

## **PRZYKŁADOWE TEMATY ZADAŃ PROJEKTOWYCH Z PRZEDMIOTU SZTUCZNA INTELIGENCJA DLA KSS**

1. Rozwiązać zadanie zadania załadunku (plecakowego) z ograniczeniami na dopuszczalne wymiary oraz ciężar [3]:
  - a) algorytmem symulowanego wyżarzania.
  - b) algorytmem genetycznym
2. Rozwiązać zadanie zbierania plonów oraz zadanie pchania wózka [3]:
  - a) algorytmem symulowanego wyżarzania.
  - b) algorytmem genetycznym
3. Rozwiązać problem komiwojażera:
  - a) algorytmem symulowanego wyżarzania.
  - b) algorytmem genetycznym

W rozwiązywanym problemie przyjąć, iż nie wszystkie połączenia między dowolnymi miastami są dopuszczalne. Zastosować odpowiednią reprezentację rozwiązania. Przyjąć funkcję kryterialną złożoną z kilku wskaźników np. droga, czas, opłaty za przejazd itp.
4. Zrealizować algorytm genetyczny dobierający parametrów regulatora PID dla wybranego obiektu opisanego w dziedzinie czasu ciągłego. Jako kryteria poszukiwań przyjąć odpowiednie wskaźniki charakterystyk czasowych lub częstotliwościowych. Dla uzyskanych najlepszych rozwiązań przeprowadzić symulacje z uzyskanymi w sposób genetyczny regulatorami. Przeprowadzić analizę porównawczą zaprojektowanego rozmytego regulatora z klasycznym (tzn. zaprojektowanym na podstawie np. metod Zieglera-Nicholsa).
5. Skonstruować mini-pakiet algorytmów genetycznych lub ewolucyjnych, w którym możliwe będzie rozwiązywanie dowolnego zadania optymalizacji parametrycznej.
6. Skonstruować algorytm genetyczny do planowania drogi w środowisku ruchomego robota [2, 3]. Program powinien wczytywać mapy (przykładowe środowiska z przeszkodami, w którym miałyby się poruszać się robot) dwuwymiarowe oraz punkty startowe i końcowe trasy. Drogę robota przykładowo budować z kawałków linii prostych. Ponadto należy zaprezentować na bieżąco najlepsze znalezione rozwiązania.
7. Wykorzystać algorytm HGA (ang. *Hierarchical Genetic Algorithm*) do identyfikacji modeli obiektów opisanych w dziedzinie czasu ciągłego. Program powinien na podstawie danych pomiarowych (np. charakterystyk częstotliwościowych) generować odpowiedni model obiektu. Zaproponować odpowiednią strukturę osobników oraz kryterium, według którego algorytm znajdzie najlepszy model opisujący rozważany proces.
8. Zrealizować układ sterowania w oparciu o logikę rozmytą dla jednego z następujących modeli obiektów. Wykorzystać pakiet narzędziowy „Fuzzy Logic Toolbox”

środowiska obliczeniowego MATLAB. Jako zmienne wejściowe takiego systemu przyjąć zmienne stanu danego obiektu.

a) model odwróconego wahadła

$$G_1(s) = \frac{-1}{\left(\frac{s}{6.25} - 1\right)\left(\frac{s}{6.25} + 1\right)}$$

b) model odwróconego wahadła wraz z serwomechanizmem

$$G_2(s) = \frac{\omega_n^2}{(s^2 + 2\zeta s + \omega_n^2)(1 - d^2 s^2)}$$

gdzie  $\omega_n = 20\pi \text{ rad/s}$ ,  $\zeta = 0.7$  oraz  $d = 0.16 \text{ rad/s}$ .

c) system dwóch zbiorników

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} -0.0197 & 0 \\ 0.0178 & -0.0129 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0.0263 \\ 0 \end{bmatrix} \cdot u$$

$$y = x_2$$

d) układ kulki i równoważni (*ball land beam*)

$$\frac{d^2 x}{dt^2} = mgr^2 \varphi / J$$

gdzie  $x$  jest położeniem kulki,  $g$  oznacza przyspieszenie ziemskie,  $r$  jest promieniem kulki,  $\varphi$  oznacza kąt nachylenia równoważni, zaś  $J$  jest momentem bezwładności kulki.

