

Andrzej GRZYB

POLITECHNIKA KRAKOWSKA, INSTYTUT POJAZDÓW SZYNOWYCH

Algorytm do polioptymalizacji z odpowiednio przystosowaną metodą ewolucji ekstrapolacyjnej

Dr hab. inż. Andrzej GRZYB

Studia, doktorat i habilitacja na na Wydziale Mechanicznym Politechniki Krakowskiej. Obecnie jest profesorem nadzwyczajnym PK. Zajmuje się mechaniką układów dynamicznych, w szczególności dynamiką pojazdów szynowych i toru, mechaniką i optymalizacją mechanizmów stosowanych w układach napędowych i zawieszaniach pojazdów lądowych oraz opracowywaniem i badaniem algorytmów optymalizacyjnych wykorzystujących ewolucję.



e-mail: agrzyb@mech.pk.edu.pl

Streszczenie

W pracy opisano zmiany dokonane w związku z przystosowaniem optymalizacyjnej metody ewolucji ekstrapolacyjnej – dokładniej przedstawionej we wcześniej opublikowanych pracach autora – do wykonywania polioptymalizacji. Jej celem jest dążenie do wyznaczenia możliwie licznego zbioru paretooptimalnych rozwiązań problemów z wieloma funkcjami kryterialnymi i ograniczeniami. Przykłady zastosowań do zadań testowych, z których wybrane dwa zamieszczono w pracy świadczą o tym, że algorytm pozwala na bardzo efektywne rozwiązywanie problemów testowych i praktycznych oraz uzyskiwać wyniki lepsze od otrzymywanych znanymi już metodami.

Słowa kluczowe: polioptymalizacja, metoda ewolucji ekstrapolacyjnej, MATHEMATICA

Polyoptimization algorithm with appropriately adjusted method of extrapolation evolution

Abstract

The article describes changes introduced for adjustment of optimisation method of extrapolation evolution – presented in details by the author in previously published papers – to realize polyoptimization. Its aim is to determine possibly numerous set of Pareto optimal solutions with multi-criteria functions and constrains. Implementation examples to test tasks, two of them included in the paper, prove that the algorithm enables very effective solving of test and practical problems and achieving better results than obtained by already known methods.

Keywords: polyoptimization, method of extrapolation evolution, MATHEMATICA

1. Wstęp

Celem pracy jest opis algorytmu do optymalizacji wielokryterialnej, czyli polioptymalizacji. Zadany jest wektor funkcji kryterialnych oraz dowolne ograniczenia. Przygotowany algorytm automatycznie wybiera i gromadzi najlepsze wyniki obliczane poprzez wywoływany przez niego kolejny, zasadniczy algorytm z odpowiednio przystosowaną metodą ewolucji ekstrapolacyjnej. Ta ostatnia metoda jest opisana w pracach [1 i 2] i służy do optymalizacji statycznej funkcji wielu zmiennych z dowolnymi ograniczeniami lub bez ograniczeń. Przykłady zastosowań do zadań testowych świadczą o jej szybkiej i dobrej zbieżności do ekstremum globalnego, nawet dla funkcji z wieloma ekstremami lokalnymi i dużą liczbą zmiennych decyzyjnych oraz ograniczeń.

Działanie prawie w pełni opisanego w tej pracy algorytmu – zrealizowanego w programie *MATHEMATICA* – ilustrują zamieszczone dwa przykłady: pierwszy prosty i drugi trudniejszy zaczerpnięty z pracy [3].

