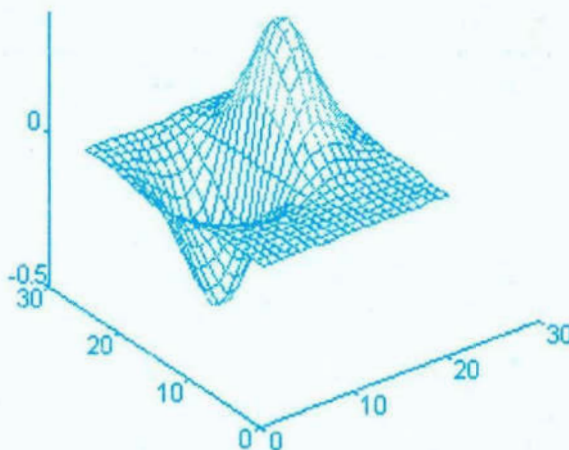
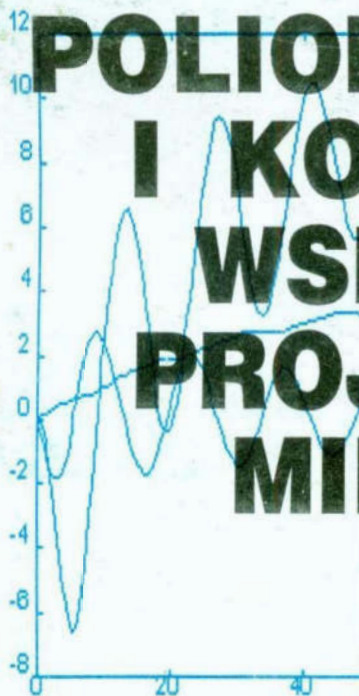


WYŻSZA SZKOŁA INŻYNIERSKA W KOSZALINIE
KATEDRA PROJEKTOWANIA SYSTEMÓW TECHNICZNYCH

Materiały XII Ogólnopolskiej Konferencji

**POLIOPTYMALIZACJA
I KOMPUTEROWE
WSPOMAGANIE
PROJEKTOWANIA
MIELNO 1994**

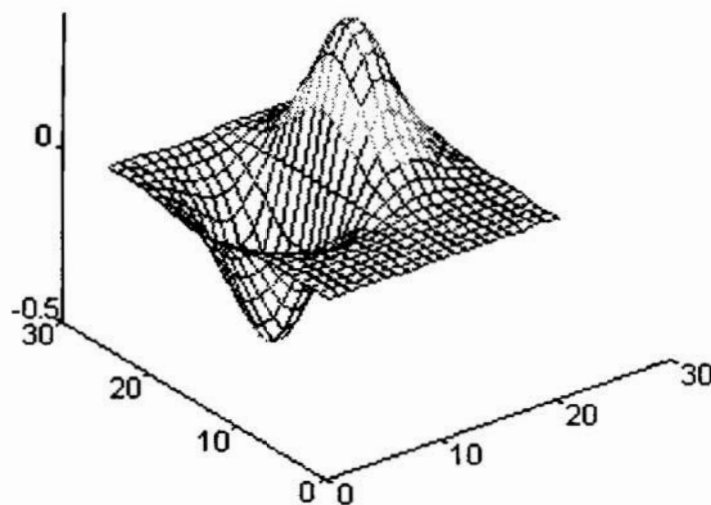


KOSZALIN - MIELNO

Wyższa Szkoła Inżynierska w Koszalinie
Katedra Projektowania Systemów Technicznych

Materiały XII Ogólnopolskiej Konferencji

Poliptymalizacja i Komputerowe Wspomaganie Projektowania Mielno '94



Koszalin – Mielno

Zeszyt Naukowy Wydziału Mechanicznego nr **18** a

ISSN 0860-0325

Przewodniczący Uczelnianej Rady Wydawniczej

Szymon PAŁKOWSKI

Redakcja:

Wojciech TARNOWSKI
Tomasz KICZKOWIAK

Wydano za zgodą Rektora WSInż. w Koszalinie

Materiały recenzowane

Wydanie materiałów dofinansowane przez KBN

Wydawnictwo Wyższej Szkoły Inżynierskiej w Koszalinie
75-620 KOSZALIN, ul. Raławicka 15-17, tel. 427-881

1995, wyd. 1, ark. wyd. 9,69, nakład 150 egz., format B5
Druk : *Pracownia Poligrafii WSInż. zam. O/3/95 r.*

Spis treści

1. Arendt R.: Modelowanie właściwości okrętowych zespołów napędu głównego statku z zastosowaniem programu Simulink ...7
2. Balicki J.: Neuronowa polioptymalizacja przydziałów modułów programowych w rozproszonych systemach komputerowych ...19
3. Barczyk J.: Wspomagane komputerowo projektowanie układów przeniesienia napędu w chwytakach robotów przemysłowych ...29
4. Bielecki Z., Wróbel J.: Organizacja baz danych w procesie komputerowo wspomaganego projektowania samojezdnych żurawi hydraulicznych ...37
5. Capanidis D.: Zastosowanie metod optymalizacji w określaniu składu polimerowych kompozytów o żądanych własnościach użytkowych ...45
6. Chanas S., Kobyłański P.: Metoda permutacji wielokryterialnej oceny decyzji - heurystyczny algorytm rozwiązania ...53
7. Chanas S., Kuchta D.: Algorytm rozwiązywania dwukryterialnych zadań programowania liniowego z parametrem we współczynnikach funkcji celu ...59
8. Cichocki K.: Komputerowa symulacja wybuchu podwodnego program ABAQUS Explicitx ...65
9. Dziama A., Wieleba W.: Wspomagane komputerowo obliczenia przekładni pasowej z pasem płaskim ...73
10. Grymek Sz., Tarnowski W.: Sztuczne sieci neuronowe rekurencyjne do modelowania procesów dynamicznych ...81
11. Kwiesielewicz M.: Grupowe wielokryterialne podejmowanie decyzji w oparciu o metodę porównywania parami ...91
12. Laudański L. M.: Imitacyjne modele pól losowych aeronautyki ...99
13. Muszyński R.: Kryterium uogólnione optymalizacji statycznej silnika przekształtnikowego ...107

14. Ostwald M.: Optymalizacja wektorowa trójwarstwowych powłok walcowych ...115
15. Paczkowski W., Karłowski J., Oldenburg S.: Rozproszona optymalizacja konstrukcji hali sportowej ...123
16. Pułka A.: Wykorzystywanie wnioskowania przy niepełnej informacji do projektowania układów cyfrowych ...135
17. Siwek B. J.: Komputerowe wspomaganie projektowania struktura danych ...143
18. Siwek J. B.: Pojęcie modelu komputerowego ...151
19. Stryczek R.: Wykorzystanie sieci Petri dla aktywnej optymalizacji przebiegu operacji tokarskich ...157
20. Tarczyński L.: Komputerowe modelowanie obciążenia temperaturą obrotowych konstrukcji powłokowych ...165
21. Wilczyński B.: Algorytm i program komputerowy optymalizacji kształtu karbów w dwuwymiarowych elementach konstrukcji ...173

Contents

1. Arendt R.: The modelling of marine propulsion unit control system property with an application of program Simulink ...7
2. Balicki J.: Neural multicriteria optimization of program modules allocations in distributed computer system ...19
3. Barczyk J.: Computer Aided design of propulsion transmission of robot's grippers ...29
4. Bielecki Z., Wróbel J.: Hierarchical - network data bases for computer - aided design of hydraulic mobile cranes ...37
5. Capanidis D.: Application of optimization methods for the determination of the composites contents to meet the required specifications ...45
6. Chanas S., Kobylański P.: The permutation method of a multicriteria decision evaluation - a heuristic algorithm ...53
7. Chanas S., Kuchta D.: An algorithm for solving bicriteria linear programming problems with a parameter in the objective functions ...59
8. Cichocki K.: Computer simulation of an underwater explosion - ABAQUS Explicit code ...65
9. Dziama A., Wieleba W.: Computer aided computing of flat belt transmissions ...73
10. Grymek Sz., Tarnowski W.: Dynamic processes modelling by means of recurrent neural networks ...81
11. Kwiesielewicz M.: Multicriteria group decision-making using pairwise comparison method ...91
12. Laudański L. M.: Digital simulation of random fields for aeronautical purposes ...99
13. Muszyński R.: The generalized criterion for steady state operation of self commutated synchronous motor ...107

14. Ostwald M.: Vector optimization of sandwich cylindrical shells ...115
15. Paczkowski W., Karłowski J., Oldenburg S.: Disperse optimization of a sports hall construction ...123
16. Pułka A.: Nonmonotonic reasoning used in the digital circuits desining ...135
17. Siwek B. J.: Data structure for computer assisted design ...143
18. Siwek J. B.: Definition of a computer model ...151
19. Stryczek R.: Petri nets in the modelling and control of automated lathe treatment ...157
20. Tarczyński L.: Computer-aided simulation of the temperature load of the circular shell structures ...165
21. Wilczyński B.: Algorithm and computer program for shape optimization of notches in 2-D structural elements ...173

MODELOWANIE WŁAŚCIWOŚCI OKRĘTOWYCH ZESPOŁÓW NAPĘDU GŁÓWNEGO STATKU Z ZASTOSOWANIEM PROGRAMU SIMULINK

1. Wstęp

W stadium projektowania i budowy statku prowadzi się zwykle badania symulacyjne zespołu napędu głównego, w celu sprawdzenia stabilności pracy w całym zakresie zmian obciążenia tj. przy różnych stanach pracy zespołu. Analizuje się i ocenia właściwości dynamiczne zespołu dla stanów przejściowych, dobiera parametry nastawcze regulatorów prędkości obrotowej oraz czasy załączania sprzęgieł zapewniające minimalny uchyb dynamiczny i czas trwania przebiegu przejściowego regulacji prędkości obrotowej dla przewidywanych stanów przejściowych.

Właściwości zespołu napędu głównego statku związane są z zastosowanymi elementami składowymi oraz wybraną strukturą [1]. Do podstawowych elementów składowych zespołów można zaliczyć: wysokoprężny okrętowy silnik spalinowy, regulator prędkości obrotowej silnika, przekładnię, sprzęgło cierne i podatne, wał śrubowy, śrubę o skoku stałym lub nastawnym. Struktury zespołów napędu głównego związane są z: liczbą i typem użytych silników, obecnością lub brakiem sprzęgła, zastosowaniem śruby o skoku stałym lub nastawnym, wspólną pracą silników na przekładnię itd.

Modele zespołu napędu głównego statku powinny odzwierciedlać właściwości zarówno zastosowanych elementów składowych, jak i przyjętej struktury powiązań. Dogodne jest rozpatrywanie wielopoziomowej struktury modelu zespołu napędowego, gdzie pierwszy poziom szczegółowości opisu matematycznego obejmuje stosowane elementy składowe. Drugi poziom tworzą zwinięte modele elementów składowych - "makrobloki", które odzwierciedlają strukturę zespołu napędowego.

Opracowane modele elementów składowych z reguły posiadają niepewne parametry, których większość zostaje określona przy "podstrajaniu" modeli składowych w procesie weryfikacji. Wybrane zbiory parametrów np. regulatora należy zwykle określić w procesie syntezy całego zespołu napędowego.

Proponowana metodyka badań symulacyjnych wymaga tworzenia modeli złożonych z "makrobloków", jak również ingerencji operatora w pierwszy poziom modelu, co ma miejsce przy zmianie wybranych parametrów elementów składowych zespołu i badaniu ich wpływu na właściwości całego zespołu. Przy rozważaniu pracy równoległej silników lub bardziej złożonych struktur zespołów napędowych wygodne jest stosowanie modeli wyższych poziomów.

Referat stanowi próbę oceny zastosowania programu SIMULINK pracującego w środowisku programowym MATLAB'a [2,3] do symulacji złożonych nieliniowych modeli o wielopoziomowej strukturze. Zaprezentowano opracowane w programie SIMULINK modele elementów: wysokoprężnego silnika spalinowego jako obiektu regulacji prędkości obrotowej, regulatora prędkości obrotowej, sprzęgła cierne i

* *Katedra Automatyki Wydziału Elektrycznego Politechniki Gdańskiej, Gdańsk ul. Narutowicza 11/12*

podatnego, śruby o skoku stałym i nastawnym, których opisy matematyczne zaczerpnięto z [4]. Przedstawiono wyniki badań symulacyjnych zespołów napędu głównego statku w konfiguracji 1: silnik spalinowy pracujący przez przekładnię, wał śrubowy na śrubę o skoku stałym oraz w konfiguracji 2: silnik spalinowy, pracujący przez sprzęgło, przekładnię, wał śrubowy na śrubę o skoku stałym. Oceniono użyteczność programu SIMULINK przy prowadzeniu badań symulacyjnych złożonych układów regulacji i sterowania.

2. Modele elementów składowych zespołu napędu głównego statku

Rozważane poniżej modele matematyczne: wysokoprężnego silnika spalinowego, regulatora prędkości obrotowej silnika, sprzęgła ciernego i podatnego, śruby o skoku stałym i nastawnym opracowane zostały w zasadzie do badań symulacyjnych dynamiki układów regulacji prędkości obrotowej [4]. Struktura poszczególnych modeli związana jest ze zjawiskami fizycznymi zachodzącymi w różnych stanach pracy, przy przyjęciu określonych założeń upraszczających. Parametry modeli wyznaczone zostały na podstawie danych konstrukcyjnych i wyników badań doświadczalnych obiektów rzeczywistych jednostki B-400 [4]. Występujące zmienne w modelach nie posiadają miana. Stosowane są wartości względne - stosunek wartości modelowanej do wartości

znamionowej.

Model silnika spalinowego

Rozważany jest model matematyczny silnika spalinowego jako obiektu sterowania wielkością wejściową h_r - położenie listwy pomp paliwowych, wytwarzający wielkości wyjściowe: ω - prędkość obrotowa wału silnika, m_e - moment efektywny, przy wielkości zakłócającej, m_o - moment obciążenia.

W silniku spalinowym można wyróżnić następujące podzespoły składowe: paliwowy, spalania, turbodoładowania i mas wirujących.

Na podstawie opisu matematycznego zredagowano model silnika w programie SIMULINK (rys. 1). Przy tworzeniu modeli przyjęto założenie, że wykorzystywane będą jedynie bloki standardowe programu. Zrezygnowano z istniejącej możliwości tworzenia własnych zdefiniowanych bloków (podprogramy z rozszerzeniem .M), ponieważ nie dają one dostępu do

Podzespół spalania i mas wirujących

$$T \frac{d\omega}{dt} = m_e - m_o \quad (1)$$

$$T_c \frac{dm_i}{dt} + k_c m_i = \begin{cases} h & \text{dla } h \leq a + bp \\ a + bp & \text{dla } h > a + bp \end{cases} \quad (2)$$

$$m_e = m_i - m_t \quad (3)$$

$$m_t = k_t \omega \quad (4)$$

gdzie: T , T_c , k_c , a , b , k_t - współczynniki modelu.

Podzespół turbodoładowania

$$T_t T_k \frac{d^2 p}{dt^2} + (T_t + T_k) \frac{dp}{dt} + p = f(P) \quad (5)$$

$$f(P) = m_e^n \omega^m \quad (6)$$

gdzie: T_t , T_k , n , m - współczynniki modelu.

Podzespół paliwowy

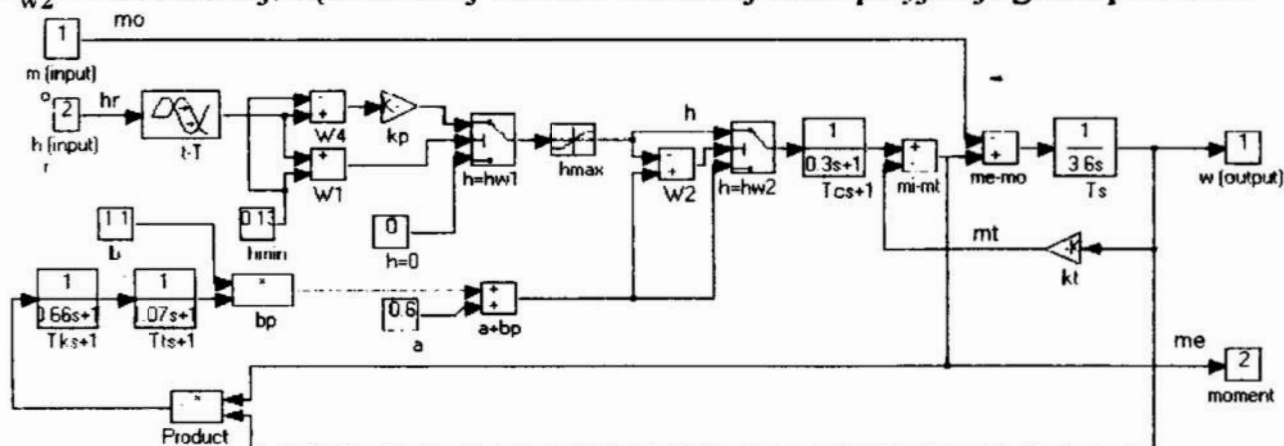
$$h_r (t - \tau) = h' \quad (7)$$

$$h = \begin{cases} 0 & \text{dla } h' < h_{\min} \\ k_p (h' - h_{\min}) & \text{dla } h_{\min} \leq h' \leq h_{\max} \\ h_{\max} & \text{dla } h' > h_{\max} \end{cases} \quad (8)$$

gdzie: k_p , h_{\min} , h_{\max} , τ - współczynniki modelu.

wszystkich używanych zmiennych, a uzyskana struktura modelu była by mało czytelna.

Zależność (1) zamodelowano w postaci członu całkującego T_s , na wejściu którego występuje sumator $m_c - m_o$. Równanie (3) obrazuje sumator $m_i - m_e$. Utworzony obwód zamyka pętlę sprzężenia zwrotnego o wzmocnieniu k_f - zależność (4). Podzespół spalania (2) zamodelowany został w postaci członu inercyjnego $1/(T_c s + 1)$ oraz klucza analogowego, na wejściach którego występują wielkości h i $a + bp$. Podzespół turbodoładowania modelują dwa człony inercyjne $1/(T_t s + 1)$ i $1/(T_k s + 1)$ połączone szeregowo, na wyjściu których występuje zmienna p . Bloki iloczynu i sumatora modelują zależność algebraiczną $a+bp$. Zmienna $a + bp - h$ steruje położenie klucza h_{w2} . Dla dodatniej, bądź zerowej wartości zmiennej klucz przyjmuje górne położenie.



Rys. 1. Struktura modelu silnika zredagowana w programie SIMULINK
 Fig. 1. The structure of the Diesel engine model used in program SIMULINK

Funkcja $f(P)$ (6) zamodelowana została w postaci bloku iloczynu realizującego operację mnożenia $m_e \omega$. Przy modelowaniu potęg o dowolnym wykładniku z zastosowaniem funkcji logarymicznej wystąpiły problemy, ponieważ przy starcie modelu wartości zmiennych posiadają wartości zerowe. Zależność (7) bloku paliwowego modeluje opóźnienie transportowe ($t - \tau$). Funkcję złożoną z odcinków (8) zamodelowano na kluczu h_{w1} i bloku z nasyceniem h_{max} . Na wejściach klucza występują zmienna $k_p(h' - h_{min})$ oraz stała 0. Klucz jest sterowany zmienną $h' - h_{min}$. Dla ujemnej wartości zmiennej klucz przyjmuje dolne położenie. Symbole zmiennych i większość przyjętych wartości współczynników modelu są widoczne na schemacie strukturalnym (rys. 1).

W modelu wyróżniono dwa wejścia: we1) m_o - moment obciążenia, we2) h_r - położenie listwy pomp paliwowych oraz dwa wyjścia: wy1) ω - prędkość obrotowa wału silnika i wy2) m_e - moment efektywny. Model został zwinięty do postaci makrobloku o nazwie "silnik", w którym obowiązuje przyjęta kolejność oznaczeń wejść i wyjść.

Model regulatora prędkości obrotowej

W najczęściej stosowanych regulatorach prędkości obrotowej okrętowych silników spalinowych można wyróżnić następujące podzespoły: mechaniczny miernik prędkości obrotowej, wzmacniacz hydrauliczny oraz człon sztywnego lub podatnego sprzężenia zwrotnego.

Podzespół pomiaru prędkości obrotowej

$$\frac{d^2 \omega_{rs}}{dt^2} + 2\mu \omega_f \frac{d\omega_{rs}}{dt} + \omega_f^2 \omega_{rs} = 2\mu \omega_f \frac{d\omega_r}{dt} + \omega_f^2 \omega_r \quad (9)$$

gdzie: μ, ω_f - współczynniki modelu

wzmacniacz hydrauliczny oraz człon sztywnego lub podatnego sprzężenia zwrotnego.

W regulatorach wielko-

ściami wejściowymi są: ω_r - mierzona wartość prędkości obrotowej i ω_z - zadana wartość prędkości obrotowej. Wielkością wyjściową jest h_r - przesunięcie listwy paliwowej.

Na podstawie opisu matematycznego regulatora prędkości obrotowej silnika zredagowano model w programie SIMULINK (rys. 2). Zależność (9) zamodelowana została w postaci transmitancji Tr Fcn. a zależność (10) obrazuje

$$k''(\omega_z - \omega_{rs} - h_{sz}) = h_c \quad (10)$$

gdzie: k'' - współczynnik modelu.

Podzespół wzmacniacza hydraulicznego

$$h_{si} = \begin{cases} 0 & \text{dla } h_s > 0 \cap h_r \geq h_{\max} \cup h_s < 0 \cap h_r \leq 0 \\ h_s & \text{w pozostałych przypadkach} \end{cases} \quad (11)$$

$$h_{si} = \begin{cases} T_{s1} \frac{dh_{r1}}{dt} = h_{si} & \text{dla } h_{si} < 0 \\ T_{s2} \frac{dh_{r2}}{dt} = h_{si} & \text{dla } h_{si} \geq 0 \end{cases} \quad (12)$$

gdzie: T_{s1} , T_{s2} , h_{\max} - współczynniki modelu.

Podzespół sprzężeń zwrotnych

$$\delta_r h_r = h_{sz} \quad (13)$$

$$h_s = h_c - h_i \quad (14)$$

$$T_i \frac{dh_i}{dt} + h_i = \delta_i T_i \frac{dh_r}{dt} \quad (15)$$

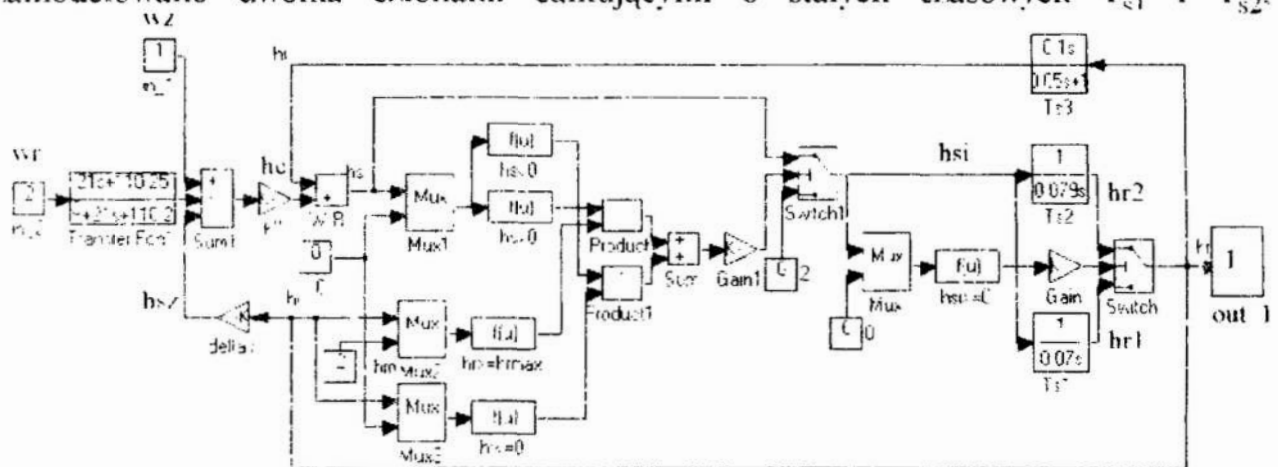
gdzie: T_i , T_r , δ_r , δ_i - współczynniki modelu.

została w postaci transmitancji Tr Fcn. a zależność (10) obrazuje trójwejściowy sumator Sum1. Dwuodcinkową funkcję (11) modeluje klucz Switch1. Spełnienie wymaganych relacji sterującej położenie klucza modelują człony Gain1, Sum, Product, Product1, $h_s < 0$, $h_s > 0$, $h_s \geq h_{\max}$, $h_r \leq 0$, Mux1, Mux2, Mux3.

Przyporządkowując wysoki stan sygnału logicznego HL wartości arytmetycznej większej od zera, a niski stan sygnału logicznego LL wartości arytmetycznej równej zero uzyskuje się umowne wartości logiczne, które mogą służyć doysterowania kluczy. Podstawowe działania

algebry Boole'a można więc modelować członami algebraicznymi sumy i iloczynu oraz blokami funkcyjnymi zgodnie z przyjętym przyporządkowaniem wartości algebraicznych stanom logicznym.

Sygnal wyjściowy członu proporcjonalnego o ujemnym wzmocnieniu Gain1ysterowuje klucz. Dla HL sygnał sterujący ma wartość mniejszą od zera - dolne położenie klucza, a dla LL wartość zero - górne położenie. Zależność (12) zamodelowano dwoma członami całkującymi o stałych czasowych T_{s1} i T_{s2} .



Rys. 2. Struktura modelu regulatora zredagowana w programie SIMULINK

Fig. 2. The structure of the Diesel engine governor model used in program SIMULINK

których wyjścia są przełączane kluczem Switch. Sterowanie klucza odpowiadające spełnieniu relacji przedstawiają bloki Mux. $h_{si} \geq 0$ oraz Gain. Równanie (13) modeluje człon proporcjonalny delta r. a zależność (15) człon o transmitancji T_{s3} .

Model został zwinięty do postaci makrobloku i posiada dwa wejścia: we1) ω_z - zadana wartość prędkości obrotowej, we2) ω_r - mierzona wartość prędkości obrotowej oraz jedno wyjście wy1) h_r - przesunięcie listwy paliwowej silnika.

Model sprzęgła ciernego i podatnego

W sprzęgle ciernym dwustopniowym sterowanym hydraulicznie, wielkość wyjściowa - moment obrotowy przenoszony przez sprzęgło (m_c), uzależniona jest od wielkości wejściowych: ciśnienia oleju sterującego, różnicy prędkości obrotowej obu połówek sprzęgła ($\omega - \omega_Q$) oraz momentu przenoszonego przez sprzęgło podatne (m_h).

Po przyjęciu skokowych zmian ciśnienia oleju sterującego można rozważać sterowanie sprzęgła sygnałem binarnym.

Podzespół sprzęgła ciernego

$$m_c = \begin{cases} 0 & \text{dla } t < t_{os} \\ m_T & \text{dla } t_{os} \leq t < t_{os} + t_s \text{ i } \omega_Q < \omega \\ m_h & \text{dla } t_{os} + t_s \leq t \text{ lub } \omega_Q = \omega \\ m_{cmax} & \text{dla } t_{os} + t_s \leq t \text{ i } m_h \geq m_{cmax} \\ 0 & \text{dla } t_{wyl} \geq t \end{cases} \quad (16)$$

gdzie: $m_T, m_{cmax}, t_{os}, t_s, t_{wyl}$ - współczynniki modelu.

Podzespół sprzęgła podatnego

$$m_h = c\varepsilon + d\varepsilon^2 + \xi(\omega - \omega_Q) \quad (17)$$

$$\varepsilon = \begin{cases} 0 & \text{dla } t < t_{os} \\ \varepsilon_T & \text{dla } t_{os} \leq t \leq t_{os} + t_s \\ \varepsilon_T + T_\varepsilon \int_{t_{os}+t_s}^t (\omega - \omega_Q) dt & \text{dla } t > t_{os} + t_s \end{cases} \quad (18)$$

gdzie: $\varepsilon_T, c, d, \xi, T_\varepsilon$ - współczynniki modelu.

Na podstawie opisu matematycznego sprzęgła ciernego i podatnego zredagowano model w programie SIMULINK (rys. 3). Wprowadzono dwie zmienne o charakterze logicznym we1 - załączanie sprzęgła, we2 - wyłączanie sprzęgła. Wejścia 1 i 2 pobudzane są z bloków generujących skok jednostkowy.

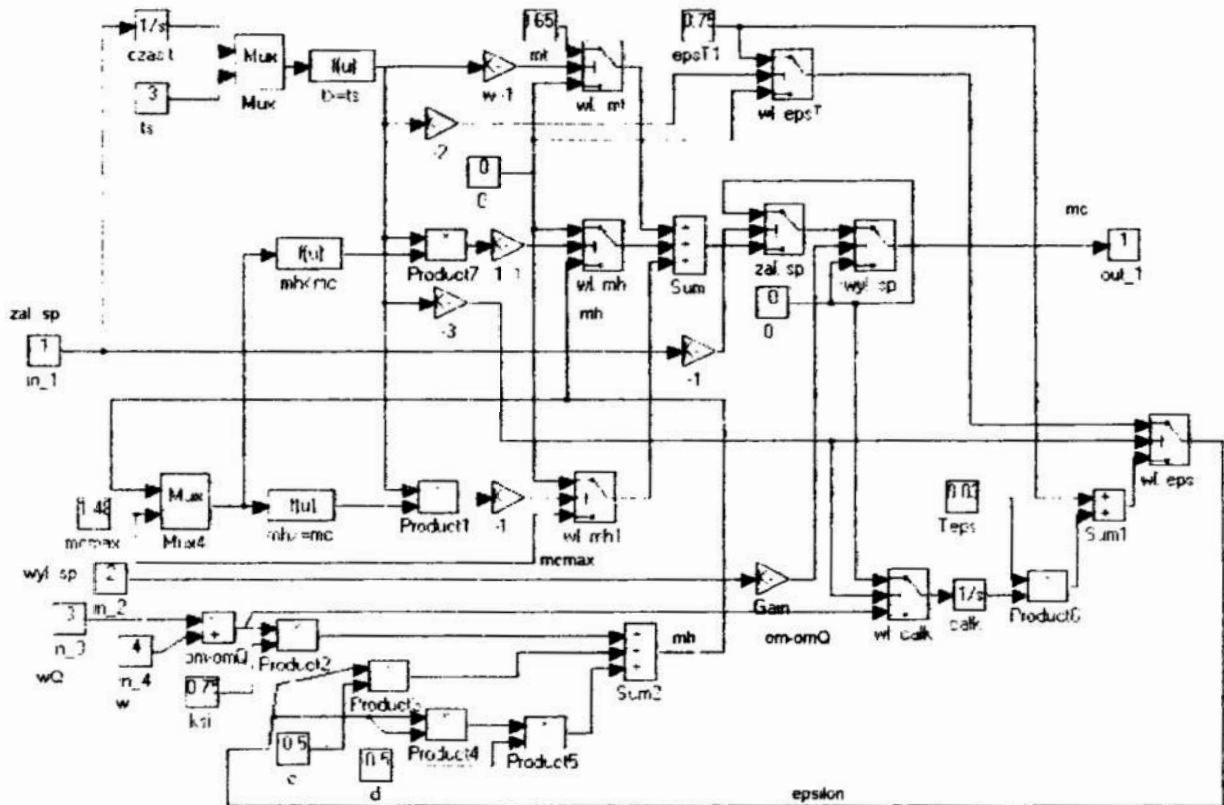
Do określenia zbiorów relacji określających odcinki funkcji (16), przyjęto pięć zmiennych o charakterze logicznym i dla każdej z nich przypisano wartość 1 przy spełnionej relacji \geq , a wartość 0 w przeciwnym przypadku.

Zestawione wartości zmiennych logicznych umożliwiają określenie funkcji boolowskich wykorzystywanych do modelowania sygnałów sterowania kluczami przełączającymi odcinki funkcji (16). Do utworzenia czterech odcinków funkcji zastosowano trzy klucze wł.mt, wł.mh, wł.mhl, które w dolnym położeniu przenoszą wartość zerową. Przyjęto takie sterowanie kluczy, że w czasie symulacji w górnym położeniu może znajdować się co najwyżej jeden klucz. Sygnały wyjściowe trzech kluczy są sumowane przez człon sumacyjny Sum, na wyjściu którego modelowana jest zmienna m_c . Układy logiczne sterowania kluczami są widoczne na rysunku 3, z lewej strony kluczy. Do pomiaru upływu czasu mierzonego od chwili załączenia sprzęgła zastosowano człon kalkujący czas t.

Zależność (17) została zamodelowana na dwóch blokach sumy om-omQ, sum2, czterech blokach iloczynu Product2 - 5 oraz trzech blokach stałej wartości: ksi, c i d.

Dwa pierwsze odcinki funkcji ϵ (18) przełącza klucz wl.epsT, a następny odcinek klucz wl.eps. Do sterowania kluczy wykorzystano zamodelowane kombinacje relacji występujące w (16). W trzecim odcinku funkcji (18), do obliczania wartości całki w granicach od $t_{os} + t_s$ do t załącza się kluczem wartość $(\omega - \omega_q)$ na człon całkujący całk. w chwili $t_{os} + t_s$.

Model został zwinięty do postaci makrobloku o nazwie sprzęgło i posiada cztery wejścia we1) zal.sprz. - sygnał binarny załączania sprzęgła, we2) wyl.sprz. - sygnał binarny wyłączania sprzęgła, we3) omegaQ - prędkość obrotowa wału śrubowego, we4) omega - prędkość obrotowa silnika spalinowego oraz jedno wyjście wy1) m_c - moment obrotowy przenoszony przez sprzęgło.



Rys. 3. Struktura modelu sprzęgła zredagowana w modelu SIMULINK
Fig. 3. The structure of the coupling model used in program SIMULINK

Model śruby o stałym i zmiennym skoku

Dla śrub o stałym skoku ($h_p = \text{const}$) zależność momentu oporowego śruby od prędkości obrotowej, dla różnych wartości prędkości statku $v = \text{const}$ jest przedstawiana

Opis matematyczny śruby o skoku nastawnym

$$m_Q = (1 - \beta)[\alpha \omega_Q^2 h_p^2 + (1 - \alpha)v^2 h_p] + \beta \omega_Q^2 \quad (19)$$

$$T_Q \frac{d\omega_Q}{dt} = m_c - m_Q \quad (20)$$

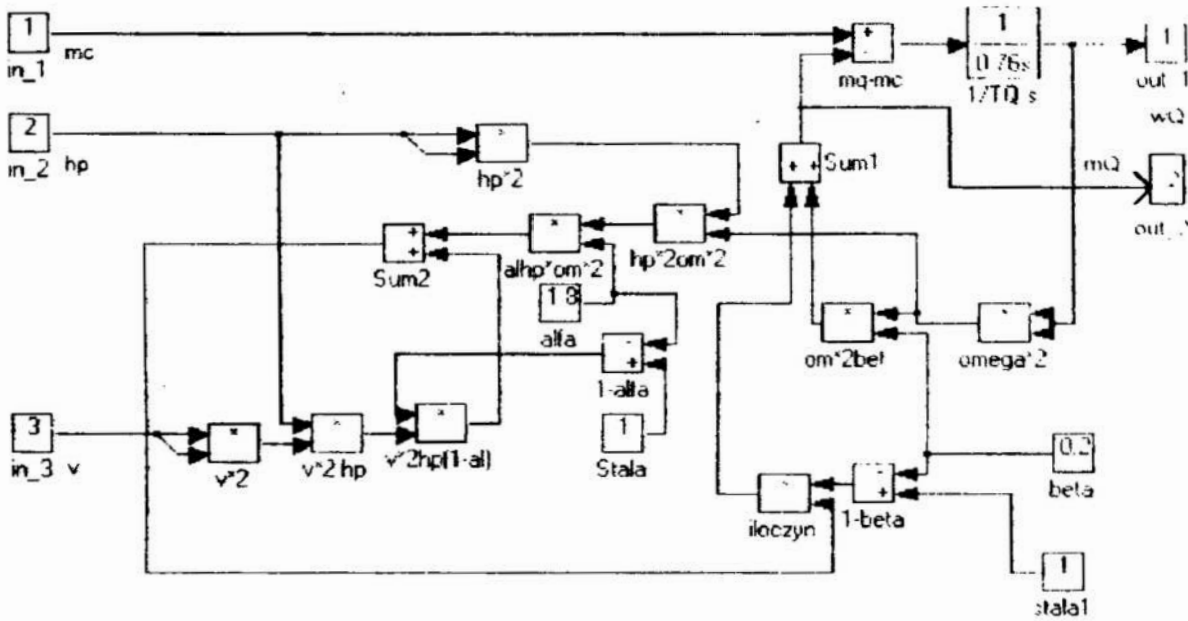
gdzie: α, β, T_q - współczynniki modelu.

graficznie tzn. krzywymi Robinsona, które przy przyjęciu względnych wartości zmiennych dość dobrze przybliża zależność (19) [7]. Dla śruby o skoku nastawnym moment oporowy śruby m_q jest funkcją trzech zmiennych: skoku śruby (h_p).

średkości obrotowej śruby (ω_q) i prędkości statku (v).

Na podstawie opisu matematycznego śruby o zmiennym skoku zredagowano model w programie SIMULINK (rys. 4). Równanie algebraiczne (19) zamodelowane zostało a dziewięciu blokach iloczynu, trzech sumatorach i czterech blokach wartości stałych zależność (20) zamodelowano w postaci członu całkującego $1/T_q s$ i sumatora $m_q - m_c$.

Model został zwinięty do postaci makrobloku o nazwie śruba i posiada trzy wejścia: we1) m_c - moment przenoszony przez sprzęgło na śrubę, we2) h_p - skok śruby, we3) v - prędkość liniowa statku oraz dwa wyjścia: wy1) ω_q - prędkość obrotowa śruby, wy2) m_q - moment oporowy śruby.

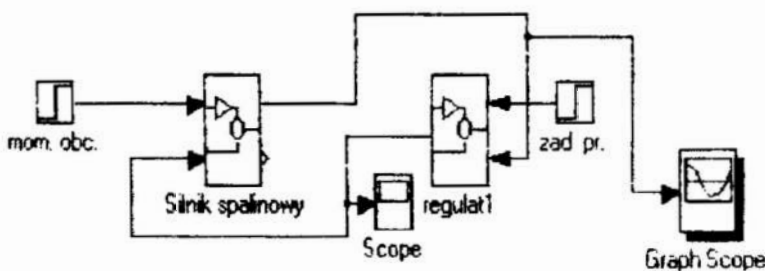


Rys. 4. Struktura modelu śruby zredagowana w programie SIMULINK
 Fig. 4. The structure of the propeller model used in program SIMULINK

Badania symulacyjne zespołów napędowych

układ regulacji prędkości obrotowej silnika

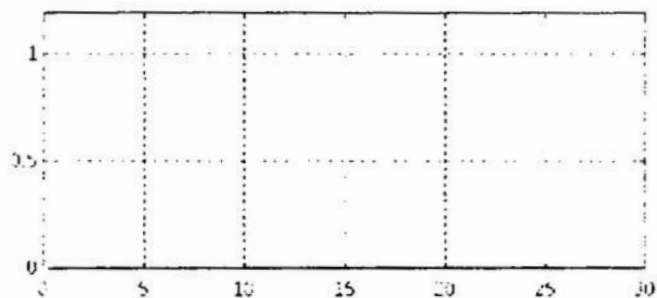
W celu oceny poprawności zredagowanych modeli i wstępnego doboru parametrów



Rys. 5. Struktura układu regulacji prędkości obrotowej
 Fig. 5. The structure of the rotational speed control system

wykonano badania symulacyjne układu regulacji prędkości obrotowej silnika (rys. 5). Oceniano pracę silnika bez obciążenia ($m_o = 0$), dla skokowej zmiany zadanej prędkości obrotowej od wartości minimalnej do nominalnej. W trakcie badań przebieg prędkości obrotowej silnika wyświetlany jest w oknie graficznym (Graph Scope), a wybrane przebiegi w razie potrzeby wyświetlane są w okienkach (Scope). Do symulacji, ze względu na szybkość obliczeń przyjęto

procedurę Linsim o minimalnym kroku 0.0001, maksymalnym kroku 0.01 i tolerancji

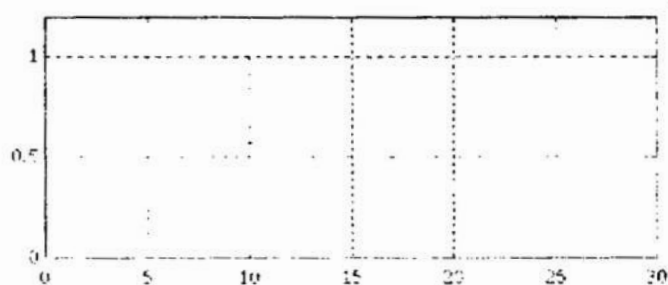


Rys. 6. Przebieg prędkości obrotowej układu

Fig. 6. The simulated rotational speed of the control system

Przeprowadzone badania wykazały niewłaściwą pracę zespołu turbodoładowania - zbyt mała porcja powietrza w stosunku do dawki paliwa, co wynikało z uproszczenia modelu (6) i przyjęcia wartości współczynników modelu $m, n = 1$.

W celu poprawy pracy zespołu

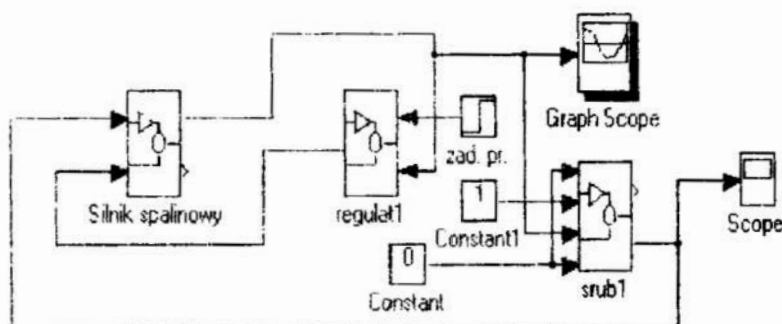


Rys. 7. Przebieg prędkości obrotowej silnika po zmianie parametrów modelu

Fig. 7. The simulated rotational speed of Diesel engine with new parameters

zjawiska niepełnego spalania stosuje się specjalne przystawki przeciwdymieniowe regulatora [5].

Zespół napędu głównego statku 1



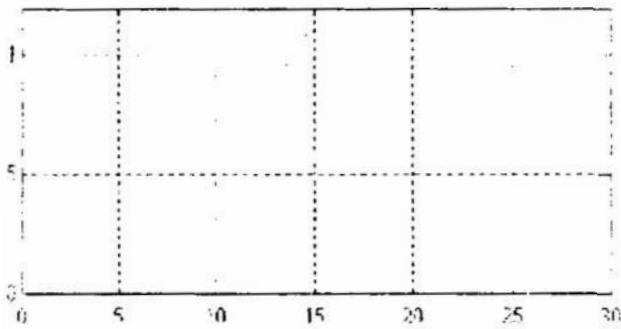
Rys. 8. Struktura zespołu napędowego 1

Fig. 8. The structure of the propulsion unit 1

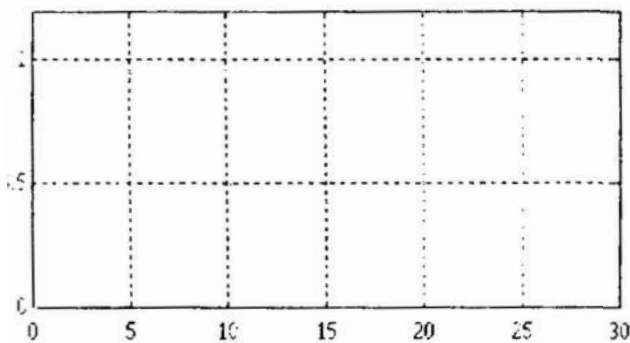
0.001. Uzyskany przebieg prędkości obrotowej (rys. 6) wykazuje zbyt długi czas osiągnięcia wartości zadanej (ok. 6s.) w stosunku do obiektu rzeczywistego. Wykorzystując pomocnicze okna stwierdzono, że sygnał listwy paliwowej przy osiągnięciu wartości zadanej posiada wartość $h_f = 1$, a więc silnik otrzymuje maksymalną dawkę paliwa. W celu stwierdzenia, czy zachodzi proces pełnego spalania zaobserwowano przebieg sygnału sterującego klucz hw2 (rys. 1).

turbodoładowania zmieniono wartości współczynników $a = 1,2; b = 1,8$ (rys. 1). Jednocześnie ze względu na niewielką oscylacyjność przebiegu prędkości obrotowej zmieniono statyzm regulatora z 7% ($\delta = 0,07$) na 10% ($\delta = 0,1$). Powtórne badania symulacyjne (rys. 7) wykazują przebieg prędkości obrotowej zbliżony do rzeczywistego obiektu. Obserwacja sygnału sterującego klucz $h=hw2$ wykazuje, że w dalszym ciągu przy narastaniu prędkości obrotowej występuje zjawisko niepełnego spalania, co ma miejsce również w układach rzeczywistych. W celu uniknięcia

Do badań zespołu napędu głównego statku 1 zredagowano strukturę i zredagowano model z zastosowaniem makrobloków (rys. 8). Ponieważ istnieje "sztywne" połączenie wału silnika z śrubą, część modelu dotycząca dynamiki śruby - wy (rys.4) nie jest wykorzystana.

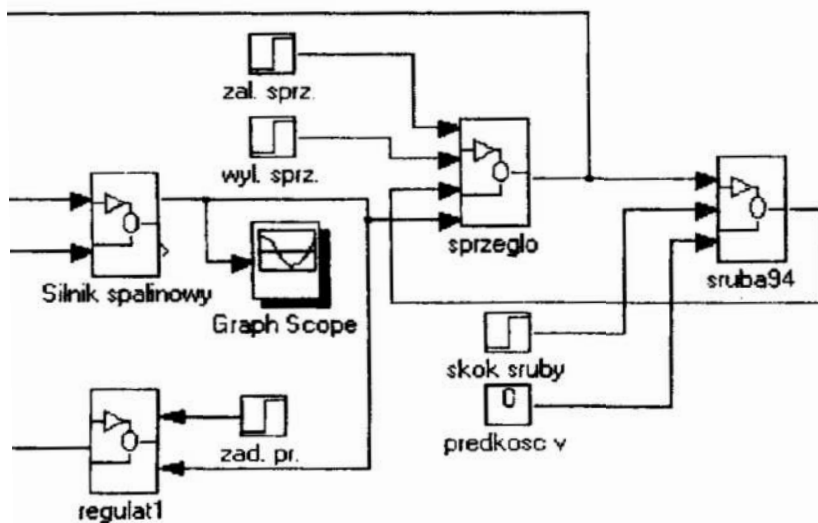


rys. 9. Przebieg prędkości obrotowej silnika zespołu napędowego 1 przy przeciążeniu
Fig. 9. The simulated rotational speed of Diesel engine propulsion unit at overload



rys.10. Przebieg prędkości obrotowej silnika zespołu napędowego 1 po zmianie parametrów
Fig. 10. The simulated rotational speed of Diesel engine propulsion unit with new parameters

Fig. 10. The simulated rotational speed of Diesel engine propulsion unit with new parameters



Rys. 11. Struktura zespołu napędowego 2

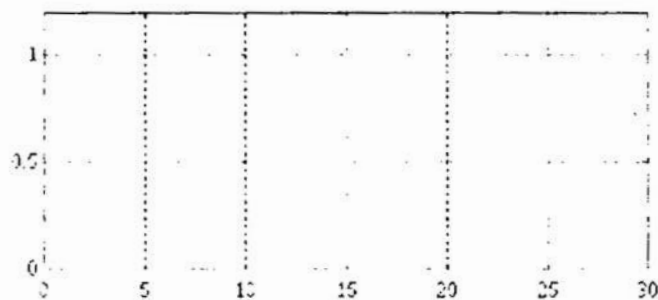
Fig. 11. The structure of the propulsion unit 2

wartości znamionowej i ustala się w czasie ok. 8s. W chwili $t = 12s$ następuje

Blok stałej wartości (Constant1) podaje na wejście modelu śruby przyjętą wartość skoku śruby h_p , a blok (Constant) przyjętą wartość prędkości liniowej statku względem wody. Przeprowadzono badania symulacyjne zespołu napędu głównego statku dla $h_p = 1$ i $v = 0$. Otrzymany przebieg prędkości obrotowej przy skokowej zmianie wartości zadanej przedstawia rys. 9. Prędkość obrotowa osiąga za małą wartość w stosunku do wartości znamionowej, przy przyjętym statycznie regulatora $\delta = 0.1$. Obserwacja przebiegu wartości listwy paliwowej wykazała przeciążenie silnika, co wskazuje na źle dobraną śrubę. Powtórzono badania symulacyjne dla wartości skoku śruby $h_p = 0.5$ uzyskując zadowalający przebieg prędkości obrotowej zespołu napędowego (rys. 10).

Zespół napędu głównego statku 2

Do badań zespołu napędu głównego statku o strukturze 2 zredagowano model z wykorzystaniem makrobloków (rys. 11). Zastosowano elementy zespołu napędowego przedstawione w p.2 referatu. Model pobudzany jest wymuszeniami skokowymi, które występują w określonych chwilach czasu. Wybór chwil czasu i wielkości skoku jednostkowego umożliwia w jednym cyklu badań symulacyjnych określenie przebiegów przy zmianach wielkości sterujących. Przyjęto, że w chwili $t = 0s$ następuje skok wartości zadanej prędkości obrotowej. Prędkość obrotowa wału silnika (rys. 12) wzrasta w czasie 4s do wartości znamionowej i ustala się w czasie ok. 8s. W chwili $t = 12s$ następuje



Rys. 12. Przebieg prędkości obrotowej silnika zespołu napędowego 2

Fig. 12. The simulated rotational speed of Diesel engine of the propulsion unit 2

16s - 20s) świadczą o niewłaściwie dobranym sprzęgle podatnym.

5. Podsumowanie

Prowadzenie badań symulacyjnych z zastosowaniem wielopoziomowych hierarchicznych struktur modelu jest bardzo wygodne. Znajduje szczególne zastosowanie przy analizie, badaniach symulacyjnych i syntezie układów automatyki i sterowania złożonych z gotowych podzespołów. Proces syntezy prowadzi do wielokrotnego wyboru elementów i struktur, których przydatność jest oceniana za pomocą metod symulacyjnych.

Zaprezentowane modele elementów nie tworzą jeszcze pełnego zbioru umożliwiającej symulację dowolnej struktury zespołu napędu głównego statku, co widać na przykładzie zespołu 1. Zastosowanie "sztywnego" połączenia wału silnika ze śrubą powoduje utratę możliwości modelowania dynamiki wału i śruby. Zbiór elementów zespołu napędu głównego statku należy rozszerzyć o model wału śrubowego.

Program SIMULINK jest wygodnym narzędziem do prowadzenia badań symulacyjnych wielopoziomowych i hierarchicznych modeli. Wprowadzanie zmian połączeń jest łatwe, można obserwować przebiegi sygnałów na każdym poziomie i w każdym punkcie, co upraszcza ocenę poprawności i funkcjonowania modeli. Redagowanie modeli przy użyciu wyłącznie bloków standardowych programu na podstawie opisu matematycznego jest niekiedy dość uciążliwe. Szczególne kłopoty stwarzają funkcje złożone z odcinkami przyporządkowanych zachodzącym relacjom między zmiennymi. W tym przypadku można posłużyć się metodyką modelowania zaproponowaną w [6].

Korzystanie z dojrzałych narzędzi jakim jest program SIMULINK nie zwalnia użytkownika z zachowania krytycznej oceny i "inżynierskiego" spojrzenia na otrzymane wyniki symulacji. Program SIMULINK przy "zwijaniu" modeli - tworzeniu makrobloków potrafi "zgubić" połączenia lub zmienić strukturę. Prowadzenie badań symulacyjnych z zastosowaniem hierarchicznych wielopoziomowych modeli układów wymaga sprawdzenia poprawności zredagowania i funkcjonowania każdego modelu cząstkowego zarówno postaci rozwiniętej - możliwość podglądu dowolnego sygnału w oknie, jak również postaci zwiniętej do makrobloku. Szybkość prowadzonej symulacji związana jest z wyborem procedury całkowania numerycznego i jej parametrów. Przyjęcie większych kroków całkowania przyspiesza obliczenia numeryczne, jednakże zniekształca uzyskane wyniki. W przypadku symulacji złożonych, hierarchicznych i wielowarstwowych układów sterowania błędy obliczeń potęgują się i należy stosować bardzo małe kroki całkowania.

załączenie sprzęgła o skoku stałym $h_p = 1$. Prędkość wału silnika na skutek wzrostu momentu chwilowo maleje osiągając 0, wartości znamionowej (zgodnie z nastawionym statyzmem regulatora). W przedziale czasu 12s - 16s można zaobserwować wpływ przenoszenia momentu przez sprzęgło cierne. W chwili $t = 20$ następuje wyłączenie sprzęgła. Skokowa zmiana momentu powoduje chwilowe przeregulowanie prędkości obrotowej wału silnika. Występujące niewielkie oscylacje przy załączonym sprzęgle (przedział czasu

liczenie symulowanego przebiegu prędkości obrotowej silnika zespołu napędowego 2 (rys. 12) przy wspomnianej w referacie procedurze i jej parametrach (p. 3) dla komputera IBM z procesorem 386, koprocesorem matematycznym i zegarem 25MHz trwało ok. 30 minut.

Przedstawione w referacie prace są związane z projektem badawczym dotyczącym modeli, programów i metodyki projektowania okrętowych systemów energetycznych.

Literatura

Kowalski Z., Tkaczyk D., Arendt R., Imajew D.: *Zastosowanie programów CLASSiC i 4TLAB/SIMULINK do analizy, syntezy i badań symulacyjnych zespołów napędowych statków*. XII Krajowa Konferencja Automatyki, Gdynia 6-8 września 1994.

Rimvall M.: *Computer-Aided Control System Design*. IEE Control System, vol. 13, Nr April 1993.

Szymkat M.: *Komputerowe wspomaganie w projektowaniu układów regulacji*. Warszawa, WNT 1993.

Kowalski Z.: *Badania symulacyjne podsystemów napędowych statków*. Zeszyty naukowe Politechniki Gdańskiej. Elektryka Nr 49, 1980.

Kowalski Z., Tittenbrun S., Łastowski W. F.: *Regulacja prędkości obrotowej okrętowych silników spalinowych*. Wydawnictwo Morskie, Gdynia 1988.

Arendt R.: *Metoda modelowania symulacyjnego obiektów o charakterze ciągłym z czasem matematycznym zależnym od dyskretnych sygnałów sterujących*. VII Krajowa Konferencja Dyskretnych Procesów Przemysłowych, Kozubnik k. Porąbki 16 - 19 września 1992.

Gajek J.: *Zespół silnika wysokoprężny - śruba nastawna - kadłub statku jako obiekt sterowania*. Budownictwo Okrętowe 1972, nr 2.

Streszczenie

W referacie omówiono zastosowanie programu SIMULINK do symulacji złożonych, nieliniowych modeli układów regulacji i sterowania o wielopoziomowej strukturze. Do obliczeniowych obliczeń symulacyjnych przyjęto modele: wysokoprężnego silnika diesla jako obiektu regulacji prędkości obrotowej, regulatora prędkości obrotowej, zębla ciernego i podatnego, śruby o skoku stałym i nastawnym, których opisy tematyczne zaczerpnięto z literatury. Przedstawiono wyniki badań symulacyjnych układu regulacji prędkości obrotowej wysokoprężnego silnika okrętowego oraz zespołu napędu głównego statku w konfiguracji 1: silnik spalinowy pracujący przez przekładnię, wał śrubowy na śrubę o skoku stałym oraz w konfiguracji 2: silnik spalinowy, pracujący przez sprzęgło, przekładnię, wał śrubowy na śrubę o skoku stałym. Omówiono otrzymane wyniki symulacji. Badanie różnych struktur zespołów napędu głównego statku wykazało, że dla poprawnego modelowania dynamiki wału i śruby należy rozszerzyć zbiór uwzględnianych elementów o model wału śrubowego. Oceniono użyteczność programu SIMULINK przy prowadzeniu badań symulacyjnych złożonych układów regulacji i sterowania o wielopoziomowej i hierarchicznej strukturze.

THE MODELLING OF MARINE PROPULSION UNIT CONTROL SYSTEMS PROPERTY WITH AN APPLICATION OF PROGRAM SIMULINK

Summary

In the paper an application of program SIMULINK for simulation of complex nonlinear control systems with hierarchical and multilevel structures is presented. For an exemplary simulation the models of:

- Diesel engine as the a control object of the rotational speed.
- the rotational speed governor.
- the friction and flexible coupling.
- the propeller with constant and adjustment pitch

are included. The mathematical descriptions of the models were taken from papers. The results of the simulation investigations of the rotational speed control system for the marine Diesel engine and the propulsion unit control systems in following configurations

1) the Diesel engine working through the gear, the propeller shaft on the propeller with a constant pitch

2) the Diesel engine working through the coupling, the gear, the propeller shaft on the propeller with constant pitch;

are illustrated. The rotational speed of the Diesel engine as the step responses on the set value of the rotational speed and on the set on (off) of the coupling are simulated. The achieved results of the simulations are discussed.

The simulation investigations of different structures of propulsion unit control system proved, that for correct modelling of the propeller shaft and the propeller screw dynamic the set of elements (models) must be enlarged. The application of the propeller shaft model is necessary.

The usefulness of the program SIMULINK at simulation investigations of complex control systems with hierarchical and multilevel structures is shown.

Jerzy BALICKI *

NEURONOWA POLIOPTYMALIZACJA PRZYDZIAŁÓW MODUŁÓW PROGRAMOWYCH W ROZPROSZONYCH SYSTEMACH KOMPUTEROWYCH

1. Wprowadzenie

W połowie lat osiemdziesiątych trwają intensywne prace nad wykorzystaniem sztucznych sieci neuronowych do rozwiązywania zagadnień optymalizacyjnych. Symulacyjne badania obliczeniowe wykazały, że możliwe jest rozwiązanie wybranych problemów optymalizacyjnych w czasie rzeczywistym (rzędu mikrosekund) [4]. Neuronowe podejście do rozwiązywania problemu komiwojażera zostało zaproponowane przez Karana i Hopfielda [13]. Wprawdzie późniejsze prace [1], [11] nie potwierdziły w pełni wyników, które uzyskali powyżsi autorzy, to jednak obecnie opracowywane są różne metody optymalizacyjne wykorzystujące sztuczne sieci neuronowe (w skrócie ANN).

Metoda minimalizacji w liniowym zadaniu programowania matematycznego została przedstawiona przez Cichockiego i Unbehauena [4]. Neuronową metodę optymalizacji za pomocą mnożników Lagrange'a opisali Zhang i Constantinescu [14]. Jedną z nieliniowych metod dla problemów z ograniczeniami zaproponowali Lillo, Hui i Żak [9]. Do wartościowych szczegółowych metod neuronowych zalicza się metodę Tagliariniego, Christa i Page'a [12], która rozwiązuje pewne nieliniowe zadanie programowania dyskretnego przydzielając wyrzutnie raketowe do zbioru ruchomych celów. Za pomocą komputera wieloprocesorowego LOREAL dokonano cyfrowej symulacji ANN o 46 tysiącach neuronów i 49 milionach połączeń synaptycznych. Ponadto pojawiły się metody, w których wykorzystuje się podobieństwo strukturalne ANN do sieci klasycznych procesorów. Sun i Fu [10] zaproponowali system wieloprocesorowy o strukturze ANN do rozwiązania problemu komiwojażera oraz zadania wyznaczenia cyklu Hamiltona.

W pracy scharakteryzowano ANN, która może być wykorzystana do wyznaczenia przydziałów modułów rozproszonego programu sekwencyjnego w wieloprocesorowych systemach komputerowych (RSK). Problem optymalizacyjny sformułowano w postaci zadania optymalizacji wielokryterialnej. Sieć neuronowa w stanie równowagi umożliwia wyznaczenie rozwiązania subkompromisowego dla $p=1$. Model ANN symulowano w środowisku Unix/Informix/C na mikrokomputerze typu IBM PC.

2. Optymalizacja przydziału programów w RSK

Większość zadań optymalizacji przydziałów modułów programowych w rozproszonych systemach komputerowych należy do klasy zadań NP-zupełnych [3]. Wyjątkiem jest zadanie poszukiwania optymalnych przydziałów modułów sekwencyjnego programu rozproszonego w systemie dwuprocessorowym z nieograniczonymi zasobami, które można rozwiązać za pomocą metody Stone'a [3] o złożoności obliczeniowej rzędu $O(V^3)$, gdzie V jest liczbą modułów programu. Jeżeli powyższy problem zostanie zmodyfikowany poprzez wprowadzenie ograniczenia związanego ze skończoną wielkością pamięci procesorów, to przekształcone zadanie należy już do klasy problemów NP-zupełnych.

Klasycznym zadaniem NP-zupełnym jest zagadnienie wyznaczania optymalnego przydziału V modułów programowych sekwencyjnego programu rozproszonego do dwóch procesorów tak, aby czas przetwarzania programów był jak najkrótszy przy spełnionych ograniczeniach na pamięć procesorów.

Niech

$x_{vi} = 1$, gdy moduł m_v przydzielono do procesora π_i , oraz 0 w przeciwnym razie dla $v = \overline{1, V}$; $i = \overline{1, 2}$;

t_{vi} - czas przetwarzania modułu na procesorze π_i ,

τ_{vu} - łączny czas komunikacji modułów m_v i m_u ($v < u$) przetwarzanych na różnych procesorach, dla $v, u = \overline{1, V}$;

d_i - wielkość pamięci dostępnej procesorowi π_i , dla $i = \overline{1, 2}$;

c_v - zajętość pamięci przez moduł m_v , dla $v = \overline{1, V}$;

Zadanie minimalizacji czasu przetwarzania sekwencyjnego programu rozproszonego na dwóch procesorach można sformułować w postaci klasycznego zadania programowania matematycznego z jednym kryterium [3]:

Dla danych : t_{vi} dla $v = \overline{1, V}$; $i = \overline{1, 2}$; τ_{vu} dla $v = \overline{1, V}$; $u = \overline{1, V}$;
 c_v dla $v = \overline{1, V}$; d_i dla $i = \overline{1, 2}$;

wyznaczyć $X^* \subset X$

taki, że $F_2(x^*) = \min_{x \in X} F(x)$ dla $x^* \in X^*$ (1)

gdzie

$$X = \{x \in B^{2V} \mid x = (x_{11}, \dots, x_{vi}, \dots, x_{V2})^T, \\ \sum_{v=1}^V c_v x_{vi} \leq d_i \quad i = \overline{1, 2}; \\ \sum_{i=1}^2 x_{vi} = 1 \quad v = \overline{1, V}; \}$$

$$B = \{0,1\}$$

$$F_2(x) = \sum_{v=1}^V \sum_{i=1}^2 t_{vi} x_{vi} + \sum_{v=1}^V \sum_{u=1}^V \sum_{i=1}^2 \tau_{vu} x_{vi} (1 - x_{ui})$$

Zadanie (1) jest nieliniowym zadaniem programowania binarnego. Ponieważ $x \in B^{2^V}$, to możliwych jest 2^{2^V} przydziałów modułów do procesorów.

3. Zadanie polioptymalizacji przydziału modułów programowych

Zbiór modułów programowych może być przetwarzany na różnych typach procesorów. Rozpatrywany problem projektowy polega na doborze pary procesorów, na której lokuje się rozdziału modułów programowych tak, aby koszt procesorów był jak najmniejszy, a czas przetwarzania modułów był jak najkrótszy.

Niech

$\Pi = \{\pi_1, \dots, \pi_j, \dots, \pi_J\}$ - zbiór typów procesorów,

δ_j - koszt procesora typu π_j ,

$V = \{w_1, w_2\}$ - zbiór węzłów przetwarzania, w którym mogą być przetwarzane moduły programowe.

$x_{ij}^\pi = 1$, jeżeli procesor typu π_j przydzielono do węzła w_i . $x_{ij}^\pi = 0$ w przeciwnym razie.

Postać zadania polioptymalizacji zależy od sytuacji decyzyjnej, na którą wpływają głównie preferencje decydenta. Decydem może być projektant, zleceniodawca, poleczeństwo itd. Preferencje decydenta determinują dobór kryteriów cząstkowych i relacji dominowania. W zagadnieniach technicznych często przyjmuje się preferencje prowadzące do rozwiązań optymalnych w sensie Pareto, leksykograficznych lub kompromisowych. Natomiast dobór kryteriów cząstkowych zależy od preferencji projektanta. Specyfika modelowanej sytuacji decyzyjnej polega na dążeniu do wyznaczania rozwiązań o jak najmniejszym koszcie oraz jak najkrótszym czasie realizacji programów, przy czym nie preferuje się żadnego z kryteriów. Pożądanymi są zatem rozwiązania dominujące w sensie relacji " \leq ". Często jednak zbiór rozwiązań dominujących jest zbiorem pustym, gdyż punkt dominujący w przestrzeni kryterialnej musiałby być również punktem idealnym [2]. Dlatego wygodniej jest sformułować problem projektowy w postaci zadania optymalizacji wektorowej, w którym poszukuje się rozwiązań optymalnych w sensie Pareto (rozwiązań niezdominowanych w sensie relacji " \leq "). W przypadku zbyt dużej liczby rozwiązań optymalnych w sensie Pareto dokonuje się selekcji końcowej wybierając rozwiązania kompromisowe za pomocą p-normy. Rozwiązania optymalne w sensie Pareto są szczególnie przydatne w sytuacji, gdy wykorzystuje się w następnym etapie optymalizacji dodatkowe kryterium (niekoniecznie p-normę) do dokonania selekcji końcowej. Powyższe kryterium selekcji końcowej nie musi być obiektywne tzn. opisane analitycznie lub algorytmicznie. W szczególnych przypadkach kryterium selekcji może wyrażać subiektywne nieujawnione preferencje decydenta, co powoduje konieczność wykorzystania dialogowych metod polioptymalizacji.

Proponuje się sformułować zadanie wyznaczania optymalnych w sensie Pareto przydziałów modułów programowych w postaci uporządkowanej trójki (X, F, \leq) , gdzie

1) X - zbiór rozwiązań dopuszczalnych

$$X = \{x \in B^{2V+J} \mid x = (x_{11}, \dots, x_{v1}, \dots, x_{v2}, x_{11}^\pi, \dots, x_{ij}^\pi, \dots, x_{2J}^\pi)^T\};$$

x spełnia poniższe ograniczenia:

a) ogr. przydziału wszystkich modułów do węzłów

$$\sum_{i=1}^2 x_{vi} = 1 \quad \text{dla } v = \overline{1, V};$$

b) ogr. na niezbędne zasoby przetwarzania danych w węzłach

$$\sum_{v=1}^V \sum_{j=1}^J c_v x_{vi} x_{ij}^\pi \leq \sum_{j=1}^J d_j x_{ij}^\pi \quad \text{dla } i = \overline{1, 2};$$

c) ogr. przydziału co najwyżej jednego komputera do węzła

$$\sum_{j=1}^J x_{ij}^\pi \leq 1 \quad \text{dla } i = \overline{1, 2}; \quad \}$$

2) F - kryterium jakości

$$F : X \rightarrow R^2$$

$$F(x) = [F_1(x), F_2(x)]^T \quad \text{dla } x \in X$$

$F_1(x)$ - wartość liczbowa kosztu procesorów,

$F_2(x)$ - wartość liczbowa czasu przetwarzania modułów programowych

$$F_1(x) = \sum_{i=1}^2 \sum_{j=1}^J \delta_j x_{ij}^\pi$$

$Y = F(X)$ - przestrzeń kryterialna (zbiór ocen).

3) " \leq " - relacja dominowania w przestrzeni kryterialnej

Niech $a, b \in Y$, $a = (a_1, a_2)$, $b = (b_1, b_2)$

$$" \leq " = \{ (a, b) \in Y \times Y \mid a_1 \leq b_1, a_2 \leq b_2 \} \in Y \times Y.$$

W przypadku powyższego zadania optymalizacji wielokryterialnej, kres dolny zbioru ocen Y w przestrzeni R^2 z relacją " \leq " jest zbiorem jednoelementowym, zawierającym element y^0 , zwany punktem idealnym (celem). Gdy liczność zbioru punktów idealnych jest większa niż 1, to powstaje zadanie optymalizacji przy wielu celach. Współrzędne punktu utopijnego spełniają równania:

$$Y_n^0 = \inf F_n(x) \quad \text{dla } n = \overline{1, 2}; \quad (3)$$

Zbiór ocen Y jest zbiorem ograniczonym ze względu na skończony zbiór X i postać kryteriów cząstkowych, które przyjmują skończone wartości dla każdego rozwiązania dopuszczalnego. Niech Λ oznacza stożek generujący relację " \leq ". Łatwo pokazać, że zbiór ograniczeń dolnych zbioru Y w sensie relacji " \leq " jest zbiorem powstałym przez dodanie stożka Λ do punktu idealnego y :

$$l_d(Y, \leq) = \{y\} + \Lambda \quad (4)$$

Zbiór ograniczeń dolnych posiada zatem element największy i jest to element jedyny:

$$\inf_{\leq} Y = \{l_d(Y, \leq)\}_{\sup}^{\leq} = \{ \{y^0\} + \Lambda \}_{\sup}^{\leq} = \{y^0\}$$

Zwykle $y^0 \in Y$. Dla zadanego y^0 , po skalaryzacji zbioru ocen, można sformułować zadanie wyznaczania rozwiązań kompromisowych w postaci (\bar{Y}, R_0) , gdzie

\bar{Y} - wyskalowany zbiór ocen [2],

R_0 - relacja bliskości ocen od punktu idealnego,

$$R_0 = \{ (a, b) \in \mathbb{R}^2 \times \mathbb{R}^2 \mid \|y^0 - a\|_p \leq \|y^0 - b\|_p \text{ dla } p \geq 1 \}$$

$$\|r\|_p = \left(\sum_{n=1}^2 |r_n|^p \right)^{1/p} \quad \text{dla } r \in \mathbb{R}^2 \quad (5)$$

Znając zbiór rozwiązań optymalnych w sensie Pareto, rozwiązania kompromisowe wyznacza się za pomocą rozwiązania poniższego problemu:

$$\|y^0 - y^k\|_p = \min_{y \in Y_{\leq}^N} \|y^0 - y\|_p \quad (6)$$

Ponieważ \bar{Y}_{\leq}^N jest podzbiorem domkniętego zbioru \bar{Y} , to

$$\|y^0 - y^k\|_p = \min_{y \in \bar{Y}} \|y^0 - y\|_p \quad (7)$$

Jeżeli minimalizowane są $F_1(\cdot)$ oraz $F_2(\cdot)$, to punkt idealny po skalaryzacji typu :

$$\bar{y}_n = \frac{y_n}{y_n^0} \text{ posiada współrzędne jednostkowe } y^0 = [1, 1]^T.$$

Dla $p = 1$ kryterium "kompromisu" ma postać:

$$K_1 = \sum_{n=1}^2 |1 - \bar{y}_n| \quad (8)$$

Dla $p = 2$ kryterium "kompromisu" jest normą euklidesową :

$$K_2 = \sqrt{(1 - \bar{y}_1)^2 + (1 - \bar{y}_2)^2} \quad (9)$$

Dla $p = \infty$ kryterium "kompromisu" ma postać:

$$K_{\infty} = \max_{n \in \{1, 2\}} \{ |1 - \bar{y}_n| \} \quad (10)$$

4. Neuronowa metoda wyznaczania rozwiązań kompromisowych dla $p=1$

Jeżeli w związku z własnościami zadania polioptymalizacji (2) problem wyznaczania rozwiązań kompromisowych można sprowadzić do zadania optymalizacji dyskretnej:

$$\min_{x \in X} K_p(x) \quad (11)$$

gdzie $K_p(x)$ - kryterium kompromisowe dla $p=1$ (8), $p=2$ (9) lub $p = \infty$ (10).

Dla $p=1$. Na podstawie (1), (2) i (9) kryterium kompromisowe ma postać (12):

$$K_1(x) = \frac{1}{y_1^0} \sum_{i=1}^2 \sum_{j=1}^J \delta_j x_{ij}^{\pi} + \frac{1}{y_2^0} \sum_{i=1}^2 \sum_{v=1}^V \sum_{j=1}^J t_{vj} x_{ij}^{\pi} x_{vi} + \sum_{i=1}^2 \sum_{v=1}^V \sum_{u=1}^V t_{vu} (1 - x_{ui}) x_{vi} - 2$$

Do wyznaczania rozwiązań kompromisowych proponuje się wykorzystać odpowiednio zaprojektowaną sieć neuronową (KANN) o $2V+I(J+2)$ neuronach. Wyjście neuronu reprezentuje zmienną decyzyjną x_{vi} , x_{ij}^{π} lub zmienną pozorną $s_{i,z}$ i . KANN umożliwia wyznaczenie pojedynczego rozwiązania subkompromisowego, po osiągnięciu stanu równowagi, rozpoczynając obliczenia od stanu początkowego. W stanie równowagi sieć wyznacza minimum lokalne. Aby uzyskać minimum globalne (w rozpatrywanym problemie rozwiązanie kompromisowe) generowane są rozwiązania startowe, które pokrywają równomiernie badany obszar. Rozpoczynając projektowanie KANN należy zapewnić jej stabilność przy przejściu z punktu początkowego do stanu równowagi wzdłuż trajektorii. Cohen i Grossberg [5] podali warunki stabilności ANN:

$$\frac{du_n}{dt} = a_n(u_n) \left[b_n(u_n) - \sum_{m=1}^N w_{nm} g_m(u_m) \right] \quad \text{dla } n = \overline{1, N}; \quad (13)$$

gdzie

N - liczba neuronów w sieci,

u_n - sygnał w neuronie x_n podlegający przekształceniu przez funkcję aktywacji $g_n(u_n)$, która nie może być funkcją malejącą dla u_n ,

w_{nn} - symetryczne wagi synaptyczne ($w_{mn}=w_{nm}$ dla $m \neq n$, $w_{nn}=0$)

a_n, b_n - funkcje u_n , na które nałożono warunek, aby dla nieujemnych wartości $a_n(\cdot)$ były funkcjami ciągłymi.

Sieć Hopfielda [7] spełnia warunki stabilności Cohena-Grossberga (13). W ciągłym modelu neuronu Hopfielda jego zachowanie opisano za pomocą poniższego układu równań różniczkowych :

$$\frac{du_n}{dt} = -\frac{u_n}{\eta_n} + \sum_{m=1}^N w_{nm} g_m(u_m) + I_n \quad \text{dla } n = \overline{1, N}; \quad (14)$$

η_n - współczynnik hamowania neuronu x_n ,

I_n - wartość wejścia zewnętrznego do neuronu x_n .

Wagi są symetryczne, funkcje aktywacji niemalejące, wyrażone za pomocą zależności:

$$g_n(u_n) = \frac{1}{2} [1 + \tanh(\alpha u_n)] \quad \text{dla } n = \overline{1, N}; \quad (15)$$

gdzie α jest współczynnikiem wzmocnienia.