e) układ dwóch mas połączonych sprężyną

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -k/m_1 & k/m_1 & 0 & 0 \\ k/m_2 & -k/m_2 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 1/m_1 \\ 0 \end{bmatrix} \cdot u + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1/m_2 \end{bmatrix} \cdot w$$

$$y = x_2 + v$$

gdzie  $x_1$  i  $x_2$  są położeniem odpowiednio masy  $m_1$  i  $m_2$ ,  $x_3$  i  $x_4$  oznaczają prędkości odpowiednio masy  $m_1$  i  $m_2$ , sygnał  $u$  jest sterowaniem,  $y$  reprezentuje pomiar, natomiast sygnał  $w$  jest szumem systemowym, a  $v$  oznacza szum pomiarowy. Współczynnik  $k$  reprezentuje stałą sprężyny.

f) nieliniowego modelu odwróconego wahadła

$$\dot{x}_1(t) = x_2,$$

$$\dot{x}_2(t) = \frac{g(M + m) \sin x_1(t) - \frac{1}{2} mlx_2^2(t) \sin 2x_1(t) + bx_4(t) \cos x_1(t) - f_x(t) \cos x_1(t)}{l(M + m \sin^2 x_1(t))}$$

$$\dot{x}_3(t) = x_4(t)$$

$$\dot{x}_4(t) = \frac{mlx_2^2(t) \sin x_1(t) - mg \sin x_1(t) \cos x_1(t) - bx_4(t) + f_x(t)}{M + m \sin^2 x_1(t)}$$

gdzie  $\mathbf{x}(t) = [x_1(t) \ x_2(t) \ x_3(t) \ x_4(t)]$  jest wektorem stanu wahadła a poszczególne jego współrzędne oznaczają: kąt nachylenia ramienia od pionu, prędkość kątowna ramienia, położenie wózka i prędkość. Natomiast  $M$  jest masą wózka,  $m$  oznacza masę wahadła,  $l$  reprezentuje długość (z założenia – nieważkiego) ramienia wahadła,  $b$  jest współczynnikiem tarcia oraz  $f_x$  oznacza siłę przyłożoną do wózka.

9. Zastosować algorytm ewolucyjny do znalezienia optymalnej bazy reguł w sterowaniu rozmytymi regulatorami dla jednego z obiektów z punktu 9.
10. Skonstruować sztuczne sieci neuronowe [9] rozwiązujące jeden z następujących problemów:
  - a) rozpoznawania znaków alfabetu łacińskiego,
  - c) klasyfikacji obiektów na trzy rodzaje (np. figury geometryczne),
  - c) aproksymacji wielowymiarowych funkcji (Dodatek II) ,
  - d) aproksymacji podstawowych funkcji logicznych: AND, OR, NAND, NOR, NOT, XOR, NXOR, implikację itp.
  - e) prognozowania wartości akcji i towarów.
  - f) klasyfikacji:
    - komórek raka piersi,
    - kwiatów irysa,
    - gatunków wina,
    - itp.

dla których zbiory danych można pobrać ze strony: <http://archive.ics.uci.edu/ml/>
11. Skonstruować sztuczne sieci neuronowe [9] rozwiązujące jeden z problemów z punktu 11 wykorzystując do uczenia algorytm genetyczny lub algorytm HGA w przypadku optymalizacji strukturalnej i parametrycznej.
12. Zastosować algorytm genetyczny do uczenia sztucznej sieci neuronowej sterującej nieliniowym modelem odwróconego wahadła (punkt 1f). Przyjąć, że na wejście sieci podawany jest wektor stanu wahadła (kąt nachylenia ramienia od pionu, prędkość kątowna ramienia, położenie wózka i jego prędkość) na podstawie którego sieć generuje odpowiednia siłę działająca na wózek.
13. Wykorzystać algorytm ewolucyjny do uczenia sieci neuronowej, która steruje sondą kosmiczną lądującą na planecie np. Mars. W zadaniu tym należy tak dobierać siłę ciągu sondy by jej prędkość zderzenia z powierzchnia planety była bliska zeru. Przyjąć, że na wejście sieci podawana jest wysokość, prędkość sondy oraz jej masa,

zaś wyjście sieci generuje odpowiednia siłę ciągu silników hamujących. Założyć w modelu ograniczoną ilość paliwa, która również wpływa na ciężar sondy.