2. Opis algorytmu

W opracowanym algorytmie możliwe jest wielokrotne wykonywanie polioptymalizacji zadaną liczbę l_v razy z równoczesnym wybieraniem, sortowaniem i gromadzeniem z kolejno obliczanych populacji końcowych najlepszych wyników. W pierwszej iteracji zapamiętywana jest w odpowiedniej tablicy pełna populacja końcowa, o ile jest poprawna. W następnych iteracjach tablica ta uzupełniana jest o nowe poprawne wyniki, jeżeli takie wyniki są wyznaczone. Gdy ich brak przez kilka iteracji, to obliczenia mogą być przerwane. W każdej iteracji wywoływana jest procedura do polioptymalizacji z algorytmem do wyznaczenia niezdominowanego zbioru punktów populacji dla zadanego wektora l_c funkcji kryterialnych n zmiennych, gdzie $n \geq 1$, z dowolnymi ograniczeniami lub bez ograniczeń. Ta ostatnia procedura realizuje odpowiednio przystosowany do wyznaczenia paretooptimalnych rozwiązań algorytm ewolucji ekstrapolacyjnej, przy czym w każdej iteracji parametry wejściowe powinny być w zadany sposób zmieniane. Najprostszym sposobem zmian może być wymaganie wyboru innego, kolejnego ciągu liczb pseudolosowych poprzez zmianę parametru służącego do startowania generatora tych liczb. Kolejnym sposobem zmian jest zadawanie innego obszaru tworzenia populacji początkowej. Ogólnie, każdy z parametrów wejściowych może być zmieniany. W algorytmie z przystosowaną do polioptymalizacji metodą ewolucji ekstrapolacyjnej przewidziano możliwość wyboru jednej z dwu metod tworzenia populacji startowej, jednej z pięciu strategii dwuetapowych krzyżowań oraz jednej z pięciu postaci dodatkowego składnika funkcji kary uwzględniającego możliwe przekroczenia ograniczeń. Wymagane jest przygotowanie funkcji wektorowej z funkcjami kryterialnymi:

$$\mathbf{c}(\mathbf{x}) = (c_1(\mathbf{x}), c_2(\mathbf{x}), \dots, c_{l_c}(\mathbf{x})), \quad (1)$$

gdzie $\mathbf{x} = (x_1, x_2, \dots, x_n)$ jest wektorem zmiennych decyzyjnych oraz m -elementowej funkcji wektorowej ograniczeń:

$$\mathbf{g}(\mathbf{x}) = (g_1(\mathbf{x}), g_2(\mathbf{x}), \dots, g_m(\mathbf{x})).$$

Mogą być zadane dowolne ograniczenia lub przy $m = 0$ polioptymalizacja może być przeprowadzana bez ograniczeń. Istnieje możliwość zadania wektora \mathbf{i}_{ex} o długości równej liczbie kryteriów l_c , którego wartości elementów odpowiednio informują, czy dla kolejnych funkcji kryterialnych ich polepszanie polegać ma na zwiększaniu, czy też na zmniejszaniu wartości tych funkcji.

Ponieważ algorytm ewolucji ekstrapolacyjnej został opisany w pracach [1 i 2], zatem pominiemy tu opis jego działania, a dalsze informacje będą dotyczyły głównie wprowadzonych w nim zmian związanych z przystosowaniem go do wykonywania polioptymalizacji. Wspomniane zmiany dotyczą przede wszystkim wprowadzenia wewnątrz procedury z algorytmem specjalnie zdefiniowanej funkcji oznaczonej symbolem q – spełniającej rolę funkcji celu w optymalizacji jednokryterialnej i nazywanej dalej zbiorczym kryterium skalarnym – obliczanej na podstawie wektorów (1) z wartościami kryteriów polioptymalizacji w punktach populacji i oceniającą te punkty na podstawie wektorów wartości kolejnych kryteriów. Zadaniem algorytmu jest dążenie do wyznaczenia populacji punktów o liczebności $l_p > 1$ z paretooptimalnymi elementami. Dla ułatwienia opisu przyjmujemy, że polepszanie wartości wszystkich funkcji kryterialnych polega na zmniejszaniu się ich wartości.

Wartość zbiorczego kryterium skalarnego przyporządkowana danemu punktowi \mathbf{x}_k populacji jest sumą względnych ocen elementów wektora $\mathbf{c}_k = \mathbf{c}(\mathbf{x}_k)$ zawierającego wartości kryteriów

optymalizacji w tym punkcie z odpowiednimi elementami wszystkich pozostałych wektorów kryteriów obliczonych dla punktów tej populacji – czyli bez wektora \mathbf{c}_k – których zbiór oznaczamy symbolem \mathbf{C}_k . W realizacji komputerowej algorytmu do tej oceny wykorzystywana jest odpowiednia podprocedura. Uzasadnieniem wprowadzenia nazwy zbiorczego kryterium skalarnego – zamiast funkcji celu – jest to, że zmiana jednego z wektorów zbioru \mathbf{C}_k może spowodować – w przeciwieństwie do zwykłej funkcji celu – nie tylko zmianę wartości tego kryterium dla wektora \mathbf{x}_k , ale także zmiany wielu jej wartości dla pozostałych wektorów ze zbioru \mathbf{C}_k .