Jeżeli wejścia zewnętrzne (biasy) nie zmieniają się w czasie, to sieć Hopfielda na mocy warunków stabilności Cohena-Grossberga osiągnie stan równowagi niezależnie od stanu początkowego. Hopfield wyprowadził następującą funkcję energetyczną (Lapunowa) [13] dla układu (14):

$$E = -\frac{1}{2} \sum_{m=1}^N \sum_{n=1}^N w_{nm} g_n(u_n) g_m(u_m) + \sum_{n=1}^N I_n g_n(u_n) \quad (16)$$

Sieć Hopfielda "nawigując" w kierunku punktu równowagi zmniejsza wartość funkcji energetycznej. W stanie równowagi nie można już dalej zmniejszyć wartości funkcji, zatem sieć wyznacza minimum lokalne funkcji energetycznej. Powyższa zaleta sieci Hopfielda została rozwinięta przy rozwiązywaniu różnych problemów optymalizacyjnych (kolorowanie i planaryzacja grafów, przydziały koncentratorów w sieci kom

puterowej, programowanie liniowe, kwadratowe i minimaksowe [12]). Dla KANN skonstruowano następującą funkcję energetyczną (metodą funkcji kary):

$$E_1 = K_1(x) + \beta \sum_{l=1}^L h_l(x) \quad (17)$$

gdzie

β - współczynnik skalujący składniki kary za niespełnienie ograniczeń,
 $h_l(x)$ - nieujemna funkcja kary za niespełnienie l-tego ograniczenia przez rozwiązanie x .

Niech funkcja kompromisowa $K_1(x)$ oprócz postaci (8) będzie wyrażona za pomocą formuły Hopfielda jako funkcja energetyczna :

$$K_1(x) = -\frac{1}{2} \sum_{n=1}^N \sum_{m=1}^N w_{nm}^K x_n x_m - \sum_{n=1}^N I_n^K x_n \quad (18)$$

gdzie

w_{nm}^K - szukane wagi synaptyczne dla funkcji kompromisowej,

I_n^K - szukane wejścia zewnętrzne dla funkcji kompromisowej.

Podobnie

$$h_l(x) = -\frac{1}{2} \sum_{n=1}^N \sum_{m=1}^N w_{nm}^l x_n x_m - \sum_{n=1}^N I_n^l x_n \quad (19)$$

Na podstawie (17), (18) i (19) dostaje się, jak niżej :

$$E_1 = -\frac{1}{2} \sum_{n=1}^N \sum_{m=1}^N (w_{nm}^K + \beta \sum_{l=1}^L w_{nm}^l) x_n x_m - \sum_{n=1}^N (I_n^K + \beta \sum_{l=1}^L I_n^l) x_n \quad (20)$$

Jeżeli funkcję kompromisową i ograniczenia można opisać w postaci funkcji energetycznej Hopfielda, to istnieje globalna funkcja energetyczna problemu, która przekształca pierwotne zadanie optymalizacyjne w równoważny problem bez ograniczeń. Ze wzoru (20) wynika, że waga między parą neuronów jest sumą wag z energetycznej funkcji celu i z energetycznych funkcji ograniczeń. Podobnie sumowane są wejścia zewnętrzne. Rozważa się neuronową implementację ograniczenia przydziału wszystkich modułów programowych do węzłów :

$$x_{v1} + x_{v2} = 1 \quad \text{dla } v = \overline{1, V};$$

Powyższe równanie wymaga, aby dokładnie jeden neuron był aktywny (stan "1" lub ON) w stanie równowagi KANN. Natomiast drugi neuron musi być wyhamowany (stan "0" lub OFF). Jako energetyczną funkcję powyższego ograniczenia proponuje się:

$$h_l = (1 - \sum_{i=1}^2 x_{vi})^2 + \sum_{i=1}^2 x_{vi}(1 - x_{vi}) \quad \text{dla } l = \overline{1, V}; \quad (21)$$

Kwadrat przy pierwszym składniku zapewnia, że wartość minimalna składnika wynosi 0 dla spełnionego ograniczenia. Natomiast drugi składnik zapewnia, że wartość minimalna tego składnika wynosi 0 dla binarnych wartości x_{vi} . Po elementarnych przekształceniach i sprowadzeniu funkcji energetycznej (21) do postaci (19) otrzymano wagi:

$$w_{vi, vj}^l = \begin{cases} -2 & \text{dla } vi \neq vj \\ 0 & \text{w przeciwnym razie} \end{cases} \quad v = \overline{1, V}; \quad i, j = \overline{1, 2}; \quad l = \overline{1, V}; \quad (22)$$

Natomiast wejścia zewnętrzne $I_{v_i}^l = 1$ dla $i = \overline{1, 2}$; $v = \overline{1, V}$;

Każdy neuron x_{v_j} posiada połączenie hamujące od neuronu x_{v_j} i pobudzające wejście zewnętrzne o wartości 1. Proste eksperymenty symulacyjne potwierdziły, że jest to podsieć typu "zwycięzca bierze wszystko". Proponuje się nazwać regułę (22) regułą wyboru pojedynczego neuronu z dwóch neuronów.

Ograniczenia przydziału co najwyżej jednego procesora do węzła :

$$\sum_{j=1}^J x_{ij}^\pi < 1 \quad \text{dla } i = \overline{1, I};$$

nie można uwzględnić w KANN za pomocą reguły wyboru pojedynczego neuronu z dwóch lub J neuronów. Dlatego też powyższe ograniczenie zostało zmodyfikowane do układu równań poprzez wprowadzenie zmiennej pozornej $s_i \in B$. Zatem

$$\left(\sum_{j=1}^J x_{ij}^\pi \right) + s_i = 1 \quad \text{dla } i = \overline{1, I}; \quad (23)$$

Dla węzła przetwarzania w_i projektowana podsieć KANN musi wybrać, w stanie równowagi dokładnie jeden neuron z $J+1$. Jeżeli zostanie uaktywniony neuron s_i , to żaden procesor nie zostanie przydzielony do węzła przetwarzania. Rozszerzając regułę (22) na regułę wyboru jednego neuronu spośród $J+1$, analogicznie do wyprowadzenia wzoru (22), otrzymuje się zależności:

$$W_{ij,ik}^l = \begin{cases} -2 & \text{dla } j \neq k \\ 0 & \text{w przeciwnym razie} \end{cases} \quad \text{dla } i = \overline{1, 2}; \quad l = \overline{V+1, V+2}; \quad (24)$$

$$I_{ij}^l = 2J + 1.$$

Ograniczenie na zasoby procesorów niezbędne do przetwarzania przydzielonych operacji ma postać :

$$\bar{h}_l(x) = \sum_{v=1}^V \sum_{j=1}^J c_{v_i} x_{v_i} x_{ij}^\pi - \sum_{j=1}^J d_j x_{ij}^\pi \leq 0 \quad \text{dla } i = \overline{1, 2};$$

Energetyczna funkcja Hopfielda dla powyższego ograniczenia przyjmuje postać :

$$h_l(x) = (-\bar{h}_l(x) - z_l)^2 + \sum_{v=1}^V x_{v_i}(1 - x_{v_i}) + \sum_{j=1}^J x_{ij}^\pi(1 - x_{ij}^\pi) + z_l(1 - z_l) \quad (25)$$

dla $l = \overline{V+3, V+4}$

Po elementarnych przekształceniach otrzymuje się następujące parametry KANN:

$$W_{ij,ik}^l = \begin{cases} -2d_j & \text{dla } j \neq k \\ 0 & \text{w przeciwnym razie} \end{cases} \quad \text{dla } i = \overline{1, 2};$$

$$W_{ij,i}^l = \begin{cases} -2d_j & \text{dla par } (x_{ij}^\pi, z_l), i \neq j \\ 0 & \text{w przeciwnym razie} \end{cases} \quad \text{dla } i = \overline{1, 2}; \quad (26)$$

$$W_{vi,ui}^l = \begin{cases} -2c_v & \text{dla } v \neq u \\ 0 & \text{w przeciwnym razie} \end{cases} \quad \text{dla } i = \overline{1, 2};$$

Natomiast wejścia zewnętrzne $I_{v_i}^l, I_i^l, I_{ij}^l$ są jednostkowe.

Analiza dla funkcji kompromisowej $K_1(x)$ daje następujące parametry KANN :

$$w_{vi,ij}^K = -\frac{2t_{vj}}{y_2^o} \quad \text{dla } i=\overline{1,2}; \quad j=\overline{1,J}; \quad v=\overline{1,V};$$

$$w_{vi,ui}^K = -\frac{2\tau_{vu}}{y_2^o} \quad \text{dla } i=\overline{1,2}; \quad j=\overline{1,J}; \quad v=\overline{1,V};$$
(27)

Wejścia zewnętrzne do neuronów x przyjmują wartości:

$$I_{ij}^K = \frac{\delta_j}{y_1^o} \quad \text{dla } i=\overline{1,2}; \quad j=\overline{1,J};$$

Po zsumowaniu wszystkich wag dla danej pary neuronów otrzymuje się zgodnie z zależnością (20) wagę danego połączenia. Analogicznie, wszystkie wejścia zewnętrzne danego neuronu zastępowane są jednym wejściem będącym ich sumą. Parametr skalujący β musi być dobrany metodą "prób i błędów". Z jednej strony, parametr powinien być odpowiednio duży, aby w funkcji energetycznej zostały nałożone kary, gdy KANN nie spełnia ograniczeń. Z drugiej strony, zbyt duża wartość parametru powoduje, że KANN oscyluje w pobliżu minimum lokalnego. Na podstawie przeprowadzonych eksperymentów oszacowano dziedzinę parametru jako [200;900]. Parametr wzmocnienia α nie wpływa na stabilność sieci dla $\alpha > 1$. Wstępne porównanie uzyskanych wyników za pomocą symulowanej cyfrowo KANN i metody klasycznej, która wykorzystuje metodę Stone'a wykazało, że KANN umożliwia wyznaczenie dokładniejszych rozwiązań. Natomiast metoda klasyczna szybciej kończy obliczenia. Ze względu na perspektywy implementacji KANN na komputerach neuronowych należy uznać uzyskane wyniki za obiecujące.

5. Zakończenie

W pracy zaproponowano neuronową sieć Hopfielda do optymalizacji przydziału modułów sekwencyjnego programu rozproszonego w dwuprocesorowym systemie z ograniczoną pamięcią. Sformułowano problem optymalizacyjny w postaci zadania optymalizacji wielokryterialnej. Skonstruowano odpowiednią funkcję Lapunowa wykorzystując metodę funkcji kary. Zastosowano ciągłą funkcję aktywacji neuronu Wyznaczono zależności definiujące wagi oraz wejścia zewnętrzne dla sieci neuronowej KANN, która w stanie równowagi osiąga rozwiązania subkompromisowe dla $p=1$.

Literatura

1. Aiyer S.V.B., Niranjana M., Fallside F.: *A Theoretical Investigation into the Performance of Hopfield Model*. IEEE Trans. on Neural Networks, vol.1, No.2.1990, pp.204-215.
2. Ameljańczyk A.: *Optymalizacja wielokryterialna*. WAT, Warszawa 1986.
3. Balicki J.: *Neuronowa optymalizacja przydziału programów w rozproszonych systemach komputerowych*. Materiały I Krajowej Konferencji "Sieci neuronowe i ich zastosowania", t.1, Częstochowa 1994, ss.136-141.
4. Cichocki A., Unbehauen R.: *Neural Networks for Solving Systems of Linear Equations and Related Problems*. IEEE Trans. on Circuits and Systems, vol. 39, No.2, February 1992, pp.124-137.

5. Cohen M.A., Grossberg S.: *Absolute Stability of Global Pattern Formation and Parallel Memory Storage by Competitive Neural Networks*. IEEE Trans. Syst., Man, and Cybern., vol. SMC-13, September/October 1983, pp.815-825.
6. Gołąb B.K.: *Anatomia czynnościowa ośrodkowego układu nerwowego*. PZWL, Warszawa 1992.
7. Hertz J., Krogh A., Palmer R.: *Wstęp do teorii obliczeń neuronowych*. WNT, Warszawa 1993.
8. Hoang T.K.: *Fuzzy Entropy in Decision Making System and Intelligent Control*. Proc. 6th Int. Sym. System-Modelling-Control, Zakopane 1990, pp.165-169.
9. Lillo W.E., Hui S., Żak H.: *Neural Networks for Constrained Optimization Problems*. Int. J. of Circuit Theory and Applications, vol.21, 1991, pp.385-399.
10. Sun K.T., Fu H.C.: *A Hybrid Neural Model for Solving Optimization Problems*. IEEE Trans. on Computers, vol.42, No.2, February 1993, pp.219-227.
11. Tadeusiewicz R.: *Sieci Neuronowe*. AOW, Warszawa 1993.
12. Tagliarini A., Christ J.F., Page W.: *Optimization Using Neural Networks*. IEEE Trans. on Computers, vol.40, No.12, December 1991, pp.1347-1357.
13. Tank D.W., Hopfield J.J.: *Simple "Neural" Optimization Networks : An A/D Converter, Signal Decision Circuit, and Linear Programming Circuit* IEEE Trans. on Circuits and Systems, vol.CAS-33, May 1986, pp.533-541.
14. Zhang S., Constantindes A.G.: *Lagrange Programming Neural Networks*. IEEE Trans. on Circuits and Systems, vol.39, No.7, July 1992.

Streszczenie

W pracy zaproponowano neuronową sieć Hopfielda do optymalizacji przydziału modułów sekwencyjnego programu rozproszonego w dwuprocessorowym systemie z ograniczoną pamięcią. Sformułowano problem w postaci zadania optymalizacji wektorowej. Skonstruowano funkcję Lapunowa dla rozważanego problemu optymalizacyjnego. Wyznaczono zależności wyznaczające wartości wag i wejść zewnętrznych dla sieci typu KANN, która w stanie równowagi osiąga rozwiązania subkompromisowe dla $p=1$.

NEURAL MULTICRITERIA OPTIMIZATION OF PROGRAM MODULES ALLOCATIONS IN DISTRIBUTED COMPUTER SYSTEM

Summary

In this paper the Hopfield's neural network for multicriteria optimization of program modules allocations in the two-processors system with the constrained memory has been proposed. The Liapunov function of that problem has been considered. Formulas for determination of weights and external inputs for the network called KANN have been presented. KANN in the equilibrium point shows the subcompromise solution with parameter $p=1$.

Jan BARCZYK

WSPOMAGANE KOMPUTEROWO PROJEKTOWANIE UKŁADÓW PRZENIESIENIA NAPĘDU W CHWYTAKACH ROBOTÓW PRZEMYSŁOWYCH

1. Wprowadzenie

Roboty przemysłowe w procesie technologicznym mogą realizować zadania transportowe lub technologiczne. Zadania transportowe polegają na dostarczaniu i odbieraniu obiektu z maszyny technologicznej (obrabiarki, wykrojnika, przenośnika taśmowego itp.) - w tego typu zastosowaniach roboty wyposażone są w urządzenia chwytające (chwytaki). Urządzenia chwytające są obecnie najbardziej zróżnicowanym konstrukcyjnie zespołem robotów przemysłowych, gdyż ogromna różnorodność obiektów manipulacji, wielorakość zadań, zmienność parametrów procesu manipulacji i inne warunki wymuszały tworzenie wciąż nowych konstrukcji chwytaków. Ocenia się, że w ponad 65% obecnych zastosowań robotów należało skonstruować specjalne urządzenie chwytające, a w pozostałych przypadkach możliwe było przystosowanie standardowego wyposażenia robota, tj. chwytaków ze sztywnymi końcówkami, podciśnieniowych lub elektromagnetycznych.

Waga problematyki chwytania obiektów manipulacji przez roboty przemysłowe nie jest ogólnie doceniana. Potencjalni użytkownicy, a często również projektanci i wykonawcy systemów zrobotyzowanych za najważniejsze uważają wybór typu robota (jego struktury kinematycznej i układu sterowania), traktując bezproblemowo urządzenia chwytające. Tymczasem wiele nieudanych prób zastosowania robotów, wiele uszkodzeń na zrobotyzowanym stanowisku, a często nawet nieszczęśliwe wypadki podczas pracy robota spowodowane zostały nieprawidłowym doбором chwytaka. Konstruktorzy widzą zwykle w chwytaku tylko mechanizm o jednym stopniu swobody, dla którego należy rozwiązać zagadnienia statyki - mimo złożoności wielu mechanizmów chwytaków (skomplikowanych zależności geometrycznych), rozwiązanie zagadnień statyki nie stwarza problemów [1].

2. Wybór urządzenia chwytającego

W procesie projektowania zrobotyzowanej technologii stawiane jest zadanie wyboru urządzeń zapewniających wypełnienie przez robota założonych zadań: określenia położenia obiektu, jego kształtu, manipulowanie obiektem w przestrzeni roboczej, umiejscowienie i uwolnienie przeniesionego obiektu a czasem również uchwycenie i posługiwanie się narzędziem.

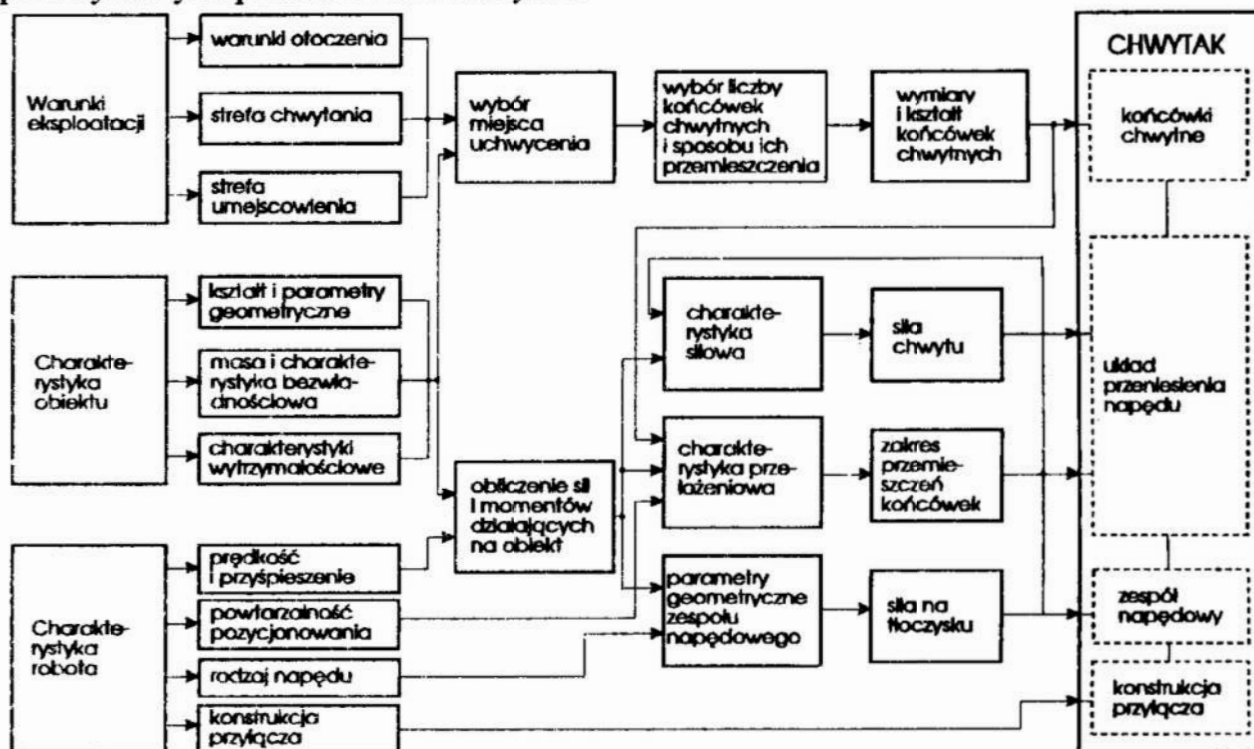
W zakresie chwytaków osiągnięto znaczący postęp - ciągle są udoskonalane rozwiązania konstrukcyjne, stosowane są różnego rodzaju urządzenia sensoryczne, produkowane są systemy automatycznej wymiany chwytaków, rozwijane są prace nad budowa chwytaków wielopalcowych itp. Zdaniem specjalistów rozwiązaniem przyszłościowym są chwytaki uniwersalne (dłoniopodobne) o szerokich możliwościach funkcjonalnych.

3. Projektowanie chwytaków robotów przemysłowych

Podstawowe etapy projektowania chwytaków obejmują:

- 1) wybór miejsca uchwycenia,
- 2) obliczenie sił i momentów działających na obiekt w warunkach dynamicznych,
- 3) określenie liczby końcówek chwytnych, sposobu oraz zakresu ich przemieszczenia,
- 4) obliczenie układu przeniesienia napędu,
- 5) dobór układu napędowego,
- 6) opracowanie konstrukcji chwytaka, w tym także konstrukcji przyłącza.

Kolejność realizacji poszczególnych zadań podczas projektowania chwytaków robotów przemysłowych przedstawiono na rys. 1.



Rys. 1. Kolejność postępowania przy projektowaniu chwytaków

Fig. 1. Algorithms for designing robot's grippers

Dane wejściowe do projektowania chwytaków określa się na podstawie warunków eksploatacji zrobotyzowanego stanowiska, charakterystyki obiektu i charakterystyki robota przemysłowego.

Uwzględnienie warunków eksploatacji (w tym charakterystyki geometrycznej stanowiska) umożliwia:

- określenie typu wykonania chwytaka (normalne, pyłoszczelne, iskrobezpieczne itp.) oraz podanie warunków (parametrów otoczenia), przy których zapewnione są parametry chwytaka o żądanych parametrach technicznych i użytkowych,
- określenie wymiaru i kształtu końcówek chwytnych na podstawie sposobu bazowania obiektu oraz wymiarów strefy chwytania i strefy umiejscowienia,
- obliczenie układu przeniesienia napędu (zakres przemieszczeń końcówek).

Charakterystyka obiektu manipulacji ma wpływ na:

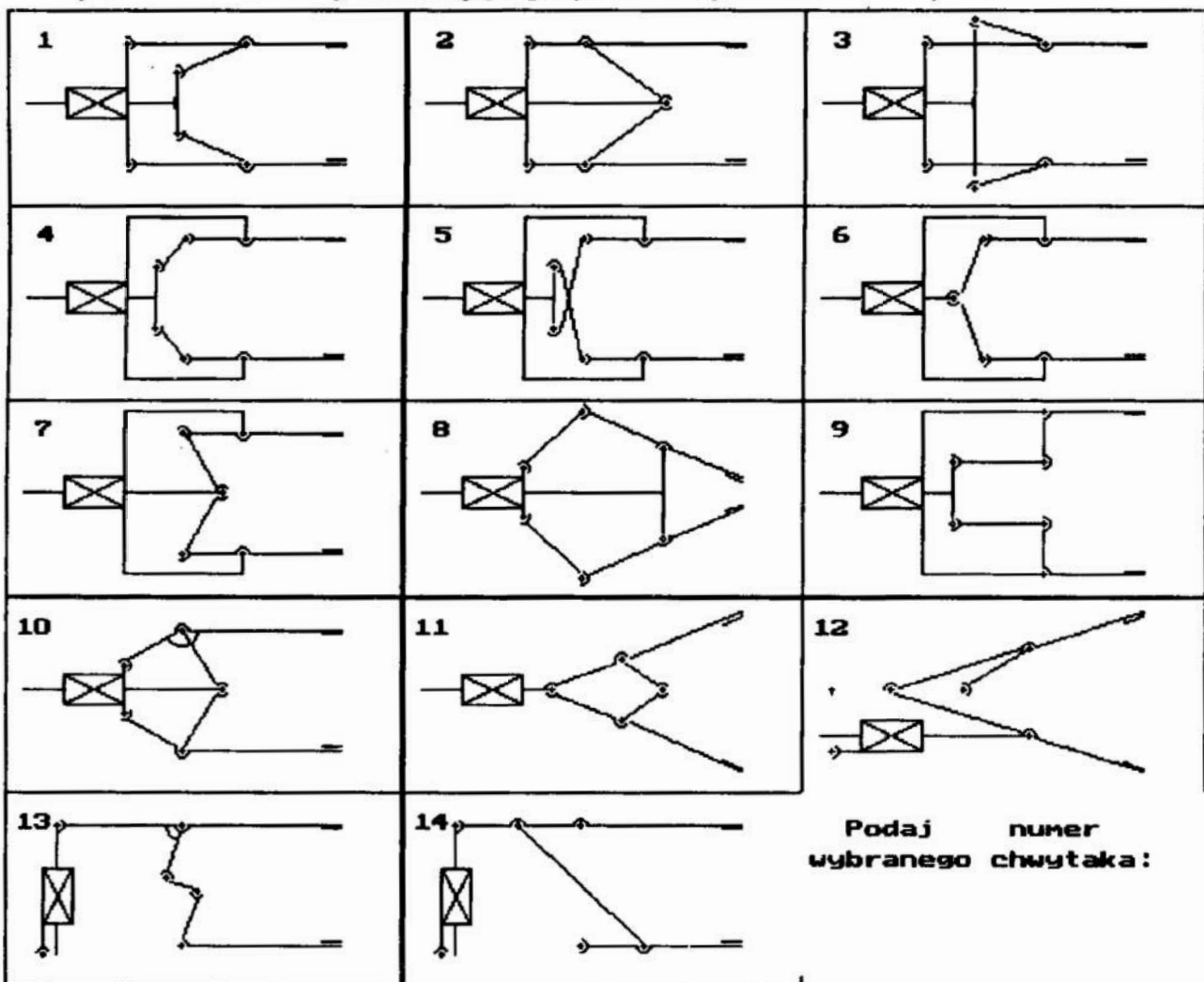
- końcówki chwytne (ich kształt, wymiary, liczba, sposób przemieszczenia itp.),
- układ przeniesienia napędu (wyznaczenie niezbędnej wartości siły chwytu oraz zakresu przemieszczenia końcówek chwytnych),
- zespół napędowy (obliczenie wartości siły na wyjściu zespołu napędowego).

Charakterystyka robota wpływa na:

- parametry układu przeniesienia napędu (obliczenie sił i momentów działających na obiekt przy maksymalnych przyśpieszeniach i najbardziej niekorzystnych warunkach pracy),
- parametry zespołu napędowego (obliczenie niezbędnej wartości siły),
- konstrukcję przyłącza chwytaka.

4. Obliczenie układu przeniesienia napędu

Zaproponowana w pracy [2] klasyfikacja układów przeniesienia napędu w chwytakach robotów przemysłowych wyróżnia mechanizmy zawierające pary kinematyczne tylko V klasy oraz mechanizmy zawierające pary kinematyczne IV i V klasy.



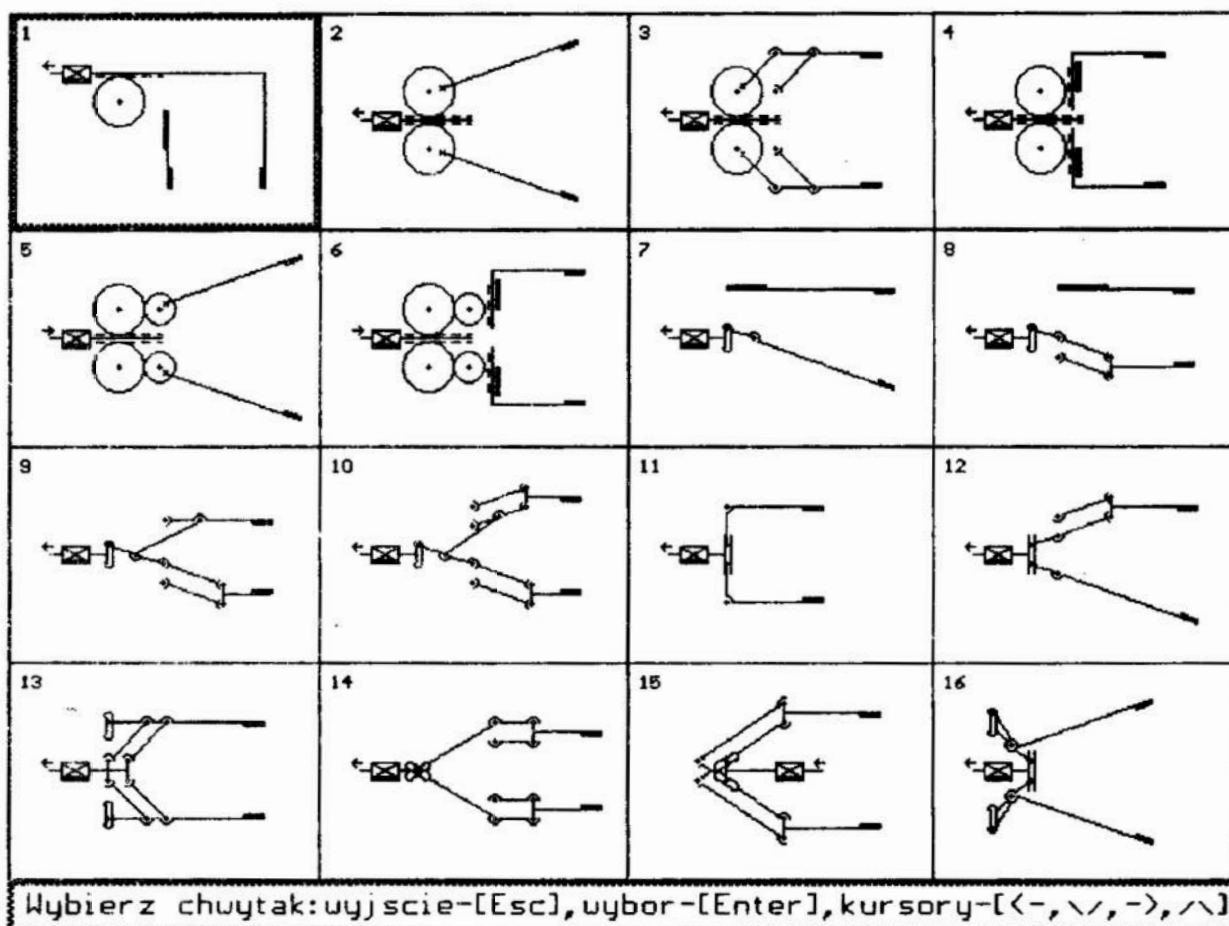
Rys. 2. Katalog chwytaków z mechanizmami zawierającymi pary kinematyczne V klasy
 Fig. 2. Gripper's mechanisms with only 5th class of kinematic pairs

Właściwości chwytaka określa się [1, 3] jego dwiema charakterystykami statycznymi:

- charakterystyką siłową, przedstawiającą zależność siły chwytu F_k od siły zespołu napędowego F_s i parametrów konstrukcyjnych układu przeniesienia napędu chwytaka,
- charakterystyką przesunięciową, przedstawiającą zależność zakresu rozwarcia końcówek chwytanych y od zakresu przemieszczenia x zespołu napędowego i parametrów konstrukcyjnych układu przeniesienia napędu.

Najlepsze właściwości ma taki chwytak, którego charakterystyka siłowa w całym zakresie rozwarcia końcówek osiąga duże wartości $f_F = F_k / F_s$ oraz którego charakterystyka przesunięciowa osiąga duże wartości y/x .

W Instytucie Automatyki Przemysłowej PW opracowano programy, które są komputerowymi katalogami chwytaków zawierających mechanizmy z parami kinematycznymi tylko V klasy (rys.2) oraz z parami kinematycznymi IV i V klasy (rys.3).



Rys. 3. Katalog chwytaków zawierających pary kinematyczne IV i V klasy

Fig. 3. Gripper's mechanisms with 4th and 5th class of kinematic pairs

Te programy komputerowego wspomaganie obliczeń wybranego układu przeniesienia napędu chwytaka umożliwiają [4]:

- wyświetlenie charakterystyk statycznych: siłowej i przesunięciowej (możliwe jest jednocześnie wyświetlanie kilku charakterystyk dla różnych parametrów konstrukcyj-

nych chwytaka, co zdecydowanie ułatwia analizowanie wpływu wybranych parametrów na przebiegi charakterystyk),

- wyświetlanie wartości podstawowych parametrów użytkowych: przełożenia siłowego, przemieszczenia zespołu napędowego oraz zakresu przemieszczenia końcówek chwytanych,
- łatwą zmianę parametrów konstrukcyjnych chwytaka.

5. Optymalizacja parametrów konstrukcyjnych chwytaka

Dobór pożądanych parametrów konstrukcyjnych chwytaka z wykorzystaniem programów komputerowego wspomaganie obliczeń jest uciążliwy i wymaga wielokrotnego wprowadzania danych, analizowania wyników itp., a wynik końcowy uzależniony jest od doświadczenia użytkownika programu.

Na przykład dla mechanizmu chwytaka oznaczonego nr 5 na rys.2 można w wyniku takiej analizy stwierdzić, że:

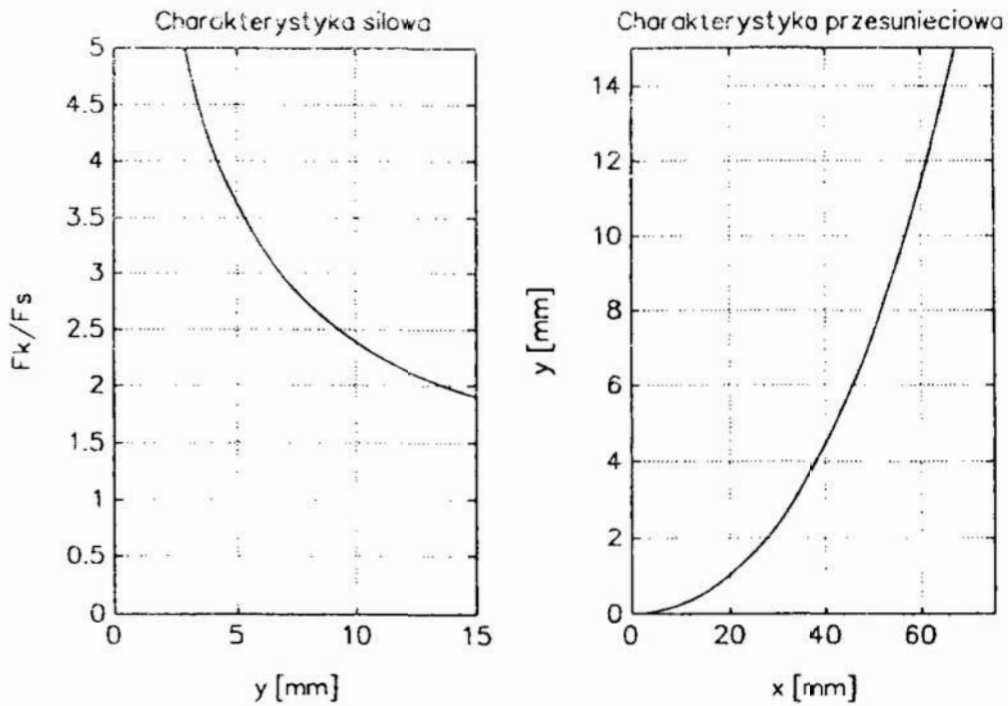
- aby maksymalizować przełożenie siłowe układu przeniesienia napędu należy dobierać małe długości końcówki chwytnej a , natomiast duże wartości parametrów b (drugie ramię końcówki), c (łącznik) i d (rozpora),
- aby minimalizować zakres przemieszczeń zespołu napędowego x należy dobierać możliwie duże wartości ramienia a , natomiast duże wartości parametrów b , c i d .

Przeprowadzono optymalizację pod kątem maksymalizacji przełożenia siłowego układu przeniesienia napędu. Do optymalizacji zastosowano procedury optymalizacji nieliniowej z ograniczeniami z pakietu MATLAB. Założono, że parametry konstrukcyjne mogą się zmieniać w granicach od 0 do 100 mm. Uzyskano optymalne rozwiązanie (rys. 4) dla następujących parametrów: $a = 19.8 \text{ mm}$, $b = 100 \text{ mm}$, $c = 87.7 \text{ mm}$, $d = 100 \text{ mm}$, $e = 12.5 \text{ mm}$. Dla takich wartości parametrów konstrukcyjnych chwytaka $x_{\max} = 67.2 \text{ mm}$, a dla $y = 15 \text{ mm}$ wartość $f_F = 1.9$

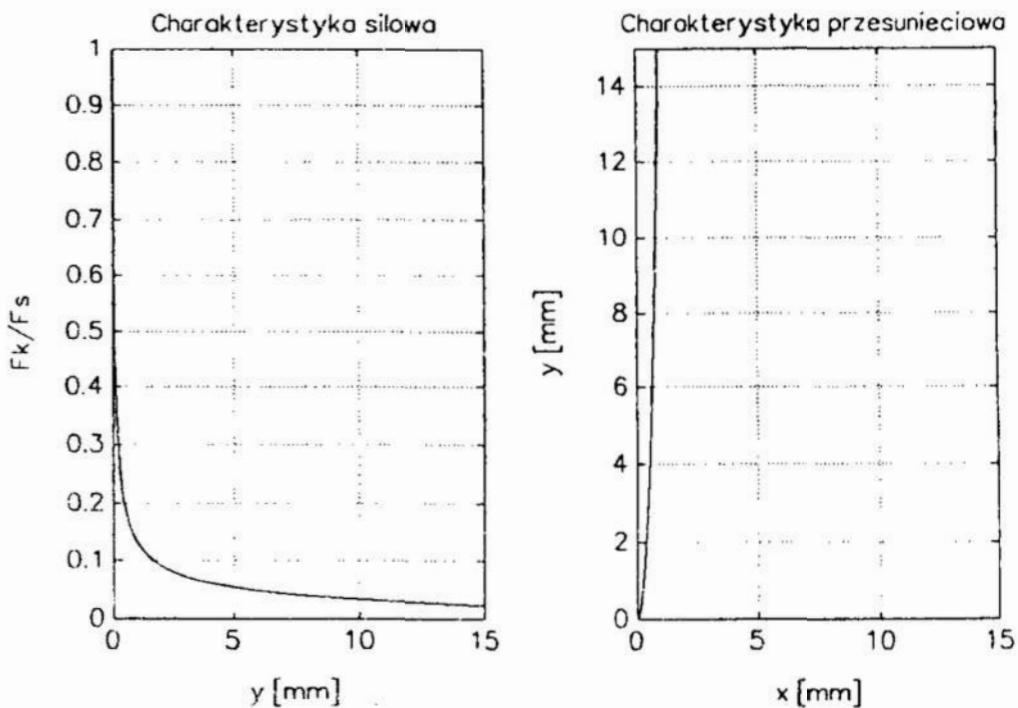
Do optymalizacji pod kątem minimalizacji zakresu przemieszczeń zespołu napędowego zastosowano procedury optymalizacji nieliniowej z ograniczeniami z pakietu MATLAB, zakładając, że wszystkie parametry konstrukcyjne chwytaka mogą się zmieniać w granicach od 0 do 100 mm. Optymalne rozwiązanie przedstawiono na rys. 5, uzyskano je dla następujących parametrów: $a = 100 \text{ mm}$, $b = 7.1 \text{ mm}$, $c = 1.0 \text{ mm}$, $d = 1.0 \text{ mm}$, $e = 1.0 \text{ mm}$. Dla takich wartości parametrów konstrukcyjnych chwytaka $x_{\max} = 0.9 \text{ mm}$, a dla $y = 15 \text{ mm}$ współczynnik $f_F = 0.02$:

Proces optymalizacji parametrów konstrukcyjnych chwytaka równocześnie według obu podanych kryteriów przeprowadzono w następujący sposób:

- 1) ustalono maksymalną wartość przemieszczenia zespołu napędowego $x = 6 \text{ mm}$,
- 2) ustalono minimalną wartość przełożenia siłowego $f_F = 0.2$,
- 3) założono przedział zmienności parametrów konstrukcyjnych chwytaka w granicach od 0 do 100 mm.

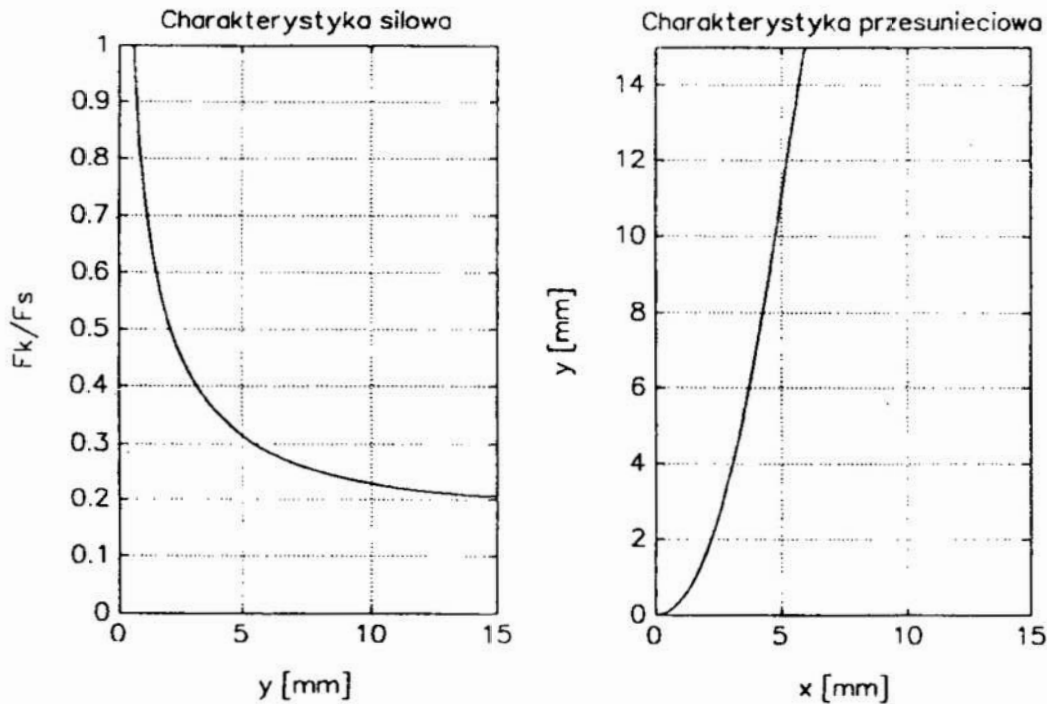


Rys. 4. Charakterystyki statyczne mechanizmu chwytaka zoptymalizowane wg. kryterium maksymalizacji przełożenia siłowego
 Fig. 4. Statistic characteristic of the gripper's mechanism - optimization of force transmission



Rys. 5. Charakterystyki statyczne mechanizmu chwytaka zoptymalizowane wg. kryterium minimalizacji zakresu przemieszczeń zespołu napędowego
 Fig. 5. Statistic characteristic of the gripper's mechanism - optimization of pass transmission

Spośród znalezionych kilkunastu rozwiązań spełniających podane warunki wybrano rozwiązanie dla następujących parametrów konstrukcyjnych: $a = 10 \text{ mm}$, $b = 5 \text{ mm}$, $c = 5.7 \text{ mm}$, $d = 4 \text{ mm}$, $e = 5 \text{ mm}$, przedstawione na rys. 6. Dla takich wartości parametrów konstrukcyjnych chwytaka $x_{\max} = 5.9 \text{ mm}$, a dla $y = 15 \text{ mm}$ współczynnik $f_F = 0.21$.



Rys. 6. Charakterystyki statyczne zoptymalizowanego układu przeniesienia napędu chwytaka

Fig. 6. Statistic characteristic of the gripper's mechanism - optima of propulsion transmission

6. Podsumowanie

Opracowane w Instytucie Automatyki Przemysłowej PW programy komputerowego wspomaganie obliczeń inżynierskich przy projektowaniu chwytaków robotów przemysłowych umożliwiają dla wybranej struktury mechanizmu chwytaka dobór parametrów, zapewniających osiągnięcie żądanych zakresów przemieszczeń końcówek chwytanych i żądanej wartości siły chwytu. Celowe jest stosowanie metod optymalizacyjnych dla wyznaczenia rozwiązania spełniającego różne kryteria konstrukcyjne.

Literatura:

1. Barczyk J.: *Mechanizmy chwytające robotów przemysłowych*. Robotyka , WNT 1986 nr 1, str. 34-38.
2. Barczyk J.: *Klasyfikacja układów przeniesienia napędu w chwytakach robotów przemysłowych*. I Krajowa Konferencja Robotyki. Prace naukowe Instytutu Cybernetyki Technicznej, Nr 67, 1985, str. 3-14.
3. Barczyk J.: *Urządzenia chwytające robotów przemysłowych*. Podstawowe problemy współczesnej techniki. t.XXV Robotyka, PWN 1987, str.269-292.
4. Barczyk J.: *Program wyznaczania parametrów mechanizmu przeniesienia napędu chwytaków z parami kinematycznymi IV i V klasy*. IV Krajowa Konferencja Robotyki, Wrocław 1993, t. 1, str. 55-60.

Streszczenie

W przemysłowych zastosowaniach robotów najczęściej zachodzi konieczność zaprojektowania chwytaka dla konkretnego obiektu manipulacji, tj. wyznaczenia jego podstawowych parametrów użytkowych: wartości siły chwytu oraz zakresu przemieszczenia końcówek chwytanych. W Instytucie Automatyki Przemysłowej PW opracowano programy komputerowego wspomagania obliczeń układu przeniesienia napędu oraz analizy wpływu parametrów konstrukcyjnych chwytaka na jego właściwości chwytne. W referacie przedstawiono problemy optymalizacji parametrów konstrukcyjnych układu przeniesienia napędu chwytaka.

COMPUTER AIDED DESIGN OF PROPULSION TRANSMISSION OF ROBOT'S GRIPPERS

Summary

The paper presents a problems of gripping force creation in robot's grippers with unflexible fingers. Some exemplary structures of propulsion transmission have been presented. Analytical connections between both of gripping force and finger's movement range and the gripper's parameters have been derived. Analysis of influence of the parameters of kinematic diagram of gripper on the statistic characteristics of some propulsion transmission systems have been found out. The problem of multicriterion optimization of gripper's parameters is presented.

rszek BIELECKI, Jerzy WRÓBEL

emysłowy Instytut Maszyn Budowlanych
itechnika Warszawska

ORGANIZACJA BAZ DANYCH W PROCESIE KOMPUTEROWO WSPOMAGANEGO PROJEKTOWANIA SAMOJEZDNYCH ŻURAWI HYDRAULICZNYCH

Wstęp

Podstawową tendencją rozwojową projektowania maszyn jest dążenie do budowy egrowanych systemów komputerowo wspomaganego projektowania umożliwiających ymalne kształtowanie maszyn i ich elementów w sposób uwzględniający wiażące wymagania i przepisy z zakresu bezpieczeństwa, ergonomii i ekologii. wnieź projektowanie samojezdnych żurawi hydraulicznych w oparciu tylko o wiadczenie i intuicję konstruktora jest praktycznie niemożliwe zarówno ze względów rnicznych i ekonomicznych.

Samojezdne żurawie hydrauliczne są jedną z najbardziej złożonych grup maszyn oczych ciężkich, przy projektowaniu których niezbędne jest korzystanie ze specjalnie ygotowanych baz danych zawierających wymagania, przepisy, normy, dyrektywy, fikę, obliczenia inżynierskie oraz symulację komputerową ruchu i zmiany obciążeń.

Projektowanie samojezdnych żurawi hydraulicznych w oparciu o hierarchiczno-sieciowe bazy danych

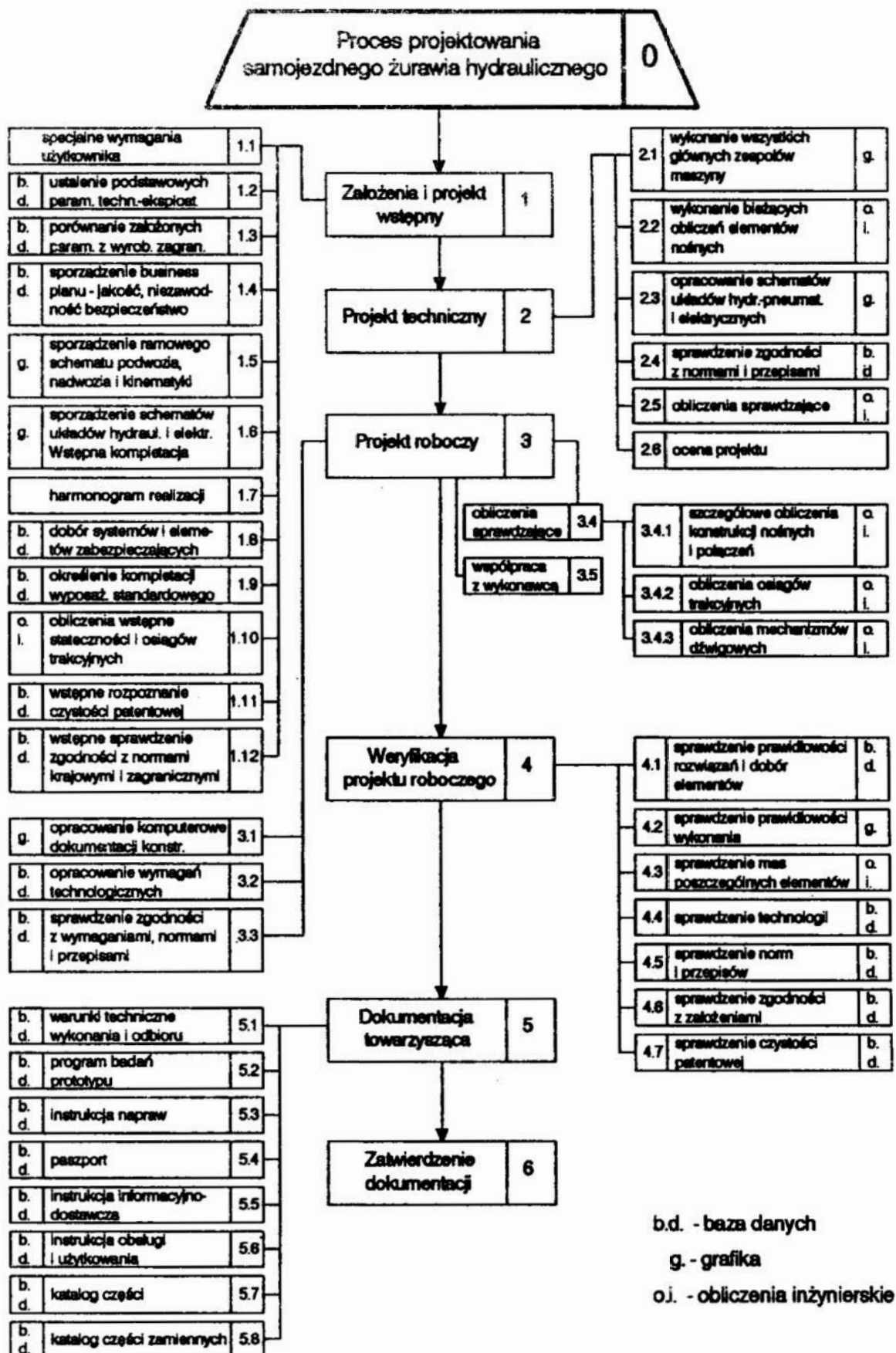
Schemat procesu projektowania samojezdnego żurawia hydraulicznego (rys.1)składa z kilku podstawowych etapów projektowania. Każdy z tych etapów dzieli się z kolei kilka podetapów wymagających korzystania z baz danych.

Jak widać z przedstawionego schematu bazy danych (Data Base)są szczególnie żne w systemach komputerowo wspomaganego projektowania (CAD), gdzie etwarzane są wielkie ilości informacji. Bazy danych są "sercem" każdego systemu.

Bazę danych można zdefiniować jako zbiór wzajemnie powiazanych danych, agazynowanych bez niepotrzebnych powtórzeń i przygotowanych do różnych osowań.

Współczesne bazy danych wyraźnie rozróżniają l o g i c z n ą bazę danych i f i z y n ą bazę danych.

Logiczna baza danych jest modelem logicznym opisującym np. dokumentację rstrukcyjną lub technologiczną maszyny z uwzględnieniem jej specyfiki np. wymagań ie musi spełniać. Baza logiczna informuje programy użytkowe jak pamiętane są dla nagań poszczególne dane. Fizyczna baza danych wskazuje jak dane są fizycznie ieszczone na nośniku informacji.



Rys. 1 Proces projektowania samojazdnego żurawia hydraulicznego
 Fig. 1 Designing process for hydraulic mobile crane

Koncepcja hierarchiczno-sieciowej bazy danych

W logicznej bazie danych występują trzy podstawowe typy modeli:

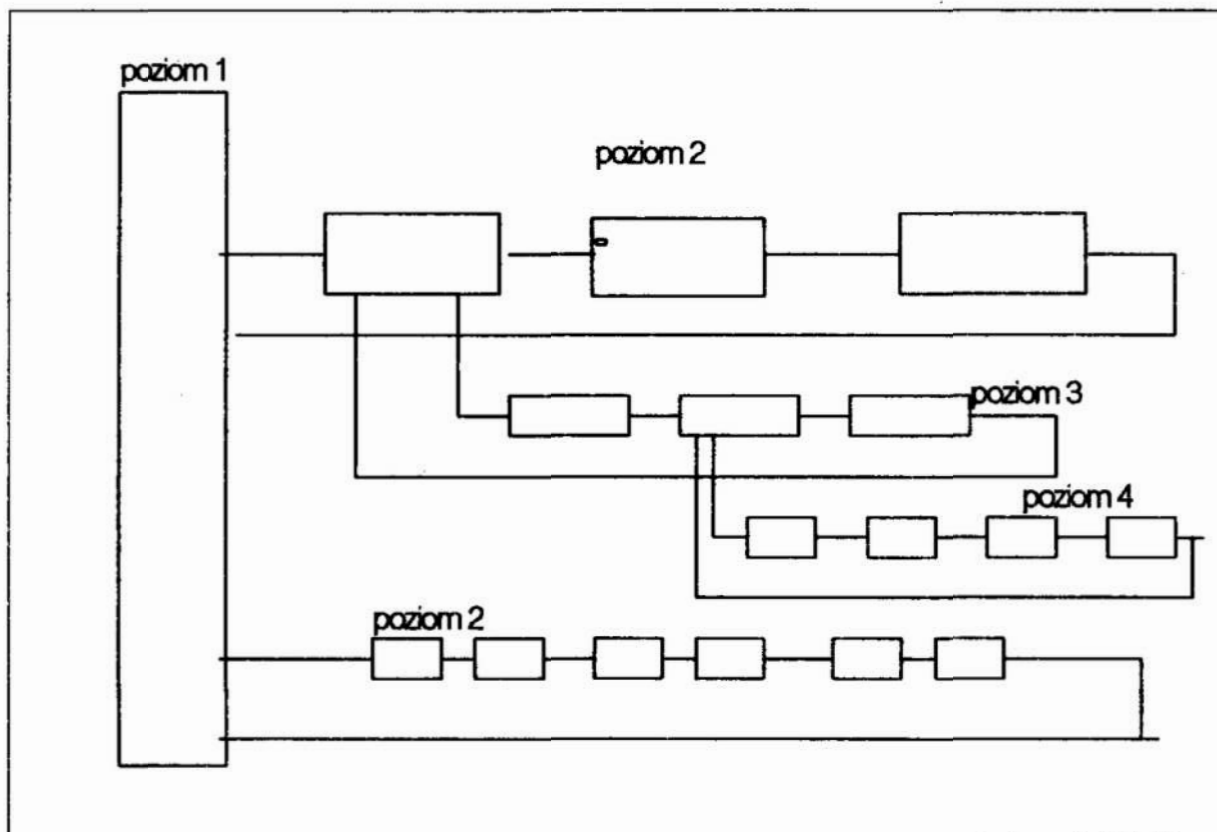
- *model relacyjny*
- *model hierarchiczny*
- *model sieciowy*

Model relacyjny jest najczęściej stosowany w bazach danych. Podstawowym elementem w relacyjnych bazach danych jest relacja, czyli kolekcje danych o zbiorze atrybutów charakteryzujących się identycznymi atrybutami i spełniających ustalone warunki.

Drugim typem modelu stosowanym przy budowie logicznych baz danych jest *model hierarchiczny*. Posiada on strukturę drzewa.

Do budowy systemów CAD stosuje się *modele sieciowe*. W modelu sieciowym każdy element może być uzależniony od szeregu innych. W praktyce stosuje się specjalną organizację sieci, w której elementy tego samego typu należące do jednego właściciela tworzą się w zamknięty pierścień (rys.2).

W ten sposób powstaje hierarchiczno-sieciowy model bazy danych.



Rys.2 Hierarchiczno-sieciowy model bazy danych
(Hierarchic-network model of data base)

Taka struktura jest dość prosta przy organizacji procesu poszukiwań. Posiada zalety modelu hierarchicznego, bez niebezpieczeństwa niespójności, ale wadą jest *znaczna złożoność w manipulowaniu*.

Typowym przykładem takiego modelu jest podział samojezdnego żurawia hydraulicznego na moduły strukturalno-funkcjonalne, podmoduły a następnie części.

Każdemu z tych elementów struktury zostały przypisane odpowiednie powiązania. Jest to więc model hierarchiczno-sieciowy.

Zaletą takiego modelu jest możliwość dynamicznego rozwijania struktury bazy hierarchiczno-sieciowej poprzez dobudowywanie nowych poziomów oraz rozbudowę bądź uzupełnianie istniejących.

Baza danych dla projektowania samojezdnych żurawi hydraulicznych zawierając obowiązujące wymagania krajowe i zagraniczne normy krajowe, ISO CEN, dyrektyw regulaminy ECE, została opracowana na oprogramowaniu działającym w środowisku ACCESS pod WINDOWS 3.1.

Wymagania te są zawarte w rekordach tablicy "wymagania". Każda informacja w rekordzie posiada własny identyfikator (nazwę, nr normy, dyrektywy, przepisy itp). Każde "wymaganie" zawiera opisy w odpowiednich polach rekordu. Opisy te są przechowywane w odpowiednich polach rekordu. Ponadto dla każdego "wymagania" jest zdefiniowany jego charakter dotyczący takich obszarów jak *ekologia*, *ergonomia*, *bezpieczeństwo* - oznaczony odpowiednim kolorem.

4. Podział samojezdnego żurawia hydraulicznego na moduły strukturalno-funkcjonalne

W ramach realizowanego w Przemysłowym Instytucie Maszyn Budowlanych, projektu badawczego p.t. "*Kształtowanie bezpieczeństwa, ergonomii i wskaźników ekologicznych samojezdnych żurawi hydraulicznych w procesie komputerowo wspomaganego projektowania*", został dokonany podział żurawia na moduły strukturalno-funkcjonalne (rys.3).

Podział ten miał na celu zbudowanie hierarchiczno-sieciowego modelu bazy danych. Zgodnie z tym podziałem zostało zbudowane menu ułatwiające zapoznanie się z poszczególnymi wymaganiami. Menu to jest jednocześnie stroną tytułową bazy danych i składa się z opisanych przy pomocy tekstu i piktogramów przycisków sterujących.

Według tego podziału w środowisku "Access" został przygotowany odpowiedni program.

Piktogramy te, zgodnie z przyjętym podziałem, zostały podzielone na dwie grupy:

- podwozie,
- nadwozie - dźwignica

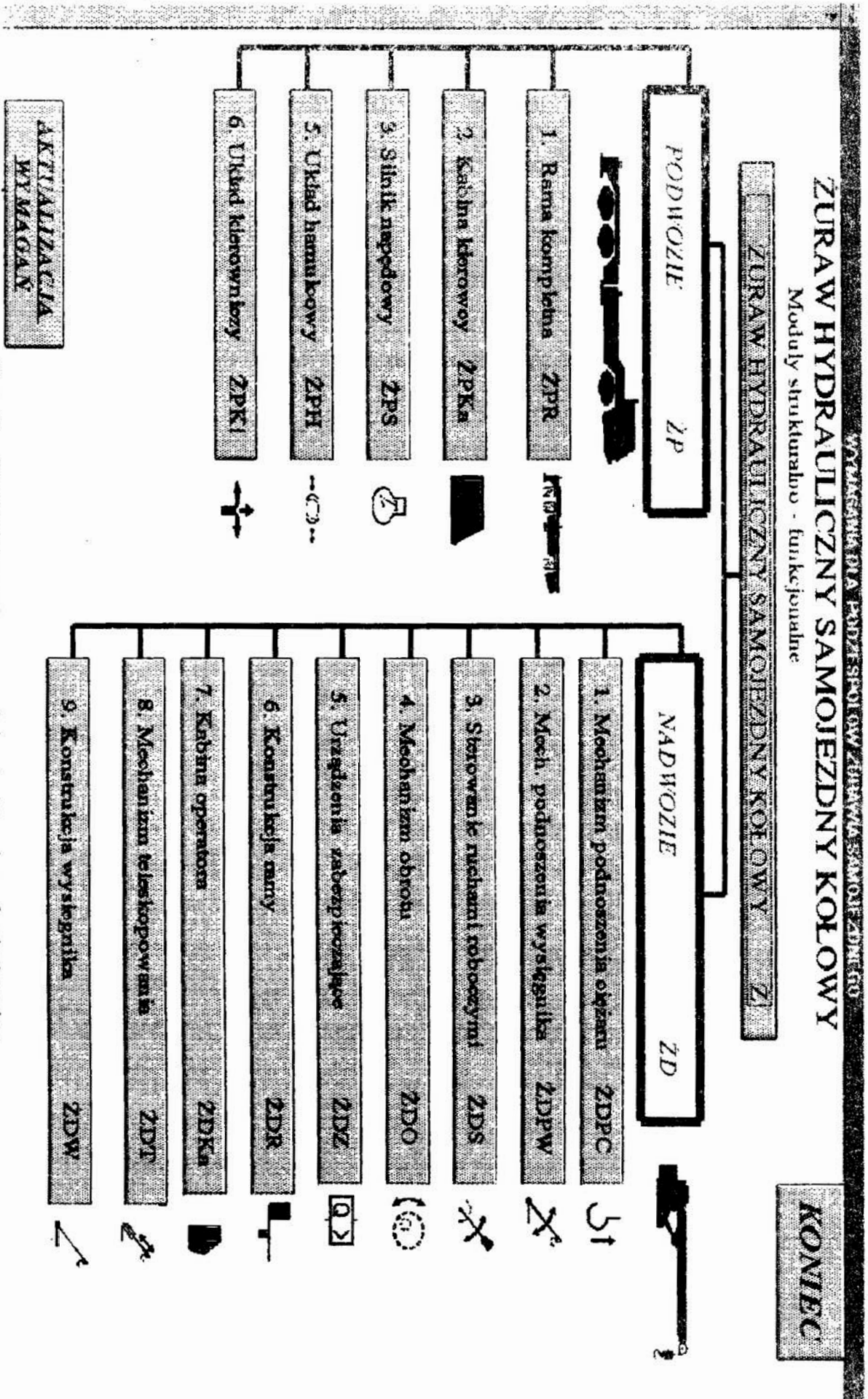
Każdy z przycisków odpowiada początkowo jednemu modułowi i może wywołać własny podmenu dla tego modułu, *podmenu*.

Z kolei przyciskom sterującym w *podmenu*, przypisano rozkazy wywołujące kolejne podmenu zawierające kolejne przyciski sterujące odpowiadające wykazowi konkretnych wymagań dla np. zespołu wchodzącego w skład danego modułu (struktura drzewa). Wymagania te zostały opisane w tablicy "wymagania".

Przykład takiego opisu przedstawia rys.4.

Następnie, w zależności od charakteru wymagań pojawiają się w ramce "Uwaga" odpowiednie kolory:

- zielony dla ekologii,
- fioletowy dla ergonomii,
- czerwony dla bezpieczeństwa



Rys. 3 Podział żurawia na moduły strukturalno - funkcjonalne (Fig. 3 Division of a crane into structural - functional moduli)

ŻPH	UKŁAD HAMULCOWY	WYMAGANIA KRAJOWE I MIĘDZYNARODOWE
SKUTECZNOŚĆ HAMOWANIA- WSZYSTKIE WYMAGANIA DOTYCZĄCE SKUTECZNOŚCI DZIAŁANIA UKŁADÓW HAMULCOWYCH ODNOSZĄ SIĘ DO CAŁEGO POJAZDU, W KTÓRYM ZASTOSOWANO ODPOWIEDNIE UKŁADY HAMULCOWE		
OKREŚLENIE/ OPIS	Reg.ECE ONZ Nr 13 Dokument krajowy-międzynarodowy (europejski) Regulamin ECE ONZ Nr 13 z serią poprawek Nr 06.z dn.22 listopada 1990. Przyjęty do stosowania w Polsce dn.13.11.1992 oraz COUNCIL	UWAGI (pole czerwone)
HAMOWANIE HAMULCEM ZASADNICZYM Wymagania dla badań typu "O" - stan zimny - pojazd obciążony i nieobciążony Silnik odłączony	Skuteczność hamowania dla pojazdów kategorii N powinna zapewnić drogę hamowania nie dłuższą niż i opóźnienie hamowania nie mniejsze niż podane poniżej: pojazdy kat.N1,N2 i N3, prędkość początkowa V[km/h 80,60, droga hamowania Sh[m], 0,15+ (V2/130)*	Badania należy przeprowadzić wg wymagań podanych w Załączniku 4 do Reg. ECE ONZ Nr 13 */ odpowiada opóźnieniu średniemu pełnemu $dm = 5,0 \text{ m/s}^2$

Rys.4 Fragment bazy danych "wymagania"
Fig. 4 Fragment of data base - "requirements"

k zorganizowana hierarchiczno-sieciowa baza danych dla komputerowo pomagane go projektowania samojezdnych żurawi hydraulicznych pozwoli na ich projektowanie z uwzględnieniem wszystkich aktualnie obowiązujących w Polsce magarów; norm, przepisów, regulaminów i dyrektyw EWG i ISO.

Literatura

- Bielecki Z., Wróbel J.: Kształtowanie bezpieczeństwa, ergonomii i ekologii maszyn w procesie komputerowego wspomaganie projektowania. Prace PIMB 2/93.
- Bielecki Z., Machniewski A., Wróbel J.: Problemy kształtowania bezpieczeństwa maszyn w procesie komputerowo wspomaganego projektowania. "Górnictwo odkrywkowe" z.1/94, s.33-39.
- Bielecki Z., Wróbel J.: Design for safety in computer design of all - terrain cranes Zeszyty Naukowe Politechniki Śląskiej z.117 Mechanika s.39-45.
- Damm M., Szuba G.: Badania a kształtowanie cech bezpieczeństwa, ergonomii i ekologii samojezdnego żurawia hydraulicznego. Prace PIMB 2/93.
- Wróbel J., Pokojski J.: The intelligent data bases for the design of mobile cranes in Applications of Artificial Intelligence EPFL Lausanne 1994 r.
- Directive EEC 70/311/EEC Steering System.
- Martin J.: Computer data - base organisation Prentice Hall. Inc. England Cliff. New Jersey 1977
- Osiński Z., Wróbel J.: Wybrane metody komputerowo wspomaganego konstruowania maszyn. PWN Warszawa 1988.
- Pokojski J., Wróbel J.: An integrated design environment. Zeszyty Naukowe Politechniki Śląskiej nr 1161, seria Mechanika, z.108, Gliwice 1992, ss.307 - 314. (International Conference on Computer Integrated Manufacturing)
-] Regulations ECEONZ no 79 Steering System.
-] Standard EN 292-1 Safety of machinery basic concepts, general principles for design 1991.
-] Ullman J. D. : Principles of data bases, Computer Science Press, Rockville 1982.
-] Wróbel J. : Technika komputerowa dla inżynierów. PWN Warszawa 1994.

Streszczenie

W pracy przedstawiono podstawowe problemy organizacji baz danych dla potrzeb komputerowo wspomaganego projektowania samojezdnych żurawi hydraulicznych.

Zaproponowano podział żurawia na moduły strukturalno-funkcjonalne z których bezpośrednio wynika zasadniczy schemat hierarchiczno-sieciowej struktury bazy.

HIERARCHICAL - NETWORK DATA BASES FOR COMPUTER - AIDED DESIGN OF HYDRAULIC MOBILE CRANES.

Summary

Basic problems regarding organization of computer data bases for computer-aided design of hydraulic mobile cranes are discussed in the paper. It is proposed to divide a crane into structural and functional modules from which basic scheme of logical hierarchical-network data base structure results directly.

ZASTOSOWANIE METOD OPTYMALIZACJI W OKREŚLANIU SKŁADU POLIMEROWYCH KOMPOZYTÓW O ŻĄDANYCH WŁASNOŚCIACH UŻYTKOWYCH

1. Wstęp

Coraz częściej w technice można spotkać sytuacje, że dla określonego rozwiązania konstrukcyjnego opracowuje się specjalne materiały o projektowanych własnościach i ściśle określonych właściwościach użytkowych. Stosunkowo szerokie wykorzystanie znalazły w tym zakresie kompozyty polimerowe. Wynika to z dużej skuteczności wpływu modyfikacji polimerów - zwłaszcza modyfikacji fizycznej - na ich własności i właściwości. W wyniku fizycznej modyfikacji, przez wypełnianie polimerów, otrzymuje się kompozyty - mieszaniny łączące w sobie cechy poszczególnych składników. Właściwości kompozytów w istotny sposób zależą od rodzaju wypełniaczy, ich postaci, wielkości cząsteczek oraz ich ilościowego udziału. Kształtowanie właściwości kompozytów nie jest jednak zadaniem łatwym, gdyż nie wynikają one z superpozycji cech użytych składników. Przyczyną tego jest między innymi wzajemne oddziaływanie wypełniaczy samych na siebie oraz łączne - na polimerową matrycę kompozytu. Dotychczas, nie opracowano dostatecznie dokładnych metod teoretycznych, pozwalających określić właściwości kompozytów na podstawie własności jego komponentów. Wpływ rodzaju wypełniaczy i ich ilościowego udziału na właściwości kompozytów można określić jedynie doświadczalnie. W rozwiązaniu tego problemu bardzo pomocne są: teoria planowania eksperymentu [3, 4] i metody optymalizacji [2, 5].

2. Zagadnienia związane z doświadczalnymi badaniami kompozytów

Optymalizacja matematyczna wymaga znajomości modelu matematycznego. Jeśli nie ma przesłanek teoretycznych do jego utworzenia, wtedy wykorzystuje się metody doświadczalne. Optymalizację - na podstawie badań doświadczalnych - umożliwiają procedury optymalizacyjne tzw. KRUPPO. Największe znaczenie ma procedura "R" (od ang. research - badania) [4]. Określa ona sytuację typową dla większości badań, w których znane są: cel badań, kryteria optymalizacji, warunki ograniczające narzucone na wielkości charakteryzujące obiekt badań: zmienne decyzyjne (skład kompozytu) i wyjścia (właściwości kompozytu). Nie są znane natomiast funkcje wiążące te wielkości (funkcje obiektu badań), które wyznacza się w toku badań doświadczalnych.

Jakościowy model obiektu badań

Pierwszy etap badań wiąże się z opracowaniem składów kompozytów, które będą poddane badaniom. Doboru rodzaju wypełniaczy i zakresów ich ilościowych udziałów w tworzonych kompozytach, dokonuje się na podstawie rozeznania literaturowego, wstępnych badań rozpoznawczych, czy też wykorzystując wiedzę i doświadczenie, a niekiedy także intuicję badacza. Stanowią one wielkości wejściowe lub wielkości stałe w badaniach doświadczalnych. Wielkościami wyjściowymi są natomiast te właściwości badanych kompozytów, które decydują o ich przydatności do danych zastosowań.

Wybór postaci funkcji aproksymującej i planu eksperymentu

Postać funkcji obiektu badań zwykle łączy się z wyborem planu eksperymentu. Z reguły jest to wielomian algebraiczny, który też najczęściej zadowalająco aproksymuje wyniki badań doświadczalnych. Niekiedy jednak zachodzi konieczność przyjęcia innej

postaci funkcji, wtedy należy ją poszukiwać wśród różnych klas funkcji i przyjęc tę, która spełnia odpowiednie testy statystyczne i ma najlepsze oceny statystyczne. Natomiast planowanie badań, rozumiane jako świadomy wybór serii doświadczeń (macierzy wejść eksperymentu), może mieć decydujące znaczenie z punktu widzenia powodzenia całości podejmowanych działań badawczych, gdyż jakość identyfikacji badanego obiektu w istotny sposób może zależeć od doboru ilości doświadczeń i od wartości poszczególnych zmiennych w kolejnych doświadczeniach [4].

Do badań właściwości mieszanin zależnych od ich składu, najbardziej efektywne jest planowanie sympleksowe [3, 4]. W planowaniu tym wykorzystuje się to, że skład kompozytu można charakteryzować za pomocą wektora zmiennych x_1, x_2, \dots, x_S składników kompozytu, spełniających ograniczenie bilansowe określone wzorem:

$$\sum_{s=1}^S x_s = 100, \quad \text{gdzie: } x_s - \text{procentowy udział } s\text{-tego składnika} \quad (1)$$

lub dla wartości unormowanych (standaryzowanych):

$$\sum_{s=1}^S t_s = 1, \quad \text{gdzie: } t_s = \frac{x_s}{100}, \quad (2)$$

w których: $x_s \geq 0$, przy czym $s = 1, 2, \dots, S$, S - jest liczbą składników kompozytu.

Z powyższego wynika, że spośród S zmiennych wejściowych tylko $(S-1)$ są niezależne liniowo. Pozwala to przyjmować do opisu wielomiany zredukowane o mniejszej liczbie współczynników. Zmniejsza się więc wymagana liczba doświadczeń - kompozytów - do wyznaczenia współczynników takiego wielomianu. Przy dużej wymiarowości S oraz przy wyższych stopniach R wielomiany zredukowane mają jeszcze zbyt dużą liczbę współczynników, dlatego często stosuje się uproszczone wielomiany zredukowane [3].

Spośród dwóch podstawowych typów planów sympleksowych: całkowitych planów typu $\{S, R\}$, umożliwiających identyfikowanie wielomianów zredukowanych stopnia R o S zmiennych (przy $R \leq S$) i ułamkowych planów typu $\{S\}$, umożliwiających identyfikowanie uproszczonych wielomianów zredukowanych stopnia S o S zmiennych, większe znaczenie mają te ostatnie. Pozwalają bowiem one lepiej określić badane właściwości wewnątrz sympleksu, umożliwiając przy tym identyfikowanie uproszczonych wielomianów zredukowanych, mających mniejszą liczbę współczynników (tym samym mniejszą liczbę wymaganych doświadczeń N), określoną wzorem:

$$N = 2^S - 1 \quad (3)$$

Plany ułamkowe stanowią zbiór wszystkich punktów x_1, x_2, \dots, x_S określonych wzorami:

$$(1, 0, \dots, 0); \left(\frac{1}{2}, \frac{1}{2}, 0, \dots, 0\right); \dots, \left(\frac{1}{S}, \frac{1}{S}, \dots, \frac{1}{S}\right) \quad (4)$$

oraz punktów otrzymanych przez przestawianie współrzędnych. Przestrzeń czynnikowa dla S składników kompozytu jest więc sympleksem o S wierzchołkach na $(S-1)$ wymiarowej hiperpłaszczyźnie.

