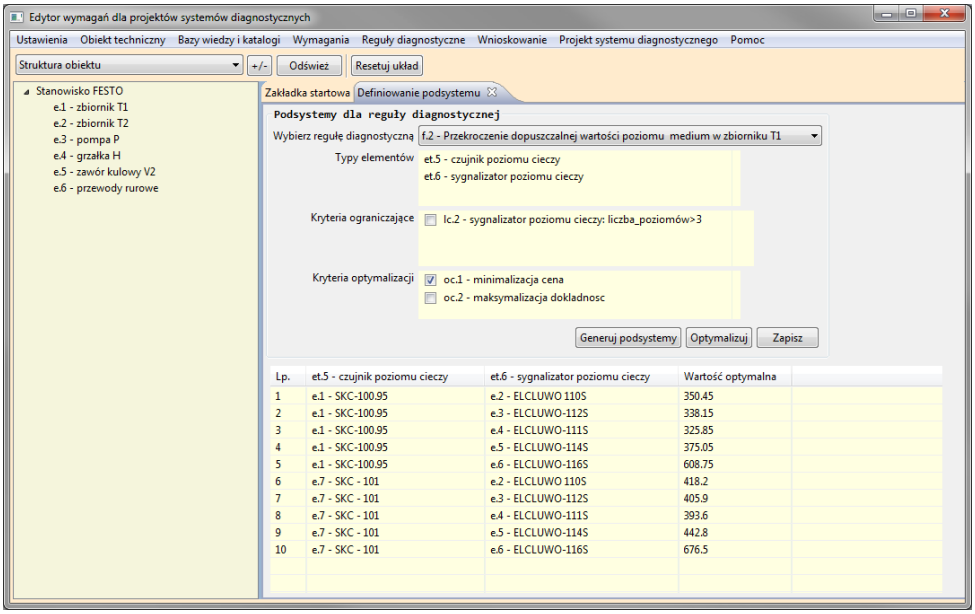
	Poziom redukcji		Stopień ważności		
Lp.	prawdopodobieństwa	kosztów	prawdopodobieństwa	kosztów	pf
1	bardzo_duzy	duży	nie	nie	0,17
2	maly	duży	tak	nie	0,34
3	maly	maly	tak	tak	0,48
4	bardzo_duzy	maly	tak	tak	0,89
5	bardzo_duzy	duży	tak	tak	0,96

Na podstawie wyznaczonych wartości stopnia preferencji, wyodrębniony został pewien podzbiór wymagań *subREQ*, zawierający wymagania najlepiej opisujące oczekiwane funkcjonalności projektowanego systemu diagnostycznego. O tym które z wymagań, powinny należeć do tego podzbioru, zdecydować musi każdorazowo użytkownik aplikacji, biorąc pod uwagę wartości stopnia preferencji. Dodatkowo w ramach każdego z wymagań wyznaczone zostały optymalne reguły diagnostyczne, przy czym rozpatrywane są przypadki w których jakość reguły diagnostycznej jest bardzo niska, zatem nie jest zasadne stosowanie związanego z nią wymagania, jak również możliwe są przypadki, w których dla tego samego wymagania dla co najmniej dwóch reguł diagnostycznych, wyznaczone zostały zbliżone wartości stopnia preferencji. O tym która z reguł w tym przypadku powinna być brana pod uwagę w ramach dalszych rozważań, rozstrzyga użytkownik aplikacji.

Kolejny etap w proponowanym podejściu projektowym to wygenerowanie podsystemów diagnostycznych na podstawie typów elementów przyporządkowanych do poszczególnych reguł diagnostycznych. W opracowanym *Edytorze wymagań*, proces ten odbywa się w sposób automatyczny. W ogólnym przypadku liczba wszystkich możliwych podsystemów, jakie mogą być zastosowane do realizacji i-tej reguły diagnostycznej wynosi:

$$\prod_{j} card(etype_{j,i}), \tag{8.1}$$

gdzie $card(etype_{j,i})$ jest liczbą elementów dostępnych w ramach j-tego typu elementu, wchodzącego skład definicji danego podsystemu. Liczba możliwych podsystemów może być ograniczana przez zdefiniowane kryteria ograniczające. Przykład wygenerowanej listy podsystemów dla wybranej reguły diagnostycznej pokazano na rys. 8.11. W pokazanym przykładzie, w definicji podsystemu zawarte są 2 typy elementów; czujnik poziomu cieczy - 2 elementy oraz sygnalizator poziomu cieczy - 5 elementów, co daje łącznie 10 możliwych podsystemów. Po zastosowaniu kryterium ograniczającego, związanego z minimalną liczbą sygnalizowanych poziomów (3 poziomy), liczba potencjalnych podsystemów maleje jedynie do 4 rozwiązań. Wygenerowane podsystemy są oceniane przy uwzględnieniu zdefiniowanych kryteriów optymalizacji, przy czym kryteria te są skorelowane z istniejącymi atrybutami poszczególnym elementów podsystemów. Zdefiniowane kryteria optymalizacji, mogą mieć zasięg lokalny albo globalny. Kryteria lokalne związane są



Rys. 8.11: Edytor wymagań; lista możliwych podsystemów dla wybranej reguły diagnostycznej

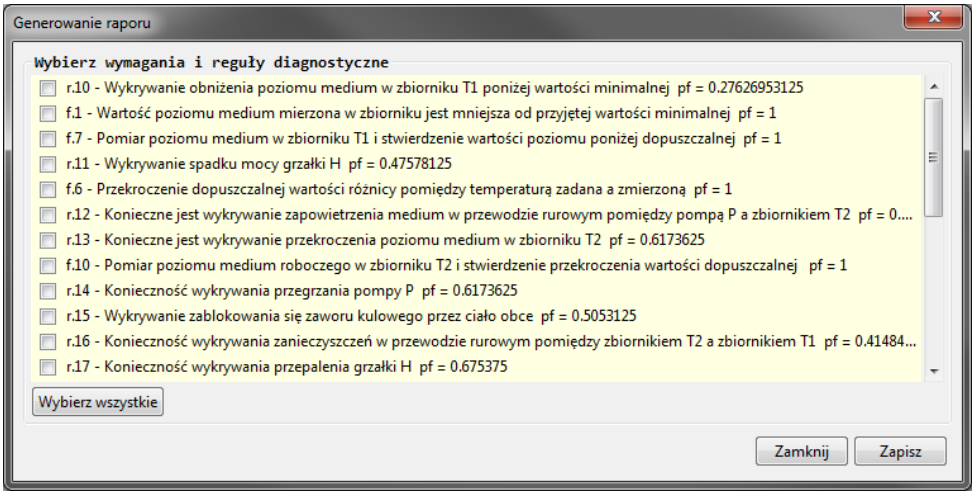
z wybranymi podsystemami, dla których zdefiniowane są specyficzne atrybuty. Natomiast kryteria globalne są bardziej ogólne i dotyczą atrybutów które są przypisane do wszystkich elementów podsystemów. Przykładem takiego atrybutu może być koszt jednostkowy elementu. Przykład wyników przeprowadzonej optymalizacji postaci podsystemu pokazany został na rys. 8.11.

W rozdz. 6 przyjęto, że przez system diagnostyczny rozumiany jest zbiór funkcjonalności tego systemu oraz zbiór elementów stosowanych do realizacji poszczególnych funkcjonalności. Dlatego też wynikiem zastosowania opracowanego podejścia projektowego do zagadnienia projektowania systemu diagnostycznego, jest lista sugerowanych funkcjonalności (w postaci par wymaganie funkcjonalne, reguła diagnostyczna) oraz odpowiadających im podsystemów. Lista ta generowana jest na podstawie wyników przeprowadzonego procesu wnioskowania, w czasie którego wymaganiom funkcjonalnym oraz regułom diagnostycznym przyporządkowywana jest wartość stopnia preferencji oraz na podstawie przeprowadzonego procesu optymalizacji postaci podsystemów. Przykłady takich list zamieszczono w kolejnym podrozdziale, natomiast na rys. 8.12 pokazano okno *Edytora wymagań*, z poziomu którego na podstawie wartości stopnia preferencji, użytkownik może wybrać najbardziej adekwatne wymagania funkcjonalne oraz reguły diagnostyczne dla potrzeb wygenerowania wspomnianej listy.

8.5. Przykłady projektów systemów diagnostycznych

W niniejszym podrozdziale zamieszczono przykłady projektów systemów diagnostycznych[2], w postaci listy funkcjonalności (pary: wymaganie funkcjonalne, reguła diagnostyczna) oraz odpowiadających tym funkcjonalnościom podsystemów diagnostycznych. Każdy z przedstawionych przypadków, rozpatrywany był dla tego samego zbioru wymagań funkcjonalnych oraz reguł diagnostycznych. Zmienne były wartości stopnia ważności wybranych atrybutów przyporządkowanych do wymagań funkcjonalnych oraz uwzględniane kryteria ograniczające dla wybranych podsystemów.

[2]Prezentowane wydruki wygenerowano automatycznie za pośrednictwem opracowanego *Edytora wymagań*. Poddano je lekkim modyfikacjom, celem zwiększenia przejrzystości wydruku.



Rys. 8.12: Edytor wymagań; wybór wymagań oraz reguł diagnostycznych dla potrzeb wygenerowania raportu podsumowującego projekt systemu diagnostycznego

Przypadek 1
Dane wejściowe:

- zbiór wymagań funkcjonalnych; rozpatrywane są wszystkie atrybuty poszczególnych wymagań,
- zbiór reguł diagnostycznych; rozpatrywane są wszystkie atrybuty poszczególnych reguł,
- zbiór podsystemów diagnostycznych, brak kryteriów ograniczających,
- kryterium optymalizacji; minimalny koszt systemu diagnostycznego.

Projekt systemu diagnostycznego:
Zaprezentowany na wydruku 8.1 fragment zestawienia funkcjonalności oraz podsystemów diagnostycznych, jest przykładowym rozwiązaniem projektu systemu diagnostycznego przy uwzględnianiu wszystkich zdefiniowanych atrybutów i przy poszukiwaniu rozwiązania cechującego się minimalnym kosztem.