14. Zastosować algorytm ewolucyjny do uczenia sieci neuronowej, która steruje układem dwóch mas połączonych sprężyną. Założyć, że na wejście sieci podawany jest wektor stanu obiektu, zaś wyjście sieci generuje odpowiednie sterowanie.
15. Wyznaczyć optymalne parametry algorytmów ewolucyjnych dla zadań „benchmarkowych” (Dodatek II) przyjmując odpowiedni rodzaj kodowania, metodę selekcji, operacje krzyżowania i mutację oraz strategię podstawień.

### Dodatek I

Krótkie opisy rozważanych zadań optymalizacji (więcej w cytowanej literaturze):

a) **zadanie załadunku** polega na doborze dla ustalonego zbioru artykułów (rzeczy) wraz z ich wartościami i rozmiarami (lub wagami), takiego podzbioru artykułów, aby suma ich rozmiarów (wag) nie przekraczała zadanego ograniczenia (pojemności lub dopuszczalnej ładowności plecaka) oraz by suma ich wartości była maksymalna.

Problem można zdefiniować w następujący sposób. Niech istnieje plecak o zadanej dopuszczalnej ładowności  $C > 0$  oraz  $N > 0$  artykułów. Każdy  $i$ -ty artykuł posiada wartość  $v_i$  oraz wagę  $w_i$ . Należy znaleźć taki wektor binarny  $\mathbf{x} = [x_1 \ x_2 \ \dots \ x_N]$  ( $x_i = 1$  oznacza wybrany artykuł do plecaka, zaś  $x_i = 0$  reprezentuje brak artykułu w plecaku), aby nienaruszone było następujące ograniczenie

$$\sum_{i=1}^N x_i w_i \leq C$$

oraz by wartość wskaźnika wartości plecaka była maksymalna

$$\sum_{i=1}^N x_i v_i$$

Przyjąć liczbę artykułów  $N$  kolejno 100, 250 i 500. Wagi artykułów wygenerować losowo (z rozkładem równomiernym) z przedziału  $[1,10]$

b) **problem zbierania plonów** definiuje się jako następujące zadanie maksymalizacji

$$J = \sum_{k=0}^{N-1} \sqrt{u_k}$$

przy ograniczeniu będącym równaniem wzrostu

$$x_{k+1} = ax_k - u_k,$$

oraz ograniczeniu równościowym

$$x_0 = x_N$$

gdzie  $x_0$  jest stanem początkowym, a oznacza pewna stałą, zaś  $x_k \in R$ ,  $u_k \in R^+$  reprezentują odpowiednio stan  $i$  (nieujemne) sterowanie. Do rozwiązania zadania przyjąć następujące parametry:  $a = 1.1$ ,  $x_0 = 100$  i  $N = 2, 4, 10, 20, 45$ .

c) **zadanie pchania wózka** określone jest jako problem maksymalizacji całkowitej drogi przebytej w zadanym czasie po odjęciu całkowitego wysiłku. Dyskretny model stanowy opisujący taki problem wyraża się następująco

$$\begin{aligned}x_1(k+1) &= x_2(k), \\x_2(k+1) &= 2x_2(k) - x_1(k) + \frac{1}{N^2}u_k(k),\end{aligned}$$

jako kryterium poszukiwań przyjmuje się następujący wskaźnik jakości sterowania

$$J = x_1(N) - \frac{1}{2N} \sum_{k=0}^{N-1} u^2(k),$$

Zadanie rozwiązać dla następujących parametrów,  $N = 5, 10, 15, 20, 25, 30, 35, 40, 45$

## Dodatek II. Funkcje benchmarkowe.

2.1. Model sfery  $f(x) = \sum_{i=1}^n x_i^2$ .

Dziedzina:  $-100 \leq x_i \leq 100$ ,  $i = 1, 2, \dots, n$ .

2.2. Funkcja Schwefel'a nr. 1  $f(x) = \sum_{i=1}^n |x_i| + \prod_{i=1}^n |x_i|$ .

Dziedzina:  $-10 \leq x_i \leq 10$ ,  $i = 1, 2, \dots, n$ .

2.3. Funkcja Schwefel'a nr.2  $f(x) = \max_i \{|x_i|, 1 \leq i \leq n\}$ .

Dziedzina:  $-100 \leq x_i \leq 100$ ,  $i = 1, 2, \dots, n$ .