Badania wpływu poszczególnych składników na właściwości kompozytu prowadzi się najczęściej tylko w pewnym zakresie ich ilościowego udziału. Wykorzystuje się wówczas planowanie lokalne, czyli badanie podobszaru sympleksu [3]. Plan lokalny otrzymuje się przez transformację planu całkowitego lub ułamkowego planu sympleksowego. Sprowadza się to do przyporządkowania wierzchołkom całkowitego sympleksu:

$$\mathbf{x}_1 = [1, 0, \dots, 0]^T, \quad \mathbf{x}_2 = [0, 1, \dots, 0]^T, \quad \mathbf{x}_S = [0, 0, \dots, 1]^T \quad (5)$$

odpowiednio S wierzchołków: $\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_S$, ograniczających podobszar sympleksu, określony wektorami:

$$\mathbf{w}_s = [w_{s1}, w_{s2}, \dots, w_{sS}]^T, \quad \text{gdzie: } s = 1, 2, \dots, S. \quad (6)$$

Wprowadzając oznaczenia macierzowe, można plan teoretyczny rozpięty na całkowitym sympleksie (5) przedstawić jako macierz wejść X , a wierzchołki podobszaru sympleksu (6) jako macierz transformacji L :

$$X = \begin{bmatrix} x_1^T \\ x_2^T \\ \cdot \\ x_S^T \\ \cdot \\ x_N^T \end{bmatrix} \quad (7) \quad , \quad L = \begin{bmatrix} w_1^T \\ w_2^T \\ \cdot \\ w_S^T \end{bmatrix} \quad (8)$$

Macierz wejść W planu lokalnego wyznacza się przez transformację plan X za pomocą macierzy L , według zależności:

$$W = XL \quad (9)$$

Współczynniki wielowymiarowej funkcji regresji mierzonych wielkości, o postaci uproszczonego wielomianu zredukowanego S -tego stopnia o S zmiennych, oblicza się znaną metodą najmniejszych kwadratów, według zależności [3]:

$$b = (X^T X)^{-1} X^T y \quad (10)$$

gdzie: X - macierz eksperymentu,
 y - wektor wyjść eksperymentu (mierzonej wielkości),
 b - wektor współczynników funkcji regresji.

2.3. Optymalizacja i polioptymalizacja

W przypadkach, gdy znana jest funkcja optymalizacji, jest jedno ściśle określone kryterium oceny i liczba zmiennych decyzyjnych nie jest duża (np. $S \leq 3$), wówczas rozwiązanie zadania optymalizacyjnego jest stosunkowo łatwe. Stan optymalny można określić wprost, na podstawie przebiegu tej funkcji - analitycznie lub wykreślnie. Natomiast, gdy liczba zmiennych decyzyjnych jest większa od trzech, a kryterium oceny nie jest jedno, wówczas wręcz konieczne jest korzystanie z komputerowych metod optymalizacji lub polioptymalizacji. Niezależnie od stosowanego algorytmu obliczeń, wymaga się formalnego sprecyzowania i opisu zadania optymalizacji, które można sformułować przy pomocy trzech zbiorów [5]:

- A - zbiór wszystkich możliwych wariantów rozwiązań,
- Ω - zbiór ograniczeń narzuconych na wielkości charakteryzujące obiekt badań,
- Φ - przyjęty system kryteriów oceny poszukiwanego rozwiązania optymalnego.

Zbiór wszystkich możliwych wariantów rozwiązań, w naszym przypadku określa model matematyczny obiektu badań w postaci wyznaczonego na podstawie wyników badań i zweryfikowanego statystycznie zbioru funkcji regresji, które można zapisać:

$$Y_v = y_v = F(x_{1v}, x_{2v}, \dots, x_{sv}, z_{1v}, z_{2v}, \dots, z_{tv}) \quad (11)$$

gdzie:

Y_v - właściwość obiektu badań, jako v -ta wielkość wyjściowa mierzona w badaniach,

y_v - funkcja regresji v -tej wielkości wyjściowej,

$x_{1v}, x_{2v}, \dots, x_{sv}$ - wielkości wejściowe (zmiennie decyzyjne), których wpływ na v -tą wielkość wyjściową badano doświadczalnie,

$z_{1v}, z_{2v}, \dots, z_{tv}$ - wielkości stałe, które celowo ustalono na pewnym poziomie wartości lub na które nie mamy wpływu - są niesterowalne.

Wymagania narzucone na obiekt badań stanowią ograniczenia, które mogą być narzucone na wartości [8]:

- zmiennych decyzyjnych (np. skład kompozytu) $x_s : s = 1, 2, \dots, S$,
- zmiennych stanu (np. warunki eksploatacji) $z_t : t = 1, 2, \dots, T$,
- właściwości obiektu (np. wartość zużycia) $y_w : w = 1, 2, \dots, W$.

Ograniczenia mogą mieć charakter nierówności (tzw. ograniczenia słabe) lub mogą to być ograniczenia równościowe (tzw. ograniczenia mocne).

Kryterium optymalizacji - w przypadku procesu optymalizacyjnego wyników badań doświadczalnych, zwykle jest funkcją tożsamą z funkcją obiektu badań (11) - funkcją regresji rozpatrywanej wielkości wyjściowej:

$$F \equiv y_v \quad (12)$$

lub też inną funkcją, której argumenty są również zmiennymi decyzyjnymi (np. koszty wykonania kompozytów zależne od ich składu) i wówczas staje się ona kryterium optymalizacji :

$$F = F(x_1, x_2, \dots, x_S) \quad (13)$$

Najczęściej nie można zastosować tylko jednego kryterium, wówczas powstaje zagadnienie polioptymalizacji. Należy wtedy sformułować jednoznacznie kryterium nadrzędne F a kryteria oceny K_i - nazywane cząstkowymi, powinny być podporządkowane kryterium nadrzędnemu [5]. W omawianej problematyce optymalizacji wyników badań doświadczalnych, można ograniczyć się do modelu funkcji liniowej, tzw. strategii kompensacyjnej:

$$F = \sum_{i=1}^I w_i \cdot k_i \quad (14)$$

gdzie: w_i - są współczynnikami wagi i -tego kryterium, k_i - wartość i -tego kryterium,

Podaną strategię, o addytywnej postaci funkcji celu F przyjmuje się zwykle, gdy nie ma wskazówek co do jej innej postaci. Funkcja ta ma tę właściwość, że niedostatek jednej właściwości k_i może być skompensowany wzrostem innej właściwości k_j .

3. Przykład

Celem badań w podanym przykładzie była optymalizacja składu samosmarnych kompozytów ślizgowych, tworzonych na podstawie polioksymetylenu (POM), ze względu na ich najkorzystniejsze właściwości tribologiczne. Na podstawie rozeznania literaturowego i własnych doświadczeń badawczych przyjęto następujące składniki i ich maksymalne udziały objętościowe w tworzonych kompozytach:

- x_1 : proszek POM (Tarnoform 300) - jako matryca kompozytów,
- x_2 : proszek PTFE (Tarflen M-500), maksymalny udział do 36% obj.,
- x_3 : proszek brązu ołowiowego, maksymalny udział do 60% obj.,
- x_4 : włókno szklane cięte o długości 3 mm z mikrokulkami szklanymi w stosunku wagowym 50% włókna i 50% kulek, maksymalny udział łączny do 24% obj.

Badany podobszar wpływu podanych składników na właściwości tworzonych kompozytów określono czterema wierzchołkami, według wzoru (6):

$$\begin{aligned} w_1 &= [100 \ 0 \ 0 \ 0] , & w_2 &= [64 \ 36 \ 0 \ 0] , \\ w_3 &= [40 \ 0 \ 60 \ 0] , & w_4 &= [76 \ 0 \ 0 \ 24] . \end{aligned} \quad (15)$$

Zgodnie z podanymi zależnościami (7) i (8), macierz wejść X oraz macierz transformacji L , dla ułamkowego planu sympleksowego typu {4}, przyjmą wartości:

$$X = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0,5 & 0,5 & 0 & 0 \\ 0,5 & 0 & 0,5 & 0 \\ 0,5 & 0 & 0 & 0,5 \\ 0 & 0,5 & 0,5 & 0 \\ 0 & 0,5 & 0 & 0,5 \\ 0 & 0 & 0,5 & 0,5 \\ 0,33 & 0,33 & 0,33 & 0 \\ 0,33 & 0,33 & 0 & 0,33 \\ 0,33 & 0 & 0,33 & 0,33 \\ 0 & 0,33 & 0,33 & 0,33 \\ 0,25 & 0,25 & 0,25 & 0,25 \end{bmatrix} \quad (16)$$

$$L = \begin{bmatrix} 100 & 0 & 0 & 0 \\ 64 & 36 & 0 & 0 \\ 40 & 0 & 60 & 0 \\ 76 & 0 & 0 & 24 \end{bmatrix} \quad (17)$$

Transformując następnie plan X (16) za pomocą macierzy L (17), zgodnie z zależnością (9), otrzymuje się macierz wejść W (18), przedstawiającą lokalny ułamkowy plan sympleksowy, przeznaczony do badania czynnikowego podobszaru sympleksu ograniczonego wierzchołkami w_1, w_2, w_3, w_4 (15). Wielkości wejściowe do badań przedstawiono także w innej formie - w tab. 1

T a b e l a 1

$$W = \begin{bmatrix} 100 & 0 & 0 & 0 \\ 64 & 36 & 0 & 0 \\ 40 & 0 & 60 & 0 \\ 76 & 0 & 0 & 24 \\ 82 & 18 & 0 & 0 \\ 70 & 0 & 30 & 0 \\ 88 & 0 & 0 & 12 \\ 52 & 18 & 30 & 0 \\ 70 & 18 & 0 & 12 \\ 58 & 0 & 30 & 12 \\ 68 & 12 & 20 & 0 \\ 80 & 12 & 0 & 8 \\ 72 & 0 & 20 & 8 \\ 60 & 12 & 20 & 8 \\ 70 & 9 & 15 & 6 \end{bmatrix} \quad (18)$$

Kompozyty przeznaczone do badań

Nr kol.	Skład kompozytu (udziały objętościowe)
1	100% POM
2	64% POM + 36% PTFE
3	40% POM + 60% brązu
4	76% POM + 24% szkła
5	82% POM + 18% PTFE
6	70% POM + 30% brązu
7	88% POM + 12% szkła
8	52% POM + 18% PTFE + 30% brązu
9	70% POM + 18% PTFE + 12% szkła
10	58% POM + 30% brązu + 12% szkła
11	68% POM + 12% PTFE + 20% brązu
12	80% POM + 12% PTFE + 8% szkła
13	72% POM + 20% brązu + 8% szkła
14	60% POM + 12% PTFE + 20% brązu + 8% szkła
15	70% POM + 9% PTFE + 15% brązu + 6% szkła

Wielkościami wyjściowymi w badaniach i zarazem kryteriami oceny właściwości tribologicznych były: intensywność zużycia I_h oraz wartość współczynnika tarcia μ , które powinny osiągać możliwie najmniejsze wartości. Jako funkcje regresji mierzonych wielkości przyjęto uproszczone zredukowane wielomiany 4 stopnia dla 4 zmiennych [3], o następującej postaci:

$$\hat{y} = b_1 x_1 + b_2 x_2 + b_3 x_3 + b_4 x_4 + b_5 x_1 x_2 + b_6 x_1 x_3 + b_7 x_1 x_4 + b_8 x_2 x_3 + b_9 x_2 x_4 + b_{10} x_3 x_4 + b_{11} x_1 x_2 x_3 + b_{12} x_1 x_2 x_4 + b_{13} x_2 x_3 x_4 + b_{14} x_1 x_2 x_3 x_4, \quad (19)$$

Współczynniki funkcji b_1, b_2, \dots, b_{14} obliczono metodą najmniejszych kwadratów, według wcześniej podanej zależności (10). Wyniki badań oraz wyznaczone współczynniki funkcji regresji wraz z testem ich istotności, zestawiono w tab.2.

Tabela 2

Wyniki badań i współczynniki funkcji regresji oraz ich analiza statystyczna

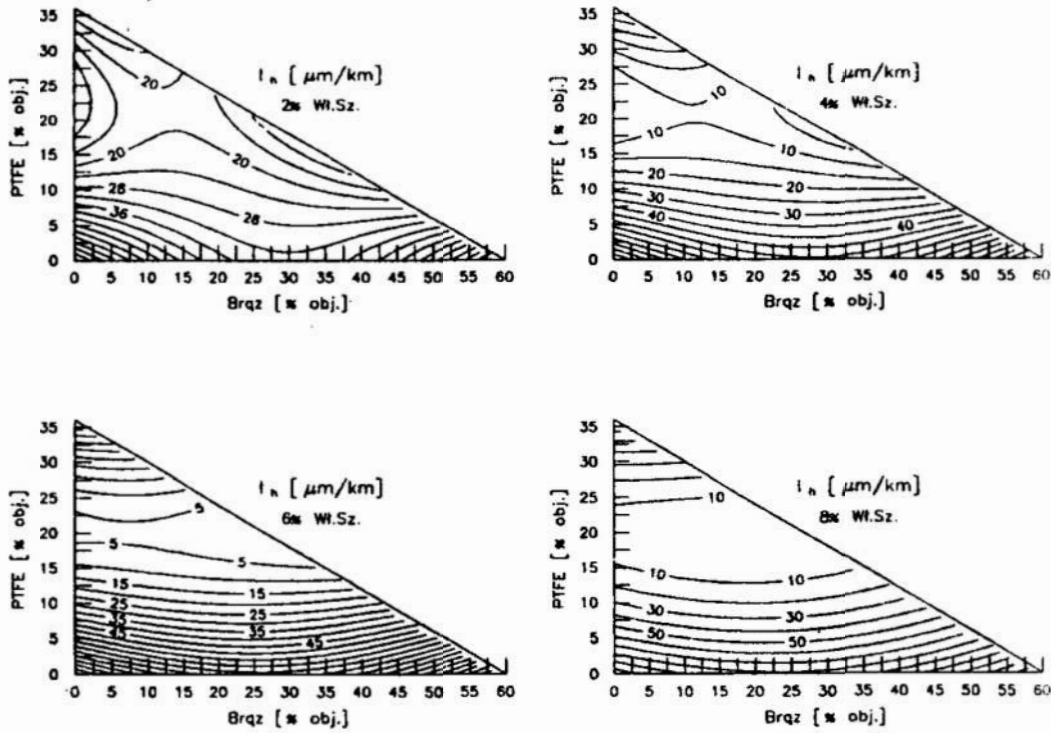
Nr kol.	WYNIKI BADAŃ $p=5 \text{ MPa}, v=1 \text{ m/s}$		WSPÓŁCZYNNIKI FUNKCJI REGRESJI oraz test t - Studenta istotności współczynników				
	I_h [$\mu\text{m/km}$]	μ [-]	Oznac. Wspól.	I_h [$\mu\text{m/km}$]		μ [-]	
				b_k	t_k	b_k	t_k
1	60,7	0,252	b_1	0,606289	79,253	0,002521	264,292
2	24,7	0,165	b_2	4,441848	23,409	0,00890	37,473
3	48,7	0,313	b_3	2,200302	49,777	0,006956	126,071
4	155,0	0,522	b_4	-4,145902	8,844	-0,037204	63,652
5	18,3	0,164	b_5	-0,075517	26,132	-0,000137	38,009
6	14,0	0,206	b_6	-0,044850	45,970	-0,000085	70,174
7	124,0	0,484	b_7	0,114233	18,712	0,000671	88,117
8	20,0	0,347	b_8	-0,345067	25,571	0,000830	47,181
9	14,7	0,182	b_9	1,335758	19,841	0,0030	33,111
10	113,0	0,345	b_{10}	0,167619	36,122	0,000328	56,713
11	31,3	0,191	b_{11}	0,006033	24,937	-0,000012	38,312
12	20,7	0,217	b_{12}	-0,022976	24,566	-0,000049	39,138
13	83,0	0,338	b_{13}	0,0	---	-0,000261	28,086
14	12,0	0,226	b_{14}	-0,000535	46,384	3,695E-6	24,789
15	25,0	0,224	$t_{kr} = 4,303$ $v = N - N_b = 2$		$t_{kr} = 12,706$ $v = N - N_b = 1$		
Odchylenie standardowe σ				0,280		0,00025	
Wsp. korelacji wielowymiarowej R				0,9999		0,9999	

Przykład graficznej interpretacji podanych zależności - jako wykresy warstwiczne - pokazano na rys.1 i 2, przy czym udział jednego ze składników (wypełniacz szklany) przyjmuje wartości dyskretne: 2, 4, 6 i 8 % objętościowo. Wybór optymalnego składu kompozytu wieloskładnikowego, już przy dwóch kryteriach jest zadaniem stosunkowo trudnym, co ilustrują przebiegi wartości I_h oraz μ . W tej sytuacji pozostaje przyjęcie rozwiązania kompromisowego. Stosując formalne procedury optymalizacji i technikę komputerową do realizacji obliczeń, możliwe jest szybkie określenie zbioru rozwiązań dopuszczalnych i wybór rozwiązania najlepiej spełniającego kryteria oceny.

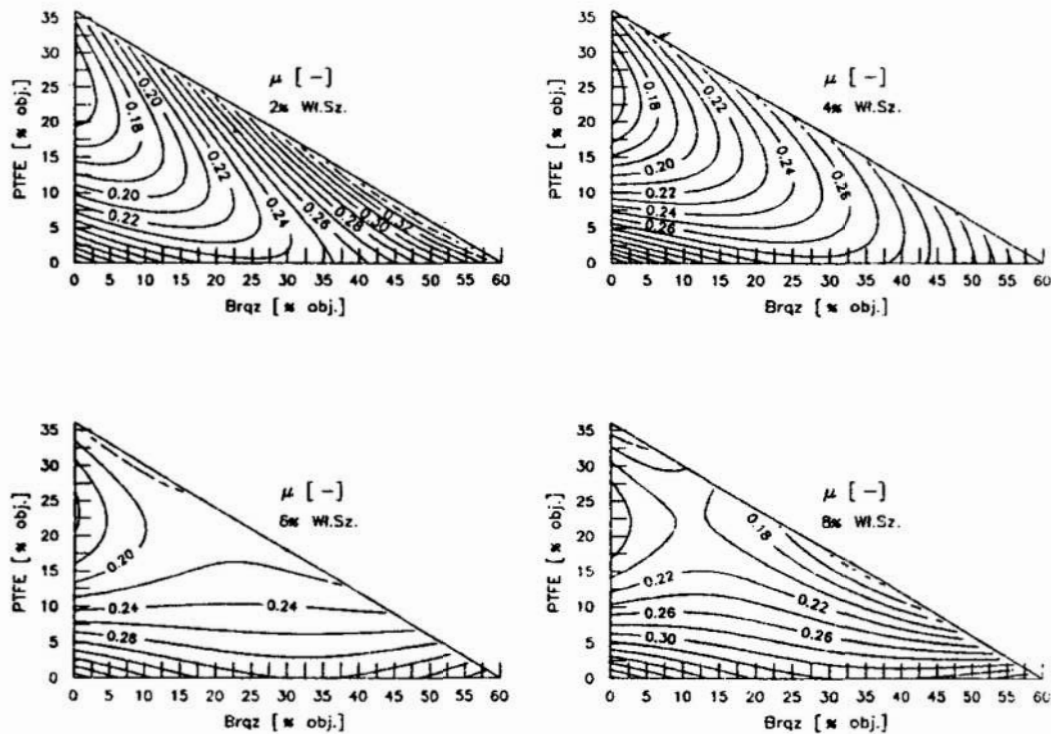
Do obliczeń wykorzystano program "POLIOPT" [1], przeznaczony do optymalizacji i polioptymalizacji dwóch ciągłych i ograniczonych funkcji kryterialnych wielu zmiennych. W przypadku większej liczby kryteriów, do polioptymalizacji wybiera się dwa najważniejsze kryteria, a pozostałe zamienia na ograniczenia nierównościowe lub równościowe. Polioptymalizacja dwóch funkcji kryterialnych (kryt[1] oraz kryt[2]) realizowana jest jako minimalizacja funkcji celu o postaci [5]:

$$F = (1 - wag) * \text{kryt}[1] + wag * \text{kryt}[2] \quad (20)$$

gdzie, kryteriami polioptymalizacji są: kryt[1]=Min(I_h) i kryt[2]=Min(μ), z ograniczeniami nierównościowymi i równościowymi, nakładanymi w zasadzie tylko na argumenty funkcji: $I_h, \mu = f(x_1, x_2, x_3, x_4)$, określającymi dopuszczalne składniki kompozytów.



Rys. 1. Wpływ ilości proszku PTFE, proszku brązu i włókna szklanego na intensywność zużycia I_h kompozytów POM podczas tarcia po stali; $p=5$ MPa, $v=1$ m/s
 Fig. 1. Influence of PTFE powder, bronze powder and glass fibre content on the linear wear intensity I_h for POM composites under friction on steel; $p=5$ MPa, $v=1$ m/s



Rys. 2. Wpływ ilości proszku PTFE, proszku brązu i włókna szklanego na wartość współczynnika tarcia μ kompozytów POM podczas tarcia po stali; $p=5$ MPa, $v=1$ m/s
 Fig. 2. Influence of PTFE powder, bronze powder and glass fibre content on the value of the friction coefficient μ of the POM composites under friction on steel; $p=5$ MPa, $v=1$ m/s

Ograniczenia uwzględniane są metodą zewnętrznej funkcji kary [5], dodawanej do funkcji celu F . Najlepsze rozwiązanie uzyskuje się, gdy współczynniki kary są możliwie małe. W przypadku ograniczeń stale spełnianych należy próbować zmniejszać wartości współczynników kary, aż do uzyskania rozwiązania satysfakcjonującego. W tab.3 zestawiono wyniki obliczeń optymalizacyjnych, w którym kryteria I_h i μ , określone są funkcjami regresji podanymi w tab.2. Obliczenia realizowano metodą DFP (Davidona-Fletchera-Powella) [2], dla punktu startowego: POM=100, PTFE=0, Brąz=0, Szkło=0.

Tabela 3

Wyniki obliczeń polioptymalizacyjnych

Współcz. wagi	Kryterium1 I_h [$\mu\text{m}/\text{km}$]	Kryterium2 μ [-]	Funkcja celu	Udział składnika w % objętościowo			
				POM	PTFE	Brąz	Szkło
0,00	3,9	0,16	3,9	73,3	21,0	0,0	5,7
0,20	3,9	0,16	3,1	73,3	21,0	0,0	5,7
0,40	3,9	0,16	2,4	73,3	21,0	0,0	5,7
0,50	0,59	0,20	0,40	50,3	19,8	23,6	6,6
0,60	3,9	0,16	1,6	73,3	21,0	0,0	5,7
0,80	0,60	0,20	0,28	50,0	19,8	23,5	6,7
1,00	23,0	0,16	0,16	77,5	19,3	3,2	0,0

Podobieństwo niektórych wyników jest efektem małych zmian wartości funkcji I_h i μ w pobliżu ich minimum i zaokrągleniem wyników obliczeń. Jako najlepszy kompozyt można uznać ten, dla którego wartość funkcji celu jest najmniejsza i wartości kryteriów spełniają narzucone wymagania. Ocenę wyników i wybór składu kompozytu pozostawia się użytkownikom. Wskazać można skład kompozytu odpowiadający współczynnikowi wagi $wag=0,80$ (lub współczynnikowi $wag=0,50$), przy którym funkcja celu ma małą wartość i właściwości tribologiczne (wartości obu kryteriów) są korzystne.

Literatura

- [1] Bil T.: Program komputerowy "POLIOPT", opisany w [5].
- [2] Findeisen W., Szymanowski I., Wierzbicki A.: *Teoria i metody obliczeniowe optymalizacji*. Wyd.2., PWN, Warszawa 1980.
- [3] Mańczak K.: *Technika planowania eksperymentu*. WNT, Warszawa 1976.
- [4] Polański Z.: *Planowanie doświadczeń w technice*. PWN, Warszawa 1984.
- [5] Tarnowski W., Kiczkowski T.: *Komputerowe wspomaganie projektowania*. Skrypt WSI w Koszalinie, Koszalin 1994.

Streszczenie

Metody planowania eksperymentu i optymalizacji umożliwiają, na podstawie wyników badań doświadczalnych, określenie optymalnych składów kompozytów ze względu na ich adresowane właściwości. Zilustrowano to na przykładzie badań wieloskładnikowych kompozytów ślizgowych tworzonych na osnowie polioksymetylenu.

APPLICATION OF OPTIMIZATION METHODS FOR THE DETERMINATION OF THE COMPOSITES CONTENTS TO MEET THE REQUIRED SPECIFICATIONS

Summary

Experiment planning and optimization methods together with the experimental results are very helpful in the determination of on optimum composites structure with regard to the projected specifications. To exemplify this approach, the investigations of multi-component, polyoxymethylene based sliding composites were carried out.

Stefan CHANAS*
Przemysław KOBYLAŃSKI*

METODA PERMUTACJI WIELOKRYTERIALNEJ OCENY DECYZJI - HEURYSTYCZNY ALGORYTM ROZWIĄZANIA

1. Wstęp

Wśród wielu metod wielokryterialnej oceny decyzji prezentowanych w książce Hwanga i Yoona [5] znajduje się także metoda nazwana przez autorów metodą permutacji. Za pomocą tej metody można uporządkować zadany zbiór wariantów decyzji w kolejności od wariantu najlepszego do wariantu najgorszego względem ustalonego zbioru kryteriów oceny. Metoda polega na obliczeniu pewnego wskaźnika optymalności dla każdego możliwego uporządkowania (permutacji) wariantów decyzji i wybraniu uporządkowania o największej wartości tego wskaźnika. Z obliczeniowego punktu widzenia metoda permutacji jest pełnym przeglądem wszystkich możliwych rozwiązań (uporządkowań wariantów decyzji) i z tego powodu może być w praktyce stosowana jedynie dla małej liczby wariantów decyzji. Problem wyboru optymalnego uporządkowania według wskaźnika zastosowanego w metodzie permutacji można sprowadzić do problemu znanego w literaturze poświęconej dyskretnej optymalizacji pod nazwą problemu liniowego uporządkowania. Problem ten należy do klasy problemów NP-trudnych i z tego powodu nie jest możliwe opracowanie dla niego dokładnego i jednocześnie efektywnego algorytmu rozwiązania. Dokładne algorytmy typu podziału i ograniczeń mogą być zastosowane także jedynie do problemów niewielkich rozmiarów - choć większych niż w przypadku metody pełnego przeglądu. Istotnym jest zatem poszukiwanie dobrych algorytmów heurystycznych rozwiązujących problem.

W niniejszej pracy jest prezentowany algorytm, który pod względem dokładności uzyskiwanego rozwiązania (jego bliskości do rozwiązania optymalnego) jest lepszy od wszystkich zaproponowanych do tej pory, heurystycznych algorytmów (zob. [1,2,3]).

2. Sformułowanie problemu i opis metody permutacji

Danych jest m wariantów decyzyjnych D_1, D_2, \dots, D_m i n kryteriów ich oceny K_1, K_2, \dots, K_n . Dla każdego kryterium K_j jest określony wektor $X_j = (x_{1j}, x_{2j}, \dots, x_{mj})$, zawierający oceny poszczególnych wariantów decyzyjnych względem tego kryterium (x_{ij} - ocena wariantu D_i względem kryterium K_j w skali przypisanej temu kryterium). Macierz o wymiarach $m \times n$, w której kolumny pokrywają się z wektorami X_j , tj. macierz $X = [X_1^T, X_2^T, \dots, X_n^T]$ jest nazywana macierzą decyzyjną. Z każdym kryterium K_j jest związana liczba w_j , $w_j > 0$, określająca ważność

*Instytut Organizacji i Zarządzania Politechniki Wrocławskiej
ul. Smoluchowskiego 25, 50-371 Wrocław

tego kryterium, nazywana wagą kryterium K_j . Problem który należy rozwiązać, polega na uporządkowaniu wariantów decyzji D_1, D_2, \dots, D_m , od wariantu najlepszego do najgorszego względem wszystkich kryteriów K_1, K_2, \dots, K_n , na podstawie macierzy decyzyjnej X i z uwzględnieniem wag kryteriów w_1, w_2, \dots, w_n . Głównym źródłem trudności w rozwiązywaniu tak sformułowanego problemu jest niemożność jednoznacznego sprecyzowania znaczenia "dobry względem wszystkich kryteriów". W zależności od przyjętej definicji znaczenia tego wyrażenia i po ewentualnym uwzględnieniu dodatkowych informacji o problemie można dochodzić do istotnie różnych metod jego rozwiązania. Wiele takich metod zostało zaproponowanych w literaturze poświęconej wielokryterialnej ocenie decyzji (zob. [5]). Jedną z nich jest metoda permutacji.

METODA PERMUTACJI (metoda pełnego przeglądu)

Niech M i N oznaczają zbiory, odpowiednio, $M = \{1, 2, \dots, m\}$ i $N = \{1, 2, \dots, n\}$ a S_m zbiór wszystkich permutacji zbioru M tj.

$$S_m = \{(p_1, p_2, \dots, p_m) \in M : \forall i \neq j \ p_i \neq p_j\}.$$

Krok 1. Wyznacz elementy macierzy $E = [e_{ij}]_{m \times m}$ za pomocą następującego wzoru:

$$e_{ij} = \begin{cases} 0 & \text{dla } i = j, \\ \sum_{k \in C_{ij}} w_k & \text{dla } i \neq j. \end{cases} \quad (1)$$

gdzie $C_{ij} = \{k : k \in N, x_{ik} \geq x_{jk}\}$.

Krok 2. Dla każdej permutacji $(p_1, p_2, \dots, p_m) \in S_m$ wyznacz wartość indeksu $w(p)$ według wzoru:

$$w(p) = \sum_{i=1}^{m-1} \sum_{j=i+1}^m e_{p_i p_j} - \sum_{i=2}^m \sum_{j=1}^{i-1} e_{p_i p_j}. \quad (2)$$

Za rozwiązanie przyjmij permutację (p_1, p_2, \dots, p_m) (wyznacza ona optymalne uporządkowanie wariantów decyzji $D_{p_1}, D_{p_2}, \dots, D_{p_m}$) dla której indeks (2) osiąga wartość maksymalną.

Wartość wskaźnika $w(p)$ jest oceną stopnia zgodności uporządkowania $p = (p_1, p_2, \dots, p_m)$ z uporządkowaniami generowanymi przez pojedyncze kryteria z preferowaniem kryteriów o większych wagach.

Łatwo można zauważyć, że problem dyskretnej optymalizacji, do którego sprowadza się wyznaczenie optymalnego uporządkowania:

$$w(p) \xrightarrow{p \in S_m} \max \quad (3)$$

można zastąpić równoważnym problemem

$$c_E(p) \xrightarrow{p \in S_m} \max, \quad (4)$$

w którym

$$c_E(p_1, p_2, \dots, p_m) = \sum_{i=1}^{m-1} \sum_{j=i+1}^m e_{p_i p_j}. \quad (5)$$

Problem (4) jest znany w literaturze, poświęconej dyskretnej optymalizacji, pod nazwą problemu liniowego uporządkowania (ang. linear ordering problem). Jak już wspomniano we wstępie problem ten należy do klasy problemów NP-trudnych i nie istnieją dla niego efektywne obliczeniowo algorytmy rozwiązania. W następnym punkcie przedstawiono opis nowego algorytmu heurystycznego rozwiązującego problem (4).

3. Opis algorytmu

Algorytm zostanie opisany w formie funkcji. Przedtem zostaną zdefiniowane prostsze przekształcenia, które posłużą do opisu zasadniczej funkcji.

Niech T_m oznacza zbiór wszystkich skończonych ciągów elementów zbioru M , w których to ciągach żaden element nie powtarza się, tj.

$$T_m = \{(t_1, t_2, \dots, t_k) \in M^k : 1 \leq k \leq m, \forall i \neq j \ t_i \neq t_j\}. \quad (6)$$

Zbiór T_m jest rozszerzeniem zbioru S_m , tj. $S_m \subset T_m$

Definicja 1. Symbol REVERSE oznacza jednoargumentowe działanie określone na zbiorze T_m , REVERSE: $T_m \rightarrow T_m$, zdefiniowane następująco:

$$\text{REVERSE}(t_1, t_2, \dots, t_k) = (t_k, t_{k-1}, \dots, t_1), \quad (t_1, t_2, \dots, t_k) \in T_m. \quad (7)$$

Definicja 2. Symbol SORT oznacza jednoargumentowe działanie określone na zbiorze T_m , SORT: $T_m \rightarrow T_m$, zdefiniowane następująco:

$$\text{SORT}(t_1, t_2, \dots, t_k) = \begin{cases} (t_1) & \text{dla } k = 1, \\ \text{INSERT}(t_k, \text{SORT}(t_1, t_2, \dots, t_{k-1})) & \text{dla } k > 1, \end{cases}$$

$$\text{INSERT}(i, (t_1, t_2, \dots, t_{k-1})) = (t_1, t_2, \dots, t_{r-1}, i, t_r, \dots, t_{k-1}),$$

gdzie $r \in \{1, 2, \dots, k\}$ i maksymalizuje wartość następującego wyrażenia:

$$\Delta_{c_E}(r, i, (t_1, t_2, \dots, t_{k-1})) = \sum_{j=1}^{r-1} e_{t_j, i} + \sum_{j=r}^{k-1} e_{it_j},$$

przy czym, jeśli maksimum osiągane jest dla kilku wartości wskaźnika r , to wybiera się spośród nich wartość najmniejszą.

Łatwo można zauważyć, że zbiór permutacji S_m , będący podzbiorem zbioru T_m , jest również zamknięty ze względu na operacje REVERSE i SORT, tzn. dla każdej macierzy $E = [e_{ij}]_{m \times m}$ i dla każdej permutacji $p \in S_m$ $\text{REVERSE}(p) \in S_m$ i $\text{SORT}(p) \in S_m$.

Definicja 3. Niech będzie dana dowolna funkcja $f: S_m \rightarrow S_m$. Iteracją funkcji f nazywa się funkcję $f^*: S_m \rightarrow S_m$ zdefiniowaną następującym wzorem:

$$f^*(p) = \begin{cases} p & \text{gdy } c_E(f(p)) = c_E(p), \\ f^*(f(p)) & \text{gdy } c_E(f(p)) \neq c_E(p), \end{cases} \quad p \in S_m$$

ALGORYTM

Krok 1. Wyznacz elementy macierzy $E = [e_{ij}]$ za pomocą wzoru (1).

Krok 2. Wybierz dowolną permutację $p \in S_m$.

Krok 3. Wykonaj operację $(\text{SORT}^* \circ \text{REVERSE})^* \circ \text{SORT}^*(p)$. Jej wynik przyjmij za rozwiązanie problemu¹.

4. Porównanie algorytmu z innymi algorytmami heurystycznymi

W pracy [3] zaproponowano algorytm, który różni się od algorytmu prezentowanego tutaj tylko tym, że w kroku 3 zamiast operacji $(\text{SORT}^* \circ \text{REVERSE})^* \circ \text{SORT}^*(p)$ wykonuje się jedynie iterację operacji SORT, tj. operację $\text{SORT}^*(p)$. W przeprowadzonych eksperymentach, których wyniki również przedstawiono w [3], algorytm ten okazał się lepszy od algorytmów przedstawionych we wcześniejszych pracach [1] i [2]. Natomiast z przedstawionego niżej zaskakującego twierdzenia wynika, że zaproponowany tutaj algorytm dostarcza zawsze niegorsze rozwiązanie niż algorytm z operacją $\text{SORT}^*(p)$ (jeśli startuje się od tego samego rozwiązania początkowego).

Twierdzenie. Dla każdego $t \in T_m$ zachodzi następujący związek:

$$c_E(\text{SORT}(\text{REVERSE}(t))) \geq c_E(t).$$

Funkcja c_E jest naturalnym rozszerzeniem funkcji (5) określonej na zbiorze S_m na szerszy zbiór T_m :

¹Symbol "o" oznacza operację superpozycji (składania) funkcji.

$$c_E(t_1, t_2, \dots, t_k) = \sum_{i=1}^{k-1} \sum_{j=i+1}^k e_{t_i, t_j}$$

Dowód twierdzenia przedstawiono w [4]. Jest ono oczywiście także prawdziwe dla permutacji $p \in S_m \subset T_m$.

Ciekawym jest pytanie jak często przedstawiony tutaj algorytm dostarcza rozwiązania lepszego od rozwiązania uzyskiwanego za pomocą algorytmu proponowanego w [3]. Albo inaczej, jak często dodatkowo wykonana operacja $(\text{SORT}^* \circ \text{REVERSE})^*$ poprawia permutację uzyskaną w wyniku zastosowania jedynie operacji SORT^* . Częściowej odpowiedzi na to pytanie dostarczył przeprowadzony eksperyment obliczeniowy, którego wyniki prezentowane są niżej.

Obliczenia przeprowadzono dla 15 losowo wygenerowanych przykładów liczbowych różnych rozmiarów. Dla każdego przykładu obliczenia przeprowadzono 15 razy przy różnych, losowo generowanych rozwiązaniach początkowych, odnotowując za każdym razem liczbę przypadków, w których uzyskiwano za pomocą proponowanego algorytmu rozwiązanie lepsze od rozwiązania uzyskanego w wyniku zastosowania algorytmu z [3]. Przyjęto następującą notację "P(m,n) - k", w której P(m,n) oznacza wygenerowany losowo problem z m wariantami decyzyjnymi i n kryteriami oceny (x_{ij} - losowo generowane liczby z przedziału od 0 do 1) a k liczbę przypadków (wśród 15 prób z różnymi rozwiązaniami początkowymi), w których uzyskano poprawę rozwiązania. We wszystkich analizowanych przykładach założono równość przyporządkowanych kryteriom wag, tj. $w_i=1, i=1, \dots, n$.

W pierwszej grupie pięciu eksperymentów założono, że $n=m$. Uzyskano wyniki: P(20,20) - 1, P(30,30) - 9, P(40,40) - 8, P(50,50) - 14, P(60,60) - 15. Liczba pozytywnych przypadków, w których uzyskano poprawę rozwiązania, rosła zatem wraz ze wzrostem liczb m i n. W problemie, w którym było $m=n=60$, uzyskano poprawę rozwiązania we wszystkich piętnastu przypadkach.

W drugiej grupie pięciu eksperymentów założono stałą wartość $m=20$ i zmieniano jedynie w kolejnych losowo generowanych problemach liczbę kryteriów n. Uzyskano wyniki: P(20,10) - 5, P(20,30) - 8, P(20,40) - 9, P(20,50) - 5, P(20,100) - 4.

W trzeciej, ostatniej grupie, złożonej z czterech eksperymentów, założono zależność $n=2m$. Uzyskano wyniki: P(30,60) - 9, P(40,80) - 8, P(50,100) - 14, P(60,120) - 15.

Wyniki przeprowadzonego eksperymentu zdają się potwierdzać wniosek, że przewaga zaproponowanego tutaj algorytmu nad algorytmem z [3] szybko się powiększa wraz ze wzrostem liczby wariantów decyzji m. Natomiast przy ustalonej liczbie $m=20$ i tylko rosnącej liczbie kryteriów przewaga ta zaczęła powyżej wartości $n=40$ maleć.

Literatura

1. Chanas S., Florkiewicz B., Galant-Pater M.: *Heuristic algorithms for the permutation method of the multiple attribute decision making*. *Badania Operacyjne i Decyzje*, Nr 3, 1991.
2. Chanas S., Florkiewicz B., Galant-Pater M.: *Computing aspects of the permutation method in the multiple attribute decision making*. *Badania Operacyjne i Decyzje*, Nr 4, 1991.
3. Chanas S., Kobyłański P.: *Heuristic algorithm solving the problem of linear ordering*. *Badania Operacyjne i Decyzje*, Nr 3, 1993.
4. Chanas S., Kobyłański P.: *A new heuristic algorithm solving the problem of linear ordering*. In preparation.
5. Hwang Ch., Yoon K.: *Multiple Attribute Decision Making. Methods and Application. A State-of-the-Art-Survey, Lecture Notes in Economics and Mathematical Systems*, Nr 186, Springer-Verlag, Berlin-Heidelberg-New York, 1981.

STRESZCZENIE

Opisano permutacyjną metodę wielokryterialnej oceny decyzji oraz sposób sprowadzenia jej do problemu liniowego uporządkowania. Zaprezentowano nowy algorytm heurystyczny rozwiązujący ten problem. Przedstawiono wyniki eksperymentów obliczeniowych wskazujących na to, że prezentowany algorytm jest lepszy od dotychczas opublikowanych.

THE PERMUTATION METHOD OF A MULTICRITERIA DECISION EVALUATION - A HEURISTIC ALGORITHM

The permutation method of a multicriteria decision evaluation and a way of reducing it to the linear ordering problem have been described. A new heuristic algorithm solving this problem has been proposed. The results of computational experiments have been presented, which show that the proposed algorithm is better than the ones published so far.

Stefan CHANAS*
Dorota KUCHTA*

ALGORYTM ROWIAZYWANIA DWUKRYTERIALNYCH ZADAŃ PROGRAMOWANIA LINIOWEGO Z PARAMETREM WE WSPÓŁCZYNNIKACH FUNKCJI CELU

1. Wstęp

W pracy przedstawiono algorytm wyznaczania niezdominowanych rozwiązań bazowych dwukryterialnego zadania programowania liniowego, w którym współczynniki funkcji celu zależą w sposób liniowy od parametru. Jak dotąd algorytm taki nie został zaproponowany w literaturze. Ze względu na szczupłość miejsca prezentacja algorytmu została skrócona. Bardziej szczegółowy opis wraz z uzasadnieniem znajduje się w [1]. Przed sformulowaniem algorytmu przeznaczonego dla problemu parametrycznego przedstawiono również algorytm wyznaczania niezdominowanych rozwiązań bazowych dla dwukryterialnych zadań nieparametrycznych. Algorytmy takie są znane w literaturze. Tutaj jednak zaproponowano pewną zmodyfikowaną wersję algorytmu. Celem modyfikacji było umożliwienie przeniesienia algorytmu na przypadek parametryczny.

2. Sformułowanie problemu

Problem dwukryterialnego programowania liniowego może być zapisany w następującej postaci:

$$\begin{aligned} f^1(x; t) &= \sum_{i=1}^n c_i^1(t) \cdot x_i \rightarrow \max, \\ f^2(x; t) &= \sum_{i=1}^n c_i^2(t) \cdot x_i \rightarrow \max, \\ \sum_{i=1}^n a_i^j \cdot x_i &= b^j, \quad j = 1, 2, \dots, m, \\ x_i &\geq 0, \quad i = 1, 2, \dots, n, \end{aligned} \tag{1}$$

gdzie x oznacza wektor (x_1, \dots, x_n) , $t \in T = [T^0, T^1]$ jest parametrem, a $c_i^1(t)$, $c_i^2(t)$ są liniowymi funkcjami tego parametru. Zakłada się, że zbiór rozwiązań dopuszczalnych problemu (1) jest niepusty i ograniczony.

Dla ustalonej wartości parametru $t = t_0$ problem (1) jest zwykłym zadaniem dwukryterialnego programowania liniowego (niech $c_i^k(t_0) = d_i^k$, $i = 1, \dots, n$, $k = 1, 2$):

*Instytut Organizacji i Zarządzania Politechniki Wrocławskiej
ul. Smoluchowskiego 25, 50-371 Wrocław

$$f^1(\mathbf{x}) = \sum_{i=1}^n d_i^1 \cdot x_i \rightarrow \max,$$

$$f^2(\mathbf{x}) = \sum_{i=1}^n d_i^2 \cdot x_i \rightarrow \max,$$

$$\sum_{i=1}^n a_i^j \cdot x_i = b_j, \quad j = 1, 2, \dots, m, \quad (2)$$

$$x_i \geq 0, \quad j = 1, 2, \dots, n.$$

Przez rozwiązanie problemu (2) rozumie się wyznaczenie wszystkich bazowych rozwiązań niezdominowanych, a przez rozwiązanie problemu (1) wyznaczenie tych wszystkich jego rozwiązań bazowych \mathbf{x} , które są niezdominowane przynajmniej dla jednej wartości parametru $t \in T$, wraz z odpowiednimi zbiorami $S(\mathbf{x})$. Symbol $S(\mathbf{x})$ oznacza zbiór tych wszystkich wartości parametru $t \in T$, dla których \mathbf{x} jest niezdominowanym rozwiązaniem odpowiedniego zadania nieparametrycznego.

3. Zmodyfikowana wersja algorytmu wyznaczania rozwiązań niezdominowanych w przypadku zdeterminowanym

Algorytm rozwiązania problemu (2) składa się z następujących kroków:

- I. Wyznaczyć dowolne rozwiązanie bazowe niezdominowane \mathbf{x}_0 problemu (2) wraz z odpowiadającą mu tablicą sympleksową $\text{Sympl}(\mathbf{x}_0)$. Podstawić:

$$\text{NIEZDOMINOWANE} := \{\mathbf{x}_0\}; \quad \text{DO_SPRAWDZENIA} := \{(\mathbf{x}_0, \text{Sympl}(\mathbf{x}_0))\};$$

- II. Jeśli $\text{DO_SPRAWDZENIA} = \emptyset$, to STOP. W przeciwnym wypadku wybrać dowolny element $(\mathbf{x}, \text{Sympl}(\mathbf{x})) \in \text{DO_SPRAWDZENIA}$ i podstawić:

$$\text{DO_SPRAWDZENIA} := \text{DO_SPRAWDZENIA} \setminus \{(\mathbf{x}, \text{Sympl}(\mathbf{x}))\};$$

- III. Dla wszystkich rozwiązań sąsiednich do \mathbf{x} sprawdzić, czy są one niezdominowane. Wszystkie rozwiązania sąsiednie niezdominowane dołączyć do listy NIEZDOMINOWANE i, wraz z odpowiednią tablicą sympleksową, do listy DO_SPRAWDZENIA - jeśli rozwiązań tych listy jeszcze nie zawierają. Przejść do kroku II.

Krok I można zrealizować np. poprzez rozwiązanie zadania jednokryterialnego z tym samym zbiorem ograniczeń co (2) i z następującą funkcją celu:

$$f^1(\mathbf{x}) + f^2(\mathbf{x}) \rightarrow \max.$$

W przypadku kroku III proponuje się nowy sposób postępowania, który w przeciwieństwie do znanych z literatury procedur łatwo daje się uogólnić na przypadek proble-

mu (1). Sposób ten wymaga wyznaczania tablic sympleksowych wszystkich rozwiązań sąsiednich do x . W niektórych specjalnych przypadkach można tego uniknąć (por. [1]), ale w niniejszym artykule jest rozważany jedynie przypadek ogólny.

Niech y oznacza aktualnie analizowane bazowe rozwiązanie sąsiednie do rozwiązania x , symbole z_i^1, z_i^2 ($i=1, \dots, n$) współczynniki optymalności rozwiązania y ze względu na funkcje celu, odpowiednio, f_1 i f_2 , a symbol I_{NB} zbiór indeksów odpowiadającym zmiennym niebazowym rozwiązania y . Prawdziwe jest następujące twierdzenie:

Twierdzenie 1: Rozwiązanie y jest niezdominowane wtedy i tylko wtedy, gdy istnieje liczba $\lambda \in (0,1)$ spełniająca następujący układ nierówności:

$$\lambda z_i^1 + (1-\lambda)z_i^2 \geq 0, \quad i \in I_{NB}. \quad (3)$$

Praktyczne zastosowanie powyższego twierdzenia prowadzi do następującej procedury stwierdzania zdominowania lub niezdominowania dowolnego rozwiązania bazowego y problemu (2) danego wraz ze swoją tablicą sympleksową (w III kroku algorytmu procedurę tę należy zrealizować dla każdego y sąsiedniego do x):

IIIa. Podstaw MAX:=1, MIN:=0;

IIIb. Dla każdego $i \in I_{NB}$:

$$\text{jeśli } z_i^1 - z_i^2 > 0, \text{ to podstaw } \text{MIN} := \max(\text{MIN}, \frac{-z_i^2}{z_i^1 - z_i^2});$$

$$\text{jeśli } z_i^1 - z_i^2 < 0, \text{ to podstaw } \text{MAX} := \min(\text{MAX}, \frac{-z_i^2}{z_i^1 - z_i^2});$$

$$\text{jeśli } z_i^1 - z_i^2 = 0, \text{ to podstaw } \text{MIN} := 1;$$

IIIc. Jeśli $\text{MIN} \leq \text{MAX}$ i $[\text{MIN}, \text{MAX}] \cap (0,1) \neq \emptyset$, to rozwiązanie y jest niezdominowane; w przeciwnym przypadku rozwiązanie y jest zdominowane.

4. Algorytm wyznaczania rozwiązań niezdominowanych w przypadku parametrycznym

Przedstawiony w poprzednim punkcie algorytm wyznaczania wszystkich bazowych rozwiązań niezdominowanych problemu (2) w naturalny sposób można uogólnić na przypadek parametryczny, tzn. na przypadek problemu (1). Efektem tego rozszerzenia jest algorytm, którego ogólny przebieg jest następujący:

I(p): Wyznaczyć dowolny zbiór rozwiązań bazowych $\{x^1, x^2, \dots, x^p\}$ problemu (1), wraz z odpowiednimi zbiorami $S(x^1), S(x^2), \dots, S(x^p)$ i tablicami sympleksowymi, o następującej własności:

$$S(x^1) \cup S(x^2) \cup \dots \cup S(x^p) \supseteq T, S(x^h), (h = 1, \dots, p).$$

Podstawić:

$$\text{NIEZDOMINOWANE} := \{(x^1, S(x^1)), (x^2, S(x^2)), \dots, (x^p, S(x^p))\};$$

DO_SPRAWDZENIA :=

$$\{(x^1, \text{Sympl}(x^1)), (x^2, \text{Sympl}(x^2)), \dots, (x^p, \text{Sympl}(x^p))\};$$

II(p): Jeśli DO_SPRAWDZENIA = \emptyset , to STOP.

W przeciwnym przypadku wybrać dowolny element $(x, \text{Sympl}(x)) \in \text{DO_SPRAWDZENIA}$ i podstawić:

$$\text{DO_SPRAWDZENIA} := \text{DO_SPRAWDZENIA} \setminus \{(x, \text{Sympl}(x))\};$$

III(p): Dla wszystkich rozwiązań y sąsiednich do x sprawdzić, czy $S(y) \neq \emptyset$. Wszystkie rozwiązania sąsiednie o tej własności dołączyć wraz ze zbiorem $S(y)$ do listy NIEZDOMINOWANE i, wraz z odpowiednią tablicą sympleksową, do listy DO_SPRAWDZENIA - jeśli rozwiązań tych listy te jeszcze nie zawierają. Przejść do kroku II(p).

Krok I(p) można zrealizować poprzez rozwiązanie jednokryterialnego zadania programowania liniowego z parametrem w funkcji celu, z tymi samymi ograniczeniami co zadanie (1) i z funkcją celu:

$$f^1(x; t) + f^2(x; t) \rightarrow \max.$$

Dla każdego znalezionej w ten sposób rozwiązania bazowego x należy dodatkowo wyznaczyć zbiór $S(x)$ - za pomocą przedstawionej niżej procedury będącej uogólnieniem procedury złożonej z kroków IIIa - IIIc. Ta sama procedura jest potrzebna również w kroku III(p).

Procedura wyznaczania zbioru $S(x)$

IIIa(p). Podstaw $\text{MAX}(t) := 1, \text{MIN}(t) := 0$, dla każdego $t \in T$;

IIIb(p). Dla każdego $i \in I_{NB}$ podstaw:

$$\text{MIN}(t) := \max\left\{\text{MIN}(t), \frac{-z_i^2(t)}{z_i^1(t) - z_i^2(t)}\right\}, \quad \text{dla } t \in T \text{ spełniających warunek:}$$

$$z_i^1(t) - z_i^2(t) > 0,$$

$$\text{MAX}(t) := \min\left\{\text{MAX}(t), \frac{-z_i^2(t)}{z_i^1(t) - z_i^2(t)}\right\}, \quad \text{dla } t \in T \text{ spełniających warunek:}$$

$$z_i^1(t) - z_i^2(t) < 0,$$

$$\text{MIN}(t) := 1, \quad \text{dla } t \in T \text{ spełniających warunek: } z_i^1(t) - z_i^2(t) = 0 \text{ i } z_i^2(t) < 0;$$

IIIc(p). Podstaw $S(\mathbf{x}) = \{t \in T : (\text{MIN}(t) \leq \text{MAX}(t) \text{ i } [\text{MIN}(t), \text{MAX}(t)] \cap (0, 1) \neq \emptyset)\}$.

W przypadku parametrycznym wskaźniki optymalności $z_i^k(t)$ ($k=1, 2; i=1, \dots, n$) są liniowymi funkcjami parametru t . Z tego też powodu wielkości MIN i MAX, które w przypadku nieparametrycznym były stałymi, stają się teraz również funkcjami parametru t : $\text{MIN}(t)$ i $\text{MAX}(t)$, $t \in T$. W wyniku wykonania procedury przedział parametru T zostaje podzielony na podprzedziały, w których funkcje $\text{MIN}(t)$ i $\text{MAX}(t)$ przyjmują wartość stałą, równą 0 lub 1, albo są równe niektórym spośród funkcji:

$$\frac{-z_i^2(t)}{z_i^1(t) - z_i^2(t)} \quad (i \in I_{NB}).$$

Aby określić końce tych podprzedziałów trzeba rozwiązywać

równości i nierówności liniowe i kwadratowe z jedną zmienną.

Działanie opisanej wyżej procedury wyznaczania zbioru $S(\mathbf{x})$ dla dowolnego rozwiązania bazowego \mathbf{x} ilustruje następujący przykład:

Przykład: Dane jest następujące dwukryterialne parametryczne zadanie programowania liniowego:

$$(-1 - 29t)x_1 + (-2 - t)x_2 \rightarrow \max,$$

$$(1 + 39t)x_1 + (3 + t)x_2 \rightarrow \max,$$

$$x_1 + x_2 = 1,$$

$$x_1, x_2 \geq 0,$$

$$t \in [0, 1].$$

Jednym z rozwiązań bazowych jednokryterialnego zadania parametrycznego z funkcją celu :

$$f_1(x; t) + f_2(x; t) = 10t \cdot x_1 + x_2 \rightarrow \max$$

jest wektor $\mathbf{x} = (1, 0)$. Odpowiadające mu współczynniki optymalności mają postać:

$$z_1^1(t) = 0, z_1^2(t) = 0, z_2^1(t) = 1 - 28t, z_2^2(t) = -2 + 38t, t \in [0, 1].$$

W wyniku realizacji wyżej opisanej procedury otrzymuje się następujące wartości funkcji MAX(t) i MIN(t):

dla $t \in [0, 1/22)$	MIN=(2-38t)/(3-66t),	MAX=1,
dla $t \in [1/22, 1/22]$	MIN=1,	MAX=1,
dla $t \in (1/22, 1]$	MIN=0,	MAX=(2-38t)/(3-66t).

Wyznaczany w kroku IIIc(p) zbiór S(x) przyjmuje wobec zależności:

- | | | |
|----|--------------------------|----------------------------|
| a) | $(2-38t)/(3-66t) < 1$ | dla $t \in [0, 1/28)$, |
| b) | $(2-38t)/(3-66t) \geq 1$ | dla $t \in [1/28, 1/22)$, |
| c) | $(2-38t)/(3-66t) > 0$ | dla $t \in (1/19, 1]$, |
| d) | $(2-38t)/(3-66t) \leq 0$ | dla $t \in (1/22, 1/19]$, |

postać: $S(x) = [0, 1/28) \cup (1/19, 1]$.

Program komputerowy realizujący algorytm rozwiązywania dwukryterialnych zadań programowania liniowego z parametrem w funkcjach celu został przedstawiony w [1]. Jest to wersja robocza programu, napisana w języku Pascal.

Literatura

1. Kuchta D.: *Zadania programowania liniowego z przedziałowymi i rozmytymi współczynnikami w funkcji celu*. Praca doktorska, Inst. Org. i Zarz. Politechniki Wr., Raport serii PRE, nr 7, 1993.
2. Steuer R.: *Multiple Criteria Optimization*. John Wiley & Sons, New York 1986.

Streszczenie

Przedstawiono algorytm wyznaczania niezdominowanych rozwiązań bazowych dwukryterialnego zadania programowania liniowego ze współczynnikami w funkcji celu zależnymi od parametru. Zaproponowany algorytm jest modyfikacją znanej metody Zele-ny'ego przeznaczonej dla przypadku zadania nieparametrycznego.

AN ALGORITHM FOR SOLVING BICRITERIA LINEAR PROGRAMMING PROBLEMS WITH A PARAMETER IN THE OBJECTIVE FUNCTIONS

An algorithm determining the non-dominated basic solutions of the bicriteria linear programming problem, in which the objective functions coefficients depend on a parameter, has been presented. The proposed algorithm is a modification the known Zeleny method, destined for the non-parametric case.

Komputerowa symulacja wybuchu podwodnego - program ABAQUS Explicit

1. WPROWADZENIE

Powstawanie podczas wybuchu podwodnego fali ciśnieniowej, jej propagacja, interakcja z ustrojem oraz określenie odpowiedzi ustroju na tego typu wymuszenie było od dawna przedmiotem licznych analiz. Aktualny stan zaawansowania badań omawianej problematyki znaleźć można w wielu materiałach konferencyjnych jak też pracach książkowych ([3], [4], [6]).

Celem niniejszej pracy było przeprowadzenie komputerowej symulacji wybuchu podwodnego ładunku TNT, umieszczonego w bezpośredniej bliskości konstrukcji tunelu podwodnego, przy wykorzystaniu do obliczeń programu metody elementów skończonych ABAQUS Explicit (1992). Metody rozwiązania stosowane w tym programie zostały szczegółowo przedstawione w jego opisach [9] i nie będą tutaj omawiane.

2. SFORMUŁOWANIE ZAGADNIENIA

Opis konstrukcji poddanej działaniu wybuchu podwodnego

Analizie poddano konstrukcję tunelu podwodnego, zakotwiczonego nad dnem na głębokości około 40 m. Tunel ma budowę cylindryczną o przekroju kołowym i średnicy zewnętrznej 15 m. Część ustroju poddaną analizie numerycznej przedstawiono na Rys.1, w postaci siatki elementów skończonych.

Główną część konstrukcji stanowi ściana warstwowa, którą tworzą zewnętrzne warstwy blachy stalowej 25 mm, przedzielone masywną warstwą betonu. Na Rys.1(b) przedstawiono sposób połączenia warstw za pomocą elementów stalowych.

Warstwowa ściana cylindryczna jest osłonięta z zewnątrz współosiową powłoką zabezpieczającą, wykonaną z blachy stalowej 35 mm, pokrytą z zewnątrz 10 cm warstwą betonu. Litera A na Rys.1(b) wskazuje punkt przyłożenia ładunku wybuchowego.

Obciążenie impulsowe generowane podczas wybuchu podwodnego

Podczas wybuchu następuje bardzo szybka zamiana stanu skupienia materiału wybuchowego - z ciała stałego w gaz o bardzo wysokiej temperaturze i ciśnieniu. Jeżeli wybuch zachodzi pod wodą, jego produkty gazowe formują obszar, który rozszerza się w ośrodku przy jednoczesnych pulsacyjnych zmianach objętości i ciśnienia. Fala ciśnieniowa utworzona podczas wybuchu propaguje się w ośrodku oddziałując na znajdujące się w nim obiekty, oraz odbijając się od nich.

W wielu komputerowych systemach obliczeniowych istnieje możliwość modelowania tego zjawiska ([9]), sprowadza to jednak problem do trójwymiarowego zagadnienia interakcji materiał wybuchowy - woda - ustrój. W rezultacie, analiza komputerowa wymaga sprzętu o bardzo dużych możliwościach obliczeniowych (pamięć wewnętrzna i zewnętrzna oraz prędkość przetwarzania danych). Z praktycznego punktu widzenia,

możliwe jest zastosowanie empirycznych związków dostępnych w literaturze ([3], [4]), dla przedstawienia obciążenia w postaci zmiennego w czasie i przestrzeni ciśnienia działającego na zewnętrzną powierzchnię ustroju.

Według Henrycha [4], dla podwodnej eksplozji kulistego ładunku TNT o wadze W [kG], ciśnienie p [kG/cm²] w punkcie odległym o R [m] od środka ładunku wybuchowego, opisane jest zależnością:

$$p(R, t) = 0 \text{ dla } t < t_0, \quad p(R, t) = P_{\max}(R) \exp\left(\frac{-(t-t_0)}{v}\right) \text{ dla } t \geq t_0$$

gdzie c_0 jest prędkością dźwięku w wodzie (=1460 m/s), $t_0 = R/c_0$, zaś

$$v = 10^{-4} W^{1/3} \bar{R}^{-0.24}$$

Wielkość $\bar{R} = R/W^{1/3}$.

Ciśnienie maksymalne P_{\max} w równaniu (2.1) określone jest w pracy [4] poprzez funkcje zależne jedynie od \bar{R} .

Ze względu na przybliżony charakter zastosowanych związków empirycznych, przeprowadzono dodatkowo ich analizę modelując część ustroju będącą w bezpośrednim kontakcie z wodą, materiał wybuchowy oraz wodę, przy pomocy dostępnych w programie Abaqus Explicit [9] równań stanu Jonesa - Wilkensa - Lee dla TNT, oraz Mie - Gruneisena dla wody. Uzyskane wyniki analizy potwierdziły w całej rozciągłości ważność związków podanych przez Henrycha w pracy [4], jak też przyjętego w dalszej analizie sposobu ich wykorzystania.

Zastosowane modele materiałowe

Dla stali przyjęto model plastyczny von Misesa ze wzmocnieniem kinematycznym oraz stowarzyszonym prawem płynięcia. Efekty związane z dużymi prędkościami przyrostu odkształceń uwzględniono poprzez zależność Cowpera - Symonds, wiążącą dynamiczną granicę plastyczności σ_y z prędkością przyrostu odkształceń. Dla przypadku jednowymiarowego ma ona postać:

$$\frac{\sigma_y}{\sigma_0} = 1 + \left(\frac{\dot{\epsilon}^{vp}}{D}\right)^{1/p}$$

gdzie σ_0 jest statyczną granicą plastyczności, D , p są parametrami materiałowymi (dla stali $D=40 \text{ s}^{-1}$, $p = 5$), zaś $\dot{\epsilon}^{vp}$ jest odkształceniem wiskoplastycznym.

W omawianym modelu nie uwzględnia się wzrostu wartości współczynnika sprężystości dla dużych wartości prędkości odkształceń ze względu na dominującą rolę odkształceń plastycznych w deformacji ustroju.

Otwartym problemem jest uwzględnienie w modelu materiałowym zjawiska zniszczenia ("failure"). W omawianym przypadku określono je poprzez podanie granicznej wartości $\epsilon_{eq}^{p,lim}$, gdzie:

$$\epsilon_{eq}^p(t) \equiv \int_0^t \left[\frac{2}{3} \dot{\epsilon}_{ij}^p(\tau) \dot{\epsilon}_{ij}^p(\tau) \right]^{1/2} d\tau$$

Dla betonu przyjęto klasyczny model plastyczności wg. Druckera - Pragera, z niestowarzyszonym prawem płynięcia. Zależność charakterystyk materiałowych od

prędkości odkształceń plastycznych uwzględniono adaptując cytowany uprzednio model Cowpera - Symonds. Na podstawie przeprowadzonej analizy przyjęto wartości współczynników: $p=6$, $D=160 \text{ s}^{-1}$.

Zastosowane elementy skończone

Przy modelowaniu ustroju przedstawionego na Rys. 1 wykorzystano dwa rodzaje modeli skończonych:

a) stal: czworoboczne elementy powłokowe o czterech węzłach, z rozkładem naprężeń ścinających wg. Mindlina-Reisnera, sześciu stopniami swobody w każdym węźle, jednym punkcie całkowania Gaussa na obszarze elementu;

b) beton: sześcioboczne elementy trójwymiarowe o ośmiu węzłach, trzech stopniach swobody w każdym węźle, jednym punkcie całkowania Gaussa na obszarze elementu.

Główne cele i rezultaty przeprowadzonej analizy

Głównymi celami było: (a) określenie sposobu modelowania wybuchu podwodnego a zwłaszcza jego efektów w postaci fali ciśnieniowej działającej na konstrukcję; (b) określenie przydatności uzyskanych rezultatów dla praktycznych potrzeb związanych z projektowaniem konstrukcji; (c) wybór metodologii prowadzenia obliczeń przy wykorzystaniu dostępnych profesjonalnych programów metody elementów skończonych.

Rys.2 przedstawia sekwencję odkształcenia konstrukcji w różnych chwilach po detonacji ładunku 40 kg TNT umieszczonego w punkcie A. Dla lepszej czytelności pominięto warstwę środkową oraz wewnętrzną ściany warstwowej.

Rys.3(a) przedstawia zdeformowany kształt ustroju w czasie $t = 15 \text{ ms}$ po wybuchu. Czas ten może być uważany za koniec procesów związanych z deformacją plastyczną ustroju. Na Rys. 3(b) przedstawiono przebiegi w czasie dla całkowitej energii kinetycznej, energii dysypowanej poprzez odkształcenia plastyczne, jak też całkowitej pracy sił wewnętrznych. Jak można zauważyć, po około 15 ms nie następuje wzrost wartości całkowitej energii odkształceń plastycznych.

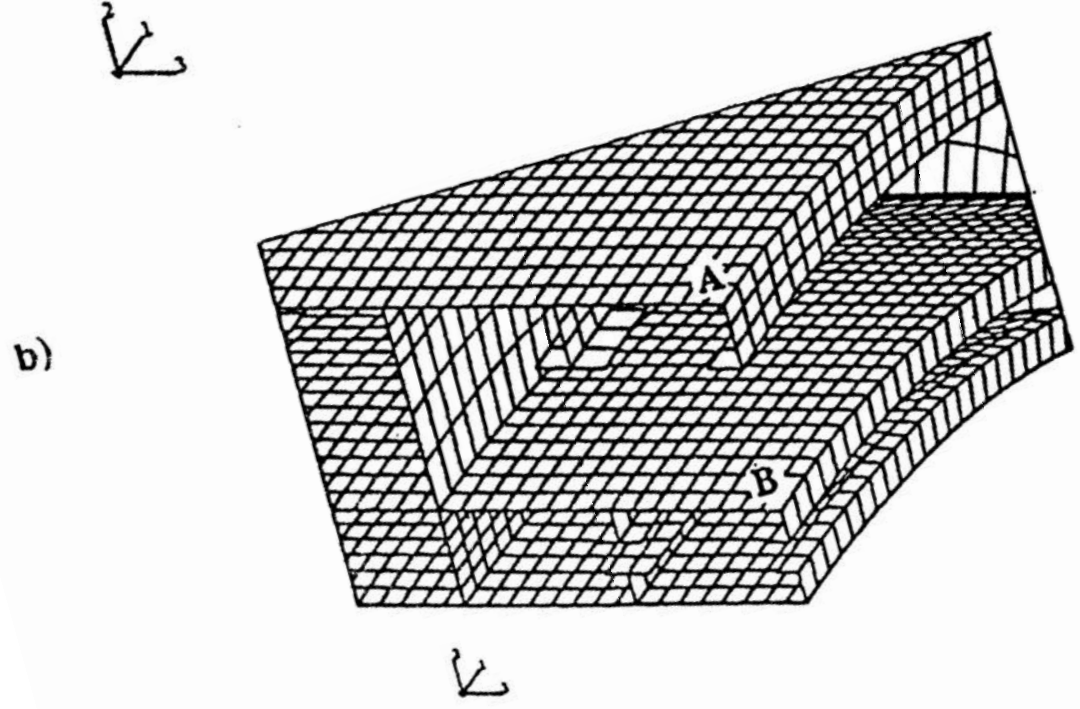
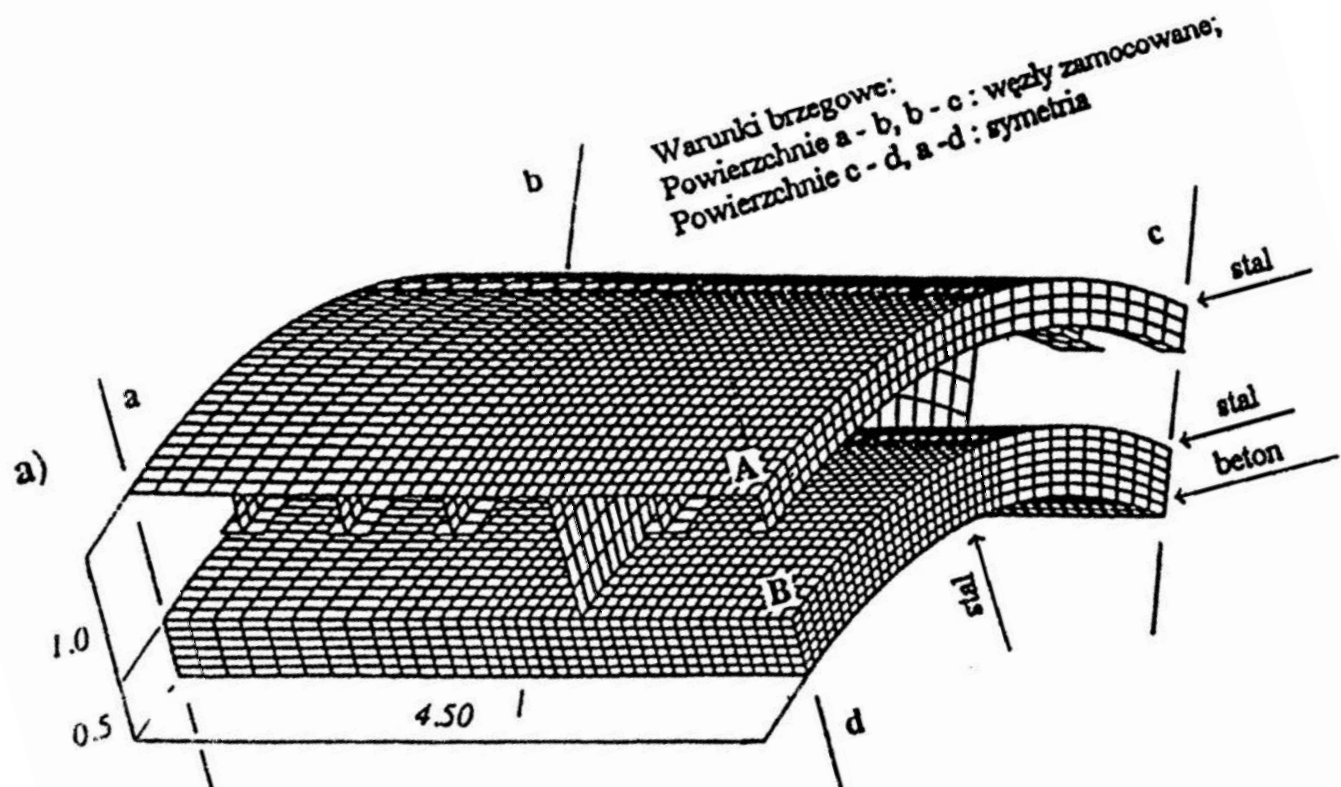
3. WNIOSKI KOŃCOWE

Na podstawie przeprowadzonej analizy można sformułować następujące wnioski o charakterze ogólnym:

1. Rozkład w czasie oraz przestrzeni ciśnienia działającego na powierzchnię w sąsiedztwie centrum wybuchu może być modelowane z użyciem empirycznych związków podanych w literaturze [3] i [4]. Alternatywnie, możliwe jest określenie tego ciśnienia na podstawie komputerowej symulacji wybuchu podwodnego. Przykłady tego typu rozwiązań omówiono przez autora w pracach [10] - [14].

2. Całkowita energia dysypowana poprzez odkształcenia plastyczne ustroju przedstawia istotną ze względu na sterowanie przebiegiem analizy wartość. W momencie, kiedy osiąga wartość końcową, przestają zachodzić efekty plastyczne. Z praktycznego punktu widzenia oznacza to zakończenie analizy.

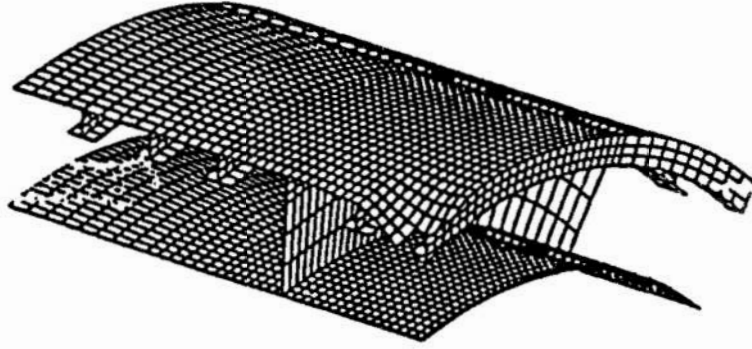
5. W nieliniowej dynamice stosowane są generalnie dwa sposoby dyskretyzacji zagadnienia w czasie: (a) bazujące na algorytmach typu "implicit", bezwarunkowo stabilnych; (b) typu "explicit", warunkowo (dla odpowiednio małych przyrostów czasu) stabilnych. Porównanie obu algorytmów, przeprowadzone przez autorów, wykazało znacznie wyższą sprawność podejścia "explicit". W danym przypadku była to zaimplementowana w programie Abaqus metoda różnic centralnych.