Zbiorcze kryterium skalarnie w danym punkcie \mathbf{x}_k populacji jest zatem funkcją dwóch argumentów, z których pierwszy jest wektorem a drugi zbiorem wektorów:

$$q_k = q(\mathbf{c}_k, \mathbf{C}_k) \quad \text{przy } k=1, 2, \dots, l_p, \quad (2)$$

przy czym liczebność populacji l_p jest stała podczas każdej iteracji polioptymalizacyjnej. Jej wartości przyporządkowane są – za pośrednictwem wektorów z wartościami kryteriów – każdemu elementowi zbioru wektorów populacji. Zbiorcze kryterium skalarnie (2) zostało określone tak, że przyjmuje wartości z przedziału domkniętego od 0 do 1, podobnie jak prawdopodobieństwo. Jego definicja jest następująca:

$$q_k = \sum_{j=1, j \neq k}^{l_p} \begin{cases} 0,5 l_c \Delta q & \text{przy } \max(\Delta \mathbf{c}_{jk}) > 0 \wedge \min(\Delta \mathbf{c}_{jk}) < 0 \\ (u_{jk} + 0,5 z_{jk}) \Delta q & \text{przy } \max(\Delta \mathbf{c}_{jk}) \leq 0 \vee \min(\Delta \mathbf{c}_{jk}) \geq 0 \end{cases} \quad (3)$$

gdzie:

$$\Delta q = \frac{1}{l_c(l_p - 1)}, \quad \Delta \mathbf{c}_{jk} = \mathbf{c}_j - \mathbf{c}_k,$$

przy czym u_{jk} oznacza liczbę ujemnych, a z_{jk} liczbę zerowych elementów wektora $\Delta \mathbf{c}_{jk}$.

Elementy: najlepszy i najgorszy w danej populacji, jeżeli takie istnieją, mają na podstawie wzoru (3) oceny q równe odpowiednio 0 i 1. W przypadku, gdy wszystkie punkty populacji stanowią zbiór punktów wzajemnie niezdominowanych, to ich oceny są równe 0,5. Zwykle wtedy w danej iteracji generacja może być uznana za ostatnią w wykonywanej polioptymalizacji. Populacja końcowa, wyznaczona w tej generacji może stanowić podstawę do selekcji rozwiązań i wyznaczenia populacji wynikowej. Selekcja polega na pominięciu tych elementów populacji końcowej, dla których jej punkty są wzajemnie bliższe niż zadana dokładność.

W przypadku dwuetapowych krzyżowań stosowanych w algorytmie ewolucji ekstrapolacyjnej dla danego zasadniczego punktu \mathbf{x} populacji \mathbf{X} w aktualnie przetwarzanej generacji – zgodnie z dokładniejszym opisem podanym w pracy [2] – warunkowo wyznaczany jest nowy punkt \mathbf{x}_c . Następnie należy sprawdzić, czy jest on lepszy od punktu \mathbf{x} . Jak już wyjaśniono zdefiniowana wzorem (3) ocena $q = q(\mathbf{c}, \mathbf{C})$ punktu \mathbf{x} jest trudniejsza, gdyż zależy nie tylko od $\mathbf{c}(\mathbf{x})$ ale zbioru \mathbf{C} wektorów z wartościami kryteriów dla wszystkich pozostałych punktów populacji \mathbf{X} . Ponieważ punkt \mathbf{x}_c nie jest wektorem z \mathbf{X} , to w proponowanym tu – przystosowanym algorytmie – do oceny jego jakości względem punktu zasadniczego przyjęto następujący sposób. Na podstawie obliczonych już wcześniej i dodatkowo dla \mathbf{x}_c wektorów kryteriów dokonywana jest ocena q_c punktu \mathbf{x}_c w pomocniczym zbiorze \mathbf{X}_c , otrzymanym z \mathbf{X} przez zastąpienie w nim punktu \mathbf{x} przez \mathbf{x}_c . W tym celu próbnie – w zbiorze wektorów wartości kryteriów dla całej populacji – wektor $\mathbf{c} = \mathbf{c}(\mathbf{x})$ zostaje zastąpiony przez $\mathbf{c}_c = \mathbf{c}(\mathbf{x}_c)$. Jeżeli ocena q_c jest lepsza niż q , to wektor \mathbf{x}_c zastępuje wektor \mathbf{x} w macierzy \mathbf{X} i następnie obliczane są nowe wartości zbiorczego kryterium skalarnego dla wszystkich punktów aktualnej populacji, a jeżeli ocena q_c nie jest lepsza, to takie zastąpienie i obliczenia nie są wykonywane. Do tej dodatkowej oceny wykorzystywana jest w realizacji komputerowej algorytmu również odpowiednia podprocedura.