2.4. Funkcja Schwefel'a nr.3  $f(x) = -\sum_{i=1}^n \left( x_i \sin(\sqrt{|x_i|}) \right)$

Dziedzina:  $-500 \leq x_i \leq 500$ ,  $i = 1, 2, \dots, n$ .

2.5. Funkcja Schwefel'a nr.4  $f(x) = \sum_{i=1}^n \left( \sum_{j=1}^i x_j \right)^2$

Dziedzina:  $-100 \leq x_i \leq 100$ ,  $i = 1, 2, \dots, n$ .

2.6. Funkcja Rosenbrock'a  $f(x) = \sum_{i=1}^{n-1} \left[ 100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2 \right]$

Dziedzina:  $-30 \leq x_i \leq 30$ ,  $i = 1, 2, \dots, n$

2.7. Funkcja skokowa  $f(x) = \sum_{i=1}^n (\lfloor x_i + 0.5 \rfloor)^2$

Dziedzina:  $-100 \leq x_i \leq 100, i = 1, 2, \dots, n$

2.8. Zaszumiona funkcja czwartego stopnia  $f(x) = \sum_{i=1}^n ix_i^4 + \text{random}[0,1)$

Dziedzina:  $-1.28 \leq x_i \leq 1.28, i = 1, 2, \dots, n$

2.9. Funkcja Rastrigin'a  $f(x) = \sum_{i=1}^n [x_i^2 - 10 \cos(2\pi x_i) + 10]$

Dziedzina:  $-5.12 \leq x_i \leq 5.12, i = 1, 2, \dots, n$

2.10. Funkcja Ackley'a

$$f(x) = -20 \exp\left(-0.2 \sqrt{\frac{1}{n} \sum_{i=1}^n x_i^2}\right) - \exp\left(\frac{1}{n} \sum_{i=1}^n \cos(2\pi x_i)\right) + 20 + \exp(1)$$

Dziedzina:  $-32 \leq x_i \leq 32, i = 1, 2, \dots, n$

2.11. Funkcja Griewank'a  $f(x) = \frac{1}{4000} \sum_{i=1}^n x_i^2 - \prod_{i=1}^n \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$

Dziedzina:  $-600 \leq x_i \leq 600, i = 1, 2, \dots, n$

2.12. Funkcja „wilcze doły”  $f(x) = \left[ \frac{1}{500} + \sum_{j=1}^{25} \frac{1}{j + \sum_{i=1}^2 (x_i - a_{ij})^6} \right]^{-1}$

gdzie  $a_{ij}$  oznaczają elementy następującej macierzy

$$\mathbf{A} = \begin{bmatrix} -32 & -16 & 0 & 16 & 32 & -32 & \dots & 0 & 16 & 32 \\ -32 & -32 & -32 & -32 & -32 & -16 & \dots & 32 & 32 & 32 \end{bmatrix}$$

Dziedzina:  $-65.536 \leq x_i \leq 65.536, i = 1, 2$

2.13. Funkcja Kowalik'a  $f(x) = \sum_{i=1}^{11} \left[ a_i - \frac{x_1(b_i^2 + b_i x_2)}{b_i^2 + b_i x_3 + x_4} \right]^2$

Dziedzina:  $-5 \leq x_j \leq 5, j = 1, 2, 3, 4$

gdzie współczynniki  $a_i, b_i$  przyjmują następujące wartości

$i$	$a_i$	$b_i^{-1}$
1	0.1957	0.25
2	0.1947	0.5
3	0.1735	1
4	0.1600	2
5	0.0844	4
6	0.0627	6
7	0.0456	8
8	0.0342	10
9	0.0323	12
10	0.0235	14
11	0.0246	16

2.14. Funkcja „sześciogarnego wielbłąda”  $f(x) = 4x_1^2 - 2.1x_1^4 + \frac{1}{3}x_1^6 + x_1x_2 - 4x_2^2 + 4x_2^4$

Dziedzina:  $-5 \leq x_i \leq 5, i = 1, 2$

2.15. Funkcja Branin'a  $f(x) = \left(x_2 - \frac{5.1}{4\pi^2}x_1^2 + \frac{5}{\pi}x_1 - 6\right)^2 + 10\left(1 - \frac{1}{8\pi}\right)\cos x_1 + 10$