Rys. 1. Siatka elementów skończonych
 Fig.1. Finite element mesh

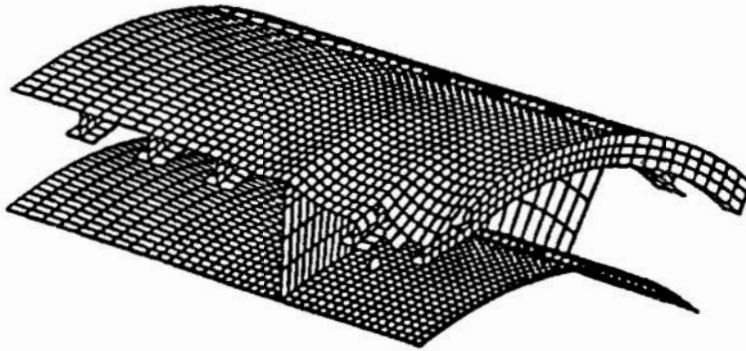
a)

$t = 1.25 \text{ ms}$



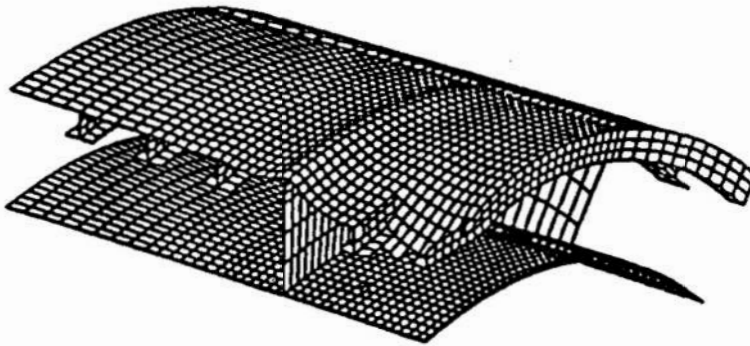
b)

$t = 2.50 \text{ ms}$



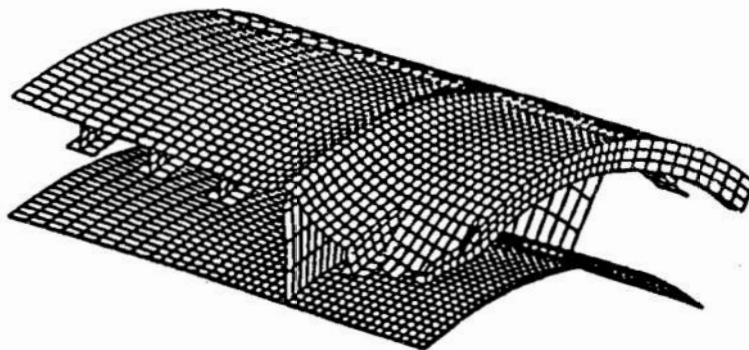
c)

$t = 3.75 \text{ ms}$



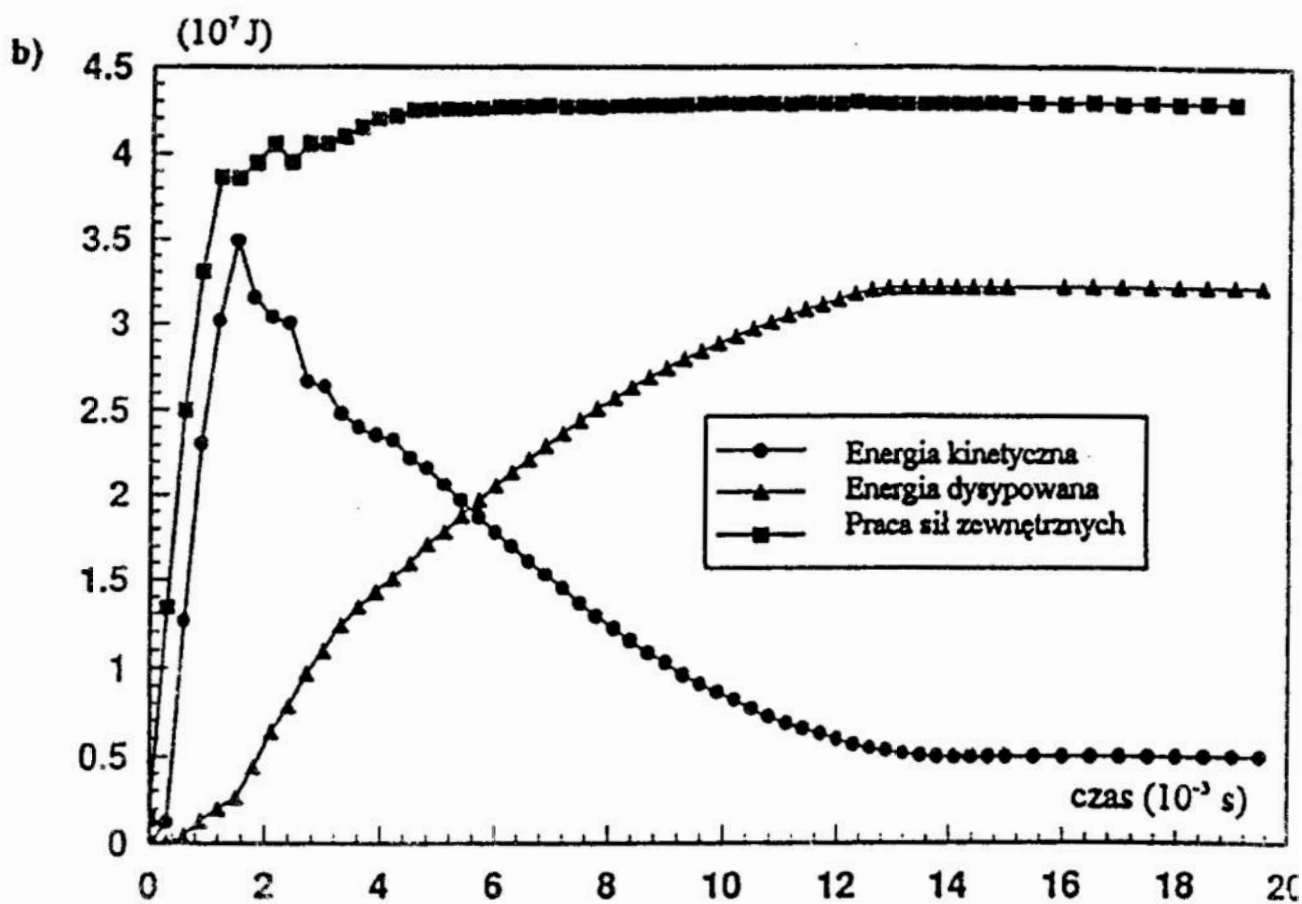
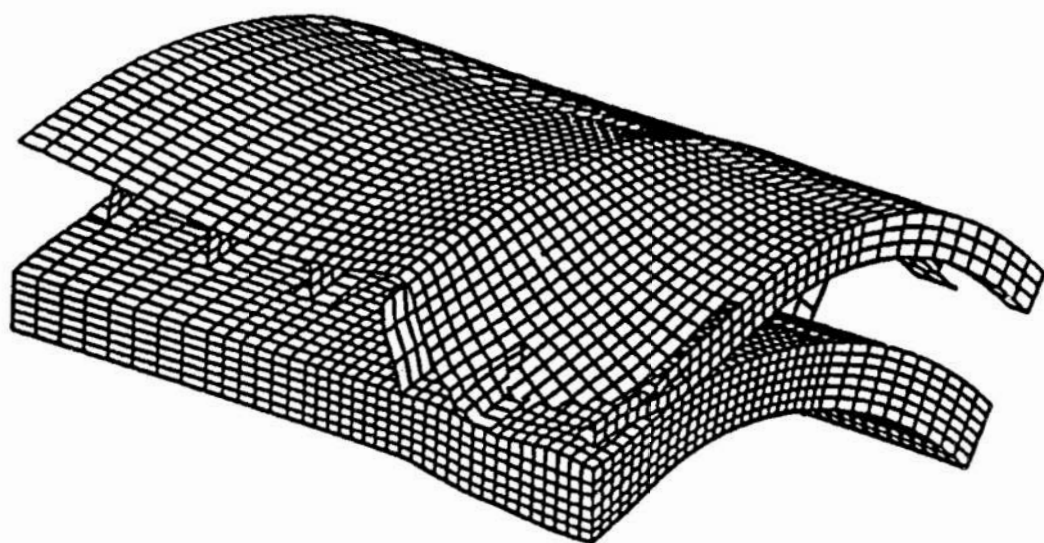
d)

$t = 5.00 \text{ ms}$



Rys. 2. Odkształcenia siatki w poszczególnych chwilach czasowych
Fig. 2. Deformation of the mesh in particular moments

a)



Rys. 3. a) Końcowa postać odkształconego ustroju

b) Energia kinetyczna, dysypowana oraz praca sił zewnętrznych w funkcji czasu

Fig. 3 a) Final form of the deformed structure

b) Kinetic energy, dissipated energy and external work in a function of time

Literatura

- [1] Salvatorelli-D'Angelo, F., Dynamic response and failure of singular and sandwich cylindrical shells under lateral blast loading. *Structures under shock and impact*, P.S.Bulson, Elsevier, Amsterdam, 1989, 381-395;
- [2] Islam, M.N., Kormi, K., Al-Hassani, S.T.S., Dynamic response of a thin-walled cylinder to side pressure pulse, *Engng. Struct.*, 1992, 14, 395-412;
- [3] Cole, R.D., *Underwater Explosions*, PUP, 1948;
- [4] Henrych, J., *The Dynamics of Explosions and Its Use*, Elsevier, Amsterdam, 1979;
- [5] Jones, N., Wierzbicki, T., *Structural Failure*, J. Wiley, New York, 1989;
- [6] Chien, C.Y., Kunze, H.D., Meyer, L.W., *Impact Loading and Dynamic Behaviour of Materials*, DGM Informationgesellschaft, Oberusel, 1988;
- [7] Zukas, J.A., *High Velocity Impact Dynamics*, J. Wiley, New York, 1990;
- [8] Jones, N., *Structural Impact*, Cambridge University Press, Cambridge, 1989;
- [9] Abaqus/Explicit Theory and User's Manual ver. 5.2, HKS, Providence, RI, USA, 1992;
- [10] Cichocki, K., Maier, G., Perego, U., *Analisi degli effetti strutturali di esplosioni su una struttura sommersa*, Raport techniczny, Dipartimento di Ingegneria Strutturale, Politecnico di Milano, Mediolan, 1993;
- [11] Cichocki, K., Maier, G., Perego, U., Esperienze computazionali in dinamica veloce nonlineare, *Settimo Convegno Italiano di Meccanica Computazionale*, Trieste, 1993, 140-144;
- [12] K. Cichocki, G. Maier, U. Perego, "On dynamic analysis of hybrid (steel - concrete) structures subjected to explosion", 8th DYMAT Technical Conference, CEC Joint Research Centre, Ispra, Włochy, Październik 1993;
- [13] K. Cichocki, G. Maier, U. Perego, "Analiza zniszczeń powłoki cylindrycznej wywołanych wybuchem podwodnym", Materiały Konferencji "Odporność udarowa konstrukcji", Gdynia, listopad 1993
- [14] K. Cichocki, "Computer Analysis of Dynamic Response Due to Underwater Explosion on a Hybrid Structure", Materiały Konferencji "ABAQUS Users Conference", Newport, USA, czerwiec 1994

Streszczenie

W pracy zaprezentowano oraz poddano dyskusji wyniki analizy skutków oddziaływania wybuchu podwodnego na powłokę cylindryczną, przeprowadzonej przy użyciu profesjonalnego programu metody elementów skończonych. Sformułowano szereg uwag o charakterze ogólnym, dotyczących metodologii analizy zagadnienia.

Computer simulation of an underwater explosion - ABAQUS Explicit code

Summary

The main results of parametric studies carried out by means of professional finite element codes on a structure subjected to explosions, are presented and comparatively discussed. Various approaches, assumptions and simulations aiming at the practical evaluation of structural damages are critically examined as for their conservativeness, accuracy and adequacy to the engineering purposes pursued.

Antoni Dziama *

Wojciech Wieleba *

WSPOMAGANE KOMPUTEROWO OBLICZENIA PRZEKŁADNI PASOWEJ Z PASEM PŁASKIM

1. Wstęp

Duży postęp w budowie przekładni pasowych powoduje, że są one obecnie dosyć często stosowane do przenoszenia średnich i małych mocy zarówno w napędzie głównym maszyn jak i w napędach zespołów pomocniczych. Przeniesienie ruchu i momentu obrotowego w tego typu przekładniach realizuje się przez sprzężenie cierne pasa z kołem. Duża podatność samego pasa oraz specyfika sprzężenia cierne umożliwiają łagodzenie gwałtownych zmian obciążenia i zabezpieczają zespoły układu napędowego i roboczego przed przeciążeniem i uszkodzeniem. Ruch przekazywany jest w sposób płynny i spokojny przez co uzyskuje się względnie cichą pracę. Podatne sprzężenie obu kół za pośrednictwem pasa skutecznie chroni przed przenikaniem drgań z układu napędzającego na napędzany i na odwrot. Dodatkowymi zaletami są: swoboda w rozstawie osi kół oraz prosta konstrukcja i łatwa eksploatacja. Niestety przekładnie pasowe posiadają również wady, które ograniczają ich zastosowanie. Do takich niekorzystnych cech należy zaliczyć niestalność przelżenia, wyciąganie się pasa w trakcie eksploatacji (konieczność regulacji naciągu), mała zwartość konstrukcji, wrażliwość na warunki otoczenia (temperatura, obecność olejów i smarów) oraz nieco niższa sprawność mechaniczna niż przekładni zębatych i łańcuchowych. Oprócz klasycznych pasów płaskich i klinowych stosuje się obecnie nowe rodzaje pasów tj. wieloklinowe i zębate. Pasy takie dobiera się na ogół według zaleceń podanych w katalogach producentów. Wypierają one w wielu przypadkach pasy płaskie. Jednak w sytuacji gdy potrzebny jest napęd kilku kół lub gdy osie kół nie są równoległe celowym będzie zastosowanie pasa płaskiego. Jednocześnie duża różnorodność nowych tworzyw na pasy, bardziej racjonalne kombinacje różnych materiałów w pasach wielowarstwowych, a także zbrojenie pasów linkami o dużej wytrzymałości umożliwiły zwiększenie obciążalności i trwałości pasów płaskich. Kształt pasów płaskich i kół jest prosty, są więc one tańsze od innych. W referacie przedstawiono metodę, algorytm i program komputerowy wspomagający obliczenia przekładni z pasem płaskim, przy uwzględnieniu trwałości zmęczeniowej pasa.

2. Algorytm obliczeń oraz omówienie programu komputerowego.

Obliczenia wytrzymałościowe napędu z pasem płaskim przeprowadza się najczęściej metodą współczynników uwzględniających wpływ czynników konstrukcyjnych i eksploatacyjnych (klasyczny sposób wg. Niemanna [2] i Pronina). Można też wykorzystywać wyniki zmęczeniowych badań pasów (Hagedorn [4]), na podstawie których sporządza się wykresy zmęczeniowe i wyznacza się wykładniki potęgowe linii Wöhlera. Przy opracowaniu algorytmu obliczeń oraz programu komputerowego korzystano z obu wymienionych metod. Zestawienie wzorów przedstawiono w tabeli 1.

* *Institut Konstrukcji i Eksploatacji Maszyn Politechniki Wrocławskiej*

Tabela 1. Zakres obliczeń i zestawienie wzorów

Lp.	Nazwa parametru	Wzór	Jednostki	Uwagi
1	Przełożenie zakładane	$i_z = \frac{n_1}{n_2}$	-	wg. założeń
2	Średnica koła małego	D_1 wg. wartości w p.2.1. i 2.2	mm	1)
2.1	z warunku $\left(\frac{D}{g}\right)_{\min}$	$D_1 \geq (1,5 + 2) \left(\frac{D}{g}\right)_{\min} g$	mm	$\left(\frac{D}{g}\right)_{\min}$ [3]
2.2	wg. zalecenia DIN 111	$D_1 \approx 6,6(d_M - 4)$	mm	
3	Średnica koła dużego	$D_2 = D_1 \cdot i_z$	mm	1)
4	Prędkość pasa	$v = \frac{\pi D_1 n_1}{60 \cdot 1000} \leq v_{\max}$	m/s	v_{\max} [3]
5	Zalecane ograniczenie prędkości	$v_{gr} = \sqrt{\frac{10^3 \sigma_{dop}}{3 \cdot \rho}}$	m/s	2)
5.1	Naprężenie	σ_{dop}	MPa	
5.2	Gęstość mater. pasa	ρ	Mg/m ³	
6	Odległość osi kół a	$a_{\min} \leq a \leq a_{\max}$	mm	
6.1	Odległość a_{\min}	$a_{\min} = 0,5(D_1 + D_2) + 20$	mm	
6.2	Odległość a_{\max}	$a_{\max} = 2(D_1 + D_2)$	mm	
7	Kąt opasania β_1	$\beta_1 = 2 \arccos \frac{D_2 - D_1}{2a}$	°	
8	Kąt opasania β_2	$\beta_2 = 360 - \beta_1$	°	
9	Długość pasa napiętego	$L_F = \pi D_b + (D_b - D_c) \operatorname{inv} \frac{\beta_1}{2}$	mm	
9.1	Średnica koła czynnego	$D_c = D_1 + g$	mm	
9.1	Średnica koła biernego	$D_b = D_2 + g$	mm	
9.3	Wzór przybliżony	$L_F = 2a + \frac{\pi}{2}(D_b + D_c) + \frac{(D_b - D_c)}{4a}$	mm	
9.4	Funkcja ewolwentowa	$\operatorname{inv} \frac{\beta_1}{2} = \operatorname{tg} \frac{\beta_1}{2} - \frac{\pi}{180} \cdot \frac{\beta_1}{2}$	-	β [°]
10	Szerokość pasa	$b = \frac{10^3 P}{v g} \cdot \frac{e^{\mu \beta}}{e^{\mu \beta} - 1} \cdot \frac{K \cdot K_\mu}{\sigma_{edop} - \sigma_g - \sigma_v}$	mm	
10.1	Współczynniki K, K_μ	wg. założeń	-	
10.2	Współczynnik tarcia μ		-	[3]
10.3	Naprężenie ekspl. dop.	$\sigma_{edop} = \sigma_{dop} \sqrt{\frac{U_o \xi_1 \xi_2}{U}}$	MPa	
10.4	Wymagana trwałość pasa	$U = 3600 \cdot T \frac{v^2}{L}$	liczba cykli obciążeń	
	Trwałość bazowa	$U_o = 10^7$	cykli obciążeń	
	Trwałość w godz.	T_b - zakładana	godz.	

Tabela 1. (c.d.) Zakres obliczeń i zestawienie wzorów

	Długość pasa	$L \approx L_F \cdot 10^3$	m	
	Prędkość pasa	v	m/s	
	Liczba kół	z	-	3)
	Wykładnik potęgowy krzywej zmęczeniowej dla pasa	p	-	[4]
	Współczynnik zginania pasa	$\xi_1 = f\left(\frac{\sigma_{2max}}{\sigma_{1max}}\right) = 1 \div 2$	-	
	Współczynnik zmienności obciążenia	$\xi_2 = 1 \div 2,5$	-	4)
10.5	Naprężenie gnące	$\sigma_g = E_g \frac{2y_o}{D_1}$	MPa	
	Moduł sprężystości przy zginaniu	E_g	MPa	[3]
	Odległość włókna skrajnego od osi obojętnej	zwykle $y_o = \frac{g}{2}$	mm	
10.6	Naprężenie wywołane siłą odśrodkową	$\sigma_v = \rho v^2 \cdot 10^3$	MPa	
11	Liczba obiegów pasa w sekundzie	$\frac{v}{L} < \left(\frac{v}{L}\right)_{max}$	s ⁻¹	[3]
12	Obliczeniowa trwałość pasa (eksploatacyjna)	$T_{he} = 2,78 \cdot 10^3 \left(\frac{\sigma_{dop}}{\sigma_{max}}\right)^p \cdot \frac{L}{v^2} \xi_1 \xi_2$	godz.	
12.1	Naprężenie maksymalne w pasie	$\sigma_{max} = \sigma_c + \sigma_g + \sigma_v$	MPa	
12.2	Naprężenie od siły czynnej	$\sigma_c = \frac{10^3 P}{vbg} \frac{e^{\mu\beta}}{e^{\mu\beta} - 1} K \cdot K_\mu$	MPa	
12.3	Naprężenia σ_g, σ_v	Wg. p. 10.5 i 10.6	MPa	
12.4	Siła czynna	$F_c = F_u \frac{e^{\mu\beta}}{e^{\mu\beta} - 1}$	N	
12.5	Siła użyteczna	$F_u = \frac{2T_1}{D_c} = \frac{2T_2}{D_b} = \frac{P \cdot 10^3}{v}$	N	
12.6	Siła bierna	$F_b = F_c - F_u$	N	
13	Siła naciągu wstępnego	$F_{m0} = 2F_o \cos \frac{\gamma}{2}$	N	
13.1	Siła w cięgnię w stanie spoczynku	$F_o = 0,5 F_u \frac{e^{\mu\beta} + 1}{e^{\mu\beta} - 1} + \rho v^2 bg$	N	
13.2	Kąt między cięgniem czynnym a biernym	$\gamma = 180^\circ - \beta_1$	°	
14	Długość pasa w stanie swobodnym	$l_s = l_f - \Delta l$	mm	
14.1	Przyrost długości pod wpływem sił	$\Delta l = \frac{\epsilon l}{100 + \epsilon}$	mm	

Tabela 1.(c.d.) Zakres obliczeń i zestawienie wzorów

14.2	Wydlużenie sprężyste względne	$\varepsilon = \frac{\sigma_o}{E} = \frac{F_o}{bgE}$	-	
	Moduł sprężystości przy rozciąganiu	E wg. mater. pasa	MPa	[3]
15	Poślizg sprężysty pasa	$\delta = \frac{v_c - v_b}{v_c} = \frac{\sigma_w}{E} = \frac{I_w}{bgE}$	-	
16	Przełożenie efektywne z uwzględnieniem poślizgu oraz prędkość obrotowa n_z	$i_e = \frac{D_b}{(1-\delta)D_c}$ $n_z = \frac{n_1}{i_e}$	-	⁵⁾
17	Siła poprzeczna obciążająca wał	$Q = \sqrt{F_c^2 + F_b^2 + 2F_c F_b \cos \gamma}$	N	
18	Kąt działania siły Q	$\theta = \arcsin\left(\frac{F_c - F_b}{F_c + F_b} \operatorname{tg} \gamma\right)$	°	
¹⁾ Przyjmować wartość znormalizowaną; ²⁾ Ograniczenie ze względu na siłę odśrodkową ³⁾ W układzie otwartym $z=2$. ⁴⁾ $\xi_z=1$ gdy pełne obciążenie występuje stale, $\xi_z>1$ w zależności od czasów pełnego i niepełnego obciążenia ⁵⁾ porównać z wartościami zakładanymi (wymaganymi)				

Opracowany program komputerowy znakomicie przyspiesza obliczenia przekładni. Zbudowany jest z poszczególnych bloków:

- wprowadzanie parametrów pasa,
- określenie warunków pracy przekładni,
- obliczanie przekładni,
- wydruk wyników.

W każdym momencie można przejść z jednego bloku do drugiego i przeprowadzić niezbędne zmiany wartości danych. Wprowadzanie danych w każdym bloku odbywa się w sposób edycyjny co ułatwia dokonywanie wszelkich zmian w danych. Bieżąca obserwacja podstawowych wyników obliczeń umożliwia łatwą kontrolę poprawności obliczeń oraz określanie wpływu wielkości wejściowych na gabaryty przekładni oraz siły i naprężenia w pasie.

Do zaprojektowania przekładni pasowej potrzebne są następujące dane i uzgodnienia wstępne:

- a) warunki pracy
- przełożenie - i
 - prędkość obrotowa wału wejściowego - n_1 [1/min]
 - moc przenoszona przez przekładnię - P [kW]
 - współczynnik przeciążenia - K
 - współczynnik wpływu środowiska - K_μ

- współczynnik zmienności obciążenia - ξ_2
- wymagana trwałość pasa - T [godz.]
- b) wielkości charakterystyczne materiału pasa
(dane dla najczęściej stosowanych materiałów patrz [3])
 - gęstość materiału - ρ [Mg/m^3]
 - granica wytrzymałości na rozciąganie - R_m [MPa]
 - moduł sprężystości przy rozciąganiu - E [MPa]
 - moduł sprężystości przy zginaniu - E_g [MPa]
 - naprężenia dopuszczalne - σ_{dop} [MPa]
 - współczynnik tarcia materiału po kole - μ
 - współczynniki eksploatacyjne - $(D/g)_{\min}$, $(v/L)_{\max}$
 - maksymalna prędkość pasa - v_{\max} [m/s]
 - wykładnik potęgowy linii Wöhlera - m.

Po podaniu założeń wstępnych przystępuje się do obliczania przekładni pasowej.

W tym celu należy podać parametry geometryczne przekładni takie jak:

- wymiary przekroju pasa (grubość - g [mm], szerokość b - [mm])
- średnice kół pasowych - D_1 , D_2 [mm]
- odległość między osiami - a [mm].

Podczas wprowadzania podanych wyżej parametrów na ekranie wyświetlane są związane z nimi ograniczenia. Należą do nich ograniczenia średnic kół pasowych ze względu na wymiary i materiał pasa, ograniczenia odległości między osiami, minimalna szerokość pasa ze względu na dopuszczalne naprężenia eksploatacyjne. Ponadto wyświetlane są wartości wielkości kontrolnych świadczące o poprawności wyników obliczeń, takie jak:

- liczba obiegów pasa - v/L ,
- przelożenie rzeczywiste uwzględniające poślizg,
- przewidywana trwałość pasa,
- siła naciągu wstępnego,
- siła poprzeczna obciążająca wał,
- maksymalne naprężenia w pasie,
- długość pasa w stanie swobodnym.

Zarówno wartości ograniczeń jak i wielkości kontrolne są aktualizowane podczas wprowadzania danych lub dokonywania ich zmian. Po zaakceptowaniu uzyskanych wyników na ekranie komputera wyświetlane są wartości wszystkich parametrów geometrycznych i eksploatacyjnych przekładni oraz siły i naprężenia w pasach. Wielkości te można również wydrukować. Przykładowy wydruk przedstawiono jako tabela 2.

3. Podsumowanie

W obecnej postaci program wykorzystywany jest do celów dydaktycznych. Możliwe jest również jego praktyczne wykorzystanie. Ułatwia on i przyspiesza obliczenia dzięki czemu konstruktor ma możliwość przeanalizowania większej ilości rozwiązań przekładni i trafniejszego wyboru najlepszego rozwiązania. Po niewielkiej modyfikacji programu polegającej na uzupełnieniu algorytmu o funkcje optymalizacyjne istnieje możliwość prowadzenia pełnej optymalizacji rozwiązań przekładni pasowej z pasem płaskim.

Tabela 2. Przykładowy wydruk wyników obliczeń programu komputerowego.

*** OBLICZENIA PRZEKŁADNI PASOWEJ Z PASEM PŁASKIM ***

Warunki pracy przekładni	Oznaczenie	Jednostka	Wartosc
1.Przełożenie	i		1.78
2.Predkosc obrotowa walca wejściowego	n1	1/min	1400
3.Moc przenoszona przez przekładnię	P	kW	13.0
4.Współczynnik przeciążenia	K		1.10
5.Współczynnik wpływu środowiska	K _s		1.00
6.Współczynnik zmienności obciążenia	K _{s12}		1.80
7.Przewidywana obliczona trwałość pasa	Trz	godz.	1618

Wielkości charakterystyczne pasa	Oznaczenie	Jednostka	Wartosc
Wymiary pasa :			
1.Grubość	g	mm	5.0
2.Szerokość	b	mm	90.0
3.Długość w stanie swobodnym	L _s	mm	2320.4
Charakterystyka materiału pasa			
1.Gęstość materiału pasa	q	Mg/m ³	0.95
2.Granica wytrzymałości na rozciąganie	R _m	MPa	29.0
3.Moduł sprężystości przy rozciąganiu	E	MPa	340.0
4.Moduł sprężystości przy zginaniu	E _g	MPa	50.0
5.Napieżenie dopuszczalne na zginanie	σ(dop)	MPa	4.3
6.Współczynnik tarcia pasa po kole	μ		0.45
7.Współczynnik I	(D/g)min		30.0
8.Współczynnik II	(V/L)max		10.0
9.Wykładnik potęgowy Wöhlera	n		8.0
10.Predkosc maksymalna pasa	V _{max}	m/s	40.0

Parametry przekładni	Oznaczenie	Jednostka	Wartosc
1.Srednica kola pasowego I	D1	mm	250.0
2.Srednica kola pasowego II	D2	mm	450.0
3.Odleglosc miedzy osiami	a	mm	600.0
4.Przełożenie rzeczywiste (obciąż.nom.)	i(rzecz)		1.792
5.Predkosc liniowa pasa	V	m/s	19.11
Sily :			
1.Sila uzyteczna	F _u	N	680.2
2.Sila czynna	F _c	N	948.4
3.Sila bierna	F _b	N	268.2
4.Sila naciagu wstepnego	F _{nw}	N	1507.6
5.Sila poprzeczna obciazajaca wal	Q	N	1205.0
6.Kat odchylenia sily Q	θ	°	5.4
7.Kat opasania na kole mniejszym	β1	°	160.8
Napiezienia :			
1.Dopuszczalne naprez. eksploatacyjne	σe(dop)	MPa	3.890
2.Napiezanie maksymalne w pasie	σ(max)	MPa	3.665
3.Napiezanie od sily czynnej	σc	MPa	2.318
4.Napiezanie od zginania pasa	σg	MPa	1.000
5.Napiezanie od sily odsrodkowej	σv	MPa	0.347

Literatura

1. Dietrych J., Korewa W., Kronberger Z., Zygmunt K.: *Podstawy konstrukcji maszyn*. cz.III, Warszawa, WNT 1970.
2. Niemann G., Winter H.: *Maschinenelemente*. B.III, Springer-Verlag, Berlin 1983.
3. *Podstawy konstrukcji maszyn*, t.IV, pod red. M Dietricha, PWN, Warszawa 1991.
4. Hagedorn H.: *Berechnungsprobleme bei Flachriementrieben*. Maschinenbautechnik 17 (1968).
5. *DIN 111 Flachriemenscheiben*. B1.1, Nenndrehmomente, B1.2, Zuordnung für elektrische Maschinen.

Streszczenie

Przekładnie pasowe są często stosowane w napędach mechanicznych. Istnieją pewne układy napędowe, w których możliwe jest zastosowanie jedynie pasów płaskich. W referacie przedstawiony został algorytm obliczeń przekładni pasowej z pasem płaskim wraz z zestawieniem odpowiednich wzorów. W oparciu o nie opracowano program komputerowy ułatwiający obliczenia tego typu przekładni. Zasygnalizowano również, które z wielkości są niezbędne do przeprowadzenia obliczeń z wykorzystaniem programu komputerowego oraz jakie wielkości uzyskuje się jako wynik.

COMPUTER AIDED COMPUTING OF FLAT BELT TRANSMISSIONS

Summary

The algorithm of the computing of flat belt transmissions and the set of necessary formulas were presented in this paper. The computer program, aiding computing that kind of transmissions was developed on the basis of this algorithm. Necessary input quantities for computing with the computer program and the output quantities were also presented.

Sztuczne Sieci Neuronowe Rekurencyjne do Modelowania Procesów Dynamicznych

1. Wprowadzenie

W literaturze możemy spotkać się z trzema głównymi nurtami zastosowań modeli w technice: badania naukowe, projektowanie urządzeń technicznych oraz sterowanie. W badaniach naukowych modele wykorzystuje się do zdobycia nowych informacji o obiekcie. W projektowaniu modele umożliwiają wariantowanie konstrukcji i ich wielokryterialną optymalizację. Bardzo szeroko stosowane są modele w sterowaniu. Dają one możliwość doboru sterowań dla złożonych obiektów, szczególnie w sterowaniu adaptacyjnym i optymalnym (np. do określenia zachowania obiektu na granicach stabilności). W niniejszej pracy poruszymy problem wykorzystania modelu w procesie uczenia regulatorów neuronowych dla procesów nieliniowych i strukturalnie niestabilnych.

Identyfikacja a także modelowanie procesów dynamicznych, szczególnie nieliniowych, jest trudne, pracochłonne i nie zawsze daje zadowalające wyniki. Ostatnio pojawiło się nowe ujęcie modelowania takich obiektów z wykorzystaniem Sztucznych Sieci Neuronowych (ANN-Artificial Neural Network) [1-5]. Do modelowania wykorzystywane są wszystkie typy tych sieci:

- klasyczne, bez sprzężenia zwrotnego (feedforward), o charakterze statycznym, w których modelowanie procesów dynamicznych osiąga się przez wprowadzenie na wejście sieci (w etapie jej uczenia) kilku poprzednich wartości sygnałów wejściowych i wyjściowych;

- rekurencyjne (np. sieć Hopfielda), ze wszystkimi wagami (parametrami sieci) modyfikowanymi w procesie uczenia;

- częściowo rekurencyjne (np. Elmana), o modyfikowanych wagach przy przepływie sygnałów 'do przodu' i niemodyfikowanych przy połączeniach sprzężenia zwrotnego.

Dotychczasowe eksperymenty wykazują [5], że biorąc pod uwagę rozmiary sieci, czas treningu i uzyskiwane wyniki (dokładność modelu), najlepsze rezultaty uzyskuje się korzystając z sieci neuronowych częściowo rekurencyjnych.

W tym artykule przedstawiono wykorzystanie zmodyfikowanej sieci Jordana o strukturze częściowo rekurencyjnej z algorytmem uczenia wstecznej propagacji błędów (BP-Back Propagation) [1,6,7,8,9]. W punkcie drugim omówiono klasyczną sieć Jordana oraz cel i sposób modyfikacji sieci zaproponowany przez Pham i Oh [4]. W punkcie trzecim przedstawiono sposób wykorzystania zmodyfikowanej sieci Jordana do identyfikacji procesu dynamicznego. W punkcie czwartym opisano przeprowadzony

¹Wydział Mechaniczny Politechniki Gdańskiej

e-mail sgrymek@sunrise.pg.gda.pl

²Wydział Mechaniczny WSI Koszalin

e-mail wtarnows@sunrise.pg.gda.pl

eksperyment polegający na zamodelowaniu obiektu silnie nieliniowego i przedstawiono uzyskane wyniki.

2. Zmodyfikowana sieć Jordana

Jedną z szerzej znanych rekurencyjnych sieci neuronowych jest sieć Jordana (Rys. 1). Sieć ta jest układem o jednym wejściu i jednym wyjściu (typu SISO: Single Input-Single Output). Strukturę wewnętrzną sieci tworzą dwie warstwy o jednakowej liczbie neuronów: warstwa ukryta (hidden layer) i warstwa stanu (state layer, context layer). W sieci występują dwa rodzaje połączeń:

- z wagami modyfikowanymi w procesie uczenia (linie ciągłe na Rys.1),
- sprzężenia zwrotne z wagami niemodyfikowanymi (linie przerywane).

Działanie sieci Jordana dla neuronów z liniowymi funkcjami przejścia opisuje układ trzech równań:

$$X_S(k) = c_O \cdot N \cdot y(k) + c_S \cdot M \cdot X_S(k-1) \quad (1)$$

$$X_H(k+1) = W_H \cdot X_S + W_I \cdot u(k) \quad (2)$$

$$y(k+1) = W_O \cdot X_H(k+1) \quad (3)$$

gdzie: k - kolejny krok,

u - sygnał wejściowy,

X_S - wektor wyjścia z warstwy stanu,

X_H - wektor wyjścia z warstwy ukrytej,

y - sygnał wyjściowy,

W_I - wektor wag pomiędzy wejściem a warstwą ukrytą,

W_H - macierz wag pomiędzy warstwami: stanu i ukrytą,

W_O - wektor wag pomiędzy warstwą ukrytą a wyjściem,

c_O - współczynnik sprzężenia zwrotnego,

c_S - współczynnik autoasocjacji warstwy stanu,

M - macierz kwadratowa jedynek,

N - wektor jedynek.

Analizując powyższe równania widać ich podobieństwo do opisu obiektu dynamicznego w przestrzeni stanów [10]. Pojawia się pytanie, czy sieć ta może symulować dowolny obiekt dynamiczny. Z równań (1)-(3) wyprowadzono zależność

$$X_S(k) = [c_O \cdot M \cdot W_O \cdot W_H + c_S \cdot N] \cdot X_S(k-1) + c_O \cdot M \cdot W_O \cdot W_I \cdot u(k-1) \quad (4)$$

Zauważmy, że wymnożenie wektorów M, W_O, W_I daje wektor kolumnowy o jednakowych elementach. Sugeruje to, że sygnał wejściowy w identyczny sposób oddziałuje na wszystkie zmienne stanu. Z tego wynika, że sieć w takiej postaci nie jest zdolna do zamodelowania dowolnego układu dynamicznego.

Phan i Oh [4] zaproponowali modyfikację usuwającą ten problem. Modyfikacja polega na dodaniu sprzężenia zwrotnego pomiędzy warstwą ukrytą a warstwą stanu. Nowo powstała sieć (Rys. 2) jest w rzeczywistości kompilacją sieci Jordana z siecią Elmana [5].

Po modyfikacji drugiego członu równanie (4) przyjęło postać:

$$X_S(k) = A \cdot X_S(k-1) + B \cdot u(k-1) \quad (5)$$

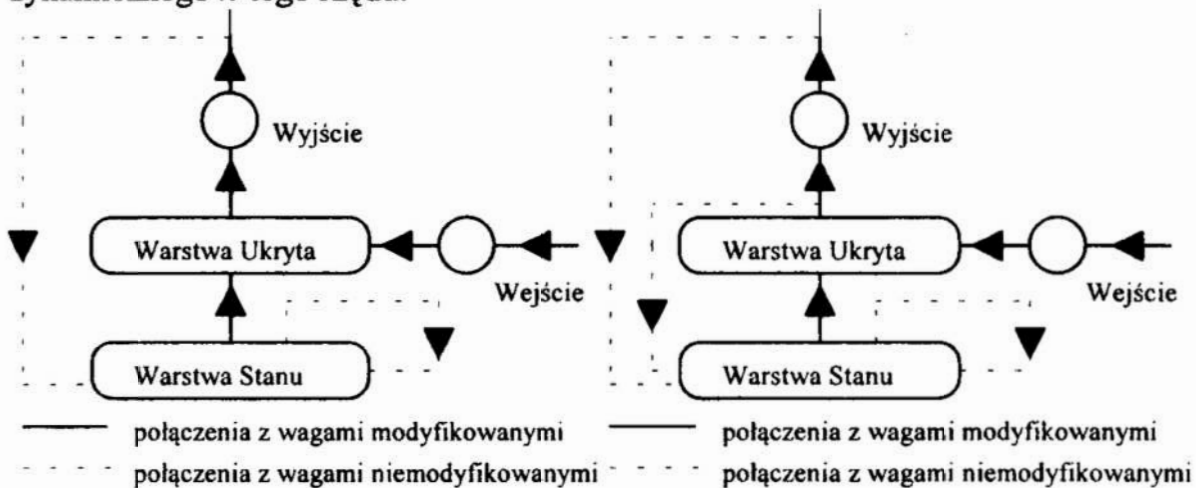
gdzie: $A = c_O \cdot M \cdot W_O \cdot W_H + c_S \cdot N + c_H \cdot W_H$ (6)

$$B = c_O \cdot M \cdot W_O \cdot W_H + c_H \cdot W_I \quad (7)$$

c_H - współczynnik sprzężenia pomiędzy warstwami: ukrytą i stanu.

Jak wynika z równania (7) modyfikacja sieci zlikwidowała identyczność wpływu sygnału wejściowego na poszczególne zmienne stanu.

Po tej analizie można przewidywać, że zmodyfikowana sieć neuronowa Jordana o n neuronach w warstwie stanu jest zdolna do zamodelowania dowolnego obiektu dynamicznego n -tego rzędu.



Rys. 1. Klasyczna sieć Jordana
Fig. 1. Jordan neural network

Rys. 2. Zmodyfikowana sieć Jordana
Fig. 2. Modified Jordan neural network

3. Metodologia wykorzystania zmodyfikowanej sieci Jordana

W celu wykorzystania sieci neuronowej do modelowania należy w etapie wstępnym określić strukturę i parametry sieci, algorytm i parametry uczenia, warunki początkowe i kryterium oceny procesu uczenia oraz warunki jego zakończenia.

3.1. Struktura i parametry sieci

Ogólna struktura zmodyfikowanej sieci Jordana została już pokazana na rysunku 2. Nie określono jednak rodzaju wykorzystywanych neuronów oraz ich liczby w warstwie stanu i w warstwie ukrytej.

Omawiając podstawy teoretyczne określono, że liczba neuronów w warstwie stanu musi być równa liczbie neuronów w warstwie ukrytej. Stwierdzono również, że liczba neuronów w obu warstwach musi być równa dynamicznemu rzędowi obiektu. W rzeczywistych realizacjach sieci stosuje się liczbę większą niż rząd obiektu. Phan i Oh [4] sugerują stosowanie od 6 do 12 neuronów w zależności od stopnia i charakteru (liniowy-nieliniowy) obiektu.

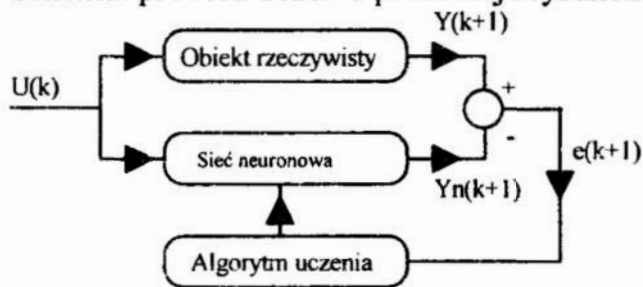
Dla obiektów liniowych wszystkie neurony sieci mogą mieć liniowe funkcje przejścia; dla obiektów nieliniowych neurony w warstwie ukrytej powinny mieć nieliniowe funkcje przejścia. Najczęściej używanymi nieliniowymi funkcjami przejścia są: funkcja logistyczna (sigmoid), tangens hiperboliczny i sinus [9].

Współczynniki wag sprzężeń zwrotnych ustalane są na etapie projektowania sieci. Połączenia wyjścia z warstwą stanu i warstwy ukrytej z warstwą stanu nie są ważone tzn. odpowiednie współczynniki mają wartość 1. Współczynnik autoasocjacji warstwy stanu powinien osłabiać w niewielkim stopniu sygnał. Proponowany zakres to 0.8-0.9 [4].

3.2. Algorytm uczenia

Algorytm stosowany do treningu (uczenia) sieci Jordana to Wsteczna Propagacja (BP-Back Propagation) zwany też Uogólnioną Regułą Delta (Generalized Delta Rule) [7,8]. Sam proces uczenia jest procesem adaptacyjnym (Adaptive Learning), to znaczy wagi są modyfikowane po każdej prezentacji.

Schemat procesu uczenia prezentuje rysunek 3.



- $U(k)$ - sygnał sterujący w chwili k
- $Y(k+1)$ - wyjście obiektu w chwili $k+1$
- $Y_n(k+1)$ - wyjście sieci neuronowej w chwili $k+1$
- $e(k+1)$ - błąd w chwili $k+1$

Rys. 3. Schemat procesu modelowania
Fig. 3. Modelling process

sinusoidalne [2,3,4,5].

Algorytm Wstecznej Propagacji wymaga podania tzw. Współczynnika Uczącego (Learning Rate). Współczynnik ten zawsze powinien być mniejszy od 1. Dla procesu uczenia adaptacyjnego powinien być dodatkowo zmniejszony. Proponowany zakres to 0.1 do 0.001 [4,6,8].

Podczas uczenia sieci wielowarstwowych (do tej klasy należy sieć Jordana) istnieje niebezpieczeństwo wpadania w minima lokalne. Niebezpieczeństwa tego unika się stosując technikę utrzymywania kierunku modyfikacji (Momentum Term). Polega ona na dodawaniu do bieżąco określonej zmiany wagi ułamka wartości poprzednio wyliczonej zmiany. Ułamek ten jest zależny od charakteru obiektu i wartości współczynnika uczącego. Proponowany zakres wynosi od 0.1 do 0.01.

3.3. Warunki początkowe

Przed rozpoczęciem procesu uczenia należy nadać wartości początkowe wszystkim modyfikowanym wagom oraz wartości początkowe sygnałom wyjściowym warstwy stanu.

Wagom nadaje się niewielkie losowe wartości. Należy przestrzegać, aby wagi nie miały jednakowych wartości początkowych.

Sygnał sterujący uczący jest podawany jednocześnie na wejście obiektu i na wejście sieci neuronowej. Otrzymane odpowiedzi służą do określenia błędu modelu. Wartość ta, po przetransformowaniu przez algorytm uczący, służy do modyfikacji wag.

Wagi modyfikuje się po każdym kroku procesu uczenia. Taki proces znany jest jako uczenie adaptacyjne (Adaptive Learning) [3].

Jako sygnał sterujący uczący stosuje się sygnały losowe, pseudolosowe lub

Wartość sygnałów wyjściowych warstwy stanu w chwili początkowej określa się jako równą średniej możliwych do wystąpienia wartości. Wartością tą jest 0 dla funkcji liniowej i tangensa hiperbolicznego i 0.5 dla funkcji logistycznej [5].

3.4. Warunki zakończenia procesu uczenia

Istnieją dwa kryteria określenia momentu zakończenia procesu uczenia:

- ☐ określenie maksymalnej liczby prezentacji zbioru uczącego (liczby iteracji);
- ☐ określenie maksymalnego błędu procesu uczenia.

O ile pierwsze kryterium nie wymaga komentarza, o tyle w drugim aktualny błąd procesu uczenia może być określony w różny sposób. Może być to błąd bezwzględny lub kwadratowy, normalizowany lub nie, maksymalny lub średni, liczony dla określonej liczby iteracji lub skumulowany od początku procesu uczenia.

Dla adaptacyjnego procesu uczenia proponuje się stosowanie błędu wyrażonego wzorem (8):

$$E(k) = \frac{1}{k} \sum_{j=1}^k [y(j) - y_n(j)]^2 \quad (8)$$

3.5. Kryteria oceny modelu neuronowego

Dla oceny procesu uczenia i dokładności uzyskanego modelu należy określić kryterium pozwalające porównać wyniki uzyskane dla obiektów o różnych wartościach sygnałów wejściowych i różnych modeli uzyskanych różnymi metodami.

Kryterium takim może być znormalizowany średni błąd kwadratowy (po odcięciu składowej stałej) w testowym okresie czasu T wyrażony wzorem (9):

$$E_N = \frac{\int_0^T (y(t) - y_n(t))^2 dt}{\int_0^T (y(t))^2 dt} \quad (9)$$

Aby uniknąć całkowania wzór (9) można zastąpić przybliżoną formułą:

$$E_N = \frac{\sum_0^N (y(k) - y_n(k))^2}{\sum_0^N (y(k))^2} \quad (10)$$

gdzie: N - liczba prób w czasie T.

4. Przykład modelowania obiektu nieliniowego

Jako przykład zastosowania częściowo rekurencyjnej zmodyfikowanej sztucznej sieci neuronowej Jordana zaprezentowano proces modelowania obiektu nieliniowego. Modelowany obiekt opisany jest iteracyjnym dyskretnym równaniem wejścia-wyjścia:

$$y(k+1) = \frac{y(k)}{1.5 + y^2(k)} - 0.3y(k-1) + 0.5\sin(u(k)) \quad (11)$$

4.1. Projekt sieci neuronowej

Do zamodelowania przedstawionego powyżej obiektu zaproponowano zmodyfikowaną sieć Jordana o następujących parametrach:

liczba neuronów w warstwie stanu i ukrytej	10;
funkcja przejścia neuronów warstwy ukrytej	tangens hiperboliczny;
współczynnik autoasocjacji c_s	0.8
współczynnik c_H	1;
współczynnik c_O	1;
wartość początkowa warstwy stanu	0.

Wartości początkowe wag połączeń pomiędzy warstwą stanu i warstwą ukrytą ustawiono na wartości losowe z zakresu od -0.5 do 0.5; dla wag pomiędzy warstwą ukrytą a wyjściem zakres zmniejszono do wartości od -0.2 do 0.2.

Symulator zaprojektowanej sieci neuronowej oraz procedury procesu uczenia napisane zostały przez autorów w języku Borland Pascal firmy Borland International Inc. i zaimplementowane na mikrokomputerze PC AT386.

4.2. Proces uczenia

Dla obiektu opisanego równaniem (11) i zaprojektowanej sieci przeprowadzono proces uczenia wykorzystując losowy sygnał wejściowy o rozkładzie prostokątnym z zakresu od -1 do 1 posługując się generatorem losowym języka Borland Pascal.

Proces uczenia zakończono po 700 tysiącach iteracji uzyskując błąd o wartości 0.0146 liczony według wzoru (8).

4.3. Uzyskane wyniki

Testy przeprowadzono dla trzech sygnałów testujących:

skoku jednostkowego	$u(k) = 1;$
wymuszenia losowego	$u(k) = \text{Random}(-1,1);$
wymuszenia sinusoidalnego	$u(k) = \sin(2 \cdot \pi \cdot 0.03 \cdot k).$

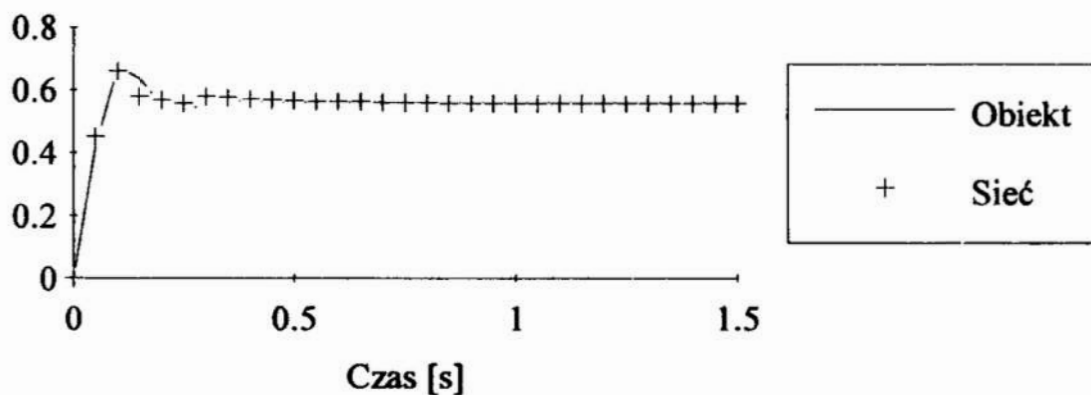
Błąd znormalizowany obliczano według wzoru (10) dla $N = 399$.

Uzyskane wyniki przedstawia poniższa tabela.

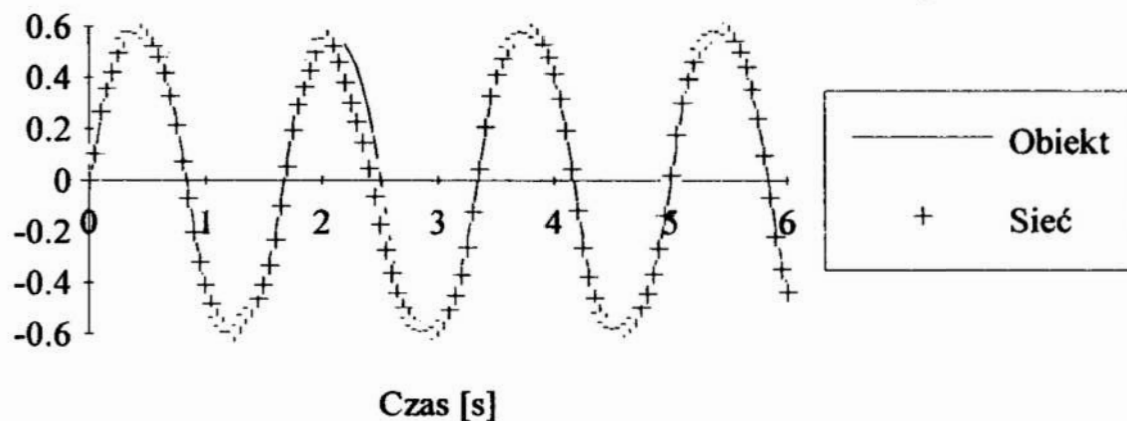
Wymuszenie	Błąd wg (11)
Skok jednostkowy	0.00014
Sinusoidalne	0.00702
Losowe	0.02214

Poszczególne wykresy prezentują przebiegi wartości odpowiedzi obiektu (linia ciągła) i modelu neuronowego (krzyżyki) w funkcji czasu.

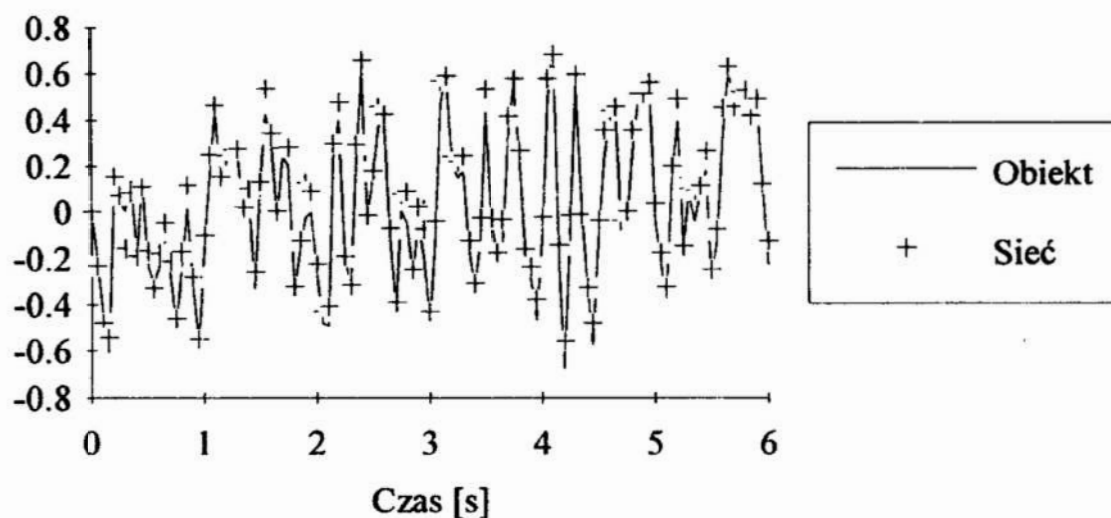
Skok jednostkowy



Wymuszenie sinusoidalne



Wymuszenie losowe



Rys. 4. Przykładowe odpowiedzi obiektu (linia ciągła) i modelu neuronowego (krzyżyki) w funkcji czasu.

Fig. 4. Exemplar response characteristics of the object (continuous line) and of the neural model (crosses)

5. Wnioski

Przeprowadzone eksperymenty potwierdzają teoretyczne rozważania co do zastosowania Sztucznych Sieci Neuronowych Częściowo Rekurencyjnych do identyfikacji i modelowania obiektów dynamicznych liniowych i nieliniowych. Uzyskane przez autorów wyniki nie odbiegają w znaczący sposób od wyników uzyskanych przez innych badaczy [4,5]. Obserwacje autorów potwierdzają również wnioski Phama i Oha dotyczące parametrów sieci i procesu uczenia, które zostały wykorzystane przy omawianiu metodologii wykorzystania sieci Jordana do modelowania w punkcie 3, chociaż liczba iteracji w procesie uczenia prowadzonym przez autorów była z reguły 20 do 30% większa.

Na bazie doświadczeń autorów zebranych w czasie eksperymentów wnioski te można wzbogacić.

1. Zakres wartości początkowych wag połączeń pomiędzy warstwą stanu i warstwą ukrytą powinien mieć niewielkie wartości losowe, natomiast zakres wartości początkowych wag połączeń pomiędzy warstwą ukrytą i wyjściem zależy od nominalnej wartości występujących sygnałów wyjściowych. Wynika to z funkcji skalującej tych wag. Proponowany orientacyjny zakres można wyrazić jako maksymalną wartość sygnału wyjściowego podzieloną przez ilość neuronów w warstwie ukrytej. Jednak, gdy wartość ta jest większa od 1 należy znormalizować sygnał wyjściowy. Niespełnienie tego warunku prowadzi często do problemów numerycznych podczas implementacji procesu uczenia (overflow).
2. Liczba neuronów w warstwie stanu i w warstwie musi być conajmniej równa rzędowi dynamiczemu obiektu. W rzeczywistości powinna być ona większa. Odpowiednią liczbę można określić przeprowadzając przebiegi próbne, przy ustalonych pozostałych parametrach, obserwując zbieżność procesu uczenia. Dla obiektów liniowych niskiego rzędu warto zacząć od liczby trzy razy większej od rzędu obiektu, a dla wyższego - dwa razy. Dla obiektów nieliniowych można przyjąć zasadę, że im bardziej różnorodne formy nieliniowości przewidywane są w obiekcie tym liczba neuronów powinna być większa. Autorzy nawet przy bardzo skomplikowanych obiektach nie brali nigdy liczby większej niż 15.
3. Stosując zmodyfikowaną sieć Jordana, mimo wielokrotnie prowadzonych eksperymentów, nie uzyskano zadawalających wyników modelowania w przypadku obiektów z silną nieciągłością (np. przekaźnik z histerezą).
4. Bardzo istotną kwestią jest przebieg uczącego sygnału wejściowego. Jeżeli model ma być ważny w pełnym zakresie częstotliwości, to najlepsze efekty daje zastosowanie w pełni losowego sygnału o rozkładzie prostokątnym. Dla uzyskania modelu w pewnym zakresie częstotliwości dobre rezultaty daje zastosowanie sygnału o losowych przyrostach, których ekstremalne wartości wynikają z określenia pasma częstotliwości lub takiego sygnału nałożonego na sinusoidę. Zdecydowanie gorsze efekty uzyskano stosując sygnał o charakterze sinusoidalnym.

6. Bibliografia

- [1] Müller B., Reinhardt J.; "Neural Networks. An Introduction", Springer-Verlag Berlin-Heidelberg-New York London-Paris-Tokyo-Hong Kong-Barcelona-Budapest;
- [2] Thibault J., Grandjean B.P.A., "Process Control Using Feedforward Neural Networks", Journal of Systems Engineering vol.2, London 1992;
- [3] Yabuta T., Yamada T., "Learning Control Using Neural Networks", Journal of Systems Engineering vol.2, London 1992;
- [4] Pham D.T., Oh S.J., "A Recurrent Backpropagation Neural Network for Dynamic System Identification", Journal of Systems Engineering vol.2, London 1992;
- [5] Pham D.T., Liu X., "Dynamic System Modelling Using Partially Recurrent Neural Networks", Journal of Systems Engineering vol.2, London 1992;
- [6] Beale R., Jackson T.; "Neural Computing: An Introduction", Adam Hilger Bristol-Philadelphia-New York;
- [7] Caudill M., "Neural Network Primer, Part III, Backpropagation Network", AI EXPERT, June 1988;
- [8] Rumelhart D.E., Hinton G.E., Williams R.J., "Learning Internal Representations by Error Propagation", in: "Parallel distributed processing", MIT Press Cambridge 1986;
- [9] Markowski A., Kostro J., Lewandowski A., "AUTOMATYKA w pytaniach i odpowiedziach", WNT Warszawa 1985;
- [10] Giergiel J., Uhl T., "Identyfikacja układów mechanicznych", PWN Warszawa 1990.

Streszczenie

Artykuł przedstawia zastosowanie Sztucznych Sieci Neuronowych w procesie identyfikacji i modelowania obiektów dynamicznych. W części teoretycznej omówiono zasadę działania Rekurencyjnej Sieci Neuronowej Jordana, jej zmodyfikowaną przez Pham'a i Oh'a wersję oraz metodologię wykorzystania tej sieci w modelowaniu i identyfikacji. Rozważania teoretyczne zilustrowano przykładem modelowania obiektu silnie nieliniowego. W zakończeniu przedstawiono wnioski z własnych prac nad zastosowaniem zmodyfikowanej sieci Jordana w modelowaniu i ustosunkowano się do wyników uzyskanych przez Phama i Oha.

Dynamic processes modelling by means of recurrent neural networks

Summary

Modelling of non-linear dynamic plants by means of the Artificial Neural Network is presented. It is shown, how the partially-recurrent modified Jordan network is built and it works. The methodology of such kind of modelling is discussed. Finally, an example of dynamic non-linear plant modelling process is demonstrated, as well as a comparison of the proposed net with that of Pham & Oh is given.

GRUPOWE WIELOKRYTERIALNE PODEJMOWANIE DECYZJI W OPARCIU O METODĘ PORÓWNYWANIA PARAMI

1. Wprowadzenie

Metoda hierarchicznego procesu decyzyjnego AHP (ang. Analytic Hierarchy Process) została wprowadzona przez Saaty'ego w roku 1980 [9]. Służy ona do wykonywania rankingu skończonej liczby obiektów poprzez porównywanie ich parami, korzystając ze skali: $S = \{1/9, \dots, 1/2, 1, 2, \dots, 9\}$. Ekspert (lub osoba podejmująca decyzję), każdej parze obiektów w sposób subiektywny, przyporządkowuje liczbę ze zbioru S . Załóżmy, że mamy n obiektów: F_1, F_2, \dots, F_n . Każdej parze obiektów (F_i, F_j) , $i, j = 1, \dots, n$ przyporządkowana jest liczba $r_{ij} \in S$ zgodnie z subiektywnymi preferencjami danego eksperta (patrz np. [9]). Następnie wyniki umieszczane są w tzw. macierzy ocen R :

$$R = \begin{pmatrix} r_{11} & r_{12} & \dots & r_{1n} \\ r_{21} & r_{22} & \dots & r_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ r_{n1} & r_{n2} & \dots & r_{nn} \end{pmatrix}. \quad (1)$$

Koncepcja Saaty'ego polega na przybliżeniu macierzy ocen R za pomocą następującej macierzy ilorazów:

$$S = \begin{pmatrix} \alpha_1/\alpha_1 & \alpha_1/\alpha_2 & \dots & \alpha_1/\alpha_n \\ \alpha_2/\alpha_1 & \alpha_2/\alpha_2 & \dots & \alpha_2/\alpha_n \\ \vdots & \vdots & \vdots & \vdots \\ \alpha_n/\alpha_1 & \alpha_n/\alpha_2 & \dots & \alpha_n/\alpha_n \end{pmatrix}. \quad (2)$$

Innymi słowy macierz R , utworzona przez eksperta jest macierzą z niezgodnymi ocenami. Naszym zadaniem jest znalezienie macierzy S z ocenami zgodnymi, które przedstawione są w postaci ilorazów $s_{ij} = \alpha_i/\alpha_j$, $i, j = 1, 2, \dots, n$. Otrzymując macierz S wyznaczamy równocześnie wektor rozwiązania rozważanego problemu, a mianowicie:

$$s = (\alpha_1, \dots, \alpha_n)^T. \quad (3)$$

Dokonując normalizacji wektora s otrzymujemy wektor:

$$s^* = (\alpha^*_1, \dots, \alpha^*_n)^T. \quad (4)$$

gdzie:

$$\alpha^*_i = \alpha_i / \sum_{i=1}^n \alpha_i, \quad i = 1, \dots, n. \quad (5)$$

W dalszych rozważaniach pomija się indeks "*" oraz zakłada, że wektor s jest znormalizowany, zgodnie z równaniami (4)-(5).

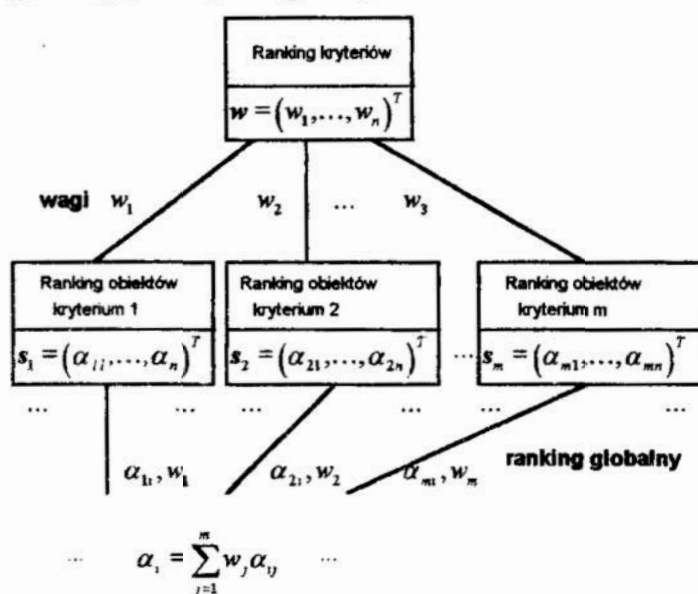
W celu znalezienia wektora s stosowane są zwykle trzy metody: metoda maksymalnej wartości własnej [9,10], metoda najmniejszych kwadratów [10,3] oraz metoda logarytmicznych najmniejszych kwadratów [10,3,1,2]. Analizę i porównanie powyższych metod można znaleźć m.in. w pracach [3,10]. W dalszej części pracy stosuje się metodę logarytmicznych najmniejszych kwadratów.

2. Sformułowanie problemu wielokryterialnego podejmowania decyzji w kategoriach podejścia AHP.

Założmy, że w rozważanym procesie podejmowania decyzji mamy n obiektów: F_1, F_2, \dots, F_n . Niech naszym zadaniem będzie porównanie ich pod względem m kryteriów C_1, C_2, \dots, C_m przez D ekspertów. Nasze zagadnienie podejmowania decyzji można zdekomponować na następujące podproblemy: ranking kryteriów oraz ranking obiektów pod względem kryterium i , $i = 1, 2, \dots, m$. Ponadto założmy, że stosując metodykę opisaną w rozdziale 1 otrzymaliśmy następujący, znormalizowany wektor wag dla kryteriów: $w = (w_1, \dots, w_n)^T$, oraz następujące wyniki rankingów obiektów odpowiednio pod względem kryterium C_i , $i = 1, 2, \dots, m$: $s_i = (\alpha_{i1}, \dots, \alpha_{in})^T$. Wówczas zgodnie z AHP otrzymujemy następujący ranking globalny (patrz rys. 1):

$$\alpha_i = \sum_{j=1}^m w_j \alpha_{ij}, \quad i = 1, 2, \dots, n. \quad (6)$$

Jak widać problem wielokryterialnego podejmowania decyzji może być zdekomponowany na odpowiednie podproblemy. Podproblemy te rozwiązuje się niezależnie, stosując metodę porównywania parami, a następnie dokonuje się agregacji, stosując równanie (6). Ze względu na powyższe, w dalszej części pracy omówione zostanie zagadnienie rankingu względem jednego kryterium.



Rys. 1. Hierarchiczny proces decyzyjny
Fig. 1. Analytic Hierarchy Process

1. Rozwiązanie problemu rankingu obiektów lub kryteriów.

Załóżmy, że rozważamy ranking n obiektów (kryteriów) F_1, F_2, \dots, F_n oraz posiadamy macierz $R = (r_{ijk})$, gdzie r_{ijk} jest liczbą wyrażającą relację dotyczącą pary obiektów (F_i, F_j) , przedstawioną przez k -tego eksperta, zgodną z przyjętą skalą. Zakładamy również, że mamy D ekspertów i w szczególności ekspert może odmówić oceny pary lub par obiektów. W efekcie mamy następującą zależność: $k = 0, 1, 2, \dots, d_{ij} \leq D \quad \forall i, j = 1, 2, \dots, n$, gdzie d_{ij} oznacza liczbę ekspertów, którzy lokowali oceny pary (ij) . W szczególnym przypadku może zachodzić $d_{ij} = 0$ (patrz przykład poniżej). Ponadto mamy [9,10]: $r_{ijk} > 0, r_{ijk} = 1/r_{jik} \quad \forall i, j, k$. Naszym zadaniem jest znalezienie macierzy ilorazów (2). Opierając się na powyższej koncepcji, metoda logarytmicznych najmniejszych kwadratów pozwala na wyznaczenie wektora $s = (\alpha_1, \alpha_2, \dots, \alpha_n)^T$, który minimalizuje następujące wyrażenie:

$$\sum_{i,j} \sum_{k=1}^{d_{ij}} \left(\ln(r_{ijk}) - \ln(\alpha_i/\alpha_j) \right)^2 \quad (7)$$

Warto zauważyć, że macierz R w przypadku wielu ekspertów i w przypadku ogólnym, kiedy brakuje pewnych ocen dla pewnych par (w szczególności może wystąpić kompletny brak ocen dla pewnych par), w sposób nieformalny z punktu widzenia matematycznego może być przedstawiona następująco:

$$R = \begin{pmatrix} r_{11,1} & r_{12,1} & \dots & r_{1n,1} \\ \vdots & \vdots & \vdots & \vdots \\ r_{11,d_{11}} & r_{12,d_{12}} & \dots & r_{1n,d_{1n}} \\ r_{21,1} & r_{22,1} & \dots & r_{2n,1} \\ \vdots & \vdots & \vdots & \vdots \\ r_{21,d_{21}} & r_{22,d_{22}} & \dots & r_{2n,d_{2n}} \\ \vdots & \vdots & \vdots & \vdots \\ r_{n1,1} & r_{n2,1} & \dots & r_{nn,1} \\ \vdots & \vdots & \vdots & \vdots \\ r_{n1,d_{n1}} & r_{n2,d_{n2}} & \dots & r_{nn,d_{nn}} \end{pmatrix}$$

Na przykład dla dwóch ekspertów mamy:

$$\begin{pmatrix} 1 & 2 & 5 \\ 1 & - & 1/5 \\ \\ 1/2 & 1 & - \\ - & 1 & - \\ \\ 1/5 & - & 1 \\ 5 & - & 1 \end{pmatrix}$$

Warto zauważyć brakujące oceny dla obiektów (1,2) oraz zupełny brak ocen dla obiektów (2,3). W tym ostatnim przypadku $d_{23} = d_{32} = 0$. Dokonując następujących podstawień: $x_i = \ln(\alpha_i)$ oraz $y_{ijk} = \ln(r_{ijk}) \quad \forall i, j, k$, nasz problem (7) przyjmie postać:

$$\min_{x_i, x_j} \sum_{i < j} \sum_{k=1}^{d_{ij}} (y_{ijk} - x_i + x_j)^2 \quad (8)$$

Po rozwiązaniu (8) ze względu na $\mathbf{x} = (x_1, \dots, x_n)^T$ otrzymujemy tzw. układ równań normalnych [1]:

$$x_i \sum_{i \neq j, j=1}^n d_{ij} - \sum_{i \neq j, j=1}^n d_{ij} x_j = \sum_{i \neq j, j=1}^n \sum_{k=1}^n y_{ijk}, \quad i = 1, 2, \dots, n, \quad (9)$$

gdzie:

$$d_{ij} \geq 0, \quad \forall i, j, \quad (10)$$

$$\sum_{i \neq j, j=1}^n d_{ij} > 0, \quad \forall i. \quad (11)$$

Układ równań (9) można przedstawić w postaci macierzowej:

$$\mathbf{Ax} = \mathbf{b}, \quad (12)$$

gdzie wektor prawych stron posiada następujące współczynniki: $b_i = \sum_{i \neq j, j=1}^n \sum_{k=1}^n y_{ijk}$,

$i = 1, 2, \dots, n$, natomiast macierz A posiada następującą strukturę:

$$A = \begin{pmatrix} \sum_{j=1, j \neq 1}^n d_{1j} & -d_{12} & \dots & -d_{1n} \\ -d_{21} & \sum_{j=1, j \neq 2}^n d_{2j} & \dots & -d_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ -d_{n1} & -d_{n2} & \dots & \sum_{j=1, j \neq n}^n d_{nj} \end{pmatrix},$$

a w szczególności dla przykładu ze strony 3:

$$\begin{pmatrix} 3 & -1 & -2 \\ -1 & 1 & 0 \\ -2 & 0 & 2 \end{pmatrix}.$$

Analiza własności macierzy A , które mają istotny wpływ na rozwiązanie rozważanego zagadnienia, znajduje się w [5,4].

Układ równań (12) posiada jeden lub więcej stopni swobody, zatem najdogodniejszą metodą rozwiązania jest zastosowanie macierzy pseudoodwrotnej. Oznaczmy przez A^+

macierz pseudoodwrotną do macierzy A (patrz np. [8]). Zachodzi następujące twierdzenie:

Twierdzenie 1 [8]. *Warunkiem koniecznym i dostatecznym istnienia rozwiązania układu równań: $Ax = b$ jest warunek: $AA^+b = b$. W tym przypadku rozwiązanie ogólne jest następujące: $x = A^+b + (I - A^+A)y$, gdzie y jest dowolnym wektorem.*

Rozwiązaniem o minimalnej normie nazywamy:

$$x = A^+b. \quad (13)$$

W pracach [5,4] przedyskutowano zgodność powyższych rozwiązań z rozwiązaniem przy pomocy średniej geometrycznej, stosowanym dla przypadku "pełnych" macierzy ocen., tzn. bez brakujących ocen. Wykazano, że dla przypadków najczęściej stosowanych w praktyce (rzęd macierzy A jest równy $n-1$) rozwiązanie ogólne układu (12) jest zgodne z rozwiązaniem przy pomocy średniej geometrycznej. Udowodniono również następujące twierdzenie:

Twierdzenie 2 [5]. *Jeśli A ze strukturą określoną przez (9), (10) i (11) posiada rząd $n-1$ to ogólne rozwiązanie układu równań (12) zawsze istnieje i ma postać: $x = A^+b + z$, gdzie $z_i = c \forall i = 1, 2, \dots, n$.*

Dowód twierdzenia można znaleźć w pracy [5].

Powracając do funkcji wykładniczych: $\alpha_i = e^{(x_i)}$, a następnie dokonując normalizacji zgodnie z (4)-(5) rozwiązanie ogólne sprowadza się do rozwiązania o minimalnej normie (13).

4. Metoda porównywania parami dla rozmytych wartości osądów.

Sformułowanie zagadnienia

Van Laarhoven i Pedrycz wprowadzili [7] rozmytą wersję metody porównywania parami, dokonując rozmycia układu równań normalnych (12) i otrzymując następujący układ równań z rozmytymi współczynnikami:

$$A\tilde{x} = \tilde{b}, \quad (14)$$

gdzie macierz A posiada takie same własności jak dla przypadku z danymi dokładnymi (12), natomiast wektor prawych stron \tilde{b} posiada elementy, które są

trójkątnymi liczbami rozmytymi: $\tilde{b}_i = \sum_{i \neq j, j=1}^n \sum_{k=1}^n \tilde{y}_{ijk}, i = 1, 2, \dots, n$, gdzie: $\tilde{y}_{ijk} = \ln(\tilde{r}_{ijk})$.

Zakłada się, że ekspert nie zawsze jest w stanie podać dokładnej liczby ze zbioru S , ale jest w stanie stwierdzić, że jego ocena zawiera się w przedziale $[l_{r_{ij}}, u_{r_{ij}}]$, a najbardziej możliwa wg. niego wartość jest równa $m_{r_{ij}}$. Na tej podstawie tworzona jest trójkątna liczba rozmyta: $\tilde{r}_{ij} = (l_{r_{ij}}, m_{r_{ij}}, u_{r_{ij}})$. Oczywiście $l_{r_{ij}}, m_{r_{ij}}, u_{r_{ij}} \in S$. Poszukuje się rozwiązania problemu (14) w postaci wektora liczb rozmytych: $\tilde{x} = (\tilde{x}_1, \tilde{x}_2, \dots, \tilde{x}_n)^T$, gdzie $\tilde{x}_i = (l_i, m_i, u_i)$, $i = 1, 2, \dots, n$, jest trójkątną liczbą rozmytą. Operacje dodawania,

odejmowania, mnożenia trójkątnych liczb rozmytych oraz definicje funkcji logarytmicznej i wykładniczej można znaleźć w pracach [6,7].

Niejawne rozwiązanie problemu

Stosując arytmetykę liczb rozmytych układ równań (14) można zapisać [7]:

$$m_i \sum_{i \neq j, j=1}^n d_{ij} - \sum_{i \neq j, j=1}^n d_{ij} m_j = \sum_{i \neq j, j=1}^n \sum_{k=1}^n m_{ijk}, i = 1, 2, \dots, n, \quad (15)$$

$$l_i \sum_{i \neq j, j=1}^n d_{ij} - \sum_{i \neq j, j=1}^n d_{ij} u_j = \sum_{i \neq j, j=1}^n \sum_{k=1}^n l_{ijk}, i = 1, 2, \dots, n, \quad (16)$$

$$u_i \sum_{i \neq j, j=1}^n d_{ij} - \sum_{i \neq j, j=1}^n d_{ij} l_j = \sum_{i \neq j, j=1}^n \sum_{k=1}^n u_{ijk}, i = 1, 2, \dots, n, \quad (17)$$

gdzie: $\tilde{x}_i = (l_i, m_i, u_i)$, $\tilde{y}_{ijk} = (l_{ijk}, m_{ijk}, u_{ijk}) = \ln(\tilde{r}_{ijk})$, $i, j = 1, 2, \dots, n$.

Okazuje się [6,7], że powyższy układ równań nie zawsze daje prawidłowe rozwiązanie, tzn. nie zawsze zachowany jest porządek :

$$l_i \leq m_i \leq u_i \quad (18)$$

Jawne rozwiązanie zagadnienia

Ze względu na brak prawidłowego rozwiązania w sensie (18) dla metody niejawnej w przypadku ogólnym, proponuje się zastosować jawne rozwiązanie układu (14). Podejście takie zasygnalizowano w pracy [7]. Rozwiązanie jawne układu dla minimalnej normy wektora wartości modalnych rozwiązania wyrazi się zależnością:

$$\tilde{x} = A^+ \tilde{b} \quad (19)$$

W tym przypadku rozwiązanie będzie zawsze prawidłowe w sensie (18).