3. Przykłady polioptymalizacji

W prostym, własnym przykładzie zakładamy, że mamy zadane dwa kryteria:

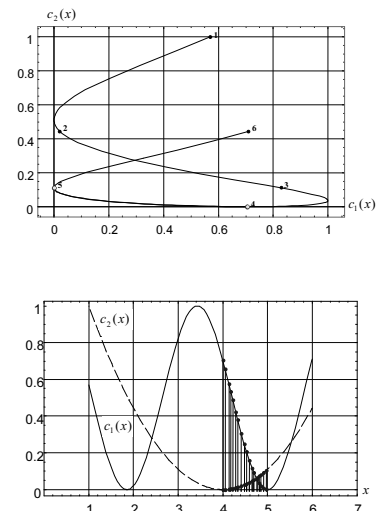
$$c_1(x) = \frac{1 + \sin(2x + 1)}{2} \quad \text{oraz} \quad c_2(x) = \frac{(x - 4)^2}{9}, \quad (4)$$

zależne od jednej zmiennej, czyli $\mathbf{c}(x) = ((c_1(x), c_2(x)))$, których zmniejszenie wartości oznacza polepszenie odpowiadającej im oceny. Przyjmujemy, że zbiorem rozwiązań dopuszczalnych jest przedział domknięty od 1 do 6.

Obliczane przykładowo wartości zbiorczego kryterium skalarnego zawarte są w niżej podanym, posortowanym zbiorze S , którego elementami są wielkości $[x_k, \mathbf{c}_k, q_k]$:

$$S = \{ [1,094, (0,477, 0,939), 0,857], [1,159, (0,412, 0,897), 0,75], [1,245, (0,33, 0,844), 0,607], [2,564, (0,423, 0,229), 0,643], [3,126, (0,912, 0,0848), 0,75], [3,904, (0,789, 0,00102), 0,464], [4,184, (0,529, 0,00375), 0,464], [4,201, (0,511, 0,00449), 0,464], [4,468, (0,255, 0,0243), 0,25], [4,512, (0,218, 0,0291), 0,25], [4,622, (0,135, 0,0429), 0,25], [4,709, (0,0812, 0,0558), 0,25], [5,018, (0,000395, 0,115), 0,286], [5,642, (0,361, 0,3), 0,571], [5,656, (0,374, 0,305), 0,643] \}.$$

W zbiorze tym liczby x_k , gdzie $k = 1, 2, \dots, l_p$, są argumentami stanowiącymi wylosowaną populację startową otrzymaną po przyjęciu $l_p = 15$, wielkości $\mathbf{c}_k = \mathbf{c}(x_k)$ są dwuelementowymi wektorami z wartościami funkcji kryterialnych (4) obliczonymi dla tych argumentów, a $q_k = q(\mathbf{c}_k, \mathbf{C}_k)$ przyporządkowanymi im ocenami (2) zdefiniowanymi wyrażeniem (3). Rozważany przykład rozwiązywany był po przyjęciu $l_p = 25$, przy strategii krzyżowań przyporządkowanej parametrowi $i_c = 11$ oraz prawdopodobieństwach kolejnych etapów krzyżowań odpowiednio: pierwszego $p_{cf} = 0,8$ i drugiego $p_{cs} = 0,3$. W ostatniej, dziesiątej generacji, po kilku sekundach obliczeń wszystkie punkty populacji miały już oceny $q_k = 0,5$.