Wydruk 8.1: Fragment raportu opisujący projekt systemu diagnostycznego dla przypadku nr 1

```
-------------------------------------------------------------------
--------------------------- Edytor wymagań ------------------------
------------ Copyright © 2015 Marcin Amarowicz IPKM PolSl ---------
-------------------------------------------------------------------
----------------- Projekt systemu diagnostycznego -----------------
-------------------------------------------------------------------

wymaganie funkcjonalne:
 r.5 - Wykrywanie przekroczenia poziomu maksymalnego medium roboczego w ↙
     ↳ zbiorniku T1  pf = 0.96
reguła diagnostyczna:
 f.2 - Przekroczenie dopuszczalnej wartości poziomu  medium w zbiorniku T1  ↙
     ↳ pf = 1
podsystem: v_fk = 325,85
 - czujnik poziomu cieczy: e.1 - SKC-100.95
 - sygnalizator poziomu cieczy: e.2 - ELCLUWO 111S
 ----------------------------------------
wymaganie funkcjonalne:
 r.8 - Wykrywanie drgań w układzie pompy  pf = 0.96
```

```
reguła diagnostyczna:
 f.22 – Przekroczenie dopuszczalnej wartości drgań w otoczeniu pompy   pf = 1
podsystem: v_fk = 1667,15
 – czujnik drgań: 627A01
 – karta pomiarowa DAQ: DAQ–2214
 ----------------------------------------
wymaganie funkcjonalne:
 r.18 – Kontrolowanie poziomu temperatury w zbiorniku T1  pf = 0.9
reguła diagnostyczna:
 f.8 – Pomiar wartości temperatury i określenie czy wartość jest zgodna z ↙
     ↳ dopuszczalnymi  pf = 1
podsystem: v_fk = 123.0
 – Czujnik temperatury: e.8 – T–120
 ----------------------------------------
wymaganie funkcjonalne:
 r.7 – Wykrywanie spadku wydajności pompy P  pf = 0.72
reguła diagnostyczna:
 f.5 – Poziom medium w zbiorniku T1 poniżej poziomu minimalnego dla ↙
     ↳ zanurzenia pompy  pf = 1
podsystem: v_fk = 325,85
 – czujnik poziomu cieczy: e.1 – SKC–100.95
 – sygnalizator poziomu cieczy: e.2 – ELCLUWO 111S
 ----------------------------------------

 ...

 ------------------------------------------------------------
 ---------------------- Koniec ------------------------------
 ------------------------------------------------------------
```

Przypadek 2
Dane wejściowe:

- zbiór wymagań funkcjonalnych; rozpatrywane są wszystkie atrybuty poszczególnych wymagań,
- zbiór reguł diagnostycznych; rozpatrywane są wszystkie atrybuty poszczególnych reguł,
- zbiór podsystemów diagnostycznych, kryterium ograniczające:
  element: sygnalizator poziomu cieczy, atrybut: liczba_poziomów warunek: > 3,
- kryterium optymalizacji; minimalny koszt systemu diagnostycznego.

Projekt systemu diagnostycznego:
W stosunku do projektu systemu diagnostycznego zaprezentowanego na wydruku 8.1, projekt systemu wygenerowany w rozpatrywanym przypadku (wydruk 8.2), różni się rozwiązaniami podsystemów dla funkcjonalności związanej z pomiarem poziomu cieczy (wymagania $r.5$ i $r.7$). Spowodowane jest to wprowadzonym kryterium ograniczającym, mówiącym o tym że sygnalizator poziomu cieczy, powinien umożliwiać jednoczesną sygnalizację co najmniej 3 poziomów cieczy (wprowadzone ograniczenie wynika z rozpatrywania 3 wymagań związanych z pomiarem poziomu cieczy). W wyniku zastosowania tego kryterium, ograniczony zostaje zbiór możliwych podsystemów, co w konsekwencji prowadzi do innego optymalnego podsystemu (sygnalizator poziomu cieczy: model ELCLUWO 114S zamiast modelu ELCLUWO 111S) dla rozpatrywanych wymagań $r.5$ oraz $r.7$, a tym samym do innego optymalnego projektu systemu diagnostycznego.

Wydruk 8.2: Fragment raportu opisujący projekt systemu diagnostycznego dla przypadku nr 2

```
-------------------------------------------------------------------
-------------------------- Edytor wymagań -------------------------
------------ Copyright © 2015 Marcin Amarowicz IPKM PolSl ---------
-------------------------------------------------------------------
----------------- Projekt systemu diagnostycznego -----------------
-------------------------------------------------------------------

wymaganie funkcjonalne:
 r.5 - Wykrywanie przekroczenia poziomu maksymalnego medium roboczego w ↙
     ↳ zbiorniku T1  pf = 0.96
reguła diagnostyczna:
 f.2 - Przekroczenie dopuszczalnej wartości poziomu  medium w zbiorniku T1   ↙
     ↳ pf = 1
podsystem: v_fk = 375,55
 - czujnik poziomu cieczy: e.1 - SKC-100.95
 - sygnalizator poziomu cieczy: e.2 - ELCLUWO 114S
 ------------------------------------------
wymaganie funkcjonalne:
 r.8 - Wykrywanie drgań w układzie pompy  pf = 0.96
reguła diagnostyczna:
 f.22 - Przekroczenie dopuszczalnej wartości drgań w otoczeniu pompy   pf = 1
podsystem: v_fk = 1667,15
 - czujnik drgań: 627A01
 - karta pomiarowa DAQ: DAQ-2214
 ------------------------------------------
wymaganie funkcjonalne:
 r.18 - Kontrolowanie poziomu temperatury w zbiorniku T1  pf = 0.9
reguła diagnostyczna:
 f.8 - Pomiar wartości temperatury i określenie czy wartość jest zgodna z ↙
     ↳ dopuszczalnymi  pf = 1
podsystem: v_fk = 123.0
 - Czujnik temperatury: e.8 - T-120
 ------------------------------------------
wymaganie funkcjonalne:
 r.7 - Wykrywanie spadku wydajności pompy P  pf = 0.72
reguła diagnostyczna:
 f.5 - Poziom medium w zbiorniku T1 poniżej poziomu minimalnego dla ↙
     ↳ zanurzenia pompy  pf = 1
podsystem: v_fk = 375,05
 - czujnik poziomu cieczy: e.1 - SKC-100.95
 - sygnalizator poziomu cieczy: e.2 - ELCLUWO 114S
 ------------------------------------------

 ...

-------------------------------------------------------------------
---------------------------- Koniec -------------------------------
-------------------------------------------------------------------
```

Przypadek 3
Dane wejściowe:

- zbiór wymagań funkcjonalnych; zrezygnowano z rozpatrywania atrybutów związanych z redukcją kosztów oraz czasu przestoju,
- zbiór reguł diagnostycznych; rozpatrywane są wszystkie atrybuty poszczególnych reguł,
- zbiór podsystemów diagnostycznych, brak kryteriów ograniczających,
- kryterium optymalizacji; minimalny koszt systemu diagnostycznego.

Projekt systemu diagnostycznego:
W stosunku do projektu rozpatrywanego w przypadku nr 1, w niniejszym zrezygnowano z rozpatrywania dwóch atrybutów przypisanych do wymagania funkcjonalnego, tj. z informacji o poziomie redukcji kosztów oraz czasu przestoju. W wyniku takiego podejścia zmieniły się wyjściowe oczekiwania co do pożądanej funkcjonalności systemu diagnostycznego. Zamiast konieczności wykrywania przekroczenia przez medium dopuszczalnego poziomu w zbiorniku T1 (wymaganie $r.5$), najbardziej oczekiwaną funkcjonalnością stała się konieczność wykrywania drgań w układzie pompy P, czyli wymaganie $r.8$.

Wydruk 8.3: Fragment raportu opisujący projekt systemu diagnostycznego dla przypadku nr 3

```
 ------------------------------------------------------------------
 ------------------------- Edytor wymagań -------------------------
 ------------ Copyright © 2015 Marcin Amarowicz IPKM PolSl ---------
 ------------------------------------------------------------------
 ----------------- Projekt systemu diagnostycznego ----------------
 ------------------------------------------------------------------

wymaganie funkcjonalne:
 r.8 - Wykrywanie drgań w układzie pompy  pf = 0.75
reguła diagnostyczna:
 f.22 - Przekroczenie dopuszczalnej wartości drgań w otoczeniu pompy   pf = 1
podsystem: v_fk = 1667,15
 - czujnik drgań: 627A01
 - karta pomiarowa DAQ: DAQ-2214
 ----------------------------------------
wymaganie funkcjonalne:
 r.13 - Wykrywanie przekroczenia poziomu medium w zbiorniku T2  pf = 0.56
reguła diagnostyczna:
 f.10 - Pomiar poziomu medium roboczego w zbiorniku T2 i stwierdzenie ↙
     ↳ przekroczenia wartości dopuszczalnej  pf = 1
podsystem: v_fk = 325,85
 - czujnik poziomu cieczy: e.1 - SKC-100.95
 - sygnalizator poziomu cieczy: e.2 - ELCLUWO 111S
----------------------------------------
wymaganie funkcjonalne:
 r.9 - Wykrywanie wycieków ze zbiornika T2  pf = 0.56
reguła diagnostyczna:
 f.1 - Różnica pomiędzy oczekiwanym poziomem medium a wartością aktualną  pf ↙
     ↳ = 0.742
podsystem: v_fk = 325,85
 - czujnik poziomu cieczy: e.1 - SKC-100.95
 - sygnalizator poziomu cieczy: e.2 - ELCLUWO 111S
----------------------------------------
wymaganie funkcjonalne:
 r.5 - Wykrywanie przekroczenia poziomu maksymalnego medium roboczego w ↙
     ↳ zbiorniku T1  pf = 0.55
reguła diagnostyczna:
 f.2 - Przekroczenie dopuszczalnej wartości poziomu  medium w zbiorniku T1  ↙
     ↳ pf = 1
podsystem: v_fk = 325,85
 - czujnik poziomu cieczy: e.1 - SKC-100.95
 - sygnalizator poziomu cieczy: e.2 - ELCLUWO 111S
 ----------------------------------------
 ...

 ------------------------------------------------------------------
 ------------------------- Koniec ---------------------------------
 ------------------------------------------------------------------
```

Zaprezentowane przykłady przedstawiają możliwe projekty systemu diagnostycznego, przy różnych zestawach danych wejściowych. Dla potrzeb oceny jakości otrzymanych rozwiązań, możliwe jest ich porównanie z systemem monitorowania zainstalowanym na rozpatrywanym stanowisku badawczym (rozumianym jako zestaw czujników wyszczególnionych na schemacie na rys. 8.2), przy czym należy pamiętać, że opracowane stanowisko badawcze dla potrzeb symulacji procesów przemysłowych może być wyposażone w nadmiarowe czujniki lub też z pewnych względów, zrezygnowano z rejestrowania wybranych sygnałów. Niemniej jednak, można zauważyć bardzo dużą zgodność pomiędzy opracowanymi projektami potencjalnego systemu diagnostycznego a zastosowanym, w rozpatrywanym stanowisku badawczym, układem pomiarowo-kontrolnym.