Załóżmy, że wektor rozmyty \tilde{x} jest rozwiązaniem układu (19). Wówczas nie-normalizowany rozmyty ranking obiektów możemy wyznaczyć następująco:

$\tilde{s} = (\tilde{\alpha}_1, \tilde{\alpha}_2, \dots, \tilde{\alpha}_n)^T$, gdzie $\tilde{\alpha}_i = e^{(\tilde{x}_i)}$, $i = 1, 2, \dots, n$. Dokonując normalizacji wektora

\tilde{s} [7] otrzymujemy wektor: $\tilde{s}^* = (\tilde{\alpha}_1^*, \tilde{\alpha}_2^*, \dots, \tilde{\alpha}_n^*)^T$, gdzie: $\tilde{\alpha}_i^* = \tilde{\alpha}_i / \sum_{i=1}^n \tilde{\alpha}_i$,

$i = 1, 2, \dots, n$. W przypadku rankingu globalnego równanie (6) zostaje zastąpione jego

wersją rozmytą: $\tilde{\alpha}_i = \sum_{j=1}^m \tilde{w}_j \tilde{\alpha}_{ij}$, $i = 1, 2, \dots, n$.

5. Przykład obliczeniowy

Przykład obliczeniowy dotyczący rankingu globalnego wersji rozmytej rozważanego zagadnienia znajduje się w pracy [7]. Autorzy zastosowali rozwiązanie niejawne układu (10). Obliczmy rozwiązanie jawne dotyczące rankingu obiektów, pod względem jednego kryterium, przedstawionego w tablicy 1 (tablica 3a w [7]). Stosując metodę jawną opisaną w rozdziale 4, znormalizowany ranking obiektów F_1 , F_2 , F_3 można przedstawić za pomocą wektora liczb rozmytych :

$$\tilde{s} = \begin{pmatrix} (0.162, 0.289, 0.524) \\ (0.141, 0.265, 0.512) \\ (0.233, 0.446, 0.838) \end{pmatrix}$$

Warto zauważyć, że zastosowano nieco inną skalę niż oryginalna, zaproponowana przez Saaty'ego. Dyskusja przyjęcia takiej skali przeprowadzona jest w [7].

obiekt	F_1	F_2	F_3
F_1	(1,1,1)	(2/3,1,3/2) (2/3,1,3/2)	(2/3,1,3/2) (2/5,1/2,2/5)
F_2	(2/3,1,3/2) (2/3,1,3/2)	(1,1,1)	(2/5,1/2,2/3)
F_3	(2/3,1,3/2) (3/2,2,5/2)	(3/2,2,5/2)	(1,1,1)

Tablica 1. Macierz ocen dla obiektów F_1, F_2, F_3 .

Table 1. Judgement matrix for the objects F_1, F_2, F_3 .

Obliczenia wektora \tilde{x} wykonano korzystając z zależności (19). Następnie, zgodnie z metodyką opisaną w rozdziale 4 powrócono do funkcji wykładniczych oraz dokonano normalizacji wektora \tilde{s} , zgodnie z arytmetyką liczb rozmytych.

6. Uwagi końcowe

W pracy przedstawiono metodykę wielokryterialnego, grupowego podejmowania decyzji dla ocen dokładnych i rozmytych. Przedstawiona metodyka może być stosowana dla ogólnego przypadku, a mianowicie wtedy, gdy mamy do czynienia z zagadnieniem brakujących ocen, gdy oryginalna metoda wprowadzona przez Saaty'ego nie może być stosowana. Warto podkreślić, że liczba rzeczywista może być stosowana jako szczególny przypadek liczby rozmytej. Zatem tablice ocen mogą zawierać zarówno liczby rzeczywiste, jak i liczby rozmyte. Zastosowanie metody jawnej rozwiązania układu równań normalnych pozwala na uzyskiwanie prawidłowych rozwiązań, co stanowiło problem w przypadku metody niejawnej [7,1]. Dyskusję ogólnego rozwiązania przy zastosowaniu metody niejawnej oraz przyjętych metod normalizacyjnych można znaleźć w [2,6]. Interesującym, ale ciągle nie rozwiązany zagadnieniem jest określenie warunków, dla których rozwiązanie niejawne jest prawidłowe.

Przedstawioną metodykę można rozszerzyć na zastosowanie trapezoidalnych liczb rozmytych. Wówczas analiza może być rozszerzona na przedziały liczbowe, ponieważ przedział liczbowy jest szczególnym przypadkiem trapezoidalnej liczby rozmytej. Przy takim podejściu można operować liczbami rzeczywistymi, przedziałami liczbowymi oraz trójkątnymi i trapezoidalnymi liczbami rozmytymi.

W kontekście zastosowanego aparatu matematycznego rozwiązanie problemu nie ulegnie zmianie, jeśli w miejsce brakujących ocen wstawimy ilorazy ocen otrzymanych z rozwiązania, innymi słowy oceny zgodne z rozwiązaniem [5].

Literatura

1. Boender C.G.E., de Graan J.G. and Lootsma F.A.: Pairwise-comparison methods using fuzzy numbers on ratio scales with geometric progression. Reports of the Department of Mathematics and Informatics 85-24, Delft University of Technology, 1985.
2. Boender C.G.E., de Graan J.G. and Lootsma F.A.: Multi-criteria decision analysis with fuzzy pairwise comparisons. *Fuzzy Sets and Systems*, 29:133-143, 1989.
3. Grawford G. and Williams C.: A note on the analysis of subjective judgement matrices. *Journal of Mathematical Psychology*, 29:387-405, 1985.
4. Kwiesielewicz M.: The logarithmic least squares and generalised pseudoinverse in estimating ratios, 1993 (submitted to European J. Operational research).
5. Kwiesielewicz M.: Pairwise comparison using the logarithmic least squares and the generalised pseudoinverse. Reports of the Department of mathematics and Informatics 93-84, Delft University of Technology, 1993.
6. Kwiesielewicz M.: A note on the fuzzy extension of Saaty's priority theory. 1994 (Submitted to Fuzzy Sets and Systems).
7. Laarhoven van P.J.M. and Pedrycz W.: A fuzzy extension of Saaty priority theory. *Fuzzy Sets and Systems*, 11:229-241, 1983.
8. Pyle L.D.: The generalised inverse in linear programming. Basic structure. *SIAM J. Appl. Math.*, 22:335-355, 1972.
9. Saaty T.L. The Analytic Hierarchy Process. Mc-Graw Hill New York 1980.
10. Saaty T.L and Vargas L.G.: Comparison of eigenvalue, logarithmic least squares and least squares method in estimating ratios. *Mathematical Modelling*, 5:309-324, 1984

Streszczenie

W pracy przedstawia się zagadnienie wielokryterialnego grupowego podejmowania decyzji, stosując metodę hierarchicznego procesu decyzyjnego dla danych rzeczywistych i rozmytych. Użycie metody logarytmicznych najmniejszych kwadratów sprowadza zagadnienie do rozwiązania układu równań normalnych, który z kolei proponuje się rozwiązać przy pomocy ogólnej pseudoodwrotności. Dla przypadku rozmytego poszukuje się rozwiązań jawnych.

MULTICRITERIA GROUP DECISION-MAKING USING PAIRWISE COMPARISON METHOD

Summary

Multicriteria group decision-making using the Analytic Hierarchy Process for deterministic and fuzzy data is considered in this paper. So-called normal equations, obtained as a result of application of logarithmic least squares method, are solved using a pseudoinverse approach. For the fuzzy case an explicit solution of the problem is calculated.

DIGITAL SIMULATION OF RANDOM FIELDS FOR AERONAUTICAL PURPOSES

1. Introduction

Paper offers an introduction to the results derived within the scope of the KBN grant No 7 7003 92 03 entitled as above. The major outcome of the mentioned grant was the software package filling eight discs of 1.44Mb which later on allowed to derive numerical results related to stochastic dynamics of six flying objects of the Polish origin: three gliders and three airplanes. This is the market package available from the Author. The scope of the paper being under the severe space restrictions can only sketch the entire field of the mentioned project that besides was placed into the body of the Report handed to the KBN and then also served to prepare a monograph [5]. Below we are briefing the subject of these two sources - complementing the oral presentation done during the Conference.

2. Transfer Functions of a Rigid Airplane

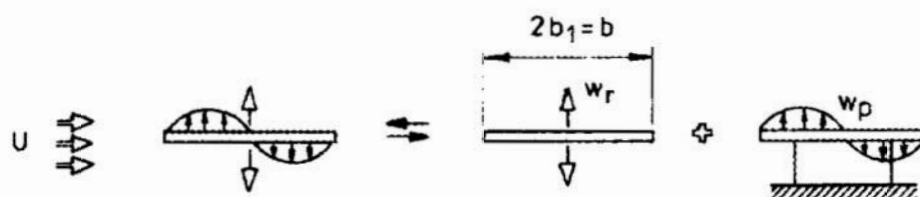


Fig.1. Aerodynamic Superposition on a Flat Plate

We begin with the equation of motion which in a Newtonian fashion combines motion and forces. The point is that to derive the governing equation of motion we follow the concepts of harmonic analysis. Due to the basic properties of the linear systems the sampling function becomes here a sinusoidal (vertical-gust-like) excitation. Therefore the response will constitute so called transfer function either admittance (we prefer the first name). Below are shown successive steps and results got in this way, together with some details concerning the main notations. In the first step we write in the dimensional form the Newtonian equation of vertical motion of a flat plate shown in Fig.1 which represents an airplane wing which is hanging like a parachute over the material point representing an airplane. Regarding the rule of the aerodynamic superposition explained in Fig.1 the forces arising due to sinusoidal gust (like on a fixed wing) and arising due to the vertical motion alone - indicated below by two different unsteady aerodynamics - Theodorsen function $C(d)$ and Sears functions $K(d)$ - were separated. Notations and symbols rather conventional follow [4] and besides the symbols explained in Fig.1 indicate: m - an airplane mass, S - its wing area, U - horizontal flight velocity, ρ - air density, ω - dimensional circular frequency, t - dimensional time.

The governing equation of motion takes the shape:

$$m\dot{w}_r = \frac{1}{2}\rho S b_1 \pi (2C(d) \frac{U}{b_1} w_r + \dot{w}_r) + \rho S U \pi K(d) w_p \quad (1)$$

The form of the equation is dimensional (except non-dimensional frequency d that is defined below). In the next step we transform it into non-dimensional form. The intermediate term may look like this:

$$\frac{m}{\pi \rho S U} \frac{1}{dt} \frac{dw_r}{U} + \frac{2\pi \rho S b_1}{2\pi \rho S b_1} C(d) \frac{w_r}{U} \frac{U}{U} + \frac{\pi \rho S}{\pi \rho S} \frac{m}{\pi \rho S U} \frac{b_1}{b_1} \frac{1}{dt} \frac{dw_r}{U} = \quad (2)$$

$$= \frac{\pi \rho S U}{\pi \rho S U} K(d) w_p$$

There can be also listed non-dimensional terms designed to replace their dimensional correspondents which the most frequently occur in this consideration:

$$\tau = t \frac{U}{b_1}; \quad d = \omega \frac{b_1}{U}; \quad w_r = \frac{w_r}{U}; \quad w_p = \frac{w_p}{U}; \quad \hat{t}_1 = \frac{m}{\pi \rho S U}; \quad t = \frac{t}{\hat{t}_1}$$

Note that some of this formulae have special names as given for instance by [1], [3] or [4] (3)

$$\frac{dw_r}{dt} \frac{\hat{t}_1}{U} = \dot{w}_r; \quad \mu_1 = \frac{2m}{\pi \rho S b_1}$$

With this notation the non dimensional form of (1) becomes now strikingly simple:

$$\dot{w}_r + (C(d)w_r + \frac{1}{\mu_1} \dot{w}_r) = K(d)w_p \quad (4)$$

This simplicity is some-what costly and some-what apparent because the Theodersen function $C(d)$ and Sears function $K(d)$ are in reality expressed by the special functions - what we provide explicitly below by using the appropriate Bessel function J and Y - called conveniently the first and the second *kind* of the zero and the first *order* (this the last is indicated by the appropriate subscript). This laborious system of notations and symbols covers:

$$C(d) = F(d) + iG(d) \quad \text{and} \quad K(d) = KR(d) + iKI(d) \quad (5)$$

$$F = \frac{1}{\circ\mathcal{H}}(J_1(J_1 + Y_0) + Y_1(Y_0 - J_0)) \quad \text{and} \quad G = -\frac{1}{\circ\mathcal{H}}(Y_1Y_0 + J_1J_0) \quad (6)$$

$$\circ\mathcal{H} = (J_1 + Y_0)^2 + (Y_1 - J_0)^2 \quad (7)$$

$$KR = FJ_0 + GJ_0 \quad \text{and} \quad KI = GJ_0 - FJ_1 + J_1 \quad (8)$$

Once we turn back to (1) we may show that the force arising due to the unsteady motion can be expressed alternatively due to the complex harmonic motion in a one of the either form:

$$e^{i\tau d} = e^{i\omega t} \quad \text{therefore with } w_r = z \quad \text{we shall get}$$

$$\frac{1}{2}\rho S b_1 \pi (2C \frac{U}{b_1} w_r + \dot{w}_r) = \pi \rho d^2 U^2 z (1 + \frac{2G}{d} + i(-\frac{2F}{d})) \frac{S}{b} \quad (9)$$

Whereas non-dimensional acceleration \dot{w}_r can be expressed by non dimensional speed w_r :

$$\dot{w}_r = i \frac{\mu_1}{2} w_r d \quad (10)$$

Therefore the centre of gravity acceleration \ddot{z}_{cg} can be expressed in the form:

$$\ddot{z}_{cg} = \dot{w}_r = K(d) \frac{\mu_1}{2} \frac{id}{C(d) + i(\frac{1}{2} + \frac{\mu_1}{2})d} \quad (11)$$

The above expression (11) gives us the desired transfer function - that is *input-output* relation with the complex harmonic vertical gust input and vertical cg acceleration as the output. Re-arranging some-what the above expression we shall get its final dimensional form as:

$$H_{\ddot{z}}^w(id) = \frac{d}{d(1 + \mu_1) + 2G - i2F} \frac{2U}{b_1} K(d) \quad (12)$$

For the purposes of analysing linear systems the above complex functions (12) should be converted into appropriate values of the square of moduli. Therefore the finally useful expression will be rather as given below by (13).

The following formula closes this introductory considerations.

$$\left| H_{\ddot{z}}^w(id) \right|^2 = \frac{d^2}{4(F^2 + G^2) + 4Gd(1 + \mu_1) + (1 + \mu_1)^2 d^2} \frac{4U^2}{b_1^2} |K(id)|^2 \quad (13)$$

It would be interesting to point out at this stage the essential differences in the solutions related to the quasi stationary and non stationary aerodynamics possible to derive explicitly from the formula (13). Quasi steady solution requires to assume $F=1$, $G=0$ and moreover the modulus square of the Sears function $K(id)$ replace by the unity (see also [1]-[4]).

3. Spectral Response of a Rigid Airframe

So far we discussed the problem of the gust response up to the level of transfer functions. Now we are going to discuss the second step in this gust response study - which does concern the spectral characteristics. To do that we enclose the Karman spectrum of the vertical gusts in the form that involves explicitly the speed of flight U .

$$\Phi_w(\omega, L) = \frac{L}{\pi U} \frac{1 + \frac{8}{3} \left(1.339 \frac{\omega L}{U}\right)^2}{\left(1 + \left(1.339 \frac{\omega L}{U}\right)^2\right)^{11/6}} \quad \text{and} \quad \sigma_w^2 = 1 \quad (14)$$

For the case under consideration - i.e. the one-dimensional gust field - we apply one of the basic theorems of the linear filtration that we write here in the form demanding the transfer function that describes the vertical acceleration at the airplane centre of gravity:

$${}^{zsc} S_w^r(\omega, L) = \left| {}^{zsc} H_{\ddot{z}}^w(i\omega) \right|^2 \Phi_w(\omega, L) \quad (15)$$

In this way we get finally the desired solution to the problem in the form of the appropriate spectral density functions where appear the values of the scale of turbulence L , and speed of flight U ; its altitude represented by the air density ρ and other parameters occur in an implicit form while evaluating the appropriate transfer functions. Such solutions describe entirely an airplane centre of gravity. To arrive at the other points on the airframe with non trivial results we have to consider the model of an elastic airplane and this is done below.

4. Spectral Response of an Elastic Airplane

The above presented consideration was generalised to the case with $\nu = 2 + s$ degrees of freedom - where symbol s denotes the number of symmetrical vibration modes of the airframe under consideration. This generalisation is not a trivial one and requires to develop appropriate numerical tools to describe first of all the income involved by the incorporated into analysis vibration modes of the airplane. We are not able to present such details in this paper so we can only sketch them and show the final outcome of the complex numerical procedure. We

$$L_z^M = \pi \rho c^3 \omega^2 (a_{11} \frac{h_1}{c} + a_{12} \alpha_1) \quad M_y^M = \pi \rho c^4 \omega^2 (a_{21} \frac{h_1}{c} + a_{22} \alpha_1) \quad (17)$$

Similar procedure as it was for the simple model of Y.C.Fung - provides coefficients a_{ij} being expressed by the Theodorsen function $C(d)$ and we show them in an explicit form:

$$\begin{aligned} a_{11} &= 1 + \frac{2G}{d} + i(-\frac{2F}{d}) \\ a_{12} &= \frac{1}{2} + \frac{2G}{d} - \frac{2F}{d^2} - (\frac{1}{2} + a)(1 + \frac{2G}{d}) + i[(\frac{1}{2} + a)\frac{2F}{d} - \frac{1+2F}{d} - \frac{2G}{d^2}] \\ a_{21} &= \frac{1}{2} - (\frac{1}{2} + a)(1 + \frac{2G}{d}) + i(\frac{1}{2} + a)\frac{2F}{d} \\ a_{22} &= \frac{3}{8} - (\frac{1}{2} + a)(1 + \frac{2G}{d} - \frac{2F}{d^2}) + (\frac{1}{2} + a)^2(1 + \frac{2G}{d}) + i[-\frac{1}{d^2} + (\frac{1}{2} + a) \\ &\quad (1 + \frac{2F}{d} + \frac{2G}{d} - (\frac{1}{2} + a)^2 \frac{2F}{d})] \end{aligned} \quad (18)$$

Besides of (17) which explains the details of the forces originated in motion there are terms which explain the details of the forces caused by the gust. They are divided into two groups - separately for the wing and separately for the tail. Forces caused by the complex harmonic gust upon the wing and the tail elements are of the form:

$$L_z^G = 2\pi\rho U c K(d) w_p \exp(i\omega t) \quad M_y^G = (\frac{1}{2} + a) c L_z^G \quad (19)$$

$$L_z^{GH} = 2\pi\rho U c K(d) w_p \exp(i\omega(t - \frac{l_k}{U})) \quad M_y^{GH} = (\frac{1}{2} + a) c L_z^{GH} \quad (20)$$

The other details referring expressions of Theodorsen and Sears are the same as given by (5) - (8). Nevertheless there is a question as how to determine generalised forces shown symbolically in (16) and related to the displacements concerned with the elastic properties of the airplane. Here we have the commencing details. Let the subscript j will be reserved to indicate the appropriate vibration mode and the subscript s the appropriate point chosen upon the airframe. Therefore the suitable deflection in vertical direction can be expressed by:

$$z_j(x_s, y_s) = \phi h_j(y_s) + x_s \phi s_j(y_s) \quad (21)$$

$\phi h_j(y)$ - indicates the amplitude of bending j - mode

$\phi s_j(y)$ - indicates the twisting angle of j - mode

The total displacement at a cross section like the one shown in Fig.3 due to the elastic properties will be expressed by:

$$h(y) = \sum_r \xi_r \phi_{h_r}(y) \quad \alpha(y) = \sum_r \xi_r \phi_{s_r}(y) \quad (22)$$

Finally, due to the sinusoidal excitation - the response functions will be harmonic function:

$$\xi_j = \xi_j^0(i\omega) \exp(i\omega t) \quad (23)$$

Therefore eventually the system of equations to be solved becomes a linear algebraic system with complex coefficients describing unknown transfer functions (23). This system has to be solved every time for the one particular value of frequency ω and for a particular point chosen upon an airframe. To build the appropriate vertical displacements at each point the solutions should be substituted into the expression leading to the desired transfer functions:

$$\left| {}_{\omega \mathcal{N}}^s H_z^w(i\omega) \right|^2 = \left[\sum_{j=1}^{\mathcal{N}} {}^s z_j(x_s, y_s) \operatorname{Re} \xi_j^0(i\omega) \right]^2 + \left[\sum_{j=1}^{\mathcal{N}} {}^s z_j(x_s, y_s) \operatorname{Im} \xi_j^0(i\omega) \right]^2 \quad (24)$$

In expression (24) we use besides of the other symbols:

- s - to denote the chosen point upon the airframe due to Fig.3,
- $\omega \mathcal{N}$ - to denote the number of degrees of freedom.

5. Example, Conclusions

Reminding now the Y.C.Fung case [2] with $\omega \mathcal{N} = 1$ we seek for the transfer functions of the accelerations and then the appropriate gust response power spectral densities. Therefore we use first formula (25) and then combine it with the gust spectrum (14) similarly like it was done for the rigid case by using (15). It is illustrated by the particular numerical result shown in Fig.4.

$$\left| {}_{\omega \mathcal{N}}^{zsc} H_z^w \right|^2 = \omega^4 \left| {}_{\omega \mathcal{N}}^{zsc} H_z^w \right|^2 \quad (25)$$

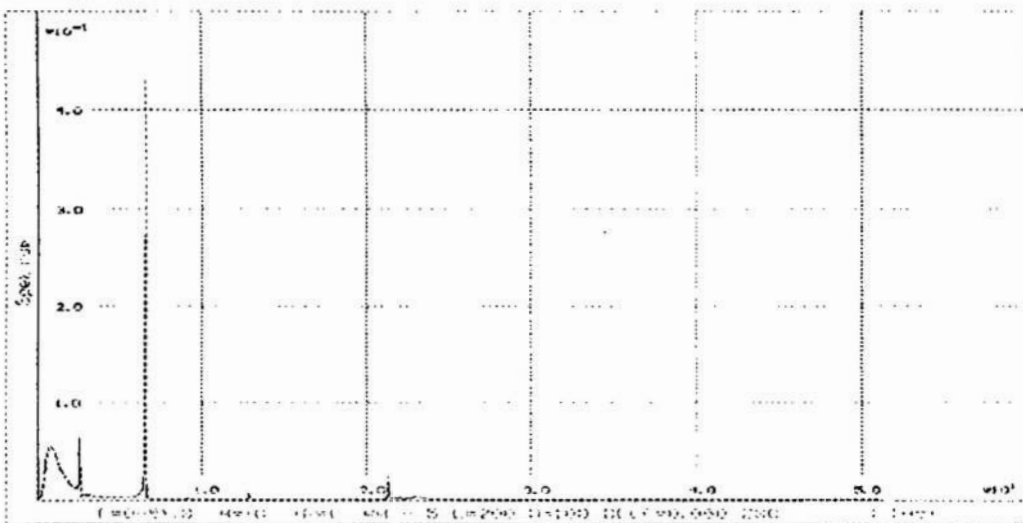


Fig.4. Spectral Density at CG of the PW-5 Glider

In this numerical example (taken from [5]) the calculations concern a one of the newest Polish gliders - the PW-5 "Smyk" taking into account 2 rigid and 8 elastic degrees of freedom and cover the frequency range up to about 50Hz. The case under attention does concern a horizontal sea level flight with velocity $U=100\text{km/h}$ and the integral scale of turbulence $L=200\text{m}$. They describe a moderate gusty conditions. The corresponding gust alleviation factor becomes equal to 0.211 - what is quite near to the appropriate value of the rigid model which becomes 0.240 by assuming a single degree of freedom. The variance of the response was for this case equal to 0.845 m/s/s (with the gust variance equal to unity). There is a single peak corresponding to the frequency 6.6 Hz what refers to the one of the natural modes of the glider and becomes quite near to the sensitive range of frequencies annoying for the human being. Appearing in this place the problem of the passenger/pilot comfort due to the lack of the space can not be discussed in this paper (for such details is also recommended [5]). And in this place we also have to end our all considerations. So, our presentation came to the point at which the gust response problem was solved reaching a stage of the appropriate power spectral density functions PSD describing loads at chosen points of the airframe. The PSD derived in this way were applied to discuss the random vibration tests aiming at an assessment of the fatigue life of a given airframe and also to discuss some aspects of the passenger comfort ([5] and [6]).

References

1. Y.C.Fung: *An Introduction to the Theory of Aeroelasticity*. J.Wiley, New York, 1955 (also translated into Russian in 1959).
2. Y.C.Fung: Statistical Aspects of Dynamic Loads. *JAS*, 20, May 1953, pp. 317-330.
3. R.L.Bisplinghoff, H.Ashley: *Principles of Aeroelasticity*. Dover, New York 1962.
4. L.M.Laudański: *An Introduction to Stochastic Dynamics of Gliders* (in Polish). Rzeszów Technical University Press 1992 (1978).
5. L.M.Laudański: *Random Fields and Airplane Loads* (in preparation for printing).
6. A.Kucaba-Piętal, L.M.Laudański: Designing Random Vibration Test. *Proceedings of the First Symposium on Mechanical System Dynamics*. Warsaw Technical University, September 1994 (in print).

IMITACYJNE MODELE PÓL LOSOWYCH AERONAUTYKI Streszczenie

Niniejsza praca ukazuje zasadniczy aspekt opracowania [5] przygotowywanego do druku przez autora i dotyczącego stochastycznej dynamiki samolotu. Oparte jest ono o KBN-owski grant No 7 7003 92 03 o tytule jak wyżej. Podstawowy problem dotyczy spektralnego modelu odpowiedzi odkształcalnego samolotu lecącego w burzliwej atmosferze. Praca [5] zawiera również szczegółowo opracowany aspekt generowania realizacji procesów stochastycznych posiadających wielomodalne spektra typowe dla tych problemów dynamiki samolotu oparty na oryginalnej metodzie filtracji. Zagadnienia te z braku miejsca zostały w niniejszej publikacji całkowicie pominięte. Można je będzie znaleźć w pracy [6].

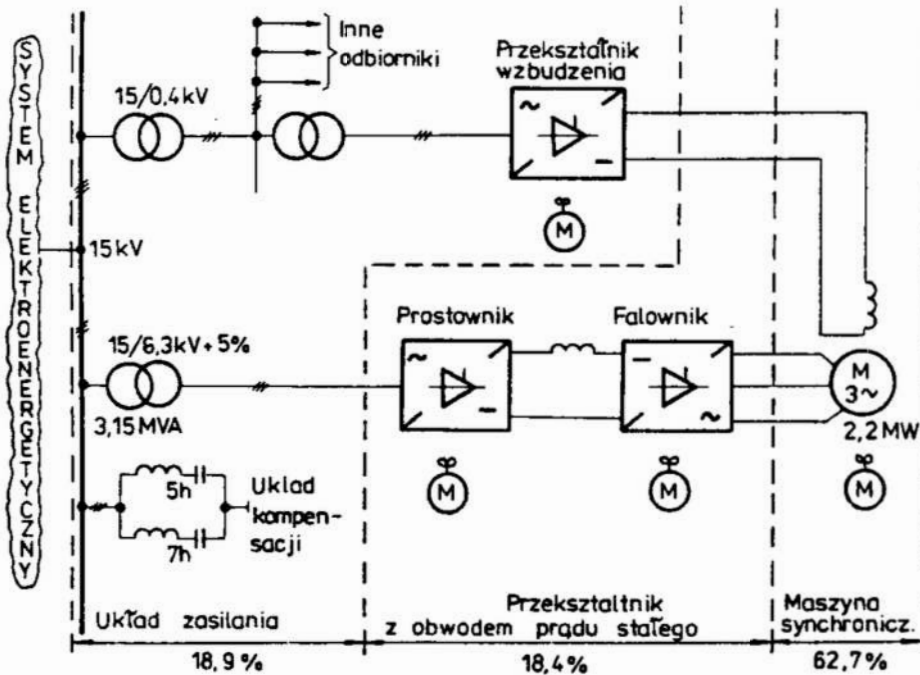
KRYTERIUM UOGÓLNIONE OPTIMALIZACJI STATYCZNEJ SILNIKA PRZEKSZTAŁNIKOWEGO

1. Wprowadzenie

Pod pojęciem silnika przekształtnikowego (SP) rozumie się maszynę synchroniczną z falownikiem prądu, zasilającą maszynę. Przy tym istotne jest, że zawory falownika załączane są w zależności od położenia wirnika, tak że nie następuje utrata synchronizmu [2].

Typowym zastosowaniem silnika przekształtnikowego są napędy dużej mocy (1-30) MW [5], pracujące często przez długie okresy czasu z ustalonym obciążeniem, np. wentylatory, sprężarki, pompy, śruby okrętowe, itp. Celowa jest więc optymalizacja statyczna napędu, polegająca na wyborze takiego sterowania, które zapewni minimalne straty mocy układu w całym zakresie wymaganych charakterystyk momentu i prędkości, przy zadanych ograniczeniach. Algorytm sterowania z nieskończonej liczby możliwych zestawów wartości parametrów sterowania (każdy zestaw zapewnia wymagany moment i prędkość, respektując ograniczenia) powinien wybrać zestaw najlepszy w sensie minimalnych strat i odpowiedniego ich rozłożenia na poszczególne części układu, gwarantujący maksymalne wykorzystanie całego napędu.

W miarę jak wzrastają koszty energii obserwuje się tendencję do coraz bardziej całościowego traktowania układu napędowego, które polega na włączaniu do rozważań urządzeń znajdujących się w torze doprowadzania energii do silnika. Zwłaszcza w przypadku napędów tyrystorowych, z powodu mocy biernej i odkształcenia prądu, znacząca część strat powstaje w urządzeniach znajdujących się poza układem napędowym w znaczeniu ścisłym. Mając to na uwadze, przyjęto do rozważań, dla układu dużej mocy, schemat ideowy silnika przekształtnikowego z układem zasilania jak na rys.1. W torze głównym, oprócz przekształtników maszynowego i sieciowego z obwodem pośredniczącym, występuje po stronie zasilania transformator, a także równoległy układ kompensacji oddziaływania na sieć, np. w postaci filtrów harmonicznych i równoległej baterii



Rys.1. Układ SP z uwzględnieniem zasilania i z rozkładem strat dla znamionowego punktu pracy silnika 2,2 MW

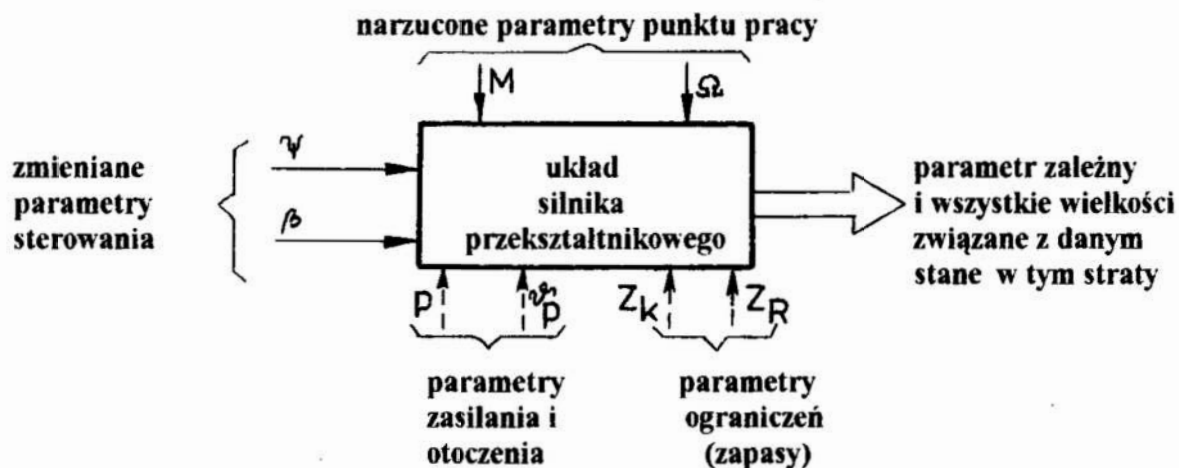
kondensatorów. W związku z dużymi zmianami prądu wzbudzenia przy zmianach prędkości i obciążenia, w optymalizacji statycznej należy uwzględnić także straty w pokazanych na rys.1 elementach zasilania wzbudzenia.

O potrzebie uwzględniania w procedurze optymalizacji także układu zasilania świadczą udziały procentowe poszczególnych części układu w stratach całkowitych, podane na rys.1. Wartości te obliczono dla układu z silnikiem o mocy 2,2 MW przy pracy znamionowej [3].

2. Potrzeba uwzględnienia wielu wymagań przy optymalizacji

Silnik przekształtnikowy z punktu widzenia stanu ustalonego (dokładnie pseudoustalonego) posiada 5 wejść - wyjść (rys.2), z którymi związane jest pięć wielkości - współrzędnych stanu. Cztery z nich mogą być narzucone lub wybierane z zewnątrz, wartość piątej wynika wg równań z tych czterech.

Przyjęto, że narzucone są współrzędne mechaniczne: moment M i prędkość obrotowa Ω , które nazywa się dalej parametrami punktu pracy. Wybierane będą wartości strumienia szczelinowego ψ silnika i kąta β wyprzedzenia falownika [4]. Poprzez wielkości ψ i β nadrzędny sterownik, optymalizujący stan ustalony, oddziałuje na układ. Wielkości te będą nazywane parametrami sterowania. Nie ma potrzeby nazywania piątej współrzędnej. Mogą nią być wielkości związane z obwodem pośredniczącym lub z przekształtnikiem sieciowym (np. kąt α opóźnienia załączania). Razem z wartością piątej współrzędnej wynikają (z parametrów punktu pracy i z parametrów sterowania) wszystkie wielkości związane z danym stanem pracy, w tym także straty.



Rys.2. Schemat układu SP jako obiektu sterowania w stanie ustalonym

Na rys.2 zaznaczono liniami przerywanymi wejścia dla czterech dalszych wielkości, które nie są współrzędnymi stanu, a wyznaczają obszar dopuszczalnych obciążeń i sterowań, a przez to wpływają na poziom strat. Są to: poziom napięcia P w sieci zasilającej, temperatura otoczenia \mathcal{G}_p , minimalny zapas komutacyjny Z_k i minimalny zapas regulacyjny Z_R .

Poziom napięcia w sieci zasilającej ma wpływ głównie na punkt pracy przekształtnika sieciowego, a przez to na moc bierną i straty mocy w układzie zasilania. Parametr P oznacza stosunek wartości napięcia sieci do jego wartości znamionowej.

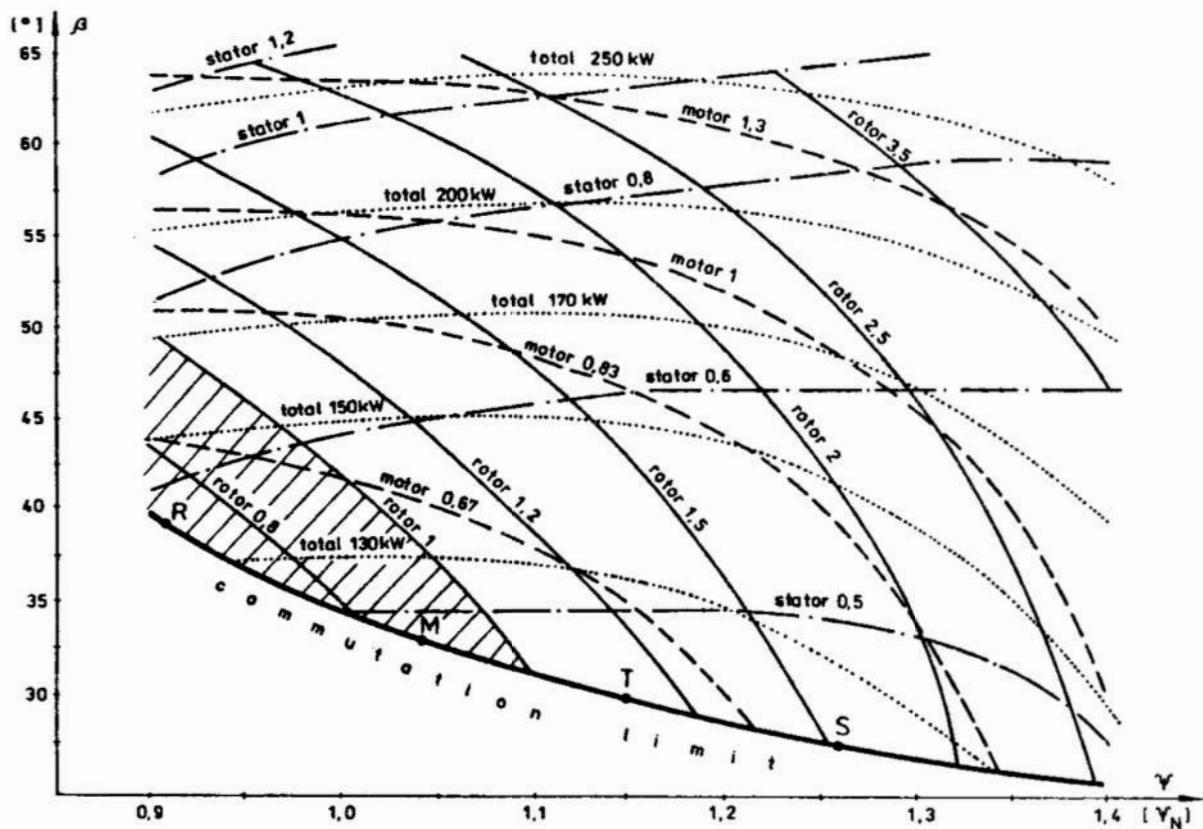
Temperatura otoczenia \mathcal{G}_p z jednej strony wpływa na rezystancje a tym samym straty w uzwojeniach, z drugiej zaś na dopuszczalny przyrost temperatury, wartość dopuszczalnego obciążenia i obszar dopuszczalnych sterowań, przy których nie następuje przegrzanie maszyny.

Obszar dopuszczalnych sterowań na płaszczyźnie ψ , β jest ograniczony. Wymaganiem nadrzędnymi są niezawodność pracy falownikowej przekształtnika maszynowego i odpowiednia dynamika przejścia do nowego punktu pracy. Wymaganiom tym odpowiadają minimalne zapasy komutacyjny Z_k i regulacyjny Z_R wprowadzone do układu [4].

Wpływ parametrów sterowania na straty dla wybranego punktu pracy, pokazany jest na rys.3. Wyniki przedstawione są w postaci "mapy", na której "poziomicami" są linie jednakowych strat, a "współrzędnymi" parametry sterowania: kąt β wyprzedzenia falownika i strumień szczelinowy silnika ψ odniesiony do wartości znamionowej ψ_N . Na rysunku pokazany jest przebieg "poziomic" dla:

- strat ciepłych w wirniku (linie ciągłe z opisem "rotor"),
- strat ciepłych w stojanie (linie osiowe z opisem "stator"),
- strat ciepłych w silniku (linie kreskowe z opisem "motor"),
- strat łącznych w całym układzie (linie kropkowe z opisem "total").

Przy poszczególnych liniach, podane są wartości strat. Dla silnika i jego części są to wartości względne (odniesieniem są znamionowe straty ciepłe silnika lub jego części). Straty w całym układzie podawane są w wartościach bezwzględnych - w kW. Obszar dopuszczalnych sterowań jest w przypadku z rys.3 ograniczony od dołu minimalnym zapasem komutacyjnym Z_k .



Rys.3. Krzywe tych samych strat na płaszczyźnie parametrów sterowania ψ, β dla pozostałych parametrów o wartościach: $\Omega = 0,5\Omega_N, M=M_N, \theta_p = 40^\circ C, Z_k=0,406 Vs, Z_R=10\%$

Optymalizacja statyczna mogłaby polegać na minimalizacji strat łącznych w układzie, włączając straty w układzie zasilania. Jednak mogą mieć miejsce przypadki, jak pokazano na rys.3, gdzie minimum strat łącznych (punkt T), a także minimum strat w stojanie (punkt S) leżą poza obszarem dopuszczalnym ze względu na nagrzewanie wirnika, tj. poza zakreskowanym obszarem pod linią "rotor 1". W innym przypadku minimum strat łącznych mogłoby znaleźć się poza obszarem dopuszczalnym ze względu na straty w stojanie. Kryterium nie może być także sumą strat w silniku, gdyż jak stwierdzono w [4], straty te nie są miarą zagrożenia cieplnego silnika. Tak więc żadne ze strat: ani łączne w układzie ani łączne w silniku, ani poszczególnych jego części oddzielnie nie powinny być kryterium optymalizacji. Kryterium to powinno być bardziej złożone.

3. Kryterium uogólnione optymalizacji

Zagadnienie sformułowania kryterium uogólnionego, uwzględniającego wiele wymagań składowych rozwiązano w oparciu o funkcję wymagań* [1]. U podstaw budowy takiego kryterium jest idea przekształcenia naturalnych wartości częściowych wskaźników w bezwymiarową skalę wg funkcji wymagań

$$H_i = \exp[-\exp(-h_i)] \quad (1)$$

gdzie: i - jest indeksem oznaczającym numer wymagania częściowego,

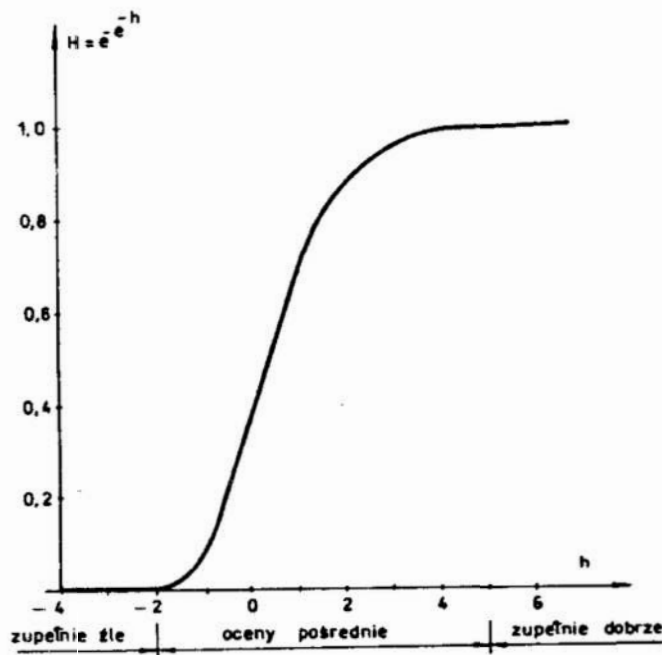
h_i - jest bezwymiarową zmienną związaną z kryterium częściowym i .

Funkcja ta jest pewną postacią krzywej wzrostu Gompertza i jej przebieg pokazany jest na rys.4, na którym podano słownie sformułowany stopień spełnienia wymagania w poszczególnych zakresach. Funkcja ma stałą wartość, wynoszącą praktycznie zero dla $h_i < -2$ i stałą wartość wynoszącą praktycznie 1 dla $h_i > 5$. Zmienna h_i powinna być powiązana z wartością wskaźnika stanowiącego miarę i -tego kryterium w jednostkach fizycznych w taki sposób, aby całkowicie niedopuszczalne wartości kryterium dawały wartości $h_i < -2$, a przy całkowitym spełnieniu wymagania było $h_i > 5$.

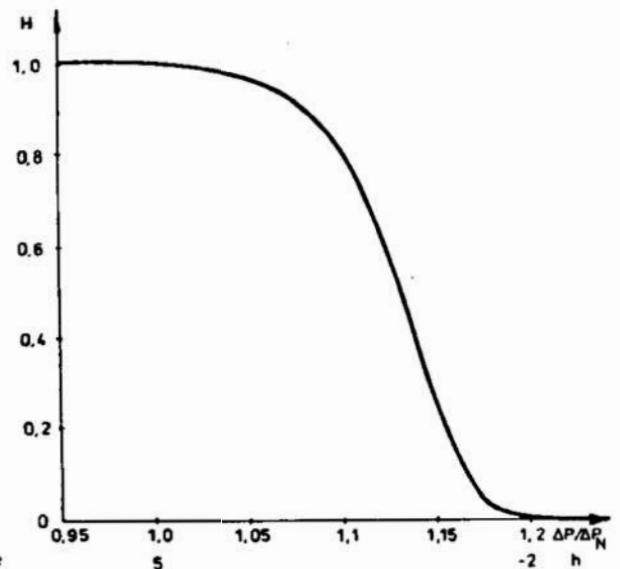
Uogólnienie funkcji wymagań polega na złożeniu kilku kryteriów w jedno wg zależności [1]:

$$H = \sqrt[n]{\prod_{i=1}^n H_i} \quad (2)$$

gdzie: n - ilość wymagań składowych.



Rys.4. Funkcja (1) z ocenami spełnienia wymagań



Rys.5. Funkcja wg (6) i (7) przy przekształceniach (8) i (9)

Można zauważyć, że postać uogólnionej funkcji wymagań (2) sprawia, że jej wartość pozostaje zerowa tak długo, jak długo choćby jedno z wymagań składowych jest całkowicie niespełnione. H przyjmuje wartość 1 dopiero wówczas, gdy wszystkie wymagania są całkowicie spełnione.

* Określenie "funkcja wymagań" jest tłumaczeniem angielskiej nazwy "desirability function"

W celu optymalizacji pracy układu silnika przekształtnikowego opracowano kryterium uogólnione w postaci

$$G = K \cdot \Delta P = \min \quad (3)$$

gdzie: K - współczynnik kary,

ΔP - łączne straty w układzie.

Współczynnik kary związany jest ze stratami w stojanie i wirniku za pomocą uogólnionej dwuargumentowanej funkcji wymagań

$$K = 2 - H \quad (4)$$

gdzie:

$$H = \sqrt{H_S H_R} \quad (5)$$

$$H_S = \exp[-\exp(-h_S)] \quad (6)$$

$$H_R = \exp[-\exp(-h_R)] \quad (7)$$

a h_S i h_R są bezwymiarowymi zmiennymi związanymi ze stratami cieplnymi odpowiednio w stojanie i w wirniku.

Zgodnie z podanym wyżej zaleceniem, dotyczącym przekształcenia liczbowej miary kryterium w wielkości h_i , przyjęto przekształcenia w postaci

$$h_S = 40 - 35 \frac{\Delta P_S}{\Delta P_{SN}} \quad (8)$$

$$h_R = 40 - 35 \frac{\Delta P_R}{\Delta P_{RN}} \quad (9)$$

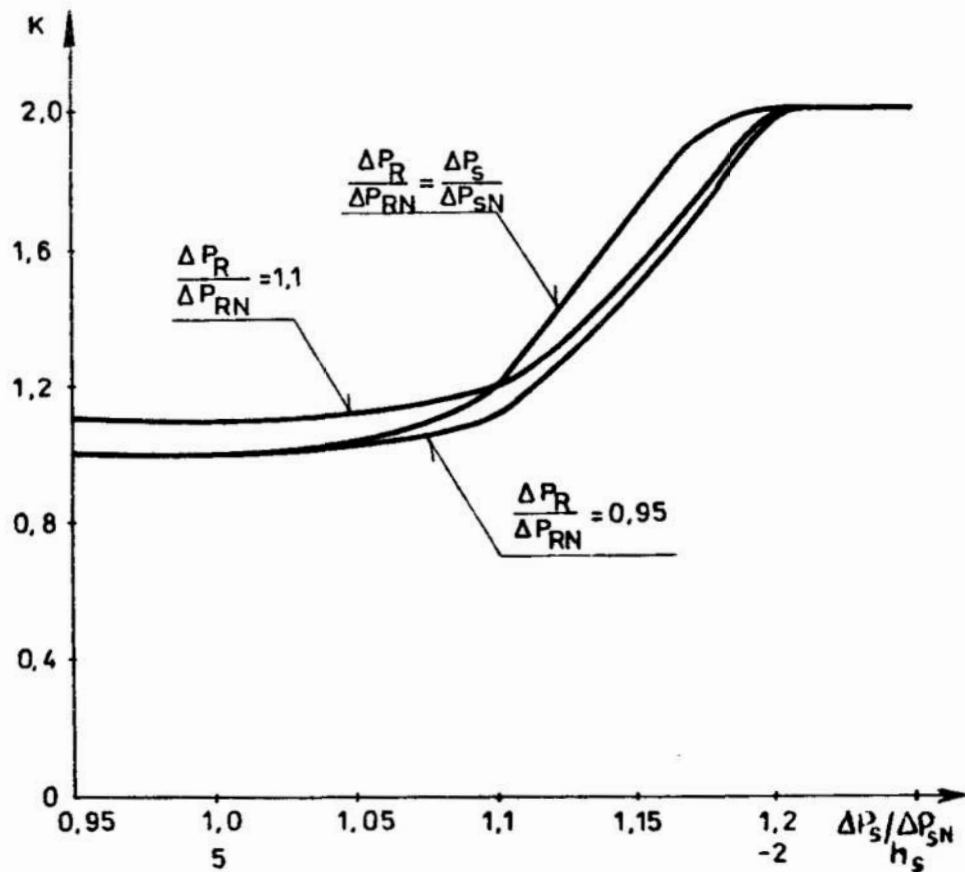
Przebieg funkcji (6) i (7) przy przekształceniach (8) i (9) pokazany jest na rys.5. Widać, że przyjęta postać przekształcenia oznacza całkowite spełnienie wymagania, gdy straty są poniżej strat znamionowych i całkowite nie spełnienie - gdy przekraczają one o 20% straty znamionowe. W przedziale strat względnych od 1,0 do 1,2 umowna ocena h zmienia płynnie wartość z 5 na -2, a funkcja H z 1 na 0.

Rys.6 przedstawia zależność współczynnika kary od względnych strat stojana, gdy ciepłe straty wirnika są parametrem. Krzywe wykreślono dla wartości parametru 0,95 i 1,10 oraz dla przypadku, gdy straty w wirniku są równe zmieniającym się stratom w stojanie.

4. Optymalizacja przy zastosowaniu kryterium uogólnionego

Przedstawiona postać uogólnionego kryterium optymalizacji pozwala na uzyskanie selektywnego działania algorytmu optymalizacji. Polega ono na tym, że w zakresie strat ciepłych stojana i wirnika mniejszych niż znamionowe, minimalizowane są straty całkowite łącznie ze stratami w układzie zasilania. Współczynnik kary w tym zakresie ma stałą wartość równą 1. Przy większych stratach stojana lub wirnika minimalizowane są przede wszystkim straty w bardziej obciążonej części silnika. Uzyskuje się to dzięki współczynnikowi kary, który przy zmianie strat ciepłych stojana lub wirnika, w zakresie od znamionowych do 1,2 znamionowych, wzrasta od 1 do 2.

Skutki działania algorytmu minimalizującego wartość uogólnionego wskaźnika jakości pokazano na rys.7, który odpowiada rysunkowi 3. Na rys.7 pominięto linie strat w silniku (linie motor), a naniesiono linie tych samych wartości uogólnionego kryterium optymalizacji (linie przerywane - generalized criterion). Widać, że linie uogólnionego kryterium pokrywają się z liniami dla strat łącznych (total) w obszarze, w którym straty w wirniku (a także stojanie) są poniżej strat znamionowych. Minimalizacja uogólnionego wskaźnika oznacza w tym przypadku także minimalizację strat całkowitych w układzie. W obszarze powyżej 1,2 strat znamionowych wirnika (lub stojana) linie uogólnionego kryterium znów pokrywają się z "poziomicami" strat całkowitych, ale wartości kryterium są dwukrotnie wyższe niż strat, w związku ze współczynnikiem kary równym 2. W obszarze przejściowym, tj. pomiędzy liniami rotor 1 i rotor 2 (rys.7) "poziomicie" kryterium mają przebieg w przybliżeniu równoległy do "poziomic" strat ciepłych w wirniku. Oznacza to, że w tej

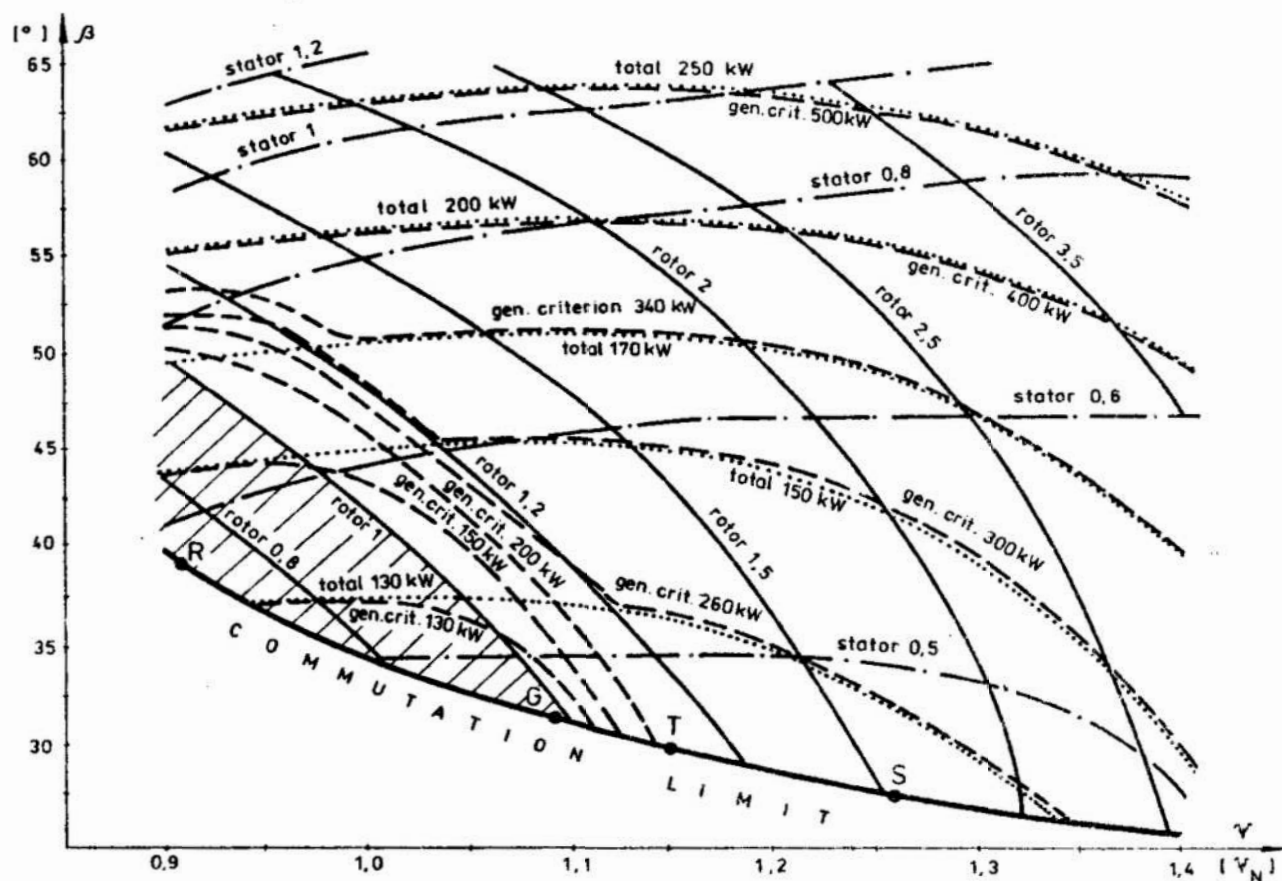


Rys. 6. Zależność współczynnika kary od strat w stojanie przy stratach w wirniku jako parametrze

strefie algorytm optymalizacji wybierze kierunek zbliżony do kierunku na minimum strat w wirniku. Zadanie algorytmowi optymalizującemu funkcji celu w postaci (3) doprowadziło do punktu G na rys. 7. Widać, że punkt ten mieści się jeszcze w obszarze dopuszczalnym ze względu na straty w wirniku, ale znajduje się jednocześnie tak blisko, jak to było możliwe, punktu T minimalnych strat w układzie. W przypadku z rys. 7 straty w stojanie nie odgrywają roli, gdyż w całym obszarze istotnym z punktu widzenia optymalizacji, są one mniejsze od znamionowych. Kryterium (3) równoprawnie traktuje straty cieplne w obu częściach silnika.

5. Podsumowanie

Rozpatrzono zagadnienie optymalizacji statycznej pracy układu z SP, w którym minimalizuje się straty całkowite. Dodatkowym wymaganiem, mającym charakter ograniczenia, które jednak może być przekraczane w pewnym zakresie, jest wymaganie dotyczące strat cieplnych w stojanie i wirniku maszyny. Wymagania połączone w jedno kryterium uogólnione w postaci iloczynu strat łącznych i współczynnika kary za przekroczenie strat znamionowych w stojanie i wirniku. Współczynnik ten ma nasyczone wartości - dolną na poziomie 1, gdy straty w obu częściach są poniżej znamionowych i górną na poziomie 2, gdy choćby jedno z tych strat przekracza o 20% wartość znamionową. W przedziale strat od 100% do 120% współczynnik kary wzrasta wg dwuwykładniczej funkcji wymagań. Tak skonstruowane kryterium sprowadza optymalizację przy wielu wymaganiach do jednokryterialnej. Działa ono selektywnie, tj. minimalizuje straty łączne w układzie, a przy zagrożeniu cieplnym silnika, minimalizuje straty w zagrożonej części.



Rys. 7. Linie tych samych wartości kryterium uogólnionego dla przypadku z rys.3

Literatura

1. Harrington E.C., *The desirability function*, in "Industry quality control", Vol 21, No 10, April 1965, p.494-498.
2. Kataoka T., Nishikata S., Tamai S., *Performance analysis of a constant-horse-power control system for a self-controlled synchronous motor*, First European Conference on Power Electronics and Applications, Proceedings Vol. 2, p.3.239-3.244, Brussels, October 1985.
3. Muszyński R., *Power losses in synchronous machine fed from current source inverter*, XII Symposium Electromagnetic Phenomena in nonlinear circuits, Proceedings p.399-404, Poznań, May 1994.
4. Muszyński R., Kaczmarek T., *Effect of control parameters on the losses in system with self-controlled synchronous motor in steady-state operation*, Conference on Modern Control for Power Electronics and electrical Drives, Vol. II, p.324-331, Łódź- Dobieszków, October 1993.
5. Strönsch V.E., *Ein Spannungsrichterantrieb mit Klauenpolmaschine und lagegeberloser Regelung*, Doktor - Ingenieur genehmigte Dissertation, Technische Universität Berlin, 1988.

Streszczenie

W referacie przedstawiono potrzebę uwzględnienia wielu wymagań przy optymalizacji statycznej pracy układu napędowego z silnikiem przekształtnikowym. Wymagania dotyczące minimalizacji strat w całym układzie, w maszynie lub jej częściach połączono przy pomocy uogólnionej funkcji wymagań. Kryterium ma postać iloczynu strat łącznych i współczynnika kary za przekroczenie strat znamionowych w stojanie lub wirniku. Pokazano efekt działania algorytmu minimalizującego uogólniony wskaźnik.

THE GENERALIZED CRITERION FOR STEADY STATE OPERATION OF SELF COMMUTATED SYNCHRONOUS MOTOR

Summary

In the paper the need of consideration of many requirements for steady-state optimization of self-commutated synchronous motor drive is presented. The requirements relative to minimization of losses in all system, in the machine or in its parts have been united in one criterion, using the generalized desirability function. The criterion has a form of product of total losses and the coefficient, which is a penalty, if the rated losses in stator or rotor are exceeded. The effect of the generalized criterion minimalization is showed.

OPTIMALIZACJA WEKTOROWA TRÓJWARSTWOWYCH POWŁOK WALCOWYCH

1. Wstęp

Typowa konstrukcja trójwarstwowa, składająca się z dwóch warstw nośnych przedzielonych warstwą wypełniającą, wykazuje wiele korzystnych cech. Zalety tych konstrukcji zostały omówione w pracach [1] i [2]. Ze względu na swoją budowę konstrukcje trójwarstwowe są predestynowane do projektowania z wykorzystaniem metod optymalizacyjnych. W przypadku stosowania optymalizacji skalarnej, naturalnym kryterium optymalizacyjnym jest masa konstrukcji, natomiast w przypadku stosowania optymalizacji wektorowej, oprócz masy, dalszymi kryteriami mogą być na przykład sztywność konstrukcji, jej własności izolacyjne i inne parametry, charakteryzujące własności użytkowe konstrukcji. Szczegółowo problemy te zostały omówione w pracy [1]. Wyniki obliczeń numerycznych optymalizacji skalarnej i wektorowej przedstawiono w pracach [1] i [2]. Niniejsza praca omawia zagadnienia związane z przestrzenią zmiennych decyzyjnych oraz z przestrzenią kryterialną.

2. Zadanie optymalizacji wektorowej

Zadanie optymalizacji wektorowej trójwarstwowej powłoki walcowej (rys. 1) w oparciu o koncepcję Pareto, sformułowane zostało następująco:

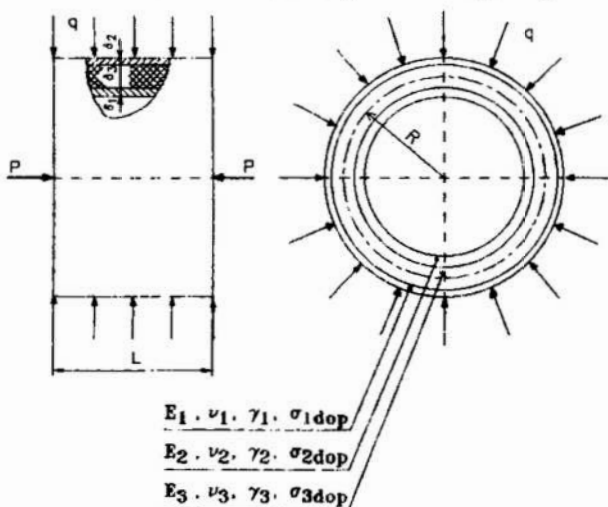
$$\bar{Q}(\bar{x}) = [Q_1(\bar{x}), Q_2(\bar{x})] = (1 - WAGA) Q_1(\bar{x}) + WAGA Q_2(\bar{x}) \rightarrow \text{minimum}, \quad (1)$$

gdzie: $\bar{Q}(\bar{x})$ - wektorowa funkcja celu,

\bar{x} - wektor zmiennych decyzyjnych (grubości warstw nośnych powłoki),

$Q_1(\bar{x}), Q_2(\bar{x})$ - odpowiednio masa i podatność powłoki,

WAGA - współczynnik wagi kryteriów, $0 \leq WAGA \leq 1.0$.



Rys. 1 Model powłoki trójwarstwowej
 Fig. 1 Model of the sandwich shell

Wektorowa funkcja celu została za pomocą współczynnika wagi przekształcona w funkcję skalarną. W oparciu o koncepcję minimum w sensie Pareto, za pomocą programu MOST [1] dla różnych wartości współczynnika WAGA otrzymano zbiór rozwiązań kompromisowych. Zbiór ten, po przyjęciu dodatkowego kryterium oceny w postaci funkcji preferencji, pozwala na wybór rozwiązania optymalnego, najlepszego.

Masa powłoki obliczana była z następującej zależności:

$$Q_1(\bar{x}) = 2\pi RL(\delta_1\gamma_1 + \delta_2\gamma_2 + \delta_3\gamma_3) \quad [\text{kg}], \quad (2)$$

podatność powłoki (odwrotność sztywności na zginanie) wynosiła:

$$Q_1(\bar{x}) = \frac{1}{E \delta_1 \delta_2 \left(\delta_3 + \frac{\delta_1 + \delta_2}{2} \right)^2} \left[\frac{1}{\text{MNm}} \right] \cdot \frac{1}{(1-\nu^2)(\delta_1 + \delta_2)}. \quad (3)$$

Zbiór ograniczeń zawierał 11 warunków:

1. Warunek stateczności:

$$P_{\text{kr dop}} \geq P, \quad P_{\text{kr dop}} = \frac{\gamma \eta}{n} P_{\text{kr}} \quad (4)$$

gdzie: P_{kr} - obciążenie krytyczne dla danej siły ściskającej P , obliczane klasycznymi metodami liniowej teorii stateczności powłok,
 γ - współczynnik redukcyjny uwzględniający losowe imperfekcje [1],
 η - współczynnik redukcyjny dla wybożenia niesprężystego, $\eta = 1$,
 n - współczynnik bezpieczeństwa, $n = 1,25$.

2 - 4. Warunki wytrzymałościowe dla naprężeń w poszczególnych warstwach powłoki:

$$\sigma_{\text{red } i} \leq \sigma_{\text{dop } i} \quad i = 1, 2, 3. \quad (5)$$

5. Warunek ograniczający ze względu na ważność wzorów do obliczania obciążeń krytycznych:

$$\frac{R}{\delta_1 + \delta_2 + \delta_3} > 20. \quad (6)$$

6. Warunek ważności wzorów do obliczania ugięć powłoki:

$$bL > 3, \quad b - \text{współczynnik [1]}.$$

7 - 11. Warunki technologiczne i konstrukcyjne:

$$0,1 \text{ mm} \leq \delta_{1,2} \leq 3,0 \text{ mm}, \quad \delta_3 \geq 5 \text{ mm}.$$

Przedstawione powyżej warunki zostały szczegółowo omówione w pracach [1] i [2].

Dla rozwiązania zadania optymalizacyjnego wykorzystany został program MOST [1], pracujący z ciągłym i dyskretnym zbiorem zmiennych decyzyjnych. Program MOST wykorzystując koncepcję minimum Pareto pozwala na otrzymanie zbioru rozwiązań kompromisowych (Pareto-optymalnych). Przyjmując dodatkowe kryterium oceny w postaci funkcji preferencji (np. min-max, kryterium globalne, funkcje użyteczności itp.) ze zbioru tego wybiera się rozwiązanie najlepsze, optymalne. Program MOST opisany został w [1].

W trakcie obliczeń numerycznych przeprowadzana była minimalizacja obciążenia krytycznego względem parametrów opisujących powierzchnię powłoki w chwili utraty stateczności [1].

3. Wyniki obliczeń numerycznych

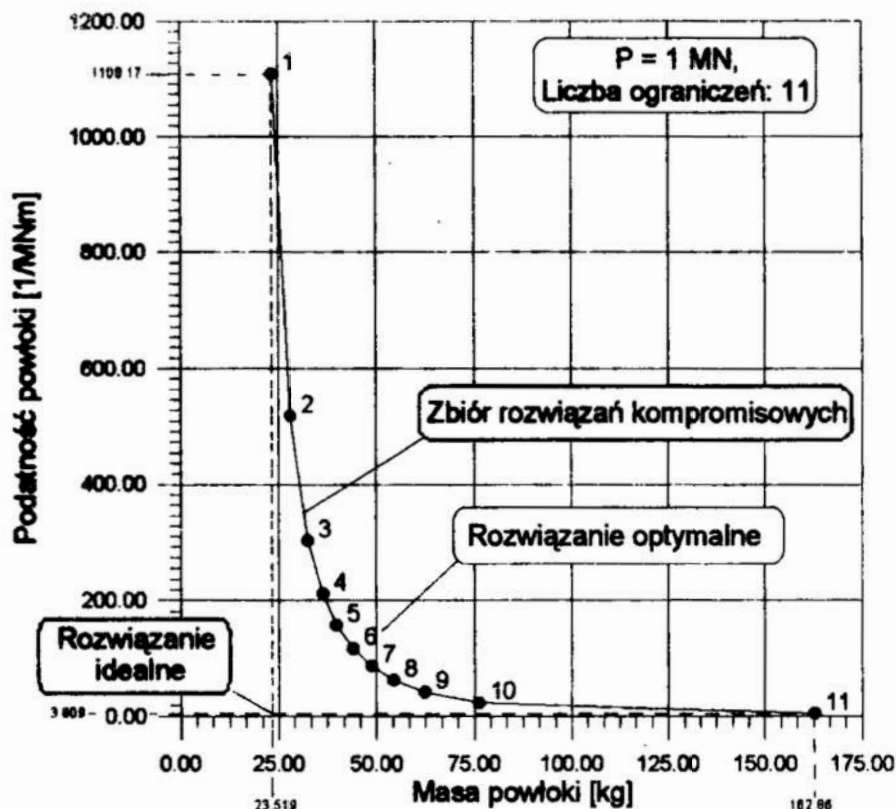
Do obliczeń numerycznych przyjęto powłokę walcową o promieniu powierzchni podstawowej $R = 1$ m, długości $L = 1$ m, wykonaną z duraluminiowych warstw nośnych ($E = 7,06 \cdot 10^4$ MPa, $\nu = 0,33$, $\gamma = 210$ kg/m³, $\sigma_{dop} = 195$ MPa) oraz z warstwy nośnej wykonanej z piankowego tworzywa sztucznego ($E = 53$ MPa, $\nu = 0$, $\sigma_{dop} = 1,8$ MPa). Powłoka była obciążona osiową siłą ściskającą $P = 1$ MN.

Wyniki obliczeń numerycznych dla różnych wartości parametru WAGA przedstawiono na rysunku 2 oraz w Tabelicy 1. Ponieważ program MOST korzysta z dyskretnych i ciągłych zmiennych decyzyjnych, w Tabelicy 1 w górnych wierszach przedstawiono rozwiązania dyskretne (grubości warstw są zgodne ze znormalizowanymi grubościami blach), natomiast w dolnych wierszach rozwiązania otrzymane z ciągłych zbiorów zmiennych decyzyjnych, pozwalające na określenie aktywnych warunków ograniczających (wartości podane w tabelicy zostały zaokrąglone).

TABELICA 1

OPTYMALNE PARAMETRY POWŁOKI, $P = 1$ MN [1]

Nr	W A G A	Optymalne grubości warstw [mm]			Q_1 [kg]	Q_7 [1/MPa]	P_{kr} [MN]	$\sigma_{1,2red}$ [MPa]	Aktywny warunek ograniczający
		δ_1	δ_2	δ_3					
1.	0	0.5	0.4	7.0	24.957	1023.34	1.092	175.89	$P_{kr}, \sigma_{1,2}$
		0.4042	0.4067	7.0904	23.519	1108.17	1.000	195.00	
2.	0.1	0.5	0.4	10.0	28.915	520.118	1.600	175.48	$\sigma_{1,2}$
		0.4031	0.4051	10.5884	28.088	516.964	1.535	195.00	
3.	0.2	0.5	0.4	14.0	34.193	272.019	2.363	174.94	$\sigma_{1,2}$
		0.4016	0.4041	14.0112	32.560	301.628	2.145	195.00	
4.	0.3	0.5	0.4	16.0	36.832	209.895	2.704	174.67	$\sigma_{1,2}$
		0.4011	0.4024	16.8668	36.290	210.707	2.560	195.00	
5.	0.4	0.5	0.4	19.0	40.790	150.140	3.205	174.27	
		0.3999	0.4016	19.6217	39.890	157.123	2.991	194.98	
6.	0.5	0.4	0.4	23.0	44.322	114.255	3.573	194.74	
		0.4371	0.4371	21.8296	44.074	116.481	3.646	178.71	
7.	0.6	0.5	0.5	24.0	49.135	84.110	4.566	156.34	
		0.4841	0.4842	24.1529	48.783	85.900	4.473	161.34	
8.	0.7	0.5	0.5	28.0	54.412	62.157	5.452	155.88	
		0.5409	0.5410	26.9707	54.485	61.654	5.566	144.40	
9.	0.8	0.6	0.6	31.0	61.864	42.133	7.080	130.11	
		0.6189	0.6188	30.8570	62.334	41.173	7.210	126.23	
10	0.9	0.8	0.8	37.0	76.768	22.084	10.576	97.77	
		0.7570	0.7570	37.8208	76.349	22.407	10.283	103.19	
11	1.0	2.9	2.9	44.0	159.36	3.957	27.726	27.29	$\delta_{1,2}, R/\delta$
		2.9998	2.9998	44.0003	162.86	3.809	28.352	28.38	



Rys. 2 Zbiór rozwiązań Pareto-optymalnych dla $P = 1 \text{ MN}$
 Fig. 2 Compromise solutions set for $P = 1 \text{ MN}$

Zbiór rozwiązań Pareto-optymalnych w przestrzeni kryterialnej przedstawiono na rysunku 2. Rozwiązaniem optymalnym, wyznaczonym w oparciu o funkcje preferencji (kryterium globalne, $p = 1, 2$ [1]) jest rozwiązanie z parametrem WAGA = 0,5 (punkt nr 6). Wyniki przedstawione w Tabelicy 1 oraz na rysunku 2 stanowią podstawę do analizy przestrzeni zmiennych decyzyjnych i przestrzeni kryterialnej.

Rysunek 3 przedstawia w przestrzeni zmiennych decyzyjnych zbiór rozwiązań dopuszczalnych, wyznaczony przez 11 warunków ograniczających. Na rysunku tym zaznaczono zbiór rozwiązań Pareto-optymalnych (punkty 1 - 11), będący odwzorowaniem punktów z przestrzeni kryterialnej (porównaj rysunek 2). Dla lepszego przedstawienia tego zbioru, na rysunkach 4 i 5 przedstawiono poziome przekroje tego obszaru, a na rysunkach 6 i 7 przekroje pionowe. Na rysunku 7 w przekroju pionowym obszaru rozwiązań dopuszczalnych widać zbiór rozwiązań kompromisowych, którego część pokrywa się z konturem obszaru dopuszczalnego (punkty 1 - 5) wyznaczonym przez warunek stateczności i warunki wytrzymałościowe oraz przez warunki geometryczne (punkt 11).

Wyniki przedstawione na rysunkach 3 - 8 otrzymano korzystając z dyskretnego zbioru zmiennych decyzyjnych. Fakt ten jest przyczyną drobnych wypukłości, które można zauważyć na rysunkach 5 i 6. Rysunek 8 przedstawia obszar celów osiągalnych (przebieg kryterialny), którego fragmentem jest zbiór rozwiązań Pareto-optymalnych pokazanych na rysunku 2. Każdy punkt tego obszaru jest odwzorowaniem odpowiedniego punktu z przestrzeni zmiennych decyzyjnych. Współrzędne kilku wybranych punktów z rysunku 3 oraz z rysunku 8 pokazano w Tabelicy 2.