Rys. 1. Wykresy funkcji (4) z zaznaczonymi paretooptimalnymi rozwiązaniami
Fig. 1. Plot of function (4) with marked Pareto optimal solutions

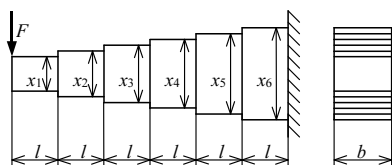
W górnej części rysunku 1 przedstawiony jest wykres parametryczny funkcji kryterialnych w układzie współrzędnych $O c_1 c_2$ z zaznaczonym pogrubioną linią obszarem rozwiązań paretooptimalnych oraz z punktami o naturalnych wartościach argumentu x . Na dolnym rysunku zamieszczone są wykresy funkcji kryterialnych (4); pierwsza narysowana linią ciągłą a druga przerywaną. Łatwo stwierdzić, że rozwiązaniami paretooptimalnymi są tu wszystkie punkty należące do przedziału domkniętego od $x = 4$ do $x = d$, gdzie:

$$d = 1,75\pi - 0,5 \approx 4,9977871.$$

Zauważmy, że w zbiorze S argumenty x_k należące do przedziału od 4 do d lub leżące w jego pobliżu mają najlepsze oceny q_k .

Dodatkowo – pionowymi odcinkami – zaznaczony jest w dolnej części rysunku 1 zestaw paretooptimalnych rozwiązań wyznaczonych w piętnastu iteracjach.

Kolejnym przykładem – pochodzącym z pracy [3] – jest problem polioptymalizacji dotyczący projektowania belki. Schemat utwierdzonej belki o skokowo zmiennym, prostokątnym przekroju, przedstawia rysunek 2.



Rys. 2. Schemat belki
Fig. 2. Scheme of the beam

Wektor zmiennych decyzyjnych:

$$\mathbf{x} = (x_1, x_2, \dots, x_n),$$

gdzie x_j , przy $j = 1, 2, \dots, n$, oznacza wysokość odcinka belki o numerze j . Belka poddana jest zginaniu siłą o wartości F przyłożoną na jej końcu. Dwoma kryteriami podlegającymi minimalizacji są:

I. Objętość belki:

$$c_1(\mathbf{x}) = bl \sum_{j=1}^n x_j.$$

II. Ugięcie w punkcie przyłożenia siły:

$$c_2(\mathbf{x}) = \frac{Fl^3}{3E} \sum_{j=1}^n \frac{3j^2 - 3j + 1}{I_j}, \quad (5)$$

gdzie momenty bezwładności przekrojów:

$$I_j = \frac{bx_j^3}{12}, \quad \text{przy } j = 1, 2, \dots, n.$$

Ograniczeniami są warunki wytrzymałości i geometryczne:

$$\frac{6Flj}{bx_j^2} \leq k_g, \quad \text{przy } j = 1, 2, \dots, n,$$

$$0 \leq x_1, \quad x_j \leq x_{j+1}, \quad x_n \leq d, \quad \text{przy } j = 1, 2, \dots, n-1.$$

Zadanie jest rozwiązywane numerycznie przy następujących danych:

$$n = 6, \quad l = 50 \text{ mm}, \quad b = 50 \text{ mm}, \quad F = 10000 \text{ N}, \\ E = 2,06 \cdot 10^5 \text{ N/mm}^2, \quad k_g = 360 \text{ N/mm}^2, \quad d = 32 \text{ mm}.$$

Zbiorem dopuszczalnych wartości dla każdej zmiennej x_j jest zbiór:

$$D_j = \{12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32\} \text{ mm}, \quad \text{przy } j = 1, 2, \dots, n$$

czyli zadanie stanowi problem programowania dyskretnego i zarazem szczególnie przypadek programowania całkowitoliczbowego. Zbiór rozwiązań dopuszczalnych zadania zawiera ich $11^6 = 1771561$, czyli tyle razy należałoby obliczyć wektor wartości kryteriów przy pełnym przeszukiwaniu dotyczącym wszyst-

kich elementów tego zbioru. Takie przeszukiwanie wymagałoby wielu godzin pracy komputera.