8.6. Podsumowanie

Zaprezentowany w niniejszym rozdziale proces weryfikacji zaproponowanego podejścia projektowego, do zagadnienia opracowywania projektu systemu diagnostycznego, miał na celu:

- weryfikację poprawności zaproponowanego podejścia projektowego oraz zasadności wykorzystania opracowanego *Edytora wymagań*,
- wykazanie słuszności sformułowanych na początku rozprawy tez.

Badania weryfikacyjne związane były z próbą opracowania projektu systemu diagnostycznego dla przykładowego obiektu technicznego, którym było stanowisko badawcze FESTO S7 EduTrainer Compact. Dla wybranego obiektu technicznego przeprowadzono wszystkie etapy związane z zaproponowanym podejściem projektowym, tj. opracowano niezbędne banki wiedzy, przeprowadzono proces pozyskiwania wymagań funkcjonalnych oraz reguł diagnostycznych, celem zapisania potrzeby projektowo-konstrukcyjnej w sformalizowany sposób. W kolejnym etapie dla poszczególnych reguł diagnostycznych, wygenerowano, przy uwzględnieniu zdefiniowanych kryteriów ograniczających, zbiory potencjalnych podsystemów realizujących poszczególne z reguł. Zdefiniowane wymagania i reguły zostały ocenione w procesie wnioskowania, natomiast na podstawie zdefiniowanych kryteriów optymalizacji wyodrębniono optymalne rozwiązania poszczególnych podsystemów. Dzięki takiemu toku postępowania, otrzymano listy oczekiwanych funkcjonalności oraz odpowiadające im elementy podsystemów, będące projektami potencjalnego systemu diagnostycznego.

Na podstawie przeprowadzonego procesu weryfikacji oraz uzyskanych wyników, można stwierdzić że zaproponowane podejście projektowe z powodzeniem może być stosowane podczas wstępnego opracowywania projektów systemów diagnostycznych. Przyjęcie sformalizowanego zapisy potrzeby, umożliwia elastyczne uwzględnianie wielu różnych czynników, mających w wielu przypadkach olbrzymi wpływ na postać projektowanego systemu, a także umożliwia łatwiejsze wygenerowanie zbioru potencjalnych rozwiązań systemu diagnostycznego. Przeprowadzenie procesu oceny zebranego zbioru wymagań i reguł diagnostycznych, w formie procesu wnioskowania przeprowadzanego z zastosowaniem systemu doradczego (*Edytora wymagań*), pozwala rozważać ten proces jako formę negocjacji (podobnie jak w przypadku projektów z obszaru inżynierii oprogramowania), w których dla określonych danych wejściowych artykułowane są stosowne wymagania i reguły diagnostyczne najlepiej opisujące oczekiwane funkcjonalności najefektywniejszego systemu diagnostycznego.

Rozdział 9

Podsumowanie i wnioski

W niniejszym rozdziale zwięźle podsumowano opracowaną rozprawę doktorską. Wyszczególniono najważniejsze wnioski płynące z zaproponowanego podejścia projektowego do zagadnienia opracowywania projektu systemu diagnostycznego oraz sformułowano potencjalne kierunki dalszych prac związanych z rozwojem przedstawionego podejścia projektowego.

9.1. Podsumowanie

Przedmiotem badań opisanych w niniejszej rozprawie było nowe podejście projektowe, do zagadnienia opracowywania projektu systemu diagnostycznego. Zaproponowane podejście projektowe, bazuje na zastosowaniu powszechnie używanych w obszarze inżynierii oprogramowania zbiorów wymagań, do sformalizowanego zapisu potrzeby projektowo-konstrukcyjnej oraz wykorzystaniu wybranych technik sztucznej inteligencji do oceny zbioru wymagań oraz optymalizacji postaci systemu diagnostycznego.

Sformułowany w rozdz. 1.2 cel pracy został osiągnięty poprzez:

- opracowanie podejścia projektowego do zagadnienia opracowywania projektu systemu diagnostycznego,
- opracowanie formalnego sposobu zapisu definiowanych wymagań, celem uwzględniania m.in. wyników analizy ryzyka, dostępnych metod i technik pomiarowych itp.,
- opracowanie systemu doradczego, przeznaczonego do przeprowadzenia procesu pozyskiwania niezbędnych wymagań oraz następnie do przeprowadzenia procesu oceny wymagań funkcjonalnych oraz reguł diagnostycznych, przez pryzmat ich przydatności do osiągnięcia przez projektowany system diagnostyczny założonej funkcji celu,
- przeprowadzenie badań weryfikacyjnych pozwalających na ocenę zaproponowanego podejścia projektowego.

Na podstawie wyników otrzymanych w trakcie badań weryfikacyjnych, można stwierdzić słuszność postawionych tez, przy czym:

- słuszność tezy 1, wykazano poprzez realizację procesu pozyskiwania i zapisu zbioru wymagań funkcjonalnych i reguł diagnostycznych, w celu opracowania sformalizowanego zapisu potrzeb dla opracowywanego projektu systemu diagnostycznego,

- natomiast słuszność tezy 2, została wykazana poprzez opracowanie stosownego systemu doradczego, który został użyty podczas badań weryfikacyjnych do przeprowadzenia procesu oceny zebranego zbioru wymagań, w formie procesu wnioskowania, dla potrzeb określenia optymalnej postaci projektowanego systemu diagnostycznego.

9.2. Spostrzeżenia i wnioski

Uzyskane podczas weryfikacji zaproponowanego podejścia projektowego wyniki, pozwalają na sformułowanie metodą indukcji niezupełnej następujących spostrzeżeń oraz wniosków:

- możliwe jest zastosowanie zbiorów wymagań do sformalizowanego zapisu potrzeb, związanych z koniecznością opracowania projektu systemu diagnostycznego,
- proces oceny zebranego zbioru wymagań funkcjonalnych oraz reguł diagnostycznych, może być zrealizowany w formie procesu wnioskowania, przeprowadzonego z zastosowaniem systemu doradczego, w czasie którego dla różnych wariantów danych wejściowych, artykułowane są przez system doradczy adekwatne wymagania,
- modele sieciowe są wygodnym sposobem reprezentacji posiadanego zbioru wiedzy,
- stosowanie wielu różnych atrybutów opisujących definiowane wymagania funkcjonalne oraz reguły diagnostyczne, znacząco zwiększa jakość formalnego zapisu potrzeby,
- bazy wiedzy, tj. katalogi uszkodzeń oraz symptomów a także katalogi typowych elementów układów diagnostycznych, mogą być pozyskiwane niezależnie od siebie, np. przez ekspertów dziedzinowych.

9.3. Kierunki dalszych badań

Uwzględniając przedstawione w poprzednim podrozdziale wnioski, możliwe jest sformułowanie następującej listy zagadnień, które z punktu widzenia prezentowanych w rozprawie doktorskiej zagadnień są ciekawe, niosą znaczna wartość naukową i wymagają rozwiązania:

1. W zakresie przedstawionego podejścia projektowego:
    - wprowadzenie możliwości integracji wiedzy pozyskiwanej z wielu źródeł dla tych samych wymagań i reguł diagnostycznych,
    - opracowanie metody oceny ryzyka technicznego na poziomie ilościowym,
    - uwzględnienie w proponowanym podejściu projektowym automatycznej propagacji uszkodzeń, a tym samym propagacji niezbędnych wymagań.
2. W zakresie sposobu weryfikacji podejścia projektowego:
    - pozyskanie i rozpatrzenie bardziej szczegółowego zbioru wymagań funkcjonalnych i reguł diagnostycznych,
    - przeprowadzenie badań weryfikacyjnych dla bardziej złożonego obiektu technicznego,
    - weryfikacja zaproponowanego podejścia projektowego dla rzeczywistego obiektu, dla którego istnieje faktyczna potrzeba opracowania systemu diagnostycznego,
3. W zakresie opracowanego *Edytora wymagań*:
    - uelastycznienie edytora, poprzez rezygnację ze stałej liczby, z góry określonych atrybutów przypisanych do poszczególnych wymagań funkcjonalnych oraz reguł diagnostycznych, rezygnacja z dwuwartościowego opisu stopnia ważności atrybutu,

- implementacja rozwiązań pozwalających na ocenę ryzyka technicznego na poziomie ilościowym,
- implementacja rozwiązań pozwalających na import katalogów elementów z zewnętrznych baz danych,
- optymalizacja struktury stosowanej bazy danych,
- przeprowadzenie procesu wnioskowania z zastosowaniem innych modeli sieciowych, np. modeli intuicjonistycznych itp.,
- zautomatyzowanie procesu wnioskowania poprzez jego uruchamianie z poziomu *Edytora wymagań*.

Bibliografia

Abran A., Bourque P., Dupuis R., Moore J. W. i Tripp L. L. (Red.). (2004). *Guide to the Software Engineering Body of Knowledge - SWEBOK*. IEEE Press, Piscataway.