TABLICA 2

**WYBRANE PUNKTY PRZESTRZENI ZMIENNYCH DECYZYJNYCH
I PRZESTRZENI KRYTERIALNEJ [1]**

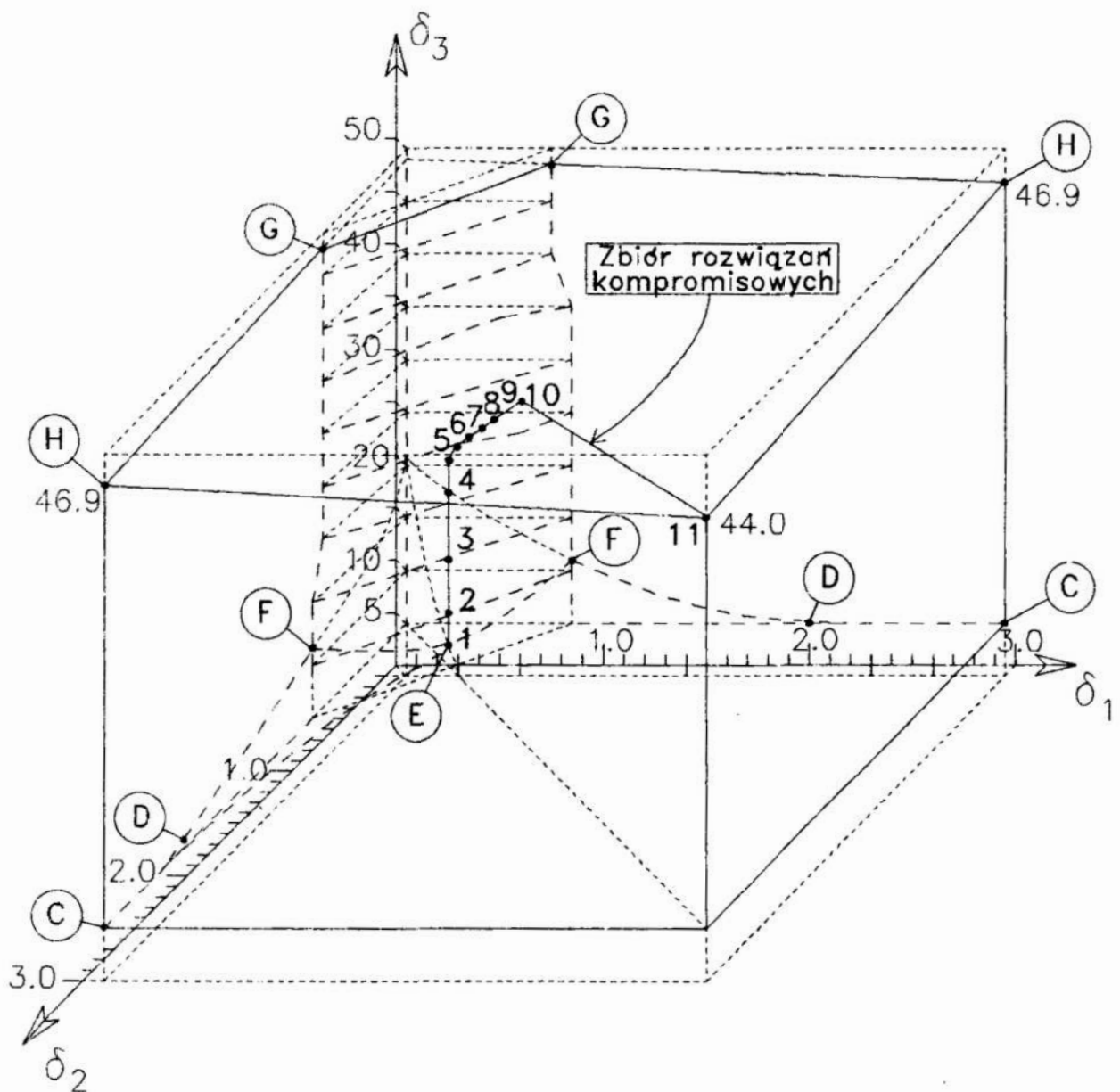
Punkt	Grubości warstw [mm]			Q ₁ [kg]	Q ₂ [1/MNm]	Uwagi
	δ_1	δ_2	δ_3			
1.	0.5	0.4	7.0	24.967	1023.344	Optymalizacja skalarna (masa)
6.	0.4	0.4	23.0	44.322	115.255	Rozwiązanie optymalne (optymalizacja wektorowa)
11.	2.9	2.9	44.0	159.367	3.957	Optymalizacja skalarna (podatność)
A	22.5	22.5	5.0	792.624	1.484	
B	0.1	44.9	5.0	792.624	167.272	
C	0.1	3.0	5.0	60.746	3040.042	
D	0.1	2.0	6.0	44.598	2666.446	
E	0.6	0.6	5.0	27.558	1341.604	
F	0.1	0.8	11.0	30.235	1083.087	
G	0.1	0.8	49.1	80.506	57.834	
H	0.1	3.0	46.9	116.032	55.562	
P	25.0	25.0	0	873.363	1.616	Powłoka jednowarstwowa

Ważnym wnioskiem wpływającym z analizy rysunku 8 jest fakt, że zbiór rozwiązań Pareto-optymalnych w przestrzeni kryterialnej jest zbiorem wypukłym. Zastosowanie do rozwiązania zadania optymalizacji wektorowej optimum w sensie Pareto jest więc uzasadnione. Z rysunku 8 widać też wyraźnie, że ograniczenia nałożone na zmienne decyzyjne ograniczają zbiór rozwiązań Pareto-optymalnych, wpływają przez to na położenie rozwiązania idealnego i dalej mają wpływ na rozwiązanie optymalne.

Na rysunku 8 zaznaczono też położenie punktu P, odpowiadającemu powłoce jednowarstwowej ($\delta_3 = 0$), dla której rozwiązanie znajduje się poza obszarem dopuszczalnym.

Literatura

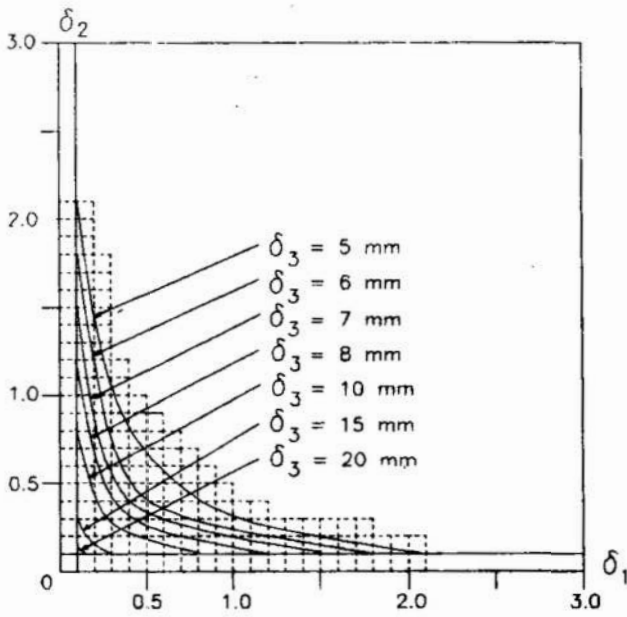
- [1] Ostwald M.: Optymalne projektowanie trójwarstwowych konstrukcji powłokowych. Politechnika Poznańska, Rozprawy, nr 290, 1993.
- [2] Ostwald M.: Optymalizacja wektorowa cienkościennych konstrukcji trójwarstwowych. Zeszyty Naukowe Wydziału Mechanicznego, Nr 16, WSI Koszalin, 1994, str. 120-135.



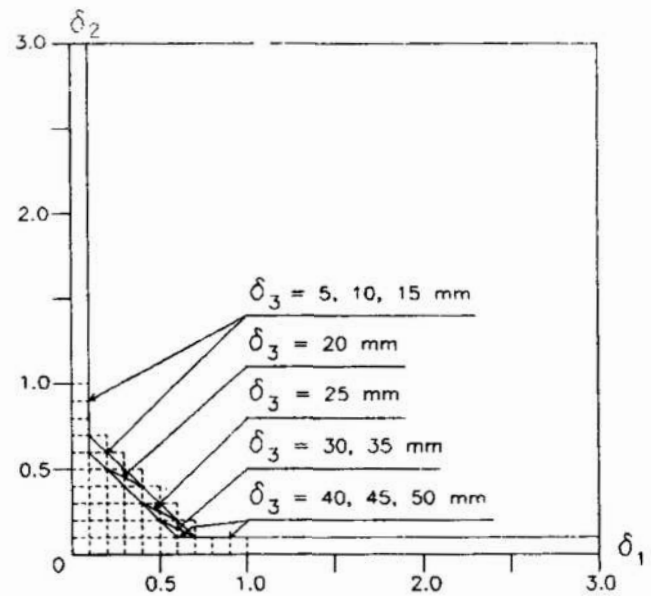
Rys. 3 Zbiór rozwiązań dopuszczalnych w przestrzeni zmiennych decyzyjnych
 Fig. 3 Set of feasible solutions in the decision space

Streszczenie

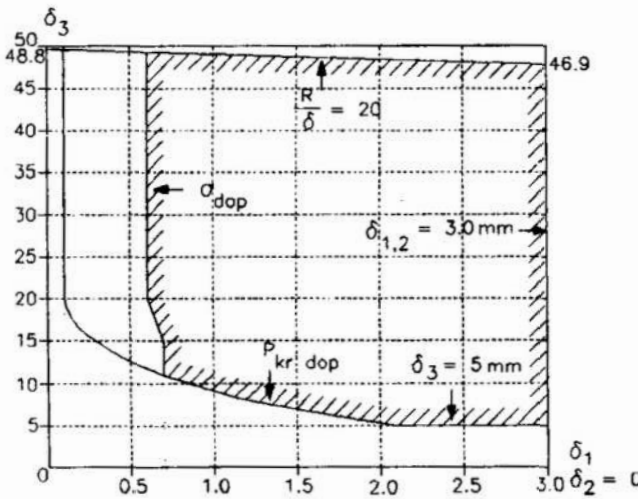
W pracy przedstawiono zadanie optymalizacji wektorowej trójwarstwowej powłoki walcowej poddanej osiowemu ścisnaniu. Kryteriami optymalizacyjnymi są masa powłoki oraz podatność na zginanie. Zbiór ograniczeń zawiera 11 warunków, w tym warunek stateczności powłoki, warunki wytrzymałościowe dla poszczególnych warstw, warunki technologiczno-konstrukcyjne i inne. Obliczenia numeryczne zrealizowano za pomocą programu MOST, wykorzystującego koncepcję Pareto dla znalezienia zbioru rozwiązań kompromisowych. W pracy przeprowadzono analizę przestrzeni zmiennych decyzyjnych oraz przestrzeni kryterialnej, wyznaczając obszary rozwiązań dopuszczalnych oraz celów osiągalnych. Wyniki obliczeń numerycznych przedstawiono w postaci tablic i rysunków.



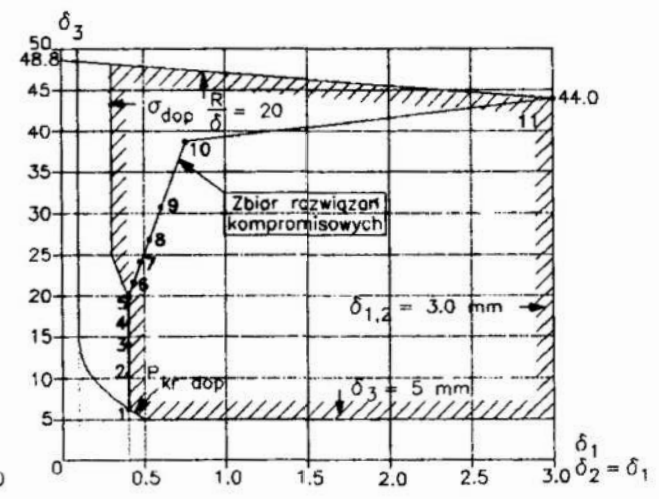
Rys. 4 Przekrój poziomy zbioru rozwiązań dopuszczalnych dla warunku stateczności
 Fig. 4 Level section of the feasible domain for stability condition



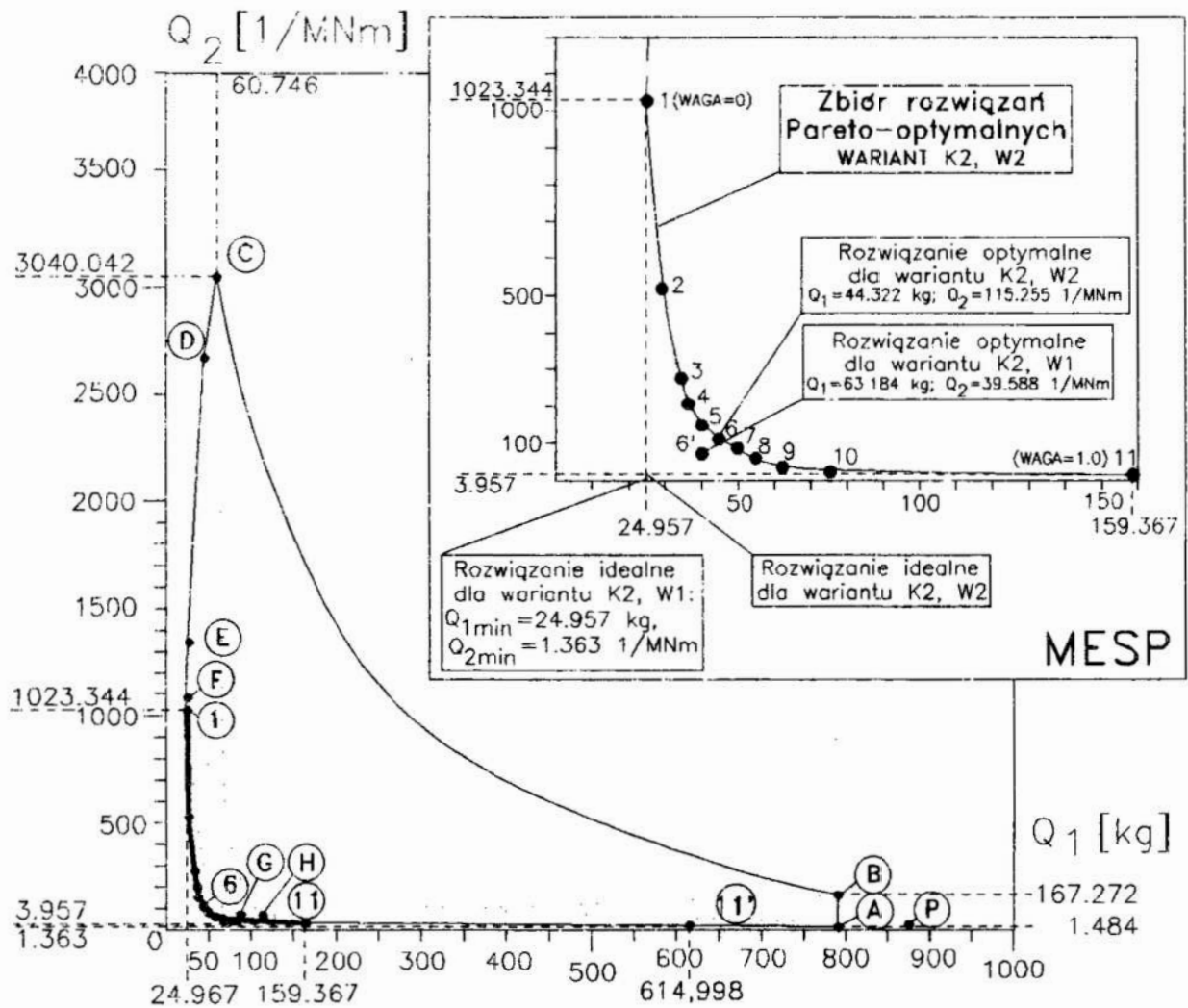
Rys. 5 Przekrój poziomy zbioru rozwiązań dopuszczalnych dla warunków wytrzymałościowych
 Fig. 5 Level section of the feasible domain for strength conditions



Rys. 6 Przekrój pionowy zbioru rozwiązań dopuszczalnych dla $\delta_2 = 0$
 Fig. 6 Vertical section of the feasible domain for $\delta_2 = 0$



Rys. 7 Przekrój pionowy zbioru rozwiązań dopuszczalnych dla $\delta_1 = \delta_2$
 Fig. 7 Vertical section of the feasible domain for $\delta_1 = \delta_2$



Rys. 8 Przestrzeń kryterialna z zaznaczonym zbiorem rozwiązań Pareto- optymalnych
 Fig. 8 Objective space with the Pareto-optimal solutions set

VECTOR OPTIMIZATION OF SANDWICH CYLINDRICAL SHELLS

Summary

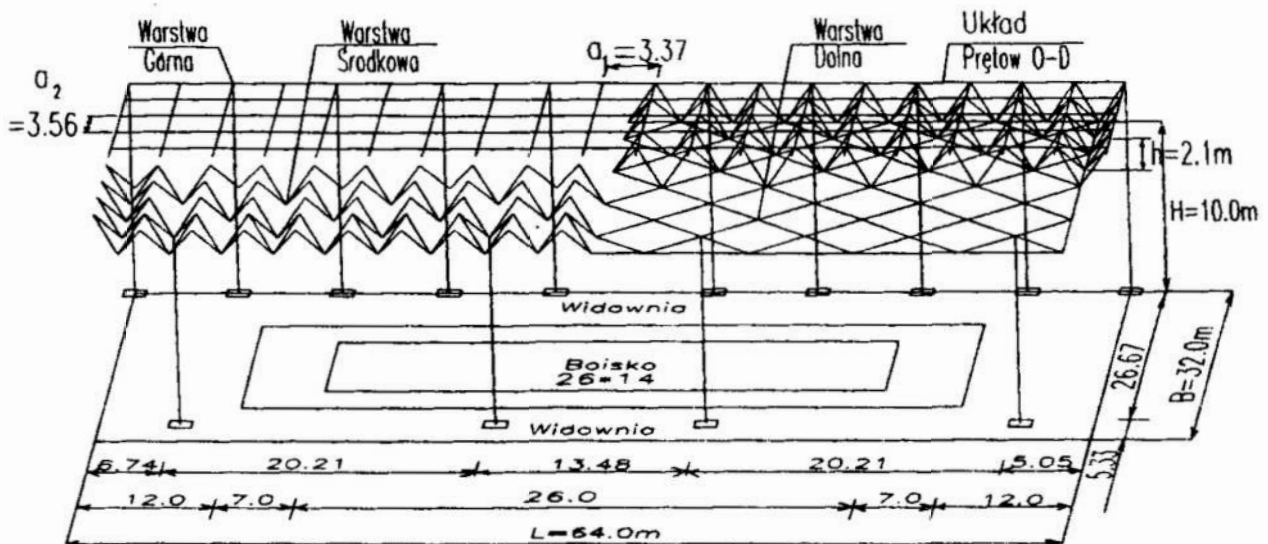
In the paper multicriteria optimization problem of the sandwich cylindrical shells subjected to axial compression is presented. In the problem weight of the shell is the first objective, and bending flexibility is the second one. Set of constraints contains 11 conditions, including stability condition, strenght conditions for each layer, constructional and technological requirements and so on. Numerical results were obtained with the help of the MOST program, based on the Pareto concept of optimality. In the paper analysis of the decision space is presented, and the feasible domain are generated. Results of calculation are presented in the form of tables and figures.

Witold M. PACZKOWSKI *
 Jarosław KARŁOWSKI *
 Stanisław OLDENBURG *

ROZPROSZONA OPTYMALIZACJA KONSTRUKCJI HALI SPORTOWEJ

1. Wprowadzenie

Proces kształtowania hali sportowej sprowadza się do cyklicznego wyboru szeregu wielkości opisujących konstrukcję, dla których przeprowadza się projektowanie zgodnie z obowiązującymi przepisami. Analizowaną konstrukcją (rys.1) podzielono myślowo na współpracujące ze sobą elementy: kratownicę przestrzenną, słupy, fundamenty, pokrycie dachowe i konstrukcję ścian [3]. Ze względu na dużą wielkość zadania przyjęto metodę optymalizacji rozproszonej. Wyróżniono zmienne decyzyjne lokalne, dotyczące poszczególnych elementów obiektu oraz globalne, wspólne dla co najmniej dwóch elementów. Poszczególne zadania rozwiązywane są w ustalonej kolejności. Mogą to być zadania optymalizacji lub polioptymalizacji. Wartości zmiennych decyzyjnych odpowiadające ekstremum funkcji celu lub rozwiązaniu preferowanemu w jednym zadaniu stają się punktami startowymi w kolejnym zadaniu i są przyjmowane jako parametry optymalizacji. O kolejności formułowanych zadań optymalizacji lub polioptymalizacji decyduje konstruktor, wykorzystując swoje doświadczenia w projektowaniu podobnych



Rys. 1. Schemat konstrukcji i funkcji hali sportowej
 Fig. 1. The construction and application scheme of a sports hall

*Politechnika Szczecińska, Instytut Inżynierii Lądowej, Al. Piastów 50, 70-311 Szczecin

obiektów. Wskazana jest współpraca z inżynierem wiedzy, kontrolującym poprawność sformułowanych zadań i dobierającym efektywne metody optymalizacji. Możliwy jest, zwłaszcza w końcowej fazie optymalizacji rozproszonej, powrót do wcześniej analizowanych zadań cząstkowych. Następuje wtedy uściślenie wartości zmiennych decyzyjnych i poprawa wartości analizowanych funkcji celu.

Przedmiotem optymalizacji jest konstrukcja hali sportowej o wymiarach $B \times L = 32 \times 64$ m, w której przewidziano boisko do gry w koszykówkę 26×14 m (rys.1). Minimalna wysokość hali mierzona od poziomu posadzki do dolnej warstwy konstrukcji przekrycia wynosi 10 m. Ze względu na funkcję obiektu istnieje możliwość umieszczenia słupów konstrukcyjnych w obrębie widowni lub pomieszczeń pomocniczych tj. około 6 m od ścian zewnętrznych hali. Obudowa dachu lekka - dwie warstwy blachy faldowej ocieplonej wełną mineralną twardą grubości 10 cm. Ściany zewnętrzne wykonane z płyt PW8/B. Obiekt zlokalizowano w pierwszej strefie obciążenia wiatrem i śniegiem. Budynek posadowiony jest na piaskach średnich, małowilgotnych, o stopniu zagęszczenia $I_D = 0,6$.

2. Przebieg rozproszonej optymalizacji hali

W pracy analizuje się elementy konstrukcji hali sportowej: kratownicę przestrzenną, konstrukcję wsporczą, fundamenty i obudowę hali. Do wymiarowania kratownicy zastosowano przekroje z rur walcowanych na gorąco, ze stali gatunku R35.

Poddano analizie 5 zmiennych decyzyjnych kształtujących geometrię kratownicy, elementy konstrukcji wsporczej i obudowy hali: odległość międzywęzłową a , odległość międzywarstwową h , liczebność katalogu przekrojów pierścieniowych kratownicy t , sposób podparcia kratownicy p , typ siatki struktury s . Wektor zmiennych decyzyjnych x można opisać następująco

$$x = [a , h , t , p , s]^T.$$

W zadaniu optymalizacji rozproszonej sformułowano cztery zadania cząstkowe. Kryterium wyboru rozwiązania optymalnego zmienia się w zależności od rozwiązywanego zadania. W zadaniu A1 polioptymalizacji jako zmienne decyzyjne przyjęto odległość międzywęzłową oraz odległość międzywarstwową kratownicy. Pozostałe zmienne zadania globalnego potraktowano jako parametry optymalizacji, przyjmując podparcie kratownicy w warstwie dolnej na 6 słupach rozgałęzionych, typoszereg składający się z 9 przekrojów oraz siatkę ortogonalną. Wysokość struktury dla płaskich przekryć przetoowych waha się w granicach od $1/9$ do $1/35$ rozpiętości [1]. W zadaniu przyjęto zmienność odległości międzywarstwowej h od 1,2 m do 3,6 m z krokiem co 30 cm

$$h \in \underline{h} = \{1.2, 1.5, 1.8, 2.1, 2.4, 2.7, 3.0, 3.3, 3.6\}.$$

Dyskretny charakter modułu kratownicy wynika z podziału szerokości hali na parzystą liczbę przedziałów od 8 do 20

$$a \in \underline{a} = \{1.60, 1.78, 2.00, 2.29, 2.67, 3.20, 4.00\}.$$

Ze względów konstrukcyjnych wprowadzono ograniczenie minimalnego kąta $\alpha \geq \pi/6$ i maksymalnego kąta $\alpha \leq \pi/3$ nachylenia krzyżulców do płaszczyzn warstw kratownicy.

Celem zadania jest określenie zbioru rozwiązań niezdominowanych dla zmiennego modułu i wysokości struktury w funkcji masy kratownicy $f_1(\mathbf{x})$ oraz masy stalowych elementów obudowy hali i konstrukcji wsporczej $f_2(\mathbf{x})$ przypadającej na m^2 powierzchni zabudowy hali

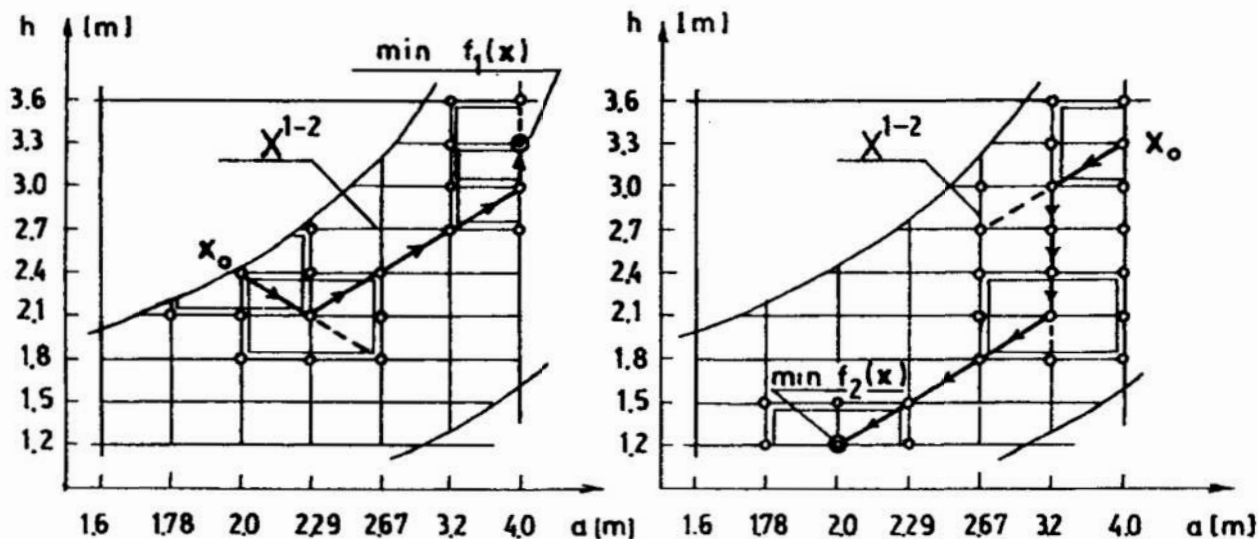
$$f_1(\mathbf{x}) = [f_1(\mathbf{x}), f_2(\mathbf{x})]^T,$$

$$f_1(\mathbf{x}) = f_1(a, h, t = 9, p = P6, s = OO) = \frac{G_{pr} + G_w}{B \cdot L} = 1,1 \frac{\gamma}{32 \cdot 64} \sum_{k=1}^{P(\mathbf{x})} l_k \cdot A_k,$$

$$f_2(\mathbf{x}) = f_2(a, h, t = 9, p = P6, s = OO) = \frac{G_b(\mathbf{x}) + G_p(\mathbf{x}) + G_{sl}(\mathbf{x}) + G_r(\mathbf{x}) + G_s(\mathbf{x})}{32 \cdot 64},$$

gdzie:

- G_{pr} - masa prętów kratownicy,
- G_w - masa węzłów kratownicy - przyjęto $G_w = 0,1 G_{pr}$,
- A_k, l_k - długość i pole przekroju k-tego pręta,
- γ - masa właściwa stali, $\gamma = 7850 \text{ kg/m}^3$,
- $P(\mathbf{x})$ - liczba prętów kratownicy,
- $G_b(\mathbf{x})$ - masa nośnej blachy fałdowej pokrycia hali,
- $G_p(\mathbf{x})$ - masa płatwi hali wykonanych z 2 kątowników,
- $G_{sl}(\mathbf{x})$ - masa słupów nośnych i osłonowych hali projektowanych z dwuteowników szerokostopowych,
- $G_r(\mathbf{x})$ - masa rygli ściennych wykonanych z dwóch ceowników 120 i stali w płytach ściennych hali PW8/B,
- $G_s(\mathbf{x})$ - masa stężeń ścian zewnętrznych hali projektowanych z 2 kątowników.



Rys. 2. Ścieżki poszukiwania minimum funkcji celu w zadaniu A1

Fig. 2. The minimum of a objective functions $f_1(\mathbf{x}), f_2(\mathbf{x})$ in the task A1 searching paths

Ze względu na monotoniczny charakter obu zmiennych zadanie rozwiązano metodą ortogonalno-diagonalną [5]. Na rysunku 2a przedstawiono ścieżkę poszukiwania minimum funkcji $f_1(\mathbf{x})$, na rysunku 2b funkcji $f_2(\mathbf{x})$. W wyniku otrzymano zbiór ocen niezdominowanych Y_{ND} (rys.3a) i odpowiadający mu zbiór rozwiązań niezdominowanych X_{ND} (rys. 3b).

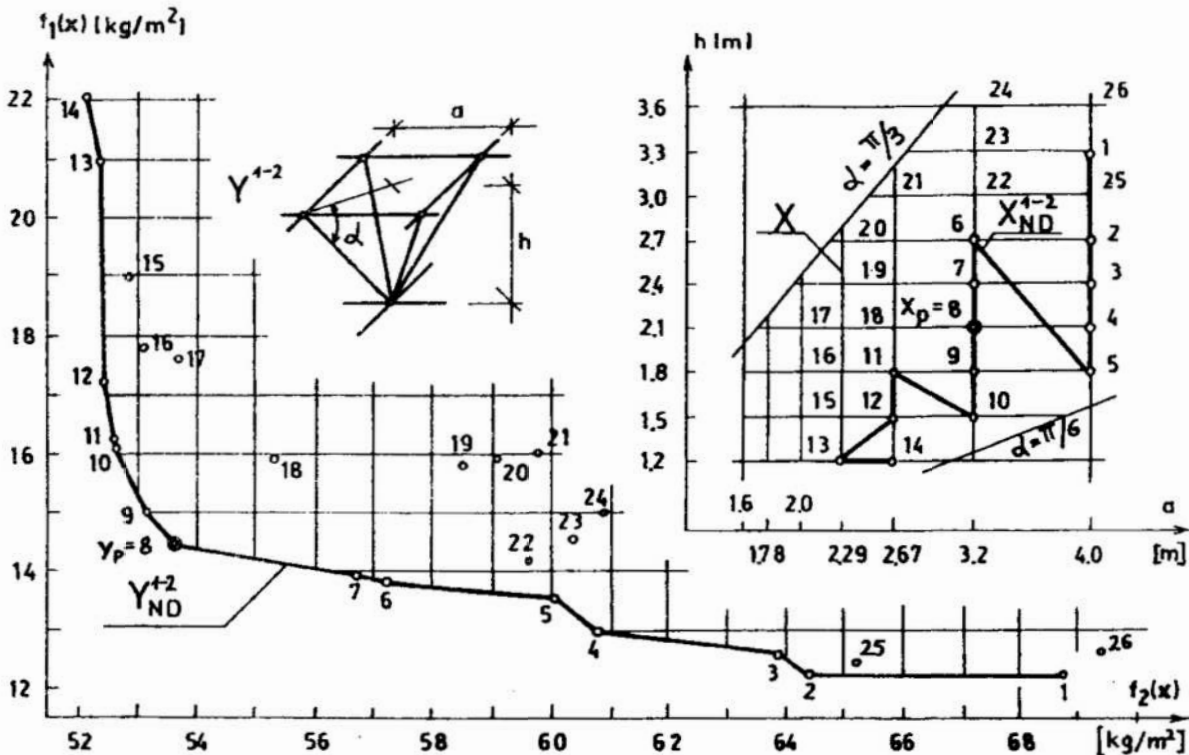
Przy wyborze rozwiązania preferowanego ze zbioru rozwiązań niezdominowanych zastosowano funkcję użyteczności, uwzględniając dodatkowo pracochłonność wykonania struktury [1,3]. Kryterium wyboru ze zbioru X_{ND} stanowi minimum funkcji

$$U_1(x_{ND}) = \frac{l_p^2 + (l_p + 1)^2}{B \cdot L} \cdot \left[1 + \left| \sin \left(\frac{\pi}{4} - \alpha \right) \right| \right] \cdot f_1(x) \cdot f_2(x),$$

gdzie:

l_p - liczba przedziałów w warstwie górnej kratownicy w zależności od przyjętego modułu, $l_p = B/a$.

Na podstawie tak sformułowanej funkcji użyteczności rozwiązaniem preferowanym okazał się przypadek, dla którego $a=3,2$ m i $h=2,1$ m.



Rys. 3. Zbiór ocen Y_{ND} i rozwiązań niezdominowanych X_{ND} zadania A1

Fig. 3. The efficient solutions set Y_{ND} and compromise solutions set X_{ND} of the task A1

W drugim zadaniu polioptymalizacji (A2) zmienną decyzyjną jest liczba przekrojów pierścieniowych w typoszeregu prętów t . Parametrami optymalizacji są: odległość międzywęzłowa i międzywarstwowa (rozwiązanie preferowane zadania A1), sposób podparcia na 6 słupach rozgałęzionych i ortogonalny typ siatki. Dobór prętów kratownicy odbywa się na bazie przekrojów pierścieniowych przedstawionych na rys. 4a. Typoszereg wyjściowy (bazowy) składa się z 33 przekrojów. Na rysunku 4b przedstawiono graficznie sposób tworzenia z katalogu bazowego katalogów pochodnych 17, 9, 5, 3 i 2 elementowych [4]

$$t \in \underline{t} = \{33, 17, 9, 5, 3, 2\}.$$

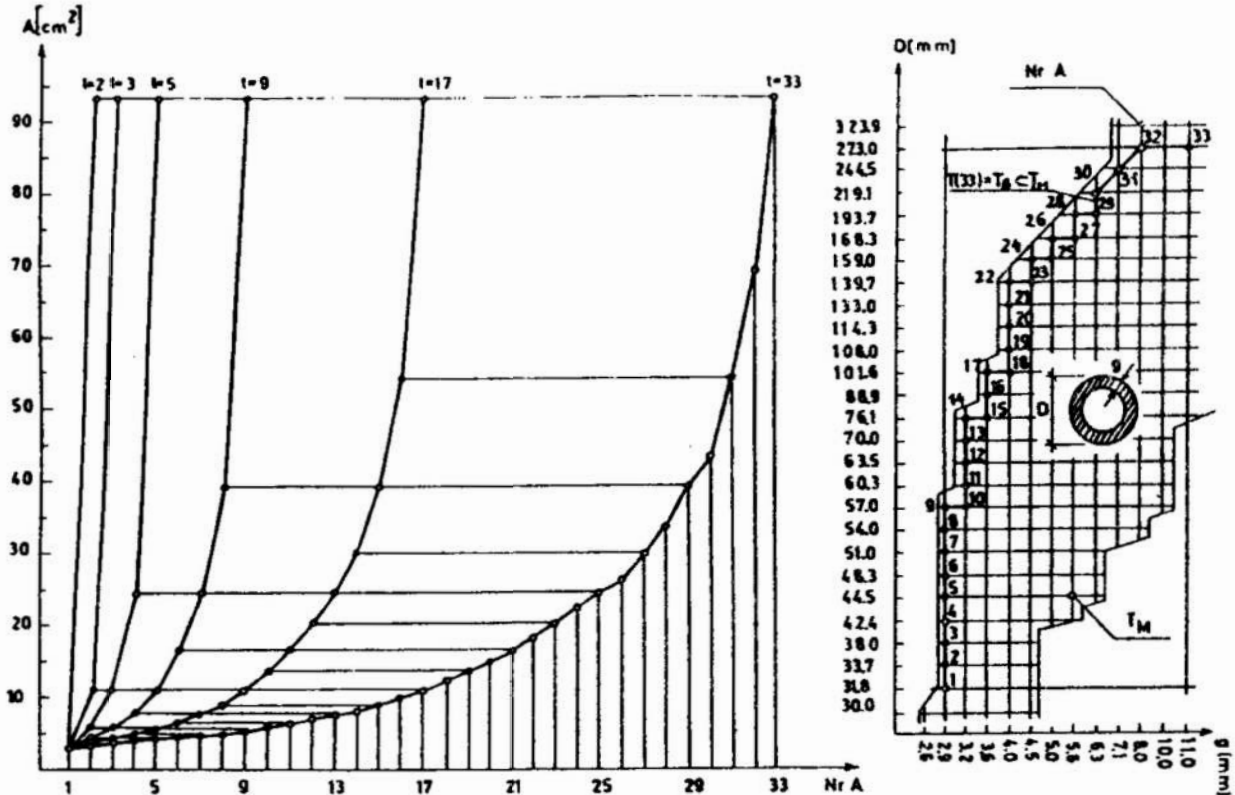
Zadanie rozwiązywano według kryterium minimum wektora funkcji celu

$$f_{II}(x) = [f_1(x), f_3(x)]^T,$$

gdzie:

$$f_3(x) = \Delta_{\max}(x) = \max_i \sqrt{u_i^2 + v_i^2 + w_i^2}$$

jest maksymalnym przemieszczeniem węzłów struktury (wypadkowa z kierunków u , v , w). Maksymalne przemieszczenie węzła kratownicy i jej masę w zależności od liczebności typoszeregu przekrojów t przedstawia rys.5



Rys. 4. Bazowy katalog przekrojów prętów T_B i tworzenie katalogów pochodnych
Fig. 4. The basic catalogue of the bar cross sections T_B and the creation of some catalogues derived from the basic catalogue

Funkcja, która pozwoliła wyłonić ze zbioru rozwiązań niezdominowanych rozwiązanie preferowane, uwzględnia dodatkowo liczbę przekrojów w typoszeregu, co wiąże się z pracochłonnością wykonania kratownicy [3]

$$U_2(x_{ND}) = (1 - 0,4e^{-0,35t}) \cdot f_1(x) \cdot f_3(x).$$

Rozwiązaniem preferowanym według przyjętego kryterium optymalizacji jest typoszereg składający się z 5 przekrojów. Na rys.5 wyróżniono ocenę preferowaną y_p .

W zadaniu A3 optymalizacji analizowano różne warianty podparcia struktury. Rozpatrywano kratownicę ortogonalną z preferowanym typoszeregiem (5-elementowym), określonym w wyniku rozwiązania zadania A2 oraz preferowanym modulem i wysokością struktury, przyjętą na podstawie analizy zadania A1. Kratownica może być podparta w warstwie górnej lub dolnej, w sposób ciągły lub punktowy, na słupach

kratowych rozwidlonych lub prostych, umieszczonych wewnątrz hali lub na jej obrzeżu. W zadaniu analizowano 12 różnych kombinacji sposobów podparcia (rys. 6)

$$p \in \underline{p} = \{P1, P2, P3, P4, P5, P6, P7, P8, P9, P10, P11, P12\}.$$

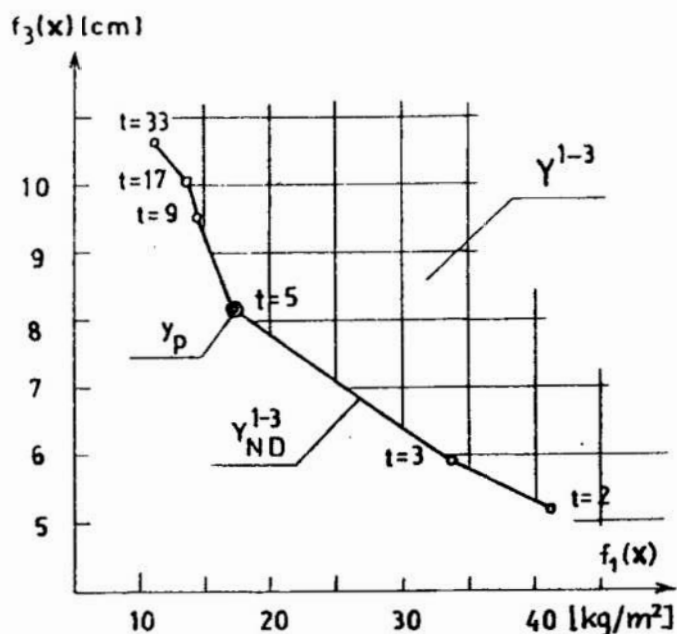
Kryterium wyboru rozwiązania optymalnego stanowi minimum kosztu porównawczego elementów konstrukcji hali - kratownicy, słupów i fundamentów - zmiennych w poszczególnych schematach podparcia. Funkcja celu ma postać

$$f_4(\mathbf{x}) = C_{41} \frac{f_1(\mathbf{x})}{f_{1P}(\mathbf{x})} + C_{42} \frac{G_{sl}(\mathbf{x})}{G_{slP}(\mathbf{x})} + C_{43} \frac{V_f(\mathbf{x})}{V_{fP}(\mathbf{x})},$$

gdzie:

C_{41}, C_{42}, C_{43} - współczynniki wag, odzwierciedlające jednostkowy udział kosztów wykonania odpowiednio: $C_{41}=0,6$ - kratownicy, $C_{42}=0,25$ - słupów, $C_{43}=0,15$ - fundamentów.

$V_f(\mathbf{x})$ - objętość betonu w stopach fundamentowych w danym sposobie podparcia, $f_{1P}(\mathbf{x}), G_{slP}(\mathbf{x}), V_{fP}(\mathbf{x})$ - wartości porównawcze masy kratownicy, słupów i objętości fundamentów przypadające na 1 m^2 rzutu hali, przyjęte dla schematu podparcia P1: $f_{1P}(\mathbf{x})=17,98 \text{ kg/m}^2, G_{slP}(\mathbf{x})=11,29 \text{ kg/m}^2, V_{fP}(\mathbf{x})=0,0074 \text{ m}^3/\text{m}^2$.



Rys. 5. Zbiór ocen niezdominowanych Y_{ND} zadania A2

Fig. 5. The efficient solutions set Y_{ND} of a task A2

Według przyjętego kryterium optymalizacji rozwiązaniem najlepszym okazał się schemat podparcia P12. Masa kratownicy na m^2 hali oraz maksymalne przemieszczenie pionowe węzła dla poszczególnych schematów podparcia przedstawiono w tabelicy 1.

Wyniki analizy zadania A3

Tablica 1

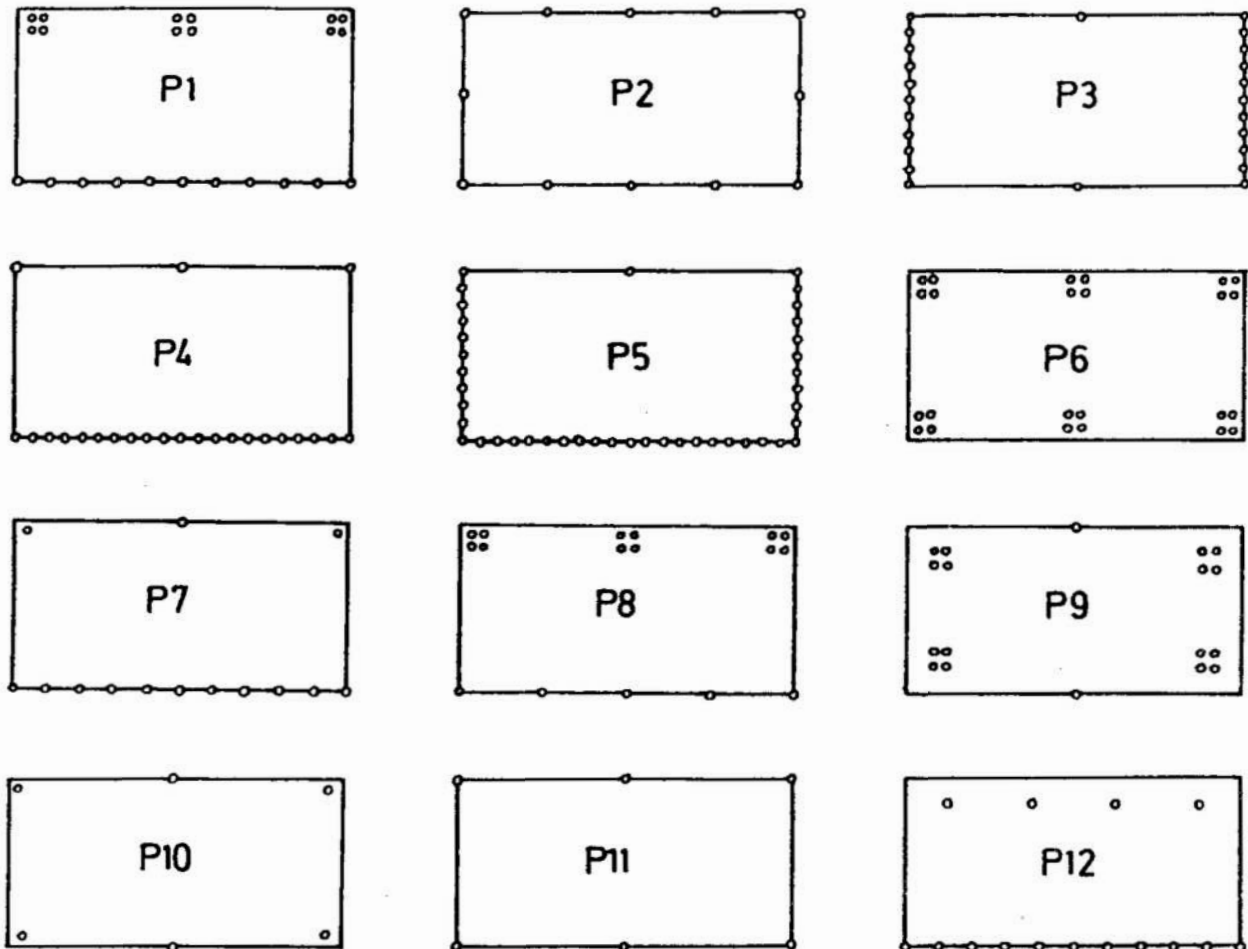
The task A3 results

Table 1

Podparcie	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12
$f_1(\mathbf{x})[\text{kg/m}^2]$	17,98	18,96	24,43	23,03	21,19	17,10	21,69	18,33	22,25	23,21	29,68	14,84
$f_3(\mathbf{x})[\text{cm}]$	8,38	8,73	8,82	9,59	8,97	8,14	9,10	8,80	8,99	10,02	9,89	6,04

Ostatnie zadanie optymalizacji A4 polega na określeniu polioptymalnego typu struktury [1,2]. Parametry opisujące konstrukcję wynikają z wcześniejszej analizy. Na rys. 7 przedstawiono pięć z ośmiu analizowanych siatek:

$$s \in \underline{s} = \{OO, OO2, DD, DD2, OD, DO, R, RK\}.$$



Rys. 6. Schematy podparcia kratownicy
Fig. 6. The schemes of a truss support

Wektor funkcji celu zadania A4 ma postać

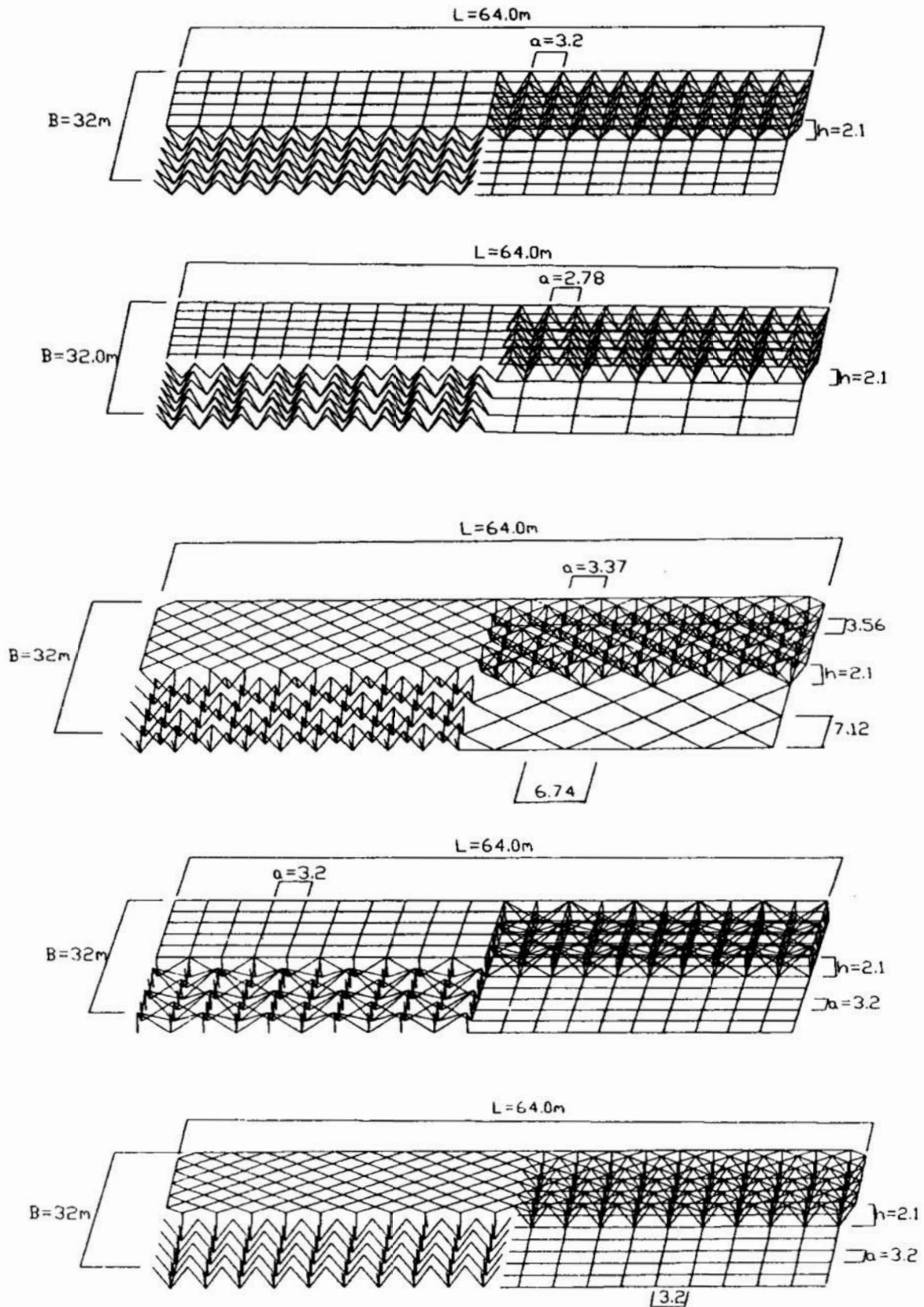
$$f_{IV}(\mathbf{x}) = [f_1(\mathbf{x}), f_2(\mathbf{x}), f_5(\mathbf{x})]^T,$$

gdzie:

$$f_5(\mathbf{x}) = W(\mathbf{x}) - \text{liczba węzłów w kratownicy.}$$

Liczba prętów w kratownicy waha się od 1014 w siatce ortogonalno-diagonalnej do 3080 w diagonalnej. Zbiór ocen niezdominowanych według kryterium minimum przyjętych funkcji celu przedstawia rys. 8.

O wyborze rozwiązania preferowanego decyduje funkcja użyteczności, odzwierciedlająca koszt materiału struktury, proporcjonalny do jej masy i koszt wykonania struktury, proporcjonalny do liczby węzłów [1]



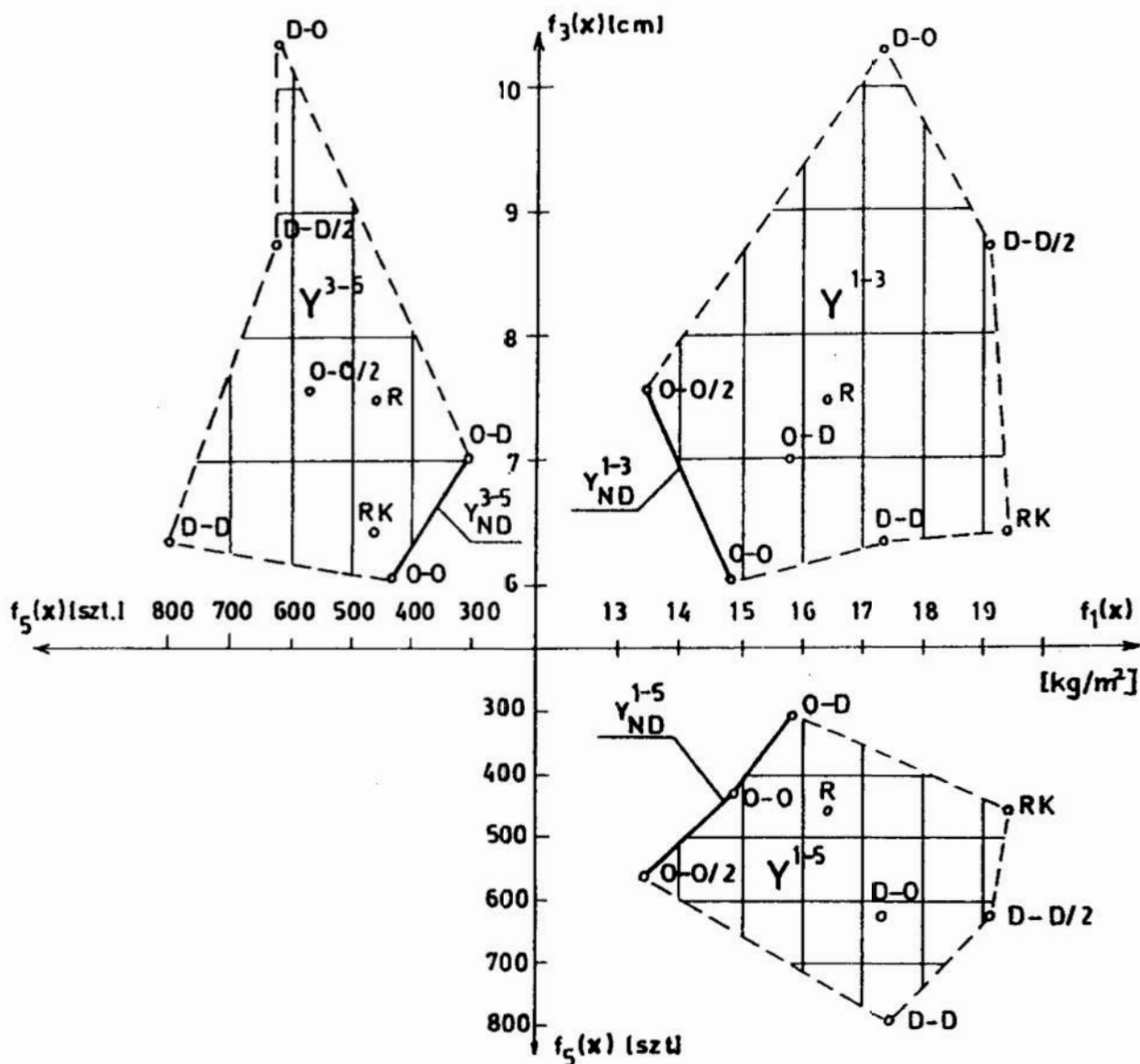
Rys. 7. Przykłady siatek analizowanych w zadaniu A4

- a) ortogonalna OO b) ortogonalna rozrzedzona OO2
 c) diagonalna rozrzedzona DD2 d) rusztowa RK e) diagonalno-ortogonalna DO

Fig. 7. The examples of some topological lattices under consideration in task A4

$$U_4(\mathbf{x}_{ND}) = C_{U1} \frac{f_1(\mathbf{x})}{f_{1P}(\mathbf{x})} + C_{U2} \frac{f_5(\mathbf{x})}{f_{5P}(\mathbf{x})},$$

gdzie: $C_{U1} = 0,6$ i $C_{U2} = 0,4$,
 $f_{1P}(\mathbf{x})$, $f_{5P}(\mathbf{x})$ - wartości porównawcze masy kratownicy i liczby węzłów w strukturze dla siatki OO: $f_{1P}(\mathbf{x}) = 14,84 \text{ kg/m}^2$, $f_{5P}(\mathbf{x}) = 431$.



Rys.8. Zbiór ocen niezdominowanych zadania A4
 Fig. 8. The set of efficient solutions of a task A4

Na podstawie przyjętej funkcji użyteczności jako rozwiązanie preferowane wskazano siatkę ortogonalno - diagonalną, charakteryzującą się najmniejszą liczbą węzłów i elementów. Schemat konstrukcji hali o parametrach określonych w zadaniu A pokazano na rys.1. Inna wartość modułu ($a = 3,37 \text{ m}$) niż w rozwiązaniu zadania A1 wynika z odmiennych możliwości kształtowania siatki O-D. To samo dotyczy rozmieszczenia podpór w wariancie P12.

3. Zadanie B rozproszonej optymalizacji hali

Równolegle rozwiązano zadanie B optymalizacji konstrukcji hali sportowej, różniące się rozwiązaniem technologicznym, analizowanymi zmiennymi decyzyjnymi i kryteriami oceny. W poszukiwaniu rozwiązania również wykorzystano metodę optymalizacji rozproszonej. Ze względu na odrębne założenia wykorzystywane w projektowaniu obiektu uzyskano dwa wyniki różniące się parametrami technicznymi. W zadaniu A jako preferowana wskazana została konstrukcja w postaci kratownicy ortogonalno-diagonalnej opartej według schematu P12, zaprojektowanej z 5-cio elementowego typoszeregu przekrojów pierścieniowych, o module $a = 3,37$ m i wysokości $h = 2,1$ m.

Schemat rozwiązania zadania B optymalizacji
The scheme of a task B solution

Tablica 2
Table 2

Zadanie B1		
zmiennie decyzyjne	k - krzywa, na której rozmieszczone są pola przekrojów typoszeregu prętów r - rozmieszczenie przekrojów typoszeregu prętów na krzywej	$k \in \underline{k} = \{k1; k2; k3; k4; k5\}$ $r \in \underline{r} = \{r1; r2; \dots; r31\}$
kryterium optymalizacji	min. masy struktury	$f_{1B}(x) = \sqrt{\sum_s (G_{ks}(x) - G_{ks \min})^2}$
metoda optymalizacji	Monte Carlo	
Zadanie B2		
zmiennie decyzyjne	a - moduł siatki s - typ siatki	$a \in \underline{a} = \{a_{DO}, a_{HT}, a_{HTT}, a_{OO}, a_{TT}, a_{TT2}, a_{TT2'}, a_{TTH}, a_{TTS}\}$ $s \in \underline{s} = \{DO, HT, HTT, OO, TT, TT2, TT2', TTH, TTS\}$
kryterium optymalizacji	min. koszt porównawczy kratownicy, pokrycia, ścian, konstrukcji wsporczej i fundamentów	$f_{2B}(x) = C_1 \frac{G_k(x)}{G_{kp}(x)} + C_2 \left(\frac{W(x)+P(x)}{W_p(x)+P_p(x)} \right) + C_3 \frac{V_s(x)}{V_{sp}(x)} + C_4 \frac{V_b(x)}{V_{bp}(x)} + C_5 \frac{V_r(x)}{V_{rp}(x)} + C_6 \frac{S_p(x)}{S_{rp}(x)} + C_7 \frac{S_s(x)}{S_{sp}(x)}$
metoda optymalizacji	pełny przegląd wariantów	
Zadanie B3		
zmiennie decyzyjne	R - gatunek stali w prętach warstwy górnej, dolnej i środkowej	$R \in \underline{R} = \{R35; R45\}$
kryteria optymalizacji	min. masy struktury min. maksymalnego przemieszczenia węzła kratownicy	$f_{3B}(x) = G_{pr} + G_w$ $f_{4B}(x) = \max_i \sqrt{u_i^2 + v_i^2 + w_i^2}$
rozwiązanie preferowane	funkcja użyteczności	$U_{1B}(x) = \sqrt{[f_3(x) - f_{3 \min}(x)]^2 + [f_4(x) - f_{4 \min}(x)]^2}$
metoda optymalizacji	pełny przegląd wariantów	
Zadanie B4		
zmiennie decyzyjne	r - rozmieszczenie przekrojów typoszeregu prętów na krzywej	$r \in \underline{r} = \{r1, r2, \dots, r8\}$
kryterium optymalizacji	min. masy struktury	$f_{1B}(x) = \sqrt{\sum_s (G_{ks}(x) - G_{ks \min})^2}$
metoda optymalizacji	sterowany przegląd wariantów	

preferowany wektor zmiennych decyzyjnych x_p	masa kratownicy [kg/m ²]	max. prze- mieszcze- nie węzła [cm]	liczba węzłów [szt]	liczba elementów [szt]	masa konstrukcji wsporczej [kg/m ²]	masa funda- mentów [kg/m ²]
ZADANIE A $x_{pA} = [a=3,37m, h=2,1m, t=5, p=P12, s=OD]^T$	stal 16,33	8,78	312	1014	stal 12,13	żelbet 20,0
ZADANIE B $x_{pB} = [k=k5, r=r25, s=OO, a=3,2m, R=\{R35, R45, R35\}]^T$	stal 18,01	9,95	431	1600	żelbet 263,2	żelbet 21,0

W zadaniu B jako optymalna wskazana została kratownica ortogonalno-ortogonalna (analizowano 8 typów siatek innych niż w zadaniu A - głównie trójkątnych T i heksagonalnych H) o module $a=3,2$ m, oparta na podłużnych krawędziach w warstwie górnej, zaprojektowana z pięcioelementowego typoszeregu przekrojów pierścieniowych leżącego na krzywej ograniczającej typoszereg metalurgiczny od dołu, ze stali R35 dla warstwy górnej i dolnej oraz R45 dla warstwy środkowej. Schemat rozwiązywania zadania B przedstawiono w tablicy 2

Podstawowe parametry techniczne obiektu zestawiono w tablicy 3. Otrzymane rozwiązania zadania A i B są rozwiązaniami zadowalającymi, tzn. poprawiającymi wyjściowe parametry analizowanej konstrukcji.

4. Uwagi końcowe

Metoda optymalizacji rozproszonej może znaleźć zastosowanie zarówno w projektowaniu konstrukcji, jak również w problemach sterowania i zarządzania - wszędzie tam, gdzie ze względu na złożoność analizowanych zadań niemożliwe jest wyznaczenie zadań globalnie polioptymalnych. Umiejętne współdziałanie specjalisty z danej dziedziny (konstruktora, architekta, technologa, ekonomisty) oraz inżyniera wiedzy koordynującego proces optymalizacji, może prowadzić do uzyskania nowych rozwiązań. Pomimo, że będą one "tylko" zadowalające, to w porównaniu z wyjściowymi przyniosą zapewne wymierne efekty. Jakość uzyskiwanych rozwiązań w optymalizacji rozproszonej zależy od możliwości obliczeniowych, dopuszczalnego czasu analizy oraz progów satysfakcji stawianych przed rozwiązaniem zadowalającym.

Literatura

1. Bródka J. (red.): Przekrycia strukturalne. Arkady, Warszawa 1985.
2. Büttner O., Stenker H.: Lekkie budownictwo metalowe. Arkady, Warszawa 1975.
3. Jendo S. Paczkowski W.M.: Multicriteria discrete optimization of large scale truss systems. *Structural Optimization*, 6, 1993, Springer-Verlag, pp.238-249.
4. Paczkowski W.M.: Dyskretna polioptymalizacja typoszeregu prętów kratownicy przestrzennej. *Szczecińskie Roczniki Naukowe, Ossolineum* 1991, t.IV, z. 1, s.83-104.
5. Paczkowski W.M.: Program dyskretnej optymalizacji OPTIM wykorzystujący metodę ortogonalno-diagonalną. *Zeszyty Naukowe Wydziału Mechanicznego, Nr 16, WSInż w Koszalinie* 1994, s. 188-201.

Streszczenie

Przedstawiono przykład dyskretnej, wielokryterialnej optymalizacji konstrukcji budowlanej. W analizie zastosowano metodę optymalizacji rozproszonej opartej na dekompozycji zadania. Optymalizacja rozproszona polega na połączeniu poprzez rozwiązania lokalne szeregu zadań optymalizacji i polioptymalizacji. Jako końcowy rezultat uzyskuje się rozwiązanie zadowalające, które nie koniecznie musi być globalnie optymalne lub globalnie preferowane.

DISPERSE OPTIMIZATION OF A SPORTS HALL CONSTRUCTION

Summary

The discrete multicriterial optimization of engineering construction is being presented. As a solution it has been chosen the disperse optimization method based on a problem decomposition. The disperse optimization comes from the local solutions of many optimization and multicriterion optimization tasks combined together. As a final result one comes to a satisfactory solution, which does not have to be a globally optimal or a globally preferable.

Praca finansowana przez Komitet Badań Naukowych - BW/RKH nr 11-0102/17-99

WYKORZYSTANIE WNIOSKOWANIA PRZY NIEPEŁNEJ INFORMACJI DO PROJEKTOWANIA UKŁADÓW CYFROWYCH

1. WSTĘP

Dynamicznie rozwijająca się w ostatnim czasie technologia produkcji złożonych układów cyfrowych nakłada na projektantów zadanie poszukiwania coraz to nowych metod umożliwiających automatyzację procesu projektowania tych układów. Działania prowadzone są dwutorowo tzn. poszukiwane są sformalizowane metody oparte o ścisły aparat matematyczno-logiczny oraz prowadzone są prace służące do praktycznej implementacji wypracowanych teoretycznie metod. Proces automatyzacji stosowany jest na wszystkich poziomach realizacji układów cyfrowych tzn. od momentu wypracowania koncepcji poprzez poziom architektury systemu, poziom struktury logicznej układu, schemat elektryczny aż do poziomu wyłożenia (ang. layout) na krzemie.

Stworzenie w pełni zautomatyzowanego narzędzia do projektowania układu od początku do końca pozostaje wciąż mało realne. Dlatego, na obecnym etapie, proponuje się systemy z interaktywnym udziałem człowieka. Jednym z narzędzi stosowanych dla potrzeb projektowania są systemy ekspertowe wykorzystujące niezalgorytmizowane, często heurystyczne metody. Modele takich systemów zostaną przedstawione poniżej.

2. IDEA PROPOWANEGO SYSTEMU EKSPERTOWEGO

Bardzo często przystępując do procesu projektowania złożonego układu cyfrowego brak gotowej, skończonej koncepcji, ponadto nie zawsze od razu wiadomo, które z rozwiązań byłoby najlepsze. Jednakże zawsze na wstępie istnieje pewien niezbędny, minimalny zestaw informacji wejściowej tzw. danych wejściowych. Pozostałe informacje nie muszą być wiadome - są to (dane dodatkowe). Wygenerowanie pierwszego, wstępnego rozwiązania odbywa się przy niepełnej, niekompletnej informacji.

Wstępny projekt może być w takim systemie przez użytkownika zaakceptowany lub skierowany do poprawy. Pierwszą koncepcję, nie koniecznie słuszną można zweryfikować dodając dodatkowe dane. Postępując w ten sposób iteracyjnie (doprecyzowując informacje) można uzyskać zadowalający wynik (rys. 1). System taki musi być wyposażony w odpowiednie narzędzia służące do wnioskowania w warunkach niepełnej informacji. Możliwości te daje sformalizowana logika niemonotoniczna.

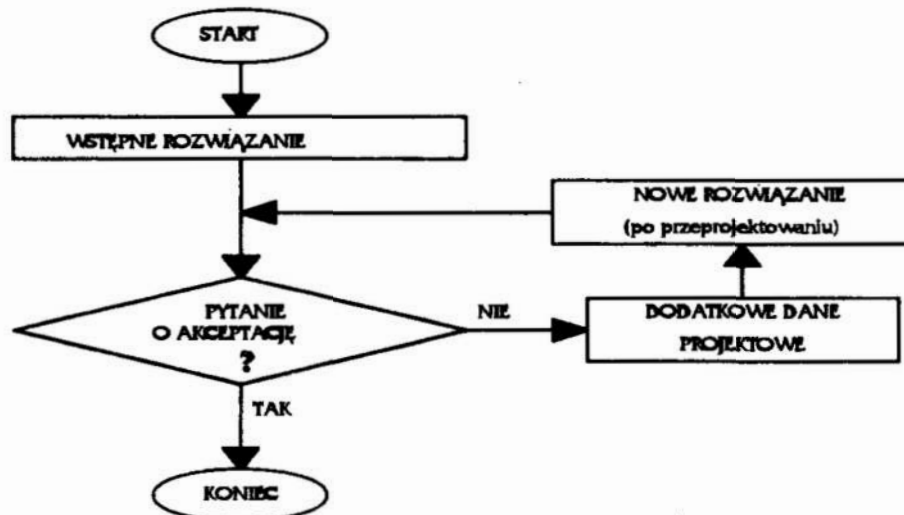
3. WPROWADZENIE DO PROBLEMU WNIOSKOWANIA NIEMONOTONICZNEGO

Wnioskowanie niemonotoniczne umożliwia wyciąganie hipotez, które nie zawsze (niekoniecznie) są słuszne, tzn. w wyniku uzyskania dodatkowych informacji mogą zostać obalone. Wszystkie dotychczasowe logiki klasyczne oparte były o zasadę monotoniczności tzn. zbiór przekonań (twierdzeń) monotonicznie wzrastał wraz ze zwiększaniem się zbioru przesłanek (aksjomatów). Obowiązujący był schemat:

^{*)} Instytut Elektroniki, Politechnika Śląska, 44-100 Gliwice, ul. Akademicka 16

jeżeli spełnione są formuły X_1, X_2, \dots, X_n to wnioskuje formułę X_{n+1} ".

W systemach wykorzystujących metody sztucznej inteligencji, a więc modelujących inteligentne zachowanie człowieka, istnieją nieklasyczne struktury dedukcyjne opierające się także na niewiedzy (ignorancji). Charakterystyczną cechą takich struktur jest ich niemonotoniczność tzn. możliwe jest wyprowadzenie takiego twierdzenia (przekonania), które można unieważnić po dodaniu nowego aksjomatu (faktu pewnego).



Rys. 1. Paradygmat projektowy systemu pracującego przy niepełnej informacji.

Fig. 1. Design paradigm in the absence of complete information.

DEFINICJA 1

Logika niemonotoniczna to taki system logiczny, którego struktura dedukcyjna:

- (1) pozwala na wyprowadzanie nowych hipotez w oparciu zarówno o wiedzę (aksjomaty), jak i w oparciu o niewiedzę (ignorancję);
- (2) umożliwia unieważnianie dotychczas wyprowadzonych hipotez, jeśli dodane zostały nowe aksjomaty pozostające w sprzeczności z tymi hipotezami.

Można łatwo pokazać, że wnioskowanie charakterystyczne dla zachowań ludzkich zwane wnioskowaniem naturalnym bądź też zdrowo-rozśądkowym (ang. common-sense reasoning) ze swej natury jest niemonotoniczne. Istnieje szereg przykładów modelujących zachowanie ludzkie oparte na takim właśnie systemie dedukcyjnym.

PRZYKŁAD 1

Wyobraźmy sobie człowieka wybierającego się w podróż samochodem. Jeżeli myśli on racjonalnie to, po pierwsze uda się na miejsce gdzie poprzednio zostawił pojazd, czyli np. do garażu, którego nie widać z okien domu. Jest oczywiste, że postępowanie takie nie jest poparte żadną wiedzą, pewnością, że tam właśnie znajduje się samochód, a wynika ono jedynie z pewnego przyzwyczajenia. Jednakże, skądinąd, takie postępowanie człowieka jest uzasadnione, ponieważ gdyby postępował on tylko opierając się na informacjach niewątpliwych, sprawdzonych, to nie byłby w stanie podjąć jakiegokolwiek działania. Istnieje bowiem zbyt wiele możliwości np. samochód mógł zostać skradziony, wyprowadzony z garażu przez kogoś z rodziny itp.. Podjęcie działania jest wynikiem wyprowadzenia wniosków zwanych przekonaniem (ang. belief), które nie są całkowicie niewątpliwe i mogą ulec unieważnieniu.

PRZYKŁAD 2

Widząc na ulicy biegnące samotnie dziecko zastanawiamy się gdzie są rodzice. Pierwszą rzeczą jaką pomyśleliśmy o tym dziecku było to, że ma rodziców. Wniosek wyprowadzony został na podstawie obserwacji, że typowe dzieci mają rodziców.

Powyższe przykłady można zapisać w następującej formie :

- (*) jeżeli nie wiesz, że jest inaczej to załóż, że samochód znajduje się w garażu;
- (**) jeżeli dana osoba jest dzieckiem i nie ma powodów by sądzić inaczej, to załóż, że ma ona rodziców.

Przykłady te pozwalają sformułować podstawowe problemy jakie pojawiają się przy konstruowaniu systemów niemonotonicznych:

- a) problem formalizacji tzn. zaproponowania sposobu reprezentacji wiedzy o opisywanym świecie oraz sposobu wyprowadzania nowych przekonań o tym świecie;
- b) problem weryfikacji przekonań lub rekonstrukcji świata (ang. revision of beliefs) polegający na sprawdzaniu słuszności (niesprzeczności) dotychczas wyprowadzonych wniosków w wyniku dopływu nowych informacji (aksjomatów).

PRZYKŁAD 3

Przyjmijmy następujące reguły projektowe dla układu elektronicznego:

- (1) jeżeli układ sterujący realizowany jest za pomocą obwodów logicznych tzw. mikro-programowanie swobodne (ang. random logic control part) to, o ile nie wiadomo, że jest inaczej, działanie części sterującej wprowadza opóźnienia w propagacji sygnału np. mniejsze niż 5ns,
- (2) jeżeli część sterująca wprowadza opóźnienia mniejsze niż 5ns to, o ile nie wiadomo, że jest inaczej, załóż, że zegar taktujący układ może pracować z częstotliwością 50 MHz.
- (3) jeżeli zegar pracuje z częstotliwością 50 MHz to, o ile nie wiadomo, że jest inaczej, przyjmij, że układ może wykonać np. 10 tys. operacji mnożenia na sekundę.

Zakładając, że baza wiedzy zawiera tylko te reguły możemy wywnioskować:

- (4) część sterująca wprowadza opóźnienia w propagacji sygnału mniejsze niż 5 ns.
- (5) zegar pracuje z częstotliwością 50 MHz
- (6) układ może wykonać 10 tys. operacji mnożenia na sekundę.

Jeżeli jednak okazałoby się, że z pewnych przyczyn prawdziwe jest stwierdzenie:

- (4') układ sterujący wprowadza opóźnienia sygnału rzędu 7.5 ns,

to wystąpi sprzeczność z (4), ale nie tylko. Aby model był niesprzeczny należy usunąć nie tylko (4), ale również (5) i (6), które pośrednio zależą od (4).

4. FORMALNE TEORIE NIEMONOTONICZNE

Najbardziej rozpowszechnione są dwa mechanizmy modelujące logikę niemonotoniczną: *logika domniemań* (ang. default logic) oraz *otaczanie* (ang. circumscription).

LOGIKA DOMNIEMAŃ (DEFAULT LOGIC)

W logice domniemań możliwe jest wyprowadzanie prawdopodobnych, choć niekoniecznie słusznych, hipotez przy braku informacji blokujących ich prawdziwość, a baza wiedzy podzielona jest na wiedzę, czyli aksjomaty (fakty niepodważalne) i niewiedzę, czyli domniemania, które są modelami wnioskowania niemonotonicznego.

DEFINICJA 2

Założmy, że wykorzystujemy język LA logiki klasycznej pierwszego rzędu, składający się ze zbioru poprawnie sformułowanych formuł wff (ang. well formed formulae) i operujących na alfabecie A złożonym z symboli zmiennych, funkcji, predykatów, stałych logicznych i znaków interpunkcji. Defaultem nazywamy regułę wnioskowania :

$$\frac{a(x) : M \ b1(x) \dots \ bn(x)}{w(x)}$$

gdzie: **a** - warunek wstępny (prerequisite), **b1...bn** uzasadnienie (ang. justification), **w** wniosek (ang. consequent).

Default nazywamy normalnym gdy $w(x) = b1(x) \dots bn(x)$ bądź pseudonormalnym gdy $w(x)$ zawiera się w $b1(x) \dots bn(x)$. Interpretacją defaultu jest reguła wnioskowania:

"dla dowolnych obiektów x spośród $x1 \dots xn$ jeżeli $a(x)$ jest hipotezą i jeżeli fakty $b1(x) \dots bn(x)$ są niesprzeczne ze zbiorem hipotez to regułę $w(x)$ należy uznać również jako hipotezę prawdziwą (należy do zbioru przekonań)".