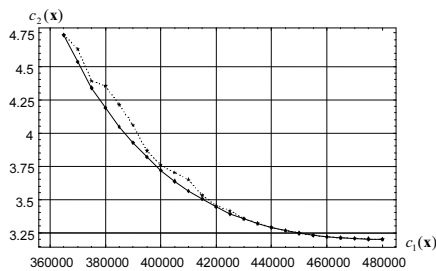
Tab. 1. Zestaw paretooptimalnych rozwiązań dla problemu projektowania belki
Tab. 1. Set of Pareto optimal solutions for the beam design problem

| Numer punktu | Zmienne decyzyjne | Funkcje kryterialne | |
|--------------|---------------------------------------|---------------------|-------------------|
| | $\mathbf{x} = (x_1, x_2, \dots, x_n)$ | $c_1(\mathbf{x})$ | $c_2(\mathbf{x})$ |
| 1 | (32, 32, 32, 32, 32, 32) | $4,8 \cdot 10^5$ | 3,2 |
| 2 | (30, 32, 32, 32, 32, 32) | $4,75 \cdot 10^5$ | 3,203 |
| 3 | (28, 32, 32, 32, 32, 32) | $4,7 \cdot 10^5$ | 3,207 |
| 4 | (26, 32, 32, 32, 32, 32) | $4,65 \cdot 10^5$ | 3,213 |
| 5 | (24, 32, 32, 32, 32, 32) | $4,6 \cdot 10^5$ | 3,22 |
| 6 | (22, 32, 32, 32, 32, 32) | $4,55 \cdot 10^5$ | 3,231 |
| 7 | (20, 32, 32, 32, 32, 32) | $4,5 \cdot 10^5$ | 3,246 |
| 8 | (20, 30, 32, 32, 32, 32) | $4,45 \cdot 10^5$ | 3,268 |
| 9 | (18, 30, 32, 32, 32, 32) | $4,4 \cdot 10^5$ | 3,29 |
| 10 | (18, 28, 32, 32, 32, 32) | $4,35 \cdot 10^5$ | 3,319 |
| 11 | (16, 28, 32, 32, 32, 32) | $4,3 \cdot 10^5$ | 3,355 |
| 12 | (16, 26, 32, 32, 32, 32) | $4,25 \cdot 10^5$ | 3,393 |
| 13 | (16, 24, 32, 32, 32, 32) | $4,2 \cdot 10^5$ | 3,446 |
| 14 | (14, 24, 32, 32, 32, 32) | $4,15 \cdot 10^5$ | 3,504 |
| 15 | (14, 24, 30, 32, 32, 32) | $4,1 \cdot 10^5$ | 3,564 |
| 16 | (14, 22, 30, 32, 32, 32) | $4,05 \cdot 10^5$ | 3,638 |
| 17 | (14, 22, 28, 32, 32, 32) | $4 \cdot 10^5$ | 3,716 |
| 18 | (14, 22, 26, 32, 32, 32) | $3,95 \cdot 10^5$ | 3,821 |
| 19 | (14, 20, 26, 32, 32, 32) | $3,9 \cdot 10^5$ | 3,926 |
| 20 | (14, 20, 26, 30, 32, 32) | $3,85 \cdot 10^5$ | 4,043 |
| 21 | (14, 20, 24, 30, 32, 32) | $3,8 \cdot 10^5$ | 4,186 |
| 22 | (14, 20, 24, 28, 32, 32) | $3,75 \cdot 10^5$ | 4,339 |
| 23 | (14, 20, 24, 28, 30, 32) | $3,7 \cdot 10^5$ | 4,532 |
| 24 | (14, 20, 24, 26, 30, 32) | $3,65 \cdot 10^5$ | 4,736 |