Adams J. (2002). *Risk*. Taylor & Francis, Londyn.

Adams M. L. (2010). *Rotating machinery vibration: from analysis to troubleshooting*. CRC Press, Boca Raton.

Alfor M. i Lawson J. (1979). *Software Requirements Engineering Methodology (Development)*. TRW Defense and Space Systems Group. RADC-TR-79-168 Final Technical Report.

Amarowicz M. (2010). *Interfejs użytkownika systemu MMNET*. w: Cholewa W. (Red.), *Szkieletowy system doradczy MMNET*, wolumen 143 serii *Zeszyty*. Politechnika Śląska, Katedra Podstaw Konstrukcji Maszyn, Gliwice, strony 49–65.

Amarowicz M. (2014a). *Simulators for requirement definition of diagnostic systems*. w: Korbicz J. i Kowal M. (Red.), *Intelligent Systems in Technical and Medical Diagnostics*, wolumen 230 serii *Advances in Intelligent Systems and Computing*. Springer, Berlin-Heidelberg, strony 187–198.

Amarowicz M. (2014b). *Diagnostic Systems Design Process With Using The Sets Of Requirements*. Diagnostyka - Applied Structural Health, Usage and Condition Monitoring, 15(2): 19–26.

Amarowicz M. (2015). *Management of requirements for the projects of the diagnostic systems*. w: *Proceedings of Federeted Conference on Computer Science and Information Systems, 21th Conference on Knowledge Acquisition and Management, Łódź, 13 - 16 września 2015*, strony 1667—1672.

Amaya E. i Alvares A. (2010). *SIMPREBAL: An expert system for real-time fault diagnosis of hydrogenerators machinery*. w: *IEEE Conference on Emerging Technologies and Factory Automation*, ETFA 2010, strony 1–8.

Angeli C. (2010). *Diagnostic Expert Systems - From Expert's Knowledge to Real-Time Systems*. w: Sajja P. S. i Akerkar R. (Red.), *Advanced Knowledge Based Systems: Models, Applications & Research*, TMRF e-Book Series. strony 50–73.

Antoniou G., van Harmelen F., Plant R. i Vathienen J. (1998). *Verification and Validation of Knowledge-Based Systems*. AI Magazine, 19(3):123 – 126.

Barker V. E. i O'Connor D. March (1989). *Expert Systems for Configuration at Digital: XCON and Beyond*. Communication of the ACM, 32(3):298–318.

Barszcz T. (2006). *Systemy monitorowania i diagnostyki maszyn*. Akademia Górniczo-Hutnicza, Kraków.

Bartyzel M. (2012). *Oprogramowanie szyte na miarę. Jak rozmawiać z klientem, który nie wie, czego chce*. Helion, Gliwice.

Berg H.-P. (2010). *Risk management: procedures, methods and experiences*. Reliability and Risk Analysis: Theory and Applications, 1(2):79 – 85.

Birolini A. (2010). *Reliability Engineering: Theory and Practice*. Springer, Berlin.

Borenstein N. (1994). *Programming As If People Mattered: Friendly Programs, Software Engineering, and Other Noble Delusions*. Princeton University Press, Princeton.

Botten N., Kusiak A. i Raz T. (1989). *Knowledge bases: Integration, verification, and partitioning*. European Journal of Operational Research, 42(2):111 – 128.

Bowles J. B. i Peláez C. (1995). *Fuzzy logic prioritization of failures in a system failure mode, effects and criticality analysis*. Reliability Engineering & System Safety, 50(2):203 – 213.

Buchanan B. G. i Shortliffe E. H. (Red.). (1984). *Rule Based Expert Systems: The Mycin Experiments of the Stanford Heuristic Programming Project*. The Addison-Wesley series in artificial intelligence. Addison-Wesley Longman Publishing Corporation, Boston.

Carlson C. (2012). *Effective FMEAs: Achieving Safe, Reliable, and Economical Products and Processes Using Failure Mode and Effects Analysis*. Quality and Reliability Engineering Series. Wiley, Nowy Jork.

CCPS, 2011. (2011). *Layer of Protection Analysis: Simplified Process Risk Assessment*, wolumen 26 serii *A Center for Chemical Process Safety Concept Book*. Wiley, Nowy Jork.

Cempel C. (1982). *Podstawy wibroakustycznej diagnostyki maszyn*. WNT, Warszawa.

Cempel C. (1989). *Diagnostyka wibroakustyczna maszyn*. PWN, Warszwa.

Cempel C. i Tomaszewski F. (1992). *Diagnostyka maszyn. Zasady ogólne. Przykłady zastosowań*. Międzyresortowe Centrum Naukowe Eksploatacji Majątku Trwałego, Radom.

Cheng B. H. C. i Atlee J. M. (2007). *Research Directions in Requirements Engineering*. w: Briand L. C. i Wolf A. L. (Red.), *Future of Software Engineering 2007*, FOSE '07, Waszyngton. IEEE Computer Society, strony 285–303.

Cholewa A. i Psiuk K. (2005). *Diagnostyczne systemy monitorowania stanu maszyn*. Przegląd Mechaniczny, 5:48–53.

Cholewa W. (2010a). *Multimodal statement networks for diagnostic applications*. w: Sas P. i Bergen B. (Red.), *Proceedings of the International Conference on Noise and Vibration Engineering ISMA 2010*. Katholieke Universiteit Leuven, strony 817–830.

Cholewa W. (Red.). (2010b). *Szkieletowy system doradczy MMNET*, wolumen 143 serii *Zeszyty*. Politechnika Śląska, Katedra Podstaw Konstrukcji Maszyn, Gliwice.

Cholewa W. i Amarowicz M. (2012). *Acquisition of requirements for diagnostic systems.* Diagnostyka - Applied Structural Health, Usage and Condition Monitoring, 62(2):23–30.

Cholewa W. i Pedrycz W. (1987). *Systemy doradcze.* Wydawnictwo Politechniki Śląskiej, Gliwice.

Cholewa W., Kosmowski K. T. i Radkowski S. (Red.). (2008). *Modele systemów oceny ryzyka i diagnostyki technicznej*, wolumen 138 serii *Zeszyty.* Politechnika Śląska, Katedra Podstaw Konstrukcji Maszyn, Gliwice.

Cholewa W., Chrzanowski P. i Rogala T. (2009). *Diagnostyka z zastosowaniem sieci przekonań.* w: Korbicz J. i Kościelny J. M. (Red.), *Modelowanie, diagnostyka i sterowanie nadrzędne procesami. Implementacja w systemie DiaSter.* Wydawnictwo Naukowo-Techniczne, Warszawa, strony 279–308.

Cholewa W., Chrzanowski P., Rogala T. i Amarowicz M. (2011a). *Platforma do budowania sieci stwierdzeń.* Pomiary Automatyka Kontrola, 57(9):1079–1082.

Cholewa W., Amarowicz M., Chrzanowski P. i Rogala T. (2013). *Multimodal Statement Networks for Diagnostic Knowledge Modeling and Integration.* w: Filipe J. i Dietz J. (Red.), *Proceedings of the International Conference on Knowledge Engineering and Ontology Development KEOD 2013, Vilamoura, Algarve, Portugal, 19-22 September 2013.* SciTePress - Science and Technology Publications, strony 140–147.

Cholewa W. (2008). *Mechanical Analogy of Statement Networks.* International Journal of Applied Mathematics and Computer Science, 18(4):477–486.

Cholewa W. (2014). *Dynamical Statement Networks.* w: Awrejcewicz J. (Red.), *Applied Non-Linear Dynamical Systems*, wolumen 93 serii *Springer Proceedings in Mathematics Statistics.* Springer International Publishing, strony 351–361.

Cholewa W., Rogala T., Chrzanowski P. i Amarowicz M. (2011b). *Statement Networks Development Environment REx.* w: Jędrzejowicz P., Nguyen N. i Hoang K. (Red.), *Computational Collective Intelligence. Technologies and Applications*, wolumen 6923 serii *Lecture Notes in Computer Science.* Springer Berlin Heidelberg, strony 30–39.

Chow T. i Cao D.-B. (2008). *A survey study of critical success factors in agile software projects.* Journal of Systems and Software, 81(6):961–971.

Chrabski B. i Zmitrowicz K. (2014). *Inżynieria wymagań w praktyce.* Wydawnictwo Naukowe PWN, Warszawa.

Christoffersen P. F. (2011). *Elements of Financial Risk Management.* Academic Press, Londyn, wydanie 2.

Chung L. i do Prado Leite J. (2009). *On Non-Functional Requirements in Software Engineering.* w: Borgida A., Chaudhri V., Giorgini P. i Yu E. (Red.), *Conceptual Modeling: Foundations and Applications*, wolumen 5600 serii *Lecture Notes in Computer Science.* Springer Berlin Heidelberg, strony 363–379.

Crawley F., Preston M. i Tyler B. (2008). *HAZOP : Guide to Best Practice: Guidelines to Best Practice for the Process and Chemical Industries.* Institution of Chemical Engineers, Rugby, Warwickshire, wydanie 2.

CWA 15740:2008. (2008). *Risk-Based Inspection and Maintenance Procedures forEuropean Industry (RIMAP)*. CEN – Europeen Committe for Standardization.

Czech J. (2011). *Wybrane metody i narzędzia diagnostyki obiektów technicznych*. Utrzymanie Ruchu, 1:8–12.

de Gea J., Nicolas J., Aleman J., Toval A., Ebert C. i Vizcaino A. (2011). *Requirements Engineering Tools*. IEEE Software, 28:86 – 91.

Deutsch D. (2011). *The Beginning of Infinity: Explanations that Transform The World*. Penguin Books, Londyn.

Dietrych J. (1978). *System i konstrukcja*. WNT, Warszawa.

Ding X. S. (2008). *Model-based Fault Diagnosis Techniques: Design Schemes, Algorithms, and Tools*. Springer, Berlin.