MECHANIZM OTACZANIA (CIRCUMSCRIPTION)

Drugim, rozbudowanym teoretycznie mechanizmem niemonotonicznym jest otaczanie. Idea otaczania opiera się na minimalizacji cech pewnych obiektów poprzez zawężenie zbioru elementów danej klasy jedynie do tych elementów, o których na pewno wiadomo, że posiadają daną cechę. W pracy W.Łukaszewicza [2], a także w [3] mowa jest o predykanie "nienormalności" (ang. abnormal), który decyduje o typowości bądź o nietypowości obiektu danej klasy.

Rozważmy np. bazę wiedzy, w której interesuje nas własność "być zielonym". Wszystkie obiekty można wówczas podzielić na 3 różne grupy :

- 1) obiekty, o których wiemy, że są zielone
- 2) obiekty, o których wiemy, że nie są zielone
- 3) obiekty, o których kolorze nic nie wiemy.

Otoczenie własności "być zielonym" będzie polegało na przyjęciu założenia, że każdy obiekt klasy (3) nie jest zielony.

DEFINICJA 3

Niech $A(P)$ będzie formułą logiki drugiego rzędu, przy czym P jest oznaczeniem kilku symboli wolnych predykatów występujących w $A(P)$. Niech ponadto $E(P,x)$ oznaczać będzie formułę spośród wff, a symbole P i x (oznaczające zmienne indywidualne) występują jako zmienne wolne. Otaczaniem $E(P,x)$ względem $A(P)$ jest formuła $A'(P)$:

$$A(P) \wedge \forall P'. [A(P') \wedge [\forall x, E(P',x) \supset E(P,x)] \supset [\forall x. E(P,x) \equiv E(P',x)]]$$

Otoczanie jest więc rodzajem minimalizacji, a symbole predykatów występujących w $A(P)$, a nie występujące w P należy traktować jak parametry tej minimalizacji.

INNE MECHANIZMY NIEMONOTONICZNE

Oprócz wymienionych podstawowych mechanizmów niemonotonicznych na uwagę zasługuje hipoteza domkniętości świata (closed world assumption) zwana "negacją jako porażką" (negation as a failure) i logika niemonotoniczna (nonmonotonic reasoning).

Są one jednak mocno niedopracowane i stosowane w ograniczonym zakresie.

5. WNIOSKOWANIE NIEMONOTONICZNE W PROJEKTOWANIU. PRZYKŁADY IMPLEMENTACJI W PROLOGU.

Poniżej przedstawione zostały przykłady systemów ekspertowych służących do projektowania ścieżki sterującej cyfrowego układu elektronicznego. Zaprezentowane implementacje w Prologu wykorzystują wnioskowanie przez domniemanie (system *DLA*) oraz otaczanie (system *CA*). W obu poniżej prezentowanych systemach w odmienny sposób wykorzystano *Prologowską negację jako porażkę* [4], a co za tym idzie w inaczej rozwiązany został problem negacji logicznej. Ponadto w obu podejściach inaczej dokonuje się wyboru jednego spośród kilku alternatywnych projektów.

PROJEKTOWANIE CZĘŚCI STERUJĄCEJ UKŁADU CYFROWEGO.

Jednym z ważnych problemów występujących podczas realizacji projektu układu cyfrowego jest właściwy wybór sposobu jego sterowania. Zaprezentowane poniżej systemy ekspertowe są przykładem zautomatyzowania procesu projektowania na tym etapie - tzn. wyboru rodzaju architektury sterowania. W systemie *DLA* skupiono się na wyborze struktury układu sterowania. Natomiast w systemie *CA* potraktowano zagadnienie nieco szerzej łącznie z wyborem alokatorów hardware'u oraz bibliotek komórek standardowych. W systemach danymi wejściowymi były typowe informacje konieczne do zaprojektowania sterowania (ilość sygnałów sterujących, ilość stanów, warunków itd.) natomiast dodatkowe informacje związane ze szczegółową realizacją układową (np. synchroniczność warunków) stanowiły uzupełniające dane. Do implementacji wybrany został wspomniany już wyżej PROLOG - język sztucznej inteligencji.

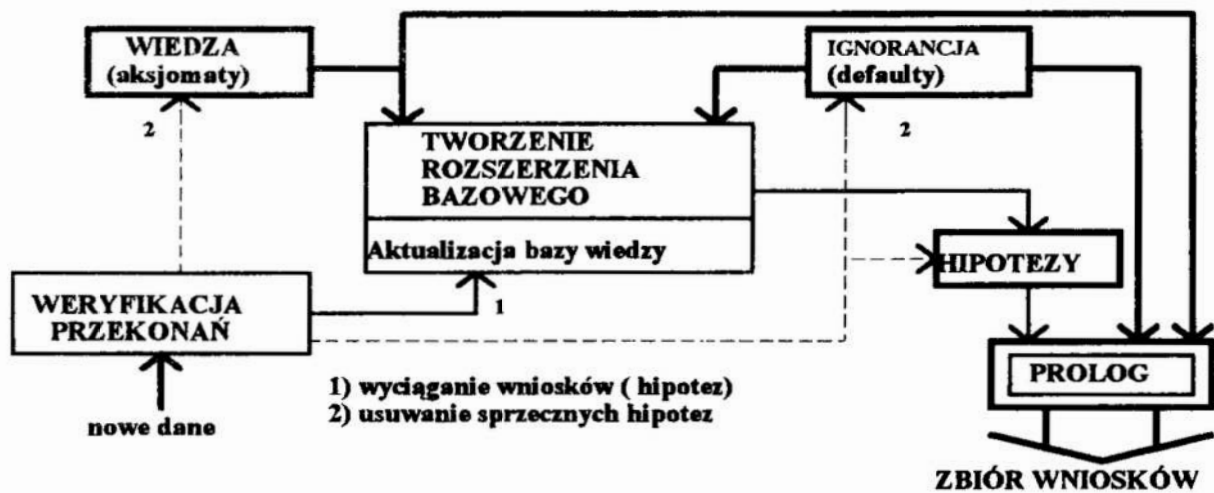
SYSTEM OPARTY NA LOGICE DOMNIEMIAŃ - DLA.

W systemie ekspertowym opartym na logice domniemań - *DLA* (ang. Default Logic Assistant) [5] (rys. 2) baza wiedzy składa się z dwojakiemu rodzaju reguł: typowych reguł produkcji i reguł defaultowych implementujących "niewiedzę". Wnioskowanie zapewnia mechanizm niemonotoniczny oraz rezolucja liniowa (PROLOG) (przykłady reguł pokazano na rys. 3). Na podstawie reguł defaultowych uzgadniane są najpierw informacje "wątpliwe", a następnie generowany jest ostateczny wynik (projekt). O wyborze danego rozwiązania decyduje kolejność reguł domniemań.

SYSTEM OPARTY O PROLOGOWSKĄ NEGACJĘ I OTACZANIE - CA.

Drugi zaimplementowany mechanizm - *CA* (ang. Circumscription Alike) [7] opiera się na otaczaniu oraz wykorzystuje niedoskonałość samego Prologu polegającą na tym, iż niemożliwe jest w nim zaimplementowanie negacji logicznej. Negacja jest tutaj traktowana "jako porażka" tzn. jeżeli nie można udowodnić danego faktu to wnioskuje się fakt przeciwny. W systemie tym proces projektowania podzielony został na etapy - związane z rozwiązaniem określonych podproblemów projektowych.

Na każdym etapie poszukiwane jest rozwiązanie, które może być "silne" lub "słabe" (zawsze istnieje conajmniej jedno). Konstrukcja systemu wykorzystuje dwa rodzaje dowodów w PROLOGU - dowód silny - nie podlegający dyskusji oraz dowód słaby - taki, który został dowiedziony przy użyciu co najmniej jednej prologowskiej negacji. W przypadku rozwiązań słabych możliwe jest istnienie kilku różnych wariantów, a wyboru najlepszego dokonuje mechanizm priorytetowy na podstawie tzw. "listy priorytetów", która wskazuje na hierarchię wymagań stawianych projektowi (rys. 4).

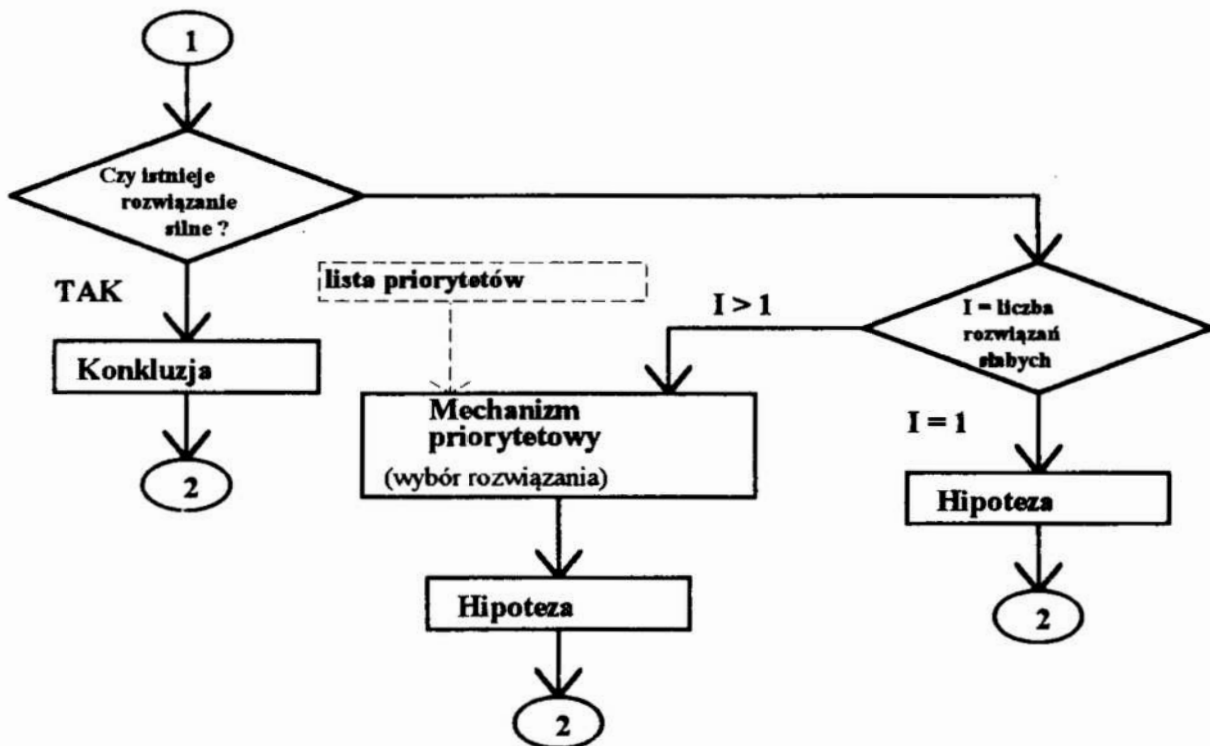


Rys. 2. Struktura systemu DLA.
 Fig. 2. The structure of the DLA.

```

infer(Hypothesis):- call(default(Prqt,Jst,Hypothesis), call(Prqt), \+ Jst, \+ Hypothesis,
                    assertz(Hypothesis), assertz(hypothesis(Hypothesis))).
infer(_).
check_contradiction(Fact):- clausex(hypothesis(Fact)_,0,Position), default(Prqt,Jst,Fact), call(Jst),
                                retractallfrom(hypothesis(Fact)).
check_contradiction(Fact):- write('koniec rewizji hipotez').
  
```

Rys. 3. Najważniejsze predykaty mechanizmu wnioskującego systemu DLA.
 Fig. 3. The main predicates of the inference engine of the DLA system.



Rys. 6. Proces podejmowania decyzji na danym etapie projektowym w systemie CA.
 Fig. 6. The process of making decision on the given design level.

```

not1(Fact):-      neg(Fact,Negation), call(Negation),!.
not1(Fact):-      not(Fact), assertz(weak(Fact)).
strenght(strong):- not(weak(_)),!.
strenght(weak).
ca(Aspect,Solution):- ver(Solution,Different), call_strenght(assumpt(Aspect,Solution),STR1),
                      call_strenght(assumpt(Aspect,Different),STR2),
                      decide(Aspect,Solution,Different, STR1,STR2),!.
ca(Aspect,Solution):- call_strenght(assumpt(Aspect,Solution),Strenght), delete_old(Aspect),
                      assertz(conclusion(Aspect,Solution), check(Strenght,Aspect,Solution),!.

```

Rys. 7. Mechanizm wnioskujący systemu CA ze "zmodyfikowaną negacją".
 Fig.7. The inference engine of the CA system with "modified negation".

reguły projektowe:

- 1.) *assumpt(cell_library, fast_library):- circuit(small), not1(abnormal(speed)).*
- 2.) *assumpt(cell_library, cheap_library):- circuit(very_large), not1(abnormal(speed)), not1(abnormal(design_cost)).*
- 4.) *assumpt(design_style_of_control, random_logic):- circuit(small), abnormal(regularity)).*
- 5.) *assumpt(design_style_of_control, pla_controler):- not1(abnormal(area)), conclusion(cell_library, cheap_library), not1(abnormal(design_cost)).*
- 6.) *assumpt(hardware_allocator, exhaustive_pipelined_allocator):- circuit(very_large), not1(abnormal(speed)).*

z reguł 4 i 6 (przy braku dodatkowych danych) system znajduje dwie hipotezy:

- a) *hypothesis(design_style_of_control, pla_controler).* oraz
- b) *hypothesis(design_style_of_control, microprogrammed_control).*

jeżeli lista wymagań wygląda następująco:

priority_list([area, design_cost, speed, regularity, power]).

to system wybierze rozwiązanie a) jeśli by jednak dodać informację mówiącą o tym, że powierzchnia (area nie jest istotna), bądź zmienić listę priorytetów to okazałoby się, że rozwiązanie b) jest silniejsze.

Rys. 6. Przykład niemonotoniczności systemu CA.

Fig 6. An example of the nonmonotonicity of the CA system.

6. PODSUMOWANIE I UWAGI KOŃCOWE.

Opisane powyżej systemy ekspertowe oparte na niemonotonicznym paradygmacie projektowym (rys. 1) udawniają celowość wykorzystania metod sztucznej inteligencji do automatyzowania procesu projektowania. Warunkiem by zastosowanie takiego systemu miało sens jest znalezienie zagadnienia, w którym występuje tzw. "swoboda projektowa" tzn. istnieje wielość rozwiązań, a wskazanie od razu jednego, najlepszego rozwiązania jest bardzo trudne lub wręcz niemożliwe.

W obu podejściach: zarówno w DLA jak i w CA wystąpiło kilka problemów natury implementacyjno - technicznej: problem wyboru jednego z wielu możliwych rozwiązań (kolejność defaultów w bazie wiedzy lub mechanizm priorytetowy z listą wymagań) oraz problem zapewnienia logicznej negacji (w DLA zaimplementowano sztuczną negację uzasadnienia defaultów, a w CA traktowano negację w sensie "jako porażkę" i rozpatrywano "siłę dowodu").

Zaprezentowany tutaj sposób podejścia do automatycznego projektowania wydaje się być obiecujący i może być zastosowany szerzej m.in. jak pokazują to [8] i [9]. Przy nieco innym potraktowaniu informacji dodatkowych wnioskowanie przy niepełnej informacji można stosować do modelowania układów wysokiej skali integracji w języku VHDL pod kątem efektywnej ich symulacji.

LITERATURA

1. Reiter R.: *A logic for default reasoning*, *Artificial Intelligence* 13 (1980).
2. W.Łukaszewicz: *Wykorzystanie logik w bazach wiedzy*, UW, Warszawa 1986.
3. McCarthy J.: *Application of Circumscription to Formalizing Common-Sense Knowledge*, *Artificial Intelligence* 28 (1986).
4. T.Grabowiecki, A.Pawlak: *PROLOG as a Formalism for VLSI Design Specification*, *Microprocessing and Microprogramming* 25, North-Holland, 1989.
5. T.Grabowiecki, A.Pułka: *Non-monotonic Reasoning in Digital Circuit Design*, In: G.Odawara (Ed.): *CAD Systems Using AI Techniques*, North-Holland, 1989.
6. J.Granacki, D.Knapp, A.Parker: *The ADAM Advanced Design AutoMation System: Overview, Planner and Natural Language Interface*, In *Proceedings of the 22th DAC '85*.
7. T.Grabowiecki, A.Pułka: *Automated Design with Prolog in the Absence of Complete Information*, *Materiały XV-tej Krajowej Konferencji Teoria Obwodów i Układy Elektroniczne*, Szczyrk 1992.
8. A.Pułka: *Generation of VHDL Models Aided by Artificial Intelligence*, *Materiały XVI-tej Krajowej Konferencji Teoria Obwodów i Układy Elektroniczne*, Kołobrzeg 1993.
9. A.Pułka: *VHDL Models Generation with Prolog in the Absence of Complete Information*, *Proceedings of the 1-st WORKSHOP ON Design Methodologies for Microelectronics and Signal Processing*, Gliwice - Cracow, Poland, 20-23 October 1993, pp. 301-305.

Streszczenie

Praca niniejsza dotyczy analizy możliwości zastosowania logik niemonotonicznych do automatyzacji procesu projektowania cyfrowych układów sterowania. Omówiono dwa podstawowe mechanizmy logikę defaultów oraz otaczanie. Następnie opisane zostały zaimplementowane mechanizmy logiki domniemań oraz otaczania w różnych systemach ekspertowych, służących do wspomaganie projektowania układów cyfrowych. Na zakończenie podsumowano uzyskane rezultaty oraz przedstawiono możliwości zastosowania logik niemonotonicznych w modelowaniu układów cyfrowych.

NONMONOTONIC REASONING USED IN THE DIGITAL CIRCUITS DESIGNING

Summary

This approach analyzes possibilities of employing non-monotonic logic in the process of digital control systems design. Two basic mechanisms: the default logic and the circumscription are described. Then the implementations of both mechanisms in the expert systems for digital circuits design are presented. Some conclusions concerning the results of the presented work are drawn and the possibilities of employing these approaches in digital circuits modelling are pointed.

KOMPUTEROWE WSPOMAGANIE PROJEKTOWANIA - STRUKTURA DANYCH

"Niezwykle ważną ... była publikacja Hoare'a *Notes on data structuring*. Wynika z niej, że jakiegokolwiek decyzje dotyczące struktury danych mogą być podjęte jedynie na podstawie znajomości algorytmów zastosowanych do danych i, vice versa, struktura i wybór algorytmów zależą ściśle od struktury danych. Krótko mówiąc, problemy tworzenia programów i struktur danych są ze sobą ściśle powiązane." [2]

Dla komputera pamięć stanowi jednolitą masę bitów bez wyraźnej struktury. Języki programowania wysokiego poziomu przejmując zarządzanie pamięcią nadają jej pośrednio czytelną i łatwo dostępną postać. Współczesne języki programowania takie jak PASCAL [1] i C, oraz ich dialekty, udostępniają użytkownikowi wysoko zorganizowane struktury danych. Przy ich użyciu postulat N.Wirth'a cytowany na początku pracy należy traktować jako punkt wyjścia przy budowie modelu komputerowego układu.

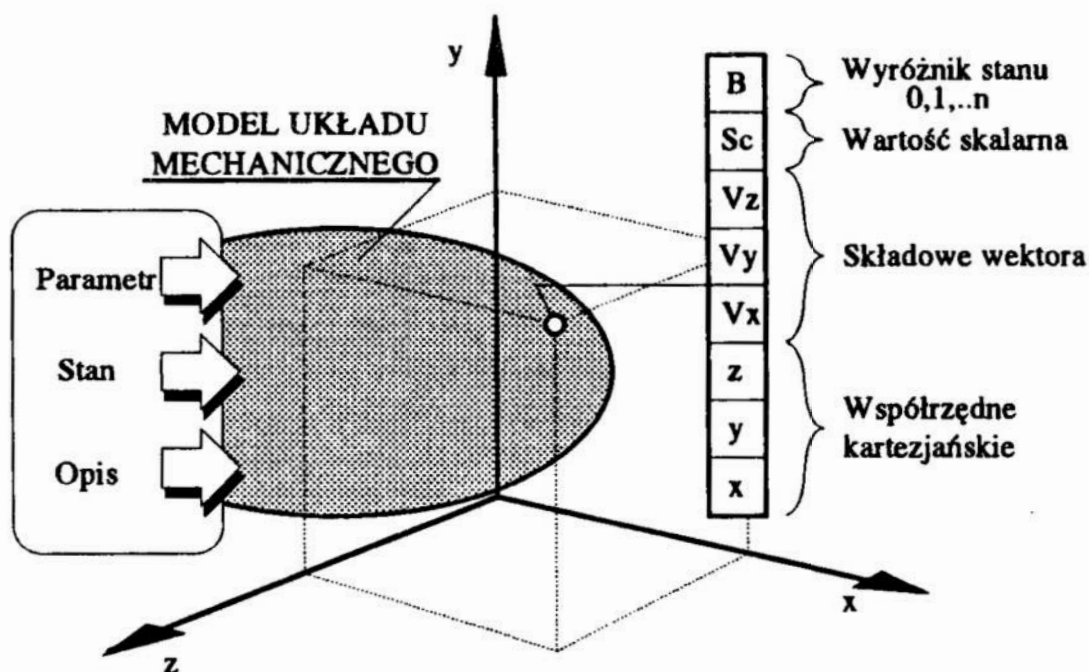
1. Dane wejściowe i wyjściowe w modelu układu mechanicznego

Modele układów mechanicznych są z natury rzeczy modelami ciągłymi. Przy określaniu postaci konstrukcyjnej model rozpatrywanego układu jest ciągły w przestrzeni. Zdyskretyzowanie modelu powodowane z jednej strony łatwością opisu i analizy, a z drugiej strony wymaganiami maszyny liczącej oznacza przypisanie punktowi w przestrzeni kartezjańskiej $p(x,y,z)$ pewnych zmiennych opisowych. Zmienne te reprezentować mogą różne klasy danych. Na rys. 1 przedstawiono model zmiennych opisowych dla dyskretnego modelu układu mechanicznego.

Dla klasy zagadnień występujących we wspomaganym komputerowo określaniu postaci konstrukcyjnej zmienne opisowe elementów modelu związane z punktem w trójwymiarowej przestrzeni kartezjańskiej (x, y, z) są danymi reprezentującymi wartości wektorowe (V_x, V_y, V_z) i/lub skalarnie (S_c) i/lub wyróżniki stanu (B) z zakresu $\{0,1,\dots,n\}$. Wyróżnik stanu jest przełącznikiem wielostanowej zmiennej opisowej. Wartość wyróżnika należącego do zmiennych typu porządkowego sama w sobie nie jest wartością zmiennej opisowej. Ogólnie można przyjąć, że jest indeksem wywołania określonej wartości zmiennej. W szczególnym przypadku zmiennej dwustanowej $\{0,1\}$ wyróżnik stanu może być traktowany jako zmienna logiczna $\{\text{true}, \text{false}\}$. W dalszej części referatu opisaną wyżej strukturę danych nazwać się będzie punktem zintegrowanym.

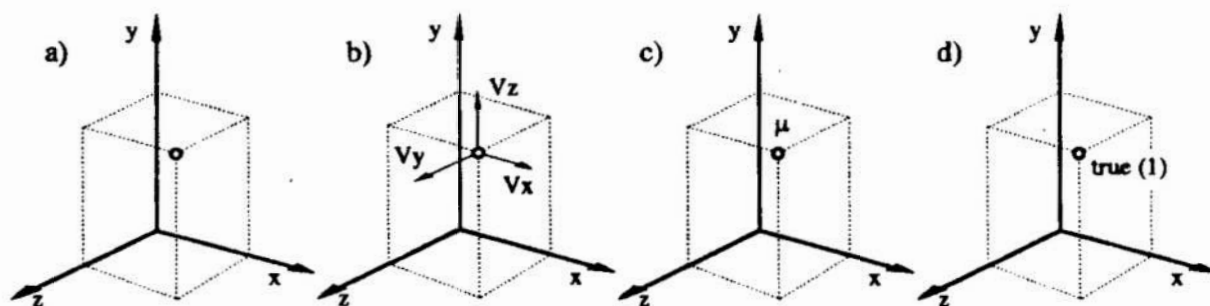
* Politechnika Gdańska

Wydział Mechaniczny Katedra Konstrukcji i Eksploatacji Maszyn
80-352 Gdańsk ul.Narutowicza 11/12



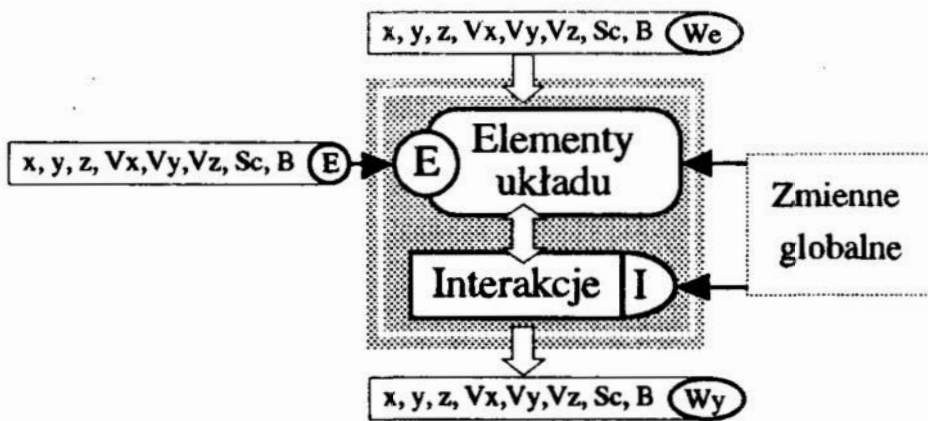
Rys. 1. Opis przestrzeni danych CAD
 Fig. 1. Description of data space in CAD

Na rys. 2 przedstawiono przykłady zmiennych opisowych związanych z punktem przestrzeni kartezjańskiej. Kolejno zilustrowano punkt bez innych zmiennych opisowych (rys. 2a), wektor opisany przez trzy składowe V_x , V_y , V_z (rys. 2b), wartość skalarną, np. współczynnik tarcia μ (rys. 2c) oraz logiczną zmienną *true* (1) jako wyróżnik stanu (rys. 2d). W szczególnych przypadkach punkt może być opisany wszystkimi zmiennymi.



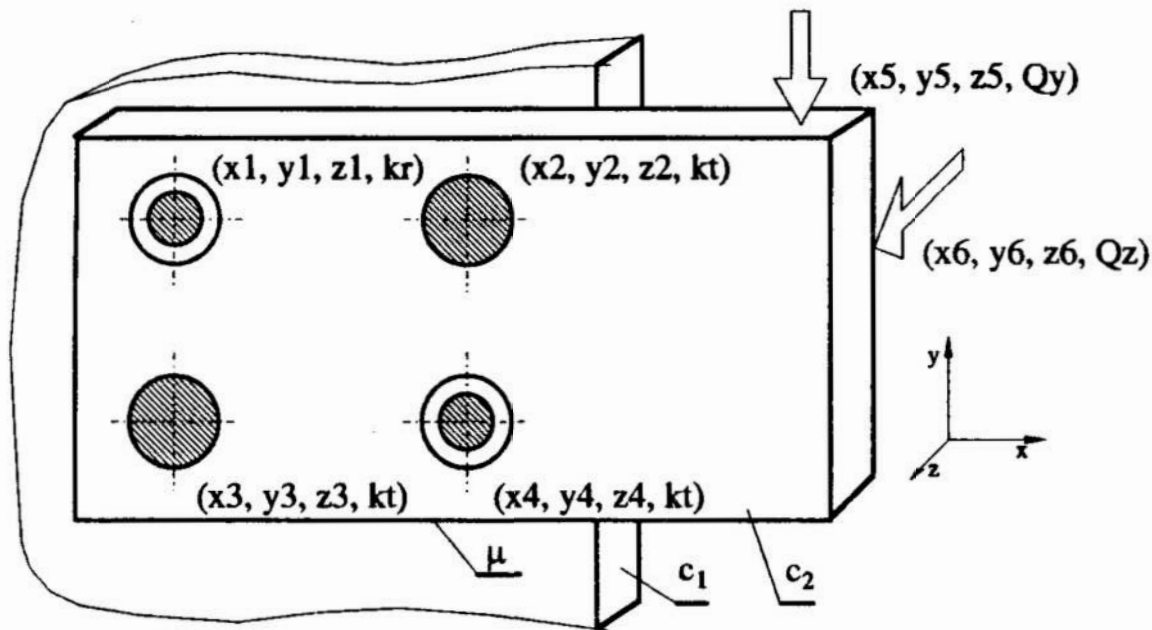
Rys. 2. Przykłady opisu punktu zintegrowanego
 Fig. 2. Examples of the integrated point

Niektóre zmienne opisowe modelu posiadają charakter globalny. Są to zmienne wchodzące do reguł opisujących interakcje pomiędzy elementami układu. Charakter zmiennych globalnych przypisać można również zmiennym, których wartość dla wszystkich elementów układu jest taka sama oraz zmiennych opisujących interakcję układu z otoczeniem. Z uwagi na możliwy typ globalnych zmiennych opisowych (w rozumieniu systemu komputerowego) wyróżnić można parametry (zmienne typu rzeczywistego), stan (zmienne typu porządkowego) oraz opis (zmienne typu łańcuchowego). Na rys. 3 przedstawiono model reakcji wejścia-wyjścia oraz zmiennych opisowych związanych z modelem struktury układu mechanicznego.



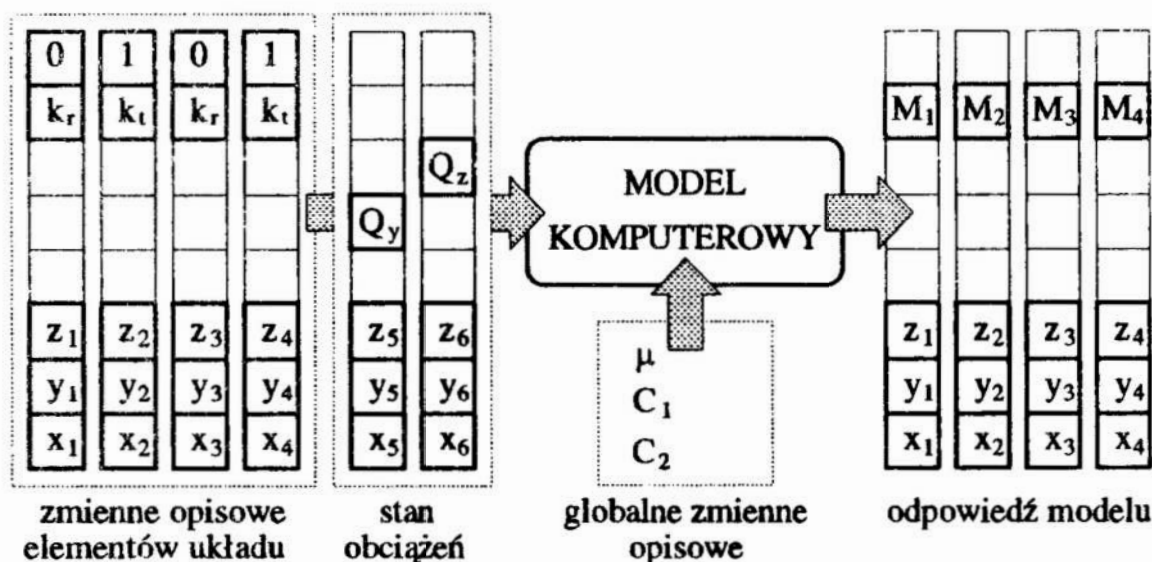
Rys. 3. Model reakcji wejścia-wyjścia oraz zmiennych opisowych układu rzeczywistego
 Fig. 3. Model of the input-output reactions and descriptive variables of a real system

Na rys. 4 przedstawiono przykład złącza śrubowego składającego się z dwóch śrub luźnych i dwóch śrub pasowanych. Zmienne opisowe złącza stanowią stan obciążenia, własności materiałowe śrub, współrzędne położenia śrub oraz zmienne globalne (współczynnik tarcia μ i sztywności łączonych elementów C_1, C_2).



Rys. 4. Schemat złącza śrubowego
 Fig. 4. Schematic representation of bolted joint

Złącze może być modelowane przy różnych reakcjach wejścia-wyjścia i różnych interakcjach elementów układu i układu z otoczeniem. Nie wnikając w szczegółowe relacje pomiędzy elementami układu, które są przedmiotem analizy na etapie przyjmowania właściwego układu eksperymentu, uproszczeń modelu podstawowego i dostosowania do wymagań branżowych oraz standaryzacji, struktura reakcji wejścia-wyjścia modelu komputerowego ma określoną postać. Na rys. 5 przedstawiono przykładową reakcję wejścia-wyjścia dla modelu złącza śrubowego, w którym należy określić wymiary śrub luźnych i pasowanych zgodne z PN na podstawie stanu obciążenia oraz położenia śrub i ich własności wytrzymałościowych.



Rys. 5. Reakcja wejścia-wyjścia złącza śrubowego

Fig. 5. Diagram of a model structure of input-output reaction for bolted joint

Proponowany punkt zintegrowany jako dana w komputerowym wspomaganie projektowania układów mechanicznych stanowi 8-elementowy wektor. W pierwszym przybliżeniu może on być traktowany jako znana struktura jednowymiarowej tablicy o 8-miu elementach typu rzeczywistego. Niezgodność typu wyróżnika stanu (B) opisanego do tej pory jako zmienna typu porządkowego powinna być wyeliminowana poprzez konwersję na typ rzeczywisty. W modelu komputerowym struktura danych jest elementem dziedziny algorytmicznej, dla której pozostaje otwarta definicja funkcji częściowych i relacji operujących na niepustym zbiorze danych typu "punkt zintegrowany".

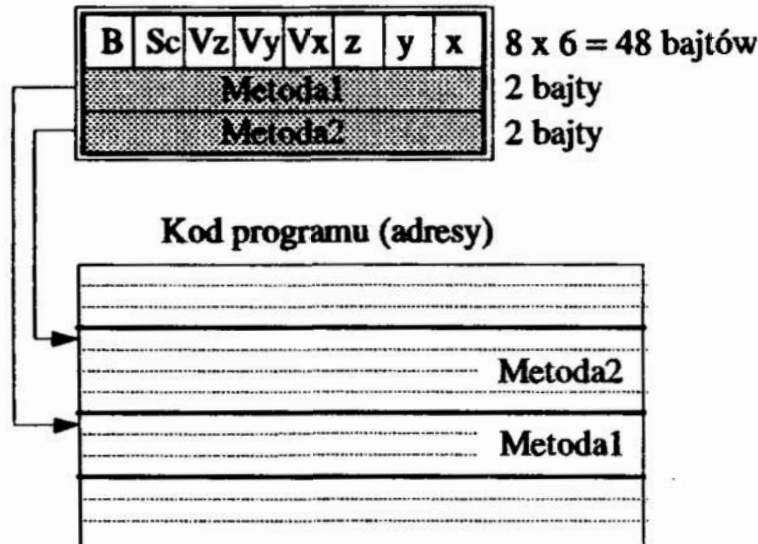
Programowanie zorientowane obiektowo (OOP - Object Oriented Programming) pozwala na programowe zrealizowanie dziedziny algorytmicznej dostosowanej do klasy rozwiązywanego problemu bez ingerencji w środowisko komputerowe na niskim poziomie translatorów.

2. Programowanie zorientowane obiektowo w Turbo PASCAL

Historyczny rozwój języków programowania prowadzi przez programowanie liniowe, strukturalne do programowania obiektowego. Współczesne języki programowania wysokiego poziomu takie jak PASCAL [1] oraz C wprowadzają typ zmiennej gdzie dane powiązane są z kodem tworząc typ obiektowy. Zmienia to zasadniczo sposób podejścia do programowania. W dalszej części referatu autor posłużył się językiem Turbo PASCAL firmy Borland Inc. v 6.0 [3]. Przyjęte rozwiązania opierają się na ogólnych zasadach programowania obiektowego co umożliwia ich łatwe przeniesienie na inne wersje PASCALA i inne języki programowania operujące OOP.

Na rys. 6 przedstawiono sposób reprezentacji w pamięci komputera zmiennej typu obiektowego zawierającą punkt zintegrowany. Metoda czyli kod procedury lub funkcji pascalowej stanowi odwołanie (adres - 2 bajty) do właściwego obszaru pamięci rezerwowanej dla kodu programu. Statyczna struktura punktu zintegrowanego reprezentowana jest w 48 bajtach. W języku PASCAL typ porządkowy przyjmujący

wartość z określonego przedziału może zostać przedstawiony w postaci typu wyliczeniowego. Dla ułatwienia identyfikacji elementów struktury punktu zintegrowanego zadeklarowano typ wyliczeniowy zawierający identyfikatory x, y, z (współrzędne geometryczne), Vx, Vy, Vz (składowe wektora), Sc (skalar) oraz B (wyróżnik stanu).



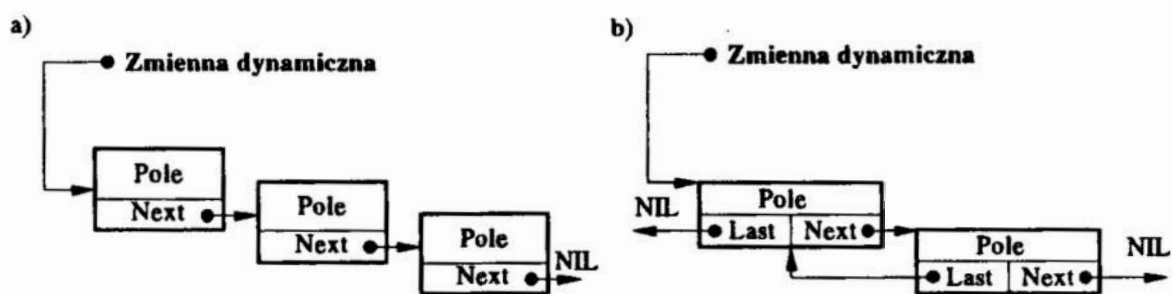
Rys. 6. Programowanie obiektowe - reprezentacja danych
Fig. 6. Object oriented programming - data representation

Programowanie obiektowe opiera się na trzech generalnych zasadach :

- dziedziczność- obiekty mogą być zagnieżdżane tworząc hierarchiczną strukturę, w której potomek (obiekt podrzędny) dziedziczy właściwości przodka (obektu nadrzędnego),
- polimorfizm - różne zmienne typu obiektowego mogą posługiwać się właściwymi im metodami wirtualnymi,
- kompleksowość - stwarza możliwość konstrukcji obiektu, którego pola danych dostępne są jedynie w sposób pośredni, poprzez odpowiednie metody obiektu.

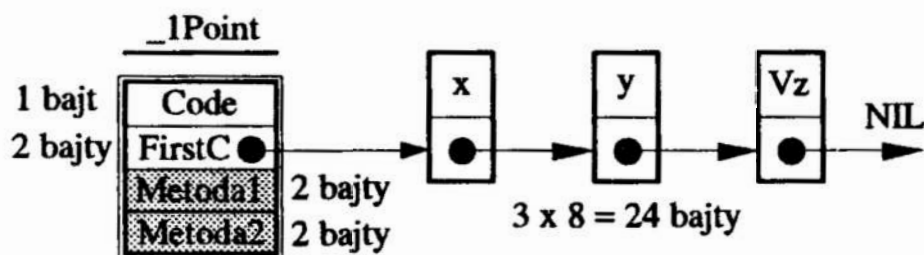
Punkt zintegrowany jako obiekt

Niech obiekt reprezentujący punkt zintegrowany ma identyfikator `_1POINT`. Metody (funkcje i procedury) obiektu operują na polu danych do tej pory opisywanym jako struktura statyczna. Różnorodność danych, które mogą być reprezentowane przez punkt zintegrowany pozwala sądzić, że przyjęcie sztywnej struktury statycznej jest nadmiarowe. Zapotrzebowanie pamięci na taką daną wynosi 48 bajtów, bez względu na faktycznie reprezentowane dane. W skrajnym przypadku liniowego (geometrycznie) układu mechanicznego typu pręt, położenie punktu na pręcie jest jedną daną x lub y lub z. Optymalizacja pola danych obiektu `_1POINT` z uwagi na zapotrzebowanie pamięci polega na wykorzystaniu struktury listy. Na rys. 7 przedstawiono listę jedno- i dwukierunkową.



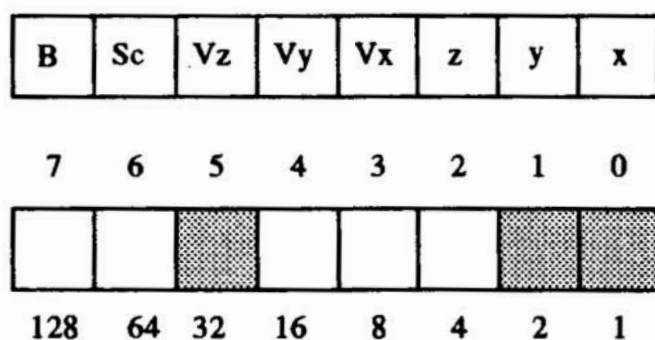
Rys. 7. Struktura listowa a) jednokierunkowa (FIFO), b) dwukierunkowa
 Fig. 7. Dynamic structure - list

Tworzenie obiektu typu `_IPOINT` z wykorzystaniem listy jednokierunkowej FIFO (First In First Out) przedstawiono na rys. 8.

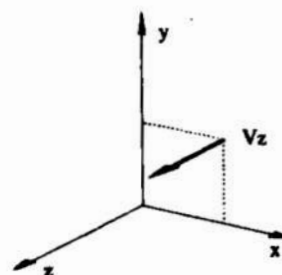


Rys. 8. Obiekt z listą danych punktu zintegrowanego
 Fig. 8. Object with data list of the integrated point

Lista jest strukturą nieograniczoną o dostępie sekwencyjnym. Kolejność elementów w liście nie identyfikuje automatycznie sensu zmiennej, stąd konieczność kodowania i dekodowania elementów listy. Sposób kodowania przedstawiono na rys. 9. Jednobajtowe pole `Code` wprowadzone zostaje do obiektu `_IPOINT`. Wartość pola `Code` określana jest na podstawie aktualnych zmiennych reprezentowanych przez punkt zintegrowany.



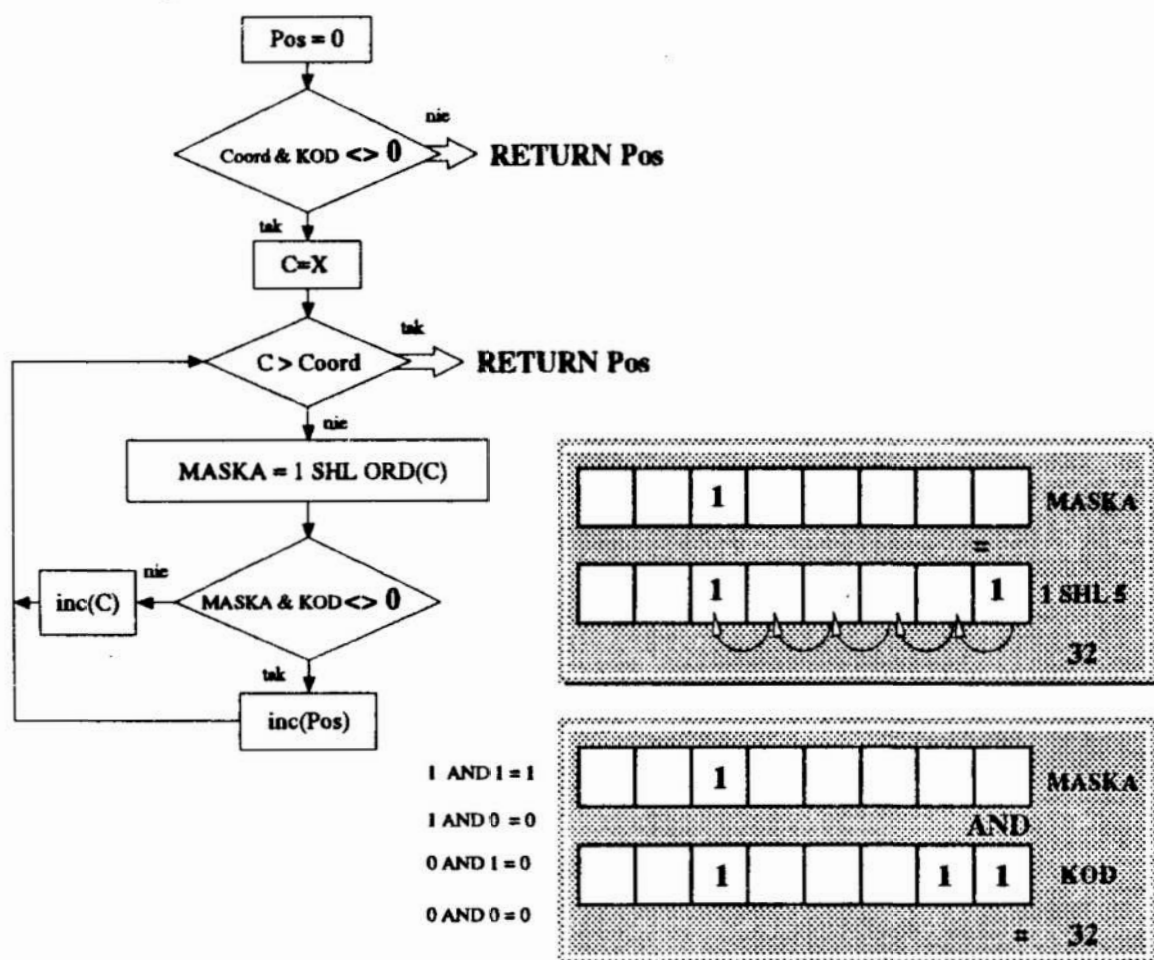
$$KOD = 1 + 2 + 32 = 35$$



Rys. 9. Sposób kodowania
 Fig. 9. Method of data coding

Czynnością odwrotną jest dekodowanie, którego schemat sieciowy przedstawiono na rys. 10. Dekodowanie polega na przypisaniu zmiennej `Pos` pozycji w liście elementu identyfikowanego przez zmienną `Coord` na podstawie wartości pola `Code`.

Ponadto na rys. 10 zilustrowano operacje bitowe (SHL i AND) użyte w algorytmie dekodowania.



Rys. 10. Sposób dekodowanie współrzędnej
Fig. 10. Method of data decoding

3. Podsumowanie

Przedstawione wyżej uwagi stanowią przesłankę do zbudowania obiektowej struktury danych zorientowanej na komputerowe wspomaganie projektowania.

Punkt zintegrowany zdefiniowany został jako kompleksowy obiekt z dynamiczną listą współrzędnych. Interpretacja pola danych zawartych w liście współrzędnych odbywa się za pomocą procedur kodującej i dekodującej listę. Struktura obiektu nie stanowi końcowej, niezmiennalnej postaci. Zmiany i modyfikacje dotyczyć mogą:

- dynamicznej listy współrzędnych punktu zintegrowanego - w proponowanym punkcie zintegrowanym lista współrzędnych ograniczona jest do ośmiu elementów. Wynika to z przyjęcia typu pola *Code* jako *byte*, czyli dana jednobajtowa (8 bitów). Wydłużenie listy współrzędnych polega na przyjęciu jako *Code* zmiennej wykorzystującej większą liczbę bajtów, np. *word* (2 bajty - 16 bitów),
- kompleksowości obiektu *_IPOINT* - klasa zagadnienia determinuje funkcje częściowe dziedziny algorytmicznej, dla której punkt zintegrowany jest daną

- bazową. Oprócz metod pośredniego dostępu do elementów listy współrzędnych korzystne może być zimplementowanie funkcji częściowych obliczających wielkości istotne z punktu widzenia danego zastosowania,
- c. zagnieżdżenia obiektów potomnych w hierarchicznej strukturze danych _ inny kierunek rozwoju struktury danych polega na rozbudowie hierarchicznej struktury obiektów bazujących na podstawowym obiekcie _1POINT.

Literatura

1. Iglewski M., Madey J., Matwin S., *Pascal*, WN-T, Warszawa, 1986
2. Wirth N., *Algorytm+Struktura Danych=Program*, WNT, Warszawa, 1989
3. O'Brien S, *Turbo Pascal 6. The Complete Reference*, Osborn McGraw-Hill, 1991

STRESZCZENIE

W referacie przedstawiono strukturę danych opisującą reakcje wejścia i wyjścia komputerowego modelu układu mechanicznego. Zaproponowano założenia implementacji tej struktury przy użyciu programowania zorientowanego obiektowo (OOP).

DATA STRUCTURE FOR COMPUTER ASSISTED DESIGN

Summary

The data structure describing the input- output reaction of the computer model of the mechanical system is presented. The implementation of data structure in object oriented programming (OOP) is suggested.

POJĘCIE MODELU KOMPUTEROWEGO

1. Modelowanie układów mechanicznych

Każdy rzeczywisty układ (system) techniczny traktowany jako obiekt materialny niesie w sobie olbrzymią ilość informacji. Jeżeli nawet założyć, że istnieje realna możliwość zkwantyfikowania tych informacji to ich przetwarzanie w komplecie może okazać się niemożliwe. W celu opisanie rzeczywistego układu (systemu) należy postawić cel, dla którego ten opis jest sporządzany. Umożliwia to wyróżnienie tych informacji z kompletnego opisu, które są istotne dla realizacji postawionego celu a pominięcie mniej ważnych lub całkowicie nieistotnych. Opis rzeczywistego systemu sporządzony pod kątem celu stanowi jego model przedstawiony w sposób formalny lub nieformalny. Jedną z definicji modelu przytaczana w literaturze [2][3][5] brzmi :

Modelem danego rzeczywistego obiektu jest układ dający się wyobrazić lub materialnie zrealizować, który odzwierciedlając lub odtwarzając obiekt zdolny jest zastąpić go tak , że jego badanie dostarcza nowych, nadających się do dalszego sprawdzania informacji o obiekcie.

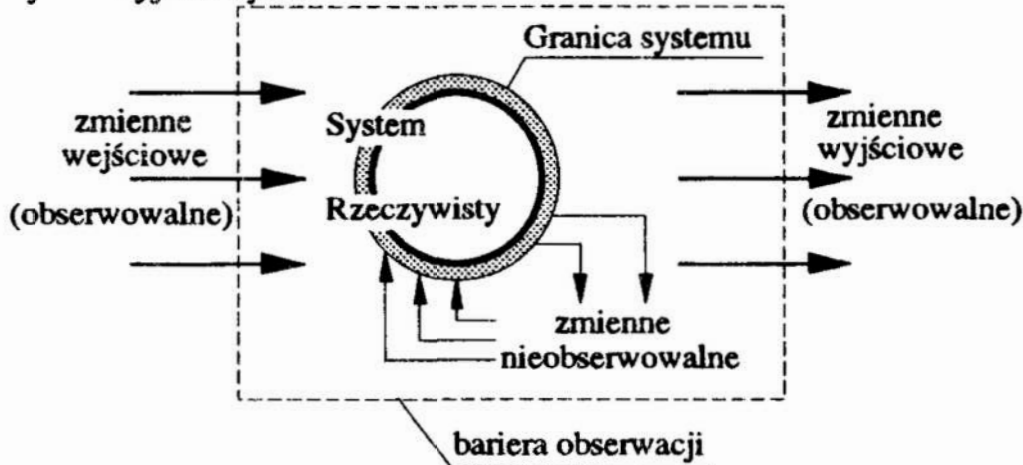
Oznacza to, że tworzenie modelu czyli modelowanie polega na wykorzystaniu podobieństwa pomiędzy dwoma układami, z których jeden jest odwzorowaniem drugiego tworząc powiązanie hierarchiczne, i prowadzone jest w celu otrzymania charakterystyki pierwszego układu na podstawie analizy drugiego układu stwarzającego dogodniejsze warunki obserwacji.

Nieformalny opis modelu zawiera opis jego elementów, zmiennych opisowych i interakcji elementów [4]. Rzeczywisty obiekt w ujęciu systemowym stanowi izolowany przez granice systemu układ, któremu przyporządkowuje się zmienne opisowe. Zmienne opisowe podzielić można na zmienne obserwowalne i zmienne nieobserwowalne. Kryterium podziału zmiennych wynika z istnienia mierników, metod pomiaru i aparatury pomiarowej zdolnej nadać wartość określonej zmiennej.

Zmienne obserwowalne dzieli się na zmienne wejściowe i wyjściowe. Zmienne wejściowe traktować należy jako przyczynę a zmienne wyjściowe jako skutek. Wszystkie zmienne wejściowe stanowią segment wejściowy, natomiast zmienne wyjściowe tworzą segment wyjściowy. Zbiór wszystkich par segmentów wejścia-wyjścia tworzy reakcję wejścia-wyjścia systemu rzeczywistego. Na rysunku 1 przedstawiono elementy opisu systemu rzeczywistego. Zmienne nieobserwowalne

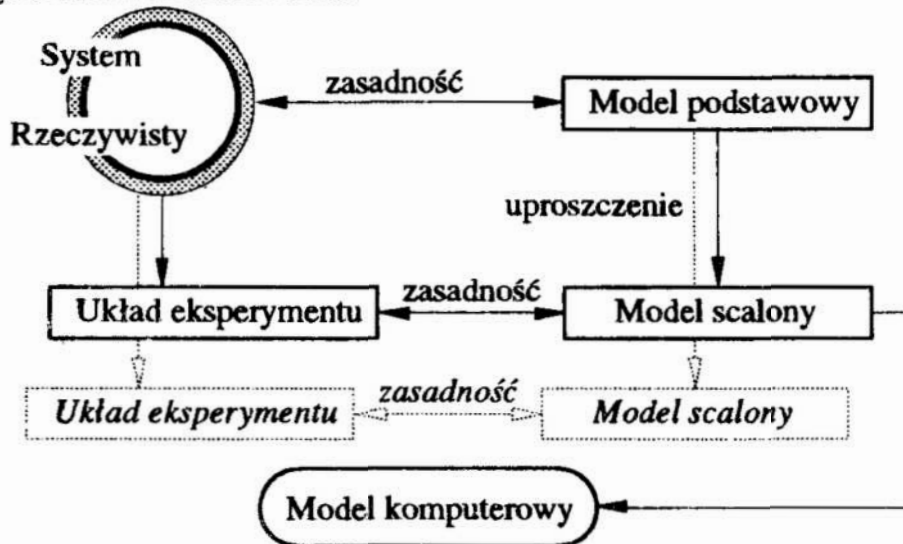
* Politechnika Gdańska
Wydział Mechaniczny Katedra Konstrukcji i Eksploatacji Maszyn
80-352 Gdańsk ul.Narutowicza 11/12

mieszczące się wewnątrz bariery obserwacji mogą mieć charakter zmiennych wejściowych i wyjściowych.



Rys. 1. Elementy opisu systemu rzeczywistego
 Fig. 1. Elements of description of a real system

Opis obiektu rzeczywistego zawierający komplet informacji o obiekcie tworzy tzw. **model podstawowy**. W praktyce kompletny opis układu rzeczywistego jest niemożliwy. Opis obiektu rzeczywistego pod kątem celu prowadzi do wyselekcjonowania niektórych jego elementów, zmiennych opisowych i interakcji elementów. Ogranicza to równocześnie zbiór reakcji wejścia-wyjścia. Taki podzbiór reakcji wejścia-wyjścia tworzy **układ eksperymentu**, w którym system rzeczywisty będzie obserwowany, badany lub analizowany. Na rysunku 2 wprowadzono dwa układy eksperymentu. Jednakże, może istnieć tyle układów eksperymentu ile jest wyobrażalnych celów modelowania.



Rys. 2. Schemat procesu modelowania
 Fig. 2. Schematic representation of modelling process

Podczas tworzenia uproszczonego modelu reakcje wejścia-wyjścia podlegają również uproszczeniu. Jeżeli reakcje wejścia-wyjścia modelu uproszczonego są zgodne, dla określonego kryterium porównawczego, ze wszystkimi reakcjami układu eksperymentu, wówczas **model skalony** (uproszczony) jest zasadny. Zasadność modelu

jest ściśle związana z układem eksperymentu i kryterium, według którego określana jest zgodność reakcji wejścia-wyjścia. Model zasadny dla jednego układu eksperymentu nie musi być zasadny dla innego układu, stąd modeli zasadnych może być tyle ile jest układów eksperymentu. Zasadność modelu posiada gradację, jest stopniowalna. Wyróżnia się następujące, o rosnącej wadze, stopnie zasadności :

- zasadność replikowalna - zasadność jest replikowalna, jeżeli generowane przez model dane odpowiadają danym uzyskanym poprzednio z systemu rzeczywistego,
- zasadność predykcyjna - zasadność jest predykcyjna, jeżeli można zapewnić zgodność reakcji modelu i systemu rzeczywistego przed uzyskaniem danych z systemu rzeczywistego,
- zasadność strukturalna - zasadność strukturalna oznacza, że model nie tylko generuje taką samą reakcję jak system rzeczywisty, lecz także odzwierciedla sposób w jaki działa rzeczywisty obiekt.

Wprowadzone na rysunku pojęcie modelu komputerowego, nieznajdujące miejsca w poprzedzającym rysunek opisie, zostanie omówione w dalszej części referatu.

Przyjęcie określonego celu modelowania implikującego układ eksperymentu i zasadność modelu prowadzi do zróżnicowania modeli. Poniżej opisano w skrócie kategorie modeli przyjmując jako kryterium podziału :

- czas : model *ciągły w czasie*, model *dyskretny w czasie*,
- zmienne opisowe : model *dyskretny* oraz model *ciągły ze względu na zmienne opisowe*,
- zmienne losowe : model *deterministyczny*, model *probabilistyczny* ,
- oddziaływanie systemu rzeczywistego z otoczeniem : model *autonomiczny*, model *nieautonomiczny*,
- stałość reguł interakcyjnych modelu : model *niezmienny w czasie*, model *zmienny w czasie*.

Oprócz wyżej wymienionych kategorii modeli, można dokonać negatywnego podziału modeli :

- model jest *niekompletny*,
- model jest *nielogiczny*,
- model jest *niejednoznaczny*.

Model, niezależnie od sposobu podejścia do wiedzy o systemie rzeczywistym a priori lub a postpriori, reprezentuje wiedzę o strukturze, o wartościach parametrów oraz o stanie systemu w pewnej chwili czasu.

2. Model komputerowy

Model scalony uzyskany w efekcie zabiegów zilustrowanych na rysunku 2 nie jest celem samym w sobie. Oczwistym celem tworzenia modelu jest jego przetworzenie dla uzyskania odpowiedzi na postawione pytanie. W ujęciu komputerowego wspomaganie projektowania przetwarzanie polega na wykorzystaniu komputera jako narzędzia w tym procesie. Określone właściwości narzędzia implikują postać modelu przystosowaną do przetwarzania komputerowego tworząc **model komputerowy**.

Komputer jest urządzeniem obliczeniowym (liczącym) za pomocą którego można generować reakcje wejścia-wyjścia modelu scalonego. Proces ten jest realizowany poprzez wykonywanie instrukcji programu komputerowego. Wykonywane *krok po kroku* instrukcje ustalane są na podstawie modelu scalonego tworząc określony algorytm.

Algorytm jest to sposób postępowania, który ma prowadzić w sposób automatyczny do rozwiązania określonego zadania.

Zakłada się, że pewne pierwotne instrukcje tego sposobu postępowania są wykonalne, tzn., że są one zdefiniowane i w algorytmie nie trzeba ich definiować lecz można je używać. Zbiór obiektów wraz z instrukcjami pierwotnymi nazywa się dziedziną algorytmiczną [1].

Dziedzinę algorytmiczną zapisać można jako :

$$(A, f_1, \dots, f_n, r_1, \dots, r_m) \quad (1)$$

gdzie :

A jest zbiorem niepustym,

$f_1 \dots f_n$ są funkcjami częściowymi określonymi dla argumentów ze zbioru A i przyjmującymi wartości ze zbioru A ,

$r_1 \dots r_m$ są relacjami zachodzącymi pomiędzy elementami zbioru A .

Jeżeli relacja r_j jest podzbiorem iloczynu kartezjańskiego A^k , to relacja ta jest k -argumentowa oraz dla układu (x_1, \dots, x_k) jest spełniona wtedy i tylko wtedy kiedy układ (x_1, \dots, x_k) należy do r_j .

Funkcja k -argumentowa częściowa f_j jest $k+1$ -argumentową relacją, taką że dla ustalonego (x_1, \dots, x_k) istnieje co najwyżej jeden argument x_{k+1} , dla którego układ $(x_1, \dots, x_k, x_{k+1})$ należy do tej relacji tzn. :

$$f_j(x_1, \dots, x_k) = x_{k+1} \quad (2)$$

Funkcja, która jest określona dla wszystkich układów (x_1, \dots, x_k) , nazywa się funkcją całkowitą.

Podstawową definicję dziedziny algorytmicznej (1) rozwinąć można w złożoną dziedzinę opisaną jako :

$$((A, f_1 \dots f_n, r_1 \dots r_m)_1, (A, f_1 \dots f_n, r_1 \dots r_m)_2, \dots, (A, F_1 \dots F_n, R_1 \dots R_m)_u) \quad (3)$$

gdzie :

$(A, f_1 \dots f_n, r_1 \dots r_m)_{u-1}$ - dziedziny algorytmiczne predefiniowane w systemie komputerowym

$(A, F_1 \dots F_n, R_1 \dots R_m)_u$ - dziedzina algorytmiczna (odpowiednio zbiór, funkcje częściowe oraz relacje) określana przez użytkownika.

Przyjmując konsekwentnie, że model zależy nie tylko od celu dla, którego zostaje tworzony ale również od narzędzia jakim dysponujemy przy jego przetwarzaniu, tzn. komputera i dziedziny algorytmicznej, pojawia się sekwencja :

Model Nominalny \Rightarrow Model Scalony \Rightarrow Model Komputerowy

Model komputerowy to dyskretny lub zdyskretyzowany model scalony lub matematyczny zapisany w postaci programu komputerowego we właściwej dziedzinie algorytmicznej. Oznacza to konieczność racjonalizacji struktury danych oraz funkcji częściowych dziedziny algorytmicznej.

Jak już wspomniano modele dzieli się na ciągłe i dyskretne w czasie oraz ze względu na zmienne opisowe. Większość układów mechanicznych są to układy o ciągłym rozłożeniu parametrów (czas lub współrzędne). Odpowiadający układowi model matematyczny budowany jest za pomocą równań różniczkowych lub całkowych o odpowiednich warunkach brzegowych. Model dyskretny to model o parametrach skupionych. Modele dyskretne stosowane są znacznie częściej do opisywania układów mechanicznych niż modele ciągłe, przede wszystkim ze względu na prostszą analizę równań opisujących układ. Często modele ciągłe zastępuje się równoważnymi modelami dyskretnymi. Stosuje się przy tym różne metody dyskretyzacji, np. metodę sztywnych elementów skończonych lub odkształcalnych elementów skończonych. Podawane w literaturze [3] podejście do modelowania układów mechanicznych na podstawie wiedzy a priori i wiedzy a postpriori zakłada dyskretyzację modelu w obu przypadkach.

W podanym wyżej określeniu modelu komputerowego akcentowane jest dyskretna postać modelu wynikająca z przyjętego sposobu działania *krok po kroku* maszyny liczącej, przyczym podany wyżej powód dyskretyzacji modelu zachowuje ważność. Na tym poziomie ogólności określenia modelu komputerowego nie można mówić o sposobie dyskretyzacji modelu. Pojęcie algorytmu zostało omówione wyżej. Strukturę danych jako integralną część modelu komputerowego należy rozważyć oddzielnie.

3. Podsumowanie

Przyjęcie pojęcia modelu komputerowego umożliwia podjęcie dyskusji na temat efektywności komputerowego wspomaganie projektowania. Użycie komputera jako narzędzia przy analizie modeli układów mechanicznych umożliwia rozpatrywanie modeli o złożoności do niedawna niewyobrażalnej. Wynika to z powszechnie znanej właściwości komputera do prowadzenia obliczeń z wielką prędkością w sposób dokładnie określony instrukcją. Pomimo tego istnieją pewne ograniczenia możliwości zarówno od strony narzędzia-komputera jak i od strony człowieka-twórcy modelu i następnie programu. Ograniczenia te można rozpatrywać z następujących punktów widzenia :

- czasu wykonania jednego kroku według założonego algorytmu skąd wynika skumulowany czas wymagany do zrealizowania programu,
- pamięci, jakiej wymaga program komputerowy do opisu struktury modelu i do przechowywania wartości zmiennych opisowych i stanu modelu,
- środowiska związanego z przekształcaniem modelu w reprezentujący go program tzn. translacji, kompilacji i tłumaczenia z języka wysokiego poziomu na język maszynowy oraz z przesyłaniem kodu lub danych w obrębie pamięci krótkookresowej lub zewnętrznej,

- czasu i wysiłku związanego z weryfikacją modelu w odniesieniu do programu komputerowego, przygotowaniem danych oraz z wynajdywaniem i usuwaniem błędów w programie.

Pierwsze trzy ograniczenia mają charakter informatyczny związany ze sprzętem i oprogramowaniem. Współczesne komputery, również klasy PC, posiadają tak olbrzymie możliwości, że problem sprzętu jest problemem ekonomicznym a nie technicznym. Natomiast problem wyboru oprogramowania, w tym również języków programowania jest ściśle związany z czwartym ograniczeniem przypisanym twórcy modeli i programów komputerowych. Poprawny model matematyczny można zapisać za pomocą różnych algorytmów, przy użyciu różnych struktur danych zapisując kod źródłowy w jednym z języków programowania. Nie wystarczy argument, że algorytm musi być poprawny (nie mówię o błędach programisty). Poprawny to nie znaczy optymalny, optymalny to nie znaczy najszybszy. Dwa podstawowe kryteria z punktu widzenia informatycznego to szybkość działania oraz wykorzystanie pamięci operacyjnej przez kod programu oraz dane, których badanie i optymalizacja jest opisana w literaturze.[1]

Literatura

1. Banachowski L., Kreczmar A., *Elementy analizy algorytmów*, WNT, W-wa, 1989
2. Dietrich M.(red), *Podstawy Konstrukcji Maszyn (t.1)*, PWN, W-wa, 1986
3. Giergiel J., Uhl T., *Identyfikacja układów mechanicznych*, PWN, W-wa, 1990
4. Zeigler P.B., *Teoria modelowania i symulacji*, PWN, Warszawa, 1984
5. Tarnowski W. Kiczowski T., *Komputerowe wspomaganie projektowania*, WSI Koszalin, 1992

Streszczenie

W referacie przedstawiono rolę modelowania układów mechanicznych w odniesieniu do komputerowego wspomaganie projektowania. Wykazano potrzebę właściwego przygotowania modelu układu w postaci akceptowanej przez komputer jako warunek efektywnego wykorzystania oraz tworzenia oprogramowania CAD. Zaproponowano definicję modelu komputerowego.

DEFINITION OF A COMPUTER MODEL

Summary

The role of models in computer assisted design is presented. The need for modelling of main system variables in order to structure them in a form acceptable by computer prior to the use of a CAD package is argued. This preparatory modelling is a necessary condition for an effective utilisation of any computer aided design software. A definition of a computer model is suggested.

Roman STRYCZEK

Katedra Technologii Maszyn i Automatykacji

Filii Politechniki Łódzkiej w Bielsku-Białej

43-300 Bielsko-Biała ul. Willowa 2

Wykorzystanie sieci Petri dla aktywnej optymalizacji przebiegu operacji tokarskich.

1. Wprowadzenie.

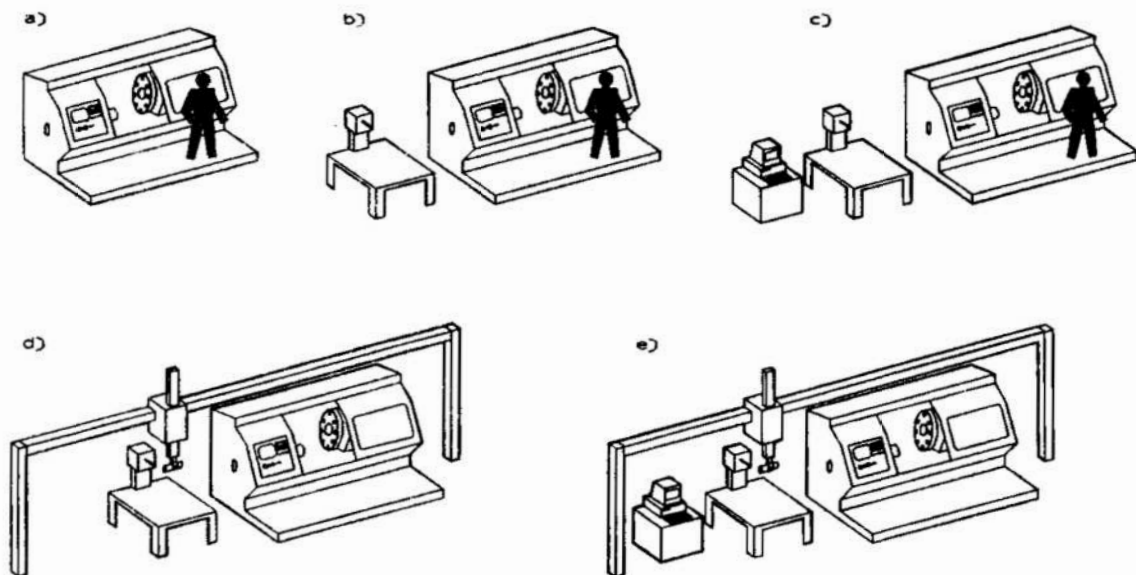
Od współczesnych obrabiarek pracujących w przemyśle maszynowym o masowym charakterze produkcji wymaga się coraz wyższej wydajności, niezawodności, jakości produkowanych wyrobów oraz coraz dłuższych okresów pracy bezobsługowej. Prowadzone w tej dziedzinie od szeregu lat intensywne badania można podzielić generalnie na dwa kierunki. Pierwszy związany z obrabianym materiałem obejmuje zespół zagadnień związanych ze skrawalnością materiału obrabianego, optymalizacją parametryczną operacji, poszukiwaniu nowych metod obróbki. Drugi kierunek związany jest z materiałem narzędzia, jego geometrią, warunkami pracy oraz monitorowaniem stanu i prognozowaniem żywotności ostrza skrawającego. Obecny stan wiedzy nie jest jednak dostateczny aby zbudować wyczerpujący, fizyczny model zużycia narzędzia [2]. Oba nurty badań są oczywiście bardzo istotne, tym niemniej nie wyczerpują czynników wpływających na efektywną wydajność współczesnego stanowiska tokarskiego. Jednym z takich czynników jest miejsce i częstotliwość czynności pomiarowych w procesie obróbki związane z warunkami zapewnienia jakości produkcji [6].

2. Mikrokomputer - jako dodatkowe wyposażenie stanowiska produkcyjnego.

W ostatnich latach coraz częściej uzupełnia się stanowiska obróbkowe (zwłaszcza autonomiczne stacje obróbkowe - ASO) w dodatkowy komputer, sterujący pracą całego stanowiska, analizą przebiegu procesu technologicznego, prognozowaniem zużycia ostrzy narzędzi, wyborem optymalnego wariantu przebiegu procesu, ewentualnie jego synchronizacją z pozostałymi komórkami produkcyjnymi. Pod koniec 1985r. w RFN 32% ASO korpusowych było sterowanych przez taki komputer, natomiast - co jest zastanawiające - żadne ASO tokarskie.

Na rys.1 przedstawiono trzy (a,b,d) najbardziej typowe z obecnie spotykanych konfiguracji stanowisk tokarskich oraz dwie dodatkowe (c,e) uzupełnione o mikrokomputer. Podstawowe elementy składowe tego rodzaju stanowisk to: obrabiarka wraz z układem sterownia numerycznego; zespół transportu i wymiany przedmiotów obrabianych; zespół urządzeń kontrolno-pomiarowych (czujniki wymiarowe, profilografometry itp.) zewnętrznych lub wewnętrznych względem obrabiarki; operator

obrabiarki; mikrokomputer sterujący procesem obróbki. W tabeli 1 sklasyfikowano najistotniejsze funkcje tego typu stanowisk wyróżniając ich podział na funkcje decyzyjne i wykonawcze. Jak z powyższej tabeli wynika komputer jest w stanie przejąć wszystkie funkcje decyzyjne oraz dodatkowo funkcje wykonawcze w zakresie korekty zmiany toru narzędzia (wynikającej np. ze ściernego zużywania się ostrza) i całościowego sterowania



Rys. 1. Konfiguracje typowych stanowisk tokarskich CNC.
Fig. 1. Presentation of typical machining cells for turning.

Funkcje decyzyjne/wykonawcze	Człowiek	USN	Komputer	Urządzenia pomocnicze
Obróbka	a,b/	d/a,b,c,d,e	c,e/	
Wymiana przedmiotu	/a,b	a,b,d/	c,e/	/d,e
Transport przedmiotu	a,b/a,b,c	d/	c,e/	/d,e
Wymiana ostrzy	a,b/a,b	d/c,d,e	c,e/	
Pomiar narzędzi	a,b/a	d/	c,e/	/b,c,d,e
Korekcja toru narzędzi	a,b/a,b	d/d	c,e/c,e	
Analiza trwałości ostrzy	a	b,d	c,e	
Pomiar przedmiotu	a,b/a	d/	c,e/	/b,c,d,e
Analiza pomiarów	a,b	d	c,e	
Sterowanie procesu	a,b/a,b	d/d	c,e/c,e	

Tab.1. Zestawienie podstawowych funkcji dla tokarek CNC.
Tab.1. Range of functions of automated lathe treatment.

procesem produkcyjnym na tego typu stanowiskach. Opcja 'd' pracująca bez operatora oraz bez mikrokomputera wymaga bardzo specjalistycznego oprogramowania, najczęściej samego sterownika (program PLC). W przypadku bardziej nowoczesnych układów sterowania numerycznego (np. USN Sinumerik serii 800), jest możliwe skorzystanie ze specjalnych języków kodowych (współpracujących z programami technologicznymi w formie nakładek) co pociąga za sobą znaczne komplikacje struktury programu i konieczność rozszerzenia go o szereg podprogramów specjalistycznych.

Użycie mikrokomputera umożliwiłoby zastosowanie szeregu rozbudowanych procedur optymalizujących operację technologiczną. Przykładem może tu być kontrola trwałości ostrzy narzędzi lub wybór optymalnego - w danej chwili - wariantu przebiegu operacji technologicznej.

Jeżeli uwzględnimy następujące zdarzenia, jakie mogą wystąpić w trakcie realizacji operacji tokarskiej na ASO:

- A - pomiar w trakcie operacji;
- B - pomiar po zakończeniu operacji;
- C - pomiar na obrabiarce;
- D - pomiar poza obrabiarką;
- E - wcięcie próbne, dla uprzedzającej kontroli nastawienia narzędzia;
- F - oczekiwanie na ostatni pomiar, w celu uwzględnienia jego wyników w analizie ustawienia i/lub zużycia narzędzia;
- G - przejście korekcyjne, jako dodatkowe przejście, w przypadku gdy wystąpił brak naprawialny,

to liczbę możliwych wariantów operacji można obliczyć jako sumę wyrażeń:

$$\begin{aligned}
 & | \{(A \wedge C), (A \wedge D)\} | \times | \{E, -E\} | \times | \{F, -F\} | = 8 \\
 & | \{(B \wedge C), (B \wedge D)\} | \times | \{F, -F\} | \times | \{G, -G, (G \wedge C), (G \wedge D)\} | = 16 \\
 H = & \{((A \wedge C) \wedge (B \wedge C)), ((A \wedge C) \wedge (B \wedge D)), ((A \wedge D) \wedge (B \wedge D)), ((A \wedge D) \wedge (B \wedge C))\} \\
 & | H | \times | \{E, -E\} | \times | \{F, -F\} | \times | \{G, -G, (G \wedge C), (G \wedge D)\} | = 64 \\
 & | \{-A \wedge -B \wedge -C \wedge -D \wedge -E \wedge -F \wedge -G\} | = 1
 \end{aligned}$$

uwaga: $-X$ oznacza zdarzenie przeciwne do X , $|X|$ - liczba elementów zbioru X .