Liczba paretooptimalnych punktów \mathbf{x} zadania wynosi 24. Zaproponowana w pracy metoda pozwoliła na wyznaczenie wszystkich 24 wariantów po obliczeniu tylko 9091 razy wektora wartości kryteriów. Ponadto spośród tych 24 wariantów 12 jest lepszych niż podane w rozwiązaniu wyznaczonym inną metodą i zamieszczonym w pracy [3]. Wyznaczone punkty paretooptimalne zestawione są w tabelicy 1. Lepszymi od podanych w [3] są warianty o numerach od 12 do 23 włącznie. Uzyskiwanie rozwiązań zadania jest trudne, ponieważ w każdym punkcie optymalnym są aktywne ograniczenia. Porównując rozwiązania należy mieć na uwadze szereg usterek zawartych we wspomnianej pracy, zwłaszcza dotyczących wzoru na ugięcia, który powinien być zgodny z (5) i obliczonych wartości ugięć końca belki.

Podany w tabelicy 1 wynik został uzyskany w 5 iteracjach, w czasie 32,3 s na komputerze z procesorem 2,21 GHz, przy zmienianych obszarach, w których były wylosowywane populacje początkowe. Obszary te dobrano wstępnie metodą prób. W szczególności przyjęto $l_p = 30$, strategię krzyżowań oznaczoną parametrem $i_c = 0$ i opisaną w [2], a prawdopodobieństwa dotyczące odpowiednio pierwszego i drugiego z kolejnych etapów rozwiązań wynoszące: $p_{cf} = 0,1$ i $p_{cs} = 0,7$. W kolejnych iteracjach liczby generacji były następujące: 83, 124, 118, 175 i 154.

Na rysunku 3 – w celu porównania wyników – w układzie współrzędnych $O_{c_1 c_2}$ przedstawione są zestawy rozwiązań dla ostatniego przykładu obliczeniowego z wartościami pierwszej funkcji kryterialnej na osi odciętych i drugiej na osi rzędnych. Zaznaczone 24 punkty uzyskane z wykonanej polioptymalizacji połączone są liniami: ciągłą w przypadku punktów zestawionych w tabelicy 1 oraz punktową dla zamieszczonych w pracy [3]. Lepszymi są te rozwiązania, których punkty – przy tych samych odciętych – mają rzędne o mniejszych wartościach.



Rys. 3. Zestawy rozwiązań dla drugiego przykładu obliczeniowego w układzie O_{c_1, c_2}
Fig. 3. Sets of solutions for second numerical example in O_{c_1, c_2} system

4. Uwagi końcowe

Zaproponowany algorytm pozwala na bardzo efektywne rozwiązywanie zadań polioptymalizacji i może być wykorzystany do wielu spotykanych zadań testowych i problemów praktycznych, w szczególności technicznych lub ekonomicznych. Drugi z zamieszczonych przykładów świadczy o możliwości uzyskiwania lepszych wyników niż pozwalają na to dotychczas stosowane metody. Analizy nowych zadań – w tym z wektorami funkcji kryterialnych zawierającymi więcej niż dwa elementy – mogą

pozwoić na uzyskanie informacji stanowiących podstawę do udoskonalenia przedstawionego w pracy algorytmu. Zaproponowana metoda może być również odpowiednio przystosowana a następnie wprowadzona i przetestowana w algorytmach genetycznych lub innych wykorzystujących przetwarzanie w kolejnych generacjach populacji punktów. W szczególności może być przystosowana do metod: ewolucji różnicowej i roju cząstek. Będzie to przedmiotem dalszych prac.

5. Literatura

- [1] Grzyb A.: Ekstrapolacyjny algorytm ewolucyjny. W książce: Tarnowski W., Kiczowski T.: Polioptymalizacja i Komputerowe Wspomaganie Projektowania, Koszalin, 33–46, 2005.
- [2] Grzyb A.: Modyfikacje i porównania algorytmów optymalizacyjnych wykorzystujących ewolucję. W książce: Tarnowski W., Kiczowski T.: Polioptymalizacja i Komputerowe Wspomaganie Projektowania, Koszalin, 72–79, 2006.
- [3] Osyczka A.: Evolutionary Algorithms for Single and Multicriteria Design Optimization. Physica Verlag, A Springer-Verlag Company, Heidelberg 2002.