Duda R., Gaschnig H. i Hart P. (1979). *Model design in the prospector consultant system for mineral exploration*. w: Michie D. (Red.), *Expert Systems in the Micro Electronic Age*. Edinburgh University Press, Edinburgh, strony 153–167.

ECSS-M-00-03A:2010. (2010). *Space project management. Risk management*. ESA Publications Division.

Ekstrand M. D., Riedl J. T. i Konstan J. A. (2010). *Collaborative Filtering Recommender Systems*. Foundations and Trends® in Human Computer Interaction, 4(2):81–173.

Ericson C. (2005). *Hazard Analysis Techniques for System Safety*. Wiley, Nowy Jork.

Falliere N., Murchu L. O. i Chien E. (2010). *W32.stuxnet dossier*. Raport techniczny firmy Symantec, dostępne online: `http://www.symantec.com/content/en/us/enterprise/media/security_response/whitepapers/w32_stuxnet_dossier.pdf`, 5 września 2015.

FESTO. (2015). *Qualification solutions for process automation*. Festo Didactic GmbH & Co. KG.

Fidali M. (2013). *Metodyka termograficznej diagnostyki obiektów technicznych*. Wydawnictwo Naukowe Instytutu Technologii Eksploatacji – PIB, Radom.

Firesmith D. (2004). *Prioritizing Requirements*. Journal of Object Technology, 3(8):35–47.

Fischer D. W. (1981). *Planning for large-scale accidents: Learning from the Three Mile Island accident*. Energy, 6(1):93 – 108.

Frawley W. J., Piatetsky-Shapiro G. i Matheus C. J. (1992). *Knowledge Discovery in Databases: An Overview*. AI Magazine, 13(3):57–70.

Gabbar H. (2007a). *Design of Intelligent Fault Diagnostic System (FDS)*. Process Safety and Environmental Protection, 85(6):566 – 578.

Gabbar H. A. (2007b). *Improved qualitative fault propagation analysis*. Journal of Loss Prevention in the Process Industries, 20(3):260 – 270.

Gao J., Li G. i Gao Z. (2008). *Fault propagation analysis for complex system based on small-world network model*. w: *Annual Reliability and Maintainability Symposium, RAMS 2008*, strony 359–364.

Gierek A. (2005). *Zużycie tribologiczne*. Wydawnictwo Politechniki Śląskiej, Gliwice.

Gómez-Pérez A., Fernández-López M. i Corcho O. (2007). *Ontological Engineering: with examples from the areas of Knowledge Management, e-Commerce and the Semantic Web*. Advanced Information and Knowledge Processing. Springer-Verlag, Nowy Jork.

Grady R. B. i Caswell D. L. (1987). *Software Metrics: Establishing a Company-wide Program*. Prentice-Hall, Nowy Jork.

Greenberg H. i Cramer J. (Red.). (1991). *Risk Assessment and Risk Management for the Chemical Process Industry*. Wiley, Nowy Jork.

Grieves M. (2005). *Product Lifecycle Management: Driving the Next Generation of Lean Thinking*. McGraw-Hill Education, Nowy Jork.

Gruber T. R. (1993). *A translation approach to portable ontology specifications*. Knowledge Acquisition, 5(2):199 – 220.

Grünbacher P. i Boehm B. (2001). *EasyWinWin: a groupware-supported methodology for requirements negotiation*. w: *Proceedings Joint 8th European Software Engineering Conference (ESEC) and 9th ACM SIGSOFT Symposium on the Foundations of Software Engineering (FSE-9), Vienna, Austria*, strony 320–321.

Grünbacher P. i Seyff N. (2005). *Requirements Negotiation*. w: Aurum A. i Wohlin C. (Red.), *Engineering and Managing Software Requirements*. Springer Berlin Heidelberg, strony 143–162.

Hawking S., Russell S., Tegmark M. i Wilczek F. (2014). *Transcendence looks at the implications of artificial intelligence - but are we taking AI seriously enough?* The Independent, 1 maj 2014. dostępne online: `http://www.independent.co.uk/news/science/stephen-hawking-transcendence-looks-at-the-implications-of-artificial-intelligence--but-are-we-taking-ai-seriously-enough-9313474.html`, 5 września 2015.

Heath L. S. i Sioson A. A. (2009). *Semantics of Multimodal Network Models*. IEEE/ACM Transactions on Computational Biology and Bioinformatics, 6(2):271–280.

Hickey A. M. i Davis A. M. (2004). *A Unified Model of Requirements Elicitation*. Journal of Management Information Systems, 20(4):65–84.

Hoffmann M., Kuhn N., Weber M. i Bittner M. (2004). *Requirements for Requirements Management Tools*. w: *Proceedings of the Requirements Engineering Conference, 12th IEEE International*, RE '04, Waszyngton. IEEE Computer Society, strony 301–308.

Hyatt N. (2010). *Guidelines for Process Hazards Analysis (PHA, HAZOP), Hazards Identification, and Risk Analysis*. CRC Press, Richmond Hill.

IAEA. (2006). *Environmental Consequences of the Chernobyl Accident and their Remediation: Twenty Years of Experience*. Radiological Assesment Reports Series. International Atomic Energy Agency, Wiedeń.

IEEE Std 610. (1990). *IEEE Standard Glossary of Software Engineering Terminology*. IEEE Press.

IEEE Std 830. (1998). *IEEE Recommended Practice for Software Requirements Specifications*. IEEE Press.

Ilieva M. i Ormandjieva O. (2005). *Automatic Transition of Natural Language Software Requirements Specification into Formal Presentation*. w: Montoyo A., Muńoz R. i Métais E. (Red.), *Natural Language Processing and Information Systems*, wolumen 3513 serii *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, strony 392–397.

Isermann R. i Balle P. (1997). *Trends in the application of model-based fault detection and diagnosis of technical processes*. Control engineering practice, 5(5):709–719.

Jain L. i Martin N. (1998). *Fusion of Neural Networks, Fuzzy Systems and Genetic Algorithms: Industrial Applications*. International Series on Computational Intelligence. Taylor & Francis.

Jensen F. V. (1996). *Introduction to Bayesian Networks*. Springer-Verlag, Nowy Jork.

Jonkman S. N., van Gelder P. H. A. J. M. i Vrijling J. K. (2003). *An overview of quantitative risk measures for loss of life and economic damage*. Journal of Hazardous Materials, 99:1–30.

Jovanovic A. (2003). *Risk-based inspection and maintenance in power and process plants in Europe*. Nuclear Engineering and Design, 226(2):165 – 182. Selected and revised papers from the 2nd MPA-SAFE workshop 2001 on Safety Evaluation of Nuclear Plant Components.

Jovanovic A., J.M. B., Peters J. i Husta S. (2014). *Qualitative damage assessment as a part of the RBI (Risk-Based Inspection) process*. 40th MPA-Seminar, Materialprüfungsanstalt Universität Stuttgart, 6–7 październik 2014.

Kacprzak P. (2011). *Layer of protection analysis in industrial hazadous installations*. w: *Summer Safety and Reliability Seminars, Gdańsk–Sopot, 03–09 lipca 2011*, strony 79–86.

Kaplan S. i Garrick B. J. (1981). *On The Quantitative Definition of Risk*. Risk Analysis, 1(1): 11–27.

Karlsson J. i Ryan K. (1997). *A cost-value approach for prioritizing requirements*. IEEE Software, 14(5):67–74.

Kauer R., Jovanovic A., Angelsen S. i Vage G. (2004). *Plant Asset Management: RIMAP (Risk-Based Inspection and Maintenance for European Industries) - The European Approach*. w: *ASME/JSME 2004 Pressure Vessels and Piping Conference Risk and Reliability and Evaluation of Components and Machinery*, strony 183–192.

Khalajzadeh H., Dadkhah C. i Mansouri M. Oct (2011). *A review on applicability of expert system in designing and control of autonomous cars*. w: *2011 Fourth International Workshop on Advanced Computational Intelligence (IWACI)*, strony 280–285.

Khan F. I. i Haddara M. M. (2003). *Risk-based maintenance (RBM): a quantitative approach for maintenance/inspection scheduling and planning*. Journal of Loss Prevention in the Process Industries, 16(6):561 – 573.

Kirchsteiger C. (1999). *On the use of probabilistic and deterministic methods in risk analysis*. Journal of Loss Prevention in the Process Industries, 12(5):399 – 419.

Kościelny J. M. (2001). *Diagnostyka zautomatyzowanych procesów przemysłowych.* Akademicka Oficyna Wydawnicza Exit, Warszawa.

Kof L. (2004). *An Application of Natural Language Processing to Domain Modelling – Two Case Studies.* International Journal on Computer Systems Science Engineering, 20:37–52.

Korbicz J. i Kościelny J. M. (Red.). (2010). *Modeling, Diagnostics and Process Control: Implementation in the DiaSter System.* Springer-Verlag, Berlin.

Korbicz J., Kościelny J. M., Kowalczuk Z. i Cholewa W. (2002). *Diagnostyka procesów. Modele. Metody sztucznej inteligencji. Zastosowania.* WNT, Warszawa.

Koski T. i Noble J. (2009). *Bayesian Networks: An Introduction.* Wiley Series in Probability and Statistics. Wiley, Nowy Jork.

Krajewski M. (2014). *The Great Lightbulb Conspiracy.* IEEE Spectrum, 24 wrzesień 2014. dostępne online: `http://spectrum.ieee.org/geek-life/history/the-great-lightbulb-conspiracy`, 5 września 2015.

Kucharski T. (2002). *System pomiaru drgań mechanicznych.* WNT, Warszawa.

Lawrowski Z. (2008). *Tribologia: tarcie, zużywanie i smarowanie.* Oficyna Wydawnicza Politechniki Wrocławskiej, Wrocław.

Lee K. i Jo G. (1999). *Expert system for predicting stock market timing using a candlestick chart.* Expert Systems with Applications, 16(4):357 – 364.

Lee W., Grosh D., Tillman F. i Lie C. (1985). *Fault Tree Analysis, Methods, and Applications – A Review.* IEEE Transactions on Reliability, R-34(3):194–203.

Leffingwell D. i Widrig D. (2003). *Zarządzanie wymaganiami.* WNT, Warszawa.