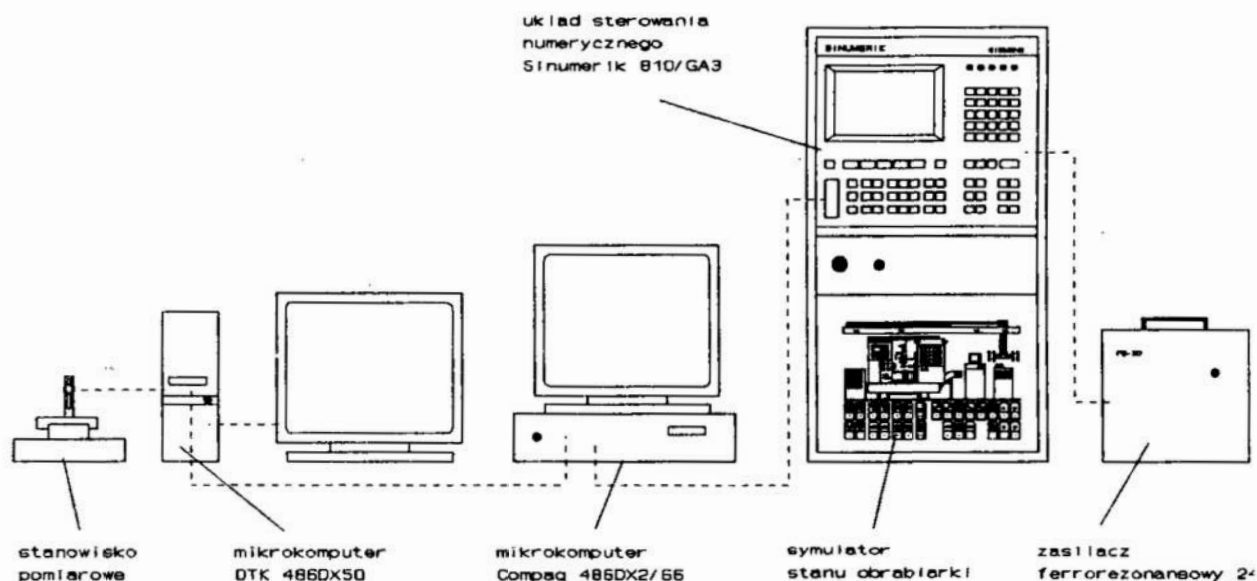
W sumie mogą więc wystąpić 89 różne warianty operacji. Gdyby przyjąć, że duża część z nich ma znaczenie tylko teoretyczne, to i tak zostanie kilkanaście realnych wariantów, spośród których należy w każdym cyklu procesu wybrać jeden w taki sposób, aby zadowolić przyjęte kryteria (max. wydajności, max. zysku, min. kosztu i/lub inne). W ogólnym przypadku problem wyboru decyzji sprowadza się do zagadnienia wielokryterialnego, wieloetapowego podejmowania decyzji w rozmytym otoczeniu i o nieskończonym horyzoncie planowania, co wynika z masowego charakteru produkcji. W praktyce zagadnienia powyższego nie można rozwiązać w sposób efektywny [4]. Dlatego proponuje się tutaj, ograniczenie horyzontu planowania do określonego czasu, zadania godzinowego lub zadania produkcyjnego oraz wypracowanie jednego zastępczego kryterium. Kryterium takie musi być elastyczne, czyli modyfikowalne w trakcie realizacji procesu produkcyjnego (np. możliwość zwiększenia tempa produkcji przy zwiększonych kosztach jednostkowych w przypadku gdyby dane stanowisko stało się wąskim gardłem procesu produkcyjnego w zakładzie). Rozmyte otoczenie sytuacji decyzyjnej wynika tutaj nie tylko z niestabilnych parametrów, odnoszących się do skrawalności materiału i trwałości ostrzy, ale przede wszystkim do informacji jakie można wykorzystać mierząc np. średnice obrabianych powierzchni.

Zdecydowanie więcej informacji może dostarczyć metoda opierająca się na danych uzyskanych z analizy mikronierówności obrabianych powierzchni. Biorąc jednak pod uwagę stosunkowo długi czas pomiaru mikronierówności metodami konwencjonalnymi, należy zwrócić baczniejszą uwagę na opto-elektroniczny wariant tej metody [5], (czas trwania samego pomiaru jest rzędu milisekund).

Zadaniem jakie autor postanowił zrealizować było:

- Zbudowanie stanowiska badawczego umożliwiającego przetestowanie założonej metody komunikacji komputer - układ sterowania numerycznego poziomu CNC oraz komputer - czujniki pomiarowe;
- Opracowanie pakietu oprogramowania zdolnego do sterowania w czasie rzeczywistym powyższego stanowiska;
- Opracowanie oprogramowania wykorzystującego metody wyboru optymalnych strategii procesu obróbki, przez prognozowanie jej przebiegu z użyciem symulacji komputerowej.

Rys.2 przedstawia ogólny szkic stanowiska jakie należało zbudować w celu sprawdzenia założonych metod komunikacji, sterowania i wykorzystania mikrokomputera w tokarskich stacjach autonomicznych. Zastosowanie dwóch mikrokomputerów zostało podyktowane możliwością pełniejszej kontroli (na bieżąco), wykonywanych często równolegle procedur sterowania i analizy procesu. W układzie rzeczywistym, nie badawczym, możliwe byłoby zastosowanie jednego komputera pracującego w trybie wielozadaniowym (co umożliwiłyby np. odpowiedni system operacyjny).



Rys. 2. Konfiguracja stanowiska badawczego.

Fig. 2. Configuration of a research station.

Głównym zadaniem mikrokomputera jest wykonywanie programu realizującego funkcje optymalizujące proces obróbki na podstawie mierzonych parametrów. Program sterujący pracą stanowiska badawczego został oparty na aparacie sieci Petri. Własności tego języka w naturalny sposób pozwalają na dynamiczny opis asynchronicznie

przebiegających zdarzeń, umożliwiając opis systemów w terminach zachodzenia wzajemnie warunkujących się zjawisk i procesów [1]. Wykorzystanie aparatu sieci Petri do modelowania różnorodnych aspektów systemów produkcyjnych jest już stosowane od szeregu lat. Obecnie sięga się do coraz to innych wersji tego języka [7]. Zastosowanie sieci Petri do sterowania procesem w czasie rzeczywistym w technologii budowy maszyn, poza problematyką elastycznych systemów produkcyjnych (ale też głównie teoretycznie), jest znikome. W tym przypadku postanowiono zastosować pewną podklasę języka sieci Petri (system miejsce/przejsie) do połączenie procedur optymalizacyjnych wykonywanych przez mikrokomputer z rzeczywistym procesem obróbki tokarskiej (w tym przypadku symulacją tego procesu na stanowisku badawczym).

3. Model procesu z wykorzystaniem języka sieci Petri.

Systemem miejsce/przejsie nazywamy uporządkowaną szóstkę postaci:

$$PN = (P, T, E, K, W, M_0),$$

gdzie:

- P - zbiór miejsc,
- T - zbiór przejść,
- E - relacja incydencji opisana na zbiorach P i T,
- K - funkcja pojemności miejsc,
- W - funkcja krotności łuków,
- M_0 - funkcja znakowania (markowania) początkowego.

Zbiór miejsc odpowiada zbiorowi warunków jakie mogą wystąpić w systemie. Zbiór przejść odpowiada zdarzeniom jakie występują w systemie. Oba te zbiory odpowiadają wierzchołkom sieci dwójakiego typu. Relacja E odpowiada zbiorowi łuków w sieci. Łuki te są typu pt, co oznacza, że warunek P musi być spełniony (posiadać marker), aby zaszło zdarzenie T lub typu tp co z kolei oznacza, że po zajściu zdarzenia T warunek P jest prawdziwy (otrzymuje marker). W ogólnym przypadku jedno miejsce może zawierać kilka markerów, o czym informuje funkcja K. Przewidziano także możliwość przepływu kilku markerów jednocześnie jednym połączeniem, o czym informuje funkcja W. Funkcja znakowania początkowego informuje, które warunki są prawdziwe przy starcie przepływu w sieci Petri.

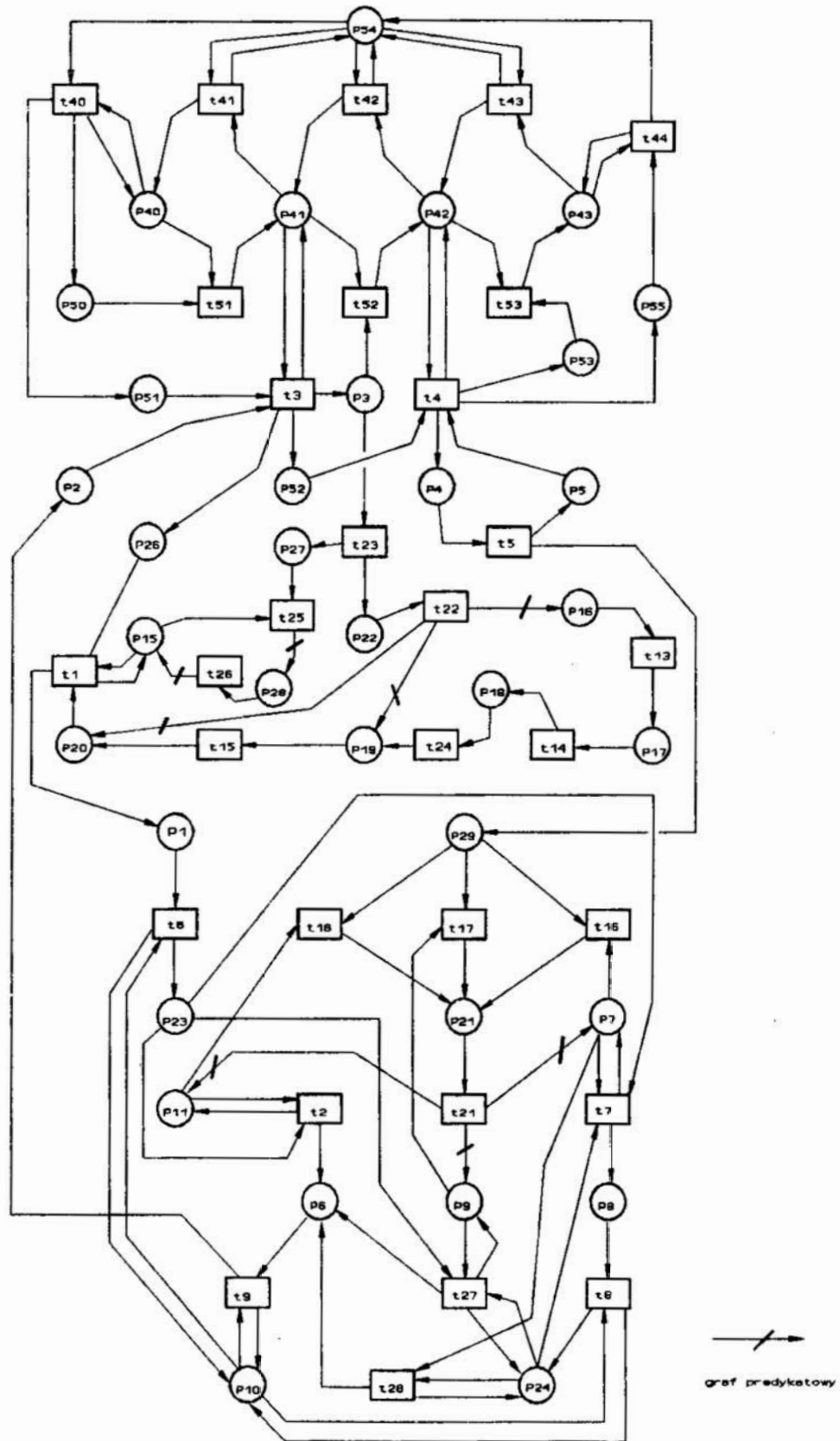
Załóżmy, że pojemność miejsc i krotność łuków jest zawsze równa 1. Wtedy elementy K i W w sieci PN można pominąć. Dodajmy równocześnie do sieci trzy dodatkowe zbiory:

- C - zbiór czasów przypisanych każdemu zdarzeniu ze zbioru T,
 - F - zbiór łuków predykatowych t_p ,
 - S - zbiór funkcji wykonywanych na łukach typu tp, oznaczanych f_{t_p} .
- W ten sposób otrzymamy uporządkowaną siódmkę postaci:

$$PN_f = (P, T, C, F, S, M_0),$$

którą można nazwać czasowo-predykatową siecią Petri.

Zbiór łuków predykatowych F umożliwia większą elastyczność modelowania systemu. Każdy element tego zbioru stanowi jakby dodatkową bramkę logiczną przepuszczającą marker przez łuk typu tp lub nie. Umożliwia to bardzo wygodne dla



Rys. 3. Sieć Petri.
 Fig. 3. Petri net.

projektanta systemu manipulowanie przepływem sieci przy zachowaniu wszystkich dodatnich cech formalnego opisu systemu siecią Petri. Wspomniana bramka logiczna może być prostą instrukcją warunkową lub bardzo skomplikowaną procedurą. Wynika to z faktu, że jej postać jest redagowana bezpośrednio w języku Prolog, co umożliwia użycie wszystkich narzędzi programowych jakie ten język zawiera. W omawianym stanowisku badawczym łukami predykatowymi posłużono się w dwóch przypadkach: przy decyzji o wyborze wariantu przebiegu operacji oraz przy analizie stanu ostrzy narzędzi w głowicy narzędziowej.

Łuki funkcyjne f_t_p podobnie jak łuki predykatowe są aktywne opcjonalnie. W odróżnieniu jednak od nich nie mogą zawierać instrukcji warunkowych (są zawsze realizowane). Przy ich pomocy można wykonywać szereg pomocniczych a bardzo użytecznych operacji jak np. zliczanie wykonanych cykli (sztuk) programu, zliczanie ilości zużytych ostrzy itp..

Opracowany program obsługi sieci Petri posiada interfejs komunikacyjny z użytkownikiem, pozwalający wybrać na starcie jeden z trzech trybów pracy programu:

- symulacja ciągła w czasie rzeczywistym;
- symulacja krok po kroku;
- symulacja przyśpieszona.

Symulacja ciągła umożliwia pracę systemu on-line, stanowi więc podstawowy tryb pracy stanowiska. Symulacja krok po kroku umożliwia testowanie poprawności sieci, poprzez śledzenie dokładne każdego odpalania przejścia. Na ekranie są kolejno ukazywane wszystkie aktywne warunki (warunki, które mają przydzielone aktualnie markery), a następnie wszystkie zdarzenia, które mają w tym czasie miejsce z zaznaczeniem, które zdarzenie skończy się najpierw. Symulacja przyśpieszona umożliwia szybkie przetestowanie sieci na dużej liczbie cykli, co pozwala na wykrycie w niej ewentualnych blokad. Ostatni tryb umożliwia również testowanie np. wydajności systemu produkcyjnego dla założonych z góry przebiegów i czasów trwania poszczególnych zdarzeń. Na rys.3 przedstawiono opracowaną sieć Petri dla wykonanego stanowiska badawczego. Zdarzenia od t_{40} do t_{53} opisują ruchy manipulatora transportującego przedmioty. Fragment sieci od P22 do P20 służy do sterowania wymianą narzędzi. Pętla t_{25} , P28, t_{26} , P15 umożliwia w przypadku przeniesienia sieci na rzeczywistą obrabiarkę zatrzymanie procesu (ewentualnie restart) przez rozkaz z komputera. Dolny fragment sieci służy do wyboru wariantu przebiegu operacji tokarskiej. Do testowania przyjęto operację wykonywaną jednym narzędziem, które wykonywało zarówno przejścia zgrubne jak i wykańczające. Przyjęto trzy warianty operacji: obróbka ciągła bez pomiaru i bez oczekiwania na pomiar ostatniej sztuki, obróbka z pomiarem po wykonaniu przejść zgrubnych oraz obróbka z oczekiwaniem przed przejściem wykańczającym na wyniki pomiaru poprzedniej sztuki.

W pełni potwierdziły się zalety języka PDC Prolog podczas realizacji oprogramowania powyższej problematyki. Pakiet oprogramowania składa się z dwóch modułów: programu Petri.pro, który zawiera oprogramowanie samej metody przetwarzania predykatowo-czasowej sieci Petri oraz modułu adaptacyjnego Sa.pro, w którym użytkownik może sobie dowolnie definiować łuki predykatowe i łuki funkcyjne oraz konfigurować dodatkowe okienka w celu pełniejszej kontroli przebiegu procesu.

LITERATURA

- [1] Banaszak Z.: Modelling and Control of FMS. Petri Net Approach. Wydawnictwo Politechniki Wrocławskiej, Wrocław 1991.
- [2] Gawlik J.: Prognozowanie stanu zużycia ostrzy narzędzi w procesie skrawania. Monografia, Wyd. Politechniki Krakowskiej, Kraków 1988.
- [3] Jamielniak K.: Diagnostyka stanu narzędzi w autonomicznych stacjach obróbkowych. mechanik 1988/4.
- [4] Kacprzyk J.: Zbiory rozmyte w analizie systemowej. PWN, W-wa 1986.
- [5] Karpiński T., Lukianowicz Cz.: Aparatura optyczno elektroniczna do oceny mikronierówności powierzchni. Materiały Konferencji "Postępy w teorii i technice obróbki materiałów", Kraków 1990.
- [6] Karpiński T.: Problemy metrologiczne w technologii maszyn. Materiały sympozjum "Wybrane problemy projektowania procesów technologicznych", Gdańsk 1994.
- [7] Siemiątkowski M.: Modelowanie współbieżnej realizacji procesów technologicznych w elastycznie zautomatyzowanym wytwarzaniu. Materiały sympozjum "Wybrane problemy projektowania procesów technologicznych", Gdańsk 1994.
- [8] Szadkowski J.: Wielonarzędziowa obróbka skrawaniem. Monografia. Zeszyt Naukowy Filii P.L. w Bielsku-Białej, Bielsko-Biała 1992.

Streszczenie

Referat rozpatruje zagadnienie optymalizacji struktury operacji tokarskiej wykonywanej na autonomicznej stacji obróbkowej, pracującej w warunkach produkcji masowej. Zajęto się wpływem dodatkowego wyposażenia stanowiska oraz struktury wykonywanej na nim operacji, ze szczególnym uwzględnieniem czynności pomiarowych na efektywną wydajność stanowiska. Zaproponowano metodologię badania tego typu problematyki, w oparciu o model sformułowany w języku czasowo-predykatowych sieci Petri. Praca została wykonana w ramach grantu nr 771379102.

PETRI NETS IN THE MODELLING AND CONTROL OF AUTOMATED LATHE TREATMENT.

Summary

The paper examines the problem of a work station efficiency. The work station represents a lathe flexible manufacturing cell working in mass production conditions. We have been engaged in the effect of both station additional equipment and the structure of an operation made in this work station. The effect of a measurement activity on the work station efficiency has been taken into account in particular. The research methodology of the problems has been proposed basing on the model formed in the language of time Petri nets. This work is supported by the State Committee for Scientific Research under grant number 771379102.

KOMPUTEROWE MODELOWANIE OBCIĄŻENIA TEMPERATURĄ OBROTOWYCH KONSTRUKCJI POWŁOKOWYCH

1. Wstęp

Do projektowania bezpiecznych i niezawodnych konstrukcji niezbędne są dokładne dane o obciążeniach, w tym również od wpływów temperatury. Obowiązująca od 1987 roku norma [1] nakłada formalny obowiązek uwzględnienia obciążenia konstrukcji temperaturą, wywołanego wpływami klimatycznymi. Dane zawarte w normie [1] oraz w jej międzynarodowym pierwowzorze [2] odnoszą się jednak do stosunkowo ograniczonej klasy konstrukcji. Są to konstrukcje, których kształt reprezentuje układ pionowych przegród płaskich ustawionych co 45° względem kierunku północy geograficznej oraz przegroda pozioma. Stąd też nie ma możliwości bezpośredniego użycia tych danych do wyznaczenia obciążenia temperaturą powłok o dowolnej geometrii. W niniejszej pracy przedstawiono dokładniejszy algorytm wyznaczania obciążenia temperaturą powłok, równocześnie wykorzystując w możliwie największym zakresie dane zawarte w normie [1]. Na tej podstawie opracowano pakiet programów komputerowych do wyznaczania obciążenia temperaturą konstrukcji o kształcie cienkiej powłoki obrotowej, poddanych wpływom klimatycznym i eksploatacyjnym. Zamieszczono również wyniki obliczeń, dla powłoki hiperboloidalnej chłodni kominowej.

2. Obciążenie powłoki temperaturą

Przyjęto, że obciążenie temperaturą powłoki stanowią funkcje określające przyrost temperatury jej powierzchni zewnętrznej ($\Delta\vartheta_e$) oraz powierzchni wewnętrznej ($\Delta\vartheta_i$) względem tzw. temperatury scalenia ϑ_o :

$$\begin{aligned}\Delta\vartheta_e(x, \Theta, \tau) &= \bar{\vartheta}(x, \Theta, \tau) + \Delta\vartheta(x, \Theta, \tau) / 2 - \vartheta_o \\ \Delta\vartheta_i(x, \Theta, \tau) &= \bar{\vartheta}(x, \Theta, \tau) - \Delta\vartheta(x, \Theta, \tau) / 2 - \vartheta_o\end{aligned}\tag{1}$$

gdzie: x, Θ, τ - współrzędne: południkowa, równoleżnikowa, czasowa, $\bar{\vartheta}$ - temperatura powłoki w powierzchni środkowej, $\Delta\vartheta$ - różnica temperatury na grubości powłoki.

Na podstawie danych klimatycznych zamieszczonych w normie [1], można przyjąć, że ekstremalne wartości funkcji (1) wystąpią w okresie letnim i zimowym. Wartości te

* Wyższa Szkoła Inżynierska w Opolu, Wydział Budownictwa
ul. Katowicka 48, 45-951 Opole

można uważać za ekstremalne obciążenie temperaturą w sensie wystąpienia maksymalnych wartości sił wewnętrznych w konstrukcji.

3. Wpływy klimatyczne i technologiczne

Uwzględniając zalecenia ujęte w [1] i [2] przyjęto, że obciążenie powłoki temperaturą uzależnione jest od następujących czynników.

(a) temperatury powietrza atmosferycznego (t_e)

$$t_e(\tau) = \bar{t}_e + a \sin[2\pi(\tau - \varphi)/24] \quad (2)$$

gdzie: \bar{t}_e, a - wartość charakterystyczna średniej dobowej temperatury powietrza atmosferycznego oraz jej amplituda dobową ($^{\circ}\text{C}$), τ - czas zegarowy (godz), φ - przesunięcie fazowe (godz).

(b) temperatury czynnika wewnątrz budowli

Czynnik ten musi być rozpatrywany indywidualnie. Przykładowo dla hiperboloidalnej chłodni mokrej o ciągu naturalnym, można przyjąć, że temperatura czynnika wewnątrz konstrukcji (t_i), (mieszanka podgrzanego powietrza, pary wodnej i aerozolu) jest związana z temperaturą powietrza atmosferycznego (t_e) wg zależności [3]:

$$t_i = 0.71t_e + 18.7 \quad (3)$$

Dla chłodni nieczynnej przyjmuje się $t_i = t_e$.

(c) natężenia promieniowania słonecznego (I)

$$I_G = I_{DIR} \cos(I, \vec{n}) + I_{DIFF} \quad (4)$$

gdzie: $\cos(I, \vec{n})$ - cosinus kąta pomiędzy normalną do powierzchni powłoki a kierunkiem promieni słonecznych, I_{DIR}, I_{DIFF} - natężenie promieniowania słonecznego bezpośredniego i rozproszonego

$$I_{DIR} = I_{SC} / (1 + \tau_s m_r), \quad I_{DIFF} = 0.38 \tau_s I_{DIR} \quad (5)$$

gdzie: I_{SC} - stała słoneczna, τ_s - współczynnik przezroczystości atmosfery, m_r - względna masa optyczna powietrza atmosferycznego, $m_r = 1/\sin\beta$, β - wysokość horyzontalna słońca (rad) [4].

Wpływ energii promieniowania słonecznego na pole temperatury w powłoce reprezentuje tzw. zastępcza temperatura powietrza atmosferycznego t_s ($^{\circ}\text{C}$):

$$t_s = t_e + I_G A / \alpha \quad (6)$$

gdzie: t_e - temperatura powietrza atmosferycznego ($^{\circ}\text{C}$), A - współczynnik absorpcji promieniowania słonecznego przez powierzchnię powłoki, α - zastępczy współczynnik wymiany ciepła pomiędzy powierzchnią powłoki a otoczeniem, uwzględniający konwekcję i promieniowanie krótkofalowe ($\text{W}/\text{m}^2\text{K}$).

4. Rozwiązanie własne

Biorąc pod uwagę wymienione w p.1 ograniczenia stosowalności wzorów ujętych w normie [1], do dalszych rozważań przyjęto jednowymiarowy, nieustalony model przepływu ciepła (jak dla powłok cienkich) oraz zastosowano przybliżenie w postaci równania różnicowego [5]:

$$\frac{\Delta \vartheta}{\Delta \tau} = \frac{\lambda}{\rho C_p} \frac{\Delta^2 \vartheta}{\Delta x^2} \quad (7)$$

Współrzędna x jest prostopadła do powierzchni środkowej powłoki w rozpatrywanym przekroju.

Przyjmując warunki brzegowe III rodzaju uzyskuje się rozwiązanie w postaci [5]

(a) dla węzłów wewnętrznych ($i = 2, 3, \dots, n$):

$$\vartheta_{i,\tau+\Delta\tau} = \frac{\lambda\Delta\tau}{\rho C_p(\Delta x)^2} \left[\vartheta_{i-1,\tau} + \vartheta_{i+1,\tau} + \vartheta_{i,\tau} \left(\frac{\rho C_p(\Delta x)^2}{\lambda\Delta\tau} - 2 \right) \right] \quad (8)$$

(b) dla węzłów brzegowych ($i = 1, n+1$):

$$\vartheta_{i,\tau+\Delta\tau} = \frac{2\lambda\Delta\tau}{\rho C_p(\Delta x)^2} \left[\vartheta_{2,\tau} + \frac{\alpha\Delta x}{\lambda} t_s + \vartheta_{1,\tau} \left(\frac{\rho C_p(\Delta x)^2}{2\lambda\Delta\tau} - \frac{\alpha\Delta x}{\lambda} - 1 \right) \right] \quad (9)$$

gdzie:

ϑ - temperatura, τ - czas,

$\Delta\tau, \Delta x$ - krok czasowy i przestrzenny,

λ, ρ, C_p - współczynnik przewodności cieplnej, gęstość i ciepło właściwe materiału powłoki,

α - współczynnik wymiany ciepła drogą konwekcji pomiędzy powierzchnią powłoki a otoczeniem,

t_s - zastępcza temperatura powietrza (jak we wzorze (6)).

Rozwiązanie jest stabilne przy zachowaniu następujących warunków:

$$\left[\frac{\lambda\Delta\tau}{C_p(\Delta x)^2} \leq \frac{1}{2} \right] \wedge \left[\frac{\lambda\Delta\tau}{\rho C_p(\Delta x)^2} \leq \left(1 + \frac{\alpha\Delta x}{\lambda} \right) / 2 \right] \quad (10)$$

Jako warunek początkowy przyjęto jednakową temperaturę całej powłoki (w zamieszczonym w p.6 przykładzie liczbowym wartość temperatury początkowej jest równa średniej dobowej temperaturze powietrza atmosferycznego dla okresu letniego).

5. Oprogramowanie

Na podstawie założeń przedstawionych w p. 2,3 i 4 opracowano pakiet programów do wyznaczania obciążenia temperaturą konstrukcji o kształcie powłoki obrotowej, poddanych wpływom klimatycznym i eksploatacyjnym. Wersje źródłowe programów

zapisano w rozszerzonym języku Fortran77 [6], zawierającego elementy języka Fortran90. Pakiet zawiera dwa programy.

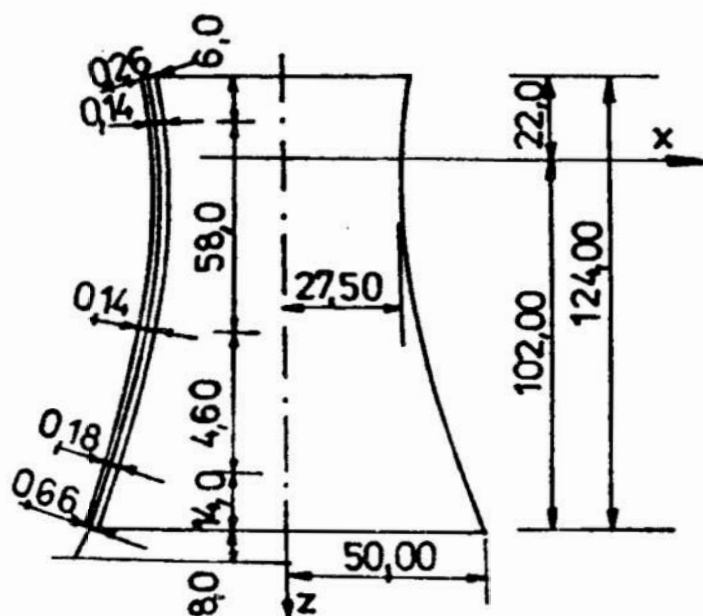
Pierwszy program przeznaczony jest do wyznaczania rozkładu temperatury w wybranych przekrojach na grubości powłoki dla określonych warunków klimatycznych i eksploatacyjnych, dla wybranych pór roku i określonych przedziałów czasu. Wyniki obliczeń przechowywane są w pamięci zewnętrznej w zbiorach binarnych o dostępie bezpośrednim.

Drugi program przeznaczony jest do analizy pola temperatury "wygenerowanego" przez program pierwszy pod kątem wyszukania ekstremalnych obciążeń temperaturą (maksymalne przyrosty temperatury względem temperatury odniesienia oraz ekstremalne różnice temperatury na grubości powłoki). Program ten umożliwia również graficzną prezentację wyników obliczeń. W tej części użyto procedur biblioteki graficznej języka Microsoft C, dostępnych z poziomu języka Microsoft Fortran v.5.0. [6].

Opisane programy pracują pod systemem MS DOS od wersji 2.1 w górę, na mikrokomputerach rodziny IBM PC/XT/AT.

6. Przykład liczbowy

Opisane w p.5 programy użyto do wyznaczenia obciążenia temperaturą powłoki żelbetowej chłodni hiperboloidalnej o wys. 132 m (Rys.1).



Lokalizacja: 18°30' E, 50°00'

Materiał powłoki: beton zwykły, $\rho = 2500 \text{ kg/m}^3$,

$C_p = 0.8 \text{ kJ/kgK}$,

$\lambda = 1.7 \text{ W/mK}$.

Temperatura odniesienia:

$\vartheta_0 = 10^\circ\text{C}$.

Dane klimatyczne:

$t_{e, \text{sr}} = +27^\circ\text{C}$ (lato), $a = 8^\circ\text{C}$,

$\varphi = 9$ godzin, $I_{\text{SC}} = 1256 \text{ W/m}^2$

$\tau_s = 0.31$.

Pozostałe dane:

$A = 0.65$ (jasny beton),

$\alpha = 20 \text{ W/m}^2\text{K}$.

Rys.1 Wymiary analizowanej chłodni kominowej

Fig.1. Geometry of the analysed cooling tower

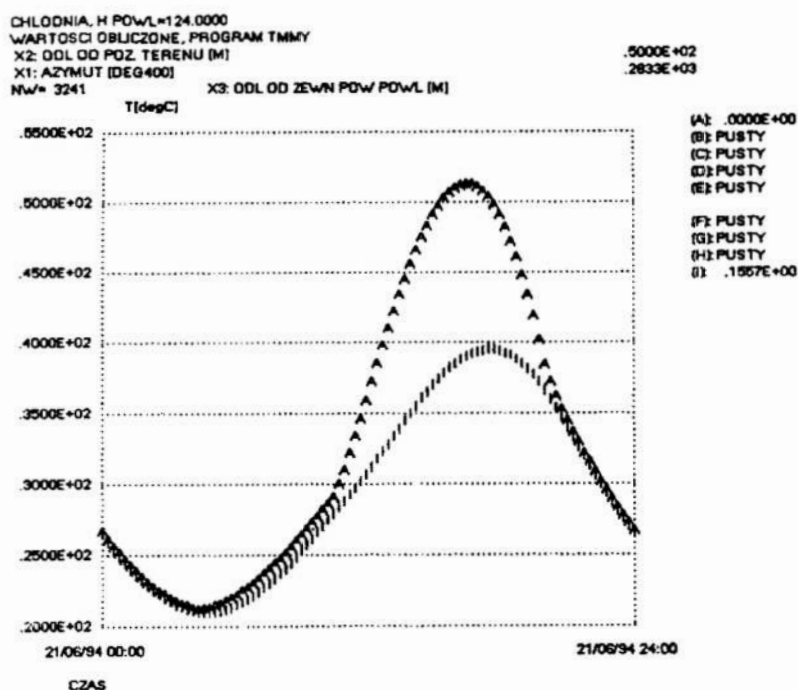
Rozkład temperatury w powłoce wyznaczono w pierwszym przybliżeniu w przekrojach wyznaczonych przez punkty przecięcia się 24 południków i 18 równoleżników, w każ-

dym przekroju obliczono wartości temperatury w 9 punktach. Obliczenia przeprowadzono dla okresu letniego, dla przypadku chłodni poddanej jedynie wpływow klimatycznym. Wyjściową temperaturę powłoki przyjęto równą średniej dobowej i powtarzano obliczenia dla doby 21/06/YY aż do spełnienia w każdym punkcie założonego kryterium zbieżności:

$$\left| \vartheta_{i,\tau} - \vartheta_{i,\tau+24h} \right| \leq \varepsilon, \quad \varepsilon = 0.1^{\circ}C.$$

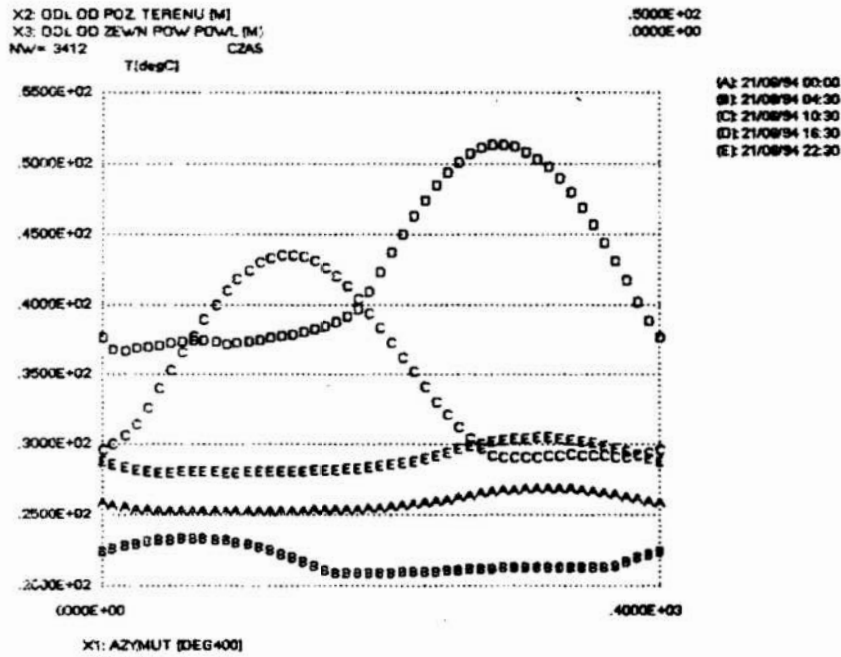
Tak otrzymany zbiór wartości temperatur poddano analizie wyznaczając współrzędne przestrzenne i czasowe wystąpienia ekstremalnego obciążenia temperaturą wg. kryteriów przyjętych w p.2. Otrzymano maksymalny przyrost temperatury powłoki względem temperatury odniesienia równy $41.3^{\circ}C$, dla wartości czasu 21/06 16:30 (poz. +50 m, azymut 283.3 grad, zewnętrzna powierzchnia powłoki). Z kolei maksymalna różnica temperatury na grubości powłoki wyniosła $21.6^{\circ}C$ i wystąpiła 21/06 16:15 w przekroju przy dolnej krawędzi powłoki (poz. +8.0 m, azymut 283.3 grad).

Na rys. 2÷ 6 przedstawiono wykresy otrzymane za pomocą programu analizującego wyniki obliczeń, reprezentujące ekstremalne wartości obciążenia powłoki temperaturą.



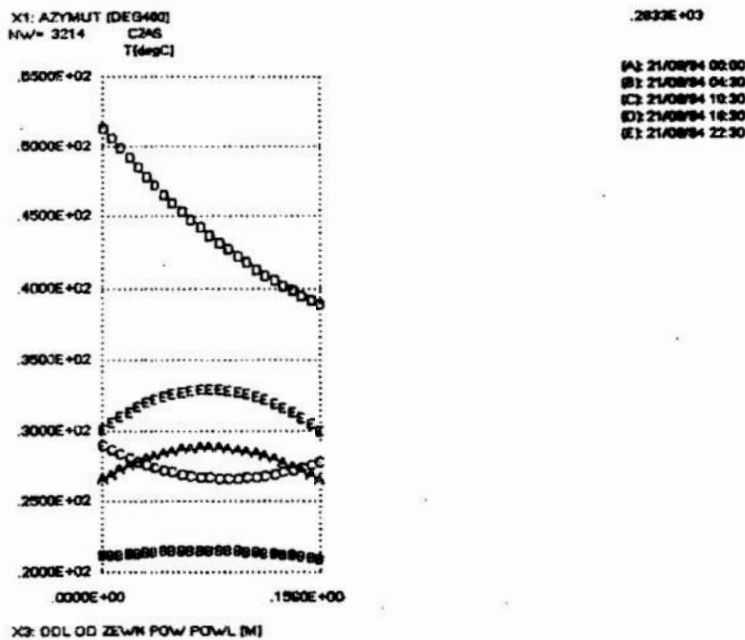
Rys.2. Dobowe zmiany temperatury w dwóch punktach przekroju powłoki, w którym wystąpiła ekstremalna różnica temperatury na jej grubości (A-powierzchnia zewnętrzna, B-powierzchnia wewnętrzna)

Fig.2. Diurnal variations of the temperature at two points of the shell cross-section - the extreme difference of temperature across the shell thickness (A-external surface, B-internal surface)



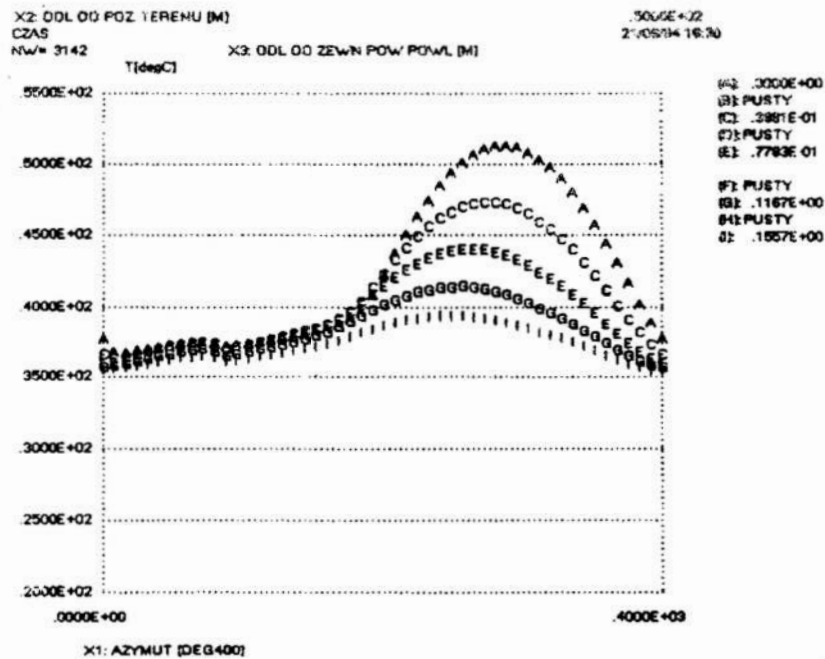
Rys.3. Temperatura na zewnętrznej powierzchni powłoki w poz. +50 m, dla wybranych wartości czasu (krzywa "D"-przypadek ekstremalnego przyrostu temperatury względem temperatury odniesienia, 21/06/XX, 16:30)

Fig.3. The temperature of the external surface of the shell (level +50 m) along its circumference representing selected time values ("D"-21/06/XX, 16:30, the case of extreme temperature increase with respect to the reference temperature)



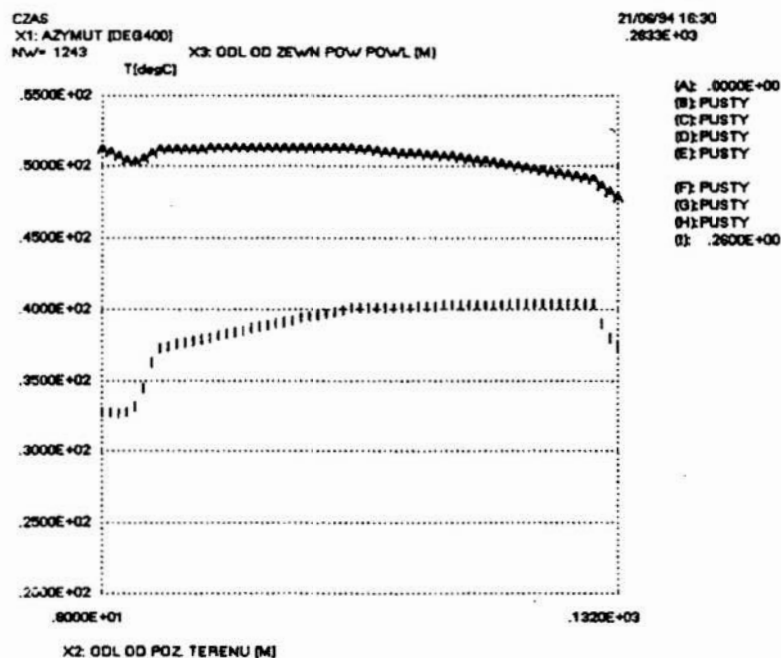
Rys.4. Rozkład temperatury na grubości powłoki dla różnych wartości czasu w przekrojach, w którym wystąpił maksymalny przyrost temperatury względem temperatury odniesienia ("D": 21/06/XX, 16:30)

Fig.4. The temperature distribution across the shell thickness representing selected time values ("D": 21/06/XX, 16:30, the case of extreme temperature increase with respect to the reference temperature)



Rys. 5. Rozkład temperatury na obwodzie powłoki, w pięciu warstwach jej grubości w poz. +50 m, data: 21/06/XX, 16:30 (przypadek maksymalnego przyrostu temperatury względem temperatury odniesienia)

Fig. 5. The temperature distribution in the five layers of the shell thickness, along its circumference (level +50 m. date: 21/06/XX, 16:30, the case of extreme temperature increase with respect to the reference temperature)



Rys. 6. Przebieg temperatury w kierunku południkowym powłoki, na pow. zew. (A) i wew. (I), w południku, w którym wystąpił maksymalny przyrost temperatury względem temperatury odniesienia (azymut 283 grad).

Fig. 6. The temperature distribution in the meridional direction of the shell on the external (A) and internal (I) surface. The case of extreme increase of the temperature with respect to the reference temperature (azimuth 283 grad)

7. Podsumowanie

Przedstawione oprogramowanie umożliwia wyznaczenie obciążenia temperaturą konstrukcji o kształcie w postaci dowolnej cienkiej powłoki obrotowej (zbiorniki walcowe, chłodnie kominowe, kominy wieloprzewodowe) poddanej wpływom klimatycznym i eksploatacyjnym, zgodnie z wymaganiami norm ISO i PN. Możliwe są obliczenia dla dowolnej lokalizacji, przy uwzględnieniu lokalnych warunków klimatycznych. Modułowa struktura wewnętrzna programów umożliwia łatwą ich modyfikację, np. w zakresie przyjętego modelu rozkładu temperatury czy też wpływu czynników klimatycznych. Moduł graficzny prezentowanego pakietu umożliwia przedstawienie wyników obliczeń w postaci wykresów rozkładu temperatury w powłoce dla dowolnej kombinacji trzech współrzędnych przestrzennych i współrzędnej czasowej.

Literatura

1. Polska Norma PN-86/B-02015. Obciążenia zmienne środowiskowe. Obciążenie temperaturą, 1987.
2. ISO/TR 9492. Bases du calcul des constructions-Actions climatiques de la temperature, 1987.
3. Jahrbuch Bautechnik im Kraftwerksbau, 2 Ausgabe, 1982/83, Vulkan - Verlag, Essen.
4. Paltridge G.W., Platt C.M.R.: Radiative Processes in Meteorology and Climatology, Elsevier, 1976.
5. Adams J.A., Rogers D.F.: Computer-Aided Heat Transfer Analysis, Mc Graw-Hill, 1973.
6. Microsoft Fortran v.5.0, Advanced Topics, Microsoft Corporation, 1989.

Streszczenie

W niniejszej pracy przedstawiono algorytm wyznaczania obciążenia temperaturą konstrukcji o kształcie cienkiej powłoki obrotowej. Uwzględniono dane i zalecenia zawarte w obowiązującej normie dotyczącej obciążenia temperaturą. Opisano pakiet programów własnych do wyznaczania obciążenia temperaturą konstrukcji powłokowych poddanych wpływom klimatycznym i eksploatacyjnym. Pokazano wyniki obliczeń dla powłoki hiperboloidalnej chłodni kominowej.

COMPUTER-AIDED SIMULATION OF THE TEMPERATURE LOAD OF THE CIRCULAR SHELL STRUCTURES

Summary

The algorithm for determination of the temperature load of the circular thin shell structures is presented. The recommendations included in domestic and ISO standards are taken into account. The package of own computer programmes for determination of the temperature load of circular shell structures subjected to climatic and operational actions is described. The numerical results for hyperbolic cooling tower shell are included.

ALGORITHM AND COMPUTER PROGRAM FOR SHAPE OPTIMIZATION OF NOTCHES IN 2-D STRUCTURAL ELEMENTS

1. Introduction

An extensive literature has developed on optimization of structures and structural elements which are defined by cross-section and thickness variables (size optimization). A more important problem, from the point of view of mechanical design, is determination of the shape of 2-D, or 3-D structural elements (shape optimization). For such problems, the shape of the structural element must be treated as design variables.

As it is already known, the presence of holes, fillets, undercuts, cut-outs etc. (known collectively as "notches") in a structural element results in increasing the local tensions which are called stress concentrations. One of the principal features of stress concentration is so called Stress Concentration Factor (SCF) [7]. To decrease a SCF, a possibility exists to minimize stress by changing the shape of a notch. Such a class of optimization problems is referred to as the shape notch optimization.

One of the first treatment (involving finite element method) of the general problem of selection shape of the structure as the design variable is presented by Zienkiewicz and Campbell [21]. Since then there have been many contributions in which mostly 2-D or 3-D finite or boundary element methods have been coupled with numerical optimization techniques [2,5,6,8-16,18-20]. Because of the lack of space, a review of literature in this field is omitted in this paper.

This paper briefly presents an algorithm and computer program NOTCH2D for the shape optimization of the notches in 2-D structural elements (Fig.1), with the object

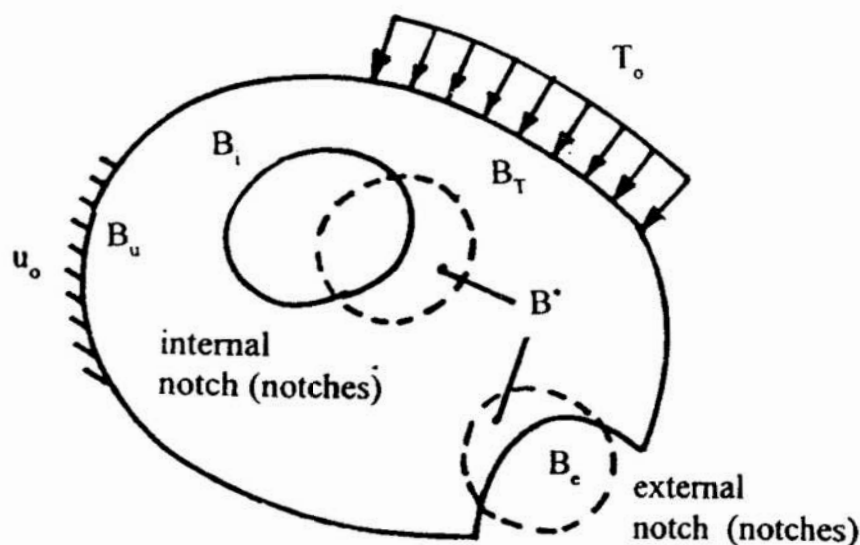


Fig. 1 2-D body with varying shape of internal B_i and external B_e notches, B_T - the boundary with given tractions, B_u - the boundary with given displacements, B^* - variation domain

minimizing stress concentrations. The computer program discussed belongs to the special software tools like OPTIM2 [12], which has been developed for minimum weight design of steel structures, in contrast to commercial, general purpose programs like, for example, ADS with using MSC/NASTRAN [19-20], SHAPE [1], CAOS [9-10] or ODESSY [9-10].

2. Algorithm for shape optimization of notches

Problem definition

A shape optimization problem of notches can be stated mathematically as one of minimizing a maximum effective stress σ_e , i.e. $\min [\max \sigma_e]$ for a given load. This original min-max problem can be converted to a simple min problem in terms of an unknown bound on the stresses [8,9,14-16,18], and can be written as

$$\min \sigma_{\text{emax}}$$

$$\text{subject to the constraints } \sigma_{e_j}(\mathbf{D}) - \sigma_{\text{emax}} \leq 0, \quad j = 1, \dots, l, \quad \mathbf{D}_l \leq \mathbf{D} \leq \mathbf{D}_u$$

where l is the number of critical points, \mathbf{D} is a vector of design variables, \mathbf{D}_l and \mathbf{D}_u are the lower and upper limits on \mathbf{D} respectively.

Geometric definition

We describe the variable boundary of a notches using one segmented Bezier's curves with the coordinates of some control nodes (key points) treated as design variables. It is convenient to give prescribed move directions to the key points.

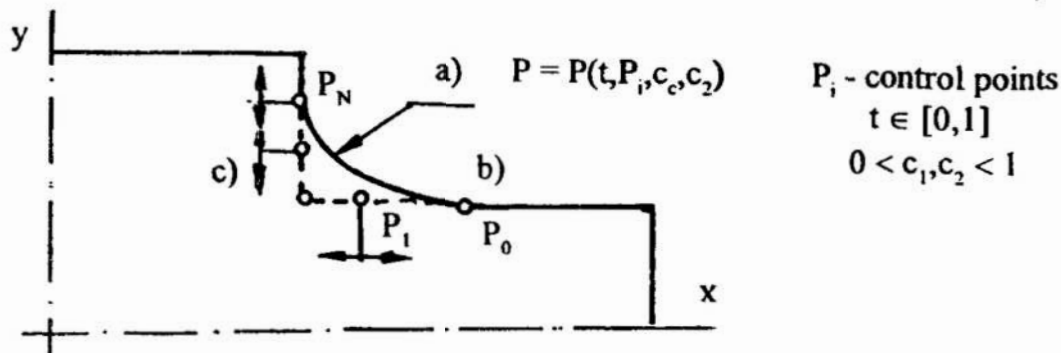


Fig. 2. Example of 2-D body (fillet problem): a) One segmented Bezier curve, b) \circ control points, c) \rightarrow movement directions

FSM method as a basis for optimization

The Fictitious Stress Method (FSM) presented in [3] is used for an analysis of the stress distribution in 2-D structural elements. This is an indirect boundary element method. Fig. 3 shows a cavity (very long in z direction) in an infinite elastic body. The

boundary of the cavity is labeled C in Fig. 3a. The dashed curve C' shown in Fig. 3b has the same shape as the curve C . Both curves are approximated by straight line segments (elements), joint end to end. The difference between curves is, that curve C' represents the location of line segments in an infinite body (without cavity), which are coincident with the real boundary C . The shear P_s^j and normal P_n^j stresses applied to the segment j induce the actual stresses σ_s^i and σ_n^i at the midpoints of each element of curve C' , $i=1$ to n . The stresses P_s^j and P_n^j are fictitious quantities and should be determined. The relation between the actual stresses σ_s^i and σ_n^i and the fictitious stresses P_s^j and P_n^j are based on the analytical singular solution to the problem of constant normal and shear stresses applied to an arbitrarily oriented, finite line segment in an infinite body (Kelvin solution), what leads directly to the system $2n \times 2n$ equations $C \cdot P = b$: C - the influence coefficient matrix, P - unknown fictitious stress components, b - given tractions. If the fictitious stresses are known, we can find the tangential stresses $\sigma_t = A_{ts} \cdot P_s + A_{tn} \cdot P_n$, where: A_{ts} and A_{tn} - the influence coefficients for tangential stresses, P_s and P_n are the shear and normal fictitious stress components, respectively.

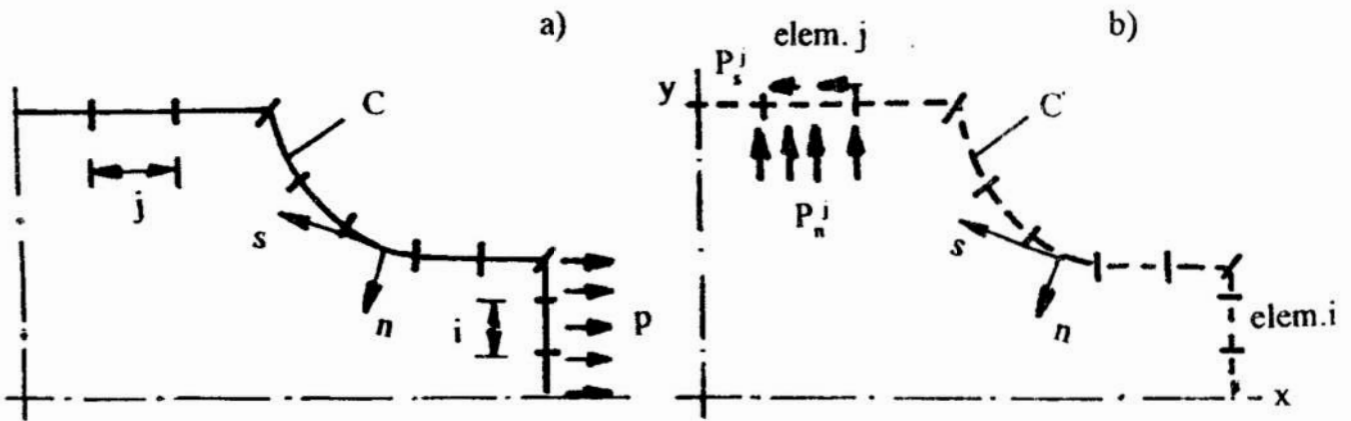


Fig. 3. Example of 2-D body (fillet): a) Real contour, b) Numerical model

Because of constant stress elements are assumed to be on the boundary there is no need for numerical integration. The accuracy of results can be increased by increasing the number of elements. The accuracy of the FSM for solution of stress concentration problem in 2-D machine components have been examined in [17]. It was found that there is no remarkable difference between the results of the FSM notch stress analysis with the analytical, and numerical results widely presented in literature.

Sensitivity analysis

Having completed the FSM notch stress analysis, we now evaluate the sensitivity of the stresses of the current design to small changes in the design variables. It should be mentioned that the objective function is linear, and only there is a need to evaluate the gradients and of the behavioural constraints with respect to design variables. Methods for evaluating the sensitivities may be purely analytical, semi-analytical or can be based on finite differences [4]. The presented algorithm is rather a very small scale problem and for this reason the forward finite difference method (with step equal 10^{-4}), which is the most straightforward and conceptually simplest way of obtaining the gradients of constraints, is used.

Optimization algorithm

Because objective function is linear, in this paper a sequential linear programming (SLP) has been used to obtain the optimal solution [8-10,12,14-16,18,21]. The behaviour constraints are sequentially linearized and the optimum solution is obtained by using linear programming (LP) which can be efficiently solved by using the Simplex algorithm. A set of so called 'move limits' are added to the constraints of the SLP problem in order to control a stability and convergence of the algorithm. These move limits may be specified as $\Delta \mathbf{D}_l \leq \Delta \mathbf{D} \leq \Delta \mathbf{D}_p$ where $\Delta \mathbf{D}_l$ and $\Delta \mathbf{D}_p$ are the lower and the upper limit on $\Delta \mathbf{D}$ respectively. These move limits are reduced during the optimization process so that a solution is found. The optimization procedure is considered to have converged to the final minimum stress concentration when the condition $|\sigma_i^{\max} - \sigma_{i-1}^{\max}| / \sigma_{i-1}^{\max} \leq \epsilon$ is satisfied for the two successive iterations, where σ_i and σ_{i-1} are the effective stresses after i and $i-1$ iterations, and ϵ is a user defined tolerance value. In the computer program a tolerance $\epsilon = 0.0001$ is assumed.

3. Shape optimization program

The shape optimization program NOTCH2D is a modularly organized package, consisting of several blocks. The following is a brief description of the package.

The INPUT block contains two subroutines which call the data for stress analysis and optimization procedures. The input data are as follows: - number of macroelements (each straight part of the boundary, and each one segmented Bezier curve is treated as a macroelement), - number of boundary elements in each macroelement, - material data (Young modulus E , Poisson ratio ν), - global dimensions of structural element, - boundary conditions, - number of design variables, - initial value of design variables, - lower and upper values on design variables, - lower and upper values on design changes ("move limits"). Some of parameters are assumed to be constant in the program, but they could be easily changed by the user. Using informations about macroelements and global dimensions of the structural element, the coordinates of endpoints of boundary elements are created automatically in the special subroutine written by the user.

The FSM block consists of subroutines, which define boundary element locations, set up the system of algebraic equations and compute the unspecified stresses (and displacements, if necessary) at the boundary elements. The above block is used to compute the gradients of stresses needed in optimization procedures.

The SLP block is a group of subroutines which by means of the simplex algorithm solves the current linear programming problem.

The OUTPUT block prints the results and presents grafically the optimized profile of the notch and the stress distribution. Because of the lack of space, the details of program are not given.

4. Numerical examples

The performance of the computer program has been tested with several examples: optimal shape of the single hole in infinite plate for various kind of loading [14,18], stress minimization around quasi-ovaloid hole in infinite plate [15], shape optimization

of interacting holes in infinite plate with respect to stress [14,18]. In this paper further examples are presented.

Example 1. Optimum shape of internal notches (holes) in finite strip under uniaxial tension.

One method for reducing the static stress concentration [7] is by introducing additional (defense) holes in the direction of loading on either side of the central hole [11]. Further reduction of the stresses is possible with simultaneous shape optimization of the both kind of holes. The problem is symmetric about both coordinate axes, so only the quarter of the strip needs to be modelled (Fig. 4). The boundary of the central hole was modelled using 21 boundary elements, and the boundary of the additional hole

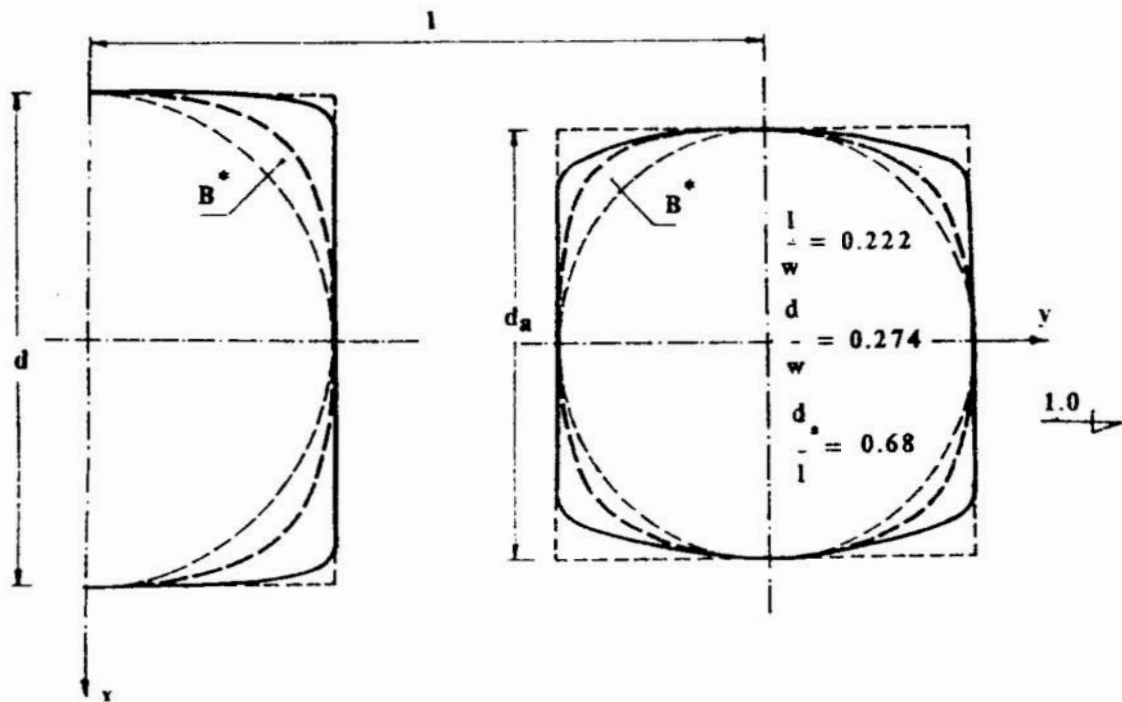


Fig. 4. Optimal profiles of central and defense holes in finite strip under uniaxial tension

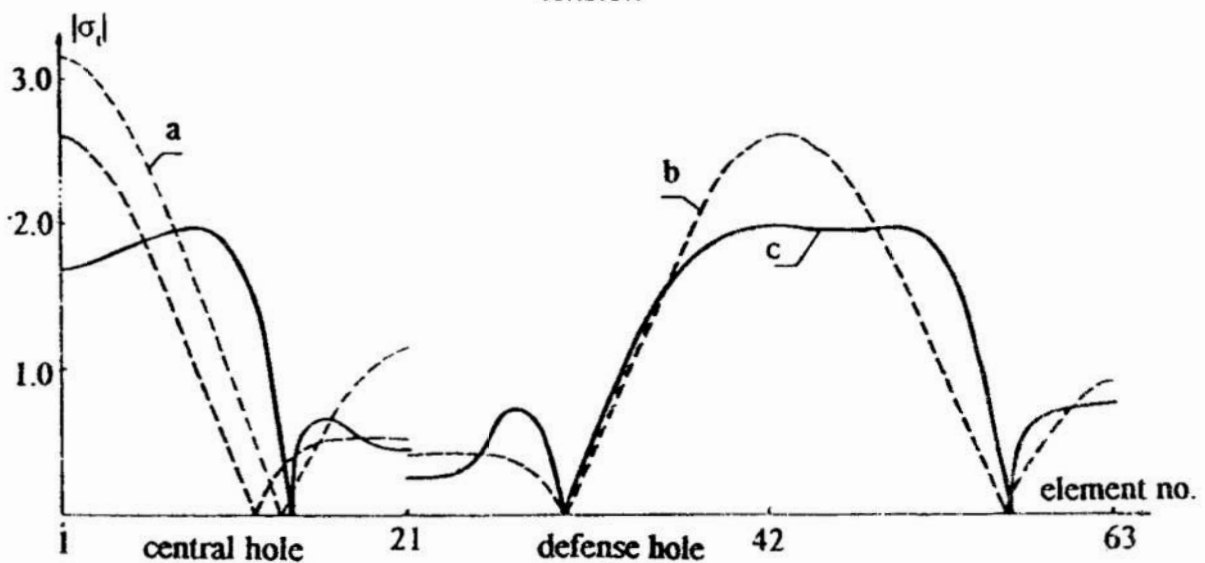


Fig. 5. Stress distribution around holes: original and defense in finite strip under uniaxial tension

using 42 boundary elements. The exterior boundary of the strip is divided on 60 boundary elements. The curvilinear parts of the holes are described by special concept of Bezier curves [18]. Figure 4 shows the optimal design of the central end defense holes, and Figure 5 shows the effective stress distribution along the design boundaries for: a) central (single) circular hole, b) starting profiles, and c) optimal profiles. The 37% reduction in stresses is obtained by the presented numerical approach (30% reduction in stresses by photoelastic method [11]).

Example 2. Optimum shape of external notches in finite strip under uniaxial tension.

In contrast to the Example 1 the external (central and defense) notches problem in a finite strip is considered. Figure 6 shows the dimensions of the strip and the optimal contours of central and defense notches, and Figure 7 shows the effective stress distribution along the optimum shape of notches. The about 30% reduction in effective stresses with comparison to a circular, single notch is obtained.

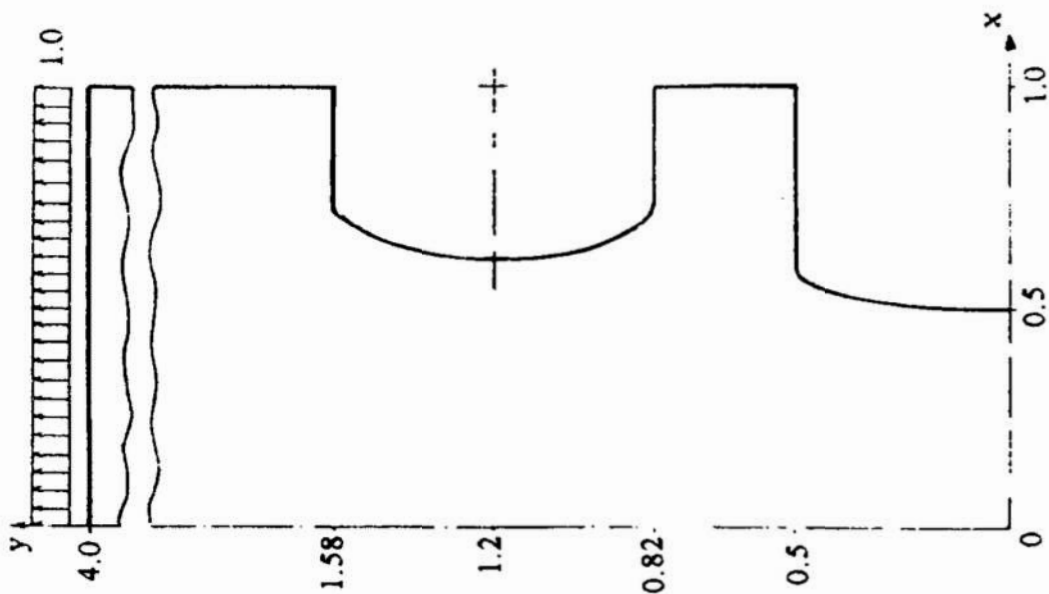


Fig. 6. Optimized geometries of central and defense external notches in finite strip subjected to uniaxial loading

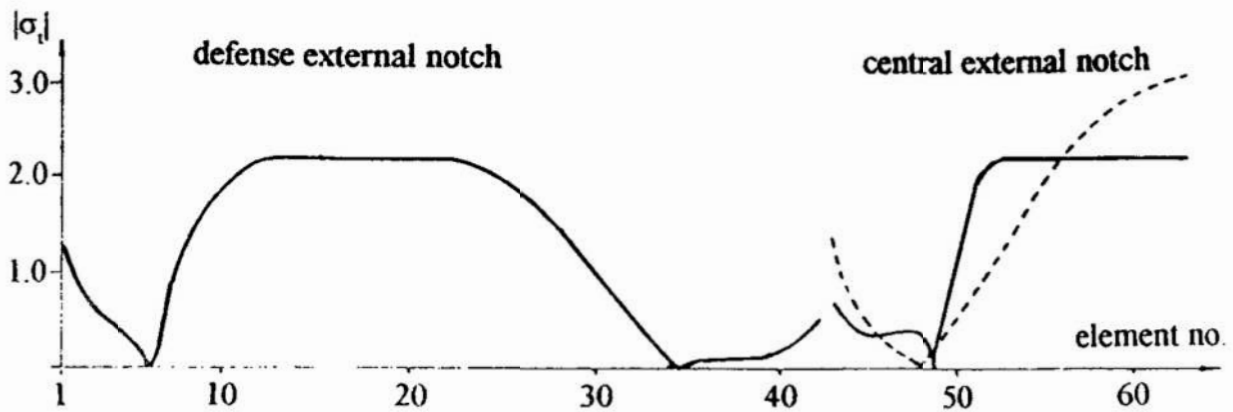


Fig. 7. Stress distribution along optimum shape of notches (—) and for a single circular notch (---).

5. Conclusions

Shape optimization of notches in 2-D structural components to minimize stress concentrations is formulated as the sequential linear programming problem with using of the Fictitious Stress method. The Fictitious Stress Method is very suitable for shape optimization problems, and in comparison with the finite element method needs much less data, and gives more accurate stress solution. A geometry of the notch is defined by Bezier technique. A significant reduction in stresses is obtained in comparison with a traditionally used shapes of the notches. With such reduction in maximum stress level, the improvement in fatigue life of the component with the notch (notches) can be very significant.

Presented algorithm is a small scale shape optimization problem. Practically few design variables are necessary to control the shape of single, or multiple notches, which means that these and similar problems can be easily solved using IBM PC. The computer program NOTCH2D is efficient and can be helpful for a structural designer in selection of the "best" distribution of stresses of notches in structural components.

References

1. Atrek E.: SHAPE: *A program for shape optimization of continuum structures*, in Computer Aided Optimum Design of Structures: Applications, Eds. Brebbia C.A., Hernandez S., Proc. First Int. Conf. Southampton, June 1989, CMP Southampton, 135-144, 1989
2. Burczyński T.: *The boundary element method for selected analysis and optimization problems of deformable systems*, Scientific Reports, (in Polish), Silesian Technical University, Z. 97, 1989
3. Crouch S.L., Starfield A.M.: *Boundary element methods in solids mechanics*, George Allen and Unwin, London, 1983
4. Dems K., Mróz Z.: *On shape sensitivity approaches in the numerical analysis of structures*, Struct. Optim., Vol. 6, No. 2, 86- 93, 1993
5. Kamiya N., Kita E.: *Local shape optimization of a two-dimensional elastic body*, Fin. Elem. Anal. Design, Vol. 6, 207-216, 1990
6. Moghaddasi-Tafreshi A. Fenner R.T.: *Design optimization using boundary element method*, Jnl Strain Anal., Vol. 26, 231-241, 1991
7. Olesiak Z.: *Stress concentration. Contact stresses*, In Strength of Structural Elements, Ed. Życzkowski M., PWN, Warszawa, Elsevier, Amsterdam, 527-612, 1991
8. Pedersen P., Carsten L.L.: *Design for minimum stress concentration by finite elements and linear programming*, J. Struct. Mech., Vol. 10, 375-391, 1982/83
9. Rasmussen J., Olhoff N., Lund E.: *Foundations for optimum system development*, Advance TEMPUS Course on Numerical Methods in Computer Aided Optimal Design, Ed. T. Burczyński, May 1992, Zakopane, Vol. 1, 168-191, 1992
10. Rasmussen J., Olhoff N., Lund E.: *Extracts from Collection of Examples CAOS Optimization System, 3rd Edition*, Advance TEMPUS Course on Numerical Methods in Computer Aided Optimal Design, Ed. T. Burczyński, May 1992, Zakopane, Vol. 1, 192-240, 1992
11. Steichen W.P.: *Experimental and computer-aided investigations and optimization of stress-relieving notches*, Jnl Strain Anal., Vol. 13, No. 3, 149-156, 1978

12. Stojanowski P., Aleksowski G.: *OPTIM2 - an efficient SLP optimizer of linear steel structures*, in Computer Aided Optimum Design of Structures: Applications, Eds. Brebbia C.A., Hernandez S., Proc. First Int. Conf. Southampton, June 1989, CMP Southampton, 69-77, 1989
13. Xu Ch., Yy M.: *Shape optimization of structures to minimize stress concentration*, Comput. Struct., Vol. 36, 491-49, 1990
14. Wilczyński B.: *Stress minimization around holes in uniaxial loaded plates*, Proc. X Conf. Comp. Meth. in Mechanics, May 1991, Świnoujście, Vol. 2, 741-748, 1991
15. Wilczyński B.: *Minimizing stress concentration around quasi-ovaloid hole in infinite plate*, Scientific Reports, (in Polish), Silesian Technical University, Z.113, 423-428, 1993
16. Wilczyński B.: *Stress minimization using Fictitious Stress Method*, II Int. Seminar on Boundary Element Techniques and Singularity Methods in Engineering, Wrocław, December 1992, Scientific Reports, Technical University of Wrocław, No. 74, 165-176, 1993
17. Wilczyński B.: *Accuracy of Fictitious Stress Method for stress concentration problems*, Scientific Reports, Technical University at Koszalin, No 15, 192-210, 1993
18. Wilczyński B.: *Shape optimization of notches in 2-D machine components*, Scientific Reports, Technical University at Koszalin No. 17, 61-70, 1994
19. Vanderplaats G.N.: *A general-purpose optimization program for engineering design*, Comp. Struct., Vol. 24, No. 1, 13-21, 1986
20. Vanderplaats G.N.: *Effective use of numerical optimization in structural design*, Fin. Elem. Anal. Design, Vol. 6, 97-112, 1989
21. Zienkiewicz O.C., Campbell J.S.: *Shape optimization and sequential linear programming*, In Optimum Structural Design, Wiley and Sons, 109-126, 1973

Summary

The shape optimal design of notches in 2-D elastic machine components is formulated using the Fictitious Stress Method. The design objective is to minimize a maximum effective stress for a given load. This formulation is based on constant stress boundary element. A special concept of segmented Bezier interpolants is adopted for defining geometry of the machine component, and the Sequential Linear Programming is used as the optimization procedure. The computer program NOTCH2D is briefly described.

ALGORYTM I PROGRAM KOMPUTEROWY OPTIMALIZACJI KSZTAŁTU KARBÓW W DWUWYMIAROWYCH ELEMENTACH KONSTRUKCJI

Streszczenie

W artykule rozpatruje się zagadnienie poszukiwania optymalnego kształtu brzegu karbu w dwuwymiarowych elementach konstrukcji. Celem jest minimalizacja naprężeń w otoczeniu karbu. Do wyznaczania naprężeń wykorzystuje się pośredni wariant metody elementów brzegowych, metodę naprężeń fikcyjnych. Modelowany brzeg karbu opisany jest elementami brzegowymi o stałej wartości naprężenia. Do opisu kształtu brzegu wykorzystuje się specjalny wariant krzywych Beziera. Zadanie optymalizacji sprowadza się do sekwencyjnego programowania liniowego. Opisane są podstawowe elementy programu komputerowego NOTCH2D.