Dziedzina:  $-5 \leq x_1 \leq 10, 0 \leq x_2 \leq 15$

2.16. Funkcja Goldstein'a-Price'a

$$f(x) = \left[1 + (x_1 + x_2 + 1)^2(19 - 14x_1 + 3x_1^2 - 14x_2 + 6x_1x_2 + 3x_2^2)\right] \cdot \left[30 + (2x_1 - 3x_2)^2(18 - 32x_1 + 12x_1^2 + 48x_2 - 36x_1x_2 + 27x_2^2)\right]$$

Dziedzina:  $-2 \leq x_i \leq 2, i = 1, 2$

2.17. Funkcja Hartman'a nr.1  $f(x) = -\sum_{i=1}^4 c_i \exp\left(\sum_{j=1}^3 a_{ij}(x_i - p_{ij})^2\right)$

gdzie współczynniki  $a_{ij}$ ,  $c_i$  oraz  $p_{ij}$  przyjmują następujące wartości

$i$	$a_{i1}$	$a_{i2}$	$a_{i3}$	$c_i$	$p_{i1}$	$p_{i2}$	$p_{i3}$
1	3	10	30	1	0.3689	0.1170	0.2673
2	0.1	10	35	1.2	0.4699	0.4387	0.7470
3	3	10	30	3	0.1091	0.8732	0.5547
4	0.1	10	35	3.2	0.03815	0.5743	0.8828

Dziedzina:  $0 \leq x_j \leq 1, j = 1, 2, 3$

2.18. Funkcja Hartman'a nr.2  $f(x) = -\sum_{i=1}^4 c_i \exp\left(\sum_{j=1}^6 a_{ij}(x_i - p_{ij})^2\right)$

Dziedzina:  $0 \leq x_j \leq 1, j = 1, 2, 3, 4, 5, 6$

gdzie współczynniki  $a_{ij}$ ,  $c_i$  oraz  $p_{ij}$  przyjmują następujące wartości

$i$	$a_{i1}$	$a_{i2}$	$a_{i3}$	$a_{i4}$	$a_{i5}$	$a_{i6}$	$c_i$	$P_{i1}$	$P_{i2}$	$P_{i3}$	$P_{i4}$	$P_{i5}$	$P_{i6}$
1	10	3	17	3.5	1.7	8	1	0.1312	0.1696	0.5569	0.0124	0.8283	0.5886
2	0.1	0.05	17	0.1	8	14	1.2	0.2329	0.4135	0.8307	0.3736	0.1004	0.9991
3	3	3.5	1.7	10	17	8	3	0.2348	0.1415	0.3522	0.2883	0.3047	0.6650
4	17	8	0.05	10	0.1	14	3.2	0.4047	0.8828	0.8732	0.5743	0.1091	0.0381

2.19. Funkcja Shekel'a nr.1  $f(x) = -\sum_{j=1}^5 \left( \sum_{i=1}^4 (x_i - a_{ij})^2 + c_j \right)^{-1}$

gdzie współczynniki  $a_{ij}$  i  $c_j$  przyjmują następujące wartości

$j$	$a_{1j}$	$a_{2j}$	$a_{3j}$	$a_{4j}$	$c_j$
1	4.0	4.0	4.0	4.0	0.1
2	1.0	1.0	1.0	1.0	0.2
3	8.0	8.0	8.0	8.0	0.2
4	6.0	6.0	6.0	6.0	0.4
5	3.0	7.0	3.0	7.0	0.6
6	2.0	9.0	2.0	9.0	0.6
7	5.0	5.0	3.0	3.0	0.3
8	8.0	1.0	8.0	1.0	0.7
9	6.0	2.0	6.0	2.0	0.5
10	7.0	3.6	7.0	3.6	0.5