Legutko S. (2004). *Podstawy eksploatacji maszyn i urządzeń.* WSiP, Poznań.

Liebowitz J. (1997). *The Handbook of Applied Expert Systems.* CRC Press, Boca Raton.

Ligęza A. (2006). *Logical Foundations for Rule-Based Systems.* Studies in Computational Intelligence. Springer, Berlin–Heidelberg.

Lim S. L. (2010). *Social Networks and Collaborative Filtering for Large-Scale Requirements Elicitation.* Praca doktorska, School of Computer Science and Engineering, University of New South Wales, Sydney, Australia.

Lim S. L. i Finkelstein A. (2012). *StakeRare: Using Social Networks and Collaborative Filtering for Large-Scale Requirements Elicitation.* IEEE Transactions on Software Engineering, 38(3): 707–735.

Lindsay R. K., Buchanan B. G., Feigenbaum E. A. i Lederberg J. (1993). *DENDRAL: a case study of the first expert system for scientific hypothesis formation.* Artificial Intelligence, 61 (2):209–261.

Lipowczan A. (1987). *Podstawy pomiarów hałasu.* Główny Instytut Górnictwa, Katowice-Warszawa.

Lupton D. (2013). *Risk: Second Edition*. Routlegde, Nowy Jork.

Maedche A. i Staab S. March (2001). *Ontology Learning for the Semantic Web*. IEEE Intelligent Systems, 16(2):72–79.

Matrosov A., Rodionov E., Harley D. i Malcho J. (2010). *Stuxnet under the microscope*. Raport techniczny firmy Eset, dostępne online: `http://go.eset.com/us/resources/white-papers/Stuxnet_Under_the_Microscope.pdf`, 5 września 2015.

McCarthy J., Minsky M., Rochester N. i Shannon C. (1955). *A Proposal for the Dartmouth Summer Research Project on Artificial Intelligence*. Dartmouth College, Hanover.

McNeil A. J., Frey R. i Embrechts P. (2015). *Quantitative Risk Management: Concepts, Techniques and Tools*. Princeton Series in Finance. Princeton University Press, Princeton.

Moczulski W. (1997). *Metody pozyskiwania wiedzy dla potrzeb diagnostyki maszyn*, wolumen 130 serii *Zeszyty Naukowe Politechniki Śląskiej. Monografie. Mechanika*. Politechnika Śląska, Gliwice.

Moczulski W. (2001). *Inductive acquisition of diagnostic knowledge for states tree with complex structure*. Mechanical Systems and Signal Processing, 15(4):813–825.

Moczulski W. (2002). *Metody pozyskiwania wiedzy diagnostycznej*. w: Korbicz J., Kościelny J. M., Kowalczuk Z. i Cholewa W. (Red.), *Diagnostyka procesów. Modele. Metody sztucznej inteligencji. Zastosowania*, wolumen 3 serii *Monografie Komitetu Automatyki i Robotyki Polskiej Akademii nauk*. WNT, Warszawa, strony 623–664.

Morel J. (1992). *Drgania maszyn i diagnostyka ich stanu technicznego*. Polskie Towarzystwo Diagnostyki Technicznej, Warszawa.

Morgan M., Henrion M. i Small M. (1992). *Uncertainty: A Guide to Dealing with Uncertainty in Quantitative Risk and Policy Analysis*. Cambridge University Press.

Mulawka J. (1996). *Systemy ekspertowe*. WNT, Warszawa.

Nan C., Khan F. i Iqbal M. (2008). *Real-time fault diagnosis using knowledge-based expert system*. Process Safety and Environmental Protection, 86(1):55–71.

Natt och Dag J., Gervasi V., Brinkkemper S. i Regnell B. (2005). *A Linguistic-Engineering Approach to Large-Scale Requirements Management*. IEEE Software, 22(1):32–39.

Nęcka E., Orzechowski J. i Szmura B. (2007). *Psychologia poznawcza*. PWN, Warszawa.

Negnevitsky M. (2005). *Artificial Intelligence: A Guide to Intelligent Systems*. Addison-Wesley, Boston, wydanie 2.

Nuseibeh B. i Easterbrook S. (2000). *Requirements engineering: A Roadmap*. w: *Proceedings of the Conference on The Future of Software Engineering*, ICSE '00, Nowy Jork. strony 35–46.

O'Connor P. i Kleyner A. (2011). *Practical Reliability Engineering*. Wiley, Nowy Jork, wydanie 5.

Oliferuk W. (2008). *Termografia podczerwieni w nieniszczących badaniach materiałów i urządzeń*. Biuro Gamma, Warszawa.

O'Riordan T. i Cameron J. (1994). *Interpreting the Precautionary Principle*, wolumen 2 serii *International law and sustainable development series*. Earthscan Publications Limited, Londyn.

Paté-Cornell M. E. (1993). *Learning from the Piper Alpha Accident: A Postmortem Analysis of Technical and Organizational Factors*. Risk Analysis, 13(2):215–232.

Pawletko R. (2007). *Pozyskiwanie wiedzy dla potrzeb ekspertowaego systemu diagnozowania okrętowego silnika spalinowego*. Diagnostyka, 3(43):49–54.

Petrović D. V., Tanasijević M., Milić V., Lilić N., Stojadinović S. i Svrkota I. (2014). *Risk assessment model of mining equipment failure based on fuzzy logic*. Expert Systems with Applications, 41(18):8157 – 8164.

Pieczyński A. (2003). *Reprezentacja wiedzy w diagnostycznym systemie ekspertowych*. Lubuskie Towarzystwo Naukowe, Zielona Góra.

Pólya G. (1957). *How to Solve It*. Doubleday, Garden City.

PN-EN ISO 12100:2012. (2012). *Bezpieczeństwo maszyn – Ogólne zasady projektowania – Ocena ryzyka i zmniejszanie ryzyka*.

PN ISO 31000:2012. (2012). *Zarządzanie ryzykiem – Zasady i wytyczne*.

Pohl K. (1993). *The Three Dimensions of Requirements Engineering*. w: Rolland C., Bodart F. i Cauvet C. (Red.), *Advanced Information Systems Engineering*, wolumen 685 serii *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, strony 275–292.

Pokojski J. (2005). *Systemy doradcze w projektowaniu maszyn*. WNT, Warszawa.

Preece A. (2001). *Evaluating Verification and Validation Methods in Knowledge Engineering*. w: Roy R. (Red.), *Industrial Knowledge Management*. Springer London, strony 91–104.

Przystałka P. (2009). *Metodyka modelowania neuronowego w diagnostyce procesów z uwzględnieniem elementów teorii chaosu*, wolumen 141 serii *Zeszyty*. Politechnika Śląska, Katedra Podstaw Konstrukcji Maszyn, Gliwice.

Psiuk K. (2001). *Obiektowe modele procesu propagacji stanów w układach maszyn*, wolumen 119 serii *Zeszyty*. Politechnika Śląska, Katedra Podstaw Konstrukcji Maszyn, Gliwice.

Raghavan S., Zelesnik G. i Ford G. (1994). *Lecture Notes on Requirements Elicitation*. Raport: CMU/SEI-94-EM-010, Software Engineering Institute, Carnegie Mellon University. dostępne online: `http://repository.cmu.edu/sei/192/`, 5 września 2015.

Raspotnig C. i Opdahl A. (2013). *Comparing risk identification techniques for safety and security requirements*. Journal of Systems and Software, 86(4):1124 – 1151.

Rausand M. i Høyland A. (2004). *System Reliability Theory: Models, Statistical Methods, and Applications*. Wiley Series in Probability and Statistics - Applied Probability and Statistics Section. Wiley, Nowy Jork.

Robertson S. (2001). *Requirements trawling: techniques for discovering requirements*. International Journal of Human-Computer Studies, 55(4):405 – 421.

Rogala T. i Amarowicz M. (2015). *Multimodal Statement Networks for Organic Rankine Cycle Diagnostics – Use Case of Diagnostic Modeling.* w: Angelov P., Atanassov K., Doukovska L., Hadjiski M., Jotsov V., Kacprzyk J., Kasabov N., Sotirov S., Szmidt E. i Zadrożny S. (Red.), *Intelligent Systems'2014*, wolumen 322 serii *Advances in Intelligent Systems and Computing.* Springer International Publishing, strony 605–612.

Rusin A. (2008). *Awaryjność, niezawodność i ryzyko techniczne w energetyce cieplnej.* Wydawnictwo Politechniki Śląskiej, Gliwice.

Russell S. J. i Norvig P. (2009). *Artificial Intelligence: A Modern Approach.* Prentice Hall Press, Nowy Jork, wydanie 3.

Rutkowski L. (2012). *Metody i techniki sztucznej inteligencji.* Wydawnictwo Naukowe PWN, Warszawa, wydanie 2 zmienione – 3 dodruk.

Ryś A. (2015). *Planowane postarzanie produktu – analiza zjawiska.* Zeszyty Naukowe Wyższej Szkoły Humanitas. Zarządzanie, (1):121–128.

Rzydzik S., Amarowicz M., Psiuk K. i Rogala T. (2014). *Condition monitoring of the sealless pump with the use of multi-modal statement network.* w: *13th IMEKO TC10 Workshop on Technical Diagnostics Advanced measurement tools in technical diagnostics for systems reliability and safety, Warszawa, 26–27 czerwca 2014.* IMEKO, strony 122–127.

Saaty T. (1980). *The Analytic Hierarchy Process: Planning, Priority Setting, Resource Allocation.* McGraw-Hill Education, Nowy Jork.

Saiedian H. i Dale R. (2000). *Requirements engineering: making the connection between the software developer and customer.* Information and Software Technology, 42(6):419 – 428.

Siler W. i Buckley J. J. (2005). *Fuzzy Expert Systems and Fuzzy Reasoning.* Wiley, Nowy Jork.

Skelton B. (1997). *Process Safety Analysis: An Introduction.* Institution of Chemical Engineers, Rugby, Warwickshire.