Artykuł recenzowany

KONFERENCJE NAUKOWO-TECHNICZNE



KONGRES METROLOGII

Drogi Koleżanki i Koledzy Metrologzy,

Serdecznie zapraszamy Państwa do udziału w kolejnym Kongresie Metrologii, który odbędzie się w dniach od 9 do 13 września 2007 roku w Krakowie. Będziecie Państwo Gośćmi Akademii Górniczo-Hutniczej im. Stanisława Staszica, która w roku 2009 obchodzić będzie 90-lecie istnienia. Organizatorem Kongresu jest Katedra Metrologii na Wydziale Elektrotechniki Automatyki Informatyki i Elektroniki, której także "stuknęło" już 50 lat.

Hasło Kongresu brzmi: **Metrologia - narzędziem poznania i droga rozwoju**. W tematyce Kongresu chcemy szczególnie zaakcentować: nowe kierunki rozwoju metrologii, nowe narzędzia poznawania mierzonych procesów, obiektów i sygnałów oraz nowe obszary zastosowań metrologii. W naszym zamiarze nowość oraz interdyscyplinarność i oryginalność tematyki powinna być głównym wyróżnikiem naszych prac, które po recenzjach opublikujemy w materiałach Kongresu. Liczymy na mobilizację Państwa energii w przygotowaniu interesujących i odważnych tematów wystąpień oraz na Państwa udział w ożywionej i rzeczowej dyskusji. Tematyka Kongresu przedstawia się następująco:

I. Współczesne problemy metrologii

1. Mikrosystemy pomiarowe; zagadnienia sprzętowe i projektowe, nowe technologie
2. Systemy rozproszone i bezprzewodowe; interfejsy i protokoły, kompresja i transmisja danych
3. Fuzja danych pomiarowych; identyfikacja modeli złożonych procesów i obiektów
4. Metody i algorytmy analizy danych
5. Nowe problemy przetwarzania a/c; granice szybkości i rozdzielczości
6. Czujniki i przetworniki z modulacją światła, czujniki elektrochemiczne i inne

II. Nowe metody pomiarowe w zastosowaniach

1. Pomiar biomedyczne; metody diagnostyki i analizy medycznej
2. Pomiar konwencjonalnych i nie konwencjonalnych źródeł i przetworników energii

3. Pomiar technologiczne i transportowe; obiekty "inteligentne"
4. Pomiar środowiskowe; rozpoznawanie stanów zagrożenia
5. Pomiar i diagnostyka obiektów mechanicznych, pomiar akustyczne
6. Pomiar w zastosowaniach militarnych

III. Współczesne problemy podstaw metrologii; dydaktyka metrologii

1. Teoria i modelowanie systemów pomiarowych
2. Wzorce i wielkości odniesienia; pomiar dokładne, wzorcowanie i metrologia prawna
3. Pomiar kwantowe, wzorce kwantowe
4. Błędy, niepewności, wrażliwość
5. Dydaktyka metrologii; plany studiów, nowe treści w podręcznikach

Mamy nadzieję, że rozproszeni w różnych branżach znajdziemy chwilę czasu na spotkanie przy tematyce, która nas łączy. To jest główny cel, dla którego Kongres organizujemy.

Jest jednak i cel drugi, równie istotny. Ten cel - to Kraków, miasto stare i tajemnicze, po którym krążą duchy przeszłości i w którym mieszka geniusz tego Narodu. Chcemy dać Państwu okazję do dotknięcia ponad 600-letnich murów krakowskiej Almae Matris - wszak my wszyscy z Niej! Chcemy dać okazję do spaceru ulicami, po których chodził Kopernik oraz do chwili zadumy nad przeszłością w komnatach Wawelu.

Jest i cel trzeci, bo przecież nie tylko poważną strawą duchową człowiek, a zwłaszcza metrolog żyje. Zachęcamy Państwa do udziału w spotkaniach o charakterze towarzyskim.

Jesteśmy przekonani, że w ciągu tych dni kongresowych nudzić się Państwo nie będziecie. Przeciwnie - zyskacie Państwo nie tylko satysfakcję zawodową, obywatelską i towarzyską, ale i miłe wspomnienia.

Zatem - do zobaczenia!

Przewodniczący Komitetu Naukowego
Kongresu Metrologii 2007

Prof. Michał Szyper

Przewodniczący Kongresu

Prof. Janusz Gajda