Dziedzina:  $0 \leq x_i \leq 10, i = 1,2,3,4$

2.20. Funkcja Shekel'a nr.2  $f(x) = -\sum_{j=1}^7 \left( \sum_{i=1}^4 (x_i - a_{ij})^2 + c_j \right)^{-1}$

Dziedzina:  $0 \leq x_i \leq 10, i = 1,2,3,4$

2.21. Funkcja Shekel'a nr.3  $f(x) = -\sum_{j=1}^{10} \left( \sum_{i=1}^4 (x_i - a_{ij})^2 + c_j \right)^{-1}$

Dziedzina:  $0 \leq x_i \leq 10, i = 1,2,3,4$

2.22. Funkcja Schaffer'a nr.1  $f(x) = 0.5 + \frac{\sin^2(\sqrt{x_1^2 + x_2^2} - 0.5)}{[1 + 0.001(x_1^2 + x_2^2)]^2}$

Dziedzina:  $-100 \leq x_i \leq 100, i = 1,2$

2.23. Funkcja Schaffer'a nr.2  $f(x) = (x_1^2 + x_2^2)^{0.25} [\sin(50(x_1^2 + x_2^2)^{0.1}) + 1]$

Dziedzina:  $-100 \leq x_i \leq 100, i = 1,2$

2.24. Funkcja Shubert'a  $f(x) = -\sum_{i=1}^5 i \cos[(i+1)x_1 + i] \cdot \sum_{i=1}^5 i \cos[(i+1)x_2 + i]$

Dziedzina:  $-10 \leq x_i \leq 10, i = 1,2$



2.25. Funkcja Easom'a  $f(x) = -\cos(x_1)\cos(x_2)\exp(-(x_1 - \pi)^2 - (x_2 - \pi)^2)$   
Dziedzina:  $-100 \leq x_i \leq 100, i = 1, 2$

2.26. Funkcja Bohachevsky'ego nr.1  
 $f(x) = x_1^2 + 2x_2^2 - 0.3\cos(3\pi x_1) - 0.4\cos(4\pi x_2) + 0.7$   
Dziedzina:  $-50 \leq x_i \leq 50, i = 1, 2$

2.27. Funkcja Bohachevsky'ego nr.2  
 $f(x) = x_1^2 + 2x_2^2 - 0.3(\cos(3\pi x_1) + \cos(4\pi x_2)) + 0.3$   
Dziedzina:  $-50 \leq x_i \leq 50, i = 1, 2$

2.28. Funkcja Bohachevsky'ego nr.3  
 $f(x) = x_1^2 + 2x_2^2 - 0.3\cos(3\pi x_1) - \cos(4\pi x_2) + 0.3$   
Dziedzina:  $-50 \leq x_i \leq 50, i = 1, 2$

2.29. Coldville's function  
 $f(x) = 100(x_2 + 2x_1^2)^2 + (1 - x_1)^2 + 90(x_4 + x_3^2)^2 + (1 - x_3)^2 +$   
 $+ 10.1((x_2 - 1)^2 + (x_4 - 1)^2) + 19.8(x_2 - 1)(x_4 - 1)$   
Dziedzina:  $-10 \leq x_i \leq 10, i = 1, 2, 3, 4$

## Literatura

- [1] Arabas J., (2001): Wykłady z algorytmów ewolucyjnych, WNT Warszawa.
- [2] Deb K., Pratap A., Agarwal S., Meyarivan T., (2000). A fast and elitist multi-objective genetic algorithm: NSGA-II, Technical Report, Kanpur Genetic Algorithms Laboratory, Kanpur, India, no. 200001 (PIN 208 016).
- [3] Goldberg D. E., (1995): Algorytmy genetyczne i ich zastosowania, WNT Warszawa.
- [4] Michalewicz Z., (1996): Algorytmy genetyczne + struktury danych = programy ewolucyjne, WNT Warszawa.
- [5] Srinivas N., Deb K., (1994). Multiobjective optimization using nondominated sorting in genetic algorithms, *Evolutionary Computation* 2 (3) 221-248.
- [6] Yager R. R., Filev D. P., (1995) Podstawy modelowania i sterowania rozmytego, WNT, Warszawa.
- [7] Zitzler E., Thiele L., (1998). An evolutionary algorithm for multiobjective optimization: The Strength Pareto Evolutionary Algorithm, Technical Report, Computer Engineering and Networks Laboratory, ETH, Zurich, Switzerland, no. 43.
- [8] Zitzler E., Laumanns M., Thiele L., (2001). SPEA-2: Improving the strength Pareto evolutionary algorithm, Technical Report, Computer Engineering and Networks Laboratory, Department of Electrical Engineering, ETH, Zurich, Switzerland, no. 103.
- [9] Żuranda J., Barski M., Jedruch W., (1996). Sztuczne sieci neuronowe, PWN, Warszawa.