Skupnik D. (2009). *Wieloaspektowe modele diagnostyczne*, wolumen 140 serii *Zeszyty.* Politechnika Śląska, Katedra Podstaw Konstrukcji Maszyn, Gliwice.

Slade G. (2009). *Made to Break: Technology and Obsolescence in America.* Harvard University Press.

Smith D. (1992). *Expert systems for medical diagnosis: A study in technology transfer.* The Journal of Technology Transfer, 17(4):45–53.

Smith S. i Kandel A. (1993). *Verification and Validation of Rule-Based Expert Systems.* Taylor & Francis, Londyn.

Soderstrom T. i Stoice P. (1997). *Identyfikacja systemów.* WNT, Warszawa.

Sommerville I. (2003). *Inżynieria oprogramowania.* WNT, Warszawa.

Sommerville I. i Sawyer P. (1997). *Requirement Engineering: A Good Practice Guide.* Wiley, Nowy Jork.

Stamatis D. (2003). *Failure Mode and Effect Analysis: FMEA from Theory to Execution.* American Society for Quality, Milwaukee.

Taleb N. N. (2007). *Black Swan.* Random House, Nowy Jork.

Tang Y. i Feng K. (2009). *An Expert System Based Approach to Modeling and Selecting Requirement Engineering Techniques.* w: Liu W., Luo X., Wang F. i Lei J. (Red.), *Web Information Systems and Mining*, wolumen 5854 serii *Lecture Notes in Computer Science.* Springer Berlin Heidelberg, strony 19–30.

Tarełko W. (2011). *Metodologia projektowania własności eksploatacyjnych złożonych obiektów technicznych.* Wydawnictwo Naukowe Instytutu Technologii Eksploatacji – PIB, Gdynia.

Timofiejczuk A. (2011). *Metodyka wnioskowania z uwzględnieniem kontekstu w diagnostyce maszyn.* Wydawnictwo Politechniki Śląskiej, Gliwice.

Timofiejczuk A. (2013). *Context based diagnostics of rotating machinery.* w: *International Conference Surveillance 7, Institute of Technology of Chartres, Chartres, Francja, 29–30 października 2013*, strony 1–9.

Tixier J., Dusserre G., Salvi O. i Gaston D. (2002). *Review of 62 risk analysis methodologies of industrial plants.* Journal of Loss Prevention in the Process Industries, 15(4):291 – 303.

Turban E., Aronson J. i Liang T. (2005). *Decision support systems and intelligent systems.* Pearson/Prentice Hall, Upper Saddle River, New Jersey.

Vachtsevanos G., Lewis L. F., Roemer M., Hess A. i Wu B. (2006). *Intelligent Fault Diagnosis and Prognosis for Engineering Systems.* Wiley, Nowy Jork.

Walsh M., Andrew P., Barnsley R., Bertalot L., Boivin R., Bora D., Bouhamou R., Ciattaglia S., Costley A., Counsell G., Direz M., Drevon J., Encheva A., Fang T., von Hellermann M., Johnson D., Kim J., Kusama Y., Lee H., Levesy B., Martin A., Maquet P., Okayama K., Reichle R., Patel K., Pitcher C., Prakash A., Simrock S., Taylor N., Udintsev V., Utin Y., Vasu P., Vayakis G., Veshchev E., Schunke B., Walker C., Watts C. i Zvonkov A. (2011). *ITER diagnostic challenges.* w: *IEEE/NPSS 24th Symposium on Fusion Engineering*, SOFE 2011, strony 1–8.

Wiegers K. i Beatty J. (2014). *Specyfikacja oprogramowania. Inżynieria wymagań.* Helion, Gliwice, wydanie 3.

Wyleżoł M. (2000). *Metoda pozyskiwania procedur i relacji diagnostycznych od specjalistów w dziedzinie eksploatacja maszyn*, wolumen 115 serii *Zeszyty.* Politechnika Śląska, Katedra Podstaw Konstrukcji Maszyn, Gliwice.

Young R. R. (2002). *Recommended Requirements Gathering Practices.* CROSSTALK The Journal of Defense Software Engineering, 15(4):9–12.

Young R. R. (2004). *The Requirements Engineering Handbook.* Artech House, Boston.

Yunusoglu M. G. i Selim H. (2013). *A fuzzy rule based expert system for stock evaluation and portfolio construction: An application to Istanbul Stock Exchange.* Expert Systems with Applications, 40(3):908 – 920. FUZZYSS11: 2nd International Fuzzy Systems Symposium, 17–18 listopada 2011, Ankara.

Zadeh L. (1965). *Fuzzy sets.* Information and Control, 8(3):338–353.

Zowghi D. i Coulin C. (2005). *Requirements Elicitation: A Survey of Techniques, Approaches, and Tools.* w: Aurum A. i Wohlin C. (Red.), *Engineering and Managing Software Requirements.* Springer Berlin Heidelberg, strony 19–46.

Żółtowski B. (1996). *Podstawy diagnostyki maszyn.* Wydawnictwo Uczelniane ATR, Bydgoszcz.

Żółtowski B. i Ćwik Z. (Red.). (1996). *Leksykon diagnostyki technicznej.* Wydawnictwo Uczelniane ATR, Bydgoszcz.

Zwicky F. (1969). *Discovery, Invention, Research Through the Morphological Approach.* The Macmillian Company, Toronto.

Żytkow J. M. i Zembowicz R. (1993). *Database exploration in search of regularities.* Journal of Intelligent Information Systems, 2:39–81.

Spis rysunków

2.1 Intensywność uszkodzeń w okresie użytkowania typowego obiektu technicznego . 7
2.2 Funkcja niezawodności $R(t)$ oraz zawodności $F(t)$ obiektu technicznego 8
2.3 Model obiektu diagnozowania (Korbicz i in., 2002) 9
2.4 Schemat wystąpienia uszkodzenia w trakcie eksploatacji obiektu, T_d – czas diagnozowania, T_n – czas naprawy, T_{ui} – czas trwania i-tego uszkodzenia, T_{zi}, T_{zi+1} – czas poprawnej pracy pomiędzy uszkodzeniami 10
2.5 Przebieg uszkodzenia nagłego (a) oraz naturalnego (b), t_s – czas detekcji uszkodzenia 10
2.6 Idea diagnostyki symptomowej (Korbicz i in., 2002) 12
2.7 Idea diagnostyki wspartej modelem (Korbicz i in., 2002) 13

3.1 Etapy procesu oceny ryzyka wg normy PN-EN ISO 12100:2012 18
3.2 Przykładowa matryca ryzyka. Oznaczenie ryzyka zgodne z tabelą 3.3 20

4.1 Wizja wyniku końcowego realizowanego projektu z punktu widzenia różnych jego uczestników, (opracowanie własne na podstawie `www.projectcartoon.com`) 26
4.2 Elementy inżynierii wymagań 28
4.3 Dwa skrajne zbiory wymagań 29
4.4 Trzy wymiary inżynierii wymagań (Pohl, 1993) 31

5.1 Schemat zależności pomiędzy danymi, informacją oraz wiedzą 36
5.2 Temperatura w węźle łożyskowym wyrażona za pomocą zbiorów rozmytych 39
5.3 Przykład sieci stwierdzeń obrazującej fragment posiadanej wiedzy, związanej z diagnostyką pompy wirnikowej; n_1 – pompa działa poprawnie, n_2 – drgania podczas pracy pompy, n_3 – zbyt mała wydajność pompy, n_4 – silnie zapowietrzone medium, n_5 – zbyt mała prędkość obrotowa, n_6 – zbyt gęste medium 40
5.4 Graf reprezentujący warunek konieczny i dostateczny 41
5.5 Przykład wielomodalnej sieci stwierdzeń 43
5.6 Fragment wiedzy o elektrociepłowni zapisany w postaci sieci semantycznej (Pieczyński, 2003) 44
5.7 Elementy składowe systemu doradczego oraz przepływ informacji pomiędzy nimi 44

6.1 Ilustracja rozpatrywanego problemu 48
6.2 Proces projektowania systemu diagnostycznego 49
6.3 Schemat zależności pomiędzy elementami proponowanego formatu zapisu wymagań 52
6.4 Tablica morfologiczna wraz z zaznaczonymi przykładowymi projektami systemu diagnostycznego 54

6.5 Trójwymiarowa przestrzeń rozwiązań systemu diagnostycznego, wraz z zaznaczony przykładowym rozwiązaniem DS_i 55
6.6 Struktura modelu sieciowego dla wymagania funkcjonalnego, AN_i – węzły dodatkowe 57
6.7 Tablica morfologiczna ze zbiorem możliwych rozwiązań systemu diagnostycznego, v_{fk} – wartość funkcji kryterialnej 58

7.1 Struktura opracowanego *Edytora wymagań* 61
7.2 Edytor wymagań; diagram *EER* przedstawiający strukturę opracowanej bazy danych 62
7.3 Edytor wymagań; okno główne aplikacji 63
7.4 Edytor wymagań; okno aplikacji umożliwiające nawiązanie połączenia z bazą danych 65
7.5 Edytor wymagań; formularz przeznaczony do definiowania katalogu symptomów . 65
7.6 Edytor wymagań; formularz przeznaczony do definiowania wymagania 66
7.7 Edytor wymagań; formularz przeznaczony do definiowania reguły diagnostycznej 66
7.8 Edytor wymagań; formularz prezentujący przykładowe wyniki procesu wnioskowania 67
7.9 Edytor wymagań; okno aplikacji umożliwiające eksport zdefiniowanych reguł diagnostycznych do tabel środowiska LaTeX 67
7.10 Edytor wymagań; zestawienie wybranych formularzy i okien aplikacji 70

8.1 Stanowisko FESTO S7 EduTrainer Compact Siemens S7-300 CPU313C (Skupnik, 2009) 72
8.2 Schemat stanowiska FESTO S7 EduTrainer Compact Siemens S7-300 CPU313C (FESTO); ALS — przełącznik pływakowy poziomu alarmowego; FS — czujnik natężenia przepływu cieczy, FTS — przełącznik pływakowy, H — grzałka, LS- — czujnik zbliżeniowy dolny, LS+ — czujnik zbliżeniowy górny, M — silnik, P — pompa odśrodkowa, PS — czujnik ciśnienia, T1, T2 — odpowiednio zbiornik dolny i górny, T3 — zbiornik ciśnieniowy, TS — czujnik temperatury, US — ultradźwiękowy czujnik poziomu, V2 — zawór kulowy, V6 — zawór proporcjonalny, Vi — zawory ręczne (i = 1, 3, 4, 5, 7, 8, 9, 10; V5 — zawór spustowy) 73
8.3 Uproszczony schemat stanowiska FESTO S7 EduTrainer Compact, obejmujący rozpatrywany przypadek; H — grzałka, M — silnik, P — pompa odśrodkowa, T1, T2 — odpowiednio zbiornik dolny i górny, V2 — zawór kulowy, V5 – zawór spustowy ręczny 73
8.4 Edytor wymagań; formularz przeznaczony do definiowania zbioru uszkodzeń . . . 74
8.5 Edytor wymagań; zdefiniowany katalog sygnalizatorów poziomu cieczy 76
8.6 Edytor wymagań; przyporządkowanie wymagań funkcjonalnych oraz reguł diagnostycznych do podzespołów obiektu technicznego 77
8.7 Edytor wymagań; przykład zdefiniowanego kryterium ograniczającego dla wybranej reguły diagnostycznej 77
8.8 Pakiet *REx*; wizualizacja modelu sieciowego dla wymagania funkcjonalnego, opis węzłów; *r.10* – wymaganie funkcjonalne, *r.1use*, *r.10prob...* – węzły przechowujące informację o wartościach wybranych atrybutów, *use*, *prob*,.... - węzły przechowujące informację o stopniu ważności danego atrybutu, *r.10anuse*, *r.10anprob...* – węzły dodatkowe 80

8.9 Pakiet *REx*; wizualizacja modelu sieciowego dla reguły diagnostycznej, opis węzłów; *f.1* – reguła diagnostyczna, *f.1use*, *r.1lim...* – węzły przechowujące informację o wartościach wybranych atrybutów, *use*, *lim*,.... - węzły przechowujące informację o stopniu ważności danego atrybutu, *f.1anuse*, *f.1anlim...* – węzły dodatkowe 81
8.10 Edytor wymagań; lista wymagań funkcjonalnych oraz przypisanych do nich reguł diagnostycznych wraz z wyznaczonymi w procesie wnioskowania wartościami stopnia preferencji, uszeregowanie wg malejącej wartości stopnia preferencji . . . 81
8.11 Edytor wymagań; lista możliwych podsystemów dla wybranej reguły diagnostycznej 83
8.12 Edytor wymagań; wybór wymagań oraz reguł diagnostycznych dla potrzeb wygenerowania raportu podsumowującego projekt systemu diagnostycznego 84

Spis tabel

2.1 Przykład binarnej macierzy diagnostycznej 11

3.1 Przykład jakościowego opisu prawdopodobieństwa/częstości wystąpienia i-tego zdarzenia 19
3.2 Przykład jakościowego opisu skutków wystąpienia i-tego zdarzenia 19
3.3 Przykład jakościowego opisu ryzyka związanego z wystąpieniem i-tego zdarzenia 20
3.4 Poziom nienaruszalności bezpieczeństwa (wg IEC EN 61508) dla systemów pracujących w trybie na żądanie, PFD – prawdopodobieństwo uszkodzenia na żądanie (ang. *Probability of Failure on Demand*) 24
3.5 Poziom nienaruszalności bezpieczeństwa (wg IEC EN 61508) dla systemów pracujących w trybie ciągłym, PFH – prawdopodobieństwo uszkodzenia na godzinę (ang. *Probability of Failure per Hour*) 24

8.1 Fragment zdefiniowanego zbioru uszkodzeń rozpatrywanego obiektu[1] 75
8.2 Fragment zdefiniowanego zbioru wymagań funkcjonalnych dla rozpatrywanego obiektu technicznego 78
8.3 Fragment zdefiniowanego zbioru reguł diagnostycznych dla rozpatrywanego obiektu technicznego 79
8.4 Wpływ wartości wybranych atrybutów na wartość stopnia preferencji pf dla wymagania o treści: *Wykrywanie drgań w układzie pompy P*, wejściowe wartości atrybutów: Poziom redukcji prawdopodobieństwa: bardzo_duży, Poziom redukcji kosztów: duzy, Poziom redukcji czasu przestoju: duzy, Stosowalność wymagania: powszechnie, Czy spełnia ograniczenia projektowe: tak, w procesie wnioskowania uwzględniane są wszystkie atrybuty (stopień ważności = tak) 82

Analiza ryzyka w inżynierii wymagań dla systemów diagnostycznych

Streszczenie

Autor: mgr inż. Marcin Amarowicz
Promotor: prof. dr hab. inż. Wojciech Cholewa
Politechnika Śląska, Wydział Mechaniczny Technologiczny

Niniejsza rozprawa doktorska związana jest z problematyką opracowywania projektów systemów diagnostycznych dla obiektów technicznych. W ogólnym przypadku, zagadnienie to jest trudne w realizacji z uwagi na konieczność uwzględniania w procesie projektowym wielu różnych czynników, w tym m.in. wyników analizy ryzyka czy też istniejących ograniczeń. Celem podjętych badań, było opracowanie takiego podejścia projektowego, które umożliwiałoby uwzględnianie wymienionych czynników w procesie projektowym.

Przyjęto że zadanie to, może zostać osiągnięte poprzez wprowadzenie, do standardowego procesu projektowego dodatkowego etapu, w czasie którego dokonany zostanie sformalizowany zapis potrzeby, opisującej konieczność opracowania systemu diagnostycznego. Zapis ten zawiera informacje o oczekiwanych funkcjonalnościach systemu diagnostycznego jak również zbiór potencjalnych podsystemów je realizujących. Na podstawie tak zapisanej potrzeby, możliwie jest wygenerowanie zbioru potencjalnych rozwiązań systemu diagnostycznego, z którego w dalszym etapie wyodrębniane jest rozwiązanie optymalne.

Dla potrzeb zapisu potrzeby w sformalizowany sposób, zaproponowano użycie powszechnie stosowanych w obszarze inżynierii oprogramowania zbiorów wymagań. Zaproponowano specjalny sposób zapisu wymagań, uwzględniający specyfikę rozpatrywanego problemu. Proces optymalizacji postaci projektu systemu diagnostycznego, został zrealizowany za pośrednictwem opracowanego systemu doradczego, jako proces wnioskowania w stosownych modelach sieciowych reprezentujących zgromadzoną wiedzę projektową.

Dla potrzeb realizacji zaproponowanego podejścia projektowego, opracowana została aplikacja, tzw. *Edytor wymagań*, za pośrednictwem której możliwe jest przeprowadzenie większości etapów proponowanego podejścia projektowego.

Praktyczna weryfikacja zaproponowanego podejścia projektowego, obejmowała opracowanie fragmentu projektu systemu diagnostycznego dla typowego obiektu technicznego. Obiektem badań było stanowisko badawcze FESTO S7 EduTrainer Compact, za pośrednictwem którego możliwa jest realizacja typowych procesów przemysłowych. Dla tego obiektu technicznego, pozyskano niezbędne zbiory wymagań funkcjonalnych oraz reguł diagnostycznych, które następnie zostały ocenione przez pryzmat ich przydatności do osiągnięcia przez projektowany system diagnostyczny założonej funkcji celu. Następnie wygenerowany został zbiór potencjalnych rozwiązań projektu systemu diagnostyczne oraz wybrane rozwiązanie optymalne. Analiza zebranych wyników, pozwoliła na stwierdzenie słuszności postawionych tez pracy.

Słowa kluczowe: systemy diagnostyczne, ryzyko techniczne, system doradczy, wymagania
Dziedziny nauki: budowa i eksploatacja maszyn, diagnostyka techniczna, sztuczna inteligencja, inżynieria wymagań

Risk analysis in requirements engineering for the diagnostic systems

Summary

Author: Marcin Amarowicz MSc, Eng.
Supervisor: prof. Wojciech Cholewa, PhD, DSc Eng.
Silesian University of Technology, Faculty of Mechanical Engineering

The presented PhD thesis deals with the problems of designing of the project of diagnostic systems for technical objects. In general case, this issue is difficult to implement, due to the need to take into account in the design process of many different factors, such as the results of risk analysis or existing limitations. The main goal of the conducted research, was the development of such design approach, that allows for consideration of these factors in the design process.

It was assumed, that this task can be achieved by introducing to the standard design approach one additional phase, during which the formal notation of the project needs will be performed. This notation contains an information about expected functionalities of the diagnostic system as well as the set of potential subsystems that implemented them. Based on such written project need, it is possible to generate a set of potential solutions of the project of the diagnostic system. In the further stage, an optimal solution will be selected from them.

For the purpose of recording the project need in the formalized way, the sets of requirements, that are commonly used in software engineering area, was used. The special notation of requirements, that takes into account all aspects of considered problem was proposed. The optimization process of the project of diagnostic system, has been carried out through a developed expert system. The optimization process was based on the gathered project knowledge and has been implemented as an inference process in the appropriate graphical models.

For the purposes of carrying out the proposed project approach, the specialized application, so called *Requirements editor*, was implemented. Through this application it is possible to carry out most of the steps of proposed design approach.

Practical verification of the proposed design approach, covered the implementation of a part of diagnostic system project for a typical technical object.The research stand FESTO S7 Edu-Trainer Compact, through which it is possible to implement common industrial processes, was considered as this object. For this technical object, the necessary sets of functional requirements and diagnostic rules was gathered. Then, they were evaluated through the prism of their suitability for achieving by the designed diagnostic system the assumed goals. Then it was generated a set of potential solutions of the diagnostic system and the optimal solution was selected. The analysis of the results allowed to confirm the rightness of the assumed theses.

Keywords:: diagnostic systems, technical risk, expert system, requirements
Science domains: design and maintenance of machinery, technical diagnostics, artificial intelligence, requirements engineering